J E S 2 J O B L O G

```
18.22.46 JOB  159  IEF677I WARNING MESSAGE(S) FOR JOB VSTESTR4 ISSUED
18.22.46 JOB  159  $HASP373 VSTESTR4 STARTED - INIT  1 - CLASS A - SYS HMVS
18.22.46 JOB  159  IEF403I VSTESTR4 - STARTED - TIME=18.22.46
18.22.47 JOB  159  CCI001C PL1L    /IEMAA   /00:00:00.16/        /00004/SYS     /VSTESTR4
18.22.47 JOB  159  CCI001C LKED    /IEWL    /00:00:00.05/        /00000/SYS     /VSTESTR4
18.22.47 JOB  159  CCI001C GO      /PGM=*.DD/00:00:00.03/        /00000/SYS     /VSTESTR4
18.22.47 JOB  159  IEF404I VSTESTR4 - ENDED - TIME=18.22.47
18.22.47 JOB  159  $HASP395 VSTESTR4 ENDED

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

 07 JUL 20 JOB EXECUTION DATE

       23 CARDS READ

    1,069 SYSOUT PRINT RECORDS

        0 SYSOUT PUNCH RECORDS

     0.01 MINUTES EXECUTION TIME
```

```
 1      //VSTESTR4 JOB (SYS),'VSAMIOP IVP RRDSLODR',CLASS=A,MSGCLASS=X,         JOB  159
        //             REGION=4096K
        ***
        ********************************************************************
        *** PL/1 MODULE: RRDSLODR     VSAM DATASET: VSTESTRR.CLUSTER (RRDS)
        ***
        *** RANDOMLY LOADS RECORDS LEAVING EMPTY "DUMMY" RECORD SLOTS
        ********************************************************************
        ***
 2      //PL1F    EXEC PL1LFCLG,
        //             PARM='LOAD,NODECK,ATR,XREF,CHAR60,MACRO'
 3      XXPL1L   EXEC  PGM=IEMAA,PARM='LOAD,NODECK',REGION=52K                  00000100
 4      XXSTEPLIB  DD DSN=SYSC.LINKLIB,DISP=SHR                                 00000200
 5      //PL1L.SYSPRINT DD SYSOUT=*
        X/SYSPRINT DD  SYSOUT=A                                                 00000300
 6      XXSYSLIN   DD  DSNAME=&&LOADSET,DISP=(MOD,PASS),UNIT=SYSSQ,            *00000400
        XX             SPACE=(80,(250,100))                                     00000500
 7      XXSYSUT3  DD  DSNAME=&&SYSUT3,UNIT=SYSDA,SPACE=(80,(250,250)),          *00000600
        XX             DCB=BLKSIZE=80                                           00000700
 8      XXSYSUT1   DD  DSNAME=&&SYSUT1,UNIT=SYSDA,SPACE=(1024,(60,60),,CONTIG),*00000800
        XX             SEP=(SYSUT3,SYSLIN),DCB=BLKSIZE=1024                     00000900
 9      //PL1L.SYSIN DD DSN=SYSC.VSAMIOP.SOURCE(RRDSLODR),DISP=SHR
10      //PL1L.SYSLIB DD DSN=SYSC.VSAMIOP.MACLIB,DISP=SHR
11      XXLKED   EXEC  PGM=IEWL,PARM='XREF,LIST',COND=(9,LT,PL1L),              *00001000
        XX             REGION=96K                                               00001100
12      //LKED.SYSLIB DD
        X/SYSLIB   DD  DSNAME=SYSC.PL1LIB,DISP=SHR                              00001201
13      //         DD  DSN=SYSC.LINKLIB,DISP=SHR
14      XXSYSLMOD  DD  DSNAME=&&GOSET(GO),DISP=(MOD,PASS),                      *00001300
        XX             UNIT=SYSDA,SPACE=(1024,(50,20,1),RLSE)                   00001400
15      XXSYSUT1   DD  DSNAME=&&SYSUT1,UNIT=SYSDA,SPACE=(1024,(200,20)),        *00001500
        XX             SEP=(SYSLMOD,SYSLIB),DCB=BLKSIZE=1024                    00001600
16      //LKED.SYSPRINT DD SYSOUT=*
        X/SYSPRINT DD  SYSOUT=A                                                 00001700
17      XXSYSLIN   DD  DSNAME=&&LOADSET,DISP=(OLD,DELETE)                       00001800
18      XX         DD  DDNAME=SYSIN                                             00001900
19      XXGO     EXEC  PGM=*.LKED.SYSLMOD,COND=((9,LT,LKED),(9,LT,PL1L))       00002000
20      //GO.STEPLIB DD DSN=SYSC.PL1LIB,DISP=SHR
        X/STEPLIB  DD  DSN=SYSC.LINKLIB,DISP=SHR                                00002102
21      XX         DD  DSN=SYSC.PL1LIB,DISP=SHR                                 00002202
22      XXSYSPRINT DD  SYSOUT=A                                                 00002300
23      //GO.PRINTR DD SYSOUT=*
24      //GO.IMAGES DD DSN=PUB001.VSAMTEST.DATA,DISP=SHR
25      //GO.SYSUDUMP DD SYSOUT=*
26      //GO.SYSPRINT DD SYSOUT=*
27      //GO.RRDSF01 DD DSN=PUB001.VSTESTRR.CLUSTER,DISP=OLD
```

```
 STMT NO. MESSAGE
-
   19     IEF686I DDNAME REFERRED TO ON DDNAME KEYWORD IN PRIOR STEP WAS NOT RESOLVED
IEF236I ALLOC. FOR VSTESTR4 PL1L PL1F
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO SYS00382
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 380  ALLOCATED TO SYSLIN
IEF237I 251  ALLOCATED TO SYSUT3
IEF237I 370  ALLOCATED TO SYSUT1
IEF237I 253  ALLOCATED TO SYSIN
IEF237I 253  ALLOCATED TO SYSLIB
IEF142I VSTESTR4 PL1L PL1F - STEP WAS EXECUTED - COND CODE 0004
IEF285I   SYSC.LINKLIB                                KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                    KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   JES2.JOB00159.SO0101                        SYSOUT
IEF285I   SYS20189.T182246.RA000.VSTESTR4.LOADSET      PASSED        *------217
IEF285I   VOL SER NOS= MVS380.
IEF285I   SYS20189.T182246.RA000.VSTESTR4.SYSUT3       DELETED       *------269
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T182246.RA000.VSTESTR4.SYSUT1       DELETED       *--------0
IEF285I   VOL SER NOS= MVS370.
IEF285I   SYSC.VSAMIOP.SOURCE                         KEPT          *--------3
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.VSAMIOP.MACLIB                         KEPT          *-------27
IEF285I   VOL SER NOS= SYSCPK.
IEF373I STEP /PL1L    / START 20189.1822
IEF374I STEP /PL1L    / STOP  20189.1822 CPU    0MIN 00.16SEC SRB    0MIN 00.04SEC VIRT  4096K SYS   212K
**** JOBCARD READ 20189 18:22:46 ************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                              *
*  STEP NAME  PL1L        USER CORE        4096K   TAPES USED/IO 000/000000000   START   TIME  18:22:46    TCB TIME  00:00:00.16  *
*  PGM  NAME  IEMAA       SYSTEM CORE       212K   DISKS USED/IO 004/000000516   STOP    TIME  18:22:47    SRB TIME  00:00:00.04  *
*   COND CODE    0004     PRIVATE AREA SZ   4096K       ALLOC TIME  18:22:46      ELAPSED TIME                PGM LOAD  18:22:46   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004          2791   00:00:00.23            0             0           0             0              0             0            0  *
************************************************************************************************************************
*  CPU $ (     0.05) + EXCP $ (     0.69) + MEMORY $ (     1.86) = TOTAL $ (     2.60)                                  *
************************************************************************************************************************
IEF236I ALLOC. FOR VSTESTR4 LKED PL1F
IEF237I 253  ALLOCATED TO SYSLIB
IEF237I 253  ALLOCATED TO
IEF237I 253  ALLOCATED TO SYS00384
IEF237I 251  ALLOCATED TO SYSLMOD
IEF237I 370  ALLOCATED TO SYSUT1
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 380  ALLOCATED TO SYSLIN
IEF237I DMY  ALLOCATED TO
IEF142I VSTESTR4 LKED PL1F - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYSC.PL1LIB                                 KEPT          *------129
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.LINKLIB                                KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                    KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYS20189.T182246.RA000.VSTESTR4.GOSET        PASSED        *-------58
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T182246.RA000.VSTESTR4.SYSUT1       DELETED       *--------0
IEF285I   VOL SER NOS= MVS370.
IEF285I   JES2.JOB00159.SO0102                        SYSOUT
```

```
IEF285I   SYS20189.T182246.RA000.VSTESTR4.LOADSET      DELETED       *------218
IEF285I   VOL SER NOS= MVS380.
IEF373I STEP /LKED    / START 20189.1822
IEF374I STEP /LKED    / STOP  20189.1822 CPU    0MIN 00.05SEC SRB    0MIN 00.01SEC VIRT   260K SYS   168K
********************************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                 *
*  STEP NAME  LKED        USER CORE          260K   TAPES USED/IO 000/000000000   START   TIME  18:22:47    TCB TIME  00:00:00.05  *
*  PGM  NAME  IEWL        SYSTEM CORE        168K   DISKS USED/IO 004/000000405   STOP    TIME  18:22:47    SRB TIME  00:00:00.01  *
*   COND CODE    0000     PRIVATE AREA SZ   4096K       ALLOC TIME  18:22:47      ELAPSED TIME                PGM LOAD  18:22:47   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004          2072   00:00:00.21            0             0         0             0              0             0             0  *
********************************************************************************************************
*  CPU $ (     0.01) + EXCP $ (     0.54) + MEMORY $ (     0.03) = TOTAL $ (     0.58)                    *
********************************************************************************************************
IEF236I ALLOC. FOR VSTESTR4 GO PL1F
IEF237I 251  ALLOCATED TO PGM=*.DD
IEF237I 253  ALLOCATED TO STEPLIB
IEF237I 253  ALLOCATED TO
IEF237I 253  ALLOCATED TO SYS00386
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I JES2 ALLOCATED TO PRINTR
IEF237I 190  ALLOCATED TO IMAGES
IEF237I 190  ALLOCATED TO SYS00388
IEF237I JES2 ALLOCATED TO SYSUDUMP
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I 190  ALLOCATED TO RRDSF01
IEF142I VSTESTR4 GO PL1F - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYS20189.T182246.RA000.VSTESTR4.GOSET        KEPT          *--------0
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYSC.PL1LIB                                  KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   SYSC.PL1LIB                                  KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   UCSYSCPK                                     KEPT          *--------0
IEF285I   VOL SER NOS= SYSCPK.
IEF285I   JES2.JOB00159.SO0103                         SYSOUT
IEF285I   JES2.JOB00159.SO0104                         SYSOUT
IEF285I   PUB001.VSAMTEST.DATA                         KEPT          *-------11
IEF285I   VOL SER NOS= PUB001.
IEF285I   UCPUB001                                     KEPT          *--------0
IEF285I   VOL SER NOS= PUB001.
IEF285I   JES2.JOB00159.SO0105                         SYSOUT
IEF285I   JES2.JOB00159.SO0106                         SYSOUT
IEF285I   PUB001.VSTESTRR.CLUSTER                      KEPT          *------204
IEF285I   VOL SER NOS= PUB001.
IEF373I STEP /GO      / START 20189.1822
IEF374I STEP /GO      / STOP  20189.1822 CPU    0MIN 00.03SEC SRB    0MIN 00.01SEC VIRT   100K SYS   180K
********************************************************************************************************
*                                      PRC-CCI  370/148 VS2 R03.8  HMVS  STEP STATISTICS                 *
*  STEP NAME  GO          USER CORE          100K   TAPES USED/IO 000/000000000   START   TIME  18:22:47    TCB TIME  00:00:00.03  *
*  PGM  NAME  PGM=*.DD    SYSTEM CORE        180K   DISKS USED/IO 003/000000215   STOP    TIME  18:22:47    SRB TIME  00:00:00.01  *
*   COND CODE    0000     PRIVATE AREA SZ   4096K       ALLOC TIME  18:22:47      ELAPSED TIME                PGM LOAD  18:22:47   *
** PGNO * NR SRV UNITS * ACTIVE TIME ** PAGES IN *** PAGES OUT ** # SWAPS * PGS SWAP IN * PGS SWAP OUT * VIO PGS IN * VIO PGS OUT **
*   004          1110   00:00:00.04            2             0         0             0              0             0             0  *
********************************************************************************************************
*  CPU $ (     0.01) + EXCP $ (     0.29) + MEMORY $ (     0.00) = TOTAL $ (     0.30)                    *
********************************************************************************************************
IEF237I 251  ALLOCATED TO SYS00001
IEF285I   SYS20189.T182247.RA000.VSTESTR4.R0000001     KEPT          *--------0
IEF285I   VOL SER NOS= WORK00.
IEF285I   SYS20189.T182246.RA000.VSTESTR4.GOSET        DELETED
```

```
IEF285I   VOL SER NOS= WORK00.
IEF375I  JOB /VSTESTR4/ START 20189.1822
IEF376I  JOB /VSTESTR4/ STOP  20189.1822 CPU    0MIN 00.24SEC SRB    0MIN 00.06SEC
```

