```
STANDARD IOTS (STANDA,35) 10/2/73  JGC/JBL

/SWAPPER IOT'S FOR FASTRAND SWAPPING
SWAP1:    STF 1              /IN
SWAP3:    JSP .SETUP         /SWAP
          STF 6
          LAC SWAP4
          DAC I (DCH 71)
          LAC SWAPT
          JDA TRACK
          000000
          JSP WAIT4
          SZF 1
          JMP SWAP2.         /READ FROM UPPER 9 BITS
          LAC I AC
          JDA SWAPR
          260000
          SZF 2
          JMP SWAPE1
SWAP2..   LAC I AC
          RAR 9S
          JDA SWAPR
          200000
          LAC I (IOR 50) /SET UP INCOMING USER
          DAC I AC
          LAC I (IOR 51)
          DAC I IO
          LAC I (IOR 52)
          DAC I FLAGS
          JMP SWAPE1

SWAP2:    STF 2
          JMP SWAP3"A"7777

SWAPC0,   560001
          0
          0
SWAPC0 3, 040000 .
          100001
          0
SWAPC0 6, 040000 .
          100001
          0
SWAPC0 11,                   040000 .
          100001
          000000
```

```
SWAPR,      0
            DAC SWPRX
            LAC SCORE
            DAC SWAPC0 2
            LAW 777
            AND SWAPR
            MUL C81.
            DIV ONE
C81.,       81.
            IOR I SWPRX
            DAC SWAPC0 1
            AND (-77)
            ADD C100
            DAC SWAPC0+5
            ADD C100
            DAC SWAPC0 10
            LAW 77
            AND SWAPC0 1
            CMA
            ADD C100
            MUL C63
            DIV ONE
C63,        63
            SUB ONE
            DAP SWAPC0 3
            CMA
            ADD (7776)
            DAP SWAPC0 6
            SUB (6277)
            SZM
            JMP SWAPR1
            DZM SWAPC0 10
SWAPR2,     IDX SWPRX
            JSP GO
            I I I SWAPC0
            JMP I SWPRX

SWAPR1,     SUB ONE
            DAP SWAPC0 11
            LAW 6277
            DAP SWAPC0 6
            JMP SWAPR2
SWPRX,      0
SWAPT,      200000
```

```
/FIXED HEAD IOT'S

/INTERNAL SUBROUTINE FOR FIXED HEAD IOT'S
NUM,    DAP NUMX
        LAC FPAR        /GET ADDRESS NO.
        SUB (2100
        SMA
        JSP ILLSPC
        LAC FPAR
        MUL ONE
        DIV C17.
C17.,   17.
        IOR (217000     /SET TO READ FROM HEAD 7 THIS SECTOR
        DAC RNMC0+1
        LAI             /GET POINTER INTO BLOCK FROM REMAINDER
        SAL 1S
        AAI
        ADD (170000 BUFF
        DAC NUM1        /POINTER TO BLOCK
        DAC NUM2
        JSP GO
        I I I RNMC0
        JSP WAIT4       /WAIT FOR CORE TO SHOW UP
        LAW 7777        /GET LOW 12 BIIS OF AC
        AND I AC
        DAP RPMA        /FIRST WORD
        IDA
        DAP RPMB
        IDA
        DAP RPMC
NUMX,   JMP .
NUM1,   0
NUM2,   0

RNMC0,  560001
        0
        630000 BUFF
        100063
        000000

RPMA,   IOR .
RPMB,   IOR .
RPMC,   IOR .
```

```
/READ NUMBER
/IO=ADDRESS NO
/IF FLAG 2 SET READ AS I OR BLOCK (THEN HAS SAME SEQ AS IOT)

RNUM;     JSP .SETUP
          LAC I (DCH 72) /GET NUMBER
          JDA TRACK
          030000
          JSP NUM        /DECODE AND SET UP FOR READ
          LIO I NUM1     /GET NUMBER
          DIO I IO
          SZF I 3        /DOES HE WANT  REWRITE NO.
