

J E S 2 J O B L O G

```
13.38.49 JOB 152 $HASP373 VTOC      STARTED - INIT 3 - CLASS S - SYS HMVS
13.38.49 JOB 152 IEF403I VTOC - STARTED - TIME=13.38.49
13.38.49 JOB 152 IEFACTRT LOADMACS/IEBUPDTE/00:00:00.04/00:00:00.07/00000/VTOC
13.38.49 JOB 152 IEFACTRT IEBUPDTE/IEBUPDTE/00:00:00.01/00:00:00.01/00000/VTOC
13.38.49 JOB 152 IEFACTRT ASM1      /IFOX00 /00:00:00.22/00:00:00.36/00000/VTOC
13.38.49 JOB 152 IEFACTRT ASM2      /IFOX00 /00:00:00.18/00:00:00.21/00000/VTOC
13.38.50 JOB 152 IEFACTRT ASM3      /IFOX00 /00:00:00.22/00:00:00.26/00000/VTOC
13.38.50 JOB 152 IEFACTRT ASM4      /IFOX00 /00:00:00.17/00:00:00.20/00000/VTOC
13.38.50 JOB 152 IEFACTRT ASM5      /IFOX00 /00:00:00.06/00:00:00.09/00000/VTOC
13.38.50 JOB 152 IEFACTRT ASM6      /IFOX00 /00:00:00.25/00:00:00.30/00000/VTOC
13.38.50 JOB 152 IEFACTRT ASM7      /IFOX00 /00:00:00.13/00:00:00.17/00000/VTOC
13.38.51 JOB 152 IEFACTRT LKED      /IEWL   /00:00:00.04/00:00:00.04/00000/VTOC
13.38.51 JOB 152 IEF404I VTOC - ENDED - TIME=13.38.51
13.38.51 JOB 152 $HASP395 VTOC      ENDED
```

----- JES2 JOB STATISTICS -----

07 JAN 25 JOB EXECUTION DATE

7,847 CARDS READ

14,089 SYSOUT PRINT RECORDS

0 SYSOUT PUNCH RECORDS

0.02 MINUTES EXECUTION TIME

```

1 //VTOC JOB (SYS),'INSTALL VTOC',CLASS=S,MSGCLASS=X JOB 152
***
*** SOURCE: CBT (V129) FILE #112
*** TARGET: SYS2.COMDLIB SYS2.HELP
***
*****
*** This job installs the VTOC TSO command and help. *
*****
***
*** Last update: 2024/11/05 15:55 via Rob Prins
***
*** Note: original VTOC command + some updates to add the RACF and
*** UPDATE indicator and Y2K support.
*** newer VTOC modules from FILE112 from CBT won't assemble
*** correctly on MVS 3.8J with IFOX00.
***
//INSTALL PROC SOUT='*', <=== SYSOUT CLASS
// LIB='SYS2.COMDLIB', <=== TARGET LOAD LIBRARY
// HELP='SYS2.HELP', <=== HELP LIBRARY
// SYSTS=SYSDA, <=== UNITNAME FOR WORK DATASETS
// ASMBLR=IFOX00, <=== NAME OF YOUR ASSEMBLER
// ALIB='SYSC.LINKLIB', <=== LOCATION OF YOUR ASSEMBLER
// MACLIB='SYS1.MACLIB', <=== MACLIB DATASET NAME
// AMODGEN='SYS1.AMODGEN' <=== AMODGEN DATASET NAME
//*
//LOADMACS EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=&SOUT
//SYSUT2 DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(,PASS),
// SPACE=(TRK,(120,,5),RLSE),DCB=(SYS1.MACLIB)
//*
//IEBUPDTE EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=&SOUT
//SYSUT1 DD DSN=&HELP,DISP=SHR
//SYSUT2 DD DSN=&HELP,DISP=SHR
//*
//ASM1 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(,PASS,DELETE),
// SPACE=(TRK,(30,15))
//*
//ASM2 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//ASM3 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')

```

```

//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//ASM4 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//ASM5 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//ASM6 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//ASM7 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
//STEPLIB DD DSN=&ALIB,DISP=SHR
//SYSTEM DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT
//SYSLIB DD DSN=&MACLIB,DISP=SHR
// DD DSN=&AMODGEN,DISP=SHR
// DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
//SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
//*
//LKED EXEC PGM=IEWL,COND=(0,NE),
// PARM='XREF,LET,LIST,SIZE=(600K,64K),RENT,REUS,REFR'

```

```

//SYSPRINT DD SYSOUT=&SOUT
//SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,10)
//SYSLMOD DD DSN=&LIB,DISP=SHR
//SYSLIN DD DSN=&&SYSLIN,DISP=(OLD,DELETE)
// DD DDNAME=SYSIN
// PEND
***
2 // EXEC INSTALL
***
3 ++INSTALL PROC SOUT='*', <=== SYSOUT CLASS
++ LIB='SYS2.COMDLIB', <=== TARGET LOAD LIBRARY
++ HELP='SYS2.HELP', <=== HELP LIBRARY
++ SYSTS=SYSDA, <=== UNITNAME FOR WORK DATASETS
++ ASMBLR=IFOX00, <=== NAME OF YOUR ASSEMBLER
++ ALIB='SYSC.LINKLIB', <=== LOCATION OF YOUR ASSEMBLER
++ MACLIB='SYS1.MACLIB', <=== MACLIB DATASET NAME
++ AMODGEN='SYS1.AMODGEN' <=== AMODGEN DATASET NAME
***
4 ++LOADMACS EXEC PGM=IEBUPDTE,PARM=NEW
5 ++SYSPRINT DD SYSOUT=&SOUT
6 ++SYSUT2 DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(,PASS),
++ SPACE=(TRK,(120,,5),RLSE),DCB=(SYS1.MACLIB)
***
7 //LOADMACS.SYSIN DD *
***----- LOADMACS
***
8 ++IEBUPDTE EXEC PGM=IEBUPDTE,PARM=NEW
9 ++SYSPRINT DD SYSOUT=&SOUT
10 ++SYSUT1 DD DSN=&HELP,DISP=SHR
11 ++SYSUT2 DD DSN=&HELP,DISP=SHR
***
12 //IEBUPDTE.SYSIN DD *
***----- IEBUPDTE
***
13 ++ASM1 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
14 ++STEPLIB DD DSN=&ALIB,DISP=SHR
15 ++SYSTEM DD SYSOUT=&SOUT
16 ++SYSPRINT DD SYSOUT=&SOUT
17 ++SYSLIB DD DSN=&MACLIB,DISP=SHR
18 ++ DD DSN=&AMODGEN,DISP=SHR
19 ++ DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
20 ++SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
21 ++SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
22 ++SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
23 ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(,PASS,DELETE),
++ SPACE=(TRK,(30,15))
***
24 //ASM1.SYSIN DD *
***----- ASM: VTOC
***
25 ++ASM2 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
26 ++STEPLIB DD DSN=&ALIB,DISP=SHR
27 ++SYSTEM DD SYSOUT=&SOUT
28 ++SYSPRINT DD SYSOUT=&SOUT
29 ++SYSLIB DD DSN=&MACLIB,DISP=SHR
30 ++ DD DSN=&AMODGEN,DISP=SHR
31 ++ DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
32 ++SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
33 ++SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
34 ++SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
35 ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)

```

```

36      ***
      //ASM2.SYSIN DD *
      ***----- ASM: VTOCCHEK
      ***
37      ++ASM3      EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
38      ++STEPLIB   DD DSN=&ALIB,DISP=SHR
39      ++SYSTEM   DD SYSOUT=&SOUT
40      ++SYSPRINT  DD SYSOUT=&SOUT
41      ++SYSLIB   DD DSN=&MACLIB,DISP=SHR
42      ++          DD DSN=&AMODGEN,DISP=SHR
43      ++          DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
44      ++SYSUT1   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
45      ++SYSUT2   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
46      ++SYSUT3   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
47      ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
      ***
48      //ASM3.SYSIN DD *
      ***----- ASM: VTOCEXCP
      ***
49      ++ASM4      EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
50      ++STEPLIB   DD DSN=&ALIB,DISP=SHR
51      ++SYSTEM   DD SYSOUT=&SOUT
52      ++SYSPRINT  DD SYSOUT=&SOUT
53      ++SYSLIB   DD DSN=&MACLIB,DISP=SHR
54      ++          DD DSN=&AMODGEN,DISP=SHR
55      ++          DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
56      ++SYSUT1   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
57      ++SYSUT2   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
58      ++SYSUT3   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
59      ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
      ***
60      //ASM4.SYSIN DD *
      ***----- ASM: VTOCFORM
      ***
61      ++ASM5      EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
62      ++STEPLIB   DD DSN=&ALIB,DISP=SHR
63      ++SYSTEM   DD SYSOUT=&SOUT
64      ++SYSPRINT  DD SYSOUT=&SOUT
65      ++SYSLIB   DD DSN=&MACLIB,DISP=SHR
66      ++          DD DSN=&AMODGEN,DISP=SHR
67      ++          DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
68      ++SYSUT1   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
69      ++SYSUT2   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
70      ++SYSUT3   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
71      ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
      ***
72      //ASM5.SYSIN DD *
      ***----- ASM: VTOCMSGX
      ***
73      ++ASM6      EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
74      ++STEPLIB   DD DSN=&ALIB,DISP=SHR
75      ++SYSTEM   DD SYSOUT=&SOUT
76      ++SYSPRINT  DD SYSOUT=&SOUT
77      ++SYSLIB   DD DSN=&MACLIB,DISP=SHR
78      ++          DD DSN=&AMODGEN,DISP=SHR
79      ++          DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
80      ++SYSUT1   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
81      ++SYSUT2   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
82      ++SYSUT3   DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
83      ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
      ***

```

```
84 //ASM6.SYSIN DD *
***----- ASM: VTOCPRNT
***
85 ++ASM7 EXEC PGM=&ASMBLR,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
86 ++STEPLIB DD DSN=&ALIB,DISP=SHR
87 ++SYSTEM DD SYSOUT=&SOUT
88 ++SYSPRINT DD SYSOUT=&SOUT
89 ++SYSLIB DD DSN=&MACLIB,DISP=SHR
90 ++ DD DSN=&AMODGEN,DISP=SHR
91 ++ DD DSN=&&LCLMAC,UNIT=&SYSTS,DISP=(OLD,PASS)
92 ++SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
93 ++SYSUT2 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
94 ++SYSUT3 DD UNIT=&SYSTS,SPACE=(TRK,(30,15))
95 ++SYSPUNCH DD DSN=&&SYSLIN,UNIT=&SYSTS,DISP=(MOD,PASS)
***
96 //ASM7.SYSIN DD *
***----- ASM: VTOCSORT
***
97 ++LKED EXEC PGM=IEWL,COND=(0,NE),
++ PARM='XREF,LET,LIST,SIZE=(600K,64K),RENT,REUS,REFR'
98 ++SYSPRINT DD SYSOUT=&SOUT
99 ++SYSUT1 DD UNIT=&SYSTS,SPACE=(TRK,10)
100 ++SYSLMOD DD DSN=&LIB,DISP=SHR
101 ++SYSLIN DD DSN=&&SYSLIN,DISP=(OLD,DELETE)
102 ++ DD DDNAME=SYSIN
103 //LKED.SYSIN DD *
***----- LKED
//
```

STMT NO. MESSAGE

```
5 IEF653I SUBSTITUTION JCL - SYSOUT=*
6 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(,PASS),
9 IEF653I SUBSTITUTION JCL - SYSOUT=*
10 IEF653I SUBSTITUTION JCL - DSN=SYS2.HELP,DISP=SHR
11 IEF653I SUBSTITUTION JCL - DSN=SYS2.HELP,DISP=SHR
13 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
14 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
15 IEF653I SUBSTITUTION JCL - SYSOUT=*
16 IEF653I SUBSTITUTION JCL - SYSOUT=*
17 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
18 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
19 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
20 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
21 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
22 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
23 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(,PASS,DELETE),
25 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
26 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
27 IEF653I SUBSTITUTION JCL - SYSOUT=*
28 IEF653I SUBSTITUTION JCL - SYSOUT=*
29 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
30 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
31 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
32 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
33 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
34 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
35 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
37 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
38 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
39 IEF653I SUBSTITUTION JCL - SYSOUT=*
40 IEF653I SUBSTITUTION JCL - SYSOUT=*
41 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
42 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
43 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
44 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
45 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
46 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
47 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
49 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
50 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
51 IEF653I SUBSTITUTION JCL - SYSOUT=*
52 IEF653I SUBSTITUTION JCL - SYSOUT=*
53 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
54 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
55 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
56 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
57 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
58 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
59 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
61 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
62 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
63 IEF653I SUBSTITUTION JCL - SYSOUT=*
64 IEF653I SUBSTITUTION JCL - SYSOUT=*
65 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
66 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
67 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
68 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
69 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
70 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
```

```

71 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
73 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
74 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
75 IEF653I SUBSTITUTION JCL - SYSOUT=*
76 IEF653I SUBSTITUTION JCL - SYSOUT=*
77 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
78 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
79 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
80 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
81 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
82 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
83 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
85 IEF653I SUBSTITUTION JCL - PGM=IFOX00,REGION=2048K,PARM=(NOLOAD,DECK,'LINECNT=55')
86 IEF653I SUBSTITUTION JCL - DSN=SYSC.LINKLIB,DISP=SHR
87 IEF653I SUBSTITUTION JCL - SYSOUT=*
88 IEF653I SUBSTITUTION JCL - SYSOUT=*
89 IEF653I SUBSTITUTION JCL - DSN=SYS1.MACLIB,DISP=SHR
90 IEF653I SUBSTITUTION JCL - DSN=SYS1.AMODGEN,DISP=SHR
91 IEF653I SUBSTITUTION JCL - DSN=&&LCLMAC,UNIT=SYSDA,DISP=(OLD,PASS)
92 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
93 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
94 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,(30,15))
95 IEF653I SUBSTITUTION JCL - DSN=&&SYSLIN,UNIT=SYSDA,DISP=(MOD,PASS)
98 IEF653I SUBSTITUTION JCL - SYSOUT=*
99 IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,10)
100 IEF653I SUBSTITUTION JCL - DSN=SYS2.CMDLIB,DISP=SHR
IEF236I ALLOC. FOR VTOC LOADMACS
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 252 ALLOCATED TO SYSUT2
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC LOADMACS - STEP WAS EXECUTED - COND CODE 0000
IEF285I JES2.JOB00152.SO0111 SYSOUT
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED *-----290
IEF285I VOL SER NOS= WORK01.
IEF285I JES2.JOB00152.SI0101 SYSIN
IEF373I STEP /LOADMACS/ START 25007.1338
IEF374I STEP /LOADMACS/ STOP 25007.1338 CPU 0MIN 00.03SEC SRB 0MIN 00.01SEC VIRT 48K SYS 244K
**** JOB NAME: VTOC JOBCARD READ 2025/007 13:38:48 370/148 VS2 R03.8 HMVS *****
*
* STEP NUMBER: 1 USER CORE: 48K START TIME: 13:38:49 CPU TIME: 00:00:00.04 ACTIVE TIME: 00:00:00.05 *
* STEP NAME: LOADMACS SYSTEM CORE: 244K STOP TIME: 13:38:49 SRB TIME: 00:00:00.01 ALLOC TIME: 13:38:49 *
* PROGRAM NAME: IEBUPDTE REGION SIZE: 512K ELAPSED TIME: 00:00:00.07 TCB TIME: 00:00:00.03 PROGRAM LOAD: 13:38:49 *
* CONDITION CODE: 00000 PERFORMANCE GROUP: 003 *
* JES2 CARDS: 331 SERVICE UNITS PAGES IN/OUT # SWAPS PAGES SWAP IN/OUT VIO PAGES IN/OUT *
* 1,956 0 / 0 0 0 / 0 0 / 0 *
*
* ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT *
* 252/D3350 290 *
*****
IEF236I ALLOC. FOR VTOC IEBUPDTE
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 151 ALLOCATED TO SYSUT1
IEF237I 151 ALLOCATED TO SYSUT2
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC IEBUPDTE - STEP WAS EXECUTED - COND CODE 0000
IEF285I JES2.JOB00152.SO0112 SYSOUT
IEF285I SYS2.HELP KEPT *-----0
IEF285I VOL SER NOS= MVS000.
IEF285I SYS2.HELP KEPT *-----5
IEF285I VOL SER NOS= MVS000.
IEF285I JES2.JOB00152.SI0102 SYSIN

```



```

IEF373I STEP /IEBUPDTE/ START 25007.1338
IEF374I STEP /IEBUPDTE/ STOP 25007.1338 CPU      OMIN 00.01SEC SRB      OMIN 00.00SEC VIRT      48K SYS      252K
*****
*
* STEP NUMBER:          2  USER CORE:          48K  START TIME:    13:38:49      CPU TIME:      00:00:00.01  ACTIVE TIME:   00:00:00.01 *
* STEP NAME:           IEBUPDTE  SYSTEM CORE:    252K  STOP TIME:     13:38:49      SRB TIME:      00:00:00.00  ALLOC TIME:   13:38:49 *
* PROGRAM NAME:       IEBUPDTE  REGION SIZE:    512K  ELAPSED TIME: 00:00:00.01  TCB TIME:      00:00:00.01  PROGRAM LOAD: 13:38:49 *
* CONDITION CODE:     00000  PERFORMANCE GROUP: 003
*
* JES2 CARDS:          35          SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT *
*                               107          0 /    0          0          0 /    0          0 /    0 *
*
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT *
* 151/D3350            0 151/D3350            5
*****
IEF236I ALLOC. FOR VTOC ASM1
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00002
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150  ALLOCATED TO SYSLIB
IEF237I 250  ALLOCATED TO
IEF237I 252  ALLOCATED TO
IEF237I 390  ALLOCATED TO SYSUT1
IEF237I 280  ALLOCATED TO SYSUT2
IEF237I 281  ALLOCATED TO SYSUT3
IEF237I 251  ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM1 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB          KEPT          *-----0
IEF285I VOL SER NOS= SYSCPK.          KEPT          *-----0
IEF285I UCSYSCPK              KEPT          *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I JES2.JOB00152.SO0113          SYSOUT
IEF285I JES2.JOB00152.SO0114          SYSOUT
IEF285I SYS1.MACLIB            KEPT          *-----70
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN            KEPT          *-----23
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC  PASSED        *-----24
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000001  DELETED       *-----680
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000002  DELETED       *-----25
IEF285I VOL SER NOS= MVS380.
IEF285I SYS25007.T133849.RA000.VTOC.R0000003  DELETED       *-----18
IEF285I VOL SER NOS= MVS381.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN  PASSED        *-----129
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0103          SYSIN
IEF373I STEP /ASM1 / START 25007.1338
IEF374I STEP /ASM1 / STOP 25007.1338 CPU      OMIN 00.20SEC SRB      OMIN 00.02SEC VIRT     2048K SYS     276K
*****
*
* STEP NUMBER:          3  USER CORE:          2048K  START TIME:    13:38:49      CPU TIME:      00:00:00.22  ACTIVE TIME:   00:00:00.34 *
* STEP NAME:           ASM1      SYSTEM CORE:    276K  STOP TIME:     13:38:49      SRB TIME:      00:00:00.02  ALLOC TIME:   13:38:49 *
* PROGRAM NAME:       IFOX00  REGION SIZE:    2048K  ELAPSED TIME: 00:00:00.36  TCB TIME:      00:00:00.20  PROGRAM LOAD: 13:38:49 *
* CONDITION CODE:     00000  PERFORMANCE GROUP: 003
*
* JES2 CARDS:          86          SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT *
*                               5,273          0 /    0          0          0 /    0          0 /    0 *
*
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT *

```

```

* 253/D3350      0 253/D3350      0 150/D3350      70 250/D3350      23 252/D3350      24 390/D3390      680 *
* 280/D3380      25 281/D3380      18 251/D3350      129 *
*****
IEF236I ALLOC. FOR VTOC ASM2
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00004
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150  ALLOCATED TO SYSLIB
IEF237I 250  ALLOCATED TO
IEF237I 252  ALLOCATED TO
IEF237I 390  ALLOCATED TO SYSUT1
IEF237I 281  ALLOCATED TO SYSUT2
IEF237I 280  ALLOCATED TO SYSUT3
IEF237I 251  ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM2 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB          KEPT          *-----0
IEF285I VOL SER NOS= SYSCP.          KEPT          *-----0
IEF285I UCSYSCPK              KEPT          *-----0
IEF285I VOL SER NOS= SYSCP.
IEF285I JES2.JOB00152.SO0115        SYSOUT
IEF285I JES2.JOB00152.SO0116        SYSOUT
IEF285I SYS1.MACLIB            KEPT          *-----47
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN          KEPT          *-----14
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED        *-----24
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000004 DELETED        *-----603
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000005 DELETED        *-----19
IEF285I VOL SER NOS= MVS381.
IEF285I SYS25007.T133849.RA000.VTOC.R0000006 DELETED        *-----14
IEF285I VOL SER NOS= MVS380.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN PASSED        *-----92
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0104        SYSIN
IEF373I STEP /ASM2      / START 25007.1338
IEF374I STEP /ASM2      / STOP 25007.1338 CPU      0MIN 00.16SEC SRB      0MIN 00.02SEC VIRT 2048K SYS 280K
*****
*
* STEP NUMBER:          4  USER CORE:          2048K  START TIME:    13:38:49      CPU TIME:      00:00:00.18  ACTIVE TIME:   00:00:00.19 *
* STEP NAME:          ASM2  SYSTEM CORE:        280K  STOP TIME:     13:38:49      SRB TIME:      00:00:00.02  ALLOC TIME:    13:38:49 *
* PROGRAM NAME:      IFOX00  REGION SIZE:      2048K  ELAPSED TIME:  00:00:00.21  TCB TIME:     00:00:00.16  PROGRAM LOAD:  13:38:49 *
* CONDITION CODE:    00000  PERFORMANCE GROUP: 003 *
* JES2 CARDS:          79      SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT *
*                               4,427      0 / 0      0      0 / 0      0 / 0 *
*
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT *
* 253/D3350      0 253/D3350      0 150/D3350      47 250/D3350      14 252/D3350      24 390/D3390      603 *
* 281/D3380      19 280/D3380      14 251/D3350      92 *
*****
IEF236I ALLOC. FOR VTOC ASM3
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00006
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150  ALLOCATED TO SYSLIB
IEF237I 250  ALLOCATED TO
IEF237I 252  ALLOCATED TO

```

```

IEF237I 390 ALLOCATED TO SYSUT1
IEF237I 281 ALLOCATED TO SYSUT2
IEF237I 280 ALLOCATED TO SYSUT3
IEF237I 251 ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM3 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I UCSYSCPK KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I JES2.JOB00152.SO0117 SYSOUT
IEF285I JES2.JOB00152.SO0118 SYSOUT
IEF285I SYS1.MACLIB KEPT *-----121
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN KEPT *-----59
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED *-----100
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000007 DELETED *-----466
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000008 DELETED *-----29
IEF285I VOL SER NOS= MVS381.
IEF285I SYS25007.T133849.RA000.VTOC.R0000009 DELETED *-----12
IEF285I VOL SER NOS= MVS380.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN PASSED *-----38
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0105 SYSIN
IEF373I STEP /ASM3 / START 25007.1338
IEF374I STEP /ASM3 / STOP 25007.1338 CPU 0MIN 00.20SEC SRB 0MIN 00.02SEC VIRT 2048K SYS 284K
*****
*
* STEP NUMBER: 5 USER CORE: 2048K START TIME: 13:38:49 CPU TIME: 00:00:00.22 ACTIVE TIME: 00:00:00.24 *
* STEP NAME: ASM3 SYSTEM CORE: 284K STOP TIME: 13:38:50 SRB TIME: 00:00:00.02 ALLOC TIME: 13:38:49 *
* PROGRAM NAME: IFOX00 REGION SIZE: 2048K ELAPSED TIME: 00:00:00.26 TCB TIME: 00:00:00.20 PROGRAM LOAD: 13:38:49 *
* CONDITION CODE: 00000 PERFORMANCE GROUP: 003 *
* JES2 CARDS: 56 SERVICE UNITS PAGES IN/OUT # SWAPS PAGES SWAP IN/OUT VIO PAGES IN/OUT *
* 4,476 0 / 0 0 0 / 0 0 / 0 *
*
* ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT *
* 253/D3350 0 253/D3350 0 150/D3350 121 250/D3350 59 252/D3350 100 390/D3390 466 *
* 281/D3380 29 280/D3380 12 251/D3350 38 *
*****
IEF236I ALLOC. FOR VTOC ASM4
IEF237I 253 ALLOCATED TO STEPLIB
IEF237I 253 ALLOCATED TO SYS00008
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150 ALLOCATED TO SYSLIB
IEF237I 250 ALLOCATED TO
IEF237I 252 ALLOCATED TO
IEF237I 390 ALLOCATED TO SYSUT1
IEF237I 280 ALLOCATED TO SYSUT2
IEF237I 281 ALLOCATED TO SYSUT3
IEF237I 251 ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM4 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I UCSYSCPK KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I JES2.JOB00152.SO0119 SYSOUT

```

```

IEF285I JES2.JOB00152.S00120          SYSOUT
IEF285I SYS1.MACLIB                    KEPT          *-----48
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN                    KEPT          *-----14
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED        *-----24
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000010 DELETED        *-----539
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000011 DELETED        *-----17
IEF285I VOL SER NOS= MVS380.
IEF285I SYS25007.T133849.RA000.VTOC.R0000012 DELETED        *-----12
IEF285I VOL SER NOS= MVS381.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN PASSED        *-----64
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0106          SYSIN
IEF373I STEP /ASM4      / START 25007.1338
IEF374I STEP /ASM4      / STOP 25007.1338 CPU      0MIN 00.15SEC SRB      0MIN 00.02SEC VIRT 2048K SYS 280K
*****
*
* STEP NUMBER:          6  USER CORE:          2048K  START TIME:    13:38:50      CPU TIME:      00:00:00.17  ACTIVE TIME:  00:00:00.17 *
* STEP NAME:           ASM4  SYSTEM CORE:       280K  STOP TIME:     13:38:50      SRB TIME:      00:00:00.02  ALLOC TIME:   13:38:50 *
* PROGRAM NAME:        IFOX00 REGION SIZE:      2048K  ELAPSED TIME:  00:00:00.20  TCB TIME:     00:00:00.15  PROGRAM LOAD: 13:38:50 *
* CONDITION CODE:     00000 PERFORMANCE GROUP: 003
* JES2 CARDS:         52          SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT *
*                               3,860      0 / 0          0          0 / 0          0 / 0 *
*
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT *
* 253/D3350           0 253/D3350           0 150/D3350           48 250/D3350           14 252/D3350           24 390/D3390           539 *
* 280/D3380           17 281/D3380           12 251/D3350           64
*****
IEF236I ALLOC. FOR VTOC ASM5
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00010
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150  ALLOCATED TO SYSLIB
IEF237I 250  ALLOCATED TO
IEF237I 252  ALLOCATED TO
IEF237I 390  ALLOCATED TO SYSUT1
IEF237I 280  ALLOCATED TO SYSUT2
IEF237I 281  ALLOCATED TO SYSUT3
IEF237I 251  ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM5 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB                    KEPT          *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I UCSYSCPK                        KEPT          *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I JES2.JOB00152.S00121          SYSOUT
IEF285I JES2.JOB00152.S00122          SYSOUT
IEF285I SYS1.MACLIB                    KEPT          *-----30
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN                    KEPT          *-----8
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED        *-----11
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000013 DELETED        *-----70
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000014 DELETED        *-----9
IEF285I VOL SER NOS= MVS380.

```

```

IEF285I  SYS25007.T133849.RA000.VTOC.R0000015      DELETED      *-----8
IEF285I  VOL SER NOS= MVS381.
IEF285I  SYS25007.T133849.RA000.VTOC.SYSLIN        PASSED      *-----8
IEF285I  VOL SER NOS= WORK00.
IEF285I  JES2.JOB00152.SI0107                      SYSIN
IEF373I  STEP /ASM5      / START 25007.1338
IEF374I  STEP /ASM5      / STOP  25007.1338 CPU    OMIN 00.06SEC SRB    OMIN 00.00SEC VIRT  2048K SYS   260K
*****
*
* STEP NUMBER:          7  USER CORE:          2048K  START TIME:    13:38:50    CPU TIME:      00:00:00.06  ACTIVE TIME:   00:00:00.07 *
* STEP NAME:           ASM5    SYSTEM CORE:      260K  STOP TIME:     13:38:50    SRB TIME:      00:00:00.00  ALLOC TIME:    13:38:50 *
* PROGRAM NAME:       IFOX00   REGION SIZE:  2048K  ELAPSED TIME:  00:00:00.09  TCB TIME:      00:00:00.06  PROGRAM LOAD:  13:38:50 *
* CONDITION CODE:    00000  PERFORMANCE GROUP: 003
* JES2 CARDS:        5          SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT *
*                               822      0 /    0      0          0 /    0          0 /    0 *
*
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT *
* 253/D3350           0 253/D3350           0 150/D3350           30 250/D3350           8 252/D3350           11 390/D3390           70 *
* 280/D3380           9 281/D3380           8 251/D3350           8
*****
IEF236I  ALLOC. FOR VTOC ASM6
IEF237I  253  ALLOCATED TO STEPLIB
IEF237I  253  ALLOCATED TO SYS00012
IEF237I  JES2 ALLOCATED TO SYSTEM
IEF237I  JES2 ALLOCATED TO SYSPRINT
IEF237I  150  ALLOCATED TO SYSLIB
IEF237I  250  ALLOCATED TO
IEF237I  252  ALLOCATED TO
IEF237I  390  ALLOCATED TO SYSUT1
IEF237I  281  ALLOCATED TO SYSUT2
IEF237I  280  ALLOCATED TO SYSUT3
IEF237I  251  ALLOCATED TO SYSPUNCH
IEF237I  JES2 ALLOCATED TO SYSIN
IEF142I  VTOC ASM6 - STEP WAS EXECUTED - COND CODE 0000
IEF285I  SYSC.LINKLIB      KEPT      *-----0
IEF285I  VOL SER NOS= SYSCPK.
IEF285I  UCSYSCPK          KEPT      *-----0
IEF285I  VOL SER NOS= SYSCPK.
IEF285I  JES2.JOB00152.SO0123  SYSOUT
IEF285I  JES2.JOB00152.SO0124  SYSOUT
IEF285I  SYS1.MACLIB        KEPT      *-----81
IEF285I  VOL SER NOS= MVSRES.
IEF285I  SYS1.AMODGEN        KEPT      *-----17
IEF285I  VOL SER NOS= SMP000.
IEF285I  SYS25007.T133849.RA000.VTOC.LCLMAC  PASSED      *-----30
IEF285I  VOL SER NOS= WORK01.
IEF285I  SYS25007.T133849.RA000.VTOC.R0000016  DELETED      *-----761
IEF285I  VOL SER NOS= WORK03.
IEF285I  SYS25007.T133849.RA000.VTOC.R0000017  DELETED      *-----33
IEF285I  VOL SER NOS= MVS381.
IEF285I  SYS25007.T133849.RA000.VTOC.R0000018  DELETED      *-----16
IEF285I  VOL SER NOS= MVS380.
IEF285I  SYS25007.T133849.RA000.VTOC.SYSLIN  PASSED      *-----104
IEF285I  VOL SER NOS= WORK00.
IEF285I  JES2.JOB00152.SI0108  SYSIN
IEF373I  STEP /ASM6      / START 25007.1338
IEF374I  STEP /ASM6      / STOP  25007.1338 CPU    OMIN 00.23SEC SRB    OMIN 00.02SEC VIRT  2048K SYS   264K
*****
*
* STEP NUMBER:          8  USER CORE:          2048K  START TIME:    13:38:50    CPU TIME:      00:00:00.25  ACTIVE TIME:   00:00:00.27 *
* STEP NAME:           ASM6    SYSTEM CORE:      264K  STOP TIME:     13:38:50    SRB TIME:      00:00:00.02  ALLOC TIME:    13:38:50 *

```

```

* PROGRAM NAME:  IFOX00  REGION SIZE:  2048K  ELAPSED TIME: 00:00:00.30  TCB TIME:  00:00:00.23  PROGRAM LOAD: 13:38:50  *
* CONDITION CODE:  00000  PERFORMANCE GROUP: 003  *
* JES2 CARDS:  104  SERVICE UNITS  PAGES IN/OUT  # SWAPS  PAGES SWAP IN/OUT  VIO PAGES IN/OUT  *
* 5,670  0 / 0  0  0 / 0  0 / 0  *
* ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  ADDR/UNIT I/O COUNT  *
* 253/D3350  0 253/D3350  0 150/D3350  81 250/D3350  17 252/D3350  30 390/D3390  761  *
* 281/D3380  33 280/D3380  16 251/D3350  104  *
*****
IEF236I ALLOC. FOR VTOC ASM7
IEF237I 253 ALLOCATED TO STEPLIB
IEF237I 253 ALLOCATED TO SYS00014
IEF237I JES2 ALLOCATED TO SYSTEM
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 150 ALLOCATED TO SYSLIB
IEF237I 250 ALLOCATED TO
IEF237I 252 ALLOCATED TO
IEF237I 390 ALLOCATED TO SYSUT1
IEF237I 281 ALLOCATED TO SYSUT2
IEF237I 280 ALLOCATED TO SYSUT3
IEF237I 251 ALLOCATED TO SYSPUNCH
IEF237I JES2 ALLOCATED TO SYSIN
IEF142I VTOC ASM7 - STEP WAS EXECUTED - COND CODE 0000
IEF285I SYSC.LINKLIB KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I UCSYSCPK KEPT *-----0
IEF285I VOL SER NOS= SYSCPK.
IEF285I JES2.JOB00152.SO0125 SYSOUT
IEF285I JES2.JOB00152.SO0126 SYSOUT
IEF285I SYS1.MACLIB KEPT *-----39
IEF285I VOL SER NOS= MVSRES.
IEF285I SYS1.AMODGEN KEPT *-----12
IEF285I VOL SER NOS= SMP000.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC PASSED *-----20
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.R0000019 DELETED *-----478
IEF285I VOL SER NOS= WORK03.
IEF285I SYS25007.T133849.RA000.VTOC.R0000020 DELETED *-----15
IEF285I VOL SER NOS= MVS381.
IEF285I SYS25007.T133849.RA000.VTOC.R0000021 DELETED *-----12
IEF285I VOL SER NOS= MVS380.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN PASSED *-----45
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0109 SYSIN
IEF373I STEP /ASM7 / START 25007.1338
IEF374I STEP /ASM7 / STOP 25007.1338 CPU 0MIN 00.12SEC SRB 0MIN 00.01SEC VIRT 2048K SYS 264K
*****
*
* STEP NUMBER: 9 USER CORE: 2048K START TIME: 13:38:50 CPU TIME: 00:00:00.13 ACTIVE TIME: 00:00:00.14 *
* STEP NAME: ASM7 SYSTEM CORE: 264K STOP TIME: 13:38:50 SRB TIME: 00:00:00.01 ALLOC TIME: 13:38:50 *
* PROGRAM NAME: IFOX00 REGION SIZE: 2048K ELAPSED TIME: 00:00:00.17 TCB TIME: 00:00:00.12 PROGRAM LOAD: 13:38:50 *
* CONDITION CODE: 00000 PERFORMANCE GROUP: 003 *
* JES2 CARDS: 16 SERVICE UNITS PAGES IN/OUT # SWAPS PAGES SWAP IN/OUT VIO PAGES IN/OUT *
* 3,269 0 / 0 0 0 / 0 0 / 0 *
* ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT *
* 253/D3350 0 253/D3350 0 150/D3350 39 250/D3350 12 252/D3350 20 390/D3390 478 *
* 281/D3380 15 280/D3380 12 251/D3350 45 *
*****
IEF236I ALLOC. FOR VTOC LKED
IEF237I JES2 ALLOCATED TO SYSPRINT

```

```

IEF237I 390 ALLOCATED TO SYSUT1
IEF237I 151 ALLOCATED TO SYSLMOD
IEF237I 251 ALLOCATED TO SYSLIN
IEF237I JES2 ALLOCATED TO
IEF142I VTOC LKED - STEP WAS EXECUTED - COND CODE 0000
IEF285I JES2.JOB00152.S00127 SYSOUT
IEF285I SYS25007.T133849.RA000.VTOC.R0000022 DELETED *-----0
IEF285I VOL SER NOS= WORK03.
IEF285I SYS2.CMDLIB KEPT *-----11
IEF285I VOL SER NOS= MVS000.
IEF285I SYS25007.T133849.RA000.VTOC.SYSLIN DELETED *-----481
IEF285I VOL SER NOS= WORK00.
IEF285I JES2.JOB00152.SI0110 SYSIN
IEF373I STEP /LKED / START 25007.1338
IEF374I STEP /LKED / STOP 25007.1338 CPU 0MIN 00.03SEC SRB 0MIN 00.01SEC VIRT 512K SYS 248K
*****
*
* STEP NUMBER: 10 USER CORE: 512K START TIME: 13:38:50 CPU TIME: 00:00:00.04 ACTIVE TIME: 00:00:00.04 *
* STEP NAME: LKED SYSTEM CORE: 248K STOP TIME: 13:38:51 SRB TIME: 00:00:00.01 ALLOC TIME: 13:38:50 *
* PROGRAM NAME: IEWL REGION SIZE: 512K ELAPSED TIME: 00:00:00.04 TCB TIME: 00:00:00.03 PROGRAM LOAD: 13:38:50 *
* CONDITION CODE: 00000 PERFORMANCE GROUP: 003 *
* JES2 CARDS: 0 SERVICE UNITS PAGES IN/OUT # SWAPS PAGES SWAP IN/OUT VIO PAGES IN/OUT *
* 2,503 0 / 0 0 0 / 0 0 / 0 *
*
* ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT ADDR/UNIT I/O COUNT *
* 390/D3390 0 151/D3350 11 251/D3350 481 *
*****
IEF237I 252 ALLOCATED TO SYS00001
IEF285I SYS25007.T133851.RA000.VTOC.R0000001 KEPT *-----0
IEF285I VOL SER NOS= WORK01.
IEF285I SYS25007.T133849.RA000.VTOC.LCLMAC DELETED
IEF285I VOL SER NOS= WORK01.
IEF375I JOB /VTOC / START 25007.1338
IEF376I JOB /VTOC / STOP 25007.1338 CPU 0MIN 01.19SEC SRB 0MIN 00.13SEC

```

```

./      ADD NAME=ALLOC
        MACRO
&NAME  ALLOC  &DUMMY, &PERM, &DSN=, &DDN=, &DDNRET=, &MEMBER=, &DISP=,      X
        &VOL=, &UNIT=, &SYSOUT=, &FREE=, &COPIES=, &LABEL=,                X
        &BLKSIZE=, &DEN=, &DSORG=, &KEYLEN=, &LRECL=, &RECFM=,            X
        &PASSWORD=, &DSNRET=, &MF=AUTO, &PREFIX=, &ERROR=,                X
        &SPACE=, &F=, &FILE=, &DA=, &QNAME=, &DSORGRT=,                  X
        &VOLRET=, &DCBDSN=, &DCBDDN=, &SPECIAL=, &DDNTO=,                X
        &FORMS=, &DEST=, &SSREQ=, &FORUSER=, &TU=, &DSNPDE=
*****
.*
.*      THIS MACRO PROVIDES A DYNAMIC ALLOCATION FUNCTION BY BUILDING      *
.*      A DYNAMIC ALLOCATION PARAMETER LIST AND INVOKING SVC 99.          *
.*      IT FIRST SETS UP A WORKAREA ENVIRONMENT FOR THE PARAMETER LIST   *
.*      AND THEN TESTS THE KEYWORDS SUPPLIED AND INVOKES INNER MACROS    *
.*      TO BUILD THE TEXT UNITS. THE INNER MACROS THEMSELVES USE INNER   *
.*      MACROS TO UPDATE GLOBAL VARIABLES, STORE TEXT UNIT POINTERS ETC  *
.*      THERE ARE THREE WAYS OF SPECIFYING THE WORK AREA ADDRESS.        *
.*      A) MF=AUTO, MF=G, MF=(E,ADDRESS,LNTHSYMB).                       *
.*      IN THE FIRST FORM, AN INNER MACRO DYNSPACE IS CALLED TO NAME     *
.*      A WORK AREA, THE NAME BEING RETURNED IN THE GLOBAL SETC         *
.*      VARIABLE &DYNSP. A DSECT IS CREATED TO MAP THIS AREA.           *
.*      THE GLOBAL VARIABLES &DTUO (TEXT UNIT OFFSET COUNTER) AND        *
.*      &DTUPO (TEXT UNIT POINTER OFFSET ACCUMULATOR) ARE SET TO ZERO.  *
.*      THESE ACCUMULATORS ARE UPDATED AS EACH TEXT UNIT PROCESSOR      *
.*      ACQUIRES STORAGE. AFTER ALL TEXT UNITS HAVE BEEN BUILT, THE     *
.*      AMOUNT OF SPACE USED IS CALCULATED, AND THE DYNSPACE MACRO IS    *
.*      THEN CALLED AGAIN TO LOG THE AMOUNT NEEDED. DYNSPACE SETS A     *
.*      GLOBAL VARIABLE &DYNSPQ TO THE HIGHEST AMOUNT ANY ALLOC OR      *
.*      FREE MACRO REQUESTED, AND WHEN CALLED WITH THE EXPAND OPTION,    *
.*      (NO OPERANDS OR NAME FIELD SUPPLIED), EXPANDS INTO A DS FOR     *
.*      THAT QUANTITY. (SEE DYNSPACE)                                     *
.*      MF=G SPECIFIES THAT THE ALLOC MACRO ENTER THE BEGIN MACRO       *
.*      WORKAREA TO ACQUIRE THE STORAGE NECESSARY. IT DOES THIS VIA    *
.*      THE RCPDS MACRO. (SEE RCPDS). HOWEVER, IF THE ALLOC MACRO IS    *
.*      CALLED SEVERAL TIMES WITH THIS OPTION, A LOT OF STORAGE WILL BE *
.*      USED UP, AS THE STORAGE WILL NOT BE SHARED. THUS, THIS FORM     *
.*      SHOULD ONLY BE USED IF THE ALLOC/FREE MACRO IS ONLY TO BE USED  *
.*      ONCE OR TWICE DURING AN ASSEMBLY.                                *
.*      MF=E CAUSES THE MACRO TO USE A USER SPECIFIED WORK AREA. THE    *
.*      SECOND PARAMETER GIVES THE NAME OF THE WORKAREA, AND AN         *
.*      OPTIONAL THIRD PARAMETER IS THE NAME OF A SYMBOL TO BE EQUATED   *
.*      TO THE LENGTH OF THE REQUIRED WORK AREA.                           *
.*
.*      DYNAMIC ALLOCATION FUNCTIONS ARE SIMILAR TO THOSE AVAILABLE      *
.*      WITH JCL, USING THE SAME KEYWORDS. HOWEVER, CERTAIN FORMATS     *
.*      ARE SLIGHTLY DIFFERENT. FOR INSTANCE, CERTAIN KEYWORDS CAN      *
.*      HAVE VARYING PARAMETERS, EG DATASET NAME, DDNAME, VOLSER ETC.   *
.*      PROVISION IS MADE FOR BOTH VARIABLE SPECIFICATION.              *
.*      IN THE ABSOLUTE FORM, THE PARAMETER IS ENTERED IN QUOTES,      *
.*      E.G.  ALLOC DSN='SYS1.LINKLIB',DISP=SHR                          *
.*      HOWEVER, THIS NAME REMAINS FIXED FOR THE ASSEMBLY.              *

```



```

.*      IN THE VARIABLE FORMAT, THE ADDRESS OF A LOCATOR IS SPECIFIED, *
.*      WHERE THE LOCATOR CONSISTS OF A SIX BYTE FIELD, THE FIRST 4 *
.*      BYTES OF WHICH POINT TO THE PARAMETER, WHILE THE NEXT TWO *
.*      CONTAIN THE LENGTH. *
.*      EG      ALLOC DSN=LOCATOR *
.*      LOCATOR DC      A(DSN),Y(12) *
.*      DSN      DC      C'SYS1.LINKLIB' *
.* *
.*      NUMERIC QUANTITIES E.G. COPIES= FOR SYSOUT, SHOULD EITHER *
.*      SPECIFY A NUMERIC VALUE, COPIES=3, *
.*      A VALUE IN A REGISTER, COPIES=(R3), *
.*      OR THE NAME OF A FULLWORD CONTAINING THE VALUE, *
.*      COPIES=NUMCOPYS, WHERE NUMCOPYS IS THE NAME OF A *
.*      FULLWORD FIELD. *
.* *
.*      OTHER KEYWORDS SUCH AS DISP= CAN ONLY HAVE THE ABSOLUTE *
.*      FORM, AND VALUES SHOULD NOT BE ENTERED WITHIN QUOTES. *
.*      ADDITIONAL FACILITIES NOT AVAILABLE WITH JCL ARE THE *
.*      RETURN BY THE SYSTEM OF INFORMATION ON THE DATASET, EG *
.*      DSORG. THIS IS DONE BY SPECIFYING DSORGRT=SYMBOL, WHERE *
.*      SYMBOL IS A SYMBOL WHICH WILL BE EQUATED TO A TWO BYTE *
.*      FIELD CONTAINING THE DSORG TYPE (SEE JOB MANAGEMENT, *
.*      SUPERVISOR AND TSO). *
.*      THE SYSTEM CAN ALSO GENERATE AND RETURN A DDNAME. THIS IS *
.*      CARRIED OUT BY ENTERING DDNTO=(ADDR1,ADDR2,,...) *
.*      WHERE ADDR1,ADDR2 ETC ARE THE NAMES OF 8 BYTE FIELDS WHICH *
.*      ARE TO RECEIVE THE DDNAME. *
.*      FOR FURTHER INFORMATION ON DYNAMIC ALLOCATION, SEE *
.*      JOB MANAGEMENT, SUPERVISOR AND TSO. *

```

```

*****
GBLA  &RCPDYN      COUNTER FOR NO ENTRIES TO MACRO
GBLA  &DTUO        OFFSET TO TEXT UNITS
GBLA  &DTUPO       OFFSET TO TEXT UNIT POINTERS
GBLB  &RCPS99(2)   TELL RCPDSECT NEED DSECTS
GBLC  &DYNP        PREFIX FOR LABELS FOR THIS CALL
GBLC  &DYNP        PREFIX FOR LABELS FOR THIS CALL
GBLC  &DYNP        NAME FOR AUTOMATIC STORAGE ALLOC
LCLA  &DDNRTO,&DSNRTO  FOR EQUATES FOR RETURNED FLDS
LCLA  &VOLRTO,&DSRGRTO  FOR EQUATES FOR RETURNED FIELDS
LCLA  &I           COUNTER
LCLB  &DSECT       DSECT NEEDED FOR STORAGE, MF=E
LCLC  &C,&T,&PAR

```

```

.*
.*      THE ALLOC MACRO PROVIDES A DYNAMIC ALLOCATION FUNCTION,
&RCPS99(1)  SETB      1
&RCPDYN  SETA  &RCPDYN+1      INCEREMENT COUNTER
&DYNP    SETC  'DYN&RCPDYN'  SET DEFAULT PREFIX
&NAME    DS      0H
AIF      ('&PREFIX' EQ ' ').TMF
AIF      (K'&PREFIX LT 4).POK
MNOTE   4,'PREFIX TOO LONG, 1ST 4 CHARS USED'
&DYNP   SETC  '&PREFIX'(1,4)

```

```

AGO .TMF
.POK ANOP
&DYNP SETC '&PREFIX'
.TMF AIF ('&MF(1)' EQ 'G').GEN
AIF ('&MF' NE 'AUTO').TMFE
NAME DYNSPACE GET NAME FOR SPACE
LA R1,&DYNP LOAD ADDRESS OF PARAM LIST
USING &DYNP.DS,R1 USE GENERATED DSECT
&T SETC 'A'
&PAR SETC '&DYNP+4'
&DSECT SETB 1
AGO .START
.TMFE AIF ('&MF(2)' NE '').E2OK
MNOTE 4,'PLIST ADDRESS OMITTED, MF=G USED'
AGO .GEN
.E2OK ANOP
&DSECT SETB 1
AIF ('&MF(2)' EQ '(').RMFE
LA R1,&MF(2) LOAD PARAM LIST ADDRESS
USING &DYNP.DS,R1 USE GENERATED DSECT
AGO .START
.RMFE AIF ('&MF(2)' EQ '(R1)' OR '&MF(2)' EQ '(1)').START
LR R1,&PAR LOAD S99 PARAM LIST ADDRESS
AGO .START
.GEN LA R1,&DYNP.RBP LOAD ADDRESS OF S99 RBP
.START LA R15,&DYNP.RB LOAD ADDRESS OF S99 RB
USING S99RB,R15
ST R15,0(R1) AND STORE IN RB POINTER
XC 4(&DYNP.LEN-4,R1),4(R1) ZERO PARAMETER LIST
MVI S99RBLN,20 MOVE IN LIST LENGTH
MVI S99VERB,S99VRBAL MOVE IN VERB CODE
LA R14,&DYNP.TUP LOAD ADDRESS OF TU POINTERS
ST R14,S99TXTPP STORE ADDRESS IN S99 RB
LA R15,&DYNP.TU POINT TO SPACE FOR TEXT UNITS
USING S99TUNIT,R15
&DTUO SETA 0
&DTUPO SETA 0
.TDSN AIF ('&SSREQ' EQ 'YES').SSREQ
AIF ('&DSN&DA' NE '').DSN
AIF ('&DSNPDE' NE '').DSNPDE
AIF ('&DSNRET' NE '').DSNRT
AIF ('&SYSOUT' NE '').SYSOUT
AIF ('&DUMMY' NE '').DUMMY
AIF ('&QNAME' NE '').QNAME
.TDDN AIF ('&DDN&FILE&F' NE '').DDN
AIF ('&DDNRET&DDNTO' NE '').DDNRT
.TUNIT AIF ('&UNIT&VOL' NE '').UNIT
.TVOLRET AIF ('&VOLRET' NE '').VOLRET
.TDSRGO AIF ('&DSORGRT' NE '').DSORGRT
.TLABEL AIF ('&LABEL' NE '').LABEL
.TPSWD AIF ('&PASSWORD' NE '').PASSWORD
.TFORUSE AIF ('&FORUSER' NE '').FORUSER

```

```

.TTU      AIF      ('&TU' NE  '').TU
.TDISP    AIF      ('&DISP' NE  '').DISP
.TSPACE   AIF      ('&SPACE' NE  '').SPACE
.TLRECL   AIF      ('&LRECL' NE  '').DCB
          AIF      ('&DEN' NE  '').DCB
          AIF      ('&RECFM' NE  '').DCB
          AIF      ('&BLKSIZE' NE  '').DCB
          AIF      ('&DSORG' NE  '').DCB
          AIF      ('&KEYLEN' NE  '').DCB
.TDCBDSN  AIF      ('&DCBDSN' NE  '').DCBDSN
.TDCBDDN  AIF      ('&DCBDDN' NE  '').DCBDDN
.TFREE    AIF      ('&FREE' EQ 'CLOSE').FREE          TE7343
.TPERM    AIF      ('&PERM' EQ 'PERM' OR '&PERM' EQ 'PERMANENT').PERM
          AIF      ('&DUMMY' EQ 'PERM' OR '&DUMMY' EQ 'PERMANENT').PERM
.TSPECI   AIF      ('&SPECIAL' NE  '').SPECIAL
          AGO      .SVC99
.SSREQ    RCPSSREQ
          AGO      .TDSN
.DSN      RCPDSN &DSN&DA,&MEMBER
          AGO      .TDDN
.DSNPDE   RCPDSNPD &DSNPDE
          AGO      .TDDN
.DSNRT    RCPDSNRT &DSNRET
&DSNRTO   SETA    &DTUO-46
          AGO      .TDDN
.SYSOUT   RCPSYSOU &SYSOUT,COPIES=&COPIES,FREE=&FREE,DEST=&DEST,      X
          FORMS=&FORMS
          AGO      .TDDN
.DUMMY    RCPDUMMY &DUMMY
          AGO      .TDDN
.QNAME    RCPQNAME &QNAME
          AGO      .TDDN
.DDN      RCPDDN &DDN&F&FILE
          AGO      .TUNIT
.DDNRT    RCPDDNRT &DDNRET
&DDNRTO   SETA    &DTUO-10
          AGO      .TUNIT
.UNIT     RCPUNIT &UNIT,&VOL
          AGO      .TVOLRET
.VOLRET   RCPVOLRT &VOLRET
&VOLRTO   SETA    &DTUO-8
          AGO      .TDSRGO
.DSORGRT  RCPDSRGR
&DSRGRTO  SETA    &DTUO-2
          AGO      .TLABEL
.LABEL    RCPLABEL &LABEL
          AGO      .TPSWD
.PASSWORD RCPPSWD &PASSWORD
          AGO      .TFORUSE
.FORUSER  RCPFORUS &FORUSER
          AGO      .TTU
.TU       RCPTU &TU

```

```

      .DISP      AGO      .TDISP
                RCPDISP &DISP
                AGO      .TSPACE
      .SPACE    RCPSPACE &SPACE
                AGO      .TLRECL
      .DCB      RCPDDCB  LRECL=&LRECL,DEN=&DEN,RECFM=&RECFM,BLKSIZE=&BLKSIZE,  X
                DSORG=&DSORG,KEYLEN=&KEYLEN
                AGO      .TDCBDSN
      .DCBDSN   RCPDCBDS &DCBDSN
                AGO      .TDCBDDN
      .DCBDDN   RCPDCBDD &DCBDDN
                AGO      .TFREE
                TE7343
      .FREE     RCPFREE  &FREE
                TE7343
                AGO      .TPERM
      .PERM     RCPPERM
                AGO      .TSPECI
      .SPECIAL  RCPSPEC  &SPECIAL
      .SVC99    ANOP
&DTUPO        SETA  &DTUPO-4
                SPACE
                MVI  &DYNP.TUP+&DTUPO,X'80'  SET HIGH ORDER BIT ON TEXT PTRS
                MVI  &DYNP.RBP,X'80'        SET HIGH ORDER BIT ON RB PTR
                RCPSR2 UNSAVE
&DTUPO        SETA  &DTUPO+4
      .DYNA     AIF  (NOT &DSECT).DYNA
                DROP R1,R15                  DEACTIVATE ADDRESSABILITY
                LA   R14,4(R1)              POINT TO REQUEST BLOCK
                DYNALLOC
                AIF  (NOT &DSECT).LTR
                USING &DYNP.RB,R14          SET UP ADDRESSABILITY
**            NOTE  R14 HAS RB ADDRESS, R15 HAS SVC 99 RETURN CODE      **
      .LTR      AIF  ('&ERROR' EQ '').TDDTO
                LTR  R15,R15                TEST RETURN CODE
                BNZ  &ERROR                  BRANCH IF NON ZERO
      .TDDTO    AIF  ('&DDNTO' EQ '').RESERVE
&I            SETA  0
      .DDNTOL   ANOP
&I            SETA  &I+1
                AIF  ('&DDNTO(&I)' EQ '').RESERVE
                AIF  ('&DDNTO(&I)'(1,1) EQ ' ').DDNTOR
                MVC  &DDNTO(&I).(8),&DYNP.TU+&DDNRTO+2
                AGO  .DDNTOL
      .DDNTOR   ANOP
&C            SETC  '&DDNTO(&I)'(2,K'&DDNTO(&I)-2)
                MVC  0(8,&C),&DYNP.TU+&DDNRTO+2
                AGO  .DDNTOL
      .RESERVE  AIF  (&DSECT).RESDS
                SPACE 1
*****
**            RESERVE SPACE FOR DYNALLOC PARAMETER LIST      **
*****
                RCPDS

```

```

.SSP      ANOP
&DYNP.RBP DS    F          SVC 99 REQ BLOCK POINTER
&DYNP.RB  DS    5F        SVC 99 REQUEST BLOCK
&DYNP.TUP DS    CL&DTUPO  SPACE FOR TEXT POINTERS
          AIF    (&DTUO EQ 0).DTU21
&DYNP.TU  DS    CL&DTUO  SPACE FOR TEXT UNITS
          AIF    (&DSNRTO EQ 0).TDDNRTO
&DSNRTO  EQU    &DYNP.TU+&DSNRTO  OFFSET TO RETURNED DSN
.TDDNRTO AIF    ('&DDNRET' EQ '').DTU11
&DDNRET  EQU    &DYNP.TU+&DDNRTO  OFFSET TO RETURNED DDNAME
.DTU11   AIF    (&VOLRTO EQ 0).DTU12
&VOLRET  EQU    &DYNP.TU+&VOLRTO  OFFSET TO RETURNED VOLSER
.DTU12   AIF    (&DSRGRTO EQ 0).DTU10
&DSORGRT EQU    &DYNP.TU+&DSRGRTO  OFFSET TO RETURNED DSORG
          AGO    .DTU10
.DTU21   ANOP
&DYNP.TU  DS    0C        NO SPACE NEEDED FOR TEXT UNITS
.DTU10   ANOP
&DYNP.LEN EQU    *-&DYNP.RBP      LENGTH OF SPACE USED
          AIF    (&DSECT).DSP
          RCPDS
          SPACE 3
          AGO    .EXIT
.RESDS   ANOP
          AIF    ('&DYNSP' EQ '').SP3
          DYNSPACE ADD
.SP3    SPACE
&DYNP.DS DSECT          DSECT TO MAP SVC 99 DATA
          AGO    .SSP
.DSP    AIF    ('&MF(3)' EQ '').END1
&MF(3) EQU    &DYNP.LEN      LENGTH OF AREA
.END1   ANOP
&SYSECT CSECT
          SPACE 3
.EXIT   MEND

```

IEB817I MEMBER NAME (ALLOC) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=CLEAR
        MACRO
&NAME  CLEAR  &FIELD,&CHAR,&LENGTH
        LCLC  &FILL,&L
&L     SETC  'L''
&FILL  SETC  '&CHAR'
        AIF   ('&CHAR' NE '').CHSPEC
&FILL  SETC  '40'
.CHSPEC ANOP
&NAME  MVI   &FIELD,X'&FILL'   SET THE FIRST POSITION
        AIF   ('&LENGTH' EQ '').NOLSPEC
        MVC   &FIELD+1(&LENGTH),&FIELD  FILL THE ENTIRE FIELD
        MEXIT
.NOLSPEC ANOP
        MVC   &FIELD+1(&L&FIELD-1),&FIELD  FILL THE ENTIRE FIELD
        MEND
```

IEB817I MEMBER NAME (CLEAR) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD  NAME=CONV
        MACRO
&LABEL CONV  &TO,&FROM,&LEN,&EDMASK,&SCOMP
        LCLC &L,&FIRSTFR,&EDM,&COMP
        LCLA &COUNT
&L      SETC  'L''
&LABEL AIF   ('&LABEL' EQ '').NOLABEL  SKIP LABEL IF NOT PRESENT
        DS    0H          SET THE LABEL
.NOLABEL ANOP
&EDM    SETC  'EDMASK'    DEFAULT EDIT MASK
        AIF   ('&EDMASK' EQ '').DEFMASK  IF NOT ENTERED USE DEFAULT
&EDM    SETC  '&EDMASK'   USE THE ENTERED VALUE
.DEFMASK ANOP
&COMP   SETC  'BLANKS'    DEFAULT COMPARISON CHARS
        AIF   ('&SCOMP' EQ '').DEFCOMP  NOT ENTERED, USE THE DEFAULT
&COMP   SETC  '&SCOMP'    GET WHAT THE GUY WANTS
.DEFCOMP ANOP
&FIRSTFR SETC  '&FROM'(1,1)  GET FIRST CHAR OF &FROM
        AIF   ('&FIRSTFR' EQ '(').REGISTR
        L     R1,&FROM      GET THE DATA TO CONVERT
        CVD  R1,DOUBLE     CONVERT TO PACKED DECIMAL
        AGO   .INDEC
.REGISTR ANOP
&COUNT SETA  K'&FROM-2
&FIRSTFR SETC  '&FROM'(2,&COUNT)  STRIP THE PERRONS
        CVD  &FIRSTFR,DOUBLE  CONVERT TO PACKED DECIMAL
.INDEC  ANOP
        MVC  CHARS,&EDM     PUT IN THE EDIT MASK
        ED   CHARS,DOUBLE   CONVERT TO CHARACTERS
        AIF   ('&LEN' NE '').LENSET
        MVC  &TO,CHARS+16-&L&TO  MOVE IN THE NUMBER
        CLC  CHARS(16-&L&TO),&COMP  WAS THERE AN OVERFLOW?
        BE   *+10          NO, EVERYTHING WAS OK
        MVC  &TO,STARS      BAD NEWS, NOTE IT
        MEXIT
.LENSET ANOP
        MVC  &TO.(&LEN),CHARS+16-&LEN  MOVE IN THE NUMBER
        CLC  CHARS(16-&LEN),&COMP  WAS THERE AN OVERFLOW?
        BE   *+10          NO, EVERYTHING WAS OK
        MVC  &TO.(&LEN),STARS  BAD NEWS, NOTE IT
        MEND

```

IEB817I MEMBER NAME (CONV) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=CONVY2K
        MACRO
&LABEL CONVY2K &TO,&FROM,&LEN,&EDMASK,&SCOMP
        LCLC  &L,&FIRSTFR,&EDM,&COMP
        LCLA  &COUNT
&L      SETC  'L''
&LABEL AIF    ('&LABEL' EQ '').NOLABEL  SKIP LABEL IF NOT PRESENT
&LABEL DS    0H          SET THE LABEL
.NOLABEL ANOP
&EDM    SETC  'EDMASK'    DEFAULT EDIT MASK
&EDM    AIF    ('&EDMASK' EQ '').DEFMASK  IF NOT ENTERED USE DEFAULT
&EDM    SETC  '&EDMASK'   USE THE ENTERED VALUE
.DEFMASK ANOP
&COMP   SETC  'BLANKS'    DEFAULT COMPARISON CHARS
&COMP   AIF    ('&SCOMP' EQ '').DEFCOMP  NOT ENTERED, USE THE DEFAULT
&COMP   SETC  '&SCOMP'    GET WHAT THE GUY WANTS
.DEFCOMP ANOP
&FIRSTFR SETC  '&FROM'(1,1)  GET FIRST CHAR OF &FROM
&FIRSTFR AIF    ('&FIRSTFR' EQ ' ').REGISTR
&FIRSTFR L     R1,&FROM      GET THE DATA TO CONVERT
&FIRSTFR CVD   R1,DOUBLE    CONVERT TO PACKED DECIMAL
&FIRSTFR AGO   .INDEC
.REGISTR ANOP
&COUNT SETA  K'&FROM-2
&FIRSTFR SETC  '&FROM'(2,&COUNT)  STRIP THE PERRONS
&FIRSTFR CVD   &FIRSTFR,DOUBLE  CONVERT TO PACKED DECIMAL
.INDEC  ANOP
&FIRSTFR MVC   CHARS,&EDM     PUT IN THE EDIT MASK
&FIRSTFR ED    CHARS,DOUBLE   CONVERT TO CHARACTERS
&FIRSTFR AIF    ('&LEN' NE '').LENSET
&FIRSTFR MVC   &TO,CHARS+16-&L&TO  MOVE IN THE NUMBER
&FIRSTFR CLC   CHARS(16-&L&TO),&COMP  WAS THERE AN OVERFLOW?
&FIRSTFR BE    *+10          NO, EVERYTHING WAS OK
&FIRSTFR MVC   &TO,STARS      BAD NEWS, NOTE IT
&FIRSTFR MEXIT
.LENSET ANOP
&FIRSTFR MVC   &TO.(&LEN),CHARS+16-&LEN  MOVE IN THE NUMBER
&FIRSTFR CLC   CHARS(16-&LEN),&COMP  WAS THERE AN OVERFLOW?
*      BE    *+10          NO, EVERYTHING WAS OK
&FIRSTFR B     *+10          ALLOW YY > 99 FOR YEAR 2000    Y2K DEC97
&FIRSTFR MVC   &TO.(&LEN),STARS  BAD NEWS, NOTE IT
&FIRSTFR MEND

```

IEB817I MEMBER NAME (CONVY2K) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```

./      ADD    NAME=DYNSPACE
        MACRO
&NAME  DYNSPACE &TYPE
.*
.*      THIS IS AN INNER MACRO TO ALLOC/FREE.
.*      IT IS CALLED TO   A) NAME AN AREA FOR THE PARMLIST
.*                       B) LOG THE VARIOUS AMOUNTS NEEDED BY
.*                       EACH, REMEMBERING THE LARGEST.
.*                       C) GENERATING A DS FOR THE LARGEST AMOUNT.
.*      THE FIRST TWO FUNCTIONS ARE INVOKED BY ALLOC/FREE MACROS ONLY,
.*      AND THE THIRD IS USED BY THE PROGRAMMER, EITHER EXPLICITLY,
.*      OR BY BEGINWKA, IF THE LATTER IS USED.
.*
.*      TO INVOKE THE NAMING FUNCTION, ALLOC/FREE GENERATE
.*      NAME DYNSPACE
.*      NOTE. THE NAMING OPERATION ONLY GENERATES A NAME ON THE
.*      FIRST CALL IN THE ASSEMBLY. THE NAME REMAINS THE SAME UNTIL
.*      DYNSPACE IS CALLED TO EXPAND INTO A DS.
.*
.*      THE SECOND FUNCTION IS INVOKED BY THE MACRO CALL
.*      DYNSPACE ADD
.*      (NO NAME FIELD AND ONE OPERAND)
.*      IT USES THE GLOBAL VARIABLES &DTUO AND &DTUPO TO CALCULATE
.*      THE SPACE FOR THIS REQUEST, AND UPDATES &DYNSPQ ONLY IF THE
.*      CURRENT REQUEST IS FOR A GREATER AMOUNT
.*
.*      THE THIRD FUNCTION IS INVOKED BY CALLING DYNSPACE WITH NO
.*      NAME OR OPERAND FIELD.
.*      THIS EXPANDS INTO A DEFINE STORAGE, CLEARS THE DYNSPACE NAME
.*      GLOBAL SETC, AND THE &DYNSPQ GLOBAL SETA.
.*      THUS, THE MACRO IS SERIALLY REUSABLE IN ALL FUNCTIONS.
.*
        GBLA  &DYNSPQ,&DTUO,&DTUPO,&RCPDYN
        GBLC  &DYNSP,&DYNSP
        LCLA  &I
        AIF   ('&NAME' NE '').NAME
        AIF   ('&TYPE' EQ '').ALLOC
.*      THE ACCUMULATE FUNCTION IS REQUIRED
&I      SETA  24+&DTUO+&DTUPO          GET AMOUNT FOR THIS REQUEST
        AIF   (&I LE &DYNSPQ).EXIT    IF CURRENT < MAX, EXIT
&DYNSPQ SETA  &I                      ELSE UPDATE CURRENT MAXIMUM
        MEXIT
.NAME   AIF   ('&DYNSP' NE '').EXIT    IF NAME ALREADY EXISTS, EXIT
&DYNSP SETC  'DYNSP&RCPDYN'          ELSE GENERATE A NAME
.EXIT  MEXIT
.ALLOC  AIF   ('&DYNSP' EQ '').EXIT
*
**      RESERVE SPACE FOR ALLOC/FREE MACRO WORK AREA
*
&DYNSP DS    0F,CL&DYNSPQ            RESERVE SPACE
&DYNSP SETC  ''                      SET MAX QUANTITY TO 0
&DYNSPQ SETA  0

```

MEND

IEB817I MEMBER NAME (DYNSPACE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD  NAME=ENTER
        MACRO
&SUBR  ENTER &BASES,&SAVE,&CSECT
.*     THIS MACRO, USED WITH THE LEAVE MACRO, WILL PERFORM
.*     STANDARD HOUSEKEEPING FOR A CSECT, INCLUDING SAVEAREA
.*     CONSTRUCTION AND CHAINING, AND GETTING SOME STORAGE,
.*     IF THAT IS DESIRED.
.*     THE LEAVE MACRO WILL FREE THE GOTTEN STORAGE
.*     THE OPERANDS ARE
.*     &SUBR  ENTER  &BASES,&SAVE,&CSECT
.*     WHERE
.*     &SUBR  IS THE NAME OF THE CSECT
.*     &BASES ARE THE BASE REGISTERS FOR THE ROUTINE
.*     &SAVE  IS THE LABEL FOR A SAVEAREA, OR A SUBPOOL
.*           AND LENGTH FOR THE GETMAIN
.*     &CSECT TO CONTINUE AN EXISTING CSECT WITH ENTRY
.*           POINT &SUBR
.*
.*     EXAMPLES -
.*           ENTER 13,*
.*
.*     THIS WILL GENERATE NON-REENTRANT CODE, USING SAVEAREA
.*     AS THE SAVE AREA LABEL, AND REGISTER 13 FOR THE BASE
.*     REGISTER.
.*
.*     RENTMOD ENTER (12,11),(&LDSECT)
.*
.*     THIS WILL GENERATE REENTRANT CODE WITH REGISTERS 12 AND
.*     11 FOR BASE REGISTERS.  A GETMAIN WILL BE DONE FOR THE
.*     DEFAULT SUBPOOL (0) WITH A LENGTH 'LDSECT'.
.*
        GBLC  &LV,&SP
        LCLA  &K,&N
        LCLC  &AREA,&B(16),&SUBNAME,&S
&SUBNAME SETC  '&SUBR'
        AIF  ('&SUBNAME' NE '').SUBSPEC
&SUBNAME SETC  'MAIN'          DEFAULT CSECT NAME
.SUBSPEC AIF  ('&CSECT' EQ '').NOTENT IS IT AN ENTRY POINT?
&CSECT   CSECT
&SUBNAME DS    0F
        AGO  .CSSPEC
.NOTENT  ANOP
&SUBNAME CSECT
.CSSPEC  ANOP
        SAVE (14,12),T,&SUBNAME  SAVE THE REGISTERS
        AIF  ('&BASES(1)' EQ '15' OR '&BASES' EQ '').R15SET
        AIF  ('&BASES(1)' EQ '13' AND '&SAVE' NE '').R15SET
        LR   &BASES(1),15  SET FIRST BASE REG
.R15SET  CNOP  0,4
&S      SETC  '&SUBNAME'
        AIF  (N'&SAVE EQ 2).P4  SUBPOOL, SIZE SPEC?
        AIF  ('&SAVE' EQ '').P3  NO SAVEAREA - DEFAULT

```

```

&AREA   SETC   '&SAVE'
        AIF   ('&SAVE' NE '*').P2
&AREA   SETC   'SAVEAREA'
.P2     AIF   ('&BASES(1)' NE '13').P4
&S      SETC   '*'
        USING &SUBNAME,15
        ST   14,&AREA+4
        LA   14,&AREA
        ST   14,8(13)
        L    14,&AREA+4
        ST   13,&AREA+4
        BAL  13,*+76          SKIP AROUND THE SAVEAREA
        DROP 15
        AGO  .P4
.P3     AIF   ('&BASES(1)' NE '13').P4
        MNOTE 8,'*** CONTENTS OF REG 13 ARE LOST. NO SAVE AREA WAS ESTABLISHED.'
.P4     AIF   ('&BASES(1)' NE '14' OR '&SAVE' EQ '').P5
        MNOTE 8,'*** MACRO RESTRICTION - REG 14 MUST NOT BE USED AS THE FIRST BASE REGISTER IF A SAVE AREA IS USED.'
.P5     AIF   ('&BASES' EQ '').P9
&N      SETA  N'&BASES
.P6     ANOP
&K      SETA  &K+1
&B(&K) SETC   ',','&BASES(&K)'
        AIF   (N'&SAVE EQ 1).PE
        AIF   ('&BASES(&K)' NE '13').P7
        MNOTE 8,'*** REG 13 MAY NOT BE USED AS A BASE REGISTER FOR REENTRANT CODE.'
        AGO  .P7
.PE     AIF   ('&BASES(&K+1)' NE '13' OR '&SAVE' EQ '').P7
        MNOTE 8,'*** WHEN USING A SAVE AREA, REG 13 MAY NOT BE USED AS A SECONDARY BASE REGISTER.'
.P7     AIF   ('&BASES(&K+1)' NE '').P6
        USING &S&B(1)&B(2)&B(3)&B(4)&B(5)&B(6)&B(7)&B(8)&B(9)&B(10)&B(11)&B(12)&B(13)&B(14)&B(15)&B(16)
&K      SETA  1
        AIF   ('&BASES(1)' NE '13' OR '&SAVE' EQ '').P8
&AREA   DC    18F'0'
.P8     AIF   (&K GE &N).P10
        LA   &BASES(&K+1),X'FFF'(&BASES(&K))
        LA   &BASES(&K+1),1(&BASES(&K+1))
&K      SETA  &K+1
        AGO  .P8
.P9     USING &SUBNAME,15
.P10    AIF   (N'&SAVE GE 2).P13
        AIF   ('&SAVE' EQ '' OR '&BASES(1)' EQ '13').P12
        AIF   ('&SAVE' GE '0').P16  NUMERIC MEANS A PASSED AREA
        ST   14,&AREA+4
        LA   14,&AREA
        ST   14,8(13)
        L    14,&AREA+4

```

```

.P11      ST      13,&AREA+4
&AREA    BAL     13,*+76      SKIP AROUND THE SAVEAREA
          DC      18F'0'
.P12      MEXIT
.P13      ANOP
&LV      SETC    '&SAVE(2) '
&SP      SETC    '0 '
          AIF     ('&SAVE(1) ' EQ ' ').P14
&SP      SETC    '&SAVE(1) '
.P14      CNOP    0,4          DO A GETMAIN FOR THE AREA
          BAL     1,*+8        POINT THE SP AND LV
ENT&SYSNDX DC AL1(&SP)        SUBPOOL FOR THE GETMAIN
          DC      AL3(&LV)      LENGTH OF THE GETMAIN
          L       0,0(1)        GET THE DATA IN REG 1
          SVC     10           ISSUE THE GETMAIN
.*
          ST      13,4(1)       PRIOR SAVEAREA ADDRESS TO MINE
          ST      1,8(13)       MY SAVEAREA ADDRESS TO HIS
          LR      2,13          KEEP THE SAVEAREA ADDRESS FOR REGS
          LR      13,1          THIS IS MY SAVEAREA
          LM      0,2,20(2)     RESTORE ORIGINAL REGS
          MEXIT
.P16      L       1,&AREA+0(1)   NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
          ST      13,4(1)       PRIOR SAVEAREA ADDRESS TO MINE
          ST      1,8(13)       MY SAVEAREA ADDRESS TO HIS
          LR      2,13          KEEP THE SAVEAREA ADDRESS FOR REGS
          LR      13,1          THIS IS MY SAVEAREA
          LM      0,2,20(2)     RESTORE ORIGINAL REGS
          MEND

```

IEB817I MEMBER NAME (ENTER) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD  NAME=ENTERX
        MACRO
&SUBR  ENTERX &BASES,&SAVE,&CSECT
.*     THIS MACRO, USED WITH THE LEAVE MACRO, WILL PERFORM
.*     STANDARD HOUSEKEEPING FOR A CSECT, INCLUDING SAVEAREA
.*     CONSTRUCTION AND CHAINING, AND GETTING SOME STORAGE,
.*     IF THAT IS DESIRED.
.*     THE LEAVE MACRO WILL FREE THE GOTTEN STORAGE
.*     THE OPERANDS ARE
.*     &SUBR  ENTER  &BASES,&SAVE,&CSECT
.*     WHERE
.*     &SUBR  IS THE NAME OF THE CSECT
.*     &BASES ARE THE BASE REGISTERS FOR THE ROUTINE
.*     &SAVE  IS THE LABEL FOR A SAVEAREA, OR A SUBPOOL
.*           AND LENGTH FOR THE GETMAIN
.*     &CSECT TO CONTINUE AN EXISTING CSECT WITH ENTRY
.*           POINT &SUBR
.*
.*     EXAMPLES -
.*           ENTER 13,*
.*
.*     THIS WILL GENERATE NON-REENTRANT CODE, USING SAVEAREA
.*     AS THE SAVE AREA LABEL, AND REGISTER 13 FOR THE BASE
.*     REGISTER.
.*
.*     RENTMOD ENTER (12,11),(&LDSECT)
.*
.*     THIS WILL GENERATE REENTRANT CODE WITH REGISTERS 12 AND
.*     11 FOR BASE REGISTERS.  A GETMAIN WILL BE DONE FOR THE
.*     DEFAULT SUBPOOL (0) WITH A LENGTH 'LDSECT'.
.*
        GBLC  &LV,&SP
        LCLA  &K,&N
        LCLC  &AREA,&B(16),&SUBNAME,&S
&SUBNAME SETC  '&SUBR'
        AIF  ('&SUBNAME' NE '').SUBSPEC
&SUBNAME SETC  'MAIN'          DEFAULT CSECT NAME
.SUBSPEC AIF  ('&CSECT' EQ '').NOTENT IS IT AN ENTRY POINT?
&CSECT  CSECT
&SUBNAME DS    0F
        AGO  .CSSPEC
.NOTENT ANOP
&SUBNAME CSECT
.CSSPEC ANOP
        SAVE (14,12),T,&SUBNAME  SAVE THE REGISTERS
        AIF  ('&BASES(1)' EQ '15' OR '&BASES' EQ '').R15SET
        AIF  ('&BASES(1)' EQ '13' AND '&SAVE' NE '').R15SET
        LR   &BASES(1),15  SET FIRST BASE REG
.R15SET CNOP  0,4
&S      SETC  '&SUBNAME'
        AIF  (N'&SAVE EQ 2).P4  SUBPOOL, SIZE SPEC?
        AIF  ('&SAVE' EQ '').P3 NO SAVEAREA - DEFAULT

```

```

&AREA   SETC   '&SAVE'
        AIF   ('&SAVE' NE '*').P2
&AREA   SETC   'SAVEAREA'
.P2     AIF   ('&BASES(1)' NE '13').P4
&S      SETC   '*'
        USING &SUBNAME,15
        ST   14,&AREA+4
        LA   14,&AREA
        ST   14,8(13)
        L    14,&AREA+4
        ST   13,&AREA+4
        BAL  13,*+76          SKIP AROUND THE SAVEAREA
        DROP 15
        AGO  .P4
.P3     AIF   ('&BASES(1)' NE '13').P4
        MNOTE 8,'*** CONTENTS OF REG 13 ARE LOST.  NO SAVE AREA WAS ESTABLISHED.'
.P4     AIF   ('&BASES(1)' NE '14' OR '&SAVE' EQ '').P5
        MNOTE 8,'*** MACRO RESTRICTION - REG 14 MUST NOT BE USED AS THE FIRST BASE REGISTER IF A SAVE AREA IS USED.'
.P5     AIF   ('&BASES' EQ '').P9
&N      SETA  N'&BASES
.P6     ANOP
&K      SETA  &K+1
&B(&K)  SETC   ',','&BASES(&K)'
        AIF   (N'&SAVE EQ 1).PE
        AIF   ('&BASES(&K)' NE '13').P7
        MNOTE 8,'*** REG 13 MAY NOT BE USED AS A BASE REGISTER FOR REENTRANT CODE.'
        AGO  .P7
.PE     AIF   ('&BASES(&K+1)' NE '13' OR '&SAVE' EQ '').P7
        MNOTE 8,'*** WHEN USING A SAVE AREA, REG 13 MAY NOT BE USED AS A SECONDARY BASE REGISTER.'
.P7     AIF   ('&BASES(&K+1)' NE '').P6
        USING &S&B(1)&B(2)&B(3)&B(4)&B(5)&B(6)&B(7)&B(8)&B(9)&B(10)&B(11)&B(12)&B(13)&B(14)&B(15)&B(16)
&K      SETA  1
        AIF   ('&BASES(1)' NE '13' OR '&SAVE' EQ '').P8
&AREA   DC    18F'0'
.P8     AIF   (&K GE &N).P10
        LA   &BASES(&K+1),X'FFF'(&BASES(&K))
        LA   &BASES(&K+1),1(&BASES(&K+1))
&K      SETA  &K+1
        AGO  .P8
.P9     USING &SUBNAME,15
.P10    AIF   (N'&SAVE GE 2).P13
        AIF   ('&SAVE' EQ '' OR '&BASES(1)' EQ '13').P12
        AIF   ('&SAVE(1,1)' GE '0').P16  NUMERIC MEANS A PASSED AREA
        ST   14,&AREA+4
        LA   14,&AREA
        ST   14,8(13)
        L    14,&AREA+4

```

```

      ST      13,&AREA+4
.P11  BAL     13,*+76      SKIP AROUND THE SAVEAREA
&AREA DC      18F'0'
.P12  MEXIT
.P13  ANOP
&LV  SETC    '&SAVE(2) '
&SP  SETC    '0 '
&SP  AIF     ('&SAVE(1) ' EQ ' ').P14
&SP  SETC    '&SAVE(1) '
.P14  CNOP    0,4          DO A GETMAIN FOR THE AREA
      BAL     1,*+8        POINT THE SP AND LV
ENT&SYSNDX DC AL1(&SP)     SUBPOOL FOR THE GETMAIN
      DC      AL3(&LV)     LENGTH OF THE GETMAIN
      L       0,0(1)      GET THE DATA IN REG 1
      SVC     10          ISSUE THE GETMAIN
      *              CHAIN THE SAVEAREAS
      ST      13,4(1)     PRIOR SAVEAREA ADDRESS TO MINE
      ST      1,8(13)    MY SAVEAREA ADDRESS TO HIS
      LR      2,13       KEEP THE SAVEAREA ADDRESS FOR REGS
      LR      13,1       THIS IS MY SAVEAREA
      LA      4,12(13)   YES, POINT PAST THE CHAIN
      L       5,ENT&SYSNDX GET THE SIZE
      LA      6,12       MINUS THE CHAIN AREA (12 BYTES )
      SR      5,6        GIVES THE AMOUNT TO CLEAR
      SR      7,7        CLEAR THE FROM COUNT AND CLEAR BYTE
      MVCL    4,6        WHEE, CLEAR IT OUT
      LM      0,7,20(2)  RESTORE THE ORIGINAL REGISTERS
      MEXIT
.P16  L       1,&AREA+0(1) NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
      ST      13,4(1)    PRIOR SAVEAREA ADDRESS TO MINE
      ST      1,8(13)    MY SAVEAREA ADDRESS TO HIS
      LR      2,13       KEEP THE SAVEAREA ADDRESS FOR REGS
      LR      13,1       THIS IS MY SAVEAREA
      LM      0,2,20(2)  RESTORE ORIGINAL REGS
      MEND

```

IEB817I MEMBER NAME (ENTERX) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```

./      ADD      NAME=FREE
        MACRO
&NAME  FREE      &UNALC ,&DSN= ,&DDN= ,&MEMBER= ,&DISP= ,&SYSOUT= ,          X
        &ERROR= ,&MF=AUTO ,&PREFIX= ,&FILE= ,&F= ,&DA= ,&HOLD=
        GBLA     &RCPDYN          COUNTER FOR NO ENTRIES TO MACRO
        GBLA     &DTUO            OFFSET TO TEXT UNITS
        GBLA     &DTUPO          OFFSET TO TEXT UNIT POINTERS
        GBLB     &RCPS99(2)      TELL RCPDSECT NEED DSECTS
        GBLC     &DYNP           PREFIX FOR LABELS FOR THIS CALL
        GBLC     &DYNSP          NAME FOR AUTOMATIC STORAGE ALLOC
        LCLB     &DSECT          DSECT NEEDED FOR STORAGE, MF=E
        LCLC     &C,&T,&PAR
&RCPS99(1) SETB          1
&RCPDYN SETA     &RCPDYN+1      INCEREMENT COUNTER
&DYNP   SETC     'DYN&RCPDYN'  SET DEFAULT PREFIX
&NAME   DS       0H
        AIF      ('&PREFIX' EQ '').TMF
        AIF      (K'&PREFIX LT 4).POK
&DYNP   MNOTE   4,'PREFIX TOO LONG, 1ST 4 CHARS USED'
        SETC     '&PREFIX'(1,4)
        AGO     .TMF
        .POK
&DYNP   SETC     '&PREFIX'
        .TMF    AIF      ('&MF(1)' EQ 'G').GEN
        AIF      ('&MF' NE 'AUTO').TMFE
NAME     DYNSPACE          GET NAME FOR SPACE
        LA      R1,&DYNSP          LOAD ADDRESS OF PARAM LIST
        USING   &DYNP.DS,R1      USE GENERATED DSECT
&T      SETC     'A'
&PAR    SETC     '&DYNSP+4'
&DSECT  SETB     1
        AGO     .START
        .TMFE   AIF      ('&MF(2)' NE '').E2OK
        MNOTE   4,'PLIST ADDRESS OMITTED, MF=G USED'
        AGO     .GEN
        .E2OK
&DSECT  SETB     1
        AIF      ('&MF(2)' EQ '(').RMFE
        LA      R1,&MF(2)          LOAD PARAM LIST ADDRESS
&T      SETC     'A'
&PAR    SETC     '&MF(2)+4'
        USING   &DYNP.DS,R1      USE GENERATED DSECT
        AGO     .START
        .RMFE   AIF      ('&MF(2)' EQ '(R1)' OR '&MF(2)' EQ '(1)').START
&PAR    SETC     '&MF(2)'(2,K'&MF(2)-2)
&T      SETC     'R'
&PAR    LR      R1,&PAR          LOAD S99 PARAM LIST ADDRESS
&PAR    SETC     '4&MF(2)'
        USING   &DYNP.DS,R1      USE GENERATED DSECT
        AGO     .START
        .GEN    LA      R1,&DYNP.RBP  LOAD ADDRESS OF S99 RBP
&T      SETC     'A'

```

```

&PAR      SETC   '&DYNP.RB'
.START    LA     R15,&DYNP.RB          LOAD ADDRESS OF S99 RB
          USING  S99RB,R15
          ST     R15,0(R1)            AND STORE IN RB POINTER
          XC     4(&DYNP.LEN-4,R1),4(R1) ZERO PARAMETER LIST
          MVI    S99RBLN,20           MOVE IN LIST LENGTH
          MVI    S99VERB,S99VRBUN     MOVE IN VERB CODE
          LA     R14,&DYNP.TUP        LOAD ADDRESS OF TU POINTERS
          ST     R14,S99TXTPP         STORE ADDRESS IN S99 RB
          LA     R15,&DYNP.TU         POINT TO SPACE FOR TEXT UNITS
          USING  S99TUNIT,R15
&DTUO     SETA   0
&DTUPO    SETA   0
          AIF    ('&DSN&DA' NE '').DSN
          AIF    ('&SYSOUT' NE '').SYSOUT
.TDDN     AIF    ('&DDN&FILE&F' NE '').DDN
.TDISP    AIF    ('&DISP' NE '').DISP
.TUNALC   AIF    ('&UNALC' NE '').PERM
.THOLD    AIF    ('&HOLD' NE '').HOLD
          AGO    .SVC99
.DSN      RCPFDSN &DSN&DA,&MEMBER
          AGO    .TDDN
.SYSOUT   RCPFSYS &SYSOUT
          AGO    .TDDN
.DDN      RCPFDDN &DDN&F&FILE
          AGO    .TDISP
.DISP     RCPFDISP &DISP
          AGO    .TUNALC
.PERM     RCPUNALC
          AGO    .THOLD
.HOLD     RCPFHOLD &HOLD
.SVC99    ANOP
&DTUPO    SETA   &DTUPO-4
          SPACE
          MVI    &DYNP.TUP+&DTUPO,X'80' SET HIGH ORDER BIT ON TEXT PTRS
          MVI    &DYNP.RBP,X'80'      SET HIGH ORDER BIT ON RB PTR
          RCPSR2 UNSAVE
&DTUPO    SETA   &DTUPO+4
          AIF    (NOT &DSECT).DYNA
          DROP  R1,R15                DEACTIVATE ADDRESSABILITY
.DYNA     DYNALLOC
          AIF    ('&ERROR' EQ '').RESERVE
          AIF    ('&PAR' EQ '').LTR
          L&T   R14,&PAR              LOAD REG 14 WITH ADDRESS OF RB
          AIF    (NOT &DSECT).LTR
          USING &DYNP.RB,R14         SET UP ADDRESSABILITY
.LTR      LTR   R15,R15              TEST RETURN CODE
          BNZ   &ERROR                BRANCH IF NON ZERO
**        NOTE.  R14 POINTS TO REQUEST BLOCK, R15 HAS RETURN CODE    **
.RESERVE  AIF    (&DSECT).RESDS
          SPACE

```

```
**          RESERVE SPACE FOR DYNALLOC DATA          **
```

```
*****
```

```
RCPDS
```

```
.SSP      ANOP
&DYNP.RBP DS    F          SVC 99 REQ BLOCK POINTER
&DYNP.RB  DS    5F        SVC 99 REQUEST BLOCK
&DYNP.TUP DS    CL&DTUPO   SPACE FOR TEXT POINTERS
          AIF    (&DTUO EQ 0).DTU11
&DYNP.TU  DS    CL&DTUO   SPACE FOR TEXT UNITS
          AGO    .DTU10
.DTU11    ANOP
&DYNP.TU  DS    0C        NO SPACE NEEDED FOR TEXT UNITS
.DTU10    ANOP
&DYNP.LEN EQU    *-&DYNP.RBP LENGTH OF SPACE USED
          AIF    (&DSECT).DSP
          RCPDS
          SPACE 3
          AGO    .EXIT
.RESDS    ANOP
          AIF    ('&DYNP' EQ '').SP3
          DYNSPACE ADD
.SP3      SPACE
&DYNP.DS  DSECT          DSECT TO MAP SVC 99 DATA
          AGO    .SSP
.DSP      AIF    ('&MF(3)' EQ '').END1
&MF(3)    EQU    &DYNP.LEN LENGTH OF AREA
.END1     ANOP
&SYSECT   CSECT
          SPACE 3
.EXIT     MEND
```

```
IEB817I MEMBER NAME (FREE ) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.
```

```
./      ADD  NAME=LEAVE
        MACRO
&NAME  LEAVE &EQ,&RC=
        GBLC &LV,&SP
&NAME  LR   2,13
        L   13,4(13)
        AIF ('&RC' EQ '').L0
        LA  15,&RC          LOAD THE RETURN CODE
.L0     STM 15,1,16(13)     STORE RETURN REGS
        AIF ('&LV' EQ '').L1  ANYTHING TO FREE?
        FREEMAIN R,LV=&LV,SP=&SP,A=(2)  FREE THE AREA
.L1     RETURN (14,12),T    RETURN FROM WHENCE WE CAME
        AIF ('&EQ' NE 'EQ').L4  REGISTERS TOO?
        COPY REGS
.L4     MEND
```

IEB817I MEMBER NAME (LEAVE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD  NAME=MSG  
        MACRO  
&NAME  MSG   &TEXT  
        LCLA  &A  
&A     SETA  K'&TEXT-2+4  SUBTRACT QUOTES, ADD PREFIX FOUR BYTES  
&NAME  DC    H'&A',H'0',C&TEXT  
        MEND
```

IEB817I MEMBER NAME (MSG) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=PDEDSNAM
        MACRO
        PDEDSNAM

*
*      DEFINE A DSECT FOR THE DSNAME PARSE DESCRIPTION
*
PDEDSNAM DSECT
PDEDSN   DS    A           POINTER TO DSNAME
PDEDSNL  DS    H           LENGTH OF DSNAME
PDEDFLG1 DS    X           DATA SET NAME FLAGS
PDEDFLD1 EQU  X'80'        ONE IF THE DSNAME IS PRESENT
PDEDFLQ1 EQU  X'40'        ONE IF THE DSNAME IS WITHIN QUOTES
PDEDMEM  DS    A           POINTER TO MEMBER NAME
PDEDMEML DS    H           LENGTH OF MEMBER NAME
PDEDFLG2 DS    X           MEMBER NAME FLAGS
PDEDFLD2 EQU  X'80'        ONE IF THE MEMBER IS PRESENT
PDEDPASS DS    A           POINTER TO PASSWORD
PDEDPASL DS    H           LENGTH OF PASSWORD
PDEDFLG3 DS    X           PASSWORD FLAGS
PDEDFLD3 EQU  X'80'        ONE IF THE PASSWORD IS PRESENT
PDEDCHAN DS    0F          CHAIN ADDRESS
PDEDCHNF DS    X           CHAIN FLAGS ( X'FF' FOR END )
PDEDCHN  DS    AL3         TRUE CHAIN POINTER
        MEND
```

```
IEB817I MEMBER NAME (PDEDSNAM) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.
```

```
./      ADD    NAME=RCPBFRGS
        MACRO
        RCPBFRGS &BUFPTR,&WKREGS
        GBLC   &RCPBFRP,&RCPBFR1,&RCPBFR2
&RCPBFRP AIF    ('&BUFPTR' EQ '').TGP
        SETC   '&BUFPTR'
        AGO    .TWK1
.TGP     AIF    ('&RCPBFRP' NE '').TWK1
&RCPBFRP SETC   'R1'
.TWK1    AIF    ('&WKREGS(1)' EQ '').TG1
&RCPBFR1 SETC   '&WKREGS(1)'
        AGO    .TWK2
.TG1     AIF    ('&RCPBFR1' NE '').TWK2
&RCPBFR1 SETC   'R14'
.TWK2    AIF    ('&WKREGS(2)' EQ '').TG2
&RCPBFR2 SETC   '&WKREGS(2)'
        MEXIT
.TG2     AIF    ('&RCPBFR2' NE '').EXIT
&RCPBFR2 SETC   'R15'
.EXIT    MEND
```

IEB817I MEMBER NAME (RCPBFRGS) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPBTU
        MACRO
        RCPBTU &KEY,&NUM,&PAR
        LCLA  &L
.*
.*  INNER MACRO FOR ALLOC, TO GENERATE TEXT UNITS ENTERED
.*  IN QUOTES
.*
&L      SETA   K'&PAR-2          GET LENGTH OF TEXT UNIT
        MVI    S99TUKEY+1,&KEY    SET TEXT UNIT KEY
        MVI    S99TUNUM+1,&NUM    SET NUMBER FIELD
        MVI    S99TULNG+1,&L      MOVE IN LENGTH
        MVC    S99TUPAR(&L.),=C&PAR  MOVE IN TEXT UNIT
&L      SETA   &L+6
        AIF    (&L/2 EQ (&L+1)/2).LOK
&L      SETA   &L+1
.LOK    RCPDINC &L
        MEND
```

IEB817I MEMBER NAME (RCPBTU) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD    NAME=RCPBTU2
        MACRO
        RCPBTU &KEY,&NUM,&PAR
        GBLA  &DTUPO
        GBLC  &DYNP
        LCLA  &L
.*
.*  INNER MACRO FOR ALLOC, TO BRANCH AROUND TEXT UNIT AND
.*  CREATE TEXT UNIT
.*
&L      SETA  K'&PAR+8          GET LENGTH TO BRANCH AROUND
        AIF  (&L/2 EQ (&L+1)/2).LOK MAKE SURE LENGTH IS EVEN
&L      SETA  &L+1
.&L     BAL  R14,*+&L          BRANCH AROUND TEXT UNIT
&L      SETA  K'&PAR-2
        DC   Y(&KEY,&NUM,&L),C&PAR TEXT UNIT
        LA   R14,0(R14)        CLEAR HIGH ORDER BYTE
        ST   R14,&DYNP.TUP+&DTUPO STORE TEXT UNIT ADDRESS
&DTUPO  SETA  &DTUPO+4
        MEND
```

IEB817I MEMBER NAME (RCPBTU2) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPCKID
        MACRO
&NAME  RCPCKID          &CHECKID
        GBLB  &RCPECT(2), &RCPPSCB(2)
        GBLC  &RCPPRE
        LCLC  &CHARVAR, &P
        LCLA  &COUNTR, &L
&P      SETC  '&RCPPRE'
&RCPPSCB(1) SETB  1
&RCPECT(1) SETB  1
        EJECT
        SPACE 4
*****
***    THE USERID OF THE USER IS CHECKED. IF IT IS NOT VALID, THE    ****
***    COMMAND PRETENDS IT DOES NOT EXIST BY LINKING TO EXEC IN      ****
***    THE SAME WAY THE TMP DOES IF IT CANNOT FIND THE COMMAND.      ****
*****
        SPACE 3
        L     R1, CPPLPSCB          LOAD ADDRESS OF PSCB
        USING PSCB, R1             PSCB ADDRESSABILITY
.NID    ANOP
&COUNTR SETA  &COUNTR+1
        AIF  ('&CHECKID(&COUNTR)' EQ '').ENDID
&CHARVAR SETC '&CHECKID(&COUNTR)'
&L      SETA  K'&CHARVAR
        AIF  ('&CHARVAR'(1,1) EQ ''').QCID
        CLC  PSCBUSER(&L), =C'&CHARVAR' IS THE USERID VALID?
        BE   &P.IDOK              YES, BRANCH OUT
        AGO  .NID
.QCID   ANOP
&L      SETA  &L-2
        CLC  PSCBUSER(&L), =C&CHARVAR IS THE USERID VALID?
        BE   &P.IDOK              YES, BRANCH OUT
        AGO  .NID
.ENDID  L     R1, CPPLECT          LOAD ECT ADDRESS
        SPACE 2
        USING ECT, R1
        MVC  ECTPCMD, &P.EXECN    MOVE IN COMMAND NAME
        DROP R1                   KILL ECT ADDRESSABILITY
        L    R1, CPPLCBUF         LOAD CBUF ADDRESS
        XC   2(2, R1), 2(R1)     ZERO OFFSET FIELD
        L    R1, &P.CPPL         RELOAD CPPL ADDRESS
        XCTL EPLOC=&P.EXECN
&P.EXECN DC  CL8'EXEC'          NAME OF EXEC PROCESSOR
&P.IDOK  DS   0H
        MEND

```

IEB817I MEMBER NAME (RCPCKID) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPDDN
        MACRO
        RCPDDN &DDN
        GBLC  &DYNP
        SPACE 1
*****
**      BUILD THE DDNAME TEXT UNIT                               **
*****
        AIF   ('&DDN'(K'&DDN,1) EQ '/') .BTU
        AIF   ('&DDN'(1,1) EQ ''') .Q
        RCPSR2
        AIF   ('&DDN'(1,1) EQ '(') .R
        L     R14,&DDN          LOAD ADDRESS OF DDNAME
        LH    R2,&DDN+4         LOAD LENGTH OF DDNAME
        AGO   .STH
        .R    L     R14,0&DDN    LOAD ADDRESS OF DDNAME
        LH    R2,4&DDN         LOAD LENGTH OF DDNAME
        .STH  STH  R2,S99TULNG   STORE DDNAME LENGTH
        BCTR  R2,0             DECREMENT FOR EXECUTE
        EX   R2,&DYNP.MVC      MOVE DDNAME
        MVI  S99TUKEY+1,DALDDNAM MOVE IN DDNAME KEY
        MVI  S99TUNUM+1,1     SET NUMBER FIELD
        RCPDINC 14
        MEXIT
        .Q    RCPBTU DALDDNAM,1,&DDN
        MEXIT
        .BTU  RCPTUBFR DALDDNAM,14,&DDN
        MEND

```

IEB817I MEMBER NAME (RCPDDN) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDDNRT
        MACRO
        RCPDDNRT
        SPACE 1
*****
**      DDNAME RETURN TEXT UNIT                               **
*****
        MVI    S99TUKEY+1,DALRTDDN      SET RETURN DDNAME KEY
        MVI    S99TUNUM+1,1             SET NUMBER FIELD
        MVI    S99TULNG+1,8            SET LENGTH FIELD
        MVC    S99TUPAR(8),=CL8' '     INITIALIZE FIELD TO BLANKS
        RCPDINC 14
        MEND
```

IEB817I MEMBER NAME (RCPDDNRT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDEBUG
        MACRO
        RCPDEBUG &ON
        GBLA  &RCPBGN#,&RCPSWS(10)
        GBLB  &RCPDEBUG
        GBLC  &RCPPRE,&RCPWKDS,&RCPWKCS
&RCPDEBUG SETB 1
.TSW    AIF    ('&ON' EQ '').TSW
        MEXIT
.DEBUG  MNOTE *, 'RCPBGN# IS &RCPBGN#'
        MNOTE *, 'RCPSWS(1) IS &RCPSWS(1)'
        MNOTE *, 'RCPSWS(2) IS &RCPSWS(2)'
        MNOTE *, 'RCPSWS(3) IS &RCPSWS(3)'
        MNOTE *, 'RCPSWS(4) IS &RCPSWS(4)'
        MNOTE *, 'RCPSWS(5) IS &RCPSWS(5)'
        MNOTE *, 'RCPWKCS IS ''&RCPWKCS'''
        MNOTE *, 'RCPWKDS IS ''&RCPWKDS'''
        MNOTE *, 'RCPPRE IS ''&RCPPRE'''
        MEND
```

IEB817I MEMBER NAME (RCPDEBUG) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPDFPL
        MACRO
        RCPDFPL
        GBLC   &RCPPRE
        GBLB   &RCPDFPL(2)
        GBLB   &RCPDFPB(2)
&P      LCLC   &P,&L,&L1
        SETC   '&RCPPRE'
        EJECT
&RCPDFPL(2) SETB 1
        IKJDFPL
L#DFPL  EQU    *-DFPL          LENGTH OF DEFAULT PARAM LIST
        IKJDFPB
L#DFPB  EQU    *-DFPB          LENGTH OF DEFAULT PARAM BLOCK
&SYSECT CSECT                  RESUME PROGRAM CSECT
        SPACE 3
        .BYPDFPL RCPDS
&P.DFPL DS     CL(L#DFPL)      RESERVE SPACE FOR DFPL
&P.DFPB DS     CL(L#DFPB)      RESERVE SPACE FOR DFPB
&P.DSNB DS     CL48            RESERVE SPACE FOR DSNAME BUFFER
        RCPDS
        EJECT

```

```

*****
***      THIS CODE GENERATES AN DEFAULT SERVICE ROUTINE PARAMETER LIST ***
***      AND PARAMETER BLOCK ***
*****

```

```

        LA     R1,&P.DFPL        LOAD DFPL ADDRESS
        USING DFPL,R1           DFPL ADDRESSABLE
        MVC    DFPLUPT,CPPLUPT   MOVE IN ADDRESS OF UPT
        MVC    DFPLECT,CPPLECT   MOVE IN ADDRESS OF ECT
        LA     R15,&P.ECB        LOAD ADDRESS OF ATTN ECB
        ST     R15,DFPLECB       AND STORE IN DFPL
        LA     R15,&P.DFPB       LOAD DFPB ADDRESS
        ST     R15,DFPLDFPB      AND STORE IT IN DFPB
        DROP   R1
        USING  DFPB,R15         ADDRESS DFPB DSECT
        XC     DFPB(L#DFPB),DFPB CLEAR DEFAULT PARAMETER BLOCK
        MVC    DFPBPSCB,CPPLPSCB MOVE IN ADDRESS OF PSCB
        LA     R1,&P.DSNB        LOAD DSNAME BUFFER ADDRESS
        ST     R1,DFPBDSN       AND STORE IT INTO DFPB
        MVI    DFPBCODE,DFPB04   SET ENTRY CODE
        DROP   R15              DFPB NO LONGER ADDRESSABLE
        EJECT
        MEND

```

IEB817I MEMBER NAME (RCPDFPL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDINC
        MACRO
        RCPDINC &L1
        GBLA   &DTUO,&DTUPO
        GBLC   &DYNP
        AIF    ('&L1' EQ '').T2
        ST     R15,&DYNP.TUP+&DTUPO    STORE TEXT UNIT ADDRESS
        LA     R15,&L1.(R15)          BUMP TEXT UNIT PTR TO NEXT SLOT
&DTUPO  SETA   &DTUPO+4
&DTUO   SETA   &DTUO+&L1
        MEXIT
.T2     ST     R14,&DYNP.TUP+&DTUPO    STORE TEXT UNIT ADDRESS
&DTUPO  SETA   &DTUPO+4
        MEND
```

IEB817I MEMBER NAME (RCPDINC) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDISP
        MACRO
        RCPDISP &DISP
        LCLA   &I
        LCLB   &B(4)
        AIF    ('&DISP(1)' EQ ' ').TD2
        SPACE
*****
**      DATA SET INITIAL STATUS                                     **
*****
&B(1)   SETB   ('&DISP(1)' EQ 'SHR')
&B(2)   SETB   ('&DISP(1)' EQ 'NEW')
&B(3)   SETB   ('&DISP(1)' EQ 'MOD')
&B(4)   SETB   ('&DISP(1)' EQ 'OLD')
        AIF    (&B(1) OR &B(2) OR &B(3) OR &B(4)).OK1
        MNOTE  8, '&DISP(1) IS INVALID, DISP=SHR USED'
&B(1)   SETB   1
.OK1    ANOP
&I      SETA   8*&B(1)+4*&B(2)+2*&B(3)+&B(4)
        MVC    S99TUKEY(8), =Y(DALSTATS,1,1,X'0&I.00')
        RCPDINC 8
.TD2    AIF    ('&DISP(2)' EQ ' ').TD3
        SPACE
*****
**      DATA SET NORMAL DISPOSITION                               **
*****
&B(1)   SETB   ('&DISP(2)' EQ 'KEEP')
&B(2)   SETB   ('&DISP(2)' EQ 'DELETE')
&B(3)   SETB   ('&DISP(2)' EQ 'CATLG')
&B(4)   SETB   ('&DISP(2)' EQ 'UNCATLG')
        AIF    (&B(1) OR &B(2) OR &B(3) OR &B(4)).OK2
        MNOTE  8, '&DISP(2) IS INVALID, DISP=(,KEEP) USED'
&B(1)   SETB   1
.OK2    ANOP
&I      SETA   8*&B(1)+4*&B(2)+2*&B(3)+&B(4)
        MVC    S99TUKEY(8), =Y(DALNDISP,1,1,X'0&I.00')
        RCPDINC 8
.TD3    AIF    ('&DISP(3)' EQ ' ').EXIT
        SPACE
*****
**      DATASET CONDITIONAL DISPOSITION                           **
*****
&B(1)   SETB   ('&DISP(3)' EQ 'KEEP')
&B(2)   SETB   ('&DISP(3)' EQ 'DELETE')
&B(3)   SETB   ('&DISP(3)' EQ 'CATLG')
&B(4)   SETB   ('&DISP(3)' EQ 'UNCATLG')
        AIF    (&B(1) OR &B(2) OR &B(3) OR &B(4)).OK3
        MNOTE  8, '&DISP(3) IS INVALID, DISP=(,,KEEP) USED'
&B(1)   SETB   1
.OK3    ANOP
&I      SETA   8*&B(1)+4*&B(2)+2*&B(3)+&B(4)
        MVI    S99TUKEY(8), =Y(DALCDISP,1,1,X'0&I.00')
```


