

z/OS



# JES2 Data Areas Volume 4



z/OS



# JES2 Data Areas

## Volume 4

**Note**

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 377.

**First Edition, September, 2013**

This edition applies to Version 2 Release 1 of z/OS (5650-ZOS) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 1988, 2013. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>About this information</b> . . . . .	v	<b>\$RDRWORK Information</b> . . . . .	187
Who should use this information . . . . .	v	<b>\$RECY Information</b> . . . . .	193
How to use this information . . . . .	v	<b>\$REQJID Information</b> . . . . .	195
The header . . . . .	v	<b>\$RESNAM Information</b> . . . . .	197
Data area map . . . . .	vii	<b>\$RESWORK Information</b> . . . . .	201
Cross reference . . . . .	viii	<b>\$RJCB Information</b> . . . . .	207
<b>Programming interface information</b> . . . . .	ix	<b>\$ROTT Information</b> . . . . .	211
<b>\$OPAWORK Information</b> . . . . .	1	<b>\$SAFINFO Information</b> . . . . .	215
<b>\$OUTWORK Information</b> . . . . .	3	<b>\$SAPID Information</b> . . . . .	219
<b>\$PAD Information</b> . . . . .	7	<b>\$SBWA Information</b> . . . . .	225
<b>\$PADDR Information</b> . . . . .	9	<b>\$SCAND Information</b> . . . . .	233
<b>\$PARMLST Information</b> . . . . .	29	<b>\$SCANWA Information</b> . . . . .	235
<b>\$PARMWRK Information</b> . . . . .	75	<b>\$SCAT Information</b> . . . . .	249
<b>\$PBLK Information</b> . . . . .	79	<b>\$SCID Information</b> . . . . .	253
<b>\$PCE Information</b> . . . . .	83	<b>\$SCK Information</b> . . . . .	257
<b>\$PCL Information</b> . . . . .	91	<b>\$SCT Information</b> . . . . .	261
<b>\$PCT Information</b> . . . . .	99	<b>\$SDB Information</b> . . . . .	265
<b>\$PCTAB Information</b> . . . . .	105	<b>\$SFRB Information</b> . . . . .	277
<b>\$PDDB Information</b> . . . . .	109	<b>\$SFRWORK Information</b> . . . . .	281
<b>\$PERFCB Information</b> . . . . .	117	<b>\$SFSWORK Information</b> . . . . .	285
<b>\$PIT Information</b> . . . . .	129	<b>\$SIG Information</b> . . . . .	289
<b>\$PPPWORK Information</b> . . . . .	133	<b>\$SJB Information</b> . . . . .	291
<b>\$PQE Information</b> . . . . .	143	<b>\$SJJOB Information</b> . . . . .	305
<b>\$PREBERT Information</b> . . . . .	147	<b>\$SJXB Information</b> . . . . .	311
<b>\$PRGWORK Information</b> . . . . .	151	<b>\$SMF Information</b> . . . . .	323
<b>\$PSO Information</b> . . . . .	155	<b>\$SNFWORK Information</b> . . . . .	371
<b>\$PSOWORK Information</b> . . . . .	161	<b>\$SPIWORK Information</b> . . . . .	373
<b>\$PSV Information</b> . . . . .	165	<b>Notices</b> . . . . .	377
<b>\$QSE Information</b> . . . . .	169		
<b>\$RAT Information</b> . . . . .	175		
<b>\$RCPWORK Information</b> . . . . .	179		



---

## About this information

This information is a graphic presentation of many data areas used by the z/OS operating system and by application programs. The data areas are one or more of the following:

- Programming interfaces
- Needed for debugging or diagnosis.

This information supports z/OS (5650-ZOS).

---

## Who should use this information

This information is for system programmers who diagnose and debug operating system and programming problems. It provides information for debugging installation-provided programs or diagnosing IBM-provided programs. The user of this information should have a working knowledge of the functions and logic of the operating system.

---

## How to use this information

Data areas are sequenced alphanumerically by data area acronym. Each data area has up to four sections:

- Programming Interface Information
- Header
- Data area map
- Cross-reference, if the data area map is long enough

## The header

The header includes some or all of the following:

<b>Common Name:</b>	The descriptive name of the data area.
<b>Macro ID:</b>	The name of the mapping macro for the data area. Mapping macros can be issued in programs to generate a copy of the data area.
<b>DSECT Name:</b>	Name of the DSECT (dummy control section) created by the mapping macro.
<b>Owning Component:</b>	Component name and component identifier in parentheses.
<b>Eye-Catcher ID:</b>	Character string identifier of the eye-catcher (sometimes called the <b>control block id</b> ) within the mapping macro. The offset and length of the eye-catcher are also included.
<b>Storage Attributes:</b>	The storage attributes of the data area, including the following: <ul style="list-style-type: none"><li><b>Main Storage:</b> Central storage attributes of the data area.</li><li><b>Virtual Storage:</b> Virtual storage attributes of the data area.</li><li><b>Auxiliary Storage:</b> Spool storage attributes of the data area.</li><li><b>Subpool and Key:</b> Subpool is the area of virtual storage that contains the data area. Key is the storage protect key for the storage represented by the data area.</li></ul>
<b>Size:</b>	The size of the data area in decimal bytes.
<b>Created by:</b>	Module, macro, or component whose use creates the data area.
<b>Pointed to by:</b>	Registers or data area fields that contain the address of the data area.
<b>Serialization:</b>	Method used to ensure that one user does not update a data area that is being updated or used by another user. The most common methods used for serialization are: <ul style="list-style-type: none"><li>• Lock or locks</li><li>• ENQ and DEQ macros</li><li>• Compare and Swap (CS) instruction</li></ul>

- Disablement, which is disabling interruptions by setting bits in the program status word (PSW) of the program using the data area

**Function:**

Brief description of the use of the data area.



## Data area map

The data area is described field by field. These field descriptions are taken directly from the system code.

The following is an example of the field descriptions for the ANYAREA data area:

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	384	ANYAREA	
0	(0)	CHARACTER		ANYBEGIN	BEGINNING OF ANYAREA
0	(0)	CHARACTER	4	ANYACRO	ACRONYM IN EBCDIC 'ANY '
4	(4)	ADDRESS	4	ANYADDR	ADDRESS OF NEXT ANYAREA ON QUEUE

For each field in the data area, the data area map provides the following information:

**Offsets** The address of the field, shown in both decimal (DEC) and hexadecimal (HEX in parentheses), relative to the beginning of the data area.

**Type** The kind of program data defined for this field, as follows:

Type	Description
ADDRESS	Address constant
BITSTRING	Bitstring constant
CHARACTER	Character value
DBL WORD	Double word boundary
FIXED	Arithmetic signed or unsigned value
HEX	Hexadecimal value
SIGNED	Arithmetic signed value
STRUCTURE	Level 1 control block name
UNSIGNED	Unsigned value

**Len** Size of the field in decimal bytes.

**Name (Dim)** The name of the field, bit, or mask.

Bit or mask names are preceded by a description of bit position and value, as follows:

1... ....	Refers to bit 0.
.... ..11	Refers to bits 6 and 7.
...1 ....	Refers to bit 3.
11.. 1111	Refers to bits 0, 1, 4, 5, 6, and 7.

**Description** A description of the purpose or meaning of the field, bit, or mask.

## Cross reference

For each data area with more than 10 fields, the cross reference shows the following:

<b>Name</b>	The name of the field, bit, or mask.
<b>Hex Offset</b>	The hexadecimal offset of the field into the data area. For bits, the hexadecimal offset of the field containing the bit.
<b>Hex Value</b>	Values are shown only for bits, equates, and initialized character strings. For bits, the hexadecimal value shown implies the position of the bit in the field containing the bit.

Bit ANYBIT in the following illustration shows how to use the hexadecimal value. In the Example, cross reference for the ANYBIT bit looks like this:

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
ANYBIT	F0	80

In the map of the data area, the ANYBIT bit appears like this:

240	(F0)	FIXED	4	ANYWORD	CONTROL WORD
240	(F0)	BITSTRING	1	ANYBYTE	FLAG BYTE
		1... ..		ANYBIT	"X'80'" BIT ON MEANS THIS . . .

X'F0' is the offset of field ANYWORD into the data area. ANYWORD is a 4-byte field, which contains a 1-byte field named ANYBYTE. Both ANYWORD and ANYBYTE have the same offset. The first bit in both fields is named ANYBIT. Ignoring the other bits in the field ANYBYTE, if the ANYBIT bit is on, the value of field ANYBYTE would be 1000 0000, which is equivalent to X'80'. This value (X'80') is shown both in the Description in the data area map and in the column of the cross reference.

---

## Programming interface information

This document contains information NOT intended to be used as programming interfaces of z/OS.

This document also contains intended programming interfaces that allow the customer to write programs to obtain the services of z/OS.

This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

Programming Interface information
End of Programming Interface information

Unless otherwise specified, for data areas classified as programming interfaces, the **MACRO ID** and **DSECT NAME(S)** in the header are part of the programming interface. **ALL** other header information is included for diagnostic purposes **ONLY**.

Since a *data area name* that is designated as part of the programming interface is one of the following:

- MACRO ID
- DSECT NAME
- commonly-used name

before including the *data area name* in a program, refer to the data area header for the applicable **MACRO ID**.

If only certain fields in a data area are intended or not intended for use as a programming interface, the specific field name(s) are differentiated within the data area.

For data areas classified as programming interfaces, "RESERVED FOR USER" fields are part of the interface; all other "**RESERVED ...**" fields are **NOT** part of the interface.

For a field that is part of the programming interface, the only information that is part of the interface for writing programs is:

- field name
- data type
- field length
- description (purpose or allowed values)

**INCLUDE ONLY** data area: **ONLY** the MACRO ID is the programming interface. The DSECT NAME, constants, and data area itself are **NOT** part of the programming interface.

**TOKEN ONLY** data area: **ONLY** the address of the data area is a programming interface. The DSECT NAME, constants, and data area itself are **NOT** part of the programming interface.



## \$OPAWORK Information

### \$OPAWORK Heading Information

**Common Name:** JES2 Output Priority Aging PCE Work Area  
**Macro ID:** \$OPAWORK  
**DSECT Name:** PCE (\$OPAWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4  
**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE  
**Size:** See symbol OPAPCEWS for the length of this work area. The overall length of the PCE is stored in field PCELENG.  
**Created by:** See \$PCE  
**Pointed to by:** The \$PRYOPCE field of the \$HCT data area  
 See \$PCE for other pointer fields that apply to all PCE types.  
**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this area are used by a JES2 Output Priority Aging Processor and by its support routines and exits. \$OPAWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$OPAWORK are actually part of the PCE DSECT, but only map PCEs with the value PCEOPAID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$OPAWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	BITSTRING	12	OPATQE	HASP Timer Queue Element
324	(144)	SIGNED	4		Reserved for future use
328	(148)	DBL WORD	8	(0)	Force double-word alignment
328	(148)	X'10'	0	OPAPCEWS	**-PCEWORK" Length of work area

## \$OPAWORK Map

---

**\$OUTWORK Information**

**\$OUTWORK Programming Interface information**

Programming Interface information

**\$OUTWORK**

End of Programming Interface information

## Heading Information • \$OUTWORK Map

### \$OUTWORK Heading Information

**Common Name:** JES2 Output PCE Work Area  
**Macro ID:** \$OUTWORK  
**DSECT Name:** PCE (\$OUTWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol OUTWKSIZ for the length of this work area. The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE  
**Pointed to by:** The \$OUTPCE field of the \$HCT data area points into the \$PCEORG/\$PCELAST chain to the first Output PCE. Since this chain contains all PCEs, use the PCEID field to determine when you have chained past the last PCE of this type. See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this work area are used by a JES2 Output Processor and by its support routines and exits. \$OUTWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$OUTWORK are actually part of the PCE DSECT, but only map PCEs with the value PCEOUTID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$OUTWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	SIGNED	4	OUTIOTBF	ADDRESS OF IOT BUFFER CHAIN
316	(13C)	SIGNED	4	OUTJCTBF	ADDRESS OF JCT BUFFER
320	(140)	SIGNED	4	OUTPDDB	RESTART PDDB POINTER
324	(144)	SIGNED	4	OUTIMEON (2)	OUTPUT PROCESSOR TIME/DATE
332	(14C)	ADDRESS	4	OUTJOAA	Addr of buffer containing PROTOTYPE JOA
336	(150)	SIGNED	4	OUTJOAL	Length of buffer containing PROTOTYPE JOA
340	(154)	SIGNED	4	OUTDBEND	1ST FREE PDDB SLOT IN IOT
344	(158)	SIGNED	4	OUTIOT	RESTART IOT ADDRESS
348	(15C)	SIGNED	4	OUTIOTM	Number of IOTs in memory
352	(160)	SIGNED	4	OUTJBKEY	JOB KEY FROM JCTJBKEY
356	(164)	BITSTRING	1	OUTJCOPY	JOB LEVEL COPY COUNT FROM JCT
357	(165)	BITSTRING	2		RESERVED
359	(167)	BITSTRING	1	OUTFLAGS	OUTPUT PROCESSOR FLAGS
360	(168)	SIGNED	4	OUTGGTOK	GENERIC GROUPING TOKEN
364	(16C)	SIGNED	4	OUTEXPRM (0)	EXIT 16 PARAMETER LIST
364	(16C)	SIGNED	4	OUTMADD	EXIT MESSAGE ADDRESS
368	(170)	SIGNED	4	OUTMPRM	EXIT PARM LIST ADDRESS
372	(174)	SIGNED	4	OUTMJCT	ADDRESS OF JCT
376	(178)	ADDRESS	4	OUTDSSCB	ADDR OF DSSCB WORK AREA



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
380	(17C)	CHARACTER	32	OUTGRPPM	OUTPUT GROUPING PARM LIST
412	(19C)	ADDRESS	4	OUTQPARM	NODE TABLE ADDRESS
416	(1A0)	ADDRESS	4		CONTROL BLOCK ADDRESS
420	(1A4)	ADDRESS	4		ADDRESS OF JQE
424	(1A8)	ADDRESS	1		QUEUE TYPE SPECIFIED
425	(1A9)	ADDRESS	1		WORK SELECTION TYPE FLAG
426	(1AA)	ADDRESS	1		RESERVED FOR FUTURE USE
426	(1AA)	X'19C'	0	OUTPLST	"OUTQPARM,*-OUTQPARM" QGET PARAMETER LIST STORAGE
427	(1AB)	CHARACTER	37	OUTNOTPL	Parm list storage for \$HNOTIFY call from DSAL
464	(1D0)	DBL WORD	8	(0)	
464	(1D0)	BITSTRING	80	OUTCTKNO	Old CTOKEN work area
544	(220)	DBL WORD	8	(0)	
544	(220)	BITSTRING	56	OUTX40PL	Exit 40 XPL parmlist
600	(258)	DBL WORD	8	(0)	
600	(258)	X'120'	0	OUTWKSIZ	**-"PCEWORK" LENGTH OF HOPE PCE WORK AREA

Comment

OUTFLAGS

End of Comment

1... ..	OUTSTATS	"B'10000000" JOB Statistics created
.... 1...	OUTJOBBER	"B'00001000" Job finished abnormally

**\$OUTWORK Cross Reference**

Name	Hex Offset	Hex Value
OUTCTKNO	1D0	
OUTDBEND	154	
OUTDSSCB	178	
OUTEXPRM	16C	
OUTFLAGS	167	
OUTGGTOK	168	
OUTGRPPM	17C	
OUTIMEON	144	
OUTIOT	158	
OUTIOTBF	138	
OUTIOTM	15C	
OUTJBKEY	160	
OUTJCOPY	164	
OUTJCTBF	13C	
OUTJOAA	14C	
OUTJOAL	150	
OUTJOBBER	258	8
OUTMADD	16C	
OUTMJCT	174	
OUTMPRM	170	
OUTNOTPL	1AB	
OUTPDDB	140	
OUTPLST	1AA	19C
OUTQPARM	19C	
OUTSTATS	258	80
OUTWKSIZ	258	120
OUTX40PL	220	
PCE	0	

## \$OUTWORK Cross Reference

## \$PAD Information

### \$PAD Heading Information

**Common Name:** PROCLIB Allocation Descriptor  
**Macro ID:** \$PAD  
**DSECT Name:** PAD  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** PAD  
 Offset: PADID  
 Length: L'PADID  
**Storage Attributes:** Subpool: N/A  
 Key: 1  
 Residency: Located in the PAD JES2 \$CPOOL in the PSO data space.  
**Size:** See PADLEN  
**Created by:** HASPSXIT for the PROCLIB command and init statement  
**Pointed to by:** CCTPAD field of the HCCT data area  
 PADPAD field of the PAD data area  
 PADALT field of the PAD data area  
 PADDAD field of the PAD data area  
**Serialization:** None required  
**Function:** The PAD represent a dynamically allocated PROCLIB DD statement.

### \$PAD Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PAD	, PROCLIB allocation DSECT
0	(0)	CHARACTER	4	PADID	Eyecatcher
4	(4)	SIGNED	4	PADSIZE	PAD length
8	(8)	CHARACTER	8	PADNAME	Logical DD name
16	(10)	CHARACTER	8	PADALCDD	Allocated DD name
24	(18)	DBL WORD	8	PADCRTIM	PAD creation time
32	(20)	DBL WORD	8	PADDLTIM	Time of last PAD deletion attempt
40	(28)	BITSTRING	1	PADFLAG1	General flag byte
		1... ....		PAD1DEL	"B'10000000" PAD deleted
		.1.. ....		PAD1UNC	"B'01000000" Unconditional allocation
		..1. ....		PAD1COND	"B'00100000" Conditional allocation
		...1 ....		PAD1UNAP	"B'00010000" Unallocation in progress
		.... 1..		PAD1STAT	"B'00001000" PAD represents a static allocation (from PROC)
		.... .1..		PAD1CIFA	"B'00000100" A C/I addr space failed to allocate PROCLIB
41	(29)	BITSTRING	3		Reserved
44	(2C)	ADDRESS	4	PADPAD	PAD chain pointer
48	(30)	ADDRESS	4	PADALT	Alternate PAD chain
52	(34)	ADDRESS	4	PADDAD	Owning (main) pad
56	(38)	SIGNED	4	PADUSE	PAD use count
60	(3C)	SIGNED	4	PADDSNLW	Low data set subscript (always 1)
64	(40)	SIGNED	4	PADDSNCT	Data set count
68	(44)	BITSTRING	12		Reserved
80	(50)	DBL WORD	8	PADDSET (0)	Data set specifications
80	(50)	BITSTRING	15304	(255)	Max data set specifications
15384	(3C18)	DBL WORD	8	PADCIALC (0)	C/I address space allocs
15384	(3C18)	BITSTRING	0	(0)	Max C/I alloc areas
15584	(3CE0)	DBL WORD	8	(0)	Ensure alignment
15584	(3CE0)	X'3CE0'	0	PADLEN	"*-PAD" Maximum PAD length

## \$PAD Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PADE	, PAD data set entry
0	(0)	CHARACTER	44	PADEDSN	Data set name
44	(2C)	CHARACTER	8	PADEUNIT	Data set unit
52	(34)	CHARACTER	6	PADEVOL	Data set VOLSER
58	(3A)	BITSTRING	1	PADEFLG1	Flag byte
		1... ....		PADE1ALF	"B'10000000" Data set not allocated (failed)
59	(3B)	BITSTRING	1		Reserved
60	(3C)	SIGNED	4	(0)	Align
60	(3C)	X'3C'	0	PADELEN	"*-PADE" Length of data set entry

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PADA	, PAD data set entry
0	(0)	ADDRESS	4	PADACICB	Owning CICB address (Zero if not in use)
4	(4)	BITSTRING	1	PADAFLG1	Flag byte (set by AS)
		1... ....		PADA1ALC	"B'10000000" DD name is allocated
		.1... ....		PADA1ALF	"B'01000000" Allocation has failed
5	(5)	BITSTRING	3		Reserved
8	(8)	DBL WORD	8	(0)	Align
8	(8)	X'8'	0	PADALEN	"*-PADA" Length of data set entry

## \$PAD Cross Reference

Name	Hex Offset	Hex Value
PAD	0	
PADA	0	
PADACICB	0	
PADAFLG1	4	
PADALCDD	10	
PADALEN	8	8
PADALT	30	
PADA1ALC	4	80
PADA1ALF	4	40
PADCIALC	3C18	
PADCRTIM	18	
PADDAD	34	
PADDLTIM	20	
PADDSET	50	
PADDSNCT	40	
PADDSNLW	3C	
PADE	0	
PADEDSN	0	
PADEFLG1	3A	
PADELEN	3C	3C
PADEUNIT	2C	
PADEVOL	34	
PADE1ALF	3A	80
PADFLAG1	28	
PADID	0	D7C1C440
PADLEN	3CE0	3CE0
PADNAME	8	
PADPAD	2C	
PADSIZE	4	
PADUSE	38	
PAD1CIFA	28	4
PAD1COND	28	20
PAD1DEL	28	80
PAD1STAT	28	8
PAD1UNAP	28	10
PAD1UNC	28	40

## \$PADDR Information

### \$PADDR Heading Information

**Common Name:** Private Storage Routine Address Table/DSECT  
**Macro ID:** \$PADDR  
**DSECT Name:** PADDR  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PADR'  
 Offset: PADDRID-PADDR  
 Length: 4

**Storage Attributes:** Subpool: The subpool of the HASJES20 load module  
 Key: 1  
 Residency: Virtual and real storage are below 16M, in the private storage of the JES2 address space.

**Size:** See PADDRLEN  
**Created by:** The \$PADDR is created by assembly of the HASPNUC module in the HASJES20 load module.

**Pointed to by:** \$PADDR field of the \$HCT data area  
**Serialization:** Read only, except for JES2 initialization processing for PC routines

**Function:** The PADDR contains the addresses of all JES2 private storage service routines to which access is required from multiple assembly modules or installation exits.

This table may be used by \$CALL to locate routines residing in private storage in the JES2 address space. \$CALL uses this table to find either the address or PC number for the called routine.

This macro has a DSECT= parameter. If DSECT=YES is used, the DSECT is generated, otherwise the table is expanded.

### \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PADDR	JES2 private storage routine address table DSECT
0	(0)	CHARACTER	4	PADDRID	PADDR TABLE EYECATCHER
4	(4)	ADDRESS	1	PADDRV	VERSION NUMBER
4	(4)	X'6'	0	PADDRVN	"6" VERSION NUMBER
5	(5)	BITSTRING	3		RESERVED FOR FUTURE USE

Comment

Module HASCOFST entries, listed alphabetically  
 (for the copy of HASCOFST that is within HASJES20)

End of Comment

8	(8)	ADDRESS	4	PADDR@OCOOFST	"V(OCOOFST)" Offset table for O C O code (data only, not \$CALLable) O C O code cannot use this PADDR field, as the PADDR is not frozen.
---	-----	---------	---	---------------	--

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Entry addresses for \$EXTP services (R14 is used for service options.)					
End of Comment					
12	(C)	ADDRESS	4	P@HASPBCA	"V(HASPBCA)" Entry to BSC \$EXTP routines
16	(10)	ADDRESS	4	P@HASPROUT	"V(HASPROUT)" Entry to NJE job route srv.
20	(14)	ADDRESS	4	P@HASPSNAA	"V(HASPSNAA)" Entry to SNA \$EXTP routines
24	(18)	ADDRESS	4	P@HASPTCPA	"V(HASPTCPA)" Entry to TCP \$EXTP routines
28	(1C)	ADDRESS	4	P@HASPXFRA	"V(HASPXFRA)" Entry to XFR \$EXTP routines
Comment					
Entry addresses for Line manager scan routines					
End of Comment					
32	(20)	ADDRESS	4	P@HASPBPPO	"V(HASPBPPO)" BSC Buffer channel end
36	(24)	ADDRESS	4	P@HASPBACT	"V(HASPBACT)" BSC Active line scan
40	(28)	ADDRESS	4	P@HASPBUPT	"V(HASPBUPT)" BSC Inactive line scan
44	(2C)	ADDRESS	4	P@HASPBSLN	"V(HASPBSLN)" BSC Secondary started line scan for SWEL processing
48	(30)	ADDRESS	4	P@HASPSPRO	"V(HASPSPRO)" SNA RPL Completion
52	(34)	ADDRESS	4	P@HASPSLOG	"V(HASPSLOG)" SNA Active logon scan
56	(38)	ADDRESS	4	P@HASPSLNE	"V(HASPSLNE)" SNA Active line scan
60	(3C)	ADDRESS	4	P@HASPSIDL	"V(HASPSIDL)" SNA Idle line scan
64	(40)	ADDRESS	4	P@HASPSUNT	"V(HASPSUNT)" SNA Inactive line scan
68	(44)	ADDRESS	4	P@HASPSACB	"V(HASPSACB)" SNA ACB completion scan
72	(48)	ADDRESS	4	P@HASPSICE	"V(HASPSICE)" SNA ICE scan
76	(4C)	ADDRESS	4	P@HASPSRAT	"V(HASPSRAT)" SNA RAT Autologon scan
80	(50)	ADDRESS	4	P@HASPSSAL	"V(HASPSSAL)" Sna Secondary started line scan for SWEL processing
84	(54)	ADDRESS	4	P@HASPTPRO	"V(HASPTPRO)" TCP/IP buffers queued to main task
88	(58)	ADDRESS	4	P@HASPTACT	"V(HASPTACT)" TCP/IP Active line scan
92	(5C)	ADDRESS	4	P@HASPTIDL	"V(HASPTIDL)" TCP/IP Idle line scan
96	(60)	ADDRESS	4	P@HASPTUNT	"V(HASPTUNT)" TCP/IP Inactive unit scan
100	(64)	ADDRESS	4	P@HASPTASV	"V(HASPTASV)" TCP/IP Active server scan
104	(68)	ADDRESS	4	P@HASPTSSV	"V(HASPTSSV)" TCP/IP Starting server scan
108	(6C)	ADDRESS	4		Reserved
112	(70)	ADDRESS	4		Reserved
116	(74)	ADDRESS	4		Reserved
120	(78)	ADDRESS	4		Reserved
124	(7C)	ADDRESS	4	P@MLMVFY	"V(MLMVFY)" MLLM Verification code
Comment					
Module HASPARMO routines listed alphabetically					
End of Comment					
128	(80)	ADDRESS	4	P@ARODREG	"V(ARODREG)" Deregister job
132	(84)	ADDRESS	4	P@AROQRYA	"V(AROQRYA)" Query registration
Comment					
Module HASPBSC routines listed alphabetically					
End of Comment					
136	(88)	ADDRESS	4	P@MPURIO	"V(MPURIO)" PURGE I/O on line

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Module HASPCDYN routines listed alphabetically					
End of Comment					
140	(8C)	ADDRESS	4	P@\$CDCTDYN	"V(\$CDCTDYN)" Common DCT CREATE/SYNCH
144	(90)	ADDRESS	4	P@\$CNITNOT	"V(\$CNITNOT)" Common NIT broadcast
Comment					
Module HASPCFAL routines listed alphabetically					
End of Comment					
148	(94)	ADDRESS	4	P@CFALOC	"V(CFALOC)" CF Allocate a structure
Comment					
Module HASPCFBF routines listed alphabetically					
End of Comment					
152	(98)	ADDRESS	4	P@CFBLDLST	"V(CFBLDLST)" CF Build list for writing
Comment					
Module HASPCFDE routines listed alphabetically					
End of Comment					
156	(9C)	ADDRESS	4	P@CFDELETE	"V(CFDELETE)" CF Delete all elements
Comment					
Module HASPCFE routines listed alphabetically					
End of Comment					
160	(A0)	ADDRESS	4	P@CFCOMP	"V(CFCOMP)" CF Complete Exit
164	(A4)	ADDRESS	4	P@CFEVEN	"V(CFEVENT)" CF Event Exit
168	(A8)	ADDRESS	4	P@CFNOTIFY	"V(CFNOTIFY)" CF Notify Exit
Comment					
Module HASPCFFC routines listed alphabetically					
End of Comment					
172	(AC)	ADDRESS	4	P@CFFCOMP	"V(CFFCOMP)" CF Force completion
Comment					
Module HASPCFLE routines listed alphabetically					
End of Comment					
176	(B0)	ADDRESS	4	P@CFRDLEC	"V(CFRDLEC)" CF Read the LECs
Comment					
Module HASPCFMT routines listed alphabetically					
End of Comment					
180	(B4)	ADDRESS	4	P@CFFORMAT	"V(CFFORMAT)" CF Format

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Module HASPCFQL routines listed alphabetically					
End of Comment					
184	(B8)	ADDRESS	4	P@CFQLOCK	"V(CFQLOCK)" CF Query Lock holder
Comment					
Module HASPCFQU routines listed alphabetically					
End of Comment					
188	(BC)	ADDRESS	4	P@CFQUERY	"V(CFQUERY)" CF Query connections to str
Comment					
Module HASPCFRD routines listed alphabetically					
End of Comment					
192	(C0)	ADDRESS	4	P@CFRDATA	"V(CFRDATA)" CF Read data
196	(C4)	ADDRESS	4	P@CFRDONE	"V(CFRDONE)" Read one track 1 record
Comment					
Module HASPCFRE routines listed alphabetically					
End of Comment					
200	(C8)	ADDRESS	4	P@CFREL	"V(CFREL)" CF Release structure lock
204	(CC)	ADDRESS	4	P@\$CFTRACE	"V(\$CFTRACE)" CF Trace routine
Comment					
Module HASPCFRL routines listed alphabetically					
End of Comment					
208	(D0)	ADDRESS	4	P@CFRDLIST	"V(CFRDLIST)" CF Read a list of elements
Comment					
Module HASPCFRS routines listed alphabetically					
End of Comment					
212	(D4)	ADDRESS	4	P@CFRESV	"V(CFRESV)" CF Obtain structure lock
Comment					
Module HASPCFR2 routines listed alphabetically					
End of Comment					
216	(D8)	ADDRESS	4	P@CFREAD2	"V(CFREAD2)" CF Read2
220	(DC)	ADDRESS	4	P@CFPURGE	"V(CFPURGE)" CF Purge processing
Comment					
Module HASPCFSI routines listed alphabetically					
End of Comment					
224	(E0)	ADDRESS	4	P@CFSTRTIO	"V(CFSTRTIO)" CF Start I/O



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Module HASPCFT1 routines listed alphabetically					
End of Comment					
228	(E4)	ADDRESS	4	P@CFTRK1IO	"V(CFTRK1IO)" CF Track1 I/O
Comment					
Module HASPCFUN routines listed alphabetically					
End of Comment					
232	(E8)	ADDRESS	4	P@CFUNAL	"V(CFUNAL)" CF Unallocate a structure
Comment					
Module HASPCFWP routines listed alphabetically					
End of Comment					
236	(EC)	ADDRESS	4	P@CFWRINPL	"V(CFWRINPL)" CF Write in place
Comment					
Module HASPCFWR routines listed alphabetically					
End of Comment					
240	(F0)	ADDRESS	4	P@CFWRITE	"V(CFWRITE)" CF Write
Comment					
MODULE HASPCCKDS ROUTINES LISTED ALPHABETICALLY					
End of Comment					
244	(F4)	ADDRESS	4	P@CKBINIT	"V(CKBINIT)" VERIFY SIZE CB'S AND INIT CKB
248	(F8)	ADDRESS	4	P@CKPALCLN	"V(CKPALCLN)" CHECKPOINT ALLOCATION CLEANUP
252	(FC)	ADDRESS	4	P@CKPTALOC	"V(CKPTALOC)" CHECKPOINT DYNAMIC ALLOCATE RTN
256	(100)	ADDRESS	4	P@CKPTUNAL	"V(CKPTUNAL)" CHECKPOINT DATASET UNALLOCATE
260	(104)	ADDRESS	4	P@CKPTVSIZ	"V(CKPTVSIZ)" Verify new ckpt size
264	(108)	ADDRESS	4	P@CKPTXPND	"V(CKPTXPND)" Expand size of the CKPT
268	(10C)	ADDRESS	4	P@KDIALOG	"V(KDIALOG)" CKPT RECOVERY DIALOG SERVICE
272	(110)	ADDRESS	4	P@KRELEASE	"V(KRELEASE)" DEQ (RELEASE) A CKPT DS
276	(114)	ADDRESS	4	P@KRESERVE	"V(KRESERVE)" RESERVE A CKPT DATA SET
280	(118)	ADDRESS	4		Reserved
Comment					
MODULE HASPCCKPT ROUTINES LISTED ALPHABETICALLY					
End of Comment					
284	(11C)	ADDRESS	4	P@\$BERTFIX	"V(\$BERTFIX)" BERT error detect/correct
288	(120)	ADDRESS	4	P@\$CKPTQUE	"V(\$CKPTQUE)" Queue work to CKPT
292	(124)	ADDRESS	4	P@\$DOGBERT	"V(\$DOGBERT)" BERT processing routine
296	(128)	ADDRESS	4	P@BERTFMT	"V(BERTFMT)" Format the BERT CTENT
300	(12C)	ADDRESS	4	P@BERTMAP	"V(BERTMAP)" Process/Build BERT map
304	(130)	ADDRESS	4	P@KBUPDJQE	"V(KBUPDJQE)" Update JQE fields for BLOB
308	(134)	ADDRESS	4	P@KBUPDSUB	"V(KBUPDSUB)" Update JQETGNBR JQE routine
312	(138)	ADDRESS	4	P@KCPYMSTR	"V(KCPYMSTR)" Copy base info to MASTER or MASTERI
316	(13C)	ADDRESS	4	P@KFORMAT	"V(KFORMAT)" REFORMAT A CHECKPOINT DATASET
320	(140)	ADDRESS	4	P@KGETCHLG	"V(KGETCHLG)" Adjust change log size
324	(144)	ADDRESS	4	P@KIOERROR	"V(KIOERROR)" ISSUE A CKPT I/O ERROR MSG
328	(148)	ADDRESS	4	P@KPRIMW	"V(KPRIMW)" Perform a primary write
332	(14C)	ADDRESS	4	P@KPROTECT	"V(KPROTECT)" Page (un)protect the CKPT
336	(150)	ADDRESS	4	P@KREAD2	"V(KREAD2)" PERFORM READ2 OF CKPT DATA SET

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
340	(154)	ADDRESS	4	P@KSETMSTR	"V(KSETMSTR)" Set master record pointers
344	(158)	ADDRESS	4	P@KTRK1IO	"V(KTRK1IO)" PERFORM I/O TO TRK1 OF CKPT DS
348	(15C)	ADDRESS	4	P@QWLMSVDF	"V(QWLMSVDF)" Get WLM service definition
Comment					
Module HASPCKRR routines listed alphabetically					
End of Comment					
352	(160)	ADDRESS	4	P@CKRRDONE	"V(CKRRDONE)" Complete MAS CKPT reconfig
356	(164)	ADDRESS	4	P@CKRRINIT	"V(CKRRINIT)" Initialize and create \$CKM
360	(168)	ADDRESS	4	P@CKRRMASK	"V(CKRRMASK)" Build affinity mask to dump
364	(16C)	ADDRESS	4	P@CKRRSTRT	"V(CKRRSTRT)" Start-up MAS CKPT reconfig
368	(170)	ADDRESS	4	P@CKRRSYNC	"V(CKRRSYNC)" Synchronize MAS reconfig
372	(174)	ADDRESS	4		Reserved for HASPCKRR use
376	(178)	ADDRESS	4		Reserved for HASPCKRR use
380	(17C)	ADDRESS	4		Reserved for HASPCKRR use
384	(180)	ADDRESS	4		Reserved for HASPCKRR use
Comment					
MODULE HASPCKVR ROUTINES LISTED ALPHABETICALLY					
End of Comment					
388	(184)	ADDRESS	4	P@SHRLIVE	"V(SHRLIVE)" SHARE THE NEW LIVE CHECKPOINT
392	(188)	ADDRESS	4	P@UNSHRLIV	"V(UNSHRLIV)" UNSHARE THE LIVE CHECKPOINT
396	(18C)	ADDRESS	4		Reserved
Comment					
Module HASPCNVT routines listed alphabetically					
End of Comment					
400	(190)	ADDRESS	4	P@PROCALOC	"V(PROCALOC)" Allocate PROCLIB data sets
Comment					
MODULE HASPCOMM ROUTINES LISTED ALPHABETICALLY					
End of Comment					
404	(194)	ADDRESS	4	P@\$JCAN	"V(\$JCAN)" Job cancel routine
408	(198)	ADDRESS	4	P@CFPARSE	"V(CFPARSE)" Move and parse command
412	(19C)	ADDRESS	4	P@CJFLCMB	"V(CJFLCMB)" MVS cancel command in CMB
416	(1A0)	ADDRESS	4	P@COFCVE	"V(COFCVE)" ADDR CONVERT TO EBCDIC HALFWORD ROUTINE
420	(1A4)	ADDRESS	4	P@COFEDTR	"V(COFEDTR)" ADDR CONVERT TO EBCDIC FULLWORD ROUTINE
424	(1A8)	ADDRESS	4	P@COFJMSG	"V(COFJMSG)" ADDR JOB INFORMATION MESSAGE ROUTINE
428	(1AC)	ADDRESS	4	P@COFRTC	"V(COFRTC)" ADDR CONVERT TO EBCDIC ROUTE CODE ROUTINE
432	(1B0)	ADDRESS	4	P@COMBEWTO	"V(COMBEWTO)" Branch Entry WTO targeted to executing job
436	(1B4)	ADDRESS	4	P@COMFRELK	"V(COMFRELK)" Free command lock
440	(1B8)	ADDRESS	4	P@CSCANDSP	"V(CSCANDSP)" HASPCOMM \$SCAN DISPLAY ROUTINE, USABLE BY \$SCANS FROM EXIT 5
444	(1BC)	ADDRESS	4	P@CSMICMD	"V(CSMICMD)" HASPCOMM Single member image routine
448	(1C0)	ADDRESS	4	P@CWTO	"V(CWTO)" ADDR WRITE TO OPERATOR RTN
452	(1C4)	ADDRESS	4	P@CWTOT	"V(CWTOT)" ADDR WRITE TO OPERATOR ROUTINE (TRUNC)
456	(1C8)	ADDRESS	4	P@DILJCAN	"V(DILJCAN)" DILBERT'ed call to \$JCAN
460	(1CC)	ADDRESS	4	P@H607RSN	"V(H607RSN)" HASP607 reasons subroutine

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

Module HASPCON routines listed alphabetically

End of Comment

464	(1D0)	ADDRESS	4	P@\$DOM	"V(\$DOM)" HASP DOM routine
468	(1D4)	ADDRESS	4	P@\$FRECMB	"V(\$FRECMB)" Free CMB routine
472	(1D8)	ADDRESS	4	P@\$GETCMBR	"V(\$GETCMBR)" Get CMB routine
476	(1DC)	ADDRESS	4	P@\$WTO	"V(\$WTO)" \$WTO routine
480	(1E0)	ADDRESS	4	P@\$WTOC	"V(\$WTOC)" \$WTO with CMB routine
484	(1E4)	ADDRESS	4	P@HASPWQUE	"V(HASPWQUE)" Addr of CMB queuing routine for callers that cannot \$WAIT
488	(1E8)	ADDRESS	4	P@HASPWQUW	"V(HASPWQUW)" Addr of CMB queuing routine for callers that can tolerate a \$WAIT

Comment

Module HASPCSV routines, listed alphabetically

End of Comment

492	(1EC)	ADDRESS	4	P@\$MODCHK	"V(\$MODCHK)" Check/resolve-from modules
496	(1F0)	ADDRESS	4	P@\$MODELET	"V(\$MODELET)" Delete a load module
500	(1F4)	ADDRESS	4	P@\$MODLOAD	"V(\$MODLOAD)" Load a load module
504	(1F8)	ADDRESS	4	P@CSV\$DEL	"V(CSV\$DEL)" Invoke \$\$\$DEL routine
508	(1FC)	ADDRESS	4	P@CSV\$LOAD	"V(CSV\$LOAD)" Invoke \$\$\$LOAD routine
512	(200)	ADDRESS	4	P@LOCENTRY	"V(LOCENTRY)" Entry point locate routine
516	(204)	ADDRESS	4	P@LOCLMOD	"V(LOCLMOD)" Locate load module by addr
520	(208)	ADDRESS	4	P@LOCMODMP	"V(LOCMODMP)" Locate MODMAP entry by addr

Comment

MODULE HASPDYN ROUTINES LISTED ALPHABETICALLY -  
BASIC CONTROL BLOCK ADDITION/DELETION SERVICES

End of Comment

524	(20C)	ADDRESS	4	P@\$DCBDYN	"V(\$DCBDYN)" DYNAMIC DCB ATTACH/DETACH SERVICE ROUTINE ADDRESS
528	(210)	ADDRESS	4	P@\$DCTDYN	"V(\$DCTDYN)" DYNAMIC DCT ATTACH/DETACH SERVICE ROUTINE ADDRESS
532	(214)	ADDRESS	4	P@\$DESTDYN	"V(\$DESTDYN)" DEST (RDT) DYNAMIC BUILD RTN
536	(218)	ADDRESS	4	P@\$DTEDYNA	"V(\$DTEDYNA)" \$DTEDYN ATTACH ROUTINE ADDRESS
540	(21C)	ADDRESS	4	P@\$DTEDYND	"V(\$DTEDYND)" \$DTEDYN DETACH ROUTINE ADDRESS
544	(220)	ADDRESS	4	P@\$PCEDYDC	"V(\$PCEDYDC)" DYNAMIC PCE ATTACH/DETACH SERVICE FOR A DCT CHAIN
548	(224)	ADDRESS	4	P@\$PCEDYN	"V(\$PCEDYN)" DYNAMIC PCE ATTACH/DETACH SERVICE ROUTINE ADDRESS
552	(228)	ADDRESS	4	P@PRTDFLT	"V(PRTDFLT)" Printer DCT default routine

Comment

MODULE HASPEVTL ROUTINES LISTED ALPHABETICALLY

End of Comment

556	(22C)	ADDRESS	4	P@\$ROLL	"V(\$ROLL)" Entry to create trace entry
560	(230)	ADDRESS	4	P@TRCDUMP	"V(TRCDUMP)" Entry to TRCDUMP routine
564	(234)	ADDRESS	4	P@TRCPUT	"V(TRCPUT)" Entry to TRCPUT routine
568	(238)	ADDRESS	4	P@TRGETTB	"V(TRGETTB)" OBTAIN MORE ECSA TRACE TABLES

Comment

MODULE HASPEXT ROUTINES LISTED ALPHABETICALLY

End of Comment

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
572	(23C)	ADDRESS	4	P@HASPEXDS	"V(HASPEXDS)" Extend dataset
Comment					
MODULE HASPFSSP ROUTINES LISTED ALPHABETICALLY					
End of Comment					
576	(240)	ADDRESS	4	P@DYNFSS	"V(DYNFSS)" DYNAMIC FSSCB FIND/ATTACH INTERNAL SERVICE ROUTINE (HASPFSPP)
Comment					
Entries to HASP Output Process Executive (HASPPOPE)					
End of Comment					
580	(244)	ADDRESS	4	P@OPGROUP	"V(OPGROUP)" Grouping routine
584	(248)	ADDRESS	4	P@OPMAILMG	"V(OPMAILMG)" Notify Routine
588	(24C)	ADDRESS	4	P@OPNULLCK	"V(OPNULLCK)" HASPPOPE - Null D S check
Comment					
ENTRIES TO HASP JOB OUTPUT SERVICES (HASPJOS)					
End of Comment					
592	(250)	ADDRESS	4	P@\$#ADD	"V(\$#ADD)" Entry to JOE add routine
596	(254)	ADDRESS	4	P@\$#ALCHK	"V(\$#ALCHK)" Entry to CHK SPOOL Alloc.
600	(258)	ADDRESS	4	P@\$#BLD	"V(\$#BLD)" Entry to JOE build routine
604	(25C)	ADDRESS	4	P@\$#BUSY	"V(\$#BUSY)" Entry to busy bit set rtne
608	(260)	ADDRESS	4	P@\$#CAN	"V(\$#CAN)" Entry to JOE cancel routine
612	(264)	ADDRESS	4	P@\$#CHK	"V(\$#CHK)" Entry to CHK I/O routine
616	(268)	ADDRESS	4	P@\$#DISPRO	"V(\$#DISPRO)" Entry to update disposition
620	(26C)	ADDRESS	4	P@\$#FORMAT	"V(\$#FORMAT)" Entry to format JOT
624	(270)	ADDRESS	4	P@\$#GET	"V(\$#GET)" Entry to JOE get routine
628	(274)	ADDRESS	4	P@\$#GTNEWS	"V(\$#GTNEWS)" Entry to GET JESNEWS CB
632	(278)	ADDRESS	4	P@\$#JOTBLD	"V(\$#JOTBLD)" Entry to format the JOT
636	(27C)	ADDRESS	4	P@\$#JOTCHK	"V(\$#JOTCHK)" Entry to verify/correct JOT
640	(280)	ADDRESS	4	P@\$#JWEL	"V(\$#JWEL)" Entry to JWEL services
644	(284)	ADDRESS	4	P@\$#MOD	"V(\$#MOD)" Entry to JOE modify routine
648	(288)	ADDRESS	4	P@\$#NEWS	"V(\$#NEWS)" Entry to create JESNEWS DS
652	(28C)	ADDRESS	4	P@\$#POST	"V(\$#POST)" Entry to specific post rtne
656	(290)	ADDRESS	4	P@\$#PUT	"V(\$#PUT)" Entry to JOE put routine
660	(294)	ADDRESS	4	P@\$#RBDCHK	"V(\$#RBDCHK)" Entry to JQE rebuild/free check routine
664	(298)	ADDRESS	4	P@\$#REM	"V(\$#REM)" Entry to JOE remove routine
668	(29C)	ADDRESS	4	P@\$#REP	"V(\$#REP)" Entry to JOE replace rtn
672	(2A0)	ADDRESS	4	P@\$#RLNEWS	"V(\$#RLNEWS)" Entry to return JESNEWS CB
676	(2A4)	ADDRESS	4	P@\$#TJEV	"V(\$#TJEV)" SAPI thread hold
680	(2A8)	ADDRESS	4	P@\$#ZAPJOE	"V(\$#ZAPJOE)" Entry to ZAPJOB JOE rtn
684	(2AC)	ADDRESS	4	P@\$#DOGJOE	"V(\$#DOGJOE)" Entry to DOGJOE service
688	(2B0)	ADDRESS	4	P@ADDTOINX	"V(ADDTOINX)" Add work JOE to PRM/ALT ind
692	(2B4)	ADDRESS	4	P@GTSCREEN	"V(GTSCREEN)" Entry to JOE screen subrtne
696	(2B8)	ADDRESS	4	P@GTSPPOOL	"V(GTSPPOOL)" Entry to chk spools avail
700	(2BC)	ADDRESS	4	P@JOECLUP	"V(JOECLUP)" JOE cleanup
704	(2C0)	ADDRESS	4	P@JOEPPSCR	"V(JOEPPSCR)" JOE post-screen subroutine
708	(2C4)	ADDRESS	4	P@JOEPSCRN	"V(JOEPSCRN)" JOE pre-screen subroutine
712	(2C8)	ADDRESS	4	P@JOESYNC	"V(JOESYNC)" JOE & JWEL time syncronize
716	(2CC)	ADDRESS	4	P@JOTFRECL	"V(JOTFRECL)" Clean up free JOEs
720	(2D0)	ADDRESS	4	P@JOTVERIF	"V(JOTVERIF)" Entry to JOT verify rtns
724	(2D4)	ADDRESS	4	P@MNENF58	"V(MNENF58)" Entry to Main tsk ENF58 rtn
728	(2D8)	ADDRESS	4	P@RTNINDEX	"V(RTNINDEX)" Free index and free JOEs
732	(2DC)	ADDRESS	4	P@SAPIPOST	"V(SAPIPOST)" Entry to SAPIPOST

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Entries to HASP Job Queue Services (HASPJQS)					
End of Comment					
736	(2E0)	ADDRESS	4	P@\$CLASSIF	"V(\$CLASSIF)" Entry to WLM Classification
740	(2E4)	ADDRESS	4	P@\$DILBERT	"V(\$DILBERT)" Entry to \$DILBERT service
744	(2E8)	ADDRESS	4	P@\$DOGDJB	"V(\$DOGDJB)" Entry to DJB processing
748	(2EC)	ADDRESS	4	P@\$DOGJQE	"V(\$DOGJQE)" Entry to artificial JQE srv
752	(2F0)	ADDRESS	4	P@\$FREJLOK	"V(\$FREJLOK)" Free job lock
756	(2F4)	ADDRESS	4	P@\$GETJLOK	"V(\$GETJLOK)" Get job lock
760	(2F8)	ADDRESS	4	P@\$JQEMERG	"V(\$JQEMERG)" Merge JQA into JQE/JQX
764	(2FC)	ADDRESS	4	P@\$QADD	"V(\$QADD)" Entry to JQE add routine
768	(300)	ADDRESS	4	P@\$QBUSY	"V(\$QBUSY)" Entry to busy bit set rtne
772	(304)	ADDRESS	4	P@\$QEXTVER	"V(\$QEXTVER)" Entry to verify JQE ext.
776	(308)	ADDRESS	4	P@\$QEXTFMT	"V(\$QEXTFMT)" Entry to format JQE ext.
780	(30C)	ADDRESS	4	P@\$QFORMAT	"V(\$QFORMAT)" Entry to format JQEs
784	(310)	ADDRESS	4	P@\$QGET	"V(\$QGET)" Entry to JQE get routine
788	(314)	ADDRESS	4	P@\$QJIX	"V(\$QJIX)" Entry to JQE JIX routine
792	(318)	ADDRESS	4	P@\$QLOC	"V(\$QLOC)" Entry to JQE locate routine
796	(31C)	ADDRESS	4	P@\$QLOCNXT	"V(\$QLOCNXT)" Locate next JQE in JIX
800	(320)	ADDRESS	4	P@\$QMOD	"V(\$QMOD)" Entry to JQE modify routine
804	(324)	ADDRESS	4	P@\$QPUT	"V(\$QPUT)" Entry to JQE put routine
808	(328)	ADDRESS	4	P@\$QRBDCHK	"V(\$QRBDCHK)" Entry to JQE rebuild/free check routine
812	(32C)	ADDRESS	4	P@\$QREBLD	"V(\$QREBLD)" Entry to job queue rebuild routine
816	(330)	ADDRESS	4	P@\$QREM	"V(\$QREM)" Entry to JQE remove routine
820	(334)	ADDRESS	4	P@\$QVERIF	"V(\$QVERIF)" Entry to job queue verify routine
824	(338)	ADDRESS	4	P@\$RBLDLOG	"V(\$RBLDLOG)" Entry to rebuild SYSLOG JQE chain routine
828	(33C)	ADDRESS	4	P@\$SCHEMSK	"V(\$SCHEMSK)" Entry to SCHENV processing
832	(340)	ADDRESS	4	P@\$CATDUP	"V(\$CATDUP)" CAT & DUPJOB reconciliation
836	(344)	ADDRESS	4	P@\$CATHMAX	"V(\$CATHMAX)" Entry to CATHMAX processing
840	(348)	ADDRESS	4	P@\$CKVREFRS	"V(\$CKVREFRS)" Refresh checkpoint version
844	(34C)	ADDRESS	4	P@\$DUPJOB	"V(\$DUPJOB)" Entry to Dup job processing
848	(350)	ADDRESS	4	P@\$JNRNGCNT	"V(\$JNRNGCNT)" Job number range counting
852	(354)	ADDRESS	4	P@\$JOBQSAMP	"V(\$JOBQSAMP)" Job queue sampling for WLM
856	(358)	ADDRESS	4	P@\$JQECAT	"V(\$JQECAT)" JQE/CAT time reconciliation
860	(35C)	ADDRESS	4	P@\$MNENF70	"V(\$MNENF70)" Issue job-level ENF
864	(360)	ADDRESS	4	P@\$MNENF78	"V(\$MNENF78)" Issue job notification ENF
868	(364)	ADDRESS	4	P@\$MODJCHG	"V(\$MODJCHG)" Change job MODJOB request
872	(368)	ADDRESS	4	P@\$MODJRLS	"V(\$MODJRLS)" Release job MODJOB request
876	(36C)	ADDRESS	4	P@\$MODJSPN	"V(\$MODJSPN)" SPIN job ds MODJOB request
880	(370)	ADDRESS	4	P@\$MODJXMBR	"V(\$MODJXMBR)" Cross member MODJOB request
884	(374)	ADDRESS	4	P@\$QBERTHRE	"V(\$QBERTHRE)" Determine BERT availability
888	(378)	ADDRESS	4	P@\$QDECHAIN	"V(\$QDECHAIN)" Entry to JQE dechain rtne
892	(37C)	ADDRESS	4	P@\$QJQEVER	"V(\$QJQEVER)" Entry to JQE address verify routine
896	(380)	ADDRESS	4	P@\$WLMDEQ	"V(\$WLMDEQ)" Entry to Dequeue JQE from WLM queue
900	(384)	ADDRESS	4	P@\$WLMENQ	"V(\$WLMENQ)" Entry to Enqueue JQE onto WLM queue
904	(388)	ADDRESS	4	P@\$ZAPJOB	"V(\$ZAPJOB)" ZAP Job service

Comment

MODULE HASPMISC ROUTINES LISTED ALPHABETICALLY

End of Comment					
908	(38C)	ADDRESS	4	P@\$CLASSI4	"V(\$CLASSI4)" Entry to \$CLASSI4 routine
912	(390)	ADDRESS	4	P@\$ENFPOLCY	"V(\$ENFPOLCY)" Entry to ENF Policy Activation Support

Comment

MODULE HASPNATS ROUTINES LISTED ALPHABETICALLY

End of Comment					
916	(394)	ADDRESS	4	P#\$NATADD	Nodes Attached Table ADD

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
920	(398)	ADDRESS	4	P@\$NATADD	"V(\$NATADD)" service routine (HASPNETS)
924	(39C)	ADDRESS	4	P@NADRECV	"V(NADRECV)" \$NATADD recovery routine
928	(3A0)	ADDRESS	4	P#\$NATGET	Nodes Attached Table GET
932	(3A4)	ADDRESS	4	P@\$NATGET	"V(\$NATGET)" service routine (HASPNETS)
936	(3A8)	ADDRESS	4	P@NGTREC V	"V(NGTREC V)" \$NATGET recovery routine
940	(3AC)	ADDRESS	4	P#\$NATMOD	Nodes Attached Table MODify
944	(3B0)	ADDRESS	4	P@\$NATMOD	"V(\$NATMOD)" service routine (HASPNETS)
948	(3B4)	ADDRESS	4	P@NMDRECV	"V(NMDRECV)" \$NATMOD recovery routine
952	(3B8)	ADDRESS	4	P#\$NATNOT	Nodes Attached Table NOTify
956	(3BC)	ADDRESS	4	P@\$NATNOT	"V(\$NATNOT)" service routine (HASPNETS)
960	(3C0)	ADDRESS	4	P@NNTREC V	"V(NNTREC V)" \$NATNOT recovery routine
964	(3C4)	ADDRESS	4	P#\$NATREM	Nodes Attached Table REMove
968	(3C8)	ADDRESS	4	P@\$NATREM	"V(\$NATREM)" service routine (HASPNETS)
972	(3CC)	ADDRESS	4	P@NRMREC V	"V(NRMREC V)" \$NATREM recovery routine
976	(3D0)	ADDRESS	4	P@\$NATREQ	"V(\$NATREQ)" Requeue NAT to appropriate queue
980	(3D4)	ADDRESS	4	P@NPMVFY	"V(NPMVFY)" Network path manager control block verification service
984	(3D8)	ADDRESS	4	P@NPVDCTV	"V(NPVDCTV)" Verify DCT storage is OK
988	(3DC)	ADDRESS	4		RESERVED FOR FUTURE USE

Comment

### MODULE HASPNET ROUTINES LISTED ALPHABETICALLY

End of Comment

992	(3E0)	ADDRESS	4	P@\$FRENHB	"V(\$FRENHB)" Free a header cell
996	(3E4)	ADDRESS	4	P@\$GETNHB	"V(\$GETNHB)" Get a header cell
1000	(3E8)	ADDRESS	4	P@\$NITSYNC	"V(\$NITSYNC)" NIT/CKPT serialization
1004	(3EC)	ADDRESS	4	P@NJDC TINT	"V(NJDC TINT)" LINE DCT INITIALIZATION
1008	(3F0)	ADDRESS	4	P@NJECH ECK	"V(NJECH ECK)" Check I/O completion
1012	(3F4)	ADDRESS	4	P@NJEH DRCV	"V(NJEH DRCV)" Receive NJE header
1016	(3F8)	ADDRESS	4	P@NJEH DRD	"V(NJEH DRD)" Read NJE header from spool
1020	(3FC)	ADDRESS	4	P@NJEH DWR	"V(NJEH DWR)" Write NJE header to spool
1024	(400)	ADDRESS	4	P@NJEH DXMT	"V(NJEH DXMT)" Transmit NJE header
1028	(404)	ADDRESS	4	P@NJEP UT	"V(NJEP UT)" Write NJE record
1032	(408)	ADDRESS	4	P@NJER DACT	"V(NJER DACT)" Clean up receiver jobs
1036	(40C)	ADDRESS	4	P@NJHBUILD	"V(NJHBUILD)" Build job header
1040	(410)	ADDRESS	4	P@NJTBUILD	"V(NJTBUILD)" Build job trailer
1044	(414)	ADDRESS	4	P@NSETESS	"V(NSETESS)" Set ESS section of SMF 24/57 record
1048	(418)	ADDRESS	4	P@NSJFSPSP	"V(NSJFSPSP)" SWBTU split/splice services
1052	(41C)	ADDRESS	4	P@NSMFBSIZ	"V(NSMFBSIZ)" Calculate SMF buffer size

Comment

### MODULE HASPNM ROUTINES LISTED ALPHABETICALLY

End of Comment

1056	(420)	ADDRESS	4	P@HASPNSNR	"V(HASPNSNR)" Initiate NJE signon
1060	(424)	ADDRESS	4	P@NCOMMREQ	"V(NCOMMREQ)" Set up NAT from NTQs
1064	(428)	ADDRESS	4	P@NMAPINIT	"V(NMAPINIT)" Initialize Notify Maps
1068	(42C)	ADDRESS	4	P@NPMHOT	"V(NPMHOT)" NPM hot start recharging
1072	(430)	ADDRESS	4	P@NPMQSUSE	"V(NPMQSUSE)" NPM request \$QSUSE
1076	(434)	ADDRESS	4	P@NSETSUBS	"V(NSETSUBS)" Set SUBNET chaining fields in the NIT
1080	(438)	ADDRESS	4		Reserved for future use
1084	(43C)	ADDRESS	4		Reserved for future use
1088	(440)	ADDRESS	4		Reserved for future use

Comment

### MODULE HASPNRM ROUTINES LISTED ALPHABETICALLY

End of Comment

1092	(444)	ADDRESS	4	P@NRMAJUST	"V(NRMAJUST)" Reset NRM STIMER
------	-------	---------	---	------------	--------------------------------

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MODULE HASPNUC ROUTINES LISTED ALPHABETICALLY					
End of Comment					
1096	(448)	ADDRESS	4	P@\$BFRBLD	"V(\$BFRBLD)" Buffer build routine
1100	(44C)	ADDRESS	4	P@\$CBIOM	"V(\$CBIOM)" I/O FOR JES2 CONTROL BLOCKS
1104	(450)	ADDRESS	4	P@\$CHECK	"V(\$CHECK)" CHECK COMPLETION OF A CKPT WRT
1108	(454)	ADDRESS	4	P@\$CKPT	"V(\$CKPT)" SCHED CKPT FOR AN ALTERED ELMT
1112	(458)	ADDRESS	4	P@\$DSCLOSE	"V(\$DSCLOSE)" Entry to \$DSCLOSE routine
1116	(45C)	ADDRESS	4	P@\$DSOPEN	"V(\$DSOPEN)" Entry to \$DSOPEN routine
1120	(460)	ADDRESS	4	P@\$DSPUT	"V(\$DSPUT)" Entry to \$DSPUT routine
1124	(464)	ADDRESS	4	P@\$DYN	"V(\$DYN)" Dynamic allocate/unallocate
1128	(468)	ADDRESS	4	P@\$DYNLERR	"V(\$DYNLERR)" DYNALOC error routine
1132	(46C)	ADDRESS	4	P@\$EXCP	"V(\$EXCP)" EXCP routine
1136	(470)	ADDRESS	4	P@\$EXTP	"V(\$EXTP)" RTAM service routines
1140	(474)	ADDRESS	4	P@\$FREEBFR	"V(\$FREEBFR)" Free a buffer
1144	(478)	ADDRESS	4	P@\$FRELOK	"V(\$FRELOK)" Free CMS lock
1148	(47C)	ADDRESS	4	P@\$FRESMF	"V(\$FRESMF)" Free an SMF buffer
1152	(480)	ADDRESS	4	P@\$FREUCBS	"V(\$FREUCBS)" Free storage for UPL
1156	(484)	ADDRESS	4	P@\$FREUNIT	"V(\$FREUNIT)" HASP unit 'FREE' routine
1160	(488)	ADDRESS	4	P@\$GETBUF	"V(\$GETBUF)" Get a buffer
1164	(48C)	ADDRESS	4	P@\$GETLOK	"V(\$GETLOK)" Get CMS lock
1168	(490)	ADDRESS	4	P@\$GETSAVE	"V(\$GETSAVE)" Get a \$SAVE area
1172	(494)	ADDRESS	4	P@\$GETSMF	"V(\$GETSMF)" Get SMF buffer
1176	(498)	ADDRESS	4	P@\$GETUCBS	"V(\$GETUCBS)" Obtain UCB address
1180	(49C)	ADDRESS	4	P@\$GETUNIT	"V(\$GETUNIT)" HASP unit 'GET' routine
1184	(4A0)	ADDRESS	4	P@\$GETWORK	"V(\$GETWORK)" Get a work area
1188	(4A4)	ADDRESS	4	P@\$GFMAIN	"V(\$GFMAIN)" ENTRY TO GET/FREE STG RTN
1192	(4A8)	ADDRESS	4	P@\$IOTCNT	"V(\$IOTCNT)" Entry to daughter count
1196	(4AC)	ADDRESS	4	P@\$JESEFF	"V(\$JESEFF)" JES2 Exit effector
1200	(4B0)	ADDRESS	4	P@\$PAWS	"V(\$PAWS)" PAWS (pause) a PCE
1204	(4B4)	ADDRESS	4	P@\$PGSRVC	"V(\$PGSRVC)" PAGE SERVICE ROUTINE
1208	(4B8)	ADDRESS	4	P@\$POST	"V(\$POST)" POST SPECIFIC EVENT ROUTINE
1212	(4BC)	ADDRESS	4	P@\$POSTSUB	"V(\$POSTSUB)" Subtask post service
1216	(4C0)	ADDRESS	4	P@\$QSUSE	"V(\$QSUSE)" Entry to \$QSUSE support
1220	(4C4)	ADDRESS	4	P@\$QUESMF	"V(\$QUESMF)" Queue SMF buffer
1224	(4C8)	ADDRESS	4	P@\$RETSAVE	"V(\$RETSAVE)" Save area free routine
1228	(4CC)	ADDRESS	4	P@\$RETURN	"V(\$RETURN)" Return a \$SAVE area
1232	(4D0)	ADDRESS	4	P@\$RETWORK	"V(\$RETWORK)" Return a work area
1236	(4D4)	ADDRESS	4	P@\$SEAS	"V(\$SEAS)" ENTRY TO SECURITY AUTH RTN
1240	(4D8)	ADDRESS	4	P@\$SEASMSG	"V(\$SEASMSG)" ISSUE THE 077 MESSAGE
1244	(4DC)	ADDRESS	4	P@\$STCK	"V(\$STCK)" HASP store clock routine
1248	(4E0)	ADDRESS	4	P@\$STCKFMT	"V(\$STCKFMT)" HASP store clock conversion
1252	(4E4)	ADDRESS	4	P@\$STIMER	"V(\$STIMER)" HASP set timer routine
1256	(4E8)	ADDRESS	4	P@\$SUBIT	"V(\$SUBIT)" SUBTASK WORK QUEUING RTN
1260	(4EC)	ADDRESS	4	P@\$TTIMER	"V(\$TTIMER)" HASP test timer routine
1264	(4F0)	ADDRESS	4	P@\$WAIT	"V(\$WAIT)" WAIT FOR AN EVENT ROUTINE
1268	(4F4)	ADDRESS	4	P@\$XECBKIL	"V(\$XECBKIL)" XECB DE-CHAINING ROUTINE
1272	(4F8)	ADDRESS	4	P@\$GETEVNTR	"V(\$GETEVNTR)" Get an event record CB
1276	(4FC)	ADDRESS	4	P@\$GETJOBKY	"V(\$GETJOBKY)" Obtain a jobkey
1280	(500)	ADDRESS	4	P@\$MOD875	"V(\$MOD875)" ISSUE \$HASP875 MESSAGE
1284	(504)	ADDRESS	4	P@\$SUBDEST	"V(\$SUBDEST)" SUBTASK A \$DESTCHK CALL
Comment					

ENTRIES TO HASP SWB MODIFY SUBTASK (HASPODSM)

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
End of Comment					
1288	(508)	ADDRESS	4	P#SWBMSUB	SWB MODIFY SUBTASK PC NUM

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Entries to HASP Process Sysout (HASPPSO)					
End of Comment					
1292	(50C)	ADDRESS	4	P@TREGROUP	"V(TREGROUP)" Regroup PDDB
1296	(510)	ADDRESS	4	P@PSOFRELK	"V(PSOFRELK)" Free job lock and JOE busy
Comment					
MODULE HASPRAS ROUTINES LISTED ALPHABETICALLY					
End of Comment					
1300	(514)	ADDRESS	4	P@\$DISTERR	"V(\$DISTERR)" Disastrous error routine
1304	(518)	ADDRESS	4	P@\$ESTACAN	"V(\$ESTACAN)" ENTRY TO \$ESTAE CANCEL RTN
1308	(51C)	ADDRESS	4	P@\$ESTAER	"V(\$ESTAER)" ENTRY TO \$ESTAE ESTAB. RTN
1312	(520)	ADDRESS	4	P@\$ESTAREP	"V(\$ESTAREP)" ENTRY TO \$ESTAE REPLACE RTN
1316	(524)	ADDRESS	4	P@\$IOERROR	"V(\$IOERROR)" I/O error logging routine
1320	(528)	ADDRESS	4	P@\$SDUMP	"V(\$SDUMP)" SVC dump routine
Comment					
Module HASPRDR routines listed alphabetically					
End of Comment					
1324	(52C)	ADDRESS	4	P@RDRPDCUP	"V(RDRPDCUP)" NJE/RJE reader cleanup rtn
1328	(530)	ADDRESS	4	P@RINTJOB	"V(RINTJOB)" Create Internal Job service
Comment					
MODULE HASPRTAM ROUTINES LISTED ALPHABETICALLY					
End of Comment					
1332	(534)	ADDRESS	4	P@\$REQBUF	"V(HASPRBUF)" Entry to requeue buffers and request ckpt
1336	(538)	ADDRESS	4	P@\$REQBUFN	"V(HASPRBFN)" Entry to requeue bfrs without requesting ckpt
1340	(53C)	ADDRESS	4	P@LNEAVRJE	"V(LNEAVRJE)" Check if Line avail for RJE
1344	(540)	ADDRESS	4	P@MLMRCPCL	"V(MLMRCPCL)" Rebuild PCL chains
1348	(544)	ADDRESS	4	P@MSAFCHK	"V(MSAFCHK)" SAF CALL FOR LM AND RCP
1352	(548)	ADDRESS	4	P@RMTDVINT	"V(RMTDVINT)" Initialize Rmt Device DCT
1356	(54C)	ADDRESS	4	P@RMTDVSET	"V(RMTDVSET)" Setup Rmt Device DCT
1360	(550)	ADDRESS	4	P@RMTLNECK	"V(RMTLNECK)" Check Rmt Line setting
1364	(554)	ADDRESS	4	P@RMTSETUP	"V(RMTSETUP)" Setup RMT Parameters
Comment					
Module HASPSASR Routines listed alphabetically					
End of Comment					
1368	(558)	ADDRESS	4	P@SAIHOT	"V(SAIHOT)" SAPI Hot Start Processing
1372	(55C)	ADDRESS	4	P@SAIRECC	"V(SAIRECC)" Update record/page counts
Comment					
MODULE HASPSERV ROUTINES LISTED ALPHABETICALLY					
End of Comment					
1376	(560)	ADDRESS	4	P@ADDCTQ	"V(ADDCTQ)" Addr Add DCT to Q routine
1380	(564)	ADDRESS	4	P@CALCBRTN	"V(CALCBRTN)" ADDR CALC BERTNUM value
1384	(568)	ADDRESS	4	P@CFJOED	"V(CFJOED)" ADDR JOE DISPLAY ROUTINE
1388	(56C)	ADDRESS	4	P@IVATE	"V(IVATE)" Addr of \$ACTIVATE routine
1392	(570)	ADDRESS	4	P@RBLDCTQ	"V(RBLDCTQ)" Addr Rebuild DCT Q rtn
1396	(574)	ADDRESS	4	P@REMDCTQ	"V(REMDCTQ)" Addr Remove DCT from Q rtn
1400	(578)	ADDRESS	4	P@ROTDCTQ	"V(ROTDCTQ)" Addr Rotate DCT on Q rtn



Offsets						
Dec	Hex	Type/Value	Len	Name (Dim)	Description	
1404	(57C)	ADDRESS	4	P@SRVCFSEL	"V(SRVCFSEL)" Addr of \$CFSEL service rtn	
1408	(580)	ADDRESS	4	P@SRVDCTD	"V(SRVDCTD)" ADDR DEVICE CONTROL TABLE DISPLAY ROUTINE	
1412	(584)	ADDRESS	4	P@SRVFNDCR	"V(SRVFNDCR)" ADDR OF FIND CRI ROUTINE	
1416	(588)	ADDRESS	4	P@SRVMOD	"V(SRVMOD)" ADDR MODIFY JOB/SYSOUT CHARS ROUTINE	
1420	(58C)	ADDRESS	4	P@SRVM630	"V(SRVM630)" Addr of routine to format \$HASP630 message	
1424	(590)	ADDRESS	4	P@SRVOLOC	"V(SRVOLOC)" ADDR LOCATE DAS DATA SET DSECT ROUTINE	
1428	(594)	ADDRESS	4	P@SRVPREFX	"V(SRVPREFX)" ADDR DEFINE PREFIX TO MCS ROUTINE	
1432	(598)	ADDRESS	4		Reserved for future use	
1436	(59C)	ADDRESS	4	P@SRVRDIR	"V(SRVRDIR)" ADDR OF ROUTINE TO REDIRECT COMMAND RESPONSES	
1440	(5A0)	ADDRESS	4	P@SRVROUT	"V(SRVROUT)" ADDR CONVERT TO BINARY ROUTE CODE ROUTINE	
1444	(5A4)	ADDRESS	4	P@SRVSASCN	"V(SRVSASCN)" ADDR SYS AFFINITY SCAN RTN	
1448	(5A8)	ADDRESS	4	P@SRVSETUP	"V(SRVSETUP)" ADDR WORK SELECT SET UP RTN	
1452	(5AC)	ADDRESS	4	P@SRVWSCAN	"V(SRVWSCAN)" ADDR WORK SELECT SCAN RTN	
1456	(5B0)	ADDRESS	4	P@SUBRRT	"V(SUBRRT)" SUBTASK \$REROUTE ROUTINE	
1460	(5B4)	ADDRESS	4	P@VETIVATE	"V(VETIVATE)" Check \$ACTIVATE viability routine	
1464	(5B8)	ADDRESS	4	P@WS2	"V(WS2)" Work selection control block errors	
1468	(5BC)	ADDRESS	4	P@XCSAPST	"V(XCSAPST)" Post SAPI WSP	
1472	(5C0)	ADDRESS	4	P@\$DOJAX	"V(\$DOJAX)" \$DOJAX service	
1476	(5C4)	ADDRESS	4	P@\$WSPXCFY	"V(\$WSPXCFY)" WSP classification WRT the JOE index	
1480	(5C8)	ADDRESS	4	P@\$WSPXKEY	"V(\$WSPXKEY)" Get next key value for WSP WRT the JOE index	
1484	(5CC)	ADDRESS	4	P@\$XCWCRT	"V(\$XCWCRT)" Create WSP cache	
1488	(5D0)	ADDRESS	4	P@\$XCWDLT	"V(\$XCWDLT)" Delete WSP cache	
1492	(5D4)	ADDRESS	4	P@\$XCWPOST	"V(\$XCWPOST)" Post all eligible WSPs	
1496	(5D8)	ADDRESS	4	P@\$XCWPRG	"V(\$XCWPRG)" Purge WSP from WSP cache	
1500	(5DC)	ADDRESS	4		RESERVED FOR FUTURE USE	
1504	(5E0)	ADDRESS	4		RESERVED FOR FUTURE USE	

Comment

MODULE HASPSIR ROUTINES LISTED ALPHABETICALLY

End of Comment

1508	(5E4)	ADDRESS	4	P@\$IOTERR	"V(\$IOTERR)" Spin IOT error recovery routine
1512	(5E8)	ADDRESS	4	P@\$ISSWTO	"V(\$ISSWTO)" \$HASP394 Output Lost message routine

Comment

MODULE HASPSNA ROUTINES LISTED ALPHABETICALLY

End of Comment

1516	(5EC)	ADDRESS	4	P@APPLDYN	"V(APPLDYN)" DYNAMIC APT LOOKUP/ATTACH SERVICE ROUTINE (HASPSNA)
1520	(5F0)	ADDRESS	4	P@SNASNET	"V(SNASNET)" START NETWORKING (\$SN) COMMAND EXIT FOR SNA (HASPSNA)

Comment

MODULE HASSPIN ROUTINES LISTED ALPHABETICALLY

End of Comment

1524	(5F4)	ADDRESS	4	P@SPCIOT	"V(\$SPCIOT)" Spin IOT in CSA (LIFO/FIFO) check routine
------	-------	---------	---	----------	---

Comment

MODULE HASSPOL ROUTINES LISTED ALPHABETICALLY

End of Comment

1528	(5F8)	ADDRESS	4	P@\$DASFMT	"V(\$DASFMT)" FORMAT new DASes
------	-------	---------	---	------------	--------------------------------

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
1532	(5FC)	ADDRESS	4	P@\$DOGMIG	"V(\$DOGMIG)" Entry to MIG processing
1536	(600)	ADDRESS	4	P@DADADDWQ	"V(DADADDWQ)" Add DAS to DAS work queue
1540	(604)	ADDRESS	4	P@DADAVAIL	"V(DADAVAIL)" DAS TG COUNT ROUTINE
1544	(608)	ADDRESS	4	P@DADCKALL	"V(DADCKALL)" Check command status rtn
1548	(60C)	ADDRESS	4	P@DADCKTGM	"V(DADCKTGM)" TGM CKPT ROUTINE
1552	(610)	ADDRESS	4	P@DADCOUNT	"V(DADCOUNT)" FREE TG COUNTING ROUTINE
1556	(614)	ADDRESS	4	P@DADDEB	"V(DADDEB)" DAS DEB EXT. INIT ROUTINE
1560	(618)	ADDRESS	4	P@DADEXIST	"V(DADEXIST)" Determine if DAS exists
1564	(61C)	ADDRESS	4	P@DADREMVE	"V(DADREMVE)" REMOVE DAS FROM QUEUES RTN
1568	(620)	ADDRESS	4	P@DADREMWQ	"V(DADREMWQ)" REMOVE DAS FROM WORK Q RTN
1572	(624)	ADDRESS	4	P@DADSTUNT	"V(DADSTUNT)" Deal with stunted volumes
1576	(628)	ADDRESS	4	P@DADSPLST	"V(DADSPLST)" RESET SPL CONTROL BLOCK RTN
1580	(62C)	ADDRESS	4	P@DADTGM	"V(DADTGM)" DAS TGM UPDATE ROUTINE
1584	(630)	ADDRESS	4	P@DADTGMSP	"V(DADTGMSP)" Get TGM Space, Last DAS Rtn
1588	(634)	ADDRESS	4	P@DADXTENT	"V(DADXTENT)" DAS EXTENT INIT ROUTINE
1592	(638)	ADDRESS	4	P@MIGRRECV	"V(MIGRRECV)" Migration recovery/restart
1596	(63C)	ADDRESS	4	P@RCDSYNC	"V(RCDSYNC)" Initialize RECYDAS
1600	(640)	ADDRESS	4	P@SIGIO	"V(SIGIO)" Signature I/O Routine
1604	(644)	ADDRESS	4	P@SNFQUE	"V(SNFQUE)" Sniffer BLOB Queueing Rtn
1608	(648)	ADDRESS	4	P@SNFQBLD	"V(SNFQBLD)" Build SNFQUE rtn
1612	(64C)	ADDRESS	4	P@SNFQPST	"V(SNFQPST)" Queue SNFQUE rtn

Comment

### MODULE HASPSSRV ROUTINES LISTED ALPHABETICALLY

End of Comment

1616	(650)	ADDRESS	4	P@\$RERROUTE	"V(\$RERROUTE)" REROUTE CMD AUTH ROUTINE
1620	(654)	ADDRESS	4	P@EXTDCTSL	"V(EXTDCTSL)" Extract DCT SECLABEL
1624	(658)	ADDRESS	4	P@NEWSCRE	"V(NEWSCRE)" JESNEWS Dataset creation
1628	(65C)	ADDRESS	4	P@PSAFSCAN	"V(PSAFSCAN)" Pddb SCAN AND SAF CALL RTN

Comment

### Module HASPSTUB routines listed alphabetically

End of Comment

1632	(660)	ADDRESS	4	P@CFPOST	"V(CFPOST)" \$\$POST checkpoint
------	-------	---------	---	----------	---------------------------------

Comment

### Module HASPSUBS routines listed alphabetically

End of Comment

1636	(664)	ADDRESS	4	P@SUBSPERF	"V(SUBSPERF)" Update subtask perf stats
------	-------	---------	---	------------	---

Comment

### Module HASPSXIT routines listed alphabetically

End of Comment

1640	(668)	ADDRESS	4	P@LPRMLIBP	"V(LPRMLIBP)" LOGICAL PARMLIB PROCESSING
1644	(66C)	ADDRESS	4	P@QQSESTAT	"V(QQSESTAT)" QSE state

Comment

### Module HASPTABS routines listed alphabetically

End of Comment

1648	(670)	ADDRESS	4	P@\$GETABLE	"V(\$GETABLE)" HASPTABS - \$GETABLE service
1652	(674)	ADDRESS	4	P@\$PUTABLE	"V(\$PUTABLE)" HASPTABS - \$PUTABLE service
1656	(678)	ADDRESS	4	P@\$RETABLE	"V(\$RETABLE)" HASPTABS - \$RETABLE service

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

Module HASPTCP routines listed alphabetically

End of Comment

1660	(67C)	ADDRESS	4	P@CNVIPAD	"V(CNVIPAD)" IP ADDRESS CONVERASION
1664	(680)	ADDRESS	4	P@SOCKDYN	"V(SOCKDYN)" DYNAMIC SCK LOOKUP/ATTACH
1668	(684)	ADDRESS	4	P@TCPSNET	"V(TCPSNET)" START TCP/IP NJE

Comment

Module HASPTERM routines listed alphabetically

End of Comment

1672	(688)	ADDRESS	4	P@\$ABEND	"V(\$ABEND)" JES2 Main task recovery rtn
1676	(68C)	ADDRESS	4	P@\$HEXIT	"V(\$HEXIT)" Normal JES2 termination
1680	(690)	ADDRESS	4	P@\$PCABEND	"V(\$PCABEND)" JES2 PC recovery routine
1684	(694)	ADDRESS	4	P@ABNDRATE	"V(ABNDRATE)" Determine ABEND rate
1688	(698)	ADDRESS	4	P@HEXINIT	"V(HEXINIT)" Termination for HASPINIT
1692	(69C)	ADDRESS	4	P@WTORTIMR	"V(WTORTIMR)" Waits for a WTOR with a timer

Comment

Module HASPTRAK routines listed alphabetically

End of Comment

1696	(6A0)	ADDRESS	4	P@\$BLDTGB	"V(\$BLDTGB)" Queue TGBs
1700	(6A4)	ADDRESS	4	P@\$PURGER	"V(\$PURGER)" Release IOT tracks
1704	(6A8)	ADDRESS	4	P@\$TGMIG	"V(\$TGMIG)" Transpose Source->target TGM bits.
1708	(6AC)	ADDRESS	4	P@\$TGMSET	"V(\$TGMSET)" Set trackgroup map
1712	(6B0)	ADDRESS	4	P@\$TRACK	"V(\$TRACK)" Get SPOOL space
1716	(6B4)	ADDRESS	4	P@COMLOPER	"V(COMLOPER)" L= PROCESSING
1720	(6B8)	ADDRESS	4	P@PURMASC	"V(PURMASC)" Purge single TGAE

Comment

-----  
 Module HASPWARM routines listed alphabetically  
 -----

End of Comment

1724	(6BC)	ADDRESS	4	P@NQPSOQ	"V(NQPSOQ)" Purge Status/Cancel and PSO queues routine
1728	(6C0)	ADDRESS	4	P@NQRELSE	"V(NQRELSE)" Release duplicate jobs

Comment

Module HASPXCF routines listed alphabetically

End of Comment

1732	(6C4)	ADDRESS	4	P@MSTNTFY	"V(MSTNTFY)" Member state change notify
1736	(6C8)	ADDRESS	4	P@\$MSTNTFY	"V(\$MSTNTFY)" Same for JES2 environment
1740	(6CC)	ADDRESS	4	P@\$TQLEVEL	"V(\$TQLEVEL)" Test MAS levels (main task)
1744	(6D0)	ADDRESS	4	P@\$TQLVLS	"V(\$TQLVLS)" Test MAS levels (subtask)
1748	(6D4)	ADDRESS	4	P@XCFBCAST	"V(XCFBCAST)" Broadcast an XCF message
1752	(6D8)	ADDRESS	4	P@XCFDHOMO	"V(XCFDHOMO)" Determine Homogeneity
1756	(6DC)	ADDRESS	4	P@XCFHELTH	"V(XCFHELTH)" Health checker Invocation
1760	(6E0)	ADDRESS	4	P@XCFJOIN	"V(XCFJOIN)" Member joins XCF group
1764	(6E4)	ADDRESS	4	P@XCFLEAVE	"V(XCFLEAVE)" Member leaves XCF group
1768	(6E8)	ADDRESS	4	P@XCFMAPEV	"V(XCFMAPEV)" Map XCF events to QSE
1772	(6EC)	ADDRESS	4	P@XCFMSTAT	"V(XCFMSTAT)" Query all members status
1776	(6F0)	ADDRESS	4	P@XCFQSTAT	"V(XCFQSTAT)" Query a members status
1780	(6F4)	ADDRESS	4	P@XCFUSTAT	"V(XCFUSTAT)" Update the user status
1784	(6F8)	ADDRESS	4	P@XCFXMAQU	"V(XCFXMAQU)" Update a members XMAQU

## \$PADDR Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Module HASPXEQ routines listed alphabetically					
End of Comment					
1788	(6FC)	ADDRESS	4	P@\$ASDXCLR	"V(\$ASDXCLR)" ASDS entry clear
1792	(700)	ADDRESS	4	P@\$ASDXUPD	"V(\$ASDXUPD)" ASDS entry update
1796	(704)	ADDRESS	4	P@\$CATCRNW	"V(\$CATCRNW)" Reset 'no selectable work' field in relevant CAT cache elements.
1800	(708)	ADDRESS	4	P@\$CATCWRT	"V(\$CATCWRT)" Write CAT/GRPOBJ cache to BERTs.
1804	(70C)	ADDRESS	4	P@\$CATDEFS	"V(\$CATDEFS)" Set defaults in the CAT
1808	(710)	ADDRESS	4	P@\$CATINIT	"V(\$CATINIT)" Set CATs in BERTs
1812	(714)	ADDRESS	4	P@\$CATJCNT	"V(\$CATJCNT)" Reset xeq count in CAT
1816	(718)	ADDRESS	4	P@\$CRWSCQ	"V(\$CRWSCQ)" Create WSC
1820	(71C)	ADDRESS	4	P@\$CREGWLM	"V(\$CREGWLM)" Register WLM class
1824	(720)	ADDRESS	4	P@\$DMNDJOB	"V(\$DMNDJOB)" Demand job start/test
1828	(724)	ADDRESS	4	P@\$DOGCAT	"V(\$DOGCAT)" Deliver or Get CAT
1832	(728)	ADDRESS	4	P@\$DOGGRP	"V(\$DOGGRP)" Deliver or Get CLASGRP
1836	(72C)	ADDRESS	4	P@\$DOGWSCQ	"V(\$DOGWSCQ)" Deliver or Get WSC
1840	(730)	ADDRESS	4	P@\$PLEXREG	"V(\$PLEXREG)" JESplex queue registration
1844	(734)	ADDRESS	4	P@\$XPURJWEL	"V(\$XPURJWEL)" Purge JWELs for AS
1848	(738)	ADDRESS	4	P@\$CATAGRP	"V(\$CATAGRP)" Add a CAT to a CLASGRP
1852	(73C)	ADDRESS	4	P@\$CATCLEAN	"V(\$CATCLEAN)" Process deleted JOBCLASS
1856	(740)	ADDRESS	4	P@\$CATDGRP	"V(\$CATDGRP)" Delete a CAT from a CLASGRP
Comment					
COMPATIBILITY					
Remove after all z8					
End of Comment					
1860	(744)	ADDRESS	4	P@\$DUPTRANS	"V(\$DUPTRANS)" Transition for duplicate jobs pre-z8 to all z8
Comment					
End Compat code					
End of Comment					
1864	(748)	ADDRESS	4	P@\$HASP051	"V(\$HASP051)" BERT Shortage message
1868	(74C)	ADDRESS	4	P@\$MODESWIT	"V(\$MODESWIT)" Mode switch for class queue
1872	(750)	ADDRESS	4	P@\$TIMECLOC	"V(\$TIMECLOC)" Manage JQE timers
1876	(754)	ADDRESS	4	P@\$WLMGOALS	"V(\$WLMGOALS)" Compute WLM goals
1880	(758)	ADDRESS	4	P@\$XDUPTEST	"V(\$XDUPTEST)" Check for duplicates
1884	(75C)	ADDRESS	4	P@\$XINSTART	"V(\$XINSTART)" Start an initiator
1888	(760)	ADDRESS	4	P@\$XPOSTXEQ	"V(\$XPOSTXEQ)" EXEC PCE POST routine
1892	(764)	ADDRESS	4		RESERVED FOR FUTURE USE
1896	(768)	ADDRESS	4		RESERVED FOR FUTURE USE
1900	(76C)	ADDRESS	4		RESERVED FOR FUTURE USE
1904	(770)	ADDRESS	4		RESERVED FOR FUTURE USE
1904	(770)	X'774'	0	PADDRLEN	"*-PADDR" Length of the PADDR table

\$PADDR Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
P#\$NATADD	394		P@\$DOGJOE	2AC	
P#\$NATGET	3A0		P@\$DOGMIG	5FC	
P#\$NATMOD	3AC		P@\$DOGWSCQ	72C	
P#\$NATNOT	3B8		P@\$DOM	1D0	
P#\$NATREM	3C4		P@\$DSCLOSE	458	
P#SWBMSUB	508		P@\$DSOPEN	45C	
P@\$ADD	250		P@\$DSPUT	460	
P@\$ALCHK	254		P@\$DTEDYNA	218	
P@\$BLD	258		P@\$DTEDYND	21C	
P@\$BUSY	25C		P@\$DYN	464	
P@\$CAN	260		P@\$DYNLERR	468	
P@\$CHK	264		P@\$ESTACAN	518	
P@\$DISPRO	268		P@\$ESTAER	51C	
P@\$FORMAT	26C		P@\$ESTAREP	520	
P@\$GET	270		P@\$EXCP	46C	
P@\$GTNEWS	274		P@\$EXTP	470	
P@\$JOTBLD	278		P@\$FRECMB	1D4	
P@\$JOTCHK	27C		P@\$FREEBFR	474	
P@\$JWEL	280		P@\$FREJLOK	2F0	
P@\$MOD	284		P@\$FRELOK	478	
P@\$NEWS	288		P@\$FRENHB	3E0	
P@\$POST	28C		P@\$FRESMF	47C	
P@\$PUT	290		P@\$FREUCBS	480	
P@\$RBDCHK	294		P@\$FREUNIT	484	
P@\$REM	298		P@\$GETABLE	670	
P@\$REP	29C		P@\$GETBUFR	488	
P@\$RLNEWS	2A0		P@\$GETCMBR	1D8	
P@\$TJEV	2A4		P@\$GETJLOK	2F4	
P@\$ZAPJOE	2A8		P@\$GETLOK	48C	
P@\$ABEND	688		P@\$GETNHB	3E4	
P@\$ASDXCLR	6FC		P@\$GETSAVE	490	
P@\$ASDXUPD	700		P@\$GETSMF	494	
P@\$BERTFIX	11C		P@\$GETUCBS	498	
P@\$BFRBLD	448		P@\$GETUNIT	49C	
P@\$BLDTGB	6A0		P@\$GETWORK	4A0	
P@\$CATCRNW	704		P@\$GFMAIN	4A4	
P@\$CATCWRT	708		P@\$HEXIT	68C	
P@\$CATDEFS	70C		P@\$IOERROR	524	
P@\$CATINIT	710		P@\$IOTCNT	4A8	
P@\$CATJCNT	714		P@\$IOTERR	5E4	
P@\$CBIOM	44C		P@\$JCAN	194	
P@\$CDCTDYN	8C		P@\$JESEFF	4AC	
P@\$CFTRACE	CC		P@\$JQEMERG	2F8	
P@\$CHECK	450		P@\$MODCHK	1EC	
P@\$CKPT	454		P@\$MODELET	1F0	
P@\$CKPTQUE	120		P@\$MODLOAD	1F4	
P@\$CLASSIF	2E0		P@\$MSTNTFY	6C8	
P@\$CLASSI4	38C		P@\$NATADD	398	
P@\$CNITNOT	90		P@\$NATGET	3A4	
P@\$CREGWLM	71C		P@\$NATMOD	3B0	
P@\$CRWSCQ	718		P@\$NATNOT	3BC	
P@\$DASFMT	5F8		P@\$NATREM	3C8	
P@\$DCBDYN	20C		P@\$NATREQ	3D0	
P@\$DCTDYN	210		P@\$NITSYNC	3E8	
P@\$DESTDYN	214		P@\$PAWS	4B0	
P@\$DILBERT	2E4		P@\$PCABEND	690	
P@\$DISTERR	514		P@\$PCEDYDC	220	
P@\$DMNDJOB	720		P@\$PCEDYN	224	
P@\$DOGBERT	124		P@\$PGSRVC	4B4	
P@\$DOGCAT	724		P@\$PLEXREG	730	
P@\$DOGDJB	2E8		P@\$POST	4B8	
P@\$DOGDJP	728		P@\$POSTSUB	4BC	
P@\$DOGGRP	728				
P@\$DOGJAX	5C0				

## \$PADDR Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
P@\$PURGER	6A4		P@CATHMAX	344	
P@\$PUTABLE	674		P@CFALOC	94	
P@\$QADD	2FC		P@CFBLDLST	98	
P@\$QBUSY	300		P@CFCOMP	A0	
P@\$QEXTFMT	308		P@CFDELETE	9C	
P@\$QEXTVER	304		P@CFEVEN	A4	
P@\$QFORMAT	30C		P@CFFCOMP	AC	
P@\$QGET	310		P@CFFORMAT	B4	
P@\$QJIX	314		P@CFJOED	568	
P@\$QLOC	318		P@CFNOTIFY	A8	
P@\$QLOCNXT	31C		P@CFPARSE	198	
P@\$QMOD	320		P@CFPOST	660	
P@\$QPUT	324		P@CFPURGE	DC	
P@\$QRBDCHK	328		P@CFQLOCK	B8	
P@\$QREBLD	32C		P@CFQUERY	BC	
P@\$QREM	330		P@CFRDATA	C0	
P@\$QSUSE	4C0		P@CFRDLEC	B0	
P@\$QUESMF	4C4		P@CFRDLIST	D0	
P@\$QVERIF	334		P@CFRDONE	C4	
P@\$RBLDLOG	338		P@CFREAD2	D8	
P@\$REQBUF	534		P@CFREL	C8	
P@\$REQBUFN	538		P@CFRESV	D4	
P@\$RERROUTE	650		P@CFSTRTIO	E0	
P@\$RETABLE	678		P@CFTRK1IO	E4	
P@\$RETSAVE	4C8		P@CFUNAL	E8	
P@\$RETURN	4CC		P@CFWRINPL	EC	
P@\$REWORK	4D0		P@CFWRITE	F0	
P@\$ROLL	22C		P@CJFLCMB	19C	
P@\$SCHEMSK	33C		P@CKBINIT	F4	
P@\$SDUMP	528		P@CKPALCLN	F8	
P@\$SEAS	4D4		P@CKPTALOC	FC	
P@\$SEASMSG	4D8		P@CKPTUNAL	100	
P@\$STCK	4DC		P@CKPTVSIZ	104	
P@\$STCKFMT	4E0		P@CKPTXPND	108	
P@\$STIMER	4E4		P@CKRRDONE	160	
P@\$SUBIT	4E8		P@CKRRINIT	164	
P@\$TGMMIG	6A8		P@CKRRMASK	168	
P@\$TGMSET	6AC		P@CKRRSTRT	16C	
P@\$TQLEVEL	6CC		P@CKRRSYNC	170	
P@\$TQLVLS	6D0		P@CKVREFRS	348	
P@\$TRACK	6B0		P@CNVIPAD	67C	
P@\$TTIMER	4EC		P@COFCVE	1A0	
P@\$WAIT	4F0		P@COFEDTR	1A4	
P@\$WSPXCFY	5C4		P@COFJMSG	1A8	
P@\$WSPXKEY	5C8		P@COFRTC	1AC	
P@\$WTO	1DC		P@COMBEWTO	1B0	
P@\$WTOC	1E0		P@COMFRELK	1B4	
P@\$XCWCRT	5CC		P@COMLOPER	6B4	
P@\$XCWDLT	5D0		P@CSCANDSP	1B8	
P@\$XCWPOST	5D4		P@CSMICMD	1BC	
P@\$XCWPRG	5D8		P@CSV\$DEL	1F8	
P@\$XECBKIL	4F4		P@CSV\$LOAD	1FC	
P@ABNDRATE	694		P@CWTO	1C0	
P@ADDCTQ	560		P@CWTOT	1C4	
P@ADDTOINX	2B0		P@DADADDWQ	600	
P@APPLDYN	5EC		P@DADAVAIL	604	
P@ARODREG	80		P@DADCKALL	608	
P@AROQRYA	84		P@DADCKTGM	60C	
P@BERTFMT	128		P@DADCOUNT	610	
P@BERTMAP	12C		P@DADDEB	614	
P@CALCBRTN	564		P@DADEXIST	618	
P@CATAGRP	738		P@DADREMVE	61C	
P@CATCLEAN	73C		P@DADREMWQ	620	
P@CATDGRP	740		P@DADSPLST	628	
P@CATDUP	340		P@DADSTUNT	624	

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
P@DADTGM	62C		P@KRELEASE	110	
P@DADTGMSP	630		P@KRESERVE	114	
P@DADXTENT	634		P@KSETMSTR	154	
P@DILJCAN	1C8		P@KTRK1IO	158	
P@DUPJOB	34C		P@LNEAVRJE	53C	
P@DUPTRANS	744		P@LOCENTRY	200	
P@DYNFSS	240		P@LOCLMOD	204	
P@ENFPOLCY	390		P@LOCMODMP	208	
P@EXTDCTSL	654		P@LPRMLIBP	668	
P@GETEVNTR	4F8		P@MIGRRECV	638	
P@GETJOBKY	4FC		P@MLMRCPCL	540	
P@GTSCREEN	2B4		P@MLMVFY	7C	
P@GTSPPOOL	2B8		P@MNENF58	2D4	
P@HASPBACT	24		P@MNENF70	35C	
P@HASPBPPO	20		P@MNENF78	360	
P@HASPBSA	C		P@MODESWIT	74C	
P@HASPBSLN	2C		P@MODJCHG	364	
P@HASPBUPT	28		P@MODJRLS	368	
P@HASPEXDS	23C		P@MODJSPN	36C	
P@HASPNSNR	420		P@MODJXMBR	370	
P@HASPROUT	10		P@MOD875	500	
P@HASPSACB	44		P@MPURIO	88	
P@HASPSICE	48		P@MSAFCHK	544	
P@HASPSIDL	3C		P@MSTNTFY	6C4	
P@HASPSLNE	38		P@NADRECV	39C	
P@HASPSLOG	34		P@NCOMMREQ	424	
P@HASPSNAA	14		P@NEWSCRE	658	
P@HASPSPRO	30		P@NGTREC	3A8	
P@HASPSRAT	4C		P@NJDTINT	3EC	
P@HASPSAL	50		P@NJEHECK	3F0	
P@HASPSUNT	40		P@NJEHDCV	3F4	
P@HASPTACT	58		P@NJEHDRD	3F8	
P@HASPTASV	64		P@NJEHDWR	3FC	
P@HASPTCPA	18		P@NJEHDXMT	400	
P@HASPTIDL	5C		P@NJEPUT	404	
P@HASPTPRO	54		P@NJERDACT	408	
P@HASPTSSV	68		P@NJHBUILD	40C	
P@HASPTUNT	60		P@NJTBUILD	410	
P@HASPWQUE	1E4		P@NMAPINIT	428	
P@HASPWQUW	1E8		P@NMDRECV	3B4	
P@HASPXFRA	1C		P@NNTRECV	3C0	
P@HASP051	748		P@NPMHOT	42C	
P@HEXTINIT	698		P@NPMQSUSE	430	
P@H607RSN	1CC		P@NPMVFY	3D4	
P@ISSWTO	5E8		P@NPVDCTV	3D8	
P@IVATE	56C		P@NQPSOQ	6BC	
P@JNRNGCNT	350		P@NQRELSE	6C0	
P@JOBQSAMP	354		P@NRMAJUST	444	
P@JOECLUP	2BC		P@NRMRECV	3CC	
P@JOEPPSCR	2C0		P@NSETESS	414	
P@JOEPPSCRN	2C4		P@NSETSUBS	434	
P@JOESYNC	2C8		P@NSJFSPSP	418	
P@JOTFRECL	2CC		P@NSMFBSIZ	41C	
P@JOTVERIF	2D0		P@OPGROUP	244	
P@JQECAT	358		P@OPMAILMG	248	
P@KBUPDJQE	130		P@OPNULLCK	24C	
P@KBUPDSUB	134		P@PROCALOC	190	
P@KCPYMSTR	138		P@PRTDFLT	228	
P@KDIALOG	10C		P@PSAFSCAN	65C	
P@KFORMAT	13C		P@PSOFRELK	510	
P@KGETCHLG	140		P@PURMASC	6B8	
P@KIOERROR	144		P@QBERTHRE	374	
P@KPRIMW	148		P@QDECHAIN	378	
P@KPROTECT	14C		P@QQEVER	37C	
P@KREAD2	150		P@QQSESTAT	66C	

## \$PADDR Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
P@QWLMSVDF	15C		P@XPURJWEL	734	
P@RBLDCTQ	570		P@ZAPJOB	388	
P@RCDSYNC	63C		PADDR	0	
P@RDRPDCUP	52C		PADDR@OCOOFST		
P@REMDCTQ	574			8	
P@RINTJOB	530		PADDRID	0	D7C1C4D9
P@RMTDVINT	548		PADDRLEN	770	774
P@RMTDVSET	54C		PADDRV	4	
P@RMTLNECK	550		PADDRVN	4	6
P@RMTSETUP	554				
P@ROTDCTQ	578				
P@RTNINDEX	2D8				
P@SAIHOT	558				
P@SAIRECC	55C				
P@SAIPOST	2DC				
P@SHRLIVE	184				
P@SIGIO	640				
P@SNASNET	5F0				
P@SNFQBLD	648				
P@SNFQPST	64C				
P@SNFQUE	644				
P@SOCKDYN	680				
P@SPCIOT	5F4				
P@SRVCFSEL	57C				
P@SRVDCTD	580				
P@SRVFNDCCR	584				
P@SRVMOD	588				
P@SRVM630	58C				
P@SRVOLOC	590				
P@SRVPREFX	594				
P@SRVRDIR	59C				
P@SRVROUT	5A0				
P@SRVSASCN	5A4				
P@SRVSETUP	5A8				
P@SRVWSCAN	5AC				
P@SUBDEST	504				
P@SUBRRT	5B0				
P@SUBSPERF	664				
P@TCPSNET	684				
P@TIMECLOC	750				
P@TRCDUMP	230				
P@TRCPUT	234				
P@TREGROUP	50C				
P@TRGETTB	238				
P@UNSHRLIV	188				
P@VETIVATE	5B4				
P@WLMDEQ	380				
P@WLMENQ	384				
P@WLMGOALS	754				
P@WS2	5B8				
P@WTORTIMR	69C				
P@XCFBCAST	6D4				
P@XCFDHOMO	6D8				
P@XCFHELTH	6DC				
P@XCFJOIN	6E0				
P@XCFLEAVE	6E4				
P@XCFMAPEV	6E8				
P@XCFMSTAT	6EC				
P@XCFQSTAT	6F0				
P@XCFUSTAT	6F4				
P@XCFXMAQU	6F8				
P@XCSAPST	5BC				
P@XDUPTEST	758				
P@XINSTART	75C				
P@XPOSTXEQ	760				



## \$PARMLST Information

### \$PARMLST Heading Information

**Common Name:** JES2 inline parameter list DSECT  
**Macro ID:** \$PARMLST  
**DSECT Name:** PARMLIST  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: N/A  
 Key: N/A  
 Residency: This macro is generated as inline code as part of macro expansions using \$CALL INLINE=. It can therefore reside in code anywhere in storage in any address space.  
**Size:** Variable  
**Created by:** Created at assembly time by \$CALL with the INLINE= parameter.  
**Pointed to by:** N/A  
**Serialization:** None required  
**Function:** This DSECT defines inline parameter lists associated with the \$CALL macro via the INLINE= parameter. See \$CALL for more information.

This DSECT is composed of a base section followed by many members which ORG back over this base section. Each \$PARMLST member represents an inline parameter list for a particular service routine. In order to use \$CALL's cross assembly calling ability and have an inline parameter list, the inline parameter list must be defined as a member of this DSECT.

### \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PARMLIST	INLINE PARAMETER LIST DSECT
0	(0)	BITSTRING	4	PARMINST	FOR INSTRUCTION AFTER THE BASR
4	(4)	SIGNED	2	PARMSTRT (0)	LABEL ALL \$PARMLST MEMBERS ORG TO
Comment					
MEMBER NAME --> \$\$PO					
ROUTINE(S) ---> \$\$POST in HASCSRIC					
MACRO(S) ----> \$\$POST					
Wake up the JES2 main task					
End of Comment					
4	(4)	BITSTRING	1	\$\$POFLG1	\$\$POST flag byte
		1... ....		\$\$PO1BRA	"B'10000000" LINKAGE=BRANCH POST
		.1... ....		\$\$PO1SYS	"B'01000000" LINKAGE=SYSTEM POST
Comment					
B'00xxxxxx' LINKAGE=SVC POST					
End of Comment					
		..1. ....		\$\$PO1ELM	"B'00100000" ELMT specified
		...1 ....		\$\$PO1RUN	"B'00010000" Run time \$DRxxx value
		.... 1...		\$\$PO1BR3	"B'00001000" LINKAGE=CVT0PT03 POST
5	(5)	ADDRESS	1		Reserved

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$AEOJ ROUTINE(S) ---> ARMEQJ in HASCARSO MACRO(S) ----> none ARM end of job routine. This parameter list is FROZEN.					
End of Comment					
4	(4)	BITSTRING 1... .... .1... ....	1	\$AEOJFL1 \$AEOJ1JT \$AEOJ1EM	\$AEOJ FLAG BYTE "B'10000000" Job termination call "B'01000000" End of memory call
Comment					
MEMBER NAME --> \$BLDTGB ROUTINE(S) ---> \$BLDTGB in HASPTRAK MACRO(S) ----> \$BLDTGB Build trackgroup block					
End of Comment					
4	(4)	BITSTRING 1... .... .1... ....	1	\$BTGFLG1 \$BTGBMTR \$BTGBTGM	\$AEOJ FLAG BYTE "B'10000000" ID=MTTR was specified "B'01000000" ID=TGM was specified
Comment					
MEMBER NAME --> \$CBI ROUTINE(S) ---> \$CBIO IN HASC SRDS \$CBIOM in HASPNUC MACRO(S) ----> \$CBIO CONTROL BLOCK I/O ROUTINE.					
End of Comment					
4	(4)	BITSTRING	1	\$CBIFLG1	\$CBIO flag byte
Comment					
EQU B'10000000' Reserved					
End of Comment					
		.1... ....		\$CB1EXIT	"B'01000000" EXIT 8 SHOULD BE TAKEN
		..1. ....		\$CB1NOVF	"B'00100000" BYPASS CNTRL BLK VERIFY
		...1 ....		\$CB1NSJB	"B'00010000" NO SJB PROVIDED
		.... 1..		\$CB1SJIO	"B'00001000" SJIOB PROVIDED
		.... .1..		\$CB1FREE	"B'00000100" FREE THE BUFFER
		.... ..1.		\$CB1WAIT	"B'00000010" ON - WAIT=YES SPECIFIED, OFF - WAIT=NO SPECIFIED.
5	(5)	.... ..1 BITSTRING 1... .... .1... .... ..1. .... ...1 .... .... 1..	1	\$CB1COND \$CBIFLG2 \$CB2WRIT \$CB2TWAT \$CB2FSSM \$CB2SUPM \$CB2MQTR \$CB2SPLQ \$CB2NORF	"B'00000001" Conditional Write \$CBIO flag byte "B'10000000" TYPE=WRITE operation "B'01000000" TYPE=WAIT requested "B'00100000" \$CBIO called from FSSM "B'00010000" Suppress error messages "B'00001000" MQTR passed in register 0 "B'00000100" SPOLPTR is an MQTR "B'00000010" WAIT=(NO,NOREF) SPECIFIED
6	(6)	BITSTRING	1		Reserved
7	(7)	BITSTRING	1	\$CBCKPTB	CKPTBIT VALUE
8	(8)	ADDRESS	2	\$CBSTORP	STORPTR VALUE
10	(A)	ADDRESS	2	\$CBSPOLP	SPOLPTR VALUE
12	(C)	ADDRESS	2	\$CBCKPTF	CKPTFLD VALUE
14	(E)	CHARACTER	4	\$CBVERID	VERIFY ID

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
18	(12)	ADDRESS	4	\$CBVERIX	Verify index (if known)

Comment

MEMBER NAME --> \$VERIFY  
 ROUTINE(S) ---> \$VERIFY in HASCSRDS  
 MACRO(S) -----> \$VERIFY  
 CBIO control block verification service

-----  
 \$VERIFY passes parameters in registers and does not use inline parameter list. However, one of the parameters is a pointer to 4 byte control block ID. It can be a 4 character EBCDIC identifier of a control block (e.g. 'IOT '); or 4 byte binary index in the verification table HASPVTAB (in HASCSRDS). Note that equates in this list should be in the same order as entries in HASPVTAB.  
 -----

End of Comment

18	(12)	X'0'	0	\$VFYCHK	"0" Verify CHK
18	(12)	X'1'	0	\$VFYDSCA	"1" Verify DSCA
18	(12)	X'2'	0	\$VFYDSIX	"2" Verify DSIX
18	(12)	X'3'	0	\$VFYHDB	"3" Verify HDB
18	(12)	X'4'	0	\$VFYIOT	"4" Verify IOT
18	(12)	X'5'	0	\$VFYJCT	"5" Verify JCT
18	(12)	X'6'	0	\$VFYOCT	"6" Verify OCT
18	(12)	X'7'	0	\$VFYSWBI	"7" Verify SWBI
18	(12)	X'8'	0	\$VFYNHSB	"8" Verify NHSB
18	(12)	X'9'	0	\$VFYTLBM	"9" Verify TLBM
18	(12)	X'A'	0	\$VFYJSMT	"10" Verify JSMT

Comment

MEMBER NAME --> \$CDCTDYN  
 ROUTINE(S) ---> \$CDCTDYN in HASPDYN  
 MACRO(S) -----> \$CDCTDYN

End of Comment

4	(4)	BITSTRING	1	\$CDCTYPE	Flag byte 1
4	(4)	X'1'	0	\$CDCTDCT	"1" DCT= was specified
4	(4)	X'2'	0	\$CDCTAPT	"2" APT= was specified
4	(4)	X'3'	0	\$CDCTSCK	"3" SCK= was specified
4	(4)	X'4'	0	\$CDCTRAT	"4" RAT= was specified
4	(4)	X'5'	0	\$CDCTCDC	"5" CDCT= was specified
4	(4)	X'6'	0	\$CDCTXRQ	"6" XREQ= was specified
5	(5)	BITSTRING	1	\$CDCFLG1	Flag byte 2
		1... ..		\$CDC1CRE	"B'10000000" CREATE=YES
		.1.. ..		\$CDC1SYN	"B'01000000" SYNCH=YES
		..1. ....		\$CDC1BRO	"B'00100000" BROADCAST=YES
		...1 ....		\$CDC1INV	"B'00010000" MARK=INVALID
		.... 1...		\$CDC1VAL	"B'00001000" MARK=VALID
		.... .1..		\$CDC1BRC	"B'00000100" BROADCAST=COND

Comment

MEMBER NAME --> \$CFX  
 ROUTINE(S) ---> CSMICMD IN HASPCOMM  
 MACRO(S) -----> \$CFXMJC  
 Single Member Image subroutine.

End of Comment

4	(4)	BITSTRING	1	\$CFXFLG1	Response flag
---	-----	-----------	---	-----------	---------------

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
5	(5)	1... .. BITSTRING	1	\$CFX1RSP	"B'10000000" Return a command response Reserved for future use
Comment					
MEMBER NAME --> \$CKPTQUE ROUTINE(S) ---> \$CKPTQUE IN HASPCKPT MACRO(S) ----> \$CKPTQUE Queue work to CKPT PCE					
End of Comment					
4	(4)	ADDRESS	4	\$CKQRTN	Routine address
Comment					
MEMBER NAME --> \$CPL ROUTINE(S) ---> CPGET, CPFREE in HASCPPOOL MACRO(S) ----> \$CPOOL CPOOL TYPE=GET SUBROUTINE.					
End of Comment					
4	(4)	BITSTRING	1	\$CPL1	
		1... ..		\$CPL1CDY	"B'10000000" \$CPOOL COND=YES
		.1... ..		\$CPL1HXN	"B'01000000" \$CPOOL HASXB=NONE
Comment					
MEMBER NAME --> \$CRJOES ROUTINE(S) ---> CRJOES in HASCSISC MACRO(S) ----> \$CRJOES Create JOEs or JOA (optionally acquire and free storage for JOA)					
End of Comment					
4	(4)	BITSTRING	1	\$CRJFLG1	
		..1. ....		\$CRJ1ALC	"B'00100000" ALLOCATE and return JOA
		...1 ....		\$CRJ1FRE	"B'00010000" FREEMAIN JOA
		.... 1...		\$CRJ1CLR	"B'00001000" Clear passed JOA
Comment					
MEMBER NAME --> \$CW ROUTINE(S) ---> CWTO IN HASPCOMM MACRO(S) ----> \$CWTO WRITE - TO - OPERATOR SUBROUTINE.					
End of Comment					
4	(4)	ADDRESS	1	\$CWTOFLG	
		1... ..		\$CWTOMVC	"B'10000000" EXECUTE OF MVC INSTRUCT. NEEDED
		.1... ..		\$CWTOLST	"B'01000000" LAST LINE OF MLWTO
		..1. ....		\$CWTONWT	"B'00100000" WAIT=NO WAS SPECIFIED
Comment					
MEMBER NAME --> \$DIL ROUTINE(S) ---> \$DILBERT in HASPJQS MACRO(S) ----> \$DILBERT Do It Later when BERT lock is available routine inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$DILTYPE	TYPE specification
4	(4)	X'1'	0	\$DILTJQE	"1" TYPE=JQE specified
4	(4)	X'2'	0	\$DILTJOE	"2" TYPE=JOE specified

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
5	(5)	BITSTRING	1	\$DILVERS	Version
5	(5)	X'1'	0	\$DILCVER	"1" Current version
6	(6)	BITSTRING	1	\$DILFLG1	Flag byte
		1... ....		\$DILF1CL	"B'10000000" CALL=YES speciied
		.1.. ....		\$DILF1IM	"B'01000000" Execute immediate instruction rather than calling routine
		..1. ....		\$DILF1WA	"B'00100000" \$WAIT for flush
		...1 ....		\$DILF1FL	"B'00010000" Flush DWAs
		.... 1...		\$DILF1PO	"B'00001000" \$POST Resource
		.... .1..		\$DILF1ND	"B'00000100" Do not queue duplicates
		.... .1.		\$DILF1QP	"B'00000010" QPOST when resource ret
		.... ...1		\$DILF1#P	"B'00000001" \$#POST when resource ret
7	(7)	BITSTRING	1	\$DILFLG2	Second Flag byte
		1... ....		\$DILF2PA	"B'10000000" Pace requests by rtn addr
		.1.. ....		\$DILF2QS	"B'01000000" Queues need not be owned
		..1. ....		\$DILF2SP	"B'00100000" Get JQA in special mode
		...1 ....		\$DILF2CK	"B'00010000" Check DWAs
		.... 1...		\$DILF2GM	"B'00001000" GETMAIN'ed DWA
		.... .1..		\$DILF2FN	"B'00000100" Don't queue a DWA if flush unsuccessful
		.... .1.		\$DILF2FP	"B'00000010" Flush only DWAs for this specific PCE
		.... ...1		\$DILF2FT	"B'00000001" Flush only DWAs for this PCE type
8	(8)	BITSTRING	4	\$DILIMME	Immed instruction

Comment

MEMBER NAME --> \$DTR  
 ROUTINE(S) ---> \$DISTERR in HASPRAS  
 MACRO(S) -----> \$DISTERR  
 Disasterous error routine inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$DTRFLG1	General flag 1
		1... ....		\$DTRRJQE	"B'10000000" JQE= specified
		.1.. ....		\$DTRRJCT	"B'01000000" JCT= specified
		..1. ....		\$DTRRND	"B'00100000" DUMP=NO specified
		...1 ....		\$DTRRNAM	"B'00010000" RECVOPTS= specified
		.... 1...		\$DTRRSIG	"B'00001000" SIGRCD= specified
		.... .1..		\$DTRMQTR	"B'00000100" MQTR= specified
5	(5)	BITSTRING	1		Reserved
6	(6)	CHARACTER	8	\$DTRNAME	Name of \$DISTERR
14	(E)	CHARACTER	8	\$DTRSECT	Name of CSECT
22	(16)	CHARACTER	8	\$DTRSEQ	Sequence number of \$DISTERR
30	(1E)	CHARACTER	8	\$DTRRECV	RECVOPTS= value

Comment

MEMBER NAME --> \$DOGJAX  
 ROUTINE(S) ---> \$DOGJAX in HASPSERV  
 MACRO(S) -----> \$DOGJAX

End of Comment

4	(4)	BITSTRING	1	\$DJXREQ	Action/request:
4	(4)	X'1'	0	\$DJXRQCR	"1" ACTION=CREATE
4	(4)	X'2'	0	\$DJXRQFE	"2" ACTION=FETCH
4	(4)	X'3'	0	\$DJXRQCK	"3" ACTION=CKPT
4	(4)	X'4'	0	\$DJXRQRT	"4" ACTION=RETURN
4	(4)	X'5'	0	\$DJXRQFR	"5" ACTION=FREE
4	(4)	X'6'	0	\$DJXRQRS	"6" ACTION=RESET
5	(5)	BITSTRING	1	\$DJXFLGS	\$DOGJAX options:
		1... ....		\$DJXFWT	"B'10000000" WAIT=YES
		.1.. ....		\$DJXFJAX	"B'01000000" JAX addr supplied in R1
		..1. ....		\$DJXFNCK	"B'00100000" do not checkpoint changes

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$DOGJOE ROUTINE(S) ---> \$DOGJOE MACRO(S) ----> \$DOGJOE Deliver or Get JOE routine's inline parameter list					
End of Comment					
0	(0)	SIGNED	1	\$DOVERS	Version of parameter list
0	(0)	X'1'	0	\$DOCVER	"1" Parameter list version
1	(1)	SIGNED	1	\$DOACT	Action requested
1	(1)	X'0'	0	\$DOAFETN	"0" Fetch next JOE
1	(1)	X'4'	0	\$DOAFET	"4" Fetch JOE
1	(1)	X'8'	0	\$DOARET	"8" RETURN JOE (CKPT and Rel)
1	(1)	X'C'	0	\$DOACKPT	"12" CKPT JOE (CKPT, *no* Rel)
1	(1)	X'10'	0	\$DOAFLD	"16" CKPTFLD
1	(1)	X'14'	0	\$DOAREFR	"20" Refresh JOA
1	(1)	X'18'	0	\$DOAFREE	"24" Free JOA
1	(1)	X'1C'	0	\$DOAQLOK	"28" QUERYLOCK
1	(1)	X'20'	0	\$DOASETA	"32" SETACCESS
2	(2)	BITSTRING	1	\$DOFLAG2	More \$DOGJOE option flags
		1... ..		\$DO2DSRV	"B'10000000" DSERV provided
		.1.. ..		\$DO2SPCL	"B'01000000" Special call (no BERT lock)
		..1. ....		\$DO2READ	"B'00100000" READ access requested
		.... 1...		\$DO2NWAT	"B'00010000" WAIT=NO
		.... .1..		\$DO2WDEF	"B'00001000" DEFER option on WAIT=NO
		.... ..1.		\$DO2CONF	"B'00000100" Conditional FREE
		.... ...1		\$DO2NROL	"B'00000010" Skip any \$ROLL trace
		.... ....1		\$DO2RCVY	"B'00000001" ACTION=(FREE,RECOVERY)
3	(3)	BITSTRING	1	\$DOFLAG3	More \$DOGJOE option flags
		1... ..		\$DO3RELE	"B'10000000" Release BERT lock
		.1.. ....		\$DO3KEEP	"B'01000000" Keep memory for JOA
		..1. ....		\$DO3UCON	"B'00100000" Unconditional return for ACTION=RETURN
		...1 ....		\$DO3NUPD	"B'00010000" RETURN,NOUPDATE
		.... 1...		\$DO3QLOB	"B'00001000" QUERYLOCK,OBTAINABLE
		.... ..1.		\$DO3MAX	"B'00000010" ACTION=(CKPT,MAXJOA)
		.... ....1		\$DO3POST	"B'00000001" POST=YES for ACTION=CKPT
4	(4)	BITSTRING	1	\$DOFLAG4	More \$DOGJOE option flags
		1... ..		\$DO4PSTA	"B'10000000" Post \$ACTIVATE FETCH call
		.1.. ....		\$DO4#PSY	"B'01000000" #POST=YES
		..1. ....		\$DO4KPJW	"B'00100000" #POST=(,KEEPJWEL)
		.... 1...		\$DO4MNJT	"B'00010000" #POST=(,JOETIME)
5	(5)	BITSTRING	1	\$DOFLAG5	More \$DOGJOE option flags
6	(6)	SIGNED	2	\$DOACTOR (0)	Action specific fields
6	(6)	SIGNED	2	\$DOCHAIN	Offset of chaining field (present only if \$DOFETN)
6	(6)	SIGNED	2	\$DOCKOFF	Field offset and length
8	(8)	SIGNED	2	\$DOCKLEN	for CKPTFLD request (present only if \$DOAFLD)
8	(8)	X'A'	0	\$DOLEN	** -PARMLIST" Length of \$DOGJOE MF=L
Comment					
MEMBER NAME --> \$DOGJQE ROUTINE(S) ---> \$DOGJQE MACRO(S) ----> \$DOGJQE Deliver or Get JQE routine's inline parameter list					
End of Comment					
0	(0)	SIGNED	1	\$DJACT	Action requested
0	(0)	X'0'	0	\$DJAFETN	"0" Fetch next JQE
0	(0)	X'4'	0	\$DJAFET	"4" Fetch JQE
0	(0)	X'8'	0	\$DJALOCK	"8" Manage BERT lock
0	(0)	X'C'	0	\$DJARET	"12" RETURN JQE (CKPT and Rel)
0	(0)	X'10'	0	\$DJACKPT	"16" CKPT JQE (CKPT, *no* Rel)

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	X'14'	0	\$DJAREFR	"20" Refresh JQA
0	(0)	X'18'	0	\$DJAFREE	"24" Free JQA
0	(0)	X'1C'	0	\$DJASETA	"28" Set access
0	(0)	X'20'	0	\$DJAQLOK	"32" QUERYLOCK
0	(0)	X'24'	0	\$DJAFLD	"36" CKPTFLD
1	(1)	BITSTRING	1	\$DJFLAG2	More \$DOGJQE option flags
		1... ....		\$DJ2DSRV	"B'10000000" DSERV provided
		.1.. ....		\$DJ2SPCL	"B'01000000" Special call (no BERT lock)
		..1. ....		\$DJ2KEEP	"B'00100000" Keep memory for JQA
		...1 ....		\$DJ2NWAT	"B'00010000" WAIT=NO or QUERYLOCK,OBTAINABLE
		.... 1..		\$DJ2CONF	"B'00001000" Conditional FREE
		.... .1..		\$DJ2POST	"B'00000100" POST=YES for ACTION=CKPT
		.... ..1.		\$DJ2UCON	"B'00000010" Unconditional return for ACTION=RETURN
		.... ...1		\$DJ2URFR	"B'00000001" Unconditional refresh
2	(2)	BITSTRING	1	\$DJFLAG3	More \$DOGJQE option flags
		1... ....		\$DJ3READ	"B'10000000" READ access requested
		.1.. ....		\$DJ3RELE	"B'01000000" Release BERT lock
		..1. ....		\$DJ3WDEF	"B'00100000" Defer RETURN if required
		...1 ....		\$DJ3NUPD	"B'00010000" RETURN,NOUPDATE
		.... 1..		\$DJ3QPSY	"B'00001000" QPOST=YES
		.... .1..		\$DJ3#PSY	"B'00000100" #POST=YES
		.... ..1.		\$DJ3MAX	"B'00000010" ACTION=(CKPT,MAXJQA)
		.... ...1		\$DJ3RCVY	"B'00000001" ACTION=(FREE,RECOVERY)
3	(3)	SIGNED	1	\$DJVERS	Version of parameter list
3	(3)	X'2'	0	\$DJCVER	"2" Parameter list version
4	(4)	BITSTRING	1	\$DJFLAG4	More \$DOGJQE option flags
5	(5)	BITSTRING	1	\$DJFLAG5	More \$DOGJQE option flags
6	(6)	SIGNED	2	\$DJCHAIN	Offset of chaining field (present only if \$DJFETN)
6	(6)	X'8'	0	\$DJLEN	** -PARMLIST" Length of \$DOGJQE MF=L

Comment

MEMBER NAME --> \$DST  
 ROUTINE(S) ---> USERDEST IN HASCSIRQ  
 MACRO(S) -----> \$DEST  
 DESTINATION CONVERSION ROUTINE'S INLINE PARAMETER LIST.

