

Table of contents

2-	1	Status flags in LSW tables
13-	1	Privilege Flags
14-	1	Job Execution States
15-	1	System message buffers
16-	1	Configuration and system generation words
17-	1	Overlay system region offsets
17-	11	Virtual overlay handler offsets
18-	1	I/O System Tables
19-	1	I/O queue entry
20-	1	Completion queue entry
21-	1	I/O timer requests
22-	1	Swapper Command Packets
23-	1	Job initiation information blocks
24-	1	Job monitoring control block
25-	1	Process window control block
27-	1	Printer attribute flags
28-	1	Key definition blocks
29-	1	Synch request control block
29-	14	Vector control block
30-	1	TSXUCL data base definitions
31-	1	PLAS region and window descriptor blocks
33-	1	Unibus Map Register descriptor block
34-	1	Mapped I/O control blocks
35-	1	Cached I/O control block
36-	2	Shared run-time descriptor block
37-	1	Device status flags
39-	1	Device handler file format
39-	46	Special device function codes
40-	1	File directory entries
41-	16	Directory cache device table
42-	1	Assign table
42-	14	Access command table
42-	30	Device allocation table
43-	1	Fork queue entry block
44-	1	Installed program table
45-	1	Memory allocation table values
46-	1	PRO-350 Related Values
47-	1	Spooling Control Tables
49-	1	Accounting file format
50-	1	Log file control flags
51-	1	Shared file control tables
54-	1	Message communication control tables
55-	1	Generic data set control and status flags
55-	12	DL11 Control and Status Registers
56-	1	DZ11 Control and Status Registers
57-	1	DH11 Control and Status Registers
58-	1	DHV11 Control and Status Registers
59-	1	Line Speed Codes
60-	1	Communication Line (CL) Handler Flags
61-	1	Memory Management values
62-	1	Terminal type names
62-	19	System Editor names
63-	1	Flags in Job Status Word (JSW)
63-	19	Performance monitor control and status flags
63-	37	Simulated RMON parameters
64-	1	Misc. parameters
65-	1	User error severity status codes

TSDEFS -- Table definitions MACRO V05.04 Thursday 17-Dec-87 07:55
Table of contents

66-	1	ASCII Character codes
67-	1	Job Context Area

```
1          .TITLE  TSDEFS -- Table definitions
2          .ENABL  LC
3          .ENABL  AMA
4 000000    .CSECT  TSDEFS
5          .GLOBL  TSDEFS
6 000000    TSDEFS:
7          ;
8          ; TSDEFS is the module of TSX-Plus that contains the template definitions
9          ; for various tables and flags used by TSX-Plus.
10         ;
11         ; Copyright (c) 1980,1981,1982,1983,1984,1985.
12         ; S&H Computer Systems, Inc.  Nashville, Tn
13         ;
14         ; -----
15         ; Macro definitions
16         ;
17         ; Macro to define the start of a template.
18         ;
19         .MACRO  RZ
20 $AC          =          0
21         .ENDM   RZ
22         ;
23         ; Macro to define a symbol and reserve the specified number of words.
24         ;
25         .MACRO  RW          NAME,SIZE
26         .DSABL  CRF
27         .GLOBL  NAME
28         .IF     NE,<$AC&1>
29 $AC          =          $AC+1
30         .ENDC
31         .ENABL  CRF
32         .LIST
33                                     NAME=$AC
34         .NLIST
35         .DSABL  CRF
36 $AC          =          $AC+<2*<SIZE>>
37         .ENABL  CRF
38         .ENDM   RW
39         ;
40         ; Macro to define a symbol and reserve the specified number of bytes.
41         ;
42         .MACRO  RB          NAME,SIZE
43         .DSABL  CRF
44         .GLOBL  NAME
45         .ENABL  CRF
46         .LIST
47                                     NAME=$AC
48         .NLIST
49         .DSABL  CRF
50 $AC          =          $AC+<SIZE>
51         .ENABL  CRF
52         .ENDM   RB
53         ;
54         ; Macro to define a symbol equated to a constant value.
55         ;
56         .MACRO  M          NAME,VALUE
57         .DSABL  CRF
```

58		. GLOBL	NAME
59		. ENABL	CRF
60	NAME	=	VALUE
61		. ENDM	M

			.SBTTL	Status flags in LSW tables
1				
2				
3				-----
4				; Status flags in LSW table.
5	000000	M	\$DILUP	100000 ;Line is logged on
6	000000	M	\$KINIT	40000 ;KMDN initialization completed
7	000000	M	\$INCOR	20000 ;Job is in memory now
8	000000	M	\$NDMEM	10000 ;Job waiting for memory expansion
9	000000	M	\$DOOFF	4000 ;In the process of logging off
10	000000	M	\$INIT	2000 ;Line has been initialized
11	000000	M	\$DISCN	1000 ;Line disconnect has occurred
12	000000	M	\$SUSPN	40 ;Job has been suspended by another job
13	000000	M	\$VNOTT	20 ;Virtual line not connected to a terminal
14	000000	M	\$DETCH	10 ;Job is detached
15	000000	M	\$1ESC	4 ;One Escape received
16	000000	M	\$FPUEX	2 ;FPU exception interrupt occurred for job
17	000000	M	\$CTRLC	1 ;Ctrl-C received. Abort task.
18	000000	M	\$SPLJB	\$VNOTT!\$DETCH ;Low-priority job

```

1      ;-----
2      ; Status flags in LSW2 table.
3      ;
4 000000 M      $SCOPE 100000 ;CRT type terminal
5 000000 M      $ECHO  40000  ;Echoplex mode
6 000000 M      $TAPE  20000  ;Terminal is in "paper-tape" mode (x-on/x-off control)
7 000000 M      $BBIT  10000  ;8-bit character support
8 000000 M      $START 4000   ;Auto start line during initialization
9 000000 M      $TAB   1000   ;Do not simulate tabs for terminal
10 000000 M      $FORM  400    ;Do not simulate form feeds for terminal
11 000000 M      $AUTO  200    ;Autobaud speed
12 000000 M      $PAGE  100    ;Enable ctrl-S/ctrl-Q processing
13 000000 M      $LC    40     ;Allow lower case character input
14 000000 M      $NOVLN 20     ;Disable virtual line switching
15 000000 M      $DEFER 10     ;Deferred type character echoing
16 000000 M      $QTSET  4      ;Set TT quiet
17 000000 M      $SYSPS  2      ;Require system password before logon
18 000000 M      $PHONE  1      ;Dial-up phone line
19      ;
20      ; Combinations of LSW2 flags based on terminal types.
21      ;
22      ; Flags for LA36.
23 000000 M      LA36FL  0
24 000000 M      LA36NO  $SCOPE!$TAB!$FORM
25      ; Flags for LA120.
26 000000 M      LA12FL  $PAGE!$TAB!$FORM
27 000000 M      LA12NO  $SCOPE
28      ; Flags for VT52.
29 000000 M      VT52FL  $SCOPE!$TAB!$PAGE
30 000000 M      VT52NO  $FORM
31      ; Flags for VT100
32 000000 M      VT10FL  $SCOPE!$TAB!$PAGE
33 000000 M      VT10NO  $FORM
34      ; Flags for VT200
35 000000 M      VT20FL  $SCOPE!$TAB!$PAGE!$BBIT
36 000000 M      VT20NO  $FORM
37      ; Flags for Diablo.
38 000000 M      DIABFL  $FORM!$PAGE
39 000000 M      DIABNO  $SCOPE!$TAB
40      ; Flags for Qume.
41 000000 M      QUMEFL  $FORM!$PAGE
42 000000 M      QUMENO  $SCOPE!$TAB
43      ; Flags for ADM3A.
44 000000 M      ADM3FL  $SCOPE
45 000000 M      ADM3NO  $TAB!$FORM
46      ; Flags for Hazeltine terminals
47 000000 M      HAZLFL  $SCOPE
48 000000 M      HAZLNO  $TAB!$FORM

```

1				
2				
3				
4	000000	M	\$NOIN 100000	; Ignore input from line
5	000000	M	\$CARUP 40000	; Carrier is detected for dial-up line
6	000000	M	\$HARD 20000	; Line is a real (hardware connected) line
7	000000	M	\$DEAD 10000	; Device is not installed
8	000000	M	\$TRNSP 4000	; Output is in transparent mode
9	000000	M	\$XCHAR 2000	; Character transmission is in progress
10	000000	M	\$CTRLD 1000	; Ctrl-D received. Discard TT output.
11	000000	M	\$RBOU 400	; Rubout sequence in progress
12	000000	M	\$NQOUT 200	; Temporarily suppress TT output
13	000000	M	\$1STCH 100	; Some input characters have been received since CR
14	000000	M	\$CTRLW 40	; Ctrl-W pending
15	000000	M	\$CTRLS 20	; Ctrl-S TT output suspension in effect
16	000000	M	\$DODFR 10	; We are now deferring char echoing
17	000000	M	\$GCECO 4	; GETCHR must echo characters
18	000000	M	\$GCESC 2	; GETCHR: next char part of VT50 esc sequence
19	000000	M	\$TRCHR 1	; Send next char in transparency mode
20				
21	000000	M	KL3CLR \$TRNSP!\$CTRLD!\$TRCHR	

1				
2				
3				
4	000000	M	\$RFRSH 100000	; Currently doing window refresh
5	000000	M	\$CFCL 40000	; Command file is expanded CCL command
6	000000	M	\$QUIET 20000	; Don't list command file
7	000000	M	\$INKMN 10000	; KMON is running
8	000000	M	\$UCTLC 4000	; Ctrl-C is user defined activation character
9	000000	M	\$SETCC 2000	; Need to tell user about 2 ctrl-C's
10	000000	M	\$ODTMD 1000	; ODT character activation mode
11	000000	M	\$CFOPN 400	; Command file channel is open
12	000000	M	\$CFALL 200	; Get all TT characters from command file
13	000000	M	\$GTLIN 100	; .GTLIN is being executed
14	000000	M	\$CFDCC 40	; Ctrl-C deferred till terminating .GTLIN
15	000000	M	\$FORMO 20	; Do form-feed on TT write of block 0
16	000000	M	\$TTERR 10	; Error occurred on TT input
17	000000	M	\$CFSOT 4	; Suppress program output
18	000000	M	\$HITTY 2	; High efficiency TT mode
19	000000	M	\$FLAGC 1	; User has been told no TT chars available
20				
21	000000	M	KL4CLR \$UCTLC!\$SETCC!\$ODTMD!\$GTLIN!\$HITTY!\$TTERR	
22	000000	M	CFLFL4 \$QUIET!\$CFALL!\$CFSOT	; Command file flags that are pushed & popped


```

1
2 ; -----
3 ; Status flags in LSW5 table.
4 000000 M      $CCLRN 100000 ; CCL Translator is running
5 000000 M      $TECO  40000  ; TECO is the default editor
6 000000 M      $WILD  20000  ; Use implicit wildcards
7 000000 M      $PRGLK 10000  ; Program is locked to line
8 000000 M      $SCCA   4000   ; Suppress Control-C abort
9 000000 M      $1CTL  2000   ; One Ctrl-C char received
10 000000 M      $CLTST 1000   ; Display CCL generated commands
11 000000 M      $OITIM  400   ; Output character interrupt timer
12 000000 M      $IITIM  200   ; Input character interrupt timer
13 000000 M      $VTESE  100   ; Activate on VT50 esc-letter sequence
14 000000 M      $CHACT  40    ; Enable single-character activation
15 000000 M      $NOWTT  20    ; Allow non-wait TT input (.TTINR)
16 000000 M      $KED    10    ; KED or K52 is default system editor
17 000000 M      $INDDF  4     ; Execute IND by default for @ files
18 000000 M      $INDRN  2     ; IND program is currently executing
19 000000 M      $CARMN  1     ; Monitor carrier and log off if carrier lost
20
21 ; LSW5 flags that are inherited by subprocesses:
22 ;
23 000000 M      ISPF5   $TECO!$WILD!$KED!$INDDF

```

```

1      ;-----
2      ; Status flags in LSW6 table.
3      ;
4 000000 M      $SNWTT 100000 ;SET TT NOWAIT done
5 000000 M      $1STLG 40000  ;First Logon on line has occurred
6 000000 M      $IDMAP 20000  ;Map user par7 to I/O page
7 000000 M      $XSTOP 10000  ;We have sent X-off to stop transmission to us
8 000000 M      $MLOCK 4000   ;Job is currently locked in memory
9 000000 M      $NLOCK 2000   ;Job is waiting to be locked in low memory
10 000000 M      $CFABT 1000   ;Abort all open command files (double ctrl-c typed)
11 000000 M      $DBGMD 400    ;Debugger is in control of terminal
12 000000 M      $NOLF  200    ;Suppress auto echo of lf following cr
13 000000 M      $CFKIL 100    ;Abort IND and any nested command files
14 000000 M      $EMTTR 40     ;Trace EMT calls (SET EMT TRACE)
15 000000 M      $INDDW 20     ;IND data segment has been written to INDTMP file
16 000000 M      $DUPRN 10     ;DUP is running
17 000000 M      $WDISP 4      ;Need to redisplay current window for job
18 000000 M      $DIBOL 2      ;DIBOL is default compiler rather than DBL
19 000000 M      $STSNG 1      ;SET TT SINGLE (default to single char activation)
20      ;
21      ; LSW6 flags that are inherited by subprocesses
22      ;
23 000000 M      ISPF6  $SNWTT!$DIBOL!$STSNG

```

```

1
2      ; -----
3      ; Status flags in LSW7
4      ;
5      ;
6      ;
7      ;
8      ;
9      ;
10     ;
11     ;
12     ;
13     ;
14     ;
15     ;
16     ;
17     ;
18     ;
19     ;
20     ;
21     ; LSW7 flags that are inherited by subprocesses
22     ;
23     ;
24     ;

```

4 000000	M	\$INDAB	1	; Abort IND command files if errors occur
5 000000	M	\$TTGAG	2	; SET TT QUIET done (suppress SEND messages)
6 000000	M	\$MAPDK	4	; Memory mapping info in context block is valid
7 000000	M	\$SOTFN	10	; Set job scheduling for output buffer low or empty
8 000000	M	\$UKMRN	20	; User command processor is running
9 000000	M	\$UCLRN	40	; TSXUCL program is running
10 000000	M	\$UCLCF	100	; SET UCL FIRST
11 000000	M	\$UCLCM	200	; SET UCL MIDDLE
12 000000	M	\$UCLCL	400	; SET UCL LAST
13 000000	M	\$SLON	1000	; Enable Single Line Editor is enabled for this line
14 000000	M	\$SLTTY	2000	; Enable SL to operate on .TTYIN input
15 000000	M	\$SLLET	4000	; Enable LET option for SL
16 000000	M	\$SLKED	10000	; Enable KED option for SL
17 000000	M	\$SLINI	20000	; SL has been initialized for current edit line
18 000000	M	\$UKMON	40000	; User command processor enabled
19 000000	M	\$NDINT	100000	; Do not schedule this job as interactive
23 000000	M	ISPF7	\$SLON!\$SLKED!\$SLTTY!\$SLLET!\$INDAB!\$TTGAG!\$UCLCF!\$UCLCM!\$UCLCL	
24 000000	M	ISPF7	ISPF7!\$UKMON	

```
1 ;-----  
2 ; Status flags in LSWB  
3 ;  
4 000000 M      $SQQ0  1      ; Signal terminal if QUANO exceeded  
5 000000 M      $SQQ1  2      ; Signal terminal if QUAN1 exceeded  
6 000000 M      $SQQ1A  4      ; Signal terminal if QUAN1A exceeded  
7 000000 M      $SQQ1B 10      ; Signal terminal if QUAN1B exceeded  
8 000000 M      $SQQ1C 20      ; Signal terminal if QUAN1C exceeded  
9 000000 M      $SQQ2  40      ; Signal terminal if QUAN2 exceeded  
10 000000 M      $SQQ3 100      ; Signal terminal if QUAN3 exceeded  
11 000000 M      $SQIID 1000    ; Signal terminal if INTIDC exceeded  
12 000000 M      $SQHID 2000    ; Signal terminal if HIPRCT exceeded  
13 ;  
14 ; All signal flags cleared by SET SIGNAL OFF  
15 ;  
16 000000 M      $SGALL  $SQQ0!$SQQ1!$SQQ1A!$SQQ1B!$SQQ1C!$SQQ2!$SQQ3!$SQIID!$SQHID
```

1				
2				
3				
4	000000	M	\$SUCF 1	; Processing a startup command file
5	000000	M	\$LOFCF 2	; Processing a logoff command file
6	000000	M	\$VIRJB 4	; Virtual job -- Don't map UPAR7 to RMON
7	000000	M	\$DEBUG 10	; Program is being run with the debugger
8	000000	M	\$DBGBK 20	; Force a debugger breakpoint (user typed ctrl-D)
9	000000	M	\$DBKMN 40	; Use debugger on TSKMON
10	000000	M	\$CTRLD 100	; Allow ctrl-D to enter debugger
11	000000	M	\$RNIDP 200	; Map PAR7 to I/O page for program being run
12	000000	M	\$NABRS 400	; Need to reset speed for autobaud detection
13	000000	M	\$NTGCC 1000	; Deferred control-C for non-terminating .GTLIN
14	000000	M	\$RTCS 2000	; Receiving terminal control sequence
15	000000	M	\$NOUCR 4000	; Do not execute any user-mode completion routines
16	000000	M	\$NOABT 10000	; Do not allow job aborts
17	000000	M	\$RNMLK 20000	; Lock program being started in low memory
18	000000	M	\$SETRN 40000	; SETUP program is running
19	000000	M	\$VBELL 100000	; Bell has been rung to signal virtual wait condition
20				
21				
22				
23	000000	M	ISPF9 \$CTRLD	

1					
2					
3					
4	000000	M	\$SXOFF	2	; Tell transmitter interrupt rtn to send an XOFF
5	000000	M	\$SXON	4	; Tell transmitter interrupt rtn to send an XON
6	000000	M	\$HISTP	10	; XOFF has been transmitted due to input silo full
7	000000	M	\$NDICP	20	; Need to do input character processing for this line
8	000000	M	\$DNICP	40	; Did input char processing during this clock cycle
9	000000	M	\$ICPFK	100	; An input character processing fork is active
10	000000	M	\$RBRK	200	; Received break (framing error)
11	000000	M	\$DHXOF	400	; Force XOFF transmission to DH11 line
12	000000	M	\$DHXON	1000	; Force XON transmission to DH11 line
13	000000	M	\$DHBF1	2000	; DH11 DMA buffer 1 is ready for transmission
14	000000	M	\$DHBF2	4000	; DH11 DMA buffer 2 is currently being transmitted
15	000000	M	\$DHCDQ	10000	; DH line needs clock driven output processing

```
1 ;-----  
2 ; Status flags in LSW11  
3 ;  
4 000000 M $PWKEY 1 ;Enable keyboard control character to print windows  
5 000000 M $NOWIN 2 ;Suppress process windowing  
6 000000 M $V52EM 4 ;VT52 emulation mode (flag set in primary line entry)  
7 000000 M $TDEAD 10 ;Flag line as dead after buffer allocation  
8 000000 M $GEMAR 20 ;Getting EMT argument block, return 0 if mem trap  
9 000000 M $RCLRV 40 ;Recall SL commands in reverse order (oldest 1st)  
10 000000 M $RDSAV 100000 ;In process of reading SAV file to start program  
11 ;  
12 ; LSW11 flags that are inherited by subprocesses  
13 ;  
14 000000 M ISPF11 $PWKEY!$RCLRV
```



```

1          .SBTTL  Privilege Flags
2          ;-----
3          ; The following flags grant privileges to jobs.
4          ; Note, if any privileges are changed, the following items must also
5          ; be changed:
6          ;
7          ; 1. The table of privilege keywords in TSKMN3.
8          ; 2. The TSAUTH program.
9          ; 3. The LOGON program.
10         ;
11         ; Flags in the first privilege word.
12         ;
13 000000  M      PO$NEW 100000 ;New privilege control has been set for job
14 000000  M      PO$ALC 40000  ;ALLOCATE - Allocate devices
15 000000  M      PO$DBG 20000  ;DEBUG - Enable use of debugging facilities
16 000000  M      PO$DET 10000  ;DETACH - Allow use of detached jobs
17 000000  M      PO$SPF 4000   ;SPFUN - Allow use of .SPFUN to dir-structured dev
18 000000  M      PO$MEM 2000   ;MEMMAP - Access any area of physical memory
19 000000  M      PO$BYP 1000   ;BYPASS - Bypass all file access restrictions
20 000000  M      PO$OPR 400    ;OPER - Provide operator privileges
21 000000  M      PO$LOK 200    ;PSWAPM - Allow job to be locked in memory
22 000000  M      PO$RT 100     ;REALTIME - Allow use of real-time facilities
23 000000  M      PO$SND 40     ;SEND - Allow use of SEND command and EMT
24 000000  M      PO$NAM 20     ;SETNAME - Allow job to change name and password
25 000000  M      PO$SPV 10     ;SETPRV - Allow job to change any privilege
26 000000  M      PO$NFR 4      ;NFSREAD - Non-file-structured open for read access
27 000000  M      PO$NFW 2      ;NFSWRITE - Non-file-structured open for write access
28 000000  M      PO$SYS 1      ;SYSPRV - Allow system manager privilege
29         ;
30         ; Privilege flags in second privilege control word
31         ;
32 000000  M      P2$TRM 100000 ;TERMINAL - Allow terminal control
33 000000  M      P2$WRL 40000  ;WORLD - Provide access to any other job
34 000000  M      P2$GRP 20000  ;GROUP - Access other jobs with same project number
35 000000  M      P2$SAM 10000  ;SAME - Access other jobs with same PPN
36 000000  M      P2$VIR 4000   ;VIRTUAL - Allow use of virtual jobs
37 000000  M      P2$MSG 2000   ;MESSAGE - Use message communication facility
38 000000  M      P2$RLK 1000   ;RLOCK - Allow use of record-locking facility
39 000000  M      P2$CGR 400    ;SYSQBL - Allow job to use global regions
40 000000  M      P2$CXT 200    ;GETCXT - Copy file context from another job
41 000000  M      P2$UP4 10     ;User privilege 4
42 000000  M      P2$UP3 4      ;User privilege 3
43 000000  M      P2$UP2 2      ;User privilege 2
44 000000  M      P2$UP1 1      ;User privilege 1
45         ;
46         ; Number of words to hold privilege flags
47         ;
48 000000  M      PVNPW 2       ;Number of words for privilege flags
49         ;
50         ; Standard privileges for users without operator privilege
51         ;
52 000000  M      PO$$NP PO$NEW!PO$ALC!PO$DBG!PO$DET!PO$SPF!PO$SND!PO$NAM!PO$NFR!PO$NFW
53 000000  M      P2$$NP P2$SAM!P2$VIR!P2$MSG!P2$RLK

```



```

1          .SBTTL  Job Execution States
2          ;-----
3          ; A logged on job will always be in one of the following states.
4          ; The state of a job is stored in the LSTATE table.
5          ; The lower the state number, the higher the priority.
6          ;
7 000000   RZ
8 000000   RB      S$DUMMY  1      ; Dummy entry to make first entry value = 1
                                S$DUMMY=$AC
9          ;
10         ; Real-time high-priority state
11         ;
12 000000   RB      S$RT      1      ; Real time priority (PRIHI - 99)
                                S$RT=$AC
13 000000   RB      S$$RT     1      ; End of Real-time priority states
                                S$$RT=$AC
14         ;
15         ; High-priority states
16         ;
17 000000   RB      S$TTSC    1      ; TT input done and doing single character activation
                                S$TTSC=$AC
18 000000   RB      S$TTFN    1      ; TT input done (activation character received)
                                S$TTFN=$AC
19 000000   RB      S$OTFN    1      ; TT output buffer empty
                                S$OTFN=$AC
20 000000   RB      S$HICP    1      ; Interactive job computation
                                S$HICP=$AC
21 000000   RB      S$TWFN    1      ; Timed wait completion (.TWAIT or .MRKT)
                                S$TWFN=$AC
22 000000   RB      S$OTLO    1      ; Output buffer low
                                S$OTLO=$AC
23 000000   RB      S$IOFN    1      ; I/O Completed
                                S$IOFN=$AC
24 000000   RB      S$$HIP    1      ; End of high-priority states
                                S$$HIP=$AC
25         ;
26         ; Compute-bound states
27         ;
28 000000   RB      S$CPU      1      ; CPU-bound job
                                S$CPU=$AC
29 000000   RB      S$LOW      1      ; Low priority computation (priority 0 - PRILOW)
                                S$LOW=$AC
30 000000   RB      S$$RUN     1      ; Lowest priority of jobs that want to run
                                S$$RUN=$AC
31         ;
32         ; Wait states
33         ;
34 000000   RB      S$NEDQ     1      ; Waiting for I/O queue element
                                S$NEDQ=$AC
35 000000   RB      S$QMIO     1      ; Waiting to do a mapped I/O operation
                                S$QMIO=$AC
36 000000   RB      S$QCCB     1      ; Waiting for data cache control block
                                S$QCCB=$AC
37 000000   RB      S$QCXB     1      ; Waiting for access to job context block buffer
                                S$QCXB=$AC
38 000000   RB      S$QUSR     1      ; Waiting for access to USR data base
                                S$QUSR=$AC

```

39 000000	000023	RB	S\$IDWT	1	;Waiting for I/O to finish	S\$IDWT=\$AC
40 000000	000024	RB	S\$OTWT	1	;Waiting for TT output buffer space	S\$OTWT=\$AC
41 000000	000025	RB	S\$SFWT	1	;Waiting for locked block in shared file	S\$SFWT=\$AC
42 000000	000026	RB	S\$WSMB	1	;Waiting for free system message buffer	S\$WSMB=\$AC
43 000000	000027	RB	S\$SPDB	1	;Waiting for spool file disk space	S\$SPDB=\$AC
44 000000	000030	RB	S\$INWT	1	;Waiting for input from TT	S\$INWT=\$AC
45 000000	000031	RB	S\$QSPD	1	;Waiting for access to special device data base	S\$QSPD=\$AC
46 000000	000032	RB	S\$SPCB	1	;Waiting for spool file control block	S\$SPCB=\$AC
47 000000	000033	RB	S\$MSWT	1	;Waiting for inter-job message	S\$MSWT=\$AC
48 000000	000034	RB	S\$SPND	1	;.SPND done -- Waiting for .RSUM to restart job	S\$SPND=\$AC
49 000000	000035	RB	S\$TMWT	1	;Waiting for timed interval to pass	S\$TMWT=\$AC
50 000000	000036	RB	S\$WFM	1	;Waiting for memory expansion	S\$WFM=\$AC

