

```
SSSSSSSSSSSSSS  YYY  YYY  SSSSSSSSSSSSS
SSSSSSSSSSSSSS  YYY  YYY  SSSSSSSSSSSSS
SSSSSSSSSSSSSS  YYY  YYY  SSSSSSSSSSSSS
SSS             YYY  YYY  SSS
SSS             YYY  YYY  SSS
SSS             YYY  YYY  SSS
SSS             YYY  YYY  SSS
SSS             YYY  YYY  SSS
SSS             YYY  YYY  SSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
SSS             SSS
SSS             SSS
SSS             SSS
SSS             SSS
SSS             SSS
SSS             SSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
SSSSSSSSSSSS    YYY  YYY  SSSSSSSSSSS
```

_S

Ps

YZ

ZS

ZS

ZS

ZS

ZS

ZS

SSS

SSS

SSS

SSS

ZS

ZS

ZS

ZS

ZS

ZS

```

SSSSSSSS YY YY SSSSSSS DDDDDDD EEEEEEEEE LL P P P P P P R R R R R R C C C C C C
SSSSSSSS YY YY SSSSSSS DDDDDDD EEEEEEEEE LL P P P P P P R R R R R R C C C C C C
SS SS YY YY SS SSSSSSS DD DD EE LL PP PP RR RR CC
SS SS YY YY SS SSSSSSS DD DD EE LL PP PP RR RR CC
SS SS YY YY SS SSSSSSS DD DD EE LL PP PP RR RR CC
SSSSSS YY YY SSSSSSS DD DD EE LL PP PP RR RR CC
SSSSSS YY YY SSSSSSS DD DD EE LL PP PP RR RR CC
SS SS YY YY SS SSSSSSS DD DD EE LL PP PP RR RR CC
SS SS YY YY SS SSSSSSS DD DD EE LL PP PP RR RR CC
SSSSSS YY YY SSSSSSS DDDDDDD EEEEEEEEE LLLLLLLLLL PP PP RR RR C C C C C C
SSSSSS YY YY SSSSSSS DDDDDDD EEEEEEEEE LLLLLLLLLL PP PP RR RR C C C C C C

```

```

LL LL I I I I I I SSSSSSSS
LL LL I I I I I I SSSSSSSS
LL LL I I I I I I SS
LL LL I I I I I I SS
LL LL I I I I I I SS
LL LL I I I I I I SSSSSS
LL LL I I I I I I SSSSSS
LL LL I I I I I I SS
LL LL I I I I I I SS
LL LL I I I I I I SS
LLLLLLLLLLLL I I I I I I SSSSSSSS
LLLLLLLLLLLL I I I I I I SSSSSSSS

```

(1)	120	DECLARATIONS
(1)	172	EX\$DELPRC - DELETE PROCESS SYSTEM SERVICE
(1)	272	DELETE - PERFORM DELETE ACTIONS IN CONTEXT OF PROCESS
(1)	663	DELPAGE - DELETE PAGE
(1)	699	TERMMBX - SEND MESSAGE TO TERMINATION MAILBOX
(1)	789	Return Unused CPU Time Limit to Parent

```

0000 1 .TITLE SYSDELPRC - DELETE PROCESS SYSTEM SERVICE
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *
0000 26 *****
0000 27
0000 28 ++
0000 29 FACILITY: EXECUTIVE, SYSTEM SERVICES
0000 30
0000 31 ABSTRACT: SYSDELPRC IMPLEMENTS THE DELETE PROCESS SYSTEM SERVICE
0000 32 WHICH CAUSES A PROCESS TO BE DELETED FROM THE SYSTEM AFTER
0000 33 RELEASING ALL OF ITS RESOURCES.
0000 34
0000 35 ENVIRONMENT:
0000 36 MODE=KERNEL
0000 37
0000 38 AUTHOR: R. I. HUSTVEDT, CREATION DATE: 30-DEC-76
0000 39
0000 40 MODIFIED BY:
0000 41
0000 42 V03-019 TMK0002 Todd M. Katz 18-Apr-1983
0000 43 Delete the job-wide logical name table and all names contained
0000 44 therein after dismounting all mounted volumes ( the table is
0000 45 actually deleted after decrementing the interactive and batch
0000 46 job counts as appropriate ). We were deleting the job table
0000 47 before dismounting all privately mounted volumes. This was
0000 48 causing random paged-pool corruption whenever a logical name had
0000 49 been specified during the mounting of the private volume. This
0000 50 logical name is now placed into the job table, instead of into
0000 51 the process logical name table. Therefore, it was being deleted
0000 52 twice - once as a result of the cleanup of the job table prior
0000 53 to its deletion, and once as a result of the dismount of the
0000 54 privately mounted disk.
0000 55
0000 56 V03-018 HH0011 Hai Huang 4-Apr-1984
0000 57 Lock I/O database when flushing the job-wide mount list.

```

0000	58	:				
0000	59	:	V03-017	HM0002	Hai Huang	1-Feb-1984
0000	60	:			Add job-wide mount support, i.e. do not issue ioc\$dismount	
0000	61	:			if the process is a sub-process.	
0000	62	:				
0000	63	:	V03-016	CDS0002	Christian D. Saether	14-Dec-1983
0000	64	:			Modify the bugcheck that looks for assigned channels	
0000	65	:			to ignore the special xqp channel.	
0000	66	:				
0000	67	:	V03-015	LJK0254	Lawrence J. Kenah	7-Dec-1983
0000	68	:			Add two consistency checks.	
0000	69	:				
0000	70	:	V03-014	TMK0001	Todd M. Katz	12-Oct-1983
0000	71	:			If the process being deleted is not a sub-process, then	
0000	72	:			delete the job-wide logical name table after the deletion	
0000	73	:			of all subprocesses has completed.	
0000	74	:				
0000	75	:	V03-013	LJK0248	Lawrence J. Kenah	30-Aug-1983
0000	76	:			Reorder EXE\$DELPRC procedure so that setting of DELPEN bit	
0000	77	:			and queuing of DELETE AST occur while remaining at SYNCH.	
0000	78	:				
0000	79	:	V03-012	WMC0002	Wayne Cardoza	15-JUL-1983
0000	80	:			Batch count.	
0000	81	:				
0000	82	:	V03-011	WMC0011	Wayne Cardoza	15-Jun-1983
0000	83	:			Decrement interactive user ccunt when appropriate.	
0000	84	:				
0000	85	:	V03-010	RSH0032	R. Scott Hanna	26-May-1983
0000	86	:			Fix linker truncation error.	
0000	87	:				
0000	88	:	V03-009	ACG0335	Andrew C. Goldstein,	10-May-1983 19:20
0000	89	:			Deallocate extended rights list when present	
0000	90	:				
0000	91	:	V03-008	LJK0202	Lawrence J. Kenah	19-Apr-1983
0000	92	:			Fix several broken word displacements	
0000	93	:				
0000	94	:	V03-007	CWH1002	CW Hobbs	24-Feb-1983
0000	95	:			Modified to support extended pids. Use SCH\$C_SWPPIX to locate	
0000	96	:			the swapper pid.	
0000	97	:				
0000	98	:	V03-006	DMW4025	DMWalp	7-Jan-1983
0000	99	:			Fixed broken branch	
0000	100	:				
0000	101	:	V03-005	CDS0001	C Saether	22-Oct-1982
0000	102	:			Add process deletion defer mechanism. Initial user F11BXQP.	
0000	103	:				
0000	104	:	V03-004	ACG0301	Andrew C. Goldstein,	21-Oct-1982 13:41
0000	105	:			Restore FILCNTNONZ bugcheck	
0000	106	:				
0000	107	:	V03-003	WMC0001	Wayne Cardoza	26-Aug-1982
0000	108	:			Reset privileged library vectors after rundown.	
0000	109	:				
0000	110	:	V03-002	LJK0169	LAWRENCE J. KENAH	2-Jun-1982
0000	111	:			Correct incorrect register bug when returning unused CPU	
0000	112	:			time. Make extremely rare error path perform in a way	
0000	113	:			that makes sense.	
0000	114	:				

SYSDELPRC
V04-000

- DELETE PROCESS SYSTEM SERVICE

D 9

16-SEP-1984 01:57:52 VAX/VMS Macro V04-00
5-SEP-1984 03:52:26 [SYS.SRC]SYSDELPRC.MAR;1

Page 3
(1)