          JMP RNUM1
          IDX NUM1
          IDX NUM1
          LAC I NUM1
          DAC I AC
RNUM1,    SZF I 2
          JMP SWAPE1     /DONE
          SPI            /WANT TO READ IS IT LEGAL DRUM ADDRESS
          JSP ILLSPC     /NO
          LAW 7777
          AND I AC
          SUB BOUND
          SPA
          JSP ILLSPC     /BELOW BOUND
          LAI"U"CLF 2
          RAL 8S
          AND (177
          ADD (FRELST)
          LIF
          SWP
          AND (6)
          SAD (6)
          JSP ILLSPC
          SAL 6S
          AAI
          JDA READ
          AND (-177
          DAC THIRD
          JMP RDINF3
```

```
/WRITE FIXED HEAD NUMBER
/FLAG 1=1,REWRITE,0,WRITE
/FLAG 1=1;FLAG 2=1,EXPUNGE
WNUM;     JSP .SETUP
          LAW 7777
          AND I (DCH 70)
          DAC TRACK        /INDIRECT THRU IT
          LAC I TRACK
          JDA TRACK
          030000
          JSP NUM          /GET BLOCK INTO CORE
          LAW I 3          /IS IT LEGAL NAME
          DAC COUNT
          SZF 3
          JMP RDNUM
          SZF 1
          JMP RWNUM
WNUM2,    CLC
          SAS I NUM2
          JMP RTERR
          IDX NUM2
          ISP COUNT
          JMP WNUM2
          LAC I IO         /0.K.=STORE NUMBER
          DAC I NUM1
          IDX NUM1
          LAC I RPMB
          DAC I NUM1       /SAVE OWN=WORD
          IDX NUM1
          LAC I RPMC       /SAVE REWRITE NUMBER
          DAC I NUM1
RWNM4,    LAC (270000
          DIP RNMC0+1      /RESET TO WRIT E OUT BLOCK
          JSP GO
          I I I RNMC0
          JMP SWAPE1

RDNUM,    LAC I NUM2       /IOT 6610 READS THREE WRD BLK INTO CORE
          DAC I RPMA
          IDX NUM2
          LAC I NUM2
          DAC I RPMB
          IDX NUM2
          LAC I NUM2
          DAC I RPMC
          JMP SWAPE1
```

```
RWNUM,      IDX NUM2
            LAC I RPMB      /CHECK OWN=WORD
            SAS I NUM2
            JMP RTERR
            IDX NUM2        /LOOK AT REWRITE NUMBER
            LAC I NUM2
            SZA I
            JMP RWNM3
            SAS I RPMC
            JMP RTERR
            IDX I NUM2
            SZA I           /SKIP ZERO
            LAW 1
            DAC I NUM2
            DAC I RPMC
RWNM3,      SZF 2
            JMP ENUM        /EXPUNGE
            LAC I IO
            DAC I NUM1
            JMP RWNM4

/EXPUNGE NUMBER
ENUM,       CLI"U"CMI       /O.K.SET TO =0
            DIO I NUM1
            IDX NUM1
            DIO I NUM1
            IDX NUM1
            DIO I NUM1
            JMP RWNM4       /GO REWRITE THE BLOCK
```

```
/READ (OR WRITE) A FIXED-HEAD BLOCK.  USEFUL FOR DIAGNOSTICS OR
/  FOR "NCP" TO EXPUNGE 256. IVN'S IN A HURRY.

FIXHD:    JSP .SETUP
          LAW 7777
          AND I (DCH 70)
          SUB BOUND       /LEGAL CORE ADR ? (NO IOPMAX CHECK)
          SPA
          JMP I 16SPCERR
          LAW I 7077
          AND I (DCH 72) /LEGAL BLOCK NUMBER
          SZA
          JMP I 16SPCERR
          SZF I 1         /WRITING ?
          JMP FIXHD2      /NO. MERELY READING.
          LAC I NETHND    /YES. IS THIS THE "NCP" ?
          SAS 16STAT
          JMP I 16SPCERR
          LAC I (DCH 72) /IS THE "NCP" CHANGING THE NETWORK QUEUES ?
          SUB (7036)
          SPA
          JMP I 16SPCERR
          SUB (7055-7036)
          SZM
          JMP I 16SPCERR
FIXHD2,   JDA TRACK
           030000
          JSP WAIT4       /GET USER CORE BACK.
