


```
SSSSSSSS YY YY SSSSSSS CCCCCCCC AAAAAA NN NN CCCCCCCC EEEEEEEEE LL
SSSSSSSS YY YY SSSSSSS CCCCCCCC AAAAAA NN NN CCCCCCCC EEEEEEEEE LL
SS SS YY YY SS SS CC CC AA AA NN NN CC CC EEEEEEEEE LL
SS SS YY YY SS SS CC CC AA AA NN NN CC CC EEEEEEEEE LL
SS SSSSSS YY YY SSSSSS CCCCCCCC AAAAAA NN NN CC CC EEEEEEEEE LL
SS SSSSSS YY YY SSSSSS CCCCCCCC AAAAAA NN NN CC CC EEEEEEEEE LL
SS YY YY SS SS CC CC AA AA NN NN CC CC EEEEEEEEE LL
SS YY YY SS SS CC CC AA AA NN NN CC CC EEEEEEEEE LL
SS YY YY SS SS CC CC AA AA NN NN CC CC EEEEEEEEE LL
SSSSSSSS YY SSSSSSS CCCCCCCC AAAAAA NN NN CCCCCCCC EEEEEEEEE LL
SSSSSSSS YY SSSSSSS CCCCCCCC AAAAAA NN NN CCCCCCCC EEEEEEEEE LL
LL I I I I I I SSSSSSS
LL I I I I I I SSSSSSS
LL I I SS
LL I I SS
LL I I SS
LL I I SSSSSS
LL I I SSSSSS
LL I I SS
LL I I SS
LL I I SS
LL I I SS
LLLLLLLLLLLL I I I I I I SSSSSSS
LLLLLLLLLLLL I I I I I I SSSSSSS
```

(1)	85	CANCEL I/O ON CHANNEL
-----	----	-----------------------

```

0000 1 .TITLE SYSCANCEL - SYSTEM SERVICE CANCEL I/O ON CHANNEL
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5
0000 6 *****
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28 D N. CUTLER 4-AUG-77
0000 29
0000 30 SYSTEM SERVICE CANCEL I/O ON CHANNEL
0000 31
0000 32 MODIFIED BY:
0000 33
0000 34 V03-005 CDS0001 Christian D. Saether 20-July-1984
0000 35 Don't do the acpcontrol function for disk devices.
0000 36
0000 37 V03-004 LMP0251 L. Mark Pilant, 9-May-1984 12:48
0000 38 Correct a bug in WMC0001 so that the correct base register
0000 39 is used to lock down the CCB.
0000 40
0000 41 V03-003 WMC0001 Wayne Cardoza 16-Apr-1984
0000 42 Don't touch CCB above IPL 2.
0000 43
0000 44 V03-002 ROW0136 Ralph O. Weber 25-OCT-1982
0000 45 Change event flag number in hand crafted IOS_ACPCONTROL IRP
0000 46 from #31 to #EXESC_SYSEFN, it symbolic equivalent.
0000 47
0000 48 V03-001 KDM0002 Kathleen D. Morse 28-Jun-1982
0000 49 Added $$$DEF.
0000 50
0000 51 **
0000 52
0000 53 MACRO LIBRARY CALLS
0000 54
0000 55
0000 56 $CADEF ;DEFINE CONDITIONAL ASSEMBLY PARAMETERS
0000 57 $CANDEF ;DEFINE CANCEL REASON CODES