RCPDINC 8

.EXIT MEND

IEB817I MEMBER NAME (RCPDISP) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD      NAME=RCPDS
        MACRO
        RCPDS
        GBLB     &RCPDSBR
        GBLC     &RCPWKDS,&RCPWKCS,&RCPDS
        AIF      ('&RCPDS' NE '').RESUME
&RCPDS  SETC     '&SYSECT'
        AIF      ('&RCPWKDS' EQ '').CSECT
&RCPWKDS DSECT   ENTER WORKAREA DSECT
        MEXIT
.CSECT  AIF      ('&RCPWKCS' EQ '').BRANCH
&RCPWKCS CSECT   ENTER WORKAREA CSECT
        MEXIT
.RESUME AIF      (&RCPDSBR).BRTO
&RCPDS  CSECT   RESUME PROGRAM CSECT
&RCPDS  SETC     ''
        MEXIT
.BRANCH ANOP
&RCPDS  SETC     'RCP&SYSNDX'
&RCPDSBR SETB    1
        B       &RCPDS          BRANCH AROUND CONSTANTS
        MEXIT
.BRTO   ANOP
&RCPDS  DS       0H
&RCPDSBR SETB    0
&RCPDS  SETC     ''
        MEND
```

IEB817I MEMBER NAME (RCPDS) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDSECT
        MACRO
&NAME  RCPDSECT &LTORG=YES
        AIF    ('&LTORG' NE 'YES') .RCPDS
*****
        LITERALS
*****
        SPACE 3
        LTORG
        EJECT
.RCPDS  RCPDS
        MEND
```

IEB817I MEMBER NAME (RCPDSECT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPDSN
        MACRO
        RCPDSN &DSN,&MEM
        LCLC  &MEMBER
        GBLC  &DYNP
        SPACE
*****
**      BUILD THE DSNAME TEXT UNIT      **
*****
        AIF   ('&DSN'(1,1) EQ ' ').Q
        AIF   ('&DSN'(K'&DSN,1) EQ '/').BD
        AIF   ('&DSN'(1,1) EQ '(').REG
        AIF   ('&DSN' EQ '*').TERM
        RCPSR2
        L     R14,&DSN          LOAD ADDRESS OF DSNAME
        LH    R2,&DSN+4        LOAD LENGTH OF DSNAME
        .STH  STH R2,S99TULNG  STORE DSNAME LENGTH
        BCTR  R2,0            DECREMENT FOR EXECUTE
        EX    R2,&DYNP.MVC     MOVE DSNAME
        MVI   S99TUKEY+1,DALDSNAM MOVE IN DSNAME KEY
        MVI   S99TUNUM+1,1    SET NUMBER FIELD
        RCPDINC 50
        .REG  L     R14,0&DSN  LOAD ADDRESS OF DSNAME
        RCPSR2
        LH    R2,4&DSN        LOAD LENGTH OF DSNAME
        AGO   .STH
        .TERM MVI   S99TUKEY+1,DALTERM
        RCPDINC 4
        MEXIT
        .BD   RCPTUBFR DALDSNAM,50,&DSN
        AGO   .TMEMBER
        .Q    RCPBTU DALDSNAM,1,&DSN
        .TMEMBER AIF ('&MEM' EQ ' ').EXIT
        SPACE
*****
**      BUILD THE MEMBER NAME TEXT UNIT  **
*****
&MEMBER SETC  '&MEM'
        AIF   ('&MEM' NE '*').MOK
        AIF   ('&DSN'(1,1) NE ' ').MAST
        MNOTE 8,'MEMBER=* INVALID WITH QUOTED DSNAME'
        MEXIT
        .MAST ANOP
&MEMBER SETC  '8+&DSN'
        .MOK ANOP
        AIF   ('&MEMBER'(K'&MEMBER,1) EQ '/').BM
        RCPSR2
        AIF   ('&MEMBER'(1,1) EQ '(').RM
        LH    R2,4+&MEMBER    LOAD LENGTH OF MEMBER NAME
        LTR   R2,R2           TEST FOR ZERO
        BZ    *+30           IF NO MEMBER, SKIP

```

```
      L      R14,&MEMBER          LOAD ADDRESS OF MEMBER
      AGO     .STHM
.RM     LH     R2,4&MEMBER        LOAD LENGTH OF MEMBER
      LTR     R2,R2              AND TEST FOR ZERO
      BZ      *+30              IF NO MEMBER, SKIP
      L      R14,0&MEMBER        LOAD ADDRESS OF MEMBER
      .STHM  STH     R2,S99TULNG  STORE LENGTH OF MEMBER
      BCTR    R2,0              DECREMENT FOR EXECUTE
      EX      R2,&DYNP.MVC       MOVE IN MEMBER NAME
      MVI     S99TUKEY+1,DALMEMBR MOVE IN MEMBER KEY
      MVI     S99TUNUM+1,1       SET NUMBER FIELD
      RCPDINC 14
      MEXIT
      .BM     RCPTUBFR DALMEMBR,14,&MEMBER
      MEXIT
      .QM     RCPBTU DALMEMBR,1,&MEMBER
      .EXIT   MEND
```

IEB817I MEMBER NAME (RCPDSN) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDSNPD  
        MACRO  
        RCPDSNPD &PDE  
        AIF    ('&PDE'(1,1) EQ ' ').RPDE  
        RCPDSN &PDE,8+&PDE  
        RCPPSWD 16+&PDE  
        MEXIT  
.RPDE  RCPDSN &PDE,8&PDE  
        RCPPSWD 16(&PDE)  
        MEND
```

IEB817I MEMBER NAME (RCPDSNPD) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDSNRT
        MACRO
        RCPDSNRT
        SPACE
*****
**      DSNAME RETURN TEXT UNIT                               **
*****
        MVI    S99TUKEY+1,DALRTDSN      SET RETURN DSNAME KEY
        MVI    S99TUNUM+1,1             SET NUMBER FIELD
        MVI    S99TULNG+1,44           SET LENGTH FIELD
        RCPDINC 50
        MEND
```

IEB817I MEMBER NAME (RCPDSNRT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPDSRGR  
        MACRO  
        RCPDSRGR  
        SPACE
```

```
**      DSORG RETURN TEXT UNIT      **
```

```
        MVI    S99TUKEY+1,DALRTORG    SET RETURN DSORG KEY  
        MVI    S99TUNUM+1,1          SET NUMBER FIELD  
        MVI    S99TULNG+1,2          SET LENGTH FIELD  
        XC     S99TUPAR(2),S99TUPAR    INITIALIZE FIELD TO ZERO  
        RCPDINC 8  
        MEND
```

IEB817I MEMBER NAME (RCPDSRGR) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD    NAME=RCPDUMMY  
        MACRO  
        RCPDUMMY &DUMMY  
        SPACE
```

```
*****
```

```
**      DUMMY DATASET TEXT UNIT      **
```

```
*****
```

```
        MVI    S99TUPAR+1,DALDUMMY    MOVE IN DUMMY DS TEXT UNIT KEY  
        RCPDINC 4  
        MEND
```

IEB817I MEMBER NAME (RCPDUMMY) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPENDD
        MACRO
&NAME  RCPENDD
        GBLB  &RCPECT(2),&RCPUPT(2),&RCPPSCB(2),&RCPS99(2)
        GBLC  &RCPPRE,&RCPWKDS,&RCPDS
        LCLC  &P,&CS
&CS    SETC   '&RCPDS'          PROGRAM CSECT
        AIF   (NOT &RCPS99(1)).TDS
        DYNSPACE
        .TDS  AIF   ('&RCPWKDS' EQ ' ').RCPDS
        DS    0D                  ALIGN TO DOUBLEWORD
&P     SETC   '&RCPPRE'
&P.WKLEN EQU  *-&RCPWKDS        LENGTH OF WORK AREA
.RCPDS RCPDS
        EJECT
        AIF   (NOT &RCPECT(1) OR &RCPECT(2)).TRYUPT
&CS    IKJECT
        CSECT                    REENTER MAIN CSECT
        EJECT
&RCPECT(2) SETB 1
        .TRYUPT AIF   (NOT &RCPUPT(1) OR &RCPUPT(2)).TRYPSCB
        IKJUPT
&CS    CSECT                    REENTER MAIN CSECT
        EJECT
&RCPUPT(2) SETB 1
        .TRYPSCB AIF   (NOT &RCPPSCB(1) OR &RCPPSCB(2)).TRYS99
        IKJPSCB
&CS    CSECT                    REENTER MAIN CSECT
        EJECT
&RCPPSCB(2) SETB 1
        .TRYS99 AIF   (NOT &RCPS99(1) OR &RCPS99(2)).TRYREST
        IEFZB4D0
        EJECT
        IEFZB4D2
&CS    CSECT                    REENTER MAIN CSECT
        EJECT
&RCPS99(2) SETB 1
        .TRYREST MEND

```

IEB817I MEMBER NAME (RCPENDD) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPFDDN
        MACRO
        RCPFDDN &DDN
        GBLC &DYNP
        SPACE
*****
**      FREE DDNAME TEXT UNIT      **
*****
        AIF    ('&DDN'(1,1) EQ ' ').Q
        AIF    ('&DDN'(K'&DDN,1) EQ '/').B
        RCPSR2
        AIF    ('&DDN'(1,1) EQ ' ').R
        L      R14,&DDN              LOAD ADDRESS OF DDNAME
        LH     R2,&DDN+4             LOAD LENGTH OF DDNAME
        AGO    .STH
        .R     L      R14,0&DDN      LOAD ADDRESS OF DDNAME
        LH     R2,4&DDN             LOAD LENGTH OF DDNAME
        .STH   STH    R2,S99TULNG    STORE DDNAME LENGTH
        BCTR  R2,0                 DECREMENT FOR EXECUTE
        EX    R2,&DYNP.MVC          MOVE DDNAME
        MVI   S99TUKEY+1,DUNDDNAM   MOVE IN DDNAME KEY
        MVI   S99TUNUM+1,1         SET NUMBER FIELD
        RCPDINC 14
        MEXIT
        .Q    RCPBTU DUNDDNAM,1,&DDN
        MEXIT
        .B    RCPTUBFR DUNDDNAM,14,&DDN
        MEND

```

IEB817I MEMBER NAME (RCPFDDN) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPFDISP
        MACRO
        RCPFDISP &DISP
        LCLB  &B(4)
        LCLA  &I
        SPACE
*****
**      OVERRIDING DISPOSITION      **
*****
&B(1)   SETB  ('&DISP' EQ 'KEEP')
&B(2)   SETB  ('&DISP' EQ 'DELETE')
&B(3)   SETB  ('&DISP' EQ 'CATLG')
&B(4)   SETB  ('&DISP' EQ 'UNCATLG')
        AIF  (&B(1) OR &B(2) OR &B(3) OR &B(4)).OK3
        MNOTE 8,'&DISP IS INVALID, DISP=KEEP USED'
&B(1)   SETB  1
        .OK3  ANOP
&I      SETA  8*&B(1)+4*&B(2)+2*&B(3)+&B(4)
        MVC   S99TUKEY(8),=Y(DUNOVDSP,1,1,X'0&I.00')
        RCPDINC 8
        .EXIT  MEND
```

IEB817I MEMBER NAME (RCPFDISP) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPFDSN
        MACRO
        RCPFDSN &DSN,&MEM
        LCLC   &MEMBER
        GBLC   &DYNP
        SPACE
*****
**      FREE DATA SET TEXT UNIT      **
*****
        AIF    ('&DSN'(1,1) EQ ' ').Q
        AIF    ('&DSN'(K'&DSN,1) EQ '/').BD
        AIF    ('&DSN'(1,1) EQ '(').REG
        RCPSR2
        L      R14,&DSN          LOAD ADDRESS OF DSNAME
        LH     R2,&DSN+4        LOAD LENGTH OF DSNAME
        .STH   STH R2,S99TULNG  STORE DSNAME LENGTH
        BCTR   R2,0            DECREMENT FOR EXECUTE
        EX     R2,&DYNP.MVC     MOVE DSNAME
        MVI    S99TUKEY+1,DUNDSNAM MOVE IN DSNAME KEY
        MVI    S99TUNUM+1,1    SET NUMBER FIELD
        RCPDINC 50
        .REG   L      R14,0&DSN  LOAD ADDRESS OF DSNAME
        RCPSR2
        LH     R2,4&DSN        LOAD LENGTH OF DSNAME
        AGO    .STH
        .BD    RCPTUBFR DUNDSNAM,50,&DSN
        AGO    .TMEMBER
        .Q     RCPBTU DUNDSNAM,1,&DSN
        .TMEMBER AIF ('&MEM' EQ ' ').EXIT
        SPACE
*****
**      FREE MEMBER NAME TEXT UNIT    **
*****
&MEMBER SETC   '&MEM'
        AIF    ('&MEM' NE '*').MOK
        AIF    ('&DSN'(1,1) NE ' ').MAST
        MNOTE  8,'MEMBER=* INVALID WITH QUOTED DSNAME'
        MEXIT
        .MAST ANOP
&MEMBER SETC   '8+&DSN'
        .MOK  ANOP
        AIF    ('&MEMBER'(K'&MEMBER,1) EQ '/').BM
        RCPSR2
        AIF    ('&MEMBER'(1,1) EQ '(').RM
        LH     R2,4+&MEMBER    LOAD LENGTH OF MEMBER NAME
        LTR    R2,R2          TEST FOR ZERO
        BZ     *+30          IF NO MEMBER, SKIP
        L      R14,&MEMBER    LOAD ADDRESS OF MEMBER
        AGO    .STHM
        .RM    LH     R2,4&MEMBER  LOAD LENGTH OF MEMBER
        LTR    R2,R2          AND TEST FOR ZERO

```

```
.STHM  BZ      *+30          IF NO MEMBER, SKIP
        L      R14,0&MEMBER  LOAD ADDRESS OF MEMBER
        STH    R2,S99TULNG   STORE LENGTH OF MEMBER
        BCTR   R2,0          DECREMENT FOR EXECUTE
        EX     R2,&DYNP.MVC   MOVE IN MEMBER NAME
        MVI    S99TUKEY+1,DUNMEMBR MOVE IN MEMBER KEY
        MVI    S99TUNUM+1,1   SET NUMBER FIELD
        RCPDINC 14
        MEXIT
.BM     RCPTUBFR DUNMEMBR,14,&MEMBER
        MEXIT
.QM     RCPBTU DUNMEMBR,1,&MEMBER
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPFDSN) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPFHOLD
        MACRO
        RCPFHOLD &H
        AIF    ('&H' EQ 'YES').YES
        AIF    ('&H' EQ 'NO').NO
        MNOTE 4,'HOLD PARMETER VALUE INCORRECT - IGNORED'
        MEXIT
.YES    ANOP
        SPACE 1
*****
**      OVERIDING SYSOUT HOLD TEXT UNIT          **
*****
        SPACE 1
        MVI    S99TUKEY+1,DUNOVSHQ MOVE IN TEXT UNIT KEY
        RCPDINC 4
        MEXIT
.NO     ANOP
        SPACE 1
*****
**      OVERIDING SYSOUT NO HOLD TEXT UNIT       **
*****
        SPACE 1
        MVI    S99TUKEY+1,DUNOVSHQ MOVE IN TEXT UNIT KEY
        RCPDINC 4
        MEND
```

IEB817I MEMBER NAME (RCPFHOLD) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPFORUS
        MACRO - TO SET UP SVC 99 TEXT UNIT 'FOR USER'
        RCPFORUS &T
        SPACE 1
*****
**      'FOR USER' TEXT UNIT                               **
*****
        RCPVCHAR 0,5,&T,N=X'7701'
        MEND
```

IEB817I MEMBER NAME (RCPFORUS) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD  NAME=RCPFREE
        MACRO
        RCPFREE &FREE
        SPACE
*****
**      UNALLOC AT CLOSE TEXT UNIT          **
*****
        MVI  S99TUPAR+1,DALCLOSE      MOVE IN CLOSE TEXT UNIT KEY
        RCPDINC 4
        MEND
```

IEB817I MEMBER NAME (RCPFREE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD      NAME=RCPIOPL
        MACRO
&NAME  RCPIOPL
        GBLC    &RCPPRE
        GBLB    &RCPIOPL(2)
        GBLB    &RCPSTPB(2),&RCPPTPB(2),&RCPGTPB(2),&RCPGTPB(2)
&P     LCLC    &P,&L,&L1
        SETC    '&RCPPRE'
        EJECT
        AIF     (&RCPIOPL(2)).BYPIOPL
&RCPIOPL(2) SETB 1
        IKJIOPL
L#IOPL EQU    *-IOPL          LENGTH OF IO PARAM LIST
&SYSECT CSECT                RESUME PROGRAM CSECT
        SPACE 3
        .BYPIOPL RCPDS
&P.IOPL DS     CL(L#IOPL)     RESERVE SPACE FOR IOPL
        RCPDS
        SPACE 5
*****
***      THIS CODE GENERATES AN I/O SERVICE ROUTINE PARAMETER LIST      ***
*****
        LA     R1,&P.IOPL      LOAD IOPL ADDRESS
        USING IOPL,R1         IOPL ADDRESSABLE
        MVC    IOPLUPT,CPPLUPT MOVE IN ADDRESS OF UPT
        MVC    IOPLECT,CPPLECT MOVE IN ADDRESS OF ECT
        LA     R15,&P.ECB      LOAD ADDRESS OF ATTN ECB
        ST     R15,IOLECB     AND STORE IN IOPL
        DROP  R1
        AIF    (&RCPSTPB(1) OR &RCPGTPB(1) OR &RCPGTPB(1) OR &RCPPTPB(1)).I
        MEXIT
        EJECT
        AIF    (NOT &RCPSTPB(1) OR &RCPSTPB(2)).TPT
        IKJSTPB
&RCPSTPB(2) SETB 1
L#STPB EQU    *-STPB          LENGTH OF STPB
&SYSECT CSECT
        .TPT  AIF    (NOT &RCPPTPB(1) OR &RCPPTPB(2)).TGT
        IKJPTPB
&RCPPTPB(2) SETB 1
L#PTPB EQU    *-PTPB          LENGTH OF PTPB
&SYSECT CSECT
        .TGT  AIF    (NOT &RCPGTPB(1) OR &RCPGTPB(2)).TPG
        IKJGTPB
&RCPGTPB(2) SETB 1
L#GTPB EQU    *-GTPB          LENGTH OF GTPB
&SYSECT CSECT
        .TPG  AIF    (NOT &RCPGTPB(1) OR &RCPGTPB(2)).STO
        IKJPGPB
&RCPGTPB(2) SETB 1
L#PGPB EQU    *-PGPB          LENGTH OF PGPB
&SYSECT CSECT

```

```
.STO      SPACE 3
&L        SETC  ''
          RCPDS
          AIF   (NOT &RCPSTPB(1)).XPT
&P.STPB   DS    CL(L#STPB)          RESERVE SPACE FOR STPB
&L        SETC  '&L.+L#STPB'
          AIF   (NOT &RCPPTPB(1)).XGT
&P.PTPB   DS    CL(L#PTPB)         RESERVE SPACE FOR PTPB
&L        SETC  '&L.+L#PTPB'
          AIF   (NOT &RCPGTPB(1)).XPG
&P.GTPB   DS    CL(L#GTPB)         RESERVE SPACE FOR GTPB
&L        SETC  '&L.+L#GTPB'
          AIF   (NOT &RCPGTPB(1)).XPG
&P.PGPB   DS    CL(L#PGPB)         RESERVE SPACE FOR PGPB
&L        SETC  '&L.+L#PGPB'
          RCPDS
          SETC  '&L'(2,K'&L-1)
&L1       SETC  '&P'.'&L1'(3,4)
&L        XC    &L.(&L1.),&L      CLEAR IOPB AREA
          MEND
```

IEB817I MEMBER NAME (RCPIOPL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD     NAME=RCPLINK
        MACRO
&NAME  RCPLINK &MODULE
        LCLC   &OFFSET,&C
        AIF    ('&MODULE' EQ  ' ').ERROR
        AIF    ('&MODULE' NE  'IKJPARS').T1
&OFFSET SETC  '524'
        AGO    .START
.T1     AIF    ('&MODULE' NE  'IKJDAIR').T2
&OFFSET SETC  '732'
        AGO    .START
.T2     AIF    ('&MODULE' NE  'IKJEHDEF').T3
&OFFSET SETC  '736'
        AGO    .START
.T3     AIF    ('&MODULE' NE  'IKJEHCIR').T4
&OFFSET SETC  '740'
        AGO    .START
.T4     AIF    ('&MODULE' NE  'IKJPUTL').T5
&OFFSET SETC  '444'
        AGO    .START
.T5     AIF    ('&MODULE' NE  'IKJGETL').T6
&OFFSET SETC  '348'
        AGO    .START
.T6     AIF    ('&MODULE' NE  'IKJSCAN').T7
&OFFSET SETC  '480'
        AGO    .START
.T7     AIF    ('&MODULE' NE  'IKJPTGT').T8
&OFFSET SETC  '464'
        AGO    .START
.T8     AIF    ('&MODULE' NE  'IKJSTCK').T9
&OFFSET SETC  '472'
        AGO    .START
.T9     ANOP
&NAME   DS     0H
*
  MNOTE *, ' EP OF &MODULE. NOT IN CVT. STANDARD LINK USED'
*
        AGO    .LINK
.START  ANOP
&NAME  L       R15,16          LOAD CVT ADDRESS
        L       R15,&OFFSET.(R15)  LOAD MODULE ADDRESS
        LTR     R15,R15          IS MODULE ADDRESS THERE?
&C     SETC    'RCP&SYSNDX'
        BNM    &C.L             IF NOT, BRANCH TO LINK
        BALR   R14,R15          ELSE BALR TO IT
        B      &C.B             AND BYPASS LINK
&C.L   LINK    EP=&MODULE
&C.B   DS      0H              BRANCHED TO IF LINK BYPASSED

        MEXIT
.LINK  ANOP
&NAME  LINK    EP=&MODULE
        MEXIT

```

.ERROR MNOTE 4, 'NO MODULE NAME SPECIFIED'
MEND

IEB817I MEMBER NAME (RCPLINK) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPLOAD
        MACRO
&NAME  RCPLOAD &MOD,&EP1
        GBLC  &RCPPTEP,&RCPGTEP,&RCPPEGEP
        GBLC  &RCPDFEP,&RCPSTEP,&RCPPREP
        GBLC  &RCPPRE
        LCLA  &I,&J
        LCLB  &EPXISTS
        LCLC  &OFFSET,&C,&EP,&MODULE
&EP    SETC  '&EP1'
&MODULE SETC  '&MOD'
        AIF   ('&MODULE' EQ ' ').ERROR
        AIF   ('&MODULE'(K'&MOD,1) NE ' ')'.NOBR
&I     SETA  K'&MOD
        .LOOP
&I     SETA  &I-1
        AIF   (&I LT 2).NOLB
        AIF   ('&MOD'(&I,1) NE ' ').LOOP
&MODULE SETC  '&MOD'(1,&I-1)
&J     SETA  K'&MOD-1-&I
&EP    SETC  '&MOD'(&I+1,&J)
        RCPDS
&EP    DS    F                                TO STORE MODULE ADDRESS
        RCPDS
        .NOBR
&EPXISTS SETB ('&EP' NE ' ')
        AIF   ('&MODULE' NE 'IKJPARS').T1
&OFFSET SETC  '524'
&RCPPREP SETC  '&EP'
        AIF   (&EPXISTS).START
        RCPDS
&RCPPREP SETC  '&RCPPRE.PREP'
&EP     SETC  '&RCPPREP'
&RCPPREP DS    F                                TO HOLD ADDRESS OF IKJPARS
        RCPDS
        AGO   .START
        .T1
&OFFSET SETC  '732'
        AGO   .START
        .T2
&RCPDFEP SETC  '&EP'
&OFFSET SETC  '736'
        AIF   (&EPXISTS).START
&RCPDFEP SETC  '&RCPPRE.DFEP'
        RCPDS
&RCPDFEP DS    F                                ADDR OF DEFAULT SERVICE ROUTINE
        RCPDS
&EP     SETC  '&RCPDFEP'
        AGO   .START
        .T3
&OFFSET SETC  '740'
        AGO   .START

```

```

.T4      AIF      ('&MODULE' NE 'IKJPUTL').T5
&RCPPTTEP SETC    '&EP'
&OFFSET   SETC    '444'
          AIF      (&EPXISTS).START
&RCPPTTEP SETC    '&RCPPRE.PTEP'
&EP       SETC    '&RCPPTTEP'
          RCPDS
&RCPPTTEP DS      F              ADDR OF PUTLINE ROUTINE
          RCPDS
          AGO      .START
.T5      AIF      ('&MODULE' NE 'IKJGETL').T6
&RCPGTEP  SETC    '&EP'
&OFFSET   SETC    '348'
          AIF      (&EPXISTS).START
&RCPGTEP  SETC    '&RCPPRE.GTEP'
&EP       SETC    '&RCPGTEP'
          RCPDS
&RCPGTEP  DS      F              ADDR OF GETLINE ROUTINE
          RCPDS
          AGO      .START
.T6      AIF      ('&MODULE' NE 'IKJSCAN').T7
&OFFSET   SETC    '480'
          AGO      .START
.T7      AIF      ('&MODULE' NE 'IKJPTGT').T8
&RCPPEGEP SETC    '&EP'
&OFFSET   SETC    '464'
          AIF      (&EPXISTS).START
&RCPPEGEP SETC    '&RCPPRE.PGEP'
&EP       SETC    '&RCPPEGEP'
          RCPDS
&RCPPEGEP DS      F              ADDR OF PUTGET ROUTINE
          RCPDS
          AGO      .START
.T8      AIF      ('&MODULE' NE 'IKJSTCK').T9
&RCPSTEP  SETC    '&EP'
&OFFSET   SETC    '472'
          AIF      (&EPXISTS).START
&RCPSTEP  SETC    '&RCPPRE.STEP'
&EP       SETC    '&RCPSTEP'
          RCPDS
&RCPSTEP  DS      F              ADDR OF STACK ROUTINE
          RCPDS
          AGO      .START
.T9      ANOP
&NAME     DS      0H
*
MNOTE *, ' EP OF &MODULE. NOT IN CVT. STANDARD LOAD USED'
*
          AGO      .LOAD
.START    ANOP
&NAME     L        R15,16          LOAD CVT ADDRESS
          L        R0,&OFFSET.(R15)  LOAD MODULE ADDRESS

```

```
&C      LTR    R0,R0                IS MODULE LOADED?
        SETC  'RCP&SYSNDX'
        BM    &C                    IF SO, BYPASS LOAD MACRO
.LOAD   LOAD EP=&MODULE.
        AIF   ('&EP' EQ '').EPERR
&C      ST    R0,&EP                STORE ENTRY POINT ADDRESS
        MEXIT
.EPERR  MNOTE 4,'EP RETURN FIELD NOT SPECIFIED'
        MEXIT
.ERROR  MNOTE 4,'NO MODULE NAME SPECIFIED'
        MEXIT
.NOLB   MNOTE 4,'INVALID MODULE NAME ''&MOD'''
        MEND
```

IEB817I MEMBER NAME (RCPLOAD) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD      NAME=RCPLOCSW
*23456789*12345*78921234567893123456789*
```

```
MACRO
```

```
RCPLOCSW &SW
```

```
*****
.*
.*      INNER MACRO USED BY GOIF, SET, RESET AND FLIP.
.*      THE PARM PASSED IS THE SWITCH OR LIST OF SWITCHES.
.*      RCPLOCSW SCANS THE ARRAYS SET UP BY DCLSW TO SEE IF THE
.*      SWITCH BIT NAMES WERE DECLARED, AND IF A LIST WAS PASSED,
.*      WHETHER ALL THE SWITCH BITS BELONG TO THE SAME BYTE.
.*      RCPLOCSW PASSES EACH THE SWITCH BYTE NAME IN GLOBAL SETC
.*      VARIABLE &RCPDSW1 AND THE SWITCH BIT NAME IN GLOBAL SETC
.*      &RCPDSW2. IF A LIST OF SWITCHES WAS PASSED, &RCPDSW2
.*      CONTAINS THE SWITCH NAMES SEPARATED BY PLUS SIGNS.
.*      IF THE FIRST OR ONLY SWITCH WAS NOT FOUND, &RCPDSW1 IS SET
.*      TO NULL. IF A LIST OF SWITCHES IS PASSED AND ANY SWITCH IS
.*      NOT DECLARED IN THE SAME SWITCH BYTE AS THE FIRST, AN MNOTE*
.*      IS ISSUED WARNING OF POSSIBLE ERROR, BUT &RCPDSW1 IS SET
.*      TO THE NAME OF THE SWITCH BYTE CONTAINING THE FIRST SWITCH
.*      BIT IN THE LIST.
.*
.*
*****
```

```
      GBLA  &RCPDSW#,&RCPGSW#      COUNTER FOR DECLARED SWITCHES
      GBLA  &RCPDSW0                NO OF SWS FOUND BY RCPLOCSW
      GBLB  &RCPDSWD(99)            DEFER DECLARE INDICATORS
      GBLB  &RCPDSW3(20)  INVERT INDICATOR
      GBLC  &RCPDSWN(99)            SWITCH BYTE NAMES
      GBLC  &RCPDSWB(800)           SWITCH BIT NAMES
      GBLC  &RCPGSWN(99)            GENERIC SWITCH BYTE NAMES
      GBLC  &RCPGSWB(99)            GENERIC SWITCH BIT PREFIXES
      GBLC  &RCPDSW1(20)            SWITCH BYTE NAMES
      GBLC  &RCPDSW2(20)            SWITCH BIT NAME(S)
      LCLA  &I,&J,&K,&L,&M,&N
      LCLB  &NOT
      LCLC  &C,&SW1,&SW2
&RCPDSW0 SETA  0                    INITIALIZE
&N        SETA  N'&SW                NO OF SWITCHES ENTERED
&J        SETA  &RCPDSW#*8+8        INDEX TO LAST DECLARED SW BIT
.LOOP1    AIF  (&M GE &N).EXIT      LOOP FOR EACH SW
&M        SETA  &M+1
&SW2     SETC  '&SW(&M)'            SWITCH TO SEARCH FOR
&I        AIF  ('&SW2' EQ ' ').LOOP1  SKIP IF NULL
&NOT     SETA  8                    INDEX TO FIRST DECLARED SW - 1
&NOT     SETB  0
&SW2     AIF  ('&SW2'(1,1) NE ' ' AND '&SW2'(1,1) NE '-').TNOT2
&SW2     SETC  '&SW2'(2,K'&SW2-1)    REMOVE NOT SIGN
&NOT     SETB  1                    INDICATE INVERT FUNCTION
&NOT     AGO  .LOOP1A                CONTINUE
.TNOT2    AIF  (K'&SW2 LT 5).LOOP1A   CHECK LENGTH
&SW2     AIF  ('&SW2'(1,4) NE 'NOT-').LOOP1A  WAS SWITCH INVERTED?
&SW2     SETC  '&SW2'(5,K'&SW2-4)    STRIP OFF 'NOT-'
```

```

&NOT      SETB  1                INDICATE INVERTED
.LOOP1A   AIF    (&I GE &J).TGEN   SEARCH NAME ARRAY
&I        SETA  &I+1
          AIF    ('&RCPDSWB(&I)' NE '&SW2').LOOP1A
.*
.* WE FOUND IT
.*
&L        SETA  (&I-1)/8         INDEX TO BYTE NAME
&SW1      SETC  '&RCPDSWN(&L)'    GET BYTE NAME
.FOUND SW ANOP                                HAVE WE HAD IT BEFORE?
&K        SETA  0
.SWL1     AIF    (&K GE &RCPDSW0).NEWSW1
&K        SETA  &K+1
          AIF    ('&RCPDSW1(&K)' NE '&SW1').SWL1
          AIF    (&RCPDSW3(&K) NE &NOT).SWL1  ENSURE INVERT BIT THE SAME
.*
.* WE FOUND IT
.*
&RCPDSW2(&K) SETC '&RCPDSW2(&K)+&SW2'  CONCATENATE CURRENT SW
          AGO    .LOOP1            GO DO NEXT
.NEWSW1   ANOP
&RCPDSW0  SETA  &K+1             NEXT SW BYTE INDEX
&RCPDSW1(&RCPDSW0) SETC '&SW1'        BYTE NAME
&RCPDSW2(&RCPDSW0) SETC '&SW2'        BIT NAME
&RCPDSW3(&RCPDSW0) SETB (&NOT)        SET INVERT INDICATOR
          AGO    .LOOP1            GO DO NEXT
.TGEN     ANOP  SEARCH GENERIC NAME ARRAY
&I        SETA  0
&L        SETA  K'&SW2
.LOOP2    ANOP
&I        SETA  &I+1
          AIF    (&I GT &RCPGSW#).NOTFND
&C        SETC  '&RCPGSWB(&I)'
          AIF    (&L LT K'&C).LOOP2
          AIF    ('&SW2'(1,K'&C) NE '&C').LOOP2
&SW1      SETC  '&RCPGSWN(&I)'
          AGO    .FOUND SW        EUREKA
.NOTFND   MNOTE 4,'SWITCH ''&SW2'' NOT DECLARED'
          AGO    .LOOP1
.EXIT     MEND

```

IEB817I MEMBER NAME (RCPLOCSW) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD      NAME=RCPLOCS1
*23456789*12345*78921234567893123456789*
```

```
MACRO
```

```
RCPLOCSW &SW
```

```
*****
.*
.*      INNER MACRO USED BY GOIF, SET, RESET AND FLIP.
.*      THE PARM PASSED IS THE SWITCH OR LIST OF SWITCHES.
.*      RCPLOCSW SCANS THE ARRAYS SET UP BY DCLSW TO SEE IF THE
.*      SWITCH BIT NAMES WERE DECLARED, AND IF A LIST WAS PASSED,
.*      WHETHER ALL THE SWITCH BITS BELONG TO THE SAME BYTE.
.*      RCPLOCSW PASSES EACH THE SWITCH BYTE NAME IN GLOBAL SETC
.*      VARIABLE &RCPDSW1 AND THE SWITCH BIT NAME IN GLOBAL SETC
.*      &RCPDSW2. IF A LIST OF SWITCHES WAS PASSED, &RCPDSW2
.*      CONTAINS THE SWITCH NAMES SEPARATED BY PLUS SIGNS.
.*      IF THE FIRST OR ONLY SWITCH WAS NOT FOUND, &RCPDSW1 IS SET
.*      TO NULL. IF A LIST OF SWITCHES IS PASSED AND ANY SWITCH IS
.*      NOT DECLARED IN THE SAME SWITCH BYTE AS THE FIRST, AN MNOTE*
.*      IS ISSUED WARNING OF POSSIBLE ERROR, BUT &RCPDSW1 IS SET
.*      TO THE NAME OF THE SWITCH BYTE CONTAINING THE FIRST SWITCH
.*      BIT IN THE LIST.
.*
*****
```

```
      GBLA  &RCPDSW#,&RCPGSW#      COUNTER FOR DECLARED SWITCHES
      GBLA  &RCPDSW0                NO OF SWS FOUND BY RCPLOCSW
      GBLB  &RCPDSWD(99)            DEFER DECLARE INDICATORS
      GBLC  &RCPDSWN(99)            SWITCH BYTE NAMES
      GBLC  &RCPDSWB(800)           SWITCH BIT NAMES
      GBLC  &RCPGSWN(99)            GENERIC SWITCH BYTE NAMES
      GBLC  &RCPGSWB(99)            GENERIC SWITCH BIT PREFIXES
      GBLC  &RCPDSW1(20)            SWITCH BYTE NAMES
      GBLC  &RCPDSW2(20)            SWITCH BIT NAME(S)
      LCLA  &I,&J,&K,&L,&M,&N        LOCAL COUNTERS
      LCLC  &C,&SW1,&SW2
&RCPDSW0 SETA  0                    INITIALIZE
&N        SETA  N'&SW                NO OF SWITCHES ENTERED
&J        SETA  &RCPDSW#*8           INDEX TO LAST DECLARED SW BIT
.LOOP1    AIF  (&M GE &N).EXIT      LOOP FOR EACH SW
&M        SETA  &M+1
&SW2     SETC  '&SW(&M)'            SWITCH TO SEARCH FOR
          AIF  ('&SW2' EQ ' ').LOOP1  SKIP IF NULL
&I        SETA  8                    INDEX TO FIRST DECLARED SW - 1
.LOOP1A   AIF  (&I GE &J).TGEN      SEARCH NAME ARRAY
&I        SETA  &I+1
          AIF  ('&RCPDSWB(&I)' NE '&SW2').LOOP1A
.*
.*      WE FOUND IT
.*
&L        SETA  (&I-1)/8            INDEX TO BYTE NAME
&SW1     SETC  '&RCPDSWN(&L) '      GET BYTE NAME
.FOUNDSW  ANOP                       HAVE WE HAD IT BEFORE?
&K        SETA  0
```

```
.SWL1   AIF   (&K GE &RCPDSW0).NEWSW1
&K      SETA  &K+1
        AIF   ('&RCPDSW1(&K)' NE '&SW1').SWL1
.*
.* WE FOUND IT
.*
&RCPDSW2(&K) SETC '&RCPDSW2(&K)+&SW2'  CONCATENATE CURRENT SW
        AGO   .LOOP1                      GO DO NEXT
.NENSW1 ANOP
&RCPDSW0 SETA  &K+1                      NEXT SW BYTE INDEX
&RCPDSW1(&RCPDSW0) SETC '&SW1'          BYTE NAME
&RCPDSW2(&RCPDSW0) SETC '&SW2'          BIT NAME
        AGO   .LOOP1                      GO DO NEXT
.TGEN   ANOP  SEARCH GENERIC NAME ARRAY
&I      SETA  0
&L      SETA  K'&SW2
.LOOP2  ANOP
&I      SETA  &I+1
&SW1    AIF   (&I GT &RCPGSW#).NOTFND
        SETC  '&RCPGSWN(&I)'
        AIF   (&L LT K'&SW1).LOOP2
        AIF   ('&SW1'(1,&L) NE '&SW2').LOOP2
        AGO   .FOUND SW                   EUREKA
.NOTFND MNOTE 4,'SWITCH ''&SW2'' NOT DECLARED'
        AGO   .LOOP1
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPLOCS1) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD      NAME=RCPLOCS2
*23456789*12345*78921234567893123456789*
      MACRO
      RCPLOCSW &SW
*****
.*
.*      INNER MACRO USED BY GOIF, SET, RESET AND FLIP.
.*      THE PARM PASSED IS THE SWITCH OR LIST OF SWITCHES.
.*      RCPLOCSW SCANS THE ARRAYS SET UP BY DCLSW TO SEE IF THE
.*      SWITCH BIT NAMES WERE DECLARED, AND IF A LIST WAS PASSED,
.*      WHETHER ALL THE SWITCH BITS BELONG TO THE SAME BYTE.
.*      RCPLOCSW PASSES EACH THE SWITCH BYTE NAME IN GLOBAL SETC
.*      VARIABLE &RCPDSW1 AND THE SWITCH BIT NAME IN GLOBAL SETC
.*      &RCPDSW2. IF A LIST OF SWITCHES WAS PASSED, &RCPDSW2
.*      CONTAINS THE SWITCH NAMES SEPARATED BY PLUS SIGNS.
.*      IF THE FIRST OR ONLY SWITCH WAS NOT FOUND, &RCPDSW1 IS SET
.*      TO NULL. IF A LIST OF SWITCHES IS PASSED AND ANY SWITCH IS
.*      NOT DECLARED IN THE SAME SWITCH BYTE AS THE FIRST, AN MNOTE*
.*      IS ISSUED WARNING OF POSSIBLE ERROR, BUT &RCPDSW1 IS SET
.*      TO THE NAME OF THE SWITCH BYTE CONTAINING THE FIRST SWITCH
.*      BIT IN THE LIST.
.*
*****
      GBLA  &RCPDSW#,&RCPGSW#      COUNTER FOR DECLARED SWITCHES
      GBLB  &RCPDSWD(99)          DEFER DECLARE INDICATORS
      GBLC  &RCPDSWN(99)          SWITCH BYTE NAMES
      GBLC  &RCPDSWB(800)        SWITCH BIT NAMES
      GBLC  &RCPGSWN(99)          GENERIC SWITCH BYTE NAMES
      GBLC  &RCPGSWB(99)          GENERIC SWITCH BIT PREFIXES
      GBLC  &RCPDSW1              SWITCH BYTE NAME
      GBLC  &RCPDSW2              SWITCH BIT NAME(S)
      LCLA  &I,&J,&K,&L,&M,&N      LOCAL COUNTERS
      LCLC  &C
&RCPDSW2 SETC  '&SW(1) '          EXTRACT 1ST SWITCH BIT
&J       SETA  &RCPDSW#*8+8      ARRAY POS OF LAST SW BIT
&I       SETA  8                  ARRAY POS-1 OF 1ST SW BIT
.LOOP1   AIF   (&I GE &J).TGEN    IF SW NOT FOUND IN 1ST ARRAY,
.*                               GO SEARCH GENERIC NAME ARRAY
&I       SETA  &I+1
      AIF   ('&RCPDSWB(&I)' NE '&RCPDSW2').LOOP1  LOOK FOR MATCH
.*
.*      OK, WE'VE FOUND A MATCH.
.*
&I       SETA  (&I-1)/8          GET POS OF SWITCH BYTE
&RCPDSW1 SETC  '&RCPDSWN(&I) '    MOVE IT TO EXIT PARM VAR
&I       SETA  &I*8+1            POINT TO 1ST SW BIT IN IT
&J       SETA  &I+8              POINT TO LAST SW BIT IN IT
&M       SETA  N'&SW              GET NO OF SWITCHES
&L       SETA  1
.*
.*      NOW WE PROCESS SUBSEQUENT SWITCHES IN THE LIST
.*

```

```

.LOOP2  AIF    (&L GE &M).EXIT      EXIT WHEN FINISHED
&L      SETA  &L+1                  POINT TO NEXT SW IN LIST
&C      SETC  '&SW(&L)'              EXTRACT IT
&RCPDSW2 SETC  '&RCPDSW2.+&C'        THEN APPEND TO PREVIOUS
.*
.*      NOW WE CHECK THAT THE SWITCH IS DECLARED IN THE SAME
.*      BYTE AS THE FIRST.
.*
&N      SETA  &I-1                  POINT TO 1ST BIT POS MINUS 1
.LOOP3  AIF    (&N GE &J).NM        IF SW NOT FOUND, ISSUE MNOTE
&N      SETA  &N+1                  POINT TO NEXT
        AIF    ('&C' NE '&RCPDSWB(&N)').LOOP3  SEARCH FOR MATCH
        AGO    .LOOP2                IF FOUND, GO PROCESS NEXT
.NM     MNOTE 4,'WARNING: SWITCH ''&C'' NOT DECLARED IN SAME BYTE AS  X
        SWITCH ''&SW(1)'' - LOGIC ERROR MAY OCCUR'
        AGO    .LOOP2                CONTINUE FOR NEXT SWITCH BIT
.*
.*      IF THE SWITCH WAS NOT LOCATED IN THE EXPLICIT NAME ARRAY,
.*      THE GENERIC NAME ARRAY IS SEARCHED.
.*
.TGEN   ANOP
&I     SETA  0
&RCPDSW2 SETC  '&SW(1)'              EXTRACT 1ST SWITCH
&L     SETA  K'&RCPDSW2              GET LENGTH OF 1ST SW
.LOOP4  AIF    (&I GE &RCPGSW#).ERROR  IF NOT SW NOT DECLARED, ERROR
&I     SETA  &I+1
&C     SETC  '&RCPGSWB(&I)'          GET GENERIC PREFIX
&K     SETA  K'&C                    GET LENGTH OF GENERIC PREFIX
        AIF    (&L LT &K).LOOP4      AND SKIP IF LEN OF SWITCH NAME
.*      < LEN OF GENERIC PREFIX
        AIF    ('&RCPDSW2'(1,&K) NE '&C').LOOP4  ALSO SKIP IF NO MATCH
&RCPDSW1 SETC  '&RCPGSWN(&I)'        SAVE SWITCH BYTE NAME
&I     SETA  1
&J     SETA  N'&SW
.LOOP5  AIF    (&I GE &J).EXIT      EXIT WHEN FINISHED
&I     SETA  &I+1
&RCPDSW2 SETC  '&RCPDSW2.+&SW(&I)'  APPEND THIS SWITCH
        AIF    ('&SW(&I) '(1,&K) EQ '&C').LOOP5  CHECK PREFIX
        MNOTE 4,'WARNING: SWITCH ''&SW(&I)'' NOT GENERICALLY EQUAL TO X
        SWITCH ''&SW(1)''
        AGO    .LOOP5
.ERROR  MNOTE 8,'SWITCH ''&SW(1)'' NOT DECLARED'
&RCPDSW1 SETC  ''                    INDICATE ERROR
.EXIT   MEND

```

IEB817I MEMBER NAME (RCPLOCS2) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPMCA
        MACRO
        RCPMCA &DSECT=YES
        GBLC  &RCPPRE
        GBLA  &RCPSWS(10)
        LCLC  &P
        RCPDEBUG
&P      SETC   '&RCPPRE'
        AIF    (&RCPSWS(2) NE 2).DSECT
&P.MCA  DS     0F          MODULE COMMUNICATIONS AREA
        AGO    .MCA2
.DSECT  ANOP
&P.MCA  DSECT          MODULE COMMUNICATIONS AREA
.MCA2   ANOP
&P.XDS  DS     F          ADDR OF EXTERNAL DUMMY SECTION
        AIF    (&RCPSWS(3) LT 1).EXIT
&P.A#GET DS     F          ADDRESS OF LIFO GET ROUTINE
&P.A#FRE DS     F          ADDRESS OF LIFO FREE ROUTINE
&P.#S   DS     F          ADDRESS OF CURRENT LIFO STACK
&P.#E   DS     F          ADDRESS OF END OF LIFO STACK
&P.#N   DS     F          ADDRESS OF NEXT FREE AREA
&P.#C   DS     F          ADDRESS OF NEXT LIFO STACK
&P.#L   DS     F          LENGTH OF CURRENT LIFO STACK
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPMCA) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPNTU
        MACRO
        RCPNTU &KEY,&LEN,&PAR

.*
.*      THIS IS AN ALLOC/FREE MACRO TEXT UNIT PROCESSOR SUBROUTINE
.*      MACRO. IT BUILDS NUMERIC TYPE TEXT UNITS.
.*
        LCLA   &L,&R
        LCLC   &C
        GBLC   &RCPTYPE
.*      ALLOC/FREE INNER MACRO TO SET UP NUMERIC TEXT UNITS
&L      SETA   1                                DEFAULT LENGTH
&L      AIF    ('&LEN' EQ ' ').NL
&L      SETA   &LEN
.&NL     MVI    S99TUKEY+1,&KEY                SET KEY FIELD
        MVI    S99TUNUM+1,1                    SET NUMBER FIELD
        MVI    S99TULNG+1,&L                    SET LENGTH FIELD
        AIF    ('&PAR'(1,1) EQ '(').REG
        RCPTYPE &PAR                            ANALYSE PARAMETER
&R      AIF    ('&RCPTYPE' EQ 'N').NUMERIC
        SETA   4-&L
        MVC    S99TUPAR(&L),&R+&PAR            MOVE IN QUANTITY
        RCPDINC 10
        MEXIT
.&NUMERIC AIF    (&L EQ 1).NL1
        MVC    S99TUPAR(&L.),=AL&L.(&PAR) MOVE IN QUANTITY
&R      SETA   &L+6
        AIF    (&R/2 EQ (&R+1)/2).LOK ENSURE LENGTH EVEN
&R      SETA   &R+1
.&LOK    RCPDINC &R
        MEXIT
.&NL1    MVI    S99TUPAR,&PAR                    MOVE IN QUANTITY
        RCPDINC 8
        MEXIT
.&REG    ANOP
&C      SETC   '&PAR'(2,K'&PAR-2)
        AIF    (&L EQ 3).STCM
        AIF    (&L EQ 2).STH
        AIF    (&L EQ 1).STC
        ST     &C,S99TUPAR                        STORE TEXT UNIT QUANTITY
        AGO    .RCPDINC
.&STH    STH    &C,S99TUPAR                        STORE TEXT UNIT QUANTITY
        AGO    .RCPDINC
.&STC    STC    &C,S99TUPAR                        STORE TEXT UNIT QUANTITY
        AGO    .RCPDINC
.&STCM   STCM   &C,7,S99TUPAR                    STORE TEXT UNIT QUANTITY
.&RCPDINC RCPDINC 10
        MEND

```

IEB817I MEMBER NAME (RCPNTU) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD  NAME=RCPPERM  
        MACRO  
        RCPPERM  
        SPACE
```

```
*****
```

```
**      PERMANENTLY ALLOCATED ATTRIBUTE TEXT UNIT      **
```

```
*****
```

```
        MVI  S99TUKEY+1,DALPERMA      SET TEXT UNIT KEY  
        RCPDINC 4  
        MEND
```

IEB817I MEMBER NAME (RCPPERM) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD      NAME=RCP PPL
        MACRO
&NAME  RCP PPL  &PCL= ,&NOPARM= ,&PARSERR= ,&PDLREG=R11 ,          X
        &PDLNAME= ,&PARSEP= ,&PARSWKA=
        GBLB   &RCP PPL(2) ,&RCPECT(2)
        GBLC   &RCP PRE ,&RCP PREP
&P     LCLC   &P
        SETC   '&RCP PRE '
        AIF    (&RCP PPL(2)) .BPPL
        EJECT
        IKJPPL
L#PPL  EQU     *-PPL          LENGTH OF PPL
&SYSECT CSECT
        SPACE 1
&RCP PPL(2) SETB 1
.BPPL  RCPDS
&P.PPL DS     CL(L#PPL)      RESERVE SPACE FOR PPL
&P.PDLP DS     F            POINTER TO PDL
        RCPDS
        SPACE 6
*****
***      THIS CODE GENERATES A PARSE PARAMETER LIST          ***
*****
        XC     &P.PDLP ,&P.PDLP          ZERO PDL POINTER
        AIF    ('&NOPARM(1)' EQ '' OR '&NOPARM(2)' NE '' ) .PB2
        L      R1 ,CPPLECT              LOAD ECT ADDRESS
&RCPECT(1) SETB 1
        USING  ECT ,R1                  ECT ADDRESSABLE
        TM     ECTSWS ,ECTNOPD           WERE ANY OPERANDS SUPPLIED?
        BO     &NOPARM(1)                NO, BRANCH OUT
        SPACE
.PB2   LA     R1 ,&P.PPL                LOAD PPL ADDRESS
        USING  PPL ,R1
        MVC    PPLUPT ,CPPLUPT           MOVE IN UPT ADDRESS
        MVC    PPLECT ,CPPLECT           MOVE IN ECT ADDRESS
        MVC    PPLCBUF ,CPPLCBUF         MOVE IN CBUF ADDRESS
        LA     R15 ,&P.ECB               LOAD ATTN ECB ADDRESS
        ST     R15 ,PPLECB               AND STORE IN PPL
        LA     R15 ,&P.PDLP              LOAD PDL POINTER ADDRESS
        ST     R15 ,PPLANS               AND STORE IN PPL
        AIF    ('&PARSWKA' EQ '' ) .PB3
        AIF    ('&PARSWKA'(1,1) EQ '' ) .PB4
        LA     R15 ,&PARSWKA             LOAD ADDRESS OF WORK AREA
        ST     R15 ,PPLUWA               AND STORE IN PPL
        AGO    .PB3
.PB4   ST     &PARSWKA(1) ,PPLUWA        STORE ADDRESS OF WORKAREA
.PB3   AIF    ('&PCL' EQ '' ) .EXIT
        L      R15 ,=V(&PCL)            LOAD PCL ADDRESS
        ST     R15 ,PPLPCL              AND STORE IN PPL
        SPACE 2
        AIF    ('&NOPARM(1)' EQ '' OR '&NOPARM(2)' EQ '' ) .PB5
        L      R1 ,CPPLECT              LOAD ECT ADDRESS

```

```

&RCPECT(1) SETB 1
  USING ECT,R1
  TM     ECTSWS,ECTNOPD          WERE ANY OPERANDS SUPPLIED?
  BO     &NOPARM(1)              NO, BRANCH OUT
  SPACE
.PB5    AIF ('&SYSPARM' EQ 'MVT').MVTBYP
        AIF ('&RCPPREP' EQ '').NOPREP
        L   R15,&RCPPREP          LOAD EP OF IKJPARS
        BALR R14,R15              AND ENTER IT
        AGO .PRET
.NOPREP ANOP
        L   R15,16                LOAD CVT ADDRESS
        TM  524(R15),X'80'        IS IKJPARS LOADED?
        AIF ('&PARSEP' EQ '').PBL1
        BZ  &P.LOAD               NO, BRANCH TO LOAD SVC
        L   R15,524(15)           LOAD EP OF IKJPARS
        ST  R15,&PARSEP           SAVE ITS ADDRESS
        BALR R14,R15              THEN BALR TO IT
        B   &P.PLNKB              BYPASS LOAD SVC
&P.LOAD LOAD EP=IKJPARS
        LR  R15,R0                LOAD EP OF IKJPARS
        ST  R15,&PARSEP           SAVE IT
        BALR R14,R15              THEN BALR TO IT
&P.PLNKB DS  0H
        AGO .PRET
.PBL1   BZ  &P.PLINK              NO, BRANCH TO LINK SVC
        L   R15,524(R15)          ELSE LOAD ITS ADDRESS
        BALR R14,R15              AND BALR TO IT
        B   &P.PLNKB              BYPASS LINK SVC
.MVTBYP ANOP
&P.PLINK LINK EP=IKJPARS
&P.PLNKB DS  0H
.PRET   AIF ('&PARSERR' EQ '').EXIT
        SPACE
        LTR R15,R15                TEST RETURN CODE
        BNZ &PARSERR              AND BRANCH ON NON-ZERO
        SPACE
        AIF ('&PDLREG' EQ '' OR '&PDLNAME' EQ '').EXIT
        L   &PDLREG,&P.PDLP        LOAD PDL ADDRESS
        USING &PDLNAME,&PDLREG    PDL DSECT ADDRESSABLE
.EXIT   MEND

```

IEB817I MEMBER NAME (RCPPPL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD      NAME=RCPPROC
        MACRO
        RCPPROC &WKCSECT=,&WKDSECT=,                X
                &REG1=,&REG0=,&ISA=,&SAVEPRE=,      X
                &SAVESUF=,&SP=
        GBLA    &RCPSWS(10)
        GBLC    &RCPPRE,&RCPWKCS,&RCPWKDS
        GBLC    &RCPSPN
        LCLC    &P,&C
        RCPDEBUG
&P      SETC    '&RCPPRE'
        AIF     ('&WKCSECT' EQ '').TDS
        SPACE
        MNOTE   4,'WKCSECT= OPTION INVALID WITH PROC OPTION, '
        MNOTE   *,'      WKDSECT=  USED INSTEAD'
&RCPWKDS SETC    '&WKCSECT'
        AGO     .SETCS
.TDS     AIF     ('&WKDSECT' EQ '').SYSECT
&RCPWKDS SETC    '&WKDSECT'
        AGO     .SETCS
.SYSECT  ANOP
&RCPWKDS SETC    '&SYSECT'
.SET1    AIF     (K'&RCPWKDS LT 8).LOK
&RCPWKDS SETC    '&RCPWKDS'(1,4) '&RCPWKDS'(6,3)'1'
        AGO     .SETCS
.LOK     ANOP
&RCPWKDS SETC    '&RCPWKDS.1'
.SETCS   ANOP
&RCPWKCS SETC    ''
&RCPSWS(4) SETA &RCPSWS(2)-1 SET W/A TO BE FREED OPT IF PROC(MAIN)
        AIF     ('&ISA' EQ '').NISA
&RCPSWS(3) SETA 1                SET LIFO FLAG IF ISA SPEC
.NISA    ANOP
        SPACE 2
        RCPDS
&P.RCODE DS     9D                SAVE AREA
        DS     F                  RETURN CODE
        RCPMCA
        RCPDS
        SPACE 2
        AIF     ('&REG1' EQ '').TR0
        LR     &REG1,R1          SAVE CONTENTS OF REG 1
.TR0     AIF     ('&REG0' EQ '').TP
        LR     &REG0,R0          SAVE CONTENTS OF REG 0
.TP      AIF     (&RCPSWS(2) EQ 2).PROCMN  PROCMAIN OPTION
        AIF     (&RCPSWS(3) EQ 1).PL      LIFO OPTION
        L      R15,0(R13)         R15 -> MODULE COMMUNIC. AREA
        L      R15,&P.XDS-&P.MCA(R15)  LOAD EXTERNAL DUMMY SECT ADDR
        AL     R15,&P.QCON         GET OFFSET TO WORK AREA
        ST     R15,8(R13)         CHAIN SAVE
        ST     R13,4(R15)         AREAS TOGETHER
        MVC    0(4,R15),0(R13)     COPY POINTER TO COMM AREA

```

```

LR      R13,R15          LOAD WORK AREA ADDRESS
USING  &RCPWKDS,R13    ESTABLISH ADDRESSABILITY TO IT
MEXIT

.PL     ANOP
*****
*      GET WORKAREA FROM LIFO STACK          *
*****
#GET   LV=&P.WKLEN
ST     R1,8(R13)        CHAIN SAVE
ST     R13,4(R1)        AREAS TOGETHER
MVC    0(4,R1),0(R13)   PROPAGATE MODULE COMM. AREA ADDR
LR     R13,R1          LOAD WORK AREA ADDRESS
USING  &RCPWKDS,R13    ESTABLISH ADDRESSABILITY TO IT
MEXIT

.PROCMN L      R0,&P.CXD      LOAD WORK AREA LENGTH
AIF    ('&SYSPARM' EQ 'MVT').MVT
MNOTE *, '      GETMAIN RU,LV=(0),SP=&SP,BNDRY=PAGE'
      GETMAIN RU,LV=(0),SP=&SP,BNDRY=PAGE
AGO    .CONT

.MVT   AIF    ('&SP' EQ '').NOSP
      ICM    R0,8,=AL1(&SP)    INSERT SUBPOOL NUMBER

.NOSP  ANOP
*      GETMAIN R,LV=(0)        OBTAIN A WORK AREA
.CONT  ANOP
&RCPSPN SETC  '&SP'
LR     R15,R13          SAVE CALLER'S SAVE AREA ADDR
LR     R13,R1           LOAD EXT DUMMY SECTION ADDR
AL     R13,&P.QCON      ADD OFFSET TO WORK AREA
ST     R13,8(R15)      CHAIN SAVE
ST     R15,4(R13)      AREAS TOGETHER
USING  &RCPWKDS,R13    GET WORKAREA ADDRESSABILITY
ST     R1,&P.XDS        STORE DUMMY SECTION ADDR IN      X
      MODULE COMMUNICATIONS AREA

LA     R15,&P.MCA        STORE COMMUNICATIONS AREA ADDR
ST     R15,0(R13)       IN WORD 1 OF SAVE AREA
AIF    (&RCPAWS(3) EQ 0 AND '&ISA' EQ '').EXIT
&RCPAWS(3) SETA 1        SET LIFO IN CASE ONLY ISA SPEC
&C     SETC  '&ISA'
AIF    ('&ISA' NE '').TK
&C     SETC  '8192'
AGO    .NK
.TK    AIF    ('&C'(K'&C,1) NE 'K').NK
&C     SETC  '&C'(1,K'&C-1)*1024'
.NK    EJECT
*****
**      INITIALIZE MODULE COMMUNICATIONS AREA WITH POINTERS          **
**      TO LIFO STACK AND LIFO GET/FREE ROUTINES                      **
*****
SPACE 1
MVC    &P.A#GET,=V(#####GET)  MOVE LIFO GET AND FREE
MVC    &P.A#FRE,=V(#####FREE)  ROUTINE ADDRESSES TO MCA
L      R15,=Q(#####ISA)        COMPUTE LIFO STACK

```

```

AL    R15,&P.XDS      PSEUDO REGISTER OFFSET
ST    R15,&P.#S      AND INITIALIZE POINTERS
ST    R15,&P.#N      IN MODULE COMMUNICATIONS AREA
L     R14,=A(&C)     LOAD SIZE OF INITIAL STACK AREA
ST    R14,&P.#L      STORE THIS IN MCA
ALR   R15,R14       THEN COMPUTE STACK END ADDRESS
ST    R15,&P.#E      AND STORE THIS INTO MCA
EJECT
*****
**      LIFO STACK GET/FREE ROUTINES      **
*****
      SPACE 1
#####ISA DXD    CL(&C)      DEFINE PSEUDO REGISTER FOR ISA
      SPACE 1
#####GET CSECT      LIFO GET ROUTINE
      USING *,R15
      USING &P.MCA,R1
      A     R0,&P.F7      ROUND LENGTH UP TO
      N     R0,&P.F8      A MULTIPLE OF 8
      AL    R0,&P.#N     COMPUTE NEXT FREE LIFO SLOT ADDR
      CL    R0,&P.#E     COMPARE TO STACK END ADDRESS
      BH    &P.GA       AND IF TOO BIG, BRANCH
      LR    R15,R1      PRESERVE MCA ADDRESS
      USING &P.MCA,R15  NEW BASE
      L     R1,&P.#N     LOAD ADDRESS OF SLOT
      ST    R0,&P.#N     AND STORE ADDRESS OF NEXT SLOT
      BR    R14        RETURN TO CALLER
&P.GA  EQU    *        IF CURRENT SLOT TOO SMALL
*      ABEND 1000,DUMP  ABEND FOR NOW
      ABEND 1000,DUMP
      SPACE 2
#####FREE DS     0H      LIFO FREE ROUTINE
      ENTRY #####FREE
      USING *,R15      BASE ADDRESS
      USING &P.MCA,R1  MCA ADDRESS
      CL    R0,&P.#S    CHECK THAT
      BL    &P.FA      ADDRESS TO BE
      CL    R0,&P.#E    FREED IS WITHIN
      BH    &P.FA      BOUND OF CURRENT STACK
      AL    R0,&P.F7    GET UPPER DOUBLE
      N     R0,&P.F8    WORD BOUNDARY
      ST    R0,&P.#N    AND UPDATE MCA
      BR    R14        RETURN TO CALLER
      SPACE 1
&P.FA  EQU    *        IF ADDRESS NOT WITHIN THIS STACK
*      ABEND 1001,DUMP  ABEND
      ABEND 1001,DUMP
      SPACE 2
&P.F7  DC     F'7'     CONSTANTS
&P.F8  DC     F'-8'    TO ROUND UP TO DOUBLEWORD SIZE
      DROP  R1,R15     KILL ADDRESSABILITY

```

&SYSECT CSECT RESUME MAIN PROGRAM CSECT
.EXIT MEND

IEB817I MEMBER NAME (RCPPROC) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD   NAME=RCPPSWD
        MACRO
        RCPPSWD &PASSW
        GBLC  &DYNP
        SPACE
*****
**      BUILD THE PASSWORD TEXT UNIT                               **
*****
        RCPVCHAR DALPASSW,14,&PASSW
        MEND
```

IEB817I MEMBER NAME (RCPPSWD) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD  NAME=RCPQNAME
        MACRO
        RCPQNAME &QNAME
        GBLC  &DYNP
        SPACE
*****
**      BUILD THE QNAME TEXT UNIT                               **
*****
        RCPVCHAR DALQNAME,14,&QNAME
        MEND
```

IEB817I MEMBER NAME (RCPQNAME) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPRNGE
        MACRO - BREAK A RANGE PARAMETER INTO TWO
        RCPRNGE &P
        GBLC  &RCPRNGE(2)
        LCLA  &I,&J,&K
&K      SETA  K'&P
&RCPRNGE(1) SETC  ''
&RCPRNGE(2) SETC  ''
.LOOP   ANOP
&I      SETA  &I+1
        AIF   (&I GT &K).NR
        AIF   ('&P'(&I,1) NE '-' AND '&P'(&I,1) NE ':').LOOP
&RCPRNGE(1) SETC '&P'(1,&I-1)
&RCPRNGE(2) SETC '&P'(&I+1,&K-&I)
        MEXIT
.NR     ANOP
&RCPRNGE(1) SETC '&P'
        MEND
```

IEB817I MEMBER NAME (RCPRNGE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD      NAME=RCPSPACE
        MACRO
        RCPSPACE &SPACE
        GBLA    &RCPSUB#          NO OF SUBLIST ELEMENTS
        GBLC    &RCPSUBL(100)     SUBLIST ELEMENTS
        *****
        *      THIS IS AN ALLOC INNER MACRO TO BUILD THE ALLOCATION SPACE
        *      QUANTITY TEXT UNIT. IT SHOULD BE SPECIFIED AS:-
        *      SPACE=(TYPE,(PRIMARY,SECONDARY,DIRECTORY),RLSE,CONTIG,ROUND)
        *      WHERE TYPE IS 'TRK', 'CYL', 'ABSTR' OR A BLOCK QUANTITY
        *      'CYL' OR 'TRK' SHOULD NOT BE ENTERED IN QUOTES. THE BLOCK
        *      QUANTITY CAN BE A NUMBER, A REGISTER (IN BRACKETS), OR THE
        *      NAME OF A FULLWORD CONTAINING THE BLOCK SIZE.
        *****
        AIF     ('&SPACE(1)' EQ '' OR '&SPACE(1)' EQ 'TRK').TRK
        AIF     ('&SPACE(1)' EQ 'CYL').CYL
        *****
        **      SPACE UNIT IN BLOCKS      **
        *****
        RCPNTU DALBLKLN,3,&SPACE(1) GENERATE BLOCK UNIT TU
        AGO     .TPRIME          GO TEST PRIME QUANTITY
        .TRK   ANOP TRACK SPEC REQ OR DEFAULTED
        SPACE
        *****
        **      SPACE QUANTITY IN TRACKS  **
        *****
        MVI     S99TUKEY+1,DALTRK      SET TEXT UNIT KEY
        RCPDINC 4
        AGO     .TPRIME
        .CYL   ANOP CYL QUANTITY
        SPACE 1
        *****
        **      SPACE UNIT IN CYLINDERS   **
        *****
        MVI     S99TUKEY+1,DALCYL      SET TEXT UNIT KEY
        RCPDINC 4                      STORE TEXT UNIT ADDR
        .TPRIME RCPSUBL &SPACE(2)      BREAK UP SUBLIST
        AIF     (&RCPSUB# EQ 0).TCONTIG
        AIF     ('&RCPSUBL(1)' EQ '').TSP2
        SPACE
        *****
        **      PRIMARY SPACE QUANTITY    **
        *****
        RCPNTU DALPRIME,3,&RCPSUBL(1)
        .TSP2  AIF     (&RCPSUB# LT 2).TCONTIG
        AIF     ('&RCPSUBL(2)' EQ '').TSP3
        SPACE
        *****
        **      SECONDARY SPACE QUANTITY  **
        *****
        RCPNTU DALSECND,3,&RCPSUBL(2)
        .TSP3  AIF     (&RCPSUB# LT 3).TCONTIG
```

```
      AIF ('&RCPSUBL(3)' EQ ' ').TCONTIG
      SPACE
*****
**      DIRECTORY BLOCK QUANTITY                               **
*****
      RCPNTU DALDIR,3,&RCPSUBL(3)
.TCONTIG AIF ('&SPACE(3)' EQ 'CONTIG' OR '&SPACE(4)' EQ 'CONTIG').CON
      AIF ('&SPACE(3)' EQ 'MXIG' OR '&SPACE(4)' EQ 'MXIG').MXIG
      AIF ('&SPACE(3)' EQ 'ALX' OR '&SPACE(4)' EQ 'ALX').ALX
.TRLSE  AIF ('&SPACE(3)' EQ 'RLSE' OR '&SPACE(4)' EQ 'RLSE').RLSE
.TROUND AIF ('&SPACE(4)' EQ 'ROUND' OR '&SPACE(5)' EQ 'ROUND').ROUND
      MEXIT
.CON    ANOP
*****
**      CONTIGUOUS SPACE TEXT UNIT                               **
*****
      RCPNTU DALSPFRM,1,8
      AGO .TRLSE
.MXIG   ANOP
*****
**      MAXIMUM CONTIGUOUS SPACE TEXT UNIT                       **
*****
      RCPNTU DALSPFRM,1,4
      AGO .TRLSE
.ALX    ANOP
*****
**      'ALX' SPACE TEXT UNIT                                    **
*****
      RCPNTU DALSPFRM,1,2
      AGO .TRLSE
.RLSE   ANOP
*****
**      RELEASE UNUSED SPACE TEXT UNIT                           **
*****
      MVI  S99TUKEY+1,DALRLSE      SET TEXT UNIT KEY
      RCPDINC 4
      AGO .TROUND
.ROUND  ANOP
*****
**      RELEASE UNUSED SPACE TEXT UNIT                           **
*****
      MVI  S99TUKEY+1,DALROUND     MOVE IN TEXT UNIT KEY
      RCPDINC 4
      MEND
```

IEB817I MEMBER NAME (RCPSPACE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPSPEC
        MACRO - SET UP USER DEFINED TEXT UNIT
        RCPSPEC &T
        LCLA  &I,&J
&I      SETA  1
&J      SETA  K'&T
        SPACE
*****
**      PROCESS SPECIAL TEXT UNITS          **
*****
.LOOP   RCPVCHAR &T(&I),&T(&I+2),&T(&I+3),N=&T(&I+1)
&I     SETA  &I+4
        AIF  (&I LE &J).LOOP
        MEND
```

IEB817I MEMBER NAME (RCPSPEC) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPSR2
        MACRO
        RCPSR2 &A
        GBLB  &RCPSR2
        GBLC  &DYNP
        LCLC  &C
.*      TO SAVE REG 2 IN REG 0 FOR ALLOC INNER MACROS FIRST TIME ONLY
.*      IF OPERAND SUPPLIED AND SAVE DONE, RESTORES REG 2 AND
.*      GENERATES MOVE INSTRUCTION FOR EXECUTE
        AIF   ('&A' NE '').UNSAVE
        AIF   (&RCPSR2).EXIT
&RCPSR2 SETB  1
        LR   R0,R2          SAVE CONTENTS OF REGISTER 2
        MEXIT
.UNSAVE AIF   (NOT &RCPSR2).EXIT
        B    *+10          SKIP NEXT INSTRUCTION
&C      SETC  '&DYNP.MVC'
&C      MVC   S99TUPAR(0),0(R14) EXECUTED MOVE
        LR   R2,R0          RESTORE CONTENTS OF REGISTER 2
&RCPSR2 SETB  0
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPSR2) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD  NAME=RCPSSREQ
        MACRO
        RCPSSREQ
        SPACE 1
*****
**      SUBSYSTEM REQUEST TEXT UNIT          **
*****
        SPACE 1
        MVI  S99TUKEY+1,DALSSREQ MOVE IN TEXT UNIT KEY
        RCPDINC          4
        MEND
```

IEB817I MEMBER NAME (RCPSSREQ) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD      NAME=RCPSUBL
        MACRO -  BREAK DOWN A SUBLIST
        RCPSUBL &L
        GBLA    &RCPSUB#          NO OF ELEMENTS FOUND
        GBLC    &RCPSUBL(100)     ELEMENTS
        LCLA    &I,&J,&K
&RCPSUB# SETA  0                  INITIALIZE
        AIF     ('&L' EQ '').EXIT  EXIT IF NULL STRING
        AIF     ('&L'(1,1) NE '(').NOSUB
&K      SETA   K'&L-1
&I      SETA   2
&J      SETA   1
.LOOP   ANOP
&J      SETA   &J+1
        AIF     (&J GT &K).LAST
        AIF     ('&L'(&J,1) NE ',').LOOP
&RCPSUB# SETA &RCPSUB#+1
        AIF     (&J EQ &I).NULL
&RCPSUBL(&RCPSUB#) SETC '&L'(&I,&J-&I)
&I      SETA   &J+1
        AGO     .LOOP
.NULL   ANOP
&RCPSUBL(&RCPSUB#) SETC ''
&I      SETA   &J+1
        AGO     .LOOP
.LAST   AIF     (&J EQ &I).LASTNUL
&RCPSUB# SETA &RCPSUB#+1
&RCPSUBL(&RCPSUB#) SETC '&L'(&I,&J-&I)
        AGO     .EXIT
.LASTNUL ANOP
&RCPSUB# SETA &RCPSUB#+1
&RCPSUBL(&RCPSUB#) SETC ''
        AGO     .EXIT
.NOSUB  ANOP
&RCPSUBL(1) SETC '&L'
&RCPSUB# SETA 1
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPSUBL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```

./      ADD      NAME=RCPSYSOU
        MACRO
        RCPSYSOU &CLASS,&COPIES=,&FREE=,&DEST=,&FORMS=
        GBLC    &DYNP
        LCLC    &C
        AIF     ('&CLASS(1)' EQ ' ').TPGN
&C      SETC    '&CLASS(1)'
        SPACE
*****
**      SYSOUT CLASS TEXT UNIT                               **
*****
        AIF     ('&C'(1,1) EQ ' ').Q
        AIF     ('&C'(K'&C,1) EQ '/ ').BS
        AIF     ('&C'(1,1) EQ '(' ).REG
        L       R14,&C          LOAD ADDRESS OF SYSOUT CLASS
        MVC     S99TUPAR(1),0(R14)  AND MOVE IT TO TEXT UNIT
        AGO     .SKEY
        .REG    MVC     S99TUPAR(1),0&C      MOVE SYSOUT CLASS TO TEXT UNIT
        .SKEY   MVI     S99TUKEY+1,DALSYSOU  SET SYSOUT KEY
        MVI     S99TUNUM+1,1          SET NUMBER FIELD
        MVI     S99TULNG+1,1          SET LENGTH FIELD
        RCPDINC 8
        AGO     .TPGN
        .BS     RCPTUBFR DALSYSOU,14,&C
        AGO     .TPGN
        .Q      RCPBTU DALSYSOU,1,&C
        .TPGN   AIF     ('&CLASS(2)' EQ ' ').TCOP
        SPACE
*****
**      SYSOUT PROGRAM NAME TEXT UNIT                       **
*****
&C      SETC    '&CLASS(2)'
        RCPVCHAR DALSPGM,14,&C
        .TCOP   AIF     ('&COPIES' EQ ' ').TFREE
        SPACE
*****
**      SYSOUT COPIES TEXT UNIT                             **
*****
        RCPNTU DALCOPYS,1,&COPIES
        .TFREE  AIF     ('&FREE' EQ ' ').TDEST
        SPACE
*****
**      FREE = CLOSE TEXT UNIT                               **
*****
        AIF     ('&FREE' EQ 'CLOSE').CLOSEOK
        MNOTE 4,' **** FREE=&FREE INVALID, FREE=CLOSE USED'
        .CLOSEOK MVI     S99TUKEY+1,DALCLOSE  MOVE IN TEXT UNIT KEY
        RCPDINC 4
        .TDEST  AIF     ('&DEST' EQ ' ').TFORMS
        SPACE
*****
**      SYSOUT DESTINATION TEXT UNIT                         **
*****

```

```
*****  
      RCPVCHAR DALUSER,14,&DEST  
.TFORMS AIF ('&FORMS' EQ ' ').EXIT  
      SPACE  
*****  
**      SYSOUT FORMS NUMBER TEXT UNIT      **  
*****  
      RCPVCHAR DALFMNO,14,&FORMS  
.EXIT      MEND
```

IEB817I MEMBER NAME (RCPSYSOU) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPTU
        MACRO
        RCPTU &TU          TEXT UNIT LIST
        GBLA  &DTUPO      TEXT UNIT POINTER OFFSET
        GBLC  &DYNP       ALLOC SYMBOL PREFIX
        LCLA  &I,&J
        LCLC  &C
        SPACE 1
*****
**      ADD SPECIAL TEXT UNITS      **
*****
&J      SETA  N'&SYSLIST
.LOOP   ANOP
&I      SETA  &I+1
        AIF   (&I GT &J).EXIT
        AIF   ('&TU(&I)''(1,1) EQ ''').R
        LA    R15,&TU(&I)          LOAD TEXT UNIT ADDRESS
        ST    R15,&DYNP.TUP+&DTUPO  AND STORE IT IN POINTER LIST
&DTUPO  SETA  &DTUPO+4
        AGO   .LOOP
.R      ANOP
&C      SETC  '&TU(&I)''(2,K'&TU(&I)-2)
        ST    &C,&DYNP.TUP+&DTUPO  STORE TEXT UNIT ADDR IN PTR LIST
&DTUPO  SETA  &DTUPO+4
        AGO   .LOOP
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPTU) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPTUBFR
        MACRO  - BUILD TEXT UNIT FROM BUFFER
        RCPTUBFR &KEY,          TEXT UNIT KEY          X
                                &L,                  MAXIMUM LENGTH VALUE      X
                                &C,                  TEXT UNIT              X
                                &N=1                TEXT UNIT NUMBER
        GBLC  &EXECNAM
        LCLC  &C1,&C2
        LCLA  &I,&K
        MVI   S99TUKEY+1,&KEY    SET TEXT UNIT KEY
        AIF   ('&N' EQ ' ' OR '&N' EQ '1').N1
        LA    R14,&N             LOAD TEXT UNIT NUMBER
        STH   R14,S99TUNUM      AND STORE INTO TEXT UNIT
        AGO   .ENDN
.N1     MVI   S99TUNUM+1,1      SET TEXT UNIT NUMBER
.ENDN   ANOP
&K     SETA  K'&C
&I     SETA  &K-1
.LOOP1 ANOP
&K     SETA  &K-1
        AIF   (&K LE 0).STD
        AIF   ('&C'(&K,1) NE '/').LOOP1
&C2    SETC  '&C'(&K+1,&I-&K)
&C1    SETC  '&C'(1,&K-1)
        AIF   ('&C1'(1,1) NE '(').TC2
&C1    SETC  '0&C1'
.TC2   AIF   ('&C2' EQ '0000').V2B
        AIF   ('&C2' EQ '00').V1B
        AIF   ('&C2' EQ '0').V0B
        AIF   ('&C2'(1,1) EQ '(').RL
        MVI   S99TULNG+1,&C2    SET LENGTH FIELD
        MVC   S99TUPAR(&C2.),&C1 MOVE IN TEXT UNIT
        RCPDINC &L
        MEXIT
.STD   ANOP
&K     SETA  &L-6
&C1    MVI   S99TULNG+1,&K      SET TEXT UNIT LENGTH
        SETC  '&C'(1,&I)        REMOVE TRAILING SLASH
        MVC   S99TUPAR(&K),&C1  MOVE IN TEXT UNIT
        RCPDINC &L
        MEXIT
.V2B   LH    R14,&C1           LOAD TEXT UNIT LENGTH
        S     R14,=A(4)        EXCLUDE LENGTH OF HEADER
&C1    SETC  '4+&C1'
        AGO   .MOVE
.V1B   LH    R14,&C1           LOAD TEXT UNIT LENGTH
&C1    SETC  '2+&C1'
        AGO   .MOVE
.V0B   SLR   R14,R14          CLEAR FOR IC
&C1    IC    R14,&C1          INSERT TEXT UNIT LENGTH
&C1    SETC  '1+&C1'
        AGO   .MOVE

```

```
.RL      ANOP
&C2     SETC  '&C2'(2,K'&C2-2)
        LR   R14,&C2          LOAD TEXT UNIT LENGTH
.MOVE   STH  R14,S99TULNG     AND STORE INTO LENGTH FIELD
        BCTR R14,0           GET MACHINE LENGTH
        EXECUTE ,MVC,S99TUPAR-S99TUNIT(0,R15),&C1
        EX   R14,&EXECNAM     MOVE IN TEXT UNIT
        RCPDINC &L
        MEND
```

IEB817I MEMBER NAME (RCPTUBFR) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPTXTL
        MACRO - TO COUNT CHARACTERS IN A STRING
        RCPTXTL &S
        GBLA  &RCPTXTL
        LCLA  &I,&K,&L
&RCPTXTL SETA  0
        AIF   (K'&S LT 3).MEND
&RCPTXTL SETA  K'&S-2
&L       SETA  &RCPTXTL
&I       SETA  1
.LOOP    ANOP
&I       SETA  &I+1
.LOOP2   AIF   (&I GT &L).MEND
        AIF   ('&S'(&I,2) NE '' AND '&S'(&I,2) NE '&&').LOOP
&I       SETA  &I+2
&RCPTXTL SETA  &RCPTXTL-1
        AGO   .LOOP2
.MEND    MEND
```

IEB817I MEMBER NAME (RCPTXTL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPTYPE
        MACRO
        RCPTYPE &T
        GBLC  &RCPTYPE
        LCLA  &I,&K
&K      SETA  K'&T
&RCPTYPE SETC  ''
        AIF   (&K EQ 0).EXIT
&RCPTYPE SETC  'C'
        .LOOP ANOP
&I      SETA  &I+1
        AIF   ('&T'(&I,1) LT '0' OR '&T'(&I,1) GT '9').EXIT
        AIF   (&I LT &K).LOOP
&RCPTYPE SETC  'N'
        .EXIT MEND
```

IEB817I MEMBER NAME (RCPTYPE) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPUNALC  
        MACRO  
        RCPUNALC  
        SPACE 1
```

```
*****
```

```
**      FREE EVEN IF PERMANENTLY ALLOCATED      **
```

```
*****
```

```
        MVI    S99TUKEY+1,DUNUNALC      SET TEXT UNIT KEY  
        RCPDINC 4  
        MEND
```

IEB817I MEMBER NAME (RCPUNALC) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD    NAME=RCPUNIT
        MACRO
        RCPUNIT &U,&V
        GBLC   &DYNP
        AIF    ('&U' EQ '').TVOL
        SPACE 1
*****
**      UNIT NAME TEXT UNIT                               **
*****
        RCPVCHAR DALUNIT,14,&U
.TVOL   AIF    ('&V' EQ '').EXIT
        SPACE 1
*****
**      VOLUME SERIAL TEXT UNIT                           **
*****
        RCPVCHAR DALVLSER,14,&V
.EXIT   MEND
```

IEB817I MEMBER NAME (RCPUNIT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=RCPVCHAR
        MACRO
        RCPVCHAR &KEY,&LEN,&C,&N=1
        GBLC   &DYNP
        AIF    ('&C'(K'&C,1) EQ '/').BM
        AIF    ('&C'(1,1) EQ ''').QM
        RCPSR2
        AIF    ('&C'(1,1) EQ '(').RM
        LH     R2,&C+4          LOAD LENGTH OF TEXT UNIT
        LTR    R2,R2           TEST FOR ZERO
        BZ     *+30           IF NO TEXT UNIT, SKIP
        L      R14,&C         LOAD ADDRESS OF TEXT UNIT
        AGO    .STHM
        .RM
        LH     R2,4&C         LOAD LENGTH OF TEXT UNIT
        LTR    R2,R2         AND TEST FOR ZERO
        BZ     *+30           IF NO TEXT UNIT, SKIP
        L      R14,0&C       LOAD ADDRESS OF TEXT UNIT
        .STHM
        STH    R2,S99TULNG    STORE LENGTH OF TEXT UNIT
        BCTR   R2,0           DECREMENT FOR EXECUTE
        EX     R2,&DYNP.MVC   MOVE IN TEXT UNIT
        MVI    S99TUKEY+1,&KEY MOVE IN TEXT UNIT KEY
        AIF    ('&N' EQ '1' OR '&N' EQ '').N1
        LA     R14,&N        LOAD TEXT UNIT NUMBER
        STH    R14,S99TUNUM   AND STORE IT IN TEXT UNIT
        AGO    .ENDN
        .N1
        MVI    S99TUNUM+1,1   SET NUMBER FIELD
        .ENDN
        RCPDINC &LEN
        MEXIT
        .BM
        RCPTUBFR &KEY,&LEN,&C
        MEXIT
        .QM
        RCPBTU &KEY,&N,&C
        MEND

```

IEB817I MEMBER NAME (RCPVCHAR) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD    NAME=RCPVOLRT
        MACRO
        RCPVOLRT
        SPACE 1
*****
**      VOLUME SERIAL RETURN TEXT UNIT      **
*****
        MVI    S99TUKEY+1,DALRTVOL          SET RETURN VOLUME SERIAL KEY
        MVI    S99TUNUM+1,1                 SET NUMBER FIELD
        MVI    S99TULNG+1,8                 SET LENGTH FIELD
        MVC    S99TUPAR(8),=CL8' '         INITIALIZE FIELD TO BLANKS
        RCPDINC 14
        MEND
```

IEB817I MEMBER NAME (RCPVOLRT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD  NAME=REGS
R0      EQU  0      *USED BY O.S.
R1      EQU  1      *USED BY O.S. // ADDRESS OF PARAMETER LIST
R2      EQU  2
R3      EQU  3
R4      EQU  4
R5      EQU  5
R6      EQU  6
R7      EQU  7
R8      EQU  8
R9      EQU  9
R10     EQU 10
R11     EQU 11
R12     EQU 12
R13     EQU 13      *USED BY O.S. // SAVE-AREA ADDRESS
R14     EQU 14      *USED BY O.S. // RETURN ADDRESS
R15     EQU 15      *USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
IEB817I MEMBER NAME (REGS ) NOT FOUND IN NM DIRECTORY.  STOWED WITH TTR.
```

```

./      ADD     NAME=S99FAIL
        MACRO
&NAME  S99FAIL &RB=(R14),&RC=(R15),&CPPL=,&MF=G,&CP=
        GBLB   &RCPCPPL(2)          CP INDICATOR
        GBLC   &RCPPRE
        LCLB   &GEN
&NAME  LCLC   &C
        DS     0H
        AIF    ('&MF(1)' EQ 'G').GEN
        AIF    ('&MF(1)' EQ 'E').EXEC
        MNOTE  4,'&MF(1) IS AN INVALID MF, MF=G USED'
        .GEN   LA     R1,FAIL&SYSNDX    LOAD PLIST ADDRESS
&GEN   SETB   1
        .EXEC  AGO    .L
        AIF    ('&MF(2)' NE '').LISTOK
        MNOTE  8,'LIST ADDRESS NOT SPECIFIED'
        MEXIT
        .LISTOK AIF   ('&MF(3)' EQ '').TMF2
&MF(3) EQU    24          LENGTH OF PARAMETER LIST
        .TMF2  AIF   ('&MF(2)' EQ '(R1)' OR '&MF(2)' EQ '(1)').L
        AIF   ('&MF(2)'(1,1) EQ ' ').REG
        LA    R1,&MF(2)      LOAD DAIRFAIL PARAM LIST ADDRESS
        .REG   AGO    .L
&C     SETC   '&MF(2)'(2,K'&MF(2)-2)
        LR    R1,&C          LOAD DAIRFAIL PARAM LIST ADDR
        .L     AIF   ('&RB'(1,1) EQ ' ').RBR
        AIF   ('&RB' NE '').RBA
        MNOTE  8,'REQ BLOCK ADDRESS NOT SPECIFIED'
        MEXIT
        .RBR   ST    &RB(1),0(R1)      STORE S99 RB ADDRESS
        .RBA   AGO    .RC
        .RC    LA    R14,&RB          LOAD ADDRESS OF REQ BLOCK
        ST    R14,0(R1)          AND STORE IN PLIST
        .RC    AIF   ('&RC'(1,1) EQ ' ').RCR
        LA    R14,&RC          LOAD ADDRESS OF RET CODE
        ST    R14,4(R1)          AND STORE IN PLIST
        .RCR   AGO    .EFF02
        .GRC   ANOP
        .EFF02 LA    R14,20(R1)       LOAD ADDR RET CODE FLD
        ST    &RC(1),0(R14)       STORE RET CODE
        ST    R14,4(R1)          AND STORE ITS ADDRESS
        .EFF02 LA    R14,-A(0)       LOAD ADDR OF FULLWORD OF 0
        ST    R14,8(R1)          STORE IT.
        AIF   ('&CP' EQ 'YES' OR &RCPCPPL(1)).CPID
        LA    R14,=X'8032'       LOAD ADDRESS OF CALLERID
        ST    R14,12(R1)         AND STORE IT
        XC    16(4,R1),16(R1)     CLEAR CPPL POINTER
        .CPID  AGO    .GO
        LA    R14,=Y(50)         LOAD ADDRESS OF CALLERID
        ST    R14,12(R1)         AND STORE IT
        AIF   ('&CPPL' EQ '').DCPPL

```

```
AIF ('&CPPL'(1,1) EQ '(').RCPPL
LA R14,&CPPL LOAD CPPL ADDRESS
ST R14,16(R1) AND STORE IT
AGO .GO
.DCPPL MVC 16(4,R1),&RCPPRE.CPPL MOVE IN CPPL ADDRESS
AGO .GO
.RCPPL ST &CPPL(1),16(R1) STORE ADDRESS OF CPPL
.GO LINK EP=IKJEFF18
AIF (NOT &GEN).EXIT
SPACE 1
RCPDS
&C SETC 'FAIL&SYSNDX'
&C DS 6F RESERVE SPACE FOR PARAM LIST
RCPDS
.EXIT MEND
```

IEB817I MEMBER NAME (S99FAIL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD  NAME=VTCALL
        MACRO
&LAB   VTCALL &RTN,&TEST
&LAB   LA    R1,VTOCOM      POINT TO THE COMMON AREA
        L    R15,VAD&RTN    POINT TO THE ROUTINE
        AIF ('&TEST' NE 'TEST').NOTEST
        LTR  R15,R15        SEE IF THE ROUTINE IS PRESENT
        BZ   *+6            DON'T CALL IT IF IT'S NOT THERE
        .NOTEST ANOP
        BALR R14,R15        THEN CALL IT
        MEND
```

IEB817I MEMBER NAME (VTCALL) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=VTFMT
        MACRO
        VTFMT

*
*      THIS DSECT DESCRIBES THE FORMATTED DSCB
*
VTFMT   DSECT
VTFNEXT DS    A          POINTER TO NEXT DSCB
VTFALLOC DS    F          ALLOCATION IN UNITS AS DEFINED BY THE
*                          COMMAND.  KBYTES, MBYTES, TRACKS, OR
*                          CYLS MAY BE THE UNIT.
VTFUSED  DS    F          AMOUNT USED, SAME UNIT
VTFUNUSD DS    F          AMOUNT UNUSED, SAME UNIT
VTFPCT   DS    H          PERCENT USED,  0-100
VTFVOLUM DS    CL6       VOLUME SERIAL NUMBER
VTFUNIT  DS    CL4       UCB UNIT TYPE                RPRINS
VTFCREDT DS    XL3       CREATION DATE YYDDD
VTFEXPDT DS    XL3       EXPIRATION DATE YYDDD
VTFNSTAC DS    XL3       LAST ACCESS DATE YYDDD
VTFNOEPV DS    AL1       NUMBER OF EXTENTS PER VOLUME
VTFDSORG DS    CL3       DATA SET ORGANIZATION
*                          PS, PO, DA, VS, IS, PERHAPS U
VTFRECFM DS    CL5       RECORD FORMAT
*                          F,V, OR U, B, T, S, A, M
VTFLRECL DS    H          LOGICAL RECORD LENTGH
VTFBLKSZ DS    H          BLOCK SIZE
VTFROUND DS    C          R IF ROUND WAS SPECIFIED
VTFPROT  DS    C          PASSWORD PROTECTION FLAG
VTFRACF  DS    C          RACF INDICATOR FLAG                TANSKY
VTFUPD   DS    C          UPDATE FLAG (UPDATED SINCE BACKUP) TANSKY
VTFCATLG DS    C          CATALOG INDICATION
VTFSECAM DS    XL2       SECONDARY AMOUNT
VTFSECAL DS    C          SECONDARY ALLOCATION TYPE
*                          C FOR CYL, T FOR TRKS, B FOR BLOCKS
*                          R FOR BLOCKS WITH ROUND
VTFDSTYP DS    C          DATA SET TYPE, USER MAY DEFINE
*                          S = SYSTEM TEMPORARY DATA SET
*                          T = TEST DATA SET
*                          P = PRODUCTION DATA SET
VTFACON  DS    CL8       REQUESTED ACTION OR COMMENT
VTFDSNL  DS    H          LENGTH OF DSNAME
VTFMTL   EQU    *-VTFMT  FIXED LENGTH OF THIS DSECT
VTFDSN   DS    44C       VARIABLE LENGTH FIELD
MEND

```

IEB817I MEMBER NAME (VTFMT) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.


```
./      ADD  NAME=VTOCMSG
        MACRO
&LAB   VTOCMSG &MSG1,&MSG2      FIRST LEVEL MESSAGE, OPTIONAL SECOND
&LAB   LA    R1,&MSG1          POINT TO THE FIRST MESSAGE
        AIF  ('&MSG2' EQ ' ').NOSEC  IF NO SECOND LEVEL MSG
        LA    R0,&MSG2          POINT TO THE SECOND MESSAGE
        AGO  .SETMSG           SET UP THE MESSAGES
.NOSEC  SR    R0,R0            NO SECOND LEVEL MESSAGE
.SETMSG STM  R0,R1,MSGADDRS    SAVE THE MESSAGE ADDRESSES
*       THEN JUST CALL THE MESSAGE ISSUING ROUTINE
        VTCALL MSG             AWAY WE GO
        MEND
```

IEB817I MEMBER NAME (VTOCMSG) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```

./      ADD    NAME=VTOCOM
        MACRO
        VTOCOM  &TYPE

*
*      THIS IS THE VTOC COMMAND COMMON AREA
*
VTOCOM  AIF    ('&TYPE' EQ 'NODSECT').NODSECT
        DSECT
        AGO    .NODS
.NODSECT ANOP
VTOCOM  DS     0D
.NODS   ANOP
*
*      WORKING STORAGE AREAS FOR THE VARIOUS ROUTINES
*
VTCWMSG DS     A           WORKING STORAGE FOR THE MSG  ROUTINE
VTCWEXIT DS     A           WORKING STORAGE FOR THE EXIT ROUTINE
VTCWEXCP DS     A           WORKING STORAGE FOR THE EXCP ROUTINE
VTCWCHEK DS     A           WORKING STORAGE FOR THE CHEK ROUTINE
VTCWFORM DS     A           WORKING STORAGE FOR THE FORM ROUTINE
VTCWPRNT DS     A           WORKING STORAGE FOR THE PRNT ROUTINE
VTCWSORT DS     A           WORKING STORAGE FOR THE SORT ROUTINE
*
*      ADDRESSES OF THE ROUTINES
*
VADMSG  DC     V(VTOCMSG)   ADDRESS OF THE MESSAGE ROUTINE
VADEXIT DC     V(VTOCEXIT)  ADDRESS OF THE EXIT ROUTINE
VADEXCP DC     V(VTOCEXIT)  ADDRESS OF THE EXCP ROUTINE
VADCHEK DC     V(VTOCEXIT)  ADDRESS OF THE CHECK ROUTINE
VADFORM DC     V(VTOCEXIT)  ADDRESS OF THE FORMAT ROUTINE
VADPRNT DC     V(VTOCEXIT)  ADDRESS OF THE PRINT ROUTINE
VADSORT DC     V(VTOCEXIT)  ADDRESS OF THE SORT ROUTINE
*
*      TSO COMMAND PROCESSOR AND PARSE DATA
*
ADDRUPT DS     A           USER PROFILE TABLE
ADDRRECT DS     A           ENVIRONMENT CONTROL TABLE
ADDRPSCB DS     A           PROTECTED STEP CONTROL BLOCK
ADDRCBUF DS     A           COMMAND BUFFER
        SPACE
ADDRANSR DS     A           PARSE ANSWER OR PDL ADDRESS
        SPACE
PARMLIST DS     8A          INTERNAL PARM AREA ( MSG )
ATTNECB DS     F           ECB FOR ATTENTIONS
DOUBLE  DS     D
        SPACE
MSGADDRS DS     2A          ADDRESSES OF MESSAGES FOR VTOCMSG
MSGTEXT1 DS     XL124
MSGTEXT2 DS     XL124
        SPACE 3
PUTOLD1 DS     3F
PUTOLD2 DS     3F

```

*
* PARAMETER LIST FOR THE EXIT ROUTINE
*

EXITLIST	DS	0F	
EXITAREA	DS	A	WORKAREA LOCATION
DSCBADDR	DS	A	ADDRESS OF THE DSCB
FORMATAD	DS	A	ADDRESS OF THE FORMATTED DSCB
CPPLADDR	DS	A	ADDRESS OF THE CPPL
ACTIONAD	DS	A	ADDRESS OF THE RECOMMENDED OR REQUESTED ACTION

*
* INTER ROUTINE FLAGS
*

VTCEFUNC	DS	X	VTOCEXCP FUNCTION FLAG
VTCFMTCK	DS	X	FORMAT IS CALLED BY CHECK RTN
VTCFMTCD	EQU	X'80'	FORMAT WAS CALLED BY CHECK
VTCFMTCC	EQU	X'08'	FORMAT WAS CALLED BY CHECK THIS CALL

TABFULL	DS	X	FLAG TABLES FULL, STOP INPUT
LOCAT	DS	X	FLAG TO PERFORM CATALOG LOCATE
VTCEPRNT	DS	X	PRINT END AND CLEANUP FLAG
DSNLEN	DS	H	LENGTH OF THE DSNAME (NON-BLANKS)
ATABTITL	DS	A	ADDRESS OF TABLE OF TITLES, LENGTHS

*
*
* WORKING STORAGE FOR VOLUME UCB SEARCH
*

ADDR	DS	CL3	UCB ADDRESS IN CHARACTERS
VOLSER	DS	CL6	VOLUME SERIAL NUMBER FROM PARSE
VOLID	DS	CL6	CURRENT VOLUME SERIAL NUMBER TO PROCESS
FLAG	DS	X	UCB SEARCH FLAG
LASTADR	DS	F	LAST UCB ADDRESS FOUND (NO DUP'S)
UCBDEVT	DS	CL4	PRINTABLE FORM OF DEVICE TYPE RPRINS

*
*
*
* SORTTAB DS 16F
*
* EACH ENTRY CONTAINS A KEY OFFSET (2 BYTES) AND A KEY LENGTH (2 BYTES)
* THIS TABLE IS BUILT AT PARSE TIME ACCORDING TO THE SORT PARAMETERS
* SPECIFIED. THE 1ST PARM IS THE HIGH KEY AND SO ON.
*

*
*
* ADDRESSES OF GETMAIN FOR FORMATTED DATA
*

VTCCURAD	DS	A	CURRENT AVAILABLE ADDRESS
VTCCURLN	DS	A	CURRENT AVAILABLE LENGTH
VTCGETMN	DS	50A	ADDRESSES OF BLOCKS
VTCGETMX	EQU	(*-VTCGETMN)/4	NUMBER OF BLOCKS MAXIMUM
VTCGETMS	EQU	32768	GETMAIN SIZE

*

```
*      HASH SORT TABLE, POINTERS TO FIRST ENTRIES
*
VTCSORTH DS    256A          POINT TO FORMATED ENTRIES
VTCSORTE EQU   *            END OF LIST
*
*      PRINT ENTRIES - PAGE AND LINE COUNTERS
*
LINECT  DS    H            LINE COUNT
LINEMAX DS    H            MAXIMUM LINES PER PAGE
PAGECT  DS    H            PAGE COUNT
LINELEN DS    H            LENGTH OF THE PRINT LINE
*
*
*      VARIOUS ITEMS
*
FMT4    DS    XL44          SPACE FOR DSCB NAME
        IECSDSL1 4          SAVE EACH FORMAT 4 DSCB
        DS      0D
FMT3    DS    0XL148        SPACE FOR FORMAT3 DSCB
        IECSDSL1 3
        DS      0D
        MEND
```

IEB817I MEMBER NAME (VTOCOM) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

```
./      ADD  NAME=VTOCPARS
        PUSH PRINT
        PRINT NOGEN
PCLMAIN IKJPARM DSECT=PDL
        SPACE 2
VOLS    IKJPOSIT DSNAME,VOLSER,LIST,          $
        PROMPT='VOLUMES TO SEARCH AND OTHER PARAMETERS',  $
        HELP=('VOLUME SERIAL NUMBERS WHICH ARE TO BE SEARCHED FO$
        DATA SETS TO LIST')
        SPACE 2
LEVKEY  IKJKEYWD
        IKJNAME 'LEVEL',SUBFLD=SUBLEV
        SPACE 2
ENDKEY  IKJKEYWD
        IKJNAME 'ENDING',SUBFLD=SUBEND
        SPACE 2
CONTAINK IKJKEYWD
        IKJNAME 'CONTAINING',SUBFLD=SUBCONT
        SPACE 2
        SPACE 2
SPACEK  IKJKEYWD DEFAULT='TRKS'
        IKJNAME 'TRKS'
        SPACE 2
CATK    IKJKEYWD
        IKJNAME 'CAT'
        SPACE 2
SORTK   IKJKEYWD
        IKJNAME 'SORT',SUBFLD=SUBSORTS
        IKJNAME 'NOSORT'
        SPACE 2
BREAKK  IKJKEYWD
        IKJNAME 'BREAK',SUBFLD=SUBBREAK
        SPACE 2
LIMITK  IKJKEYWD
        IKJNAME 'LIMIT',SUBFLD=SUBLIMIT
        SPACE 2
ANDOR1K IKJKEYWD
        IKJNAME 'AND1',SUBFLD=SUBAO1,ALIAS='AND'
        IKJNAME 'OR1',SUBFLD=SUBAO1,ALIAS='OR'
        SPACE 2
ANDOR2K IKJKEYWD
        IKJNAME 'AND2',SUBFLD=SUBAO2
        IKJNAME 'OR2',SUBFLD=SUBAO2
        SPACE 2
ANDOR3K IKJKEYWD
        IKJNAME 'AND3',SUBFLD=SUBAO3
        IKJNAME 'OR3',SUBFLD=SUBAO3
        SPACE 2
PRINTK  IKJKEYWD
        IKJNAME 'PRINT',SUBFLD=SUBPRINT
        IKJNAME 'NOPRINT'
        SPACE 2
```

CHARSK	IKJKEYWD IKJNAME 'CHARS',SUBFLD=SUBCHARS SPACE 2	
LINESK	IKJKEYWD IKJNAME 'LINES',SUBFLD=SUBLINES SPACE 2	
HEADK	IKJKEYWD IKJNAME 'HEADING',SUBFLD=SUBHEAD IKJNAME 'NOHEADING' SPACE 2	
TOTALK	IKJKEYWD IKJNAME 'TOTALS',SUBFLD=SUBTOTAL SPACE 2	
OUTPUTK	IKJKEYWD IKJNAME 'OUTPUT' SPACE 2	
FORMATK	IKJKEYWD IKJNAME 'FORMAT',SUBFLD=SUBFORMT SPACE 2	
DSNPLNK	IKJKEYWD IKJNAME 'DSNLEN',SUBFLD=SUBDSNLN SPACE 5	
SUBLEV LEVEL	IKJSUBF IKJPOSIT DSNAME,LIST, PROMPT='BEGINNING CHARACTERS OF DSNAMES TO PROCESS' SPACE 2	X
SUBEND ENDING	IKJSUBF IKJPOSIT DSNAME,LIST, PROMPT='ENDING CHARACTERS OF DSNAMES TO PROCESS' SPACE 2	X
SUBCONT CONTAIN	IKJSUBF IKJPOSIT DSNAME,LIST, PROMPT='CHARACTER STRING CONTAINED IN DSNAMES TO PROCESS' , SPACE 2	X
SUBSORTS SUBSORT	IKJSUBF IKJIDENT 'SORT FIELDS',LIST,FIRST=ALPHA,MAXLNTH=6 SPACE 2	
SUBBREAK BREAK	IKJSUBF IKJIDENT 'NUMBER OF CHARACTERS FOR A BREAK',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=2,DEFAULT='3' SPACE 2	X
SUBCHARS CHARSPL	IKJSUBF IKJIDENT 'NUMBER OF CHARACTERS PER LINE',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=3	X
BLKSZSET	IKJIDENT 'PHYSICAL BLOCK SIZE',FIRST=NUMERIC,OTHER=NUMERIC, MAXLNTH=5 SPACE 2	X
SUBLINES LINESPP	IKJSUBF IKJIDENT 'NUMBER OF LINES PER PAGE',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=3 SPACE 2	X

SUBPRINT	IKJSUBF		
SUBPRTKY	IKJIDENT 'ADD, REP, NEW, OR DEL',		X
	FIRST=ALPHA,OTHER=ALPHA,MAXLNTH=3		
SUBPRTIT	IKJIDENT 'ITEMS TO PRINT',LIST,FIRST=ALPHA,MAXLNTH=6		
	SPACE 2		
SUBHEAD	IKJSUBF		
HEADING	IKJPOSIT QSTRING		
	SPACE 2		
SUBTOTAL	IKJSUBF		
TOTALN	IKJIDENT 'NUMBER OF CHARACTERS FOR TOTALS',FIRST=NUMERIC,		X
	OTHER=NUMERIC,MAXLNTH=2,DEFAULT='0'		
	SPACE 3		
SUBLIMIT	IKJSUBF		
SUBLKEY	IKJIDENT 'FIELD IN DATA SET CONTROL BLOCK TO COMPARE',		X
	FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8		
	SPACE 2		
SUBLOPER	IKJIDENT 'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA,		X
	MAXLNTH=2,		X
	PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE'		
	SPACE 2		
SUBLVALU	IKJIDENT 'COMPARISON VALUE',FIRST=ALPHANUM,		X
	OTHER=ALPHANUM,		X
	PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'		
	SPACE 3		
SUBAO1	IKJSUBF		
SUB1KEY	IKJIDENT 'FIELD IN DATA SET CONTROL BLOCK TO COMPARE',		X
	FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8		
	SPACE 2		
SUB1OPER	IKJIDENT 'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA,		X
	MAXLNTH=2,		X
	PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE'		
	SPACE 2		
SUB1VALU	IKJIDENT 'COMPARISON VALUE',FIRST=ALPHANUM,		X
	OTHER=ALPHANUM,		X
	PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'		
	SPACE 2		
SUBAO2	IKJSUBF		
SUB2KEY	IKJIDENT 'FIELD IN DATA SET CONTROL BLOCK TO COMPARE',		X
	FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8		
	SPACE 2		
SUB2OPER	IKJIDENT 'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA,		X
	MAXLNTH=2,		X
	PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE'		
	SPACE 2		
SUB2VALU	IKJIDENT 'COMPARISON VALUE',FIRST=ALPHANUM,		X
	OTHER=ALPHANUM,		X
	PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'		
	SPACE 2		
SUBAO3	IKJSUBF		
SUB3KEY	IKJIDENT 'FIELD IN DATA SET CONTROL BLOCK TO COMPARE',		X
	FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8		
	SPACE 2		

```

SUB3OPER IKJIDENT 'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA, X
              MAXLNTH=2, X
              PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE '
              SPACE 2
SUB3VALU IKJIDENT 'COMPARISON VALUE',FIRST=ALPHANUM, X
              OTHER=ALPHANUM, X
              PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'
              SPACE 2
SUBFORMT IKJSUBF
FORMATSP IKJIDENT 'FORMAT TYPES TO OUTPUT',LIST,FIRST=NUMERIC, X
              MAXLNTH=1,DEFAULT='1'
              SPACE 2
SUBDSNLN IKJSUBF
DSNPLN IKJIDENT 'LENGTH OF DSNAME TO PRINT',FIRST=NUMERIC, X
        OTHER=NUMERIC,MAXLNTH=2,DEFAULT='44'
DSNLNTYP IKJKEYWD DEFAULT='TRUNCATE'
        IKJNAME 'TRUNCATE'
        IKJNAME 'MULTILINE'
        IKJENDP
        SPACE 2
        POP PRINT
ACTION EQU 1
VOLUME EQU 2
CDATE EQU 3
LSTUS EQU 4
EXPDT EQU 5
ALLOC EQU 6
UNUSED EQU 7
PCT EQU 8
EXT EQU 9
DSORG EQU 10
RECFM EQU 11
BLKSZ EQU 12
LRECL EQU 13
PASS EQU 14
CAT EQU 15
SECT EQU 16
SECQ EQU 17
UNIT EQU 18
ROUND EQU 19
TYPE EQU 20
USED EQU 21
CCHH EQU 22
UPD EQU 23 TANSKY
RACF EQU 24 TANSKY
DUMMY5 EQU 25
DSNAME EQU 26
./ ENDUP

```

IEB817I MEMBER NAME (VTOCPARS) NOT FOUND IN NM DIRECTORY. STOWED WITH TTR.