		.SBTTL System message buffers			
1		-----			
2					
3		; Format of a system message buffer block.			
4					
5	000000	RZ			
6	000000	RW	SB\$LNK	1	;Link to next message block
	000000				SB\$LNK=\$AC
7	000000	RW	SB\$PNT	1	;Pointer to next character in buffer
	000002				SB\$PNT=\$AC
8	000000	RW	SB\$TXT	44.	;Storage area for message text
	000004				SB\$TXT=\$AC
9	000000	RW	SB\$END	0	;End of text storage area
	000134				SB\$END=\$AC
10	000000	RW	SB\$\$SZ	0	;Size of system message block
	000134				SB\$\$SZ=\$AC

```

1          .SBTTL    Configuration and system generation words
2          ;-----
3          ;    System configuration word (300)
4          ;
5 000000    M        CW$FB    1        ; FB monitor
6 000000    M        CW$GDH   4        ; Graphics display hardware (VT-11 or VS-60)
7 000000    M        CW$BTH   10       ; Batch is in control
8 000000    M        CW$SLE   20       ; Single-line editor active
9 000000    M        CW$50H   40       ; 50-Hz clock
10 000000   M        CW$FPU   100       ; Floating point unit installed
11 000000   M        CW$FGJ   200       ; Foreground job active
12 000000   M        CW$LGS   400       ; Linked with graphics scroller
13 000000   M        CW$USR   1000       ; USR permanently resident
14 000000   M        CW$LSI   4000       ; CPU is LSI-11
15 000000   M        CW$XM    10000      ; XM Monitor
16 000000   M        CW$CSR   20000      ; System clock has a status register
17 000000   M        CW$KWP   40000      ; KWIIP clock exists
18 000000   M        CW$LPC   100000     ; System has a clock
19          ;
20          ;    Extended configuration word (370)
21          ;
22 000000   M        CW$CSH    1        ; Cache memory present
23 000000   M        CW$PAR    2        ; Parity memory installed
24 000000   M        CW$RSR    4        ; Readable switch register
25 000000   M        CW$WCD   10        ; Writable console display
26 000000   M        CW$RLH   20        ; A handler has been released
27 000000   M        CW$ESP   40        ; Set exit noswap
28 000000   M        CW$QBS   100       ; Running on Q-bus system
29 000000   M        CW$CIS   200       ; Hardware has CIS instructions
30 000000   M        CW$EIS   400       ; Hardware has EIS instructions
31 000000   M        CW$V60   1000       ; VT60 display exists
32 000000   M        CW$PRD   20000      ; Running on PRD-xxx "personal" computer
33 000000   M        CW$70    40000      ; Processor is 11/70
34 000000   M        CW$60    100000     ; Processor is 11/60
35          ;
36          ;    System generation flag word (372)
37          ;
38 000000   M        SG$ELG    1        ; Error logging is present
39 000000   M        SG$MMU    2        ; Memory management option is present
40 000000   M        SG$IOT    4        ; I/O timeout support is present
41 000000   M        SG$EMT   10        ; Running under RT-11 emulator
42 000000   M        SG$PAR   1000       ; Memory parity option is present
43 000000   M        SG$MTM   2000       ; SJ mark-time option present
44 000000   M        SG$MTS   20000      ; Multi-terminal option present
45 000000   M        SG$SYJ   40000      ; System job support present
46 000000   M        SG$TSX   100000     ; Running under TSX-Plus
47          ;
48          ;    Spooler control flags stored in SPSTAT (RMON offset 414)
49          ;
50 000000   M        SS$RUN    200       ; Spooler is running
51 000000   M        SS$PRT   10000      ; Print screen from PI handler
52          ;
53          ;    IND control flags stored in INDSTA (RMON offset 417)
54          ;
55 000000   M        IN$ACT    200       ; IND is active and should be recalled on command exit
56 000000   M        IN$CNT    100       ; IND is being reentered by KMON to continue processing
57 000000   M        IN$CMD    40        ; IND is passing a command to KMON for execution

```

```
58      ;  
59      ; Command file control word flags CFSTS (RMON offset 366)  
60      ;  
61 000000 M      CFACFL 110400 ;Input is coming from a command file  
62      ;  
63      ; RT-11 TT option word (word preceding word pointed to by location 30)  
64      ;  
65 000000 M      TO$TAB 1      ;Hardware tab supported by terminal  
66 000000 M      TO$FF  4      ;Hardware form-feed supported by terminal  
67 000000 M      TO$LC  40000 ;Allow lower-case character input  
68 000000 M      TO$SCP 100000 ;Do scope type processing of rub-out characters  
69      ;  
70      ; RT-11 command file option flags in CFSTS (RMON offset 366)  
71      ;  
72 000000 M      CF$IND  4      ;Command files are to be processed by IND  
73 000000 M      CF$QUT  2000   ;Set TT quiet has been done  
74      ;  
75      ; RT-11 offset from start of RMON to EMT dispatch table (RMON offset 316)  
76      ;  
77      ; This word is now reserved for use by DBL.  It must be zeroed at  
78      ; job initialization and not used by the operating system.  
79      ;
```

```

1          .SBTTL Overlay system region offsets
2          ;-----
3          ; Format of the overlay table $OVTAB.
4          ;
5 000000   RZ
6 000000   RW      O. ADR    1      ; Overlay virtual address
                                O. ADR=$AC
7 000000   RW      O. PAR    0      ; Overlay par5 value (initialized when loaded)
                                O. PAR=$AC
8 000000   RW      O. BLK    1      ; Overlay starting disk block number
                                O. BLK=$AC
9 000000   RW      O. SIZ    1      ; Overlay size in number of words
                                O. SIZ=$AC
10
11          .SBTTL Virtual overlay handler offsets
12          ;-----
13          ; The following offsets are relative to a jobs overlay table.
14          ; The address of the overlay table is stored in location 64 of the SAV file.
15          ;
16 000000   M      VO$WDE  -16      ; Pointer past end of PLAS Window Definition Blocks
17 000000   M      VO$WDB  -14      ; Pointer to first PLAS Window Definition Block
18 000000   M      VO$RDB  -12      ; Pointer to Region Definition Block
19 000000   M      VO$HIR   -4      ; Address beyond end of root segment
20 000000   M      VO$HIO  -2      ; Address beyond end of /O overlays

```

1		.SBTTL	I/O System Tables	
2				
3			Format of an I/O Channel.	
4				
5	000000	RZ		
6	000000	RW	C. CSW 1	; Channel Status Word (CSW) C. CSW=\$AC
7	000000	RW	C. SBLK 1	; Starting block number of file C. SBLK=\$AC
8	000000	RW	C. LENG 1	; Number of blocks allocated for file C. LENG=\$AC
9	000000	RW	C. USED 1	; Highest block number written to file C. USED=\$AC
10	000000	RB	C. NUMQ 1	; Number of I/O requests pending for channel C. NUMQ=\$AC
11	000000	RB	C. DEVQ 1	; Device unit number C. DEVQ=\$AC
12	000000	RW	CHNSIZ 0	; Size of I/O channel block CHNSIZ=\$AC
13				
14			Flags in the Channel Status Word (C. CSW).	
15				
16	000000	M	CS\$OPN 100000	; Channel is open
17	000000	M	CS\$RON 40000	; Read-only access
18	000000	M	CS\$EOF 20000	; End of file hit
19	000000	M	CS\$SEG 17400	; Directory segment number containing entry for file
20	000000	M	CS\$ENT 200	; Channel was opened with a .ENTER
21	000000	M	CS\$SPL 100	; Channel is opened to spooled device
22	000000	M	CS\$NMX 76	; Device table index
23	000000	M	CS\$ERR 1	; Hard error occurred


```

1          .SBTTL  I/O queue entry
2          ;-----
3          ; Format of an I/O queue entry.
4          ;
5 000000   RZ
6 000000   RW      Q. LINK  1      ; Flink to next Queue entry
                                Q. LINK=$AC
7 000000   RW      Q. CSW   1      ; Address of CSW for channel making request
                                Q. CSW=$AC
8 000000   RW      Q. BLKN  1      ; Physical block number of request
                                Q. BLKN=$AC
9 000000   RB      Q. FUNC  1      ; Special function code
                                Q. FUNC=$AC
10 000000  RB      Q. UNIT  1      ; Device unit number
                                Q. UNIT=$AC
11 000000  M       Q. JNUM  Q. UNIT ; Job number issuing request
12 000000  RW      Q. BUFF  1      ; User buffer address relative to Q. PAR
                                Q. BUFF=$AC
13 000000  RW      Q. WCNT  1      ; Word count (+=>Read, 0=>Seek, -=>Write)
                                Q. WCNT=$AC
14 000000  RW      Q. COMP  1      ; Address of completion routine for request
                                Q. COMP=$AC
15 000000  RW      Q. PAR   1      ; PAR relocation bias for buffer address
                                Q. PAR=$AC
16 000000  RW      Q. PA5   1      ; Mapping value for kernel PAR 5
                                Q. PA5=$AC
17 000000  RW      Q. UMRX  1      ; Address of unibus map register block assigned for I/O
                                Q. UMRX=$AC
18 000000  RW      Q. CHAN  1      ; User channel # associated with I/O request
                                Q. CHAN=$AC
19 000000  RB      Q. DEVX  1      ; Device index number
                                Q. DEVX=$AC
20 000000  RB      Q. FLAG  1      ; QF$xxx control flags (see below)
                                Q. FLAG=$AC
21 000000  RB      Q. JOB   1      ; Number of job that is making request
                                Q. JOB=$AC
22 000000  RB      Q. UMBV  1      ; Unibus base register number
                                Q. UMBV=$AC
23 000000  RW      Q. UMPB  1      ; Original value of Q. BUFF when I/O was initiated
                                Q. UMPB=$AC
24 000000  RW      Q. UMPP  1      ; Original value of Q. PAR when I/O was initiated
                                Q. UMPP=$AC
25 000000  RW      Q. PA6   1      ; Mapping for job's context block
                                Q. PA6=$AC
26 000000  RW      Q. UCSW  1      ; Address of user's channel block
                                Q. UCSW=$AC
27 000000  RW      Q. ICSW  5      ; Internal channel block passed with Q elem to handlers
                                Q. ICSW=$AC
28 000000  RW      IQQSIZ  0      ; Size of I/O queue entry
                                IQQSIZ=$AC

```



```

1          .SBTTL  Completion queue entry
2          ;-----
3          ; Format of a completion queue element.
4          ; Note: A completion queue entry is allocated the same number of words
5          ;       as an I/O queue entry and the fields in a completion queue entry
6          ;       are mapped over the fields in an I/O queue entry.
7          ;
8 000000   M      CQ$LNK  Q.LINK  ;Link to next queue entry
9 000000   M      CQ$HOT  Q.CSW   ;High-order time for timer queue elements
10 000000  M      CQ$LOT  Q.BLKN  ;Low-order time for timer queue elements
11 000000  M      CQ$JOB  Q.FUNC  ;Job number (byte)
12 000000  M      CQ$RNS  Q.UNIT  ;Execution state for completion routine (byte)
13 000000  M      CQ$RO   Q.BUFF  ;Value to pass in R0 when compl routine called
14 000000  M      CQ$R1   Q.WCNT  ;Value to pass in R1 when compl routine called
15 000000  M      CQ$RTN  Q.COMP  ;Address of completion routine
16 000000  M      CQ$PA5  Q.PA5   ;Value for kernel PAR 5 mapping for compl routine
17 000000  M      CQ$FLG  Q.FLAG  ;QF$xxx control flags (byte) (see below)
18 000000  M      CQ$PRI  Q.DEVX  ;Execution priority for completion routine (byte)
19 000000  M      CQ$CP   Q.UMRX  ;Completion routine class priority (see CP$xxx below)
20
21          ;-----
22          ; Control flags stored in the Q.FLAG cell of I/O queue elements and the
23          ; CQ$FLG cell of completion routine queue elements.
24          ;
25 000000  M      QF$SCR   1       ;Completion routine is for system - run in kernel mode
26 000000  M      QF$MIO   2       ;Secondary I/O operation for mapped I/O
27 000000  M      QF$CID   4       ;Secondary I/O operation for data caching operation
28 000000  M      QF$QWC  10      ; (For mapped I/O) Use original word count
29 000000  M      QF$IOT  20      ;Completion routine for handler .TIMIO request
30 000000  M      QF$SYN  40      ;Completion routine for .SYNCH

```

1				.SBTTL I/O timer requests	
2				-----	
3				; Format of a queue element used to store a device handler i/o	
4				; timeout request.	
5					
6	000000		RZ		
7	000000		RW	IT\$HOT 1 ;High-order time value	IT\$HOT=\$AC
8	000000	000000	RW	IT\$LOT 1 ;Low-order time value	IT\$LOT=\$AC
9	000000	000002	RW	IT\$LNK 1 ;Link to next timer queue element	IT\$LNK=\$AC
10	000000	000004	RW	IT\$JOB 1 ;Number of job to synch with	IT\$JOB=\$AC
11	000000	000006	RW	IT\$SEQ 1 ;Request sequence number	IT\$SEQ=\$AC
12	000000	000010	RW	IT\$SYS 1 ;Not used	IT\$SYS=\$AC
13	000000	000012	RW	IT\$RTN 1 ;Address of completion routine	IT\$RTN=\$AC
		000014			

		.SBTTL Swapper Command Packets	
1		;-----	
2		; The following packets are used to pass special function commands	
3		; to the job swapper.	
4		;	
5		;	
6	000000	RZ	
7	000000	RW	SP\$LNK 1 ;Link to next packet on list
	000000		SP\$LNK=\$AC
8	000000	RB	SP\$CMD 1 ;Command value (see SA\$xxx below)
	000002		SP\$CMD=\$AC
9	000000	RB	SP\$JOB 1 ;Number of associated job
	000003		SP\$JOB=\$AC
10	000000	RW	SP\$DW1 1 ;Data word 1
	000004		SP\$DW1=\$AC
11	000000	RW	SP\$\$SZ 0 ;Size of a command packet
	000006		SP\$\$SZ=\$AC
12		;	
13		; Command values that can be passed to the swapper (must be even values)	
14		;	
15	000000	M	SA\$LOK 0 ;Lock a job in low memory
16	000000	M	SA\$RGN 2 ;Allocate memory for a shared global PLAS region


```

1          .SBTTL  Job monitoring control block
2          ;-----
3          ; The following control blocks are used to implement the facility
4          ; that allows one TSX-Plus job to monitor the status of another job.
5          ;
6 000000   RZ
7 000000   RW      JM$LNK  1      ;Link to next control block          JM$LNK=$AC
                                000000
8 000000   RW      JM$RTN  1      ;Address of completion routine      JM$RTN=$AC
                                000002
9 000000   RB      JM$JOB  1      ;Job index # of monitoring job      JM$JOB=$AC
                                000004
10 000000  RW      JM$$SZ  0      ;Size of control block           JM$$SZ=$AC
                                000006
11          ;
12          ; Status code values generated by the system to indicate job state changes
13          ;
14 000000   M      JS$ON  1      ;Job initialization
15 000000   M      JS$LOG  2      ;Job logged on with LOGON program
16 000000   M      JS$RUN  3      ;Began execution of a program
17 000000   M      JS$KMN  4      ;Program exited to TSKMON
18 000000   M      JS$OFF  5      ;Job logged off

```

```

1          .SBTTL  Process window control block
2          ;-----
3          ; Format of a process window control block.
4          ;
5 000000   M      TCSBSZ  12.      ;Max number of chars in a terminal control sequence
6          ;
7 000000   RZ
8 000000   RB      DW$JOB  1        ;Job number of owner job
          000000                               DW$JOB=$AC
9 000000   RB      DW$ID   1        ;Window id number
          000001                               DW$ID=$AC
10 000000  RW      DW$LIN  1        ;Current line number
          000002                               DW$LIN=$AC
11 000000  RW      DW$COL  1        ;Current column number
          000004                               DW$COL=$AC
12 000000  RW      DW$LPP  1        ;Number of lines per page
          000006                               DW$LPP=$AC
13 000000  RW      DW$CPL  1        ;Number of columns per line
          000010                               DW$CPL=$AC
14 000000  RW      DW$TLN  1        ;Number of line at base of buffer
          000012                               DW$TLN=$AC
15 000000  RW      DW$SRT  1        ;Top line of scrolling region
          000014                               DW$SRT=$AC
16 000000  RW      DW$SRB  1        ;Bottom line of scrolling region
          000016                               DW$SRB=$AC
17 000000  RW      DW$AW   1        ;Control flags (see AW$xxx flags below)
          000020                               DW$AW=$AC
18 000000  RW      DW$CSP  1        ;Pointer to next char position in DW$CCB
          000022                               DW$CSP=$AC
19 000000  RW      DW$CSR  1        ;Address of processing routine for next char
          000024                               DW$CSR=$AC
20 000000  RW      DW$LPT  1        ;Pointer to 1st char of current line
          000026                               DW$LPT=$AC
21 000000  RW      DW$RID  1        ;Address of region control block
          000030                               DW$RID=$AC
22 000000  RW      DW$MAP  1        ;Value to map PAR to region
          000032                               DW$MAP=$AC
23 000000  RB      DW$SLN  1        ;Saved line number
          000034                               DW$SLN=$AC
24 000000  RB      DW$SCL  1        ;Saved column number
          000035                               DW$SCL=$AC
25 000000  RB      DW$SCA  1        ;Saved character attributes
          000036                               DW$SCA=$AC
26 000000  RB      DW$MSL  1        ;Max # lines that may scroll while detached from term
          000037                               DW$MSL=$AC
27 000000  RB      DW$NSL  1        ;Number of lines which have scrolled since detached
          000040                               DW$NSL=$AC
28 000000  RB      DW$CCA  1        ;Current character attribute flags
          000041                               DW$CCA=$AC
29 000000  RB      DW$GOM  1        ;Current designation for G0 characters *keep together*
          000042                               DW$GOM=$AC
30 000000  RB      DW$G1M  1        ;Current designation for G1 characters *keep together*
          000043                               DW$G1M=$AC
31 000000  RB      DW$G2M  1        ;Current designation for G2 characters *keep together*
          000044                               DW$G2M=$AC
32 000000  RB      DW$G3M  1        ;Current designation for G3 characters *keep together*
          000045                               DW$G3M=$AC

```

33 000000	000046	RB	DW\$GLM	1	; Current mapping for GL	DW\$GLM=\$AC
34 000000	000047	RB	DW\$GRM	1	; Current mapping for GR	DW\$GRM=\$AC
35 000000	000050	RB	DW\$GLS	1	; Mapping for next GL char if doing single shifting	DW\$GLS=\$AC
36 000000	000051	RB	DW\$CSB	TCSBSZ	; Control sequence buffer	DW\$CSB=\$AC
37						
38 000000	000066	RW	DW\$\$SZ	0	; Size of control block	DW\$\$SZ=\$AC

1				
2				
3				
4	000000	M	AL\$DHB 1	;Bottom portion of a double-high line
5	000000	M	AL\$DHT 2	;Top portion of a double-high line
6	000000	M	AL\$DWD 4	;Double wide line
7				
8				
9				
10	000000	M	AC\$SET 3	;Character set for this character
11	000000	M	AC\$BLD 20	;Bold
12	000000	M	AC\$BLK 40	;Blinking
13	000000	M	AC\$REV 100	;Reverse video
14	000000	M	AC\$ULN 200	;Underlined
15				
16				
17				
18	000000	M	AW\$52 1	;VT52 terminal mode
19	000000	M	AW\$200 2	;VT200 terminal mode
20	000000	M	AW\$132 4	;132 column mode
21	000000	M	AW\$INS 10	;Insert mode
22	000000	M	AW\$ACK 20	;Application mode for cursor keys
23	000000	M	AW\$REV 40	;Reverse video (dark chars on light background)
24	000000	M	AW\$ORS 100	;Origin relative to scroll region
25	000000	M	AW\$AKM 200	;Application keypad mode
26	000000	M	AW\$VCR 400	;Make cursor visible
27	000000	M	AW\$SS 1000	;Single shift for next character
28	000000	M	AW\$S52 2000	;Simulate VT52 mode (term is actually VT100/200)
29	000000	M	AW\$RPT 4000	;Automatic keypad repeat is on
30	000000	M	AW\$PRT 10000	;Printer port has been selected (suspend windowing)
31	000000	M	AW\$SPN 20000	;Suspend window data processing
32	000000	M	AW\$DDC 40000	;Don't pass characters through to terminal
33	000000	M	AW\$PRM 100000	;Window is permanent


```
1          .SBTTL  Printer attribute flags
2          ;-----
3          ; The following flags define printer attributes used to pass information
4          ; from the SET PRINTWINDOW Kmon command to the WINPRT program.
5          ;
6 000000 M      PA$GRC  1      ; Can print graphics (line drawing) character set
7 000000 M      PA$UKC  2      ; Can print U.K. national character set
8 000000 M      PA$DSC  4      ; Can print Dec supplemental character set
9 000000 M      PA$BLD  10     ; Can print bold characters
10 000000 M     PA$ULN  20     ; Can underline characters
11 000000 M     PA$DWD  40     ; Can print double-wide characters
12 000000 M     PA$HQL  100    ; Can print in both draft and letter quality modes
13 000000 M     PA$LET  200    ; Select letter-quality mode
14 000000 M     PA$BEL  400    ; Ring bell when window data queued for printing
15 000000 M     PA$NWD  1000   ; Suppress printer width control (/NOWIDTH)
16 000000 M     PA$DTS  2000   ; Date/time stamp printed window
```

```

1          .SBTTL  Key definition blocks
2          ;-----
3          ; Information related to user-defined keys (i.e., DEFINE/KEY ...)
4          ;
5          ; Maximum number of characters in key definition string (including null).
6          ;
7 000000 M      KEYMXT  65.      ;Max # chars in key definition
8          ;
9          ; Control block used to hold each key definition.
10         ;
11 000000 RZ
12 000000 RB      KD$COD  1      ;Key code (KC$xxx)
13         000000                                KD$COD=$AC
14 000000 RB      KD$TYP  1      ;Key type (KT$xxx)
15         000001                                KD$TYP=$AC
16 000000 RB      KD$FLG  1      ;KF$xxx flags (see below)
17         000002                                KD$FLG=$AC
18 000000 RB      KD$TXT  KEYMXT ;Key definition text string (asciz)
19         000003                                KD$TXT=$AC
20 000000 RW      KD$$SZ  0      ;Size of key definition block
21         000104                                KD$$SZ=$AC
22         ;
23         ; Status flags stored in KD$FLG cell
24         ;
25 000000 M      KF$ECD  1      ;Echo the key definition when it is used
26 000000 M      KF$TRM  2      ;Terminate input line with key definition
27         ;
28         ; Key type codes
29         ;
30 000000 M      KT$NRM  1      ;Normal function key
31 000000 M      KT$GLD  2      ;Gold key pressed with function key
32 000000 M      KT$LET  3      ;Key from keyboard
33 000000 M      KT$GLT  4      ;Gold letter key
34         ;
35         ; Key codes
36         ;
37 000000 M      KC$PF1  1      ;PF1
38 000000 M      KC$PF2  2      ;PF2
39 000000 M      KC$PF3  3      ;PF3
40 000000 M      KC$PF4  4      ;PF4
41 000000 M      KC$KP0  5      ;KP0
42 000000 M      KC$KP1  6      ;KP1
43 000000 M      KC$KP2  7      ;KP2
44 000000 M      KC$KP3 10      ;KP3
45 000000 M      KC$KP4 11      ;KP4
46 000000 M      KC$KP5 12      ;KP5
47 000000 M      KC$KP6 13      ;KP6
48 000000 M      KC$KP7 14      ;KP7
49 000000 M      KC$KP8 15      ;KP8
50 000000 M      KC$KP9 16      ;KP9
51 000000 M      KC$DOT  17      ;PERIOD
52 000000 M      KC$COM  20      ;COMMA
53 000000 M      KC$MIN  21      ;MINUS
54 000000 M      KC$ENT  22      ;ENTER
55 000000 M      KC$UP   23      ;UP ARROW
56 000000 M      KC$DWN  24      ;DOWN ARROW
57 000000 M      KC$LFT  25      ;LEFT ARROW

```

53 000000	M	KC\$RIT	26	; RIGHT ARROW
54 000000	M	KC\$E1	27	; E1 - Find
55 000000	M	KC\$E2	30	; E2 - Insert here
56 000000	M	KC\$E3	31	; E3 - Remove
57 000000	M	KC\$E4	32	; E4 - Select
58 000000	M	KC\$E5	33	; E5 - Prev screen
59 000000	M	KC\$E6	34	; E6 - Next screen
60 000000	M	KC\$F6	35	; F6
61 000000	M	KC\$F7	36	; F7
62 000000	M	KC\$F8	37	; F8
63 000000	M	KC\$F9	40	; F9
64 000000	M	KC\$F10	41	; F10
65 000000	M	KC\$F11	42	; F11
66 000000	M	KC\$F12	43	; F12
67 000000	M	KC\$F13	44	; F13
68 000000	M	KC\$F14	45	; F14
69 000000	M	KC\$F15	46	; F15 - Help
70 000000	M	KC\$F16	47	; F16 - Do
71 000000	M	KC\$F17	50	; F17
72 000000	M	KC\$F18	51	; F18
73 000000	M	KC\$F19	52	; F19
74 000000	M	KC\$F20	53	; F20

```

1          .SBTTL  Synch request control block
2          ;-----
3          ; Format of a .synch request control block.
4          ;
5 000000   RZ
6 000000   RW      SN$LNK  1      ;Link to next active synch control block for job
                                SN$LNK=$AC
                                000000
7 000000   RW      SN$JOB  1      ;Number of job doing .synch request
                                SN$JOB=$AC
                                000002
8 000000   RW      SN$XX1  1      ;(unused)
                                SN$XX1=$AC
                                000004
9 000000   RW      SN$XX2  1      ;(unused)
                                SN$XX2=$AC
                                000006
10 000000  RW      SN$ID   1      ;Synch ID code passed to synch routine in RO
                                SN$ID=$AC
                                000010
11 000000  RW      SN$XX3  1      ;(unused)
                                SN$XX3=$AC
                                000012
12 000000  RW      SN$RTN  1      ;Address of synch routine to be called
                                SN$RTN=$AC
                                000014
13
14          .SBTTL  Vector control block
15          ;-----
16          ; A vector control block is used to associate an interrupt vector with
17          ; a real-time program.
18          ; Note: if the size of a vector control block is changed, TSGEN must be
19          ; changed since it allocates space for vector control blocks.
20          ;
21 000000   RZ
22 000000   RW      VC$JSR  2      ;Space for [JSR R2,RTINT] instruction
                                VC$JSR=$AC
                                000000
23 000000   RW      VC$RTN  1      ;Address of user's completion routine
                                VC$RTN=$AC
                                000004
24 000000   RB      VC$VEC  1      ;Address of interrupt vector / 2
                                VC$VEC=$AC
                                000006
25 000000   RB      VC$JOB  1      ;Number of job associated with vector
                                VC$JOB=$AC
                                000007
26 000000   RB      VC$PRI  1      ;Priority of completion routine
                                VC$PRI=$AC
                                000010
27 000000   RB      VC$FLG  1      ;VF$xxx flags (see below)
                                VC$FLG=$AC
                                000011
28 000000   RW      VC$$SZ  0      ;Size of a vector control block
                                VC$$SZ=$AC
                                000012
29
30          ; Status flags stored in VC$FLG cell
31          ;
32 000000   M      VF$DIR  1      ;Interrupt is directly connected to service routine
33 000000   M      VF$DET  2      ;Interrupt has been disconnected by .DEVICE