End of Comment

4	(4)	BITSTRING	1	\$DSTFLG1	\$DEST MACRO OPTION FLAGS
		1... ....		\$DSTCHAR	"B'10000000" CHARACTER INPUT
		.1.. ....		\$DSTRDT	"B'01000000" DESTIDs allowed
		..1. ....		\$DSTGNRC	"B'00100000" Generic userids allowed at local node
		...1 ....		\$DST1EXP	"B'00010000" EXPLICIT=YES was specified
		.... 1..		\$DSTNRP	"B'00001000" RMTPOOL=NO WAS REQUESTED
		.... .1..		\$DSTPRIM	"B'00000100" PRIMARY=YES, RETURN NODENM
		.... ..1.		\$DSTUSER	"B'00000010" USERID SUPPLIED OR DESIRED
		.... ...1		\$DSTNSPR	"B'00000001" DO NOT SUPPRESS NODE FOR LOCAL BINARY TO CHARACTER CONV
5	(5)	BITSTRING	1	\$DSTFLG2	\$DEST macro options flag 2
		1... ....		\$DST2IGN	"B'10000000" NODENAME=IGNORED
		.1.. ....		\$DST2DFM	"B'01000000" DLMFAIL=YES
		..1. ....		\$DST2NUS	"B'00100000" DONTUSE= was specified
		...1 ....		\$DST2IPY	"B'00010000" IPFORM=YES (or LONG) was specified
		.... 1..		\$DST2IGS	"B'00001000" SHOWUSER=IGNORED
		.... .1..		\$DST2IPD	"B'00000100" IPFORM=SHORT was specified
		.... ...1		\$DST2NVU	"B'00000010" VALUSR=NO was specified

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$DSD					
ROUTINE(S) ---> \$DESTDYN IN HASPDYN					
MACRO(S) ----> \$DESTDYN					
DESTINATION DEFINITION ROUTINE'S INLINE PARAMETER LIST.					
End of Comment					

4	(4)	BITSTRING	1	\$DSDFLG1	\$DESTDYN MACRO OPTION FLAGS (VALUES ARE RDT1NODE/RDT1DEST)
5	(5)	BITSTRING 1... ....	1	\$DSDFLG2 \$DSD2FOR	\$DESTDYN MACRO OPTION FLAG 2 "B'10000000" Force NODAL destination

Comment					
MEMBER NAME --> \$DGB					
ROUTINE(S) ---> \$DOGBERT in HASPCKPT BERTREAD in HASCSRIC					
MACRO(S) ----> \$DOGBERT					
BERT Deliver and Get services inline parm list					
End of Comment					

4	(4)	SIGNED	1	\$DGBACT	ACTION= being requested
4	(4)	X'1'	0	\$DGBFTCH	"1" FETCH action
4	(4)	X'2'	0	\$DGBNEXT	"2" FETCHNEXT action
4	(4)	X'3'	0	\$DGBCKPT	"3" CKPT action
4	(4)	X'4'	0	\$DGBRETN	"4" RETURN action
4	(4)	X'5'	0	\$DGBFREE	"5" FREE action
4	(4)	X'6'	0	\$DGBSTSP	"6" SETSPECIAL action
5	(5)	BITSTRING	1	\$DGBFLAG	General parameter flags
		1... ....		\$DGBWAIT	"B'10000000" \$WAIT is allowed
		.1.. ....		\$DGBQSUS	"B'01000000" Get the queues
		..1. ....		\$DGBUPDT	"B'00100000" Update access needed
		...1 ....		\$DGBNAME	"B'00010000" NAME= was passed
		.... 1..		\$DGBTOKN	"B'00001000" TOKEN= was passed
		.... .1..		\$DGBSPEC	"B'00000100" SPECIAL=YES was specified
		.... ..1.		\$DGBNUPD	"B'00000010" No write update requested
		.... ...1		\$DGBNRDD	"B'00000001" No read data requested

Comment

-----

The following must match the values in BRTTYPE in the \$BERT control block.

-----

End of Comment					
6	(6)	SIGNED	1	\$DGBCB	Control block type
6	(6)	X'0'	0	\$DGBINT	"0" Internal control block
6	(6)	X'1'	0	\$DGBJQE	"1" JQE control block ext
6	(6)	X'2'	0	\$DGBCAT	"2" CAT control block
6	(6)	X'3'	0	\$DGBWSCQ	"3" WLM Service Class Queue
		1111 1111		\$DGBDYN	"X'FF" Dynamically defined type
7	(7)	BITSTRING	1	\$DGBFLG2	Second flag byte
		1... ....		\$DGB2CRE	"B'10000000" ACTION=(,CREATE) spec
		.1.. ....		\$DGB2UNK	"B'01000000" CB type unknown
		..1. ....		\$DGB2PAD	"B'00100000" ACTION=(CKPT,PAD)
		...1 ....		\$DGB2PBE	"B'00010000" Flag bytes 1 and 2 in PBEDGBF1 & PBEDGBF2
		.... 1..		\$DGB2NEV	"B'00001000" MOREBERTS=NEVER
		.... .1..		\$DGB2UNC	"B'00000100" ACTION=(CKPT,UNCOND)



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
-----					
The following field is only generated for dynamic BERT types					
-----					
End of Comment					
8	(8)	CHARACTER	8	\$DGBTNAM	BERT type in EBCDIC
Comment					
MEMBER NAME --> \$DGD					
ROUTINE(S) ---> \$DOGDJB routine in HASPJQS					
ROUTINE(S) ---> DJBREAD routine in HASCSRIC					
MACRO(S) ----> \$DOGDJB					
Deliver Or Get Duplicate Jobname Block					
End of Comment					
4	(4)	BITSTRING	1	\$DGDFLG1	\$DOGDJB Macro options
		1... ....		\$DGD1FET	"B'1000000" ACTION=FETCH
		.1.. ....		\$DGD1FTN	"B'0100000" ACTION=FETCHNEXT
		..1. ....		\$DGD1UPD	"B'0010000" ACTION=(...,UPDATE)
		...1 ....		\$DGD1NUP	"B'0001000" ACTION=(...,NOUPDATE)
		.... 1..		\$DGD1FRE	"B'00001000" ACTION=FREE
		.... .1..		\$DGD1BRO	"B'00000100" MOREBERTS=NEVER
		.... ..1.		\$DGD1CRE	"B'00000010" ACTION=(FETCH,CREATE)
		.... ...1		\$DGD1RES	"B'00000001" ACTION=RESET
5	(5)	BITSTRING	1	\$DGDFLG2	\$DOGDJB Macro options
		1... ....		\$DGD2ACC	"B'1000000" ACTION=SETACCESS
		.1.. ....		\$DGD2CAC	"B'0100000" CACHE=YES
		..1. ....		\$DGD2WAI	"B'0010000" WAIT=NO
Comment					
MEMBER NAME --> \$DGG					
ROUTINE(S) ---> \$DOGGRP routine in HASPXEQ					
MACRO(S) ----> \$DOGGRP					
Deliver Or Get class group in support of 8 character jobclass.					
End of Comment					
4	(4)	BITSTRING	1	\$DGGFLG1	\$DOGGRP Macro options
		1... ....		\$DGG1CRE	"B'1000000" ACTION=(FETCH,CREATE)
		.1.. ....		\$DGG1RED	"B'0100000" ACTION=(FETCH,READ)
		..1. ....		\$DGG1FET	"B'0010000" ACTION=(FETCH...) UPDATE
		...1 ....		\$DGG1UPD	"B'0001000" ACTION=(RETURN,UPDATE)
		.... 1..		\$DGG1NUP	"B'00001000" ACTION=(RETURN,NOUPDATE)
		.... .1..		\$DGG1FRE	"B'00000100" ACTION=FREE
		.... ..1.		\$DGG1CKP	"B'00000010" ACTION=CKPT
		.... ...1		\$DGG1WAI	"B'00000001" WAIT=NO
5	(5)	BITSTRING	1	\$DGGFLG2	\$DOGGRP Macro options
		1... ....		\$DGG2FTN	"B'1000000" ACTION=FETCHNEXT
Comment					
MEMBER NAME --> \$DGV					
ROUTINE(S) ---> \$DOGMIG routine in HASPSPOL					
MACRO(S) ----> \$DOGMIG					
Deliver Or Get temporary migration object in support of spool volume migration.					
End of Comment					

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
4	(4)	BITSTRING	1	\$DGVFLG1	\$DOGMIG Macro options
		1... ..		\$DGV1FET	"B'10000000" ACTION=(FETCH,UPDATE)
		.1... ..		\$DGV1CRE	"B'01000000" ACTION=(FETCH,CREATE)
		..1... ..		\$DGV1RED	"B'00100000" ACTION=(FETCH,READ)
		...1... ..		\$DGV1UPD	"B'00010000" ACTION=(RETURN,UPDATE)
		.... 1... ..		\$DGV1NUP	"B'00001000" ACTION=(RETURN,NOUPDATE)
		.... .1... ..		\$DGV1FRE	"B'00000100" ACTION=FREE
		.... ..1... ..		\$DGV1CKP	"B'00000010" ACTION=CKPT
		.... ...1... ..		\$DGV1WAI	"B'00000001" WAIT=NO

Comment

MEMBER NAME --> \$DGT  
 ROUTINE(S) ---> \$DOGCAT routine in HASPXEQ  
 MACRO(S) ----> \$DOGCAT  
 Deliver Or Get Class Attribute Table

End of Comment

4	(4)	BITSTRING	1	\$DGTFLG1	\$DOGCAT Macro options
		1... ..		\$DGT1FET	"B'10000000" ACTION=FETCH
		.1... ..		\$DGT1FTN	"B'01000000" ACTION=FETCHNEXT
		..1... ..		\$DGT1UPD	"B'00100000" ACTION=(...,UPDATE)
		.... .1... ..		\$DGT1AQS	"B'00000100" ALLQUES=YES specified
		.... ..1... ..		\$DGT1AQR	"B'00000010" ALLQUES=(YES,REBLD)
		.... ...1... ..		\$DGT1TYP	"B'00000001" TYPE= was specified
5	(5)	BITSTRING	1	\$DGTFLG2	\$DOGCAT Macro options
		1... ..		\$DGT2BRO	"B'10000000" MOREBERTS=NEVER
		.1... ..		\$DGT2WAI	"B'01000000" WAIT=NO
		..1... ..		\$DGT2JBC	"B'00100000" JOBCLASS= was specified
		...1... ..		\$DGT2CRE	"B'00010000" ACTION=(FETCH,CREATE)
		.... 1... ..		\$DGT2FRE	"B'00001000" ACTION=FREE
		.... ..1... ..		\$DGT2CPY	"B'00000100" GETCOPY was specified.
		.... ...1... ..		\$DGT2BCH	"B'00000010" BATCHONLY=YES specified

Comment

MEMBER NAME --> \$DGW  
 ROUTINE(S) ---> \$DOGWSCQ routine in HASPXEQ  
 MACRO(S) ----> \$DOGWSCQ  
 Deliver Or Get Service Class Queue

End of Comment

4	(4)	BITSTRING	1	\$DGWFLG1	\$DOGWSCQ Macro options
		1... ..		\$DGW1FET	"B'10000000" ACTION=FETCH
		.1... ..		\$DGW1FTN	"B'01000000" ACTION=FETCHNEXT
		..1... ..		\$DGW1UPD	"B'00100000" ACTION=(...,UPDATE)
		...1... ..		\$DGW1NUP	"B'00010000" ACTION=(...,NOUPDATE)
		.... 1... ..		\$DGW1FRE	"B'00001000" ACTION=FREE
		.... .1... ..		\$DGW1CKP	"B'00000100" ACTION=CKPT
		.... ..1... ..		\$DGW1CRE	"B'00000010" ACTION=(FETCH,CREATE)
		.... ...1... ..		\$DGW1BRO	"B'00000001" MOREBERTS=NEVER
5	(5)	BITSTRING	1	\$DGWFLG2	\$DOGWSCQ Macro options
		1... ..		\$DGW2WAI	"B'10000000" WAIT=NO

Comment

MEMBER NAME --> \$DSN  
 ROUTINE(S) ---> DAATSET SET NAME VERIFICATION IN HASPSRDS  
 MACRO(S) ----> \$DSNVFY  
 DESTINATION DEFINITION ROUTINE'S INLINE PARAMETER LIST.

End of Comment

.... ..	DSNVALL	"B'00000000" COMPLETE DATASET NAME VERIFICATION
1... ..	DSNRONLY	"B'10000000" RESERVE WORD ONLY VERIFICATION

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$DSR ROUTINE(S) ---> GOFDSERV ROUTINE IN HASCSRIC MACRO(S) -----> \$DSERV OBTAIN OR RELEASE A CHECKPOINT VERSION					
End of Comment					
4	(4)	BITSTRING	1	\$DSRFLG1	Flag byte 1
		1... ....		\$DSR1GET	"B'10000000" GET request
		.1.. ....		\$DSR1FRE	"B'01000000" FREE request
		..1. ....		\$DSR1LIV	"B'00100000" Use "live" version
		...1 ....		\$DSR1RFR	"B'00010000" Refresh non checkpoint resident data pointers
Comment					
MEMBER NAME --> \$DV ROUTINE(S) ---> CNVDEVID ROUTINE IN HASCSISC MACRO(S) -----> \$DVIDBLD CONVERT A DEVID TO A DEVICE NAME					
End of Comment					
4	(4)	BITSTRING	1	\$DVFLG1	Flags
		1... ....		\$DV1JQE	"B'10000000" JQE address passed
		.1.. ....		\$DV1CHAR	"B'01000000" CONV=CHAR specified
5	(5)	SIGNED	1	\$DVLENG	Length of device name field
Comment					
MEMBER NAME --> \$EST ROUTINE(S) ---> SSISESTA in HASCLINK MACRO(S) -----> \$ESTAE (assembler) \$ESTAEP (PL/X) JES2 Establish ESTAE Inline Parameter List.					
End of Comment					
4	(4)	BITSTRING	1	\$ESTFCN	Requested function
		1... ....		\$ESTCRAT	"B'10000000" Create
		.1.. ....		\$ESTDLET	"B'01000000" Delete
5	(5)	BITSTRING	1	\$ESTFLAG	Flags
		1... ....		\$ESTFNDM	"B'10000000" Do not capture dump
6	(6)	ADDRESS	4	\$ESTRECX	Recovery exit addr if any
10	(A)	SIGNED	2	\$ESTNBR	Number of retry addresses - currently always 1
12	(C)	ADDRESS	4	\$ESTRTYA	Retry address
Comment					
MEMBER NAME --> EXI ROUTINE(S) ---> \$JESEFF IN HASPNUC, USERSUB IN HASCSIRQ MACRO(S) -----> \$EXIT JES2 EXIT EFFECTOR'S INLINE PARAMETER LIST.					
End of Comment					
4	(4)	CHARACTER	8	EXITNAME	LABEL ON \$EXIT OR CSECT NAME IF NO LABEL WAS SPECIFIED
12	(C)	BITSTRING	1	EXITFLGS	EXIT FLAGS

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>The bits EXITUSER, EXITSTSK, EXITJES2 and EXITFSS have a one to one correspondence to the following bits in \$TTE : TDXFENVU, TDXFENV5, TDXFENVJ and TDXFENVF. These bits MUST remain in the currently defined order.</p>					
End of Comment					
		1... ....		EXITTR	"B'10000000" EXIT EFFECTOR SHOULD DO TRACE
		.1.. ....		EXITUSER	"B'01000000" USER ENVIRONMENT
		..1. ....		EXITSTSK	"B'00100000" SUBTASK ENVIRONMENT
		...1 ....		EXITJES2	"B'00010000" JES2 ENVIRONMENT
		.... 1...		EXITFSS	"B'00001000" FSS ENVIRONMENT
		.... .1..		EXITXPL	"B'00000100" \$XPL passed to exit
13	(D)	ADDRESS	1	EXITID	EXIT ID
14	(E)	ADDRESS	1	EXITMRC	MAXIMUM RETURN CODE
15	(F)	ADDRESS	1	EXITRSVD	RESERVED FOR FUTURE USE
15	(F)	X'C'	0	EXITLNG	"(*-PARMSTRT+1)/2*2" LENGTH OF EXIT PARAMETER LIST
Comment					
<p>MEMBER NAME --&gt; \$FAC            ROUTINE(S) ---&gt; \$FMSTACC in HASMIPSV            MACRO(S) -----&gt; \$FMSTACC and \$FMSTFMT            \$FMSTACC and \$FMSTFMT inline parameter list</p>					
End of Comment					
4	(4)	CHARACTER	8	\$FACSECT	Control section name
12	(C)	CHARACTER	8	\$FACSEQF	Invoking sequence number
12	(C)	X'4'	0	\$FACMOSQ	"\$FACSECT,*-\$FACSECT,C'C" Module/sequence
Comment					
<p>MEMBER NAME --&gt; \$FB            ROUTINE(S) ---&gt; \$MLTFBUF IN HASCLINK            MACRO(S) -----&gt; \$FREEBUF TYPE=MULT            \$FREEBUF'S INLINE PARAMETER LIST</p>					
End of Comment					
4	(4)	BITSTRING	1	\$FBFLAG1	\$FREEBUF OPTION FLAG 1
		1... ....		\$FB1PROT	"B'10000000" BUFFER TYPE=PROT
		..1. ....		\$FB1HOLD	"B'00100000" BUFFER TYPE=HOLD
5	(5)	BITSTRING	1		Reserved
6	(6)	SIGNED	2	\$FBSTORP	Buffer chain offset
Comment					
<p>Member name --&gt; \$FM            Routine(s) ---&gt; \$FBUFRTN in HASCLINK            Macro(s) -----&gt; \$CALL \$FBUFRTN,INLINE=            \$FBUFRTN'S inline parameter list</p>					
End of Comment					
4	(4)	BITSTRING	1	\$FMFLAG1	\$FBUFRTN option FLAG 1
		1... ....		\$FM1\$ERR	"B'10000000" Issue \$ERROR macro if unfreed buffers remain
		..1. ....		\$FM1CLOS	"B'01000000" Called out of DS CLOSE

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$FBM					
ROUTINE(S) ---> \$FMTBLDM in HASMIPSV					
MACRO(S) -----> \$FMTBLDM					
\$FMTBLDM inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FBMFLG1	Flag byte 1
		1... ....		\$FBM1INT	"B'10000000" Initialize message
		.1.. ....		\$FBM1ADD	"B'01000000" Add text to message
		..1. ....		\$FBM1LAS	"B'00100000" Last (issue message)
		...1 ....		\$FBM1MFL	"B'00010000" Text is list form msg
		.... 1..		\$FBM1TTB	"B'00001000" Truncate trailing blanks
		.... .1..		\$FBM1CNV	"B'00000100" Data conversion or a standard text string if \$FBM1MFL is also set
		.... ..1.		\$FBM1ABB	"B'00000010" Add blank before text/dat
		.... ...1		\$FBM1ABA	"B'00000001" Add blank after text/data
5	(5)	BITSTRING	1	\$FBMFLG2	Flag byte 2
		1... ....		\$FBM2RES	"B'10000000" Reset message
		.1.. ....		\$FBM2MTB	"B'01000000" Process \$FMTMTAB
		..1. ....		\$FBM2WID	"B'00100000" Use wide message width
		...1 ....		\$FBM2IVA	"B'00010000" Internal request via atab
		.... 1..		\$FBM2IFI	"B'00001000" Internal field info req
		.... .1..		\$FBM2INA	"B'00000100" Internal data specified
		.... ..1.		\$FBM2ADJ	"B'00000010" ADJUST= specified
		.... ...1		\$FBM2TTZ	"B'00000001" Truncate trailing zeroes
6	(6)	ADDRESS	1	\$FBMCTYP	Conversion or std text type

Comment

----- Conversion types -----

End of Comment					
6	(6)	X'1'	0	\$FBMCTAD	"1" ADDRESS
6	(6)	X'2'	0	\$FBMCTAS	"2" ASID
6	(6)	X'3'	0	\$FBMCTAC	"3" ASID_COND
6	(6)	X'4'	0	\$FBMCTDS	"4" DSPNAME
6	(6)	X'5'	0	\$FBMCTDC	"5" DSPNAME_COND
6	(6)	X'6'	0	\$FBMCTEP	"6" CHAR
6	(6)	X'7'	0	\$FBMCTHX	"7" HEX
6	(6)	X'8'	0	\$FBMCTRH	"8" HEXRAW
6	(6)	X'9'	0	\$FBMCTKM	"9" KM
6	(6)	X'A'	0	\$FBMCTOF	"10" OFFSET
6	(6)	X'B'	0	\$FBMCTSI	"11" SIGNINT
6	(6)	X'C'	0	\$FBMCTSR	"12" SIGNINTRAW
6	(6)	X'D'	0	\$FBMCTST	"13" STCK
6	(6)	X'E'	0	\$FBMCTSE	"14" STCKE
6	(6)	X'F'	0	\$FBMCTSM	"15" STCKE_MICRO
6	(6)	X'10'	0	\$FBMCTUI	"16" UNSIGNINT
6	(6)	X'11'	0	\$FBMCTUR	"17" UNSIGNINTRAW
6	(6)	X'12'	0	\$FBMCTA8	"18" ADDRESS64

Comment

----- Standard text strings -----

End of Comment					
6	(6)	X'1'	0	\$FBMCTPN	"1" NOTICE msg prefix
6	(6)	X'2'	0	\$FBMCTPW	"2" WARNING msg prefix
6	(6)	X'3'	0	\$FBMCTPE	"3" ERROR msg prefix
7	(7)	BITSTRING	1	\$FBMID	Parameter list ID
		1... ...1		\$FBMIDV	"X'81" Parm list verification val

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
8	(8)	CHARACTER	8	\$FBMSECT	Control section name
16	(10)	CHARACTER	8	\$FBMSEQF	Invoking sequence number
16	(10)	X'14'	0	\$FBMPLEN	"*-PARMSTRT" Length of parms to trace
Comment					
MEMBER NAME --> \$FDI ROUTINE(S) ---> \$FMTDIAL in HASMIPSV MACRO(S) ----> \$FMTDIAL \$FMTDIAL inline parameter list					
End of Comment					
4	(4)	ADDRESS	1	\$FDITYPE	Type indicator byte
4	(4)	X'1'	0	\$FDITCHR	"1" Character data
4	(4)	X'2'	0	\$FDITHEX	"2" Hexadecimal data
4	(4)	X'3'	0	\$FDITUSI	"3" Unsigned integer
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$FDISECT	Control section name
14	(E)	CHARACTER	8	\$FDISEQF	Invoking sequence number
Comment					
MEMBER NAME --> \$FEV ROUTINE(S) ---> \$FMTENV in HASMIPSV MACRO(S) ----> \$FMTENV \$FMTENV inline parameter list					
End of Comment					
4	(4)	ADDRESS	1	\$FEVPVER	\$FMTENV parm list version (Set in \$FMTENV macro expansion and checked in service routine)
5	(5)	ADDRESS	1	\$FEVREQ	Request type
5	(5)	X'1'	0	\$FEVRCU	"1" CREATE request
5	(5)	X'2'	0	\$FEVRDU	"2" DELETE request
5	(5)	X'3'	0	\$FEVRCC	"3" COND_CREATE request
5	(5)	X'4'	0	\$FEVRDC	"4" COND_DELETE request
6	(6)	BITSTRING	1	\$FEVUSE	USE flags - Caution; bit tests use both TM and CLI
		1... ....		\$FEVUIVE	"B'10000000" Used as IPCS VERBEXIT
		.1.. ....		\$FEVUIFR	"B'01000000" Used as IPCS FORMAT rtn
		..1. ....		\$FEVUJDR	"B'00100000" Used as JES2 DISPLAY_RTN
		...1 ....		\$FEVUNUL	"B'00010000" Null USE
7	(7)	BITSTRING	1	\$FEVFLG1	Flag byte 1
		1... ....		\$FEV1CSA	"B'10000000" Issue cond \$FMTSETA
8	(8)	ADDRESS	2	\$FEVWVER	Current \$IPCSWRK version #
10	(A)	CHARACTER	8	\$FEVSECT	Control section name
18	(12)	CHARACTER	8	\$FEVSEQF	Invoking sequence number
18	(12)	X'A'	0	\$FEVMOSQ	"\$FEVSECT,*-\$FEVSECT,C'C" Module/sequence
Comment					
MEMBER NAME --> \$FGA ROUTINE(S) ---> \$FMTGADR in HASMIPSV MACRO(S) ----> \$FMTGADR \$FMTGADR inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FGAFLG1	Flag byte 1
		1... ....		\$FGA1LUP	"B'10000000" Lookup a cb address
		.1.. ....		\$FGA1GNS	"B'01000000" Setup for a cb set
		..1. ....		\$FGA1GTN	"B'00100000" Get next cb in a set
4	(4)	X'E0'	0	\$FGA1STD	"\$FGA1LUP+\$FGA1GNS+\$FGA1GTN" Standard call if any bit on
		...1 ....		\$FGA1EYE	"B'00010000" Verify eye (cont if err)
		.... 1...		\$FGA1ZPM	"B'00001000" Issue msg if zero pointer
		.... .1..		\$FGA1ACM	"B'00000100" Issue msg if access error

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$FGASECT	Control section name
14	(E)	CHARACTER	8	\$FGASEQF	Invoking sequence number
14	(E)	X'12'	0	\$FGAPLEN	** -PARMSTRT" Length of parms to trace

Comment

MEMBER NAME --> \$FGF  
 ROUTINE(S) ---> \$FMTGFLD in HASMIPSV  
 MACRO(S) -----> \$FMTGFLD  
 \$FMTGFLD inline parameter list

End of Comment

4	(4)	CHARACTER	8	\$FGFSECT	Control section name
12	(C)	CHARACTER	8	\$FGFSEQF	Invoking sequence number

Comment

MEMBER NAME --> \$FID  
 ROUTINE(S) ---> FIFODEQ in HASCSRIC  
 MACRO(S) -----> \$FIFODEQ  
 \$FIFODEQ inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$FIDFLG1	Flag byte 1
		1... ....		\$FID1CNT	"B'10000000" COUNT= specified
		.1.. ....		\$FID1ABN	"B'01000000" ABENDERR=YES
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	ADDRESS	2	\$FIDCOFF	Chain field offset

Comment

MEMBER NAME --> \$FIE  
 ROUTINE(S) ---> FIFOENQ in HASCSRIC  
 MACRO(S) -----> \$FIFOENQ  
 \$FIFOENQ inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$FIEFLG1	Flag byte 1
		1... ....		\$FIE1CNT	"B'10000000" COUNT= specified
		.1.. ....		\$FIE1ABN	"B'01000000" ABENDERR=YES
		..1. ....		\$FIE1HEA	"B'00100000" Add element to head
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	ADDRESS	2	\$FIECOFF	Chain field offset

Comment

MEMBER NAME --> \$FIG  
 ROUTINE(S) ---> FIFOGTQ in HASCSRIC  
 MACRO(S) -----> \$FIFOGTQ  
 \$FIFOGTQ inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$FIGFLG1	Flag byte 1
		1... ....		\$FIG1CNT	"B'10000000" COUNT= specified
5	(5)	BITSTRING	1		Reserved for future use

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$FLE ROUTINE(S) ---> \$FMTLERR in HASMIPSV MACRO(S) ----> \$FMTLERR \$FMTLERR inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FLEFLG1	Flag byte 1
		1... ....		\$FLE1RC	"B'10000000" Reason code provided
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$FLESECT	Control section name
14	(E)	CHARACTER	8	\$FLESEQF	Invoking sequence number
22	(16)	ADDRESS	4	\$FLEAPAR	Addr of APARNUM symbol
Comment					
MEMBER NAME --> \$FMS ROUTINE(S) ---> \$FMTMSG in HASMIPSV MACRO(S) ----> \$FMTMSG \$FMTMSG inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FMSFLG1	Flag byte 1
		1... ....		\$FMS1WID	"B'10000000" Wide message width
		.1.. ....		\$FMS1BLN	"B'01000000" Display blank line
		..1. ....		\$FMS1CBL	"B'00100000" Conditional blank line
		...1 ....		\$FMS1IND	"B'00010000" Indentation requested
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$FMSSECT	Control section name
14	(E)	CHARACTER	8	\$FMSSEQF	Invoking sequence number
14	(E)	X'12'	0	\$FMSPLEN	"*-PARAMSTR" Length of parms to trace
Comment					
MEMBER NAME --> \$FPR ROUTINE(S) ---> \$FMTPROC in HASMIPSV MACRO(S) ----> \$FMTPROC \$FMTPROC inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FPRFLG1	Flag byte 1
		1... ....		\$FPR1MLT	"B'10000000" Process MULTIPLE FMTCTABs
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$FPRSECT	Control section name
14	(E)	CHARACTER	8	\$FPRSEQF	Invoking sequence number
Comment					
MEMBER NAME --> \$FRE ROUTINE(S) ---> \$FREJLOK in HASPJQS MACRO(S) ----> \$FRELOK \$FRELOK inline parameter list for TYPE=JOB					
End of Comment					
4	(4)	BITSTRING	1	\$FREFLG1	\$FRELOK option flag
		1... ....		\$FRE1NW	"B'10000000" Cannot \$WAIT
		.1.. ....		\$FRE1NTR	"B'01000000" Do not take trace
		..1. ....		\$FRE1JQA	"B'00100000" Free JQA



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$FSA ROUTINE(S) ---> \$FMTSETA in HASMIPSV MACRO(S) ----> \$FMTSETA \$FMTSETA inline parameter list					
End of Comment					
4	(4)	CHARACTER	8	\$FSASECT	Control section name
12	(C)	CHARACTER	8	\$FSASEQF	Invoking sequence number
12	(C)	X'4'	0	\$FSAMOSQ	"\$FSASECT,*-\$FSASECT,C'C" Module/sequence
Comment					
MEMBER NAME --> \$FST ROUTINE(S) ---> \$FMTSTOR in HASMIPSV MACRO(S) ----> \$FMTSTOR \$FMTSTOR inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FSTFLG1	Flag byte 1
4	(4)	X'1'	0	\$FST1GTU	"1" Get storage unconditional
4	(4)	X'2'	0	\$FST1GTC	"2" Get storage conditional
4	(4)	X'3'	0	\$FST1FRE	"3" Free storage
5	(5)	BITSTRING	1	\$FSTFLG2	Flag byte 2
		1... ....		\$FST2FAD	"B'10000000" ADDR specified on free
		.1.. ....		\$FST2CTS	"B'01000000" *CTAB_WORK
		..1. ....		\$FST2CTM	"B'00100000" *CTAB_WORK_LEVEL
		...1 ....		\$FST2PTS	"B'00010000" *PARENT_CTAB_WORK
		.... 1...		\$FST2PTM	"B'00001000" *PARENT_CTAB_WORK_LEVEL
		.111 1...		\$FST2TAB	"B'01111000" Ctab related request
6	(6)	CHARACTER	8	\$FSTSECT	Control section name
14	(E)	CHARACTER	8	\$FSTSEQF	Invoking sequence number
Comment					
MEMBER NAME --> \$FTB ROUTINE(S) ---> \$FRETBUF IN HASCNJAS MACRO(S) ----> \$FRETBUF \$FRETBUF INLINE PARAMETER LIST					
End of Comment					
4	(4)	BITSTRING	1	\$FTBFLG1	\$FRETBUF option flag
		1... ....		\$FTB1CHN	"B'10000000" CHAIN=YES specified
Comment					
Member name --> \$FTR Routine(s) ---> \$FRETRE in HASCLINK MACRO(s) ----> \$FRETRE \$FRETRE Inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$FTRFLG1	\$FRETRE option flag
		1... ....		\$FTR1IOW	"B'10000000" WAITIO=YES specified

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$FUC ROUTINE(S) ---> \$FREUCBS IN HASPNUC MACRO(S) ----> \$FREUCBS \$FREUCBS' INLINE PARAMETER LIST					
End of Comment					
4	(4)	BITSTRING 1... ..	1	\$FUCFLG1 \$FUC1UNP	\$FREUCBS OPTION FLAG "B'10000000" UNPIN=YES specified
Comment					
MEMBER NAME --> \$GTA ROUTINE(S) ---> \$GETABLE in HASPTABS MACRO(S) ----> \$GETABLE \$GETABLE routine inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$GTATYPE	Table type (See \$MITETBL for valid types)
5	(5)	BITSTRING 1... ..	1	\$GTAFLG1 \$GTAH1ST	Flag byte "B'10000000" Run HASP tables first
Comment					
MEMBER NAME --> \$GTB ROUTINE(S) ---> \$GETBUFR IN HASPNUC MACRO(S) ----> \$GETBUF \$GETBUF'S INLINE PARAMETER LIST					
End of Comment					
4	(4)	BITSTRING	1	\$GTBFLG1	\$GETBUF OPTION FLAG
Comment					
B'10000000' \$GBUFWT used in \$HASPEQU					
End of Comment					
		1... ..		\$GTB1WAT	"B'10000000" Wait requested
		.1. ....		\$GTB1FIX	"B'01000000" Wait requested
		..1. ....		\$GTB1MUL	"B'00100000" Multiple buffers requested
		...1 ....		\$GTB1LOW	"B'00010000" GET STORAGE BELOW 16MB
5	(5)	BITSTRING	1	\$GTBFTYP	Buffer type flag
Comment					
MEMBER NAME --> \$GTTB ROUTINE(S) ---> \$GETTBUF IN HASCNJAS MACRO(S) ----> \$GETTBUF \$GETTBUF INLINE PARAMETER LIST					
End of Comment					
4	(4)	BITSTRING	1	\$GTTBUSE	USE= (See TBFTYPE for values in byte)
Comment					
MEMBER NAME --> GCMB ROUTINE(S) ---> \$GETCMBR MACRO(S) ----> \$GETCMB \$GETCMB's inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$GTCFLG1	\$GETCMB option flag byte

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		1... ....		\$GTC1WAT	"B'10000000" WAIT=YES requested
		.1... ....		\$GTC1DMC	"B'01000000" DEMANDCMB=YES specified
5	(5)	BITSTRING	1		Reserved for future use
Comment					
<p>MEMBER NAME --&gt; \$GNH            ROUTINE(S) ---&gt; \$GETNHB            MACRO(S) ----&gt; \$GETNHB            \$GETNHB's inline parameter list</p>					
End of Comment					
4	(4)	BITSTRING	1	\$GNHFLG1	\$GETNHB option flag byte
		1... ....		\$GNH1WAT	"B'10000000" WAIT=YES requested
5	(5)	BITSTRING	1		Reserved for future use
Comment					
<p>MEMBER NAME --&gt; \$GUC            ROUTINE(S) ---&gt; \$GETUCBS IN HASPNUC            MACRO(S) ----&gt; \$GETUCBS            \$GETUCBS' INLINE PARAMETER LIST</p>					
End of Comment					
4	(4)	BITSTRING	1	\$GUCFLG1	\$GETUCBS OPTION FLAG
		1... ....		\$GUC1CNT	"B'10000000" CONT=YES specified
		.1... ....		\$GUC1UNT	"B'01000000" UNIT= specified
Comment					
<p>MEMBER NAME --&gt; \$GC            ROUTINE(S) ---&gt; \$GETCEL IN HASLINK            MACRO(S) ----&gt; \$GETCEL            JES2 CSA CELL POOL GET ROUTINE (\$GETCEL) PARAMETER LIST            THIS PARM LIST VARIES IN LENGTH. IF THE SIZE= PARAMETER            ON THE \$GETCEL MACRO IS SPECIFIED IN REGISTER NOTATION,            THEN REGISTER 2 IS LOADED WITH THE SIZE. OTHERWISE, THE            THE SIZE OF THE CSA CELL TO OBTAIN IS PLACED AT THE END            OF THE PARAMETER LIST.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$GCFLAG1	FLAG BYTE FOR \$GETCEL
		1... ....		\$GC1LPRM	"B'10000000" LONG FORM OF VARIABLE PARM LIST
5	(5)	BITSTRING	1	\$GCRSVRD	RESERVED FOR FUTURE USE
Comment					
<p>VARIABLE PORTION OF THE \$GETCEL PARAMETER LIST.</p>					
End of Comment					
6	(6)	ADDRESS	2	\$GCSIZE	SIZE OF CSA CELL REQUESTED
Comment					
<p>MEMBER NAME --&gt; \$GF            ROUTINE(S) ---&gt; \$GFMAIN IN HASPNUC, \$HGFMAIN IN HASLINK            MACRO(S) ----&gt; \$GETMAIN            BRANCH ENTRY GETMAIN/FREEMAIN SERVICES INLINE PARM LIST.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$GFFLG3	\$GETMAIN/\$FREEMAIN flag 3
		1... ....		\$GF3LVR0	"B'10000000" Indicate LV passed in R0
		.1... ....		\$GF3BUFR	"B'01000000" Indicate buffer get/free
		..1. ....		\$GF3HTCB	"B'00100000" Indicate TCB=HIGH

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		...1 ....		\$GF3FREE	"B'00010000" Indicate free main/buf
		.... 1...		\$GF3TCBY	"B'00001000" Indicate TCB=YES, TCB in R1
		.... .1..		\$GF3JTCB	"B'00000100" Indicate JOB STEP TCB
		.... ..1.		\$GF3TCBK	"B'00000010" Indicate TCBPKF used as key
		.... ...1		\$GF3PSWK	"B'00000001" Indicate PSW used as key
5	(5)	BITSTRING	1	\$GFFLG4	\$GETMAIN/\$FREEMAIN flag 4
		1... ....		\$GF4SPR0	"B'10000000" Subpool passed in R0
		.1.. ....		\$GF4STOR	"B'01000000" KEY=STORAGE for UBUFs only
		..1. ....		\$GF4ZERO	"B'00100000" ZEROSTOR=YES specified
		...1 ....		\$GF4NOLV	"B'00010000" Subpool freemain (no LV=)
		.... 1...		\$GF4BAKR	"B'00001000" REGS=SYSTEM was specified
		.... .1..		\$GF4OAUZ	"B'00000100" OWNER=AUX was specified
		.... ..1.		\$GF4TCUR	"B'00000010" TCB=CURRENT was specified
8	(8)	SIGNED	4	\$GFLENV	\$GETMAIN/\$FREEMAIN length
Comment					
<p>-----</p> <p>\$GFFLG1 through \$GFFLG2 are passed to the service in R15 and not in \$PARMLST. These flags must match the register 3 value passed to branch entry GETMAIN/FREEMAIN</p> <p>-----</p>					
End of Comment					
12	(C)	BITSTRING	1	\$GFFLG1	\$GETMAIN/\$FREEMAIN flag 1
Comment					
<p>EQU B'10000000' Reserved</p> <p>EQU B'01000000' Reserved</p>					
End of Comment					
		..1. ....		\$GF1AR15	"B'00100000" AR 15 is in use
		...1 ....		\$GF1RS64	"B'00010000" Indicate LOC=(,64)
		.... 1...		\$GF1CHK0	"B'00001000" Indicate CHECKZERO=YES
Comment					
<p>EQU B'00000100' Reserved</p>					
End of Comment					
		.... ....		\$GF1OHOM	"B'00000000" Indicate OWNER=HOME
		.... ...1		\$GF1OPRI	"B'00000001" Indicate OWNER=PRIMARY
		.... ..1.		\$GF1OSEC	"B'00000010" Indicate OWNER=SECONDARY
		.... ...11		\$GF1OSYS	"B'00000011" Indicate OWNER=SYSTEM
13	(D)	BITSTRING	1	\$GFKEY	KEY STORAGE REQUESTED IN
14	(E)	BITSTRING	1	\$GFSUBPL	SUBPOOL STORAGE REQUESTED IN
15	(F)	BITSTRING	1	\$GFFLG2	\$GETMAIN/\$FREEMAIN flag 2
Comment					
<p>EQU B'10000000' Reserved</p>					
End of Comment					
		.1.. ....		\$GF2RS31	"B'01000000" Indicate LOC=(,31)
		..1. ....		\$GF2LC31	"B'00100000" Indicate LOC=31
		...1 ....		\$GF2LC24	"B'00010000" Indicate LOC=24
Comment					
<p>EQU B'00001000' Ind variable request</p>					
End of Comment					

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... .1..		\$GF2PGB	"B'00000100" INDICATE BNDRY=PAGE
		.... ..1.		\$GF2UNCD	"B'00000010" INDICATE UNCONDITIONAL REQUEST
		.... ...1		\$GF2FMN	"B'00000001" INDICATE FREEMAIN

Comment

MEMBER NAME --> \$GLW  
 ROUTINE(S) ---> \$GETLOKW  
 MACRO(S) ----> \$GETLOKW  
 \$GETLOKW'S INLINE PARAMETER LIST

End of Comment

4	(4)	BITSTRING	1	\$GLWFLG1	\$GETLOKW FLAG 1
		1... ....		\$GLW1WT	"B'10000000" WAIT=YES

Comment

MEMBER NAME --> \$IBL  
 ROUTINE(S) ---> \$IOTBLD IN HASCSRDS  
 MACRO(S) ----> \$IOTBLD  
 \$IOTBLD'S INLINE PARAMETER LIST

End of Comment

4	(4)	BITSTRING	1	\$IBFLAG1	\$IOTBLD OPTION FLAG 1
		1... ....		\$IB1DPDB	"B'10000000" INDICATE TYPE=Pddb
		.1. ....		\$IB1DSPN	"B'01000000" INDICATE TYPE=SPIN
		..1. ....		\$IB1D2ND	"B'00100000" INDICATE TYPE=SECOND
		...1 ....		\$IB1DPRI	"B'00010000" Indicate TYPE=PRIMARY
		.... 1..		\$IB1DAUG	"B'00001000" Indicate SPIN=DAUGHTER
		.... ..1.		\$IB1DSJI	"B'00000100" Indicate SJJOB provided
		.... ...1.		\$IB1DNCH	"B'00000010" Indicate skip chaining of new IOT of TYPE=Pddb
5	(5)	BITSTRING	1		RESERVED FOR FUTURE USE

Comment

MEMBER NAME --> \$ICL  
 ROUTINE(S) ---> \$IOTCNT IN HASPNUC  
 MACRO(S) ----> \$IOTCNT  
 \$IOTCNT'S INLINE PARAMETER LIST

End of Comment

4	(4)	BITSTRING	1	\$ICFLAG1	\$IOTBLD OPTION FLAG 1
		1... ....		\$IC1LKNO	"B'10000000" INDICATE LOCK=NO
		.1. ....		\$IC1LOCK	"B'01000000" INDICATE LOCK=YES
		..1. ....		\$IC1IOT	"B'00100000" IN STORAGE IOT ADDRESS IS SUPPLIED
		...1 ....		\$IC1JOE	"B'00010000" R0 contains addr of JOE
5	(5)	BITSTRING	1		RESERVED FOR FUTURE USE

Comment

MEMBER NAME --> \$JCN  
 ROUTINE(S) ---> \$JCANR IN HASPCOMM  
 MACRO(S) ----> \$JCAN  
 \$JCAN'S INLINE PARAMETER LIST

End of Comment

4	(4)	BITSTRING	1	\$JCNFLG1	\$JCAN Flag
		1... ....		\$JCN1PRO	"B'10000000" Output Is Protected
		.1. ....		\$JCN1TST	"B'01000000" ACTION=TEST
		..1. ....		\$JCN1TSU	"B'00100000" TSU=NO

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> JOBCLASV ROUTINE(S) ---> JOBCLASV in HASCSTRIP MACRO(S) -----> None JOBCLASV INLINE PARAMETER LIST					
End of Comment					
4	(4)	BITSTRING 1... .... .1... ....	1	\$JOBCLAV \$JCLSRM \$JCLSNS	JOBCLASV flag "B'10000000" Return SAF messages "B'01000000" Do not check submitter RESERVED FOR FUTURE USE
5	(5)	BITSTRING	1		
Comment					
MEMBER NAME --> \$#ADD ROUTINE(S) ---> \$#ADD in HASPJOS MACRO(S) -----> \$#ADD \$#ADD caller requested JOA address be returned verses a real work JOE address,					
End of Comment					
4	(4)	BITSTRING 1... ....	1	#\$ADDFLG #\$ADDJOA	"B'10000000" Caller of \$#ADD requested an update mode JOA be returned verses a real work JOE address. This JOA/JOE was added to the JOT.
Comment					
MEMBER NAME --> \$#DISPRO ROUTINE(S) ---> \$#DISPRO in HASPJOS MACRO(S) -----> \$#DISPRO \$#DISPRO routine inline parameter list.					
End of Comment					
4	(4)	BITSTRING 1... ....	1	#\$DSPFLG #\$DSFJOA	#\$DISPRO parameter flag "B'10000000" If caller supplied JOA then free it. Default is to free.
Comment					
MEMBER NAME --> \$#JWEL ROUTINE(S) ---> \$#JWEL in HASPJOS MACRO(S) -----> \$#JWEL \$#JWEL routine inline parameter list.					
End of Comment					
4	(4)	BITSTRING 1... .... .1... .... ..1. .... ...1 .... .... 1... .... .1.. .... .1.. .... ...1	1	#\$JWLFLG #\$JLONG #\$JSERCH #\$JPURGE #\$JADD #\$JFORCE #\$JCOND #\$JALL #\$JANY	\$JWEL parameter flag "B'10000000" Long form of device number "B'01000000" Search for existing JWEL "B'00100000" Purge all JWELs for JOE "B'00010000" Add a JWEL "B'00001000" Force purge of JWEL chain "B'00000100" ADD or PURGE conditional "B'00000010" Made SEARCH match all JWELs "B'00000001" Made SEARCH find any JWEL
5	(5)	BITSTRING 1... .... .1... .... ..1. .... ...1 ....	1	\$JWLFL2 \$JDETCH \$JATTCH \$JINIT \$JANCHR	Second \$JWEL parameter flg "B'10000000" Detach JWEL chain "B'01000000" Attach JWEL chain "B'00100000" INIT JWEL anchor "B'00010000" Determine JWEL anchor addr

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$#PUT            ROUTINE(S) ---&gt; \$#PUT in HASPJOS            MACRO(S) ----&gt; \$#PUT            \$#PUT routine inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$#PUTFLG	\$#PUT parameter flag
		...1 ....		\$#PJWEL	"B'00010000" Purge SAPI JWELs
		.... 1...		\$#PFRJOA	"B'00001000" If caller supplied JOA then free it. Default is to free.
Comment					
<p>MEMBER NAME --&gt; \$LG            ROUTINE(S) ---&gt; \$LOGMSG IN HASPSSRV            MACRO(S) ----&gt; \$LOGMSG            PLACING JOB RELATED MESSAGES INTO A JOB'S JOBLOG OR SYSMMSG            DATA SET. NOTE PARAMETER LIST VARIES IN LENGTH IN ORDER            TO PHYSICALLY CONTAIN THE REQUESTOR'S EBCDIC NAME. THE            LENGTH OF THE NAME IS IN FIELD \$LG1TXTL.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$LGSUBP	SUBPOOL TO FREEMAIN MSGAREA
5	(5)	BITSTRING	1	\$LGFLAG1	FLAG BYTE
		1... ....		\$LG1MFRE	"B'10000000" MSGFREE=YES WAS SPECIFIED
		.1.. ....		\$LG1WTO	"B'01000000" WTO=YES WAS SPECIFIED
6	(6)	BITSTRING	1	\$LG1TXTL	MACHINE LENGTH (LENGTH-1) OF REQUESTOR NAME
7	(7)	CHARACTER	1	\$LGRQSTR	START OF REQUESTOR NAME
Comment					
<p>MEMBER NAME --&gt; \$LV            ROUTINE(S) ---&gt; \$TQLEVEL in HASPXCF            MACRO(S) ----&gt; \$LEVEL</p>					
End of Comment					
4	(4)	BITSTRING	1	\$LVFLAG1	Flag byte 1
		1... ....		\$LV1QSE	"B'10000000" QSE fields provided
		.1.. ....		\$LV1WAIT	"B'01000000" Wait for homogeneity
5	(5)	BITSTRING	1	\$LVQFLAG	QSE flag value to test
6	(6)	ADDRESS	2	\$LVQOFF	Offset of QSE flag
Comment					
<p>MEMBER NAME --&gt; \$BTM            ROUTINE(S) ---&gt; \$BITMAP routine in HASCSRIC            MACRO(S) ----&gt; \$BITMAP            \$BITMAP inline parameter list.</p>					
End of Comment					
4	(4)	SIGNED	1	\$BTMACT	ACTION= requested
4	(4)	X'0'	0	\$BTMCRT	"0" ACTION=CREATE
4	(4)	X'4'	0	\$BTMDSTP	"4" ACTION=DESTROY_PRIVATE
4	(4)	X'8'	0	\$BTMSET	"8" ACTION=SET
4	(4)	X'C'	0	\$BTMRST	"12" ACTION=RESET
4	(4)	X'10'	0	\$BTMTST	"16" ACTION=TEST
4	(4)	X'14'	0	\$BTMSON	"20" ACTION=SCANON
4	(4)	X'18'	0	\$BTMSOFF	"24" ACTION=SCANOFF
4	(4)	X'1C'	0	\$BTMCLR	"28" ACTION=CLEAR
4	(4)	X'20'	0	\$BTMSETR	"32" ACTION=SETRANGE
4	(4)	X'24'	0	\$BTMTSRO	"36" ACTION=TESTRANGE_ON

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
4	(4)	X'28'	0	\$BTMTRSF	"40" ACTION=TESTRANGE_OFF
4	(4)	X'2C'	0	\$BTMDSTC	"44" ACTION=DESTROY_COMMON
5	(5)	BITSTRING	1	\$BTMFLG	\$BITMAP option flags
		1... ....		\$BTMCRTC	"B'10000000" For ACTION=CREATE: ON = (CREATE,COMMON) OFF = (CREATE,PRIVATE)
		.1.. ....		\$BTMCRON	"B'01000000" For ACTION=CREATE: ON = (CREATE,...,INITON) OFF = (CREATE,...,INITOFF)
		...1. ....		\$BTMNOAT	"B'00100000" ATOMIC=NO was specified.
		...1 ....		\$BTMERCDC	"B'00010000" BOUNDARY_ABEND=CODE/ABEND ON = Return error code for boundary errors OFF = \$ERROR on boundary errors.

Comment

MEMBER NAME --> \$JSMTSRV  
 ROUTINE(S) ---> JSMTSRV in HASCSRDS  
 MACRO(S) -----> \$JSMTSRV

End of Comment

4	(4)	BITSTRING	1	\$JSMACTN	Action:
4	(4)	X'1'	0	\$JSMACTA	"1" ACTION=ADD
4	(4)	X'2'	0	\$JSMACTX	"2" ACTION=EXTRACT
5	(5)	BITSTRING	1	\$JSMTBLF	Symbol table format:
5	(5)	X'1'	0	\$JSMTBJS	"1" SYMTBLF=JESSYM
5	(5)	X'2'	0	\$JSMTBCL	"2" SYMTBLF=JCLSYM
5	(5)	X'3'	0	\$JSMTBSS	"3" SYMTBLF=SYSSYM
6	(6)	BITSTRING	1	\$JSMSYMT	Symbol type:
6	(6)	X'1'	0	\$JSMSYCV	"1" SYMTYPE=CNV
6	(6)	X'2'	0	\$JSMSYIN	"2" SYMTYPE=INT
6	(6)	X'3'	0	\$JSMSYAS	"3" SYMTYPE=ALTSYS
7	(7)	BITSTRING	1	\$JSMFLGS	JSMTSRV options:
		1... ....		\$JSMJQE	"B'10000000" R0 points to JQE
		.1.. ....		\$JSMSJB	"B'01000000" R0 points to SJB

Comment

MEMBER NAME --> \$MODCHK  
 ROUTINE(S) ---> \$MODCHK in HASPCSV  
 MACRO(S) -----> \$MODCHK

End of Comment

4	(4)	BITSTRING	1	\$MCFLAG1	Flag byte 1
		1... ....		\$MCMMSG	"B'10000000" MESSAGE=YES specified
		.1.. ....		\$MCMMSG	"B'01000000" MESSAGE=SUPPRESS specified
5	(5)	BITSTRING	1		Reserved
5	(5)	X'2'	0	\$MCBYTES	"L'\$MCTESTS" Number of bytes for test flags \$MCBYTES*8 must be larger than or equal to \$MCNTEST
6	(6)	BITSTRING	2	\$MCTESTS	Test requested byte
6	(6)	BITSTRING	0	\$MCRMD24	"B'1000000000000000" Module below 16meg line
6	(6)	BITSTRING	0	\$MCCOMMN	"B'0100000000000000" Module in common storage
6	(6)	BITSTRING	0	\$MCMIT	"B'0010000000000000" Module large enough for MIT, MIT id valid, MITETBL in module
6	(6)	BITSTRING	0	\$MCVERS	"B'0001000000000000" HCT version=version in MIT
6	(6)	BITSTRING	0	\$MCNAME	"B'0000100000000000" Module name = name in MIT
6	(6)	BITSTRING	0	\$MCPROPX	"B'0000010000000000" Propagate \$EXIT points to XIT table of defined exits
6	(6)	BITSTRING	0	\$MCRSLVX	"B'0000001000000000" Resolve exit routine addresses to XRT
6	(6)	BITSTRING	0	\$MCTABL	"B'0000000100000000" Process dynamic tables
		1... ....		\$MCADDR	"B'0000000010000000" Resolve routine addresses
		.1.. ....		\$MCDYNAM	"B'0000000001000000" Supports dynamic commands
		...1. ....		\$MCDELET	"B'0000000000100000" Can the module be deleted
6	(6)	X'B'	0	\$MCNTEST	"11" Number of tests now defined



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$MSTNTFY            ROUTINE(S) ---&gt; \$MSTNTFY in HASPSSRV                      XCF PCE in HASPXCF            MACRO(S) -----&gt; \$MSTNTFY</p>					
End of Comment					
4	(4)	BITSTRING	1	\$MSTTYPE	Flag byte 1
4	(4)	X'1'	0	\$MSTSET	"1" TYPE=SET specified
4	(4)	X'2'	0	\$MSTCLR	"2" TYPE=CLEAR specified
5	(5)	BITSTRING	1	\$MSTFLG1	Flag byte 2
		1... ..		\$MST1ECB	"B'10000000" ECB was supplied
		.1.. ..		\$MST1EXT	"B'01000000" EXITPGM was supplied
		..1. ....		\$MST1EXP	"B'00100000" EXITPRM was supplied
Comment					
<p>MEMBER NAME --&gt; \$NATADD            ROUTINE(S) ---&gt; \$NATADD in HASPNATS            MACRO(S) -----&gt; \$NATADD            Nodes Attached Table ADD routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NADSTAT	NAT queue to \$NATADD element to (see NATCSTAT)
5	(5)	BITSTRING	1	\$NADFLG1	\$NATADD parameter flag
		1... ..		\$NAD1NAT	"B'10000000" Use prototype NAT element
		.1.. ..		\$NAD1NCC	"B'01000000" Use prototype NCC record
		..1. ....		\$NAD1STA	"B'00100000" Add static connection
		...1 ....		\$NAD1CES	"B'00010000" Bypass CES TIMETOL check
Comment					
<p>MEMBER NAME --&gt; \$NATGET            ROUTINE(S) ---&gt; \$NATGET in HASPNATS            MACRO(S) -----&gt; \$NATGET            Nodes Attached Table GET routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NGTSTAT	NAT queue to \$NATGET element from (see NATCSTAT)
5	(5)	BITSTRING	1	\$NGTFLG1	\$NATGET parameter flag
		1... ..		\$NGT1NAT	"B'10000000" Use prototype NAT element
		.1.. ..		\$NGT1NCC	"B'01000000" Use prototype NCC record
		..1. ....		\$NGT1TOK	"B'00100000" Token provided to routine
		...1 ....		\$NGT1FST	"B'00010000" Use Fast Path \$NATGET
Comment					
<p>MEMBER NAME --&gt; \$NATMOD            ROUTINE(S) ---&gt; \$NATMOD in HASPNATS            MACRO(S) -----&gt; \$NATMOD            Nodes Attached Table MODify routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NMDSTAT	NAT queue to \$NATMOD element to (see NATCSTAT)
5	(5)	BITSTRING	1	\$NMDFLG1	\$NATMOD parameter flag 1
		1... ..		\$NMD1NAT	"B'10000000" Use prototype NAT element
		.1.. ..		\$NMD1NCC	"B'01000000" Use prototype NCC record
		..1. ....		\$NMD1FST	"B'00100000" Use Fast Path \$NATMOD
		...1 ....		\$NMD1STA	"B'00010000" Modify STATIC connection
		.... 1...		\$NMD1CES	"B'00001000" Bypass CES TIMETOL check

## \$PARMLST Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$NATNOT            ROUTINE(S) ---&gt; \$NATNOT in HASPNATS            MACRO(S) ----&gt; \$NATNOT            Nodes Attached Table NOTify routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NNTFLG1	\$NATNOT parameter flag 1
		1... ....		\$NNT1SET	"B'10000000" TYPE=SET or TESTSET
		.1.. ....		\$NNT1TST	"B'01000000" TYPE=TEST or TESTSET
		..1. ....		\$NNT1NOT	"B'00100000" NOTIFIED=YES
		...1 ....		\$NNT1FST	"B'00010000" PATH=FAST
		.... 1..		\$NNT1MTR	"B'00001000" Update MASTER notify map
		.... .1..		\$NNT1MMA	"B'00000100" SCOPE=MAS was specified
Comment					
<p>MEMBER NAME --&gt; \$NATREM            ROUTINE(S) ---&gt; \$NATREM in HASPNATS            MACRO(S) ----&gt; \$NATREM            Nodes Attached Table REMove routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NRMFLG1	\$NATREM parameter flag 1
		1... ....		\$NRM1STA	"B'10000000" Remove static NAT
		.1.. ....		\$NRM1ALL	"B'01000000" Remove all NATs
Comment					
<p>MEMBER NAME --&gt; \$NHR            ROUTINE(S) ---&gt; NJEHDRCV in HASPNET            MACRO(S) ----&gt; \$NHDRCV            Network Header Receive routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NHRFLG1	\$NHDRCV parameter flag 1
		1... ....		\$NHR1XIT	"B'10000000" Invoke exit 47 after recv
5	(5)	BITSTRING	1	\$NHRSRCB	SRCB of received header
Comment					
<p>MEMBER NAME --&gt; \$NHW            ROUTINE(S) ---&gt; NJEHDRWR in HASPNET            MACRO(S) ----&gt; \$NHDWRT            Network Header Write routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NHWFLG1	\$NHDWRT parameter flag 1
		1... ....		\$NHW1FRE	"B'10000000" Free header after write
		.1.. ....		\$NHW1WAT	"B'01000000" Wait for write to complete
Comment					
<p>MEMBER NAME --&gt; \$NHX            ROUTINE(S) ---&gt; NJEHDXTM in HASPNET            MACRO(S) ----&gt; \$NHDXMT            Network Header Transmit routine's inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$NHXFLG1	\$NHDXMT parameter flag 1
		1... ....		\$NHX1FRE	"B'10000000" Free header after xmit

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
5	(5)	.1.. .... BITSTRING	1	\$NHX1XIT \$NHXSRCB	"B'01000000" Invoke exit 46 before xmit SRCB of xmited header
Comment					
<p>MEMBER NAME --&gt; \$NSL            ROUTINE(S) ---&gt; \$NSSTLOK in HASCNJAS            MACRO(S) ----&gt; \$NSSTLOK            Synchronize local and checkpointed NITs.</p>					
End of Comment					
4	(4)	.1.. .... BITSTRING	1	\$NSLFLG1 \$NSL1OBT \$NSL1WAT \$NSL1SHR	\$NSSTLOK parameter flag 1 "B'10000000" REQUEST=OBTAIN "B'01000000" WAIT=YES "B'00100000" SHARED=YES
Comment					
<p>MEMBER NAME --&gt; \$NSY            ROUTINE(S) ---&gt; \$NITSYNC in HASPNET            MACRO(S) ----&gt; \$NITSYNC            Synchronize local and checkpointed NITs.</p>					
End of Comment					
4	(4)	.1.. .... BITSTRING	1	\$NSYFLG1 \$NSY1REF \$NSY1CMP \$NSY1LOC \$NSY1CKP \$NSY1MSG \$NSY1NIT	\$NITSYNC parameter flag 1 "B'10000000" ACTION=(REFRESH,) "B'01000000" ACTION=(COMPARE,) "B'00100000" ACTION=(,LOCAL) "B'00010000" ACTION=(,CKPT) "B'00001000" MESSAGE=YES "B'00000100" Update single NIT
Comment					
<p>MEMBER NAME --&gt; \$NOT            ROUTINE(S) ---&gt; \$NOTIFY in HASCSIRQ            MACRO(S) ----&gt; \$NOTIFY            Send a NJE notify message</p>					
End of Comment					
4	(4)	.1.. .... BITSTRING	1	\$NOTFLG1 \$NOT1NJB \$NOT1NUM \$NOT1JQE \$NOT1NUS \$NOT1MEM	\$NOTIFY parameter flag 1 "B'10000000" JOB=NO was specified "B'01000000" NODENUM was specified "B'00100000" JQE was specified "B'00010000" USERID=NONE "B'00001000" MEMBER= was specified
Comment					
<p>MEMBER NAME --&gt; \$PBL            ROUTINE(S) ---&gt; \$PDBBLD IN HASCDSAL            MACRO(S) ----&gt; \$PDBBLD            BUID AND INITIALIZE A PDDB ROUTINE'S INLINE PARAMETER LIST.</p>					
End of Comment					
4	(4)	.1.. .... BITSTRING	1	\$PBLFLAG1 \$PB1DPDB \$PB1DSPN \$PB1DAUG	\$PDBBLD MACRO OPTION FLAGS FOR USER ENVIRONMENT "B'10000000" INDICATE TYPE=PDDB "B'01000000" INDICATE TYPE=SPIN "B'00100000" Indicate SPIN=DAUGHTER
5	(5)	..... BITSTRING	1		RESERVED FOR FUTURE USE

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$PRG ROUTINE(S) ---> \$PURGER IN HASPTRAK MACRO(S) ----> \$PURGE PURGER routine inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$PRGFLG1	Flag byte
		1... ....		\$PRG1Vfy	"B'10000000" SAF verification required
		.1.. ....		\$PRG1ENF	"B'01000000" Issue PURGE ENF
		..1. ....		\$PRG1JOA	"B'00100000" JOA address provided
Comment					
MEMBER NAME --> \$PSQ ROUTINE(S) ---> XPOSTXEQ in HASPXEQ MACRO(S) ----> \$POSTXEQ XPOSTXEQ routine inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$PSQFLG1	Flag byte
		1... ....		\$PSQ1MAS	"B'10000000" Wake up all members of MAS
		.1.. ....		\$PSQ1JQE	"B'01000000" Caller provided JQE in R1
Comment					
MEMBER NAME --> \$PTA ROUTINE(S) ---> \$PUTABLE in HASPTABS MACRO(S) ----> \$PUTABLE \$PUTABLE routine inline parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$PTAFLG1	Flag byte
		1... ....		\$PTA1MCT	"B'10000000" Offset of \$PAIR is in MCT
		.1.. ....		\$PTA1UCT	"B'01000000" Offset of \$PAIR is in UCT
		..1. ....		\$PTA1CCT	"B'00100000" Offset of \$PAIR is in HCCT
Comment					
MEMBER NAME --> \$QB ROUTINE(S) ---> \$QBUSY in HASPJQS MACRO(S) ----> \$QBUSY \$QBUSY routines inline parameter list.					
End of Comment					
4	(4)	BITSTRING	1	\$QBSYFLG	\$QBUSY parameter flag
		1... ....		\$QBACTON	"B'10000000" Set the busy bits for this JQE on
		.1.. ....		\$QBACTOF	"B'01000000" Set the busy bits for this JQE off
		..1. ....		\$QBTRACE	"B'00100000" Trace this call
		...1 ....		\$QBBREAL	"B'00010000" Real JQE was passed
		.... 1..		\$QBBNALCT	"B'00001000" Don't alter xeq class count
		.... .1..		\$QBHV CAT	"B'00000100" CAT passed in by caller
		.... ..1.		\$QBDODEV	"B'00000010" Set JQEDEV from PCEDCT
		.... ...1		\$QBDEVID	"B'00000001" Set JQEDEV from passed field

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$#B                      ROUTINE(S) ---&gt; \$#BUSY in HASPJOS                      MACRO(S) ----&gt; \$#BUSY                      \$#BUSY routines inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$#BSYFLG	\$#BUSY parameter flag
		1... ....		\$#BACTON	"B'10000000" Set the busy bits for this JOE on
		.1.. ....		\$#BACTOF	"B'01000000" Set the busy bits for this JOE off
		..1. ....		\$#BTRACE	"B'00100000" Trace this call
		...1 ....		\$#BREAL	"B'00010000" Real JOE was passed
		.... 1..		\$#BPOST	"B'00001000" \$#POST the JOE
		.... .1..		\$#BDODEV	"B'00000100" Set JOEDEVID from PCEDCT
		.... ..1.		\$#BDEVID	"B'00000010" Set JOEDEVID from passed field
		.... ...1		\$#BCKPNO	"B'00000001" No Checkpoint of JOE
5	(5)	BITSTRING	1	\$#BSYFL2	Second inline flag byte
		1... ....		\$#BCKPON	"B'10000000" Checkpoint only JOEBUSY and JOEDEVID
Comment					
<p>MEMBER NAME --&gt; \$QRBDCHK                      ROUTINE(S) ---&gt; \$QRBDCHK in HASPJQS                      MACRO(S) ----&gt; \$QRBDCHK                      \$QRBDCHK routines inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$QRBDFLG	\$QRBDCHK parameter flag
		1... ....		\$QRQNONE	"B'10000000" This JQE is currently not on a job queue
		.1.. ....		\$QRQOTH	"B'01000000" This JQE is currently on an other queue
		..1. ....		\$QRQRBLD	"B'00100000" This JQE is currently on the rebuild queue
Comment					
<p>MEMBER NAME --&gt; \$#RBDCHK                      ROUTINE(S) ---&gt; \$#RBDCHK in HASPJOS                      MACRO(S) ----&gt; \$#RBDCHK                      \$#RBDCHK routines inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$#RBDFLG	\$#RBDCHK parameter flag
		1... ....		\$#RQNONE	"B'10000000" This JOE is currently not on an output queue
		.1.. ....		\$#RQRBLD	"B'01000000" This JOE is currently on the rebuild queue
		..1. ....		\$#RQOTH	"B'00100000" This JOE is on one of the normal output queues
Comment					
<p>MEMBER NAME --&gt; QJQEVER                      ROUTINE(S) ---&gt; QJQEVER in HASPJQS                      MACRO(S) ----&gt; None                      QJQEVER routine inline parameter list.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$QJVPFLG	QJQEVER parameter flag
		1... ....		\$QJVVALNF	"B'10000000" Validate that this is not a free JQE
		.1.. ....		\$QJVRETC	"B'01000000" Validate JQE and return RC(do not abend)
		..1. ....		\$QJVNJQA	"B'00100000" Do not allow if JQA

## \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$QSUSE ROUTINE(S) ---> \$QSUSE in HASPNUC MACRO(S) ----> \$QSUSE Obtain JES2 queues parameter list Note: Update both HASMPERF and the QSUCB DSECT in \$PERFCB if this inline parameter list changes.					
End of Comment					
4	(4)	BITSTRING 1... ....	1	\$QSUFLG1 \$QSU1LUR	\$QSUSE parameter flag "B'10000000" Passively wait for queues
5	(5)	BITSTRING	1		Reserved for future use
6	(6)	CHARACTER	8	\$QSUSECT	Control Section name
14	(E)	CHARACTER	8	\$QSUSEQF	Invoking seq number
14	(E)	X'16'	0	\$QSUPLEN	"*-PARMLIST" Length of this parm list
Comment					
MEMBER NAME --> QTYPESET ROUTINE(S) ---> QTYPESET in HASPJQS MACRO(S) ----> None QTYPESET routine inline parameter list.					
End of Comment					
4	(4)	BITSTRING 1... ....	1	\$QTYPFLG \$QTYALTE	QTYPESET parameter flag "B'10000000" Begin processing at the alternate spot (QTSTPRG)
Comment					
MEMBER NAME --> \$#REP ROUTINE(S) ---> \$#REP in HASPJOS MACRO(S) ----> \$#REP \$#REP routine inline parameter list.					
End of Comment					
4	(4)	BITSTRING 1... .... .1.. .... .... 1..	1	\$#REPFLG \$#REPW \$#REPC \$#REPREM	\$#REP parameter flag "B'10000000" Wait if JOT is full "B'01000000" Copy JWELs from orig JOE "B'00001000" REMJOE was specified.
Comment					
MEMBER NAME --> \$RET ROUTINE(S) ---> \$CRETRN IN HASCLINK \$RETURN IN HASPNUC FSMRETRN IN HASPFSSM MACRO(S) ----> \$RETURN (assembler) \$RETURNP (PL/X) COMMON RETURN SERVICE ROUTINE'S INLINE PARAMETER LIST.					
End of Comment					
4	(4)	BITSTRING 1... .... .1.. .... .1.. .... .... 1111	1	\$RETFLAG \$RETANY \$RETPARM \$RETRCD \$RETRREG	\$RETURN MACRO OPTION FLAGS FOR USER & SUBTASK ENVIRONMENT "B'10000000" (USER,ANY) ENVIRONMENT "B'01000000" PARM=YES WAS SPECIFIED "B'00100000" Trace data passed "B'00001111" Register in save area with return address (if not R14)
5	(5)	BITSTRING	1		Reserved
6	(6)	BITSTRING	1	\$RETTFLG	Trace data flags (see \$SAVTFLG for bits)

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
7	(7)	BITSTRING	1	\$RETTDAT	TRACE data address register
8	(8)	BITSTRING	1	\$RETTLEN	TRACE data length
9	(9)	BITSTRING	1		Reserved

Comment

MEMBER NAME --> \$ROLL  
 ROUTINE(S) ---> \$ROLL in HASPEVTL  
 MACRO(S) ----> \$ROLL (assembler)  
 \$ROLL inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$ROLLSRV	Trace caller Service ID 1
5	(5)	BITSTRING	1		Reserved
6	(6)	SIGNED	2	\$ROLLOFF	HCT offset of Trace Tbl @
		.... ....		\$ROLJQEI	"X'02000000" CTRACE format ID for JQEs
		.... ....		\$ROLJOEI	"X'03000000" CTRACE format ID for JOEs
		.... ....		\$ROLDSPi	"X'04000000" CTRACE format ID for DISPs

Comment

MEMBER NAME --> \$RQGT  
 ROUTINE(S) ---> \$RQUEGET IN HASCRQUE  
 MACRO(S) ----> \$RQUE  
 \$RQUE 'GET' INLINE PARAMETER LIST.

End of Comment

4	(4)	BITSTRING	1	\$RQGTFL1	Flag byte
		1... ....		\$RQGT1RC	"B'10000000" Recovery request
		.1.. ....		\$RQGT1VE	"B'01000000" Veto routine provided

Comment

MEMBER NAME --> \$RRA  
 ROUTINE(S) ---> \$RROUTE IN HASPSERV  
 MACRO(S) ----> \$RROUTE  
 RROUTE AUTHORIZATION INLINE PARAMETER LIST.

End of Comment

4	(4)	BITSTRING	1	\$RRAFLG1	\$RRA FLAG BYTE
		1... ....		\$RRA1JOB	"B'10000000" RROUTE JOB REQUEST

Comment

MEMBER NAME --> \$RTA  
 ROUTINE(S) ---> \$RETABLE in HASPTABS  
 MACRO(S) ----> \$RETABLE  
 \$RETABLE routine inline parameter list

End of Comment

4	(4)	BITSTRING	1	\$RTAFLG1	Flag byte
		1... ....		\$RTA1MCT	"B'10000000" Offset of \$PAIR is in MCT
		.1.. ....		\$RTA1UCT	"B'01000000" Offset of \$PAIR is in UCT
		..1. ....		\$RTA1CCT	"B'00100000" Offset of \$PAIR is in HCCT
		.... ..1.		\$RTA1FRY	"B'00000010" Free DYNT YES specified
		.... ...1		\$RTA1FRN	"B'00000001" Free DYNT NO specified Free DYNT COND both off

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$SAV ROUTINE(S) ---> \$CSAVE IN HASCLINK \$GETSAVE IN HASPNUC FSMSAVE IN HASPFSSM MACRO(S) ----> \$SAVE (assembler) \$SAVEP (PL/X) COMMON SAVE SERVICE ROUTINE'S INLINE PARAMETER LIST.					
End of Comment					
4	(4)	BITSTRING	1	\$SAVFLAG	\$SAVE MACRO OPTION FLAGS FOR THE USER & SUBTASK ENVIRONMENT
		1... ....		\$SAVTRC	"B'10000000" TRACE THIS SAVE
		.1.. ....		\$SAVTRE	"B'01000000" TRE WAS PROVIDED
		..1. ....		\$SAVNRG	"B'00100000" REGS=NO WAS PROVIDED
		...1 ....		\$SAVARS	"B'00010000" SYSSTATE=AR at macro time
		.... 1...		\$SAVANY	"B'00001000" SYSSTATE=ANY at macro time
		.... .1..		\$SAVUANY	"B'00000100" \$ENVIRON (USER,ANY)
		.... ..1.		\$SAVTRCD	"B'00000010" Trace data passed
		.... ...1		\$SAVNRLS	"B'00000001" Registers in linkage stack (and REGS=NO)
5	(5)	BITSTRING	1	\$SAVFLG2	More option flags
		1... ....		\$SAVAM64	"B'10000000" SYSSTATE AMODE64=YES at macro time
6	(6)	CHARACTER	8	\$SAVNAME	EBCDIC LABEL
14	(E)	BITSTRING	1	\$SAVTFLG	Trace data flags
		1... ....		\$SAVTLRR	"B'10000000" \$SAVTLEN low half of reg
		.1.. ....		\$SAVTLHR	"B'01000000" \$SAVTLEN high half of reg
		..1. ....		\$SAVTLAR	"B'00100000" \$SAVTLEN access register
		...1 ....		\$SAVTLOF	"B'00010000" \$SAVTLEN is an offset
		.... 1...		\$SAVTDRR	"B'00001000" \$SAVTDAT low half of reg
		.... .1..		\$SAVTDHR	"B'00000100" \$SAVTDAT high half of reg
		.... ..1.		\$SAVTDAR	"B'00000010" \$SAVTDAT access register
15	(F)	BITSTRING	1	\$SAVTDAT	TRACE data address register
16	(10)	BITSTRING	1	\$SAVTLEN	TRACE data length
17	(11)	BITSTRING	1	\$SAVAREG	Access regs (of 0,1,15) to save
Comment					
MEMBER NAME --> \$SCD ROUTINE(S) ---> SCANDIAG IN HASCSCAN MACRO(S) ----> \$SCANDIA \$SCAN Diagnostic message routine					
End of Comment					
4	(4)	BITSTRING	1	\$SCDFLG1	\$SCANDIA MACRO OPTION FLAGS
		1... ....		\$SCD1WAR	"B'10000000" TYPE=WARN message
Comment					
MEMBER NAME --> \$SF ROUTINE(S) ---> \$SJBFINDD IN HASCSRJB MACRO(S) ----> \$SJBFINDD SUBSYSTEM JOB BLOCK FIND ROUTINE'S INLINE PARAMETER LIST.					
End of Comment					
4	(4)	BITSTRING	1	\$SFFLAG1	\$SJBFINDD MACRO OPTION FLAGS
		1... ....		\$SF1LOJ	"B'10000000" FIND THE LIFE OF JOB SJB
		.1.. ....		\$SF1SSIB	"B'01000000" FIND THE SSIB SJB
		..1. ....		\$SF1FRST	"B'00100000" FIND THE FIRST SJB FOR THE A.S.
		...1 ....		\$SF1LAST	"B'00010000" FIND THE LAST SJB FOR THE A.S.
		.... 1...		\$SF1ASID	"B'00001000" ASCB ASID PASSED IN R0



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$\$SJF            ROUTINE(S) ---&gt; \$\$SJBFREE IN HASCSRJB            MACRO(S) -----&gt; \$\$SJBFREE            CLEANUP SJB RELATED STORAGE ROUTINE INLINE            PARAMETER LIST.</p>					
End of Comment					
4	(4)	BITSTRING 1... .... .1... ....	1	\$\$SJFLAG1 \$\$SJFNPVT \$\$SJFMEM	\$\$SJBFREE MACRO OPTION FLAGS "B'10000000" PRIVATE STORAGE NOT AVAILBL "B'01000000" Dechain SJB, don't FREEMAIN
Comment					
<p>MEMBER NAME --&gt; \$\$SIGIO            ROUTINE(S) ---&gt; \$\$SIGIO in HASCSRDS and HASPSPOL            MACRO(S) -----&gt; \$\$SIGIO            Signature Record I/O parameter list</p>					
End of Comment					
4	(4)	BITSTRING 1... .... .1... .... ..1... .... ...1 .... .... 1..	1	\$\$SIGFLG1 \$\$SIG1W \$\$SIG1R \$\$SIG1SKP \$\$SIG1MQT \$\$SIG1VAV	\$\$SIGIO parameter flag "B'10000000" Write Signature record "B'01000000" Read Signature record "B'00100000" Skip invalid extents "B'00010000" MQT= specified "B'00001000" Verify TG is available
Comment					
<p>MEMBER NAME --&gt; \$\$SL            ROUTINE(S) ---&gt; \$\$SJBLOCK IN HASCSRJB            MACRO(S) -----&gt; \$\$SJBLOCK            SUBSYSTEM JOB BLOCK LOCK ROUTINE'S INLINE PARAMETER LIST.</p>					
End of Comment					
4	(4)	BITSTRING	1	\$\$SLFLAG1	\$\$SJBLOCK MACRO OPTION FLAGS
Comment					
<p>EQU B'10000000' RESERVED            EQU B'01000000' RESERVED</p>					
End of Comment					
		..1. ....		\$\$SL1RETN	"B'00100000" RETURN TO CALLER IF SJB LOCK OWNER IS NON-DISPATCHABLE
		...1 ....		\$\$SL1WAIT	"B'00010000" RETURN TO CALLER IF SJB LOCK IS NOT AVAILABLE (RC=16)
		.... 1..		\$\$SL1ACPT	"B'00001000" Return with RC=4 if lock is already held by the same TCB
Comment					
<p>MEMBER NAME --&gt; \$\$SPIN            ROUTINE(S) ---&gt; HFEXSPIN in HASCDSOC            MACRO(S) -----&gt; \$\$SPIN            Data set dynamic spin routine</p>					
End of Comment					
4	(4)	BITSTRING 1... .... .1... ....	1	\$\$SPNFLG1 \$\$SPN1CY \$\$SPN1NL	Option flags "B'10000000" Spin companion file too "B'01000000" SDB lock not held

# \$PARMLST Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$SU ROUTINE(S) ---> \$SJBUNLK in HASCSRJB MACRO(S) ----> \$SJBLOCK TYPE=FREE SJB Unlock routine's inline parameter list					
End of Comment					
4	(4)	BITSTRING 1... ..	1	\$SUFLAG1 \$SU1FREE	\$SJBLOCK macro option flags "B'10000000" FREESJB=YES, free the SJB after unlocking it
Comment					
MEMBER NAME --> \$SYMTT ROUTINE(S) ---> \$SYMTT in HASCSRDS MACRO(S) ----> none SYMREC creation for sniffer					
End of Comment					
4	(4)	BITSTRING	1	\$SYMTTF1	\$SYMTT parameter flag
4	(4)	X'1'	0	\$SYM1SNF	"1" Trackgroup falsely thought to be unavail.
4	(4)	X'2'	0	\$SYM1ALT	"2" Trackgroup falsely thought to be available by \$STRACK
4	(4)	X'3'	0	\$SYM1ALS	"3" Trackgroup falsely thought to be available by \$STRACK
4	(4)	X'4'	0	\$SYM1UNA	"4" Trackgroup being purged not owned by purging job
4	(4)	X'5'	0	\$SYM1BLO	"5" Trackgroup which was in BLOB returned to service
4	(4)	X'6'	0	\$SYM1JQA	"6" JQASUMSK did not reflect sniffed track group
4	(4)	X'7'	0	\$SYM1R0	"7" Key of first SPOOL block <> to signature key
Comment					
MEMBER NAME --> \$TGMSET ROUTINE(S) ---> \$TGMSET in HASPTRAK MACRO(S) ----> \$TGMSET \$TGMSET routine inline parameter list. NOTE: - \$TGFLAG AND \$TGFLAG2 MUST STAY TOGETHER!					
End of Comment					
Comment					
----- Option flags -----					
End of Comment					
4	(4)	BITSTRING 1... .. ..1. .... ...1 ....	1	\$TGFLAG \$TGCNTYS \$TGTTTEST \$TGTPSET	\$TGMSET parameter flag "B'10000000" COUNT=YES, Upd DAS counts "B'00100000" TYPE=TEST, Test bit only "B'00010000" TYPE=SET, Set the bit only
4	(4)	X'30' .... 1... .... .1.. .... ..1.	0	\$TGTTSET \$TGQSYES \$TGSETON \$TGMMQT	"\$TGTTTEST+\$TGTPSET" TYPE=TESTSET, Test and set "B'00001000" QSUSE=YES, get the QSUSE "B'00000100" SET=ON Turn bit on in map "B'00000010" MQT= was specified
Comment					
----- Flags for TGMTYPE= -----					
End of Comment					
5	(5)	BITSTRING 1... ..	1	\$TGFLAG2 \$TG2MAP	\$TGMSET parameter flag "B'10000000" TGMTYPE=TGMAP specified

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		\$TG2BAD	"B'01000000" TGMTYPE=TGBAD specified
		..1. ....		\$TG2BTRK	"B'00100000" TGMTYPE=BADTRTG specified
		...1 ....		\$TG2OTHR	"B'00010000" TGMTYPE=OTHER specified

Comment

MEMBER NAME --> \$#TJEV  
 ROUTINE(S) ---> \$#TJEV in HASPJOS  
 MACRO(S) ----> \$#TJEV  
 \$#TJEV routine inline parameter list.

End of Comment

4	(4)	BITSTRING	1	\$#TJVFLG	\$#TJEV parameter flag
		1... ....		\$#TADD	"B'10000000" Add JOE to exclusion vector
		.1.. ....		\$#TSEARCH	"B'01000000" Search for JOE in excl list
		..1. ....		\$#TSEARCH	"B'00100000" Search for JOE in all lists
		...1 ....		\$#TPURGE	"B'00010000" Purge JOE from all lists
		.... 1..		\$#TMOVE	"B'00001000" Move excl bit to new JOE

Comment

MEMBER NAME --> \$TRK  
 ROUTINE(S) ---> \$TRACK IN HASPTRAK, \$STRAK IN HASCSRIC  
 MACRO(S) ----> \$TRACK  
 SPOOL SPACE ALLOCATION ROUTINE'S INLINE PARAMETER LIST.

End of Comment

4	(4)	BITSTRING	1	\$TRFLAG1	\$TRACK MACRO OPTION FLAGS
		.... 1..		\$TR1SDB	"B'00001000" SDB provided
		.... .1..		\$TR1SJIO	"B'00000100" SJIOB provided
		.... ..1.		\$TR1WRPM	"B'00000010" WRPRIM=NO, DON'T WRITE PRIMARY IOT
		.... ...1		\$TR1WTNO	"B'00000001" WAIT=NO, DO NOT WAIT FOR BLOBBING
5	(5)	BITSTRING	1		RESERVED FOR FUTURE USE

Comment

MEMBER NAME --> TRP  
 ROUTINE(S) ---> \$TRACER IN HASCSRIC, HASPEVTL IN HASPEVTL  
 MACRO(S) ----> \$TRACE  
 JES2 EVENT TRACE LOG PROCESSOR INLINE PARAMETER LIST.

End of Comment

4	(4)	BITSTRING	1	TRPID	TRACE ID
5	(5)	BITSTRING	1	TRPFLAG1	ENVIRON/TYPE-MOVED TO TTETRPET
		1... ....		TRP1USER	"B'10000000" ENVIRON=USER
		.1.. ....		TRP1STSK	"B'01000000" ENVIRON=SUBTASK
		..1. ....		TRP1JES2	"B'00100000" ENVIRON=JES2
		...1 ....		TRP1FSS	"B'00010000" ENVIRON=FSS

Comment

EQU B'00001000' RESERVED FOR FUTURE USE  
 EQU B'00000100' RESERVED FOR FUTURE USE  
 EQU B'00000010' RESERVED FOR FUTURE USE

End of Comment

6	(6)	CHARACTER	8	TRP1SPIN TRPNAME	"B'00000001" SPIN THE LOG AT CURRENT TABLE TRACE SYMBOL
---	-----	-----------	---	---------------------	--

# \$PARMLST Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> TSR ROUTINE(S) ---> \$TOKENSR IN HASCSRIC MACRO(S) ----> \$TOKENSR Name/Token Service parameter list					
End of Comment					
4	(4)	BITSTRING	1	\$TSRFLG1	Flags
		1... ....		\$TSR1RET	"B'10000000" RETRIEVE
		.1.. ....		\$TSR1DEL	"B'01000000" DELETE
		..1. ....		\$TSR1CRE	"B'00100000" CREATE
		...1 ....		\$TSR1PRS	"B'00010000" PERSIST=YES
		.... 1...		\$TSR1CBA	"B'00001000" CBADDR= was specified
Comment					
MEMBER NAME --> VALSCQJQ ROUTINE(S) ---> VALSCQJQ IN HASPRTAM MACRO(S) ----> Passes inline parameter list for \$SCQJQE validation					
End of Comment					
4	(4)	ADDRESS	1	\$VSJFLAG	VALSCQJQ parameter flag
		1... ....		\$RCPINIT	"B'10000000" Remote console processor initialization
Comment					
MEMBER NAME --> \$WT ROUTINE(S) ---> \$WAIT IN HASPNUC MACRO(S) ----> \$WAIT, \$XECBSRV PCE WAIT MACRO INTERFACE TO THE JES2 DISPATCHER. Note: Update both HASMPERF and the WTCB DSECT in \$PERFCB if this inline parameter list changes.					
End of Comment					
4	(4)	BITSTRING	1	\$WTFLAG1	\$WAIT MACRO OPTION FLAGS
		1... ....		\$WT1RES	"B'10000000" \$WAIT FOR A RESOURCE
		.1.. ....		\$WT1XECB	"B'01000000" \$WAIT ON AN EXTENDED ECB
		..1. ....		\$WT1RETN	"B'00100000" \$WAIT IS TO RETURN WITHOUT WAITING (USED BY \$XECBSRV)
		...1 ....		\$WT1INHNT	"B'00010000" INHIBIT=NO specified on \$WAIT call (ie ignore \$WTINHBT)
		.... 1...		\$WT1MCLR	"B'00001000" PERFDATA monitor caller id
5	(5)	BITSTRING	1	\$WTINHBT	INHIBITOR (PREVENTS REDISPATCHING PCE BEFORE SPECIFIC \$POST)
6	(6)	CHARACTER	8	\$WTCSECT	Control Section name
14	(E)	CHARACTER	8	\$WTSEQF	Invoking seq number
22	(16)	ADDRESS	2	\$WTRESQO	RESOURCE QUEUE OFFSET OR 0
22	(16)	X'18'	0	\$WTPLEN	"*-PARMLIST" Length of this parm list
Comment					
MEMBER NAME --> \$WS ROUTINE(S) ---> SRVWSCAN IN HASPSERV MACRO(S) ----> \$WSSCAN SCAN THE WS OPERAND AND CREATE A WORK SELECTION LIST PARAMETER LIST					
End of Comment					
4	(4)	ADDRESS	1	\$WSLISTL	

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>MEMBER NAME --&gt; \$WSU            ROUTINE(S) ---&gt; SRVSETUP IN HASPSERV            MACRO(S) ----&gt; \$WSSETUP            SCAN THE WS OPERAND AND CREATE A WORK SELECTION LIST PARAMETER LIST</p>					
End of Comment					
4	(4)	ADDRESS	1	\$VOLFLD	DEVICE VOLUME FIELD
5	(5)	ADDRESS	1	\$VOLNMFD	DEVICE VOLUME NUMBER FIELD
6	(6)	ADDRESS	1	\$WSFLGOF	Work selection flag offset
7	(7)	BITSTRING	1	\$WSTYPE	CB type passed
7	(7)	X'0'	0	\$WSTUNK	"0" Unknown CB type
7	(7)	X'1'	0	\$WSTWSP	"1" WSP was passed
Comment					
<p>MEMBER NAME --&gt; \$WR            ROUTINE(S) ---&gt; \$WTOC or \$WTOR in HASPCON            MACRO(S) ----&gt; \$WTO            JES2 Main task environment \$WTO inline parameter list. The code in HASPCON depends on this parm list being in this order.</p>					
End of Comment					
4	(4)	ADDRESS	1	\$WRPFLAG	WTO inline parm. flag byte
4	(4)	X'5'	0	\$WREXEND	*** End of execute form parameter list
5	(5)	ADDRESS	1	\$WRTYPE	WTO type
6	(6)	ADDRESS	1	\$WRCLSPR	Class (high order 4 bits), Priority (low order 4 bits)
7	(7)	ADDRESS	1	\$WRROUTE	WTO Routing Information
8	(8)	ADDRESS	1	\$WRLEN	Message Length
9	(9)	BITSTRING	1	\$WRLINTP	MLWTO line type
9	(9)	X'A'	0	\$WRSTEND	*** End of standard form parameter list
Comment					
<p>MEMBER NAME --&gt; \$WT            ROUTINE(S) ---&gt; \$PREWTO IN HASCSRIC            MACRO(S) ----&gt; \$WTO            USER AND SUBTASK ENVIRONMENT \$WTO INLINE PARAMETER LIST.</p>					
End of Comment					
4	(4)	ADDRESS	1	\$WTPFLAG	JES2 Parameter flag byte
5	(5)	ADDRESS	1	\$WTCLASS	JES2 DESCRIPTOR CODE
6	(6)	ADDRESS	1	\$WTRROUTE	JES2 ROUTE CODE
7	(7)	BITSTRING	1	\$WTLINTP	MLWTO line type
Comment					
<p>\$WTO IN-LINE PARAMETER FLAGS            \$WRPFLAG and \$WTPFLAG. For \$WTPFLAG only            \$WTOJOBY and \$WTODOMT are processed.</p>					
End of Comment					
	1... ..			\$WTOSTDL	"B'10000000" STANDARD OR LIST FORM \$WTO, CMB TEMPLATE FOLLOWS DIRECTLY
	.1.. ....			\$WTOCMBL	"B'01000000" MF=EX SPECIFIED (LONG CMB TMPL)
	..1. ....			\$WTOJOBY	"B'00100000" JOB=YES SPECIFIED
	...1 ....			\$WTOWAIT	"B'00010000" WAIT=YES SPECIFIED
	.... 1...			\$WTOLNRO	"B'00001000" MESSAGE LENGTH IN R0 (BYTE 3)
	.... .1..			\$WTODOMT	"B'00000100" Delete outstanding action message if task terminates

# \$PARMLST Map

Offsets		Dec	Hex	Type/Value	Len	Name (Dim)	Description
				.... .1.		\$WTODMND	"B'00000010" Use \$GETCMB DEMANDCMB=YES if necessary
				.... ...1		\$WTOTEXT	"B'00000001" TEXT= specified on \$WTO (implies R0 - CONNECT and R1 - TEXT)
Comment							
<p>\$WROUTE - Logical console definitions</p>							
End of Comment							
				.... ...1		\$LOG	"X'01" SYSTEM LOG CONSOLE
				.... .1.		\$ERR	"X'02" ERROR CONSOLE MCS ROUTING CODE=(10)
				.... .1..		\$UR	"X'04" UNIT RECORD OPERATIONS AREA MCS ROUTING CODE=(7)
				.... 1...		\$TP	"X'08" TELE-PROCESSING OPERATIONS MCS ROUTING CODE=(8)
				...1 ....		\$TAPE	"X'10" TAPE HANDLING OPERATIONS MCS ROUTING CODE=(3,4,5,6)
				..1. ....		\$MAIN	"X'20" CHIEF OPERATORS AREA MCS ROUTING CODE=(1,2)
				.1.. ....		\$SEC	"X'40" SYSTEM SECURITY MCS ROUTING CODE=(9)
				1... ....		\$SPARE1	"X'80" SPARE 1 MCS ROUTING CODE=(14)
				.111 1111		\$ALL	"X'7F" ALL UNRESERVED LOCAL CONS.
Comment							
<p>\$WROUTE - Logical console definitions for user or JES2 subtask environment.            \$LOG EQU X'01' SYSTEM LOG CONSOLE (DEFINED ABOVE)            \$ERR EQU X'02' ERROR CONSOLE (DEFINED ABOVE)</p>							
End of Comment							
				.... .1..		\$MCINFO	"X'04" MASTER CONSOLE INFORMATION
				.... 1...		\$PGINFO	"X'08" PROGRAMMER INFORMATION
Comment							
<p>\$WTCLASS or \$WRCLSPR            Message Class Definitions - High order 4 bits.            (Only those bits should be used to maintain network compatability due to this byte being part of the NJE architecture).            In the main task, only \$DOMACT has meaning, which determines whether the the message is descriptor code 2 (when on) or 4 (when off).            Priority - Low order 4 bits. Only used by \$WRCLSPR.</p>							
End of Comment							
				...1 ....		\$TRIVIA	"X'10" NON-ESSENTIAL MESSAGES
				..11 ....		\$NORMAL	"X'30" NORMAL MESSAGES
				.1.. ....		\$JOBSTAT	"X'40" JOB STATUS MESSAGES
				.1.1 ....		\$ACTION	"X'50" MESSAGES REQUIRING OPERATOR ACTION
				.111 ....		\$ALWAYS	"X'70" MESSAGES WHICH SHOULD ALWAYS BE SENT
				1... ....		\$DOMACT	"X'80" ACTION REQUIRING A \$DOM FLAG
7	(7)	X'1'			0	\$LO	"1" LOW PRIORITY
7	(7)	X'4'			0	\$ST	"4" STANDARD PRIORITY
7	(7)	X'7'			0	\$HI	"7" HIGH PRIORITY

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
MEMBER NAME --> \$VWP					
ROUTINE(S) ---> VALWTOPL IN HASPCON					
MACRO(S) ---->					
PASSES THE MODULE NAME AND FUNCTION ID THRU THE					
INLINE PARAMETER LIST. FOR \$SYMREC GENERATION					
End of Comment					

4	(4)	CHARACTER	8	\$VWPMODN	MODULE NAME PARAMETER 1
12	(C)	CHARACTER	8	\$VWPFUNC	FUNCTION ID PARAMETER 2

Comment					
MEMBER NAME --> \$XM					
ROUTINE(S) ---> \$XMPOST IN HASCSTRIC					
MACRO(S) ----> \$XMPOST					
CROSS MEMORY POST SERVICE ROUTINE INLINE PARAMETER LIST					
End of Comment					

4	(4)	BITSTRING	1	\$XMFLAG1	FLAG 1
		1... ....		\$XM1XMP	"B'10000000" CROSS MEMORY PLIST WAS PASSED
		.1... ....		\$XM1QUIK	"B'01000000" QUICK POSTING IS ALLOWED
		..1. ....		\$XM1COMP	"B'00100000" COMPLETION CODE WAS CODED
		...1 ....		\$XM1LPST	"B'00010000" Parm list mapping that enables lost POST detection being used
		.... 1...		\$XM1STKN	"B'00001000" STOKEN= was specified
		.... .1..		\$XM1TTKN	"B'00000100" TTOKEN= was specified
		.... ..1.		\$XM1NPST	"B'00000010" TERM_TCB=NOPOST specified

**\$PARMLST Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$\$POFLG1	4		\$\$JINIT	5	20
\$\$PO1BRA	4	80	\$\$JLONG	4	80
\$\$PO1BR3	4	8	\$\$JPURGE	4	20
\$\$PO1ELM	4	20	\$\$JSERCH	4	40
\$\$PO1RUN	4	10	\$\$JWLFLG	4	
\$\$PO1SYS	4	40	\$\$JWLFL2	5	
\$\$ADDFLG	4		\$\$PFRJOA	4	8
\$\$ADDJOA	4	80	\$\$PJWEL	4	10
\$\$BACTOF	4	40	\$\$PUTFLG	4	
\$\$BACTON	4	80	\$\$RBDFLG	4	
\$\$BCKPNO	4	1	\$\$REPC	4	40
\$\$BCKPON	5	80	\$\$REPFLG	4	
\$\$BDEVID	4	2	\$\$REPREM	4	8
\$\$BDODEV	4	4	\$\$REPW	4	80
\$\$BPOST	4	8	\$\$RQNONE	4	80
\$\$BREAL	4	10	\$\$RQOTH	4	20
\$\$BSYFLG	4		\$\$RQRBLD	4	40
\$\$BSYFL2	5		\$\$TADD	4	80
\$\$BTRACE	4	20	\$\$TJVFLG	4	
\$\$DSFJOA	4	80	\$\$TMOVE	4	8
\$\$DSPFLG	4		\$\$TPURGE	4	10
\$\$JADD	4	10	\$\$TSERAN	4	20
\$\$JALL	4	2	\$\$TSERCH	4	40
\$\$JANCHR	5	10	\$\$ACTION	7	50
\$\$JANY	4	1	\$\$AEOJFL1	4	
\$\$JATTCH	5	40	\$\$AEOJ1EM	4	40
\$\$JCOND	4	4	\$\$AEOJ1JT	4	80
\$\$JDETC	5	80	\$\$ALL	7	7F
\$\$JFORCE	4	8	\$\$ALWAYS	7	70

## \$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$BTGBMTR	4	80	\$CRJ1CLR	4	8
\$BTGBTGM	4	40	\$CRJ1FRE	4	10
\$BTGFLG1	4		\$CWTOFLG	4	
\$BTMACT	4		\$CWTOFLG	4	40
\$BTMCLR	4	1C	\$CWTOFLG	4	80
\$BTMCRON	5	40	\$CWTONWT	4	20
\$BTMCRT	4	0	\$DGBACT	4	
\$BTMCRTC	5	80	\$DGBCAT	6	2
\$BTMDSTC	4	2C	\$DGBCB	6	
\$BTMDSTP	4	4	\$DGBCKPT	4	3
\$BTMERCDC	5	10	\$DGBDYN	6	FF
\$BTMFLG	5		\$DGBFLAG	5	
\$BTMNOAT	5	20	\$DGBFLG2	7	
\$BTMRST	4	C	\$DGBFREE	4	5
\$BTMSET	4	8	\$DGBFTCH	4	1
\$BTMSETR	4	20	\$DGBINT	6	0
\$BTMSOFF	4	18	\$DGBJQE	6	1
\$BTMSON	4	14	\$DGBNAME	5	10
\$BTMTEST	4	10	\$DGBNEXT	4	2
\$BTMTSRF	4	28	\$DGBNRDD	5	1
\$BTMTSRO	4	24	\$DGBNUPD	5	2
\$CBCKPTB	7		\$DGBQUSUS	5	40
\$CBCKPTF	C		\$DGBRETN	4	4
\$CBIFLG1	4		\$DGBSPEC	5	4
\$CBIFLG2	5		\$DGBSTSP	4	6
\$CBSPOLP	A		\$DGBTNAM	8	
\$CBSTORP	8		\$DGBTOKN	5	8
\$CBVERID	E		\$DGBUPDT	5	20
\$CBVERIX	12		\$DGBWAIT	5	80
\$CB1COND	4	1	\$DGBWSCQ	6	3
\$CB1EXIT	4	40	\$DGB2CRE	7	80
\$CB1FREE	4	4	\$DGB2NEV	7	8
\$CB1NOVF	4	20	\$DGB2PAD	7	20
\$CB1NSJB	4	10	\$DGB2PBE	7	10
\$CB1SJIO	4	8	\$DGB2UNC	7	4
\$CB1WAIT	4	2	\$DGB2UNK	7	40
\$CB2FSSM	5	20	\$DGDFLG1	4	
\$CB2MQTR	5	8	\$DGDFLG2	5	
\$CB2NORF	5	2	\$DGD1BRO	4	4
\$CB2SPLQ	5	4	\$DGD1CRE	4	2
\$CB2SUPM	5	10	\$DGD1FET	4	80
\$CB2TWAT	5	40	\$DGD1FRE	4	8
\$CB2WRIT	5	80	\$DGD1FTN	4	40
\$CDCFLG1	5		\$DGD1NUP	4	10
\$CDCTAPT	4	2	\$DGD1RES	4	1
\$CDCTCDC	4	5	\$DGD1UPD	4	20
\$CDCTDCT	4	1	\$DGD2ACC	5	80
\$CDCTRAT	4	4	\$DGD2CAC	5	40
\$CDCTSCK	4	3	\$DGD2WAI	5	20
\$CDCTXRQ	4	6	\$DGGFLG1	4	
\$CDCTYPE	4		\$DGGFLG2	5	
\$CDC1BRC	5	4	\$DGG1CKP	4	2
\$CDC1BRO	5	20	\$DGG1CRE	4	80
\$CDC1CRE	5	80	\$DGG1FET	4	20
\$CDC1INV	5	10	\$DGG1FRE	4	4
\$CDC1SYN	5	40	\$DGG1NUP	4	8
\$CDC1VAL	5	8	\$DGG1RED	4	40
\$CFXFLG1	4		\$DGG1UPD	4	10
\$CFX1RSP	4	80	\$DGG1WAI	4	1
\$CKQRTN	4		\$DGG2FTN	5	80
\$CPL1	4		\$DGTFLG1	4	
\$CPL1CDY	4	80	\$DGTFLG2	5	
\$CPL1HXN	4	40	\$DGT1AQR	4	2
\$CRJFLG1	4		\$DGT1AQS	4	4
\$CRJ1ALC	4	20	\$DGT1FET	4	80



\$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$DGT1FTN	4	40	\$DJCHAIN	6	
\$DGT1TYP	4	1	\$DJCVER	3	2
\$DGT1UPD	4	20	\$DJFLAG2	1	
\$DGT2BCH	5	2	\$DJFLAG3	2	
\$DGT2BRO	5	80	\$DJFLAG4	4	
\$DGT2CPY	5	4	\$DJFLAG5	5	
\$DGT2CRE	5	10	\$DJLEN	6	8
\$DGT2FRE	5	8	\$DJVERS	3	
\$DGT2JBC	5	20	\$DJXFJAX	5	40
\$DGT2WAI	5	40	\$DJXFLGS	5	
\$DGVFLG1	4		\$DJXFNCK	5	20
\$DGV1CKP	4	2	\$DJXFWT	5	80
\$DGV1CRE	4	40	\$DJXREQ	4	
\$DGV1FET	4	80	\$DJXRQCK	4	3
\$DGV1FRE	4	4	\$DJXRQCR	4	1
\$DGV1NUP	4	8	\$DJXRQFE	4	2
\$DGV1RED	4	20	\$DJXRQFR	4	5
\$DGV1UPD	4	10	\$DJXRQRS	4	6
\$DGV1WAI	4	1	\$DJXRQRT	4	4
\$DGWFLG1	4		\$DJ2CONF	1	8
\$DGWFLG2	5		\$DJ2DSRV	1	80
\$DGW1BRO	4	1	\$DJ2KEEP	1	20
\$DGW1CKP	4	4	\$DJ2NWAT	1	10
\$DGW1CRE	4	2	\$DJ2POST	1	4
\$DGW1FET	4	80	\$DJ2SPCL	1	40
\$DGW1FRE	4	8	\$DJ2UCON	1	2
\$DGW1FTN	4	40	\$DJ2URFR	1	1
\$DGW1NUP	4	10	\$DJ3#PSY	2	4
\$DGW1UPD	4	20	\$DJ3MAX	2	2
\$DGW2WAI	5	80	\$DJ3NUPD	2	10
\$DILCVER	5	1	\$DJ3QPSY	2	8
\$DILFLG1	6		\$DJ3RCVY	2	1
\$DILFLG2	7		\$DJ3READ	2	80
\$DILF1#P	6	1	\$DJ3RELE	2	40
\$DILF1CL	6	80	\$DJ3WDEF	2	20
\$DILF1FL	6	10	\$DOACKPT	1	C
\$DILF1IM	6	40	\$DOACT	1	
\$DILF1ND	6	4	\$DOACTOR	6	
\$DILF1PO	6	8	\$DOAFET	1	4
\$DILF1QP	6	2	\$DOAFETN	1	0
\$DILF1WA	6	20	\$DOAFLD	1	10
\$DILF2CK	7	10	\$DOAFREE	1	18
\$DILF2FN	7	4	\$DOAQLOK	1	1C
\$DILF2FP	7	2	\$DOAREFR	1	14
\$DILF2FT	7	1	\$DOARET	1	8
\$DILF2GM	7	8	\$DOASETA	1	20
\$DILF2PA	7	80	\$DOCHAIN	6	
\$DILF2QS	7	40	\$DOCKLEN	8	
\$DILF2SP	7	20	\$DOCKOFF	6	
\$DILIMME	8		\$DOCOVER	0	1
\$DILTJOE	4	2	\$DOFLAG2	2	
\$DILTJQE	4	1	\$DOFLAG3	3	
\$DILTYPE	4		\$DOFLAG4	4	
\$DILVERS	5		\$DOFLAG5	5	
\$DJACKPT	0	10	\$DOLEN	8	A
\$DJACT	0		\$DOMACT	7	80
\$DJAFET	0	4	\$DOVERS	0	
\$DJAFETN	0	0	\$DO2CONF	2	4
\$DJAFLD	0	24	\$DO2DSRV	2	80
\$DJAFREE	0	18	\$DO2NROL	2	2
\$DJALOCK	0	8	\$DO2NWAT	2	10
\$DJAQLOK	0	20	\$DO2RCVY	2	1
\$DJAREFR	0	14	\$DO2READ	2	20
\$DJARET	0	C	\$DO2SPCL	2	40
\$DJASETA	0	1C	\$DO2WDEF	2	8

## \$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$DO3KEEP	3	40	\$FBMCTAD	6	1
\$DO3MAX	3	2	\$FBMCTAS	6	2
\$DO3NUPD	3	10	\$FBMCTA8	6	12
\$DO3POST	3	1	\$FBMCTDC	6	5
\$DO3QLOB	3	8	\$FBMCTDS	6	4
\$DO3RELE	3	80	\$FBMCTEP	6	6
\$DO3UCON	3	20	\$FBMCTHX	6	7
\$DO4#PSY	4	40	\$FBMCTKM	6	9
\$DO4KPJW	4	20	\$FBMCTOF	6	A
\$DO4MNJT	4	10	\$FBMCTPE	6	3
\$DO4PSTA	4	80	\$FBMCTPN	6	1
\$DSDFLG1	4		\$FBMCTPW	6	2
\$DSDFLG2	5		\$FBMCTRH	6	8
\$DSD2FOR	5	80	\$FBMCTSE	6	E
\$DSRFLG1	4		\$FBMCTSI	6	B
\$DSR1FRE	4	40	\$FBMCTSM	6	F
\$DSR1GET	4	80	\$FBMCTSR	6	C
\$DSR1LIV	4	20	\$FBMCTST	6	D
\$DSR1RFR	4	10	\$FBMCTUI	6	10
\$DSTCHAR	4	80	\$FBMCTUR	6	11
\$DSTFLG1	4		\$FBMCTYP	6	
\$DSTFLG2	5		\$FBMFLG1	4	
\$DSTGNRC	4	20	\$FBMFLG2	5	
\$DSTNRP	4	8	\$FBMID	7	
\$DSTNSPR	4	1	\$FBMIDV	7	81
\$DSTPRIM	4	4	\$FBMPLEN	10	14
\$DSTRDT	4	40	\$FBMSECT	8	
\$DSTUSER	4	2	\$FBMSEQF	10	
\$DST1EXP	4	10	\$FBM1ABA	4	1
\$DST2DFM	5	40	\$FBM1ABB	4	2
\$DST2IGN	5	80	\$FBM1ADD	4	40
\$DST2IGS	5	8	\$FBM1CNV	4	4
\$DST2IPD	5	4	\$FBM1INT	4	80
\$DST2IPY	5	10	\$FBM1LAS	4	20
\$DST2NUS	5	20	\$FBM1MFL	4	10
\$DST2NVU	5	2	\$FBM1TTB	4	8
\$DTRFLG1	4		\$FBM2ADJ	5	2
\$DTRMQTR	4	4	\$FBM2IFI	5	8
\$DTRNAME	6		\$FBM2INA	5	4
\$DTRRECV	1E		\$FBM2IVA	5	10
\$DTRRJCT	4	40	\$FBM2MTB	5	40
\$DTRRJQE	4	80	\$FBM2RES	5	80
\$DTRRNAM	4	10	\$FBM2TTZ	5	1
\$DTRRND	4	20	\$FBM2WID	5	20
\$DTRRSIG	4	8	\$FBSTORP	6	
\$DTRSECT	E		\$FB1HOLD	4	20
\$DTRSEQ	16		\$FB1PROT	4	80
\$DVFLG1	4		\$FDISECT	6	
\$DVLENG	5		\$FDISEQF	E	
\$DV1CHAR	4	40	\$FDITCHR	4	1
\$DV1JQE	4	80	\$FDITHEX	4	2
\$ERR	7	2	\$FDITUSI	4	3
\$ESTCRAT	4	80	\$FDITYPE	4	
\$ESTDLET	4	40	\$FEVFLG1	7	
\$ESTFCN	4		\$FEVMOSQ	12	A
\$ESTFLAG	5		\$FEVPVER	4	
\$ESTFNDM	5	80	\$FEVRCC	5	3
\$ESTNBR	A		\$FEVRCCU	5	1
\$ESTRECX	6		\$FEVRDC	5	4
\$ESTRTYA	C		\$FEVRDU	5	2
\$FACMOSQ	C	4	\$FEVREQ	5	
\$FACSECT	4		\$FEVSECT	A	
\$FACSEQF	C		\$FEVSEQF	12	
\$FBFLAG1	4		\$FEVUIFR	6	40
\$FBMCTAC	6	3	\$FEVUIVE	6	80

\$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$FEVUJDR	6	20	\$FST2FAD	5	80
\$FEVUNUL	6	10	\$FST2PTM	5	8
\$FEVUSE	6		\$FST2PTS	5	10
\$FEVWVER	8		\$FST2TAB	5	78
\$FEV1CSA	7	80	\$FTBFLG1	4	
\$FGAFLG1	4		\$FTB1CHN	4	80
\$FGAPLEN	E	12	\$FTRFLG1	4	
\$FGASECT	6		\$FTR1IOW	4	80
\$FGASEQF	E		\$FUCFLG1	4	
\$FGA1ACM	4	4	\$FUC1UNP	4	80
\$FGA1EYE	4	10	\$GCFLAG1	4	
\$FGA1GNS	4	40	\$GCRSVRD	5	
\$FGA1GTN	4	20	\$GCSIZE	6	
\$FGA1LUP	4	80	\$GC1LPRM	4	80
\$FGA1STD	4	E0	\$GFFLG1	C	
\$FGA1ZPM	4	8	\$GFFLG2	F	
\$FGFSECT	4		\$GFFLG3	4	
\$FGFSEQF	C		\$GFFLG4	5	
\$FIDCOFF	6		\$GFKEY	D	
\$FIDFLG1	4		\$GFLENV	8	
\$FID1ABN	4	40	\$GFSUBPL	E	
\$FID1CNT	4	80	\$GF1AR15	C	20
\$FIECOFF	6		\$GF1CHK0	C	8
\$FIEFLG1	4		\$GF1OHOM	C	0
\$FIE1ABN	4	40	\$GF1OPRI	C	1
\$FIE1CNT	4	80	\$GF1OSEC	C	2
\$FIE1HEA	4	20	\$GF1OSYS	C	3
\$FIGFLG1	4		\$GF1RS64	C	10
\$FIG1CNT	4	80	\$GF2FMN	F	1
\$FLEAPAR	16		\$GF2LC24	F	10
\$FLEFLG1	4		\$GF2LC31	F	20
\$FLESECT	6		\$GF2PGB	F	4
\$FLESEQF	E		\$GF2RS31	F	40
\$FLE1RC	4	80	\$GF2UNCD	F	2
\$FMFLAG1	4		\$GF3BUFR	4	40
\$FMSFLG1	4		\$GF3FREE	4	10
\$FMSPLEN	E	12	\$GF3HTCB	4	20
\$FMSSECT	6		\$GF3JTCB	4	4
\$FMSSEQF	E		\$GF3LVR0	4	80
\$FMS1BLN	4	40	\$GF3PSWK	4	1
\$FMS1CBL	4	20	\$GF3TCBK	4	2
\$FMS1IND	4	10	\$GF3TCBY	4	8
\$FMS1WID	4	80	\$GF4BAKR	5	8
\$FM1\$ERR	4	80	\$GF4NOLV	5	10
\$FM1CLOS	4	40	\$GF4OAUZ	5	4
\$FPRFLG1	4		\$GF4SPR0	5	80
\$FPRSECT	6		\$GF4STOR	5	40
\$FPRSEQF	E		\$GF4TCUR	5	2
\$FPR1MLT	4	80	\$GF4ZERO	5	20
\$FREFLG1	4		\$GLWFLG1	4	
\$FRE1JQA	4	20	\$GLW1WT	4	80
\$FRE1NTR	4	40	\$GNHFLG1	4	
\$FRE1NW	4	80	\$GNH1WAT	4	80
\$FSAMOSQ	C	4	\$GTAFLG1	5	
\$FSASECT	4		\$GTAH1ST	5	80
\$FSASEQF	C		\$GTATYPE	4	
\$FSTFLG1	4		\$GTBFLG1	4	
\$FSTFLG2	5		\$GTBFTYP	5	
\$FSTSECT	6		\$GTB1FIX	4	40
\$FSTSEQF	E		\$GTB1LOW	4	10
\$FST1FRE	4	3	\$GTB1MUL	4	20
\$FST1GTC	4	2	\$GTB1WAT	4	80
\$FST1GTU	4	1	\$GTCFLG1	4	
\$FST2CTM	5	20	\$GTC1DMC	4	40
\$FST2CTS	5	40	\$GTC1WAT	4	80

## \$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$GTTBUSE	4		\$MCNTEST	6	B
\$GUCFLG1	4		\$MCPROPX	6	400
\$GUC1CNT	4	80	\$MCRMD24	6	8000
\$GUC1UNT	4	40	\$MCRSLVX	6	200
\$HI	7	7	\$MCTABL	6	100
\$IBFLAG1	4		\$MCTESTS	6	
\$IB1DAUG	4	8	\$MCVERS	6	1000
\$IB1DNCH	4	2	\$MSTCLR	4	2
\$IB1DPDB	4	80	\$MSTFLG1	5	
\$IB1DPRI	4	10	\$MSTSET	4	1
\$IB1DSJI	4	4	\$MSTTYPE	4	
\$IB1DSPN	4	40	\$MST1ECB	5	80
\$IB1D2ND	4	20	\$MST1EXP	5	20
\$ICFLAG1	4		\$MST1EXT	5	40
\$IC1IOT	4	20	\$NADFLG1	5	
\$IC1JOE	4	10	\$NADSTAT	4	
\$IC1LKNO	4	80	\$NAD1CES	5	10
\$IC1LOCK	4	40	\$NAD1NAT	5	80
\$JCLSNS	4	40	\$NAD1NCC	5	40
\$JCLSRM	4	80	\$NAD1STA	5	20
\$JCNFLG1	4		\$NGTFLG1	5	
\$JCN1PRO	4	80	\$NGTSTAT	4	
\$JCN1TST	4	40	\$NGT1FST	5	10
\$JCN1TSU	4	20	\$NGT1NAT	5	80
\$JOBCLAV	4		\$NGT1NCC	5	40
\$JOBSTAT	7	40	\$NGT1TOK	5	20
\$JSMACTA	4	1	\$NHRFLG1	4	
\$JSMACTN	4		\$NHRSRCB	5	
\$JSMACTX	4	2	\$NHR1XIT	4	80
\$JSMFLGS	7		\$NHWFLG1	4	
\$JSMJQE	7	80	\$NHW1FRE	4	80
\$JSMSJB	7	40	\$NHW1WAT	4	40
\$JSMSYAS	6	3	\$NHXFLG1	4	
\$JSMSYCV	6	1	\$NHXSRCB	5	
\$JSMSYIN	6	2	\$NHX1FRE	4	80
\$JSMSYMT	6		\$NHX1XIT	4	40
\$JSMTBCL	5	2	\$NMDFLG1	5	
\$JSMTBJS	5	1	\$NMDSTAT	4	
\$JSMTBLF	5		\$NMD1CES	5	8
\$JSMTBSS	5	3	\$NMD1FST	5	20
\$LGFLAG1	5		\$NMD1NAT	5	80
\$LGRQSTR	7		\$NMD1NCC	5	40
\$LGSUBP	4		\$NMD1STA	5	10
\$LG1MFRE	5	80	\$NNTFLG1	4	
\$LG1TXTL	6		\$NNT1FST	4	10
\$LG1WTO	5	40	\$NNT1MMA	4	4
\$LO	7	1	\$NNT1MTR	4	8
\$LOG	7	1	\$NNT1NOT	4	20
\$LVFLAG1	4		\$NNT1SET	4	80
\$LVQFLAG	5		\$NNT1TST	4	40
\$LVQOFF	6		\$NORMAL	7	30
\$LV1QSE	4	80	\$NOTFLG1	4	
\$LV1WAIT	4	40	\$NOT1JQE	4	20
\$MAIN	7	20	\$NOT1MEM	4	8
\$MCADDR	6	80	\$NOT1NJB	4	80
\$MCBYTES	5	2	\$NOT1NUM	4	40
\$MCCOMMN	6	4000	\$NOT1NUS	4	10
\$MCDELET	6	20	\$NRMFLG1	4	
\$MCDYNAM	6	40	\$NRM1ALL	4	40
\$MCFLAG1	4		\$NRM1STA	4	80
\$MCINFO	7	4	\$NSLFLG1	4	
\$MCMIT	6	2000	\$NSL1OBT	4	80
\$MCMMSG	4	80	\$NSL1SHR	4	20
\$MCMMSGS	4	40	\$NSL1WAT	4	40
\$MCNAME	6	800	\$NSYFLG1	4	

\$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$NSY1CKP	4	10	\$RTAFLG1	4	
\$NSY1CMP	4	40	\$RTA1CCT	4	20
\$NSY1LOC	4	20	\$RTA1FRN	4	1
\$NSY1MSG	4	8	\$RTA1FRY	4	2
\$NSY1NIT	4	4	\$RTA1MCT	4	80
\$NSY1REF	4	80	\$RTA1UCT	4	40
\$PBFLAG1	4		\$\$SAVAM64	5	80
\$PB1DAUG	4	20	\$\$SAVANY	4	8
\$PB1DPDB	4	80	\$\$SAVAREG	11	
\$PB1DSPN	4	40	\$\$SAVARS	4	10
\$PGINFO	7	8	\$\$SAVFLAG	4	
\$PRGFLG1	4		\$\$SAVFLG2	5	
\$PRG1ENF	4	40	\$\$SAVNAME	6	
\$PRG1JOA	4	20	\$\$SAVNRG	4	20
\$PRG1VFY	4	80	\$\$SAVNRLS	4	1
\$PSQFLG1	4		\$\$SAVTDAR	E	2
\$PSQ1JQE	4	40	\$\$SAVTDAT	F	
\$PSQ1MAS	4	80	\$\$SAVTDHR	E	4
\$PTAFLG1	4		\$\$SAVTDRR	E	8
\$PTA1CCT	4	20	\$\$SAVTFLG	E	
\$PTA1MCT	4	80	\$\$SAVTLAR	E	20
\$PTA1UCT	4	40	\$\$SAVTLEN	10	
\$QBACTOF	4	40	\$\$SAVTLHR	E	40
\$QBACTON	4	80	\$\$SAVTLOF	E	10
\$QBDEVID	4	1	\$\$SAVTLRR	E	80
\$QBDODEV	4	2	\$\$SAVTRC	4	80
\$QBHVCAT	4	4	\$\$SAVTRCD	4	2
\$QBNALCT	4	8	\$\$SAVTRE	4	40
\$QBREAL	4	10	\$\$SAVUANY	4	4
\$QBSYFLG	4		\$\$SCDFLG1	4	
\$QBTRACE	4	20	\$\$SCD1WAR	4	80
\$QJVALNF	4	80	\$\$SEC	7	40
\$QJVNJQA	4	20	\$\$SFFLAG1	4	
\$QJVPFLG	4		\$\$SF1ASID	4	8
\$QJVRETC	4	40	\$\$SF1FRST	4	20
\$QRBDFLG	4		\$\$SF1LAST	4	10
\$QRQNONE	4	80	\$\$SF1LOJ	4	80
\$QRQOTH	4	40	\$\$SF1SSIB	4	40
\$QRQRBLD	4	20	\$\$SIGFLG1	4	
\$QSUFLG1	4		\$\$SIG1MQT	4	10
\$QSUPLEN	E	16	\$\$SIG1R	4	40
\$QSUSECT	6		\$\$SIG1SKP	4	20
\$QSUSEQF	E		\$\$SIG1VAV	4	8
\$QSU1LUR	4	80	\$\$SIG1W	4	80
\$QTYALTE	4	80	\$\$SJFLAG1	4	
\$QTYPFLG	4		\$\$SJFMEM	4	40
\$RCPINIT	4	80	\$\$SJFNPVT	4	80
\$RETANY	4	80	\$\$SLFLAG1	4	
\$RETFLAG	4		\$\$SL1ACPT	4	8
\$RETPARM	4	40	\$\$SL1RETN	4	20
\$RETRREG	4	F	\$\$SL1WAIT	4	10
\$RETTDAT	7		\$\$SPARE1	7	80
\$RETTFLG	6		\$\$SPNFLG1	4	
\$RETTLEN	8		\$\$SPN1CY	4	80
\$RETTTRCD	4	20	\$\$SPN1NL	4	40
\$ROLDSPI	6	0	\$\$ST	7	4
\$ROLJOEI	6	0	\$\$SUFLAG1	4	
\$ROLJQEI	6	0	\$\$SU1FREE	4	80
\$ROLLOFF	6		\$\$SYMTTF1	4	
\$ROLLSRV	4		\$\$SYM1ALS	4	3
\$RQGTFL1	4		\$\$SYM1ALT	4	2
\$RQGT1RC	4	80	\$\$SYM1BLO	4	5
\$RQGT1VE	4	40	\$\$SYM1JQA	4	6
\$RRAFLG1	4		\$\$SYM1R0	4	7
\$RRA1JOB	4	80	\$\$SYM1SNF	4	1

## \$PARMLST Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
\$SYM1UNA	4	4	\$WTODOMT	7	4
\$TAPE	7	10	\$WTOJOBY	7	20
\$TGCNTYS	4	80	\$WTOLNR0	7	8
\$TGFLAG	4		\$WTOSTDL	7	80
\$TGFLAG2	5		\$WTOTEXT	7	1
\$TGMMQT	4	2	\$WTOWAIT	7	10
\$TGQSYES	4	8	\$WTPFLAG	4	
\$TGSETON	4	4	\$WTPLEN	16	18
\$TGTPSET	4	10	\$WTRESQO	16	
\$TGTTTEST	4	20	\$WTRROUTE	6	
\$TGTTSET	4	30	\$WTSEQF	E	
\$TG2BAD	5	40	\$WT1INHN	4	10
\$TG2BTRK	5	20	\$WT1MCLR	4	8
\$TG2MAP	5	80	\$WT1RES	4	80
\$TG2OTHR	5	10	\$WT1RETN	4	20
\$TP	7	8	\$WT1XECB	4	40
\$TRFLAG1	4		\$XMFLAG1	4	
\$TRIVIA	7	10	\$XM1COMP	4	20
\$TR1SDB	4	8	\$XM1LPST	4	10
\$TR1SJIO	4	4	\$XM1NPST	4	2
\$TR1WRPM	4	2	\$XM1QUIK	4	40
\$TR1WTNO	4	1	\$XM1STKN	4	8
\$TSRFLG1	4		\$XM1TTKN	4	4
\$TSR1CBA	4	8	\$XM1XMP	4	80
\$TSR1CRE	4	20	DSNRONLY	4	80
\$TSR1DEL	4	40	DSNVALL	4	0
\$TSR1PRS	4	10	EXITFLGS	C	
\$TSR1RET	4	80	EXITFSS	C	8
\$UR	7	4	EXITID	D	
\$VFYCHK	12	0	EXITJES2	C	10
\$VFYDSCA	12	1	EXITLNG	F	C
\$VFYDSIX	12	2	EXITMRC	E	
\$VFYHDB	12	3	EXITNAME	4	
\$VFYIOT	12	4	EXITRSVD	F	
\$VFYJCT	12	5	EXITSTSK	C	20
\$VFYJSMT	12	A	EXITTR	C	80
\$VFYNHSB	12	8	EXITUSER	C	40
\$VFYOCT	12	6	EXITXPL	C	4
\$VFYSWBI	12	7	PARMINST	0	
\$VFYTLBM	12	9	PARMLIST	0	
\$VOLFLD	4		PARMSTRT	4	
\$VOLNMFD	5		TRPFLAG1	5	
\$VSJFLAG	4		TRPID	4	
\$VWPFUNC	C		TRPNAME	6	
\$VWPMODN	4		TRP1FSS	5	10
\$WRCLSPR	6		TRP1JES2	5	20
\$WREXEND	4	5	TRP1SPIN	5	1
\$WRLEN	8		TRP1STSK	5	40
\$WRLINTP	9		TRP1USER	5	80
\$WRPFLAG	4				
\$WRROUTE	7				
\$WRSTEND	9	A			
\$WRTYPE	5				
\$WSFLGOF	6				
\$WSLISTL	4				
\$WSTUNK	7	0			
\$WSTWSP	7	1			
\$WSTYPE	7				
\$WTCLASS	5				
\$WTCSECT	6				
\$WTFLAG1	4				
\$WTINHBT	5				
\$WTLINTP	7				
\$WTOCMBL	7	40			
\$WTODMND	7	2			

## \$PARMWRK Information

### \$PARMWRK Heading Information

**Common Name:** PARMLIB Work Area  
**Macro ID:** \$PARMWRK  
**DSECT Name:** PRW  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** PRW  
 Offset: PRWID  
 Length: L'PRWID  
**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: Virtual storage is below the 16M line (contains DCB to read PARMLIB).  
 Real storage is anywhere  
**Size:** See PRWLEN  
**Created by:** HASPSXIT for the INCLUDE initialization statement  
**Pointed to by:** CIRPRMWR field of the CIRWORK data area  
 PRWPRW field of the PARMWRK data area  
**Serialization:** None required  
**Function:** The PARMWRK DSECT represents a data set used to read JES2 initialization statements from

### \$PARMWRK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PRW	, PROCLIB allocation DSECT
0	(0)	CHARACTER	4	PRWID	Eyecatcher
4	(4)	CHARACTER	54	PRWDSNMB	Data set name input area
58	(3A)	CHARACTER	1	PRWDSNEN	Work area for PSTINCDS
60	(3C)	SIGNED	4	(0)	Alignment
60	(3C)	CHARACTER	8	PRWUNIT	Data set unit
68	(44)	CHARACTER	6	PRWVOL	Data set VOLSER
74	(4A)	CHARACTER	44	PRWDSN (0)	Dataset name
74	(4A)	BITSTRING	58		DSN with work area
132	(84)	SIGNED	4	(0)	Alignment
132	(84)	CHARACTER	8	PRWMEMB (0)	Member name
132	(84)	BITSTRING	55		Member with work area
187	(BB)	CHARACTER	8	PRWIMEMB	Member of current dataset
195	(C3)	CHARACTER	8	PRWLMEMB	Logical Parmlib_member
204	(CC)	SIGNED	4	PRWSEND (0)	End of SCAN mapped areas
204	(CC)	X'CC'	0	PRWSLEN	"PRWSEND-PRW" Short PRW length (all fields referenced by SCAN)
204	(CC)	ADDRESS	4	PRWLDSB	Logical dataset Read buffer
208	(D0)	ADDRESS	4	PRWPRW	PRW chain pointer
212	(D4)	SIGNED	4	PRWLRCNT	Logical dataset rec counter
216	(D8)	BITSTRING	1	PRWFLAG1	General flags
		1... ....		PRW1DCON	"B'10000000" Enter console mode when EOF
		.1.. ....		PRW1DECN	"B'01000000" Enter error console mode on EOF
217	(D9)	BITSTRING	1	PRWFLAG2	General flags
		1... ....		PRW2IDSN	"B'10000000" DSNAME specified
		.1.. ....		PRW2IVOL	"B'01000000" VOLSER specified
		..1. ....		PRW2IUNT	"B'00100000" UNIT specified
		...1 ....		PRW2IMEM	"B'00010000" Member field specified
		.... 1...		PRW2ILPM	"B'00001000" Parmlib_member specified
		.... .1..		PRW2ISEQ	"B'00000100" Dataset is sequential
		.... ..1.		PRW2INST	"B'00000010" Nested INCLUDE
218	(DA)	BITSTRING	3		Reserved
224	(E0)	ADDRESS	4	PRWINIDS	INIDSNE pointer for include

# \$PARMWRK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Dynamic allocation work area					
End of Comment					
228	(E4)	ADDRESS	4	PRWRBPTR	Request block pointer
232	(E8)	SIGNED	4	(0)	Ensure aligned
232	(E8)	BITSTRING	20	PRWRB	DYNALLOC request block
252	(FC)	SIGNED	4	(0)	Ensure aligned
252	(FC)	BITSTRING	36	PRWRBX	Request block Extension
288	(120)	SIGNED	4	PRWTXTPT (0)	Text pointers
288	(120)	ADDRESS	4	PRWTXTP1	Address of text unit 1
292	(124)	ADDRESS	4	PRWTXTP2	Address of text unit 2
296	(128)	ADDRESS	4	PRWTXTP3	Address of text unit 3
300	(12C)	ADDRESS	4	PRWTXTP4	Address of text unit 4
304	(130)	ADDRESS	4	PRWTXTP5	Address of text unit 5
308	(134)	ADDRESS	4	PRWTXTP6	Address of text unit 5
Comment					
DYNAMIC Allocation text units:					
- Return DDNAME					
- DISP=SHR					
- DSNAME=dsname					
- UNIT=unit					
- VOLSER=volser					
End of Comment					
312	(138)	SIGNED	4	PRWATX (0)	Text units
312	(138)	BITSTRING	6	PRWATX1	Text unit 1 (Return DDNAME)
318	(13E)	CHARACTER	8	PRWDDNM	Returned DD Name
326	(146)	BITSTRING	6	PRWATX2	Text unit 2 (DISP)
332	(14C)	BITSTRING	8	PRWATX2D	Text unit 2 parm
340	(154)	BITSTRING	6	PRWATX3	Text unit 3 (Data set name)
346	(15A)	CHARACTER	44	PRWDDSN	Data set name
390	(186)	BITSTRING	6	PRWATX4	Text unit 4 (member name)
396	(18C)	CHARACTER	8	PRWDMEMB	Data set member name
404	(194)	BITSTRING	6	PRWATX5	Text unit 5 (UNIT)
410	(19A)	CHARACTER	8	PRWDUNIT	Data set unit
418	(1A2)	BITSTRING	6	PRWATX6	Text unit 6 (VOLSER)
424	(1A8)	CHARACTER	6	PRWDVOL	Data set VOLSER
Comment					
DYNAMIC Unallocate text units:					
- DDNAME					
End of Comment					
312	(138)	BITSTRING	6	PRWUTX1	Text unit 1 (DDNAME)
318	(13E)	BITSTRING	8	PRWUTX1D	Text unit 1 parm
326	(146)	BITSTRING	6	PRWUTX2	Text unit 2 unallocate perm
332	(14C)	BITSTRING	8	PRWUTX2D	Text unit 2 parm
Comment					
OPEN/CLOSE Work areas and DCB					
End of Comment					
432	(1B0)	DBL WORD	8	(0)	Alignment DATA CONTROL BLOCK
432	(1B0)	SIGNED	4	PRWD CB (0)	ORIGIN ON WORD BOUNDARY DIRECT ACCESS DEVICE INTERFACE
432	(1B0)	ADDRESS	4		DCBE ADDRESS
436	(1B4)	BITSTRING	12		FDAD, DVTBL



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
448	(1C0)	ADDRESS	4		KEYLEN, DEVT, TRBAL COMMON ACCESS METHOD INTERFACE
452	(1C4)	ADDRESS	1		BUFNO, NUMBER OF BUFFERS
453	(1C5)	ADDRESS	3		BUFCB, BUFFER POOL CONTROL BLOCK
456	(1C8)	ADDRESS	2		BUFL, BUFFER LENGTH
458	(1CA)	BITSTRING	2		DSORG, DATA SET ORGANIZATION
460	(1CC)	ADDRESS	4		IOBAD FOR EXCP OR RESERVED FOUNDATION EXTENSION
464	(1D0)	BITSTRING	1		BFTEK, BFALN, DCBE INDICATORS
465	(1D1)	ADDRESS	3		EODAD (END OF DATA ROUTINE ADDRESS)
468	(1D4)	BITSTRING	1		RECFM (RECORD FORMAT)
469	(1D5)	ADDRESS	3		EXLST (EXIT LIST ADDRESS) FOUNDATION BLOCK
472	(1D8)	CHARACTER	8		DDNAME
480	(1E0)	BITSTRING	1		OFLGS (OPEN FLAGS)
481	(1E1)	BITSTRING	1		IFLGS (IOS FLAGS)
482	(1E2)	BITSTRING	2		MACR (MACRO FORMAT) BSAM-BPAM-QSAM INTERFACE
484	(1E4)	BITSTRING	1		OPTCD, OPTION CODES
485	(1E5)	ADDRESS	3		CHECK OR INTERNAL QSAM SYNCHRONIZING RTN.
488	(1E8)	ADDRESS	4		SYNAD, SYNCHRONOUS ERROR RTN. (3 BYTES)
492	(1EC)	SIGNED	2		INTERNAL ACCESS METHOD FLAGS
494	(1EE)	ADDRESS	2		BLKSIZE, BLOCK SIZE
496	(1F0)	SIGNED	4		INTERNAL ACCESS METHOD FLAGS
500	(1F4)	ADDRESS	4		INTERNAL ACCESS METHOD USE QSAM INTERFACE
504	(1F8)	ADDRESS	4		EOBAD
508	(1FC)	ADDRESS	4		RECAD
512	(200)	SIGNED	2		QSW (FLAGS) AND EITHER DIRCT OR BUFOFF
514	(202)	ADDRESS	2		LRECL
516	(204)	BITSTRING	1		EROPT, ERROR OPTION
517	(205)	ADDRESS	3		CNTRL
520	(208)	SIGNED	2	(2)	RESERVED AND PRECL
524	(20C)	ADDRESS	4		EOB, INTERNAL ACCESS METHOD FIELD

Comment

DATA CONTROL BLOCK EXTENSION.