0000 115 :
0000 116 :
0000 117 :
0000 118 :--

V03-001 ACG0178 Andrew C. Goldstein, 1-Apr-1982 11:21
Remove FILECNTNONZ bugcheck

```

0000 120      .SBTTL  DECLARATIONS
0000 121      :
0000 122      : INCLUDE FILES:
0000 123      :
0000 124      :
0000 125      $ACBDEF      : DEFINE  AST CONTROL BLOCK OFFSETS
0000 126      $ACCDEF      : DEFINE  TERMINATION MESSAGE OFFSETS
0000 127      $ACMDEF      : DEFINE  ACCOUNTING MESSAGE OFFSETS
0000 128      $ARBDEF      : DEFINE  ACCESS RIGHTS BLOCK
0000 129      $CCBDEF      : DEFINE  CHANNEL CONTROL BLOCK
0000 130      $IODEF       : DEFINE  I/O FUNCTION CODES
0000 131      $IPLDEF      : DEFINE  INTERRUPT PRIORITY LEVELS
0000 132      $JIBDEF      : DEFINE  JOB INFORMATION BLOCK OFFSETS
0000 133      $MSGDEF      : DEFINE  SYSTEM MESSAGE TYPE CODES
0000 134      $PCBDEF      : DEFINE  PCB OFFSETS
0000 135      $PFNDEF      : DEFINE  PFN CONSTANTS
0000 136      $PHDDEF      : DEFINE  PROCESS HEADER OFFSETS
0000 137      $PRDEF       : DEFINE  PROCESSOR REGISTER NAMES
0000 138      $PRIDEF      : DEFINE  PRIORITY INCREMENTS
0000 139      $PTEDEF      : DEFINE  PTE FIELDS
0000 140      $PSLDEF      : DEFINE  PSL FIELDS AND VALUES
0000 141      $RSNDEF      : DEFINE  RESOURCE NUMBERS
0000 142      $SECDEF      : DEFINE  SECTION TABLE ENTRY
0000 143      $SSDEF       : DEFINE  SYSTEM SERVICE STATUS CODES
0000 144      $UCBDEF      : DEFINE  UNIT CONTROL BLOCK OFFSETS
0000 145      :
0000 146      :
0000 147      : MACROS:
0000 148      :
0000 149      :
0000 150      :
0000 151      : EQUATED SYMBOLS:
0000 152      :
0000 153      :
0000001C 0000 154 ACB_L_CPULIM = ACB$L_KAST + 4
00000020 0000 155 ACB_L_CPUTIM = ACB_L_CPULIM + 4
00000010 0000 156 JOB_STRING_LEN = 16
0000 157      :
0000 158      :
0000 159      : OWN STORAGE:
0000 160      :
0000 161      :
00000000 162      .PSECT  YSEXEPAGED
0000 163      :
0000 164 LNM_SYSTEM_DIRECTORY_DESC: ; Descriptor of System Directory Table
59 53 24 4D 4E 4C 00000008'010E0000' 0000 165      .ASCID  /LNMS$SYSTEM_DIRECTORY/
4F 54 43 45 52 49 44 5F 4D 45 54 53 000E
59 52 001A
001C 166
001C 167 JOB_FAO: ; $FAO control string descriptor for
4F 4A 24 4D 4E 4C 00000024'010E0000' 001C 168      .ASCID  /LNMS$JOB_!XL/
4C 58 21 5F 42 002A ; construction of job-wide table name
002F 169
00000000 170      .PSECT  AEXENONPAGED

```

```

0000 172      .SBTTL EXE$DELPRC - DELETE PROCESS SYSTEM SERVICE
0000 173      :++
0000 174      : FUNCTIONAL DESCRIPTION:
0000 175      : EXE$DELPRC IMPLEMENTS THE DELETE PROCESS SYSTEM SERVICE WHICH CAUSES
0000 176      : A PROCESS TO BE DELETED FROM THE SYSTEM. IF THE SPECIFIED
0000 177      : PROCESS EXISTS, A KERNEL AST IS QUEUED FOR IT TO
0000 178      : PERFORM THE DELETION PROCESSING IN THE CONTEXT OF THE TARGET
0000 179      : PROCESS. A PROCESS THEREFORE DELETES ITSELF.
0000 180      :
0000 181      : CALLING SEQUENCE:
0000 182      : CALLG  ARGLIST,EXE$DELPRC
0000 183      :
0000 184      : INPUT PARAMETERS:
0000 185      : PIDADR(AP) - ADDRESS OF THE PROCESS IDENTIFICATION OF THE
0000 186      : PROCESS TO BE DELETED.
0000 187      : PRCNAM(AP) - ADDRESS OF STRING DESCRIPTOR FOR PROCESS LOGICAL NAME
0000 188      : STRING.
0000 189      : R4 - PCB ADDRESS OF CURRENT PROCESS
0000 190      :
0000 191      : IMPLICIT INPUTS:
0000 192      : CURRENT PROCESS PCB AND PHD
0000 193      : PCB OF TARGET PROCESS
0000 194      :
0000 195      : OUTPUT PARAMETERS:
0000 196      : R0 - COMPLETION STATUS CODE
0000 197      : @PIDADR(AP) - PROCESS IDENTIFICATION OF DELETED PROCESS
0000 198      :
0000 199      : IMPLICIT OUTPUTS:
0000 200      : NONE
0000 201      :
0000 202      : COMPLETION CODES:
0000 203      : SSS_NORMAL - SUCCESSFUL COMPLETION
0000 204      : SSS_NONEXPR - NONEXISTENT PROCESS
0000 205      : SSS_NOPRIV - NO PRIVILEGE, TARGET PROCESS IS NOT A SUBPROCESS
0000 206      : AND ISSUING PROCESS DOES NOT HAVE PROCESS CONTROL
0000 207      : PRIVILEGE FOR GROUP OR WORLD.
0000 208      : SSS_INSMEM - INSUFFICIENT DYNAMIC MEMORY AVAILABLE FOR SERVICE
0000 209      : AND RESOURCE WAIT MODE DISABLED.
0000 210      :
0000 211      : SIDE EFFECTS:
0000 212      : A KERNEL AST WILL BE ENQUEUED FOR THE PROCESS TO BE
0000 213      : DELETED WHICH MAY CAUSE RESCHEDULING TO OCCUR.
0000 214      :
0000 215      :--
0000 216      :
0000 217      : .ENTRY EXE$DELPRC,^M<R2,R3,R4,R5,R6,R7> ; Save R2 through R7
0000 218      : BSBW EXE$ALLOCIRP ; Allocate AST control block
0000 219      : BLBC R0,10$ ; Return error if none
0000 220      : MOVL R2,R5 ; Set address for SCH$QAST
0000 221      : BSBW EXE$NAMPID ; Convert name/PID to PCB address
0000 222      : BLBC R0,30$ ; Exit if error
0000 223      : CMPW R1,S^#SCH$C_SWPPIX ; Do not allow SWAPPER or NULL
0000 224      : BLEQU 20$ ; Exit with error if either of those
0000 225      :
0000 226      : If the target process is already marked for delete, the system service
0000 227      : simply returns a success code to the caller after deallocating the ACB.
0000 228      : Note that SSS_NORMAL was loaded into R0 by the success path of EXE$NAMPID.

```

```

00FC 0000
FFFB' 30 0002
34 50 E9 0005
55 52 D0 0008
FFF2' 30 000B
34 50 E9 000E
00' 51 B1 0011
2A 18 0014
0016 225
0016 226
0016 227
0016 228