```

```

1          .SBTTL  TSXUCL data base definitions
2          ;-----
3          ; Define data structures in TSXUCL data base.
4          ;
5          M      UCLMKL  11.      ; Maximum length of a command keyword
6          M      UCLMCL  80.      ; Maximum length of a command string
7          ;
8          ; Command control information
9          ;
10         RZ
11         RW      UC$NDC  1          ; Number of defined commands
12                                     UC$NDC=$AC
13         RW      UC$MDC  1          ; Maximum allowed defined commands
14                                     UC$MDC=$AC
15         RW      UC$$SZ  0          ; Size of control block
16                                     UC$$SZ=$AC
17         ;
18         ; Command name template
19         ;
20         RZ
21         RB      UK$NAM  UCLMKL+1    ; Asciz keyword name string
22                                     UK$NAM=$AC
23         RW      UK$PTR  1          ; Pointer to command string descriptor
24                                     UK$PTR=$AC
25         RW      UK$$SZ  0          ; Size of keyword descriptor block
26                                     UK$$SZ=$AC
27         ;
28         ; Command string storage
29         ;
30         RZ
31         RB      US$TXT  UCLMCL      ; Asciz command string
32                                     US$TXT=$AC
33         RW      US$$SZ  0          ; Size of command descriptor
34                                     US$$SZ=$AC

```

```

1          .SBTTL  PLAS region and window descriptor blocks
2          ;-----
3          ; Region Control Block format
4          ;
5 000000   RZ
6 000000   RW      RC$LEN  1      ;Number 64-byte pages allocated for region
                                RC$LEN=$AC
                                000000
7 000000   RW      RC$PAG  1      ;Number 512-byte pages allocated for region
                                RC$PAG=$AC
                                000002
8 000000   RW      RC$BAS  1      ;Base 64-byte page number of region in memory
                                RC$BAS=$AC
                                000004
9 000000   RW      RC$BLK  1      ;Block number in swap file / ptr to global RCB
                                RC$BLK=$AC
                                000006
10 000000  RW      RC$FLG  1      ;Control flags -- See below
                                RC$FLG=$AC
                                000010
11 000000  RW      RC$NAM  2      ;Name of region
                                RC$NAM=$AC
                                000012
12 000000  RB      RC$CNT  1      ;Number of jobs attached to region
                                RC$CNT=$AC
                                000016
13 000000  RB      RC$OWN  1      ;Job index number of job that created region
                                RC$OWN=$AC
                                000017
14 000000  RW      RC$$SZ  0      ;Size of a region control block
                                RC$$SZ=$AC
                                000020
15          ;
16          ; Control flags stored in RC$FLG
17          ;
18 000000  M      RC$INM  1      ;Region is currently in memory
19 000000  M      RC$SFA  2      ;Space in swap file has been allocated for region
20 000000  M      RC$GBL  4      ;This is a named region
21 000000  M      RC$PVT 10      ;This region is private (local) to a job
22 000000  M      RC$LCG 20      ;This RCB is local copy of global RCB
23 000000  M      RC$EXC 40      ;Creating job has exclusive access to region
24 000000  M      RC$AGE 100     ;Automatically eliminate region when idle
25 000000  M      RC$EXI 200     ;Eliminate region on program exit or abort
26 000000  M      RC$OFF 400     ;Eliminate region on logoff
27 000000  M      RC$AEP 1000    ;Automatical elimination (AGE) pending next attach
28 000000  M      RC$PRM 2000    ;This is a permanent global region
29 000000  M      RC$USE 100000   ;This RCB is in use
30          ;
31          ;-----
32          ; Region Definition Block format (in user's job area)
33          ;
34 000000   RZ
35 000000   RW      R. GID  1      ;Address of region control block
                                R. GID=$AC
                                000000
36 000000   RW      R. GSIZ 1      ;Number of 64-byte pages to allocate for region
                                R. GSIZ=$AC
                                000002
37 000000   RW      R. GSTS 1      ;Status flags (see below)
                                R. GSTS=$AC
                                000004
38 000000   RW      R. NAME 2      ;Global region name (Rad50)
                                R. NAME=$AC
                                000006
39          ;
40          ; Status flags stored in R. GSTS
41          ;
42 000000  M      RS. CRR 100000   ;Region was successfully created (status)
43 000000  M      RS. UNM 40000    ;Windows were unmapped during region creation (status)
44 000000  M      RS. NAL 20000    ;Region was not previously allocated (status)

```

45 000000	M	RS. NEW	10000	;Global region created due to attach request (status)
46 000000	M	RS. GBL	4000	;Create local region within global region (w. error)
47 000000	M	RS. CGR	2000	;Create local region within global region (create)
48 000000	M	RS. AGE	1000	;Automatic global region elimination
49 000000	M	RS. EGR	400	;Eliminate global region
50 000000	M	RS. EXI	200	;Eliminate global region on exit or abort
51 000000	M	RS. PVT	1	;This is a private region


```

1      ; -----
2      ; Window Control Block
3      ;
4 000000 RZ
5 000000 RW      WC$RCB  1      ; Address of region control block (0 if not mapped)
                                WC$RCB=$AC
6 000000 000000 RW      WC$SIZ  1      ; # 64-byte pages for window (0 if not allocated)
                                WC$SIZ=$AC
7 000000 000002 RW      WC$VLO  1      ; Virtual address of window base
                                WC$VLO=$AC
8 000000 000004 RW      WC$VHI  1      ; Virtual address of window top (addr of last byte)
                                WC$VHI=$AC
9 000000 000006 RW      WC$LEN  1      ; # 64-byte pages actually mapped by window
                                WC$LEN=$AC
10 000000 000010 RW      WC$OFF  1      ; Offset into region where window base starts
                                WC$OFF=$AC
11 000000 000012 RB      WC$PAR  1      ; Base PAR index # (2 * PAR #)
                                WC$PAR=$AC
12 000000 000014 RB      WC$NPR  1      ; # PAR's affected by window
                                WC$NPR=$AC
13 000000 000015 RW      WC$$SZ  0      ; Size of a Window Control Block
                                WC$$SZ=$AC
14      ; -----
15      ; Window Definition Block (in user's job area)
16      ;
17      ;
18 000000 RZ
19 000000 RB      W. NID  1      ; Window ID
                                W. NID=$AC
20 000000 000000 RB      W. NAPR  1      ; Base PAR for window
                                W. NAPR=$AC
21 000000 000001 RW      W. NBAS  1      ; Base virtual address
                                W. NBAS=$AC
22 000000 000002 RW      W. NSIZ  1      ; # 64-byte pages for window
                                W. NSIZ=$AC
23 000000 000004 RW      W. NRID  1      ; Address of region control block
                                W. NRID=$AC
24 000000 000006 RW      W. NOFF  1      ; Offset into region where window base starts
                                W. NOFF=$AC
25 000000 000010 RW      W. NLEN  1      ; # 64-byte pages to map
                                W. NLEN=$AC
26 000000 000012 RW      W. NSTS  1      ; Status flags -- See below
                                W. NSTS=$AC
27 000000 000014 RW      W. SIZE  0      ; Size of window definition block
                                W. SIZE=$AC
28      ;
29      ; Status flags stored in W. NSTS
30      ;
31 000000 M      WS. MAP  400      ; Automatically map after .CRAW
32 000000 M      WS. ELW  20000    ; A window was eliminated
33 000000 M      WS. UNM  40000    ; A window was unmapped
34 000000 M      WS. CRW  100000   ; Window was successfully created

```


Unibus Map Register descriptor block

```

1          .SBTTL    Unibus Map Register descriptor block
2          ;-----
3          ; The following descriptor block is used to hold information about
4          ; a Unibus map register set.
5          ; Note: The format of this descriptor must match the table generated
6          ;       by the UMRDEF macro in TSGEN.
7          ;
8 000000    RZ
9 000000    RB      UM$UMR  1      ; Number of register that is at base of this set
                                UM$UMR=$AC
000000
10 000000    RB      UM$NMR  1      ; Number of registers in this set
                                UM$NMR=$AC
000001
11 000000    RW      UM$WDS  1      ; Number of words that can be handled by this set
                                UM$WDS=$AC
000002
12 000000    RW      UM$IOQ  1      ; 0==>UMR free; non-zero==>Address of associated IOQ
                                UM$IOQ=$AC
000004
13 000000    RW      UM$$SZ  0      ; Size of UMR descriptor block
                                UM$$SZ=$AC
000006

```

```

1          .SBTTL  Mapped I/O control blocks
2          ;-----
3          ; The following control blocks are used to store information about
4          ; I/O transfers that need to be mapped because the device controller
5          ; only supports 18-bit addresses and we are running on a 22-bit
6          ; Q-bus system.
7          ; The head of the free list of these control blocks is MIOBHD.
8          ; Note: If the size of this control block is changed, TSGEN must
9          ; also be changed since it allocates space for the blocks.
10         ;
11         RZ
12         RW      MI$LNK  1          ;Link to next control block          MI$LNK=$AC
13         RW      MI$OQE  1          ;Address of original I/O queue element  MI$OQE=$AC
14         RW      MI$TRW  1          ;Total number of words left to be transferred  MI$TRW=$AC
15         RW      MI$CWC  1          ;Number of words being transferred by current op.  MI$CWC=$AC
16         RW      MI$UBP  1          ;64-byte block base of user's buffer      MI$UBP=$AC
17         RW      MI$UBO  1          ;Offset within 64-byte block of base of user's buffer  MI$UBO=$AC
18         RW      MI$SBP  1          ;64-byte block base of system buffer      MI$SBP=$AC
19         RB      MI$RWF  1          ;Read/Write flag (0=Read, 1=Write)        MI$RWF=$AC
20         RB      MI$JOB  1          ;Job number doing the I/O operation      MI$JOB=$AC
21         RW      MI$$SZ  0          ;Size of a control block              MI$$SZ=$AC
22         ;
23         ;-----
24         ; Wait blocks used to queue up requests for mapped I/O operations
25         ; by the system.
26         ;
27         RZ
28         RW      MW$LNK  1          ;Link to next block              MW$LNK=$AC
29         RW      MW$IOQ  1          ;Address of pending I/O queue element  MW$IOQ=$AC
30         RW      MW$$SZ  0          ;Size of control block              MW$$SZ=$AC

```

```

1          .SBTTL  Cached I/O control block
2          ;-----
3          ; Control block used by an I/O operation that is accessing a data
4          ; cached device.
5          ;
6 000000   RZ
7 000000   RW      CC$LNK  1      ;Link to next cache control block
                                CC$LNK=$AC
8 000000   RW      CC$BLK  1      ;Block number
                                CC$BLK=$AC
000002
9 000000   RW      CC$DVU  1      ;Unit number (high byte), device index (low byte)
                                CC$DVU=$AC
000004
10 000000  RW      CC$WCT  1      ;Word count (always positive)
                                CC$WCT=$AC
000006
11 000000  RW      CC$OQE  1      ;Address of original I/O queue element
                                CC$OQE=$AC
000010
12 000000  RW      CC$UBP  1      ;64-byte block number of base of user's buffer
                                CC$UBP=$AC
000012
13 000000  RW      CC$UBD  1      ;Offset within 64-byte block of user buffer base
                                CC$UBD=$AC
000014
14 000000  RW      CC$CBP  1      ;64-byte block number of cache buffer
                                CC$CBP=$AC
000016
15 000000  RB      CC$WFL  1      ;Non-zero ==> Write to device taking place
                                CC$WFL=$AC
000020
16          ;
17 000000  RW      CC$$SZ  0      ;Size of block
                                CC$$SZ=$AC
000022

```

```

1
2          .SBTTL  Shared run-time descriptor block
3          ;-----
4          ; The following descriptor block is used to hold information
5          ; about a shared run-time system.
6          ;
7 000000  RZ
8 000000  RW      RT$DEV  1      ;Rad50 device name          RT$DEV=$AC
          000000
9 000000  RW      RT$NAM  2      ;Rad50 file name          RT$NAM=$AC
          000002
10 000000 RW      RT$EXT  1      ;Rad50 extension          RT$EXT=$AC
          000006
11 000000 RW      RT$BAS  1      ;Base 64-byte block # of run-time in phys memory
          000010 RT$BAS=$AC
12 000000 RW      RT$TOP  1      ;Top 64-byte block # of run-time in phys memory
          000012 RT$TOP=$AC
13 000000 RB      RT$FLG  1      ;Control flags (RF$xxx see below)
          000014 RT$FLG=$AC
14 000000 RB      RT$SKP  1      ;# blocks to skip at front of file
          000015 RT$SKP=$AC
15 000000 RW      RT$$SZ  0      ;Size of descriptor block
          000016 RT$$SZ=$AC
16          ;
17          ; Control flags stored in RT$FLG
18          ;
19 000000  M      RF$WRT  1      ;Write access is allowed to run-time

```

```

1          .SBTTL  Device status flags
2          ;-----
3          ; Fields defined in the device status table DVSTAT.
4          ;
5 000000   M      DS$DIR  100000 ;Device is directory structured
6 000000   M      DS$RON  40000  ;Read-only device
7 000000   M      DS$WON  20000  ;Write-only device
8 000000   M      DS$NRD  10000  ;Non RT-11 directory device (MT & CT)
9 000000   M      DS$ABT  4000   ;Handler abort code flag
10 000000  M      DS$SFN  2000   ;Handler accepts .SPFUN requests
11 000000  M      DS$AJT  1000   ;Enter handler abort entry every time a job terminates
12 000000  M      DS$VSZ  400    ;.SPFUN 373 can be used to determine volume size
13 000000  M      DS$ID   377    ;Device type code (See DI$xxx below)
14          ;
15          ; Device type codes
16          ;
17 000000   M      DI$RK    0      ;RK05
18 000000   M      DI$DT    1      ;TC11 DEctape
19 000000   M      DI$EL    2      ;Error logger
20 000000   M      DI$LP    3      ;Line printer
21 000000   M      DI$TT    4      ;Terminal
22 000000   M      DI$DL    5      ;RL01/RL02
23 000000   M      DI$DY    6      ;RX02
24 000000   M      DI$PC    7      ;PC11 Reader/Punch
25 000000   M      DI$MT   11      ;TM11 tape
26 000000   M      DI$RF   12      ;RF11
27 000000   M      DI$CT   13      ;TA11
28 000000   M      DI$CR   14      ;CR11 Card reader
29 000000   M      DI$DS   16      ;RJS03/RJS04
30 000000   M      DI$MM   20      ;TJU16/TU45 magtape
31 000000   M      DI$DP   21      ;RP11/RP02/RP03 disk
32 000000   M      DI$DX   22      ;RX01
33 000000   M      DI$DM   23      ;RK06/RK07 disk
34 000000   M      DI$NL   25      ;Null device
35 000000   M      DI$DD   34      ;TU58
36 000000   M      DI$MS   35      ;TS11/TS04 magtape
37 000000   M      DI$PD   36      ;PDT-11/130
38 000000   M      DI$LS   41      ;Serial line printer
39 000000   M      DI$MQ   42      ;Message queue server
40 000000   M      DI$DR   43      ;DRV11J
41 000000   M      DI$LD   46      ;Logical disk
42 000000   M      DI$VM   47      ;Virtual memory handler
43 000000   M      DI$DU   50      ;MSCP disk (RAB0,RC25)
44 000000   M      DI$SL   51      ;Single line editor
45 000000   M      DI$PI   54      ;Professional PI handler
46 000000   M      DI$XL   57      ;XL handler for VTCOM
47 000000   M      DI$CL  DI$XL   ;TSX-Plus communications line (CL)
48 000000   M      DI$MU   60      ;TK50 cassette magtape
49 000000   M      DI$NC   61      ;Ethernet class handler

```

```

1      ;
2      ;   Flags defined in device table DVFLAG.
3      ;
4 000000 M      DX$DMA  1      ; This is a DMA device
5 000000 M      DX$MAP  2      ; 18-bit controller -- may require mapped I/O
6 000000 M      DX$EBA  4      ; Buffer must be on even byte boundary
7 000000 M      DX$NCA 10      ; Do not do caching for this device
8 000000 M      DX$NMT 20      ; Do not allow mounts for this device
9 000000 M      DX$RAL 40      ; Require device to be allocated before use
10 000000 M     DX$MPH 100     ; Map the handler for this device
11 000000 M     DX$NHM 200     ; Do not map the handler for this device
12 000000 M     DX$IBH 400     ; Handler contains internal I/O buffer
13 000000 M     DX$NRD 1000    ; Do .SPFUN to tell handler about directory ops
14 000000 M     DX$NST 2000    ; Do not reload handler after SET done to it
15
16      ; -----
17      ;   Logical Disk Status Flags
18      ;   The following flags are stored in the LDFLAG cell for a logical disk:
19      ;
20 000000 M     LD$RON  1      ; Read-only disk
21

```

```

1          .SBTTL Device handler file format
2          ;-----
3          ; The following offsets are to cells containing values in blocks 0 and 1
4          ; of handler files.
5          ;
6          M      H.FET  02      ; Address of handler fetch code
7          M      H.LOAD 04      ; Address of handler load code
8          M      H.SIZ  52      ; Size of handler (bytes)
9          M      H.DVSZ 54      ; Number of 256-word blocks on device
10         M      H.DSTS 56      ; Device status flags
11         M      H.GEN  60      ; Sysgen options for handler
12         M      H.CSR  176     ; CSR address
13         M      H.INS  200     ; Installation routine
14         M      H.VEC  1000    ; Vector address
15         M      H.INT  1002    ; Pointer to interrupt service routine
16         M      H.PRI  1004    ; Priority (340)
17         M      H.LQE  1006    ; Pointer to last queue element
18         M      H.CQE  1010    ; Pointer to current queue element
19         M      H.ENT  1012    ; Handler entry point and HSR$xx flags
20         ;
21         ; Handler service routine flags set in the low-order 5 bits of
22         ; the H.ENT word (the rest of the word is a NOP instruction).
23         ; (Note: This was added in RT-11 version 5.03)
24         ;
25         M      HSR$FE  1      ; Handler Fetch service routine
26         M      HSR$RE  2      ; Handler Release service routine
27         M      HSR$LO  4      ; Handler Load service routine
28         M      HSR$UN  10     ; Handler Unload service routine
29         ;-----
30         ; The following table format is used to describe handler "SET" options.
31         ;
32         RZ
33         RW      SH$VAL  1      ; Value passed in R3 to handler set routine
34         000000                                SH$VAL=$AC
35         RW      SH$NAM  2      ; Rad50 value of option name
36         000000                                SH$NAM=$AC
37         RB      SH$RTN  1      ; Offset to routine in handler for process set option
38         000006                                SH$RTN=$AC
39         RB      SH$FLG  1      ; Control flags for option (See SO$xxx below)
40         000007                                SH$FLG=$AC
41         RW      SH$$SZ  0      ; Size of option descriptor block
42         000010                                SH$$SZ=$AC
43         ;
44         ; Option flags stored in SH$FLG
45         ;
46         M      SO$NO  200     ; "NO" allowed with option word
47         M      SO$NVL 100     ; Numeric parameter required with option
48         M      SO$OCT  40     ; Numeric parameter is in octal
49         .SBTTL Special device function codes
50         ;-----
51         ; The following function codes are passed to the handler for directory
52         ; operations on "Special Devices" such as magnetic tapes and cassettes.
53         ; The function code is stored in the Q.FUNC cell of the I/O queue entry.
54         ;
55         M      DF$CLS  1      ; Close file

```


53 000000	M	DF\$DEL	2	;Delete file
54 000000	M	DF\$LOK	3	;Lookup file
55 000000	M	DF\$ENT	4	;Enter file
56 000000	M	DF\$REN	5	;Rename file


```

1          .SBTTL  File directory entries
2          ;-----
3          ; Format of a file directory entry.
4          ;
5          000000  RZ
6          000000  RW      FD$STA  1      ;Entry status word (see below)
7          000000  RW      FD$NAM  3      ;File name and extension
8          000000  RW      FD$LEN  1      ;Allocated size of file
9          000000  RB      FD$JOB  1      ;Number of job using tentative file entry
10         000000  RB      FD$CHN  1      ;Number of channel using tentative file entry
11         000000  M       FD$TIM  FD$JOB ;Time of file creation (3-second units)
12         000000  RW      FD$DAT  1      ;Creation date of file
13         000000  RW      FD$OPT  0      ;Start of optional words
14         000000  RW      FD$$SZ  0      ;Size of directory entry
15         000016
16         ;
17         ; The following additional entries are allocated in directory entries
18         ; stored in the directory cache table.
19         ; They are not present in file directory entries stored on disk.
20         ; Note, if the size of this entry is changed, TSGEN must be changed.
21         000000  RW      FC$CDX  1      ;Address of cached-device table entry for device
22         000000  RW      FC$LNK  1      ;Link to next entry in cache table
23         000000  M       FC$SBL  FD$JOB ;Starting block # of file
24         000000  RW      FC$$SZ  0      ;Size of cache table entry
25         000022
26         ;
27         ; Status flags stored in FD$STA.
28         000000  M       FS$PRO  100000 ;File is "Protected"
29         000000  M       FS$EOS  4000   ;End of directory segment marker
30         000000  M       FS$PRM  2000   ;Permanent file entry
31         000000  M       FS$EMP  1000   ;Empty file slot
32         000000  M       FS$TEN  400    ;Tentative file entry

```

```

1      ;-----
2      ; Format of Directory Segment header.
3      ;
4 000000 RZ
5 000000 RW      DH$NSG  1      ;Number of directory segments available
                                DH$NSG=$AC
6 000000 RW      DH$NXT  1      ;Number of next segment in directory
                                DH$NXT=$AC
7 000000 RW      DH$HIS  1      ;Number of highest segment currently in use
                                DH$HIS=$AC
8 000000 RW      DH$NEB  1      ;Number of extra bytes per directory entry
                                DH$NEB=$AC
9 000000 RW      DH$BLK  1      ;Block number where files in this segment begin
                                DH$BLK=$AC
10 000000 RW      DH$$SZ  0      ;Size of Directory header
                                DH$$SZ=$AC
11      ;
12 000000 M      DH$$BS  6.      ;Block number where 1st directory segment is located
13 000000 M      DH$$MS  31.     ;Maximum number of directory segments
14 000000 M      DH$$LB  DH$$BS+<2*DH$$MS>-1 ;# of highest block that can have dir
15
16      .SBTTL  Directory cache device table
17      ;-----
18      ; A table with the following types of entries is allocated by TSGEN.
19      ; The table is used to indicate which devices are to have their
20      ; directory entries cached.
21      ; Note, if the size of this table is changed, TSGEN must be changed.
22      ;
23      ; Number of bytes to use for flags to indicate which jobs have
24      ; mounted the device. This should be large enough to allow enough
25      ; bits equal to the maximum job number.
26      ;
27 000000 M      CD$$UB  8.      ;Number of bytes for user-mount flags (64 jobs)
28      ;
29 000000 RZ
30 000000 RW      CD$DVU  1      ;Start of cached-device table entry
                                ;Physical device # (low-order), Phys unit # (high byt)
                                CD$DVU=$AC
31 000000 RW      CD$BAS  1      ;Base block number if this is a logical disk
                                CD$BAS=$AC
32 000000 RW      CD$TOP  1      ;1 above top block if this is a logical disk
                                CD$TOP=$AC
33 000000 RW      CD$NAM  2      ;File name if this is a logical disk
                                CD$NAM=$AC
34 000000 RB      CD$JOB  CD$$UB ;Bit flags indicating which users have device mounted
                                CD$JOB=$AC
35 000000 RW      CD$$SZ  0      ;Size of entry
                                CD$$SZ=$AC

```

```

1          .SBTTL  Assign table
2          ;-----
3          ; Format of table used to hold information for device assignments.
4          ; ASNTBL is label for 1st entry in table, ASNEND is label past last entry.
5          ;
6 000000   RZ
7 000000   RW      AT$LOG  1      ; Logical device name
                                AT$LOG=$AC
8 000000   RW      AT$SIZ  1      ; Size specified with file that is assigned to
                                AT$SIZ=$AC
000000
9 000000   RW      AT$DEV  1      ; Physical device name
                                AT$DEV=$AC
000000
10 000000  RW      AT$FIL  2      ; File name
                                AT$FIL=$AC
000000
11 000000  RW      AT$EXT  1      ; File extension
                                AT$EXT=$AC
000012
12 000000  RW      AT$$SZ  0      ; Size of table entry
                                AT$$SZ=$AC
000014
13
14          .SBTTL  Access command table
15          ;-----
16          ; Format of table entry used to hold information about device-file
17          ; specifications provided by the ACCESS command.
18          ;
19 000000   RZ
20 000000   RW      OF$FIL  3      ; File name and extension
                                OF$FIL=$AC
000000
21 000000   RB      OF$DEV  1      ; Device index number
                                OF$DEV=$AC
000000
22 000000   RB      OF$UNT  1      ; Unit number
                                OF$UNT=$AC
000007
23 000000   RB      OF$FLG  1      ; Control flags (see OT$xxx below)
                                OF$FLG=$AC
000010
24 000000   RW      OF$$SZ  0      ; Size of table entry
                                OF$$SZ=$AC
000012
25          ;
26          ; Control flags stored in OF$FLG.
27          ;
28 000000   M      OT$RON  1      ; Read-only access allowed
29
30          .SBTTL  Device allocation table
31          ;-----
32          ; The following table entries are used to keep track of which device
33          ; units have been allocated by use of the ALLOCATE command.
34          ; Note if the size of these table entries is changed, TSGEN must be changed.
35          ;
36 000000   RZ
37 000000   RW      AD$DVU  1      ; Physical device # (low-order), Phys unit # (high byt)
                                AD$DVU=$AC
000000
38 000000   RB      AD$JOB  1      ; Index number of job to which this device is allocated
                                AD$JOB=$AC
000002
39 000000   RB      AD$FLG  1      ; Control flags
                                AD$FLG=$AC
000003
40 000000   RW      AD$$SZ  0      ; Size of the control block
                                AD$$SZ=$AC
000004

```

1				.SBTTL Fork queue entry block	
2				-----	
3				; Format of a fork queue entry.	
4					
5				; Note: When change the size of the fork queue entry, TSGEN must also	
6				; be altered to reflect the increase.	
7					
8	000000		RZ		
9	000000		RW	FQ\$LNK 1 ;Link to next entry in queue	FQ\$LNK=\$AC
		000000			
10	000000		RW	FQ\$RTN 1 ;Address of routine to call	FQ\$RTN=\$AC
		000002			
11	000000		RW	FQ\$R5 1 ;Saved value of R5	FQ\$R5=\$AC
		000004			
12	000000		RW	FQ\$R4 1 ;Saved value of R4	FQ\$R4=\$AC
		000006			
13	000000		RW	FQ\$R3 1 ;Saved value of R3	FQ\$R3=\$AC
		000010			
14	000000		RW	FQ\$R2 1 ;Saved value of R2	FQ\$R2=\$AC
		000012			
15	000000		RW	FQ\$R1 1 ;Saved value of R1	FQ\$R1=\$AC
		000014			
16	000000		RW	FQ\$UFB 1 ;Addr of FQ\$RTN is user specified fork block	FQ\$UFB=\$AC
		000016			
17	000000		RW	FQ\$PA5 1 ;Saved value of KPAR5	FQ\$PA5=\$AC
		000020			
18	000000		RW	FQ\$PA6 1 ;Saved value of KPAR6	FQ\$PA6=\$AC
		000022			
19	000000		RB	FQ\$PRI 1 ;Fork processing priority value	FQ\$PRI=\$AC
		000024			
20	000000		RW	FQ\$\$SZ 0 ;Size of a fork block	FQ\$\$SZ=\$AC
		000026			

```

1          .SBTTL  Installed program table
2          ;-----
3          ; There is an entry of the following form for each INSTALLED program.
4          ;
5 000000   RZ
6 000000   RW      II$NAM  4      ;Device, file-name, extension      II$NAM=$AC
                                000000
7 000000   RW      II$FLG  1      ;Run flags (see AF$xxx) below      II$FLG=$AC
                                000010
8 000000   RW      II$PRV  PVNPW   ;Privilege flags to enable when program is running  II$PRV=$AC
                                000012
9 000000   RW      II$NPV  PVNPW   ;Privilege flags to disable when program is running  II$NPV=$AC
                                000016
10 000000  RW      II$$SZ  0      ;Size of install table entry      II$$SZ=$AC
                                000022
11          ;
12          ; Flags stored in II$FLG
13          ;
14 000000  M        AF$NOW  1      ;Non-wait TT input
15 000000  M        AF$HIE  2      ;High efficiency mode
16 000000  M        AF$NOI  4      ;Non-interactive execution
17 000000  M        AF$IOP  10     ;Map user PAR 7 to I/O page
18 000000  M        AF$SCA  20     ;Single character activation
19 000000  M        AF$MEM  40     ;Lock program in low memory
20 000000  M        AF$PLK  100    ;Lock program to line (RUN/LOCK)
21 000000  M        AF$DBG  200    ;Run program with debugger
22 000000  M        AF$BYA  400    ;Bypass user ASSIGNs
23 000000  M        AF$TPD  1000   ;Transparent output
24 000000  M        AF$DUP  2000   ;DUP program special handling
25 000000  M        AF$IND  4000   ;IND program special handling
26 000000  M        AF$UCL  10000  ;TSXUCL program special handling
27 000000  M        AF$SET  20000  ;SETUP program special handling
28 000000  M        AF$CCA  40000  ;Suppress control-C abort
29 000000  M        AF$NPW  100000 ;No process windowing while running