          LAC (210000)
          SZF 1
          LAC (270000)
          DIP FIXCMD+1
          LAC I IO
          DAP FIXCMD+1
          LAW 7777
          AND I AC
          IOR SCORE
          DAC FIXCMD+2
          JSP GO
           I+I+I FIXCMD
          JMP SWAPE1

FIXCMD,   560001
          0
          0
          100063
          0
```

```
/RELEASE HALF TRACK

RTRACK:   JSP .FSET
          STF 2
          JSP WHERE
          JDA TRACK
          000000
          LAC HTR
          CLI"U"CMI       /MARK BITS AS NOT IN CORE
          SAD TRACKA
          DIO TRACKA
          SAD TRACKB
          DIO TRACKB
          SAD TRACKC
          DIO TRACKC
          SAD TRACKD
          DIO TRACKD
          CLF 6
          RAR 8S
          AND (6000)
          DAP RTRKC0+1
          DAP RTRKC2+1    /SET STARTING DRUM ADDRESSES
          ADD (262000)
          DAC RTRKE
RTRK1,    DZM RTRKC0+7    /SET CORRECT HALT
          JSP WAIT2
          30000 RTRKC0
          170000 RTRKC1+7
          STF 6
          LAW 100
          ADD RTRKC0+1    /INCREMENT TO NEXT HEAD
          DAP RTRKC1+1
          DZM RTRKC1+7
          JSP WAIT2
          30000 RTRKC1
          JDA RTRKC0+7
          LAW 100
          ADD RTRKC1+1
          DAP RTRKC0+1
          SAS RTRKE       /WAS THIS THE LAST HALF TRACK
          JMP RTRK1       /NO
          LAW BUFF        /YES BUILD A BIT TABLE
          DAP RTRK2+1
RTRK2,    LIO THREE
RTRK2+1,  DIO .
          IDX RTRK2+1
          SAS (DIO BUFF 76
          JMP RTRK2+1
          JSP WAIT2
          30000 RTRKC2
          170000 RTRKC1+7
          SZF I 3
          DZM I HTR
          JSP WAIT4
          JMP SWAPE1
```

```
RTRKC0,    560001
RTRKC0+1,  260000 .
           630000 ZERO
           050001
MZERO,     777777
           046276
           100001
RTRKC0+7,  000000

RTRKC1,    560001
RTRKC1+1,  260000 .
           630000 ZERO
           050001
           777777
           046276
           100001
RTRKC1+7,  000000

RTRKC2,    560001
RTRKC2+1,  260000 .
           630000 RTRKF
           040002
           630000 BUFF
           040076
           630000 (-1776
           100001
           000000

RTRKF,     400003
           400003
RTRKE,     0               /END CONDITION STORAGE
```

```
.FSET,     DAP .FSETX
           JSP .SETUP
           LAI
           AND (6
           RAR 3S
           LIA
           SAS (600000
.FSETX,    JMP .
           JMP I 16SPCERR

/LINK SET,RETURN 3
.SETUP,    DAC .SETX
           LAC I RSTAT
           DAP 16STAT
           ADD MSTAT
           DAC 16USER
           LAC I (DCH 71)
           DAC RPC
           IDA
           SZF 6
           IDA
           CLL"U"SZL
           IDA
           DAC I (DCH 71)
           JMP I .SETX
.SETX,     0
RPC,       0

/ROUTINE TO CALCULATE IOPMAX FOR READS
MCHECK,    0               /ENTER WITH COUNT IN AC
           DAP MCHX
           LAC MCHECK
           LIO ONE
           ADD CORADR
           SUB MAXM
           SPQ
           JMP MCH1
           SAS ONE
           SWP"U"STF 1
MCH1,      ADD MAXM
MCHX,      JMP .