IEB818I HIGHEST CONDITION CODE WAS 00000000

IEB819I END OF JOB IEBUPDTE.


```
./      ADD   NAME=VTOC
)F FUNCTION -
  THE VTOC COMMAND DISPLAYS SELECTED DATA SETS ON A DISK OR SET OF
  DISKS.  EACH DISK HAS A VOLUME TABLE OF CONTENTS ( VTOC ).  THIS
  TABLE IS SEARCHED FOR DATA SETS THAT MEET THE SPECIFICATIONS.
)X SYNTAX -
      VTOC      'VOLUME-LIST'          LEVEL('DSNAME-START')
                                          EXLEVEL('DSNAME-START')
                                          CONTAINING('DSNAME-STRING')
                                          ENDING('DSNAME-END')
                                          BREAK('BREAK-CHARS')

      CAT
      ENQ
      TRK/CYL/KB/MB
      NOSORT/SORT('SORT-FIELDS')
      NOPRINT/PRINT('PRINT-OP' ('PRINT-ITEM-LIST'))
      LIMIT('KEYWORD' 'OPER' 'VALUE')
      AND1/OR1('KEYWORD' 'OPER' 'VALUE')
      AND2/OR2('KEYWORD' 'OPER' 'VALUE')
      AND3/OR3('KEYWORD' 'OPER' 'VALUE')
      CHARS('CHARS-PER-LINE')
      LINES('LINES-PER-PAGE')
      NOHEADING/HEADING('TEXT')
      DSNLEN('LENGTH')

  REQUIRED - 'VOLUME-LIST'
  DEFAULTS - LIST ALL DATA SETS ON THE VOLUME(S) SELECTED.
             SORT, PRINT, TRK
)O OPERANDS -
))'VOLUME-LIST'      - A VOLUME SERIAL NUMBER OR A LIST OF VOLUMES.
  IF THE FIRST ONE TO FIVE CHARACTERS OF A VOLUME
  SERIAL NUMBER ARE ENTERED, ALL VOLUMES THAT ARE
  MOUNTED ON THE MACHINE WHICH START WITH THOSE
  CHARACTERS WILL BE LISTED.
  IF 'ALL' IS SPECIFIED, ALL NON-VIRTUAL VOLUMES
  WHICH ARE ONLINE AND READY WILL BE PROCESSED.
  IF 'ALV' IS SPECIFIED, ALL VIRTUAL VOLUMES
  WHICH ARE ONLINE AND READY WILL BE PROCESSED,
  IF THEY ARE MOUNTED PRIVATE/RESERVED OR
  PRIVATE/RESIDENT.

))LEVEL('DSNAME-START') - SPECIFIES THE HIGH LEVEL QUALIFIERS TO BE
  SEARCHED.  THIS WILL NOT BE PREFIXED BY YOUR USERID OR
  PREFIX.  ONLY DATA SETS STARTING WITH THESE PREFIXES WILL
  BE LISTED.

))EXLEVEL('DSNAME-START') - SPECIFIES THE HIGH LEVEL QUALIFIERS TO
  BE OMITTED.  THIS WILL NOT BE PREFIXED BY YOUR USERID OR
  PREFIX.  DATA SETS STARTING WITH THESE PREFIXES WILL NOT
  BE LISTED.

))CONTAINING('DSNAME-STRING') - SPECIFIES A CHARACTER STRING CONTAINED
  IN THE DATA SET NAME.  AT LEAST ONE OF THE STRINGS MUST
  BE IN THE DSNAME FOR THE DATA SET TO BE LISTED.
  THESE STRINGS NEED NOT CONFORM TO DSNAME STANDARDS.
  THEY CAN BEGIN WITH A PERIOD OR A NUMBER.
```

))ENDING('DSNAME-END') - SPECIFIES THE ENDING CHARACTERS OF THE DSNAME. THE FINAL NONBLANK CHARACTERS OF THE DSNAME MUST BE ONE OF THESE STRINGS TO ALLOW THE DATA SET TO BE LISTED. THESE STRINGS MUST CONFORM TO DSNAME STANDARDS.

))CAT - A LOCATE IS DONE FOR EACH DSNAME ON THE VOLUMES LISTED AND STATUS IS INDICATED. NOTE - THIS OPTION USES A CONSIDERABLE AMOUNT OF PROCESSING TIME.
C - CATALOGED ON THIS VOLUME
N - NOT CATALOGED
W - CATALOGED ON ANOTHER VOLUME
E - CATALOG PROCESSING ERROR

))ENQ - A TEST IS PERFORMED FOR AN ENQ WITH A MAJOR NAME OF SYSDSN AND A MINOR NAME OF THE DATASET NAME TO FIND OUT IF THE DATASET IS IN USE. IF THE DATASET IS VSAM THE CLUSTER NAME IS LOCATED AND USED FOR THE ENQ TEST.
NOTE - THIS OPTION USES A LOT OF PROCESSING TIME.
N - NOT ENQUEUED
S - ALLOCATED WITH DISP=SHR
E - EXCLUSIVELY ALLOCATED
? - CATALOG PROCESSING ERROR
* - ENQUEUE PROCESSING ERROR

))TRK - SPACE IS TO BE REPORTED IN TRACK UNITS.
))CYL - SPACE IS TO BE REPORTED IN CYLINDER UNITS.
))KB - SPACE IS TO BE REPORTED IN KILOBYTE UNITS.
))MB - SPACE IS TO BE REPORTED IN MEGABYTE UNITS.
))NOSORT - THE DATA SETS ARE NOT SORTED. THEY ARE OUTPUT AS THEY ARE FOUND.

))SORT('SORT-FIELDS') DATA SETS ARE SORTED INTO ALPHABETICAL ORDER, BASED UPON THE SORT FIELDS SPECIFIED.
DSNAME,VOLUME,ALLOC,USED,UNUSED,PCT,EX,DSO,RFM,
LRECL,BKLSZ,CDATE,EXPDT,REFDT ARE VALID SORT FIELDS.
'A/D' IS REQUIRED AFTER EACH SORT FIELD TO INDICATE ASCENDING/DESCENDING SEQUENCE.

))BREAK('BREAK-CHARS') - THE LISTING WILL CONTAIN A NEW HEADER, (ON A NEW PAGE IF THE VTOCOUT DD CARD OPTION IS USED), WHENEVER THE SPECIFIED NUMBER OF CHARACTERS DIFFERS FROM THE PRECEDING DATA SET. THIS OPTION FUNCTIONS ONLY WITH THE SORT OPTION.

))CHARS('CHARS-PER-LINE') - SPECIFIES THE NUMBER OF CHARACTERS ON EACH LINE OF OUTPUT. THE DEFAULT IS 150 FOR PRINT AND THE LINESIZE OF THE TERMINAL FOR TSO SESSIONS. YOU CAN GET MORE INFORMATION BY SPECIFYING A LARGER NUMBER OF CHARACTERS PER LINE OR YOU CAN LIMIT THE PRINTING BY SETTING A SMALLER NUMBER OF CHARACTERS PER LINE OF OUTPUT.

))LINES('LINES-PER-PAGE') - SPECIFIES THE NUMBER OF LINES BEFORE A NEW TITLE LINE IS PRODUCED. IT DEFAULTS TO 60 FOR PRINT AND TO THE SCREEN SIZE FOR TSO SESSIONS.

))NOHEADING - DO NOT PRODUCE A HEADING. THE HEADING WILL ONLY BE OUTPUT IF THE VTOCOUT DD STATEMENT IS PRESENT.

))HEADING('TEXT') - IF A DD STATEMENT WITH A DDNAME OF VTOCOUT IS PRESENT, THIS TEXT WILL BE USED TO BEGIN EVERY PAGE.

CARRIAGE CONTROL SHOULD BE INCLUDED (ASA). THE
DEFAULT HEADER CONSISTS OF VTOC COMMAND VERSION 02
AND THE COMMAND THAT WAS ENTERED.

-))NOTOTALS - DO NOT PRODUCE A TOTALS LINE. THE TOTALS LINE WILL BE
OMITTED EVEN IF THE VTOCOUT DD STATEMENT IS PRESENT. THE
DEFAULT IS TO PRODUCE ONE FINAL TOTAL LINE.
-))DSNLEN('LENGTH') - SPECIFIES THE LENGTH OF THE DSNAME TO PRINT.
THE REST OF THE DSNAME IS TRUNCATED. THE CHARS PARAMETER
WILL ALSO CAUSE THE DSNAME TO BE TRUNCATED, IF THE NAME
AND THE PRECEDING INFORMATION EXCEEDS THE PRINT LINE.
-))NOPRINT - SPECIFIES THAT INDIVIDUAL ITEMS ARE NOT TO BE LISTED.
THE COMMAND CAN BE USED TO CALCULATE TOTALS.
-))PRINT('PRINT-OP' ('PRINT-ITEM-LIST')) -
SPECIFIES THE ITEMS TO PRINT. THE 'PRINT-OP' IS THE
OPERATION TO BE DONE. THEY INCLUDE THE FOLLOWING.
- NEW - THE 'PRINT-ITEM-LIST' IS A COMPLETE LIST OF WHAT
TO PRINT.
 - REP - THE FIRST 'PRINT-ITEM' WILL BE REPLACED WITH THE
REST OF THE ITEMS ON THE LIST.
 - ADD - THE REST OF THE 'PRINT-ITEM-LIST' WILL BE ADDED
AFTER THE FIRST ITEM ON THE LIST.
 - DEL - THE ITEMS ON THE 'PRINT-ITEM-LIST' WILL NOT BE
PRINTED.

THE 'PRINT-ITEM-LIST' NAMES ARE THE SAME KEYWORDS USED
IN LIM, AND, AND OR FUNCTIONS AND ARE ALSO THE TITLES
AS PRINTED. THESE KEYWORDS CAN BE ALLOC, UNUSED, USED,
PCT, EX, DSO, RFM, BLKSZ, LRECL, CDATE, EXPDT, REFDT,
SECT, PASS, ROUND, CCHH, VOLUME, UNIT, DSNAME, ACTION,
TYPE, RACF OR UPD.

THE ADD, DEL, AND REP PRINT OPERATIONS REFER TO THE
DEFAULT PRINT LIST. THE DEFAULT LIST IS ALLOC, UNUSED,
PCT, EX, DSO, RFM, BLKSZ, LRECL, REFDT, CDATE, VOLUME,
DSNAME, EXPDT, SECQ, SECT, ROUND, PASS, ACTION, AND
TYPE. AS NOTED UNDER CHARS ABOVE, ONLY THE ITEMS THAT
WILL FIT ON THE PRINT LINE WILL BE LISTED.

NOTE: IF CCHH IS CHOSEN IT DEFAULTS TO THE CCHH RANGE
OF THE FIRST EXTENT; HOWEVER, IF LIM/AND/OR SELECTION
IS PERFORMED, THE CCHH RANGE OF THE EXTENT WHICH
SATISFIES THE SELECTION WILL BE USED INSTEAD.

-))LIMIT('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. ONLY DATA SETS THAT SATISFY THE RELATION
ARE LISTED.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.
'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.
'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

)) 'KEYWORD' - IS THE NAME OF A DATA SET FIELD. THE LIST OF CURRENTLY PROVIDED FIELDS FOLLOWS. THE KEYWORDS AND THEIR VALUES ARE THE SAME AS IN THE VTOC OUTPUT.

ALLOC	DATA SET ALLOCATION NUMBER OF KILOBYTES, TRACKS, CYLINDERS, OR MEGABYTES ALLOCATED. DEFAULT IS IN TRACKS.
UNUSED	AMOUNT OF UNUSED SPACE IN THE DATA SET. SAME UNITS AS IN ALLOC.
USED	AMOUNT OF SPACE USED IN THE DATA SET. SAME UNITS AS IN ALLOC.
PCT	PERCENTAGE OF SPACE USED IN THE DATA USED.
EX	NUMBER OF EXTENTS IN THE DATA SET.
DSO	DATA SET ORGANIZATION PS = SEQUENTIAL PO = PARTITIONED VS = VSAM PE = PDSE DA = DIRECT IS = ISAM U = UNMOVEABLE HF = HFS X = EXTENDED FORMAT
RFM	RECORD FORMAT F = FIXED V = VARIABLE U = UNDEFINED B = BLOCKED T = TRACK OVERFLOW S = SPANNED OR STANDARD A = ASA CARRIAGE CONTROL M = MACHINE CARRIAGE CONTROL
BLKSZ	BLOCKSIZE FOR PHYSICAL BLOCKS OF DATA.
LRECL	LOGICAL RECORD LENGTH IN BYTES.
CDATE	CREATION DATE IN THE FORM YYDDD, SOMETIMES CALLED JULIAN.
EXPDT	EXPIRATION DATE IN THE SAME FORM. THIS FIELD IS RARELY USED HERE.
REFDT	LAST USE DATE IN THE SAME FORM. THIS DATE IS WHEN THE DATA SET WAS LAST OPENED.
SECT	TYPE OF ALLOCATION A = ABSOLUTE TRACK B = BLOCKS T = TRACKS C = CYLINDERS
PASS	PROTECTION INDICATORS N = NONE R = READ AND WRITE PROTECTION W = WRITE PROTECTION
ROUND	SPACE ROUNDED UP TO CYLINDERS R = ROUND N = NO ROUND
CCHH	CYLINDER AND HEAD ADDRESS, IN 4 OR 8 HEXADECIMAL DIGITS. IF 4 DIGITS

ARE USED, ONLY THE CYLINDER IS USED
FOR COMPARISON, OTHERWISE, THE CYL
AND TRACK ARE COMPARED.

VOLUME	VOLUME SERIAL NUMBER OR DISK NAME
UNIT	UNIT OR DEVICE TYPE
DSNAME	NAME OF THE DATA SET
ACTION	SOME ERROR INDICATIONS
TYPE	RESERVED FOR EXIT USAGE.
RACF	RACF INDICATOR

N = NOT INDICATED
Y = INDICATED

UPD	UPDATED SINCE LAST BACKUP
-----	---------------------------

N = NOT UPDATED
Y = UPDATED

))'OPER' - IS AN OPERATOR. THE LIST OF OPERATORS FOLLOWS.

EQ	IS EQUAL TO
NE	IS NOT EQUAL TO
LE	IS LESS THAN OR EQUAL TO
LT	IS LESS THAN
GE	IS GREATER THAN OR EQUAL TO
GT	IS GREATER THAN

))'VALUE' - GIVES THE VALUE OF THE ITEM FOR COMPARISON, SUCH AS
FB, PS, R, OR A NUMBER.

))AND1('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. BOTH THE LIMIT AND THIS CONDITION MUST
BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

))OR1('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. EITHER THE LIMIT OR THIS CONDITION MUST
BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

))AND2('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. BOTH THE PREVIOUS RESULT AND THIS
CONDITION MUST BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR

A NUMBER LIKE 51.

))OR2('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. EITHER THE PREVIOUS RESULT OR THIS
CONDITION MUST BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

))AND3('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. BOTH THE PREVIOUS RESULT AND THIS
CONDITION MUST BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

))OR3('KEYWORD' 'OPER' 'VALUE') - SPECIFIES WHICH DATA SETS ARE TO
BE LISTED. EITHER THE PREVIOUS RESULT OR THIS
CONDITION MUST BE TRUE TO ALLOW THE LISTING.

'KEYWORD' CAN BE ALLOC, UNUSED, USED, PCT, EX, DSO, RFM,
BLKSZ, LRECL, CDATE, EXPDT, REFDT, SECT, PASS, ROUND,
CCHH, VOLUME, UNIT, DSNAME, ACTION, TYPE, RACF OR UPD.

'OPER' CAN BE EQ, NE, LE, LT, GE OR GT.

'VALUE' CAN BE A COMPARISON VALUE SUCH AS FB, PS, R OR
A NUMBER LIKE 51.

))EXAMPLES -

1. LIST ALL DATA SETS ON VOL.
VTOC VOL
2. LIST ALL DATA SETS ON ALL VOLUMES.
VTOC ALL
3. LIST ALL DATA SETS THAT START WITH XXX ON ANY MVSXX VOLUME.
VTOC MVS LEV(XXX)
4. LIST ALL DATA SETS CONTAINING LIST NOT BEGINNING WITH SYS1.
VTOC VOL CONT(LIST) EXLEV(SYS1)
5. LIST ALL RECENTLY CREATED DATA SETS.
VTOC VOL LIM(CDATE GT 94001)
6. LIST DATA SETS WITH UNUSED SPACE.
VTOC VOL LIM(PCT LT 50) OR1(UNUSED GT 30)
7. LIST DATA SETS WITH MULTIPLE EXTENTS.

VTOC VOL LIM(EX GT 1)

8. LIST ALL OF A USER'S DATA SETS ON TSO001.
VTOC TSO001 LEVEL(TMTCEXX)
 9. LIST ALL DATA SETS OVER 100 TRKS.
VTOC VOL LIM(ALLOC GT 100)
 10. LIST DATA SETS IN VTOC SEQUENCE.
VTOC STR902 NOSORT
 11. LIST DATA SETS UNDER THE FIXED HEADS.
VTOC VOL LIM(CC EQ 0001) OR1(CC EQ 0002)
 12. LIST OLD PDS'S WITH FREE SPACE.
VTOC VOL LIM(CDATE LT 92001) AND1(DSO EQ PO) AND2(UNUSED GT 30)
 13. LIST USED SPACE INSTEAD OF UNUSED IN TRACKS.
VTOC VOL PRINT(REP (UNUSED USED)) TRK
 14. LIST CLIST DATA SETS.
VTOC VOL END(CLIST)
 15. LIST ALL OF THE INFORMATION ABOUT USER'S DATA SETS AT A 3270.
VTOC VOL CHAR(150)
 16. CHECK IF DATASETS ARE CATALOGED ON THIS VOLUME.
VTOC VOL CAT PRINT(NEW (ALLOC PCT CAT DSNAME))
 17. LIST ANY DATA SET ON A BAD TRACK.
VTOC STR902 LIM(CCHH EQ 003C0003) PRINT(REP (REFDT CCHH))
 18. LIST ANY DATA SETS ON A BAD CYLINDER.
VTOC STR902 LIM(CCHH EQ 003C) PRINT(REP (REFDT CCHH))
 19. LIST ALL DATA SETS ON 335XXX VOLUMES SORTED BY ALLOC IN DESCENDING
SEQUENCE, VOLUME AND DSNAME IN ASCENDING SEQUENCE.
VTOC 335 SORT(ALLOC,D,VOLUME,A,DSNAME,A)
- ./ ENDUP

IEB816I MEMBER NAME (VTOC) FOUND IN NM DIRECTORY. TTR IS NOW ALTERED.

IEB818I HIGHEST CONDITION CODE WAS 00000000

IEB819I END OF JOB IEBUPDTE.

SYMBOL TYPE ID ADDR LENGTH LDID

ASM 0201 13.38 01/07/25

VTOCCMD	SD	0001	000000	00120C	
VTOCMSG	ER	0002			
VTOCEXCP	ER	0003			
VTOCCHEK	ER	0004			
VTOCFORM	ER	0005			
VTOCPRNT	ER	0006			
VTOCSORT	ER	0007			
PCLMAIN	SD	0008	001210	0007CE	


```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25
2  *****
3  *
4  *
5  * TITLE -      VTOC COMMAND - LIST DATA SETS AND ATTRIBUTES
6  *
7  * FUNCTION -   PROVIDE THE ABILITY FOR A TSO USER OR A BATCH JOB
8  *              TO LIST THE CONTENTS OF VARIOUS VOLUMES, WITH A
9  *              FAIR AMOUNT OF SELECTION.
10 *
11 *
12 * OPERATION -  ACCEPT FROM THE TSO USER OR BATCH JOB A COMMAND
13 *              WITH THE FOLLOWING SYNTAX.  THEN CHECK THE COMMAND
14 *              AND LOOP THROUGH, GETTING A DSCB, FORMATTING IT,
15 *              PERFORMING THE DATA SET NAME AND LIMIT CHECKS, AND
16 *              CALLING AN EXIT ROUTINE IF DESIRED, THEN PUT THE
17 *              ENTRY IN THE CORRECT SORT SEQUENCE.
18 *              FINALLY CALL THE PRINT ROUTINE TO PRINT THE
19 *              SPECIFIED ITEMS, HEADERS, AND BREAKS, OR JUST
20 *              THE TOTALS.
21 *
22 *
23 * INPUT -      STANDARD COMMAND PROCESSOR PARAMETER LIST
24 *              POINTED TO BY REGISTER 1
25 *
26 *
27 * OUTPUT -     TO SYSOUT, A LIST OF THE REQUESTED DATA SETS AND
28 *              THEIR ATTRIBUTES.
29 *
30 *
31 * ATTRIBUTES - REENTRANT, REUSEABLE, REFRESHABLE.
32 *
33 *
34 *              PROGRAMMED BY R. L. MILLER (415) 485-6241
35 *              FIREMAN'S FUND INSURANCE  CPSD 2N
36 *              ONE LUCAS GREEN
37 *              SAN RAFAEL, CA  94911
38 *
39 * Nov 05,2024 RPrins: Add sort field RACF and UNIT. Add the device
40 *                   type (like 3390) in the Unit field (VSFUNIT)
41 *
42 *****
43 *
44          MACRO
45 &LABEL  VTOCEXCP  &FUNC
46          AIF    ('&FUNC' NE 'EQ').CALL
47 VTCOPEN EQU    1          DEFINE FUNCTION CODES FOR VTOCEXCP
48 VTCCLOSE EQU    2
49 VTCREAD EQU    0
50          MEXIT
51 .CALL    ANOP          CALL VTOCEXCP
52 &LABEL  MVI    VTCEFUNC,VTC&FUNC  SET THE FUNCTION CODE
53          VTCALL EXCP    GO GET A DSCB
54          MEND
55 *
56 *          MACRO FOR INITIALIZING SUBROUTINE WORK AREA ADDRESSES

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
-----	-------------	-------	-------	------	------------------	-------------------------

57 *

58 MACRO

59 &LABEL WORKADDR &RTN,&PRMADDR

60 &LABEL L R1,=A(WORK&RTN-WORKAREA) GET THE OFFSET (OVER 4K)

61 LA R1,0(R1,R13) RELOCATE IT

62 ST R1,&PRMADDR THEN STORE IT FOR THE ROUTINES

63 MEND

64 *

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				66	VTOCCMD	ENTERX 12,(1,LENWORK,C)	DO THE HOUSEKEEPING
000000				67+	VTOCCMD	CSECT	
000000	47F0 F00C	0000C		68+	B	12(0,15)	BRANCH AROUND ID 00450000
000004	07			69+	DC	AL1(7)	LENGTH OF IDENTIFIER 00550000
000005	E5E3D6C3C3D4C4			70+	DC	CL7'VTOCCMD'	IDENTIFIER 00750000
00000C	90EC D00C	0000C		71+	STM	14,12,12(13)	SAVE REGISTERS 03700000
000010	18CF			72+	LR	12,15	SET FIRST BASE REG
000012	0700			73+	CNOP	0,4	
			00000	74+	USING	VTOCCMD,12	
000014				75+	CNOP	0,4	DO A GETMAIN FOR THE AREA
000014	4510 C01C	0001C		76+	BAL	1,*+8	POINT THE SP AND LV
000018	01			77+	ENT0001	DC AL1(1)	SUBPOOL FOR THE GETMAIN
000019	002370			78+	DC	AL3(LENWORK)	LENGTH OF THE GETMAIN
00001C	5801 0000	00000		79+	L	0,0(1)	GET THE DATA IN REG 1
000020	0A0A			80+	SVC	10	ISSUE THE GETMAIN
000022	50D1 0004	00004		81+	ST	13,4(1)	PRIOR SAVEAREA ADDRESS TO MINE
000026	501D 0008	00008		82+	ST	1,8(13)	MY SAVEAREA ADDRESS TO HIS
00002A	182D			83+	LR	2,13	KEEP THE SAVEAREA ADDRESS FOR REGS
00002C	18D1			84+	LR	13,1	THIS IS MY SAVEAREA
00002E	414D 000C	0000C		85+	LA	4,12(13)	YES, POINT PAST THE CHAIN
000032	5850 C018	00018		86+	L	5,ENT0001	GET THE SIZE
000036	4160 000C	0000C		87+	LA	6,12	MINUS THE CHAIN AREA (12 BYTES)
00003A	1B56			88+	SR	5,6	GIVES THE AMOUNT TO CLEAR
00003C	1B77			89+	SR	7,7	CLEAR THE FROM COUNT AND CLEAR BYTE
00003E	0E46			90+	MVCL	4,6	WHEE, CLEAR IT OUT
000040	9807 2014	00014		91+	LM	0,7,20(2)	RESTORE THE ORIGINAL REGISTERS
000044	1821			92	LR	R2,R1	SAVE ADDR OF CPPL
			00000	94	USING	WORKAREA,WORKREG	

```

ASM 0201 13.38 01/07/25

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
000046 45E0 C43C      0043C      96      BAL  R14,PARSINIT  PERFORM THE PARSING
00004A 12FF                97      LTR  R15,R15      TEST THE RETURN CODE
00004C 4770 C396      00396      98      BNZ  RETURN      BAD NEWS, GET OUT
                99      VTCALL PRNT      INITIALIZE FOR PRINTING
000050 4110 D070      00070     100+     LA   R1,VTOCOM   POINT TO THE COMMON AREA
000054 58F0 D0A0      000A0     101+     L    R15,VADPRNT POINT TO THE ROUTINE
000058 05EF                102+     BALR R14,R15     THEN CALL IT
00005A 5890 D0B8      000B8     103      L    R9,ADDRANSR ADDR OF PARSE DESCRIPTOR LIST
                00000  104      USING PDL,R9     RETURNED BY PARSE
                105 *
                106 *
                107 *      SCAN SORT PARSE LIST AND BUILD SORT FIELD TABLE
                108 *
                109 *
00005E 4140 90A0      000A0     110 SORTPAR LA   R4,SUBSORT  SORT PARSE LIST
000062 4150 D238      00238     111      LA   R5,SORTTAB  SORT FIELD TABLE
000066 D73F 5000 5000 00000 00000  112      XC   0(64,R5),0(R5) CLEAR SORT FIELD TABLE
00006C D203 5000 C63C 00000 0063C  113      MVC  0(4,R5),SORTTABX DEFAULT TO DSNAME

000072 4110 C630      00630     115 SORTPAR1 LA   R1,SORTTABX-12 SORT COMPARE TABLE

000076 4110 100C      0000C     117 SORTPAR2 LA   R1,12(0,R1)  POINT TO NEXT COMPARE ENTRY
00007A D503 1000 C5C0 00000 005C0  118      CLC  0(4,R1),=F'0' END OF TABLE
000080 4780 C0A0      000A0     119      BE   SORTPAR3    ITEM NOT FOUND, IGNORE
000084 5860 4000      00000     120      L    R6,0(0,R4)  POINT TO TEXT
000088 4830 4004      00004     121      LH   R3,4(0,R4)  TEXT LENGTH
00008C 1233                122      LTR  R3,R3        IGNORE IF ZERO
00008E 4780 C0A0      000A0     123      BZ   SORTPAR3
000092 0630                124      BCTR R3,0
000094 4430 C0E6      000E6     125      EX   R3,SORTCOMP FIELD NAME MATCH
000098 4780 C0AC      000AC     126      BE   SORTPAR4    YES
00009C 47F0 C076      00076     127      B    SORTPAR2    NO, TRY NEXT

0000A0 BF47 4009      00009     129 SORTPAR3 ICM  R4,7,9(R4)    NEXT ITEM
0000A4 4770 C072      00072     130      BNZ  SORTPAR1    CONTINUE IF MORE
0000A8 47F0 C0EC      000EC     131      B    SORTPAR5

0000AC D203 5000 1000 00000 00000  133 SORTPAR4 MVC  0(4,R5),0(R1)  SET UP SORT FIELD
0000B2 BF47 4009      00009     134      ICM  R4,7,9(R4)  ASCENDING/DESCENDING INDICATOR
0000B6 4780 C38C      0038C     135      BZ   PARMERR     ERROR IF MISSING
0000BA 5860 4000      00000     136      L    R6,0(0,R4)  INDICATOR ADDR
0000BE D501 4004 C5C0 00004 005C0  137      CLC  4(2,R4),=F'0' ERROR IF MISSING
0000C4 4780 C38C      0038C     138      BE   PARMERR
0000C8 D200 5000 6000 00000 00000  139      MVC  0(1,R5),0(R6) A/D INDICATOR
0000CE 4150 5004      00004     140      LA   R5,4(0,R5)
0000D2 95C1 6000      00000     141      CLI  0(R6),C'A'  ASCENDING SORT
0000D6 4780 C0A0      000A0     142      BE   SORTPAR3    YES, OK
0000DA 95C4 6000      00000     143      CLI  0(R6),C'D'  DESCENDING SORT
0000DE 4770 C38C      0038C     144      BNE  PARMERR     NO, ERROR
0000E2 47F0 C0A0      000A0     145      B    SORTPAR3    CHECK IF ANY MORE

0000E6 D500 1004 6000 00004 00000  147 SORTCOMP CLC  4(0,R1),0(R6)

0000EC 4130 C7B4      007B4     149 SORTPAR5 LA   R3,SORTKTAB-12 SORT HEADER INDEX TABLE
0000F0 4130 300C      0000C     150 SORTK1  LA   R3,12(0,R3)  NEXT ENTRY

```

```

ASM 0201 13.38 01/07/25
LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
0000F4 D503 3000 C5C0 00000 005C0  151      CLC  0(4,R3),=F'0'  END OF TABLE
0000FA 4780 C14C          0014C  152      BE   SORTK3        YES
0000FE D500 D239 3001 00239 00001  153      CLC  SORTTAB+1(1),1(R3)  ENTRY MATCH
000104 4770 C0F0          000F0  154      BNE  SORTK1        NO, CHECK NEXT
000108 1B44          155      SR   R4,R4
00010A 4850 3002          00002  156      LH   R5,2(0,R3)    LOAD TABLE LENGTH
00010E 5D40 C5C4          005C4  157      D    R4,=F'12'     TABLE ENTRIES
000112 4150 5001          00001  158      LA   R5,1(0,R5)
000116 4160 D348          00348  159      LA   R6,VTCSORTH
00011A 5840 3004          00004  160      L    R4,4(0,R3)    LOAD TABLE BEGIN ADDR
00011E 95C4 D238          00238  161      CLI  SORTTAB,C'D'  DESCENDING SORT
000122 4780 C12A          0012A  162      BE   SORTK2        YES
000126 5840 3008          00008  163      L    R4,8(0,R3)    LOAD TABLE END ADDR
00012A D20B 6000 4000 00000 00000  164 SORTK2 MVC  0(12,R6),0(R4)
000130 4140 400C          0000C  165      LA   R4,12(0,R4)
000134 4160 600C          0000C  166      LA   R6,12(0,R6)
000138 95C4 D238          00238  167      CLI  SORTTAB,C'D'  DESCENDING SORT
00013C 4780 C144          00144  168      BE   *+8           YES
000140 5B40 C5C8          005C8  169      S    R4,=F'24'
000144 4650 C12A          0012A  170      BCT  R5,SORTK2
000148 47F0 C152          00152  171      B    SORTK4
00014C D20B D348 C5CC 00348 005CC  172 SORTK3 MVC  VTCSORTH(12),=3F'0'
000152 D20B 6000 C5CC 00000 005CC  173 SORTK4 MVC  0(12,R6),=3F'0'
174 *
175 *      CHECK THROUGH THE UCB'S TO SELECT THE VOLUMES TO PROCESS
176 *
177 *
178 **   FIND A VOLUME SERIAL NUMBER
179 *
000158 4130 9008          00008  180      LA   R3,VOLS      POINT TO THE PDL
00015C 5853 0000          00000  181 LOOP1 L    R5,0(R3)     GET THE ADDRESS OF THE TEXT
000160 4843 0004          00004  182      LH   R4,4(R3)     ALSO GET ITS LENGTH
000164 1244          183      LTR  R4,R4        FOR EXECUTES, GET THE LENGTH
000166 4780 C37E          0037E  184      BZ   PHASE2       NO MORE VOLUMES, CONTINUE TO NEXT PHASE
00016A 0640          185      BCTR R4,0         MAKE IT READY FOR THE EX INSTR
00016C D205 D223 C61C 00223 0061C  186      MVC  VOLSER,BLANKS INITIALIZE FIELD
000172 4440 C630          00630  187      EX   R4,MOVVOL
188 *
189 **   VOLUME FOUND - VERIFY AND CHECK FOR GLOBAL OR SPECIAL REQUESTS
190 *
000176 4940 C628          00628  191      CH   R4,H5        IS THE ENTIRE NAME THERE?
00017A 4780 C1AE          001AE  192      BE   VOLSET       YES, IT'S A SPECIFIC VOLUME
00017E 9201 D22F          0022F  193      MVI  FLAG,X'01'   IT'S A GENERIC REQUEST
000182 4940 C624          00624  194      CH   R4,H2        CHECK FOR THE ALL KEYWORD, FIRST LENGTH
000186 4770 C1AE          001AE  195      BNE  VOLSET       NOT A GLOBAL REQUEST
00018A D502 5000 C62D 00000 0062D  196      CLC  0(3,R5),CHARALV IS THIS THE KEYWORD 'ALLV'?
000190 4780 C1A6          001A6  197      BE   VOLSETV      NO, NOT A GLOBAL REQUEST
000194 D502 5000 C62A 00000 0062A  198      CLC  0(3,R5),CHARALL IS THIS THE KEYWORD 'ALL'?
00019A 4770 C1AE          001AE  199      BNE  VOLSET       NO, NOT A GLOBAL REQUEST
00019E 9202 D22F          0022F  200      MVI  FLAG,X'02'   GLOBAL REQUEST
0001A2 47F0 C1AE          001AE  201      B    VOLSET
202 *
203 **   FIND THE A(UCB)
204 *
0001A6 9282 D22F          0022F  205 VOLSETV MVI  FLAG,X'82'   GLOBAL REQUEST FOR VIRTUAL

```

```

ASM 0201 13.38 01/07/25

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
0001AA 47F0 C1AE      001AE      206      B      VOLSET
0001AE D703 D230 D230 00230 00230  207 VOLSET XC      LASTADR, LASTADR CLEAR THE UCB COMPARE ADDRESS
0001B4 5850 0010      00010      208      L      R5,16  A(CVT)
0001B8 1B66      209      SR      R6,R6
0001BA 5855 0028      00028      210      L      R5,40(R5)  A(UCB ADDRESSES)
0001BE BF63 5000      00000      211 NEXTUCB ICM     R6,3,0(R5)  A(A UCB)
0001C2 1266      212      LTR     R6,R6  CHECK FOR VALID ENTRIES
0001C4 4780 C20A      0020A      213      BZ      INCR1  UCB HOLE
0001C8 5960 C618      00618      214      C      R6,FMIN1  CHECK FOR END
0001CC 4780 C212      00212      215      BE      NOTMNT  END OF UCB LIST - VOLUME NOT FOUND
0001D0 9520 6012      00012      216      CLI     18(R6),X'20'  MUST BE DIRECT ACCESS
0001D4 4770 C20A      0020A      217      BNE     INCR1
0001D8 5960 D230      00230      218      C      R6, LASTADR  UCB ADDRESSES MUST INCREASE
0001DC 47D0 C20A      0020A      219      BNH     INCR1  OTHERWISE THEY REPEAT.
0001E0 5060 D230      00230      220      ST      R6, LASTADR  NEW ADDRESS
0001E4 9102 D22F      0022F      221      TM      FLAG,X'02'  CHECK FOR GLOBAL
0001E8 4710 C25A      0025A      222      BO      FNDGBL  IT IS
0001EC 9101 D22F      0022F      223      TM      FLAG,X'01'  CHECK FOR SPECIAL REQUESTS
0001F0 4710 C202      00202      224      BO      SPECUCB  IT IS
0001F4 D505 D223 601C 00223 0001C  225      CLC     VOLSER,28(R6)  COMPARE FULL VOLSER
0001FA 4780 C28E      0028E      226      BE      FNDUCB  FOUND IT
0001FE 47F0 C20A      0020A      227      B      INCR1
000202 4440 C636      00636      228 SPECUCB EX      R4,CLCVOL  COMPARE FIRST X CHARACTERS ONLY
000206 4780 C262      00262      229      BE      CHKRDY
00020A 4155 0002      00002      230 INCR1  LA      R5,2(R5)
00020E 47F0 C1BE      001BE      231      B      NEXTUCB  TRY NEXT UCB
232 *
233 *      VARIOUS ERRORS, LET THE PERSON KNOW
234 *
000212 9104 D22F      0022F      235 NOTMNT TM      FLAG,X'04'  WAS A VOLUME FOUND?
000216 4710 C376      00376      236      BO      NEXTVOL  YES, LOOK FOR THE NEXT SPEC
00021A D27B D16C C730 0016C 00730  237      MVC     MSGTEXT2,MSGNOTMT  NO, GET THE ERROR MESSAGE
000220 D205 D171 D223 00171 00223  238 SETVOL MVC     MSGTEXT2+5(6),VOLSER  ADD THE VOLUME SERIAL NUMBER
239      VTOCMSG MSGTEXT2  AND ISSUE THE MESSAGE
000226 4110 D16C      0016C      240+     LA      R1,MSGTEXT2  POINT TO THE FIRST MESSAGE
00022A 1B00      241+     SR      R0,R0  NO SECOND LEVEL MESSAGE
00022C 9001 D0E8      000E8      242+     STM     R0,R1,MSGADDRS  SAVE THE MESSAGE ADDRESSES
243+*     THEN JUST CALL THE MESSAGE ISSUING ROUTINE
000230 4110 D070      00070      244+     LA      R1,VTOCOM  POINT TO THE COMMON AREA
000234 58F0 D08C      0008C      245+     L      R15,VADMSG  POINT TO THE ROUTINE
000238 05EF      246+     BALR   R14,R15  THEN CALL IT
00023A 47F0 C376      00376      247      B      NEXTVOL  GO GET THE NEXT VOLUME FROM PARSE
00023E D27B D16C C770 0016C 00770  248 PENDING MVC     MSGTEXT2,MSGPEND  SET UP THE MESSAGE
249 *
250 *      SEE IF THIS IS A GENERIC OR GLOBAL REQUEST
251 *
000244 9103 D22F      0022F      252      TM      FLAG,X'03'  WAS IT ALL OR A PARTIAL VOLUME SERIAL?
000248 4770 C20A      0020A      253      BNZ     INCR1  IN EITHER CASE, SKIP THE MESSAGE
254 *      THEN FIND MORE VOLUMES
255 *
256 *      OUTPUT THE OFFLINE PENDING MESSAGE
00024C 47F0 C220      00220      257      B      SETVOL  THEN ADD THE VOLUME
000250 D27B D16C C752 0016C 00752  258 OFFLINE MVC     MSGTEXT2,MSGOFFLN  SET UP THE MESSAGE
000256 47F0 C220      00220      259      B      SETVOL  THEN ADD THE VOLUME
260 *

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

261 ** FOR GLOBAL REQUESTS JUST LIST ONLINE PACKS
262 *
00025A 9180 6003      00003 263 FNDGBL  TM    3(R6),X'80'  ONLINE BIT
00025E 4780 C20A      0020A 264          BZ    INCR1   NOPE
265 *
266 ** FOR GLOBAL AND SPECIAL REQUESTS, CHECK FOR DEVICE READY
267 *
000262 9140 6006      00006 268 CHKRDY  TM    6(R6),X'40'  TEST READY BIT
000266 4710 C20A      0020A 269          BO    INCR1   NOT READY
00026A 9180 D22F      0022F 270          TM    FLAG,X'80'  GLOBAL REQUEST FOR VIRTUAL
00026E 4710 C286      00286 271          BO    CHKVIRT
000272 9102 D22F      0022F 272          TM    FLAG,X'02'  GLOBAL REQUEST
000276 4780 C28E      0028E 273          BZ    FNDUCB
00027A 9108 6011      00011 274          TM    17(R6),X'08'  VIRTUAL UCB
00027E 4710 C20A      0020A 275          BO    INCR1   YES
000282 47F0 C28E      0028E 276          B     FNDUCB
000286 9108 6011      00011 277 CHKVIRT  TM    17(R6),X'08'  VIRTUAL UCB
00028A 4780 C20A      0020A 278          BZ    INCR1   NO
279 *
280 ** MOVE UCB INFORMATION TO OUTPUT LINE
281 *
00028E D205 D229 601C 00229 0001C 282 FNDUCB  MVC   VOLID,28(R6)  MOVE VOLID
000294 D202 D220 600D 00220 0000D 283          MVC   ADDR,13(R6)  MOVE UNIT ADDRESS
00029A D203 D234 6010 00234 00010 284          MVC   UCBDEVT(4),16(R6)  UCB UNIT TYPE           RPRINS
0002A0 4110 C795      00795 285          LA    R1,DEVTYPS      UCCTYPE, UNIT NAME TABLE RPRINS
0002A4 41E0 0008      00008 286          LA    R14,DEVT#       NUMBER OF DEVICES        RPRINS
0002A8 D500 D237 1000 00237 00000 287 FNDUCB2 CLC   UCBDEVT+3(1),0(R1)  RPRINS
0002AE 4780 C2C4      002C4 288          BE    FNDUCB3         RPRINS
0002B2 4110 1005      00005 289          LA    R1,5(,R1)      NEXT DEVICE IN TABLE    RPRINS
0002B6 46E0 C2A8      002A8 290          BCT   R14,FNDUCB2    LOOP UNTIL FOUND         RPRINS
0002BA D203 D234 C5D8 00234 005D8 291          MVC   UCBDEVT(4),=C'UNKN'  RPRINS
0002C0 47F0 C2CA      002CA 292          B     ENDUNIT          RPRINS
0002C4 D203 D234 1001 00234 00001 293 FNDUCB3 MVC   UCBDEVT(4),1(R1)  MOVE UNIT NAME           RPRINS
0002CA      294 ENDUNIT  DS    0H                RPRINS
0002CA 9604 D22F      0022F 295          OI    FLAG,X'04'      NOTE THE VOLUME AS FOUND
296 *
297 ** IF OFFLINE, DO NOT PROCESS
298 *
0002CE 9140 6003      00003 299          TM    3(R6),X'40'  PENDING BIT - SHOULD BE OFF
0002D2 4710 C23E      0023E 300          BO    PENDING
0002D6 9180 6003      00003 301          TM    3(R6),X'80'  ONLINE BIT - SHOULD BE ON
0002DA 4780 C250      00250 302          BZ    OFFLINE
303 *
304 *      NOW GET DSCB'S FROM THE VOLUME
305 *
306 *
307 *      SET UP THE PARM LIST FOR VTOCEXCP
308 *
309          VTOCEXCP OPEN      OPEN THE VTOC
0002DE 9201 D214      00214 310+         MVI   VTCEFUNC,VTCPEN    SET THE FUNCTION CODE
0002E2 4110 D070      00070 311+         LA    R1,VTOCOM         POINT TO THE COMMON AREA
0002E6 58F0 D094      00094 312+         L     R15,VADEXCP       POINT TO THE ROUTINE
0002EA 05EF      313+         BALR  R14,R15           THEN CALL IT
0002EC 12FF      314          LTR   R15,R15           DID IT OPEN OK?
0002EE 4770 C396      00396 315          BNE   RETURN            NO, JUST EXIT

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

0002F2 9500 D216      00216      316 READDSCB CLI  TABFULL,0      CHECK FOR FULL TABLES
0002F6 4770 C360      00360      317          BNE  ENDVTOC      IF FULL, TRY END OF VTOC TO CLEAR
          318          VTOCEXCP READ      GET A DSCB
0002FA 9200 D214      00214      319+        MVI  VTCEFUNC,VTCREAD  SET THE FUNCTION CODE
0002FE 4110 D070      00070      320+        LA   R1,VTOCOM      POINT TO THE COMMON AREA
000302 58F0 D094      00094      321+        L    R15,VADEXCP    POINT TO THE ROUTINE
000306 05EF          322+        BALR R14,R15        THEN CALL IT
000308 49F0 C626      00626      323          CH  R15,H4          CHECK THE RETURN CODE
00030C 4780 C360      00360      324          BE  ENDVTOC      END OF VTOC
000310 4720 C396      00396      325          BH  RETURN        BAD ERROR, VTOCEXCP GAVE THE MESSAGE
          326 *
          327 *      CHECK THE DATA SET QUALIFICATIONS, LIMIT, AND, OR
          328 *
          329          VTCALL CHEK      CALL THE CHECK ROUTINE
000314 4110 D070      00070      330+        LA   R1,VTOCOM      POINT TO THE COMMON AREA
000318 58F0 D098      00098      331+        L    R15,VADCHEK    POINT TO THE ROUTINE
00031C 05EF          332+        BALR R14,R15        THEN CALL IT
00031E 12FF          333          LTR  R15,R15        DOES THIS DATA SET GET PASSED ON?
000320 4770 C2F2      002F2      334          BNZ  READDSCB      NO, GET ANOTHER
          335 *      YES, CONTINUE PROCESSING
          336 *
          337 *      FORMAT THE DSCB INFORMATION
          338 *
000324 9180 D215      00215      339          TM  VTCFMTCK,VTCFMTCD WAS FORMAT CALLED BY CHECK?
000328 4710 C33C      0033C      340          BO  CALLEXIT      YES, DON'T CALL IT AGAIN
          341          VTCALL FORM      CALL THE FORMATTING ROUTINE
00032C 4110 D070      00070      342+        LA   R1,VTOCOM      POINT TO THE COMMON AREA
000330 58F0 D09C      0009C      343+        L    R15,VADFORM    POINT TO THE ROUTINE
000334 05EF          344+        BALR R14,R15        THEN CALL IT
000336 12FF          345          LTR  R15,R15        DID IT FUNCTION?
000338 4770 C2F2      002F2      346          BNZ  READDSCB      NO, GET ANOTHER DSCB
          347 *
          348 *      CALL THE EXIT ROUTINE IF ONE WAS SPECIFIED
          349 *
          350 CALLEXIT VTCALL EXIT,TEST  CALL THE EXIT ROUTINE
00033C 4110 D070      00070      351+CALLEXIT LA   R1,VTOCOM      POINT TO THE COMMON AREA
000340 58F0 D090      00090      352+        L    R15,VADEXIT    POINT TO THE ROUTINE
000344 12FF          353+        LTR  R15,R15        SEE IF THE ROUTINE IS PRESENT
000346 4780 C34C      0034C      354+        BZ   *+6          DON'T CALL IT IF IT'S NOT THERE
00034A 05EF          355+        BALR R14,R15        THEN CALL IT
00034C 12FF          356          LTR  R15,R15        SHOULD THE DATA SET BE PASSED ON?
00034E 4770 C2F2      002F2      357          BNZ  READDSCB      NO, GET ANOTHER DSCB
          358 *
          359 *      SORT THE ENTRIES INTO THE NEW LIST
          360 *
          361          VTCALL SORT      CALL THE SORT ROUTINE
000352 4110 D070      00070      362+        LA   R1,VTOCOM      POINT TO THE COMMON AREA
000356 58F0 D0A4      000A4      363+        L    R15,VADSORT    POINT TO THE ROUTINE
00035A 05EF          364+        BALR R14,R15        THEN CALL IT
00035C 47F0 C2F2      002F2      365          B    READDSCB      GET ANOTHER DSCB
          366 *
          367 *      END OF THE VOLUME, CHECK FOR MORE
          368 *
          369 ENDVTOC VTOCEXCP CLOSE FIRST CLOSE THE VTOC
000360 9202 D214      00214      370+ENDVTOC MVI  VTCEFUNC,VTCCLOSE  SET THE FUNCTION CODE

```