```

1
2
3
4
5
6
7
8
9
10 000000
11 000000
12 000000

.SBTTL Memory allocation table values

; The following values are stored in the memory allocation table
; to indicate portion of memory that are being used by the system.
; Memory allocation values for system use must be negative.
; A zero value indicates the memory area is free.
; A positive value indicates the memory area is being used by the
; job whose index number is the value.
;
M MA\$SYS -1 ;Operating system itself
M MA\$SRT -2 ;Shared run-time system
M MA\$RGN -3 ;Shared global PLAS region

		.SBTTL PRO-350 Related Values		
1		-----		
2		; Values related to the PRO-350 system clock.		
3		;		
4		; Registers and vectors		
5		;		
6		;		
7	000000	M	PCCSEC 173000	; Seconds value register
8	000000	M	PCCSAL 173002	; Seconds alarm register
9	000000	M	PCCMIN 173004	; Minutes value register
10	000000	M	PCCMAL 173006	; Minutes alarm register
11	000000	M	PCCHRS 173010	; Hours value register
12	000000	M	PCCHAL 173012	; Hours alarm register
13	000000	M	PCCDAY 173014	; Day value register
14	000000	M	PCCDAT 173016	; Date of month
15	000000	M	PCCMON 173020	; Month value register
16	000000	M	PCCYR 173022	; Year value register
17	000000	M	PCCCR0 173024	; CSR0
18	000000	M	PCCCR1 173026	; CSR1
19	000000	M	PCCCR2 173030	; CSR2
20	000000	M	PCCCR3 173032	; CSR3
21		;		
22	000000	M	PCCVEC 100	; Vector address
23		;		
24		; Flags in CSR2		
25		;		
26	000000	M	PC\$UF 20	; Update ended interrupt flag
27	000000	M	PC\$AF 40	; Alarm interrupt flag
28	000000	M	PC\$PF 100	; Periodic interrupt flag
29	000000	M	PC\$IRQ 200	; Interrupt request flag


```

1          .SBTTL  Spooling Control Tables
2          ;-----
3          ; Format of a Spool File Control Block (SFCB).
4          ; There is one SFCB for each spool file.
5          ; Warning: if the SFCB definition is changed here, it must also be
6          ; changed in TSGEN.
7          ;
8 000000   RZ
9 000000   RB      SFUSER  1      ;Number of user writing to this file
                                SFUSER=$AC
10 000000   RB      SFFLAG  1      ;Control flags (see below)
                                SFFLAG=$AC
11 000000   RW      SFCHAN  1      ;Number of channel opened to this file
                                SFCHAN=$AC
12 000000   RW      SFSDCB  1      ;SDCB wanted by this file
                                SFSDCB=$AC
13 000000   RW      SFNMBL  1      ;Number of blocks for file on disk
                                SFNMBL=$AC
14 000000   RW      SFSTRT  1      ;Block number of start of file on disk
                                SFSTRT=$AC
15 000000   RW      SFFLNK  1      ;Disk address where next block goes
                                SFFLNK=$AC
16 000000   RW      SFFORM  3      ;Form name (6 characters)
                                SFFORM=$AC
17 000000   RW      SFFILE  2      ;File name (Rad50)
                                SFFILE=$AC
18 000000   RW      SFID    1      ;File ID
                                SFID=$AC
19 000000   RW      SFQLNK  1      ;Queue link of files for same device
                                SFQLNK=$AC
20 000000   RW      SFCBSZ  0      ;Size of spool file control block
                                SFCBSZ=$AC
21          ;
22          ; Control flags stored in SFFLAG.
23          ;
24 000000   M      SF$BSY  1      ;File is being processed by spooler
25 000000   M      SF$BN1  2      ;Make 1st write be block # 1 rather than 0
26 000000   M      SF$1ST  4      ;First write has been done
27 000000   M      SF$HLD  10     ;Don't start output till channel closed
28 000000   M      SF$DEL  20     ;Delete this spool file

```



```

1      ; -----
2      ; Format of Spooled Device Control Block (SDCB).
3      ; There is one SDCB for each spooled device.
4      ; Warning: if the definition of a SDCB is changed here, it also must be
5      ; changed in TSGEN.
6      ;
7 000000 RZ
8 000000 RW      SDCHAN  5      ; System channel block for spooled device
                                SDCHAN=$AC
9 000000 000000 RW      SDSFCB  1      ; Current SFCB being processed by spooler
                                SDSFCB=$AC
10 000000 000012 RW      SDFLNK  1      ; Disk address of next block in file
                                SDFLNK=$AC
11 000000 000014 RW      SDBUF1  1      ; Buffer being written by spooler
                                SDBUF1=$AC
12 000000 000016 RW      SDBUF2  1      ; Buffer being read into
                                SDBUF2=$AC
13 000000 000020 RW      SDUSER  1      ; Number of user device is reserved for
                                SDUSER=$AC
14 000000 000022 RW      SDNAME  1      ; RAD50 name of spooled device
                                SDNAME=$AC
15 000000 000024 RW      SDDVU   1      ; Spooled device index (low byte), unit # (high byte)
                                SDDVU=$AC
16 000000 000026 RW      SDWLST  1      ; Queue flink if waiting for a buffer
                                SDWLST=$AC
17 000000 000030 RW      SDBLK   1      ; Block number for next spooled device write
                                SDBLK=$AC
18 000000 000032 RW      SDFHD   1      ; Pointer to first SFCB waiting for this device
                                SDFHD=$AC
19 000000 000034 RW      SDFORM  3      ; Current form name (6 chars)
                                SDFORM=$AC
20 000000 000036 RB      SDANAM  3      ; ASCII name of spooled device (only 3 chars used)
                                SDANAM=$AC
21 000000 000044 RW      SDFLAG  1      ; Control flags (see below)
                                SDFLAG=$AC
22 000000 000050 RW      SDSKIP  1      ; Number of blocks to skip
                                SDSKIP=$AC
23 000000 000052 RW      SDFRBL  1      ; Number of free private spool blocks for device
                                SDFRBL=$AC
24 000000 000054 RW      SDBU    0      ; Backup save block numbers (must be last)
                                SDBU=$AC
25      ;
26      ; Control flags in SDFLAG.
27      ;
28 000000 M      SD$INR  1      ; Read from spool file in progress
29 000000 M      SD$BWT  2      ; Spooler is waiting for a buffer
30 000000 M      SD$DEL  4      ; Delete current file
31 000000 M      SD$WFM  10     ; Waiting for form mount
32 000000 M      SD$FLK  20     ; Form is locked
33 000000 M      SD$HLD  40     ; Don't start output till channel is closed
34 000000 M      SD$SNG  100    ; Single file mode
35 000000 M      SD$SMS  200    ; Form mount message sent
36 000000 M      SD$BAK  400    ; Backup has been requested
37 000000 M      SD$CLR  SD$DEL!SD$WFM!SD$SMS!SD$BAK

```

1			. SBTTL Accounting file format		
2			-----		
3			; The following items describe the format of an accounting file record.		
4					
5	000000		RZ		
6	000000		RB	AR\$PRJ 2.	; Project number
		000000			AR\$PRJ=\$AC
7	000000		RB	AR\$PRG 2.	; Programmer number
		000002			AR\$PRG=\$AC
8	000000		RB	AR\$PWD 8.	; Password
		000004			AR\$PWD=\$AC
9	000000		RB	AR\$PRV 2.	; Privilege flags
		000014			AR\$PRV=\$AC
10	000000		RB	AR\$SUF 16.	; Start-up command file
		000016			AR\$SUF=\$AC
11	000000		RB	AR\$CON 2.	; Connect time (minutes)
		000036			AR\$CON=\$AC
12	000000		RB	AR\$CNT 2.	; Number of sessions
		000040			AR\$CNT=\$AC
13	000000		RB	AR\$CPH 2.	; High-order CPU time
		000042			AR\$CPH=\$AC
14	000000		RB	AR\$CPL 2.	; Low-order CPU time
		000044			AR\$CPL=\$AC
15	000000		RB	AR\$UNM 12.	; User name
		000046			AR\$UNM=\$AC
16	000000		RB	AR\$PRI 1.	; Maximum authorized priority
		000062			AR\$PRI=\$AC
17	000000		RB	AR\$DMY 13.	; Unused (reserved)
		000063			AR\$DMY=\$AC
18	000000		RW	AR\$\$SZ 0	; Size of authorization record
		000100			AR\$\$SZ=\$AC
19					
20	000000		M	ARNRPB 512./AR\$\$SZ	; Number of authorization records per block

```
1          .SBTTL  Log file control flags
2          ;-----
3          ; Control flags stored in LOGFLG used to control the operation of
4          ; the log file.
5          ;
6 000000 M      LF$OPN  1      ;Log file is open
7 000000 M      LF$WRT  2      ;Enable writes to log file
8 000000 M      LF$IN   4      ;Log input (received) characters
9 000000 M      LF$OUT  10     ;Log output (transmitted) characters
```

```

1          .SBTTL  Shared file control tables
2          ;-----
3          ; Shared file Channel Descriptor Block (CDB).
4          ; There is one active CDB for each channel opened to a shared file.
5          ; Free list head = FCFREE.
6          ;
7 000000   RZ
8 000000   RB      FC$CHN  1      ; Channel number
                                FC$CHN=$AC
9 000000   RB      FC$UN   1      ; User number
                                FC$UN=$AC
000001
10 000000  RB      FC$FLG  1      ; Control flags (see below)
                                FC$FLG=$AC
000002
11 000000  RB      FC$NLB  1      ; Number of blocks currently locked by channel
                                FC$NLB=$AC
000003
12 000000  RW      FC$ACC  1      ; Access protection flags
                                FC$ACC=$AC
000004
13 000000  RW      FC$FDB  2      ; Pointer to FDB for file
                                FC$FDB=$AC
000006
14 000000  RW      FC$FLK  2      ; Flink to next CDB for this file
                                FC$FLK=$AC
000012
15 000000  RW      FC$CLK  2      ; Flink to next CDB for this user
                                FC$CLK=$AC
000016
16 000000  RW      FC$LBN  0      ; Start of table of locked block numbers
                                FC$LBN=$AC
000022
17          ;
18 000000  RW      FC$$SS  0      ; Size of block
                                FC$$SS=$AC
000022
19          ;
20          ; Control flags found in FC$FLG.
21          ;
22 000000  M      FL$ACT  1      ; Some other user has written to this file
23 000000  M      FL$EFL  2      ; Entire file is locked
24 000000  M      FL$NDC  4      ; Suppress data caching
25 000000  M      FL$SPN 10      ; CDB is suspended (because of save status)

```

```

1      ; -----
2      ; Format of Shared File Descriptor Block (FDB).
3      ; There is one active FDB for each shared file that is open.
4      ; Free head = FFFREE; Active list head = FFHEAD.
5      ;
6 000000 RZ
7 000000 RW      FF$FID  2      ;File identification          FF$FID=$AC
                                000000
8 000000 RW      FF$CDB  2      ;Pointer to first CDB for this file  FF$CDB=$AC
                                000004
9 000000 RW      FF$DCD  2      ;Pointer to first data cache descriptor for this file  FF$DCD=$AC
                                000010
10 000000 RW      FF$FWD  2      ;Pointer to first wait block for this file  FF$FWD=$AC
                                000014
11 000000 RW      FF$NLB  1      ;Number of blocks currently locked  FF$NLB=$AC
                                000020
12 000000 RB      FF$FLG  1      ;Status flags (see below)      FF$FLG=$AC
                                000022
13 000000 RW      FF$FLK  2      ;Flink to next active FDB      FF$FLK=$AC
                                000024
14 000000 RW      FF$$SZ  0      ;Size of FDB      FF$$SZ=$AC
                                000030
15      ;
16      ; Status flags stored in FF$FLG.
17      ;
18 000000 M      FT$EFL  1      ;Entire file locked
19
20      ; -----
21      ; Format of shared file wait queue element.
22      ; There is one entry for each user who is waiting for a locked block.
23      ; Free head = FWFREE.
24      ;
25 000000 RZ
26 000000 RW      FW$DBN  1      ;Block number we are waiting for  FW$DBN=$AC
                                000000
27 000000 RW      FW$UN  1      ;User number      FW$UN=$AC
                                000002
28 000000 RW      FW$WLK  2      ;Flink to next wait element for this file  FW$WLK=$AC
                                000004
29 000000 RW      FW$$SZ  0      ;Size of wait queue element  FW$$SZ=$AC
                                000010

```

```

1      ;-----
2      ; Format of a data cache descriptor block
3      ; There is one of these descriptor blocks for each data cache buffer.
4      ; (Note: the size of these descriptor blocks is assumed in TSGEN)
5      ;
6 000000 RZ
7 000000 RW      DC$NXT  2      ;Pointer to next DCD
                                DC$NXT=$AC
8 000000 RW      DC$FDB  2      ;Addr of FDB for shared file assoc with this entry
                                DC$FDB=$AC
9 000000 RW      DC$BLK  1      ;File block # whose data is in this cache buffer
                                DC$BLK=$AC
10 000000 RW      DC$USE  1      ;Use counter
                                DC$USE=$AC
11 000000 RW      DC$LNK  2      ;Link to next cache descriptor for this file
                                DC$LNK=$AC
12 000000 RW      DC$PAR  1      ;Map PAR address for this cache buffer
                                DC$PAR=$AC
13 000000 RW      DC$$SZ  0      ;Size of cache descriptor
                                DC$$SZ=$AC
000000

```

1		.SBTTL	Message communication control tables	
2			-----	
3			Offsets into EMT block for message EMT's.	
4				
5	000000	RZ		
6	000000	RW	EB\$XX 1 ; Not used	EB\$XX=\$AC
	000000			
7	000000	RW	EB\$NAM 1 ; Pointer to channel name string	EB\$NAM=\$AC
	000002			
8	000000	RW	EB\$BUF 1 ; Address of user's buffer	EB\$BUF=\$AC
	000004			
9	000000	RW	EB\$SIZ 1 ; Message byte size	EB\$SIZ=\$AC
	000006			
10	000000	RW	EB\$RTN 1 ; Address of completion routine	EB\$RTN=\$AC
	000010			
11				
12			Format of message control block	
13				
14	000000	RZ		
15	000000	RW	MB\$FLK 1 ; Link to next message control block	MB\$FLK=\$AC
	000000			
16	000000	RW	MB\$NAM 3 ; Name of message channel (6 characters)	MB\$NAM=\$AC
	000002			
17	000000	RW	MB\$BUF 1 ; Pointer to 1st pending message text block	MB\$BUF=\$AC
	000010			
18	000000	RW	MB\$REQ 1 ; Pointer to 1st pending message request block	MB\$REQ=\$AC
	000012			
19	000000	RW	MB\$\$SZ 0 ; Size of message control block	MB\$\$SZ=\$AC
	000014			
20				
21			Format of a message request block	
22				
23	000000	RZ		
24	000000	RW	MR\$LNK 1 ; Link to next message request block	MR\$LNK=\$AC
	000000			
25	000000	RW	MR\$UBA 1 ; Address of user's buffer area	MR\$UBA=\$AC
	000002			
26	000000	RW	MR\$UBS 1 ; Size of user's buffer area	MR\$UBS=\$AC
	000004			
27	000000	RW	MR\$RTN 1 ; Address of user's completion routine	MR\$RTN=\$AC
	000006			
28	000000	RW	MR\$BUF 1 ; Pointer to message text buffer to be passed	MR\$BUF=\$AC
	000010			
29	000000	RB	MR\$JOB 1 ; Job index number	MR\$JOB=\$AC
	000012			
30	000000	RW	MR\$\$SZ 0 ; Size of block	MR\$\$SZ=\$AC
	000014			
31				
32			Format of a message text buffer	
33				
34	000000	RZ		
35	000000	RW	MU\$FLK 1 ; Link to next buffer	MU\$FLK=\$AC
	000000			
36	000000	RW	MU\$SIZ 1 ; Message size (bytes)	MU\$SIZ=\$AC
	000002			
37	000000	RB	MU\$JOB 1 ; Job index number of job that sent message	MU\$JOB=\$AC
	000004			

38 000000

000006

RW

MU\$TXT 0

;Start of message text

MU\$TXT=\$AC


```

1      .SBTTL    Generic data set control and status flags
2      ;-----
3      ;    The following generic status and control flags are used to check the
4      ;    status of data sets and perform control operations.
5      ;
6 000000 M        MS$RNG    1        ;Phone is ringing
7 000000 M        MS$CAR    2        ;Carrier is detected
8 000000 M        MS$DTR    4        ;Data Terminal Ready is asserted
9 000000 M        MS$BRK   10        ;Transmit a break
10
11
12      .SBTTL    DL11 Control and Status Registers
13      ;-----
14      ;    DL11 Line interface control and status flags.
15      ;
16      ;    Receiver status register
17      ;
18 000000 M        STATCH   100000 ;Data set status change
19 000000 M        RING     40000   ;Ring indication
20 000000 M        SNDCLR   20000   ;Clear to send
21 000000 M        CARDET   10000   ;Carrier detect
22 000000 M        RCVACT   4000    ;Receiver is active
23 000000 M        RCVDON   200     ;Receive done
24 000000 M        RCVINT   100     ;Receiver interrupt enable
25 000000 M        DSINT    40      ;Data set interrupt enable
26 000000 M        REQSND   4       ;Request to send
27 000000 M        TRMRDY   2       ;Data terminal ready
28 000000 M        RDINT    RCVINT
29
30      ;    Receiver data buffer register
31      ;
32 000000 M        RBERR    100000 ;Error in input
33 000000 M        OVRRUN   40000   ;Character overrun
34 000000 M        FRMERR   20000   ;Framing error
35 000000 M        RCVPAR   10000   ;Receive parity error
36
37      ;    Transmitter status register
38      ;
39 000000 M        TRRDY    200     ;Transmitter ready
40 000000 M        TRINT    100     ;Transmitter interrupt enable
41 000000 M        TRBRK    1       ;Transmit a break

```

1			.SBTTL DZ11 Control and Status Registers
2			-----
3			; DZ11 Mux flags
4			;
5			;
6			; Control and Status Register (CSR).
7			;
8	000000	M	TRDY 100000 ;Transmitter ready
9	000000	M	TIE 40000 ;Transmitter interrupt enable
10	000000	M	TLINE 3400 ;Transmitter line number
11	000000	M	RDONE 200 ;Receiver done
12	000000	M	RIE 100 ;Receiver interrupt enable
13	000000	M	MSE 40 ;Master scan enable
14	000000	M	ZCLR 20 ;Master clear
15	000000	M	INTMX1 TIE!RIE!MSE ;Initial flags for DZ11
16			;
17			; Receiver Buffer Register (RBUF).
18			;
19	000000	M	DATVAL 100000 ;Data value
20	000000	M	ZFRMER 20000 ;Framing error
21	000000	M	DZERR 70000 ;DZ11 input error flags
22	000000	M	RLINE 3400 ;Receive line number
23			;
24			; Line parameter register (LPR)
25			;
26	000000	M	DZ\$PAR 100 ;Enable parity
27	000000	M	DZ\$ODD 200 ;Odd parity
28	000000	M	DZ\$LEN 30 ;Character length code
29	000000	M	DZ\$7BT 20 ;7 bit characters
30	000000	M	DZ\$8BT 30 ;8 bit characters

```

1          .SBTTL  DH11 Control and Status Registers
2          ;-----
3          ; DH11 Control and Status Registers
4          ;
5          ; Status flags stored in the DH11 System Control Register (MH$SCR)
6          ;
7 000000 M      HF$TI  100000 ; Transmitter interrupt flag
8 000000 M      HF$SI  40000  ; Storage interrupt flag
9 000000 M      HF$TIE 20000  ; Transmitter interrupt enable flag
10 000000 M     HF$SIE 10000  ; Storage interrupt enable flag
11 000000 M     HF$MC   4000  ; Master clear
12 000000 M     HF$NXM  2000  ; Non-existent memory flag
13 000000 M     HF$MM   1000  ; Maintenance mode
14 000000 M     HF$CNI  400   ; Clear non-existent memory interrupt flag
15 000000 M     HF$RI   200   ; Receiver interrupt flag
16 000000 M     HF$RIE  100   ; Receiver interrupt enable flag
17 000000 M     HF$LIN  17    ; Line number field mask
18          ;
19          ; Status flags stored in DH11 Received Character Register (MH$RCR)
20          ;
21 000000 M     HF$VDP  100000 ; Valid Data Character Present flag
22 000000 M     HF$DO   40000  ; Data overrun
23 000000 M     HF$FE   20000  ; Framing error
24 000000 M     HF$PE   10000  ; Parity error
25          ;
26          ; Status flags stored in DH11 Line Parameter Register (MH$LPR)
27          ;
28 000000 M     HF$AEE  100000 ; Auto echo enable
29 000000 M     HF$HD   40000  ; Half-duplex
30 000000 M     HF$ODD   40    ; Odd parity
31 000000 M     HF$PAR   20    ; Parity enable
32 000000 M     HF$TSB   4     ; Two stop bits
33 000000 M     HF$LEN   3     ; Character length field
34 000000 M     HF$7BT   2     ; 7 bit characters
35 000000 M     HF$8BT   3     ; 8 bit characters
36          ;
37          ; Status flags stored in DM11 Control Status Register (DM$CSR)
38          ;
39 000000 M     MF$CS   4000    ; Clear scanner
40 000000 M     MF$CM   2000    ; Clear multiplexer
41 000000 M     MF$STP   400    ; Step to next line
42 000000 M     MF$DON   200    ; Done
43 000000 M     MF$IE   100    ; Interrupt enable
44 000000 M     MF$SE   40     ; Scanner enable
45 000000 M     MF$BSY   20     ; Busy
46 000000 M     MF$LIN  17     ; Line number field mask
47          ;
48          ; Status flags stored in DM11 Line Status Register (DM$LSR)
49          ;
50 000000 M     MF$RNG   200    ; Ring signal
51 000000 M     MF$CAR   100    ; Carrier detect
52 000000 M     MF$CTS   40     ; Clear to send
53 000000 M     MF$SR    20     ; Secondary receive
54 000000 M     MF$ST    10     ; Secondary transmit
55 000000 M     MF$RTS   4     ; Request to send
56 000000 M     MF$DTR   2     ; Data terminal ready
57 000000 M     MF$LE    1     ; Line enable

```

```

1          .SBTTL   DHV11 Control and Status Registers
2          ;-----
3          ; Status and control flags for DHV11 registers
4          ;
5          ; Status flags stored in the DHV11 Control Status Register (VH$CSR)
6          ;
7 000000   M      VF$TR   100000 ;Transmitter ready for another character
8 000000   M      VF$TIE  40000  ;Transmitter interrupt enable
9 000000   M      VF$DF   20000  ;Diagnostics failure
10 000000  M      VF$TDE  10000  ;Transmit DMA error
11 000000  M      VF$XLN  3400   ;Transmitter interrupt line # field
12 000000  M      VF$RDA  200    ;Receive data available
13 000000  M      VF$RIE  100    ;Receive interrupt enable
14 000000  M      VF$MR   40     ;Master reset
15 000000  M      VF$LIN  17     ;Line number select field
16          ;
17          ; Status flags stored in the DHV11 Data Buffer Register (VH$DBR)
18          ; The following flags are in the receiver buffer register which is accessed
19          ; when VH$DBR is read.
20          ;
21 000000   M      VF$DV   100000 ;Data valid
22 000000   M      VF$OE   40000  ;Data overrun error
23 000000   M      VF$FE   20000  ;Framing error
24 000000   M      VF$PER  10000  ;Parity error
25          ;
26          ; The following status flags are used with the transmitter buffer register
27          ; which is accessed when VH$DBR is written.
28          ;
29 000000   M      VF$TDV  100000 ;Transmit data valid
30          ;
31          ; Status flags stored in the DHV11 Line Parameter Register (VH$LPR)
32          ;
33 000000   M      VF$LEN  30      ;Field for character length
34 000000   M      VF$7BT  20      ;7 bit characters
35 000000   M      VF$8BT  30      ;8 bit characters
36 000000   M      VF$PAR  40      ;Parity enable
37 000000   M      VF$EVN  100     ;Even parity
38 000000   M      VF$SC   200     ;Stop code
39          ;
40          ; Status flags stored in the DHV11 Line Status Register (VH$LSR)
41          ;
42 000000   M      VF$DSR  100000 ;Data set ready
43 000000   M      VF$RNG  20000  ;Ring indication
44 000000   M      VF$DCD  10000  ;Carrier detected
45 000000   M      VF$CTS  4000   ;Clear to send
46          ;
47          ; Status flags stored in the DHV11 Line Control Register (VH$LCR)
48          ;
49 000000   M      VF$RTS  10000  ;Request to send
50 000000   M      VF$DTR  1000   ;Data terminal ready
51 000000   M      VF$LT   400    ;Link type (1=>Modem RTS/CTS protocol)
52 000000   M      VF$XOF  40     ;Force transmission of XOFF
53 000000   M      VF$OFC  20     ;Do automatic output flow control
54 000000   M      VF$BC   10     ;Break control
55 000000   M      VF$RE   4      ;Receiver enable
56 000000   M      VF$IFC  2      ;Automatic flow control
57 000000   M      VF$ABT  1      ;DMA transfer abort

```

58				
59				; Status flags stored in the DHV11 transmitter buffer addr reg 2
60				
61	000000	M	VF\$TEN 100000	;Enable transmitter
62	000000	M	VF\$TGO 200	;Start DMA transmission
63				

