```
SCATTER GATHER (SG,44)  1Ø/2/73  JBL/JGC

/FLAG 1=1_BLOCK,Ø_ITEM
/FLAG 2=1_HELD, Ø_NOT HELD
/FLAG 3=1_NON-ADDRESSED, Ø_ADDRESSED

ESG:      JSP .FSET
          DIO TRACK
          LAC B5
          SZF 2
          LAC B3
          SZF I 3
          CLA
          DIP ESG1
          LAW 1
          SZF I 3
          JMP ESG6
          SZF 1
          JMP .ESG5
          LAC I (DCH 7Ø)
          DAP ESG4       /AC 6-17 NOW IN ESG4
          LAW I 7777
          AND I ESG4     /FIRST WORD OF FIRST COMMAND PAIR
          SAS B3         /WRITE
          JMP I 16SPCERR
          LAC I ESG4
          DAP ESG4       /ADDRESS OF FIRST WORD IN ITEM, I.E. WORD COUNT
          LAC I ESG4
          SAS ONE
          SPQ
          JMP I 16SPCERR
          MUL ONE        /TO DIVIDE FORMAT
          DIV C61
C7777,    7777
          SNI I
          IDA
.ESG5,    IOR TRACK


.ESG5+1,  JDA TRACK
ESG1,     ØØØØØØ

          JSP WAIT4
          JMP ESG3
```

```
/ADDRESSED SG OPERATION--CALL WHERE ONLY IF WRITING OR ZEROING

ESG6,     LAC I (DCH 7Ø)
          DAP ESG4
ESG7,     LAC I ESG4      /GET COMMAND
          SPA
          JMP PSWHER      /DONE, NO WRITING
          SUB (Ø4ØØØØ)
          SMA
          JMP ESG2        /FOUND A WRITE, CALL WHERE
          IDX ESG4
          IDX ESG4        /NEXT COMMAND
          JMP ESG7

PSWHER,   LAC I (DCH 72) /WHERE DOES THIS FOR TRACK
          SPA
          JMP I 16SPCERR
          SAR 1S
          IAI
          JMP .ESG5+1

ESG2,     JSP WHERE
          JMP .ESG5+1

/FLAG 4=1⊥FIRST BLOCK CHECKING ALREADY DONE
/FLAG 5=1⊥WRITING ALLOWED IN THIS ITEM

BUFR=17ØØØØ BUFF

ESG4,     Ø

ESG3,     CLF 6
ESG5,     CLF 4
          CLF 5
          SZF 2
          STF 5
          CLF 2
          LAW 7777
          AND I AC
          IOR B3
          DAC TABPTR
          LAC I IO
          DAC NXTTAG
          LAW CMPERR
          DAP COMPX
          DZM RWSTAT
          DZM ERSTPS
          DZM COMSTP
          DZM COMPST
          DZM BLKSUM
          LAW BUFF-1
          DAP GBKADR
          SZF I 1
          JMP SGIT
```

```
/READ SCATTER GATHER BLOCK

SGBL,     STF 5            /ALLOWS WRITES
          STF 4            /DON'T DO FIRST BLOCK OF ITEM JAZZ
          JSP GTCMTA
          JSP SVPTRS
          LAW 50.
          DAC LNGTH
          LAC NXTTAG
          SZF 3
          JSP GBKAD
          DAC NXTTAG
          DAC TAGWD
          LAW 100
          SZF I 3
          DAC COMPST
          LAC THSBUF
          DAP SGBL1
          DAC CSPTR
          JSP SGCS
          LAC NXTTAG
          JDA .READ
          JSP GO
SGBL1,    I I I .
          LAC BUFF
          JMP SGBLX
```

```
SGTAP1,   JSP GO
SGTAP2,   I I I
SGTAP3,   JMP SWAPE1



SGIT,     LAW 49.
          DAC LNGTH
          DZM CMDWCT
          JSP GTCMTA     /GET FIRST TABLE PAIR
          DZM TOTBLK
          DZM PULPTR     /?
