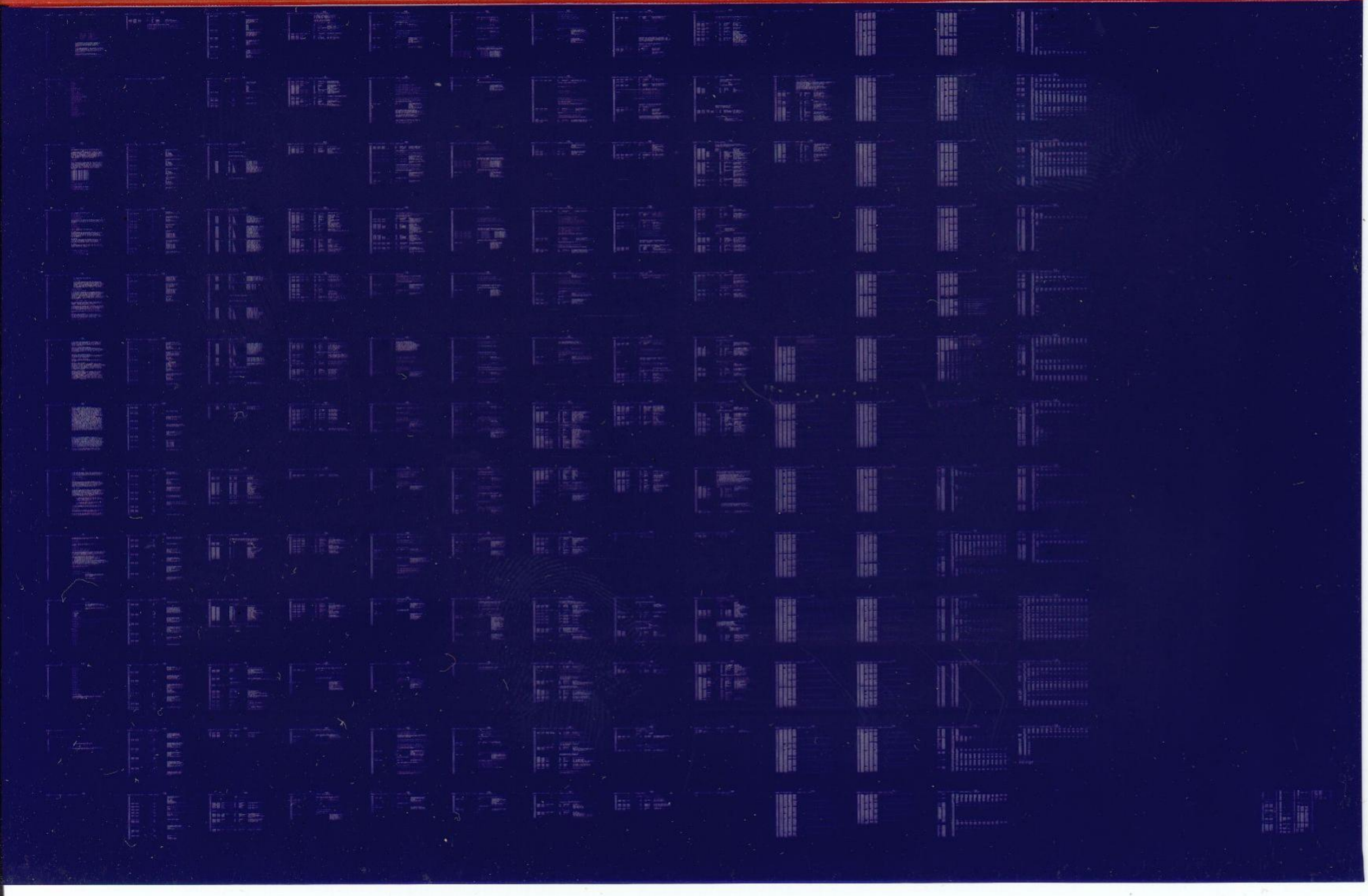
RP04/5/6

FUNCTIONAL CONTROL 2
CZRJJB0

AH-9225B-MC
COPYRIGHT © 74-77
FICHE 1 OF 1

JAN 1978
digital
MADE IN USA



.REM @

IDENTIFICATION

PRODUCT CODE: AC-9223B-MC
PRODUCT NAME: CZRJJB0 RP04/5/6 FUNCTIONAL CONTROLLER TEST PART II
DATE CREATED: DECEMBER 1977
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: PETE BLACKSTONE

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE
WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A
COMMITTMENT BY DIGITAL EQUIPMENT CORPORATION.
DIGITIAL EQUIPMENT CORPORATION ASSUMES NO
RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS
MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED
TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE
COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF
DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH
SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN
WRITING BY DIGITAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO
RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1974,1977 DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT
2. REQUIREMENTS
2.1 EQUIPMENT
2.2 STORAGE
2.3 PRELIMINARY PROGRAMS
3. LOADING PROCEDURE
3.1 METHOD
4. STARTING PROCEDURE
4.1 CONTROL SWITCH SETTINGS
4.2 STARTING ADDRESS OR ADDRESSES
4.3 PROGRAM AND/OR OPERATOR ACTION
5. OPERATING PROCEDURE
5.1 OPERATIONAL SWITCH SETTINGS
5.2 SUB-ROUTINE ABSTRACTS
6. ERRORS
6.1 'FATAL' ERRORS
7. RESTRICTIONS
8. MISCELLANEOUS
8.1 EXECUTION TIME
8.2 STACK POINTER
8.3 OPERATOR SELECTABLE SCOPE LOOPS
8.4 PROGRAM REVISION HISTORY
9. PROGRAM DESCRIPTION

1.0 ABSTRACT

THIS DIAGNOSTIC TESTS THE DCL OF THE RP04/5/6 DISK SUBSYSTEM WHEN CONNECTED TO EITHER AN RH11 OR RH70 CONTROLLER.

IT USES THE DISK SURFACE AND THE DRIVE MECHANICS TO PROVE THE PROPER WORKING OF THE SUBSYSTEM. IT DOES NOT NEED A FORMATTED DISK PACK. A DISK PACK WITH NO VITAL INFORMATION WRITTEN ON IT IS ESSENTIAL. AFTER A SUCCESSFUL RUN (WITH NO ERRORS) OF THIS DIAGNOSTIC IT CAN BE ASSERTED THAT THE DCL IN THE RP04/5/6 SUBSYSTEM WORKS SUCCESSFULLY WHILE STANDING ALONE. SYSTEMS INTERACTION AND DRIVE TIMING IS LEFT TO OTHER DIAGNOSTICS. THIS IS WITH THE ASSUMPTION THAT STATIC 1 (DZRPS AND DZRPT) HAS BEEN RUN SUCCESSFULLY.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11 COMPUTER WITH CONSOLE TELETYPE, AND A RP04/5/6 DISK SYSTEM. THE RP04/5/6 DISK SYSTEM WILL CONSIST OF AN RH11 CONTROLLER, A DISK CONTROL LOGIC (DCL), A DEC 733 DISK DRIVE, AND ITS APPROPRIATE DISK PACK. THE DISK PACK NEED NOT BE FORMATTED. USED SECTION OF THE DISK SURFACE SHALL BE GOOD (HOLE FREE). THE SURFACE FOR THE FOLLOWING SECTORS MUST BE GOOD, THAT IS, FREE OF ANY HOLES OR SURFACE IRREGULARITY BEFORE ANY DATA ERROR CAN BE ATTRIBUTED TO THE LOGIC.

CYLINDER 00, TRACK 00, SECTOR 00
CYLINDER 00, TRACK 00, SECTOR 01
CYLINDER 00, TRACK 18, SECTOR 21
CYLINDER 01, TRACK 00, SECTOR 00
CYLINDER 02, TRACK 00, SECTOR 00
CYLINDER 03, TRACK 00, SECTOR 00
CYLINDER 04, TRACK 00, SECTOR 00
CYLINDER 05, TRACK 00, SECTOR 00
CYLINDER 05, TRACK 07, SECTOR 04
CYLINDER 06, TRACK 00, SECTOR 00
CYLINDER 07, TRACK 00, SECTOR 00
CYLINDER 08, TRACK 00, SECTOR 00
CYLINDER 09, TRACK 18, SECTOR 21
CYLINDER 410, TRACK 18, SECTOR 21

2.2 STORAGE

THIS PROGRAM REQUIRES 16K WORDS OF MEMORY

2.3 PRELIMINARY PROGRAMS

THIS PROGRAM ASSUMES THAT MAINDEC-11-DZRJG-(LATEST REV) HAS BEEN RUN WITHOUT ERRORS.

IT ASSUMES THAT MAINDEC-11-DZRJH-(LATEST REV) HAS BEEN RUN WITHOUT ERRORS.

AND IT ASSUMES THAT MAINDEC-11-DZRJI-(LATEST
REV) HAS BEEN RUN WITHOUT ERRORS.

3.0 LOADING PROCEDURE

USE STANDARD PROCEDURE FOR LOADING .ABS TAPES

4.0 STARTING PROCEDURE

SWITCH 12 MUST BE SET WHEN THIS PROGRAM IS TO BE RUN USING AN
RH70 CONTROLLER. IT CAN BE SET AT THE FRONT PANEL, OR IN THE
SOFTWARE SWITCH REGISTER IF THE OPERATOR SO DERSIRES. SEE
PARAGRAPH 5.1 FOR A DESCRIPTION OF SOFTWARE SWITCH REGISTER
OPERATION.

4.1 CONTROL SWITCH SETTINGS

SEE SECTION 5.1

4.2 STARTING ADDRESS

START AT ADDRESS 200---FOR NORMAL RUN
START AT ADDRESS 210---FOR UNIT SELECTION

200 START
ALL SWITCHES MUST BE DOWN FOR WORST CASE RUN. WITH THIS
STARTING ADDRESS ALL THE RP04/5/6S ON THE SYSTEM WILL BE
TESTED ONE AT A TIME BEFORE "END PASS" IS PRINTED OUT.
TESTING WILL START WITH THE LOWEST UNIT NUMBER DRIVE
THAT IS POWERED UP (THAT IS THE LOWEST UNIT NUMBER RHAS REGISTER
THAT RESPONDS)THEN GO ON TO THE NEXT HIGHER UNIT NUMBER
THAT IS POWERED UP.

204 RESTART
SAME AS 200 START, WITH THE FOLLOWING EXCEPTIONS: THE
PROGRAM WILL INTERROGATE THE OPERATOR FOR THE NON-DEFAULT
C.S.R. AND VECTOR ADDRESS FOR THE RHXX CONTROLLER.
WHEN THESE QUESTIONS HAVE CORRECTLY BEEN ANSWERED, THE
PROGRAM WILL AUTOMATICALLY RESTART FROM ADDRESS 200.

210 START
ALL SWITCHES MUST BE DOWN FOR WORST CASE RUN. WITH THIS
STARTING ADDRESS THE CONSOLE TELETYPE WILL ASK FOR THE UNIT
NUMBER TO BE TESTED. THEN ONLY THAT UNIT WILL BE TESTED
FOR EACH PASS OF THE PROGRAM.

4.3 PROGRAM AND/OR OPERATOR ACTION

1. LOAD THE PROGRAM INTO MEMORY.
2. SET STARTING ADDRESS ON THE SWITCH REGISTER
3. PRESS "LOAD ADDRESS".

4. SET "OPERATIONAL SWITCH SETTINGS" (SEE SECTION 5.1)
   WORST CASE IS ALL SWITCHES DOWN.

5. PRESS "START".

6. FOR THE FIRST PASS EACH TEST WILL BE EXECUTED ONCE
   ON THE DRIVES PRESENT OR DRIVE SELECTED BEFORE "END
   PASS" IS PRINTED. THE FIRST PASS WILL REQUIRE OPERATOR
   INTERVENTION IF THE PROGRAM IS NOT RUN UNDER AN "ACT-11"
   MONITOR. THE SECOND AND SUBSEQUENT PASSES WILL EXECUTE
   EACH TEST FOUR TIMES ON EACH DRIVES PRESENT OR DRIVE
   SELECTED BEFORE "END PASS" IS PRINTED. THE SECOND
   AND SUBSEQUENT PASSED DO NOT NEED ANY OPERATOR INTERVENTION.

5.0 OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

IF THE PROGRAM IS BEING RUN ON A SWITCHLESS PROCESSOR (I. E.
AN 11/34) IT WILL DETERMINE THAT A HARDWARE SWITCH REGISTER
IS NOT PRESENT, AND WILL USE A "SOFTWARE" SWITCH REGISTER.
THE SETTINGS OF THE "SOFTWARE" SWITCHES ARE CONTROLLED
THROUGH A KEYBOARD ROUTINE WHICH IS CALED BY TYPING A
'CONTROL G'. THE PROGRAM WILL RECOGNIZE A 'CONTROL G' AT ANY
TIME EXCEPT WHEN IT IS AT A HIGHER PRIORITY PROCESSING AN
RP04/5/6 INTERRUPT. THE "SOFTWARE" SWITCH VALUES ARE ENTERED AS
AN OCTAL NUMBER IN RESPONSE TO PROMPTING FROM THE SWITCH
ENTRY ROUTINE:

'SWR = NNNNNN   NEW ='

EACH TIME SWITCH SETTINGS ARE ENTERED, THE ENTIRE SWITCH
REGISTER IMAGE MUST BE ENTERED. LEADING ZEROS ARE NOT
REQUIRED. 'RUBOUT' AND 'CONTROL U' FUNCTIONS MAY BE USED TO
CORRECT TYPING ERRORS DURING SWITCH ENTRY.

ON PROCESSORS WITH HARDWARE SWITCH REGISTERS, THE "SOFTWARE"
SWITCH REGISTER MAY ALSO BE USED. IF THE PROGRAM FINDS ALL
16 SWITCHES IN THE 'UP' POSITION WHEN IT IS STARTED, ALL
SWITCH REGISTER REFERENCES WILL BE TO THE "SOFTWARE" REGISTER
AND THE PROCEDURES DESCRIBED ABOVE MUST BE FOLLOWED.

SWITCH DEFINITIONS ARE GIVEN IN SECTION 9 "OPERATIONAL
SWITCH SETTINGS" HOWEVER THE DETAIL DESCRIPTION ARE GIVEN
HERE.

SWITCH 15 - HALT ON ERROR
WHEN THIS SWITCH IS SET, IF THE PROGRAM FINDS AN ERROR
THEN THE APPROPIATE INFORMATION WILL BE PRINTED OUT
AND THEN THE PROGRAM WILL HALT. AFTER THIS HALT, PRESSING
"CONTINUE" WILL CONTINUE WITH THE PROGRAM TILL THE NEXT
ERROR IS FOUND WHEN THE SAME THING WILL HAPPEN.

SWITCH 14 - LOOP ON TEST
WHEN THIS SWITCH IS SET THE PROGRAM WILL BEGIN TO LOOP

ON THE CURRENT TEST BEING EXECUTED. FOR EXAMPLE IF THIS
SWITCH IS SET WHEN THE PROGRAM IS IN TEST 10 THEN THE
PROGRAM WILL KEEP EXECUTING ALL OF TEST 10 REPEATEDLY.
ONE WAY TO BE SURE THAT THE PROGRAM IS IN THE EXPECTED
TEST IS TO SET THIS SWITCH DURING AN ERROR PRINTOUT OR
DURING A PROGRAM HALT.

SWITCH 13 - INHIBIT ERROR TYPEOUTS
WHEN THIS SWITCH IS SET FURTHER ERROR PRINTOUTS WILL
CEASE, HOWEVER OPERATOR INSTRUCTIONS SUCH AS "STOP DRIVE X"
WILL CONTINUE. AT THE END OF PASS "TOTAL NUMBER OF ERRORS
ON THIS PASS ON DRIVE X" WILL BE TRUE, THAT IS, ALTHOUGH
PRINTOUTS WERE INHIBITED IF THAT PASS FOUND 6 ERRORS,
IT WILL SAY SO.

SWITCH 12 - RH70 CONTROLLER SELECT
THIS SWITCH MUST BE SET AT THE START OF THE PROGRAM WHEN THE
DISK DRIVES TO TESTED ARE CONNECTED TO AN RH70 CONTROLLER.
IT MUST NOT BE SET WHEN DISK DRIVES TO BE TESTED ARE
CONNECTED TO AN RH11 CONTROLLER.

SWITCH 11 - INHIBIT ITERATIONS
WHEN THIS SWITCH IS SET THE PROGRAM ON SECOND PASS WILL
NOT REPEAT EACH TEST FOUR TIMES BUT WILL DO EACH TEST
ONCE ONLY.

SWITCH 10 - BELL ON ERROR
WHEN THIS SWITCH IS SET, IF THE PROGRAM FINDS AN ERROR
THE "BELL" OR "ALARM" WILL BE SOUNDED. THIS SWITCH IS USEFUL
WHEN SWITCH 11 IS SET YET INFORMATION IS NEEDED WHEN ANY ERROR
IS DETECTED. TAKE THE EXAMPLE OF A PROGRAM LOOPING ON A TEST WITH
SWITCH 11 SET TO HELP SCOPING. THEN IF THIS SWITCH IS
SET AND THE BELL OR ALARM SOUNDS IT MEANS THAT THE ERROR
IS PRESENT BUT IF THE BELL OR ALARM STOPS IT MEANS THAT
THE ERROR IS NOT PRESENT.

SWITCH 9 - LOOP ON ERROR
WHEN THIS SWITCH IS SET, IF THE PROGRAM FINDS AN ERROR
THEN GENERALLY THE PROGRAM WILL LOOP BACK TO THE LAST
EXECUTED "SCOPE" STATEMENT. IF ON THE SECOND TIME
THROUGH AN ERROR IS FOUND IT WILL AGAIN LOOP BACK TO
THAT "SCOPE" STATEMENT. THIS LOOPING WILL CONTINUE AS LONG
AS THE ERROR IS PRESENT AND THIS SWITCH IS SET. HOWEVER
IF THE ERROR IS NOT PRESENT AT ANY TIME THEN IT WILL
CONTINUE NORMALLY WITH THE PROGRAM. EACH TIME THE ERROR
IS ENCOUNTERED PRINTOUT WILL TAKE PLACE UNLESS SWITCH 11
IS ALSO SET. DURING BEGUG, USING A SCOPE, IT IS RECOMMENDED
THAT SWITCH 11 IS ALSO SET.

NOTE: SEE SECTION 8.3

```
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
```

SWITCH 8 - LOOP ON TEST IN SWR <7:0>
THIS IS A SPECIAL SWITCH. WHEN SET SWITCHES 0 THRU 7
HAVE ONE MEANING AND WHEN RESET SWITCHES 0 THRU 7 HAVE
ANOTHER MEANING. THIS MEANS THAT ANY SETTING OF SWITCH
0 THRU 7 MUST BE DONE WITH SWITCH 8 IN THE APPROPIATE
POSITION. WHEN THIS SWITCH IS SET THEN SWITCHES 0 THRU
7 GIVE THE TEST NUMBER TO BE LOOPED ON. FOR EXAMPLE
WITH SWITCH 8 SET AND SWITCH 3 SET THE PROGRAM WILL LOOP
ON TEST 10. HOWEVER THIS SETTING MUST BE DONE AT THE
BEGINNING OF THE PROGRAM THEN ALL THE TESTS FROM 1 TO 10
WILL BE EXECUTED AND THEN TEST 10 WILL BE REPEATED OVER
AND OVER AGAIN. WHEN THIS SWITCH IS NOT SET THEN SWITCHES
0 THRU 7 HAVE THE MEAING ITS NAME INDICATES.
FOR EXAMPLE SWITCH 7 IS "STOP FURTHER COMPARES: THAT IS
IF SWITCH 8 IS NOT SET AND SWITCH 7 IS SET THEN WHEN A
DATA ERROR IS DETECTED NO FURTHER COMPARES WILL BE DONE.
FOR EXAMPLE IN A 256 WORD BUFFER IF ALL THE WORDS ARE IN
ERROR THEN AFTER SEEING THE PRINTOUT FOR THE FIRST FEW
WORDS SETTING SWITCH 7 ONLY WILL STOP FURTHER PRINTOUTS
OF THIS ERROR AND GO ON WITH THE TEST RATHER THAN PRINT
ALL THE 256 WORDS. HOWEVER IF THIS WAS DONE WITH SWITCH 11
THEN THE NEXT ERROR THAT THE PROGRAM DETECTS IN A SUB-
SEQUENT TEST WILL ALSO BE LOST. BUT WITH SWITCH 7, ONLY
THIS GROUP OF DATA ERRORS ARE NOT PRINTED OUT. ANOTHER
EXAMPLE OF SWITCH 8 BEING LOW IS WITH SWITCH 6, WHICH
IS "ECC TEST-COMPARE END RESULT ONLY". THAT IS IF SWITCH
8 IS NOT SET AND SWITCH 6 IS SET THEN ON ECC TESTS (TEST
120 THRU TEST 134) INSTEAD OF COMPARING CONTENTS OF THE
POSITION REGISTER AND PATTERN REGISTER AFTER EVERY CLOCK,
COMPARES WILL ONLY BE DONE AT THE END OF ALL THE CLOCKS.

NOTE: SEE SECTION 8.3

SWITCH 7 - STOP FURTHER COMPARES IF SW08 IS LOW.
IF SWITCH 8 IS SET AND THIS SWITCH IS ALSO SET THEN THIS
SWITCH GIVES THE TEST NUMBER TO BE LOOPED ON AS INDICATED
IN THE DESCRIPTION OF SWITCH 8. IF SWITCH 8 IS NOT SET
AND THIS SWITCH IS SET THEN THE PROGRAM WILL DO AS THE
NAME INDICATES. FOR EXAMPLE IN A 256 WORD BUFFER IF ALL
THE WORDS ARE IN ERROR THEN AFTER SEEING THE ERROR PRINTOUTS
FOR THE FIRST FEW WORDS THEN SETTING SWITCH 7 WITH SWITCH 8
NOT SET WILL STOP THE PRINTOUT OF ALL 256 WORDS BUT WILL
NOT STOP THE PRINTOUT OF ANOTHER ERROR IN ANY SUBSEQUENT
TEST. IT IS EXPECTED THAT SWITCH 7 AFTER BEING SET FOR
A WHILE TO STOP PRINTING ALL THE 256 WORDS WILL BE RESET
AGAIN TO ENABLE THE PRINTING OF OTHER DATA ERRORS.

SWITCH 6 - TYPE ALL REGISTERS WITH ERROR IF SW08 IS LOW
IF SWITCH 8 IS SET AND THIS SWITCH IS ALSO SET THEN
THIS SWITCH GIVES THE TEST NUMBER TO BE LOOPED ON AS
INDICATED IN THE DESCRIPTION OF SWITCH 8. IF SWITCH 8 IS

NOT SET AND THIS SWITCH IS SET THEN THE PROGRAM WILL DO
AS THE NAME INDICATES. THAT IS ON FINDING AN ERROR INSTEAD
OF ONLY GIVING THE ERROR MESSAGE AND RELEVANT REGISTERS
AS WILL BE DONE IF SWITCH 11 IS NOT SET BUT WILL ALSO
GIVE ALL THE REGISTER CONTENTS (EXCEPT "DATA BUFFER" RHDB).

5.2 SUB-ROUTINE ABSTRACTS

SEE SECTION 9 "SUBROUTINES".

6.0 ERRORS

ERROR PRINTOUTS CONTAIN THE ERROR ADDRESS AND OTHER
PERTINENT INFORMATION CONCERNING THE PARTICULAR FAILURE.
THIS INFORMATION MAY BE THE CONTENTS OF RELEVANT RP04/5/6
REGISTERS OR GOOD/RECEIVED DATA. IF THE ERROR OCCURRED IN A
SUBROUTINE, THE ADDRESS OF THE SUBROUTINE CALL IS ALSO
GIVEN. REFER TO THE PROGRAM LISTING AT THE STATED
ADDRESS TO DETERMINE THE CAUSE OF THE ERROR.

6.1 'FATAL' ERRORS

IN THE EVENT THAT THE DISK DRIVE ECOMES UNAVAILABLE TO THE
CONTROLLER, POWERS DOWN, OR CERTAIN CRITICAL STATUS BITS
CANOT BE CLEARED PRIOR TO THE START OF ATEST SEQUENCE - THIS
INFORMATION WILL BE COMMUNICATED TO THE OPERATOR. IN
ADDITION, THE TTY BELL WILL RING AND THE PROGRAM WILL HALT.
IT IS SUGGESTED THAT IF THIS HAPPENS, THE OPERATOR LOAD
ADDRESS 200 (210) AND RESTART THE PROGRAM AS A FIRST ATTEMPT
TO SOLVE THE PROBLEM. IF THE FILURE CONTINUES TO OCCUR,
THERE ARE TWO OPTIONS FOR THE OPERATOR:

1. LOOK IN THE TEST LISTING FOR THE 'HALT'
INSTRUCTION AND REPLACE IT, PLUS THE TWO WORDS
("TYPE CPHALT") ABOVE WITH 'NOP'S. WITH TTY ERROR
PRINTOUTS INHIBITED, A SCOPE LOOP CAN BE INITIATED
FOR THE TEST IN QUESTION.

2. GO BACK AND RERUN THE DZRPS DIAGNOSTIC AS IT IS
QUITE POSSIBLE THAT A HARD FAILURE HAS OCCURRED IN
ONE OF THE HARDWARE REGISTERS.

IT IS ALSO POSSIBLE TO CONTINUE FROM THE 'HALT' POINT, BUT
THIS IS NOT RECOMMENDED AS ALL FOLLOWING TESTS WILL EXHIBIT
THE SAME SYMPTOMS AND GIVE MISLEADING ERROR PRINTOUTS.

7.0 RESTRICTIONS

BEFORE STARTING THE PROGRAM THE OPERATOR MUST HAVE THE
DRIVE PORT SWITCH LOCKED EITHER ON PORT A OR PORT B BUT
MUST NEVER LEAVE IT IN THE PROGRAMMABLE STATE.

SWITCH 12 MUST BE SET WHEN RUNNING ON AN RH70 CONTROLLER AND
IT MUST NOT BE SET WHEN RUNNING ON AN RH11 CONTROLLER.
BECAUSE OF THE REQUIREMENT FOR IT TO BE SET WHEN USING AN
RH70, THE PROGRAM CANNOT BE RUN IN CHAIN MODE WHEN USING THE

SOFTWARE REGISTER FEATURE WHILE RUNNING ON AN RH70. THIS IS
BECAUSE THE ROUTINE WHICH GETS SOFTWARE SWITCH SETTINGS IS
NOT OPERABLE WHEN IN CHAIN MODE.

8. MISCELLANEOUS

8.1 EXECUTION TIME

THE FIRST PASS OF THE PROGRAM WILL TAKE APPROXIMATELY 20
SECONDS. SUBSEQUENT PASSES WILL TAKE
60 SECONDS .

8.2 STACK POINTER

THE STACK IS INITIALLY SET TO 1000

8.3 OPERATOR SELECTABLE SCOPE LOOPS

HERE IS A DETAILED EXPLAINATION OF HOW THE LOOP ON ERROR WORKS.
FOR INSTRUCTIONS REGARDING THE USAGE OF THIS TECHNIQUE, HIT ↑C
ANY TIME WHILE THE PROGRAM IS RUNNING. ON HITTING AN ERROR
IF THE LOOP ON ERROR SWITCH IS SET, THE PROGRAM GOES BACK -
USUALLY BACK TO THE BEGINNING OF THE TEST.

WHEN THIS OPERATOR SELECTABLE SCOPE LOOP IS USED THEN THE POINT
THE PROGRAM GOES BACK TO CAN BE CHANGED.
THE RESTRICTIONS TO THE POINT WHERE THE PROGRAM CAN GO ARE: -
1. IT MUST BE WITHIN THE TEST UNDER CONSIDERATION
2. LOOP ON ERROR SWITCH MUST BE SET
3. THE ERROR MUST OCCUR WITHIN THE TEST UNDER CONSIDERATION
IF THE ERROR DOES NOT OCCUR WITHIN THE TEST UNDER CONSIDERATION
THE PROGRAM WILL REVERT TO NORMAL OPERATION. HOWEVER, IF LOOP ON
TEST SWITCH IS SET AND THIS OPERATOR SELECTABLE SCOPE LOOP IS USED
THEN THE PROGRAM WILL LOOP BACK TO THE SELECTED POINT WHEN IT
COMES TO THE END OF THE TEST UNDER CONSIDERATION.

AFTER LOOPING FOR SOME TIME IF THE LOOP SWITCH IS PUT DOWN THEN
NORMAL OPERATION WILL CONTINUE.

8.4 PROGRAM REVISION HISTORY

9.0 PROGRAM DESCRIPTION

9.1 LOGIC DIVISION IN HARDWARE MODULES

REGISTER BOARD (RG) - ERROR REGISTER 1 STATUS REGISTERS
MUX FOR REGISTERS GO HANDLING REGISTER
DECODE COMMAND DECODE EXECUTION OF
MECH. COMMANDS

SYNC. DATA BOARD (SN) - DATA CONTROL PARALLEL TO SERIAL
SYNC. BYTE DETECT.