PL/I F COMPILER OPTIONS SPECIFIED ARE AS FOLLOWS--

```
     LOAD,NODECK,ATR,XREF,CHAR60,MACRO


THE COMPLETE LIST OF OPTIONS USED DURING THIS COMPILATION IS--   EBCDIC
                                                                CHAR60
                                                                MACRO
                                                                SOURCE2
                                                              NOMACDCK
                                                                COMP
                                                                SOURCE
                                                                ATR
                                                                XREF
                                                              NOEXTREF
                                                              NOLIST
                                                                LOAD
                                                              NODECK
                                                                FLAGW
                                                              NOSTMT
                                                                SIZE=4154608
                                                                LINECNT=050
                                                                OPT=01
                                                                SORMGIN=(002,072)
                                                              NOEXTDIC
                                                              NONEST
                                                              OPLIST
                                                              SYNCHKT

*OPTIONS IN EFFECT*     EBCDIC,CHAR60,MACRO,SOURCE2,NOMACDCK,COMP,SOURCE,ATR,XREF,NOEXTREF,NOLIST,LOAD,
*OPTIONS IN EFFECT*     NODECK,FLAGW,NOSTMT,SIZE=4154608,LINECNT=050,OPT=01,SORMGIN=(002,072),NOEXTDIC,
*OPTIONS IN EFFECT*     NONEST,OPLIST,SYNCHKT
```