```



```

0016 229
2A 24 A4 01 E2 0016 230          BBSS #PCBSV_DELPEN,PCBSL_STS(R4),30$
001B 231
001B 232 ; The following three instructions are the guts of the $RESUME system service.
001B 233 ; That same functionality must be reproduced here while remaining at IPL
001B 234 ; SYNCH so the system service cannot be used.
001B 235
001B 236          ASSUME PCBSV_RESPEN LE 7
001B 237
24 A4 20 88 001B 238          BISB #<1@PCBSV_RESPEN>,PCBSL_STS(R4) ; Set resume pending bit
52 02 9A 001F 239          MOVZBL #PRIS_RESAVL,R2 ; Set priority increment class
0022 240          RPTEVT RESUME ; Report RESUME event to scheduler
0026 241
10 A5 0000002F'EF 9E 0026 242          MOVAB L^DELETE,ACBSL_AST(R5) ; Set address for delete action
0B A5 94 002E 243          CLRB ACBSB_RMOD(R5) ; Mark as normal kernel mode AST
0C A5 60 A4 D0 0031 244          MOVL PCBSL_PID(R4),ACBSL_PID(R5) ; Set PID of target
0036 245
0036 246 ; R2 has already been loaded with the correct priority increment class
0036 247 ; and R2 is preserved by SCH$RSE (called from within the RPTEVT macro)
0036 248 ; so that a
0036 249 :
0036 250 :          MOVZBL #PRIS_RESAVL,R2 ; Set priority increment
0036 251 :
0036 252 ; is not necessary here.
0036 253
50 FFC7' 30 0036 254          BSBW SCH$QAST ; And queue ast for process
01 3C 0039 255          MOVZWL #SS$_NORMAL,R0 ; Set normal status
003C 256 10$: SETIPL #0 ; Enable rescheduling
003F 257          RET ; ... and return
0040 258
0040 259 ; The following error paths must deallocate the unused ACB before returning.
0040 260
50 08E8 8F 3C 0040 261 20$: MOVZWL #SS$_NONEXPR,k0 ; NULL and SWAPPER cannot be deleted
50 50 DD 0045 262 30$: PUSHL R0 ; Save error status
50 55 DO 0047 263          MOVL R5,R0 ; Get address of ACB
FFB3' 30 004A 264          BSBW EXESDEANONPAGED ; Deallocate packet back to pool
50 8ED0 004D 265          POPL R0 ; Retrieve saved error status
EA 11 0050 266          BRB 10$ ; Join common exit code
0052 267
0052 268 ; The rest of the code in this module can exist in a pageable part of the exec
0052 269
0000002F 270          .PSECT Y$EXEPAGED
    
```

```

002F 272 .SBTTL DELETE - PERFORM DELETE ACTIONS IN CONTEXT OF PROCESS
002F 273 :++
002F 274 : FUNCTIONAL DESCRIPTION: DELETE EXECUTES AS THE RESULT OF A KERNEL
002F 275 : AST INITIATED BY THE DELETE PROCESS SYSTEM SERVICE. IT PERFORMS
002F 276 : THE ACTIONS NECESSARY TO DELETE A PROCESS AND RETURN ITS RESOURCES.
002F 277 : TERMINATION MESSAGES ARE SENT TO THE TERMINATION MAILBOX UNIT
002F 278 : SPECIFIED WHEN THE PROCESS WAS CREATED AND TO THE SYMBIONT
002F 279 : MANAGER IF ACCOUNTING IS NOT DISABLED FOR THIS PROCESS.
002F 280
002F 281 : CALLING SEQUENCE:
002F 282 : (SAME EFFECT AS) DCLAST ASTADR=DELETE MODE=KERNEL
002F 283
002F 284 : INPUT PARAMETERS:
002F 285 : NONE
002F 286
002F 287 : OUTPUT PARAMETERS:
002F 288 : NONE
002F 289
002F 290 : IMPLICIT INPUTS:
002F 291 : PCB OF CURRENT PROCESS (LOCATED VIA SCH$GL_CURPCB)
002F 292 : PHD OF CURRENT PROCESS
002F 293
002F 294 : IMPLICIT OUTPUTS:
002F 295 : NONE
002F 296
002F 297 : SIDE EFFECTS:
002F 298 : NONE
002F 299 :--
002F 300
002F 301 DELETE: : PERFORM DELETE OPERATIONS
002F 302 : ENTRY MASK
54 00000000'GF 0000 0031 303 : MOVL G^SCH$GL CURPCB, R4 : GET PCB ADDRESS
OC A4 01 8A 0038 304 : BICB2 #1, PCB$B ASTACT(R4) : CLEAR KERNEL AST ACTIVE
00000000'EF 16 003C 305 : JSB L^SCH$NEWLVL : COMPUTE NEW AST LEVEL
2A A4 95 0042 306 4$: SETIPL 8$ : RAISE TO SYNCH FOR DPC CHECK
OD 13 0049 307 : TSTB PCB$B_DPC(R4) : IS DPC NON-ZERO?
00 00 DD 004C 308 : BEQL 7$ : EQL THEN DELETION IS ALLOWED
50 01 D0 004E 309 : PUSHL #0 : KERNEL MODE, IPL 0 PSL FOR RWAIT
00000000'EF 16 0050 310 : MOVL #RSNS$ASTWAIT, R0 : NOTE AST RESOURCE
E7 11 0053 311 : JSB L^SCH$RWAIT : WAIT FOR AST
06 24 A4 0A E5 0059 312 : BRB 4$ : MAKE THE TEST AGAIN
04 11 005B 313 7$: SETIPL #0 : AND DROP IPL
00000008 005E 314 : BBCC #PCB$V_SSRWAIT, PCB$L_STS(R4), 10$ : ENABLE RESOURCE WAIT
0069 315 : BRB 10$
0069 316 8$: .LONG IPL$ SYNCH : MAKE SURE CODE DOESN'T PAGE AT SYNCH
0069 317 : ASSUME -.4$ LE 512 : IF NOT TRUE, IPL PAGE FAULT HACK NG
0069 318 10$:
0069 319 :-----
0069 320 :
0069 321 : INVOKE THE USER RUNDOWN SERVICE(S)
0069 322 :
0069 323 : The user service is invoked in KERNEL mode with a JSB and must
0069 324 : exit with a RSB.
0069 325 :
0069 326 : R4 - Pointer to current PCB
0069 327 : R7 - Access mode of call to rundown routine (always 0)
0069 328 : ACMODE(AP) - always 0

```

```

0069 329 :
0069 330 : The above parameters are the same as in the call from SYSRUNDWN.
0069 331 :
0069 332 :-----
0069 333 :
57 D4 0069 334 CLRL R7 ; KERNEL MODE RUNDOWN
5C DD 006B 335 PUSHL AP
7E D4 006D 336 CLRL -(SP) ; ANOTHER KERNEL MODE ACCESS PARAMETER
01 DD 006F 337 PUSHL #1
5C 5E D0 0071 338 MOVL SP,AP
55 00000000'9F D0 0074 339 MOVL @#CTL$GL_USRUNDWN, R5 ; GET PER-PROCESS USER RUNDOWN VECTOR
02 13 007B 340 BEQL 15$ ; NOT PRESENT, SKIP ON
65 16 007D 341 JSB (R5) ; CALL THRU THE VECTOR(S)
55 00000000'GF D0 007F 342 15$: MOVL G^EXE$GL_USRUNDWN, R5 ; GET SYSTEM-WIDE USER RUNDOWN VECTOR
02 13 0086 343 BEQL 18$ ; NOT PRESENT, SKIP ON
65 16 0088 344 JSB (R5) ; CALL THRU THE VECTOR(S)
5E 08 C0 008A 345 18$: ADDL #8,SP ; CLEAN UP
5C 8E D0 008D 346 MOVL (SP)+,AP
00000000'EF 16 0090 347 JSB L^EXE$RESETVEC ; RESET THE PRIVILEGED LIBRARY VECTORS
0096 348 :-----
0096 349 :
0096 350 :
0096 351 : RMS RUNDOWN
0096 352 :-----
0096 353 :-----
5E FF78 CE 9E 0096 355 MOVAB -136(SP),SP ; ALLOCATE STRING BUFFER
6E 80 8F 9A 009B 356 20$: MOVZBL #128,(SP) ; SET COUNT FOR STRING BUFFER
04 AE 08 AE DE 009F 357 MOVAL 8(SP),4(SP) ; FORM ADDRESS PART OF DESCRIPTOR
02 DD 00A4 358 PUSHL #2 ; LAST CHANCE RMS CLEANUP
00000000'9F 04 AE 9F 00A6 359 PUSHAB 4(SP) ; SET ADDRESS OF STRING BUFFER
E8 50 E9 00A9 360 CALLS #2,@#SYS$RMSRUNDWN ; DEFAULT CLOSE OF OPEN RMS FILES
00B3 361 BLBC R0,20$ ; REPEAT IF MORE TO CLOSE
00B3 362 :-----
00B3 363 :
00B3 364 :
00B3 365 : FIND AND DELETE ALL SUB-PROCESSES
00B3 366 :-----
00B3 367 :-----
00B3 368 :
42 A4 B5 00B3 369 SUBDELETE: ; DELETE SUB-PROCESSES
56 00000000'EF 42 13 00B6 370 TSTW PCB$W_PRCNT(R4) ; ANY SUB-PROCESSES?
53 00000000'FF46 3C 00B8 371 BEQL 50$ ; NO, CONTINUE
1C A3 60 A4 D1 00BF 372 MOVZWL SCH$GL_MAXPIX,R6 ; INITIALIZE INDEX FOR PROCESS SCAN
0C 12 00C7 373 10$: MOVL @L^SCH$GL_PCBVEC[R6],R3 ; GET A PCB ADDRESS
E2 56 F5 00DA 374 CMPL PCB$L_PID(R4),PCB$L_OWNER(R3) ; DO WE OWN IT
7E DC 00DD 375 BNEQ 20$ ; NO, TRY ANOTHER
00CE 376 $DELPRC_S PCB$L_EPID(R3) ; MARK IT FOR DELETE USING THE EXTENDED PID
00DA 377 20$: SOBGTR R6,10$ ; CONTINUE
00DD 378 30$: MOVPSL -(SP) ; SAVE PSL
00DF 379 DSBINT W^SUBWIPL,R6 ; BLOCK SYSTEM EVENTS, SAVE IPL IN R6
42 A4 B5 00E7 380 TSTW PCB$W_PRCNT(R4) ; CHECK COUNT OF SUB-PROCESSES STILL RUNNING
08 13 00EA 381 BEQL 40$ ; DONE
50 01 3C 00EC 382 MOVZWL #RSNS$ASTWAIT,R0 ; SET RESOURCE NUMBER
00000000'EF 16 00EF 383 JSB L^SCH$RWAIT ; AND WAIT FOR AN AST
E6 11 00F5 384 BRB 30$ ; CHECK AGAIN WHEN AWAKENED
00F7 385 40$: SETIPL R6 ; RESTORE IPL