```

SYS
Sym

ACM
EXE
EXE
EXE
EXE
EXE
EXE
IPL
LOG
PCB
PID
PRS
REQ
SSS
TQE
TQE

PSE

: B
\$AB

Pha

Ini
Com
Pas
Sym
Pas
Sym
Pse
Cro
Ass

The
329
The
146
13

```

0000 58 $CCBDEF ;DEFINE CCB OFFSETS
0000 59 $DCDEF ;DEFINE DEVICE CLASS CONSTANTS
0000 60 $DDBDEF ;DEFINE DDB OFFSETS
0000 61 $DDTDEF ;DEFINE DDT OFFSETS
0000 62 $DEVDEF ;DEFINE DEVICE CHARACTERISTIC BITS
0000 63 $DYNDEF ;DEFINE DATA STRUCTURE TYPE CODES
0000 64 $IODEF ;DEFINE I/O FUNCTION CODES
0000 65 $IPLDEF ;DEFINE INTERRUPT PRIORITY LEVELS
0000 66 $IRPDEF ;DEFINE IRP OFFSETS
0000 67 $PCBDEF ;DEFINE PCB OFFSETS
0000 68 $PRDEF ;DEFINE PROCESSOR REGISTERS
0000 69 $RSNDEF ;DEFINE RESOURCE WAIT NUMBERS
0000 70 $SSDEF ;DEFINE SYSTEM STATUS CODES
0000 71 $UCBDEF ;DEFINE UCB OFFSETS
0000 72 $VCBDEF ;DEFINE VCB OFFSETS
0000 73 $WCBDEF ;DEFINE WCB OFFSETS
0000 74
0000 75 :
0000 76 : LOCAL SYMBOLS
0000 77 :
0000 78 : ARGUMENT LIST OFFSET DEFINITIONS
0000 79 :
0000 80
00000000 0000 81 NARG=0 ;NUMBER OF ARGUMENTS PASSED
00000004 0000 82 CHAN=4 ;I/O CHANNEL NUMBER
00000008 0000 83 CODE=8 ;SPECIAL CANCEL CODE

```

```

0000 85      .SBTTL  CANCEL I/O ON CHANNEL
0000 86      :+
0000 87      : EX$CANCEL - CANCEL I/O ON CHANNEL
0000 88      : EX$CANCELN - CANCEL I/O ON CHANNEL WITH REASON CODE
0000 89      :
0000 90      : THIS SERVICE CANCELS ALL I/O ISSUED TO A DEVICE FROM THE SPECIFIED CHANNEL.
0000 91      :
0000 92      : INPUTS:
0000 93      :
0000 94      : CODE(AP) = REASON CODE FOR CANCEL CALL (EX$CANCELN ONLY).
0000 95      : CHAN(AP) = NUMBER OF THE I/O CHANNEL TO CANCEL I/O FOR.
0000 96      : NARG(AP) = NUMBER OF ARGUMENTS PASSED (EX$CANCELN ONLY).
0000 97      :
0000 98      : R4 = CURRENT PROCESS PCB ADDRESS.
0000 99      :
0000 100     : OUTPUTS:
0000 101     :
0000 102     : R0 LOW BIT CLEAR INDICATES FAILURE TO CANCEL I/O.
0000 103     :
0000 104     : S$$_EXQUOTA - DIRECT I/O QUOTA EXCEEDED WHILE TRYING TO
0000 105     : CANCEL FILE I/O.
0000 106     :
0000 107     : S$$_INSFMEM - INSUFFICIENT MEMORY AVAILABLE TO ALLOCATE I/O
0000 108     : PACKET.
0000 109     :
0000 110     : S$$_IVCHAN - INVALID CHANNEL NUMBER SPECIFIED.
0000 111     :
0000 112     : S$$_NOPRIV - SPECIFIED CHANNEL IS NOT ASSIG'ED TO A DEVICE
0000 113     : OR THE CALLER DOES NOT HAVE SUFFICIENT PRIVILEGE TO
0000 114     : ACCESS THE CHANNEL.
0000 115     :
0000 116     : R0 LOW BIT SET INDICATES SUCCESSFUL COMPLETION.
0000 117     :
0000 118     : S$$_NORMAL - NORMAL COMPLETION.
0000 119     : -
0000 120     :
0000 121     : .ENABL  LSB
0000 122     :
0000 123     : .ENTRY  EX$CANCELN,^M<R2,R3,R4,R5,R6,R7,R8>
0002 124     ASSUME  CAN$C_CANCEL EQ 0
0002 125     CLRL   R8                      ;ASSUME NO REASON CODE
0004 126     :
0004 127     : We do not check access to argument list here - since this is a
0004 128     : VMS internal entry point.
0004 129     :
0004 130     : CMPB   S^#2,NARG(AP)              ;REASON CODE GIVEN?
0007 131     BNEQ   S$                       ;IF NEQ - NO REASON CODE
0009 132     MOVL  CODE(AP),R8              ;ELSE, GET REASON CODE
000D 133     BRB   S$                       ;JOIN COMMON CODE
000F 134     :
000F 135     : .ENTRY  EX$CANCEL,^M<R2,R3,R4,R5,R6,R7,R8>
0011 136     ASSUME  CAN$C_CANCEL EQ 0
0011 137     CLRL   R8                      ;NEVER A REASON CODE FOR OLD CANCEL
58 04 AC 3C 0013 138 5$: MOVZWL CHAN(AP),R0      ;GET I/O CHANNEL NUMBER
00000000'EF 16 0017 139     JSB   IOCSVERIFYCHAN ;VERIFY I/O CHANNEL NUMBER
0066 50 E9 001D 140     BLBC  R0,50$          ;IF LBC INVALID CHANNEL
0057 52 D0 0020 141     MOVL  R2,R7          ;SAVE CHANNEL INDEX