          DZM BLKSWD
          DZM 1ERR

          LAC NXTTAG
          SZF 3
          JSP GBKAD
          DAC TAGWD
          DAC NXTTAG+2
          SZF I 3
          JMP ITØ3
          LAC I OWNWRD
          DAC NXTTAG+1
          STF 5
          JSP GBKAD
          SAD MZERO
          LAW 1
          IOR BØ
          DAC NXTTAG
          LAC B3         /GET THE WORD COUNT
          IOR WD1
          DAC PULPTR
          JMP ITØ3

GBKADI,   DAP GBKADX
          JMP GBKADX+4
GBKAD,    DAP GBKADX
          IDX .+1
GBKADR,   LAC .
          SMA
GBKADX,   JMP .
          LIO FBLOCK
          SNI
          JMP GBKADX
GBKADX+4, LAW BUFF
          DAP GBKADR
          JSP WAIT
          LAC TAGWD
          JDA GBLOCK
          JMP GBKADR
```

```
ITØ,        SZF I 3
            JMP ITØ3
            JSP GBKAD
            SPA
            LAW 1
            DAC NXTTAG


ITØ3,       DZM RWSTAT
            LIO THSBUF
            LAC NXTBUF
            DIO NXTBUF
            DAC THSBUF
            DAC CSPTR
            DAP IT2
            JSP SVPTRS
            JSP SGCS

IT1,        SZF 4
            JSP WAIT
            LAC TAGWD
            SAD ONE
            JSP ILLITM
            JDA ,READ
            JSP GO
IT2,        I I I .
```

```
IT6,      LAC NXTTAG     /JUST READ OR WRITTEN, IN EITHER
          DAC TAGWD      /CASE, THE NEXT BLOCK TO BE ACCESSED
          SZM"U"SZF 4
          JMP SGIT8      /NOT FIRST BLOCK AND AC>Ø
          SPA
          SZF 4
          JSP WILLIT
          XOR BØ
          SAD NXTTAG+2
          JSP WILLIT
          CLA"U"STF 4    /FIRST BLOCK AND AC<Ø
          LIO I PULPTR
          SAS PULPTR
          DIO WRDCNT
          LAC NXTTAG+2
          DAC NXTTAG+1
SGIT8,    LAC NXTTAG+1
          SAS NXTTAG+2
          JSP WILLIT
          CLA
          SAS ERSTPS
          STF 2          /WORKS

          SZF I 2
          JMP .+3
          SAD NEWDCT     /QUIT IF NO CHANGE IN LENGTH OF ITEM
          JMP SGIT2Ø     /AND NO MORE COMMANDS
          LAW 49.
          ADD TOTBLK
          DAC TOTBLK
          SUB WRDCNT
          SMA
          JMP SGIT2Ø     /LAST BLOCK ALREADY DONE
          ADD C61
          DZM BLKSUM
          SPA
          JMP ITØ
          DAC BLKSUM
```

```
SGIT11,    LAC BLKSWD       /LAST BLOCK IN CURRENT ITEM COMING NEXT
           SZA I
           JMP IT0          /NO CHANGE IN THE NO. OF BLOCKS
           IDX RWFLAG
           STF 3
           JMP RSTRT1


RBIBUF,    DAP RBIBFX       /READ CURRENT BLOCK INTO BUFF 100
           LAC TAGWD
           DAP DCS+1
           JDA .READ
           JSP GO
           I I I DCS
RBIBFX,    JMP .

DCS,       560001
           200000 .