```

S
P

P
S
Y
A

P
I
C
P
S
P
C
A
T
1
T
8
4

M
-
T
2
T
M

```

04 11 00FA 386 50$: : DONE, ALL SUB-PROCESSES DELETED
00000008 00FA 387 : CONTINUE WITH RUNDOWN
00FC 388 SUBWIPL: :
00FC 389 : .LONG IPL$_SYNCH : MARKER FOR END OF TEMP NON-PAGED
0100 390 :
0100 391 :-----
0100 392 :
0100 393 : RUNDOWN PROCESS ACTIVITY
0100 394 :
0100 395 :-----
0100 396 :
0100 397 RUNDWN: : RUNDOWN PROCESS ACTIVITY
00 DD 0100 398 : SET MODE FOR RUNDOWN
00000000'EF 01 FB 0102 399 : RUNDOWN AT KERNEL MODE
55 00000000'9F 01 DO 0109 400 : GET PROCESS HEADER ADDRESS
52 55 20 A5 C1 0110 401 : GET ADDRESS OF SECTION TABLE
7E D4 0115 402 : ASSUME NO PAGES WILL BE DELETED
5E 10 C2 0117 403 : SAVE ROOM FOR INRANGE AND RETRANGE
53 5E DO 011A 404 : REMEMBER SCRATCH AREA ADDRESS
56 24 A5 32 011D 405 : GET INDEX TO LAST SEC TBL ENTRY USED
5E 11 0121 406 : JOIN COMMON LOOP
18 A246 D5 0123 407 4$: TSTL SEC$L_REFCNT(R2)[R6] : ANY PAGES MAPPED?
55 19 0127 408 : BR IF NOTHING MAPPED TO THIS SECTION
0129 409 :
0129 410 : DO A $DELTVA FOR THIS SECTION.
0129 411 :
04 A3 08 A7 57 6246 DE 0129 412 : MOVAL (R2)[R6],R7 : ADDRESS OF SECTION TABLE ENTRY
04 A3 00CC C5 18 00 EF 012D 413 : EXTZV #SEC$V_VPX,#SEC$S_VPX,SEC$L_VXPFC(R7),4(R3) : GET START VPX
50 00000000'EF 07 9C 013E 414 : CMPZV #PHD$V_POLR,#PHD$S_POLR,PHD$L_POLRSTL(R5),4(R3) : IS VA IN PO?
04 A3 04 A3 50 C2 0146 415 : BGTRU 5$ : BR IF IT IS IN PO SPACE
04 A3 04 A3 16 00 EF 014A 416 : ROTL #7,SGN$GL_PTPAGCNT,R0 : GET LONGWD OFFSET FROM BEGIN OF PAGTBL
04 A3 04 A3 09 78 0151 417 : SUBL R0,4(R3) : GET VPX IN P1 SPACE
04 A3 04 A3 09 78 0157 418 5$: EXTZV #SEC$V_VPX,#SEC$S_VPX,4(R3),4(R3) : MASK OFF SIGN BITS
63 1C A7 09 78 0157 419 : ASHL #9,4(R3),4(R3) : GET VA OF FIRST PAGE MAPPED
63 04 A3 09 78 0157 420 : ASHL #9,SEC$L_PAGCNT(R7),(R3) : GET # OF BYTES IN SECTION
08 A3 FFFFFFFF 8F D1 015C 421 : ADDL2 4(R3),(R3) : GET VA OF LAST PAGE MAPPED
10 A3 01 DO 0160 422 : DECL (R3) : GET ADR OF LAST BYTE MAPPED
56 08 CO 0170 423 : $DELTVA_S (R3),8(R3) : DELETE PAGES FOR THIS SECTION
5E 10 CO 0178 424 : CMPL #-1,8(R3) : WERE ANY PAGES DELETED?
8E D5 0181 425 : BEQL 6$ : BR IF NONE DELETED
28 13 0183 426 6$: MOVL #1,16(R3) : INDICATE SECOND RUNDWN REQUIRED
00 DD 0186 427 8$: ADDL2 #SEC$C_LENGTH@-2,R6 : GET OFFSET TO NEXT SECTION TABLE ENTRY
00 FB 0188 428 : BLSS 4$ : GO CHECK NEXT SECTION
00000000'EF 01 FB 0183 429 : ADDL #16,SP : CLEAN INRANGE AND RETRANGE OFF STACK
0193 430 : TSTL (SP)+ : WERE ANY PAGES DELETED?
0193 431 : BEQL 55$ : NO, THEN DON'T BOTHER WITH 2ND RUNDWN
0193 432 : PUSHL #0 : SET MODE FOR RUNDOWN
0193 433 : CALLS #1,SYS$RUNDWN : RUNDOWN AT KERNEL MODE
0193 434 :
0193 435 :-----
0193 436 :
0193 437 : SCAN THE CHANNEL TABLE AND LOOK FOR ASSIGNED
0193 438 : CHANNELS. THERE SHOULD BE NONE.
0193 439 :
0193 440 :-----
55 00000000'9F 3C 0193 441 :
0193 442 10$: MOVZWL @#CTL$GW_CHINDX,R5 : GET MAXIMUM INDEX + 16

```

```

00000000'9F 55 19 13 019A 443 BEQL 55$ ; NO CHANNEL TABLE
55 55 CE 019C 444 MNEGL R5,R5 ; CONVERT TO NEGATIVE OFFSET
09 C1 019F 445 ADDL3 #CCB$B_AMOD, -
53 01A6 446 @#CTL$GL-CCBASE,R3 ; COMPUTE ADDRESS OF HIGHEST CHANNEL
6345 95 01A7 447 40$: TSTB (R3)[R5] ; IS THIS CHANNEL STILL ASSIGNED?
04 15 01AA 448 BLEQ 50$ ; BRANCH IF NOT.
01AC 449
01AC 450
01AC 451 : A NEGATIVE ACCESS MODE IN THE CCB IS USED TO RESERVE THE CHANNEL
01AC 452 : FOR USE BY THE F11BXQP TO PERFORM LOGICAL I/O ON BEHALF OF VIRTUAL
01AC 453 : I/O FUNCTIONS THAT IT PROCESSES (THE XQP MODIFIES THE CCB$L_UCB AND
01AC 454 : CCB$B_AMOD FIELDS PRIOR TO ACTUALLY USING IT). THE CHANNEL DOES NOT
01AC 455 : HAVE TO BE DEASSIGNED BECAUSE NO DEVICE REFERENCE COUNTS REFLECT ITS
01AC 456 : EXISTENCE.
01AC 457
01AC 458
01AC 459 BUG_CHECK FILCNTNONZ,FATAL ; IT REALLY SHOULD HAVE WORKED
01B0 460
55 10 CO 01B0 461 50$: ADDL2 #CCB$C_LENGTH,R5 ; NEXT CHANNEL
F2 19 01B3 462 BLSS 40$ ; LOOP THRU
01B5 463 55$:
01B5 464
01B5 465 :*****
01B5 466 : END OF BUG TRAP
01B5 467 :*****
01B5 468
01B5 469 : DISMOUNT ALL MOUNTED VOLUMES
01B5 470
1C A4 D5 01B5 471 TSTL PCB$L_OWNER(R4) ; IS THIS A SUB-PROCESS ?
56 0080 2A 12 01B8 472 BNEQ 20$ ; IF NEQ YES, SKIP IOC$DISMOUNT
00000000'EF C4 D0 01BA 473 MOVL PCB$JIB(R4),R6 ; GET JIB ADDRESS
56 00 B6 OF 01BF 474 JSB L^SCH$IOLCKW ; LOCK I/O DATABASE
00000000'EF 10 1D 01C9 475 REMQUE @JIB$M_TLFL(R6),R6 ; GET A VOLUME TO BE DISMOUNTED
53 D4 01CB 476 BVS 15$ ; NONE, FINISHED WITH DISMOUNT LIST
FE27' 30 01D4 477 JSB L^SCH$IOUNLOCK ; UNLOCK I/O DATABASE
BB 11 01D6 478 SETIPL #0 ; DROP IPL
00000000'EF 16 01D9 479 CLRL R3 ; UNLOAD UPON DISMOUNT
01DB 480 BSBW IOC$DISMOUNT ; DISMOUNT IT
01E1 481 BRB 10$ ; AND TRY FOR ANOTHER
01E4 482 15$: REF LABEL
01E4 483 JSB L^SCH$IOUNLOCK ; UNLOCK I/O DATABASE
40 A4 3E A4 B1 01E4 484 SETIPL #0 ; DROP IPL
EE 12 01EF 485 $DALLOC_S ACMODE=#0 ; DEALLOCATE ALL
3C A4 3A A4 B1 01F4 486 CMPW PCB$W_DIOCNT(R4),PCB$W_DIOLM(R4) ; WAIT FOR DIRECT I/O COMPLETION
F9 12 01F6 487 BNEQ 20$
01FB 488 30$: CMPW PCB$W_BIOCNT(R4),PCB$W_BIOLM(R4) ; AND BUFFERED I/O
01FD 489 BNEQ 30$
01FD 490
01FD 491
01FD 492 :-----
01FD 493 : DECREMENT INTERACTIVE AND BATCH JOB COUNT WHEN APPROPRIATE
01FD 494 :-----
01FD 495
01FD 496
01FD 497
1C A4 D5 01FD 498 TSTL PCB$L_OWNER(R4)
48 12 0200 499 BNEQ RELQUOTA ; WE DON'T DO IT FOR A SUBPROCESS