```

```
/READ INFORMATION
/FLAG 1=1 BLOCK, 0 ITEM
/FLAG 2=1 PUT ONLY DRAS IN USER CORE

RDINF:    JSP .FSET        /INITIALIZE IOT
          LAW 7777         /CHECK IF BELOW BOUND
          AND I (DCH 70)
          SUB BOUND
          SPA
          JMP I 16SPCERR
          LAC I (DCH 72) /CHECK IF LEGAL DRUM ADDRESS
          SPA
          JMP I 16SPCERR
          SAR 1S
          IAI
          JDA TRACK
ZERO,     000000           /MOVE BOOM TO PROPER LOCATION
          JSP WAIT4
RDINF3,   LIO I IO
          DIO DRMPTR
          LAW 7777
          AND I AC
          DAP RAITEM
RDINF1,   IOR SCORE
          DAC CORADR
          DAP CORPTR
          SUB C61          /SET UP DAC COMANDS IN ITEM LOOP
          DAC RAIDAC
          LAW 7777
          AND I IOPMAX
          SZA I
          LAW 7777
          IOR SCORE
          DAC MAXM
          DAC COUNT        /EXTRA SET FOR READ ITEM CASE
          SUB CORADR
          SPA
          JMP MAXERR       /IOPMAX<STARTING LOCATION
          SZF I 1
          JMP RIT          /READ ITEM
          CLF 7
          LAC DRMPTR       /SET HEAD-SECTOR
          DAP RBKC+1
          LAW 62           /CALCULATE WORD COUNT WANTED
          JDA MCHECK
          SUB CORADR
          DAP RBKC+5       /SET WORD COUNT
          JSP GO
          I I I RBKC
          LAC PTR1
          DAC I IO
          SAS DRMPTR
          JSP ILLITM
          JMP RIT4
```

```
RBKC,       560001
RBKC+1,     200000+,
            630000 PTR1
C40001,     040001
CORADR,     000000
            100000 ,
            000000
```

```
RIT,    LAW ILLITM      /MARK IN COMPARE MODE FOR ABNORMAL ROUTINE
        DAP COMPX
        LAW 61
        DAP RITC1+7
        CLC"U"CLF 3
        CLF 5
        DAC RITC1+4
        DAC RITC1+6
        LAC (040001)
        DAC RITC1+5
        DIP RAITEM
        DZM I ERCOD2
        LAC (30000
        ADD MAXM
        SZF 2
        DAC MAXM
        LAC DRMPTR
        DAC PTR1
        DAC PTR2
        DAC LTRK
RIT1,   LAC PTR1
        SZF I 5
        SAD ONE         /WAS THIS LAST BLOCK
        JMP RTRN        /YES
        SPQ
        JSP ILLITM      /ILLEGAL ITEM
        DAP RITC1+1
        XOR LTRK        /IS TRACK THE SAME
        AND (770000
        SZA I
        JMP RIT5        /CHANGING TRACKS?
        LAC PTR1        /YES
        JDA .READ
RIT5,   LAW 61
        ADD RAIDAC
        DAC RAIDAC
        SZF I 2
        DAC RITC1+6
        ADD (60)
        SUB COUNT
        SPA
        JMP RIT7
        SUB C61
        CMA"U"STF 5
        DAP RITC1+7
RIT7,   LAC PTR1
        SZF 2
        DAC I RAITEM
        IDX RAITEM
        IDX I ERCOD2
        JSP GO
        I I I RITC1
```

```
            SZF 3
            JMP RIT1
            STF 3
            LAC CPTR2       /RESET D,C, PROGRAMS
            DAC RITC1+4
            LAC CXOR
            DIP RITC1+5
            LAC PTR1        /CHECK IF START OF ITEM
            SMA
            JSP ILLITM      /ILLEGAL ITEM
            XOR B0
            SAD PTR2        /IS IT A DOG-IN-THE-MANGER?
            JSP ILLITM
            DAC PTR1
            LAC I CORPTR    /GET COUNT
            SZF 2
            LAW 7742
            JDA MCHECK
            SUB ONE
            DAC COUNT
            SUB (60)        /ARE WE DONE
            SUB RAIDAC
            SZM
            JMP RIT1

RIT4,       STF 5
RTRN,       SZF 1
            JMP MAXERR
            SZF I 2
            SZF 5
            JMP SWAPE1
ILLITM,     LAW 1000
            JDA 1ERR


,READ,      0
            DAP ,READX
            LAC ,READ
            DAC LTRK
            RAL 8S
            AND (177
            ADD THIRD
            JDA READ
,READX,     JMP .

RITC1,      560101
            200000+,
            630000 PTR1
RITC1+3,    040001
            0
            0               /XOR 1  OR  IOR 1
RITC1+6,    0
            100000 ,
            000000

RAITEM,     IOR .

RAIDAC,     0
```

```
/EXPUNGE INFORMATION
/FLAG 1=1 BLOCK,0 ITEM
/FLAG 2=1 NEGLECT OWN=WRDS(MUST BE HELD)

EPINF:    JSP .FSET
          JSP WHERE       /BUILD UP POINTER
          JDA TRACK
          020000
          JSP WAIT4
          LAC I AC
          DAP CORPTR
          LAC I IO
          DAC DRMPTR
EPINF1,   DAP EXB0+1      /SECOND ENTRY FOR PARAMETER EXPUNGE,ETC,
          DAP EXB1+1
          JSP GO          /READ BLOCK INTO BUFF
          I I I EXB0
          SZF I 1
          JMP EXPITM      /EXPUNGE ITEM
          LAC BUFF
          SAS DRMPTR
          JSP ILLITM
          JSP GO
          I I I EXB1
          LAC DRMPTR      /PUT BIT BACK IN TABLE
          JDA EXP12
          JSP WAIT        /WAIT FOR FINISH OF BLOCK ERASE
          JMP SWAPE1

RWITC2,
EXB0,     560001
          200000+.