```
LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25
000364 4110 D070      00070      371+      LA    R1,VTOCOM    POINT TO THE COMMON AREA
000368 58F0 D094      00094      372+      L     R15,VADEXCP   POINT TO THE ROUTINE
00036C 05EF              373+      BALR  R14,R15     THEN CALL IT
              374 *
00036E 9103 D22F      0022F      375 ENDEVOL  TM     FLAG,X'03'     IS THIS A GENERIC VOLUME SEARCH
000372 4770 C20A      0020A      376              BNZ   INCR1       YES, SEARCH FOR MORE
000376 BF37 3019      00019      377 NEXTVOL  ICM   R3,B'0111',25(R3) GET THE NEXT VOLUME FROM THE PDL
00037A 4720 C15C      0015C      378              BP    LOOP1       THERE IS ANOTHER, GET IT
              379 *
              380 *      PRINT THE SELECTED ITEMS FOR THE SELECTED DATA SETS
              381 *
00037E              382 PHASE2  DS    0H
              383              VTCALL PRNT     CALL THE PRINT ROUTINE
00037E 4110 D070      00070      384+      LA    R1,VTOCOM    POINT TO THE COMMON AREA
000382 58F0 D0A0      000A0      385+      L     R15,VADPRNT  POINT TO THE ROUTINE
000386 05EF              386+      BALR  R14,R15     THEN CALL IT
000388 47F0 C394      00394      387              B     EXIT0
```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				389 *		
				390 *	PROCESSING IS COMPLETE, EXEUNT	
				391 *		
00038C	41F0 0010	00010		392	PARMERR LA R15,16	
000390	47F0 C396	00396		393	B RETURN	
000394	1BFF			394	EXIT0 SR R15,R15	
				396	RETURN LTR R2,R15	NORMAL EXIT?
000396	122F			397	BZ RETURN1	YES, LEAVE EVERY THING ALONE
				399	LA R1,PARMLIST	AREA FOR STACK PARM LIST
00039C	4110 D0BC	000BC	00000	400	USING IOPL,R1	AN ERROR WAS FOUND, FLUSH THE STACK
0003A0	D203 1000 D0A8	00000	000A8	402	MVC IOPLUPT,ADDRUPT	
0003A6	D203 1004 D0AC	00004	000AC	403	MVC IOPECT,ADDRECT	
0003AC	4100 D0DC	000DC		404	LA R0,ATTNECB	
0003B0	9200 D0DC	000DC		405	MVI ATTNECB,0	
0003B4	5000 1008	00008		406	ST R0,IOPLECB	
				408	STACK PARM=PARMLIST+16,DELETE=ALL,MF=(E,(1))	
0003B8	41E0 D0CC	000CC		409+	LA 14,PARMLIST+16	LOAD PARM ADDR 06250000
0003BC	50E0 100C	0000C		410+	ST 14,12(0,1)	STORE PARM ADDR IN PARM LIST 06300000
0003C0	9210 E000	00000		411+	MVI 0(14),X'10'	SET OPERATION FIELD 09050000
0003C4	58F0 0010	00010		412+	L 15,16(0,0)	LOAD CVT POINTER 09361100
0003C8	9180 F1D8	001D8		413+	TM 472(15),B'10000000'	IS STACK ADDR HERE ? 09372200
0003CC	47E0 C3DA	003DA		414+	BNO IKJ@0017	NO- BRANCH TO LINK 09383300
0003D0	58FF 01D8	001D8		415+	L 15,472(15)	YES- BALR TO STACK 09394400
0003D4	05EF			416+	BALR 14,15	09405500
0003D6	47F0 C3F2	003F2		417+	B IKJ\$0017	BRANCH AROUND LINK 09416600
0003DA				418+	IKJ@0017 DS 0H	30000000
0003DA	0700			419+	CNOP 0,4	04900000
0003DC	45F0 C3F0	003F0		420+	BAL 15,*+20	BRANCH AROUND CONSTANTS 04950000
0003E0	000003E8			421+	DC A(*+8)	ADDR. OF PARM. LIST 05050000
0003E4	00000000			422+	DC A(0)	DCB ADDRESS PARAMETER 06650000
0003E8	C9D2D1E2E3C3D240			423+	DC CL8'IKJSTCK'	EP PARAMETER 06750000
0003F0	0A06			424+	SVC 6	ISSUE LINK SVC 48000000
0003F2				425+	IKJ\$0017 DS 0H	09438800
				427	TCLEARQ INPUT	CLEAR INPUT BUFFERS
0003F2	4100 0001	00001		428+	LA 0,1	SET ENTRY CODE 58000020
0003F6	4110 0080	00080		429+	LA 1,X'80'	INDICATE INPUT 62000020
0003FA	8D00 0018	00018		430+	SLDL 0,24	MOVE TO HI-ORDER BYTES 68000020
0003FE	0A5E			431+	SVC 94	ISSUE SVC 70000020
000400				433	RETURN1 DS 0H	
000400	45E0 C588	00588		434	BAL R14,FREOPDL	FREE THE PARSE STROAGE
000404	920F D218	00218		435	MVI VTCEPRNT,15	TELL PRINT TO CLEAN UP HIS ACT

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	
				436 *		CLOSE DATA SETS AND FREE MAIN STORAGE
				437	VTCALL PRNT	CALL THE PRINT ROUTINE
000408	4110 D070	00070		438+	LA R1,VTOCOM	POINT TO THE COMMON AREA
00040C	58F0 D0A0	000A0		439+	L R15,VADPRNT	POINT TO THE ROUTINE
000410	05EF			440+	BALR R14,R15	THEN CALL IT
000412	18F2			442	LR R15,R2	GET THE RETURN CODE AGAIN
				443	LEAVE EQ	
000414	182D			444+	LR 2,13	
000416	58DD 0004	00004		445+	L 13,4(13)	
00041A	90F1 D010	00010		446+	STM 15,1,16(13)	STORE RETURN REGS
				447+*	OS/V52 RELEASE 3 VERSION	-- 10/25/74
00041E	0700			448+	CNOP 0,4	
000420	47F0 C428	00428		449+	B *+8	BRANCH AROUND LENGTH
000424	01			450+	DC AL1(1)	SUBPOOL VALUE
000425	002370			451+	DC AL3(LENWORK)	LENGTH
000428	5800 C424	00424		452+	L 0,*-4	LOAD SP AND LV
00042C	4110 2000	00000		453+	LA 1,0(0,2)	LOAD AREA ADDRESS
000430	0A0A			454+	SVC 10	ISSUE FREEMAIN SVC
000432	98EC D00C	0000C		455+	LM 14,12,12(13)	RESTORE THE REGISTERS
000436	92FF D00C	0000C		456+	MVI 12(13),X'FF'	SET RETURN INDICATION
00043A	07FE			457+	BR 14	RETURN
		00000		458+R0	EQU 0	*USED BY O.S.
		00001		459+R1	EQU 1	*USED BY O.S. // ADDRESS OF PARAMETER LIST
		00002		460+R2	EQU 2	
		00003		461+R3	EQU 3	
		00004		462+R4	EQU 4	
		00005		463+R5	EQU 5	
		00006		464+R6	EQU 6	
		00007		465+R7	EQU 7	
		00008		466+R8	EQU 8	
		00009		467+R9	EQU 9	
		0000A		468+R10	EQU 10	
		0000B		469+R11	EQU 11	
		0000C		470+R12	EQU 12	
		0000D		471+R13	EQU 13	*USED BY O.S. // SAVE-AREA ADDRESS
		0000E		472+R14	EQU 14	*USED BY O.S. // RETURN ADDRESS
		0000F		473+R15	EQU 15	*USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
		0000D		474 WORKREG	EQU 13	
				475 *		
				476 *	PARSE INITIALIZATION	
				477 *		
00043C				479	PARSINIT DS 0H	
00043C	5020 D20C	0020C		480	ST R2,CPPLADDR	AND THE CPPL ADDRESS
		00000		481	USING CPPL,R2	BASE FOR COMMAND PARM LIST
000440	D203 D0A8 2004	000A8 00004		482	MVC ADDRPUPT,CPPLUPT	ADDR OF USER PROFILE TABLE
000446	D203 D0B0 2008	000B0 00008		483	MVC ADDRPSCB,CPPLPSCB	
00044C	D203 D0AC 200C	000AC 0000C		484	MVC ADDRRECT,CPPLECT	ADDR OF ENVIROMENT TABLE
000452	D203 D0B4 2000	000B4 00000		485	MVC ADDRRCBUF,CPPLCBUF	
				486	DROP R2	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				488 *		
				489 *	PUT THE WORK AREA ADDRESSES INTO THE PARM LISTS	
				490 *		
				491	WORKADDR MSG,VTCWMSG WORK AREA FOR VTOCMSG	
000458	5810 C5DC	005DC		492+	L R1,=A(WORKMSG-WORKAREA) GET THE OFFSET (OVER 4K)	
00045C	4111 D000	00000		493+	LA R1,0(R1,R13) RELOCATE IT	
000460	5010 D070	00070		494+	ST R1,VTCWMSG THEN STORE IT FOR THE ROUTINES	
				495	WORKADDR EXCP,VTCWEXCP WORK AREA FOR VTOCEXCP	
000464	5810 C5E0	005E0		496+	L R1,=A(WORKEXCP-WORKAREA) GET THE OFFSET (OVER 4K)	
000468	4111 D000	00000		497+	LA R1,0(R1,R13) RELOCATE IT	
00046C	5010 D078	00078		498+	ST R1,VTCWEXCP THEN STORE IT FOR THE ROUTINES	
				499	WORKADDR CHEK,VTCWCHEK WORK AREA FOR VTOCCHEK	
000470	5810 C5E4	005E4		500+	L R1,=A(WORKCHEK-WORKAREA) GET THE OFFSET (OVER 4K)	
000474	4111 D000	00000		501+	LA R1,0(R1,R13) RELOCATE IT	
000478	5010 D07C	0007C		502+	ST R1,VTCWCHEK THEN STORE IT FOR THE ROUTINES	
				503	WORKADDR FORM,VTCWFORM WORK AREA FOR VTOCFORM	
00047C	5810 C5E8	005E8		504+	L R1,=A(WORKFORM-WORKAREA) GET THE OFFSET (OVER 4K)	
000480	4111 D000	00000		505+	LA R1,0(R1,R13) RELOCATE IT	
000484	5010 D080	00080		506+	ST R1,VTCWFORM THEN STORE IT FOR THE ROUTINES	
				507	WORKADDR EXIT,VTCWEXIT WORK AREA FOR VTOCEXIT	
000488	5810 C5EC	005EC		508+	L R1,=A(WORKEEXIT-WORKAREA) GET THE OFFSET (OVER 4K)	
00048C	4111 D000	00000		509+	LA R1,0(R1,R13) RELOCATE IT	
000490	5010 D074	00074		510+	ST R1,VTCWEXIT THEN STORE IT FOR THE ROUTINES	
				511	WORKADDR SORT,VTCWSORT WORK AREA FOR VTOCSORT	
000494	5810 C5F0	005F0		512+	L R1,=A(WORKSORT-WORKAREA) GET THE OFFSET (OVER 4K)	
000498	4111 D000	00000		513+	LA R1,0(R1,R13) RELOCATE IT	
00049C	5010 D088	00088		514+	ST R1,VTCWSORT THEN STORE IT FOR THE ROUTINES	
				515	WORKADDR PRNT,VTCWPRNT WORK AREA FOR VTOCPRNT	
0004A0	5810 C5F4	005F4		516+	L R1,=A(WORKPRNT-WORKAREA) GET THE OFFSET (OVER 4K)	
0004A4	4111 D000	00000		517+	LA R1,0(R1,R13) RELOCATE IT	
0004A8	5010 D084	00084		518+	ST R1,VTCWPRNT THEN STORE IT FOR THE ROUTINES	
				520 *	SET UP THE ADDRESSES FOR CALLING	
				521 *		
0004AC	D21B D08C C5F8	0008C 005F8		522	MVC VADMSG(RTNADLEN),RTNADDRS MOVE IN THE ADDRESSES	
				523 *		
				524 *		
				525 *		
				526 *	BUILD PARSE PARAMETER LIST AND INVOKE	
				527 *	IKJPARS TO ANALYZE COMMAND OPERANDS	
				528 *		
0004B2				530	GOPARSE DS 0H	
0004B2	50E0 D06C	0006C		531	ST R14,R14PARSE SAVE THE RETURN ADDRESS	
0004B6	4110 D048	00048		532	LA R1,PARSELST AREA FOR PARSE PARAMETERS	
		00000		533	USING PPL,R1 BASE FOR PARSE PARAMETER LIST	
0004BA	D203 1000 D0A8	00000 000A8		535	MVC PPLUPT,ADDRUPT PASS UPT ADDRESS	
0004C0	D203 1004 D0AC	00004 000AC		536	MVC PPLECT,ADDRECT AND ECT ADDRESS	
0004C6	D203 1014 D0B4	00014 000B4		537	MVC PPLCBUF,ADDRCBUF AND COMMAND BUFFER ADDR	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	
0004CC	50D0 1018	00018		539	ST	WORKREG,PPLUWA	ALSO WORK AREA ADDR FOR VALIDITY EXITS
0004D0	4100 D0DC	000DC		541	LA	R0,ATTNECB	ECB FOR ATTN INTERRUPTS
0004D4	9200 D0DC	000DC		542	MVI	ATTNECB,0	CLEAR ECB
0004D8	5000 1008	00008		543	ST	R0,PPL ECB	PASSE TO PARSE
0004DC	4100 D0B8	000B8		545	LA	R0,ADDRANSR	PASS ADDR OF WORD WHERE PARSE
0004E0	5000 1010	00010		546	ST	R0,PPLANS	RETURNS PDL ADDRESS
0004E4	D203 100C C614	0000C 00614		548	MVC	PPLPCL,ADDRPCL	STORE PCL ADDRESS
				550		CALLTSSR EP=IKJPARS	INVOKE PARSE
				551+*			69600002
				552+*			76800002
0004EA	58F0 0010	00010		553+	L	15,CVTPTR *	GET ADDRESS OF CVT
			00018	554+	USING	CVTMAP,15 *	SET ADDRESSABILITY TO CVT
				555+*			78600002
0004EE	9180 F20C	00224		556+	TM	CVTPARS,X'80' *	IF HIGH ORDER BIT IS NOT ONE
0004F2	47E0 C500	00500		557+	BNO	TSRL0032 *	THEN DO LINK, NOT CALL
				558+*			80400002
0004F6	58F0 F20C	00224		559+	L	15,CVTPARS *	GET ADDR OF ROUTINE
0004FA	05EF			560+	BALR	14,15 *	BRANCH TO ROUTINE
0004FC	47F0 C516	00516		561+	B	TSRE0032 *	SKIP AROUND LINK
				562+*			82200002
			00500	563+TSRL0032	EQU	*	82800002
000500				564+	CNOP	0,4	83400002
000500	45F0 C514	00514		565+	BAL	15,*+20	BRANCH AROUND CONSTANTS
000504	0000050C			566+	DC	A(*+8)	ADDR. OF PARM. LIST
000508	00000000			567+	DC	A(0)	DCB ADDRESS PARAMETER
00050C	C9D2D1D7C1D9E240			568+	DC	CL8'IKJPARS'	EP PARAMETER
000514	0A06			569+	SVC	6	ISSUE LINK SVC
				570+*			48000000
			00516	571+TSRE0032	EQU	*	84600002
				572+	DROP	15	85200002
				573	DROP	R1	85800002
000516	41E0 0018	00018		575	LA	R14,MAXPARSE	RETURN CODE LIMIT
00051A	19FE			577	CR	R15,R14	VERIFY RETURN CODE WITHIN LIMITS
00051C	4720 C53C	0053C		578	BH	PARSEBAD	NO, ERROR
000520	47FF C524	00524		580	B	*+4(R15)	PROCESS RETURN CODE
000524	47F0 C580	00580		582	PARSERET B	PARSEOK	0- SUCESSFUL
000528	47F0 C578	00578		583	B	PARSEERR	4- PARSE UNABLE TO PROMPT
00052C	47F0 C578	00578		584	B	PARSEERR	8- USER ENTERED ATTENTION
000530	47F0 C53C	0053C		585	B	PARSEBAD	12- INVALID PARAMETERS
000534	47F0 C53C	0053C		586	B	PARSEBAD	16- PARSE INTERNAL FAILURE
000538	47F0 C578	00578		587	B	PARSEERR	20 - VALIDITY CHECK ERROR
			00018	588	MAXPARSE EQU	*-PARSERET	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
00053C				590	PARSEBAD DS	0H	
00053C	D201 D170 C700	00170	00700	591	MVC	MSGTEXT2+4(L'MSGPARSE),MSGPARSE	
000542	4110 D172	00172		592	LA	R1,MSGTEXT2+4+L'MSGPARSE	
000546	4EF0 D0E0	000E0		594	CVD	R15,DOUBLE	
00054A	960F D0E7	000E7		595	OI	DOUBLE+7,X'0F'	
00054E	F317 1000 D0E0	00000	000E0	596	UNPK	0(2,R1),DOUBLE	
000554	4100 D16A	0016A		598	LA	R0,MSGTEXT2-2	
000558	1B10			599	SR	R1,R0	
00055A	8910 0010	00010		600	SLL	R1,16	
00055E	5010 D16C	0016C		601	ST	R1,MSGTEXT2	
				603	VTOCMSG MSGCMDER,MSGTEXT2	PUT OUT 'COMMAND ERROR' MSG	
000562	4110 C716	00716		604+	LA	R1,MSGCMDER POINT TO THE FIRST MESSAGE	
000566	4100 D16C	0016C		605+	LA	R0,MSGTEXT2 POINT TO THE SECOND MESSAGE	
00056A	9001 D0E8	000E8		606+	STM	R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				607+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
00056E	4110 D070	00070		608+	LA	R1,VTOCOM POINT TO THE COMMON AREA	
000572	58F0 D08C	0008C		609+	L	R15,VADMSG POINT TO THE ROUTINE	
000576	05EF			610+	BALR	R14,R15 THEN CALL IT	
000578	41F0 000C	0000C		612	PARSEERR LA	R15,12 ERROR CODE 12 - COMMAND FAILED	
00057C	47F0 C582	00582		613	B	PARSERTN RETURN FROM PARSE	
000580	1BFF			615	PARSEOK SR	R15,R15 CLEAR THE RETURN CODE	
000582	58E0 D06C	0006C		616	PARSERTN L	R14,R14PARSE GET THE RETURN LOCATION	
000586	07FE			617	BR	R14 AND GET OUT OF HERE	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				620 *		
				621 *	PARSE CLEANUP ROUTINE	
				622 *		
000588				624	FREEDL DS 0H	
000588	50E0 D068		00068	626	ST R14,R14SAVE	
				628	IKJRLSA ADDRANSR	RELEASE THE STORAGE
00058C	5810 D0B8		000B8	629+	L 1,ADDRANSR	FETCH PDL ADDRESS 00080000
000590	4110 1000		00000	630+	LA 1,0(0,1)	CLEAR HIGH ORDER BYTE YM2296 00130002
000594	1211			631+	LTR 1,1	WAS A PDL RETURNED 00140000
000596	4780 C5B2		005B2	632+	BZ IKJ\$0037	IF NO SKIP FREEMAIN 00150000
00059A				633+	IKJ@0037 DS 0H	FREE PDL LOOP 00160000
00059A	58E0 1000		00000	634+	L 14,0(0,1)	LOAD NEXT AREA ADDRESS TO FREE 00170002
00059E	5800 1004		00004	635+	L 0,4(0,1)	LOAD SUBPOOL AND LENGTH YM2296 00182002
				636+*	OS/V52 RELEASE 3 VERSION -- 10/25/74	00001603
0005A2	4110 1000		00000	637+	LA 1,0(0,1)	CLEAR HI ORDER BYTE 00150802
0005A6	0A0A			638+	SVC 10	ISSUE FREEMAIN SVC 00311202
0005A8	41E0 E000		00000	639+	LA 14,0(0,14)	CLEAR HIGH ORDER BYTE YM2296 00200002
0005AC	121E			640+	LTR 1,14	ANY MORE CORE TO FREE 00210000
0005AE	4770 C59A		0059A	641+	BNZ IKJ@0037	REPEAT LOOP UNTIL DONE 00220000
0005B2				642+	IKJ\$0037 DS 0H	BRANCHED TO IF FREEMAIN SKIPPED 00230000
0005B2	D703 D0B8 D0B8		000B8 000B8	644	XC ADDRANSR,ADDRANSR	
0005B8	58E0 D068		00068	646	L R14,R14SAVE	
0005BC	07FE			647	BR R14	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				649	*		
				650	*		
				651	*	CONSTANTS	
				652	*		
				653	*		
0005C0				654		LTORG	
0005C0	00000000			655		=F'0'	
0005C4	0000000C			656		=F'12'	
0005C8	00000018			657		=F'24'	
0005CC	0000000000000000			658		=3F'0'	
0005D8	E4D5D2D5			659		=C'UNKN'	
0005DC	00000870			660		=A(WORKMSG-WORKAREA)	
0005E0	00000970			661		=A(WORKEXCP-WORKAREA)	
0005E4	00000D70			662		=A(WORKCHEK-WORKAREA)	
0005E8	00000E70			663		=A(WORKFORM-WORKAREA)	
0005EC	00001070			664		=A(WORKEEXIT-WORKAREA)	
0005F0	00001870			665		=A(WORKSORT-WORKAREA)	
0005F4	00001970			666		=A(WORKPRNT-WORKAREA)	
0005F8	00000000			667	RTNADDRS	DC	V(VTOCMSG)
0005FC	00000000			668		DC	A(0) DUMMY ENTRY FOR THE EXIT ROUTINE
000600	00000000			669		DC	V(VTOCEXCP)
000604	00000000			670		DC	V(VTOCCHEK)
000608	00000000			671		DC	V(VTOCFORM)
00060C	00000000			672		DC	V(VTOCPRNT)
000610	00000000			673		DC	V(VTOCSORT)
		0001C		674	RTNADLEN	EQU	*-RTNADDRS
000614	00001210			675	ADDRPCL	DC	A(PCLMAIN) ADDR OF MAIN PARSE CONTROL LIST
000618	0000FFFF			676	FMIN1	DC	X'0000FFFF' END OF UCB LIST
00061C	4040404040404040			677	BLANKS	DC	CL8' ' BALNKS
000624	0002			678	H2	DC	H'2'
000626	0004			679	H4	DC	H'4'
000628	0005			680	H5	DC	H'5'
				681	*		
				682	*		
				683	*		
				684	*		
				685	*		
00062A	C1D3D3			686	CHARALL	DC	CL3'ALL'
00062D	C1D3E5			687	CHARALV	DC	CL3'ALV'
000630	D200 D223 5000 00223 00000			688	MOVVOL	MVC	VOLSER(0),0(R5)
000636	D500 D223 601C 00223 0001C			689	CLCVOL	CLC	VOLSER(0),28(R6)

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
00063C	0046002BC4E2D5C1			691	SORTTABX	DC AL2(VTFDSN-VTFMT),AL2(43),CL8'DSNAME'	
000648	00120005E5D6D3E4			692		DC AL2(VTFVOLUM-VTFMT),AL2(5),CL8'VOLUME'	
000654	00040003C1D3D3D6			693		DC AL2(VTFALLOC-VTFMT),AL2(3),CL8'ALLOC'	
000660	00080003E4E2C5C4			694		DC AL2(VTFUSED-VTFMT),AL2(3),CL8'USED'	
00066C	000C0003E4D5E4E2			695		DC AL2(VTFUNUSD-VTFMT),AL2(3),CL8'UNUSED'	
000678	00100001D7C3E340			696		DC AL2(VTFPCT-VTFMT),AL2(1),CL8'PCT'	
000684	00250000C5E74040			697		DC AL2(VTFNOEPV-VTFMT),AL2(0),CL8'EX'	
000690	00260002C4E2D640			698		DC AL2(VTFDSORG-VTFMT),AL2(2),CL8'DSO'	
00069C	00290004D9C6D440			699		DC AL2(VTFRECFM-VTFMT),AL2(4),CL8'RFM'	
0006A8	002E0001D3D9C5C3			700		DC AL2(VTFLRECL-VTFMT),AL2(1),CL8'LRECL'	
0006B4	00300001C2D3D2E2			701		DC AL2(VTFBLKSZ-VTFMT),AL2(1),CL8'BLKSZ'	
0006C0	001C0002C3C4C1E3			702		DC AL2(VTFCREDT-VTFMT),AL2(2),CL8'CDATE'	
0006CC	001F0002C5E7D7C4			703		DC AL2(VTFEXPDT-VTFMT),AL2(2),CL8'EXPDT'	
0006D8	00220002D9C5C6C4			704		DC AL2(VTFLSTAC-VTFMT),AL2(2),CL8'REFDT'	
0006E4	00340000D9C1C3C6			705		DC AL2(VTFRACF-VTFMT),AL2(0),CL8'RACF'	RPRINS
0006F0	00180003E4D5C9E3			706		DC AL2(VTFUNIT-VTFMT),AL2(3),CL8'UNIT'	RPRINS
0006FC	00000000			707		DC F'0'	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				709 *		
				710 *	PROGRAM MESSAGES	
				711 *		
				713	PRINT NOGEN	
				715 MSGPARSE MSG	' PARSE ERROR CODE '	
				717 MSGCMDEP MSG	' COMMAND SYSTEM ERROR '	
				719 MSGNOTMT MSG	' VVVVVV VOLUME IS NOT MOUNTED '	
				721 MSGOFFLN MSG	' VVVVVV VOLUME IS OFFLINE '	
				723 MSGPEND MSG	' VVVVVV VOLUME IS PENDING OFFLINE '	
				725 *		
				726 *		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000795	08F2F3F1F4			728	DEVTYPS	DC X'08',CL4'2314' UCBTYP AND UNIT NAME	RPRINS
00079A	09F3F3F3F0			729		DC X'09',CL4'3330'	RPRINS
00079F	0AF3F3F4F0			730		DC X'0A',CL4'3340'	RPRINS
0007A4	0BF3F3F5F0			731		DC X'0B',CL4'3350'	RPRINS
0007A9	0CF3F3F7F5			732		DC X'0C',CL4'3375'	RPRINS
0007AE	0DF3F3F3F1			733		DC X'0D',CL4'3331'	RPRINS
0007B3	0EF3F3F8F0			734		DC X'0E',CL4'3380'	RPRINS
0007B8	0FF3F3F9F0			735		DC X'0F',CL4'3390'	RPRINS
		00008		736	DEVT#	EQU (*-DEVTYPS)/5	RPRINS
0007C0				737		DS 0F	
0007C0	004602C4			738	SORTKTAB	DC AL2(VTFDSN-VTFMT),AL2(DSNSORTE-DSNSORT)	
0007C4	0000088800000B4C			739		DC A(DSNSORT),A(DSNSORTE)	
0007CC	00120138			740		DC AL2(VTFVOLUM-VTFMT),AL2(VOLSORTE-VOLSORT)	
0007D0	00000B5800000C90			741		DC A(VOLSORT),A(VOLSORTE)	
0007D8	000800F0			742		DC AL2(VTFUSED-VTFMT),AL2(USESORTE-USESORT)	
0007DC	00000C9C00000D8C			743		DC A(USESORT),A(USESORTE)	
0007E4	000400F0			744		DC AL2(VTFALLOC-VTFMT),AL2(ALCSORTE-ALCSORT)	
0007E8	00000C9C00000D8C			745		DC A(ALCSORT),A(ALCSORTE)	
0007F0	000C00F0			746		DC AL2(VTFUNUSD-VTFMT),AL2(UNUSORTE-UNUSORT)	
0007F4	00000C9C00000D8C			747		DC A(UNUSORT),A(UNUSORTE)	
0007FC	00100078			748		DC AL2(VTFPCT-VTFMT),AL2(PCTSORTE-PCTSORT)	
000800	00000D9800000E10			749		DC A(PCTSORT),A(PCTSORTE)	
000808	0025000C			750		DC AL2(VTFNOEPV-VTFMT),AL2(EXTSORTE-EXTSORT)	
00080C	00000E1C00000E28			751		DC A(EXTSORT),A(EXTSORTE)	
000814	00260030			752		DC AL2(VTFDSORG-VTFMT),AL2(DSOSORTE-DSOSORT)	
000818	00000E3400000E64			753		DC A(DSOSORT),A(DSOSORTE)	
000820	0029006C			754		DC AL2(VTFRECFM-VTFMT),AL2(RFMSORTE-RFMSORT)	
000824	00000E7000000EDC			755		DC A(RFMSORT),A(RFMSORTE)	
00082C	002E00CC			756		DC AL2(VTFLRECL-VTFMT),AL2(LRCSORTE-LRCSORT)	
000830	00000EE800000FB4			757		DC A(LRCSORT),A(LRCSORTE)	
000838	003000CC			758		DC AL2(VTFBLKSZ-VTFMT),AL2(BLKSORTE-BLKSORT)	
00083C	00000EE800000FB4			759		DC A(BLKSORT),A(BLKSORTE)	
000844	001C01BC			760		DC AL2(VTFCREDIT-VTFMT),AL2(CDTSORTE-CDTSORT)	
000848	00000FC00000117C			761		DC A(CDTSORT),A(CDTSORTE)	
000850	002201BC			762		DC AL2(VTFLSTAC-VTFMT),AL2(RDTSORTE-RDTSORT)	
000854	00000FC00000117C			763		DC A(RDTSORT),A(RDTSORTE)	
00085C	001F01BC			764		DC AL2(VTFEXPDT-VTFMT),AL2(EDTSORTE-EDTSORT)	
000860	00000FC00000117C			765		DC A(EDTSORT),A(EDTSORTE)	
000868	00340018			766		DC AL2(VTFRACF-VTFMT),AL2(RACSORTE-RACSORT)	RPRINS
00086C	00001188000011A0			767		DC A(RACSORT),A(RACSORTE)	RPRINS
000874	0018006C			768		DC AL2(VTFUNIT-VTFMT),AL2(UNTSORTE-UNTSORT)	RPRINS
000878	000011A00000120C			769		DC A(UNTSORT),A(UNTSORTE)	RPRINS
000880	0000000000000000			770		DC 2F'0'	
				772	*	Each entry in the tables below is 12 bytes long	RPRINS
				773	*	Bytes 0-3: contains the address of the next entry	RPRINS
				774	*	Bytes 4-5: contains the length of the sort field - 1	RPRINS
				775	*	Bytes 6-11: contains the value of the sort field	RPRINS
				776	*	If shorter the field is padded with zeroes	RPRINS
				777	*	or blanks.	RPRINS
000888	0000000000000E940			778	DSNSORT	DC A(0),AL2(0),CL6'Z'	
000894	0000000000001E3E5			779		DC A(0),AL2(1),CL6'TV'	
0008A0	0000000000001E3D4			780		DC A(0),AL2(1),CL6'TM'	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
0008AC	0000000000002E34B			781	DC	A(0),AL2(2),CL6'T.Z'	
0008B8	0000000000002E34B			782	DC	A(0),AL2(2),CL6'T.Y'	
0008C4	0000000000002E34B			783	DC	A(0),AL2(2),CL6'T.X'	
0008D0	0000000000002E34B			784	DC	A(0),AL2(2),CL6'T.W'	
0008DC	0000000000002E34B			785	DC	A(0),AL2(2),CL6'T.V'	
0008E8	0000000000002E34B			786	DC	A(0),AL2(2),CL6'T.U'	
0008F4	0000000000002E34B			787	DC	A(0),AL2(2),CL6'T.T'	
000900	0000000000002E34B			788	DC	A(0),AL2(2),CL6'T.S'	
00090C	0000000000002E34B			789	DC	A(0),AL2(2),CL6'T.R'	
000918	0000000000002E34B			790	DC	A(0),AL2(2),CL6'T.Q'	
000924	0000000000002E34B			791	DC	A(0),AL2(2),CL6'T.P'	
000930	0000000000002E34B			792	DC	A(0),AL2(2),CL6'T.O'	
00093C	0000000000002E34B			793	DC	A(0),AL2(2),CL6'T.N'	
000948	0000000000002E34B			794	DC	A(0),AL2(2),CL6'T.M'	
000954	0000000000002E34B			795	DC	A(0),AL2(2),CL6'T.L'	
000960	0000000000002E34B			796	DC	A(0),AL2(2),CL6'T.K'	
00096C	0000000000002E34B			797	DC	A(0),AL2(2),CL6'T.J'	
000978	0000000000002E34B			798	DC	A(0),AL2(2),CL6'T.I'	
000984	0000000000002E34B			799	DC	A(0),AL2(2),CL6'T.H'	
000990	0000000000002E34B			800	DC	A(0),AL2(2),CL6'T.G'	
00099C	0000000000002E34B			801	DC	A(0),AL2(2),CL6'T.F'	
0009A8	0000000000002E34B			802	DC	A(0),AL2(2),CL6'T.E'	
0009B4	0000000000002E34B			803	DC	A(0),AL2(2),CL6'T.D'	
0009C0	0000000000002E34B			804	DC	A(0),AL2(2),CL6'T.C'	
0009CC	0000000000002E34B			805	DC	A(0),AL2(2),CL6'T.B'	
0009D8	0000000000002E34B			806	DC	A(0),AL2(2),CL6'T.A'	
0009E4	0000000000001E2E8			807	DC	A(0),AL2(1),CL6'SY'	
0009F0	0000000000001E2E5			808	DC	A(0),AL2(1),CL6'SV'	
0009FC	0000000000001D7E5			809	DC	A(0),AL2(1),CL6'PV'	
000A08	0000000000002D74B			810	DC	A(0),AL2(2),CL6'P.Z'	
000A14	0000000000002D74B			811	DC	A(0),AL2(2),CL6'P.Y'	
000A20	0000000000002D74B			812	DC	A(0),AL2(2),CL6'P.X'	
000A2C	0000000000002D74B			813	DC	A(0),AL2(2),CL6'P.W'	
000A38	0000000000002D74B			814	DC	A(0),AL2(2),CL6'P.V'	
000A44	0000000000002D74B			815	DC	A(0),AL2(2),CL6'P.U'	
000A50	0000000000002D74B			816	DC	A(0),AL2(2),CL6'P.T'	
000A5C	0000000000002D74B			817	DC	A(0),AL2(2),CL6'P.S'	
000A68	0000000000002D74B			818	DC	A(0),AL2(2),CL6'P.R'	
000A74	0000000000002D74B			819	DC	A(0),AL2(2),CL6'P.Q'	
000A80	0000000000002D74B			820	DC	A(0),AL2(2),CL6'P.P'	
000A8C	0000000000002D74B			821	DC	A(0),AL2(2),CL6'P.O'	
000A98	0000000000002D74B			822	DC	A(0),AL2(2),CL6'P.N'	
000AA4	0000000000002D74B			823	DC	A(0),AL2(2),CL6'P.M'	
000AB0	0000000000002D74B			824	DC	A(0),AL2(2),CL6'P.L'	
000ABC	0000000000002D74B			825	DC	A(0),AL2(2),CL6'P.K'	
000AC8	0000000000002D74B			826	DC	A(0),AL2(2),CL6'P.J'	
000AD4	0000000000002D74B			827	DC	A(0),AL2(2),CL6'P.I'	
000AE0	0000000000002D74B			828	DC	A(0),AL2(2),CL6'P.H'	
000AEC	0000000000002D74B			829	DC	A(0),AL2(2),CL6'P.G'	
000AF8	0000000000002D74B			830	DC	A(0),AL2(2),CL6'P.F'	
000B04	0000000000002D74B			831	DC	A(0),AL2(2),CL6'P.E'	
000B10	0000000000002D74B			832	DC	A(0),AL2(2),CL6'P.D'	
000B1C	0000000000002D74B			833	DC	A(0),AL2(2),CL6'P.C'	
000B28	0000000000002D74B			834	DC	A(0),AL2(2),CL6'P.B'	
000B34	0000000000002D74B			835	DC	A(0),AL2(2),CL6'P.A'	

ASM 0201 13.38 01/07/25

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
000B40	000000000000D540			836	DC	A(0),AL2(0),CL6'N'
000B4C	0000000000004040			837	DSNSORTE DC	A(0),AL2(0),CL6' '
000B58	000000000004F3F3			839	VOLSORT DC	A(0),AL2(4),CL6'33509'
000B64	000000000004F3F3			840	DC	A(0),AL2(4),CL6'33508'
000B70	000000000004F3F3			841	DC	A(0),AL2(4),CL6'33507'
000B7C	000000000004F3F3			842	DC	A(0),AL2(4),CL6'33506'
000B88	000000000004F3F3			843	DC	A(0),AL2(4),CL6'33505'
000B94	000000000004F3F3			844	DC	A(0),AL2(4),CL6'33504'
000BA0	000000000004F3F3			845	DC	A(0),AL2(4),CL6'33503'
000BAC	000000000004F3F3			846	DC	A(0),AL2(4),CL6'33502'
000BB8	000000000004F3F3			847	DC	A(0),AL2(4),CL6'33501'
000BC4	000000000004F3F3			848	DC	A(0),AL2(4),CL6'33500'
000BD0	000000000004F3F3			849	DC	A(0),AL2(4),CL6'33309'
000BDC	000000000004F3F3			850	DC	A(0),AL2(4),CL6'33308'
000BE8	000000000004F3F3			851	DC	A(0),AL2(4),CL6'33307'
000BF4	000000000004F3F3			852	DC	A(0),AL2(4),CL6'33306'
000C00	000000000004F3F3			853	DC	A(0),AL2(4),CL6'33305'
000C0C	000000000004F3F3			854	DC	A(0),AL2(4),CL6'33304'
000C18	000000000004F3F3			855	DC	A(0),AL2(4),CL6'33303'
000C24	000000000004F3F3			856	DC	A(0),AL2(4),CL6'33302'
000C30	000000000004F3F3			857	DC	A(0),AL2(4),CL6'33301'
000C3C	000000000004F3F3			858	DC	A(0),AL2(4),CL6'33300'
000C48	000000000000E340			859	DC	A(0),AL2(0),CL6'T'
000C54	000000000000D940			860	DC	A(0),AL2(0),CL6'R'
000C60	000000000000D740			861	DC	A(0),AL2(0),CL6'P'
000C6C	000000000000D440			862	DC	A(0),AL2(0),CL6'M'
000C78	000000000000C940			863	DC	A(0),AL2(0),CL6'I'
000C84	000000000000C840			864	DC	A(0),AL2(0),CL6'H'
000C90	0000000000004040			865	VOLSORTE DC	A(0),AL2(0),CL6' '
000C9C				867	USESORT DS	0F
000C9C				868	UNUSORT DS	0F
000C9C	0000000000030000			869	ALCSORT DC	A(0),AL2(3),XL4'0000F000',XL2'00'
000CA8	0000000000030000			870	DC	A(0),AL2(3),XL4'0000C000',XL2'00'
000CB4	0000000000030000			871	DC	A(0),AL2(3),XL4'0000A000',XL2'00'
000CC0	0000000000030000			872	DC	A(0),AL2(3),XL4'00008000',XL2'00'
000CCC	0000000000030000			873	DC	A(0),AL2(3),XL4'00006000',XL2'00'
000CD8	0000000000030000			874	DC	A(0),AL2(3),XL4'00005000',XL2'00'
000CE4	0000000000030000			875	DC	A(0),AL2(3),XL4'00004000',XL2'00'
000CF0	0000000000030000			876	DC	A(0),AL2(3),XL4'00003000',XL2'00'
000CFC	0000000000030000			877	DC	A(0),AL2(3),XL4'00002000',XL2'00'
000D08	0000000000030000			878	DC	A(0),AL2(3),XL4'00001000',XL2'00'
000D14	0000000000030000			879	DC	A(0),AL2(3),XL4'00000C00',XL2'00'
000D20	0000000000030000			880	DC	A(0),AL2(3),XL4'00000800',XL2'00'
000D2C	0000000000030000			881	DC	A(0),AL2(3),XL4'00000400',XL2'00'
000D38	0000000000030000			882	DC	A(0),AL2(3),XL4'00000300',XL2'00'
000D44	0000000000030000			883	DC	A(0),AL2(3),XL4'00000200',XL2'00'
000D50	0000000000030000			884	DC	A(0),AL2(3),XL4'00000100',XL2'00'
000D5C	0000000000030000			885	DC	A(0),AL2(3),XL4'000000C0',XL2'00'
000D68	0000000000030000			886	DC	A(0),AL2(3),XL4'00000080',XL2'00'

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000D74	00000000000030000			887	DC	A(0),AL2(3),XL4'00000040',XL2'00'	
000D80	00000000000030000			888	DC	A(0),AL2(3),XL4'00000010',XL2'00'	
000D8C				889	USESORTE DS	0F	
000D8C				890	UNUSORTE DS	0F	
000D8C	00000000000030000			891	ALCSORTE DC	A(0),AL2(3),XL6'00'	
000D98	00000000000010064			893	PCTSORT DC	A(0),AL2(1),XL2'0064',XL4'00'	
000DA4	0000000000001005A			894	DC	A(0),AL2(1),XL2'005A',XL4'00'	
000DB0	00000000000010050			895	DC	A(0),AL2(1),XL2'0050',XL4'00'	
000DBC	00000000000010046			896	DC	A(0),AL2(1),XL2'0046',XL4'00'	
000DC8	0000000000001003C			897	DC	A(0),AL2(1),XL2'003C',XL4'00'	
000DD4	00000000000010032			898	DC	A(0),AL2(1),XL2'0032',XL4'00'	
000DE0	00000000000010028			899	DC	A(0),AL2(1),XL2'0028',XL4'00'	
000DEC	0000000000001001E			900	DC	A(0),AL2(1),XL2'001E',XL4'00'	
000DF8	00000000000010014			901	DC	A(0),AL2(1),XL2'0014',XL4'00'	
000E04	0000000000001000A			902	DC	A(0),AL2(1),XL2'000A',XL4'00'	
000E10	00000000000010000			903	PCTSORTE DC	A(0),AL2(1),XL6'00'	
000E1C	0000000000000F040			905	EXTSORT DC	A(0),AL2(0),CL6'0'	
000E28	0000000000000F040			906	EXTSORTE DC	A(0),AL2(0),CL6'0'	
000E34	0000000000001E5E2			908	DSOSORT DC	A(0),AL2(1),CL6'VS'	
000E40	0000000000001D7E2			909	DC	A(0),AL2(1),CL6'PS'	
000E4C	0000000000001D7D6			910	DC	A(0),AL2(1),CL6'PO'	
000E58	0000000000001C4C1			911	DC	A(0),AL2(1),CL6'DA'	
000E64	00000000000014040			912	DSOSORTE DC	A(0),AL2(1),CL6' '	
000E70	0000000000001E5E2			914	RFMSORT DC	A(0),AL2(1),CL6'VS'	
000E7C	0000000000002E5C2			915	DC	A(0),AL2(2),CL6'VBS'	
000E88	0000000000001E5C2			916	DC	A(0),AL2(1),CL6'VB'	
000E94	0000000000000E540			917	DC	A(0),AL2(0),CL6'V'	
000EA0	0000000000000E440			918	DC	A(0),AL2(0),CL6'U'	
000EAC	0000000000001C6E2			919	DC	A(0),AL2(1),CL6'FS'	
000EB8	0000000000002C6C2			920	DC	A(0),AL2(2),CL6'FBS'	
000EC4	0000000000001C6C2			921	DC	A(0),AL2(1),CL6'FB'	
000ED0	0000000000000C640			922	DC	A(0),AL2(0),CL6'F'	
000EDC	00000000000004040			923	RFMSORTE DC	A(0),AL2(0),CL6' '	
000EE8				925	LRCSORT DS	0F	
000EE8	00000000000014650			926	BLKSORT DC	A(0),AL2(1),XL2'4650',XL4'00'	
000EF4	00000000000013A98			927	DC	A(0),AL2(1),XL2'3A98',XL4'00'	
000F00	00000000000012EE0			928	DC	A(0),AL2(1),XL2'2EE0',XL4'00'	
000F0C	00000000000012328			929	DC	A(0),AL2(1),XL2'2328',XL4'00'	
000F18	00000000000011770			930	DC	A(0),AL2(1),XL2'1770',XL4'00'	
000F24	00000000000010BB8			931	DC	A(0),AL2(1),XL2'0BB8',XL4'00'	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000F30	000000000000107D0			932	DC	A(0),AL2(1),XL2'07D0',XL4'00'	
000F3C	00000000000010640			933	DC	A(0),AL2(1),XL2'0640',XL4'00'	
000F48	000000000000104B0			934	DC	A(0),AL2(1),XL2'04B0',XL4'00'	
000F54	00000000000010320			935	DC	A(0),AL2(1),XL2'0320',XL4'00'	
000F60	00000000000010258			936	DC	A(0),AL2(1),XL2'0258',XL4'00'	
000F6C	00000000000010190			937	DC	A(0),AL2(1),XL2'0190',XL4'00'	
000F78	000000000000100C8			938	DC	A(0),AL2(1),XL2'00C8',XL4'00'	
000F84	000000000000100A0			939	DC	A(0),AL2(1),XL2'00A0',XL4'00'	
000F90	00000000000010078			940	DC	A(0),AL2(1),XL2'0078',XL4'00'	
000F9C	00000000000010050			941	DC	A(0),AL2(1),XL2'0050',XL4'00'	
000FA8	00000000000010028			942	DC	A(0),AL2(1),XL2'0028',XL4'00'	
000FB4				943	BLKSORTE DS	0F	
000FB4	00000000000010000			944	LRCSORTE DC	A(0),AL2(1),XL6'00'	
000FC0				946	CDTSORT DS	0F	
000FC0				947	EDTSORT DS	0F	
000FC0	00000000000026300			948	RDTSORT DC	A(0),AL2(2),AL1(99),AL2(0),XL3'00'	
000FCC	00000000000025301			949	DC	A(0),AL2(2),AL1(83),AL2(300),XL3'00'	
000FD8	00000000000025300			950	DC	A(0),AL2(2),AL1(83),AL2(200),XL3'00'	
000FE4	00000000000025300			951	DC	A(0),AL2(2),AL1(83),AL2(100),XL3'00'	
000FF0	00000000000025300			952	DC	A(0),AL2(2),AL1(83),AL2(000),XL3'00'	
000FFC	00000000000025201			953	DC	A(0),AL2(2),AL1(82),AL2(300),XL3'00'	
001008	00000000000025200			954	DC	A(0),AL2(2),AL1(82),AL2(200),XL3'00'	
001014	00000000000025200			955	DC	A(0),AL2(2),AL1(82),AL2(100),XL3'00'	
001020	00000000000025200			956	DC	A(0),AL2(2),AL1(82),AL2(000),XL3'00'	
00102C	00000000000025101			957	DC	A(0),AL2(2),AL1(81),AL2(300),XL3'00'	
001038	00000000000025100			958	DC	A(0),AL2(2),AL1(81),AL2(200),XL3'00'	
001044	00000000000025100			959	DC	A(0),AL2(2),AL1(81),AL2(100),XL3'00'	
001050	00000000000025100			960	DC	A(0),AL2(2),AL1(81),AL2(000),XL3'00'	
00105C	00000000000025001			961	DC	A(0),AL2(2),AL1(80),AL2(300),XL3'00'	
001068	00000000000025000			962	DC	A(0),AL2(2),AL1(80),AL2(200),XL3'00'	
001074	00000000000025000			963	DC	A(0),AL2(2),AL1(80),AL2(100),XL3'00'	
001080	00000000000025000			964	DC	A(0),AL2(2),AL1(80),AL2(000),XL3'00'	
00108C	00000000000024F01			965	DC	A(0),AL2(2),AL1(79),AL2(300),XL3'00'	
001098	00000000000024F00			966	DC	A(0),AL2(2),AL1(79),AL2(200),XL3'00'	
0010A4	00000000000024F00			967	DC	A(0),AL2(2),AL1(79),AL2(100),XL3'00'	
0010B0	00000000000024F00			968	DC	A(0),AL2(2),AL1(79),AL2(000),XL3'00'	
0010BC	00000000000024E01			969	DC	A(0),AL2(2),AL1(78),AL2(300),XL3'00'	
0010C8	00000000000024E00			970	DC	A(0),AL2(2),AL1(78),AL2(200),XL3'00'	
0010D4	00000000000024E00			971	DC	A(0),AL2(2),AL1(78),AL2(100),XL3'00'	
0010E0	00000000000024E00			972	DC	A(0),AL2(2),AL1(78),AL2(000),XL3'00'	
0010EC	00000000000024D01			973	DC	A(0),AL2(2),AL1(77),AL2(300),XL3'00'	
0010F8	00000000000024D00			974	DC	A(0),AL2(2),AL1(77),AL2(200),XL3'00'	
001104	00000000000024D00			975	DC	A(0),AL2(2),AL1(77),AL2(100),XL3'00'	
001110	00000000000024D00			976	DC	A(0),AL2(2),AL1(77),AL2(000),XL3'00'	
00111C	00000000000024C01			977	DC	A(0),AL2(2),AL1(76),AL2(300),XL3'00'	
001128	00000000000024C00			978	DC	A(0),AL2(2),AL1(76),AL2(200),XL3'00'	
001134	00000000000024C00			979	DC	A(0),AL2(2),AL1(76),AL2(100),XL3'00'	
001140	00000000000024C00			980	DC	A(0),AL2(2),AL1(76),AL2(000),XL3'00'	
00114C	00000000000024B01			981	DC	A(0),AL2(2),AL1(75),AL2(300),XL3'00'	
001158	00000000000024B00			982	DC	A(0),AL2(2),AL1(75),AL2(200),XL3'00'	
001164	00000000000024B00			983	DC	A(0),AL2(2),AL1(75),AL2(100),XL3'00'	
001170	00000000000024B00			984	DC	A(0),AL2(2),AL1(75),AL2(000),XL3'00'	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
00117C				985	EDTSORTE DS	0F	
00117C				986	RDTSORTE DS	0F	
00117C	00000000000020000			987	CDTSORTE DC	A(0),AL2(2),XL6'00'	
				988	*		RPRINS
001188	0000000000000D500			989	RACSORT DC	A(0),AL2(0),C'N',XL5'00'	RPRINS
001194	0000000000000E800			990		A(0),AL2(0),C'Y',XL5'00'	RPRINS
0011A0				991	RACSORTE DS	0F	RPRINS
				992	*		RPRINS
0011A0	0000000000003F3F3			993	UNTSORT DC	A(0),AL2(3),CL4'3390',XL2'00'	RPRINS
0011AC	0000000000003F3F3			994		A(0),AL2(3),CL4'3380',XL2'00'	RPRINS
0011B8	0000000000003F3F3			995		A(0),AL2(3),CL4'3375',XL2'00'	RPRINS
0011C4	0000000000003F3F3			996		A(0),AL2(3),CL4'3350',XL2'00'	RPRINS
0011D0	0000000000003F3F3			997		A(0),AL2(3),CL4'3340',XL2'00'	RPRINS
0011DC	0000000000003F3F3			998		A(0),AL2(3),CL4'3331',XL2'00'	RPRINS
0011E8	0000000000003F3F3			999		A(0),AL2(3),CL4'3330',XL2'00'	RPRINS
0011F4	0000000000003F2F3			1000		A(0),AL2(3),CL4'2314',XL2'00'	RPRINS
001200	0000000000003E4D5			1001		A(0),AL2(3),CL4'UNKN',XL2'00'	RPRINS
00120C				1002	UNTSORTE DS	0F	RPRINS

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				1004 *		
				1005 *		
				1006 *	P A R S E C O N T R O L L I S T	
				1007 *		
				1008 *		
				1010	COPY VTOCPARS	
				1011	PUSH PRINT	
				1012	PRINT NOGEN	
				1013 PCLMAIN	IKJPARM DSECT=PDL	
				1023 VOLS	IKJPOSIT DSNAME,VOLSER,LIST, PROMPT='VOLUMES TO SEARCH AND OTHER PARAMETERS', HELP=('VOLUME SERIAL NUMBERS WHICH ARE TO BE SEARCHED FOR DATA SETS TO LIST')	\$ \$
				1043 LEVKEY	IKJKEYWD	
				1054	IKJNAME 'LEVEL',SUBFLD=SUBLEV	
				1065 ENDKEY	IKJKEYWD	
				1075	IKJNAME 'ENDING',SUBFLD=SUBEND	
				1086 CONTAINK	IKJKEYWD	
				1096	IKJNAME 'CONTAINING',SUBFLD=SUBCONT	
				1108 SPACEK	IKJKEYWD DEFAULT='TRKS'	
				1120	IKJNAME 'TRKS'	
				1130 CATK	IKJKEYWD	
				1140	IKJNAME 'CAT'	
				1150 SORTK	IKJKEYWD	
				1160	IKJNAME 'SORT',SUBFLD=SUBSORTS	
				1170	IKJNAME 'NOSORT'	
				1180 BREAKK	IKJKEYWD	
				1190	IKJNAME 'BREAK',SUBFLD=SUBBREAK	
				1201 LIMITK	IKJKEYWD	
				1211	IKJNAME 'LIMIT',SUBFLD=SUBLIMIT	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				1222	ANDOR1K	IKJKEYWD	
				1232		IKJNAME 'AND1',SUBFLD=SUBAO1,ALIAS='AND'	
				1246		IKJNAME 'OR1',SUBFLD=SUBAO1,ALIAS='OR'	
				1261	ANDOR2K	IKJKEYWD	
				1271		IKJNAME 'AND2',SUBFLD=SUBAO2	
				1281		IKJNAME 'OR2',SUBFLD=SUBAO2	
				1292	ANDOR3K	IKJKEYWD	
				1302		IKJNAME 'AND3',SUBFLD=SUBAO3	
				1312		IKJNAME 'OR3',SUBFLD=SUBAO3	
				1323	PRINTK	IKJKEYWD	
				1333		IKJNAME 'PRINT',SUBFLD=SUBPRINT	
				1343		IKJNAME 'NOPRINT'	
				1353	CHARSK	IKJKEYWD	
				1363		IKJNAME 'CHARS',SUBFLD=SUBCHARS	
				1374	LINESK	IKJKEYWD	
				1384		IKJNAME 'LINES',SUBFLD=SUBLINES	
				1395	HEADK	IKJKEYWD	
				1405		IKJNAME 'HEADING',SUBFLD=SUBHEAD	
				1415		IKJNAME 'NOHEADING'	
				1425	TOTALK	IKJKEYWD	
				1435		IKJNAME 'TOTALS',SUBFLD=SUBTOTAL	
				1446	OUTPUTK	IKJKEYWD	
				1456		IKJNAME 'OUTPUT'	
				1466	FORMATK	IKJKEYWD	
				1476		IKJNAME 'FORMAT',SUBFLD=SUBFORMT	
				1487	DSNPLNK	IKJKEYWD	
				1497		IKJNAME 'DSNLEN',SUBFLD=SUBDSNLN	
				1508	SUBLEV	IKJSUBF	
				1513	LEVEL	IKJPOSIT DSNAME,LIST, PROMPT='BEGINNING CHARACTERS OF DSNAME TO PROCESS'	X

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				1528	SUBEND	IKJSUBF	
				1534	ENDING	IKJPOSIT DSNAME,LIST, PROMPT='ENDING CHARACTERS OF DSNAME TO PROCESS'	X
				1549	SUBCONT	IKJSUBF	
				1555	CONTAIN	IKJPOSIT DSNAME,LIST, PROMPT='CHARACTER STRING CONTAINED IN DSNAME TO PROCESSX ,	X
				1570	SUBSORTS	IKJSUBF	
				1576	SUBSORT	IKJIDENT 'SORT FIELDS',LIST,FIRST=ALPHA,MAXLNTH=6	
				1597	SUBBREAK	IKJSUBF	
				1603	BREAK	IKJIDENT 'NUMBER OF CHARACTERS FOR A BREAK',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=2,DEFAULT='3'	X
				1626	SUBCHARS	IKJSUBF	
				1632	CHARSPL	IKJIDENT 'NUMBER OF CHARACTERS PER LINE',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=3	X
				1652	BLKSZSET	IKJIDENT 'PHYSICAL BLOCK SIZE',FIRST=NUMERIC,OTHER=NUMERIC, MAXLNTH=5	X
				1673	SUBLINES	IKJSUBF	
				1679	LINESPP	IKJIDENT 'NUMBER OF LINES PER PAGE',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=3	X
				1700	SUBPRINT	IKJSUBF	
				1706	SUBPRTKY	IKJIDENT 'ADD, REP, NEW, OR DEL', FIRST=ALPHA,OTHER=ALPHA,MAXLNTH=3	X
				1726	SUBPRTIT	IKJIDENT 'ITEMS TO PRINT',LIST,FIRST=ALPHA,MAXLNTH=6	
				1747	SUBHEAD	IKJSUBF	
				1753	HEADING	IKJPOSIT QSTRING	
				1765	SUBTOTAL	IKJSUBF	
				1771	TOTALN	IKJIDENT 'NUMBER OF CHARACTERS FOR TOTALS',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=2,DEFAULT='0'	X
				1794	SUBLIMIT	IKJSUBF	
				1800	SUBLKEY	IKJIDENT 'FIELD IN DATA SET CONTROL BLOCK TO COMPARE', FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8	X
				1821	SUBLOPER	IKJIDENT 'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA, MAXLNTH=2,	X

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
					PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE '	
1844	SUBLVALU			IKJIDENT	'COMPARISON VALUE',FIRST=ALPHANUM, OTHER=ALPHANUM, PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'	X X
1866	SUBAO1			IKJSUBF		
1872	SUB1KEY			IKJIDENT	'FIELD IN DATA SET CONTROL BLOCK TO COMPARE', FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8	X
1893	SUB1OPER			IKJIDENT	'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA, MAXLNTH=2, PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE '	X X
1916	SUB1VALU			IKJIDENT	'COMPARISON VALUE',FIRST=ALPHANUM, OTHER=ALPHANUM, PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'	X X
1938	SUBAO2			IKJSUBF		
1944	SUB2KEY			IKJIDENT	'FIELD IN DATA SET CONTROL BLOCK TO COMPARE', FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8	X
1965	SUB2OPER			IKJIDENT	'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA, MAXLNTH=2, PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE '	X X
1988	SUB2VALU			IKJIDENT	'COMPARISON VALUE',FIRST=ALPHANUM, OTHER=ALPHANUM, PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'	X X
2010	SUBAO3			IKJSUBF		
2016	SUB3KEY			IKJIDENT	'FIELD IN DATA SET CONTROL BLOCK TO COMPARE', FIRST=ALPHA,OTHER=ALPHANUM,MAXLNTH=8	X
2037	SUB3OPER			IKJIDENT	'OPERATOR FOR COMPARISON',FIRST=ALPHA,OTHER=ALPHA, MAXLNTH=2, PROMPT=' OPERATORS ARE EQ, NE, LT, LE, GT, AND GE '	X X
2060	SUB3VALU			IKJIDENT	'COMPARISON VALUE',FIRST=ALPHANUM, OTHER=ALPHANUM, PROMPT='VALUE TO COMPARE FOR DATA SET KEYWORDS'	X X
2082	SUBFORMT			IKJSUBF		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				2088	FORMATSP IKJIDENT 'FORMAT TYPES TO OUTPUT',LIST,FIRST=NUMERIC, MAXLNTH=1,DEFAULT='1'	X
				2111	SUBDSNLN IKJSUBF	
				2117	DSNPLN IKJIDENT 'LENGTH OF DSNAME TO PRINT',FIRST=NUMERIC, OTHER=NUMERIC,MAXLNTH=2,DEFAULT='44'	X
				2139	DSNLNTYP IKJKEYWD DEFAULT='TRUNCATE'	
				2152	IKJNAME 'TRUNCATE'	
				2161	IKJNAME 'MULTILINE'	
				2170	IKJENDP	
				2180	POP PRINT	
	00001			2181	ACTION EQU 1	
	00002			2182	VOLUME EQU 2	
	00003			2183	CDATE EQU 3	
	00004			2184	LSTUS EQU 4	
	00005			2185	EXPDT EQU 5	
	00006			2186	ALLOC EQU 6	
	00007			2187	UNUSED EQU 7	
	00008			2188	PCT EQU 8	
	00009			2189	EXT EQU 9	
	0000A			2190	DSORG EQU 10	
	0000B			2191	RECFM EQU 11	
	0000C			2192	BLKSZ EQU 12	
	0000D			2193	LRECL EQU 13	
	0000E			2194	PASS EQU 14	
	0000F			2195	CAT EQU 15	
	00010			2196	SECT EQU 16	
	00011			2197	SECQ EQU 17	
	00012			2198	UNIT EQU 18	
	00013			2199	ROUND EQU 19	
	00014			2200	TYPE EQU 20	
	00015			2201	USED EQU 21	
	00016			2202	CCHH EQU 22	
	00017			2203	UPD EQU 23	TANSKY
	00018			2204	RACF EQU 24	TANSKY
	00019			2205	DUMMY5 EQU 25	
	0001A			2206	DSNAME EQU 26	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25

                                           2208 *
                                           2209 *      DYNAMIC WORK AREA
                                           2210 *

000000      2212 WORKAREA DSECT
000000      2213 MAINSAVE DS      18A

                                           2215      VTOCEXCP EQ      DEFINE VTOCEXCP CODES

000048      2220 PARSELST DS      8A      AREA FOR PARSE PARAMETER LIST

000068      2222 R14SAVE DS      A
00006C      2223 R14PARSE DS      A
                                           2224 *
                                           2225 *      VTOC COMMAND COMMON AREA
                                           2226 *
                                           2227      PRINT GEN
                                           2228      VTOCOM NODSECT
                                           2229+*
                                           2230+*      THIS IS THE VTOC COMMAND COMMON AREA
                                           2231+*
000070      2232+VTOCOM DS      0D
                                           2233+*
                                           2234+*      WORKING STORAGE AREAS FOR THE VARIOUS ROUTINES
                                           2235+*
000070      2236+VTCWMSG DS      A      WORKING STORAGE FOR THE MSG ROUTINE
000074      2237+VTCWEXIT DS      A      WORKING STORAGE FOR THE EXIT ROUTINE
000078      2238+VTCWEXCP DS      A      WORKING STORAGE FOR THE EXCP ROUTINE
00007C      2239+VTCWCHEK DS      A      WORKING STORAGE FOR THE CHEK ROUTINE
000080      2240+VTCWFORM DS      A      WORKING STORAGE FOR THE FORM ROUTINE
000084      2241+VTCWPRNT DS      A      WORKING STORAGE FOR THE PRNT ROUTINE
000088      2242+VTCWSORT DS      A      WORKING STORAGE FOR THE SORT ROUTINE
                                           2243+*
                                           2244+*      ADDRESSES OF THE ROUTINES
                                           2245+*
00008C 00000000      2246+VADMSG DC      V(VTOCMSG)      ADDRESS OF THE MESSAGE ROUTINE
000090 00000000      2247+VADEXIT DC      V(VTOCEXIT)      ADDRESS OF THE EXIT ROUTINE
000094 00000000      2248+VADEXCP DC      V(VTOCEXIT)      ADDRESS OF THE EXCP ROUTINE
000098 00000000      2249+VADCHEK DC      V(VTOCEXIT)      ADDRESS OF THE CHECK ROUTINE
00009C 00000000      2250+VADFORM DC      V(VTOCEXIT)      ADDRESS OF THE FORMAT ROUTINE
0000A0 00000000      2251+VADPRNT DC      V(VTOCEXIT)      ADDRESS OF THE PRINT ROUTINE
0000A4 00000000      2252+VADSORT DC      V(VTOCEXIT)      ADDRESS OF THE SORT ROUTINE
                                           2253+*
                                           2254+*      TSO COMMAND PROCESSOR AND PARSE DATA
                                           2255+*
0000A8      2256+ADDRUPT DS      A      USER PROFILE TABLE
0000AC      2257+ADDRECT DS      A      ENVIRONMENT CONTROL TABLE
0000B0      2258+ADDRPSCB DS      A      PROTECTED STEP CONTROL BLOCK
0000B4      2259+ADDRCBUF DS      A      COMMAND BUFFER

0000B8      2261+ADDRANSR DS      A      PARSE ANSWER OR PDL ADDRESS

0000BC      2263+PARMLIST DS      8A      INTERNAL PARM AREA ( MSG )

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
0000DC				2264+	ATTNECB DS	F	ECB FOR ATTENTIONS
0000E0				2265+	DOUBLE DS	D	
0000E8				2267+	MSGADDRS DS	2A	ADDRESSES OF MESSAGES FOR VTOCMSG
0000F0				2268+	MSGTEXT1 DS	XL124	
00016C				2269+	MSGTEXT2 DS	XL124	
0001E8				2271+	PUTOLD1 DS	3F	
0001F4				2272+	PUTOLD2 DS	3F	
				2273+	*		
				2274+	*	PARAMETER LIST FOR THE EXIT ROUTINE	
				2275+	*		
000200				2276+	EXITLIST DS	0F	
000200				2277+	EXITAREA DS	A	WORKAREA LOCATION
000204				2278+	DSCBADDR DS	A	ADDRESS OF THE DSCB
000208				2279+	FORMATAD DS	A	ADDRESS OF THE FORMATTED DSCB
00020C				2280+	CPPLADDR DS	A	ADDRESS OF THE CPPL
000210				2281+	ACTIONAD DS	A	ADDRESS OF THE RECOMMENDED OR REQUESTED ACTION
				2282+	*		
				2283+	*	INTER ROUTINE FLAGS	
				2284+	*		
000214				2285+	VTCEFUNC DS	X	VTOCEXCP FUNCTION FLAG
000215				2286+	VTCFMTCK DS	X	FORMAT IS CALLED BY CHECK RTN
		00080		2287+	VTCFMTCD EQU	X'80'	FORMAT WAS CALLED BY CHECK
		00008		2288+	VTCFMTCC EQU	X'08'	FORMAT WAS CALLED BY CHECK THIS CALL
				2289+	*		
000216				2290+	TABFULL DS	X	FLAG TABLES FULL, STOP INPUT
000217				2291+	LOCAT DS	X	FLAG TO PERFORM CATALOG LOCATE
000218				2292+	VTCEPRNT DS	X	PRINT END AND CLEANUP FLAG
00021A				2293+	DSNLEN DS	H	LENGTH OF THE DSNAME (NON-BLANKS)
00021C				2294+	ATABTITL DS	A	ADDRESS OF TABLE OF TITLES, LENGTHS
				2295+	*		
				2296+	*		
				2297+	*		
				2298+	*	WORKING STORAGE FOR VOLUME UCB SEARCH	
				2299+	*		
000220				2300+	ADDR DS	CL3	UCB ADDRESS IN CHARACTERS
000223				2301+	VOLSER DS	CL6	VOLUME SERIAL NUMBER FROM PARSE
000229				2302+	VOLID DS	CL6	CURRENT VOLUME SERIAL NUMBER TO PROCESS
00022F				2303+	FLAG DS	X	UCB SEARCH FLAG
000230				2304+	LASTADR DS	F	LAST UCB ADDRESS FOUND (NO DUP'S)
000234				2305+	UCBDEVT DS	CL4	PRINTABLE FORM OF DEVICE TYPE RPRINS
				2306+	*		
				2307+	*		
				2308+	*		
000238				2309+	SORTTAB DS	16F	
				2310+	*		
				2311+	*	EACH ENTRY CONTAINS A KEY OFFSET (2 BYTES) AND A KEY LENGTH (2 BYTES)	
				2312+	*	THIS TABLE IS BUILT AT PARSE TIME ACCORDING TO THE SORT PARAMETERS	
				2313+	*	SPECIFIED. THE 1ST PARM IS THE HIGH KEY AND SO ON.	
				2314+	*		
				2315+	*		
				2316+	*		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				2317+*		ADDRESSES OF GETMAIN FOR FORMATTED DATA	
				2318+*			
000278				2319+VTCCURAD	DS	A	CURRENT AVAILABLE ADDRESS
00027C				2320+VTCCURLN	DS	A	CURRENT AVAILABLE LENGTH
000280				2321+VTCGETMN	DS	50A	ADDRESSES OF BLOCKS
		00032		2322+VTCGETMX	EQU	(* -VTCGETMN)/4	NUMBER OF BLOCKS MAXIMUM
		08000		2323+VTCGETMS	EQU	32768	GETMAIN SIZE
				2324+*			
				2325+*		HASH SORT TABLE, POINTERS TO FIRST ENTRIES	
				2326+*			
000348				2327+VTCSORTH	DS	256A	POINT TO FORMATED ENTRIES
		00748		2328+VTCSORTE	EQU	*	END OF LIST
				2329+*			
				2330+*		PRINT ENTRIES - PAGE AND LINE COUNTERS	
				2331+*			
000748				2332+LINECT	DS	H	LINE COUNT
00074A				2333+LINEMAX	DS	H	MAXIMUM LINES PER PAGE
00074C				2334+PAGECT	DS	H	PAGE COUNT
00074E				2335+LINELEN	DS	H	LENGTH OF THE PRINT LINE
				2336+*			
				2337+*			
				2338+*		VARIOUS ITEMS	
				2339+*			
000750				2340+FMT4	DS	XL44	SPACE FOR DSCB NAME
		0077C		2341+IECSDSL4	EQU	*	FORMAT 4 DSCB
		0077C		2342+IECSDSF4	EQU	IECSDSL4	16000000
00077C				2343+DS4IDFMT	DS	CL1	FORMAT IDENTIFIER
00077D				2344+DS4HPCHR	DS	XL5	HIGHEST ADDRESS OF A FORMAT 1 DSCB
000782				2345+DS4DSREC	DS	XL2	NUMBER OF AVAILABLE DSCB'S
000784				2346+DS4HCCHH	DS	XL4	CCHH OF NEXT AVAILABLE ALTERNATE TRK
000788				2347+DS4NOATK	DS	XL2	NUMBER OF REMAINING ALTERNATE TRACKS
00078A				2348+DS4VTOCI	DS	XL1	VTOC INDICATORS
		00080		2349+DS4DOSBT	EQU	X'80'	DOS BIT
		00010		2350+DS4DSTKP	EQU	X'10'	DOS STACKED PACK
		00008		2351+DS4DOCVT	EQU	X'08'	DOS CONVERTED VTOC
		00004		2352+DS4DIRF	EQU	X'04'	DIRF BIT
		00002		2353+DS4DICVT	EQU	X'02'	DIRF RECLAIMED
		00001		2354+DS4IVTOC	EQU	X'01'	VOLUME USES AN INDEXED VTOC @01A
00078B				2355+DS4NOEXT	DS	XL1	NUMBER OF EXTENTS IN THE VTOC
00078C				2356+	DS	XL2	RESERVED
00078E				2357+DS4DEVCT	DS	0XL14	DEVICE CONSTANTS
00078E				2358+DS4DEVSZ	DS	XL4	DEVICE SIZE
000792				2359+DS4DEVTK	DS	XL2	DEVICE TRACK LENGTH
000794				2360+DS4DEVOV	DS	0XL2	KEYED RECORD OVERHEAD
000794				2361+DS4DEVI	DS	XL1	NON-LAST KEYED RECORD OVERHEAD
000795				2362+DS4DEVL	DS	XL1	LAST KEYED RECORD OVERHEAD
000796				2363+DS4DEVK	DS	XL1	NON-KEYED RECORD OVERHEAD
				2364+*		DIFFERENTIAL	17150000
000797				2365+DS4DEVFG	DS	XL1	FLAG BYTE
000798				2366+DS4DEVTL	DS	XL2	DEVICE TOLERANCE
00079A				2367+DS4DEVDT	DS	XL1	NUMBER OF DSCB'S PER TRACK
00079B				2368+DS4DEVDB	DS	XL1	NUMBER OF DIRECTORY BLOCKS PER TRACK
00079C				2369+DS4AMTIM	DS	XL8	VSAM TIME STAMP
0007A4				2370+DS4AMCAT	DS	0XL3	VSAM CATALOG INDICATOR
0007A4				2371+DS4VSIND	DS	XL1	VSAM INDICATORS

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
0007A5				2372+DS4VSCRA	DS	XL2	RELATIVE TRACK LOCATION OF THE CRA 17550000
0007A7				2373+DS4R2TIM	DS	XL8	VSAM VOLUME/CATALOG MATCH 17600000
				2374+*			TIME STAMP 17650000
0007AF				2375+	DS	XL5	RESERVED 17700000
0007B4				2376+DS4F6PTR	DS	XL5	POINTER TO FIRST FORMAT 6 DSCB 17750000
0007B9				2377+DS4VTOCE	DS	XL10	VTOC EXTENT DESCRIPTION 17800000
0007C3				2378+	DS	XL25	RESERVED 17850000
		007DC		2379+DS4END	EQU	*	17900000
0007E0				2380+	DS	0D	
0007E0				2381+FMT3	DS	0XL148	SPACE FOR FORMAT3 DSCB
		007E0		2382+IECSDSL3	EQU	*	FORMAT 3 DSCB 15250000
		007E0		2383+IECSDSF3	EQU	IECSDSL3	15300000
0007E0				2384+	DS	XL4	KEY IDENTIFIER 15350000
0007E4				2385+DS3EXTNT	DS	XL40	FOUR EXTENT DESCRIPTIONS 15400000
				2386+*		FIRST BYTE	EXTENT TYPE INDICATOR 15450000
				2387+*		SECOND BYTE	EXTENT SEQUENCE NUMBER 15500000
				2388+*		THIRD - SIXTH BYTES	LOWER LIMIT 15550000
				2389+*		SEVENTH - TENTH BYTES	UPPER LIMIT 15600000
00080C				2390+DS3FMTID	DS	CL1	FORMAT IDENTIFIER 15650000
00080D				2391+DS3AEXT	DS	XL90	NINE ADDITIONAL EXTENT DESCRIPTIONS 15700000
000867				2392+DS3PTRDS	DS	XL5	RESERVED 15750000
		0086C		2393+DS3END	EQU	*	15800000
000870				2394+	DS	0D	
				2395		PRINT NOGEN	
				2397 *			
				2398 *		WORK AREAS FOR SUBROUTINES	
				2399 *			
000870				2400 WORKMSG	DS	XL256	
000970				2401 WORKEXCP	DS	4XL256	
000D70				2402 WORKCHEK	DS	XL256	
000E70				2403 WORKFORM	DS	2XL256	
001070				2404 WORKEEXIT	DS	8XL256	
001870				2405 WORKSORT	DS	XL256	
001970				2406 WORKPRNT	DS	10XL256	
002370				2407	DS	0D	
		02370		2408 LENWORK	EQU	*-WORKAREA	
				2410		VTFMT	
				2452		PDEDSNAM	
				2474		IKJPPL	
				2490		IKJIOPL	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				2504	IKJPSCB	
				2547	IKJECT	
				2595	IKJCPPL	
				2607	IKJUPT	
				2635	PRINT NOGEN	
				2636	CVT	
				2685+	PRINT OFF	03150001
				3636	END	

POS.ID	REL.ID	FLAGS	ADDRESS	ASM 0201 13.38 01/07/25
0001	0001	0C	0003E0	
0001	0001	0C	000504	
0001	0001	0C	0007C4	
0001	0001	0C	0007C8	
0001	0001	0C	0007D0	
0001	0001	0C	0007D4	
0001	0001	0C	0007DC	
0001	0001	0C	0007E0	
0001	0001	0C	0007E8	
0001	0001	0C	0007EC	
0001	0001	0C	0007F4	
0001	0001	0C	0007F8	
0001	0001	0C	000800	
0001	0001	0C	000804	
0001	0001	0C	00080C	
0001	0001	0C	000810	
0001	0001	0C	000818	
0001	0001	0C	00081C	
0001	0001	0C	000824	
0001	0001	0C	000828	
0001	0001	0C	000830	
0001	0001	0C	000834	
0001	0001	0C	00083C	
0001	0001	0C	000840	
0001	0001	0C	000848	
0001	0001	0C	00084C	
0001	0001	0C	000854	
0001	0001	0C	000858	
0001	0001	0C	000860	
0001	0001	0C	000864	
0001	0001	0C	00086C	
0001	0001	0C	000870	
0001	0001	0C	000878	
0001	0001	0C	00087C	
0001	0002	1C	0005F8	
0001	0003	1C	000600	
0001	0004	1C	000604	
0001	0005	1C	000608	
0001	0006	1C	00060C	
0001	0007	1C	000610	
0001	0008	0C	000614	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
ADDR	00003	00000220	02300	00283
ADDRANSR	00004	000000B8	02261	00103 00545 00629 00644 00644
ADDRCBUF	00004	000000B4	02259	00485 00537
ADDRECT	00004	000000AC	02257	00403 00484 00536
ADDRPCL	00004	00000614	00675	00548
ADDRPSCB	00004	000000B0	02258	00483
ADDRUPT	00004	000000A8	02256	00402 00482 00535
ALCSORT	00004	00000C9C	00869	00744 00745
ALCSORTE	00004	00000D8C	00891	00744 00745
AL00651	00001	00001341	01242	01241
AL00661	00001	0000134F	01256	01255
ANDOR1K	00002	00000034	01230	01227
ANDOR2K	00002	00000036	01269	01266
ANDOR3K	00002	00000038	01300	01297
ATTNECB	00004	000000DC	02264	00404 00405 00541 00542
BLANKS	00008	0000061C	00677	00186
BLKSORT	00004	00000EE8	00926	00758 00759
BLKSORTE	00004	00000FB4	00943	00758 00759
BLKSZSET	00004	000000BC	01670	01659
BREAK	00004	000000AC	01623	01610
BREAKK	00002	00000030	01188	01185
CALLEXIT	00004	0000033C	00351	00340
CATK	00002	0000002C	01138	01135
CDTSORT	00004	00000FC0	00946	00760 00761
CDTSORTE	00004	0000117C	00987	00760 00761
CHARALL	00003	0000062A	00686	00198
CHARALV	00003	0000062D	00687	00196
CHARSK	00002	0000003C	01361	01358
CHARSPL	00004	000000B4	01650	01639
CHKRDY	00004	00000262	00268	00229
CHKVIRT	00004	00000286	00277	00271
CLCVOL	00006	00000636	00689	00228
CONTAIN	00004	00000084	01566	01560
CONTAINK	00002	00000028	01094	01091
CPPL	00001	00000000	02600	00481
CPPLADDR	00004	0000020C	02280	00480
CPPLCBUF	00004	00000000	02601	00485
CPPLECT	00004	0000000C	02604	00484
CPPLPSCB	00004	00000008	02603	00483
CPPLUPT	00004	00000004	02602	00482
CVTBRET	00002	0000006A	02748	03315
CVTMAP	00001	00000018	02691	00554
CVTPARS	00004	00000224	03143	00556 00559
CVTPTR	00001	00000010	02686	00553
CVTXTNT1	00001	00000450	03498	02925
CVTXTNT2	00001	0000045C	03507	02996
CVT4MS1	00001	00000010	02767	02774
CVT6DAT	00001	00000002	02770	02774
DEND0098	00001	000014F1	01592	01582
DEND0100	00001	00001524	01621	01609
DEND0102	00001	00001555	01648	01638
DEND0103	00001	00001576	01668	01658
DEND0105	00001	000015A7	01695	01685
DEND0107	00001	000015CD	01722	01712
DEND0108	00001	000015E9	01742	01732

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
DEND0112	00001	00001625	01789	01777	
DEND0114	00001	00001660	01816	01806	
DEND0115	00001	000016AF	01839	01827	
DEND0116	00001	000016F3	01861	01850	
DEND0118	00001	0000172E	01888	01878	
DEND0119	00001	0000177D	01911	01899	
DEND0120	00001	000017C1	01933	01922	
DEND0122	00001	000017FC	01960	01950	
DEND0123	00001	0000184B	01983	01971	
DEND0124	00001	0000188F	02005	01994	
DEND0126	00001	000018CA	02032	02022	
DEND0127	00001	00001919	02055	02043	
DEND0128	00001	0000195D	02077	02066	
DEND0130	00001	00001986	02106	02094	
DEND0132	00001	000019B3	02135	02123	
DENT0098	00001	000014D8	01580	01582	
DENT0100	00001	000014F4	01607	01609	
DENT0102	00001	00001527	01636	01638	
DENT0103	00001	00001555	01656	01658	
DENT0105	00001	00001579	01683	01685	
DENT0107	00001	000015AA	01710	01712	
DENT0108	00001	000015CD	01730	01732	
DENT0112	00001	000015F6	01775	01777	
DENT0114	00001	00001628	01804	01806	
DENT0115	00001	00001660	01825	01827	
DENT0116	00001	000016AF	01848	01850	
DENT0118	00001	000016F6	01876	01878	
DENT0119	00001	0000172E	01897	01899	
DENT0120	00001	0000177D	01920	01922	
DENT0122	00001	000017C4	01948	01950	
DENT0123	00001	000017FC	01969	01971	
DENT0124	00001	0000184B	01992	01994	
DENT0126	00001	00001892	02020	02022	
DENT0127	00001	000018CA	02041	02043	
DENT0128	00001	00001919	02064	02066	
DENT0130	00001	00001960	02092	02094	
DENT0132	00001	00001989	02121	02123	
DEVT#	00001	00000008	00736	00286	
DEVTYPS	00001	00000795	00728	00285 00736	
DOUBLE	00008	000000E0	02265	00594 00595 00596	
DSNLNTYP	00002	00000164	02150	02145	
DSNPLN	00004	0000015C	02137	02124	
DSNPLNK	00002	00000048	01495	01492	
DSNSORT	00004	00000888	00778	00738 00739	
DSNSORTE	00004	00000B4C	00837	00738 00739	
DSOSORT	00004	00000E34	00908	00752 00753	
DSOSORTE	00004	00000E64	00912	00752 00753	
EDTSORT	00004	00000FC0	00947	00764 00765	
EDTSORTE	00004	0000117C	00985	00764 00765	
ENDING	00004	00000068	01545	01539	
ENDKEY	00002	00000026	01073	01070	
ENDUNIT	00002	000002CA	00294	00292	
ENDVTOC	00004	00000360	00370	00317 00324	
ENT0001	00001	00000018	00077	00086	
EXIT0	00002	00000394	00394	00387	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
EXTSORT	00004	00000E1C	00905	00750 00751	
EXTSORTE	00004	00000E28	00906	00750 00751	
FLAG	00001	0000022F	02303	00193 00200 00205 00221 00223 00235 00252 00270 00272 00295 00375	
FMIN1	00004	00000618	00676	00214	
FNDGBL	00004	0000025A	00263	00222	
FNDUCB	00006	0000028E	00282	00226 00273 00276	
FNDUCB2	00006	000002A8	00287	00290	
FNDUCB3	00006	000002C4	00293	00288	
FORMATK	00002	00000046	01474	01471	
FORMATSP	00004	00000150	02108	02095	
FREEPDL	00002	00000588	00624	00434	
HEADING	00004	000000E0	01762	01758	
HEADK	00002	00000040	01403	01400	
H2	00002	00000624	00678	00194	
H4	00002	00000626	00679	00323	
H5	00002	00000628	00680	00191	
IECSDSL3	00001	000007E0	02382	02383	
IECSDSL4	00001	0000077C	02341	02342	
IKJ\$0017	00002	000003F2	00425	00417	
IKJ\$0037	00002	000005B2	00642	00632	
IKJ\$0044	00001	000019DE	02177	01015	
IKJ\$0046	00001	0000128D	01036	01032	
IKJ\$0100	00001	00001524	01620	01619	
IKJ\$0112	00001	00001625	01788	01787	
IKJ\$0115	00001	000016AF	01838	01837	
IKJ\$0116	00001	000016F3	01860	01859	
IKJ\$0119	00001	0000177D	01910	01909	
IKJ\$0120	00001	000017C1	01932	01931	
IKJ\$0123	00001	0000184B	01982	01981	
IKJ\$0124	00001	0000188F	02004	02003	
IKJ\$0127	00001	00001919	02054	02053	
IKJ\$0128	00001	0000195D	02076	02075	
IKJ\$0130	00001	00001986	02105	02104	
IKJ\$0132	00001	000019B3	02134	02133	
IKJ@0017	00002	000003DA	00418	00414	
IKJ@0037	00002	0000059A	00633	00641	
IKJ@0044	00001	00000168	02175	01016	
IKJ@0045	00001	0000007D	01045	01017	
IKJ@0046	00001	00001244	01031	01030	
IKJ@0048	00001	0000129D	01060	01059	
IKJ@0050	00001	000012B0	01081	01080	
IKJ@0052	00001	000012C7	01102	01101	
IKJ@0053	00001	000012D4	01115	01114	
IKJ@0054	00001	000012DD	01126	01125	
IKJ@0056	00001	000012EB	01146	01145	
IKJ@0058	00001	000012FA	01166	01165	
IKJ@0059	00001	00001307	01176	01175	
IKJ@0061	00001	00001317	01196	01195	
IKJ@0063	00001	00001329	01217	01216	
IKJ@0065	00001	0000133A	01238	01237	
IKJ@0066	00001	00001349	01252	01251	
IKJ@0068	00001	0000135E	01277	01276	
IKJ@0069	00001	00001368	01287	01286	
IKJ@0071	00001	00001379	01308	01307	
IKJ@0072	00001	00001383	01318	01317	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
IKJ@0074	00001	00001395	01339	01338	
IKJ@0075	00001	000013A3	01349	01348	
IKJ@0077	00001	000013B3	01369	01368	
IKJ@0079	00001	000013C5	01390	01389	
IKJ@0081	00001	000013D9	01411	01410	
IKJ@0082	00001	000013E9	01421	01420	
IKJ@0084	00001	000013FA	01441	01440	
IKJ@0086	00001	0000140D	01462	01461	
IKJ@0088	00001	0000141E	01482	01481	
IKJ@0090	00001	00001431	01503	01502	
IKJ@0091	00001	00000258	01530	01511	
IKJ@0092	00001	00001468	01521	01520	
IKJ@0093	00001	0000028A	01551	01532	
IKJ@0094	00001	0000149A	01542	01541	
IKJ@0095	00001	000002C5	01572	01553	
IKJ@0096	00001	000014D5	01563	01562	
IKJ@0097	00001	000002E1	01599	01574	
IKJ@0098	00001	000014F0	01590	01589	
IKJ@0099	00001	00000314	01628	01601	
IKJ@0100	00001	00001521	01617	01616	
IKJ@0101	00001	00000366	01675	01630	
IKJ@0102	00001	00001554	01646	01645	
IKJ@0103	00001	00001575	01666	01665	
IKJ@0104	00001	00000397	01702	01677	
IKJ@0105	00001	000015A6	01693	01692	
IKJ@0106	00001	000003D9	01749	01704	
IKJ@0107	00001	000015CC	01720	01719	
IKJ@0108	00001	000015E8	01740	01739	
IKJ@0109	00001	000003E3	01767	01751	
IKJ@0111	00001	00000415	01796	01769	
IKJ@0112	00001	00001622	01785	01784	
IKJ@0113	00001	000004E3	01868	01798	
IKJ@0114	00001	0000165F	01814	01813	
IKJ@0115	00001	00001684	01835	01834	
IKJ@0116	00001	000016CC	01858	01857	
IKJ@0117	00001	000005B1	01940	01870	
IKJ@0118	00001	0000172D	01886	01885	
IKJ@0119	00001	00001752	01907	01906	
IKJ@0120	00001	0000179A	01930	01929	
IKJ@0121	00001	0000067F	02012	01942	
IKJ@0122	00001	000017FB	01958	01957	
IKJ@0123	00001	00001820	01979	01978	
IKJ@0124	00001	00001868	02002	02001	
IKJ@0125	00001	0000074D	02084	02014	
IKJ@0126	00001	000018C9	02030	02029	
IKJ@0127	00001	000018EE	02051	02050	
IKJ@0128	00001	00001936	02074	02073	
IKJ@0129	00001	00000776	02113	02086	
IKJ@0130	00001	00001983	02102	02101	
IKJ@0131	00001	000007A3	02141	02115	
IKJ@0132	00001	000019AF	02131	02130	
IKJ@0133	00001	000019C2	02147	02146	
IKJ@0134	00001	000019CF	02158	02157	
IKJ@0135	00001	000019DD	02167	02166	
IKJ00461	00001	0000128D	01035	01034	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
INCR1	00004	0000020A	00230	00213 00217 00219 00227 00253 00264 00269 00275 00278 00376	
IOPL	00001	00000000	02491	00400	
IOPLECB	00004	00000008	02500	00406	
IOPLECT	00004	00000004	02499	00403	
IOPLUPT	00004	00000000	02498	00402	
KEND0047	00001	00001293	01050	01048	
KEND0049	00001	000012A5	01071	01069	
KEND0051	00001	000012B8	01092	01090	
KEND0053	00001	000012D4	01116	01112	
KEND0055	00001	000012E3	01136	01134	
KEND0057	00001	000012F1	01156	01154	
KEND0060	00001	0000130D	01186	01184	
KEND0062	00001	0000131F	01207	01205	
KEND0064	00001	00001331	01228	01226	
KEND0067	00001	00001355	01267	01265	
KEND0070	00001	00001370	01298	01296	
KEND0073	00001	0000138B	01329	01327	
KEND0076	00001	000013A9	01359	01357	
KEND0078	00001	000013BB	01380	01378	
KEND0080	00001	000013CD	01401	01399	
KEND0083	00001	000013EF	01431	01429	
KEND0085	00001	00001402	01452	01450	
KEND0087	00001	00001413	01472	01470	
KEND0089	00001	00001426	01493	01491	
KEND0133	00001	000019C2	02148	02144	
KEYW0047	00001	0000128D	01046	01048	
KEYW0049	00001	0000129F	01067	01069	
KEYW0051	00001	000012B2	01088	01090	
KEYW0053	00001	000012C9	01110	01112	
KEYW0055	00001	000012DD	01132	01134	
KEYW0057	00001	000012EB	01152	01154	
KEYW0060	00001	00001307	01182	01184	
KEYW0062	00001	00001319	01203	01205	
KEYW0064	00001	0000132B	01224	01226	
KEYW0067	00001	0000134F	01263	01265	
KEYW0070	00001	0000136A	01294	01296	
KEYW0073	00001	00001385	01325	01327	
KEYW0076	00001	000013A3	01355	01357	
KEYW0078	00001	000013B5	01376	01378	
KEYW0080	00001	000013C7	01397	01399	
KEYW0083	00001	000013E9	01427	01429	
KEYW0085	00001	000013FC	01448	01450	
KEYW0087	00001	0000140D	01468	01470	
KEYW0089	00001	00001420	01489	01491	
KEYW0133	00001	000019B3	02142	02144	
LASTADR	00004	00000230	02304	00207 00207 00218 00220	
LENWORK	00001	00002370	02408	00078 00451	
LEVEL	00004	0000004C	01524	01518	
LEVKEY	00002	00000024	01052	01049	
LIMITK	00002	00000032	01209	01206	
LINESK	00002	0000003E	01382	01379	
LINESPP	00004	000000C4	01697	01686	
LOOP1	00004	0000015C	00181	00378	
LRCSORT	00004	00000EE8	00925	00756 00757	
LRCSORTE	00004	00000FB4	00944	00756 00757	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
MAXPARSE	00001	00000018	00588	00575	
MOVVOL	00006	00000630	00688	00187	
MSGADDRS	00004	000000E8	02267	00242	00606
MSGCMDEP	00002	00000716	00718	00604	
MSGNOTMT	00002	00000730	00720	00237	
MSGOFFLN	00002	00000752	00722	00258	
MSGPARSE	00002	00000700	00716	00591	00591 00592
MSGPEND	00002	00000770	00724	00248	
MSGTEXT2	00124	0000016C	02269	00237	00238 00240 00248 00258 00591 00592 00598 00601 00605
NAME0048	00001	00001293	01056	01058	
NAME0050	00001	000012A5	01077	01079	
NAME0052	00001	000012B8	01098	01100	
NAME0054	00001	000012D4	01122	01124	
NAME0056	00001	000012E3	01142	01144	
NAME0058	00001	000012F1	01162	01164	
NAME0059	00001	000012FC	01172	01174	
NAME0061	00001	0000130D	01192	01194	
NAME0063	00001	0000131F	01213	01215	
NAME0065	00001	00001331	01234	01236	
NAME0066	00001	00001341	01248	01250	
NAME0068	00001	00001355	01273	01275	
NAME0069	00001	00001360	01283	01285	
NAME0071	00001	00001370	01304	01306	
NAME0072	00001	0000137B	01314	01316	
NAME0074	00001	0000138B	01335	01337	
NAME0075	00001	00001397	01345	01347	
NAME0077	00001	000013A9	01365	01367	
NAME0079	00001	000013BB	01386	01388	
NAME0081	00001	000013CD	01407	01409	
NAME0082	00001	000013DB	01417	01419	
NAME0084	00001	000013EF	01437	01439	
NAME0086	00001	00001402	01458	01460	
NAME0088	00001	00001413	01478	01480	
NAME0090	00001	00001426	01499	01501	
NAME0134	00001	000019C2	02154	02156	
NAME0135	00001	000019CF	02163	02165	
NEND0048	00001	0000129F	01062	01058	
NEND0050	00001	000012B2	01083	01079	
NEND0052	00001	000012C9	01104	01100	
NEND0054	00001	000012DD	01127	01124	
NEND0056	00001	000012EB	01147	01144	
NEND0058	00001	000012FC	01168	01164	
NEND0059	00001	00001307	01177	01174	
NEND0061	00001	00001319	01198	01194	
NEND0063	00001	0000132B	01219	01215	
NEND0065	00001	00001341	01244	01236	
NEND0066	00001	0000134F	01258	01250	
NEND0068	00001	00001360	01279	01275	
NEND0069	00001	0000136A	01289	01285	
NEND0071	00001	0000137B	01310	01306	
NEND0072	00001	00001385	01320	01316	
NEND0074	00001	00001397	01341	01337	
NEND0075	00001	000013A3	01350	01347	
NEND0077	00001	000013B5	01371	01367	
NEND0079	00001	000013C7	01392	01388	

SYMBOL	LEN	VALUE	DEFN	REFERENCES
NEND0081	00001	000013DB	01413	01409
NEND0082	00001	000013E9	01422	01419
NEND0084	00001	000013FC	01443	01439
NEND0086	00001	0000140D	01463	01460
NEND0088	00001	00001420	01484	01480
NEND0090	00001	00001433	01505	01501
NEND0134	00001	000019CF	02159	02156
NEND0135	00001	000019DD	02168	02165
NEXTUCB	00004	000001BE	00211	00231
NEXTVOL	00004	00000376	00377	00236 00247
NOTMNT	00004	00000212	00235	00215
OFFLINE	00006	00000250	00258	00302
OUTPUTK	00002	00000044	01454	01451
PARMERR	00004	0000038C	00392	00135 00138 00144
PARMLIST	00004	000000BC	02263	00399 00409
PARSEBAD	00002	0000053C	00590	00578 00585 00586
PARSEERR	00004	00000578	00612	00583 00584 00587
PARSELST	00004	00000048	02220	00532
PARSEOK	00002	00000580	00615	00582
PARSERET	00004	00000524	00582	00588
PARSERTN	00004	00000582	00616	00613
PARSINIT	00002	0000043C	00479	00096
PCLMAIN	00001	00001210	01014	00675 01015 01018 01024 01044 01045 01055 01061 01066 01076 01082 01087 01097 01103 01109 01121 01131 01141 01151 01161 01167 01171 01181 01191 01197 01202 01212 01218 01223 01233 01239 01247 01253 01262 01272 01278 01282 01288 01293 01303 01309 01313 01319 01324 01334 01340 01344 01354 01364 01370 01375 01385 01391 01396 01406 01412 01416 01426 01436 01442 01447 01457 01467 01477 01483 01488 01498 01504 01509 01514 01529 01530 01535 01550 01551 01556 01571 01572 01577 01598 01599 01604 01627 01628 01633 01653 01674 01675 01680 01701 01702 01707 01727 01748 01749 01754 01766 01767 01772 01795 01796 01801 01822 01845 01867 01868 01873 01894 01917 01939 01940 01945 01966 01989 02011 02012 02017 02038 02061 02083 02084 02089 02112 02113 02118 02140 02141 02153 02162 02171 02176
PCTSORT	00004	00000D98	00893	00748 00749
PCTSORTE	00004	00000E10	00903	00748 00749
PDL	00001	00000000	01019	00104 01028 01038 01049 01051 01070 01072 01091 01093 01113 01117 01135 01137 01155 01157 01185 01187 01206 01208 01227 01229 01266 01268 01297 01299 01328 01330 01358 01360 01379 01381 01400 01402 01430 01432 01451 01453 01471 01473 01492 01494 01518 01523 01539 01544 01560 01565 01583 01593 01610 01622 01639 01649 01659 01669 01686 01696 01713 01723 01733 01743 01758 01761 01778 01790 01807 01817 01828 01840 01851 01862 01879 01889 01900 01912 01923 01934 01951 01961 01972 01984 01995 02006 02023 02033 02044 02056 02067 02078 02095 02107 02124 02136 02145 02149 02173 02175
PENDING	00006	0000023E	00248	00300
PEND0046	00001	0000128D	01037	01027
PEND0092	00001	00001468	01522	01517
PEND0094	00001	0000149A	01543	01538
PEND0096	00001	000014D5	01564	01559
PEND0110	00001	000015F3	01760	01757
PHASE2	00002	0000037E	00382	00184
POST0046	00001	00001216	01025	01027
POST0092	00001	00001436	01515	01517
POST0094	00001	0000146B	01536	01538
POST0096	00001	0000149D	01557	01559
POST0110	00001	000015EC	01755	01757
PPL	00001	00000000	02475	00533
PPLANS	00004	00000010	02485	00546
PPLCBUF	00004	00000014	02486	00537

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
PPLECB	00004	00000008	02483	00543
PPLECT	00004	00000004	02482	00536
PPLPCL	00004	0000000C	02484	00548
PPLUPT	00004	00000000	02481	00535
PPLUWA	00004	00000018	02487	00539
PRINTK	00002	0000003A	01331	01328
RACSORT	00004	00001188	00989	00766 00767
RACSORTE	00004	000011A0	00991	00766 00767
RDTSORT	00004	00000FC0	00948	00762 00763
RDTSORTE	00004	0000117C	00986	00762 00763
READDCB	00004	000002F2	00316	00334 00346 00357 00365
RETURN	00002	00000396	00396	00098 00315 00325 00393
RETURN1	00002	00000400	00433	00397
RFMSORT	00004	00000E70	00914	00754 00755
RFMSORTE	00004	00000EDC	00923	00754 00755
RTNADDRS	00004	000005F8	00667	00522 00674
RTNADLEN	00001	0000001C	00674	00522
R0	00001	00000000	00458	00241 00241 00242 00404 00406 00541 00543 00545 00546 00598 00599 00605 00606
R1	00001	00000001	00459	00092 00100 00115 00117 00117 00118 00133 00147 00240 00242 00244 00285 00287 00289 00289 00293 00311 00320 00330 00342 00351 00362 00371 00384 00399 00400 00438 00492 00493 00493 00494 00496 00497 00497 00498 00500 00501 00501 00502 00504 00505 00505 00506 00508 00509 00509 00510 00512 00513 00513 00514 00516 00517 00517 00518 00532 00533 00573 00592 00596 00599 00600 00601 00604 00606 00608
R13	00001	0000000D	00471	00493 00497 00501 00505 00509 00513 00517
R14	00001	0000000E	00472	00096 00102 00246 00286 00290 00313 00322 00332 00344 00355 00364 00373 00386 00434 00440 00531 00575 00577 00610 00616 00617 00626 00646 00647
R14PARSE	00004	0000006C	02223	00531 00616
R14SAVE	00004	00000068	02222	00626 00646
R15	00001	0000000F	00473	00097 00097 00101 00102 00245 00246 00312 00313 00314 00314 00321 00322 00323 00331 00332 00333 00333 00343 00344 00345 00345 00352 00353 00353 00355 00356 00356 00363 00364 00372 00373 00385 00386 00392 00394 00394 00396 00439 00440 00442 00577 00580 00594 00609 00610 00612 00615 00615
R2	00001	00000002	00460	00092 00396 00442 00480 00481 00486
R3	00001	00000003	00461	00121 00122 00122 00124 00125 00149 00150 00150 00151 00153 00156 00160 00163 00180 00181 00182 00377 00377
R4	00001	00000004	00462	00110 00120 00121 00129 00129 00134 00134 00136 00137 00155 00155 00157 00160 00163 00164 00165 00165 00169 00182 00183 00183 00185 00187 00191 00194 00228
R5	00001	00000005	00463	00111 00112 00112 00113 00133 00139 00140 00140 00156 00158 00158 00170 00181 00196 00198 00208 00210 00210 00211 00230 00230 00688
R6	00001	00000006	00464	00120 00136 00139 00141 00143 00147 00159 00164 00166 00166 00173 00209 00209 00211 00212 00212 00214 00216 00218 00220 00225 00263 00268 00274 00277 00282 00283 00284 00299 00301 00689
R9	00001	00000009	00467	00103 00104
SETVOL	00006	00000220	00238	00257 00259
SORTCOMP	00006	000000E6	00147	00125
SORTK	00002	0000002E	01158	01155
SORTKTAB	00002	000007C0	00738	00149
SORTK1	00004	000000F0	00150	00154
SORTK2	00006	0000012A	00164	00162 00170
SORTK3	00006	0000014C	00172	00152
SORTK4	00006	00000152	00173	00171
SORTPAR1	00004	00000072	00115	00130
SORTPAR2	00004	00000076	00117	00127
SORTPAR3	00004	000000A0	00129	00119 00123 00142 00145
SORTPAR4	00006	000000AC	00133	00126

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
SORTPAR5	00004	000000EC	00149	00131
SORTTAB	00004	00000238	02309	00111 00153 00161 00167
SORTTABX	00002	0000063C	00691	00113 00115
SPACEK	00002	0000002A	01118	01113
SPECUCB	00004	00000202	00228	00224
SUBAO1	00002	000016F4	01870	01239 01253
SUBAO2	00002	000017C2	01942	01278 01288
SUBAO3	00002	00001890	02014	01309 01319
SUBBREAK	00002	000014F2	01601	01197
SUBCHARS	00002	00001525	01630	01370
SUBCONT	00002	0000149B	01553	01103
SUBDSNLN	00002	00001987	02115	01504
SUBEND	00002	00001469	01532	01082
SUBFORMT	00002	0000195E	02086	01483
SUBHEAD	00002	000015EA	01751	01412
SUBLEV	00002	00001434	01511	01061
SUBLIMIT	00002	00001626	01798	01218
SUBLINES	00002	00001577	01677	01391
SUBLKEY	00004	000000F0	01818	01807
SUBLOPER	00004	000000F8	01841	01828
SUBLVALU	00004	00000100	01863	01851
SUBPRINT	00002	000015A8	01704	01340
SUBPRTIT	00004	000000D4	01744	01733
SUBPRTKY	00004	000000CC	01724	01713
SUBSORT	00004	000000A0	01594	00110 01583
SUBSORTS	00002	000014D6	01574	01167
SUBTOTAL	00002	000015F4	01769	01442
SUB1KEY	00004	00000108	01890	01879
SUB1OPER	00004	00000110	01913	01900
SUB1VALU	00004	00000118	01935	01923
SUB2KEY	00004	00000120	01962	01951
SUB2OPER	00004	00000128	01985	01972
SUB2VALU	00004	00000130	02007	01995
SUB3KEY	00004	00000138	02034	02023
SUB3OPER	00004	00000140	02057	02044
SUB3VALU	00004	00000148	02079	02067
TABFULL	00001	00000216	02290	00316
TOTALK	00002	00000042	01433	01430
TOTALN	00004	000000E8	01791	01778
TSRE0032	00001	00000516	00571	00561
TSRL0032	00001	00000500	00563	00557
UCBDEVT	00004	00000234	02305	00284 00287 00291 00293
UNTSORT	00004	000011A0	00993	00768 00769
UNTSORTE	00004	0000120C	01002	00768 00769
UNUSORT	00004	00000C9C	00868	00746 00747
UNUSORTE	00004	00000D8C	00890	00746 00747
USESORT	00004	00000C9C	00867	00742 00743
USESORTE	00004	00000D8C	00889	00742 00743
VADCHEK	00004	00000098	02249	00331
VADEXCP	00004	00000094	02248	00312 00321 00372
VADEXIT	00004	00000090	02247	00352
VADFORM	00004	0000009C	02250	00343
VADMSG	00004	0000008C	02246	00245 00522 00609
VADPRNT	00004	000000A0	02251	00101 00385 00439
VADSORT	00004	000000A4	02252	00363

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
VOLID	00006	00000229	02302	00282	
VOLS	00004	00000008	01039	00180 01028	
VOLSER	00006	00000223	02301	00186 00225 00238 00688 00689	
VOLSET	00006	000001AE	00207	00192 00195 00199 00201 00206	
VOLSETV	00004	000001A6	00205	00197	
VOLSORT	00004	00000B58	00839	00740 00741	
VOLSORTE	00004	00000C90	00865	00740 00741	
VTCCLOSE	00001	00000002	02217	00370	
VTCEFUNC	00001	00000214	02285	00310 00319 00370	
VTCEPRNT	00001	00000218	02292	00435	
VTCFMTCD	00001	00000080	02287	00339	
VTCFMTCK	00001	00000215	02286	00339	
VTCGETMN	00004	00000280	02321	02322	
VTCOPEN	00001	00000001	02216	00310	
VTCREAD	00001	00000000	02218	00319	
VTCSORTH	00004	00000348	02327	00159 00172	
VTCWCHEK	00004	0000007C	02239	00502	
VTCWEXCP	00004	00000078	02238	00498	
VTCWEXIT	00004	00000074	02237	00510	
VTCWFORM	00004	00000080	02240	00506	
VTCWMSG	00004	00000070	02236	00494	
VTCWPRNT	00004	00000084	02241	00518	
VTCWSORT	00004	00000088	02242	00514	
VTFALLOC	00004	00000004	02416	00693 00744	
VTFBLKSZ	00002	00000030	02433	00701 00758	
VTFCREDIT	00003	0000001C	02424	00702 00760	
VTFDSN	00001	00000046	02450	00691 00738	
VTFDSORG	00003	00000026	02428	00698 00752	
VTFEXPDT	00003	0000001F	02425	00703 00764	
VTFRECL	00002	0000002E	02432	00700 00756	
VTFSTAC	00003	00000022	02426	00704 00762	
VTFMT	00001	00000000	02414	00691 00692 00693 00694 00695 00696 00697 00698 00699 00700 00701 00702 00703 00704 00705 00706 00738 00740 00742 00744 00746 00748 00750 00752 00754 00756 00758 00760 00762 00764 00766 00768 02449	
VTFNOEPV	00001	00000025	02427	00697 00750	
VTFPCT	00002	00000010	02421	00696 00748	
VTFRACF	00001	00000034	02436	00705 00766	
VTFRECFM	00005	00000029	02430	00699 00754	
VTFUNIT	00004	00000018	02423	00706 00768	
VTFUNUSD	00004	0000000C	02420	00695 00746	
VTFUSED	00004	00000008	02419	00694 00742	
VTFVOLUM	00006	00000012	02422	00692 00740	
VTOCCMD	00001	00000000	00067	00074 01021 01041 01053 01063 01074 01084 01095 01105 01119 01128 01139 01148 01159 01169 01178 01189 01199 01210 01220 01231 01245 01259 01270 01280 01290 01301 01311 01321 01332 01342 01351 01362 01372 01383 01393 01404 01414 01423 01434 01444 01455 01464 01475 01485 01496 01506 01512 01526 01533 01547 01554 01568 01575 01595 01602 01624 01631 01651 01671 01678 01698 01705 01725 01745 01752 01763 01770 01792 01799 01819 01842 01864 01871 01891 01914 01936 01943 01963 01986 02008 02015 02035 02058 02080 02087 02109 02116 02138 02151 02160 02169 02178	
VTOCOM	00008	00000070	02232	00100 00244 00311 00320 00330 00342 00351 00362 00371 00384 00438 00608	
WORKAREA	00001	00000000	02212	00094 00660 00661 00662 00663 00664 00665 00666 02408	
WORKCHEK	00256	00000D70	02402	00662	
WORKEXCP	00256	00000970	02401	00661	
WORKEEXIT	00256	00001070	02404	00664	
WORKFORM	00256	00000E70	02403	00663	

SYMBOL	LEN	VALUE	DEFN	REFERENCES
WORKMSG	00256	00000870	02400	00660
WORKPRNT	00256	00001970	02406	00666
WORKREG	00001	0000000D	00474	00094 00539
WORKSORT	00256	00001870	02405	00665

ASM 0201 13.38 01/07/25

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
=F'0'	00004	000005C0	00655	00118 00137 00151
=F'12'	00004	000005C4	00656	00157
=F'24'	00004	000005C8	00657	00169
=3F'0'	00004	000005CC	00658	00172 00173
=C'UNKN'	00004	000005D8	00659	00291
=A(WORKMSG-WORKAREA)	00004	000005DC	00660	00492
=A(WORKEXCP-WORKAREA)	00004	000005E0	00661	00496
=A(WORKCHEK-WORKAREA)	00004	000005E4	00662	00500
=A(WORKFORM-WORKAREA)	00004	000005E8	00663	00504
=A(WORKEEXIT-WORKAREA)	00004	000005EC	00664	00508
=A(WORKSORT-WORKAREA)	00004	000005F0	00665	00512
=A(WORKPRNT-WORKAREA)	00004	000005F4	00666	00516

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 864

TOTAL RECORDS READ FROM SYSTEM LIBRARY 5411

TOTAL RECORDS PUNCHED 129

TOTAL RECORDS PRINTED 2144

SYMBOL	TYPE	ID	ADDR	LENGTH	LDID
VTOCCHEK	SD	0001	000000	000B4D	
PCLMAIN	SD	0002	000B50	0007CE	

ASM 0201 13.38 01/07/25

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
2					*****	
3					*	*
4					*	*
5					* TITLE - VTOC COMMAND CHECK ROUTINE	*
6					*	*
7					* FUNCTION - CHECK THE CONDITIONS SPECIFIED ON THE VTOC COMMAND.	*
8					* SEE IF THE DATA SET PASSED SHOULD BE PROCESSED.	*
9					* THE LIMIT, ENDING, CONTAINING, CCHH, LIMIT, AND,	*
10					* AND OR KEYWORDS ARE PROCESSED BY THIS ROUTINE.	*
11					*	*
12					* OPERATION - FIRST GET THE LENGTH OF THE DSNAME AND SAVE IT.	*
13					* THEN GET THE FORMAT 3 DSCB, IF ONE EXISTS. TRY	*
14					* EACH KEYWORD TO SEE IF IT WILL EXCLUDE THE DATA	*
15					* SET FROM FURTHER PROCESSING.	*
16					*	*
17					* INPUT - VTOC COMMON AREA (VTOCOM)	*
18					* POINTED TO BY REGISTER 1	*
19					* USE PARSE DATA, FORMAT 1, 3, AND 4 DSCB'S	*
20					* FOR DSORG, RECFM, ALLOC, USED, PROT, CATLG, OR SECAL	*
21					* CALL VTOCFORM TO FORMAT THE PARMS. USE FORMATTED DSCB	*
22					*	*
23					* OUTPUT - A RETURN CODE OF 0 TO CONTINUE PROCESSING OR 8 TO	*
24					* EXCLUDE THIS DATA SET.	*
25					*	*
26					* ATTRIBUTES - REENTRANT, REUSEABLE, REFRESHABLE.	*
27					*	*
28					*	*
29					* PROGRAMMED BY R. L. MILLER (415) 485-6241	*
30					*	*
31					* 12/19/97- MODIFIED BY JOHN KALINICH AT USA LSSC TO ADD YEAR Y2K	*
32					* 2000 SUPPORT FOR DATES. Y2K	*
33					*	*
34					*****	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25

      36 *          MACROS FOR CHECK ROUTINE
      37 *
      38          MACRO
      39 &LABEL    VTCHL &KEY          CALL THE KEYWORD CHECK ROUTINE
      40 &LABEL    L      R1,SUB&KEY.OPER  GET THE OPERATOR VALUE
      41          BAL    R8,GETOPER    TRANSLATE TO A NUMBER
      42          ST     R15,REFOPER    SAVE IT TOO
      43          LA    R1,SUB&KEY.VALU  GET THE VALUE PDL
      44          ST     R1,REFVAL      SAVE THAT ADDRESS THREE
      45          CLI   FLAGM&KEY,0    HAS IT BEEN CONVERTED?
      46          BNE   VTP&SYSNDX    YES, SKIP ALONG
      47          MVI   FLAGM&KEY,1    NOTE IT AS CONVERTED
      48          BAL   R8,PDLNUM      GO CONVERT IT
      49          ST     R15,NUMBER&KEY  SAVE THE VALUE
      50          LA    R4,SUB&KEY.KEY  POINT TO THE IKJIDENT FOR THE KEYWORD
      51          BAL   R8,GETKEY      CONVERT TEXT TO A NUMERIC KEY
      52          STC   R15,NUMKEY&KEY  SAVE THAT NUMERIC KEY
      53          LTR   R15,R15        WAS IT SUCCESSFUL?
      54          BNZ   VTP&SYSNDX    YES, SKIP ALONG
      55 *          ISSUE A MESSAGE - A BAD LIM, AND, OR KEYWORD
      56          MVC   MSGTEXT2,KEYERR  START THE ERROR MESSAGE
      57          L     R1,0(R4)        POINT TO THE TEXT
      58          MVC   MSGTEXT2+49(6),0(R1)  THEN ADD IT TO THE MESSAGE
      59          VTOCMSG MSGTEXT2     ISSUE THE ERROR MESSAGE
      60 VTP&SYSNDX DS  0H
      61          SR    R1,R1          CLEAR A REGISTER
      62          ICM   R1,1,NUMKEY&KEY  GET THE KEYWORD VALUE
      63          BZ    VTE&SYSNDX     IF NOT SET, SKIP THE EVALUATION
      64          ST    R1,REFKEY      SAVE THE ADDRESS
      65          LA    R1,NUMBER&KEY  GET THE ADDRESS OF CONVERTED NUMBER
      66          ST    R1,REFNUM      SAVE IT'S ADDRESS
      67          BAL   R8,LIMEVAL     GO EVALUATE THE EXPRESSION
      68 VTE&SYSNDX DS  0H
      69          MEND

      71          MACRO
      72 &LABEL    VTANDOR &NUM        EVALUATE, THEN DO AND OR OR FUNCTION
      73 &LABEL    CLI   ANDOR&NUM.K+1,0  WAS THIS KEYWORD SET?
      74          BE    LIMCOMP        NO, JUST CHECK THE FINAL RESULT
      75          VTCHL &NUM          YES, EVALUATE
      76          CLI   ANDOR&NUM.K+1,1  WAS IT AN AND ?
      77          BE    VTA&SYSNDX     YES, DO THE AND
      78          O     R15,LIMVAL     NO, OR IT
      79          B     VTE&SYSNDX     FINISHED WITH THIS EXPRESSION
      80 VTA&SYSNDX N  R15,LIMVAL     AND THE EXPRESSION VALUE
      81 VTE&SYSNDX ST R15,LIMVAL     SAVE THE VALUE
      82          MEND
      83 *

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				85	GBLC &WIN19XX	Y2K DEC97
				87 *	THE FOLLOWING VARIABLE DETERMINES THE 19XX DATE WINDOW.	Y2K DEC97
				88 *		Y2K DEC97
				89 *	&WIN19XX IS THE START OF THE 19XX DATE WINDOW. ANY	Y2K DEC97
				90 *	YEAR LESS THAN &WIN19XX WILL BE CONSIDERED A 20XX DATE.	Y2K DEC97
				91 *		Y2K DEC97
				92	&WIN19XX SETC '72' START OF 19XX DATE WINDOW	Y2K DEC97
				93 *		
				94	VTOCCHK ENTER 12,12 DO THE HOUSEKEEPING	
000000				95+	VTOCCHK CSECT	
000000	47F0 F00E	0000E		96+	B 14(0,15) BRANCH AROUND ID	00450000
000004	08			97+	DC AL1(8) LENGTH OF IDENTIFIER	00550000
000005	E5E3D6C3C3C8C5D2			98+	DC CL8'VTOCCHK' IDENTIFIER	00750000
00000D	00					
00000E	90EC D00C	0000C		99+	STM 14,12,12(13) SAVE REGISTERS	03700000
000012	18CF			100+	LR 12,15 SET FIRST BASE REG	
000014				101+	CNOP 0,4	
			00000	102+	USING VTOCCHK,12	
000014	5811 000C	0000C		103+	L 1,12+0(1) NUMERIC &SAVE IMPLIES A PASSED SAVEAREA	
000018	50D1 0004	00004		104+	ST 13,4(1) PRIOR SAVEAREA ADDRESS TO MINE	
00001C	501D 0008	00008		105+	ST 1,8(13) MY SAVEAREA ADDRESS TO HIS	
000020	182D			106+	LR 2,13 KEEP THE SAVEAREA ADDRESS FOR REGS	
000022	18D1			107+	LR 13,1 THIS IS MY SAVEAREA	
000024	9802 2014	00014		108+	LM 0,2,20(2) RESTORE ORIGINAL REGS	
000028	18B1			109	LR R11,R1 SAVE ADDR OF VTOCOM	
			00000	110	USING VTOCOM,R11 SET ITS ADDRESSABILITY	
00002A	5890 B048	00048		111	L R9,ADDRANSR POINT TO THE PARSE ANSWER	
			00000	112	USING PDL,R9 SET ITS ADDRESSABILITY	
			00000	113	USING CHEKWORK,R13 SET ADDRESSABILITY FOR LOCAL WORK AREA	
				114 *		
				115 *	SEE WHAT THE FORMAT ID IS	
				116 *		
00002E	5870 B194	00194		117	CHEKFMT L R7,DSCBADDR POINT TO THE DSCB	
000032	4177 0008	00008		118	LA R7,8(R7) GET PAST THE HEADER	
			00000	119	USING DSCB1,R7 SET ADDRESSABILITY	
000036	9500 9047	00047		120	CLI FORMATK+1,0 DID HE SPECIFY VARIOUS DSCB'S	
00003A	4770 C03E	0003E		121	BNE CHEKFMTI YES, GO DO HIS CHECKS	
00003E				122	CHEKFMTI DS 0H NOT YET PROGRAMMED	
				123 *		
				124 *	STANDARD IS ONLY TO ALLOW FORMAT ONES TO GO	
				125 *		
00003E	95F1 702C	0002C		126	CLI DS1FMTID,C'1' IS THIS A FORMAT 1?	
000042	4770 C754	00754		127	BNE CHECKOUT NO, EXCLUDE IT FROM FURTHER PROCESSING	
				128 *		
				129 *	FIRST SEE HOW BIG THE DSNAME IS	
				130 *		
000046	4110 702C	0002C		131	LA R1,DS1FMTID POINT PAST THE DSNAME	
00004A	DD2B 7000 C8DB	00000 008DB		132	TRT DS1DSNAM,BLKTRTAB FIND THE FIRST BLANK	
000050	1B17			133	SR R1,R7 SUBTRACT TO GET THE LENGTH	
000052	4010 B1AA	001AA		134	STH R1,DSNLEN SAVE THE DSNAME LENGTH	
000056	1831			135	LR R3,R1 KEEP THE LENGTH FOR LATER	
				136 *		
				137 *	GET THE FORMAT 3 DSCB, IF IT EXISTS	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                ASM 0201 13.38 01/07/25
                                138 *
000058 D793 B770 B770 00770 00770 139 XC FMT3,FMT3 CLEAR IT FIRST
00005E D504 7087 CB48 00087 00B48 140 CLC DS1PTRDS,=XL5'0000000000' IS THERE A FORMAT 3?
000064 4780 C092 00092 141 BE FMT3NO NO, SKIP ALONG
                                142 *
                                143 * SET UP THE CAMLST
                                144 *
000068 D203 D058 C9DC 00058 009DC 145 MVC CAMSEEK(4),CAMSCON MOVE IN THE FIRST WORD OF CAMLST
00006E 4110 7087 00087 146 LA R1,DS1PTRDS GET THE CCHHR ADDRESS
000072 5010 D05C 0005C 147 ST R1,CAMSEEK+4 SAVE IT
000076 4110 B1B9 001B9 148 LA R1,VOLID POINT TO THE VOLUME SERIAL
00007A 5010 D060 00060 149 ST R1,CAMSEEK+8 SAVE IT
00007E 4110 B770 00770 150 LA R1,FMT3 POINT TO THE AREA FOR THE DSCB3
000082 5010 D064 00064 151 ST R1,CAMSEEK+12 SAVE IT
                                152 OBTAIN CAMSEEK GET THE DSCB
000086 4110 D058 00058 153+ LA 1,CAMSEEK LOAD PARAMETER REG 1 01900002
00008A 0A1B 154+ SVC 27 ISSUE OBTAIN SVC 00100019
00008C 12FF 155 LTR R15,R15 TEST THE RETURN CODE
00008E 4770 C740 00740 156 BNZ OBT3ERR BAD NEWS, ISSUE THE MESSAGE
                                157 *
                                158 * PROCESS THE LEVEL KEYWORD
                                159 *
000092 9500 9025 00025 160 FMT3NO CLI LEVKEY+1,0 WAS LEVEL SPECIFIED?
000096 4780 C0CA 000CA 161 BE LEVEND NO, SKIP ON
00009A 4140 904C 0004C 162 LA R4,LEVEL YES, POINT TO THE PDE
                                00000 163 USING PDEDSNAM,R4 SET ADDRESSABILITY
00009E 1827 164 LR R2,R7 POINT TO THE START OF THE DSNAME
0000A0 1B55 165 LEVNEXT SR R5,R5 CLEAR FOR INSERT
0000A2 BF53 4004 00004 166 ICM R5,B'0011',PDEDSNL GET THE DSNAME LENGTH
0000A6 4780 C754 00754 167 BZ CHECKOUT END OF THE LINE, EXCLUDE IT
0000AA 5860 4000 00000 168 L R6,PDEDSN POINT TO THE LEVEL
0000AE 1935 169 CR R3,R5 COMPARE LENGTHS
0000B0 4740 C0BE 000BE 170 BL LEVINC THIS LEVEL IS LONGER THAN DSN, NO MATCH
0000B4 0650 171 BCTR R5,0 MINUS ONE FOR THE EX
0000B6 4450 C9EC 009EC 172 EX R5,COMPARE CHECK THE LENGTHS
0000BA 4780 C0CA 000CA 173 BE LEVEND IT MATCHES, ALLOW IT
0000BE BF47 4019 00019 174 LEVINC ICM R4,B'0111',PDEDCHN GET THE NEXT LEVEL PDE POINTER
0000C2 4770 C0A0 000A0 175 BNZ LEVNEXT IF IT'S THERE, KEEP LOOKING
0000C6 47F0 C754 00754 176 B CHECKOUT NO MATCHES, EXCLUDE THIS DSNAME
0000CA 177 LEVEND DS 0H
                                178 DROP R4 FINISHED WITH THE PDE
                                179 *
                                180 * PROCESS THE ENDING KEYWORD
                                181 *
0000CA 9500 9027 00027 182 CLI ENDKEY+1,0 WAS ENDING SPECIFIED?
0000CE 4780 C106 00106 183 BE ENDEND NO, SKIP ON
0000D2 4140 9068 00068 184 LA R4,ENDING YES, POINT TO THE PDE
                                00000 185 USING PDEDSNAM,R4 SET ADDRESSABILITY
0000D6 1B55 186 ENDNEXT SR R5,R5 CLEAR FOR INSERT
0000D8 BF53 4004 00004 187 ICM R5,B'0011',PDEDSNL GET THE DSNAME LENGTH
0000DC 4780 C754 00754 188 BZ CHECKOUT END OF THE LINE, EXCLUDE IT
0000E0 5860 4000 00000 189 L R6,PDEDSN POINT TO THE ENDING
0000E4 1935 190 CR R3,R5 COMPARE LENGTHS
0000E6 4740 C0FA 000FA 191 BL ENDINC THIS ENDING IS LONGER THAN DSN, NO MATCH
0000EA 1827 192 LR R2,R7 POINT TO THE START OF THE DSNAME

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

0000EC 1A23                193      AR    R2,R3      POINT TO THE END
0000EE 1B25                194      SR    R2,R5      BACKUP TO COMPARE THIS LENGTH
0000F0 0650                195      BCTR  R5,0      MINUS ONE FOR THE EX
0000F2 4450 C9EC          009EC    196      EX    R5,COMPARE CHECK THE LENGTHS
0000F6 4780 C106          00106    197      BE    ENDEND    IT MATCHES, ALLOW IT
0000FA BF47 4019          00019    198 ENDINC ICM    R4,B'0111',PDEDCHN GET THE NEXT ENDING PDE POINTER
0000FE 4770 C0D6          000D6    199      BNZ   ENDNEXT  IF IT'S THERE, KEEP LOOKING
000102 47F0 C754          00754    200      B     CHECKOUT  NO MATCHES, EXCLUDE THIS DSNAME
000106                201 ENDEND  DS     0H
                202      DROP  R4      FINISHED WITH THE PDE
                203 *
                204 *      PROCESS THE CONTAINING KEYWORD
                205 *
000106 9500 9029          00029    206      CLI   CONTAINK+1,0 WAS CONTAINING SPECIFIED?
00010A 4780 C14E          0014E    207      BE    CONEND    NO, SKIP ON
00010E 4140 9084          00084    208      LA    R4,CONTAIN YES, POINT TO THE PDE
                00000 209      USING PDEDSNAM,R4 SET ADDRESSABILITY
000112 1B55                210 CONNEXT SR    R5,R5      CLEAR FOR INSERT
000114 BF53 4004          00004    211      ICM   R5,B'0011',PDEDSNL GET THE DSNAME LENGTH
000118 4780 C754          00754    212      BZ    CHECKOUT  END OF THE LINE, EXCLUDE IT
00011C 5860 4000          00000    213      L     R6,PDEDSN  POINT TO THE CONTAINING
000120 1935                214      CR    R3,R5      COMPARE LENGTHS
000122 4740 C142          00142    215      BL    CONINC    THIS CONTAIN IS LONGER THAN DSN, NO MATCH
000126 1817                216      LR    R1,R7      POINT TO THE START OF THE DSNAME
000128 1A13                217      AR    R1,R3      POINT TO THE END
00012A 1B15                218      SR    R1,R5      BACKUP TO COMPARE THIS LENGTH - LAST ONE
00012C 1827                219      LR    R2,R7      POINT TO THE START OF THE DSNAME
00012E 0650                220      BCTR  R5,0      MINUS ONE FOR THE EX
000130 4450 C9EC          009EC    221 CONCOMP EX    R5,COMPARE CHECK THE LENGTHS
000134 4780 C14E          0014E    222      BE    CONEND    IT MATCHES, ALLOW IT
000138 4122 0001          00001    223      LA    R2,1(R2)  CHECK THE WHOLE DSNAME
00013C 1921                224      CR    R2,R1      CHECK FOR THE END OF THE REAL DSN
00013E 47D0 C130          00130    225      BNH   CONCOMP  NOT THERE YET
000142 BF47 4019          00019    226 CONINC ICM    R4,B'0111',PDEDCHN GET THE NEXT CONTAINING PDE POINTER
000146 4770 C112          00112    227      BNZ   CONNEXT  IF IT'S THERE, KEEP LOOKING
00014A 47F0 C754          00754    228      B     CHECKOUT  NO MATCHES, EXCLUDE THIS DSNAME
00014E                229 CONEND  DS     0H
                230      DROP  R4      FINISHED WITH THE PDE
                231 *
                232 *      NOW THE BIG MESS, CHECK FOR LIMIT, AND'S, AND OR'S
                233 *
00014E 9500 9033          00033    234      CLI   LIMITK+1,0 WAS LIMIT SPECIFIED
000152 4780 C3A8          003A8    235      BE    LIMEND    NO, THEN THERE CAN BE NO AND'S OR OR'S
000156 94F7 B1A5          001A5    236      NI    VTCFMTCK,255-VTCFMTCC TURN OFF THE ROUTINE CALL FLAG
                237      VTCHL L      EVALUATE THE LIMIT 1=TRUE 0=FALSE
00015A 5810 90F8          000F8    238+     L     R1,SUBLOPER GET THE OPERATOR VALUE
00015E 4580 C83A          0083A    239+     BAL   R8,GETOPER TRANSLATE TO A NUMBER
000162 50F0 D098          00098    240+     ST    R15,REFOPER SAVE IT TOO
000166 4110 9100          00100    241+     LA    R1,SUBLVALU GET THE VALUE PDL
00016A 5010 D09C          0009C    242+     ST    R1,REFVAL  SAVE THAT ADDRESS THREE
00016E 9500 D07A          0007A    243+     CLI   FLAGNML,0 HAS IT BEEN CONVERTED?
000172 4770 C1B8          001B8    244+     BNE   VTP0005  YES, SKIP ALONG
000176 9201 D07A          0007A    245+     MVI   FLAGNML,1 NOTE IT AS CONVERTED
00017A 4580 C772          00772    246+     BAL   R8,PDLNUM GO CONVERT IT
00017E 50F0 D074          00074    247+     ST    R15,NUMBERL SAVE THE VALUE

```

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25	
000182	4140	90F0	000F0		248+	LA	R4,SUBLKEY	POINT TO THE IKJIDENT FOR THE KEYWORD	
000186	4580	C7E0	007E0		249+	BAL	R8,GETKEY	CONVERT TEXT TO A NUMERIC KEY	
00018A	42F0	D07B	0007B		250+	STC	R15,NUMKEYL	SAVE THAT NUMERIC KEY	
00018E	12FF				251+	LTR	R15,R15	WAS IT SUCCESSFUL?	
000190	4770	C1B8	001B8		252+	BNZ	VTP0005	YES, SKIP ALONG	
					253+*	ISSUE	A MESSAGE - A BAD LIM, AND, OR KEYWORD		
000194	D27B	B0FC	CA7A	000FC	00A7A	254+	MVC	MSGTEXT2,KEYERR	START THE ERROR MESSAGE
00019A	5814	0000	00000		255+	L	R1,0(R4)	POINT TO THE TEXT	
00019E	D205	B12D	1000	0012D	00000	256+	MVC	MSGTEXT2+49(6),0(R1)	THEN ADD IT TO THE MESSAGE
0001A4	4110	B0FC	000FC		257+	LA	R1,MSGTEXT2	POINT TO THE FIRST MESSAGE	
0001A8	1B00				258+	SR	R0,R0	NO SECOND LEVEL MESSAGE	
0001AA	9001	B078	00078		259+	STM	R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES	
					260+*	THEN	JUST CALL THE MESSAGE ISSUING ROUTINE		
0001AE	4110	B000	00000		261+	LA	R1,VTOCOM	POINT TO THE COMMON AREA	
0001B2	58F0	B01C	0001C		262+	L	R15,VADMSG	POINT TO THE ROUTINE	
0001B6	05EF				263+	BALR	R14,R15	THEN CALL IT	
0001B8					264+VTP0005	DS	0H		
0001B8	1B11				265+	SR	R1,R1	CLEAR A REGISTER	
0001BA	BF11	D07B	0007B		266+	ICM	R1,1,NUMKEYL	GET THE KEYWORD VALUE	
0001BE	4780	C1D2	001D2		267+	BZ	VTE0005	IF NOT SET, SKIP THE EVALUATION	
0001C2	5010	D094	00094		268+	ST	R1,REFKEY	SAVE THE ADDRESS	
0001C6	4110	D074	00074		269+	LA	R1,NUMBERL	GET THE ADDRESS OF CONVERTED NUMBER	
0001CA	5010	D0A0	000A0		270+	ST	R1,REFNUM	SAVE IT'S ADDRESS	
0001CE	4580	C3AC	003AC		271+	BAL	R8,LIMEVAL	GO EVALUATE THE EXPRESSION	
0001D2					272+VTE0005	DS	0H		
0001D2	50F0	D070	00070		273	ST	R15,LIMVAL	SAVE THE ANSWER	
					274	VTANDOR	1	CHECK AND1 OR OR1	
0001D6	9500	9035	00035		275+	CLI	ANDOR1K+1,0	WAS THIS KEYWORD SET?	
0001DA	4780	C39E	0039E		276+	BE	LIMCOMP	NO, JUST CHECK THE FINAL RESULT	
0001DE	5810	9110	00110		277+	L	R1,SUB1OPER	GET THE OPERATOR VALUE	
0001E2	4580	C83A	0083A		278+	BAL	R8,GETOPER	TRANSLATE TO A NUMBER	
0001E6	50F0	D098	00098		279+	ST	R15,REFOPER	SAVE IT TOO	
0001EA	4110	9118	00118		280+	LA	R1,SUB1VALU	GET THE VALUE PDL	
0001EE	5010	D09C	0009C		281+	ST	R1,REFVAL	SAVE THAT ADDRESS THREE	
0001F2	9500	D082	00082		282+	CLI	FLAGNM1,0	HAS IT BEEN CONVERTED?	
0001F6	4770	C23C	0023C		283+	BNE	VTP0009	YES, SKIP ALONG	
0001FA	9201	D082	00082		284+	MVI	FLAGNM1,1	NOTE IT AS CONVERTED	
0001FE	4580	C772	00772		285+	BAL	R8,PDLNUM	GO CONVERT IT	
000202	50F0	D07C	0007C		286+	ST	R15,NUMBER1	SAVE THE VALUE	
000206	4140	9108	00108		287+	LA	R4,SUB1KEY	POINT TO THE IKJIDENT FOR THE KEYWORD	
00020A	4580	C7E0	007E0		288+	BAL	R8,GETKEY	CONVERT TEXT TO A NUMERIC KEY	
00020E	42F0	D083	00083		289+	STC	R15,NUMKEY1	SAVE THAT NUMERIC KEY	
000212	12FF				290+	LTR	R15,R15	WAS IT SUCCESSFUL?	
000214	4770	C23C	0023C		291+	BNZ	VTP0009	YES, SKIP ALONG	
					292+*	ISSUE	A MESSAGE - A BAD LIM, AND, OR KEYWORD		
000218	D27B	B0FC	CA7A	000FC	00A7A	293+	MVC	MSGTEXT2,KEYERR	START THE ERROR MESSAGE
00021E	5814	0000	00000		294+	L	R1,0(R4)	POINT TO THE TEXT	
000222	D205	B12D	1000	0012D	00000	295+	MVC	MSGTEXT2+49(6),0(R1)	THEN ADD IT TO THE MESSAGE
000228	4110	B0FC	000FC		296+	LA	R1,MSGTEXT2	POINT TO THE FIRST MESSAGE	
00022C	1B00				297+	SR	R0,R0	NO SECOND LEVEL MESSAGE	
00022E	9001	B078	00078		298+	STM	R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES	
					299+*	THEN	JUST CALL THE MESSAGE ISSUING ROUTINE		
000232	4110	B000	00000		300+	LA	R1,VTOCOM	POINT TO THE COMMON AREA	
000236	58F0	B01C	0001C		301+	L	R15,VADMSG	POINT TO THE ROUTINE	
00023A	05EF				302+	BALR	R14,R15	THEN CALL IT	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25
00023C          303+VTP0009  DS  0H
00023C 1B11          304+          SR  R1,R1          CLEAR A REGISTER
00023E BF11 D083      00083  305+          ICM R1,1,NUMKEY1    GET THE KEYWORD VALUE
000242 4780 C256      00256  306+          BZ  VTE0009      IF NOT SET, SKIP THE EVALUATION
000246 5010 D094      00094  307+          ST  R1,REFKEY    SAVE THE ADDRESS
00024A 4110 D07C      0007C  308+          LA  R1,NUMBER1   GET THE ADDRESS OF CONVERTED NUMBER
00024E 5010 D0A0      000A0  309+          ST  R1,REFNUM    SAVE IT'S ADDRESS
000252 4580 C3AC      003AC  310+          BAL R8,LIMEVAL   GO EVALUATE THE EXPRESSION
000256          311+VTE0009  DS  0H
000256 9501 9035      00035  312+          CLI ANDOR1K+1,1  WAS IT AN AND ?
00025A 4780 C266      00266  313+          BE  VTA0008      YES, DO THE AND
00025E 56F0 D070      00070  314+          O   R15,LIMVAL   NO, OR IT
000262 47F0 C26A      0026A  315+          B   VTE0008      FINISHED WITH THIS EXPRESSION
000266 54F0 D070      00070  316+VTA0008  N   R15,LIMVAL   AND THE EXPRESSION VALUE
00026A 50F0 D070      00070  317+VTE0008  ST  R15,LIMVAL   SAVE THE VALUE
          318          VTANDOR 2      CHECK AND2 OR OR2
00026E 9500 9037      00037  319+          CLI ANDOR2K+1,0  WAS THIS KEYWORD SET?
000272 4780 C39E      0039E  320+          BE  LIMCOMP      NO, JUST CHECK THE FINAL RESULT
000276 5810 9128      00128  321+          L   R1,SUB2OPER  GET THE OPERATOR VALUE
00027A 4580 C83A      0083A  322+          BAL R8,GETOPER   TRANSLATE TO A NUMBER
00027E 50F0 D098      00098  323+          ST  R15,REFOPER  SAVE IT TOO
000282 4110 9130      00130  324+          LA  R1,SUB2VALU  GET THE VALUE PDL
000286 5010 D09C      0009C  325+          ST  R1,REFVAL   SAVE THAT ADDRESS THREE
00028A 9500 D08A      0008A  326+          CLI FLAGNM2,0    HAS IT BEEN CONVERTED?
00028E 4770 C2D4      002D4  327+          BNE VTP0013      YES, SKIP ALONG
000292 9201 D08A      0008A  328+          MVI FLAGNM2,1    NOTE IT AS CONVERTED
000296 4580 C772      00772  329+          BAL R8,PDLNUM    GO CONVERT IT
00029A 50F0 D084      00084  330+          ST  R15,NUMBER2  SAVE THE VALUE
00029E 4140 9120      00120  331+          LA  R4,SUB2KEY   POINT TO THE IKJIDENT FOR THE KEYWORD
0002A2 4580 C7E0      007E0  332+          BAL R8,GETKEY    CONVERT TEXT TO A NUMERIC KEY
0002A6 42F0 D08B      0008B  333+          STC R15,NUMKEY2  SAVE THAT NUMERIC KEY
0002AA 12FF          334+          LTR R15,R15      WAS IT SUCCESSFUL?
0002AC 4770 C2D4      002D4  335+          BNZ VTP0013      YES, SKIP ALONG
          336+*          ISSUE A MESSAGE - A BAD LIM, AND, OR KEYWORD
0002B0 D27B B0FC CA7A 000FC 00A7A 337+          MVC MSGTEXT2,KEYERR  START THE ERROR MESSAGE
0002B6 5814 0000          00000  338+          L   R1,0(R4)     POINT TO THE TEXT
0002BA D205 B12D 1000 0012D 00000 339+          MVC MSGTEXT2+49(6),0(R1) THEN ADD IT TO THE MESSAGE
0002C0 4110 B0FC          000FC  340+          LA  R1,MSGTEXT2  POINT TO THE FIRST MESSAGE
0002C4 1B00          341+          SR  R0,R0        NO SECOND LEVEL MESSAGE
0002C6 9001 B078          00078  342+          STM R0,R1,MSGADDRS  SAVE THE MESSAGE ADDRESSES
          343+*          THEN JUST CALL THE MESSAGE ISSUING ROUTINE
0002CA 4110 B000          00000  344+          LA  R1,VTOCOM    POINT TO THE COMMON AREA
0002CE 58F0 B01C          0001C  345+          L   R15,VADMSG   POINT TO THE ROUTINE
0002D2 05EF          346+          BALR R14,R15     THEN CALL IT
0002D4          347+VTP0013  DS  0H
0002D4 1B11          348+          SR  R1,R1        CLEAR A REGISTER
0002D6 BF11 D08B      0008B  349+          ICM R1,1,NUMKEY2  GET THE KEYWORD VALUE
0002DA 4780 C2EE      002EE  350+          BZ  VTE0013      IF NOT SET, SKIP THE EVALUATION
0002DE 5010 D094      00094  351+          ST  R1,REFKEY    SAVE THE ADDRESS
0002E2 4110 D084      00084  352+          LA  R1,NUMBER2   GET THE ADDRESS OF CONVERTED NUMBER
0002E6 5010 D0A0      000A0  353+          ST  R1,REFNUM    SAVE IT'S ADDRESS
0002EA 4580 C3AC      003AC  354+          BAL R8,LIMEVAL   GO EVALUATE THE EXPRESSION
0002EE          355+VTE0013  DS  0H
0002EE 9501 9037      00037  356+          CLI ANDOR2K+1,1  WAS IT AN AND ?
0002F2 4780 C2FE      002FE  357+          BE  VTA0012      YES, DO THE AND

```


LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25	
0002F6	56F0	D070	00070		358+	O	R15,LIMVAL	NO, OR IT	
0002FA	47F0	C302	00302		359+	B	VTE0012	FINISHED WITH THIS EXPRESSION	
0002FE	54F0	D070	00070		360+VTA0012	N	R15,LIMVAL	AND THE EXPRESSION VALUE	
000302	50F0	D070	00070		361+VTE0012	ST	R15,LIMVAL	SAVE THE VALUE	
					362		VTANDOR 3	CHECK AND3 OR OR3	
000306	9500	9039	00039		363+	CLI	ANDOR3K+1,0	WAS THIS KEYWORD SET?	
00030A	4780	C39E	0039E		364+	BE	LIMCOMP	NO, JUST CHECK THE FINAL RESULT	
00030E	5810	9140	00140		365+	L	R1,SUB3OPER	GET THE OPERATOR VALUE	
000312	4580	C83A	0083A		366+	BAL	R8,GETOPER	TRANSLATE TO A NUMBER	
000316	50F0	D098	00098		367+	ST	R15,REFOPER	SAVE IT TOO	
00031A	4110	9148	00148		368+	LA	R1,SUB3VALU	GET THE VALUE PDL	
00031E	5010	D09C	0009C		369+	ST	R1,REFVAL	SAVE THAT ADDRESS THREE	
000322	9500	D092	00092		370+	CLI	FLAGNM3,0	HAS IT BEEN CONVERTED?	
000326	4770	C36C	0036C		371+	BNE	VTP0017	YES, SKIP ALONG	
00032A	9201	D092	00092		372+	MVI	FLAGNM3,1	NOTE IT AS CONVERTED	
00032E	4580	C772	00772		373+	BAL	R8,PDLNUM	GO CONVERT IT	
000332	50F0	D08C	0008C		374+	ST	R15,NUMBER3	SAVE THE VALUE	
000336	4140	9138	00138		375+	LA	R4,SUB3KEY	POINT TO THE IKJIDENT FOR THE KEYWORD	
00033A	4580	C7E0	007E0		376+	BAL	R8,GETKEY	CONVERT TEXT TO A NUMERIC KEY	
00033E	42F0	D093	00093		377+	STC	R15,NUMKEY3	SAVE THAT NUMERIC KEY	
000342	12FF				378+	LTR	R15,R15	WAS IT SUCCESSFUL?	
000344	4770	C36C	0036C		379+	BNZ	VTP0017	YES, SKIP ALONG	
					380+*		ISSUE A MESSAGE - A BAD LIM, AND, OR KEYWORD		
000348	D27B	B0FC	CA7A	000FC	00A7A	381+	MVC	MSGTEXT2,KEYERR	START THE ERROR MESSAGE
00034E	5814	0000		00000		382+	L	R1,0(R4)	POINT TO THE TEXT
000352	D205	B12D	1000	0012D	00000	383+	MVC	MSGTEXT2+49(6),0(R1)	THEN ADD IT TO THE MESSAGE
000358	4110	B0FC		000FC		384+	LA	R1,MSGTEXT2	POINT TO THE FIRST MESSAGE
00035C	1B00					385+	SR	R0,R0	NO SECOND LEVEL MESSAGE
00035E	9001	B078		00078		386+	STM	R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES
						387+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
000362	4110	B000		00000		388+	LA	R1,VTOCOM	POINT TO THE COMMON AREA
000366	58F0	B01C		0001C		389+	L	R15,VADMSG	POINT TO THE ROUTINE
00036A	05EF					390+	BALR	R14,R15	THEN CALL IT
00036C						391+VTP0017	DS	0H	
00036C	1B11					392+	SR	R1,R1	CLEAR A REGISTER
00036E	BF11	D093		00093		393+	ICM	R1,1,NUMKEY3	GET THE KEYWORD VALUE
000372	4780	C386		00386		394+	BZ	VTE0017	IF NOT SET, SKIP THE EVALUATION
000376	5010	D094		00094		395+	ST	R1,REFKEY	SAVE THE ADDRESS
00037A	4110	D08C		0008C		396+	LA	R1,NUMBER3	GET THE ADDRESS OF CONVERTED NUMBER
00037E	5010	D0A0		000A0		397+	ST	R1,REFNUM	SAVE IT'S ADDRESS
000382	4580	C3AC		003AC		398+	BAL	R8,LIMEVAL	GO EVALUATE THE EXPRESSION
000386						399+VTE0017	DS	0H	
000386	9501	9039		00039		400+	CLI	ANDOR3K+1,1	WAS IT AN AND ?
00038A	4780	C396		00396		401+	BE	VTA0016	YES, DO THE AND
00038E	56F0	D070		00070		402+	O	R15,LIMVAL	NO, OR IT
000392	47F0	C39A		0039A		403+	B	VTE0016	FINISHED WITH THIS EXPRESSION
000396	54F0	D070		00070		404+VTA0016	N	R15,LIMVAL	AND THE EXPRESSION VALUE
00039A	50F0	D070		00070		405+VTE0016	ST	R15,LIMVAL	SAVE THE VALUE
00039E	58F0	D070		00070		406 LIMCOMP	L	R15,LIMVAL	GET THE RESULT OF ALL THIS
0003A2	12FF					407	LTR	R15,R15	TEST IT
0003A4	4780	C754		00754		408	BZ	CHECKOUT	IT GETS EXCLUDED
0003A8						409 LIMEND	DS	0H	
0003A8	47F0	C75C		0075C		410	B	CHECKIN	ALL TESTS PASSED, INCLUDE THIS ONE

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				412 *		
				413 *	EVALUATION ROUTINE FOR	KEYWORD OPER VALUE
				414 *		
0003AC	5850 D094	00094		415	LIMEVAL L R5,REFKEY	GET THE KEYWORD VALUE
0003B0	4C50 CA36	00A36		416	MH R5,H12	MULTIPLY IT BY 12
0003B4	5A50 B1AC	001AC		417	A R5,ATABTITL	THEN RELOCATE IT
0003B8	1BAA			418	SR R10,R10	CLEAR THE ROUTINE POINTER
0003BA	9180 5001	00001		419	TM 1(R5),X'80'	IS IT A FORMATTED ITEM?
0003BE	4710 C406	00406		420	BO LIMFORM	YES, GO DO IT
0003C2	1B66			421	SR R6,R6	CLEAR A REGISTER
0003C4	4365 0002	00002		422	IC R6,2(R5)	GET THE OFFSET INTO VTFMT
0003C8	43A5 0001	00001		423	IC R10,1(R5)	GET THE ROUTINE NUMBER
0003CC	12AA			424	LTR R10,R10	SEE IF IT'S A GOOD NUMBER
0003CE	4720 C3E2	003E2		425	BP LIMEVAL1	IT'S ALL RIGHT
				426	LIMABEND ABEND 702,DUMP	CRASH AND BURN
0003D2				427+	LIMABEND DS 0H	00400002
0003D2	4110 02BE	002BE		428+	LA 1,702	LOAD PARAMETER REG 1 01900002
0003D6	4100 0080	00080		429+	LA 0,128(0,0)	PICK UP DUMP/STEP/DUMPOPTS YM1995 01800002
0003DA	8900 0018	00018		430+	SLL 0,24(0)	SHIFT TO HIGH ORDER 01850002
0003DE	1610			431+	OR 1,0	OR IN WITH COMPCODE 01900002
0003E0	0A0D			432+	SVC 13	LINK TO ABEND ROUTINE 02050002
0003E2	47FA C3E2	003E2		433	LIMEVAL1 B *(R10)	AND GO TO IT
0003E6	47F0 C4BE	004BE		434	B LIMDATE	CDATE, EXPDT, REFDT
0003EA	47F0 C506	00506		435	B LIMLUSE	*** DUMMY ENTRY ***
0003EE	47F0 C406	00406		436	B LIMFORM	FORM , USED SPACE
0003F2	47F0 C406	00406		437	B LIMFORM	UNUSED SPACE
0003F6	47F0 C406	00406		438	B LIMFORM	PCT USED
0003FA	47F0 C52A	0052A		439	B LIMEXT	EXTENTS
0003FE	47F0 C538	00538		440	B LIMBLREC	LRECL, BLKSZ, SEC Q
000402	47F0 C5EC	005EC		441	B LIMCCHH	CCHH CHECKING
				443 *		
				444 *	PERFORM THE FORMATTED ITEM CHECKING	
				445 *		
000406				446	LIMFORM DS 0H	
000406	9108 B1A5	001A5		447	TM VTCFMTCK,VTCFMTCC	WAS FORMAT CALLED BEFORE FOR THIS DS
00040A	4710 C41C	0041C		448	BO LIMFCALD	YES, DON'T CALL IT AGAIN
				449	VTCALL FORM	NO, CALL IT TO GET THE ITEMS
00040E	4110 B000	00000		450+	LA R1,VTOCOM	POINT TO THE COMMON AREA
000412	58F0 B02C	0002C		451+	L R15,VADFORM	POINT TO THE ROUTINE
000416	05EF			452+	BALR R14,R15	THEN CALL IT
000418	9688 B1A5	001A5		453	OI VTCFMTCK,VTCFMTCC+VTCFMTCD	THEN SET THE SWITCHES
00041C	1B22			454	LIMFCALD SR R2,R2	CLEAR A WORK REG
00041E	5A60 B198	00198		455	A R6,FORMATAD	RELOCATE THE BLOCK
000422	47FA C426	00426		456	B *+4(R10)	AND GO TO IT
000426	47F0 C43E	0043E		457	B LIMFORMA	FORMATTED ITEM
00042A	47F0 C3D2	003D2		458	B LIMABEND	CDATE, EXPDT
00042E	47F0 C3D2	003D2		459	B LIMABEND	LAST USE DATE
000432	47F0 C574	00574		460	B LIMALLOC	ALLOC, USED SPACE
000436	47F0 C586	00586		461	B LIMUNUSD	UNUSED SPACE
00043A	47F0 C5A8	005A8		462	B LIMPCT	PCT USED

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

00043E          463 LIMFORMA DS  0H
00043E 4325 0002      00002      464          IC  R2,2(R5)      GET THE OFFSET IN VTFMT
000442 5A20 B198      00198      465          A   R2,FORMATAD    THEN RELOCATE IT
000446 5840 D09C      0009C      466          L   R4,REFVAL     GET THE VALUE PDE
00044A 4834 0004      00004      467          LH  R3,4(R4)     GET THE LENGTH OF THE STRING
00044E 5814 0000      00000      468          L   R1,0(R4)     AND ITS ADDRESS
469 *
470 *          DO THE ACTUAL COMPARE
471 *
000452 0630          472          BCTR R3,0        DOWN ONE FOR AN EX
000454 4430 C87A      0087A      473          EX  R3,COMPLIM   COMPARE AS SPECIFIED
000458          474 COMPDONE DS  0H        GET THE OPERATOR ADDRESS
000458 4740 C498      00498      475          BL  COMPLOW     CHECK THE OPERATOR, VALUE LESS THAN ITEM
00045C 4780 C47C      0047C      476          BE  COMPEQ      CHECK THE OPERATOR, KEYWORD EQUALS VALUE
477 *          KEYWORD IS GREATER THAN THE VALUE
000460 9502 D09B      0009B      478 COMPHI  CLI  REFOPER+3,NE  WAS OPERATOR NE
000464 4780 C4B4      004B4      479          BE  COMPYES     HIGH SATISFIES THE EXPRESSION
000468 9505 D09B      0009B      480          CLI REFOPER+3,GT  ALSO FOR GT
00046C 4780 C4B4      004B4      481          BE  COMPYES     HIGH SATISFIES THE EXPRESSION
000470 9506 D09B      0009B      482          CLI REFOPER+3,GE  AND FOR GE
000474 4780 C4B4      004B4      483          BE  COMPYES     HIGH SATISFIES THE EXPRESSION
000478 47F0 C4BA      004BA      484          B   COMPNO      THIS ONE DOESN'T FIT
485 *          KEYWORD IS EQUAL TO THE VALUE
00047C 9501 D09B      0009B      486 COMPEQ  CLI  REFOPER+3,EQ  WAS OPERATOR EQ
000480 4780 C4B4      004B4      487          BE  COMPYES     EQ SATISFIES THE EXPRESSION
000484 9504 D09B      0009B      488          CLI REFOPER+3,LE  ALSO FOR LE
000488 4780 C4B4      004B4      489          BE  COMPYES     EQ SATISFIES THE EXPRESSION
00048C 9506 D09B      0009B      490          CLI REFOPER+3,GE  AND FOR GE
000490 4780 C4B4      004B4      491          BE  COMPYES     EQ SATISFIES THE EXPRESSION
000494 47F0 C4BA      004BA      492          B   COMPNO      THIS ONE DOESN'T FIT
493 *          KEYWORD IS LESS THAN THE VALUE
000498 9502 D09B      0009B      494 COMPLow  CLI  REFOPER+3,NE  WAS OPERATOR NE
00049C 4780 C4B4      004B4      495          BE  COMPYES     LOW SATISFIES THE EXPRESSION
0004A0 9503 D09B      0009B      496          CLI REFOPER+3,LT  ALSO FOR LT
0004A4 4780 C4B4      004B4      497          BE  COMPYES     LOW SATISFIES THE EXPRESSION
0004A8 9504 D09B      0009B      498          CLI REFOPER+3,LE  AND FOR LE
0004AC 4780 C4B4      004B4      499          BE  COMPYES     LOW SATISFIES THE EXPRESSION
0004B0 47F0 C4BA      004BA      500          B   COMPNO      THIS ONE DOESN'T FIT
501 *
502 *          IT FITS OR IT DOESN'T
503 *
0004B4 41F0 0001      00001      504 COMPYES  LA  R15,1        SET A TRUE VALUE
0004B8 07F8          505          BR  R8          THEN RETURN
0004BA 1BFF          506 COMPNO   SR  R15,R15     SET A FALSE VALUE
0004BC 07F8          507          BR  R8          THEN RETURN
508 *
509 *          SPECIAL ROUTINES TO CHECK NON-FORMATTED ITEMS
510 *
0004BE          511 LIMDATE DS  0H
512 *
513 *          COMPARE DATES
514 *
0004BE 41E0 7035      00035      515          LA  R14,DS1CREDT POINT TO CREATION DATE
0004C2 9503 D097      00097      516          CLI REFKEY+3,CDATE IS THAT IT?
0004C6 4780 C4DA      004DA      517          BE  LIMDGET     YES, THIS IS IT

```

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25	
0004CA	41E0	7038	00038		518	LA	R14,DS1EXPDT	POINT TO EXPIRATION DATE	
0004CE	9505	D097	00097		519	CLI	REFKEY+3,EXPDT	IS THAT IT?	
0004D2	4780	C4DA	004DA		520	BE	LIMDGET	YES, THIS IS IT	
0004D6	41E0	704B	0004B		521	LA	R14,DS1REFD	NO, USE REFERENCE DATE	
0004DA	1BFF				522	LIMDGET SR	R15,R15	CLEAR A WORK REG	
0004DC	43FE	0000	00000		523	IC	R15,0(R14)	GET THE YEAR	
0004E0	4CF0	CA40	00A40		524	MH	R15,H1000	TIMES 1000	
0004E4	1B11				525	SR	R1,R1	CLEAR ANOTHER WORK REG	
0004E6	BF13	E001	00001		526	ICM	R1,3,1(R14)	GET THE DAYS	
0004EA	1AF1				527	AR	R15,R1	PUT THE DATE TOGETHER	
0004EC	5810	D0A0	000A0		528	L	R1,REFNUM	GET THE ADDRESS OF THE VALUE	
0004F0	5811	0000	00000		529	L	R1,0(R1)	GET THE VALUE ITSELF	
					530	C	R1,=A(&WIN19XX*1000)	BELOW 19XX WINDOW? Y2K DEC97	
0004F4	5910	CB40	00B40		531+	C	R1,=A(72*1000)	BELOW 19XX WINDOW? Y2K DEC97	
0004F8	47B0	C500	00500		532	BNL	*+8	NO, THEN 19XX Y2K DEC97	
0004FC	5A10	CB44	00B44		533	A	R1,=F'100000'	YES, THEN 20XX Y2K DEC97	
000500	19F1				534	CR	R15,R1	COMPARE THEM	
000502	47F0	C458	00458		535	B	COMPDONE	GO CHECK OPERANDS	
000506					536	LIMLUSE DS	0H		
					537	*			
					538	*	LAST USE DATE		
					539	*			
000506	1BEE				540	SR	R14,R14	CLEAR THE DATE	
000508	D503	CA38	704B	00A38	0004B	541	CLC	ZERO,75(R7)	CHECK FOR NO DATA
00050E	4780	C51C		0051C		542	BE	LIMLUCMP	RIGHT, SKIP ON
000512	D202	D0CD	704B	000CD	0004B	543	MVC	CHEKDBLW+5(3),75(R7)	MOVE IN THE LAST USE DATE
000518	4FE0	D0C8		000C8		544	CVB	R14,CHEKDBLW	CONVERT IT TO BINARY
00051C	5810	D0A0	000A0		545	LIMLUCMP L	R1,REFNUM	GET THE ADDRESS OF THE VALUE	
000520	5811	0000	00000		546	L	R1,0(R1)	GET THE VALUE	
000524	19E1				547	CR	R14,R1	DO THE COMPARE	
000526	47F0	C458	00458		548	B	COMPDONE	THEN CHASE DOWN THE OPERANDS	
00052A					549	LIMEXT DS	0H		
					550	*			
					551	*	EXTENTS		
					552	*			
00052A	5810	D0A0	000A0		553	L	R1,REFNUM	GET THE COMPARE VALUE ADDRESS	
00052E	D500	703B	1003	0003B	00003	554	CLC	DS1NOEPV,3(R1)	COMPARE THEM
000534	47F0	C458	00458		555	B	COMPDONE	GO CHECK OPERANDS	
000538					556	LIMBLREC DS	0H		
					557	*			
					558	*	LRECL, BLKSZ, SECQ		
					559	*			
000538	5810	D0A0	000A0		560	L	R1,REFNUM	GET THE ADDRESS OF THE COMPARE VALUE	
00053C	58F1	0000	00000		561	L	R15,0(R1)	THEN GET THE VALUE ITSELF	
000540	950C	D097	00097		562	CLI	REFKEY+3,BLKSZ	BLOCK SIZE?	
000544	4770	C552	00552		563	BNE	LIMB1	NO, KEEP CHECKING	
000548	4810	7056	00056		564	LH	R1,DS1BLKL	COMPARE TO THE BLOCK SIZE	
00054C	191F				565	CR	R1,R15	COMPARE THEM	
00054E	47F0	C458	00458		566	B	COMPDONE	GO SIFT THROUGH THE OPERANDS	
000552	950D	D097	00097		567	LIMB1	CLI REFKEY+3,LRECL	LOGICAL RECORD LENGTH	
000556	4770	C564	00564		568	BNE	LIMB2	NO, KEEP GOING	
00055A	4810	7058	00058		569	LH	R1,DS1LRECL	COMPARE TO THE LRECL	
00055E	191F				570	CR	R1,R15	COMPARE THEM	
000560	47F0	C458	00458		571	B	COMPDONE	GO CHECK THE OPERANDS	
000564	D201	D06E	7060	0006E	00060	572	LIMB2	MVC HWORK,DS1SCALO+2	GET THE SECONDARY QUANTITY