```

```

08 24 A4 19 E1 0202 500
0000C000'EF B7 0207 501
OB 11 020D 502
OE E1 C20F 503 59$:
06 24 A4 0211 504
00000000'EF B7 0214 505
021A 506
021A 507
021A 508
021A 509
021A 510
021A 511
021A 512
021A 513
021A 514
SE FO AE 9E 021A 515
6E 9F 021E 516
10 DD 0220 517
53 5E DO 0222 518
0225 519
0225 520
0225 521
0225 522
0225 523
0238 524
0238 525
0238 526
0238 527
SE 18 CO 0247 528
024A 529
024A 530
024A 531
024A 532
024A 533
024A 534
024A 535
56 70 A4 94 024A 536
1C A4 DO 024D 537
4E 13 0251 538
54 DD 0253 539
56 56 3C 0255 540
54 00000000'FF46 DO 0258 541
00000000'EF 16 0260 542
33 50 E9 0266 543
55 52 DO 0269 544
18 A5 04F3'CF 9E 026C 545
0272 546
OB A5 80 8F 90 0272 547
OC A5 60 A4 DO 0277 548
52 6E DO 027C 549
51 00000000'9F DO 027F 550
1C A5 5C A1 0110 C1 C3 0286 551
028E 552
20 A5 38 A1 DO 028E 553
0293 554
52 02 3C 0293 555
00000000'EF 16 0296 556

```

```

BBC #PCBSV_INTER,PCBSL_STS(R4),59$ : NOT INTERACTIVE
DECW SYS$GW_IJOBcnt : ONE LESS INTERACTIVE JOB
BRB DELETE_JT
BBC #PCBSV_BATCH,- : NOT BATCH JOB
PCBSL_STS(R4),DELETE_JT
DECW SYS$GW_BJOBcnt : ONE LESS BATCH JOB
-----
DELETE THE JOB-WIDE LOGICAL NAME TABLE IF THE PROCESS
IS NOT A SUB-PROCESS
-----
DELETE_JT:
MOVAB -JOB_STRING_LEN(SP),SP : DELETE THE JOB-WIDE LOGICAL NAME TABLE
PUSHAB (SP) : MAKE ROOM ON STACK FOR TABLE NAME
PUSHL #JOB_STRING_LEN : CONSTRUCT A DESCRIPTOR OF THE BUFFER
MOVL SP,R3 : WHICH IS TO CONTAIN THE TABLE NAME
: SAVE THE ADDRESS OF THE DESCRIPTOR
$FAO_S - : CONSTRUCT THE JOB-WIDE TABLE NAME
CTRSTR = JOB_FAO, -
OUTBUF = (R3), -
P1 = PCBSL_JIB(R4)
$DELLNM_S - : DELETE THE JOB-WIDE LOGICAL NAME TABLE
LOGNAM = (R3), -
TABNAM = LNM_SYSTEM_DIRECTORY_DESC
ADDL2 #JOB_STRING_LEN+8,SP : RESTORE THE STACK
-----
SEND REMAINING QUOTAS TO OWNER IF SUB-PROCESS
-----
RELQUOTA:
CLRB PCB$T_LNAME(R4) : RELEASE QUOTAS
MOVL PCB$S_OWNER(R4),R6 : REMOVE PROCESS NAME
BEQL MESSAGES : GET PID OF OWNER
: IF EQL, NOT A SUBPROCESS
PUSHL R4 : SAVE PCB ADDRESS
MOVZWL R6,R6 : EXTRACT PIX FROM OWNER PID
MOVL @L^SCH$GL_PCBVEC[R6],R4 : AND GET PCB
JSB L^EXE$ALLOCIRP : ALLOCATE AST CONTROL BLOCK
BLBC R0,20$ : SKIP QUOTA RETURN IF ERROR
MOVL R2,R5 : SET ADDRESS OF QUOTA AST BLOCK
MOVAB W^RETQUOTA,ACBSL_KAST(R5) : SET AST ADDRESS FOR SPECIAL
: KERNEL AST TO RETURN QUOTAS
MOVAB #<1@ACBSV_KAST>,ACBSB_RMOD(R5) : FLAG AS SPECIAL KERNEL AST
MOVL PCB$S_PID(R4),ACBSL_PID(R5) : SET TARGET PID FOR AST
MOVL (SP),R2 : GET PCB ADDRESS FOR PROCESS BEING DELETED
MOVL @#CTL$GL_PHD,R1 : GET PROCESS HEADER ADDRESS
SUBL3 PHD$S_EXTRACPU(R1),PHD$S_CPULIM(R1),ACB_L_CPULIM(R5) : SAVE CPULIMIT-LESS BONUS
: AND ACCUMULATED CPU TIME
MOVL PHD$S_CPUTIM(R1),ACB_L_CPUTIM(R5) : SET PRIORITY INCREMENT CLASS
MOVZWL #PRIS_RESAVL,R2 : QUEUE AST FOR CREATOR
JSB L^SCH$QAST

```

```

10 BA 029C 557 20$: POPR #*M<R4> ; RESTORE PCB ADDRESS
    029E 558 SETIPL #0 ; DROP IPL TO ISSUE SYSTEM SERVICE REQUESTS
    02A1 559
    02A1 560 -----
    02A1 561 :
    02A1 562 : TERMINATION MESSAGES
    02A1 563 :
    02A1 564 -----
    02A1 565 :
32 A4 B5 02A1 566 : SEND TERMINATION MESSAGES
    03 13 02A4 567 : TERMINATION MAILBOX SPECIFIED ?
    015A 30 02A6 568 : IF EQL, NO
    02A9 569 : SEND MESSAGE TO TERMINATION MAILBOX
05 24 A4 OE E1 02A9 570 : SEND MESSAGE TO JOB CONTROLLER
00 24 A4 OA E5 02AE 571 : IF BC, NOT A BATCH JOB
    02B3 572 20$: BBCC #PCBSV_SSRWAIT,PCBSL_STS(R4),20$; ENABLE RESOURCE WAIT MODE
    02B3 573 :
    02B5 574 : CLRL R5 ; NOT A SPECIAL KERNEL AST ROUTINE
    02BB 575 : JSB EXE$PRCDELMSG ; SEND PROCESS DELETE MESSAGE
    02BB 576 -----
    02BB 577 :
    02BB 578 : DESTROY PROCESS
    02BB 579 :
    02BB 580 -----
    02BB 581 :
    02BB 582 : .ENABLE LSB
    02BB 583
    02BB 584 DESTROY:
    02BB 585 SETIPL #IPL$ ASTDEL ; NO MORE ASTS AFTER THIS POINT
    SE 10 C2 02BE 586 : SUBL #16,SP ; 4 LONGWORDS OF STORAGE
    52 5E D0 02C1 587 : MOVL SP,R2 ; SAVE BASE
04 62 40000000 8F D0 02C4 588 : MOVL #<1@30>,(R2) ; SET STARTVA
    04 A2 7FFFFFFF 8F D0 02CB 589 : MOVL #<<1@31>-1>,4(R2) ; END AT VECTOR PAGE
    02D3 590 : WE CAN'T USE THE MACRO SINCE THE P1 VECTORS WILL DISAPPEAR
    02D3 591 : $DELTVA_S (R2),8(R2) ; DELETE FROM KERNEL STACK TO VECTORS
    00 DD 02D3 592 : PUSHL #0
    08 A2 7F 02D5 593 : PUSHAQ 8(R2)
    62 7F 02D8 594 : PUSHAQ (R2)
    80000000'EF 03 FB 02DA 595 : CALLS #3,SYS$DELTVA-P1SYSVECTORS+^X8000000 ; DELETE KSTK TO VECTORS
    02E1 596 : ENTER & EXIT DELTVA AT IPL(ASTDEL)
    02E1 597 : SETIPL W^DESTROY_SYNCH ; BLOCK SYSTEM EVENTS
    02E6 598 : SVPCTX ; AND LOCK PAGABLE CODE
    07 02E6 599 : ; SAVE PROCESS CONTEXT
    02E7 600 : ; AND GO TO INTERRUPT STACK
    55 6C A4 D0 02E7 601 : MOVL PCBSL_PHD(R4),R5 ; GET PROCESS HEADER ADDRESS
    5A 42 A5 3C 02EB 602 : MOVZWL PHD$W-PHVINDEX(R5),R10 ; GET BALANCE SLOT INDEX
    53 00000000'EF 9E 02EF 603 : MOVAB L^SCH$GL_NULLPCB,R3 ; GET POINT TO NULL PROCESS PCB
    00000000'EF 53 D0 02F6 604 : MOVL R3,L^SCH$GL_CURPCB ; SET AS CURRENT PCB
    51 00000000'FF4A 3C 02FD 605 : MOVZWL @L^PHV$GL_PIXBAS[R10],R1 ; GET PROCESS INDEX
    00000000'FF41 53 D0 0305 606 : MOVL R3,@L^SCH$GL_PCBVEC[R1] ; RELEASE PROCESS INDEX SLOT
    59 0C A5 3C 030D 607 : MOVZWL PHD$W_WSLOCK(R5),R9 ; POINT TO FIRST NON PERM PAGE
    6549 D4 0311 608 : CLRL (R5)[R9] ; MARK WITH A ZERO
    58 08 A5 3C 0314 609 : MOVZWL PHD$W_WSLIST(R5),R8 ; LONGWORD INDEX TO START OF WORKING SET LIS
    58 6548 DE 0318 610 : MOVAL (R5)[R8],R8 ; FORM BYTE ADDRESS OF START OF WSLIST
    52 88 D0 031C 611 40$: MOVL (R8)+,R2 ; GET VA OF PERM PAGE
    OA 13 031F 612 : BEQL 60$ ; DONE, NONE LEFT
    FB 52 E9 0321 613 : BLBC R2,40$ ; IGNORE IF NOT VALID