End of Comment

528	(210)	SIGNED	4	PRWDCBE (0)	0 Alignment and identifier
532	(214)	SIGNED	2		4 Length of DCBE, minimum is 56
534	(216)	BITSTRING	2		6 Reserved, should be zero
536	(218)	ADDRESS	4		8 0 if not open, OPEN points to DCB
540	(21C)	BITSTRING	4		C Disk address of current member
544	(220)	BITSTRING	1		10 Flags set by system
545	(221)	BITSTRING	1		11 Flags set by user
546	(222)	SIGNED	2		12 Number of stripes if extended format
548	(224)	BITSTRING	1		14 Flags set by user
549	(225)	BITSTRING	3		15 Reserved
552	(228)	BITSTRING	4		18 Reserved
556	(22C)	SIGNED	4		1C Block size
560	(230)	BITSTRING	8		20 Reserved & number of blocks in ds
568	(238)	ADDRESS	4		28 End of data routine address or 0
572	(23C)	ADDRESS	4		2C I/O error routine (synchronous) or 0
576	(240)	BITSTRING	6		30 Reserved, should be zero
582	(246)	ADDRESS	1	(2)	36 MULTACC and MULTSDN

Comment

SHORTEST POSSIBLE DCBE IN ANY RELEASE.

End of Comment

582	(246)	X'98'	0	PRWDCBL	** -PRWDCB" DCB length
584	(248)	BITSTRING	1	PRWDCBFG	DCB EOF indicator X'FF' -> EOF
592	(250)	DBL WORD	8	(0)	Alignment
592	(250)	SIGNED	4	PRWOP (0)	ALIGN LIST TO FULLWORD

## \$PARMWRK Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
592	(250)	ADDRESS	1		OPTION BYTE
593	(251)	ADDRESS	3		DCB ADDRESS
593	(251)	X'250'	0	PRWOPEN	"PRWOP,*-PRWOP"
596	(254)	SIGNED	4	PRWCL (0)	ALIGN LIST TO FULLWORD
596	(254)	ADDRESS	1		OPTION BYTE
597	(255)	ADDRESS	3		DCB ADDRESS
597	(255)	X'254'	0	PRWCLOSE	"PRWCL,*-PRWCL"
600	(258)	SIGNED	4	PRWEXJFC (0)	Alignment
600	(258)	BITSTRING	1		JFCB
601	(259)	ADDRESS	3		exit
604	(25C)	SIGNED	4	PRWJFCB (0)	JFCB
604	(25C)	BITSTRING	1	(176)	work area
784	(310)	DBL WORD	8	(0)	Ensure alignment
784	(310)	X'310'	0	PRWLEN	"*-PRW" PRW length

## \$PARMWRK Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PRW	0		PRWTXTP2	124	
PRWATX	138		PRWTXTP3	128	
PRWATX1	138		PRWTXTP4	12C	
PRWATX2	146		PRWTXTP5	130	
PRWATX2D	14C		PRWTXTP6	134	
PRWATX3	154		PRWUNIT	3C	
PRWATX4	186		PRWUTX1	138	
PRWATX5	194		PRWUTX1D	13E	
PRWATX6	1A2		PRWUTX2	146	
PRWCL	254		PRWUTX2D	14C	
PRWCLOSE	255	254	PRWVOL	44	
PRWDCB	1B0		PRW1DCON	D8	80
PRWDCBE	210	C4C3C2C5	PRW1DECN	D8	40
PRWDCBFG	248		PRW2IDSN	D9	80
PRWDCBL	246	98	PRW2ILPM	D9	8
PRWDDNM	13E		PRW2IMEM	D9	10
PRWDDSN	15A		PRW2INST	D9	2
PRWDMEMB	18C		PRW2ISEQ	D9	4
PRWDSN	4A		PRW2IUNT	D9	20
PRWDSNEN	3A		PRW2IVOL	D9	40
PRWDSNMB	4				
PRWDUNIT	19A				
PRWDVOL	1A8				
PRWEXJFC	258				
PRWFLAG1	D8				
PRWFLAG2	D9				
PRWID	0	D7D9E640			
PRWIMEMB	BB				
PRWINIDS	E0				
PRWJFCB	25C				
PRWLDSB	CC				
PRWLEN	310	310			
PRWLMEMB	C3				
PRWLRCNT	D4				
PRWMEMB	84				
PRWOP	250				
PRWOPEN	251	250			
PRWPRW	D0				
PRWRB	E8				
PRWRBPTR	E4				
PRWRBX	FC				
PRWSEND	CC				
PRWSLEN	CC	CC			
PRWTXTP	120				
PRWTXTP1	120				

## \$PBLK Information

### \$PBLK Heading Information

**Common Name:** HAM Protected Block DSECT  
**Macro ID:** \$PBLK  
**DSECT Name:** PBLK  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** PBLK  
 Offset: PBLKID-PBLK  
 Length: L'PBLKID

**Storage Attributes:** Subpool: 229  
 Key: 1  
 Residency: Virtual and Real storage are limited to 31 bit because to the channel programs generated to read data into these blocks.

**Size:** See BATPBKSZ of the BAT that owns this PBLOCK

**Created by:** HASCHAM GET processing

**Pointed to by:** SDBPBLIN field of the SDB data area  
 SDBPBLAC field of the SDB data area  
 SDBPBLFL field of the SDB data area  
 SDBAPBL field of the SDB data area  
 PBLNEXT field of the PBL data area

**Serialization:** Local lock held by EXCPVR serializes most updates to this data area. In some cases, the SDB lock of the owning SDB is used for serialization.

**Function:** A PBLOCK contains control information, CCWs and data buffers used to read JES2 data sets from SPOOL. The size is dependant on the exact despooling method in use. Normal despooling uses 1 page of storage. Track cell despooling uses track cell (CCTTKCEL) number of 4K pages. Full track despooling uses 11 or 12 pages depending on buffer size (1944 vs 3992 bytes).

### \$PBLK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PBLK	, Protected block DSECT
0	(0)	CHARACTER	4	PBLKID	Eyecatcher
4	(4)	SIGNED	4	PBLSIZE	Size of data area
8	(8)	ADDRESS	4	PBLNEXT	Next PBLOCK on SDB chain
12	(C)	SIGNED	4	PBLUSECT	Use count for buffers
16	(10)	BITSTRING	1	PBLGFLG1	PBLOCK flag bytes
		1... ....		PBLGF1EL	"B'10000000" At least one end record in this PBLOCK
17	(11)	SIGNED	1	PBLSTRBF	First MTTRE to process
18	(12)	ADDRESS	2	PBLBUFNM	Number of buffers in block
20	(14)	ADDRESS	4	PBLCCWA	Address of CCWs
24	(18)	ADDRESS	4	PBLBUFA	Address of first buffer
28	(1C)	SIGNED	4	PBLECB	ECB to post when I/O completes
32	(20)	DBL WORD	8	PBLMTTTR (0)	Start of MTTTR/ADDRESS table

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PBLMTTRE	, MTTTR/ADDRESS table mapping
0	(0)	SIGNED	4	PBLMTTR	MTTTR of address (0 if not valid)
4	(4)	BITSTRING	1	PBLMFLG1	General flag byte

## \$PBLK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		1... ....		PBLMF1ER	"B'10000000" I/O error on buffer
		.1.. ....		PBLMF1VL	"B'01000000" Logical error on buffer
		..1. ....		PBLMF1OK	"B'00100000" Buffer verified OK
		...1 ....		PBLMF1PR	"B'00010000" Buffer processed (copied to UBUF)
		.... 1...		PBLMF1IP	"B'00001000" Locate mode partly assigned buffer
5	(5)	BITSTRING	3		Reserved
8	(8)	ADDRESS	4	PBLMADDR	Address of buffer for MTTR
12	(C)	ADDRESS	4	PBLMIDA1	IDAWs
16	(10)	ADDRESS	4	PBLMIDA2	to read
20	(14)	ADDRESS	4	PBLMIDA3	records

Comment

-----  
 If we are processing this PBLOCK in locate mode (FSS), then  
 the IDAWs are used as work areas.  
 -----

End of Comment

12	(C)	ADDRESS	4	PBLMLOC	Next LRC to process
16	(10)	BITSTRING	8	PBLMRBA	RBA of current LRC
24	(18)	DBL WORD	8	PBLMTTRN (0)	Next entry (DWORD aligned)
24	(18)	X'18'	0	PBLMELEN	** -PBLMTTRE" Length of an entry

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PBLCCWS	, CCW and data areas

Comment

-----  
 CKD Data area and CCWs  
 -----

End of Comment

0	(0)	BITSTRING	6	PBLDADDR (0)	(00CCHH) Seek address (BBCCHH)
0	(0)	BITSTRING	2	PBLDBB	BB value (always zero)
2	(2)	BITSTRING	5	PBLDCHR (0)	CCHHR value
2	(2)	BITSTRING	4		CCHH calue
6	(6)	BITSTRING	1	PBLDREC	R value
7	(7)	BITSTRING	1	PBLDSECT	Sector number
8	(8)	DBL WORD	8	PBLCCWS1 (0)	CKD CCW chain
32	(20)	X'20'	0	PBLCKDLN	** -PBLCCWS1"

Comment

-----  
 ECKD Data area and CCWs  
 -----

End of Comment

8	(8)	BITSTRING	16	PBLDIPRM (0)	Locate record parameter list
8	(8)	BITSTRING	1	PBLDOPER	Operation byte
9	(9)	BITSTRING	1	PBLDAUX	Auxiliary byte
10	(A)	BITSTRING	1		Reserved (must be 0)
11	(B)	BITSTRING	1	PBLDNREC	Number of records to process
12	(C)	BITSTRING	4	PBLDCCHH	(CCHH) Seek address (CCHH)
16	(10)	BITSTRING	5	PBLDCHR1 (0)	(CCHHR) Search address
16	(10)	BITSTRING	4	PBLDCH1	(CCHH) Cylinder and head numbers
20	(14)	BITSTRING	1	PBLDREC1	(R) Record number
21	(15)	BITSTRING	1	PBLDSC1	Sector number
22	(16)	BITSTRING	2	PBLDTLEN	Transfer length factor

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
32	(20)	DBL WORD	8	PBLCCWS2 (0)	ECKD CCW chain
Comment					
<p>Length fields. PBLSIZEC is the constant portion of the PBLOCK (base section plus positioning CCWs). PBLSIZEV is the length per input buffer for control areas and 1 read CCW.</p>					
End of Comment					
40	(28)	X'48'	0	PBLSIZEC	"(PBLMTRT-PBLK)+(PBLCRW-PBLCCWS)"
40	(28)	X'20'	0	PBLSIZEV	"(PBLMTRN-PBLMTTRE)+L'PBLCRW"

**\$PBLK Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PBLBUFA	18		PBLSIZEV	28	20
PBLBUFNM	12	0	PBLSTRBF	11	0
PBLCCWA	14		PBLUSECT	C	0
PBLCCWS	0				
PBLCCWS1	8				
PBLCCWS2	20				
PBLCKDLN	20	20			
PBLDADDR	0				
PBLDAUX	9				
PBLDBB	0				
PBLDCCHH	C				
PBLDCHR	2				
PBLDCHR1	10				
PBLDCH1	10				
PBLDIPRM	8				
PBLDNREC	B				
PBLDOPER	8				
PBLDREC	6				
PBLDREC1	14				
PBLDSCT1	15				
PBLDSECT	7				
PBLDTLEN	16				
PBLECB	1C	0			
PBLGFLG1	10	0			
PBLGF1EL	10	80			
PBLK	0				
PBLKID	0	D7C2D3D2			
PBLMADDR	8				
PBLMELEN	18	18			
PBLMFLG1	4				
PBLMF1ER	4	80			
PBLMF1IP	4	8			
PBLMF1OK	4	20			
PBLMF1PR	4	10			
PBLMF1VL	4	40			
PBLMIDA1	C				
PBLMIDA2	10				
PBLMIDA3	14				
PBLMLOC	C				
PBLMRBA	10				
PBLMTTR	0				
PBLMTTRE	0				
PBLMTTRN	18				
PBLMTTRT	20				
PBLNEXT	8				
PBLSIZE	4	0			
PBLSIZEC	28	48			

## \$PBLK Cross Reference

---

## \$PCE Information

### \$PCE Programming Interface information

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

#### \$PCE

The following field is **NOT** programming interface information:

- PCEPRE

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information

### \$PCE Heading Information

**Common Name:** JES2 Processor Control Element DSECT  
**Macro ID:** \$PCE  
**DSECT Name:** PCE  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
Offset: PCEEYE-PCE  
Length: 4

**Storage Attributes:** Subpool: 25  
Key: 1  
Residency: Virtual and real storage are anywhere, above or below 16M, in private storage of the JES2 address space.

**Size:** The length of a PCE is the length of the base PCE (defined by the expression, PCEWORK-PCE) plus the length of a variable length work area beginning at symbol PCEWORK.  
The length of the work area depends on the type of PCE. These work areas and their lengths are defined in separate mapping macros and are extensions of the PCE DSECT. See the definitions for PCEID byte 2 in this macro (\$PCE) for the names of the work area mapping macros.  
The overall length of the PCE is stored in field PCELENG.

**Created by:** The \$PCEDYN service. Most PCEs are created during JES2 initialization, others are created as needed (for example, PCEs for remote devices are created when a remote starts).



**Pointed to by:**

The \$PCEORG field of the \$HCT data area points to a chain containing all PCEs. This is a double threaded chain (see PCEPREV and PCENEXT below).

The \$PCELAST field of the \$HCT data area points to the last PCE on the \$PCEORG chain.

The PCENEXT and PCEPREV fields of the \$PCE data area points to the next/previous PCE on the \$PCEORG chain.

The \$CURPCE field of the \$HCT data area points to the PCE currently dispatched by the JES2 dispatcher. \$CURPCE is set to zero when a PCE gives up control with a \$WAIT.

The \$READY field of the \$HCT data area is the head of a circular queue of PCEs ready to be dispatched by the JES2 dispatcher. \$READYF (forward chain pointer) and \$READYL (backward chain pointer) are defined at the \$READY location. The PCEs on the queue are chained through the PCEPCEA (forward) and PCEPCEB (backward) fields. The queue head itself has a virtual origin in \$HCT at the offset defined by the expression \$READY-(PCEPCEA-PCE) so that the queue head is a dummy PCE called "PCE zero". When the ready queue is empty (that is, no PCEs are ready to be dispatched), the forward and backward pointers point to PCE zero. When the queue is not empty, \$READYF either points to a currently dispatched PCE or \$READYF points to the next PCE to be dispatched (\$CURPCE is zero).

The \$DRQUES field of the \$HCT data area points to the JES2 dispatcher resource wait queues, a table of double-word queue heads ordered by resource number. These queues are similar to the ready queue (above), e.g., a queue is empty when it points to PCE zero.

PCEPCEA and PCEPCEB fields of the \$PCE data area are used to chain PCEs on the ready queue or resource wait queues. A PCE is waiting for a specific \$POST when these fields point to the PCE itself.

DCTPCE field of the \$DCT data area.

XECBPCE field of the \$XECB data area.

In addition to the pointer fields described here, the PCE work area mapping macros describe additional pointers specific to the PCE type(s) of the work areas.

**Serialization:**

Normal PCE dispatch serialization

**Function:**

The Processor Control Element (PCE) represents an instance of a "process" running under the control of the JES2 main task. The JES2 main task runs under a single TCB that is sub-dispatched by the JES2 dispatcher. The JES2 dispatcher uses the PCE as its dispatchable unit.

## \$PCE Map

There are one or more PCEs for each JES2 processor type ID, as defined by the second byte of the PCEID field. Each of the ID types has a mapping macro that defines an extension to the PCE DSECT that begins at field PCEWORK. The names of the extension macros are given with the PCExxxID symbol definitions.

PCEs are related to JES2 devices in the following ways:

For non device related PCEs, PCEDCT is zero and no DCTPCE fields point to the PCE.

For a PCE that controls a single device, PCEDCT points to the Device Control Table (DCT) of the device the PCE manages.

For a PCE that controls multiple devices, PCEDCT is zero, but the DCTPCE field points to the PCE in each Device Control Table (DCT) that the PCE manages.

## \$PCE Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Processor Control Element
0	(0)	CHARACTER	4	PCEEYE (0)	Eye catcher
0	(0)	BITSTRING	1	(0)	\$SAVE area (see \$PSV)
0	(0)	X'4C'	0	PCELPSV	"PSVLABAD-PSV+PCE,4,C'A" Last \$SAVE area
0	(0)	X'99'	0	PCEXITID	"PSVEXID-PSV+PCE,1,C'F" Exit ID last invoked
0	(0)	X'48'	0	PCEDOM68	"PSVADDR-PSV+PCE,4,C'F" Domid for \$HASP068
168	(A8)	ADDRESS	4	PCEPREV	ADDRESS OF PREVIOUS PCE
172	(AC)	ADDRESS	4	PCENEXT	ADDRESS OF NEXT PCE
176	(B0)	ADDRESS	4	PCEPCEA	NEXT READY/WAITING PCE
180	(B4)	ADDRESS	4	PCEPCEB	PREVIOUS READY/WAITING PCE
184	(B8)	ADDRESS	4	PCEERA	ADDR OF ERA FOR ERROR FROM WHICH PCE IS ATTEMPTING TO RECOVER
188	(BC)	ADDRESS	4	PCEPRE	ADDRESS OF NEWEST PRE
192	(C0)	BITSTRING	1	PCEEWF	PROCESSOR EVENT WAIT FIELD
193	(C1)	BITSTRING	1	PCEFLAGS	PROCESSOR FLAGS
		1... ....		PCETRACE	"B'10000000" Processor eligible for tracing
		.1.. ....		PCEDSPXP	"B'01000000" Processor permanently exempt from non-dispatchability
		..1. ....		PCEDSPXT	"B'00100000" Processor temporarily exempt from non-dispatchability
		...1 ....		PCENWIOP	"B'00010000" Implicit \$WAITs in I/O processing should be prohibited (currently used only by \$IOERROR)
		.... 1...		PCETRPSF	"B'00001000" Short \$TRACE requested
		.... .1..		PCETRLDS	"B'00000100" Relds indicator
		.... ..1.		PCEPRIO	"B'00000010" High priority pce
		.... ...1		PCEREQIR	"B'00000001" PCE is required (terminate JES2 if PCE abends).
194	(C2)	BITSTRING	1	PCEFLAG2	More PCE flags
		1... ....		PCE2ENDD	"B'10000000" PCE has been terminated & will never be dispatched
		.1.. ....		PCE2EBUF	"B'01000000" Emergency buffers allowed
		..1. ....		PCE2EVNT	"B'00100000" An exception event has occurred for PCE this dispatcher cycle
		...1 ....		PCE2INIT	"B'00010000" PCE has initialized far enough to allow DETACH
		.... 1...		PCE2QSUS	"B'00001000" PCE needed CKPT queues last time it ran
		.... .1..		PCE2XWTR	"B'00000100" PCE is a PSO or SAPI PCE
		.... ..1.		PCE2SJID	"B'00000010" Maintain PCEJOBID
		.... ...1		PCE2SJIX	"B'00000001" Maintain PCEJQEIX
195	(C3)	BITSTRING	1	PCEFLGCS	PCE 'compare-and-swap' flag Flags in this byte may be manipulated by subtasks and therefore must use CS logic (OIL and NIL) to set/reset the bits.
		1... ....		PCEGEMOK	"B'10000000" PCE is allowed to obtain emergency CMBs

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
196	(C4)	.1.. .... SIGNED	2	PCECEMER PCENDSPC	"B'01000000" PCE currently owns an emergency CMB NON-DISPATCHABILITY COUNT - IF NON-ZERO PROCESSOR NOT DISPATCHED UNLESS EXEMPT VIA PCEDSPXP/XT
198	(C6)	SIGNED	2	PCEID	PROCESSOR TYPE
200	(C8)	SIGNED	4	PCEUSER0	RESERVED FOR USER
204	(CC)	SIGNED	4	PCEUSER1	RESERVED FOR USER
208	(D0)	SIGNED	4	PCEPOSTD (0)	PCE Post EWF fullword
208	(D0)	BITSTRING	3		Reserved for future use
211	(D3)	BITSTRING	1	PCEPSTEW	PCE POST EWF value
212	(D4)	ADDRESS	4	PCEPSTCH	PCE POST chain pointer
216	(D8)		16	PCEWTTIM	Time PCE \$WAITed (STCKE)

Comment

NOTE THAT THE FOLLOWING FIELDS (THROUGH PCEDEVTP) MUST  
CORRESPOND TO THE DCT FIELDS (THROUGH DCTDEVTP)

End of Comment

232	(E8)	DBL WORD	8	(0)	Ensure DWORD alignment
-----	------	----------	---	-----	------------------------

Comment

ORG -(DCTPCE-DCT) ESTABLISH THE PCEDADCT

End of Comment

224	(E0)	DBL WORD	8	PCEDADCT (0)	USING STORAGE FOR THE DCT FIELDS NOT IN A DA DCT
232	(E8)	ADDRESS	4	PCEDCTPC	DA DCT - DCTPCE
236	(EC)	SIGNED	4	PCEDCTFL	DCTSTAT-DCTFLAGS-DCTFLAG2
240	(F0)	BITSTRING	4	PCSEEEK	MTRR value for \$EXCP
244	(F4)	BITSTRING	4		Reserved (part of MQTR)
240	(F0)	DBL WORD	8	PCEMQTRD (0)	8 byte addr for STG
240	(F0)	BITSTRING	1	PCSEEEKF	'FF'x if MQTR is set
241	(F1)	BITSTRING	1		Reserved (part of MTRR)
242	(F2)	BITSTRING	6	PCEMQTR	MQTR value for \$EXCP
248	(F8)	ADDRESS	4	PCEBUFAD	BUFFER ADDRESS FOR \$EXCP
252	(FC)	ADDRESS	4	PCEIOEWF	PCE WITH EWF TO POST OR EXIT ADDRESS
256	(100)	SIGNED	2	PCEBUFCN	Active buffer count
258	(102)	BITSTRING	1	PCDEVTP	DA DCT FLAGS FOR \$EXCP
		.... ....		PCEDARD	"B'00000000" Direct access read request
		.... ...1		PCEDAWR	"B'00000001" Direct access write request
259	(103)	BITSTRING	1	PCEDAFL3	DA DCT flag byte (see DCTFLAG3 for bits)
259	(103)	X'1C'	0	PCEDALEN	"*-PCEDCTPC" Length of DA DCT in PCE
260	(104)	SIGNED	2	PCELENG	PCE LENGTH
262	(106)	ADDRESS	1	PCEROLOQ	Holding area for JQE/JOE 'prior' queue type
263	(107)	SIGNED	1	PCSESEQ	PCE sequence number
264	(108)	ADDRESS	4	PCEDCT	ADDRESS OF DCT (IF ANY)
268	(10C)	ADDRESS	4	PCEJQE	ADDRESS OF JQE (IF ANY)

Comment

-----  
 JOBID or JQE index of the current job. This is  
 maintained by \$DOGJQE for use by the monitor.  
 If the first 4 bytes of PCEJOBID is zero, then  
 either there is no current job or the second 4  
 bytes is a JQE index.  
 -----

End of Comment

272	(110)	CHARACTER	8	PCEJOBID	JOB ID of current job
272	(110)	SIGNED	4		Zero
276	(114)	SIGNED	4	PCEJQEIX	JOB index of current job
280	(118)	ADDRESS	4	PCEPTAB	Addr of PCETAB

# \$PCE Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
284	(11C)	ADDRESS	4	PCEFSACB	ADDRESS OF FSACB, IF ANY
288	(120)	ADDRESS	4	PCEWAVE	ADDRESS OF WAVE (IF ANY)
292	(124)	ADDRESS	4	PCENTITY	ADR OF ENTITY AREA (IF ANY)
296	(128)	SIGNED	2	PCEASID	Associated address space
298	(12A)	BITSTRING	1	PCEFLAG3	More PCE flags
		1... ....		PCE3HPFL	"B'10000000" PCE invoked HPUTFULL
299	(12B)	BITSTRING	1		Reserved for future use
300	(12C)	ADDRESS	4	PCEACTCT	PCE active count
304	(130)	SIGNED	4		Reserved for future use
308	(134)	ADDRESS	4	PCEWORKA	ADDRESS OF PCE WORK AREA
312	(138)	DBL WORD	8	PCEWORK (0)	VARIABLE LENGTH PROCESSOR WORK AREA

Comment

PCEID - BYTE1

End of Comment

.... ....	PCENODEV	"X'00" PCEID BYTE1 = 0 INDICATES NON- DEVICE PROCESSOR
.... ...1	PCELCLID	"X'01" LOCAL SPECIAL PCE ID
.... ..1.	PCERJEID	"X'02" REMOTE SPECIAL PCE ID
.... .1..	PCENJEID	"X'04" NETWORK SPECIAL PCE ID, INDICATES NJE OR XFR JT/JR/ST/SR
.... 1...	PCEINRID	"X'08" INTERNAL SPECIAL PCE ID
1... ....	PCEPRSID	"X'80" PRINT SPECIAL PCE ID
.1. ....	PCEPUSID	"X'40" PUNCH SPECIAL PCE ID
..1. ....	PCEXFRID	"X'20" XFR SPECIAL PCE ID

Comment

PCEID - BYTE2 (UNIQUE PCE ID) - HASP DEFINITIONS  
 HASP PCE IDS ARE ASSIGNED FROM 1 AND INCREASE. USER PCE IDS  
 PCE IDS SHOULD BE ASSIGNED FROM 255 AND DECREASE. EACH PCE  
 TYPE IS DEFINED IN THE HASP OR USER PCE TABLE, WITH POSSIBLY  
 MULTIPLE ENTRIES FOR EACH UNIQUE PCE ID (WITH DIFFERENT  
 VALUES FOR PCEID BYTE1).

Work  
 Area  
 Macro Descriptive name  
 -----

End of Comment

312	(138)	X'1'	0	PCERDRID	"1" \$RDRWORK - Input Services
312	(138)	X'2'	0	PCEASYID	"2" \$ASYWORK - Asynchronous I/O
312	(138)	X'3'	0	PCECNVID	"3" \$CNVWORK - Converter
312	(138)	X'4'	0	PCEXEQID	"4" \$XEQWORK - Execution
312	(138)	X'5'	0	PCEPSOID	"5" \$PSOWORK - Process SYSOUT
312	(138)	X'6'	0	PCEOUTID	"6" \$OUTWORK - Output
312	(138)	X'7'	0	PCEPRTID	"7" \$PPPWORK - Print \$FSSWORK - FSS Print Support
312	(138)	X'8'	0	PCEPUNID	"8" \$PPPWORK - Punch
312	(138)	X'9'	0	PCEPRGID	"9" \$PRGWORK - Purge
312	(138)	X'A'	0	PCECONID	"10" \$COMWORK - Command
312	(138)	X'B'	0	PCEMLMID	"11" \$MLMWORK - Multi-leaving Line Mgr
312	(138)	X'C'	0	PCETIMID	"12" \$TIMWORK - STIMER/TTIMER
312	(138)	X'D'	0	PCECKPID	"13" \$CKPWORK - Checkpoint
312	(138)	X'E'	0	PCEJPAID	"14" \$JPAWORK - Priority Aging
312	(138)	X'F'	0	PCEWRMID	"15" \$WARMWRK - Warm Start
312	(138)	X'10'	0	PCENJTID	"16" \$NJTWORK - NJE Job Transmitter
312	(138)	X'11'	0	PCENJRID	"17" \$RDRWORK - NJE Job Receiver
312	(138)	X'12'	0	PCENSTID	"18" \$NSTWORK - NJE SYSOUT Transmitter
312	(138)	X'13'	0	PCENSRID	"19" \$NSRWORK - NJE SYSOUT Receiver
312	(138)	X'14'	0	PCENPMID	"20" \$NPMWORK - NJE Path Manager
312	(138)	X'15'	0	PCERCPID	"21" \$RCPWORK - Remote Console
312	(138)	X'16'	0	PCETEXID	"22" \$TEXWORK - Time Excession Monitor

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
312	(138)	X'17'	0	PCEINTID	"23" \$CIRWORK - Initialization
312	(138)	X'18'	0	PCEVTLID	"24" \$TLGWORK - Event Trace Log
312	(138)	X'19'	0	PCEXFMID	"25" \$XFMWORK - SPOOL Transfer I/O Mgr
312	(138)	X'1A'	0	PCESPMID	"26" \$SPMWORK - SPOOL Manager
312	(138)	X'1B'	0	PCENRRID	"27" \$RDRWORK - NJE Route Receiver
312	(138)	X'1C'	0	PCENRTID	"28" \$NJTWORK - NJE Route Transmitter
312	(138)	X'1D'	0	PCSNFID	"29" \$SNFWORK - SPOOL sniffer
312	(138)	X'1E'	0	PCERESID	"30" \$RESWORK - Resource Manager
312	(138)	X'1F'	0	PCESTCID	"31" \$STCWORK - STATUS/CANCEL
312	(138)	X'20'	0	PCESPNID	"32" \$SPNWORK - Spin Services
312	(138)	X'21'	0	PCESFSID	"33" \$SFRWORK - Scheduler Services
312	(138)	X'22'	0	PCEOPAID	"34" \$OPAWORK - Output Priority Aging
312	(138)	X'23'	0	PCEFLID	"35" \$FCLWORK - FSS Cleanup on EOM
312	(138)	X'24'	0	PCEXCFID	"36" \$XPWORK - Coupling
312	(138)	X'25'	0	PCEJCMID	"37" \$JCMWORK - Job Command Processor
312	(138)	X'26'	0	PCEARMID	"38" \$ARMWORK - ARM support processor
312	(138)	X'27'	0	PCEXCMID	"39" \$XCMWORK - XCF Command Processor
312	(138)	X'28'	0	PCESPIID	"40" \$SPIWORK - Sysout API Processor
312	(138)	X'29'	0	PCEDILID	"41" \$DILWORK - 'Do It Later' processor
312	(138)	X'2A'	0	PCEENFID	"42" \$ENFWORK - ENF LISTEN processor
312	(138)	X'2B'	0	PCEALIID	"43" \$ALIWORK - Acquire lock & cleanup
312	(138)	X'2C'	0	PCEMSCID	"44" \$MSCWORK - Miscellaneous processor
312	(138)	X'2D'	0	PCEEOMID	"45" \$EOMWORK - End-of-Memory processor
312	(138)	X'2E'	0	PCEJQRID	"46" \$JQRWORK - JQE Request processor
312	(138)	X'2F'	0	PCEIRCID	"47" \$IRCWORK - Internal reader cleanup
312	(138)	X'30'	0	PCEDWNID	"48" \$DAWNWRK - DAWN processor
312	(138)	X'31'	0	PCENRMID	"49" \$NRMWORK - NJE Resource Monitor
312	(138)	X'32'	0	PCECDCID	"50" \$CDCWORK - Cross-system Device
312	(138)	X'33'	0	PCETPSID	"51" \$TIPSWRK - TIPS processor
312	(138)	X'34'	0	PCEJOIID	"52" \$JOEIWRK - JOEI processor

\$PCE Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCE	0		PCEERA	B8	
PCEACTCT	12C		PCEEWF	C0	
PCEALIID	138	2B	PCEEYE	0	
PCEARMID	138	26	PCEFLID	138	23
PCEASID	128		PCEFLAGS	C1	
PCEASYID	138	2	PCEFLAG2	C2	
PCEBUFAD	F8		PCEFLAG3	12A	
PCEBUFCN	100		PCEFLGCS	C3	
PCECDCID	138	32	PCEFSACB	11C	
PCECEMER	C3	40	PCEGEMOK	C3	80
PCECKPID	138	D	PCEID	C6	
PCECNVID	138	3	PCEINRID	138	8
PCECONID	138	A	PCEINTID	138	17
PCEDADCT	E0		PCEIOEWF	FC	
PCEDAFL3	103		PCEIRCID	138	2F
PCEDALEN	103	1C	PCEJCMID	138	25
PCEDARD	102	0	PCEJOBID	110	
PCEDAWR	102	1	PCEJOIID	138	34
PCEDCT	108		PCEJPAID	138	E
PCEDCTFL	EC		PCEJQE	10C	
PCEDCTPC	E8		PCEJQEIX	114	
PCEDEVTP	102		PCEJQRID	138	2E
PCEDILID	138	29	PCELCLID	138	1
PCEDOM68	0	48	PCELENG	104	
PCEDSPXP	C1	40	PCELPSV	0	4C
PCEDSPXT	C1	20	PCEMLMID	138	B
PCEDWNID	138	30	PCEMQTR	F2	
PCEENFID	138	2A	PCEMQTRD	F0	
PCEEOMID	138	2D	PCEMSCID	138	2C

## \$PCE Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCENDSPC	C4		PCEXITID	0	99
PCENEXT	AC		PCE2EBUF	C2	40
PCENJEID	138	4	PCE2ENDD	C2	80
PCENJRID	138	11	PCE2EVNT	C2	20
PCENJTID	138	10	PCE2INIT	C2	10
PCENODEV	138	0	PCE2QSUS	C2	8
PCENPMID	138	14	PCE2SJID	C2	2
PCENRMID	138	31	PCE2SJIX	C2	1
PCENRRID	138	1B	PCE2XWTR	C2	4
PCENRTID	138	1C	PCE3HPFL	12A	80
PCENSRID	138	13			
PCENSTID	138	12			
PCENTITY	124				
PCENWIOP	C1	10			
PCEOPAID	138	22			
PCEOUTID	138	6			
PCEPCEA	B0				
PCEPCEB	B4				
PCEPOSTD	D0				
PCEPRE	BC				
PCEPREV	A8				
PCEPRGID	138	9			
PCEPRIO	C1	2			
PCEPRSID	138	80			
PCEPRTID	138	7			
PCEPSOID	138	5			
PCEPSTCH	D4				
PCEPSTEW	D3				
PCEPTAB	118				
PCEPUNID	138	8			
PCEPUSID	138	40			
PCERCPID	138	15			
PCERDRID	138	1			
PCEREQIR	C1	1			
PCERESID	138	1E			
PCERJEID	138	2			
PCEROLOQ	106				
PCESEEK	F0				
PCESEEFK	F0				
PCESQ	107				
PCESFSID	138	21			
PCESNFID	138	1D			
PCESPIID	138	28			
PCESPMID	138	1A			
PCESPNID	138	20			
PCESTCID	138	1F			
PCETEXID	138	16			
PCETIMID	138	C			
PCETPSID	138	33			
PCETRACE	C1	80			
PCETRLDS	C1	4			
PCETRPSF	C1	8			
PCEUSER0	C8				
PCEUSER1	CC				
PCEVTLID	138	18			
PCEWAVE	120				
PCEWORK	138				
PCEWORKA	134				
PCEWRMID	138	F			
PCEWTTIM	D8				
PCEXCFID	138	24			
PCEXCMID	138	27			
PCEXEQID	138	4			
PCEXF MID	138	19			
PCEXFRID	138	20			

## \$PCL Information

### \$PCL Heading Information

**Common Name:** Persistent Connection Line element  
**Macro ID:** \$PCL  
**DSECT Name:** PCL  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCL '  
 Offset: PCLID-PCL  
 Length: 4  
**Storage Attributes:** Subpool: n/a  
 Key: 1  
 Residency: In the jesxPCL data space in cpool PCL  
**Size:** See PCLLEN  
**Created by:** JES2 line manager  
**Pointed to by:** MDCTPCL field of the \$DCT data area  
 NSCSPCL field of the \$NSCT data area  
 NSSLPCL field of the \$NSST data area  
 PCTPCLAQ field of the \$PCT data area  
 PCTPCLSQ field of the \$PCT data area  
 PCLPCL field of the \$PCL data area  
 PCLPCLSV field of the \$PCL data area  
 PCLNEXT field of the \$PCL data area  
 PCLPREV field of the \$PCL data area  
 PCLSNEXT field of the \$PCL data area  
 PCLSPREV field of the \$PCL data area  
 TBFPCCL field of the \$TBUF data area  
 TBFLNPCL field of the \$TBUF data area  
**Serialization:** Most fields require only JES2 main task serialization.  
 Chaining fields generally are serialized by the FIFOENQ service.  
**Function:** Contains parameters for a line or NETSRV device which are shared between the JES2 address space and the NJE server address space.

### \$PCL Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCL	
0	(0)	CHARACTER	4	PCLID	PCL eyecatcher
4	(4)	ADDRESS	1	PCLVER	PCL version number
4	(4)	X'1'	0	PCLVERN	"1" PCL version
5	(5)	BITSTRING	1	PCLDTYPE	PCL Device type
5	(5)	X'1'	0	PCLDTLNE	"1" PCL associated with Line
5	(5)	X'2'	0	PCLDTSRV	"2" PCL associated with Server
6	(6)	BITSTRING	1	PCLTTYPE	PCL Connection type
6	(6)	X'1'	0	PCLTTCP	"1" TCP/IP connection
8	(8)	ADDRESS	4	PCLNEXT	Address of next PCL
12	(C)	ADDRESS	4	PCLPREV	Address of prior PCL

## \$PCL Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
-----					
The node definitions below define the local node (for server PCLs) or the adjacent node (for line PCLs)					
-----					
End of Comment					
16	(10)	CHARACTER	8	PCLNNAME	Node name
24	(18)	SIGNED	2	PCLNNUM	Node number
26	(1A)	BITSTRING	2		Reserved
28	(1C)	ADDRESS	4	PCLDCT	DCT address (JES2 Private)
32	(20)	SIGNED	2	PCLDNUM	Device number of DCT
34	(22)	BITSTRING	1	PCLFLAG1	Common PCL Flags
		1... ....		PCL1TRAC	"B'10000000" JES tracing active
		.1... ....		PCL1CTRC	"B'01000000" NETSRV tracing active
		..1. ....		PCL1VERB	"B'00100000" Verbose diagnostics
35	(23)	BITSTRING	1		Reserved
Comment					
-----					
Outbound buffers are queued here. These buffers are in the jesxTBUF data space. The buffers are added to the tail of the queue and removed from the head atomically via the PLO instruction.					
-----					
End of Comment					
36	(24)	ADDRESS	4	PCLOBUFH	Outbound buffer queue head
40	(28)	ADDRESS	4	PCLOBUFT	Outbound buffer queue tail
Comment					
-----					
Inbound buffers are queued here. These buffers are in the jesxTBUF data space. The buffers are added to the tail of the queue and removed from the head atomically via the PLO instruction.					
-----					
End of Comment					
44	(2C)	ADDRESS	4	PCLIBUFH	Inbound buffer queue head
48	(30)	ADDRESS	4	PCLIBUFT	Inbound buffer queue tail
52	(34)	SIGNED	4	PCLICNT	Inbound buffer queue count
56	(38)	DBL WORD	8	PCLEXORG (0)	Origins for PCL extension
Comment					
-----					
Extension for LINE type PCLs					
-----					
End of Comment					
56	(38)	BITSTRING	2	PCLREST	Connection resistance
58	(3A)	BITSTRING	1	PCLQUAL	Member # of adjacent node
60	(3C)	ADDRESS	4	PCLSNEXT	Next line on server chain
64	(40)	ADDRESS	4	PCLSPREV	Prior line on server chain
68	(44)	ADDRESS	4	PCLPCLSV	Pointer to server PCL
72	(48)	BITSTRING	1	PCLLFLG1	Flags
		1... ....		PCLL1DED	"B'10000000" A dedicated packet of subdevices is associated with this line
		.1... ....		PCLL1DND	"B'01000000" Line is dedicated to node
		..1. ....		PCLL1DSC	"B'00100000" Line is dedicated to socket



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		...1 ....		PCLL1CON	"B'00010000" This line is connected
		.... 1...		PCLL1NET	"B'00001000" Networking has started
		.... .1..		PCLL1DPR	"B'00000100" Default port used
		.... ..1.		PCLL1KIL	"B'00000010" Disconnect line
		.... ...1		PCLL1STR	"B'00000001" Subtask requested
73	(49)	BITSTRING	1	PCLLFLG2	Flags
		1... ....		PCLL2SSL	"B'10000000" Secure socket used
		.1.. ....		PCLL2NTY	"B'01000000" Non-retryable error
74	(4A)	BITSTRING	2		Reserved
76	(4C)	ADDRESS	4	PCLNLDV (0)	Subdevice counts
76	(4C)	ADDRESS	1	PCLJTNM	Number of job xmitters
77	(4D)	ADDRESS	1	PCLJRNM	Number of job receivers
78	(4E)	ADDRESS	1	PCLSTNM	Number of SYSOUT xmitters
79	(4F)	ADDRESS	1	PCLSRNM	Number of sysout receivers

Comment

-----  
Data specific to each subdevice (line counts, etc)  
-----

End of Comment

80	(50)	DBL WORD	8	PCLLELM (0)	Force alignment
80	(50)	BITSTRING	112	PCLJTELM (7)	Data for 7 job xmitters
192	(C0)	BITSTRING	112	PCLJRELM (7)	Data for 7 job receivers
304	(130)	BITSTRING	168	PCLSTELM (7)	Data for 7 SYSOUT xmitters
472	(1D8)	BITSTRING	1	PCLSRELM (7)	Data for 7 SYSOUT receivers
472	(1D8)	X'1F8'	0	PCLLEML	"*-PCLLELM" Composite area length
584	(248)	DBL WORD	8	PCLDATA (0)	Variable data, based on PCLTYPE

Comment

-----  
TCP/IP Line specific data  
-----

End of Comment

584	(248)	ADDRESS	4	PCLNSST	Address of NSST (in netsrv address space)
588	(24C)	CHARACTER	8	PCLLSCKN	Name of associated SCK
596	(254)	CHARACTER	255	PCLLIPNM	IP address (EBCDIC)
851	(353)	CHARACTER	1		Reserved
852	(354)	BITSTRING	16	PCLLIPAD	IP address (binary)
868	(364)	CHARACTER	16	PCLLPRTN	Port name (EBCDIC)
884	(374)	SIGNED	2	PCLLPORT	Port number (binary)
886	(376)	SIGNED	2	PCLLOPRT	Original port # from socket
888	(378)	BITSTRING	4	PCLLFEAT	Signon feature flags
892	(37C)	SIGNED	4	PCLLSKID	Socket id (assigned by IAZNJSTK)
896	(380)	BITSTRING	16	PCLLSTTT	NETSRV subtask TCB token
912	(390)	SIGNED	4	(4)	Reserved for future use
928	(3A0)	DBL WORD	8	(0)	

Comment

Extension for SERVER type PCLs

End of Comment

56	(38)	CHARACTER	8	PCLPGM	Program to start
64	(40)	CHARACTER	8	PCLPROC	Proc to use
72	(48)	CHARACTER	8	PCLNAME	Address space name
80	(50)	CHARACTER	8	PCLSTACK	TCP/IP Stack to listen on
88	(58)	BITSTRING	8	PCLASCBT	Address Space Token
96	(60)	ADDRESS	4	PCLASCB	ASCB address
100	(64)	SIGNED	2	PCLASID	ASID
102	(66)	SIGNED	2	PCLSBFSZ	Maximum buffer size

## \$PCL Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
104	(68)	BITSTRING	4	PCLSFEAT	Signon feature flags
108	(6C)	ADDRESS	2	PCLKEEP1	Keepalive interval
110	(6E)	BITSTRING	2		Reserved for future use
-----					
Comment					
-----					
Queue heads for line DCTs running under this server					
-----					
End of Comment					
112	(70)	ADDRESS	4	PCLLPCLH	Line PCL queue head
116	(74)	ADDRESS	4	PCLLPCLT	Line PCL queue tail
120	(78)	DBL WORD	8	PCLSDATA (0)	Server dependent data
-----					
Comment					
-----					
TCP/IP Server specific data					
-----					
End of Comment					
120	(78)	ADDRESS	4	PCLTCT	Address of IAZYTCT (in netsrv address space)
124	(7C)	ADDRESS	4	PCLNCPE	Address of post element
128	(80)	CHARACTER	8	PCLSSCKN	Name of associated SCK
136	(88)	CHARACTER	255	PCLSIPNM	Server IP address (EBCDIC)
391	(187)	CHARACTER	1		Reserved
392	(188)	BITSTRING	16	PCLSIPAD	Server IP address (binary)
408	(198)	CHARACTER	16	PCLSPRTN	Server Port name (EBCDIC)
424	(1A8)	SIGNED	2	PCLSPORT	Server Port number (binary)
426	(1AA)	BITSTRING	1	PCLSFLG1	Server Flags
		1... ....		PCLS1SSL	"B'10000000" Secure socket
		.1.. ....		PCLS1DIP	"B'01000000" Default PORT value used
		..1. ....		PCLS1DPR	"B'00100000" Default IP address used
		...1 ....		PCLS1DRN	"B'00010000" \$P NETSRV issued
432	(1B0)	DBL WORD	8	(0)	End of type dependent data
928	(3A0)	X'3A0'	0	PCLLEN	** -PCL "

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCLJT	Job Transmitter data
0	(0)	SIGNED	4	PCLJTJID	Job # of active job
4	(4)	BITSTRING	1	PCLJTFG1	Status flags
		1... ....		PCLJTF1C	"B'10000000" Job complete
		.1.. ....		PCLJTF1R	"B'01000000" Restart job
		..1. ....		PCLJTF1H	"B'00100000" Hold job
		...1 ....		PCLJTF1D	"B'00010000" Drain transmitter
		.... 1..		PCLJTF1S	"B'00001000" Start transmitter
		.... .1..		PCLJTF1E	"B'00000100" \$C/\$E of xmitter issued
		.... ..1.		PCLJTF1F	"B'00000010" EOF sent for current job
		.... ...1		PCLJTF1W	"B'00000001" Wake up xmitter PCE
5	(5)	BITSTRING	3		Reserved for future use
8	(8)	SIGNED	4	PCLJTRCT	Records sent so far
12	(C)	SIGNED	4	PCLJTRTO	Total records in job stream
12	(C)	X'10'	0	PCLJTLEN	** -PCLJT" Size

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCLJR	Job Receiver data
0	(0)	SIGNED	4	PCLJRJID	Job # of active job
4	(4)	BITSTRING	1	PCLJRFG1	Status flags

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		1... ....		PCLJRF1D	"B'10000000" Receiver is drained
		.1.. ....		PCLJRF1C	"B'01000000" \$C of receiver issued
		..1. ....		PCLJRF1E	"B'00100000" \$E of receiver issued
		...1 ....		PCLJRF1F	"B'00010000" EOF received current job
5	(5)	BITSTRING	3		Reserved for future use
8	(8)	SIGNED	4	PCLJRRCT	Records received so far
12	(C)	SIGNED	4	PCLJRRTO	Total records in job stream
12	(C)	X'10'	0	PCLJRLen	**-"PCLJR" Size

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCLST	SYSOUT Transmitter data
0	(0)	SIGNED	4	PCLSTJID	Job # of active job
4	(4)	SIGNED	4	PCLSTNCH	Offset of first JOE
8	(8)	BITSTRING	1	PCLSTFG1	Status flags
		1... ....		PCLSTF1C	"B'10000000" Job complete
		.1.. ....		PCLSTF1R	"B'01000000" Restart job
		..1. ....		PCLSTF1H	"B'00100000" Hold job
		...1 ....		PCLSTF1D	"B'00010000" Drain transmitter
		.... 1..		PCLSTF1S	"B'00001000" Start transmitter
		.... .1..		PCLSTF1E	"B'00000100" \$C/\$E of xmitter issued
		.... .1.		PCLSTF1F	"B'00000010" EOF sent for current job
		.... ...1		PCLSTF1W	"B'00000001" Wake up xmitter PCE
9	(9)	BITSTRING	1	PCLSTFG2	Status flags - Flags to be serialized by OIL & NIL
		1... ....		PCLSTF2O	"B'10000000" Hold Joes
10	(A)	BITSTRING	2		Reserved for future use
12	(C)	SIGNED	4	PCLSTRCT	Records sent so far
16	(10)	SIGNED	4	PCLSTRTO	Total records in SYSOUT stream
20	(14)	SIGNED	4	PCLSTEXC	# of logical puts to line
20	(14)	X'18'	0	PCLSTLEN	**-"PCLST" Size

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCLSR	SYSOUT Receiver data
0	(0)	SIGNED	4	PCLSRJID	Job # of active job
4	(4)	BITSTRING	1	PCLSRFG1	Status flags
		1... ....		PCLSRF1D	"B'10000000" Receiver is drained
		.1.. ....		PCLSRF1C	"B'01000000" \$C of receiver issued
		..1. ....		PCLSRF1E	"B'00100000" \$E of receiver issued
		...1 ....		PCLSRF1F	"B'00010000" EOF received current job
5	(5)	BITSTRING	3		Reserved for future use
8	(8)	SIGNED	4	PCLSRRCT	Records received so far
12	(C)	SIGNED	4	PCLSRRTTO	Total records in SYSOUT stream
12	(C)	X'10'	0	PCLSRLEN	**-"PCLSR" Size

## \$PCL Cross Reference

### \$PCL Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCL	0		PCLL1DND	48	40
PCLASCB	60		PCLL1DPR	48	4
PCLASCBT	58		PCLL1DSC	48	20
PCLASID	64		PCLL1KIL	48	2
PCLDATA	248		PCLL1NET	48	8
PCLDCT	1C		PCLL1STR	48	1
PCLDNUM	20		PCLL2NTY	49	40
PCLDTLNE	5	1	PCLL2SSL	49	80
PCLDTSRV	5	2	PCLNAME	48	9185A2A7
PCLDTYPE	5		PCLNCPE	7C	
PCELEM	50		PCLNEXT	8	
PCELEML	1D8	1F8	PCLNLDV	4C	
PCEXORG	38		PCLNNAME	10	
PCLFLAG1	22		PCLNNUM	18	
PCLIBUFH	2C		PCLNSST	248	
PCLIBUFT	30		PCLOBUFH	24	
PCLICNT	34		PCLOBUFT	28	
PCLID	0	D7C3D340	PCLPCLSV	44	
PCLJR	0		PCLPGM	38	C9C1E9D5
PCLJRELM	C0		PCLPREV	C	
PCLJRFG1	4		PCLPROC	40	C9C5C5E2
PCLJRF1C	4	40	PCLQUAL	3A	
PCLJRF1D	4	80	PCLREST	38	
PCLJRF1E	4	20	PCLSBFSZ	66	
PCLJRF1F	4	10	PCLSDATA	78	
PCLJRJID	0		PCLSFEAT	68	
PCLJRLEN	C	10	PCLSFLG1	1AA	
PCLJRNM	4D		PCLSIPAD	188	
PCLJRRCT	8		PCLSIPNM	88	
PCLJRRTO	C		PCLSNEXT	3C	
PCLJT	0		PCLSPORT	1A8	
PCLJTELM	50		PCLSPREV	40	
PCLJTFG1	4		PCLSPRTN	198	
PCLJTF1C	4	80	PCLSR	0	
PCLJTF1D	4	10	PCLSRELM	1D8	
PCLJTF1E	4	4	PCLSRFG1	4	
PCLJTF1F	4	2	PCLSRF1C	4	40
PCLJTF1H	4	20	PCLSRF1D	4	80
PCLJTF1R	4	40	PCLSRF1E	4	20
PCLJTF1S	4	8	PCLSRF1F	4	10
PCLJTF1W	4	1	PCLSRJID	0	
PCLJTJID	0		PCLSRLN	C	10
PCLJTLEN	C	10	PCLSRNM	4F	
PCLJTNM	4C		PCLSRRCT	8	
PCLJTRCT	8		PCLSRRTO	C	
PCLJTRTO	C		PCLSSCKN	80	
PCLKEEPI	6C		PCLST	0	
PCLLEN	3A0	3A0	PCLSTACK	50	40404040
PCLLFEAT	378		PCLSTEML	130	
PCLLFLG1	48		PCLSTEXC	14	
PCLLFLG2	49		PCLSTFG1	8	
PCLLIPAD	354		PCLSTFG2	9	
PCLLIPNM	254		PCLSTF1C	8	80
PCLLOPRT	376		PCLSTF1D	8	10
PCLLPCLH	70		PCLSTF1E	8	4
PCLLPCLT	74		PCLSTF1F	8	2
PCLLPORT	374		PCLSTF1H	8	20
PCLLPRTN	364		PCLSTF1R	8	40
PCLLSCKN	24C		PCLSTF1S	8	8
PCLLSKID	37C		PCLSTF1W	8	1
PCLLSTTT	380		PCLSTF2O	9	80
PCLL1CON	48	10	PCLSTJID	0	
PCLL1DED	48	80	PCLSTLEN	14	18

Name	Hex Offset	Hex Value
PCLSTNCH	4	
PCLSTNM	4E	
PCLSTRCT	C	
PCLSTRTO	10	
PCLS1DIP	1AA	40
PCLS1DPR	1AA	20
PCLS1DRN	1AA	10
PCLS1SSL	1AA	80
PCLTCT	78	
PCLTTCP	6	1
PCLTTYPE	6	
PCLVER	4	
PCLVERN	4	1
PCL1CTRC	22	40
PCL1TRAC	22	80
PCL1VERB	22	20

## \$PCL Cross Reference

## \$PCT Information

### \$PCT Programming Interface information

Programming Interface information

#### \$PCT

The following fields are **NOT** programming interface information:

- PCTPCLAH
- PCTPCLAT
- PCTPCLSQ
- PCTPCLST
- PCTPCLAQ
- PCTPCLSH

End of Programming Interface information

## Heading Information • \$PCT Map

### \$PCT Heading Information

**Common Name:** Path Manager Control Table  
**Macro ID:** \$PCT  
**DSECT Name:** PCT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCT '  
 Offset: PCTID-PCT  
 Length: 4  
**Storage Attributes:** Subpool: 241  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M) in CSA  
**Size:** See PCTLEN  
**Created by:** Routine IRSSI during JES2 initialization  
**Pointed to by:** CCTPCT field of the \$HCCT data area  
**Serialization:** Most fields require only JES2 main task serialization. However, some fields also require serialization with a general purpose subtask if it is possible to affect the field while a "full path" analysis is in progress.  
**Function:** Contains the main parameters for, and anchors the work queues for, the JES2 network path manager.

### \$PCT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCT	
0	(0)	CHARACTER	4	PCTID	PCT eyecatcher
4	(4)	ADDRESS	1	PCTVER	PCT version number
4	(4)	X'3'	0	PCTVERN	"3" PCT version
5	(5)	BITSTRING	3		Reserved for future use
-----					
Comment					
-----					
NAT queue heads and other fields are maintained across a hot start					
-----					
End of Comment					
8	(8)	ADDRESS	4	PCTNATAH	Ptr to head of active NAT queue
12	(C)	ADDRESS	4	PCTNATAT	Ptr to tail of active NAT queue
16	(10)	ADDRESS	4	PCTNATUH	Ptr to head of unconnected NAT que
20	(14)	ADDRESS	4	PCTNATUT	Ptr to tail of unconnected NAT que
24	(18)	ADDRESS	4	PCTNATHH	Ptr to head of held NAT queue
28	(1C)	ADDRESS	4	PCTNATHT	Ptr to tail of held NAT queue
32	(20)	ADDRESS	4	PCTNATNH	Head of temp ACTIVE NAT que
36	(24)	ADDRESS	4	PCTNATNT	Tail of temp ACTIVE NAT que
36	(24)	X'8'	0	PCTNATQS	"PCTNATAH,*-PCTNATAH" All NAT queue heads/tails
40	(28)	SIGNED	4	PCTTTOL	TOD tolerance for connections
40	(28)	X'15180'	0	PCTTTOLD	"1440*60" Default TOD tolerance
44	(2C)	ADDRESS	4	PCTTINQ (0)	TCP/IP NPM buffer queue
44	(2C)	ADDRESS	4	PCTTINQH	TCP/IP NPM buffer q head
48	(30)	ADDRESS	4	PCTTINQT	TCP/IP NPM buffer q tail
52	(34)	ADDRESS	4	PCTTRCPQ (0)	TCP/IP RCP buffer chain
52	(34)	ADDRESS	4	PCTTRCPH	TCP/IP RCP buffer q head
56	(38)	ADDRESS	4	PCTTRCPT	TCP/IP RCP buffer q tail
60	(3C)	ADDRESS	4	PCTTMPIQ (0)	Temp inbound TCP buf q
60	(3C)	ADDRESS	4	PCTTMPIH	Temp inbound TCP buf q head
64	(40)	ADDRESS	4	PCTTMPIT	Temp inbound TCP buf q tail
68	(44)	SIGNED	4	PCTTBFCT	Count of allocated TBUFs



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
72	(48)	ADDRESS	4		
76	(4C)	BITSTRING	4	PCTNALET	ALET for NAT cell pool
80	(50)	BITSTRING	4	PCTPALET	ALET for PCL cell pool
84	(54)	BITSTRING	4	PCTTALET	ALET for TBUF cell pool
88	(58)	ADDRESS	4	PCTPCLAQ (0)	Active line PCL queue
88	(58)	ADDRESS	4	PCTPCLAH	Active line PCL head
92	(5C)	ADDRESS	4	PCTPCLAT	Active line PCL tail
96	(60)	ADDRESS	4	PCTPCLSQ (0)	Active Server PCL queue
96	(60)	ADDRESS	4	PCTPCLSH	Active Server PCL head
100	(64)	ADDRESS	4	PCTPCLST	Active Server PCL tail
104	(68)	CHARACTER	16	PCTMBNAM	NPM Mail box name

Comment

Line manager post flags. These flags correspond directly to the flags define in MLMSCNR1, MLMSCNR2, and MLMEVNTR and must be maintained in this order. OIL and NIL should be used to set the flags; Compare-and-swap is used in the line manager PCE to clear them.

End of Comment

120	(78)	SIGNED	4	PCTMLMSC (0)	MLM SCAN flag bits
120	(78)	BITSTRING	1	PCTMLMR1	Corresponds to MLMSCNR1
121	(79)	BITSTRING	1	PCTMLMR2	Corresponds to MLMSCNR2
122	(7A)	BITSTRING	1	PCTMLMER	Corresponds to MLMEVNTR
123	(7B)	BITSTRING	1		Fourth byte of PCTMLMSC
124	(7C)	SIGNED	2	PCTNPATH	Number of paths
126	(7E)	SIGNED	2	PCTMXHOP	Maximum NJE hop count
128	(80)	SIGNED	2	PCTANINT	Default NJE retry interval (minutes)
130	(82)	SIGNED	2		Reserved for future use
132	(84)	SIGNED	4	(3)	Reserved for future use

Comment

Pointers to private areas, work fields, etc. must be cleared on a hot start

End of Comment

132	(84)	X'90'	0	PCTHOTC	*** Start of area to clear on hot start
144	(90)	ADDRESS	1	PCTFLAG1	NPM process control flags
		1... ....		PCT1PATH	"B'10000000" Full path required
		.1.. ....		PCT1FPNP	"B'01000000" Full path in progress
		..1. ....		PCT1NTUP	"B'00100000" A NAT update has occurred
		...1 ....		PCT1NOT	"B'00010000" Notify required
		.... 1...		PCT1NERR	"B'00001000" NAT error detected
		.... .1..		PCT1NREC	"B'00000100" NRECEIVE in progress
		.... ..1.		PCT1DOWN	"B'00000010" NPM is down
145	(91)	ADDRESS	1	PCTFLAG2	Flags
		1... ....		PCT2NSUB	"B'10000000" NSETSUBS recovery is in progress
		.1.. ....		PCT2NSSS	"B'01000000" NETSRV SUBSYS switch
		.... ..1		PCT2NSIM	"B'00000001" NPMSIM flag (used for internal JES2 testing)
146	(92)	BITSTRING	6		Reserved for future use
152	(98)	DBL WORD	8	PCTFPTIM	Time last fullpath started
160	(A0)	ADDRESS	4	PCTPATHS	NPMNITPs used by full path
164	(A4)	ADDRESS	4	PCTWORKQ	Queue of nodes for full path to process
168	(A8)	ADDRESS	4	PCTNTQUQ	Unprocessed NTQ chain
172	(AC)	ADDRESS	4	PCTNTQPQ	Processed NTQ chain
176	(B0)	ADDRESS	4	PCTSONQ	Signon queue anchor
180	(B4)	ADDRESS	4	PCTRESPQ	Response queue anchor

## \$PCT Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
184	(B8)	ADDRESS	4	PCTACTL	Active net line DCTs anchor
188	(BC)	ADDRESS	4	PCTRSTL	DCT line restart queue head
192	(C0)	ADDRESS	4	PCTRSTN	DCT NJE restart queue head
196	(C4)	ADDRESS	4	PCTINQ	BSC input buffer queue anchor
200	(C8)	ADDRESS	4	PCTVINQ	VTAM input buffer queue anchor
204	(CC)	ADDRESS	4	PCTMASDN	MAS line drain queue
208	(D0)	ADDRESS	4	PCTPRPIQ	I/J across MAS pending DCT
212	(D4)	SIGNED	2	PCTLNENM	Total number of lines that can do NJE
214	(D6)	SIGNED	2	PCTMAPLN	Length of notify maps
216	(D8)	ADDRESS	4	PCTMINX	Master notify map anchor
220	(DC)	ADDRESS	4	PCTMINXM	MAS master notify map addr
224	(E0)	ADDRESS	4	PCTWINX	Work notify map anchor
228	(E4)	ADDRESS	4	PCTMAPQ	Queue of available notify maps
232	(E8)	ADDRESS	4	PCTNSAAQ	Active net subnet ct head

Comment

Addresses of MAS line DCTs for Nodal SPOOLing

End of Comment

236	(EC)	SIGNED	4	(0)	
236	(EC)	SIGNED	4	PCTDCT1 (0)	MAS line DCT address for members 1 through n
364	(16C)	SIGNED	4	PCTMTIME	Time of last NMAINT call
368	(170)	ADDRESS	4	PCTNTW	NPM \$TRACE work area

Comment

The following fields are used by routine NPDDMSG to build a symptom record.

End of Comment

372	(174)	CHARACTER	44	PCTNCC	Current NCC record being received and processed
416	(1A0)	ADDRESS	4	PCTNTQ	Address of current NTQ being processed
420	(1A4)	ADDRESS	4	PCTEDCT	Address of DCT that is related to the error
424	(1A8)	ADDRESS	4	PCTENIT	Address of NIT that is related to the error

Comment

The following fields are used by NPEVENT to set CES values for signons.

End of Comment

432	(1B0)	DBL WORD	8	PCTTOD	Time of day clock value
440	(1B8)	SIGNED	4	PCTEVENT	Current CES value

Comment

The following field contains NJE feature flags for features supported by this system

End of Comment

444	(1BC)	SIGNED	4	PCTIFEAT	NJE feature flags supported by JES2
448	(1C0)	SIGNED	4	PCTNSFEA	NJE feature flags owned by NETSRV, not JES2
452	(1C4)	SIGNED	4	PCTTFEAT	Feature flags to be turned off (test purposes only)
452	(1C4)	X'138'	0	PCTHOTCL	** -PCTHOTC" Length to be cleared
452	(1C4)	X'1C8'	0	PCTLEN	** -PCT" Length of the PCT

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>Input error reason codes from all the NAT service routines. These reason codes are provided here to ensure that they are consistent from routine to routine. The reason codes are returned in register 0 when an input error has been detected in a service routine. Note that the RETURN code for input errors varies from routine to routine, but the reason codes associated with that return code are the same for all routines.</p>					
End of Comment					
452	(1C4)	X'0'	0	PCT\$RC00	"0" A required control block address was not provided
452	(1C4)	X'4'	0	PCT\$RCN1	"4" The primary node in the prototype NAT was invalid, or omitted when required
452	(1C4)	X'8'	0	PCT\$RCM1	"8" The primary member in the prototype NAT was invalid, or omitted when required
452	(1C4)	X'C'	0	PCT\$RCN2	"12" The 2ndary node in the prototype NAT was invalid, or omitted when required
452	(1C4)	X'10'	0	PCT\$RCM2	"16" The 2ndary member in the prototype NAT was invalid, or omitted when required
452	(1C4)	X'14'	0	PCT\$RCRS	"20" The resistance specified in the prototype NAT was invalid
452	(1C4)	X'18'	0	PCT\$RCDP	"24" The primary and secondary node and member of the prototype were identical (\$NATADD only)
452	(1C4)	X'1C'	0	PCT\$RCST	"28" The status of the NAT was not one of ACTIVE, INACTIVE, or HELD, or was invalid for that NAT

**\$PCT Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCT	0		PCTMLMR1	78	
PCT\$RCDP	1C4	18	PCTMLMR2	79	
PCT\$RCM1	1C4	8	PCTMLMSC	78	
PCT\$RCM2	1C4	10	PCTMTIME	16C	
PCT\$RCN1	1C4	4	PCTMXHOP	7E	
PCT\$RCN2	1C4	C	PCTNALET	4C	
PCT\$RCRS	1C4	14	PCTNATAH	8	
PCT\$RCST	1C4	1C	PCTNATAT	C	
PCT\$RC00	1C4	0	PCTNATHH	18	
PCTACTL	B8		PCTNATHT	1C	
PCTANINT	80		PCTNATNH	20	
PCTDCT1	EC		PCTNATNT	24	
PCTEDCT	1A4		PCTNATQS	24	8
PCTENIT	1A8		PCTNATUH	10	
PCTEVENT	1B8		PCTNATUT	14	
PCTFLAG1	90		PCTNCC	174	
PCTFLAG2	91		PCTNPATH	7C	
PCTFPTIM	98		PCTNSAAQ	E8	
PCTHOTC	84	90	PCTNSFEA	1C0	
PCTHOTCL	1C4	138	PCTNTQ	1A0	
PCTID	0	D7C3E340	PCTNTQPQ	AC	
PCTIFEAT	1BC		PCTNTQUQ	A8	
PCTINQ	C4		PCTNTW	170	
PCTLEN	1C4	1C8	PCTPALET	50	
PCTLNENM	D4		PCTPATHS	A0	
PCTMAPLN	D6		PCTPCLAH	58	
PCTMAPQ	E4		PCTPCLAQ	58	
PCTMASDN	CC		PCTPCLAT	5C	
PCTMBNAM	68	E2E8E2D1	PCTPCLSH	60	
PCTMINX	D8		PCTPCLSQ	60	
PCTMINXM	DC		PCTPCLST	64	
PCTMLMER	7A		PCTPRPIQ	D0	

## \$PCT Cross Reference

Name	Hex Offset	Hex Value
PCTRESPQ	B4	
PCTRSTL	BC	
PCTRSTN	C0	
PCTSONQ	B0	
PCTTALET	54	
PCTTBFCT	44	
PCTTFEAT	1C4	
PCTTINQ	2C	
PCTTINQH	2C	
PCTTINQT	30	
PCTTMPIH	3C	
PCTTMPIQ	3C	
PCTTMPIT	40	
PCTTOD	1B0	
PCTTRCPH	34	
PCTTRCPQ	34	
PCTTRCPT	38	
PCTTTOL	28	
PCTTTOLD	28	15180
PCTVER	4	
PCTVERN	4	3
PCTVINQ	C8	
PCTWINX	E0	
PCTWORKQ	A4	
PCT1DOWN	90	2
PCT1FPNP	90	40
PCT1NERR	90	8
PCT1NOT	90	10
PCT1NREC	90	4
PCT1NTUP	90	20
PCT1PATH	90	80
PCT2NSIM	91	1
PCT2NSSF	91	40
PCT2NSUB	91	80

**\$PCTAB Information**

**\$PCTAB Programming Interface information**

Programming Interface information

\$PCTAB

End of Programming Interface information

## Heading Information • \$PCTAB Map

### \$PCTAB Heading Information

**Common Name:** PC table entry  
**Macro ID:** \$PCTAB  
**DSECT Name:** PCRT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: N/A  
 Key: N/A  
 Residency: These table entries are part of the HASJES20 load module and are located below 16M. Real storage can be anywhere.  
**Size:** See PCRTLEN  
**Created by:** \$PCTAB macro expansion in HASPTAB  
**Pointed to by:** MCTPCRTU field of the \$MCT data area  
 MCTPCRTH field of the \$MCT data area  
**Serialization:** None required  
**Function:** This DSECT maps entries in the PC routine table pairs which describe JES2 main task and user address space PC routines.

### \$PCTAB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCRT	
0	(0)	CHARACTER	8	PCRTNAME	PC routine name
8	(8)	CHARACTER	24	PCRTDESC	PC routine description
32	(20)	BITSTRING	1	PCRTFLG1	PC table flags
		1... ....		PCRT1PCU	"B'10000000" Entry is USER PC routine
		.1.. ....		PCRT1PCH	"B'01000000" Entry is HASP PC routine
		..1. ....		PCRT1SLX	"B'00100000" System LX to be used
		...1 ....		PCRT1CKA	"B'00010000" CALLKEY=ANY specified
33	(21)	BITSTRING	1	PCRTFLG2	PC address locations
Comment					
EQU B'000000xx' ENTRYPT field in PADDR					
End of Comment					
		1... ....		PCRT2EPU	"B'10000000" ENTRYPT field in the UCT
		.1.. ....		PCRT2EPM	"B'01000000" ENTRYPT field in MODMAP
		..1. ....		PCRT2EPC	"B'00100000" ENTRYPT field in CADDR
		...1 ....		PCRT2EPD	"B'00010000" ENTRYPT field in UCADDR
		.... 1...		PCRT2EPN	"B'00001000" ENTRYPT field is RTN name
		.... .1..		PCRT2EAV	"B'00000100" ENTRYPT field is ADDR/VCON
34	(22)	BITSTRING	1	PCRTFLG3	PC routine flags
		1... ....		PCRT3SUP	"B'10000000" Routine runs in supervisor state
Comment					
EQU B'x000000x' RECOVPT field in PADDR					
End of Comment					
		.1.. ....		PCRT3RVU	"B'01000000" RECOVPT field in the UCT
		..1. ....		PCRT3RVM	"B'00100000" RECOVPT field in MODMAP
		...1 ....		PCRT3RVC	"B'00010000" RECOVPT field in CADDR
		.... 1...		PCRT3RVD	"B'00001000" RECOVPT field in UCADDR
		.... .1..		PCRT3RVN	"B'00000100" RECOVPT field is RTN name
		.... ..1.		PCRT3RAV	"B'00000010" RECOVPT field is ADDR/VCON
35	(23)	ADDRESS	1	PCRTKEY	PC routine run key
36	(24)	ADDRESS	4	PCRTENTY	Offset/Addr ENTRYPT field
40	(28)	ADDRESS	4	PCRTRECV	Offset/Addr RECOVPT field

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
44	(2C)	ADDRESS	2	PCRTADDR	Offset in xADDR/UxADDR to store PC #
46	(2E)	ADDRESS	1	PCRTNAML	Length-1 for name in PCRTNAME
47	(2F)	ADDRESS	1	PCRTDESL	Length-1 for description in PTABDESC
48	(30)	ADDRESS	4	PCRTPCNM	PC number for this service
52	(34)	CHARACTER	8	PCRTENTN	ENTRYPT routine name
60	(3C)	CHARACTER	8	PCRTRECN	RECOVPT routine name
68	(44)	BITSTRING	4		RESERVED
72	(48)	SIGNED	4	(0)	Align PCRT entry
72	(48)	X'48'	0	PCRTELEN	**-PCRT" Length of PCRT entry DSECT

**\$PCTAB Cross Reference**

Name	Hex Offset	Hex Value
PCRT	0	
PCRTADDR	2C	
PCRTDESC	8	
PCRTDESL	2F	
PCRTELEN	48	48
PCRTENTN	34	40404040
PCRTENTY	24	
PCRTFLG1	20	
PCRTFLG2	21	
PCRTFLG3	22	
PCRTKEY	23	
PCRTNAME	0	
PCRTNAML	2E	
PCRTPCNM	30	
PCRTRECN	3C	40404040
PCRTRECV	28	
PCRT1CKA	20	10
PCRT1PCH	20	40
PCRT1PCU	20	80
PCRT1SLX	20	20
PCRT2EAV	21	4
PCRT2EPC	21	20
PCRT2EPD	21	10
PCRT2EPM	21	40
PCRT2EPN	21	8
PCRT2EPU	21	80
PCRT3RAV	22	2
PCRT3RVC	22	10
PCRT3RVD	22	8
PCRT3RVM	22	20
PCRT3RVN	22	4
PCRT3RVU	22	40
PCRT3SUP	22	80

## \$PCTAB Cross Reference



---

## \$PDDB Information

### \$PDDB Programming Interface information

Programming Interface information

\$PDDB

End of Programming Interface information

## Heading Information • \$PDDB Map

### \$PDDB Heading Information

**Common Name:** JES2 Peripheral Data Definition Block  
**Macro ID:** \$PDDB  
**DSECT Name:** PDB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: Same as the \$IOT that contains the PDDB  
 Key: Same as the \$IOT that contains the PDDB  
 Residency: See \$IOT  
**Size:** See PDBLEN1 and PDBLENM  
**Created by:** \$PDBBLD creates a PDDB within an IOT. Storage is obtained when the IOT in which it resides is created. See \$IOT for additional information.  
**Pointed to by:** IOTPDDB field of the \$IOT data area contains the offset from the beginning of the IOT to the first PDDB within the IOT. Various fields in the processor work areas. Various fields in the exit parameter lists (\$XPL).  
**Serialization:** JES2 reentrancy techniques for PDDBs in the JES2 main task environment. SJB lock for PDDBs in the USER environment.  
**Function:** The Peripheral Data Set Definition Block (\$PDDB) contains or points to all characteristics, known at the time of creation of the PDDB, of each subsystem data set known to JES2. PDDBs are contained in the Input/Output Table (\$IOT), which is a spool resident JES2 job control block.

There is a PDDB for each instance of a spool data set. An instance is defined as a set of characteristics combined with a set of data. For example, a single data set may have 5 JCL output cards and 5 PDDBs will be created.

### \$PDDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PDB	HASP PDDB DSECT
0	(0)	BITSTRING	1	PDBFLAG1 (0)	Flag byte
1	(1)	BITSTRING	1	PDBRECFM	Data set record format
2	(2)	SIGNED	2	PDBLRECL	Maximum data set LRECL
4	(4)	BITSTRING	4	PDBMTTR	Starting track address of data set
8	(8)	BITSTRING	4	PDBMTTRL	Last track address of DS
12	(C)	SIGNED	4	PDBDSKEY	Data set number of data set
16	(10)	ADDRESS	2	PDBSIZE	Size of a PDDB
18	(12)	CHARACTER	1	PDBCLASS	Output class of data set
19	(13)	ADDRESS	1	PDBCOPYS	Copies of this data set
20	(14)	SIGNED	4	PDBDEST (0)	Data set output dest
20	(14)	SIGNED	2	PDBDNODE	Node number (binary)
22	(16)	SIGNED	2	PBDRMT	Remote number (binary)
24	(18)	CHARACTER	8	PDBUSER (0)	Dataset dest USERID/rmtid
32	(20)	BITSTRING	1	(2)	Reserved for future use
34	(22)	BITSTRING	1	PDBFLAG2	Second flag byte
35	(23)	BITSTRING	1	PDBCPTN	Compaction table number
36	(24)	SIGNED	4	PDBRECCT	Data set record count
40	(28)	SIGNED	4	PDBPGCT	Page data page count
44	(2C)	SIGNED	4	PDBBYTCT	Actual byte count
48	(30)	CHARACTER	8	PDBFORMS (0)	Eight-byte forms number

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
48	(30)	X'30'	0	PDBLOGDD	"PDBFORMS,8,C'C" DD name of logging dataset (see PDBYLOG)
56	(38)	CHARACTER	4	PDBFCB (0)	Four-byte 3211 FCB id
60	(3C)	CHARACTER	4	PDBUCS (0)	Four-byte 1403 or 3211 UCS id
64	(40)	CHARACTER	8	PDBDSID (0)	3540 dataset id
64	(40)	X'40'	0	PDBWTRID	"PDBDSID,,C'C" 8-byte output writer id
72	(48)	BITSTRING	8	PDBRBA	8-byte checkpoint record

Comment

THE FOLLOWING FIELDS MUST BE KEPT TOGETHER FOR SJF UPDATE

End of Comment

80	(50)	CHARACTER	4	PDBCHAR1 (0)	N/I Printer xlate tab 1
84	(54)	CHARACTER	4	PDBCHAR2 (0)	N/I Printer xlate tab 2
88	(58)	CHARACTER	4	PDBCHAR3 (0)	N/I Printer xlate tab 3
92	(5C)	CHARACTER	4	PDBCHAR4 (0)	N/I Printer xlate tab 4

Comment

END OF THE FIELDS NEEDED FOR SJF UPDATE

End of Comment

96	(60)	CHARACTER	4	PDBFLASH (0)	N/I Printer flash cart id
100	(64)	CHARACTER	4	PDBMODF (0)	N/I Printer copy mod image
104	(68)	BITSTRING	1	PDBFLSHC	N/I printer # flash copies
105	(69)	BITSTRING	1	PDBMODFT	N/I printer tbl ref char
106	(6A)	BITSTRING	8	PDBCOPYG (0)	N/I Printer copy groups
114	(72)	BITSTRING	2	PDBCKPTP	Nr of logical page/ckpt
116	(74)	BITSTRING	2	PDBCKPTL	Nr of lines/logical page
118	(76)	BITSTRING	1	PDBFLAG3	The third flag byte
119	(77)	BITSTRING	1	PDBFLAGY	Symbol substitution flags for DD * and DD DATA
120	(78)	CHARACTER	2	PDBID (0)	Output id qualifier for JOE
122	(7A)	CHARACTER	8	PDBNAME (0)	Output name for this PDDB
132	(84)	SIGNED	4	PDBCRTME	Create Time
136	(88)	SIGNED	4	PDBSEGID	Segment identifier
140	(8C)	SIGNED	4	PDBGGTOK	Generic grouping token

Comment

WHEN USED AS A SPIN PDDB

End of Comment

144	(90)	ADDRESS	4	PDBPLIOT	Pointer to normal IOT place holder
148	(94)	SIGNED	4	PDBPLOFF	The offset to related PDDB

Comment

WHEN USED AS A PLACE HOLDER PDDB

End of Comment

144	(90)	SIGNED	4	PDBSPTRR	MTTR of spin IOT
152	(98)	CHARACTER	4	PDBSSOFM (0)	Save forms at allocation
156	(9C)	CHARACTER	8	PDBPNAME (0)	Proc step name
164	(A4)	CHARACTER	8	PDBSNAME (0)	Step name
172	(AC)	CHARACTER	8	PDBDDNAM (0)	DD name
180	(B4)	CHARACTER	8	PDBPRMD (0)	PRMODE name
188	(BC)	BITSTRING	80	PDBTOKEN (0)	Security tkn
268	(10C)	CHARACTER	8	PDBCRUID (0)	Creator userid
276	(114)	CHARACTER	8	PDBSECLB (0)	Security label
284	(11C)	BITSTRING	1	PDBLINCT	Dataset line cnt (LINECNT)
285	(11D)	BITSTRING	1	PDBINDEX	3211 FCB index value
286	(11E)	BITSTRING	1	PDBFUNC	3525 function byte
287	(11F)	BITSTRING	1	PDBPRIO	Data set priority

# \$PDDDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
All PDDDB flags should be cleared on return from exit 47 in module HASPNSR.					
End of Comment					
288	(120)	BITSTRING	1	PDBFLAG4	The fourth flag byte
289	(121)	BITSTRING	1	PDBFLAG5	The fifth flag byte
290	(122)	BITSTRING	1	PDBFLAG6	The sixth flag byte
291	(123)	BITSTRING	1	PDBFLAG7	The seventh flag byte
292	(124)	SIGNED	4	PDBSWBOT	Starting trk output SWBs
296	(128)	BITSTRING	8	PDBOUTOK (0)	Data sets output SWB token
304	(130)	CHARACTER	8	PDBCPTNM (0)	Compact table name
312	(138)	SIGNED	4		Reserved for future use
316	(13C)	SIGNED	4	PDBNDHTR	MTTR of dataset header
320	(140)	BITSTRING	1	PDBFLAG8	Eighth flag byte
321	(141)	BITSTRING	1	PDBFLAG9	Ninth flag byte
322	(142)	SIGNED	2	PDBDSINS	Multi-dest instance count
324	(144)	SIGNED	4		Reserved for future use
328	(148)	BITSTRING	6	PDBINDXM	MQTR of 1st data set index
334	(14E)	BITSTRING	6	PDBCATLM	MQTR of data set catalog
340	(154)	CHARACTER	44	PDBDSNAM (0)	Data set name
384	(180)	SIGNED	4	(0)	
384	(180)	X'180'	0	PDBLEN1	** -PDB" Base PDDDB length (should never change)
384	(180)	X'180'	0	PDBLENM	** -PDB" Maximum PDDDB length (can change rel to rel)
Comment					
PDBFLAG1					
End of Comment					
	1... ....			PDB1NEWS	"B'10000000" JESNEWS DATA SET
	.1.. ....			PDB1NULL	"B'01000000" THIS IS A NULL PDDDB
	..1. ....			PDB1LOG	"B'00100000" THIS PDDDB IS FOR THE HASP JOB LOG
	...1 ....			PDB1MOC	"B'00010000" Multiple Output Characteristic(MOC) Spin data set indicator
	.... 1...			PDB1NSOT	"B'00001000" DATA SET IS NOT FOR SYSOUT
	.... .1..			PDB1SPIN	"B'00000100" THIS PDDDB IS FOR A SPIN DATA SET
	.... ..1.			PDB1USER	"B'00000010" This PDDDB is for a user
	.... ...1			PDB1DAUG	"B'00000001" This is a daughter spin IOT
Comment					
PDBFLAG2					
End of Comment					
	1... ....			PDB2TCEL	"B'10000000" DATA SET IS TRAKCELL'ED
	.1.. ....			PDB2OPTJ	"B'01000000" OPTCD=J SPECIFIED
	..1. ....			PDB2BRST	"B'00100000" BURST=YES SPECIFIED
	...1 ....			PDB2PRIO	"B'00010000" Installation set JOE prio. Or dataset re-loaded.
	.... 1...			PDB2JFMS	"B'00001000" PDBFORMS SET FROM JCTFORMS
	.... .1..			PDB2HLDS	"B'00000100" HOLD= SPECIFIED ON DD
	.... ..1.			PDB2PSOR	"B'00000010" PSO ROUTE CHANGE
	.... ...1			PDB2FOLD	"B'00000001" JFCFOLD WAS SPECIFIED
Comment					
PDBFLAG3					
End of Comment					
	1... ....			PDB3PLHD	"B'10000000" PLACE HOLDER PDDDB
	.1.. ....			PDB3PSOC	"B'01000000" PSO CLASS CHANGE
	..1. ....			PDBLNCTF	"B'00100000" LINECT SPECIFIED

Offsets		Dec	Hex	Type/Value	Len	Name (Dim)	Description
				...1 ....		PDB3STAT	"B'00010000" JOB STATISTICS IN JOB LOG
				.... 1...		PDB3LINE	"B'00001000" DATA SET HAS LINE MODE RECORDS
				.... .1..		PDB3PAGE	"B'00000100" DATA SET HAS PAGE DATA RECORDS
				.... ..1.		PDB3SP2	"B'00000010" FORCED DOUBLE SPACING
				.... ...1		PDB3SP1	"B'00000001" FORCED SINGLE SPACING
Comment							
PDBFLAG4							
End of Comment							
				1... ....		PDB4OUTJ	"B'10000000" PDDDB IS REFERENCE BY OUTPUT JCL
				.1.. ....		PDB4BRST	"B'01000000" BURST (Y OR N) IN DD CARD
				..1. ....		PDB4JFCB	"B'00100000" MERGE JFCB INTO THIS PDDDB
				...1 ....		PDB4SYSN	"B'00010000" SYSTEM GENERATED NAME
				.... 1...		PDB4OCLS	"B'00001000" CLASS SET FROM OUTPUT JCL
				.... .1..		PDB4OCPY	"B'00000100" COPY SET FROM OUTPUT JCL
				.... ..1.		PDBPLNUL	"B'00000010" NULLIFIED PLACEHOLDER PDDDB
				.... ...1		PDB4SDBT	"B'00000001" FORCE SDB TO TRACKCELL
Comment							
PDBFLAG5							
End of Comment							
				1... ....		PDB5OPEN	"B'10000000" DATASET IS OPEN FOR OUTPUT
				.1.. ....		PDB5NUNK	"B'01000000" Token is NJE unknown user
				..1. ....		PDB5SPAU	"B'00100000" DATA SET FAILED SPOOL OFFLOAD AUTHORIZATION
				...1 ....		PDB5PTKN	"B'00010000" Token to be propagated
				.... 1...		PDB5AODS	"B'00001000" Use abnormal disp for spin
				.... .1..		PDB5XBM2	"B'00000100" Data set is SYSIN for XBM/2
				.... ..1.		PDB5PRGA	"B'00000010" \$PURGE auth check required
				.... ...1		PDB5SAFD	"B'00000001" \$PURGE auth check footprint
Comment							
PDBFLAG6							
End of Comment							
				..1. ....		PDB6AOSO	"B'00100000" ABNORMAL OUTDISP SPECIFIED ON OUTPUT JCL STATEMENT
384	(180)	X'10'	0			PDB6AODP	"\$ODPURGE" ABNORMAL OUTDISP=PURGE
384	(180)	X'8'	0			PDB6AODW	"\$ODWRITE" ABNORMAL OUTDISP=WRITE
384	(180)	X'4'	0			PDB6AODH	"\$ODHOLD" ABNORMAL OUTDISP=HOLD
384	(180)	X'2'	0			PDB6AODK	"\$ODKEEP" ABNORMAL OUTDISP=KEEP
384	(180)	X'1'	0			PDB6AODL	"\$ODLEAVE" ABNORMAL OUTDISP=LEAVE
384	(180)	X'1F'	0			PDB6AODA	"\$ODANYWP" CHECK ALL BIT SETTINGS
Comment							
PDBFLAG7							
End of Comment							
				1... ....		PDB7DSWB	"B'10000000" OUTPUT SWB to be deleted
				.1.. ....		PDB7HOPX	"B'01000000" SYSTEM HOLD DATASET DUE TO HOP COUNT EXCESSION
				..1. ....		PDB7NOSO	"B'00100000" NORMAL OUTDISP SPECIFIED ON OUTPUT JCL STATEMENT
384	(180)	X'10'	0			PDB7NODP	"\$ODPURGE" NORMAL OUTDISP=PURGE
384	(180)	X'8'	0			PDB7NODW	"\$ODWRITE" NORMAL OUTDISP=WRITE
384	(180)	X'4'	0			PDB7NODH	"\$ODHOLD" NORMAL OUTDISP=HOLD
384	(180)	X'2'	0			PDB7NODK	"\$ODKEEP" NORMAL OUTDISP=KEEP
384	(180)	X'1'	0			PDB7NODL	"\$ODLEAVE" NORMAL OUTDISP=LEAVE

## \$PDDDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
384	(180)	X'1F'	0	PDB7NODA	"\$ODANYWP" CHECK ALL BIT SETTINGS
Comment					
PDBFLAG8					
End of Comment					
		1... ....		PDB8DSID	"B'10000000" DSID IN PDBSID
		.1.. ....		PDB8FORM	"B'01000000" FORM was explicitly coded either on the DD, OUTPUT JCL or inherited from JOBPARM FORM=
		..1. ....		PDB8UNDF	"B'00100000" Userid is undefined
		...1 ....		PDB8NREU	"B'00010000" THE SPIN IOT REPRESENTED BY THIS PLACE HOLDER PDDDB IS NOT REUSABLE
		.... 1...		PDB8UPRI	"B'00001000" PRTY was specified on the OUTPUT JCL statement
		.... .1..		PDB8UNAL	"B'00000100" SYSOUT data set has been unallocated (not set for any special data sets)
		.... ..1.		PDB8SYIN	"B'00000010" SYSIN data set (from input services)
		.... ...1		PDB8RERT	"B'00000001" Dataset was rerouted by SYSOUT receiver
Comment					
PDBFLAG9					
<p>-----</p> <p>The following flag is set only at sysout data set allocation time and later used in setting the corresponding bit JOE2IPAD in \$JOE during output grouping. Subsequent processing will only use the \$JOE flag.</p> <p>-----</p>					
End of Comment					
		1... ....		PDB9IPAD	"B'10000000" Dataset's dest in IP-format
Comment					
<p>-----</p> <p>PDB9CTKN indicates that a client token was returned as part of the dynamic allocation for the dataset. PDB9CTKN is only turned on for the first data set for MOCHA. PDB9CTKN is never turned off.</p> <p>-----</p>					
End of Comment					
		.1.. ....		PDB9CTKN	"B'01000000" Client token returned for data set
		..1. ....		PDB9JESL	"B'00100000" SPIN-ANY spin data set
		...1 ....		PDB9SALC	"B'00010000" Separate track group map used to create data set
		.... 1...		PDB9TRC	"B'00001000" TRC was specified on the OUTPUT JCL statement
		.... .1..		PDB9CRTM	"B'00000100" PDBCRTME is from JOE (spool reload)
		.... ..1.		PDB9ONOT	"B'00000010" Issue NOTIFY from HOPE
		.... ...1		PDB9INDX	"B'00000001" Data set is indexed

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
PDBFLAGY					
-----					
PDBFLAGY flag byte is used to control instream symbol substitution.					
PDBYSCTL defines the substitution control bits.					
PDBYJCL/PDBYEXEC/PDBYCNVT should be compared for equality, not as individual bits.					
-----					
End of Comment					
	111.	....		PDBYSCTL	"B'11100000" Substitution type: 000 - no substitution
	..1.	....		PDBYJCL	"B'00100000" SYMBOLS=JCLONLY
	.1..	....		PDBYEXEC	"B'01000000" SYMBOLS=EXECSYS
	.11.	....		PDBYCNVT	"B'01100000" SYMBOLS=CNVTSYS
	...1	....		PDBYLOG	"B'00010000" substitution logging requested

Comment					
DATA SET KEYS FOR SPECIAL DATA SETS					
End of Comment					
384	(180)	X'1'	0	PDBINJCL	"1" INPUT JCL STATEMENTS
384	(180)	X'2'	0	PDBOUHJL	"2" HASP JOB LOG
384	(180)	X'3'	0	PDBOUJCI	"3" JCL IMAGES
384	(180)	X'4'	0	PDBOUMSG	"4" SYSTEM MESSAGES
384	(180)	X'5'	0	PDBINTXT	"5" INTERNAL TEXT
384	(180)	X'6'	0	PDBINJNL	"6" JOB JOURNAL
384	(180)	X'7'	0	PDBSWABL	"7" SWA blocks
384	(180)	X'64'	0	PDBUISKY	"100" INITIAL DATA SET KEY NUMBER

**\$PDDDB Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PDB	0		PDBFLAG2	22	0
PDBBYTCT	2C	0	PDBFLAG3	76	0
PDBCATLM	14E	0	PDBFLAG4	120	0
PDBCHAR1	50	5C5C5C5C	PDBFLAG5	121	0
PDBCHAR2	54	5C5C5C5C	PDBFLAG6	122	0
PDBCHAR3	58	5C5C5C5C	PDBFLAG7	123	0
PDBCHAR4	5C	5C5C5C5C	PDBFLAG8	140	0
PDBCKPTL	74	FFFF	PDBFLAG9	141	0
PDBCKPTP	72	FFFF	PDBFLASH	60	5C5C5C5C
PDBCLASS	12	C1	PDBFLSHC	68	FF
PDBCOPYG	6A	0	PDBFORMS	30	0
PDBCOPYS	13		PDBFUNC	11E	0
PDBCPTN	23	FF	PDBGGTOK	8C	0
PDBCPTNM	130	0	PDBID	78	0
PDBCRTME	84	0	PDBINDEX	11D	0
PBCRUID	10C	40404040	PDBINDXM	148	0
PBDDNAM	AC	40404040	PDBINJCL	180	1
PDBDEST	14		PDBINJNL	180	6
PBDNODE	14	0	PDBINTXT	180	5
PBDRMT	16	0	PDBLENM	180	180
PBDSID	40	40404040	PDBLEN1	180	180
PBDSINS	142	0	PDBLINCT	11C	0
PBDSKEY	C	0	PDBLNCTF	180	20
PBDSNAM	154	40404040	PDBLOGDD	30	30
PDBFCB	38	5C5C5C5C	PDBLRECL	2	0
PDBFLAGY	77	0	PDBMODF	64	5C5C5C5C
PDBFLAG1	0	40	PDBMODFT	69	0

## \$PDDDB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PDBMTTR	4	0	PDB4SYSN	180	10
PDBMTTRL	8	0	PDB5AODS	180	8
PDBNAME	7A	0	PDB5NUNK	180	40
PDBNDHTR	13C	0	PDB5OPEN	180	80
PDBOUHJL	180	2	PDB5PRGA	180	2
PDBOUJCI	180	3	PDB5PTKN	180	10
PDBOUMSG	180	4	PDB5SAFD	180	1
PDBOUTOK	128	0	PDB5SPAU	180	20
PDBPGCT	28	0	PDB5XBM2	180	4
PDBPLIOT	90		PDB6AODA	180	1F
PDBPLNUL	180	2	PDB6AODH	180	4
PDBPLOFF	94	0	PDB6AODK	180	2
PDBPNAME	9C	40404040	PDB6AODL	180	1
PDBPRIO	11F	0	PDB6AODP	180	10
PDBPRMD	B4	40404040	PDB6AODW	180	8
PDBRBA	48	0	PDB6AOSO	180	20
PDBRECCT	24	0	PDB7DSWB	180	80
PDBRECFM	1	0	PDB7HOPX	180	40
PDBSECLB	114	40404040	PDB7NODA	180	1F
PDBSEGID	88	0	PDB7NODH	180	4
PDBSIZE	10		PDB7NODK	180	2
PDBSNAME	A4	40404040	PDB7NODL	180	1
PDBSPTRR	90	0	PDB7NODP	180	10
PDBSSOFM	98	0	PDB7NODW	180	8
PDBSWABL	180	7	PDB7NOSO	180	20
PDBSWBOT	124	0	PDB8DSID	180	80
PDBTOKEN	BC	0	PDB8FORM	180	40
PDBUCS	3C	5C5C5C5C	PDB8NREU	180	10
PDBUISKY	180	64	PDB8RERT	180	1
PDBUSER	18	0	PDB8SYIN	180	2
PDBWTRID	40	40	PDB8UNAL	180	4
PDBYCNVT	180	60	PDB8UNDF	180	20
PDBYEXEC	180	40	PDB8UPRI	180	8
PDBYJCL	180	20	PDB9CRTM	180	4
PDBYLOG	180	10	PDB9CTKN	180	40
PDBYSCTL	180	E0	PDB9INDX	180	1
PDB1DAUG	180	1	PDB9IPAD	180	80
PDB1LOG	180	20	PDB9JESL	180	20
PDB1MOC	180	10	PDB9ONOT	180	2
PDB1NEWS	180	80	PDB9SALC	180	10
PDB1NSOT	180	8	PDB9TRC	180	8
PDB1NULL	180	40			
PDB1SPIN	180	4			
PDB1USER	180	2			
PDB2BRST	180	20			
PDB2FOLD	180	1			
PDB2HLDS	180	4			
PDB2JFMS	180	8			
PDB2OPTJ	180	40			
PDB2PRIO	180	10			
PDB2PSOR	180	2			
PDB2TCEL	180	80			
PDB3LINE	180	8			
PDB3PAGE	180	4			
PDB3PLHD	180	80			
PDB3PSOC	180	40			
PDB3SP1	180	1			
PDB3SP2	180	2			
PDB3STAT	180	10			
PDB4BRST	180	40			
PDB4JFCB	180	20			
PDB4OCLS	180	8			
PDB4OCPY	180	4			
PDB4OUTJ	180	80			
PDB4SDBT	180	1			



---

## \$PERFCB Information

### \$PERFCB Heading Information

**Common Name:** Performance data anchor CB  
**Macro ID:** \$PERFCB  
**DSECT Name:** PERFCB INITSTAT QSUCB PPB PTPB WTCB PSCBD EVENT PCBCKIO PCBSBST  
PMIG  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** PCB  
Offset: PCBID  
Length: L'PCBID  
**Storage Attributes:** Subpool: 24 (Except PPB which are subpool 25)  
Key: 1  
Residency: Virtual and real storage are above 16M, in the private storage of the JES2  
address space.  
**Size:** See PCBLEN for PERFCB  
L'INITRENT for INITSTAT  
QSUCBLEN for QSUCB  
PPBLEN for PPB  
PTPBLEN for PTPB  
WTCBLEN for WTCB  
PSCBLEN for PSCBD  
EVNTLEN for EVENT  
PCBCKIOL for PCBCKIO  
PCBSBSTL for PCBSBST  
PMIGLEN for PMIG  
**Created by:** HASPIRA for the PCB and INITSTAT  
HASPDYN for the PPB and PTPB  
HASPNUC for the QSUCB, WTCB and PSCB  
HASPSUBS for the PCBSBST  
HASPSPOL for the PMIG

## \$PERFCB Map

### Pointed to by:

PERFCB  
 - \$PERFCB field of the HCT data area  
 INITSTAT  
 - PCBINITS field of the PERFCB data area  
 QSUCB  
 - PCBQSUHD field of the PERFCB data area  
 - PCBQSUTL field of the PERFCB data area  
 - PCBQSNDX field of the PERFCB data area  
 - QSUCBANX field of the QSUCB data area  
 - QSUCBUNX field of the QSUCB data area  
 - QSUCBUPR field of the QSUCB data area  
 PPB  
 - Prefix area in front of every PCE  
 PTPB  
 - PCBPTPB field of the PERFCB data area  
 - PPBPTPB field of the PPB data area  
 - PCBNEXT field of the PTPB data area  
 WTCB  
 - PTPBWTCB field of the PTPB data area  
 - WTCBNEXT field of the WTCB data area  
 PSCBD  
 - QTCBPSCB field of the WTCB data area  
 - PSCBNEXT field of the PSCBD data area  
 EVENT  
 - PCBEVNTF field of the PERFCB data area  
 - PCBEVNTL field of the PERFCB data area  
 - EVNTNEXT field of the EVENT data area  
 PCBSBST  
 - PCBSBQUE field of the PERFCB data area  
 PMIG  
 - PCBMIGR field of the PERFCB data area

### Serialization:

Normal PCE dispatch serialization

### Function:

The \$PERFCB is the anchor control block for performance related data collected by JES2. This macro also contains DSECTs that describe areas that the PERFCB points to.

## \$PERFCB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Performance anchor CB
0	(0)	CHARACTER	4	PCBID	Control block id
4	(4)	ADDRESS	1	PCBVERSN	Control block version
4	(4)	X'2'	0	PCBVERN	"2" Version number
5	(5)	BITSTRING	3		Reserved
8	(8)	DBL WORD	8	PCBWORK	Double word work area

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>JES2 Initialization performance information. PCBINITS points to a vector of entries, one per 'initialization' routine. The last entry in the list is all zero. The DSECT maps the data within each vector element. All times are in micro-seconds.</p>					
End of Comment					
16	(10)	ADDRESS	4	PCBINITS	Pointer to the init stats

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	INITSTAT	, DSECT for initstats
0	(0)	CHARACTER	8	INITRNAM	Routine name
8	(8)	BITSTRING	8	INITRTIM	Run time for routine
16	(10)	BITSTRING	8	INITRCPU	CPU time for routine
16	(10)	X'0'	0	INITRENT	"INITSTAT,*-INITSTAT" Equate for entire entry

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT
Comment					
<p>\$QSUSE wait statistics. PCBQSNDX is a vector that of \$QSUSE statistics control blocks. Each element on the list represents a \$QSUSE macro that was called when JES2 did not own the CKPT data set queues. Each vector slot corresponds to an address range where the \$QSUSE was invoked. PCBQSUHC is another chain through the same control blocks, sorted with the most frequently used \$QSUSE entries first.</p>					
End of Comment					

20	(14)	ADDRESS	4	PCBQSUHD	Head for use sorted chain
24	(18)	ADDRESS	4	PCBQSUTL	Tail for use sorted chain
28	(1C)	ADDRESS	4	PCBQSNDX (32)	Index into address chain
156	(9C)	ADDRESS	4		Index for big addresses
156	(9C)	X'20'	0	PCBQSNXN	"32" Number of index entries
160	(A0)		16	PCBQSLRS	STCKE time at last reset
176	(B0)	DBL WORD	8	PCBQSINT	Interval since reset (microseconds)

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	QSUCB	, QSUSE performance CB
0	(0)	CHARACTER	4	QSUCBID	Eyecatcher
4	(4)	BITSTRING	1	QSUCBVRN	Version number
4	(4)	X'1'	0	QSUCBVR	"1" Current version
5	(5)	BITSTRING	3		Reserved
8	(8)	ADDRESS	4	QSUCBANX	Next entry address chain
12	(C)	ADDRESS	4	QSUCBUNX	Next entry use chain
16	(10)	ADDRESS	4	QSUCBUPR	Previous entry use chain
20	(14)	ADDRESS	4	QSUCBADR	Addr of \$QSUSE parmlist
24	(18)	SIGNED	4		Reserved
28	(1C)	SIGNED	4	QSUCBCNT	Use count

## \$PERFCB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
32	(20)	BITSTRING	8	QSUCBTIM	Wait time (microseconds)
40	(28)	SIGNED	4	QSUCBPLT (0)	Copy of \$QSUSE parmlist:
40	(28)	BITSTRING	4	QSUCBPIN	(Instruction after BASR)
44	(2C)	BITSTRING	1	QSUCBPFL	\$QSUSE parameter flag -see \$PARMLST/\$QSUFLG1
45	(2D)	BITSTRING	1		Reserved for future use
46	(2E)	CHARACTER	8	QSUCBPSC	Control section name
54	(36)	CHARACTER	8	QSUCBPSQ	Invoking seq number
62	(3E)	BITSTRING	2		Reserved for future use
64	(40)	DBL WORD	8	(0)	Alignment
64	(40)	X'40'	0	QSUCBLEN	**"QSUCB" Length of element

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Comment

PCE performance information. Information is tracked for each PCE and for each type of PCE. The PCE performance block (PPB) is located before each PCE in storage. The PCE type performance blocks (PTPB) represents a type of PCE and tracks information based on PCE types (as defined by \$PCETABs). The WTCB track \$WAIT based on \$WAIT invocation. The PSCBD tracks \$POSTs based on \$POST type.

End of Comment

184	(B8)	DBL WORD	8	PCBCPULD	CPU time of last dispatch
192	(C0)		16	PCBRUNLD	STCKE time of last dispatch
208	(D0)	DBL WORD	8	PCBCPULW	CPU time of last \$WAIT
216	(D8)		16	PCBRUNLW	STCKE time of last \$WAIT
232	(E8)	DBL WORD	8	PCBLSTRS	CPU time at last reset
240	(F0)		16	PCBLSTRT	STCKE time at last reset
256	(100)	DBL WORD	8	PCBRSTIS	CPU reset interval (micro)
264	(108)	DBL WORD	8	PCBRSTIT	STCK reset interval (micro)
272	(110)	ADDRESS	4	PCBPTPB	PTPB chain pointer
276	(114)	ADDRESS	4	PCBPTPBS	Sorted chain anchor
280	(118)	BITSTRING	2	PCBPCEID	PCEID of last disp PCE
282	(11A)	BITSTRING	2		Reserved
284	(11C)	CHARACTER	8	PCBPCNAM	PCE name of last disp PCE

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PPB	, PCE performance block
0	(0)	CHARACTER	4	PPBID	Eyecatcher
4	(4)	BITSTRING	1	PPBVERN	Version number
4	(4)	X'1'	0	PPBVER	"1" Current version
5	(5)	BITSTRING	1		Reserved

Comment

-----  
 This 2 byte code is post code for the last post of this PCE. Byte 1 is the post type (equates below). Byte 2 is the specific event post that placed the PCE on the ready queue. An event \$POST with byte 2 = 0 indicates a \$POST xx,FORCE.  
 -----

End of Comment

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
6	(6)	BITSTRING	2	PPBLPOST	Last post type (valid only when PCE is on the ready Q)
6	(6)	X'0'	0	PPBLPRES	"0" Resource post (Must be 0)
6	(6)	X'1'	0	PPBLPEVN	"1" Event post (Must be 1)
6	(6)	X'2'	0	PPBLPxec	"2" XECB post
6	(6)	X'3'	0	PPBLPSSI	"3" \$\$POST of a event
6	(6)	X'4'	0	PPBLPSUB	"4" \$POST from a subtask
8	(8)	ADDRESS	4	PPBPTPB	Addr of related PTPB
12	(C)	SIGNED	4	PPBWAITC	\$WAIT count for this PCE
16	(10)	DBL WORD	8	PPBCPUT	CPU time used by this PCE
24	(18)	DBL WORD	8	PPBRUNT	Run time used by this PCE
32	(20)	DBL WORD	8	PPBWAITT	Total \$WAIT time for PCE
40	(28)	DBL WORD	8	PPBQSUSE	\$QSUSE time used by PCE (in microseconds)
48	(30)	SIGNED	4	PPBIOCNT	I/O count for this PCE
52	(34)	SIGNED	4	PPBCKPTN	Num of \$CKPTs for this PCE
56	(38)	SIGNED	4	PPBPAWCT	Num of \$PAWSs for this PCE
60	(3C)	SIGNED	4		Reserved
64	(40)	SIGNED	4		Reserved
72	(48)	DBL WORD	8	(0)	Ensure alignment
72	(48)	X'48'	0	PPBLEN	** -PPB" Length of control block

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PTPB	, PCE type performance block
0	(0)	CHARACTER	4	PTPBID	Eyecatcher
4	(4)	ADDRESS	4	PTPBNEXT	Chain pointer
8	(8)	ADDRESS	4	PTPBNXT2	2ndary chain word
12	(C)	ADDRESS	4	PTPBTAB	Address of PCETAB
16	(10)	SIGNED	4	PTPBIOCT	I/O count for PCE type
24	(18)	DBL WORD	8	PTPBRUNT	Total run time for all PCEs
32	(20)	DBL WORD	8	PTPBCPUT	Total CPU time for all PCEs
40	(28)	DBL WORD	8	PTPBQSUS	Total \$QSUSE time for all PCEs (in microseconds)
48	(30)	ADDRESS	4	PTPBWTCH	Chain of \$WAIT CBs
52	(34)	SIGNED	4	PTPBCKPT	Num of \$CKPTs for all PCEs
56	(38)	SIGNED	4	PTPBPAWC	Num of \$PAWSs for all PCEs
60	(3C)	SIGNED	4		Reserved
64	(40)	SIGNED	4		Reserved
72	(48)	DBL WORD	8	(0)	Ensure alignment
72	(48)	X'48'	0	PTPBLEN	** -PTPB" Length of control block

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT
0	(0)	STRUCTURE	0	WTCB	, Wait performance block
0	(0)	CHARACTER	4	WTCBID	Eyecatcher
4	(4)	ADDRESS	4	WTCBNEXT	Pointer to next WTCB
8	(8)	DBL WORD	8	WTCBINFO (0)	Wait address information
8	(8)	ADDRESS	4	WTCBADR	Address of \$WAIT parmlist
12	(C)	ADDRESS	4	WTCBADR2	2ndary parmlist pointer
16	(10)	ADDRESS	4	WTCBPSCB	Address of PSCB chain
20	(14)	SIGNED	4	WTCBWCNT	Wait count
24	(18)	DBL WORD	8	WTCBWAIT	Total \$WAIT time for PCEs
32	(20)	DBL WORD	8	WTCBPLST (0)	Copy of \$WAIT parmlist:
32	(20)	BITSTRING	4	WTCBPINS	(Instruction after BASR)
36	(24)	BITSTRING	1	WTCBPFL1	\$WAIT macro option flags -see \$PARMLST/\$WTFLAG1
37	(25)	BITSTRING	1	WTCBPINH	Inhibitor: prevs PCE redispatch before specific \$POST

## \$PERFCB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
38	(26)	CHARACTER	8	WTCBPSEC	Control section name
46	(2E)	CHARACTER	8	WTCBPSEQ	Invoking seq number
54	(36)	ADDRESS	2	WTCBPRQO	Resource queue offset/0
56	(38)	SIGNED	4	WTCB2PLT (0)	Copy of 2ndary parmlst:
56	(38)	CHARACTER	8	WTCB2SEC	Control section name
64	(40)	CHARACTER	8	WTCB2SEQ	Invoking seq number
72	(48)	ADDRESS	4	WTCB2PLA	Addr of copy of 2ndary parmlst
76	(4C)	SIGNED	4		Reserved for future use
80	(50)	DBL WORD	8	(0)	Alignment
80	(50)	X'50'	0	WTCBLEN	"*-WTCB" Length of control block
80	(50)	X'10'	0	WTCB2PLL	"L'WTCB2SEC+L'WTCB2SEQ" Len of 2ndary parmlst

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT
0	(0)	STRUCTURE	0	PSCBD	, \$POST performance block
0	(0)	CHARACTER	4	PSCBID	Eyecatcher
4	(4)	ADDRESS	4	PSCBNEXT	Pointer to next PSCB
8	(8)	SIGNED	2	PSCBTYP	\$POST Type (see PPBLPOST)
10	(A)	BITSTRING	2		Reserved
12	(C)	SIGNED	4	PSCBWCNT	Wait count
16	(10)	DBL WORD	8	PSCBWAIT	Total \$WAIT time for PCEs
24	(18)	DBL WORD	8	(0)	Alignment
24	(18)	X'18'	0	PSCBLEN	"*-PSCBD" Length of control block

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Comment

Event reporting information. Every time an noteworthy event is noticed, a record is created to note the time and PCE that had the event. These are then displayed by \$D PERFDATA(EVENT). As new event types are added (EVNTTYPE), the \$SCANTAB in HASPSTAB must be updated.

End of Comment

292	(124)	ADDRESS	4	PCBEVNTF	1st event control block
296	(128)	ADDRESS	4	PCBEVNTL	Last (newest) event CB
300	(12C)	SIGNED	4	PCBEVCNT	Count of exception CBs
300	(12C)	X'64'	0	PCBEVNLM	"100" Limit of events tracked
304	(130)		16	PCBEVLRS	STCKE time at last reset
320	(140)	DBL WORD	8	PCBEVINT	STCK reset interval (micro)

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	EVENT	, Exception description
0	(0)	CHARACTER	4	EVNTID	Eyecatcher
4	(4)	ADDRESS	4	EVNTNEXT	Next exception
8	(8)	DBL WORD	8	EVNTTIME	STCK of exception
16	(10)	DBL WORD	8	EVNTDUR	Duration of event or 0 (micro seconds)
24	(18)	CHARACTER	8	EVNTPNAM	PCE/DCT name
32	(20)	CHARACTER	8	EVNTMODN	Related module name
40	(28)	CHARACTER	8	EVNTSEQN	Related seq number/offset
48	(30)	CHARACTER	8	EVNTJID	Related job id
56	(38)	SIGNED	1	EVNTTYPE	Type of event
56	(38)	X'1'	0	EVNTRUN	"1" Excessive run time

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
56	(38)	X'2'	0	EVNTABND	"2" ABEND/\$ERROR
56	(38)	X'3'	0	EVNTDIST	"3" \$DISTERR
57	(39)	BITSTRING	3		Reserved
60	(3C)	CHARACTER	8	EVNTDATA	Additional data
72	(48)	DBL WORD	8	(0)	Alignment
72	(48)	X'48'	0	EVNTLEN	**"EVENT" Length of control block

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Comment

CKPT reporting information.