SEEK AND SEARCH (SS) - SEEK LOGIC SEARCH LOGIC HEADER

```
                                                    HANDLING.

                                ERROR CORRECTION (EC) - ECC LOGIC ERROR REGISTER 2 & 3
                                                        MUX FOR ERROR REG. 2 & 3 LOOK AHEAD
                                                        REG. SECTOR COUNTER DATA FORMATION
                                                        RING COUNTER.

                                DUAL PORT (DP) -        DUAL PORT ARBITRATION ATTENTION LOGIC
                                                        SERIAL NO REGISTER MASS BUS REGISTER
                                                        STORAGE

                        9.2     DISK SURFACE USAGE

                                SYMBOLS USED
                                C = CYLINDER
                                T = TRACK
                                S = SECTOR
                                W = WRITE
                                R = READ
                                TT = TEST NUMBER

                                C0, T0, S0
                                TT22-W,R, TT23-R, TT24-W,R, TT25-W,R, TT26-W,R, TT35-W,R, TT37-W, TT50-W, TT51-W

                                C0, T0, S1
                                TT27-W,R, TT37-W,R, TT40-R, TT41-W,R, TT42-W,R, TT43-W,R

                                C0, T18, S21
                                TT30-W, TT31-W,R

                                C1, T0, S0
                                TT30-W,R, TT31-W,R, TT53-W,R, TT54-W,R

                                C1, T18, S21
                                TT31-W

                                C2, T0, S0
                                TT31-W,R

                                C2, T18, S21
                                TT31-W

                                C3, T0, S0
                                TT31-W,R

                                C3, T18, S21
                                TT31-W

                                C4, T0, S0
                                TT31-W,R

                                C4, T18, S21
                                TT31-W

                                C5, T0, S0
                                TT31-W,R
```

```
                                C5, T7, S4
                                TT33-W,R, TT34-W,R

                                C5, T18, S21
                                TT31-W

                                C6, T0, S0
                                TT31-W,R

                                C6, T18, S21
                                TT31-W

                                C7, T0, S0
                                TT31-W,R

                                C7, T18, S18
                                TT31-W

                                C8, T0, S0
                                TT31-W,R

                                C8, T18, S21
                                TT31-W

                                C9, T0, S0
                                TT31-W

                                C9, T18, S21
                                TT31-W, TT32-R

                                C10, T0, S0
                                TT31-W,R

                                C410, T18, S21
                                TT36-W,R, TT50-W,R

                        9.3     THE FOLLOWING SECTION DESCRIBES EACH TEST AND SUBROUTINES
                                IN DETAIL AND CAN BE USED AS AN INDEX TO THE LISTING.
                                THE LEFT MOST COLUMN IS THE LINE NUMBER WITHIN THE LISTING
                                WHERE THAT ITEM WILL BE FOUND.
                                @
```

```
;DRIVE MUST BE LOCKED ON PORT A OR PORT B


                                        ;*INTERNAL PROGRAM MACROS BEGIN HERE
                                        ;**************************************


                                ;*
                                ;*NOTE:  ALL MACRO CALLS BEGINNING WITH ".$" ARE SUPPLIED FROM AN
                                ;*       EXTERNAL SYSMAC.SML PACKAGE WHICH MUST BE MADE AVAILABLE
                                ;*       TO THE SOURCE PROGRAM AT ASSEMBLY TIME.
                                ;*
```

```
620
```

```
                                .SBTTL  STARTING ADDRESS

                        000200          .=200
000200  000137  004710  RA:     JMP     @#BEGIN         ;NORMAL START
000204  000137  035232  ADDMOD: JMP     @#BASECH        ;GET DEVICE PARAMETERS
000210  000137  004674          JMP     @#BEGIN2        ;JUMP TO SELECT DRIVE START
                        000220          .=220
000220  000137  004660          JMP     @#BEGIN1        ;JUMP TO NO OPERATOR TESTS START

                                ;*STARTING ADDRESS 200 FOR NORMAL STARTS
                                ;*THIS WILL TEST ALL RP04'S ON THE SYSTEM A SINGLE DRIVE AT A TIME
                                ;*
                                ;*STARTING ADDRESS 210 WILL TEST ONLY ONE SPECIFIED DRIVE
                                ;*
                                ;*STARTING ADDRESS 220 WILL JUMP OVER THE TESTS REQUIRING AN OPERATOR
                                ;*AT THE DRIVE
```

```
640
641          001110                          .=1110                  ; ?
```

```
                                ;ITEM1
001226  042566                          EM1             ;RP04 DID NOT INTERRUPT
                                                        ;WAITED ON BIT DID NOT OCCUR
001230  057444                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;REG CONTENTS
                                                        ;RHCS1 CONTENTS
001232  061714                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001234  062234                          DF1             ;0,0,0,0,0,0

                                ;ITEM2
001236  042615                          EM2             ;INTERRUPT ENABLE BIT DOWN BUT
                                                        ;WAITED ON BIT DID NOT OCCUR
001240  057444                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;REG CONTENTS
                                                        ;RHCS1 CONTENTS
001242  061714                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001244  062234                          DF1             ;0,0,0,0,0,0

                                ;ITEM3
001246  042704                          EM3             ;RP04 DID NOT INTERRUPT WHEN
                                                        ;WAITED ON BIT DID SET
001250  057444                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;RHCS1 CONTENTS
001252  061714                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001254  062234                          DF1             ;0,0,0,0,0,0

                                ;ITEM4
001256  042765                          EM4             ;WAITED ON BIT DID SET BUT
                                                        ;TIME IS IN ERROR
                                                        ;TIME IS GIVEN IN 10 MICRO SEC.
                                                        ;(DECIMAL)
001260  057624                          DH4             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;TIME IN 10 MSEC
001262  061734                          DT4             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,WAITIM
001264  062243                          DF4             ;0,0,0,0,0,1

                                ;ITEM5
001266  043076                          EM5             ;RHAS DOES NOT CLEAR BY
                                                        ;MOVING IN ALL ONES
001270  057765                          DH5             ;PC
                                                        ;REG. ADDR.
```

```
                                                                ;GOOD DATA
                                                                ;RECEIVED DATA
001272  061756                        DT5                     ;$ERRPC,REGADR,$GDDAT,$BDDAT
001274  062252                        DF5                     ;0,0,0,0

                            ;ITEM6
001276  043150                        EM6                     ;LOADING RHER1 FOR ALL
                                                                ;UNITS DID NOT SET ANY BITS
                                                                ;IN RHAS-NO UNITS PRESENT
001300  060104                        DH6                     ;PC
                                                                ;REG ADDR
                                                                ;RECEIVED DATA
001302  061772                        DT6                     ;$ERRPC,REGADR,$BDDAT
001304  062257                        DF6                     ;0,0,0

                            ;ITEM7
001306  043236                        EM7                     ;SPECIFIED REGISTER NONEXISTANT
                                                                ;SO ABORT PROGRAM
001310  060203                        DH7                     ;PC
                                                                ;ADDR. OF REG.
001312  062004                        DT7                     ;$ERRPC,TEMP1
001314  062263                        DF7                     ;0,0

                            ;ITEM10
001316  043306                        EM10                    ;STOPED DRIVE HAS MOL BIT
                                                                ;IN RHDS1 = 1
001320  060243                        DH10                    ;PC
                                                                ;TEST NO
                                                                ;FAILING REG ADDR
                                                                ;CONTENTS OF RHCS1
                                                                ;CONTENTS OF RHCS2
                                                                ;CONTENTS OF RHDS1
                                                                ;CONTENTS OF RHER1
001322  062014                        DT10                    ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
001324  062266                        DF10                    ;0,0,0,0,0,0,0

                            ;ITEM11
001326  043355                        EM11                    ;WITH SPINDLE POWERED DOWN
                                                                ;RHCS2 SHOULD HAVE ONLY
                                                                ;UNIT NUMBER AND IR HIGH
001330  060243                        DH10                    ;PC
                                                                ;TEST NO
                                                                ;FAILING REG. ADR
                                                                ;CONTENTS OF RHCS1
                                                                ;CONTENTS OF RHCS2
                                                                ;CONTENTS OF RHDS1
                                                                ;CONTENTS OF RHER1
001332  062014                        DT10                    ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
001334  062266                        DF10                    ;0,0,0,0,0,0,0

                            ;ITEM12
001336  043462                        EM12                    ;AFTER A POWER UP WITH
                                                                ;NO PACK ACKNOWLEDGE COMMAND
                                                                ;RHDS1 SHOULD HAVE MOL=1, VV=0
001340  060243                        DH10                    ;PC
                                                                ;TEST NO
```

```
                                                                ;FAILING REGISTER ADDR.
                                                                ;CONTENTS OF RHCS1
                                                                ;CONTENTS OF RHCS2
                                                                ;CONTETNS OF RHDS1
                                                                ;CONTENTS OF RHER1
001342  062014                          DT10                  ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
001344  062266                          DF10                  ;0,0,0,0,0,0,0

                                ;ITEM13
001346  043570                          EM13                  ;AFTER A POWER UP WITHOUT
                                                              ;ANY INIT RHCS1 SHOULD
                                                              ;HAVE GO=0, DVA=1, RDY=1
                                                              ;IE=0, DISREGARD
                                                              ;ALL OTHER BITS
001350  060243                          DH10                 ;PC
                                                              ;TEST NO
                                                              ;FAILING REGISTER ADDR.
                                                              ;CONTENTS OF RHCS1
                                                              ;CONTENTS OF RHCS2
                                                              ;CONTENTS OF RHDS1
                                                              ;CONTENTS OF RHER1
001352  062014                          DT10                  ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
001354  062266                          DF10                  ;0,0,0,0,0,0,0

                                ;ITEM14
001356  043707                          EM14                  ;AFTER POWER UP RHCC
                                                              ;SHOULD BE=0
001360  057765                          DH5                   ;PC
                                                              ;REG. ADDR.
                                                              ;GOOD DATA
                                                              ;RECEIVED DATA
001362  061756                          DT5                   ;$ERRPC,REGADR,$GDDAT,$BDDAT
001364  062252                          DF5                   ;0,0,0,0
```

```
                                ;ITEM15
001366  043761                          EM15            ;PACK ACKNOWLEDGE CAUSED
                                                        ;AN ERROR
                                                        ;GOOD DATA IS BEFORE COMMAND
                                                        ;RECEIVED DATA IS AFTER COMMAND
001370  057765                          DH5             ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001372  061756                          DT5             ;$ERRPC,REGADR,$GDDAT,$BDDAT
001374  062252                          DF5             ;0,0,0,0

                                ;ITEM16
001376  044122                          EM16            ;GIVING A NO-OP COMMAND CAUSED
                                                        ;AN ERROR
                                                        ;GOOD DATA GIVES REGISTER
                                                        ;CONTENTS BEFORE COMMAND
                                                        ;RECEIVED DATA GIVES REGISTER
                                                        ;CONTENTS AFTER COMMAND
001400  057765                          DH5             ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001402  061756                          DT5             ;$ERRPC,REGADR,$GDDAT,$BDDAT
001404  062252                          DF5             ;0,0,0,0

                                ;ITEM17
001406  044250                          EM17            ;DRIVE CLEAR COMMAND
                                                        ;CAUSED AN ERROR
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS
                                                        ;THERE AFTER COMMAND
001410  057765                          DH5             ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001412  061756                          DT5             ;$ERRPC,REGADR,$GDDAT,$BDDAT
001414  062252                          DF5             ;0,0,0,0

                                ;ITEM20
001416  044405                          EM20            ;READ-IN COMMAND GAVE AN ERROR
                                                        ;GOOD DATA HAS WHAT SHOULD BE THERE
                                                        ;RECEIVED DATA HAS WHAT WAS
                                                        ;AFTER COMMAND
001420  057765                          DH5             ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001422  061756                          DT5             ;$ERRPC,REGADR,$GDDAT,$BDDAT
001424  062252                          DF5             ;0,0,0,0

                                ;ITEM 21
001426  044554                          EM21            ;RHCS1 CONTENTS DURING
                                                        ;COMMAND WAS IN ERROR
```

```
   844  001430  057765                          DH5
   845  001432  061756                          DT5
   846  001434  062252                          DF5
   847
   848                                  ;ITEM 22
   849  001436  044627                          EM22            ;RHDS1 CONTENTS DURING
   850                                                          ;COMM ANS WAS IN ERROR
   851  001440  057765                          DH5
   852  001442  061756                          DT5
   853  001444  062252                          DF5
   854
   855                                  ;ITEM 23
   856  001446  044702                          EM23            ;UNLOAD COMMAND GAVE AN ERROR
   857                                                          ;GOOD DATA GIVES WHAT SHOULD
   858                                                          ;BE THERE
   859                                                          ;RECEIVED DATA GIVES WHAT WAS
   860                                                          ;THERE AFTER COMMAND
   861  001450  057765                          DH5
   862  001452  061756                          DT5
   863  001454  062252                          DF5
   864
   865                                  ;ITEM 24
   866  001456  045051                          EM24            ;OFFSET COMMAND CAUSED AN ERROR
   867                                                          ;GOOD DATA IS WHAT SHOULD BE THERE
   868                                                          ;RECEIVED DATA GIVES WHAT WAS THERE
   869                                                          ;AFTER AN OFFSET COMMAND
   870  001460  057765                          DH5
   871  001462  061756                          DT5
   872  001464  062252                          DF5
   873
   874                                  ;ITEM 25
   875  001466  045214                          EM25            ;RETURN TO CENTER LINE COMMAND
   876                                                          ;CAUSED AN ERROR
   877                                                          ;GOOD DATA GIVES WHAT SHOULD BE
   878                                                          ;THERE
   879                                                          ;RECEIVED DATA GIVES WHAT WAS
   880                                                          ;THERE AFTER COMMAND
   881  001470  057765                          DH5
   882  001472  061756                          DT5
   883  001474  062252                          DF5
   884
   885                                  ;ITEM 26
   886  001476  045376                          EM26            ;500 OFFSETS CAUSED AN ERROR
   887  001500  060422                          DH26            ;PC
   888                                                          ;CONT. OF RHCS1
   889                                                          ;CONT. OF RHCS2
   890                                                          ;CONT. OF RHDS1
   891                                                          ;CONT. OF RHER1
   892                                                          ;CONT. OF RHER2
   893                                                          ;CONT. OF RHER3
   894  001502  062034                          DT26            ;$ERRPC,CS1,CS2,DS1,ER1,ER2,ER3
   895  001504  062275                          DF26            ;0,0,0,0,0,0,0
   896
   897                                  ;ITEM 27
   898  001506  045466                          EM27            ;WRITE HEADER AND DATA
   899                                                          ;CAUSED IMPROPER REGISTER CHANGE
```

```
                                                        ;GOOD DATA GIVES WHAT
                                                        ;SHOULD BE THERE
                                                        ;RECEIVED DATA GIVES WHAT
                                                        ;WAS THERE AFTER COMMAND
001510  057765                      DH5
001512  061756                      DT5
001514  062252                      DF5

                            ;ITEM 30
001516  045704                      EM30                  ;WRITE HEADER AND DATA
                                                        ;CHANGED WRITE FROM BUFFER
001520  060622                      DH30                  ;PC
                                                        ;WORD NO
                                                        ;GOOD DATA
                                                        ;BAD DATA
001522  062056                      DT30                  ;$ERRPC,ERWORD,$GDDAT,$BDDAT
001524  062305                      DF30                  ;0,0,0,0

                            ;ITEM 31
001526  045764                      EM31                  ;READ HEADER AND DATA CAUSED
                                                        ;IMPROPER REGISTER CHANGE
                                                        ;GOOD DATA HAS WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT
                                                        ;WAS THERE AFTER COMMAND

001530  057765                      DH5
001532  061756                      DT5
001534  062252                      DF5

                            ;ITEM 32
001536  046201                      EM32                  ;WRITE HEADER AND DATA FOLLOWED
                                                        ;BY A READ HEADER AND DATA
                                                        ;CAUSED A READ/WRITE ERROR
001540  060622                      DH30
001542  062056                      DT30
001544  062305                      DF30

                            ;ITEM 33
001546  046306                      EM33                  ;READ DATA CAUSED IMPROPER REGISTER
                                                        ;CHANGE
                                                        ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS THERE AFTER
                                                        ;COMMAND

001550  057765                      DH5
001552  061756                      DT5
001554  062252                      DF5

                            ;ITEM 34
001556  046510                      EM34                  ;READ DATA INCORRECT
001560  060622                      DH30
001562  062056                      DT30
001564  062305                      DF30

                            ;ITEM 35
001566  046534                      EM35                  ;WRITE DATA COMMAND CAUSED
                                                        ;IMPROPER REGISTER CHANGE
```

```
                                                        ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                        ;RECEIVED DATA GIVES REGISTER
                                                        ;CONTENTS AFTER WRITE DATA
001570  057765                          DH5
001572  061756                          DT5
001574  062252                          DF5

                        ;ITEM 36
001576  046752                  EM36    ;WRITE DATA COMMAND CHANGED
                                        ;WRITE FROM BUFFER
001600  060622                          DH30
001602  062056                          DT30
001604  062305                          DF30

                        ;ITEM 37
001606  047027                  EM37    ;SEEK COMMAND CAUSED AN
                                        ;ERROR
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT
                                        ;WAS THERE AFTER SEEK COMMAND
001610  057765                          DH5                     ;
001612  061756                          DT5                     ;
001614  062252                          DF5                     ;

                        ;ITEM 40
001616  047244                  EM40    ;WRITE CHECK CAUSED AN
                                        ;IMPROPER REGISTER CHANGE
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT WAS
                                        ;THERE AFTER COMMAND
001620  057765                          DH5
001622  061756                          DT5
001624  062252                          DF5

                        ;ITEM 41
001626  047453                  EM41    ;LOCKING OUT WRITES BY WRITE
                                        ;LOCK BUTTON CAUSED IMPROPER
                                        ;REGISTER CHANGE
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT
                                        ;WAS THERE AFTER WRITES
                                        ;WERE LOCKED OUT BY
                                        ;BUTTON
001630  057765                          DH5
001632  061756                          DT5
001634  062252                          DF5

                        ;ITEM 42
001636  047734                  EM42    ;ATTEMPTING TO WRITE WITH WRITE
                                        ;LOCKED OUT CAUSED IMPROPER
                                        ;REGISTER CHANGE
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
```

```
                                                                ;RECEIVED DATA GIVES WHAT WAS
                                                                ;THERE AFTER ATTEMPT
001640  057765                          DH5
001642  061756                          DT5
001644  062252                          DF5

                        ;ITEM 43
001646  050212                  EM43    ;WRITING WITH WRITE LOCKED
                                        ;OUT CHANGED DISK DATA
                                        ;GOOD DATA GIVES WHAT WAS
                                        ;ON DISK BEFORE WRITE WITH
                                        ;WRITE LOCK WAS ATTEMPTED
                                        ;RECEIVED DATA GIVES WHAT WAS
                                        ;READ BACK AFTER WRITE WITH
                                        ;WRITE LOCK WAS ATTEMPTED
001650  060622                          DH30
001652  062056                          DT30
001654  062305                          DF30

                        ;ITEM 44
001656  050550                  EM44    ;ENABLING WRITES BY WRITE LOCK
                                        ;BUTTON CAUSED AN ERROR
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT WAS
                                        ;THERE AFTER WRITE LOCK
                                        ;BUTTON ENABLED WRITES
001660  057765                          DH5                     ;
001662  061756                          DT5                     ;
001664  062252                          DF5                     ;

                        ;ITEM 45
001666  051042                  EM45    ;TRANSFERRING ON LAST BLOCK IE. CYLINDER
                                        ;410, SECTOR 21, TRACK 18
                                        ;CAUSED IMPROPER REGISTER
                                        ;CHANGE
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT WAS
                                        ;THERE AFTER TRANSFER
001670  057765                          DH5
001672  061756                          DT5
001674  062252                          DF5

                        ;ITEM 46
001676  051350                  EM46    ;DATA READ FROM LAST
                                        ;BLOCK IE. CYLINDER 410
                                        ;SECTOR 21, TRACK 18 IS IN
                                        ;ERROR
001700  060622                          DH30
001702  062056                          DT30
001704  062305                          DF30

                        ;ITEM 47
001706  051474                  EM47    ;TRANSFERRING FROM NONEXISTANT
                                        ;SECTOR CAUSED IMPROPER
```

```
                                                                ;REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS
                                                                ;THERE AFTER ATTEMPTED
                                                                ;TRANSFER
001710  057765                          DH5
001712  061756                          DT5
001714  062252                          DF5

                        ;ITEM 50
001716  051756                          EM50                    ;TRANSFERRING FROM NONEXISTANT
                                                                ;SECTOR CAUSED DATA ERROR
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE IN BUFFER
                                                                ;RECEIVED DATA GIVES WHAT WAS
                                                                ;IN BUFFER AFTER TRANSFER
001720  060622                          DH30
001722  062056                          DT30
001724  062305                          DF30

                        ;ITEM 51
001726  052175                          EM51                    ;GIVING ILLEGAL FUNCTION CAUSED
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD BE
                                                                ;THERE
                                                                ;RECEIVED DATA GIVES REGISTER
                                                                ;CONTENTS AFTER ILLEGAL FUNCTION
001730  060734                          DH51                    ;PC
                                                                ;REG. ADDR.
                                                                ;GOOD DATA
                                                                ;RECEIVED DATA
                                                                ;ILLEGAL FUNCTION
001732  062072                          DT51                    ;$ERRPC,REGADR,$GDDAT,$BDDAT,ILLEGL
001734  062312                          DF51                    ;0,0,0,0,0

                        ;ITEM 52
001736  052442                          EM52                    ;WRITE DATA ON NONEXISTANT
                                                                ;SECTOR CAUSED IMPROPER
                                                                ;REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER ATTEMPTED
                                                                ;WRITE DATA
001740  057765                          DH5
001742  061756                          DT5
001744  062252                          DF5

                        ;ITEM 53
001746  052713                          EM53                    ;READ HEADER AND DATA AFTER
                                                                ;A SEARCH CAUSED AN ERROR
001750  060622                          DH30
001752  062056                          DT30
001754  062305                          DF30
```

```
                                ;ITEM 54
001756  053001                          EM54          ;ATTEMPTED OPERATION WITH
                                                      ;INVALID ADDRESS CAUSED
                                                      ;IMPROPER REGISTER CHANGE
                                                      ;GOOD DATA GIVES WHAT SHOULD
                                                      ;BE THERE
                                                      ;RECEIVED DATA GIVES WHAT WAS
                                                      ;THERE AFTER OPERATION
001760  057765                            DH5
001762  061756                            DT5
001764  062252                            DF5

                                ;ITEM 55
001766  053246                          EM55          ;WRITING/READING WITH EXPECTED
                                                      ;ADDRESS OVERFLOW ERROR CAUSED
                                                      ;IMPROPER REGISTER CHANGE
                                                      ;GOOD DATA GIVES WHAT SHOULD
                                                      ;BE THERE
                                                      ;RECEIVED DATA GIVES WHAT
                                                      ;WAS THERE AFTER OPERATION
001770  057765                            DH5
001772  061756                            DT5
001774  062252                            DF5

                                ;ITEM 56
001776  053534                          EM56          ;DATA READ WITH AN EXPECTED
                                                      ;ADDRESS OVERFLOW ERROR IS
                                                      ;INCORRECT
                                                      ;WORD NO 1 TO 260 SHOULD
                                                      ;BE READ
                                                      ;WORD NOS 261 TO 266 SHOULD
                                                      ;NOT CHANGE DUE TO READ
002000  060622                            DH30
002002  062056                            DT30
002004  062305                            DF30

                                ;ITEM 57
002006  053744                          EM57          ;ATTEMPTING DATA COMMAND
                                                      ;WITH WRONG FORMAT BIT CAUSED
                                                      ;IMPROPER REGISTER CHANGE
                                                      ;GOOD DATA GIVES WHAT SHOULD BE
                                                      ;THERE
                                                      ;RECEIVED DATA GIVES WHAT WAS
                                                      ;THERE AFTER ATTEMPTED DATA
                                                      ;TRANSFER
002010  057765                            DH5
002012  061756                            DT5
002014  062252                            DF5

                                ;ITEM 60
002016  054236                          EM60          ;ATTEMPTING TO MODIFY REGISTER
                                                      ;DURING AN OPERATION CAUSED
                                                      ;IMPROPER REGISTER CHANGE
                                                      ;GOOD DATA GIVES WHAT SHOULD
                                                      ;BE THERE
```

```
1180
1181
1182
1183  002020  061073                        DH60            ;RECEIVED DATA GIVES WHAT WAS
1184                                                        ;THERE AFTER OPERATION
1185                                                        ;WAS COMPLETE
1186                                                        ;PC
1187                                                        ;REG. ADDR.
1188  002022  062110                        DT60            ;GOOD DATA
1189  002024  062320                        DF60            ;RECEIVED DATA
1190                                                        ;MODFING REGISTER
1191                                ;ITEM 61                ;$ERRPC,REGADR,$GDDAT,$BDDAT,$BDADR
1192  002026  054645                        EM61            ;0,0,0,0,0
1193  002030  061230                        DH61
1194                                                        ;DEVICE NOT AVAIBLE BEFOR COMMAND WAS TO BE GIVEN
1195                                                        ;PC
1196  002032  062126                        DT61            ;PC OF JSR
1197  002034  062326                        DF61            ;RHDS1
1198                                                        ;$ERRPC,PCJSR,$BDADR
1199                                ;ITEM 62                ;0,0,0
1200  002036  054645                        EM61
1201  002040  061322                        DH62
1202                                                        ;DEVICE NOT AVAIBLE BEFOR COMMAND WAS TO BE GIVEN
1203                                                        ;PC
1204  002042  062140                        DT62            ;PC OF JSR
1205  002044  062332                        DF62            ;RHCS1 WAS
1206                                                        ;$ERRPC,PCJSR,$BDADR
1207                                                        ;0,0,0
1208                                ;ITEM 63
1209  002046  054731                        EM63            ;RHDS1 CONTENTS DURING
1210                                                        ;COMMAND WAS IN ERROR
1211  002050  057765                        DH5
1212  002052  061756                        DT5
1213  002054  062252                        DF5
1214
1215
1216                                ;ITEM 64
1217  002056  055004                        EM64            ;RECALIBRATE COMMAND CAUSED
1218                                                        ;IMPROPER REGISTER CHANGE.
1219                                                        ;GOOD DATA GIVES WHAT SHOULD BE
1220                                                        ;THERE.
1221                                                        ;RECEIVED DATA GIVES WHAT WAS THERE
1222                                                        ;AFTER COMMAND
1223  002060  057765                        DH5
1224  002062  061756                        DT5
1225  002064  062252                        DF5
1226
1227
1228                                ;ITEM65
1229
1230  002066  055223                        EM65            ;INTERRUPT FAILING
1231  002070  061375                        DH65            ;PC
1232                                                        ;TEST NO
1233                                                        ;CONTENTS OF RHCS1
1234                                                        ;CONTENTS OF RHAS
1235                                                        ;CONTENTS OF RHDS1
```

```
1236  002072  062152                    DT65          ;$ERRPC,TSTNM,CS1,AS,DS1
1237  002074  062336                    DF65          ;0,0,0,0,0
1238
1239
1240                              ;ITEM66
1241  002076  055245                    EM66          ;HEADER AND DATA COMMAND
1242                                                   ;FOR HEAD SELECTION TEST
1243                                                   ;CAUSED AN ERROR
1244                                                   ;RHDST GIVES WHAT TRACK
1245                                                   ;WAS BEING WRITTEN ON CYLINDER 0
1246                                                   ;SECTOR 0
1247  002100  061511                    DH66          ;PC
1248                                                   ;RHDST
1249                                                   ;RHER1
1250                                                   ;RHER2
1251                                                   ;RHER3
1252                                                   ;RHCS1
1253                                                   ;RHCS2
1254  002102  062166                    DT66          ;$ERRPC,DST,ER1,ER2,ER3,CS1,CS2
1255  002104  062343                    DF66          ;0,0,0,0,0,0,0
1256                              ;ITEM67
1257  002106  055456                    EM67          ;READ HEADER AND DATA ERROR
1258                                                   ;IN HEAD SELECTION TEST
1259                                                   ;FIRST FOUR WORDS GIVE HEADER
1260                                                   ;NEXT WORDS ARE DATA
1261                                                   ;GOOD DATA WORDS GIVE
1262                                                   ;THE TRACK NUMBER IN
1263                                                   ;BITS 4,5,6,7,8
1264  002110  060622                    DH30
1265  002112  062056                    DT30
1266  002114  062305                    DF30
1267                              ;ITEM70
1268  002116  055732                    EM70          ;READ HEADER AND DATA ERROR
1269                                                   ;IN DIFFERENCE LINE TEST
1270                                                   ;WORD NOS. 1-4 GIVE
1271                                                   ;HEADER
1272                                                   ;WORD NOS. 5-260 GIVE DATA
1273                                                   ;WHICH IS THE CYLINDER