```

```

      56 51 D0 0023 142      MOVL R1,R6      :SAVE ADDRESS OF CCB
      55 66 D0 0026 143      MOVL CCB$$_UCB(R6),R5 :GET ASSIGNED DEVICE UCB ADDRESS
      7E   DC 0029 144 10$: MOVPSL -(SP)      :SAVE CURRENT PROCESSOR STATUS
      002B 145      SETIPL #IPL$_ASTDEL :RAISE TO AST DELIVERY LEVEL
      002E 146      ASSUME CCB$$_LENGTH EQ 16 :MAKE SURE CCB IS PAGE ALIGNED
  7E 0A A6 B0 002E 147      MOVW CCB$$_IOC(R6),-(SP) :WE NEED A SPOT FOR IPL - SAFE AT IPL 2
  0A A6 0B A5 90 0032 148      MOVW CCB$$_FIPL(R5),CCB$$_IOC(R6) :THE SETIPL WILL LOCK THE CCB
      0037 149      SETIPL CCB$$_IOC(R6) :RAISE TO DRIVER FORK IPL
      0A A6 8E B0 003B 150      MOVW (SP)+,CCB$$_IOC(R6) :ANY I/O OUTSTANDING? - RESTORE CCB
      49 13 003F 151      BEQL 70$ :IF EQL NO
      53 4C A5 9E 0041 152      MOVAB UCB$$_IOQFL(R5),R3 :GET ADDRESS OF I/O QUEUE LISTHEAD
      52 53 D0 0045 153      MOVL R3,R2 :COPY ADDRESS OF I/O QUEUE LISTHEAD
      52 62 D0 0048 154 20$: MOVL IRP$$_IOQFL(R2),R2 :GET ADDRESS OF NEXT I/O PACKET IN QUEUE
      53 52 D1 004B 155      CMLP R2,R3 :END OF QUEUE?
      3A 13 004E 156      BEQL 70$ :IF EQL YES
  F3 2A A2 04 E0 0050 157      BBS #IRP$$_VIRTUAL,IRP$$_STS(R2),20$ :IF SET, VIRTUAL I/O REQUEST
  60 A4 0C A2 D1 0055 158      CMLP IRP$$_PID(R2),PCB$$_PID(R4) :PROCESS ID MATCH?
      EC 12 005A 159      BNEQ 20$ :IF NEQ NO
      28 A2 57 B1 005C 160      CMPW R7,IRP$$_CHAN(R2) :I/O CHANNEL NUMBER MATCH?
      E6 12 0060 161      BNEQ 20$ :IF NEQ NO
      52 04 A2 D0 0062 162      MOVL IRP$$_IOQBL(R2),R2 :GET BACKWARD LINK OF CURRENT ENTRY
      51 00 B2 0F 0066 163      REMQUE @IRP$$_IOQFL(R2),R1 :REMOVE I/O PACKET FROM QUEUE
  04 2A A1 00 E1 006A 164      BBC #IRP$$_BUFIO,IRP$$_STS(R1),30$ :IF CLR, DIRECT I/O REQUEST
      2A A1 02 AA 006F 165      BICW #IRP$$_FUNC,IRP$$_STS(R1) :CLEAR BUFFERED READ
  38 A1 0830 8F 3C 0073 166 30$: MOVZWL #SS$_CANCEL,IRP$$_MEDIA(R1) :SET COMPLETION STATUS