CMUMBL,    630000 NXTTAG
           040001
DCS+4,     630000 BUFF 100
           100062
           000000
```

```
SGIT20,    JSP WAIT         /WAY OUT
           CLA
           SAS 1ERR
           JMP 1ERR+1
           LAW I 7777
           AND I TABPTR
           SAS (770000
           JSP ILLSPC
           LAC WRDCNT       /NO. OF WORDS REQUESTED
           SUB CMDWCT       /TO BE PROCESSED
           SPA              /GREATER
           JSP ILLSPC       /THAN WORD COUNT
           LAC NXTTAG+2
SGBLX,     SZF 3
           DAC I IO
SGRET1,    JMP SWAPE1
```

```
/BEGINNING OF THE ACTUAL DC SEQUENCE SETTING UP

SGCS,     DAP REDX
          LAC RITC1       /(56Ø1Ø1
          SZF I 3         /FLAG 3=1⊥NON-ADDRESSED, Ø⊥ADDRESSED
          SZF I 1         /FLAG 1=1⊥BLOCK, Ø⊥ITEM
          XOR C1ØØ        /NOT AN ADDRESSED BLOCK
          DAC I CSPTR
          IDX CSPTR

          LAC .           /READ TAG, READ DATA
          SZF 3
          LAC SGCS        /WRITE TAG, WRITE DATA
          DIP I CSPTR
          LAC TAGWD
          SZA I
          JSP WILLIT
          DAP I CSPTR
          IDX CSPTR

          LAC CMUMBL      /(63ØØØØ NXTTAG
          DAC I CSPTR
          IDX CSPTR
          LAC C6ØØØ1
          SZF 1
          JMP SGCS1       /BLOCK
          LAC C5ØØØ1      /ITEM
          DAC I CSPTR
          IDX CSPTR

          LAC C4ØØØ1
SGCS1,    SZF 3
          AND (-2ØØØØ     /NON-ADDRESSED BLOCK
          DAC I CSPTR
          IDX CSPTR

          SZF I 4         /FIRST BLOCK OF ITEM SWITCH
          JMP FBOI
```

```
DISPAT,    LAW DISPTB
           ADD COMMAN
           DAP DISPT1
           SZF I 2
DISPT1,    JMP .

EOTAB,     LAW I 1
           ADD CSPTR
           DAC CSPTR
           LAC I CSPTR      /CHANGE Ø4ØØXX TO 1ØØØXX
           AND (37777)
           JMP RED11

DISPTB,    JMP IREAD
           JMP SKIP
           JMP COMPAR
           JMP NONCOM
           JMP WRIT
           JMP WZERO
```

```
RSTATC,   DAP RSTATX     /READ MODE STATUS CHECK
          LAC RWSTAT     /IO TRANSPARENT
          SAD ONE
RSTATX,   JMP .
          SZF 3
          JMP REGBOU
          SZA
          JMP RESTRT
          IDX RWSTAT
          JMP RSTATX

SKIP,     JSP RSTATC
          LAC DCS+4      /UNTIL DAC ALL 1'S IS RELIABLE
          JMP RED1

CLIO,COMPAR,             LIO C100
          JMP . 2

NONCOM,   LIO C200
          JSP RSTATC
          LAC COMPST
          IAI
          XAI
          SZA
          JMP RESTRT     /BOTH REQUESTED IN SAME BLOCK
          DIO COMPST
          LAC I THSBUF   /START OF COMMAND SEQ. = 560001
          IAI
          DAC I THSBUF
          LAC B4         /DATA DIRECTION BIT
          DAC COMPBT
          JMP RED05      /SKIP ERROR CHECKING
```

```
IREAD,    JSP RSTATC
RED00,    LAC B5          /(I)
          SUB WD1
          SUB WD2
          SPA
          JMP REGMAX
RED04,    LAC WD1
          SUB BOUND
          SPA
          JMP REGBOU      /BOUND ERROR
RED05,    LAC WD1
          IOR SCORE

RED1,     DAC I CSPTR     /DATA ADDRESS
          IDX CSPTR
          LAC BLKSUM
          SUB LNGTH
          LIA
          ADD WD2
          SPQ
          JMP RED2
          DAC WD2
          LAC LNGTH
          DAC BLKSUM
          CMI"U"LAI
          ADD WD1
          DAC WD1
          LAI
RED12,    IOR COMPBT
RED11,    IOR B2
          DAC I CSPTR     /WD COUNT
          IDX CSPTR
RED13,    DZM I CSPTR     /D.C. HALT
          LAW 2
          SAS RWSTAT
REDX,     JMP .