          630000 BUFF
          100063
          000000

EXB1,     560001
          260000+.
          630000 ZERO
C50001,   050001
          777777
          100062
          000000
```

```
WHERE,     DAP WHERX
           LAC I (DCH 72)
           SPA
           JMP I 16SPCERR
           SAR 1S
           IAI
           DAC WHER3
           RAL 9S
           AND C777
           ADD (FRELST)
           DAC WHRPTR
           LAC I WHRPTR
           RAL 1S
           SPA
           JMP WHER1
           SZF 2           /FREE=DO WE WANT TO NEGLECT OWNWRDS
           JMP I NOTYOURS  /YES=ILLEGAL
WHER2,     LAC WHER3
WHERX,     JMP .
WHER1,     LAC I WHRPTR
           XOR B1
           AND (340177
           SAD 16USER
           JMP WHER2
           XOR (DCH 200)
           SZF I 2
           SAS I NETHND
           JMP I NOTYOURS
           JMP WHER2       /NETHANDLER QTRACKS ARE ACCESSIBLE
WHRPTR,    0
WHER3,     0

EXPITM,    JSP CKITM
EXPT1,     LAW ILLITM
           DAP COMPX
           CLC             /DON'T COMPARE FIRST BLOCK
           DAC EXPC0 4
           LAC (040001)
           DAC EXPC0 5
           LAC DRMPTR
           DAC PTR1        /SET UP DRA FOR COMPARING
           SZF I 3         /DOG IN THE MANGER REQUESTED?
           JMP EXP5        /NO
           IOR B0
           DAC FEXP
           DAP EXPC4+1
           JSP GO
           I I I EXPC4
           LAC BUFF
           AND (377777)
           DAC DRMPTR
EXP4,      LAC (JSP I PTR1)          /WE SHOULD COMPARE NOW
           DAC EXPC0 4
           LAC (060001)
           DAC EXPC0 5
```

```
EXP5,     LAC DRMPTR
          SAD ONE
          JMP APREXT
          DAC LTRK
          RAL 8S
          AND (177
          ADD THIRD
          JDA REWRITE

EXP2,     LAW BUFF        /SET POINTER TO FIRST OF BUFFER
          DAP EXP3
          CLF 7
EXP9,     LAC DRMPTR
          AND (377777)
          DAC DRMPTR
          SPQ
          JSP ILLITM
          DAP EXPC0+1
EXP.11,   SAD ONE
          JMP EXP6        /THE END
EXP3,     DAC .           /O.K.-STORE IN BUFFER
          XOR LTRK        /DOES IT CHANGE QUARTER TRACK
          AND (376000)
          SZA
          JMP EXP6        /YES
          IDX EXP3
          SAD (DAC BUFF 477)
          JMP EXP7
          JSP GO
          I I I EXPC0
          LAC (060001)
          DAC EXPC0 5
          LAC (JSP I PTR1)
          DAC EXPC0 4
          JMP EXP9

EXP7,     LAW EXP2
          JMP EXP1
```

```
/PUT BLOCKS BACK IN TABLE
/ASSUME ON RIGHT TRACK WITH BITS IN CORE
/BLOCKS IN BUFF
/TERMINATE BY -0

EXP6,     LAW EXP5
EXP1,     DAP EXP1X
          LAW BUFF
          DAP EXP11
          CLC"U"CLF 6
          XCT EXP3

EXP11,    LAC .           /GET A BLOCK ADDRESS
          SPA
EXP1X,    JMP .
          DAP EXPC2+1     /PLACE HEAD AND SECTOR
          JSP GO
          I I I EXPC2
          XCT EXP11
          JDA EXP12       /RESET THE BIT TABLE
          IDX EXP11
          JSP WAIT
          JMP EXP11
```

```
/CHECK REWRITE NUMBER, OWNWORD ETC

CKITM,    DAP CKITMX
          LAC BUFF        /IS IT AN ITEM
          SMA
          JSP ILLITM
CXOR,     XOR B0
          SAD DRMPTR
          JSP ILLITM
          SZF 2
          JMP CKITMX
          LAC BUFF+1
          SAS I OWNWRD
          JMP RTERR
          IDX CORPTR
          LAC I CORPTR
          SAD BUFF+3
CKITMX,   JMP .