End of Comment

328	(148)		16	PCBCKLRS	STCKE time at last reset
344	(158)	DBL WORD	8	PCBCKINT	STCK reset interval (micro)
352	(160)	DBL WORD	8	PCBCKTHL	Total hold time (STCK)
360	(168)	DBL WORD	8	PCBCKAHL	Average hold time (micro)
368	(170)	SIGNED	4	PCBCKCHL	Hold time count
372	(174)	SIGNED	4	PCBCKCDR	Dormancy time count
376	(178)	DBL WORD	8	PCBCKTDR	Total dormancy time (STCK)
384	(180)	DBL WORD	8	PCBCKADR	Avg dormancy time (micro)
392	(188)	SIGNED	4	PCBCKCNT	\$CKPT count
396	(18C)	SIGNED	4	PCBCKOCK	\$CKPT optimization count
400	(190)	SIGNED	4	PCBCKO4K	4K page optimization count
404	(194)	SIGNED	4	PCBCK4KC	Count of 4K pages written (in IW and FW)
408	(198)	SIGNED	4	PCBCKCBC	Count of control blocks written (in IW and FW)
412	(19C)	SIGNED	4		Reserved

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCBCKIO	,
0	(0)	CHARACTER	4	PCBCKION	Name of the I/O
4	(4)	SIGNED	4	PCBCKIOC	Count of I/Os
8	(8)	DBL WORD	8	PCBCKIOT	Total I/O time (STCK)
16	(10)	DBL WORD	8	PCBCKIOA	Average I/O time (micro)
24	(18)	SIGNED	4	PCBCKIO4	Total 4K page count
28	(1C)	SIGNED	4	PCBCKIOB	Total CB count
28	(1C)	X'20'	0	PCBCKIOL	**"PCBCKIO" Length of area

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT
416	(1A0)	DBL WORD	8	(0)	Align I/O data areas
416	(1A0)	BITSTRING	32	PCBCKR1	Data area for read 1
448	(1C0)	BITSTRING	32	PCBCKR2	Data area for read 2
480	(1E0)	BITSTRING	32	PCBCKPW	Data area for Primary Write
512	(200)	BITSTRING	32	PCBCKIW	Data area for Intermediate
544	(220)	BITSTRING	1	PCBCKFW	Data area for Final Write
544	(220)	X'5'	0	PCBCKIO#	"(*-PCBCKR1)/PCBCKIOL" Number of I/O entries

Comment

General subtask statistics.

End of Comment

576	(240)		16	PCBSBLRS	STCKE time at last reset
-----	-------	--	----	----------	--------------------------

## \$PERFCB Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
592	(250)	DBL WORD	8	PCBSBINT	STCK reset interval (micro)
600	(258)	ADDRESS	4	PCBSBQUE	Queue of subtask statistics
604	(25C)	SIGNED	4		Reserved

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCBSBST	,
0	(0)	CHARACTER	4	PCBSBID	Eyecatcher
4	(4)	CHARACTER	8	PCBSBNAM	Routine name
12	(C)	ADDRESS	4	PCBSBNXT	Chain field
16	(10)	SIGNED	4	PCBSBCNT	Count of requests
20	(14)	SIGNED	4		Reserved
24	(18)	DBL WORD	8	PCBSBQTM	Total queue time (micro)
32	(20)	DBL WORD	8	PCBSBQTA	Average queue time (micro)
40	(28)	DBL WORD	8	PCBSBRM	Total run time (micro)
48	(30)	DBL WORD	8	PCBSBRTA	Average run time (micro)
56	(38)	DBL WORD	8	PCBSBCTM	Total CPU time (micro)
64	(40)	DBL WORD	8	PCBSBCTA	Average CPU time (micro)
64	(40)	X'48'	0	PCBSBSTL	**"-PCBSBST" Length of area

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Comment

Device gatherer performance information.

End of Comment

608	(260)		16	PCBDGLRS	STCKE time at last reset
624	(270)	DBL WORD	8	PCBDGINT	STCK reset interval (micro)
632	(278)	SIGNED	4	PCBDGSYN	SYNCH count
636	(27C)	SIGNED	4	PCBDGUPS	Updates sent count
640	(280)	SIGNED	4	PCBDGUPR	Updates received count
644	(284)	SIGNED	4	PCBDGNTS	NITs sent
648	(288)	SIGNED	4	PCBDGNTB	NIT broadcasts
652	(28C)	SIGNED	4	PCBDGCNR	CNIT updates received
656	(290)	SIGNED	4	PCBDGCNU	CNITs updated
660	(294)	SIGNED	4		Reserved

Comment

SPOOL Migration performance information.

End of Comment

664	(298)	ADDRESS	4	PCBMIGR	Newest migration CB
-----	-------	---------	---	---------	---------------------

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PMIG	, Migration information
0	(0)	CHARACTER	4	PMIGID	Eyecatcher
4	(4)	ADDRESS	4	PMIGNEXT	Next migration
8	(8)	CHARACTER	6	PMIGSORC	Source vol being migrated
14	(E)	CHARACTER	6	PMIGTARG	Target vol of migration
20	(14)	BITSTRING	1	PMIGFLG1	Flag byte
		1... ....		PMIG1MOV	"B'10000000" Move migration
21	(15)	BITSTRING	3		Reserved
24	(18)	DBL WORD	8	PMIGSTRT	STCK time migration started
32	(20)	DBL WORD	8	PMIGINTT	Init phase time (micro)



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
40	(28)	DBL WORD	8	PMIGSETT	Setup phase time (micro)
48	(30)	DBL WORD	8	PMIGCPYT	Copy phase time (micro)
56	(38)	SIGNED	4	PMIGCPYC	Copy phase track count
60	(3C)	SIGNED	4	PMIGCPYM	Copy phase message count
64	(40)	DBL WORD	8	PMIGCUPT	Catchup phase time (micro)
72	(48)	SIGNED	4	PMIGCUPC	Catchup phase track count
76	(4C)	SIGNED	4	PMIGCUPM	Catchup phase message count
80	(50)	DBL WORD	8	PMIGCLNT	Cleanup phase time (micro)
88	(58)	DBL WORD	8	PMIGOVRT	Overall time for migration (micro)
96	(60)	DBL WORD	8	(0)	Alignment
96	(60)	X'60'	0	PMIGLEN	**-"PMIG" Length of control block

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PERFCB	, Resume PERFCB DSECT

Comment

Work selection performance information (WSSTAT)

End of Comment

668	(29C)	SIGNED	4	PCB#PNUM	Number of calls to \$#POST
672	(2A0)	DBL WORD	8	PCB#PCPU	\$#POST tot CPU time (micro)
680	(2A8)	DBL WORD	8	PCB#GCPU	\$#GET tot CPU time (micro)
688	(2B0)	SIGNED	4	PCB#GNUM	Number of calls to \$#GET
692	(2B4)	SIGNED	4	PCB#GXNM	Number of calls to \$#GET which used JOE index
696	(2B8)		16	PCB#WLRS	STCKE time at last reset
712	(2C8)	DBL WORD	8	PCB#WINT	STCK reset interval (micro)
720	(2D0)	ADDRESS	4	PCBTEWA	Address of TEWA (Timed Event Work Area)
724	(2D4)	SIGNED	4	(3)	Reserved
736	(2E0)	DBL WORD	8	(0)	Alignment
736	(2E0)	X'2E0'	0	PCBLEN	**-"PERFCB" Length of PERFCB

**\$PERFCB Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
EVENT	0		PCB#WINT	2C8	
EVNTABND	38	2	PCB#WLRS	2B8	
EVNTDATA	3C		PCBCKADR	180	
EVNTDIST	38	3	PCBCKAHL	168	
EVNTDUR	10		PCBCKCBC	198	
EVNTID	0	C5E5D5E3	PCBCKCDR	174	
EVNTJID	30		PCBCKCHL	170	
EVNTLEN	48	48	PCBCKCNT	188	
EVNTMODN	20		PCBCKFW	220	
EVNTNEXT	4		PCBCKINT	158	
EVNTPNAM	18		PCBCKIO	0	
EVNTRUN	38	1	PCBCKIO#	220	5
EVNTSEQN	28		PCBCKIOA	10	
EVNTTIME	8		PCBCKIOB	1C	
EVNTTYPE	38		PCBCKIOC	4	
INITRCPU	10		PCBCKIOL	1C	20
INITRENT	10	0	PCBCKION	0	
INITRNAM	0		PCBCKIOT	8	
INITRTIM	8		PCBCKIO4	18	
INITSTAT	0		PCBCKIW	200	
PCB#GCPU	2A8		PCBCKLRS	148	
PCB#GNUM	2B0		PCBCKOCK	18C	
PCB#GXNM	2B4		PCBCKO4K	190	
PCB#PCPU	2A0		PCBCKPW	1E0	
PCB#PNUM	29C		PCBCKR1	1A0	

## \$PERFCB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCBCKR2	1C0		PERFCB	0	
PCBCKTDR	178		PERFCB	0	
PCBCKTHL	160		PERFCB	0	
PCBCK4KC	194		PERFCB	0	
PCBCPULD	B8		PERFCB	0	
PCBCPULW	D0		PERFCB	0	
PCBDGCNR	28C		PMIG	0	
PCBDGCNU	290		PMIGCLNT	50	
PCBDGINT	270		PMIGCPYC	38	
PCBDGLRS	260		PMIGCPYM	3C	
PCBDGNTB	288		PMIGCPYT	30	
PCBDGNTS	284		PMIGCUPC	48	
PCBDGSYN	278		PMIGCUPM	4C	
PCBDGUPR	280		PMIGCUPT	40	
PCBDGUPS	27C		PMIGFLG1	14	
PCBEVCNT	12C		PMIGID	0	D7D4C9C7
PCBEVINT	140		PMIGINTT	20	
PCBEVLR	130		PMIGLEN	60	60
PCBEVNLM	12C	64	PMIGNEXT	4	
PCBEVNTF	124		PMIGOVRT	58	
PCBEVNTL	128		PMIGSETT	28	
PCBID	0	D7C3C240	PMIGSORC	8	
PCBINITS	10		PMIGSTRT	18	
PCBLEN	2E0	2E0	PMIGTARG	E	
PCBLSTRS	E8		PMIG1MOV	14	80
PCBLSTRT	F0		PPB	0	
PCBMIGR	298		PPBCKPTN	34	
PCBPCEID	118		PPBCPUT	10	
PCBPCNAM	11C		PPBID	0	
PCBPTPB	110		PPBIOCNT	30	
PCBPTPBS	114		PPBLEN	48	48
PCBQSINT	B0		PPBLPEVN	6	1
PCBQSLRS	A0		PPBLPOST	6	
PCBQSNDX	1C		PPBLPRES	6	0
PCBQSXXN	9C	20	PPBLPSSI	6	3
PCBQSUHD	14		PPBLPSUB	6	4
PCBQSUTL	18		PPBLPXC	6	2
PCBRSTIS	100		PPBPAWCT	38	
PCBRSTIT	108		PPBPTPB	8	
PCBRUNLD	C0		PPBQSUSE	28	
PCBRUNLW	D8		PPBRUNT	18	
PCBSBCNT	10		PPBVER	4	1
PCBSBCTA	40		PPBVERN	4	
PCBSBCTM	38		PPBWAITC	C	
PCBSBID	0	E2C2E2E3	PPBWAITT	20	
PCBSBINT	250		PSCBD	0	
PCBSBLRS	240		PSCBID	0	
PCBSBNAM	4		PSCBLEN	18	18
PCBSBNXT	C		PSCBNEXT	4	
PCBSBQTA	20		PSCBTYPE	8	
PCBSBQTM	18		PSCBWAIT	10	
PCBSBQUE	258		PSCBWCNT	C	
PCBSBRTA	30		PTPB	0	
PCBSBRTM	28		PTPBCKPT	34	
PCBSBST	0		PTPBPCPUT	20	
PCBSBSTL	40	48	PTPBID	0	
PCBTEWA	2D0		PTPBIOCT	10	
PCBVERN	4	2	PTPBLEN	48	48
PCBVERSN	4		PTPBNEXT	4	
PCBWORK	8		PTPBNXT2	8	
PERFCB	0		PTPBPAWC	38	
PERFCB	0		PTPBQSUS	28	
PERFCB	0		PTPBRUNT	18	
PERFCB	0		PTPBTAB	C	
PERFCB	0		PTPBWTCH	30	

Name	Hex Offset	Hex Value
QSUCB	0	
QSUCBADR	14	
QSUCBANX	8	
QSUCBCNT	1C	
QSUCBID	0	
QSUCBLEN	40	40
QSUCBPFL	2C	
QSUCBPIN	28	
QSUCBPLT	28	
QSUCBPSC	2E	
QSUCBPSQ	36	
QSUCBTIM	20	
QSUCBUNX	C	
QSUCBUPR	10	
QSUCBVR	4	1
QSUCBVRN	4	
WTCB	0	
WTCBADR	8	
WTCBADR2	C	
WTCBID	0	
WTCBINFO	8	
WTCBLEN	50	50
WTCBNEXT	4	
WTCBPFL1	24	
WTCBPINH	25	
WTCBPINS	20	
WTCBPLST	20	
WTCBPRQO	36	
WTCBPSCB	10	
WTCBPSEC	26	
WTCBPSEQ	2E	
WTCBWAIT	18	
WTCBWCNT	14	
WTCB2PLA	48	
WTCB2PLL	50	10
WTCB2PLT	38	
WTCB2SEC	38	
WTCB2SEQ	40	

## \$PERFCB Cross Reference

---

## \$PIT Information

### \$PIT Programming Interface information

Programming Interface information

\$PIT

End of Programming Interface information

## Heading Information • \$PIT Map

### \$PIT Heading Information

**Common Name:** Partition Information Table dsect  
**Macro ID:** \$PIT  
**DSECT Name:** PIT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PIT '  
Offset: LMT-\$CSBPRFX+\$CSBID  
Length: 4

**Storage Attributes:** Subpool: 241  
Key: 1  
Residency: Virtual and real storage are anywhere (above or below 16M) in common storage (CSA), once JES2 has been initialized. During a non-hot start JES2 initialization, temporary PITs exist in JES2 extended private in subpool 6.

**Size:** See the PITLEN equate.

**Created by:** Temporary PITs are created during JES2 initialization for the maximum number possible (the \$MAXINIT equate). Permanent PITs are created by JES2 initialization in CSA, and kept across possible JES2 outages, after the installation's INITDEF PARTNUM parameter is defined.

**Pointed to by:**

- The CCTPIT field of the \$HCCT data area in CSA points to the first PIT.
- The \$PITABLE field of the \$HCT data area in the JES2 address space points to the first PIT.
- The PITNEXT field of the previous \$PIT data area points to the next numbered PIT.
- The SJBPIT field of a batch-job \$SJB data area points to the PIT representing the initiator under which that job is running.

**Serialization:** Serialization of the PITs is implicit in the status flags in the PITs, and the way they are used by the JES2 main task. Except for the PITSTAT2 flag byte only the main task can alter the fields in the PITs.

**Function:** The Partition Information Table (PIT) represents a logic batch-job initiator. JES2 manages multiple logical initiators, which can be separately started, drained, and halted. Each can separately define the ordered list of job classes they will select from.

When an initiator is started by an operator, via a \$\$ I command, JES2 submits a START command to create a started task address space, running the MVS Initiator program. As that address space is started up, it is associated with the PIT for which it was started. The Initiator then makes generic subsystem-interface calls for any batch job to be run, without care to what job class or other criteria they have. JES2 applies the checks of criteria based on the logical initiator, the PIT.

Logical initiators can be managed as groups. That is, multiple PITs may have the same value in PITPATID. If, for example, 25 PITs are defined during initialization to have NAME= (PITPATID) XYZ, then commands to start, drain, display, etc 'I XYZ' will apply to all 25, and messages will indicate only XYZ (not the original number). The current ASID for the initiator is the only qualifying information.

### \$PIT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PIT	HASP PARTITION INFORMATION TBL DSECT
0	(0)	ADDRESS	4	PITNEXT	ADDRESS OF NEXT PIT
4	(4)	ADDRESS	4	PITSJB	ADDRESS OF SJB FOR EXECUTING JOB
8	(8)	ADDRESS	4	PITASCB	Address of initiator's ASCB
12	(C)	BITSTRING	8	PITASCBT	Address Space Token
20	(14)	ADDRESS	4	PITJQOFF	Offset of initiator's JQE
24	(18)	BITSTRING	1	PITFLAGS	PIT FLAG BYTE
		1... ....		PITDUPJ	"B'10000000" Job with duplicate jobname exists
		.1.. ....		PITSMVER	"B'01000000" Waiting for memory create
		..1. ....		PITSIVER	"B'00100000" PIT Awaiting \$\$I verification
		...1 ....		PITSRSOK	"B'00010000" Initiator start-up has progressed far enough to allow automatic restart
		.... 1...		PITCLFMT	"B'00001000" PITCLASS format : OFF = PITCLASS contains up to 36 one char class names. ON = PITCLASS contains a mixture of up to 8 eight char class names and class group names.
25	(19)	BITSTRING	1	PITSTAT	LOGICAL PARTITION STATUS BYTE
		1... ....		PITHOLDA	"B'10000000" PIT is drained (\$P I)
		.1.. ....		PITHOLD1	"B'01000000" PIT is drained (\$P IN)
		..1. ....		PITHOLDR	"B'00100000" PIT is stopped, but not via a command. PIT will automatically be restarted if PITHOLDA and PITHOLD1 are not on.
25	(19)	X'E0'	0	PITHOLD	"PITHOLDA+PITHOLD1+PITHOLDR" PIT is drained
		...1 ....		PITBUSY	"B'00010000" Partition busy indicator
		.... 1...		PITHALTA	"B'00001000" PIT is halted (\$Z I)
		.... .1..		PITHALT1	"B'00000100" PIT is halted (\$Z IN)
		.... ..1.		PITINIT	"B'00000010" OS initiator exists for PIT
		.... ...1		PITIDLE	"B'00000001" PIT 'idle' message sent and no jobs are executing on this started initiator
26	(1A)	BITSTRING	1	PITSTAT2	PIT status byte that may be updated outside JES2 address space and needs proper serialization. Update to this field should be via OIL/NIL
		1... ....		PIT2NSJB	"B'10000000" Init with no SJB needs to be cleaned up
27	(1B)	SIGNED	1	PITCLEAN	ID of PIT cleaner
27	(1B)	X'1'	0	PITCLN1	"1" PIT cleanup routine
27	(1B)	X'2'	0	PITCLN2	"2" Pending by class
27	(1B)	X'3'	0	PITCLN3	"3" XPITTERM
27	(1B)	X'4'	0	PITCLN4	"4" XINSTART (ASCRE failed)
27	(1B)	X'5'	0	PITCLN5	"5" XDRINIT
27	(1B)	X'6'	0	PITCLN6	"6" XREENQ
27	(1B)	X'7'	0	PITCLN7	"7" XINSTART (Before ASCRE)

Comment

- 
- The PITPATID is the initiator's 'name' or 'id'.
- 1) If NAME= was not coded on an INIT(n) init stmt for the initiator, PITPATID is the character value for its number, left-justified.
  - 2) If NAME= was coded on an initialization stmt for the initiator, PITPATID is that NAME= value, left-justified. The value does not have to be unique (and probably is not).

Commands against PITs can be done with subscript compares in character format, or with numerical indices. For example, \$\$I(1-20) starts all initiators in the numerical range 1-20 regardless of the value of PITPATID. \$\$I(ABC), on the other hand, starts all initiators with a name of ABC in PITPATID.

Normal \$SCAN rules apply when determining whether the specified subscript corresponds to the

## \$PIT Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
symbolic name or the numerical index.					
-----					
End of Comment					
28	(1C)	CHARACTER	4	PITPATID	Initiator partition 'id'
32	(20)	CHARACTER	64	PITCLASS	Class list. Contents depend on the PITCLFMT bit.
32	(20)	CHARACTER	36	PITCLS1	1 character class list (PITCLFMT bit off)
32	(20)	CHARACTER	8	PITCLS8 (0)	8 character class list (PITCLFMT bit on)
96	(60)	BITSTRING	4	PITJBKEY	Job key for INIT STC
100	(64)	SIGNED	2	PITASID	Initiator ASID
102	(66)	BITSTRING	6		Reserved
112	(70)	DBL WORD	8	(0)	Ensure doubleword length
112	(70)	X'70'	0	PITLEN	**-'PIT' PIT LENGTH

## \$PIT Cross Reference

Name	Hex Offset	Hex Value
PIT	0	
PITASCB	8	
PITASCBT	C	
PITASID	64	
PITBUSY	19	10
PITCLASS	20	
PITCLEAN	1B	
PITCLFMT	18	8
PITCLN1	1B	1
PITCLN2	1B	2
PITCLN3	1B	3
PITCLN4	1B	4
PITCLN5	1B	5
PITCLN6	1B	6
PITCLN7	1B	7
PITCLS1	20	
PITCLS8	20	
PITDUPJ	18	80
PITFLAGS	18	
PITHALTA	19	8
PITHALT1	19	4
PITHOLD	19	E0
PITHOLDA	19	80
PITHOLDR	19	20
PITHOLD1	19	40
PITIDLE	19	1
PITINIT	19	2
PITJBKEY	60	
PITJQOFF	14	
PITLEN	70	70
PITNEXT	0	
PITPATID	1C	
PITSIVER	18	20
PITSJB	4	
PITSMVER	18	40
PITSRSOK	18	10
PITSTAT	19	
PITSTAT2	1A	
PIT2NSJB	1A	80



---

## \$PPWORK Information

### \$PPWORK Programming Interface information

Programming Interface information

**\$PPWORK**

End of Programming Interface information

## Heading Information • \$PPPWORK Map

### \$PPPWORK Heading Information

**Common Name:** JES2 Print/Punch PCE Work Area  
**Macro ID:** \$PPPWORK  
**DSECT Name:** PCE (\$PPPWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol PPLEN for the length of this work area.  
 The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE for local printers and punches  
 Created by \$PCEDYN when RJE devices sign on

**Pointed to by:** The \$PRTPCE field of the \$HCT data area, and the \$PUNPCE field of the \$HCT data area, and the \$TPPRPCE field of the \$HCT data area, and the \$TPPUPCE field of the \$HCT data area points into the \$PCEORG/\$PCELAST chain to the first Print/Punch PCE. Since this chain contains all PCEs, use the PCEID field to determine when you have chained past the last PCE of this type.  
 The DCTPCE field of the \$DCT data area (see "Function" below)  
 See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization

**Function:** The fields in this work area are used by a JES2 Print/Punch Processor and by its support routines and exits. \$PPPWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$PPPWORK are actually part of the PCE DSECT, but only map PCEs with the value PCEPRTID or PCEPUNID in the second byte of field PCEID.

The \$PPPWORK mapping is used for printers in JES mode. The PCE work area for printers in FSS mode is mapped by \$FSSWORK.

This PCE is device related. This processor type has a one-to-one relationship to devices. Field PCEDCT points to a Device Control Table (DCT) and field DCTPCE in that DCT points to this PCE.

### \$PPPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	HASP PRINT/PUNCH PROCESSOR
312	(138)	BITSTRING	1	PPPFLAG1	PRINT/PUNCH FLAG BYTE
		1... ....		PPP1WRTS	"X'80" PRINT/PUNCH WRITE SWITCH
		.1.. ....		PPP1SUSP	"X'40" PRINT/PUNCH SUSPEND SWITCH
		..1. ....		PPP1NSKP	"X'20" No Skip-to-channel-1
		...1 ....		PPP1TERM	"X'10" PRINT/PUNCH TERMINATION SWITCH
		.... 1...		PPP1FNCI	"X'08" PUNCH INTERPRET REQUESTED

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... .1..		PPP1DRER	"X'04" PRINT/PUNCH DATA READ ERROR
		.... .1.		PPP1JIRE	"X'02" PRINT/PUNCH JCT/IOT READ ERROR
313	(139)	BITSTRING	1	PPFFLAG2	PRINT/PUNCH FLAG BYTE
		1... ....		PPP2TCEL	"X'80" TRACK-CELL DE-SPOOLING SWITCH
		.1.. ....		PPP2READ	"X'40" PRINT/PUNCH READ SWITCH
		.1. ....		PPP2CKPN	"X'20" PRINT/PUNCH CKPT-NEEDED SWITCH
		...1 ....		PPP2CKPA	"X'10" PRINT/PUNCH CKPT-ALLOWED SWITCH
		.... 1...		PPP2PCIW	"X'08" PRINT/PUNCH PCI WAIT SWITCH
		.... .1..		PPP2OPTJ	"X'04" PRINTER OPTCD=J SWITCH
		.... .1.		PPP2FDS	"X'02" FIRST SYSOUT DATA SET SWITCH
		.... ...1		PPP2SMFE	"X'01" DATA BUFFER ERROR FLAG FOR SMF
314	(13A)	BITSTRING	1	PPFFLAG3	PRINT/PUNCH FLAG BYTE 3
		1... ....		PPP3TRNC	"X'80" TRUNCATE OUTPUT
		.1.. ....		PPP3RECV	"X'40" RECOVERY IN PROCESS
		.1. ....		PPP3NAVL	"X'20" DEVICE NO LONGER AVAILABLE
		...1 ....		PPP3CK38	"X'10" 3800 CHECKPOINT FLAG
		.... 1...		PPP3CKRP	"X'08" 3800 REPOSITION BIT
		.... .1..		PPP3CKSU	"X'04" 3800 PPQ SUSPEND BIT
		.... .1.		PPP3CKRS	"X'02" 3800 RESTART BIT G38E
		.... ...1		PPP3INIT	"X'01" FIRST USE BIT
315	(13B)	BITSTRING	1	PPPS6DCI	PRINT/PUNCH SMF FLAGS (SMF6DCI)
316	(13C)	BITSTRING	1	PPPDCTFL	PR/PU/RMT DCT FLAGS (DCTFLAGS)
317	(13D)	BITSTRING	1	PPPBFOPT	PRINT/PUNCH BUFFERING OPTION
318	(13E)	SIGNED	2		Reserved
320	(140)	SIGNED	4	PPPUCB	ADDRESS OF OUTPUT DEVICE UCB
324	(144)	SIGNED	4	PPPXTCW (0)	3800 SELECT-XTAB CCW OP-CODES
324	(144)	SIGNED	4	PPPUCCW	ADDRESS OF PUNCH ERROR CCW
328	(148)	SIGNED	8	PPPTIMON	PRT/PUN SIGN-ON TIME/DATE
336	(150)	SIGNED	4	PPBFADR	ADDR OF 1ST DATA BUFFER CHAIN
340	(154)	SIGNED	4	PPBFSAV	ADDR OF NEXT DATA BUFFER CHAIN
344	(158)	SIGNED	4	PPPJCTBF	ADDRESS OF JCT BUFFER
348	(15C)	SIGNED	4	PPPLCCWA	ADDRESS OF LAST CCW
352	(160)	SIGNED	2	PPPLCCWO	OFFSET TO LAST CCW IN CHAIN
354	(162)	BITSTRING	1	PPPRMSEL	SNA REMOTE SELECT BYTE SAVE AREA
355	(163)	BITSTRING	1	PPP3211I	3211 INDEX VALUE
356	(164)	BITSTRING	1	PPFFLAG4	PRINT/PUNCH FLAG BYTE 4
		1... ....		PPP4PS38	"X'80" 3800 PATH SET INDICATOR
		.1.. ....		PPP43081	"X'40" WCS PATH INDICATOR
		.1. ....		PPP4EX15	"X'20" EXIT 15 SWITCH
		...1 ....		PPP4RPBS	"X'10" REPOSITION DUE TO BSPACE
		.... 1...		PPP4QSMF	"X'08" \$QUESMFB INDICATOR
		.... .1..		PPP4FPRD	"X'04" \$F PRTN,D IN PROGRESS
		.... .1.		PPP4FAUT	"X'02" A PDDDB within the work JOE failed the authorization check
		.... ...1		PPP4CALL	"X'01" All skips to channel treated as new page
357	(165)	BITSTRING	1	PPFFLAG5	PRPU FLAG BYTE 5, HASP185 USE
		1... ....		PPP5DSOP	"B'10000000" DATASET OPEN, NOT CLOSED
		.1.. ....		PPP5JIBK	"B'01000000" INVALID JOB KEY FOR BUFFER
		.1. ....		PPP5IDSK	"B'00100000" INVALID DATASET KEY FOR BUFFER
		...1 ....		PPP5IOER	"B'00010000" I/O ERROR ON CNTRL BLOCK READ
		.... 1...		PPP5IMTT	"B'00001000" INVALID TRACK ADDR FOR READ
		.... .1..		PPP5OJOE	"B'00000100" PRINT/PUNCH OBSOLETE JOE SWITCH
		.... .1.		PPP5AUTF	"B'00000010" AUTHORIZATION FAILED
		.... ...1		PPP5T185	"B'00000001" ERROR ENCOUNTERED WHEN DATASET WAS CLOSED
358	(166)	BITSTRING	1	PPFFLAG6	Print/Punch Flag byte 6
		1... ....		PPP6LERR	"B'10000000" Logical error during SWBTUREQ service
		.1.. ....		PPP6NEWS	"B'01000000" JESNEWS requested and available
		.1. ....		PPP6SWBD	"B'00100000" Unsuccessful I/O for SWBITs
		...1 ....		PPP6NODS	"B'00010000" No data set found in IOT
		.... 1...		PPP6BFER	"B'00001000" Large SMF buffer is too small to hold type 6 with SWBTU
		.... .1..		PPP6DSMR	"B'00000100" PDSMRSWB routine entered
		.... .1.		PPP6MRGR	"B'00000010" \$SWBMERG invoked

# \$PPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... ...1		PPP6ABND	"B'0000001" Abend in PDSMRSWB and/or \$SWBMERG service
359	(167)	BITSTRING	1	PPPFLAG7	Print/Punch Flag byte 7
		1... ....		PPP7RTRI	"B'1000000" Recovering from abend in \$SWBMERG service
		.1.. ....		PPP7SMRC	"B'0100000" Abend in \$SWBMERG cleanup call
		..1. ....		PPP7SMFU	"B'0010000" SMF RECORD UPDATED
		...1 ....		PPP7TRAN	"B'0001000" Translate lines
		.... 1...		PPP71PPF	"B'00001000" In case of impact printer restarting from backspace
		.... .1..		PPP7TRLR	"B'00000100" Printing job trailer page
360	(168)	BITSTRING	1		RESERVED FOR FUTURE USE
361	(169)	BITSTRING	1	PPPSAVDF	WORK AREA FOR SPACING FLAG
362	(16A)	BITSTRING	1	PPPJOEFL	COPY OF JOEFLAG1
363	(16B)	BITSTRING	2	PPPFCEBMP	FCB BIT MAP
365	(16D)	BITSTRING	1	PPPERCNT	PERM ERP ERROR COUNT
366	(16E)	SIGNED	2	PPPBFLRC	NO. OF LRC WITHIN CURRENT BUFFR
368	(170)	SIGNED	4	PPPRCBSV	RCB WORK AREA
376	(178)	DBL WORD	8	PPPCWWRK	CCW ANALYSIS SAVE AREA
384	(180)	SIGNED	4	PPPNUMR	PAGE NUMBER FOR RE-POSITIONING
388	(184)	BITSTRING	1	PPBUBFRO	BUFFER OFFSET FOR RE-POSITIONING
389	(185)	BITSTRING	1	PPBUBFAV	COUNT OF AVAILABLE INPUT BUFFERS
390	(186)	BITSTRING	1	PPDSCPY	DATA SET COPY COUNT
391	(187)	BITSTRING	1	PPPDIRI	PDIR IDENTIFIER
392	(188)	ADDRESS	4	PPCHKBF	ADDRESS OF CHK BUFFER
396	(18C)	ADDRESS	4	PPSESSA	Address of the ES section in the SMF type 24 record
400	(190)	SIGNED	2	PPNSWB	Number of SWBITS for a Pddb
402	(192)	SIGNED	2	PPPTUSUM	Total size of all SWBTUs
404	(194)	BITSTRING	1	PPPRESP	Copy of the response byte returned from Exit 1 & 15
405	(195)	BITSTRING	1	PPSEPPG	Copy of \$SEPPAGE from HCT
406	(196)	SIGNED	2	PPSPCTR	Separator page counter
408	(198)	BITSTRING	6	PPPIOTK	Track address (MQTR) of the IOT for the 1st Pddb of the JOE
414	(19E)	BITSTRING	2		Reserved
416	(1A0)	ADDRESS	4	PPPIOTA	Buffer address of the IOT for the 1st Pddb of the JOE
420	(1A4)	SIGNED	4	PPFPDDB	Address of the 1st Pddb of the JOE
424	(1A8)	SIGNED	4	PPPSWTR	Address of the 1st SWBIT MTR of the JOE
428	(1AC)	SIGNED	4	PPPSWBIT	Address of SWBIT chain
432	(1B0)	SIGNED	4	PPPTURET	Address of TU Pointer List for SWBTUREQ RETRIEVE
436	(1B4)	SIGNED	4	PPTUSPL	Address of TU Pointer List for SWBTUREQ SPLICE
440	(1B8)	SIGNED	4	PPPKYLST	Address of the Key List used for SWBTUREQ SERVICE
444	(1BC)	SIGNED	4	PPPTUADR	Address of TU output area used for SWBTUREQ SERVICE
448	(1C0)	ADDRESS	4	PPJSWBT	Address of JOE SWBIT chain
452	(1C4)	SIGNED	2	PPPMOSZ	Modify SWBTU size
454	(1C6)	SIGNED	2	PPMESZ	Merged SWBTU size
456	(1C8)	ADDRESS	4	PPMEST	Merged SWBTU address
460	(1CC)	ADDRESS	4	PPSBMPL	Address of \$SWBMERG parameter list
464	(1D0)	SIGNED	2	PPPTUSZ	Size of the TU output area used for SWBTUREQ SERVICE
466	(1D2)	SIGNED	2	PPPADNUM	Number of lines of ADDRESS
468	(1D4)	ADDRESS	4	PPPIOB2	Save field for the IIOB
472	(1D8)	ADDRESS	4	PPPBFSV2	Buffer save field
476	(1DC)	ADDRESS	4	PPBFAD2	Buffer save field
480	(1E0)	SIGNED	4	PPSEgid	Segment ID for SPIN data sets
484	(1E4)	ADDRESS	4	PPXPARM	Exit point parameter list
488	(1E8)	BITSTRING	6	PPPWKPTN	Segment number work area
494	(1EE)	BITSTRING	2	PPPSWBRC	Logical error return code
496	(1F0)	BITSTRING	4	PPPSWBRS	Logical error reason code
500	(1F4)	BITSTRING	1	PPPHRSN	JOE hold reason
501	(1F5)	BITSTRING	3		Reserved for future use
504	(1F8)	CHARACTER	36	PPPERMSG	Logical error message
540	(21C)	ADDRESS	4	PPPIOTAD	Address of IOT buffer save area
544	(220)	CHARACTER	80	PPBUFVSV	PCOMMENT WORK AREA
544	(220)	DBL WORD	8	PPSSRWK	SEPARATOR SERVICE RTN WORK

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
-----					
BEGIN PRINT/PUNCH CHECKPOINT DATA AREA. THESE FIELDS MUST CORRESPOND EXACTLY TO THE CHKJESWK FIELDS MAPPED IN THE \$CHK DSECT.					
-----					
End of Comment					
624	(270)	SIGNED	4	(0)	FORCE FULLWORD ALIGNMENT
624	(270)	SIGNED	2	PPPEBR CB	NUMBER OF LAST EJECT RCB
626	(272)	SIGNED	2	PPPDD B DS	CURRENT PDD B DISPLACEMENT
628	(274)	SIGNED	4	PPPDD B PC	CURRENT PDD B PAGE COUNTER
632	(278)	SIGNED	4	PPPLCC NT	CURRENT LINE OR CARD COUNT
636	(27C)	SIGNED	4	PPPPAG CT	CURRENT PAGE COUNT (PHYSICAL)
640	(280)	SIGNED	4		Unused - for alignment
644	(284)	SIGNED	4		Unused - for alignment
648	(288)	BITSTRING	1	PPPCOP CT	COPY COUNTER
649	(289)	BITSTRING	1	PPPLEBO F	LAST EJECT BUFFER OFFSET
650	(28A)	BITSTRING	1	PPPCGC NT	CURRENT COPY GROUP COUNT
651	(28B)	BITSTRING	1	PPPDSC T	CURRENT DATA SET COUNT
Comment					
KEEP NEXT TWO FIELDS TOGETHER FOR \$DU COMMAND					
End of Comment					
652	(28C)	SIGNED	4	PPPRECN M	CURRENT RECORD NUMBER
656	(290)	SIGNED	4	PPPPAG NM	CURRENT PAGE NUMBER
660	(294)	CHARACTER	13		Unused - for alignment
673	(2A1)	BITSTRING	1	PPPCVER S	CHK version - see \$CHK
674	(2A2)	SIGNED	2	PPPBPH PC	PDD B PHYSICAL PAGE COUNT
676	(2A4)	BITSTRING	6	PPPLEB TK	LAST EJECT BUFFER MQTR
682	(2AA)	BITSTRING	6	PPPIOT TK	CURRENT IOT MQTR
682	(2AA)	X'270'	0	PPPCCK PT D	"PPPEBR CB,*-PPPEBR CB" CKPT-DATA EQUATE AND LENGTH
Comment					
-----					
END PRINT/PUNCH CHECKPOINT DATA AREA.					
-----					
End of Comment					
688	(2B0)	SIGNED	4	(0)	ESTABLISH THE
680	(2A8)	SIGNED	4	PPPDAD CT (0)	USING STORAGE FOR THE DCT FIELDS NOT IN A DA DCT
688	(2B0)	SIGNED	4	(0)	DA DCT FOR \$EXCP
716	(2CC)	SIGNED	4	PPPDEV TP	PRINT/PUNCH DEVICE TYPE
716	(2CC)	X'2CF'	0	PPPDEV B1	"PPPDEVTP+3,1" UCB BYTE4 HAS 1BYTE DEV CODE
716	(2CC)	X'2CE'	0	PPPDEV B2	"PPPDEVTP+2,1" UCB BYTE3 HAS 1BYTE CODE
720	(2D0)	SIGNED	4	PPPLSA VE	LINK REGISTER SAVE WORD
724	(2D4)	SIGNED	4	PPPLNE CT	MAXIMUM LINES PER PAGE
728	(2D8)	SIGNED	4	PPPPLC	PRINT/PUNCH PAGE LINE COUNTER
732	(2DC)	BITSTRING	2	PPPPLC2	Continuation of PPPPLC for when we use PPPPLC as an MQTR save area.
734	(2DE)	SIGNED	2	PPPCCK PPS	LOG PAGES/CKPT SAVE AREA
736	(2E0)	SIGNED	2	PPPCCK PP	LOG PAGES/CKPT COUNTER
738	(2E2)	SIGNED	2	PPPCCK PLS	LINES/LOG PAGE SAVE AREA
740	(2E4)	SIGNED	2	PPPCCK PL	LINES/LOG PAGE COUNTER
744	(2E8)	ADDRESS	4	PPPWKJO E	Address of real work JOE
748	(2EC)	ADDRESS	4	PPPJOA A	Address of JOA that corresponds to PPPWKJOE
752	(2F0)	CHARACTER	8	PPPWKFR M	WORK AREA FOR DEVICE FORMS
760	(2F8)	CHARACTER	0	PPPWKWF M (0)	BACKUP AREA FOR LIST OF FORMS

# \$PPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
824	(338)	CHARACTER	8	PPPBKWK (0)	WORK AREA FOR BLOCK ROUTINE
824	(338)	CHARACTER	4	PPPWKFCB	WORK AREA FOR DEVICE FCB
828	(33C)	CHARACTER	4	PPPWKUCS	WORK AREA FOR DEVICE UCS
832	(340)	SIGNED	4	PPPBSFCT	BSP FRAME PAGE COUNTER
-----					
Comment					
-----					
NOTE: Keep PPPCCWPU,PPPPURCB, and PPPUBFO together					
-----					
End of Comment					
836	(344)	SIGNED	4	PPPLNDSP (0)	PRT BLK LETTER NEW LINE DISPLT
836	(344)	BITSTRING	6	PPPCCWPU	MQTR OF LAST PU CCW
842	(34A)	SIGNED	2	PPPLNCTR (0)	PRT BLK LETTER LINE COUNTER
842	(34A)	BITSTRING	2	PPPPURCB	RCB OF LAST PU CCW
844	(34C)	BITSTRING	1	PPPPUBFO	BUFFER OFFSET OF LAST PU CCW
845	(34D)	BITSTRING	1	PPPL3625	LAST 3525 PRINT LINE COMMAND
846	(34E)	BITSTRING	1	PPPRSC38	MOST RECENT 3800 SELECT CCW
847	(34F)	BITSTRING	1	PPPTRC38	MAXIMUM 3800 TABLE REF CHAR VALUE
848	(350)	CHARACTER	80	PPPMMSG	MESSAGE WORK AREA
848	(350)	X'350'	0	PPPMFRM	"PPPMMSG,L'PDBFORMS" SAVE AREA FOR FORMS
848	(350)	X'358'	0	PPPMFCB	"PPPMMSG+L'PDBFORMS,L'PDBFCB" SAVE AREA FOR FCB
848	(350)	X'35C'	0	PPPMUCS	"PPPMMSG+L'PDBFORMS+L'PDBFCB,L'PDBUCS" SAVE AREA UCS
928	(3A0)	CHARACTER	8	PPPSFORM	SETUP FORM NUMBER
936	(3A8)	CHARACTER	4	PPPSFCB	SETUP FCB IMAGE
940	(3AC)	CHARACTER	4	PPPSCHR1	SETUP TRANSLATE TABLE 1
944	(3B0)	CHARACTER	4	PPPSCHR2	SETUP TRANSLATE TABLE 2
948	(3B4)	CHARACTER	4	PPPSCHR3	SETUP TRANSLATE TABLE 3
952	(3B8)	CHARACTER	4	PPPSCHR4	SETUP TRANSLATE TABLE 4
956	(3BC)	CHARACTER	4	PPPSFLSH	SETUP FLASH ID
960	(3C0)	CHARACTER	4	PPPSMODI	SETUP MODIFICATION IMAGE
964	(3C4)	BITSTRING	1	PPPSFLSC	SETUP FLASH COUNT
965	(3C5)	BITSTRING	1	PPPSMODT	SETUP MODIFICATION TRC
966	(3C6)	BITSTRING	1	PPPSSCOP	SETUP STARTING COPY NUMBER
967	(3C7)	BITSTRING	1	PPPSCOPN	SETUP NO. COPIES THIS XMISSION
968	(3C8)	BITSTRING	1	PPPSFL38	3800 SETUP FLAG
		1... ....		PPPSBRST	"B'10000000" SETUP BURST=YES FLAG
		.1.. ....		PPPSREXM	"B'01000000" RETRANSMISSION FLAG
		..1. ....		PPPSSEP	"B'00100000" SETUP FOR SEPARATORS
		...1 ....		PPPSNFLS	"B'00010000" SETUP FLASH=NO FLAG
		.... 1...		PPPSPCMD	"B'00001000" SETUP FOR CMD MSG
969	(3C9)	BITSTRING	1		Reserved for future use
970	(3CA)	CHARACTER	8	PPPKEY (0)	JOB KEY AND DATA SET KEY
970	(3CA)	CHARACTER	4	PPPJKEY	JOB KEY FROM JCT
974	(3CE)	CHARACTER	4	PPPDSKEY	DATA SET KEY FROM Pddb
978	(3D2)	BITSTRING	2	PPPDSORT	Sort key generation field (low order two bytes of PDBCRTME)
980	(3D4)	BITSTRING	2	PPPDSINS	Instance id from Pddb
984	(3D8)	SIGNED	4	PPPFBPgc	\$/B PAGE COUNT
988	(3DC)	SIGNED	4	PPPOIOB	OUTPUT IOB ADDRESS
992	(3E0)	SIGNED	4	PPPIIOB	INPUT IOB ADDRESS
996	(3E4)	SIGNED	4	PPPOCCWP	PRIMARY OUTPUT CCW AREA
1000	(3E8)	SIGNED	4	PPPOCCWS	SECONDARY OUTPUT CCW AREA
1004	(3EC)	BITSTRING	6	PPPNXTCL	MQTR OF NEXT TRAKCELL
1010	(3F2)	BITSTRING	2		Reserved
1012	(3F4)	SIGNED	4	PPPTRBFT	ADDR OF MTRR/BUFFER ADDR TABLE
1016	(3F8)	SIGNED	4	PPPCkPTP	CHECKPOINT DATA POINTER
1020	(3FC)	ADDRESS	4	PPPIMDTE	ADDR OF HASPIMAG TASK DTE
1024	(400)	SIGNED	4	PPPRATAD (0)	SAVE AREA FOR RAT ADDRESS
1024	(400)	SIGNED	4	PPPPQH38	3800 PAGE QUEUE HEADER ADR G38E
1028	(404)	SIGNED	4	PPPSRCT	DATA SET RECORD COUNT
1032	(408)	SIGNED	4	PPPNBLK	NUMBER OF DATA BLOCKS READ
1036	(40C)	SIGNED	4	PPPSMFBF	SMF BUFFER SAVE AREA

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
1040	(410)	SIGNED	4	PPPSMFPQ	SMF BUFS ON PAGE QUEUE
1044	(414)	BITSTRING	1	PPPPSC38	PREVIOUS 3800 STARTING COPY NO.
1045	(415)	BITSTRING	1	PPPPCC38	PREVIOUS 3800 COPY COUNT
1046	(416)	BITSTRING	8	PPPCGR28	3800 COPY GROUPS DISTRIBUTION
1054	(41E)	BITSTRING	1	PPPFCL38	3800 FLASH COUNT
1055	(41F)	BITSTRING	1	PPPPRFLC	PREVIOUS FLASH COUNT
1056	(420)	CHARACTER	28	PPPBSPT	FOUR ENTRY BACKSPACE TABLE
1084	(43C)	ADDRESS	4	PPPBUFER	BUFFER OBTAINED IN \$#GET

Comment

PARAMETER LIST FOR PRTAUTH ROUTINE CALLED FROM HASPPRPU.  
THIS MATCHES THE ONE DEFINED IN \$FSACB.

End of Comment

1088	(440)	SIGNED	4	PPPAPARM (0)	PARAM LIST FOR PRTAUTH
1088	(440)	ADDRESS	4	PPPJCTAD	JCT ADDRESS
1092	(444)	ADDRESS	4	PPPPDDBA	Pddb ADDRESS
1096	(448)	ADDRESS	4	PPPANEWS	JESNEWS ADDRESS
1100	(44C)	CHARACTER	40	PPPLGOST	LOG STRING (ENTITY NAME WITH LENGTH IN THE FIRST BYTE)
1140	(474)	ADDRESS	4		RESERVED FOR FUTURE USE
1140	(474)	X'38'	0	PPPAPLEN	**-PPPAPARM" LENGTH OF PRTAUTH PARMLIST END OF PRTAUTH PARM LIST

Comment

Work area for ASAXWC macros  
MACDATE -06/16/09-<0>

End of Comment

0	(0)	X'478'	0	M00M1296	"PPPLIST" ++ ASAXWC NAME
1144	(478)	SIGNED	4	PPPLIST (0)	++ ASAXWC PARM LIST
1144	(478)	CHARACTER	4	PPPLIST_XPARAMAREA1	++ FIELD_LABEL
1148	(47C)	CHARACTER	24	PPPLIST_XPARAMAREA2	++ FIELD_LABEL
1148	(47C)	X'494'	0	PPPLIST_PL_END	*** ++ END OF BASE PLIST
1144	(478)	ADDRESS	4	PPPLIST_XPATTERNSTR_ADDR	++ ADDR
1148	(47C)	SIGNED	4	PPPLIST_XPATTERNSTRLEN	++
1152	(480)	ADDRESS	4	PPPLIST_XSTRING_ADDR	++ ADDR
1156	(484)	SIGNED	4	PPPLIST_XSTRINGLEN	++
1160	(488)	ADDRESS	4	PPPLIST_XZEROORMORE_ADDR	++ ADDR
1164	(48C)	ADDRESS	4	PPPLIST_XONECHAR_ADDR	++ ADDR
1168	(490)	ADDRESS	4	PPPLIST_XDELIMITER_ADDR	++ ADDR
1144	(478)	ADDRESS	4	PPPLIST_XPPPATTERNINFO_ADDR	++ ADDR
1148	(47C)	ADDRESS	4	PPPLIST_XPPPATTERNSTR_ADDR	++ ADDR
1152	(480)	SIGNED	4	PPPLIST_XPPPATTERNSTRLEN	++
1156	(484)	ADDRESS	4	PPPLIST_XPPZEROORMORE_ADDR	++ ADDR
1160	(488)	ADDRESS	4	PPPLIST_XPPONECHAR_ADDR	++ ADDR
1164	(48C)	ADDRESS	4	PPPLIST_XPPDELIMITER_ADDR	

## \$PPPWORK Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
1148	(47C)	ADDRESS	4	PPPLIST_XPPSTRING_ADDR	++ ADDR
1152	(480)	SIGNED	4	PPPLIST_XPPSTRINGLEN	++ ADDR
1172	(494)	X'1C'	0	PPPLISTL	++ "-PPPLIST" ++ LENGTH OF PLIST
Comment					
ASAXWC-0					
End of Comment					
1172	(494)	BITSTRING	256	PPPAREA	Work area passed to ASAXWC
1172	(494)	X'494'	0	PPPWRK18	"PPPAREA,18,C'C" 18-byte work area
1428	(594)	SIGNED	4	PPPOILNL (3)	OIL/NIL regs save area
1440	(5A0)	DBL WORD	8	(0)	
1440	(5A0)	X'468'	0	PPPLEN	"*-PCEWORK" SIZE OF PPU PCE WORK AREA

## \$PPPWORK Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
M00M1296	0	478	PPPDSCCT	28B	
PCE	0		PPPDSSINS	3D4	
PPPADNUM	1D2		PPPDSSKEY	3CE	
PPPANEWS	448		PPPDSSORT	3D2	
PPPAPARM	440		PPPDSSRCT	404	
PPPAPLEN	474	38	PPPEBRCB	270	
PPPAREA	494		PPPERCNT	16D	
PPPBFADR	150		PPPERMSG	1F8	
PPPBFAD2	1DC		PPPFBPGC	3D8	
PPPBFRLC	16E		PPPFCEBMP	16B	
PPPBFOPF	13D		PPPFIOF	1A0	
PPPBFSAV	154		PPPFIOFCK	198	
PPPBFV2	1D8		PPPFLAG1	138	
PPPBLKWK	338		PPPFLAG2	139	
PPPBPBPC	2A2		PPPFLAG3	13A	
PPPBSPT	340		PPPFLAG4	164	
PPPBSPT	420		PPPFLAG5	165	
PPPBUFAV	185		PPPFLAG6	166	
PPPBUFER	43C		PPPFLAG7	167	
PPPBUFRO	184		PPPFLC38	41E	
PPPBUFSV	220		PPPFDDDB	1A4	
PPPCCWPU	344		PPPFWSWTR	1A8	
PPPCCWRK	178		PPPHRSN	1F4	
PPPCGCNT	28A		PPPIIOB	3E0	
PPPCGR28	416		PPPIIOB2	1D4	
PPPCHKBF	188		PPPIMDTE	3FC	
PPPCKPL	2E4		PPPIOTAD	21C	
PPPCKPLS	2E2		PPPIOTTK	2AA	
PPPCKPP	2E0		PPPJCTAD	440	
PPPCKPPS	2DE		PPPJCTBF	158	
PPPCKPTD	2AA	270	PPPJKEY	3CA	
PPPCKPTP	3F8		PPPJOAA	2EC	
PPPCOPCT	288		PPPJOEFL	16A	
PPPCVERS	2A1		PPPJSWBT	1C0	
PPPDADCT	2A8		PPPKEY	3CA	
PPPDCTFL	13C		PPPKYLST	1B8	
PPPDDBDS	272		PPPLCCNT	278	
PPPDDBPC	274		PPPLCCWA	15C	
PPPDEVB1	2CC	2CF	PPPLCCWO	160	
PPPDEVB2	2CC	2CE	PPPLEBOF	289	
PPPDEVTP	2CC		PPPLEBTK	2A4	
PPPDSCPYP	186		PPPLEN	5A0	468



Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PPPLIST	478		PPPPNUMR	180	
PPPLIST_PL_END	47C	494	PPPPQH38	400	
PPPLIST_XDELIMITER_ADDR	490		PPPPRFLC	41F	
PPPLIST_XONECHAR_ADDR	48C		PPPPSC38	414	
PPPLIST_XPARAMAREA1	478		PPPPUBFO	34C	
PPPLIST_XPARAMAREA2	47C		PPPPURCB	34A	
PPPLIST_XPATTERNSTR_ADDR	478		PPPRATAD	400	
PPPLIST_XPATTERNSTRLEN	47C		PPPRCBSV	170	
PPPLIST_XPPDELIMITER_ADDR	48C		PPPRECNM	28C	
PPPLIST_XPPONECHAR_ADDR	488		PPPRESP	194	
PPPLIST_XPPPATTERNINFO_ADDR	478		PPPRMSEL	162	
PPPLIST_XPPPATTERNSTR_ADDR	47C		PPPRSC38	34E	
PPPLIST_XPPPATTERNSTRLEN	480		PPPSAVDF	169	
PPPLIST_XPPSTRING_ADDR	47C		PPPSBMPL	1CC	
PPPLIST_XPPSTRINGLEN	480		PPPSBRST	3C8	80
PPPLIST_XPPZEROORMORE_ADDR	484		PPPSCHR1	3AC	
PPPLIST_XSTRING_ADDR	480		PPPSCHR2	3B0	
PPPLIST_XSTRINGLEN	484		PPPSCHR3	3B4	
PPPLIST_XZEROORMORE_ADDR	488		PPPSCHR4	3B8	
PPPLISTL	494	1C	PPPSGOPN	3C7	
PPPLNCTR	34A		PPPSEGID	1E0	
PPPLNDSP	344		PPPSEPPG	195	
PPPLNECT	2D4		PPPSSESA	18C	
PPPLLOGST	44C		PPPSFCB	3A8	
PPPLSAVE	2D0		PPPSFLSC	3C4	
PPPL3625	34D		PPPSFLSH	3BC	
PPPMEST	1C8		PPPSFL38	3C8	
PPPMESZ	1C6		PPPSFORM	3A0	
PPPMOSZ	1C4		PPPSMFBF	40C	
PPPMSFCB	350	358	PPPSMFPQ	410	
PPPMSFRM	350	350	PPPSMODI	3C0	
PPPMSG	350		PPPSMODT	3C5	
PPPMSUCS	350	35C	PPPSNFLS	3C8	10
PPPNBLK	408		PPPSPCMD	3C8	8
PPPNSWB	190		PPPSPCTR	196	
PPPNXTCL	3EC		PPPSREXM	3C8	40
PPPOCCWP	3E4		PPPSSEXP	3C6	
PPPOCCWS	3E8		PPPSSEP	3C8	20
PPPOILNL	594		PPPSRWK	220	
PPPOIOB	3DC		PPPSWBIT	1AC	
PPPPAGCT	27C		PPPSWBRC	1EE	
PPPPAGNM	290		PPPSWBRS	1F0	
PPPPCC38	415		PPPS6DCI	13B	
PPPPDDBA	444		PPPTIMON	148	
PPPPDIRI	187		PPPTRBFT	3F4	
PPPPLC	2D8		PPPTRC38	34F	
PPPPLC2	2DC		PPPTUADR	1BC	
			PPPTURET	1B0	
			PPPTUSPL	1B4	
			PPPTUSUM	192	
			PPPTUSZ	1D0	
			PPPUCEB	140	
			PPPUCCW	144	
			PPPWKFCB	338	
			PPPWKFRM	2F0	
			PPPWKJOE	2E8	
			PPPWKPTN	1E8	
			PPPWKUCS	33C	
			PPPWKWM	2F8	
			PPPWK18	494	494
			PPPXPARM	1E4	
			PPPXTCW	144	
			PPP1DRER	138	4
			PPP1FNCI	138	8
			PPP1JIRE	138	2

## \$PPPWORK Cross Reference

Name	Hex Offset	Hex Value
PPP1NSKP	138	20
PPP1SUSP	138	40
PPP1TERM	138	10
PPP1WRTS	138	80
PPP2CKPA	139	10
PPP2CKPN	139	20
PPP2FDS	139	2
PPP2OPTJ	139	4
PPP2PCIW	139	8
PPP2READ	139	40
PPP2SMFE	139	1
PPP2TCEL	139	80
PPP3CKRP	13A	8
PPP3CKRS	13A	2
PPP3CKSU	13A	4
PPP3CK38	13A	10
PPP3INIT	13A	1
PPP3NAVL	13A	20
PPP3RECV	13A	40
PPP3TRNC	13A	80
PPP3211I	163	
PPP4CALL	164	1
PPP4EX15	164	20
PPP4FAUT	164	2
PPP4FPRD	164	4
PPP4PS38	164	80
PPP4QSMF	164	8
PPP4RPBS	164	10
PPP43081	164	40
PPP5AUTF	165	2
PPP5DSOP	165	80
PPP5IDSK	165	20
PPP5IJBK	165	40
PPP5IMTT	165	8
PPP5IOER	165	10
PPP5OJOE	165	4
PPP5T185	165	1
PPP6ABND	166	1
PPP6BFER	166	8
PPP6DSMR	166	4
PPP6LERR	166	80
PPP6MRGR	166	2
PPP6NEWS	166	40
PPP6NODS	166	10
PPP6SWBD	166	20
PPP7RTRI	167	80
PPP7SMFU	167	20
PPP7SMRC	167	40
PPP7TRAN	167	10
PPP7TRLR	167	4
PPP71PPF	167	8

---

## \$PQE Information

### \$PQE Programming Interface information

Programming Interface information

\$PQE

End of Programming Interface information

## Heading Information • \$PQE Map

### \$PQE Heading Information

**Common Name:** JES2 3800 Page Queue Entry  
**Macro ID:** \$PQE  
**DSECT Name:** PQE  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: Virtual and real storage are anywhere, above or below 16M, in private storage of the JES2 address space.  
**Size:** See PQELEN  
**Created by:** JES2 Print/Punch PCE for a 3800 printer  
**Pointed to by:** PQENEXT field of the PQE data area  
 PQEPREV field of the PQE data area  
 PQECPQED field of the PQE data area  
 PQEJNEXT field of the PQE data area  
 PQHFIRST field of the PQH data area  
 PQHLAST field of the PQH data area  
 PQHFREE field of the PQH data area  
 PQHOPQE field of the PQH data area  
 PQHTPQE field of the PQH data area  
 PQHPQEJ field of the PQH data area  
**Serialization:** Serialized under the JES2 TCB  
**Function:** The PQEs contain 3800 printer page information

### \$PQE Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PQE	3800 PAGE QUEUE ENTRY G38E
0	(0)	SIGNED	4	PQENEXT	ADDRESS OF NEXT PQE G38E
4	(4)	SIGNED	4	PQEPREV	ADDRESS OF PREVIOUS PQE G38E
8	(8)	SIGNED	4	PQEHDR	ADDRESS OF EXTENT HEADER
12	(C)	BITSTRING	1	PQETYPE	TYPE OF PQE G38E
		1... ....		PQEC	"B'10000000" CHECKPOINT PQE G38E
		.1.. ....		PQES	"B'01000000" SMF TYPE 6 PQE G38E
		..1. ....		PQEJ	"B'00100000" JOB START PQE G38E
		...1 ....		PQED	"B'00010000" DATA SET PQE G38E
13	(D)	BITSTRING	1	PQESPEC (0)	START OF SPECIFIC SECTION G38E

Comment

PQE FIELDS -- USED AS A CHECKPOINT ENTRY G38E

End of Comment

13	(D)	BITSTRING	1	PQECFLAG	CHECKPOINT PQE FLAG G38E
		1... ....		PQECFPG	"B'10000000" FIRST PAGE OF DATA SET G38E
		.1.. ....		PQECLPG	"B'01000000" LAST PAGE OF DATA SET G38E
		..1. ....		PQECBSP	"B'00100000" PQE SAVED FOR BACKSPACE G38E
14	(E)	SIGNED	2	PQECPGID	CHANNEL PAGE ID G38E
16	(10)	SIGNED	4	PQECSENS (0)	ADR OF SENSED ID AND FCBLN G38E
16	(10)	SIGNED	2	PQERPGID	REPOSITION PAGE ID G38E
18	(12)	SIGNED	2	PQEFCBLN	FCB LINE POSITION G38E
20	(14)	SIGNED	4	PQECPQED	ADDRESS OF DATA SET PQE G38E
24	(18)	SIGNED	4	PQECPPCT	PDDb LOGICAL PAGE COUNT G38E
28	(1C)	SIGNED	4	PQECTLNC	TOTAL JOE LINE COUNT G38E
32	(20)	SIGNED	4	PQECTPCT	TOTAL JOE PAGE COUNT G38E
36	(24)	BITSTRING	6	PQECMQTR	Track address (MQTR) of spool data.
42	(2A)	BITSTRING	2		Reserved

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
44	(2C)	SIGNED	4	PQECRECN	# RECORDS PRINTED SO FAR G38E
48	(30)	SIGNED	2	PQECJRCB	DISP INTO EJECT BUFFER G38E
Comment					
PQE FIELDS -- USED AS AN SMF TYPE 6 PQE G38E					
End of Comment					
13	(D)	BITSTRING	1		CHECKPOINT PQE FLAG G38E
14	(E)	SIGNED	2		CHANNEL PAGE ID G38E
16	(10)	SIGNED	2		REPOSITION PAGE ID G38E
18	(12)	SIGNED	2		FCB LINE POSITION G38E
20	(14)	SIGNED	4	PQESBUF	ADDRESS OF SMF BUFFER G38E
Comment					
PQE FIELDS -- USED AS A JOB START PQE G38E					
End of Comment					
13	(D)	BITSTRING	1		CHECKPOINT PQE FLAG G38E
14	(E)	SIGNED	2		CHANNEL PAGE ID G38E
16	(10)	SIGNED	2		REPOSITION PAGE ID G38E
18	(12)	SIGNED	2		FCB LINE POSITION G38E
20	(14)	SIGNED	4	PQEJWJOE	ADDRESS OF WORK JOE G38E
24	(18)	SIGNED	4	PQEJNEXT	ADR OF NEXT PQEJ ON QUEUE G38E
Comment					
PQE FIELDS -- USED AS A DATA SET PQE G38E					
End of Comment					
13	(D)	BITSTRING	1	PQEDCOPY	COPY NUMBER IN PROGRESS G38E
14	(E)	BITSTRING	1	PQEDCPYG	OFFSET INTO COPY GROUP G38E
15	(F)	BITSTRING	1	PQEDTNS	TOTAL JOE DATA SET COUNT G38E
16	(10)	SIGNED	4	PQEDWJOE	ADDRESS OF WORK JOE G38E
20	(14)	BITSTRING	6	PQEDIOTK	Current IOT track address (MQTR).
26	(1A)	BITSTRING	2		Reserved
28	(1C)	SIGNED	4	PQEDJBNO	Job number
32	(20)	BITSTRING	4	PQEDJKEY	JOB IDENTIFIER KEY G38E
36	(24)	SIGNED	2	PQEDPDDDB	DISP OF PDDDB INTO IOT G38E
38	(26)	BITSTRING	1	PQEDCGCT	COPY GROUP CT FOR DATA SET G38E
39	(27)	BITSTRING	1	PQEDFLAG	DATA SET PQE FLAG BYTE G38E
		1... ....		PQEDLAST	"B'10000000" LAST DATA SET OF JOE G38E
		.1.. ....		PQEDCAN	"B'01000000" JOE CANCELLED G38E
		..1. ....		PQEDINT	"B'00100000" JOE INTERRUPTED G38E
		...1 ....		PQEDRPT	"B'00010000" JOE REPEATED G38E
		.... 1..		PQEDRST	"B'00001000" JOE RESTARTED G38E
		.... .1..		PQEDALOC	"B'00000100" ALLOCATION IOT G38E
		.... ..1.		PQEDCJP	"B'00000010" JOE CANCELLED BY \$CJP
		.... ...1		PQEDADD	"B'00000001" JOE ADDED FOR \$EPRT
40	(28)	BITSTRING	8	PQEDCGRP	DATASET COPY GROUPS
48	(30)	BITSTRING	1	PQEDSCPY	DATASET COPY COUNT
49	(31)	BITSTRING	4	PQEDSKEY	DATASET KEY
53	(35)	BITSTRING	2	PQEDSINS	Dataset multi-dest instance
55	(37)	BITSTRING	2	PQEDSORT	Sort key generation field (low order two bytes of PDBCRTME)
57	(39)	CHARACTER	64	PQEDJCOR	Job Correlator
121	(79)	BITSTRING	1	PQEFLAGF	FREE PQE INDICATOR
		1111 1111		PQEFFREE	"X'FF" ON FREE QUEUE
124	(7C)	SIGNED	4	PQEEND (0)	END OF PQEC G38E
124	(7C)	X'C'	0	PQEDATA	"PQETYPE,PQEEND-PQETYPE" START OF PQE DATA
124	(7C)	X'7C'	0	PQELENG	"PQEEND-PQE" LENGTH OF PQEC G38E

## \$PQE Cross Reference

## \$PQE Cross Reference

Name	Hex Offset	Hex Value
PQE	0	
PQEC	C	80
PQECBSP	D	20
PQECFLAG	D	
PQECFPG	D	80
PQECJRCB	30	
PQECLPG	D	40
PQECMQTR	24	
PQECPGID	E	
PQECPPCT	18	
PQECPQED	14	
PQECRECN	2C	
PQECSENS	10	
PQECTLNC	1C	
PQECTPCT	20	
PQED	C	10
PQEDADD	27	1
PQEDALOC	27	4
PQEDATA	7C	C
PQEDCAN	27	40
PQEDCGCT	26	
PQEDCGRP	28	
PQEDCJP	27	2
PQEDCOPY	D	
PQEDCPYG	E	
PQEDFLAG	27	
PQEDINT	27	20
PQEDIOTK	14	
PQEDJBNO	1C	
PQEDJCOR	39	
PQEDJKEY	20	
PQEDLAST	27	80
PQEDPddb	24	
PQEDRPT	27	10
PQEDRST	27	8
PQEDSCPY	30	
PQEDSINS	35	
PQEDSKEY	31	
PQEDSORT	37	
PQEDTnds	F	
PQEDWJOE	10	
PQEEND	7C	
PQEFCBLN	12	
PQEFFREE	79	FF
PQEFLAGF	79	
PQEHDR	8	
PQEJ	C	20
PQENEXT	18	
PQEWJOE	14	
PQELENG	7C	7C
PQENEXT	0	
PQEPREV	4	
PQERPGID	10	
PQES	C	40
PQESBUF	14	
PQESPEC	D	
PQETYPE	C	

## \$PREBERT Information

### \$PREBERT Heading Information

**Common Name:** Prefix for BERT processing  
**Macro ID:** \$PREBERT  
**DSECT Name:** PREBERT and PBEUSER  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** Varies according to block being mapped  
 Offset: 0  
 Length: 4

**Storage Attributes:** Subpool: 0  
 Key: 1 or 8 (Depends upon whether the control following this prefix is READONLY or UPDATE  
 Residency: JES2 address space, above or below the line

**Size:** See PBESIZE and PBEUSIZE

**Created by:** Any service routine which in turn uses the \$DOGBERT service routine.

**Pointed to by:** \$PBELST field of the \$HCT data area points to chain of locked PREBERTs in UPDATE mode  
 PBEUSERS field of the PREBERT points to first PBEUSER  
 PBEUPBEU field of the PBEUSER points to next PBEUSER  
 Implicitly pointed to by anyone calling the \$DOGBERT service. The PREBERT is always in front of the block of memory specified via \$CBADDR keyword

**Serialization:** None

**Function:** The PREBERT is used by the \$DOGBERT service to control and record access to data in the BERT.  
 The PBEUSER is used by \$DOGJQE service to control stack of users using a given PREBERT in UPDATE mode.

### \$PREBERT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PREBERT	, Prefix for BERT processing
0	(0)	CHARACTER	4	PBEID	Identity of block following
4	(4)	BITSTRING	4	PBETOKEN	BERT token
8	(8)	BITSTRING	1	PBEUSERS	Owner information (see PBEUSER DSECT for mapping)
Comment					
<p>-----</p> <p>PBEPCE points to the PCE that owns this element.                      If the high bit is on in the word, that indicates that TERM should never free this if the PCE ABENDs.                      This is because the PCE may be serving multiple non-main task requestors.</p> <p>-----</p>					
End of Comment					
28	(1C)	ADDRESS	4	PBEPCE	PCE address or zero
28	(1C)	X'1C'	0	PBETCB	"PBEPCE,4,C'A" TCB addr if not main task
32	(20)		16	PBETOKEN	Owning TTOKEN (if \$JQESERV)
48	(30)	SIGNED	4	PBEINDEX	Index into CTENT for block
52	(34)	SIGNED	4	PBEINDEX2	2ndary control block index
56	(38)	ADDRESS	4	PBENEXT	Address of next PREBERT on PCE chain
60	(3C)	ADDRESS	4	PBEPREV	Address of previous PREBERT on PCE chain

## \$PREBERT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
64	(40)		16	PBEPWAIT	Time PCE last \$WAITed when \$DOGBERT called for FETCH - or - Time PCE ABENDED while owning the BERT lock
80	(50)	BITSTRING	1	PBEUSEQ	Update sequence field (managed by \$DOGBERT)
81	(51)	BITSTRING	1	PBEDGBF1	In line flag \$DGBFLAG moved here if \$DGB2PBE on
Comment					
DGBWAIT EQU B'10000000' \$WAIT is allowed DGBQSUS EQU B'01000000' Get the queues DGBUPDT EQU B'00100000' Update access needed DGBNAME EQU B'00010000' NAME= was passed DGBTOKN EQU B'00001000' TOKEN= was passed DGBSPEC EQU B'00000100' SPECIAL=YES was specified DGBNUPD EQU B'00000010' No write update requested DGBNRDD EQU B'00000001' No read data requested					
End of Comment					
82	(52)	BITSTRING	1	PBEDGBF2	In line flag \$DGBFLAG2 moved here if \$DGB2PBE on
Comment					
DGB2CRE EQU B'10000000' ACTION=(,CREATE) spec DGB2UNK EQU B'01000000' CB type unknown DGB2PAD EQU B'00100000' ACTION=(CKPT,PAD) DGB2PBE EQU B'00010000' Reserved DGB2NEV EQU B'00001000' MOREBERTS=NEVER					
End of Comment					
83	(53)	BITSTRING	1	PBEMVERS	\$MSTRVER level when built
84	(54)	BITSTRING	1	PBEFLAG1	Access flags (managed by DOGBERT, INIT, and TERM)
		1... ..		PBE1UPDT	"B'10000000" UPDATE mode
		.1.. ..		PBE1READ	"B'01000000" READ mode
		..1. ....		PBE1SPEC	"B'00100000" SPECIAL mode
		...1 ....		PBE1DOGB	"B'00010000" DOGBERT used to construct block
		.... 1..		PBE1NEWB	"B'00001000" New BERT was obtained
		.... .1..		PBE1ABND	"B'00000100" Owned by ABENDING PCE
		.... ...1		PBE1PAD	"B'00000001" Retain maximum BERTs
85	(55)	BITSTRING	1	PBEFLAG2	Flags (managed by services at a higher level than DOGBERT)
		1... ..		PBE2UPDT	"B'10000000" Data orig. update mode
		.1.. ..		PBE2PSEU	"B'01000000" Pseudo-BERT (does not contain real BERT data)
		..1. ....		PBE2FREB	"B'00100000" BERT lock freed via (MANAGELOCK,RELEASE)
		...1 ....		PBE2LONG	"B'00010000" GETWORK area includes JQO
		.... 1..		PBE2EMPT	"B'00001000" Block following is empty
		.... .1..		PBE2FREE	"B'00000100" Memory for block freed
		.... ...1		PBE2DSTK	"B'00000001" When creating a stack element, turn on PBE1PCE
Comment					
DOGDJB managed bits					
End of Comment					
		1... ..		PBE2CACH	"B'10000000" DJB in read cache
		.1.. ..		PBE2INV	"B'01000000" Cached DJB invalidated
		..1. ....		PBE2PXEQ	"B'00100000" Pretend job in family is executing
		...1 ....		PBE2CACE	"B'00010000" DJB eligible for caching
86	(56)	BITSTRING	1	PBE2TYPE	BERT type (same as BRTTYPE)
87	(57)	BITSTRING	1	PBEFLAG3	Flags (managed by services at a higher level than DOGBERT)
		1... ..		PBE3QPSY	"B'10000000" Perform QPOST when JQA returned no matter what
		.1.. ..		PBE3#PSY	"B'01000000" Perform \$#POST when JQA returned no matter what
		..1. ....		PBE3#KPJ	"B'00100000" \$#POST to be done with CLEAR_JWEL=NO
		...1 ....		PBE3#JOT	"B'00010000" \$#POST required to manage JOEVRTME



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... 1...		PBE3CTCI	"B'00001000" 'CAT Cache' element indicator. Denotes if an element in the 'CAT Cache' is a CAT(OFF) or a GRPOBJ(ON). See \$CAT 'CAT Cache' info.
		.... .1..		PBE3ALLQ	"B'00000100" If PBE2PSEU=ON, this indicates if \$DOGCAT/CATREAD is processing 'Pseudo CATs' ( NON- execution queues ) for ALLQUES=YES.
88	(58)	SIGNED	2	PBEDLEN	Size of the BERT data area
90	(5A)	BITSTRING	6		Reserved
90	(5A)	X'60'	0	PBEDATA	*** Beginning of actual data
90	(5A)	X'60'	0	PBESIZE	** -PREBERT"

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PBEUSER	, PREBERT user stack element
0	(0)	CHARACTER	4	PBEUID	Eye catcher
4	(4)	ADDRESS	4	PBEUPBEU	Address of prior user block
8	(8)	ADDRESS	4	PBEUPBE	Address of PREBERT
12	(C)	ADDRESS	4	PBEULINK	R14 value of \$DOGxxx caller
16	(10)	BITSTRING	1	PBEUFLG1	Flags
		1... ....		PBEU1USE	"B'10000000" User block in use
		.1.. ....		PBEU1PCE	"B'01000000" An existing PREBERT for this PCE was used
		..1. ....		PBEU1RD	"B'00100000" Caller wanted read mode
17	(11)	SIGNED	1	PBEUEXID	Exit ID in control when built (0 if IBM code)
18	(12)	BITSTRING	2		Reserved for future use
18	(12)	X'14'	0	PBEUSIZE	** -PBEUSER" Size of User block

**\$PREBERT Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PBEBTYPE	56		PBEU1USE	10	80
PBEDATA	5A	60	PBE1ABND	54	4
PBEDGBF1	51		PBE1DOGB	54	10
PBEDGBF2	52		PBE1NEWB	54	8
PBEDLEN	58		PBE1PAD	54	1
PBEFLAG1	54		PBE1READ	54	40
PBEFLAG2	55		PBE1SPEC	54	20
PBEFLAG3	57		PBE1UPDT	54	80
PBEID	0		PBE2CACE	55	10
PBEINDEX	30		PBE2CACH	55	80
PBEINDX2	34		PBE2DSTK	55	1
PBEMVERS	53		PBE2EMPT	55	8
PBENEXT	38		PBE2FREB	55	20
PBEPCE	1C		PBE2FREE	55	4
PBEPREV	3C		PBE2INV	55	40
PBEPWAIT	40		PBE2LONG	55	10
PBESIZE	5A	60	PBE2PSEU	55	40
PBETCB	1C	1C	PBE2PXEQ	55	20
PBETOKEN	4		PBE2UPDT	55	80
PBETOKN	20		PBE3#JOT	57	10
PBEUEXID	11		PBE3#KPJ	57	20
PBEUFLG1	10		PBE3#PSY	57	40
PBEUID	0	D7E4E2D9	PBE3ALLQ	57	4
PBEULINK	C		PBE3CTCI	57	8
PBEUPBE	8		PBE3QPSY	57	80
PBEUPBEU	4		PREBERT	0	
PBEUSEQ	50				
PBEUSER	0				
PBEUSERS	8				
PBEUSIZE	12	14			
PBEU1PCE	10	40			
PBEU1RD	10	20			

## \$PREBERT Cross Reference

---

## \$PRGWORK Information

### \$PRGWORK Programming Interface information

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$PRGWORK**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$PRGWORK Map

### \$PRGWORK Heading Information

**Common Name:** JES2 Purge PCE Work Area  
**Macro ID:** \$PRGWORK  
**DSECT Name:** PCE (\$PRGWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4  
**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE  
**Size:** See symbol PRGWLEN for the length of this work area. The overall length of the PCE is stored in field PCELENG.  
**Created by:** See \$PCE  
**Pointed to by:** The \$PURGPCE field of the \$HCT data area points into the \$PCEORG/\$PCELAST chain to the first Purge PCE. Since this chain contains all PCEs, use the PCEID field to determine when you have chained past the last PCE of this type. See \$PCE for other pointer fields that apply to all PCE types.  
**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this work area are used by a JES2 Purge Processor and by its support routines and exits. \$PRGWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$PRGWORK are actually part of the PCE DSECT, but only map PCEs with the value PCEPRGID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$PRGWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	ADDRESS	4	PRGWPRM	NODE TABLE ADDRESS
316	(13C)	ADDRESS	4		CONTROL BLOCK ADDRESS
320	(140)	ADDRESS	4		ADDRESS OF JQE
324	(144)	ADDRESS	1		QUEUE TYPE SPECIFIED
325	(145)	ADDRESS	1		WORK SELECTION TYPE FLAG
326	(146)	ADDRESS	1		RESERVED FOR FUTURE USE
326	(146)	X'138'	0	PRGWLST	"PRGWPRM,*-PRGWPRM" QGET PARAMETER LIST STORAGE
328	(148)	SIGNED	4	PRGBLD (0)	Control block ID
332	(14C)	BITSTRING	4		Console ID
336	(150)	ADDRESS	4		Address of the CART
340	(154)	ADDRESS	4		Pointer for JOBID
344	(158)	ADDRESS	4		Control block address
348	(15C)	ADDRESS	4		Display routine address
352	(160)	ADDRESS	4	(6)	6 word work area
376	(178)	ADDRESS	4		Caller's R11 value
380	(17C)	BITSTRING	2		ROUT code for Message
382	(17E)	BITSTRING	2		Not used
384	(180)	CHARACTER	4		Message ID

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
388	(184)	CHARACTER	1		Separator character
389	(185)	ADDRESS	1		Flag byte 1
390	(186)	ADDRESS	1		'DISPER'
391	(187)	ADDRESS	1		Flag byte 2
392	(188)	ADDRESS	1		Flag byte 3
393	(189)	CHARACTER	8		Symbolic name of dest.
401	(191)	BITSTRING	15		Not used
416	(1A0)	ADDRESS	4	(0)	Ensure multiple of 4
416	(1A0)	ADDRESS	2	(0)	
416	(1A0)	BITSTRING	1	PRGFLAG1	Flags
		1... ....		PRG1DUPL	"B'10000000" Job held for duplicate jobname at least once
		.1... ....		PRG1JCTR	"B'01000000" The JCT for the currently purging job has been read
		...1. ....		PRG1LSPN	"B'00100000" Job went thru unspun
		...1 ....		PRG1XWTR	"B'00010000" At lease one work JOE was removed by external dev.
		.... 1...		PRG1EXJO	"B'00001000" Exited JOE loop early
417	(1A1)	BITSTRING	1	PRGJQEF	JQEFLAG1 saved here
418	(1A2)	BITSTRING	1	PRGJQEF5	JQEFLAG5 saved here
419	(1A3)	BITSTRING	1		Reserved
420	(1A4)	ADDRESS	4	PRGSNFL	Anchor for MTT list
424	(1A8)	CHARACTER	64	PRGJCOR	Job correlator for SMF
488	(1E8)	DBL WORD	8	(0)	Ensure double word length
488	(1E8)	X'B0'	0	PRGWLEN	"*-PCEWORK" WORK AREA LENGTH

**\$PRGWORK Cross Reference**

Name	Hex Offset	Hex Value
PCE	0	
PRGBLD	148	C2D3C440
PRGFLAG1	1A0	
PRGJCOR	1A8	
PRGJQEF	1A1	
PRGJQEF5	1A2	
PRGSNFL	1A4	
PRGWLEN	1E8	B0
PRGWLST	146	138
PRGWPRM	138	
PRG1DUPL	1A0	80
PRG1EXJO	1A0	8
PRG1JCTR	1A0	40
PRG1LSPN	1A0	20
PRG1XWTR	1A0	10



## \$PSO Information

### \$PSO Heading Information

**Common Name:** HASP Process Sysout Work Area DSECT  
**Macro ID:** \$PSO  
**DSECT Name:** PSO  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** '\$PSO'  
 Offset: PSOID-PSO  
 Length: 4  
**Storage Attributes:** Subpool: n/a  
 Key: 1  
 Residency: In the jesxPSO data space  
**Size:** See PSOLNGTH  
**Created by:** Process Sysout Subsystem Interface code running in the requestor's address space.  
**Pointed to by:** SJBPSO field of the \$SJB data area  
 CCTPSO field of the \$HCCT data area  
 PSONEXT field of the \$PSO data area  
 MTRBPARAM field of the \$MTRB data area  
**Serialization:** Compare and Swap  
**Function:** The PSO contains an image of the IEFSSSO SSOB extension in order that data set selection for External Writers and the TSO OUTPUT command can be supported in the JES2 address space.

### \$PSO Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PSO	PSO PARAMETER LIST DSECT
0	(0)	CHARACTER	4	PSOID	Eye catcher
4	(4)	ADDRESS	4	PSONEXT	Addr of next PSO on queue
8	(8)	SIGNED	4	PSOCRDT	ELIGIBILITY CUT-OFF DATE
12	(C)	BITSTRING	1	PSOPFLAG	FLAG USED BY PSO PROCESSOR
		1... ....		PSOPFSWT	"B'10000000" READ BOTH IOT CHAINS
		.1... ....		PSOPFREQ	"B'01000000" NEW SELECTION REQUIREMENTS
		.... 1...		PSOPFHLD	"B'00001000" HOLD AT UNALLOC SPECIFIED
		.... .1..		PSOPFDAU	"B'00000100" PROCESSING A DAUGHTER CHAIN
		.... ..1.		PSOPFACT	"B'00000010" Accounting cell addr valid
		.... ...1		PSOPFUSR	"B'00000001" Userid included in PSODEST
13	(D)	BITSTRING	1	PSOPFLG2	Flag used by PSO processor
		1... ....		PSOP2E58	"B'10000000" ENF58 Select issued, ENF58 Deselect not yet issued
14	(E)	BITSTRING	18		Reserved for future use
32	(20)	SIGNED	4	PSORSVD2	RESERVED
36	(24)	BITSTRING	768	PSOPDDB	Work area for Pddb
806	(326)	BITSTRING	1	PSOUFLG	GROUP REQUEST OPTIONS BYTE
807	(327)	CHARACTER	1		RESERVED
808	(328)	BITSTRING	1	PSOFLG1	DATA SET SELECTION CONTROL FLAGS
809	(329)	BITSTRING	1	PSOFLG2	DATA SET DISPOSITION AND CTL FLAGS
810	(32A)	SIGNED	2	PSOCOPY	NUMBER OF DATA SET COPIES
812	(32C)	CHARACTER	8	PSOJOBN	JOB NAME
820	(334)	CHARACTER	8	PSOJOBI	HASP JOB ID (JOBNNNNN)
828	(33C)	CHARACTER	1	PSOCLS	NEW SYSOUT CLASS (GROUP REQ ONLY)
829	(33D)	SIGNED	2	PSOMLRL	Maximum Logical record length

## \$PSO Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

Prior to OY64290, userid was not allowed to be included in destination (SSSODEST) on a Process SYSOUT request. Therefore, flag SSSOUSER was assumed to indicate a TSO RECEIVE request by HASPPSO processing.

With OY64290, userid is now allowed:

- If SSSOUSER is ON and PSOPFUSR is OFF, the userid in PSOPGMN is for a TSO RECEIVE call.
- If both PSOPFUSR and SSSOUSER are ON, then the userid in PSOPGMN is the userid included in the input destination SSSODEST/PSODEST.

End of Comment

831	(33F)	BITSTRING	1	PSOFLGA	Flag byte
832	(340)	CHARACTER	8	PSODEST	REMOTE USER ID FOR SELECTION
840	(348)	CHARACTER	8	PSOPGMN	USER WRITER NAME
848	(350)	CHARACTER	8	PSORBA	RBA OF SYSOUT DATA SET
856	(358)	CHARACTER	44	PSODSN	SYSOUT DATA SET NAME
900	(384)	CHARACTER	4	PSOFORM	DATA SET FORM NAME (first 4 bytes if 8 byte form name)
904	(388)	CHARACTER	8	PSOCLAS	CLASS(ES) TO BE PROCESSED
912	(390)	ADDRESS	4	PSOWTRC	ADDR OF XWTR PARAMETER LIST
916	(394)	CHARACTER	8	PSODSID	DATA SET IDENT CHAR STRING

Comment

PROCESS SYSOUT EXTENSION (This section is present if flag SSSOPSEE is on in byte PSOFLG2)

End of Comment

916	(394)	X'39C'	0	PSOPSE	*** PROCESS SYSOUT EXTENSION
924	(39C)	BITSTRING	1	PSOFLG3	BDT CONTROL BYTE
925	(39D)	BITSTRING	1	PSOFLG4	USER JOB OPTION FLAG

Comment

The first release of support for security tokens provided the field SSSOJECT as the address of the security token area which was to be provided by the caller. There was no requirement that the caller provide the length or version that was expected to be returned. It was assumed that the caller would provide an area large enough for the version one form of the SAF token. This new support will allow the caller to specify the length and version of the SAF token. The token will be transformed from the current version and length to the version and length requested by the caller via the TOKENMAP service of the SAF interface.

In order to allow migration of process sysout users, a two stage 'commit' is provided. The PSO user can ask that the length and version in the area pointed to by SSSOJECT be used for TOKENMAP by setting SSSOTKNR.



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
<p>If the JES servicing the request has had the other half of this update, it will return the data in the requested format and set SSSOTKNG that says it did so.</p> <p>If the process sysout user does not ask for this service, (by not setting SSSOTKNR), the JES will copy the token to the address specified in SSSOJECT assuming that the length of the area is the same as the SAF version one token length</p>					
End of Comment					
926	(39E)	BITSTRING	1	PSOFLG5	Flags
927	(39F)	CHARACTER	1	PSORSV6	RESERVED
928	(3A0)	CHARACTER	8		JES3 use only
936	(3A8)	SIGNED	4	PSOLNCT	DATASET LINE COUNT
940	(3AC)	CHARACTER	8	PSOPRCD	DATASET PROC NAME
948	(3B4)	CHARACTER	8	PSOSTPD	DATASET STEP NAME
956	(3BC)	CHARACTER	8	PSODDND	DATASET DD NAME
964	(3C4)	ADDRESS	4	PSOJECT	Address of SAF token
968	(3C8)	CHARACTER	8	PSOFOR8	Form number
976	(3D0)	ADDRESS	4	PSOACCT	Address of acctng string

Comment

The following field has the 26 character JES2 JOE name (Job Output Element name). The string can be used as given in JES2 commands which require OUTGRP= specifications. Flag SSSOJNVA (in SSSOFLG5) is set if the field is valid.

The data set returned with a given output group name will not necessarily continue to have the given output group name if this request (or a later request) asks for held data sets (SSSOUFLG on) and data set characteristics are changed (via a non-zero SSSOUFLG).

End of Comment

980	(3D4)	CHARACTER	26	PSOOGNM	JES2 output group name
1006	(3EE)	CHARACTER	14		Reserved for future use
1020	(3FC)	ADDRESS	4	PSOTCB	TCB ADDRESS OF LAST PSO USER
1024	(400)	BITSTRING	4	PSORETN	SUB-SYSTEM RETURN CODE
1028	(404)	BITSTRING	6	PSOIOTTK	Track address (MQTR) of IOT
1034	(40A)	BITSTRING	2		Reserved
1036	(40C)	BITSTRING	6	PSOIOTCH	MQTR FOR IOT CHAIN
1042	(412)	BITSTRING	2		Reserved
1044	(414)	ADDRESS	4	PSOSJB	ADDRESS OF SJB
1048	(418)	ADDRESS	4	PSOECBP	Address of ECB for JOT WAIT
1052	(41C)	BITSTRING	4	PSORDRON	TIME ON INPUT PROCESSOR
1056	(420)	BITSTRING	4	PSORDTON	DATE ON INPUT PROCESSOR
1060	(424)	CHARACTER	8	PSOUSEID	JMR installation data field
1068	(42C)	ADDRESS	4	PSOWKOFF	OFFSET OF WORK JOE
1072	(430)	ADDRESS	4	PSOCHOFF	OFFSET OF CHAR JOE
1076	(434)	BITSTRING	1	PSOJOEFL	COPY OF JOEFLAG1
1077	(435)	BITSTRING	1	PSOFLGR	DATA SET SECURITY ATTR FLGS
		1... ..		PSORSEL	"B'10000000" SELECT REQUEST (NON-DISP)
1078	(436)	CHARACTER	1	PSOMCLAS	MESSAGE CLASS FOR JOB
1079	(437)	BITSTRING	1		RESERVED FOR FUTURE USE
1080	(438)	BITSTRING	4	PSOJBKEY	JOB IDENTIFIER KEY
1084	(43C)	ADDRESS	4	PSOMTRB	Address of MTRB
1088	(440)	ADDRESS	4	PSOPCE	Address of current or last PCE that processed
1092	(444)	ADDRESS	4	PSOJQEP	HASP JQE OFFSET
1096	(448)	SIGNED	4	PSOJBNUM	HASP job number (binary)
1100	(44C)	SIGNED	4	PSOROUTE	SELECTION ROUTE CODE (BINARY)
1104	(450)	SIGNED	2	PSOHDFFF	IOT offset of selected held PDDB and held support indicator

## \$PSO Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
1106	(452)	SIGNED	2	PSOXDOFF	IOT offset of selected non-held Pddb
1108	(454)	CHARACTER	80	PSOTOKEN	PSO SECURITY TOKEN FOR REQUESTOR
1188	(4A4)	CHARACTER	80	PSODSTOK	DATA SET TOKEN
1268	(4F4)	BITSTRING	148	PSOACCT2	Accounting string
1416	(588)	ADDRESS	4	PSOACTGC	Address of accounting cell
1420	(58C)	BITSTRING	2		Reserved for future use
1422	(58E)	BITSTRING	1	PSOFLG6	Flags (use CS to serialize)
		1... ....		PSO6FFGX	"B'10000000" XWTR wait for JOT post
		.1.. ....		PSO6FFGJ	"B'01000000" PSO wait for JOT post
1422	(58E)	X'CO'	0	PSO6FFGW	"PSO6FFGJ+PSO6FFGX" Waiting for JOT processing
		...1. ....		PSO6JWEL	"B'00100000" A JWEL created
		...1 ....		PSO6SJID	"B'00010000" PSO application is an STC
1423	(58F)	BITSTRING	1	PSOFLG7	Flags (use CS to serialize)
		...1 ....		PSO7ABRT	"B'00010000" PSO request aborted
		.... 1...		PSO7XPST	"B'00001000" Application has been POSTed
		.... .1..		PSO7PCE	"B'00000100" PCE is processing
1424	(590)	BITSTRING	8	PSOASCBT	Address space token
1432	(598)	SIGNED	2	PSOASID	Address space ID
1434	(59A)	CHARACTER	8	PSOJOBID	Job ID of PSO application
1442	(5A2)	CHARACTER	8	PSOJOBNM	Job name of PSO applicaiton
1450	(5AA)	CHARACTER	8	PSOCHKEY	CHKEY from CSCB
1464	(5B8)	DBL WORD	8	(0)	
1464	(5B8)	X'5B8'	0	PSOLNGTH	**-'PSO' LENGTH OF PSO DSECT

## \$PSO Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PSO	0		PSOJOBID	59A	
PSOACCT	3D0		PSOJOBNM	32C	
PSOACCT2	4F4		PSOJOBFL	434	
PSOACTGC	588		PSOJQEP	444	
PSOASCBT	590		PSOLNCT	3A8	
PSOASID	598		PSOLNGTH	5B8	5B8
PSOCHKEY	5AA		PSOMCLAS	436	
PSOCHOFF	430		PSOMLRL	33D	
PSOCLAS	388		PSOMTRB	43C	
PSOCLS	33C		PSONEXT	4	
PSOCOPY	32A		PSOOGNM	3D4	
PSOCRDT	8		PSOPCE	440	
PSODDND	3BC		PSOPDDB	24	
PSODEST	340		PSOPFACT	C	2
PSODSID	394	C3C3C3C3	PSOPFAU	C	4
PSODSN	358		PSOPFLD	C	8
PSODSTOK	4A4		PSOPFLAG	C	
PSOECBP	418		PSOPFLG2	D	
PSOFLGA	33F		PSOPFREQ	C	40
PSOFLGR	435		PSOPFSWT	C	80
PSOFLG1	328		PSOPFUSR	C	1
PSOFLG2	329		PSOPGMN	348	
PSOFLG3	39C		PSOPRCD	3AC	
PSOFLG4	39D		PSOPSE	394	39C
PSOFLG5	39E		PSOP2E58	D	80
PSOFLG6	58E		PSORBA	350	
PSOFLG7	58F		PSORDRON	41C	
PSOFORM	384		PSORDTON	420	
PSOFOR8	3C8		PSORETN	400	
PSOHDOFF	450		PSOROUTE	44C	
PSOID	0	5BD7E2D6	PSORSEL	435	80
PSOIOTCH	40C		PSORSVD2	20	
PSOIOTTK	404		PSORSV6	39F	
PSOJBKEY	438		PSOSECT	3C4	
PSOJBNUM	448		PSOSJB	414	
PSOJOBI	334				

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
PSOSTPD	3B4	
PSOTCB	3FC	
PSOTOKEN	454	
PSOUFLG	326	
PSOUSEID	424	
PSOWKOFF	42C	
PSOWTRC	390	
PSOXDOFF	452	
PSO6FFGJ	58E	40
PSO6FFGW	58E	C0
PSO6FFGX	58E	80
PSO6JWEL	58E	20
PSO6SJID	58E	10
PSO7ABRT	58F	10
PSO7PCE	58F	4
PSO7XPST	58F	8

## \$PSO Cross Reference

## \$PSOWORK Information

### \$PSOWORK Heading Information

**Common Name:** JES2 Process SYSOUT Work Area  
**Macro ID:** \$PSOWORK  
**DSECT Name:** PCE (\$PSOWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol PSPWKSIZ for the length of this work area. The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE  
**Pointed to by:** The \$PSOPCE field of the \$HCT data area points into the \$PCEORG/\$PCELAST chain to the first Process SYSOUT PCE. Since this chain contains all PCEs, use the PCEID field to determine when you have chained past the last PCE of this type. See \$PCE for other pointer fields that apply to all PCE types.  
 PSOPCE of the \$PSO data area

**Serialization:** Normal PCE dispatch serialization

**Function:** The fields in this work area are used by a JES2 Process SYSOUT Processor and by its support routines and exits. \$PSOWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$PSOWORK are actually part of the PCE DSECT, but only map PCEs with the value PCEPSOID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$PSOWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	ADDRESS	4	PSPCKPTB	OUTPUT CKPT I/O BUFFER ADDRESS
316	(13C)	ADDRESS	4	PSPCHARJ	Addr of current Char JOE (only valid if SSSOSPGM is on and SSSOSCLS off in PSOFLG1)
320	(140)	BITSTRING	1	PSPMCLAS	SYSOUT MESSAGE CLASS (FROM JCT)
321	(141)	BITSTRING	1		Reserved for future use
322	(142)	SIGNED	2	PSPWORKA	PSO PROCESSING WORK AREA
324	(144)	SIGNED	4	PSPRSVD2	RESERVED
328	(148)	BITSTRING	3	PSPHD SCT	HELD DATA SET COUNT
331	(14B)	BITSTRING	1		RESERVED
332	(14C)	CHARACTER	18	PSPDEST	PSO DESTINATION WORK AREA
350	(15E)	BITSTRING	12	PSPXWTR E	SAVE XWTR ECB
362	(16A)	BITSTRING	1	PSPFLAG1	FLAGS
		1... ....		PSP1RLSE	"B'10000000" BUFFER WAS PAGE RELEASED, HENCE NEEDS IOB BUILD

# \$PSOWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		PSP1RACR	"B'01000000" RACROUTE HAS BEEN DONE
		..1. ....		PSP1FSAM	"B'00100000" Pddb is repeated
		...1 ....		PSP1JRUL	"B'00010000" SET OFF USE JOE BUILD RULE 1 SET ON USE JOE BUILD RULE 2
363	(16B)	.... 1...	1	PSP1IOTR	"B'00001000" IOT is in memory (flag bit is only valid for xwtr)
		BITSTRING		PSPFLAG2	FLAGS FOR PSO QUEUE PROC
		1... ....		PSP2LQUE	"B'10000000" PROCESS LOCAL QUEUE
		.1.. ....		PSP2UQUE	"B'01000000" PROCESS USERID QUEUE
		..1. ....		PSP2RQUE	"B'00100000" PROCESS REMOTE QUEUE
		...1 ....		PSP2UQP	"B'00010000" USERID Q HAS BEEN PROCESSED
		.... 1...		PSP2RQP	"B'00001000" REMOTE Q HAS BEEN PROCESSED
		.... .1..		PSP2CQP	"B'00000100" CHAR Q IS BEING PROCESSED
364	(16C)	ADDRESS	4	PSPBUFAD	Buffer address \$EXCP
368	(170)	BITSTRING	192	PSPRGRPM	TREGROUP Parm list
560	(230)	ADDRESS	4	PSPCLASS	Current class list address
564	(234)	ADDRESS	4	PSPCALET	and its ALET
568	(238)	BITSTRING	4	PSPJBKEY	Job key

Comment

-----  
 PSPPSO is populated just prior to \$SEAS requests.  
 Do not use this copy of the PSO in other  
 circumstances.  
 -----

End of Comment

576	(240)	DBL WORD	8	(0)	Ensure double word aligned
576	(240)	BITSTRING	1464	PSPPSO	Copy of PSO (See above)
2040	(7F8)	SIGNED	4	PSPROUTE	Selection route code (bin)
2044	(7FC)	BITSTRING	1	PSPTOKEN	Data set token field

Comment

-----  
 The following fields are needed for HASP186 message.  
 -----

End of Comment

2124	(84C)	CHARACTER	8	PSPJOBID	Job id of PSO application
2132	(854)	CHARACTER	8	PSPJOBNM	Job name of PSO application
2140	(85C)	ADDRESS	4	PSPCHKEY	CHKEY from CSCB

Comment

End of HASP186 fields

End of Comment

2144	(860)	BITSTRING	8	PSPASCBT	Application ASCB token
2152	(868)	CHARACTER	18	PSPRTXT	Reason text area for ENF58

Comment

--BLDM \$BLDMSG MF=L List form of \$BLDMSG

End of Comment

2172	(87C)	SIGNED	4	PSPBLDM (0)	Control block ID
2176	(880)	BITSTRING	4		Console ID
2180	(884)	ADDRESS	4		Address of the CART
2184	(888)	ADDRESS	4		Pointer for JOBID
2188	(88C)	ADDRESS	4		Control block address
2192	(890)	ADDRESS	4		Display routine address
2196	(894)	ADDRESS	4	(6)	6 word work area
2220	(8AC)	ADDRESS	4		Caller's R11 value

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
2224	(8B0)	BITSTRING	2		ROUT code for Message
2226	(8B2)	BITSTRING	2		Not used
2228	(8B4)	CHARACTER	4		Message ID
2232	(8B8)	CHARACTER	1		Separator character
2233	(8B9)	ADDRESS	1		Flag byte 1
2234	(8BA)	ADDRESS	1		'DISPER'
2235	(8BB)	ADDRESS	1		Flag byte 2
2236	(8BC)	ADDRESS	1		Flag byte 3
2237	(8BD)	CHARACTER	8		Symbolic name of dest.
2245	(8C5)	BITSTRING	15		Not used
2260	(8D4)	ADDRESS	4	(0)	Ensure multiple of 4
2260	(8D4)	ADDRESS	2	(0)	
2264	(8D8)	DBL WORD	8	(0)	Ensure double word aligned
2264	(8D8)	X'7A0'	0	PSPWKSIZ	**-PCEWORK" LENGTH OF PSO PCE WORK AREA

**\$PSOWORK Cross Reference**

Name	Hex Offset	Hex Value
PCE	0	
PSPASCBT	860	
PSPBLDM	87C	C2D3C440
PSPBUFAD	16C	
PSPCALET	234	
PSPCHARJ	13C	
PSPCHKEY	85C	
PSPCKPTB	138	
PSPCLASS	230	
PSPDEST	14C	
PSPFLAG1	16A	
PSPFLAG2	16B	
PSPHDSCT	148	
PSPJBKEY	238	
PSPJOBID	84C	
PSPJOBNM	854	
PSPMCLAS	140	
PSPPSO	240	
PSPRGRPM	170	
PSPROUTE	7F8	
PSPRSVD2	144	
PSPRTXT	868	
PSPTOKEN	7FC	
PSPWKSIZ	8D8	7A0
PSPWORKA	142	
PSPXWTRE	15E	
PSP1FSAM	16A	20
PSP1IOTR	16A	8
PSP1JRUL	16A	10
PSP1RACR	16A	40
PSP1RLSE	16A	80
PSP2CQP	16B	4
PSP2LQUE	16B	80
PSP2RQP	16B	8
PSP2RQUE	16B	20
PSP2UQP	16B	10
PSP2UQUE	16B	40

## \$PSOWORK Cross Reference



---

## \$PSV Information

### \$PSV Programming Interface information

Programming Interface information

\$PSV

End of Programming Interface information

## Heading Information • \$PSV Map

### \$PSV Heading Information

**Common Name:** JES2 save area DSECT  
**Macro ID:** \$PSV  
**DSECT Name:** PSV and PSVAREGS  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SAVE, CSAV, and ARSA  
 Offset: PSVID-PSV and PSVARID-PSVAREGS  
 Length: L'PSVID and L'PSVARID  
**Storage Attributes:** Subpool: 0 or 230  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M), in the private storage of the JES2 or user address space.  
**Size:** See PSVLENG and PSVARLEN  
**Created by:** \$SAVE service  
**Pointed to by:** \$SAVAREA field of the \$HCT data area  
 \$SAVEARS field of the \$HCT data area  
 PCELPSV field of the \$PCE data area  
 PREPSVAD field of the \$PRE data area  
 PSVNEXT field of the \$PSV data area  
 PSVPREV field of the \$PSV data area  
 PSVARPTR field of the \$PSV data area  
 PSVARCHN field of the \$PSV data area  
 TRERSAVE field of the \$TRE data area  
**Serialization:** None required  
**Function:** This macro generates an MVS style save area DSECT mapping with JES2 extensions added on the end. The DSECT generated is dependent on the caller's environment.

### \$PSV Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PSV	
0	(0)	CHARACTER	4	PSVID	SAVE AREA IDENTIFIER
4	(4)	ADDRESS	4	PSVPREV	PREVIOUS SAVE AREA ADDRESS
8	(8)	ADDRESS	4	PSVNEXT	NEXT SAVE AREA ADDRESS
12	(C)	SIGNED	4	PSVR14	REGISTER 14 STORAGE
16	(10)	SIGNED	4	PSVR15	REGISTER 15 STORAGE
20	(14)	SIGNED	4	PSVR0	REGISTER 0 STORAGE
24	(18)	SIGNED	4	PSVR1	REGISTER 1 STORAGE
28	(1C)	SIGNED	4	PSVR2	REGISTER 2 STORAGE
32	(20)	SIGNED	4	PSVR3	REGISTER 3 STORAGE
36	(24)	SIGNED	4	PSVR4	REGISTER 4 STORAGE
40	(28)	SIGNED	4	PSVR5	REGISTER 5 STORAGE
44	(2C)	SIGNED	4	PSVR6	REGISTER 6 STORAGE
48	(30)	SIGNED	4	PSVR7	REGISTER 7 STORAGE
52	(34)	SIGNED	4	PSVR8	REGISTER 8 STORAGE
56	(38)	SIGNED	4	PSVR9	REGISTER 9 STORAGE
60	(3C)	SIGNED	4	PSVR10	REGISTER 10 STORAGE
64	(40)	SIGNED	4	PSVR11	REGISTER 11 STORAGE
68	(44)	SIGNED	4	PSVR12	REGISTER 12 STORAGE
72	(48)	ADDRESS	4	PSVADDR	ENVIRONMENT DEPENDENT ADDRS (CSAV area in USER env.)
72	(48)	X'48'	0	PSVPCE	"PSVADDR,4,C'A" PCE addr (main task only)
76	(4C)	ADDRESS	4	PSVLABAD	ADDRESS OF \$SAVE IDENTIFIER

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----					
The offset of the following fields are referenced in PLAS code. The offset is frozen					
-----					
End of Comment					
80	(50)	ADDRESS	2	(0)	
80	(50)	ADDRESS	2	(0)	
80	(50)	SIGNED	4	PSVHR13	High half register 13
84	(54)	SIGNED	4	PSVHR14	High half Register 14
88	(58)	SIGNED	4	PSVHR15	High half Register 15
92	(5C)	SIGNED	4	PSVHR0	High half Register 0
96	(60)	SIGNED	4	PSVHR1	High half Register 1
100	(64)	SIGNED	4	PSVHR2	High half Register 2
104	(68)	SIGNED	4	PSVHR3	High half Register 3
108	(6C)	SIGNED	4	PSVHR4	High half Register 4
112	(70)	SIGNED	4	PSVHR5	High half Register 5
116	(74)	SIGNED	4	PSVHR6	High half Register 6
120	(78)	SIGNED	4	PSVHR7	High half Register 7
124	(7C)	SIGNED	4	PSVHR8	High half Register 8
128	(80)	SIGNED	4	PSVHR9	High half Register 9
132	(84)	SIGNED	4	PSVHR10	High half Register 10
136	(88)	SIGNED	4	PSVHR11	High half Register 11
140	(8C)	SIGNED	4	PSVHR12	High half Register 12
144	(90)	ADDRESS	4	PSVLSPTR	LINKAGE STACK POINTER (AT TIME OF \$SAVE)
148	(94)	ADDRESS	4	PSVARPTR	Pointer to AR save area (or zero if none)
152	(98)	BITSTRING	1	PSVMODE	AR mode and key of caller Bits 0-3 - ASC mode 4-7 - PSW key
153	(99)	SIGNED	1	PSVEXID	Exit number when \$SAVE done
154	(9A)	BITSTRING	1	PSVAMODE	AMODE of caller
		1... ....		PSVAM31	"B'10000000" AMODE 31
		.... ...1		PSVAM64	"B'00000001" AMODE 64
155	(9B)	BITSTRING	1		Reserved
156	(9C)	ADDRESS	4	PSVADDR2	Environ dependent addr 2 (TRX in USER environment)
160	(A0)	DBL WORD	8	PSVSTCK	Time PSV was created
160	(A0)	X'A8'	0	PSVLENG	"*-PSV" LENGTH OF SAVE AREA

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PSVAREGS	, Start of access registers
0	(0)	CHARACTER	4	PSVARID	Eyecatcher
4	(4)	ADDRESS	4	PSVARCHN	Free chain pointer or ptr to real save area
8	(8)	SIGNED	4	PSVAR0	Access register 0 storage
12	(C)	SIGNED	4	PSVAR1	Access register 1 storage
16	(10)	SIGNED	4	PSVAR2	Access register 2 storage
20	(14)	SIGNED	4	PSVAR3	Access register 3 storage
24	(18)	SIGNED	4	PSVAR4	Access register 4 storage
28	(1C)	SIGNED	4	PSVAR5	Access register 5 storage
32	(20)	SIGNED	4	PSVAR6	Access register 6 storage
36	(24)	SIGNED	4	PSVAR7	Access register 7 storage
40	(28)	SIGNED	4	PSVAR8	Access register 8 storage
44	(2C)	SIGNED	4	PSVAR9	Access register 9 storage
48	(30)	SIGNED	4	PSVAR10	Access register 10 storage
52	(34)	SIGNED	4	PSVAR11	Access register 11 storage
56	(38)	SIGNED	4	PSVAR12	Access register 12 storage
60	(3C)	SIGNED	4	PSVAR13	Access register 13 storage
64	(40)	SIGNED	4	PSVAR14	Access register 14 storage
68	(44)	SIGNED	4	PSVAR15	Access register 15 storage
72	(48)	BITSTRING	1	PSVARFLG	Access register flags
		1... ....		PSVARF15	"B'10000000" Restore access register 15

## \$PSV Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		PSVARF0	"B'01000000" Restore access register 0
		..1. ....		PSVARF1	"B'00100000" Restore access register 1
73	(49)	BITSTRING	7		Reserved
73	(49)	X'50'	0	PSVARLEN	** -PSVAREGS" Length of access registers

## \$PSV Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PSV	0		PSVPREV	4	
PSVADDR	48		PSVR0	14	
PSVADDR2	9C		PSVR1	18	
PSVAMODE	9A		PSVR10	3C	
PSVAM31	9A	80	PSVR11	40	
PSVAM64	9A	1	PSVR12	44	
PSVARCHN	4		PSVR14	C	
PSVAREGS	0		PSVR15	10	
PSVARFLG	48		PSVR2	1C	
PSVARF0	48	40	PSVR3	20	
PSVARF1	48	20	PSVR4	24	
PSVARF15	48	80	PSVR5	28	
PSVARID	0		PSVR6	2C	
PSVARLEN	49	50	PSVR7	30	
PSVARPTR	94		PSVR8	34	
PSVAR0	8		PSVR9	38	
PSVAR1	C		PSVSTCK	A0	
PSVAR10	30				
PSVAR11	34				
PSVAR12	38				
PSVAR13	3C				
PSVAR14	40				
PSVAR15	44				
PSVAR2	10				
PSVAR3	14				
PSVAR4	18				
PSVAR5	1C				
PSVAR6	20				
PSVAR7	24				
PSVAR8	28				
PSVAR9	2C				
PSVEXID	99				
PSVHR0	5C				
PSVHR1	60				
PSVHR10	84				
PSVHR11	88				
PSVHR12	8C				
PSVHR13	50				
PSVHR14	54				
PSVHR15	58				
PSVHR2	64				
PSVHR3	68				
PSVHR4	6C				
PSVHR5	70				
PSVHR6	74				
PSVHR7	78				
PSVHR8	7C				
PSVHR9	80				
PSVID	0	C3E2C1E5			
PSVLABAD	4C				
PSVLENG	A0	A8			
PSVLSPTR	90				
PSVMODE	98				
PSVNEXT	8				
PSVPCE	48	48			

---

## \$QSE Information

### \$QSE Programming Interface information

Programming Interface information

\$QSE

End of Programming Interface information

## Heading Information • \$QSE Map

### \$QSE Heading Information

**Common Name:** Multi-access SPOOL shared communications queue control element  
**Macro ID:** \$QSE  
**DSECT Name:** QSE  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: 0 or 231  
 Key: 1  
 Residency: Virtual and Real storage are anywhere  
**Size:** See QSELEN  
**Created by:** HASPIRDA  
**Pointed to by:** \$QSE1 field of the \$HCT data area  
 \$AQSE field of the \$HCT data area  
**Serialization:** Fields are updated only when the JES2 checkpoint is owned by the member updating.  
**Function:** One QSE exists for each member of a multi-access spool. The QSE describes this potential member of the complex.  
 The QSEs are checkpointed control blocks. This means there are two or more copies of each QSE in storage at any one time. The actual and I/O copies are always there and reside in subpool 0. If the system is running with an application copy of the checkpoint a third copy will be in subpool 0 if the application copy is in private, and in subpool 231 if the application copy is in common. If the system is running with checkpoint versioning, then zero or more copies will be in a data space.  
 Note: the QSEs are contiguous in storage and must stay that way since the way to get to the QSE for a system is to use the system busy byte to index into the QSE table which begins at the address held in \$QSE1.

### \$QSE Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	QSE	
0	(0)	DBL WORD	8	QSESITIM	TOD of last CKPT access for this member
8	(8)	CHARACTER	4	QSESID	Member name-default SMF ID
12	(C)	BITSTRING	1	QSERSTID	Id of member doing \$ESYS
13	(D)	BITSTRING	1		Reserved for future IBM use
14	(E)	BITSTRING	1	QSESIBSY	Member id for busy indicators and equal to the member number
15	(F)	BITSTRING	3	QSEAFFIN	This members affinity token
18	(12)	BITSTRING	6		Reserved for future use
Comment					
Beginning of fields zeroed at all member warm start					
End of Comment					
24	(18)	BITSTRING	8	QSEPLXID	Sysplex id
32	(20)	BITSTRING	8	QSEMEMTK	XCF member token
40	(28)	DBL WORD	8	QSEECF (0)	CROSS-SYSTEM RESOURCE \$POST ECF
48	(30)	CHARACTER	8	QSEPLXNM	MVS sysplex name

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
End of fields zeroed at all member warm start					
End of Comment					
48	(30)	X'18'	0	QSEWARM1	"QSEPLXID,*-QSEPLXID" Zeroed at all-mbr wrmstart
56	(38)	CHARACTER	8	QSEMVSNM	MVS system name
Comment					
Beginning of fields zeroed at all member warm start					
End of Comment					
64	(40)	CHARACTER	8	QSEJ2VRN	The JES2 version that last warmstarted this member
72	(48)	BITSTRING	4	QSESYTOK	System token of the MVS system
76	(4C)	BITSTRING	1	QSEPOSTS	CROSS-SYSTEM \$POST FLAG BYTES
		1... ....		QSEPXEQ	"B'10000000" Cross-system \$POST execution
		.1... ....		QSEPJOT	"B'01000000" Cross-member \$#POST for JOE/JQE
Comment					
EQU B'00100000' Reserved (was QSEPSOST)					
End of Comment					
77	(4D)	BITSTRING	1	QSEJCMD	JCMD processor post
		1... ....		QSEJWRK	"B'10000000" Batch work for JCMD
Comment					
End of fields zeroed at all member warm start					
End of Comment					
77	(4D)	X'40'	0	QSEWARM2	"QSEJ2VRN,*-QSEJ2VRN" Zeroed at all-mbr wrmstart
78	(4E)	BITSTRING	1	QSESTAT	SYSTEM STATUS BYTE
Comment					

Valid states for QSEQUICK and QSE\$EMEM:  
 QSEQUICK QSE\$EMEM Explanation  
 off off no \$EMEMBER nor AMWS ever done  
     This is normal state for a  
     running member or a crashed  
     member for which no warmstart  
     has been attempted.  
 off on \$EMEMBER or AMWS has completed,  
     but at least one JQE was not  
     processed because of inability  
     to get BERT lock. The ALICE  
     PCE will set QSEQUICK on when  
     all JQEs have been processed.  
     This setting can also occur if  
     SMWS is done after an AMWS by  
     a member with OW35410 installed.  
 on off An AMWS or \$EMEMBER has been  
     done before OW35410 was

# \$QSE Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
					installed. This will disappear after the first ALICE processor successfully scans the jobqueue and should never appear in the MAS again unless an AMWS or \$EMEMBER is performed by a member without OW35410 installed.
					on on Member has been successfully warmstarted by an AMWS or via \$EMEMBER with no BERT problems.
					Note: AMWS = All Member Warm Start SMWS = Single Member Warm Start
End of Comment					
		1... ....		QSEACTIV	"B'10000000" Member is active
		.1. ....		QSEQUICK	"B'01000000" Member is warm started
		.1. ....		QSELAST	"B'00100000" Last QSE
		...1 ....		QSE\$PCMD	"B'00010000" \$P command in effect
		.... 1..		QSE\$PXEQ	"B'00001000" \$P XEQ command in effect
		.... .1.		QSEBOSS	"B'00000100" This member is boss
		.... ..1		QSE\$MEM	"B'00000010" \$E MEMBER finished
		.... ...1		QSECKPT2	"B'00000001" QSECPKLV is for CKPT2
79	(4F)	BITSTRING	1	QSESTAT2	ADDITIONAL SYSTEM STATUS BYTE
		1... ....		QSE2EDEL	"B'10000000" Member deleted, ESYS'D required
		.1. ....		QSE2\$IND	"B'01000000" Member in ind mode
		.1. ....		QSE2SIOT	"B'00100000" SPIN IOT being purged
		...1 ....		QSE2NMAL	"B'00010000" This member has two ckpt datasets allocated
		.... 1..		QSE2EGON	"B'00001000" XCF system gone, ESYS,SID required
Comment					
EQU B'00000100' Reserved (was QSE2NPST)					
End of Comment					
		.... ..1.		QSE2PRIM	"B'00000010" Member is a primary subsystem
		.... ...1		QSE2SPLX	"B'00000001" Command Prefix has SYSplex scope.
Comment					
Beginning of fields zeroed at all member warm start					
End of Comment					
80	(50)	BITSTRING	4	QSEMAXMS	Members that ceased sending msgs because of \$MAXMSGQ
84	(54)	BITSTRING	1	QSESCMSK	SHRD COMM QUE SPLS USED MSK
Comment					
The following fields are used by SDSF on their MEMBER display. QSEHOLD, QSEMIND, and QSEMAXD are in hundredths of a second. QSESYNC is in seconds. QSEAHOLD and QSEADORM are bits 16-47 of a STCK value. Bit 47 of a STCK is incremented every 0.000016 seconds.					
End of Comment					
116	(74)	SIGNED	4	QSEHOLD	MASDEF HOLD=
120	(78)	SIGNED	4	QSEMIND	MASDEF DORMANCY=(xxxx)
124	(7C)	SIGNED	4	QSEMAXD	MASDEF DORMANCY=(,xxxx)
128	(80)	SIGNED	4	QSESYNC	MASDEF SYNCTOL=
132	(84)	SIGNED	4	QSEAHOLD	Actual HOLD value
136	(88)	SIGNED	4	QSEADORM	Actual dormancy value



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
End of section for SDSF MEMBER display					
End of Comment					
140	(8C)	CHARACTER	1	QSECCHAR	CONDEF CONCHAR=
141	(8D)	BITSTRING	1	QSESTYPE	Last start type - see flag definitions in \$WARMTYP
142	(8E)	SIGNED	2		Possible number of lost TGs (not used in SP 5.2)
144	(90)	SIGNED	4	QSESTIME	STCK time of last start
148	(94)	BITSTRING	3	QSESNIFF	MTT being sniffed this mbr (if \$SPLLGD is off)
151	(97)	SIGNED	1	QSEJ2VR2	JES2 version last active on this member
152	(98)	SIGNED	1	QSEHIVER	Highest compatible JES2 version
153	(99)	BITSTRING	3	QSEFEAT	Reserved for future use as feature flags
Comment					
EQU B'10000000' Reserved (was QSEFXMT)					
End of Comment					
	.1..	....		QSEFLVLS	"B'01000000" Service level fields set (QSEJ2PLV/QSEJ2SLV)
	..1.	....		QSEFXNM	"B'00100000" QSEXLNM field set
	...1	....		QSEFDASA	"B'00010000" OA18783 applied
	....	1...		QSEFXGM	"B'00001000" Use XCFGRPNM for SYSJ2\$XD
Comment					
End of fields zeroed at all member warm start					
End of Comment					
153	(99)	X'50'	0	QSEWARM3	"QSEMAXMS,*-QSEMAXMS" Zeroed at all-mbr warmstart
156	(9C)	SIGNED	4	QSESYSLG	Current SYSLOG job index
160	(A0)	SIGNED	1	QSEJ2PLV	Product level of JES2 last active on this member (&J2PLVL)
161	(A1)	SIGNED	1	QSEJ2SLV	Service level of JES2 last active on this member (&J2SLVL)
162	(A2)	SIGNED	2		Reserved for future use
Comment					
Beginning of fields zeroed at all member warm start					
End of Comment					
164	(A4)	CHARACTER	4	QSESSNAM	Subsys name (JES2/JESA...)
168	(A8)	SIGNED	4	QSECKPLV	Last CKPT level number
Comment					
COMPATIBILITY					
<p>QSEGMTOF is used to determine whether the time stamps in CKBWRTIM (checkpoint write time) can be compared to each other. In Z2 mode, CKBWRTIM is stored in GMT rather than local time. In that case, time stamps may be compared regardless of GMT offset because GMT must be consistent on all systems in the sysplex, so QSEGMTOF is not used. When all members of the MAS must be in Z2 mode, (and therefore store CKBWRTIM in GMT), QSEGMTOF can be deleted.</p>					
End of Comment					
172	(AC)	SIGNED	4	QSEGMTOF	GMT offset of this member
176	(B0)	CHARACTER	4	QSEMVSID	SMF id of MVS system
180	(B4)	CHARACTER	8	QSEXLNM	Local node name at time of XCF join
188	(BC)	BITSTRING	5	QSESNFMT	MQT this memb is sniffing - 1 byte M and 4 byte TTTT (if \$SPLLGD is on)

## \$QSE Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
End of fields zeroed at all member warm start					
End of Comment					
188	(BC)	X'A4'	0	QSEWARM4	"QSESSNAM,*-QSESSNAM" Zeroed at all-mbr warmstart
193	(C1)	BITSTRING	3		Reserved
196	(C4)	SIGNED	4		Reserved (as of HJE7705)
196	(C4)	X'C8'	0	QSELEN	"*-QSE" LENGTH OF QSE
200	(C8)	ADDRESS	4	QSEEND (0)	End of QSE

## \$QSE Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
QSE	0		QSESNIFF	94	
QSE\$EMEM	4E	2	QSESSNAM	A4	
QSE\$PCMD	4E	10	QSESTAT	4E	
QSE\$PXEQ	4E	8	QSESTAT2	4F	
QSEACTIV	4E	80	QSESTIME	90	
QSEADORM	88		QSESTYPE	8D	
QSEAFFIN	F		QSESYNC	80	
QSEAHOLD	84		QSESYSLG	9C	
QSEBOSS	4E	4	QSESYTOK	48	
QSECCHAR	8C		QSEWARM1	30	18
QSECKPLV	A8		QSEWARM2	4D	40
QSECKPT2	4E	1	QSEWARM3	99	50
QSEECF	28		QSEWARM4	BC	A4
QSEEND	C8		QSEXLNNM	B4	
QSEFDASA	99	10	QSE2\$IND	4F	40
QSEFEAT	99		QSE2EDEL	4F	80
QSEFLVLS	99	40	QSE2EGON	4F	8
QSEFXGNM	99	8	QSE2NMAL	4F	10
QSEFXNNM	99	20	QSE2PRIM	4F	2
QSEGMTOF	AC		QSE2SIOT	4F	20
QSEHIVER	98		QSE2SPLX	4F	1
QSEHOLD	74				
QSEJCMD	4D				
QSEJWRK	4D	80			
QSEJ2PLV	A0				
QSEJ2SLV	A1				
QSEJ2VRN	40				
QSEJ2VR2	97				
QSELAST	4E	20			
QSELEN	C4	C8			
QSEMAXD	7C				
QSEMAXMS	50				
QSEMEMTK	20				
QSEMIND	78				
QSEMVSID	B0				
QSEMVSNM	38				
QSEPJOT	4C	40			
QSEPLXID	18				
QSEPLXNM	30				
QSEPOSTS	4C				
QSEPXEQ	4C	80			
QSEQUICK	4E	40			
QSERSTID	C				
QSESCMSK	54				
QSESIBSY	E				
QSESID	8				
QSESITIM	0				
QSESNFMT	BC				

---

## **\$RAT Information**

### **\$RAT Programming Interface information**

Programming Interface information

**\$RAT**

End of Programming Interface information

## Heading Information • \$RAT Map

### \$RAT Heading Information

**Common Name:** Remote Attribute Table  
**Macro ID:** \$RAT  
**DSECT Name:** RAT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** The permanent RATs are in contiguous storage with an eyecatcher 'RAT ' and storage length before the RAT table.  
Offset: -8 from value of \$RATABLE  
Length: 4

**Storage Attributes:** Subpool: 23 (temporary RAT during initialization), 0 (permanent RAT after initialization)  
Key: 1  
Residency: Virtual and real storage are anywhere in the private storage of the JES2 address space.