1274                                                   ;ADDRESS
1275  002120  060622                    DH30
1276  002122  062056                    DT30
1277  002124  062305                    DF30
1278
1279                              ;ITEM 71
1280  002126  056140                    EM71          ;FORCING OPI CAUSED IMPROPER REGISTER
1281                                                   ;  CHANGE
1282                                                   ;GOOD DATA GIVES WHAT SHOULD
1283                                                   ;BE THERE
1284                                                   ;RECEIVED DATA GIVES WHAT WAS
1285                                                   ;THERE AFTER 3 INDEX PULSES
1286  002130  057765                    DH5           ;PC
1287                                                   ;REG. ADDR.
1288                                                   ;GOOD DATA
1289                                                   ;RECEIVED DATA
1290  002132  061756                    DT5           ;$ERRPC,REGADR,$GDDAT,$BDDAT
1291  002134  062252                    DF5           ;0,0,0,0
```

```
                                ;ITEM72
002136  056401                          EM72            ;THERE WAS AN ERROR
                                                        ;AFTER A WRITE HEADER
                                                        ;AND DATA COMMAND

002140  061607                          DH72            ;PC
                                                        ;RHCS1
                                                        ;RHCS2
                                                        ;RHDS1
                                                        ;RHDST
                                                        ;RHCA
                                                        ;RHER1
                                                        ;RHWC
002142  062210                          DT72            ;$ERRPC,CS1,CS2,DS1,DST,CA,ER1,WC
002144  062354                          DF72            ;0,0,0,0,0,0,0,0

                                ;ITEM73
002146  056647                          EM73            ;READING OVER 3 INDEX
                                                        ;PULSES CAUSED SC
002150  061607                          DH72
002152  062210                          DT72
002154  062354                          DF72

                                ;ITEM74
002156  057017                          EM74            ;READING OVER 3 INDEX
                                                        ;PULSES CAUSED OPI
002160  061607                          DH72
002162  062210                          DT72
002164  062354                          DF72
```

```
                                ;RH11 REGISTER BITS


002166  000254                  RPVEC:  254                     ;RP04 VECTOR ADDRESS


                                ;WORD COUNT REGISTER (RHWC)
                                ;EACH BIT IS CALLED BY BIT NUMBER


                                ;BUS ADDRESS REGISTER (RHBA)
                                ;EACH BIT IS CALLED BY BIT NUMBER


                                ;CONTROL AND STATUS REGISTER 2 (RHCS2)

000001                          US1=    1               ;UNIT SELECT (BIT #0)
000002                          US2=    2               ;UNIT SELECT (BIT #1)
000004                          US4=    4               ;UNIT SELECT (BIT #2)
000010                          BAI=    10              ;BUS ADDRESS INCREMENT INHIBIT (BIT #3)
000020                          UNIB=   20              ;UNIBUS B DC LO (BIT #4)
000040                          CLR=    40              ;CLEAR (BIT #5)
000100                          IR=     100             ;INPUT READY (BIT #6)
000200                          OR=     200             ;OUTPUT READY (BIT #7)
000400                          MPE=    400             ;MASS BUS PARITY ERROR (BIT #8)
001000                          MXF=    1000            ;MISSED TRANSFER ERROR (BIT #9)
002000                          PGE=    2000            ;PROGRAM ERROR (BIT #10)
004000                          NEM=    4000            ;NON EXISTANT MEMORY (BIT #11)
010000                          NED=    10000           ;NON EXISTANT DRIVE (BIT #12)
020000                          UPE=    20000           ;UNIBUS PARITY ERROR (BIT #13)
040000                          WCE=    40000           ;WRITE CHECK ERROR (BIT #14)
100000                          DLT=    100000          ;DATA LATE (BIT #15)


                                ;DATA BUFFER REGISTER (RHDB)
                                ;EACH BIT IS CALLED BY BIT NUMBER
```

```
1372                                ;RP04 REGISTER BITS
1373
1374
1375
1376                                ;CONTROL AND STATUS 1 REGISTER. (#00)
1377
1378        000001                  GO=     1                       ;GO (BIT #0)
1379        000100                  IE=     100                     ;INTERRUPT ENABLE (BIT #6)
1380        000200                  RDY=    200                     ;READY (BIT #7)
1381        000400                  A16=    400                     ;HIGH ORDER UNIBUS BITS (BIT #8)
1382        001000                  A17=    1000                    ;HIGH ORDER UNIBUS BITS (BIT #9)
1383        002000                  PSEL=   2000                    ;FORT SELECT (BIT #10)
1384        004000                  DVA=    4000                    ;DEVICE AVAILABLE (BIT #11)
1385        020000                  MCPE=   20000                   ;MASSBUSS PARITY ERROR (BIT #13)
1386        040000                  TRE=    40000                   ;TRANSFER ERROR (BIT #14)
1387        100000                  SC=     100000                  ;SPECIAL CONDITION (BIT #15)
1388
1389                                ;STATUS REGISTER (RHDS1) (#01)
1390
1391        000001                  DFF5=   1                       ;DRIVE FORWARD 5"/SEC.  (BIT #0)
1392        000002                  DFF20=  2                       ;DRIVE FORWARD 20"/SEC.  (BIT #1)
1393        000004                  DIGB=   4                       ;DRIVE TO INNER GAVRD BAND (BIT #2)
1394        000010                  GRV=    10                      ;GO REVERSE (BIT #3)
1395        000020                  DL64=   20                      ;DIFFERENCE LESS THAN 64 (BIT #4)
1396        000040                  DE1=    40                      ;DIFFERENCE EQUALS 1 (BIT #5)
1397        000100                  VV=     100                     ;VOLUME VALID (BIT #6)
1398        000200                  DRY=    200                     ;DRIVE READY (BIT #7)
1399        000400                  DPR=    400                     ;DRIVE PRESENT (BIT #8)
1400        001000                  PROG=   1000                    ;PROGRAMABLE (BIT #9)
1401        002000                  LBT=    2000                    ;LAST SECTOR TRANSFERRED (BIT #10)
1402        004000                  WRL=    4000                    ;WRITE LOCK (BIT #11)
1403        010000                  MOL=    10000                   ;MEDIUM ON-LINE (BIT #12)
1404        020000                  PIP=    20000                   ;POSITIONING OPERATION IN PROGRESS (BIT #13)
1405        040000                  ERR=    40000                   ;COMPOSIT ERROR. (BIT #14)
1406        100000                  ATA=    100000                  ;ATTENTION ACTIVE (BIT #15)
1407
1408                                ;ERROR REGISTER #01 (RHER1) (#02)
1409        000001                  ILF=    1                       ;ILLEGAL FUNCTION (BIT #0)
1410        000002                  ILR=    2                       ;ILLEGAL REGISTER (BIT #1)
1411        000004                  RMR=    4                       ;REGISTER MODIFICATION REFUSED (BIT #2)
1412        000010                  PAR=    10                      ;PARITY ERROR (BIT #3)
1413        000020                  FER=    20                      ;FORMAT ERROR (BIT #4)
1414        000040                  WCF=    40                      ;WRITE CLOCK FAIL (BIT #5)
1415        000100                  ECH=    100                     ;ECC HARD ERROR (BIT #6)
1416        000200                  HCE=    200                     ;HEADER COMPARE ERROR (BIT #7)
1417        000400                  HCRC=   400                     ;HEADER CRC ERROR (BIT #8)
1418        001000                  AOE=    1000                    ;ADDRESS OVERFLOW ERROR (BIT #9)
1419        002000                  IAE=    2000                    ;INVALID ADDRESS ERROR (BIT #10)
1420        004000                  WLE=    4000                    ;WRITE LOCK ERROR (BIT #11)
1421        010000                  DTE=    10000                   ;DRIVE TIMING ERROR (BIT #12)
1422        020000                  OPI=    20000                   ;OPERATION INCOMPLETE (BIT #13)
1423        040000                  UNS=    40000                   ;DRIVE UNSAFE (BIT #14)
1424        100000                  DCK=    100000                  ;DATA CHECK ERROR (BIT 15)
1425
1426                                ;MAINTAINABILITY REGISTER (RHMR)(#03)
1427
```

```
                        DMD=    1               ;DIAGINOSTIC MODE (BIT #0)
                        MCLK=   2               ;MAINTAINABILITY CLOCK (BIT #1)
                        MINX=   4               ;MAINTAINABILITY INDEX (BIT #2)
                        MSTCK=  10              ;MAINTAINABILITY SECTOR CLOCK (BIT #3)
                        MRD=    20              ;MAINTAINABILITY READ (BIT #4)
                        MWR=    40              ;MAINTAINABILITY WRITE (BIT #5)
                        DTSY=   1000            ;MAINTAINABILITY SYNC DETECTED (BIT #9)

                        ;ATTENTION SUMMARY PSEUDO-REGISTER (RHAS) (#04)

                        AT0=    1               ;DEVICE 0 (BIT #0)
                        AT1=    2               ;DEVICE 1 (BIT #1)
                        AT2=    4               ;DEVICE 2 (BIT #2)
                        AT3=    10              ;DEVICE 3 (BIT #3)
                        AT4=    20              ;DEVICE 4 (BIT #4)
                        AT5=    40              ;DEVICE 5 (BIT #5)
                        AT6=    100             ;DEVICE 6 (BIT #6)
                        AT7=    200             ;DEVICE 7 (BIT #7)



                        ;DESIRED SECTOR/TRACK ADDRESS REGISTER (RHDST) (#1)
                        ;EACH BIT IS CALLED BY BIT NUMBER



                        ;DRIVE TYPE REGISTER (RHDT) (#06)
                        ;EACH BIT IS CALLED BY BIT NUMBER



                        ;LOOK-AHEAD REGISTER (RHLA) (#07)

                        EXT1=   1               ;EXTENSION 1 (BIT #0)
                        EXT2=   2               ;EXTENSION 2 (BIT #1)
                        EXT4=   4               ;EXTENSION 3 (BIT #2)
                        EXT10=  10              ;EXTENSION 4 (BIT #3)
                        EXT20=  20              ;EXTENSION 5 (BIT #4)
                        EXT40=  40              ;EXTENSION 6 (BIT #5)
                        SC1=    100             ;SECTOR COUNT FIELD 0 (BIT #6)
                        SC2=    200             ;SECTOR COUNT FIELD 1 (BIT #7)
                        SC4=    400             ;SECTOR COUNT FIELD 2 (BIT #8)
                        SC10=   1000            ;SECTOR COUNT FIELD 3 (BIT #9)
                        SC20=   2000            ;SECTOR COUNT FIELD 4 (BIT #10)
                        TRK1=   4000            ;TRACK FIELD 1 (BIT #11)
                        TRK2=   10000           ;TRACK FIELD 2 (BIT #12)
                        TRK4=   20000           ;TRACK FIELD 3 (BIT #13)
                        TRK10=  40000           ;TRACK FIELD 4 (BIT #14)
                        TRK20=  100000          ;TRACK FIELD 5 (BIT #15)
```

```
1484
1485                                    ;ERROR REGISTER #2 (RHER2) (#10)
1486
1487            000001                  WCU=    1                       ;WRITE CURRENT UNSAFE (BIT #0)
1488            000002                  CSF=    2                       ;CURRENT SINK FAILURE (BIT #1)
1489            000004                  WSU=    4                       ;WRITE SELECT UNSAFE (BIT #2)
1490            000010                  CSU=    10                      ;CURRENT SWITCH UNSAFE (BIT #3)
1491            000020                  MSE=    20                      ;MOTOR SEQUENCE ERROR (BIT #4)
1492            000040                  TDF=    40                      ;TRANSITIONS DETECTOR FAILURE (BIT #5)
1493            000100                  TUF=    100                     ;TRANSITIONS UNSAFE (BIT #6)
1494            000200                  FEN=    200                     ;FAILSAFE ENABLED (BIT #7)
1495            000400                  WRU=    400                     ;WRITE READY UNSAFE (BIT #8)
1496            001000                  MHS=    1000                    ;MULTIPLE HEAD SELECT (BIT #9)
1497            002000                  NHS=    2000                    ;NO HEAD SELECTION (BIT #10)
1498            004000                  IXE=    4000                    ;INDEX ERROR (BIT #11)
1499            010000                  VU30=   10000                   ;30VOLT UNSAFE (BIT #12)
1500            020000                  PLU=    20000                   ;PLO UNSAFE (BIT #13)
1501            100000                  ACU=    100000                  ;ACUNSAFE (BIT #15)
1502
1503                                    ;OFFSET REGISTER (RHOF) (#11)
1504
1505            000001                  OF25=   1                       ;OFFSET 25 MICRO INCHES (BIT #0)
1506            000002                  OF50=   2                       ;OFFSET 50 MICRO INCHES (BIT #1)
1507            000004                  OF100=  4                       ;OFFSET 100 MICRO INCHES (BIT #2)
1508            000010                  OF200=  10                      ;OFFSET 200 MICRO INCHES (BIT #3)
1509            000020                  OF400=  20                      ;OFFSET 400 MICRO INCHES (BIT #4)
1510            000040                  OF800=  40                      ;OFFSET 800 MICRO INCHES (BIT #5)
1511
1512            000200                  OFREV=  200                     ;OFFSET NEGATIVE (REVERSE) (BIT #5)
1513            002000                  HCI=    2000                    ;HEADER COMPARE INHIBIT (BIT #10)
1514            004000                  ECI=    4000                    ;ERROR CORRECTION CODE INHIBIT (BIT #11)
1515            010000                  FMT22=  10000                   ;FORMAT BIT (BIT #12
1516
1517                                    ;DESIRED CYLINDER ADDRESS (RHCA) (#12)
1518                                    ;EACH BIT IS CALLED BY BIT NUMBER.
1519
1520
1521
1522
1523                                    ;CURRENT CYLINDER ADDRESS (RHCC) (#13)
1524                                    ;EACH BIT IS CALLED BY BIT NUMBER
1525
1526
1527
1528
1529                                    ;SERIAL NUMBER REGISTER (RHSN) (#14)
1530                                    ;EACH IS CALLED BY BIT NUMBER
1531
1532
1533
1534
1535                                    ;ERROR REGISTER #03 (RHER3) (#15)
1536
1537            000001                  PSU=    1                       ;PACK SPEED UNSAFE (BIT #0)
1538            000002                  VUF=    2                       ;VELOCITY UNSAFE (BIT #1)
1539            000010                  UWR=    10                      ;ANY UNSAFE EXCEPT READ/WRITE (BIT #3)
```

```
1540            000020                  PRE=    20              ;DISK PACK ROTATION ERROR (BIT #4)
1541            000040                  ACL=    40              ;AC LOW (BIT #5)
1542            000100                  DCL=    100             ;DC LOW (BIT #6)
1543            040000                  SKI=    40000           ;SEEK INCOMPLETE (BIT #14)
1544            100000                  OCYL=   100000          ;OFF CYLINDER (BIT #15)
1545
1546
1547
1548                                    ;ECC POSITION REGISTER (RHEC1) (#16)
1549                                    ;EACH BIT IS CALLED BY BIT NUMBER
1550
1551
1552
1553
1554                                    ;ECC PATTERN REGISTER (RHEC2) (#17)
1555                                    ;EACH BIT IS CALLED BY BIT NUMBER
1556
```

```
1557                                    .SBTTL  REGISTER ADDRESSES
1558
1559
1560
1561
1562                                    ;RP04/5/6 DISK I/O REGISTERS LOCATED IN THE RH11 CONTROLLER
1563
1564   002170   176722                  RHDB:   176722                  ;DATA BUFFER
1565   002172   176702                  RHWC:   176702                  ;WORD COUNT
1566   002174   176704                  RHBA:   176704                  ;BUS ADDRESS
1567   002176   176710                  RHCS2:  176710                  ;CONTROL AND STATUS 2
1568
1569                                    ;RP04/5/6 DISK I/O REGISTERS LOCATED IN THE DEVICE CONTROL LOGIC (DCL)
1570
1571   002200   176700                  RHCS1:  176700                  ;CONTROL AND STATUS 1
1572   002202   176714                  RHER1:  176714                  ;ERROR #1
1573   002204   176706                  RHDST:  176706                  ;DESIRED SECTOR/TRACK ADDRESS
1574   002206   176740                  RHER2:  176740                  ;ERROR #2
1575   002210   176732                  RHOF:   176732                  ;OFFSET
1576   002212   176734                  RHCA:   176734                  ;DESIRED CYLINDER ADDRESS
1577   002214   176742                  RHER3:  176742                  ;ERROR #3
1578   002216   176716                  RHAS:   176716                  ;ATTENTION SUMMARY
1579   002220   176724                  RHMR:   176724                  ;MAINTAINABILITY
1580   002222   176712                  RHDS1:  176712                  ;DRIVE STATUS
1581   002224   176726                  RHDT:   176726                  ;DRIVE TYPE
1582   002226   176730                  RHSN:   176730                  ;SERIAL NUMBER
1583   002230   176744                  RHEC1:  176744                  ;ECC POSITION
1584   002232   176746                  RHEC2:  176746                  ;ECC PATTERN
1585   002234   176736                  RHCC:   176736                  ;CURRENT CYLINDER ADDRESS
1586   002236   176720                  RHLA:   176720                  ;LOOK-AHEAD
1587
1588                                    ;ADDITIONAL REGISTERS LOCATED IN THE RH70 CONTROLLER LOGIC
1589
1590   002240   176750                  RHBAE:  176750                  ;BUS ADDRESS EXTENSION REGISTER
1591   002242   176752                  RHCS3:  176752                  ;CONTROL AND STATUS REGISTER #3
1592
1593
1594                                    ;P-CLOCK (KW11-P) I/O REGISTERS
1595
1596   002244   172540                  PCLCSR: 172540                  ;CONTROL AND STATUS REGISTERS
1597   002246   172542                  PCLBUF: 172542                  ;COUNT SET BUFFER
1598   002250   172544                  PCLCTR: 172544                  ;COUNTER
1599
```

```
1600
1601
1602                                    ;THE FOLLOWING LOCATIONS ARE RESERVED FOR REGISTER SAVES
1603                                    ;ANY TIME THERE IS AN ERROR ALL THESE WILL BE FILLED
1604                                    ;ONLY SOME MAY BE PRINTED BUT ALL WILL BE FILLED TRUE
1605                                    ;FOR THE TIME JUST AFTER THE "ERROR" ERROR COMMAND
1606
1607   002252  000000                   DB:     0                       ;DATA BUFFER
1608   002254  000000                   WC:     0                       ;WORD COUNT
1609   002256  000000                   BA:     0                       ;BUS ADDRESS
1610   002260  000000                   CS2:    0                       ;CONTROL AND STATUS 2
1611
1612
1613   002262  000000                   CS1:    0                       ;CONTROL AND STATUS 1
1614   002264  000000                   ER1:    0                       ;ERROR #1
1615   002266  000000                   DST:    0                       ;DESIRED SECTOR/TRACK ADDRESS
1616   002270  000000                   ER2:    0                       ;ERROR #2
1617   002272  000000                   OF:     0                       ;OFFSET
1618   002274  000000                   CA:     0                       ;DESIRED CYLINDER ADDRESS
1619   002276  000000                   ER3:    0                       ;ERROR #3
1620   002300  000000                   AS:     0                       ;ATTENTION SUMMARY
1621   002302  000000                   MR:     0                       ;MAINTAINABILITY
1622   002304  000000                   DS1:    0                       ;DRIVE STATUS
1623   002306  000000                   DT:     0                       ;DRIVE TYPE
1624   002310  000000                   SN:     0                       ;SERIAL NUMBER
1625   002312  000000                   EC1:    0                       ;ECC POSITION
1626   002314  000000                   EC2:    0                       ;ECC PATTERN
1627   002316  000000                   CC:     0                       ;CURRENT CYLINDER ADDRESS
1628   002320  000000                   LA:     0                       ;LOOK-AHEAD
1629
1630
1631
```

```
                                        ;FUNCTION EQUATES

                                                ;*TABLE OF FUNCTIONS FOR RHCS1 THEN "GO" BIT HAS TO BE SET
002322                                  FUTABL:
002322  000000                          NOPERA: 0                       ;NO OPERATION
002324  000002                          UNLOAD: 2                       ;UNLOAD (STAND BY)
002326  000006                          RECALI: 6                       ;RECALIBRATE
002330  000010                          DCLEAR: 10                      ;DRIVE CLEAR
002332  000012                          RELEAS: 12                      ;RELEASE (DUAL-PORT OPERATION)
002334  000030                          SERCH:  30                      ;SEARCH COMMAND
002336  000050                          WRCHEK: 50                      ;WRITE CHECK DATA
002340  000052                          WRCHDT: 52                      ;WRITE CHECK HEADER AND DATA
002342  000060                          WRIDAT: 60                      ;WRITE DATA
002344  000062                          WRIFOR: 62                      ;WRITE HEADER AND DATA (FORMAT)
002346  000070                          READAT: 70                      ;READ DATA
002350  000072                          REFOR:  72                      ;READ HEADER AND DATA
002352  000004                          SEECOM: 4                       ;SEEK COMMAND
002354  000014                          OFSETC: 14                      ;OFFSET COMMAND
002356  000016                          RETCL:  16                      ;RETURNTO CENTERLINE
002360  000022                          PKACK:  22                      ;PACK ACKNOWLEDGE
002362  000020                          READIN: 20                      ;READ IN
002364  000000                          ILLEGL: .WORD   0               ;COMPUTED ILLEGAL FUNCTION


                                                ;*DATA BUFFER FOR READ WRITE

002370  000422                          WRFROM: .BLKW   274.            ;WRITE FROM THIS BUFFER
003434  000422                          REINTO: .BLKW   274.            ;READ INTO THIS BUFFER
```

```
1666
1667
1668                                    ;RESERVED LOCATIONS FOR FLAGS AND INTERNAL PROGRAM CONTROL WORDS
1669
1670  004500  000000                    REGADR: 0                       ;SAVE REGISTER ADDRESS HERE
1671  004502  000000                    ERWORD: 0                       ;SAVE ERROR WORD NUMBER HERE
1672  004504  000000                    TSTNM:  0                       ;TEST NUMBER
1673  004506  000000                    RP4VEC: 0                       ;CONTAINS ADDRESS OF LOCATION
1674                                                                    ;WHERE AN RP04 INTERRUPT IS TO VECTOR TO
1675                                                                    ;THIS MUST BE MOVED INTO 'RPVEC' TO BE
1676                                                                    ;EFFECTIVE.
1677
1678  004510  000000                    OFSTVL: 0                       ;OFFSET VALUE USED IN OFFSET TEST
1679
1680
1681  004512  000024                    SAVERE: .BLKW   20.             ;BLOCK TO SAVE REGISTERS
1682  004562  000000                    FINALA: 0                       ;SAVE LOOK AHEAD REGISTER AT END OF OPERATION
1683  004564  000000                    FINACC: 0                       ;SAVE CURRENT CYLINDER REGISTER AT END OF OPERATION
1684
1685
1686                                    ;TABLE FOR ATTENTION BITS
1687                                    ;ATTENTION TABLE
1688
1689  004566    001     002     004     ATABLE: .BYTE   1,2,4,10,20,40,100,200
1690  004571    010     020     040
1691  004574    100     200
1692
1693
1694
1695                                    ;RESERVED LOCATIONS FOR UNIT SELECT
1696
1697  004576  000010                    UNITS:  .BLKW   8.              ;THIS IS FILLED WITH -1
1698  004616  000000                    UNIT:   .WORD   0               ;UNIT UNDER TEST
1699  004620  000000                    NOUNIT: .WORD   0               ;NUMBER OF UNITS PRESENT
1700                                                                    ;USED TO KEEP TRACK OF UNIT UNDER TEST
1701  004622  000000                    NUNIT:  .WORD   0               ;USED TO DETERMIN IF THERE ARE MORE
1702                                                                    ;THAN ONE UNIT
1703  004624  000000                    NOPUSH: 0                       ;ALL ONES INDICATE NONE OF THE OPERATOR
1704                                                                    ;INTERVENTION TESTE WILL BE PERFORMED
1705  004626  000000                    SELECT: .WORD   0               ;ALL ONES INDICATE UNIT TO BE SELECTED
1706  004630  000000                    UNITSL: .WORD   0               ;UNIT NO. SELECTED
1707
1708
1709
1710  004632  000000                    ERFLG$: 0                       ;ERROR FLAG
1711
1712  004634  000000                    FIRST:  0                       ;IF ZERO WILL TYPE HEADER
1713                                                                    ;IF ONES WILL NOT TYPE HEADER
1714
1715  004636  000000                    RP06:   0                       ;DEVICE TYPE FLAG
1716
1717  004640  000000                    RH70:   0                       ;IF 1, PROGRAM IS RUNNING ON AN RH70
1718                                                                    ;IF 0, PROGRAM IS ON AN RH11
1719
1720  004642  000000                    RUNCTR: .WORD   0               ;'RUN' LINE DELAY COUNTER TO BE USED
1721                                                                    ;WHILE THE SILO IS FILLING
```

```
 1722
 1723  004644  000000                ATTENT: 0                   ;ATTENTION BIT FOR PRESENT UNIT
 1724  004646  000000                TOTALAT:        0           ;TOTAL ATTENTION BITS
 1725
 1726  004650  000000                TMP0:   .WORD   0           ;TEMP STORAGE
 1727  004652  000000                TMP1:   .WORD   0
 1728  004654  000000                TMP4:   .WORD   0           ;TEMP STORAGE
 1729  004656  000000                TMP5:   .WORD   0           ;TEMP STORAGE
```

```
                                        .SBTTL
                                        .SBTTL  *** DIAGNOSTIC CODE ***
                                        .SBTTL
                                        .SBTTL  SETUP TESTS
004660  012737  177777  004624  BEGIN1: MOV     #-1,@#NOPUSH    ;JUMP OVER OPERATOR REQUIRED TESTS
004666  005037  004626                  CLR     @#SELECT        ;DO NOT SELECT UNIT
004672  000412                          BR      START
004674  012737  177777  004626  BEGIN2: MOV     #-1,@#SELECT    ;SELECT UNIT
004702  005037  004624                  CLR     @#NOPUSH        ;DO NOT JUMP OVER ANY TEST
004706  000404                          BR      START
004710  005037  004626          BEGIN:  CLR     @#SELECT        ;DO NOT SELECT UNIT
004714  005037  004624                  CLR     @#NOPUSH        ;DO NOT JUMP OVER ANY TEST
                                                                ;NORMAL RUN
004720                          START:
004720  000005                          RESET
005142  012737  000000  177776          MOV     #0,PS           ;SET PROCESSOR STATUS TO 0
005150  012737  000200  000036          MOV     #200,@#TRAPVEC+2 ;TRAP PRIORITY = 4
005156  013700  002166                  MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
005162  012720  036306                  MOV     #RPVECT,(R0)+   ;THIS IS FOR UNTIMELY INTERRUPTS
005166  012710  000340                  MOV     #340,(R0)       ;RP04 INTERRUPT SERVICE ROUTINE
                                                                ;PRIORITY = 7
005172  004737  037334                  JSR     PC,@#$TKINT     ;INITIALIZE THE TTY KEYBOARD
005176  005737  004634                  TST     @#FIRST         ;IS THIS FIRST TIME ROUND ?
005202  001001                          BNE     1$              ;DO NOT GIVE HEADER IF NOT
005204  000402                          BR      2$              ;GIVE HEADER IF SO
005206  000137  005774          1$:     JMP     @#SND1          ;SKIP OVERALL PROGRAM HEADER
005212                          2$:
005774  012737  177777  004634  SND1:   MOV     #-1,@#FIRST     ;NEXT TIME DO NOT GIVE HEADER
006032  032777  010000  173100  RH70CK: BIT     #SW12,@SWR      ;LOOK TO SEE IF USING RH70
006040  001403                          BEQ     3$              ;IF SW12 = 0, SKIP NEXT
006042  012737  000001  004640          MOV     #1,@#RH70       ;IF SW12 = 1, CU IS AN RH70
006050                          3$:
```

```
1776                                            ;*IS THERE A P-CLOCK (KW11-P) ON THE SYSTEM ?
1777                                            ;*IF SO MAKE 'WAT" TRAPS GO TO 'WAIT.P'
1778                                            ;*IF SO MAKE RP04 INTERRUPTS GO TO 'TIME 1'
1779                                            ;*IF NOT MAKE 'WAT' TRAPS GO TO 'WAIT.T'
1780                                            ;*IF NOT MAKE RP04 INTERRUPTS GO TO 'TIME 2'
1781
1782                                            ;*THE NEXT LINE IS TO BE ADDED LATER
1783                                            ;*AND THE JUMP AND NOP REMOVED
1784                                            ;*FOR NOW NO CLOCK WILL BE USED
1785
1786                                            ;*MOV   @#1$,@#ERRVEC   ;SET TIME-OUT VECTOR
1787
1788                                    ;       JMP     @#1$            ;DO NOT USE CLOCK
1789                                    ;       NOP
1790                                    ;       TST     @#PCLCSR        ;REFERENCE P-CLOCK STATUS REGISTER
1791                                    ;                               ;ADDRESS = 172540
1792                                    ;       MOV     #WAIT.P,@#STRPAD+20 ;THERE IS A P-CLOCK
1793                                    ;       MOV     #TIME1,@#RP4VEC ;THERE IS A P CLOCK SO
1794                                    ;                               ;VECTOR TO TIME1
1795                                    ;       BR      2$
1796                                    ;1$:    MOV     #WAIT.T,@#STRPAD+20 ;THERE IS NO P-CLOCK
1797
1798
1799  006050  012737  033300  004506          MOV     #TIME2,@#RP4VEC ;MAKE RP04/5/6 INTERRUPTS GO TO 'TIME 2'
1800  006056  012737  177777  040742  2$:     MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
1801
1802  006064  005737  004626                  TST     @#SELECT        ;WAS IT A 200 START
1803  006150  104412                          RDOCT
1804  006152  042716  177770                  BIC     #177770,(SP)    ;ONLY KEEP LAST 3 BITS
1805  006156  011637  004616                  MOV     (SP),@#UNIT     ;SAVE UNIT TO BE TESTED
1806  006162  012637  004630                  MOV     (SP)+,@#UNITSL  ;SAVE UNIT TO BE TESTED
1807  006170  013737  004630  004616          MOV     @#UNITSL,@#UNIT ;SET UNIT NUMBER
1808
```

```
1809
1810  006214  012706  001000                  MOV     #STACK, SP      ;SET UP STACK POINTER
1811  006220  012737  040604  000030          MOV     #REGSAI,@#EMTVEC;ERROR VECTOR SO THAT
1812                                                                    ;NO REGISTERS ARE SAVED
1813  006226  012737  006254  000004          MOV     #2$,@#ERRVEC    ;SET UP FOR BUS TIMEOUT
1814
1815  006234  012700  000024                  MOV     #24,R0          ;THERE ARE 24 REG TO TEST
1816  006240  012701  002170                  MOV     #RHDB,R1        ;R1 NOW HAS ADDR OF ADDR OF FIRST REG.
1817  006244  013102                  1$:     MOV     @(R1)+, R2      ;READ HARDWARE REG.
1818  006246  005300                          DEC     R0              ;COUNT DOWN
1819  006250  001375                          BNE     1$              ;BRANCH IF 24 NOT DONE
1820  006252  000454                          BR      3$              ;BRANCH IF 24 DONE
1821  006254  012737  000006  000004  2$:     MOV     #ERRVEC+2,@#ERRVEC ;RESTORE TRAP CATCHER
1822  006262  022626                          CMP     (SP)+,(SP)+     ;CLEAN STACK
1823  006264  016137  177776  001200          MOV     -2(R1), $TMP1   ;STORE FAILING REG ADDR
1824  006272  104007                          ERROR   7               ;REGISTER NON EXISTANT
1825  006274  032777  020000  172636          BIT     #SW13,@SWR      ;INHIBIT ERROR PRINTOUT ?
1826  006302  001036                          BNE     4$              ;BRANCH IF YES
1827
1828
1829  006370  012746  000204                  MOV     #ADDMOD,-(SP)   ;GET READY TO TYPE STARTING ADDRESS
1830                                                                    ;OF "CHANGE OF BASE ADDRESS" ROUTINE
1831  006374  104402                          TYPOC
1832  006376  000000                          HALT
1833  006400  000137  032570          4$:     JMP     @#$EOP          ;GO TO END OF PROGRAM ---------->
1834
1835  006404  012737  006460  000004  3$:     MOV     #TRP,@#4        ;INITIALIZE VECTOR
1836  006412  005737  002240                  TST     @#RHBAE         ;ADDRESS RPBAE(RH11/RH70?)
1837  006416  005237  004640                  INC     @#RH70          ;FOUND AN RH70-SET MASK
1838  006456  000417                          BR      RTN             ;GET OUT
1839  006460  022626                  TRP:    CMP     (SP)+,(SP)+     ;SET UP THE
1840  006516  012737  040574  000030  RTN:    MOV     #$ERROR,@#EMTVEC;RESTORE ERROR VECTOR
1841                                                                    ;SO THAT REGISTERS ARE SAVED
1842  006524  012737  000006  000004          MOV     #ERRVEC+2,@#ERRVEC ;RESTORE TRAP CATCHER
1843
```

```
1844
1845   006542   012706   001000                    MOV     #STACK,SP       ;SET STACK POINTER
1846
1847   006546   013701   002216                    MOV     @#RHAS,R1       ;R1 HAS ADDRESS OF RHAS
1848   006552   012711   177777                    MOV     #-1,@R1         ;WRITE ALL ONES INTO RHAS
1849   006556   105711                             TSTB    @R1             ;TEST IT FOR ALL 0'S
1850   006562   011137   001126                    MOV     @R1,@#$BDDAT    ;BAD DATA
1851   006566   005037   001124                    CLR     @#$GDDAT        ;GOOD DATA
1852   006572   010137   004500                    MOV     R1,@#REGADR     ;FAILING REG. RHAS
1853   006576   104005                             ERROR   5               ;RHAS DOES NOT CLEAR BY WRITING ALL
1854                                                                       ;ONES INTO IT
1855
```

```
1856
1857
1858  006616  000005                          RESET                   ;START WITH AN INIT
1859  006620  004737  037334                  JSR     PC,@#$TKINT     ;INITILIZE THE TTY KEYBOARD
1860
1861  006624  032777  020000  172306          BIT     #SW13,@SWR      ;INHIBIT ERROR TYPEOUT?
1862  006632  001026                          BNE     4$              ;BRANCH IF YES
1863  006710  013701  002216          4$:     MOV     @#RHAS,R1       ;R1 HAS ADDR. OF RHAS
1864  006714  013702  002176                  MOV     @#RHCS2,R2      ;R2 HAS ADDR. OF RHCS2
1865  006720  005012                          CLR     @R2             ;CLEAR RHCS2
1866  006722  012700  000010                  MOV     #8.,R0          ;COUNT
1867  006726  013704  002202                  MOV     @#RHER1,R4      ;R4 HAS ADDR. OF RHER1
1868  006732  012714  177777          1$:     MOV     #-1,@R4         ;MOVE ERRORS INTO RHER1
1869  006736  005212                          INC     @R2             ;INCREMENT UNIT NO.
1870  006740  005300                          DEC     R0              ;COUNT
1871  006742  001373                          BNE     1$              ;BRANCH IF 8 NOT DONE
1872  006744  111137  004646                  MOVB    @R1,@#TOTALAT   ;SAVE TOTAL ATTENTION
1873                                                                  ;USED IN DRIVE CLEAR TEST
1874  006750  105037  004647                  CLRB    @#TOTALAT+1     ;CLEAR UPPER BYTE
1875  006754  105711                          TSTB    @R1             ;TEST FOR ANY DRIVES PRESENT
1876  006756  001402                          BEQ     2$              ;IF NONE THERE - SAY SO
1877  006760  000137  007342                  JMP     XE2             ;SOME THERE - LOAD TABLE
1878
1879  006764  032777  020000  172146  2$:     BIT     #SW13,@SWR      ;INHIBIT ERROR TYPE OUT?
1880  006772  001402                          BEQ     3$              ;BRANCH IF NO
1881  006774  000137  007700                  JMP     SELTST          ;CHECK FOR SELECTED UNIT START AND LOAD
1882                                                                  ;"UNITS" TABLE WITH DESIRED DRIVE IF SO
1883
1884  007000                          3$:
1885
1886  007336  000137  032570                  JMP     @#$EOP          ;GO OUT ---------------------------->
1887
1888
1889                                          ;*SET UP THE UNITS TABLE
1890
1891  007342                          XE2:
1892  007342  012700  000010          2$:     MOV     #8.,R0          ;COUNTER
1893  007346  012703  004576                  MOV     #UNITS,R3       ;POINTER
1894  007352  012723  177777          3$:     MOV     #-1,(R3)+       ;PRESET BLOCK TO ALL ONES
1895  007356  005300                          DEC     R0              ;COUNT
1896  007360  001374                          BNE     3$              ;BRANCH IF 8 NOT DONE
1897  007362  012703  004576                  MOV     #UNITS,R3       ;POINTER
1898  007366  005005                          CLR     R5
1899  007370  005037  004620                  CLR     @#NOUNIT        ;NO. OF UNITS PRESENT
1900  007374  012700  000010                  MOV     #8.,R0          ;COUNTER
1901  007400  011137  001176                  MOV     @R1,@#$TMP0     ;TEMPORARY STORAGE
1902  007404  006037  001176          4$:     ROR     @#$TMP0         ;SET CARRY IF ONE IN  0 BIT
1903  007410  103120                          BCC     5$              ;CHECK NEXT UNIT IF ONE NOT IN BIT 0
1904
1905  007412  010577  172560                  MOV     R5,@RHCS2       ;INSERT UNIT NUMBER INTO UA BITS
1906  007416  022777  024020  172600          CMP     #24020,@RHDT    ;IS THIS A DUAL PORT RP04 ?
1907  007424  001503                          BEQ     6$              ;TYPE DRIVE NO. IF SO
1908  007426  022777  020020  172570          CMP     #20020,@RHDT    ;IS THIS A SINGLE PORT RP04 ?
1909  007434  001477                          BEQ     6$              ;TYPE NO. IF SO
1910
1911  007436  022777  024021  172560          CMP     #24021,@RHDT    ;IS THIS A DUAL PORT RP05 ?
```

```
1912  007444  001473                        BEQ     6$                ;TYPE DRIVE NO. IF SO
1913  007446  022777  020021  172550        CMP     #20021,@RHDT      ;IS THIS A SINGLE PORT RP05 ?
1914  007454  001467                        BEQ     6$                ;TYPE THE NO. IF SO
1915
1916  007456  022777  024022  172540        CMP     #24022,@RHDT      ;IS THIS A DUAL PORT RP06 ?
1917  007464  001463                        BEQ     6$                ;TYPE DRIVE NO. IF SO
1918  007466  022777  020022  172530        CMP     #20022,@RHDT      ;IS THIS A SINGLE PORT RP06 ?
1919  007474  001457                        BEQ     6$                ;TYPE DRIVE NO. IF SO
1920
1921
1922  007524  010546                        MOV     R5,-(SP)          ;GET READY TO TYPE UNIT NUMBER
1923  007526  104405                        TYPDS
1924  007552  017746  172446                MOV     @RHDT,-(SP)       ;GET READY TO TYPE RHDT
1925  007556  104402                        TYPOC
1926  007632  000407                        BR      5$                ;NO RP04/RP05/RP06 FOUND SO INCR TABLE
1927
1928  007634  010523              6$:       MOV     R5,(R3)+
1929  007636  104401  001223                TYPE    ,$CRLF
1930  007642  010546                        MOV     R5,-(SP)
1931  007644  104405                        TYPDS                     ;TYPE DRIVE NO.