          LAC LNGTH
          SAS BLKSUM
          SZF 3
          JMP REDX
          JMP RESTRT
```

```
RED2,     LIA
          ADD LNGTH
          DAC BLKSUM
          LAC B3
          SNI
          LAC B2
          IOR WD2
          IOR COMPBT
          DAC I CSPTR
          CLA
          SAD COMSTP
          JMP .+4
          IDX ERSTPS
          LAC WD2
          JMP RED12
          IDX CSPTR
          SNI I
          JMP GTCMTB
          JSP GTCMTA
          JMP RED13
```

```
WSTATC,   DAP WSTATX      /WRITE MODE STATUS CHECK
          SZF I 5
          JMP REGBOU
          LAC RWSTAT
          SAD TWO
WSTATX,   JMP .
          SZA
          JMP RESTRT
          IDX RWSTAT
          ADD THSBUF
          DAC JUNK
          LAC B4
          IOR I JUNK
          DAC I JUNK
          IDX RWSTAT
          JMP WSTATX

WZERO,    JSP WSTATC
          CLC
          JMP RED1

WRIT,     JSP WSTATC
          JMP REDØ5

REGMAX,   ADD WD2
          DAC WD2
          LAC B2
          DAC 1ERR
          DAC COMSTP
          JMP REDØ4

REGBOU,   LAC B3
          DAC 1ERR
          STF 2
          IDX ERSTPS
          JMP EOTAB
```

```
/FIRST BLOCK OF ITEM ROUTINE
/DATA CHANNEL COMMANDS ALREADY SET UP:
/  560001
/  20XXXX
/  630000+NXTTAG
/  050001
/  040001

FBOI,     SZF 3
          JMP DISPAT
          SZF 2
          JMP SGRET1
          DZM WRDCNT
          DZM NEWDCT
          LAC COMMAN
          SZA                /READ
          SAD TWO            /CONT. ON COMPARE
          JMP FB11           /SET UP PULPTR
          SAS THREE          /CONT. ON NON-COMPARE
          SAD FOUR           /WRITE
          JMP FB02
          SAD ONE            /SKIP
          JMP FB10           /READ ONE WORD INTO WRDCNT
          JSP ILLSPC

FB039,    DAP FB0399
          IDX WD1            /SUBROUTINE USED TWICE, SAVES 2 REGISTERS
          LAW I 1
          ADD WD2
          DAC WD2
          SZA I
          JSP GTCMTA
FB0399,   JMP .

DIVC61,   DAP DVC61X
          RIL 1S
          CLA
          DIV C61
C777,     777
          SNI I
          IDA
          SAD ONE
          STF 3
DVC61X,   JMP .