**Size:** RATTLE \* \$MAXRJE (temporary RAT during initialization)  
RATTLE \* \$RMTNUM (permanent RAT after initialization)

**Created by:** HASPIRMA (temporary RAT during initialization)  
HASPIRRE (permanent RAT after initialization)

**Pointed to by:** \$RATABLE field of the \$HCT data area  
MDCTRAT field of the \$DCT data area

**Serialization:** Logon of a remote is not permitted on an MAS member if the remote is logged on another member (which is indicated in the \$RMTSON vector).

**Function:** The RAT describes the attributes of a JES2 remote. Remotes are devices (remote workstations) or pseudo devices (e.g. programs emulating RJE protocols connected to JES2 via an SNA line). The protocol defines a single transmission mechanism (the line), with single transmitter/receivers at either end (e.g. the JES2 Line Manager PCE, and an RJE workstation CPU or RJE emulation program). Multiple streams of data records can be interleaved in the traffic to/from the MLLM and the workstation, which are broken out at each end as data to/for workstation peripherals (PRTs, RDRs, PUNs, CON) and the matching JES2 logical processors.

The RAT is a set of contiguous entries, one for each remote (the first is for remote 1, not 0). The size of each entry is RATTLE, and you can index into the RAT to find the desired entry using a remote number. The number of entries is \$RMTNUM, which is the largest allowed remote number (may be different on each MAS member). This is the RAT definition after initialization - during parmlib processing there is a temporary RAT with \$MAXRJE entries.

### \$RAT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RAT	REMOTE ATTRIBUTE TABLE DSECT
0	(0)	CHARACTER	8	RATNAME	REMOTE NAME

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
8	(8)	ADDRESS	4	RATRDCT	ADDRESS OF FIRST REMOTE DCT
12	(C)	ADDRESS	4	RATLDCT	ADDRESS OF LINE DCT
16	(10)	ADDRESS	4	RATTOKA	Address of SAF token
20	(14)	SIGNED	4	RATROUTE (0)	REMOTES ROUTE CODE
20	(14)	SIGNED	2	RATNODE	NODE NUMBER
22	(16)	SIGNED	2	RATRTE	REMOTE ROUTE
24	(18)	BITSTRING	1	RATTYPE	TERMINAL TYPE (SEE MDCTTYPE)
25	(19)	BITSTRING	1	RATFMT	TERMINAL DATA FORMAT
26	(1A)	BITSTRING	1	RATFEAT	TERMINAL FEATURES
27	(1B)	ADDRESS	1	RATNUMRD	NUMBER OF READERS
28	(1C)	ADDRESS	1	RATNUMPR	NUMBER OF PRINTERS
29	(1D)	ADDRESS	1	RATNUMPU	NUMBER OF PUNCHES
30	(1E)	BITSTRING	1	RATCONF	REMOTE CONSOLE FLAGS
31	(1F)	BITSTRING	1		RESERVED FOR FUTURE USE
32	(20)	SIGNED	4	RATCONRT	REMOTE CONSOLE ROUTE CODE
36	(24)	SIGNED	2	RATBUFSZ	TERMINAL BUFFER SIZE
38	(26)	SIGNED	2	RATWTIME	TERMINAL WAIT TIME
40	(28)	BITSTRING	1	RATFLAGS	REMOTE FLAGS
41	(29)	BITSTRING	1	RATFLAG2	REMOTE FLAGS
42	(2A)	SIGNED	2	RATDINTV	REMOTE DISCONNECT INTERVAL
44	(2C)	SIGNED	4	RATIMER	REMOTE CLOCK VALUE
48	(30)	CHARACTER	8	RATSYMB	REMOTES PRIMARY LUNAME
56	(38)	CHARACTER	8	RATPSWD	REMOTE PASSWORD
64	(40)	ADDRESS	4	RATRMJQE	RMT MSG DS JQE OFFSET - IF NON- ZERO, MSGS EXIST FOR PRT
68	(44)	SIGNED	2	RATLOGN	LOGON DCT NUMBER TO USE
70	(46)	SIGNED	2	RATRMTNO	Remote number
72	(48)	ADDRESS	8	RATCDCT	CDCT address
80	(50)	SIGNED	4	RATEND (0)	END OF RAT DSECT
80	(50)	X'50'	0	RATTLE	** -RAT" LENGTH OF RAT

Comment

RATCONF

End of Comment

1... ..	RATCONF	"B'10000000" DISPLAY TIME STAMP, JOB ID, TEXT
.1. ....	RATCONFJ	"B'01000000" DISPLAY JOB ID, TEXT
..1. ....	RATCONF	"B'00100000" REMOTE HAS A CONSOLE

Comment

ATCONFO EQU B'00010000' REMOTE CONSOLE OPERATIONAL

End of Comment

.... 1...	RATCONF	"B'00001000" ISSUE SETUP MSGS AS 'INFO'
.... .1..	RATCONF	"B'00000100" ISSUE HASP150 ON-DEVICE MSG TO RMT CONSOLE AS WELL AS OPERATOR, INEFFECTIVE IF CONFC IS OFF
.... ..1.	RATCONF	"B'00000010" SUPPRESS RMT MSG PRINTING (MSGPRT=NO ON RMTNN STMT)

Comment

RATFLAGS

End of Comment

1... ..	RATALM	"B'10000000" REMOTE IS IN AUTOLOGON MODE
.1. ....	RATSRMT	"B'01000000" REMOTE SHOULD BE STARTED
..1. ....	RATTINT	"B'00100000" TEMP DISC INTERVAL IN EFFECT
...1 ....	RATPILUN	"B'00010000" RATSYMB PERMANENTLY INITIALIZED

## \$RAT Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
EQU B'00001000' Reserved for future use					
EQU B'00000100' Reserved for future use					
End of Comment					
	....	..1.		RATVALID	"B'00000010" RAT ENTRY IS VALID, IF FLAG IS OFF NO RDR/PRT/PUN DCTS ARE ALLOCATED, SIGNON IS NOT BE PERMITTED
	....	...1		RATOUTPT	"B'00000001" OUTPUT EXISTS FOR THIS RMT
Comment					
RATFLAG2					
-----					
RAT2NSHR and RAT2SHRE are mutually exclusive flags, and are meaningless if RATLDCT contains zero or if remote is signed on to an unleased line.					
-----					
End of Comment					
	1... ..	....		RAT2QSCN	"B'10000000" AUTOLG FULL Q SCAN REQUIRED
	..1. ....	....		RAT2NSHR	"B'01000000" RATLDCT holds nonshared line
	..1. ....	....		RAT2SHRE	"B'00100000" RATLDCT holds shared line

## \$RAT Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
RAT	0		RATSRMT	50	40
RATALM	50	80	RATSYMB	30	
RATBUFSZ	24		RATTINT	50	20
RATCDCT	48		RATTLE	50	50
RATCONF	1E		RATTOKA	10	
RATCONFC	50	20	RATTYPE	18	
RATCONFD	50	4	RATVALID	50	2
RATCONFI	50	8	RATWTIME	26	
RATCONFJ	50	40	RAT2NSHR	50	40
RATCONFS	50	2	RAT2QSCN	50	80
RATCONFT	50	80	RAT2SHRE	50	20
RATCONRT	20				
RATDINTV	2A				
RATEND	50				
RATFEAT	1A				
RATFLAGS	28				
RATFLAG2	29				
RATFMT	19				
RATIMER	2C				
RATLDCT	C				
RATLOGN	44				
RATNAME	0				
RATNODE	14				
RATNUMPR	1C				
RATNUMPU	1D				
RATNUMRD	1B				
RATOUTPT	50	1			
RATPILUN	50	10			
RATPSWD	38				
RATRDCT	8				
RATRMJQE	40				
RATRMTNO	46				
RATROUTE	14				
RATRTE	16				

---

## \$RCPWORK Information

### \$RCPWORK Programming Interface information

Programming Interface information

\$RCPWORK

End of Programming Interface information

## Heading Information • \$RCPWORK Map

### \$RCPWORK Heading Information

**Common Name:** JES2 Remote Console Processor  
**Macro ID:** \$RCPWORK  
**DSECT Name:** PCE (\$RCPWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4  
**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE  
**Size:** See symbol RCPPCEWS for the length of this work area. The overall length of the PCE is stored in field PCELENG.  
**Created by:** See \$PCE  
**Pointed to by:** The \$MCONPCE field of the \$HCT data area points to the remote console PCE. See \$PCE for other pointer fields that apply to all PCE types.  
**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this work area are used by the JES2 remote console processor. \$RCPWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$RCPWORK are actually part of the PCE DSECT, but only map the PCE with the value PCERCPID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$RCPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	HASP REMOTE CONSOLE PROCESSOR
Comment					
REMOTE CONSOLE DEVICE CONTROL TABLE (WITH COMBINED BSC AND SNA FOUNDATION EXTENSIONS)					
End of Comment					
312	(138)	DBL WORD	8	(0)	BEGINNING OF DCT
312	(138)	BITSTRING	1	RCPDCT	BSC RMT DCT & EXTNSN
Comment					
END OF REMOTE CONSOLE DEVICE CONTROL TABLE					
End of Comment					
448	(1C0)	ADDRESS	1	RCPWF	WORK/WAIT FLAGS
449	(1C1)	BITSTRING	1	RCPWF2	MORE WORK/WAIT FLAGS
450	(1C2)	ADDRESS	1	RCPMF	MESSAGE PENDING FLAGS
452	(1C4)	ADDRESS	4	RCPJQTMR	WAITING FOR JOB QUEUE TIMER
456	(1C8)	DBL WORD	8	RCPEXTPL	RCP EXTP PARAMETER LIST AREA
464	(1D0)	DBL WORD	8	RCPBASET	TIME OF SCAN



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

CURRENT CONCURRENT FUNCTION EXIT ADDRESSES

End of Comment

472	(1D8)	SIGNED	4	RCPXIT (0)	
472	(1D8)	ADDRESS	4	RCPMSXIT	EXIT TO MESSAGE SPOOLING CHECK RTN
476	(1DC)	ADDRESS	4	RCPSOXIT	EXIT TO SPOOLING OUT FUNCTION
480	(1E0)	ADDRESS	4	RCPSIXIT	EXIT TO SPOOLING IN FUNCTION
484	(1E4)	ADDRESS	4	RCPIOXIT	EXIT TO INPUT FUNCTION
484	(1E4)	X'1D8'	0	RCPDMXCN	"RCPXIT,*-RCPXIT"
488	(1E8)	SIGNED	4	RCPREGSV (16)	INTERNAL RCP SAVE/WORK AREA
552	(228)	ADDRESS	4	RCPIOT	IOT BUFFER
556	(22C)	ADDRESS	4	RCPSWELQ	Pending SAF requests

Comment

MESSAGE SPOOLING CONTROLS

End of Comment

560	(230)	ADDRESS	4	RCPMSB1	FIRST MESSAGE BUFFER ADDRESS
564	(234)	ADDRESS	4	RCPMSBL	LAST MESSAGE BUFFER ADDRESS
568	(238)	ADDRESS	4	RCPMSLBW	LAST BUFFER WRITTEN TO MSG DS + 1
572	(23C)	ADDRESS	4	RCPMSCB	NEXT MESSAGE SPOOL BUFFER TO CHECK
576	(240)	ADDRESS	4	RCPMSBSV	NEXT MSG BUFFER TO PROCESS
580	(244)	ADDRESS	4	RCPMSRRD	BFR CURRENTLY BEING REREAD
584	(248)	ADDRESS	4	RCPMSIOT	ADDRESS OF CURRENT RMT MSG IOT
588	(24C)	ADDRESS	4	RCPMSBAT	BAT for IOT buffer
592	(250)	ADDRESS	4	RCPMSRAT	ADDRESS OF CURRENT RAT ELEMENT
596	(254)	ADDRESS	4	RCPMSNTK	MTRR OF NEXT MSG REC TO WRITE
600	(258)	ADDRESS	4	RCPMSCMB	CMB address
604	(25C)	ADDRESS	4	RCPMSHDR	SAVE AREA FOR CMB HEADER
608	(260)	ADDRESS	4	RCPMSRTE	RMT NO. OF CMB BEING SPOOLED
612	(264)	ADDRESS	4	RCPROUT	SCANNED OUTPUT REMOTE NUMBER
616	(268)	CHARACTER	8	RCPMSKEY (0)	REMOTE MESSAGE DS KEY
616	(268)	SIGNED	4	RCPMSKJK	JOB IDENTIFIER KEY
620	(26C)	SIGNED	4	RCPMSKDK	DATA SET KEY
624	(270)	ADDRESS	2	RCPBFSZ	CONSOLE BUFFER SIZE
626	(272)	BITSTRING	1	RCPCFLG	CMB processing flags
		1... ....		RCPCFQCH	"B'10000000" CMB has been dechained from \$BUSYRQ, OK to run entire chain
		.1.. ....		RCPCFVTS	"B'01000000" SNA buffer shortage HASP248 has been issued
		..1. ....		RCPCFBSS	"B'00100000" BSC buffer shortage HASP248 has been issued
		...1 ....		RCPCFMLM	"B'00010000" MLLM suspended HASP248 has been issued
		.... 1...		RCPCFPLF	"B'00001000" All CMBs dequeued from CCTNOUSQ
627	(273)	ADDRESS	1		RESERVED FOR FUTURE USE

Comment

SPOOL OUT CONTROLS

End of Comment

628	(274)	ADDRESS	4	RCPSOBAK	BACK UP ADDRESS TO SPOOL OUT BUFFER
632	(278)	ADDRESS	4	RCPSOOUT	SHARED QUEUE OUTPUT BUFFER ADDRESS
636	(27C)	ADDRESS	4	RCPSOLR	LOCATION OF LOGICAL RECORD
640	(280)	ADDRESS	2	RCPSOFRE	FREE DATA COUNT
642	(282)	ADDRESS	2	RCPSOCTR	COUNTER
644	(284)	ADDRESS	4	RCPSOQSE	CURRENT OUTPUT QSE ADDRESS
644	(284)	X'C'	0	RCPTABBL	"L'IOTRCPBA+L'TABRCPBA" BACK-UP AREA FOR 1 SYSTEM
648	(288)	BITSTRING	1	RCPTABBA (0)	Work data for \$TRACK
1032	(408)	ADDRESS	3	RCPSORT	CURRENT NODE-QUALIFIER TO SPOOL

## \$RCPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
NODAL OUTPUT CONTROL					
End of Comment					
1035	(40B)	ADDRESS	3	RCPNORT	CURRENT NODE TO NODE OUT
1038	(40E)	BITSTRING	3	RCPNMRP	3-BYTE PREFACE TO NODAL MSG REC
Comment					
NOTE: MUST ALWAYS PRECEDE NMR ASSEMBLY AREA --- USED BY \$EXTP PUT TO BUILD RID					
End of Comment					
1041	(411)	BITSTRING	178	RCPNMR	ASSEMBLY AREA FOR NODAL MESSAGE RECS
1219	(4C3)	ADDRESS	1	RCPAUTH	NODAL COMMAND AUTHORITY RESTRICTIONS
1220	(4C4)	ADDRESS	4	RCPNODCT	CURRENT NODAL OUTPUT DCT ADDR
1224	(4C8)	ADDRESS	4	RCPNPMB	DELAY TIMER FOR BUSY PATH MANAGER
Comment					
SHARED QUEUE INPUT CONTROL					
End of Comment					
1228	(4CC)	ADDRESS	4	RCPSIIN	SHARED QUEUE INPUT BUFFER
1232	(4D0)	ADDRESS	4	RCPSILR	LOCATION OF LOGICAL RECORD
1236	(4D4)	ADDRESS	2	RCPSICTR	COUNTER
1238	(4D6)	ADDRESS	2	RCPSICTA	COUNTER
1240	(4D8)	ADDRESS	4	RCPSIQSQ	CURRENT INPUT QSE CONTROL ELEMENT
1244	(4DC)	ADDRESS	4	RCPMTTR	SAVE AREA - CURRENT INPUT MTTR
1248	(4E0)	ADDRESS	2	RCPSID	System id number
1250	(4E2)	BITSTRING	48	RCPSIRT	CMB SPOOL HEADER IN MLWTO
1298	(512)	ADDRESS	2	RCPSIQCT	QSE SEARCH COUNTER SAVE AREA
Comment					
INPUT FUNCTION SPOOL OUT CONTROL					
End of Comment					
1300	(514)	ADDRESS	4	RCPIOOUT	INPUT OVERFLOW OUTPUT BUFFER ADDRESS
1304	(518)	ADDRESS	4	RCPIOLR	LOCATION OF LOGICAL RECORD
1308	(51C)	ADDRESS	2	RCPIOFRE	FREE DATA COUNT
1310	(51E)	ADDRESS	2	RCPIOCTR	COUNTER
1312	(520)	ADDRESS	4	RCPIOTTR	ACTIVE INPUT SPOOL OUTPUT RECRD
Comment					
NODAL INPUT CONTROL					
End of Comment					
1316	(524)	ADDRESS	4	RCPININ	INPUT OVERFLOW INPUT BUFFER
1320	(528)	ADDRESS	4	RCPINLNE	CURRENT INPUT SOURCE DCT ADDR
1324	(52C)	ADDRESS	4	RCPINTME	LINE INPUT DELAY TIMER

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
-----					
Area to receive commands from RJE's and NJE nodes					
Note that RCPTOKN is not referred to by name; the					
token is moved to the next available byte after the					
command.					
-----					
End of Comment					
1328	(530)	BITSTRING	216	RCPIN	Cmd/msg input work area
1544	(608)	BITSTRING	256	RCPINA	AREA TO RECEIVE DATA
1800	(708)	BITSTRING	1	RCPTOKN	Area for token

Comment					
MISCELLANEOUS					
-----					
End of Comment					
1880	(758)	SIGNED	4	(0)	Ensure alignment
1880	(758)	CHARACTER	64	RCPSAFPM	Parameters for MSAFCHK
1944	(798)	ADDRESS	1	RCPFL	FLAGS
1945	(799)	ADDRESS	1	RCPFI	FLAGS FOR INPUT FUNCTIONS
1946	(79A)	ADDRESS	2	RCPSCQOF	Offset of this member's SCQ
1948	(79C)	ADDRESS	4	RCPSCQAD	Addr of this member's SCQS
1952	(7A0)	ADDRESS	4	RCPRESV	RESERVED BUFFER
1956	(7A4)	BITSTRING	48	RCPLSAV	MLWTO CNTRL FOR INPUT LINES
2004	(7D4)	CHARACTER	18	RCPDSTWK	DEST WORK AREA
2022	(7E6)	CHARACTER	1	RCPRSV1	RESERVED FOR FUTURE USE

Comment					
LIST FORM OF GETMAIN					
-----					
End of Comment					
2024	(7E8)	ADDRESS	4	RCPGM	LENGTH
2028	(7EC)	ADDRESS	4		ADDR. OF ADDR. LIST
2032	(7F0)	BITSTRING	1		MODE AND OPTION FLAGS
2033	(7F1)	ADDRESS	1		SUBPOOL VALUE
2024	(7E8)	ADDRESS	4	RCPXSAV (3)	EXIT ROUTINE ACTIVATOR SAVE AREA
2036	(7F4)	ADDRESS	2	RCPSIDEL (0)	Offsets for interrupted READs
2100	(834)	ADDRESS	4	RCPSIDL	CURRENT INTERRUPTED READ OFFSET
2104	(838)	SIGNED	4	RCPREGS (16)	Save area for \$SETAFF macro

Comment					
Parameter list and other work areas for					
MVS Cloning translation service routine.					
-----					
End of Comment					
2168	(878)	SIGNED	4	RCPSYMBP (0)	Parameter List
2196	(894)	SIGNED	4	RCPSYMLN	Length of translated cmd
2200	(898)	SIGNED	4	RCPSYMRC	RC from translation service
2204	(89C)	ADDRESS	4	RCPSYMBF	Addr of trans output bfr
2204	(89C)	X'768'	0	RCPPCEWS	**'-PCEWORK'

Comment					
RCPWF DEFINITIONS					
-----					
End of Comment					
		1... ....		RCPWFCMB	"B'10000000" WAIT ON CMB
		.1.. ....		RCPWFQX	"B'01000000" WAIT ON JOB QUEUE IN EXIT

# \$RCPWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		..1. ....		RCPWFQO	"B'00100000" WAIT ON JOB QUEUE FOR OUTPUT
		...1 ....		RCPWFTRK	"B'00010000" WAIT ON \$TRACK
		.... 1...		RCPWFPCB	"B'00001000" PURGE CMB (CMB SHORTAGE)
		.... .1..		RCPWFBUF	"B'00000100" WAIT ON TP BUFFER
		.... .1.		RCPWFIBF	"B'00000010" WAIT ON INPUT TP BUFFER
		.... ...1		RCPWFPSO	"B'00000001" PURGE SPOOL OUTPUT BUFFER
Comment					
RCPWF2 DEFINITIONS					
End of Comment					
		.1.. ....		RCPWF2ML	"B'01000000" MLWTO IN PROGRESS
Comment					
RCPFL DEFINITIONS					
End of Comment					
		1... ....		RCPFLSOM	"B'10000000" SPOOLING OUT WORKING ON MLWTO
		.1.. ....		RCPFLSOE	"B'01000000" SPOOLING OUT DISASTROUS ERROR FLAG
		..1. ....		RCPFLNOM	"B'00100000" NODAL OUT WORKING ON MLWTO
Comment					
EQU B'00010000' Reserved for future use					
End of Comment					
		.... 1...		RCPFLTML	"B'00001000" WE ARE CURRENTLY TRANSMITTING MLWTO
		.... .1..		RCPFLT	"B'00000100" TEMPORARY CONDITION INDICATOR
		.... .1.		RCPFLMXM	"B'00000010" &SPOLMSG MAX MSG RECS REACHED
		.... ...1		RCPFLIOE	"B'00000001" I/O ERROR SPOOLING RMT MSGS
Comment					
RCPMF DEFINITIONS					
End of Comment					
		1... ....		RCPMFSPF	"B'10000000" MTTRVAL error
		.1.. ....		RCPMFRRD	"B'01000000" ERROR READING REMOTE MSGS
		..1. ....		RCPMFRWR	"B'00100000" ERROR WRITING REMOTE MSGS
		...1 ....		RCPMFMNU	"B'00010000" SPOOL MEM NOT UP-MSGS DISCARDED
		.... 1...		RCPMFMAX	"B'00001000" MSG LIMIT REACHED ON INPUT
		.... .1..		RCPMFMQE	"B'00000100" MAX MSGS TO SPOOL Q EXCEEDED
		.... .1.		RCPMFMSMP	"B'00000010" SPOOL OUT ERROR MESSAGE PENDING
		.... ...1		RCPMFIMP	"B'00000001" INPUT ERROR MESSAGE PENDING
		.111 1111		RCPMFPND	"B'01111111" Bits used in RCPMF - used to determine if a message needs to be issued.
Comment					
RCPFI DEFINITIONS					
End of Comment					
		1... ....		RCPFIIIE	"B'10000000" ERROR ON INPUT
		.1.. ....		RCPFIM	"B'01000000" NODE INPUT IS PROCESSING MLWTO
		..1. ....		RCPFISP	"B'00100000" INPUT SPOOLING IS ACTIVE
		...1 ....		RCPFIOE	"B'00010000" INPUT SPOOL OUT ERROR
		.... 1...		RCPFISID	"B'00001000" SPOOL BUFFER HAS DATA
		.... .1..		RCPFINUL	"B'00000100" INPUT SPOOL DISCARDING
		.... .1.		RCPFIA	"B'00000010" INPUT SPOOL WRITING
		.... ...1		RCPFIR	"B'00000001" PAGE RELEASE REQUIRED ON SPOOL INPUT

**\$RCPWORK Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCE	0		RCPMSIOT	248	
RCPAUTH	4C3		RCPMSKDK	26C	
RCPBASET	1D0	0	RCPMSKEY	268	
RCPBFSZ	270		RCPMSKJK	268	
RCPCFBSS	272	20	RCPMSLBW	238	
RCPCFLG	272		RCPMSNTK	254	
RCPCFMLM	272	10	RCPMSRAT	250	
RCPCFPLF	272	8	RCPMSRRD	244	
RCPCFQCH	272	80	RCPMSRTE	260	
RCPCFVTS	272	40	RCPMSXIT	1D8	
RCPDCT	138	0	RCPMTTR	4DC	
RCPDMXCN	1E4	1D8	RCPNMR	411	0
RCPDSTWK	7D4		RCPNMRP	40E	0
RCPEXTPL	1C8	0	RCPNODCT	4C4	
RCPFI	799		RCPNORT	40B	
RCPFIA	89C	2	RCPNPMB	4C8	
RCPFIIE	89C	80	RCPNCEWS	89C	768
RCPFIM	89C	40	RCPREGS	838	
RCPFINUL	89C	4	RCPREGSV	1E8	
RCPFIOE	89C	10	RCPRESV	7A0	
RCPFIR	89C	1	RCPROUT	264	
RCPFISID	89C	8	RCPRSV1	7E6	
RCPFISP	89C	20	RCPSAFPM	758	
RCPFL	798		RCPSCQAD	79C	
RCPFLIOE	89C	1	RCPSCQOF	79A	0
RCPFLMXM	89C	2	RCPSICTA	4D6	0
RCPFLNOM	89C	20	RCPSICTR	4D4	0
RCPFLSOE	89C	40	RCPSID	4E0	0
RCPFLSOM	89C	80	RCPSIDEL	7F4	0
RCPFLT	89C	4	RCPSIDL	834	
RCPFLTML	89C	8	RCPSIIN	4CC	
RCPGM	7E8		RCPSILR	4D0	
RCPIN	530	0	RCPSIQCT	512	0
RCPINA	608	0	RCPSIQSQ	4D8	
RCPININ	524		RCPSIRT	4E2	0
RCPINLNE	528		RCPSIXIT	1E0	
RCPINTME	52C		RCPSOBAK	274	
RCPIOCTR	51E	0	RCPSOCTR	282	0
RCPIOFRE	51C	0	RCPSOFRE	280	0
RCPIOLR	518		RCPSOLR	27C	
RCPIOOUT	514		RCPSOOUT	278	
RCPIOT	228		RCPSOQSE	284	
RCPIOTTR	520		RCPSORT	408	
RCPIOXIT	1E4		RCPSOXIT	1DC	
RCPJQTM	1C4		RCPSWELQ	22C	
RCPLSAV	7A4	0	RCPSYMBF	89C	
RCPMF	1C2		RCPSYMBP	878	
RCPMFIMP	89C	1	RCPSYMLN	894	
RCPMFMAX	89C	8	RCPSYMRC	898	
RCPMFMNU	89C	10	RCPTABBA	288	0
RCPMFMQE	89C	4	RCPTABBL	284	C
RCPMFPND	89C	7F	RCPTOKN	708	0
RCPMFRRD	89C	40	RCPWF	1C0	
RCPMFRWR	89C	20	RCPWFBUF	89C	4
RCPMFSMP	89C	2	RCPWFCMB	89C	80
RCPMFSPF	89C	80	RCPWFIBF	89C	2
RCPMSBAT	24C		RCPWFPCB	89C	8
RCPMSBL	234		RCPWFPSO	89C	1
RCPMSBSV	240		RCPWFQO	89C	20
RCPMSB1	230		RCPWFQX	89C	40
RCPMSCB	23C		RCPWFTRK	89C	10
RCPMSCMB	258		RCPWF2	1C1	0
RCPMSHDR	25C		RCPWF2ML	89C	40

## \$RCPWORK Cross Reference

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
RCPXIT	1D8	
RCPXSAV	7E8	

---

## \$RDRWORK Information

### \$RDRWORK Programming Interface information

Programming Interface information

#### \$RDRWORK

The following fields are **NOT** programming interface information:

- RDWOCT
- RDWRJCB
- RDWRJCBS

End of Programming Interface information

## Heading Information

### \$RDRWORK Heading Information

**Common Name:** JES2 Input Services PCE Work Area  
**Macro ID:** \$RDRWORK  
**DSECT Name:** PCE (\$RDRWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
Offset: PCEEYE-PCE  
Length: 4

**Storage Attributes:** Subpool: See \$PCE  
Key: See \$PCE  
Residency: See \$PCE

**Size:** See RDWLEN for normal devices.  
See RDWRJELN for RJE Input devices.  
See RDWNJRLN for Network Job Receivers.  
See RDWILEN for Internal Readers.  
The overall length of the PCE is stored in field PCELENG.

**Created by:** Created by \$PCEDYN during JES2 initialization for most input services PCEs. PCEs for remote readers are an exception, they are created by \$PCEDYN when the remote for that reader is started.

**Pointed to by:** \$RDRPCE field of the \$HCT data area  
\$INRPCE field of the \$HCT data area  
\$TPRDPCE field of the \$HCT data area  
\$NJR PCE field of the \$HCT data area  
\$OJRPCE field of the \$HCT data area  
\$NRRPCE field of the \$HCT data area  
\$EXEC PCE field of the \$HCT data area  
\$TRCPCE field of the \$HCT data area  
\$OUTPCE field of the \$HCT data area  
\$STACPCE field of the \$HCT data area  
\$RESMPCE field of the \$HCT data area  
\$SPOLPCE field of the \$HCT data area  
DCTPCE field of the \$DCT data area  
See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this work area are used by a JES2 Input Service Processor and by its support routines and exits. \$RDRWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$RDRWORK are actually part of the PCE DSECT, but only map PCEs with the value PCERDRID, PCENJRID or PCENRRID in the second byte of field PCEID, and a few other PCE work areas that also contain the \$RDRWORK area; \$COMWORK, \$OUTWORK, \$PSOWORK, \$SPNWORK, \$TLGWORK, \$XEQWORK, and \$SPIWORK.

This PCE is device related. This processor type has a one-to-one relationship to devices. Field PCEDCT points to a Device Control Table (DCT) and field DCTPCE in that DCT points to this PCE.



\$RDRWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	BITSTRING	1	RDWSW1	Reader switches
		1... ....		RDW1EOF	"B'10000000" End-of-file switch
		.1... ....		RDW1DRN	"B'01000000" SPOF recver being drained
313	(139)	BITSTRING	2		Reserved for future use
315	(13B)	BITSTRING	1	RDWFLAGX	READER EXITS FLAG BYTE
316	(13C)	SIGNED	4	RDWIBEND	ADDRESS OF LAST CARD IN INPUT BUFFER
320	(140)	ADDRESS	4	RDWIBSTD	INPUT BUFFER START DISPLACEMENT

Comment

RPUT parameter lists

End of Comment

324	(144)	BITSTRING	48	RDWPJCL	JCLIN data set parm list
372	(174)	BITSTRING	48	RDWPSYSN	SYSIN data set parm list
420	(1A4)	CHARACTER	10	RDWDEST	DESTINATION WORK AREA
430	(1AE)	BITSTRING	2		Reserved space
432	(1B0)	ADDRESS	4	RDWSMFB	SMF BUFFER POINTER
440	(1B8)	DBL WORD	8	RDWCWKAR (0)	Common work area

Comment

RDWRECCT and RDWCURRC are for SDSF use

End of Comment

440	(1B8)	X'218'	0	RDWCURRC	"(JRWCUREC-JRW)+RDWCWKAR,4,C'F" Count of records received
440	(1B8)	X'214'	0	RDWRECCT	"(JRWRCOUN-JRW)+RDWCWKAR,4,C'F" Total input records (from NJH)
1728	(6C0)	ADDRESS	4	RDWIBCUR	Input buffer current card
1732	(6C4)	SIGNED	4	(3)	RESERVED
1732	(6C4)	X'6D0'	0	RDWORG	*** START OF READER EXTENSIONS

Comment

WORK AREA FOR RJE INPUT DEVICES

End of Comment

1744	(6D0)	CHARACTER	260	RDWRJECD (0)	REMOTE READER INPUT AREA
1744	(6D0)	CHARACTER	256	RDWRCARD	MAX RJE CARD IMAGE SIZE
2000	(7D0)	CHARACTER	4	RDWRCDXT	RESERVED FOR XTRA WK SPACE
2004	(7D4)	BITSTRING	8		Reserved
2004	(7D4)	X'6A4'	0	RDWRJELN	** -PCEWORK" LENGTH OF RJE INPUT PCE WORK AREA

Comment

WORK AREA FOR NETWORK JOB RECEIVERS

End of Comment

1744	(6D0)	BITSTRING	260	RDWNJRCD (0)	JOB RECEIVER INPUT AREA
1744	(6D0)	BITSTRING	256	RDWNCARD	MAXIMUM NJE HEADER SIZE
2000	(7D0)	BITSTRING	4	RDWNCDXT	RESERVED FOR XTRA WK SPACE
2004	(7D4)	ADDRESS	4	RDWNIBUF	Address of 32K input buffer
2008	(7D8)	BITSTRING	1	RDWNSRCB	SRCB work area
2009	(7D9)	BITSTRING	3		Reserved
2009	(7D9)	X'6A4'	0	RDWNJRLN	** -PCEWORK" JOB RECEIVER PCE WORK AREA LENGTH
2009	(7D9)	X'6A4'	0	RDWLEN	** -PCEWORK" Length of normal input PCE work area

# \$RDRWORK Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Ensure lengths of extensions are equal. Assembly errors will show in following SCONS if not equal.					
End of Comment					
2012	(7DC)	ADDRESS	2	(0)	
2012	(7DC)	ADDRESS	2	(0)	

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RDWPUTPL	, RPUT parameter list DSECT
0	(0)	ADDRESS	4	RDWPLBUF	Current output buffer
4	(4)	ADDRESS	4	RDWPLBNX	Next card in buffer addr
8	(8)	ADDRESS	4	RDWPLBND	End of output buffer addr
12	(C)	BITSTRING	6	RDWPLCTK	Current buffer track address (MQTR)
18	(12)	BITSTRING	2		Reserved
20	(14)	ADDRESS	4	RDWPLPDB	Associated PDDB
24	(18)	BITSTRING	6	RDWPLIMQ	MQTR of IOT containing PDDB
30	(1E)	BITSTRING	2		Reserved
32	(20)	ADDRESS	4	RDWPLREC	Record pointer
36	(24)	BITSTRING	1	RDWPLCC	Carriage control
37	(25)	BITSTRING	1	RDWPLLR	LRC to be used
38	(26)	SIGNED	2	RDWPLRCL	Record LRECL
38	(26)	X'26'	0	RDWPLSLN	"RDWPLRCL,1" SCR length (SCRTLENG)
38	(26)	X'27'	0	RDWPLSTY	"RDWPLRCL+1,1" SCR Type (SCRSTYPE)
40	(28)	BITSTRING	1	RDWPLFG1	Control flags
		1... ....		RDWPL1TR	"B'10000000" Truncate current buffer
		.1.. ....		RDWPL1CL	"B'01000000" Truncate and close steam
		..1. ....		RDWPL1JL	"B'00100000" JCL data set (will be read by converter)

Comment

-----  
Work areas used by RPUT and close processing  
-----

End of Comment

41	(29)	BITSTRING	1	RDWPLFG2	Data flag byte
		1... ....		RDWPL2FR	"B'10000000" At least one record proc
		.1.. ....		RDWPL2VA	"B'01000000" Record sized vary
		..1. ....		RDWPL2CA	"B'00100000" ASA control character found
		...1 ....		RDWPL2CM	"B'00010000" Machine control chars found
42	(2A)	SIGNED	2	RDWPLMLR	Max LRECL seen by RPUT
44	(2C)	SIGNED	2	RDWPLCLN	Blank truncated card length
48	(30)	SIGNED	4	(0)	Align
48	(30)	X'30'	0	RDWPLSIZ	"*-RDWPOTPL" Parameter list size
		...1 ....		RDWSIOCS	"X'10" SIO CSW STORED BIT
		..11 ....		RDWSIOCC	"X'30" SIO COMPLETION CODE BITS

**\$RDRWORK Cross Reference**

Name	Hex Offset	Hex Value
PCE	0	
RDWCURRC	1B8	218
RDWCWKAR	1B8	
RDWDEST	1A4	
RDWFLAGX	13B	
RDWIBCUR	6C0	
RDWIBEND	13C	
RDWIBSTD	140	
RDWLEN	7D9	6A4
RDWNCARD	6D0	
RDWNCDXT	7D0	
RDWNIBUF	7D4	
RDWNJRCD	6D0	
RDWNJRLN	7D9	6A4
RDWNSRCB	7D8	
RDWORG	6C4	6D0
RDWPJCL	144	
RDWPLBND	8	
RDWPLBNX	4	
RDWPLBUF	0	
RDWPLCC	24	
RDWPLCLN	2C	
RDWPLCTK	C	
RDWPLFG1	28	
RDWPLFG2	29	
RDWPLIMQ	18	
RDWPLLRC	25	
RDWPLMLR	2A	
RDWPLPDB	14	
RDWPLRCL	26	
RDWPLREC	20	
RDWPLSIZ	30	30
RDWPLSLN	26	26
RDWPLSTY	26	27
RDWPL1CL	28	40
RDWPL1JL	28	20
RDWPL1TR	28	80
RDWPL2CA	29	20
RDWPL2CM	29	10
RDWPL2FR	29	80
RDWPL2VA	29	40
RDWPSYSN	174	
RDWPUTPL	0	
RDWRCARD	6D0	
RDWRCDXT	7D0	
RDWRECCT	1B8	214
RDWRJECD	6D0	
RDWRJELN	7D4	6A4
RDWSIOCC	30	30
RDWSIOCS	30	10
RDWSMFB	1B0	
RDWSW1	138	
RDW1DRN	138	40
RDW1EOF	138	80

## \$RDRWORK Cross Reference

## \$RECY Information

### \$RECY Heading Information

**Common Name:** Recovery CTENT  
**Macro ID:** \$RECY  
**DSECT Name:** RECYDAS  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** The pool of RECY CTENTs are preceded by an eyecatcher '\*\*\*RECY POOL\*\*\*' in the header for the pool.  
 Offset: HDPID-HDP  
 Length: 13

**Storage Attributes:** Subpool: 0,dataspace  
 Key: 1  
 Residency: Virtual storage is anywhere (below or above 16M) in the JES2 address space. Real storage is anywhere.

**Size:** \$RCDSIZE in the \$HCT data contains the size of a RECYDAS array element.

**Created by:** JES2 initialization allocates storage for the RECY CTENTS in JES2 private. The checkpoint versions subtask creates copies of the RECYDAS checkpoint versions dataspace.

**Pointed to by:** The \$RCDFRST field of the \$HCT data area points to the first RECY in the JES2 private area.

**Serialization:** None

**Function:** Represents recovery data for a control block when rebuilding the control block.

The RECYDAS is used to rebuild individual DASes and their chaining on the track queue, work queue, and/or neither queue when an error has been detected during validation performed at initialization.

### \$RECY Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RECYDAS	
0	(0)	CHARACTER	6	RCDVOLID	EBCDIC Volser id
6	(6)	BITSTRING	1	RCDFLAG1	Shadow copy of DASFLAG
7	(7)	BITSTRING	1	RCDFLAG2	Shadow copy of DASFLAG2
8	(8)	BITSTRING	1	RCDFLAG3	Shadow copy of DASFLAG3
9	(9)	BITSTRING	1	RCDFLAG4	Shadow copy of DASFLAG4
10	(A)	BITSTRING	1	RCDFLAG5	Shadow copy of DASFLAG5
11	(B)	BITSTRING	1	RCDFLAG6	Shadow copy of DASFLAG6
12	(C)	BITSTRING	1	RCD7PHAS	Shadow copy of DASPHAS7
13	(D)	BITSTRING	1	RCDFLAG8	Shadow copy of DASFLAG8
14	(E)	SIGNED	2	RCDNOTGP	Number of tracks per group
16	(10)	SIGNED	4	RCDTRK (0)	Valid track range
16	(10)	BITSTRING	2	RCDLOTRK	Lower limit
18	(12)	BITSTRING	2	RCDUPTRK	Upper limit
20	(14)	SIGNED	4	RCDSTRK	Start track for extent
24	(18)	SIGNED	4	RCDMAPO	Offset from \$TGMAP for this extent
28	(1C)	SIGNED	4	(0)	Ensure fullword alignment
28	(1C)	BITSTRING	4	RCDALOCS	Sys. with ext alloc'ed
32	(20)	BITSTRING	4	RCDERROR	Sys. error during alloc
36	(24)	BITSTRING	32	RCDERCDE	Error codes for systems

## \$RECY Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
68	(44)	SIGNED	4	RCDTGNUM	Num of trackgroup in extent
72	(48)	ADDRESS	1	RCDMIGTR	SYSID of migrator system - system housing the migrator subtask. DASMIGTR
73	(49)	CHARACTER	6	RCDTARG	Represents target volser for spool migration. DASTARG.
79	(4F)	BITSTRING	1	RCDFLAGA	Shadow copy of DASFLAGA
80	(50)	CHARACTER	8		Reserved for future use
88	(58)	DBL WORD	8	(0)	Align end of RECY
88	(58)	SIGNED	4	RCDBSEND (0)	End of base section

Comment

-----  
 The INTERNAL format of the RECY is different depending on the mode of the JES2 checkpoint:  
 - If the JES2 checkpoint is in z/OS 1.2 mode, each checkpointed RCD entry DOES NOT include the expanded area.  
 - If the JES2 checkpoint is in z/OS 1.11 mode, each checkpointed RCD entry DOES include the expanded area.  
 -----

End of Comment

88	(58)	X'58'	0	RCDLEN_Z2	"*-RECYDAS" Length of checkpointed z/OS 1.2 mode RECY.
88	(58)	CHARACTER	44	RCDDSN	Data set name for spool data set (only valid in z/OS 1.11 mode)
132	(84)	BITSTRING	4	RCDMAPTR	Mapped track number in target volume. DASMAPPTR
132	(84)	X'88'	0	RCDZ11LN	"*-RECYDAS" Length of checkpointed z/OS 1.11 mode RECY.
132	(84)	X'1'	0	RCDVRZ2	"1" RECY control block version for z/OS version 1.2 checkpoint mode.
132	(84)	X'2'	0	RCDVRZ11	"2" RECY control block version for z/OS version 1.11 checkpoint mode.

## \$RECY Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
RCDALOCS	1C	0	RCD7PHAS	C	
RCDBSEND	58		RECYDAS	0	
RCDDSN	58	E2E8E2F1			
RCDERCDE	24	0			
RCDERROR	20	0			
RCDFLAGA	4F				
RCDFLAG1	6				
RCDFLAG2	7				
RCDFLAG3	8				
RCDFLAG4	9				
RCDFLAG5	A				
RCDFLAG6	B				
RCDFLAG8	D				
RCDLEN_Z2	58	58			
RCDLOTRK	10				
RCDMAPO	18				
RCDMAPTR	84				
RCDMIGTR	48				
RCDNOTGP	E				
RCDSTRK	14				
RCDTARG	49	40404040			
RCDTGNUM	44				
RCDTRK	10				
RCDUPTRK	12				
RCDVOLID	0	40404040			
RCDVRZ11	84	2			
RCDVRZ2	84	1			
RCDZ11LN	84	88			

---

## **\$REQJID Information**

### **\$REQJID Programming Interface information**

Programming Interface information

**\$REQJID**

End of Programming Interface information

## Heading Information • \$REQJID Map

### \$REQJID Heading Information

**Common Name:** Request job id specifications  
**Macro ID:** \$REQJID  
**DSECT Name:** RJI  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** RJI  
 Offset: RJIID  
 Length: L'RJIID  
**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: Virtual storage is 31 bit. No restriction on real storage  
**Size:** See RJILEN  
**Created by:** JES2 Initialization  
**Pointed to by:** CCTREQJI field of the HCCT data area  
**Serialization:** Normal PCE dispatch serialization  
**Function:** Specifications for Request jobid jobs. See below for the information stored.

### \$REQJID Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RJI	
0	(0)	CHARACTER	4	RJIID	
4	(4)	BITSTRING	6	RJIJLOG	JES log control
10	(A)	BITSTRING	1	RJIFLAG1	Flag byte
		1... ....		RJI1TY6	"B'10000000" Create type 6 SMF records
		.1.. ....		RJI1TY26	"B'01000000" Create type 26 SMF records
		..1. ....		RJI1UJP	"B'00100000" Allow IEFUJP exit
10	(A)	X'B'	0	RJILEN	"*-RJI"



---

## **\$RESNAM Information**

### **\$RESNAM Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$RESNAM**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$RESNAM Map

### \$RESNAM Heading Information

**Common Name:** JES2 Resource Name Mapping  
**Macro ID:** \$RESNAM  
**DSECT Name:** RESNAM  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: N/A  
 Key: N/A  
 Residency: N/A  
**Size:** See RESJLEN and RESILEN  
**Created by:** \$RESNAME is normally included as part of another control block, for example \$WAVE.  
**Pointed to by:** N/A  
**Serialization:** None required  
**Function:** The Resource names are built by routines in JES that require a resource name in the parameter list for RACROUTE / \$SEAS when making SAF calls.

The \$RESNAM DSECT maps the SAF Resource names.

The Resource names mapped by this DSECT have the following format:

```

JESSPOOL Resource name is
nodename.userid.jobname.jobid.GROUP.Groupname
WHERE
  nodename = The eight character nodename
  userid   = The eight character USER ID
  jobname  = The eight character JOB NAME
  Jobid    = The eight character JOBID
  GROUP    = The Constant 'GROUP'
  groupname = The eight character output group
  
```

```

ISF DEST caller Resource name is
ISFAUTH.DEST.destname
WHERE
  ISFAUTH = The 7 character constant 'ISFAUTH'
  DEST    = The 4 character constant 'DEST'
  destname = The destination name (converted by
             $DEST to character format).
  
```

### \$RESNAM Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RESNAM	
0	(0)	SIGNED	4	RESJSPL (0)	JES2 JESSPOOL RESOURCE NAME
0	(0)	CHARACTER	8	RESJNODE	Nodename
8	(8)	CHARACTER	1	RESJSEP1	separator 1
9	(9)	CHARACTER	8	RESJUSER	USER ID
17	(11)	CHARACTER	1	RESJSEP2	separator 2
18	(12)	CHARACTER	8	RESJBNM	JOB NAME
26	(1A)	CHARACTER	1	RESJSEP3	separator 3
27	(1B)	CHARACTER	8	RESJBID	JOB ID
35	(23)	CHARACTER	1	RESJSEP4	separator 4
36	(24)	CHARACTER	8	RESJRPC	Constant GROUP
44	(2C)	CHARACTER	1	RESJSEP5	separator 5
45	(2D)	CHARACTER	8	RESJRPN	Groupname
45	(2D)	X'35'	0	RESJLEN	**-RESJSPL" Length of JESPOOL resource name

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
The following mapping is used for the ISF DEST authority resource name.					
End of Comment					
0	(0)	CHARACTER	63	RESISFNM (0)	ISFAUTH resource name
0	(0)	CHARACTER	8	RESIAUTH	Constant 'ISFAUTH'
8	(8)	CHARACTER	1	RESISEP1	separator 1
9	(9)	CHARACTER	4	RESIDEST	constant 'DEST'
13	(D)	CHARACTER	1	RESISEP2	separator 2
14	(E)	CHARACTER	1	RESIDSTN	Converted destination name
14	(E)	X'20'	0	RESISUBL	*** Length without padding
32	(20)	CHARACTER	1	RESIPADN	padding
32	(20)	X'3F'	0	RESILEN	** -RESISFNM" REQUIRED LENGTH OF NAME

Comment					
The following mapping is used for the JESJOBS job modify resource name :					
ENTITY.nodename.userid.jobname					

End of Comment					
0	(0)	SIGNED	4	RESJMRNM (0)	JESJOBS JOB MODIFY RESOURCE NAME
0	(0)	CHARACTER	8	RESJMENT	Entity name
8	(8)	CHARACTER	1	RESJMSE1	separator
9	(9)	CHARACTER	8	RESJMNDE	Node name
17	(11)	CHARACTER	1	RESJMSE2	separator
18	(12)	CHARACTER	8	RESJMUID	user ID
26	(1A)	CHARACTER	1	RESJMSE3	separator
27	(1B)	CHARACTER	8	RESJMJB N	Job name
27	(1B)	X'23'	0	RESJMLEN	** -RESJMRNM" Length of JESJOBS job modify resource name

\$RESNAM Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
RESIAUTH	0	C9E2C6C1	RESJSEP3	1A	4B
RESIDEST	9	C4C5E2E3	RESJSEP4	23	4B
RESIDSTN	E	40404040	RESJSEP5	2C	4B
RESILEN	20	3F	RESJSPL	0	
RESIPADN	20	40404040	RESJUSER	9	40404040
RESISEP1	8	4B	RESNAM	0	
RESISEP2	D	4B			
RESISFNM	0				
RESISUBL	E	20			
RESJGRPC	24				
RESJGRP N	2D	40404040			
RESJJBID	1B	D1D6C2F1			
RESJJB N M	12	40404040			
RESJLEN	2D	35			
RESJMENT	0	40404040			
RESJMJB N	1B	40404040			
RESJMLEN	1B	23			
RESJMNDE	9	40404040			
RESJMRNM	0				
RESJMSE1	8	4B			
RESJMSE2	11	4B			
RESJMSE3	1A	4B			
RESJMUID	12	40404040			
RESJNODE	0	40404040			
RESJSEP1	8	4B			
RESJSEP2	11	4B			

## \$RESNAM Cross Reference

## \$RESWORK Information

### \$RESWORK Heading Information

**Common Name:** JES2 Resource Manager PCE Work Area  
**Macro ID:** \$RESWORK  
**DSECT Name:** PCE (\$RESWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol RESPCEWL for the length of this work area. The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE  
**Pointed to by:** The \$RESMPCE field of the \$HCT data area  
 See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization

**Function:** The fields in this work area are used by a JES2 Resource Manager Processor. \$RESWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$RESWORK are actually part of the PCE DSECT, but only map PCEs with the value PCERESID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$RESWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT

Comment

The following fields are used to hold information required to manage the issuance, deletion, and timing for the \$HASP050 messages They are each pointed to by the table in the RESMGR processor. These fields do not have to be in the same order, and grouped, for each resource type. However, that format may prove useful eventually.

End of Comment

312	(138)	X'3'	0	RESNTHR	"3" Number of 2-byte fields used for threshold comparison
312	(138)	ADDRESS	4	RESTBERT	HASP050 time offset - BERTs
316	(13C)	SIGNED	2	RESPBERT	HASP050 prct offset - BERTs
318	(13E)	SIGNED	2	RESOBERT (0)	Threshold prct offset BERTs
324	(144)	ADDRESS	4	RESDBERT	\$HASP050 DOM id for BERTs
328	(148)	ADDRESS	4	RETCMBS	HASP050 ISSUED TIME AND
332	(14C)	SIGNED	2	RESPCMBS	ISSUED THRESHOLD FOR CMBS
334	(14E)	SIGNED	2	RESOCMBS (0)	ISSUED THRESHOLD VALUES
340	(154)	ADDRESS	4	RESDCMBS	\$HASP050 DOM id for CMBS

## \$RESWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
344	(158)	ADDRESS	4	RESTLBUF	HASP050 ISSUED TIME AND
348	(15C)	SIGNED	2	RESPLBUF	ISSUED THRESHOLD FOR LBUF
350	(15E)	SIGNED	2	RESOLBUF (0)	ISSUED THRESHOLD VALUES
356	(164)	ADDRESS	4	RESDLBUF	\$HASP050 DOM id for LBUF
360	(168)	ADDRESS	4	RESTBFX	HASP050 ISSUED TIME AND
364	(16C)	SIGNED	2	RESPBFX	ISSUED THRESHOLD FOR CB
366	(16E)	SIGNED	2	RESOBFX (0)	ISSUED THRESHOLD VALUES
372	(174)	ADDRESS	4	RESDBFX	\$HASP050 DOM id for BFX
376	(178)	ADDRESS	4	RESTBSCB	HASP050 ISSUED TIME AND
380	(17C)	SIGNED	2	RESPBSCB	ISSUED THRESHOLD FOR BSC
382	(17E)	SIGNED	2	RESOBSCB (0)	ISSUED THRESHOLD VALUES
388	(184)	ADDRESS	4	RESDBSCB	\$HASP050 DOM id for BSC
392	(188)	ADDRESS	4	RESTVTAM	HASP050 ISSUED TIME AND
396	(18C)	SIGNED	2	RESPVTAM	ISSUED THRESHOLD FOR VTAM
398	(18E)	SIGNED	2	RESOVTAM (0)	ISSUED THRESHOLD VALUES
404	(194)	ADDRESS	4	RESDVTAM	\$HASP050 DOM id for VTAM
408	(198)	ADDRESS	4	RESTSMFB	HASP050 ISSUED TIME AND
412	(19C)	SIGNED	2	RESPSMFB	ISSUED THRESHOLD FOR SMFB
414	(19E)	SIGNED	2	RESOSMFB (0)	ISSUED THRESHOLD VALUES
420	(1A4)	ADDRESS	4	RESDSMFB	\$HASP050 DOM id for SMFB
424	(1A8)	ADDRESS	4	RESTJQES	HASP050 ISSUED TIME AND
428	(1AC)	SIGNED	2	RESPJQES	ISSUED THRESHOLD FOR JQES
430	(1AE)	SIGNED	2	RESOJQES (0)	ISSUED THRESHOLD VALUES
436	(1B4)	ADDRESS	4	RESDJQES	\$HASP050 DOM id for JQES
440	(1B8)	ADDRESS	4	RESTJOES	HASP050 ISSUED TIME AND
444	(1BC)	SIGNED	2	RESPJOES	ISSUED THRESHOLD FOR JOES
446	(1BE)	SIGNED	2	RESOJOES (0)	ISSUED THRESHOLD VALUES
452	(1C4)	ADDRESS	4	RESDJOES	\$HASP050 DOM id for JOES
456	(1C8)	ADDRESS	4	RESTJNUM	HASP050 ISSUED TIME AND
460	(1CC)	SIGNED	2	RESPJNUM	ISSUED THRESHOLD FOR JNUM
462	(1CE)	SIGNED	2	RESOJNUM (0)	ISSUED THRESHOLD VALUES
468	(1D4)	ADDRESS	4	RESDJNUM	\$HASP050 DOM id for JNUM
472	(1D8)	ADDRESS	4	RESTTGS	HASP050 ISSUED TIME AND
476	(1DC)	SIGNED	2	RESPTGS	ISSUED THRESHOLD FOR TGS
478	(1DE)	SIGNED	2	RESOTGS (0)	ISSUED THRESHOLD VALUES
484	(1E4)	ADDRESS	4	RESDTGS	\$HASP050 DOM id for TGS
488	(1E8)	ADDRESS	4	RESTTTAB	HASP050 ISSUED TIME AND
492	(1EC)	SIGNED	2	RESPTTAB	ISSUED THRESHOLD FOR TTAB
494	(1EE)	SIGNED	2	RESOTTAB (0)	ISSUED THRESHOLD VALUES
500	(1F4)	ADDRESS	4	RESDTTAB	\$HASP050 DOM id for TTAB
504	(1F8)	ADDRESS	4	RESTCKVR	HASP050 ISSUED TIME AND
508	(1FC)	SIGNED	2	RESPCKVR	ISSUED THRESHOLD FOR CKVR
510	(1FE)	SIGNED	2	RESOCKVR (0)	ISSUED THRESHOLD VALUES
516	(204)	ADDRESS	4	RESDCCKVR	\$HASP050 DOM id for CKVR
520	(208)	ADDRESS	4	RESTNHBS	HASP050 ISSUED TIME AND
524	(20C)	SIGNED	2	RESPNHBS	ISSUED THRESHOLD FOR NHBS
526	(20E)	SIGNED	2	RESONHBS (0)	ISSUED THRESHOLD VALUES
532	(214)	ADDRESS	4	RESDNHBS	\$HASP050 DOM id for NHBS
536	(218)	ADDRESS	4	RESTICES	HASP050 ISSUED TIME AND
540	(21C)	SIGNED	2	RESPICES	ISSUED THRESHOLD FOR ICES
542	(21E)	SIGNED	2	RESOICES (0)	ISSUED THRESHOLD VALUES
548	(224)	ADDRESS	4	RESDICES	\$HASP050 DOM id for ICES
552	(228)	ADDRESS	4	RESTCMDS	HASP050 ISSUED TIME AND
556	(22C)	SIGNED	2	RESPCMDS	ISSUED THRESHOLD FOR CMDS
558	(22E)	SIGNED	2	RESOCMDS (0)	ISSUED THRESHOLD VALUES
564	(234)	ADDRESS	4	RESDCMDS	\$HASP050 DOM id for CMDS

Comment

The following fields are for various other data area required by the RESMGR processor.

End of Comment

568	(238)	BITSTRING	1	RESMFLAG	RESOURCE MANAGER WORK FLAG
-----	-------	-----------	---	----------	----------------------------

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		1... ..		RESWANTQ	"B'10000000" This processor needs the CKPT to process JESPLEX resources
569	(239)	BITSTRING 1... ..	1	RESBERTD RESDUMMY	Dispers for HASP052 "B'10000000" DISPER valid settings
Comment					
See \$FLAG4 for \$BERT32 and \$BERT16 definitions					
End of Comment					
570	(23A)	BITSTRING	2		Reserved for future use
572	(23C)	BITSTRING	12	RESTQE	TIMER QUEUE ELEMENT
584	(248)	SIGNED	4		RESERVED FOR FUTURE USE
Comment					
Temporary FREE/INUSE/TOTAL counts that are computed at the start of RESMGR processing.					
End of Comment					
588	(24C)	SIGNED	4	RESTGFRE	Count of free track groups
592	(250)	SIGNED	4	RESJOFRE	Count of free JOEs
596	(254)	SIGNED	4	RESTTFRE	Count of free TTABs
600	(258)	SIGNED	4	RESTTNUM	Total number of TTABs
604	(25C)	SIGNED	4	RESJNFRE	Count of free job #'s
608	(260)	SIGNED	4	RESJNNUM	Count of assignable job #'s
612	(264)	SIGNED	4	RESBRFRE	Count of free BERTs
616	(268)	SIGNED	4	RESBRCNT	Total number of BERTs
620	(26C)	SIGNED	4	RESBABS	Absolute value used for critical BERT shortage M
624	(270)	SIGNED	4	RESBDOM	HASP052 DOMID
628	(274)	SIGNED	4	RESBSTCK	STCK last HASP052
Comment					
Work fields for \$HASP050 message					
End of Comment					
632	(278)	DBL WORD	8	RESCTIME	Current time - filled in using \$STCK
Comment					
----- \$BLDMSG MSGID=050,TYPE=WTO,SEPAR=NULL,MF=L					
End of Comment					
640	(280)	SIGNED	4	RESBM050 (0)	Control block ID
644	(284)	BITSTRING	4		Console ID
648	(288)	ADDRESS	4		Address of the CART
652	(28C)	ADDRESS	4		Pointer for JOBID
656	(290)	ADDRESS	4		Control block address
660	(294)	ADDRESS	4		Display routine address
664	(298)	ADDRESS	4	(6)	6 word work area
688	(2B0)	ADDRESS	4		Caller's R11 value
692	(2B4)	BITSTRING	2		ROUT code for Message
694	(2B6)	BITSTRING	2		Not used
696	(2B8)	CHARACTER	4		Message ID
700	(2BC)	BITSTRING	1		Indicate SEPAR=NULL
701	(2BD)	ADDRESS	1		Flag byte 1
702	(2BE)	ADDRESS	1		'DISPER'
703	(2BF)	ADDRESS	1		Flag byte 2
704	(2C0)	ADDRESS	1		Flag byte 3
705	(2C1)	CHARACTER	8		Symbolic name of dest.
713	(2C9)	BITSTRING	15		Not used
728	(2D8)	ADDRESS	4	(0)	Ensure multiple of 4
728	(2D8)	ADDRESS	2	(0)	

## \$RESWORK Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
728	(2D8)	SIGNED	4	RES50WRK (0)	
728	(2D8)	CHARACTER	4	RESTYPE	Resource name
732	(2DC)	SIGNED	2	RESRPT	Total percent required
734	(2DE)	SIGNED	2	RESRPTA	Percent actually in use
736	(2E0)	SIGNED	2	RESRPTB	Percent waited for
738	(2E2)	SIGNED	2	RESUNFRQ	Number of unfulfilled requests for resource
738	(2E2)	X'2DE'	0	RESRPTS	"RESRPTA,*-RESRPTA" Percents from last 050
740	(2E4)	SIGNED	4	RESNUM	NUMBER DEFINED
744	(2E8)	SIGNED	4	RESNUMRQ	NUMBER REQUIRED
748	(2EC)	SIGNED	4	RESNUMA	Number actually in use
752	(2F0)	SIGNED	4	RESNUMB	Number waited for
756	(2F4)	SIGNED	4	RESNUMC	Largest unfulfilled request for resource
760	(2F8)	BITSTRING	1	RESDISPR	DISPER for \$HASP050 message
		1... ....		RESSHORT	"X'80" DISPER for short form
		.1... ....		RESLONG	"X'40" DISPER for long form
760	(2F8)	X'21'	0	RES50LEN	** -RES50WRK"
760	(2F8)	X'1C1'	0	RESPCEWL	** -PCEWORK" LENGTH OF RESOURCE PCE WORK AREA

## \$RESWORK Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PCE	0		RESOCMBS	14E	
RESBABS	26C		RESOCMDS	22E	
RESBDDOM	270		RESOICES	21E	
RESBERTD	239		RESOJNUM	1CE	
RESBM050	280	C2D3C440	RESOJOES	1BE	
RESBRCNT	268		RESOJQES	1AE	
RESBRFRE	264		RESOLBUF	15E	
RESBSTCK	274		RESONHBS	20E	
RESCTIME	278		RESOSMFB	19E	
RESDBERT	144		RESOTGS	1DE	
RESDBFX	174		RESOTTAB	1EE	
RESDBSCB	184		RESOVTAM	18E	
RESDCCKVR	204		RESPBERT	13C	
RESDCMBS	154		RESPBFX	16C	
RESDCMDS	234		RESPBSCB	17C	
RESDICES	224		RESPCEWL	2F8	1C1
RESDISPR	2F8		RESPCKVR	1FC	
RESDJNUM	1D4		RESPCMBS	14C	
RESDJQES	1C4		RESPCMDS	22C	
RESDJQES	1B4		RESPICES	21C	
RESDLBUF	164		RESPJNUM	1CC	
RESDNHBS	214		RESPJOES	1BC	
RESDSMFB	1A4		RESPJQES	1AC	
RESDTGS	1E4		RESPLBUF	15C	
RESDTTAB	1F4		RESPNHBS	20C	
RESDUMMY	239	80	RESRPT	2DC	
RESDVTAM	194		RESRPTA	2DE	
RESJNFRE	25C		RESRPTB	2E0	
RESJNNUM	260		RESRPTS	2E2	2DE
RESJOFRE	250		RESPSMFB	19C	
RESLONG	2F8	40	RESPTGS	1DC	
RESMFLAG	238		RESPTTAB	1EC	
RESNTHR	138	3	RESPVTAM	18C	
RESNUM	2E4		RESSHORT	2F8	80
RESNUMA	2EC		RESTBERT	138	
RESNUMB	2F0		RESTBFX	168	
RESNUMC	2F4		RESTBSCB	178	
RESNUMRQ	2E8		RESTCKVR	1F8	
RESOBERT	13E		RESTCMBS	148	
RESOBFX	16E		RESTCMDS	228	
RESOBSCB	17E		RESTGFRE	24C	
RESOCKVR	1FE		RESTICES	218	



<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
RESTJNUM	1C8	
RESTJOES	1B8	
RESTJQES	1A8	
RETLBUF	158	
RESTNHBS	208	
RESTQE	23C	
RESTSMFB	198	
RESTTFRE	254	
RESTTGS	1D8	
RESTTNUM	258	
RESTTTAB	1E8	
RESTVTAM	188	
RESTYPE	2D8	
RESUNFRQ	2E2	
RESWANTQ	238	80
RES50LEN	2F8	21
RES50WRK	2D8	

## \$RESWORK Cross Reference

## \$RJCB Information

### \$RJCB Heading Information

**Common Name:** Job Reader Card Buffer  
**Macro ID:** \$RJCB  
**DSECT Name:** RJCB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'RJCB'  
 Offset: RJCBID-RJCB  
 Length: 4  
**Storage Attributes:** Subpool: 1  
 Key: 1  
 Residency: Virtual and real storage can be anywhere in private storage of the JES2 address space.  
**Size:** See RJCBLONG  
**Created by:** HASPRDR Input Reader Processor (via \$GETWORK)  
 HASPSRIP Input Service Routine (via \$GETMAIN)  
**Pointed to by:** JRWRJCB field of the \$JRW data area  
 JRWRJCBN field of the \$JRW data area  
 JRWRJCBF field of the \$JRW data area  
 JRWRJCBM field of the \$JRW data area  
 RJCBRJCB field of the \$RJCB data area  
**Serialization:** None required  
**Function:** This macro provides the mapping for the buffer that is used to contain one card that makes up a JCL statement.

### \$RJCB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	RJCB	Reader JOB Card buffer
0	(0)	CHARACTER	4	RJCBID	Control Block identifier
4	(4)	BITSTRING	1	RJCBFLGT	Flag byte (not cleared)
		1... ..		RJCBTGTW	"B'10000000" RJCB was \$GETWORKed
4	(4)	X'5'	0	RJCBCSTR	*** Start of where to clear data area on reuse
5	(5)	BITSTRING	1	RJCBFLG1	Flag byte
		1... ..		RJCB1GEN	"B'10000000" Generated card image
		.1.. ..		RJCB1XBM	"B'01000000" Add card to XBM input DS
		..1. ....		RJCB1MSG	"B'00100000" RJCBCARD contains a msg
		...1 ...		RJCB1CMB	"B'00010000" RJCBCARD contains a CMB
		.... 1..		RJCB1SCR	"B'00001000" RJCBCARD contains an SCR
6	(6)	BITSTRING	1	RJCBFLG2	Flag byte
		1... ..		RJCB2XIT	"B'10000000" Exit has seen card once
		.1.. ....		RJCB2XMO	"B'01000000" Exit modified card image
		..1. ....		RJCB2XA2	"B'00100000" Exit 2 added card image
		...1 ...		RJCB2XA4	"B'00010000" Exit 4 added card image
6	(6)	X'30'	0	RJCB2XAD	"RJCB2XA2+RJCB2XA4" Some exit added card
		.... 1..		RJCB2LOP	"B'00001000" Last operand is on card
		.... .1..		RJCB2QUO	"B'00000100" Unfinished quote at end of card
		.... ..1.		RJCB2CCM	"B'00000010" Card is a cont comment
		.... ...1		RJCB2LST	"B'00000001" Last card in statement
6	(6)	X'F'	0	RJCB2CRD	"RJCB2LOP+RJCB2QUO+RJCB2CCM+RJCB2LST" Card type
7	(7)	BITSTRING	1	RJCBFLG3	Flag byte
		1... ..		RJCB3LOC	"B'10000000" An exit modified or added card should not be sent to other nodes or offloaded
8	(8)	ADDRESS	4	RJCBRJCB	Pointer to next RJCB

## \$RJCB Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>RJCBOCRD - is the card image prior to any changes made by an exit</p> <p>RJCBCARD - is the card image with any exit changes</p> <p>RJCBMSG - if RJCB1MSG is on, RJCBMSGL contains the message length and RJCBMSG contains the a message of the format:            \$MSG nnn,'text'            text is assumed to begin with "-- "</p> <p>RJCBMCM - if RJCB1CMB is on, RJCBMCM contains a CMB to be processed by JES2. If CMBFLAGC is on the CMB contains a JES2 command, otherwise it contains a message</p>					
End of Comment					
12	(C)	SIGNED	4	(0)	Align following
12	(C)	CHARACTER	80	RJCBOCRD	Original card image
92	(5C)	CHARACTER	80	RJCBCARD	Card image
12	(C)	ADDRESS	2	RJCBMSGL	Message length
14	(E)	CHARACTER	120	RJCBMSG	Message (\$MSGID nnn,'text')
134	(86)	BITSTRING	1	RJCBMSGF	Message flags
		1... ....		RJCBMFRJ	"B'10000000" Send message to RJE
		.1.. ....		RJCBMFNJ	"B'01000000" Send message to NJE
		..1. ....		RJCBMFJC	"B'00100000" Write message to JCLIN
		...1 ....		RJCBMFEX	"B'00010000" Extend local msg with source information
		.... 1..		RJCBMF1S	"B'00001000" Add msg to front of queue
		.... .1.		RJCBMERR	"B'00000100" Message is for an error
		.... ..1.		RJCBMWAR	"B'00000010" Message is a warning
135	(87)	BITSTRING	1		Reserved
12	(C)	BITSTRING	200	RJCBMCM	CMB for JES2 processing
12	(C)	BITSTRING	80	RJCBSCR	SCR to be written
92	(5C)	ADDRESS	4	RJCBPDDB	Address of related PDDB
96	(60)	SIGNED	2	RJCBSROF	Off in SCR for chain track
98	(62)	BITSTRING	1	RJCBSCRFB	SCR flag bytes
		1... ....		RJCBSFTR	"B'10000000" Trunc buffer after SCR
216	(D8)	DBL WORD	8	(0)	Alignment
216	(D8)	X'5'	0	RJCBCLR	"RJCBSTR,*-RJCBSTR" Area of RJCB to be cleared
216	(D8)	X'D8'	0	RJCBLENG	"*-RJCB" Length of RJCB in bytes
216	(D8)	X'36'	0	RJCBWORD	"RJCBLENG/4" Length of RJCB in words

## \$RJCB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
RJCB	0		RJCBMSGF	86	
RJCBCARD	5C		RJCBMSGL	C	
RJCBCLR	D8	5	RJCBMWAR	86	2
RJCBMCM	C		RJCBOCRD	C	
RJCBSTR	4	5	RJCBPDDB	5C	
RJCBFLGT	4		RJCBRJCB	8	
RJCBFLG1	5		RJCBSCR	C	
RJCBFLG2	6		RJCBSCRFB	62	
RJCBFLG3	7		RJCBSFTR	62	80
RJCBID	0		RJCBSROF	60	
RJCBLENG	D8	D8	RJCBTGTW	4	80
RJCBMERR	86	4	RJCBWORD	D8	36
RJCBMFEX	86	10	RJCB1CMB	5	10
RJCBMFJC	86	20	RJCB1GEN	5	80
RJCBMFNJ	86	40	RJCB1MSG	5	20
RJCBMFRJ	86	80	RJCB1SCR	5	8
RJCBMF1S	86	8	RJCB1XBM	5	40
RJCBMSG	E		RJCB2CCM	6	2

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
RJCB2CRD	6	F
RJCB2LOP	6	8
RJCB2LST	6	1
RJCB2QUO	6	4
RJCB2XAD	6	30
RJCB2XA2	6	20
RJCB2XA4	6	10
RJCB2XIT	6	80
RJCB2XMO	6	40
RJCB3LOC	7	80



## \$ROTT Information

### \$ROTT Heading Information

**Common Name:** ROTT  
**Macro ID:** \$ROTT  
**DSECT Name:** ROTT, ROTE  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** ROTT  
 Offset: ROTID  
 Length: L'ROTID

**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: Virtual and real anywhere

**Size:** This DSECT defines a number of tables. Each table has a fixed size header (length is ROTSIZE) and then the actual trace data. The size of each entry is dependant on the data area being traced. The following table list the sizes.

Table	Anchor	SIZE
JQEs	\$ROTJQE	ROTSIZE + \$ROTQNUM * ROTJSIZE
JOEs	\$ROTJOE	ROTSIZE + \$ROTONUM * ROTJSIZE
Dispatcher	\$ROTDISP	ROTSIZE + \$ROTDNUM * ROTDSIZE

**Created by:** HASPIRDA  
**Pointed to by:** \$ROTJQE field of the HCT (for the JQE table)  
 \$ROTJOE field of the HCT (for the JOE table)  
 \$ROTDISP field of the HCT (for the \$WAIT table)

**Serialization:** None - table is updated only by the main JES2 TCB.  
**Function:** This DSECT maps the CTRACE rolling trace entries used by JES2. The following CTRACE SUBs exist in JES2:  
 JQE - JQE services (\$Qxxxx services and \$DOGBERT)  
 JOE - JOE services (\$#xxxx services)  
 DISP - \$WAIT, PCE dispatch and MVS WAIT services  
 The intent of these traces is to provide a history of what happened in JES2 to the various control blocks to aid in debugging JES2 problems.

### \$ROTT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	ROTT	,
Comment					
Table control					
End of Comment					
0	(0)	CHARACTER	4	ROTID	Eye catcher
4	(4)	SIGNED	1	ROTVER	Version
4	(4)	X'1'	0	ROTCVER	"1" Current version
5	(5)	BITSTRING	1	ROTFLAG1	Flags
6	(6)	SIGNED	2	ROTELEN	Length of an element
8	(8)	ADDRESS	4	ROTFIRST	Addr of first element
12	(C)	ADDRESS	4	ROTLAST	Addr of last element

# \$ROTT Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
16	(10)	ADDRESS	4	ROTCURR	Addr of current element
24	(18)	DBL WORD	8	ROTELEM (0)	Element definition begins here
24	(18)	X'18'	0	ROTSIZE	""-ROTT" Length of header portion

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	ROTE	, Map trace element

Comment

Information common to all trace entries  
 Goal: Make this information identical to the  
 CTRACE element header

End of Comment

0	(0)	SIGNED	2	ROTELENP	Length of trace element
2	(2)	SIGNED	2	ROTEOFF	Data offset
4	(4)	SIGNED	4	ROTEFMTI	Format ID key
8	(8)	BITSTRING	8	ROTE TIME	TOD clock value
16	(10)	BITSTRING	1	ROTE DATA (0)	Variable data goes here

Comment

JES2 component information common to all elements

End of Comment

16	(10)	ADDRESS	4	ROTEPCE	PCE address
20	(14)	BITSTRING	1	ROTEFLG1	Flags
		1... ..		ROTE1ART	"B'10000000" Artificial JQE
		.1... ..		ROTE1JOA	"B'01000000" Artificial JOE
21	(15)	BITSTRING	1	ROTEEXIT	Current exit number
22	(16)	BITSTRING	2		Reserved for future use
24	(18)	SIGNED	4	ROTEJNUM	Job number
28	(1C)	ADDRESS	4	ROTEOFFS	Offset of JQE/JOE

Comment

Service IDs represent the service (e.g.\$QMOD,  
 \$#REM) which caused the trace entry to be built.

End of Comment

32	(20)	BITSTRING	1	ROTESERV	Service id
		.... ..		ROTQSRV	"X'00" First JQE service id
		.... ..		ROTQADD	"X'00" \$QADD
		.... ..1		ROTQPUT	"X'01" \$QPUT
		.... ..1.		ROTQREM	"X'02" \$QREM
		.... ..11		ROTQMOD	"X'03" \$QMOD
		.... ..1..		ROTQJIX	"X'04" \$QJIX (alloc new number)
		.... ..1.1		ROTQJIXS	"X'05" \$QJIX (swap job numbers)
		.... ..11.		ROTGETJL	"X'06" \$GETJLOK
		.... ..111		ROTFREJL	"X'07" \$FREJLOK
		.... 1...		ROTQRBDC	"X'08" \$QRBDC (add to queue)
		.... 1..1		ROTQBUSY	"X'09" \$QBUSY
		.... 1.1.		ROTGETLO	"X'0A" \$GETLOKW
		.... 1.11		ROTFRELO	"X'0B" \$FRELOKW
		.... 11..		ROTDGJQ	"X'0C" \$DOGJQE



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		...1 ....		ROT#SRV	"X'10" First JOE service id
		...1 ....		ROT#ADD	"X'10" \$#ADD
		...1 ...1		ROT#PUT	"X'11" \$#PUT
		...1 ..1.		ROT#REM	"X'12" \$#REM
		...1 ..11		ROT#MOD	"X'13" \$#MOD
		...1 ..1..		ROT#RBDC	"X'14" \$#RBDCCHK (add to queue)
		...1 1..1		ROT#BUSY	"X'19" \$#BUSY
		...1 1.1.		ROT#GET	"X'1A" \$#GET
		...1 1.11		ROT#CAN	"X'1B" \$#CAN
		...1 11..		ROT#REP	"X'1C" \$#REP
		...1 11.1		ROTDGJO	"X'1D" \$DOGJOE
		..1. ....		ROTDSRV	"X'20" First \$WAIT service id
		..1. ....		ROTWAIT	"X'20" \$WAIT
		..1. ...1		ROTDISP	"X'21" Dispatch
		..1. ...1		ROTMVSWA	"X'22" MVS WAIT
33	(21)	BITSTRING	1	ROTESRV2	Sub service number
34	(22)	BITSTRING	2		Reserved for future use

Comment

JES2 component information common to JOE and JQEs

End of Comment

36	(24)	SIGNED	4	ROTESPEC (0)	Start of 'specific' data
36	(24)	ADDRESS	4	ROTECALR	Caller of service
40	(28)	BITSTRING	1	ROTEOQUE	Original queue (or class)
41	(29)	BITSTRING	1	ROTENQUE	New queue (or class)
42	(2A)	BITSTRING	1	ROTEBUSY	Busy byte
43	(2B)	BITSTRING	1	ROTQLOCK	Lock (JQE only)
43	(2B)	X'2B'	0	ROT#TYPE	"ROTQLOCK,1,C'X" Type (JOE only)
48	(30)	DBL WORD	8	ROTEND (0)	Ensure entry ends on double word boundary
46	(2E)	SIGNED	2	ROTELENE	Length of element
46	(2E)	X'30'	0	ROTEJSIZ	"*-ROTE" Size of one entry

Comment

Field for a dispatcher trace entry  
 Note ROTEEVNT is incremented by one to simplify  
 IPCS code.

End of Comment

36	(24)	BITSTRING	1	ROTEEVNT	Event byte
37	(25)	BITSTRING	1	ROTRESO	Resource byte
38	(26)	BITSTRING	1	ROTEWFG1	\$WAIT parm \$WTFLAG1
39	(27)	BITSTRING	1		Reserved
40	(28)	CHARACTER	8	ROTECSCT	CSECT
48	(30)	CHARACTER	8	ROTESEQ	Sequence
56	(38)	BITSTRING	8	ROTEWTME	\$WAIT time or Run time (In microseconds)
64	(40)	BITSTRING	8	ROTEWCPU	CPU Used (\$WAIT Entry) (In microseconds)
64	(40)	BITSTRING	2	ROTEPSTR	\$POST reason (\$DISP entry) See PPBLPOST
88	(58)	DBL WORD	8	ROTEND2 (0)	Ensure entry ends on double word boundary
86	(56)	SIGNED	2	ROTELEN	Length of element
86	(56)	X'58'	0	ROTEDSIZ	"*-ROTE" Size of one entry

## \$ROTT Cross Reference

## \$ROTT Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
ROT#ADD	20	10	ROTQBUSY	20	9
ROT#BUSY	20	19	ROTQJIX	20	4
ROT#CAN	20	1B	ROTQJIXS	20	5
ROT#GET	20	1A	ROTQLOCK	2B	
ROT#MOD	20	13	ROTQMOD	20	3
ROT#PUT	20	11	ROTQPUT	20	1
ROT#RBDC	20	14	ROTQRBDC	20	8
ROT#REM	20	12	ROTQREM	20	2
ROT#REP	20	1C	ROTQSRV	20	0
ROT#SRV	20	10	ROTSIZE	18	18
ROT#TYPE	2B	2B	ROTT	0	
ROTCURR	10		ROTVR	4	
ROTCVER	4	1	ROTWAIT	20	20
ROTDISP	20	21			
ROTDGJO	20	1D			
ROTDGJQ	20	C			
ROTDSPV	20	20			
ROTE	0				
ROTEBUSY	2A				
ROTECALR	24				
ROTECSCT	28				
ROTEDATA	10				
ROTELEN	56				
ROTEDSIZ	56	58			
ROTEEVNT	24				
ROTEEXIT	15				
ROTEFLG1	14				
ROTEFMTI	4				
ROTEJNUM	18				
ROTEJSIZ	2E	30			
ROTELEM	18				
ROTELEN	6				
ROTELENE	2E				
ROTELENP	0				
ROTEND	30				
ROTEND2	58				
ROTENQUE	29				
ROTEOFF	2				
ROTEOFFS	1C				
ROTEOQUE	28				
ROTEPCE	10				
ROTEPSTR	40				
ROTRESO	25				
ROTESEQ	30				
ROTESERV	20				
ROTESPEC	24				
ROTESRV2	21				
ROTETIME	8				
ROTEWCPU	40				
ROTEWFG1	26				
ROTEWTME	38				
ROTE1ART	14	80			
ROTE1JOA	14	40			
ROTFIRST	8				
ROTFLAG1	5				
ROTFREJL	20	7			
ROTFRELO	20	B			
ROTGETJL	20	6			
ROTGETLO	20	A			
ROTID	0				
ROTLAST	C				
ROTMVSWA	20	22			
ROTQADD	20	0			

---

## **\$SAFINFO Information**

### **\$SAFINFO Programming Interface information**

Programming Interface information

**\$SAFINFO**

End of Programming Interface information

## Heading Information • \$SAFINFO Map

### \$SAFINFO Heading Information

**Common Name:** HASP Security Information Block  
**Macro ID:** \$SAFINFO  
**DSECT Name:** SAFINFO  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SFI '  
 Offset: SFIEYE-SAFINFO  
 Length: 4  
**Storage Attributes:** Subpool: N/A  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M) in the private storage of the JES2 address space.  
**Size:** See SFILEN  
**Created by:** JOBVALM caller and SYSOVFY caller  
**Pointed to by:** Register one upon entry to the called routine  
**Serialization:** None  
**Function:** This is the parameter list to both the JOBVALM and SYSOVFY routines. Values in this DSECT will be used to construct the RACROUTE VERIFYX, AUTH, and TOKENBLD parameter lists.

### \$SAFINFO Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SAFINFO	Security Information Parameter List
0	(0)	CHARACTER	4	SFIEYE	Control block ID
4	(4)	ADDRESS	1	SFILELEVEL	Control block version
4	(4)	X'1'	0	SFIVRSN	"1" Control block version equate
5	(5)	BITSTRING	1	SFIFLAG1	SAFINFO Flag Byte 1
		1... ....		SF11PASE	"B'10000000" NJHGPASS is encrypted
		.1.. ....		SF11NPSE	"B'01000000" NJHGNPAS is encrypted
		..1. ....		SFIRESV1	"B'00100000" Reserved for IBM dvlmt use
		...1 ....		SF11XMIT	"B'00010000" XMIT request
		.... 1...		SF11XBM	"B'00001000" This is an XBM joblet
		.... .1..		SF11NORM	"B'00000100" Get a token for the job
		.... ..1.		SF11SREQ	"B'00000010" Get a submitter for the job
		.... ...1		SF11DFLT	"B'00000001" Get an undefined user token
6	(6)	BITSTRING	1	SFIFLAG2	SAFINFO Flag Byte 2
		1... ....		SFI2STKN	"B'10000000" Submitter token in SFITOKEN or returned in JCTTOKEN
		.1.. ....		SFI2VTKN	"B'01000000" SFITOKEN is a pre-verified token (used by SYSOVFY only)
		..1. ....		SFI2VXPS	"B'00100000" JOB/OUTPUT passed VERIFYX. Used as input if SFI2VTKN set. Set on output if VX return code is 0 or 4 (used by SYSOVFY only)
7	(7)	BITSTRING	1	SFIFLAG3	SAFINFO FLAG BYTE 3
7	(7)	X'3'	0	SFI3JOB	"JQE3JOB" BATCH JOB (WHEN BITS ZERO)
7	(7)	X'1'	0	SFI3STC	"JQE3STC" FLAG FOR STC TYPE JOB
7	(7)	X'2'	0	SFI3TSU	"JQE3TSU" FLAG FOR TSU TYPE JOB
8	(8)	SIGNED	2	SFIRESCD	Error reason code for RC=4 or 8 only (else 0)
12	(C)	ADDRESS	4	SFIJCT	Address of Job Control Table
16	(10)	ADDRESS	4	SFITWA	Address Token Work Area for TOKENBLD
20	(14)	ADDRESS	4	SFILOT	Address of job's primary alloc IOT
24	(18)	ADDRESS	4	SFITOKEN	Address of input token
28	(1C)	ADDRESS	4	SFIHTOKN	Address of header token
32	(20)	SIGNED	2	SFIHTLEN	Length of header token
34	(22)	BITSTRING	1	SFIHTFLG	Header token flags
		1... ....		SFIHTJOB	"B'10000000" Header token is job token

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
35	(23)	BITSTRING	1		Reserved for future use
36	(24)	ADDRESS	4	SFIWAVE	Address of WAVE
40	(28)	CHARACTER	8	SFIPOE	Port Of Entry name
48	(30)	ADDRESS	4	SFITOKA	Submitter token (DCTTOKA)
52	(34)	BITSTRING	1	SFIDEVTP	Device type (DCTDEVTP)
53	(35)	BITSTRING	3		Reserved

Comment

The following fields are available/used by SYSOVFY only.

End of Comment

56	(38)	ADDRESS	4	SFIPDDB	Address of Pddb for verify
60	(3C)	CHARACTER	8	SFIDSNM	DSNAME or jobname from DSH

Comment

SFICONTG maps a contiguous storage area. Caution should be exercised when placing fields within this area.

End of Comment

60	(3C)	X'48'	0	SFICONTL	"SFICEND-SFICBEGN" Length of total contiguous area
60	(3C)	X'8'	0	SFIELEML	"8" Length of individual element
60	(3C)	X'44'	0	SFICBEGN	*** Beginning of contiguous area
68	(44)	ADDRESS	1	SFIUIDL	USERID length + value
68	(44)	X'45'	0	SFIUID	"SFIUIDL+1,SFIELEML,C'C" USERID for this job
77	(4D)	ADDRESS	1	SFIGRPL	GROUP length + value
77	(4D)	X'4E'	0	SFIGRP	"SFIGRPL+1,SFIELEML,C'C" GROUP for this job
86	(56)	ADDRESS	1	SFIPASL	PASSWORD length + value
86	(56)	X'57'	0	SFIPAS	"SFIPASL+1,SFIELEML,C'C" PASSWORD for this job
95	(5F)	ADDRESS	1	SFINPASL	New PASSWORD len + value
95	(5F)	X'60'	0	SFINPAS	"SFINPASL+1,SFIELEML,C'C" New PASSWORD for this job
104	(68)	ADDRESS	1	SFIXNDEL	Execution node len + val
104	(68)	X'69'	0	SFIXNDE	"SFIXNDEL+1,SFIELEML,C'C" Execution node for this job
113	(71)	ADDRESS	1	SFISNDEL	Submittor node len + val
113	(71)	X'72'	0	SFISNDE	"SFISNDEL+1,SFIELEML,C'C" Submittor node for this job
122	(7A)	ADDRESS	1	SFISUIDL	Submittor USERID len+val
122	(7A)	X'7B'	0	SFISUID	"SFISUIDL+1,SFIELEML,C'C" Submittor USERID for this job
131	(83)	ADDRESS	1	SFISGRPL	Submittor GROUP len+val
131	(83)	X'84'	0	SFISGRP	"SFISGRPL+1,SFIELEML,C'C" Submittor GROUP for this job
131	(83)	X'8C'	0	SFICEND	*** End of contiguous area
140	(8C)	CHARACTER	8	SFISECL	Security label (blanks if none)
148	(94)	ADDRESS	4		RESERVED FOR FUTURE IBM USE
152	(98)	DBL WORD	8	(0)	End of SAFINFO
152	(98)	X'98'	0	SFILEN	** -SAFINFO" Length of SAFINFO

### \$SAFINFO Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SAFINFO	0		SFIHTFLG	22	
SFICBEGN	3C	44	SFIHTJOB	22	80
SFICEND	83	8C	SFIHTLEN	20	
SFICONTL	3C	48	SFIHTOKN	1C	
SFIDEVTP	34		SFIHOT	14	
SFIDSNM	3C		SFIJCT	C	
SFIELEML	3C	8	SFILEN	98	98
SFIEYE	0	E2C6C940	SFILEVEL	4	
SFIFLAG1	5		SFINPAS	5F	60
SFIFLAG2	6		SFINPASL	5F	
SFIFLAG3	7		SFIPAS	56	57
SFIGRP	4D	4E	SFIPASL	56	
SFIGRPL	4D		SFIPDDB	38	

## \$SAFINFO Cross Reference

Name	Hex Offset	Hex Value
SFIPOE	28	
SFIRESCD	8	
SFIRESV1	5	20
SFISECL	8C	40404040
SFISGRP	83	84
SFISGRPL	83	
SFISNDE	71	72
SFISNDEL	71	
SFISUID	7A	7B
SFISUIDL	7A	
SFITOKA	30	
SFITOKEN	18	
SFITWA	10	
SFIUID	44	45
SFIUIDL	44	
SFIVRSN	4	1
SFIWAVE	24	
SFIXNDE	68	69
SFIXNDEL	68	
SFI1DFLT	5	1
SFI1NORM	5	4
SFI1NPSE	5	40
SFI1PASE	5	80
SFI1SREQ	5	2
SFI1XBM	5	8
SFI1XMIT	5	10
SFI2STKN	6	80
SFI2VTKN	6	40
SFI2VXPS	6	20
SFI3JOB	7	3
SFI3STC	7	1
SFI3TSU	7	2

## \$SAPID Information

### \$SAPID Heading Information

**Common Name:** Sysout API data area  
**Macro ID:** \$SAPID  
**DSECT Name:** SAPID and TJEV  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** \$SAP  
 Offset: SAPEYE-SAPID  
 Length: L'SAPEYE

**Storage Attributes:** Subpool: n/a  
 Key: 1  
 Residency: In the jesxSAPI data space in cpool SAPID In the jesxTJEV data space in cpool TJEV

**Size:** See SAPLEN  
 See TJELEN

**Created by:** HASCSAPI (SAPID)  
 HASPJOS (TJEV)

**Pointed to by:** SAPID - SSS2JEST field of the IAZSS2 SSOB extension and by TJESAP  
 SAPID - SAPNEXT field of the SAPID  
 TJEV - SAPTJEV field of the SAPID

**Serialization:** SAPID Compare and Swap  
 TJEV None (only used in JES2 address space)

**Function:** The SAPID contains the specifications of the SAPI user for the work desired. It also contains status information of the SAPI "thread".  
 The TJEV holds a vector of bits (one bit per potential JOE). The absence of a TJEV means that the thread has not excluded any JOE. The presense of a TJEV with the bit corresponding to a given work JOE non-zero means that that the corresponding JOE is excluded for selection by the thread.

### \$SAPID Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SAPID	, SYSOUT API Data Area
0	(0)	CHARACTER	4	SAPEYE	Eye catcher
4	(4)	BITSTRING	1	SAPFLAG1	Flags (serialized via compare and swap)
		1... ....		SAP1RQUE	"B'10000000" This SAPID managed by RQUE
		.1.. ....		SAP1GAVE	"B'01000000" The last time control was returned to this caller, work was given
		..1. ....		SAP1WSPV	"B'00100000" The SAPWSP has a WSP which has been constructed by \$WSSCAN
		...1 ....		SAP1TERM	"B'00010000" Terminate this SAPID
		.... 1...		SAP1HOT	"B'00001000" RQUE Post because of hot start
		.... .1..		SAP1PCE	"B'00000100" Being processed by SPI PCE
		.... ..1.		SAP1JWEL	"B'00000010" Non-Bulk Modify JWELs have been created
		.... ...1		SAP1BJWL	"B'00000001" JWELs for Bulk Modify have been created
5	(5)	BITSTRING	1	SAPFLAGJ	Flags representing JOE state
		1... ....		SAPJCOMP	"B'10000000" JOE has been completely processed
		.1.. ....		SAPJSAF	"B'01000000" JOE access rejected by SAF
		..1. ....		SAPJALLO	"B'00100000" JOE is allocated (to us)
		...1 ....		SAPJFINI	"B'00010000" JOE is no longer suitable

## \$SAPID Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... 1...		SAPJCTRL	"B'00001000" Do not give new JOE
		.... .1..		SAPJSYSH	"B'00000100" Put JOE in system hold
		.... ..1.		SAPJASH	"B'00000010" Put JOE in address space hold (do not give to this AS again)
		.... ...1		SAPJBUST	"B'00000001" At least one PDDDB has been busted out of this JOE
6	(6)	BITSTRING	1	SAPFLGJ2	More flags for JOE state
		1... ....		SAPJ2TH	"B'10000000" Thread hold this data set
		.1.. ....		SAPJ2OLD	"B'01000000" JOE is no longer valid
		..1. ....		SAPJ2SFP	"B'00100000" Started with first PDDDB in JOE
		.... 1...		SAPJ2SON	"B'00001000" Last data set obtained with SAF READ access
		.... .1..		SAPJ2PRI	"B'00000100" One or more data sets prio set via CSPDISP
		.... ..1.		SAPJ2CKV	"B'00000010" SAPI has written CHK - CHK valid should be set
		.... ...1		SAPJ2CKE	"B'00000001" I/O error while R/W CHK spool record
7	(7)	SIGNED	1	SAPMSTRV	Level of checkpoint in use when SAIFETCH populated (\$MSTRVER is saved here)
8	(8)	BITSTRING	1	SAPCKEY	SSI caller's key
9	(9)	BITSTRING	3		Reserved for future use

Comment

SAPSSS2 is an exact duplicate of the caller's SSOB extension. To gain addressability, specify: USING SSS2,SAPSSS2

End of Comment

12	(C)	ADDRESS	4	SAPNEXT	SAP.Addr of next SAPID in data space SERIALIZATION: Compare and Swap
16	(10)	BITSTRING	1	SAPSSS2	Shadow of caller's SSOB extension
16	(10)	X'38'	0	SAPECBP	"SAPSSS2+SSS2ECBP-SSS2,L'SSS2ECBP,C'A"
1176	(498)	BITSTRING	4	SAPROUTE	<----+ Selection route code in form nnrr
1180	(49C)	CHARACTER	8	SAPUSER	<----+ and userid
1188	(4A4)	SIGNED	4	SAPJNOLO	Low job number for selection
1192	(4A8)	SIGNED	4	SAPJNOHI	High job number for selection
1196	(4AC)	CHARACTER	8	SAPJCHLO	Low char version of job id
1204	(4B4)	CHARACTER	8	SAPJCHHI	High char version of job id
1212	(4BC)	BITSTRING	4	SAPROUTN	New route code for group requests
1216	(4C0)	CHARACTER	8	SAPUSERN	New userid for group requests
1224	(4C8)	SIGNED	4	SAPRETN	SSOBRETN equivalent
1228	(4CC)	BITSTRING	8	SAPPRIV	Copied to the SSS2
1236	(4D4)	ADDRESS	4	SAPASCB	COM.ASCB address of SAPI address space
1240	(4D8)	BITSTRING	8	SAPASCBT	Address space token
1240	(4D8)	X'4D8'	0	SAPWRASI	"SAPASCBT,L'SAPASCBT" Address space level JWEL
1248	(4E0)	ADDRESS	4	SAPTCB	UAS.TCB address of last SAPID user
1252	(4E4)	ADDRESS	4	SAPOTCB	UAS.Owning TCB address (TCB which created the SAPID)
1256	(4E8)	ADDRESS	4	SAPIOT	UAS.Address of current IOT
1260	(4EC)	ADDRESS	4	SAPIOTPA	UAS.Address of previous IOT
1264	(4F0)	ADDRESS	4	SAPJCT	UAS.Address of current JCT
1268	(4F4)	ADDRESS	4	SAPCHK	UAS.Address of current CHK
1272	(4F8)	ADDRESS	4	SAPWAVE	UAS.Address of WAVE
1276	(4FC)	ADDRESS	4	SAPBTOK	UAS.Address of SPOOL browse token
1280	(500)	ADDRESS	4	SAPACCT	UAS.Addr of accounting information
1284	(504)	ADDRESS	4	SAPDTKN	UAS.Addr of Data set token
1288	(508)	ADDRESS	4	SAPNJH	UAS.Addr of NJE job header
1292	(50C)	ADDRESS	4	SAPNDH	UAS.Addr of NJE data set header
1296	(510)	ADDRESS	4	SAPSWB	UAS.Addr of SWBTU buffer
1300	(514)	ADDRESS	4	SAPJOA	UAS.Addr of JOA
1304	(518)	ADDRESS	4	SAPTJEV	TJE.Addr of JOE exclusion vector for this thread
1308	(51C)	BITSTRING	8	SAPSWBTK	SJF token for non-SWA SWBs
1316	(524)	BITSTRING	6	SAPANCHR	MQTR of first regular IOT
1322	(52A)	BITSTRING	6	SAPIOTW	MQTR of IOT waiting in the "wings"
1328	(530)	BITSTRING	6	SAPIOTC	MQTR of current IOT
1334	(536)	BITSTRING	6	SAPIOTF	MQTR of first IOT for JOE
1340	(53C)	BITSTRING	6		Reserved for future use



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
1346	(542)	SIGNED	2	SAPPDDBW	Offset of PDDDB waiting in the "wings" (See routine CSPNPDDDB in HASCSAPI)
1348	(544)	SIGNED	2	SAPPDDBO	Offset of current PDDDB
1350	(546)	SIGNED	2	SAPPDDBF	Offset of first PDDDB
1352	(548)	SIGNED	2		Reserved for future use
1356	(54C)	ADDRESS	4	SAPSJB	COM.Address of SJB
1360	(550)	ADDRESS	4	SAPSDB	COM.Address of SDB
1364	(554)	ADDRESS	4	SAPMTRB	COM.Address of MTRB representing this request
1368	(558)	SIGNED	4	SAPWKOFF	Offset of work JOE into JOT
1372	(55C)	SIGNED	4	SAPWJTOF	Offset of work JOE matching the SYSOUT token

Comment

-----  
PDDDB work areas below have standard JES2 prefix (see \$CSBPRFX in \$HASPEQU).

Work area address point to a start of PDDDB in a work area.

Work areas are allocated in the private storage of the address space which initiated SAPI session and therefore is not easily accessible outside of that address space.

-----

End of Comment

1376	(560)	ADDRESS	4	SAPPDDB	Copy of currently allocated PDDDB
1380	(564)	SIGNED	4	SAPPDDBM	Max size of PDDDB which will fit in SAPPDDB
1384	(568)	ADDRESS	4	SAPPDDB2	Copy of PDDDB after the current PDDDB (if any)
1388	(56C)	SIGNED	4	SAPPDB2M	Max size of PDDDB which will fit in SAPPDDB2
1392	(570)	SIGNED	4	SAPRJOEO	Offset of \$TREGROUP JOE
1396	(574)	SIGNED	4	(4)	Reserved for future use
1412	(584)	BITSTRING	180	SAPWKJOA	Copy of WORK/CHAR JOA (never modified)
1592	(638)	BITSTRING	1	SAPJQEAR	Copy of JQE (no SPOOLs mask)
1592	(638)	X'648'	0	SAPJBKEY	"SAPJQEAR+JQEBKEY-JQE,L'JQEBKEY" Job key
1688	(698)	BITSTRING	180	SAPWWJOA	Working WORK/CHAR JOA Updated at PUT-GET TIME in the user ADDRESS SPACE
1868	(74C)	BITSTRING	180	SAP2WJOA	2nd Working WORK/CHAR JOA updated at unallocation TIME IN the user address space
2048	(800)	BITSTRING	80	SAPCTKN	Copy of client or JOE token
2128	(850)	BITSTRING	568	SAPWSP	Copy of WSP used for \$#GET
2696	(A88)	BITSTRING	232	SAPWS	EBCDIC WS list
2928	(B70)	DBL WORD	8	(0)	
2928	(B70)	SIGNED	2	SAP#SKIP	Number of PDDDBs skipped for SAF reasons
2930	(B72)	SIGNED	2	SAP#PDDDB	Number of PDDDBs processed within the current JOE. Meaningless IF SAPJCOMP is on.
2932	(B74)	SIGNED	2	SAPCLFT	Number of copies left for the last PDDDB in this grp
2934	(B76)	SIGNED	2	SAPONODE	Origin node for selection
2936	(B78)	BITSTRING	8	SAPRBA	RBA for last PDDDB in group (SAP2CHKP must be set)
2944	(B80)	ADDRESS	1	SAPTYPE	Application call type
2945	(B81)	CHARACTER	8	SAPAPPL	Application thread name
2953	(B89)	CHARACTER	8	SAPJNAME	Job name of the application
2961	(B91)	CHARACTER	8	SAPJOBID	Application jobid
2969	(B99)	CHARACTER	8	SAPCHKEY	Application CSCB CHKEY
2977	(BA1)	BITSTRING	3		Reserved for future use
2980	(BA4)	SIGNED	4	SAPWRNUM	Unique number identifying this SAPID. Used in JWEL tables. High order bit always on to differentiate from DCT addresses
2984	(BA8)	BITSTRING	1	SAPFLAG2	Miscellaneous flags
		1... ....		SAP2UNAV	"B'10000000" Data set not available
		.1.. ....		SAP2COPY	"B'01000000" User's SSS2 copied to SAPID
		..1. ....		SAP2NPRO	"B'00100000" PDDDB at offset SAPPDDBO not yet given to caller but been SAF verified ==> SAPPDDB2 validated
		...1 ....		SAP2END	"B'00010000" No more PDDDBs this JOE
		.... 1...		SAP2NEW	"B'00001000" The JOE associated with this SAPID has changed (either there is a new JOE or there is no JOE)

## \$SAPID Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... .1..		SAP2NEWS	"B'00000100" This is NEWS PDDB
		.... ..1.		SAP2CHKP	"B'00000010" SAPRBA is valid
		.... ...1		SAP2NCKU	"B'00000001" Do not update CHK
2985	(BA9)	CHARACTER	1	SAPMCLAS	Message class of job
2986	(BAA)	BITSTRING	1	SAPFLAG3	More miscellaneous flags information
		1... ....		SAP3VTOK	"B'10000000" Data set obtained via token
		.1.. ....		SAP3DNFJ	"B'01000000" Do not find new JOE
		..1. ....		SAP3GENC	"B'00100000" Low job id has a generic
		...1 ....		SAP3GEN1	"B'00010000" Low job id has generic ' ' as the first character
2987	(BAB)	SIGNED	1	SAPREAS	Reason code for SSS2EODS
2992	(BB0)	DBL WORD	8	SAPSTCK	STCK when application last made an SSI call
3000	(BB8)	CHARACTER	64	SAPCJCOR	Job correlator from caller work selection criteria
3064	(BF8)	CHARACTER	64	SAPJCOR	Job correlator of selected job
3128	(C38)	DBL WORD	8	(0)	Double word aligned
3128	(C38)	X'C38'	0	SAPLEN	**-SAPID" Length of SAPID dsect

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	TJEV	,
0	(0)	CHARACTER	4	TJEEYE	Eye catcher
4	(4)	ADDRESS	4	TJESAP	SAP.Address of corresponding SAPID
8	(8)	BITSTRING	62500	TJEJOES	Exclusion indicators
62508	(F42C)	BITSTRING	62500		Exclusion indicators
125008	(1E850)	DBL WORD	8	(0)	Double word aligned
125008	(1E850)	X'1E850'	0	TJELEN	**-TJEV" Length of TJEV

## \$SAPID Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SAP#PDDB	B72		SAPJCHHI	4B4	
SAP#SKIP	B70		SAPJCHLO	4AC	
SAPACCT	500		SAPJCOMP	5	80
SAPANCHR	524		SAPJCOR	BF8	
SAPAPPL	B81		SAPJCT	4F0	
SAPASCB	4D4		SAPJCTRL	5	8
SAPASCBT	4D8		SAPJFINI	5	10
SAPBTOK	4FC		SAPJNAME	B89	
SAPCHK	4F4		SAPJNOHI	4A8	
SAPCHKEY	B99		SAPJNOLO	4A4	
SAPCJCOR	BB8		SAPJOA	514	
SAPCKEY	8		SAPJOBID	B91	
SAPCLFT	B74		SAPJQEAR	638	
SAPCTKN	800		SAPJSAF	5	40
SAPDTKN	504		SAPJSYSH	5	4
SAPECBP	10	38	SAPJ2CKE	6	1
SAPEYE	0	5BE2C1D7	SAPJ2CKV	6	2
SAPFLAGJ	5		SAPJ2OLD	6	40
SAPFLAG1	4		SAPJ2PRI	6	4
SAPFLAG2	BA8		SAPJ2SFP	6	20
SAPFLAG3	BAA		SAPJ2SON	6	8
SAPFLGJ2	6		SAPJ2TH	6	80
SAPID	0		SAPLEN	C38	C38
SAPIOT	4E8		SAPMCLAS	BA9	
SAPIOTC	530		SAPMSTRV	7	
SAPIOTF	536		SAPMTRB	554	
SAPIOTPA	4EC		SAPNDH	50C	
SAPIOTW	52A		SAPNEXT	C	
SAPJALLO	5	20	SAPNJH	508	
SAPJASH	5	2	SAPONODE	B76	
SAPJBKEY	638	648	SAPOTCB	4E4	
SAPJBUST	5	1	SAPPDB2M	56C	

Name	Hex Offset	Hex Value
SAPPDDB	560	
SAPPDDBF	546	
SAPPDDBM	564	
SAPPDDBO	544	
SAPPDDBW	542	
SAPPDDB2	568	
SAPPRIV	4CC	
SAPRBA	B78	
SAPREAS	BAB	
SAPRETN	4C8	
SAPRJOEO	570	
SAPROUTE	498	
SAPROUTN	4BC	
SAPSDB	550	
SAPSJB	54C	
SAPSSS2	10	
SAPSTCK	BB0	
SAPSWB	510	
SAPSWBTK	51C	
SAPTCB	4E0	
SAPTJEV	518	
SAPTYPE	B80	
SAPUSER	49C	
SAPUSERN	4C0	
SAPWAVE	4F8	
SAPWJTOF	55C	
SAPWKJOA	584	
SAPWKOFF	558	
SAPWRASI	4D8	4D8
SAPWRNUM	BA4	
SAPWS	A88	
SAPWSP	850	
SAPWWJOA	698	
SAP1BJWL	4	1
SAP1GAVE	4	40
SAP1HOT	4	8
SAP1JWEL	4	2
SAP1PCE	4	4
SAP1RQUE	4	80
SAP1TERM	4	10
SAP1WSPV	4	20
SAP2CHKP	BA8	2
SAP2COPY	BA8	40
SAP2END	BA8	10
SAP2NCKU	BA8	1
SAP2NEW	BA8	8
SAP2NEWS	BA8	4
SAP2NPRO	BA8	20
SAP2UNAV	BA8	80
SAP2WJOA	74C	
SAP3DNFJ	BAA	40
SAP3GENC	BAA	20
SAP3GEN1	BAA	10
SAP3VTOK	BAA	80
TJEEYE	0	E3D1C5E5
TJEJOES	8	
TJELEN	1E850	1E850
TJESAP	4	
TJEV	0	

## \$SAPID Cross Reference

## \$SBWA Information

### \$SBWA Heading Information

**Common Name:** Hasp Spool Browse Work Area  
**Macro ID:** \$SBWA  
**DSECT Name:** SBWA  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SBWA  
 Offset: SBWAID-SBWA  
 Length: 4  
**Storage Attributes:** Subpool: 231  
 Key: 0  
 Residency: Virtual - Anywhere Real - Anywhere  
**Size:** See SBWASIZE  
**Created by:** SVCSR (HASCHAM)  
**Pointed to by:** SRBPARM field of the SRB data area  
**Serialization:** None required  
**Function:** The \$SBWA data area provides the mapping DSECT for the data areas used for passing the "unwritten buffer" from a job's address space to CSA so HASCHAM can pass records to the user.

### \$SBWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SBWA	SPOOL BROWSE WORK AREA
0	(0)	CHARACTER	4	SBWAID	\$SBWA IDENTIFIER
4	(4)	ADDRESS	4	SBWAHCCT	Address of the HCCT
8	(8)	SIGNED	4	SBWAMTTR	TRACK ADDRESS OF BUFFER
12	(C)	ADDRESS	4	SBWAFSDB	FIRST SDB OF JOB W/ BUFFER
16	(10)	DBL WORD	8	SBWAKEY (0)	UNIQUE KEY FOR BUFFER
16	(10)	SIGNED	4	SBWAJKEY	JOB KEY FOR BUFFER
20	(14)	SIGNED	4	SBWADKEY	DATA SET KEY FOR BUFFER
24	(18)	CHARACTER	4	SBWAMEMB	Owning member name
28	(1C)	ADDRESS	4	SBWAHSXB	Target HASXB address
32	(20)	BITSTRING	8	SBWASTKN	Target address space token
40	(28)	BITSTRING	1	SBWAQNUM	PBF Queue counter
42	(2A)	SIGNED	2	SBWASID	Target ASID
44	(2C)	BITSTRING	1	SBWAMEMN	Owning member number
46	(2E)	SIGNED	2	SBWAWKAL	LENGTH OF WORK AREA
48	(30)	SIGNED	2	SBWABFSZ	LENGTH OF IO BUFFER
50	(32)	BITSTRING	1	SBWAFLG1	UBSR flags
		1... ....		SBWAISDB	"B'10000000" Invalid SDB
		.1.. ....		SBWAFSD	"B'01000000" SDB found
		..1. ....		SBWAIBFD	"B'00100000" Invalid BFD
		...1 ....		SBWAFBFD	"B'00010000" BFD found
51	(33)	BITSTRING	1	SBWAFLG2	Data set information
		1... ....		SBWA2SPB	"B'10000000" Spool browse
		.1.. ....		SBWA2JLG	"B'01000000" Job log data set
52	(34)	BITSTRING	2		Reserved for alignment
56	(38)	ADDRESS	4	SBWALOC	BFDLOC for copied buffer
60	(3C)	ADDRESS	4	SBWASCDR	SDBSCDR for copied buffer

Comment

Parameter list for SJB ENQ

End of Comment

## \$SBWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
64	(40)	CHARACTER	12	SBWAMINN	SJB ENQ minor name
Comment					
MACRO-DATE = 06/24/03					
End of Comment					
76	(4C)	SIGNED	4	(0)	ESTABLISH A FULLWORD BOUNDARY
76	(4C)	ADDRESS	4		PREFIX - TCB ADDRESS X02113
80	(50)	ADDRESS	4		PREFIX - ECB ADDRESS
80	(50)	X'54'	0	SBWAENQ	*** X02113
84	(54)	ADDRESS	1		PELLAST flag byte. X02113
85	(55)	ADDRESS	1		PELMILEN - RNAME length.
86	(56)	BITSTRING	1		
Comment					
PELFLAG - flag byte 2.					
End of Comment					
87	(57)	ADDRESS	1		PELRET - return code byte.
88	(58)	ADDRESS	4		QNAME ADDRESS
92	(5C)	ADDRESS	4		RNAME ADDRESS
92	(5C)	X'C'	0	SBWAENQL	** -SBWAENQ'
96	(60)	BITSTRING	8	SBWAWORK	Work area
Comment					
ERROR EQUATE VALUES FROM UBSRB AND SVCSRB.					
When adding a new return code, there are two branch tables which need to be updated:					
1. HASCHAM: near label PRSRCBT					
2. HASCSISC: near label SIOSRBOK					
End of Comment					
104	(68)	SIGNED	4	SBWARETC	RETURN CODE FROM SRB
104	(68)	X'0'	0	SBWAOK	"0" PROCESSING SUCCESSFUL
104	(68)	X'4'	0	SBWABFNF	"4" BUFFER NOT FOUND
104	(68)	X'8'	0	SBWAINBF	"8" INVALID BUFFER
104	(68)	X'C'	0	SBWASDNF	"12" SDB NOT FOUND
104	(68)	X'10'	0	SBWAINSD	"16" INVALID SDB
104	(68)	X'14'	0	SBWASRBF	"20" SRB FAILURE
104	(68)	X'18'	0	SBWANBWA	"24" No storage for SBWA
104	(68)	X'1C'	0	SBWANES1	"28" ESTAE1 not established
104	(68)	X'20'	0	SBWASJNF	"32" SJB not found
104	(68)	X'24'	0	SBWAINHB	"36" Invalid HASB
104	(68)	X'28'	0	SBWAINSJ	"40" Invalid SJB
104	(68)	X'2C'	0	SBWAPRNF	"44" Point record not found
104	(68)	X'30'	0	SBWANDAT	"48" Dataset never written to
104	(68)	X'30'	0	SBWAMXRC	"SBWANDAT" Largest valid return code

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Return status information from IEAMSCHD					
Values returned in SBWASCCC:					
		0 Successful Completion			
		8 SRB ABENDED and there is an associated reason code.			
		12 SRB ABENDED and there is no associated reason code.			
		16 SRB Purged by PurgeDQ Processing.			
		20 Undetermined. The SRB did not complete, but was dispatched -- MEMTERM and/or DATERR probable cause.			
		24 SRB was not scheduled, Return Code is in SBWASCRC.			
		28 SRB was not scheduled, ABEND Code is in SBWASCRC.			
		32 SRB was scheduled, however the caller's workunit was ABENDED while suspended waiting for the SYNCH(YES) SRB to complete. ( ' <- needed to make PLX compile work)			
Meaning of SBWASCRC based on SBWASCCC values:					
SBWASCCC SBWASCRC value					
		0 Register 15 when SRB completed.			
		8 ABEND Code (Same format as SDWAABCC.)			
		12 ABEND Code (Same format as SDWAABCC.)			
		16 -1			
		20 -1			
		24 Return Code propagated from the SUSPEND service. The SRB was not scheduled because this workunit could not be successfully suspended.			
		28 ABEND Code propagated from the SUSPEND service. The SRB was not scheduled because this workunit could not be successfully suspended.			
		32 ABEND Code that the workunit received when it was awoken from the Suspend.			
Meaning of SBWASCRS based on SBWASCCC values:					
SBWASCCC SBWASCRS value					
		0 Register 0 when SRB completed.			
		8 Reason Code associated with an ABEND Code.			
		12 -1			
		16 -1			
		20 -1			
		24 -1			
		28 Reason Code associated with a ABEND code from the attempting to suspend the current workunit.			
		32 Reason Code associated with the ABEND Code that the workunit received when it was awoken from the Suspend.			
End of Comment					
108	(6C)	ADDRESS	4	SBWASCC@	SRB Completion code address
112	(70)	ADDRESS	4	SBWASCR@	SRB Return code address
116	(74)	ADDRESS	4	SBWASCS@	SRB Reason code address

# \$SBWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
120	(78)	SIGNED	4	SBWASCCC	SRB completion code
124	(7C)	SIGNED	4	SBWASCRC	SRB Return code
128	(80)	SIGNED	4	SBWASCRS	SRB Reason code
Comment					
Parameter list for IEAMSCHD					
End of Comment					
132	(84)	ADDRESS	4	SBWASRTN	Address of SRB routine
136	(88)	ADDRESS	4	SBWASBWA	Address of this SBWA
144	(90)	DBL WORD	8	(0)	Align parm list MACDATE -12/18/12-<4>
0	(0)	X'90'	0	M00M1325	"SBWASCHA" ++ IEAMSCHD NAME
144	(90)	DBL WORD	8	SBWASCHA (0)	++ IEAMSCHD PARM LIST
144	(90)	BITSTRING	1	SBWASCHA_XVERSION	++ INPUT XVERSION
145	(91)	BITSTRING	1	SBWASCHA_XFLAG1	++ FIELD_LABEL
		.... 1...		SBWASCHA_XENV_STOKEN	"B'00001000" ++ XENV.STOKEN KEYWORD
		.... .1..		SBWASCHA_XENV_FULLXM	"B'00000100" ++ XENV.FULLXM KEYWORD
		.... ..1.		SBWASCHA_XENV_PRIMARY	"B'00000010" ++ XENV.PRIMARY KEYWORD
		.... ...1		SBWASCHA_XENV_HOME	"B'00000001" ++ XENV.HOME KEYWORD
146	(92)	BITSTRING	1	SBWASCHA_XFLAG2	++ FIELD_LABEL
		.1.. ....		SBWASCHA_KEYUSED_SRBIDTOKEN	"B'01000000" ++ KEYUSED.SRBIDTOKEN KEYWORD
		..1. ....		SBWASCHA_KEYUSED_DUALPOOLTOKEN	"B'00100000" ++ KEYUSED.DUALPOOLTOKEN KEYWORD
		...1 ....		SBWASCHA_XSYNCH_YES	"B'00010000" ++ XSYNCH.YES KEYWORD
		.... 1...		SBWASCHA_KEYUSED_KEYVALUE	"B'00001000" ++ KEYUSED.KEYVALUE KEYWORD
		.... .1..		SBWASCHA_XLLOCK_YES	"B'00000100" ++ XLLOCK.YES KEYWORD
		.... ..1.		SBWASCHA_XFEATURE_CPMASK	"B'00000010" ++ XFEATURE.CPMASK KEYWORD
		.... ...1		SBWASCHA_XFEATURE_CRYPTO	"B'00000001" ++ XFEATURE.CRYPTO KEYWORD
147	(93)	BITSTRING	1	SBWASCHA_XFLAG3	++ FIELD_LABEL
		..1. ....		SBWASCHA_XPRIORITY_CLIENT	"B'00100000" ++ XPRIORITY.CLIENT KEYWORD
		...1 ....		SBWASCHA_XPRIORITY_ENCLAVE	"B'00010000" ++ XPRIORITY.ENCLAVE KEYWORD
		.... 1...		SBWASCHA_XPRIORITY_PREEMPT	"B'00001000" ++ XPRIORITY.PREEMPT KEYWORD
		.... .1..		SBWASCHA_XPRIORITY_CURRENT	"B'00000100" ++ XPRIORITY.CURRENT KEYWORD
		.... ..1.		SBWASCHA_XPRIORITY_GLOBAL	"B'00000010" ++ XPRIORITY.GLOBAL KEYWORD
		.... ...1		SBWASCHA_XPRIORITY_LOCAL	"B'00000001" ++ XPRIORITY.LOCAL KEYWORD
148	(94)	ADDRESS	4	SBWASCHA_XEPADDR	++
152	(98)	BITSTRING	8	SBWASCHA_XTARGETSTOKEN	++
160	(A0)	CHARACTER	8	SBWASCHA_XENCLAVETOKEN	++
168	(A8)	BITSTRING	1	SBWASCHA_XMINORPRIORITY	++



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
169	(A9)	BITSTRING	1	SBWASCHA_XKEYVALUE	
170	(AA)	BITSTRING	2	SBWASCHA_XCPUMASK	++
172	(AC)	SIGNED	4	SBWASCHA_XPARAM	++
176	(B0)	ADDRESS	4	SBWASCHA_XFRRADDR	++
180	(B4)	ADDRESS	4	SBWASCHA_XRMTRADDR	++
184	(B8)	BITSTRING	8	SBWASCHA_XPURGESTOKEN	++
192	(C0)	ADDRESS	4	SBWASCHA_XPTCBADDR	++
196	(C4)	BITSTRING	8	SBWASCHA_XCLIENTSTOKEN	++
204	(CC)	ADDRESS	4	SBWASCHA_XSYNCHCOMPADDR	++
208	(D0)	ADDRESS	4	SBWASCHA_XSYNCHCODEADDR	++
212	(D4)	ADDRESS	4	SBWASCHA_XSYNCHRSNADDR	++
216	(D8)	CHARACTER	16	SBWASCHA_XDUALPOOLTOKEN	++
216	(D8)	X'E8'	0	SBWASCHA_PL_END	*** ++ END OF BASE PLIST
176	(B0)	CHARACTER	3	SBWASCHA_XRSV0001	++ RESERVED
179	(B3)	BITSTRING	1	SBWASCHA_XFRRFLAG	++ FIELD_LABEL
		.... ...1		SBWASCHA_XSDWALOC31_YES	"B'00000001" ++ XSDWALOC31.YES KEYWORD
232	(E8)	X'58'	0	SBWASCHAL	** -SBWASCHA" ++ LENGTH OF PLIST

Comment

IEAMSCHD-4  
Footprint area for HASCUBSR

End of Comment

232	(E8)	BITSTRING	0	SBWASROA (0)	Output area from the SRB for debugging. Keep fields SBWASDBA through SBWABATA together.
232	(E8)	ADDRESS	4	SBWASDBA	---+ A(SDB) that matches the key
236	(EC)	ADDRESS	4	SBWAPBF	SDBPBF
240	(F0)	ADDRESS	4	SBWAPBFI	Inflight PBF buffer address
244	(F4)	ADDRESS	4	SBWABPTR	Address of found buffer
248	(F8)	SIGNED	4	SBWABTRK	MTTR of found buffer
252	(FC)	ADDRESS	4	SBWABDBF	Address of invalid buffer
256	(100)	ADDRESS	4	SBWABDSD	Address of invalid SDB
260	(104)	ADDRESS	4	SBWABATF	BATPBF if checking this Q
264	(108)	ADDRESS	4	SBWABATA	---+ BATPBFA if checking this Q
264	(108)	X'24'	0	SBWAOLEN	** -SBWASDBA" Length of output area
268	(10C)	SIGNED	4	SBWARSV1	RESERVED
272	(110)	SIGNED	4	SBWARSV2	RESERVED
276	(114)	SIGNED	4	SBWARSV3	RESERVED
280	(118)	BITSTRING	1	SBWAEND (0)	
280	(118)	X'118'	0	SBWASIZE	"SBWAEND-SBWA" SIZE OF \$SBWA DATA AREA
280	(118)	X'118'	0	SBWABFFR	*** LOCATION OF BUFFER FOR MOVE

## \$SBWA Cross Reference

### \$SBWA Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
M00M1325	0	90	SBWASCHA_XCLIENTSTOKEN		
SBWA	0		C4		
SBWABATA	108		SBWASCHA_XCPUMASK		
SBWABATF	104		AA		
SBWABDBF	FC		SBWASCHA_XDUALPOOLTOKEN		
SBWABDSD	100		D8		
SBWABFFR	118	118	SBWASCHA_XENCLAVETOKEN		
SBWABFNF	68	4	A0		
SBWABFSZ	30		SBWASCHA_XENV_FULLXM		
SBWABPTR	F4		91	4	
SBWABTRK	F8		SBWASCHA_XENV_HOME		
SBWADKEY	14		91	1	
SBWAEND	118		SBWASCHA_XENV_PRIMARY		
SBWAENQ	50	54	91	2	
SBWAENQL	5C	C	SBWASCHA_XENV_STOKEN		
SBWAFBFD	32	10	91	8	
SBWAFLG1	32		SBWASCHA_XEPADDR		
SBWAFLG2	33		94		
SBWAFSD	32	40	SBWASCHA_XFEATURE_CPMASK		
SBWAFSDB	C		92	2	
SBWAHCCT	4		SBWASCHA_XFEATURE_CRYPTO		
SBWAHSXB	1C		92	1	
SBWAIBFD	32	20	SBWASCHA_XFLAG1		
SBWAID	0		91		
SBWAINBF	68	8	SBWASCHA_XFLAG2		
SBWAINHB	68	24	92		
SBWAINSD	68	10	SBWASCHA_XFLAG3		
SBWAINSJ	68	28	93		
SBWAISDB	32	80	SBWASCHA_XFRRADDR		
SBWAJKEY	10		B0		
SBWAKEY	10		SBWASCHA_XFRRFLAG		
SBWALOC	38		B3		
SBWAMEMB	18		SBWASCHA_XKEYVALUE		
SBWAMEMN	2C		A9		
SBWAMINN	40	E2D1C24B	SBWASCHA_XLLOCK_YES		
SBWAMTTR	8		92	4	
SBWAMXRC	68	30	SBWASCHA_XMINORPRIORITY		
SBWANBWA	68	18	A8		
SBWANDAT	68	30	SBWASCHA_XPARAM		
SBWANES1	68	1C	AC		
SBWAOK	68	0	SBWASCHA_XPRIORITY_CLIENT		
SBWAOLEN	108	24	93	20	
SBWAPBF	EC		SBWASCHA_XPRIORITY_CURRENT		
SBWAPBFI	F0		93	4	
SBWAPRNF	68	2C	SBWASCHA_XPRIORITY_ENCLAVE		
SBWAQNUM	28		93	10	
SBWARETC	68		SBWASCHA_XPRIORITY_GLOBAL		
SBWARSV1	10C		93	2	
SBWARSV2	110		SBWASCHA_XPRIORITY_LOCAL		
SBWARSV3	114		93	1	
SBWASBWA	88		SBWASCHA_XPRIORITY_PREEMPT		
SBWASCC@	6C		93	8	
SBWASCCC	78		SBWASCHA_XPTCBADDR		
SBWASCDR	3C		C0		
SBWASCHA	90		SBWASCHA_XPURGESTOKEN		
SBWASCHA_KEYUSED_DUALPOOLTOKEN			B8		
	92	20	SBWASCHA_XRMTRADDR		
SBWASCHA_KEYUSED_KEYVALUE			B4		
	92	8	SBWASCHA_XRSV0001		
SBWASCHA_KEYUSED_SRBIDTOKEN			B0		
	92	40	SBWASCHA_XSDWALOC31_YES		
SBWASCHA_PL_END			B3	1	
	D8	E8	SBWASCHA_XSYNCH_YES		

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
	92	10
SBWASCHA_XSYNCHCODEADDR	D0	
SBWASCHA_XSYNCHCOMPADDR	CC	
SBWASCHA_XSYNCHRSNADDR	D4	
SBWASCHA_XTARGETSTOKEN	98	
SBWASCHA_XVERSION	90	
SBWASCHAL	E8	58
SBWASCR@	70	
SBWASCRC	7C	
SBWASCRS	80	
SBWASCS@	74	
SBWASDBA	E8	
SBWASDNF	68	C
SBWASID	2A	
SBWASIZE	118	118
SBWASJNF	68	20
SBWASRBF	68	14
SBWASROA	E8	
SBWASRTN	84	
SBWASTKN	20	
SBWAWKAL	2E	
SBWAWORK	60	
SBWA2JLG	33	40
SBWA2SPB	33	80

## \$SBWA Cross Reference

---

## **\$SCAND Information**

### **\$SCAND Programming Interface information**

Programming Interface information

**\$SCAND**

End of Programming Interface information

## Heading Information • \$SCAND Cross Reference

### \$SCAND Heading Information

**Common Name:** \$SCAND parameter list  
**Macro ID:** \$SCAND  
**DSECT Name:** SCDW  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: any  
 Key: 0 or 1  
 Residency: Virtual and real storage are below 2G, in private storage.  
**Size:** See SCDWLEN  
**Created by:** \$SCAND list form  
**Pointed to by:** Register 15 on entry to the \$SCAND service  
**Serialization:** N/A  
**Function:** Maps the parameters specified for the \$SCAND service

### \$SCAND Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCDW	Map the \$SCAND DSECT
0	(0)	ADDRESS	4	SCDWFLD	Address of field to display
4	(4)	ADDRESS	4	SCDWFLDA	ALET associated with field
8	(8)	SIGNED	2	SCDWFLDL	Field length
10	(A)	SIGNED	2	SCDWWDTH	Width
12	(C)	BITSTRING	1	SCDWOPT1	\$SCAND options 1
		1... ....		SCDW1DBK	"B'10000000" BREAK OPTION REQUESTED
		.1.. ....		SCDW1DBL	"B'01000000" DEBLANKING OPTION REQUESTED
		..1. ....		SCDW1DMK	"B'00100000" MARK TEXT FOR BACKOUT
		...1 ....		SCDW1DCR	"B'00010000" CRLF was requested
		.... 1...		SCDW1NBN	"B'00001000" Disallow break on next display at this level
		.... .1..		SCDW1NIN	"B'00000100" Do not indent this line
13	(D)	BITSTRING	1	SCDWOPT2	\$SCAND options 2
		1... ....		SCDW2LTC	"B'10000000" Control line->WPLLTFB
		.1.. ....		SCDW2LTL	"B'01000000" Label line--->WPLLTFB
		..1. ....		SCDW2LTD	"B'00100000" Data line---->WPLLTFB
		...1 ....		SCDW2LTE	"B'00010000" End line----->WPLLTFB
		.... 1...		SCDW2BKS	"B'00001000" BRKOPT=STAB specified
14	(E)	BITSTRING	1	SCDWCNV1	Conversion flags 1 (Maps to STABCNV1)
15	(F)	BITSTRING	1	SCDWCNV2	Conversion flags 2 (Maps to STABCNV2)
16	(10)	BITSTRING	1	SCDWCNV3	Conversion flags 3 (Maps to STABCNV3)
17	(11)	BITSTRING	1	SCDWCNV4	Conversion flags 4 (Maps to STABCNV4)
18	(12)	SIGNED	4	SCDWMULT	Multiplier
18	(12)	X'16'	0	SCDWLEN	**"-SCDW" Length of parameter list

### \$SCAND Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SCDW	0		SCDW1DBK	C	80
SCDWCNV1	E		SCDW1DBL	C	40
SCDWCNV2	F		SCDW1DCR	C	10
SCDWCNV3	10		SCDW1DMK	C	20
SCDWCNV4	11		SCDW1NBN	C	8
SCDWFLD	0		SCDW1NIN	C	4
SCDWFLDA	4		SCDW2BKS	D	8
SCDWFLDL	8		SCDW2LTC	D	80
SCDWLEN	12	16	SCDW2LTD	D	20
SCDWMULT	12		SCDW2LTE	D	10
SCDWOPT1	C		SCDW2LTL	D	40
SCDWOPT2	D				
SCDWWDTH	A				

---

## **\$SCANWA Information**

### **\$SCANWA Programming Interface information**

Programming Interface information

**\$SCANWA**

End of Programming Interface information

## Heading Information

### \$SCANWA Heading Information

**Common Name:** \$SCAN Facility Work Area  
**Macro ID:** \$SCANWA  
**DSECT Name:** SCWA, SCWABA, SCWADA, XWCWA  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** "SCWA" or "TEMP"  
Offset: SCWAID-SCWA  
Length: L'SCWAID

**Storage Attributes:** Subpool: 1  
Key: 1  
Residency: Virtual and real storage are above 16M, in the private storage of the JES2 address space.