1932  007646  005237  004620                INC     @#NOUNIT          ;NUMBER OF DRIVES
1933
1934  007652  005205              5$:       INC     R5                ;INCR UNIT NUMBER
1935  007654  005300                        DEC     R0                ;DECR NO. OF UNITS LOOKED AT
1936  007656  001252                        BNE     4$                ;TEST THE NEXT UNIT
1937
1938  007660  013737  004576  004616        MOV     @#UNITS,@#UNIT    ;SET UNIT NO. TO FIRST ONE FOUND/OR 0
1939  007666  013737  004620  004622        MOV     @#NOUNIT,@#NUNIT  ;SAVE NO. OF UNITS
1940  007674  005337  004622                DEC     @#NUNIT           ;IF NUNIT = 0 THEN ONLY ONE UNIT
1941                                                                  ;IF NUNIT > 0 THEN MORE THAN ONE UNIT
1942
1943  007700  005737  004626      SELTST:   TST     @#SELECT          ;STARTING ADDRESS 200 ?
1944  007706  013737  004630  004616        MOV     @#UNITSL,@#UNIT   ;SET UNIT NUMBER
```

```
1945
1946
1947
1948  007750  005037  004644                  CLR     @#ATTENT        ;CLEAR UNIT UNDER TEST ATTENTION
1949
1950  007754  005737  004616                  TST     @#UNIT          ;IS THE "UNIT" = 0 ?
1951  007760  001107                          BNE     20$             ;IF NOT, SKIP NEXT MODS
1952  007762  012700  000041                  MOV     #41,R0          ;IF SO, CHECK THE LOAD MEDIA LOCATION
1953  007766  122710  000011                  CMPB    #11,(R0)        ;WAS IT AN RP04/5/6 ?
1954  007772  001102                          BNE     20$             ;IF NOT, GO AHEAD AND TEST UNIT #0
1955  007774  005737  004626                  TST     @#SELECT        ;WAS UNIT #0 SELECTED ?
1956                                                                  ;(IE. 210 START ?)
1957  010000  001006                          BNE     19$             ;IF SO, CHANGE PACK
1958
1959                                          ;*INCREMENT THE UNITS TABLE TO NEXT DRIVE (IF ANY)
1960                                          ;*& DECREMENT "NOUNITS" PRESENT TO BE TESTED
1961
1962  010002  012700  004576                  MOV     #UNITS,R0       ;IF NOT, LOAD THE UNITS TABLE POINTER
1963  010006  005720                          TST     (R0)+           ;SELECT THE NEXT UNIT IN TABLE
1964                                                                  ;(DOUBLE INCREMENT THE POINTER, R0)
1965  010010  022710  177777                  CMP     #-1,(R0)        ;IS THERE ANOTHER TABLE ENTRY PRESENT ?
1966  010014  001065                          BNE     18$             ;IF SO, USE NEXT DRIVE & DECR "NOUNITS"
1967                                                                  ;IF NOT, CHANGE PACK ON UNIT #0
1968
1969  010016                          19$:
1970  010164  000000                          HALT
1971  010166  000404                          BR      20$             ;CONTINUE, USING SCRATCH PACK ON UNIT #0
1972
1973  010170  011037  004616          18$:    MOV     (R0),@#UNIT     ;SET UP NEW UNIT UNDER TEST
1974  010174  005337  004620                  DEC     @#NOUNITS       ;DECR BECAUSE UNIT #0 WON'T BE TESTED
1975
1976  010200  013700  004616          20$:    MOV     @#UNIT,R0       ;R0 NOW CONTAINS UNIT NO
1977
1978
1979
1980
1981  010204  005037  004636                  CLR     @#RP06          ;CLEAR RP06 DEVICE TYPE FLAG
1982  010210  010077  171762                  MOV     R0,@RHCS2       ;SET UP UNIT ADDRESSING
1983  010214  022777  024022  172002          CMP     #24022,@RHDT    ;DUAL PORT RP06 ?
1984  010222  001405                          BEQ     2$              ;YES..SET FLAG
1985  010224  022777  020022  171772          CMP     #20022,@RHDT    ;SINGLE PORT RP06 ?
1986  010232  001401                          BEQ     2$              ;YES...SET FLAG
1987  010234  000403                          BR      3$              ;DON'T SET RP06 FLAG
1988  010236  012737  177777  004636  2$:     MOV     #-1,@#RP06      ;SET THE FLAG
1989
1990  010244                          3$:                             ;ASSUME THE NEXT UNIT IS AN RP04/RP05
1991
1992
1993  010244  116037  004566  004644          MOVB    ATABLE(R0),@#ATTENT ;SET APPROPRIATE ATTENTION BIT
1994  010310  013746  004616                  MOV     @#UNIT,-(SP)    ;UNIT NO. TO STACK
1995  010314  104405                          TYPDS                   ;TYPE  DRIVE NO.
```

```
1996
1997                                            ;*TYPE OUT THE DRIVE TYPE IN ASCII
1998
1999
2000  010406  022777  024020  171610           CMP     #24020,@RHDT    ;DUAL PORT RP04 ?
2001  010414  001424                           BEQ     4$              ;TYPE ASCII MESSAGE OUT
2002  010416  022777  020020  171600           CMP     #20020,@RHDT    ;SINGLE PORT RP04 ?
2003  010424  001420                           BEQ     4$              ;TYPE THE MESSAGE
2004
2005  010426  022777  024021  171570           CMP     #24021,@RHDT    ;DUAL PORT RP05 ?
2006  010434  001433                           BEQ     5$              ;TYPE THE MESSAGE
2007  010436  022777  020021  171560           CMP     #20021,@RHDT    ;SINGLE PORT RP05 ?
2008  010444  001427                           BEQ     5$              ;TYPE THE MESSAGE
2009
2010  010446  022777  024022  171550           CMP     #24022,@RHDT    ;DUAL PORT RP06 ?
2011  010454  001442                           BEQ     6$              ;TYPE THE MESSAGE
2012  010456  022777  020022  171540           CMP     #20022,@RHDT    ;SINGLE PORT RP06 ?
2013  010464  001436                           BEQ     6$              ;TYPE THE MESSAGE
2014
2015  010466                          4$:
2016  010522  000435                           BR      1$              ;SKIP NEXT ONES
2017  010524                          5$:
2018  010560  000416                           BR      1$              ;SKIP NEXT
2019  010562                          6$:
2020
2021
2022
2023
2024  010616  005777  171404          1$:     TST     @RHSN           ;READ SERIAL NO. AND DRIVE TYPE
2025  010622  005777  171376                  TST     @RHDT           ;THESE TWO ARE TO HELP SCOPE LOOPS
2026  010626  017737  171374  002310          MOV     @RHSN,@#SN      ;SAVE TO CHECK IF DRIVE CLEAR CLEARS ANY BITS
2027  010634  017737  171364  002306          MOV     @RHDT,@#DT      ;SAVE TO CHECK IF DRIVE CLEAR CLEARS ANY BITS
```

```
2028
2029
2030
2031
2032  010652  004737  033066              JSR     PC,@#CLDISK     ;GIVE INITILIZE
2033  010656  032713  010000              BIT     #MOL,@R3        ;CHECK MOL IN RHDS1
2034  011100  032713  010000      1$:     BIT     #MOL,@R3        ;CHECK MOL IN RHDS1
2035  011104  001775                      BEQ     1$              ;WAIT IF MOL IS STILL LOW
2036
2037
2038
```

```
2039
2040
2041  011204  012706  001000                    MOV     #STACK,SP       ;RESET STACK
2042
2043  011214  013700  002166                    MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
2044  011220  012720  011266                    MOV     #RPTRP1,(R0)+   ;THIS IS FOR TIMELY INTERRUPTS
2045  011224  012710  000340                    MOV     #340,(R0)       ;RP04 INTERRUPT SERVICE ROUTINE
2046                                                                    ;PRIORITY = 7
2047  011230  012737  000200  177776            MOV     #200,PS         ;SET PROCESSOR PRIORITY @ 4
2048  011236  012711  000300                    MOV     #RDY!IE,@R1     ;RDY, IE IN RHSC1 SHOULD CAUSE INTERRUPT
2049
2050  011242  013737  033564  001200            MOV     @#TIMCNT,@#$TMP1;COUNTER
2051  011250  005337  001200          1$:       DEC     @#$TMP1         ;WAIT FOR INTERRUPT
2052  011254  001375                            BNE     1$              ;BRANCH IF NOT ZERO
2053                                                                    ;BEFORE THIS IS ZERO INTERRUPT SHOULD OCCUR
2054  011256  104065                            ERROR   65              ;INTERRUPT DID NOT OCCUR
2055  011260  012712  000040                    MOV     #CLR,@R2        ;CLEAR CONTROLLER VIA CS2
2056
2057  011266  022626                  RPTRP1:   CMP     (SP)+,(SP)+     ;RESTORE STACK
2058  011270  022711  004200                    CMP     #DVA!RDY,@R1    ;IE SHOULD BE LOW
2059  011276  104065                            ERROR   65              ;INTERRUPT OCCURED BUT
2060                                                                    ;IE FAILED TO RESET
2061  011300  012712  000040                    MOV     #CLR,@R2        ;CLEAR CONTROLLER
```

```
2062
2063
2064
2065  011314  012706  001000                    MOV     #STACK,SP       ;RESET STACK
2066
2067  011324  013700  002166                    MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
2068  011330  012720  011374                    MOV     #RPTRP2,(R0)+   ;THIS IS FOR UNTIMELY INTERRUPTS
2069  011334  012710  000340                    MOV     #340,(R0)       ;RP04 INTERRUPT SERVICE ROUTINE
2070                                                                    ;PRIORITY = 7
2071  011340  012737  000240  177776            MOV     #240,PS         ;SET PROCESSOR PRIORITY = 5
2072  011346  012711  000300                    MOV     #RDY!IE,@R1     ;RDY, IE IN RHSC1 WHOULD CAUSE INTERRUPT
2073
2074  011352  013737  033564  001200            MOV     @#TIMCNT,@#$TMP1;COUNTER
2075  011360  005337  001200          1$:       DEC     @#$TMP1         ;WAIT FOR INTERRUPT
2076  011364  001375                            BNE     1$              ;BRANCH IF NOT ZERO
2077                                                                    ;BEFORE THIS IS ZERO INTERRUPT WHOULD OCCUR
2078  011366  012712  000040                    MOV     #CLR,@R2        ;CLEAR CONTROLLER
2079
2080  011374  022626                  RPTRP2:   CMP     (SP)+,(SP)+     ;RESTORE STACK
2081  011376  104065                            ERROR   65              ;INTERRUPT OCCURRED WITH
2082                                                                    ;PROCESSOR PRIORITY SAME AS DISK
2083  011400  012712  000040                    MOV     #CLR,@R2        ;CLEAR CONTROLLER
2084
2085
2086
2087
```

```
                                        ;*IN CASE THERE IS ANY DRIVE ERRORS DURING POWER UP
                                        ;*OR POWER DOWN OR ANY PARITY ERRORS A RESET IS GIVEN
                                        RESET
011406  000005                          
011410  004737  037334                  JSR     PC,@#$TKINT     ;INITILIZE TK
011414  012737  000000  177776          MOV     #0,PS

                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER PACK ACKNOWLEDGE

                                        ;*NOW COMPARE REGISTERS BEFORE PACK ACKNOWLEDGE
                                        ;*WITH AFTER PACK ACKNOWLEDGE

011644  104015                  1$:     ERROR   15              ;GIVING A PACK ACKNOWLEDGE
011646  000207                          RTS     PC              ;CAUSED AN ERROR
                                                                ;PACK ACKNOWLEDGE SHOULD
                                                                ;SET VV IN RHDS1
                                                                ;INTERRUPT SHOULD MAKE
                                                                ;IE = 0
                                                                ;NO OTHER REGISTERS SHOULD
                                                                ;CHANGE
                                                                ;GOOD DATA GIVES CONTENTS
                                                                ;OF REGISTER BEFORE COMMAND
                                                                ;RECEIVED DATA GIVES CONTENTS
                                                                ;OF REGISTER AFTER COMMAND
011650                          2$:
```

```
                                .SBTTL  DATA TRANSFER RELATED ERRORS (USING MEDIA)


                                        ;*CHECK THE DRIVE TYPE AND THEN FILL THE
                                        ;*WRITE FROM BUFFER WITH APPROPRIATE HEADER
011670  005737  004636                  TST     @#RP06  ;TEST FOR RP06 DRIVE
011674  001412                          BEQ     11$             ;TREAT UNIT AS AN RP04
                                                                ;TREAT UNIT AS AN RP06

011720  000411                          BR      12$             ;CONTINUE WITH SET UP

011722                          11$:
011744                          12$:

                                        ;*FILL READ INTO BUFFER WITH ALL ONES
```

```
                                        ;*DRIVE TYPE IS CHECKED AND THEN THE APPROPRIATE
                                        ;*WRITE HEADER AND DATA COMMAND IS LOADED

011756  005737  004636                  TST     @#RP06  ;TEST FOR RP06 DRIVE
011762  001412                          BEQ     7$              ;TREAT UNIT AS RP04

012006  000411                          BR      8$              ;CONTINUE WITH TEST

012010                          7$:
012032                          8$:

                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE

                                        ;*TIME IS NOT CRITICAL HERE

                                        ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE

                                        ;*COMPARE ALL REGISTERS

012210  104045                  1$:     ERROR   45              ;WRITING ON THE LAST BLOCK
012212  000207                          RTS     PC              ;IE. CYLINDER 410./814., SECTOR 21
                                                                ;TRACK 18 CAUSED
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER WRITE
                                                                ;ON LAST BLOCK
```

```
                                                ;*NOW A READ DATA WILL BE DONE ON SAME CYLINDER, SECTOR & TRACK
                                                ;*CLEAR ANY PREVIOUS ERRORS
012214                                  2$:
                                                ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
                                                ;*FIRST THE DRIVE TYPE IS CHECKED AND THEN THE APPROPRIATE
                                                ;*READ COMMAND IS LOADED
012244  005737  004636                        TST     @#RP06  ;TEST FOR RP06 DRIVE
012250  001412                                BEQ     9$              ;TREAT UNIT AS RP04
012274  000411                                BR      10$             ;CONTINUE WITH TEST
012276                                  9$:
012320                                  10$:
                                                ;*SAVE REGISTERS FOR COMPARISON AFTER READ DATA
                                                ;*TIME IS NOT CRITICAL HERE
                                                ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
                                                ;*COMPARE ALL REGISTERS
012476  104045                          3$:     ERROR   45              ;READING ON LAST BLOCK IE.
012500  000207                                RTS     PC              ;CYLINDER 410./814., SECTOR 21, TRACK 18
                                                                        ;CAUSED AN ERROR
                                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                                        ;BE THERE
                                                                        ;RECEIVED DATA GIVES WHAT
                                                                        ;WAS THERE AFTER READ
                                                                        ;FROM LAST BLOCK
                                                ;*READ DATA WILL BE COMPARED
012502                                  4$:
012520  104046                          5$:     ERROR   46              ;DATA READ FROM
012522  000207                                RTS     PC              ;LAST BLOCK IN ERROR
012524                                  6$:
```

```
                                        ;*GET HEADS TO CYLINDER 0



                                        ;*FILL WRITE FROM BUFFER WITH HEADER
                                        ;*FILL WRITE FROM BUFFER WITH DATA
                                        ;*FILL WRITE FROM BUFFER WITH NEXT SECTOR HEADER
                                        ;*FILL WRITE FROM BUFFER WITH NEXT SECTOR DATA
                                        ;*CLEAR READ INTO BUFFER WITH DATA OTHER THAN EXPECTED DATA
                                        ;*THE WRITE HEADER AND DATA WILL BE LOADED
                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE HEADER AND DATA



                                        ;*ONE REVOLUTION=16670 MICRO SEC, ONE SECTOR = 760 MICRO SEC
                                        ;*NOW CHANGE SAVE REGISTERS TO EXPECTED VALUES
                                        ;*NOW COMPARE REGISTERS BEFORE WRITE HEADER AND DATA
                                        ;*WITH REGISTERS AFTER COMMAND

013150  104027                  1$:   ERROR   27              ;WRITE HEADER AND DATA
013152  000207                        RTS     PC              ;CAUSED IMPROPER REGISTER
                                                              ;CHANGE
                                                              ;GOOD DATA GIVES WHAT SHOULD
                                                              ;BE THERE
                                                              ;RECEIVED DATA GIVES WHAT
                                                              ;WAS THERE AFTER COMMAND

                                        ;*NOW A SEARCH COMMAND WILL BE GIVEN
                                        ;*BUT BEFORE THAT ALL POSSIBLE REGISTERS
                                        ;*WILL BE FILLED FOR THE READ HEADER AND DATA SECTOR 1
                                        ;*AS THERE WILL NOT BE MUCH TIME BETWEEN THE
                                        ;*COMPLETION OF THE SEARCH AND THE SECTOR 1 COMING.

                                        ;*FILL FOR THE READ HEADER AND DATA COMMAND WHICH WILL NOT
                                        ;*BE EXECUTED TILL AFTER THE SEARCH
                                        ;*THE SEARCH WILL ONLY LEAVE RHCS1 AND RHDST
                                        ;*CHANGED ALL THE REST WILL BE UNCHANGED

013154                            2$:

                                        ;*SAVE REGISTERS FOR COMPARISON NOT AFTER THE
                                        ;*SEARCH COMMAND BUT AFTER THE READ HEADER AND DATA

                                        ;*NOW SAVE VALUES FOR RHCS1 AND RHDST WHICH
                                        ;*WILL CHANGE AFTER THE SEARCH
```

```
2279
2280  013214  013746  002350                  MOV     @#REFOR,-(SP)   ;SAVE READ HEADER AND DATA
2281  013220  052716  000101                  BIS     #IE!GO,(SP)     ;INTERRUPT ENABLE AND GO
2282  013224  012637  004650                  MOV     (SP)+,@#TMP0    ;SAVE IN R0 FOR RHCS1
2283  013230  012737  000001  004656          MOV     #1,@#TMP5       ;SAVE TRACK 0 SECTOR 1 FOR RHDST
2284
2285                                          ;*THE INTERRUPT VECTOR WILL BE SET TO GO TO 2$
2286                                          ;*AFTER THE SEARCH
2287
2288  013236  012777  013304  166722          MOV     #7$,@RPVEC      ;SET INTERRUPT VECTOR TO 2$
2289
2290                                          ;*TIME IS NOT CRITICAL THIS ONLY WAITS FOR SEARCH COMPLETION
2291
2292  013304  012737  000000  177776  7$:     MOV     #0,PS           ;SET PROSESSOR STATUS TO
2293                                                                  ;PRIORITY 0 IN CASE IT WAS
2294                                                                  ;TAKEN OUT OF WAT ROUTINE
2295                                                                  ;BEFORE RTI
2296  013312  013777  004656  166664          MOV     @#TMP5,@RHDST   ;SET DESIRED SECTOR/TRACK
2297                                                                  ;REGISTER TO SECTOR 1,TRACK 0
2298
2299  013326  013777  004650  166644          MOV     @#TMP0,@RHCS1   ;FILL RHCS1 WITH READ COMMAND
2300                                                                  ;TOGETHER WITH INTERRUPT ENABLE
2301                                                                  ;AND GO
2302
2303                                          ;*TIME ALLOWED HERE IS CRITICAL ANY TIME ERROR
2304                                          ;*INDICATES WRONG SEARCH IN THE SEARCH COMMAND
2305
2306                                          ;*WRITE FROM BUFFER WILL BE FILLED WITH EXPECTED DATA
2307
2308                                          ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
2309
2310                                          ;*COMPARE REGISTER BEFORE READ HEADER AND DATA
2311                                          ;*WITH REGISTERS AFTER COMMAND
2312
2313  013456  104031                  3$:     ERROR   31              ;READ HEADER AND DATA CAUSED
2314  013460  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
2315                                                                  ;GOOD DATA GIVES WHAT SHOULD
2316                                                                  ;BE THERE
2317                                                                  ;RECEIVED DATA GIVES WHAT WAS
2318                                                                  ;THERE AFTER COMMAND
2319
2320                                          ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
2321                                          ;*THE READ WAS GOOD
2322  013462                          4$:
2323
2324  013500  104053                  5$:     ERROR   53              ;READ HEADER AND DATA
2325  013502  000207                          RTS     PC              ;AFTER A SEARCH CAUSED
2326                                                                  ;AN ERROR
2327  013504                          6$:
2328
```

```
2329
2330
2331                                            ;*GET THE HEADS TO CYLINDER 10
2332
2333                                            ;*FILL REGISTERS FOR READ HEADER AND DATA TO BE DONE AFTER SEARCH
2334
2335                                            ;*SAVE REGISTERS FOR COMPARISON AFTER SEARCH
2336                                            ;*AND READ HEADER AND DATA
2337
2338                                            ;*NOW GIVE THE SEARCH COMMAND
2339  013652  012777  014036  166306            MOV     #3$,@RPVEC      ;INTERRUPT VECTOR SET TO 3$
2340  013670  013700  002334                    MOV     @#SERCH,R0      ;EXPECTED CONTENTS OF RHCS1
2341                                                                    ;IMMEDIATELY AFTER GO
2342  013674  052700  004301                    BIS     #DVA!RDY!IE!GO,R0 ;EXPECTED BITS IN RHCS1
2343  013700  012705  010500                    MOV     #MOL!DPR!VV,R5 ;EXPECTED BITS IN RHDS1
2344                                                                    ;IMMEDIATELY AFTER GO
2345
2346  013720  021100                            CMP     @R1,R0          ;IS RHCS1 GOOD
2347  013722  001413                            BEQ     1$              ;BRANCH IF GOOD
2348  013724  011137  001126                    MOV     @R1,@#$BDDAT    ;BAD DATA FOR RHCS1
2349  013730  010037  001124                    MOV     R0,@#$GDDAT     ;GOOD DATA
2350  013734  010137  004500                    MOV     R1,@#REGADR     ;FAILING REGISTER RHCS1
2351  013740  012737  000340  000036            MOV     #340,@#TRAPVEC+2 ;TRAP PRIORITY = 7
2352  013746  104021                            ERROR   21              ;DURING SEARCH COMMAND
2353                                                                    ;CONTENTS OF RHCS1 WAS
2354                                                                    ;NOT AS EXPECTED
2355  013750  000414                            BR      2$              ;IF LAST ERROR 21 OCCURRED
2356                                                                    ;THEN DO NOT CHECK RHDS1
2357                                                                    ;AS TOO MUCH TIME HAS
2358                                                                    ;PASSED
2359
2360  013752  021305                    1$:     CMP     @R3,R5          ;IS RHDS1 GOOD
2361  013754  001412                            BEQ     2$              ;BRANCH IF GOOD
2362  013756  011337  001126                    MOV     @R3,@#$BDDAT    ;BAD DATA FOR RHDS1
2363  013762  010537  001124                    MOV     R5,@#$GDDAT     ;GOOD DATA
2364  013766  010337  004500                    MOV     R3,@#REGADR     ;FAILING REGISTER RHDS1
2365  013772  012737  000340  000036            MOV     #340,@#TRAPVEC+2 ;TRAP PRIORITY = 7
2366  014000  104063                            ERROR   63              ;DURING SEARCH COMMAND
2367                                                                    ;CONTENTS OF RHDS1 WAS
2368                                                                    ;IN CORRECT
2369
2370  014002  013737  002350  004650    2$:     MOV     @#REFOR,@#TMP0  ;SAVE READ HEADER AND DATA
2371  014010  052737  000101  004650            BIS     #IE!GO,@#TMP0   ;INCLUDE INTERRUPT ENABLE, GO
2372  014016  012737  000001  004656            MOV     #1,@#TMP5       ;SAVE TRACK 0, SECTOR 1
2373
2374                                            ;*THIS IS ONLY A WAIT LOOP
2375
2376  014036  012737  000200  000036    3$:     MOV     #200,@#TRAPVEC+2 ;TRAP PRIORITY = 4
2377  014044  012737  000000  177776            MOV     #0,PS           ;SET PROSESSOR STATUS TO 0
2378  014052  013777  004656  166124            MOV     @#TMP5,@RHDST   ;SET DESIRED SECTOR/TRACK
2379                                                                    ;REGISTER TO SECTOR 1, TRACK 0
2380
2381  014066  013711  004650                    MOV     @#TMP0,@R1      ;FILL RHCS1 WITH READ COMMAND
2382                                                                    ;TOGETHER WITH INTERRUPT ENABLE
2383                                                                    ;AND GO
2384
```

```
2385                                            ;*TIME ALLOWED HERE IS CRITICAL ANY TIME ERROR
2386                                            ;*INDICATES WRONG SEARCH IN THE SEARCH COMMAND
2387
2388                                            ;*WRITE FROM BUFFER WILL BE FILLED WITH EXPECTED DATA
2389
2390                                            ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
2391
2392                                            ;*COMPARE REGISTER BEFORE READ HEADER AND DATA
2393                                            ;*WITH REGISTERS AFTER COMMAND
2394
2395
2396  014224  104031                    4$:     ERROR   31              ;READ HEADER AND DATA CAUSED
2397  014226  000207                            RTS     PC              ;IMPROPER REGISTER CHANGE
2398                                                                    ;GOOD DATA GIVES WHAT SHOULD
2399                                                                    ;BE THERE
2400                                                                    ;RECEIVED DATA GIVES WHAT WAS
2401                                                                    ;THERE AFTER COMMAND
2402
2403                                            ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
2404                                            ;*THE READ WAS GOOD
2405  014230                            5$:
2406
2407  014246  104053                    6$:     ERROR   53              ;READ HEADER AND DATA
2408  014250  000207                            RTS     PC              ;AFTER A SEARCH CAUSED
2409                                                                    ;AN ERROR
2410  014252                            7$:
2411
2412
2413
2414
```

```
                                        ;*      THE NEXT TEST REMOVES SECTOR 1 ON CYLINDER 0
                                        ;*      TRACK0 AND PUTS SECTOR 0 THERE.
                                        ;*      HENCE THE PACK IS UNFORMATTED FROM
                                        ;*      THIS POINT ON TO THE TEST WHEN SECTOR
                                        ;*      1 IS REPLACED.  IF TESTING IS STOPPED WITH
                                        ;*      AN ERROR IN THE SECTION OF THE PROGRAM BETWEEN
                                        ;*      THIS AND WHEN SECTOR 1 IS REPLACED THEN THE
                                        ;*      DISK BEING USED MAY HAVE BEEN UNFORMATTED
                                        ;*      IF THE LAST PASS OF THIS PROGRAM GIVES
                                        ;*      NO ERRORS IN THIS SECTION THEN THE DISK
                                        ;*      MAY NOT HAVE BEEN UNFORMATTED. HOWEVER IT
                                        ;*      IS RECOMMENDED THAT AFTER A PASS OF THIS
                                        ;*      PROGRAM THE DISK BE REFORMATTED.




                                                ;*FILL WRITE FROM BUFFER WITH HEADER

                                                ;*FILL READ INTO BUFFER WITH ALL ONES

                                                ;*WRITE HEADER AND DATA IS LOADED

                                                ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE




                                                ;*TIME IS NOT CRITICAL

                                                ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE

                                                ;*COMPARE ALL REGISTERS

014476  104027                          1$:     ERROR   27              ;WRITING HEADER AND DATA CAUSED
014500  000207                                  RTS     PC
                                                                        ;IMPROPER REGISTER CHANGE
                                                                        ;GOOD DATA GIVES WHAT
                                                                        ;SHOULD BE THERE
                                                                        ;RECEIVED DATA GIVES WHAT
                                                                        ;WAS THERE AFTER WRITE
```

```
2462
2463
2464                                        ;*NOW A READ DATA WILL BE DONE ON CYLINDER=0, SECTOR=1,
2465                                        ;*TRACK=0
2466                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
2467
2468  014502                          2$:
2469
2470                                        ;*READ COMMAND IS LOADED
2471
2472                                        ;*SAVE REGISTERS FOR COMPARISON AFTER READ DATA
2473
2474
2475
2476
2477                                        ;*TIME IS NOT CRITICAL
2478
2479                                        ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
2480
2481                                        ;*COMPARE ALL REGISTERS
2482  014732  104047                  3$:   ERROR   47                ;READING ON NON EXISTANT SECTOR
2483  014734  000207                        RTS     PC
2484                                                                  ;CAUSED AN ERROR
2485                                                                  ;GOOD DATA GIVES WHAT SHOULD
2486                                                                  ;BE THERE
2487                                                                  ;RECEIVED DATA GIVES WHAT
2488                                                                  ;WAS THERE AFTER READ
2489
2490                                        ;*READ DATA WILL BE COMPARED
2491
2492  014736                          4$:
2493  014754  104050                  5$:   ERROR   50                ;DATA READ FROM NON
2494  014756  000207                        RTS     PC                ;EXISTANT SECTOR CAUSED AN ERROR
2495  014760                          6$:
2496
```

```
2497
2498                                            ;*FILL WRITE FROM BUFFER WITH HEADER AND DATA
2499
2500                                            ;*FILL READ INTO BUFFER WITH ALL ONES
2501
2502                                            ;*WRITE HEADER AND DATA IS LOADED
2503
2504                                            ;*NOW SAVE REGISTERS FOR COMPARISON AFTER
2505                                            ;*WRITE HEADER AND DATA
2506
2507
2508
2509
2510
2511                                            ;*TIME IS NOT CRITICAL
2512
2513                                            ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
2514
2515                                            ;*COMPARE ALL REGISTERS
2516  015206  104027                    1$:     ERROR   27                  ;WRITING HEADER AND DATA CAUSED
2517  015210  000207                            RTS     PC                  ;IMPROPER REGISTER CHANGE
2518                                                                        ;GOOD DATA GIVES WHAT
2519                                                                        ;SHOULD BE THERE
2520                                                                        ;RECEIVED DATA GIVES WHAT
2521                                                                        ;WAS THERE AFTER WRITE
2522
```

```
                                        ;*NOW A WRITE DATA WILL BE DONE ON SAME CYLINDER, SECTOR
                                        ;*TRACK
015212                          2$:
                                        ;*FILL WRITE FROM BUFFER WITH DATA

                                        ;*WRITE DATA COMMAND IS LOADED

                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE DATA




                                        ;*TIME IS NOT CRITICAL

                                        ;*NOW CHANGE REGISTERS TO EXPECTED VALUE
015326  005737  004640                  TST     @#RH70          ;RUNNING ON RH70 ?
015332  001421                          BEQ     9$              ;IF NOT, SKIP NEXT RH70 CODE

015374  000416                          BR      10$             ;SKIP NEXT RH11 CODE

015376                          9$:

015432                          10$:

                                        ;*COMPARE ALL REGISTERS
015542  104052                  3$:     ERROR   52              ;WRITE DATA ON NON EXISTANT SECTOR
015544  000207                          RTS     PC              ;CAUSED IMPROPER REGISTER CHANGE
                                                                ;ATTEMPTED WRITE WAS ON
                                                                ;CYLINDER 0,SECTOR 1, TRACK 0
                                                                ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS THERE
                                                                ;AFTER COMMAND


                                        ;*READ HEADER AND DATA SECTOR 1, TRACK 0, CYLINDER 0
                                        ;*WILL BE ATTEMPTED
015546                          4$:
                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA

                                        ;*FILL READ INTO BUFFER WITH ALL ONES
                                        ;*FILL REGISTERS WITH READ HEADER AND DATA COMMAND

                                        ;*SAVE REGISTERS FOR COMPARISON AFTER READ
                                        ;*HEADER AND DATA



                                        ;*TIME IS NOT CRITICAL

                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE
```

```
                                        ;*COMPARE REGISTERS BEFORE READ HEADER AND DATA
                                        ;*WITH AFTER
                
016062  104031                  5$:   ERROR   31              ;READ HEADER AND DATA WITH
016064  000207                        RTS     PC              ;FORCED HEADER COMPARE ERROR
                                                              ;CAUSED ERROR
                                                              ;GOOD DATA GIVES WHAT SHOULD
                                                              ;BE THERE
                                                              ;RECEIVED DATA GIVES WHAT
                                                              ;WAS THERE AFTER READ
                
                                        ;*NOW COMPARE READ DATA
                                        ;*THE COMMAND READ ONLY 204 WORDS, 4 HEADER WORDS
                                        ;*AND 200 DATA WORDS
                
016066                            6$:
016104  104034                  7$:   ERROR   34              ;DATA READ FROM A FORCED
016106  000207                        RTS     PC              ;HEADER COMPARE ERROR IS
                                                              ;INCORRECT
                                                              ;GOOD DATA GIVES WHAT
                                                              ;THE READ HEADER AND DATA
                                                              ;SHOULD HAVE READ
                                                              ;BAD DATA GIVES WHAT
                                                              ;WAS IN BUFFER AFTER
                                                              ;READ COMMAND
016110                            8$:
```

```
                                        ;*GET HEADS TO CYLINDER 0



                                        ;*FILL REGISTERS FOR SEARCH
                                        ;*SAVE REGISTERS FOR COMPARISON AFTER SEARCH



                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE
                                        ;*COMPARE REGISTERS BEFORE SEARCH WITH AFTER SEARCH
016436  104047                  1$:     ERROR   47              ;SEARCH TO A NON-EXISTANT
016440  000207                          RTS     PC              ;SECTOR CAUSED IMPROPER
                                                                ;REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES
                                                                ;WHAT WAS THERE AFTER
                                                                ;SEARCH
016442                          2$:
```

```
                                        ;*FILL WRITE FROM BUFFER WITH HEADER
                                        ;*FILL WRITE FROM BUFFER WITH DATA
                                        ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA
                                        ;*AS WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
                                        ;*CAN BE MADE TO MAKE SURE THAT WRITE DID NOT
                                        ;*CHANGE WRITE FROM BUFFER

                                        ;*NOW THE WRITE HEADER AND DATA COMMAND WILL BE FILLED
                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE HEADER AND DATA


                                        ;*ONE REVOLUTION=16670 MICRO SEC, ONE SECTOR = 760 MICRO SEC
                                        ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
                                        ;*NOW COMPARE REGISTERS BEFORE WRITE HEADER AND DATA
                                        ;*WITH REGISTERS AFTER COMMAND

017002  104027                  1$:    ERROR   27              ;WRITE HEADER AND DATA
017004  000207                         RTS     PC              ;CAUSED IMPROPER REGISTER
                                                                ;CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER COMMAND

                                        ;*NOW WRITE FROM BUFFER WILL BE CHECKED TO SEE THAT
                                        ;*NOTHER GOT CHANGED
017006                           2$:

017024  104030                  3$:    ERROR   30              ;WRITE HEADER AND DATA
017026  000207                         RTS     PC              ;CHANGED WRITE FROM BUFFER

                                        ;*NOW A READ HEADER AND DATA COMMAND WILL BE GIVEN
                                        ;*READ INTO BUFFER IS FILLED WITH ONES
017030                           4$:

                                        ;*NOW FILL COMMAND
                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ HEADER AND DATA




                                        ;*CHEANGE SAVED REGISTERS TO EXPECTED VALUES

                                        ;*COMPARE REGISTER BEFORE READ HEADER AND DATA
                                        ;*WITH REGISTERS AFTER COMMAND
```

```
2699
2700
2701  017302  104031                  5$:     ERROR   31              ;READ HEADER AND DATA CAUSED
2702  017304  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
2703                                                                  ;GOOD DATA GIVES WHAT SHOULD
2704                                                                  ;BE THERE