```

```

      F6 19 0324 614 BLSS 40$ ; OR IF HEADER PAGES (SYSTEM ADDRESS)
009A 30 0326 615 BSBW DELPAGE ; DELETE PAGE
      F1 11 0329 616 BRB 40$ ; CONTINUE
      032B 617
50 10 B4 0F 032B 618 60$: REMQUE @PCBSL_ASTQFL(R4),R0 ; REMOVE PENDING AST CONTROL BLOCKS
      OF 1D 032F 619 BVS 70$ ; NONE LEFT
      05 E0 0331 620 BBS #ACBSV NODELETE,- ; BR. IF NODELETE BIT IS SET. THIS
      F5 0B A0 0333 621 ACBSB_RMOD(R0),60$ ; SHOULD NOT HAPPEN BUT IF IT DOES MAY
      0336 622 ; RESULT IN LOSING POOL (WHICH IS
      0336 623 ; PREFERABLE OVER A DOUBLE DEALLOCATE).
      02 10 0336 624 BSBW DEANONPAGED ; DEALLOCATE SPACE FOR ACB
      F1 11 0338 625 BRB 60$ ; AND CONTINUE
      033A 626
00000000'EF 17 033A 627 DEANONPAGED: ;
      0340 629 JMP L^EXE$DEANONPAGED ; JUMP TO REACH REAL DEANONPAGED
      50 20 A4 D0 0340 630 70$: MOVL PCBSL_WSSWP(R4),R0 ; GET VBN OF AREA TO FREE IN SWAPFILE
      1C 13 0344 631 BEQL 80$ ; NONE, CONTINUE
      51 52 A5 3C 0346 632 MOVZWL PHD$W SWAPSIZE(R5),R1 ; SIZE OF AREA TO FREE
53 50 08 18 EF 034A 633 EXTZV #24,#8,R0,R3 ; SET SWAP FILE NUMBER
50 50 18 00 EF 034F 634 EXTZV #0,#24,R0,R0 ; GET VBN OF AREA
53 00000000'FF43 D0 0354 635 MOVL @L^MMG$GL_PAGSWPVC[R3],R3 ; GET ADDRESS OF PFL CONTROL BLOCK
      00000000'EF 16 035C 636 JSB L^MMG$DEACLOCAPAGFIL ; DEALLOCATE AREA
      50 00AC C4 D0 0362 637 80$: MOVL PCBSQ_PRIV+ARB$SL_RIGHTSLIST+8(R4),R0 ; GET EXTENDED RIGHTS LIST
      02 13 0367 638 BEQL 85$ ; BRANCH IF NONE
      CF 10 0369 639 BSBW DEANONPAGED ; DEALLOCATE IT
      50 0080 C4 D0 036B 640 85$: MOVL PCBSL_JIB(R4),R0 ; GET ADDRESS OF JIB FOR RELEASE
      0C 13 0370 641 BEQL 90$ ; BR IF NONE
      44 A0 B7 0372 642 DECW JIB$W_PRCNT(R0) ; ONE LESS PROCESS IN THIS TREE
54 A0 60 A4 D1 0375 643 Cmpl PCBSL_PID(R4),JIB$SL_MPID(R0) ; IS THIS THE MASTER PROCESS?
      02 12 037A 644 BNEQ 90$ ; BR IF NOT
      BC 10 037C 645 BSBW DEANONPAGED ; DEALLOCATE JIB
      50 1C A4 3C 037E 646 90$: MOVZWL PCBSL_OWNER(R4),R0 ; GET PIX FROM OWNER UIC
      14 13 0382 647 BEQL 100$ ; BR IF NOT A SUB-PROCESS
50 00000000'FF40 D0 0384 648 MOVL @L^SCH$GL_PCBVECC[R0],R0 ; GET ADDRESS OF OWNER PCB
      42 A0 B7 038C 649 DECW PCBSW_PRCNT(R0) ; DECREMENT COUNT OF SUB-PROCESSES
      50 01 D0 038F 650 MOVL #RSNS$ASTWAIT,R0 ; NOTIFY PARENT PROCESS THAT ITS
      00000000'EF 16 0392 651 JSB L^SCH$RAVAIL ; SUBPROCESS COUNT HAS BEEN DECREMENTED
      50 54 D0 0398 652 100$: MOVL R4,R0 ; SET ADDRESS OF PCB FOR RELEASE
      9D 10 039B 653 BSBW DEANONPAGED ; AND DELETE IT
      00000000'EF B7 039D 654 DECW L^SWP$GW_BALCNT ; DECREASE COUNT IN BALANCE SET
      00000000'EF 16 03A3 655 JSB L^SCH$SWPWAKE ; WAKE SWAPPER
00000000'FF4A 00 B2 03A9 656 MCOMW #0,@L^PHV$GL_PIXBAS[R10] ; INDICATE PROCESS HEADER TO BE DELETED
      00000000'EF B6 03B1 657 INCW L^SCH$GW_DELPHDCT ; FLAG SWAPPER'S ATTENTION
      00000000'EF B7 03B7 658 DECW L^SCH$GW_PROCCNT ; ONE LESS PROCESS NOW
      00000000'EF 17 03BD 659 JMP L^SCH$SCRED ; AND SCHEDULE FOR ANOTHER PROCESS
      03C3 660
      03C3 661 .DISABLE LSB

```



```

03C3 663 .SBTTL DELPAGE - DELETE PAGE
03C3 664 :++
03C3 665 : FUNCTIONAL DESCRIPTION:
03C3 666 : DELPAGE DELETES THE CONTENTS OF THE SPECIFIED PAGE AND PLACES
03C3 667 : THE PFN ON THE FREE PAGE LIST.
03C3 668 :
03C3 669 : CALLING SEQUENCE:
03C3 670 : BSB DELPAGE
03C3 671 :
03C3 672 : INPUT PARAMETERS:
03C3 673 : R2 - VIRTUAL ADDRESS OF PAGE TO RELEASE
03C3 674 : R4 - PCB ADDRESS OF PROCESS OWNING PAGE
03C3 675 : R5 - PROCESS HEADER ADDRESS OF PROCESS OWNING PAGE
03C3 676 :
03C3 677 : OUTPUT PARAMETERS:
03C3 678 : NONE
03C3 679 :
03C3 680 :--
03C3 681 :
03C3 682 DELPAGE:
03C3 683 JSB MMG$SVAPTECHK ; DELETE PAGE
03C9 684 BBCC #PTE$V_VALID,(R3),50$ ; GET SVA OF PTE FOR PAGE
03CD 685 BBCC #PTE$V_MODIFY,(R3),10$ ; EXIT IF NOT VALID
03D1 686 10$: EXTZV #0,#PTE$S_PFN,(R3),R0 ; CLEAR MODIFY INDICATOR
03D6 687 DECF L^,JSB ; GET PFN FROM PTE
03E5 688 BGTR 50$ ; DROP REFERENCE COUNT ON PAGE
03E7 689 BICB #PFNSM_MODIFY,@L^PFNSAB_STATE[R0] ; BR IF CANT RELEASE YET
03F0 690 20$: JSB MMG$DE[CONPFN] ; MAKE SURE NO MODIFY BIT
03F6 691 ASSUME PFNSC_FREPAGLST EQ 0 ; DELETE CONTENT OF PFN
03F6 692 CLRL R2 ; INDICATE FREE LIST
03F8 693 JSB MMG$INSPFNH ; INSERT AT HEAD OF FREE LIST
03FE 694 50$: RSB ;
03FF 695 ;
03FF 696 DESTROY_SYNCH: ; END OF NONPAGABLE AREA
03FF 697 .LONG IPL$_SYNCH ; SYNCHRONIZATION IPL TO BLOCK SYSTEM EVENTS