COMPILE-TIME MACRO PROCESSOR
MACRO SOURCE2 LISTING

```
  1    /*****************************************************************17090000
  2                                                                      17100000
  3      RRDSLODR - TESTS THE VSAMIO ROUTINE BY LOADING AN RRDS CLUSTER  17110000
  4                 DIRECTLY WITH RECORDS FROM A SEQUENTIAL DATASET.  THE17120000
  5                 RELATIVE RECORD NUMBER IS TAKEN FROM THE SEQUENCE PART17130000
  6                 OF THE INPUT RECORD AND THERE ARE GAPS LEFT IN THE   17140000
  7                 RESULTING CLUSTER.                                   17150000
  8                                                                      17160000
  9     *****************************************************************/17170000
 10    RRDSLOR:                                                          17180000
 11       PROCEDURE OPTIONS(MAIN);                                       17190000
 12                                                                      17200000
 13          ON ERROR                                                    17210000
 14             BEGIN;                                                   17220000
 15                ON ERROR SYSTEM;                                      17230000
 16                PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');17240000
 17                PUT SKIP DATA;                                        17250000
 18                PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');17260000
 19             END;                                                     17270000
 20                                                                      17280000
 21          OPEN                                                        17290000
 22             FILE(IMAGES),                                            17300000
 23             FILE(PRINTR) LINESIZE(121);                              17310000
 24                                                                      17320000
 25          ON ENDFILE(IMAGES)                                          17330000
 26             MORE_RECORDS = NO;                                       17340000
 27                                                                      17350000
 28          PRINT_AREA = 'RRDSLODR: WRITE RRDS DIRECT (VACANT SLOTS)';  17360000
 29          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        17370000
 30          PRINT_AREA = '--------------------------------------';      17380000
 31          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        17390000
 32          PRINT_AREA = ' ';                                           17400000
 33          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        17410000
 34                                                                      17420000
 35          MORE_RECORDS = YES;                                         17430000
 36                                                                      17440000
 37    /*****************************************************************17450000
 38      ESTABLISH PARAMETERS FOR VSAM DATASET AND CALL ROUTINE TO OPEN   17460000
 39     *****************************************************************/17470000
 40          VSFB_DDNAME = 'RRDSF01';                                    17480000
 41          VSFB_ORGANIZATION = VSIO_RRDS;                              17490000
 42          VSFB_ACCESS = VSIO_DIRECT;                                  17500000
 43          VSFB_MODE = VSIO_OUTPUT;                                    17510000
 44          VSFB_RECORD_LENGTH = 80;                                    17520000
```

MACRO SOURCE2 LISTING

```
 45          VSFB_KEY_POSITION = 0;                                  17530000
 46          VSFB_KEY_LENGTH = 0;                                    17540000
 47          VSIO_COMMAND = VSIO_OPEN;                               17550000
 48          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                     17560000
 49                        VSIO_FILE_BLOCK,                          17570000
 50                        RECORD_IMAGE);                            17580000
 51          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN           17590000
 52             DO;                                                  17600000
 53                CALL VSIO_ERROR;                                  17610000
 54                RETURN;                                           17620000
 55             END;                                                 17630000
 56                                                                  17640000
 57          DO WHILE(MORE_RECORDS);                                 17650000
 58             READ FILE(IMAGES) INTO(RECORD_IMAGE);                17660000
 59             IF (MORE_RECORDS) THEN                               17670000
 60                DO;                                               17680000
 61                   VSFB_KEY_LENGTH = RECORD_SEQUENCE;             17690000
 62                   CALL WRITE_RR;                                 17700000
 63                END;                                              17710000
 64          END;                                                    17720000
 65                                                                  17730000
 66          CLOSE FILE(IMAGES);                                     17740000
 67                                                                  17750000
 68     /*************************************************************17760000
 69       CALL ROUTINE TO CLOSE VSAM DATASET                         17770000
 70     *************************************************************/17780000
 71          VSIO_COMMAND = VSIO_CLOSE;                              17790000
 72          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                     17800000
 73                        VSIO_FILE_BLOCK,                          17810000
 74                        RECORD_IMAGE);                            17820000
 75          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN           17830000
 76             CALL VSIO_ERROR;                                     17840000
 77                                                                  17850000
 78          COUNTER_EDIT = RECORD_COUNTER;                          17860000
 79          PRINT_AREA = COUNTER_EDIT ||                            17870000
 80                       ' RECORDS WERE LOADED SUCCESSFULLY';       17880000
 81          WRITE FILE(PRINTR) FROM(PRINT_LINE);                    17890000
 82                                                                  17900000
 83          RETURN;                                                 17910000
 84                                                                  17920000
 85     WRITE_RR:                                                    17930000
 86        PROCEDURE;                                                17940000
 87                                                                  17950000
 88     /*************************************************************17960000
 89       CALL ROUTINE TO WRITE RECORD INTO VSAM DATASET             17970000
```

```
 90     *****************************************************************/17980000
 91          VSIO_COMMAND = VSIO_WRITE;                                  17990000
 92          CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                         18000000
 93                        VSIO_FILE_BLOCK,                              18010000
 94                        RECORD_IMAGE);                                18020000
 95          IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN               18030000
 96             CALL VSIO_ERROR;                                         18040000
 97          ELSE                                                        18050000
 98             RECORD_COUNTER = RECORD_COUNTER + 1;                     18060000
 99                                                                      18070000
100          RETURN;                                                     18080000
101                                                                      18090000
102       END WRITE_RR;                                                  18100000
103                                                                      18110000
104    VSIO_ERROR:                                                       18120000
105       PROCEDURE;                                                     18130000
106          PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||             18140000
107                       VSIO_COMMAND;                                  18150000
108          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        18160000
109          PRINT_AREA = 'VSIO_RETURN_CODE = ' ||                       18170000
110                       VSIO_RETURN_CODE;                              18180000
111          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        18190000
112          PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||                  18200000
113                       VSIO_VSAM_RETURN_CODE;                         18210000
114          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        18220000
115          PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||                18230000
116                       VSIO_VSAM_FUNCTION_CODE;                       18240000
117          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        18250000
118          PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||                18260000
119                       VSIO_VSAM_FEEDBACK_CODE;                       18270000
120          WRITE FILE(PRINTR) FROM(PRINT_LINE);                        18280000
121          PRINT_AREA = ' ';                                           18290000
122                                                                      18300000
123          RETURN;                                                     18310000
124                                                                      18320000
125       END VSIO_ERROR;                                                18330000
126                                                                      18340000
127       DECLARE                                                        18350000
128          IMAGES FILE INPUT RECORD SEQUENTIAL EXTERNAL                18360000
129             ENV(F),                                                  18370000
130          PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL               18380000
131             ENV(F CTLASA);                                           18390000
132                                                                      18400000
133       DECLARE                                                        18410000
134          COUNTER_EDIT                    PICTURE 'ZZ,ZZZ,ZZ9V',      18420000
```