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
00056A 4810 D06E      0006E      573          LH   R1,HWORK      DO THE COMPARE
00056E 191F              574          CR   R1,R15        COMPARE THEM
000570 47F0 C458      00458      575          B    COMPDONE      THEN CHECK THE OPERANDS
576 *
577 *          SPACE CHECKING ROUTINES
578 *
000574      579 LIMALLOC DS   0H
580 *
581 *          ALLOC AND USED
582 *
000574 5810 D0A0      000A0      583          L    R1,REFNUM     GET THE ADDRESS OF THE CONVERTED NUMBER
000578 5811 0000      00000      584          L    R1,0(R1)      GET THE VALUE
00057C 58F6 0000      00000      585          L    R15,0(R6)     GET THE AMOUNT
000580 19F1              586          CR   R15,R1        COMPARE THEM
000582 47F0 C458      00458      587          B    COMPDONE      THEN CHECK THE OPERANDS
000586      588 LIMUNUSD DS   0H
589 *
590 *          UNUSED
591 *
000586 5860 B198      00198      592          L    R6,FORMATAD   POINT TO THE FORMATTED VTOC
00058A BFEF 6008      00008      593          USING VTFMT,R6     SET ADDRESSABILITY
00058E 4740 C59A      0059A      594          ICM  R14,15,VTFUSED GET THE AMOUNT USED
000592 58E0 6004      00004      595          BM   LIMUNUAL      IF MINUS, WE DON'T KNOW
000596 5BE0 6008      00008      596          L    R14,VTFALLOC  GET ALLOC
00059A 5810 D0A0      000A0      597          S    R14,VTFUSED   MINUS THE AMOUNT USED
00059E 5811 0000      00000      598 LIMUNUAL L    R1,REFNUM     GET THE ENTERED VALUE
0005A2 19E1              599          L    R1,0(R1)      NOW ITS VALUE FOR REAL
0005A4 47F0 C458      00458      600          CR   R14,R1        COMPARE THE VALUES
0005A8      601          B    COMPDONE      THEN GO CHECK THE OPERANDS
602 LIMPCT DS   0H
603 *
604 *          PER CENT
605 *
0005A8 5860 B198      00198      606          L    R6,FORMATAD   POINT TO THE FORMATTED VTOC
0005AC 1BEE              607          USING VTFMT,R6     SET ADDRESSABILITY
0005AE BFFF 6008      00008      608          SR   R14,R14       CLEAR A REGISTER
0005B2 4740 C5CE      005CE      609          ICM  R15,15,VTFUSED GET THE AMOUNT USED
0005B6 D503 6004 CA38 00004 00A38 610          BM   LIMP100       IF UNKNOWN USED, SET 100 PER CENT
0005BC 4770 C5D6      005D6      611          CLC  VTFALLOC,ZERO ZERO ALLOCATED SPACE?
0005C0 D503 6008 CA38 00008 00A38 612          BNE  LIMPCTOK      NO, CONTINUE
0005C6 4780 C5DE      005DE      613          CLC  VTFUSED,ZERO  ZERO USED SPACE?
614          BE   LIMPCOMP     YES, PCT IS ZERO
615 *          ZERO ALLOCATED, NONZERO USED, INCLUDE THIS ONE
0005CA 47F0 C4B4      004B4      616          B    COMPYES
0005CE 41F0 0064      00064      617 LIMP100 LA   R15,100       SET UP 100 PER CENT
0005D2 47F0 C5DE      005DE      618          B    LIMPCOMP      GO COMPARE
0005D6 5CE0 CA3C      00A3C      619 LIMPCTOK M    R14,F100       MULTIPLY BY 100 PERCENT
0005DA 5DE0 6004      00004      620          D    R14,VTFALLOC  DIVIDE BY THE ALLOCATION
0005DE 5810 D0A0      000A0      621 LIMPCOMP L    R1,REFNUM     GET THE VALUE ADDRESS
0005E2 5811 0000      00000      622          L    R1,0(R1)      THEN THE VALUE
0005E6 19F1              623          CR   R15,R1        THEN COMPARE THEM
0005E8 47F0 C458      00458      624          B    COMPDONE      THEN GO SIFT THROUGH THE OPERANDS
625 *
626 *          CCHH CHECKING IS NOT QUITE STANDARD BECAUSE THERE MAY
627 *          MAY BE UP TO 16 EXTENTS TO COMPARE.  THE DATA MAY BE

```

ASM 0201 13.38 01/07/25

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				628 *		CC OR CCHH FORMATS. THE DATA SET MAY BE EQUAL TO,	
				629 *		LESS THAN, AND GREATER THAN ANY PARTICULAR VALUE.	
				630 *			
0005EC				631	LIMCCHH	DS 0H	
				632 *			
				633 *		FIRST SEE IF THE CCHH WAS CONVERTED	
				634 *		CONVERT IT IF NOT, SKIP IF IT'S DONE	
				635 *			
0005EC	5850	D0A0	000A0	636	L	R5,REFNUM GET THE ADDRESS OF THE COMPARISON VALUE	
0005F0	5845	0000	00000	637	L	R4,0(R5) GET THE VALUE	
0005F4	BF63	5004	00004	638	ICM	R6,3,4(R5) GET THE COMPARE LENGTH	
0005F8	4720	C680	00680	639	BP	LIMCSET IF IT'S SET, THE CONVERSION IS DONE	
				640 *		IT WASN'T SET, CONVERT FROM CHARS TO BINARY	
0005FC	5810	D09C	0009C	641	L	R1,REFVAL GET THE ADDRESS OF THE IKJIDENT	
000600	5821	0000	00000	642	L	R2,0(R1) POINT TO THE TEXT	
000604	4831	0004	00004	643	LH	R3,4(R1) GET THE LENGTH OF THE TEXT	
				644 *		IT SHOULD BE 4 OR 8 CHARACTERS	
000608	D707	B070	B070	00070	00070	645 XC DOUBLE,DOUBLE CLEAR OUT A PLACE TO WORK	
00060E	4930	CA2E	00A2E	646	CH	R3,H4 IS IT A CYLINDER ONLY?	
000612	4720	C636	00636	647	BH	LIMCCON2 NO, TRY FOR A CCHH	
000616	4780	C62E	0062E	648	BE	LIMCCON1 YES, JUST CONVERT IT	
				649		VTOCMSG CCHHLEN LESS THAN FOUR CHARS, ISSUE A MSG	
00061A	4110	CAB2	00AB2	650+	LA	R1,CCHHLEN POINT TO THE FIRST MESSAGE	
00061E	1B00			651+	SR	R0,R0 NO SECOND LEVEL MESSAGE	
000620	9001	B078	00078	652+	STM	R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				653+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
000624	4110	B000	00000	654+	LA	R1,VTOCOM POINT TO THE COMMON AREA	
000628	58F0	B01C	0001C	655+	L	R15,VADMSG POINT TO THE ROUTINE	
00062C	05EF			656+	BALR	R14,R15 THEN CALL IT	
00062E	4160	0001	00001	657	LIMCCON1	LA R6,1 SET THE COMPARE LENGTH	
000632	47F0	C662	00662	658	B	LIMCMOVE GO MOVE IT IN	
000636	4160	0003	00003	659	LIMCCON2	LA R6,3 SET THE COMPARE LENGTH	
00063A	4930	CA32	00A32	660	CH	R3,H8 WAS IT A CCHH?	
00063E	4780	C662	00662	661	BE	LIMCMOVE YES, JUST THE RIGHT LENGTH	
				662		VTOCMSG CCHHLEN WARN THE PERSON	
000642	4110	CAB2	00AB2	663+	LA	R1,CCHHLEN POINT TO THE FIRST MESSAGE	
000646	1B00			664+	SR	R0,R0 NO SECOND LEVEL MESSAGE	
000648	9001	B078	00078	665+	STM	R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				666+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
00064C	4110	B000	00000	667+	LA	R1,VTOCOM POINT TO THE COMMON AREA	
000650	58F0	B01C	0001C	668+	L	R15,VADMSG POINT TO THE ROUTINE	
000654	05EF			669+	BALR	R14,R15 THEN CALL IT	
000656	4930	CA32	00A32	670	CH	R3,H8 CHECK AGAIN	
00065A	4740	C662	00662	671	BL	LIMCMOVE IS IT OVER 8 CHARS?	
00065E	4830	CA32	00A32	672	LH	R3,H8 YES, SET IT FOR THE MAX - IGNORE RR	
000662	0630			673	LIMCMOVE	BCTR R3,0 MINUS ONE FOR THE EX	
000664	4430	C880	00880	674	EX	R3,MOVECCHH MOVE IN THE CHARS	
000668	DC07	B070	C931	00070	00931	675 TR DOUBLE,DECTABLE TRANSLATE HEX EBCDIC TO HEX BINARY	
00066E	F248	D068	B070	00068	00070	676 PACK CYLH(5),DOUBLE(9) SQUISH OUT THE ZONES	
000674	5840	D068	00068	677	L	R4,CYLH GET THE CCHH	
000678	5045	0000	00000	678	ST	R4,0(R5) SAVE IT FOR LATER	
00067C	4065	0004	00004	679	STH	R6,4(R5) SAVE THE LENGTH TOO	
000680				680	LIMCSET	DS 0H THE NUMBER IS CONVERTED	
				681 *			
				682 *		COMPARE THE EXTENTS TO THE CCHH VALUE.	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                ASM 0201 13.38 01/07/25
                                683 *
                                684 *
                                685 *
000680 9200 D0D0      000D0  686      MVI  CCHHCOMP,0    CLEAR THE FLAGS
000684 1B22          687      SR   R2,R2         CLEAR A REG FOR AN EXTENT COUNTER
000686 BF21 703B      0003B  688      ICM  R2,1,DS1NOEPV GET THE NUMBER OF EXTENTS
00068A 4780 C4BA      004BA  689      BZ   COMPNO        NO EXTENTS, JUST GO SEE
                                690 *
                                691 *
                                692 *
00068E 1B11          693      SR   R1,R1         FIRST EXTENT
000690 1831          694 EXTNEXT LR   R3,R1         GET THE CURRENT EXTENT
000692 8930 0002      00002  695      SLL  R3,2          TIMES 4
000696 4403 C700      00700  696      EX   R0,GETEXT(R3) GET THE EXTENT ADDRESS INTO R3
                                697 *
                                698 *
                                699 *
00069A 94F7 D0D0      000D0  700      NI   CCHHCOMP,255-CCHHX TURN OFF THE STRADDLE FLAG
00069E 4460 C886      00886  701      EX   R6,CLCEXTLO   DO THE COMPARE
0006A2 4720 C6B2      006B2  702      BH   SETH1         THE FIELD IS HIGHER THAN THE VALUE
0006A6 4780 C6BA      006BA  703      BE   SETEQ1        THE FIELD IS EQUAL TO THE VALUE
0006AA 9628 D0D0      000D0  704      OI   CCHHCOMP,CCHHLOW+CCHHX LOWER - POSSIBLE STRADDLE
0006AE 47F0 C6BE      006BE  705      B    CHECKHI        GO CHECK THE TOP OF THIS EXTENT
0006B2 9680 D0D0      000D0  706 SETH1 OI   CCHHCOMP,CCHHHIGH SET THE FLAG
0006B6 47F0 C6BE      006BE  707      B    CHECKHI        GO CHECK THE TOP OF THIS EXTENT
0006BA 9640 D0D0      000D0  708 SETEQ1 OI   CCHHCOMP,CCHHEQ  SET THE FLAG
                                709 *
                                710 *
                                711 *
0006BE 4460 C88C      0088C  712 CHECKHI EX   R6,CLCEXTHI   DO THE COMPARE
0006C2 4780 C6D6      006D6  713      BE   SETEQ2        EQUAL, GO SET IT
0006C6 4740 C6DA      006DA  714      BL   EXTSET        LOW, GO SET IT
                                715 *
                                716 *
                                717 *
                                718 *
                                719 *
                                720 *
0006CA 9680 D0D0      000D0  721      OI   CCHHCOMP,CCHHHIGH SET THE HIGH FLAG
0006CE 9108 D0D0      000D0  722      TM   CCHHCOMP,CCHHX WAS THE BOTTOM LOWER THAN THE VALUE?
0006D2 4780 C6DA      006DA  723      BZ   EXTSET        NO, SKIP ON
0006D6 9640 D0D0      000D0  724 SETEQ2 OI   CCHHCOMP,CCHHEQ  SET THE EQ FLAG
                                725 *
                                726 *
                                727 *
0006DA          728 EXTSET DS   0H
0006DA 4111 0001      00001  729      LA   R1,1(R1)      INCREMENT THE EXTENT COUNTER
0006DE 1912          730      CR   R1,R2         CHECK THE EXTENT COUNTER
0006E0 47B0 C6EC      006EC  731      BNL  LIMCOPER      THAT'S ALL FOLKS
0006E4 91E0 D0D0      000D0  732      TM   CCHHCOMP,CCHHHIGH+CCHHEQ+CCHHLOW ARE THEY ALL SET?
0006E8 47E0 C690      00690  733      BNO  EXTNEXT       NO, CONTINUE LOOKING
                                734 *
                                735 LIMCOPER L   R4,REFOPER      GET THE NUMERIC VALUE OF THE KEY
0006EC 5840 D098      00098  736      IC   R4,CCHHTAB(R4) GET A FLAG MASK
0006F0 4344 C896      00896  737      EX   R4,CCHHOPER   CHECK TO SEE IF THE CONDITION IS SET

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT		
0006F8	4780 C4BA	004BA		738	BZ	COMPNO	NOT THERE	
0006FC	47F0 C4B4	004B4		739	B	COMPYES	YES	
				740	*			
				741	*	EXECUTED INSTRUCTIONS TO GET THE ADDRESS OF THIS EXTENT		
				742	*			
000700	4130 7069	00069		743	GETEXT	LA R3,DS1EXT1	1ST EXTENT	
000704	4130 7073	00073		744		LA R3,DS1EXT2	2ND EXTENT	
000708	4130 707D	0007D		745		LA R3,DS1EXT3	3RD EXTENT	
00070C	4130 B774	00774		746		LA R3,DS3EXTNT	4TH EXTENT	
000710	4130 B77E	0077E		747		LA R3,DS3EXTNT+10	5TH EXTENT	
000714	4130 B788	00788		748		LA R3,DS3EXTNT+20	6TH EXTENT	
000718	4130 B792	00792		749		LA R3,DS3EXTNT+30	7TH EXTENT	
00071C	4130 B79D	0079D		750		LA R3,DS3ADEXT	8TH EXTENT	
000720	4130 B7A7	007A7		751		LA R3,DS3ADEXT+10	9TH EXTENT	
000724	4130 B7B1	007B1		752		LA R3,DS3ADEXT+20	10TH EXTENT	
000728	4130 B7BB	007BB		753		LA R3,DS3ADEXT+30	11TH EXTENT	
00072C	4130 B7C5	007C5		754		LA R3,DS3ADEXT+40	12TH EXTENT	
000730	4130 B7CF	007CF		755		LA R3,DS3ADEXT+50	13TH EXTENT	
000734	4130 B7D9	007D9		756		LA R3,DS3ADEXT+60	14TH EXTENT	
000738	4130 B7E3	007E3		757		LA R3,DS3ADEXT+70	15TH EXTENT	
00073C	4130 B7ED	007ED		758		LA R3,DS3ADEXT+80	16TH EXTENT	
				759	*			
				760	*	ISSUE ERROR MESSAGES AND RETURN		
				761	*			
				762	OBT3ERR	VTOCMSG OBT3ERRM	OBTAIN ERROR MESSAGE	
000740	4110 CA48	00A48		763	+OBT3ERR	LA R1,OBT3ERRM	POINT TO THE FIRST MESSAGE	
000744	1B00			764	+	SR R0,R0	NO SECOND LEVEL MESSAGE	
000746	9001 B078	00078		765	+	STM R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES	
				766	+	THEN JUST CALL THE MESSAGE ISSUING ROUTINE		
00074A	4110 B000	00000		767	+	LA R1,VTOCOM	POINT TO THE COMMON AREA	
00074E	58F0 B01C	0001C		768	+	L R15,VADMSG	POINT TO THE ROUTINE	
000752	05EF			769	+	BALR R14,R15	THEN CALL IT	
000754	41F0 0008	00008		770	CHECKOUT	LA R15,8	EXCLUDE THIS DATA SET	
000758	47F0 C75E	0075E		771		B CHEKRET	RETURN	
				772	*			
00075C	1BFF			773	CHECKIN	SR R15,R15	CLEAR THE REGISTER, PROCESS THIS DATA SET	
				774	CHEKRET	LEAVE EQ		
00075E	182D			775	+CHEKRET	LR 2,13		
000760	58DD 0004	00004		776	+	L 13,4(13)		
000764	90F1 D010	00010		777	+	STM 15,1,16(13)	STORE RETURN REGS	
000768	98EC D00C	0000C		778	+	LM 14,12,12(13)	RESTORE THE REGISTERS	00650000
00076C	92FF D00C	0000C		779	+	MVI 12(13),X'FF'	SET RETURN INDICATION	01600000
000770	07FE			780	+	BR 14	RETURN	02000000
		00000		781	+R0	EQU 0	*USED BY O.S.	
		00001		782	+R1	EQU 1	*USED BY O.S. // ADDRESS OF PARAMETER LIST	
		00002		783	+R2	EQU 2		
		00003		784	+R3	EQU 3		
		00004		785	+R4	EQU 4		
		00005		786	+R5	EQU 5		
		00006		787	+R6	EQU 6		
		00007		788	+R7	EQU 7		
		00008		789	+R8	EQU 8		
		00009		790	+R9	EQU 9		
		0000A		791	+R10	EQU 10		
		0000B		792	+R11	EQU 11		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
			0000C	793+R12	EQU	12	
			0000D	794+R13	EQU	13	*USED BY O.S. // SAVE-AREA ADDRESS
			0000E	795+R14	EQU	14	*USED BY O.S. // RETURN ADDRESS
			0000F	796+R15	EQU	15	*USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
				797 *			
				798 *			

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
-----	-------------	-------	-------	------	------------------

ASM 0201 13.38 01/07/25

				800 *	
				801 *	ROUTINES USED ABOVE
				802 *	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25
                                           804 *
                                           805 *      PDLNUM - CONVERT FROM CHARACTERS ( EBCDIC ) TO AN INTEGER
                                           806 *      BINARY FORM, PASSED BACK VIA REGISTER 15
                                           807 *      A PARSE PDE IS THE INPUT AS SHOWN IN THE SAMPLE BELOW
                                           808 *      LA      R1,PDL      POINT TO THE PARSE DECRPTION
                                           809 *      BAL     R8,PDLNUM   GO CONVERT TO NUMERICS
                                           810 *      THE ROUTINE WILL TERMINATE IF IT FINDS NON-NUMERICS
                                           811 *      ANY CHARACTERS OTHER THEN 0-9, +, -
                                           812 *      REGISTERS 1, 2, 5, 6, AND 7 ARE USED
                                           813 *
000772 9018 D0A4      000A4      814 PDLNUM  STM  R1,R8,PDLNSAVE  SAVE THE REGISTERS
000776 4821 0004      00004      815      LH    R2,4(R1)    GET THE STRING ADDRESS
00077A 5811 0000      00000      816      L    R1,0(R1)    GET THE STRING ADDRESS
00077E 9200 D0C4      000C4      817      MVI  PDLMINUS,0  CLEAR THE NEGATIVE NUMBER FLAG
000782 1B55                818      SR   R5,R5      CLEAR THE CHARACTER COUNTER
000784 1BFF                819      SR   R15,R15   CLEAR THE ANSWER
000786 4165 1000      00000      820 PDLLOOP LA  R6,0(R5,R1)  POINT TO THIS DIGIT
00078A 4155 0001      00001      821      LA  R5,1(R5)    GET TO THE NEXT DIGIT
00078E 1952                822      CR   R5,R2      IS THIS THE END OF THE STRING?
000790 4720 C7CC      007CC      823      BH  PDLFINI     YES, EXIT
000794 1B77                824      SR   R7,R7      CLEAR A WORK REGISTER
000796 4376 0000      00000      825      IC  R7,0(R6)    GET THE CHARACTER
00079A 4B70 C7DE      007DE      826      SH  R7,PDLH240  SUBTRACT THE CHARACTER C'0'
00079E 4740 C7AC      007AC      827      BM  PDLSP       IF NEGATIVE, CHECK SPECIAL CHARACTERS
0007A2 4CF0 C7DC      007DC      828      MH  R15,PDLH10  IT'S A DIGIT, MULTIPLY PRIOR NUM BY TEN
0007A6 1AF7                829      AR  R15,R7      ADD ON THE NEW DIGIT
0007A8 47F0 C786      00786      830      B   PDLLOOP     AND LOOP FOR MORE
                                           831 *
                                           832 *      CHECK FOR SPECIAL CHARACTERS
                                           833 *
0007AC 9540 6000      00000      834 PDLSP  CLI  0(R6),C' '   IS IT A BLANK?
0007B0 4780 C786      00786      835      BE  PDLLOOP     THEN IT'S OK
0007B4 954E 6000      00000      836      CLI  0(R6),C'+'  IS IT A PLUS?
0007B8 4780 C786      00786      837      BE  PDLLOOP     THAT'S ALSO OK
0007BC 9560 6000      00000      838      CLI  0(R6),C'-'  IS IT A MINUS?
0007C0 4770 C7CC      007CC      839      BNE PDLFINI     NO, JUST QUIT
0007C4 9201 D0C4      000C4      840      MVI PDLMINUS,1  YES, NOTE IT
0007C8 47F0 C786      00786      841      B   PDLLOOP     AND LOOK FOR MORE
                                           842 *
                                           843 *      QUIT, AFTER SETTING R15 TO NEGATIVE IF NEEDED
                                           844 *
0007CC 9501 D0C4      000C4      845 PDLFINI CLI  PDLMINUS,1  WAS A MINUS SIGN FOUND?
0007D0 4770 C7D6      007D6      846      BNE PDLLEAVE    NO, EXIT
0007D4 11FF                847      LNR R15,R15     YES, MAKE IT NEGATIVE
0007D6 9818 D0A4      000A4      848 PDLLEAVE LM  R1,R8,PDLNSAVE  RESTORE THE REGISTERS
0007DA 07F8                849      BR  R8          RETURN
0007DC 000A                850 PDLH10  DC  H'10'
0007DE 00F0                851 PDLH240 DC  H'240'

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25	
				853	*		
				854	*	ROUTINE TO CONVERT A TEXT DSCB ITEM	
				855	*	INTO ITS KEY NUMBER	
				856	*	INPUT IS REG 4 - IKJIDENT PTR	
				857	*	OUTPUT IS REG 15 - KEY NUMBER	
				858	*	ENTRY VIA BAL R8,GETKEY	
				859	*		
0007E0	5810	B1AC	001AC	860	GETKEY L	R1,ATABTITL POINT TO THE TABLE	
0007E4	4111	000C	0000C	861	LA	R1,12(R1) POINT TO THE FIRST ENTRY	
0007E8	41F0	0001	00001	862	LA	R15,1 SET UP THE KEY NUMBER COUNTER	
0007EC	5864	0000	00000	863	L	R6,0(R4) POINT TO THE ENTERED TEXT	
0007F0	BF33	4004	00004	864	ICM	R3,3,4(R4) GET THE LENGTH OF THE ENTERED TEXT	
0007F4	47D0	C82E	0082E	865	BNP	GETKNOTF NOT FOUND IF ZERO	
0007F8	0630			866	BCTR	R3,0 MINUS ONE FOR THE EX	
0007FA	4121	0004	00004	867	GETKLOOP LA	R2,4(R1) POINT TO THE COMPARISON TEXT	
0007FE	9540	2000	00000	868	CLI	0(R2),C' ' IS IT HERE?	
000802	4770	C816	00816	869	BNE	GETKSTD YES, THIS IS IT	
000806	4122	0001	00001	870	LA	R2,1(R2) NO, MOVE OVER ONE MORE	
00080A	9540	2000	00000	871	CLI	0(R2),C' ' IS IT HERE?	
00080E	4770	C816	00816	872	BNE	GETKSTD YES, THIS IS IT	
000812	4122	0001	00001	873	LA	R2,1(R2) NO, MOVE OVER ONE MORE	
000816	4430	C832	00832	874	GETKSTD EX	R3,GETKCOMP COMPARE THE KEY TEXT	
00081A	4780	C830	00830	875	BE	GETKFND I FOUND IT	
00081E	4111	000C	0000C	876	LA	R1,12(R1) GET TO THE NEXT KEY	
000822	41FF	0001	00001	877	LA	R15,1(R15) INCREMENT THE KEY COUNTER	
000826	49F0	C838	00838	878	CH	R15,H26 CHECK FOR THE END OF THE TABLE	
00082A	47D0	C7FA	007FA	879	BNH	GETKLOOP NOT YET, KEEP LOOKING	
				880	*		
				881	*	KEY WAS NOT FOUND, SEND BACK A ZERO	
				882	*		
00082E	1BFF			883	GETKNOTF SR	R15,R15 SET UP THE ZERO AND RETURN	
000830	07F8			884	GETKFND BR	R8 JUST RETURN	
000832	D500	6000	2000	00000	00000	885 GETKCOMP CLC	0(0,R6),0(R2) EXECUTED TEXT COMPARE
000838	001A			886	H26	DC H'26'	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				888 *		
				889 *	ROUTINE TO CONVERT THE OPERATOR TEXT	
				890 *	INTO A NUMERIC VALUE	
				891 *		
00083A	41F0 0001	00001		892	GETOPER LA R15,1 NUMERIC VALUE COUNTER	
00083E	18EF			893	GETOLOOP LR R14,R15 GET THE NUMBER	
000840	8BE0 0001	00001		894	SLA R14,1 MULTIPLY BY 2	
000844	41EE C89D	0089D		895	LA R14,OPERS(R14) RELOCATE IT	
000848	D501 E000 1000	00000 00000		896	CLC 0(2,R14),0(R1) IS THIS THE TEXT?	
00084E	4780 C878	00878		897	BE GETOFND YES, RETURN THE NUMBER	
000852	41FF 0001	00001		898	LA R15,1(R15) NO, TRY THE NEXT ONE	
000856	49F0 CA30	00A30		899	CH R15,H7 CHECK FOR THE END	
00085A	4740 C83E	0083E		900	BL GETOLOOP NOT THERE YET, KEEP TRYING	
				901	VTOCMSG OPERERR,OPERERR2 ISSUE THE MESSAGE	
00085E	4110 CAE8	00AE8		902+	LA R1,OPERERR POINT TO THE FIRST MESSAGE	
000862	4100 CB22	00B22		903+	LA R0,OPERERR2 POINT TO THE SECOND MESSAGE	
000866	9001 B078	00078		904+	STM R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				905+*	THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
00086A	4110 B000	00000		906+	LA R1,VTOCOM POINT TO THE COMMON AREA	
00086E	58F0 B01C	0001C		907+	L R15,VADMSG POINT TO THE ROUTINE	
000872	05EF			908+	BALR R14,R15 THEN CALL IT	
000874	41F0 0001	00001		909	LA R15,1 SET THE DEFAULT OPERATOR, EQ	
000878	07F8			910	GETOFND BR R8 THEN RETURN	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
-----	-------------	-------	-------	------	------------------

ASM 0201 13.38 01/07/25

970 *

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				972	*	
				973	*	
				974	*	P A R S E C O N T R O L L I S T
				975	*	
				976	*	
				977		PRINT OFF
				979		PUSH PRINT
				980		PRINT NOGEN
				2148		POP PRINT
				2175		PRINT ON
				2176	*	
				2177	*	DYNAMIC WORK AREA
				2178	*	
000000				2180	CHEKWORK DSECT	
000000				2181	DS 18A	PRINT ROUTINE SAVE AREA
000048				2182	CHARS DS CL16	CONVERSION TO CHARACTERS
				2183	CAMSEEK CAMLST SEEK,*,*,*	
		00010		2192	CAMLEN EQU *-CAMSEEK	
000068				2193	DS 0D	
000068				2194	CYLH DS F	
00006C				2195	DS X	PAD FOR CCHH
		00001		2196	EQ EQU 1	EQUATES FOR OPERATOR VALUES
		00002		2197	NE EQU 2	
		00003		2198	LT EQU 3	
		00004		2199	LE EQU 4	
		00005		2200	GT EQU 5	
		00006		2201	GE EQU 6	
00006E				2202	HWORK DS H	
000070				2203	LIMVAL DS F	
000074				2204	NUMBERL DS F	
000078				2205	NUMLENL DS H	
00007A				2206	FLAGNML DS X	
00007B				2207	NUMKEYL DS X	
00007C				2208	NUMBER1 DS F	
000080				2209	NUMLEN1 DS H	
000082				2210	FLAGNM1 DS X	
000083				2211	NUMKEY1 DS X	
000084				2212	NUMBER2 DS F	
000088				2213	NUMLEN2 DS H	
00008A				2214	FLAGNM2 DS X	
00008B				2215	NUMKEY2 DS X	
00008C				2216	NUMBER3 DS F	
000090				2217	NUMLEN3 DS H	
000092				2218	FLAGNM3 DS X	
000093				2219	NUMKEY3 DS X	
000094				2220	REFKEY DS F	
000098				2221	REFOPER DS F	
00009C				2222	REFVAL DS A	
0000A0				2223	REFNUM DS A	
0000A4				2224	PDLNSAVE DS 8A	REGISTER SAVE AREA FOR PDLNUM RTN
0000C4	00			2225	PDLMINUS DC X'00'	
0000C8				2226	CHEKDBLW DS D	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
0000D0				2227	CCHHCOMP	DS X
			00080	2228	CCHHHIGH	EQU X'80'
			00040	2229	CCHHEQ	EQU X'40'
			00020	2230	CCHHLOW	EQU X'20'
			00008	2231	CCHHX	EQU X'08'
0000D8				2232		DS 0D
			000D8	2233	LENWORK	EQU *-CHEKWORK
				2234	*	
				2235	*	VTOC COMMAND COMMON AREA
				2236	*	
				2237		PRINT NOGEN
				2238		VTOCOM
				2406	*	
				2407	*	FORMATTED DSCB
				2408	*	
				2409		VTFMT
				2451		PDEDSNAM
000000				2474	DSCB1	DSECT
				2475		IECSDSL1 1
				2557		END
000B40	00011940			2558		=A(72*1000)
000B44	000186A0			2559		=F'100000'
000B48	0000000000			2560		=XL5'0000000000'

POS.ID	REL.ID	FLAGS	ADDRESS	ASM 0201 13.38 01/07/25
0001	0001	0C	0009E0	
0001	0001	0C	0009E4	
0001	0001	0C	0009E8	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
ADDRANSR	00004	00000048	02271	00111
AL00601	00001	00000C81	01210	01209
AL00611	00001	00000C8F	01224	01223
ANDOR1K	00002	00000034	01198	00275 00312 01195
ANDOR2K	00002	00000036	01237	00319 00356 01234
ANDOR3K	00002	00000038	01268	00363 00400 01265
ATABTITL	00004	000001AC	02304	00417 00860
BLKSZ	00001	0000000C	02160	00562
BLKSZSET	00004	000000BC	01638	01627
BLKTRTAB	00064	000008DB	00926	00132
BREAK	00004	000000AC	01591	01578
BREAKK	00002	00000030	01156	01153
CAMSCON	00004	000009DC	00928	00145
CAMSEEK	00004	00000058	02184	00145 00147 00149 00151 00153 02192
CATK	00002	0000002C	01106	01103
CCHHCOMP	00001	000000D0	02227	00686 00700 00704 00706 00708 00721 00722 00724 00732 00919
CCHHEQ	00001	00000040	02229	00708 00724 00732
CCHHHIGH	00001	00000080	02228	00706 00721 00732
CCHHLEN	00002	00000AB2	00965	00650 00663
CCHHLOW	00001	00000020	02230	00704 00732
CCHHOPER	00004	00000892	00919	00737
CCHHTAB	00007	00000896	00920	00736
CCHHX	00001	00000008	02231	00700 00704 00722
CDATE	00001	00000003	02151	00516
CHARSK	00002	0000003C	01329	01326
CHARSPL	00004	000000B4	01618	01607
CHECKHI	00004	000006BE	00712	00705 00707
CHECKIN	00002	0000075C	00773	00410
CHECKOUT	00004	00000754	00770	00127 00167 00176 00188 00200 00212 00228 00408
CHEKDBLW	00008	000000C8	02226	00543 00544
CHEKFMTI	00002	0000003E	00122	00121
CHEKRET	00002	0000075E	00775	00771
CHEKWORK	00001	00000000	02180	00113 02233
CLCEXTHI	00006	0000088C	00918	00712
CLCEXTLO	00006	00000886	00917	00701
COMPARE	00006	000009EC	00936	00172 00196 00221
COMPDONE	00002	00000458	00474	00535 00548 00555 00566 00571 00575 00587 00601 00624
COMPEQ	00004	0000047C	00486	00476
COMPLIM	00006	0000087A	00915	00473
COMFLOW	00004	00000498	00494	00475
COMPNO	00002	000004BA	00506	00484 00492 00500 00689 00738
COMPYES	00004	000004B4	00504	00479 00481 00483 00487 00489 00491 00495 00497 00499 00616 00739
CONCOMP	00004	00000130	00221	00225
CONEND	00002	0000014E	00229	00207 00222
CONINC	00004	00000142	00226	00215
CONNEXT	00002	00000112	00210	00227
CONTAIN	00004	00000084	01534	00208 01528
CONTAINK	00002	00000028	01062	00206 01059
CYLH	00004	00000068	02194	00676 00677
DECTABLE	00001	00000931	00937	00675
DEND0093	00001	00000E31	01560	01550
DEND0095	00001	00000E64	01589	01577
DEND0097	00001	00000E95	01616	01606
DEND0098	00001	00000EB6	01636	01626
DEND0100	00001	00000EE7	01663	01653

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
DEND0102	00001	00000F0D	01690	01680	
DEND0103	00001	00000F29	01710	01700	
DEND0107	00001	00000F65	01757	01745	
DEND0109	00001	00000FA0	01784	01774	
DEND0110	00001	00000FEF	01807	01795	
DEND0111	00001	00001033	01829	01818	
DEND0113	00001	0000106E	01856	01846	
DEND0114	00001	000010BD	01879	01867	
DEND0115	00001	00001101	01901	01890	
DEND0117	00001	0000113C	01928	01918	
DEND0118	00001	0000118B	01951	01939	
DEND0119	00001	000011CF	01973	01962	
DEND0121	00001	0000120A	02000	01990	
DEND0122	00001	00001259	02023	02011	
DEND0123	00001	0000129D	02045	02034	
DEND0125	00001	000012C6	02074	02062	
DEND0127	00001	000012F3	02103	02091	
DENT0093	00001	00000E18	01548	01550	
DENT0095	00001	00000E34	01575	01577	
DENT0097	00001	00000E67	01604	01606	
DENT0098	00001	00000E95	01624	01626	
DENT0100	00001	00000EB9	01651	01653	
DENT0102	00001	00000EEA	01678	01680	
DENT0103	00001	00000F0D	01698	01700	
DENT0107	00001	00000F36	01743	01745	
DENT0109	00001	00000F68	01772	01774	
DENT0110	00001	00000FA0	01793	01795	
DENT0111	00001	00000FEF	01816	01818	
DENT0113	00001	00001036	01844	01846	
DENT0114	00001	0000106E	01865	01867	
DENT0115	00001	000010BD	01888	01890	
DENT0117	00001	00001104	01916	01918	
DENT0118	00001	0000113C	01937	01939	
DENT0119	00001	0000118B	01960	01962	
DENT0121	00001	000011D2	01988	01990	
DENT0122	00001	0000120A	02009	02011	
DENT0123	00001	00001259	02032	02034	
DENT0125	00001	000012A0	02060	02062	
DENT0127	00001	000012C9	02089	02091	
DOUBLE	00008	00000070	02275	00645 00645 00675 00676 00916	
DSCBADDR	00004	00000194	02288	00117	
DSCB1	00001	00000000	02474	00119	
DSNLEN	00002	000001AA	02303	00134	
DSNLNTYP	00002	00000164	02118	02113	
DSNPLN	00004	0000015C	02105	02092	
DSNPLNK	00002	00000048	01463	01460	
DS1BLKL	00002	00000056	02518	00564	
DS1CREDT	00003	00000035	02482	00515	
DS1DSNAM	00044	00000000	02478	00132	
DS1EXPDT	00003	00000038	02483	00518	
DS1EXT1	00010	00000069	02548	00743	
DS1EXT2	00010	00000073	02553	00744	
DS1EXT3	00010	0000007D	02554	00745	
DS1FMTID	00001	0000002C	02479	00126 00131	
DS1IND01	00001	00000001	02542	02543	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
DS1IND02	00001	00000002	02538	02541
DS1LRECL	00002	00000058	02519	00569
DS1NOEPV	00001	0000003B	02484	00554 00688
DS1PTRDS	00005	00000087	02555	00140 00146
DS1REFD	00003	0000004B	02489	00521
DS1SCALO	00004	0000005E	02544	00572
DS3ADEXT	00090	0000079D	02401	00750 00751 00752 00753 00754 00755 00756 00757 00758
DS3EXTNT	00040	00000774	02395	00746 00747 00748 00749
ENDEND	00002	00000106	00201	00183 00197
ENDINC	00004	000000FA	00198	00191
ENDING	00004	00000068	01513	00184 01507
ENDKEY	00002	00000026	01041	00182 01038
ENDNEXT	00002	000000D6	00186	00199
EQ	00001	00000001	02196	00486
EXPDT	00001	00000005	02153	00519
EXTNEXT	00002	00000690	00694	00733
EXTSET	00002	000006DA	00728	00714 00723
FLAGNML	00001	0000007A	02206	00243 00245
FLAGNM1	00001	00000082	02210	00282 00284
FLAGNM2	00001	0000008A	02214	00326 00328
FLAGNM3	00001	00000092	02218	00370 00372
FMT3	00148	00000770	02391	00139 00139 00150
FMT3NO	00004	00000092	00160	00141
FORMATAD	00004	00000198	02289	00455 00465 00592 00606
FORMATK	00002	00000046	01442	00120 01439
FORMATSP	00004	00000150	02076	02063
F100	00004	00000A3C	00951	00619
GE	00001	00000006	02201	00482 00490
GETEXT	00004	00000700	00743	00696
GETKCOMP	00006	00000832	00885	00874
GETKEY	00004	000007E0	00860	00249 00288 00332 00376
GETKFND	00002	00000830	00884	00875
GETKLOOP	00004	000007FA	00867	00879
GETKNOTF	00002	0000082E	00883	00865
GETKSTD	00004	00000816	00874	00869 00872
GETOFND	00002	00000878	00910	00897
GETOLOOP	00002	0000083E	00893	00900
GETOPER	00004	0000083A	00892	00239 00278 00322 00366
GT	00001	00000005	02200	00480
HEADING	00004	000000E0	01730	01726
HEADK	00002	00000040	01371	01368
HWORk	00002	0000006E	02202	00572 00573
H1000	00002	00000A40	00952	00524
H12	00002	00000A36	00949	00416
H26	00002	00000838	00886	00878
H4	00002	00000A2E	00945	00646
H7	00002	00000A30	00946	00899
H8	00002	00000A32	00947	00660 00670 00672
IECSDSL1	00001	00000000	02476	02477
IECSDSL3	00001	00000770	02392	02393
IECSDSL4	00001	0000070C	02351	02352
IKJ\$0039	00001	0000131E	02145	00983
IKJ\$0041	00001	00000BCD	01004	01000
IKJ\$0095	00001	00000E64	01588	01587
IKJ\$0107	00001	00000F65	01756	01755

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
IKJ\$0110	00001	00000FEF	01806	01805	
IKJ\$0111	00001	00001033	01828	01827	
IKJ\$0114	00001	000010BD	01878	01877	
IKJ\$0115	00001	00001101	01900	01899	
IKJ\$0118	00001	0000118B	01950	01949	
IKJ\$0119	00001	000011CF	01972	01971	
IKJ\$0122	00001	00001259	02022	02021	
IKJ\$0123	00001	0000129D	02044	02043	
IKJ\$0125	00001	000012C6	02073	02072	
IKJ\$0127	00001	000012F3	02102	02101	
IKJ@0039	00001	00000168	02143	00984	
IKJ@0040	00001	0000007D	01013	00985	
IKJ@0041	00001	00000B84	00999	00998	
IKJ@0043	00001	00000BDD	01028	01027	
IKJ@0045	00001	00000BF0	01049	01048	
IKJ@0047	00001	00000C07	01070	01069	
IKJ@0048	00001	00000C14	01083	01082	
IKJ@0049	00001	00000C1D	01094	01093	
IKJ@0051	00001	00000C2B	01114	01113	
IKJ@0053	00001	00000C3A	01134	01133	
IKJ@0054	00001	00000C47	01144	01143	
IKJ@0056	00001	00000C57	01164	01163	
IKJ@0058	00001	00000C69	01185	01184	
IKJ@0060	00001	00000C7A	01206	01205	
IKJ@0061	00001	00000C89	01220	01219	
IKJ@0063	00001	00000C9E	01245	01244	
IKJ@0064	00001	00000CA8	01255	01254	
IKJ@0066	00001	00000CB9	01276	01275	
IKJ@0067	00001	00000CC3	01286	01285	
IKJ@0069	00001	00000CD5	01307	01306	
IKJ@0070	00001	00000CE3	01317	01316	
IKJ@0072	00001	00000CF3	01337	01336	
IKJ@0074	00001	00000D05	01358	01357	
IKJ@0076	00001	00000D19	01379	01378	
IKJ@0077	00001	00000D29	01389	01388	
IKJ@0079	00001	00000D3A	01409	01408	
IKJ@0081	00001	00000D4D	01430	01429	
IKJ@0083	00001	00000D5E	01450	01449	
IKJ@0085	00001	00000D71	01471	01470	
IKJ@0086	00001	00000258	01498	01479	
IKJ@0087	00001	00000DA8	01489	01488	
IKJ@0088	00001	0000028A	01519	01500	
IKJ@0089	00001	00000DDA	01510	01509	
IKJ@0090	00001	000002C5	01540	01521	
IKJ@0091	00001	00000E15	01531	01530	
IKJ@0092	00001	000002E1	01567	01542	
IKJ@0093	00001	00000E30	01558	01557	
IKJ@0094	00001	00000314	01596	01569	
IKJ@0095	00001	00000E61	01585	01584	
IKJ@0096	00001	00000366	01643	01598	
IKJ@0097	00001	00000E94	01614	01613	
IKJ@0098	00001	00000EB5	01634	01633	
IKJ@0099	00001	00000397	01670	01645	
IKJ@0100	00001	00000EE6	01661	01660	
IKJ@0101	00001	000003D9	01717	01672	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
IKJ@0102	00001	00000F0C	01688	01687	
IKJ@0103	00001	00000F28	01708	01707	
IKJ@0104	00001	000003E3	01735	01719	
IKJ@0106	00001	00000415	01764	01737	
IKJ@0107	00001	00000F62	01753	01752	
IKJ@0108	00001	000004E3	01836	01766	
IKJ@0109	00001	00000F9F	01782	01781	
IKJ@0110	00001	00000FC4	01803	01802	
IKJ@0111	00001	0000100C	01826	01825	
IKJ@0112	00001	000005B1	01908	01838	
IKJ@0113	00001	0000106D	01854	01853	
IKJ@0114	00001	00001092	01875	01874	
IKJ@0115	00001	000010DA	01898	01897	
IKJ@0116	00001	0000067F	01980	01910	
IKJ@0117	00001	0000113B	01926	01925	
IKJ@0118	00001	00001160	01947	01946	
IKJ@0119	00001	000011A8	01970	01969	
IKJ@0120	00001	0000074D	02052	01982	
IKJ@0121	00001	00001209	01998	01997	
IKJ@0122	00001	0000122E	02019	02018	
IKJ@0123	00001	00001276	02042	02041	
IKJ@0124	00001	00000776	02081	02054	
IKJ@0125	00001	000012C3	02070	02069	
IKJ@0126	00001	000007A3	02109	02083	
IKJ@0127	00001	000012EF	02099	02098	
IKJ@0128	00001	00001302	02115	02114	
IKJ@0129	00001	0000130F	02126	02125	
IKJ@0130	00001	0000131D	02135	02134	
IKJ00411	00001	00000BCD	01003	01002	
KEND0042	00001	00000BD3	01018	01016	
KEND0044	00001	00000BE5	01039	01037	
KEND0046	00001	00000BF8	01060	01058	
KEND0048	00001	00000C14	01084	01080	
KEND0050	00001	00000C23	01104	01102	
KEND0052	00001	00000C31	01124	01122	
KEND0055	00001	00000C4D	01154	01152	
KEND0057	00001	00000C5F	01175	01173	
KEND0059	00001	00000C71	01196	01194	
KEND0062	00001	00000C95	01235	01233	
KEND0065	00001	00000CB0	01266	01264	
KEND0068	00001	00000CCB	01297	01295	
KEND0071	00001	00000CE9	01327	01325	
KEND0073	00001	00000CFB	01348	01346	
KEND0075	00001	00000D0D	01369	01367	
KEND0078	00001	00000D2F	01399	01397	
KEND0080	00001	00000D42	01420	01418	
KEND0082	00001	00000D53	01440	01438	
KEND0084	00001	00000D66	01461	01459	
KEND0128	00001	00001302	02116	02112	
KEYERR	00002	00000A7A	00963	00254 00293 00337 00381	
KEYW0042	00001	00000BCD	01014	01016	
KEYW0044	00001	00000BDF	01035	01037	
KEYW0046	00001	00000BF2	01056	01058	
KEYW0048	00001	00000C09	01078	01080	
KEYW0050	00001	00000C1D	01100	01102	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
KEYW0052	00001	00000C2B	01120	01122	
KEYW0055	00001	00000C47	01150	01152	
KEYW0057	00001	00000C59	01171	01173	
KEYW0059	00001	00000C6B	01192	01194	
KEYW0062	00001	00000C8F	01231	01233	
KEYW0065	00001	00000CAA	01262	01264	
KEYW0068	00001	00000CC5	01293	01295	
KEYW0071	00001	00000CE3	01323	01325	
KEYW0073	00001	00000CF5	01344	01346	
KEYW0075	00001	00000D07	01365	01367	
KEYW0078	00001	00000D29	01395	01397	
KEYW0080	00001	00000D3C	01416	01418	
KEYW0082	00001	00000D4D	01436	01438	
KEYW0084	00001	00000D60	01457	01459	
KEYW0128	00001	000012F3	02110	02112	
LE	00001	00000004	02199	00488	00498
LEVEL	00004	0000004C	01492	00162	01486
LEVEND	00002	000000CA	00177	00161	00173
LEVINC	00004	000000BE	00174	00170	
LEVKEY	00002	00000024	01020	00160	01017
LEVNEXT	00002	000000A0	00165	00175	
LIMABEND	00002	000003D2	00427	00458	00459
LIMALLOC	00002	00000574	00579	00460	
LIMBLREC	00002	00000538	00556	00440	
LIMB1	00004	00000552	00567	00563	
LIMB2	00006	00000564	00572	00568	
LIMCCHH	00002	000005EC	00631	00441	
LIMCCON1	00004	0000062E	00657	00648	
LIMCCON2	00004	00000636	00659	00647	
LIMCMOVE	00002	00000662	00673	00658	00661 00671
LIMCOMP	00004	0000039E	00406	00276	00320 00364
LIMCOPER	00004	000006EC	00735	00731	
LIMCSET	00002	00000680	00680	00639	
LIMDATE	00002	000004BE	00511	00434	
LIMDGET	00002	000004DA	00522	00517	00520
LIMEND	00002	000003A8	00409	00235	
LIMEVAL	00004	000003AC	00415	00271	00310 00354 00398
LIMEVAL1	00004	000003E2	00433	00425	
LIMEXT	00002	0000052A	00549	00439	
LIMFCALD	00002	0000041C	00454	00448	
LIMFORM	00002	00000406	00446	00420	00436 00437 00438
LIMFORMA	00002	0000043E	00463	00457	
LIMITK	00002	00000032	01177	00234	01174
LIMLUCMP	00004	0000051C	00545	00542	
LIMLUSE	00002	00000506	00536	00435	
LIMPCOMP	00004	000005DE	00621	00614	00618
LIMPCT	00002	000005A8	00602	00462	
LIMPCTOK	00004	000005D6	00619	00612	
LIMP100	00004	000005CE	00617	00610	
LIMUNUAL	00004	0000059A	00598	00595	
LIMUNUSD	00002	00000586	00588	00461	
LIMVAL	00004	00000070	02203	00273	00314 00316 00317 00358 00360 00361 00402 00404 00405 00406
LINESK	00002	0000003E	01350	01347	
LINESPP	00004	000000C4	01665	01654	
LRECL	00001	0000000D	02161	00567	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
LT	00001	00000003	02198	00496
MOVECCHH	00006	00000880	00916	00674
MSGADDRS	00004	00000078	02277	00259 00298 00342 00386 00652 00665 00765 00904
MSGTEXT2	00124	000000FC	02279	00254 00256 00257 00293 00295 00296 00337 00339 00340 00381 00383 00384
NAME0043	00001	00000BD3	01024	01026
NAME0045	00001	00000BE5	01045	01047
NAME0047	00001	00000BF8	01066	01068
NAME0049	00001	00000C14	01090	01092
NAME0051	00001	00000C23	01110	01112
NAME0053	00001	00000C31	01130	01132
NAME0054	00001	00000C3C	01140	01142
NAME0056	00001	00000C4D	01160	01162
NAME0058	00001	00000C5F	01181	01183
NAME0060	00001	00000C71	01202	01204
NAME0061	00001	00000C81	01216	01218
NAME0063	00001	00000C95	01241	01243
NAME0064	00001	00000CA0	01251	01253
NAME0066	00001	00000CB0	01272	01274
NAME0067	00001	00000CBB	01282	01284
NAME0069	00001	00000CCB	01303	01305
NAME0070	00001	00000CD7	01313	01315
NAME0072	00001	00000CE9	01333	01335
NAME0074	00001	00000CFB	01354	01356
NAME0076	00001	00000D0D	01375	01377
NAME0077	00001	00000D1B	01385	01387
NAME0079	00001	00000D2F	01405	01407
NAME0081	00001	00000D42	01426	01428
NAME0083	00001	00000D53	01446	01448
NAME0085	00001	00000D66	01467	01469
NAME0129	00001	00001302	02122	02124
NAME0130	00001	0000130F	02131	02133
NE	00001	00000002	02197	00478 00494
NEND0043	00001	00000BDF	01030	01026
NEND0045	00001	00000BF2	01051	01047
NEND0047	00001	00000C09	01072	01068
NEND0049	00001	00000C1D	01095	01092
NEND0051	00001	00000C2B	01115	01112
NEND0053	00001	00000C3C	01136	01132
NEND0054	00001	00000C47	01145	01142
NEND0056	00001	00000C59	01166	01162
NEND0058	00001	00000C6B	01187	01183
NEND0060	00001	00000C81	01212	01204
NEND0061	00001	00000C8F	01226	01218
NEND0063	00001	00000CA0	01247	01243
NEND0064	00001	00000CAA	01257	01253
NEND0066	00001	00000CBB	01278	01274
NEND0067	00001	00000CC5	01288	01284
NEND0069	00001	00000CD7	01309	01305
NEND0070	00001	00000CE3	01318	01315
NEND0072	00001	00000CF5	01339	01335
NEND0074	00001	00000D07	01360	01356
NEND0076	00001	00000D1B	01381	01377
NEND0077	00001	00000D29	01390	01387
NEND0079	00001	00000D3C	01411	01407
NEND0081	00001	00000D4D	01431	01428

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
NEND0083	00001	00000D60	01452	01448
NEND0085	00001	00000D73	01473	01469
NEND0129	00001	0000130F	02127	02124
NEND0130	00001	0000131D	02136	02133
NUMBERL	00004	00000074	02204	00247 00269
NUMBER1	00004	0000007C	02208	00286 00308
NUMBER2	00004	00000084	02212	00330 00352
NUMBER3	00004	0000008C	02216	00374 00396
NUMKEYL	00001	0000007B	02207	00250 00266
NUMKEY1	00001	00000083	02211	00289 00305
NUMKEY2	00001	0000008B	02215	00333 00349
NUMKEY3	00001	00000093	02219	00377 00393
OBT3ERR	00004	00000740	00763	00156
OBT3ERRM	00002	00000A48	00961	00763
OPERERR	00002	00000AE8	00967	00902
OPERERR2	00002	00000B22	00969	00903
OPERS	00014	0000089D	00921	00895
OUTPUTK	00002	00000044	01422	01419
PCLMAIN	00001	00000B50	00982	00983 00986 00992 01012 01013 01023 01029 01034 01044 01050 01055 01065 01071 01077 01089 01099 01109 01119 01129 01135 01139 01149 01159 01165 01170 01180 01186 01191 01201 01207 01215 01221 01230 01240 01246 01250 01256 01261 01271 01277 01281 01287 01292 01302 01308 01312 01322 01332 01338 01343 01353 01359 01364 01374 01380 01384 01394 01404 01410 01415 01425 01435 01445 01451 01456 01466 01472 01477 01482 01497 01498 01503 01518 01519 01524 01539 01540 01545 01566 01567 01572 01595 01596 01601 01621 01642 01643 01648 01669 01670 01675 01695 01716 01717 01722 01734 01735 01740 01763 01764 01769 01790 01813 01835 01836 01841 01862 01885 01907 01908 01913 01934 01957 01979 01980 01985 02006 02029 02051 02052 02057 02080 02081 02086 02108 02109 02121 02130 02139 02144
PDEDCHN	00003	00000019	02471	00174 00198 00226
PDEDSN	00004	00000000	02456	00168 00189 00213
PDEDSNAM	00001	00000000	02455	00163 00185 00209
PDEDSNL	00002	00000004	02457	00166 00187 00211
PDL	00001	00000000	00987	00112 00996 01006 01017 01019 01038 01040 01059 01061 01081 01085 01103 01105 01123 01125 01153 01155 01174 01176 01195 01197 01234 01236 01265 01267 01296 01298 01326 01328 01347 01349 01368 01370 01398 01400 01419 01421 01439 01441 01460 01462 01486 01491 01507 01512 01528 01533 01551 01561 01578 01590 01607 01617 01627 01637 01654 01664 01681 01691 01701 01711 01726 01729 01746 01758 01775 01785 01796 01808 01819 01830 01847 01857 01868 01880 01891 01902 01919 01929 01940 01952 01963 01974 01991 02001 02012 02024 02035 02046 02063 02075 02092 02104 02113 02117 02141 02143
PDLFINI	00004	000007CC	00845	00823 00839
PDLH10	00002	000007DC	00850	00828
PDLH240	00002	000007DE	00851	00826
PDLLEAVE	00004	000007D6	00848	00846
PDLLOOP	00004	00000786	00820	00830 00835 00837 00841
PDLMINUS	00001	000000C4	02225	00817 00840 00845
PDLNSAVE	00004	000000A4	02224	00814 00848
PDLNUM	00004	00000772	00814	00246 00285 00329 00373
PDLSP	00004	000007AC	00834	00827
PEND0041	00001	00000BCD	01005	00995
PEND0087	00001	00000DA8	01490	01485
PEND0089	00001	00000DDA	01511	01506
PEND0091	00001	00000E15	01532	01527
PEND0105	00001	00000F33	01728	01725
POST0041	00001	00000B56	00993	00995
POST0087	00001	00000D76	01483	01485
POST0089	00001	00000DAB	01504	01506

SYMBOL	LEN	VALUE	DEFN	REFERENCES
POST0091	00001	00000DDD	01525	01527
POST0105	00001	00000F2C	01723	01725
PRINTK	00002	0000003A	01299	01296
REFKEY	00004	00000094	02220	00268 00307 00351 00395 00415 00516 00519 00562 00567
REFNUM	00004	000000A0	02223	00270 00309 00353 00397 00528 00545 00553 00560 00583 00598 00621 00636
REFOPER	00004	00000098	02221	00240 00279 00323 00367 00478 00480 00482 00486 00488 00490 00494 00496 00498 00735
REFVAL	00004	0000009C	02222	00242 00281 00325 00369 00466 00641
R0	00001	00000000	00781	00258 00258 00259 00297 00297 00298 00341 00341 00342 00385 00385 00386 00651 00651 00652 00664 00664 00665 00696 00764 00764 00765 00903 00904
R1	00001	00000001	00782	00109 00131 00133 00134 00135 00146 00147 00148 00149 00150 00151 00216 00217 00218 00224 00238 00241 00242 00255 00256 00257 00259 00261 00265 00265 00266 00268 00269 00270 00277 00280 00281 00294 00295 00296 00298 00300 00304 00304 00305 00307 00308 00309 00321 00324 00325 00338 00339 00340 00342 00344 00348 00348 00349 00351 00352 00353 00365 00368 00369 00382 00383 00384 00386 00388 00392 00392 00393 00395 00396 00397 00450 00468 00525 00525 00526 00527 00528 00529 00529 00531 00533 00534 00545 00546 00546 00547 00553 00554 00560 00561 00564 00565 00569 00570 00573 00574 00583 00584 00584 00586 00598 00599 00599 00600 00621 00622 00622 00623 00641 00642 00643 00650 00652 00654 00663 00665 00667 00693 00693 00694 00729 00729 00730 00763 00765 00767 00814 00815 00816 00816 00820 00848 00860 00861 00861 00867 00876 00876 00896 00902 00904 00906 00915
R10	00001	0000000A	00791	00418 00418 00423 00424 00424 00433 00456
R11	00001	0000000B	00792	00109 00110
R13	00001	0000000D	00794	00113
R14	00001	0000000E	00795	00263 00302 00346 00390 00452 00515 00518 00521 00523 00526 00540 00540 00544 00547 00594 00596 00597 00600 00608 00608 00619 00620 00656 00669 00769 00893 00894 00895 00895 00896 00908
R15	00001	0000000F	00796	00155 00155 00240 00247 00250 00251 00251 00262 00263 00273 00279 00286 00289 00290 00290 00301 00302 00314 00316 00317 00323 00330 00333 00334 00334 00345 00346 00358 00360 00361 00367 00374 00377 00378 00378 00389 00390 00402 00404 00405 00406 00407 00407 00451 00452 00504 00506 00506 00522 00522 00523 00524 00527 00534 00561 00565 00570 00574 00585 00586 00609 00617 00623 00655 00656 00668 00669 00768 00769 00770 00773 00773 00819 00819 00828 00829 00847 00847 00862 00877 00877 00878 00883 00883 00892 00893 00898 00898 00899 00907 00908 00909
R2	00001	00000002	00783	00164 00192 00193 00194 00219 00223 00223 00224 00454 00454 00464 00465 00642 00687 00687 00688 00730 00815 00822 00867 00868 00870 00870 00871 00873 00873 00885 00915 00916 00936
R3	00001	00000003	00784	00135 00169 00190 00193 00214 00217 00467 00472 00473 00643 00646 00660 00670 00672 00673 00674 00694 00695 00696 00743 00744 00745 00746 00747 00748 00749 00750 00751 00752 00753 00754 00755 00756 00757 00758 00864 00866 00874 00917 00918
R4	00001	00000004	00785	00162 00163 00174 00178 00184 00185 00198 00202 00208 00209 00226 00230 00248 00255 00287 00294 00331 00338 00375 00382 00466 00467 00468 00637 00677 00678 00735 00736 00736 00737 00863 00864
R5	00001	00000005	00786	00165 00165 00166 00169 00171 00172 00186 00186 00187 00190 00194 00195 00196 00210 00210 00211 00214 00218 00220 00221 00415 00416 00417 00419 00422 00423 00464 00636 00637 00638 00678 00679 00818 00818 00820 00821 00821 00822 00917 00918
R6	00001	00000006	00787	00168 00189 00213 00421 00421 00422 00455 00585 00592 00593 00606 00607 00638 00657 00659 00679 00701 00712 00820 00825 00834 00836 00838 00863 00885 00936
R7	00001	00000007	00788	00117 00118 00118 00119 00133 00164 00192 00216 00219 00541 00543 00824 00824 00825 00826 00829
R8	00001	00000008	00789	00239 00246 00249 00271 00278 00285 00288 00310 00322 00329 00332 00354 00366 00373 00376 00398 00505 00507 00814 00848 00849 00884 00910
R9	00001	00000009	00790	00111 00112
SETEQ1	00004	000006BA	00708	00703
SETEQ2	00004	000006D6	00724	00713
SETH1	00004	000006B2	00706	00702
SORTK	00002	0000002E	01126	01123
SPACEK	00002	0000002A	01086	01081

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
SUBAO1	00002	00001034	01838	01207 01221	
SUBAO2	00002	00001102	01910	01246 01256	
SUBAO3	00002	000011D0	01982	01277 01287	
SUBBREAK	00002	00000E32	01569	01165	
SUBCHARS	00002	00000E65	01598	01338	
SUBCONT	00002	00000DDB	01521	01071	
SUBDSNLN	00002	000012C7	02083	01472	
SUBEND	00002	00000DA9	01500	01050	
SUBFORMT	00002	0000129E	02054	01451	
SUBHEAD	00002	00000F2A	01719	01380	
SUBLEV	00002	00000D74	01479	01029	
SUBLIMIT	00002	00000F66	01766	01186	
SUBLINES	00002	00000EB7	01645	01359	
SUBLKEY	00004	000000F0	01786	00248 01775	
SUBLOPER	00004	000000F8	01809	00238 01796	
SUBLVALU	00004	00000100	01831	00241 01819	
SUBPRINT	00002	00000EE8	01672	01308	
SUBPRTIT	00004	000000D4	01712	01701	
SUBPRTKY	00004	000000CC	01692	01681	
SUBSORT	00004	000000A0	01562	01551	
SUBSORTS	00002	00000E16	01542	01135	
SUBTOTAL	00002	00000F34	01737	01410	
SUB1KEY	00004	00000108	01858	00287 01847	
SUB1OPER	00004	00000110	01881	00277 01868	
SUB1VALU	00004	00000118	01903	00280 01891	
SUB2KEY	00004	00000120	01930	00331 01919	
SUB2OPER	00004	00000128	01953	00321 01940	
SUB2VALU	00004	00000130	01975	00324 01963	
SUB3KEY	00004	00000138	02002	00375 01991	
SUB3OPER	00004	00000140	02025	00365 02012	
SUB3VALU	00004	00000148	02047	00368 02035	
TOTALK	00002	00000042	01401	01398	
TOTALN	00004	000000E8	01759	01746	
VADFORM	00004	0000002C	02260	00451	
VADMSG	00004	0000001C	02256	00262 00301 00345 00389 00655 00668 00768 00907	
VOLID	00006	000001B9	02312	00148	
VOLS	00004	00000008	01007	00996	
VTA0008	00004	00000266	00316	00313	
VTA0012	00004	000002FE	00360	00357	
VTA0016	00004	00000396	00404	00401	
VTCFMTCC	00001	00000008	02298	00236 00447 00453	
VTCFMTCD	00001	00000080	02297	00453	
VTCFMTCK	00001	000001A5	02296	00236 00447 00453	
VTCGETMN	00004	00000210	02331	02332	
VTE0005	00002	000001D2	00272	00267	
VTE0008	00004	0000026A	00317	00315	
VTE0009	00002	00000256	00311	00306	
VTE0012	00004	00000302	00361	00359	
VTE0013	00002	000002EE	00355	00350	
VTE0016	00004	0000039A	00405	00403	
VTE0017	00002	00000386	00399	00394	
VTFALLOC	00004	00000004	02415	00596 00611 00620	
VTFMT	00001	00000000	02413	00593 00607 02448	
VTFUSED	00004	00000008	02418	00594 00597 00609 00613	
VTOCCHK	00001	00000000	00095	00102 00989 01009 01021 01031 01042 01052 01063 01073 01087 01096 01107 01116 01127 01137	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
				01146 01157 01167 01178 01188 01199 01213 01227 01238 01248 01258 01269 01279 01289 01300	
				01310 01319 01330 01340 01351 01361 01372 01382 01391 01402 01412 01423 01432 01443 01453	
				01464 01474 01480 01494 01501 01515 01522 01536 01543 01563 01570 01592 01599 01619 01639	
				01646 01666 01673 01693 01713 01720 01731 01738 01760 01767 01787 01810 01832 01839 01859	
				01882 01904 01911 01931 01954 01976 01983 02003 02026 02048 02055 02077 02084 02106 02119	
				02128 02137 02146	
VTOCOM	00001	00000000	02242	00110 00261 00300 00344 00388 00450 00654 00667 00767 00906	
VTP0005	00002	000001B8	00264	00244 00252	
VTP0009	00002	0000023C	00303	00283 00291	
VTP0013	00002	000002D4	00347	00327 00335	
VTP0017	00002	0000036C	00391	00371 00379	
ZERO	00004	00000A38	00950	00541 00611 00613	

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

=A(72*1000)				
	00004	00000B40	02558	00531
=F'100000'				
	00004	00000B44	02559	00533
=XL5'0000000000'				
	00005	00000B48	02560	00140

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 793

TOTAL RECORDS READ FROM SYSTEM LIBRARY 3055

TOTAL RECORDS PUNCHED 92

TOTAL RECORDS PRINTED 1703

SYMBOL	TYPE	ID	ADDR	LENGTH	LDID
--------	------	----	------	--------	------

ASM 0201 13.38 01/07/25

VTOCEXCP	SD	0001	000000	000754	
----------	----	------	--------	--------	--


```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25
2  *****
3  *          SPACE
4  * AUTHOR;  R. F. MORSE, MIT INSTRUMENTATION LABORATORY  AUG 5,1968.
5  * MODIFIED;  E.BANK, FIREMAN'S FUND  MAY 15,1975.
6  * MODIFIED;  R.MILLER  FIREMAN'S FUND  MAR 20,1977.
7  * FUNCTION; THIS SUBROUTINE READS THE VOLUME TABLE OF CONTENTS (VTOC)
8  *          FROM A DIRECT-ACCESS DEVICE AND PRESENTS IT TO THE CALLER
9  *          ONE RECORD (DSCB) AT A TIME.
10 *
11 * OPERATION; THIS ROUTINE IS A SPECIALIZED SEQUENTIAL ACCESS METHOD
12 *          FOR VTOC'S.  ITS ADVANTAGE OVER ORDINARY BSAM IS THAT IT READS
13 *          AN ENTIRE TRACK IN ONE REVOLUTION, THUS SAVING CONSIDERABLE
14 *          TIME.  THE ROUTINE HAS THREE CALL MODES;
15 *
16 *          0 - READ.  RETURNS WITH THE CORE ADDRESS OF A DSCB IN THE 3RD
17 *          PARAMETER.  THE CORE CONSISTS OF 148 CONSECUTIVE BYTES,
18 *          CONTAINING THE COUNT (8 BYTES), KEY (44 BYTES), AND DATA
19 *          (96 BYTES) FOR ONE DSCB.  RETURN CODES (REGISTER 15)
20 *          ARE;
21 *
22 *          0 - NORMAL;
23 *          4 - END OF FILE, NO DATA PRESENTED;
24 *          8 - PERMANENT I/O ERROR.  THE KEY AND DATA AREAS
25 *          WILL BE SET TO ZEROS; THE COUNT AREA WILL
26 *          CONTAIN THE CORRECT CCHHR.  SINCE READING
27 *          IS DONE A TRACK AT A TIME, ALL THE DSCB'S
28 *          FOR THAT TRACK WILL BE MARKED IN ERROR.
29 *          READING MAY CONTINUE ON TO THE NEXT TRACK.
30 *
31 *          1 - OPEN.  THE SECOND PARAMETER SHOULD POINT TO
32 *          A 6-BYTE FIELD CONTAINING THE VOLSER TO BE USED FOR THE
33 *          ALLOCATION.
34 *          RETURN CODES ( REG 15 ) ARE DIRECT FROM DYNAMIC ALLOC.
35 *          0 - NORMAL;
36 *          4 - UNABLE TO OPEN (PROBABLY MISSING DD CARD);
37 *          8 - DD CARD DID NOT REFER TO A DIRECT-ACCESS
38 *          DEVICE, OR DEVICE TYPE UNKNOWN.
39 *
40 *          2 - CLOSE.  NO ARGUMENTS ARE REQUIRED OR RETURNED.  RETURN
41 *          CODE ( REG 15 ) IS FROM DYNAMIC UNALLOCATION.
42 * ENTRY POINTS:  ENTRY IS ALWAYS TO 'VTOCEXCP'.
43 * ARGUMENTS ARE:
44 *          1 - A(FULL-WORD BINARY ENTRY TYPE);
45 *          2 - A(PTR FOR DSCB);
46 *          3 - A(VOLSER).
47 * DATA SETS:  READS VOLUME TABLE OF CONTENTS FROM ANY DIRECT-ACCESS
48 *          DEVICE.  USES EXCP TO EXECUTE A CHAINED CHANNEL PROGRAM TO
49 *          READ AN ENTIRE TRACK AT A TIME.
50 *
51 * EXTERNAL ROUTINES:  USES SUPERVISOR ROUTINE 'IEPCNVT' TO CONVERT
52 *          A RELATIVE TRACK NUMBER TO AN ABSOLUTE ADDRESS.
53 *
54 * EXITS - NORMAL;  RETURNS TO CALLER VIA R14 WITH RETURN
55 *          CODE IN REGISTER 15.  (SEE ABOVE FOR RETURN CODE VALUES.)
56 *

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				57 *	TABLES AND WORK AREAS; USES AN AREA PROVIDED BY THE CALLER FOR	
				58 *	ITS SAVEAREA AND FOR WORKING STORAGE IMMEDIATELY FOLLOWING	
				59 *	THE PRIOR SAVEAREA. IT USES GETMAIN TO OBTAIN AN AREA FOR	
				60 *	THE DSCB'S TO BE READ INTO. THIS COULD BE AS LARGE AS	
				61 *	8K FOR 3350'S. IT IS FREED BY THE FINAL CALL.	
				62 *		
				63 *	ATTRIBUTES; REENTRANT, REFRESHABLE.	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				65	* ENTER HERE AND PERFORM STANDARD REGISTER SAVE AREA HOUSEKEEPING.	
				67	VTOCEXCP ENTER 12,8	USE THE PROVIDED SAVEAREA
000000				68+	VTOCEXCP CSECT	
000000	47F0 F00E	0000E		69+	B 14(0,15)	BRANCH AROUND ID 00450000
000004	08			70+	DC AL1(8)	LENGTH OF IDENTIFIER 00550000
000005	E5E3D6C3C5E7C3D7			71+	DC CL8 'VTOCEXCP'	IDENTIFIER 00750000
00000D	00					
00000E	90EC D00C	0000C		72+	STM 14,12,12(13)	SAVE REGISTERS 03700000
000012	18CF			73+	LR 12,15	SET FIRST BASE REG
000014				74+	CNOP 0,4	
			00000	75+	USING VTOCEXCP,12	
000014	5811 0008	00008		76+	L 1,8+0(1)	NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
000018	50D1 0004	00004		77+	ST 13,4(1)	PRIOR SAVEAREA ADDRESS TO MINE
00001C	501D 0008	00008		78+	ST 1,8(13)	MY SAVEAREA ADDRESS TO HIS
000020	182D			79+	LR 2,13	KEEP THE SAVEAREA ADDRESS FOR REGS
000022	18D1			80+	LR 13,1	THIS IS MY SAVEAREA
000024	9802 2014	00014		81+	LM 0,2,20(2)	RESTORE ORIGINAL REGS
			00000	82	USING VTOWORK,R13	SET ADDRESSABILITY FOR WORK AREA
000028	18B1			83	LR R11,R1	SAVE PARAMETER REGISTER
			00000	84	USING VTOCOM,R11	SET ADDRESSABILITY
				85	*	
				86	*	POINT TO THE DCB FOR LATER REFERENCES
				87	*	
00002A	4180 D0C8	000C8		88	LA RDCB,VTOWDCB	POINT TO IT
			00000	89	USING IHADCB,RDCB	SET ADDRESSABILITY
				91	*	SELECT MODE FROM CONTENTS AT ADDRESS IN REGISTER 1.
00002E	1B22			93	SR RWA,RWA	CLEAR THE REGISTER
000030	4320 B1A4	001A4		94	IC RWA,VTCEFUNG	GET CALL MODE
000034	8920 0002	00002		95	SLL RWA,2	MODE TIMES 4
000038	47F2 C03C	0003C		96	B *+4(RWA)	BRANCH ON MODE
00003C	47F0 C062	00062		98	B GETDSB	MODE 0, GET A DSCB
000040	47F0 C128	00128		99	B OPEN	MODE 1, OPEN A NEW VTOC
000044	47F0 C39A	0039A		100	B CLOSE	MODE 2, CLOSE
000048	47F0 C04C	0004C		101	B RETURN0	MODE 3 NOT DEFINED, NO OP
				103	*****	
				104	* RETURNS *	
				105	*****	
00004C	1BFF			107	RETURN0 SR R15,R15	CLEAR THE RETURN CODE
				108	RETURN LEAVE EQ	EXIT WITH THE CURRENT RET CODE
00004E	182D			109+	RETURN LR 2,13	
000050	58DD 0004	00004		110+	L 13,4(13)	
000054	90F1 D010	00010		111+	STM 15,1,16(13)	STORE RETURN REGS
000058	98EC D00C	0000C		112+	LM 14,12,12(13)	RESTORE THE REGISTERS 00650000
00005C	92FF D00C	0000C		113+	MVI 12(13),X'FF'	SET RETURN INDICATION 01600000
000060	07FE			114+	BR 14	RETURN 02000000
			00000	115+R0	EQU 0	*USED BY O.S.
			00001	116+R1	EQU 1	*USED BY O.S. // ADDRESS OF PARAMETER LIST

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
			00002	117+R2	EQU	2	
			00003	118+R3	EQU	3	
			00004	119+R4	EQU	4	
			00005	120+R5	EQU	5	
			00006	121+R6	EQU	6	
			00007	122+R7	EQU	7	
			00008	123+R8	EQU	8	
			00009	124+R9	EQU	9	
			0000A	125+R10	EQU	10	
			0000B	126+R11	EQU	11	
			0000C	127+R12	EQU	12	
			0000D	128+R13	EQU	13	*USED BY O.S. // SAVE-AREA ADDRESS
			0000E	129+R14	EQU	14	*USED BY O.S. // RETURN ADDRESS
			0000F	130+R15	EQU	15	*USED BY O.S. // ENTRY-PT ADDR, RETURN CODE

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25

                                           132 *****
                                           133 * MODE 0 - GET DSCB *
                                           134 *****

                                           136 * IF END-OF-FILE WAS REACHED, RETURN AT ONCE.

000062 41F0 0004      00004      138 GETDSB  LA      R15,4          SET THE RETURN CODE, IN CASE
000066 9110 D0C4      000C4      139          TM      MODESW,EOFSW    TEST END-OF-FILE BIT
00006A 4710 C04E      0004E      140          BO      RETURN          RETURN CODE 4 IF ON

                                           142 * IF CHANNEL PROGRAM HAS BEEN STARTED, GO TO CHECK IT.  OTHERWISE,
                                           143 * ASSUME THERE IS AT LEAST ONE FULL BUFFER.

00006E 9140 D0C4      000C4      145          TM      MODESW,XCPRUN    TEST IF EXCP ISSUED
000072 4710 C0BE      000BE      146          BO      XCPTEST          BRANCH IF SO

                                           148 * SET BUFFER ADDRESS TO NEXT DSCB AND TEST IF LAST ON TRACK.  IF NOT,
                                           149 * EXIT WITH ITS ADDRESS IN R1.

000076 5820 D0A4      000A4      151          L      RWA,DSCBADR        LOAD BUFFER POINTER
00007A 4122 0094      00094      152          LA      RWA,148(RWA)      ADVANCE TO NEXT DSCB
00007E 5020 D0A4      000A4      153 NDXSTORE ST      RWA,DSCBADR        STORE UPDATED POINTER
000082 5920 D0A0      000A0      154          C      RWA,DSCBLIM       TEST IF LAST DSCB IN BUFFER
000086 47B0 C0A0      000A0      155          BNL    LASTDSCB          BRANCH IF SO
00008A 1812              156          LR      R1,RWA           PASS ADDRESS TO USER
00008C 5010 B194      00194      157 GETOUT  ST      R1,DSCBADDR      STORE IT FOR THE CALLER
000090 9120 D0C4      000C4      158          TM      MODESW,RDERR     TEST IF ERROR ON THIS TRACK
000094 4780 C04C      0004C      159          BZ      RETURN0          RETURN CODE 0 IF NOT
000098 41F0 0008      00008      160          LA      R15,8            SET THE RETURN CODE
00009C 47F0 C04E      0004E      161          B      RETURN          RETURN CODE 8 IF ERROR

                                           163 * IF THIS IS THE LAST DSCB, MOVE IT TO THE INTERNAL BUFFER AND START
                                           164 * READING THE NEXT TRACK.

0000A0 D293 D134 2000 00134 00000 166 LASTDSCB MVC     BUFF(148),0(RWA)    MOVE LAST DSCB
0000A6 5830 D0A8      000A8      167          L      RWB,TTRN          LOAD RELATIVE TRACK NUMBER
0000AA 5E30 C740      00740      168          AL     RWB,=X'00010000'  INCREMENT TO NEXT TRACK
0000AE 5030 D0A8      000A8      169          ST     RWB,TTRN          STORE TRACK NUMBER
0000B2 4590 C480      00480      170          BAL    RRET,EXCP         START CHANNEL PROGRAM
0000B6 4110 D134      00134      171          LA     R1,BUFF           LOAD DSCB ADDRESS FOR CALLER
0000BA 47F0 C08C      0008C      172          B      GETOUT           TO RETURN

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

      174 * WAIT FOR CHANNEL PROGRAM COMPLETION AND TEST THE OUTCOME.

0000BE 4110 D130      00130      176 XCPTEST  WAIT  ECB=VTOCECB
0000C2 4100 0001      00001      177+XCPTEST LA   1,VTOCECB          LOAD PARAMETER REG 1  01900002
0000C6 0A01          00001      178+          LA   0,1(0,0)          COUNT OMITTED,1 USED  00950002
0000C8 94BF D0C4      000C4      179+          SVC  1                  LINK TO WAIT ROUTINE  01400002

0000C8 94BF D0C4      000C4      181          NI   MODESW,X'FF'-XCPRUN  TURN EXCP STARTED BIT OFF
0000CC 957F D130      00130      182          CLI  VTOCECB,X'7F'       TEST COMPLETION CODE
0000D0 4770 C0DC      000DC      183          BNE  PERMERR            BRANCH IF ERROR
0000D4 5820 D09C      0009C      184 SETDSCBA L   RWA,DSCBSTRT        SET BUFFER POINTER TO 1ST DSCB
0000D8 47F0 C07E      0007E      185          B    NDXSTORE

      187 * PERMANENT ERROR FOR THIS TRACK. ZERO THE DSCB'S AND FILL IN THE
      188 * CCHHR PORTIONS OF THE COUNT AREAS.

0000DC 9620 D0C4      000C4      190 PERMERR  OI   MODESW,RDERR        SIGNAL READ ERROR
0000E0 94FB D108      00108      191          NI   IOBFLAG1,X'FB'     TURN OFF BIT 5 OF IOB FLAG
0000E4 943F 802C      0002C      192          NI   DCBIFLGS,X'3F'     TURN OFF BITS 0 AND 1
0000E8 5820 D09C      0009C      193          L    RWA,DSCBSTRT        LOAD ADDRESS OF FIRST DSCB
0000EC 4130 0001      00001      194          LA   RWB,1              LOAD RECORD NUMBER

0000F0 D793 2000 2000 00000 00000      196 DSCBELUP XC   0(148,RWA),0(RWA)      ZERO DSCB BUFFER
0000F6 D203 2000 D12B 00000 0012B      197          MVC  0(4,RWA),IOBSEEK+3  INSERT CCHH IN COUNT FIELD
0000FC 4232 0004      00004      198          STC  RWB,4(RWA)         INSERT R IN COUNT FIELD
000100 4122 0094      00094      199          LA   RWA,148(RWA)        POINT TO NEXT BUFFER
000104 4133 0001      00001      200          LA   RWB,1(RWB)         INCREMENT RECORD NUMBER
000108 5920 D0A0      000A0      201          C    RWA,DSCBLIM        TEST FOR LAST BUFFER
00010C 47D0 C0F0      000F0      202          BNH  DSCBELUP
      203          VTOCMSG TRACKERR      ISSUE THE ERROR MESSAGE
000110 4110 C63C      0063C      204+          LA   R1,TRACKERR        POINT TO THE FIRST MESSAGE
000114 1B00          00000      205+          SR   R0,R0              NO SECOND LEVEL MESSAGE
000116 9001 B078      00078      206+          STM  R0,R1,MSGADDRS    SAVE THE MESSAGE ADDRESSES
      207+*          THEN JUST CALL THE MESSAGE ISSUING ROUTINE
00011A 4110 B000      00000      208+          LA   R1,VTOCOM          POINT TO THE COMMON AREA
00011E 58F0 B01C      0001C      209+          L    R15,VADMSG         POINT TO THE ROUTINE
000122 05EF          00000      210+          BALR R14,R15           THEN CALL IT
000124 47F0 C0D4      000D4      211          B    SETDSCBA           BRANCH TO RESET BUFFER POINTER

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

      213 *****
      214 * MODE 1 - OPEN *
      215 *****

      217 * ENTER WITH A DDNAME IN SECOND PARAMETER POSITION.  PERFORM CLOSE
      218 * SUBROUTINE FIRST TO BE SURE EVERYTHING IS INITIALIZED.

000128      220 OPEN      DS      0H
000128 4590 C3A2      003A2  221      BAL      RRET,CLOSESUB      CALL CLOSE SUBROUTINE

      223 *
      224 *      INITIALIZE THE DATA AREAS
      225 *
      226 *      FIRST THE DCB
00012C D233 D0C8 C6B8 000C8 006B8  227      MVC      VTOCDCB(DCBLLEN),VTOCDCBM  SET UP THE DCB
      228 *
      229 *      SET UP THE JFCB LISTS
      230 *
000132 4110 D280      00280  231      LA      R1,JEXLST      POINT TO THE EXIT LIST
000136 BE17 8025      00025  232      STCM   R1,B'0111',DCBEXLSA  PUT IT INTO THE DCB
00013A 4110 D284      00284  233      LA      R1,JFCBAREA     POINT TO THE JFCB AREA
00013E 5010 D280      00280  234      ST      R1,JEXLST      AND PUT THAT INTO THE EXIT LIST
000142 9287 D280      00280  235      MVI     JEXLST,X'87'    END OF LIST, JFCB EXIT
000146 9280 D0FC      000FC  236      MVI     OPENLIST,X'80' END OF THE OPEN LIST TOO
      237 *
00014A D227 D108 C6F0 00108 006F0  238      MVC      VTOCIOB(IOBCONL),IOBCONST START IT OUT
000150 4110 D130      00130  239      LA      R1,VTOCECB     GET THE ECB ADDRESS
000154 5010 D10C      0010C  240      ST      R1,IOBECB     AND STORE IT INTO THE IOB
000158 5080 D11C      0011C  241      ST      RDCB,IOBDCB   STORE THE DCB ADDRESS INTO THE IOB
      242 *
00015C D203 D1C8 C6A8 001C8 006A8  243      MVC      DSCBFMT4(4),DSCBCON SET UP THE FIRST WORD
000162 4110 D12B      0012B  244      LA      R1,IOBSEEK+3   SEEK ADDRESS
000166 5010 D1CC      001CC  245      ST      R1,DSCBFMT4+4  INTO THE CAMLST
00016A 4110 B1B9      001B9  246      LA      R1,VOLID      VOLUME SERIAL NUMBER
00016E 5010 D1D0      001D0  247      ST      R1,DSCBFMT4+8  INTO THE CAMLST
000172 4110 B6E0      006E0  248      LA      R1,FMT4      DSCB AREA
000176 5010 D1D4      001D4  249      ST      R1,DSCBFMT4+12 INTO THE CAMLST
      250 *
      251 *      ALLOCATE THE VTOC OF THE CHOSEN PACK
      252 *
00017A 4110 B1B0      001B0  253      LA      R1,ADDR      POINT TO THE UNIT ADDRESS
00017E 5010 D0B4      000B4  254      ST      R1,UNITADDR   SAVE THE ADDRESS
000182 4110 0003      00003  255      LA      R1,3          ALSO GET THE LENGTH
000186 4010 D0B8      000B8  256      STH     R1,UNITLEN    AND SAVE IT FOR DYNAMIC ALLOCATION MACRO
00018A 4110 B1B9      001B9  257      LA      R1,VOLID      POINT TO THE VOLUME SERIAL
00018E 5010 D0AC      000AC  258      ST      R1,VOLADDR    SAVE THE ADDRESS
000192 4110 0006      00006  259      LA      R1,6          ALSO GET THE LENGTH
000196 4010 D0B0      000B0  260      STH     R1,VOLLEN    AND SAVE IT FOR DYNAMIC ALLOCATION MACRO
      261      ALLOC   DSN=VTOCNM,VOL=VOLADDR,UNIT=UNITADDR,DISP=SHR,      X
      DDNTO=DCBDDNAM,ERROR=S99FAIL

00019A      262+      DS      0H
00019A 4110 D1D8      001D8  263+      LA      R1,DYNSP1      LOAD ADDRESS OF PARAM LIST
      00000  264+      USING  DYN1DS,R1      USE GENERATED DSECT
00019E 41F0 1004      00004  265+      LA      R15,DYN1RB     LOAD ADDRESS OF S99 RB
      00000  266+      USING  S99RB,R15

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

0001A2 50F1 0000      00000      267+      ST    R15,0(R1)      AND STORE IN RB POINTER
0001A6 D78B 1004 1004 00004 00004      268+      XC    4(DYN1LEN-4,R1),4(R1)  ZERO PARAMETER LIST
0001AC 9214 F000      00000      269+      MVI   S99RBLN,20      MOVE IN LIST LENGTH
0001B0 9201 F001      00001      270+      MVI   S99VERB,S99VRBAL  MOVE IN VERB CODE
0001B4 41E0 1018      00018      271+      LA    R14,DYN1TUP      LOAD ADDRESS OF TU POINTERS
0001B8 50E0 F008      00008      272+      ST    R14,S99TXTPP      STORE ADDRESS IN S99 RB
0001BC 41F0 102C      0002C      273+      LA    R15,DYN1TU      POINT TO SPACE FOR TEXT UNITS
                   00000      274+      USING S99TUNIT,R15

276+*****
277+**      BUILD THE DSNAME TEXT UNIT      **
278+*****
0001C0 1802      279+      LR    R0,R2      SAVE CONTENTS OF REGISTER 2
0001C2 58E0 C718      00718      280+      L     R14,VTOCNM      LOAD ADDRESS OF DSNAME
0001C6 4820 C71C      0071C      281+      LH    R2,VTOCNM+4      LOAD LENGTH OF DSNAME
0001CA 4020 F004      00004      282+      STH   R2,S99TULNG      STORE DSNAME LENGTH
0001CE 0620      283+      BCTR  R2,0      DECREMENT FOR EXECUTE
0001D0 4420 C268      00268      284+      EX    R2,DYN1MVC      MOVE DSNAME
0001D4 9202 F001      00001      285+      MVI   S99TUKEY+1,DALDSNAM  MOVE IN DSNAME KEY
0001D8 9201 F003      00003      286+      MVI   S99TUNUM+1,1      SET NUMBER FIELD
0001DC 50F0 1018      00018      287+      ST    R15,DYN1TUP+0      STORE TEXT UNIT ADDRESS
0001E0 41FF 0032      00032      288+      LA    R15,50(R15)      BUMP TEXT UNIT PTR TO NEXT SLOT

290+*****
291+**      DDNAME RETURN TEXT UNIT      **
292+*****
0001E4 9255 F001      00001      293+      MVI   S99TUKEY+1,DALRTDDN  SET RETURN DDNAME KEY
0001E8 9201 F003      00003      294+      MVI   S99TUNUM+1,1      SET NUMBER FIELD
0001EC 9208 F005      00005      295+      MVI   S99TULNG+1,8      SET LENGTH FIELD
0001F0 D207 F006 C730 00006 00730      296+      MVC   S99TUPAR(8),=CL8' '  INITIALIZE FIELD TO BLANKS
0001F6 50F0 101C      0001C      297+      ST    R15,DYN1TUP+4      STORE TEXT UNIT ADDRESS
0001FA 41FF 000E      0000E      298+      LA    R15,14(R15)      BUMP TEXT UNIT PTR TO NEXT SLOT

300+*****
301+**      UNIT NAME TEXT UNIT      **
302+*****
0001FE 4820 D0B8      000B8      303+      LH    R2,UNITADDR+4      LOAD LENGTH OF TEXT UNIT
000202 1222      304+      LTR   R2,R2      TEST FOR ZERO
000204 4780 C222      00222      305+      BZ    *+30      IF NO TEXT UNIT, SKIP
000208 58E0 D0B4      000B4      306+      L     R14,UNITADDR      LOAD ADDRESS OF TEXT UNIT
00020C 4020 F004      00004      307+      STH   R2,S99TULNG      STORE LENGTH OF TEXT UNIT
000210 0620      308+      BCTR  R2,0      DECREMENT FOR EXECUTE
000212 4420 C268      00268      309+      EX    R2,DYN1MVC      MOVE IN TEXT UNIT
000216 9215 F001      00001      310+      MVI   S99TUKEY+1,DALUNIT  MOVE IN TEXT UNIT KEY
00021A 9201 F003      00003      311+      MVI   S99TUNUM+1,1      SET NUMBER FIELD
00021E 50F0 1020      00020      312+      ST    R15,DYN1TUP+8      STORE TEXT UNIT ADDRESS
000222 41FF 000E      0000E      313+      LA    R15,14(R15)      BUMP TEXT UNIT PTR TO NEXT SLOT

315+*****
316+**      VOLUME SERIAL TEXT UNIT      **
317+*****
000226 4820 D0B0      000B0      318+      LH    R2,VOLADDR+4      LOAD LENGTH OF TEXT UNIT
00022A 1222      319+      LTR   R2,R2      TEST FOR ZERO
00022C 4780 C24A      0024A      320+      BZ    *+30      IF NO TEXT UNIT, SKIP
000230 58E0 D0AC      000AC      321+      L     R14,VOLADDR      LOAD ADDRESS OF TEXT UNIT

```



```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

000234 4020 F004      00004      322+      STH  R2,S99TULNG      STORE LENGTH OF TEXT UNIT
000238 0620              323+      BCTR R2,0             DECREMENT FOR EXECUTE
00023A 4420 C268      00268      324+      EX   R2,DYN1MVC       MOVE IN TEXT UNIT
00023E 9210 F001      00001      325+      MVI  S99TUKEY+1,DALVLSER  MOVE IN TEXT UNIT KEY
000242 9201 F003      00003      326+      MVI  S99TUNUM+1,1      SET NUMBER FIELD
000246 50F0 1024      00024      327+      ST   R15,DYN1TUP+12    STORE TEXT UNIT ADDRESS
00024A 41FF 000E      0000E      328+      LA   R15,14(R15)      BUMP TEXT UNIT PTR TO NEXT SLOT

                                330+*****
00024E D207 F000 C738 00000 00738 333+      MVC  S99TUKEY(8),=Y(DALSTATS,1,1,X'0800')
000254 50F0 1028      00028      334+      ST   R15,DYN1TUP+16    STORE TEXT UNIT ADDRESS
000258 41FF 0008      00008      335+      LA   R15,8(R15)      BUMP TEXT UNIT PTR TO NEXT SLOT

00025C 9280 1028      00028      337+      MVI  DYN1TUP+16,X'80'   SET HIGH ORDER BIT ON TEXT PTRS
000260 9280 1000      00000      338+      MVI  DYN1RBP,X'80'     SET HIGH ORDER BIT ON RB PTR
000264 47F0 C26E      0026E      339+      B    *+10              SKIP NEXT INSTRUCTION
000268 D200 F006 E000 00006 00000 340+DYN1MVC MVC  S99TUPAR(0),0(R14)  EXECUTED MOVE
00026E 1820              341+      LR   R2,R0             RESTORE CONTENTS OF REGISTER 2
                                342+      DROP R1,R15           DEACTIVATE ADDRESSABILITY
000270 41E1 0004      00004      343+      LA   R14,4(R1)        POINT TO REQUEST BLOCK
                                344+*      MACDATE Y-2 73082      00300002
000274 0A63              345+      SVC  99 CALL DYNAMIC ALLOCATION 00400002
                                346+      USING DYN1RB,R14      SET UP ADDRESSABILITY
                                347+**      NOTE R14 HAS RB ADDRESS, R15 HAS SVC 99 RETURN CODE **
000276 12FF              348+      LTR  R15,R15          TEST RETURN CODE
000278 4770 C4D0      004D0      349+      BNZ  S99FAIL          BRANCH IF NON ZERO
00027C D207 8028 E060 00028 00064 350+      MVC  DCBDDNAM(8),DYN1TU+54+2

000000              352+DYN1DS  DSECT          DSECT TO MAP SVC 99 DATA
000000              353+DYN1RBP DS  F          SVC 99 REQ BLOCK POINTER
000004              354+DYN1RB  DS  5F        SVC 99 REQUEST BLOCK
000018              355+DYN1TUP DS  CL20      SPACE FOR TEXT POINTERS
00002C              356+DYN1TU  DS  CL100     SPACE FOR TEXT UNITS
                                357+DYN1LEN  EQU  *-DYN1RBP  LENGTH OF SPACE USED
000282              358+VTOCEXCP CSECT

000282 9608 D0C4      000C4      360      OI   MODESW,ALLOCSW    SET ALLOCATE FLAG ON

                                362 * OPEN THE VTOC.

                                364 *
                                365 *      FIRST READ THE JFCB TO SWITCH THE DSNAME TO HEX 04'S
                                366 *
                                367      RDJFCB ((RDCB)),MF=(E,OPENLIST)  READ THE JFCB
000286 4110 D0FC      000FC      368+      LA   1,OPENLIST        LOAD PARAMETER REG 1 01900002
00028A 43E1 0000      00000      369+      IC   14,0(1,0)        SAVE OPTION BYTE 03440000
00028E 5081 0000      00000      370+      ST   RDCB,0(1,0)     STORE INTO LIST 03620000
000292 42E1 0000      00000      371+      STC  14,0(1,0)        RESTORE OPTION BYTE 03520001
000296 0A40              372+      SVC  64                ISSUE RDJFCB SVC 00200000
000298 12FF              373      LTR  R15,R15          TEST THE RETURN CODE
00029A 4770 C578      00578      374      BNZ  ERRJFCB          BAD NEWS

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
00029E	4110 D284	00284		375	LA R1,JFCBAREA POINT TO THE JFCB	
			00000	376	USING JFCB,R1 SET UP ADDRESSABILITY	
0002A2	9204 1000	00000		377	MVI JFCBDSNM,X'04' PUT IN THE FIRST ONE	
0002A6	D22A 1001 1000	00001 00000		378	MVC JFCBDSNM+1(L'JFCBDSNM-1),JFCBDSNM PROPAGATE IT	
0002AC	9608 1034	00034		379	OI JFCBTSDM,JFCNWRIT DON'T REWRITE IT	
				380	DROP R1	
				381	OPEN ((RDCB),(INPUT)),MF=(E,OPENLIST),TYPE=J OPEN THE VTOC	
0002B0	4110 D0FC	000FC		382+	LA 1,OPENLIST LOAD PARAMETER REG 1	01900002
0002B4	94F0 1000	00000		383+	NI 0(1),X'F0' CLEAR OPTION 1 BITS	03780001
0002B8	43E1 0000	00000		384+	IC 14,0(1,0) SAVE OPTION BYTE	03440000
0002BC	5081 0000	00000		385+	ST RDCB,0(1,0) STORE INTO LIST	03620000
0002C0	42E1 0000	00000		386+	STC 14,0(1,0) RESTORE OPTION BYTE	03520001
0002C4	0A16			387+	SVC 22 ISSUE OPENJ SVC	04040000
0002C6	9110 8030	00030		388	TM DCBOFLGS,OPENBIT TEST IF OPEN WORKED	
0002CA	4780 C518	00518		389	BZ OPENERR ERROR IF OPEN FAILED	
				391	* ISSUE AN OBTAIN FOR THE FIRST DSCB ON THE VTOC (FORMAT 4)	
0002CE	902D D048	00048		392	D3 STM R2,R13,EXCPSAVE SAVE OUR REGS	
0002D2	4130 D048	00048		393	LA R3,EXCPSAVE POINT TO THE REGISTER SAVE AREA	
0002D6	BF0F C744	00744		394	ICM R0,B'1111',=X'00000100' FIRST DSCB	
0002DA	5810 802C	0002C		395	L R1,DCBDEBAD DEB ADDRESS	
0002DE	4120 D128	00128		396	LA R2,I0BSEEK SAVE ADDRESS OF CCHHR	
0002E2	58F0 0010	00010		397	L R15,CVT GET ADDRESS OF CVT	
0002E6	58FF 001C	0001C		398	L R15,CVTPCNVT(R15) GET ADDRESS OF CONVERT ROUTINE	
0002EA	05EF			399	BALR R14,R15 GO TO CONVERT ROUTINE	
0002EC	982D 3000	00000		400	LM R2,R13,0(R3) GET MY REGS BACK	
				401	OBTAIN DSCBFMT4 GET FORMAT 4 DSCB	
0002F0	4110 D1C8	001C8		402+	LA 1,DSCBFMT4 LOAD PARAMETER REG 1	01900002
0002F4	0A1B			403+	SVC 27 ISSUE OBTAIN SVC	00100019
0002F6	12FF			404	LTR R15,R15 DID WE GET IT	
0002F8	4770 C530	00530		405	BNZ OBTErr NO - THEN ERROR, KEEP R15	
0002FC	95F4 B70C	0070C		406	CLI DS4IDFMT,X'F4' MAKE SURE WE HAVE FORMAT 4	
000300	4770 C548	00548		407	BNE NOTFMT4 NO - THEN ERROR	
000304	43F0 B72A	0072A		408	IC R15,DS4DEVDT GET NUMBER OF DSCBS PER TRACK	
000308	50F0 D098	00098		409	ST R15,NDSCBS SAVE THE NUMBER OF DSCBS	
00030C	D603 D098 D098	00098 00098		410	OC NDSCBS,NDSCBS MAKE SURE NOT ZERO	
000312	4780 C560	00560		411	BZ DSCBNUM0 YES - GO TELL CALLER	
				412	*	
				413	* OBTAIN CORE FOR CHANNEL PROGRAM AND DSCB BUFFERS.	
000316	4100 009C	0009C		415	LA R0,156 CORE FOR ONE DSCB AND ITS CCW	
00031A	4C00 D09A	0009A		416	MH R0,NDSCBS+2 TIMES NUMBER PER TRACK	
00031E	4A00 C750	00750		417	AH R0,=H'15' PLUS 1 CCW AND ROUNDING	
000322	5400 C748	00748		418	N R0,=X'FFFFFFFF8' ROUND TO DOUBLE-WORD MULTIPLE	
000326	5000 D090	00090		419	ST R0,CBSIZE SAVE SIZE OF GOTTEN CORE	
				420	GETMAIN R,LV=(0) GET TRACK BUFFERS	
00032A	4510 C32E	0032E		421+	BAL 1,*+4 INDICATE GETMAIN	
00032E	0A0A			422+	SVC 10 ISSUE GETMAIN SVC	
000330	5010 D094	00094		423	ST R1,CBADDR SAVE ADDRESS OF GOTTEN CORE	
000334	9680 D0C4	000C4		424	OI MODESW,CBGOT INDICATE CORE GOTTEN	
				426	* GENERATE CHANNEL PROGRAM. IT CONSISTS OF A 'READ R0' ORDER WITH	
				427	* THE SKIP FLAG ON, FOLLOWED BY A 'READ COUNT-KEY-AND-DATA' ORDER FOR	
				428	* EACH DSCB.	

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000338	5820	D098	00098		430	L	RWA,NDSCBS	NUMBER OF DSCB'S
00033C	8920	0003	00003		431	SLL	RWA,3	TIMES 8
000340	4122	1008	00008		432	LA	RWA,8(RWA,R1)	PLUS 8 AND BASE = 1ST BUFFER ADD
000344	5020	D09C	0009C		433	ST	RWA,DSCBSTRT	SAVE ADDRESS OF FIRST BUFFER
000348	5010	D118	00118		435	ST	R1,IOBSTART	ADDRESS OF CHANNEL PROGRAM
00034C	D207	1000	C698 00000	00698	436	MVC	0(8,R1),INITCCW	INSERT FIRST CCW
000352	4131	0008	00008		437	LA	RWB,8(R1)	PLACE FOR NEXT CCW
000356	4140	0001	00001		438	LA	RWC,1	BUFFER COUNTER
00035A	D207	3000	C6A0 00000	006A0	440	CCWLOOP MVC	0(8,RWB),READCCW	INSERT READ CCW FOR ONE DSCB
000360	5023	0000	00000		441	ST	RWA,0(RWB)	SET ITS BUFFER ADDRESS
000364	921E	3000	00000		442	MVI	0(RWB),READCKD	RESTORE COMMAND CODE
000368	5940	D098	00098		443	C	RWC,NDSCBS	TEST BUFFER COUNTER
00036C	47B0	C380	00380		444	BNL	LASTCCW	BRANCH IF LAST BUFFER
000370	4133	0008	00008		445	LA	RWB,8(RWB)	INCREMENT CCW ADDRESS
000374	4122	0094	00094		446	LA	RWA,148(RWA)	INCREMENT BUFFER ADDRESS
000378	4144	0001	00001		447	LA	RWC,1(RWC)	INCREMENT BUFFER COUNTER
00037C	47F0	C35A	0035A		448	B	CCWLOOP	DO NEXT BUFFER
000380	94BF	3004	00004		450	LASTCCW NI	4(RWB),X'FF'-CC	TURN OFF COMMAND CHAIN BIT
000384	5020	D0A0	000A0		451	ST	RWA,DSCBLIM	SAVE ADDRESS OF LAST DSCB BUFFER
					453	* SET OTHER THINGS AND START PROGRAM TO FILL BUFFER.		
000388	1B00				455	SR	R0,R0	
00038A	5000	D0A8	000A8		456	ST	R0,TTRN	SET RELATIVE TRACK NUMBER TO 0
00038E	948F	D0C4	000C4		457	NI	MODESW,X'FF'-XCPRUN-RDERR-EOWSW	SET FLAGS OFF
000392	4590	C480	00480		458	BAL	RRET,EXCP	START CHANNEL PROGRAM
000396	47F0	C04C	0004C		459	B	RETURN0	INDICATE SUCCESSFUL OPEN

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

      461 *****
      462 * MODE 2 - CLOSE *
      463 *****

00039A 4590 C3A2      003A2      465 CLOSE  BAL  RRET,CLOSESUB      CALL CLOSED CLOSE SUBROUTINE
00039E 47F0 C04C      0004C      466                B  RETURN0

      468 * IF THE CHANNEL PROGRAM IS RUNNING, WAIT FOR IT BEFORE TAKING FURTHER
      469 * ACTION.

0003A2      471 CLOSESUB DS  0H
0003A2 9140 D0C4      000C4      472                TM  MODESW,XCPRUN      TEST IF CHANNEL PROGRAM RUNNING
0003A6 4780 C3B8      003B8      473                BZ  NOEXCP          BRANCH IF NOT
      474                WAIT ECB=VTOCECB      WAIT UNTIL COMPLETE
0003AA 4110 D130      00130      475+              LA  1,VTOCECB          LOAD PARAMETER REG 1  01900002
0003AE 4100 0001      00001      476+              LA  0,1(0,0)          COUNT OMITTED,1 USED  00950002
0003B2 0A01      477+              SVC  1                LINK TO WAIT ROUTINE  01400002
0003B4 94BF D0C4      000C4      478                NI  MODESW,X'FF'-XCPRUN  TURN RUNNING SWITCH OFF
0003B8      479 NOEXCP  DS  0H

      481 * CLOSE THE DCB.