```

```
/FOR SKIP THE WORD COUNT
FB10,   LAC (630000+WRDCNT
        DAC I CSPTR
        IDX CSPTR
        LAC C40001
        DAC I CSPTR
        IDX CSPTR
        JSP FB039
        IDX BLKSUM
        LAW 4
        SAS COMMAN
        JMP DISPAT      /CONTINUE WITH SGCS
FB01,   JSP UNSVPT
        JMP FB02

FB11,   LAC WD1         /SET UP PULPTR, TO TRANSFER ONE WORD FROM USER CORE
        IOR B3          /ON READ OR CONT, ON COMPARE OF WORD COUNT
        DAC PULPTR
        LAW 1
        SAS WD2
        JMP DISPAT
        LAW I 7777
        AND I TABPTR
        SAS B3
        JMP DISPAT

FB02,   JSP RBIBUF      /READ BLOCK INTO BUFF
        LAC NXTTAG
        SMA
        JSP ILLITM
        LAC BUFF 101
        DAC WRDCNT
        SZM
        SAD ONE
        JSP ILLITM

        LAC WD1
        IOR B3
        DAC SPPTR1
```

```
                LAW 4
                SAS COMMAN
                JMP FBØ3
                LAC I PTR1
                SAD WRDCNT
                JMP FBØ3
                DAC NEWDCT
                SZM
                SAD ONE
                JSP ILLSPC
                LIA
                JSP DIVC61
                DAC BLKSWD
                LIO WRDCNT
                JSP DIVC61
                SUB BLKSWD
                DAC BLKSWD
                DAC FBLOCK
                SZA I
                CLF 3
                LIO NEWDCT
                SMA
                DIO WRDCNT

FBØ3,           JSP FBØ39
                LAC WD1
                IOR B3
                DAC SPPTR1
                LAC COMMAN
                SAD (5)
                JSP ILLSPC
                SAS FOUR
                JMP FBØ8

FBØ7,           SZF 5
                JMP FBØ9
                LAC BUFF 1ØØ
                DAC NXTTAG+1     /IF ENTIRE FIRST BLOCK IS TO BE WRITTEN
                SAS I OWNWRD
                JMP RTERR        /PASS WORD INCORRECT
                LAC I SPPTR1
                SAS BUFF 1Ø2
                JMP RTERR        /REWRITE NUMBER ERROR
                IDX I SPPTR1     /INCREMENT REWRITE NO.S
FBØ9,           LAC I SPPTR1
                DAC BUFF 1Ø2
                STF 5            /INDICATES WRITING LEGAL IN THIS ITEM
FBØ8,           IDX ALREDS       /SET ALREADY READ SWITCH
```

```
/HERE WHEN INCOMPATIBLE COMMANDS OCCUR

RESTRT,   JSP UNSVPT      /RESTORE POINTERS TO START OF BLOCK STATE
          DZM RWFLAG

RSTRT1,   DZM BLKSUM      /ENTRY FOR TRANSITION BLOCK
          LAC ALREDS      /CHECK IF ALREADY READ
          DZM ALREDS      /IF SO, SKIP READING AND ERROR CHECKING
          SZA
          JMP REST15
          JSP WAIT
          JSP RBIBUF
RESTØ,    SZF 1
          JMP REST1
          LAC NXTTAG+2
          SZF 4
          SAD BUFF 1ØØ
          JMP REST15
          JSP ILLITM
REST1,    LAC NXTTAG
          SAS TAGWD
          JSP ILLITM      /NOT A BLOCK

REST15,   LAC (BUFR 1ØØ)
          SZF I 1
          IDA
          DAC SPPTR2
```

```
REST2,     LAC SPPTR2
           SAD (BUFR 162
           JMP FINA
           LAC WD2
           SZA I
           JSP GTCMTA
           SZF 2
           JMP FINA
           LAC WD1
           IOR BØ+3        /RENAME IOP USER CORE
           DAC SPPTR1
           LAC WD2
           ADD BLKSUM
           SUB LNGTH
           SZM
           JMP REST3
           DZM WD2
           ADD LNGTH
           DAC BLKSUM
           ADD (BUFR 1ØØ)
           SZF I 1
           IDA
           JMP REST4
REST3,     DAC WD2
           LAC WD1
           ADD LNGTH
           SUB BLKSUM
           DAC WD1
           LAC (BUFR 162
REST4,     DAC TERMIT
           LAW DISP2
           ADD COMMAN
           DAP .+2
           SZF I 2
           JMP .