      0000 DF 61 0E 0079 167      INSQUE IRP$$_IOQFL(R1),@IRP$$_IOCSGL PSBL :INSERT PACKET IN POST PROCESS QUEUE
      C5 11 0081 168      SOFTINT #IPL$_IOPOST :INITIATE SOFTWARE INTERRUPT
      0083 169      BRB 20$ :
      50 01 3C 0083 171 40$: MOVZWL S^#SS$_NORMAL,R0 :SET NORMAL COMPLETION STATUS
      0086 172 50$: SETIPL #0 :ALLOW INTERRUPTS
      04 0089 173      RET :
      008A 174
  50 0088 C5 D0 008A 175 70$: MOVL UCB$$_DDT(R5),R0 :GET ADDRESS OF DDT
  53 58 A5 D0 008F 176      MOVL UCB$$_IRP(R5),R3 :GET CURRENT I/O PACKET ADDRESS
      52 57 D0 0093 177      MOVL R7,R2 :SET CHANNEL INDEX
      0096 178
      0096 179 : WARNING - Some drivers do a RET from the driver CANCEL I/O ROUTINE, in error
      0096 180 : cases.
      0096 181
      0C B0 16 0096 182      JSB @DDT$$_CANCEL(R0) :CALL CANCEL I/O ROUTINE
  01 40 A5 91 0099 183      CMPB UCB$$_DEVCLASS(R5),#DC$_DISK :IS THIS A DISK?
      E4 13 009D 184      BEQL 40$ :ALL DONE IF SO
      50 0A A6 3C 009F 185      MOVZWL CCB$$_IOC(R6),R0 :GET OUTSTANDING I/O COUNT
  50 04 A6 C8 00A3 186      BLSL CCB$$_WIND(R6),R0 :OUTSTANDING I/O OR FILE ACTIVITY?
      DA 13 00A7 187      BEQL 40$ :IF EQL NO
  D5 38 A5 13 E1 00A9 188      BBC #DEV$$_MNT,UCB$$_DEVCHAR(R5),40$ :IF CLR, DEVICE DISMOUNTED
  D0 38 A5 18 E0 00AE 189      BBS #DEV$$_FOR,UCB$$_DEVCHAR(R5),40$ :IF SET, MOUNTED FOREIGN
  50 04 A6 01 C8 00B3 190      BICL3 #1,CCB$$_WIND(R6),R0 :FILE ACCESSED OR PROCESS SECTION?
      C9 14 00B8 191      BGTR 40$ :IF GTR PROCESS SECTION
      05 13 00BA 192      BEQL 80$ :IF EQL NO FILE ACCESSED
  C2 0B A0 02 E0 00BC 193      BBS #WCBS$_NOTFCP,WCBS$_ACCESS(R0),40$ :IF SET, NOT ACP ACCESS
      51 1C 3C 00C1 194 80$: MOVZWL #SS$_EXQUOTA,R1 :SET EXCEEDED QUOTA STATUS
      50 01 3C 00C4 195      MOVZWL #RSNS$_ASTWAIT,R0 :SET AST WAIT RESOURCE WAIT NUMBER
      3A A4 B5 00C7 196      TSTW PCB$$_BIOCNT(R4) :BUFFERED I/O QUOTA EXCEEDED?
      12 13 00CA 197      BEQL 90$ :IF EQL YES
      51 C4 8F 9A 00CC 198      MOVZBL #IRP$$_LENGTH,R1 :SET LENGTH OF I/O PACKET