0003B8 9110 8030      00030      483                TM  DCBOFLGS,OPENBIT    TEST IF DCB OPEN
0003BC 4780 C3D2      003D2      484                BZ  NOCLOSE          BRANCH IF NOT
      485                CLOSE ((RDCB)),MF=(E,OPENLIST)  CLOSE THE VTOC
0003C0 4110 D0FC      000FC      486+              LA  1,OPENLIST        LOAD PARAMETER REG 1  01900002
0003C4 43E1 0000      00000      487+              IC  14,0(1,0)         SAVE OPTION BYTE      01140000
0003C8 5081 0000      00000      488+              ST  RDCB,0(1,0)       STORE DCB ADDR IN LIST 01320000
0003CC 42E1 0000      00000      489+              STC 14,0(1,0)         RESTORE OPTION BYTE   01340000
0003D0 0A14      490+              SVC  20                ISSUE CLOSE SVC      01640000
0003D2      491 NOCLOSE  DS  0H

      493 * FREE UP THE DDNAME AND VOLUME

0003D2 9108 D0C4      000C4      495                TM  MODESW,ALLOCSW     DID WE ALLOCATE A DEVICE
0003D6 47E0 C464      00464      496                BNO NOALLOC           NO - THEN NOTHING TO FREEUP
0003DA 4110 8028      00028      497                LA  R1,DCBDDNAM       POINT TO THE DDNAME
0003DE 5010 D0BC      000BC      498                ST  R1,DDNPDL         SAVE IT FOR FREE
0003E2 4110 0008      00008      499                LA  R1,8              GET THE DDNAME LENGTH
0003E6 4010 D0C0      000C0      500                STH R1,DDNPDL+4      SAVE IT FOR FREE
      501                FREE UNALC,DDN=DDNPDL,ERROR=S99FAIL  FREE THE DDNAME
0003EA      502+              DS  0H
0003EA 4110 D1D8      001D8      503+              LA  R1,DYN2SP1        LOAD ADDRESS OF PARAM LIST
      00000      504+              USING DYN2DS,R1       USE GENERATED DSECT
0003EE 41F0 1004      00004      505+              LA  R15,DYN2RB        LOAD ADDRESS OF S99 RB
      00000      506+              USING S99RB,R15
0003F2 50F1 0000      00000      507+              ST  R15,0(R1)        AND STORE IN RB POINTER
0003F6 D72D 1004 1004 00004 00004      508+              XC  4(DYN2LEN-4,R1),4(R1)  ZERO PARAMETER LIST
0003FC 9214 F000      00000      509+              MVI S99RBLN,20        MOVE IN LIST LENGTH
000400 9202 F001      00001      510+              MVI S99VERB,S99VRBUN  MOVE IN VERB CODE
000404 41E0 1018      00018      511+              LA  R14,DYN2TUP       LOAD ADDRESS OF TU POINTERS
000408 50E0 F008      00008      512+              ST  R14,S99TXTPP      STORE ADDRESS IN S99 RB
00040C 41F0 1020      00020      513+              LA  R15,DYN2TU        POINT TO SPACE FOR TEXT UNITS
      00000      514+              USING S99TUNIT,R15

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				516+	*****	
				517+ **	FREE DDNAME TEXT UNIT	**
				518+	*****	
000410	1802			519+	LR R0,R2	SAVE CONTENTS OF REGISTER 2
000412	58E0 D0BC	000BC		520+	L R14,DDNPDL	LOAD ADDRESS OF DDNAME
000416	4820 D0C0	000C0		521+	LH R2,DDNPDL+4	LOAD LENGTH OF DDNAME
00041A	4020 F004	00004		522+	STH R2,S99TULNG	STORE DDNAME LENGTH
00041E	0620			523+	BCTR R2,0	DECREMENT FOR EXECUTE
000420	4420 C44C	0044C		524+	EX R2,DYN2MVC	MOVE DDNAME
000424	9201 F001	00001		525+	MVI S99TUKEY+1,DUNDDNAM	MOVE IN DDNAME KEY
000428	9201 F003	00003		526+	MVI S99TUNUM+1,1	SET NUMBER FIELD
00042C	50F0 1018	00018		527+	ST R15,DYN2TUP+0	STORE TEXT UNIT ADDRESS
000430	41FF 000E	0000E		528+	LA R15,14(R15)	BUMP TEXT UNIT PTR TO NEXT SLOT
				530+	*****	
				531+ **	FREE EVEN IF PERMANENTLY ALLOCATED	**
				532+	*****	
000434	9207 F001	00001		533+	MVI S99TUKEY+1,DUNUNALC	SET TEXT UNIT KEY
000438	50F0 101C	0001C		534+	ST R15,DYN2TUP+4	STORE TEXT UNIT ADDRESS
00043C	41FF 0004	00004		535+	LA R15,4(R15)	BUMP TEXT UNIT PTR TO NEXT SLOT
000440	9280 101C	0001C		537+	MVI DYN2TUP+4,X'80'	SET HIGH ORDER BIT ON TEXT PTRS
000444	9280 1000	00000		538+	MVI DYN2RBP,X'80'	SET HIGH ORDER BIT ON RB PTR
000448	47F0 C452	00452		539+	B *+10	SKIP NEXT INSTRUCTION
00044C	D200 F006 E000	00006 00000		540+	DYN2MVC MVC S99TUPAR(0),0(R14)	EXECUTED MOVE
000452	1820			541+	LR R2,R0	RESTORE CONTENTS OF REGISTER 2
				542+	DROP R1,R15	DEACTIVATE ADDRESSABILITY
				543+*	MACDATE Y-2 73082	00300002
000454	0A63			544+	SVC 99 CALL DYNAMIC ALLOCATION	00400002
000456	41E0 D1DC	001DC		545+	LA R14,DYN2RBP+4	LOAD REG 14 WITH ADDRESS OF RB
			00004	546+	USING DYN2RBP,R14	SET UP ADDRESSABILITY
00045A	12FF			547+	LTR R15,R15	TEST RETURN CODE
00045C	4770 C4D0	004D0		548+	BNZ S99FAIL	BRANCH IF NON ZERO
				549+**	NOTE. R14 POINTS TO REQUEST BLOCK, R15 HAS RETURN CODE	**
000000				551+	DYN2DS DSECT	DSECT TO MAP SVC 99 DATA
000000				552+	DYN2RBP DS F	SVC 99 REQ BLOCK POINTER
000004				553+	DYN2RB DS 5F	SVC 99 REQUEST BLOCK
000018				554+	DYN2TUP DS CL8	SPACE FOR TEXT POINTERS
000020				555+	DYN2TU DS CL18	SPACE FOR TEXT UNITS
			00032	556+	DYN2LEN EQU *-DYN2RBP	LENGTH OF SPACE USED
000460				557+	VTOCEXCP CSECT	
000460	94F7 D0C4	000C4		559	NI MODESW,X'FF'-ALLOCSW	TURN OFF ALLOCATE SW
000464				561	NOALLOC DS 0H	
				563	* RELEASE CORE OBTAINED FOR DSCB BUFFERS.	
000464	9180 D0C4	000C4		565	TM MODESW,CBGOT	TEST IF CORE GOTTEN
000468	4780 C47A	0047A		566	BZ NOFREE	BRANCH IF NOT
00046C	9801 D090	00090		567	LM R0,R1,CBSIZE	LOAD SIZE AND LOCATION
				568	FREEMAIN R,LV=(0),A=(1)	FREE CORE

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				569+*	OS/VS2 RELEASE 3 VERSION -- 10/25/74	00001603
000470	4110 1000	00000		570+	LA 1,0(0,1)	CLEAR HI ORDER BYTE 00150802
000474	0A0A			571+	SVC 10	ISSUE FREEMAIN SVC 00311202
000476	947F D0C4	000C4		572	NI MODESW,X'FF'-CBGOT	SET CORE GOTTEN BIT OFF
00047A				573 NOFREE	DS 0H	
00047A	94DF D0C4	000C4		575	NI MODESW,X'FF'-RDERR	CLEAR ERROR SWITCH
00047E	07F9			576	BR RRET	

```

LOC  OBJECT CODE      ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25

                                           578 *****
                                           579 * EXCP ROUTINE *
                                           580 *****

                                           582 * CONVERT RELATIVE TRACK ADDRESS IN 'TTRN' TO ABSOLUTE SEEK ADDRESS IN
                                           583 * 'IOBSEEK', USING SUPERVISOR CONVERSION ROUTINE.

000480                                585 EXCP      DS      0H
000480 902D D048          00048      586          STM     R2,R13,EXCPSAVE      SAVE IMPORTANT REGISTERS
000484 4130 D048          00048      587          LA      R3,EXCPSAVE      SAVE REGS FOR RESTORING AFTER CL
000488 5800 D0A8          000A8      588          L       R0,TTRN          LOAD RELATIVE TRACK NUMBER
00048C 5810 802C          0002C      589          L       R1,DCBDEBAD      LOAD DEB ADDRESS
000490 4120 D128          00128      590          LA      R2,IOBSEEK      LOAD ADDR TO RECEIVE MBBCCHHR
000494 58F0 0010          00010      591          L       R15,CVT          LOAD CVT ADDRESS
000498 58FF 001C          0001C      592          L       R15,CVTPCNVT(R15)  LOAD ADDR OF CONVERT ROUTINE
00049C 05EF                                593          BALR   R14,R15          CONVERT TTRN TO MBBCCHHR
                                           594 *                                THAT CLOBBERED BASE REG
00049E 982D 3000          00000      595          LM     R2,R13,0(R3)      RESTORE REGISTERS
0004A2 12FF                                596          LTR    R15,R15          TEST IF EXTENT VIOLATED (RC=4)
0004A4 4770 C4C8          004C8      597          BNZ    SETEOF           IF SO, MEANS END-OF-FILE
0004A8 D504 B70D D12B 0070D 0012B 598          CLC    DS4HPCHR,IOBSEEK+3 CHECK FOR THE LAST FMT1
0004AE 4740 C4C8          004C8      599          BL     SETEOF           IF SO, PRETEND END-OF-FILE

                                           601 * ZERO ECB AND START CHANNEL PROGRAM.

0004B2 1B00                                603          SR     R0,R0
0004B4 5000 D130          00130      604          ST     R0,VTOCECB       CLEAR ECB
0004B8 94DF D0C4          000C4      605          NI     MODESW,X'FF'-RDERR  RESET ERROR SWITCH
                                           606          EXCP   VTOCIOB         START CHANNEL PROGRAM
0004BC 4110 D108          00108      607+        LA     1,VTOCIOB         LOAD PARAMETER REG 1    01900002
0004C0 0A00                                608+        SVC    0                ISSUE SVC FOR EXCP     00100000
0004C2 9640 D0C4          000C4      609          OI     MODESW,XCPRUN      SET 'RUNNING' FLAG
0004C6 07F9                                610          BR     RRET

                                           612 * WHEN EXTENT IS VIOLATED, SET END-FILE AND EXIT VIA CLOSE ROUTINE.

0004C8 9610 D0C4          000C4      614 SETEOF   OI     MODESW,EOFSW      SET END-OF-FILE BIT
0004CC 47F0 C3A2          003A2      615          B      CLOSESUB          EXIT VIA CLOSE SUBROUTINE

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				617	*****	
				618	* DAIRFAIL ROUTINE *	
				619	*****	
0004D0	18FA			620	S99FAIL LR R15,RRCODE SAVE THE RETURN CODE	
				621	S99FAIL MF=(E,S99FLIST,S99FLEN) ISSUE THE APPROPRIATE MSG	
0004D2				622+	DS 0H	
		00018		623+	S99FLEN EQU 24 LENGTH OF PARAMETER LIST	
0004D2	4110 D268	00268		624+	LA R1,S99FLIST LOAD DAIRFAIL PARAM LIST ADDRESS	
0004D6	50E1 0000	00000		625+	ST R14,0(R1) STORE S99 RB ADDRESS	
0004DA	41E1 0014	00014		626+	LA R14,20(R1) LOAD ADDR RET CODE FLD	
0004DE	50FE 0000	00000		627+	ST R15,0(R14) STORE RET CODE	
0004E2	50E1 0004	00004		628+	ST R14,4(R1) AND STORE ITS ADDRESS	
0004E6	41E0 C74C	0074C		629+	LA R14,=A(0) LOAD ADDR OF FULLWORD OF 0	
0004EA	50E1 0008	00008		630+	ST R14,8(R1) STORE IT.	
0004EE	41E0 C752	00752		631+	LA R14,=X'8032'	
0004F2	50E1 000C	0000C		632+	ST R14,12(R1) AND STORE IT	
0004F6	D703 1010 1010	00010 00010		633+	XC 16(4,R1),16(R1) CLEAR CPPL POINTER	
0004FC				634+	CNOP 0,4	04900000
0004FC	45F0 C510	00510		635+	BAL 15,*+20 BRANCH AROUND CONSTANTS	04950000
000500	00000508			636+	DC A(*+8) ADDR. OF PARM. LIST	05050000
000504	00000000			637+	DC A(0) DCB ADDRESS PARAMETER	06650000
000508	C9D2D1C5C6C6F1F8			638+	DC CL8'IKJEFF18' EP PARAMETER	06750000
000510	0A06			639+	SVC 6 ISSUE LINK SVC	48000000
000512	18FA			640	LR R15,RRCODE RELOAD THE RETURN CODE	
000514	47F0 C04E	0004E		641	B RETURN AND THEN EXIT	
				643	*	
				644	* VARIOUS OTHER ERROR ROUTINES	
				645	*	
				646	OPENERR VTOCMSG OPENERRM ISSUE THE MESSAGE	
000518	4110 C594	00594		647+	OPENERR LA R1,OPENERRM POINT TO THE FIRST MESSAGE	
00051C	1B00			648+	SR R0,R0 NO SECOND LEVEL MESSAGE	
00051E	9001 B078	00078		649+	STM R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				650+	* THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
000522	4110 B000	00000		651+	LA R1,VTOCOM POINT TO THE COMMON AREA	
000526	58F0 B01C	0001C		652+	L R15,VADMSG POINT TO THE ROUTINE	
00052A	05EF			653+	BALR R14,R15 THEN CALL IT	
00052C	47F0 C58C	0058C		654	B ERRET THEN RETURN	
				655	OBTERR VTOCMSG OBTERRM ISSUE THE MESSAGE	
000530	4110 C5BA	005BA		656+	OBTERR LA R1,OBTERRM POINT TO THE FIRST MESSAGE	
000534	1B00			657+	SR R0,R0 NO SECOND LEVEL MESSAGE	
000536	9001 B078	00078		658+	STM R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				659+	* THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
00053A	4110 B000	00000		660+	LA R1,VTOCOM POINT TO THE COMMON AREA	
00053E	58F0 B01C	0001C		661+	L R15,VADMSG POINT TO THE ROUTINE	
000542	05EF			662+	BALR R14,R15 THEN CALL IT	
000544	47F0 C58C	0058C		663	B ERRET THEN RETURN	
				664	NOTFMT4 VTOCMSG NOTFMT4M ISSUE THE MESSAGE	
000548	4110 C5DA	005DA		665+	NOTFMT4 LA R1,NOTFMT4M POINT TO THE FIRST MESSAGE	
00054C	1B00			666+	SR R0,R0 NO SECOND LEVEL MESSAGE	
00054E	9001 B078	00078		667+	STM R0,R1,MSGADDRS SAVE THE MESSAGE ADDRESSES	
				668+	* THEN JUST CALL THE MESSAGE ISSUING ROUTINE	
000552	4110 B000	00000		669+	LA R1,VTOCOM POINT TO THE COMMON AREA	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000556	58F0 B01C	0001C		670+	L	R15,VADMSG	POINT TO THE ROUTINE
00055A	05EF			671+	BALR	R14,R15	THEN CALL IT
00055C	47F0 C58C	0058C		672	B	ERRET	THEN RETURN
				673	DSCBNUM0	VTOCMSG DSCBNUMM	ISSUE THE MESSAGE
000560	4110 C606	00606		674+	DSCBNUM0	LA R1,DSCBNUMM	POINT TO THE FIRST MESSAGE
000564	1B00			675+	SR	R0,R0	NO SECOND LEVEL MESSAGE
000566	9001 B078	00078		676+	STM	R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES
				677+*			THEN JUST CALL THE MESSAGE ISSUING ROUTINE
00056A	4110 B000	00000		678+	LA	R1,VTOCOM	POINT TO THE COMMON AREA
00056E	58F0 B01C	0001C		679+	L	R15,VADMSG	POINT TO THE ROUTINE
000572	05EF			680+	BALR	R14,R15	THEN CALL IT
000574	47F0 C58C	0058C		681	B	ERRET	THEN RETURN
				682 *			
				683	ERRJFCB	VTOCMSG ERRJFCBM	ERROR IN READING JFCB
000578	4110 C66E	0066E		684+	ERRJFCB	LA R1,ERRJFCBM	POINT TO THE FIRST MESSAGE
00057C	1B00			685+	SR	R0,R0	NO SECOND LEVEL MESSAGE
00057E	9001 B078	00078		686+	STM	R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES
				687+*			THEN JUST CALL THE MESSAGE ISSUING ROUTINE
000582	4110 B000	00000		688+	LA	R1,VTOCOM	POINT TO THE COMMON AREA
000586	58F0 B01C	0001C		689+	L	R15,VADMSG	POINT TO THE ROUTINE
00058A	05EF			690+	BALR	R14,R15	THEN CALL IT
				691 *			
00058C	41F0 0008	00008		692	ERRET	LA R15,8	SET AN ERROR RETURN CODE
000590	47F0 C04E	0004E		693	B	RETURN	THEN EXIT
				694 *			

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				696	*****	
				697	* CONSTANTS, VARIABLES, ETC... *	
				698	*****	
				700	* ERROR MESSAGES	
				701	*	
				702	OPENERM MSG ' VTOCEXCP - ERROR IN OPENING VTOC '	
000594	0026000040E5E3D6			703+	OPENERM DC H'38',H'0',C' VTOCEXCP - ERROR IN OPENING VTOC '	
				704	OBTERM MSG ' VTOCEXCP - ERROR IN OBTAIN '	
0005BA	0020000040E5E3D6			705+	OBTERM DC H'32',H'0',C' VTOCEXCP - ERROR IN OBTAIN '	
				706	NOTFMT4M MSG ' VTOCEXCP - FORMAT 4 DSCB WAS NOT FIRST'	
0005DA	002B000040E5E3D6			707+	NOTFMT4M DC H'43',H'0',C' VTOCEXCP - FORMAT 4 DSCB WAS NOT FIRST'	
				708	DSCBNUMM MSG ' VTOCEXCP - THE FORMAT 4 DSCB HAS DSCB S/TRK = 0 '	
000605	00					
000606	0035000040E5E3D6			709+	DSCBNUMM DC H'53',H'0',C' VTOCEXCP - THE FORMAT 4 DSCB HAS DSCB S/TRX	
00060E	C3C5E7C3D7406040				+ K = 0 '	
				710	TRACKERR MSG ' VTOCEXCP - A READ ERROR OCCURRED ON THE VTOC '	
00063B	00					
00063C	0032000040E5E3D6			711+	TRACKERR DC H'50',H'0',C' VTOCEXCP - A READ ERROR OCCURRED ON THE VTX	
000644	C3C5E7C3D7406040				+ OC '	
				712	ERRJFCBM MSG ' VTOCEXCP - A RDJFCB ERROR OCCURRED '	
00066E	0028000040E5E3D6			713+	ERRJFCBM DC H'40',H'0',C' VTOCEXCP - A RDJFCB ERROR OCCURRED '	
000696	0000					
000698	1600000070000008			715	INITCCW CCW READR0,0,CC+SLI+SKIP,8	
0006A0	1E00000040000094			716	READCCW CCW READCKD,0,CC,148	
				718	DSCBCON CAMLST SEEK,0,0,0 FILLED IN WITH IOBSEEK+3, VOLID, FMT4	
0006A8				719+	DSCBCON DS 0F ALIGN ON FULL WORD 00349401	
0006A8	C0			720+	DC AL1(192) THREE BYTES OF FLAGS 00349501	
0006A9	80			721+	DC AL1(128) INDICATING THE FUNC- 00349601	
0006AA	00			722+	DC AL1(0) TION TO BE PERFORMED 00399601	
0006AB	00			723+	DC AL1(0) NO OPTION THREE 00419601	
0006AC	00000000			724+	DC A(0) PARAMETER TWO 00441601	
0006B0	00000000			725+	DC A(0) PARAMETER THREE 00448001	
0006B4	00000000			726+	DC A(0) PARAMETER FOUR 00448801	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	
				728	* DATA CONTROL BLOCK	
				729	PRINT GEN	
				730	VTOCDCBM DCB DDNAME=VTOCDD,MACRF=(E),EXLST=1	
				732+*	DATA CONTROL BLOCK	22770020
				733+*		22860020
0006B8				734+	VTOCDCBM DS 0F'0' ORIGIN ON WORD BOUNDARY	23040020
0006B8		006BC		735+	ORG *+4 TO ORIGIN GENERATION	23130020
				737+*	DIRECT ACCESS DEVICE INTERFACE	27360020
0006BC	0000000000000000			739+	DC BL12'0' FDAD,DVTBL	27540020
0006C8	00000000			740+	DC A(0) KEYLE,DEVT,TRBAL	27720020
				742+*	COMMON ACCESS METHOD INTERFACE	48690020
0006CC	00			744+	DC AL1(0) BUFNO	49050020
0006CD	000001			745+	DC AL3(1) BUFCB	54720020
0006D0	0000			746+	DC AL2(0) BUFL	55170020
0006D2	0000			747+	DC BL2'0000000000000000'	*55800020
				+	DSORG	55890020
0006D4	00000001			748+	DC A(1) IOBAD	56340020
				750+*	FOUNDATION EXTENSION	56610020
0006D8	00			752+	DC BL1'00000000' BFTEK,BFLN,HIARCHY	59850020
0006D9	000001			753+	DC AL3(1) EODAD	65970020
0006DC	00			754+	DC BL1'00000000'	*66150020
				+	RECFM	66240020
0006DD	000001			755+	DC AL3(1) EXLST	66330020
				757+*	FOUNDATION BLOCK	66690020
0006E0	E5E3D6C3C4C44040			759+	DC CL8'VTOCDD' DDNAME	66870020
0006E8	02			760+	DC BL1'00000010' OFLGS	68220020
0006E9	00			761+	DC BL1'00000000' IFLG	68310020
0006EA	D004			762+	DC BL2'1101000000000100'	*68400020
				+		*68490020
				+	MACR	68580020
		00034		763	DCBLEN EQU *-VTOCDCBM	
				765	* IOB FOR CHANNEL PROGRAM	
0006F0				767	IOBCONST DS 0D	
0006F0	42000000			768	DC X'42000000' COMMAND CHAIN, NOT RELATED	
0006F4	00000000			769	DC A(0) ECB ADDRESS	
0006F8	0000000000000000			770	DC 2F'0'	
000700	00000000			771	DC A(0) CHANNEL PROGRAM BEGINNING	
000704	00000000			772	DC A(0) DCB ADDRESS	
000708	03000000			773	DC X'03000000'	
00070C	00000000			774	DC F'0'	
000710	0000000000000000			775	DC D'0' INITIAL SEEK ADDRESS	
		00028		776	IOBCONL EQU *-IOBCONST	
				777	* VTOC NAME FOR ALLOCATION	

```
LOC  OBJECT CODE      ADDR1 ADDR2  STMT  SOURCE STATEMENT                                ASM 0201 13.38 01/07/25
000718 0000071E          778 VTOCNM  DC    A(VTOCNAME)
00071C 000C              779          DC    Y(12)
00071E C6D6D9D4C1E3F44B  780 VTOCNAME DC    CL12'FORMAT4.DSCB'  DATA SET NAME FOR VTOC
              781 *
              782 *
000730              783          LTORG
000730 4040404040404040  784          =CL8' '
000738 0004000100010800  785          =Y(DALSTATS,1,1,X'0800')
000740 00010000          786          =X'00010000'
000744 00000100          787          =X'00000100'
000748 FFFFFFFF8          788          =X'FFFFFFF8'
00074C 00000000          789          =A(0)
000750 000F              790          =H'15'
000752 8032              791          =X'8032'
```

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM 0201 13.38 01/07/25

793 * SECTION DEFINITION AND REGISTER ASSIGNMENTS;

00002	795	RWA	EQU	2	
00003	796	RWB	EQU	3	
00004	797	RWC	EQU	4	
00008	798	RDCB	EQU	8	DCB POINTER
0000A	799	RRCODE	EQU	10	RETURN CODE REGISTER
00009	800	RRET	EQU	9	LOCAL SUBROUTINE EXIT REGISTER

802 * TAGS FOR CHANNEL COMMANDS AND FLAG BITS:

00016	804	READR0	EQU	X'16'	READ RECORD 0
0001E	805	READCKD	EQU	X'1E'	READ COUNT, KEY, AND DATA
00040	807	CC	EQU	X'40'	COMMAND CHAIN FLAG
00020	808	SLI	EQU	X'20'	SUPPRESS LENGTH INDICATION FLAG
00010	809	SKIP	EQU	X'10'	SKIP DATA TRANSFER FLAG

811 * COMMUNICATION VECTOR TABLE (CVT) DEFINITIONS:

00010	813	CVT	EQU	16	LOCATION OF CVT BASE ADDRESS
0001C	814	CVTPCNVT	EQU	28	OFFSET TO CONVERT ROUTINE ADDRESS

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	
				816 *		
				817 *	AREA USED BY VTOCREAD, PASSED VIA R13	
				818 *		
000000				819	VTOCWORK DSECT	
000000				820	DS 18F	SAVE AREA
000048				822	EXCPSAVE DS 18F	INTERNAL SAVE AREA
000090				823	CBSIZE DS 2F	SIZE AND LOCATION OF GOTTEN CORE
		00094		824	CBADDR EQU CBSIZE+4	
000098				825	NDSCBS DS F	NUMBER OF DSCB'S PER TRACK
00009C				826	DSCBSTRT DS F	ADDRESS OF 1ST DSCB BUFFER
0000A0				827	DSCBLIM DS F	ADDRESS OF LAST DSCB BUFFER
0000A4				828	DSCBADR DS F	ADDRESS OF CURRENT DSCB
0000A8				829	TTRN DS F	RELATIVE TRACK NUMBER
0000AC				830	VOLADDR DS A	FAKE PDL FOR ALLOC MACRO - ADDRESS
0000B0				831	VOLLEN DS H	AND LENGTH OF VOLID
0000B4				832	UNITADDR DS A	FAKE PDL FOR ALLOC MACRO - ADDRESS
0000B8				833	UNITLEN DS H	AND LENGTH OF UNIT ADDRESS
0000BC				834	DDNPDL DS 2F	SPACE FOR DDNAME PDL
				836 *	MODE SWITCH AND BIT DEFINITIONS	
0000C4	00			838	MODESW DC X'00'	
		00080		839	CBGOT EQU X'80'	CORE GOTTEN FOR BUFFER
		00040		840	XCPRUN EQU X'40'	CHANNEL PROGRAM STARTED BUT NOT CHECKED
		00020		841	RDERR EQU X'20'	PERMANENT I/O ERROR
		00010		842	EOFSW EQU X'10'	END-OF-FILE SENSED
		00008		843	ALLOCSW EQU X'08'	ALLOCATE VOLUME FLAG
				845	VTOCDCB DCB DDNAME=VTOCDD,MACRF=(E),EXLST=1	
				847+*		DATA CONTROL BLOCK 22770020
				848+*		22860020
0000C8				849+VTOCDCB	DS 0F'0'	ORIGIN ON WORD BOUNDARY 23040020
0000C8		000CC		850+	ORG *+4	TO ORIGIN GENERATION 23130020
				852+*		DIRECT ACCESS DEVICE INTERFACE 27360020
0000CC	000000000000000000			854+	DC BL12'0'	FDAD,DVTBL 27540020
0000D8	00000000			855+	DC A(0)	KEYLE,DEV,TRBAL 27720020
				857+*		COMMON ACCESS METHOD INTERFACE 48690020
0000DC	00			859+	DC AL1(0)	BUFNO 49050020
0000DD	000001			860+	DC AL3(1)	BUFCB 54720020
0000E0	0000			861+	DC AL2(0)	BUFL 55170020
0000E2	0000			862+	DC BL2'0000000000000000'	*55800020
				+		DSORG 55890020
0000E4	00000001			863+	DC A(1)	IOBAD 56340020
				865+*		FOUNDATION EXTENSION 56610020
0000E8	00			867+	DC BL1'00000000'	BFTEK,BFLN,HIARCHY 59850020
0000E9	000001			868+	DC AL3(1)	EODAD 65970020

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
0000EC	00			869+	DC BL1'00000000'	*66150020
				+		66240020
0000ED	000001			870+	DC AL3(1)	66330020
				872+*	FOUNDATION BLOCK	66690020
0000F0	E5E3D6C3C4C44040			874+	DC CL8'VTOCDD'	66870020
0000F8	02			875+	DC BL1'00000010'	68220020
0000F9	00			876+	DC BL1'00000000'	68310020
0000FA	D004			877+	DC BL2'1101000000000100'	*68400020
				+		*68490020
				+	MACR	68580020
0000FC		00010		879	OPENBIT EQU X'10'	
				880	OPENLIST DS 2F	
				882	* IOB FOR CHANNEL PROGRAM	
000108				884	VTOCIOB DS 0D	
000108	42000000			885	IOBFLAG1 DC X'42000000'	COMMAND CHAIN, NOT RELATED
00010C	00000130			886	IOBECB DC A(VTOCECB)	
000110	0000000000000000			887	DC 2F'0'	
000118	00000000			888	IOBSTART DC A(0)	CHANNEL PROGRAM BEGINNING
00011C	000000C8			889	IOBDCB DC A(VTOCDCB)	
000120	03000000			890	DC X'03000000'	
000124	00000000			891	DC F'0'	
000128	0000000000000000			892	IOBSEEK DC D'0'	INITIAL SEEK ADDRESS
				894	* EVENT CONTROL BLOCK FOR CHANNEL PROGRAM:	
000130	00000000			896	VTOCECB DC F'0'	EVENT CONTROL BLOCK
				898	* INTERNAL BUFFER FOR LAST DSCB	
000134				899	BUFF DS XL148	
				901	DSCBFMT4 CAMLST SEEK, IOBSEEK+3, VOLID, FMT4	
0001C8				902+	DSCBFMT4 DS 0F	ALIGN ON FULL WORD 00349401
0001C8	C0			903+	DC AL1(192)	THREE BYTES OF FLAGS 00349501
0001C9	80			904+	DC AL1(128)	INDICATING THE FUNC- 00349601
0001CA	00			905+	DC AL1(0)	TION TO BE PERFORMED 00399601
0001CB	00			906+	DC AL1(0)	NO OPTION THREE 00419601
0001CC	0000012B			907+	DC A(IOBSEEK+3)	PARAMETER TWO 00441601
0001D0	000001B9			908+	DC A(VOLID)	PARAMETER THREE 00448001
0001D4	000006E0			909+	DC A(FMT4)	PARAMETER FOUR 00448801
				911	* WORK AREA FOR DYNAMIC ALLOCATION	
				912	DYNSPACE	
				913+*		
				914+**	RESERVE SPACE FOR ALLOC/FREE MACRO WORK AREA	
				915+*		
0001D8				916+	DYNSP1 DS 0F, CL144	RESERVE SPACE
000268				917	S99FLIST DS XL(S99FLEN)	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				919 *			
				920 *	JFCB	EXIT LIST AND AREA	
				921 *			
000280				922	JEXLST DS	F	
000284				923	JFCBAREA DS	XL176	
000338				924	DS	0D	
		00338		925	VTOCWLEN EQU	*-VTOCWORK	
				927		PRINT GEN	
				928		VTOCOM	
				929+*			
				930+*		THIS IS THE VTOC COMMAND COMMON AREA	
				931+*			
000000				932+VTOCOM	DSECT		
				933+*			
				934+*		WORKING STORAGE AREAS FOR THE VARIOUS ROUTINES	
				935+*			
000000				936+VTCWMSG	DS A	WORKING STORAGE FOR THE MSG ROUTINE	
000004				937+VTCWEXIT	DS A	WORKING STORAGE FOR THE EXIT ROUTINE	
000008				938+VTCWEXCP	DS A	WORKING STORAGE FOR THE EXCP ROUTINE	
00000C				939+VTCWCHEK	DS A	WORKING STORAGE FOR THE CHEK ROUTINE	
000010				940+VTCWFORM	DS A	WORKING STORAGE FOR THE FORM ROUTINE	
000014				941+VTCWPRNT	DS A	WORKING STORAGE FOR THE PRNT ROUTINE	
000018				942+VTCWSORT	DS A	WORKING STORAGE FOR THE SORT ROUTINE	
				943+*			
				944+*		ADDRESSES OF THE ROUTINES	
				945+*			
00001C	00000000			946+VADMSG	DC	V(VTOCMSG) ADDRESS OF THE MESSAGE ROUTINE	
000020	00000000			947+VADEXIT	DC	V(VTOCEXIT) ADDRESS OF THE EXIT ROUTINE	
000024	00000000			948+VADEXCP	DC	V(VTOCEXIT) ADDRESS OF THE EXCP ROUTINE	
000028	00000000			949+VADCHEK	DC	V(VTOCEXIT) ADDRESS OF THE CHECK ROUTINE	
00002C	00000000			950+VADFORM	DC	V(VTOCEXIT) ADDRESS OF THE FORMAT ROUTINE	
000030	00000000			951+VADPRNT	DC	V(VTOCEXIT) ADDRESS OF THE PRINT ROUTINE	
000034	00000000			952+VADSORT	DC	V(VTOCEXIT) ADDRESS OF THE SORT ROUTINE	
				953+*			
				954+*		TSO COMMAND PROCESSOR AND PARSE DATA	
				955+*			
000038				956+ADDRUPT	DS A	USER PROFILE TABLE	
00003C				957+ADDRECT	DS A	ENVIRONMENT CONTROL TABLE	
000040				958+ADDRPSCB	DS A	PROTECTED STEP CONTROL BLOCK	
000044				959+ADDRCBUF	DS A	COMMAND BUFFER	
000048				961+ADDRANSR	DS A	PARSE ANSWER OR PDL ADDRESS	
00004C				963+PARMLIST	DS 8A	INTERNAL PARM AREA (MSG)	
00006C				964+ATTNECB	DS F	ECB FOR ATTENTIONS	
000070				965+DOUBLE	DS D		
000078				967+MSGADDRS	DS 2A	ADDRESSES OF MESSAGES FOR VTOCMSG	
000080				968+MSGTEXT1	DS XL124		
0000FC				969+MSGTEXT2	DS XL124		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000178				971+	PUTOLD1 DS	3F	
000184				972+	PUTOLD2 DS	3F	
				973+*			
				974+*		PARAMETER LIST FOR THE EXIT ROUTINE	
				975+*			
000190				976+	EXITLIST DS	0F	
000190				977+	EXITAREA DS	A	WORKAREA LOCATION
000194				978+	DSCBADDR DS	A	ADDRESS OF THE DSCB
000198				979+	FORMATAD DS	A	ADDRESS OF THE FORMATTED DSCB
00019C				980+	CPPLADDR DS	A	ADDRESS OF THE CPPL
0001A0				981+	ACTIONAD DS	A	ADDRESS OF THE RECOMMENDED OR REQUESTED ACTION
				982+*			
				983+*		INTER ROUTINE FLAGS	
				984+*			
0001A4				985+	VTCEFUNC DS	X	VTOCEXCP FUNCTION FLAG
0001A5				986+	VTCFMTCK DS	X	FORMAT IS CALLED BY CHECK RTN
		00080		987+	VTCFMTCD EQU	X'80'	FORMAT WAS CALLED BY CHECK
		00008		988+	VTCFMTCC EQU	X'08'	FORMAT WAS CALLED BY CHECK THIS CALL
				989+*			
0001A6				990+	TABFULL DS	X	FLAG TABLES FULL, STOP INPUT
0001A7				991+	LOCAT DS	X	FLAG TO PERFORM CATALOG LOCATE
0001A8				992+	VTCEPRNT DS	X	PRINT END AND CLEANUP FLAG
0001AA				993+	DSNLEN DS	H	LENGTH OF THE DSNAME (NON-BLANKS)
0001AC				994+	ATABTITL DS	A	ADDRESS OF TABLE OF TITLES, LENGTHS
				995+*			
				996+*			
				997+*			
				998+*		WORKING STORAGE FOR VOLUME UCB SEARCH	
				999+*			
0001B0				1000+	ADDR DS	CL3	UCB ADDRESS IN CHARACTERS
0001B3				1001+	VOLSER DS	CL6	VOLUME SERIAL NUMBER FROM PARSE
0001B9				1002+	VOLID DS	CL6	CURRENT VOLUME SERIAL NUMBER TO PROCESS
0001BF				1003+	FLAG DS	X	UCB SEARCH FLAG
0001C0				1004+	LASTADR DS	F	LAST UCB ADDRESS FOUND (NO DUP'S)
0001C4				1005+	UCBDEVT DS	CL4	PRINTABLE FORM OF DEVICE TYPE
				1006+*			RPRINS
				1007+*			
				1008+*			
0001C8				1009+	SORTTAB DS	16F	
				1010+*			
				1011+*		EACH ENTRY CONTAINS A KEY OFFSET (2 BYTES) AND A KEY LENGTH (2 BYTES)	
				1012+*		THIS TABLE IS BUILT AT PARSE TIME ACCORDING TO THE SORT PARAMETERS	
				1013+*		SPECIFIED. THE 1ST PARM IS THE HIGH KEY AND SO ON.	
				1014+*			
				1015+*			
				1016+*			
				1017+*		ADDRESSES OF GETMAIN FOR FORMATTED DATA	
				1018+*			
000208				1019+	VTCCURAD DS	A	CURRENT AVAILABLE ADDRESS
00020C				1020+	VTCCURLN DS	A	CURRENT AVAILABLE LENGTH
000210				1021+	VTCGETMN DS	50A	ADDRESSES OF BLOCKS
		00032		1022+	VTCGETMX EQU	(*-VTCGETMN)/4	NUMBER OF BLOCKS MAXIMUM
		08000		1023+	VTCGETMS EQU	32768	GETMAIN SIZE
				1024+*			
				1025+*		HASH SORT TABLE, POINTERS TO FIRST ENTRIES	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				1026+*			
0002D8				1027+VTCSORTH	DS	256A	POINT TO FORMATED ENTRIES
		006D8		1028+VTCSORTE	EQU	*	END OF LIST
				1029+*			
				1030+*		PRINT ENTRIES - PAGE AND LINE COUNTERS	
				1031+*			
0006D8				1032+LINECT	DS	H	LINE COUNT
0006DA				1033+LINEMAX	DS	H	MAXIMUM LINES PER PAGE
0006DC				1034+PAGECT	DS	H	PAGE COUNT
0006DE				1035+LINELEN	DS	H	LENGTH OF THE PRINT LINE
				1036+*			
				1037+*			
				1038+*		VARIOUS ITEMS	
				1039+*			
0006E0				1040+FMT4	DS	XL44	SPACE FOR DSCB NAME
		0070C		1041+IECSDSL4	EQU	*	FORMAT 4 DSCB
		0070C		1042+IECSDSF4	EQU	IECSDSL4	16000000
00070C				1043+DS4IDFMT	DS	CL1	FORMAT IDENTIFIER
00070D				1044+DS4HPCHR	DS	XL5	HIGHEST ADDRESS OF A FORMAT 1 DSCB
000712				1045+DS4DSREC	DS	XL2	NUMBER OF AVAILABLE DSCB'S
000714				1046+DS4HCCHH	DS	XL4	CCHH OF NEXT AVAILABLE ALTERNATE TRK
000718				1047+DS4NOATK	DS	XL2	NUMBER OF REMAINING ALTERNATE TRACKS
00071A				1048+DS4VTOCI	DS	XL1	VTOC INDICATORS
		00080		1049+DS4DOSBT	EQU	X'80'	DOS BIT
		00010		1050+DS4DSTKP	EQU	X'10'	DOS STACKED PACK
		00008		1051+DS4DOCVT	EQU	X'08'	DOS CONVERTED VTOC
		00004		1052+DS4DIRF	EQU	X'04'	DIRF BIT
		00002		1053+DS4DICVT	EQU	X'02'	DIRF RECLAIMED
		00001		1054+DS4IVTOC	EQU	X'01'	VOLUME USES AN INDEXED VTOC @01A
00071B				1055+DS4NOEXT	DS	XL1	NUMBER OF EXTENTS IN THE VTOC
00071C				1056+	DS	XL2	RESERVED
00071E				1057+DS4DEVCT	DS	0XL14	DEVICE CONSTANTS
00071E				1058+DS4DEVSZ	DS	XL4	DEVICE SIZE
000722				1059+DS4DEVTK	DS	XL2	DEVICE TRACK LENGTH
000724				1060+DS4DEVOV	DS	0XL2	KEYED RECORD OVERHEAD
000724				1061+DS4DEVI	DS	XL1	NON-LAST KEYED RECORD OVERHEAD
000725				1062+DS4DEVL	DS	XL1	LAST KEYED RECORD OVERHEAD
000726				1063+DS4DEVK	DS	XL1	NON-KEYED RECORD OVERHEAD
				1064+*			DIFFERENTIAL
000727				1065+DS4DEVFG	DS	XL1	FLAG BYTE
000728				1066+DS4DEVTL	DS	XL2	DEVICE TOLERANCE
00072A				1067+DS4DEVDT	DS	XL1	NUMBER OF DSCB'S PER TRACK
00072B				1068+DS4DEVDB	DS	XL1	NUMBER OF DIRECTORY BLOCKS PER TRACK
00072C				1069+DS4AMTIM	DS	XL8	VSAM TIME STAMP
000734				1070+DS4AMCAT	DS	0XL3	VSAM CATALOG INDICATOR
000734				1071+DS4VSIND	DS	XL1	VSAM INDICATORS
000735				1072+DS4VSCRA	DS	XL2	RELATIVE TRACK LOCATION OF THE CRA
000737				1073+DS4R2TIM	DS	XL8	VSAM VOLUME/CATALOG MATCH
				1074+*			TIME STAMP
00073F				1075+	DS	XL5	RESERVED
000744				1076+DS4F6PTR	DS	XL5	POINTER TO FIRST FORMAT 6 DSCB
000749				1077+DS4VTOCE	DS	XL10	VTOC EXTENT DESCRIPTION
000753				1078+	DS	XL25	RESERVED
		0076C		1079+DS4END	EQU	*	17850000
000770				1080+	DS	0D	17900000

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
000770				1081+FMT3	DS 0XL148	SPACE FOR FORMAT3 DSCB
			00770	1082+IECSDSL3	EQU *	FORMAT 3 DSCB 15250000
			00770	1083+IECSDSF3	EQU IECSDSL3	15300000
000770				1084+	DS XL4	KEY IDENTIFIER 15350000
000774				1085+DS3EXTNT	DS XL40	FOUR EXTENT DESCRIPTIONS 15400000
				1086+*	FIRST BYTE	EXTENT TYPE INDICATOR 15450000
				1087+*	SECOND BYTE	EXTENT SEQUENCE NUMBER 15500000
				1088+*	THIRD - SIXTH BYTES	LOWER LIMIT 15550000
				1089+*	SEVENTH - TENTH BYTES	UPPER LIMIT 15600000
00079C				1090+DS3FMTID	DS CL1	FORMAT IDENTIFIER 15650000
00079D				1091+DS3ADEXT	DS XL90	NINE ADDITIONAL EXTENT DESCRIPTIONS 15700000
0007F7				1092+DS3PTRDS	DS XL5	RESERVED 15750000
		007FC		1093+DS3END	EQU *	15800000
000800				1094+	DS 0D	
				1095	PRINT NOGEN	
				1097	IEFZB4D0	
				1237	IEFZB4D2	
				1434	DCBD DEVD=DA,DSORG=PS	
000000				1993 JFCB	DSECT	
				1994	IEFJFCBN	
				2019+	PRINT OFF	00803202
				2606	END	

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000500
0001	0001	0C	000718

ASM 0201 13.38 01/07/25

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
ADDR	00003	000001B0	01000	00253
ALLOCSW	00001	00000008	00843	00360 00495 00559
BUFF	00148	00000134	00899	00166 00171
CBADDR	00004	00000094	00824	00423
CBGOT	00001	00000080	00839	00424 00565 00572
CBSIZE	00004	00000090	00823	00419 00567 00824
CC	00001	00000040	00807	00450 00715 00716
CCWLOOP	00006	0000035A	00440	00448
CLOSE	00004	0000039A	00465	00100
CLOSESUB	00002	000003A2	00471	00221 00465 00615
CVT	00001	00000010	00813	00397 00591
CVTPCNVT	00001	0000001C	00814	00398 00592
DALDSNAM	00001	00000002	01275	00285
DALRTDDN	00001	00000055	01358	00293
DALSTATS	00001	00000004	01277	00785
DALUNIT	00001	00000015	01294	00310
DALVLSER	00001	00000010	01289	00325
DCBBIT0	00001	00000080	01456	01542 01550 01562 01585 01612 01614 01615 01617 01640 01643 01663 01667 01682 01719 01774 01805 01844 01848 01861 01961 01963 01973
DCBBIT1	00001	00000040	01457	01543 01551 01564 01586 01587 01596 01612 01614 01616 01617 01645 01663 01665 01667 01685 01686 01687 01722 01723 01774 01807 01850 01852 01864 01908 01961 01965 01974
DCBBIT2	00001	00000020	01458	01544 01552 01565 01566 01567 01586 01587 01591 01597 01612 01613 01618 01647 01668 01669 01690 01691 01692 01726 01727 01775 01812 01853 01869 01911 01914 01961 01975
DCBBIT3	00001	00000010	01459	01545 01565 01567 01568 01586 01599 01619 01650 01668 01671 01694 01695 01696 01730 01731 01775 01814 01817 01819 01855 01870 01911 01915 01961
DCBBIT4	00001	00000008	01460	01553 01600 01620 01651 01673 01678 01679 01699 01700 01734 01735 01737 01738 01776 01822 01871 01911 01916
DCBBIT5	00001	00000004	01461	01554 01601 01623 01624 01653 01673 01675 01676 01679 01703 01705 01706 01707 01741 01742 01743 01744 01776 01824 01827 01857 01873 01906
DCBBIT6	00001	00000002	01462	01546 01602 01603 01606 01623 01625 01654 01710 01711 01712 01713 01747 01748 01749 01750 01777 01830 01875 01917
DCBBIT7	00001	00000001	01463	01547 01602 01604 01606 01627 01658 01715 01716 01753 01754 01756 01757 01833 01859 01876 01919
DCBDDNAM	00008	00000028	01637	00350 00497
DCBDEBAD	00004	0000002C	01772	00395 00589
DCBEXLSA	00003	00000025	01629	00232
DCBFDAD	00008	00000005	01483	01486
DCBIFLGS	00001	0000002C	01773	00192
DCBLEN	00001	00000034	00763	00227
DCBOFLGS	00001	00000030	01639	00388 00483
DDNPDL	00004	000000BC	00834	00498 00500 00520 00521
DSCBADDR	00004	00000194	00978	00157
DSCBADR	00004	000000A4	00828	00151 00153
DSCBCON	00004	000006A8	00719	00243
DSCBELUP	00006	000000F0	00196	00202
DSCBFMT4	00004	000001C8	00902	00243 00245 00247 00249 00402
DSCBLIM	00004	000000A0	00827	00154 00201 00451
DSCBNUMM	00002	00000606	00709	00674
DSCBNUM0	00004	00000560	00674	00411
DSCBSTRT	00004	0000009C	00826	00184 00193 00433
DS4DEVDT	00001	0000072A	01067	00408
DS4HPCHR	00005	0000070D	01044	00598
DS4IDFMT	00001	0000070C	01043	00406
DUNDDNAM	00001	00000001	01423	00525
DUNUNALC	00001	00000007	01427	00533

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
DYNSP1	00004	000001D8	00916	00263 00503 00545
DYN1DS	00001	00000000	00352	00264
DYN1LEN	00001	00000090	00357	00268
DYN1MVC	00006	00000268	00340	00284 00309 00324
DYN1RB	00004	00000004	00354	00265 00346
DYN1RBP	00004	00000000	00353	00338 00357
DYN1TU	00100	0000002C	00356	00273 00350
DYN1TUP	00020	00000018	00355	00271 00287 00297 00312 00327 00334 00337
DYN2DS	00001	00000000	00551	00504
DYN2LEN	00001	00000032	00556	00508
DYN2MVC	00006	0000044C	00540	00524
DYN2RB	00004	00000004	00553	00505 00546
DYN2RBP	00004	00000000	00552	00538 00556
DYN2TU	00018	00000020	00555	00513
DYN2TUP	00008	00000018	00554	00511 00527 00534 00537
EOFSW	00001	00000010	00842	00139 00457 00614
ERRET	00004	0000058C	00692	00654 00663 00672 00681
ERRJFCB	00004	00000578	00684	00374
ERRJFCBM	00002	0000066E	00713	00684
EXCP	00002	00000480	00585	00170 00458
EXCPSAVE	00004	00000048	00822	00392 00393 00586 00587
FMT4	00044	000006E0	01040	00248 00909
GETDSB	00004	00000062	00138	00098
GETOUT	00004	0000008C	00157	00172
IECSDSL3	00001	00000770	01082	01083
IECSDSL4	00001	0000070C	01041	01042
IHADCB	00001	00000000	01439	00089 01524 01571 01636 01765 01780 01787 01800 01896 01902 01929 01952
INITCCW	00008	00000698	00715	00436
IOBCONL	00001	00000028	00776	00238
IOBCONST	00008	000006F0	00767	00238 00776
IOBCB	00004	0000011C	00889	00241
IOBECB	00004	0000010C	00886	00240
IOBFLAG1	00004	00000108	00885	00191
IOBSEEK	00008	00000128	00892	00197 00244 00396 00590 00598 00907
IOBSTART	00004	00000118	00888	00435
JEXLST	00004	00000280	00922	00231 00234 00235
JFCB	00001	00000000	01993	00376
JFCBAREA	00176	00000284	00923	00233 00375
JFCBDSNM	00044	00000000	02025	00377 00378 00378 00378
JFCBTSDM	00001	00000034	02031	00379
JFCNWRIT	00001	00000008	02040	00379
JFCRESRV	00004	0000006C	02469	02494
LASTCCW	00004	00000380	00450	00444
LASTDSB	00006	000000A0	00166	00155
MODESW	00001	000000C4	00838	00139 00145 00158 00181 00190 00360 00424 00457 00472 00478 00495 00559 00565 00572 00575 00605 00609 00614
MSGADDRS	00004	00000078	00967	00206 00649 00658 00667 00676 00686
NDSCBS	00004	00000098	00825	00409 00410 00410 00416 00430 00443
NDXSTORE	00004	0000007E	00153	00185
NOALLOC	00002	00000464	00561	00496
NOCLOSE	00002	000003D2	00491	00484
NOEXCP	00002	000003B8	00479	00473
NOFREE	00002	0000047A	00573	00566
NOTFMT4	00004	00000548	00665	00407
NOTFMT4M	00002	000005DA	00707	00665

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
OBTERR	00004	00000530	00656	00405
OBTERRM	00002	000005BA	00705	00656
OPEN	00002	00000128	00220	00099
OPENBIT	00001	00000010	00879	00388 00483
OPENERR	00004	00000518	00647	00389
OPENERRM	00002	00000594	00703	00647
OPENLIST	00004	000000FC	00880	00236 00368 00382 00486
PERMERR	00004	000000DC	00190	00183
RDCB	00001	00000008	00798	00088 00089 00241 00370 00385 00488
RDERR	00001	00000020	00841	00158 00190 00457 00575 00605
READCCW	00008	000006A0	00716	00440
READCKD	00001	0000001E	00805	00442 00716
READR0	00001	00000016	00804	00715
RETURN	00002	0000004E	00109	00140 00161 00641 00693
RETURN0	00002	0000004C	00107	00101 00159 00459 00466
RRCODE	00001	0000000A	00799	00620 00640
RRET	00001	00000009	00800	00170 00221 00458 00465 00576 00610
RWA	00001	00000002	00795	00093 00093 00094 00095 00096 00151 00152 00152 00153 00154 00156 00166 00184 00193 00196 00196 00197 00198 00199 00199 00201 00430 00431 00432 00432 00433 00441 00446 00446 00451
RWB	00001	00000003	00796	00167 00168 00169 00194 00198 00200 00200 00437 00440 00441 00442 00445 00445 00450
RWC	00001	00000004	00797	00438 00443 00447 00447
R0	00001	00000000	00115	00205 00205 00206 00279 00341 00394 00415 00416 00417 00418 00419 00455 00455 00456 00519 00541 00567 00588 00603 00603 00604 00648 00648 00649 00657 00657 00658 00666 00666 00667 00675 00675 00676 00685 00685 00686
R1	00001	00000001	00116	00083 00156 00157 00171 00204 00206 00208 00231 00232 00233 00234 00239 00240 00244 00245 00246 00247 00248 00249 00253 00254 00255 00256 00257 00258 00259 00260 00263 00264 00267 00268 00268 00342 00343 00375 00376 00380 00395 00423 00432 00435 00436 00437 00497 00498 00499 00500 00503 00504 00507 00508 00508 00542 00567 00589 00624 00625 00626 00628 00630 00632 00633 00633 00647 00649 00651 00656 00658 00660 00665 00667 00669 00674 00676 00678 00684 00686 00688
R11	00001	0000000B	00126	00083 00084
R13	00001	0000000D	00128	00082 00392 00400 00586 00595
R14	00001	0000000E	00129	00210 00271 00272 00280 00306 00321 00340 00343 00346 00399 00511 00512 00520 00540 00545 00546 00593 00625 00626 00627 00628 00629 00630 00631 00632 00653 00662 00671 00680 00690
R15	00001	0000000F	00130	00107 00107 00138 00160 00209 00210 00265 00266 00267 00273 00274 00287 00288 00288 00297 00298 00298 00312 00313 00313 00327 00328 00328 00334 00335 00335 00342 00348 00348 00373 00373 00397 00398 00398 00399 00404 00404 00408 00409 00505 00506 00507 00513 00514 00527 00528 00528 00534 00535 00535 00542 00547 00547 00591 00592 00592 00593 00596 00596 00620 00627 00640 00652 00653 00661 00662 00670 00671 00679 00680 00689 00690 00692
R2	00001	00000002	00117	00279 00281 00282 00283 00284 00303 00304 00304 00307 00308 00309 00318 00319 00319 00322 00323 00324 00341 00392 00396 00400 00519 00521 00522 00523 00524 00541 00586 00590 00595
R3	00001	00000003	00118	00393 00400 00587 00595
SETDSCBA	00004	000000D4	00184	00211
SETEOF	00004	000004C8	00614	00597 00599
SKIP	00001	00000010	00809	00715
SLI	00001	00000020	00808	00715
S99FAIL	00002	000004D0	00620	00349 00548
S99FLEN	00001	00000018	00623	00917
S99FLIST	00024	00000268	00917	00624
S99RB	00001	00000000	01153	00266 00506
S99RBLN	00001	00000000	01155	00269 00509
S99TUKEY	00002	00000000	01225	00285 00293 00310 00325 00333 00525 00533
S99TULNG	00002	00000004	01228	00282 00295 00307 00322 00522
S99TUNIT	00001	00000000	01224	00274 00514
S99TUNUM	00002	00000002	01226	00286 00294 00311 00326 00526

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
S99TUPAR	00001	00000006	01229	00296 00340 00540
S99TXTPP	00004	00000008	01187	00272 00512
S99VERB	00001	00000001	01156	00270 00510
S99VRBAL	00001	00000001	01157	00270
S99VRBUN	00001	00000002	01158	00510
TRACKERR	00002	0000063C	00711	00204
TTRN	00004	000000A8	00829	00167 00169 00456 00588
UNITADDR	00004	000000B4	00832	00254 00303 00306
UNITLEN	00002	000000B8	00833	00256
VADMSG	00004	0000001C	00946	00209 00652 00661 00670 00679 00689
VOLADDR	00004	000000AC	00830	00258 00318 00321
VOLID	00006	000001B9	01002	00246 00257 00908
VOLLEN	00002	000000B0	00831	00260
VTCEFUNC	00001	000001A4	00985	00094
VTCGETMN	00004	00000210	01021	01022
VTOCDCB	00004	000000C8	00849	00088 00227 00889
VTOCDCBM	00004	000006B8	00734	00227 00763
VTOCECB	00004	00000130	00896	00177 00182 00239 00475 00604 00886
VTOCEXCP	00001	00000000	00068	00075 00358 00557
VTOCIOB	00008	00000108	00884	00238 00607
VTOCNAME	00012	0000071E	00780	00778
VTOCNM	00004	00000718	00778	00280 00281
VTOCOM	00001	00000000	00932	00084 00208 00651 00660 00669 00678 00688
VTOCWORK	00001	00000000	00819	00082 00925
XCPRUN	00001	00000040	00840	00145 00181 00457 00472 00478 00609
XCPTTEST	00004	000000BE	00177	00146

SYMBOL	LEN	VALUE	DEFN	REFERENCES
=CL8' '	00008	00000730	00784	00296
=Y(DALSTATS,1,1,X'0800')				
	00002	00000738	00785	00333
=X'00010000'				
	00004	00000740	00786	00168
=X'00000100'				
	00004	00000744	00787	00394
=X'FFFFFFFF8'				
	00004	00000748	00788	00418
=A(0)	00004	0000074C	00789	00629
=H'15'	00002	00000750	00790	00417
=X'8032'	00002	00000752	00791	00631

ASM 0201 13.38 01/07/25

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 560

TOTAL RECORDS READ FROM SYSTEM LIBRARY 9615

TOTAL RECORDS PUNCHED 38

TOTAL RECORDS PRINTED 1395

SYMBOL	TYPE	ID	ADDR	LENGTH	LDID
VTOCFORM	SD	0001	000000	00053A	
PCLMAIN	SD	0002	000540	0007CE	

ASM 0201 13.38 01/07/25

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
2					*****	
3					*	*
4					*	*
5					* TITLE - VTOC COMMAND FORMAT ROUTINE	*
6					*	*
7					* FUNCTION - FORMAT THE DATA INTO THE VTFMT DSECT FROM THE	*
8					FORMAT 1 (AND 3 IF NEEDED) DSCB. THIS ROUTINE	*
9					ALSO GETS THE AREA TO CONTAIN THE FORMATTED	*
10					DSCB INFORMATION.	*
11					*	*
12					* OPERATION - FIRST GET AN AREA FROM THE CURRENT BLOCK, OR GET	*
13					A BLOCK (32K) OF STORAGE TO USE FOR THE FORMATTED	*
14					DSCB'S. MOVE THE DATA OVER FROM THE FORMAT 1 DSCB.	*
15					THE SPACE CALCULATIONS MAY NEED THE FORMAT 3 DSCB.	*
16					CATALOG INFORMATION IS OBTIANED VIA LOCATE. SOME	*
17					OF THE DSCB INFORMATION IS CONVERTED HERE.	*
18					*	*
19					* INPUT - VTOC COMMON AREA (VTOCOM)	*
20					POINTED TO BY REGISTER 1	*
21					USE PARSE DATA, CURRENT FORMATTED DSCB, LOCATE	*
22					*	*
23					* OUTPUT - THE FORMATTED DSCB INFORMATION WITH ITS ADDRESS IN	*
24					FORMATAD.	*
25					*	*
26					* ATTRIBUTES - REENTRANT, REUSEABLE, REFRESHABLE.	*
27					*	*
28					* PROGRAMMED BY R. L. MILLER (415) 485-6241	*
29					*	*
30					* 3/24/92 - MODIFIED BY ART TANSKY AT SUNGARD TO MAKE ALL VSAM TANSKY*	
31					* DATASETS "FULL" INSTEAD OF 0% USED. TANSKY*	
32					* - ADDED UPD AND RACF INDICATORS AND OPTIONS TANSKY*	
33					*	*
34					*	*
35					*****	
36					*	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				38	MACRO	
				39	&LAB DS1TST &FIELD,&VALUE,&CODE	
				40	&LAB TM DS1&FIELD,X'&VALUE' TEST IT	
				41	BNO D&SYSNDX IF NOT THERE, SKIP ALONG	
				42	MVC VTF&FIELD,=CL3'&CODE'	
				43	D&SYSNDX DS 0H	
				44	MEND	
				45	*	
				46	*	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				48	VTOCFORM ENTER 12,16	DO THE HOUSEKEEPING
000000				49+	VTOCFORM CSECT	
000000	47F0 F00E	0000E		50+	B 14(0,15)	BRANCH AROUND ID 00450000
000004	08			51+	DC AL1(8)	LENGTH OF IDENTIFIER 00550000
000005	E5E3D6C3C6D6D9D4			52+	DC CL8 'VTOCFORM'	IDENTIFIER 00750000
00000D	00					
00000E	90EC D00C	0000C		53+	STM 14,12,12(13)	SAVE REGISTERS 03700000
000012	18CF			54+	LR 12,15	SET FIRST BASE REG
000014				55+	CNOP 0,4	
			00000	56+	USING VTOCFORM,12	
000014	5811 0010	00010		57+	L 1,16+0(1)	NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
000018	50D1 0004	00004		58+	ST 13,4(1)	PRIOR SAVEAREA ADDRESS TO MINE
00001C	501D 0008	00008		59+	ST 1,8(13)	MY SAVEAREA ADDRESS TO HIS
000020	182D			60+	LR 2,13	KEEP THE SAVEAREA ADDRESS FOR REGS
000022	18D1			61+	LR 13,1	THIS IS MY SAVEAREA
000024	9802 2014	00014		62+	LM 0,2,20(2)	RESTORE ORIGINAL REGS
000028	18B1			63	LR R11,R1	SAVE ADDR OF VTOCOM
			00000	64	USING VTOCOM,R11	SET ITS ADDRESSABILITY
00002A	5890 B048	00048		65	L R9,ADDRANSR	POINT TO THE PARSE ANSWER
			00000	66	USING PDL,R9	SET ITS ADDRESSABILITY
			00000	67	USING FORMWORK,R13	SET ADDRESSABILITY FOR LOCAL WORK AREA
				69 *		
				70 *	CHECK FOR THE FIRST TIME THROUGH	
				71 *	IF SO, PERFORM SOME INITIALIZATION	
				72 *		
00002E	9500 D048	00048		73	CLI FIRSTFRM,0	IS THIS THE FIRST TIME?
000032	4770 C040	00040		74	BNE GETAREA	NO, KEEP ON TRUCKIN'
				75 *		
				76 *	ROUTINE INITIALIZATION	
				77 *		
000036	92FF D048	00048		78	MVI FIRSTFRM,255	NOTE THE INITIALIZATION AS DONE
00003A	D20F D05C C490	0005C 00490		79	MVC CAMLOC(CAMLEN),CAMCONST	SET UP THE CAMLST
				80 *		
				81 *	FIND OR GET AN AREA FOR THE FORMATTED DSCB	
				82 *	FIRST SEE HOW BIG IT IS	
				83 *		
000040	5870 B194	00194		84	GETAREA L R7,DSCBADDR	POINT TO THE DSCB
000044	4177 0008	00008		85	LA R7,8(R7)	GET PAST THE HEADER
			00000	86	USING DSCB1,R7	SET ADDRESSABILITY
000048	4810 B1AA	001AA		87	LH R1,DSNLEN	GET THE DSNAME LENGTH
00004C	4141 0046	00046		88	LA R4,VTFMTL(R1)	GET THE FORMATTED DSCB LENGTH
				89 *		
				90 *	SEE IF THE CURRENT BLOCK CAN HANDLE IT	
				91 *		
000050	5830 B20C	0020C		92	FORMFIT L R3,VTCCURLN	GET THE CURRENT AVAILABLE
000054	1B34			93	SR R3,R4	SEE IF IT WILL FIT
000056	4740 C3C8	003C8		94	BM GOGETMN	NO, GET ANOTHER BLOCK
				95 *		
				96 *	NO SWEAT, GET THE SPACE FROM THIS BLOCK	
				97 *		
00005A	5030 B20C	0020C		98	ST R3,VTCCURLN	STORE THE NEW (REDUCED) CURRENT LENGTH
00005E	5830 B208	00208		99	L R3,VTCCURAD	POINT TO THE CURRENT ADDRESS

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

000062 4153 4000      00000      100      LA    R5,0(R3,R4)    POINT TO THE END OF THE BLOCK
000066 5050 B208      00208      101      ST    R5,VTCCURAD   AND PLACE THE NEW AVAILABLE ADDRESS
                                102 *
                                103 *
                                104 *
                                105      USING VTFMT,R3      SET FORMATTED DSCB ADDRESSABILITY
00006A 5030 B198      00198      106      ST    R3,FORMATAD   SAVE THIS BLOCK'S ADDRESS
00006E D703 3000 3000 00000 00000 107      XC    VTFNEXT,VTFNEXT CLEAR THE SORT POINTER
000074 D205 3012 B1B9 00012 001B9 108      MVC  VTFVOLUM,VOLID SAVE THE VOLUME SERIAL NUMBER
00007A D203 3018 B1C4 00018 001C4 109      MVC  VTFUNIT,UCBDEVT SAVE THE DEVICE TYPE
000080 4810 B1AA      001AA      110      LH    R1,DSNLEN     GET THE LENGTH OF THE DSNAME
000084 4010 3044      00044      111      STH   R1,VTFDSNL    SAVE THE DSNAME LENGTH
000088 0610      112      BCTR  R1,0           SUBTRACT ONE FOR THE EX
00008A 4410 C464      00464      113      EX    R1,MOVEDSN    MOVE IN THE DSNAME
00008E D200 3025 703B 00025 0003B 114      MVC  VTFNOEPV,DS1NOEPV NUMBER OF EXTENTS
000094 D201 302E 7058 0002E 00058 115      MVC  VTFRECL,DS1LRECL LOGICAL RECORD LENGTH
00009A D201 3030 7056 00030 00056 116      MVC  VTFBLKSZ,DS1BLKL BLOCK SIZE
                                117 *
                                118 *
                                119 *
                                120      MVC  VTFCREDIT,DS1CREDIT MOVE OVER CREATION DATE
0000A0 D202 301C 7035 0001C 00035 120
0000A6 D202 301F 7038 0001F 00038 121      MVC  VTFEXPDT,DS1EXPDT MOVE OVER EXPIRATION DATE
0000AC D202 3022 704B 00022 0004B 122      MVC  VTFSTAC,DS1REFD  MOVE OVER LAST ACCESS DATE
                                123 *
                                124 *
                                125 *
                                126 *
                                127      MVC  VTFRECFM,BLANKS  BLANK THE FIELD TO START
0000B2 D204 3029 C488 00029 00488 127
0000B8 D207 303B C488 0003B 00488 128      MVC  VTFACON,BLANKS  ANOTHER BLANK FIELD
0000BE 9240 303A      0003A      129      MVI  VTFDSTYP,C' '  AND STILL ANOTHER
0000C2 4120 3029      00029      130      LA    R2,VTFRECFM   POINT TO THE FIELD
0000C6 91C0 7054      00054      131      TM    DS1RECFM,X'C0' UNKNOWN RECFM?
0000CA 4780 C0FA      000FA      132      BZ    RECFM2        YES, TROUBLE
0000CE 9140 7054      00054      133      TM    DS1RECFM,X'40' IS IT FIXED?
0000D2 4770 C0E2      000E2      134      BNZ   RECFM3        NO, KEEP TRYING
0000D6 92C6 2000      00000      135      MVI  0(R2),C'F'    YES, SET UP THE FIRST CHAR
0000DA 4122 0001      00001      136      LA    R2,1(R2)     AND BUMP THE POINTER
0000DE 47F0 C0FA      000FA      137      B     RECFM2        CHECK OTHER ATTRIBUTES
0000E2 9180 7054      00054      138 RECFM3 TM    DS1RECFM,X'80' SEE IF IT'S V OR U
0000E6 4780 C0F2      000F2      139      BZ    RECFM4        VARIABLE RECFM
0000EA 92E4 2000      00000      140      MVI  0(R2),C'U'    RECFM = U
0000EE 47F0 C0F6      000F6      141      B     RECFM4A       ADD TO THE POINTER AND KEEP LOOKING
0000F2 92E5 2000      00000      142 RECFM4 MVI  0(R2),C'V'    VARIABLE
0000F6 4122 0001      00001      143 RECFM4A LA    R2,1(R2)     GET PAST THIS CHAR
0000FA      144 RECFM2 DS    0H
0000FA 9110 7054      00054      145 RECFM5 TM    DS1RECFM,X'10' IS IT BLOCKED?
0000FE 4780 C10A      0010A      146      BZ    RECFM6        NO, SKIP ON
000102 92C2 2000      00000      147      MVI  0(R2),C'B'    YES, SET THE SYMBOL
000106 4122 0001      00001      148      LA    R2,1(R2)     GET PAST THE CHAR
00010A 9108 7054      00054      149 RECFM6 TM    DS1RECFM,X'08' IS IT SPANNED OR STANDARD?
00010E 4780 C11A      0011A      150      BZ    RECFM6A       NO
000112 92E2 2000      00000      151      MVI  0(R2),C'S'    YES, SET IT
000116 4122 0001      00001      152      LA    R2,1(R2)     GET PAST THIS CHARACTER
00011A 9120 7054      00054      153 RECFM6A TM    DS1RECFM,X'20' CHECK TRACK OVERFLOW
00011E 4780 C12A      0012A      154      BZ    RECFM7        NO DICE

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000122	92E3 2000	00000		155	MVI	0(R2),C'T'	YES, SET IT
000126	4122 0001	00001		156	LA	R2,1(R2)	PUSH THE POINTER ON
00012A	9104 7054	00054		157	RECFM7	TM DS1RECFM,X'04'	IS IT ASA CONTROL
00012E	4780 C13A	0013A		158	BZ	RECFM8	NO, SKIP ON
000132	92C1 2000	00000		159	MVI	0(R2),C'A'	YES, SET IT
000136	4122 0001	00001		160	LA	R2,1(R2)	GET PAST THIS CHAR
00013A	9102 7054	00054		161	RECFM8	TM DS1RECFM,X'02'	HOW ABOUT MACHINE CARRIAGE CONTROL
00013E	4780 C146	00146		162	BZ	RECFM9	NO, SKIP ON
000142	92D4 2000	00000		163	MVI	0(R2),C'M'	YES, SET IT
000146				164	RECFM9	DS 0H	
				165	*		
				166	*	FORMAT THE DSORG	
				167	*		
000146	D202 3026 C528	00026 00528		168	MVC	VTFDSORG,=CL3'	' CLEAR THE FIELD
				169	DS1TST	DSORG,80,IS	TRY ISAM
00014C	9180 7052	00052		170+	TM	DS1DSORG,X'80'	TEST IT
000150	47E0 C15A	0015A		171+	BNO	D0003	IF NOT THERE, SKIP ALONG
000154	D202 3026 C52B	00026 0052B		172+	MVC	VTFDSORG,=CL3'IS'	
00015A				173+	D0003	DS 0H	
				174	DS1TST	DSORG,40,PS	TRY SEQUENTIAL
00015A	9140 7052	00052		175+	TM	DS1DSORG,X'40'	TEST IT
00015E	47E0 C168	00168		176+	BNO	D0004	IF NOT THERE, SKIP ALONG
000162	D202 3026 C52E	00026 0052E		177+	MVC	VTFDSORG,=CL3'PS'	
000168				178+	D0004	DS 0H	
				179	DS1TST	DSORG,20,DA	TRY DIRECT ACCESS
000168	9120 7052	00052		180+	TM	DS1DSORG,X'20'	TEST IT
00016C	47E0 C176	00176		181+	BNO	D0005	IF NOT THERE, SKIP ALONG
000170	D202 3026 C531	00026 00531		182+	MVC	VTFDSORG,=CL3'DA'	
000176				183+	D0005	DS 0H	
				184	DS1TST	DSORG,02,PO	TRY PARTITIONED
000176	9102 7052	00052		185+	TM	DS1DSORG,X'02'	TEST IT
00017A	47E0 C184	00184		186+	BNO	D0006	IF NOT THERE, SKIP ALONG
00017E	D202 3026 C534	00026 00534		187+	MVC	VTFDSORG,=CL3'PO'	
000184				188+	D0006	DS 0H	
000184	D501 7052 C520	00052 00520		189	CLC	DS1DSORG(2),=X'0008'	IS IT VSAM?
00018A	4770 C194	00194		190	BNE	DSORG05	NO, KEEP LOOKING
00018E	D202 3026 C537	00026 00537		191	MVC	VTFDSORG,=CL3'VS'	YES, FLAG IT
000194	9101 7052	00052		192	DSORG05	TM DS1DSORG,X'01'	IS IT UNMOVEABLE?
000198	47E0 C1A0	001A0		193	BNO	DSORG06	NO, KEEP ON TRUCKIN'
00019C	92E4 3028	00028		194	MVI	VTFDSORG+2,C'U'	YES, NOTE IT
0001A0				195	DSORG06	DS 0H	
				196	*		
				197	*	FORMAT THE SECONDARY ALLOCATION	
				198	*		
0001A0	1B11			199	SR	R1,R1	CLEAR A WORK REGISTER
0001A2	4310 705E	0005E		200	IC	R1,DS1SCALO	GET THE ALLOCATION FLAG
0001A6	8810 0006	00006		201	SRL	R1,6	REMOVE THE BOTTOM 6 BITS (75 CENTS)
0001AA	4321 C4A2	004A2		202	IC	R2,SECAL(R1)	GET THE CHARACTER CODE
0001AE	4220 3039	00039		203	STC	R2,VTFSECAL	AND SAVE IT FOR LATER
0001B2	D201 3037 7060	00037 00060		204	MVC	VTFSECAM,DS1SCALO+2	SAVE THE SECONDARY AMOUNT TOO
0001B8	92D5 3032	00032		205	MVI	VTFROUND,C'N'	SET CODE FOR NO ROUND
0001BC	9101 705E	0005E		206	TM	DS1SCALO,X'01'	SEE IF ROUND WAS SET
0001C0	47E0 C1C8	001C8		207	BNO	PROTFORM	NO, THE CODE IS SET RIGHT
0001C4	92D9 3032	00032		208	MVI	VTFROUND,C'R'	YES, RESET THE CODE
				209	*		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				210 *	FORMAT THE PASSWORD PROTECTION	
				211 *		
0001C8	9114 705D	0005D		212	PROTFORM TM DS1DSIND,X'14' CHECK THE PASSWORD BITS	
0001CC	4710 C1DC	001DC		213	BO PROTWRIT WRITE PROTECT IS X'14'	
0001D0	4740 C1E4	001E4		214	BM PROTREAD READ PROTECT IS X'10'	
0001D4	92D5 3033	00033		215	MVI VTFPROT,C'N' NO PASSWORD PROTECTION	
0001D8	47F0 C1E8	001E8		216	B PROTEND END OF PROTECTION FORMATTING	
0001DC	92E6 3033	00033		217	PROTWRIT MVI VTFPROT,C'W' SET CODE FOR WRITE PROTECT	
0001E0	47F0 C1E8	001E8		218	B PROTEND THEN CHECK OTHER ITEMS	
0001E4	92D9 3033	00033		219	PROTREAD MVI VTFPROT,C'R' SET CODE FOR READ/WRITE PROTECT	
0001E8				220	PROTEND DS 0H END OF PROTECTION FORMATTING	
				221 *		TANSKY
				222 *	FORMAT THE RACF INDICATOR	TANSKY
				223 *		TANSKY
0001E8	92D5 3034	00034		224	MVI VTFRACF,C'N' ASSUME NOT RACF INDICATED	TANSKY
0001EC	9140 705D	0005D		225	TM DS1DSIND,DS1IND40 RACF IND?	TANSKY
0001F0	47E0 C1F8	001F8		226	BNO RACFEND NO, WE'RE RIGHT	TANSKY
0001F4	92E8 3034	00034		227	MVI VTFRACF,C'Y' YES, SHOW IT	TANSKY
0001F8				228	RACFEND DS 0H	TANSKY
				229 *		TANSKY
				230 *	FORMAT THE UPDATED INDICATOR	TANSKY
				231 *		TANSKY
0001F8	92D5 3035	00035		232	MVI VTFUPD,C'N' ASSUME NOT UPDATED	TANSKY
0001FC	9102 705D	0005D		233	TM DS1DSIND,DS1IND02 UPDATED?	TANSKY
000200	47E0 C208	00208		234	BNO UPDEND NO, WE'RE RIGHT	TANSKY
000204	92E8 3035	00035		235	MVI VTFUPD,C'Y' YES, SHOW IT	TANSKY
000208				236	UPDEND DS 0H	TANSKY
				237 *		
				238 *	FORMAT THE CATLG	
				239 *		
000208	9240 3036	00036		240	MVI VTFCATLG,C' ' INITIALIZE IT TO BLANKS	
00020C	9500 902D	0002D		241	CLI CATK+1,0 SHOULD WE DO THE LOCATE?	
000210	4780 C252	00252		242	BE CATEND NO, SKIP PAST IT	
				243 *		
				244 *	SET UP THE CAMLST	
				245 *		
000214	4110 7000	00000		246	LA R1,DS1DSNAM POINT TO THE DSNAME	
000218	5010 D060	00060		247	ST R1,CAMLOC+4 SAVE IT IN THE CAMLST	
00021C	4110 D070	00070		248	LA R1,LOCWORK LOCATE WORKAREA	
000220	5010 D068	00068		249	ST R1,CAMLOC+12 SAVE IT IN THE CAMLST	
				250	LOCATE CAMLOC CHECK THE CATALOG	
000224	4110 D05C	0005C		251+	LA 1,CAMLOC LOAD PARAMETER REG 1 01900002	
000228	0A1A			252+	SVC 26 LINK TO SERVICE ROUT. 00100000	
00022A	12FF			253	LTR R15,R15 TEST THE CATALOG RETURN CODE	
00022C	4780 C240	00240		254	BZ CATOK ZERO, THERE IS AN ENTRY	
000230	92D5 3036	00036		255	MVI VTFCATLG,C'N' SET CODE FOR NOT CATALOGED	
000234	49F0 C4A0	004A0		256	CH R15,H8 SEE IF THAT'S THE CASE	
000238	4780 C252	00252		257	BE CATEND YES, LET IT STAND	
00023C	92C5 3036	00036		258	MVI VTFCATLG,C'E' CATALOG ERROR, PROBLEMS	
				259 *		
				260 *	CATALOG ENTRY IS THERE, SEE THAT THE VOLUME IS THIS ONE	
				261 *		
000240	92C3 3036	00036		262	CATOK MVI VTFCATLG,C'C' SET UP AS A GOOD ENTRY	
000244	D505 B1B9 D076	001B9 00076		263	CLC VOLID,LOCWORK+6 COMPARE THE VOLUME SERIAL NUMBERS	
00024A	4780 C252	00252		264	BE CATEND GOOD, WE'RE DONE	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

00024E 92E6 3036      00036      265      MVI  VTFCATLG,C'W'  WRONG VOLUME, NOT CATALOGED
000252      266  CATEND  DS    0H
      267  *
      268  *      FORMAT THE ALLOCATION AND USED QUANTITIES
      269  *
      271  *      CHECK THROUGH THE EXTENTS

000252 1B22      273      SR    R2,R2      CLEAR A WORK REGISTER
000254 BF21 703B      0003B      274      ICM  R2,B'0001',DS1NOEPV  GET THE NUMBER OF EXTENTS
000258 4780 C2D2      002D2      275      BZ   SPACEND      NO EXTENTS MEANS NO SPACE
00025C 1B44      276      SR    R4,R4      ZERO THE SPACE COUNTER FOR THE DATA SET
      277  *
      278  *      GET EACH EXTENT AND PROCESS IT
      279  *
00025E 1B66      280      SR    R6,R6      FIRST EXTENT
000260 1856      281  EXTNEXT  LR    R5,R6      GET THE CURRENT EXTENT NUMBER
000262 8950 0002      00002      282      SLL  R5,2        MULTIPLY IT BY FOUR
000266 4405 C424      00424      283      EX   R0,GETEXT(R5)  GET THE CORRECT ADDRESS
      284  *
      285  *      PROCESS THIS EXTENT
      286  *
      00000 287      USING  XTDSECT,R5  SET ADDRESSABILITY
00026A 9500 5000      00000      288      CLI  XTFLAGS,XTNOEXT  IS THERE AN EXTENT
00026E 4780 C2C0      002C0      289      BE   NOEXT       NO, THE EXTENT ISN'T THERE
000272 9581 5000      00000      290      CLI  XTFLAGS,XTCYLBD  IS IT ON CYLINDER BOUNDARIES
000276 4770 C2A6      002A6      291      BNE  FORMALOC     NO, DO IT FOR CYLS AND TRACKS
      292  *
      293  *      CYLINDER BOUNDS - BE SURE THE ALLOCATION IS CORRECT
      294  *
00027A BF13 5004      00004      295      ICM  R1,B'0011',XTLOWHH  GET THE LOWER TRACK
00027E 4780 C28C      0028C      296      BZ   LOWOK       IT'S ZERO
000282 D205 303B C522 0003B 00522 297      MVC  VTFACTON(6),=C'CYLERR'  NOTE THE ERROR
000288 92D3 3041      00041      298      MVI  VTFACTON+6,C'L'  ON THE LOW CCHH
00028C 4810 5008      00008      299  LOWOK  LH    R1,XTHIHH    GET THE HIGH TRACK
000290 4111 0001      00001      300      LA   R1,1(R1)     ADD ONE FOR ZERO ADDRESSING
000294 4910 B720      00720      301      CH   R1,DS4DEVSZ+2  IS THIS THE NUMBER OF TRACKS/CYL
000298 4780 C2A6      002A6      302      BE   FORMALOC     YES, GO CALCULATE
00029C D205 303B C522 0003B 00522 303      MVC  VTFACTON(6),=C'CYLERR'  NOTE THE ERROR
0002A2 92C8 3042      00042      304      MVI  VTFACTON+7,C'H'  ON THE HIGH CCHH
      305  *
      306  *      GET THE SPACE FOR NON-CYLINDER ALLOCATIONS
      307  *
0002A6 4810 5006      00006      308  FORMALOC LH    R1,XTHICC    GET THE HIGH CYLINDER
0002AA 4B10 5002      00002      309      SH   R1,XTLOWCC    MINUS THE LOW CYLINDER
0002AE 4C10 B720      00720      310      MH   R1,DS4DEVSZ+2  TIMES THE NUMBER OF TRACKS PER CYLINDER
0002B2 4880 5008      00008      311      LH   R8,XTHIHH    GET THE HIGH TRACK
0002B6 4B80 5004      00004      312      SH   R8,XTLOWHH    MINUS THE LOW TRACK
0002BA 1A81      313      AR   R8,R1        TRACKS IN THIS EXTENT ( MINUS 1 )
0002BC 4144 8001      00001      314      LA   R4,1(R4,R8)  ADD THE TRACKS TOGETHER FOR THIS DATA SET
      315  *
      316  *      GET THE NEXT EXTENT
      317  *
0002C0 4166 0001      00001      318  NOEXT  LA   R6,1(R6)     INCREMENT THE EXTENT COUNTER
0002C4 1962      319      CR   R6,R2        CHECK FOR THE END