           JMP FINA

DISP2,     JMP RREAD
           JMP RSKIP
           JMP RCOMP
           JMP RNONC
           JMP RWRIT
           JMP RZERO
```

```
RREAD,     LAW 7777
           AND SPPTR1
           SUB BOUND
           SPA
           JMP INTBOU        /BOUND ERROR

RREAD1,    LAC (407777)
           SUB SPPTR1
           SPA
           JMP INTMAX
           LAC I SPPTR2
           DAC I SPPTR1
           IDX SPPTR1
           IDX SPPTR2
           SAS TERMIT
           JMP RREAD1
           JMP REST2

RSKIP,     LAC TERMIT
           DAC SPPTR2
           JMP REST2

RWRIT,     SZF I 5
           JSP ILLSPC
           IDX RWFLAG
RWRIT3,    LAC I SPPTR1
           DAC I SPPTR2
RWRIT4,    IDX SPPTR1
           IDX SPPTR2
           SAS TERMIT
           JMP RWRIT3
           JMP REST2
```

```
RCOMP,    LAW LOSE
          DAP COMP1
          LAW REST2
          DAP COMP2
          JMP COMPS

RNONC,    LAW RSKIP      /WIN
          DAP COMP1
          LAW LOSE
          DAP COMP2

COMPS,    LAC I SPPTR1
          SAS I SPPTR2
COMP1,    JMP .
          IDX SPPTR1
          IDX SPPTR2
          SAS TERMIT
          JMP COMPS
COMP2,    JMP .

RZERO,    SZF I 5
          JSP ILLSPC
          IDX RWFLAG
          DZM I SPPTR2
          IDX SPPTR2
          SAS TERMIT
          JMP .-3
          JMP REST2


CMPERR,   RRI 7ØØ        /GET DAC
          LAW I 1
          AAI
          AND (177777)
          SAD (I I I NXTTAG)
          JSP ILLITM
          AND C7777
          LIA
          LAW 3          /CONT. ON NON-COMPARE MUST BE DONE INTERPRETIVELY
          SAS COMMAN
          JMP CMPER2
          CLA
          SAS WD2        /THIS IS A CLUDGE
          JMP IT6
CMPER2,   DIO I ERCOD2
CMPER1,   LAW 4ØØØ
          DAC 1ERR
          CLA
          SZF I 1
          SAD NEWDCT
          JMP 1ERR+1
          STF 2
          JMP IT6
```

```
FINA,           LAW FINX+1
FINB,           DAP FINX
                CLA
                SAD WD2
                JSP GTCMTA
                CLA
                SAD RWFLAG
                JMP FINX
                SZF I 3
                JMP FINE
                LIO BLKSWD
                SPI
                JMP FINZ
                LAC NXTTAG
                DAC DRMPTR
                LAW EXP4
                LAW ILLITM
                DAP COMPX
                LAC I IO
                DAC PTR1
                LAW 1
                JMP FINC

FINZ,           LAC NEWDCT
                DAC WRDCNT
                DZM BLKSWD
                JSP GBKADI
FINC,           SZF I 4
                IOR BØ
                DAC NXTTAG
                LAC B3
FINE,           IOR CLIO
                DIP DCS+1
                LAC TAGWD
                JDA .READ
                JSP GO
                I I I DCS
                LAC .
                DIP DCS+1
FINX,           JMP .
                SZF I 1
                JMP IT6
                JMP SWAPE1


LOSE,           JSP FINB
                LAW 7777
                AND SPPTR1
                DAC I ERCOD2
                JMP CMPER1

INTMAX,         LAC B2
                JMP .+2
INTBOU,         LAC B3
                DAC 1ERR
                STF 2
                JMP FINA
```

```
/GET NEXT COMMAND PAIR

GTCMTB,   LAW DISPAT     /OPEN ROUTINE EXIT
GTCMTA,   DAP GTCMTX     /CLOSED ROUTINE ENTRY POINT
          DZM COMPBT
          DZM WD1
GETCM3,   LAC I TABPTR   /OR SOME SUCH
          DAP WD1
          RAL 6S
          AND (77)
          DAC COMMAN
          SAD (77
          JMP GETCM4
          SUB (6
          SMA
          JMP GETCM5
          IDX TABPTR
          LIO I TABPTR
          DIO WD2
          LAI
          ADD CMDWCT
          DAC CMDWCT
          IDX TABPTR
          SNI
          JMP GETCM3
          SPI I
GTCMTX,   JMP .