```

		FF2D'	30	00D0	199	BSBW	EXESALONONPAGED	:ALLOCATE I/O PACKET
		1A 50	EB	00D3	200	BLBS	R0,100\$:IF LBS SUCCESSFUL ALLOCATION
51	0124	BF	3C	00D6	201	MOVZWL	#SS\$ INSMEM,R1	:SET INSUFFICIENT MEMORY STATUS
	50	03	3C	00DB	202	MOVZWL	#RSNS NPDYNMEM,R0	:SET NONPAGED DYNAMIC MEMORY RESOURCE NUMBER
06	24	A4	EO	00DE	203	BBS	#PCB\$V SSRWAIT,PCB\$S_STS(R4),95\$:IF SET, NO WAIT
		FF1A'	30	00E3	204	BSBW	SCH\$RWAIT	:WAIT FOR RESOURCE
		FF40	31	00E6	205	BRW	10\$:TRY AGAIN
	50	51	DO	00E9	206	SETIPL	#0	:ALLOW INTERRUPTS
			04	00EF	207	MOVL	R1,R0	:SET COMPLETION STATUS
	3A	A4	B7	00F0	208	RET		
	0A	A6	B6	00F3	209	DECW	PCB\$W_BIOCNT(R4)	:UPDATE BUFFERED I/O COUNT
	53	52	DO	00F6	210	INCW	CCB\$W_IOC(R6)	:INCREMENT I/O COUNT
	52	08	CO	00F9	211	MOVL	R2,R3	:COPY ADDRESS OF I/O PACKET
82	C4	8F	9B	00FC	212	ADDL	#IRP\$W_SIZE,R2	:CALCULATE ADDRESS OF PACKET SIZE
	82	0A	9B	0100	213	MOVZBW	#IRP\$C_LENGTH,(R2)+	:INSERT SIZE OF I/O REQUEST PACKET
82	60	A4	DO	0103	214	MOVZBW	#DYN\$C_IRP,(R2)+	:INSERT DATA STRUCTURE TYPE AND ZERO MODE
		82	7C	0107	215	MOVL	PCB\$S_PID(R4),(R2)+	:INSERT CURRENT PROCESS ID
82	04	A6	DO	0109	216	CLRQ	(R2)+	:CLEAR AST ADDRESS AND PARAMETER
	82	55	DO	010D	217	MOVL	CCB\$S_WIND(R6),(R2)+	:INSERT ADDRESS OF WINDOW
	82	38	BO	0110	218	MOVL	R5,(R2)+	:INSERT DEVICE UCB ADDRESS
82	00	8F	90	0113	219	MOVW	#IOS_ACPCONTROL,(R2)+	:INSERT I/O FUNCTION CODE
82	2F	A4	90	0117	220	MOVW	#EXESC_SYSEFN,(R2)+	:SET EVENT FLAG NUMBER
		82	D4	011B	221	MOVW	PCB\$B_PRI(R4),(R2)+	:INSERT PROCESS BASE PRIORITY
	82	57	BO	011D	222	CLRL	(R2)+	:CLEAR I/O STATUS BLOCK ADDRESS
	82	01	BO	0120	223	MOVW	R7,(R2)+	:INSERT CHANNEL NUMBER
		82	7C	0123	224	MOVW	#IRP\$M_BUFIO,(R2)+	:INSERT PACKET STATUS
58	A3	008C	C4	DO	225	CLRQ	(R2)+	:CLEAR BUFFER PARAMETERS
	50	34	A5	DO	226	MOVL	PCB\$S_ARB(R4),IRP\$S_ARB(R3)	:GET ACCESS RIGHTS BLOCK ADDRESS
		OC	A0	DO	227	MOVL	UCB\$S_VCB(R5) R0	:GET ADDRESS OF VCB
			B6	012F	228	INCW	VCB\$W_TRANS(R0)	:UPDATE TRANSACTION COUNT
				0132	229			
				0132	230	.IF DF	CAS_MEASURE_IOT	
				0132	231			
	52	53	DO	0132	232	MOVL	R3,R2	:SET ADDRESS OF I/O REQUEST PACKET
		FEC8'	30	0135	233	BSBW	PMS\$START_RQ	:INSERT START OF I/O REQUEST MESSAGE
				0138	234			
				0138	235	.ENDC		
				0138	236			
		FEC5'	31	0138	237	BRW	EXESQIOACPKT	:QUEUE ACP PACKET
				0138	238			
				0138	239	.DSABL	LSB	
				0138	240			
				0138	241	.END		

SYSCANCEL
Symbol table

- SYSTEM SERVICE CANCEL I/O ON CHANNEL M 3

16-SEP-1984 01:44:48 VAX/VMS Macro V04-00
5-SEP-1984 03:49:13 [SYS.SRC]SYSCANCEL.MAR;1

Page 6
(1)