MACRO SOURCE2 LISTING

```
135          MORE_RECORDS                  BIT(1),                          18430000
136          NO                            BIT(1)    INIT('0'B),            18440000
137          RECORD_COUNTER                FIXED BINARY(15,0) INIT(0),      18450000
138          YES                           BIT(1)    INIT('1'B);            18460000
139                                                                          18470000
140       DECLARE                                                            18480000
141          1 RECORD_IMAGE,                                                 18490000
142             2 RECORD_IDENT             CHAR(7),                         18500000
143             2 RECORD_SEQUENCE          CHAR(3),                         18510000
144             2 RECORD_FIELDS            CHAR(70);                        18520000
145                                                                          18530000
146       DECLARE                                                            18540000
147          1 PRINT_LINE,                                                   18550000
148             2 CARRIAGE_CONTROL         CHAR(1)   INIT(' '),             18560000
149             2 PRINT_AREA               CHAR(120);                       18570000
150                                                                          18580000
151    %INCLUDE (VSAMIO);                                                    18590000
152    %INCLUDE (VSAMIOFB);                                                  18600000
153                                                                          18610000
154       END RRDSLOR;                                                       18620000
```

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIO

```
155    /*****************************************************************31100000
156                                                                         31110000
157           VV   VV    SSSSS       A      M     M    IIII      OOOOO       31120000
158           VV   VV   SS   SS     AAA     MM   MM     II      OO   OO      31130000
159           VV   VV   SS         AA AA    MMM MMM     II      OO   OO      31140000
160           VV   VV    SSSSS    AA   AA   MMMMMMM     II      OO   OO      31150000
161           VV   VV        SS   AA   AA   MM M MM     II      OO   OO      31160000
162            VV VV    SS   SS   AAAAAAA   MM   MM     II      OO   OO      31170000
163             VVV     SS   SS   AA   AA   MM   MM     II      OO   OO      31180000
164              V       SSSSS    AA   AA   MM   MM    IIII      OOOOO       31190000
165                                                                         31200000
166     *****************************************************************31210000
167      THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS  31220000
168      ROUTINE.                                                           31230000
169                                                                         31240000
170      THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO      31250000
171      PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND  31260000
172      TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.                          31270000
173     *****************************************************************/31280000
174                                                                         31290000
175           DECLARE                                                       31300000
```

MACRO SOURCE2 LISTING

```
176                1 VSIO_PARAMETER_VALUES   STATIC,                          31310000
177                   2 VSIO_OPEN            CHAR(8)   INIT('OPEN    '),     31320000
178                   2 VSIO_CLOSE           CHAR(8)   INIT('CLOSE   '),     31330000
179                   2 VSIO_READ            CHAR(8)   INIT('READ    '),     31340000
180                   2 VSIO_WRITE           CHAR(8)   INIT('WRITE   '),     31350000
181                   2 VSIO_REWRITE         CHAR(8)   INIT('REWRITE '),     31360000
182                   2 VSIO_DELETE          CHAR(8)   INIT('DELETE  '),     31370000
183                   2 VSIO_START_EQUAL     CHAR(8)   INIT('STARTEQ '),     31380000
184                   2 VSIO_START_NOTLESS   CHAR(8)   INIT('STARTGE '),     31390000
185                   2 VSIO_KSDS            CHAR(4)   INIT('KSDS'),         31400000
186                   2 VSIO_ESDS            CHAR(4)   INIT('ESDS'),         31410000
187                   2 VSIO_RRDS            CHAR(4)   INIT('RRDS'),         31420000
188                   2 VSIO_SEQUENTIAL      CHAR(10)  INIT('SEQUENTIAL'),   31430000
189                   2 VSIO_DIRECT          CHAR(10)  INIT('DIRECT    '),   31440000
190                   2 VSIO_DYNAMIC         CHAR(10)  INIT('DYNAMIC   '),   31450000
191                   2 VSIO_INPUT           CHAR(6)   INIT('INPUT '),       31460000
192                   2 VSIO_OUTPUT          CHAR(6)   INIT('OUTPUT'),       31470000
193                   2 VSIO_INPUT_OUTPUT    CHAR(6)   INIT('UPDATE'),       31480000
194                   2 (VSIO_RC_SUCCESS               INIT(0),              31490000
195                      VSIO_RC_LOGIC_ERROR           INIT(8),              31500000
196                      VSIO_RC_END_OF_FILE           INIT(9999),           31510000
197                      VSIO_RC_UNKNOWN_COMMAND       INIT(20),             31520000
198                      VSIO_RC_DATASET_ALREADY_OPEN  INIT(21),             31530000
199                      VSIO_RC_DATASET_NOT_OPEN      INIT(22),             31540000
200                      VSIO_RC_ORGANIZATION_UNKNOWN  INIT(23),             31550000
201                      VSIO_RC_ACCESS_UNKNOWN        INIT(24),             31560000
202                      VSIO_RC_ORG_ACCESS_MISMATCH   INIT(25),             31570000
203                      VSIO_RC_MODE_UNKNOWN          INIT(26),             31580000
204                      VSIO_RC_MODE_UNSUPPORTED      INIT(27),             31590000
205                      VSIO_RC_DDNAME_BLANK          INIT(28))             31600000
206                                        FIXED BINARY(15,0),               31610000
207                   2 (VSIO_FB_DUPLICATE_RECORD      INIT(8),              31620000
208                      VSIO_FB_KEY_SEQUENCE          INIT(12),             31630000
209                      VSIO_FB_RECORD_NOT_FOUND      INIT(16),             31640000
210                      VSIO_FB_NO_MORE_SPACE         INIT(28),             31650000
211                      VSIO_FB_READ_WITHOUT_START    INIT(88))             31660000
212                                        FIXED BINARY(15,0),               31670000
213       /*****************************************************************31680000
214         THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE  31690000
215         THE ROUTINE.                                                    31700000
216        *****************************************************************/31710000
217                                                                          31720000
218                1 VSIO_PARAMETER_BLOCK    STATIC,                         31730000
219                   2 VSIO_COMMAND         CHAR(8)   INIT(' '),            31740000
220                   2 (VSIO_RETURN_CODE,                                   31750000
```

MACRO SOURCE2 LISTING

```
221                     VSIO_VSAM_RC,                                    31760000
222                     VSIO_VSAM_FUNCTION,                              31770000
223                     VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0);  31780000
224                                                                      31790000
225     /*****************************************************************31800000
226                              END OF VSAMIO COPY BOOK                 31810000
227      *****************************************************************/31820000
```