```

```

0403 699 .SBTTL TERMMBX - SEND MESSAGE TO TERMINATION MAILBOX
0403 700
0403 701 -----
0403 702
0403 703 BUILD PROCESS TERMINATION MESSAGE AND SEND TO TERMINATION
0403 704 MAILBOX UNIT.
0403 705 -----
0403 706
0403 707
0403 708 TERMMSG:
55 5A 5E DO 0403 709 ; SEND TERMINATION MESSAGES
    68 A4 DD 0406 710 ; SAVE STACK STATE
7E 00000000'9F DO 0409 711 ; SEND EXTENDED PID OF SUBPROCESS OWNER (0 =
    00000000'9F 7D 0410 712 ; GET PROCESS HEADER ADDRESS
    00000000'9F DD 0417 713 ; LOGIN TIME
    00000000'9F DD 0417 714 ; COUNT OF MOUNTED VOLUMES
    54 A5 DD 041D 715 ; DIRECT I/O COUNT
    58 A5 DD 0420 716 ; BUFFERED I/O COUNT
    00000000'9F DD 0423 719 ; PEAK WORKING SET SIZE
    00 DD 0429 721 ; PEAK PAGING FILE USAGE
    4C A5 DD 042B 723 ; TOTAL PROCESS PAGE FAULTS
    38 A5 DD 042E 725 ; SAVE ACCUMULATED CPU TIME FOR PROCESS
50 0000000C'9F 9E 0431 727 ; GET BASE ADDRESS
    7E 70 7D 0438 729 ; LAST HALF OF USER NAME
    70 DD 043B 730 ; FIRST LONGWORD OF USER NAME
50 0000'CO 9E 043D 731 ; POINT PAST ACCOUNT
    7E 60 7D 0442 732 ; SET ACCOUNT INTO RECORD
7E 00000000'EF 7D 0445 734 ; LOGOUT TIME
    00 DD 044C 736 ; JOBID
    64 A4 DD 044E 738 ; EXTENDED PROCESS ID
    00000000'9F DD 0451 740 ; FINAL STATUS OF PROCESS
    03 DD 0457 742 ; MESSAGE TYPE
    57 5E DO 0459 744 ; MESSAGE ADDRESS
    50 32 A4 3C 045C 746 ; TERMINATION MAILBOX UNIT
7E 3A303030 3030424D 8F 7D 0460 747 ; SKELETON FOR DEVICE STRING
    51 D4 046B 748 ; INIT FOR EDIV
    53 07 AE 9E 046D 749 ; POINTER TO LSD OF DEVICE NAME
52 50 50 0A 7B 0471 750 10$: ; CONVERT A DECIMAL DIGIT
    73 52 80 0476 751 ; SET DIGIT INTO NAME STRING IN ASCII
    50 D5 0479 752 ; CHECK FOR END
    F4 12 047B 753 ; NOT YET, CONVERT ANOTHER DIGIT
    5E DD 047D 754 ; BUILD STRING DESCRIPTOR FOR
    08 DD 047F 755 ; MAILBOX NAME

```

S
V

M
-
T
B
T
M

```

58 SE DO 0481 756      MOVL  SP,R8      ; AND SAVE POINTER TO NAME DESCRIPTOR
59 7E DE 0484 757      MOVAL -(SP),R9    ; RESERVE SPACE FOR CHANNEL
      0487 758      :
      0487 759      : CHECK FOR PRESENCE OF CHANNEL TABLE, IF NON-EXISTENT (I.E. PROCESS DELETED
      0487 760      : BEFORE THE CREATION OF IT) USE A PAGE OF SUPERVISOR STACK FOR IT.
      0487 761      :
00000000'9F B5 0487 762      TSTW  @#CTLSGW_NMIOCH ; IS THERE A VALID CHANNEL TABLE?
      25 12 048D 763      BNEQ  20$          ; YES, SKIP TO ASSIGN
      54 DD 048F 764      PUSHL  R4              ; SAVE PCB ADDRESS
53 00000008'9F DO 0491 765      MOVL  @#CTLSAL STACK+<4*PSL$C_SUPER>,R3 ; PICK UP SUPER SP
      FO A3 9E 0498 766      MOVAB -CCB$C LENGTH(R3),-
00000000'9F 049B 767      @#CTLSGL CCBASE ; MAKE IT TABLE BASE
0200 BF 00 6E 00 2C 04A0 768      MOVCS #0,(SP),#0,#512,-512(R3); CLEAR IT OUT
      FE00 C3 04A7
      54 8ED0 04AA 769      POPL  R4              ; RESTORE PCB ADDRESS
      1F 3C 04AD 770      MOVZWL #<512/CCB$C LENGTH>-1,- ; SET NUMBER OF CHANNELS
00000000'9F 04AF 771      @#CTLSGW_NMIOCH
      04B4 772      :
      04B4 773      : ASSIGN CHANNEL TO MAILBOX AND WRITE THE MESSAGE
      04B4 774      :
      04B4 775      20$:
      04B4 776      $ASSIGN_S  DEVNAM=(R8),- ; ATTEMPT TO ASSIGN CHANNEL TO
      04B4 777      CHAN=(R9) ; MAILBOX
      21 50 E9 04C1 778      BLBC  R0,30$ ; UNABLE TO ASSIGN, IGNORE
      04C4 779      $QIO_S  CHAN=(R9),- ; SEND TERMINATION MESSAGE
      04C4 780      FUNC=#<IO$_WRITEVBLK!IOSM_NOW>,-
      04C4 781      P1=(R7),- ; MESSAGE ADDRESS
      04C4 782      P2=#ACC$C_TERMLEN; MESSAGE SIZE
      04E5 783      30$:
      04E5 784      $DASSGN_S CHAN=(R9) ; DEASSIGN CHANNEL
5E SA DO 04EF 785      MOVL  -R10,SP ; RESTORE STACK STATE
      05 04F2 786      RSB
      04F3 787

```

```

04F3 789      .SUBTITLE      Return Unused CPU Time Limit to Parent
04F3 790
04F3 791      :+
04F3 792      : Functional Description:
04F3 793      :
04F3 794      : RETQUOTA is entered via a special kernel AST directed to the
04F3 795      : creator of a subprocess. The AST control block contains the
04F3 796      : unused CPU time limit to be returned to the creator.
04F3 797      :
04F3 798      : The CPU limit of the parent has any unused CPU time from the
04F3 799      : subprocess added to it.
04F3 800      :
04F3 801      : Input Parameters:
04F3 802      :
04F3 803      : R4      PCB Address of Current (Parent) Process
04F3 804      : R5      Pointer to AST Control Block
04F3 805      :
04F3 806      : ACB_L_CPULIM(R5)      Original CPU Limit of Subprocess
04F3 807      : ACB_L_CPUTIM(R5)   CPU Time Used by Subprocess
04F3 808      :
04F3 809      : Output Parameter:
04F3 810      :
04F3 811      : PHDSL_CPULIM(R1)      The CPU limit of the parent is adjusted
04F3 812      : according to the following rules.
04F3 813      :
04F3 814      : If the subprocess has unused CPU time limit (a credit) ,
04F3 815      : that credit is added to the parent's CPU limit.
04F3 816      :
04F3 817      : If the subprocess exceeded its CPU time allotment, an attempt
04F3 818      : is made to pass the deficit to the parent.
04F3 819      :
04F3 820      : If the deficit is smaller than the parent's CPU limit, the
04F3 821      : deficit is simply subtracted from CPU limit.
04F3 822      :
04F3 823      : If the deficit is larger than the parent's CPU limit, the
04F3 824      : parent is forced into the CPU time expiration path by
04F3 825      : loading its CPUTIM into its CPULIM field, indicating
04F3 826      : that its CPU time limit has expired. (Note that loading CPULIM
04F3 827      : into CPUTIM has the same effect but causes double accounting
04F3 828      : for the CPU time already charged to the subprocess.)
04F3 829      :
04F3 830      : Side Effects:
04F3 831      :
04F3 832      : The AST control block is released to nonpaged pool.
04F3 833      :
04F3 834      : Environment:
04F3 835      :
04F3 836      : IPLS_ASTDEL while carrying pointer to AST control block.
04F3 837      :-
04F3 838
04F3 839 RETQUOTA:
04F3 840      MOVL      @#CTLSGL PHD,R1      ; Get safe pointer to process header
04F3 841      TSTL      PHDSL_CPULIM(R1)     ; Check for no quota
04F3 842      BEQL      20$                    ; Skip return if no quota
04F3 843      SUBL3     ACB_L_CPUTIM(R5),ACB_L_CPULIM(R5),R0 ; Get difference
04F3 844      BGEQU     10$                    ; Excess or deficit (Branch if excess)
04F3 845      MNEGL     R0,-(SP)              ; Handle deficit very carefully

```

```

51 00000000'9F D0
      SC A1 D5
      1C 13
50 1C A5 20 A5 C3
      10 1E
      7E 50 CE

```

```
5C A1 8E D1 050A 846      CML  (SP)+,PHDSL_CPULIM(R1) ; Can creator accommodate deficit?
      07 1F 050E 847      BLSSU 10$ ; Branch if yes
5C A1 38 A1 D0 0510 848      MOVL  PHDSL_CPUTIM(R1),PHDSL_CPULIM(R1) ; Otherwise, force CPU time
      04 11 0515 849      ; expiration by setting limit to time
      0517 850      BRB 20$ ; Deallocate AST control block
      0517 851
5C A1 50 C0 0517 852 10$: ADDL2 R0,PHDSL_CPULIM(R1) ; Give back excess
      50 55 D0 051B 853 20$: MOVL R5,R0 ; Get address for release
00000000'GF 17 051E 854      JMP G^EXE$DEANONPAGED ; Return via EXE$DEANONPAGED
      0524 855
      0524 856      .END
```