```

```

ASM 0201 13.38 01/07/25
LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
0002C6 4740 C260      00260      320      BL      EXTNEXT      NOT YET, KEEP GOING
      321 *
      322 *      ALL THE EXTENTS ARE SUMMED REGISTER 4 HAS THE SUM
      323 *
0002CA 4580 C366      00366      324      BAL     R8,SPACUNIT    CHANGE IT TO THE APPROPRIATE UNITS
0002CE 5040 3004      00004      325      ST      R4,VTFALLOC    STORE IT FOR LATER
0002D2      326 SPACEND DS      0H
      327 *
      328 *      GET THE TRACKS USED
      329 *
0002D2 1B44      330      SR      R4,R4          CLEAR THE TRACK ( WOO WOO )
0002D4 D502 7062 C46C 00062 0046C 331      CLC     DS1LSTAR,ZEROES IS THE TRACK USED COUNTER SET?
0002DA 4770 C314      00314      332      BNE     USEDOK      YES, ACCEPT IT
      333 *      NO, SEE IF THE ZERO IS VALID
0002DE 9140 7052      00052      334      TM      DS1DSORG,X'40' IS IT SEQUENTIAL?
0002E2 4710 C320      00320      335      BO      USEDOK0      YES,THE ZERO IS VALID
0002E6 9108 7053      00053      336      TM      DS1DSORG+1,DS1ACBM VSAM? TANSKY
0002EA 47E0 C2F8      002F8      337      BNO     USEDNVSM      NO, MORE CHECKING TANSKY
0002EE D203 3008 3004 00008 00004 338      MVC     VTFUSED,VTFALLOC IF VSAM, SET USED=ALLOC TANSKY
0002F4 47F0 C324      00324      339      B       USEDEND      AND CONTINUE TANSKY
0002F8      340 USEDNVSM DS      0H TANSKY
0002F8 D503 000A C46C 0000A 0046C 341      CLC     DSORG(4),ZEROES MAYBE IT WASN'T EVER OPENED
0002FE 4780 C320      00320      342      BE      USEDOK0      THEN NO SPACE USED IS OK
000302 910C 7052      00052      343      TM      DS1DSORG,X'0C' CHECK FOR AN INVALID DSORG
000306 4710 C320      00320      344      BO      USEDOK0      NO SPACE USED IS STILL OK
00030A D203 3008 C474 00008 00474 345      MVC     VTFUSED,FMIN1 SET A FLAG UNUSED SPACE UNKNOWN
000310 47F0 C324      00324      346      B       USEDEND      USED SPACE IS SET
      347 *
      348 *      THE TRACKS USED COUNTER SEEMS OK
      349 *
000314 4840 7062      00062      350 USEDOK LH      R4,DS1LSTAR    GET THE LAST TRACK USED
000318 4144 0001      00001      351      LA      R4,1(R4)     ADD ONE ( ZERO ADDRESSING )
00031C 4580 C366      00366      352      BAL     R8,SPACUNIT    CONVERT TO APPROPRIATE UNITS
000320 5040 3008      00008      353 USEDOK0 ST      R4,VTFUSED    SAVE THE AMOUNT OF SPACE USED
000324      354 USEDEND DS      0H
000324 58E0 3004      00004      355      L       R14,VTFALLOC   ALLOCATED TRACKS
000328 5BE0 3008      00008      356      S       R14,VTFUSED     MINUS USED TRACKS
00032C 50E0 300C      0000C      357      ST      R14,VTFUNUSD   EQUALS UNUSED TRACKS
000330 1BEE      358      SR      R14,R14
000332 1BFF      359      SR      R15,R15
000334 D503 3004 C518 00004 00518 360      CLC     VTFALLOC(4),=F'0'
00033A 4780 C34A      0034A      361      BE      USEDEND1
00033E 58F0 3008      00008      362      L       R15,VTFUSED    USED TRACKS
000342 5CE0 C51C      0051C      363      M       R14,=F'100'    MULT BY 100 TO GET PCT
000346 5DE0 3004      00004      364      D       R14,VTFALLOC   DIVIDE BY ALLOC TO GET PCT USED
00034A 40F0 3010      00010      365 USEDEND1 STH    R15,VTFPCT    SAVE PCT USED
      366 *
      367 *      RETURN
      368 *
      369 FORMRET LEAVE EQ,RC=0
00034E 182D      370+FORMRET LR     2,13
000350 58DD 0004      00004      371+      L       13,4(13)
000354 41F0 0000      00000      372+      LA      15,0          LOAD THE RETURN CODE
000358 90F1 D010      00010      373+      STM    15,1,16(13)   STORE RETURN REGS
00035C 98EC D00C      0000C      374+      LM     14,12,12(13)   RESTORE THE REGISTERS 00650000

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
000360	92FF D00C	0000C		375+	MVI 12(13),X'FF'	SET RETURN INDICATION 01600000
000364	07FE			376+	BR 14	RETURN 02000000
		00000		377+R0	EQU 0	*USED BY O.S.
		00001		378+R1	EQU 1	*USED BY O.S. // ADDRESS OF PARAMETER LIST
		00002		379+R2	EQU 2	
		00003		380+R3	EQU 3	
		00004		381+R4	EQU 4	
		00005		382+R5	EQU 5	
		00006		383+R6	EQU 6	
		00007		384+R7	EQU 7	
		00008		385+R8	EQU 8	
		00009		386+R9	EQU 9	
		0000A		387+R10	EQU 10	
		0000B		388+R11	EQU 11	
		0000C		389+R12	EQU 12	
		0000D		390+R13	EQU 13	*USED BY O.S. // SAVE-AREA ADDRESS
		0000E		391+R14	EQU 14	*USED BY O.S. // RETURN ADDRESS
		0000F		392+R15	EQU 15	*USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
				393 *		
				394 *		

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				396 *		
				397 *	ROUTINES USED ABOVE	
				398 *		
				399 *		
				400 *	CONVERT FROM TRACKS TO THE APPROPRIATE UNITS	
				401 *	KBYTES, MBYTES, TRKS, OR CYLS	
				402 *		
000366	4810 902A	0002A		403	SPACUNIT LH R1,SPACEK	GET THE UNIT TYPE
00036A	8910 0002	00002		404	SLL R1,2	MULTIPLY BY 4
00036E	47F1 C372	00372		405	B *+4(R1)	THEN BRANCH TO THE CORRECT ROUTINE
000372	47F0 C386	00386		406	B SPACTRK	R1=0 KILOBYTES
000376	47F0 C386	00386		407	B SPACTRK	R1=1 KILOBYTES
00037A	47F0 C386	00386		408	B SPACTRK	R1=2 MEGABYTES
00037E	47F0 C386	00386		409	B SPACTRK	R1=3 TRACKS
000382	47F0 C386	00386		410	B SPACTRK	R1=4 CYLINDERS
				411 *	TRACKS	
000386	07F8			412	SPACTRK BR R8	WAS SET WHEN WE STARTED
				413 *	CYLINDERS	
000388	1B00			414	SPACCYL SR R0,R0	CLEAR A REGISTER
00038A	1814			415	LR R1,R4	GET THE NUMBER OF TRACKS
00038C	4840 B720	00720		416	LH R4,DS4DEVSZ+2	GET THE NUMBER OF TRACKS PER CYLINDER
000390	8840 0002	00002		417	SRL R4,2	DIVIDE BY 2 FOR ROUNDING
000394	1A14			418	AR R1,R4	ADD IT IN
000396	4840 B720	00720		419	LH R4,DS4DEVSZ+2	GET THE NUMBER OF TRACKS PER CYLINDER
00039A	1D04			420	DR R0,R4	DIVIDE TO GET ROUNDED CYLINDERS
00039C	1841			421	LR R4,R1	GET THE ANSWER BACK INTO R4
00039E	07F8			422	BR R8	THEN RETURN
				423 *	KILOBYTES	
0003A0	4C40 B722	00722		424	SPACKB MH R4,DS4DEVTK	MULTIPLY BY BYTES PER TRACK
0003A4	1B00			425	SR R0,R0	CLEAR THE TOP
0003A6	1814			426	LR R1,R4	GET THE NUMBER TO DIVIDE
0003A8	5A10 C478	00478		427	A R1,F500	ADD UP TO ROUND
0003AC	5D00 C47C	0047C		428	D R0,F1000	DIVIDE TO GET KILOBYTES
0003B0	1841			429	LR R4,R1	GET THE ANSWER BACK INTO R4
0003B2	07F8			430	BR R8	THEN RETURN
				431 *	MEGABYTES	
0003B4	4C40 B722	00722		432	SPACMB MH R4,DS4DEVTK	MULTIPLY BY BYTES PER TRACK
0003B8	1B00			433	SR R0,R0	CLEAR THE TOP
0003BA	1814			434	LR R1,R4	GET THE NUMBER TO DIVIDE
0003BC	5A10 C480	00480		435	A R1,F500000	ADD UP TO ROUND
0003C0	5D00 C484	00484		436	D R0,F1000000	DIVIDE TO GET MEGABYTES
0003C4	1841			437	LR R4,R1	GET THE ANSWER BACK INTO R4
0003C6	07F8			438	BR R8	THEN RETURN
				439 *		
				440 *	GET A NEW BLOCK OF MAIN STORAGE	
				441 *		
				442	GOGETMN GETMAIN R,LV=VTCGETMS	GET SOME
0003C8				443+	CNOP 0,4	
0003C8	4510 C3D0	003D0		444+	GOGETMN BAL 1,*+8	BRANCH AROUND LENGTH
0003CC	00008000			445+	DC A(VTCGETMS)	LENGTH
0003D0	5800 1000	00000		446+	L 0,0(0,1)	LOAD LENGTH
0003D4	0A0A			447+	SVC 10	ISSUE GETMAIN SVC
0003D6	5010 B208	00208		448	ST R1,VTCCURAD	SET UP THE AVAILABLE ADDRESS
0003DA	4120 0020	00020		449	LA R2,VTCGETMS/1024	GET THE SIZE OF THE BLOCK IN K
0003DE	8920 000A	0000A		450	SLL R2,10	GET IT INTO BYTES (TIMES 1024)

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
0003E2 5020 B20C    0020C    451      ST    R2,VTCCURLN    SO THE FORMATTED DSCB'S CAN USE IT
      452 *
      453 *      SAVE THE BLOCK ADDRESS IN THE VTCGETMN TABLE
      454 *
0003E6 4120 B210    00210    455      LA    R2,VTCGETMN    POINT TO THE TABLE
0003EA 4150 0032    00032    456      LA    R5,VTCGETMX    GET THE NUMBER OF ENTRIES IN THE TABLE
0003EE BF3F 2000    00000    457 GOGETTAB ICM  R3,B'1111',0(R2)  GET THIS ENTRY
0003F2 4770 C3FE    003FE    458      BNZ   GOGETINC      IF NOT ZERO, KEEP LOOKING
0003F6 5012 0000    00000    459      ST    R1,0(R2)      SAVE THE NEW ENTRY
0003FA 47F0 C050    00050    460      B     FORMFIT       THEN GO ALLOCATE A FORMATTED DSCB
      461 *
      462 *      THIS ENTRY WAS TAKEN, GET THE NEXT ONE
      463 *
0003FE 4122 0004    00004    464 GOGETINC LA    R2,4(R2)      POINT TO THE NEXT ENTRY
000402 4650 C3EE    003EE    465      BCT   R5,GOGETTAB   COUNT AND LOOP
      466 *
      467 *      TABLE OVERFLOW - ISSUE ERROR MSG
      468 *      SET A FLAG TO STOP INPUT
      469 *
      470      VTOCMSG TABOVFLW,TABOVSEC  ISSUE A MESSAGE
000406 4110 C4A6    004A6    471+     LA    R1,TABOVFLW    POINT TO THE FIRST MESSAGE
00040A 4100 C4F0    004F0    472+     LA    R0,TABOVSEC    POINT TO THE SECOND MESSAGE
00040E 9001 B078    00078    473+     STM   R0,R1,MSGADDRS  SAVE THE MESSAGE ADDRESSES
      474+*     THEN JUST CALL THE MESSAGE ISSUING ROUTINE
000412 4110 B000    00000    475+     LA    R1,VTOCOM      POINT TO THE COMMON AREA
000416 58F0 B01C    0001C    476+     L     R15,VADMSG      POINT TO THE ROUTINE
00041A 05EF          477+     BALR  R14,R15        THEN CALL IT
00041C 92FF B1A6    001A6    478      MVI   TABFULL,255    SET A STOP FLAG
000420 47F0 C34E    0034E    479      B     FORMRET        RETURN FROM FORMATTING

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				481	*	
				482	*	
				483	*	
				484	*	
				485	*	
				487	*	
					INSTRUCTIONS EXECUTED TO GET THE NEXT EXTENT	
000424	4150 7069	00069		488	GETEXT LA R5,DS1EXT1 1ST EXTENT	
000428	4150 7073	00073		489	LA R5,DS1EXT2 2ND EXTENT	
00042C	4150 707D	0007D		490	LA R5,DS1EXT3 3RD EXTENT	
000430	4150 B774	00774		491	LA R5,DS3EXTNT+00 4TH EXTENT	
000434	4150 B77E	0077E		492	LA R5,DS3EXTNT+10 5TH EXTENT	
000438	4150 B788	00788		493	LA R5,DS3EXTNT+20 6TH EXTENT	
00043C	4150 B792	00792		494	LA R5,DS3EXTNT+30 7TH EXTENT	
000440	4150 B79D	0079D		495	LA R5,DS3ADEXT+00 8TH EXTENT	
000444	4150 B7A7	007A7		496	LA R5,DS3ADEXT+10 9TH EXTENT	
000448	4150 B7B1	007B1		497	LA R5,DS3ADEXT+20 10TH EXTENT	
00044C	4150 B7BB	007BB		498	LA R5,DS3ADEXT+30 11TH EXTENT	
000450	4150 B7C5	007C5		499	LA R5,DS3ADEXT+40 12TH EXTENT	
000454	4150 B7CF	007CF		500	LA R5,DS3ADEXT+50 13TH EXTENT	
000458	4150 B7D9	007D9		501	LA R5,DS3ADEXT+60 14TH EXTENT	
00045C	4150 B7E3	007E3		502	LA R5,DS3ADEXT+70 15TH EXTENT	
000460	4150 B7ED	007ED		503	LA R5,DS3ADEXT+80 16TH EXTENT	
000464	D200 3046 7000	00046 00000		504	MOVEDSN MVC VTFDSN(0),DS1DSNAM EXECUTED COMPARE	
00046A	0000					
00046C	0000000000000000			505	ZEROES DC 2F'0'	
000474	FFFFFFFF			506	FMIN1 DC F'-1'	
000478	000001F4			507	F500 DC F'500'	
00047C	000003E8			508	F1000 DC F'1000'	
000480	0007A120			509	F500000 DC F'500000'	
000484	000F4240			510	F1000000 DC F'1000000'	
000488	4040404040404040			511	BLANKS DC CL8'	
				512	CAMCONST CAMLST NAME,*,*,*	
000490				513+	CAMCONST DS 0F ALIGN ON FULL WORD 00349401	
000490	44			514+	DC AL1(68) THREE BYTES OF FLAGS 00349501	
000491	00			515+	DC AL1(0) INDICATING THE FUNC- 00349601	
000492	00			516+	DC AL1(0) TION TO BE PERFORMED 00399601	
000493	00			517+	DC AL1(0) NO OPTION THREE 00419601	
000494	00000494			518+	DC A(*) PARAMETER TWO 00441601	
000498	00000000			519+	DC A(0) PARAM. THREE OMMITTED 00445601	
00049C	0000049C			520+	DC A(*) PARAMETER FOUR 00448801	
0004A0	0008			521	H8 DC H'8'	
0004A2	C1C2E3C3			522	SECAL DC C'ABTC' SECONDARY ALLOCATION CODES	
				523	*	ABSOLUTE TRK, BLOCKS, TRACKS, CYLINDERS
				524	*	
				525	*	
				526	*	
				527		PRINT NOGEN
				528	*	
				529	*	PROGRAM MESSAGES
				530	*	
				531	TABOVFLW MSG	' THE VTOC TABLES (1.6 MEG) ARE NOT LARGE ENOUGH TO HANDX LE THIS REQUEST'
				533	TABOVSEC MSG	' PARTIAL PROCESSING WILL CONTINUE '
				535	*	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
-----	-------------	-------	-------	------	------------------

ASM 0201 13.38 01/07/25

				536 *	
				537 *	
				538 *	
				539 *	
				540 *	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				542 *		
				543 *		
				544 *	P A R S E C O N T R O L L I S T	
				545 *		
				546 *		
				547	PRINT OFF	
				549	PUSH PRINT	
				550	PRINT NOGEN	
				1718	POP PRINT	
				1745	PRINT ON	
				1746 *		
				1747 *	DYNAMIC WORK AREA	
				1748 *		
000000				1750	FORMWORK DSECT	
000000				1751	DS 18A	PRINT ROUTINE SAVE AREA
000048				1752	FIRSTFRM DS X	INITIALIZATION FOR THIS ROUTINE
000049				1753	CHARS DS CL16	CONVERSION TO CHARACTERS
				1754	CAMLOC CAMLST NAME,*,*,*	
		00010		1763	CAMLEN EQU *-CAMLOC	
000070				1764	DS 0D	
000070				1765	LOCWORK DS 265C	
000180				1767	DS 0D	
		00180		1768	LENWORK EQU *-FORMWORK	
				1769 *		
				1770 *	VTOC COMMAND COMMON AREA	
				1771 *		
				1772	PRINT NOGEN	
				1773	VTOCOM	
				1941 *		
				1942 *	FORMATTED DSCB	
				1943 *		
				1944	PRINT GEN	
				1945	VTFMT	
				1946+*		
				1947+*	THIS DSECT DESCRIBES THE FORMATTED DSCB	
				1948+*		
000000				1949+VTFMT	DSECT	
000000				1950+VTFNEXT	DS A	POINTER TO NEXT DSCB
000004				1951+VTFALLOC	DS F	ALLOCATION IN UNITS AS DEFINED BY THE
				1952+*		COMMAND. KBYTES, MBYTES, TRACKS, OR
				1953+*		CYLS MAY BE THE UNIT.
000008				1954+VTFUSED	DS F	AMOUNT USED, SAME UNIT
00000C				1955+VTFUNUSD	DS F	AMOUNT UNUSED, SAME UNIT
000010				1956+VTFPCT	DS H	PERCENT USED, 0-100
000012				1957+VTFVOLUM	DS CL6	VOLUME SERIAL NUMBER
000018				1958+VTFUNIT	DS CL4	UCB UNIT TYPE RPRINS
00001C				1959+VTFCREDIT	DS XL3	CREATION DATE YYDDD
00001F				1960+VTFEXPDT	DS XL3	EXPIRATION DATE YYDDD

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000022				1961+VTFLSTAC	DS XL3	LAST ACCESS DATE YYDDD	
000025				1962+VTFNOEPV	DS AL1	NUMBER OF EXTENTS PER VOLUME	
000026				1963+VTFDSORG	DS CL3	DATA SET ORGANIZATION	
				1964+*		PS, PO, DA, VS, IS, PERHAPS U	
000029				1965+VTFRECFM	DS CL5	RECORD FORMAT	
				1966+*		F,V, OR U, B, T, S, A, M	
00002E				1967+VTFLRECL	DS H	LOGICAL RECORD LENTGH	
000030				1968+VTFBLKSZ	DS H	BLOCK SIZE	
000032				1969+VTFROUND	DS C	R IF ROUND WAS SPECIFIED	
000033				1970+VTFPROT	DS C	PASSWORD PROTECTION FLAG	
000034				1971+VTFRACF	DS C	RACF INDICATOR FLAG	TANSKY
000035				1972+VTFUPD	DS C	UPDATE FLAG (UPDATED SINCE BACKUP)	TANSKY
000036				1973+VTFCATLG	DS C	CATALOG INDICATION	
000037				1974+VTFSECAM	DS XL2	SECONDARY AMOUNT	
000039				1975+VTFSECAL	DS C	SECONDARY ALLOCATION TYPE	
				1976+*		C FOR CYL, T FOR TRKS, B FOR BLOCKS	
				1977+*		R FOR BLOCKS WITH ROUND	
00003A				1978+VTFDSTYP	DS C	DATA SET TYPE, USER MAY DEFINE	
				1979+*		S = SYSTEM TEMPORARY DATA SET	
				1980+*		T = TEST DATA SET	
				1981+*		P = PRODUCTION DATA SET	
00003B				1982+VTFACON	DS CL8	REQUESTED ACTION OR COMMENT	
000044				1983+VTFDSNL	DS H	LENGTH OF DSNAME	
		00046		1984+VTFMTL	EQU *-VTFMT	FIXED LENGTH OF THIS DSECT	
000046				1985+VTFDSN	DS 44C	VARIABLE LENGTH FIELD	
				1986		PRINT NOGEN	
				1988		PDEDSNAM	
000000				2011 DSCB1	DSECT		
				2012	IECSDSL1 1		
				2095 *		FORMAT 1 AND 3 EXTENT DESCRIPTION	
000000				2096 XTDSECT	DSECT		
000000				2097 XTFLAGS	DS X		
		00000		2098 XTNOEXT	EQU X'00'	NO EXTENT	
		00001		2099 XTDATAB	EQU X'01'	DAT BLOCKS	
		00002		2100 XTOVFLW	EQU X'02'	OVERFLOW AREA	
		00004		2101 XTINDEX	EQU X'04'	INDEX AREA	
		00040		2102 XTUSRLAB	EQU X'40'	USER LABEL EXTENT	
		00080		2103 XTSHRCYL	EQU X'80'	SHARING CYLINDERS	
		00081		2104 XTCYLABD	EQU X'81'	CYLINDER BOUNDARIES	
000001				2105 XTSEQ	DS X	EXTENT SEQUENCE NUMBER	
000002				2106 XTLOWCC	DS H	LOWER CYLINDER	
000004				2107 XTLOWHH	DS H	LOWER TRACK	
000006				2108 XTHICC	DS H	UPPER CYLINDER	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000008				2109	XTHIHH	DS H	UPPER TRACK
				2110		END	
000518	00000000			2111		=F'0'	
00051C	00000064			2112		=F'100'	
000520	0008			2113		=X'0008'	
000522	C3E8D3C5D9D9			2114		=C'CYLERR'	
000528	404040			2115		=CL3' '	
00052B	C9E240			2116		=CL3'IS'	
00052E	D7E240			2117		=CL3'PS'	
000531	C4C140			2118		=CL3'DA'	
000534	D7D640			2119		=CL3'PO'	
000537	E5E240			2120		=CL3'VS'	

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	000494
0001	0001	0C	00049C

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
ADDRANSR	00004	00000048	01806	00065	
AL00381	00001	00000671	00780	00779	
AL00391	00001	0000067F	00794	00793	
ANDOR1K	00002	00000034	00768	00765	
ANDOR2K	00002	00000036	00807	00804	
ANDOR3K	00002	00000038	00838	00835	
BLANKS	00008	00000488	00511	00127	00128
BLKSZSET	00004	000000BC	01208	01197	
BREAK	00004	000000AC	01161	01148	
BREAKK	00002	00000030	00726	00723	
CAMCONST	00004	00000490	00513	00079	
CAMLEN	00001	00000010	01763	00079	
CAMLOC	00004	0000005C	01755	00079	00247 00249 00251 01763
CATEND	00002	00000252	00266	00242	00257 00264
CATK	00002	0000002C	00676	00241	00673
CATOK	00004	00000240	00262	00254	
CHARSK	00002	0000003C	00899	00896	
CHARSPL	00004	000000B4	01188	01177	
CONTAIN	00004	00000084	01104	01098	
CONTAINK	00002	00000028	00632	00629	
DEND0071	00001	00000821	01130	01120	
DEND0073	00001	00000854	01159	01147	
DEND0075	00001	00000885	01186	01176	
DEND0076	00001	000008A6	01206	01196	
DEND0078	00001	000008D7	01233	01223	
DEND0080	00001	000008FD	01260	01250	
DEND0081	00001	00000919	01280	01270	
DEND0085	00001	00000955	01327	01315	
DEND0087	00001	00000990	01354	01344	
DEND0088	00001	000009DF	01377	01365	
DEND0089	00001	00000A23	01399	01388	
DEND0091	00001	00000A5E	01426	01416	
DEND0092	00001	00000AAD	01449	01437	
DEND0093	00001	00000AF1	01471	01460	
DEND0095	00001	00000B2C	01498	01488	
DEND0096	00001	00000B7B	01521	01509	
DEND0097	00001	00000BBF	01543	01532	
DEND0099	00001	00000BFA	01570	01560	
DEND0100	00001	00000C49	01593	01581	
DEND0101	00001	00000C8D	01615	01604	
DEND0103	00001	00000CB6	01644	01632	
DEND0105	00001	00000CE3	01673	01661	
DENT0071	00001	00000808	01118	01120	
DENT0073	00001	00000824	01145	01147	
DENT0075	00001	00000857	01174	01176	
DENT0076	00001	00000885	01194	01196	
DENT0078	00001	000008A9	01221	01223	
DENT0080	00001	000008DA	01248	01250	
DENT0081	00001	000008FD	01268	01270	
DENT0085	00001	00000926	01313	01315	
DENT0087	00001	00000958	01342	01344	
DENT0088	00001	00000990	01363	01365	
DENT0089	00001	000009DF	01386	01388	
DENT0091	00001	00000A26	01414	01416	
DENT0092	00001	00000A5E	01435	01437	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
DENT0093	00001	00000AAD	01458	01460
DENT0095	00001	00000AF4	01486	01488
DENT0096	00001	00000B2C	01507	01509
DENT0097	00001	00000B7B	01530	01532
DENT0099	00001	00000BC2	01558	01560
DENT0100	00001	00000BFA	01579	01581
DENT0101	00001	00000C49	01602	01604
DENT0103	00001	00000C90	01630	01632
DENT0105	00001	00000CB9	01659	01661
DSCBADDR	00004	00000194	01823	00084
DSCB1	00001	00000000	02011	00086
DSNLEN	00002	000001AA	01838	00087 00110
DSNLNTYP	00002	00000164	01688	01683
DSNPLN	00004	0000015C	01675	01662
DSNPLNK	00002	00000048	01033	01030
DSORG	00001	0000000A	01728	00341
DSORG05	00004	00000194	00192	00190
DSORG06	00002	000001A0	00195	00193
DS1ACBM	00001	00000008	02049	00336
DS1BLKL	00002	00000056	02055	00116
DS1CREDT	00003	00000035	02019	00120
DS1DSIND	00001	0000005D	02059	00212 00225 00233
DS1DSNAM	00044	00000000	02015	00246 00504
DS1DSORG	00002	00000052	02028	00170 00175 00180 00185 00189 00192 00334 00336 00343
DS1EXPDT	00003	00000038	02020	00121
DS1EXT1	00010	00000069	02085	00488
DS1EXT2	00010	00000073	02090	00489
DS1EXT3	00010	0000007D	02091	00490
DS1IND01	00001	00000001	02079	02080
DS1IND02	00001	00000002	02075	00233 02078
DS1IND40	00001	00000040	02062	00225
DS1LRECL	00002	00000058	02056	00115
DS1LSTAR	00003	00000062	02082	00331 00350
DS1NOEPV	00001	0000003B	02021	00114 00274
DS1RECFM	00001	00000054	02053	00131 00133 00138 00145 00149 00153 00157 00161
DS1REFD	00003	0000004B	02026	00122
DS1SCALO	00004	0000005E	02081	00200 00204 00206
DS3ADEXT	00090	0000079D	01936	00495 00496 00497 00498 00499 00500 00501 00502 00503
DS3EXTNT	00040	00000774	01930	00491 00492 00493 00494
DS4DEVSZ	00004	0000071E	01903	00301 00310 00416 00419
DS4DEVTK	00002	00000722	01904	00424 00432
D0003	00002	0000015A	00173	00171
D0004	00002	00000168	00178	00176
D0005	00002	00000176	00183	00181
D0006	00002	00000184	00188	00186
ENDING	00004	00000068	01083	01077
ENDKEY	00002	00000026	00611	00608
EXTNEXT	00002	00000260	00281	00320
FIRSTFRM	00001	00000048	01752	00073 00078
FMIN1	00004	00000474	00506	00345
FORMALOC	00004	000002A6	00308	00291 00302
FORMATAD	00004	00000198	01824	00106
FORMATK	00002	00000046	01012	01009
FORMATSP	00004	00000150	01646	01633
FORMFIT	00004	00000050	00092	00460

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
FORMRET	00002	0000034E	00370	00479	
FORMWORK	00001	00000000	01750	00067	01768
F1000	00004	0000047C	00508	00428	
F1000000	00004	00000484	00510	00436	
F500	00004	00000478	00507	00427	
F500000	00004	00000480	00509	00435	
GETAREA	00004	00000040	00084	00074	
GETEXT	00004	00000424	00488	00283	
GOGETINC	00004	000003FE	00464	00458	
GOGETMN	00004	000003C8	00444	00094	
GOGETTAB	00004	000003EE	00457	00465	
HEADING	00004	000000E0	01300	01296	
HEADK	00002	00000040	00941	00938	
H8	00002	000004A0	00521	00256	
IECSDSL1	00001	00000000	02013	02014	
IECSDSL3	00001	00000770	01927	01928	
IECSDSL4	00001	0000070C	01886	01887	
IKJ\$0017	00001	00000D0E	01715	00553	
IKJ\$0019	00001	000005BD	00574	00570	
IKJ\$0073	00001	00000854	01158	01157	
IKJ\$0085	00001	00000955	01326	01325	
IKJ\$0088	00001	000009DF	01376	01375	
IKJ\$0089	00001	00000A23	01398	01397	
IKJ\$0092	00001	00000AAD	01448	01447	
IKJ\$0093	00001	00000AF1	01470	01469	
IKJ\$0096	00001	00000B7B	01520	01519	
IKJ\$0097	00001	00000BBF	01542	01541	
IKJ\$0100	00001	00000C49	01592	01591	
IKJ\$0101	00001	00000C8D	01614	01613	
IKJ\$0103	00001	00000CB6	01643	01642	
IKJ\$0105	00001	00000CE3	01672	01671	
IKJ@0017	00001	00000168	01713	00554	
IKJ@0018	00001	0000007D	00583	00555	
IKJ@0019	00001	00000574	00569	00568	
IKJ@0021	00001	000005CD	00598	00597	
IKJ@0023	00001	000005E0	00619	00618	
IKJ@0025	00001	000005F7	00640	00639	
IKJ@0026	00001	00000604	00653	00652	
IKJ@0027	00001	0000060D	00664	00663	
IKJ@0029	00001	0000061B	00684	00683	
IKJ@0031	00001	0000062A	00704	00703	
IKJ@0032	00001	00000637	00714	00713	
IKJ@0034	00001	00000647	00734	00733	
IKJ@0036	00001	00000659	00755	00754	
IKJ@0038	00001	0000066A	00776	00775	
IKJ@0039	00001	00000679	00790	00789	
IKJ@0041	00001	0000068E	00815	00814	
IKJ@0042	00001	00000698	00825	00824	
IKJ@0044	00001	000006A9	00846	00845	
IKJ@0045	00001	000006B3	00856	00855	
IKJ@0047	00001	000006C5	00877	00876	
IKJ@0048	00001	000006D3	00887	00886	
IKJ@0050	00001	000006E3	00907	00906	
IKJ@0052	00001	000006F5	00928	00927	
IKJ@0054	00001	00000709	00949	00948	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
IKJ@0055	00001	00000719	00959	00958	
IKJ@0057	00001	0000072A	00979	00978	
IKJ@0059	00001	0000073D	01000	00999	
IKJ@0061	00001	0000074E	01020	01019	
IKJ@0063	00001	00000761	01041	01040	
IKJ@0064	00001	00000258	01068	01049	
IKJ@0065	00001	00000798	01059	01058	
IKJ@0066	00001	0000028A	01089	01070	
IKJ@0067	00001	000007CA	01080	01079	
IKJ@0068	00001	000002C5	01110	01091	
IKJ@0069	00001	00000805	01101	01100	
IKJ@0070	00001	000002E1	01137	01112	
IKJ@0071	00001	00000820	01128	01127	
IKJ@0072	00001	00000314	01166	01139	
IKJ@0073	00001	00000851	01155	01154	
IKJ@0074	00001	00000366	01213	01168	
IKJ@0075	00001	00000884	01184	01183	
IKJ@0076	00001	000008A5	01204	01203	
IKJ@0077	00001	00000397	01240	01215	
IKJ@0078	00001	000008D6	01231	01230	
IKJ@0079	00001	000003D9	01287	01242	
IKJ@0080	00001	000008FC	01258	01257	
IKJ@0081	00001	00000918	01278	01277	
IKJ@0082	00001	000003E3	01305	01289	
IKJ@0084	00001	00000415	01334	01307	
IKJ@0085	00001	00000952	01323	01322	
IKJ@0086	00001	000004E3	01406	01336	
IKJ@0087	00001	0000098F	01352	01351	
IKJ@0088	00001	000009B4	01373	01372	
IKJ@0089	00001	000009FC	01396	01395	
IKJ@0090	00001	000005B1	01478	01408	
IKJ@0091	00001	00000A5D	01424	01423	
IKJ@0092	00001	00000A82	01445	01444	
IKJ@0093	00001	00000ACA	01468	01467	
IKJ@0094	00001	0000067F	01550	01480	
IKJ@0095	00001	00000B2B	01496	01495	
IKJ@0096	00001	00000B50	01517	01516	
IKJ@0097	00001	00000B98	01540	01539	
IKJ@0098	00001	0000074D	01622	01552	
IKJ@0099	00001	00000BF9	01568	01567	
IKJ@0100	00001	00000C1E	01589	01588	
IKJ@0101	00001	00000C66	01612	01611	
IKJ@0102	00001	00000776	01651	01624	
IKJ@0103	00001	00000CB3	01640	01639	
IKJ@0104	00001	000007A3	01679	01653	
IKJ@0105	00001	00000CDF	01669	01668	
IKJ@0106	00001	00000CF2	01685	01684	
IKJ@0107	00001	00000CFF	01696	01695	
IKJ@0108	00001	00000D0D	01705	01704	
IKJ00191	00001	000005BD	00573	00572	
KEND0020	00001	000005C3	00588	00586	
KEND0022	00001	000005D5	00609	00607	
KEND0024	00001	000005E8	00630	00628	
KEND0026	00001	00000604	00654	00650	
KEND0028	00001	00000613	00674	00672	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
KEND0030	00001	00000621	00694	00692	
KEND0033	00001	0000063D	00724	00722	
KEND0035	00001	0000064F	00745	00743	
KEND0037	00001	00000661	00766	00764	
KEND0040	00001	00000685	00805	00803	
KEND0043	00001	000006A0	00836	00834	
KEND0046	00001	000006BB	00867	00865	
KEND0049	00001	000006D9	00897	00895	
KEND0051	00001	000006EB	00918	00916	
KEND0053	00001	000006FD	00939	00937	
KEND0056	00001	0000071F	00969	00967	
KEND0058	00001	00000732	00990	00988	
KEND0060	00001	00000743	01010	01008	
KEND0062	00001	00000756	01031	01029	
KEND0106	00001	00000CF2	01686	01682	
KEYW0020	00001	000005BD	00584	00586	
KEYW0022	00001	000005CF	00605	00607	
KEYW0024	00001	000005E2	00626	00628	
KEYW0026	00001	000005F9	00648	00650	
KEYW0028	00001	0000060D	00670	00672	
KEYW0030	00001	0000061B	00690	00692	
KEYW0033	00001	00000637	00720	00722	
KEYW0035	00001	00000649	00741	00743	
KEYW0037	00001	0000065B	00762	00764	
KEYW0040	00001	0000067F	00801	00803	
KEYW0043	00001	0000069A	00832	00834	
KEYW0046	00001	000006B5	00863	00865	
KEYW0049	00001	000006D3	00893	00895	
KEYW0051	00001	000006E5	00914	00916	
KEYW0053	00001	000006F7	00935	00937	
KEYW0056	00001	00000719	00965	00967	
KEYW0058	00001	0000072C	00986	00988	
KEYW0060	00001	0000073D	01006	01008	
KEYW0062	00001	00000750	01027	01029	
KEYW0106	00001	00000CE3	01680	01682	
LEVEL	00004	0000004C	01062	01056	
LEVKEY	00002	00000024	00590	00587	
LIMITK	00002	00000032	00747	00744	
LINESK	00002	0000003E	00920	00917	
LINESPP	00004	000000C4	01235	01224	
LOCWORK	00001	00000070	01765	00248	00263
LOWOK	00004	0000028C	00299	00296	
MOVEDSN	00006	00000464	00504	00113	
MSGADDRS	00004	00000078	01812	00473	
NAME0021	00001	000005C3	00594	00596	
NAME0023	00001	000005D5	00615	00617	
NAME0025	00001	000005E8	00636	00638	
NAME0027	00001	00000604	00660	00662	
NAME0029	00001	00000613	00680	00682	
NAME0031	00001	00000621	00700	00702	
NAME0032	00001	0000062C	00710	00712	
NAME0034	00001	0000063D	00730	00732	
NAME0036	00001	0000064F	00751	00753	
NAME0038	00001	00000661	00772	00774	
NAME0039	00001	00000671	00786	00788	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25																								
NAME0041	00001	00000685	00811	00813																									
NAME0042	00001	00000690	00821	00823																									
NAME0044	00001	000006A0	00842	00844																									
NAME0045	00001	000006AB	00852	00854																									
NAME0047	00001	000006BB	00873	00875																									
NAME0048	00001	000006C7	00883	00885																									
NAME0050	00001	000006D9	00903	00905																									
NAME0052	00001	000006EB	00924	00926																									
NAME0054	00001	000006FD	00945	00947																									
NAME0055	00001	0000070B	00955	00957																									
NAME0057	00001	0000071F	00975	00977																									
NAME0059	00001	00000732	00996	00998																									
NAME0061	00001	00000743	01016	01018																									
NAME0063	00001	00000756	01037	01039																									
NAME0107	00001	00000CF2	01692	01694																									
NAME0108	00001	00000CFF	01701	01703																									
NEND0021	00001	000005CF	00600	00596																									
NEND0023	00001	000005E2	00621	00617																									
NEND0025	00001	000005F9	00642	00638																									
NEND0027	00001	0000060D	00665	00662																									
NEND0029	00001	0000061B	00685	00682																									
NEND0031	00001	0000062C	00706	00702																									
NEND0032	00001	00000637	00715	00712																									
NEND0034	00001	00000649	00736	00732																									
NEND0036	00001	0000065B	00757	00753																									
NEND0038	00001	00000671	00782	00774																									
NEND0039	00001	0000067F	00796	00788																									
NEND0041	00001	00000690	00817	00813																									
NEND0042	00001	0000069A	00827	00823																									
NEND0044	00001	000006AB	00848	00844																									
NEND0045	00001	000006B5	00858	00854																									
NEND0047	00001	000006C7	00879	00875																									
NEND0048	00001	000006D3	00888	00885																									
NEND0050	00001	000006E5	00909	00905																									
NEND0052	00001	000006F7	00930	00926																									
NEND0054	00001	0000070B	00951	00947																									
NEND0055	00001	00000719	00960	00957																									
NEND0057	00001	0000072C	00981	00977																									
NEND0059	00001	0000073D	01001	00998																									
NEND0061	00001	00000750	01022	01018																									
NEND0063	00001	00000763	01043	01039																									
NEND0107	00001	00000CFF	01697	01694																									
NEND0108	00001	00000D0D	01706	01703																									
NOEXT	00004	000002C0	00318	00289																									
OUTPUTK	00002	00000044	00992	00989																									
PCLMAIN	00001	00000540	00552	00553	00556	00562	00582	00583	00593	00599	00604	00614	00620	00625	00635	00641	00647	00659											
				00669	00679	00689	00699	00705	00709	00719	00729	00735	00740	00750	00756	00761	00771	00777											
				00785	00791	00800	00810	00816	00820	00826	00831	00841	00847	00851	00857	00862	00872	00878											
				00882	00892	00902	00908	00913	00923	00929	00934	00944	00950	00954	00964	00974	00980	00985											
				00995	01005	01015	01021	01026	01036	01042	01047	01052	01067	01068	01073	01088	01089	01094											
				01109	01110	01115	01136	01137	01142	01165	01166	01171	01191	01212	01213	01218	01239	01240											
				01245	01265	01286	01287	01292	01304	01305	01310	01333	01334	01339	01360	01383	01405	01406											
				01411	01432	01455	01477	01478	01483	01504	01527	01549	01550	01555	01576	01599	01621	01622											
				01627	01650	01651	01656	01678	01679	01691	01700	01709	01714																
PDL	00001	00000000	00557	00066	00566	00576	00587	00589	00608	00610	00629	00631	00651	00655	00673	00675	00693	00695											

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25													
				00723 00725 00744 00746 00765 00767 00804 00806 00835 00837 00866 00868 00896 00898 00917														
				00919 00938 00940 00968 00970 00989 00991 01009 01011 01030 01032 01056 01061 01077 01082														
				01098 01103 01121 01131 01148 01160 01177 01187 01197 01207 01224 01234 01251 01261 01271														
				01281 01296 01299 01316 01328 01345 01355 01366 01378 01389 01400 01417 01427 01438 01450														
				01461 01472 01489 01499 01510 01522 01533 01544 01561 01571 01582 01594 01605 01616 01633														
				01645 01662 01674 01683 01687 01711 01713														
PEND0019	00001	000005BD	00575	00565														
PEND0065	00001	00000798	01060	01055														
PEND0067	00001	000007CA	01081	01076														
PEND0069	00001	00000805	01102	01097														
PEND0083	00001	00000923	01298	01295														
POST0019	00001	00000546	00563	00565														
POST0065	00001	00000766	01053	01055														
POST0067	00001	0000079B	01074	01076														
POST0069	00001	000007CD	01095	01097														
POST0083	00001	0000091C	01293	01295														
PRINTK	00002	0000003A	00869	00866														
PROTEND	00002	000001E8	00220	00216 00218														
PROTFORM	00004	000001C8	00212	00207														
PROTREAD	00004	000001E4	00219	00214														
PROTWRT	00004	000001DC	00217	00213														
RACFEND	00002	000001F8	00228	00226														
RECFM2	00002	000000FA	00144	00132 00137														
RECFM3	00004	000000E2	00138	00134														
RECFM4	00004	000000F2	00142	00139														
RECFM4A	00004	000000F6	00143	00141														
RECFM6	00004	0000010A	00149	00146														
RECFM6A	00004	0000011A	00153	00150														
RECFM7	00004	0000012A	00157	00154														
RECFM8	00004	0000013A	00161	00158														
RECFM9	00002	00000146	00164	00162														
R0	00001	00000000	00377	00283 00414 00414 00420 00425 00425 00428 00433 00433 00436 00472 00473														
R1	00001	00000001	00378	00063 00087 00088 00110 00111 00112 00113 00199 00199 00200 00201 00202 00246 00247 00248														
				00249 00295 00299 00300 00300 00301 00308 00309 00310 00313 00403 00404 00405 00415 00418														
				00421 00426 00427 00429 00434 00435 00437 00448 00459 00471 00473 00475														
R11	00001	0000000B	00388	00063 00064														
R13	00001	0000000D	00390	00067														
R14	00001	0000000E	00391	00355 00356 00357 00358 00358 00363 00364 00477														
R15	00001	0000000F	00392	00253 00253 00256 00359 00359 00362 00365 00476 00477														
R2	00001	00000002	00379	00130 00135 00136 00136 00140 00142 00143 00143 00147 00148 00148 00151 00152 00152 00155														
				00156 00156 00159 00160 00160 00163 00202 00203 00273 00273 00274 00319 00449 00450 00451														
				00455 00457 00459 00464 00464														
R3	00001	00000003	00380	00092 00093 00098 00099 00100 00105 00106 00457														
R4	00001	00000004	00381	00088 00093 00100 00276 00276 00314 00314 00325 00330 00330 00350 00351 00351 00353 00415														
				00416 00417 00418 00419 00420 00421 00424 00426 00429 00432 00434 00437														
R5	00001	00000005	00382	00100 00101 00281 00282 00283 00287 00456 00465 00488 00489 00490 00491 00492 00493 00494														
				00495 00496 00497 00498 00499 00500 00501 00502 00503														
R6	00001	00000006	00383	00280 00280 00281 00318 00318 00319														
R7	00001	00000007	00384	00084 00085 00085 00086														
R8	00001	00000008	00385	00311 00312 00313 00314 00324 00352 00412 00422 00430 00438														
R9	00001	00000009	00386	00065 00066														
SECAL	00004	000004A2	00522	00202														
SORTK	00002	0000002E	00696	00693														
SPACEK	00002	0000002A	00656	00403 00651														
SPACEND	00002	000002D2	00326	00275														

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
SPACTRK	00002	00000386	00412	00406 00407 00408 00409 00410
SPACUNIT	00004	00000366	00403	00324 00352
SUBAO1	00002	00000A24	01408	00777 00791
SUBAO2	00002	00000AF2	01480	00816 00826
SUBAO3	00002	00000BC0	01552	00847 00857
SUBBREAK	00002	00000822	01139	00735
SUBCHARS	00002	00000855	01168	00908
SUBCONT	00002	000007CB	01091	00641
SUBDSNLN	00002	00000CB7	01653	01042
SUBEND	00002	00000799	01070	00620
SUBFORMT	00002	00000C8E	01624	01021
SUBHEAD	00002	0000091A	01289	00950
SUBLEV	00002	00000764	01049	00599
SUBLIMIT	00002	00000956	01336	00756
SUBLINES	00002	000008A7	01215	00929
SUBLKEY	00004	000000F0	01356	01345
SUBLOPER	00004	000000F8	01379	01366
SUBLVALU	00004	00000100	01401	01389
SUBPRINT	00002	000008D8	01242	00878
SUBPRTIT	00004	000000D4	01282	01271
SUBPRTKY	00004	000000CC	01262	01251
SUBSORT	00004	000000A0	01132	01121
SUBSORTS	00002	00000806	01112	00705
SUBTOTAL	00002	00000924	01307	00980
SUB1KEY	00004	00000108	01428	01417
SUB1OPER	00004	00000110	01451	01438
SUB1VALU	00004	00000118	01473	01461
SUB2KEY	00004	00000120	01500	01489
SUB2OPER	00004	00000128	01523	01510
SUB2VALU	00004	00000130	01545	01533
SUB3KEY	00004	00000138	01572	01561
SUB3OPER	00004	00000140	01595	01582
SUB3VALU	00004	00000148	01617	01605
TABFULL	00001	000001A6	01835	00478
TABOVFLW	00002	000004A6	00532	00471
TABOVSEC	00002	000004F0	00534	00472
TOTALK	00002	00000042	00971	00968
TOTALN	00004	000000E8	01329	01316
UCBDEVT	00004	000001C4	01850	00109
UPDEND	00002	00000208	00236	00234
USEDEND	00002	00000324	00354	00339 00346
USEDEND1	00004	0000034A	00365	00361
USEDNVSM	00002	000002F8	00340	00337
USEDOK	00004	00000314	00350	00332
USEDOK0	00004	00000320	00353	00335 00342 00344
VADMSG	00004	0000001C	01791	00476
VOLID	00006	000001B9	01847	00108 00263
VOLS	00004	00000008	00577	00566
VTCCURAD	00004	00000208	01864	00099 00101 00448
VTCCURLN	00004	0000020C	01865	00092 00098 00451
VTCGETMN	00004	00000210	01866	00455 01867
VTCGETMS	00001	00008000	01868	00445 00449
VTCGETMX	00001	00000032	01867	00456
VTFACTON	00008	0000003B	01982	00128 00297 00298 00303 00304
VTFALLOC	00004	00000004	01951	00325 00338 00355 00360 00364

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
VTFBLKSZ	00002	00000030	01968	00116
VTFCATLG	00001	00000036	01973	00240 00255 00258 00262 00265
VTFCREDIT	00003	0000001C	01959	00120
VTFDSN	00001	00000046	01985	00504
VTFDSNL	00002	00000044	01983	00111
VTFDSORG	00003	00000026	01963	00168 00172 00177 00182 00187 00191 00194
VTFDSTYP	00001	0000003A	01978	00129
VTFEXPDT	00003	0000001F	01960	00121
VTFRECL	00002	0000002E	01967	00115
VTFRLSTAC	00003	00000022	01961	00122
VTFMT	00001	00000000	01949	00105 01984
VTFMTL	00001	00000046	01984	00088
VTFNEXT	00004	00000000	01950	00107 00107
VTFNOEPV	00001	00000025	01962	00114
VTFPCT	00002	00000010	01956	00365
VTFPROT	00001	00000033	01970	00215 00217 00219
VTFRACF	00001	00000034	01971	00224 00227
VTFRECFM	00005	00000029	01965	00127 00130
VTFROUND	00001	00000032	01969	00205 00208
VTFSECAL	00001	00000039	01975	00203
VTFSECAM	00002	00000037	01974	00204
VTFUNIT	00004	00000018	01958	00109
VTFUNUSD	00004	0000000C	01955	00357
VTFUPD	00001	00000035	01972	00232 00235
VTFUSED	00004	00000008	01954	00338 00345 00353 00356 00362
VTFVOLUM	00006	00000012	01957	00108
VTOCFORM	00001	00000000	00049	00056 00559 00579 00591 00601 00612 00622 00633 00643 00657 00666 00677 00686 00697 00707 00716 00727 00737 00748 00758 00769 00783 00797 00808 00818 00828 00839 00849 00859 00870 00880 00889 00900 00910 00921 00931 00942 00952 00961 00972 00982 00993 01002 01013 01023 01034 01044 01050 01064 01071 01085 01092 01106 01113 01133 01140 01162 01169 01189 01209 01216 01236 01243 01263 01283 01290 01301 01308 01330 01337 01357 01380 01402 01409 01429 01452 01474 01481 01501 01524 01546 01553 01573 01596 01618 01625 01647 01654 01676 01689 01698 01707 01716
VTOCOM	00001	00000000	01777	00064 00475
XTCYLB	00001	00000081	02104	00290
XTDSECT	00001	00000000	02096	00287
XTFLAGS	00001	00000000	02097	00288 00290
XTHICC	00002	00000006	02108	00308
XTHIHH	00002	00000008	02109	00299 00311
XTLOWCC	00002	00000002	02106	00309
XTLOWHH	00002	00000004	02107	00295 00312
XTNOEXT	00001	00000000	02098	00288
ZEROES	00004	0000046C	00505	00331 00341

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
=F'0'	00004	00000518	02111	00360
=F'100'	00004	0000051C	02112	00363
=X'0008'	00002	00000520	02113	00189
=C'CYLERR'	00006	00000522	02114	00297 00303
=CL3'	00003	00000528	02115	00168
=CL3'IS'	00003	0000052B	02116	00172
=CL3'PS'	00003	0000052E	02117	00177
=CL3'DA'	00003	00000531	02118	00182
=CL3'PO'	00003	00000534	02119	00187
=CL3'VS'	00003	00000537	02120	00191

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 523

TOTAL RECORDS READ FROM SYSTEM LIBRARY 3411

TOTAL RECORDS PUNCHED 64

TOTAL RECORDS PRINTED 1216

SYMBOL	TYPE	ID	ADDR	LENGTH	LDID
--------	------	----	------	--------	------

ASM 0201 13.38 01/07/25

VTOCMSG	SD	0001	000000	0000E6	
---------	----	------	--------	--------	--

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				1 *		
				2 *	VTOC ERROR MESSAGE ROUTINE, R1 POINTS TO VTOC COMMON AT ENTRY	
				3 *		
				4	VTOCMSG ENTER 12,0 DO THE STANDARD HOUSEKEEPING	
000000				5+	VTOCMSG CSECT	
000000	47F0 F00C	0000C		6+	B 12(0,15) BRANCH AROUND ID 00450000	
000004	07			7+	DC AL1(7) LENGTH OF IDENTIFIER 00550000	
000005	E5E3D6C3D4E2C7			8+	DC CL7'VTOCMSG' IDENTIFIER 00750000	
00000C	90EC D00C	0000C		9+	STM 14,12,12(13) SAVE REGISTERS 03700000	
000010	18CF			10+	LR 12,15 SET FIRST BASE REG	
000012	0700			11+	CNOP 0,4	
			00000	12+	USING VTOCMSG,12	
000014	5811 0000	00000		13+	L 1,0+0(1) NUMERIC &SAVE IMPLIES A PASSED SAVEAREA	
000018	50D1 0004	00004		14+	ST 13,4(1) PRIOR SAVEAREA ADDRESS TO MINE	
00001C	501D 0008	00008		15+	ST 1,8(13) MY SAVEAREA ADDRESS TO HIS	
000020	182D			16+	LR 2,13 KEEP THE SAVEAREA ADDRESS FOR REGS	
000022	18D1			17+	LR 13,1 THIS IS MY SAVEAREA	
000024	9802 2014	00014		18+	LM 0,2,20(2) RESTORE ORIGINAL REGS	
000028	18B1			19	LR R11,R1 GET THE PARM REGISTER	
			00000	20	USING VTOCOM,R11 SET ADDRESSABILITY	
00002A	9801 B078	00078		22	LM R0,R1,MSGADDRS GET THE MESSAGE(S) TO SEND	
00002E	1200			23	LTR R0,R0 SECOND LEVEL MSG?	
000030	4780 C060	00060		24	BZ ERRORM1 NO	
000034	D27B B080 1000	00080 00000		26	MVC MSGTEXT1,0(R1) INSURE MSG IN WORK AREA	
00003A	4110 B080	00080		27	LA R1,MSGTEXT1	
00003E	48E1 0000	00000		29	LH R14,0(R1) LENGTH OF FIRST LEVEL MSG	
000042	41FE 1000	00000		30	LA R15,0(R14,R1) ADDR OF END OF MSG	
000046	41EE 0001	00001		31	LA R14,1(R14) JUMP MSG LENGTH	
00004A	40E1 0000	00000		32	STH R14,0(R1)	
00004E	924E F000	00000		33	MVI 0(R15),C'+' INDICATE SECOND LEVEL MSG EXISTS	
000052	1BEE			35	SR R14,R14 CLEAR CHAIN FIELD	
000054	41F0 0001	00001		36	LA R15,1 ONE SEGMENT IN 2ND MSG	
000058	90E0 B184	00184		37	STM R14,R0,PUTOLD2 CREATE SECOND-LEVEL	
				38 *	OUTPUT LINE DESCRIPTOR ('OLD')	
00005C	4100 B184	00184		39	LA R0,PUTOLD2	
000060	18E0			41	ERRORM1 LR R14,R0 NEXT 'OLD' ADDR OR ZERO	
000062	41F0 0001	00001		42	LA R15,1 ONE SEGMENT	
000066	1801			43	LR R0,R1 MSG ADDR	
000068	90E0 B178	00178		44	STM R14,R0,PUTOLD1 FIRST LEVEL 'OLD'	
00006C	4110 B04C	0004C		46	LA R1,PARMLIST	
			00000	47	USING IOPL,R1	
000070	D203 1004 B03C	00004 0003C		49	MVC IOPECT,ADDRECT	
000076	D203 1000 B038	00000 00038		50	MVC IOPLUPT,ADDRUPT	
00007C	4100 B06C	0006C		52	LA R0,ATTNECB	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
000080	5000 1008	00008		53	ST R0,IOPLECB	
000084	9200 B06C	0006C		54	MVI ATTNECB,0	
000088	D703 B05C B05C	0005C 0005C		56 57	XC PARMLIST+16(4),PARMLIST+16 PUTLINE PARM=PARMLIST+16,MF=(E,(1)), OUTPUT=(PUTOLD1,TERM,MULTLVL,INFOR)	X
00008E	41E0 B05C	0005C		58+	LA 14,PARMLIST+16	LOAD PARM ADDR 06500000
000092	50E0 100C	0000C		59+	ST 14,12(0,1)	STORE PARM ADDR IN PARM LIST 06550000
000096	9206 E000	00000		60+	MVI 0(14),B'00000110'	SET CONTROL FIELD 09450000
00009A	9200 E001	00001		61+	MVI 1(14),B'00000000'	SET CONTROL FIELD 09500000
00009E	4100 B178	00178		62+	LA 0,PUTOLD1	LOAD ADDR OF OUTPUT 09650000
0000A2	5000 E004	00004		63+	ST 0,4(0,14)	STORE ADDR OF OUTPUT 09700000
0000A6	58F0 0010	00010		64+	L 15,16(0,0)	LOAD CVT POINTER 12754002
0000AA	9180 F1BC	001BC		65+	TM 444(15),B'10000000'	IS PUTLINE LOADED ? 12766002
0000AE	47E0 C0BC	000BC		66+	BNO IKJ@0003	NO - BRANCH TO LINK 12770002
0000B2	58F0 F1BC	001BC		67+	L 15,444(0,15)	YES - BALR TO PUTLINE 12774002
0000B6	05EF			68+	BALR 14,15	12778002
0000B8	47F0 C0D2	000D2		69+	B IKJ\$0003	12782002
0000BC				70+IKJ@0003	DS 0H	30000000
0000BC				71+	CNOP 0,4	04900000
0000BC	45F0 C0D0	000D0		72+	BAL 15,*+20	BRANCH AROUND CONSTANTS 04950000
0000C0	000000C8			73+	DC A(*+8)	ADDR. OF PARM. LIST 05050000
0000C4	00000000			74+	DC A(0)	DCB ADDRESS PARAMETER 06650000
0000C8	C9D2D1D7E4E3D340			75+	DC CL8'IKJPUTL'	EP PARAMETER 06750000
0000D0	0A06			76+	SVC 6	ISSUE LINK SVC 48000000
0000D2				77+IKJ\$0003	DS 0H	12790002
0000D2	182D			79	LEAVE EQ	
0000D4	58DD 0004	00004		80+	LR 2,13	
0000D8	90F1 D010	00010		81+	L 13,4(13)	
0000DC	98EC D00C	0000C		82+	STM 15,1,16(13)	STORE RETURN REGS
0000E0	92FF D00C	0000C		83+	LM 14,12,12(13)	RESTORE THE REGISTERS 00650000
0000E4	07FE			84+	MVI 12(13),X'FF'	SET RETURN INDICATION 01600000
				85+	BR 14	RETURN 02000000
		00000		86+R0	EQU 0	*USED BY O.S.
		00001		87+R1	EQU 1	*USED BY O.S. // ADDRESS OF PARAMETER LIST
		00002		88+R2	EQU 2	
		00003		89+R3	EQU 3	
		00004		90+R4	EQU 4	
		00005		91+R5	EQU 5	
		00006		92+R6	EQU 6	
		00007		93+R7	EQU 7	
		00008		94+R8	EQU 8	
		00009		95+R9	EQU 9	
		0000A		96+R10	EQU 10	
		0000B		97+R11	EQU 11	
		0000C		98+R12	EQU 12	
		0000D		99+R13	EQU 13	*USED BY O.S. // SAVE-AREA ADDRESS
		0000E		100+R14	EQU 14	*USED BY O.S. // RETURN ADDRESS
		0000F		101+R15	EQU 15	*USED BY O.S. // ENTRY-PT ADDR, RETURN CODE

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				103	IKJIOPL	
000000				104+IOPL	DSECT	08000056
				105+*****	*****	10000056
				106+*	THE I/O SERVICE ROUTINE PARAMETER LIST (IOPL) IS A LIST OF	* 12000056
				107+*	FULLWORD ADDRESSES PASSED BY THE INVOKER OF ANY I/O SERVICE	* 14000056
				108+*	ROUTINE TO THE APPROPRIATE SERVICE ROUTINE VIA REGISTER ONE.	* 16000056
				109+*****	*****	18000056
000000				111+IOPLUPT	DS A PTR TO UPT	22000056
000004				112+IOPECT	DS A PTR TO ECT	24000056
000008				113+IOPLECB	DS A PTR TO USER'S ECB	26000056
00000C				114+IOPLIOPB	DS A PTR TO THE I/O SERVICE RTN PARM BLOCK	28000000
				115+**/		30000056
				117	VTOCOM	
				118+*		
				119+*	THIS IS THE VTOC COMMAND COMMON AREA	
				120+*		
000000				121+VTOCOM	DSECT	
				122+*		
				123+*	WORKING STORAGE AREAS FOR THE VARIOUS ROUTINES	
				124+*		
000000				125+VTCWMSG	DS A WORKING STORAGE FOR THE MSG ROUTINE	
000004				126+VTCWEXIT	DS A WORKING STORAGE FOR THE EXIT ROUTINE	
000008				127+VTCWEXCP	DS A WORKING STORAGE FOR THE EXCP ROUTINE	
00000C				128+VTCWCHEK	DS A WORKING STORAGE FOR THE CHEK ROUTINE	
000010				129+VTCWFORM	DS A WORKING STORAGE FOR THE FORM ROUTINE	
000014				130+VTCWPRNT	DS A WORKING STORAGE FOR THE PRNT ROUTINE	
000018				131+VTCWSORT	DS A WORKING STORAGE FOR THE SORT ROUTINE	
				132+*		
				133+*	ADDRESSES OF THE ROUTINES	
				134+*		
00001C	00000000			135+VADMSG	DC V(VTOCMSG) ADDRESS OF THE MESSAGE ROUTINE	
000020	00000000			136+VADEXIT	DC V(VTOCEXIT) ADDRESS OF THE EXIT ROUTINE	
000024	00000000			137+VADEXCP	DC V(VTOCEXIT) ADDRESS OF THE EXCP ROUTINE	
000028	00000000			138+VADCHEK	DC V(VTOCEXIT) ADDRESS OF THE CHECK ROUTINE	
00002C	00000000			139+VADFORM	DC V(VTOCEXIT) ADDRESS OF THE FORMAT ROUTINE	
000030	00000000			140+VADPRNT	DC V(VTOCEXIT) ADDRESS OF THE PRINT ROUTINE	
000034	00000000			141+VADSORT	DC V(VTOCEXIT) ADDRESS OF THE SORT ROUTINE	
				142+*		
				143+*	TSO COMMAND PROCESSOR AND PARSE DATA	
				144+*		
000038				145+ADDRUPT	DS A USER PROFILE TABLE	
00003C				146+ADDRECT	DS A ENVIRONMENT CONTROL TABLE	
000040				147+ADDRPSCB	DS A PROTECTED STEP CONTROL BLOCK	
000044				148+ADDRCBUF	DS A COMMAND BUFFER	
000048				150+ADDRANSR	DS A PARSE ANSWER OR PDL ADDRESS	
00004C				152+PARMLIST	DS 8A INTERNAL PARM AREA (MSG)	
00006C				153+ATTNECB	DS F ECB FOR ATTENTIONS	
000070				154+DOUBLE	DS D	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000078				156+	MSGADDRS DS	2A	ADDRESSES OF MESSAGES FOR VTOCMSG
000080				157+	MSGTEXT1 DS	XL124	
0000FC				158+	MSGTEXT2 DS	XL124	
000178				160+	PUTOLD1 DS	3F	
000184				161+	PUTOLD2 DS	3F	
				162+	*		
				163+	*	PARAMETER LIST FOR THE EXIT ROUTINE	
				164+	*		
000190				165+	EXITLIST DS	0F	
000190				166+	EXITAREA DS	A	WORKAREA LOCATION
000194				167+	DSCBADDR DS	A	ADDRESS OF THE DSCB
000198				168+	FORMATAD DS	A	ADDRESS OF THE FORMATTED DSCB
00019C				169+	CPPLADDR DS	A	ADDRESS OF THE CPPL
0001A0				170+	ACTIONAD DS	A	ADDRESS OF THE RECOMMENDED OR REQUESTED ACTION
				171+	*		
				172+	*	INTER ROUTINE FLAGS	
				173+	*		
0001A4				174+	VTCEFUNC DS	X	VTOCEXCP FUNCTION FLAG
0001A5				175+	VTCFMTCK DS	X	FORMAT IS CALLED BY CHECK RTN
		00080		176+	VTCFMTCD EQU	X'80'	FORMAT WAS CALLED BY CHECK
		00008		177+	VTCFMTCC EQU	X'08'	FORMAT WAS CALLED BY CHECK THIS CALL
				178+	*		
0001A6				179+	TABFULL DS	X	FLAG TABLES FULL, STOP INPUT
0001A7				180+	LOCAT DS	X	FLAG TO PERFORM CATALOG LOCATE
0001A8				181+	VTCEPRNT DS	X	PRINT END AND CLEANUP FLAG
0001AA				182+	DSNLEN DS	H	LENGTH OF THE DSNAME (NON-BLANKS)
0001AC				183+	ATABTITL DS	A	ADDRESS OF TABLE OF TITLES, LENGTHS
				184+	*		
				185+	*		
				186+	*		
				187+	*	WORKING STORAGE FOR VOLUME UCB SEARCH	
				188+	*		
0001B0				189+	ADDR DS	CL3	UCB ADDRESS IN CHARACTERS
0001B3				190+	VOLSER DS	CL6	VOLUME SERIAL NUMBER FROM PARSE
0001B9				191+	VOLID DS	CL6	CURRENT VOLUME SERIAL NUMBER TO PROCESS
0001BF				192+	FLAG DS	X	UCB SEARCH FLAG
0001C0				193+	LASTADR DS	F	LAST UCB ADDRESS FOUND (NO DUP'S)
0001C4				194+	UCBDEVT DS	CL4	PRINTABLE FORM OF DEVICE TYPE RPRINS
				195+	*		
				196+	*		
				197+	*		
0001C8				198+	SORTTAB DS	16F	
				199+	*		
				200+	*	EACH ENTRY CONTAINS A KEY OFFSET (2 BYTES) AND A KEY LENGTH (2 BYTES)	
				201+	*	THIS TABLE IS BUILT AT PARSE TIME ACCORDING TO THE SORT PARAMETERS	
				202+	*	SPECIFIED. THE 1ST PARM IS THE HIGH KEY AND SO ON.	
				203+	*		
				204+	*		
				205+	*		
				206+	*	ADDRESSES OF GETMAIN FOR FORMATTED DATA	
				207+	*		
000208				208+	VTCCURAD DS	A	CURRENT AVAILABLE ADDRESS

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
00020C				209+VTCCURLN	DS	A	CURRENT AVAILABLE LENGTH
000210				210+VTCGETMN	DS	50A	ADDRESSES OF BLOCKS
		00032		211+VTCGETMX	EQU	(* -VTCGETMN)/4	NUMBER OF BLOCKS MAXIMUM
		08000		212+VTCGETMS	EQU	32768	GETMAIN SIZE
				213+*			
				214+*		HASH SORT TABLE, POINTERS TO FIRST ENTRIES	
				215+*			
0002D8				216+VTCSORTH	DS	256A	POINT TO FORMATED ENTRIES
		006D8		217+VTCSORTE	EQU	*	END OF LIST
				218+*			
				219+*		PRINT ENTRIES - PAGE AND LINE COUNTERS	
				220+*			
0006D8				221+LINECT	DS	H	LINE COUNT
0006DA				222+LINEMAX	DS	H	MAXIMUM LINES PER PAGE
0006DC				223+PAGECT	DS	H	PAGE COUNT
0006DE				224+LINELEN	DS	H	LENGTH OF THE PRINT LINE
				225+*			
				226+*			
				227+*		VARIOUS ITEMS	
				228+*			
0006E0				229+FMT4	DS	XL44	SPACE FOR DSCB NAME
		0070C		230+IECSDSL4	EQU	*	FORMAT 4 DSCB
		0070C		231+IECSDSF4	EQU	IECSDSL4	16000000
00070C				232+DS4IDFMT	DS	CL1	FORMAT IDENTIFIER
00070D				233+DS4HPCHR	DS	XL5	HIGHEST ADDRESS OF A FORMAT 1 DSCB
000712				234+DS4DSREC	DS	XL2	NUMBER OF AVAILABLE DSCB'S
000714				235+DS4HCCHH	DS	XL4	CCHH OF NEXT AVAILABLE ALTERNATE TRK
000718				236+DS4NOATK	DS	XL2	NUMBER OF REMAINING ALTERNATE TRACKS
00071A				237+DS4VTOCI	DS	XL1	VTOC INDICATORS
		00080		238+DS4DOSBT	EQU	X'80'	DOS BIT
		00010		239+DS4DSTKP	EQU	X'10'	DOS STACKED PACK
		00008		240+DS4DOCVT	EQU	X'08'	DOS CONVERTED VTOC
		00004		241+DS4DIRF	EQU	X'04'	DIRF BIT
		00002		242+DS4DICVT	EQU	X'02'	DIRF RECLAIMED
		00001		243+DS4IVTOC	EQU	X'01'	VOLUME USES AN INDEXED VTOC @01A
00071B				244+DS4NOEXT	DS	XL1	NUMBER OF EXTENTS IN THE VTOC
00071C				245+	DS	XL2	RESERVED
00071E				246+DS4DEVCT	DS	0XL14	DEVICE CONSTANTS
00071E				247+DS4DEVSZ	DS	XL4	DEVICE SIZE
000722				248+DS4DEVTK	DS	XL2	DEVICE TRACK LENGTH
000724				249+DS4DEVOV	DS	0XL2	KEYED RECORD OVERHEAD
000724				250+DS4DEVI	DS	XL1	NON-LAST KEYED RECORD OVERHEAD
000725				251+DS4DEVL	DS	XL1	LAST KEYED RECORD OVERHEAD
000726				252+DS4DEVK	DS	XL1	NON-KEYED RECORD OVERHEAD
				253+*			DIFFERENTIAL
000727				254+DS4DEVFG	DS	XL1	FLAG BYTE
000728				255+DS4DEVTL	DS	XL2	DEVICE TOLERANCE
00072A				256+DS4DEVDT	DS	XL1	NUMBER OF DSCB'S PER TRACK
00072B				257+DS4DEVDB	DS	XL1	NUMBER OF DIRECTORY BLOCKS PER TRACK
00072C				258+DS4AMTIM	DS	XL8	VSAM TIME STAMP
000734				259+DS4AMCAT	DS	0XL3	VSAM CATALOG INDICATOR
000734				260+DS4VSIND	DS	XL1	VSAM INDICATORS
000735				261+DS4VSCRA	DS	XL2	RELATIVE TRACK LOCATION OF THE CRA
000737				262+DS4R2TIM	DS	XL8	VSAM VOLUME/CATALOG MATCH
				263+*			TIME STAMP

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38	01/07/25
00073F				264+	DS	XL5	RESERVED	17700000
000744				265+DS4F6PTR	DS	XL5	POINTER TO FIRST FORMAT 6 DSCB	17750000
000749				266+DS4VTOCE	DS	XL10	VTOC EXTENT DESCRIPTION	17800000
000753				267+	DS	XL25	RESERVED	17850000
			0076C	268+DS4END	EQU	*		17900000
000770				269+	DS	0D		
000770				270+FMT3	DS	0XL148	SPACE FOR FORMAT3 DSCB	
			00770	271+IECSDSL3	EQU	*	FORMAT 3 DSCB	15250000
			00770	272+IECSDSF3	EQU	IECSDSL3		15300000
000770				273+	DS	XL4	KEY IDENTIFIER	15350000
000774				274+DS3EXTNT	DS	XL40	FOUR EXTENT DESCRIPTIONS	15400000
				275+*		FIRST BYTE	EXTENT TYPE INDICATOR	15450000
				276+*		SECOND BYTE	EXTENT SEQUENCE NUMBER	15500000
				277+*		THIRD - SIXTH BYTES	LOWER LIMIT	15550000
				278+*		SEVENTH - TENTH BYTES	UPPER LIMIT	15600000
00079C				279+DS3FMTID	DS	CL1	FORMAT IDENTIFIER	15650000
00079D				280+DS3AEXT	DS	XL90	NINE ADDITIONAL EXTENT DESCRIPTIONS	15700000
0007F7				281+DS3PTRDS	DS	XL5	RESERVED	15750000
			007FC	282+DS3END	EQU	*		15800000
000800				283+	DS	0D		
				284		END		

POS.ID	REL.ID	FLAGS	ADDRESS
0001	0001	0C	0000C0

ASM 0201 13.38 01/07/25

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
ADDRECT	00004	0000003C	00146	00049
ADDRUPT	00004	00000038	00145	00050
ATTNECB	00004	0000006C	00153	00052 00054
ERRORM1	00002	00000060	00041	00024
IECSDSL3	00001	00000770	00271	00272
IECSDSL4	00001	0000070C	00230	00231
IKJ\$0003	00002	000000D2	00077	00069
IKJ@0003	00002	000000BC	00070	00066
IOPL	00001	00000000	00104	00047
IOPLECB	00004	00000008	00113	00053
IOPLECT	00004	00000004	00112	00049
IOPLUPT	00004	00000000	00111	00050
MSGADDRS	00004	00000078	00156	00022
MSGTEXT1	00124	00000080	00157	00026 00027
PARMLIST	00004	0000004C	00152	00046 00056 00056 00058
PUTOLD1	00004	00000178	00160	00044 00062
PUTOLD2	00004	00000184	00161	00037 00039
R0	00001	00000000	00086	00022 00023 00023 00037 00039 00041 00043 00044 00052 00053
R1	00001	00000001	00087	00019 00022 00026 00027 00029 00030 00032 00043 00046 00047
R11	00001	0000000B	00097	00019 00020
R14	00001	0000000E	00100	00029 00030 00031 00031 00032 00035 00035 00037 00041 00044
R15	00001	0000000F	00101	00030 00033 00036 00042
VTCGETMN	00004	00000210	00210	00211
VTOCMMSG	00001	00000000	00005	00012
VTOCOM	00001	00000000	00121	00020

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 51

TOTAL RECORDS READ FROM SYSTEM LIBRARY 2541

TOTAL RECORDS PUNCHED 8

TOTAL RECORDS PRINTED 343

SYMBOL	TYPE	ID	ADDR	LENGTH	LDID
VTOCPRNT	SD	0001	000000	000E13	
PCLMAIN	SD	0002	000E18	0007CE	

ASM 0201 13.38 01/07/25

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
2					*****	
3					*	*
4					*	*
5					* TITLE - VTOC COMMAND PRINT ROUTINE	*
6					*	*
7					* FUNCTION - PRINT THE DATA PASSED TO IT. IT WILL USE THE DDNAME	*
8					* SYSOUT IF IT IS ALLOCATED, AND WILL USE THE VTOCMSG	*
9					* ROUTINE IF NOT. IT CAN ALSO PUT OUT THE DATA TO	*
10					* AN OUTPUT DATA SET. TOTALS ARE COMPUTED HERE.	*
11					*	*
12					* OPERATION - FOR UNSORTED DATA, GET THE CURRENT ENTRY, ADD TO THE	*
13					* TOTALS, AND OUTPUT IT. IF THE DATA IS SORTED, THE	*
14					* ACTION IS MORE COMPLEX, BECAUSE ALL THE DATA SETS	*
15					* ARE TO BE OUTPUT. THE TOTALS AND BREAKS MAY BE	*
16					* NEEDED AT ANY POINT.	*
17					*	*
18					* INPUT - VTOC COMMON AREA (VTOCOM)	*
19					* POINTED TO BY REGISTER 1	*
20					* USE PARSE DATA, CURRENT FORMATTED DSCB, SORTED LIST	*
21					*	*
22					* OUTPUT - TO SYSOUT, A LIST OF THE REQUESTED DATA SETS AND	*
23					* THEIR ATTRIBUTES, WITH TOTALS AND BREAKS AS NEEDED.	*
24					* ALSO TO THE OUTPUT DATA SET, IF NEEDED. IF SYSOUT	*
25					* IS NOT ALLOCATED, VTOCMSG IS USED FOR OUTPUT.	*
26					*	*
27					* ATTRIBUTES - REENTRANT, REUSEABLE, REFRESHABLE.	*
28					*	*
29					*	*
30					* PROGRAMMED BY R. L. MILLER (415) 485-6241	*
31					*	*
32					* 12/19/97- MODIFIED BY JOHN KALINICH AT USA LSSC TO ADD YEAR Y2K	*
33					* 2000 SUPPORT FOR DATES. Y2K	*
34					*	*
35					*****	
36					*	
37					* MACRO FOR DEFINING FAKE PDE FOR A DEFAULT LIST	
38					*	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
				40	VTOCPRNT	ENTER 12,20	DO THE HOUSEKEEPING
000000				41+	VTOCPRNT	CSECT	
000000	47F0 F00E	0000E		42+	B	14(0,15)	BRANCH AROUND ID 00450000
000004	08			43+	DC	AL1(8)	LENGTH OF IDENTIFIER 00550000
000005	E5E3D6C3D7D9D5E3			44+	DC	CL8 'VTOCPRNT'	IDENTIFIER 00750000
00000D	00						
00000E	90EC D00C	0000C		45+	STM	14,12,12(13)	SAVE REGISTERS 03700000
000012	18CF			46+	LR	12,15	SET FIRST BASE REG
000014				47+	CNOP	0,4	
			00000	48+	USING	VTOCPRNT,12	
000014	5811 0014	00014		49+	L	1,20+0(1)	NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
000018	50D1 0004	00004		50+	ST	13,4(1)	PRIOR SAVEAREA ADDRESS TO MINE
00001C	501D 0008	00008		51+	ST	1,8(13)	MY SAVEAREA ADDRESS TO HIS
000020	182D			52+	LR	2,13	KEEP THE SAVEAREA ADDRESS FOR REGS
000022	18D1			53+	LR	13,1	THIS IS MY SAVEAREA
000024	9802 2014	00014		54+	LM	0,2,20(2)	RESTORE ORIGINAL REGS
000028	18B1			55	LR	R11,R1	SAVE ADDR OF VTOCOM
			00000	56	USING	VTOCOM,R11	SET ITS ADDRESSABILITY
00002A	5890 B048	00048		57	L	R9,ADDRANSR	POINT TO THE PARSE ANSWER
			00000	58	USING	PDL,R9	SET ITS ADDRESSABILITY
			00000	59	USING	PRNTWORK,R13	SET ADDRESSABILITY FOR LOCAL WORK AREA
				61 *			
				62 *		CHECK FOR THE PRINT CLEAN - CLOSE AND FREEMAIN	
				63 *			
00002E	9500 B1A8	001A8		64	CLI	VTCEPRNT,0	IS IT TIME
000032	4770 C198	00198		65	BNE	PRNTCLEN	YES, GO DO IT
				66 *			
				67 *		CHECK FOR THE FIRST TIME THROUGH	
				68 *		IF SO, SET UP THE DCB'S AND OPEN THEM	
				69 *			
000036	9500 D185	00185		70	CLI	FIRSTIM,0	IS THIS THE FIRST TIME?
00003A	4770 C042	00042		71	BNE	CHKSORT	NO, KEEP ON TRUCKIN'
00003E	47F0 C740	00740		72	B	PRTINIT	INITIALIZE FOR PRINTING
				73 *			
				74 *		CHECK TO SEE IF THE DATA IS SORTED	
				75 *			
000042	9502 902F	0002F		76	CHKSORT	CLI SORTK+1,2	IS THIS NOSORT?
000046	4770 C0A2	000A2		77	BNE	SORTED	NO, THE ENTRIES ARE SORTED
				78 *			
				79 *		NOSORT WAS SPECIFIED. ONLY THE CURRENT ENTRY IS AVAILABLE	
				80 *			
00004A	5830 B198	00198		81	L	R3,FORMATAD	POINT TO THE ENTRY
			00000	82	USING	VTFMT,R3	FORMATTED DSCB ADDRESSABILITY
00004E	1233			83	LTR	R3,R3	IS IT THERE?
000050	4780 C08E	0008E		84	BZ	CHEKTOT	NO, SEE IF A TOTAL HAS BEEN OUTPUT
000054	D703 B198 B198	00198 00198		85	XC	FORMATAD,FORMATAD	CLEAR THE ADDRESS FOR LATER
				86 *			
				87 *		ADD TO THE TOTALS	
				88 *			
00005A	4580 C208	00208		89	BAL	R8,ADDTOT	GO DO IT
				90 *			
				91 *		SEE IF THE OUTPUT DATA SET IS WANTED	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                ASM 0201 13.38 01/07/25
                                92 *
00005E 9110 D154    00154    93      TM      OUTDCB+48,X'10'  IS THE DCB OPEN?
000062 47E0 C06A    0006A    94      BNO     NOOUTPT      NO, SKIP ALONG
000066 4580 C572    00572    95      BAL     R8,OUTPUT     YES, GO DO IT
                                96 *
                                97 *      SEE IF THERE'S PRINTING TO DO
                                98 *
00006A 9502 903B    0003B    99 NOOUTPT CLI     PRINTK+1,2    WAS NOPRINT SPECIFIED?
00006E 4780 C076    00076   100      BE      VTRET          YES, WE'RE DONE FOR NOW
000072 4580 C2D8    002D8   101      BAL     R8,PRINT     NO, PRINT OUT THE ENTRY
                                102 *
                                103 *      RETURN FROM WHENCE WE CAME
                                104 *
000076 182D                105 VTRET  LEAVE  EQ,RC=0    EXEUNT
000078 58DD 0004    00004   106+VTRET LR     2,13
00007C 41F0 0000    00000   107+      L       13,4(13)
000080 90F1 D010    00010   108+      LA      15,0        LOAD THE RETURN CODE
000084 98EC D00C    0000C   109+      STM    15,1,16(13) STORE RETURN REGS
000088 92FF D00C    0000C   110+      LM     14,12,12(13) RESTORE THE REGISTERS 00650000
00008C 07FE                111+      MVI    12(13),X'FF' SET RETURN INDICATION 01600000
                                112+      BR     14          RETURN 02000000
                                00000 113+R0  EQU    0          *USED BY O.S.
                                00001 114+R1  EQU    1          *USED BY O.S. // ADDRESS OF PARAMETER LIST
                                00002 115+R2  EQU    2
                                00003 116+R3  EQU    3
                                00004 117+R4  EQU    4
                                00005 118+R5  EQU    5
                                00006 119+R6  EQU    6
                                00007 120+R7  EQU    7
                                00008 121+R8  EQU    8
                                00009 122+R9  EQU    9
                                0000A 123+R10 EQU   10
                                0000B 124+R11 EQU   11
                                0000C 125+R12 EQU   12
                                0000D 126+R13 EQU   13      *USED BY O.S. // SAVE-AREA ADDRESS
                                0000E 127+R14 EQU   14      *USED BY O.S. // RETURN ADDRESS
                                0000F 128+R15 EQU   15      *USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
                                129 *
                                130 *      NOSORT, CHECK FOR OUTPUTTING THE TOTALS
                                131 *
00008E 9180 D184    00184   132 CHEKTOT TM     ENDTOTAL,ENTOTOUT WAS THE TOTAL OUTPUT BEFORE?
000092 4710 C076    00076   133      BO     VTRET          YES, JUST RETURN
000096 4580 C22E    0022E   134      BAL     R8,PRNTOT    NO, OUTPUT THE TOTAL
00009A 9680 D184    00184   135      OI     ENDTOTAL,ENTOTOUT REMEMBER THE TOTAL IS OUT
00009E 47F0 C076    00076   136      B      VTRET          THEN RETURN

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

      138 *
      139 *      THE DATA IS SORTED, SO THE LISTS MUST BE EMPTIED.
      140 *      CHECK EACH ITEM FOR A TOTAL AND FOR A BREAK
      141 *      OUTPUT TO PRINT AND/OR THE OUTPUT DATA SET
      142 *
      143 *
      144 *      FIRST GET THE TOTAL AND BREAK COUNTS FOR COMPARES
      145 *
0000A2 4110 90E8      000E8      146 SORTED  LA      R1,TOTALN      POINT TO THE TOTAL PDL
0000A6 4580 C59E      0059E      147      BAL      R8,PDLNUM      CONVERT IT TO A NUMBER
0000AA 12FF          148      LTR      R15,R15      IS IT ZERO?
0000AC 4770 C0B4      000B4      149      BNZ      SETTNUM      NO, IT'S GOOD
0000B0 9610 D184      00184      150      OI      ENDTOTAL,ENDTONLY  YES, TOTALS AT THE END ONLY
0000B4 06F0          151 SETTNUM  BCTR    R15,0      CUT IT DOWN ONE
0000B6 40F0 D0C2      000C2      152      STH      R15,NUMTOTAL  SET THE TOTAL COUNT
0000BA 4110 90AC      000AC      153      LA      R1,BREAK      POINT TO THE BREAK PDL
0000BE 4580 C59E      0059E      154      BAL      R8,PDLNUM      CONVERT IT TO A NUMBER
0000C2 12FF          155      LTR      R15,R15      IS IT ZERO?
0000C4 4770 C0CC      000CC      156      BNZ      SETBNUM      NO, IT'S GOOD
0000C8 9608 D184      00184      157      OI      ENDTOTAL,NOBREAK  YES, BREAK AT THE END ONLY
0000CC 06F0          158 SETBNUM  BCTR    R15,0      CUT IT DOWN ONE
0000CE 40F0 D0C0      000C0      159      STH      R15,NUMBREAK  SET THE BREAK COUNT
      160 *
      161 *      START GOING THROUGH THE LISTS, PROCESS THE ENTRIES
      162 *
0000D2 4140 B2D8      002D8      163      LA      R4,VTCSORTH  POINT TO THE LISTS
0000D6 4150 B6D8      006D8      164      LA      R5,VTCSORTE  POINT TO THE END OF THE LISTS
0000DA 5050 D06C      0006C      165      ST      R5,ADDREND  SAVE THE ADDRESS
0000DE 5834 0000      00000      166 NEWLIST  L      R3,0(R4)      GET THE FIRST ENTRY FROM THIS LIST
0000E2 1233          167      LTR      R3,R3      ANYTHING ON THIS LIST?
0000E4 4780 C16C      0016C      168      BZ      NEXTLIST      NO, GET ANOTHER LIST
      169 *
      170 *      THIS IS AN ENTRY, DO THE TOTALS, PRINT, AND OUTPUT
      171 *      CHECK FOR TOTALS AND BREAKS FIRST
      172 *
0000E8 9502 903B      0003B      173 GOTENTRY CLI    PRINTK+1,2  WAS NOPRINT SET?
0000EC 4780 C154      00154      174      BE      CHKOUTPT  YES, SKIP INTERIM TOTALS
0000F0 9200 D187      00187      175      MVI     TOTLAST,0  SET UP FLAG FOR TOTAL AS LAST ACTION
      176 *      NOTE - EXTENSION - OUTPUT INTERIM TOTALS WITH THE KEY
0000F4 9110 D184      00184      177      TM      ENDTOTAL,ENDTONLY  END TOTAL ONLY?
0000F8 4710 C110      00110      178      BO      CHKBREAK      YES, SEE ABOUT BREAKS
0000FC 4820 D0C2      000C2      179      LH      R2,NUMTOTAL  GET THE LENGTH TO COMPARE
000100 5810 D0BC      000BC      180      L      R1,LASTKEY      GET THE LAST ENTRY
000104 4420 CAEC      00AEC      181      EX      R2,COMPKEY      SEE IF THIS IS THE SAME
000108 4780 C110      00110      182      BE      CHKBREAK      YES, KEEP COUNTING
      183 *
      184 *      THIS ONE IS DIFFERENT, PRINT THE TOTALS FIRST
      185 *
00010C 4580 C22E      0022E      186      BAL      R8,PRNTOT  PRINT THE TOTALS
      187 *
      188 *      CHECK FOR A BREAK
      189 *
000110 9108 D184      00184      190 CHKBREAK TM    ENDTOTAL,NOBREAK  NO BREAKS THIS TIME?
000114 4710 C13A      0013A      191      BO      SKPBREAK      YES, SKIP PAST BREAKS
000118 4820 D0C0      000C0      192      LH      R2,NUMBREAK  GET THE LENGTH TO COMPARE

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

00011C 5810 D0BC      000BC      193      L      R1, LASTKEY      GET THE LAST ENTRY
000120 4420 CAEC      00AEC      194      EX     R2, COMPKEY      SEE IF THIS IS THE SAME
000124 4780 C13A      0013A      195      BE     SKPBREAK       YES, KEEP COUNTING
196 *
197 *      THIS ONE IS DIFFERENT, GET A NEW PAGE
198 *
000128 D201 B6D8 B6DA 006D8 006DA 199      MVC     LINECT, LINEMAX BE SURE THE NEXT ITEM GETS A NEW PAGE
00012E 9200 D187      00187      200      MVI     TOTLAST, 0      DON'T SKIP A LINE AFTER TOTAL
000132 4110 3046      00046      201      LA     R1, VTFDSN      POINT TO THE DSNAME
000136 5010 D0BC      000BC      202      ST     R1, LASTKEY      SAVE THE ADDRESS FOR BREAK COMPARES
203 *
204 *      PRINT THE ITEM
205 *
00013A 9500 D187      00187      206 SKPBREAK CLI     TOTLAST, 0      WAS A TOTAL NOT FOLLOWED BY A BREAK?
00013E 4780 C150      00150      207      BE     SKPBREA2       NO, CONTINUE NORMALLY
000142 D213 D5AC CC90 005AC 00C90 208      MVC     MSGBL, MSGBLC  YES, SET UP A BLANK MESSAGE
000148 4110 D5AC      005AC      209      LA     R1, MSGBL       POINT TO IT
00014C 4580 C60C      0060C      210      BAL    R8, PRNTLINE    THEN PUTPUT IT
000150 4580 C2D8      002D8      211 SKPBREA2 BAL    R8, PRINT       FINAL FORMAT AND PRINT
212 *
213 *      CHECK FOR DATA SET OUTPUT
214 *
000154 9110 D154      00154      215 CHKOUTPT TM     OUTDCB+48, X'10'  IS IT OPEN AND READY
000158 47E0 C160      00160      216      BNO    GOTOT           NO, GO DO THE TOTALS
00015C 4580 C572      00572      217      BAL    R8, OUTPUT      YES, PUT OUT THE DATA SET ENTRY
218 *
219 *      ADD UP THE TOTALS
220 *
000160 4580 C208      00208      221 GOTOT  BAL    R8, ADDTOT  SUM THEM
222 *
223 *      GET THE NEXT ENTRY
224 *
000164 BF3F 3000      00000      225      ICM    R3, B'1111', VTFNEXT FOLLOW THE CHAIN
000168 4770 C0E8      000E8      226      BNZ    GOTENTRY       SOMETHING'S THERE, USE IT
227 *
228 *      END OF THIS LIST, TRY THE NEXT LIST
229 *
00016C 4140 400C      0000C      230 NEXTLIST LA     R4, 12(0, R4)    MOVE OVER ONE
000170 5940 D06C      0006C      231      C      R4, ADDREND         WAS THAT THE LAST LIST?
000174 4740 C0DE      000DE      232      BL     NEWLIST         NO, KEEP TRYING
233 *
234 *      END OF THE LISTS, OUTPUT THE FINAL TOTAL AND RETURN
235 *
000178 4580 C22E      0022E      236      BAL    R8, PRNTOT      LIST THE TOTAL
237 *
238 *      CLEAR OUT THE PRINT LISTS
239 *
00017C D7FF B2D8 B2D8 002D8 002D8 240      XC     VTCSORTH+000(256), VTCSORTH CLEAR 64 ENTRIES
000182 D7FF B3D8 B3D8 003D8 003D8 241      XC     VTCSORTH+256(256), VTCSORTH+256 CLEAR 64 ENTRIES
000188 D7FF B4D8 B4D8 004D8 004D8 242      XC     VTCSORTH+512(256), VTCSORTH+512 CLEAR 64 ENTRIES
00018E D7FF B5D8 B5D8 005D8 005D8 243      XC     VTCSORTH+768(256), VTCSORTH+768 CLEAR 64 ENTRIES
000194 47F0 C076      00076      244      B      VTRET           THEN GET OUT OF HERE

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				246 *		
				247 *	PRINT CLEANUP ROUTINE - CLOSE DCB'S FIRST	
				248 *		
000198	9110 D0F4	000F4		249	PRNTCLEN TM SYSOUT+48,X'10' IS SYSOUT OPEN?	
00019C	47E0 C1B6	001B6		250	BNO PRNTCLO NO, CHECK THE OUTDCB	
				251	CLOSE (SYSOUT),MF=(E,OPENLIST) DO THE CLOSE	
0001A0	4110 D074	00074		252+	LA 1,OPENLIST LOAD PARAMETER REG 1	01900002
0001A4	43E1 0000	00000		253+	IC 14,0(1,0) SAVE OPTION BYTE	01140000
0001A8	4100 D0C4	000C4		254+	LA 0,SYSOUT PICK UP DCB ADDRESS	01180000
0001AC	5001 0000	00000		255+	ST 0,0(1,0) STORE INTO LIST	01200000
0001B0	42E1 0000	00000		256+	STC 14,0(1,0) RESTORE OPTION BYTE	01220000
0001B4	0A14			257+	SVC 20 ISSUE CLOSE SVC	01640000
0001B6	9110 D154	00154		258	PRNTCLO TM OUTDCB+48,X'10' IS OUTDCB OPEN?	
0001BA	47E0 C1D4	001D4		259	BNO PRNTFREE NO, SKIP DOWN TO THE FREEMAINS	
				260	CLOSE (OUTDCB),MF=(E,OPENLIST) DO THE CLOSE	
0001BE	4110 D074	00074		261+	LA 1,OPENLIST LOAD PARAMETER REG 1	01900002
0001C2	43E1 0000	00000		262+	IC 14,0(1,0) SAVE OPTION BYTE	01140000
0001C6	4100 D124	00124		263+	LA 0,OUTDCB PICK UP DCB ADDRESS	01180000
0001CA	5001 0000	00000		264+	ST 0,0(1,0) STORE INTO LIST	01200000
0001CE	42E1 0000	00000		265+	STC 14,0(1,0) RESTORE OPTION BYTE	01220000
0001D2	0A14			266+	SVC 20 ISSUE CLOSE SVC	01640000
				267 *		
				268 *	FREE UP THE STORAGE	
				269 *		
0001D4	4120 B210	00210		270	PRNTFREE LA R2,VTCGETMN POINT TO THE TABLE	
0001D8	4150 0032	00032		271	LA R5,VTCGETMX GET THE NUMBER OF ENTRIES IN THE TABLE	
0001DC	BF3F 2000	00000		272	PRNTFRL ICM R3,B'1111',0(R2) GET THE STORAGE ADDRESS	
0001E0	4780 C076	00076		273	BZ VTRET IF ZERO, WE'RE DONE	
				274	FREEMAIN R,LV=VTCGETMS,A=(R3) FREE IT	
				275+*	OS/VS2 RELEASE 3 VERSION -- 10/25/74	00001603
0001E4				276+	CNOP 0,4	00144002
0001E4	47F0 C1EC	001EC		277+	B *+8 BRANCH AROUND LENGTH	00145002
0001E8	00008000			278+	DC A(VTCGETMS) LENGTH	00147802
0001EC	5800 C1E8	001E8		279+	L 0,*-4 LOAD SP AND LV	00148002
0001F0	4110 3000	00000		280+	LA 1,0(0,R3) LOAD AREA ADDRESS	00164002
0001F4	0A0A			281+	SVC 10 ISSUE FREEMAIN SVC	00311202
0001F6	D703 2000 2000	00000 00000		282	XC 0(4,R2),0(R2) CLEAR THE ADDRESS	
0001FC	4122 0004	00004		283	LA R2,4(R2) GET THE NEXT BLOCK ADDRESS	
000200	4650 C1DC	001DC		284	BCT R5,PRNTFRL AND LOOP UNTIL DONE	
000204	47F0 C076	00076		285	B VTRET THEN RETURN	
				286 *		

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT ASM 0201 13.38 01/07/25

288 *
289 * ROUTINES USED ABOVE
290 * ADDTOT - ADD TO THE CURRENT TOTALS
291 * PRNTOT - PRINT OUT THE TOTALS AND CLEAR THEM
292 * PRINT - PRINT OUT AN ENTRY
293 * OUTPUT - OUTPUT THE DATA SET ENTRY
294 * PDLNUM - GET A NUMBER FROM A PDL ENTRY
295 * PRNTLINE - INTERNAL ROUTINE TO COUNT LINES, OUTPUT HEAD
296 *
000208 5810 D048 00048 297 ADDTOT L R1,TOTDS NUMBER OF DATA SETS
00020C 4111 0001 00001 298 LA R1,1(R1) ADD ONE
000210 5010 D048 00048 299 ST R1,TOTDS STORE IT BACK
000214 5810 D04C 0004C 300 L R1,TOTALLOC TOTAL ALLOCATION
000218 5A10 3004 00004 301 A R1,VTFALLOC ADD IN THIS DATA SET
00021C 5010 D04C 0004C 302 ST R1,TOTALLOC STORE IT BACK
000220 5810 D050 00050 303 L R1,TOTUSED TOTAL USED SPACE
000224 5A10 3008 00008 304 A R1,VTFUSED ADD IN THIS DATA SET
000228 5010 D050 00050 305 ST R1,TOTUSED STORE IT BACK
00022C 07F8 306 BR R8 RETURN
307 *

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                ASM 0201 13.38 01/07/25
                                309 *
                                310 *      PRINT THE TOTALS
                                311 *      FIRST FORMAT THEM, THEN PUT THE LINE OUT TO SYSOUT
                                312 *      OR USE VTOCMSG TO LIST IT
                                313 *
00022E 5080 D068      00068      314 PRNTOT  ST    R8,PRNTTOT8  SAVE THE RETURN ADDRESS
000232 D24B D1A2  CC44 001A2 00C44 315      MVC    MSGWORK(MSGTLEN),MSGTOTC  INIT THE MSG
                                316      CONV  MSGWORK+4+11,TOTDS,5  CONVERT NO OF DATA SETS
000238 5810 D048      00048      317+     L     R1,TOTDS      GET THE DATA TO CONVERT
00023C 4E10 B070      00070      318+     CVD   R1,DOUBLE     CONVERT TO PACKED DECIMAL
000240 D20F D192  CB08 00192 00B08 319+     MVC   CHARS,EDMASK  PUT IN THE EDIT MASK
000246 DE0F D192  B070 00192 00070 320+     ED    CHARS,DOUBLE  CONVERT TO CHARACTERS
00024C D204 D1B1  D19D 001B1 0019D 321+     MVC   MSGWORK+4+11(5),CHARS+16-5  MOVE IN THE NUMBER
000252 D50A D192  CB39 00192 00B39 322+     CLC   CHARS(16-5),BLANKS  WAS THERE AN OVERFLOW?
000258 4780 C262      00262      323+     BE    *+10         NO, EVERYTHING WAS OK
00025C D204 D1B1  CB49 001B1 00B49 324+     MVC   MSGWORK+4+11(5),STARS BAD NEWS, NOTE IT
                                325     CONV  MSGWORK+4+28,TOTALLOC,8  CONVERT ALLOCATION
000262 5810 D04C      0004C      326+     L     R1,TOTALLOC   GET THE DATA TO CONVERT
000266 4E10 B070      00070      327+     CVD   R1,DOUBLE     CONVERT TO PACKED DECIMAL
00026A D20F D192  CB08 00192 00B08 328+     MVC   CHARS,EDMASK  PUT IN THE EDIT MASK
000270 DE0F D192  B070 00192 00070 329+     ED    CHARS,DOUBLE  CONVERT TO CHARACTERS
000276 D207 D1C2  D19A 001C2 0019A 330+     MVC   MSGWORK+4+28(8),CHARS+16-8  MOVE IN THE NUMBER
00027C D507 D192  CB39 00192 00B39 331+     CLC   CHARS(16-8),BLANKS  WAS THERE AN OVERFLOW?
000282 4780 C28C      0028C      332+     BE    *+10         NO, EVERYTHING WAS OK
000286 D207 D1C2  CB49 001C2 00B49 333+     MVC   MSGWORK+4+28(8),STARS BAD NEWS, NOTE IT
                                334     CONV  MSGWORK+4+51,TOTUSED,8   CONVERT USED SPACE
00028C 5810 D050      00050      335+     L     R1,TOTUSED   GET THE DATA TO CONVERT
000290 4E10 B070      00070      336+     CVD   R1,DOUBLE     CONVERT TO PACKED DECIMAL
000294 D20F D192  CB08 00192 00B08 337+     MVC   CHARS,EDMASK  PUT IN THE EDIT MASK
00029A DE0F D192  B070 00192 00070 338+     ED    CHARS,DOUBLE  CONVERT TO CHARACTERS
0002A0 D207 D1D9  D19A 001D9 0019A 339+     MVC   MSGWORK+4+51(8),CHARS+16-8  MOVE IN THE NUMBER
0002A6 D507 D192  CB39 00192 00B39 340+     CLC   CHARS(16-8),BLANKS  WAS THERE AN OVERFLOW?
0002AC 4780 C2B6      002B6      341+     BE    *+10         NO, EVERYTHING WAS OK
0002B0 D207 D1D9  CB49 001D9 00B49 342+     MVC   MSGWORK+4+51(8),STARS BAD NEWS, NOTE IT
0002B6 D205 D1CB  D188 001CB 00188 343     MVC   MSGWORK+4+37(6),SPACTYPE MOVE IN THE UNITS
0002BC D205 D1E2  D188 001E2 00188 344     MVC   MSGWORK+4+60(6),SPACTYPE MOVE IN THE UNITS
                                345 *
                                346 *      NOW OUTPUT THE MESSAGE
                                347 *
0002C2 92F0 D1A6      001A6      348     MVI   MSGWORK+4,C'0'  ADD A CARRIAGE CONTROL
0002C6 4110 D1A2      001A2      349     LA    R1,MSGWORK     POINT TO THE TOTAL LINE
0002CA 4580 C60C      0060C      350     BAL   R8,PRNTLINE    PUT OUT THE TOTAL LINE
0002CE 9201 D187      00187      351     MVI   TOTLAST,1     NOTE THAT A TOTAL WAS THE LAST ITEM
0002D2 5880 D068      00068      352     L     R8,PRNTTOT8    GET THE RETURN ADDRESS
0002D6 07F8              353     BR    R8             THEN RETURN

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				355	*	
				356	*	PRINT OUT THE FORMATTED DSCB
				357	*	FIRST FORMAT IT
				358	*	
0002D8	5080 D064	00064		359	PRINT ST R8,PRINTR8	SAVE REGISTER 8 FOR RETURNING
0002DC	9240 D2A6	002A6		360	MVI WORKLINE+4,C' '	BLANK OUT THE LINE
0002E0	D2F9 D2A7 D2A6	002A7 002A6		361	MVC WORKLINE+5(250),WORKLINE+4	SO INDIVIDUAL FIELDS DON'T
0002E6	5820 D774	00774		362	L R2,VTPRNTLS	GET THE PRINT ITEM LIST
0002EA	4110 D2A6	002A6		363	LA R1,WORKLINE+4	POINT TO THE WORK LINE
0002EE	1B66			364	PRTLOOP SR R6,R6	GET THE RESERVED WORD NUMBER
0002F0	4362 0000	00000		365	IC R6,0(R2)	FROM THE TOP BYTE
0002F4	4C60 CAE0	00AE0		366	MH R6,H12	MULTIPLY BY 12 FOR THE TABLE ENTRIES
0002F8	4166 CCCC	00CCC		367	LA R6,TABTITL(R6)	THEN RELOCATE THE MESS
0002FC	1B77			368	SR R7,R7	CLEAR A REGISTER
0002FE	4376 0000	00000		369	IC R7,0(R6)	GET THE EXECUTE LENGTH
000302	4151 0001	00001		370	LA R5,1(R1)	SAVE A PLACE TO MOVE FIELD INTO
000306	4111 7001	00001		371	LA R1,1(R1,R7)	MOVE THE POINTER OVER
00030A	4100 D2A6	002A6		372	LA R0,WORKLINE+4	POINT TO THE BEGINNING AGAIN
00030E	1B10			373	SR R1,R0	AND FIND THE CURRENT LENGTH
000310	4910 B6DE	006DE		374	CH R1,LINELEN	IS IT TOO LONG?
000314	47B0 C34C	0034C		375	BNL PRTEND	YES, WE'RE DONE
000318	1A10			376	AR R1,R0	NO, KEEP GOING
				377	*	
				378	*	MOVE IN OR CONVERT THIS ITEM
				379	*	
00031A	1BEE			380	SR R14,R14	CLEAR A REG FOR LENGTH OF RTN NO
00031C	43E6 0001	00001		381	IC R14,1(R6)	GET THE LENGTH OR ROUTINE
000320	54E0 CAD0	00AD0		382	N R14,F127	CLEAR THE TOP BIT
000324	1BFF			383	SR R15,R15	CLEAR A REG FOR VTFMT
000326	43F6 0002	00002		384	IC R15,2(R6)	DISPLACEMENT
00032A	1AF3			385	AR R15,R3	RELOCATE IT
				386	*	
				387	*	DECIDE WHERE TO PUT IT
				388	*	
00032C	9180 6001	00001		389	TM 1(R6),X'80'	IS IT IN CHARS
000330	47E0 C360	00360		390	BNO PRTRTN	NO, USE THE SPECIAL ROUTINE
000334	1B7E			391	SR R7,R14	GET THE DIFFERENCE IN LENGTHS
000336	8870 0001	00001		392	SRL R7,1	GET HALF THE DIFFERENCE
00033A	1A57			393	AR R5,R7	PUT THE DATA HERE
00033C	44E0 C35A	0035A		394	EX R14,PRTMOVE	MOVE IN THE CHARACTERS
000340	4122 0001	00001		395	PRTINC LA R2,1(R2)	GET THE NEXT CHAIN POINTER
000344	9500 2000	00000		396	CLI 0(R2),0	ARE WE DONE?
000348	4770 C2EE	002EE		397	BNE PRTLOOP	GO GET MORE ITEMS
00034C	4110 D2A2	002A2		398	PRTEND LA R1,WORKLINE	POINT TO THIS LINE
000350	4580 C60C	0060C		399	BAL R8,PRNTLINE	THEN GO PRINT IT
000354	5880 D064	00064		400	L R8,PRINTR8	GET THE RETURN ADDRESS
000358	07F8			401	BR R8	RETURN
00035A	D200 5000 F000	00000 00000		402	PRTMOVE MVC 0(0,R5),0(R15)	EXECUTED MOVE
				403	*	
				404	*	VARIOUS ROUTINES TO FORMAT BEFORE MOVING IN THE DATA
				405	*	
000360	12EE			406	PRTRTN LTR R14,R14	BE SURE THE OFFSET IS OK
000362	4720 C376	00376		407	BP PRTRTN1	IT SEEMS ALL RIGHT
				408	ABEND 701,DUMP	CRASH AND BURN
000366				409+	DS 0H	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000366	4110 02BD	002BD		410+	LA	1,701	LOAD PARAMETER REG 1 01900002
00036A	4100 0080	00080		411+	LA	0,128(0,0)	PICK UP DUMP/STEP/DUMPOPTS YM1995 01800002
00036E	8900 0018	00018		412+	SLL	0,24(0)	SHIFT TO HIGH ORDER 01850002
000372	1610			413+	OR	1,0	OR IN WITH COMPCODE 01900002
000374	0A0D			414+	SVC	13	LINK TO ABEND ROUTINE 02050002
000376	47FE C376	00376		415	PRTRTN1 B	*(R14)	GO DO YOUR THING
00037A	47F0 C39A	0039A		416	B	PRDATES	4 CDATE, EXPDT
00037E	47F0 C3F4	003F4		417	B	PRLSTUS	8 LSTUS
000382	47F0 C40A	0040A		418	B	PRALLOC	12 ALLOC, USED
000386	47F0 C446	00446		419	B	PRUNUSED	16 ALLOC - USED
00038A	47F0 C484	00484		420	B	PRPCT	20 100 * USED / ALLOC
00038E	47F0 C4F6	004F6		421	B	PREXT	24 EXT
000392	47F0 C526	00526		422	B	PRBLREC	28 BLKSZ LRECL
000396	47F0 C554	00554		423	B	PRDSN	32 DSNAME
				424 *			
				425 *			
				426 *		CDATE AND EXPDT	
				427 *			
00039A	1BEE			428	PRDATES SR	R14,R14	CLEAR REG FOR YEAR
00039C	43EF 0000	00000		429	IC	R14,0(R15)	GET THE YEAR
		00000		430		USING DUMMD,R5	ALLOW CONV TO USE SYMBOL
				431		CONVY2K DUMMA,(R14),2	CONVERT YEAR Y2K DEC97
0003A0	4EE0 B070	00070		432+	CVD	R14,DOUBLE	CONVERT TO PACKED DECIMAL
0003A4	D20F D192 CB08	00192 00B08		433+	MVC	CHARS,EDMASK	PUT IN THE EDIT MASK
0003AA	DE0F D192 B070	00192 00070		434+	ED	CHARS,DOUBLE	CONVERT TO CHARACTERS
0003B0	D201 5000 D1A0	00000 001A0		435+	MVC	DUMMA(2),CHARS+16-2	MOVE IN THE NUMBER
0003B6	D50D D192 CB39	00192 00B39		436+	CLC	CHARS(16-2),BLANKS	WAS THERE AN OVERFLOW?
				437+*	BE	*+10	NO, EVERYTHING WAS OK
0003BC	47F0 C3C6	003C6		438+	B	*+10	ALLOW YY > 99 FOR YEAR 2000 Y2K DEC97
0003C0	D201 5000 CB49	00000 00B49		439+	MVC	DUMMA(2),STARS	BAD NEWS, NOTE IT
0003C6	BFE3 F001	00001		440	ICM	R14,B'0011',1(R15)	GET THE DAY
				441	CONV	DUMMA+2,(R14),3,EDMASK0,COMP0	CONVERT THE DAY
0003CA	4EE0 B070	00070		442+	CVD	R14,DOUBLE	CONVERT TO PACKED DECIMAL
0003CE	D20F D192 CB18	00192 00B18		443+	MVC	CHARS,EDMASK0	PUT IN THE EDIT MASK
0003D4	DE0F D192 B070	00192 00070		444+	ED	CHARS,DOUBLE	CONVERT TO CHARACTERS
0003DA	D202 5002 D19F	00002 0019F		445+	MVC	DUMMA+2(3),CHARS+16-3	MOVE IN THE NUMBER
0003E0	D50C D192 CB29	00192 00B29		446+	CLC	CHARS(16-3),COMP0	WAS THERE AN OVERFLOW?
0003E6	4780 C3F0	003F0		447+	BE	*+10	NO, EVERYTHING WAS OK
0003EA	D202 5002 CB49	00002 00B49		448+	MVC	DUMMA+2(3),STARS	BAD NEWS, NOTE IT
0003F0	47F0 C340	00340		449	B	PRTINC	GO GET MORE PRINT ITEMS
				450 *			
				451 *		LAST USE DATE	
				452 *			
0003F4	D20F D192 CB08	00192 00B08		453	PRLSTUS MVC	CHARS,EDMASK	SET UP THE EDIT MASK
0003FA	DE05 D192 F000	00192 00000		454	ED	CHARS(6),0(R15)	CONVERT TO CHARACTERS
000400	D204 5000 D193	00000 00193		455	MVC	0(5,R5),CHARS+1	THEN MOVE THEM IN
000406	47F0 C340	00340		456	B	PRTINC	GO GET MORE PRINT ITEMS
				457 *			
				458 *		ALLOCATION AND USED	
				459 *			
00040A	BFEF F000	00000		460	PRALLOC ICM	R14,B'1111',0(R15)	GET THE AMOUNT
00040E	4740 C43C	0043C		461	BM	PRUNKN6	IF NEGATIVE, IT'S NOT KNOWN
				462	CONV	DUMMA,(R14),6	CONVERT THE NUMBER
000412	4EE0 B070	00070		463+	CVD	R14,DOUBLE	CONVERT TO PACKED DECIMAL
000416	D20F D192 CB08	00192 00B08		464+	MVC	CHARS,EDMASK	PUT IN THE EDIT MASK

```

ASM 0201 13.38 01/07/25

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
00041C DE0F D192 B070 00192 00070 465+ ED CHARS,DOUBLE CONVERT TO CHARACTERS
000422 D205 5000 D19C 00000 0019C 466+ MVC DUMMA(6),CHARS+16-6 MOVE IN THE NUMBER
000428 D509 D192 CB39 00192 00B39 467+ CLC CHARS(16-6),BLANKS WAS THERE AN OVERFLOW?
00042E 4780 C438 00438 468+ BE *+10 NO, EVERYTHING WAS OK
000432 D205 5000 CB49 00000 00B49 469+ MVC DUMMA(6),STARS BAD NEWS, NOTE IT
000438 47F0 C340 00340 470 B PRTINC GO GET MORE PRINT ITEMS
00043C D205 5000 CB39 00000 00B39 471 PRUNKN6 MVC 0(6,R5),BLANKS UNKNOWN AMOUNT, LEAVE IT BLANK
000442 47F0 C340 00340 472 B PRTINC GO GET MORE PRINT ITEMS
473 *
474 * UNUSED SPACE
475 *
000446 BFEF 3008 00008 476 PRUNUSED ICM R14,B'1111',VTFUSED GET THE USED SPACE
00044A 4740 C43C 0043C 477 BM PRUNKN6 IF NOT KNOWN, SKIP ALONG
00044E 58E0 3004 00004 478 L R14,VTFALLOC GET THE ALLOCATED SPACE
000452 5BE0 3008 00008 479 S R14,VTFUSED MINUS THE USED SPACE GIVES UNUSED
000456 4780 C340 00340 480 BZ PRTINC IF NO UNUSED SPACE, LEAVE BLANK
481 CONV DUMMA,(R14),6 CONVERT FOR PRINTING
00045A 4EE0 B070 00070 482+ CVD R14,DOUBLE CONVERT TO PACKED DECIMAL
00045E D20F D192 CB08 00192 00B08 483+ MVC CHARS,EDMASK PUT IN THE EDIT MASK
000464 DE0F D192 B070 00192 00070 484+ ED CHARS,DOUBLE CONVERT TO CHARACTERS
00046A D205 5000 D19C 00000 0019C 485+ MVC DUMMA(6),CHARS+16-6 MOVE IN THE NUMBER
000470 D509 D192 CB39 00192 00B39 486+ CLC CHARS(16-6),BLANKS WAS THERE AN OVERFLOW?
000476 4780 C480 00480 487+ BE *+10 NO, EVERYTHING WAS OK
00047A D205 5000 CB49 00000 00B49 488+ MVC DUMMA(6),STARS BAD NEWS, NOTE IT
000480 47F0 C340 00340 489 B PRTINC GO GET MORE PRINT ITEMS
490 *
491 * PCT - PERCENTAGE USED
492 *
000484 1B66 493 PRPCT SR R6,R6 CLEAR THE TOP PART OF THE NUMBER
000486 BF7F 3008 00008 494 ICM R7,B'1111',VTFUSED GET THE AMOUNT OF USED SPACE
00048A 4740 C4EC 004EC 495 BM PRUNKN3 IF NOT KNOWN, LEAVE BLANK
00048E D503 3004 CAC8 00004 00AC8 496 CLC VTFALLOC,ZERO IS THE ALLOCATION ZERO?
000494 4770 C4AC 004AC 497 BNE PRPCTM NO, DO THE STANDARD STUFF
000498 D503 3008 CAC8 00008 00AC8 498 CLC VTFUSED,ZERO IS THE USED SPACE ZERO?
00049E 4780 C4EC 004EC 499 BE PRUNKN3 YES, JUST USE BLANKS
0004A2 D202 5000 CE10 00000 00E10 500 PRPCTERR MVC 0(3,R5),=C'ERR' NO, IT'S AN ERROR
0004A8 47F0 C340 00340 501 B PRTINC THEN GO TRY FOR MORE
0004AC 5C60 CACC 00ACC 502 PRPCTM M R6,F100 MULTIPLY BY 100 FOR PERCENT
0004B0 5D60 3004 00004 503 D R6,VTFALLOC DIVIDE BY ALLOC TO GET THE PERCENT
0004B4 1277 504 LTR R7,R7 IS IT LESS THAN ZERO?
0004B6 4740 C4A2 004A2 505 BM PRPCTERR YES, FLAG THE ERROR
0004BA 5970 CACC 00ACC 506 C R7,F100 ALSO CHECK FOR OVER 100 PERCENT
0004BE 4720 C4A2 004A2 507 BH PRPCTERR THAT'S ALSO AN ERROR
508 CONV DUMMA,(R7),3 CONVERT FOR PRINTING
0004C2 4E70 B070 00070 509+ CVD R7,DOUBLE CONVERT TO PACKED DECIMAL
0004C6 D20F D192 CB08 00192 00B08 510+ MVC CHARS,EDMASK PUT IN THE EDIT MASK
0004CC DE0F D192 B070 00192 00070 511+ ED CHARS,DOUBLE CONVERT TO CHARACTERS
0004D2 D202 5000 D19F 00000 0019F 512+ MVC DUMMA(3),CHARS+16-3 MOVE IN THE NUMBER
0004D8 D50C D192 CB39 00192 00B39 513+ CLC CHARS(16-3),BLANKS WAS THERE AN OVERFLOW?
0004DE 4780 C4E8 004E8 514+ BE *+10 NO, EVERYTHING WAS OK
0004E2 D202 5000 CB49 00000 00B49 515+ MVC DUMMA(3),STARS BAD NEWS, NOTE IT
0004E8 47F0 C340 00340 516 B PRTINC GO GET MORE PRINT ITEMS
0004EC D202 5000 CB39 00000 00B39 517 PRUNKN3 MVC 0(3,R5),BLANKS BLANK THE UNKNOWN
0004F2 47F0 C340 00340 518 B PRTINC GO GET MORE PRINT ITEMS
519 *

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				520 *	EXTENTS	
				521 *		
0004F6	1BEE			522	PREXT SR R14,R14 CLEAR REG FOR YEAR	
0004F8	43EF 0000	00000		523	IC R14,0(R15) GET THE YEAR	
				524	CONV DUMMA,(R14),2 CONVERT THE EXTENTS	
0004FC	4EE0 B070	00070		525+	CVD R14,DOUBLE CONVERT TO PACKED DECIMAL	
000500	D20F D192 CB08	00192 00B08		526+	MVC CHARS,EDMASK PUT IN THE EDIT MASK	
000506	DE0F D192 B070	00192 00070		527+	ED CHARS,DOUBLE CONVERT TO CHARACTERS	
00050C	D201 5000 D1A0	00000 001A0		528+	MVC DUMMA(2),CHARS+16-2 MOVE IN THE NUMBER	
000512	D50D D192 CB39	00192 00B39		529+	CLC CHARS(16-2),BLANKS WAS THERE AN OVERFLOW?	
000518	4780 C522	00522		530+	BE *+10 NO, EVERYTHING WAS OK	
00051C	D201 5000 CB49	00000 00B49		531+	MVC DUMMA(2),STARS BAD NEWS, NOTE IT	
000522	47F0 C340	00340		532	B PRTINC GO GET MORE PRINT ITEMS	
				533 *		
				534 *	LRECL AND BLKSZ	
				535 *		
000526	48EF 0000	00000		536	PRBLREC LH R14,0(R15) GET THE DATA (HALFWORD)	
				537	CONV DUMMA,(R14),5 CONVERT THE DATA	
00052A	4EE0 B070	00070		538+	CVD R14,DOUBLE CONVERT TO PACKED DECIMAL	
00052E	D20F D192 CB08	00192 00B08		539+	MVC CHARS,EDMASK PUT IN THE EDIT MASK	
000534	DE0F D192 B070	00192 00070		540+	ED CHARS,DOUBLE CONVERT TO CHARACTERS	
00053A	D204 5000 D19D	00000 0019D		541+	MVC DUMMA(5),CHARS+16-5 MOVE IN THE NUMBER	
000540	D50A D192 CB39	00192 00B39		542+	CLC CHARS(16-5),BLANKS WAS THERE AN OVERFLOW?	
000546	4780 C550	00550		543+	BE *+10 NO, EVERYTHING WAS OK	
00054A	D204 5000 CB49	00000 00B49		544+	MVC DUMMA(5),STARS BAD NEWS, NOTE IT	
000550	47F0 C340	00340		545	B PRTINC GO GET MORE PRINT ITEMS	
				546 *		
				547 *	DATA SET NAME	
				548 *		
000554	48E0 3044	00044		549	PRDSN LH R14,VTFDSNL GET THE DSNAME LENGTH	
000558	49E0 D18E	0018E		550	CH R14,DSNLENGT CHACK FOR MAX LENGTH	
00055C	4740 C564	00564		551	BL PRDMOVE THIS DSN IS SHORT ENOUGH	
000560	48E0 D18E	0018E		552	LH R14,DSNLENGT CUT IT DOWN	
000564	06E0			553	PRDMOVE BCTR R14,0 COUNT DOWN ONE FOR THE EX	
000566	44E0 C35A	0035A		554	EX R14,PRTMOVE MOVE IN THE DSNAME	
00056A	4A10 D190	00190		555	AH R1,DSNLENOF CORRECT THE LINE POINTER	
00056E	47F0 C340	00340		556	B PRTINC GO GET MORE PRINT ITEMS	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				558 *		
				559 *	OUTPUT THE FORMATTED DSCB AS IT IS	
				560 *		
				561 OUTPUT	CLEAR OUTWORK CLEAR THE OUTPUT REC	
000572	9240 D5C0	005C0		562+OUTPUT	MVI OUTWORK,X'40' SET THE FIRST POSITION	
000576	D2FE D5C1 D5C0	005C1 005C0		563+	MVC OUTWORK+1(L'OUTWORK-1),OUTWORK FILL THE ENTIRE FIELD	
00057C	4820 3044	00044		564	LH R2,VTFDSNL GET THE LENGTH OF THE DSNAME	
000580	4122 0045	00045		565	LA R2,VTFMTL-1(R2) GET THE FULL LENGTH MINUS ONE	
000584	4420 C598	00598		566	EX R2,OUTMOVE MOVE IN THE ACTUAL RECORD	
				567	PUT OUTDCB,OUTWORK OUTPUT THE RECORD	
000588	4110 D124	00124		568+	LA 1,OUTDCB LOAD PARAMETER REG 1 01900002	
00058C	4100 D5C0	005C0		569+	LA 0,OUTWORK LOAD PARAMETER REG 0 02500002	
000590	58F0 1030	00030		570+	L 15,48(0,1) LOAD PUT ROUTINE ADDR 00550000	
000594	05EF			571+	BALR 14,15 LINK TO PUT ROUTINE 00600000	
000596	07F8			572	BR R8 RETURN	
000598	D200 D5C0 3000	005C0 00000		573 OUTMOVE	MVC OUTWORK(0),0(R3) EXECUTED MOVE	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				575	*	
				576	*	PDLNUM - CONVERT FROM CHARACTERS (EBCDIC) TO AN INTEGER
				577	*	BINARY FORM, PASSED BACK VIA REGISTER 15
				578	*	A PARSE PDE IS THE INPUT AS SHOWN IN THE SAMPLE BELOW
				579	*	LA R1,PDL POINT TO THE PARSE DESCRIPTION
				580	*	BAL R8,PDLNUM GO CONVERT TO NUMERICS
				581	*	THE ROUTINE WILL TERMINATE IF IT FINDS NON-NUMERICS
				582	*	ANY CHARACTERS OTHER THEN 0-9, +, -
				583	*	REGISTERS 1, 2, 5, 6, AND 7 ARE USED
				584	*	
00059E	9018 D07C	0007C		585	PDLNUM STM	R1,R8,PDLNSAVE SAVE THE REGISTERS
0005A2	4821 0004	00004		586	LH	R2,4(R1) GET THE STRING ADDRESS
0005A6	5811 0000	00000		587	L	R1,0(R1) GET THE STRING ADDRESS
0005AA	9200 D186	00186		588	MVI	PDLMINUS,0 CLEAR THE NEGATIVE NUMBER FLAG
0005AE	1B55			589	SR	R5,R5 CLEAR THE CHARACTER COUNTER
0005B0	1BFF			590	SR	R15,R15 CLEAR THE ANSWER
0005B2	4165 1000	00000		591	PDLLOOP LA	R6,0(R5,R1) POINT TO THIS DIGIT
0005B6	4155 0001	00001		592	LA	R5,1(R5) GET TO THE NEXT DIGIT
0005BA	1952			593	CR	R5,R2 IS THIS THE END OF THE STRING?
0005BC	4720 C5F8	005F8		594	BH	PDLFINI YES, EXIT
0005C0	1B77			595	SR	R7,R7 CLEAR A WORK REGISTER
0005C2	4376 0000	00000		596	IC	R7,0(R6) GET THE CHARACTER
0005C6	4B70 C60A	0060A		597	SH	R7,PDLH240 SUBTRACT THE CHARACTER C'0'
0005CA	4740 C5D8	005D8		598	BM	PDLSP IF NEGATIVE, CHECK SPECIAL CHARACTERS
0005CE	4CF0 C608	00608		599	MH	R15,PDLH10 IT'S A DIGIT, MULTIPLY PRIOR NUM BY TEN
0005D2	1AF7			600	AR	R15,R7 ADD ON THE NEW DIGIT
0005D4	47F0 C5B2	005B2		601	B	PDLLOOP AND LOOP FOR MORE
				602	*	
				603	*	CHECK FOR SPECIAL CHARACTERS
				604	*	
0005D8	9540 6000	00000		605	PDLSP CLI	0(R6),C' ' IS IT A BLANK?
0005DC	4780 C5B2	005B2		606	BE	PDLLOOP THEN IT'S OK
0005E0	954E 6000	00000		607	CLI	0(R6),C'+' IS IT A PLUS?
0005E4	4780 C5B2	005B2		608	BE	PDLLOOP THAT'S ALSO OK
0005E8	9560 6000	00000		609	CLI	0(R6),C'-' IS IT A MINUS?
0005EC	4770 C5F8	005F8		610	BNE	PDLFINI NO, JUST QUIT
0005F0	9201 D186	00186		611	MVI	PDLMINUS,1 YES, NOTE IT
0005F4	47F0 C5B2	005B2		612	B	PDLLOOP AND LOOK FOR MORE
				613	*	
				614	*	QUIT, AFTER SETTING R15 TO NEGATIVE IF NEEDED
				615	*	
0005F8	9501 D186	00186		616	PDLFINI CLI	PDLMINUS,1 WAS A MINUS SIGN FOUND?
0005FC	4770 C602	00602		617	BNE	PDLLEAVE NO, EXIT
000600	11FF			618	LNR	R15,R15 YES, MAKE IT NEGATIVE
000602	9818 D07C	0007C		619	PDLLEAVE LM	R1,R8,PDLNSAVE RESTORE THE REGISTERS
000606	07F8			620	BR	R8 RETURN
000608	000A			621	PDLH10 DC	H'10'
00060A	00F0			622	PDLH240 DC	H'240'

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				624 *		
				625 *	PRNTLINE - GET EACH LINE FOR SYSOUT, THEN COUNT THE LINES	
				626 *	OUTPUT HEADERS AND TITLE LINES AS NECESSARY	
				627 *		
00060C	9018 D09C	0009C		628	PRNTLINE STM R1,R8,PRTLSAVE SAVE THE REGISTERS	
000610	9110 D0F4	000F4		629	TM SYSOUT+48,X'10' IS SYSOUT OPEN FOR BUSINESS?	
000614	47E0 C6EA	006EA		630	BNO PRNTTERM NO, IT'S NOT OPEN, USE VTOCMSG	
000618	9502 9041	00041		631	CLI HEADK+1,2 NOHEADING REQUEST?	
00061C	4780 C6BE	006BE		632	BE PRNTLIN YES, JUST OUTPUT THE DATA LINE	
				633 *		
				634 *	DO THE LINE COUNTING	
				635 *		
000620	4820 B6D8	006D8		636	LH R2,LINECT GET THE LINE COUNT	
000624	95F1 1004	00004		637	CLI 4(R1),C'1' IS IT REQUESTING A NEW PAGE?	
000628	4780 C658	00658		638	BE PRNTPAGE YES, DO IT	
00062C	9560 1000	00000		639	CLI 0(R1),C'-' SKIP 3 LINES	
000630	4780 C640	00640		640	BE PRNTSKP3 YES, TRY IT	
000634	95F0 1000	00000		641	CLI 0(R1),C'0' SKIP 2 LINES?	
000638	4780 C644	00644		642	BE PRNTSKP2 YES, DO IT	
00063C	47F0 C648	00648		643	B PRNTSKP1 JUST SKIP ONE	
000640	4122 0001	00001		644	PRNTSKP3 LA R2,1(R2) ADD ONE TO THE LINE COUNT	
000644	4122 0001	00001		645	PRNTSKP2 LA R2,1(R2) ADD ONE TO THE LINE COUNT	
000648	4122 0001	00001		646	PRNTSKP1 LA R2,1(R2) ADD ONE TO THE LINE COUNT	
00064C	4020 B6D8	006D8		647	STH R2,LINECT SAVE THE LINE COUNT	
000650	4920 B6DA	006DA		648	CH R2,LINEMAX DOES THIS OVERFLOW HE PAGE?	
000654	4740 C6BE	006BE		649	BL PRNTLIN NO, JUST PUT OUT THIS LINE	
				650 *		
				651 *	PRINT OUT THE PAGE HEADER AND ITEM TITLES	
				652 *		
000658	4860 B6DC	006DC		653	PRNTPAGE LH R6,PAGECT GET THE PAGE COUNT	
00065C	4166 0001	00001		654	LA R6,1(R6) ADD ONE TO IT	
000660	4060 B6DC	006DC		655	STH R6,PAGECT THEN STORE IT BACK	
000664	5870 D070	00070		656	L R7,PAGEADDR GET THE PLACE TO PUT THE PAGE	
		00000		657	USING DUMMD,R7 DUMMY DSECT	
				658	CONV DUMMA,(R6),5 GET THE CHARACTERS	
000668	4E60 B070	00070		659+	CVD R6,DOUBLE CONVERT TO PACKED DECIMAL	
00066C	D20F D192 CB08 00192 00B08			660+	MVC CHARS,EDMASK PUT IN THE EDIT MASK	
000672	DE0F D192 B070 00192 00070			661+	ED CHARS,DOUBLE CONVERT TO CHARACTERS	
000678	D204 7000 D19D 00000 0019D			662+	MVC DUMMA(5),CHARS+16-5 MOVE IN THE NUMBER	
00067E	D50A D192 CB39 00192 00B39			663+	CLC CHARS(16-5),BLANKS WAS THERE AN OVERFLOW?	
000684	4780 C68E	0068E		664+	BE *+10 NO, EVERYTHING WAS OK	
000688	D204 7000 CB49 00000 00B49			665+	MVC DUMMA(5),STARS BAD NEWS, NOTE IT	
				666	DROP R7	
00068E	5010 D060	00060		667	ST R1,PRNTLSAV SAVE THE INPUT REG1	
				668	PUT SYSOUT,PRNTHEAD OUTPUT THE HEADER	
000692	4110 D0C4	000C4		669+	LA 1,SYSOUT LOAD PARAMETER REG 1 01900002	
000696	4100 D3A8	003A8		670+	LA 0,PRNTHEAD LOAD PARAMETER REG 0 02500002	
00069A	58F0 1030	00030		671+	L 15,48(0,1) LOAD PUT ROUTINE ADDR 00550000	
00069E	05EF			672+	BALR 14,15 LINK TO PUT ROUTINE 00600000	
				673	PUT SYSOUT,PRNTTITL OUTPUT THE ITEM TITLES	
0006A0	4110 D0C4	000C4		674+	LA 1,SYSOUT LOAD PARAMETER REG 1 01900002	
0006A4	4100 D4AC	004AC		675+	LA 0,PRNTTITL LOAD PARAMETER REG 0 02500002	
0006A8	58F0 1030	00030		676+	L 15,48(0,1) LOAD PUT ROUTINE ADDR 00550000	
0006AC	05EF			677+	BALR 14,15 LINK TO PUT ROUTINE 00600000	
0006AE	5810 D060	00060		678	L R1,PRNTLSAV GET THE ORIGINAL LINE	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
0006B2	92F0 1004	00004		679	MVI 4(R1),C'0'	ALWAYS SKIP THE FIRST LINE AFTER NEW PAGE
0006B6	4170 0005	00005		680	LA R7,5	SET THE LINE COUNT
0006BA	4070 B6D8	006D8		681	STH R7,LINECT	SAVE IT FOR LATER
				682	PRNTLIN CLEAR OUTWORK	CLEAR A PRINT RECORD
0006BE	9240 D5C0	005C0		683	PRNTLIN MVI OUTWORK,X'40'	SET THE FIRST POSITION
0006C2	D2FE D5C1 D5C0	005C1 005C0		684+	MVC OUTWORK+1(L'OUTWORK-1),OUTWORK	FILL THE ENTIRE FIELD
0006C8	4861 0000	00000		685	LH R6,0(R1)	GET THE RECORD LENGTH
0006CC	4B60 CADA	00ADA		686	SH R6,H5	MINUS 4 FOR PREFIX, 1 FOR EX
0006D0	4131 0004	00004		687	LA R3,4(R1)	POINT TO THE TEXT
0006D4	4460 C598	00598		688	EX R6,OUTMOVE	THEN MOVE IN THE LINE
				689	PUT SYSOUT,OUTWORK	AND FINALLY PRINT IT
0006D8	4110 D0C4	000C4		690+	LA 1,SYSOUT	LOAD PARAMETER REG 1 01900002
0006DC	4100 D5C0	005C0		691+	LA 0,OUTWORK	LOAD PARAMETER REG 0 02500002
0006E0	58F0 1030	00030		692+	L 15,48(0,1)	LOAD PUT ROUTINE ADDR 00550000
0006E4	05EF			693+	BALR 14,15	LINK TO PUT ROUTINE 00600000
0006E6	47F0 C73A	0073A		694	B PRNTLRET	THEN RETURN
				695 *		
				696 *	SIMPLER CHECKING FOR VTOCMSG OUTPUT	
				697 *		
0006EA	4820 B6D8	006D8		698	PRNTTERM LH R2,LINECT	GET THE LINE COUNT
0006EE	4122 0001	00001		699	LA R2,1(R2)	IGNORE CARRIAGE CONTROL
0006F2	4020 B6D8	006D8		700	STH R2,LINECT	SAVE IT BACK
0006F6	1861			701	LR R6,R1	SAVE THE ORIGINAL REG 1
0006F8	9240 1004	00004		702	MVI 4(R1),C' '	BLANK THE CARRIAGE CONTROL
0006FC	9502 9041	00041		703	CLI HEADK+1,2	NOHEADING REQUEST?
000700	4780 C726	00726		704	BE PRNTTTLIN	YES, JUST OUTPUT THE DATA LINE
000704	4920 B6DA	006DA		705	CH R2,LINEMAX	NEED A NEW TITLE?
000708	4740 C726	00726		706	BL PRNTTTLIN	NO, KEEP GOING
				707	VTOCMSG PRNTTITH	YES, PUT IT OUT
00070C	4110 D4A8	004A8		708+	LA R1,PRNTTITH	POINT TO THE FIRST MESSAGE
000710	1B00			709+	SR R0,R0	NO SECOND LEVEL MESSAGE
000712	9001 B078	00078		710+	STM R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES
				711+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE
000716	4110 B000	00000		712+	LA R1,VTOCOM	POINT TO THE COMMON AREA
00071A	58F0 B01C	0001C		713+	L R15,VADMSG	POINT TO THE ROUTINE
00071E	05EF			714+	BALR R14,R15	THEN CALL IT
000720	D201 B6D8 CAD8	006D8 00AD8		715	MVC LINECT,H2	RESET THE LINE COUNT
				716	PRNTTTLIN VTOCMSG 0(R6)	OUTPUT THE PASSED LINE
000726	4116 0000	00000		717+	PRNTTTLIN LA R1,0(R6)	POINT TO THE FIRST MESSAGE
00072A	1B00			718+	SR R0,R0	NO SECOND LEVEL MESSAGE
00072C	9001 B078	00078		719+	STM R0,R1,MSGADDRS	SAVE THE MESSAGE ADDRESSES
				720+*		THEN JUST CALL THE MESSAGE ISSUING ROUTINE
000730	4110 B000	00000		721+	LA R1,VTOCOM	POINT TO THE COMMON AREA
000734	58F0 B01C	0001C		722+	L R15,VADMSG	POINT TO THE ROUTINE
000738	05EF			723+	BALR R14,R15	THEN CALL IT
00073A	9818 D09C	0009C		724	PRNTLRET LM R1,R8,PRTLSAVE	RESTORE THE REGISTERS
00073E	07F8			725	BR R8	THEN RETURN