INCLUDED TEXT FOLLOWS FROM DD.MEMBER = SYSLIB .VSAMIOFB

```
228     /*****************************************************************00000100
229                                                                      00000200
230       VV   VV   SSSSS      A     M     M  IIII   OOOOO   FFFFFFF  BBBBBB  00000300
231       VV   VV  SS   SS    AAA    MM   MM   II   OO   OO  FF       BB   BB 00000400
232       VV   VV  SS        AA AA   MMM MMM   II   OO   OO  FF       BB   BB 00000500
233       VV   VV   SSSSS   AA   AA  MMMMMMM   II   OO   OO  FFFFF    BBBBBB  00000600
234       VV   VV       SS  AA   AA  MM M MM   II   OO   OO  FF       BB   BB 00000700
235        VV VV   SS   SS  AAAAAAA  MM   MM   II   OO   OO  FF       BB   BB 00000800
236         VVV    SS   SS  AA   AA  MM   MM   II   OO   OO  FF       BB   BB 00000900
237          V      SSSSS   AA   AA  MM   MM  IIII   OOOOO   FF       BBBBBB  00001000
238                                                                      00001100
239      *****************************************************************00001200
240       THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS 00001300
241       ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE   00001400
242       VSAM DATASET.                                                      00001500
243                                                                          00001600
244       WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE    00001700
245       LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD  00001800
246       DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE  00001900
247       ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE  00002000
248       DATASET HAS BEEN CLOSED.                                           00002100
249      *****************************************************************/00002200
250                                                                          00002300
251          DECLARE                                                        00002400
252             1 VSIO_FILE_BLOCK            STATIC,                       00002500
253                2 VSFB_DDNAME             CHAR(8)   INIT(' '),          00002600
254                2 VSFB_ORGANIZATION       CHAR(4)   INIT(' '),          00002700
255                2 VSFB_ACCESS             CHAR(10)  INIT(' '),          00002800
256                2 VSFB_MODE               CHAR(6)   INIT(' '),          00002900
257                2 (VSFB_RECORD_LENGTH,                                  00003000
258                   VSFB_KEY_POSITION,                                   00003100
259                   VSFB_KEY_LENGTH)       FIXED BINARY(15,0) INIT(0),   00003200
260                2 VSFB_FILE_STATUS        CHAR(1)   INIT('C'),          00003300
261                2 VSFB_RESERVED           CHAR(161);                    00003400
```

MACRO SOURCE2 LISTING

```
262                                                                   00003500
263    /*****************************************************************00003600
264                          END OF VSAMIOFB COPY BOOK                 00003700
265     ****************************************************************/00003800
```

NO ERROR OR WARNING CONDITION HAS BEEN DETECTED FOR THIS MACRO PASS.