RTERR,    LAW 2000
          JDA 1ERR

/PUT THINGS BACK ON BIT TABLE
EXP12,    0
          DAP EXP12X      /AC=17 BIT DRUM POINTER
          LAW 77
          AND EXP12
          ADD BITPTR      /GET POINTER TO WORDS THIS SECTOR
          DAC EXP121
          LAW 1700
          AND EXP12
          SAR 6S          /GET HEAD
          ADD (B0
          DAP EXP124      /GET POINTER TO RIGHT BIT
          LAC I EXP121
EXP124,   XOR .
          DAC I EXP121
          LAC (757777)
          AND I ONTRACK
          DAC I ONTRACK
          LAW I 1         /REST COUNT OF BLOCKS
          ADD I CNTPTR
          DAC I CNTPTR
          SAD (-1776)
          DZM I ONTRACK
EXP12X,   JMP .

EXP121,   0
```

```
B0,         400000
B1,         200000
B2,         100000
B3,         40000
B4,         20000
B5,         10000
B6,         4000
B7,         2000
B8,         1000
C400,       400
C200,       200
C100,       100
C40,        40
C20,        20
C10,        10
FOUR,       4
TWO,        2
ONE,        1
C60001,     60001

EXPC0,      560101
            200000+,
            630000 DRMPTR
EXPC0+3,    040001
            0
            0                   /IOR 1    OR    XOR 1
EXPC0+6,    777777
            100061
            000000

EXPC2,      560001
EXPC2+1,    260000+,
            630000 ZERO
            050001
            777777
            100062
EXPC2+6,    000000

EXPC4,      560001
            260000+,
            630000 FEXP
            040001
            777777
            100062
            000000

FEXP,       0
```

```
/WRITE NON-ADDRESSED INFORMATION
/FLAG 1=1 BLOCK,0 ITEM
/FLAG 2=1 HELD,0 NOT HELD
/FLAG 3=1 AND FLAG 1=1 . WRITE A ZEROED BLOCK
/FLAG 3=1 AND FLAG 1=0 . WRITE OUT THE BIT TABLE (RESET THE FRELST)
/  AND THEN WRITE THE NON-ADDRESSED ITEM

WNAI:   JSP .FSET
        DIO TRACK
        LAC B5
        SZF 2
        LAC B3
        DIP WNAI1
        LAW 1           /BLOCK OR ITEM
        SZF 1
        JMP WNAI2
        LAW 7777        /ITEM=GET WORD COUNT
        AND I (DCH 70)
        DAC WNAI4
        LAW I 1
        ADD I WNAI4
        ADD WNAI4
        AND (770000)
        SZA
        JMP I 16SPCERR
        LAC I WNAI4
        SAS ONE
        SPQ
        JMP I 16SPCERR
        SCR 9S
        SCR 8S
        DIV C61
C61,    61
        SNI I
        IDA
WNAI2,  IOR TRACK
        JDA TRACK
WNAI1,  0
WNAI3,  JSP WAIT4       /WAIT FOR CORE
        LAW 7777
        AND I AC
        IOR SCORE
        DAC WITC0+6
        DAC WNBC+4
        SZF I 1
        JMP WNAI6       /ITEM
        CLC
        SZF 3           /DO WE WANT TO WRITE ZEROS
        DAC WNBC+4      /YES
        LAC BUFF
        DAP WNBC+1      /SET HEAD AND SECTOR
        DAC I IO
        JSP GO          /GO!
        I I I WNBC
        JMP SWAPE1
WNAI4,  0
```

```
WNBC,       560001
            260000+.
            630000 BUFF
            040001
WNBC+4,     0
            100062
            000000

WNAI6,      SZF I 3          /WANT TO WRITE OUT BIT TABLES ?
            JMP WIT          /NO.  NORMAL CASE.
            LAC THIRD
            SUB (FRELST)
            SAR 7S
            LIA
            ADD (TRACKA)
            DAP .+1
            LAC .
            SAS ONTRACK      /ALL HEALTHY ?