		.SBTTL Line Speed Codes			
1		-----			
2					
3		; The following symbolic values are used to encode the line speed codes.			
4					
5	000000	M	S50	0	;50
6	000000	M	S75	1	;75
7	000000	M	S110	2	;110
8	000000	M	S134.5	3	;134.5
9	000000	M	S150	4	;150
10	000000	M	S300	5	;300
11	000000	M	S600	6	;600
12	000000	M	S1200	7	;1200
13	000000	M	S1800	10	;1800
14	000000	M	S2000	11	;2000
15	000000	M	S2400	12	;2400
16	000000	M	S3600	13	;3600
17	000000	M	S4800	14	;4800
18	000000	M	S7200	15	;7200
19	000000	M	S9600	16	;9600
20	000000	M	S19200	17	;19200
21					
22		; The following flags are used to encode the character length, parity			
23		; on/off, and even/odd status for a line.			
24					
25	000000	M	LP\$SPD	17	;Bits used to encode speed (see above for values)
26	000000	M	LP\$7BT	40	;0==>8 bit characters, 1==>7 bit characters
27	000000	M	LP\$PAR	100	;0==>No parity, 1==>Enable parity
28	000000	M	LP\$ODD	200	;0==>Even parity, 1==>Odd parity

```

1          .SBTTL    Communication Line (CL) Handler Flags
2          ;-----
3          ;    The following flags relate to the CL handler
4          ;
5          ;    Option flags stored in CL$OPT table
6          ;
7 000000    M        CD$FF    1            ; Device supports hardware form feeds
8 000000    M        CD$TAB   2            ; Device supports hardware tabs
9 000000    M        CD$LC    4            ; Allow lower-case characters to be sent
10 000000   M        CD$LFO   10           ; Transmit line-feed chars to device
11 000000   M        CD$LFI   20           ; Accept line feed characters coming in
12 000000   M        CD$FFO   40           ; Send form feed on write of block 0
13 000000   M        CD$BNO   100          ; Binary output mode
14 000000   M        CD$BNI   200          ; Binary input mode
15 000000   M        CD$CR    400          ; Allow carriage return to be sent
16 000000   M        CD$CTL   1000        ; Allow control characters to be sent
17 000000   M        CD$DTR   2000        ; Raise Data Terminal Ready (DTR)
18 000000   M        CD$BBT   4000        ; Eight bit character support
19          ;
20 000000   M        CD$DEF   CD$LC!CD$LFO!CD$LFI!CD$CR!CD$CTL ; Default option flags
21          ;
22          ;    Status flags stored in CL$STA table
23          ;
24 000000   M        CM$WRT   2            ; A write operation has been done to this unit
25 000000   M        CM$EFP   4            ; Currently doing end-of-file output processing
26 000000   M        CM$CRL   10           ; Carriage return was last char transmitted
27 000000   M        CM$TBS   20           ; Doing tab expansion simulation
28 000000   M        CM$EOF   40           ; Return end-of-file on next read
29 000000   M        CM$ON    100          ; Line is turned on
30 000000   M        CM$BRK   200          ; Currently transmitting a break
31 000000   M        CM$FFS   400          ; Doing form-feed simulation
32 000000   M        CM$IRG   1000        ; IRINGG routine active for this unit
33 000000   M        CM$ORP   2000        ; ORINGP routine active for this unit
34 000000   M        CM$DTR   4000        ; Data Terminal Ready has been asserted
35 000000   M        CM$MCC   10000       ; Next char is modem control or literal char
36 000000   M        CM$FFI   20000       ; Ignore FF if it immediately follows a skip
37          ;
38          ;    Status flags returned by SPFUN 204 (get CL status)
39          ;
40 000000   M        XL$XFX   1            ; XOFF has been transmitted
41 000000   M        XL$XFR   2            ; XOFF has been received
42 000000   M        XL$CTS   4            ; CTS is asserted
43 000000   M        XL$CD    10           ; Carrier is detected
44 000000   M        XL$RI    20           ; Ring is detected
45          ;
46          ;    Special function code values for CL handler
47          ;
48 000000   M        CLSFCH   201          ; Clear handler status
49 000000   M        CLSFBC   202          ; Break transmission control
50 000000   M        CLSFRB   203          ; Read with byte count
51 000000   M        CLSFHS   204          ; Get handler status
52 000000   M        CLSFDL   205          ; Deactivate line
53 000000   M        CLSFSD   250          ; Set selected option flags
54 000000   M        CLSFCD   251          ; Clear selected option flags
55 000000   M        CLSFSL   252          ; Set page length
56 000000   M        CLSFSS   253          ; Set number of lines to skip
57 000000   M        CLSFSW   254          ; Set line width

```


58 000000	M	CLSFMS	255	;Get data set status code
59 000000	M	CLSFSP	256	;Set receive/transmit speed
60 000000	M	CLSFAB	257	;Abort all pending reads and writes
61 000000	M	CLSFRL	260	;Read line with byte count
62 000000	M	CLSFIC	261	;Get # pending input bytes
63 000000	M	CLSFOC	262	;Get # pending output bytes
64 000000	M	CLSFWB	263	;Write with byte count
65 000000	M	CLSFEP	264	;Set end-of-file output control
66 000000	M	CLSFRR	265	;Reset status of CL unit
67 000000	M	CLSFGR	266	;Get CL options and settings


```

1          .SBTTL  Memory Management values
2          ;-----
3          ; Values related to memory management
4          ;
5          ; Status and control registers
6          ;
7          ; Kernel-mode page base registers
8          ;
9 000000    M      KPAR0    172340
10 000000   M      KPAR1    KPAR0+2
11 000000   M      KPAR2    KPAR0+4
12 000000   M      KPAR3    KPAR0+6
13 000000   M      KPAR4    KPAR0+10
14 000000   M      KPAR5    KPAR0+12
15 000000   M      KPAR6    KPAR0+14
16 000000   M      KPAR7    KPAR0+16
17          ;
18          ; Kernel-mode page descriptor registers
19          ;
20 000000   M      KPDR0    172300
21 000000   M      KPDR1    KPDR0+2
22 000000   M      KPDR2    KPDR0+4
23 000000   M      KPDR3    KPDR0+6
24 000000   M      KPDR4    KPDR0+10
25 000000   M      KPDR5    KPDR0+12
26 000000   M      KPDR6    KPDR0+14
27 000000   M      KPDR7    KPDR0+16
28          ;
29          ; User-mode page address registers
30          ;
31 000000   M      UPAR0    177640
32 000000   M      UPAR1    UPAR0+2
33 000000   M      UPAR2    UPAR0+4
34 000000   M      UPAR3    UPAR0+6
35 000000   M      UPAR4    UPAR0+10
36 000000   M      UPAR5    UPAR0+12
37 000000   M      UPAR6    UPAR0+14
38 000000   M      UPAR7    UPAR0+16
39          ;
40          ; User-mode page descriptor registers
41          ;
42 000000   M      UPDR0    177600
43 000000   M      UPDR1    UPDR0+2
44 000000   M      UPDR2    UPDR0+4
45 000000   M      UPDR3    UPDR0+6
46 000000   M      UPDR4    UPDR0+10
47 000000   M      UPDR5    UPDR0+12
48 000000   M      UPDR6    UPDR0+14
49 000000   M      UPDR7    UPDR0+16
50          ;
51          ; Control and status registers
52          ;
53 000000   M      SROMMR    177572
54 000000   M      SR1MMR    177574
55 000000   M      SR2MMR    177576
56 000000   M      SR3MMR    172516
57          ;

```

58				; Unibus map registers
59				;
60	000000	M	UMRADR 170200	; Start of Unibus map registers
61				;
62				; Memory parity control registers
63				;
64	000000	M	MPARO 172100	; Memory parity control register # 0
65	000000	M	MPAR16 172136	; Memory parity control register # 16
66				;
67				; Misc values
68				;
69	000000	M	MMENBL 1	; Memory management enable bit in SRO
70	000000	M	EMMAP 20	; Extended 22-bit memory mapping
71	000000	M	IDMAP 40	; Extended UNIBUS I/O 22-bit mapping
72	000000	M	PARENL 1	; Enable memory parity

```

1
2
3
4
5
6
7 000000
8 000000
9 000000
10 000000
11 000000
12 000000
13 000000
14 000000
15 000000
16 000000
17 000000
18
19
20
21
22
23
24 000000
25 000000
26 000000
27 000000

```

.SBTTL Terminal type names			

; The following terminal type names are used with the TRMTYP macro			
; in TSGEN to define terminal types. The actual type code is stored			
; in the LTRMTP line table.			
M	LA36	1	; DEC LA36
M	LA120	2	; DEC LA120
M	VT52	10	; DEC VT52
M	VT100	20	; DEC VT100
M	ADM3A	100	; Lear Siegler ADM3A
M	HAZEL	200	; Hazeltine
M	DIABLO	1000	; Diablo
M	QUME	2000	; Qume
M	VT2007	4000	; VT200 with 7 bit control codes
M	VT2008	10000	; VT200 with 8 bit control codes
M	VT200	VT2007	; Generic VT200
.SBTTL System Editor names			

; The following symbolic names are used in TSGEN to declare the default			
; system editor.			
M	EDIT	1	; EDIT
M	TECO	2	; TECO
M	KED	3	; KED or K52
M	K52	4	; K52 or KED

```

1          .SBTTL  Flags in Job Status Word (JSW)
2          ;-----
3          ;  Flags stored in the Job Status Word (JSW).
4          ;
5 000000   M      NOUSWP  100000 ;Disable USR swapping
6 000000   M      LCBIT   40000  ;Enable lower case input
7 000000   M      REENT   20000  ;Program can be reentered
8 000000   M      SPCTTY  10000  ;Special TT mode (i.e., EDIT)
9 000000   M      PASLIN  4000   ;Pass line to KMON on exit
10 000000  M      VIMAGE  2000   ;Virtual image flag
11 000000  M      OVLBIT  1000   ;Program is overlaid
12 000000  M      CHAIN   400    ;Program was chained to
13 000000  M      ERRHLT  200    ;Halt on I/O error
14 000000  M      NOWAIT  100    ;No wait on .TTYIN requests
15 000000  M      SCHAIN  40     ;Special chain - pass command to KMON
16 000000  M      DISSLE  20     ;Disable single-line editor
17 000000  M      GTLTYY  10     ;.GTLIN from TT not @file
18
19          .SBTTL  Performance monitor control and status flags
20          ;-----
21          ;  The following flags are stored in PMFLQS and control the operation
22          ;  of the TSX-Plus performance monitoring feature.
23          ;
24 000000   M      PF$IQW  1      ;Include I/O wait time in analysis
25 000000   M      PF$SYS  2      ;Analyze system execution rather than user job
26 000000   M      PF$OVF  100000 ;An overflow occurred
27
28          ;-----
29          ;  The following flags are returned in response to the JBINFO EMT
30          ;  that is used to determine the status of a specific line.
31          ;
32 000000   M      JIVLN   1      ;This is a virtual line
33 000000   M      JIDLN   2      ;This is a detached line
34 000000   M      JIMLOK  100    ;Job is locked in memory
35 000000   M      JIPRIV  200    ;Job has operator command privilege
36
37          .SBTTL  Simulated RMON parameters
38          ;-----
39          ;  The following parameters relate to the simulated RMON that is
40          ;  mapped into page 7 of the user's virtual address space.
41          ;
42 000000   M      RMNBAS  160002 ;Virtual address of simulated RMON value vector
43 000000   M      INDERR  RMNBAS+416 ;IND error cell virtual address
44 000000   M      INDSTA  RMNBAS+417 ;IND status flag cell virtual address

```

Misc. parameters

```

1          .SBTTL  Misc. parameters
2          ;-----
3          ; Misc. parameters.
4          ;
5 000000   M      MAXSLO 255.    ;Max size of TT and CL silo buffers (bytes)
6 000000   M      MAXASN 25.     ;Max # ASSIGN commands allowed
7 000000   M      MAXACC 30.     ;Max # ACCESS commands allowed
8 000000   M      MXCPRM 6.      ;Max # parameters on command file call
9 000000   M      MXCCHR 60.     ;Max # chars in command file paramameter string
10 000000  M      MAXLD  8.       ;Max # of logical disks
11 000000  M      NUMRCB 8.       ;Max # PLAS regions that can be created by job
12 000000  M      NUMWCB 8.       ;Max # PLAS windows that can be created by job
13 000000  M      MXPRMT 8.       ;Max # chars in KMON prompt string
14 000000  M      SLMXLN 80.     ;Max # chars in Single Line Editor buffer
15 000000  M      SLBFSZ 300.    ;Size of SLE save buffer
16 000000  M      MAXSRD 40.     ;Max # shared run-time region descriptors
17 000000  M      MAXPRI 127.    ;Maximum job priority
18 000000  M      VPAR1  20000   ;Virtual address of PAR 1 region
19 000000  M      VPAR5  120000  ;Virtual address of PAR 5 system mapping region
20 000000  M      CXTBAS 140000  ;Virtual address of Job Context Area
21 000000  M      VPAR6  140000  ;Virtual address of PAR 6 mapping region
22 000000  M      IOPAGE 177600  ;I/O page address
23 000000  M      CPLEMT 2       ;Location where completion routine exit emt is stored
24 000000  M      SUPRTN 300     ;Location where GETCSR routine for SETUP is stored
25 000000  M      USRLOC 46      ;Cell with USR load address
26 000000  M      JSWLOC 44      ;Cell with Job Status Word
27 000000  M      HIMLOC 50      ;Cell with Top of memory address
28 000000  M      ERRLOC 52      ;Cell with EMT error code
29 000000  M      USERRB 53      ;Error severity set by user
30 000000  M      RMON   54      ;Cell with RMON base address
31 000000  M      BLKWDS 256.    ;Number of words per disk block
32 000000  M      ACFLAG 100000  ;Flag character as an activation char
33 000000  M      ESCFLG 377     ;Flag char that says following char part of escape seq
34 000000  M      TRNSFL 100000  ;Send char in transparency mode
35 000000  M      CCFLG  100000  ;Ctrl-C trap flag for .SCCA
36 000000  M      DIBFSZ 155.    ;Size of Diablo character buffer
37 000000  M      SBUFSZ 512.    ;Size of spool buffer (bytes)
38 000000  M      SBUFWD 256.    ;Size of spool buffer (words)
39 000000  M      PR7    340     ;Priority 7 in PS
40 000000  M      CFLAG 1        ;C-flag in PS
41 000000  M      KMONBAS 40000   ;KMON base address
42 000000  M      PSW    177776  ;PS word
43 000000  M      UMODE  140000  ;User-mode bits in PS
44 000000  M      UPMODE 30000   ;User-previous-mode bits in PS
45 000000  M      CTTSR  177564  ;Transmitter status register for console terminal
46 000000  M      CTTBR  177566  ;Transmitter buffer register for console terminal
47 000000  M      DMYDEV 126370  ;Dummy device name to reserve patch space ("$$")
48 000000  M      WLDNAM 132500  ;Wildcard file name (Rad50 /*/)
49 000000  M      DMPOVL 350     ;Loc overlay name is passed to dump
50 000000  M      DMPHND 352     ;Loc handler name is passed to dump
51 000000  M      DMPTXT 354     ;Loc of start of asciz dump message text string

```

1				.SBTTL User error severity status codes
2				-----
3				; Error severity status code values stored in USERRB (location 53).
4				;
5	000000	M	SC\$SUC 1	; Success
6	000000	M	SC\$WRN 2	; Warning
7	000000	M	SC\$ERR 4	; Error
8	000000	M	SC\$SEV 10	; Severe
9	000000	M	SC\$FTL 10	; Fatal
10	000000	M	SC\$UNC 20	; Unconditional
11	000000	M	SC\$NDN 40	; None

1	SBTTL	ASCII Character codes
2		
3		-----
4		; Octal values of some ascii characters.
5	000000	M ETX 3 ;ETX (Diablo end-of-buffer command)
6	000000	M CTRLC 3 ;Ctrl-C
7	000000	M ACK 6 ;ACK (Diablo restart-output command)
8	000000	M BELL 7 ;BELL
9	000000	M CTRLG 7 ;Ctrl-G
10	000000	M BKSPAC 10 ;Backspace
11	000000	M TAB 11 ;Horizontal tab
12	000000	M LF 12 ;Line feed
13	000000	M FF 14 ;Form feed
14	000000	M CR 15 ;Carriage return
15	000000	M CTRLD 17 ;Ctrl-D. Discard TT output.
16	000000	M CTRLQ 21 ;Ctrl-Q. Continue TT output.
17	000000	M CTRLR 22 ;Ctrl-R. Retype input line.
18	000000	M CTRLS 23 ;Ctrl-S. Suspend TT output.
19	000000	M CTRLU 25 ;Ctrl-U. Erase input line.
20	000000	M CTRLW 27 ;Ctrl-W. Switch to a virtual line.
21	000000	M CTRLX 30 ;Ctrl-X.
22	000000	M CTRLZ 32 ;Ctrl-Z. TT end of file.
23	000000	M ESC 33 ;Escape
24	000000	M SPACE 40 ;Space
25	000000	M RUBOUT 177 ;Rubout. Erase last character typed.
26	000000	M CSICHR 233 ;Control sequence start character for VT200
27	000000	M SS3CHR 217 ;Control sequence start character for VT200


```

1          .SBTTL  Job Context Area
2          ;-----
3          ; Job Context Area Definition
4          ;
5          140000  $AC      =      CXTBAS      ; Base of Job Context Area
6          ;
7          ; Job stack used in kernel mode, user state.
8 000000  RW      JSTKND  410.
9          ;                                           JSTKND=$AC
10         140000  RW      JSTK      0.
11         141464  ;                                           JSTK=$AC
12         ;
13         ; EMT processing temp cells
14         ;-----
15         ; -- Cells in this region are pushed on EMT entry.
16 000000  RW      EMTCTX  0      ; Start of emt context area that is pushed
17         141464  ;                                           EMTCTX=$AC
18 000000  RW      EMTSP    1      ; EMT processing frame pointer
19         141464  ;                                           EMTSP=$AC
20 000000  RW      URO      1      ; User's RO
21         141466  ;                                           URO=$AC
22 000000  RW      CUREMT   1      ; Current EMT instruction
23         141470  ;                                           CUREMT=$AC
24 000000  RW      EMTBLK   8.     ; Local copy of EMT argument block
25         141472  ;                                           EMTBLK=$AC
26 000000  RW      CHNADR   1      ; Address of channel block we are working on
27         141512  ;                                           CHNADR=$AC
28 000000  RW      CHNNUM   1      ; Number of channel we are working on
29         141514  ;                                           CHNNUM=$AC
30 000000  RW      EMTMAP   1      ; Kernel PAR 5 mapping value when EMT executed
31         141516  ;                                           EMTMAP=$AC
32 000000  RB      EMTErr    1      ; EMT error code
33         141520  ;                                           EMTErr=$AC
34 000000  RB      INTERR   1      ; Internal error code
35         141521  ;                                           INTERR=$AC
36 000000  RW      EMTCTXN   0      ; End of emt context area that is pushed
37         141522  ;                                           EMTCTXN=$AC
38 000000  M      EMTCTXW  <<EMTCTXN-EMTCTX>/2> ; # of words in area to be pushed
39         ; -- End of region that is pushed.
40         ;-----
41 000000  RW      EMTPS     1      ; PS on emt entry
42         141522  ;                                           EMTPS=$AC
43 000000  RW      EMTADR     1      ; PC on emt entry
44         141524  ;                                           EMTADR=$AC
45 000000  RW      EMTLEV     1      ; EMT nesting level
46         141526  ;                                           EMTLEV=$AC
47 000000  RW      EMTASP     1      ; Pointer to arguments on stack
48         141530  ;                                           EMTASP=$AC
49 000000  RW      SPCPS      1      ; Address of user's block assoc with .SPCPS
50         141532  ;                                           SPCPS=$AC
51 000000  RW      EMTCAD     1      ; Pointer to top entry in EMTCAS stack
52         141534  ;                                           EMTCAD=$AC
53 000000  RW      EMTRAD     1      ; In process of exiting from completion routine
54         141536  ;                                           EMTRAD=$AC
55         ; Return stack for completion routines.
56 000000  RW      EMTCAx     4.     ; Reserve space for stack / Keep /

```


38	000000	141540	RW	EMTCAS	0.	; Top of stack	EMTCAX=\$AC / Together / EMTCAS=\$AC
39		141550					
40						; CSI temp cells	
41							
42	000000		RW	CSIBUF	45.	; Holds command line	/ Keep / CSIBUF=\$AC
43	000000	141550	RB	CSIBND	0.	; End of CSIBUF buffer	/ Together / CSIBND=\$AC
44	000000	141702	RW	CSIARE	6.	; Internal emt arg block	CSIARE=\$AC
45	000000	141702	RW	CSIFIL	5.	; Holds a file spec	CSIFIL=\$AC
46	000000	141716	RW	CSIDEV	1	; Default device name	CSIDEV=\$AC
47	000000	141730	RW	CSIUSP	1	; User's stack pointer at start of	CSISPC/GEN CSIUSP=\$AC
48	000000	141732	RB	SPCFLG	1	; 0==>. CSIGEN 1==>. CSISPC	SPCFLG=\$AC
49	000000	141734	RB	SWTCNT	1	; Number of switches accrued	SWTCNT=\$AC
50	000000	141735	RB	CSIEQL	1	; Equal sign seen flag	CSIEQL=\$AC
51		141736					
52						; The following cells contain the current mapping values for the	
53						; user-mode PAR and PDR values for the job.	
54						; The \$MAPOK flag in LSW7 indicates if this data is valid.	
55							
56	000000		RW	CUPARO	1	; User-mode PAR 0	CUPARO=\$AC
57	000000	141740	RW	CUPAR1	1	; User-mode PAR 1	CUPAR1=\$AC
58	000000	141742	RW	CUPAR2	1	; User-mode PAR 2	CUPAR2=\$AC
59	000000	141744	RW	CUPAR3	1	; User-mode PAR 3	CUPAR3=\$AC
60	000000	141746	RW	CUPAR4	1	; User-mode PAR 4	CUPAR4=\$AC
61	000000	141750	RW	CUPAR5	1	; User-mode PAR 5	CUPAR5=\$AC
62	000000	141752	RW	CUPAR6	1	; User-mode PAR 6	CUPAR6=\$AC
63	000000	141754	RW	CUPAR7	1	; User-mode PAR 7	CUPAR7=\$AC
64	000000	141756	RW	CUPDRO	1	; User-mode PDR 0	CUPDRO=\$AC
65	000000	141760	RW	CUPDR1	1	; User-mode PDR 1	CUPDR1=\$AC
66	000000	141762	RW	CUPDR2	1	; User-mode PDR 2	CUPDR2=\$AC
67	000000	141764	RW	CUPDR3	1	; User-mode PDR 3	CUPDR3=\$AC
68	000000	141766	RW	CUPDR4	1	; User-mode PDR 4	CUPDR4=\$AC
69	000000	141770	RW	CUPDR5	1	; User-mode PDR 5	CUPDR5=\$AC
		141772					

Line	Address	Operation	Symbol	Value	Description	Symbol
70	000000	RW	CUPDR6	1	; User-mode PDR 6	CUPDR6=\$AC
71	000000	RW	CUPDR7	1	; User-mode PDR 7	CUPDR7=\$AC
72						
73					; Data cells related to shared run-time system support.	
74						
75	000000	RW	CURRDB	1	; Address of current run-time descriptor block	CURRDB=\$AC
76	000000	RW	RPDR	8.	; PAR values for job	RPDR=\$AC
77	000000	RW	RPDR	8.	; PDR values for job	RPDR=\$AC
78	000000	RW	RPDRND	0	; End of RPAR-RPDR tables	RPDRND=\$AC
79						
80					; PLAS Region Control Blocks and Window Control Blocks	
81						
82	000000	RW	RCBBAS	NUMRCB*<RCB\$\$SZ/2>	; Region control blocks for job	RCBBAS=\$AC
83	000000	RW	RCBEND	0	; End of RCB area	RCBEND=\$AC
84	000000	RW	WCBAS	NUMWCB*<WCB\$\$SZ/2>	; Window control blocks for job	WCBAS=\$AC
85	000000	RW	WCBEND	0	; End of WCB area	WCBEND=\$AC
86	000000	M	RUNRDB	CSIBUF	; Region Descriptor Block used during job start	
87						
88					; Shared run-time region descriptor tables	
89						
90	000000	RW	SR\$PAR	MAXSRD	; Value to load into PAR register	SR\$PAR=\$AC
91	000000	RW	SR\$PDR	MAXSRD	; Value to load into PDR register	SR\$PDR=\$AC
92	000000	RB	SR\$PX	MAXSRD	; Index # of which PAR to load	SR\$PX=\$AC
93						
94					; Assign table	
95						
96	000000	RW	ASNTBL	<MAXASN*<CAT\$\$SZ/2>>	; ASSIGN information	ASNTBL=\$AC
97	000000	RW	ASNEND	0	; End of table	ASNEND=\$AC
98						
99					; Data related to the ACCESS command	
100						
101	000000	RW	RESDEV	1	; Non-zero ==> Some ACCESS controls	RESDEV=\$AC
102	000000	RW	OKFILE	<MAXACC*<OF\$\$SZ/2>>		OKFILE=\$AC
103	000000	RW	OKFEND	0		OKFEND=\$AC
104						
105					; Data related to logical disks	
106						
107	000000	RW	LDNAME	4*MAXLD	; File spec for logical disk file (rad50)	LDNAME=\$AC

108 000000		RW	LDPDEV	MAXLD	; Physical device index # and unit #	
	144164				LDPDEV=\$AC	
109 000000		RW	LDSIZE	MAXLD	; Number of blocks in logical disk	
	144204				LDSIZE=\$AC	
110 000000		RW	LDBASE	MAXLD	; Base block # on real disk of log disk start	
	144224				LDBASE=\$AC	
111 000000		RW	LDFLAG	MAXLD	; LD\$xxx status flags	
	144244				LDFLAG=\$AC	
112						
113					; Data regarding job privileges	
114						
115 000000		RW	PRIVA0	1	; Authorized privileges	
	144264				PRIVA0=\$AC	
116 000000		RW	PRIVA2	1		
	144266				PRIVA2=\$AC	
117 000000		RW	PRIVS0	1	; Privileges as controlled by SET command	
	144270				PRIVS0=\$AC	
118 000000		RW	PRIVS2	1		
	144272				PRIVS2=\$AC	
119 000000		RW	PRIVF0	1	; Privileges for current command file	
	144274				PRIVF0=\$AC	
120 000000		RW	PRIVF2	1		
	144276				PRIVF2=\$AC	
121 000000		RW	PRIVC0	1	; Combined, current privileges for job	
	144300				PRIVC0=\$AC	
122 000000		RW	PRIVC2	1		
	144302				PRIVC2=\$AC	
123						
124					; Data regarding command file usage.	
125						
126 000000		RW	CFBUF	256.	; Command file buffer	
	144304				CFBUF=\$AC	
127 000000		RW	CFEND	0.	; End of command file buffer	
	145304				CFEND=\$AC	
128 000000		RW	CFPNT	1	; Index into command file buffer	
	145304				CFPNT=\$AC	
129 000000		RW	CFBLK	1	; Next command file block number	
	145306				CFBLK=\$AC	
130 000000		RW	CFSPND	1	; Suspended command file pointer	
	145310				CFSPND=\$AC	
131 000000		RW	CFIND	1	; Holds IND status flags	
	145312				CFIND=\$AC	
132 000000		RW	CURPRM	1	; Parameter string pointer	
	145314				CURPRM=\$AC	
133 000000		RW	PRMPNT	MXCPRM	; Pointers to command parameter strings	
	145316				PRMPNT=\$AC	
134 000000		RW	LSTPRM	0	; End of PRMPNT vector	
	145332				LSTPRM=\$AC	
135 000000		RB	PRMBUF	MXCCHR	; Buffer to hold command file parameter string	
	145332				PRMBUF=\$AC	
136 000000		RW	PRMEND	1	; End of PRMBUF	
	145426				PRMEND=\$AC	
137 000000		RW	PBFEND	1	; Pointer to end of parameter string	
	145430				PBFEND=\$AC	
138 000000		RW	CFARG	5	; Used for I/O argument block for command file reads	
	145432				CFARG=\$AC	
139					; Command file push stack	