SOURCE LISTING.

```
                /*****************************************************************    1
                                                                                       1
                  RRDSLODR - TESTS THE VSAMIO ROUTINE BY LOADING AN RRDS CLUSTER        1
                             DIRECTLY WITH RECORDS FROM A SEQUENTIAL DATASET.  THE      1
                             RELATIVE RECORD NUMBER IS TAKEN FROM THE SEQUENCE PART     1
                             OF THE INPUT RECORD AND THERE ARE GAPS LEFT IN THE         1
                             RESULTING CLUSTER.                                        1
                                                                                       1
                 *****************************************************************/    1
                                                                                       9
1                RRDSLOR:                                                             10
                   PROCEDURE OPTIONS(MAIN);                                          11
                                                                                     12
2                     ON ERROR                                                       13
3                     BEGIN;                                                         14
4                        ON ERROR SYSTEM;                                            15
5                        PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');      16
6                        PUT SKIP DATA;                                              17
7                        PUT SKIP(3) LIST((54)'*' || ' DEBUG AID ' || (54)'*');      18
8                     END;                                                           19
                                                                                     20
9                     OPEN                                                           21
                         FILE(IMAGES),                                               22
                         FILE(PRINTR) LINESIZE(121);                                 23
                                                                                     24
10                    ON ENDFILE(IMAGES)                                             25
11                       MORE_RECORDS = NO;                                          26
                                                                                     27
12                    PRINT_AREA = 'RRDSLODR: WRITE RRDS DIRECT (VACANT SLOTS)';     28
13                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                           29
14                    PRINT_AREA = '---------------------------------------';        30
15                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                           31
16                    PRINT_AREA = ' ';                                              32
17                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                           33
                                                                                     34
18                    MORE_RECORDS = YES;                                            35
                                                                                     36
                   /*****************************************************************  37
                     ESTABLISH PARAMETERS FOR VSAM DATASET AND CALL ROUTINE TO OPEN     37
                    *****************************************************************/  37
                                                                                     39
19                    VSFB_DDNAME = 'RRDSF01';                                       40
20                    VSFB_ORGANIZATION = VSIO_RRDS;                                 41
```

```
                    VSFB_ACCESS = VSIO_DIRECT;                                  
                    VSFB_MODE = VSIO_OUTPUT;                                    
                    VSFB_RECORD_LENGTH = 80;                                    
                    VSFB_KEY_POSITION = 0;                                      
                    VSFB_KEY_LENGTH = 0;                                        
                    VSIO_COMMAND = VSIO_OPEN;                                   
                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                         
                                  VSIO_FILE_BLOCK,                              
                                  RECORD_IMAGE);                                
                    IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN               
                       DO;                                                      
                          CALL VSIO_ERROR;                                      
                          RETURN;                                               
                       END;                                                     

                    DO WHILE(MORE_RECORDS);                                     
                       READ FILE(IMAGES) INTO(RECORD_IMAGE);                    
                       IF (MORE_RECORDS) THEN                                   
                          DO;                                                   
                             VSFB_KEY_LENGTH = RECORD_SEQUENCE;                 
                             CALL WRITE_RR;                                     
                          END;                                                  
                    END;                                                        

                    CLOSE FILE(IMAGES);                                         

                /*****************************************************************
                 CALL ROUTINE TO CLOSE VSAM DATASET                             
                *****************************************************************/
                                                                                
                    VSIO_COMMAND = VSIO_CLOSE;                                  
                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                         
                                  VSIO_FILE_BLOCK,                              
                                  RECORD_IMAGE);                                
                    IF (VSIO_RETURN_CODE ^= VSIO_RC_SUCCESS) THEN               
                       CALL VSIO_ERROR;                                         

                    COUNTER_EDIT = RECORD_COUNTER;                              
                    PRINT_AREA = COUNTER_EDIT ||                                
                                 ' RECORDS WERE LOADED SUCCESSFULLY';           
                    WRITE FILE(PRINTR) FROM(PRINT_LINE);                        

                    RETURN;                                                     

                WRITE_RR:                                                       
                   PROCEDURE;                                                   
```

```
                /*****************************************************************   88
                  CALL ROUTINE TO WRITE RECORD INTO VSAM DATASET                      88
                 *****************************************************************/   88
                                                                                      90
51                    VSIO_COMMAND = VSIO_WRITE;                                      91
52                    CALL VSAMIOP (VSIO_PARAMETER_BLOCK,                             92
                                    VSIO_FILE_BLOCK,                                  93
                                    RECORD_IMAGE);                                    94
53                    IF (VSIO_RETURN_CODE ª= VSIO_RC_SUCCESS) THEN                   95
54                       CALL VSIO_ERROR;                                             96
55                    ELSE                                                            97
55                       RECORD_COUNTER = RECORD_COUNTER + 1;                         98
                                                                                      99
56                    RETURN;                                                        100
                                                                                     101
57                  END WRITE_RR;                                                    102
                                                                                     103
58                VSIO_ERROR:                                                        104
                    PROCEDURE;                                                     105
59                     PRINT_AREA = 'VSAMIO ERROR OCCURRED DURING ' ||             106
                                    VSIO_COMMAND;                                  107
60                     WRITE FILE(PRINTR) FROM(PRINT_LINE);                        108
61                     PRINT_AREA = 'VSIO_RETURN_CODE = ' ||                       109
                                    VSIO_RETURN_CODE;                              110
62                     WRITE FILE(PRINTR) FROM(PRINT_LINE);                        111
63                     PRINT_AREA = 'VSIO_VSAM_RETURN_CODE = ' ||                  112
                                    VSIO_VSAM_RETURN_CODE;                         113
64                     WRITE FILE(PRINTR) FROM(PRINT_LINE);                        114
65                     PRINT_AREA = 'VSIO_VSAM_FUNCTION_CODE = ' ||                115
                                    VSIO_VSAM_FUNCTION_CODE;                       116
66                     WRITE FILE(PRINTR) FROM(PRINT_LINE);                        117
67                     PRINT_AREA = 'VSIO_VSAM_FEEDBACK_CODE = ' ||                118
                                    VSIO_VSAM_FEEDBACK_CODE;                       119
68                     WRITE FILE(PRINTR) FROM(PRINT_LINE);                        120
69                     PRINT_AREA = ' ';                                           121
                                                                                   122
70                     RETURN;                                                     123
                                                                                   124
71                  END VSIO_ERROR;                                               125
                                                                                   126
72                  DECLARE                                                       127
                       IMAGES FILE INPUT RECORD SEQUENTIAL EXTERNAL               128
                          ENV(F),                                                 129
                       PRINTR FILE OUTPUT RECORD SEQUENTIAL EXTERNAL              130
                          ENV(F CTLASA);                                          131
                                                                                  132
73                  DECLARE                                                       133
```

```
                  COUNTER_EDIT                    PICTURE 'ZZ,ZZZ,ZZ9V',          134
                  MORE_RECORDS                    BIT(1),                          135
                  NO                              BIT(1)    INIT('0'B),            136
                  RECORD_COUNTER                  FIXED BINARY(15,0) INIT(0),      137
                  YES                             BIT(1)    INIT('1'B);            138
                                                                                   139
74                DECLARE                                                          140
                  1 RECORD_IMAGE,                                                  141
                     2 RECORD_IDENT               CHAR(7),                         142
                     2 RECORD_SEQUENCE            CHAR(3),                         143
                     2 RECORD_FIELDS              CHAR(70);                        144
                                                                                   145
75                DECLARE                                                          146
                  1 PRINT_LINE,                                                    147
                     2 CARRIAGE_CONTROL           CHAR(1)   INIT(' '),             148
                     2 PRINT_AREA                 CHAR(120);                       149
                                                                                   150
               /*******************************************************************  155
                                                                                   155
                    VV   VV    SSSSS       A      M     M    IIII      00000      155
                    VV   VV   SS   SS     AAA     MM   MM     II      00   00     155
                    VV   VV   SS         AA AA    MMM MMM     II      00   00     155
                    VV   VV    SSSSS    AA   AA   MMMMMMM     II      00   00     155
                    VV   VV        SS   AA   AA   MM M MM     II      00   00     155
                     VV VV    SS   SS   AAAAAAA   MM   MM     II      00   00     155
                      VVV     SS   SS   AA   AA   MM   MM     II      00   00     155
                       V       SSSSS    AA   AA   MM   MM    IIII      00000      155
                                                                                   155
                  *******************************************************************  155
                   THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS   155
                   ROUTINE.                                                        155
                                                                                   155
                   THE VSIO_PARAMETER_VALUES SUPPLY THE VALUES USED TO MOVE INTO   155
                   PARAMETER ENTRIES TO TAILOR THE ROUTINE TO A SPECIFIC DATASET AND   155
                   TO PROVIDE COMMANDS TO DRIVE THE ROUTINE.                       155
                  *******************************************************************/  155
                                                                                   173
76                      DECLARE                                                    175
                        1 VSIO_PARAMETER_VALUES   STATIC,                          176
                           2 VSIO_OPEN            CHAR(8)   INIT('OPEN    '),     177
                           2 VSIO_CLOSE           CHAR(8)   INIT('CLOSE   '),     178
                           2 VSIO_READ            CHAR(8)   INIT('READ    '),     179
                           2 VSIO_WRITE           CHAR(8)   INIT('WRITE   '),     180
                           2 VSIO_REWRITE         CHAR(8)   INIT('REWRITE '),     181
                           2 VSIO_DELETE          CHAR(8)   INIT('DELETE  '),     182
                           2 VSIO_START_EQUAL     CHAR(8)   INIT('STARTEQ '),     183
                           2 VSIO_START_NOTLESS   CHAR(8)   INIT('STARTGE '),     184
```

```
                    2 VSIO_KSDS            CHAR(4)   INIT('KSDS'),           185
                    2 VSIO_ESDS            CHAR(4)   INIT('ESDS'),           186
                    2 VSIO_RRDS            CHAR(4)   INIT('RRDS'),           187
                    2 VSIO_SEQUENTIAL      CHAR(10)  INIT('SEQUENTIAL'),     188
                    2 VSIO_DIRECT          CHAR(10)  INIT('DIRECT    '),     189
                    2 VSIO_DYNAMIC         CHAR(10)  INIT('DYNAMIC   '),     190
                    2 VSIO_INPUT           CHAR(6)   INIT('INPUT '),         191
                    2 VSIO_OUTPUT          CHAR(6)   INIT('OUTPUT'),         192
                    2 VSIO_INPUT_OUTPUT    CHAR(6)   INIT('UPDATE'),         193
                    2 (VSIO_RC_SUCCESS                INIT(0),              194
                       VSIO_RC_LOGIC_ERROR            INIT(8),              195
                       VSIO_RC_END_OF_FILE            INIT(9999),           196
                       VSIO_RC_UNKNOWN_COMMAND        INIT(20),             197
                       VSIO_RC_DATASET_ALREADY_OPEN   INIT(21),             198
                       VSIO_RC_DATASET_NOT_OPEN       INIT(22),             199
                       VSIO_RC_ORGANIZATION_UNKNOWN   INIT(23),             200
                       VSIO_RC_ACCESS_UNKNOWN         INIT(24),             201
                       VSIO_RC_ORG_ACCESS_MISMATCH    INIT(25),             202
                       VSIO_RC_MODE_UNKNOWN           INIT(26),             203
                       VSIO_RC_MODE_UNSUPPORTED       INIT(27),             204
                       VSIO_RC_DDNAME_BLANK           INIT(28))             205
                                          FIXED BINARY(15,0),                206
                    2 (VSIO_FB_DUPLICATE_RECORD       INIT(8),              207
                       VSIO_FB_KEY_SEQUENCE           INIT(12),             208
                       VSIO_FB_RECORD_NOT_FOUND       INIT(16),             209
                       VSIO_FB_NO_MORE_SPACE          INIT(28),             210
                       VSIO_FB_READ_WITHOUT_START     INIT(88))             211
                                          FIXED BINARY(15,0),                212
                  /*****************************************************     213
                    THE VSIO_PARAMETER_BLOCK IS THE COMMUNICATION INTERFACE TO THE          213
                    THE ROUTINE.                                            213
                   *****************************************************/   213
                                                                            216
                       1 VSIO_PARAMETER_BLOCK    STATIC,                    218
                          2 VSIO_COMMAND         CHAR(8)   INIT(' '),       219
                          2 (VSIO_RETURN_CODE,                              220
                             VSIO_VSAM_RC,                                  221
                             VSIO_VSAM_FUNCTION,                            222
                             VSIO_VSAM_FEEDBACK) FIXED BINARY(15,0) INIT(0); 223
                                                                            224
                  /*****************************************************     225
                                   END OF VSAMIO COPY BOOK                  225
                   *****************************************************/   225
                  /*****************************************************     228
                                                                            228
                     VV   VV   SSSSS      A     M     M  IIII   OOOOO   FFFFFFF  BBBBBB    228
                     VV   VV  SS   SS    AAA    MM   MM   II   OO   OO  FF       BB   BB   228
```

```
                    VV   VV  SS        AA AA   MMM MMM   II   OO   OO  FF       BB   BB   228
                    VV   VV   SSSSS   AA   AA  MMMMMMM   II   OO   OO  FFFFF    BBBBBB    228
                    VV   VV       SS  AA   AA  MM M MM   II   OO   OO  FF       BB   BB   228
                     VV VV   SS   SS  AAAAAAA  MM   MM   II   OO   OO  FF       BB   BB   228
                      VVV    SS   SS  AA   AA  MM   MM   II   OO   OO  FF       BB   BB   228
                       V      SSSSS   AA   AA  MM   MM  IIII   OOOOO   FF       BBBBBB    228
                                                                                           228
                  ************************************************************************ 228
                    THESE PARAMETERS ARE USED TO INTERFACE WITH THE VSAM DATASET ACCESS    228
                    ROUTINE, AND ARE USED TO COMMUNICATE CHARACTERISTICS FOR A SINGLE      228
                    VSAM DATASET.                                                          228
                                                                                           228
                    WITH THE 2 EXCEPTIONS FOR RECORD LENGTH (TO ACCOMODATE VARIABLE        228
                    LENGTH RECORDS) AND RELATIVE RECORD (TO ACCOMODATE RELATIVE RECORD     228
                    DATASETS), THESE DATA NAMES MUST BE POPULATED PRIOR TO CALLING THE     228
                    ROUTINE TO OPEN THE DATASET AND MUST NOT THEN BE CHANGED UNTIL THE     228
                    DATASET HAS BEEN CLOSED.                                               228
                  ************************************************************************/ 228
                                                                                           249
77                     DECLARE                                                             251
                          1 VSIO_FILE_BLOCK           STATIC,                              252
                             2 VSFB_DDNAME            CHAR(8)   INIT(' '),                 253
                             2 VSFB_ORGANIZATION      CHAR(4)   INIT(' '),                 254
                             2 VSFB_ACCESS            CHAR(10)  INIT(' '),                 255
                             2 VSFB_MODE              CHAR(6)   INIT(' '),                 256
                             2 (VSFB_RECORD_LENGTH,                                        257
                                VSFB_KEY_POSITION,                                         258
                                VSFB_KEY_LENGTH)      FIXED BINARY(15,0) INIT(0),          259
                           2 VSFB_FILE_STATUS         CHAR(1)   INIT('C'),                 260
                           2 VSFB_RESERVED            CHAR(161);                           261
                                                                                           262
                  /************************************************************************ 263
                                          END OF VSAMIOFB COPY BOOK                        263
                   ************************************************************************/ 263
                                                                                           153
78                       END RRDSLOR;                                                      154
```