            JMP WIT          /STRANGE.
            JSP BTWRIT       /SET UP COMMANDS
             JMP WIT         / STRANGE.
            JSP GO           /WRITE THEM OUT
            I I I NDONC1
            JMP WIT          /ON TO NORMAL WRITE NON-ADR ITEM
```

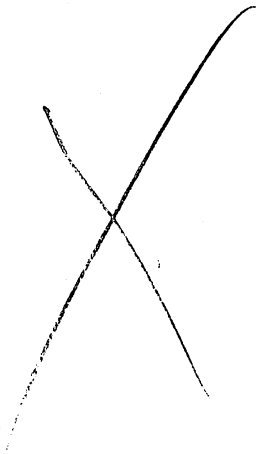
```
WIT,    CLL"U"CML       /FIRST BLOCK OF ITEM
        LAC I OWNWRD
        DAC PTR2
        LAC BUFF        /SAVE DRA FOR BACK PTRS
        DAC PTR3
WIP,    LAW BUFF
        DAP WIP12
        XCT WIP12
        SAD MZERO
        JSP WIP1        /GET MORE BLOCKS
        DAC PTR1        /SET UP FORWARD PTR
        DAC LTRK
        IDX WIP12
        CLF 7
WIP2,   LAC PTR1
        DAP WITC0+1
        XOR LTRK
        AND (376000
        SZA
        JSP WIP3
WIP12,  LAC .
        SPA
        JSP WIP1
        SZL
        IOR B0          /FIX UP FIRST TAG
        DAC PTR1
        IDX WIP12
        JSP GO
        I I I WITC0
        LAC PTR3
        CLL"U"SZL
        DAC PTR2
        SZF 5
        JMP WIP6
        LAW 61
        ADD WITC0+6
        DAP WITC0+6
        JMP WIP2
```

```
WITC0,     560001
WITC0+1,   260000+.
           630000 PTR1
           040001
CPTR2,     630000 PTR2
           040001
WITC0+6,   0
           100061
WITC0+10,  000000
```

```
/MOVE BOOM FOR  WRITE
WIP3,       DAP WIP3X
            JSP WAIT
            LAC PTR1
WIP5,       DAC LTRK
            RAL 8S
            AND (177
            ADD THIRD
            JDA REWRITE
WIP3X1,     LAC PTR1
WIP3X,      JMP .

/GET MORE BLOCKS FOR WRITE

WIP1,       DAP WIP3X
WIP11,      LAC FBLOCK      /ARE WE DONE
            SMA
            JMP WIP4        /DONE
            LAC PTR1
            JDA GBLOCK      /GET MORE BLOCKS
            LAW BUFF
            DAP WIP12
            XCT WIP12
            LIO PTR1        /SET UP SO MOVES BACK TO OLD TRACK BUT SEES NEW POI
            DAC PTR1
            LAI
            JMP WIP5

WIP4,       LAW 1           /TO WRITE LAST BLOCKS
            DAC PTR1
            STF 5
            JMP WIP3X1

WIP6,       LAC PTR3        /SET USER IO
            DAC I IO
APREXT,     JMP SWAPE1

SPPTR1,PTR1,                0
SPPTR2,PTR2,                0
PULPTR,PTR3,                0
DRMPTR,     0
CORPTR,     IOR .
MAXM,       0
COUNT,      0
LTRK,       0
APRFLG,     0
PATFLG,     0
```

```
/REWRITE INFO
/FLAG 1=1␣BLOCK,0␣ITEM
/FLAG 2=1␣DO NOT CHECK REWRITE NO. (MUST BE HELD)
/FLAG 3=1␣APR (IF ITEM)

RWINF:  JSP .FSET       /INITIALIZE AND DECODE THIRD
        JSP WHERE       /FIGURE OUT WHERE DRUM POINTER IS
        JDA TRACK       /MOVE BOOM IN A READ FASHION
        000000
        JSP WAIT4       /WAIT FOR CORE TO SHOW UP
        LAW 7777
        AND I AC
        IOR SCORE
        DAC RWTC0+4
        DAC RWITC0+6    /SET UP DAC COMMANDS
        DAP CORPTR
        LAC I IO
        DAC DRMPTR
        DAP RWTC0+1     /SET UP HEAD AND SECTOR
        DAC PTR1
        DAC PTR3
        DAP RWITC2+1
        AND (1777)
        SAS ONE
        SZA I
        JMP ILLITM
        SZF I 1         /ITEM OR BLOCK
        JMP RWIT        /ITEM
        LAW ILLITM      /BLOCK=SET FOR COMPARE MODE
        DAP COMPX
        LAC (060001)    /ARE WE WRITING OFF HELD OR FREE
        LIO I ONTRACK
        RIL 1S
        SPI
        LAC (040001)
        DAC RWTC0+3
        CLC
        SZF 3
        DAC RWTC0+4
        JSP GO
        I I I RWTC0
        LAC PTR1
        DAC I IO
        JMP SWAPE1

RWTC0,  560101
        220000+.