**Size:** See SCWALEN, SCWALEND

**Created by:** \$SCAN macro expansion (normal SCWAs)  
\$SCANB service (backup SCWAs)  
\$SCAND service (display SCWAs)  
Internal \$SCAN processing (filter and subscript SCWAs)

**Pointed to by:** R1 on entry to prescan and postscan exits  
SCWAPWA field of the \$SCANWA data area  
SCWADNWA field of the \$SCANWA data area  
SCWADPWA field of the \$SCANWA data area  
SCWABNWA field of the \$SCANWA data area  
SCWABPWA field of the \$SCANWA data area  
SCWAFNWA field of the \$SCANWA data area  
SCWASNWA field of the \$SCANWA data area  
SCWAOLDP field of the \$SCANWA data area  
SCWAWCWA field of the \$SCANWA data area

**Serialization:** None required.



**Function:**

The SCWA is used as a general work area for \$SCAN.

There are several types of SCWAs:

1) Normal SCWAs - these contain general information regarding the parsing of a string (for example, pointers and lengths of text within the string, subscript and control block information, etc.)

One normal SCWA exists for each recursive level of \$SCAN used in parsing a particular string.

2) Display SCWAs - these are chained to the "oldest" normal SCWA and contain text to be displayed on a \$SCAN display request, specified by the \$SCAND macro. This text is represented by smaller units within the display SCWA (SCWADAs), which contain additional information, such as whether it is allowed (or required) to begin a new display line.

3) Backup SCWAs - contain original values of fields modified by \$SCAN, and are used to restore the original value in case of an error. The backup SCWA is broken up into smaller units (SCWABAs), which contain additional information, such as the length, address, and original value of the backed-up field.

4) Filter SCWAs - keep track of which keywords are specified as filters in the \$SCAN input string.

5) Subscript SCWAs - keep track of additional subscripts or ranges of subscripts in the \$SCAN input string.

**\$SCANWA Map**

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCWA	INTERNAL SCAN WORK AREA DSECT
0	(0)	CHARACTER	4	SCWAID	EBCDIC CONTROL BLOCK ID, SET BY \$GETWORK VIA USE=SCWA
4	(4)	ADDRESS	4	SCWADPWA	ADDR OF PREVIOUS DISPLAY SCWA
8	(8)	ADDRESS	4	SCWABPWA	ADDR OF PREVIOUS BACKUP SCWA
12	(C)	ADDRESS	4	SCWADNWA	ADDR OF NEXT DISPLAY SCWA
16	(10)	ADDRESS	4	SCWABNWA	ADDR OF NEXT BACKUP SCWA
20	(14)	ADDRESS	4	SCWAFNWA	ADDR OF NEXT FILTER SCWA
24	(18)	ADDRESS	4	SCWASNWA	ADDR OF NEXT SUBSCRIPT SCWA
28	(1C)	ADDRESS	4	SCWAENWA	ADDR of next TYPE=ERROR BACKUP SCWA
32	(20)	BITSTRING	1	SCWAKIND	\$SCANWA WORK AREA KIND
		1... ....		SCWAKNOR	"B'10000000" NORMAL SCWA
		.1... ....		SCWAKDSP	"B'01000000" DISPLAY SCWA
		..1. ....		SCWAKBAK	"B'00100000" BACKUP SCWA
		...1 ....		SCWAKFLT	"B'00010000" FILTER SCWA
		.... 1...		SCWAKSUB	"B'00001000" SUBSCRIPT SCWA
33	(21)	BITSTRING	1	SCWAFLG6	GENERAL FLAG BYTE 6
		1... ....		SCWA6GEN	"B'10000000" FIRST GENERIC ENTRY SAVED
		.1... ....		SCWA6BNO	"B'01000000" BRKNEXT=NO specified for last \$SCAND call
		..1. ....		SCWA6NCR	"B'00100000" Creates disallowed due to generic subscript
		...1 ....		SCWA6MSS	"B'00010000" Multiple subscripts
		.... 1...		SCWA60SS	"B'00001000" No subscripts specified, "" assumed
		.... .1..		SCWA6GT	"B'00000100" Filter should match if >
		.... .1.		SCWA6LT	"B'00000010" Filter should match if <
		.... ...1		SCWA6EQ	"B'00000001" Filter should match if =
33	(21)	X'7'	0	SCWA6NOT	"SCWA6GT+SCWA6EQ+SCWA6LT" Composite for ~ (NOT)
34	(22)	BITSTRING	1	SCWASLVL	This SCWA scan call level (starting at 0 for the oldest parent)

# \$SCANWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
35	(23)	BITSTRING	1	SCWAELVL	Scan level of error (to be propagated to oldest parent)
36	(24)	ADDRESS	4	SCWARLWA	Related SCWA (eg. set SCWA for display request)
40	(28)	DBL WORD	8	SCWADWRK	Doubleword work area
48	(30)	DBL WORD	8	SCWADWK1	Doubleword work area
56	(38)	DBL WORD	8	SCWADWK2	Doubleword work area
64	(40)	DBL WORD	8	SCWADWK3	Doubleword work area
64	(40)	X'28'	0	SCWAWK16	"SCWADWRK,16,C'X'" 16-byte work area
64	(40)	X'28'	0	SCWAWK20	"SCWADWRK,20,C'X'" 20-byte work area
64	(40)	X'28'	0	SCWAWK24	"SCWADWRK,24,C'X'" 24-byte work area
64	(40)	X'28'	0	SCWAWK32	"SCWADWRK,32,C'X'" 32-byte work area
64	(40)	X'38'	0	SCWAW16B	"SCWADWK2,16,C'X'" 16-byte work area 2
72	(48)	CHARACTER	16	SCWAXWA	SCAN work area
88	(58)	ADDRESS	4	SCWAWCWA	Address of ASAXWC parm list
92	(5C)	ADDRESS	4	SCWAORG1 (0)	ORG POINT FOR DISPLAY AND BACKUP SCWA'S
92	(5C)	ADDRESS	4	SCWACR11	\$SCAN caller's R11 value
96	(60)	ADDRESS	4	SCWATOKN	ADDR OF TOKEN
100	(64)	ADDRESS	4	SCWASTBS	ADDR OF \$SCAN TABLES DOUBLEWORD
104	(68)	ADDRESS	4	SCWASTMT	ADDR OF PARM STMT TO SCAN
108	(6C)	SIGNED	2	SCWASLEN	LEN OF PARM STMT TO SCAN
110	(6E)	SIGNED	2	SCWADLEN	LEN OF DISPLAY OUTPUT AREA
112	(70)	ADDRESS	4	SCWADOUT	ADDR OF DISPLAY OUTPUT AREA
116	(74)	ADDRESS	4	SCWADR TN	ADDR OF DISPLAY OUTPUT ROUTINE
120	(78)	ADDRESS	4	SCWAPWA	ADDR OF PARENT SCWA (0 IN THE OLDEST PARENT SCWA)
124	(7C)	ADDRESS	4	SCWANWA	ADDR OF DAUGHTER SCWA (0 in the youngest SCWA)
128	(80)	ADDRESS	4	SCWASTAB	ADDR OF CURRENT SCAN TABLE ENTRY
132	(84)	ADDRESS	4	SCWAOTAB	ADDR OF ORIGINAL SCAN TABLE PRIOR TO ALIAS RESOLUTION
136	(88)	ADDRESS	4	SCWACBCL	ADDR OF CONTROL BLOCK PROVIDED BY CALLER
140	(8C)	ADDRESS	4	SCWACBAD	ADDR OF CURRENT CONTROL BLOCK
144	(90)	SIGNED	4	SCWACBAL	and current CB ALET
148	(94)	ADDRESS	4	SCWAFAD	ADDR OF CURRENT FIELD
152	(98)	SIGNED	4	SCWAFAL	and current field ALET
156	(9C)	ADDRESS	4	SCWATEMP	ADDR OF TEMPORARY AREA STACK
160	(A0)	SIGNED	4	SCWADAD2	Work storage for \$GETABLE
164	(A4)	ADDRESS	4	SCWAWORK (0)	WORK AREA, USED ONLY BY HIGH LEVEL \$SCAN SUBRTNS/EXITS, E.G. A(STAB) IN FINDTAB, DCTNAME IN FINDCB
174	(AE)	SIGNED	2	SCWARTCD	RETURN CODE OF PROCESSED REQUEST
176	(B0)	ADDRESS	4	SCWAKPTR	PTR TO CURRENT KEYWORD IN STMT
180	(B4)	ADDRESS	4	SCWARPTR	PTR TO REMAINING TEXT IN STMT
184	(B8)	SIGNED	2	SCWARLEN	LEN OF REMAINING TEXT IN STMT
186	(BA)	SIGNED	2	SCWAILEN	LEN OF CURRENT INPUT STRING
188	(BC)	ADDRESS	4	SCWAIPTR	PTR TO CURRENT INPUT STRING
192	(C0)	SIGNED	4	SCWACNTR	COUNTER FIELD AVAILABLE FOR PRE AND POST-SCAN EXIT USE ONLY
196	(C4)	BITSTRING	1	SCWAEXFL	FLAG BYTE AVAILABLE FOR PRE AND POST-SCAN EXIT USE ONLY
		1... ....		SCWAJNET	"B'10000000" ON JES2 NETACCT CHAIN SEARCH
		.1.. ....		SCWARM TA	"B'01000000" RMT CURRENTLY AUTOLOG MODE
		..1. ....		SCWARTRY	"B'00100000" RETRY INDICATOR
		...1 ....		SCWARMTL	"B'00010000" \$T/\$ADD RMT spec LINE
		.... 1...		SCWARM SH	"B'00001000" \$T/\$ADD RMT spec SHARABLE
		.... .1..		SCWACPCT	"B'00000100" \$T RPR/RPU specified CMPCT
		.... ..1.		SCWACMPR	"B'00000010" \$T RPR/RPU specif COMPRESS
		1... ....		SCWA\$IND	"B'10000000" \$T MEMBER,IND=YES/NO

Comment

Definitions used by \$TJ command

End of Comment

1... ....	SCWA\$TJP	"B'10000000" \$TJ PRIORITY specified
.1.. ....	SCWA\$TJC	"B'01000000" \$TJ CLASS specified

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		..1. ....		SCWA\$TJX	"B'00100000" \$TJ XEQ specified
		...1 ....		SCWA\$TJS	"B'00010000" \$TJ SRVCLASS specified
		.... 1...		SCWA\$TJE	"B'00001000" \$TJ SCHENV specified
		.... .1..		SCWA\$TJI	"B'00000100" \$TJ SPIN specified
		.... ..1.		SCWA\$TJD	"B'00000010" \$TJ SPIN,DDNAME= specified

Comment

Definitions used by \$TOJ command

End of Comment

1... ....		SCWA\$TOX		"B'10000000" Skip to next JQE
-----------	--	-----------	--	-------------------------------

Comment

The following definitions are used by \$C and \$P job commands and must match parameters passed in R0 to the \$JCAN macro

End of Comment

...1 ....		SCWA\$JPR		"B'00010000" 'PROTECTED' specified
.1.. ....		SCWA\$CDU		"B'01000000" 'DUMP' specified

Comment

EQU B'00100000' Internal \$JCAN use

End of Comment

.... 1...		SCWA\$FRC		"B'00001000" 'FORCE' was specified
.... .1..		SCWA\$ARM		"B'00000100" 'ARMRESTART' specified
.... ...1		SCWA\$CPU		"B'00000001" Cancel with purge

Comment

Definitions used by \$EJ command

End of Comment

.... .1..		SCWA\$EST		"B'00000100" \$EJOB,STEP requested
.... .1.		SCWA\$ESH		"B'00000010" \$EJOB,STEP,HOLD requested
.... ...1		SCWA\$ECA		"B'00000001" \$EJOB,CANCEL requested

Comment

The following are used for \$T/\$D JOBCLASS(x)

End of Comment

1... ....		SCWASTMD		"B'10000000" JOBCLASS MODE= changed
.1.. ....		SCWACATL		"B'01000000" Looping through CATs

Comment

The following is used for \$T NODE

End of Comment

1... ....		SCWA\$NTC		"B'10000000" Checkpointed attribute changed
-----------	--	-----------	--	---

Comment

The following is used for \$T NJEDEF

End of Comment

1... ....		SCWA\$NNM		"B'10000000" Checkpointed attribute changed
-----------	--	-----------	--	---

# \$SCANWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
The following is used for \$\$J JOBCORR					
End of Comment					
		1... ..		SCWA\$COR	"B'10000000" Job correlator filter provided
Comment					
The following is used for SPL cmds with the RESERVED keyword.					
End of Comment					
		1... ..		SCWA\$RSV	"B'10000000" Prescan was called for =RESERVED keyword
197	(C5)	ADDRESS	1	SCWAWARN	\$\$SCAN WARNING MASK
198	(C6)	BITSTRING	1	SCWAFLG7	Flag byte 7
		1... ..		SCWA7BOU	"B'10000000" Sets to back out at this level of scan exist
		.1. ....		SCWA7DNF	"B'01000000" A conflict exists between set and filter keywords
		.1. ....		SCWA7FLF	"B'00100000" Current keyword MUST be processed as a filter
		...1 ....		SCWA7DDN	"B'00010000" The maximum number of display messages has been exceeded
		.... 1..		SCWA7FXT	"B'00001000" Filter SCWA - additional processing required
		.... .1..		SCWA7DAL	"B'00000100" Display all was requested for this keyword
		.... .1.		SCWA7DSP	"B'00000010" Something displayed at this level on this iteration
		.... ...1		SCWA7SDS	"B'00000001" Something done at subscript SCWA level (similar to SCWA4SDL but reset for new subscript SCWA)
199	(C7)	ADDRESS	1	SCWACALD	CALLER FOR DISPLAY ON \$\$SCAN CALLS THAT ARE SETDISP, SETCRDISP, ETC, DURING DISP = ORIG SET CALLER
200	(C8)	SIGNED	4	SCWASUBS	LOWER BOUNDARY AND/OR SBSCPT
204	(CC)	SIGNED	4	SCWASUBH	UPPER BOUNDARY OF SUBSCRIPT (MAY BE A LOWER VALUE THAN SSCR, IMPLYING A DECREMENTING LOOP)
208	(D0)	SIGNED	2	SCWASCRL	LENGTH OF SUBSCRIPT AREA
210	(D2)	SIGNED	2	SCWAVCNT	COUNT OF VECTOR ELMTS PROCESSED FOR ENTIRE VECTOR SUBSCAN
212	(D4)	SIGNED	2	SCWASTVC	COUNT OF VECTOR ELMTS PROCESSED WITHIN CURRENT SCANTAB ENTRY
214	(D6)	SIGNED	2	SCWASBL	FIELD LENGTH FOR \$\$SCANB
216	(D8)	BITSTRING	1	SCWAFLG9	Still more flags
		1... ..		SCWA9NFT	"B'10000000" SCWA for nested filter
217	(D9)	BITSTRING	1		Reserved
218	(DA)	SIGNED	2	SCWAINDL	INDENT VALUE FOR DISPARE RTN
220	(DC)	ADDRESS	4	SCWADADD	ADDR OF CURRENT TABLE ENTRY FOR A DISPLAY ALL ENTRIES REQUEST
224	(E0)	BITSTRING	1	SCWATYPE	\$\$SCAN CALL TYPE
		1... ..		SCWASET	"B'10000000" \$\$SCAN SCAN=SET
		.1. ....		SCWADISP	"B'01000000" \$\$SCAN SCAN=DISPLAY
		.1. ....		SCWADSPA	"B'00100000" FLAG FOR DISPLAY-AFTER
		.1.1 ....		SCWAMSG	"B'01010000" \$\$SCAN SCAN=MSG
		.... 1..		SCWACR	"B'00001000" \$\$SCAN SCAN=CR
		.... .1..		SCWADELE	"B'00000100" \$\$SCAN SCAN=DELETE
224	(E0)	X'88'	0	SCWASETC	"SCWASET+SCWACR" \$\$SCAN SCAN=SETCR
224	(E0)	X'A0'	0	SCWASETD	"SCWASET+SCWADSPA" \$\$SCAN SCAN=SETDISP
224	(E0)	X'A8'	0	SCWASCD	"SCWASETC+SCWADSPA" \$\$SCAN SCAN=SETCRDISP
224	(E0)	X'28'	0	SCWACRDI	"SCWACR+SCWADSPA" \$\$SCAN SCAN=CRDISP
224	(E0)	X'44'	0	SCWADDEL	"SCWADISP+SCWADELE" \$\$SCAN SCAN=DISPDEL
224	(E0)	X'C4'	0	SCWACRRT	"SCWASET+SCWADISP+SCWADELE" Flags to indicate (all off) CR(new CB) required
225	(E1)	BITSTRING	1	SCWAFLG1	GENERAL FLAG BYTE
		1... ..		SCWAPAR	"B'10000000" SCAN STARTED WITH A PARENTHESIS (MUST END WITH ONE)
		.1. ....		SCWASING	"B'01000000" SCAN RESTRICTED TO SINGLE KEYWORD (POSSIBLY NEEDING MULTI-SUBSCAN)

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		..1. ....		SCWASSER	"B'00100000" POSSIBLE SUBSCRIPT ERROR
		...1 ....		SCWAVECT	"B'00010000" VECTOR SCAN BEING PROCESSED
		.... 1...		SCWAPSCN	"B'00001000" PRESCAN EXIT DID SCANNING
		.... .1..		SCWARANG	"B'00000100" POSSIBLE SUBSCRIPT RANGE FOUND
		.... ..1.		SCWASCAN	"B'00000010" SUBSCAN IS REQUIRED
		.... ...1		SCWADCOM	"B'00000001" COMMA REQUIRED WITHIN A DISPLAY
226	(E2)	BITSTRING	1	SCWAFLG2	GENERAL FLAG BYTE 2
		1... ....		SCWADALL	"B'10000000" DISPLAY ALL SUBPARAMETERS
		.1.. ....		SCWAHASP	"B'01000000" HAVE COMPLETED HASP TABLE
		..1. ....		SCWA2LNG	"B'00100000" Do LONG display all
		...1 ....		SCWASPAN	"B'00010000" TEXT SPANS AN SCWA
		.... 1...		SCWALoop	"B'00001000" DISPLAY LOOP AS GENERATED FROM A PRE OR POST-SCAN EXIT AND ONLY AVAILABLE FOR THEIR USE
		.... .1..		SCWA2QSS	"B'00000100" Quotes around subscript
		.... ..1.		SCWAPAR2	"B'00000010" Copy of SCWAPAR for loops
		.... ...1		SCWAPERD	"B'00000001" INDICATE MULTI-SECTIONED KYWRD
227	(E3)	BITSTRING	1	SCWAFLG3	GENERAL FLAG BYTE 3
		1... ....		SCWAMLVL	"B'10000000" DISPLAY MORE THAN ONE SUB-KEYWORD SECTION
		.1.. ....		SCWAPARN	"B'01000000" SCWA CONTAINS PART(S) OF THE HIGHEST LEVEL KYWRD SPECIFIED
		..1. ....		SCWAERR	"B'00100000" SCANDIAG BUILDING DIAGNSTC MSG
		...1 ....		SCWAGRPD	"B'00010000" INDICATES SOMETHING DISPLAYED
		.... 1...		SCWAD1ST	"B'00001000" FIRST CALL TO DISPRTN
		.... ..1.		SCWADLST	"B'00000100" LAST CALL TO DISPRTN
		.... .1..		SCWA3TCB	"B'00000010" SCWACBAD POINTS TO TEMP CB
		.... ...1		SCWA3DCT	"B'00000001" SCWACBAD POINTS TO A DCT
228	(E4)	BITSTRING	1	SCWAFLG4	GENERAL FLAG BYTE 4
		1... ....		SCWA4SSG	"B'10000000" GENERIC SYMBOLIC SUBSCRIPT
		.1.. ....		SCWA4PSS	"B'01000000" PARENS AROUND SUBSCRIPT
		..1. ....		SCWA4ACT	"B'00100000" ACTIVITY DETERMINED THIS LEVEL
		...1 ....		SCWA4SDL	"B'00010000" SOMETHING DONE IN POSSIBLE LOOP, USED TO REPORT IF NO MATCHES
		.... 1...		SCWA4LFC	"B'00001000" LKUPFLD HAS BEEN CHANGED BY SET, DISPLAY MUST USE NEW SUBSCRIP
		.... .1..		SCWA4FLM	"B'00000100" Filter match found
		.... ..1.		SCWA4ETL	"B'00000010" END OF SCAN FOR THIS LEVEL
		.... ...1		SCWA4RDE	"B'00000001" ERROR FLAG FOR RESTDISP
229	(E5)	BITSTRING	1	SCWAFLG5	GENERAL FLAG BYTE 5
		1... ....		SCWA5FLT	"B'10000000" FILTER REQUEST DETECTED
		.1.. ....		SCWA5FRJ	"B'01000000" FILTER REQUEST REJECTED
		..1. ....		SCWA5DSP	"B'00100000" Something done at this level other than filters
		...1 ....		SCWA5PS2	"B'00010000" SECOND 'DISPLAY ALL' PASS IN PROGRESS FOR KEYWORD
		.... 1...		SCWA5FND	"B'00001000" FILTER DETECTED WHICH ALSO REQUIRES A DISPLAY
		.... .1..		SCWA5XPR	"B'00000100" DO NOT TAKE ANY MORE PRESCAN EXIT ROUTINES FOR THIS KEYWORD ITERATION
		.... ..1.		SCWA5XPO	"B'00000010" DO NOT TAKE ANY MORE POSTSCAN EXIT ROUTINES FOR THIS KEYWORD ITERATION
		.... ...1		SCWA5NSS	"B'00000001" Input at this level contained a numeric subscript (if symbolic then SCWASSSL is set)
230	(E6)	SIGNED	2	SCWARPMM	MAXIMUM RPTR MOVED IN LOOP
232	(E8)	CHARACTER	1	SCWASEPR	SEPARATOR CHARACTER USED DURING DISPLAY CREATION
233	(E9)	BITSTRING	1	SCWADSPR	\$SCAN DISPLAYER ID
234	(EA)	ADDRESS	1	SCWAKWDL	LENGTH OF FIRST SECTION OF A MULTI- SECTIONED KEYWORD
235	(EB)	ADDRESS	1	SCWACALR	\$SCAN CALLER ID - HASP IDS ARE DEFINED IN \$SHASPEQU, USERS SHOULD USE IDS FROM 255 DOWN (IF NEEDED)
236	(EC)	BITSTRING	1	SCWABFLG	FLAG BYTE USED BY \$SCANB MACRO TO PASS TYPE TO \$SCANB ROUTINE - ALL BIT DEFINITIONS IN SCWABAFG

## \$SCANWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
237	(ED)	SIGNED	1	SCWASSSL	LENGTH OF INPUT SYMBOLIC SS
238	(EE)	SIGNED	1	SCWASSL2	LENGTH of second symbolic in range
239	(EF)	BITSTRING	1	SCWAFLG8	Even more flags
		1... ....		SCWA8LTC	"B'10000000" Control line->WPLLTFB
		.1.. ....		SCWA8LTL	"B'01000000" Label line---->WPLLTFB
		..1. ....		SCWA8LTD	"B'00100000" Data line---->WPLLTFB
		...1 ....		SCWA8LTE	"B'00010000" End line----->WPLLTFD
		.... 1...		SCWA8DOU	"B'00001000" Display area obtained by \$SCAN
		.... .1..		SCWA8DCC	"B'00000100" Display routine R11=HCCT
		.... ..1.		SCWA8DHC	"B'00000010" Display routine R11=HCT
		.... ...1		SCWA8HIC	"B'00000001" Hi delimiter flag
240	(F0)	BITSTRING	2	SCWAFVCT	Counter for vector filters specifying NOVORDER
242	(F2)	BITSTRING	2	SCWAMIDL	Message id length for current message
244	(F4)	ADDRESS	4	SCWASSIE	Indirection entry save area If SUBFLD or LKUPFLD, it is current index value If SUBSCRIP (direct index) it is looping CBIND adr
248	(F8)	CHARACTER	16	SCWASSS (0)	Symbolic subscript values in normal and subscript SCWAs
248	(F8)	CHARACTER	8	SCWASSSC	SYMBOLIC SUBSCRIPT VALUE
256	(100)	CHARACTER	8	SCWASSSH	SYMBOLIC SUBSCRIPT VALUE (HIGH RANGE VALUE)
256	(100)	X'F8'	0	SCWAFW16	"SCWASSS" 16-byte work area - only in filter SCWAs
256	(100)	X'F8'	0	SCWAFW8	"SCWAFW16,8" 8-byte work area
256	(100)	X'100'	0	SCWAFW8A	"SCWAFW16+8,8" 8-byte work area
264	(108)	ADDRESS	4	SCWASSSS	SYM SUBSCRIPT CB SAVE AREA
268	(10C)	SIGNED	4	SCWASSSA	and ALET
272	(110)	ADDRESS	4	SCWASSDR	HIGHEST LEVEL SYMBOLIC LKUPFLD ADDR (USED FOR LATER DISPLAY IF SCWA4LFC IS TURNED ON)
276	(114)	BITSTRING	1	SCWAPRRRC	Highest RC encountered from prescan routine
277	(115)	BITSTRING	1	SCWAPCNT	Count of nested parens for CONV=CHAR
278	(116)	ADDRESS	1	SCWAMSDL	MAXIMUM SUBSCRIPT DISPLAY LENGTH
279	(117)	ADDRESS	1	SCWANBLN	ACTUAL SUBSCRIPT LENGTH
280	(118)	ADDRESS	4	SCWAVERB	Addr of verb (1 byte length followed by char verb)
284	(11C)	ADDRESS	4		Reserved for future use
288	(120)	ADDRESS	4	SCWASCND (0)	Start of list form
288	(120)	ADDRESS	4		Text address unknown
292	(124)	ADDRESS	4		ALET of field
296	(128)	ADDRESS	2		Text length
298	(12A)	ADDRESS	2		Text width
300	(12C)	ADDRESS	1		
301	(12D)	ADDRESS	1		Option flag 2
302	(12E)	ADDRESS	1		Conversion flag 1
303	(12F)	ADDRESS	1		Conversion flag 2
304	(130)	ADDRESS	1		Conversion flag 3
305	(131)	ADDRESS	1		Conversion flag 4
306	(132)	ADDRESS	4		Multiplier

Comment

-----  
 Dual use work area for filtering.  
 The following fields map the data areas in a normal SCWAs. They are used to remember display or backup areas that have to be backed out later due to a filter mismatch.  
 -----

End of Comment

312	(138)	ADDRESS	4	SCWAFLTA (0)	Start of remapped area
312	(138)	ADDRESS	4	SCWADCWA	Addr of display SCWA of prefix area for text to back out
316	(13C)	SIGNED	2	SCWADCOF	Offset within display SCWA of prefix area for keyword (SCWANXPT)
318	(13E)	SIGNED	2	SCWADCLN	Remaining length in display SCWA after text is backed out (SCWADFAL)
320	(140)	ADDRESS	4	SCWADCTA	Addr of last prefix area (SCWALTA)
324	(144)	ADDRESS	4	SCWABCWA	Addr of backup SCWA of prefix area for sets to back out (SCWA address)

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
328	(148)	SIGNED	2	SCWABCOF	Offset within backup SCWA of prefix area for keyword (SCWABLUO)
330	(14A)	SIGNED	2	SCWABCLN	Remaining length in backup SCWA after sets are backed out (SCWABLA)
332	(14C)	SIGNED	2	SCWABCNA	Next remaining area (SCWABNO)
334	(14E)	SIGNED	2		Reserved for future use

Comment

-----  
 Work area in Filter SCWAs  
 The following fields map the data areas in a filter SCWA. These fields are used to store working fields and STAB addresses from earlier levels of \$SCAN.  
 -----

End of Comment

312	(138)	ADDRESS	4	SCWAFLST	Looping level \$SCANTAB for this filter SCWA
316	(13C)	SIGNED	4	SCWAFWA_START (0)	Begin filter WA (\$SCANTAB addr in filter SCWAs)
336	(150)	SIGNED	4	SCWAFWA_END (0)	End filter WA
336	(150)	X'13C'	0	SCWAFWA	"SCWAFWA_START,SCWAFWA_END-SCWAFWA_START" Define work area

Comment

SET SCWA BACKUP AREA FIELDS MAPPED OVER COMMON SCWA

End of Comment

92	(5C)	SIGNED	2	SCWABLA	LEN OF AVAILABLE BACKUP SPACE
94	(5E)	SIGNED	2	SCWABLT	LEN OF TOTAL BACKUP SPACE
96	(60)	SIGNED	2	SCWABLUO	OFFSET OF LAST USED BACKUP AREA
98	(62)	SIGNED	2	SCWABNO	OFFSET OF NEXT AVAILABLE BACKUP AREA
100	(64)	SIGNED	4	(0)	BACKUP AREA ELEMENTS
100	(64)	BITSTRING	16	SCWABELM	Ensure at least one fits

Comment

SET SCWA DISPLAY AREA FIELDS MAPPED OVER BACKUP FIELDS

End of Comment

92	(5C)	ADDRESS	4	SCWAOLDP	ADDR OF OLDEST PARENT SCWA
96	(60)	SIGNED	2	SCWADFAL	LENGTH OF FREE AREA LEFT
98	(62)	SIGNED	2	SCWANXPT	OFFSET TO NXT POSSIBLE TXT AREA
100	(64)	SIGNED	4	SCWALTA	ADDRESS TO PREVIOUS TEXT AREA
104	(68)	SIGNED	4	(0)	Beginning of display elmts
104	(68)	BITSTRING	16	SCWADELm	Ensure at least one fits
336	(150)	X'150'	0	SCWALEN	"*-SCWA" LEN OF GENERAL SCWA WORK AREA
336	(150)	X'1000'	0	SCWALEND	"4096" Len of DISPLAY SCWA

**Offsets**

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCWADA	, Mapping for display element
0	(0)	SIGNED	2	SCWADTXL	LEN OF TEXT
2	(2)	SIGNED	2	SCWADTLS	LEN OF TEXT IN THIS SCWA
4	(4)	SIGNED	4	SCWANDTA	ADDR OF NEXT TEXT AREA
8	(8)	ADDRESS	4	SCWADSTB	ADDR OF THIS TEXT'S STAB
12	(C)	BITSTRING	1	SCWADFLG	Flags for display
		1... ....		SCWADLTC	"B'10000000" Control line

## \$SCANWA Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		SCWADLTL	"B'01000000" Label line
		..1. ....		SCWADLTD	"B'00100000" Data line
		...1 ....		SCWADLTE	"B'00010000" End line
		.... 1..		SCWADFCR	"B'00001000" Place CRLF before text
		.... .1..		SCWADFCT	"B'00000100" This SCWADA continued in next display SCWA
		.... ..1.		SCWANIND	"B'00000010" Do not indent this output
13	(D)	BITSTRING	3		Reserved for future use
16	(10)	SIGNED	4	SCWADTXT (0)	START OF TEXT
16	(10)	X'10'	0	SCWADAL	**-SCWADA"

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCWABA	, BACKUP AREA ELEMENT
0	(0)	BITSTRING	1	SCWABAFG	FLAG BYTE FIELD CORRESPONDING TO THE FLAG BYTE OF SCWABFLG
		1... ....		SCWABABA	"B'10000000" BACKUP AREA CONTAINS BACKED UP STORAGE
		.1.. ....		SCWABADI	"B'01000000" BACKUP AREA CONTAINS STABNAME TO DISPLAY (FOR SET-DISPLAY)
		..1. ....		SCWABAER	"B'00100000" BACKUP AREA CONTAINS KEYWORD OR VALUE IN ERROR
		.... ..1.		SCWABASC	"B'00000010" Secondary TYPE=ERROR entry
		.... ...1		SCWABAIN	"B'00000001" Backup area is no longer valid
1	(1)	BITSTRING	1	SCWABALV	LEVEL OF THE VALUE SAVED
2	(2)	BITSTRING	2		RESERVED FOR FUTURE USE
4	(4)	ADDRESS	4	SCWABAAD	ADDRESS OF STORAGE BACKED UP
8	(8)	SIGNED	4	SCWABAAL	and ALET
12	(C)	SIGNED	2	SCWABALN	LENGTH OF STORAGE BACKED UP
14	(E)	SIGNED	2	SCWABAPO	OFFSET OF PREVIOUS BA IN SCWA OR 0
16	(10)	SIGNED	4	SCWABAFC (0)	CONTENTS OF BACKED-UP FIELD
16	(10)	X'10'	0	SCWABAL	**-SCWABA"

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	XWCWA	, ASAXWC work area
0	(0)	CHARACTER	4		Eyecatcher
4	(4)	CHARACTER	256	XWCDATA	Data work area
260	(104)	SIGNED	4	XWCDATAL	Length of data
264	(108)	CHARACTER	256	XWCSTR	Input string area
520	(208)	SIGNED	4	XWCSTRL	Length of input string MACDATE -06/16/09-<0>
0	(0)	X'20C'	0	M00M1330	"XWCLIST" ++ ASAXWC NAME
524	(20C)	SIGNED	4	XWCLIST (0)	++ ASAXWC PARM LIST
524	(20C)	CHARACTER	4	XWCLIST_XPARMAREA1	++ FIELD_LABEL
528	(210)	CHARACTER	24	XWCLIST_XPARMAREA2	++ FIELD_LABEL
528	(210)	X'228'	0	XWCLIST_PL_END	*** ++ END OF BASE PLIST
524	(20C)	ADDRESS	4	XWCLIST_XPATTERNSTR_ADDR	++ ADDR
528	(210)	SIGNED	4	XWCLIST_XPATTERNSTRLEN	++
532	(214)	ADDRESS	4	XWCLIST_XSTRING_ADDR	++ ADDR
536	(218)	SIGNED	4	XWCLIST_XSTRINGLEN	++
540	(21C)	ADDRESS	4	XWCLIST_XZEROORMORE_ADDR	++ ADDR
544	(220)	ADDRESS	4	XWCLIST_XONECHAR_ADDR	++ ADDR
548	(224)	ADDRESS	4	XWCLIST_XDELIMITER_ADDR	



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
524	(20C)	ADDRESS	4	XWCLIST_XPPPATTERNINFO_ADDR	++ ADDR
528	(210)	ADDRESS	4	XWCLIST_XPPPATTERNSTR_ADDR	++ ADDR
532	(214)	SIGNED	4	XWCLIST_XPPPATTERNSTRLEN	++ ADDR
536	(218)	ADDRESS	4	XWCLIST_XPPZEROORMORE_ADDR	++ ADDR
540	(21C)	ADDRESS	4	XWCLIST_XPPONECHAR_ADDR	++ ADDR
544	(220)	ADDRESS	4	XWCLIST_XPPDELIMITER_ADDR	++ ADDR
528	(210)	ADDRESS	4	XWCLIST_XPPSTRING_ADDR	++ ADDR
532	(214)	SIGNED	4	XWCLIST_XPPSTRINGLEN	++ ADDR
552	(228)	X'1C'	0	XWCLISTL	++ "-XWCLIST" ++ LENGTH OF PLIST

Comment

---

ASAXWC-0

---

End of Comment

552	(228)	BITSTRING	256	XWCAREA	Work area passed to ASAXWC
808	(328)	BITSTRING	40	XWCGENWA	List of diagnostic levels
848	(350)	CHARACTER	100	XWCGENWB	Message work area
952	(3B8)	DBL WORD	8	(0)	
952	(3B8)	X'3B8'	0	XWCLEN	"*-XWCWA" Length of work area
952	(3B8)	X'EE'	0	XWCWORDS	"XWCLEN/4" Length in words

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCWA	Restore SCWA DSECT

**\$SCANWA Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
M00M1330	0	20C	SCWA\$TOX	C4	80
SCWA	0		SCWABA	0	
SCWA	0		SCWABAAD	4	
SCWA\$ARM	C4	4	SCWABAAL	8	
SCWA\$CDU	C4	40	SCWABABA	0	80
SCWA\$COR	C4	80	SCWABADI	0	40
SCWA\$CPU	C4	1	SCWABAER	0	20
SCWA\$ECA	C4	1	SCWABAFB	10	
SCWA\$ESH	C4	2	SCWABAFG	0	
SCWA\$EST	C4	4	SCWABAIN	0	1
SCWA\$FRC	C4	8	SCWABAL	10	10
SCWA\$IND	C4	80	SCWABALN	C	
SCWA\$JPR	C4	10	SCWABALV	1	
SCWA\$NNM	C4	80	SCWABAPO	E	
SCWA\$NTC	C4	80	SCWABASC	0	2
SCWA\$RSV	C4	80	SCWABCLN	14A	
SCWA\$TJC	C4	40	SCWABCNA	14C	
SCWA\$TJD	C4	2	SCWABCOF	148	
SCWA\$TJE	C4	8	SCWABCWA	144	
SCWA\$TJI	C4	4	SCWABELM	64	
SCWA\$TJP	C4	80	SCWABFLG	EC	
SCWA\$TJS	C4	10	SCWABLA	5C	
SCWA\$TJX	C4	20	SCWABLT	5E	

## \$SCANWA Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SCWABLUO	60		SCWAFLG4	E4	
SCWABNO	62		SCWAFLG5	E5	
SCWABNWA	10		SCWAFLG6	21	
SCWABPWA	8		SCWAFLG7	C6	
SCWACALD	C7		SCWAFLG8	EF	
SCWACALR	EB		SCWAFLG9	D8	
SCWACATL	C4	40	SCWAFLST	138	
SCWACBAD	8C		SCWAFLTA	138	
SCWACBAL	90		SCWAFNWA	14	
SCWACBCL	88		SCWAFVCT	F0	
SCWACMPR	C4	2	SCWAFWA	150	13C
SCWACNTR	C0		SCWAFWA_END	150	
SCWACPCT	C4	4	SCWAFWA_START		
SCWACR	E0	8		13C	
SCWACRDI	E0	28	SCWAFW16	100	F8
SCWACRRT	E0	C4	SCWAFW8	100	F8
SCWACR11	5C		SCWAFW8A	100	100
SCWADA	0		SCWAGRPD	E3	10
SCWADADD	DC		SCWAHASP	E2	40
SCWADAD2	A0		SCWAID	0	
SCWADAL	10	10	SCWAILEN	BA	
SCWADALL	E2	80	SCWAINDL	DA	
SCWADCLN	13E		SCWAIPTR	BC	
SCWADCOF	13C		SCWAJNET	C4	80
SCWADCOM	E1	1	SCWAKBAK	20	20
SCWADCTA	140		SCWAKDSP	20	40
SCWADCWA	138		SCWAKFLT	20	10
SCWADDEL	E0	44	SCWAKIND	20	
SCWADELE	E0	4	SCWAKNOR	20	80
SCWADELM	68		SCWAKPTR	B0	
SCWADFAL	60		SCWAKSUB	20	8
SCWADFCR	C	8	SCWAKWDL	EA	
SCWADFACT	C	4	SCWALEN	150	150
SCWADFLG	C		SCWALEND	150	1000
SCWADISP	E0	40	SCWALOOOP	E2	8
SCWADLEN	6E		SCWALTA	64	
SCWADLST	E3	4	SCWAMIDL	F2	
SCWADLTC	C	80	SCWAMLVL	E3	80
SCWADLTD	C	20	SCWAMSDL	116	
SCWADLTE	C	10	SCWAMSG	E0	50
SCWADLTL	C	40	SCWANBLN	117	
SCWADNWA	C		SCWANDTA	4	
SCWADOUT	70		SCWANIND	C	2
SCWADPWA	4		SCWANWA	7C	
SCWADRTRN	74		SCWANXPT	62	
SCWADSPA	E0	20	SCWAOLDP	5C	
SCWADSPR	E9		SCWAORG1	5C	
SCWADSTB	8		SCWAOTAB	84	
SCWADTLS	2		SCWAPAR	E1	80
SCWADTXL	0		SCWAPARN	E3	40
SCWADTXT	10		SCWAPAR2	E2	2
SCWADWK1	30		SCWAPCNT	115	
SCWADWK2	38		SCWAPERD	E2	1
SCWADWK3	40		SCWAPRRC	114	
SCWADWRK	28		SCWAPSCN	E1	8
SCWAD1ST	E3	8	SCWAPWA	78	
SCWAELVL	23		SCWARANG	E1	4
SCWAENWA	1C		SCWARLEN	B8	
SCWAERR	E3	20	SCWARLWA	24	
SCWAEXFL	C4		SCWARMSH	C4	8
SCWAFAD	94		SCWARMTA	C4	40
SCWAFAL	98		SCWARMTL	C4	10
SCWAFLG1	E1		SCWARPMM	E6	
SCWAFLG2	E2		SCWARPTR	B4	
SCWAFLG3	E3		SCWARTCD	AE	

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SCWARTRY	C4	20	SCWA5XPO	E5	2
SCWASBL	D6		SCWA5XPR	E5	4
SCWASCAN	E1	2	SCWA6BNO	21	40
SCWASCD	E0	A8	SCWA6EQ	21	1
SCWASCND	120		SCWA6GEN	21	80
SCWASCRL	D0		SCWA6GT	21	4
SCWASEPR	E8		SCWA6LT	21	2
SCWASET	E0	80	SCWA6MSS	21	10
SCWASETC	E0	88	SCWA6NCR	21	20
SCWASETD	E0	A0	SCWA6NOT	21	7
SCWASING	E1	40	SCWA60SS	21	8
SCWASLEN	6C		SCWA7BOU	C6	80
SCWASLVL	22		SCWA7DAL	C6	4
SCWASNWA	18		SCWA7DDN	C6	10
SCWASPAN	E2	10	SCWA7DNF	C6	40
SCWASSDR	110		SCWA7DSP	C6	2
SCWASSER	E1	20	SCWA7FLF	C6	20
SCWASSIE	F4		SCWA7FXT	C6	8
SCWASSL2	EE		SCWA7SDS	C6	1
SCWASSS	F8		SCWA8DCC	EF	4
SCWASSSA	10C		SCWA8DHC	EF	2
SCWASSSC	F8		SCWA8DOU	EF	8
SCWASSSH	100		SCWA8HIC	EF	1
SCWASSSL	ED		SCWA8LTC	EF	80
SCWASSSS	108		SCWA8LTD	EF	20
SCWASTAB	80		SCWA8LTE	EF	10
SCWASTBS	64		SCWA8LTL	EF	40
SCWASTMD	C4	80	SCWA9NFT	D8	80
SCWASTMT	68		XWCAREA	228	
SCWASTVC	D4		XWCDATA	4	
SCWASUBH	CC		XWCATAL	104	
SCWASUBS	C8		XWCGENWA	328	
SCWATEMP	9C		XWCGENWB	350	
SCWATOKN	60		XWCLEN	3B8	3B8
SCWATYPE	E0		XWCLIST	20C	
SCWAVCNT	D2		XWCLIST_PL_END		
SCWAVECT	E1	10		210	228
SCWAVERB	118		XWCLIST_XDELIMITER_ADDR		
SCWAWARN	C5			224	
SCWAWCWA	58		XWCLIST_XONECHAR_ADDR		
SCWAWK16	40	28		220	
SCWAWK20	40	28	XWCLIST_XPARMAREA1		
SCWAWK24	40	28		20C	
SCWAWK32	40	28	XWCLIST_XPARMAREA2		
SCWAWORK	A4			210	
SCWAW16B	40	38	XWCLIST_XPATTERNSTR_ADDR		
SCWAXWA	48	40404040		20C	
SCWA2LNG	E2	20	XWCLIST_XPATTERNSTRLEN		
SCWA2QSS	E2	4		210	
SCWA3DCT	E3	1	XWCLIST_XPPDELIMITER_ADDR		
SCWA3TCB	E3	2		220	
SCWA4ACT	E4	20	XWCLIST_XPPONECHAR_ADDR		
SCWA4ETL	E4	2		21C	
SCWA4FLM	E4	4	XWCLIST_XPPPATTERNINFO_ADDR		
SCWA4LFC	E4	8		20C	
SCWA4PSS	E4	40	XWCLIST_XPPPATTERNSTR_ADDR		
SCWA4RDE	E4	1		210	
SCWA4SDL	E4	10	XWCLIST_XPPPATTERNSTRLEN		
SCWA4SSG	E4	80		214	
SCWA5DSP	E5	20	XWCLIST_XPPSTRING_ADDR		
SCWA5FLT	E5	80		210	
SCWA5FND	E5	8	XWCLIST_XPPSTRINGLEN		
SCWA5FRJ	E5	40		214	
SCWA5NSS	E5	1	XWCLIST_XPPZEROORMORE_ADDR		
SCWA5PS2	E5	10		218	

## \$SCANWA Cross Reference

Name	Hex Offset	Hex Value
XWCLIST_XSTRING_ADDR	214	
XWCLIST_XSTRINGLEN	218	
XWCLIST_XZEROORMORE_ADDR	21C	
XWCLISTL	228	1C
XWCSTR	108	
XWCSTRL	208	
XWCWA	0	
XWCWORDS	3B8	EE

---

## **\$SCAT Information**

### **\$SCAT Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SCAT**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$SCAT Map

### \$SCAT Heading Information

**Common Name:** SYSOUT Class Attribute Table DSECT  
**Macro ID:** \$SCAT  
**DSECT Name:** SCAT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: 241  
 Key: 1  
 Residency: The SCAT is in 24 bit virtual storage and 64 bit real storage.  
**Size:** See SCATLEN  
**Created by:** HASPIRSI during JES2 initialization  
**Pointed to by:** CCTSCATP field of the \$HCCT data area  
 CCTSCAT FIELD of the \$HCCT data area  
 (CCTSCAT is the actual address of the SCAT which resides in the \$HCCT. It is NOT a pointer and should not be used to address the SCAT.)  
**Serialization:** None required  
**Function:** The SCAT defines the attributes of the JES2 SYSOUT classes. There are 64 SCAT entries arranged contiguously. The appropriate SCAT entry for a particular class is found by taking the class (e.g. class A = X'C1'), turning off the high order two bits (e.g. class A =X'01'), multiplying by the SCATLEN equate, and adding to the address pointed to by CCTSCATP.

### \$SCAT Map

Offsets		Type/Value	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	SCAT	SYSOUT CLASS ATTRIBUTE TABLE ELEMENT
0	(0)	BITSTRING	1	SCATFLG1	SYSOUT CLASS FLAG BYTE 1
		111. ....		SCATYPES	"B'11100000" SCAT ELEMENT TYPES
		1... ....		SCAT1PRT	"B'10000000" SYSOUT CLASS NORMALLY PRINTED
		.1.. ....		SCAT1PCH	"B'01000000" SYSOUT CLASS NORMALLY PUNCHED
		..1. ....		SCAT1DUM	"B'00100000" TREAT SYSOUT CLASS AS DUMMY
		...1 ....		SCAT1BLK	"B'00010000" TRUNCATE THIS SYSOUT CLASS
		.... 1...		SCAT1TCL	"B'00001000" TRAKCELL THIS SYSOUT CLASS
		.... ...1		SCAT1INV	"B'00000001" INVALID SYSOUT CLASS
1	(1)	BITSTRING	1	SCATFLG2	SYSOUT CLASS FLAG BYTE 2
1	(1)	X'10'	0	SCT2NODP	"\$ODPURGE" NORMAL OUTDISP=PURGE
1	(1)	X'8'	0	SCT2NODW	"\$ODWRITE" NORMAL OUTDISP=WRITE
1	(1)	X'4'	0	SCT2NODH	"\$ODHOLD" NORMAL OUTDISP=HOLD
1	(1)	X'2'	0	SCT2NODK	"\$ODKEEP" NORMAL OUTDISP=KEEP
1	(1)	X'1'	0	SCT2NODL	"\$ODLEAVE" NORMAL OUTDISP=LEAVE
1	(1)	X'1F'	0	SCT2NODA	"\$ODANYWP" CHECK ALL BIT SETTINGS
2	(2)	BITSTRING	1	SCATFLG3	SYSOUT CLASS FLAG BYTE 3
2	(2)	X'10'	0	SCT3AODP	"\$ODPURGE" ABNORMAL OUTDISP=PURGE
2	(2)	X'8'	0	SCT3AODW	"\$ODWRITE" ABNORMAL OUTDISP=WRITE
2	(2)	X'4'	0	SCT3AODH	"\$ODHOLD" ABNORMAL OUTDISP=HOLD
2	(2)	X'2'	0	SCT3AODK	"\$ODKEEP" ABNORMAL OUTDISP=KEEP
2	(2)	X'1'	0	SCT3AODL	"\$ODLEAVE" ABNORMAL OUTDISP=LEAVE
2	(2)	X'1F'	0	SCT3AODA	"\$ODANYWP" CHECK ALL BIT SETTINGS
2	(2)	X'3'	0	SCATLEN	**SCAT" LENGTH OF A SCAT ENTRY

**\$\$SCAT Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SCAT	0	
SCATFLG1	0	
SCATFLG2	1	
SCATFLG3	2	
SCATLEN	2	3
SCATYPES	0	E0
SCAT1BLK	0	10
SCAT1DUM	0	20
SCAT1INV	0	1
SCAT1PCH	0	40
SCAT1PRT	0	80
SCAT1TCL	0	8
SCT2NODA	1	1F
SCT2NODH	1	4
SCT2NODK	1	2
SCT2NODL	1	1
SCT2NODP	1	10
SCT2NODW	1	8
SCT3AODA	2	1F
SCT3AODH	2	4
SCT3AODK	2	2
SCT3AODL	2	1
SCT3AODP	2	10
SCT3AODW	2	8

## \$SCAT Cross Reference



---

## \$SCID Information

### \$SCID Programming Interface information

---

Programming Interface information

#### \$SCID

The following fields are **NOT** programming interface information:

- SCIDDSB
- SCIDSPUD
- SPUD\_ACTIVE\_COUNT
- SPUD\_CHECK\_AREA
- SPUD\_DATASET\_SIZE
- SPUD\_ENQUEUE\_COUNT
- SPUD\_FREE\_ALET
- SPUD\_FREE\_QUEUE
- SPUD\_FREE\_STOKEN
- SPUD\_HOLD
- SPUD\_HOLD\_ALET
- SPUD\_HOLD\_STOKEN
- SPUD\_LATEST\_ALET
- SPUD\_LATEST\_STOKEN
- SPUD\_LATEST\_VERSION
- SPUD\_LIVE\_ALET
- SPUD\_LIVE\_STOKEN
- SPUD\_LIVE\_VERSION

---

End of Programming Interface information

## Heading Information • \$SCID Map

### \$SCID Heading Information

**Common Name:** Summary of Checkpoint Information  
**Macro ID:** \$SCID  
**DSECT Name:** SCID  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SCID  
 Offset: SCIDID-SCID  
 Length: L'SCIDID  
**Storage Attributes:** Subpool: 231  
 Key: 1  
 Residency: Virtual and real storage are anywhere, above or below 16M, in common storage.  
**Size:** See SCIDSIZE  
**Created by:** HASPCKVR during initialization processing  
**Pointed to by:** CCTSCIDS field of the \$HCCT data area  
**Serialization:** All applicable techniques  
**Function:** This control block contains the necessary information needed by the Checkpoint Versions Subtask. It provides the means by which authorized programs access the checkpoint versions contained within the checkpoint data spaces.

### \$SCID Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCID	
0	(0)		16	(0)	Quad word align
0	(0)	CHARACTER	4	SCIDID	SCID eye catcher
4	(4)	BITSTRING	1	SCIDSUBP	SCID subpool
5	(5)	BITSTRING	3	SCIDLEN	SCID length
8	(8)	BITSTRING	1	SCIDCBVN	Control block vers. no
9	(9)	BITSTRING	1	SCIDFLG	Flag Byte-subtask stats
		1... ....		SCIDDSE	"B'10000000" Data spaces exist
		.1.. ....		SCIDDISA	"B'01000000" Subtask disabled
		..1. ....		SCIDPJ2	"B'00100000" Subtask in PJES2
		...1 ....		SCIDSINA	"B'00010000" Subtask inactive
		.... 1...		SCIDSREC	"B'00001000" Subtask in recovery
10	(A)	BITSTRING	1	SCIDFLG2	Flag Byte-Versning stat
		.1.. ....		SCIDVACT	"B'01000000" Versioning active
11	(B)	BITSTRING	1		
12	(C)	CHARACTER	8	SCIDNAME (0)	Requested name of data sp
12	(C)	CHARACTER	4	SCIDSTNM	Subtask name 'CKVR'
16	(10)	CHARACTER	4	SCIDSSNM	Subsystem name 'JESX'
20	(14)	CHARACTER	8	SCIDDSPN	Official name of d.s.
28	(1C)	SIGNED	4	SCIDDSIZ	Size of data space
32	(20)	ADDRESS	4	SCIDSORG	Origin of data space
36	(24)	SIGNED	4	SCIDVRNO	Running version number
40	(28)	ADDRESS	4	SCIDDSB	Addr of data space DSB
44	(2C)	CHARACTER	8	SCIDSTCK (0)	Time of last request
44	(2C)	SIGNED	4	SCIDREQT	Primary part -time
48	(30)	SIGNED	4		Last part of time
52	(34)	CHARACTER	16	SCIDCVMN	Cur version ENQ minor name
68	(44)	CHARACTER	16	SCIDNVMN	Next version ENQ minor name
84	(54)	SIGNED	2	SPUD_MAX_VERSIONS	
					Max versions in dataspace
86	(56)	BITSTRING	10		Reserved for alignment
96	(60)			(0)	Quad word align
96	(60)	CHARACTER	16	SPUD_LATEST_VERSION	
				(0)	

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
					Active Q head area
96	(60)	ADDRESS	4	SPUD_LATEST_ADDR	Active Q head addr
100	(64)	SIGNED	4	SPUD_LATEST_ALET	Active ALET
104	(68)	BITSTRING	8	SPUD_LATEST_STOKEN	Active STOKEN
112	(70)	BITSTRING	16	SPUD_FREE_QUEUE (0)	Free Q head area
112	(70)	ADDRESS	4	SPUD_FREE_ADDR	Free Q head addr
116	(74)	SIGNED	4	SPUD_FREE_ALET	Free ALET
120	(78)	BITSTRING	8	SPUD_FREE_STOKEN	STOKEN of free version
128	(80)	BITSTRING	16	SPUD_HOLD (0)	Hold area CVCB
128	(80)	ADDRESS	4	SPUD_HOLD_ADDR	Hold area Address
132	(84)	SIGNED	4	SPUD_HOLD_ALET	Hold ALET
136	(88)	BITSTRING	8	SPUD_HOLD_STOKEN	STOKEN of free area
144	(90)	BITSTRING	16	SPUD_LIVE_VERSION (0)	Live version CVCB area
144	(90)	ADDRESS	4	SPUD_LIVE_ADDR	Ptr to live version CVCB
148	(94)	SIGNED	4	SPUD_LIVE_ALET	Live ALET
152	(98)	BITSTRING	8	SPUD_LIVE_STOKEN	STOKEN of live version
160	(A0)	BITSTRING	16	OLD_LIVE_VERSION (0)	Old live version CVCB area
160	(A0)	ADDRESS	4	OLD_LIVE_ADDR	Old live version CVCB addr
164	(A4)	SIGNED	4	OLD_LIVE_ALET	Old live ALET
168	(A8)	BITSTRING	8	OLD_LIVE_STOKEN	Old STOKEN of live version
176	(B0)	ADDRESS	4	SPUD_CHECK_AREA	Low end of area
180	(B4)	SIGNED	4	SPUD_ACTIVE_COUNT	Number in active queue
184	(B8)	SIGNED	4	SPUD_ENQUEUE_COUNT	Number of CVCBs w/ enqueues
188	(BC)	SIGNED	4		Reserved
192	(C0)	SIGNED	4	SPUD_DATASET_SIZE	Size MR+4K pages
196	(C4)	SIGNED	4	(4)	Reserved
196	(C4)	X'D4'	0	SCIDSIZE	"*-SCID"
		.... ..1.		SCIDCVNO	"X'02"
196	(C4)	X'C3C9C4'	0	SCIDEYEC	"C'SCID"

## \$SCID Cross Reference

### \$SCID Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
OLD_LIVE_ADDR			SPUD_LATEST_STOKEN		
	A0			68	
OLD_LIVE_ALET			SPUD_LATEST_VERSION		
	A4			60	
OLD_LIVE_STOKEN			SPUD_LIVE_ADDR		
	A8			90	
OLD_LIVE_VERSION			SPUD_LIVE_ALET		
	A0			94	
SCID	0		SPUD_LIVE_STOKEN		
SCIDCBVN	8			98	
SCIDCVMN	34		SPUD_LIVE_VERSION		
SCIDCVNO	C4	2		90	
SCIDDISA	9	40	SPUD_MAX_VERSIONS		
SCIDDSB	28			54	
SCIDDSSE	9	80			
SCIDDSIZ	1C				
SCIDDSPN	14				
SCIDEYEC	C4	C3C9C4			
SCIDFLG	9				
SCIDFLG2	A				
SCIDID	0				
SCIDLEN	5				
SCIDNAME	C				
SCIDNVMN	44				
SCIDPJ2	9	20			
SCIDREQT	2C				
SCIDSINA	9	10			
SCIDSIZE	C4	D4			
SCIDSORG	20				
SCIDSREC	9	8			
SCIDSSNM	10				
SCIDSTCK	2C				
SCIDSTNM	C				
SCIDSUBP	4				
SCIDVACT	A	40			
SCIDVRNO	24				
SPUD_ACTIVE_COUNT					
	B4				
SPUD_CHECK_AREA					
	B0				
SPUD_DATASET_SIZE					
	C0				
SPUD_ENQUEUE_COUNT					
	B8				
SPUD_FREE_ADDR					
	70				
SPUD_FREE_ALET					
	74				
SPUD_FREE_QUEUE					
	70				
SPUD_FREE_STOKEN					
	78				
SPUD_HOLD	80				
SPUD_HOLD_ADDR					
	80				
SPUD_HOLD_ALET					
	84				
SPUD_HOLD_STOKEN					
	88				
SPUD_LATEST_ADDR					
	60				
SPUD_LATEST_ALET					
	64				

---

## **\$SCK Information**

### **\$SCK Programming Interface information**

Programming Interface information

**\$SCK**

End of Programming Interface information

## Heading Information • \$SCK Map

### \$SCK Heading Information

**Common Name:** NJE/TCP Socket  
**Macro ID:** \$SCK  
**DSECT Name:** SCK  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SCK  
 Offset: SCKID  
 Length: 4  
**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: VIRTUAL - anywhere REAL - anywhere  
**Size:** See SCKLEN  
**Created by:** SOCKDYN service in HASPTCP  
**Pointed to by:** MDCTSCK field of the \$DCT data area  
 \$SOCKTBL field of the \$HCT data area  
 SCKNEXT field of the \$SCK data area  
**Serialization:** JES2 main task  
**Function:** An SCK describes a TCP/IP socket.

### \$SCK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCK	TCP/IP socket DSECT
0	(0)	CHARACTER	4	SCKID	Control block identifier
0	(0)	X'1'	0	SCKVRNUM	"1" Control block version equate
4	(4)	ADDRESS	1	SCKVRSN	Control block version
5	(5)	ADDRESS	3		Reserved for future use
8	(8)	CHARACTER	8	SCKNAME	SCK symbolic name
16	(10)	ADDRESS	4	SCKNEXT	Addr of next SCK
20	(14)	CHARACTER	255	SCKHNAME	IP host name for this socket
275	(113)	CHARACTER	1		
276	(114)	BITSTRING	16	SCKIPAD	IP address (binary)
292	(124)	CHARACTER	16	SCKPORTN	Associated Port Name
308	(134)	SIGNED	2	SCKPORT	Associated Port Number
310	(136)	SIGNED	2	SCKNODE	Node number where socket exists
312	(138)	SIGNED	2	SCKLINE	Dedicated line number
314	(13A)	SIGNED	2	SCKREST	Resistance
316	(13C)	SIGNED	2	SCKSERV	Server DCT number
318	(13E)	BITSTRING	1	SCKFLAG1	Flags
		1... ..		SCK1SECU	"B'10000000" Secure socket
		.1.. ....		SCK1DPRT	"B'01000000" Default port number used
		..1. ....		SCK1DIP	"B'00100000" Default IP address used
		...1 ....		SCK1ACT	"B'00010000" Socket active
		.... 1...		SCK1ANCY	"B'00001000" Automatically start NJE to this socket
		.... .1..		SCK1ANCN	"B'00000100" Never automatically start NJE to this socket
319	(13F)	BITSTRING	1		Reserved
320	(140)	ADDRESS	4	SCKSDCT	Address of NETSRV DCT
324	(144)	ADDRESS	4	SCKLDCT	Address of LINE DCT
328	(148)	SIGNED	4	SCKSKID	Socket id (assigned by IAZNJSTK)
332	(14C)	SIGNED	2	SCKANINT	Restart interval (minutes)
334	(14E)	BITSTRING	2		Reserved
336	(150)	SIGNED	4	SCKANTIM	Disconnect time (STCK)
344	(158)	ADDRESS	8	SCKCDCT	CDCT address
352	(160)	DBL WORD	8	(0)	
352	(160)	X'160'	0	SCKLEN	"*-SCK" LENGTH OF SCK

**\$SCK Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SCK	0	
SCKANINT	14C	
SCKANTIM	150	
SCKCDCT	158	
SCKFLAG1	13E	
SCKHNAME	14	
SCKID	0	
SCKIPAD	114	
SCKLDCT	144	
SCKLEN	160	160
SCKLINE	138	
SCKNAME	8	40404040
SCKNEXT	10	
SCKNODE	136	0
SCKPORT	134	
SCKPORTN	124	
SCKREST	13A	
SCKSDCT	140	
SCKSERV	13C	
SCKSKID	148	
SCKVRNUM	0	1
SCKVRSN	4	
SCK1ACT	13E	10
SCK1ANCN	13E	4
SCK1ANCY	13E	8
SCK1DIP	13E	20
SCK1DPRT	13E	40
SCK1SECU	13E	80

## \$SCK Cross Reference



## \$SCT Information

### \$SCT Heading Information

**Common Name:** SCT  
**Macro ID:** \$SCT  
**DSECT Name:** SCT  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SCT  
 Offset: SCTSCTID  
 Length: L'SCTSCTID  
**Storage Attributes:** Subpool: 0  
 Key: 1  
 Residency: Virtual and real storage are above 16M, in the private storage of the JES2 address space.  
**Size:** See SCTLLEN  
**Created by:** HASPIRMA  
**Pointed to by:** \$SCT field of the HCT data area  
**Serialization:** Normal JES2 PCE dispatch serialization  
**Function:** The SCT contains data relevant to the execution of the Spin PCEs. It is used by the Spin PCEs for spin processing related communication.

### \$SCT Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCT	
0	(0)	X'1000'	0	SCTLLEN	"4096" Length of SCT
0	(0)	CHARACTER	4	SCTSCTID	SCT eyecatcher
4	(4)	ADDRESS	1	SCTVER	SCT version number
4	(4)	X'2'	0	SCTVERN	"2" SCT version
5	(5)	ADDRESS	1	SCTFLAG1	Spin PCE flags
		1... ....		SCT1TDIS	"B'10000000" Spin PCE trace disabled
		.1... ....		SCT1TRCI	"B'01000000" Trace table initialized
6	(6)	SIGNED	2		Reserved for future use

Comment

Spin processing count fields. The following counts are maintained:

SCTSPPR: Count of all spin IOTs processed. This field is incremented by one each time HASPSPIN processes a spin IOT from the FIFO queue in the HCCT. This includes IOTs for which JOEs are built as well as IOTs which are unspun.

SCTSPUN: Count of all spin IOTs which have been unspun.

SCTUNPR: Count of all unspun IOTs which have been successfully processed for output.

In general: > SCTSPUN = SCTUNPR when no unspun IOTs exist.

> CCTSPINC = SCTSPPR when no spin IOTs exist in the HCCT (CCTSPIOT=CCTFIFOQ=0).

Errors (as reported by \$DISTERR) will affect the consistency of these counts.

End of Comment

8	(8)	SIGNED	4	SCTSPPR	Count of spin IOTs proc'd
12	(C)	SIGNED	4	SCTSPUN	Count of spin IOTs unspun
16	(10)	SIGNED	4	SCTUNPR	Count of unspun IOTs proc'd
20	(14)	SIGNED	4		Reserved

## \$SCT Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>HASPSPIN Recovery information and parameters.            The time interval values in seconds are used to compare against the first word of the STCK value. POPs indicates that bit position 31 of the clock is incremented every 1.048576 seconds. For the purposes of determining abend intervals this is considered a close enough approximation of one second.</p>					
End of Comment					
24	(18)	SIGNED	4	SCTABDT	Count of abends - total
32	(20)	DBL WORD	8	SCTABTIM	HASPSPIN abend time marker
32	(20)	X'258'	0	SCT10MIN	"10*60" 10 minutes (in seconds)
32	(20)	X'4B0'	0	SCT20MIN	"20*60" 20 minutes (in seconds)
40	(28)	SIGNED	4	SCTABD20	Count of abends in 20 min.
44	(2C)	SIGNED	4		Reserved for future use
48	(30)	SIGNED	4		Reserved for future use
52	(34)	SIGNED	4		Reserved for future use
56	(38)	SIGNED	4		Reserved for future use
60	(3C)	SIGNED	4		Reserved for future use
64	(40)	DBL WORD	8		Reserved for future use

### Comment

The Spin PCE trace table occupies the remainder of the SCT.

Note that a minimum of ten entries are defined. The actual number of entries is a function of the remaining space in the SCT up to the actual size as defined by SCTLLEN.

### End of Comment

72	(48)	DBL WORD	8	(0)	
72	(48)	SIGNED	4	SCTTCUR	Addr of current trace entry
76	(4C)	SIGNED	4	SCTTLAST	Addr of last trace entry
80	(50)	DBL WORD	8	(0)	
80	(50)	ADDRESS	4	SCTTTAB (0)	HASPSPIN trace table ** minimum of 10 entries **

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SCTTRENT	
0	(0)	SIGNED	4	SCTTWD0	Spin PCE trace entry word 0
0	(0)	X'0'	0	SCTTTYP0	"0" Trace type 0
0	(0)	X'1'	0	SCTTTYP1	"1" Trace type 1
0	(0)	X'2'	0	SCTTTYP2	"2" Trace type 2
0	(0)	X'3'	0	SCTTTYP3	"3" Trace type 3
0	(0)	X'4'	0	SCTTTYP4	"4" Trace type 4
0	(0)	X'5'	0	SCTTTYP5	"5" Trace type 5
0	(0)	X'6'	0	SCTTTYP6	"6" Trace type 6
4	(4)	SIGNED	4	SCTTWD1	Spin PCE trace entry word 1
8	(8)	SIGNED	4	SCTTWD2	Spin PCE trace entry word 2
12	(C)	SIGNED	4	SCTTWD3	Spin PCE trace entry word 3
16	(10)	SIGNED	4	SCTTWD4	Spin PCE trace entry word 4
20	(14)	SIGNED	4	SCTTWD5	Spin PCE trace entry word 5
24	(18)	SIGNED	4	SCTTWD6	Spin PCE trace entry word 6
28	(1C)	SIGNED	4	SCTTWD7	Spin PCE trace entry word 7
28	(1C)	X'20'	0	SCTTESIZ	**SCTTRENT" Size of single trace entry

**\$SCT Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SCT	0	
SCTABDT	18	
SCTABD20	28	
SCTABTIM	20	
SCTFLAG1	5	
SCTLEN	0	1000
SCTSCTID	0	E2C3E340
SCTSPPR	8	0
SCTSPUN	C	0
SCTTCUR	48	
SCTTESIZ	1C	20
SCTTLAST	4C	
SCTTRENT	0	
SCTTTAB	50	
SCTTTYP0	0	0
SCTTTYP1	0	1
SCTTTYP2	0	2
SCTTTYP3	0	3
SCTTTYP4	0	4
SCTTTYP5	0	5
SCTTTYP6	0	6
SCTTWD0	0	
SCTTWD1	4	
SCTTWD2	8	
SCTTWD3	C	
SCTTWD4	10	
SCTTWD5	14	
SCTTWD6	18	
SCTTWD7	1C	
SCTUNPR	10	0
SCTVER	4	
SCTVERN	4	2
SCT1TDIS	5	80
SCT1TRCI	5	40
SCT10MIN	20	258
SCT20MIN	20	4B0



## **\$SDB Information**

### **\$SDB Programming Interface information**

Programming Interface information

#### **\$SDB**

The following fields are **NOT** programming interface information:

- SDBAPBL
- SDBDEB
- SDBJFCB
- SDBJFCBE
- SDBPBLAC
- SDBPBLFL
- SDBPBLIN
- SDBRPL
- SDBTCBM
- SDBTCBO
- SDBTRK
- SDBTRKF
- SDBTRKL
- SDBUPRBA

End of Programming Interface information

## Heading Information • \$SDB Map

### \$SDB Heading Information

**Common Name:** SDB - JES2 Subsystem Dataset Block  
**Macro ID:** \$SDB  
**DSECT Name:** SDB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SDB '  
 Offset: SDBID-SDB  
 Length: 4

**Storage Attributes:** Subpool: 249  
 Key: 1  
 Residency: Virtual storage is in 31 bit storage, real can be in 64 bit storage, in the address space of the application that is reading or writing the subsystem dataset represented by this SDB.

**Size:** See SDBLNG. The actual length will be stored in SDBLENG when the \$SDB is created.

**Created by:** The \$SDBINIT service routine and the FGDSALOC routine.

**Pointed to by:** GCBSDB field of the \$GDB data area  
 SJBSDB field of the \$SJB data area  
 SJXWTCHN field of the \$SJXB data area  
 SDBSDB field of the \$SDB data area  
 SDBWTCHN field of the \$SDB data area  
 DEBIRBB field of the DEB data area (after OPEN)  
 contains bits 1-24 of the address

**Serialization:** HAM uses an SDB lock (with an ENQ) to serialize all puts to the SDB and any authorized functions.

**Function:** The SDB represents a subsystem dataset. It indicates the state of the dataset (open/closed, input/output, I/O active, etc). It holds pointers to other subsystem control blocks and holds the address (MTTR) of the next available record on SPOOL for output. The chain of buffers needed for I/O is chained to it.

### \$SDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SDB	
0	(0)	BITSTRING	168	SDBSAVE	Save area
168	(A8)	CHARACTER	4	SDBID	SDB IDENTIFIER
172	(AC)	SIGNED	2	SDBLENG	SDB LENGTH
174	(AE)	BITSTRING	2		Reserved
176	(B0)	SIGNED	4	SDBR14SV	HPMOVE R14 save area
180	(B4)	BITSTRING	1	SDBFLG1	FLAG BYTE 1
		1... ....		SDB1GET	"B'10000000" GET ALLOWED
		.1.. ....		SDB1ENQ	"B'01000000" Exit 9 ENQ obtained
		..1. ....		SDB1PUT	"B'00100000" PUT ALLOWED
		...1 ....		SDB1HPUT	"B'00010000" At least one put done (data set is not null)
		.... 1...		SDB1OUT	"B'00001000" CARRIAGE CONTROL ALLOWED
		.... .1..		SDB1CLOS	"B'00000100" Do not get chaining track, data set is closing
		.... ..1.		SDB1ENDR	"B'00000010" Do not get chaining track, ENDREQ request
		.... ...1		SDB1FOPN	"B'00000001" INIT DATA SET - FAKE-OPENED
181	(B5)	BITSTRING	1	SDBFLG2	FLAG BYTE 2
		1... ....		SDB2IOE	"B'10000000" Permanent I/O error
		.1.. ....		SDB2VAL	"B'01000000" Validation error
		..1. ....		SDB2VDK	"B'00100000" Data set key mismatch
		...1 ....		SDB2EOD	"B'00010000" End of data set

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... 1...		SDB2GLM	"B'00001000" Locate mode gets allowed
		.... .1..		SDB2DSRS	"B'00000100" Data set restart- EOD or IOE
		.... ..1.		SDB2XBIN	"B'00000010" This is batch input unit
		.... ...1		SDB2INDX	"B'00000001" Data set to be/is indexed
182	(B6)	BITSTRING	1	SDBFLGM	Miscellaneous flag byte
		1... ....		SDBMJML	"B'10000000" JESMSG LG dataset
		.1.. ....		SDBMJSM	"B'01000000" JESYSMSG dataset
182	(B6)	X'CO'	0	SDBMSJDS	"SDBMJML+SDBMJSM" Special JES2 data sets
		..1. ....		SDBMRSKP	"B'00100000" Records skipped due to I/O error on GET

Comment

SDBFLAGY flag byte is used to control instream symbol substitution.  
See PDBFLAGY for description.

End of Comment

183	(B7)	BITSTRING	1	SDBFLAGY	Symbol substitution flags for DD * and DD DATA:
184	(B8)	CHARACTER	8	SDBDDNM	DDNAME OF DATA SET
192	(C0)	SIGNED	4	SDBOPNCT	DATA SET OPEN COUNT

Comment

SDBTCBM is always job step TCB. When the job step TCB ends, the SDB is removed. SDBTCBO is the TCB that owns the memory for the SDB. That is normally the Region Control task.

End of Comment

196	(C4)	ADDRESS	4	SDBTCBM	TCB managing SDB memory
200	(C8)	ADDRESS	4	SDBTCBO	TCB owning the SDB memory
204	(CC)	ADDRESS	4	SDBHCCT	POINTER TO HASP HCCT
208	(D0)	ADDRESS	4	SDBSJB	POINTER TO SJB
212	(D4)	ADDRESS	4	SDBSDB	POINTER TO NEXT SDB OFF SJB
216	(D8)	ADDRESS	4	SDBJFCBE	POINTER TO JFCB EXTENSION
220	(DC)	ADDRESS	4	SDBPIOT	POINTER TO PDDB IOT
224	(E0)	ADDRESS	4	SDBPDDB	POINTER TO PDDB
228	(E4)	SIGNED	4	SDBPDDBA	ALET of PDDB

Comment

Related data area. SDBCBAADR is a related CB  
CB address and SDBCBALE is the ALET for the related  
CB. The data area is based on bits in SDBFLG4 or  
SDBFLG6.

End of Comment

232	(E8)	SIGNED	4	SDBCBALE	ALET for related CB
236	(EC)	ADDRESS	4	SDBCBAADR	Address of related CB

Comment

If SDB6SAPI then related CB is a SAPID

End of Comment

236	(EC)	X'E8'	0	SDBSAPAL	"SDBCBALE,4,C'F'" ALET for the SAPI data spc
236	(EC)	X'EC'	0	SDBSAPID	"SDBCBAADR,4,C'A'" Address of SAPID. The SAPID is in a data space.

# \$SDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
If SDB4IRDR or SDB4NJET related CB is the JRW in the IRWD					
End of Comment					
236	(EC)	X'EC'	0	SDBJRW	"SDBCADR,4,C'A" Address of JRW.
240	(F0)	DBL WORD	8	(0)	ALIGN SDBTAB ON DOUBLE WORD
240	(F0)	BITSTRING	1	SDBTAB	Major/minor TAB
240	(F0)	X'F8'	0	SDBAIOT	"TABAIOT-TAB+SDBTAB,,C'A" POINT TO ALLOCATION IOT
252	(FC)	ADDRESS	4	SDBDEB	Pointer to last DEB that was OPENed
256	(100)	DBL WORD	8	SDBTRKF	FIRST RBA
264	(108)	DBL WORD	8	SDBTRK	RBA OF 1ST LREC IN CURRENT UBF
272	(110)	DBL WORD	8	SDBTRKL	ENDING RBA
280	(118)	CHARACTER	8	SDBKEY (0)	RECORD VERIFICATION KEY --- (must stay as C type for HASCOFST to compile)
280	(118)	BITSTRING	4	SDBJKEY	4-BYTE UNIQUE JOB KEY
284	(11C)	BITSTRING	4	SDBDKEY	4-BYTE UNIQUE DS NO. IN JOB
288	(120)	SIGNED	8	SDBSRECN	Record number of DS start
296	(128)	BITSTRING	1	SDBJMEMN	Target job's member number
297	(129)	BITSTRING	1	SDBHPFCT	HPUTFULL call count
298	(12A)	SIGNED	2	SDBJASID	TARGET JOB'S ASID (BROWSE)
300	(12C)	CHARACTER	8	SDBRCID	8 CHAR RECVR ID FOR BROWSE
308	(134)	SIGNED	4	SDBLOGAD	ADDR OF BROWSE LOG STRING
312	(138)	BITSTRING	1	SDBFLGAS	Asynchronous flag byte (updates serialized by local lock)
		...1 ....		SDBASJBL	"B'00010000" Waiting for job buf limit
		.... 1...		SDBASBWT	"B'00001000" Waiting for buffer (data set or job limit)
313	(139)	BITSTRING	1	SDBFLG3	FLAG BYTE 3
		1... ....		SDB3NIRB	"B'10000000" Release TCBNOIRB when SDBLOCK is released
		.1... ....		SDB3SRBO	"B'01000000" Do not position to start of UBF. It was SRB obtained
		..1. ....		SDB3LINE	"B'00100000" RECORD IS LINE MODE
		...1 ....		SDB3PAGE	"B'00010000" RECORD IS PAGE DATA
		.... 1...		SDB3OUTX	"B'00001000" ABEND722 IN PROGRESS
		.... .1..		SDB3PBAD	"B'00000100" PUT was unsuccessful
		.... ..1.		SDB3PSP	"B'00000010" HPMOVE spanning record
		.... ...1		SDB3BTRC	"B'00000001" DATASET BLANKS TRUNCATED
314	(13A)	BITSTRING	1	SDBFLG4	FLAG BYTE 4
		1... ....		SDB4PSO	"B'10000000" PROCESS-SYSOUT DATA SET
		.1.. ....		SDB4SYIN	"B'01000000" SYSIN DATA SET
		..1. ....		SDB4SOUT	"B'00100000" SYSOUT DATA SET
		...1 ....		SDB4RECV	"B'00010000" DATA SET ALLOCATED FOR RECV
		.... 1...		SDB4SPBR	"B'00001000" SPOOL BROWSE DATA SET
		.... ..1.		SDB4NJET	"B'00000100" NJE/TCP data set
		.... ..1.		SDB4TNJR	"B'00000010" NJE/TCP Job Receiver
		.... ...1		SDB4IRDR	"B'00000001" Internal reader data set
315	(13B)	BITSTRING	1	SDBFLG5	FLAG BYTE 5 SERIALIZATION: NONE (DO NOT USE IN SIO OR CHANNEL END APPENDAGES)
		1... ....		SDB5ASY	"B'10000000" Asynchronous request
		.1.. ....		SDB5OUTL	"B'01000000" OUTLIM exceeded for normal PUT request
		..1. ....		SDB5SBNS	"B'00100000" Spool browse - Do not attempt SRB for this DS (no more data available)
		...1 ....		SDB5INCI	"B'00010000" Logical data set being processed-JCL, JOBLOG..
		.... 1...		SDB5ADFR	"B'00001000" Defer excession limit ABEND (set during close)
		.... ..1.		SDB5ABND	"B'00000100" ABEND for output limit excession (Never reset)
		.... ...1.		SDB5ADMP	"B'00000010" A DUMP is requested for the 722 ABEND (Never reset)
		.... ...1		SDB5ADON	"B'00000001" An ABEND 722 has been for this data set (reset if a second ABEND is needed)
316	(13C)	BITSTRING	1	SDBFLG6	Flag byte 6
		1... ....		SDB6SAPI	"B'10000000" Sysout API data set This serves as a modifier of SDB4PSO
		.1.. ....		SDB6GONE	"B'01000000" SAPIID has been freed



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		..1. ....		SDB6PRT	"B'00100000" Print data set
		...1 ....		SDB6PUN	"B'00010000" Punch data set
		.... 1...		SDB6TCL	"B'00001000" Track cell data set
		.... .1..		SDB6FTRK	"B'00000100" Full track despool data set
		.... ..1.		SDB6NSRB	"B'00000010" No SRBs for this data set
		.... ...1		SDB6UPDG	"B'00000001" Update mode GET active
317	(13D)	BITSTRING	1	SDBFLG7	Flag byte 7
		1... ....		SDB7PUT1	"B'10000000" Indicate first put is done in a segment
		.1.. ....		SDB7SPIN	"B'01000000" SPIN is required for segmentation
		..1. ....		SDB7SUPD	"B'00100000" Segmentation is suspended
		...1 ....		SDB7NULL	"B'00010000" PDB1NULL was on when data set (fake) opened
		.... 1...		SDB7JLOG	"B'00001000" Joblog data set (set only if SPOOL browse SDB)
		.... .1..		SDB7JRPL	"B'00000100" Extended JES2 mode RPL in use
		.... ..1.		SDB7KSET	"B'00000010" Callers KEY is already set in SDB (do not reset)
		.... ...1		SDB7SPC	"B'00000001" Data set spinning support via \$SPIN

Comment

SDB7PSCR indicates SCR's are to be processed as normal records. This bit is only honored if an extended RPL is passed (SDB7JRPL is on). Processing is altered as follows:  
 GET - SCR's are returned to the caller. RPLH1SCR is set if the current record is an SCR.

End of Comment

.... ..1	SDB7PSCR	"B'00000001" GET - Request SCR's returned
----------	----------	---

Comment

The flag bits in SDBFLG8 are used to document why an open failed in the HASP708 message.

End of Comment

318	(13E)	BITSTRING	1	SDBFLG8	Flag byte 8
		1... ....		SDB8TRAK	"B'10000000" Internal \$TRAK error
		.1.. ....		SDB8CBIO	"B'01000000" Internal \$CBIO error
		..1. ....		SDB8GASN	"B'00100000" \$GASSIGN error
		...1 ....		SDB8SJFR	"B'00010000" SJFREQ error
		.... .1..		SDB8GETB	"B'00000100" GETBUF failure
		.... ...1		SDB8NRA	"B'00000010" Suspend read ahead
319	(13F)	BITSTRING	1		Reserved
320	(140)	ADDRESS	4	SDBDSCA	DSCA chain pointer
324	(144)	ADDRESS	4	SDBDSCAW	Working DSCA CB address
328	(148)	ADDRESS	4	SDBDSCE	Current DSCE address
332	(14C)	ADDRESS	4	SDBCDSCE	Channel end DSCE address
336	(150)	ADDRESS	4	SDBSJIOB	SJIOB for CBIO
340	(154)	ADDRESS	4	SDBJFCB	Pointer to JFCB
344	(158)	DBL WORD	8	(0)	Ensure doubleword alignment
344	(158)		8	SDBOUTLM	SYSDOUTLIM= PARAMETER
344	(158)	X'158'	0	SDBPRECN	"SDBOUTLM,8,C'D" Prev record number (POINT)
352	(160)	ADDRESS	4	SDBWTCN	Buffer wait chain
356	(164)	SIGNED	4	SDBSGMT	NUMBER OF LOGICAL PAGES PER SEGMENT
360	(168)	SIGNED	4	SDBSEGID	SEGMENT IDENTIFIER
364	(16C)	SIGNED	4	SDBPGCT	Logical page counter use for segmentation
368	(170)	DBL WORD	8	SDBOWNER	Owning TCB information
368	(170)	X'168'	0	SDBTTOKN	"SDBOWNER-8,16" TCB Token placed here at SDB Free time; not formatted in dump
376	(178)	BITSTRING	1	SDBCPSWK	Caller PSW byte 1 (key)
377	(179)	BITSTRING	1		Reserved

## \$SDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
378	(17A)	SIGNED	1	SDBGETCT	PROTGET count (number of calls without a point)
379	(17B)	SIGNED	1	SDBSRBCT	PROTSRB count (number of SRB calls in a row)
380	(17C)	SIGNED	4	SDBSRECB	Waiter ECB
384	(180)	ADDRESS	4	SDBBAT	GET processing BAT chain
388	(184)	SIGNED	4	SDBBATCT	GET BAT count

Comment

SDBGMTTR is the next track address that needs to be read for this data set. If this field is zero, that does not imply that we are done reading the data set. It is zero if we are waiting for an I/O to complete.

An EXCPVR needs to be issued if:

- SDBGMTTR is non-zero
- SDBPBLIN is non-zero
- There is a BAT (on the SDBBAT chain) that does not have I/O active.

End of Comment

392	(188)	DBL WORD	8	SDBGMTTI (0)	Next MTTR and flags
392	(188)	SIGNED	4	SDBGMTTR	Next MTTR for GET to read
396	(18C)	BITSTRING	2		Reserved for MQTR
398	(18E)	BITSTRING	1	SDBGMFG1	Migration I/O flags
		1... ....		SDBG1MPR	"B'10000000" This MTTR processed for migration
		.1.. ....		SDBG1SOV	"B'01000000" Source override - use source DAS for I/O
399	(18F)	BITSTRING	1	SDBGMIGT	Migration transition count captured from DAS when I/O is queued
400	(190)	DBL WORD	8	SDBINPBL (0)	--+ Align for CDS/CSG
400	(190)	SIGNED	4	SDBINPBC	Count of chain updates
404	(194)	ADDRESS	4	SDBPBLIN	--+ GET inactive PBLOCK chain
408	(198)	DBL WORD	8	(0)	--+ Align for CDS
408	(198)	ADDRESS	4	SDBPBLAC	GET active PBLOCK chain
412	(19C)	ADDRESS	4	SDBPBLFL	--+ GET full PBLOCK chain
416	(1A0)	SIGNED	4	SDBPBLCT	GET PBLOCK count
420	(1A4)	SIGNED	4	SDBPBUFC	GET buffer count in PBLKs
424	(1A8)	ADDRESS	4	SDBAPBL	Current PBLOCK
428	(1AC)	ADDRESS	4	SDBAMTRE	Current MTTR entry
432	(1B0)	CHARACTER	12	SDBENQNM	SDB lock minor name

Comment

MACRO-DATE = 06/24/03

End of Comment

444	(1BC)	SIGNED	4	(0)	ESTABLISH A FULLWORD BOUNDARY
444	(1BC)	X'1BC'	0	SDBENQ	*** X02113
444	(1BC)	ADDRESS	1		PELLAST flag byte. X02113
445	(1BD)	ADDRESS	1		PELMILEN - RNAME length.
446	(1BE)	BITSTRING	1		

Comment

PELFLAG - flag byte 2.

End of Comment

447	(1BF)	ADDRESS	1		PELRET - return code byte.
448	(1C0)	ADDRESS	4		QNAME ADDRESS
452	(1C4)	ADDRESS	4		RNAME ADDRESS
452	(1C4)	X'C'	0	SDBENQL	**SDBENQ"
456	(1C8)	ADDRESS	4	SDBUSER1	RESERVED FOR USER
460	(1CC)	ADDRESS	4	SDBUSER2	RESERVED FOR USER

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
464	(1D0)	SIGNED	4	SDBISTDS	Current instream data set number (initialized to 1)
Comment					
BUFFER POINTERS USED BY HAM (USE DEPENDENT ON TYPE OF I/O REQUEST BEING PROCESSED)					
End of Comment					
468	(1D4)	ADDRESS	4	SDBUBF	Input - Unprotected buffer
472	(1D8)	ADDRESS	4	SDBPBF	Protected buffer address
476	(1DC)	ADDRESS	4	SDBSCDR	SPOOL data record in PBF
480	(1E0)	ADDRESS	4	SDBPBFS	PBUF save area (Put update)
484	(1E4)	ADDRESS	4	SDBSCDRS	SCDR save area (Put update)
488	(1E8)	SIGNED	4	SDBPBFCS (0)	----+ Keep next fields together
488	(1E8)	SIGNED	2	SDBPBFAC	PUT - PBF active count
490	(1EA)	SIGNED	2	SDBPBFCT	----+ PUT - PBF obtained buffers
492	(1EC)	BITSTRING	1	SDBASKEY	Async requestor's key
493	(1ED)	BITSTRING	1	SDBASREQ	Async last request type
494	(1EE)	SIGNED	2	SDBPBFM	PUT - PBF buffer limit
496	(1F0)	ADDRESS	4	SDBPBFIN	PUT - PBF on it's way from SDBPBF to PBUF chain
500	(1F4)	ADDRESS	4	SDBDSIX	DSIX pointer
504	(1F8)	ADDRESS	4	SDBCDSXE	Current DSXE pointer
508	(1FC)	ADDRESS	4	SDBYSUBF	Buffer used for instream symbol substitution
512	(200)	SIGNED	8	SDBRECN	Current record number

Comment

SDBRPL is used for asynch, locate mode requests

End of Comment					
520	(208)	ADDRESS	4	SDBRPL	Pointer to active RPL chain
524	(20C)	SIGNED	4	SDBXCPCT	PHYSICAL BUFFER I/O COUNT
528	(210)	DBL WORD	8	SDBUPRBA	RBA FOR GET/PUT-UPDATE & POINT
536	(218)	DBL WORD	8	SDBURBAS	POINT RBA save area
544	(220)	BITSTRING	6	SDBPTIME	Time stamp for POINT
550	(226)	BITSTRING	1	SDBFLGP	POINT flag byte
		1... ....		SDBPPNT	"B'10000000" Point operation active
		.1. ....		SDBPRTOK	"B'01000000" Point operation retryable
		..1. ....		SDBPRTRY	"B'00100000" Point retry not attempted
		...1 ....		SDBPTIMN	"B'00010000" Point by next time
		.... 1..		SDBPTIMP	"B'00001000" Point by previous time
551	(227)	BITSTRING	1		Reserved
552	(228)	SIGNED	4	SDBHFRCT	HFINDRBA residual count
556	(22C)	SIGNED	4	SDBMTTR	MTTR OF NEXT BLOCK
560	(230)	ADDRESS	4	SDBASECB	Address of asynch ECB
564	(234)	SIGNED	4	SDBNBLK	NUMBER OF DATA BLOCKS READ
568	(238)	SIGNED	4	SDBBFECB	WAIT-BUF ECB FOR SVCPUTS
572	(23C)	ADDRESS	4	SDBYLGC	Ptr to logging YLGC
576	(240)	DBL WORD	8	SDBDWORK	Doubleword work area
584	(248)	DBL WORD	8	SDBDWRK2	Doubleword work area
584	(248)	X'240'	0	SDBWRK16	"SDBDWORK,16" 16 byte work area
592	(250)	BITSTRING	7	SDBSCDRQ	SCDR work area
599	(257)	BITSTRING	1		Reserved
600	(258)	BITSTRING	6	SDBIOTMQ	MQTR of PDDB IOT
606	(25E)	BITSTRING	2		Reserved

Comment

-----  
Packed decimal versions of count fields  
-----

End of Comment					
608	(260)		8	SDBDRECD	Logical record count in packed decimal format

## \$SDB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
616	(268)		8	SDBDPAGE	Actual page count in packed decimal format
624	(270)		8	SDBDBYTE	Actual byte count in packed decimal format
Comment					
----- Binary versions of count fields (for Pddb) -----					
End of Comment					
632	(278)	SIGNED	4	SDBRECCT	Logical record count
636	(27C)	SIGNED	4	SDBPAGCT	Actual page count
640	(280)	SIGNED	4	SDBBYTCT	Actual byte count
644	(284)	ADDRESS	4	SDBSPC	Address of spin control
Comment					
----- Pddb pending update section. When the buffer whose address is specified is seen in the channel end appendage, the corresponding counts are updated in the Pddb. -----					
End of Comment					
648	(288)	SIGNED	4	SDBPNCNT	Buffers updates since last Pddb update
652	(28C)	BITSTRING	4	SDBPNMTT	Buffer address
656	(290)	SIGNED	4	SDBPNREC	Logical record count
660	(294)	SIGNED	4	SDBPNPAG	Actual page count
664	(298)	SIGNED	4	SDBPNBYT	Actual byte count
668	(29C)	ADDRESS	4	SDBLCKRB	RB pointer used by PGEXCPCK
Comment					
----- Diagnostic area for unwritten buffer processing. The following are input areas passed to the SRB. -----					
End of Comment					
672	(2A0)	SIGNED	4	SDBMTTRT	Target MTTR
676	(2A4)	ADDRESS	4	SDBFSDB	SDB address given to SRB
680	(2A8)	DBL WORD	8	SDBBKEY	Unique buffer key
688	(2B0)	ADDRESS	4	SDBSTKN	STOKEN given to IEAMSCHD
Comment					
----- The following are output fields from the SRB. -----					
End of Comment					
692	(2B4)	SIGNED	4	SDBRETC	Return code from PROTSRB
696	(2B8)	BITSTRING	0	SDBSRBOA (0)	Output area from HASCUBSR. Keep fields SDBSDBA thru SDBBPTFA together.
696	(2B8)	ADDRESS	4	SDBSDBA	---+ A(SDB) that matches the key
700	(2BC)	ADDRESS	4	SDBAPBF	SDBPBF
704	(2C0)	ADDRESS	4	SDBPBFI	Inflight PBF buffer address
708	(2C4)	ADDRESS	4	SDBBPTR	Address of found buffer
712	(2C8)	SIGNED	4	SDBBTRK	MTTR of found buffer
716	(2CC)	ADDRESS	4	SDBBADBF	Address of invalid buffer
720	(2D0)	ADDRESS	4	SDBADSDB	Address of invalid SDB
724	(2D4)	ADDRESS	4	SDBBPBF	BATPBF if checking this Q
728	(2D8)	ADDRESS	4	SDBBPTFA	---+ BATPBFA if checking this Q

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
728	(2D8)	X'24'	0	SDBSOLEN	**SDBSDBA" L'(PROTSRB) diagnostic area
Comment					
-----					
ASOK fields.					
-----					
End of Comment					
732	(2DC)	BITSTRING	8	SDBASOK (0)	ASOK fields
734	(2DE)	SIGNED	2	SDBASOKO	Ordinality of ASOK L1
736	(2E0)	SIGNED	2	SDBASOK1	Offset into Level 1 ASOK
738	(2E2)	SIGNED	2	SDBASOK2	Offset into Level 2 ASOK
Comment					
-----					
Reserved space					
-----					
End of Comment					
740	(2E4)	SIGNED	4		Reserved for future use
740	(2E4)	X'300'	0	SDBLNG	"((-SDB+127)/128)*128" Length of DSECT

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	ASOK	'
0	(0)	CHARACTER	4	ASOEYE	Eye catcher ASOK level 1
4	(4)	ADDRESS	4	ASONEXT	Addr of next level 1 ASOK
8	(8)	ADDRESS	4	ASOPTR (0)	Addresses of level 2 ASOKs
8	(8)	X'4'	0	ASOELEN1	"4" Length of one entry ... caution! Length must be a power of 2
8	(8)	X'3FE'	0	ASONRL1	"1022" Number of L1 entries
8	(8)	X'1000'	0	ASOLENL1	"8+ASOELEN1*ASONRL1" Length of L1 ASOKs
Comment					
-----					
DC C'ASO2' Eye catcher level 2 ASOK					
-----					
End of Comment					
4	(4)	ADDRESS	4	ASOCOUNT	Count of available entries
8	(8)	ADDRESS	4	ASOSDB (0)	Addresses of SDBs
8	(8)	X'8'	0	ASOELEN2	"8" Length of one entry ... caution! Length must be a power of 2
8	(8)	X'1FF'	0	ASONRL2	"511" Number of L2 entries
8	(8)	X'1000'	0	ASOLENL2	"8+ASOELEN2*ASONRL2" Length of L2 ASOKs
8	(8)	X'8'	0	ASOIRWD	"ASOSDB,4,C'A" Addresses of IRWDs

## \$SDB Cross Reference

## \$SDB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
ASOCOUNT	4		SDBFLG2	B5	
ASOELEN1	8	4	SDBFLG3	139	
ASOELEN2	8	8	SDBFLG4	13A	
ASOEYE	0	C1E2D6F1	SDBFLG5	13B	
ASOIRWD	8	8	SDBFLG6	13C	
ASOK	0		SDBFLG7	13D	
ASOLENL1	8	1000	SDBFLG8	13E	
ASOLENL2	8	1000	SDBFSDB	2A4	
ASONEXT	4		SDBGETCT	17A	
ASONRL1	8	3FE	SDBGMFG1	18E	
ASONRL2	8	1FF	SDBGMIGT	18F	
ASOPTR	8		SDBGMTTI	188	
ASOSDB	8		SDBGMTTR	188	
SDB	0		SDBG1MPR	18E	80
SDBADSDB	2D0		SDBG1SOV	18E	40
SDBAIOT	F0	F8	SDBHCCT	CC	
SDBAMTRE	1AC		SDBHFRCT	228	
SDBAPBF	2BC		SDBHPFCT	129	
SDBAPBL	1A8		SDBID	A8	
SDBASBWT	138	8	SDBINPBC	190	
SDBASECB	230		SDBINPBL	190	
SDBASJBL	138	10	SDBIOTMQ	258	
SDBASKEY	1EC		SDBISTDS	1D0	
SDBASOK	2DC		SDBJASID	12A	
SDBASOKO	2DE		SDBJFCB	154	
SDBASOK1	2E0		SDBJFCBE	D8	
SDBASOK2	2E2		SDBJKEY	118	
SDBASREQ	1ED		SDBJMEMN	128	
SDBBADBF	2CC		SDBJRW	EC	EC
SDBBAT	180		SDBKEY	118	
SDBBATCT	184		SDBLCKRB	29C	
SDBBFECB	238		SDBLENG	AC	
SDBBKKEY	2A8		SDBLNG	2E4	300
SDBBPBF	2D4		SDBLOGAD	134	
SDBBPTFA	2D8		SDBMJML	B6	80
SDBBPTR	2C4		SDBMJSM	B6	40
SDBBTRK	2C8		SDBMRSKP	B6	20
SDBBYTCT	280		SDBMSJDS	B6	C0
SDBCBAADR	EC		SDBMTTR	22C	
SDBCBALE	E8		SDBMTTRT	2A0	
SDBCDSCE	14C		SDBNBLK	234	
SDBCDSXE	1F8		SDBOPNCT	C0	
SDBCPSWK	178		SDBOUTLM	158	
SDBDBYTE	270		SDBOWNER	170	
SDBDDNM	B8		SDBPAGCT	27C	
SDBDEB	FC		SDBPBF	1D8	
SDBDKEY	11C		SDBPBFAC	1E8	
SDBDPAGE	268		SDBPBFCS	1E8	
SDBDRECD	260		SDBPBFCT	1EA	
SDBDSCA	140		SDBPBFI	2C0	
SDBDSCAW	144		SDBPBFIN	1F0	
SDBDSCE	148		SDBPBFILM	1EE	
SDBDSIX	1F4		SDBPBFIS	1E0	
SDBDWORK	240		SDBPBLAC	198	
SDBDWRK2	248		SDBPBLCT	1A0	
SDBENQ	1BC	1BC	SDBPBLFL	19C	
SDBENQL	1C4	C	SDBPBLIN	194	
SDBENQNM	1B0	E2C4C24B	SDBPBUFC	1A4	
SDBFLAGY	B7		SDBPDDDB	E0	
SDBFLGAS	138		SDBPDDBA	E4	
SDBFLGM	B6		SDBPGCT	16C	
SDBFLGP	226		SDBPIOT	DC	
SDBFLG1	B4		SDBPNBYT	298	

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SDBPNCNT	288		SDB2IOE	B5	80
SDBPNMTT	28C		SDB2VAL	B5	40
SDBPNPAG	294		SDB2VDK	B5	20
SDBPNREC	290		SDB2XBIN	B5	2
SDBPPNT	226	80	SDB3BTRC	139	1
SDBPRECN	158	158	SDB3LINE	139	20
SDBPR TOK	226	40	SDB3NIRB	139	80
SDBPRTRY	226	20	SDB3OUTX	139	8
SDBPTIME	220		SDB3PAGE	139	10
SDBPTIMN	226	10	SDB3PBAD	139	4
SDBPTIMP	226	8	SDB3PSP	139	2
SDBRCID	12C		SDB3SRBO	139	40
SDBRECCT	278		SDB4IRDR	13A	1
SDBRECN	200		SDB4NJET	13A	4
SDBRETC	2B4		SDB4PSO	13A	80
SDBRPL	208		SDB4RECV	13A	10
SDBR14SV	B0		SDB4SOUT	13A	20
SDBSAPAL	EC	E8	SDB4SPBR	13A	8
SDBSAPID	EC	EC	SDB4SYIN	13A	40
SDBSAVE	0		SDB4TNJR	13A	2
SDBSCDR	1DC		SDB5ABND	13B	4
SDBSCDRQ	250		SDB5ADFR	13B	8
SDBSCDRS	1E4		SDB5ADMP	13B	2
SDBSDB	D4		SDB5ADON	13B	1
SDBSDBA	2B8		SDB5ASY	13B	80
SDBSEGID	168		SDB5INCI	13B	10
SDBSGMT	164		SDB5OUTL	13B	40
SDBSJB	D0		SDB5SBNS	13B	20
SDBSJIOB	150		SDB6FTRK	13C	4
SDBSOLEN	2D8	24	SDB6GONE	13C	40
SDBSPC	284		SDB6NSRB	13C	2
SDBSRBCT	17B		SDB6PRT	13C	20
SDBSRBOA	2B8		SDB6PUN	13C	10
SDBSRECB	17C		SDB6SAPI	13C	80
SDBSRECN	120		SDB6TCL	13C	8
SDBSTKN	2B0		SDB6UPDG	13C	1
SDBTAB	F0		SDB7JLOG	13D	8
SDBTCBM	C4		SDB7JRPL	13D	4
SDBTCBO	C8		SDB7KSET	13D	2
SDBTRK	108		SDB7NULL	13D	10
SDBTRKF	100		SDB7PSCR	13D	1
SDBTRKL	110		SDB7PUT1	13D	80
SDBTTOKN	170	168	SDB7SPC	13D	1
SDBUBF	1D4		SDB7SPIN	13D	40
SDBUPRBA	210		SDB7SUPD	13D	20
SDBURBAS	218		SDB8CBIO	13E	40
SDBUSER1	1C8		SDB8GASN	13E	20
SDBUSER2	1CC		SDB8GETB	13E	4
SDBWRK16	248	240	SDB8NRA	13E	2
SDBWTCHN	160		SDB8SJFR	13E	10
SDBXCPCCT	20C		SDB8TRAK	13E	80
SDBYLGC	23C				
SDBYSUBF	1FC				
SDB1CLOS	B4	4			
SDB1ENDR	B4	2			
SDB1ENQ	B4	40			
SDB1FOPN	B4	1			
SDB1GET	B4	80			
SDB1HPUT	B4	10			
SDB1OUT	B4	8			
SDB1PUT	B4	20			
SDB2DSRS	B5	4			
SDB2EOD	B5	10			
SDB2GLM	B5	8			
SDB2INDX	B5	1			





---

## **\$SFRB Information**

### **\$SFRB Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SFRB**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$SFRB Map

### \$SFRB Heading Information

**Common Name:** Scheduler Facility Request Block  
**Macro ID:** \$SFRB  
**DSECT Name:** SFRB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SFRB'  
 Offset: SFRBID-SFRB  
 Length: 4  
**Storage Attributes:** Subpool: 231 (ECSA)  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M) in common storage.  
**Size:** SFRHSZE - Header size  
 SFRMRSZ - Size of Modify function area  
**Created by:** Scheduler JCL Facility Services (routine SSISFS)  
**Pointed to by:** CCTSFREQ field of the \$HCCT data area  
 CCTSFPNQ field of the \$HCCT data area  
 CCTSFPRQ field of the \$HCCT data area  
 SFRBNXT field of the \$SFRB data area  
 SFRBLIFO field of the \$SFRB data area  
 TRESFRB field of the \$TRE data area  
 SSWSFRB field of the \$SFSWORK data area  
**Serialization:** Use of separate queues (Request/Pending/Processing).  
 Use of CDS to serialize the use of the Request queue.  
**Function:** This macro provides the mapping of the request block used as an interface between the Scheduler Facility Services SSI and PCE Processor.  
 This request block will be on one of three queues as noted above under 'POINTED TO BY'.

### \$SFRB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SFRB	
0	(0)	X'0'	0	SFRBGN	***
0	(0)	CHARACTER	4	SFRBID	Acronym set to 'SFRB'
4	(4)	ADDRESS	1	SFRBVER	Version number of SFRB
		.... ...1		SFRBCVR	"X'01" Current version no.of SFRB
5	(5)	BITSTRING	1	SFRBRSV1	Reserved
6	(6)	SIGNED	2	SFRBRSV2	Reserved
8	(8)	SIGNED	4	SFRBNXT	Next SFRB block
12	(C)	SIGNED	4	SFRBLIFO	Next SFRB (in LIFO order)
Comment					
Flag byte input from caller to function rtn Bits defined in function dependent area					
End of Comment					
16	(10)	BITSTRING	1	SFRFFLG	Function Request Flag
Comment					
Indicator byte for \$BLDMSG processing of msgs					
End of Comment					
17	(11)	BITSTRING	1	SFRBMSGI	Indicator byte for \$BLDMSG

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Flag byte to footprint processing of block					
End of Comment					
18	(12)	BITSTRING	1	SFRFLG1	Flag Byte
		1... ....		SFR1PROC	"B'10000000" Process Indicator
		.1.. ....		SFR1SUBT	"B'01000000" Block given to subtask
		..1. ....		SFR1ERR	"B'00100000" Error occurred
		...1 ....		SFR1JBLK	"B'00010000" Job lock acquired
		.... 1..		SFR1JOEB	"B'00001000" JOE made busy
		.... .1..		SFR1MSGP	"B'00000100" Awaiting resources msg sent
Comment					
Status byte to indicate status of request					
End of Comment					
19	(13)	BITSTRING	1	SFRSTAT	Status Byte
		.1.. ....		SFRSFINI	"B'01000000" Processing Complete
		..1. ....		SFRSSCOM	"B'00100000" Subtask completed block
20	(14)	SIGNED	4	SFRRC	Return code for subtask
24	(18)	SIGNED	4	SFRCKTKN	Checkpoint token
28	(1C)	ADDRESS	4	SFRSQD	SQD pointer
32	(20)	ADDRESS	4	SFRTOKN	Address of UTOKEN
36	(24)	CHARACTER	8	SFRRJABI	Requestor jobid
44	(2C)	CHARACTER	8	SFRRJABN	Requestor jobname
52	(34)	CHARACTER	8	SFRTIME (0)	Time Stamp of request
52	(34)	SIGNED	4	SFRCTME	Significant part of time
56	(38)	SIGNED	4	SFRISTM	INSIGNIFICANT PART OF TIME
60	(3C)	CHARACTER	8	SFRFTIM	Time Stamp of GETLOK failure
60	(3C)	X'44'	0	SFRHSZE	"*-SFRBGN" Header size
Comment					
Specific function request data area begins here					
End of Comment					
60	(3C)	X'44'	0	SFRBFOR	*** Functional area origin
Comment					
Specific function request bit definitions for Flag byte SFRFFLG Bit definitions should correspond to input flag SSSFFLG1 in macro IAZSSSF and in X045FLG1 in \$XPL.					
End of Comment					
		1... ....		SFRFDES	"B'10000000" Destination Check Request
		.1.. ....		SFRFSECL	"B'01000000" Seclabel check request
		..1. ....		SFRFJSSP	"B'00100000" JESSPOOL check (default)
Comment					
Modify request data					
End of Comment					
68	(44)	SIGNED	2	SFRMREA	Reason code for Modify rtn
70	(46)	SIGNED	2		Reserved
72	(48)	SIGNED	4	SFRJBNUM	Converted job number
76	(4C)	SIGNED	4	SFRDOP	ODPARM pointer
80	(50)	SIGNED	4	SFRJOEP	JOE pointer

## \$SFRB Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
The following area corresponds to modify request area in the SSOB extension IAZSSSF					
End of Comment					
84	(54)	CHARACTER	8	SFRJBNM	JOBNAME
92	(5C)	CHARACTER	8	SFRJBID	JOBID
100	(64)	CHARACTER	8	SFRGRP1	Output group name
108	(6C)	SIGNED	2	SFRGRP1	Output group - first ID
110	(6E)	SIGNED	2	SFRGRP2	Output group - second ID
112	(70)	SIGNED	4	SFRMRSV2	Reserved
116	(74)	CHARACTER	8	SFRCART	CART for WTO responses
124	(7C)	SIGNED	4	SFRCNID	Console ID for WTO responses Output descriptor lists are SWBTU/TU format as required SCHEDULER JCL facility (SJF)
128	(80)	ADDRESS	4	SFRMDAD	Address of Modify list in SWBTU format
132	(84)	ADDRESS	4	SFRERAD	Address of Erase list in TU format
136	(88)	SIGNED	2	SFRMDLN	Length of Modify list(SWBTU)
138	(8A)	SIGNED	2	SFRERLN	Length of Erase list (TU)
144	(90)	DBL WORD	8	(0)	Alignment
144	(90)	X'4C'	0	SFRMRSZ	**SFRMOD" Size of modify function area

## \$SFRB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SFRB	0		SFRODP	4C	
SFRBCVR	4	1	SFRRC	14	
SFRBFOR	3C	44	SFRRJOBI	24	
SFRBGN	0	0	SFRRJOBN	2C	
SFRBID	0	E2C6D9C2	SFRSFINI	13	40
SFRBLIFO	C		SFRSQD	1C	
SFRBMSGI	11		SFRSSCOM	13	20
SFRBNXT	8		SFRSTAT	13	
SFRBRSV1	5		SFRTIME	34	
SFRBRSV2	6		SFRTOKN	20	
SFRBVER	4		SFR1ERR	12	20
SFRCART	74		SFR1JBLK	12	10
SFRCKTKN	18		SFR1JOEB	12	8
SFRCNID	7C		SFR1MSGP	12	4
SFRCTME	34		SFR1PROC	12	80
SFRERAD	84		SFR1SUBT	12	40
SFRERLN	8A				
SFRFDES	3C	80			
SFRFFLG	10				
SFRFJSSP	3C	20			
SFRFLG1	12				
SFRFSECL	3C	40			
SFRFTIM	3C				
SFRGRP1	64				
SFRGRP1	6C				
SFRGRP2	6E				
SFRHSZE	3C	44			
SFRISTM	38				
SFRJBID	5C				
SFRJBNM	54				
SFRJBNUM	48				
SFRJOEP	50				
SFRMDAD	80				
SFRMDLN	88				
SFRMREA	44				
SFRMRSV2	70				
SFRMRSZ	90	4C			

---

## **\$SFRWORK Information**

### **\$SFRWORK Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SFRWORK**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$SFRWORK Map

### \$SFRWORK Heading Information

**Common Name:** JES2 Scheduler Services PCE Work Area  
**Macro ID:** \$SFRWORK  
**DSECT Name:** PCE (\$SFRWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4  
**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE  
**Size:** See symbol SRWPCEWL for the length of this work area. The overall length of the PCE is stored in field PCELENG.  
**Created by:** See \$PCE  
**Pointed to by:** \$SFSPCE field of the \$HCT data area  
 See \$PCE for other pointer fields that apply to all PCE types.  
**Serialization:** Normal PCE dispatch serialization  
**Function:** The fields in this work area are used by a JES2 Scheduler Services Processor and by its support routines and exits. \$SFRWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$SFRWORK are actually part of the PCE DSECT, but only map PCEs with the value PCESFSID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$SFRWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	BITSTRING	1	SRWFLG1	Processing flag
		1... ....		SRW1ACTV	"B'10000000" PCE active indicator
		.1.. ....		SRW1RCVY	"B'01000000" Recovery situation
		..1. ....		SRW1SNXT	"B'00100000" Use SFRB NXT (FIFO) chain
313	(139)	BITSTRING	3	SRWRSV1	Reserved for IBM use
316	(13C)	SIGNED	2	SRWREQCT	Count of lost request blks
318	(13E)	SIGNED	2	SRWABEND	Count of abends
318	(13E)	X'3'	0	SRWLIMIT	"3" Reasonable limit of abends
320	(140)	SIGNED	4	(0)	
320	(140)	BITSTRING	16	SRWTQE	Timer Queue Element
336	(150)	DBL WORD	8	(0)	Alignment
336	(150)	X'18'	0	SFRPCEWL	**-"PCEWORK" LENGTH OF SCHED. SERVICE WORK AREA

**\$SFRWORK Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
PCE	0	
SFRPCEWL	150	18
SRWABEND	13E	
SRWFLG1	138	
SRWLIMIT	13E	3
SRWREQCT	13C	
SRWRSV1	139	
SRWTQE	140	
SRW1ACTV	138	80
SRW1RCVY	138	40
SRW1SNXT	138	20

## \$SFRWORK Cross Reference



## \$SFSWORK Information

### \$SFSWORK Heading Information

**Common Name:** - HASP Scheduler Facility Service SSI work area dsect.  
**Macro ID:** \$SFSWORK  
**DSECT Name:** SFSWORK  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SFSW'  
 Offset: SSWID-SFSWORK  
 Length: 4  
**Storage Attributes:** Subpool: 229  
 Key: 1  
 Residency: Virtual and Real storage are anywhere (above or below 16M) in the User address space.  
**Size:** See SSWWLEN  
**Created by:** Scheduler Services SSI  
**Pointed to by:** N/A  
**Serialization:** None  
**Function:** This DSECT provides the work area required by the JES2 Scheduler Facility Service SSI.