2705                                                                  ;RECEIVED DATA GIVES WHAT WAS
2706                                                                  ;THERE AFTER COMMAND
2707
2708                                          ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
2709                                          ;*THE READ WAS GOOD
2710
2711  017306                          6$:
2712
2713  017324  104032                  7$:     ERROR   32              ;WRITE HEADER AND DATA
2714  017326  000207                          RTS     PC              ;FOLLOWED BY A READ HEADER
2715                                                                  ;AND DATA GAVE A READ ERROR
2716                                                                  ;ERROR MAY BE IN READ OR WRITE
2717  017330                          10$:
2718
2719
2720
```

```
2721
2722
2723
2724                                        ;*CHECK THE DRIVE TYPE AND THEN FILL THE
2725                                        ;*WRITE FROM BUFFER WITH APPROPRIATE HEADER
2726
2727  017350  005737  004636               TST     @#RP06  ;TEST FOR RP06 DRIVE
2728  017354  001411                       BEQ     5$              ;TREAT UNIT AS AN RP04
2729                                                        ;TREAT AS AN RP06
2730
2731  017376  000410                       BR      6$              ;CONTINUE WITH SET UP
2732
2733  017400                        5$:
2734  017420                        6$:                             ;CONTINUE
2735                                        ;*FILL WRITE FROM BUFFER WITH DATA
2736
2737
2738                                        ;*THE DRIVE TYPE IS CHECKED AND THE APPROPRIATE
2739                                        ;*WRITE HEADER AND DATA COMMAND IS LOADED
2740
2741
2742  017432  005737  004636               TST     @#RP06  ;TEST FOR RP06 DRIVE
2743  017436  001412                       BEQ     3$              ;TREAT UNIT AS RP04
2744                                                        ;TREAT UNIT AS RP06
2745
2746
2747  017462  000411                       BR      4$              ;CONTINUE WITH TESTING
2748  017464                        3$:
2749  017506                        4$:                             ;CONTINUE
2750
2751
2752                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER
2753                                        ;*WRITE HEADER AND DATA
2754
2755
2756
2757
2758
2759
2760
2761
2762                                        ;*CHANGE        THE SAVED REGISTERS TO EXPECTED VALUES
2763
2764                                        ;*AS EXCEPTION IS ASSERTED BEFORE RUN IS LATCHED,
2765                                        ;*RHWC,RHBA,RHCS1 & RHCS2 CANNOT BE PREDETERMINED -
2766                                        ;*THEY WILL VARY DEPENDING ON GATE DELAYS FOR DIFFERENT UNITS
2767
2768  017566  017737  162400  004512       MOV     @RHWC,@#SAVERE  ;RHWC IS UNPREDICTABLE
2769                                                        ;AS EXPLAINED ABOVE
2770  017574  017737  162374  004514       MOV     @RHBA,@#SAVERE+2;RHBA IS UNPREDICTABLE
2771                                                        ;AS EXPLAINED ABOVE
2772  017602  017737  162370  004516       MOV     @RHCS2,@#SAVERE+4;RHCS2 IS UNPREDICTABLE
2773                                                        ;AS EXPLAINED ABOVE
2774  017610  017737  162364  004520       MOV     @RHCS1,@#SAVERE+6;RHCS1 IS UNPREDICTABLE
2775                                                        ;AS EXPLAINED ABOVE
2776  017660  017737  162320  004524       MOV     @RHDST,@#SAVERE+12 ;RHDST IS INDETERMINATE
```

```
                                                        ;SO IT IS NOT CHECKED


                                        ;*COMPARE REGISTERS BEFORE ATTEMPTED WRITE WITH
                                        ;*CONTENTS AFTER ATTEMPTED WRITE WITH AN 'IAE' ERROR


017704  104054                  1$:   ERROR   54          ;ATTEMPTED OPERATION WITH
017706  000207                        RTS     PC          ;INVALID ADDRESS CAUSED
                                                          ;IMPROPER REGISTER CHANGE
                                                          ;GOOD DATA GIVES WHAT SHOULD
                                                          ;BE THERE
017710                            2$:                     ;RECEIVED DATA GIVES REGISTER
                                                          ;CONTENTS AFTER ATTEMPTED
                                                          ;WRITE HEADER AND DATA
```

```
                                        ;*FILL WRITE FROM BUFFER WITH DATA
                                        ;*WRITE DATA COMMAND WILL BE FILLED
                                        ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED WRITE DATA



                                        ;*CHANGE        SAVED REGISTERS TO EXPECTED VALUES


                                        ;*AS EXCEPTION IS ASSERTED BEFORE RUN IS LATCHED
                                        ;*RHWC,RHBA,RHCS1,RHCS2,  CANNOT BE PEREDETERMINED
                                        ;*THEY WILL VARY DEPENDING ON GATE DELAYS ON DIFFRENT UNITS

020044  017737  162122  004512           MOV     @RHWC,@#SAVERE  ;RHWC IS UNPREDICTABLE
                                                                 ;AS EXPLAINED ABOVE
020052  017737  162116  004514           MOV     @RHBA,@#SAVERE+2;RHBA IS UNPREDICTABLE
                                                                 ;AS EXPLAINED ABOVE
020060  017737  162112  004516           MOV     @RHCS2,@#SAVERE+4;RHCS2 IS UNPREDICTABLE
                                                                 ;AS EXPLAINED ABOVE
020066  017737  162106  004520           MOV     @RHCS1,@#SAVERE+6;RHCS1 IS UNPREDICTABLE
                                                                 ;AS EXPLAINED ABOVE
020114  017737  162064  004524           MOV     @RHDST,@#SAVERE+12 ;RHDST IS INDETERMINATE SO IT IS NOT CHECKED


                                        ;*COMPARE REGISTERS BEFORE ATTEMPTED WRITE DATA
                                        ;*WITH AFTER ATTEMPT, IAE SHOULD BE SET

020162  104054                  1$:     ERROR   54              ;ATTEMPTED WRITE DATA
020164  000207                          RTS     PC              ;WITH INVALID ADDRESS
                                                                ;CAUSED IMPROPER REGISTER
                                                                ;CHANGE
020166                          2$:                             ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER AFTER ATTEMPT
```

```
                                        ;*GET THE HEADS TO CYLINDER 10



                                        ;*FILL READ INTO BUFFER WITH 125252
                                        ;*THE READ HEADER AND DATA COMMAND IS FILLED
                                        ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED READ



                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES


                                        ;*AS EXCEPTION IS ASSERTED BEFORE RUN IS LATCHED
                                        ;*RHWC,RHBA,RHCS1,RHCS2,  CANNOT BE PEREDETERMINED
                                        ;*THEY WILL VARY DEPENDING ON GATE DELAYS ON DIFFRENT UNITS

020402  017737  161564  004512        MOV     @RHWC,@#SAVERE          ;RHWC IS UNPREDICTABLE
                                                                      ;AS EXPLAINED ABOVE
020410  017737  161560  004514        MOV     @RHBA,@#SAVERE+2        ;RHBA IS UNPREDICTABLE
                                                                      ;AS EXPLAINED ABOVE
020416  017737  161554  004516        MOV     @RHCS2,@#SAVERE+4       ;RHCS2 IS UNPREDICTABLE
                                                                      ;AS EXPLAINED ABOVE
020424  017737  161550  004520        MOV     @RHCS1,@#SAVERE+6       ;RHCS1 IS UNPREDICTABLE
                                                                      ;AS EXPLAINED ABOVE
020452  017737  161526  004524        MOV     @RHDST,@#SAVERE+12      ;RHDST IS INDETERMINATE SO IT IS NOT CHECKED


                                        ;*COMPARE REGISTERS BEFORE ATTEMPTED READ HEADER
                                        ;*AND DATA WITH AFTER ATTEMPTED READ

020520  104054                  1$:    ERROR   54              ;ATTEMPTED READ HEADER
020522  000207                         RTS     PC              ;AND DATA WITH INVALID
                                                                ;ADDRESS CAUSED IMPROPER
                                                                ;REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES
                                                                ;REGISTER CONTENTS
                                                                ;AFTER ATTEMPTED
                                                                ;READ
020524                           2$:
```

```
                                        ;*FILL READ INTO BUFFER WITH 125252
                                        ;*THE READ HEADER AND DATA COMMAND IS FILLED
                                        ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED READ
                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
020700  017737  161300  004524            MOV     @RHDST,@#SAVERE+12;RHDST IS UNPREDICTABLE
                                        ;*COMPARE REGISTERS BEFORE ATTEMPTED READ
                                        ;*DATA WITH AFTER ATTEMPTED READ DATA
020766  104054                  1$:       ERROR   54              ;ATTEMPTED READ
020770  000207                            RTS     PC              ;DATA WITH INVALID
                                                                ;ADDRESS CAUSED IMPROPER
                                                                ;REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES
                                                                ;REGISTERS CONTENTS
                                                                ;AFTER ATTEMPTED
                                                                ;READ
020772                              2$:
```

```
                                        ;*FILL WRITE FROM BUFFER WITH HEADER
                                        ;*FILL WRITE FROM BUFFER WITH DATA

                                        ;*THE FIRST WRITE OPERATION IS DONE
                                        ;*FILL WRITE HEADER AND DATA COMMAND




                                        ;*CHECK THE DRIVE TYPE AND DO THE
                                        ;*APPROPRIATE SECOND WRITE OPERATION

                                        ;*FILL WRITE FROM BUFFER WITH HEADER

021140  005737  004636                  TST     @#RP06  ;TEST FOR RP06 DRIVE
021144  001411                          BEQ     15$             ;TREAT DRIVE AS AN RP04
                                                                ;TREAT AS AN RP06

021166  000410                          BR      16$             ;CONTINUE WITH THE SECOND WRITE


021170                          15$:
021210                          16$:                            ;CONTINUE WRITE

                                        ;*FILL WRITE FROM BUFFER WITH DATA - 65125


                                        ;*CHECK THE DRIVE TYPE AND
                                        ;*FILL WRITE FROM BUFFER WITH APPROPRIATE NEXT HEADER

                                        ;*THIS IS A NON EXISTANT HEADER AND SHOULD NOT BE WRITTEN
                                        ;*SINCE 'AOE' SHOULD INHIBIT THE WRITE OPERATION


021222  005737  004636                  TST     @#RP06  ;TEST FOR RP06 DRIVE
021226  001411                          BEQ     17$             ;TREAT UNIT AS AN RP04
                                                                ;TREAT AS AN RP06

021250  000410                          BR      18$             ;CONTINUE WITH TEST


021252                          17$:
021272                          18$:                            ;CONTINUE
                                        ;*FILL WRITE FROM BUFFER WITH DATA FOR NEXT SECTOR


                                        ;*CHECK THE DRIVE TYPE AND DO THE APPROPRIATE
                                        ;*FILL WRITE HEADER AND DATA COMMAND
```

```
                                TST     @#RP06  ;TEST FOR RP06 DRIVE
021304  005737  004636          BEQ     7$              ;TREAT UNIT AS AN RP04
021310  001412                                          ;TREAT UNIT AS AN RP06

021334  000411                  BR      8$
021336                  7$:
021360                  8$:

                                ;*SAVE REGISTERS FOR COMPARISON AFTER WIRTE HEADER AND DATA

                                ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES

                                ;*CHECK DEVICE TYPE BEFORE SETTING UP 'RHCA' & 'RHCC'

021562  005737  004636          TST     @#RP06  ;TEST FOR RP06 DRIVE
021566  001411                  BEQ     9$              ;TREAT AS RP04
                                                        ;TREAT AS RP06

021610  000410                  BR      10$             ;CONTINUE WITH TEST
021612                  9$:
021632                  10$:                            ;CONTINUE WITH TEST

021632  017737  160346  004524  MOV     @RHDST,@#SAVERE+12 ;RHDST IS UNPREDICTABLE

                                ;*COMPARE REGISTERS BEFORE WRITE HEADER AND DATA WITH AFTER

021656  104055          1$:     ERROR   55              ;WRITING HEADER AND DATA WITH
021660  000207                  RTS     PC              ;EXPECTED ADDRESS OVERFLOW ERROR
                                                        ;CAUSED IMPROPER REGISTER
                                                        ;CHANGE
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT
                                                        ;WAS THERE AFTER WRITE
                                                        ;HEADER AND DATA
```

```
3018
3019                                        ;*NOW PREPARE TO DO A READ HEADER AND DATA
3020                                        ;*(THE FIRST READ OPERATION)
3021
3022                                        ;*CHECK THE DRIVE TYPE AND FILL
3023                                        ;*WRITE FROM BUFFER WITH APPROPRIATE EXPECTED HEADER
3024  021662                          2$:
3025
3026
3027  021666  005737  004636               TST     @#RP06  ;TEST FOR RP06 DRIVE
3028  021672  001411                       BEQ     19$             ;TREAT UNIT AS AN RP04
3029                                                                ;TREAT AS AN RP06
3030
3031  021714  000410                       BR      20$             ;CONTINUE WITH TEST
3032
3033
3034  021716                          19$:
3035  021736                          20$:                        ;CONTINUE
3036
3037
3038                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3039
3040                                        ;*FILL WRITE FROM BUFFER WITH 377 FROM WORDS 261 TO 266
3041
3042                                        ;*CLEAR READ INTO BUFFER
3043
3044
3045
3046                                        ;*CHECK THE DRIVE TYPE AND DO THE APPROPRIATE
3047                                        ;*FILL READ HEADER AND DATA COMMAND
3048
3049  022000  005737  004636               TST     @#RP06  ;TEST FOR RP06 DRIVE
3050  022004  001412                       BEQ     11$             ;TREAT UNIT AS AN RP04
3051                                                                ;TREAT UNIT AS AN RP06
3052  022030  000411                       BR      12$             ;CONTINUE
3053  022032                          11$:
3054  022054                          12$:                        ;CONTINUE WITH TESTING
3055
3056
3057                                        ;*SAVE REGISTERS FOR COMPARISON AFTER
3058                                        ;*READ HEADER AND DATA
3059
3060
3061
3062
3063
3064                                        ;*CHANGE        SAVED REGISTERS TO EXPECTED VALUES
3065
3066                                        ;*CHECK DRIVE TYPE BEFORE SETTING UP 'RHCA'
3067
3068  022230  005737  004636               TST     @#RP06  ;TEST FOR RP06 DRIVE
3069  022234  001405                       BEQ     13$             ;TREAT UNIT AS AN RP04
3070                                                                ;TREAT UNIT AS AN RP06
3071  022246  000404                       BR      14$             ;CONTINUE
3072  022250                          13$:
3073  022260                          14$:                        ;CONTINUE WITH TEST
```

```
3074
3075
3076  022266  017737  157712  004524          MOV     @RHDST,@#SAVERE+12 ;RHDST IS UNPREDICTABLE
3077
3078                                          ;*COMPARE REGISTERS BEFORE READ HEADER AND DATA WITH
3079                                          ;*REGISTERS AFTER COMMAND
3080
3081  022312  104055                  3$:     ERROR   55              ;READING HEADER AND DATA WITH
3082  022314  000207                          RTS     PC              ;EXPECTED ADDRESS OVERFLOW
3083                                                                  ;ERROR CAUSED IMPROPER
3084                                                                  ;REGISTER CHANGE
3085                                                                  ;GOOD DATA GIVES WHAT SHOULD
3086                                                                  ;BE THERE
3087                                                                  ;RECEIVED DATA GIVES WHAT
3088                                                                  ;WAS THERE AFTER COMMAND
3089
3090                                          ;*NOW COMPARE THE DATA READ
3091  022316                          4$:
3092
3093  022334  104056                  5$:     ERROR   56              ;DATA READ WITH AN EXPECTED
3094  022336  000207                          RTS     PC              ;ADDRESS OVERFLOW ERROR
3095                                                                  ;IS INCORRECT
3096                                                                  ;WORD NO 1 TO 260 SHOULD
3097                                                                  ;BE READ CORRECTLY
3098                                                                  ;WORD NO 261 TO 266 SHOULD
3099                                                                  ;NOT CHANGE DUE TO THE READ
3100  022340                          6$:
3101
```

```
                                        ;*NOW PREPARE TO READ CYLINDER 0, SECTOR 0, TRACK 0
                                        ;*TO SEE THAT NOTHING GOT WRITTEN ON THERE
                                        ;*WITH THE ADDRESS OVER FLOW BIT SET (AOE)

                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED HEADER

                                        ;*FILL READ INTO BUFFER WITH 377


                                        ;*FILL COMMAND FOR READ HEADER AND DATA

                                        ;*SAVE REGISTERS FOR COMPARISON AFTER READ




                                        ;*CHANGE        REGISTERS TO EXPECTED VALUE

022546                          ST22A:  ;COMPARE REGISTER BEFORE READ WITH AFTER

022564  104031                  4$:     ERROR   31              ;READ HEADER AND DATA ON
022566  000207                          RTS     PC              ;CYLINDER 0, SECTOR 0
                                                                ;TRACK 0 AFTER A FORCED
                                                                ;ADDRESS OVER FLOW ERROR
                                                                ;CAUSED IMPROPER REGISTER
                                                                ;CHANGE
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER READ
                                                                ;HEADER AND DATA
                                                                ;IF HEADER COMPARE ERROR
                                                                ;IS FOUND AND THE DATA
                                                                ;ERROR GIVES THE NEW
                                                                ;HEADER TO
                                                                ;CYLINDER 633/1457 (OCTAL)
                                                                ;THEN 'AOE' OVER FLOWED
                                                                ;INTO HERE

                                        ;*COMPARE DATA/READ
022570                          1$:

022606  104032                  2$:     ERROR   32              ;READ HEADER AND DATA
022610  000207                          RTS     PC              ;ON CYLINDER 0, TRACK 0
                                                                ;SECTOR 0 AFTER A FORCED
                                                                ;'AOE' ERROR CAUSED
                                                                ;AN ERROR
                                                                ;IF FIRST WORD IS
                                                                ;10633/11457 (OCTAL) THEN
                                                                ;'AOE' OVER FLOWED INTO HERE

022612                          3$:
```

```
3157
3158
3159                                    ;*FIRST WRITE HEADER AND DATA CYLINDER 0, TRACK 0, SECTOR 0
3160                                    ;*FILL WRITE FROM BUFFER WITH HEADER
3161
3162                                    ;*FILL WRITE FROM BUFFER WITH DATA
3163
3164                                    ;*FILL COMMAND
3165
3166
3167
3168
3169
3170
```

```
3171
3172                                        ;*NOW PREPARE TO WRITE WITH WRONG FORMAT
3173
3174                                        ;*FILL WRITE FROM BUFFER
3175
3176                                        ;*FILL WRITE DATA COMMAND
3177
3178                                        ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED WRITE DATA
3179                                        ;*WITH WRONG FORMAT
3180
3181
3182
3183
3184
3185                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE
3186  023074  005737  004640                 TST     @#RH70          ;RH70 CONTROLLER ?
3187  023100  001411                         BEQ     7$              ;IF NOT, SKIP NEXT RH70 CODE
3188
3189  023122  000410                         BR      8$              ;SKIP NEXT RH11 CODE
3190
3191
3192
3193  023124                          7$:
3194
3195  023144                          8$:
3196
3197
3198
3199                                        ;*COMPARE REGISTERS BEFORE WRITE DATA WITH AFTER ATTEMPT
3200
3201  023312  104057                  1$:     ERROR   57              ;ATTEMPTING TO WRITE DATA
3202  023314  000207                         RTS     PC              ;WITH WRONG FORMAT BIT CAUSED
3203                                                                  ;IMPROPER REGISTER CHANGE
3204                                                                  ;GOOD DATA GIVES WHAT SHOULD
3205                                                                  ;BE THERE
3206                                                                  ;RECEIVED DATA GIVES WHAT WAS
3207                                                                  ;THERE AFTER ATTEMPTED WRITE
3208
3209                                        ;*NOW PREPARE TO READ WITH CORRECT FORMAT TO CHECK
3210                                        ;*THAT NOTHING GOT WRITTEN
3211  023316                          2$:
3212                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3213
3214                                        ;*FILL READ INTO BUFFER WITH 125252
3215
3216                                        ;*FILL COMMAND TO READ DATA
3217
3218                                        ;*SAVE REGISTERS FOR COMPARISON AFTER NORMAL READ
3219
3220
3221
3222
3223
3224                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE
3225
3226
```

```
                                        ;*COMPARE REGISTERS BEFORE READ WITH AFTER
023540  104033          3$:   ERROR   33          ;READ DATA AFTER AN
023542  000207                RTS     PC          ;ATTEMPTED WRITE WITH WRONG
                                                  ;IMPROPER REGISTER CHANGE
                                                  ;FORMAT CAUSED
                                                  ;GOOD DATA GIVES WHAT SHOULD
                                                  ;BE THERE
                                                  ;RECEIVED DATA GIVES WHAT
                                                  ;WAS THERE AFTER READ

                                        ;*COMPARE DATA READ AFTER ATTEMPTED WRITE WITH
                                        ;*WRONG FORMAT BIT
023544                          4$:

023562  104034          5$:   ERROR   34          ;DATA READ AFTER AN ATTEMPT
023564  000207                RTS     PC          ;TO WRITE WITH WRONG FORMAT
                                                  ;WAS INCORRECT

023566                          6$:
```

```
                                        ;*FILL WRITE FROM BUFFER WITH 107070
                                        ;*FILL READ INTO BUFFER WITH 107070
                                        ;*FILL COMMAND TO READ WITH WRONG FORMAT
                                        ;*SAVE REGISTERS FOR COMPARAISON AFTER READ




                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE




                                        ;*COMPARE REGISTERS BEFORE WRITE DATA WITH AFTER ATTEMPT
024066  104057                  1$:    ERROR   57              ;ATTEMPTING TO READ DATA
024070  000207                         RTS     PC              ;WITH WRONG FORMAT BIT CAUSED
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD BE
                                                                ;THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS THERE
                                                                ;AFTER READ DATA

                                        ;*COMPARE READ INTO BUFFER TO CHECK THAT NOTHING WAS READ
024072                           2$:
024110  104034                  3$:    ERROR   34              ;ATTEMPT TO READ
024112  000207                         RTS     PC              ;WITH WRONG FORMAT BIT
                                                                ;CHANGED READ INTO BUFFER
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;BAD DATA GIVES WHAT WAS
                                                                ;THERE AFTER READ DATA

024114                           4$:
```

```
3291
3292  024134  012737  002200  004650          MOV     #RHCS1,@#TMP0   ;FIRST REGISTER TO BE TESTED
3293  024142  012737  000007  004656          MOV     #7,@#TMP5       ;NUMBER OF REGISTERS TO BE TESTED
3294
3295                                          ;*PREPARE TO WRITE HEADER AND DATA CYLINDER 1, TRACK 0, SECTOR 0
3296                                          ;*FILL WRITE FROM BUFFER WITH HEADER
3297
3298  024150                          ST22:
3299
3300                                          ;*FILL WRITE FROM BUFFER WITH DATA
3301                                          ;*FILL COMMAND
3302
3303
3304
3305
3306                                          ;*TIME IS NOT IMPORTANT
3307
3308                                          ;*NOW BRING THE HEADS TO CYLINDER 0
3309
3310
3311
3312
3313
3314
3315                                          ;*PREPARE FOR A READ DATA
3316
3317                                          ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA FROM READ
3318
3319                                          ;*FILL READ INTO BUFFER WITH DATA OTHER THAN WHAT IS EXPECTED
3320                                          ;*FILL READ DATA COMMAND
3321
3322                                          ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED WRITE
3323                                          ;*INTO A REGISTER WHILE THE READ IS GOING ON
3324
3325
3326
3327
3328  024500  013700  004650          MOV     @#TMP0,R0               ;SET UP R0 FOR WRITE
3329  024504  012730  002006          MOV     #BIT1!BIT2!BIT10,@(R0)+ ;ATTEMPT TO WRITE INTO
3330                                                                  ;REGISTERS DURING IMPLIED SEEK
3331  024510  010037  004650          MOV     R0,@#TMP0               ;SAVE OFF  R0
3332
3333                                          ;*NOW RMR IS SET BUT THE COMPLETION OF READ MUST BE
3334                                          ;*WAITED ON
3335
3336
3337                                          ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE
3338
3339                                          ;*COMPARE REGISTERS BEFORE READ DATA WITH REGISTERS
3340                                          ;*AFTER READ AND ATTEMPTED MODIFICATION OF REGISTER
3341
3342  024666                          2$:
3343  024670  013700  004650          MOV     @#TMP0,R0               ;GET REGISTER BEEING MODIFYED + 2 POINTER
3344  024674  014037  001122          MOV     -(R0),@#$BDADR          ;GET ADDRESS OF REGISTER BEING MODIFYED
3345  024700  104060                  ERROR   60                      ;ATTEMPTING TO MODIFY REGISTER
3346  024704  000207                  RTS     PC                      ;DURING A READ COMMAND CAUSED
```

```
3347                                                                    ;IMPROPER REGISTER CHANGE
3348                                                                    ;GOOD DATA GIVES WHAT SHOULD
3349                                                                    ;BE THERE
3350                                                                    ;RECEIVED DATA GIVES WHAT WAS
3351                                                                    ;THERE AFTER READ
3352                                            ;*COMPARE DATA READ
3353   024706                           3$:
3354
3355   024722  104034                  4$:     ERROR   34              ;DATA READ WITH AN ATTEMPTED
3356   024724  000207                          RTS     PC              ;MODIFICATION OF REGISTER
3357                                                                    ;DURING READ CAUSED ERROR
3358   024726  005337  004656          ST23:   DEC     @#TMP5          ;COUNT DOWN
3359   024732  001002                          BNE     1$              ;BRANCH IF 7 NOT DONE
3360   024740  000137  024150          1$:     JMP     @#ST22          ;JUMP TO BEGINING OF TEST
3361
```

```
                                                ;*PREPARE TO WRITE HEADER AND DATA

025000                                  ST24:

                                                ;*FILL WRITE FROM BUFFER WITH HEADER

                                                ;*FILL WRITE FROM BUFFER WITH DATA

                                                ;*FILL WRITE FROM BUFFER WITH NEXT SECTOR HEADER

                                                ;*FILL WRITE FROM BUFFER WITH WITH NEXT SECTOR DATA

                                                ;*NOW THE WRITE HEADER AND DATA COMMAND WILL BE FILLED



                                                ;*ONE REVOLUTION=16670 MICRO SEC, ONE SECTOR=760 MICRO SEC

                                                ;*NOW PREPARE FOR THE WRITE DATA COMMAND

                                                ;*FILL WRITE FROM BUFFER WITH 256 OF 2000 AND 4 OF 2001

                                                ;*FILL WRITE DATA COMMAND

                                                ;*SAVE REGISTERS FOR COMPARISON AFTER ATTEMPTED
                                                ;*REGISTER MODIFICATION DURING A WRITE DATA


025274  013700  004650                          MOV     @#TMP0,R0               ;SET R0 TO REG ADDRESS
025300  012730  002002                          MOV     #BIT1!BIT10,@(R0)+      ;ATTEMPT TO WRITE INTO A REGISTER
                                                                                ;DURING WRITE DATA
025304  010037  004650                          MOV     R0,@#TMP0               ;SAVE OFF NEW REG ADDRESS

                                                ;*NOW RMR MUST BE SET BUT THE COMPLETION OF
                                                ;*WRITE DATA MUST BE WAITED ON


                                                ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE

                                                ;*COMPARE REGISTERS BEFORE WRITE DATA WITH REGISTERS
                                                ;*AFTER WRITE AND ATTEMPTED MODIFICATION OF REGISTER

025462                                  2$:
025464  013700  004650                          MOV     @#TMP0,R0               ;GET REGISTER BEEING MODIFYED + 2 POINTER
025470  014037  001122                          MOV     -(R0),@#$BDADR          ;GET ADDRESS OF REGISTER BEING MODIFYED
025474  104060                                  ERROR   60                      ;ATTEMPTING TO MODIFY REGISTER
025500  000207                                  RTS     PC                      ;DURING A WRITE COMMAND CAUSED
                                                                                ;IMPROPER REGISTER GIVES WHAT SHOULD
```

```
3418                                                    ;GOOD DATA GIVES WHAT SHOULD
3419                                                    ;BE THERE
3420                                                    ;RECEIVED DATA GIVES WHAT WAS
3421                                                    ;THERE AFTER READ
3422                                    ;*CLEAR ALL ERROR FLAGS
3423  025502                      3$:
3424
3425                                    ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3426
3427                                    ;*NOW THE READ DATA COMMAND WILL BE FILLED
3428
3429                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
3430
3431
3432
3433
3434
3435                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
3436
3437
3438                                    ;*NOW COMPARE REGISTERS BEFORE READ DATA WITH
3439                                    ;*AFTER COMMAND
3440
3441  025766  104033                4$:     ERROR   33              ;READ DATA CAUSED IMPROPER REGISTER
3442  025770  000207                        RTS     PC              ;CHANGE
3443                                                                ;GOOD DATA GIVES WHAT SHOULD BE THERE
3444                                                                ;RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND
3445                                    ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE THAT READ
3446                                    ;*WAS GOOD
3447  025772                        5$:
3448
3449  026010  104034                6$:     ERROR   34              ;READ DATA ERROR AFTER A WRITE DATA
3450  026012  000207                        RTS     PC              ;WITH REGISTER MODIFICATION
3451                                                                ;WITHIN THE WRITE DATA
3452                                    ;*IF ALL 7 REGISTERS NOT COMPLETE THEN REPEAT
3453  026014  005337  004656        ST28:   DEC     @#TMP5          ;COUNT DOWN
3454  026020  001002                        BNE     1$              ;BRANCH IF 7 NOT DONE
3455  026026  000137  025000        1$:     JMP     ST24            ;JUMP TO BEGINING OF TEST
3456
```

```
3457
3458
3459                                            ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3460                                            ;*FILL READ INTM BUFFER WITH ALL ONES
3461                                            ;*NOW THE READ DATA COMMAND WILL BE FILLED
3462                                            ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
3463
3464
3465
3466
3467
3468  026172  012777  177777  154016           MOV     #-1,@RHAS       ;WRITE INTO RHAS THIS SHOULD
3469                                                                    ;NOT SET RMR
3470
3471                                            ;*TIME IS NOT IMPORTANT
3472
3473                                            ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
3474
3475                                            ;*NOW COMPARE REGISTERS BEFORE READ DATA WITH
3476                                            ;*AFTER COMMAND
3477
3478  026350  104033                  1$:       ERROR   33              ;READ DATA CAUSED IMPROPER REGISTER
3479  026352  000207                            RTS     PC              ;CHANGE
3480                                                                    ;GOOD DATA GIVES WHAT SHOULD BE THERE
3481                                                                    ;RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND
3482                                            ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE THAT READ
3483                                            ;*WAS GOOD
3484  026354                          2$:
3485
3486  026372  104034                  3$:       ERROR   34              ;READ DATA ERROR AFTER WRITING INTO
3487  026374  000207                            RTS     PC              ;RHAS DURING READ
3488
3489  026376                          4$:
```

```
3490
3491
3492  026416  005737  004640                      TST     @#RH70          ;RH70 CONTROLLER ?
3493  026422  001402                              BEQ     30$             ;SKIP NEXT IF NOT = 1
3494  026430                              30$:
3495
3496                                              ;*GENERATE ILLEGAL FUNCTION
3497
3498  026430  005037  001200                      CLR     @#$TMP1         ;GET READY TO MAKE ILLEGAL FUNCTION
3499  026434  012700  002322              1$:     MOV     #FUTABL,R0      ;GET POINTER TO BEGINNING OF COMMANDS
3500  026440  012705  000021                      MOV     #17.,R5         ;COUNTER (17 GOOD FUNCTIONS)
3501  026444  023720  001200              2$:     CMP     @#$TMP1,(R0)+   ;IS THIS A LEGAL FUNCTION
3502  026450  001004                              BNE     3$              ;BRANCH IF NOT LEGAL
3503  026452  062737  000002  001200              ADD     #2,@#$TMP1      ;MAKE ANOTHER FUNCTION
3504  026460  000765                              BR      1$              ;GET READY TO TEST NEW FUNCTION
3505  026462  005305                      3$:     DEC     R5              ;NOT LEGAL SO DECREMENT COUNTER
3506  026464  001367                              BNE     2$              ;BRANCH IF 17 NOT DONE
3507  026466  032737  000100  001200              BIT     #100,@#$TMP1    ;ALL BITS UP TO BIT #5 COMPARED?
3508  026474  001001                              BNE     20$             ;BRANCH OUT IF DONE
3509  026476  000402                              BR      19$             ;BRANCH TO CONTINUE
3510  026500  000137  027224              20$:    JMP     @#7$            ; DONE
3511  026504  013737  001200  002364      19$:    MOV     @#$TMP1,@#ILLEGL ;AN ILLEGAL FUNCTION IS FOUND
3512  026512  062737  000002  001200              ADD     #2,@#$TMP1      ;GET READY FOR NEW FUNCTION NEXT TIME
3513
3514                                              ;*ILLEGAL FUNCTION HAS BEEN FOUND
3515                                              ;*IT IS IN 'ILLEGL'
3516  026520  012737  026526  001110              MOV     #4$,@#$LPERR    ;ERROR RETURN POINT
3517
3518                                              ;*SAVE REGISTERS FOR COMPARISON AFTER GO
3519  026526                              4$:
3520  026532  005077  153434                      CLR     @RHWC           ;CLEAR WORD COUNT
3521  026536  005077  153432                      CLR     @RHBA           ;CLEAR BUS ADDRESS
3522  026542  023727  002364  000050              CMP     @#ILLEGL,#50    ;50 AND HIGHER FUNCTIONS ARE DATA
3523                                                                      ;FUNCTIONS WHICH WILL SET MXF AND TRE
3524  026550  103014                              BHIS    13$             ;BRANCH IF ILLEGL IS HIGHER THAN 50
3525  026552  012737  100000  027126              MOV     #SC,@#11$+12    ;EXPECTED VALUE OF RHCS1 SHOULD HAVE
3526                                                                      ;ONLY SC ADDED
3527  026560  005037  027150                      CLR @#12$+12            ;EXPECTED VALUE OF RHCS2 SHOULD HAVE
3528                                                                      ;NOTHING ADDED
3529  026564  005037  027154                      CLR     @#12$+16        ;NO BITS TO BE CLEARED IN RHCS2
3530  026570  005037  027164                      CLR     @#15$+6         ;RHBA SHOULD BE 0
3531  026574  005037  027174                      CLR     @#16$+6         ;CLEAR SAVED RHWC
3532  026600  000500                              BR      14$             ;BRANCH