        630000 PTR1
        0               /IOR 1  OR  XOR 1
RWTC0+4, 0
        100062
        000000
```

```
/DATA CHANNEL COMMANDS FOR REWRTIE ITEM
RWITC0,    560001
           220000+,
           630000 PTR1
           040001
           630000 PTR2
           040001
RWITC0+6,  0
           100061
           000000
```



```
/REWRITE ITEM
/FLAG 3=1 APR ITEM

RWIT,     LAW 7777        /DOES ITEM GO OFF THE END OF CORE?
          AND CORPTR
          ADD I CORPTR
          SUB ONE
          AND (770000)
          SZA
          JSP ILLSPC      /LENGTH TOO LONG
          LAC I CORPTR    /IS IT A LEGAL BLOCK COUNT
          SAD ONE
          JSP ILLSPC      /ILLEGAL ITEM LENGTH
          MUL ONE         /PUT IN DIVIDE FORMAT
          DIV C61
C62,      62
          SNI I
          IDA
          SPQ             /IS IT A LEGAL NUMBER
          JSP ILLSPC      /NO
          CMA
          DAC FBLOCK
          SZF 2
          JMP ARWIT
          JSP GO          /GET BLOCK INTO BUFF=BUFF 62
          I I I RWITC2
          JSP CKITM
          IDX I CORPTR    /UPDATE THE REWRITE NUMBER
ARWIT,    CLL"U"CML"U"SCF
          LAC I OWNWRD
          DAC PTR2
```

```
RWIT1,    LAC PTR1          /SET HEAD AND SECTOR
          AND (377777
          DAC PTR1
          SAD ONE           /IS THIS LAST BLOCK OF PREVIOUS ITEM
          JMP RWIT2         /YES-GO WRITE MORE
          DAP RWITC0 1
          SZF 1
          CLL
          LIO PTR3
          SZL I
          DIO PTR2
          LAW 61
          ADD RWITC0 6
          SZF 1
          DAP RWITC0 6
          STF 1
          LAC PTR1
          XOR LTRK          /DO WE WANT TO CHANGE TRACKS
          AND (376000
          SZA
          JSP WIP3          /YES-GO MOVE THE BOOM
          JSP GO
          I I I RWITC0
          ISP FBLOCK        /ARE WE DONE WITH BLOCKS WANTED
          JMP RWIT1
          LIO RWITC0+6
          LAC RWITC0+1
RWIT4,    DAP WITC0+1       /REWRITE LAST BLOCK WITH PROPER TAG
          DIO WITC0+6
          LAC PTR1
          AND (377777
          SAD ONE
          JMP APREXT        /TAG ALREADY SET PROPERLY
          DAC DRMPTR
          LAW 1
          SZL
          IOR B0
          DAC PTR1
          JSP GO
          30000 WITC0
          LAC (JSP I PTR1)          /MAKE COMPARE STUFF WORK
          DAC EXPC0 4
          LAC (060001)
          DAC EXPC0 5
          LAC PTR3
          DAC PTR1
          LAW EXP2
          JMP WIP3          /GO GET RID OF THE REST OF THE BLOCKS
```

```
/WRITE MORE BLOCKS=FIRST HALF
RWIT2,     LIO RWITC0+6    /GET DAC
           LAC RWITC0+1
RWIT7,     DAP BUFF
           LAC LTRK
           DIP BUFF
           CLC
           DAC BUFF 1
           DIO WITC0+6
           JMP WIP         /GO WRITE AS A WRITE ITEM

/INDEX IOT (SEE CORE 15)
INDEX2,    JSP .FSET
           SZF I 4
           SZF 5
           JMP I NOTYOURS
           LAI
           JDA TRACK
           000000
           JSP WAIT4
           JMP I INDEX1

START XX=JMP
```