140 000000		RW	CFSEND	120.	; End of stack	
	145444					CFSEND=\$AC
141 000000		RW	CFSTK	0	; Start of stack	
	146024					CFSTK=\$AC
142 000000		RW	CFSP	1	; Pointer into stack	
	146024					CFSP=\$AC
143						
144					; Data cells related to log file	
145						
146 000000		RW	LOGBUF	256.	; Log file buffer	** Keep **
	146026					LOGBUF=\$AC
147 000000		RW	LOGEND	0	; End of log file buffer	** Together **
	147026					LOGEND=\$AC
148 000000		RW	LOGPTR	1	; Current pointer into log file buffer	
	147026					LOGPTR=\$AC
149 000000		RW	LOGBLK	1	; Number of next block to be written	
	147030					LOGBLK=\$AC
150 000000		RW	LOGFLG	1	; Log file control flags (see LF\$xxx flags)	
	147032					LOGFLG=\$AC
151 000000		RW	LOGDVU	1	; Physical dev # and unit # of logging device	
	147034					LOGDVU=\$AC
152 000000		RW	LOGBAS	1	; Base block # of log file subdevice	
	147036					LOGBAS=\$AC
153						
154					; Data cells related to Single Line Editor	
155						
156 000000		RW	SLCX	1	; Pointer to character in SLEBUF under cursor	
	147040					SLCX=\$AC
157 000000		RW	SLCCOL	1	; Display column position of cursor	
	147042					SLCCOL=\$AC
158 000000		RW	SLECOL	1	; Display column position of end of line	
	147044					SLECOL=\$AC
159 000000		RW	SLSCOL	1	; Display column position of start of line	
	147046					SLSCOL=\$AC
160 000000		RW	SLOPTR	1	; Pointer to next char to return to program	
	147050					SLOPTR=\$AC
161 000000		RW	SLSPTR	1	; Pointer to most recently saved line	
	147052					SLSPTR=\$AC
162 000000		RW	SLLPTR	1	; Pointer to next saved line to recall	
	147054					SLLPTR=\$AC
163 000000		RW	SLCYC1	1	; Pointer to 1st command in cycle	
	147056					SLCYC1=\$AC
164 000000		RW	SLCYC2	1	; Pointer to last command in cycle	
	147060					SLCYC2=\$AC
165 000000		RW	SLRPTR	1	; Pointer to pending recall line	
	147062					SLRPTR=\$AC
166 000000		RW	SLCSPT	1	; Pointer to next char position in SLCSBF	
	147064					SLCSPT=\$AC
167 000000		RW	SLCSR	1	; Address of routine to process next char	
	147066					SLCSR=\$AC
168 000000		RB	SLEBUF	SLMXLN+1	; Line currently being edited	
	147070					SLEBUF=\$AC
169 000000		RB	SLLBUF	SLBFSZ	; Previous lines (recalled by use of up-arrow)	
	147211					SLLBUF=\$AC
170 000000		RB	SLEEND	0	; End of SLLBUF -must immediately follow SLLBUF	
	147665					SLEEND=\$AC
171 000000		RB	SLDBUF	SLMXLN+1	; Last deleted word or portion of line	

172	000000	147665	RB	SLSBUF	SLMXLN+1	; Line saved with gold-down-arrow key	SLDBUF=\$AC
		150006					SLSBUF=\$AC
173	000000	150127	RB	SLCBUF	2	; Last deleted character	SLCBUF=\$AC
174	000000	150131	RB	SLCSBF	6	; Buffer used to accrue terminal control seq	SLCSBF=\$AC
175	000000	150137	RB	SLCSBX	0	; End of SLCSBF -must immediately follow SLCSBF	SLCSBX=\$AC
176	000000	150137	RB	SLGOLD	1	; Non-zero==>Gold key (PF1) was pressed	SLGOLD=\$AC
177	000000	150140	RB	SLCR	1	; Carriage-return was last character, need LF	SLCR=\$AC
178	000000	150141	RB	SLBACK	1	; Non-zero==>In reverse direction mode	SLBACK=\$AC
179	000000	150142	RB	SLDOWN	1	; Non-zero==>Last recall was in down direction	SLDOWN=\$AC
180	000000	150143	RB	SLOVER	1	; Non-zero==>Overstrike mode	SLOVER=\$AC
181	000000	150144	RB	RCLREV	1	; Non-zero==>Display recall/all reverse order	RCLREV=\$AC
182							
183						; Data cells related to user-defined keys	
184							
185	000000	150146	RW	KEYRCB	1	; Address of region control block for key defs	KEYRCB=\$AC
186	000000	150150	RW	KEYPAR	1	; Address to use to map a PAR to key region	KEYPAR=\$AC
187							
188						; Data cells related to spooling.	
189							
190	000000	150152	RW	SXSFCB	1	; SFCB that is using spool buffer	SXSFCB=\$AC
191	000000	150154	RW	SXBUF1	1		SXBUF1=\$AC
192	000000	150156	RW	SPUBUF	256.	; COOP buffer	SPUBUF=\$AC
193	000000	151156	RW	SPUBND	0	; End of COOP buffer	SPUBND=\$AC
194	000000	151156	RW	SXBPNT	1	; Pointer into spool buffer	SXBPNT=\$AC
195	000000	151160	RW	SPLARG	5	; Used for spool EMT arg block	SPLARG=\$AC
196	000000	151172	RW	SPDLBF	1	; Used when deleting a spool file	SPDLBF=\$AC
197	000000	151174	RW	UFORM	3	; Current spool form name	UFORM=\$AC
198							
199						; Data cells for TSDEBUG debugging system	
200							
201	000000	151202	RW	D. START	0	; Start of debugging data area	D. START=\$AC
202	000000	151202	RW	D. R0	1	; User's R0	D. R0=\$AC
203	000000	151204	RW	D. R1	1	; User's R1	D. R1=\$AC
204	000000		RW	D. R2	1	; User's R2	

205	000000	151206	RW	D. R3	1	; User's R3	D. R2=\$AC
		151210					D. R3=\$AC
206	000000		RW	D. R4	1	; User's R4	D. R4=\$AC
		151212					D. R5=\$AC
207	000000		RW	D. R5	1	; User's R5	D. R5=\$AC
		151214					D. R6=\$AC
208	000000		RW	D. R6	1	; User's R6 (SP)	D. R6=\$AC
		151216					D. R7=\$AC
209	000000		RW	D. R7	1	; User's R7 (PC)	D. R7=\$AC
		151220					D. PS=\$AC
210	000000		RW	D. PS	1	; User's PSW	D. PS=\$AC
		151222					D. SPSV=\$AC
211	000000		RW	D. SPSV	1	; Saved initial stack pointer	D. SPSV=\$AC
		151224					D. LOC=\$AC
212	000000		RW	D. LOC	1	; Address of currently open location	D. LOC=\$AC
		151226					D. DADR=\$AC
213	000000		RW	D. DADR	1	; Address of monitored data word	D. DADR=\$AC
		151230					D. DOLD=\$AC
214	000000		RW	D. DOLD	1	; Old value of monitored data word	D. DOLD=\$AC
		151232					D. DTRG=\$AC
215	000000		RW	D. DTRG	1	; Target value for monitored data word	D. DTRG=\$AC
		151234					D. MASK=\$AC
216	000000		RW	D. MASK	1	; Mask used for data monitoring (\$M register)	D. MASK=\$AC
		151236					D. PFMT=\$AC
217	000000		RW	D. PFMT	1	; Printout format (\$F register)	D. PFMT=\$AC
		151240					D. LVAL=\$AC
218	000000		RW	D. LVAL	1	; Last displayed value	D. LVAL=\$AC
		151242					D. VAL1=\$AC
219	000000		RW	D. VAL1	1	; Command argument value 1	D. VAL1=\$AC
		151244					D. VAL2=\$AC
220	000000		RW	D. VAL2	1	; Command argument value 2	D. VAL2=\$AC
		151246					D. FLAG=\$AC
221	000000		RW	D. FLAG	1	; D\$xxxx flags (see below)	D. FLAG=\$AC
		151250					D. PCNT=\$AC
222	000000		RW	D. PCNT	1	; Proceed repeat count	D. PCNT=\$AC
		151252					D. PCOL=\$AC
223	000000		RW	D. PCOL	1	; Current print column counter	D. PCOL=\$AC
		151254					D. ILEN=\$AC
224	000000		RW	D. ILEN	1	; Number of bytes used by current instruction	D. ILEN=\$AC
		151256					D. RLBS=\$AC
225	000000		RW	D. RLBS	8.	; Relocation base offsets	D. RLBS=\$AC
		151260					D. BKAD=\$AC
226	000000		RW	D. BKAD	9.	; Breakpoint addresses	D. BKAD=\$AC
		151300					D. BKSV=\$AC
227	000000		M	D. CBRK	D. BKAD+16.	; Address of temp breakpoint following a CALL	
228	000000		RW	D. BKSV	9.	; Instruction saved from breakpoint location	D. BKSV=\$AC
		151322					D. BKNM=\$AC
229	000000		RB	D. BKNM	1	; Current breakpoint number	D. BKNM=\$AC
		151344					D. V1FL=\$AC
230	000000		RB	D. V1FL	1	; Flag indicating if arg value 1 specified	D. V1FL=\$AC
		151345					D. V2FL=\$AC
231	000000		RB	D. V2FL	1	; Flag indicating if arg value 2 specified	D. V2FL=\$AC
		151346					D. LOCM=\$AC
232	000000		RB	D. LOCM	1	; Mode of currently open location	D. LOCM=\$AC
		151347					
233	000000		RB	D. SVCH	1	; "Pushed" input character	

234 000000	151350	RB	D. BYTM	1	; Word/Byte mode indicator	D. SVCH=\$AC
235 000000	151351	RB	D. NMBF	8	; Buffer used to hold numeric values	D. BYTM=\$AC
236 000000	151352	RB	D. NMBE	0	; End of D. NMBF	D. NMBF=\$AC
237 000000	151362	RW	D. END	0	; End of debugging data area	D. NMBE=\$AC
238	151362					D. END=\$AC
239					; Debugger control flags stored in D. FLAG word	
240						
241 000000		M	D\$SSTP	1	; Doing single stepping	
242 000000		M	D\$IBRK	2	; An instruction breakpoint occurred	
243 000000		M	D\$DBRK	4	; A data breakpoint occurred	
244 000000		M	D\$DMON	10	; Data cell monitoring in effect	
245 000000		M	D\$DVAL	20	; Target value specified for monitored data	
246 000000		M	D\$IPND	40	; Replaced breakpoint instruction pending	
247 000000		M	D\$SBRK	100	; Single step breakpoint occurred	
248 000000		M	D\$FBRK	200	; Forced breakpoint (user typed ctrl-B)	
249 000000		M	D\$BKST	400	; Breakpoints are in place in program	
250 000000		M	D\$CKBK	1000	; Some instruction breakpoints are specified	
251 000000		M	D\$INIT	2000	; Debugger initialization has been done	
252 000000		M	D\$TSTP	4000	; Single step one instruction	
253 000000		M	D\$RUN	100000	; Debugger program is executing now	
254						
255					; Print format control flags stored in the D. PFMT register	
256						
257 000000		M	DP\$DAA	1	; Print absolute addresses for decoded instruct	
258 000000		M	DP\$LAA	2	; Print abs addresses for locations	
259						
260					; Misc. parameters	
261						
262 000000	151362	RW	UPPN	2	; Project - Programmer number	UPPN=\$AC
263 000000	151366	RW	RUNDEV	4	; Name of device and program that is currently running	RUNDEV=\$AC
264 000000	151376	RW	RUNFLG	1	; AF\$xxx flags for program currently running	RUNFLG=\$AC
265 000000	151400	RW	AFCF	1	; AF\$xxx flags for current command file	AFCF=\$AC
266 000000	151402	RW	JCDB	2	; Point to 1st shared file CDB block for job	JCDB=\$AC
267 000000	151406	RW	ERRSPC	4	; Name of file to print with KMON err message	ERRSPC=\$AC
268 000000	151416	RW	DEVLS	1	; Address of non-linked .DEVICE reset list	DEVLS=\$AC
269 000000	151420	RW	DEVLL	1	; Address of linked .DEVICE reset list	DEVLL=\$AC
270 000000	151422	RW	UHIMEM	1	; Max virtual address assigned to job	UHIMEM=\$AC
271 000000	151424	RW	USRSTK	1	; Initial user stack pointer	USRSTK=\$AC
272 000000	151426	RW	SPSAVE	1		SPSAVE=\$AC
273 000000	151430	RW	USTART	1		USTART=\$AC

274 000000		RW	ODTBAS	1	; Addr of top of memory available to job
	151432				ODTBAS=\$AC
275 000000		RW	NEWJSW	1	; JSW for SAV file being started
	151434				NEWJSW=\$AC
276 000000		RW	UMSPSV	1	; User-mode SP saved by PKSTAT (0==>SP active)
	151436				UMSPSV=\$AC
277 000000		RW	MAXMEM	1	
	151440				MAXMEM=\$AC
278 000000		RW	PRGTOP	1	; Top address of program about to be started
	151442				PRGTOP=\$AC
279 000000		RW	PRGSIZ	1	; Address above top of total area for program
	151444				PRGSIZ=\$AC
280 000000		RW	UTRPAD	1	
	151446				UTRPAD=\$AC
281 000000		RW	UFPTRP	1	
	151450				UFPTRP=\$AC
282 000000		RW	LSTFDT	1	; File directory time entry from last .LOOKUP
	151452				LSTFDT=\$AC
283 000000		RW	LSTFDD	1	; File directory date entry from last .LOOKUP
	151454				LSTFDD=\$AC
284 000000		RW	JOBCCB	1	; Active cache control blocks for job
	151456				JOBCCB=\$AC
285 000000		RW	NPCCB	1	; Number of pending cache control blocks
	151460				NPCCB=\$AC
286 000000		RW	UCHAN	1	; CDFN channel space
	151462				UCHAN=\$AC
287 000000		RW	CINDAT	<<1000-500>/2>	; Save area for .chain data
	151464				CINDAT=\$AC
288 000000		RW	ABRTAD	1	
	151764				ABRTAD=\$AC
289 000000		RW	UCISPC	4	; User Command Interface (UCI) file spec
	151766				UCISPC=\$AC
290 000000		RW	LOFSPC	4	; Logoff command file spec
	151776				LOFSPC=\$AC
291 000000		RW	MXJPRI	1	; Max priority allowed for this job
	152006				MXJPRI=\$AC
292 000000		RW	JPWDEV	1	; Device to use for print-window function
	152010				JPWDEV=\$AC
293 000000		RW	JPWTYP	1	; Print-window device type
	152012				JPWTYP=\$AC
294 000000		RW	JPWFLG	1	; Print-window control flags (PA\$xxx)
	152014				JPWFLG=\$AC
295 000000		RW	KMPRMT	<<MXPRMT+2>/2>	; String to use for KMDN command prompt
	152016				KMPRMT=\$AC
296 000000		RW	SBPSUF	8.	; Start-up command file name for subprocesses
	152030				SBPSUF=\$AC
297					
298					
299					
300 000000		RB	CINFLG	1	; Chain in progress
	152050				CINFLG=\$AC
301 000000		RB	RUNARG	1	; RUN command arg string pending for chain data
	152051				RUNARG=\$AC
302 000000		RB	LSTCHR	1	
	152052				LSTCHR=\$AC
303 000000		RB	ABRTCD	1	
	152053				ABRTCD=\$AC

Job Context Area

304 000000	152054	RB	SERFLG	1		SERFLG=\$AC
305 000000	152055	RB	CFNEST	1	; Command file nesting level	CFNEST=\$AC
306 000000	152056	RB	ERRSEV	1		ERRSEV=\$AC
307 000000	152057	RB	UERSEV	1	; User specified error severity level (from 53)	UERSEV=\$AC
308 000000	152060	RB	CURCP	1	; Non-zero ==> executing a completion routine	CURCP=\$AC
309 000000	152061	RB	CFHOLD	1	; Look-ahead command file char being held	CFHOLD=\$AC
310 000000	152062	RB	SPIJ	1	; Index of job that initiated subprocess	SPIJ=\$AC
311 000000	152063	RB	SUCF2	SF2LEN	; Name of secondary start-up command file	SUCF2=\$AC
312 000000	152103	RB	DOTRMP	1	; Non-zero ==> use TRAP for mapping	DOTRMP=\$AC
313						
314						
315 000000	152104	RW	CXTEND	0	; End of Job Context Area	CXTEND=\$AC
316						
317 000000		M	CXTSIZ	<CXTEND-CXTBAS>	; Size of job context area	
318	000001					

Errors detected: 0

*** Assembler statistics

Work file reads: 0
Work file writes: 0
Size of work file: 8325 Words (33 Pages)
Size of core pool: 17920 Words (70 Pages)
Operating system: RT-11

Elapsed time: 00:03:05.29

DK: TSDEFS, LP: TSDEFS=DK: TSDEFS. MAC/C/N: SYM

\$1CTL	6-9#											
\$1ESC	2-15#											
\$1STCH	4-13#											
\$1STLG	7-5#											
\$8BIT	3-7#	3-35										
\$AC	14-7#	14-8	14-12	14-13	14-17	14-18	14-19	14-20	14-21	14-22	14-23	14-24
	14-28	14-29	14-30	14-34	14-35	14-36	14-37	14-38	14-39	14-40	14-41	14-42
	14-43	14-44	14-45	14-46	14-47	14-48	14-49	14-50	15-5#	15-6	15-7	15-8
	15-9	15-10	17-5#	17-6	17-7	17-8	17-9	18-5#	18-6	18-7	18-8	18-9
	18-10	18-11	18-12	19-5#	19-6	19-7	19-8	19-9	19-10	19-12	19-13	19-14
	19-15	19-16	19-17	19-18	19-19	19-20	19-21	19-22	19-23	19-24	19-25	19-26
	19-27	19-28	21-6#	21-7	21-8	21-9	21-10	21-11	21-12	21-13	22-6#	22-7
	22-8	22-9	22-10	22-11	23-8#	23-9	23-10	23-12	24-6#	24-7	24-8	24-9
	24-10	25-7#	25-8	25-9	25-10	25-11	25-12	25-13	25-14	25-15	25-16	25-17
	25-18	25-19	25-20	25-21	25-22	25-23	25-24	25-25	25-26	25-27	25-28	25-29
	25-30	25-31	25-32	25-33	25-34	25-35	25-36	25-38	28-11#	28-12	28-13	28-14
	28-15	28-16	29-5#	29-6	29-7	29-8	29-9	29-10	29-11	29-12	29-21#	29-22
	29-23	29-24	29-25	29-26	29-27	29-28	30-10#	30-11	30-12	30-13	30-17#	30-18
	30-19	30-20	30-24#	30-25	30-26	31-5#	31-6	31-7	31-8	31-9	31-10	31-11
	31-12	31-13	31-14	31-34#	31-35	31-36	31-37	31-38	32-4#	32-5	32-6	32-7
	32-8	32-9	32-10	32-11	32-12	32-13	32-18#	32-19	32-20	32-21	32-22	32-23
	32-24	32-25	32-26	32-27	33-8#	33-9	33-10	33-11	33-12	33-13	34-11#	34-12
	34-13	34-14	34-15	34-16	34-17	34-18	34-19	34-20	34-21	34-27#	34-28	34-29
	34-30	35-6#	35-7	35-8	35-9	35-10	35-11	35-12	35-13	35-14	35-15	35-17
	36-7#	36-8	36-9	36-10	36-11	36-12	36-13	36-14	36-15	39-33#	39-34	39-35
	39-36	39-37	39-38	40-5#	40-6	40-7	40-8	40-9	40-10	40-12	40-13	40-14
	40-21	40-22	40-24	41-4#	41-5	41-6	41-7	41-8	41-9	41-10	41-29#	41-30
	41-31	41-32	41-33	41-34	41-35	42-6#	42-7	42-8	42-9	42-10	42-11	42-12
	42-19#	42-20	42-21	42-22	42-23	42-24	42-36#	42-37	42-38	42-39	42-40	43-8#
	43-9	43-10	43-11	43-12	43-13	43-14	43-15	43-16	43-17	43-18	43-19	43-20
	44-5#	44-6	44-7	44-8	44-9	44-10	47-8#	47-9	47-10	47-11	47-12	47-13
	47-14	47-15	47-16	47-17	47-18	47-19	47-20	48-7#	48-8	48-9	48-10	48-11
	48-12	48-13	48-14	48-15	48-16	48-17	48-18	48-19	48-20	48-21	48-22	48-23
	48-24	49-5#	49-6	49-7	49-8	49-9	49-10	49-11	49-12	49-13	49-14	49-15
	49-16	49-17	49-18	51-7#	51-8	51-9	51-10	51-11	51-12	51-13	51-14	51-15
	51-16	51-18	52-6#	52-7	52-8	52-9	52-10	52-11	52-12	52-13	52-14	52-25#
	52-26	52-27	52-28	52-29	53-6#	53-7	53-8	53-9	53-10	53-11	53-12	53-13
	54-5#	54-6	54-7	54-8	54-9	54-10	54-14#	54-15	54-16	54-17	54-18	54-19
	54-23#	54-24	54-25	54-26	54-27	54-28	54-29	54-30	54-34#	54-35	54-36	54-37
	54-38	67-5#	67-8	67-9	67-15	67-16	67-17	67-18	67-19	67-20	67-21	67-22
	67-23	67-24	67-25	67-29	67-30	67-31	67-32	67-33	67-34	67-35	67-37	67-38
	67-42	67-43	67-44	67-45	67-46	67-47	67-48	67-49	67-50	67-56	67-57	67-58
	67-59	67-60	67-61	67-62	67-63	67-64	67-65	67-66	67-67	67-68	67-69	67-70
	67-71	67-75	67-76	67-77	67-78	67-82	67-83	67-84	67-85	67-90	67-91	67-92
	67-96	67-97	67-101	67-102	67-103	67-107	67-108	67-109	67-110	67-111	67-115	67-116
	67-117	67-118	67-119	67-120	67-121	67-122	67-126	67-127	67-128	67-129	67-130	67-131
	67-132	67-133	67-134	67-135	67-136	67-137	67-138	67-140	67-141	67-142	67-146	67-147
	67-148	67-149	67-150	67-151	67-152	67-156	67-157	67-158	67-159	67-160	67-161	67-162
	67-163	67-164	67-165	67-166	67-167	67-168	67-169	67-170	67-171	67-172	67-173	67-174
	67-175	67-176	67-177	67-178	67-179	67-180	67-181	67-185	67-186	67-190	67-191	67-192
	67-193	67-194	67-195	67-196	67-197	67-201	67-202	67-203	67-204	67-205	67-206	67-207
	67-208	67-209	67-210	67-211	67-212	67-213	67-214	67-215	67-216	67-217	67-218	67-219
	67-220	67-221	67-222	67-223	67-224	67-225	67-226	67-228	67-229	67-230	67-231	67-232
	67-233	67-234	67-235	67-236	67-237	67-262	67-263	67-264	67-265	67-266	67-267	67-268
	67-269	67-270	67-271	67-272	67-273	67-274	67-275	67-276	67-277	67-278	67-279	67-280
	67-281	67-282	67-283	67-284	67-285	67-286	67-287	67-288	67-289	67-290	67-291	67-292

[illegible]

\$5003 9-10# 9-16

[illegible]

AF\$CCA	44-28#	
AF\$DBG	44-21#	
AF\$DUP	44-24#	
AF\$HIE	44-15#	
AF\$IND	44-25#	
AF\$IOP	44-17#	
AF\$MEM	44-19#	
AF\$NOI	44-16#	
AF\$NOW	44-14#	
AF\$NPW	44-29#	
AF\$PLK	44-20#	
AF\$SCA	44-18#	
AF\$SET	44-27#	
AF\$TPO	44-23#	
AF\$UCL	44-26#	
AFCF	67-265#	
AL\$DHB	26-4#	
AL\$DHT	26-5#	
AL\$DWD	26-6#	
AR\$\$SZ	49-18#	49-20
AR\$CNT	49-12#	
AR\$CON	49-11#	
AR\$CPH	49-13#	
AR\$CPL	49-14#	
AR\$DMY	49-17#	
AR\$PRG	49-7#	
AR\$PRI	49-16#	
AR\$PRJ	49-6#	
AR\$PRV	49-9#	
AR\$PWD	49-8#	
AR\$SUF	49-10#	
AR\$UNM	49-15#	
ARNRPB	49-20#	
ASNEND	67-97#	
ASNTBL	67-96#	
AT\$\$SZ	42-12#	
AT\$DEV	42-9#	
AT\$EXT	42-11#	
AT\$FIL	42-10#	
AT\$LOG	42-7#	
AT\$SIZ	42-8#	
AW\$132	26-20#	
AW\$200	26-19#	
AW\$52	26-18#	
AW\$ACK	26-22#	
AW\$AKM	26-25#	
AW\$DDC	26-32#	
AW\$INS	26-21#	
AW\$ORS	26-24#	
AW\$PRM	26-33#	
AW\$PRT	26-30#	
AW\$REV	26-23#	
AW\$RPT	26-29#	
AW\$S52	26-28#	
AW\$SPN	26-31#	
AW\$SS	26-27#	

AW\$VCR	26-26#
BELL	66-8#
BKSPAC	66-10#
BLKWDS	64-31#
C. CSW	18-6#
C. DEVQ	18-11#
C. LENG	18-8#
C. NUMQ	18-10#
C. SBLK	18-7#
C. USED	18-9#
CARDET	55-21#
CC\$SZ	35-17#
CC\$BLK	35-8#
CC\$CBP	35-14#
CC\$DVU	35-9#
CC\$LNK	35-7#
CC\$QGE	35-11#
CC\$UBD	35-13#
CC\$UBP	35-12#
CC\$WCT	35-10#
CC\$WFL	35-15#
CCFLG	64-35#
CD\$SZ	41-35#
CD\$UB	41-27#
CD\$BAS	41-31#
CD\$DVU	41-30#
CD\$JOB	41-34#
CD\$NAM	41-33#
CD\$TOP	41-32#
CF\$IND	16-72#
CF\$QUT	16-73#
CFACFL	16-61#
CFARG	67-138#
CFBLK	67-129#
CFBUF	67-126#
CFEND	67-127#
CFHOLD	67-309#
CFIND	67-131#
CFLAG	64-40#
CFLFL4	5-22#
CFNEST	67-305#
CFPNT	67-128#
CFSEND	67-140#
CFSP	67-142#
CFSPND	67-130#
CFSTK	67-141#
CHAIN	63-12#
CHNADR	67-20#
CHNNUM	67-21#
CHNSIZ	18-12#
CINDAT	67-287#
CINFLG	67-300#
CLSFAB	60-60#
CLSFBC	60-49#
CLSFCH	60-48#
CLSFCD	60-54#