3533  026602  022737  000064  002364  13$:        CMP     #64,@#ILLEGL    ;IS FUNCTION 64
3534  026610  001020                              BNE     17$             ;BRANCH IF NOT
3535  026612  012737  140000  027126              MOV     #SC!TRE,@#11$+12;SAVED RHCS1 SHOULD HAVE SC AND TRE
3536  026620  012737  000204  027164              MOV     #204,@#15$+6    ;RHBA SHOULD HAVE 204
3537  026626  012737  000102  027174              MOV     #102,@#16$+6    ;RHWC SHOULD HAVE 102
3538  026634  012737  001200  027150              MOV     #MXF!OR,@#12$+12;RHCS2 SHOULD HAVE MXF AND OR
3539  026642  012737  000100  027154              MOV     #IR,@#12$+16    ;RHCS2 SHOULD HAVE IR CLEARED
3540  026650  000454                              BR      14$             ;BRANCH
3541  026652  022737  000066  002364  17$:        CMP     #66,@#ILLEGL    ;IS FUNCTION 66
3542  026660  001030                              BNE     18$             ;BRANCH IF NOT
3543  026662  012777  177672  153302              MOV     #-70.,@RHWC     ; MOVE 70 INTO RHWC
3544  026670  012777  002370  153276              MOV     #WRFROM,@RHBA   ;FILL RHBA WITH WRFROM
3545  026676  012737  140000  027126              MOV     #SC!TRE,@#11$+12;SAVED RHCS1
```

```
                                        MOV     #WRFROM-<66.*2>,15$+6;RHBA
                                        MOV     #-4.,16$+6      ;SAVED RHWC
                                        MOV     #MXF!OR,@#12$+12 ;SAVED RHCS2
                                        CLR     @#12$+16        ;RHCS2
                                        MOV     #IR,@#12$+16    ;RHCS2 SHOULD HAVE IR CLEARED
                                        BR      14$             ;BRANCH
                                18$:    CLR     @RHWC           ;CLEAR RHWC
                                        CLR     @RHBA           ;CLEAR RHBA
                                        MOV     #SC!TRE,@#11$+12;RHCS1 SHOULD HAVE SC AND TRE
                                        CLR     @#15$+6         ;RHBA
                                        CLR     @#16$+6         ;RHWC
                                        MOV     #MXF,@#12$+12   ;RHCS2
                                        CLR     @#12$+16        ;RHCS2
                                14$:
                                        MOV     @#ILLEGL,-(SP)  ;GET ILLEGAL FUNCTION
                                        BIS     #GO!IE,(SP)     ;INCLUDE IE AND GO
                                        MOV     (SP)+,@R1       ;GO TO RHCS1 WITH ILLEGAL FUNCTION

                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE

                                        ;*RHCS1 WILL HAVE SC AND TRE ADDED IF FUNCTION IS GREATER THAN 50
                                11$:
                                        BIS     @#ILLEGL,@#SAVERE+6;INCLUDE ILLEGAL FUNCTION
                                                                   ;IN RHCS1
                                        ;*RHCS2 WILL HAVE NOTHING ADDED IF FUNCTION IS LESS THAN 50
                                12$:
                                15$:
                                16$:

                                        ;*NOW COMPARE REGISTERS AFTER GIVING AN ILLEGAL COMMAND

                                5$:     ERROR   51              ;GIVING ILLEGAL FUNCTION CAUSED
                                        RTS     PC              ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT
                                                                ;SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES REGISTER
                                                                ;CONTENTS AFTER ILLEGAL
                                                                ;FUNCTION WA GIVEN
                                6$:     JMP     @#1$            ;BRANCH FOR NEXT FUNCTION
                                7$:

                                10$:
```

```
3593
3594
3595                                                ;*THESE ARE REGULAR SETUPS
3596  027244  012777  177374  152720              MOV     #-260.,@RHWC      ;256 DATA WORDS 4 HEADER WORDS
3597  027252  012700  002370                      MOV     #WRFROM,R0        ;THESE TWO INSTRUCTIONS GETS
3598  027256  010077  152712                      MOV     R0,@RHBA          ;ADDR. OF WRFROM BUFFER INTO R0 AND
3599                                                                        ;BUS ADDRESS REGISTER
3600  027262  012710  010000                      MOV     #FMT22,(R0);      ;FORMAT=16 BIT WORDS
3601                                                                        ;CYLINDER=0
3602  027266  012720  000001                      MOV     #1,(R0)+          ;TRACK=0, SECTOR=1, KEYS=0
3603  027272  005020                              CLR     (R0)+             ;KEY1=0
3604  027274  005020                              CLR     (R0)+             ;KEY2=0
3605  027276  012705  000400                      MOV     #256.,R5          ;COUNTER
3606
3607                                                ;*SETUP DATA, WRITE HEADER & DATA, AND FORMAT OF THE WRITE
3608  027302  012720  177777          1$:         MOV     #-1,(R0)+         ;MOVE ALL ONES FOR DATA
3609  027306  005305                              DEC     R5
3610  027310  001374                              BNE     1$                ;BRANCH IF DATA NOT COMPLETE
3611  027312  012777  000001  152664              MOV     #1,@RHDST         ;TRACK=0 SECTOR=1
3612  027332  013711  002344                      MOV     @#WRIFOR,@R1      ;GET READY FOR WRITE HEADER AND
3613                                                                        ;DATA WITH 62 IN RHCS1
3614  027336  005037  004632                      CLR     @#ERFLG$          ;CLEAR ERROR FLAG
3615  027342  012777  010000  152640              MOV     #FMT22,@RHOF      ;FORMAT BIT=1 (16 BIT WORDS)
3616  027350  005077  152636                      CLR     @RHCA             ;CYLINDER =0
3617
3618                                                ;*SAVE REGISTERS FOR COMPARISON AFTER READ
```

```
                                        ;*GO TO WRITE HEADER AND DATA
                                        ;*BUT BEFORE GO, ONE INDEX PULSE IS GIVEN
                                        ;*TO CLEAR OUT THE SECTOR CLOCK COUNTER IN THE RH11
                                        ;*SO THAT NO SECTOR PULSES COME DURING THIS TEST

027366  013700  002220                  MOV     @#RHMR,R0       ;NOW R0 HAS MAINTENANCE REG. ADDR.
027372  012710  000001                  MOV     #DMD,@R0        ;SET DIAGNOSTIC MODE
027376  052710  000004                  BIS     #MINX,@R0       ;SET INDEX
027402  042710  000004                  BIC     #MINX,@R0       ;CLEAR INDEX THIS GIVES
                                                                ;ONE INDEX PULSE

027406  052777  000001  152564          BIS     #GO,@RHCS1      ;ISSUE THE 'GO' BIT TO THE RH11
027414  012737  000113  004642  RUNWAT: MOV     #75.,@#RUNCTR   ;LOAD 'RUN' LINE DELAY COUNTER
                                                                ;= APPROX 450 US ON 11/50 CPU WITH CORE
                                                                ;AND PROVIDES FOR TIME TO FILL THE SILO
027422  005337  004642          1$:     DEC     @#RUNCTR        ;COUNT DOWN ONCE
027426  001375                  BNE     1$              ;CONTINUE UNTIL = 0

                                        ;*ISSUE THE FIRST DIAGNOSTIC INDEX PULSE
027430  052710  000004                  BIS     #MINX,@R0       ;SET INDEX PULSE
027434  042710  000004                  BIC     #MINX,@R0       ;RESET INDEX

                                        ;*SECOND INDEX PULSE
027440  052710  000004                  BIS     #MINX,@R0       ;SET INDEX
027444  042710  000004                  BIC     #MINX,@R0       ;CLEAR INDEX

                                        ;*THIRD INDEX PULSE
027450  052710  000004                  BIS     #MINX,@R0       ;SET INDEX
027454  042710  000004                  BIC     #MINX,@R0       ;CLEAR INDEX


                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE


                                        ;*RHWC,RHBA AND OR AND IR BITS OF RHCS2 WILL NOT BE CHECKED
027566  017737  152400  004512          MOV     @RHWC,@#SAVERE  ;SAVED RHWC
027574  017737  152374  004514          MOV     @RHBA,@#SAVERE+2;SAVED RHBA
027602  017746  152370                  MOV     @RHCS2,-(SP)    ;GET RHCS2
027606  042716  177477                  BIC     #^C<IR!OR>,(SP) ;GET 'IR' & 'OR' STATES
027612  042737  000300  004516          BIC     #IR!OR,@#SAVERE+4;CLEAR 'IR' & 'OR' BITS
027620  052637  004516                  BIS     (SP)+,@#SAVERE+4;SET 'OR' & 'IR' AS REQUIRED


                                        ;*COMPARE REGISTERS BEFORE WRITE WITH RESULTS AFTER WRITE

027642  104071                  2$:     ERROR   71              ;FORCING OPI CAUSED
027644  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS THERE
                                                                ;AFTER 3 INDEX PULSES WERE ISSUED


027646  004737  033066          3$:     JSR     PC,@#CLDISK     ;CLEAR THE 'GO' BIT
```

```
3673
3674  027672  012737  000025  027726          MOV     #21.,@#1$+12    ;SET UP TO START FROM
3675  027700  012737  000025  027742          MOV     #21.,@#2$+6     ;SECTOR 21.
3676  027706  012737  000056  004652          MOV     #46.,@#TMP1     ;46 SECTORS TO COVER 3 TRACKS
3677
3678                                          ;*FILL WRITE FROM BUFFER WITH THE HEADER
3679  027714                          1$:
3680
3681                                          ;*NOW THE WRITE HEADER AND DATA COMMAND WILL BE SETUP
3682  027734                          2$:
3683
3694
3685
3686
3687                                          ;*ONE REVOLUTION=16670 MICRO SEC, ONE SECTOR=760 MICRO SEC
3688
3689                                          ;*NOW ONE MORE SECTOR HAS BEEN WRITTEN
3690                                          ;*'SC' WILL BE CHECKED TO MAKE SURE
3691                                          ;*NO ERRORS OCCURED
3692
3693  030030  017737  152144  002262          MOV     @RHCS1,@#CS1    ;GET RHCS1
3694  030036  032737  100000  002262          BIT     #SC,@#CS1       ;IS 'SC' SET ?
3695  030044  001403                          BEQ     3$              ;BRANCH IF "SPECIAL CONDITION" NOT SET
3696  030046  004737  035010                  JSR     PC,@#PUTREG     ;READ & SAVE  ALL RH11 & RP04 REGISTERS
3697  030052  104072                          ERROR   72              ;THERE WAS AN UNDEFINED ERROR AFTER
3698                                                                  ;A WRITE HEADER AND DATA
3699
3700                                          ;*A SECTOR HAS BEEN FORMATTED NOW,
3701                                          ;*THE HARDWARE WILL BE CLEARED AND
3702                                          ;*CHANGES WILL BE MADE TO FORMAT NEXT SECTOR.
3703
3704  030054                          3$:
3705  030060  013705  027726                  MOV     @#1$+12,R5      ;GET SECTOR TRACK WORD
3706  030064  005205                          INC     R5              ;+ 1
3707  030066  122705  000026                  CMPB    #22.,R5         ;IS IT 22 SECTORS (WHOLE TRACK DONE) ?
3708  030072  001405                          BEQ     4$              ;YES...DO NEXT TRACK
3709  030074  010537  027726                  MOV     R5,@#1$+12      ;NO...RESTORE SECTOR TRACK FOR DATA
3710  030100  010537  027742                  MOV     R5,@#2$+6       ;RESTORE SECTOR TRACK FOR "RUN" ROUTINE
3711  030104  000410                          BR      5$              ;CHECK FOR 46 SECTORS COMPLETED
3712
3713  030106  105037  027726          4$:     CLRB    @#1$+12         ;SET SECTOR = 0 FOR DATA WRITTEN
3714  030112  105237  027727                  INCB    @#1$+13         ;INCR TRACK FOR DATA WRITTEN
3715  030116  105037  027742                  CLRB    @#2$+6          ;SET SECTOR = 0 FOR "RUN" ROUTINE
3716  030122  105237  027743                  INCB    @#2$+7          ;INCR TRACK FOR THE "RUN" ROUTINE
3717
3718  030126  005337  004652          5$:     DEC     @#TMP1          ;ARE 46 SECTORS DONE ?
3719  030132  001270                          BNE     1$              ;CONTINUE FORMATTING IF NOT
3720
3721  030134                          6$:                             ;GO ON TO NEXT TEST IF SO
```

```
3722
3723
3724  030154  012737  000025  030210          MOV     #21.,@#1$+12    ;SET UP TO START FROM
3725  030162  012737  000025  030224          MOV     #21.,@#2$+6     ;SECTOR 21.
3726  030170  012737  000056  004652          MOV     #46.,@#TMP1     ;46 SECTORS TO COVER 3 TRACKS
3727
3728                                          ;*FILL WRITE FROM BUFFER WITH HEADER
3729  030176                          1$:
3730
3731                                          ;*NOW THE WRITE HEADER AND DATA COMMAND WILL BE FILLED
3732  030216                          2$:
3733
3734
3735
3736
3737                                          ;*ONE REVOLUTION=16670 MICRO SEC, ONE SECTOR=760 MICRO SEC
3738
3739                                          ;*NOW ONE MORE SECTOR HAS BEEN WRITTEN
3740                                          ;*'SC' WILL BE CHECKED TO MAKE SURE
3741                                          ;*NO ERRORS OCCURED
3742
3743  030312  017737  151662  002262          MOV     @RHCS1,@#CS1    ;GET RHCS1
3744  030320  032737  100000  002262          BIT     #SC,@#CS1       ;IS 'SC' SET ?
3745  030326  001405                          BEQ     3$              ;BRANCH IF "SPECIAL CONDITION" NOT SET
3746  030330  004737  035010                  JSR     PC,@#PUTREG     ;READ & SAVE  ALL RH11 & RP04 REGISTERS
3747  030334  104072                          ERROR   72              ;THERE WAS AN UNDEFINED ERROR AFTER
3748                                                                  ;A WRITE HEADER AND DATA
3749
3750                                                                  ;GO ON TO NEXT TEST
3751
3752                                          ;*ONE SECTOR HAS BEEN FORMATTED NOW,
3753                                          ;*THE HARDWARE WILL BE CLEARED AND
3754                                          ;*CHANGES WILL BE MADE TO FORMAT NEXT SECTOR.
3755
3756  030342                          3$:
3757  030346  013705  030210                 MOV     @#1$+12,R5      ;GET SECTOR TRACK WORD
3758  030352  005205                          INC     R5              ;+ 1
3759  030354  122705  000026                  CMPB    #22.,R5         ;IS IT 22 (WHOLE TRACK) ?
3760  030360  001405                          BEQ     4$              ;YES...DO NEXT TRACK
3761  030362  010537  030210                  MOV     R5,@#1$+12      ;NO...RESTORE SECTOR TRACK FOR DATA WRITTEN
3762  030366  010537  030224                  MOV     R5,@#2$+6       ;RESTORE SECTOR TRACK FOR "RUN" ROUTINE
3763  030372  000410                          BR      5$              ;CHECK FOR 46 SECTORS COMPLETED
3764
3765  030374  105037  030210          4$:     CLRB    @#1$+12         ;SET SECTOR = 0 FOR DATA WRITTEN
3766  030400  105237  030211                  INCB    @#1$+13         ;INCR TRACK FOR THE "RUN" ROUTINE
3767  030404  105037  030224                  CLRB    @#2$+6          ;SET SECTOR = 0 FOR DATA WRITTEN
3768  030410  105237  030225                  INCB    @#2$+7          ;INCR TRACK FOR THE "RUN" ROUTINE
3769
3770  030414  005337  004652          5$:     DEC     @#TMP1          ;ARE 46 SECTORS DONE ?
3771  030420  001266                          BNE     1$              ;CONTINUE IF NOT
3772
3773
3774                                          ;*NOW 46 SECTORS HAVE BEEN FORMATTED
3775
3776                                          ;*READ HEADER AND DATA FOR 46 SECTORS=11960 WORDS
3777                                          ;*WITH BUS ADDRESS INHIBITED
```

```
3778
3779
3780                                        ;*FILL READ HEADER AND DATA COMMAND
3781
3782
3783                                        ;*TIME IS NOT IMPORTANT
3784
3785
3786                                        ;*NOW THAT ALL 11960 WORDS HAVE BEEN READ
3787                                        ;*'OPI' WILL BE CHECKED TO BE NOT SET
3788
3789  030522  017737  151454  002264        MOV     @RHER1,@#ER1    ;GET RHER1
3790  030530  032737  020000  002264        BIT     #OPI,@#ER1      ;IS 'OPI' SET ?
3791  030536  001403                        BEQ     6$              ;CHECK 'SC' IF NOT
3792  030540  004737  035010                JSR     PC,@#PUTREG     ;READ & SAVE ALL RH11 & RP04 REGISTERS
3793  030544  104074                        ERROR   74              ;READ HEADER AND DATA
3794                                                                ;OVER 3 INDEX PULSES
3795                                                                ;CAUSED 'OPI' TO SET
3796
3797                                        ;*'SC' WILL BE CHECKED
3798
3799  030546  017737  151426  002262  6$:   MOV     @RHCS1,@#CS1    ;GET RHCS1
3800  030554  032737  100000  002262        BIT     #SC,@#CS1       ;IS 'SC' SET ?
3801  030562  001403                        BEQ     7$              ;CONTINUE TESTING IF NOT
3802  030564  004737  035010                JSR     PC,@#PUTREG     ;READ & SAVE ALL RH11 & RP04 REGISTERS
3803  030570  104072                        ERROR   72              ;READ HEADER AND DATA
3804                                                                ;FOR 11960 WORDS, THAT IS OVER THREE
3805                                                                ;INDEX PULSES, CAUSED AN UNDEFINED ERROR
3806
3807  030572                          7$:                           ;CONTINUE WITH THE NEXT TEST
```

```
3808
3809
3810
3811                                                ;*THE FOLLOWING CLEARS ARE TO INITIALIZE TEST FROM CYLINDER 0
3812  030612  005037  030720                       CLR     @#1$+12         ;START WITH SECTOR/TRACK = 0
3813  030616  005037  030736                       CLR     @#2$+10         ;START WITH DATA = 0
3814  030622  005037  030746                       CLR     @#3$+6          ;START WITH 0 FOR COMMAND
3815
3816  030626  012737  000023  001200               MOV     #19.,@#STMP1    ;19 TRACKS TO BE WRITTEN
3817
3818                                                ;*THIS GETS THE HEADS TO CYLINDER 0
3819
3820
3821
3822
3823
3824
3825
3826                                                ;*FILL WRITE FROM BUFFER WITH HEADER
3827  030706                            1$:
3828
3829                                                ;*FILL WRITE FROM BUFFER WITH DATA
3830  030726                            2$:
3831
3832                                                ;*THE WRITE HEADER AND DATA COMMAND WILL BE FILLED
3833  030740                            3$:
3834
3835
3836
3837
3838                                                ;*ONE REVOLUTION = 16670 MICRO SEC., ONE SECTOR = 760
3839                                                ;*MICRO SEC. MAX TIME ALLOWED = ONE REVOLUTION + HEAD
3840                                                ;*SWITCH + 2 SECTORS, MIN TIME ALLOWED = SECTOR (FIRST CASE).
3841                                                ;*IF THERE IS A FAILURE HERE HALT PROGRAM AFTER ERROR WITH
3842                                                ;*SWITCH 15 AND SEE CURRENT CYLINDER REGISTER TO DETERMINE
3843                                                ;*WHAT CYLINDER IS FAILING
3844
3845
3846                                                ;*NOW SECTOR 0 OF ONE TRACK HAS BEEN WRITTEN CHECK COMPOSIT
3847                                                ;*ERROR BIT TO BE SURE NO ERRORS HAPPENED
3848
3849                                                ;*SAVE REGISTERS IN SAVE TABLE
3850  031030  004737  035010                       JSR     PC,@#PUTREG
3851
3852  031034  032737  040000  002304               BIT     #ERR,@#DS1      ;ANY DISK ERRORS
3853  031042  001004                               BNE     9$              ;BRANCH IF YES
3854  031044  032737  040000  002262               BIT     #TRE,@#CS1      ;ANY RH ERRORS
3855  031052  001401                               BEQ     4$              ;BRANCH IF NO
3856
3857  031054  104066                      9$:      ERROR   66              ;SOME ERRORS OCCURRED
3858                                                                       ;WHILE DOING WRITE HEADER
3859                                                                       ;AND DATA
3860
3861                                                ;*THE FOLLOWING 3 ADDS SETS UP FOR NEXT TRACK WRITING
3862
3863  031056  062737  000400  030720  4$:      ADD     #400,@#1$+12    ;NEXT TRACK FOR HEADER
```

```
3864  031064  062737  000040  030736          ADD     #40,@#2$+10     ;NEXT TRACK FOR DATA
3865  031072  062737  000400  030746          ADD     #400,@#3$+6     ;NEXT TRACK FOR COMMAND
3866
3867  031100  005337  001200                  DEC     @#$TMP1         ;COUNT 19 TRACKS
3868  031104  001300                          BNE     1$
3869
3870                                          ;*THE FOLLOWING CLEARS SETS UP FOR READ HEADER AND DATA
3871  031106  005037  031166                  CLR     @#SST3+12       ;START WITH SECTOR/TRACK = 0
3872  031112  005037  031204                  CLR     @#SST4+10       ;START WITH DATA = 0
3873  031116  005037  031214                  CLR     @#SST5+6        ;START WITH 0 FOR COMMAND
3874
3875
3876                                          ;*SET UP FOR READ HEADER AND DATA
3877  031126  012737  000023  001200  SST1:   MOV     #19.,@#$TMP1    ;19 TRACKS TO BE READ
3878
3879                                          ;*FILL READ INTO BUFFER WITH ALL ONES
3880  031134                          SST2:
3881  031146  013737  031134  001110          MOV     @#SST2,@#$LPERR ;SET LOOP POINT
3882
3883                                          ;*FILL WRITE FROM BUFFER WITH EXPECTED HEADER
3884  031154                          SST3:
3885
3886                                          ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3887  031174                          SST4:
3888
3889                                          ;*FILL COMMAND FOR READ HEADER AND DATA
3890  031206                          SST5:
3891
3892
3893
3894
3895
3896                                          ;*NOW SECTOR 0 OF ONE TRACK HAS BEEN READ CHECK COMPOSIT
3897                                          ;*ERROR BIT TO BE SURE NO ERROR HAPPENED
3898
3899                                          ;*SAVE REGISTERS IN SAVE TABLE
3900  031276  004737  035010                  JSR     PC,@#PUTREG
3901
3902  031302  032737  040000  002304          BIT     #ERR,@#DS1      ;ANY DISK ERRORS
3903  031310  001004                          BNE     10$             ;BRANCH IF YES
3904  031312  032737  040000  002262          BIT     #TRE,@#CS1      ;ANY RH ERRORS
3905  031320  001401                          BEQ     11$             ;BRANCH IF NO
3906
3907  031322  104066                  10$:    ERROR   66              ;SOME ERRORS OCCURRED
3908                                                                  ;WHILE DOING READ
3909                                                                  ;HEADER AND DATA
3910
3911                                          ;*NOW THE READ DATA WILL BE COMPARED DATA IN EACH SECTOR
3912                                          ;*IS UNIQUE IF PROGRAM IS HALTED ON ERROR THEN LOOK AT
3913                                          ;*RHDST TO GET WHAT TRACK IS IN ERROR. LOOKING AT THE DATA
3914                                          ;*BITS NO 4,5,6,7,8 IN GOOD DATA ALSO GIVES TRACK NUMBER
3915                                          ;*IN GOOD DATA ALSO GIVES TRACK NUMBER
3916
3917  031324                          11$:
3918
3919                                                                  ;BITS 4,5,6,7,8
```

```
3920  031342  104067                  12$:    ERROR   67              ;READ HEADER AND DATA
3921  031344  000207                          RTS     PC              ;ERROR
3922                                                                  ;HEAD SELECTION ERROR
3923                                                                  ;DATA READ GIVES NATURE
3924                                                                  ;OF ERROR
3925                                                                  ;EXCEPT FOR THE
3926                                                                  ;FOUR  HEADER WORDS
3927                                                                  ;THE BITS 4,5,6,7,8
3928                                                                  ;GIVE THE TRACK NUMBER
3929
3930                                          ;*NOW INCREMENT TO READ NEXT TRACK
3931
3932  031346  062737  000400  031166  13$:    ADD     #400,@#SST3+12  ;NEXT TRACK FOR HEADER
3933  031354  062737  000040  031204          ADD     #40,@#SST4+10   ;NEXT TRACK FOR DATA
3934  031362  062737  000400  031214          ADD     #400,@#SST5+6   ;NEXT TRACK FOR COMMAND
3935
3936  031370  005337  001200                  DEC     @#$TMP1         ;COUNT 19 TRACKS
3937  031374  001001                          BNE     5$
3938  031400  000137  031134          5$:     JMP     @#SST2          ;JUMP BACK
```

```
3939
3940
3941
3942                                            ;*SET UP TO INITIALIZE TEST FROM CYLINDER 0, TRACK 0,
3943                                            ;*SECTOR 0
3944  031436  012737  010000  031550            MOV     #10000,@#1$+10  ;CYLINDER HEADER DATA
3945  031444  005037  031570                    CLR     @#2$+10         ;DATA
3946  031450  005037  031576                    CLR     @#3$+4          ;CYLINDER COMMAND RHCA
3947
3948                                            ;*THIS IS TO GET THE HEADS TO CYLINDER ZERO
3949
3950
3951
3952                                            ;*THE DRIVE TYPE IS CHECKED AND THE APPROPRIATE MAX.
3953                                            ;*CYLINDER DIFFERENCE IS SET UP
3954
3955  031510  005737  004636                    TST     @#RP06  ;TEST FOR RP06 DRIVE
3956  031514  001404                            BEQ     14$             ;TREAT UNIT AS AN RP04
3957                                                                    ;TREAT AS AN RP06
3958
3959  031516  012737  001001  001200            MOV     #513.,@#$TMP1   ;513 CYLINDERS
3960  031524  000403                            BR      15$             ;CONTINUE
3961
3962  031526  012737  000401  001200  14$:     MOV     #257.,@#$TMP1   ;257 CYLINDERS
3963  031534                            15$:                            ;CONTINUE WITH TEST
3964
3965
3966                                            ;*FILL WRITE FROM BUFFER WITH HEADER
3967  031540                            1$:
3968
3969                                            ;*FILL WRITE FROM BUFFER WITH DATA
3970  031560                            2$:
3971
3972                                            ;*THE WRITE HEADER AND DATA COMMAND WILL BE LOADED
3973  031572                            3$:
3974
3975
3976
3977
3978                                            ;*ONE REVOLUTION = 16670 MICRO SECONDS, ONE SECTOR = 760
3979                                            ;*MICRO SECONDS, ONE SEEK = 7000 MICRO SECONDS.
3980                                            ;*MAX TIME = 1 REVOLUTION + 1 SEEK + 2 SECTORS
3981                                            ;*MIN TIME = 1 SECTOR
3982
3983
3984
3985                                            ;*NOW ONE SECTOR WRITE IS COMPLETE.  CHANGES WILL BE MADE
3986                                            ;*FOR THE NEXT SECTOR, THEN THE ABOVE WILL BE REPEATED
3987  031662  005237  031550                    ;*UNTIL CYLINDER 256./512. IS REACHED
3988  031666  005237  031570                    INC     @#1$+10         ;CYLINDER HEADER DATA
3989  031672  005237  031576                    INC     @#2$+10         ;DATA
3990  031676  005337  001200                    INC     @#3$+4          ;CYLINDER COMMAND (RHCA)
3991  031702  001316                            DEC     @#$TMP1         ;COUNT DOWN FOR 256./512. CYLINDERS
3992                                            BNE     1$              ;DO NEXT WRITE IF 256./512. NOT DONE
3993
3994                                            ;*NOW ALL 256./512. CYLINDERS HAVE CYLINDER NUMBER WRITTEN
                                                ;*AS DATA ON SECTOR 0, TRACK 0. NOW A RECALIBRATE, FOLLOWED
```

```
 3995                                          ;*BY READ HEADER AND DATA, THEN A CHECK WILL BE DONE ON
 3996                                          ;*CYLINDER 0,1,2,4,8,16,32,64,128,256,512, AND 0
 3997
 3998  031704  013737  031734  001110          MOV     @#4$,@#$LPERR   ;LOOP ON ERROR
 3999  031712  005037  001200                  CLR     @#$TMP1         ;CYLINDER COUNTER
```

```
4000
4001                                            ;*INITIALIZE, RECALIBRATE, AND READ CYLINDERS
4002
4003                                            ;*SETUP FOR CYLINDER 0
4004  031716  012737  010000  032016            MOV     #10000,@#5$+10  ;CYLINDER HEADER (DATA)
4005  031724  005037  032036                    CLR     @#6$+10         ;DATA
4006  031730  005037  032044                    CLR     @#7$+4          ;CYLINDER COMMAND (RHCA)
4007  031734                            4$:
4008
4009
4010
4011                                            ;*CLEAR READ INTO BUFFER WITH ALL ONES
4012
4013                                            ;*FILL WRITE FROM BUFFER WITH EXPECTED HEADER
4014
4015  032006                            5$:
4016  032026                            6$:
4017
4018                                            ;*FILL READ HEADER AND DATA COMMAND
4019  032040                            7$:
4020
4021
4022
4023                                            ;*ONE SECTOR = 760 MICRO SECONDS, ONE REVOLUTION =
4024                                            ;*16670 MICRO SECONDS, MAX SEEK = 52000 MICRO SECONDS
4025                                            ;*MAX TIME = ONE REV + 1 SEEK + 1 SECTOR
4026                                            ;*MIN TIME = 1 SECTOR
4027
4028
4029
4030                                            ;*CHECK READ WORDS AS ALL READ COMMANDS HAVE BEEN CHECKED
4031
4032                                            ;*(DATA ERRORS MAY IMPLY "IMPLIED SEEK" ERRORS)
4033
4034
4035
4036  032146  104070                    8$:     ERROR   70              ;READ HEADER AND DATA ERROR
4037  032150  000207                            RTS     PC              ;DATA GIVES EXPECTED CYLINDER
4038
4039                                            ;*NOW ONE CYLINDER HAS BEEN CHECKED.  CHANGES WILL BE MADE
4040                                            ;*TO READ THE NEXT CYLINDER AND THE ABOVE SECTOR READ WILL BE
4041                                            ;*REPEATED
4042
4043  032152  005737  001200            9$:     TST     @#$TMP1         ;IS IT ZERO ?
4044  032156  001003                            BNE     10$             ;BRANCH IF NOT ZERO
4045  032160  005237  001200                    INC     @#$TMP1         ;ADD ONE IF = 0
4046  032164  000416                            BR      11$             ;PUT ONE IN CYLINDER
4047
4048  032166  005737  004636            10$:    TST     @#RP06  ;TEST FOR RP06 DRIVE
4049  032172  001404                            BEQ     16$             ;TREAT UNIT AS AN RP04
4050                                                                    ;TREAT AS AN RP06
4051
4052  032174  022737  001000  001200            CMP     #512.,@#$TMP1   ;IS IT PASSED 512 CYLINDERS YET ?
4053  032202  000403                            BR      17$             ;CONTINUE
4054  032204  022737  000400  001200    16$:    CMP     #256.,@#$TMP1   ;IS IT PASSED 256 CYLINDERS YET ?
4055  032212                            17$:                            ;CONTINUE
```

```
4056
4057  032212  101421                          BLOS    12$                 ;YES, SO GO TO ZERO
4058  032214  063737  001200  001200          ADD     @#$TMP1,@#$TMP1     ;DOUBLE THE CYLINDER
4059  032222  013737  001200  032036  11$:    MOV     @#$TMP1,@#6$+10     ;MAKE CYLINDER ADDRESS THE DATA
4060  032230  013746  001200                  MOV     @#$TMP1,-(SP)       ;GET CYLINDER NUMBER
4061  032234  052716  010000                  BIS     #FMT22,(SP)         ;INCLUDE FORMAT BIT
4062  032240  012637  032016                  MOV     (SP)+,@#5$+10       ;HEADER DATA (CYLINDER)
4063  032244  013737  001200  032044          MOV     @#$TMP1,@#7$+4      ;CYLINDER COMMAND (RHCA)
4064  032252  000137  031734                  JMP     @#4$                ;RETURN TO RECALIBRATE
4065
4066  032256  005737  004636          12$:    TST     @#RP06  ;TEST FOR RP06 DRIVE
4067  032262  001405                  BEQ     18$                         ;TREAT UNIT AS AN RP04
4068                                                                      ;TREAT AS AN RP06
4069
4070  032264  022737  002000  001200          CMP     #1024.,@#$TMP1      ;512 DONE YET ?
4071  032272  001421                  BEQ     13$                         ;OUT ---------------------------->
4072  032274  000404                  BR      19$                         ;CONTINUE
4073  032276  022737  001000  001200  18$:    CMP     #512.,@#$TMP1       ;256 DONE YET ?
4074  032304  001414                  BEQ     13$                         ;OUT ---------------------------->
4075  032306                          19$:                                ;CONTINUE
4076
4077  032306  063737  001200  001200          ADD     @#$TMP1,@#$TMP1     ;DOUBLE THE CYLINDER
4078  032314  012737  010000  032016          MOV     #10000,@#5$+10      ;CYLINDER HEADER DATA
4079  032322  005037  032036                  CLR     @#6$+10             ;DATA
4080  032326  005037  032044                  CLR     @#7$+4              ;CYLINDER COMMAND (RHCA)
4081  032332  000137  031734                  JMP     @#4$                ;RETURN TO THE RECALIBRATE
4082
4083  032336                          13$:                                ;END OF TEST
```

```
4084
4085  032346  012737  000000  177776          MOV     #0,PS           ;REINSTATE PS TO 0
4086  032434  013746  004616                  MOV     @#UNIT,-(SP)    ;GET READY TO TYPE UNIT NUMBER
4087  032440  104405                          TYPDS
4088  032454  013746  001112                  MOV     @#$ERTTL,-(SP)  ;GET READY TO TYPE NUMBER OF ERRORS
4089  032460  104405                          TYPDS
4090  032462  005037  001112                  CLR     @#$ERTTL        ;CLEAR TOTAL NUMBER OF ERRORS
4091  032466  005037  001102                  CLR     @#$TSTNM        ;CLEAR TEST NUMBER
4092  032472  005737  004626                  TST     @#SELECT        ;STARTING FROM 200 ?
4093  032476  001413                          BEQ     3$              ;TEST NEXT DRIVE IF SO
4094                                                                  ;CONTINUE TESTING THIS ONE IF NOT
4095
4096  032500  005237  001100                  INC     @#$PASS         ;INCREASE PASS COUNT
4097  032504  104401  032675                  TYPE    ,$ENDMG         ;TYPE END PASS #
4098  032510  013746  001100                  MOV     @#$PASS,-(SP)   ;GET PASS NO.
4099  032514  104405                          TYPDS                   ;TYPE IT OUT
4100  032516  104401  032672                  TYPE    ,$ENULL
4101  032522  000137  007714                  JMP     @#TST4          ;JUMP TEST 4 ---------------------->
4102
4103  032526  012737  177777  040742  3$:     MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
4104  032534  005337  004620                  DEC     @#NOUNITS       ;NO. OF UNITS PRESENT DECREMENTED
4105  032540  001413                          BEQ     $EOP            ;BRANCH IF ALL DRIVES COMPLETE
4106  032542  013700  004616                  MOV     @#UNIT,R0       ;UNIT UNDER TEST
4107  032546  012701  004576                  MOV     #UNITS,R1       ;TABLE
4108  032552  022100          1$:     CMP     (R1)+,R0        ;IS THIS UNIT JUST TESTED ?
4109  032554  001401                          BEQ     2$              ;CONTINUE IF YES
4110  032556  000775                          BR      1$              ;INCREMENT IF NO
4111  032560  011137  004616          2$:     MOV     (R1),@#UNIT     ;THIS IS NEXT UNIT
4112  032564  000137  007714                  JMP     @#TST4          ;TEST THE NEXT DRIVE ---------------->
4113
```

```
4114
4115                                            .SBTTL
4116                                            .SBTTL  ***SUBROUTINES***
4117                                            .SBTTL
4118
4119
4120
4121
4122
```

```
                                ;THIS FILLS MEMORY WITH GIVEN DATA
                                ;USED CHIEFLY FOR HEADER INFORMATION
                                ;CALL IS
                                ;       JSR     R0,@#FLHEAD     ;FILL HEADER
                                ;       LOC                     ;LOCATION WHERE SAVED
                                ;       XN                      ;NUMBER OF WORDS
                                ;       XD1                     ;DATA REPEATED XN TIMES
                                ;       XD2                     ;DATA REPEATED XN TIMES
                                ;
                                ;
                                ;
032712                          FLHEAD:
032716  012001                          MOV     (R0)+,R1        ;R1 HAS ADDRESS OF WHERE TO SAVE
032720  012002                          MOV     (R0)+,R2        ;R2 HAS NUMBER OF WORDS
                                        ;*NOW FILL DATA
032722  012021                  1$:     MOV     (R0)+,(R1)+     ;SAVE DATA
032724  005302                          DEC     R2              ;DECREMENT COUNT
032726  001375                          BNE     1$              ;BRANCH IF INCOMPLETE
032734  000200                          RTS     R0


                                ;THIS CLEARS ANY BLOCK OF MEMORY.
                                ;FILLING IT WITH ANY DATA
                                ;CALL IS
                                ;       JSR     R0,@#CLAREA
                                ;       F                       ;FROM
                                ;       N                       ;NUMBER OF WORDS
                                ;       D                       ;DATA  TO BE FILLED
                                ;
                                ;R1 WILL HAVE STARTING ADDRESS OF BLOCK TO BE FILLED
                                ;R2 WILL HAVE NUMBER OF WORDS
                                ;R3 WILL HAVE DATA

032736                          CLAREA:
032744  012001                          MOV     (R0)+,R1        ;FROM
032746  012002                          MOV     (R0)+,R2        ;NUMBER
032750  012003                          MOV     (R0)+,R3        ;DATA
032752  010321                  1$:     MOV     R3,(R1)+        ;MOVE DATA
032754  005302                          DEC     R2              ;COUNT
032756  001375                          BNE     1$              ;BRANCH IF NOT COMPLETE
032766  000200                          RTS     R0              ;RETURN TO MAIN PROGRAM


                                ;THIS IS A SUBROUTINE TO FILL SAVED REGISTER LOCATION
                                ;WITH GIVEN VALUE
                                ;CALL IS
```

```
4179                                    ;       JSR     R0,@#FILLRE
4180                                    ;       RHXX            ;REGISTER NAME
4181                                    ;       D               ;DATA
4182                                    ;
4183
4184   032770                           FILLRE:
4185   032774  012001                           MOV     (R0)+,R1        ;ADDRESS OF ADDRESS OF REGISTER
4186   032776  012002                           MOV     (R0)+,R2        ;DATA
4187   033000  162701  002172                    SUB     #RHWC,R1        ;OFSET
4188   033004  010261  004512                    MOV     R2,SAVERE(R1)   ;DATA IS MOVED IN
4189   033014  000200                           RTS     R0              ;RETURN TO MAIN PROGRAM
4190
4191
4192
```

```
 4193                                   ;THIS SUBROUTINE SETS UP FOR SEARCH
 4194                                   ;CALL IS
 4195                                   ;       JSR     R0,@#SRCH
 4196                                   ;       C                       ;CYLINDER
 4197                                   ;.BYTE  S                       ;SECTOR
 4198                                   ;.BYTE  T                       ;TRACK
 4199
 4200  033016  012077  147170           SRCH:   MOV     (R0)+,@RHCA     ;SET DESIRED CYLINDER ADDRESS
 4201  033022  012077  147156                   MOV     (R0)+,@RHDST    ;SET DESIRED SECTOR/TRACK ADDRESS
 4202  033026  013777  002334  147144           MOV     @#SERCH,@RHCS1  ;GET READY FOR SEARCH
 4203                                                                   ;WITH 30 IN RHCS1