ATTRIBUTE AND CROSS-REFERENCE TABLE

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
75		CARRIAGE_CONTROL	IN PRINT_LINE,AUTOMATIC,UNALIGNED,INITIAL,STRING(1),CHARACTER
73		COUNTER_EDIT	AUTOMATIC,UNALIGNED,DECIMAL,PICTURE(ZZ,ZZZ,ZZ9V) 46,47
72		IMAGES	FILE,EXTERNAL,INPUT,RECORD,SEQUENTIAL,ENVIRONMENT(F) 9,10,34,41
73		MORE_RECORDS	AUTOMATIC,UNALIGNED,STRING(1),BIT 11,18,33,35
73		NO	AUTOMATIC,UNALIGNED,INITIAL,STRING(1),BIT 11
75		PRINT_AREA	IN PRINT_LINE,AUTOMATIC,UNALIGNED,STRING(120),CHARACTER 12,14,16,47,59,61,63,65,67,69
75		PRINT_LINE	AUTOMATIC,STRUCTURE 13,15,17,48,60,62,64,66,68
72		PRINTR	FILE,EXTERNAL,OUTPUT,RECORD,SEQUENTIAL,ENVIRONMENT(F CTLASA) 9,13,15,17,48,60,62,64,66,68
73	*********	RECORD_COUNTER	AUTOMATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0) 46,55,55
74		RECORD_FIELDS	IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(70),CHARACTER
74		RECORD_IDENT	IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(7),CHARACTER
74		RECORD_IMAGE	AUTOMATIC,STRUCTURE 27,34,43,52
74		RECORD_SEQUENCE	IN RECORD_IMAGE,AUTOMATIC,UNALIGNED,STRING(3),CHARACTER 37
1		RRDSLOR	ENTRY,DECIMAL,FLOAT(SINGLE)
		SYSPRINT	FILE,EXTERNAL 5,6,7

```
DCL NO.          IDENTIFIER                      ATTRIBUTES AND REFERENCES

                 VSAMIOP                         EXTERNAL,ENTRY,DECIMAL,FLOAT(SINGLE)
                                                 27,43,52

77               VSFB_ACCESS                     IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(10),CHARACTER
                                                 21

77               VSFB_DDNAME                     IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(8),CHARACTER
                                                 19

77               VSFB_FILE_STATUS                IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(1),CHARACTER

77    ********** VSFB_KEY_LENGTH                 IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0)
                                                 25,37

77    ********** VSFB_KEY_POSITION               IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0)
                                                 24

77               VSFB_MODE                       IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(6),CHARACTER
                                                 22

77               VSFB_ORGANIZATION               IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(4),CHARACTER
                                                 20

77    ********** VSFB_RECORD_LENGTH              IN VSIO_FILE_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED(15,0)
                                                 23

77               VSFB_RESERVED                   IN VSIO_FILE_BLOCK,STATIC,UNALIGNED,STRING(161),CHARACTER

76               VSIO_CLOSE                      IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8),
                                                 CHARACTER
                                                 42

76               VSIO_COMMAND                    IN VSIO_PARAMETER_BLOCK,STATIC,UNALIGNED,INITIAL,STRING(8),
                                                 CHARACTER
                                                 26,42,51,59

76               VSIO_DELETE                     IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8),
                                                 CHARACTER

76               VSIO_DIRECT                     IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10),
                                                 CHARACTER
                                                 21

76               VSIO_DYNAMIC                    IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10),
                                                 CHARACTER
```