CLSFDL	60-52#	
CLSFEF	60-65#	
CLSFGB	60-67#	
CLSFHS	60-51#	
CLSFIC	60-62#	
CLSFMS	60-58#	
CLSFOD	60-63#	
CLSFRB	60-50#	
CLSFRL	60-61#	
CLSFRS	60-66#	
CLSFSL	60-55#	
CLSFSD	60-53#	
CLSFSP	60-59#	
CLSFSS	60-56#	
CLSFWS	60-57#	
CLSFWB	60-64#	
CM\$BRK	60-30#	
CM\$CRL	60-26#	
CM\$DTR	60-34#	
CM\$EFP	60-25#	
CM\$EOF	60-28#	
CM\$FFI	60-36#	
CM\$FFS	60-31#	
CM\$IRG	60-32#	
CM\$MCC	60-35#	
CM\$ON	60-29#	
CM\$ORP	60-33#	
CM\$TBS	60-27#	
CM\$WRT	60-24#	
CO\$BBT	60-18#	
CO\$BNI	60-14#	
CO\$BND	60-13#	
CO\$CR	60-15#	60-20
CO\$CTL	60-16#	60-20
CO\$DEF	60-20#	
CO\$DTR	60-17#	
CO\$FF	60-7#	
CO\$FFO	60-12#	
CO\$LC	60-9#	60-20
CO\$LFI	60-11#	60-20
CO\$LFO	60-10#	60-20
CO\$TAB	60-8#	
CPLMT	64-23#	
CQ\$CP	20-19#	
CQ\$FLG	20-17#	
CQ\$HOT	20-9#	
CQ\$JOB	20-11#	
CQ\$LNK	20-8#	
CQ\$LOT	20-10#	
CQ\$PA5	20-16#	
CQ\$PRI	20-18#	
CQ\$RO	20-13#	
CQ\$R1	20-14#	
CQ\$RNS	20-12#	
CQ\$RTN	20-15#	
CR	66-14#	

CS\$ENT	18-20#	
CS\$EOF	18-18#	
CS\$ERR	18-23#	
CS\$NMX	18-22#	
CS\$OPN	18-16#	
CS\$RON	18-17#	
CS\$SEG	18-19#	
CS\$SPL	18-21#	
CSIARE	67-44#	
CSIBND	67-43#	
CSIBUF	67-42#	67-86
CSICHR	66-26#	
CSIDEV	67-46#	
CSIEQL	67-50#	
CSIFIL	67-45#	
CSIUSP	67-47#	
CTRLC	66-6#	
CTRLG	66-9#	
CTRLD	66-15#	
CTRLQ	66-16#	
CTRLR	66-17#	
CTRLS	66-18#	
CTRLU	66-19#	
CTRLW	66-20#	
CTRLX	66-21#	
CTRLZ	66-22#	
CTTBR	64-46#	
CTTSR	64-45#	
CUPARO	67-56#	
CUPAR1	67-57#	
CUPAR2	67-58#	
CUPAR3	67-59#	
CUPAR4	67-60#	
CUPAR5	67-61#	
CUPAR6	67-62#	
CUPAR7	67-63#	
CUPDRO	67-64#	
CUPDR1	67-65#	
CUPDR2	67-66#	
CUPDR3	67-67#	
CUPDR4	67-68#	
CUPDR5	67-69#	
CUPDR6	67-70#	
CUPDR7	67-71#	
CURCP	67-308#	
CUREMT	67-18#	
CURPRM	67-132#	
CURRDB	67-75#	
CW\$50H	16-9#	
CW\$60	16-34#	
CW\$70	16-33#	
CW\$BTH	16-7#	
CW\$CIS	16-29#	
CW\$CSH	16-22#	
CW\$CSR	16-16#	
CW\$EIS	16-30#	

CW\$ESP	16-27#		
CW\$FB	16-5#		
CW\$FGJ	16-11#		
CW\$FPU	16-10#		
CW\$GDH	16-6#		
CW\$KWP	16-17#		
CW\$LGS	16-12#		
CW\$LPC	16-18#		
CW\$LSI	16-14#		
CW\$PAR	16-23#		
CW\$PRO	16-32#		
CW\$QBS	16-28#		
CW\$RLH	16-26#		
CW\$RSR	16-24#		
CW\$SLE	16-8#		
CW\$USR	16-13#		
CW\$V60	16-31#		
CW\$WCD	16-25#		
CW\$XM	16-15#		
CXTBAS	64-20#	67-5	67-317
CXTEND	67-315#	67-317	
CXTSIZ	67-317#		
D\$BKST	67-249#		
D\$CKBK	67-250#		
D\$DBRK	67-243#		
D\$DMON	67-244#		
D\$DVAL	67-245#		
D\$FBRK	67-248#		
D\$IBRK	67-242#		
D\$INIT	67-251#		
D\$IPND	67-246#		
D\$RUN	67-253#		
D\$SBRK	67-247#		
D\$SSTP	67-241#		
D\$TSTP	67-252#		
D. BKAD	67-226#	67-227	
D. BKNM	67-229#		
D. BKSV	67-228#		
D. BYTM	67-234#		
D. CBRK	67-227#		
D. DADR	67-213#		
D. DOLD	67-214#		
D. DTRG	67-215#		
D. END	67-237#		
D. FLAG	67-221#		
D. ILEN	67-224#		
D. LOC	67-212#		
D. LOCM	67-232#		
D. LVAL	67-218#		
D. MASK	67-216#		
D. NMBE	67-236#		
D. NMBF	67-235#		
D. PCNT	67-222#		
D. PCOL	67-223#		
D. PFMT	67-217#		
D. PS	67-210#		

D. R0	67-202#	
D. R1	67-203#	
D. R2	67-204#	
D. R3	67-205#	
D. R4	67-206#	
D. R5	67-207#	
D. R6	67-208#	
D. R7	67-209#	
D. RLBS	67-225#	
D. SPSV	67-211#	
D. STAR	67-201#	
D. SVCH	67-233#	
D. V1FL	67-230#	
D. V2FL	67-231#	
D. VAL1	67-219#	
D. VAL2	67-220#	
DATVAL	56-19#	
DC\$SZ	53-13#	
DC\$BLK	53-9#	
DC\$FDB	53-8#	
DC\$LNK	53-11#	
DC\$NXT	53-7#	
DC\$PAR	53-12#	
DC\$USE	53-10#	
DEVLL	67-269#	
DEVLS	67-268#	
DF\$CLS	39-52#	
DF\$DEL	39-53#	
DF\$ENT	39-55#	
DF\$LOK	39-54#	
DF\$REN	39-56#	
DH\$BS	41-12#	41-14
DH\$LB	41-14#	
DH\$MS	41-13#	41-14
DH\$SZ	41-10#	
DH\$BLK	41-9#	
DH\$HIS	41-7#	
DH\$NEB	41-8#	
DH\$NSG	41-5#	
DH\$NXT	41-6#	
DI\$CL	37-47#	
DI\$CR	37-28#	
DI\$CT	37-27#	
DI\$DD	37-35#	
DI\$DL	37-22#	
DI\$DM	37-33#	
DI\$DP	37-31#	
DI\$DR	37-40#	
DI\$DS	37-29#	
DI\$DT	37-18#	
DI\$DU	37-43#	
DI\$DX	37-32#	
DI\$DY	37-23#	
DI\$EL	37-19#	
DI\$LD	37-41#	
DI\$LP	37-20#	

DI\$LS	37-38#	
DI\$MM	37-30#	
DI\$MQ	37-39#	
DI\$MS	37-36#	
DI\$MT	37-25#	
DI\$MU	37-48#	
DI\$NC	37-49#	
DI\$NL	37-34#	
DI\$PC	37-24#	
DI\$PD	37-37#	
DI\$PI	37-45#	
DI\$RF	37-26#	
DI\$RK	37-17#	
DI\$SL	37-44#	
DI\$TT	37-21#	
DI\$VM	37-42#	
DI\$XL	37-46#	37-47
DIABFL	3-38#	
DIABLO	62-13#	
DIABNO	3-39#	
DIBFSZ	64-36#	
DISSLE	63-16#	
DMPHND	64-50#	
DMPOVL	64-49#	
DMPTXT	64-51#	
DMYDEV	64-47#	
DOTRMP	67-312#	
DP\$DAA	67-257#	
DP\$LAA	67-258#	
DS\$ABT	37-9#	
DS\$AJT	37-11#	
DS\$DIR	37-5#	
DS\$ID	37-13#	
DS\$NRD	37-8#	
DS\$RON	37-6#	
DS\$SFN	37-10#	
DS\$VSZ	37-12#	
DS\$WON	37-7#	
DSINT	55-25#	
DW\$\$SZ	25-38#	
DW\$AW	25-17#	
DW\$CCA	25-28#	
DW\$COL	25-11#	
DW\$CPL	25-13#	
DW\$CSB	25-36#	
DW\$CSP	25-18#	
DW\$CSR	25-19#	
DW\$GOM	25-29#	
DW\$G1M	25-30#	
DW\$G2M	25-31#	
DW\$G3M	25-32#	
DW\$GLM	25-33#	
DW\$GLS	25-35#	
DW\$GRM	25-34#	
DW\$ID	25-9#	
DW\$JOB	25-8#	

DW\$LIN	25-10#	
DW\$LPP	25-12#	
DW\$LPT	25-20#	
DW\$MAP	25-22#	
DW\$MSL	25-26#	
DW\$NSL	25-27#	
DW\$RID	25-21#	
DW\$SCA	25-25#	
DW\$SCL	25-24#	
DW\$SLN	25-23#	
DW\$SRB	25-16#	
DW\$SRT	25-15#	
DW\$TLN	25-14#	
DX\$DMA	38-4#	
DX\$EBA	38-6#	
DX\$IBH	38-12#	
DX\$MAP	38-5#	
DX\$MPH	38-10#	
DX\$NCA	38-7#	
DX\$NHM	38-11#	
DX\$NMT	38-8#	
DX\$NRD	38-13#	
DX\$NST	38-14#	
DX\$RAL	38-9#	
DZ\$7BT	56-29#	
DZ\$8BT	56-30#	
DZ\$LEN	56-28#	
DZ\$ODD	56-27#	
DZ\$PAR	56-26#	
DZERR	56-21#	
EB\$BUF	54-8#	
EB\$NAM	54-7#	
EB\$RTN	54-10#	
EB\$SIZ	54-9#	
EB\$XX	54-6#	
EDIT	62-24#	
EMMAP	61-70#	
EMTADR	67-30#	
EMTASP	67-32#	
EMTBLK	67-19#	
EMTCAD	67-34#	
EMTCAS	67-38#	
EMTCAX	67-37#	
EMTCXN	67-25#	67-26
EMTCXT	67-15#	67-26
EMTCXW	67-26#	
EMTERR	67-23#	
EMTLEV	67-31#	
EMTMAP	67-22#	
EMTPS	67-29#	
EMTRAD	67-35#	
EMTSP	67-16#	
ERRHLT	63-13#	
ERRLOC	64-28#	
ERRSEV	67-306#	
ERRSPC	67-267#	

ESC	66-23#		
ESCFLG	64-33#		
ETX	66-5#		
FC\$\$SS	51-18#		
FC\$\$SZ	40-24#		
FC\$ACC	51-12#		
FC\$CDX	40-21#		
FC\$CHN	51-8#		
FC\$CLK	51-15#		
FC\$FDB	51-13#		
FC\$FLG	51-10#		
FC\$FLK	51-14#		
FC\$LBN	51-16#		
FC\$LNK	40-22#		
FC\$NLB	51-11#		
FC\$SBL	40-23#		
FC\$UN	51-9#		
FD\$\$SZ	40-14#		
FD\$CHN	40-10#		
FD\$DAT	40-12#		
FD\$JOB	40-9#	40-11	40-23
FD\$LEN	40-8#		
FD\$NAM	40-7#		
FD\$OPT	40-13#		
FD\$STA	40-6#		
FD\$TIM	40-11#		
FF	66-13#		
FF\$\$SZ	52-14#		
FF\$CDB	52-8#		
FF\$DCD	52-9#		
FF\$FID	52-7#		
FF\$FLG	52-12#		
FF\$FLK	52-13#		
FF\$FWD	52-10#		
FF\$NLB	52-11#		
FL\$ACT	51-22#		
FL\$EFL	51-23#		
FL\$NDC	51-24#		
FL\$SPN	51-25#		
FQ\$\$SZ	43-20#		
FQ\$LNK	43-9#		
FQ\$PA5	43-17#		
FQ\$PA6	43-18#		
FQ\$PRI	43-19#		
FQ\$R1	43-15#		
FQ\$R2	43-14#		
FQ\$R3	43-13#		
FQ\$R4	43-12#		
FQ\$R5	43-11#		
FQ\$RTN	43-10#		
FQ\$UFB	43-16#		
FRMERR	55-34#		
FS\$EMP	40-31#		
FS\$EOS	40-29#		
FS\$PRM	40-30#		
FS\$PRO	40-28#		

FS\$TEN 40-32#
FT\$EFL 52-18#
FW\$\$SZ 52-29#
FW\$DBN 52-26#
FW\$UN 52-27#
FW\$WLK 52-28#
GTLTTY 63-17#
H. CQE 39-18#
H. CSR 39-12#
H. DSTS 39-10#
H. DVSZ 39-9#
H. ENT 39-19#
H. FET 39-6#
H. GEN 39-11#
H. INS 39-13#
H. INT 39-15#
H. LOAD 39-7#
H. LQE 39-17#
H. PRI 39-16#
H. SIZ 39-8#
H. VEC 39-14#
HAZEL 62-12#
HAZLFL 3-47#
HAZLNO 3-48#
HF\$7BT 57-34#
HF\$8BT 57-35#
HF\$AEE 57-28#
HF\$CNI 57-14#
HF\$DO 57-22#
HF\$FE 57-23#
HF\$HD 57-29#
HF\$LEN 57-33#
HF\$LIN 57-17#
HF\$MC 57-11#
HF\$MM 57-13#
HF\$NXM 57-12#
HF\$ODD 57-30#
HF\$PAR 57-31#
HF\$PE 57-24#
HF\$RI 57-15#
HF\$RIE 57-16#
HF\$SI 57-8#
HF\$SIE 57-10#
HF\$TI 57-7#
HF\$TIE 57-9#
HF\$TSB 57-32#
HF\$VDP 57-21#
HIMLOC 64-27#
HSR\$FE 39-25#
HSR\$LO 39-27#
HSR\$RE 39-26#
HSR\$UN 39-28#
IB\$\$SZ 23-12#
IB\$IJ 23-10#
IB\$SF2 23-9#
II\$\$SZ 44-10#

II\$FLG	44-7#		
II\$NAM	44-6#		
II\$NPV	44-9#		
II\$PRV	44-8#		
IN\$ACT	16-55#		
IN\$CMD	16-57#		
IN\$CNT	16-56#		
INDERR	63-43#		
INDSTA	63-44#		
INTERR	67-24#		
INTMX1	56-15#		
IOMAP	61-71#		
IOPAGE	64-22#		
IOQSI2	19-28#		
ISPF11	12-14#		
ISPF5	6-23#		
ISPF6	7-23#		
ISPF7	8-23#	8-24	8-24#
ISPF9	10-23#		
IT\$HOT	21-7#		
IT\$JOB	21-10#		
IT\$LNK	21-9#		
IT\$LOT	21-8#		
IT\$RTN	21-13#		
IT\$SEQ	21-11#		
IT\$SYS	21-12#		
JCDB	67-266#		
JIDLN	63-33#		
JIMLOK	63-34#		
JIPRIV	63-35#		
JIVLN	63-32#		
JM\$\$SZ	24-10#		
JM\$JOB	24-9#		
JM\$LNK	24-7#		
JM\$RTN	24-8#		
JOBCCB	67-284#		
JPWDEV	67-292#		
JPWFLG	67-294#		
JPWTYP	67-293#		
JS\$KMN	24-17#		
JS\$LOG	24-15#		
JS\$OFF	24-18#		
JS\$ON	24-14#		
JS\$RUN	24-16#		
JSTK	67-9#		
JSTKND	67-8#		
JSWLOC	64-26#		
K52	62-27#		
KC\$COM	28-47#		
KC\$DOT	28-46#		
KC\$DWN	28-51#		
KC\$E1	28-54#		
KC\$E2	28-55#		
KC\$E3	28-56#		
KC\$E4	28-57#		
KC\$E5	28-58#		

KC%E6	28-59#
KC\$ENT	28-49#
KC\$F10	28-64#
KC\$F11	28-65#
KC\$F12	28-66#
KC\$F13	28-67#
KC\$F14	28-68#
KC\$F15	28-69#
KC\$F16	28-70#
KC\$F17	28-71#
KC\$F18	28-72#
KC\$F19	28-73#
KC\$F20	28-74#
KC\$F6	28-60#
KC\$F7	28-61#
KC\$F8	28-62#
KC\$F9	28-63#
KC\$KP0	28-36#
KC\$KP1	28-37#
KC\$KP2	28-38#
KC\$KP3	28-39#
KC\$KP4	28-40#
KC\$KP5	28-41#
KC\$KP6	28-42#
KC\$KP7	28-43#
KC\$KP8	28-44#
KC\$KP9	28-45#
KC\$LFT	28-52#
KC\$MIN	28-48#
KC\$PF1	28-32#
KC\$PF2	28-33#
KC\$PF3	28-34#
KC\$PF4	28-35#
KC\$RIT	28-53#
KC\$UP	28-50#
KD\$\$SZ	28-16#
KD\$COD	28-12#
KD\$FLG	28-14#
KD\$TXT	28-15#
KD\$TYP	28-13#
KED	62-26#
KEYMXT	28-7#
KEYPAR	67-186#
KEYRCB	67-185#
KF\$ECO	28-20#
KF\$TRM	28-21#
KL3CLR	4-21#
KL4CLR	5-21#
KMNBAS	64-41#
KMPRMT	67-295#
KPAR0	61-9#
KPAR1	61-10#
KPAR2	61-11#
KPAR3	61-12#
KPAR4	61-13#
KPAR5	61-14#

KPAR6	61-15#							
KPAR7	61-16#							
KPDRO	61-20#	61-21	61-22	61-23	61-24	61-25	61-26	61-27
KPDR1	61-21#							
KPDR2	61-22#							
KPDR3	61-23#							
KPDR4	61-24#							
KPDR5	61-25#							
KPDR6	61-26#							
KPDR7	61-27#							
KT\$GLD	28-26#							
KT\$GLT	28-28#							
KT\$LET	28-27#							
KT\$NRM	28-25#							
LA120	62-8#							
LA12FL	3-26#							
LA12NO	3-27#							
LA36	62-7#							
LA36FL	3-23#							
LA36NO	3-24#							
LCBIT	63-6#							
LD\$RON	38-20#							
LDBASE	67-110#							
LDFLAG	67-111#							
LDNAME	67-107#							
LDPDEV	67-108#							
LDSIZE	67-109#							
LF	66-12#							
LF\$IN	50-8#							
LF\$OPN	50-6#							
LF\$OUT	50-9#							
LF\$WRT	50-7#							
LOFSPC	67-290#							
LOGBAS	67-152#							
LOGBLK	67-149#							
LOGBUF	67-146#							
LOGDVU	67-151#							
LOGEND	67-147#							
LOGFLG	67-150#							
LOGPTR	67-148#							
LP\$7BT	59-26#							
LP\$ODD	59-28#							
LP\$PAR	59-27#							
LP\$SPD	59-25#							
LSTCHR	67-302#							
LSTFDD	67-283#							
LSTFDT	67-282#							
LSTPRM	67-134#							
MA\$RGN	45-12#							
MA\$SRT	45-11#							
MA\$SYS	45-10#							
MAXACC	64-7#							
MAXASN	64-6#							
MAXLD	64-10#							
MAXMEM	67-277#							
MAXPRI	64-17#							

MAXSLO 64-5#
MAXSRD 64-16#
MB\$\$SZ 54-19#
MB\$BUF 54-17#
MB\$FLK 54-15#
MB\$NAM 54-16#
MB\$REQ 54-18#
MF\$BSY 57-45#
MF\$CAR 57-51#
MF\$CM 57-40#
MF\$CS 57-39#
MF\$CTS 57-52#
MF\$DON 57-42#
MF\$DTR 57-56#
MF\$IE 57-43#
MF\$LE 57-57#
MF\$LIN 57-46#
MF\$RNG 57-50#
MF\$RTS 57-55#
MF\$SE 57-44#
MF\$SR 57-53#
MF\$ST 57-54#
MF\$STP 57-41#
MI\$\$SZ 34-21#
MI\$CWC 34-15#
MI\$JOB 34-20#
MI\$LNK 34-12#
MI\$OQE 34-13#
MI\$RWF 34-19#
MI\$SBP 34-18#
MI\$TRW 34-14#
MI\$UBO 34-17#
MI\$UBP 34-16#
MMENBL 61-69#
MPARO 61-64#
MPAR16 61-65#
MR\$\$SZ 54-30#
MR\$BUF 54-28#
MR\$JOB 54-29#
MR\$LNK 54-24#
MR\$RTN 54-27#
MR\$UBA 54-25#
MR\$UBS 54-26#
MS\$BRK 55-9#
MS\$CAR 55-7#
MS\$DTR 55-8#
MS\$RNG 55-6#
MSE 56-13#
MU\$FLK 54-35#
MU\$JOB 54-37#
MU\$SIZ 54-36#
MU\$TXT 54-38#
MW\$\$SZ 34-30#
MW\$IOQ 34-29#
MW\$LNK 34-28#
MXCCHR 64-9#

56-15

MXCPRM	64-8#	
MXJPRI	67-291#	
MXPRMT	64-13#	
NEWJSW	67-275#	
NOUSWP	63-5#	
NOWAIT	63-14#	
NPCCB	67-285#	
NUMRCB	64-11#	
NUMWCB	64-12#	
O. ADR	17-6#	
O. BLK	17-8#	
O. PAR	17-7#	
O. SIZ	17-9#	
ODTBAS	67-274#	
OF\$\$SZ	42-24#	
OF\$DEV	42-21#	
OF\$FIL	42-20#	
OF\$FLG	42-23#	
OF\$UNT	42-22#	
OKFEND	67-103#	
OKFILE	67-102#	
OT\$RON	42-28#	
OVLBIT	63-11#	
OVERRUN	55-33#	
PO\$\$NP	13-52#	
PO\$ALC	13-14#	13-52
PO\$BYP	13-19#	
PO\$DBG	13-15#	13-52
PO\$DET	13-16#	13-52
PO\$LOK	13-21#	
PO\$MEM	13-18#	
PO\$NAM	13-24#	13-52
PO\$NEW	13-13#	13-52
PO\$NFR	13-26#	13-52
PO\$NFW	13-27#	13-52
PO\$OPR	13-20#	
PO\$RT	13-22#	
PO\$SND	13-23#	13-52
PO\$SPF	13-17#	13-52
PO\$SPV	13-25#	
PO\$SYS	13-28#	
P2\$\$NP	13-53#	
P2\$CGR	13-39#	
P2\$CXT	13-40#	
P2\$GRP	13-34#	
P2\$MSG	13-37#	13-53
P2\$RLK	13-38#	13-53
P2\$SAM	13-35#	13-53
P2\$TRM	13-32#	
P2\$UP1	13-44#	
P2\$UP2	13-43#	
P2\$UP3	13-42#	
P2\$UP4	13-41#	
P2\$VIR	13-36#	13-53
P2\$WRL	13-33#	
PA\$BEL	27-14#	

PA\$BLD	27-9#	
PA\$DSC	27-8#	
PA\$DTS	27-16#	
PA\$DWD	27-11#	
PA\$GRC	27-6#	
PA\$HQL	27-12#	
PA\$LET	27-13#	
PA\$NWD	27-15#	
PA\$UKC	27-7#	
PA\$ULN	27-10#	
PARENL	61-72#	
PASLIN	63-9#	
PBFEND	67-137#	
PC\$AF	46-27#	
PC\$IRQ	46-29#	
PC\$PF	46-28#	
PC\$UF	46-26#	
PCCCR0	46-17#	
PCCCR1	46-18#	
PCCCR2	46-19#	
PCCCR3	46-20#	
PCCDAT	46-14#	
PCCDAY	46-13#	
PCCHAL	46-12#	
PCCHRS	46-11#	
PCCMAL	46-10#	
PCCMIN	46-9#	
PCCMON	46-15#	
PCCSAL	46-8#	
PCCSEC	46-7#	
PCCVEC	46-22#	
PCCYR	46-16#	
PF\$IOW	63-24#	
PF\$OVF	63-26#	
PF\$SYS	63-25#	
PR7	64-39#	
PRGSIZ	67-279#	
PRGTOP	67-278#	
PRIVA0	67-115#	
PRIVA2	67-116#	
PRIVC0	67-121#	
PRIVC2	67-122#	
PRIVF0	67-119#	
PRIVF2	67-120#	
PRIVS0	67-117#	
PRIVS2	67-118#	
PRMBUF	67-135#	
PRMEND	67-136#	
PRMPNT	67-133#	
PSW	64-42#	
PVNPW	13-48#	
Q. BLKN	19-8#	20-10
Q. BUFF	19-12#	20-13
Q. CHAN	19-18#	
Q. COMP	19-14#	20-15
Q. CSW	19-7#	20-9

Q.DEVX	19-19#	20-18	
Q.FLAG	19-20#	20-17	
Q.FUNC	19-9#	20-11	
Q.ICSW	19-27#		
Q.JNUM	19-11#		
Q.JOB	19-21#		
Q.LINK	19-6#	20-8	
Q.PA5	19-16#	20-16	
Q.PA6	19-25#		
Q.PAR	19-15#		
Q.UCSW	19-26#		
Q.UMPB	19-23#		
Q.UMPP	19-24#		
Q.UMRX	19-17#	20-19	
Q.UMVB	19-22#		
Q.UNIT	19-10#	19-11	20-12
Q.WCNT	19-13#	20-14	
QF\$CIC	20-27#		
QF\$IOT	20-29#		
QF\$MIO	20-26#		
QF\$OWC	20-28#		
QF\$SCR	20-25#		
QF\$SYN	20-30#		
QUME	62-14#		
QUMEFL	3-41#		
QUMENO	3-42#		
R.GID	31-35#		
R.GSIZ	31-36#		
R.GSTS	31-37#		
R.NAME	31-38#		
RBERR	55-32#		
RC\$SZ	31-14#		
RC\$AEP	31-27#		
RC\$AGE	31-24#		
RC\$BAS	31-8#		
RC\$BLK	31-9#		
RC\$CNT	31-12#		
RC\$EXC	31-23#		
RC\$EXI	31-25#		
RC\$FLG	31-10#		
RC\$GBL	31-20#		
RC\$INM	31-18#		
RC\$LCG	31-22#		
RC\$LEN	31-6#		
RC\$NAM	31-11#		
RC\$OFF	31-26#		
RC\$OWN	31-13#		
RC\$PAG	31-7#		
RC\$PRM	31-28#		
RC\$PVT	31-21#		
RC\$SFA	31-19#		
RC\$USE	31-29#		
RCBBAS	67-82#		
RCBEND	67-83#		
RCLREV	67-181#		
RCVACT	55-22#		