 4204  033034  000200                           RTS     R0
 4205
 4206
 4207
 4208
 4209
 4210
 4211
 4212                                   ;THIS SUBROUTINE SETS UP FOR SEEK COMMANDS
 4213                                   ;CALL IS
 4214                                   ;       JSR     R0,@#SEEKCY
 4215                                   ;       C                       ;CYLINDER
 4216                                   ;
 4217
 4218  033036  012077  147150           SEEKCY: MOV     (R0)+,@RHCA     ;SET DESIRED CYLINDER ADDRESS
 4219  033042  013777  002352  147130           MOV     @#SEECOM,@RHCS1 ;MOV 4 INTO RHCS1
 4220  033050  000200                           RTS     R0              ;RETURN TO MAIN PROGRAM
```

```
 4221
 4222                                   ;THIS SUBROUTINE SETS UP FOR OFFSET COMMANDS
 4223                                   ;CALL IS
 4224                                   ;       JSR     R0,@#OFSET
 4225                                   ;       0                       ;MICRO INCHES OFFSET
 4226
 4227  033052  052077  147132  OFSET:  BIS     (R0)+,@RHOF             ;SET OFFSET REGISTER
 4228  033056  013777  002354  147114  MOV     @#OFSETC,@RHCS1         ;MOV14 INTO RHCS1
 4229  033064  000200                  RTS     R0                      ;RETURN TO MAIN PROGRAM
 4230
 4231
 4232  033066  013701  002200  CLDISK: MOV     @#RHCS1,R1              ;R1 WILL BE CONTROL AND STATUS1
 4233  033072  013702  002176          MOV     @#RHCS2,R2              ;R2 WILL BE CONTROL AND STATUS2
 4234  033076  013703  002222          MOV     @#RHDS1,R3              ;R3 WILL BE DISK STATUS REGISTER1
 4235  033102  013704  002202          MOV     @#RHER1,R4              ;R4 WILL BE ERROR REGISTER #1
 4236
 4237  033106  012712  000040          MOV     #CLR,@R2                ;CLEAR ALL REG.
 4238  033112  013712  004616          MOV     @#UNIT,@R2              ;REINSTATE UNIT NO.
 4239  033116  005011                  CLR     @R1                     ;CLEAR FUNCTION BITS
 4240  033120  000207                  RTS     PC
```

```
                                        ;THIS CHECKS DEVICE AVAILABLE (DVA) AND READY (RDY) IN RHCS1
                                        ;AND CHECKS MEDIUM ON LINE (MOL), DEVICE PRESENT (DPR), DEVICE READY (DRY) IN RHDS1
                                        ;IT MAY CHECK VOLUME VALID (VV) IN RHDS1, DEPENDING ON ENTRY POINT

033122  000000                  PCJSR:  0                       ;PC OF JSR

033124  011637  033122          CHECK:  MOV     (SP),@#PCJSR    ;SAVE PC OF JSR+4
033130  162737  000004  033122          SUB     #4,@#PCJSR      ;GET PC OF JSR
033136  011346                          MOV     @R3,-(SP)       ;GET RHDS1
033140  052716  000100                  BIS     #VV,(SP)        ;DONT CHECK VV BIT
033144  000406                          BR      CHECKC          ;GOTO COMMON CHECK ROUTINE

033146  011637  033122          CHECKT: MOV     (SP),@#PCJSR    ;SAVE PC OF JSR+4
033152  162737  000004  033122          SUB     #4,@#PCJSR      ;GET PC OF JSR
033160  011346                          MOV     @R3,-(SP)       ;GET RHDS1 & DO VV CHECK AT 3$

033162  011146                  CHECKC: MOV     @R1,-(SP)       ;GET CS1
033164  042716  173577                  BIC     #173577,(SP)    ;CLEAR UNWANTED BITS
033170  022726  004200                  CMP     #DVA!RDY,(SP)+  ;RHCS1 SHOULD HAVE DEVICE AVAILABLE
                                                                ;AND BE READY
033174  001403                          BEQ     3$              ;BRANCH IF GOOD
033176  011137  001122                  MOV     @R1,@#$BDADR    ;BAD DATA REGISTER (RHCS1)
033202  104062                          ERROR   62              ;RHCS1 DID NOT HAVE DEVICE
                                                                ;AVAILABLE RIGHT AT THE START
                                                                ;ALL OTHER BITS SHOULD BE 0
033204  042716  102000          3$:     BIC     #ATA!LBT,(SP)   ;CLEAR UNWANTED BITS
033210  022726  010700                  CMP     #MOL!DPR!DRY!VV,(SP)+ ;RHDS1 SHOULD HAVE THESE SET
033214  001404                          BEQ     7$              ;BRANCH IF GOOD
033216  011337  001122                  MOV     @R3,@#$BDADR    ;BAD DATA IN REGISTER (RHDS1)
033222  104061                          ERROR   61              ;RHDS1 HAS SOME BITS OTHER
                                                                ;THAN MOL, DRY, DPR,VV SET
                                                                ;ALL OTHER BITS SHOULD BE 0
033224  000207                          RTS     PC              ;RETURN TO TEST AND HALT

033226  062716  000006          7$:     ADD     #6,(SP)         ;ADJUST STACK TO JUMP OVER HALT IN TEST
033232  000207                          RTS     PC              ;RETURN TO TEST AND CONTINUE
```

```
                                        ;*THIS IS A SUBROUTINE TO SAVE REGISTERS
                                        ;*IN THE REGISTER TABLE TO ANY LOCATION
                                        ;*THE CALL IS
                                        ;*JSR   R0,@#SAVER
                                        ;*      F       ;FROM
                                        ;*      T       ;TO
                                        ;*      N       ;NUMBER OF WORDS SAVED
                                        ;*F MUST ALWAYS BE RHCS1
                                        ;*T MUST ALWAYS BE SAVRE

033234                          SAVER:
033242  012001                          MOV     (R0)+,R1        ;FROM
033244  012002                          MOV     (R0)+,R2        ;TO
033246  012003                          MOV     (R0)+,R3        ;NUMBER
033250  013122                  1$:     MOV     @(R1)+,(R2)+    ;SAVE REGISTER CONTENTS
033252  005303                          DEC     R3              ;COUNT
033254  001375                          BNE     1$              ;BRANCH IF NOT DONE
033264  000200                          RTS     R0


                                        ;WHEN AN EVENT IS TO BE TIMED THE RP04 VECTORS TO "TIME 1"
                                        ;PRIORITY OF PROCESS OR IS 4
                                        ;PRIORITY OF TRAPS MUST BE 6
                                        ;PRIORITY OF RP04 INTERRUPTS IS 7
                                        ;

033266  005077  146752          TIME1:  CLR     @PCLCSR         ;STOP THE CLOCK
033272  017737  146752  033324          MOV     @PCLCTR,@#WAITTM ;GET TIME ON CLOCK
033300  017737  146730  004564  TIME2:  MOV     @RHCC,@#FINACC  ;GET CURRENT CYLINDER
033306  017737  146724  004562          MOV     @RHLA,@#FINALA  ;GET LOOK AHEAD
033314  000002                          RTI                     ;RETURN TO WAIT P OR WAIT.T


                                        ;THIS IS A WAIT LOOP WHEN AN EVENT IS TO BE TIMED
                                        ;THE CALL IS
                                        ;       WAT
                                        ;       A       ;ABSOLUTE REGISTER ADDRESS
                                        ;       B       ;BIT WAITED FOR
                                        ;       TA      ;TIME ALLOWED GIVEN IN 10 MICROSEC
                                        ;       TO      ;TOLERANCE PLUS/MINUS IN 10 MICROSEC
                                        ;
```

```
                                        ;R1-WILL HAVE TIME ALLOWED IN 10 MICRO SECONDS
                                        ;R2-WILL HAVE TOLERANCE PLUS/MINUS IN 10 MICRO SECONDS
                                        ;MINIMUM TIME THAT CAN BE MEASURED IS ABOUT 12 MICRO SECONDS
                                        ;FOR THE SLOWEST PROCESSOR

033316  000000                  WAITPC: 0                       ;WAT PC
033320  000000                  WAITRE: 0                       ;WAIT ON REGISTER ADDRESS
033322  000000                  WAITBT: 0                       ;WAIT ON BIT
033324  000000                  WAITTM: 0                       ;WAITED TIME
033326  005037  033324          WAIT.P: CLR     @#WAITTM        ;CLEAR WAITED TIME
033332  005077  146710                  CLR     @PCLBUF ;CLEAR COUNT SET BUFFER
033336  012777  000021  146700          MOV     #GO!BIT4,@PCLCSR ;COUNT UP, 100 KHZ, START CLOCK
033354  016600  000010                  MOV     10(SP),R0       ;R0 HAS ADDRESS OF NEXT LOCATION
033360  010037  033316                  MOV     R0,@#WAITPC     ;NOW WAITPC HAS WAT PC + 2
033364  162737  000002  033316          SUB     #2,@#WAITPC     ;WAT PC IS IN WAITPC
033372  013037  033320                  MOV     @(R0)+,@#WAITRE ;WAIT ON REGISTER ADDRESS
033376  012037  033322                  MOV     (R0)+,@#WAITBT  ;WAIT ON BIT
033402  012001                          MOV     (R0)+,R1        ;R1 HAS TIME IN 10 MSEC
033404  012002                          MOV     (R0)+,R2        ;R2 HAS TOLERANCE IN 10 MSEC
033406  010066  000010                  MOV     R0,10(SP)       ;RESTORE RETURN ON STACK

                                        ;*THIS SECTION WAITS FOR BIT, THROUGH TWO COUNT DOWNS

033412  013703  033564                  MOV     @#TIMCNT,R3     ;R3 IS A TEMPORARY COUNTER
033416  033777  033322  177674  1$:     BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
033424  001025                          BNE     4$              ;BRANCH IF YES
033426  005303                          DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
033430  001372                          BNE     1$
033432  013703  033564                  MOV     @#TIMCNT,R3     ;TEMPORARY COUNTER
033436  033777  033322  177654  2$:     BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
033444  001015                          BNE     4$              ;BRANCH IF YES
033446  005303                          DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
033450  001372                          BNE     2$
033452  017737  177642  001126          MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
033460  032777  000100  146512          BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
033466  001402                          BEQ     3$              ;BRANCH IF YES
033470  104001                          ERROR   1               ;RP04 DID NOT INTERRUPT
033472  000427                          BR      7$              ;OUT
033474  104002                  3$:     ERROR   2               ;RP04 INTERRUPTED BUT WAITED
                                        ;ON BIT DID NOT OCCUR
                                        ;EVEN AFTER TWO COUNT DOWNS
                                        ;FROM 177777 TO 0
033476  000425                          BR      7$              ;OUT

                                        ;*NOW TIME AND TOLERANCE WILL BE CHECKED
033500  017737  177614  001126  4$:     MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
033506  032777  000100  146464          BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
033514  001402                          BEQ     5$              ;BRANCH IF YES
033516  104003                          ERROR   3               ;INTERRUPT DID NOT OCCUR EVEN
                                        ;AFTER ONE BNE AND ONE MOV
                                        ;OF THE WAITED ON BIT SETTING
033520  000414                          BR      7$              ;OUT
033522  160201                  5$:     SUB     R2,R1           ;R1 NOW HAS LOWER LIMIT OF TIME
033524  023701  033324                  CMP     @#WAITTM,R1     ;FOR GOOD RESULTS, WAITTM
                                        ;MUST BE GREATER OR EQUAL
                                        ;TORI
```

```
4395  033530  103002                          BHIS    6$              ;BRANCH IF GOOD
4396  033532  104004                          ERROR   4               ;BIT DID OCCUR BUT TIME
4397                                                                  ;TAKEN IS BELOW LOWER LIMIT
4398  033534  000406                          BR      7$              ;OUT
4399
4400  033536  060202                  6$:     ADD     R2,R2           ;DOUBLE TOLERANCE
4401  033540  060201                          ADD     R2,R1           ;R1 NOW HAS UPPER LIMIT OF TIME
4402  033542  020137  033324                  CMP     R1,@#WAITTM     ;FOR GOOD RESULTS, WAITTM
4403                                                                  ;MUST BE LESS OR EQUAL TO R1
4404  033546  103001                          BHIS    7$              ;BRANCH IF GOOD
4405  033550  104004                          ERROR   4               ;BIT DID OCCUR BUT TIME TAKEN
4406                                                                  ;IS ABOVE UPPER LIMIT
4407  033552                          7$:
4408  033562  000002                          RTI                     ;RETURN TO MAIN TEST
4409
4410
4411
4412
4413
4414
4415                                  ;THIS IS A WAIT LOOP WHEN NO P-CLOCK IS AVAILABLE
4416                                  ;NO TIMING IS DONE
4417                                  ;CALL IS
4418                                  ;       WAT
4419                                  ;       A       ;ABSOLUTE REGISTER ADDRESS
4420                                  ;       B       ;BIT WAITED FOR
4421                                  ;       TA      ;TIME-NOT USED HERE
4422                                  ;       TO      ;TIME-NOT USED HERE
4423                                  ;R3-IS A TEMPORARY COUNTER
4424
4425  033564  177777                  TIMCNT: 177777                  ;COUNT FOR WAIT LOOP
4426  033566  000025                  RPTCTR: 25                      ;COUNT FOR INTERRUPT WAIT (11/70 CPU)
4427
4428
4429  033570                          WAIT.T:
4430
4431  033574  016600  000004                  MOV     4(SP),R0        ;R0 HAS ADDRESS OF NEXT LOCATION
4432  033600  010037  033316                  MOV     R0,@#WAITPC     ;WAT PC +2 IS IN WAITPC
4433  033604  162737  000002  033316          SUB     #2,@#WAITPC     ;WAT PC IS IN WAITPC
4434  033612  013037  033320                  MOV     @(R0)+,@#WAITRE ;WAIT ON REGISTER ADDRESS
4435  033616  012037  033322                  MOV     (R0)+,@#WAITBT  ;WAIT ON BIT
4436  033622  022020                          CMP     (R0)+,(R0)+     ;DUMP NEXT TWO WORDS-TA, TO
4437  033624  010066  000004                  MOV     R0,4(SP)        ;RESTORE RETURN ON STACK
4438
4439                                  ;*THIS HAS THE TWO COUNT DOWNS FROM 177777
4440
4441  033630  013703  033564                  MOV     @#TIMCNT,R3     ;R3 HAS TEMPORARY COUNT
4442  033634  033777  033322  177456  1$:     BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE ?
4443  033642  001025                          BNE     4$              ;CHECK FOR THE INTERRUPT
4444  033644  005303                          DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
4445  033646  001372                          BNE     1$
4446  033650  013703  033564                  MOV     @#TIMCNT,R3     ;SECOND COUNT DOWN FROM 177777
4447  033654  033777  033322  177436  2$:     BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE ?
4448  033662  001015                          BNE     4$              ;CHECK FOR INTERRUPT
4449  033664  005303                          DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
4450  033666  001372                          BNE     2$
```

```
4451  033670  017737  177424  001126          MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
4452  033676  032777  000100  146274          BIT     #IE,@RHCS1       ;DID ANY INTERRUPT OCCUR ?
4453  033704  001402                          BEQ     3$               ;BRANCH IF YES
4454
4455  033706  104001                          ERROR   1                ;RP04 DID NOT INTERRUPT
4456                                                                   ;BIT DID NOT OCCUR
4457  033710  000417                          BR      5$               ;OUT ------------------------------->
4458
4459  033712  104002                  3$:     ERROR   2                ;RP04 INTERRUPTED BUT
4460                                                                   ;WAITED ON BIT DID NOT OCCUR
4461                                                                   ;EVEN AFTER TWO COUNT DOWNS
4462                                                                   ;FROM 177777 TO 0
4463  033714  000415                          BR      5$               ;OUT ------------------------------->
4464
4465                                          ;*BIT DID SET SO CHECK IF INTERRUPT OCCURRED
4466
4467                                          ;*THE AMOUNT OF TIME ALLOWED CAN BE CHANGED BY ALTERING LOCATION
4468                                          ;*"RPTCTR" ABOVE
4469
4470  033716  013703  033566          4$:     MOV     @#RPTCTR,R3      ;LOAD COUNTER WITH COUNT
4471  033722  005303                  6$:     DEC     R3               ;COUNT DOWN ONE
4472  033724  001376                          BNE     6$               ;DO AGAIN IF NOT ZERO YET
4473
4474
4475  033726  032777  000100  146244          BIT     #IE,@RHCS1       ;DID ANY INTERRUPT OCCUR ?
4476  033734  001405                          BEQ     5$               ;BRANCH IF YES
4477  033736  017737  177356  001126          MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
4478  033744  104003                          ERROR   3                ;INTERRUPT DID NOT OCCUR
4479                                                                   ;EVEN AFTER ONE BNE OF
4480                                                                   ;THE WAITED ON BIT OCCURING
4481  033746  000400                          BR      5$               ;OUT ------------------------------->
4482
4483  033750                          5$:
4484  033754  000002                          RTI                      ;RETURN TO MAIN TEST
4485
```

```
                                ;THIS CHANGES REGISTER SAVED VALUE
                                ;CALL IS
                                ;       JSR     R0,@#CHREG
                                ;       R                       ;REGISTER TO BE CHANGED
                                ;       N                       ;NUMBER OF BITS TO BE CHANGED
                                ;       NEW                     ;NEW VALUE OF BIT MUST BE 0 OR1
                                ;       P                       ;POSITION OF BIT TO BE CHANGED
                                ;NEW AND P WILL BE REPEATED N NUMBER OF TIMES

033756                          CHREG:
033762  012001                          MOV     (R0)+,R1        ;R1 HAS ADDRESS OF ADDRESS OF REGISTER
033764  012002                          MOV     (R0)+,R2        ;R2 HAS NUMBER OF CHANGES
033766  162701  002172                  SUB     #RHWC,R1        ;R1 HAS OFSET OF REQUIRED REGISTER
033772  005720                  1$:     TST     (R0)+           ;IS A BIC OR A BIS TO BE DONE
033774  001403                          BEQ     2$              ;BRANCH IF A BIC IS REQUIRED
033776  052061  004512                  BIS     (R0)+,SAVERE(R1) ;SET REQUIRED BIT
034002  000402                          BR      3$              ;BRANCH TO DECREMENT COUNT
034004  042061  004512          2$:     BIC     (R0)+,SAVERE(R1) ;CLEAR REQUIRED BIT
034010  005302                  3$:     DEC     R2              ;DECREMENT NUMBER OF CHANGES
034012  001367                          BNE     1$              ;BRANCH IF NOT COMPLETE
034020  000200                          RTS     R0              ;RETURN TO MAIN PROGRAM


                                ;THIS FILLS A BLOCK WITH INCREMENTAL DATA
                                ;CALL IS
                                ;       JSR     R0,@#FILL
                                ;       F                       ;FROM
                                ;       N                       ;NUMBER OF WORDS
                                ;       S                       ;STARTING VALUE OF DATA
                                ;       I                       ;INCREMENT DATA BY

034022                          FILL:
034032  012001                          MOV     (R0)+,R1        ;R1 HAS ADDRESS WHERE DATA IS TO GO
034034  012002                          MOV     (R0)+,R2        ;R2 HAS NUMBER OF WORDS TO BE FILLED
034036  012003                          MOV     (R0)+,R3        ;STARTING VALUE OF DATA
034040  012004                          MOV     (R0)+,R4        ;R4 HAS INCREMENT

                                        ;*NOW DATA WILL BE FILLED
034042  010321                  1$:     MOV     R3,(R1)+        ;FILL DATA
034044  060403                          ADD     R4,R3           ;GET NEXT VALUE OF DATA
034046  005302                          DEC     R2              ;DECREMENT COUNT
034050  001374                          BNE     1$              ;BRANCH IF ALL NOT DONE
034062  000200                          RTS     R0              ;RETURN TO MAIN PROGRAM


                                ;THIS IS A SUBROUTINE TO COMPARE REGISTERS
                                ;GOOD DATA IS ALREADY SAVED IN 'SAVERE'
```

```
 4542                                   ;TEST DATA IS IN THE REGISTERS
 4543                                   ;CALL IS
 4544                                   ;       JSR     R0,@#COMREG
 4545                                   ;       SAVERE                  ;GOOD DATA
 4546                                   ;       RHCS1                   ;ADDRESS OF ADDRESS TEST DATA
 4547                                   ;       N.                      ;RETURN FOR ERROR
 4548                                   ;       RG                      ;RETURN FOR GOOD COMPARISON
 4549                                   ;ON RETURN WITH ERROR '$GDDAT' HAS GOOD DATA, '$BDDAT' HAS BAD DATA
 4550                                   ;'REGADR' HAS REGISTER ADDRESS
 4551
 4552  034064                           COMREG:
 4553  034076  012001                           MOV     (R0)+,R1        ;R1 HAS ADDRESS OF GOOD DATA
 4554  034100  012002                           MOV     (R0)+,R2        ;R2 HAS ADDRESS OF ADDRESS OF TEST DATA
 4555  034102  012003                           MOV     (R0)+,R3        ;R3 HAS NUMBER OF WORDS
 4556  034104  012004                           MOV     (R0)+,R4        ;R4 HAS RETURN FOR ERROR
 4557  034106  011000                           MOV     (R0),R0         ;R0 HAS RETURN ON NO ERROR
 4558                                           ;*NOW SAVE REGISTERS
 4559  034110  004737  035010                   JSR     PC,@#PUTREG     ;SAVE REGISTERS
 4560  034114  113737  004537  002301           MOVB    @#$AVERE+25,@#AS+1;MAKE UPPER BYTE OF RHAS SAME
 4561  034122  012705  177776                   MOV     #-2,R5          ;PRESET R5 TO -2
 4562                                           ;*NOW COMPARES WILL MADE
 4563  034126  062705  000002           1$:     ADD     #2,R5           ;INCREMENT TO INDEX
 4564  034132  022122                           CMP     (R1)+,(R2)+     ;COMPARE REGISTER CONTENTS
 4565  034134  001420                           BEQ     2$              ;BRANCH IF GOOD
 4566  034136  014137  001124                   MOV     -(R1),@#$GDDAT  ;SAVE GOOD DATA
 4567  034142  014237  001126                   MOV     -(R2),@#$BDDAT  ;SAVE BAD DATA
 4568  034146  016537  002172  004500           MOV     RHWC(R5),@#REGADR       ;SAVE ADDRESS OF FAILING REGISTER
 4569  034154  004714                           JSR     PC,@R4          ;RETURN TO MAIN PROGRAM
 4570                                                                   ;TO PRINT ERROR
 4571  034156  022122                           CMP     (R1)+,(R2)+     ;UNDO -(R1) AND -(R2) FOR ERRORS
 4572  034160  017746  144754                   MOV     @SWR,-(SP)      ;GET SWITCH SETTING
 4573  034164  042716  177177                   BIC     #^C600,(SP)     ;KEEP ONLY SWITCH 7 AND 8
 4574  034170  022726  000200                   CMP     #SW07,(SP)+     ;IS 7 SET AND 8 DOWN
 4575  034174  001402                           BEQ     3$              ;BRANCH OUT IF YES
 4576  034176  005303                   2$:     DEC     R3              ;ARE ALL COMPARES DONE
 4577  034200  001352                           BNE     1$              ;BRANCH IF NOT COMPLETE
 4578
 4579  034202                           3$:
 4580  034214  000200                           RTS     R0              ;RETURN TO MAIN PROGRAM
 4581  034216  000000                   4$:     .WORD   0               ;TEMP STORAGE
```

```
                                ;HERE IS A DETAILED EXPLAINATION OF HOW THE LOOP ON ERROR WORKS.
                                ;ON HITTING AN ERROR IF THE LOOP ON ERROR SWITCH IS SET, THE
                                ;PROGRAM GOES BACK - USUALLY BACK TO THE BEGINNING OF THE TEST.

                                ;WHEN THIS OPERATOR SELECTABLE SCOPE LOOP IS USED THEN THE POINT
                                ;THE PROGRAM GOES BACK TO CAN BE CHANGED.
                                ;THE RESTRICTIONS TO THE POINT WHERE THE PROGRAM CAN GO ARE: -
                                ;1.  IT MUST BE WITHIN THE TEST UNDER CONSIDERATION
                                ;2.  LOOP ON ERROR SWITCH MUST BE SET
                                ;3.  THE ERROR MUST OCCUR WITHIN THE TEST UNDER CONSIDERATION
                                ;IF THE ERROR DOES NOT OCCUR WITHIN THE TEST UNDER CONSIDERATION
                                ;THE PROGRAM WILL REVERT TO NORMAL OPERATION.  HOWEVER, IF LOOP ON
                                ;TEST SWITCH IS SET AND THIS OPERATOR SELECTABLE SCOPE LOOP IS USED
                                ;THEN THE PROGRAM WILL LOOP BACK TO THE SELECTED POINT WHEN IT
                                ;COMES TO THE END OF THE TEST UNDER CONSIDERATION.
                                ;
                                ;AFTER LOOPING FOR SOME TIME IF THE LOOP SWITCH IS PUT DOWN THEN
                                ;NORMAL OPERATION WILL CONTINUE.

034220  000000          TESTAD: 0                       ;FIRST ADDRESS OF TEST
034222                  OPERSEL:
034222  005037  177776          CLR     PS              ;MAKE PROCESSOR STATUS ZERO
034226  012737  177777  040742  MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
034304  013746  004504          MOV     @#TSTNM,-(SP)   ;GET READY TO TYPE TEST
034310  104402                  TYPOC                   ;NUMBER
034350  013746  001110          MOV     @#$LPERR,-(SP)  ;GET READY TO TYPE LOOP BACK PC
034354  104402                  TYPOC
034356  104401  001223          TYPE    ,$CRLF
034610  104412                  RDOCT
034612  062716  000002          ADD     #2,(SP)         ;GET LPADR
034616  012637  001106          MOV     (SP)+,@#$LPADR
034774  104412                  RDOCT
034776  012637  001110          MOV     (SP)+,@#$LPERR  ;GET LPERR
035002  013746  001106          MOV     @#$LPADR,-(SP)
035006  000002                  RTI


                                ;*THIS SAVES THE CONTENTS OF ALL HARDWARE REGISTERS
                                ;*IN MEMORY LOCATIONS TAGED FROM "WC" TO "EC2"
                                ;*THIS IS DONE SO THAT COMPARES ARE DONE WITH SAVED LOCATIONS
                                ;*AND NOT THE REGISTERS THEMSELVES.   THIS WILL MAKE
                                ;*ERROR PRINTOUTS FOR GOOD AND BAD DATA ALWAYS DIFFRENT


035010                  PUTREG:
035016  012700  002172          MOV     #RHWC,R0        ;STARTING ADDRESS OF REGISTERS
035022  012701  002254          MOV     #WC,R1          ;STARTING ADDRESS OF SAVING LOCATIONS
035026  012702  000022          MOV     #RHCC-RHWC+2/2,R2 ;NUMBER OF REG. INTO R2
```

```
 4638  035032  013021                  10$:    MOV     @(R0)+,(R1)+    ;SAVE HARDWARE REG.
 4639  035034  005302                          DEC     R2
 4640  035036  001375                          BNE     10$
 4641  035046  000207                          RTS     PC
```

```
                                ;THIS IS A DATA  COMMAND SETUP SUBROUTINE
                                ;THE CALL IS
                                ;       JSR     R0,@#RUN
                                ;       C                       ;CYLINDER
                                ;.BYTE  S                       ;SECTOR
                                ;.BYTE  T                       ;TRACK
                                ;       -W                      ;WORD COUNT
                                ;       B                       ;BUS ADDRESS
                                ;       BAI                     ;BUS ADDRESS INHIBIT
                                ;       FMT22!ECI!HCI           ;FMT22=1 =16 BIT WORDS
                                ;                               ;ECI = ECC CORRECTION INHIBIT
                                ;                               ;HCI = HEADER COMPARE INHIBIT
                                ;       COM                     ;COMMAND ADDRESS
035050  012077  145136          RUN:    MOV     (R0)+,@RHCA     ;CYLINDER
035054  012077  145124                  MOV     (R0)+,@RHDST    ;DESIRED SECTOR/TRACK
035060  012077  145106                  MOV     (R0)+,@RHWC     ;WORD COUNT
035064  012077  145104                  MOV     (R0)+,@RHBA     ;BUS ADDRESS
035070  013746  004616                  MOV     @#UNIT,-(SP)    ;GET UNIT NO
035074  052016                        BIS     (R0)+,(SP)      ;SET BUS ADDRESS INHIBIT
035076  012677  145074                  MOV     (SP)+,@RHCS2    ;UNIT NO AND BAI TO RHCS2
035102  012077  145102                  MOV     (R0)+,@RHOF     ;FORMAT, ECC INHIBIT, HEADER
                                                                ;COMPARE, IF THERE
035106  013077  145066                  MOV     @(R0)+,@RHCS1   ;COMMAND IN RHCS1
035112  000200                        RTS     R0              ;RETURN TO MAIN PROGRAM


                                ;THIS IS A SUBROUTINE TO COMPARE TWO BLOCKS IN MEMORY
                                ;R1 HAS GOOD DATA BUFFER ADDRESS
                                ;R2 HAS TEST DATA BUFFER ADDRESS
                                ;R5 HAS ADDRESS OF RETURN ON ERROR
                                ;R3 HAS NUMBER OF WORDS TO BE COMPARED
                                ;R4 HAS ONE MORE THAN NUMBER OF WORDS TO BE COMPARED
                                ;CALL IS
                                ;       JSR     R0,@#COMPAR
                                ;       G                       ;ADDRESS OF GOOD DATA
                                ;       T                       ;ADDRESS OF TEST DATA
                                ;       N                       ;NUMBER OF WORDS TO BE COMPARED
                                ;       RE                      ;RETURN ON ERROR
                                ;       RG                      ;RETURN ON NO ERROR


035114                          COMPAR:
035126  012001                          MOV     (R0)+,R1        ;ADDRESS OF GOOD DATA BUFFER
035130  012002                          MOV     (R0)+,R2        ;ADDRESS OF TEST DATA BUFFER
035132  012003                          MOV     (R0)+,R3        ;NO OF WORDS TO BE COMPARED
035134  012005                          MOV     (R0)+,R5        ;RETURN ON ERROR
035136  011000                          MOV     (R0),R0         ;RETURN ON NO ERROR
035140  010304                          MOV     R3,R4           ;NO OF WORDS TO BE COMPARED
035142  005204                          INC     R4
035144  010437  004502          1$:     MOV     R4,@#ERWORD     ;FOR ERROR WORD NO
035150  022122                          CMP     (R1)+,(R2)+     ;COMPARE GOOD WITH TEST DATA
035152  001417                          BEQ     2$              ;BRANCH IF GOOD
```

```
 4698  035154  014137  001124                 MOV     -(R1),@#$GDDAT  ;GOOD DATA
 4699  035160  014237  001126                 MOV     -(R2),@#$BDDAT  ;BAD DATA
 4700  035164  160337  004502                 SUB     R3,@#ERWORD     ;ERROR WORD NO.
 4701  035170  004715                         JSR     PC,@R5  ;RETURN TO PRINT ERROR
 4702  035172  022122                         CMP     (R1)+,(R2)+     ;UNDO -(R1) AND -(R2) FOR ERRORS
 4703  035174  017746  143740                 MOV     @SWR,-(SP)      ;GET SWITCH SETTING
 4704  035200  042716  177177                 BIC     #^C600,(SP)     ;KEEP ONLY SWITCH 7 AND 8
 4705  035204  022726  000200                 CMP     #SW07,(SP)+     ;IS 7 SET AND 8 RESET
 4706  035210  001402                         BEQ     3$              ;BRANCH OUT IF YES
 4707  035212  005303                 2$:     DEC     R3              ;COUNT
 4708  035214  001353                         BNE     1$              ;BRANCH IF ALL NOT DEVICE
 4709  035216                         3$:
 4710  035230  000200                         RTS     R0              ;RETURN TO MAIN PROGRAM
 4711                                 ;*      THIS ROUTINE WILL ALLOW THE CHANGE OF THE BASE
 4712                                 ;*      ADDRESS FROM 176700 TO ANY TYPED VALUE
 4713
 4714  035232                         BASECH:
 4715  035312  013746  002200                 MOV     @#RHCS1,-(SP)   ;GET READY TO TYPE OLD BASE
 4716  035316  104402                         TYPOC
 4717  035400  004737  037334                 JSR     PC,@#$TKINT     ;INITIALIZE THE TTY KEYBOARD
 4718  035404  104412                         RDOCT
 4719  035406  012700  002170                 MOV     #RHDB,R0        ;GET STARTING ADDRESS OF REGISTERS
 4720  035412  012701  000026                 MOV     #22.,R1         ;NUMBER OF REGISTERS
 4721  035416  012737  036222  000004         MOV     #ADTIMO,@#4     ;SET UP TRAP CATCHER FOR TEST
 4722  035424  021637  002200                 CMP     @SP,@#RHCS1     ;NEW ADDRESS
 4723  035430  001407                         BEQ     1$              ;NO, JUST OLD ONE RETYPED
 4724  035432  005776  000000                 TST     @0(SP)          ;DO THE ADDRESS ACCESS
 4725  035436  163716  002200                 SUB     @#RHCS1,@SP     ;GET THE ADDRESS OFFSET
 4726  035442  061620                 2$:     ADD     @SP,(R0)+       ;AND PLUG IT IN
 4727  035444  005301                         DEC     R1              ;ONE LESS REGISTER TO DO
 4728  035446  001375                         BNE     2$              ;BUT WE'RE NOT DONE YET!
 4729  035450                         1$:
 4730  035514  013746  002166                 MOV     @#RPVEC,-(SP)   ;GET READY TO TYPE OLD VECTOR ADDRESS
 4731  035520  104402                         TYPOC
 4732  035626  104412                         RDOCT
 4733  035630  012637  002166                 MOV     (SP)+,@#RPVEC   ;SETUP VECTOR ADDRESS
 4734  035700  013746  002200                 MOV     @#RHCS1,-(SP)
 4735  035704  104402                         TYPOC
 4736  035752  013746  002166                 MOV     @#RPVEC,-(SP)
 4737  035756  104402                         TYPOC
 4738  036176  012746  000200                 MOV     #RA,-(SP)
 4739  036202  104402                         TYPOC
 4740  036216  000137  004710                 JMP     @#BEGIN         ;DO IT OVER AGAIN
 4741  036222                         ADTIMO:
 4742  036300  022626                         CMP     (SP)+,(SP)+     ;RESTORE STACK
 4743  036302  000137  035232                 JMP     @#BASECH        ;AND DO THE QUERY AGAIN!
 4744
```

```
 4745  036306                                RPVECT:
 4746  036364  104402                              TYPOC                                ;TYPE FROM PC
 4747  036366  012777  036306  143572              MOV     #RPVECT,@RPVEC               ;RESTORE TRAP RP04 VECTOR
 4748  036374  000000                              HALT                                 ;CHANGE TO CONTINUE
 4749
 4750
```

```
                                .SBTTL  SYSMAC LIBRARY ROUTINES
```

```
4754                                    ;FROM THE TTY
```

```
                                ;;*****************************************************************
                                .SBTTL  ERROR MESSAGE TYPEOUT ROUTINE

                                ;*THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
                                ;*ERROR IS TO BE REPORTED.  IT THEN OBTAINS, FROM THE "ERROR TABLE" ($ERRTB),
                                ;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.
                                ;*IT IS A COPY OF THE $ERRTYP SUBROUTINE FROM SYSMAC.
                                ;*WITH ONLY MINOR CHANGES
                                ;*FIRST IF SWITCH 6 IS SET AND SWITCH 8 RESET THEN
                                ;*ALL REGISTER CONTENTS WILL BE TYPED BEFOR REPORTING THE ERROR
                                ;*SECOND IF THE CURRENT ERROR HAS THE SAME ITEM NUMBER
                                ;*AS THE PREVIOUS ERROR THEN ONLY THE DATA WILL BE TYPED
                                ;*AND NOT THE ERROR MESSAGE AND HEADER.

040742  000000                  PRITEM: 0                       ;PREVIOUS ITEM NO. LOCATION

040744  017746  140170          $ERRTYP: MOV    @SWR,-(SP)      ;GET SWITCH SETTING
040750  042716  177277                  BIC     #^C500,(SP)     ;KEEP ONLY SWITCH 8 AND 6
040754  022726  000100                  CMP     #SW06,(SP)+     ;IS 6 SET AND 8 RESET
040760  001001                          BNE     1$              ;IF NOT BRANCH
040762  000402                          BR      2$              ;BRANCH IF SW 6 IS SET AND 8 RESET
040764  000137  041704          1$:     JMP     @#TYPERR        ;JUMP IF SW 8 IS SET
                                                                ;OR IF SW 8 IS RESET AND SW 6 IS RESET
040770
041700  005037  040742          2$:             CLR     @#PRITEM        ;CLEAR PREVIOUS ERROR ITEM

041704                          TYPERR:
041704  104401  001223                  TYPE    ,$CRLF          ;"CARRIAGE RETURN" & "LINE FEED"
041710  010046                          MOV     R0,-(SP)        ;SAVE R0
041712  005000                          CLR     R0              ;PICKUP THE ITEM INDEX
041714  153700  001114                  BISB    @#$ITEMB,R0
041720  001004                          BNE     1$              ;IF ITEM NUMBER IS ZERO, JUST
                                                                ;TYPE THE PC OF THE ERROR
041722  013746  001116                  MOV     $ERRPC,-(SP)    ;SAVE $ERRPC FOR TYPEOUT
                                                                ;ERROR ADDRESS
041726  104402                          TYPOC                   ;GO TYPE--OCTAL ASCII(ALL DIGITS)
041730  000454                          BR      10$             ;GET OUT
041732  005300          1$:             DEC     R0              ;ADJUST THE INDEX SO THAT IT WILL
041734  006300                          ASL     R0              ;       WORK FOR THE ERROR TABLE
041736  006300                          ASL     R0
041740  006300                          ASL     R0
041742  062700  001226                  ADD     #$ERRTB,R0      ;FORM TABLE POINTER
041746  020037  040742                  CMP     R0,@#PRITEM     ;WAS PREVIOUS ERROR SAME
041752  001002                          BNE     13$             ;BRANCH IF NOT
041754  022020                          CMP     (R0)+,(R0)+     ;POP R0 OVER EM AND DH
041756  000420                          BR      5$
041760  010037  040742          13$:    MOV     R0,@#PRITEM     ;SAVE NEW ERROR ITEM
041764  012037  041774                  MOV     (R0)+,2$        ;PICKUP "ERROR MESSAGE" POINTER
041770  001404                          BEQ     3$              ;SKIP TYPEOUT IF NO POINTER
041772  104401                          TYPE                    ;TYPE THE "ERROR MESSAGE"
041774  000000          2$:             .WORD   0               ;"ERROR MESSAGE" POINTER GOES HERE
041776  104401  001223                  TYPE    ,$CRLF          ;"CARRIAGE RETURN" & "LINE FEED"
042002  012037  042012          3$:     MOV     (R0)+,4$        ;PICKUP "DATA HEADER" POINTER
042006  001404                          BEQ     5$              ;SKIP TYPEOUT IF 0
```

```
4811  042010  104401                  TYPE                    ;TYPE THE "DATA HEADER"
4812  042012  000000            4$:   .WORD   0               ;"DATA HEADER" POINTER GOES HERE
4813  042014  104401  001223        TYPE    ,$CRLF            ;"CARRIAGE RETURN" & "LINE FEED"
4814  042020  010146            5$:   MOV     R1,-(SP)        ;SAVE R1
4815  042022  012001                  MOV     (R0)+,R1        ;PICKUP "DATA TABLE" POINTER
4816  042024  001415                  BEQ     9$              ;BR IF NO DATA TO BE TYPED
4817  042026  012000                  MOV     (R0)+,R0        ;PICKUP "DATA FORMAT" POINTER
4818  042030  105720            6$:   TSTB    (R0)+           ;"OCTAL" OR "DECIMAL"
4819  042032  001003                  BNE     7$              ;BR IF DECIMAL
4820  042034  013146                  MOV     @(R1)+,-(SP)    ;SAVE @(R1)+ FOR TYPEOUT
4821  042036  104402                  TYPOC                   ;GO TYPE--OCTAL ASCII(ALL DIGITS)
4822  042040  000402                  BR      8$
4823  042042                    7$:
4824  042042  013146                  MOV     @(R1)+,-(SP)    ;SAVE @(R1)+ FOR TYPEOUT