### \$SFSWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SFSWORK	Scheduler Service Work Area
0	(0)	CHARACTER	4	SSWID	Eyecatcher for SFSWork
4	(4)	SIGNED	4	SSWTOTL	Total length of storage acquired.
Comment					
Area of block addresses used by the routine					
End of Comment					
8	(8)	ADDRESS	4	SSWTRE	Addr of SSI TRE
12	(C)	ADDRESS	4	SSWSFRB	Addr of SFRB
16	(10)	ADDRESS	4	SSWIOT	Addr of IOT
Comment					
Storage needed for token extract					
End of Comment					
20	(14)	ADDRESS	4	SSWWAVE	Addr of the WAVE
24	(18)	ADDRESS	4	SSWTOKN	Addr of the User Token
Comment					
Parameter input for SSI 70					
End of Comment					
28	(1C)	ADDRESS	4	SSWSOB70	Addr of SSOB for SSI 70
Comment					
Process byte for various processing paths					
End of Comment					
32	(20)	BITSTRING	1	SSWFLG1	Flag byte 1 -Indicators

# \$SFSWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		SSW1EXIT	"B'01000000" Exit given control
Comment					
Condition byte used to show errors in processing					
End of Comment					
33	(21)	BITSTRING	1	SSWFLG2	Flag byte 2 -Error flags
		1... ....		SSW2PCED	"B'10000000" PCE is disabled
		.1.. ....		SSW2JESD	"B'01000000" JES is down
		..1. ....		SSW2NOXT	"B'00100000" No Extension exists
		...1 ....		SSW2EXTE	"B'00010000" Error in extension
		.... 1...		SSW2NOAU	"B'00001000" Token Extract error
		.... .1..		SSW2INVF	"B'00000100" Invalid function request
		.... ..1.		SSW2INVI	"B'00000010" Invalid input to function
		.... ...1		SSW2NOST	"B'00000001" No storage indicator
Comment					
Response byte from Exit					
End of Comment					
34	(22)	BITSTRING	1	SSWXPLR	
		1... ....		SSWXCAN	"B'10000000" Exit cancel request
		.1.. ....		SSWXSRC	"B'01000000" Exit supplied RC/reas codes
		1111 1111		SSWANY	"X'FF" Test for any bits on
Comment					
Other goodies					
End of Comment					
35	(23)	BITSTRING	1	SSWCKEY	SSI callers key, used post-exit 45 in case exit changed X045CKEY
36	(24)	BITSTRING	1	SSWJTYPE	Job type
37	(25)	BITSTRING	1		Reserved
38	(26)	SIGNED	2	SSWERCD	Processing reason code
40	(28)	SIGNED	4	SSWRC	Processing return code
44	(2C)	SIGNED	4	SSWJBNUM	Converted job number
48	(30)	BITSTRING	28	SSWIINFO	Info block for CPOOL QCELL
76	(4C)	BITSTRING	80	SSWDSTKN	Data set token
156	(9C)	BITSTRING	1	SSWJOTKN	JOE token
Comment					
<p>-----</p> <p>Copies of fields from the SSOB extension (IAZSSSF).            These are needed so that we can access data in user key storage that was passed by the caller.            Note that some IAZSSSF fields are copied directly into the \$XPL instead of appearing here, to reduce duplication of data.</p> <p>-----</p>					
End of Comment					
236	(EC)	SIGNED	4	(0)	Word align
236	(EC)	CHARACTER	4	SSWEID	Eyecatcher
240	(F0)	BITSTRING	1	SSWVER	Version
241	(F1)	BITSTRING	1	SSWREQF	Function request number
242	(F2)	SIGNED	2	SSWMREA	Error reason code
244	(F4)	SIGNED	2	SSWLEN	SSSF length
246	(F6)	SIGNED	2		Reserved
248	(F8)	ADDRESS	4	SSWMDAD	Addr of output descriptor Modify list - SWBTU format This points to SSI caller key storage.

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
252	(FC)	ADDRESS	4	SSWERAD	Addr of output descriptor Erase list in TU format This points to SSI caller key storage.
256	(100)	ADDRESS	4	SSWIJTKN	Address of JOE token provided by SSI 80
260	(104)	ADDRESS	4	SSWIDTKN	Address of dataset token provided by SSI 80
264	(108)	ADDRESS	4	SSWIFSWU	Address of SWBTU buffer
268	(10C)	ADDRESS	4	SSWIFSWT	Address of SWB token
272	(110)	SIGNED	4	SSWIWRTN	Result of merge (rtn code)
276	(114)	SIGNED	4	SSWIWRSN	Merge Reason code
280	(118)	BITSTRING	1	SSWIFLG1	Input flag byte
281	(119)	BITSTRING	1	SSWIRFLG	Returned flag byte
282	(11A)	BITSTRING	86	SSWTKMAP	Token area for tokenmap
368	(170)	DBL WORD	8	(0)	Alignment

Comment

-----  
 \$XPL for exit 45 is here.  
 -----

End of Comment

368	(170)	BITSTRING	88	SSW45XPL	XPL for exit 45
456	(1C8)	DBL WORD	8	(0)	Alignment
456	(1C8)	X'1C8'	0	SSWWLEN	**-SFSWORK" Length of SSWORK storage

### \$SFSWORK Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SFSWORK	0		SSWWLEN	1C8	1C8
SSWANY	22	FF	SSWXCAN	22	80
SSWCKEY	23		SSWXPLR	22	
SSWDSTKN	4C		SSWXSRC	22	40
SSWEID	EC		SSW1EXIT	20	40
SSWERAD	FC		SSW2EXTE	21	10
SSWERCD	26		SSW2INVF	21	4
SSWFLG1	20		SSW2INVI	21	2
SSWFLG2	21		SSW2JESD	21	40
SSWID	0		SSW2NOAU	21	8
SSWIDTKN	104		SSW2NOST	21	1
SSWIFLG1	118		SSW2NOXT	21	20
SSWIFSWT	10C		SSW2PCED	21	80
SSWIFSWU	108		SSW45XPL	170	
SSWIINFO	30				
SSWIJTKN	100				
SSWIOT	10				
SSWIRFLG	119				
SSWIWRSN	114				
SSWIWRTN	110				
SSWJBNUM	2C				
SSWJOTKN	9C				
SSWJTYPE	24				
SSWLEN	F4				
SSWMDAD	F8				
SSWMREA	F2				
SSWRC	28				
SSWREQF	F1				
SSWSFRB	C				
SSWSOB70	1C				
SSWTKMAP	11A				
SSWTOKN	18				
SSWTOTL	4				
SSWTRE	8				
SSWVER	F0				
SSWWAVE	14				

## \$SFSWORK Cross Reference

## \$SIG Information

### \$SIG Heading Information

**Common Name:** SIG  
**Macro ID:** \$SIG  
**DSECT Name:** SIG  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** "None"  
 Offset: N/A  
 Length: N/A  
**Storage Attributes:** Subpool: 10  
 Key: 1  
 Residency: Virtual is any, Real is any in JES2 address space or user address space  
**Size:** See SIGSIZE  
**Created by:** Callers of \$SIGIO  
**Pointed to by:** Parameters passed to \$SIGIO macro  
**Serialization:** No serialization  
**Function:** This is the mapping for record zero (R0) records on SPOOL. The first track of each trackgroup has a signature record placed in the data portion of R0.

### \$SIG Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SIG	,
0	(0)	BITSTRING	1	SIGFLAG1	Flags
		1... ....		SIG1UNAL	"B'10000000" Trackgroup has begun unallocation
1	(1)	BITSTRING	3	SIGJBNUM	Job number
4	(4)	SIGNED	4	SIGJBKEY	Job key
4	(4)	X'8'	0	SIGLEN	"8" Length of signature record (DASD architected)
8	(8)	ADDRESS	2	(0)	Assembly error if length of fields not 8



---

## \$SJB Information

### \$SJB Programming Interface information

Programming Interface information

#### \$SJB

The following fields are **NOT** programming interface information:

- SJBSCB
- SJBOCT
- SJBPIT
- SJBQUEUE
- SJBSTAC

End of Programming Interface information

## Heading Information

### \$SJB Heading Information

**Common Name:** Subsystem Job Block dsect  
**Macro ID:** \$SJB  
**DSECT Name:** SJB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SJB '  
Offset: SJBID-SJB  
Length: 4

**Storage Attributes:** Subpool: 241 or 230  
Key: 1  
Residency: Virtual and real storage are anywhere (above or below 16M) in common storage or private storage (in the case of internal readers and NJE/TCP devices)

**Size:** See the SJBSIZE equate.

**Created by:** SJBs are created by the \$SJBINIT service. They are built during subsystem-interface (SSI) calls for job selection by-number (for STCs and TSUs), during the first SSI call by an MVS initiator for job selection by-class, during SSI request-jobid calls and during SYSOUT dataset allocation SSI requests for short-term cross-subsystem browse applications. An SJB is also built for each JES2 address space JCL conversion subtask when it runs its first job after JES2 initialization or after an abnormal subtask end. A temporary SJB is built by the \$LOGMSG service when it puts messages into a job's joblog.



**Pointed to by:**

- The HSBSJB field of the \$HASB data area in CSA points to the first SJB for an address space. The remaining SJBs in the address space are chained off of the SJBSJB field of the \$SJB data area.
- The MVS life-of-job SSIB control block for a job step points to the SJB supporting that job step via the SSIBSUSE field. This connection does not exist for short-term and request-job-id cases.
- The SDBSJB field of the \$SDB data area for each subsystem dataset allocated under an SJB points to the SJB. The SDBs are in the address space's private storage.
- Each SJB has an extension in the address space's private storage, called the SJXB. The SJXBSJBA field of the \$SJXB data area points to the SJB.
- The TRESJBLK field of the \$TRE data area points to the SJB if the TRE represents the MVS task that has acquired the SJB lock of this SJB.
- Several queues of SJBs exist to queue and track executing jobs. Each of these uses the SJBXQCHN field of the \$SJB data area as the chain field. The anchors are in the \$HCCT data area and include CCTJPCLS (pending selection by job class, for batch), CCTJPNUM (pending selection by number, for STCs/TSUs), CCTJXCLS (executing by class), CCTJXNUM (executing by number), CCTJTERM (jobs terminating), CCTJRENQ (jobs terminating for re-execution).
- When PSO requests or dataset processing are outstanding for an SJB, the PSOSJB field of the \$PSO data area points to the SJB.
- The DCNVSJBP field of each JES2 address space JCL conversion subtask \$DTECNV data area points to the SJB.
- The PITSJB field of the \$PIT data area points to the batch job SJB being used to manage batch jobs for the initiator represented by the PIT.
- The HSUSJB field of the \$HSU data area points to the SJB during a 'HOCSETUP' service routine call for a subsystem dataset.
- The EMSSJB field of the \$DTEEOM data area
- The CCTEOMJT field of the \$HCCT
- The JRWSJB field of the \$JRW data area for internal readers and NJE/TCP job receivers (private storage SJB)
- The SRWSJB field of the \$SRW data area for NJE/TCP SYSOUT receiver (private storage SJB)
- The JTWSJB field of the \$JTW data area for NJE/TCP job transmitter (private storage SJB)
- The STWSJB field of the \$STW data area for NJE/TCP SYSOUT transmitter (private storage SJB)
- The RIDSJB field of the \$IRWD data area for internal readers (private storage SJB)

## \$SJB Map

### Serialization:

Serialization of the SJBs is done in various ways. An SJB can be locked by a task against activity by any other task in the address space using the \$SJBLOCK service. Many JES2 SSI function routines use this service to hold the SJB lock for the duration of the SSI call. The SJB queues are locked using a mechanism similar to the SJB lock, with the lock words being in the HCCT control block instead of an SJB. This lock is frequently called the Job Communications Queue lock, or JCQ lock, or sometimes the SJB queues lock. The JCQ lock is used between the JES2 main task and the tasks in the job's address space. Various SJB chain and anchor fields are serialized with these two JES2 locks, with the system LOCAL lock, and just by timing dependencies in some cases where appropriate (e.g. when an address space is unexpectedly lost). See the comments in the definition of the dsect and various fields for more information.

### Function:

The Subsystem Job Block (SJB) represents a executing unit of work, or 'job', for the JES2 subsystem. It is the main and central JES2 control block for an executing job, and contains the job identifiers, flags defining the job type and status, indications of the type of processing required or being done for the job, locking fields, etc. It is the anchor for the in-storage control blocks such as the JCT for the job, the IOTs and SDBs for subsystem datasets, the PSO and other control blocks for current requests, etc.

The main use of the SJB for the purposes stated above are in the JES2 subsystem interface (SSI) function routines, supporting services for MVS facilities such as the initiator, allocation, and data management. It is in ECSA, and also used by the JES2 main task to track active jobs, process request queues, provide status, and reconnect to executing work during a hot start after an abnormal termination. Communication is done using the SJB queue anchors and chain fields described in the 'pointed to by' section, \$\$POSTs of JES2 by the SSI routines, \$XMPOSTs of the requesting tasks running in the SSI routines by JES2 using the SJBECB field, and serialization provided normally by two JES2-defined locks (see 'serialization').

An SJB normally represents, roughly, the work running under one MVS job step task, in however many MVS tasks there are running at that job step level. This is the case for started task 'jobs' (STCs) and time sharing user 'jobs' (TSUs), running in the top job step in an address space. It is also the usual case for a batch job, running in the second job step in an address space, where the top job step is the batch initiator STC. SJBs are also built to handle other cases where a set of one or more tasks are executing work on the behalf of a normal or special job.

The types of SJBs are: STC/TSU, batch job, request jobid, cross-subsystem browse, and JCL conversion.

## \$SJB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SJB	
0	(0)	CHARACTER	4	SJBID	SJB IDENTIFIER
4	(4)	ADDRESS	1	SJBVRSN	CURRENT VERSION IN STORAGE
4	(4)	X'6'	0	SJBCURVN	"6" Current version number
5	(5)	BITSTRING	1		RESERVED FOR FUTURE USE
6	(6)	BITSTRING	1	SJBFLG1	FIRST FLAG BYTE ---
		1... ....		SJB1PI	"X'80" Stop AND drain the initiator
		.1.. ....		SJB1SJID	"X'40" SELECT JOB BY ID - SJBJOBID
		..1. ....		SJB1CRAL	"X'20" BROWSE CROSS-ALLOCATION SJB
		.... .1..		SJB1EJOB	"X'04" HASP-RESTART JOB (\$E JOB)
		.... .1.		SJB1SWBU	"X'02" Update the OUTPUT SWB
		.... ...1		SJB1WIN	"X'01" WLM managed initiator
7	(7)	BITSTRING	1	SJBFLG2	SECOND FLAG BYTE ---
		1... ....		SJB2PNIT	"X'80" STOP THE INITIATOR
		.1.. ....		SJB2EJST	"X'40" \$EJOB,STEP was processed
		..1. ....		SJB2EOM	"X'20" END-OF-MEMORY DETECTED
		...1 ....		SJB2CNCL	"X'10" CANCEL AFTER SWA CREATE
		.... 1...		SJB2CONV	"X'08" SJB CREATED FOR CONVERTER
		.... .1..		SJB2HOLD	"X'04" HOLD JOB AFTER RE-QUEUE
		.... .1.		SJB2JNL	"X'02" JOB IS JOURNALED
		.... ...1		SJB2INIT	"X'01" INITIATOR FLAG
8	(8)	ADDRESS	4	SJBSJXB	POINTER TO SJB EXTENSION
12	(C)	ADDRESS	4	SJBWAVE	POINTER TO WAVE ADDRESS
16	(10)	ADDRESS	4	SJBUSER	*** RESERVED FOR USER ***
20	(14)	ADDRESS	4	SJBSSIB	POINTER TO SSIB
24	(18)	ADDRESS	4	SJBSJB	SJB CHAIN FROM CCTHAVT
28	(1C)	ADDRESS	4	SJBSDB	POINTER TO CHAIN OF SDBS
32	(20)	ADDRESS	4	SJBJKEY	HDBDSKEY FOR THIS JOB
36	(24)	ADDRESS	4	SJBJCT	ADDRESS OF JCT FOR JOB
40	(28)	BITSTRING	6	SJBJCTTK	JCT TRACK ADDRESS (MQTR)
46	(2E)	BITSTRING	2		Reserved
48	(30)	ADDRESS	4	SJBTCBP	ADDRESS OF INIT OR STC TCB
52	(34)	BITSTRING	16	SJBTCBT	TCB token for INIT or STC
68	(44)	SIGNED	4	SJBJQOFF	OFFSET OF JQE WITHIN JOB QUEUE
72	(48)	CHARACTER	4	SJBSSNM	SUBSYSTEM ID OF OWNER
76	(4C)	ADDRESS	4	SJBIOT	ADR OF FIRST REGULAR IOT
80	(50)	ADDRESS	4	SJBSPIOT	ADR OF FIRST SPIN IOT
84	(54)	ADDRESS	4	SJBOCT	ADR OF OUTPUT CONTROL TABLE
88	(58)	ADDRESS	4	SJBSJPTR	ADR OF SJF PARAMETER LIST
92	(5C)	ADDRESS	4	SJB SWBUF	ADR OF SJF SWB BUFFER
96	(60)	ADDRESS	4	SJBSECB (0)	STOP-ECB ADR FOR CREATED-ID JOB
96	(60)	ADDRESS	4	SJBPIT	ADDRESS OF PIT IN HASP
100	(64)	ADDRESS	4	SJBASCBP	ASCB address

Comment

SJBCKID IS USED BEFORE EXECUTION ONLY  
SJBSTQE IS USED DURING EXECUTION

End of Comment

104	(68)	SIGNED	4	SJBCKID	EXECUTION PCE CKPT TOKEN
108	(6C)	BITSTRING	1	SJBSTQE	EXECUTION TIMER QUEUE ELEMENT

Comment

THE ESTIMATED COUNT FIELDS MUST BE KEPT TOGETHER AND ARE  
MAPPED BY THE EST DSECT GENERATED BY THE \$EST MACRO

End of Comment

120	(78)	SIGNED	4	SJBTIMX (0)	---+ Time excession fields
120	(78)	SIGNED	4	SJBTMINT	Est time message interval
124	(7C)	BITSTRING	1	SJBTIMOP	Execution time option
125	(7D)	BITSTRING	3		Reserved (part of \$EST)
128	(80)	SIGNED	4	SJBXSTIM	---+ Time estimate excession amnt

# \$SJB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
END OF ESTIMATED COUNT FIELDS					
End of Comment					
132	(84)	SIGNED	4	SJB EOMCC	EOM comp code (ASCBMCC)
136	(88)	SIGNED	4	SJBFLGEF (0)	-----+ SJBFLGE fullword for CS
136	(88)	BITSTRING	3		Reserved
139	(8B)	BITSTRING	1	SJBFLGE	\$C JQ processing flags
		1... ....		SJBEJSAC	"B'10000000"   Job select active
		.1.. ....		SJBEJSCN	"B'01000000"   \$C in progress for job
		..1. ....		SJBEJSDM	"B'00100000"   \$C requested a dump
		...1 ....		SJBEVICT	"B'00010000"   Evict job at next step
		.... 1..		SJBEVICH	"B'00001000" ---+ Hold job after evicting
140	(8C)	SIGNED	4	SJBJBSEL	STCK job given to job select
144	(90)	BITSTRING	4	SJBMAXRC (0)	Completion information See JQXMAXCC for values
144	(90)	BITSTRING	1	SJBMXIND	Completion type indicator
145	(91)	BITSTRING	3	SJBMAXCC	Completion/ABEND code info
148	(94)	ADDRESS	4	SJBCSCB	ADDRESS OF CSCB
152	(98)	BITSTRING	12	SJBECBL (0)	ECB list for WAIT
152	(98)	ADDRESS	4	SJBECBA	Pointer to SJB's ECB
156	(9C)	ADDRESS	4	SJBECBS	Pointer to STOP INIT ECB
160	(A0)	ADDRESS	4	SJBECBW	Ptr to WLM STOP INIT ECB
		1... ....		SJBEEND	"X'80" To initialize end of list
168	(A8)	DBL WORD	8	(0)	Insure that SJBASCBA is on doubleword boundary for CDS
168	(A8)	BITSTRING	16	SJBXMPL (0)	\$XMPOST parameter list
168	(A8)	ADDRESS	4	SJBERRET	Pointer to error return
172	(AC)	ADDRESS	4	SJBECBP	POINTER TO SJB'S ECB
Comment					
<p>-----</p> <p>During end of memory, SJBASCBA and SJBSASCB will be HASP's ASCB (so that any POSTs will be directed to the JES2 EOM subtask). SJBASID will remain unchanged. SJBASCBP remains as a pointer to the original ASCB.</p> <p>-----</p>					
End of Comment					
176	(B0)	ADDRESS	4	SJBASCBA	ASCB address of AS to post
180	(B4)	ADDRESS	4	SJBECB	ECB for SSI code and JES2
184	(B8)	DBL WORD	8	(0)	Ensure that SJBSASCB is on doubleword boundary for CDS
184	(B8)	BITSTRING	16	SJBXMPL (0)	\$XMPOST parameter list
184	(B8)	ADDRESS	4	SJBSERRE	Pointer to error return
188	(BC)	ADDRESS	4	SJBSECBP	Pointer to SJBs ECB
192	(C0)	ADDRESS	4	SJBSASCB	Pointer to related ASCB
196	(C4)	ADDRESS	4	SJBSECBS	ECB for SSI code and JES2
200	(C8)	CHARACTER	4	SJBPATID	EBCDIC init ID (PITPATID)
204	(CC)	BITSTRING	1	SJBPRI0	HASP EXECUTION SELECTION PRTY
205	(CD)	SIGNED	3	SJBFAMILY	Highest family ID used by MOCA IOTs
208	(D0)	SIGNED	2	SJBXQFN1	HASPXEQ FUNCTION INDICATOR
210	(D2)	BITSTRING	1	SJBFLG3	TERMINATION FLAG ONE ---
		1... ....		SJB3CLS	"X'80" CLOSE ALL SUBSYSTEM DATA SETS
		.1.. ....		SJB3FSDB	"X'40" FREE ALL SDBS
		..1. ....		SJB3TERM	"X'20" TERMINATE THE JOB
		...1 ....		SJB3PPOU	"X'10" PURGE PARTIAL OUTPUT
		.... .1..		SJB3CKPT	"X'04" WRITE IOTS, JCT
		.... ..1.		SJB3FIOT	"X'02" FREE ALL IOTS
		.... ...1		SJB3FJCT	"X'01" FREE JCT
211	(D3)	BITSTRING	1	SJBFLG4	TERMINATION FLAG TWO ---
		1... ....		SJB4MEND	"X'80" MSG 'ENDED'
		.1.. ....		SJB4MTRM	"X'40" MSG 'TERMINATED'

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		..1. ....		SJB4MREQ	"X'20" MSG 'RE-ENQUEUED'
		...1 ....		SJB4MREX	"X'10" MSG 'QUEUED FOR RE-EXECUTION'
		.... 1...		SJB4FSJB	"X'08" FREE THE SJB
		.... .1..		SJB4MRQH	"X'04" MSG 'RE-ENQUEUED AND HELD'
		.... .1.		SJB4OCAN	"X'02" Operator cancelled this SJB
		.... ...1		SJB4TERM	"X'01" BATCH JOB HAS TERMINATED SJB4TERM DIRECTLY INFLUENCES THE CREATION AND DELETION OF THE JSAB
212	(D4)	ADDRESS	4	SJBQUEUE	ADDRESS OF CCTJ QUEUE HEADER
216	(D8)	ADDRESS	4	SJBXQCHN	HASPXEQ CHAINING WORD

Comment

Control information for EOM processing

End of Comment					
220	(DC)	BITSTRING	8	SJB EOMCH (0)	<-----+ EOM chaining fields
220	(DC)	ADDRESS	4	SJB EOMN	Addr next SJB on EOM queue
224	(E0)	ADDRESS	4	SJB EOMP	<-----+ Addr prior SJB on EOM queue
228	(E4)	ADDRESS	4	SJB EOMPC	PCE processing SJB
232	(E8)	BITSTRING	1	SJB EOMFL	End of memory switches
		1... ....		SJB EOMF1	"B'10000000" SJB being processed by PCE
		.1... ....		SJB EOMF2	"B'01000000" SJB being processed by EOM sub-task
233	(E9)	BITSTRING	3	SJB EOMRS	Reserved for future EOM use
236	(EC)	ADDRESS	4	SJB EOMQ	Addr of Queue head at EOM SSI time

Comment

End of EOM data

End of Comment					
240	(F0)	SIGNED	2	SJB INTCT	COUNT OF OPEN INTRDRS
242	(F2)	BITSTRING	1	SJB RSNCD	Reason Job not selectable

Comment

Return codes/ reason codes for non selection of job  
 Used for message HASP361 based on return code from \$DMNDJOB routine. Also used for message HASP003 RC(109)

End of Comment					
242	(F2)	X'0'	0	SJB RSN00	"0" No message needed
242	(F2)	X'4'	0	SJB RSN01	"4" Job not found
242	(F2)	X'8'	0	SJB RSN02	"8" System draining
242	(F2)	X'C'	0	SJB RSN03	"12" Job with same name running
242	(F2)	X'10'	0	SJB RSN04	"16" Sched. env. not available
242	(F2)	X'14'	0	SJB RSN05	"20" Independent mode mismatch
242	(F2)	X'18'	0	SJB RSN06	"24" Spools(s) not available
242	(F2)	X'1C'	0	SJB RSN07	"28" Rejected by exit 49
242	(F2)	X'20'	0	SJB RSN08	"32" SECLABEL not available
242	(F2)	X'24'	0	SJB RSN09	"36" No affinity to active sys
242	(F2)	X'28'	0	SJB RSN10	"40" Unexpected WLM response
242	(F2)	X'2C'	0	SJB RSN11	"44" ARM restart pending
242	(F2)	X'30'	0	SJB RSN12	"48" Busy
242	(F2)	X'34'	0	SJB RSN13	"52" Not batch job
242	(F2)	X'38'	0	SJB RSN14	"56" Not on execution queue
242	(F2)	X'3C'	0	SJB RSN15	"60" No JES2 that can select
242	(F2)	X'40'	0	SJB RSN16	"64" Jobclass no affinity to active member

# \$SJB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
242	(F2)	X'44'	0	SJBRSN17	"68" Job requires a higher z/OS level
243	(F3)	BITSTRING	5		RESERVED
248	(F8)	DBL WORD	8	SJBLOCKH (0)	SJB LOCKING DOUBLEWORD
248	(F8)	ADDRESS	4	SJBTCB	LOCK-HOLDING TCB ADDRESS
252	(FC)	ADDRESS	4	SJBNEXTL	0 - SJB IS UNLOCKED - - SJB LOCKED, NO WAIT CHAIN + - SJB LOCKED, ADR OF WAITER
256	(100)	ADDRESS	4	SJBTINA	Address of TINA
260	(104)	SIGNED	4	SJBTINAA	ALET of TINA (zero only if SJB for converter)
264	(108)	ADDRESS	4	SJBPSO	Process Sysout Block
268	(10C)	SIGNED	4	SJBPSOA	Process Sysout Block ALET
272	(110)	ADDRESS	4	SJBSTAC	Addr of Status/Cancel Block
276	(114)	SIGNED	4	SJBSTACA	Status/Cancel Block ALET
280	(118)	CHARACTER	1		Reserved
281	(119)	BITSTRING	1	SJBLKFG	SERIALIZATION FLAG
		1... ..		SJBFIRST	"X'80" First CCTJPCLS request for an initiator; Only meaningful in batch job SJBs.
Comment					
EQU X'40' Reserved for future use					
End of Comment					
		...1 .....		SJBPUKCEL	"X'20" Job may have unprocessed SPIN output
		...1 .....		SJBTKCEL	"X'10" SYSOUT MUST BE TRACKCELLED
		.... 1...		SJBTPST	"X'08" SJB HAS BEEN POSTED TO TERM
		.... ..1.		SJBLCKPT	"X'02" PARTIALLY SELECTED \$SJB
		.... ...1		SJBMSWBP	"X'01" NEW PDDB FOR MULTI SWBS
282	(11A)	SIGNED	2	SJBASID	USERS ASID
284	(11C)	BITSTRING	1	SJBFLG5	JOB RELATED FLAG BYTE
Comment					
THE FOLLOWING JOB TYPE FLAGS ARE IDENTICAL WITH THE JOB TYPE FLAGS IN THE JQE (I.E. JQE3JOB, JQE3STC AND JQE3TSU)					
End of Comment					
		.... ..11		SJB5JOB	"B'00000011" BATCH JOB (WHEN BITS ZERO)
		.... ..1		SJB5STC	"B'00000001" FLAG FOR THE STC JOB
		.... ..1.		SJB5TSU	"B'00000010" FLAG FOR THE TSU JOB
		.... ..1..		SJB5REST	"B'00000100" ALLOW \$EJ RESTART TO XEQ BIT ON INDICATES RESTART=Y
		.... 1...		SJB5SWAC	"B'00001000" SWA CREATED
		1... ....		SJB5JL	"B'10000000" JESDS processing Job log
		..1. ....		SJB5JCI	"B'01000000" JESDS processing JCL images
		..1. ....		SJB5MSG	"B'00100000" JESDS processing SYMSG
285	(11D)	BITSTRING	1	SJBFLG6	FLAG BYTE 6
285	(11D)	X'1F'	0	SJB6NODA	"\$ODANYWP" NORMAL OUTDISP FROM CAT
286	(11E)	BITSTRING	1	SJBFLG7	FLAG BYTE 7
286	(11E)	X'1F'	0	SJB7AODA	"\$ODANYWP" ABNORMAL OUTDISP FROM CAT
287	(11F)	BITSTRING	1	SJBFLGC	Flag Byte SERIALIZATION: NONE
		1... ....		SJBCARMI	"B'10000000" Notify ARM of job term
		..1. ....		SJBCHASP	"B'01000000" EOM processing required in HASP address space
		...1 .....		SJBCFJST	"B'00010000" First batch job select
288	(120)	BITSTRING	1	SJBSBCNT	Number of data sets opened for spool browse (count never decremented)
289	(121)	BITSTRING	1	SJBFLGD	Flags Serialized via compare and swap
		1... ....		SJBDSAPI	"B'10000000" Job has at least 1 SAPID
		..1. ....		SJBDJWEL	"B'01000000" Appl hold JWEL created
		...1 .....		SJBDUNSP	"B'00100000" IOT(s) must be unspun
		.... 1...		SJBDJLSP	"B'00001000" JESLOG spin deferred waiting for SJB lock
290	(122)	SIGNED	2	SJBHJE00	Footprint for progress through HASCJBST HJE000
292	(124)	SIGNED	4	SJBJBNUM	Binary job number
296	(128)	CHARACTER	8	SJBJOBID	JOB IDENTIFIER - EBCDIC, NUMERIC

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
304	(130)	CHARACTER	8	SJBJOBNM	JOBNAME FROM JOB CARD
312	(138)	CHARACTER	8	SJBUSRID	USERID FROM JOB CARD
320	(140)	BITSTRING	8	SJBASCBT	Address space token
328	(148)	CHARACTER	8	SJBSECLB	SECLABEL for address space (SDSF use)
336	(150)	CHARACTER	8	SJBJOBCL	JES2 8 character job class
344	(158)	BITSTRING	1	SJBFLG8	Second excession limit flgs
		1... ....		SJB8LJLC	"X'80" Lines JCL Limit is Cancel
		.1... ....		SJB8LJLD	"X'40" Lines JCL Limit is Dump
		..1. ....		SJB8LJLW	"X'20" Lines JCL Limit is Warning
Comment					
EQU X'10' RESERVED for future use					
End of Comment					
		.... 1...		SJB8PJLC	"X'08" Pages JCL Limit is Cancel
		.... .1..		SJB8PJLD	"X'04" Pages JCL Limit is Dump
		.... ..1.		SJB8PJLW	"X'02" Pages JCL Limit is Warning
Comment					
EQU X'01' RESERVED for future use					
End of Comment					
345	(159)	BITSTRING	1	SJBFLG9	First excession limit flags
		1... ....		SJB9BJLC	"X'80" Bytes JCL Limit is Cancel
		.1... ....		SJB9BJLD	"X'40" Bytes JCL Limit is Dump
		..1. ....		SJB9BJLW	"X'20" Bytes JCL Limit is Warning
Comment					
EQU X'10' RESERVED for future use					
End of Comment					
		.... 1...		SJB9CJLC	"X'08" Cards JCL Limit is Cancel
		.... .1..		SJB9CJLD	"X'04" Cards JCL Limit is Dump
		.... ..1.		SJB9CJLW	"X'02" Cards JCL Limit is Warning
Comment					
EQU X'01' RESERVED for future use					
End of Comment					
346	(15A)	BITSTRING	1	SJBFLGA	APPC flag byte
		1... ....		SJBFAFALL	"X'80" First allocation processing
		.1... ....		SJBATP	"X'40" Transaction Program
		..1. ....		SJBAPROT	"X'20" Job is Protected
		...1 ....		SJBASPOT	"X'10" Spin output produced
		.... 1...		SJBASTIN	"X'08" STOP initiator
		.... .1..		SJBATI	"X'04" Transaction Initiator
		.... ..1.		SJBAWSTP	"X'02" WLM posted initiator
347	(15B)	BITSTRING	1	SJBFLGB	Yet another flag byte
		1... ....		SJBBRJI	"X'80" Request job id flag
		.1... ....		SJBBSYSL	"X'40" SYSLOG flag
		..1. ....		SJBBSYSA	"X'20" System address space
		...1 ....		SJBBSPIN	"X'10" Joblog is spinnable
		.... 1...		SJBBOBL	"X'08" Joblog is to be opened for request jobid caller
		.... .1..		SJBNSPN	"X'04" Joblog is not spinnable
		.... ..1.		SJBBRJCR	"X'02" Request job id set Job Correlator in JSAB (SYS_CORR_CURRJOB)
352	(160)	DBL WORD	8	SJDBLWK	DOUBLEWORD WORKAREA #1
360	(168)	DBL WORD	8	SJDBLW1	DOUBLEWORD WORKAREA #2
368	(170)	CHARACTER	1	SJBTOKEN	Security token work area

# \$\$SJB Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>The following fields contain the current excession limits (in packed decimal format) for a job.            When the job's output reaches one of these limits, message \$HASP375 will be issued and a new limit is generated by adding whether a default or exit9-supplied increment amount.</p>					
End of Comment					
448	(1C0)	DBL WORD	8	(0)	Ensure doubleword alignment
448	(1C0)		8	SJBDELIN	Line excession limit
456	(1C8)		8	SJBDEPUN	Punch (card) excess. limit
464	(1D0)		8	SJBDEPAG	Page excession limit
472	(1D8)		8	SJBDEBYT	Byte excession limit
480	(1E0)	CHARACTER	8	SJBWSCNO	WLM Init counted srv class
Comment					
<p>SJBWSCN - Job Select SSI processing field. Contains the service class associated with a batch job executing in a JES2 Initiator.            Note: The field SJBWSCN contains transient data and should not be used to supply the service class for a WLM Initiator or the batch job executing in a WLM initiator. Use field SJBWSCNO to obtain the service class that pertains to a WLM Initiator and any batch job it may be executing.</p>					
End of Comment					
488	(1E8)	CHARACTER	8	SJBWSCN	WLM Service class name
496	(1F0)	SIGNED	4	(10)	Reserved for future use
536	(218)	DBL WORD	8	SJBSSIWK (0)	SSI ROUTINE WORK AREA
Comment					
<p>Job Select SSI processing fields</p>					
End of Comment					
536	(218)	BITSTRING	4	SJBWSC TK	WLM Service class token
540	(21C)	BITSTRING	1	SJBWFLG1	WLM Flags
		1... ..		SJBW1DMD	"B'10000000" Demand select initiator
		.1.. ..		SJBW1SCS	"B'01000000" Service class set via cmd
		..1. ....		SJBW1\$SJ	"B'00100000" Job started via \$\$ J cmd
541	(21D)	BITSTRING	1	SJBWPRIO	Jobs current priority
542	(21E)	BITSTRING	1	SJBWFLG2	Additional job flags
		1... ..		SJBW2AST	"B'10000000" Job has alternate system symbol table
		.1.. ..		SJBW2IST	"B'01000000" Job has datasets with symbol substitution
543	(21F)	BITSTRING	1		Reserved
544	(220)	BITSTRING	8	SJBWDBJI (0)	Demand job's identifier --+
544	(220)	BITSTRING	4	SJBWDBJN	Job number
548	(224)	BITSTRING	4	SJBWDBJK	Job key --+
552	(228)	BITSTRING	16	SJBSCENV	WLM Scheduling environment
568	(238)	BITSTRING	4	SJB SRMTK	SRM token (from IWMCLSFY)
572	(23C)	BITSTRING	4	SJBARRIV	Job arrival time
576	(240)	BITSTRING	8	SJBRHLD	Duration job was ineligible for selection due to a hold
584	(248)	BITSTRING	8	SJBRRSC	Duration job was ineligible for selection due to unsatisfied resource requirements.
592	(250)	BITSTRING	8	SJBRTOC	Duration job was in conversion
600	(258)	CHARACTER	64	SJBSJCOR	Job correlator



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Allocation SSI processing fields					
End of Comment					
536	(218)	SIGNED	2	SJBINSCT	Dataset instance count
664	(298)	DBL WORD	8	(0)	Ensure size fullword mult
664	(298)	X'298'	0	SJBEND	*** END OF SJB
664	(298)	X'298'	0	SJBFSIZE	**SJB" SIZE OF SJB CONTROL BLOCK

**\$SJB Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SJB	0		SJB EOMFL	E8	
SJBAFALL	15A	80	SJBEOMF1	E8	80
SJBAPROT	15A	20	SJBEOMF2	E8	40
SJBARRIV	23C		SJBEOMN	DC	
SJBASCBA	B0		SJBEOMP	E0	
SJBASCBP	64		SJBEOMPC	E4	
SJBASCBT	140		SJBEOMQ	EC	
SJBASID	11A		SJBEOMRS	E9	
SJBASPOT	15A	10	SJBERRET	A8	
SJBASTIN	15A	8	SJBEVICH	8B	8
SJBATI	15A	4	SJBEVICT	8B	10
SJBATP	15A	40	SJBFAMLY	CD	
SJBAWSTP	15A	2	SJBFIRST	119	80
SJBBJOBL	15B	8	SJBFLGA	15A	
SJBBNSPN	15B	4	SJBFLGB	15B	
SJBBRJCR	15B	2	SJBFLGC	11F	
SJBBRJI	15B	80	SJBFLGD	121	
SJBBSPIN	15B	10	SJBFLGE	8B	
SJBBSYSA	15B	20	SJBFLGEF	88	
SJBBSYSL	15B	40	SJBFLG1	6	
SJBCARMI	11F	80	SJBFLG2	7	
SJBCFJST	11F	10	SJBFLG3	D2	
SJBCHASP	11F	40	SJBFLG4	D3	
SJBCKID	68		SJBFLG5	11C	
SJBCSCB	94		SJBFLG6	11D	
SJBCURVN	4	6	SJBFLG7	11E	
SJBDBLWK	160		SJBFLG8	158	
SJBDBLW1	168		SJBFLG9	159	
SJBDEBYT	1D8		SJBHJE00	122	
SJBDELIN	1C0		SJBID	0	
SJBDEPAG	1D0		SJBINSCT	218	
SJBDEPUN	1C8		SJBINTCT	F0	
SJBDJLSP	121	8	SJBIOT	4C	
SJBDJWEL	121	40	SJBJBNUM	124	
SJBDSAPI	121	80	SJBJBSEL	8C	
SJBDUNSP	121	20	SJBJCT	24	
SJBECB	B4		SJBJCTTK	28	
SJBECBA	98		SJBJKEY	20	
SJBECBL	98		SJBJOBCL	150	
SJBECBP	AC		SJBJOBID	128	
SJBECBS	9C		SJBJOBNM	130	
SJBECBW	A0		SJBJQOFF	44	
SJBEEND	A0	80	SJBLCKPT	119	2
SJBEJSAC	8B	80	SJBLKFG	119	
SJBEJSCN	8B	40	SJBLOCKH	F8	
SJBEJSDM	8B	20	SJBMAXCC	91	
SJBEND	298	298	SJBMAXRC	90	
SJB EOMCC	84		SJBMSWBP	119	1
SJB EOMCH	DC		SJBMXIND	90	

## \$SJB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SJBNEXTL	FC		SJBTPST	119	8
SJBOCT	54		SJBUSER	10	
SJBPATID	C8		SJBUSRID	138	
SJBPIIT	60		SJBVRSN	4	
SJBPRIO	CC		SJBWAVE	C	
SJBPSO	108		SJBWDBJI	220	
SJBPSOA	10C		SJBWDBJK	224	
SJBPUFSP	119	20	SJBWDBJN	220	
SJBQUEUE	D4		SJBWFLG1	21C	
SJBRHLD	240		SJBWFLG2	21E	
SJBRFSC	248		SJBWPRI0	21D	
SJBRSNCD	F2		SJBWSCN	1E8	
SJBRSN00	F2	0	SJBWSCNO	1E0	
SJBRSN01	F2	4	SJBWSTCK	218	
SJBRSN02	F2	8	SJBW1\$SJ	21C	20
SJBRSN03	F2	C	SJBW1DMD	21C	80
SJBRSN04	F2	10	SJBW1SCS	21C	40
SJBRSN05	F2	14	SJBW2AST	21E	80
SJBRSN06	F2	18	SJBW2IST	21E	40
SJBRSN07	F2	1C	SJBXMPL	A8	
SJBRSN08	F2	20	SJBXQCHN	D8	
SJBRSN09	F2	24	SJBXQFN1	D0	
SJBRSN10	F2	28	SJBXSTIM	80	
SJBRSN11	F2	2C	SJB1CRAL	6	20
SJBRSN12	F2	30	SJB1EJOB	6	4
SJBRSN13	F2	34	SJB1PI	6	80
SJBRSN14	F2	38	SJB1SJID	6	40
SJBRSN15	F2	3C	SJB1SWBU	6	2
SJBRSN16	F2	40	SJB1WIN	6	1
SJBRSN17	F2	44	SJB2CNCL	7	10
SJBRTOC	250		SJB2CONV	7	8
SJBSASCB	C0		SJB2EJST	7	40
SJBSBCNT	120		SJB2EOM	7	20
SJBSCENV	228		SJB2HOLD	7	4
SJBSDB	1C		SJB2INIT	7	1
SJBSECB	60		SJB2JNL	7	2
SJBSECBP	BC		SJB2PNIT	7	80
SJBSECBS	C4		SJB3CKPT	D2	4
SJBSECLB	148		SJB3CLS	D2	80
SJBSERRE	B8		SJB3FIOT	D2	2
SJBSIZE	298	298	SJB3FJCT	D2	1
SJBSJB	18		SJB3FSDB	D2	40
SJBSJCOR	258		SJB3PPOU	D2	10
SJBSJPTR	58		SJB3TERM	D2	20
SJBSJXB	8		SJB4FSJB	D3	8
SJBSPLOT	50		SJB4MEND	D3	80
SJBSRMTK	238		SJB4MREQ	D3	20
SJBSSIB	14		SJB4MREX	D3	10
SJBSSIWK	218		SJB4MRQH	D3	4
SJBSSNM	48		SJB4MTRM	D3	40
SJBSTAC	110		SJB4OCAN	D3	2
SJBSTACA	114		SJB4TERM	D3	1
SJBSTQE	6C		SJB5JCI	11C	40
SJBWBUF	5C		SJB5JL	11C	80
SJBSXMPL	B8		SJB5JOB	11C	3
SJBTCB	F8		SJB5MSG	11C	20
SJBTCBP	30		SJB5REST	11C	4
SJBTCBT	34		SJB5STC	11C	1
SJBTIMOP	7C		SJB5SWAC	11C	8
SJBTIMX	78		SJB5TSU	11C	2
SJBTINA	100		SJB6NODA	11D	1F
SJBTINAA	104		SJB7AODA	11E	1F
SJBTKCEL	119	10	SJB8LJLC	158	80
SJBTMINT	78		SJB8LJLD	158	40
SJBTOKEN	170		SJB8LJLW	158	20

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SJB8PJLC	158	8
SJB8PJLD	158	4
SJB8PJLW	158	2
SJB9BJLC	159	80
SJB9BJLD	159	40
SJB9BJLW	159	20
SJB9CJLC	159	8
SJB9CJLD	159	4
SJB9CJLW	159	2



---

## **\$SJIOB Information**

### **\$SJIOB Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SJIOB**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$\$SJIOB Map

### \$\$SJIOB Heading Information

**Common Name:** Subsystem Job I/O Buffer  
**Macro ID:** \$\$SJIOB  
**DSECT Name:** SJIOB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** SJIO  
 Offset: SJIOBID-SJIOB  
 Length: L'SJIOBID  
**Storage Attributes:** Subpool: 230  
 Key: 1  
 Residency: Virtual - 24 bit storage (contains IOB) Real - 31 bit storage (contains CCW and data buffers)  
**Size:** See SJIOBSZE  
**Created by:** SJBINIT/SJIOBINT  
**Pointed to by:** SJXBPIOB (for permanent SJIOBs)  
 HXBSJIOB (for SIGIO SJIOBs)  
 A register (for temporary SJIOBs)  
**Serialization:** None  
**Function:** The \$\$SJIOB contains the I/O fields needed in the user or subtask environments. It is also used by the \$\$SIGIO service in the user, subtask and main task environment.  
 The SJIOB exist in two forms defined as permanent and temporary. The permanent SJIOB is pointed to from the SJXB or the HASXB for SIGIO. Whereas, the temporary SJIOB is anchored in a register. The SJIOB contains DCB, DEB, IOB and ECB used by the CBIO and SIGIO routines.

### \$\$SJIOB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SJIOB	
0	(0)	CHARACTER	4	SJIOBID	SJIOB IDENTIFIER
4	(4)	BITSTRING	1	SJIFLAG2	
		1... ....		SJ2TEMP	"B'10000000" TEMPORARY SJIOB
		.1.. ....		SJ2USE	"B'01000000" SJIOB in use
		..1. ....		SJ2SIGI	"B'00100000" SJIOB for \$\$SIGIO
		...1 ....		SJ2MIGO	"B'00010000" During spool migration, override mapped volume consideration.
		.... 1...		SJ2SIOA	"B'00001000" STARTIO appendage entered
		.... .1..		SJ2CEAA	"B'00000100" CE appendage entered
5	(5)	BITSTRING	1	SJIMIGT	DAS transition count
6	(6)	BITSTRING	1	SJICBFG1	CBIO or SIGIO flag 1 copy
7	(7)	BITSTRING	1	SJICBFG2	Copy of CBIO flag 2
8	(8)	ADDRESS	4	SJIOSJXB	ADDRESS OF SJXB
12	(C)	ADDRESS	4	SJIOCBP	\$\$CBIO CBP address

Comment

-----  
 \$\$SIGIO in the main task uses the \$\$SJIOB in the main task. Main task needs an XECB to post the waiting PCE. XECB remaps fields used by \$\$CBIO.  
 -----

End of Comment

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
16	(10)	DBL WORD	8	SJIMECB (0)	Define an XECB for main task usage (\$SIGIO)
16	(10)	SIGNED	4	SJIECB	ECB whose addr is in IOB
20	(14)	SIGNED	4	SJIJBMSK	Exit job mask
24	(18)	DBL WORD	8	SJIMQTR	MQTR for I/O (00MT TTTR)
32	(20)	SIGNED	4	SJIKEY	Job key for data area
36	(24)	CHARACTER	4	SJIVERID	Control block verify id
40	(28)	DBL WORD	8	(0)	Alignment
40	(28)	BITSTRING	0	SJIOB (0)	IOB FOR JOB CONTROL BLOCKS
40	(28)	BITSTRING	1	SJIIFLG1	IOB - FLAG BYTE
41	(29)	BITSTRING	1	SJIIFLG2	IOB - Flag byte 2
42	(2A)	BITSTRING	1	SJIISNS0	IOB - FIRST SENSE BYTE
43	(2B)	BITSTRING	1	SJIISNS1	IOB - SECOND SENSE BYTE
44	(2C)	ADDRESS	4	SJIECB (0)	IOB - EVENT CNTRL BLK ADR
44	(2C)	BITSTRING	1	SJIICMP	IOB - COMPLETION CODE
45	(2D)	ADDRESS	3	SJIECBP	IOB - ECB POINTER (SJBECB)
48	(30)	BITSTRING	1	SJIIFLG3	IOB - Flag byte 3
49	(31)	BITSTRING	7	SJIICSW (0)	IOB - CHANNEL STATUS WORD
49	(31)	ADDRESS	3	SJIICSWA	IOB - CSW ADDRESS PORTION
52	(34)	BITSTRING	1	SJIIST0	IOB - FIRST STATUS BYTE
53	(35)	BITSTRING	1	SJIIST1	IOB - SECOND STATUS BYTE
54	(36)	SIGNED	2	SJIILEN	IOB - RESIDUAL LENGTH
56	(38)	BITSTRING	1	SJIISIO (0)	IOB - SIO condition code
56	(38)	ADDRESS	4	SJIIST	IOB - Channel program addr
60	(3C)	BITSTRING	1	SJIIFLG4 (0)	IOB - Flag byte
60	(3C)	ADDRESS	4	SJIIDCB	IOB - DCB ADDRESS
64	(40)	ADDRESS	4	SJIIRS	IOB - RESTART CHAN PGM ADR
68	(44)	ADDRESS	4		
72	(48)	DBL WORD	8	SJIIFDAD	IOB - FULL DISK ADDRESS Form - MBCCcchR Note: this is absolute address format
72	(48)	X'49'	0	SJIIBCC	"SJIIFDAD+1,7" IOB - BBCCcchR part
80	(50)	SIGNED	4	(0)	Ensure word alignment
80	(50)	BITSTRING	48	SJIOBE	Reserve space for IOB extension
128	(80)	SIGNED	4	(0)	Ensure word alignment
128	(80)	BITSTRING	48	SJIIEDB	Reserve space for I/O error data block
136	(88)	SIGNED	4	SJIDCB (0)	Start of DCB
176	(B0)	BITSTRING	1		Space for DCB foundation
176	(B0)	X'B4'	0	SJIDEBAD	"SJIDCB+(DCBDEBAD-IHADCB),L'DCBDEBAD" Ptr from DCB to DEB

Comment

-----  
 SJIDASID has index of DAS which is a target of JES2 I/O request (M in MTTR/MQTR).  
 SJIRDASX has index of DAS which is a target of actual channel program.  
 They could be different. e.g. for a mapped volume or during some migration phases.  
 -----

End of Comment

188	(BC)	BITSTRING	1	SJIDASID	DAS index
189	(BD)	BITSTRING	1	SJIRDASX	Index of real DAS
190	(BE)	BITSTRING	2		Reserved
192	(C0)	BITSTRING	1	SJIDEB	Space for DEB
192	(C0)	X'DC'	0	SJIAPPAD	"SJIDEB+(DEBAPPAD-DEBBASIC),L'DEBAPPAD" Appendage vector addr
192	(C0)	X'D8'	0	SJIDCBAD	"SJIDEB+(DEBDCBAD-DEBBASIC),L'DEBDCBAD" Ptr from DEB to DCB
224	(E0)	BITSTRING	16	SJIDEBXT	Space for one DA DEB extent
240	(F0)		8	SJICCW1	SET SECTOR/NO-OP
248	(F8)		8	SJICCW2	SEARCH ID EQUAL
256	(100)		8	SJICCW3	TIC *-8
264	(108)		8	SJICCW4	WRITE/READ DATA

## \$SJIOB Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
IDAW'S - CURRENTLY SUPPORT UP TO 4K CONTROL BLOCK SIZE					
End of Comment					
272	(110)	ADDRESS	4	SJIIDAW1	ADDRESS OF CNTRL BLK BUFFER
276	(114)	ADDRESS	4	SJIIDAW2	2K PAGE BNDRY GT THE BUFR ADDR
280	(118)	ADDRESS	4	SJIIDAW3	NEXT 2K PAGE BOUNDARY
284	(11C)	SIGNED	4		Reserved
Comment					
----- Provide data area for signature record I/O -----					
End of Comment					
272	(110)	BITSTRING	8	SJISIG	Data read/written here
280	(118)	ADDRESS	4	SJISGBUF	SIGIO input buffer address
284	(11C)	BITSTRING	1	SJISGRPS	RPS value for CKD CCWs
285	(11D)	BITSTRING	1	SJISGFG1	SIGIO output flags
		1... ....		SJISG1VE	"B'10000000" Validation error
		.1.. ....		SJISG1RD	"B'01000000" Read request
286	(11E)	BITSTRING	2		Reserved
288	(120)	DBL WORD	8	(0)	Ensure alignment
288	(120)	X'120'	0	SJIOBSZE	"*-SJIOB" Size of SJIOB

## \$SJIOB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SJIAPPAD	C0	DC	SJIOB	28	
SJICBFG1	6		SJIOBE	50	
SJICBFG2	7		SJIIRS	40	
SJICCW1	F0		SJIISIO	38	
SJICCW2	F8		SJIINS0	2A	
SJICCW3	100		SJIINS1	2B	
SJICCW4	108		SJIIST	38	
SJIDASID	BC		SJIIST0	34	
SJIDCB	88		SJIIST1	35	
SJIDCBAD	C0	D8	SJIJBMSK	14	
SJIDEB	C0		SJIKKEY	20	
SJIDEBAD	B0	B4	SJIMECB	10	
SJIDEBXT	E0		SJIMIGT	5	
SJIECB	10		SJIMQTR	18	
SJIFLAG2	4		SJIOB	0	
SJIIBBCC	48	49	SJIOBID	0	
SJIICMP	2C		SJIOBSZE	120	120
SJIICSW	31		SJIOCBP	C	
SJIICSWA	31		SJIOSJXB	8	
SJIIDAW1	110		SJIRDASX	BD	
SJIIDAW2	114		SJISGBUF	118	
SJIIDAW3	118		SJISGFG1	11D	
SJIIDCB	3C		SJISGRPS	11C	
SJIECB	2C		SJISG1RD	11D	40
SJIECBP	2D		SJISG1VE	11D	80
SJIEEDB	80		SJISIG	110	
SJIIFDAD	48		SJIVERID	24	
SJIIFLG1	28		SJI2CEAA	4	4
SJIIFLG2	29		SJI2MIGO	4	10
SJIIFLG3	30		SJI2SIGI	4	20
SJIIFLG4	3C		SJI2SIOA	4	8
SJIILEN	36		SJI2TEMP	4	80



Name	Hex Offset	Hex Value
SJ12USE	4	40

## \$SJIOB Cross Reference

---

## **\$SJXB Information**

### **\$SJXB Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SJXB**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$SJXB Map

### \$SJXB Heading Information

**Common Name:** Subsystem Job Block Extension  
**Macro ID:** \$SJXB  
**DSECT Name:** SJXB  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'SJXB'  
 Offset: SJXBID-SJXB  
 Length: 4  
**Storage Attributes:** Subpool: 230  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M) in the address space where the job that the \$SJXB represents is active.  
**Size:** See SJXBSIZE  
**Created by:** Created by the \$SJBINIT service when the job enters execution.  
**Pointed to by:** SJBSJXB field of the \$SJB data area  
**Serialization:** Serialized via \$SJBLOCK service.  
**Function:** The SJB and SJXB are the main control blocks representing a job in the subsystem. The SJXB contains the information that is needed only in the user address space. The SJB contains the information that needs to be shared between the user and the subsystem address spaces.

The SJXB contains work area fields used by SSI functions and a pointer to the SJB. It also has a pointer to the SJIOB which contains the IOB and ECB used by \$CBIO for control block I/O in addition to the ACB and DEB control blocks for the subsystem data sets. It also contains the RPL control block for the job log data set.

### \$SJXB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SJXB	
0	(0)	CHARACTER	4	SJXBID	SJXB IDENTIFIER
4	(4)	ADDRESS	4	SJXBSJBA	ADDRESS OF SJB
8	(8)	ADDRESS	4	SJXBDEBS	Address of SJXB DEB area
12	(C)	ADDRESS	4	SJXBPIOB	ADDRESS OF SJIOB
16	(10)	ADDRESS	4	SJXGGST	ADDRESS OF GROUPING STRINGS OBJECT
Comment					
THE FIELDS FROM SJXCLBEG TO SJXCLEND ARE CLEARED OUT IN HASCBST EVERYTIME THE SJXB IS REUSED FOR ANOTHER JOB.					
End of Comment					
20	(14)	ADDRESS	4	SJXCLBEG (0)	START OF CLEARED SECTION
24	(18)	DBL WORD	8	SJXBUSAV (0)	SAVE AREA FOR UNALLOCATION
24	(18)	ADDRESS	4	SJXBBIOT	NEXT IOT ADDR SAVE AREA
28	(1C)	BITSTRING	1	SJXBSPDB	PDBFLAG1 SAVE AREA
29	(1D)	BITSTRING	1	SJXFLAG1	FLAGS - Flag needs to be serialized by OIL & NIL
		1... ....		SJX1PLHD	"B'10000000" SJXRIOT POINTS TO PLACEHOLDER
		.1.. ....		SJX1JPGM	"B'01000000" Job has page mode records
		..1. ....		SJX1JSDS	"B'00100000" JES2 system dataset involved in excession
		...1 ....		SJX1J722	"B'00010000" Job abended with abend722 (never reset)
30	(1E)	BITSTRING	1		Reserved for future use
31	(1F)	BITSTRING	1	SJXFLAG2	Flags - serialized via Compare and Swap
		1... ....		SJX2TITL	"B'10000000" Joblog title line written

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.1.. ....		SJX2INTR	"B'01000000" Interpreter called during conversion phase
32	(20)	BITSTRING	3		Reserved for future use
35	(23)	BITSTRING	1	SJXJOBRC	JOBRC value
		.1.. ....		SJXJBMRC	"X'40" JOBRC=MAXRC specified
		..1. ....		SJXJBLRC	"X'20" JOBRC=LASTRC specified
		...1 ....		SJXJBSRC	"X'10" JOBRC=(STEP,x) specified
36	(24)	CHARACTER	8	SJXJBPSN	JOBRC name on 'EXEC PGM='
44	(2C)	CHARACTER	8	SJXJBPPS	JOBRC name on 'EXEC PROC='
52	(34)	SIGNED	4	SJXSPNUM	NUMBER OF SPIN IOTS UNALLOCATED AND NOT YET REUSED
56	(38)	ADDRESS	4	SJXRIOT	ADDRESS OF LAST NORMAL IOT USED IN REUSE SEARCH
60	(3C)	ADDRESS	4	SJXRPDDB	OFFSET OF LAST NULL PLACEHOLDER Pddb USED IN REUSE SEARCH
64	(40)	ADDRESS	4	SJXDSCA	For SYSLOG jobs, address of first data set catalog (DSCA) block
68	(44)	ADDRESS	4	SJXALTST	Addr of alternate system symbol table for instream symbol substitution
72	(48)	BITSTRING	8		Reserved
80	(50)	ADDRESS	4	SJXSLOGP	Ptr to 1st logging YLGC

Comment

THE ESTIMATED COUNT FIELDS MUST BE KEPT TOGETHER AND ARE MAPPED BY THE EST DSECT GENERATED BY THE \$EST MACRO

End of Comment

84	(54)	SIGNED	4	SJXLINES (0)	LINE EXCESSION FIELDS
84	(54)	SIGNED	4	SJXLNINT	EST LINE MESSAGE INTERVAL
88	(58)	BITSTRING	1	SJXLNOP	EXECUTION LINE OPTION
89	(59)	BITSTRING	3		RESERVED FOR FUTURE USE
92	(5C)	SIGNED	4	SJXPUNCH (0)	PUNCH EXCESSION FIELDS
92	(5C)	SIGNED	4	SJXPUNINT	EST CARD MESSAGE INTERVAL
96	(60)	BITSTRING	1	SJXPUNOP	EXECUTION CARD OPTION
97	(61)	BITSTRING	3		RESERVED FOR FUTURE USE
100	(64)	SIGNED	4	SJXPAGES (0)	PAGES EXCESSION FIELDS
100	(64)	SIGNED	4	SJXPGININT	EST PAGES MESSAGE INTERVAL
104	(68)	BITSTRING	1	SJXPGOP	EXECUTION PAGES OPTION
105	(69)	BITSTRING	3		RESERVED FOR FUTURE USE
108	(6C)	SIGNED	4	SJXBYTES (0)	BYTES EXCESSION FIELDS
108	(6C)	SIGNED	4	SJXBYINT	EST BYTES MESSAGE INTERVAL
112	(70)	BITSTRING	1	SJXBYTOP	EXECUTION BYTE OPTION
113	(71)	BITSTRING	3		RESERVED FOR FUTURE USE

Comment

END OF THE ESTIMATED COUNT FIELDS  
The following area, to SJXUSER, should be used for any new fields retrieved using the SJFACC facility and module HASPSJFA if any are added in the future.

End of Comment

116	(74)	CHARACTER	4	SJXACCT	Account number
120	(78)	CHARACTER	1	SJXMSGCL	MSGCLASS value for TPs only
121	(79)	BITSTRING	3		Reserved for future use

Comment

FIELD RESERVED FOR THE USER

End of Comment

124	(7C)	ADDRESS	4	SJXUSER	*** RESERVED FOR USER ***
128	(80)	ADDRESS	4	SJXJMR	Address of extended JMR in 24 bit storage
132	(84)	SIGNED	4	(4)	Reserved for future use
148	(94)	SIGNED	4	SJXPBFCT	NUMBER OF EXTRA PBUFS

## \$SJXB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
152	(98)	ADDRESS	4	SJXWTCHN	Chain of SDBs waiting for PBUFs (for PUT)
156	(9C)	SIGNED	4	SJXRESRV	RESERVED FOR FUTURE IBM USE
160	(A0)	BITSTRING	4	SJXDUMST	Dummy system symbol table
168	(A8)	DBL WORD	8	SJXDWORK	General work area
176	(B0)	DBL WORD	8	(0)	Ensure doubleword alignment
176	(B0)		8	SJXJBPRT	Job total printed output (in packed decimal form)
184	(B8)		8	SJXJPUN	Job total punched output (in packed decimal form)
192	(C0)		8	SJXJPAG	Job total page count (in packed decimal form)
200	(C8)		8	SJXJBYT	Job total byte count (in packed decimal form)
208	(D0)		8	SJXSPUNB	Job total spun byte count (in packed decimal form)
216	(D8)	CHARACTER	64	SJXJCOR	Job Correlator
280	(118)	DBL WORD	8	SJXCLEND (0)	END OF CLEARED SECTION
280	(118)	SIGNED	4	SJXPBFLM	Limit on number of PBUFs allowed in this addr space
Comment					
Addresses of SPCs for Joblog and SYSMSG					
End of Comment					
284	(11C)	ADDRESS	4	SJXLSPC	JOBLOG SPC address
288	(120)	ADDRESS	4	SJXMSPC	SYSMSG SPC address
Comment					
ACB FOR HASP JOB LOG DATASET					
End of Comment					
292	(124)	SIGNED	4	SJXLACB (0)	
292	(124)	BITSTRING	1		. ACB IDENTIFICATION
293	(125)	ADDRESS	1		ACB SUBTYPE X04SVHS
294	(126)	ADDRESS	2		. ACB LENGTH X03004HS
296	(128)	ADDRESS	4		. AMB LIST POINTER
300	(12C)	ADDRESS	4		. INTERFACE ROUTINE POINTER
304	(130)	BITSTRING	1		MACRF(1) X04SVHS
305	(131)	BITSTRING	1		MACRF(2) X04SVHS FOR NUMERIC IN PARENS
306	(132)	ADDRESS	1		. NO OF CONCURRENT X04SVHS STRINGS FOR AIX PATH X04SVHS FOR NUMERIC IN PARENS
307	(133)	ADDRESS	1		. NUMBER OF STRINGS X04SVHS FOR NUMERIC IN PARENS
308	(134)	ADDRESS	2		. NUMBER OF DATA BUFFERS FOR NUMERIC IN PARENS
310	(136)	ADDRESS	2		. NUMBER OF INDEX BUFFERS
312	(138)	BITSTRING	1		MACRF(3) X04SVHS FOR NUMERIC IN PARENS
313	(139)	ADDRESS	1		SHARED RESOURCE POOL ID
314	(13A)	ADDRESS	2		. JES BUFFER POOL/NUMBER X04SVHS OF JOURNAL BUFFERS
316	(13C)	BITSTRING	1		. RECFM=A
317	(13D)	BITSTRING	1		READ INTEGRITY OPTIONS
318	(13E)	BITSTRING	2		. DSORG=ACB
320	(140)	ADDRESS	4		X04SVHS
324	(144)	ADDRESS	4		. PASSWORD POINTER
328	(148)	ADDRESS	4		. EXIT LIST POINTER
332	(14C)	CHARACTER	8		
340	(154)	BITSTRING	1		OFLAGS
341	(155)	ADDRESS	1		. ERFLAGS
342	(156)	BITSTRING	1		INFLGS(1) X04SVHS
343	(157)	BITSTRING	1		INFLGS(2) X04SVHS
344	(158)	ADDRESS	4		. OPENJ JFCB POINTER
348	(15C)	ADDRESS	4		BUFFER SPACE
352	(160)	ADDRESS	2		. BLOCK SIZE
354	(162)	ADDRESS	2		. RECORD SIZE
356	(164)	ADDRESS	4		. USER WORKAREA POINTER
360	(168)	ADDRESS	4		CONTROL BLOCK MANIPULATION WORKAREA POINTER
364	(16C)	ADDRESS	4		. PTR TO APPLICATION NAME X03004

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
RPL FOR HASP JOB LOG DATASET					
End of Comment					
368	(170)	SIGNED	4	SJXRPL (0)	
368	(170)	ADDRESS	1		RPL IDENTIFICATION
369	(171)	ADDRESS	1		RPL SUBTYPE X04SVHS
370	(172)	ADDRESS	1		RPL REQUEST TYPE
371	(173)	ADDRESS	1		RPL LENGTH X03004
372	(174)	ADDRESS	4		. POINTER TO PLACEHOLDER
376	(178)	ADDRESS	4		. ECB
380	(17C)	BITSTRING	1		. STATUS BYTE
381	(17D)	BITSTRING	3		FEEDBACK CODES
384	(180)	ADDRESS	2		. KEY LENGTH
386	(182)	ADDRESS	2		. TRANSID
388	(184)	ADDRESS	4		POINTER TO CONTROL CHARACTER
392	(188)	ADDRESS	4		
396	(18C)	ADDRESS	4		. POINTER TO TCB
400	(190)	ADDRESS	4		. POINTER TO RECORD AREA
404	(194)	ADDRESS	4		. POINTER TO ARGUMENT
408	(198)	BITSTRING	1		. OPTCD BYTE 1
409	(199)	BITSTRING	1		
410	(19A)	BITSTRING	1		OPTCD BYTE 3
411	(19B)	BITSTRING	1		OPTCD BYTE 4
412	(19C)	ADDRESS	4		. POINTER TO NEXT RPL
416	(1A0)	ADDRESS	4		. RESERVED FOR LOGICAL REC LEN
420	(1A4)	ADDRESS	4		. RESERVED FOR BUFFER LENGTH OPTCD BYTES 5-8 X03004
424	(1A8)	BITSTRING	1		
425	(1A9)	BITSTRING	1		
426	(1AA)	BITSTRING	1		
427	(1AB)	BITSTRING	1		
428	(1AC)	BITSTRING	8		. RBA
436	(1B4)	BITSTRING	1		
437	(1B5)	ADDRESS	1		ACTIVE INDICATOR
438	(1B6)	ADDRESS	2		. MAXIMUM ERROR MSG LENGTH
440	(1B8)	ADDRESS	4		. RESERVED FOR MESSAGE AREA PTR
440	(1B8)	X'1BC'	0	SJXLOGE	*** End of job log ACB/RPL

Comment

-----  
 ACB, RPL, etc. for NJE from NETSRV address space  
 -----

End of Comment					
292	(124)	SIGNED	4	SJXNJACB (0)	
292	(124)	BITSTRING	1		. ACB IDENTIFICATION
293	(125)	ADDRESS	1		ACB SUBTYPE X04SVHS
294	(126)	ADDRESS	2		. ACB LENGTH X03004HS
296	(128)	ADDRESS	4		. AMB LIST POINTER
300	(12C)	ADDRESS	4		. INTERFACE ROUTINE POINTER
304	(130)	BITSTRING	1		MACRF(1) X04SVHS
305	(131)	BITSTRING	1		MACRF(2) X04SVHS FOR NUMERIC IN PARENS
306	(132)	ADDRESS	1		. NO OF CONCURRENT X04SVHS STRINGS FOR AIX PATH X04SVHS FOR NUMERIC IN PARENS
307	(133)	ADDRESS	1		. NUMBER OF STRINGS X04SVHS FOR NUMERIC IN PARENS
308	(134)	ADDRESS	2		. NUMBER OF DATA BUFFERS FOR NUMERIC IN PARENS
310	(136)	ADDRESS	2		. NUMBER OF INDEX BUFFERS
312	(138)	BITSTRING	1		MACRF(3) X04SVHS FOR NUMERIC IN PARENS
313	(139)	ADDRESS	1		SHARED RESOURCE POOL ID

## \$SJXB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
314	(13A)	ADDRESS	2		. JES BUFFER POOL/NUMBER X04SVHS OF JOURNAL BUFFERS
316	(13C)	BITSTRING	1		. RECFM=A
317	(13D)	BITSTRING	1		READ INTEGRITY OPTIONS
318	(13E)	BITSTRING	2		. DSORG=ACB
320	(140)	ADDRESS	4		X04SVHS
324	(144)	ADDRESS	4		. PASSWORD POINTER
328	(148)	ADDRESS	4		. EXIT LIST POINTER
332	(14C)	CHARACTER	8		
340	(154)	BITSTRING	1		OFLAGS
341	(155)	ADDRESS	1		. ERFLAGS
342	(156)	BITSTRING	1		INFLGS(1) X04SVHS
343	(157)	BITSTRING	1		INFLGS(2) X04SVHS
344	(158)	ADDRESS	4		. OPENJ JFCB POINTER
348	(15C)	ADDRESS	4		BUFFER SPACE
352	(160)	ADDRESS	2		. BLOCK SIZE
354	(162)	ADDRESS	2		. RECORD SIZE
356	(164)	ADDRESS	4		. USER WORKAREA POINTER
360	(168)	ADDRESS	4		CONTROL BLOCK MANIPULATION WORKAREA POINTER
364	(16C)	ADDRESS	4		. PTR TO APPLICATION NAME X03004
368	(170)	SIGNED	4	(0)	
368	(170)	BITSTRING	80	SJXNJRPL	RPL uses SJXNJACB
Comment					
Transaction Processor Fields					
End of Comment					
448	(1C0)	ADDRESS	4	SJXSJBS	Address of SJB save area
452	(1C4)	SIGNED	4	SJXTRPL (0)	
452	(1C4)	ADDRESS	1		RPL IDENTIFICATION
453	(1C5)	ADDRESS	1		RPL SUBTYPE X04SVHS
454	(1C6)	ADDRESS	1		RPL REQUEST TYPE
455	(1C7)	ADDRESS	1		RPL LENGTH X03004
456	(1C8)	ADDRESS	4		. POINTER TO PLACEHOLDER
460	(1CC)	ADDRESS	4		. ECB
464	(1D0)	BITSTRING	1		. STATUS BYTE
465	(1D1)	BITSTRING	3		FEEDBACK CODES
468	(1D4)	ADDRESS	2		. KEY LENGTH
470	(1D6)	ADDRESS	2		. TRANSID
472	(1D8)	ADDRESS	4		POINTER TO CONTROL CHARACTER
476	(1DC)	ADDRESS	4		. POINTER TO ACB
480	(1E0)	ADDRESS	4		. POINTER TO TCB
484	(1E4)	ADDRESS	4		. POINTER TO RECORD AREA
488	(1E8)	ADDRESS	4		. POINTER TO ARGUMENT
492	(1EC)	BITSTRING	1		. OPTCD BYTE 1
493	(1ED)	BITSTRING	1		
494	(1EE)	BITSTRING	1		OPTCD BYTE 3
495	(1EF)	BITSTRING	1		OPTCD BYTE 4
496	(1F0)	ADDRESS	4		. POINTER TO NEXT RPL
500	(1F4)	ADDRESS	4		. RESERVED FOR LOGICAL REC LEN
504	(1F8)	ADDRESS	4		. RESERVED FOR BUFFER LENGTH OPTCD BYTES 5-8 X03004
508	(1FC)	BITSTRING	1		
509	(1FD)	BITSTRING	1		
510	(1FE)	BITSTRING	1		
511	(1FF)	BITSTRING	1		
512	(200)	BITSTRING	8		. RBA
520	(208)	BITSTRING	1		
521	(209)	ADDRESS	1		ACTIVE INDICATOR
522	(20A)	ADDRESS	2		. MAXIMUM ERROR MSG LENGTH
524	(20C)	ADDRESS	4		. RESERVED FOR MESSAGE AREA PTR
528	(210)	ADDRESS	4	SJXTACB	Address of ACB for SYSLOG
532	(214)	BITSTRING	4		Reserved for future use



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
536	(218)	SIGNED	4	SJXJSPCT	Number of SAPI threads awaiting JES2 addresspace
Comment					
MACDATE = 06/13/1996					
End of Comment					
540	(21C)	SIGNED	4	SJXALES (0)	.ALESERV PC PARAMETER LIST
540	(21C)	BITSTRING	1		.SERVICE TYPE CODE
541	(21D)	BITSTRING	1		.OPTIONS FLAG BYTE
542	(21E)	ADDRESS	2		.RESERVED
544	(220)	ADDRESS	4		.ALET
548	(224)	BITSTRING	8		.STOKEN (SPACE TOKEN)
548	(224)	X'10'	0	SJXALESL	**SJXALES" Length of block
Comment					
MACDATE = 04/03/89					
End of Comment					
540	(21C)	SIGNED	4	SJXTTOK (0)	
540	(21C)	CHARACTER	16	(0)	TCB TOKEN (INPUT/OUTPUT)
540	(21C)	BITSTRING	8		
548	(224)	SIGNED	4		
552	(228)	ADDRESS	4		
556	(22C)	ADDRESS	4		ASCB ADDRESS (INPUT)
560	(230)	SIGNED	4	(0)	FLAGS (INPUT)
560	(230)	SIGNED	1		TYPE OF TCBTOKEN REQUEST
561	(231)	SIGNED	3		RESERVED
561	(231)	X'18'	0	SJXTTOKL	**SJXTTOK" Length of block
564	(234)	BITSTRING	196	SJXS35D	Memory in primary space for WTOLOGQ processing (serialized by SJB lock)
760	(2F8)	DBL WORD	8	(0)	
760	(2F8)	X'2F8'	0	SJXBSIZE	**SJXB" SIZE OF SJB EXTENSION

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SJXDEB	, Start of DSECT
0	(0)	CHARACTER	4	SJXDID	Eyecatcher
4	(4)	ADDRESS	4	SJXDSJXB	Address of SJXB
8	(8)	SIGNED	4	SJXDLEN	Length of SJXDEB
12	(C)	SIGNED	4	SJXDSTRT (0)	Start of DEBs
Comment					
----- DEB for HASP job log data set -----					
End of Comment					
12	(C)	BITSTRING	1	SJXLDEB	
Comment					
----- ACB for internal text/SWA blocks data set -----					
End of Comment					
44	(2C)	SIGNED	4	SJXIACB (0)	
44	(2C)	BITSTRING	1		. ACB IDENTIFICATION
45	(2D)	ADDRESS	1		ACB SUBTYPE X04SVHS

# \$SJXB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
46	(2E)	ADDRESS	2		. ACB LENGTH X03004HS
48	(30)	ADDRESS	4		. AMB LIST POINTER
52	(34)	ADDRESS	4		. INTERFACE ROUTINE POINTER
56	(38)	BITSTRING	1		MACRF(1) X04SVHS
57	(39)	BITSTRING	1		MACRF(2) X04SVHS FOR NUMERIC IN PARENS
58	(3A)	ADDRESS	1		. NO OF CONCURRENT X04SVHS STRINGS FOR AIX PATH X04SVHS FOR NUMERIC IN PARENS
59	(3B)	ADDRESS	1		. NUMBER OF STRINGS X04SVHS FOR NUMERIC IN PARENS
60	(3C)	ADDRESS	2		. NUMBER OF DATA BUFFERS FOR NUMERIC IN PARENS
62	(3E)	ADDRESS	2		. NUMBER OF INDEX BUFFERS
64	(40)	BITSTRING	1		MACRF(3) X04SVHS FOR NUMERIC IN PARENS
65	(41)	ADDRESS	1		SHARED RESOURCE POOL ID
66	(42)	ADDRESS	2		. JES BUFFER POOL/NUMBER X04SVHS OF JOURNAL BUFFERS
68	(44)	BITSTRING	1		. RECFM=A
69	(45)	BITSTRING	1		READ INTEGRITY OPTIONS
70	(46)	BITSTRING	2		. DSORG=ACB
72	(48)	ADDRESS	4		X04SVHS
76	(4C)	ADDRESS	4		. PASSWORD POINTER
80	(50)	ADDRESS	4		. EXIT LIST POINTER
84	(54)	CHARACTER	8		
92	(5C)	BITSTRING	1		OFLAGS
93	(5D)	ADDRESS	1		. ERFLAGS
94	(5E)	BITSTRING	1		INFLGS(1) X04SVHS
95	(5F)	BITSTRING	1		INFLGS(2) X04SVHS
96	(60)	ADDRESS	4		. OPENJ JFCB POINTER
100	(64)	ADDRESS	4		BUFFER SPACE
104	(68)	ADDRESS	2		. BLOCK SIZE
106	(6A)	ADDRESS	2		. RECORD SIZE
108	(6C)	ADDRESS	4		. USER WORKAREA POINTER
112	(70)	ADDRESS	4		CONTROL BLOCK MANIPULATION WORKAREA POINTER
116	(74)	ADDRESS	4		. PTR TO APPLICATION NAME X03004

Comment

-----  
DEB for internal text/SWA blocks data set  
-----

End of Comment

120	(78)	BITSTRING	1	SJXIDEB	
-----	------	-----------	---	---------	--

Comment

-----  
ACB for system messages dataset  
-----

End of Comment

152	(98)	SIGNED	4	SJXMACB (0)	
152	(98)	BITSTRING	1		. ACB IDENTIFICATION
153	(99)	ADDRESS	1		ACB SUBTYPE X04SVHS
154	(9A)	ADDRESS	2		. ACB LENGTH X03004HS
156	(9C)	ADDRESS	4		. AMB LIST POINTER
160	(A0)	ADDRESS	4		. INTERFACE ROUTINE POINTER
164	(A4)	BITSTRING	1		MACRF(1) X04SVHS
165	(A5)	BITSTRING	1		MACRF(2) X04SVHS FOR NUMERIC IN PARENS
166	(A6)	ADDRESS	1		. NO OF CONCURRENT X04SVHS STRINGS FOR AIX PATH X04SVHS FOR NUMERIC IN PARENS
167	(A7)	ADDRESS	1		. NUMBER OF STRINGS X04SVHS FOR NUMERIC IN PARENS
168	(A8)	ADDRESS	2		. NUMBER OF DATA BUFFERS FOR NUMERIC IN PARENS
170	(AA)	ADDRESS	2		. NUMBER OF INDEX BUFFERS

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
172	(AC)	BITSTRING	1		MACRF(3) X04SVHS FOR NUMERIC IN PARENS
173	(AD)	ADDRESS	1		SHARED RESOURCE POOL ID
174	(AE)	ADDRESS	2		. JES BUFFER POOL/NUMBER X04SVHS OF JOURNAL BUFFERS
176	(B0)	BITSTRING	1		. RECFM=A
177	(B1)	BITSTRING	1		READ INTEGRITY OPTIONS
178	(B2)	BITSTRING	2		. DSORG=ACB
180	(B4)	ADDRESS	4		X04SVHS
184	(B8)	ADDRESS	4		. PASSWORD POINTER
188	(BC)	ADDRESS	4		. EXIT LIST POINTER
192	(C0)	CHARACTER	8		
200	(C8)	BITSTRING	1		OFLAGS
201	(C9)	ADDRESS	1		. ERFLAGS
202	(CA)	BITSTRING	1		INFLGS(1) X04SVHS
203	(CB)	BITSTRING	1		INFLGS(2) X04SVHS
204	(CC)	ADDRESS	4		. OPENJ JFCB POINTER
208	(D0)	ADDRESS	4		BUFFER SPACE
212	(D4)	ADDRESS	2		. BLOCK SIZE
214	(D6)	ADDRESS	2		. RECORD SIZE
216	(D8)	ADDRESS	4		. USER WORKAREA POINTER
220	(DC)	ADDRESS	4		CONTROL BLOCK MANIPULATION WORKAREA POINTER
224	(E0)	ADDRESS	4		. PTR TO APPLICATION NAME X03004

Comment

-----  
 DEB for system messages data set  
 -----

End of Comment

228	(E4)	BITSTRING	1	SJXMDEB	
-----	------	-----------	---	---------	--

Comment

-----  
 RPL for system messages dataset  
 -----

End of Comment

260	(104)	SIGNED	4	SJXMRPL (0)	
260	(104)	ADDRESS	1		RPL IDENTIFICATION
261	(105)	ADDRESS	1		RPL SUBTYPE X04SVHS
262	(106)	ADDRESS	1		RPL REQUEST TYPE
263	(107)	ADDRESS	1		RPL LENGTH X03004
264	(108)	ADDRESS	4		. POINTER TO PLACEHOLDER
268	(10C)	ADDRESS	4		. ECB
272	(110)	BITSTRING	1		. STATUS BYTE
273	(111)	BITSTRING	3		FEEDBACK CODES
276	(114)	ADDRESS	2		. KEY LENGTH
278	(116)	ADDRESS	2		. TRANSID
280	(118)	ADDRESS	4		POINTER TO CONTROL CHARACTER
284	(11C)	ADDRESS	4		
288	(120)	ADDRESS	4		. POINTER TO TCB
292	(124)	ADDRESS	4		. POINTER TO RECORD AREA
296	(128)	ADDRESS	4		. POINTER TO ARGUMENT
300	(12C)	BITSTRING	1		. OPTCD BYTE 1
301	(12D)	BITSTRING	1		
302	(12E)	BITSTRING	1		OPTCD BYTE 3
303	(12F)	BITSTRING	1		OPTCD BYTE 4
304	(130)	ADDRESS	4		. POINTER TO NEXT RPL
308	(134)	ADDRESS	4		. RESERVED FOR LOGICAL REC LEN
312	(138)	ADDRESS	4		. RESERVED FOR BUFFER LENGTH OPTCD BYTES 5-8 X03004
316	(13C)	BITSTRING	1		

# \$SJXB Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
317	(13D)	BITSTRING	1		
318	(13E)	BITSTRING	1		
319	(13F)	BITSTRING	1		
320	(140)	BITSTRING	8		. RBA
328	(148)	BITSTRING	1		
329	(149)	ADDRESS	1		ACTIVE INDICATOR
330	(14A)	ADDRESS	2		. MAXIMUM ERROR MSG LENGTH
332	(14C)	ADDRESS	4		. RESERVED FOR MESSAGE AREA PTR

Comment

-----  
 ACB for journal dataset  
 -----

End of Comment

336	(150)	SIGNED	4	SJXJACB (0)	
336	(150)	BITSTRING	1		. ACB IDENTIFICATION
337	(151)	ADDRESS	1		ACB SUBTYPE X04SVHS
338	(152)	ADDRESS	2		. ACB LENGTH X03004HS
340	(154)	ADDRESS	4		. AMB LIST POINTER
344	(158)	ADDRESS	4		. INTERFACE ROUTINE POINTER
348	(15C)	BITSTRING	1		MACRF(1) X04SVHS
349	(15D)	BITSTRING	1		MACRF(2) X04SVHS FOR NUMERIC IN PARENS
350	(15E)	ADDRESS	1		. NO OF CONCURRENT X04SVHS STRINGS FOR AIX PATH X04SVHS FOR NUMERIC IN PARENS
351	(15F)	ADDRESS	1		. NUMBER OF STRINGS X04SVHS FOR NUMERIC IN PARENS
352	(160)	ADDRESS	2		. NUMBER OF DATA BUFFERS FOR NUMERIC IN PARENS
354	(162)	ADDRESS	2		. NUMBER OF INDEX BUFFERS
356	(164)	BITSTRING	1		MACRF(3) X04SVHS FOR NUMERIC IN PARENS
357	(165)	ADDRESS	1		SHARED RESOURCE POOL ID
358	(166)	ADDRESS	2		. JES BUFFER POOL/NUMBER X04SVHS OF JOURNAL BUFFERS
360	(168)	BITSTRING	1		. RECFM=A
361	(169)	BITSTRING	1		READ INTEGRITY OPTIONS
362	(16A)	BITSTRING	2		. DSORG=ACB
364	(16C)	ADDRESS	4		X04SVHS
368	(170)	ADDRESS	4		. PASSWORD POINTER
372	(174)	ADDRESS	4		. EXIT LIST POINTER
376	(178)	CHARACTER	8		
384	(180)	BITSTRING	1		OFLAGS
385	(181)	ADDRESS	1		. ERFLAGS
386	(182)	BITSTRING	1		INFLGS(1) X04SVHS
387	(183)	BITSTRING	1		INFLGS(2) X04SVHS
388	(184)	ADDRESS	4		. OPENJ JFCB POINTER
392	(188)	ADDRESS	4		BUFFER SPACE
396	(18C)	ADDRESS	2		. BLOCK SIZE
398	(18E)	ADDRESS	2		. RECORD SIZE
400	(190)	ADDRESS	4		. USER WORKAREA POINTER
404	(194)	ADDRESS	4		CONTROL BLOCK MANIPULATION WORKAREA POINTER
408	(198)	ADDRESS	4		. PTR TO APPLICATION NAME X03004
408	(198)	X'4C'	0	SJACBLGH	**-'SJXJACB' Length of JACB

Comment

-----  
 DEB for journal dataset  
 -----

End of Comment

412	(19C)	BITSTRING	1	SJXJDEB	
-----	-------	-----------	---	---------	--

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Parameter list for the external writer. A copy of the parameter list must be kept below the line since the external writer can not access data above the line.					
End of Comment					
444	(1BC)	SIGNED	4	SJXXWPL (0)	External writer parameter list
444	(1BC)	SIGNED	4	SJXXWECB	ECB for the external writer
448	(1C0)	BITSTRING	4	SJXRDRON	Time on input processor
452	(1C4)	BITSTRING	4	SJXRDTON	Date on input processor
456	(1C8)	CHARACTER	8	SJXUSEID	JMR installation data field
Comment					
End of external writer parameter list					
-----					
Single DEB for special processing cases (NETSERV, \$LOGMSG, etc).					
-----					
End of Comment					
12	(C)	BITSTRING	36	SJX1DEB	
48	(30)	DBL WORD	8	(0)	Round up area length
48	(30)	X'30'	0	SJXD1SIZ	**-'SJXDEB' Length with a single DEB
464	(1D0)	DBL WORD	8	(0)	Round up area
464	(1D0)	X'1D0'	0	SJXDSIZE	**-'SJXDEB' Length of extension

\$SJXB Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SJACBLGH	198	4C	SJXFLAG2	1F	
SJXACCT	74		SJXGGST	10	
SJXALES	21C		SJXIACB	2C	
SJXALES	224	10	SJXIDEB	78	0
SJXALTST	44		SJXJACB	150	
SJXB	0		SJXJBBYT	C8	
SJXBDEBS	8		SJXJBLRC	23	20
SJXBID	0		SJXJBMRC	23	40
SJXBPIOB	C		SJXJBPAF	C0	
SJXBBIOT	18		SJXJBPPS	2C	
SJXBBSIZE	2F8	2F8	SJXJBPRT	B0	
SJXBSJBA	4		SJXJBPSN	24	
SJXBSPDB	1C		SJXJBPN	B8	
SJXBUSAV	18		SJXJBSRC	23	10
SJXBYINT	6C		SJXJCOR	D8	
SJXBYTES	6C		SJXJDEB	19C	0
SJXBYTOP	70		SJXJMR	80	
SJXCLBEG	14		SJXJOBRC	23	
SJXCLEND	118		SJXJSPCT	218	
SJXDEB	0		SJXLACB	124	
SJXDID	0	E2D1E7C4	SJXLDEB	C	0
SJXDLEN	8		SJXLINES	54	
SJXDSCA	40		SJXLNINT	54	
SJXDSIZE	1D0	1D0	SJXLNOP	58	
SJXDSJXB	4		SJXLOGE	1B8	1BC
SJXDSTRT	C		SJXLRPL	170	
SJXDUMST	A0		SJXLSPC	11C	
SJXDWORK	A8		SJXMACB	98	
SJXD1SIZ	30	30	SJXMDEB	E4	0
SJXFLAG1	1D		SJXMRPL	104	

## \$SJXB Cross Reference

Name	Hex Offset	Hex Value
SJXMSGCL	78	
SJXMSPC	120	
SJXNJACB	124	
SJXNJRPL	170	0
SJXPAGES	64	
SJXPBFCT	94	
SJXPBFLM	118	
SJXPGINT	64	
SJXPGOP	68	
SJXPUIINT	5C	
SJXPUNCH	5C	
SJXPUNOP	60	
SJXRDRON	1C0	
SJXRDTON	1C4	
SJXRESRV	9C	
SJXRIOT	38	
SJXRPDDB	3C	
SJXSJBS	1C0	
SJXSLOGP	50	
SJXSPNUM	34	
SJXSPUNB	D0	
SJXS35D	234	
SJXTACB	210	
SJXTRPL	1C4	
SJXTTOK	21C	
SJXTTOKL	231	18
SJXUSEID	1C8	
SJXUSER	7C	
SJXWTCHN	98	
SJXXWECB	1BC	
SJXXWPL	1BC	
SJX1DEB	C	0
SJX1JPGM	1D	40
SJX1JSDS	1D	20
SJX1J722	1D	10
SJX1PLHD	1D	80
SJX2INTR	1F	40
SJX2TITL	1F	80

---

## **\$SMF Information**

### **\$SMF Programming Interface information**

\_\_\_\_\_ Programming Interface information \_\_\_\_\_

**\$SMF**

\_\_\_\_\_ End of Programming Interface information \_\_\_\_\_

## Heading Information • \$SMF Map

### \$SMF Heading Information

**Common Name:** HASP SMF BUFFER DSECT  
**Macro ID:** \$SMF  
**DSECT Name:** SMF  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** None  
**Storage Attributes:** Subpool: See symbols SMFPOOL and B32KPOOL in macro \$HASPEQU.  
 Key: 1  
 Residency: Virtual and real storage are anywhere (above or below 16M) in the JES2 address space.  
**Size:** See SMFLNG  
**Created by:** An SMF buffer is allocated from either the SMF cell pool or the B32K cell pool. These cell pools are created during JES2 initialization. The \$GETSMFB service is used to allocate an SMF buffer from the appropriate cell pool.  
**Pointed to by:** Field \$SMFBUSY in the \$HCT data area points to the queue of SMF buffers to be written. The buffers are chained via field SMFCHAIN. Some PCE work areas point to an SMF buffer.  
**Serialization:** A PCE obtains an SMF buffer using the \$GETSMFB macro. It has exclusive control of the buffer until it queues it for writing (\$QUESMFB macro) or frees it (\$FRESMFB macro). After queueing a buffer, the PCE cannot use it. The HASPACCT subtask frees the buffer after writing it. The \$QUESMFB service uses compare and swap to stack a buffer onto \$SMFBUSY. The HASPACCT subtask uses compare and swap to dequeue the last buffer chained from \$SMFBUSY.  
**Function:** \$SMF contains mappings for types 6,24,26,43,45,47, 48,49,52,53,54,54,55,56,57,and 58 SMF records. IFASMFR is called by \$SMF and expanded within for each SMF record.

When computing actual SMF displacements, remember the JES2 SMF headers contribute 8 bytes to all \$SMF macro displacements.

### \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SMF	HASP SMF BUFFER DSECT
0	(0)	SIGNED	4	SMFCHAIN	SMF BUFFER CHAIN TO NEXT BUFFER
4	(4)	BITSTRING	1	SMFTYPE	TYPE OF BUFFER
		1... ....		SMFJMRTD	"B'10000000" JMR BUFFER
		.1.. ....		SMFLRGTP	"B'01000000" LARGE SMF RECORD BUFFER
		..1. ....		SMFQUED	"B'00100000" Buffer is queued to HASPACCT subtask
5	(5)	CHARACTER	1	SMFWFL26	RECORD 26 WRITE FLAG
5	(5)	X'1'	0	SMFNO26	"1" DO NOT WRITE SMF RECORD
6	(6)	CHARACTER	1	SMFCLFLG	CLASS SMF AFFINITY
		.... ...1		SMFAPPC	"X'01" System affinity for transaction programs
7	(7)	CHARACTER	1	SMFPARM	RESERVED
7	(7)	X'8'	0	SMFLNHDR	** -SMF" LENGTH OF JES2 BUFFER HEADER



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>THE FOLLOWING ARE COMMON FIELD NAMES USED FOR MULTIPLE RECORDS            EACH INDIVIDUAL RECORD HAS A RECORD SPECIFIC NAME FOR EACH FIELD            SUCH AS SMFNNXXX WHERE, NN = RECORD NUMBER, AND XXX = FIELD NAME</p>					
End of Comment					
8	(8)	CHARACTER	4	SMFJMRCH (0)	POINTER TO PURGE REC BUFFER
8	(8)	CHARACTER	4	SMFRDW (0)	SMF RECORD DESCRIPTOR WORD
8	(8)	CHARACTER	2	SMFLEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMFSEG	SEGMENT DESCRIPTOR
Comment					
<p>BEGINNING OF JMR OR HASP SMF RECORD</p>					
End of Comment					
10	(A)	X'C'	0	SMFJMR	*** JMR DATA AREA
Comment					
<pre> %IFABGN1 ; METHOD OF ACCESS PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE INCLUDE MACRO FROM LIBRARY EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP- DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON %INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD A DIAGNOSTIC.  THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER RECORDS AS FOLLOWS: MACRO RECORDS IFASMFR1 07-19 IFASMFR2 20-27 IFASMFR3 28-36 IFASMFR4 37-46 IFASMFR5 47-54 IFASMFR6 55-69 IFASMFR9 80-84 IFASMFR8 85-103 IFASMFRB 104-113 IFASMFRD 114-123 IFASMFRD 124-127  %GOTO IFABGN2; THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS IFASMFR &amp;RECTYPE NOTE: VALUES FOR &amp;RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1). IN JES2, THIS RECORD IS WRITTEN FOR EACH JOB OUTPUT ELEMENT, WHICH REPRESENTS A GROUP OF DS DIFFERENTIATED BY PUNCH OR PRINTER SETUP &amp; TYPE OF OUTPUT(EG HELD VS NON-HELD). FOR JES3, WRITTEN FOR EACH COPY OF A DATA SET           </pre>					
End of Comment					
8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDARY
8	(8)	X'8'	0	SMFRCD6	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF6LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF6SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF6FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF6RTY	RECORD TYPE 6
13	(D)	X'6'	0	SMFJ6	"6" PRINT/PUNCH RECORD TYPE

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
14	(E)	BITSTRING	4	SMF6TME	TOD, USING FORMAT FROM TIME MACRO W/BIN. INTVL
18	(12)		4	SMF6DTE	DATE IN PACKED DECIMAL FORM: 00YYDDDF
22	(16)	CHARACTER	4	SMF6SID	SYSTEM IDENTIFICATION Y02901
26	(1A)	CHARACTER	8	SMF6JBN	JOB NAME
34	(22)	BITSTRING	4	SMF6RST	RDR START TIME, TIME JOB CARD 1ST READ
38	(26)		4	SMF6RSD	READER START DATE 00YYDDDF
42	(2A)	CHARACTER	8	SMF6UIF	USER ID FIELD
50	(32)	CHARACTER	1	SMF6OWC	OUTPUT WTR CLASS, BLANK FOR NON-SYSOUT
51	(33)	BITSTRING	4	SMF6WST	WRITER START TIME
55	(37)		4	SMF6WSD	WRITER START DATE
59	(3B)	BITSTRING	4	SMF6NLR	# OF LOGICAL RECORDS HANDLED BY WRITER PER FORM # PER CLASS, INCLUDES REPEATS AND RESTARTS.
63	(3F)	BITSTRING	1	SMF6IOE	IO ERROR INDICATOR: BITS 0-4 RESERVED Y02120
		.... .1..		SMF6DIE	"X'04" 5 - DATA INPUT ERROR 6 - RESV Y02120
		.... ...1		SMFCBIE	"X'01" 7 - CONTROL BLOCK INPUT ERROR
64	(40)	BITSTRING	1	SMF6NDS	# OF DATA SETS PROCESSED BY THE OUTPUT Y02120 WRITER AND INCLUDED IN THIS RECORD. Y02120 (COUNT FOR EACH TIME A DS IS PRINTED) Y02120 DOES NOT INCLUDE RESTARTS.
65	(41)	CHARACTER	4	SMF6FMN	FORM NUMBER
69	(45)	BITSTRING	1	SMF6PAD1	STATUS INDICATORS - THE SECTIONS WILL BE IN THE ORDER LISTED BELOW WHEN THE BIT IS TURNED ON BIT MEANING
		1... ....		SMF6FEXT	"X'80" 0 1 - FIRST EXTENSION PRESENT
		.1.. ....		SMF6REXT	"X'40" 1 1 - COMMON SECTION PRESENT
		..1. ....		SMF6SEXT	"X'20" 2 1 - SECOND EXTENSION PRESENT
		...1 ....		SMF6ESS1	"X'10" 3 1 - ENHANCED SYSOUT SECTION PRESENT
		.... 1...		SMF6FTFR	"X'08" 4 1 - FILE TRANSFER SECTION PRESENT 5-7 RESERVED
70	(46)	BITSTRING	2	SMF6SBS	SUBSYSTEM GENERATING ID EXTWTR=0, JES2=2, JES3=5, PSF=7, IP PrintWay = 9
72	(48)	BITSTRING	2	SMF6LN1	LENGTH OF SECTION INCLUDING THIS FIELD
74	(4A)	BITSTRING	1	SMF6DCI	DS CONTROL INDICATORS FOR DATA GROUP
		1... ....		SMF6DCRV	"X'80" 0 - RESERVED
		.1.. ....		SMF6SDS	"X'40" 1 - SPUN OFF DS
		..1. ....		SMF6OCN	"X'20" 2 - TERMINATED BY OPERATOR
		...1 ....		SMF6ORD	"X'10" 3 - INTERRUPTED BY OPERATOR (JES2) OPERATOR RESTARTED DATA SET WITH DESTINATION (JES3)
		.... 1...		SMF6OR	"X'08" 4 - RESTARTED BY OPERATOR
		.... .1..		SMF6ROR	"X'04" 5 - CONT OF INTERRUPTED GROUP (JES2) RECEIVED OP RESTARTED DS(JES3)
		.... ..1.		SMF6OSS	"X'02" 6 - CARRIAGE OVERRIDEN BY OPER(JES2) OPERATOR STARTED WITH SINGLE SPACE(JES3)
		.... ...1		SMF6INT	"X'01" 7 - PUNCH WAS INTERPRETED
75	(4B)	BITSTRING	1	SMF6INDC	INDICATOR BITS BITS 0-3 ARE RESERVED FOR FUTURE EXPANSION OF DATASET CONTROL INDICATORS BITS 4-7 ARE RECORD LEVEL INDICATORS IN BIT VALUE FORMAT. EXAMPLE: LEVEL 1=X'01' LEVEL 12=X'0C' LEVEL 15=X'0F' THIS NUMBER WILL BE INCREMENTED BY 1 EACH TIME A NEW RELEASE CHANGES THE RECORD
		.... ...1		SMF6LEV2	"X'01" THIS VARIABLE IS FOR JES2 TO SET THE LEVEL INDICATOR BITS.
		.... ..11		SMF6J2L3	"X'03" THIS VARIABLE IS FOR JES2 TO SET THE LEVEL INDICATOR BITS.
		.... .1..		SMF6J2L4	"X'04" THIS VARIABLE IS FOR JES2 TO SET THE LEVEL INDICATOR BITS FOR SECURITY SUPPORT
		.... ...1		SMF6LEV3	"X'01" THIS VARIABLE IS FOR JES3 TO SET THE LEVEL INDICATOR BITS.
		.... ..11		SMF6J3L3	"X'03" THIS VARIABLE IS FOR JES3 TO SET THE LEVEL INDICATOR BITS.
		.... .1..		SMF6J3L4	"X'04" THIS VARIABLE IS FOR JES3 TO SET THE LEVEL INDICATOR BITS FOR SECURITY SUPPORT INDICATOR BITS.
		.... .1.1		SMF6LEV4	"X'05" MVS/JES2 RELEASE 4.1.0

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		.... .11.		SMF6LEV6	"X'06" PSF/MVS RELEASE 3.1.0
		.... .111		SMF6LEV7	"X'07" Z/OS RELEASE V1R5
76	(4C)	CHARACTER	4	SMF6JNM	WHEN SMF6INDC CONTAINS A X'1', THIS FIELD CONTAINS A FOUR-DIGIT EBCDIC JOB NUMBER. WHEN SMF6INDC CONTAINS A X'3' OR GREATER, AND THE JOB NUMBER HAS MORE THAN 4 DIGITS, THIS FIELD CONTAINS ZEROS. IF THE JOB NUMBER IS < OR = TO 9999, THIS FIELD CONTAINS THE JOB NUMBER. FOR AN APPC TRANSACTION, THIS FIELD CONTAINS ZEROES. THE CORRECT JOB NUMBER OR APPC TRANSACTION ID IS FOUND IN SMF6JBID.
80	(50)	CHARACTER	8	SMF6OUT	LOGICAL OUTPUT DEVICE NAME FOR THE 3820, ACF/VTAM LOGICAL UNIT NAME
88	(58)	CHARACTER	4	SMF6FCB	FCB ID Y02120
92	(5C)	CHARACTER	4	SMF6UCS	UCS ID Y02120 END OF RECORD FOR EXTERNAL WTR
96	(60)	BITSTRING	4	SMF6PGE	APPROXIMATE PHYSICAL PAGE COUNT
96	(60)	X'64'	0	SMF6J2S	*** BEGIN JES2 ONLY SECTION
100	(64)	BITSTRING	2	SMF6RTE	OUTPUT ROUTE CODE OR ZERO
102	(66)	BITSTRING	1	SMF6END2 (0)	END OF JES2 RECORD
102	(66)	BITSTRING	0	SMF6SIZ2 (0)	SIZE OF JES2 SMF6 RECORD EXCLUDING OPTIONAL EXTENSIONS
102	(66)	BITSTRING	0	SMF6SIZ3 (0)	SIZE OF JES2 SMF6 RECORD FROM SMF6LN1 TO HERE
100	(64)	X'64'	0	SMF6J3S	*** BEGIN JES3 ONLY SECTION
100	(64)	BITSTRING	2	SMF6DFE	DATA FORMAT ERROR INDICATORS BITS 0-5 RESV
		.... ..1.		SMF6CCE	"X'02" 6 - SOME 1ST CHAR CONTROL DATA BAD, DEFAULT USED
		.... ..1		SMF6RBE	"X'01" 7 - BAD RECORD LENGTH(TRUNCATE OR PAD) 8-15 RESV
102	(66)	BITSTRING	2	SMF6OPR	OUTPUT PRIORITY
104	(68)	CHARACTER	8	SMF6GRP	LOGICAL OUTPUT DEVICE GROUP NAME
112	(70)	CHARACTER	8	SMF6RSVJ	RESERVED FOR JES3
120	(78)	CHARACTER	4	SMF6RSVU	RESERVED FOR USER
124	(7C)	BITSTRING	1	SMF6END (0)	END OF JES3 RECORD
124	(7C)	BITSTRING	0	SMF6SIZ (0)	SIZE OF JES3 SMF6 RECORD EXCLUDING OPTIONAL EXTENSIONS
124	(7C)	BITSTRING	1	SMF6LSIZ (0)	SIZE OF JES3 SMF6 RECORD FROM SMF6LN1 TO HERE

Comment

FIRST EXTENSION - NON-IMPACT PRINTING SUBSYSTEM SECTION  
 THIS SECTION WILL ONLY BE PRESENT WHEN  
 SMF6SBS IS SET TO 2, 5 OR 7 INDICATING THAT  
 JES2, JES3 OR PSF HAS GENERATED THIS RECORD

End of Comment

72	(48)	BITSTRING	2	SMF6LN2	LENGTH FIRST EXTENSION INCLUDING THIS FLD
74	(4A)	CHARACTER	1	SMF6CPS (8)	COPIES DISTRIBUTION
82	(52)	CHARACTER	4	SMF6CHR (4)	TRANSLATE TABLE NAMES FRO CHARS PARM
98	(62)	CHARACTER	4	SMF6MID	COPY MODIFICATION MODULE NAME
102	(66)	CHARACTER	4	SMF6FLI	FLASH OVERLAY NAME
106	(6A)	BITSTRING	1	SMF6FLC	NUMBER OF COPIES FLASHED
107	(6B)	BITSTRING	1	SMF6BID	FLAG BYTE
		1... ....		SMF6BTS	"X'80" THE BTSS WAS USED FOR OUTPUT
		..1. ....		SMF6OPJ	"X'40" OPTCD=J WAS USED FOR OUTPUT
		..1. ....		SMF6CSP	"X'20" CUT SHEET PRINTER
108	(6C)	BITSTRING	1	SMF6FEND (0)	END OF FIRST EXTENSION
108	(6C)	BITSTRING	1	SMF6FSIZ (0)	SIZE OF FIRST EXTENSION

# \$SMF Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
COMMON SECTION - THIS SECTION IS AN EXTENSION OF THE FIXED HEADER SECTION AND WILL BE WRITTEN BY ALL GENERATORS OF THE TYPE 6 RECORD. THIS WAS PREVIOUSLY CALLED THE ROUTING SECTION.					
End of Comment					
72	(48)	BITSTRING	2	SMF6LN3	LENGTH OF SECTION INCLUDING THIS FIELD
74	(4A)	CHARACTER	4	SMF6ROUT	OUTPUT ROUTE CODE
78	(4E)	CHARACTER	8	SMF6EFMN	OUTPUT FORM NUMBER
86	(56)	BITSTRING	1	SMF6REND (0)	END OF OLD ROUTING SECTION
86	(56)	BITSTRING	0	SMF6RSIZ (0)	SIZE OF OLD ROUTING SECTION
86	(56)	CHARACTER	16		RESERVED
102	(66)	CHARACTER	8	SMF6JBID	JOB ID
110	(6E)	CHARACTER	8	SMF6STNM	STEPNAME
118	(76)	CHARACTER	8	SMF6PRNM	PROCEDURE STEP NAME
126	(7E)	CHARACTER	8	SMF6DDNM	DD NAME
134	(86)	CHARACTER	8	SMF6USID	USER ID
142	(8E)	CHARACTER	8	SMF6SECS	SECURITY LABEL (SECLABEL)
150	(96)	CHARACTER	8	SMF6PRMD	PROCESSING MODE
158	(9E)	CHARACTER	53	SMF6DSNM	DATA SET RESOURCE NAME
211	(D3)	CHARACTER	3		RESERVED
214	(D6)	CHARACTER	20	SMF6OTOK	OUTPUT GROUP TOKEN
234	(EA)	BITSTRING	1	SMF6DEND (0)	END OF ROUTING SECTION
234	(EA)	BITSTRING	1	SMF6DSIZ (0)	SIZE OF ROUTING SECTION

## Comment

### ENHANCED SYSOUT SECTION

End of Comment					
72	(48)	BITSTRING	2	SMF6LN5	LENGTH ENHANCED SYSOUT SECTION INCLUDING THIS FIELD
74	(4A)	BITSTRING	4	SMF6SGID	SEGMENT IDENTIFIER
78	(4E)	BITSTRING	1	SMF6IND	SECTION INDICATOR
		1... ..		SMF6SJF	"X'80" ERROR OBTAINING SWBTU - SWBTU DATA AREA NOT PRESENT
79	(4F)	BITSTRING	1	SMF6RSV	RESERVED
80	(50)	CHARACTER	8	SMF6JDVT	JDVTNAME
88	(58)	BITSTRING	2	SMF6TUL	SWBTU DATA AREA LENGTH
90	(5A)	CHARACTER	1	SMF6TU (0)	SWBTU DATA AREA - DATA AREA CAN BE PROCESSED USING SWBTUREQ MACRO
90	(5A)	BITSTRING	1	SMF6EEND (0)	END OF ENHANCED SYSOUT SECTION
90	(5A)	BITSTRING	1	SMF6ESIZ (0)	SIZE OF ENHANCED SYSOUT SEC. MOVED SMF6LN4 TO AOPSMF6 2 MOVED SMF6BNLN TO AOPSMF6 2 MOVED SMF6BNNO TO AOPSMF6 4 MOVED SMF6LN6 TO AOPSMF6 11

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
%AOPBGN1: ; METHOD OF ACCESS PLAS: %INCLUDE SYSLIB(AOPSMF6) ASSEMBLER: AOPSMF6 NOTES: PL/AS - INCLUDED BY IFASMFR BAL - CALLED FROM IFASMFR					
%GOTO AOPBGN2; THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF PORTIONS OF THE SMF TYPE 6 RECORD. THE SECTIONS ARE: SECOND EXTENSION - APA SECTION - WRITTEN BY PSF (SMF6SBS=7) MULTI-BINS HEADER SECTION - WRITTEN BY PSF (SMF6SBS=7) MULTI-BINS COUNTER SECTION - WRITTEN BY PSF (SMF6SBS=7) FILE TRANSFER SECTION - WRITTEN BY IP PRINTWAY (SMF6SBS=9) SECOND EXTENSION - APA (ALL POINTS ADDRESSABLE) PRINTING SUBSYSTEM SECTION THIS SECTION WILL ONLY BE PRESENT WHEN SMF6SBS IS SET TO 7 INDICATING THAT PSF HAS GENERATED THIS RECORD					
End of Comment					
72	(48)	BITSTRING	2	SMF6LN4	LENGTH SECOND EXTENSION INCLUDING THIS FLD
74	(4A)	BITSTRING	2	SMF6BNOF	OFFSET TO BIN SECTION
74	(4A)	BITSTRING	2	SMF6RES	RESERVED - REDEFINES SMF6BNOF
76	(4C)	BITSTRING	4	SMF6FONT	NUMBER OF FONTS USED
80	(50)	BITSTRING	4	SMF6LFNT	NUMBER OF FONTS LOADED
84	(54)	BITSTRING	4	SMF6OVLY	NUMBER OF OVERLAYS USED
88	(58)	BITSTRING	4	SMF6LOLY	NUMBER OF OVERLAYS LOADED
92	(5C)	BITSTRING	4	SMF6PGSG	NUMBER OF PAGE SEGMENTS USED
96	(60)	BITSTRING	4	SMF6LPSG	NUMBER OF PAGE SEGMENTS LOADED
100	(64)	BITSTRING	4	SMF6IMPS	COUNT OF LOGICAL IMPRESSIONS PROCESSED
104	(68)	BITSTRING	4	SMF6FEET	NUMBER OF FEET OF DOCUMENT PRINTED (ZERO FOR THE 3820)
108	(6C)	BITSTRING	4	SMF6PGDF	NUMBER OF PAGEDEFS USED
112	(70)	BITSTRING	4	SMF6FMDF	NUMBER OF FORMDEFS USED
116	(74)	BITSTRING	1	SMF6BIN	FLAG BYTE
		1... ....		SMF6BIN1	"X'80" BIN1 WAS USED FOR ANY PART OF THE DATA SET
		.1.. ....		SMF6BIN2	"X'40" BIN2 WAS USED FOR ANY PART OF THE DATA SET
		..1. ....		SMF6BIN3	"X'20" BIN3 WAS USED FOR ANY PART OF THE DATA SET
		...1 ....		SMF6BIN4	"X'10" BIN4 WAS USED FOR ANY PART OF THE DATA SET
117	(75)	BITSTRING	1	SMF6PGOP	FLAG BYTE
		1... ....		SMF6DUPS	"X'80" STNDARD DUPLEX WAS USED FOR ANY PART OF DS
		.1.. ....		SMF6DUPT	"X'40" TUMBLE DUPLEX WAS USED FOR ANY PART OF DS
		..1. ....		SMF6SYSA	"X'20" KEYWORD SYSAREA=Y
		...1 ....		SMF6DPGL	"X'10" KEYWORD DPAGELBL=Y
		.... 1..		SMF6SUCC	"X'08" PRINT OPERATION WAS SUCCESSFUL
		.... .1..		SMF6SPGL	"X'04" KEYWORD SPAGELBL=Y
		.... ..1.		SMF6SOER	"X'02" ERROR OCCURRED PROCESSING SECURITY OVERLAY
		.... ...1		SMF6IGER	"X'01" IMAGE GENERATOR OVERRUN ERROR OCCURRED
118	(76)	BITSTRING	1	SMF6FLG3	FLAG BYTE
		1... ....		SMF6SLIG	"X'80" SECURITY LABEL INTEGRITY GUARANTEED
		.1.. ....		SMF6JHPP	"X'40" THE JOB HEADER PAGE WAS PRINTED
		..1. ....		SMF6JTPP	"X'20" THE JOB TRAILER PAGE WAS PRINTED
		...1 ....		SMF6DPLS	"X'10" DATA PAGE LABELING WAS SUPPRESSED
		.... 1..		SMF6UPAS	"X'08" USER PRINTABLE AREA WAS SUPPRESSED
119	(77)	BITSTRING	1	SMF6APAL	LEVEL INDICATOR FOR APA SECTION
		.... ...1		SMF6APA1	"X'01" INITIAL LEVEL OF APA SECTION
120	(78)	BITSTRING	4	SMF6NSOL	NUMBER OF SECURITY OVERLAYS USED
124	(7C)	BITSTRING	4	SMF6NSFO	NUMBER OF SECURITY FONTS USED

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
128	(80)	BITSTRING	4	SMF6NSPS	NUMBER OF SECURITY PAGE SEGMENTS USED
132	(84)	CHARACTER	8	SMF6FDNM	FORMDEF NAME
140	(8C)	CHARACTER	8	SMF6PDNM	PAGEDEF NAME
148	(94)	CHARACTER	8	SMF6PTDV	PRINTDEV NAME
156	(9C)	CHARACTER	32	SMF6OCNM	OBJECT CONTAINER NAME(S)
156	(9C)	CHARACTER	8	SMF6SETU	COMSETUP OBJECT CONTAINER NAME
164	(A4)	CHARACTER	8		RESERVED OBJECT CONTAINER NAME
172	(AC)	CHARACTER	8		RESERVED OBJECT CONTAINER NAME
180	(B4)	CHARACTER	8		RESERVED OBJECT CONTAINER NAME
188	(BC)	BITSTRING	4	SMF6LPGE	Count of logical pages processed
192	(C0)	BITSTRING	1	SMF6SEND (0)	END OF SECOND EXTENSION
192	(C0)	BITSTRING	1	SMF6SSIZ (0)	SIZE OF SECOND EXTENSION

Comment

MULTI-BINS HEADER SECTION (OFFSET DEFINED BY SMF6BNOF)

End of Comment

8	(8)	BITSTRING	2	SMF6BNLN	LENGTH BINS SECTION INCLUDING THIS FLD
10	(A)	BITSTRING	2	SMF6BNUM	NUMBER OF COUNTERS ENTRIES

Comment

MULTI-BINS COUNTER SECTION  
- FOLLOWS "MULTI-BIN" HEADER SECTION

End of Comment

8	(8)	BITSTRING	1	SMF6BNNO	BIN NUMBER
9	(9)	BITSTRING	3	SMF6BNCT	BIN COUNTER
12	(C)	BITSTRING	2	SMF6BNLE	Paper length in millimeters
14	(E)	BITSTRING	2	SMF6BNWI	Paper width in millimeters

Comment

FILE TRANSFER SECTION

End of Comment

72	(48)	BITSTRING	2	SMF6LN6	LENGTH OF FILE TRANSFER SECTION INCLUDING THIS FIELD
74	(4A)	BITSTRING	4	SMF6BYTE	TOTAL NUMBER OF BYTES SENT
78	(4E)	BITSTRING	1	SMF6IP1	1ST SEGMENT OF TARGET ADDRESS
79	(4F)	BITSTRING	1	SMF6IP2	2ND SEGMENT OF TARGET ADDRESS
80	(50)	BITSTRING	1	SMF6IP3	3RD SEGMENT OF TARGET ADDRESS
81	(51)	BITSTRING	1	SMF6IP4	4TH SEGMENT OF TARGET ADDRESS
82	(52)	BITSTRING	1	SMF6FTL	LEVEL INDICATOR FOR FILE TRANSFER SECTION
		.... ...1		SMF6FTL1	"X'01" Z/OS V1R5
83	(53)	CHARACTER	9		RESERVED
92	(5C)	BITSTRING	2	SMF6URIL	Length of Host URI
94	(5E)	BITSTRING	2	SMF6PQLN	Length of Print Queue Name
96	(60)	CHARACTER	24	SMF6PRTQ	Print Queue Name
120	(78)	CHARACTER	1	SMF6URI (0)	Target Device URI
120	(78)	BITSTRING	1	SMF6TEND (0)	END OF FILE TRANSFER SECTION
120	(78)	BITSTRING	0	SMF6TSIZ (0)	SIZE OF FILE TRANSFER SECTION

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	SMF	, DSECT may be destroyed

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description

Comment

%IFABGN1: ;  
 METHOD OF ACCESS  
 PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM  
 DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE  
 INCLUDE MACRO FROM LIBRARY  
 EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-  
 DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON  
 %INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD  
 A DIAGNOSTIC.  
 THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT  
 ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER  
 RECORDS AS FOLLOWS:  
 MACRO RECORDS  
 IFASMFR1 07-19  
 IFASMFR2 20-27  
 IFASMFR3 28-36  
 IFASMFR4 37-46  
 IFASMFR5 47-54  
 IFASMFR6 55-69  
 IFASMFR9 80-84  
 IFASMFR8 85-103  
 IFASMFRB 104-113  
 IFASMFRC 114-123  
 IFASMFRD 124-127

%GOTO IFABGN2;  
 THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE  
 REQUIRED FORMAT IS  
 IFASMFR &RECTYPE  
 NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).  
 MODULE NAME = IAZSMF24  
 DESCRIPTIVE NAME = JES SMF SPOOL OFFLOAD RECORD  
 SWITCH TO DECIDE WHETHER TO GENERATE EQUATES FOR WRITING SMF RECORDS

End of Comment					
8	(8)	X'8'	0	SMFRCD24	*** START OF RECORD
8	(8)	X'8'	0	SMF24PTR	*** HEADER LENGTH
8	(8)	CHARACTER	2	SMF24LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF24SEG	SEGMENT DESCRIPTOR

Comment

BEGINNING OF JMR OR HASP SMF RECORD

End of Comment					
12	(C)	BITSTRING	1	SMF24FLG	HEADER FLAG BYTE
		.1.. ....		SMF24STS	"B'01000000" SUBTYPES USED
13	(D)	BITSTRING	1	SMF24RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF24TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF24DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF24SID	SYSTEM IDENTIFICATION

Comment

HEADER FOR HASP SUBSYS RECORD TYPE 24

End of Comment					
22	(16)	X'18'	0	SMFJ24	"24" SPOOL OFFLOAD RECORD TYPE
26	(1A)	BITSTRING	4	SMF24SSI	SUBSYSTEM ID
30	(1E)	BITSTRING	2	SMF24SUB	RECORD SUBTYPE
30	(1E)	X'1'	0	SMF24JT	"1" JOB TRANSMITTER
30	(1E)	X'2'	0	SMF24JR	"2" JOB RECEIVER

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
30	(1E)	X'3'	0	SMF24ST	"3" SYSOUT TRANSMITTER
30	(1E)	X'4'	0	SMF24SR	"4" SYSOUT RECEIVER
32	(20)	BITSTRING	2	SMF24NTR	NUMBER OF TRIPLETS
34	(22)	BITSTRING	2	SMF24RSV	RESERVED
34	(22)	X'1C'	0	SMF24LHD	"*-SMFRCD24" LEN OF HEADER SECTION
34	(22)	X'24'	0	SMF24TPS	*** BEGINNING OF TRIPLETS
36	(24)	BITSTRING	4	SMF24OPS	OFFSET TO PRODUCT SECTION
40	(28)	BITSTRING	2	SMF24LPS	LENGTH OF PRODUCT SECTION
42	(2A)	BITSTRING	2	SMF24NPS	NUMBER OF PRODUCT SECTIONS
44	(2C)	BITSTRING	4	SMF24OGN	OFFSET TO GENERAL SECTION
48	(30)	BITSTRING	2	SMF24LGN	LENGTH OF GENERAL SECTION
50	(32)	BITSTRING	2	SMF24NGN	NUMBER OF GENERAL SECTIONS
52	(34)	BITSTRING	4	SMF24OSP	OFFSET TO SPOF SECTION
56	(38)	BITSTRING	2	SMF24LSP	LENGTH OF SPOF SECTION
58	(3A)	BITSTRING	2	SMF24NSP	NUMBER OF SPOF SECTIONS
60	(3C)	BITSTRING	4	SMF24OSW	OFFSET TO ESS SECTION
64	(40)	BITSTRING	2	SMF24LSW	LENGTH OF ESS SECTION
66	(42)	BITSTRING	2	SMF24NSW	NUMBER OF ESS SECTIONS
68	(44)	BITSTRING	4	SMF24OSA	Offset to sysaff section
72	(48)	BITSTRING	2	SMF24LSA	Length of sysaff section
74	(4A)	BITSTRING	2	SMF24NSA	Number of sysaff sections
74	(4A)	X'28'	0	SMF24TRP	"*-SMF24TPS" LENGTH OF TRIPLETS
74	(4A)	X'5'	0	SMF24NTP	"SMF24TRP/8" NUMBER OF TRIPLETS

Comment

BEGINNING OF JES2 PRODUCT SECTION

End of Comment

76	(4C)	CHARACTER	2	SMF24PVR	RECORD VERSION
78	(4E)	CHARACTER	8	SMF24PNM	PRODUCT NAME
86	(56)	BITSTRING	2	SMF24RS2	RESERVED

Comment

GENERAL SECTION FOR SPOOL OFFLOAD DEVICES

End of Comment

88	(58)	BITSTRING	2	SMF24GLN	LENGTH OF GENERAL SECTION
90	(5A)	BITSTRING	1	SMF24BCF	BUFFER CONTINUATION FLAG
		1... ....		SMF24FST	"B'10000000" FIRST SMF BUFFER FOR JOB
		.1.. ....		SMF24CON	"B'01000000" SMF BUFFER CONTINUED
		..1. ....		SMF24LST	"B'00100000" LAST SMF BUFFER - END OF JOB
91	(5B)	BITSTRING	1	SMF24EOJ	END OF JOB FLAG
		1... ....		SMF24COM	"B'10000000" JOB COMPLETELY OFFLOADED
		.1.. ....		SMF24SDS	"B'01000000" JOB COMPLETED WITH SKIPPED DATA SETS
		..1. ....		SMF24INJ	"B'00100000" INCOMPLETE JOB OFFLOADED
		...1 ....		SMF24OPR	"B'00010000" OPERATOR CANCELED JOB
92	(5C)	CHARACTER	8	SMF24JBN	JOB NAME
100	(64)	CHARACTER	8	SMF24JID	ORIGINAL JOB IDENTIFICATION
108	(6C)	CHARACTER	8	SMF24CJD	CURRENT JOB IDENTIFICATION
116	(74)	CHARACTER	4	SMF24SYS	SYSTEM ID
120	(78)	CHARACTER	44	SMF24DSN	OFFLOAD DATA SET NAME
164	(A4)	BITSTRING	4	SMF24CNT	NUMBER OF RECORDS DUMPED/LOADED
168	(A8)	BITSTRING	4	SMF24TDS	TIME OFFLOAD DATA SET ALLOCATED
172	(AC)	BITSTRING	4	SMF24DDS	DATE OFFLOAD DATA SET ALLOCATED
176	(B0)	CHARACTER	8	SMF24ORG	ORIGIN NODE
184	(B8)	BITSTRING	4	SMF24TRD	TIME ON READER
188	(BC)	BITSTRING	4	SMF24DRD	DATE ON READER



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

EITHER THE JOB SECTION OR THE SYSOUT SECTION IS WRITTEN,  
NOT BOTH. THE SPOF TRIPLET REFERS TO WHICHEVER ONE IS WRITTEN  
IN THE CURRENT RECORD.  
JOB SELECTION CRITERIA SECTION

End of Comment

192	(C0)	BITSTRING	2	SMF24LN1	LENGTH OF JOB SECTION
194	(C2)	BITSTRING	1	SMF24JFG	JOB FLAGS
		1... ....		SMF24JHL	"B'10000000" HELD JOB
		.1... ....		SMF24AFF	"B'01000000" AFFINITY = ANY
195	(C3)	CHARACTER	1	SMF24JCL	JOB CLASS
196	(C4)	SIGNED	4	SMF24JRT (0)	ROUTE CODE
196	(C4)	CHARACTER	8	SMF24JND	NODE NAME
204	(CC)	CHARACTER	28	SMF24JAF	AFFINITY SYSTEM ID'S
232	(E8)	CHARACTER	8	SMF248CL	8 CHAR JOB CLASS
240	(F0)	CHARACTER	1	SMF24EJS (0)	End of job selection

Comment

SYSOUT SELECTION CRITERIA SECTION

End of Comment

192	(C0)	BITSTRING	2	SMF24LN2	LENGTH OF SYSOUT SECTION
194	(C2)	BITSTRING	1	SMF24SFG	SYSOUT FLAGS
		1... ....		SMF24SHL	"B'10000000" HELD SYSOUT
		.1.. ....		SMF24SBT	"B'01000000" BURSTED SYSOUT
		..1. ....		SMF24SJH	"B'00100000" HELD JOB
		...1 ....		SMF24INC	"B'00010000" INCOMPLETE DATA SET
		... 1...		SMF24MUL	"B'00001000" MULTI-DEST DATA SET
195	(C3)	CHARACTER	1	SMF24SCL	SYSOUT CLASS
196	(C4)	SIGNED	4	SMF24SRT (0)	ROUTE CODE
196	(C4)	CHARACTER	8	SMF24SND	NODE NAME
204	(CC)	CHARACTER	8	SMF24SRN	REMOTE NAME
212	(D4)	CHARACTER	4	SMF24FCB	FCB
216	(D8)	CHARACTER	8	SMF24FOR	FORMS
224	(E0)	CHARACTER	4	SMF24FLS	FLASH
228	(E4)	CHARACTER	8	SMF24PRM	PR MODE
236	(EC)	CHARACTER	4	SMF24UCS	UCS
240	(F0)	CHARACTER	8	SMF24WID	WRITER
248	(F8)	BITSTRING	4	SMF24REC	DATA SET RECORD COUNT
252	(FC)	BITSTRING	1	SMF24PRY	PRIORITY
252	(FC)	X'C'	0	SMF24PRD	"SMF24RS2+L'SMF24RS2-SMF24PVR" LEN OF PRODUCT SEC
252	(FC)	X'68'	0	SMF24LN	"SMF24DRD+L'SMF24DRD-SMF24GLN" LEN OF GENERAL SEC
252	(FC)	X'30'	0	SMF24L1	"SMF248CL+L'SMF248CL-SMF24LN1" LEN OF JOB SEL SEC
252	(FC)	X'3D'	0	SMF24L2	"SMF24PRY+L'SMF24PRY-SMF24LN2" LEN OF SYSOUT SEC
252	(FC)	X'44'	0	SMF24POF	"SMF24LHD+SMF24TRP" OFFSET TO PRODUCT SECTION
252	(FC)	X'50'	0	SMF24GOF	"SMF24POF+SMF24PRD" OFFSET TO GENERAL SECTION
252	(FC)	X'B8'	0	SMF24SOF	"SMF24GOF+SMF24LN" OFFSET TO SPOF SECTION
252	(FC)	X'E8'	0	SMF24AOF	"SMF24SOF+SMF24L1" Offset to aff section

## \$SMF Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
Enhanced SYSOUT Support (ESS) Ssection This section contains the OUTPUT descriptor (if any) in SWBTU format (IEFSJPFx plus text units) for the first offloaded data set included in this SMF record. The SWBTU may be processed using the SWBTUREC macro or other Scheduler JCL Facility (SJF) services.					
End of Comment					
253	(FD)	BITSTRING	2	SMF24LN3	LENGTH OF ESS SECTION
255	(FF)	BITSTRING	4	SMF24SGT	SEGMENT IDENTIFIER
259	(103)	BITSTRING	1	SMF24IND	ESS SECTION INDICATOR
		1... ..		SMF24SJV	"B'10000000" ERROR OBTAINING SWBTU (SWBTU DATA NOT PRESENT)
260	(104)	BITSTRING	1		RESERVED
261	(105)	CHARACTER	8	SMF24JDT	JDVT NAME
269	(10D)	BITSTRING	2	SMF24TUL	SWBTU DATA AREA LENGTH
271	(10F)	CHARACTER	1	SMF24TU (0)	SWBTU DATA AREA
271	(10F)	X'12'	0	SMF24ESL	**SMF24LN3" Length of the fixed portion of the ESS section

Comment

Enhanced SYSTEM AFFINITY suport section.  
This section contains the system names for all the systems for which this job has affinity. The one exception is if it has an affinity of ANY in which case the flag bit SMF24AFF is on.