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
000740	920A D185	00185		727	PRTINIT MVI FIRSTIM,10	FLAG THE INITIALIZATION AS DONE
000744	4110 CCCC	00CCC		728	LA R1,TABTITL	POINT TO THE BUG TABLE
000748	5010 B1AC	001AC		729	ST R1,ATABTITL	LET CHECK KNOW WHERE IT IS
				730	*	
				731	*	SET UP THE CHARACTERS FOR TYPE OF SPACE ALLOC.
				732	*	
00074C	4820 902A	0002A		733	LH R2,SPACEK	GET THE SPACE TYPE KEYWORD
000750	8B20 0003	00003		734	SLA R2,3	MULTIPLY BY 8
000754	4122 CB59	00B59		735	LA R2,TABSPACE(R2)	RELOCATE IT
000758	D205 D188 2000	00188 00000		736	MVC SPACTYPE,0(R2)	SAVE THE CHARACTERS
00075E	9502 903B	0003B		737	CLI PRINTK+1,2	IS THIS NOPRINT?
000762	4780 C7E6	007E6		738	BE NOOPEN	YES, SKIP THE OPEN
000766	D25F D0C4 CB84	000C4 00B84		739	MVC SYSOUT(SYSOUTL),SYSOUTC	INITIALIZE THE DCB
00076C	9280 D074	00074		740	MVI OPENLIST,X'80'	TERMINATE THE LIST
000770	4110 D6C4	006C4		741	LA R1,JFCB	POINT TO THE JFCB
000774	5010 D6C0	006C0		742	ST R1,DCBEXIT	AND PUT THE ADDR IN THE DCB EXIT
000778	9287 D6C0	006C0		743	MVI DCBEXIT,X'87'	NOTE IT AS A JFCB EXIT
00077C	4110 D6C0	006C0		744	LA R1,DCBEXIT	POINT TO THE EXIT LIST
000780	4120 D0C4	000C4		745	LA R2,SYSOUT	AND TO THE DCB FOR ADDRESSABILITY
			00000	746	USING IHADCB,R2	TELL THE ASSEMBLER ABOUT IT
000784	BE17 2025	00025		747	STCM R1,B'0111',DCBEXLSA	STUFF IT INTO THE DCB
000788	9180 90BA	000BA		748	TM CHARSPL+6,X'80'	CHARS PER LINE ENTERED?
00078C	4780 C7B4	007B4		749	BZ RDJFCB	NO, CONTINUE ALONG
000790	4110 90B4	000B4		750	LA R1,CHARSPL	YES, POINT TO THE PDL
000794	4580 C59E	0059E		751	BAL R8,PDLNUM	CONVERT TO A NUMBER
000798	40F0 2052	00052		752	STH R15,DCBLRECL	SAVE THE NEW LRECL
00079C	40F0 B6DE	006DE		753	STH R15,LINELEN	ALSO THE LINE LENGTH
0007A0	9180 90C2	000C2		754	TM BLKSZSET+6,X'80'	BLOCKSIZE ENTERED?
0007A4	4780 C7B0	007B0		755	BZ BLKEQREC	NO, BLOCKSIZE EQUALS LRECL
0007A8	4110 90BC	000BC		756	LA R1,BLKSZSET	POINT TO THE PDL
0007AC	4580 C59E	0059E		757	BAL R8,PDLNUM	GET THE NUMBER
0007B0	40F0 205A	0005A		758	BLKEQREC STH R15,DCBPRECL	STUFF IT AWAY
				759	DROP R2	FINISHED WITH THE DCB
				760	RDJFCB RDJFCB ((R2)),MF=(E,OPENLIST)	SEE IF IT'S THERE
0007B4	4110 D074	00074		761	RDJFCB LA 1,OPENLIST	LOAD PARAMETER REG 1 01900002
0007B8	43E1 0000	00000		762+	IC 14,0(1,0)	SAVE OPTION BYTE 03440000
0007BC	5021 0000	00000		763+	ST R2,0(1,0)	STORE INTO LIST 03620000
0007C0	42E1 0000	00000		764+	STC 14,0(1,0)	RESTORE OPTION BYTE 03520001
0007C4	0A40			765+	SVC 64	ISSUE RDJFCB SVC 00200000
0007C6	12FF			766	LTR R15,R15	WAS IT THERE?
0007C8	4770 C7E6	007E6		767	BNZ NOOPEN	NO, SKIP ALONG
				768	OPEN ((R2),OUTPUT),MF=(E,OPENLIST)	OPEN THE PRINT DCB
0007CC	4110 D074	00074		769+	LA 1,OPENLIST	LOAD PARAMETER REG 1 01900002
0007D0	94F0 1000	00000		770+	NI 0(1),X'F0'	CLEAR OPTION 1 BITS 03780001
0007D4	960F 1000	00000		771+	OI 0(1),15	INSERT OPTION BITS 03720000
0007D8	43E1 0000	00000		772+	IC 14,0(1,0)	SAVE OPTION BYTE 03440000
0007DC	5021 0000	00000		773+	ST R2,0(1,0)	STORE INTO LIST 03620000
0007E0	42E1 0000	00000		774+	STC 14,0(1,0)	RESTORE OPTION BYTE 03520001
0007E4	0A13			775+	SVC 19	ISSUE OPEN SVC 04000000
				776	*	
				777	*	INITIALIZE PRINT VARIABLES
				778	*	
				779	*	SET LINES/PAGE AND LINESIZE
				780	*	
0007E6	D201 B6DA CAD4	006DA 00AD4		781	NOOPEN MVC LINEMAX,DEFLMAX	SET THE DEFAULT NUMBER OF LINES/PAGE

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
0007EC	9180 90CA	000CA		782	TM	LINESPP+6,X'80' LINES PER PAGE ENTERED?	
0007F0	4780 C800	00800		783	BZ	DEFLINPP NO, SKIP ON	
0007F4	4110 90C4	000C4		784	LA	R1,LINESPP YES, POINT TO THE PDL	
0007F8	4580 C59E	0059E		785	BAL	R8,PDLNUM CONVERT TO A NUMBER	
0007FC	40F0 B6DA	006DA		786	STH	R15,LINEMAX AND SAVE IT	
000800	D201 B6D8 B6DA	006D8 006DA		787	DEFLINPP MVC	LINECT,LINEMAX SET UP TO PAGE ON THE FIRST WRITE	
000806	9180 90BA	000BA		788	TM	CHARSPL+6,X'80' CHARS PER LINE ENTERED?	
00080A	4710 C846	00846		789	BO	LENSET YES, USE IT	
00080E	D201 B6DE CAD6	006DE 00AD6		790	MVC	LINELEN,DEFLEN SET UP A DEFAULT LENGTH	
000814	9110 D0F4	000F4		791	TM	SYSOUT+48,X'10' DO WE USE SYSOUT?	
000818	4710 C846	00846		792	BO	LENSET YES, USE WHAT WE'VE GOT	
				793 *		GET THE TERMINAL LINE SIZE TO SEE IF IT MAKES SENSE	
				794		GTSIZE	
00081C	1B11			795+	SR	1,1 PREPARE PARM	68000020
00081E	4100 000B	0000B		796+	LA	0,11 LOAD ENTRY CODE	70000020
000822	8900 0018	00018		797+	SLL	0,24 PUT ENTRY CODE IN LEFT MOST	72000020
000826	0A5E			798+	SVC	94 ISSUE SVC	74000020
000828	1211			799	LTR	R1,R1 SEE IF IT'S GOOD	
00082A	4780 C846	00846		800	BZ	LENSET NO, JUST A ZERO, KEEP THE DEFAULTS	
00082E	0610			801	BCTR	R1,0 CUT IT DOWN ONE TO AVOID A MESS	
000830	4010 B6DE	006DE		802	STH	R1,LINELEN SAVE THIS LENGTH	
000834	1200			803	LTR	R0,R0 FOR DISPLAYS, IT'S SCREEN SIZE	
000836	4780 C846	00846		804	BZ	LENSET KEEP WHAT WE'VE GOT	
00083A	9180 90CA	000CA		805	TM	LINESPP+6,X'80' WAS LINES PER PAGE ENTERED?	
00083E	4710 C846	00846		806	BO	LENSET YES, DON'T OVERRIDE IT	
000842	4000 B6DA	006DA		807	STH	R0,LINEMAX AND SAVE THE NEW PAGE LOCATION	
000846				808	LENSET DS	0H	
				809 *			
				810 *		SET UP THE PAGE COUNTER	
				811 *			
000846	4810 B6DE	006DE		812	PAGEAD LH	R1,LINELEN GET THE LENGTH OF THE LINE	
00084A	4B10 CADE	00ADE		813	SH	R1,H10 MINUS TEN CHARACTERS	
00084E	4111 D3A8	003A8		814	LA	R1,PRNTHD(R1) THEN RELOCATE IT	
				815	CLEAR	PRNTHD CLEAR THE LINE FIRST	
000852	9240 D3A8	003A8		816+	MVI	PRNTHD,X'40' SET THE FIRST POSITION	
000856	D2FE D3A9 D3A8	003A9 003A8		817+	MVC	PRNTHD+1(L'PRNTHD-1),PRNTHD FILL THE ENTIRE FIELD	
00085C	D203 1000 CB04	00000 00B04		818	MVC	0(4,R1),CPAGE MOVE IN THE CHARACTERS PAGE	
000862	4111 0005	00005		819	LA	R1,5(R1) MOVE OVER 5 MORE	
000866	5010 D070	00070		820	ST	R1,PAGEADDR THIS IS THE PLACE	
				821 *			
				822 *		BUILD THE PRINT HEADER LINE	
				823 *			
00086A	9180 90E6	000E6		824	TM	HEADING+6,X'80' IS A USER HEADING PRESENT	
00086E	4710 C8A0	008A0		825	BO	USERHEAD YES, USE IT	
000872	D213 D3A8 CCA4	003A8 00CA4		826	MVC	PRNTHD(L'DEFHEAD),DEFHEAD NO, GET A DEFAULT	
				827 *			
				828 *		ADD THE COMMAND BUFFER TO THE HEADING	
				829 *			
000878	4810 B6DE	006DE		830	LH	R1,LINELEN GET THE LINE LENGTHE	
00087C	4B10 CAE4	00AE4		831	SH	R1,H32 MINUS SPACES FOR PAGE, START OF HDR	
000880	47D0 C8AE	008AE		832	BNP	OUTOPEN IF IT'S SHORT, SKIP ON	
000884	58E0 B044	00044		833	L	R14,ADDRCBUF POINT TO THE COMMAND BUFFER	
000888	491E 0000	00000		834	CH	R1,0(R14) COMPARE LENGTHS	
00088C	4740 C894	00894		835	BL	CBUFBIG THE COMMAND BUFFER TOO BIG	
000890	481E 0000	00000		836	LH	R1,0(R14) GET THE COMMAND BUFFER SIZE	

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

000894 4B10 CADA      00ADA      837 CBUFBIG  SH   R1,H5      MINUS 1 FOR EX, 4 FOR CBUF PREFIX
000898 4410 CAF8      00AF8      838          EX   R1,MOVEHED MOVE THE CBUF TO THE DEFAULT HEADER
00089C 47F0 C8AE      008AE      839          B    OUTOPEN  GO SET UP THE PAGE COUNTER
0008A0 4810 90E4      000E4      840 USERHEAD LH   R1,HEADING+4 GET THE LENGTH
0008A4 0610          841          BCTR R1,0      MINUS ONE FOR THE EX
0008A6 5820 90E0      000E0      842          L    R2,HEADING POINT TO THE USER HEAD
0008AA 4410 CAF2      00AF2      843          EX   R1,MOVEHEAD THEN MOVE IT IN
0008AE          844 OUTOPEN  DS    0H
0008AE          845 *
0008AE          846 *      SET UP THE WORK LINE
0008AE          847 *
0008AE 4810 B6DE      006DE      848          LH   R1,LINELEN  GET THE LINE LENGTH
0008B2 4111 0004      00004      849          LA   R1,4(R1)    ADD FOUR FOR THE PREFIX
0008B6 4010 D2A2      002A2      850          STH  R1,WORKLINE OUTPUT TEXT
0008BA 4010 D4A8      004A8      851          STH  R1,PRNTTITH ITEM TITLES
0008BA          852 *      GET THE PRINT SPECIFICATION
0008BA          853 *
0008BE 4110 CCB8      00CB8      854          LA   R1,DEFPRNT
0008C2 5010 D774      00774      855          ST   R1,VTPRNTLS SAVE THE PRINT ITEM LIST ADDRESS
0008C6 9180 90D2      000D2      856          TM   SUBPRTKY+6,X'80' WERE ANY ITEMS SET UP
0008CA 4780 C992      00992      857          BE   PRTITSET    NO, THE DEFAULT LIST IS OK
0008CA          858 *
0008CA          859 *      GET THE ADD, REPLACE, DELETE, AND NEW ITEMS
0008CA          860 *      AND BUILD THE NEW LIST
0008CA          861 *
0008CA          862 *      FIRST CONVERT THE ENTERED TEXT INTO NUMERIC KEYS
0008CA          863 *
0008CE 4140 90D4      000D4      864          LA   R4,SUBPRTIT POINT TO THE ITEMS
0008D2 4150 D7A0      007A0      865          LA   R5,VTPRNTE POINT TO THE OUTPUT KEYS
0008D6 4100 D7C8      007C8      866          LA   R0,VTPRNTEX POINT TO THE END OF THE LIST
0008DA 4580 CA6E      00A6E      867 ENTKEY  BAL  R8,GETKEY  GET A KEY
0008DE 42F5 0000      00000      868          STC  R15,0(R5)  SAVE IT
0008E2 4155 0001      00001      869          LA   R5,1(R5)   GET TO THE NEXT ONE
0008E6 1905          870          CR   R0,R5      CHECK FOR THE END
0008E8 47D0 C8F4      008F4      871          BNH  ENTKEND    IF THAT'S ALL
0008EC BF47 4009      00009      872          ICM  R4,7,9(R4) GET THE CHAIN POINTER
0008F0 4770 C8DA      008DA      873          BNZ  ENTKEY     AND KEEP GOING IF THERE'S MORE
0008F4          874 ENTKEND  DS    0H      THE KEYS ARE ENTERED INTO THE LIST
0008F4          875 *
0008F4          876 *      MERGE THE ENTERED ITEMS AND THE DEFAULT LIST INTO A NEW LIST
0008F4          877 *
0008F4 4160 D778      00778      878          LA   R6,VTPRNTE POINT TO THE NEW LIST
0008F8 5060 D774      00774      879          ST   R6,VTPRNTLS SAVE IT'S ADDRESS
0008FC 4140 D7A0      007A0      880          LA   R4,VTPRNTE POINT TO THE ENTERED ITEMS
000900 4120 CCB8      00CB8      881          LA   R2,DEFPRNT POINT TO THE DEFAULT LIST
000904 1832          882          LR   R3,R2      POINT TO THE BEGINNING - NO DEFAULT IF NEW
000906 5810 90CC      000CC      883          L    R1,SUBPRTKY POINT TO THE KEYWORD
00090A 95D5 1000      00000      884          CLI  0(R1),C'N' IS THIS A NEW LIST?
00090E 4780 C98A      0098A      885          BE   PRTINEW    YES, SKIP PAST DEFAULT COPY
000912 4130 CCCC      00CCC      886          LA   R3,DEFPRNTE POINT TO THE END OF THE DEFAULT LIST
000912          887 *
000912          888 *      ADD, REPLACE, DELETE - COPY THE DEFAULT LIST
000912          889 *
000916 1923          890 PRTICOPY CR   R2,R3      IS THIS THE END OF THE DEFAULT LIST?
000918 47B0 C992      00992      891          BNL  PRTITSET    YES, END OF PROCESSING FOR PRINT ITEMS

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				892	*	
				893	*	CHECK FOR ENTERED ITEMS THAT ARE ALSO
				894	*	IN THE DEFAULT LIST.
				895	*	
00091C	D500	2000	4000	00000	00000	896 PRTICDLP CLC 0(1,R2),0(R4) IS THIS THE SAME ITEM
000922	4780	C948		00948		897 BE PRTIFND YES, SEE WHAT TO DO
000926	4144	0001		00001		898 LA R4,1(R4) NO, GET TO THE NEXT ITEM
00092A	9500	4000		00000		899 CLI 0(R4),0 WAS THIS THE LAST ENTERED ITEM?
00092E	4770	C91C		0091C		900 BNE PRTICDLP NO, KEEP LOOKING
000932	4140	D7A0		007A0		901 LA R4,VTPRNTEN POINT BACK TO THE TOP OF THE LIST
				902	*	
				903	*	ADD THIS ITEM TO THE NEW LIST
				904	*	
000936	D200	6000	2000	00000	00000	905 PRTICSKP MVC 0(1,R6),0(R2) MOVE IN THE NEW KEY
00093C	4166	0001		00001		906 LA R6,1(R6) POINT PAST IT
000940	4122	0001		00001		907 PRTICDEL LA R2,1(R2) GO DOWN THE DEFAULT LIST
000944	47F0	C916		00916		908 B PRTICOPY THEN KEEP ON CHECKING
				909	*	
				910	*	AN ITEM WAS ENTERED AND WAS IN THE DEFAULT LIST
				911	*	FOR DELETE, JUST DELETE ITEMS
				912	*	FOR ADD AND REPLACE, DELETE ALL BUT THE FIRST ITEM
				913	*	TO AVOID DUPLICATES
				914	*	IF IT IS THE FIRST ITEM, INSERT THE ENTERED LIST
				915	*	
000948	4100	D7A0		007A0		916 PRTIFND LA R0,VTPRNTEN POINT TO THE FIRST ITEM
00094C	1904					917 CR R0,R4 COMPARE WITH THE ITEM FOUND
00094E	4720	C940		00940		918 BH PRTICDEL NOT THE FIRST ITEM, DELETE IT
000952	95C4	1000		00000		919 CLI 0(R1),C'D' IS THIS DELETE TIME?
000956	4780	C940		00940		920 BE PRTICDEL THEN JUST DELETE IT
00095A	95D9	1000		00000		921 CLI 0(R1),C'R' IS THIS A REPLACE?
00095E	4780	C96C		0096C		922 BE PRTIREPA YES, IGNORE THIS DEFAULT ITEM
				923	*	
				924	*	ADD THE DEFAULT ITEM FIRST
				925	*	
000962	D200	6000	2000	00000	00000	926 PRTIASKP MVC 0(1,R6),0(R2) MOVE IN THE NEW KEY
000968	4166	0001		00001		927 LA R6,1(R6) POINT PAST IT
				928	*	
				929	*	MOVE THE ITEMS IN FROM THE ADD OR REPLACE LIST
				930	*	
00096C	4122	0001		00001		931 PRTIREPA LA R2,1(R2) GET PAST THE DEFAULT LIST ITEM
000970	4144	0001		00001		932 PRTIREP LA R4,1(R4) GET PAST THE FIRST ENTRY
000974	9500	4000		00000		933 PRTINEXT CLI 0(R4),0 IS THIS THE LAST ITEM?
000978	4780	C916		00916		934 BE PRTICOPY YES, SEE ABOUT MORE DEFAULTS
				935	*	
				936	*	ADD AN ENTERED ITEM TO THE LIST
				937	*	
00097C	D200	6000	4000	00000	00000	938 PRTINSKP MVC 0(1,R6),0(R4) MOVE IN THE NEW KEY
000982	4166	0001		00001		939 LA R6,1(R6) POINT PAST IT
000986	47F0	C970		00970		940 B PRTIREP GO GET MORE ENTERED ITEMS
				941	*	
				942	*	NEW LIST, JUST USE IT AS ENTERED
				943	*	
00098A	4110	D7A0		007A0		944 PRTINEW LA R1,VTPRNTEN POINT TO THE ENTERED LIST
00098E	5010	D774		00774		945 ST R1,VTPRNTLS THEN SAVE ITS ADDRESS FOR LATER
000992						946 PRTITSET DS 0H

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
					947	*		
					948	*	BUILD THE TITLE LINE FOR THE DSNNAME FIELDS	
					949	*		
					950		CLEAR PRNTTTITL BLANK OUT THE TITLE LINE	
000992	9240	D4AC	004AC		951+		MVI PRNTTTITL,X'40' SET THE FIRST POSITION	
000996	D2FE	D4AD	D4AC	004AD	004AC	952+	MVC PRNTTTITL+1(L'PRNTTTITL-1),PRNTTTITL FILL THE ENTIRE FIELD	
00099C	D28A	D4AD	D4AC	004AD	004AC	953	MVC PRNTTTITL+1(139),PRNTTTITL SO THE WHOLE THING IS GOOD	
0009A2	5820	D774		00774		954	L R2,VTPRNTLS GET THE PRINT ITEM LIST	
0009A6	4110	D4AC		004AC		955	LA R1,PRNTTTITL POINT TO THE TITLE LINE	
0009AA	1B66					956	TITLOOP SR R6,R6 GET THE RESERVED WORD NUMBER	
0009AC	4362	0000		00000		957	IC R6,0(R2) FROM THE TOP BYTE	
0009B0	4C60	CAE0		00AE0		958	MH R6,H12 MULTIPLY BY 12 FOR THE TABLE ENTRIES	
0009B4	4166	CCCC		00CCC		959	LA R6,TABTTITL(R6) THEN RELOCATE THE MESS	
0009B8	1B77					960	SR R7,R7 CLEAR A REGISTER	
0009BA	4376	0000		00000		961	IC R7,0(R6) GET THE EXECUTE LENGTH	
0009BE	4470	CAE6		00AE6		962	EX R7,MOVETIT MOVE IN THE TITLE	
0009C2	4111	7001		00001		963	LA R1,1(R1,R7) MOVE THE POINTER OVER	
0009C6	951A	2000		00000		964	CLI 0(R2),DSNAME IS THIS THE DSNNAME KEY	
0009CA	4780	C9EE		009EE		965	BE TITDSN YES, SPECIAL PROCESSING	
0009CE	4100	D4AC		004AC		966	LA R0,PRNTTTITL POINT TO THE BEGINNING AGAIN	
0009D2	1B10					967	SR R1,R0 AND FIND THE CURRENT LENGTH	
0009D4	4910	B6DE		006DE		968	CH R1,LINELEN IS IT TOO LONG?	
0009D8	47B0	CA22		00A22		969	BNL TITOVER YES, PULL BACK	
0009DC	1A10					970	AR R1,R0 NO, KEEP GOING	
0009DE	4122	0001		00001		971	TITINC LA R2,1(R2) GET THE NEXT CHAIN POINTER	
0009E2	9500	2000		00000		972	CLI 0(R2),0 ARE WE DONE?	
0009E6	4770	C9AA		009AA		973	BNE TITLOOP GO GET MORE TITLES	
0009EA	47F0	CA2E		00A2E		974	B TITEND ALL DONE	
0009EE	1861					975	TITDSN LR R6,R1 SAVE THE ADDRESS POINTER	
0009F0	4110	915C		0015C		976	LA R1,DSNPLN POINT TO THE PDL FOR DSN LENGTH	
0009F4	4580	C59E		0059E		977	BAL R8,PDLNUM GO TRANSLATE IT	
0009F8	12FF					978	LTR R15,R15 WAS IT THERE?	
0009FA	4720	CA02		00A02		979	BP TITDSN2 YES, USE IT	
0009FE	41F0	002C		0002C		980	LA R15,44 NO, SET THE DEFAULT	
000A02	40F0	D18E		0018E		981	TITDSN2 STH R15,DSNLENGT SAVE THE LENGTH	
000A06	4BF0	CADC		00ADC		982	SH R15,H9 SUBTRACT THE 9 CHARS MOVED ALREADY	
					983	*	MINUS ONE FOR EX, PLUS ONE FOR SPACE	
000A0A	9240	6000		00000		984	MVI 0(R6),C' ' GET AN INITIAL BLANK	
000A0E	44F0	CAFE		00AFE		985	EX R15,DSNBLMOV MOVE IN THE BLANKS	
000A12	411F	6001		00001		986	LA R1,1(R15,R6) RESET THE POINTER (INCLUDE A SPACE)	
000A16	41FF	0001		00001		987	LA R15,1(R15) ADD ON THE SPACE CHARACTER	
000A1A	40F0	D190		00190		988	STH R15,DSNLENOF SAVE THE OFFSET	
000A1E	47F0	C9DE		009DE		989	B TITINC ALLOW DSNNAME TO OVERFLOW THE LINE	
					990	*		
					991	*	TITLE RAN OFF THE END, CUT IT OFF	
					992	*		
000A22	1B17					993	TITOVER SR R1,R7 SUBTRACT PAST THIS FIELD	
000A24	1A10					994	AR R1,R0 RELOCATE IT	
000A26	0610					995	BCTR R1,0 THEN GET THE LAST CHARACTER	
000A28	D208	1000	CB39	00000	00B39	996	MVC 0(9,R1),BLANKS THEN BLANK IT OUT	
000A2E	9110	D0F4		000F4		997	TITEND TM SYSOUT+48,X'10' IS THE DCB OPEN	
000A32	47E0	CA3A		00A3A		998	BNO CKOUTPT NO, TERMINAL OUTPUT, NO CC	
000A36	92F0	D4AC		004AC		999	MVI PRNTTTITL,C'0' ALWAYS SKIP A LINE FOR IT	
000A3A	9501	9045		00045		1000	CKOUTPT CLI OUTPUTK+1,1 OUTPUT THIS RUN?	
000A3E	4770	C076		00076		1001	BNE VTRET YES, SKIP THE OPEN, JUST RETURN	

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM 0201 13.38 01/07/25
000A42	D25F	D124	CBE4	00124	00BE4	1002	MVC OUTDCB(OUTDCBL),OUTDCBC	INITIALIZE THE DCB
000A48	9280	D074		00074		1003	MVI OPENLIST,X'80'	TERMINATE THE LIST
						1004	OPEN (OUTDCB,OUTPUT),MF=(E,OPENLIST)	OPEN THE DATA SET DCB
000A4C	4110	D074		00074		1005+	LA 1,OPENLIST	LOAD PARAMETER REG 1 01900002
000A50	94F0	1000		00000		1006+	NI 0(1),X'F0'	CLEAR OPTION 1 BITS 03780001
000A54	960F	1000		00000		1007+	OI 0(1),15	INSERT OPTION BITS 03720000
000A58	43E1	0000		00000		1008+	IC 14,0(1,0)	SAVE OPTION BYTE 03440000
000A5C	4100	D124		00124		1009+	LA 0,OUTDCB	PICK UP DCB ADDRESS 03480000
000A60	5001	0000		00000		1010+	ST 0,0(1,0)	STORE INTO LIST 03500000
000A64	42E1	0000		00000		1011+	STC 14,0(1,0)	RESTORE OPTION BYTE 03520001
000A68	0A13					1012+	SVC 19	ISSUE OPEN SVC 04000000
000A6A	47F0	C076		00076		1013	B VTRET	RETURN, INITIALIZATION IS DONE

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25		
				1015	*			
				1016	*	ROUTINE TO CONVERT A TEXT DSCB ITEM		
				1017	*	INTO ITS KEY NUMBER		
				1018	*	INPUT IS REG 4 - IKJIDENT PTR		
				1019	*	OUTPUT IS REG 15 - KEY NUMBER		
				1020	*	ENTRY VIA BAL R8,GETKEY		
				1021	*			
000A6E	5810	B1AC	001AC	1022	GETKEY L	R1,ATABTITL POINT TO THE TABLE		
000A72	4111	000C	0000C	1023	LA	R1,12(R1) POINT TO THE FIRST ENTRY		
000A76	41F0	0001	00001	1024	LA	R15,1 SET UP THE KEY NUMBER COUNTER		
000A7A	5864	0000	00000	1025	L	R6,0(R4) POINT TO THE ENTERED TEXT		
000A7E	BF33	4004	00004	1026	ICM	R3,3,4(R4) GET THE LENGTH OF THE ENTERED TEXT		
000A82	47D0	CABC	00ABC	1027	BNP	GETKNOTF NOT FOUND IF ZERO		
000A86	0630			1028	BCTR	R3,0 MINUS ONE FOR THE EX		
000A88	4121	0004	00004	1029	GETKLOOP LA	R2,4(R1) POINT TO THE COMPARISON TEXT		
000A8C	9540	2000	00000	1030	CLI	0(R2),C' ' IS IT HERE?		
000A90	4770	CAA4	00AA4	1031	BNE	GETKSTD YES, THIS IS IT		
000A94	4122	0001	00001	1032	LA	R2,1(R2) NO, MOVE OVER ONE MORE		
000A98	9540	2000	00000	1033	CLI	0(R2),C' ' IS IT HERE?		
000A9C	4770	CAA4	00AA4	1034	BNE	GETKSTD YES, THIS IS IT		
000AA0	4122	0001	00001	1035	LA	R2,1(R2) NO, MOVE OVER ONE MORE		
000AA4	4430	CAC0	00AC0	1036	GETKSTD EX	R3,GETKCOMP COMPARE THE KEY TEXT		
000AA8	4780	CABE	00ABE	1037	BE	GETKFND I FOUND IT		
000AAC	4111	000C	0000C	1038	LA	R1,12(R1) GET TO THE NEXT KEY		
000AB0	41FF	0001	00001	1039	LA	R15,1(R15) INCREMENT THE KEY COUNTER		
000AB4	49F0	CAC6	00AC6	1040	CH	R15,H26 CHECK FOR THE END OF THE TABLE		
000AB8	47D0	CA88	00A88	1041	BNH	GETKLOOP NOT YET, KEEP LOOKING		
				1042	*			
				1043	*	KEY WAS NOT FOUND, SEND BACK A ZERO		
				1044	*			
000ABC	1BFF			1045	GETKNOTF SR	R15,R15 SET UP THE ZERO AND RETURN		
000ABE	07F8			1046	GETKFND BR	R8 JUST RETURN		
000AC0	D500	6000	2000	00000	00000	1047	GETKCOMP CLC	0(0,R6),0(R2) EXECUTED TEXT COMPARE
000AC6	001A			1048	H26	DC	H'26'	


```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
ASM 0201 13.38 01/07/25

000CBA 08          1209      DC    AL1(PCT)
000CBB 09          1210      DC    AL1(EXT)
000CBC 0A          1211      DC    AL1(DSORG)
000CBD 0B          1212      DC    AL1(RECFM)
000CBE 0D          1213      DC    AL1(LRECL)
000CBF 0C          1214      DC    AL1(BLKSZ)
000CC0 03          1215      DC    AL1(CDATE)
000CC1 04          1216      DC    AL1(LSTUS)
000CC2 02          1217      DC    AL1(VOLUME)
000CC3 1A          1218      DC    AL1(DSNAME)
000CC4 05          1219      DC    AL1(EXPDT)
000CC5 11          1220      DC    AL1(SECQ)
000CC6 10          1221      DC    AL1(SECT)
000CC7 13          1222      DC    AL1(ROUND)
000CC8 0E          1223      DC    AL1(PASS)
000CC9 01          1224      DC    AL1(ACTION)
000CCA 14          1225      DC    AL1(TYPE)
000CCB 00          1226      DC    AL1(0)          END OF THE LIST
                00CCC 1227 DEFPRNTE EQU    *
                00014 1228 DEFPRNTL EQU   *-DEFPRNT
                1229 *
                1230 *          TABLE OF PRINT ITEM LENGTHS AND TITLES
                1231 *
                1232 *          ENTRIES IN THE TABLE FOR EACH FORMATTED ITEM -
                1233 *          FIRST BYTE IS FIELD LENGTH FOR OUTPUT ( MINUS ONE FOR EX )
                1234 *          SECOND BYTE - X'80' BIT INDICATES A CHARACTER FIELD
                1235 *                   THEN BITS 0-7 GIVE VTFMT LENGTH
                1236 *                   OTHERWISE IT'S A KEY TO WHICH ROUTINE TO USE
                1237 *          THIRD BYTE - OFFSET IN FORMATTED DSCB, VTFMT
                1238 *          4-12 TH BYTES, THE TITLE FOR THE FIELD
                1239 *
                1240 *
000CCC 0000000000000000 1241 TABTITL DC    XL12'00'  DUMMY ENTRY FOR 0 ADDRESSING
000CD8 08873B40C1C3E3C9 1242      DC    AL1(8),AL1(128+7),AL1(VTFACTON-VTFMT),CL9' ACTION ' 1
000CE4 06851240E5D6D3E4 1243      DC    AL1(6),AL1(128+5),AL1(VTFVOLUM-VTFMT),CL9' VOLUME ' 2
000CF0 05041C40C3C4C1E3 1244      DC    AL1(5),AL1(000+4),AL1(VTFCREDT-VTFMT),CL9' CDATE ' 3
000CFC 05042240D9C5C6C4 1245      DC    AL1(5),AL1(000+4),AL1(VTFLSTAC-VTFMT),CL9' REFDT ' 4
000D08 05041F40C5E7D7C4 1246      DC    AL1(5),AL1(000+4),AL1(VTFEXPDT-VTFMT),CL9' EXPDT ' 5
000D14 060C044040C1D3D3 1247      DC    AL1(6),AL1(00+12),AL1(VTFALLOC-VTFMT),CL9' ALLOC ' 6
000D20 06100440E4D5E4E2 1248      DC    AL1(6),AL1(00+16),AL1(VTFALLOC-VTFMT),CL9' UNUSED ' 7
000D2C 03140440D7C3E340 1249      DC    AL1(3),AL1(00+20),AL1(VTFALLOC-VTFMT),CL9' PCT ' 8
000D38 02182540C5E74040 1250      DC    AL1(2),AL1(00+24),AL1(VTFNOEPV-VTFMT),CL9' EX ' 9
000D44 03822640C4E2D640 1251      DC    AL1(3),AL1(128+2),AL1(VTFDSORG-VTFMT),CL9' DSO ' 10
000D50 03832940D9C6D440 1252      DC    AL1(3),AL1(128+3),AL1(VTFRECFM-VTFMT),CL9' RFM ' 11
000D5C 051C3040C2D3D2E2 1253      DC    AL1(5),AL1(00+28),AL1(VTFBLKSZ-VTFMT),CL9' BLKSZ ' 12
000D68 051C2E40D3D9C5C3 1254      DC    AL1(5),AL1(00+28),AL1(VTFLRECL-VTFMT),CL9' LRECL ' 13
000D74 04803340D7C1E2E2 1255      DC    AL1(4),AL1(128+0),AL01(VTFPROT-VTFMT),CL9' PASS ' 14
000D80 03803640C3C1E340 1256      DC    AL1(3),AL1(128+0),AL1(VTFCATLG-VTFMT),CL9' CAT ' 15
000D8C 04803940E2C5C3E3 1257      DC    AL1(4),AL1(128+0),AL1(VTFSECAL-VTFMT),CL9' SECT ' 16
000D98 051C374040E2C5C3 1258      DC    AL1(5),AL1(00+28),AL1(VTFSECAM-VTFMT),CL9' SECQ ' 17
000DA4 04831840E4D5C9E3 1259      DC    AL1(4),AL1(128+3),AL1(VTFUNIT-VTFMT),CL9' UNIT ' 18
000DB0 05803240D9D6E4D5 1260      DC    AL1(5),AL1(128+0),AL1(VTFROUND-VTFMT),CL9' ROUND ' 19
000DBC 04803A40E3E8D7C5 1261      DC    AL1(4),AL1(128+0),AL1(VTFDSTYP-VTFMT),CL9' TYPE ' 20
000DC8 060C08404040E4E2 1262      DC    AL1(6),AL1(00+12),AL01(VTFUSED-VTFMT),CL9' USED ' 21
000DD4 082012404040C3C3 1263      DC    AL1(8),AL1(00+32),AL1(VTFVOLUM-VTFMT),CL9' CCHH ' 22

```

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				1264	* 2 LINES BELOW	TANSKY
000DE0	03803540E4D7C440			1265	DC AL1(3),AL1(128+0),AL1(VTFUPD-VTFMT),CL9' UPD	' 23
000DEC	04803440D9C1C3C6			1266	DC AL1(4),AL1(128+0),AL1(VTFRACF-VTFMT),CL9' RACF	' 24
000DF8	06801240C4E4D4D4			1267	DC AL1(6),AL1(128+0),AL1(VTFVOLUM-VTFMT),CL9' DUMMY5	' 25
000E04	08204640C4E2D5C1			1268	DC AL1(8),AL1(00+32),AL001(VTFDSN-VTFMT),CL9' DSNAME	' 26

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM 0201 13.38 01/07/25
				1270 *		
				1271 *		
				1272 *	P A R S E C O N T R O L L I S T	
				1273 *		
				1274 *		
				1275	PRINT OFF	
				1277	PUSH PRINT	
				1278	PRINT NOGEN	
				2446	POP PRINT	
				2473	PRINT ON	
				2474 *		
				2475 *	DYNAMIC WORK AREA	
				2476 *		
000000				2478	PRNTWORK DSECT	
000000				2479	DS 18A PRINT ROUTINE SAVE AREA	
000048				2480	TOTDS DS F TOTAL COUNTER	
00004C				2481	TOTALLOC DS F TOTAL ALLOCATION	
000050				2482	TOTUSED DS F TOTAL USED	
000054				2483	FTOTDS DS F FINAL TOTAL DATA SETS	
000058				2484	FTOTALLC DS F FINAL TOTAL ALLOC	
00005C				2485	FTOTUSED DS F FINAL TOTAL USED	
000060				2486	PRNTLSAV DS A	
000064				2487	PRINTR8 DS A	
000068				2488	PRNTTOT8 DS A	
00006C				2489	ADDREND DS A	
000070				2490	PAGEADDR DS A	
000074				2491	OPENLIST DS 2A PARM LIST FOR OPEN	
00007C				2492	PDLNSAVE DS 8A REGISTER SAVE AREA FOR PDLNUM RTN	
00009C				2493	PRTLSAVE DS 8A REGISTER SAVE AREA FOR PRNTLINE RTN	
0000BC				2494	LASTKEY DS A ADDRESS OF LAST KEY FOR SUBTOTALS, BREAKS	
0000C0				2495	NUMBREAK DS H CHARACTERS TO COMPARE FOR BREAK	
0000C2				2496	NUMTOTAL DS H CHARACTERS TO COMPARE FOR SUBTOTALS	
				2497	PRINT NOGEN	
				2498	SYSOUT DCB DSORG=PS,DDNAME=VTOCOUT,MACRF=PM, X	
					RECFM=FBA,LRECL=150,BLKSIZE=1500	
	00060			2552	SYSOUTL EQU *-SYSOUT	
				2553	OUTDCB DCB DSORG=PS,DDNAME=OUTPUT,MACRF=PM, X	
					RECFM=FB,LRECL=100,BLKSIZE=6000	
	00060			2607	OUTDCBL EQU *-OUTDCB	
000184				2608	ENDTOTAL DS X PROGRAM SWITCHES	
		00080		2609	ENTOTOUT EQU X'80' THE FINAL TOTALS HAVE BEEN OUTPUT	
		00010		2610	ENDTONLY EQU X'10' NO SUBTOTALS, END TOTALS ONLY	
		00008		2611	NOBREAK EQU X'08' NO BREAKS	
000185				2612	FIRSTIM DS X INITIALIZATION FOR THIS ROUTINE	
000186	00			2613	PDLMINUS DC X'00'	
000187				2614	TOTLAST DS X	
000188				2615	SPACTYPE DS CL6 CHARACTERS FOR SPACE UNITS	
00018E				2616	DSNLENGT DS H	
000190				2617	DSNLENOF DS H	
000192				2618	CHARS DS CL16 CONVERSION TO CHARACTERS	
0001A2				2619	MSGWORK DS CL256 AREA FOR BUILDING MESSAGES	
0002A2				2620	WORKLINE DS CL256 AREA FOR DATA SET	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
0003A4				2621	PRNTHDRH DS	F
0003A8				2622	PRNTHDRH DS	CL256 AREA FOR HEADER
0004A8				2623	PRNTTITH DS	F
0004AC				2624	PRNTTITL DS	CL256 AREA FOR ITEM TITLES
0005AC				2625	MSGBL DS	CL20 AREA FOR BLANK LINE
0005C0				2626	OUTWORK DS	CL256 WORKING AREA FOR OUTPUT
0006C0				2627	DCBEXIT DS	F
0006C4				2628	JFCB DS	XL176
000774				2630	VTPRNTLS DS	A PRINT ITEM LIST ADDRESS
000778				2631	VTPRNTL DS	40C PRINT ITEM LIST (IF MODIFIED)
0007A0				2632	VTPRNTEN DS	40C
0007C8				2633	VTPRNTEX DS	C
0007D0				2635	DS	0D
		007D0		2636	LENWORK EQU	*-PRNTWORK
				2637	*	
				2638	*	VTOC COMMAND COMMON AREA
				2639	*	
				2640		PRINT NOGEN
				2641		VTOCOM
				2809	*	
				2810	*	FORMATTED DSCB
				2811	*	
				2812		VTFMT
				2854		PDEDSNAM
000000				2876	DUMMD DSECT	
000000				2877	DUMMA DS	C DUMMY ENTRY TO USE FOR CONV
				2878		PRINT NOGEN
				2879	DCBD	DSORG=PS,DEV D=DA
				3437		END
000E10	C5D9D9			3438		=C'ERR'

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
ACTION	00001	00000001	02447	01224
ADDRANSR	00004	00000048	02674	00057
ADDRCBUF	00004	00000044	02672	00833
ADDREND	00004	0000006C	02489	00165 00231
ADDTOT	00004	00000208	00297	00089 00221
ALLOC	00001	00000006	02452	01207
AL00741	00001	00000F49	01508	01507
AL00751	00001	00000F57	01522	01521
ANDOR1K	00002	00000034	01496	01493
ANDOR2K	00002	00000036	01535	01532
ANDOR3K	00002	00000038	01566	01563
ATABTITL	00004	000001AC	02707	00729 01022
BLANKS	00016	00000B39	01075	00322 00331 00340 00436 00467 00471 00486 00513 00517 00529 00542 00663 00996
BLKEQREC	00004	000007B0	00758	00755
BLKSZ	00001	0000000C	02458	01214
BLKSZSET	00004	000000BC	01936	00754 00756 01925
BREAK	00004	000000AC	01889	00153 01876
BREAKK	00002	00000030	01454	01451
CATK	00002	0000002C	01404	01401
CBUFBIG	00004	00000894	00837	00835
CDATE	00001	00000003	02449	01215
CHARS	00016	00000192	02618	00319 00320 00321 00322 00328 00329 00330 00331 00337 00338 00339 00340 00433 00434 00435 00436 00443 00444 00445 00446 00453 00454 00455 00464 00465 00466 00467 00483 00484 00485 00486 00510 00511 00512 00513 00526 00527 00528 00529 00539 00540 00541 00542 00660 00661 00662 00663
CHARSK	00002	0000003C	01627	01624
CHARSPL	00004	000000B4	01916	00748 00750 00788 01905
CHEKTOT	00004	0000008E	00132	00084
CHKBREAK	00004	00000110	00190	00178 00182
CHKOUTPT	00004	00000154	00215	00174
CHKSORT	00004	00000042	00076	00071
CKOUTPT	00004	00000A3A	01000	00998
COMPKEY	00006	00000AEC	01066	00181 00194
COMP0	00016	00000B29	01074	00446
CONTAIN	00004	00000084	01832	01826
CONTAINK	00002	00000028	01360	01357
CPAGE	00004	00000B04	01070	00818
DCBBIT0	00001	00000080	02901	02987 02995 03007 03030 03057 03059 03060 03062 03085 03088 03108 03112 03127 03164 03219 03250 03289 03293 03306 03406 03408 03418
DCBBIT1	00001	00000040	02902	02988 02996 03009 03031 03032 03041 03057 03059 03061 03062 03090 03108 03110 03112 03130 03131 03132 03167 03168 03219 03252 03295 03297 03309 03353 03406 03410 03419
DCBBIT2	00001	00000020	02903	02989 02997 03010 03011 03012 03031 03032 03036 03042 03057 03058 03063 03092 03113 03114 03135 03136 03137 03171 03172 03220 03257 03298 03314 03356 03359 03406 03420
DCBBIT3	00001	00000010	02904	02990 03010 03012 03013 03031 03044 03064 03095 03113 03116 03139 03140 03141 03175 03176 03220 03259 03262 03264 03300 03315 03356 03360 03406
DCBBIT4	00001	00000008	02905	02998 03045 03065 03096 03118 03123 03124 03144 03145 03179 03180 03182 03183 03221 03267 03316 03356 03361
DCBBIT5	00001	00000004	02906	02999 03046 03068 03069 03098 03118 03120 03121 03124 03148 03150 03151 03152 03186 03187 03188 03189 03221 03269 03272 03302 03318 03351
DCBBIT6	00001	00000002	02907	02991 03047 03048 03051 03068 03070 03099 03155 03156 03157 03158 03192 03193 03194 03195 03222 03275 03320 03362
DCBBIT7	00001	00000001	02908	02992 03047 03049 03051 03072 03103 03160 03161 03198 03199 03201 03202 03278 03304 03321 03364
DCBEXIT	00004	000006C0	02627	00742 00743 00744
DCBEXLSA	00003	00000025	03074	00747

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
DCBFDAD	00008	00000005	02928	02931	
DCBLRECL	00002	00000052	03388	00752	
DCBPRECL	00002	0000005A	03423	00758	
DEFHEAD	00020	00000CA4	01203	00826	00826
DEFLEN	00002	00000AD6	01057	00790	
DEFLINPP	00006	00000800	00787	00783	
DEFLMAX	00002	00000AD4	01056	00781	
DEFPRNT	00001	00000CB8	01207	00854	00881 01228
DEFPRNTE	00001	00000CCC	01227	00886	
DEND0107	00001	000010F9	01858	01848	
DEND0109	00001	0000112C	01887	01875	
DEND0111	00001	0000115D	01914	01904	
DEND0112	00001	0000117E	01934	01924	
DEND0114	00001	000011AF	01961	01951	
DEND0116	00001	000011D5	01988	01978	
DEND0117	00001	000011F1	02008	01998	
DEND0121	00001	0000122D	02055	02043	
DEND0123	00001	00001268	02082	02072	
DEND0124	00001	000012B7	02105	02093	
DEND0125	00001	000012FB	02127	02116	
DEND0127	00001	00001336	02154	02144	
DEND0128	00001	00001385	02177	02165	
DEND0129	00001	000013C9	02199	02188	
DEND0131	00001	00001404	02226	02216	
DEND0132	00001	00001453	02249	02237	
DEND0133	00001	00001497	02271	02260	
DEND0135	00001	000014D2	02298	02288	
DEND0136	00001	00001521	02321	02309	
DEND0137	00001	00001565	02343	02332	
DEND0139	00001	0000158E	02372	02360	
DEND0141	00001	000015BB	02401	02389	
DENT0107	00001	000010E0	01846	01848	
DENT0109	00001	000010FC	01873	01875	
DENT0111	00001	0000112F	01902	01904	
DENT0112	00001	0000115D	01922	01924	
DENT0114	00001	00001181	01949	01951	
DENT0116	00001	000011B2	01976	01978	
DENT0117	00001	000011D5	01996	01998	
DENT0121	00001	000011FE	02041	02043	
DENT0123	00001	00001230	02070	02072	
DENT0124	00001	00001268	02091	02093	
DENT0125	00001	000012B7	02114	02116	
DENT0127	00001	000012FE	02142	02144	
DENT0128	00001	00001336	02163	02165	
DENT0129	00001	00001385	02186	02188	
DENT0131	00001	000013CC	02214	02216	
DENT0132	00001	00001404	02235	02237	
DENT0133	00001	00001453	02258	02260	
DENT0135	00001	0000149A	02286	02288	
DENT0136	00001	000014D2	02307	02309	
DENT0137	00001	00001521	02330	02332	
DENT0139	00001	00001568	02358	02360	
DENT0141	00001	00001591	02387	02389	
DOUBLE	00008	00000070	02678	00318 00320 00327 00329 00336 00338 00432 00434 00442 00444 00463 00465 00482 00484 00509 00511 00525 00527 00538 00540 00659 00661	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
DSNAME	00001	0000001A	02472	00964 01218	
DSNBLMOV	00006	00000AFE	01069	00985	
DSNLENGT	00002	0000018E	02616	00550 00552 00981	
DSNLENOF	00002	00000190	02617	00555 00988	
DSNLNTYP	00002	00000164	02416	02411	
DSNPLN	00004	0000015C	02403	00976 02390	
DSNPLNK	00002	00000048	01761	01758	
DSORG	00001	0000000A	02456	01211	
DUMMA	00001	00000000	02877	00435 00439 00445 00448 00466 00469 00485 00488 00512 00515 00528 00531 00541 00544 00662	
DUMMD	00001	00000000	02876	00430 00657	
EDMASK	00016	00000B08	01071	00319 00328 00337 00433 00453 00464 00483 00510 00526 00539 00660	
EDMASK0	00016	00000B18	01072	00443	
ENDING	00004	00000068	01811	01805	
ENDKEY	00002	00000026	01339	01336	
ENDTONLY	00001	00000010	02610	00150 00177	
ENDTOTAL	00001	00000184	02608	00132 00135 00150 00157 00177 00190	
ENTKEND	00002	000008F4	00874	00871	
ENTKEY	00004	000008DA	00867	00873	
ENTTOTOUT	00001	00000080	02609	00132 00135	
EXPDT	00001	00000005	02451	01219	
EXT	00001	00000009	02455	01210	
FIRSTIM	00001	00000185	02612	00070 00727	
FORMATAD	00004	00000198	02692	00081 00085 00085	
FORMATK	00002	00000046	01740	01737	
FORMATSP	00004	00000150	02374	02361	
F100	00004	00000ACC	01054	00502 00506	
F127	00004	00000AD0	01055	00382	
GETKCOMP	00006	00000AC0	01047	01036	
GETKEY	00004	00000A6E	01022	00867	
GETKFND	00002	00000ABE	01046	01037	
GETKLOOP	00004	00000A88	01029	01041	
GETKNOTF	00002	00000ABC	01045	01027	
GETKSTD	00004	00000AA4	01036	01031 01034	
GOTENTRY	00004	000000E8	00173	00226	
GOTOT	00004	00000160	00221	00216	
HEADING	00004	000000E0	02028	00824 00840 00842 02024	
HEADK	00002	00000040	01669	00631 00703 01666	
H10	00002	00000ADE	01061	00813	
H12	00002	00000AE0	01062	00366 00958	
H2	00002	00000AD8	01058	00715	
H26	00002	00000AC6	01048	01040	
H32	00002	00000AE4	01064	00831	
H5	00002	00000ADA	01059	00686 00837	
H9	00002	00000ADC	01060	00982	
IECSDSL3	00001	00000770	02795	02796	
IECSDSL4	00001	0000070C	02754	02755	
IHADCB	00001	00000000	02884	00746 02969 03016 03081 03210 03225 03232 03245 03341 03347 03374 03397	
IKJ\$0053	00001	000015E6	02443	01281	
IKJ\$0055	00001	00000E95	01302	01298	
IKJ\$0109	00001	0000112C	01886	01885	
IKJ\$0121	00001	0000122D	02054	02053	
IKJ\$0124	00001	000012B7	02104	02103	
IKJ\$0125	00001	000012FB	02126	02125	
IKJ\$0128	00001	00001385	02176	02175	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
IKJ\$0129	00001	000013C9	02198	02197	
IKJ\$0132	00001	00001453	02248	02247	
IKJ\$0133	00001	00001497	02270	02269	
IKJ\$0136	00001	00001521	02320	02319	
IKJ\$0137	00001	00001565	02342	02341	
IKJ\$0139	00001	0000158E	02371	02370	
IKJ\$0141	00001	000015BB	02400	02399	
IKJ@0053	00001	00000168	02441	01282	
IKJ@0054	00001	0000007D	01311	01283	
IKJ@0055	00001	00000E4C	01297	01296	
IKJ@0057	00001	00000EA5	01326	01325	
IKJ@0059	00001	00000EB8	01347	01346	
IKJ@0061	00001	00000ECF	01368	01367	
IKJ@0062	00001	00000EDC	01381	01380	
IKJ@0063	00001	00000EE5	01392	01391	
IKJ@0065	00001	00000EF3	01412	01411	
IKJ@0067	00001	00000F02	01432	01431	
IKJ@0068	00001	00000F0F	01442	01441	
IKJ@0070	00001	00000F1F	01462	01461	
IKJ@0072	00001	00000F31	01483	01482	
IKJ@0074	00001	00000F42	01504	01503	
IKJ@0075	00001	00000F51	01518	01517	
IKJ@0077	00001	00000F66	01543	01542	
IKJ@0078	00001	00000F70	01553	01552	
IKJ@0080	00001	00000F81	01574	01573	
IKJ@0081	00001	00000F8B	01584	01583	
IKJ@0083	00001	00000F9D	01605	01604	
IKJ@0084	00001	00000FAB	01615	01614	
IKJ@0086	00001	00000FBB	01635	01634	
IKJ@0088	00001	00000FCD	01656	01655	
IKJ@0090	00001	00000FE1	01677	01676	
IKJ@0091	00001	00000FF1	01687	01686	
IKJ@0093	00001	00001002	01707	01706	
IKJ@0095	00001	00001015	01728	01727	
IKJ@0097	00001	00001026	01748	01747	
IKJ@0099	00001	00001039	01769	01768	
IKJ@0100	00001	00000258	01796	01777	
IKJ@0101	00001	00001070	01787	01786	
IKJ@0102	00001	0000028A	01817	01798	
IKJ@0103	00001	000010A2	01808	01807	
IKJ@0104	00001	000002C5	01838	01819	
IKJ@0105	00001	000010DD	01829	01828	
IKJ@0106	00001	000002E1	01865	01840	
IKJ@0107	00001	000010F8	01856	01855	
IKJ@0108	00001	00000314	01894	01867	
IKJ@0109	00001	00001129	01883	01882	
IKJ@0110	00001	00000366	01941	01896	
IKJ@0111	00001	0000115C	01912	01911	
IKJ@0112	00001	0000117D	01932	01931	
IKJ@0113	00001	00000397	01968	01943	
IKJ@0114	00001	000011AE	01959	01958	
IKJ@0115	00001	000003D9	02015	01970	
IKJ@0116	00001	000011D4	01986	01985	
IKJ@0117	00001	000011F0	02006	02005	
IKJ@0118	00001	000003E3	02033	02017	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
IKJ@0120	00001	00000415	02062	02035
IKJ@0121	00001	0000122A	02051	02050
IKJ@0122	00001	000004E3	02134	02064
IKJ@0123	00001	00001267	02080	02079
IKJ@0124	00001	0000128C	02101	02100
IKJ@0125	00001	000012D4	02124	02123
IKJ@0126	00001	000005B1	02206	02136
IKJ@0127	00001	00001335	02152	02151
IKJ@0128	00001	0000135A	02173	02172
IKJ@0129	00001	000013A2	02196	02195
IKJ@0130	00001	0000067F	02278	02208
IKJ@0131	00001	00001403	02224	02223
IKJ@0132	00001	00001428	02245	02244
IKJ@0133	00001	00001470	02268	02267
IKJ@0134	00001	0000074D	02350	02280
IKJ@0135	00001	000014D1	02296	02295
IKJ@0136	00001	000014F6	02317	02316
IKJ@0137	00001	0000153E	02340	02339
IKJ@0138	00001	00000776	02379	02352
IKJ@0139	00001	0000158B	02368	02367
IKJ@0140	00001	000007A3	02407	02381
IKJ@0141	00001	000015B7	02397	02396
IKJ@0142	00001	000015CA	02413	02412
IKJ@0143	00001	000015D7	02424	02423
IKJ@0144	00001	000015E5	02433	02432
IKJ00551	00001	00000E95	01301	01300
JFCB	00176	000006C4	02628	00741
KEND0056	00001	00000E9B	01316	01314
KEND0058	00001	00000EAD	01337	01335
KEND0060	00001	00000EC0	01358	01356
KEND0062	00001	00000EDC	01382	01378
KEND0064	00001	00000EEB	01402	01400
KEND0066	00001	00000EF9	01422	01420
KEND0069	00001	00000F15	01452	01450
KEND0071	00001	00000F27	01473	01471
KEND0073	00001	00000F39	01494	01492
KEND0076	00001	00000F5D	01533	01531
KEND0079	00001	00000F78	01564	01562
KEND0082	00001	00000F93	01595	01593
KEND0085	00001	00000FB1	01625	01623
KEND0087	00001	00000FC3	01646	01644
KEND0089	00001	00000FD5	01667	01665
KEND0092	00001	00000FF7	01697	01695
KEND0094	00001	0000100A	01718	01716
KEND0096	00001	0000101B	01738	01736
KEND0098	00001	0000102E	01759	01757
KEND0142	00001	000015CA	02414	02410
KEYW0056	00001	00000E95	01312	01314
KEYW0058	00001	00000EA7	01333	01335
KEYW0060	00001	00000EBA	01354	01356
KEYW0062	00001	00000ED1	01376	01378
KEYW0064	00001	00000EE5	01398	01400
KEYW0066	00001	00000EF3	01418	01420
KEYW0069	00001	00000F0F	01448	01450
KEYW0071	00001	00000F21	01469	01471

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
KEYW0073	00001	00000F33	01490	01492	
KEYW0076	00001	00000F57	01529	01531	
KEYW0079	00001	00000F72	01560	01562	
KEYW0082	00001	00000F8D	01591	01593	
KEYW0085	00001	00000FAB	01621	01623	
KEYW0087	00001	00000FBD	01642	01644	
KEYW0089	00001	00000FCF	01663	01665	
KEYW0092	00001	00000FF1	01693	01695	
KEYW0094	00001	00001004	01714	01716	
KEYW0096	00001	00001015	01734	01736	
KEYW0098	00001	00001028	01755	01757	
KEYW0142	00001	000015BB	02408	02410	
LASTKEY	00004	000000BC	02494	00180	00193 00202
LENSET	00002	00000846	00808	00789	00792 00800 00804 00806
LEVEL	00004	0000004C	01790	01784	
LEVKEY	00002	00000024	01318	01315	
LIMITK	00002	00000032	01475	01472	
LINECT	00002	000006D8	02745	00199	00636 00647 00681 00698 00700 00715 00787
LINELEN	00002	000006DE	02748	00374	00753 00790 00802 00812 00830 00848 00968
LINEMAX	00002	000006DA	02746	00199	00648 00705 00781 00786 00787 00807
LINESK	00002	0000003E	01648	01645	
LINESPP	00004	000000C4	01963	00782	00784 00805 01952
LRECL	00001	0000000D	02459	01213	
LSTUS	00001	00000004	02450	01216	
MOVEHEAD	00006	00000AF2	01067	00843	
MOVEHED	00006	00000AF8	01068	00838	
MOVETIT	00006	00000AE6	01065	00962	
MSGADDRS	00004	00000078	02680	00710	00719
MSGBL	00020	000005AC	02625	00208	00209
MSGBLC	00002	00000C90	01201	00208	
MSGTLEN	00001	0000004C	01199	00315	
MSGTOTC	00002	00000C44	01198	00315	01199
MSGWORK	00256	000001A2	02619	00315	00321 00324 00330 00333 00339 00342 00343 00344 00348 00349
NAME0057	00001	00000E9B	01322	01324	
NAME0059	00001	00000EAD	01343	01345	
NAME0061	00001	00000EC0	01364	01366	
NAME0063	00001	00000EDC	01388	01390	
NAME0065	00001	00000EEB	01408	01410	
NAME0067	00001	00000EF9	01428	01430	
NAME0068	00001	00000F04	01438	01440	
NAME0070	00001	00000F15	01458	01460	
NAME0072	00001	00000F27	01479	01481	
NAME0074	00001	00000F39	01500	01502	
NAME0075	00001	00000F49	01514	01516	
NAME0077	00001	00000F5D	01539	01541	
NAME0078	00001	00000F68	01549	01551	
NAME0080	00001	00000F78	01570	01572	
NAME0081	00001	00000F83	01580	01582	
NAME0083	00001	00000F93	01601	01603	
NAME0084	00001	00000F9F	01611	01613	
NAME0086	00001	00000FB1	01631	01633	
NAME0088	00001	00000FC3	01652	01654	
NAME0090	00001	00000FD5	01673	01675	
NAME0091	00001	00000FE3	01683	01685	
NAME0093	00001	00000FF7	01703	01705	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25															
NAME0095	00001	0000100A	01724	01726																
NAME0097	00001	0000101B	01744	01746																
NAME0099	00001	0000102E	01765	01767																
NAME0143	00001	000015CA	02420	02422																
NAME0144	00001	000015D7	02429	02431																
NEND0057	00001	00000EA7	01328	01324																
NEND0059	00001	00000EBA	01349	01345																
NEND0061	00001	00000ED1	01370	01366																
NEND0063	00001	00000EE5	01393	01390																
NEND0065	00001	00000EF3	01413	01410																
NEND0067	00001	00000F04	01434	01430																
NEND0068	00001	00000F0F	01443	01440																
NEND0070	00001	00000F21	01464	01460																
NEND0072	00001	00000F33	01485	01481																
NEND0074	00001	00000F49	01510	01502																
NEND0075	00001	00000F57	01524	01516																
NEND0077	00001	00000F68	01545	01541																
NEND0078	00001	00000F72	01555	01551																
NEND0080	00001	00000F83	01576	01572																
NEND0081	00001	00000F8D	01586	01582																
NEND0083	00001	00000F9F	01607	01603																
NEND0084	00001	00000FAB	01616	01613																
NEND0086	00001	00000FBD	01637	01633																
NEND0088	00001	00000FCF	01658	01654																
NEND0090	00001	00000FE3	01679	01675																
NEND0091	00001	00000FF1	01688	01685																
NEND0093	00001	00001004	01709	01705																
NEND0095	00001	00001015	01729	01726																
NEND0097	00001	00001028	01750	01746																
NEND0099	00001	0000103B	01771	01767																
NEND0143	00001	000015D7	02425	02422																
NEND0144	00001	000015E5	02434	02431																
NEWLIST	00004	000000DE	00166	00232																
NEXTLIST	00004	0000016C	00230	00168																
NOBREAK	00001	00000008	02611	00157 00190																
NOOPEN	00006	000007E6	00781	00738 00767																
NOOUTPT	00004	0000006A	00099	00094																
NUMBREAK	00002	000000C0	02495	00159 00192																
NUMTOTAL	00002	000000C2	02496	00152 00179																
OPENLIST	00004	00000074	02491	00252 00261 00740 00761 00769 01003 01005																
OUTDCB	00004	00000124	02557	00093 00215 00258 00263 00568 01002 01009 02607																
OUTDCBC	00004	00000BE4	01142	01002																
OUTDCBL	00001	00000060	02607	01002																
OUTMOVE	00006	00000598	00573	00566 00688																
OUTOPEN	00002	000008AE	00844	00832 00839																
OUTPUT	00004	00000572	00562	00095 00217																
OUTPUTK	00002	00000044	01720	01000 01717																
OUTWORK	00256	000005C0	02626	00562 00563 00563 00563 00569 00573 00683 00684 00684 00684 00691																
PAGEADDR	00004	00000070	02490	00656 00820																
PAGECT	00002	000006DC	02747	00653 00655																
PASS	00001	0000000E	02460	01223																
PCLMAIN	00001	00000E18	01280	01281 01284 01290 01310 01311 01321 01327 01332 01342 01348 01353 01363 01369 01375 01387																
				01397 01407 01417 01427 01433 01437 01447 01457 01463 01468 01478 01484 01489 01499 01505																
				01513 01519 01528 01538 01544 01548 01554 01559 01569 01575 01579 01585 01590 01600 01606																
				01610 01620 01630 01636 01641 01651 01657 01662 01672 01678 01682 01692 01702 01708 01713																

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25												
				01723 01733 01743 01749 01754 01764 01770 01775 01780 01795 01796 01801 01816 01817 01822													
				01837 01838 01843 01864 01865 01870 01893 01894 01899 01919 01940 01941 01946 01967 01968													
				01973 01993 02014 02015 02020 02032 02033 02038 02061 02062 02067 02088 02111 02133 02134													
				02139 02160 02183 02205 02206 02211 02232 02255 02277 02278 02283 02304 02327 02349 02350													
				02355 02378 02379 02384 02406 02407 02419 02428 02437 02442													
PCT	00001	00000008	02454	01209													
PDL	00001	00000000	01285	00058 01294 01304 01315 01317 01336 01338 01357 01359 01379 01383 01401 01403 01421 01423													
				01451 01453 01472 01474 01493 01495 01532 01534 01563 01565 01594 01596 01624 01626 01645													
				01647 01666 01668 01696 01698 01717 01719 01737 01739 01758 01760 01784 01789 01805 01810													
				01826 01831 01849 01859 01876 01888 01905 01915 01925 01935 01952 01962 01979 01989 01999													
				02009 02024 02027 02044 02056 02073 02083 02094 02106 02117 02128 02145 02155 02166 02178													
				02189 02200 02217 02227 02238 02250 02261 02272 02289 02299 02310 02322 02333 02344 02361													
				02373 02390 02402 02411 02415 02439 02441													
PDLFINI	00004	000005F8	00616	00594 00610													
PDLH10	00002	00000608	00621	00599													
PDLH240	00002	0000060A	00622	00597													
PDLLEAVE	00004	00000602	00619	00617													
PDLLOOP	00004	000005B2	00591	00601 00606 00608 00612													
PDLMINUS	00001	00000186	02613	00588 00611 00616													
PDLNSAVE	00004	0000007C	02492	00585 00619													
PDLNUM	00004	0000059E	00585	00147 00154 00751 00757 00785 00977													
PDLSP	00004	000005D8	00605	00598													
PEND0055	00001	00000E95	01303	01293													
PEND0101	00001	00001070	01788	01783													
PEND0103	00001	000010A2	01809	01804													
PEND0105	00001	000010DD	01830	01825													
PEND0119	00001	000011FB	02026	02023													
POST0055	00001	00000E1E	01291	01293													
POST0101	00001	0000103E	01781	01783													
POST0103	00001	00001073	01802	01804													
POST0105	00001	000010A5	01823	01825													
POST0119	00001	000011F4	02021	02023													
PRALLOC	00004	0000040A	00460	00418													
PRBLREC	00004	00000526	00536	00422													
PRDATES	00002	0000039A	00428	00416													
PRDMOVE	00002	00000564	00553	00551													
PRDSN	00004	00000554	00549	00423													
PREXT	00002	000004F6	00522	00421													
PRINT	00004	000002D8	00359	00101 00211													
PRINTK	00002	0000003A	01597	00099 00173 00737 01594													
PRINTR8	00004	00000064	02487	00359 00400													
PRLSTUS	00006	000003F4	00453	00417													
PRNTCLEN	00004	00000198	00249	00065													
PRNTCLO	00004	000001B6	00258	00250													
PRNTFREE	00004	000001D4	00270	00259													
PRNTFRL	00004	000001DC	00272	00284													
PRNTHEAD	00256	000003A8	02622	00670 00814 00816 00817 00817 00817 00826 01067 01068													
PRNTLIN	00004	000006BE	00683	00632 00649													
PRNTLINE	00004	0000060C	00628	00210 00350 00399													
PRNTLRET	00004	0000073A	00724	00694													
PRNTLSAV	00004	00000060	02486	00667 00678													
PRNTOT	00004	0000022E	00314	00134 00186 00236													
PRNTPAGE	00004	00000658	00653	00638													
PRNTSKP1	00004	00000648	00646	00643													
PRNTSKP2	00004	00000644	00645	00642													

SYMBOL	LEN	VALUE	DEFN	REFERENCES
PRNTSKP3	00004	00000640	00644	00640
PRNTTERM	00004	000006EA	00698	00630
PRNTTITH	00004	000004A8	02623	00708 00851
PRNTTITL	00256	000004AC	02624	00675 00951 00952 00952 00952 00953 00953 00955 00966 00999
PRNTTLIN	00004	00000726	00717	00704 00706
PRNTTOT8	00004	00000068	02488	00314 00352
PRNTWORK	00001	00000000	02478	00059 02636
PRPCT	00002	00000484	00493	00420
PRPCTERR	00006	000004A2	00500	00505 00507
PRPCTM	00004	000004AC	00502	00497
PRTEND	00004	0000034C	00398	00375
PRTICDEL	00004	00000940	00907	00918 00920
PRTICDLP	00006	0000091C	00896	00900
PRTICOPY	00002	00000916	00890	00908 00934
PRTIFND	00004	00000948	00916	00897
PRTINC	00004	00000340	00395	00449 00456 00470 00472 00480 00489 00501 00516 00518 00532 00545 00556
PRTINEW	00004	0000098A	00944	00885
PRTINIT	00004	00000740	00727	00072
PRTIREP	00004	00000970	00932	00940
PRTIREPA	00004	0000096C	00931	00922
PRTITSET	00002	00000992	00946	00857 00891
PRTLOOP	00002	000002EE	00364	00397
PRTLSAVE	00004	0000009C	02493	00628 00724
PRTMOVE	00006	0000035A	00402	00394 00554
PRTRTN	00002	00000360	00406	00390
PRTRTN1	00004	00000376	00415	00407
PRUNKN3	00006	000004EC	00517	00495 00499
PRUNKN6	00006	0000043C	00471	00461 00477
PRUNUSED	00004	00000446	00476	00419
RDJFCB	00004	000007B4	00761	00749
RECFM	00001	0000000B	02457	01212
ROUND	00001	00000013	02465	01222
R0	00001	00000000	00113	00372 00373 00376 00709 00709 00710 00718 00718 00719 00803 00803 00807 00866 00870 00916 00917 00966 00967 00970 00994
R1	00001	00000001	00114	00055 00146 00153 00180 00193 00201 00202 00209 00297 00298 00298 00299 00300 00301 00302 00303 00304 00305 00317 00318 00326 00327 00335 00336 00349 00363 00370 00371 00371 00373 00374 00376 00398 00555 00585 00586 00587 00587 00591 00619 00628 00637 00639 00641 00667 00678 00679 00685 00687 00701 00702 00708 00710 00712 00717 00719 00721 00724 00728 00729 00741 00742 00744 00747 00750 00756 00784 00799 00799 00801 00802 00812 00813 00814 00814 00818 00819 00819 00820 00830 00831 00834 00836 00837 00838 00840 00841 00843 00848 00849 00849 00850 00851 00854 00855 00883 00884 00919 00921 00944 00945 00955 00963 00963 00967 00968 00970 00975 00976 00986 00993 00994 00995 00996 01022 01023 01023 01029 01038 01038 01065 01066
R11	00001	0000000B	00124	00055 00056
R13	00001	0000000D	00126	00059
R14	00001	0000000E	00127	00380 00380 00381 00382 00391 00394 00406 00406 00415 00428 00428 00429 00432 00440 00442 00460 00463 00476 00478 00479 00482 00522 00522 00523 00525 00536 00538 00549 00550 00552 00553 00554 00714 00723 00833 00834 00836 01068
R15	00001	0000000F	00128	00148 00148 00151 00152 00155 00155 00158 00159 00383 00383 00384 00385 00402 00429 00440 00454 00460 00523 00536 00590 00590 00599 00600 00618 00618 00713 00714 00722 00723 00752 00753 00758 00766 00766 00786 00868 00978 00978 00980 00981 00982 00985 00986 00987 00987 00988 01024 01039 01039 01040 01045 01045
R2	00001	00000002	00115	00179 00181 00192 00194 00270 00272 00282 00282 00283 00283 00362 00365 00395 00395 00396 00564 00565 00565 00566 00586 00593 00636 00644 00644 00645 00645 00646 00646 00647 00648 00698 00699 00699 00700 00705 00733 00734 00735 00735 00736 00745 00746 00759 00763 00773

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25
SUB2OPER	00004	00000128	02251	02238	
SUB2VALU	00004	00000130	02273	02261	
SUB3KEY	00004	00000138	02300	02289	
SUB3OPER	00004	00000140	02323	02310	
SUB3VALU	00004	00000148	02345	02333	
SYSOUT	00004	000000C4	02502	00249 00254 00629 00669 00674 00690 00739 00745 00791 00997 02552	
SYSOUTC	00004	00000B84	01088	00739	
SYSOUTL	00001	00000060	02552	00739	
TABSPACE	00008	00000B59	01077	00735	
TABTITL	00012	00000CCC	01241	00367 00728 00959	
TITDSN	00002	000009EE	00975	00965	
TITDSN2	00004	00000A02	00981	00979	
TITEND	00004	00000A2E	00997	00974	
TITINC	00004	000009DE	00971	00989	
TITLOOP	00002	000009AA	00956	00973	
TITOVER	00002	00000A22	00993	00969	
TOTALK	00002	00000042	01699	01696	
TOTALLOC	00004	0000004C	02481	00300 00302 00326	
TOTALN	00004	000000E8	02057	00146 02044	
TOTDS	00004	00000048	02480	00297 00299 00317	
TOTLAST	00001	00000187	02614	00175 00200 00206 00351	
TOTUSED	00004	00000050	02482	00303 00305 00335	
TYPE	00001	00000014	02466	01225	
UNUSED	00001	00000007	02453	01208	
USERHEAD	00004	000008A0	00840	00825	
VADMSG	00004	0000001C	02659	00713 00722	
VOLS	00004	00000008	01305	01294	
VOLUME	00001	00000002	02448	01217	
VTCEPRNT	00001	000001A8	02705	00064	
VTCGETMN	00004	00000210	02734	00270 02735	
VTCGETMS	00001	00008000	02736	00278	
VTCGETMX	00001	00000032	02735	00271	
VTCSORTE	00001	000006D8	02741	00164	
VTCSORTH	00004	000002D8	02740	00163 00240 00240 00241 00241 00242 00242 00243 00243	
VTFACON	00008	0000003B	02849	01242	
VTFALLOC	00004	00000004	02818	00301 00478 00496 00503 01247 01248 01249	
VTFBLKSZ	00002	00000030	02835	01253	
VTFCATLG	00001	00000036	02840	01256	
VTFCREDT	00003	0000001C	02826	01244	
VTFDSN	00001	00000046	02852	00201 01066 01268	
VTFDSNL	00002	00000044	02850	00549 00564	
VTFDSORG	00003	00000026	02830	01251	
VTFDSTYP	00001	0000003A	02845	01261	
VTFEXPDT	00003	0000001F	02827	01246	
VTFRECL	00002	0000002E	02834	01254	
VTFRLSTAC	00003	00000022	02828	01245	
VTFMT	00001	00000000	02816	00082 01242 01243 01244 01245 01246 01247 01248 01249 01250 01251 01252 01253 01254 01255 01256 01257 01258 01259 01260 01261 01262 01263 01265 01266 01267 01268 02851	
VTFMTL	00001	00000046	02851	00565	
VTFNEXT	00004	00000000	02817	00225	
VTFNOEPV	00001	00000025	02829	01250	
VTFPROT	00001	00000033	02837	01255	
VTFRACF	00001	00000034	02838	01266	
VTFRECFM	00005	00000029	02832	01252	
VTFROUND	00001	00000032	02836	01260	

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES
VTFSECAL	00001	00000039	02842	01257
VTFSECAM	00002	00000037	02841	01258
VTFUNIT	00004	00000018	02825	01259
VTFUPD	00001	00000035	02839	01265
VTFUSED	00004	00000008	02821	00304 00476 00479 00494 00498 01262
VTFVOLUM	00006	00000012	02824	01243 01263 01267
VTOCOM	00001	00000000	02645	00056 00712 00721
VTOCPRNT	00001	00000000	00041	00048 01287 01307 01319 01329 01340 01350 01361 01371 01385 01394 01405 01414 01425 01435 01444 01455 01465 01476 01486 01497 01511 01525 01536 01546 01556 01567 01577 01587 01598 01608 01617 01628 01638 01649 01659 01670 01680 01689 01700 01710 01721 01730 01741 01751 01762 01772 01778 01792 01799 01813 01820 01834 01841 01861 01868 01890 01897 01917 01937 01944 01964 01971 01991 02011 02018 02029 02036 02058 02065 02085 02108 02130 02137 02157 02180 02202 02209 02229 02252 02274 02281 02301 02324 02346 02353 02375 02382 02404 02417 02426 02435 02444
VTPRNTEN	00001	000007A0	02632	00865 00880 00901 00916 00944
VTPRNTEX	00001	000007C8	02633	00866
VTPRNTL	00001	00000778	02631	00878
VTPRNTLS	00004	00000774	02630	00362 00855 00879 00945 00954
VTRET	00002	00000076	00106	00100 00133 00136 00244 00273 00285 01001 01013
WORKLINE	00256	000002A2	02620	00360 00361 00361 00363 00372 00398 00850
ZERO	00004	00000AC8	01053	00496 00498

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

=C'ERR'	00003	00000E10	03438	00500
---------	-------	----------	-------	-------

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 1044

TOTAL RECORDS READ FROM SYSTEM LIBRARY 7595

TOTAL RECORDS PUNCHED 104

TOTAL RECORDS PRINTED 1965

SYMBOL TYPE ID ADDR LENGTH LDID

ASM 0201 13.38 01/07/25

VTOCSORT SD 0001 000000 000148

PCLMAIN SD 0002 000148 0007CE

```
LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT                                ASM 0201 13.38 01/07/25
2  *****
3  *
4  *
5  * TITLE -      VTOC COMMAND  SORT  ROUTINE
6  *
7  * FUNCTION -   PUT THIS FORMATTED DSCB INTO THE SORTED LIST.
8  *
9  * OPERATION -  IF THIS IS A NOSORT RUN, JUST CALL THE PRINT ROUTINE.
10 *
11 *              TO BUILD THE SORTED LIST, FIRST DO A SIMPLE HASH
12 *              ON THE FIRST CHARACTER.  BUILD UP TO 256 SEPARATE
13 *              LISTS TO SAVE SORT TIME.  THEN SEARCH THROUGH THESE
14 *              LISTS SEQUENTIALLY.
15 * INPUT -     VTOC COMMON AREA ( VTOCOM )
16 *              POINTED TO BY REGISTER 1
17 *              USE PARSE DATA, CURRENT FORMATTED DSCB, SORTED LIST
18 *
19 * OUTPUT -    THE FORMATTED DSCB IS PLACED INTO THE SORTED LIST.
20 *
21 * ATTRIBUTES - REENTRANT, REUSEABLE, REFRESHABLE.
22 *
23 *
24 *              PROGRAMMED BY R. L. MILLER  (415) 485-6241
25 *
26 *
27 *
28 * MODIFIED 8/29/88 BY A. BRUCE LELAND AT HITACHI TO CORRECT  ABL-SOR
29 *              THE SORT FOR DATA SET NAMES.                  ABL-SOR
30 *
31 *****
```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT  ASM 0201 13.38 01/07/25

          33 VTOCSORT ENTER 12,24          DO THE HOUSEKEEPING
000000          34+VTOCSORT CSECT
000000 47F0 F00E          0000E          35+          B          14(0,15)          BRANCH AROUND ID          00450000
000004 08          36+          DC          AL1(8)          LENGTH OF IDENTIFIER          00550000
000005 E5E3D6C3E2D6D9E3          37+          DC          CL8'VTOCSORT'          IDENTIFIER          00750000
00000D 00
00000E 90EC D00C          0000C          38+          STM          14,12,12(13)          SAVE REGISTERS          03700000
000012 18CF          39+          LR          12,15          SET FIRST BASE REG
000014          40+          CNOP          0,4
          00000          41+          USING VTOCSORT,12
000014 5811 0018          00018          42+          L          1,24+0(1)          NUMERIC &SAVE IMPLIES A PASSED SAVEAREA
000018 50D1 0004          00004          43+          ST          13,4(1)          PRIOR SAVEAREA ADDRESS TO MINE
00001C 501D 0008          00008          44+          ST          1,8(13)          MY SAVEAREA ADDRESS TO HIS
000020 182D          45+          LR          2,13          KEEP THE SAVEAREA ADDRESS FOR REGS
000022 18D1          46+          LR          13,1          THIS IS MY SAVEAREA
000024 9802 2014          00014          47+          LM          0,2,20(2)          RESTORE ORIGINAL REGS
000028 18B1          48          LR          R11,R1          SAVE ADDR OF VTOCOM
          00000          49          USING VTOCOM,R11          SET ITS ADDRESSABILITY
00002A 5890 B048          00048          50          L          R9,ADDRANSR          POINT TO THE PARSE ANSWER
          00000          51          USING PDL,R9          SET ITS ADDRESSABILITY
          00000          52          USING SORTWORK,R13          SET ADDRESSABILITY FOR LOCAL WORK AREA

          54 *
          55 *          IS THIS A NOSORT RUN ?
          56 *          IF SO, JUST CALL PRINT
          57 *
00002E 9502 902F          0002F          58          CLI          SORTK+1,2          IS THIS NOSORT?
000032 4770 C044          00044          59          BNE          GOSORT          NO, KEEP ON TRUCKIN'
          60          VTCALL PRNT          YES, CALL PRINT AND GET OUT
000036 4110 B000          00000          61+          LA          R1,VTOCOM          POINT TO THE COMMON AREA
00003A 58F0 B030          00030          62+          L          R15,VADPRNT          POINT TO THE ROUTINE
00003E 05EF          63+          BALR          R14,R15          THEN CALL IT
000040 47F0 C11A          0011A          64          B          SORTRET          GET OUT OF HERE
          65 *
          66 *          PUT THIS ENTRY WHERE IT BELONGS
          67 *
000044 5830 B198          00198          68 GOSORT          L          R3,FORMATAD          POINT TO THE FORMATTED DSCB
          00000          69          USING VTFMT,R3          SET ADDRESSABILITY
000048 4160 B1C8          001C8          70          LA          R6,SORTTAB          POINT TO THE SORT FIELDS TABLE
00004C 1B44          71          SR          R4,R4
00004E 4340 6001          00001          72          IC          R4,1(0,R6)          LOAD HIGH KEY OFFSET
000052 4144 3000          00000          73          LA          R4,VTFMT(R4)          POINT TO HIGH KEY
000056 4120 B2CC          002CC          74          LA          R2,VTCSORTH-12          SORT HEADER AREA
00005A 4120 200C          0000C          75 GOSORT1          LA          R2,12(0,R2)          NEXT ENTRY
00005E 4850 2004          00004          76          LH          R5,4(0,R2)          LOAD COMAPRE LENGTH
000062 95C4 6000          00000          77          CLI          0(R6),C'D'          DESCENDING SORT
000066 4780 C07E          0007E          78          BE          GOSORT3          YES
00006A 47F0 C08A          0008A          79          B          GOSORT4          NO
00006E BF5F 2000          00000          80 GOSORT2          ICM          R5,B'1111',0(R2)          GET THE HEAD OF THE LIST
000072 4770 C096          00096          81          BNZ          NOTFIRST          IF NON-ZERO, SEARCH THE LIST
          82 *
          83 *          FIRST ENTRY ON THE LIST, IT'S EASY
          84 *

```

```

          ASM 0201 13.38 01/07/25
LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
000076 5032 0000      00000      85      ST      R3,0(R2)      START UP THE LIST
00007A 47F0 C11A      0011A      86      B        SORTRET      THEN RETURN
00007E 4450 C138      00138      87  GOSORT3  EX      R5,GOSORTCL  COMPARE TO GET CORRECT LIST
000082 4740 C05A      0005A      88      BL      GOSORT1
000086 47F0 C06E      0006E      89      B        GOSORT2
00008A 4450 C138      00138      90  GOSORT4  EX      R5,GOSORTCL  COMPARE TO GET CORRECT LIST
00008E 4720 C05A      0005A      91      BH      GOSORT1
000092 47F0 C06E      0006E      92      B        GOSORT2
          93 *
          94 *          FIND A SLOT FOR THIS ENTRY
          95 *          FIRST GET THE SHORTER DSN LENGTH
          96 *
000096 1B11          97  NOTFIRST SR      R1,R1
000098 4310 6001      00001      98      IC      R1,1(0,R6)  OFFSET OF SORT FIELD
00009C 4171 5000      00000      99      LA      R7,0(R1,R5)  LOAD PREV ENTRY FIELD ADDR
0000A0 4181 3000      00000      100     LA      R8,0(R1,R3)  LOAD NEW ENTRY FIELD ADDR
0000A4 5910 C140      00140      101     C        R1,=A(VTFDSN-VTFMT)  DSN
0000A8 4770 C0C0      000C0      102     BNE     NOTFRST1
          103 *** LENGTH; IT SHOULD ALWAYS USE LENGTH 44 FOR DATA SET NAMES.  ABL-SOR
0000AC 4810 3044      00044      104     LH      R1,VTFDSNL-VTFMT(0,R3)
0000B0 4910 5044      00044      105     CH      R1,VTFDSNL-VTFMT(0,R5)
0000B4 47D0 C0BC      000BC      106     BNH     NOTFRST0
0000B8 4810 5044      00044      107     LH      R1,VTFDSNL-VTFMT(0,R5)
          108 ***FRST0 BCTR  R1,0          ABL-SOR
0000BC          109  NOTFRST0 DS      0H          ***COMPARE FOLLOWING BLANK TOO  ABL-SOR
0000BC 47F0 C0C4      000C4      110     B        NOTFRST2
0000C0 4810 6002      00002      111  NOTFRST1 LH      R1,2(0,R6)  LOAD SORT FIELD EXEC LENGTH
0000C4 95C4 6000      00000      112  NOTFRST2 CLI     0(R6),C'D'  DESCENDING SORT
0000C8 4780 C0DC      000DC      113     BE      NOTFRST4  YES
0000CC 4410 C132      00132      114  NOTFRST3 EX      R1,COMPVTF  COMPARE THE FIELDS
0000D0 4740 C106      00106      115     BL      NEXTENT  LIST ENTRY IS LOWER, UP THE CHAIN
0000D4 4780 C0F4      000F4      116     BE      CHECKNXT  IDENTICAL, CHECK NEXT FIELD
0000D8 47F0 C0E8      000E8      117     B        INSERT
0000DC 4410 C132      00132      118  NOTFRST4 EX      R1,COMPVTF  COMPARE THE FIELDS
0000E0 4720 C106      00106      119     BH      NEXTENT  LIST ENTRY IS LOWER, UP THE CHAIN
0000E4 4780 C0F4      000F4      120     BE      CHECKNXT  IDENTICAL, CHECK NEXT FIELD
          121 *
          122 *          THE NEW ENTRY GOES HERE
          123 *
0000E8 5032 0000      00000      124  INSERT  ST      R3,0(R2)  SAVE THE NEW POINTER
0000EC 5050 3000      00000      125     ST      R5,VTFNEXT  JUST BEFORE THIS LIST ENTRY
0000F0 47F0 C11A      0011A      126     B        SORTRET  THEN EXIT
          127 *
          128 *
0000F4 4160 6004      00004      129  CHECKNXT LA      R6,4(0,R6)  NEXT SORT FIELD
0000F8 D503 6000  C144 00000 00144 130     CLC     0(4,R6),=F'0'  ANY MORE FIELDS
0000FE 4780 C0E8      000E8      131     BE      INSERT  NO, PUT IT HERE
000102 47F0 C096      00096      132     B        NOTFIRST  YES, CHECK IT
          133 *
          134 *          GET THE NEXT ENTRY ON THIS LIST
          135 *
000106 4125 0000      00000      136  NEXTENT LA      R2,VTFNEXT-VTFMT(R5)  POINT BACK TO THIS ENTRY
00010A 4160 B1C8      001C8      137     LA      R6,SORTTAB  RELOAD SORT FIELD TABLE ADDR
00010E BF5F 5000      00000      138     ICM     R5,B'1111',VTFNEXT-VTFMT(R5)  GET THE NEXT ENTRY
000112 4770 C096      00096      139     BNZ     NOTFIRST  THERE IS ONE, CHECK IT

```



```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
000116 5032 0000      00000      140      ST   R3,0(R2)      LAST ENTRY ON THE LIST, PUT IT THERE
      141 *
      142 *      RETURN
      143 *
      144 SORTRET LEAVE EQ,RC=0
00011A 182D      145+SORTRET LR   2,13
00011C 58DD 0004      00004      146+      L    13,4(13)
000120 41F0 0000      00000      147+      LA   15,0          LOAD THE RETURN CODE
000124 90F1 D010      00010      148+      STM 15,1,16(13)   STORE RETURN REGS
000128 98EC D00C      0000C      149+      LM  14,12,12(13)  RESTORE THE REGISTERS 00650000
00012C 92FF D00C      0000C      150+      MVI 12(13),X'FF'  SET RETURN INDICATION 01600000
000130 07FE      151+      BR   14          RETURN 02000000
      00000      152+R0      EQU  0          *USED BY O.S.
      00001      153+R1      EQU  1          *USED BY O.S. // ADDRESS OF PARAMETER LIST
      00002      154+R2      EQU  2
      00003      155+R3      EQU  3
      00004      156+R4      EQU  4
      00005      157+R5      EQU  5
      00006      158+R6      EQU  6
      00007      159+R7      EQU  7
      00008      160+R8      EQU  8
      00009      161+R9      EQU  9
      0000A      162+R10     EQU 10
      0000B      163+R11     EQU 11
      0000C      164+R12     EQU 12
      0000D      165+R13     EQU 13          *USED BY O.S. // SAVE-AREA ADDRESS
      0000E      166+R14     EQU 14          *USED BY O.S. // RETURN ADDRESS
      0000F      167+R15     EQU 15          *USED BY O.S. // ENTRY-PT ADDR, RETURN CODE
      168 *
      169 *
      170 *
      171 *      PROGRAM CONSTANTS
      172 *
000132 D500 7000 8000 00000 00000 173 COMPVTF CLC  0(0,R7),0(R8)   EXECUTED COMPARE
000138 D500 4000 2006 00000 00006 174 GOSORTCL CLC  0(0,R4),6(R2)   EXECUTED COMPARE
      175 *
      176 *
      177      PRINT NOGEN

```

```

LOC  OBJECT CODE  ADDR1 ADDR2  STMT  SOURCE STATEMENT
                                           ASM 0201 13.38 01/07/25
                                           179 *
                                           180 *
                                           181 *      P A R S E   C O N T R O L   L I S T
                                           182 *
                                           183 *
                                           184      PRINT OFF
                                           186      PUSH  PRINT
                                           187      PRINT NOGEN
1355      POP   PRINT
1382      PRINT ON
1383 *
1384 *      DYNAMIC WORK AREA
1385 *

000000      1387 SORTWORK DSECT
000000      1388      DS    18A      PRINT ROUTINE SAVE AREA

000048      1390      DS    0D
           00048 1391 LENWORK EQU   *-SORTWORK
           1392 *
           1393 *      VTOC COMMAND COMMON AREA
           1394 *
           1395      PRINT NOGEN
           1396      VTOCOM

           1564 *
           1565 *      FORMATTED DSCB
           1566 *
           1567      VTFMT

           1609      PDEDSNAM

           1631      END
000140 00000046      1632      =A(VTFDSN-VTFMT)
000144 00000000      1633      =F'0'

```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

ADDRANSR	00004	00000048	01429	00050
AL00271	00001	00000279	00417	00416
AL00281	00001	00000287	00431	00430
ANDOR1K	00002	00000034	00405	00402
ANDOR2K	00002	00000036	00444	00441
ANDOR3K	00002	00000038	00475	00472
BLKSZSET	00004	000000BC	00845	00834
BREAK	00004	000000AC	00798	00785
BREAKK	00002	00000030	00363	00360
CATK	00002	0000002C	00313	00310
CHARSK	00002	0000003C	00536	00533
CHARSPL	00004	000000B4	00825	00814
CHECKNXT	00004	000000F4	00129	00116 00120
COMPVTF	00006	00000132	00173	00114 00118
CONTAIN	00004	00000084	00741	00735
CONTAINK	00002	00000028	00269	00266
DEND0060	00001	00000429	00767	00757
DEND0062	00001	0000045C	00796	00784
DEND0064	00001	0000048D	00823	00813
DEND0065	00001	000004AE	00843	00833
DEND0067	00001	000004DF	00870	00860
DEND0069	00001	00000505	00897	00887
DEND0070	00001	00000521	00917	00907
DEND0074	00001	0000055D	00964	00952
DEND0076	00001	00000598	00991	00981
DEND0077	00001	000005E7	01014	01002
DEND0078	00001	0000062B	01036	01025
DEND0080	00001	00000666	01063	01053
DEND0081	00001	000006B5	01086	01074
DEND0082	00001	000006F9	01108	01097
DEND0084	00001	00000734	01135	01125
DEND0085	00001	00000783	01158	01146
DEND0086	00001	000007C7	01180	01169
DEND0088	00001	00000802	01207	01197
DEND0089	00001	00000851	01230	01218
DEND0090	00001	00000895	01252	01241
DEND0092	00001	000008BE	01281	01269
DEND0094	00001	000008EB	01310	01298
DENT0060	00001	00000410	00755	00757
DENT0062	00001	0000042C	00782	00784
DENT0064	00001	0000045F	00811	00813
DENT0065	00001	0000048D	00831	00833
DENT0067	00001	000004B1	00858	00860
DENT0069	00001	000004E2	00885	00887
DENT0070	00001	00000505	00905	00907
DENT0074	00001	0000052E	00950	00952
DENT0076	00001	00000560	00979	00981
DENT0077	00001	00000598	01000	01002
DENT0078	00001	000005E7	01023	01025
DENT0080	00001	0000062E	01051	01053
DENT0081	00001	00000666	01072	01074
DENT0082	00001	000006B5	01095	01097
DENT0084	00001	000006FC	01123	01125
DENT0085	00001	00000734	01144	01146
DENT0086	00001	00000783	01167	01169

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

DENT0088	00001	000007CA	01195	01197
DENT0089	00001	00000802	01216	01218
DENT0090	00001	00000851	01239	01241
DENT0092	00001	00000898	01267	01269
DENT0094	00001	000008C1	01296	01298
DSNLNTYP	00002	00000164	01325	01320
DSNPLN	00004	0000015C	01312	01299
DSNPLNK	00002	00000048	00670	00667
ENDING	00004	00000068	00720	00714
ENDKEY	00002	00000026	00248	00245
FORMATAD	00004	00000198	01447	00068
FORMATK	00002	00000046	00649	00646
FORMATSP	00004	00000150	01283	01270
GOSORT	00004	00000044	00068	00059
GOSORTCL	00006	00000138	00174	00087 00090
GOSORT1	00004	0000005A	00075	00088 00091
GOSORT2	00004	0000006E	00080	00089 00092
GOSORT3	00004	0000007E	00087	00078
GOSORT4	00004	0000008A	00090	00079
HEADING	00004	000000E0	00937	00933
HEADK	00002	00000040	00578	00575
IECSDSL3	00001	00000770	01550	01551
IECSDSL4	00001	0000070C	01509	01510
IKJ\$0006	00001	00000916	01352	00190
IKJ\$0008	00001	000001C5	00211	00207
IKJ\$0062	00001	0000045C	00795	00794
IKJ\$0074	00001	0000055D	00963	00962
IKJ\$0077	00001	000005E7	01013	01012
IKJ\$0078	00001	0000062B	01035	01034
IKJ\$0081	00001	000006B5	01085	01084
IKJ\$0082	00001	000006F9	01107	01106
IKJ\$0085	00001	00000783	01157	01156
IKJ\$0086	00001	000007C7	01179	01178
IKJ\$0089	00001	00000851	01229	01228
IKJ\$0090	00001	00000895	01251	01250
IKJ\$0092	00001	000008BE	01280	01279
IKJ\$0094	00001	000008EB	01309	01308
IKJ@0006	00001	00000168	01350	00191
IKJ@0007	00001	0000007D	00220	00192
IKJ@0008	00001	0000017C	00206	00205
IKJ@0010	00001	000001D5	00235	00234
IKJ@0012	00001	000001E8	00256	00255
IKJ@0014	00001	000001FF	00277	00276
IKJ@0015	00001	0000020C	00290	00289
IKJ@0016	00001	00000215	00301	00300
IKJ@0018	00001	00000223	00321	00320
IKJ@0020	00001	00000232	00341	00340
IKJ@0021	00001	0000023F	00351	00350
IKJ@0023	00001	0000024F	00371	00370
IKJ@0025	00001	00000261	00392	00391
IKJ@0027	00001	00000272	00413	00412
IKJ@0028	00001	00000281	00427	00426
IKJ@0030	00001	00000296	00452	00451
IKJ@0031	00001	000002A0	00462	00461
IKJ@0033	00001	000002B1	00483	00482

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

IKJ@0034	00001	000002BB	00493	00492
IKJ@0036	00001	000002CD	00514	00513
IKJ@0037	00001	000002DB	00524	00523
IKJ@0039	00001	000002EB	00544	00543
IKJ@0041	00001	000002FD	00565	00564
IKJ@0043	00001	00000311	00586	00585
IKJ@0044	00001	00000321	00596	00595
IKJ@0046	00001	00000332	00616	00615
IKJ@0048	00001	00000345	00637	00636
IKJ@0050	00001	00000356	00657	00656
IKJ@0052	00001	00000369	00678	00677
IKJ@0053	00001	00000258	00705	00686
IKJ@0054	00001	000003A0	00696	00695
IKJ@0055	00001	0000028A	00726	00707
IKJ@0056	00001	000003D2	00717	00716
IKJ@0057	00001	000002C5	00747	00728
IKJ@0058	00001	0000040D	00738	00737
IKJ@0059	00001	000002E1	00774	00749
IKJ@0060	00001	00000428	00765	00764
IKJ@0061	00001	00000314	00803	00776
IKJ@0062	00001	00000459	00792	00791
IKJ@0063	00001	00000366	00850	00805
IKJ@0064	00001	0000048C	00821	00820
IKJ@0065	00001	000004AD	00841	00840
IKJ@0066	00001	00000397	00877	00852
IKJ@0067	00001	000004DE	00868	00867
IKJ@0068	00001	000003D9	00924	00879
IKJ@0069	00001	00000504	00895	00894
IKJ@0070	00001	00000520	00915	00914
IKJ@0071	00001	000003E3	00942	00926
IKJ@0073	00001	00000415	00971	00944
IKJ@0074	00001	0000055A	00960	00959
IKJ@0075	00001	000004E3	01043	00973
IKJ@0076	00001	00000597	00989	00988
IKJ@0077	00001	000005BC	01010	01009
IKJ@0078	00001	00000604	01033	01032
IKJ@0079	00001	000005B1	01115	01045
IKJ@0080	00001	00000665	01061	01060
IKJ@0081	00001	0000068A	01082	01081
IKJ@0082	00001	000006D2	01105	01104
IKJ@0083	00001	0000067F	01187	01117
IKJ@0084	00001	00000733	01133	01132
IKJ@0085	00001	00000758	01154	01153
IKJ@0086	00001	000007A0	01177	01176
IKJ@0087	00001	0000074D	01259	01189
IKJ@0088	00001	00000801	01205	01204
IKJ@0089	00001	00000826	01226	01225
IKJ@0090	00001	0000086E	01249	01248
IKJ@0091	00001	00000776	01288	01261
IKJ@0092	00001	000008BB	01277	01276
IKJ@0093	00001	000007A3	01316	01290
IKJ@0094	00001	000008E7	01306	01305
IKJ@0095	00001	000008FA	01322	01321
IKJ@0096	00001	00000907	01333	01332
IKJ@0097	00001	00000915	01342	01341

SYMBOL	LEN	VALUE	DEFN	REFERENCES
IKJ00081	00001	000001C5	00210	00209
INSERT	00004	000000E8	00124	00117 00131
KEND0009	00001	000001CB	00225	00223
KEND0011	00001	000001DD	00246	00244
KEND0013	00001	000001F0	00267	00265
KEND0015	00001	0000020C	00291	00287
KEND0017	00001	0000021B	00311	00309
KEND0019	00001	00000229	00331	00329
KEND0022	00001	00000245	00361	00359
KEND0024	00001	00000257	00382	00380
KEND0026	00001	00000269	00403	00401
KEND0029	00001	0000028D	00442	00440
KEND0032	00001	000002A8	00473	00471
KEND0035	00001	000002C3	00504	00502
KEND0038	00001	000002E1	00534	00532
KEND0040	00001	000002F3	00555	00553
KEND0042	00001	00000305	00576	00574
KEND0045	00001	00000327	00606	00604
KEND0047	00001	0000033A	00627	00625
KEND0049	00001	0000034B	00647	00645
KEND0051	00001	0000035E	00668	00666
KEND0095	00001	000008FA	01323	01319
KEYW0009	00001	000001C5	00221	00223
KEYW0011	00001	000001D7	00242	00244
KEYW0013	00001	000001EA	00263	00265
KEYW0015	00001	00000201	00285	00287
KEYW0017	00001	00000215	00307	00309
KEYW0019	00001	00000223	00327	00329
KEYW0022	00001	0000023F	00357	00359
KEYW0024	00001	00000251	00378	00380
KEYW0026	00001	00000263	00399	00401
KEYW0029	00001	00000287	00438	00440
KEYW0032	00001	000002A2	00469	00471
KEYW0035	00001	000002BD	00500	00502
KEYW0038	00001	000002DB	00530	00532
KEYW0040	00001	000002ED	00551	00553
KEYW0042	00001	000002FF	00572	00574
KEYW0045	00001	00000321	00602	00604
KEYW0047	00001	00000334	00623	00625
KEYW0049	00001	00000345	00643	00645
KEYW0051	00001	00000358	00664	00666
KEYW0095	00001	000008EB	01317	01319
LEVEL	00004	0000004C	00699	00693
LEVKEY	00002	00000024	00227	00224
LIMITK	00002	00000032	00384	00381
LINESK	00002	0000003E	00557	00554
LINESPP	00004	000000C4	00872	00861
NAME0010	00001	000001CB	00231	00233
NAME0012	00001	000001DD	00252	00254
NAME0014	00001	000001F0	00273	00275
NAME0016	00001	0000020C	00297	00299
NAME0018	00001	0000021B	00317	00319
NAME0020	00001	00000229	00337	00339
NAME0021	00001	00000234	00347	00349
NAME0023	00001	00000245	00367	00369

ASM 0201 13.38 01/07/25

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM 0201 13.38 01/07/25													
				00422 00428 00437 00447 00453 00457 00463 00468 00478 00484 00488 00494 00499 00509 00515														
				00519 00529 00539 00545 00550 00560 00566 00571 00581 00587 00591 00601 00611 00617 00622														
				00632 00642 00652 00658 00663 00673 00679 00684 00689 00704 00705 00710 00725 00726 00731														
				00746 00747 00752 00773 00774 00779 00802 00803 00808 00828 00849 00850 00855 00876 00877														
				00882 00902 00923 00924 00929 00941 00942 00947 00970 00971 00976 00997 01020 01042 01043														
				01048 01069 01092 01114 01115 01120 01141 01164 01186 01187 01192 01213 01236 01258 01259														
				01264 01287 01288 01293 01315 01316 01328 01337 01346 01351														
PDL	00001	00000000	00194	00051 00203 00213 00224 00226 00245 00247 00266 00268 00288 00292 00310 00312 00330 00332														
				00360 00362 00381 00383 00402 00404 00441 00443 00472 00474 00503 00505 00533 00535 00554														
				00556 00575 00577 00605 00607 00626 00628 00646 00648 00667 00669 00693 00698 00714 00719														
				00735 00740 00758 00768 00785 00797 00814 00824 00834 00844 00861 00871 00888 00898 00908														
				00918 00933 00936 00953 00965 00982 00992 01003 01015 01026 01037 01054 01064 01075 01087														
				01098 01109 01126 01136 01147 01159 01170 01181 01198 01208 01219 01231 01242 01253 01270														
				01282 01299 01311 01320 01324 01348 01350														
PEND0008	00001	000001C5	00212	00202														
PEND0054	00001	000003A0	00697	00692														
PEND0056	00001	000003D2	00718	00713														
PEND0058	00001	0000040D	00739	00734														
PEND0072	00001	0000052B	00935	00932														
POST0008	00001	0000014E	00200	00202														
POST0054	00001	0000036E	00690	00692														
POST0056	00001	000003A3	00711	00713														
POST0058	00001	000003D5	00732	00734														
POST0072	00001	00000524	00930	00932														
PRINTK	00002	0000003A	00506	00503														
R1	00001	00000001	00153	00048 00061 00097 00097 00098 00099 00100 00101 00104 00105 00107 00111 00114 00118														
R11	00001	0000000B	00163	00048 00049														
R13	00001	0000000D	00165	00052														
R14	00001	0000000E	00166	00063														
R15	00001	0000000F	00167	00062 00063														
R2	00001	00000002	00154	00074 00075 00075 00076 00080 00085 00124 00136 00140 00174														
R3	00001	00000003	00155	00068 00069 00085 00100 00104 00124 00140														
R4	00001	00000004	00156	00071 00071 00072 00073 00073 00174														
R5	00001	00000005	00157	00076 00080 00087 00090 00099 00105 00107 00125 00136 00138 00138														
R6	00001	00000006	00158	00070 00072 00077 00098 00111 00112 00129 00129 00130 00137														
R7	00001	00000007	00159	00099 00173														
R8	00001	00000008	00160	00100 00173														
R9	00001	00000009	00161	00050 00051														
SORTK	00002	0000002E	00333	00058 00330														
SORTRET	00002	0000011A	00145	00064 00086 00126														
SORTTAB	00004	000001C8	01477	00070 00137														
SORTWORK	00001	00000000	01387	00052 01391														
SPACEK	00002	0000002A	00293	00288														
SUBAO1	00002	0000062C	01045	00414 00428														
SUBAO2	00002	000006FA	01117	00453 00463														
SUBAO3	00002	000007C8	01189	00484 00494														
SUBBREAK	00002	0000042A	00776	00372														
SUBCHARS	00002	0000045D	00805	00545														
SUBCONT	00002	000003D3	00728	00278														
SUBDSNLN	00002	000008BF	01290	00679														
SUBEND	00002	000003A1	00707	00257														
SUBFORMT	00002	00000896	01261	00658														
SUBHEAD	00002	00000522	00926	00587														
SUBLEV	00002	0000036C	00686	00236														
SUBLIMIT	00002	0000055E	00973	00393														

SYMBOL	LEN	VALUE	DEFN	REFERENCES
--------	-----	-------	------	------------

ASM 0201 13.38 01/07/25

=A(VTFDSN-VTFMT)				
	00004	00000140	01632	00101
=F'0'	00004	00000144	01633	00130

ASM 0201 13.38 01/07/25

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

HIGHEST SEVERITY WAS 0

OPTIONS FOR THIS ASSEMBLY

ALIGN, ALOGIC, BUFSIZE(STD), DECK, ESD, FLAG(0), LINECOUNT(55), LIST, NOMCALL, YFLAG, WORKSIZE(2097152)

NOMLOGIC, NONUMBER, NOOBJECT, NORENT, RLD, NOSTMT, NOLIBMAC, NOTERMINAL, NOTEST, XREF(SHORT)

SYSPARM()

WORK FILE BUFFER SIZE/NUMBER =32758/ 1

TOTAL RECORDS READ FROM SYSTEM INPUT 169

TOTAL RECORDS READ FROM SYSTEM LIBRARY 2861

TOTAL RECORDS PUNCHED 45

TOTAL RECORDS PRINTED 624

F64-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED XREF,LET,LIST,SIZE=(600K,64K),RENT,REUS,REFR
VARIABLE OPTIONS USED - SIZE=(491520,55296)

IEW0000 ENTRY VTOCCMD
IEW0000 NAME VTOC(R)

CROSS REFERENCE TABLE

CONTROL SECTION

ENTRY

NAME	ORIGIN	LENGTH	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION
VTOCCMD	00	120C								
PCLMAIN	1210	7CE								
VTOCCHEK	19E0	B4D								
VTOCEXCP	2530	754								
VTOCFORM	2C88	53A								
VTOCMSG	31C8	E6								
VTOCPRNT	32B0	E13								
VTOCSORT	40C8	148								

LOCATION REFERS TO SYMBOL IN CONTROL SECTION

LOCATION REFERS TO SYMBOL IN CONTROL SECTION

5F8	VTOCMSG	VTOCMSG	600	VTOCEXCP	VTOCEXCP
604	VTOCCHEK	VTOCCHEK	608	VTOCFORM	VTOCFORM
60C	VTOCPRNT	VTOCPRNT	610	VTOCSORT	VTOCSORT
614	PCLMAIN	PCLMAIN			
ENTRY ADDRESS	00				

TOTAL LENGTH 4210

***VTOC NOW REPLACED IN DATA SET

AUTHORIZATION CODE IS 0.

**MODULE HAS BEEN MARKED REENTERABLE, REUSABLE, AND REFRESHABLE.