SST1	= 00000001	EXESRESETVEC	*****	X	02
SST2	= 00000004	IOSM NOW	= 00000040		
ACBSB_RMOD	= 00000008	IOS WRITEVBLK	= 00000030		
ACBSL_AST	= 00000010	IOCSDISMOUNT	*****	X	02
ACBSL_KAST	= 00000018	IPLS_ASTDEL	= 00000002		
ACBSL_PID	= 0000000C	IPLS_SYNCH	= 00000008		
ACBSV_KAST	= 00000007	JIBSL_MPID	= 00000054		
ACBSV_NODELETE	= 00000005	JIBSL_MTLFL	= 00000000		
ACB_L_CPULIM	= 0000001C	JIBSW_PRCNT	= 00000044		
ACB_L_CPUTIM	= 00000020	JOB_FAO	= 0000001C	R	02
ACCSL_TERMLEN	= 00000054	JOB_STRING_LEN	= 00000010		
ACCSL_BIOCNT	= 0000003C	LNMSYSTEM_DIRECTORY_DESC	= 00000000	R	02
ACCSL_CPUTIM	= 0000002C	MESSAGES	000002A1	R	02
ACCSL_DIOCNT	= 00000040	MMGSDEALLOCPAGFIL	*****	X	02
ACCSL_FINALSTS	= 00000004	MMGSDELCONPFN	*****	X	02
ACCSL_JOBID	= 0000000C	MMGSGL_PAGSWPVC	*****	X	02
ACCSL_OWNER	= 00000050	MMGSINSPFNH	*****	X	02
ACCSL_PAGEFLTS	= 00000030	MMGSREFCNTNEG	*****	X	02
ACCSL_PGFLPEAK	= 00000034	MMGS\$VAPTECHK	*****	X	02
ACCSL_PID	= 00000008	MSG\$ DELPROC	= 00000003		
ACCSL_VOLUMES	= 00000044	P1SYSVECTORS	*****	X	02
ACCSL_WSPEAK	= 00000038	PCBSB_ASTACT	= 0000000C		
ACCSQ_LOGIN	= 00000048	PCBSB_DPC	= 0000002A		
ACCSQ_TERMTIME	= 00000010	PCBSL_ASTQFL	= 0C000010		
ACCST_ACCOUNT	= 00000018	PCBSL_EOWNER	= 00000068		
ACCST_USERNAME	= 00000020	PCBSL_EPID	= 00000064		
ACCSW_MSGTYP	= 00000000	PCBSL_JIB	= 00000080		
ARBSL_RIGHTSLIST	= 00000020	PCBSL_OWNER	= 0000001C		
BUGS_FILCNTNONZ	*****	PCBSL_PHD	= 0000006C		
CCBSB_AMOD	= 00000009	PCBSL_PID	= 00000060		
CCBSC_LENGTH	= 00000010	PCBSL_STS	= 00000024		
CTLSAC_STACK	*****	PCBSL_WSSWP	= 00000020		
CTLSGL_CCBBASE	*****	PCBSQ_PRIV	= 00000084		
CTLSGL_FINALSTS	*****	PCBST_LNAME	= 00000070		
CTLSGL_PHD	*****	PCBSV_BATCH	= 0000000E		
CTLSGL_USRUNDWN	*****	PCBSV_DELPEN	= 00000001		
CTLSGL_VOLUMES	*****	PCBSV_INTER	= 00000019		
CTLSGL_WSPEAK	*****	PCBSV_RESPEN	= 00000005		
CTLSGQ_LOGIN	*****	PCBSV_SSRWAIT	= 0000000A		
CTLSGW_CHINDX	*****	PCBSW_BIOCNT	= 0000003A		
CTLSGW_NMI0CH	*****	PCB_J_BIOLM	= 0000003C		
CTLST_ACCOUNT	*****	PCBSW_DIOCNT	= 0000003E		
CTLST_USERNAME	*****	PCBSW_DIOLM	= 00000040		
DEANONPAGED	0000033A	PCBSW_PRCNT	= 00000042		
DELETE	0000002F	PCBSW_TMBU	= 00000032		
DELETE_JT	0000021A	PFNSAB_STATE	*****	X	02
DELPAGE	000003C3	PFNSAW_REFcnt	*****	X	02
DESTROY	000002BB	PFNSC_FREPAGLST	= 00000000		
DESTROY_SYNCH	000003FF	PFNSM_MODIFY	= 00000080		
EVTS_RESUME	*****	PHDSL_BIOCNT	= 00000058		
EXESALLOCIRP	*****	PHDSL_CPULIM	= 0000005C		
EXESDEANONPAGED	*****	PHDSL_CPUTIM	= 00000038		
EXESDELPRC	00000000	PHDSL_DIOCNT	= 00000054		
EXESGL_USRUNDWN	*****	PHDSL_EXTRACPU	= 00000110		
EXESGQ_SYSTIME	*****	PHDSL_POLRASTL	= 000000CC		
EXESNAMPID	*****	PHDSL_PAGEFLTS	= 0000004C		
EXESPRCDELMSG	*****	PHDSL_PSTBASOFF	= 00000020		

SYSDELPRC
Symbol table

- DELETE PROCESS SYSTEM SERVICE H 10

16-SEP-1984 01:57:52 VAX/VMS Macro V04-00
5-SEP-1984 03:52:26 [SYS.SRC]SYSDELPRC.MAR;1

S
V

PHDSS_POLR	=	00000018		
PHDSV_POLR	=	00000000		
PHDSW_PHVINDEX	=	00000042		
PHDSW_PSTLAST	=	00000024		
PHDSW_SWAPSIZE	=	00000052		
PHDSW_WSLIST	=	00000008		
PHDSW_WSLOCK	=	0000000C		
PHVSGC_PIXBAS		*****	X	02
PRS_IPC	=	00000012		
PRIS_RESAVL	=	00000002		
PSLSC_SUPER	=	00000002		
PTESS_PFN	=	00000015		
PTESV_MODIFY	=	0000001A		
PTESV_VALID	=	0000001F		
RELQUOTA		0000024A	R	02
RETQUOTA		000004F3	R	02
RSNS_ASTWAIT	=	00000001		
RUNDWN		00000100	R	02
SCHSC_SUPPIX		*****	X	03
SCHSGC_CURPCB		*****	X	02
SCHSGL_MAXPIX		*****	X	02
SCHSGL_NULLPCB		*****	X	02
SCHSGL_PCBVEC		*****	X	02
SCHSGW_DELPHDCT		*****	X	02
SCHSGW_PROCCNT		*****	X	02
SCHSIOLOCKW		*****	X	02
SCHSIOUNLOCK		*****	X	02
SCHSNEWLVL		*****	X	02
SCHSQAST		*****	X	03
SCHSRAVAIL		*****	X	02
SCHSRSE		*****	X	03
SCHSRWAIT		*****	X	02
SCHSSCHED		*****	X	02
SCHSSWPWAKE		*****	X	02
SECSC_LENGTH	=	00000020		
SECSL_PAGCNT	=	0000001C		
SECSL_REFCNT	=	00000018		
SECSL_VPXPC	=	00000008		
SECSS_VPX	=	00000016		
SECSV_VPX	=	00000000		
SGNSGC_PTPAGCNT		*****	X	02
SSS_NOREXPR	=	000008E8		
SSS_NORMAL	=	00000001		
SUBDELETE		000000B3	R	02
SUBWIPL		000000FC	R	02
SWPSGW_BALCNT		*****	X	02
SYSSASSIGN		*****	GX	02
SYSSDALLOC		*****	GX	02
SYSSDASSGN		*****	GX	02
SYSSDELLNM		*****	GX	02
SYSSDELPRC		*****	GX	02
SYSSDELTVA		*****	GX	02
SYSSFAO		*****	X	02
SYSSGW_BJOBcnt		*****	X	02
SYSSGW_IJOBcnt		*****	X	02
SYSSQID		*****	GX	02
SYSSRMSRUNDWN		*****	X	02

SYSSRUNDWN
TERMMSG
TMP...

***** X 02
00000403 R 02
= 00000000

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
YSEXEPAGED	00000524 (1316.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
AEXENONPAGED	00000052 (82.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.03	00:00:02.38
Command processing	110	00:00:00.53	00:00:05.52
Pass 1	494	00:00:19.82	00:01:01.05
Symbol table sort	0	00:00:03.34	00:00:10.41
Pass 2	173	00:00:03.97	00:00:12.52
Symbol table output	21	00:00:00.19	00:00:00.91
Psect synopsis output	8	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	838	00:00:27.92	00:01:32.83

The working set limit was 1950 pages.
113562 bytes (222 pages) of virtual memory were used to buffer the intermediate code.
There were 110 pages of symbol table space allocated to hold 2048 non-local and 46 local symbols.
856 source lines were read in Pass 1, producing 25 object records in Pass 2.
46 pages of virtual memory were used to define 44 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	18
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	23
TOTALS (all libraries)	41

2279 GETS were required to define 41 macros.

There were no errors, warnings or information messages.

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

0383 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

This image shows a grid of 144 small, faint terminal window screenshots arranged in a 12x12 pattern. Each window displays a different system utility or diagnostic tool. The text within the windows is mostly illegible due to low contrast and small size, but several are clearly labeled with their names:

- SYSCRMPSC LIS (top-left)
- SYSDCLEXH LIS (row 2, column 7)
- SYSDVALC LIS (row 3, column 10)
- SYSDGBLSC LIS (row 4, column 11)
- SYSENQDEQ LIS (row 5, column 12)
- SYSDCLMH LIS (row 6, column 8)
- SYSDERLMB LIS (row 7, column 9)
- SYSDASSGN LIS (row 8, column 6)
- SYSDLPRC LIS (row 9, column 7)

The overall appearance is that of a dense array of system diagnostic outputs, typical of a VAX/VMS environment.