End of Comment

240	(F0)	BITSTRING	2	SMF24LS4	Length of sysaff section
242	(F2)	BITSTRING	2		Reserved for future IBM use
244	(F4)	BITSTRING	4	SMF24SAN	Number of system affinities
248	(F8)	BITSTRING	4	SMF24LN4	Length of system name
252	(FC)	CHARACTER	1	SMF24SAC (0)	Start of system aff. names
252	(FC)	X'C'	0	SMF24SAL	**SMF24LS4" Length of the fixed portion of the SYS. AFF SECTION

Comment

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
```

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMF6 55-69			
		IFASMF9 80-84			
		IFASMFRA 85-103			
		IFASMF8 104-113			
		IFASMFRC 114-123			
		IFASMF8 124-127			

%GOTO IFABGN2;

THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS

IFASMF &RECTYPE

NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

%IAZPRO26; ;

MODULE NAME = IAZSMF26

DESCRIPTIVE NAME = JES SMF PURGE RECORD

%GOTO IAZ26;

SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC

THIS RECORD IS WRITTEN WHEN A JOB IS READY TO BE PURGED FOR

BOTH FOREGROUND AND BACKGROUND JOBS IN THE SYSTEM.

End of Comment

8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD26	*** START OF RECORD
8	(8)	X'8'	0	SMF26PTR	*** HEADER SEGMENT (LGTH 46 WITHOUT RDW)
8	(8)	BITSTRING	2	SMF26LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF26SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF26FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF26RTY	RECORD TYPE 26
13	(D)	X'1A'	0	SMFJ26	"26" PURGE RECORD TYPE
14	(E)	BITSTRING	4	SMF26TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF26DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF26SID	SYSTEM INDICATOR
26	(1A)	CHARACTER	8	SMF26JBN	JOB NAME
34	(22)	BITSTRING	4	SMF26RST	RDR START TIME, TIME JOB CARD 1ST READ
38	(26)		4	SMF26RSD	READER START DATE
42	(2A)	CHARACTER	8	SMF26UIF	USER IDENTIFICATION FIELD
50	(32)	BITSTRING	4	SMF26RSV	RESV
54	(36)	BITSTRING	2	SMF26SBS	SUBSYSTEM GENERATING ID(JES2=2, JES3=5)
		.... .1.		SMF26HSP	"X'0002" JES2 ID
		.... .1.1		SMF26ASP	"X'0005" JES3 ID
56	(38)	BITSTRING	2	SMF26IND	INDICATORS
56	(38)	BITSTRING	0	SMF26DES	"X'8000" DESCRIPTIVE SECTION PRESENT
56	(38)	BITSTRING	0	SMF26EVT	"X'4000" EVENT SECTION PRESENT
56	(38)	BITSTRING	0	SMF26ATU	"X'2000" ACTUALS SECTION PRESENT
56	(38)	BITSTRING	0	SMF26NTW	"X'1000" JES2 Network section present
56	(38)	BITSTRING	0	SMF26J2R	"X'0800" JES2 ROUTING SECTION PRESENT
56	(38)	BITSTRING	0	SMF26JXP	"X'0400" JES PRINTER SECTION PRESENT
56	(38)	BITSTRING	0	SMF26R02	"X'0200" Reserved
56	(38)	BITSTRING	0	SMF26ACP	"X'0100" Triplets Section present

Comment

BEGINNING OF DESCRIPTIVE SECTION

End of Comment

58	(3A)	BITSTRING	2	SMF26LN1	LGTH OF THIS SECTION INCLUDING SELF
60	(3C)	BITSTRING	2	SMF26RV1	RESV
62	(3E)	BITSTRING	1	SMF26IN2	ADDITIONAL JOB INFORMATION(JES2 ONLY)
		1... ....		SMF26BCH	"X'80" BIT 0 - BACKGROUND BATCH
		.1.. ....		SMF26FTS	"X'40" 1 - FOREGROUND TIME SHARING
		..1. ....		SMF26STK	"X'20" 2 - SYSTEM TASK
		...1 ....		SMF26NOJ	"X'10" 3 - NO JOURNAL OPTION
		.... 1..		SMF26NOU	"X'08" 4 - NO OUTPUT OPTION
		.... .1..		SMF26SCN	"X'04" 5 - TYPRUN=SCAN
		.... .1.		SMF26CPY	"X'02" 6 - TYPRUN=COPY

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
62	(3E)	.... ...1	1	SMF26JBF	"X'01" 7 - RESTART=Y
		1... ....		SMF26IN3	ADDITIONAL JOB INFORMATION(JES3 ONLY)
		.1.. ....		SMF26DJC	"X'80" BIT 0 - DEPENDENT JOB(/ NET JOB PROCESSED)
		..1. ....		SMF26DLJ	"X'40" 1 - JOB SPECIFIED DEADLINE SCHEDULING
		...1 ....		SMF26DLM	"X'20" 2 - DEADLINE JOB MET DEADLINE
		.... 1...		SMF26PRJ	"X'10" 3 - / PROCESS STMT PROCESSED
		.... .1..		SMF26NJX	"X'08" 4 - JOB LEFT SYSTEM VIA NJP(NETWORK JOB PROCESSING)
		.... ..1.		SMF26NJE	"X'04" 5 - JOB ENTERED SYSTEM VIA NJP
		.... ...1		SMF26DJO	"X'02" 6 - JOB LEFT SYSTEM VIA DJ(DUMP JOB)
		63		(3F)	.... ...1
1... ....	SMF26INF		JOB INFORMATION		
.... .1..	SMF26JCP		"X'80" 0 - JOB PRIORITY EXTERNALLY ASSIGNED (JES2-VIA PRIORITY STMT) (JES3-VIA PRTY PARM ON JOB STMT)		
.... ..1.	SMF26STU		"X'40" 1 - SETUP JOB (JES2- SETUP STMT PROCESSED) (JES3-PROCESSED BY PREEEXEC SETUP)		
.... ...1	SMF26TRH		"X'20" 2 - JOB HELD VIA TYPERUN=HOLD		
.... 1...	SMF26NLG		"X'10" 3 - JOB REQUESTED NO JES JOB LOG(JES2)		
.... .1..	SMF26XBC		"X'08" 4 - EXEC BATCHING JOB (JES2 ONLY)		
.... ..1.	SMF26EIR		"X'04" 5 - JOB ENTERED VIA INTERNAL RDR		
.... ...1	SMF26MRE		"X'02" 6 - JOB WAS RERUN BY JES		
64	(40)		.... ...1		4
		1... ....	SMF26JNM	JES ASSIGNED JOB #	
		.... ..1.	SMF26JID	8-character job identifier	
		.... ...1	SMF26NAM	PROGRAMMER'S NAME FROM JOB CARD	
		.... 1...	SMF26MSG	MESSAGE CLASS FROM JOB CARD	
		.... .1..	SMF26CLS	JOB CLASS FROM JOB CARD	
		.... ..1.	SMF26XPI	INITIAL JOB PRIORITY	
		.... ...1	SMF26XPS	SELECTION PRIORITY AT TIME JOB SELECTED	
		.... ..1.	SMF26IX2	Additional JOB information (JES2 ONLY)	
		68	(44)	1... ....	
.... ..1.	SMF26JOL			"X'40" Job purged as a result of spool offload	
.... ...1	SMF26LPN			"X'20" Job went thru unspun in its lifetime	
.... ..1.	SMF26XWR			"X'10" Job had at least one JOE purged due to PSO/SAPI	
.... ...1	SMF26OPS			Reserved	
.... ..1.	SMF26LOC			INPUT ROUTE CODE OR ZERO (JES2 ONLY)	
.... ...1	SMF26RV8			RESERVED(JES3)	
.... ..1.	SMF26DEV			LOG INPUT DEV NAME OF WHERE JOB READ USERID IF TSO SUBMIT SYSTEM NAME IF NJP	
.... ...1	SMF26ACT			PROGRAMMER'S ACCOUNTING # (JES2 ONLY)	
112	(70)			.... ..1.	4
		.... ...1	SMF26RVA	RESERVED(JES3)	
		.... ..1.	SMF26XTM	ESTIMATED EXECUTION TIME(SEC)	
		.... ...1	SMF26ELN	ESTIMATED OUTPUT LINES	
		.... ..1.	SMF26EPU	ESTIMATED OUTPUT PUNCHED CARDS	
		.... ...1	SMF26J2D	*** JES2 ONLY DESCRIPTIVE SECTION	
		.... ..1.	SMF26FRM	DEFAULT OUTPUT FORM #	
		.... ...1	SMF26CYP	PRINT COPY COUNT IF FOR ALL OF JOB	
		.... ..1.	SMF26LIN	LINES PER PAGE	
		136	(88)	.... ...1	
.... ..1.	SMF26PUR			DEFAULT PUNCH DESTINATION	
.... ...1	SMF26PDD			JES2 PROC DDNAME FOR JCL CONVERSION	
.... ..1.	SMF26J3D			*** JES3 ONLY DESCRIPTIVE SECTION	
.... ...1	SMF26DTY			DEADLINE SCHEDULE TYPE	
.... ..1.	SMF26RV6			RESERVED	
.... ...1	SMF26IGP			LOG INPUT DEV GROUP NAME(JOB SOURCE)	
.... ..1.	SMF26PD3			PROCEDURE DD NAME	
.... ...1	SMF26NJO			SYS NAME TO WHICH JOB SENT VIA NJP	
160	(A0)			.... ..1.	8
		.... ...1	SMF26NET	ID OF DEPENDENT JOB NET TO WHICH THIS JOB BELONGS(FROM / NET STMT)	
		.... ..1.	SMF26DTM	DEADLINE SCHEDULE TIME	
		.... ...1	SMF26DDT	DEADLINE SCHEDULE DATE	

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
184	(B8)	CHARACTER	8	SMF26CLN	JOB CLASS NAME

Comment

BEGINNING OF EVENT SECTION

End of Comment

58	(3A)	BITSTRING	2	SMF26LN2	LGTH OF THIS SECTION(INCLUDING SELF )
60	(3C)	BITSTRING	2	SMF26RV2	RESV
62	(3E)	BITSTRING	4	SMF26RPT	READER STOP TIME
66	(42)		4	SMF26RPD	READER STOP DATE
70	(46)	BITSTRING	4	SMF26CST	CONVERTER START TIME
74	(4A)		4	SMF26CSD	CONVERTER START DATE
78	(4E)	BITSTRING	4	SMF26CPT	CONVERTER STOP TIME
82	(52)		4	SMF26CPD	CONVERTER STOP DATE
86	(56)	BITSTRING	4	SMF26XST	EXECUTION START TIME
90	(5A)		4	SMF26XSD	EXECUTION START DATE
94	(5E)	BITSTRING	4	SMF26XPT	EXECUTION STOP TIME
98	(62)		4	SMF26XPD	EXECUTION STOP DATE
102	(66)	BITSTRING	4	SMF26OST	OUTPUT PROCESSOR START TIME
106	(6A)		4	SMF26OSD	OUTPUT PROCESSOR START DATE
110	(6E)	BITSTRING	4	SMF26OPT	OUTPUT PROCESSOR STOP TIME
114	(72)		4	SMF26OPD	OUTPUT PROCESSOR STOP DATE

Comment

BEGINNING OF ACTUALS SECTION

JES2 creates the Actuals section only up to (and including) SMF26OID. The fields from SMF26JAF to SMF26SRC are filled in by JES3. The block from NJEJMR to NJEJMREN contains some JES3 networking information but it is part of the Actuals section.

End of Comment

58	(3A)	BITSTRING	2	SMF26LN3	LGTH OF THIS SECTION(INCLUDING SELF )
60	(3C)	BITSTRING	2	SMF26RV4	RESV
62	(3E)	BITSTRING	4	SMF26ICD	# OF INPUT CARDS (JCL AND DATA)
66	(42)	BITSTRING	4	SMF26XLN	OUTPUT LINES GENERATED TO SPOOL
70	(46)	BITSTRING	4	SMF26XPU	OUTPUT PUNCH CARDS GENERATED TO SPOOL
74	(4A)	CHARACTER	4	SMF26RID	INPUT PROCESSOR SYSTEM ID
78	(4E)	CHARACTER	4	SMF26CID	JCL CONVERSION PROCESSOR SYSTEM ID
82	(52)	CHARACTER	4	SMF26XID	EXECUTION PROCESSOR SYSTEM ID
86	(56)	CHARACTER	4	SMF26OID	OUTPUT PROCESSOR SYSTEM ID
90	(5A)	CHARACTER	42	SMF26JAF	Job accounting fields - maximum length 42 - filled in by JES3 only
132	(84)	BITSTRING	4	NJEJMR (0)	
132	(84)	CHARACTER	8	NJEJMRID	
140	(8C)	ADDRESS	2	NJEJMRLN	- Length of NJEJMR

Comment

COMPATIBILITY CODE

NJEJOBNO is maintained for compatibility with pre-HJS7705 levels of JES3. Once HJS7703 and below are no longer supported, this field does not need to be maintained.

End of Comment

142	(8E)	ADDRESS	2	NJEJOBNO	- Origin node job number (compatible) - contains FFFF if NJEJOBNO > 65534
144	(90)	CHARACTER	8	NJEJOBNM	- JOB NAME
152	(98)	CHARACTER	8	NJEXEQN	- EXECUTION NODE
160	(A0)	CHARACTER	20	NJEPRGMR	- PROGRAMMER NAME
180	(B4)	CHARACTER	8	NJEUSRID	- TSO USER ID

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
188	(BC)	CHARACTER	8	NJEACCT	- NETWORK ACCT NUM
196	(C4)	CHARACTER	8	NJEDEPT	- PROGRAMMER DEPT NUM
204	(CC)	CHARACTER	8	NJEBLDG	- PROGRAMMER BLDG NUM
212	(D4)	CHARACTER	8	NJEROOM	- PROGRAMMER ROOM NUM
220	(DC)	CHARACTER	8	NJEXEQU	- EXECUTION USER ID

Comment

### COMPATIBILITY CODE

NJETRANS is maintained for compatibility with pre-HJS7705 levels of JES3. Once HJS7703 and below are no longer supported, this field does not need to be maintained.

End of Comment

228	(E4)	BITSTRING	4	NJETRANS (0)	Maintained for compile compatibility
228	(E4)	ADDRESS	4	NJEJOBX	Origin node job number, extended
232	(E8)	BITSTRING	4	NJEJMREN (0)	
232	(E8)	BITSTRING	0	NJEJMRSZ (0)	
232	(E8)	CHARACTER	4	SMF26SRC	NUMBER OF SPOOL RECORDS

Comment

Beginning of JES2 Network section

End of Comment

58	(3A)	BITSTRING	2	SMF26LN4	LENGTH OF THIS SECTION(INCLUDING SELF)
60	(3C)	BITSTRING	2	SMF26RV5	RESERVED
62	(3E)	CHARACTER	4	SMF26NID	JOB TRANSMITTER SYSTEM IDENTIFIER
66	(42)	BITSTRING	4	SMF26NST	JOB TRANSMITTER START TIME
70	(46)		4	SMF26NSD	JOB TRANSMITTER START DATE
74	(4A)	BITSTRING	4	SMF26NPT	JOB TRANSMITTER STOP TIME
78	(4E)		4	SMF26NPD	JOB TRANSMITTER STOP DATE
82	(52)	CHARACTER	8	SMF26NAC	NETWORK ACCOUNTING NUMBER
90	(5A)	CHARACTER	8	SMF26NJB	Original job identification
98	(62)	CHARACTER	8	SMF26NDV	JOB TRANSMITTER DEVICE NAME
106	(6A)	CHARACTER	8	SMF26NON	Original node name
114	(72)	CHARACTER	8	SMF26NXN	EXECUTION NODE NAME
122	(7A)	CHARACTER	8	SMF26NNM	NEXT NODE NAME
130	(82)	CHARACTER	8	SMF26NLN	LAST NODE NAME
138	(8A)	CHARACTER	8	SMF26SUI	Submitting Userid
146	(92)	CHARACTER	8	SMF26NN	JOB End Execution Notify Node
154	(9A)	CHARACTER	8	SMF26NU	Job End Execution Notify Userid
58	(3A)	BITSTRING	2	SMF26LN5	LGTH OF THIS SECTION
60	(3C)	CHARACTER	4	SMF26INR	INPUT ROUTING
64	(40)	CHARACTER	4	SMF26PRD	DEFAULT PRINT DESTINATION
68	(44)	CHARACTER	4	SMF26PUD	DEFAULT PUNCH DESTINATION
58	(3A)	BITSTRING	2	SMF26LN6	LGTH OF THIS SECTION
60	(3C)	CHARACTER	4	SMF26EBT	ESTIMATED BYTE COUNT
64	(40)	CHARACTER	4	SMF26XBT	ACTUAL BYTE COUNT
68	(44)	CHARACTER	4	SMF26EPG	ESTIMATED PAGE COUNT
72	(48)	CHARACTER	4	SMF26XPG	ACTUAL PAGE COUNT
76	(4C)	CHARACTER	8	SMF26EFM	EXPANDED FORMS ID

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

This is the header for all future extensions to the SMF 26 record. Sections beyond this point must be accessed by using the values stored in the triplets (below) that contain the offset, length, and number of sections of the type corresponding to the triplet. New sections will be appended to this header and their presence can be detected by an increase in the number of triplets and by a non-zero section offset, length and number of sections. Each offset to a section is added to the address of SMFRCD26 to obtain the start of the section that it locates.

Comment

End of Comment

58	(3A)	BITSTRING	2	SMF26LN7	Length of triplet section
60	(3C)	SIGNED	4	SMF26OAG	Offset of accounting section
64	(40)	BITSTRING	2	SMF26LAG	Length of accounting section
66	(42)	BITSTRING	2	SMF26NAG	Number of accounting sections
68	(44)	SIGNED	4	SMF26OWL	Offset of Work Load Manager section
72	(48)	BITSTRING	2	SMF26LWL	Length of Work Load Manager section
74	(4A)	BITSTRING	2	SMF26NWL	Number of Work Load Manager sections
76	(4C)	SIGNED	4	SMF26OJC	Offset of Job Correlator section
80	(50)	BITSTRING	2	SMF26LJC	Length of Job Correlator section
82	(52)	BITSTRING	2	SMF26NJC	Number of Job Correlator sections
84	(54)	CHARACTER	64	SMF26JCR	Job correlator
148	(94)	CHARACTER	8	SMF26WCL	Service class queue name
156	(9C)	CHARACTER	8	SMF26WOC	Original Service class
164	(A4)	BITSTRING	1	SMF26WIN	Indicators
		1... ....		SMF26WLM	"B'10000000" Job ran in MODE=WLM
		.1... ....		SMF26SJB	"B'01000000" Job ran because of the \$S J JES2 command or the *F,J=job,RUN JES3 command
165	(A5)	CHARACTER	8	SMF26WJC	Eight character job class
173	(AD)	CHARACTER	16	SMF26WSE	Scheduling environment (SCHENV)
189	(BD)	BITSTRING	2	SMF26LN8	Length of Accounting Section
191	(BF)	SIGNED	1	SMF26NRA	Number of accounting pairs that follow

Comment

Accounting pairs are of the form:  
AL1(length),C'string of length "length"  
A length of 0 indicates an omitted field

End of Comment

192	(C0)	SIGNED	1	SMF26AC1 (0)
-----	------	--------	---	--------------

Comment

LENGTH EQUATES

End of Comment

192	(C0)	X'5E'	0	SMF26L1	"SMF26PDD+L'SMF26PDD-SMF26LN1" DESCRIPTIVE SECT LEN
192	(C0)	X'3C'	0	SMF26L2	"SMF26OPD+L'SMF26OPD-SMF26LN2" EVENT SECTION LENGTH
192	(C0)	X'20'	0	SMF26L3	"SMF26OID+L'SMF26OID-SMF26LN3" ACTUALS SECTION LEN
192	(C0)	X'68'	0	SMF26L4	"SMF26NU+L'SMF26NU-SMF26LN4" NETWORK SECTION LEN
192	(C0)	X'E'	0	SMF26L5	"SMF26PUD+L'SMF26PUD-SMF26LN5" ROUTING SECTION LEN
192	(C0)	X'1A'	0	SMF26L6	"SMF26EFM+L'SMF26EFM-SMF26LN6" PRINT SECTION LEN

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
192	(C0)	X'1A'	0	SMF26L7	"SMF26NJC+L'SMF26NJC-SMF26LN7" Triplets sect len
192	(C0)	X'29'	0	SMF26L9	"SMF26WSE+L'SMF26WSE-SMF26WCL" WLM section len
192	(C0)	X'40'	0	SMF26LA	"L'SMF26JCR" Job corr sect len
Comment					
<p>SMF26L10 is used for the total length of the triplet section and any fixed length section that may follow. This equate is used to ensure compatability between a low level JES assembled with a higher level BCP maclib. For example, a HJE5520 JES2 assembled with a JBB6604 BCP will include the WLM triplet section as part of the SMF26L10 equate. If new triplet sections are added in the future, the SMF26L10 equate must be changed to add the length of the new section.</p>					
End of Comment					
192	(C0)	X'83'	0	SMF26L10	"SMF26L7+SMF26L9+SMF26LA" Triplet + WLM len + Job corr
192	(C0)	X'10'	0	SMF26SZ1	"L'SMF26JBN+L'SMF26RST+L'SMF26RSD" LENGTH OF JOB NAME, AND RDR START TIME AND DATE FOR MOVE
192	(C0)	X'4C'	0	SMF26SZ2	"SMF26NLN+L'SMF26NLN-SMF26NID" LEN OF NETWORK FIELDS
192	(C0)	X'8'	0	SMF26SZ3	"L'SMF26RPT+L'SMF26RPD" LEN OF RDR FIELDS FOR MOVE
192	(C0)	X'4'	0	SMF26SZ4	"L'SMF26PRR+L'SMF26PUR" LEN OF PRPU ROUTES FOR MOVE
192	(C0)	X'4F'	0	SMF26SZ5	"SMF26NU+L'SMF26NU-SMF26NAC-1" LEN OF NET FIELDS
192	(C0)	X'32'	0	SMF26LN	"SMF26IND+L'SMF26IND-SMF26LEN" LEN OF BASE + HEADER
192	(C0)	X'17C'	0	SMF26TLN	"SMF26LN+SMF26L1+SMF26L2+SMF26L3+SMF26L4+SMF26L5+SMF26L6" TOTAL LENGTH OF TYPE 26 RECORD
Comment					

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
```



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMFR6 55-69			
		IFASMFR9 80-84			
		IFASMFRA 85-103			
		IFASMFRB 104-113			
		IFASMFR6 114-123			
		IFASMFRD 124-127			
%GOTO IFABGN2;					
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS					
IFASMFR &RECTYPE					
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).					
%IAZPRO43; ;					
MODULE NAME = IAZSMF43					
DESCRIPTIVE NAME = JES SMF SUBSYSTEM START RECORD					
%GOTO IAZ43;					
SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC					
SUBSYSTEM START RECORD TYPE 43					

End of Comment					
8	(8)	SIGNED	4	(0)	ALIGNMENT TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD43	*** START OF RECORD
8	(8)	X'8'	0	SMF43PTR	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF43LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF43SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF43FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF43RTY	RECORD TYPE 43
13	(D)	X'2B'	0	SMFJ43	"43" START SUBSYSTEM RECORD TYPE
14	(E)	BITSTRING	4	SMF43TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF43DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF43SID	SYSTEM IDENTIFIER

Comment

SUBSYSTEM IDENTIFICATION SECTION

End of Comment					
26	(1A)	BITSTRING	2	SMF43SBS	SUBSYSTEM IDENTIFIER
		.... .1.		SMF43HSP	"X'0002" JES2 ID X'0005' JES3 X'0006' SS06
28	(1C)	BITSTRING	2	SMF43RSV	RESV
30	(1E)	BITSTRING	2	SMF43LRR	LGTH OF REMAINDER OF RECORD (NOT INCLUDING THIS FIELD)
30	(1E)	X'20'	0	SMF43SBP	*** SUBSYSTEM SECTION BEGINNING

Comment

JES2 AND JES3 COMMON SECTION

End of Comment					
32	(20)	BITSTRING	2	SMF43RV1	RESV
34	(22)	BITSTRING	1	SMF43RST	START RECORD FLAGS
		1... ....		SMF\$ESYS	"X'80" IF 1 THEN SMF43EID FIELD IS RESTART SYSTEM ID
		1... ....		SMF43CLD	"X'80" COLD START (JES3)
		.1.. ....		SMF43WRM	"X'40" WARM START
		.1. ....		SMF43HOT	"X'20" HOT START
		...1 ....		SMF43ANL	"X'10" QUEUE ANALYSIS REQUIRED
		.... 1...		SMF43GBL	"X'08" GLOBAL SYSTEM
		.... .1..		SMF43LCL	"X'04" LOCAL SYSTEM
		.... .1.		SMF43REF	"X'02" Configuration refresh was requested
		.... ...1		SMF43DSI	"X'01" DYNAMIC SYSTEM INTERCHANGE
34	(22)	X'23'	0	SMF43SBG	*** JES2,JES3 UNCOMMON SECTIONS

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
JES2 SECTION					
End of Comment					
35	(23)	BITSTRING	1	SMF43OPT	JES2 OPTIONS BIT MEANING WHEN SET 0 - FORMAT THE SPOOL 1 - COLD START 2 - REQUEST INIT AUTO 3 - LIST REPLACEMENTS 4-7 RESV
36	(24)	CHARACTER	4	SMF43EID	SYSTEM ID OF SYSTEM TO BE WARMSTARTED IF &ESYS OR 0 FOR START JES 2
Comment					
JES3 SECTION					
End of Comment					
35	(23)	BITSTRING	1	SMF43RV2	RESERVED FLAGS
36	(24)	BITSTRING	1	SMF43US1	USER FLAGS
37	(25)	CHARACTER	1	SMF43NMU	INITIALIZATION DECK ORIGIN TYPE
38	(26)	CHARACTER	8	SMF43ORG	INITIALIZATION DECK ORIGIN TYPE-ORIGIN CONTENTS N-MEMBER NAME(JCL DEFAULT) M-MEMBER NAME(OPER CHOICE) U-UNIT ADDRESS(OP CHOICE)
38	(26)	X'26'	0	SMF43UN4	"SMF43ORG,4" 4-Digit Device Number
38	(26)	X'26'	0	SMF43UNT	"SMF43ORG,3" 3-Digit Device Number
46	(2E)	CHARACTER	4	SMF43PJ3	JES3 PROCEDURE NAME
50	(32)	CHARACTER	8	SMF43RVJ	RESERVED FOR JES3
58	(3A)	CHARACTER	4	SMF43RVU	RESERVED FOR USER
62	(3E)	BITSTRING	1	SMF43END (0)	END OF JES3 RECORD
62	(3E)	BITSTRING	0	SMF43SIZ (0)	SIZE OF JES3 SMF43 RECORD
Comment					
SS06 SECTION					
End of Comment					
32	(20)	CHARACTER	8	SMF43PRC	SS06 START PROC NAME
40	(28)	CHARACTER	73	SMF43INT	INITIALIZATION
Comment					
LENGTH EQUATES					
End of Comment					
40	(28)	X'20'	0	SMF43L1	"SMF43EID+L'SMF43EID-SMF43LEN" LEN OF TYPE 43 RECORD
40	(28)	X'8'	0	SMF43L2	"SMF43EID+L'SMF43EID-SMF43RV1" LEN OF JES2 SECTION

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description

Comment

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
IFASMFR6 55-69
IFASMFR9 80-84
IFASMFR8 85-103
IFASMFRB 104-113
IFASMFRD 114-123
IFASMFRF 124-127
```

```
%GOTO IFABGN2;
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE
REQUIRED FORMAT IS
IFASMFR &RECTYPE
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).
```

```
%IAZPRO45 ;
MODULE NAME = IAZSMF45
DESCRIPTIVE NAME = JES SMF SUBSYSTEM STOP RECORD
%GOTO IAZ45;
SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC
SUBSYSTEM STOP RECORD TYPE 45
```

End of Comment					
8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD45	*** START OF RECORD
8	(8)	X'8'	0	SMF45PTR	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF45LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF45SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF45FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF45RTY	RECORD TYPE 45
13	(D)	X'2D'	0	SMFJ45	"45" STOP SUBSYSTEM RECORD TYPE
14	(E)	BITSTRING	4	SMF45TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF45DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF45SID	SYSTEM IDENTIFICATION

Comment					
---------	--	--	--	--	--

SUBSYSTEM IDENTIFICATION SECTION

End of Comment					
26	(1A)	BITSTRING	2	SMF45SBS	SUBSYSTEM IDENTIFIER
		.... ..1.		SMF45HSP	"X'0002" JES2 ID X'0005' JES3 X'0006' SS06
28	(1C)	BITSTRING	2	SMF45RSV	RESV
30	(1E)	BITSTRING	2	SMF45LRR	LGTH OF REMAINDER OF RECORD (NOT INCLUDING THIS FIELD)
30	(1E)	X'20'	0	SMF45SBG	*** SUBSYSTEM SECTION BEGINNING

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
JES2 SECTION					
End of Comment					
32	(20)	BITSTRING 1... ....	2	SMF45IND	INDICATORS
34	(22)	BITSTRING	2	SMF45ATM	"X'80" BIT 0 ABNORMAL TERMINATION 1-15 RESERVED
				SMF45JCC	JES2 COMPLETION CODE
Comment					
JES3 SECTION					
End of Comment					
32	(20)	BITSTRING 1... .... .1... ....	1	SMF45FG1	STOP RECORD FLAGS
				SMF45ABN	"X'80" SUBSYSTEM ENDED DUE TO ABEND
				SMF45DSI	"X'40" DSI HAS BEEN INVOKED
33	(21)	BITSTRING	3	SMF45J3C	COMPLETION CODE(SYS/USER)
36	(24)	BITSTRING	1	SMF45RV1	RESERVED
37	(25)	BITSTRING	1	SMF45US1	USER FLAG
38	(26)	CHARACTER	8	SMF45RVJ	RESERVED FOR JES3
46	(2E)	CHARACTER	4	SMF45RVU	RESERVED FOR USER
50	(32)	BITSTRING	1	SMF45END (0)	END OF JES3 RECORD
50	(32)	BITSTRING	0	SMF45SIZ (0)	SIZE OF JES3 45 RECORD
Comment					
SS06 SECTION					
End of Comment					
32	(20)	CHARACTER	8	SMF45PRC	SS06 PROCNAME
40	(28)	BITSTRING	1	SMF45STF	SYSTEM TERMINATION FLAGS
40	(28)	X'80'	0	SMF45HLT	"128" HALT ISSUED
40	(28)	X'40'	0	SMF45OPS	"64" OPERATOR STOP
40	(28)	X'20'	0	SMF45ABT	"32" ABNORMAL TERMINATION
40	(28)	X'10'	0	SMF45NOS	"16" NOSAVE SPECIFIED IN HALT
41	(29)	BITSTRING	3	SMF45UID	USER ID FOR HALT
44	(2C)	BITSTRING	2	SMF45NUL	NO. USERS LOGGED ON AT TERMINATION
Comment					
LENGTH EQUATES					
End of Comment					
44	(2C)	X'1C'	0	SMF45L1	"SMF45JCC+L'SMF45JCC-SMF45LEN" LEN OF TYPE 45 RECORD
44	(2C)	X'4'	0	SMF45L2	"SMF45JCC+L'SMF45JCC-SMF45IND" LEN OF JES2 SECTION

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description

Comment

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
IFASMFR6 55-69
IFASMFR9 80-84
IFASMFR8 85-103
IFASMFRB 104-113
IFASMFRD 114-123
IFASMFRF 124-127
```

```
%GOTO IFABGN2;
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE
REQUIRED FORMAT IS
IFASMFR &RECTYPE
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).
```

```
%IAZPRO47;;
MODULE NAME = IAZSMF47
DESCRIPTIVE NAME = JES SMF SUBSYSTEM EVENT START
%GOTO IAZ47;
SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC
SUBSYSTEM EVENT START RECORD TYPE 47
```

End of Comment					
8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD47	*** START OF RECORD
8	(8)	X'8'	0	SMF47PTR	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF47LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF47SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF47FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF47RTY	RECORD TYPE 47
13	(D)	X'2F'	0	SMFJ47	"47" START SUBSYSTEM EVENT ID
14	(E)	BITSTRING	4	SMF47TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF47DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF47SID	SYSTEM IDENTIFICATION

Comment					
---------	--	--	--	--	--

SUBSYSTEM IDENTIFICATION SECTION

End of Comment					
26	(1A)	BITSTRING	2	SMF47SBS	SUBSYSTEM IDENTIFIER
		.... ..1.		SMF47HSP	"X'0002" JES2 ID X'0005' JES3 X'0006' SS06
28	(1C)	BITSTRING	2	SMF47RSV	RESV
30	(1E)	BITSTRING	2	SMF47LRR	LGTH OF REMAINDER OF RECORD (NOT INCLUDING THIS FIELD)
30	(1E)	X'20'	0	SMF47SBG	*** SUBSYSTEM SECTION BEGINNING

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
JES2 AND JES3 SECTION					
End of Comment					
32	(20)	BITSTRING	2	SMF47EVT	EVENT STARTING
32	(20)	X'1'	0	SMF47SON	"1" TERMINAL SIGNED ON
32	(20)	X'2'	0	SMF47STL	"2" LINE STARTED
32	(20)	X'4'	0	SMF47LON	"4" TERMINAL LOGGED ON
Comment					
BEGINNING OF GENERAL SECTION					
End of Comment					
34	(22)	BITSTRING	2	SMF47LN1	LGTH OF THIS SECTION(INCLUDING SELF-26)
36	(24)	CHARACTER	8	SMF47RMT	REMOTE NAME
44	(2C)	CHARACTER	8	SMF47LIN	LINE NAME
52	(34)	CHARACTER	8	SMF47PSW	PASSWORD
Comment					
BEGINNING OF SIGNON MESSAGE SECTION					
THIS SECTION EXISTS ONLY FOR SIGNON EVENT STARTS					
End of Comment					
60	(3C)	BITSTRING	2	SMF47LN2	LGTH OF THIS SECTION(INCLUDING SELF-38)
62	(3E)	CHARACTER	36	SMF47MSG	MESSAGE FOR SIGNON, COLUMNS 35-70 OF SIGNON CARD.
98	(62)	CHARACTER	8	SMF47RVJ	RESERVED FOR JES3
106	(6A)	CHARACTER	4	SMF47RVU	RESERVED FOR USER
110	(6E)	BITSTRING	1	SMF47END (0)	END OF JES3 RECORD
110	(6E)	BITSTRING	0	SMF47SIZ (0)	SIZE OF JES3 47 RECORD
Comment					
SS06 SECTION					
End of Comment					
32	(20)	BITSTRING	1	SMF47LCF	LOGON CONDITION FLAG
32	(20)	X'80'	0	SMF47UNL	"128" USER NOT LOGGED OFF
32	(20)	X'20'	0	SMF47CWK	"32" CONTINUE WORKSPACE EXISTED AT LOGON
32	(20)	X'4'	0	SMF47SPA	"4" SERVICE PROGRAM ACCOUNT RECORD
32	(20)	X'1'	0	SMF47ULK	"1" USER LOCKED
33	(21)	BITSTRING	3	SMF47UID	USER ID
36	(24)	BITSTRING	1	SMF47LTC	LIBRARY TYPE CODE
37	(25)	BITSTRING	3	SMF47PLI	PROJECT LIB ID
40	(28)	CHARACTER	6	SMF47JID	JOB ENTRY ID CODE
46	(2E)	BITSTRING	1	SMF47LAA	LANGUAGE ATTRIBUTE ASSIGNED
47	(2F)	BITSTRING	1	SMF47PCI	PRIVILEGED CLASS INDICATORS
48	(30)	BITSTRING	4	SMF47DSL	DASD SPACE IN LIBRARY (1K UNITS)
52	(34)	BITSTRING	4	SMF47DPL	DASD SPACE PROJECT/PUBLIC LIBRARIES(1K UNITS)
Comment					
LENGTH EQUATES					
End of Comment					
32	(20)	X'1A'	0	SMF47L1	"SMF47LN2-SMF47LN1" LEN OF GENERAL SECTION
32	(20)	X'26'	0	SMF47L2	"SMF47MSG+L'SMF47MSG-SMF47LN2" LEN OF SIGNON MSG SEC
32	(20)	X'5A'	0	SMF47L3	"SMF47MSG+L'SMF47MSG-SMF47LEN" LEN OF TYPE 47 RECORD

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
32	(20)	X'34'	0	SMF47L4	"SMF47LN2-SMF47LEN" LEN OF RECORD - MESSAGE SECTION

Comment

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
IFASMFR6 55-69
IFASMFR9 80-84
IFASMFR8 85-103
IFASMFRB 104-113
IFASMFRD 114-123
IFASMFRF 124-127

%GOTO IFABGN2;
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE
REQUIRED FORMAT IS
IFASMFR &RECTYPE
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

%IAZPRO48 ;
MODULE NAME = IAZSMF48
DESCRIPTIVE NAME = JES SMF SUBSYSTEM EVENT STOP RECORD
%GOTO IAZ48;
SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC
SUBSYSTEM EVENT STOP RECORD TYPE 48
```

End of Comment

8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD48	*** START OF RECORD
8	(8)	X'8'	0	SMF48PTR	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF48LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF48SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF48FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF48RTY	RECORD TYPE 48
13	(D)	X'30'	0	SMFJ48	"48" STOP SUBSYSTEM EVENT ID
14	(E)	BITSTRING	4	SMF48TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF48DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF48SID	SYSTEM IDENTIFICATION

Comment

SUBSYSTEM IDENTIFICATION SECTION

End of Comment

26	(1A)	BITSTRING	2	SMF48SBS	SUBSYSTEM IDENTIFIER
		.... ..1.		SMF48HSP	"X'0002" JES2 ID X'0005' JES3 X'0006' SS06
28	(1C)	BITSTRING	2	SMF48RSV	RESV
30	(1E)	BITSTRING	2	SMF48LRR	LGTH OF REMAINDER OF RECORD (NOT INCLUDING THIS FIELD)

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
30	(1E)	X'20'	0	SMF48SBP	*** SUBSYSTEM SECTION BEGINNING
Comment					
JES2 AND JES3 COMMON SECTION					
End of Comment					
32	(20)	BITSTRING	2	SMF48EVT	TYPE OF EVENT STOPPED
32	(20)	X'1'	0	SMF48SOF	"1" LINE HAS SIGNED OFF
32	(20)	X'2'	0	SMF48CAN	"2" LINE CANCELLED BY OPERATOR
32	(20)	X'4'	0	SMF48LOF	"4" TERMINAL LOGGED OFF
34	(22)	BITSTRING	2	SMF48RV1	RESV
36	(24)	CHARACTER	8	SMF48RMT	REMOTE NAME
44	(2C)	CHARACTER	8	SMF48LIN	LINE NAME
52	(34)	CHARACTER	8	SMF48PSW	PASSWORD
52	(34)	X'3C'	0	SMF48SBG	*** JES2 AND JES3 UNCOMMON SECTIONS
Comment					
JES2 SECTION					
End of Comment					
60	(3C)	BITSTRING	4	SMF48IO	# EXCPS(NOT INCLUDING LINE REPEATS)
64	(40)	BITSTRING	4	SMF48NAK	# NAKS TO WRITE TEXT-NEG ACKNOWLEDGEMTS
68	(44)	BITSTRING	4	SMF48DCK	# DATA CHECKS TO READ TEXT
72	(48)	BITSTRING	4	SMF48OUT	# TIME OUTS TO READ TEXT
76	(4C)	BITSTRING	4	SMF48ERR	SUM OF ALL OTHER LINE ERRORS
80	(50)	CHARACTER	3	SMF48LAA	LINE ADAPTER ADDRESS FROM UCB
83	(53)	CHARACTER	4	SMF48LA4	4-Digit Line Adapter Address
Comment					
JES3 SECTION					
End of Comment					
60	(3C)	BITSTRING	28	SMF48XCP	EXCP COUNTS AND ERROR STATISTICS
60	(3C)	BITSTRING	4	SMF48TRN	NUMBER OF TRANSMISSIONS
64	(40)	BITSTRING	4	SMF48ERS	NUMBER OF LINE ERRORS
68	(44)	BITSTRING	2	SMF48TOT	NUMBER OF TIME-OUTS
70	(46)	BITSTRING	2	SMF48NKS	NUMBER OF NAK RESPONSES TO WRITE
72	(48)	BITSTRING	1	SMF48S0	NUMBER OF COMMAND REJECTS
73	(49)	BITSTRING	1	SMF48S1	NUMBER OF INTERVENTIONS REQUIRED
74	(4A)	BITSTRING	1	SMF48S2	NUMBER OF BUS-OUT CHECKS
75	(4B)	BITSTRING	1	SMF48S3	NUMBER OF EQUIPMENT CHECKS
76	(4C)	BITSTRING	1	SMF48S4	NUMBER OF DATA CHECKS
77	(4D)	BITSTRING	1	SMF48S5	NUMBER OF DATA OVERRUNS
78	(4E)	BITSTRING	1	SMF48S6	NUMBER OF LOST DATAS
79	(4F)	BITSTRING	9	SMF48USR	RESERVED FOR USER
88	(58)	CHARACTER	3	SMF48ADP	LINE ADAPTER ADDRESS
91	(5B)	CHARACTER	4	SMF48AD4	4-Digit Line Adapter Address
95	(5F)	CHARACTER	4	SMF48RVJ	Reserved for JES3
99	(63)	CHARACTER	4	SMF48RVU	RESERVED FOR USER
103	(67)	BITSTRING	1	SMF48END (0)	END OF JES3 RECORD
103	(67)	BITSTRING	0	SMF48SIZ (0)	SIZE OF JES3 48 RECORD
Comment					
SS06 SECTION					
End of Comment					
32	(20)	BITSTRING	1	SMF48FLS	LOGOFF FLAGS
32	(20)	X'80'	0	SMF48RPI	"128" RECORD FOR PREVIOUS INCOMPLETE SESSION
32	(20)	X'40'	0	SMF48CNI	"64" CANCEL ISSUED
32	(20)	X'20'	0	SMF48CWK	"32" CONTINUE WORKSPACE SAVED



Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
32	(20)	X'10'	0	SMF48CPG	"16" CONTINUE PURGED
32	(20)	X'8'	0	SMF48HSI	"8" HALT OR STOP ISSUED
32	(20)	X'4'	0	SMF48SPA	"4" SERVICE PROGRAM ACCOUNT RECORD
32	(20)	X'1'	0	SMF48ULK	"1" USER LOCKED
33	(21)	BITSTRING	3	SMF48UID	USER ID
36	(24)	BITSTRING	4	SMF48CPU	CPU TIME
40	(28)	BITSTRING	4	SMF48CNT	CONNECT TIME (SECONDS FOR THIS TERMINAL SESSION
44	(2C)	BITSTRING	4	SMF48CTH	CONNECT TIME (SECONDS) FOR THIS TERMINAL SESSION FOR ATTACHED HARDCOPY DEVICE
48	(30)	BITSTRING	4	SMF48VIR	VIRTUAL STORAGE USED (THOUSANDS OF BYTE-SECONDS) DURING TERMINAL SESSION
52	(34)	BITSTRING	4	SMF48DIO	DASD I/O COUNT FOR THIS TERMINAL SESSION
56	(38)	BITSTRING	4	SMF48TIO	TELEPROCESSING I/O COUNTS DURING TERMINAL SESSION
60	(3C)	BITSTRING	4	SMF48DSL	DASD SPACE IN THIS LIBRARY (IN 1K UNITS)
64	(40)	BITSTRING	4	SMF48DSP	DASD SPACE IN PROJECT/PUB LIBRARIES (1K UNITS)
68	(44)	BITSTRING	4	SMF48CPD	CPU TIME TO DATE (HUNDREDTHS OF SECONDS)
72	(48)	BITSTRING	4	SMF48CTD	CONNECT TIME TO DATE (SECS)
76	(4C)	BITSTRING	4	SMF48CDH	CONNECT TIME FOR HARDCOPY DEVICE TO DATE (SECONDS)
80	(50)	BITSTRING	4	SMF48VSD	VIRT STORAGE USED TO DATE (THOUSANDS OF BYTE-SECONDS)
84	(54)	BITSTRING	4	SMF48DID	DASD I/O COUNTS TO DATE
88	(58)	BITSTRING	4	SMF48TID	TP I/O COUNTS TO DATE

Comment

LENGTH EQUATES

End of Comment

88	(58)	X'4F'	0	SMF48L1	"SMF48LA4+L'SMF48LA4-SMF48LEN" Type 48 Record Len
88	(58)	X'14'	0	SMF48CT	"SMF48ERR+L'SMF48ERR-SMF48IO" LENGTH OF LINE EVENT COUNT SECTION

Comment

%IFABGN1 ;

METHOD OF ACCESS

PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM

DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE

INCLUDE MACRO FROM LIBRARY

EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-

DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON

%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD

A DIAGNOSTIC.

THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT

ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER

RECORDS AS FOLLOWS:

MACRO RECORDS

IFASMFR1 07-19

IFASMFR2 20-27

IFASMFR3 28-36

IFASMFR4 37-46

IFASMFR5 47-54

# \$SMF Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMFR6	55-69		
		IFASMFR9	80-84		
		IFASMFRA	85-103		
		IFASMFRB	104-113		
		IFASMFRC	114-123		
		IFASMFRD	124-127		

%GOTO IFABGN2;

THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS

IFASMFR &RECTYPE

NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

%IAZPRO49; ;

MODULE NAME = IAZSMF49

DESCRIPTIVE NAME = JES SMF SUBSYSTEM INTEGRITY RECORD

%GOTO IAZ49;

SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC

SUBSYSTEM INTEGRITY RECORD TYPE 49

End of Comment

8	(8)	SIGNED	4	(0)	ALIGN TO FULL WORD BOUNDRY
8	(8)	X'8'	0	SMFRCD49	*** START OF RECORD
8	(8)	X'8'	0	SMF49PTR	*** HEADER SEGMENT
8	(8)	BITSTRING	2	SMF49LEN	RECORD LENGTH
10	(A)	BITSTRING	2	SMF49SEG	SEGMENT DESCRIPTOR
12	(C)	BITSTRING	1	SMF49FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF49RTY	RECORD TYPE 49
13	(D)	X'31'	0	SMFJ49	"49" INTEGRITY EVENT RECORD TYPE
14	(E)	BITSTRING	4	SMF49TME	TOD FROM TIME MACRO BINARY
18	(12)		4	SMF49DTE	DATE FROM TIME MACRO
22	(16)	CHARACTER	4	SMF49SID	SYSTEM IDENTIFICATION

Comment

SUBSYSTEM IDENTIFICATION SECTION

End of Comment

26	(1A)	BITSTRING	2	SMF49SBS	SUBSYSTEM IDENTIFIER X'0002' JES2 X'0005' JES3 X'0006' SS06
28	(1C)	BITSTRING	2	SMF49RSV	RESV
30	(1E)	BITSTRING	2	SMF49LRR	LGTH OF REMAINED OF RECORD (NOT INCLUDING THIS FIELD)
30	(1E)	X'20'	0	SMF49SBG	*** SUBSYSTEM SECTION BEGINNING

Comment

JES2 AND JES3 SECTION

End of Comment

32	(20)	BITSTRING	2	SMF49EVT	EVENT STARTING
----	------	-----------	---	----------	----------------

Comment

FOLLOWING BIT DEFINITIONS APPLY TO JES2

End of Comment

32	(20)	X'1'	0	SMF49SON	"1" SIGNON
32	(20)	X'2'	0	SMF49STL	"2" START LINE

Comment

FOLLOWING BIT DEFINITIONS APPLY TO JES3

End of Comment

32	(20)	X'1'	0	SMF49NER	"1" TERMINAL NOT DEFINED (BSC)
----	------	------	---	----------	--------------------------------

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
32	(20)	X'2'	0	SMF49PER	"2" SECURITY FAILURE (BSC)
32	(20)	X'4'	0	SMF49LER	"4" LINE ALREADY SIGNED ON (BSC)
32	(20)	X'8'	0	SMF49TER	"8" TERMINAL ALREADY SIGNED ON (BSC)
32	(20)	X'5'	0	SMF49LIM	"5" SESSION LIMIT EXCEEDED (SNA)
32	(20)	X'6'	0	SMF49DEF	"6" WORK STATION UNDEFINED (SNA)
32	(20)	X'7'	0	SMF49SPW	"7" SECURITY FAILURE (SNA)
32	(20)	X'8'	0	SMF49BND	"8" BIND FAILURE (SNA)

Comment

BEGINNING OF GENERAL SECTION

End of Comment

34	(22)	BITSTRING	2	SMF49LN1	LGTH OF THIS SECTION(INCLUDING SELF-26)
36	(24)	CHARACTER	8	SMF49RMT	REMOTE NAME
44	(2C)	CHARACTER	8	SMF49LIN	LINE NAME
52	(34)	CHARACTER	8	SMF49PSW	PASSWORD USED(INVALID)

Comment

BEGINNING OF SIGNON MESSAGE SECTION  
THIS SECTION EXISTS ONLY FOR SIGNON EVENT STARTS

End of Comment

60	(3C)	BITSTRING	2	SMF49LN2	LGTH OF THIS SECTION(INCLUDING SELF-38)
62	(3E)	CHARACTER	36	SMF49MSG	MESSAGE FOR SIGNON, COLUMNS 35-70 OF SIGNON CARD.
98	(62)	BITSTRING	1	SMF49END (0)	END OF JES3 RECORD
98	(62)	BITSTRING	0	SMF49SIZ (0)	SIZE OF JES3 49 RECORD

Comment

SS06 SECTION

End of Comment

32	(20)	BITSTRING	3	SMF49VID	VIOLATOR IDENTIFICATION
35	(23)	BITSTRING	3	SMF49LNA	LIBRARY NUMBER OR ACCESSED LIBRARY
38	(26)	BITSTRING	2	SMF49RV1	RESERVED
40	(28)	CHARACTER	12	SMF49FLN	FILENAME OF FILE ATTEMPTED
52	(34)	BITSTRING	3	SMF49UFO	USERNUMBER OF FILE OWNER

Comment

LENGTH EQUATES

End of Comment

32	(20)	X'1A'	0	SMF49L1	"SMF49LN2-SMF49LN1" LEN OF GENERAL SECTION
32	(20)	X'26'	0	SMF49L2	"SMF49MSG+L'SMF49MSG-SMF49LN2" LEN OF SIGNON MSG SEC
32	(20)	X'5A'	0	SMF49L3	"SMF49MSG+L'SMF49MSG-SMF49LEN" LEN OF TYPE 49 RECORD

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>%IFABGN1: ;            METHOD OF ACCESS            PLS - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM            DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE            INCLUDE MACRO FROM LIBRARY            EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-            DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON            %INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD            A DIAGNOSTIC.            THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT            ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER            RECORDS AS FOLLOWS:            MACRO RECORDS            IFASMFR1 07-19            IFASMFR2 20-27            IFASMFR3 28-36            IFASMFR4 37-46            IFASMFR5 47-54            IFASMFR6 55-69            IFASMFR9 80-84            IFASMFR8 85-103            IFASMFRB 104-113            IFASMFRC 114-123            IFASMFRD 124-127</p>					
<p>%GOTO IFABGN2;            THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE            REQUIRED FORMAT IS            IFASMFR &amp;RECTYPE            NOTE: VALUES FOR &amp;RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).            MODULE NAME = IAZSMF52            DESCRIPTIVE NAME = JES SMF SNA START EVENT RECORD            SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC</p>					
End of Comment					
8	(8)	X'8'	0	SMFRCD52	*** START OF RECORD
8	(8)	X'8'	0	SMF52PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF52LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF52SEG	SEGMENT DESCRIPTOR
Comment					
BEGINNING OF JMR OR HASP SMF RECORD					
End of Comment					
12	(C)	BITSTRING	1	SMF52FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF52RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF52TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF52DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF52SID	SYSTEM IDENTIFICATION
Comment					
HEADER FOR HASP SUBSYSTEM RECORD TYPE 52					
End of Comment					
22	(16)	X'34'	0	SMFJ52	"52" RECORD TYPE SNA START EVENT
26	(1A)	BITSTRING	2	SMF52POF	OFFSET TO PRODUCT SECTION
28	(1C)	BITSTRING	2	SMF52PRL	LENGTH OF PRODUCT SECTION
30	(1E)	BITSTRING	2	SMF52PRN	NUMBER OF PRODUCT SECTION
32	(20)	BITSTRING	2	SMF52IDO	OFFSET TO ID SECTION
34	(22)	BITSTRING	2	SMF52IDL	LENGTH OF ID SECTION

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
36	(24)	BITSTRING	2	SMF52IDN	NUMBER OF ID SECTION

Comment

PRODUCT SECTION HASP SUBSYSTEM RECORD TYPES 52

End of Comment

38	(26)	BITSTRING	2	SMF52SUB	SUBTYPE ID NUMBER
38	(26)	X'1'	0	SMF52LON	"1" LOGON EVENT
38	(26)	X'2'	0	SMF52SLN	"2" START LINE EVENT
40	(28)	CHARACTER	2	SMF52VER	RECORD VERSION NUMBER
42	(2A)	CHARACTER	4	SMF52SYS	SUBSYSTEM NAME
46	(2E)	CHARACTER	1	SMF52IDS (0)	DEFINE START OF ID SECTION

Comment

ID SECTION OF HASP TYPE 52 (SNA) START EVENT AFTER TWO HDRS

End of Comment

46	(2E)	CHARACTER	8	SMF52RMT	REMOTE NAME
54	(36)	CHARACTER	8	SMF52LIN	LINE NAME
62	(3E)	CHARACTER	8	SMF52PSW	LINE PASSWORD
70	(46)	CHARACTER	1	SMF52END (0)	END OF TYPE 52 RECORD

Comment

LENGTH EQUATES

End of Comment

70	(46)	X'1E'	0	SMF52OFP	"SMF52IDN+L'SMF52IDN-SMFRCD52" OFFSET TO PROD SECT
70	(46)	X'8'	0	SMF52LPR	"SMF52SYS+L'SMF52SYS-SMF52SUB" LENGTH OF PROD SECT
70	(46)	X'26'	0	SMF52OFI	"SMF52SYS+L'SMF52SYS-SMFRCD52" OFFSET TO ID SECT
70	(46)	X'18'	0	SMF52LID	"SMF52PSW+L'SMF52PSW-SMF52IDS" LENGTH OF ID SECT

Comment

%IFABGN1 ;

METHOD OF ACCESS

PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM

DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE

INCLUDE MACRO FROM LIBRARY

EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-

DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON

%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD

A DIAGNOSTIC.

THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT

ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER

RECORDS AS FOLLOWS:

MACRO RECORDS

IFASMFR1 07-19

IFASMFR2 20-27

IFASMFR3 28-36

IFASMFR4 37-46

IFASMFR5 47-54

# \$SMF Map

## Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMFR6	55-69		
		IFASMFR9	80-84		
		IFASMFR6	85-103		
		IFASMFRB	104-113		
		IFASMFR6	114-123		
		IFASMFRD	124-127		

%GOTO IFABGN2;

THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS

IFASMFR &RECTYPE

NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

MODULE NAME = IAZSMF53

DESCRIPTIVE NAME = JES SMF SNA STOP EVENT RECORD

SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC

End of Comment

8	(8)	X'8'	0	SMFRCD53	*** START OF RECORD
8	(8)	X'8'	0	SMF53PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF53LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF53SEG	SEGMENT DESCRIPTOR

Comment

BEGINNING OF JMR OR HASP SMF RECORD

End of Comment

12	(C)	BITSTRING	1	SMF53FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF53RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF53TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF53DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF53SID	SYSTEM IDENTIFICATION

Comment

HEADER FOR HASP SUBSYSTEM RECORD TYPE 53

End of Comment

22	(16)	X'35'	0	SMFJ53	"53" RECORD TYPE SNA STOP EVENT
26	(1A)	BITSTRING	2	SMF53PRO	OFFSET TO PRODUCT SECTION
28	(1C)	BITSTRING	2	SMF53PRL	LENGTH OF PRODUCT SECTION
30	(1E)	BITSTRING	2	SMF53PRN	NUMBER OF PRODUCT SECTION
32	(20)	BITSTRING	2	SMF53IDO	OFFSET TO ID SECTION
34	(22)	BITSTRING	2	SMF53IDL	LENGTH OF ID SECTION
36	(24)	BITSTRING	2	SMF53IDN	NUMBER OF ID SECTION

Comment

PRODUCT SECTION HASP SUBSYSTEM RECORD TYPE 53

End of Comment

38	(26)	BITSTRING	2	SMF53SUB	SUBTYPE ID NUMBER
----	------	-----------	---	----------	-------------------

Comment

THE FOLLOWING EQUATES APPLY TO RECORD TYPE 53

End of Comment

38	(26)	X'1'	0	SMF53LOF	"1" LOGOFF EVENT
38	(26)	X'2'	0	SMF53PLN	"2" STOP LINE EVENT
40	(28)	CHARACTER	2	SMF53VER	RECORD VERSION NUMBER
42	(2A)	CHARACTER	4	SMF53SYS	SUBSYSTEM NAME
46	(2E)	CHARACTER	1	SMF53IDS (0)	DEFINE START OF ID SECTION

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
ID SECTION OF HASP TYPE 53 (SNA) STOP EVENT AFTER TWO HDRS					
End of Comment					
46	(2E)	CHARACTER	8	SMF53RMT	REMOTE NAME
54	(36)	CHARACTER	8	SMF53LIN	LINE NAME
62	(3E)	CHARACTER	8	SMF53PSW	LINE PASSWORD
70	(46)	BITSTRING	20	SMF53CTR	LINE EVENT COUNTERS
90	(5A)	CHARACTER	3	SMF53ADP	LINE IDENTIFIER
93	(5D)	CHARACTER	1	SMF53END (0)	END OF TYPE 53 RECORD
93	(5D)	X'1E'	0	SMF53OFP	"SMF53IDN+L'SMF53IDN-SMFRCD53" OFFSET TO PROD SECT
93	(5D)	X'8'	0	SMF53LPR	"SMF53SYS+L'SMF53SYS-SMF53SUB" LENGTH OF PROD SECT
93	(5D)	X'26'	0	SMF53OFI	"SMF53SYS+L'SMF53SYS-SMFRCD53" OFFSET TO ID SECT
93	(5D)	X'2F'	0	SMF53LID	"SMF53END-SMF53IDS" LENGTH OF ID SECT

Comment

```
%IFABGN1 ;
METHOD OF ACCESS
PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM
DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE
INCLUDE MACRO FROM LIBRARY
EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-
DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON
%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD
A DIAGNOSTIC.
THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT
ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER
RECORDS AS FOLLOWS:
MACRO RECORDS
IFASMFR1 07-19
IFASMFR2 20-27
IFASMFR3 28-36
IFASMFR4 37-46
IFASMFR5 47-54
IFASMFR6 55-69
IFASMFR9 80-84
IFASMFR8 85-103
IFASMFRB 104-113
IFASMFRD 114-123
IFASMFRD 124-127
%GOTO IFABGN2;
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE
REQUIRED FORMAT IS
IFASMFR &RECTYPE
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).
MODULE NAME = IAZSMF54
DESCRIPTIVE NAME = JES SMF SPOOL INTEGRITY EVENT RECORD
```

End of Comment					
8	(8)	X'8'	0	SMFRCD54	*** START OF RECORD
8	(8)	X'8'	0	SMF54PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF54LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF54SEG	SEGMENT DESCRIPTOR

Comment

BEGINNING OF JMR OR HASP SMF RECORD

End of Comment

## \$SMF Map

### Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
12	(C)	BITSTRING	1	SMF54FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF54RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF54TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)	BITSTRING	4	SMF54DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF54SID	SYSTEM IDENTIFICATION

Comment

HEADER FOR HASP SUBSYSTEM RECORD TYPES 54

End of Comment

22	(16)	X'36'	0	SMFJ54	"54" SPOOL INTEGRITY EVENT REC
26	(1A)	BITSTRING	2	SMF54POF	OFFSET TO PRODUCT SECTION
28	(1C)	BITSTRING	2	SMF54PRL	LENGTH OF PRODUCT SECTION
30	(1E)	BITSTRING	2	SMF54PRN	NUMBER OF PRODUCT SECTION
32	(20)	BITSTRING	2	SMF54IDO	OFFSET TO ID SECTION
34	(22)	BITSTRING	2	SMF54IDL	LENGTH OF ID SECTION
36	(24)	BITSTRING	2	SMF54IDN	NUMBER OF ID SECTION

Comment

PRODUCT SECTION HASP SUBSYSTEM RECORD TYPE 54

End of Comment

38	(26)	BITSTRING	2	SMF54SUB	SUBTYPE ID NUMBER
38	(26)	X'1'	0	SMF54LON	"1" LOGON EVENT
40	(28)	CHARACTER	2	SMF54VER	RECORD VERSION NUMBER
42	(2A)	CHARACTER	4	SMF54SYS	SUBSYSTEM NAME
46	(2E)	CHARACTER	1	SMF54IDS (0)	DEFINE START OF ID SECTION

Comment

ID SECTION OF HASP 54 (SNA) INTEGRITY RECORD AFTER TWO HDRS

End of Comment

46	(2E)	CHARACTER	8	SMF54RMT	REMOTE NAME
54	(36)	CHARACTER	8	SMF54RPW	REMOTE PASSWORD
62	(3E)	CHARACTER	8	SMF54PSW	LINE PASSWORD
70	(46)	CHARACTER	1	SMF54END (0)	END OF TYPE 54 RECORD

Comment

%IFABGN1: ;

METHOD OF ACCESS

PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM

DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE

INCLUDE MACRO FROM LIBRARY

EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-

DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON

%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD

A DIAGNOSTIC.

THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT

ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER

RECORDS AS FOLLOWS:

MACRO RECORDS

IFASMFR1 07-19

IFASMFR2 20-27

IFASMFR3 28-36

IFASMFR4 37-46

IFASMFR5 47-54



Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMFR6	55-69		
		IFASMFR9	80-84		
		IFASMFR6	85-103		
		IFASMFRB	104-113		
		IFASMFR6	114-123		
		IFASMFRD	124-127		

%GOTO IFABGN2;

THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS

IFASMFR &RECTYPE

NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

MODULE NAME = IAZSMF55

DESCRIPTIVE NAME = JES SMF NETWORK SIGNON RECORD

SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC

End of Comment

8	(8)	X'8'	0	SMFRCD55	*** START OF RECORD
8	(8)	X'8'	0	SMF55PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF55LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF55SEG	SEGMENT DESCRIPTOR

Comment

BEGINNING OF JMR OR HASP SMF RECORD

End of Comment

12	(C)	BITSTRING	1	SMF55FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF55RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF55TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF55DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF55SID	SYSTEM IDENTIFICATION

Comment

HEADER FOR HASP SUBSYS RECORD TYPE 55

End of Comment

22	(16)	X'37'	0	SMFJ55	"55" NETWORK SIGNON RECORD TYPE
26	(1A)	BITSTRING	2	SMF55SBS	HASP SUBSYSTEM ID
		.... ..1.		SMF55HSP	"X'0002" JES2 ID
28	(1C)	BITSTRING	2	SMF55SUB	RECORD SUBTYPE
30	(1E)	BITSTRING	2	SMF55LRR	LENGTH OF REST OF RECORD
30	(1E)	X'18'	0	SMF55STR	** -SMF55LEN" LENGTH OF HEADING SECTIONS

Comment

BEGINNING OF HASP TYPE 55 NETWORKING SIGNON RECORD AFTER 2 HDRS

End of Comment

32	(20)	CHARACTER	8	SMF55NNM	NODE NAME
40	(28)	BITSTRING	1	SMF55MEM	MEMBER NUMBER
41	(29)	BITSTRING	1	SMF55FG1	SIGNON STATUS FLAGS
		1... ....		SMF55RSO	"B'10000000" RESPONSE SIGNON
		.1.. ....		SMF55CON	"B'01000000" Reset/concur signon
		..1. ....		SMF55SEC	"B'00100000" Secure signon protocol (SMF55LPW and SMF55NPW are not set with this protocol)
42	(2A)	CHARACTER	8	SMF55LPW	LINE PASSWORD
50	(32)	CHARACTER	8	SMF55NPW	NODE PASSWORD
58	(3A)	CHARACTER	8	SMF55LNM	LINE NAME
66	(42)	BITSTRING	4	SMF55BSZ	Negotiated buffer size
70	(46)	CHARACTER	1	SMF55END (0)	END OF TYPE 55 RECORD (SHORT)

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
OPTIONAL INFORMATION ABOUT ADJACENT NODE					
End of Comment					
70	(46)	BITSTRING	16	SMF55IPA	BINARY IP ADDRESS, IPv6 FORMAT
86	(56)	BITSTRING	2	SMF55PRT	PORT NUMBER
88	(58)	CHARACTER	127	SMF55HNM	TCP/IP HOST NAME
215	(D7)	CHARACTER	1	SMF55EN2 (0)	END OF TYPE 55 RECORD (LONG)
Comment					

%IFABGN1 ;

METHOD OF ACCESS

PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM

DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE

INCLUDE MACRO FROM LIBRARY

EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-

DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON

%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD

A DIAGNOSTIC.

THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT

ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER

RECORDS AS FOLLOWS:

MACRO RECORDS

IFASMFR1 07-19

IFASMFR2 20-27

IFASMFR3 28-36

IFASMFR4 37-46

IFASMFR5 47-54

IFASMFR6 55-69

IFASMFR9 80-84

IFASMFRA 85-103

IFASMFRA 104-113

IFASMFRC 114-123

IFASMFRA 124-127

%GOTO IFABGN2;

THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE

REQUIRED FORMAT IS

IFASMFR &RECTYPE

NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).

MODULE NAME = IAZSMF56

DESCRIPTIVE NAME = JES SMF NETWORK INTEGRITY RECORD

End of Comment					
8	(8)	X'8'	0	SMFRCD56	*** START OF RECORD
8	(8)	X'8'	0	SMF56PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF56LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF56SEG	SEGMENT DESCRIPTOR
Comment					

BEGINNING OF JMR OR HASP SMF RECORD

End of Comment					
12	(C)	BITSTRING	1	SMF56FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF56RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF56TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)	CHARACTER	4	SMF56DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF56SID	SYSTEM IDENTIFICATION

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

HEADER FOR HASP SUBSYS RECORD TYPE 56

End of Comment

22	(16)	X'38'	0	SMFJ56	"56" NETWORK INTEGRITY REC TYPE
26	(1A)	BITSTRING	2	SMF56SBS	HASP SUBSYSTEM ID
		.... ..1.		SMF56HSP	"X'0002" JES2 ID
28	(1C)	BITSTRING	2	SMF56SUB	RECORD SUBTYPE
30	(1E)	BITSTRING	2	SMF56LRR	LENGTH OF REST OF RECORD

Comment

BEGINNING OF HASP TYPE 56 NETWORKING SIGNON RECORD AFTER 2 HDRS

End of Comment

32	(20)	CHARACTER	8	SMF56NNM	NODE NAME
40	(28)	BITSTRING	1	SMF56MEM	MEMBER NUMBER
41	(29)	BITSTRING	1	SMF56FG1	SIGNON STATUS FLAGS
		1... ....		SMF56RSO	"B'10000000" RESPONSE SIGNON
		.1.. ....		SMF56CON	"B'01000000" Reset/concur signon
		..1. ....		SMF56SEC	"B'00100000" Secure signon protocol (SMF56LPW and SMF56NPW are not set with this protocol)
42	(2A)	CHARACTER	8	SMF56LPW	LINE PASSWORD
50	(32)	CHARACTER	8	SMF56NPW	NODE PASSWORD
58	(3A)	CHARACTER	8	SMF56LNM	LINE NAME
66	(42)	BITSTRING	4	SMF56BSZ	Negotiated buffer size
70	(46)	CHARACTER	1	SMF56END (0)	END OF TYPE 56 RECORD (SHORT)

Comment

OPTIONAL INFORMATION ABOUT ADJACENT NODE

End of Comment

70	(46)	BITSTRING	16	SMF56IPA	BINARY IP ADDRESS, IPv6 FORMAT
86	(56)	BITSTRING	2	SMF56PRT	PORT NUMBER
88	(58)	CHARACTER	127	SMF56HNM	TCP/IP HOST NAME
215	(D7)	CHARACTER	1	SMF56EN2 (0)	END OF TYPE 56 RECORD (LONG)

Comment

%IFABGN1 ;

METHOD OF ACCESS

PL/S - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM

DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE

INCLUDE MACRO FROM LIBRARY

EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP-

DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON

%INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD

A DIAGNOSTIC.

THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER RECORDS AS FOLLOWS:

MACRO RECORDS

IFASMFR1 07-19

IFASMFR2 20-27

IFASMFR3 28-36

IFASMFR4 37-46

IFASMFR5 47-54

## \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
		IFASMFR6 55-69			
		IFASMFR9 80-84			
		IFASMFR6 85-103			
		IFASMFRB 104-113			
		IFASMFR6 114-123			
		IFASMFRD 124-127			
%GOTO IFABGN2;					
THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS					
IFASMFR &RECTYPE					
NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1).					
MODULE NAME = IAZSMF57					
DESCRIPTIVE NAME = JES SMF NETWORK SYSOUT TRANSMISSION RECORD					
SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC					
End of Comment					
8	(8)	X'8'	0	SMFRCD57	*** START OF RECORD
8	(8)	X'8'	0	SMF57PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF57LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF57SEG	SEGMENT DESCRIPTOR
Comment					
BEGINNING OF JMR OR HASP SMF RECORD					
End of Comment					
12	(C)	BITSTRING	1	SMF57FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF57RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF57TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF57DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF57SID	SYSTEM IDENTIFICATION
Comment					
HEADER FOR HASP SUBSYS RECORD TYPES 57					
End of Comment					
22	(16)	X'39'	0	SMF57J	"57" NETWORK SYSOUT TRANSMISSION
26	(1A)	BITSTRING	2	SMF57SBS	HASP SUBSYSTEM ID
		.... ..1.		SMF57HSP	"X'0002" JES2 ID
28	(1C)	BITSTRING	2	SMF57SUB	RECORD SUBTYPE
30	(1E)	BITSTRING	2	SMF57LRR	LENGTH OF REST OF RECORD
30	(1E)	X'18'	0	SMF57STR	** -SMF57LEN" LENGTH OF HEADING SECTIONS
Comment					
BEGINNING OF HASP TYPE 57 SYSOUT TRANSMISSION RECRD AFTER 2 HDRS					
End of Comment					
32	(20)	CHARACTER	8	SMF57JID	ORIGINAL JOB IDENTIFICATION
40	(28)	CHARACTER	8	SMF57CJD	CURRENT JOB IDENTIFICATION
48	(30)	CHARACTER	8	SMF57ONN	ORIGINAL NODE NAME
56	(38)	CHARACTER	8	SMF57ENN	EXECUTION NODE NAME
64	(40)	CHARACTER	8	SMF57NNN	NEXT NODE NAME
72	(48)	CHARACTER	8	SMF57DVN	SYSOUT TRANSMITTER DEVICE NAME
80	(50)	BITSTRING	4	SMF57TSS	TIME ON SYSOUT TRANSMITTER
84	(54)		4	SMF57DSS	DATE ON SYSOUT TRANSMITTER
88	(58)	BITSTRING	4	SMF57TPS	TIME OFF SYSOUT TRANSMITTER
92	(5C)		4	SMF57DPS	DATE OFF SYSOUT TRANSMITTER
96	(60)	CHARACTER	8	SMF57ACN	NETWORK ACCOUNT NUMBER
104	(68)	CHARACTER	4	SMF57TSI	SYSOUT TRANSMITTER SYSTEM ID
108	(6C)	BITSTRING	4	SMF57CNT	NUMBER OF LOGICAL TP RECORDS
112	(70)	CHARACTER	1	SMF57END (0)	End of type 57 base section

Offsets

Dec	Hex	Type/Value	Len	Name (Dim)	Description
-----	-----	------------	-----	------------	-------------

Comment

This is the header for all future extensions to the SMF 57 record. Sections beyond this point must be accessed by using the values stored in the triplets (below) that contain the offset, length, and number of sections of the type corresponding to the triplet. New sections will be appended to this header and their presence can be detected by an increase in the number of triplets and by a non-zero section offset, length and number of sections.

End of Comment

112	(70)	BITSTRING	2	SMF57NTR	NUMBER OF TRIPLETS
114	(72)	BITSTRING	2		RESERVED
114	(72)	X'74'	0	SMF57TRP	*** BEGINNING OF TRIPLETS
116	(74)	BITSTRING	4	SMF57OSW	OFFSET TO ESS SECTION
120	(78)	BITSTRING	2	SMF57LSW	LENGTH OF ESS SECTION
122	(7A)	BITSTRING	2	SMF57NSW	NUMBER OF ESS SECTIONS
122	(7A)	X'8'	0	SMF57LTP	**SMF57TRP" LENGTH OF TRIPLETS
122	(7A)	X'1'	0	SMF57NTP	"SMF57LTP/8" NUMBER OF TRIPLETS
122	(7A)	X'C'	0	SMF57TPL	**SMF57NTR" Length of Triplets section and number of triplets

Comment

Enhanced SYSOUT Support (ESS) Ssection  
 This section contains the OUTPUT descriptor (if any) in SWBTU format (IEFSJPFx plus text units) for the first offloaded data set included in this SMF record. The SWBTU may be processed using the SWBTUREC macro or other Scheduler JCL Facility (SJF) services.

End of Comment

124	(7C)	BITSTRING	2	SMF57LN1	LENGTH OF ESS SECTION
126	(7E)	BITSTRING	4	SMF57SGT	SEGMENT IDENTIFIER
130	(82)	BITSTRING	1	SMF57IND	ESS SECTION INDICATOR
		1... ..		SMF57SJF	"B'10000000" ERROR OBTAINING SWBTU (SWBTU DATA NOT PRESENT)
131	(83)	BITSTRING	1		RESERVED
132	(84)	CHARACTER	8	SMF57JDT	JDVT NAME
140	(8C)	BITSTRING	2	SMF57TUL	SWBTU DATA AREA LENGTH
142	(8E)	CHARACTER	1	SMF57TU (0)	SWBTU DATA AREA
142	(8E)	X'12'	0	SMF57ESL	**SMF57LN1" Length of the fixed portion of the ESS section

# \$SMF Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
%IFABGN1: ; METHOD OF ACCESS PLS - USER DEFINES MACRO VARIABLE IFARXX WHERE XX=RECORD NUM DECLARE SMFXXPTR PTR(31) OR RESPECIFY FOR BASE INCLUDE MACRO FROM LIBRARY EXAMPLE %IFAR08 = 'YES', NOTE. THE COMMA REP- DCL SMF08PTR PTR(31), RESENTS A SEMI-COLON %INCLUDE SYSLIB(IFASMFR), BUT ISN'T TO AVIOD A DIAGNOSTIC. THIS MACRO PROCESSES RECORDS IN THE RANGE 00-06. IT ACTS AS A ROUTER TO OTHER MACROS TO PROCESS OTHER RECORDS AS FOLLOWS: MACRO RECORDS IFASMFR1 07-19 IFASMFR2 20-27 IFASMFR3 28-36 IFASMFR4 37-46 IFASMFR5 47-54 IFASMFR6 55-69 IFASMFR9 80-84 IFASMFR8 85-103 IFASMFRB 104-113 IFASMFRC 114-123 IFASMFRD 124-127					
%GOTO IFABGN2; THIS IS AN SMF MACRO WHICH CONTROLS THE BUILDING OF SMF RECORDS. THE REQUIRED FORMAT IS IFASMFR &RECTYPE NOTE: VALUES FOR &RECTYPE MUST BE ENCLOSED IN PARENS(UNLESS ONLY 1). MODULE NAME = IAZSMF58 DESCRIPTIVE NAME = JES SMF NETWORK SIGNOFF RECORD SWITCH TO DETERMINE WHETHER TO GENERATE EQUATES FOR WRITING REC					
End of Comment					
8	(8)	X'8'	0	SMFRCD58	*** START OF RECORD
8	(8)	X'8'	0	SMF58PTR	*** HEADER SEGMENT
8	(8)	CHARACTER	2	SMF58LEN	RECORD LENGTH
10	(A)	CHARACTER	2	SMF58SEG	SEGMENT DESCRIPTOR
Comment					
BEGINNING OF JMR OR HASP SMF RECORD					
End of Comment					
12	(C)	BITSTRING	1	SMF58FLG	HEADER FLAG BYTE
13	(D)	BITSTRING	1	SMF58RTY	RECORD TYPE
14	(E)	BITSTRING	4	SMF58TME	TOD, USING FORMAT FROM TIME MACRO
18	(12)		4	SMF58DTE	DATE IN PACKD DECIMAL FORM 00YYDDDF
22	(16)	CHARACTER	4	SMF58SID	SYSTEM IDENTIFICATION
Comment					
HEADER FOR HASP SUBSYS RECORD TYPES 58					
End of Comment					
22	(16)	X'3A'	0	SMFJ58	"58" NETWORK SIGNOFF REC TYPE
26	(1A)	BITSTRING	2	SMF58SBS	HASP SUBSYSTEM ID
		.... ..1.		SMF58HSP	"X'0002" JES2 ID
28	(1C)	BITSTRING	2	SMF58SUB	RECORD SUBTYPE
30	(1E)	BITSTRING	2	SMF58LRR	LENGTH OF REST OF RECORD
30	(1E)	X'18'	0	SMF58STR	**-SMF58LEN" LENGTH OF HEADING SECTIONS

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
BEGINNING OF HASP TYPE 58 NETWORK SIGNOFF RECORD AFTER TWO HDRS					
End of Comment					
32	(20)	CHARACTER	8	SMF58NNM	NODE NAME
40	(28)	BITSTRING	1	SMF58MEM	MEMBER NUMBER
41	(29)	BITSTRING	1	SMF58RV1	RESERVED
42	(2A)	CHARACTER	8	SMF58LNM	LINE NAME
50	(32)	CHARACTER	1	SMF58END (0)	END OF TYPE 58 RECORD
Comment					
THE FOLLOWING ORGS ENSURE THAT A JES2 SMF BUFFER IS AS LARGE AS THE LARGEST SMF RECORD (PLUS THE BUFFER PREFIX) THAT JES2 WRITES. THE LENGTH OF EACH RECORD, OTHER THAN THE TYPE 6 AND 26, IS HANDLED BY THE FIRST 'ORG ,'. THE TYPE 6 AND 26 CONTAIN SEVERAL SECTIONS, AND THEIR LENGTHS ARE THEREFORE DEFINED USING THE EQUATIONS BELOW.					
End of Comment					
272	(110)	SIGNED	4	(0)	
920	(398)	SIGNED	4	(0)	
388	(184)	SIGNED	4	(0)	
920	(398)	SIGNED	4	(0)	
920	(398)	X'398'	0	SMFLNG	**SMF" LEN OF LARGEST RECORD

**\$SMF Cross Reference**

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
NJEACCT	BC		SMFJ48	D	30
NJEBLDG	CC		SMFJ49	D	31
NJEDEPT	C4		SMFJ52	16	34
NJEJMR	84		SMFJ53	16	35
NJEJMREN	E8		SMFJ54	16	36
NJEJMRID	84	D5D1C5D1	SMFJ55	16	37
NJEJMLN	8C		SMFJ56	16	38
NJEJMRSZ	E8		SMFJ57	16	39
NJEJOBNM	90		SMFJ58	16	3A
NJEJOBNO	8E		SMFJ6	D	6
NJEJOBNX	E4		SMFLEN	8	
NJEPRGMR	A0		SMFLNG	398	398
NJEROOM	D4		SMFLNHDR	7	8
NJETRANS	E4		SMFLRGTP	4	40
NJEUSRID	B4		SMFNO26	5	1
NJEXEQN	98		SMFPARM	7	
NJEXEQU	DC		SMFQUED	4	20
SMF	0		SMFRCD24	8	8
SMF	0		SMFRCD26	8	8
SMF\$ESYS	22	80	SMFRCD43	8	8
SMFAPPC	6	1	SMFRCD45	8	8
SMFCBIE	3F	1	SMFRCD47	8	8
SMFCHAIN	0		SMFRCD48	8	8
SMFCLFLG	6		SMFRCD49	8	8
SMFJMR	A	C	SMFRCD52	8	8
SMFJMRCH	8		SMFRCD53	8	8
SMFJMRTTP	4	80	SMFRCD54	8	8
SMFJ24	16	18	SMFRCD55	8	8
SMFJ26	D	1A	SMFRCD56	8	8
SMFJ43	D	2B	SMFRCD57	8	8
SMFJ45	D	2D	SMFRCD58	8	8
SMFJ47	D	2F	SMFRCD6	8	8

## \$SMF Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SMFRDW	8		SMF24OPS	24	
SMFSEG	A		SMF24ORG	B0	40404040
SMFTYPE	4		SMF24OSA	44	
SMFWFL26	5		SMF24OSP	34	
SMF24AFF	C2	40	SMF24OSW	3C	
SMF24AOF	FC	E8	SMF24PNM	4E	
SMF24BCF	5A	0	SMF24POF	FC	44
SMF24CJD	6C	40404040	SMF24PRD	FC	C
SMF24CNT	A4	0	SMF24PRM	E4	40404040
SMF24COM	5B	80	SMF24PRY	FC	0
SMF24CON	5A	40	SMF24PTR	8	8
SMF24DDS	AC	0	SMF24PVR	4C	
SMF24DRD	BC	0	SMF24REC	F8	0
SMF24DSN	78	40404040	SMF24RSV	22	
SMF24DTE	12	C	SMF24RS2	56	
SMF24EJS	F0		SMF24RTY	D	0
SMF24EOJ	5B	0	SMF24SAC	FC	
SMF24ESL	10F	12	SMF24SAL	FC	C
SMF24FCB	D4	40404040	SMF24SAN	F4	
SMF24FLG	C	0	SMF24SBT	C2	40
SMF24FLS	E0	40404040	SMF24SCL	C3	40
SMF24FOR	D8	40404040	SMF24SDS	5B	40
SMF24FST	5A	80	SMF24SEG	A	
SMF24GLN	58	0	SMF24SFG	C2	0
SMF24GOF	FC	50	SMF24SGT	FF	
SMF24INC	C2	10	SMF24SHL	C2	80
SMF24IND	103		SMF24SID	16	40404040
SMF24INJ	5B	20	SMF24SIF	103	80
SMF24JAF	CC	40404040	SMF24SJH	C2	20
SMF24JBN	5C	40404040	SMF24SND	C4	40404040
SMF24JCL	C3	40	SMF24SOF	FC	B8
SMF24JDT	105		SMF24SR	1E	4
SMF24JFG	C2	0	SMF24SRN	CC	40404040
SMF24JHL	C2	80	SMF24SRT	C4	
SMF24JID	64	40404040	SMF24SSI	1A	
SMF24JND	C4	40404040	SMF24ST	1E	3
SMF24JR	1E	2	SMF24STS	C	40
SMF24JRT	C4		SMF24SUB	1E	0
SMF24JT	1E	1	SMF24SYS	74	40404040
SMF24LEN	8		SMF24TDS	A8	0
SMF24LGN	30		SMF24TME	E	0
SMF24LHD	22	1C	SMF24TPS	22	24
SMF24LN	FC	68	SMF24TRD	B8	0
SMF24LN1	C0	0	SMF24TRP	4A	28
SMF24LN2	C0	0	SMF24TU	10F	
SMF24LN3	FD	0	SMF24TUL	10D	
SMF24LN4	F8		SMF24UCS	EC	40404040
SMF24LPS	28		SMF24WID	F0	40404040
SMF24LSA	48		SMF248CL	E8	40404040
SMF24LSP	38		SMF26ACP	38	100
SMF24LST	5A	20	SMF26ACT	70	
SMF24LSW	40		SMF26AC1	C0	
SMF24LS4	F0	0	SMF26ASP	36	5
SMF24L1	FC	30	SMF26ATU	38	2000
SMF24L2	FC	3D	SMF26BCH	3E	80
SMF24MUL	C2	8	SMF26CID	4E	
SMF24NGN	32		SMF26CLN	B8	
SMF24NPS	2A		SMF26CLS	61	
SMF24NSA	4A		SMF26CPD	52	
SMF24NSP	3A		SMF26CPT	4E	
SMF24NSW	42		SMF26CPY	3E	2
SMF24NTP	4A	5	SMF26CSD	4A	
SMF24NTR	20		SMF26CST	46	
SMF24OGN	2C		SMF26CYP	88	
SMF24OPR	5B	10	SMF26DDT	B4	



**\$SMF Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>	<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SMF26DES	38	8000	SMF26L6	C0	1A
SMF26DEV	68		SMF26L7	C0	1A
SMF26DJC	3E	80	SMF26L9	C0	29
SMF26DJE	3E	1	SMF26MRE	3F	2
SMF26DJO	3E	2	SMF26MSG	60	
SMF26DLJ	3E	40	SMF26NAC	52	
SMF26DLM	3E	20	SMF26NAG	42	
SMF26DTE	12		SMF26NAM	4C	
SMF26DTM	B0		SMF26NDV	62	
SMF26DTY	84		SMF26NET	A8	
SMF26EBT	3C		SMF26NID	3E	
SMF26EFM	4C		SMF26NJB	5A	
SMF26EIR	3F	4	SMF26NJC	52	
SMF26ELN	7C		SMF26NJE	3E	4
SMF26EPG	44		SMF26NJI	A0	
SMF26EPU	80		SMF26NJO	98	
SMF26EVT	38	4000	SMF26NJV	3E	8
SMF26FLG	C		SMF26NLG	3F	10
SMF26FRM	84		SMF26NLN	82	
SMF26FTS	3E	40	SMF26NN	92	
SMF26HSP	36	2	SMF26NNM	7A	
SMF26ICD	3E		SMF26NOJ	3E	10
SMF26IGP	88		SMF26NON	6A	
SMF26IND	38		SMF26NOU	3E	8
SMF26INF	3F		SMF26NPD	4E	
SMF26INR	3C		SMF26NPT	4A	
SMF26IN2	3E		SMF26NRA	BF	
SMF26IN3	3E		SMF26NSD	46	
SMF26IX2	64		SMF26NST	42	
SMF26JAF	5A		SMF26NTW	38	1000
SMF26JBF	3E	1	SMF26NU	9A	
SMF26JBN	1A		SMF26NWL	4A	
SMF26JCP	3F	80	SMF26NXN	72	
SMF26JCR	54		SMF26OAG	3C	
SMF26JDL	64	80	SMF26OID	56	
SMF26JID	44		SMF26OJC	4C	
SMF26JNM	40		SMF26OPC	3F	1
SMF26JOL	64	40	SMF26OPD	72	
SMF26JXP	38	400	SMF26OPS	65	
SMF26J2D	80	84	SMF26OPT	6E	
SMF26J2R	38	800	SMF26OSD	6A	
SMF26J3D	84	84	SMF26OST	66	
SMF26LA	C0	40	SMF26OWL	44	
SMF26LAG	40		SMF26PDD	90	
SMF26LEN	8		SMF26PD3	90	
SMF26LIN	8A		SMF26PRD	40	
SMF26LJC	50		SMF26PRJ	3E	10
SMF26LN	C0	32	SMF26PRR	8C	
SMF26LN1	3A		SMF26PTR	8	8
SMF26LN2	3A		SMF26PUD	44	
SMF26LN3	3A		SMF26PUR	8E	
SMF26LN4	3A		SMF26RID	4A	
SMF26LN5	3A		SMF26ROM	74	
SMF26LN6	3A		SMF26RPD	42	
SMF26LN7	3A		SMF26RPT	3E	
SMF26LN8	BD		SMF26RSD	26	
SMF26LOC	66		SMF26RST	22	
SMF26LPN	64	20	SMF26RSV	32	
SMF26LWL	48		SMF26RTY	D	
SMF26L1	C0	5E	SMF26RVA	70	
SMF26L10	C0	83	SMF26RV1	3C	
SMF26L2	C0	3C	SMF26RV2	3C	
SMF26L3	C0	20	SMF26RV4	3C	
SMF26L4	C0	68	SMF26RV5	3C	
SMF26L5	C0	E	SMF26RV6	85	

## \$SMF Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SMF26RV8	64		SMF43RTY	D	
SMF26R02	38	200	SMF43RVJ	32	
SMF26SBS	36		SMF43RVU	3A	
SMF26SCN	3E	4	SMF43RV1	20	
SMF26SEG	A		SMF43RV2	23	
SMF26SID	16		SMF43SBG	22	23
SMF26SJB	A4	40	SMF43SBP	1E	20
SMF26SRC	E8		SMF43SBS	1A	
SMF26STK	3E	20	SMF43SEG	A	
SMF26STU	3F	40	SMF43SID	16	
SMF26SUI	8A		SMF43SIZ	3E	
SMF26SZ1	C0	10	SMF43TME	E	
SMF26SZ2	C0	4C	SMF43UNT	26	26
SMF26SZ3	C0	8	SMF43UN4	26	26
SMF26SZ4	C0	4	SMF43US1	24	
SMF26SZ5	C0	4F	SMF43WRM	22	40
SMF26TLN	C0	17C	SMF45ABN	20	80
SMF26TME	E		SMF45ABT	28	20
SMF26TRH	3F	20	SMF45ATM	20	80
SMF26UIF	2A		SMF45DSI	20	40
SMF26WCL	94		SMF45DTE	12	
SMF26WIN	A4		SMF45END	32	
SMF26WJC	A5		SMF45FG1	20	
SMF26WLM	A4	80	SMF45FLG	C	
SMF26WOC	9C		SMF45HLT	28	80
SMF26WSE	AD		SMF45HSP	1A	2
SMF26XBC	3F	8	SMF45IND	20	
SMF26XBT	40		SMF45JCC	22	
SMF26XID	52		SMF45J3C	21	
SMF26XLN	42		SMF45LEN	8	
SMF26XPD	62		SMF45LRR	1E	
SMF26XPG	48		SMF45L1	2C	1C
SMF26XPI	62		SMF45L2	2C	4
SMF26XPS	63		SMF45NOS	28	10
SMF26XPT	5E		SMF45NUL	2C	
SMF26XPU	46		SMF45OPS	28	40
SMF26XSD	5A		SMF45PRC	20	40404040
SMF26XST	56		SMF45PTR	8	8
SMF26XTM	78		SMF45RSV	1C	
SMF26XWR	64	10	SMF45RTY	D	
SMF43ANL	22	10	SMF45RVJ	26	
SMF43CLD	22	80	SMF45RVU	2E	
SMF43DSI	22	1	SMF45RV1	24	
SMF43DTE	12		SMF45SBG	1E	20
SMF43EID	24		SMF45SBS	1A	
SMF43END	3E		SMF45SEG	A	
SMF43FLG	C		SMF45SID	16	
SMF43GBL	22	8	SMF45SIZ	32	
SMF43HOT	22	20	SMF45STF	28	
SMF43HSP	1A	2	SMF45TME	E	
SMF43INT	28	40404040	SMF45UID	29	
SMF43LCL	22	4	SMF45US1	25	
SMF43LEN	8		SMF47CWK	20	20
SMF43LRR	1E		SMF47DPL	34	
SMF43L1	28	20	SMF47DSL	30	
SMF43L2	28	8	SMF47DTE	12	
SMF43NMU	25		SMF47END	6E	
SMF43OPT	23		SMF47EVT	20	
SMF43ORG	26		SMF47FLG	C	
SMF43PJ3	2E		SMF47HSP	1A	2
SMF43PRC	20	40404040	SMF47JID	28	
SMF43PTR	8	8	SMF47LAA	2E	
SMF43REF	22	2	SMF47LCF	20	
SMF43RST	22		SMF47LEN	8	
SMF43RSV	1C		SMF47LIN	2C	

**\$SMF Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>	<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SMF47LN1	22		SMF48L1	58	4F
SMF47LN2	3C		SMF48NAK	40	
SMF47LON	20	4	SMF48NKS	46	
SMF47LRR	1E		SMF48OUT	48	
SMF47LTC	24		SMF48PSW	34	
SMF47L1	20	1A	SMF48PTR	8	8
SMF47L2	20	26	SMF48RMT	24	
SMF47L3	20	5A	SMF48RPI	20	80
SMF47L4	20	34	SMF48RSV	1C	
SMF47MSG	3E		SMF48RTY	D	
SMF47PCI	2F		SMF48RVJ	5F	
SMF47PLI	25		SMF48RVU	63	
SMF47PSW	34		SMF48RV1	22	
SMF47PTR	8	8	SMF48SBG	34	3C
SMF47RMT	24		SMF48SBP	1E	20
SMF47RSV	1C		SMF48SBS	1A	
SMF47RTY	D		SMF48SEG	A	
SMF47RVJ	62		SMF48SID	16	
SMF47RVU	6A		SMF48SIZ	67	
SMF47SBG	1E	20	SMF48SOF	20	1
SMF47SBS	1A		SMF48SPA	20	4
SMF47SEG	A		SMF48S0	48	
SMF47SID	16		SMF48S1	49	
SMF47SIZ	6E		SMF48S2	4A	
SMF47SON	20	1	SMF48S3	4B	
SMF47SPA	20	4	SMF48S4	4C	
SMF47STL	20	2	SMF48S5	4D	
SMF47TME	E		SMF48S6	4E	
SMF47UID	21		SMF48TID	58	
SMF47ULK	20	1	SMF48TIO	38	
SMF47UNL	20	80	SMF48TME	E	
SMF48ADP	58		SMF48TOT	44	
SMF48AD4	5B		SMF48TRN	3C	
SMF48CAN	20	2	SMF48UID	21	
SMF48CDH	4C		SMF48ULK	20	1
SMF48CNI	20	40	SMF48USR	4F	
SMF48CNT	28		SMF48VIR	30	
SMF48CPD	44		SMF48VSD	50	
SMF48CPG	20	10	SMF48XCP	3C	
SMF48CPU	24		SMF49BND	20	8
SMF48CT	58	14	SMF49DEF	20	6
SMF48CTD	48		SMF49DTE	12	
SMF48CTH	2C		SMF49END	62	
SMF48CWK	20	20	SMF49EVT	20	
SMF48DCK	44		SMF49FLG	C	
SMF48DID	54		SMF49FLN	28	
SMF48DIO	34		SMF49LEN	8	
SMF48DSL	3C		SMF49LER	20	4
SMF48DSP	40		SMF49LIM	20	5
SMF48DTE	12		SMF49LIN	2C	
SMF48END	67		SMF49LNA	23	
SMF48ERR	4C		SMF49LN1	22	
SMF48ERS	40		SMF49LN2	3C	
SMF48EVT	20		SMF49LRR	1E	
SMF48FLG	C		SMF49L1	20	1A
SMF48FLS	20		SMF49L2	20	26
SMF48HSI	20	8	SMF49L3	20	5A
SMF48HSP	1A	2	SMF49MSG	3E	
SMF48IO	3C		SMF49NER	20	1
SMF48LAA	50		SMF49PER	20	2
SMF48LA4	53		SMF49PSW	34	
SMF48LEN	8		SMF49PTR	8	8
SMF48LIN	2C		SMF49RMT	24	
SMF48LOF	20	4	SMF49RSV	1C	
SMF48LRR	1E		SMF49RTY	D	

## \$SMF Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SMF49RV1	26		SMF53SEG	A	
SMF49SBG	1E	20	SMF53SID	16	40404040
SMF49SBS	1A		SMF53SUB	26	0
SMF49SEG	A		SMF53SYS	2A	D1C5E2F2
SMF49SID	16		SMF53TME	E	0
SMF49SIZ	62		SMF53VER	28	F0F1
SMF49SON	20	1	SMF54DTE	12	C
SMF49SPW	20	7	SMF54END	46	
SMF49STL	20	2	SMF54FLG	C	0
SMF49TER	20	8	SMF54IDL	22	0
SMF49TME	E		SMF54IDN	24	0
SMF49UFO	34		SMF54IDO	20	0
SMF49VID	20		SMF54IDS	2E	
SMF52DTE	12	C	SMF54LEN	8	
SMF52END	46		SMF54LON	26	1
SMF52FLG	C	0	SMF54POF	1A	0
SMF52IDL	22	0	SMF54PRL	1C	0
SMF52IDN	24	0	SMF54PRN	1E	0
SMF52IDO	20	0	SMF54PSW	3E	40404040
SMF52IDS	2E		SMF54PTR	8	8
SMF52LEN	8		SMF54RMT	2E	40404040
SMF52LID	46	18	SMF54RPW	36	40404040
SMF52LIN	36	40404040	SMF54RTY	D	0
SMF52LON	26	1	SMF54SEG	A	
SMF52LPR	46	8	SMF54SID	16	40404040
SMF52OFI	46	26	SMF54SUB	26	0
SMF52OFF	46	1E	SMF54SYS	2A	D1C5E2F2
SMF52POF	1A	0	SMF54TME	E	0
SMF52PRL	1C	0	SMF54VER	28	F0F1
SMF52PRN	1E	0	SMF55BSZ	42	0
SMF52PSW	3E	40404040	SMF55CON	29	40
SMF52PTR	8	8	SMF55DTE	12	C
SMF52RMT	2E	40404040	SMF55END	46	
SMF52RTY	D	0	SMF55EN2	D7	
SMF52SEG	A		SMF55FG1	29	0
SMF52SID	16	40404040	SMF55FLG	C	0
SMF52SLN	26	2	SMF55HNM	58	40404040
SMF52SUB	26	0	SMF55HSP	1A	2
SMF52SYS	2A	D1C5E2F2	SMF55IPA	46	0
SMF52TME	E	0	SMF55LEN	8	
SMF52VER	28	F0F1	SMF55LNM	3A	40404040
SMF53ADP	5A	E2D5C1	SMF55LPW	2A	40404040
SMF53CTR	46	0	SMF55LRR	1E	0
SMF53DTE	12	C	SMF55MEM	28	0
SMF53END	5D		SMF55NNM	20	40404040
SMF53FLG	C	0	SMF55NPW	32	40404040
SMF53IDL	22	0	SMF55PRT	56	0
SMF53IDN	24	0	SMF55PTR	8	8
SMF53IDO	20	0	SMF55RSO	29	80
SMF53IDS	2E		SMF55RTY	D	0
SMF53LEN	8		SMF55SBS	1A	2
SMF53LID	5D	2F	SMF55SEC	29	20
SMF53LIN	36	40404040	SMF55SEG	A	
SMF53LOF	26	1	SMF55SID	16	40404040
SMF53LPR	5D	8	SMF55STR	1E	18
SMF53OFI	5D	26	SMF55SUB	1C	0
SMF53OFF	5D	1E	SMF55TME	E	0
SMF53PLN	26	2	SMF56BSZ	42	0
SMF53PRL	1C	0	SMF56CON	29	40
SMF53PRN	1E	0	SMF56DTE	12	C
SMF53PRO	1A	0	SMF56END	46	
SMF53PSW	3E	40404040	SMF56EN2	D7	
SMF53PTR	8	8	SMF56FG1	29	0
SMF53RMT	2E	40404040	SMF56FLG	C	0
SMF53RTY	D	0	SMF56HNM	58	40404040

**\$\$SMF Cross Reference**

<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>	<b>Name</b>	<b>Hex Offset</b>	<b>Hex Value</b>
SMF56HSP	1A	2	SMF58HSP	1A	2
SMF56IPA	46	0	SMF58LEN	8	
SMF56LEN	8		SMF58LNM	2A	40404040
SMF56LNM	3A	40404040	SMF58LRR	1E	0
SMF56LPW	2A	40404040	SMF58MEM	28	0
SMF56LRR	1E	0	SMF58NNM	20	40404040
SMF56MEM	28	0	SMF58PTR	8	8
SMF56NNM	20	40404040	SMF58RTY	D	0
SMF56NPW	32	40404040	SMF58RV1	29	0
SMF56PRT	56	0	SMF58SBS	1A	2
SMF56PTR	8	8	SMF58SEG	A	
SMF56RSO	29	80	SMF58SID	16	40404040
SMF56RTY	D	0	SMF58STR	1E	18
SMF56SBS	1A	2	SMF58SUB	1C	0
SMF56SEC	29	20	SMF58TME	E	0
SMF56SEG	A		SMF6APAL	77	
SMF56SID	16	40404040	SMF6APA1	77	1
SMF56SUB	1C	0	SMF6BID	6B	
SMF56TME	E	0	SMF6BIN	74	
SMF57ACN	60	40404040	SMF6BIN1	74	80
SMF57CJD	28	40404040	SMF6BIN2	74	40
SMF57CNT	6C	0	SMF6BIN3	74	20
SMF57DPS	5C	C	SMF6BIN4	74	10
SMF57DSS	54	C	SMF6BNCT	9	
SMF57DTE	12	C	SMF6BNLE	C	
SMF57DVN	48	40404040	SMF6BNLN	8	
SMF57END	70		SMF6BNNO	8	
SMF57ENN	38	40404040	SMF6BNOF	4A	
SMF57ESL	8E	12	SMF6BNUM	A	
SMF57FLG	C	0	SMF6BNWI	E	
SMF57HSP	1A	2	SMF6BTS	6B	80
SMF57IND	82		SMF6BYTE	4A	
SMF57JDT	84		SMF6CCE	64	2
SMF57JID	20	40404040	SMF6CHR	52	
SMF57LEN	8		SMF6CPS	4A	
SMF57LN1	7C		SMF6CSP	6B	20
SMF57LRR	1E	0	SMF6DCI	4A	
SMF57LSW	78		SMF6DCRV	4A	80
SMF57LTP	7A	8	SMF6DDNM	7E	
SMF57NNN	40	40404040	SMF6DEND	EA	
SMF57NSW	7A		SMF6DFE	64	
SMF57NTP	7A	1	SMF6DIE	3F	4
SMF57NTR	70		SMF6DPGL	75	10
SMF57ONN	30	40404040	SMF6DPLS	76	10
SMF57OSW	74		SMF6DSIZ	EA	
SMF57PTR	8	8	SMF6DSNM	9E	
SMF57RTY	D	0	SMF6DTE	12	C
SMF57SBS	1A	2	SMF6DUPS	75	80
SMF57SEG	A		SMF6DUPT	75	40
SMF57SGT	7E		SMF6EEND	5A	
SMF57SID	16	40404040	SMF6EFMN	4E	
SMF57SJF	82	80	SMF6END	7C	
SMF57STR	1E	18	SMF6END2	66	
SMF57SUB	1C	0	SMF6ESIZ	5A	
SMF57TME	E	0	SMF6ESS1	45	10
SMF57TPL	7A	C	SMF6FCB	58	
SMF57TPS	58	0	SMF6FDNM	84	
SMF57TRP	72	74	SMF6FEET	68	
SMF57TSI	68	40404040	SMF6FEND	6C	
SMF57TSS	50	0	SMF6FEXT	45	80
SMF57TU	8E		SMF6FLC	6A	
SMF57TUL	8C		SMF6FLG	C	0
SMF58DTE	12	C	SMF6FLG3	76	
SMF58END	32		SMF6FLI	66	
SMF58FLG	C	0	SMF6FMDF	70	

## \$SMF Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
SMF6FMN	41	40404040	SMF6PGDF	6C	
SMF6FONT	4C		SMF6PGE	60	
SMF6FSIZ	6C		SMF6PGOP	75	
SMF6FTFR	45	8	SMF6PGSG	5C	
SMF6FTL	52		SMF6PQLN	5E	
SMF6FTL1	52	1	SMF6PRMD	96	
SMF6GRP	68		SMF6PRNM	76	
SMF6IGER	75	1	SMF6PRTQ	60	
SMF6IMPS	64		SMF6PTDV	94	
SMF6IND	4E		SMF6RBE	64	1
SMF6INDC	4B		SMF6REND	56	
SMF6INT	4A	1	SMF6RES	4A	
SMF6IOE	3F	0	SMF6REXT	45	40
SMF6IP1	4E		SMF6ROR	4A	4
SMF6IP2	4F		SMF6ROUT	4A	
SMF6IP3	50		SMF6RSD	26	C
SMF6IP4	51		SMF6RSIZ	56	
SMF6JBID	66		SMF6RST	22	0
SMF6JBN	1A	40404040	SMF6RSV	4F	
SMF6JDVT	50		SMF6RSVJ	70	
SMF6JHPP	76	40	SMF6RSVU	78	
SMF6JNM	4C		SMF6RTE	64	
SMF6JTPP	76	20	SMF6RTY	D	0
SMF6J2L3	4B	3	SMF6SBS	46	
SMF6J2L4	4B	4	SMF6SDS	4A	40
SMF6J2S	60	64	SMF6SECS	8E	
SMF6J3L3	4B	3	SMF6SEG	A	
SMF6J3L4	4B	4	SMF6SEND	C0	
SMF6J3S	64	64	SMF6SETU	9C	
SMF6LEN	8		SMF6SEXT	45	20
SMF6LEV2	4B	1	SMF6SGID	4A	
SMF6LEV3	4B	1	SMF6SID	16	40404040
SMF6LEV4	4B	5	SMF6SIZ	7C	
SMF6LEV6	4B	6	SMF6SIZ2	66	
SMF6LEV7	4B	7	SMF6SIZ3	66	
SMF6LFNT	50		SMF6SJF	4E	80
SMF6LN1	48		SMF6SLIG	76	80
SMF6LN2	48		SMF6SOER	75	2
SMF6LN3	48		SMF6SPGL	75	4
SMF6LN4	48		SMF6SSIZ	C0	
SMF6LN5	48		SMF6STNM	6E	
SMF6LN6	48		SMF6SUCC	75	8
SMF6LOLY	58		SMF6SYSA	75	20
SMF6LPGE	BC		SMF6TEND	78	
SMF6LPSG	60		SMF6TME	E	0
SMF6LSIZ	7C		SMF6TSIZ	78	
SMF6MID	62		SMF6TU	5A	
SMF6NDS	40	0	SMF6TUL	58	
SMF6NLR	3B	0	SMF6UCS	5C	
SMF6NSFO	7C		SMF6UIF	2A	40404040
SMF6NSOL	78		SMF6UPAS	76	8
SMF6NSPS	80		SMF6URI	78	
SMF6OCN	4A	20	SMF6URIL	5C	
SMF6OCNM	9C		SMF6USID	86	
SMF6OPJ	6B	40	SMF6WSD	37	C
SMF6OPR	66		SMF6WST	33	0
SMF6OR	4A	8			
SMF6ORD	4A	10			
SMF6OSS	4A	2			
SMF6OTOK	D6				
SMF6OUT	50				
SMF6OVLY	54				
SMF6OWC	32	40			
SMF6PAD1	45	0			
SMF6PDNM	8C				

## \$SNFWORK Information

### \$SNFWORK Heading Information

**Common Name:** JES2 SPOOL Sniffer Work Area  
**Macro ID:** \$SNFWORK  
**DSECT Name:** PCE (\$SNFWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol SNWPCEWS for the length of this work area. The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE  
**Pointed to by:** The \$SNFPCE field of the \$HCT data area  
 See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization

**Function:** The fields in this area are used by the JES2 SPOOL Management Processor and by its support routines and exits. \$SNFWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$SNFWORK are actually part of the PCE DSECT, but only map PCEs with the value PCESNFID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$SNFWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	BITSTRING	6	SNWLSNIF (0)	Extent, offset within extent and bit within byte of last sniffed
318	(13E)	BITSTRING	2		Reserved for future use
320	(140)	ADDRESS	4	SNWQUEUE	Anchor for immediate work to do queue
324	(144)	BITSTRING	1	SNWFLAG1	Flags
		...1. ....		SNW1BADT	"B'00100000" Sniffing bad trackgroup
		.... 1...		SNW1MFID	"B'00001000" Inactive DAS found during examination of DASes for mini fmt
325	(145)	BITSTRING	3		Available for use by IBM
328	(148)	SIGNED	4		Reserved for future use
332	(14C)	BITSTRING	12	SNWTQE	TQE for SNFWAIT processing
344	(158)	BITSTRING	5	SNWMQT	MQT for next sniffing
349	(15D)	BITSTRING	1		Reserved for future use

## \$SNFWORK Cross Reference

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
Comment					
<p>-----</p> <p>The following field represents the MOOB (extent offset bit) of the track group of a DAS whose DAS3EFWZ flag is off. This track group signature record will be re-written with zeros if its SIGJBKEY field in signature record contains FFs and TG is not allocated.</p> <p>-----</p>					
End of Comment					
350	(15E)	BITSTRING	6	SNWMFMOB	Extent number, Extent TG offset and bit of track group of a DAS whose DAS3EFWZ is off(will be mini reformatted with zeros)
360	(168)	DBL WORD	8	(0)	Force double-word alignment
360	(168)	X'30'	0	SNWPCEWS	**_PCEWORK" Length of work area

## \$SNFWORK Cross Reference

Name	Hex Offset	Hex Value
PCE	0	
SNWFLAG1	144	
SNWLSNIF	138	
SNWMFMOB	15E	
SNWMQT	158	
SNWPCEWS	168	30
SNWQUEUE	140	
SNWTQE	14C	
SNW1BADT	144	20
SNW1MFID	144	8



## \$SPIWORK Information

### \$SPIWORK Heading Information

**Common Name:** JES2 Sysout API Work Area  
**Macro ID:** \$SPIWORK  
**DSECT Name:** PCE (\$SPIWORK is part of the PCE DSECT)  
**Owning Component:** JES2 (SC1BH)  
**Eye-Catcher ID:** 'PCE '  
 Offset: PCEEYE-PCE  
 Length: 4

**Storage Attributes:** Subpool: See \$PCE  
 Key: See \$PCE  
 Residency: See \$PCE

**Size:** See symbol SPIWKSIZ for the length of this work area. The overall length of the PCE is stored in field PCELENG.

**Created by:** See \$PCE  
**Pointed to by:** The \$SPIPCE field of the \$HCT data area points into the \$PCEORG/\$PCELAST chain to the first SYSOUT API PCE. Since this chain contains all PCEs, use the PCEID field to determine when you have chained past the last PCE of this type. See \$PCE for other pointer fields that apply to all PCE types.

**Serialization:** Normal PCE dispatch serialization

**Function:** The fields in this work area are used by a JES2 Sysout API Processor and by its support routines and exits. \$SPIWORK maps the variable PCE work area that begins at label PCEWORK. The fields defined in \$SPIWORK are actually part of the PCE DSECT, but only map PCEs with the value PCESPIID in the second byte of field PCEID.

This PCE is not device related. Field PCEDCT is zero.

### \$SPIWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
0	(0)	STRUCTURE	0	PCE	, Continue PCE DSECT
312	(138)	BITSTRING	228	SPIWS	WS EBCDIC list
540	(21C)	SIGNED	4	SPIWKOFF	Offset of work JOE
544	(220)	SIGNED	2	SPI#PDDDB	Number of PDDDBs processed in SSI
546	(222)	SIGNED	2	SPIPDDDB#	Number of PDDDBs processed in SASR
548	(224)	BITSTRING	1	SPIFLAG1	Copy of SAPFLAG1
549	(225)	BITSTRING	1	SPIFLAGJ	Copy of SAPFLAGJ
550	(226)	BITSTRING	1	SPIFLAGS	Local (SASR) flags
		1... ....		SPISDISC	"B'10000000" Discard the data sets that have been processed in this JOE
		.1.. ....		SPISDUPJ	"B'01000000" Caller wants to reject job if there is another with same name
		..1. ....		SPISJOE	"B'00100000" JOE handled by SAIDISP
		...1 ....		SPISJLOK	"B'00010000" SAIDISP has job lock
551	(227)	CHARACTER	1	SPIMCLAS	Message class of job
552	(228)	BITSTRING	1	SPIFLAG3	Copy of SAPFLAG3
553	(229)	BITSTRING	11		Reserved for future use

## \$SPIWORK Map

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
564	(234)	SIGNED	4	SPIWRNUM	Thread level value used for JWEL proc
568	(238)	BITSTRING	8	SPIWRASI	Address space level value used for JWEL processing
576	(240)	BITSTRING	3	SPIDEVID	Device ID
579	(243)	BITSTRING	1	SPIFLGJ2	Copy of SAPFLGJ2
580	(244)	SIGNED	4	SPIRECCT	PDDDB record count
584	(248)	SIGNED	4	SPIPGCT	PDDDB page count
588	(24C)	SIGNED	4	SPIOJOE	Offset of prior JOE
592	(250)	SIGNED	4	SPIOCRTM	Create time of JOE
596	(254)	BITSTRING	192	SPIRGRPM	Parameter list for TREGROUP
788	(314)	BITSTRING	180	SPIWKJOA	Temporary JOA
968	(3C8)	SIGNED	4	SPIXECB (0)	SAPID queue mod ENQ ECB
992	(3E0)	SIGNED	4	SPIENQST (0)	True start of ENQ list

Comment

MACRO-DATE = 06/24/03

End of Comment

992	(3E0)	SIGNED	4	(0)	ESTABLISH A FULLWORD BOUNDARY
992	(3E0)	ADDRESS	4		PREFIX - ECB ADDRESS
992	(3E0)	X'3E4'	0	SPIENQPL	*** X02113
996	(3E4)	ADDRESS	1		PELLAST flag byte. X02113
997	(3E5)	ADDRESS	1		PELMILEN - RNAME length.
998	(3E6)	BITSTRING	1		

Comment

PELFLAG - flag byte 2.

End of Comment

999	(3E7)	ADDRESS	1		PELRET - return code byte.
1000	(3E8)	ADDRESS	4		QNAME ADDRESS
1004	(3EC)	ADDRESS	4		RNAME ADDRESS
1004	(3EC)	X'3E0'	0	SPIENQL	"SPIENQST,*-SPIENQST" ENQ parm length, IPCS use
1008	(3F0)	SIGNED	4	SPIDEQST (0)	True start of DEQ list

Comment

MACRO-DATE = 10/06/2004

End of Comment

1008	(3F0)	SIGNED	4	(0)	ESTABLISH A FULLWORD BOUNDARY
1008	(3F0)	X'3F0'	0	SPIDEQPL	*** X02113
1008	(3F0)	ADDRESS	1		PELLAST flag byte. X02113
1009	(3F1)	ADDRESS	1		PELMILEN - RNAME length.
1010	(3F2)	BITSTRING	1		

Comment

PELFLAG - flag byte 2.

End of Comment

1011	(3F3)	ADDRESS	1		PELRET - return code byte.
1012	(3F4)	ADDRESS	4		QNAME ADDRESS
1016	(3F8)	ADDRESS	4		RNAME ADDRESS
1016	(3F8)	X'3F0'	0	SPIDEQL	"SPIDEQST,*-SPIDEQST" DEQ parm length, IPCS use
1020	(3FC)	ADDRESS	4	SPIIOT	Address of IOT buffer
1024	(400)	BITSTRING	4	SPIJBKEY	Job key
1028	(404)	BITSTRING	6	SPIANCHR	MQTR of first regular IOT
1034	(40A)	BITSTRING	50	SPIPRTBL	Room for PRMODE table
1084	(43C)	SIGNED	1	SPIATYPE	Type of SSI call (See SSS2TYPE in IAZSSS2)
1085	(43D)	BITSTRING	1		Reserved for future use
1088	(440)	ADDRESS	4	(0)	Align on full word
1088	(440)	CHARACTER	8	SPIAJOBN	Application jobname

Offsets					
Dec	Hex	Type/Value	Len	Name (Dim)	Description
1096	(448)	CHARACTER	8	SPIAJOB1	Application jobid
1104	(450)	CHARACTER	8	SPIACHKY	Application CSCB CHKEY
1112	(458)	BITSTRING	4	SPIIOTF	MTTR of first IOT in JOE
1116	(45C)	SIGNED	2	SPIPDDBF	PDDDB offset of first PDDDB in JOE
1118	(45E)	CHARACTER	8	SPITHRED	Thread name (from SSS2APPL)
1128	(468)	DBL WORD	8	(0)	Multiple Double words long
1128	(468)	X'330'	0	SPIWKSIZ	**"PCEWORK" LENGTH OF PSO PCE WORK AREA

**\$SPIWORK Cross Reference**

Name	Hex Offset	Hex Value
PCE	0	
SPI#PDDDB	220	
SPIACHKY	450	
SPIAJOB1	448	
SPIAJOB1N	440	
SPIANCHR	404	
SPIDEQL	3F8	3F0
SPIDEQPL	3F0	3F0
SPIDEQST	3F0	
SPIDEV1D	240	
SPIENQL	3EC	3E0
SPIENQPL	3E0	3E4
SPIENQST	3E0	
SPIFLAGJ	225	
SPIFLAGS	226	
SPIFLAG1	224	
SPIFLAG3	228	
SPIFLGJ2	243	
SPIIOT	3FC	
SPIIOTF	458	
SPIJBKEY	400	
SPI1CLAS	227	
SPIOCRTM	250	
SPIOJOE	24C	
SPIPDDB#	222	
SPIPDDBF	45C	
SPIPGCT	248	
SPIPR1BL	40A	
SPIRECCT	244	
SPIRGRPM	254	
SPISDISC	226	80
SPISDUPJ	226	40
SPISJLOK	226	10
SPI1SJOE	226	20
SPI1HRED	45E	
SPI1TYPE	43C	
SPIWKJOA	314	
SPIWKOFF	21C	
SPIWKSIZ	468	330
SPIWRASI	238	
SPIWRNUM	234	
SPIWS	138	
SPIXECB	3C8	

## \$SPIWORK Cross Reference

---

## Notices

This information was developed for products and services offered in the U.S.A. or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel  
IBM Corporation  
2455 South Road  
Poughkeepsie, NY 12601-5400  
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

## **Policy for unsupported hardware**

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

---

## **Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at:

<http://www.ibm.com/legal/us/en/copytrade.shtml>





Program Number: 5650-ZOS

Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

GA32-1000-00