GETCM5,   LAC B3
          DAC 1ERR
GETCM4,   STF 2
          JMP GTCMTX

SVPTRS,   DAP SVPTRX
          LAC WD1
          DAC SWD1
          LAC WD2
          DAC SWD2
          LAC COMMAN
          DAC SCOMMA
          LAC TABPTR
          DAC STABPTR
          LAC CMDWCT
          DAC SCMDWC
SVPTRX,   JMP .

UNSVPT,   DAP UNSVPX
          LAC SWD1
          DAC WD1
          LAC SWD2
          DAC WD2
          LAC SCOMMA
          DAC COMMAN
          LAC STABPTR
          DAC TABPTR
          LAC SCMDWC
          DAC CMDWCT
          CLF 2
UNSVPX,   JMP .
```

```
/VARIABLES

THSBUF,   BUFR 162
NXTBUF,   BUFR 331

TAGWD,    0
NXTTAG,   0              0           0

COMPST,   0
COMPBT,   0

WD1,      0
WD2,      0
COMMAN,   0
CMDWCT,   0

TOTBLK,   0              /TOTAL NO. OF WORDS PROCESSED SO FAR THIS ITEM
BLKSWD,   0              /3 WAY SWITCH RE LENGTH OF ITEM
                                     /0 = NO CHANGE
                                     /+ = GETTING SHORTER
                                     /- = GETTING LONGER
RWSTAT,   0              /READ-WRITE STATUS
WRDCNT,   0              /NO. OF WORDS IN ITEM NOW, EXCEPT AFTER ADDING TO I
                         /STARTS, WHEN IT IS NO. OF WORDS IN ITEM
NEWDCT,   0              /WHAT WRDCNT WILL BE AFTER ADDING STARTS BEFORE AD
JUNK,     0              /A TEMP STORAGE REG
RWFLAG,   0              /SWITCH FOR INTERP, SIGNALS THAT THE BLOCK IS REWRI
ALREDS,   0              /BLOCK ALREADY READ SWITCH
TERMIT,   0              /1> THAN ADDR OF LAST WORD IN BUFF+100 PROCESSED
ERSTPS,   0              /ERROR STOP SWITCH, AFTER COMMANDS EXECUTED
COMSTP,   0              /ERROR STOP SWITCH, AFTER THIS COMMAND SET UP
SWD1,     0
SWD2,     0
SCOMMA,   0
STABPTR,  0
SCMDWC,   0

TABPTR,   0
BLKSUM,   0
LNGTH,    0              /49. OR 50.

CSPTR,    0
TAPCTR,   0              /COUNTS THE NUMBER OF COMMAND PAIRS USED
```

```
/FASTRAND ERROR RETURNS
1ERR,      Ø
           JSP WAIT4
           LAC 1ERR
           LIO DCONT         /IS IT TAPE
           SPI
           JMP I TERR
           LIO SWPFLG
           LFI
           DAC I ERCOD1
           LAW I 1
           SZF 6
           LAW I 2
           ADD I PC
           LIA
           DAC I TRAPPC
           IDA
           SZF I 6
           LAW IOPTSU
           SPI
           IOR BØ
           DAC I PC
           JMP SWAPE1

FHLT,      JSP METH1         /CHANNEL 4 TOTAL CRASH
1HLT,      CLC
           DAC ONTRAC
           LAC B1            /FASTRAND=CHANNEL 1 TOTAL CRASH
           JDA 1ERR

MAXERR,    LAC MAXM          /GIVE NO. OF WORDS READ TO USER
           SUB CORADR
           IDA
           DAC I ERCOD2
           LAC B2
           JDA 1ERR

ILLSPC,    DAC I ERCOD2
           LAC B3
           JDA 1ERR

NSERR,     LAW 4ØØ
           JDA 1ERR
```

```
/ERROR RETURNS WHEN THE CHANNEL STILL FLYING

WILLIT,    LIO 88
           DIO 1ERR
W1ERR,     DAC I ERCOD2
           JSP WAIT
           LIF
           RCR 6S
           DIP I ERCOD2
           JMP 1ERR+1
```

```
/MOVE BIT TABLES OUT

DONE,    DAC DONX
         DSC 100
         LSM
         CLI            /START WITH 1ST BIT TABLE
         DIO DON4
DON5,    JSP BTWRIT     /SET UP COMMANDS
          JMP DON6      /NONE THERE
         LIO (30000 NDONC1)
         JSP DGO        /WRITE IT OUT
         CLC
         DAC ONTRACK
DON6,    IDX DON4
         LIA
         SAS (4)        /DO ALL BIT TABLES
         JMP DON5
         LAC (270000)
         DIP NDONC2+1
         LIO (30000 NDONC2)
         JSP DGO
         ASC 100
         ESM
         JMP I DONX
DONX,    0
DON4,    0

DGO,     DAP DGOX
         LAI
DGO+2,   RRO+500
         XX
         RRI
         RIR 5S
         SPI
         JMP .-3
         RIL 5S
         SPI I
DGOX,    JMP .