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
58		VSIO_ERROR	ENTRY,DECIMAL,FLOAT(SINGLE) 30,45,54
76		VSIO_ESDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4), CHARACTER
76	*********	VSIO_FB_DUPLICATE_RECORD	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_FB_KEY_SEQUENCE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_FB_NO_MORE_SPACE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_FB_READ_WITHOUT_START	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_FB_RECORD_NOT_FOUND	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
77		VSIO_FILE_BLOCK	STATIC,STRUCTURE 27,43,52
76		VSIO_INPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER
76		VSIO_INPUT_OUTPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER
76		VSIO_KSDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4), CHARACTER
76		VSIO_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER 26
76		VSIO_OUTPUT	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(6), CHARACTER 22
76		VSIO_PARAMETER_BLOCK	STATIC,STRUCTURE 27,43,52

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
76		VSIO_PARAMETER_VALUES	STATIC,STRUCTURE
76	*********	VSIO_RC_ACCESS_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_DATASET_ALREADY_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_DATASET_NOT_OPEN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_DDNAME_BLANK	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_END_OF_FILE	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_LOGIC_ERROR	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_MODE_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_MODE_UNSUPPORTED	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_ORG_ACCESS_MISMATCH	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_ORGANIZATION_UNKNOWN	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76	*********	VSIO_RC_SUCCESS	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0) 28,44,53
76	*********	VSIO_RC_UNKNOWN_COMMAND	IN VSIO_PARAMETER_VALUES,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
76		VSIO_READ	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
76	*********	VSIO_RETURN_CODE	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0) 28,44,53,61

DCL NO.		IDENTIFIER	ATTRIBUTES AND REFERENCES
76		VSIO_REWRITE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
76		VSIO_RRDS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(4), CHARACTER 20
76		VSIO_SEQUENTIAL	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(10), CHARACTER
76		VSIO_START_EQUAL	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
76		VSIO_START_NOTLESS	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER
76	*********	VSIO_VSAM_FEEDBACK	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_FEEDBACK_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 67
76	*********	VSIO_VSAM_FUNCTION	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_FUNCTION_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 65
76	*********	VSIO_VSAM_RC	IN VSIO_PARAMETER_BLOCK,STATIC,ALIGNED,INITIAL,BINARY,FIXED (15,0)
		VSIO_VSAM_RETURN_CODE	AUTOMATIC,ALIGNED,DECIMAL,FLOAT(SINGLE) 63
76		VSIO_WRITE	IN VSIO_PARAMETER_VALUES,STATIC,UNALIGNED,INITIAL,STRING(8), CHARACTER 51
50		WRITE_RR	ENTRY,DECIMAL,FLOAT(SINGLE) 38
73		YES	AUTOMATIC,UNALIGNED,INITIAL,STRING(1),BIT 18

AGGREGATE LENGTH TABLE

STATEMENT NO.	IDENTIFIER	LENGTH IN BYTES
75	PRINT_LINE	121
74	RECORD_IMAGE	80
77	VSIO_FILE_BLOCK	196
76	VSIO_PARAMETER_BLOCK	16
76	VSIO_PARAMETER_VALUES	158

STORAGE REQUIREMENTS.

THE STORAGE AREA FOR THE PROCEDURE LABELLED RRDSLOR IS 536 BYTES LONG.

THE STORAGE AREA FOR THE ON UNIT AT STATEMENT NO. 3 IS 184 BYTES LONG.

THE STORAGE AREA FOR THE ON UNIT AT STATEMENT NO. 10 IS 176 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED WRITE_RR IS 176 BYTES LONG.

THE STORAGE AREA (IN STATIC) FOR THE PROCEDURE LABELLED VSIO_ERROR IS 256 BYTES LONG.

THE PROGRAM CSECT IS NAMED RRDSLOR AND IS 1734 BYTES LONG.

THE STATIC CSECT IS NAMED RRDSLORA AND IS 5632 BYTES LONG.

STATISTICS MACRO RECORDS = 265,SOURCE RECORDS = 267,PROG TEXT STMNTS = 78,OBJECT BYTES = 1734

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN ON UNIT

OFFSET (HEX) 0000 0050 005C 007A 0094 00B2
STATEMENT NO 3 4 5 6 7 8

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN ON UNIT

OFFSET (HEX) 0000 0048 0052
STATEMENT NO 11 11

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE WRITE_RR

OFFSET (HEX) 0000 0038 003E 005E 006A 0078 0088 008E
STATEMENT NO 50 51 52 53 54 55 56 57

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE VSIO_ERROR

OFFSET (HEX) 0000 0038 0052 006A 00B6 00CE 0104 011C 014E 0166 0198 01B0 01BC 01C2
STATEMENT NO 58 59 60 61 62 63 64 65 66 67 68 69 70 71

TABLE OF OFFSETS AND STATEMENT NUMBERS WITHIN PROCEDURE RRDSLOR

OFFSET (HEX) 0000 00F0 00FE 0108 0116 0122 013A 0146 015E 016A 0182 0188 018E 0194 019A 01A0 01A6 01AC 01B2 01B8 01D4
STATEMENT NO 1 2 9 10 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

OFFSET (HEX) 01E0 01E0 01EA 01F0 01F0 01F8 0210 0218 0218 0232 023C 023C 0240 024A 0250 026C 0278 0282 029C 02B8 02D0
STATEMENT NO 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49

OFFSET (HEX) 02D6
STATEMENT NO 78

COMPILER DIAGNOSTICS.

WARNINGS.

IEM0227I NO FILE/STRING OPTION SPECIFIED IN ONE OR MORE GET/PUT STATEMENTS. SYSIN/SYSPRINT HAS BEEN ASSUMED IN EACH CASE.

IEM0764I ONE OR MORE FIXED BINARY ITEMS OF PRECISION 15 OR LESS HAVE BEEN GIVEN HALFWORD STORAGE. THEY ARE FLAGGED '*********' IN THE XREF/ATR LIST.

IEM1790I DATA CONVERSIONS WILL BE DONE BY SUBROUTINE CALL IN THE FOLLOWING STATEMENTS 37, 63, 65, 67.

END OF DIAGNOSTICS.

AUXILIARY STORAGE WILL NOT BE USED FOR DICTIONARY WHEN SIZE = 138K

COMPILE TIME .00 MINS

ELAPSED TIME .00 MINS

```
 F64-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED   NONE
          DEFAULT OPTION(S) USED -  SIZE=(231424,55296)
****GO        DOES NOT EXIST BUT HAS BEEN ADDED TO DATA SET
AUTHORIZATION CODE IS         0.
```

```
RRDSLODR: WRITE RRDS DIRECT (VACANT SLOTS)
-----------------------------------------

       100 RECORDS WERE LOADED SUCCESSFULLY
```