CANSC_CANCEL	= 00000000		
CCBSK_LENGTH	= 00000010		
CCBSL_UCB	= 00000000		
CCBSL_WIND	= 00000004		
CCBSW_IOC	= 0000000A		
CHAN	= 00000004		
CODE	= 00000008		
DCS_DISK	= 00000001		
DDTSL_CANCEL	= 0000000C		
DEVSU_FOR	= 00000018		
DEVSU_MNT	= 00000013		
DYNSC_IRP	= 0000000A		
EXESACONONPAGED	*****	X	01
EXESCANCEL	0000000F	RG	01
EXESCANCELN	00000000	RG	01
EXESC_SYSEFN	*****	X	01
EXESQIOACPPKT	*****	X	01
IOS_ACPCONTROL	= 00000038		
IOCSGL_PSBL	*****	X	01
IOCSVERIFYCHAN	*****	X	01
IPLS_ASTDEL	= 00000002		
IPLS_IOPOST	= 00000004		
IRPSC_LENGTH	= 000000C4		
IRPSL_ARB	= 00000058		
IRPSL_IOQBL	= 00000004		
IRPSL_IOQFL	= 00000000		
IRPSL_MEDIA	= 00000038		
IRPSL_PID	= 0000000C		
IRPSM_BUFIO	= 00000001		
IRPSM_FUNC	= 00000002		
IRPSV_BUFIO	= 00000000		
IRPSV_VIRTUAL	= 00000004		
IRPSW_CHAN	= 00000028		
IRPSW_SIZE	= 00000008		
IRPSW_STS	= 0000002A		
NARG	= 00000000		
PCBSB_PRIB	= 0000002F		
PCBSL_ARB	= 0000008C		
PCBSL_PID	= 00000060		
PCBSL_STS	= 00000024		
PCBSV_SSRWAIT	= 0000000A		
PCBSW_BIOCNT	= 0000003A		
PMS\$START_RQ	*****	X	01
PRS_IPL	= 00000012		
PRS_SIRR	= 00000014		
RSNS_ASTWAIT	= 00000001		
RSNS_NPDYMEM	= 00000003		
SCH\$RWAIT	*****	X	01
SS\$CANCEL	= 00000830		
SS\$EXQUOTA	= 0000001C		
SS\$INSFMEM	= 00000124		
SS\$NORMAL	= 00000001		
UCBSB_DEVCLASS	= 00000040		
UCBSB_FIPL	= 00000008		
UCBSL_DDT	= 00000088		
UCBSL_DEVCHAR	= 00000038		
UCBSL_IOQFL	= 0000004C		

UCBSL_IRP	= 00000058
UCBSL_VCB	= 00000034
VCBSW_TRANS	= 0000000C
WCBSB_ACCESS	= 00000008
WCBSV_NOTFCP	= 00000002

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS :	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
. BLANK :	0000013B (315.)	01 (1.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
\$ABSS	00000000 (0.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.03	00:00:02.02
Command processing	111	00:00:00.53	00:00:05.19
Pass 1	489	00:00:19.40	00:01:01.39
Symbol table sort	0	00:00:03.38	00:00:10.12
Pass 2	66	00:00:02.95	00:00:10.62
Symbol table output	8	00:00:00.10	00:00:00.10
Psect synopsis output	2	00:00:00.03	00:00:00.26
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	708	00:00:26.43	00:01:30.00

The working set limit was 1500 pages.
110437 bytes (216 pages) of virtual memory were used to buffer the intermediate code.
There were 120 pages of symbol table space allocated to hold 2259 non-local and 11 local symbols.
241 source lines were read in Pass 1, producing 18 object records in Pass 2.
78 pages of virtual memory were used to define 27 macros.

! Macro library statistics !

Macro library name	Macros defined
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	15
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	9
TOTALS (all libraries)	24

2393 GETS were required to define 24 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SYSCANCEL/OBJ=OBJ\$:SYSCANCEL MSRC\$:SYSCANCEL/UPDATE=(ENH\$:SYSCANCEL)+EXECMLS/LIB