RCVDON	55-23#		
RCVINT	55-24#	55-28	
RCVPAR	55-35#		
RDINT	55-28#		
RDONE	56-11#		
REENT	63-7#		
REQSND	55-26#		
RESDEV	67-101#		
RF\$WRT	36-19#		
RIE	56-12#	56-15	
RING	55-19#		
RLINE	56-22#		
RMNBAS	63-42#	63-43	63-44
RMON	64-30#		
RPAR	67-76#		
RPDR	67-77#		
RPDRND	67-78#		
RS. AGE	31-48#		
RS. CGR	31-47#		
RS. CRR	31-42#		
RS. EGR	31-49#		
RS. EXI	31-50#		
RS. GBL	31-46#		
RS. NAL	31-44#		
RS. NEW	31-45#		
RS. PVT	31-51#		
RS. UNM	31-43#		
RT\$SZ	36-15#		
RT\$BAS	36-11#		
RT\$DEV	36-8#		
RT\$EXT	36-10#		
RT\$FLG	36-13#		
RT\$NAM	36-9#		
RT\$SKP	36-14#		
RT\$TOP	36-12#		
RUBOUT	66-25#		
RUNARG	67-301#		
RUNDEV	67-263#		
RUNFLG	67-264#		
RUNRDB	67-86#		
S\$HIP	14-24#		
S\$RT	14-13#		
S\$RUN	14-30#		
S\$CPU	14-28#		
S\$DUMY	14-8#		
S\$HICP	14-20#		
S\$INWT	14-44#		
S\$IOFN	14-23#		
S\$IOWT	14-39#		
S\$LOW	14-29#		
S\$MSWT	14-47#		
S\$NEDQ	14-34#		
S\$OTFN	14-19#		
S\$OTLO	14-22#		
S\$OTWT	14-40#		
S\$QCCB	14-36#		

S\$QCXB	14-37#	
S\$QMIO	14-35#	
S\$QSPD	14-45#	
S\$QUSR	14-38#	
S\$RT	14-12#	
S\$SFWT	14-41#	
S\$SPCB	14-46#	
S\$SPDB	14-43#	
S\$SPND	14-48#	
S\$TMWT	14-49#	
S\$TTFN	14-18#	
S\$TTSC	14-17#	
S\$TWFN	14-21#	
S\$WFM	14-50#	
S\$WSMB	14-42#	
S110	59-7#	
S1200	59-12#	
S134.5	59-8#	
S150	59-9#	
S1800	59-13#	
S19200	59-20#	
S2000	59-14#	
S2400	59-15#	
S300	59-10#	
S3600	59-16#	
S4800	59-17#	
S50	59-5#	
S600	59-11#	
S7200	59-18#	
S75	59-6#	
S9600	59-19#	
SA\$LOK	22-15#	
SA\$RGN	22-16#	
SB\$SZ	15-10#	
SB\$END	15-9#	
SB\$LNK	15-6#	
SB\$PNT	15-7#	
SB\$TXT	15-8#	
SBPSUF	67-296#	
SBUFSZ	64-37#	
SBUFWO	64-38#	
SC\$ERR	65-7#	
SC\$FTL	65-9#	
SC\$NON	65-11#	
SC\$SEV	65-8#	
SC\$SUC	65-5#	
SC\$UNC	65-10#	
SC\$WRN	65-6#	
SCHAIN	63-15#	
SD\$BAK	48-36#	48-37
SD\$BWT	48-29#	
SD\$CLR	48-37#	
SD\$DEL	48-30#	48-37
SD\$FLK	48-32#	
SD\$HLD	48-33#	
SD\$INR	48-28#	

SD\$SMS	48-35#	48-37
SD\$SNG	48-34#	
SD\$WFM	48-31#	48-37
SDANAM	48-20#	
SDBLK	48-17#	
SDBU	48-24#	
SDBUF1	48-11#	
SDBUF2	48-12#	
SDCHAN	48-8#	
SDDVU	48-15#	
SDFHD	48-18#	
SDFLAG	48-21#	
SDFLNK	48-10#	
SDFORM	48-19#	
SDFRBL	48-23#	
SDNAME	48-14#	
SDSFCB	48-9#	
SDSKIP	48-22#	
SDUSER	48-13#	
SDWLST	48-16#	
SERFLG	67-304#	
SF\$1ST	47-26#	
SF\$BN1	47-25#	
SF\$BSY	47-24#	
SF\$DEL	47-28#	
SF\$HLD	47-27#	
SF2LEN	23-6#	
SFCBSZ	47-20#	
SFCHAN	47-11#	
SFFILE	47-17#	
SFFLAG	47-10#	
SFFLNK	47-15#	
SFFORM	47-16#	
SFID	47-18#	
SFNMBL	47-13#	
SFQLNK	47-19#	
SFSDCB	47-12#	
SFSTRT	47-14#	
SFUSER	47-9#	
SG\$ELG	16-38#	
SG\$EMT	16-41#	
SG\$IOT	16-40#	
SG\$MMU	16-39#	
SG\$MTM	16-43#	
SG\$MTS	16-44#	
SG\$PAR	16-42#	
SG\$SYJ	16-45#	
SG\$TSX	16-46#	
SH\$SZ	39-38#	
SH\$FLG	39-37#	
SH\$NAM	39-35#	
SH\$RTN	39-36#	
SH\$VAL	39-34#	
SLBACK	67-178#	
SLBFSZ	64-15#	
SLCBUF	67-173#	

SLCCOL	67-157#
SLCR	67-177#
SLCSBF	67-174#
SLCSBX	67-175#
SLCSPT	67-166#
SLCSR	67-167#
SLCX	67-156#
SLCYC1	67-163#
SLCYC2	67-164#
SLDBUF	67-171#
SLDOWN	67-179#
SLEBUF	67-168#
SLECOL	67-158#
SLGOLD	67-176#
SLLBUF	67-169#
SLLEND	67-170#
SLLPTR	67-162#
SLMXLN	64-14#
SLOPTR	67-160#
SLOVER	67-180#
SLRPTR	67-165#
SLSBUF	67-172#
SLSCOL	67-159#
SLSPTR	67-161#
SN\$ID	29-10#
SN\$JOB	29-7#
SN\$LNK	29-6#
SN\$RTN	29-12#
SN\$XX1	29-8#
SN\$XX2	29-9#
SN\$XX3	29-11#
SNDCLR	55-20#
SO\$NO	39-42#
SO\$NVL	39-43#
SO\$OCT	39-44#
SP\$SZ	22-11#
SP\$CMD	22-8#
SP\$DW1	22-10#
SP\$JOB	22-9#
SP\$LNK	22-7#
SPACE	66-24#
SPCFLG	67-48#
SPCPS	67-33#
SPCTTY	63-8#
SPDLBF	67-196#
SPIJ	67-310#
SPLARG	67-195#
SPSAVE	67-272#
SPUBND	67-193#
SPUBUF	67-192#
SR\$PAR	67-90#
SR\$PDR	67-91#
SR\$PX	67-92#
SROMMR	61-53#
SR1MMR	61-54#
SR2MMR	61-55#

SR3MMR	61-56#								
SS\$PRT	16-51#								
SS\$RUN	16-50#								
SS3CHR	66-27#								
STATCH	55-18#								
SUCF2	67-311#								
SUPRTN	64-24#								
SWTCNT	67-49#								
SXBPNT	67-194#								
SXBUF1	67-191#								
SXSFCB	67-190#								
TAB	66-11#								
TCSBSZ	25-5#								
TECO	62-25#								
TIE	56-9#	56-15							
TLINE	56-10#								
TO\$FF	16-66#								
TO\$LC	16-67#								
TO\$SCP	16-68#								
TO\$TAB	16-65#								
TRBRK	55-41#								
TRDY	56-8#								
TRINT	55-40#								
TRMRDY	55-27#								
TRNSFL	64-34#								
TRRDY	55-39#								
TSDEFS	1-5	1-6#							
UC\$\$SZ	30-13#								
UC\$MDC	30-12#								
UC\$NDC	30-11#								
UCHAN	67-286#								
UCISPC	67-289#								
UCLMCL	30-6#								
UCLMKL	30-5#								
VERSEV	67-307#								
UFORM	67-197#								
UFPTRP	67-281#								
UHIMEM	67-270#								
UK\$\$SZ	30-20#								
UK\$NAM	30-18#								
UK\$PTR	30-19#								
UM\$\$SZ	33-13#								
UM\$IDQ	33-12#								
UM\$NMR	33-10#								
UM\$UMR	33-9#								
UM\$WDS	33-11#								
UMODE	64-43#								
UMRADR	61-60#								
UMSPSV	67-276#								
UPARO	61-31#	61-32	61-33	61-34	61-35	61-36	61-37	61-38	
UPAR1	61-32#								
UPAR2	61-33#								
UPAR3	61-34#								
UPAR4	61-35#								
UPAR5	61-36#								
UPAR6	61-37#								

UPAR7	61-38#							
UPDR0	61-42#	61-43	61-44	61-45	61-46	61-47	61-48	61-49
UPDR1	61-43#							
UPDR2	61-44#							
UPDR3	61-45#							
UPDR4	61-46#							
UPDR5	61-47#							
UPDR6	61-48#							
UPDR7	61-49#							
UPMODE	64-44#							
UPPN	67-262#							
URO	67-17#							
US\$SZ	30-26#							
US\$TXT	30-25#							
USERRB	64-29#							
USRLOC	64-25#							
USRSTK	67-271#							
U\$TART	67-273#							
UTRPAD	67-280#							
VC\$SZ	29-28#							
VC\$FLG	29-27#							
VC\$JOB	29-25#							
VC\$JSR	29-22#							
VC\$PRI	29-26#							
VC\$RTN	29-23#							
VC\$VEC	29-24#							
VF\$7BT	58-34#							
VF\$8BT	58-35#							
VF\$ABT	58-57#							
VF\$BC	58-54#							
VF\$CTS	58-45#							
VF\$DCD	58-44#							
VF\$DET	29-33#							
VF\$DF	58-9#							
VF\$DIR	29-32#							
VF\$DSR	58-42#							
VF\$DTR	58-50#							
VF\$DV	58-21#							
VF\$EVN	58-37#							
VF\$FE	58-23#							
VF\$IFC	58-56#							
VF\$LEN	58-33#							
VF\$LIN	58-15#							
VF\$LT	58-51#							
VF\$MR	58-14#							
VF\$OE	58-22#							
VF\$OFC	58-53#							
VF\$PAR	58-36#							
VF\$PER	58-24#							
VF\$RDA	58-12#							
VF\$RE	58-55#							
VF\$RIE	58-13#							
VF\$RNG	58-43#							
VF\$RTS	58-49#							
VF\$SC	58-38#							
VF\$TDE	58-10#							

VF\$TDV	58-29#	
VF\$TEN	58-61#	
VF\$TGO	58-62#	
VF\$TIE	58-8#	
VF\$TR	58-7#	
VF\$XLN	58-11#	
VF\$XOF	58-52#	
VIMAGE	63-10#	
VO\$HIO	17-20#	
VO\$HIR	17-19#	
VO\$RDB	17-18#	
VO\$WDB	17-17#	
VO\$WDE	17-16#	
VPAR1	64-18#	
VPAR5	64-19#	
VPAR6	64-21#	
VT100	62-10#	
VT10FL	3-32#	
VT10NO	3-33#	
VT200	62-17#	
VT2007	62-15#	62-17
VT2008	62-16#	
VT20FL	3-35#	
VT20NO	3-36#	
VT52	62-9#	
VT52FL	3-29#	
VT52NO	3-30#	
W. NAPR	32-20#	
W. NBAS	32-21#	
W. NID	32-19#	
W. NLEN	32-25#	
W. NOFF	32-24#	
W. NRID	32-23#	
W. NSIZ	32-22#	
W. NSTS	32-26#	
W. SIZE	32-27#	
WC\$SZ	32-13#	
WC\$LEN	32-9#	
WC\$NPR	32-12#	
WC\$OFF	32-10#	
WC\$PAR	32-11#	
WC\$RCB	32-5#	
WC\$SIZ	32-6#	
WC\$VHI	32-8#	
WC\$VLO	32-7#	
WCBBAS	67-84#	
WCBEND	67-85#	
WLDNAM	64-48#	
WS. CRW	32-34#	
WS. ELW	32-32#	
WS. MAP	32-31#	
WS. UNM	32-33#	
XL\$CD	60-43#	
XL\$CTS	60-42#	
XL\$RI	60-44#	
XL\$XFR	60-41#	

XL\$XFX 60-40#
ZCLR 56-14#
ZFRMER 56-20#

M	1-56#	2-5	2-6	2-7	2-8	2-9	2-10	2-11	2-12	2-13	2-14	2-15
	2-16	2-17	2-18	3-4	3-5	3-6	3-7	3-8	3-9	3-10	3-11	3-12
	3-13	3-14	3-15	3-16	3-17	3-18	3-23	3-24	3-26	3-27	3-29	3-30
	3-32	3-33	3-35	3-36	3-38	3-39	3-41	3-42	3-44	3-45	3-47	3-48
	4-4	4-5	4-6	4-7	4-8	4-9	4-10	4-11	4-12	4-13	4-14	4-15
	4-16	4-17	4-18	4-19	4-21	5-4	5-5	5-6	5-7	5-8	5-9	5-10
	5-11	5-12	5-13	5-14	5-15	5-16	5-17	5-18	5-19	5-21	5-22	6-4
	6-5	6-6	6-7	6-8	6-9	6-10	6-11	6-12	6-13	6-14	6-15	6-16
	6-17	6-18	6-19	6-23	7-4	7-5	7-6	7-7	7-8	7-9	7-10	7-11
	7-12	7-13	7-14	7-15	7-16	7-17	7-18	7-19	7-23	8-4	8-5	8-6
	8-7	8-8	8-9	8-10	8-11	8-12	8-13	8-14	8-15	8-16	8-17	8-18
	8-19	8-23	8-24	9-4	9-5	9-6	9-7	9-8	9-9	9-10	9-11	9-12
	9-16	10-4	10-5	10-6	10-7	10-8	10-9	10-10	10-11	10-12	10-13	10-14
	10-15	10-16	10-17	10-18	10-19	10-23	11-4	11-5	11-6	11-7	11-8	11-9
	11-10	11-11	11-12	11-13	11-14	11-15	12-4	12-5	12-6	12-7	12-8	12-9
	12-10	12-14	13-13	13-14	13-15	13-16	13-17	13-18	13-19	13-20	13-21	13-22
	13-23	13-24	13-25	13-26	13-27	13-28	13-32	13-33	13-34	13-35	13-36	13-37
	13-38	13-39	13-40	13-41	13-42	13-43	13-44	13-48	13-52	13-53	16-5	16-6
	16-7	16-8	16-9	16-10	16-11	16-12	16-13	16-14	16-15	16-16	16-17	16-18
	16-22	16-23	16-24	16-25	16-26	16-27	16-28	16-29	16-30	16-31	16-32	16-33
	16-34	16-38	16-39	16-40	16-41	16-42	16-43	16-44	16-45	16-46	16-50	16-51
	16-55	16-56	16-57	16-61	16-65	16-66	16-67	16-68	16-72	16-73	17-16	17-17
	17-18	17-19	17-20	18-16	18-17	18-18	18-19	18-20	18-21	18-22	18-23	19-11
	20-8	20-9	20-10	20-11	20-12	20-13	20-14	20-15	20-16	20-17	20-18	20-19
	20-25	20-26	20-27	20-28	20-29	20-30	22-15	22-16	23-6	24-14	24-15	24-16
	24-17	24-18	25-5	26-4	26-5	26-6	26-10	26-11	26-12	26-13	26-14	26-18
	26-19	26-20	26-21	26-22	26-23	26-24	26-25	26-26	26-27	26-28	26-29	26-30
	26-31	26-32	26-33	27-6	27-7	27-8	27-9	27-10	27-11	27-12	27-13	27-14
	27-15	27-16	28-7	28-20	28-21	28-25	28-26	28-27	28-28	28-32	28-33	28-34
	28-35	28-36	28-37	28-38	28-39	28-40	28-41	28-42	28-43	28-44	28-45	28-46
	28-47	28-48	28-49	28-50	28-51	28-52	28-53	28-54	28-55	28-56	28-57	28-58
	28-59	28-60	28-61	28-62	28-63	28-64	28-65	28-66	28-67	28-68	28-69	28-70
	28-71	28-72	28-73	28-74	29-32	29-33	30-5	30-6	31-18	31-19	31-20	31-21
	31-22	31-23	31-24	31-25	31-26	31-27	31-28	31-29	31-42	31-43	31-44	31-45
	31-46	31-47	31-48	31-49	31-50	31-51	32-31	32-32	32-33	32-34	36-19	37-5
	37-6	37-7	37-8	37-9	37-10	37-11	37-12	37-13	37-17	37-18	37-19	37-20
	37-21	37-22	37-23	37-24	37-25	37-26	37-27	37-28	37-29	37-30	37-31	37-32
	37-33	37-34	37-35	37-36	37-37	37-38	37-39	37-40	37-41	37-42	37-43	37-44
	37-45	37-46	37-47	37-48	37-49	38-4	38-5	38-6	38-7	38-8	38-9	38-10
	38-11	38-12	38-13	38-14	38-20	39-6	39-7	39-8	39-9	39-10	39-11	39-12
	39-13	39-14	39-15	39-16	39-17	39-18	39-19	39-25	39-26	39-27	39-28	39-42
	39-43	39-44	39-52	39-53	39-54	39-55	39-56	40-11	40-23	40-28	40-29	40-30
	40-31	40-32	41-12	41-13	41-14	41-27	42-28	44-14	44-15	44-16	44-17	44-18
	44-19	44-20	44-21	44-22	44-23	44-24	44-25	44-26	44-27	44-28	44-29	45-10
	45-11	45-12	46-7	46-8	46-9	46-10	46-11	46-12	46-13	46-14	46-15	46-16
	46-17	46-18	46-19	46-20	46-22	46-26	46-27	46-28	46-29	47-24	47-25	47-26
	47-27	47-28	48-28	48-29	48-30	48-31	48-32	48-33	48-34	48-35	48-36	48-37
	49-20	50-6	50-7	50-8	50-9	51-22	51-23	51-24	51-25	52-18	55-6	55-7
	55-8	55-9	55-18	55-19	55-20	55-21	55-22	55-23	55-24	55-25	55-26	55-27
	55-28	55-32	55-33	55-34	55-35	55-39	55-40	55-41	56-8	56-9	56-10	56-11
	56-12	56-13	56-14	56-15	56-19	56-20	56-21	56-22	56-26	56-27	56-28	56-29
	56-30	57-7	57-8	57-9	57-10	57-11	57-12	57-13	57-14	57-15	57-16	57-17
	57-21	57-22	57-23	57-24	57-28	57-29	57-30	57-31	57-32	57-33	57-34	57-35
	57-39	57-40	57-41	57-42	57-43	57-44	57-45	57-46	57-50	57-51	57-52	57-53
	57-54	57-55	57-56	57-57	58-7	58-8	58-9	58-10	58-11	58-12	58-13	58-14
	58-15	58-21	58-22	58-23	58-24	58-29	58-33	58-34	58-35	58-36	58-37	58-38

	58-42	58-43	58-44	58-45	58-49	58-50	58-51	58-52	58-53	58-54	58-55	58-56
	58-57	58-61	58-62	59-5	59-6	59-7	59-8	59-9	59-10	59-11	59-12	59-13
	59-14	59-15	59-16	59-17	59-18	59-19	59-20	59-25	59-26	59-27	59-28	60-7
	60-8	60-9	60-10	60-11	60-12	60-13	60-14	60-15	60-16	60-17	60-18	60-20
	60-24	60-25	60-26	60-27	60-28	60-29	60-30	60-31	60-32	60-33	60-34	60-35
	60-36	60-40	60-41	60-42	60-43	60-44	60-48	60-49	60-50	60-51	60-52	60-53
	60-54	60-55	60-56	60-57	60-58	60-59	60-60	60-61	60-62	60-63	60-64	60-65
	60-66	60-67	61-9	61-10	61-11	61-12	61-13	61-14	61-15	61-16	61-20	61-21
	61-22	61-23	61-24	61-25	61-26	61-27	61-31	61-32	61-33	61-34	61-35	61-36
	61-37	61-38	61-42	61-43	61-44	61-45	61-46	61-47	61-48	61-49	61-53	61-54
	61-55	61-56	61-60	61-64	61-65	61-69	61-70	61-71	61-72	62-7	62-8	62-9
	62-10	62-11	62-12	62-13	62-14	62-15	62-16	62-17	62-24	62-25	62-26	62-27
	63-5	63-6	63-7	63-8	63-9	63-10	63-11	63-12	63-13	63-14	63-15	63-16
	63-17	63-24	63-25	63-26	63-32	63-33	63-34	63-35	63-42	63-43	63-44	64-5
	64-6	64-7	64-8	64-9	64-10	64-11	64-12	64-13	64-14	64-15	64-16	64-17
	64-18	64-19	64-20	64-21	64-22	64-23	64-24	64-25	64-26	64-27	64-28	64-29
	64-30	64-31	64-32	64-33	64-34	64-35	64-36	64-37	64-38	64-39	64-40	64-41
	64-42	64-43	64-44	64-45	64-46	64-47	64-48	64-49	64-50	64-51	65-5	65-6
	65-7	65-8	65-9	65-10	65-11	65-5	66-6	66-7	66-8	66-9	66-10	66-11
	66-12	66-13	66-14	66-15	66-16	66-17	66-18	66-19	66-20	66-21	66-22	66-23
	66-24	66-25	66-26	66-27	67-26	67-86	67-227	67-241	67-242	67-243	67-244	67-245
	67-246	67-247	67-248	67-249	67-250	67-251	67-252	67-253	67-257	67-258	67-317	
RB	1-42#	14-8	14-12	14-13	14-17	14-18	14-19	14-20	14-21	14-22	14-23	14-24
	14-28	14-29	14-30	14-34	14-35	14-36	14-37	14-38	14-39	14-40	14-41	14-42
	14-43	14-44	14-45	14-46	14-47	14-48	14-49	14-50	18-10	18-11	19-9	19-10
	19-19	19-20	19-21	19-22	22-8	22-9	23-9	23-10	24-9	25-8	25-9	25-23
	25-24	25-25	25-26	25-27	25-28	25-29	25-30	25-31	25-32	25-33	25-34	25-35
	25-36	28-12	28-13	28-14	28-15	29-24	29-25	29-26	29-27	30-18	30-25	31-12
	31-13	32-11	32-12	32-19	32-20	33-9	33-10	34-19	34-20	35-15	36-13	36-14
	39-36	39-37	40-9	40-10	41-34	42-21	42-22	42-23	42-38	42-39	43-19	47-9
	47-10	48-20	49-6	49-7	49-8	49-9	49-10	49-11	49-12	49-13	49-14	49-15
	49-16	49-17	51-8	51-9	51-10	51-11	52-12	54-29	54-37	67-23	67-24	67-43
	67-48	67-49	67-50	67-92	67-135	67-168	67-169	67-170	67-171	67-172	67-173	67-174
	67-175	67-176	67-177	67-178	67-179	67-180	67-181	67-229	67-230	67-231	67-232	67-233
	67-234	67-235	67-236	67-300	67-301	67-302	67-303	67-304	67-305	67-306	67-307	67-308
	67-309	67-310	67-311	67-312								
RW	1-25#	15-6	15-7	15-8	15-9	15-10	17-6	17-7	17-8	17-9	18-6	18-7
	18-8	18-9	18-12	19-6	19-7	19-8	19-12	19-13	19-14	19-15	19-16	19-17
	19-18	19-23	19-24	19-25	19-26	19-27	19-28	21-7	21-8	21-9	21-10	21-11
	21-12	21-13	22-7	22-10	22-11	23-12	24-7	24-8	24-10	25-10	25-11	25-12
	25-13	25-14	25-15	25-16	25-17	25-18	25-19	25-20	25-21	25-22	25-38	28-16
	29-6	29-7	29-8	29-9	29-10	29-11	29-12	29-22	29-23	29-28	30-11	30-12
	30-13	30-19	30-20	30-26	31-6	31-7	31-8	31-9	31-10	31-11	31-14	31-35
	31-36	31-37	31-38	32-5	32-6	32-7	32-8	32-9	32-10	32-13	32-21	32-22
	32-23	32-24	32-25	32-26	32-27	33-11	33-12	33-13	34-12	34-13	34-14	34-15
	34-16	34-17	34-18	34-21	34-28	34-29	34-30	35-7	35-8	35-9	35-10	35-11
	35-12	35-13	35-14	35-17	36-8	36-9	36-10	36-11	36-12	36-15	39-34	39-35
	39-38	40-6	40-7	40-8	40-12	40-13	40-14	40-21	40-22	40-24	41-5	41-6
	41-7	41-8	41-9	41-10	41-30	41-31	41-32	41-33	41-35	42-7	42-8	42-9
	42-10	42-11	42-12	42-20	42-24	42-37	42-40	43-9	43-10	43-11	43-12	43-13
	43-14	43-15	43-16	43-17	43-18	43-20	44-6	44-7	44-8	44-9	44-10	47-11
	47-12	47-13	47-14	47-15	47-16	47-17	47-18	47-19	47-20	48-8	48-9	48-10
	48-11	48-12	48-13	48-14	48-15	48-16	48-17	48-18	48-19	48-21	48-22	48-23
	48-24	49-18	51-12	51-13	51-14	51-15	51-16	51-18	52-7	52-8	52-9	52-10
	52-11	52-13	52-14	52-26	52-27	52-28	52-29	53-7	53-8	53-9	53-10	53-11
	53-12	53-13	54-6	54-7	54-8	54-9	54-10	54-15	54-16	54-17	54-18	54-19

	54-24	54-25	54-26	54-27	54-28	54-30	54-35	54-36	54-38	67-8	67-9	67-15
	67-16	67-17	67-18	67-19	67-20	67-21	67-22	67-25	67-29	67-30	67-31	67-32
	67-33	67-34	67-35	67-37	67-38	67-42	67-44	67-45	67-46	67-47	67-56	67-57
	67-58	67-59	67-60	67-61	67-62	67-63	67-64	67-65	67-66	67-67	67-68	67-69
	67-70	67-71	67-75	67-76	67-77	67-78	67-82	67-83	67-84	67-85	67-90	67-91
	67-96	67-97	67-101	67-102	67-103	67-107	67-108	67-109	67-110	67-111	67-115	67-116
	67-117	67-118	67-119	67-120	67-121	67-122	67-126	67-127	67-128	67-129	67-130	67-131
	67-132	67-133	67-134	67-136	67-137	67-138	67-140	67-141	67-142	67-146	67-147	67-148
	67-149	67-150	67-151	67-152	67-156	67-157	67-158	67-159	67-160	67-161	67-162	67-163
	67-164	67-165	67-166	67-167	67-185	67-186	67-190	67-191	67-192	67-193	67-194	67-195
	67-196	67-197	67-201	67-202	67-203	67-204	67-205	67-206	67-207	67-208	67-209	67-210
	67-211	67-212	67-213	67-214	67-215	67-216	67-217	67-218	67-219	67-220	67-221	67-222
	67-223	67-224	67-225	67-226	67-228	67-237	67-262	67-263	67-264	67-265	67-266	67-267
	67-268	67-269	67-270	67-271	67-272	67-273	67-274	67-275	67-276	67-277	67-278	67-279
	67-280	67-281	67-282	67-283	67-284	67-285	67-286	67-287	67-288	67-289	67-290	67-291
	67-292	67-293	67-294	67-295	67-296	67-315						
RZ	1-19#	14-7	15-5	17-5	18-5	19-5	21-6	22-6	23-8	24-6	25-7	28-11
	29-5	29-21	30-10	30-17	30-24	31-5	31-34	32-4	32-18	33-8	34-11	34-27
	35-6	36-7	39-33	40-5	41-4	41-29	42-6	42-19	42-36	43-8	44-5	47-8
	48-7	49-5	51-7	52-6	52-25	53-6	54-5	54-14	54-23	54-34		