         LIA            /AGH !  (ABNORMAL)
         JMP DGO+2

NDONC2,  560001
         213000         /FIXED-HEAD 3 HOLDS FRELST WHEN EXEC NOT RUNNING,
         617000
         040677
         100001
         000000
```

```
NDONC1, 560001
        300000 .
        140000
        260000 .
        610000 .
        040100
        630000 .
        100001
        000000

/SUBROUTINE TO WRITE OUT BIT TABLE
BTWRIT, DAP BTWRX
        LAW TRACKA
        AAI             /WHICH TABLE ?  3>=IO>=0
        DAP BTWR1
        DAP BTWR2
        LAW CNT1
        AAI
        DAP NDONC1+6
        LAW BITTB1
        RIL 6S
        AAI
        DAP NDONC1+4
BTWR1,  LAC .
        SPA             /IS THERE A BIT TABLE HERE ?
        JMP BTWRX       /NO.  R1
        CLI"U"CMI
BTWR2,  DIO .           /MARK IT
        SUB (FRELST)
        CLI
        RCR 2S
        IOR C400
        DAP NDONC1+1
        LAI
        RAR 6S
        DAP NDONC1+3
        IDX BTWRX       /R2 AFTER SETTING UP COMMANDS
BTWRX,  JMP .
```

```
/CORE 4,Ø REFERENCES FROM CORE 17

C16A,     CAL Ø
C16B,     CAL 1
IOPMAX,   IOR 74
ERCOD1,   IOR 1Ø2
ERCOD2,   IOR 1Ø3
OWNWRD,   IOR 1Ø4
TRAPPC,   IOR 77
BOUND,    36
OVERFLOW, -Ø
OVFX,     JMP .
COMPX,    JMP .

CONS,     CONSTANTS

FOO,      FLEXO FOO
372ØØ/    BUFF,
377ØØ/
CH1BRK,   DCH .
CH15BK,   DCH .
RØ,       DCH .
R1,       DCH .
R2,       DCH .
SWORG,    DCH .
HOTFLG,   DCH .
RSTAT,    DCH .
MSTAT,    .
INIT,     DCH .
DSWAPX,   DCH .
DCORE,    DCH .

3772Ø/
LISTPC,   TAD .                /NON CORE 16 STARTUP
ELSTPC,   TAD .
SWAP4,    TAD .
```

```
AC,       JDA 7775

37724/
T1,       XCT .          /CORE 1Ø POINTER

PC,       IOR 35
IO,       JDA 7776
FLAGS,    JDA 7777

3773Ø/
ABNORMAL, DCH I .
CONTR2,   DCH I .
FTAPE1,   DCH I .
4CHAN1,   DCH I .
NOTYOURS, DCH I .
16SPCERR, DCH I .
EMPTY,    DCH I .
TERR,     DCH I .
TAPPAR,   DCH I .
TAPSTT,   DCH I .
TUNIT,    DCH I .
INDEX1,   DCH I .
SWAPE7,   DCH I .
SWAPE2,   DCH I .
NETHND,   DCH I .

START HLT-JMP
```