4825  042044  104405                  TYPDS                   ;GO TYPE--DECIMAL ASCII WITH SIGN
4826  042046  005711            8$:   TST     (R1)            ;IS THERE ANOTHER NUMBER?
4827  042050  001403                  BEQ     9$              ;BR IF NO
4828  042052  104401  042066        TYPE    ,11$              ;TYPE TWO(2) SPACES
4829  042056  000764                  BR      6$              ;LOOP
4830
4831  042060  012601            9$:   MOV     (SP)+,R1        ;RESTORE R1
4832  042062  012600            10$:  MOV     (SP)+,R0        ;"CARRIAGE RETURN" & "LINE FEED"
4833  042064  000207                  RTS     PC              ;RETURN
4834  042066  020040      000   11$:  .ASCIZ  /  /            ;TWO(2) SPACES
4835          042072                  .EVEN
```

```
4836
```

```
4837
```

```
                                ;*****************************************************************************
                                ;
                                ;ERROR AND MESSAGE TABLE CONDIMENTS
                                ;
                                ;*****************************************************************************


042566  050122  032060  042040  EM1:    .ASCIZ  /RP04 DID NOT INTERRUPT/
042574  042111  047040  052117
042602  044440  052116  051105
042610  052522  052120  000
042615  111     052116  051105  EM2:    .ASCIZ  /INTERRUPT ENABLE BIT DOWN BUT EXPECTED BIT DID NOT SET/
042622  052522  052120  042440
042630  040516  046102  020105
042636  044502  020124  047504
042644  047127  041040  052125
042652  042440  050130  041505
042660  042524  020104  044502
042666  020124  044504  020104
042674  047516  020124  042523
042702  000124
042704  050122  032060  042040  EM3:    .ASCIZ  /RP04 DID NOT INTERRUPT WHEN EXPECTED BIT DID SET/
042712  042111  047040  052117
042720  044440  052116  051105
042726  052522  052120  053440
042734  042510  020116  054105
042742  042520  052103  042105
042750  041040  052111  042040
042756  042111  051440  052105
042764  000
042765  105     050130  041505  EM4:    .ASCIZ  /EXPECTED BIT DID SET BUT TIME IS IN ERROR - TIME IN 10 MICROSEC. DECIMA
042772  042524  020104  044502
043000  020124  044504  020104
043006  042523  020124  052502
043014  020124  044524  042515
043022  044440  020123  047111
043030  042440  051122  051117
043036  026440  052040  046511
043044  020105  047111  030440
043052  020060  044515  051103
043060  051517  041505  020056
043066  042504  044503  040515
043074  000114
043076  044122  051501  042040  EM5:    .ASCIZ  /RHAS DOES NOT CLEAR BY MOVING IN ALL ONES/
043104  042517  020123  047516
043112  020124  046103  040505
043120  020122  054502  046440
043126  053117  047111  020107
043134  047111  040440  046114
043142  047440  042516  000123
043150  047514  042101  047111  EM6:    .ASCIZ  /LOADING RHER1 FOR ALL UNITS DID NOT SET ANY RHAS BITS/
043156  020107  044122  051105
043164  020061  047506  020122
```

```
4894   043172  046101  020114  047125
4895   043200  052111  020123  044504
4896   043206  020104  047516  020124
4897   043214  042523  020124  047101
4898   043222  020131  044122  051501
4899   043230  041040  052111  000123
4900   043236  047516  020116  054105  EM7:    .ASCIZ  /NON EXISTENT REGISTER, PROGRAM ABORTED./
4901   043244  051511  042524  052116
4902   043252  051040  043505  051511
4903   043260  042524  026122  050040
4904   043266  047522  051107  046501
4905   043274  040440  047502  052122
4906   043302  042105  000056
4907   043306  052123  050117  042520  EM10:   .ASCIZ  /STOPPED DRIVE HAS MOL BIT IN RHDS1 SET/
4908   043314  020104  051104  053111
4909   043322  020105  040510  020123
4910   043330  047515  020114  044502
4911   043336  020124  047111  051040
4912   043344  042110  030523  051440
4913   043352  052105     000
4914
4915   043355     127  052111  020110  EM11:   .ASCIZ  /WITH SPINDLE POWERED DOWN RHCS2 SHOULD ONLY HAVE UNIT NO: AND IR SET/
4916   043362  050123  047111  046104
4917   043370  020105  047520  042527
4918   043376  042522  020104  047504
4919   043404  047127  051040  041510
4920   043412  031123  051440  047510
4921   043420  046125  020104  047117
4922   043426  054514  044040  053101
4923   043434  020105  047125  052111
4924   043442  047040  035117  040440
4925   043450  042116  044440  020122
4926   043456  042523  000124
4927   043462  043101  042524  020122  EM12:   .ASCIZ  /AFTER SPINDLE POWERED UP, NO PACK ACKN. RHDS1 SHOULD HAVE MOL=1, VV=0/
4928   043470  050123  047111  046104
4929   043476  020105  047520  042527
4930   043504  042522  020104  050125
4931   043512  020054  047516  050040
4932   043520  041501  020113  041501
4933   043526  047113  020056  044122
4934   043534  051504  020061  044123
4935   043542  052517  042114  044040
4936   043550  053101  020105  047515
4937   043556  036514  026061  053040
4938   043564  036526  000060
4939   043570  044527  044124  051440  EM13:   .ASCIZ  /WITH SPINDLE POWERED, NO INTIALIZE, RHCS1 SHOULD HAVE GO=0, DVA=1, RDY=
4940   043576  044520  042116  042514
4941   043604  050040  053517  051105
4942   043612  042105  020054  047516
4943   043620  044440  052116  040511
4944   043626  044514  042532  020054
4945   043634  044122  051503  020061
4946   043642  044123  052517  042114
4947   043650  044040  053101  020105
4948   043656  047507  030075  020054
4949   043664  053104  036501  026061
```

```
4950  043672  051040  054504  030475
4951  043700  020054  042511  030075
4952  043706     000
4953  043707     101  052106  051105  EM14:   .ASCIZ  /AFTER SPINDLE POWERED UP RHCC SHOULD BE=0/
4954  043714  051440  044520  042116
4955  043722  042514  050040  053517
4956  043730  051105  042105  052440
4957  043736  020120  044122  041503
4958  043744  051440  047510  046125
4959  043752  020104  042502  030075
4960  043760     000
4961  043761     120  041501  020113  EM15:   .ASCII  /PACK ACKNOWLEDGE COMMAND CAUSED AN ERROR/<15><12>
4962  043766  041501  047113  053517
4963  043774  042514  043504  020105
4964  044002  047503  046515  047101
4965  044010  020104  040503  051525
4966  044016  042105  040440  020116
4967  044024  051105  047522  006522
4968  044032     012
4969  044033     107  047517  020104          .ASCIZ  /GOOD DATA IS BEFORE COMMAND, REC DATA IS AFTER COMMAND/
4970  044040  040504  040524  044440
4971  044046  020123  042502  047506
4972  044054  042522  041440  046517
4973  044062  040515  042116  020054
4974  044070  042522  020103  040504
4975  044076  040524  044440  020123
4976  044104  043101  042524  020122
4977  044112  047503  046515  047101
4978  044120  000104
4979  044122  047516  047455  020120  EM16:   .ASCII  /NO-OP COMMAND CAUSED AN ERROR/<15><12>
4980  044130  047503  046515  047101
4981  044136  020104  040503  051525
4982  044144  042105  040440  020116
4983  044152  051105  047522  006522
4984  044160     012
4985  044161     107  047517  020104          .ASCIZ  /GOOD DATA IS BEFORE COMMAND, REC DATA IS AFTER COMMAND/
4986  044166  040504  040524  044440
4987  044174  020123  042502  047506
4988  044202  042522  041440  046517
4989  044210  040515  042116  020054
4990  044216  042522  020103  040504
4991  044224  040524  044440  020123
4992  044232  043101  042524  020122
4993  044240  047503  046515  047101
4994  044246  000104
4995  044250  051104  053111  020105  EM17:   .ASCII  /DRIVE CLEAR COMMAND CAUSED AN ERROR/<15><12>
4996  044256  046103  040505  020122
4997  044264  047503  046515  047101
4998  044272  020104  040503  051525
4999  044300  042105  040440  020116
5000  044306  051105  047522  006522
5001  044314     012
5002  044315     107  047517  020104          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES AFTER COMMAND/
5003  044322  040504  040524  043440
5004  044330  053111  051505  051440
5005  044336  047510  046125  020104
```

```
5006   044344   042502   020054   042522
5007   044352   020103   040504   040524
5008   044360   043440   053111   051505
5009   044366   040440   052106   051105
5010   044374   041440   046517   040515
5011   044402   042116      000
5012   044405      122   040505   026504   EM20:    .ASCII  /READ-IN COMMAND CAUSED AN ERROR/<15><12>
5013   044412   047111   041440   046517
5014   044420   040515   042116   041440
5015   044426   052501   042523   020104
5016   044434   047101   042440   051122
5017   044442   051117   005015
5018   044446   047507   042117   042040            .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONTENTS AFTER COMMAND/
5019   044454   052101   020101   044507
5020   044462   042526   020123   044123
5021   044470   052517   042114   041040
5022   044476   026105   051040   041505
5023   044504   042040   052101   020101
5024   044512   044507   042526   020123
5025   044520   042522   027107   041440
5026   044526   047117   042524   052116
5027   044534   020123   043101   042524
5028   044542   020122   047503   046515
5029   044550   047101   000104
5030
5031   044554   044122   051503   020061   EM21:    .ASCIZ  /RHCS1 CONTENTS DURING COMMAND WAS IN ERROR/
5032   044562   047503   052116   047105
5033   044570   051524   042040   051125
5034   044576   047111   020107   047503
5035   044604   046515   047101   020104
5036   044612   040527   020123   047111
5037   044620   042440   051122   051117
5038   044626      000
5039   044627      122   042110   030523   EM22:    .ASCIZ  /RHDS1 CONTENTS DURING COMMAND WAS IN ERROR/
5040   044634   041440   047117   042524
5041   044642   052116   020123   052504
5042   044650   044522   043516   041440
5043   044656   046517   040515   042116
5044   044664   053440   051501   044440
5045   044672   020116   051105   047522
5046   044700   000122
5047   044702   047125   047514   042101   EM23:    .ASCII  /UNLOAD COMMAND CAUSED AN ERROR/<15><12>
5048   044710   041440   046517   040515
5049   044716   042116   041440   052501
5050   044724   042523   020104   047101
5051   044732   042440   051122   051117
5052   044740   005015
5053   044742   047507   042117   042040            .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REGISTER CONT. AFTER COMMAND/
5054   044750   052101   020101   044507
5055   044756   042526   020123   044123
5056   044764   052517   042114   041040
5057   044772   026105   051040   041505
5058   045000   042040   052101   020101
5059   045006   044507   042526   020123
5060   045014   042522   044507   052123
5061   045022   051105   041440   047117
```

```
5062   045030   027124   040440   052106
5063   045036   051105   041440   046517
5064   045044   040515   042116      000
5065   045051      117   043106   042523  EM24:   .ASCII  /OFFSET COMMAND CAUSED AN ERROR/<15><12>
5066   045056   020124   047503   046515
5067   045064   047101   020104   040503
5068   045072   051525   042105   040440
5069   045100   020116   051105   047522
5070   045106   006522      012
5071   045111      107   047517   020104          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5072   045116   040504   040524   043440
5073   045124   053111   051505   051440
5074   045132   047510   046125   020104
5075   045140   042502   020054   042522
5076   045146   020103   040504   040524
5077   045154   043440   053111   051505
5078   045162   051040   043505   020056
5079   045170   047503   052116   020056
5080   045176   043101   042524   020122
5081   045204   047503   046515   047101
5082   045212   000104
5083   045214   042522   052524   047122  EM25:   .ASCII  /RETURN TO CENTER LINE COMMAND CAUSED AN ERROR/<15><12>
5084   045222   052040   020117   042503
5085   045230   052116   051105   046040
5086   045236   047111   020105   047503
5087   045244   046515   047101   020104
5088   045252   040503   051525   042105
5089   045260   040440   020116   051105
5090   045266   047522   006522      012
5091   045273      107   047517   020104          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5092   045300   040504   040524   043440
5093   045306   053111   051505   051440
5094   045314   047510   046125   020104
5095   045322   042502   020054   042522
5096   045330   020103   040504   040524
5097   045336   043440   053111   051505
5098   045344   051040   043505   020056
5099   045352   047503   052116   020056
5100   045360   043101   042524   020122
5101   045366   047503   046515   047101
5102   045374   000104
5103   045376   030065   020060   043117  EM26:   .ASCIZ  /500 OFFSET COMMANDS ONE AFTER THE OTHER CAUSED AN ERROR/
5104   045404   051506   052105   041440
5105   045412   046517   040515   042116
5106   045420   020123   047117   020105
5107   045426   043101   042524   020122
5108   045434   044124   020105   052117
5109   045442   042510   020122   040503
5110   045450   051525   042105   040440
5111   045456   020116   051105   047522
5112   045464   000122
5113   045466   051127   052111   020105  EM27:   .ASCII  /WRITE HEADER AND DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5114   045474   042510   042101   051105
5115   045502   040440   042116   042040
5116   045510   052101   020101   040503
5117   045516   051525   042105   044440
```

```
5118   045524   050115   047522   042520
5119   045532   020122   042522   044507
5120   045540   052123   051105   041440
5121   045546   040510   043516   006505
5122   045554      012
5123   045555      107   047517   020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5124   045562   040504   040524   043440
5125   045570   053111   051505   053440
5126   045576   040510   020124   044123
5127   045604   052517   042114   041040
5128   045612   020105   044124   051105
5129   045620   006505      012
5130   045623      122   041505   044505          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5131   045630   042526   020104   040504
5132   045636   040524   043440   053111
5133   045644   051505   053440   040510
5134   045652   020124   040527   020123
5135   045660   044124   051105   020105
5136   045666   043101   042524   020122
5137   045674   047503   046515   047101
5138   045702   000104
5139   045704   051127   052111   020105  EM30:   .ASCIZ  /WRITE HEADER AND DATA CHANGED WRITE FROM BUFFER/
5140   045712   042510   042101   051105
5141   045720   040440   042116   042040
5142   045726   052101   020101   044103
5143   045734   047101   042507   020104
5144   045742   051127   052111   020105
5145   045750   051106   046517   041040
5146   045756   043125   042506   000122
5147
5148   045764   042522   042101   044040  EM31:   .ASCII  /READ HEADER AND DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5149   045772   040505   042504   020122
5150   046000   047101   020104   040504
5151   046006   040524   041440   052501
5152   046014   042523   020104   046511
5153   046022   051120   050117   051105
5154   046030   051040   043505   051511
5155   046036   042524   020122   044103
5156   046044   047101   042507   005015
5157   046052   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5158   046060   052101   020101   044507
5159   046066   042526   020123   044127
5160   046074   052101   051440   047510
5161   046102   046125   020104   042502
5162   046110   052040   042510   042522
5163   046116   005015
5164   046120   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5165   046126   042105   042040   052101
5166   046134   020101   044507   042526
5167   046142   020123   044127   052101
5168   046150   053440   051501   052040
5169   046156   042510   042522   040440
5170   046164   052106   051105   041440
5171   046172   046517   040515   042116
5172   046200      000
5173   046201      127   044522   042524  EM32:   .ASCIZ  /WRITE HEADER DATA FOLLOWED BY READ HEADER AND DATA CAUSED DATA ERROR/
```

```
5174  046206  044040  040505  042504
5175  046214  020122  040504  040524
5176  046222  043040  046117  047514
5177  046230  042527  020104  054502
5178  046236  051040  040505  020104
5179  046244  042510  042101  051105
5180  046252  040440  042116  042040
5181  046260  052101  020101  040503
5182  046266  051525  042105  042040
5183  046274  052101  020101  051105
5184  046302  047522  000122
5185  046306  042522  042101  042040  EM33:   .ASCII  /READ DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5186  046314  052101  020101  040503
5187  046322  051525  042105  044440
5188  046330  050115  047522  042520
5189  046336  020122  042522  044507
5190  046344  052123  051105  041440
5191  046352  040510  043516  006505
5192  046360     012
5193  046361     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5194  046366  040504  040524  043440
5195  046374  053111  051505  053440
5196  046402  040510  020124  044123
5197  046410  052517  042114  041040
5198  046416  020105  044124  051105
5199  046424  006505     012
5200  046427     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5201  046434  042526  020104  040504
5202  046442  040524  043440  053111
5203  046450  051505  053440  040510
5204  046456  020124  040527  020123
5205  046464  044124  051105  020105
5206  046472  043101  042524  020122
5207  046500  047503  046515  047101
5208  046506  000104
5209  046510  042522  042101  042040  EM34:   .ASCIZ  /READ DATA INCORRECT/
5210  046516  052101  020101  047111
5211  046524  047503  051122  041505
5212  046532  000124
5213  046534  051127  052111  020105  EM35:   .ASCII  /WRITE DATA COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
5214  046542  040504  040524  041440
5215  046550  046517  040515  042116
5216  046556  041440  052501  042523
5217  046564  020104  046511  051120
5218  046572  050117  051105  051040
5219  046600  043505  051511  042524
5220  046606  020122  044103  047101
5221  046614  042507  005015
5222  046620  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5223  046626  052101  020101  044507
5224  046634  042526  020123  044127
5225  046642  052101  051440  047510
5226  046650  046125  020104  042502
5227  046656  052040  042510  042522
5228  046664  005015
5229  046666  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
```

```
5230   046674   042105   042040   052101
5231   046702   020101   044507   042526
5232   046710   020123   042522   044507
5233   046716   052123   051105   041440
5234   046724   047117   042524   052116
5235   046732   020123   043101   042524
5236   046740   020122   047503   046515
5237   046746   047101   000104
5238   046752   051127   052111   020105  EM36:   .ASCIZ  /WRITE DATA COMMAND CHANGED WRITE FROM BUFFER/
5239   046760   040504   040524   041440
5240   046766   046517   040515   042116
5241   046774   041440   040510   043516
5242   047002   042105   053440   044522
5243   047010   042524   043040   047522
5244   047016   020115   052502   043106
5245   047024   051105      000
5246   047027      123   042505   020113  EM37:   .ASCII  /SEEK COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
5247   047034   047503   046515   047101
5248   047042   020104   040503   051525
5249   047050   042105   044440   050115
5250   047056   047522   042520   020122
5251   047064   042522   044507   052123
5252   047072   051105   041440   040510
5253   047100   043516   006505      012
5254   047105      107   047517   020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5255   047112   040504   040524   043440
5256   047120   053111   051505   053440
5257   047126   040510   020124   044123
5258   047134   052517   042114   041040
5259   047142   020105   044124   051105
5260   047150   006505      012
5261   047153      122   041505   044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER SEEK COMMAND/
5262   047160   042526   020104   040504
5263   047166   040524   043440   053111
5264   047174   051505   051040   043505
5265   047202   051511   042524   020122
5266   047210   047503   052116   047105
5267   047216   051524   040440   052106
5268   047224   051105   051440   042505
5269   047232   020113   047503   046515
5270   047240   047101   000104
5271   047244   051127   052111   020105  EM40:   .ASCII  /WRITE CHECK CAUSED IMPROPER REGISTER CHANGE/<15><12>
5272   047252   044103   041505   020113
5273   047260   040503   051525   042105
5274   047266   044440   050115   047522
5275   047274   042520   020122   042522
5276   047302   044507   052123   051105
5277   047310   041440   040510   043516
5278   047316   006505      012
5279   047321      107   047517   020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5280   047326   040504   040524   043440
5281   047334   053111   051505   053440
5282   047342   040510   020124   044123
5283   047350   052517   042114   041040
5284   047356   020105   044124   051105
5285   047364   006505      012
```

```
5286  047367    122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
5287  047374 042526  020104  040504
5288  047402 040524  043440  053111
5289  047410 051505  051040  043505
5290  047416 051511  042524  020122
5291  047424 047503  052116  047105
5292  047432 051524  040440  052106
5293  047440 051105  041440  046517
5294  047446 040515  042116     000
5295
5296  047453    114  041517  044513  EM41:   .ASCII  /LOCKING OUT WRITE BY WRITE LOCK BUTTON CAUSED IMPROPER REGISTER CHANGE/
5297  047460 043516  047440  052125
5298  047466 053440  044522  042524
5299  047474 041040  020131  051127
5300  047502 052111  020105  047514
5301  047510 045503  041040  052125
5302  047516 047524  020116  040503
5303  047524 051525  042105  044440
5304  047532 050115  047522  042520
5305  047540 020122  042522  044507
5306  047546 052123  051105  041440
5307  047554 040510  043516  006505
5308  047562    012
5309  047563    107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5310  047570 040504  040524  043440
5311  047576 053111  051505  053440
5312  047604 040510  020124  044123
5313  047612 052517  042114  041040
5314  047620 020105  044124  051105
5315  047626 006505     012
5316  047631    122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER WRITES WERE LOCKED OUT/
5317  047636 042526  020104  040504
5318  047644 040524  043440  053111
5319  047652 051505  051040  043505
5320  047660 051511  042524  020122
5321  047666 047503  052116  047105
5322  047674 051524  040440  052106
5323  047702 051105  053440  044522
5324  047710 042524  020123  042527
5325  047716 042522  046040  041517
5326  047724 042513  020104  052517
5327  047732 000124
5328  047734 052101  042524  050115  EM42:   .ASCII  /ATTEMPTING TO WRITE WITH WRITES LOCKED OUT CAUSED IMPROPER REGISTER CHA
5329  047742 044524  043516  052040
5330  047750 020117  051127  052111
5331  047756 020105  044527  044124
5332  047764 053440  044522  042524
5333  047772 020123  047514  045503
5334  050000 042105  047440  052125
5335  050006 041440  052501  042523
5336  050014 020104  046511  051120
5337  050022 050117  051105  051040
5338  050030 043505  051511  042524
5339  050036 020122  044103  047101
5340  050044 042507  005015
5341  050050 047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
```

```
5342   050056   052101   020101   044507
5343   050064   042526   020123   044127
5344   050072   052101   051440   047510
5345   050100   046125   020104   042502
5346   050106   052040   042510   042522
5347   050114   005015
5348   050116   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED WRITE/
5349   050124   042105   042040   052101
5350   050132   020101   044507   042526
5351   050140   020123   042522   044507
5352   050146   052123   051105   041440
5353   050154   047117   042524   052116
5354   050162   020123   043101   042524
5355   050170   020122   052101   042524
5356   050176   050115   042524   020104
5357   050204   051127   052111   000105
5358   050212   051127   052111   047111  EM43:   .ASCII  /WRITING WITH WRITES LOCKED OUT CHANGED DISK DATA/<15><12>
5359   050220   020107   044527   044124
5360   050226   053440   044522   042524
5361   050234   020123   047514   045503
5362   050242   042105   047440   052125
5363   050250   041440   040510   043516
5364   050256   042105   042040   051511
5365   050264   020113   040504   040524
5366   050272   005015
5367   050274   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT WAS ON DISK BEFORE WRITE WITH WRITE LOCKED OUT/<15
5368   050302   052101   020101   044507
5369   050310   042526   020123   044127
5370   050316   052101   053440   051501
5371   050324   047440   020116   044504
5372   050332   045523   041040   043105
5373   050340   051117   020105   051127
5374   050346   052111   020105   044527
5375   050354   044124   053440   044522
5376   050362   042524   046040   041517
5377   050370   042513   020104   052517
5378   050376   006524      012
5379   050401      127   051501   040440          .ASCII  /WAS ATTEMPTED/<15><12>
5380   050406   052124   046505   052120
5381   050414   042105   005015
5382   050420   042522   042503   053111          .ASCII  /RECEIVED DATA GIVES WHAT WAS READ BACK AFTER WRITE/<15><12>
5383   050426   042105   042040   052101
5384   050434   020101   044507   042526
5385   050442   020123   044127   052101
5386   050450   053440   051501   051040
5387   050456   040505   020104   040502
5388   050464   045503   040440   052106
5389   050472   051105   053440   044522
5390   050500   042524   005015
5391   050504   044527   044124   053440          .ASCIZ  /WITH WRITE LOCKED OUT WAS ATTEMPTED/
5392   050512   044522   042524   046040
5393   050520   041517   042513   020104
5394   050526   052517   020124   040527
5395   050534   020123   052101   042524
5396   050542   050115   042524   000104
5397   050550   047105   041101   044514  EM44:   .ASCII  /ENABLING WRITES BY WRITE LOCK BUTTON CAUSED IMPROPER REGISTER CHANGE/<1
```

```
5398   050556   043516   053440   044522
5399   050564   042524   020123   054502
5400   050572   053440   044522   042524
5401   050600   046040   041517   020113
5402   050606   052502   052124   047117
5403   050614   041440   052501   042523
5404   050622   020104   046511   051120
5405   050630   050117   051105   051040
5406   050636   043505   051511   042524
5407   050644   020122   044103   047101
5408   050652   042507   005015
5409   050656   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5410   050664   052101   020101   044507
5411   050672   042526   020123   044127
5412   050700   052101   051440   047510
5413   050706   046125   020104   042502
5414   050714   052040   042510   042522
5415   050722   005015
5416   050724   042522   042503   053111          .ASCII  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER WRITE LOCK BUTTON/<15><12>
5417   050732   042105   042040   052101
5418   050740   020101   044507   042526
5419   050746   020123   042522   044507
5420   050754   052123   051105   041440
5421   050762   047117   042524   052116
5422   050770   020123   043101   042524
5423   050776   020122   051127   052111
5424   051004   020105   047514   045503
5425   051012   041040   052125   047524
5426   051020   006516      012
5427   051023      105   040516   046102          .ASCIZ  /ENABLED WRITES/
5428   051030   042105   053440   044522
5429   051036   042524   000123
5430   051042   051124   047101   043123  EM45:   .ASCII  /TRANSFERRING ON LAST BLOCK - CYLINDER 410. - 814., SECTOR 21, /<15><12>
5431   051050   051105   044522   043516
5432   051056   047440   020116   040514
5433   051064   052123   041040   047514
5434   051072   045503   026440   041440
5435   051100   046131   047111   042504
5436   051106   020122   030464   027060
5437   051114   026440   034040   032061
5438   051122   026056   051440   041505
5439   051130   047524   020122   030462
5440   051136   020054   005015
5441   051142   051124   041501   020113          .ASCII  /TRACK 18, CAUSED IMPROPER REGISTER CHANGE/<15><12>
5442   051150   034061   020054   040503
5443   051156   051525   042105   044440
5444   051164   050115   047522   042520
5445   051172   020122   042522   044507
5446   051200   052123   051105   041440
5447   051206   040510   043516   006505
5448   051214      012
5449   051215      107   047517   020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5450   051222   040504   040524   043440
5451   051230   053111   051505   053440
5452   051236   040510   020124   044123
5453   051244   052517   042114   041040
```

```
5454   051252   020105   044124   051105
5455   051260   006505      012
5456   051263      122   041505   044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER TRANSFER/
5457   051270   042526   020104   040504
5458   051276   040524   043440   053111
5459   051304   051505   051040   043505
5460   051312   051511   042524   020122
5461   051320   047503   052116   047105
5462   051326   051524   040440   052106
5463   051334   051105   052040   040522
5464   051342   051516   042506   000122
5465   051350   040504   040524   051040  EM46:   .ASCII  /DATA READ FROM LAST BLOCK - CYLINDER 410. - 814., SECTOR 21,/<15><12>
5466   051356   040505   020104   051106
5467   051364   046517   046040   051501
5468   051372   020124   046102   041517
5469   051400   020113   020055   054503
5470   051406   044514   042116   051105
5471   051414   032040   030061   020056
5472   051422   020055   030470   027064
5473   051430   020054   042523   052103
5474   051436   051117   031040   026061
5475   051444   005015
5476   051446   051124   041501   020113          .ASCIZ  /TRACK 18, IS IN ERROR/
5477   051454   034061   020054   051511
5478   051462   044440   020116   051105
5479   051470   047522   000122
5480   051474   051124   047101   043123  EM47:   .ASCII  /TRANSFERRING DATA FROM NONEXISTANT SECTOR CAUSED IMPROPER /<15><12>
5481   051502   051105   044522   043516
5482   051510   042040   052101   020101
5483   051516   051106   046517   047040
5484   051524   047117   054105   051511
5485   051532   040524   052116   051440
5486   051540   041505   047524   020122
5487   051546   040503   051525   042105
5488   051554   044440   050115   047522
5489   051562   042520   020122   005015
5490   051570   042522   044507   052123          .ASCII  /REGISTER CHANGE, GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5491   051576   051105   041440   040510
5492   051604   043516   026105   043440
5493   051612   047517   020104   040504
5494   051620   040524   043440   053111
5495   051626   051505   053440   040510
5496   051634   020124   044123   052517
5497   051642   042114   041040   020105
5498   051650   044124   051105   006505
5499   051656      012
5500   051657      122   041505   044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED TRANSFER/
5501   051664   042526   020104   040504
5502   051672   040524   043440   053111
5503   051700   051505   051040   043505
5504   051706   051511   042524   020122
5505   051714   047503   052116   047105
5506   051722   051524   040440   052106
5507   051730   051105   040440   052124
5508   051736   046505   052120   042105
5509   051744   052040   040522   051516
```

```
5510  051752  042506  000122
5511  051756  051124  047101  043123  EM50:   .ASCII  /TRANSFERRING FROM NONEXISTANT SECTOR CAUSED DATA ERROR/<15><12>
5512  051764  051105  044522  043516
5513  051772  043040  047522  020115
5514  052000  047516  042516  044530
5515  052006  052123  047101  020124
5516  052014  042523  052103  051117
5517  052022  041440  052501  042523
5518  052030  020104  040504  040524
5519  052036  042440  051122  051117
5520  052044  005015
5521  052046  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5522  052054  052101  020101  044507
5523  052062  042526  020123  044127
5524  052070  052101  051440  047510
5525  052076  046125  020104  042502
5526  052104  052040  042510  042522
5527  052112  005015
5528  052114  040502  020104  040504          .ASCIZ  /BAD DATA GIVES WHAT WAS IN BUFFER AFTER TRANSFER/
5529  052122  040524  043440  053111
5530  052130  051505  053440  040510
5531  052136  020124  040527  020123
5532  052144  047111  041040  043125
5533  052152  042506  020122  043101
5534  052160  042524  020122  051124
5535  052166  047101  043123  051105
5536  052174     000
5537
5538  052175     107  053111  047111  EM51:   .ASCII  /GIVING ILLEGAL FUNCTION CAUSED IMPROPER REGISTER CHANGE/<15><12>
5539  052202  020107  046111  042514
5540  052210  040507  020114  052506
5541  052216  041516  044524  047117
5542  052224  041440  052501  042523
5543  052232  020104  046511  051120
5544  052240  050117  051105  051040
5545  052246  043505  051511  042524
5546  052254  020122  044103  047101
5547  052262  042507  005015
5548  052266  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5549  052274  052101  020101  044507
5550  052302  042526  020123  044127
5551  052310  052101  051440  047510
5552  052316  046125  020104  042502
5553  052324  052040  042510  042522
5554  052332  005015
5555  052334  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ILLEGAL FUNCTION IS GIVEN/
5556  052342  042105  042040  052101
5557  052350  020101  044507  042526
5558  052356  020123  042522  044507
5559  052364  052123  051105  041440
5560  052372  047117  042524  052116
5561  052400  020123  043101  042524
5562  052406  020122  046111  042514
5563  052414  040507  020114  052506
5564  052422  041516  044524  047117
5565  052430  044440  020123  044507
```

```
5566  052436  042526  000116
5567  052442  051127  052111  020105  EM52:   .ASCII  /WRITE DATA ON NONEXISTANT SECTOR CAUSED IMPROPER REGISTER CHANGE/<15><1
5568  052450  040504  040524  047440
5569  052456  020116  047516  042516
5570  052464  044530  052123  047101
5571  052472  020124  042523  052103
5572  052500  051117  041440  052501
5573  052506  042523  020104  046511
5574  052514  051120  050117  051105
5575  052522  051040  043505  051511
5576  052530  042524  020122  044103
5577  052536  047101  042507  005015
5578  052544  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5579  052552  052101  020101  044507
5580  052560  042526  020123  044127
5581  052566  052101  051440  047510
5582  052574  046125  020104  042502
5583  052602  052040  042510  042522
5584  052610  005015
5585  052612  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED WRITE DATA/
5586  052620  042105  042040  052101
5587  052626  020101  044507  042526
5588  052634  020123  042522  044507
5589  052642  052123  051105  041440
5590  052650  047117  042524  052116
5591  052656  020123  043101  042524
5592  052664  020122  052101  042524
5593  052672  050115  042524  020104
5594  052700  051127  052111  020105
5595  052706  040504  040524     000
5596  052713     122  040505  020104  EM53:   .ASCIZ  /READ HEADER AND DATA AFTER A SEARCH CAUSED DATA ERROR/
5597  052720  042510  042101  051105
5598  052726  040440  042116  042040
5599  052734  052101  020101  043101
5600  052742  042524  020122  020101
5601  052750  042523  051101  044103
5602  052756  041440  052501  042523
5603  052764  020104  040504  040524
5604  052772  042440  051122  051117
5605  053000     000
5606  053001     101  052124  046505  EM54:   .ASCII  /ATTEMPTING COMMAND WITH INVALID ADDRESS CAUSED IMPROPER REGISTER CHANGE
5607  053006  052120  047111  020107
5608  053014  047503  046515  047101
5609  053022  020104  044527  044124
5610  053030  044440  053116  046101
5611  053036  042111  040440  042104
5612  053044  042522  051523  041440
5613  053052  052501  042523  020104
5614  053060  046511  051120  050117
5615  053066  051105  051040  043505
5616  053074  051511  042524  020122
5617  053102  044103  047101  042507
5618  053110  005015
5619  053112  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5620  053120  052101  020101  044507
5621  053126  042526  020123  044127
```

```
5622   053134   052101   051440   047510
5623   053142   046125   020104   042502
5624   053150   052040   042510   042522
5625   053156   005015
5626   053160   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION/
5627   053166   042105   042040   052101
5628   053174   020101   044507   042526
5629   053202   020123   042522   044507
5630   053210   052123   051105   041440
5631   053216   047117   042524   052116
5632   053224   020123   043101   042524
5633   053232   020122   050117   051105
5634   053240   052101   047511   000116
5635   053246   051127   052111   047111  EM55:   .ASCII  /WRITING OR READING WITH EXPECTED ADDRESS OVERFLOW ERROR/<15><12>
5636   053254   020107   051117   051040
5637   053262   040505   044504   043516
5638   053270   053440   052111   020110
5639   053276   054105   042520   052103
5640   053304   042105   040440   042104
5641   053312   042522   051523   047440
5642   053320   042526   043122   047514
5643   053326   020127   051105   047522
5644   053334   006522      012
5645   053337      103   052501   042523          .ASCII  /CAUSED IMPROPER REGISTER CHANGE/<15><12>
5646   053344   020104   046511   051120
5647   053352   050117   051105   051040
5648   053360   043505   051511   042524
5649   053366   020122   044103   047101
5650   053374   042507   005015
5651   053400   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5652   053406   052101   020101   044507
5653   053414   042526   020123   044127
5654   053422   052101   051440   047510
5655   053430   046125   020104   042502
5656   053436   052040   042510   042522
5657   053444   005015
5658   053446   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION/
5659   053454   042105   042040   052101
5660   053462   020101   044507   042526
5661   053470   020123   042522   044507
5662   053476   052123   051105   041440
5663   053504   047117   042524   052116
5664   053512   020123   043101   042524
5665   053520   020122   050117   051105
5666   053526   052101   047511   000116
5667   053534   040504   040524   051040  EM56:   .ASCII  /DATA READ WITH AN EXPECTED ADDRESS OVERFLOW ERROR IS INCORRECT/<15><12>
5668   053542   040505   020104   044527
5669   053550   044124   040440   020116
5670   053556   054105   042520   052103
5671   053564   042105   040440   042104
5672   053572   042522   051523   047440
5673   053600   042526   043122   047514
5674   053606   020127   051105   047522
5675   053614   020122   051511   044440
5676   053622   041516   051117   042522
5677   053630   052103   005015
```

```
5678   053634   047527   042122   047040          .ASCII  /WORD NO. 1 TO 260 SHOULD BE READ, WORD NO 261 TO 266 SHOULD/<15><12>
5679   053642   027117   030440   052040
5680   053650   020117   033062   020060
5681   053656   044123   052517   042114
5682   053664   041040   020105   042522
5683   053672   042101   020054   047527
5684   053700   042122   047040   020117
5685   053706   033062   020061   047524
5686   053714   031040   033066   051440
5687   053722   047510   046125   006504
5688   053730      012
5689   053731      102   020105   044103          .ASCIZ  /BE CHANGED/
5690   053736   047101   042507   000104
5691   053744   052101   042524   050115  EM57:   .ASCII  /ATTEMPTING DATA COMMAND WITH WRONG FORMAT BIT CAUSED/<15><12>
5692   053752   044524   043516   042040
5693   053760   052101   020101   047503
5694   053766   046515   047101   020104
5695   053774   044527   044124   053440
5696   054002   047522   043516   043040
5697   054010   051117   040515   020124
5698   054016   044502   020124   040503
5699   054024   051525   042105   005015
5700   054032   046511   051120   050117          .ASCII  /IMPROPER REGISTER CHANGE/<15><12>
5701   054040   051105   051040   043505
5702   054046   051511   042524   020122
5703   054054   044103   047101   042507
5704   054062   005015
5705   054064   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5706   054072   052101   020101   044507
5707   054100   042526   020123   044127
5708   054106   052101   051440   047510
5709   054114   046125   020104   042502
5710   054122   052040   042510   042522
5711   054130   005015
5712   054132   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED DATA TRANSFER/
5713   054140   042105   042040   052101
5714   054146   020101   044507   042526
5715   054154   020123   042522   044507
5716   054162   052123   051105   041440
5717   054170   047117   042524   052116
5718   054176   020123   043101   042524
5719   054204   020122   052101   042524
5720   054212   050115   042524   020104
5721   054220   040504   040524   052040
5722   054226   040522   051516   042506
5723   054234   000122
5724   054236   052101   042524   050115  EM60:   .ASCII  /ATTEMPTING TO MODIFY REGISTER DURING AN OPERATION CAUSED IMPROPER/<15><
5725   054244   044524   043516   052040
5726   054252   020117   047515   044504
5727   054260   054506   051040   043505
5728   054266   051511   042524   020122
5729   054274   052504   044522   043516
5730   054302   040440   020116   050117
5731   054310   051105   052101   047511
5732   054316   020116   040503   051525
5733   054324   042105   044440   050115
```

```
 5734   054332   047522   042520   006522
 5735   054340      012
 5736   054341      122   043505   051511          .ASCII  /REGISTER CHANGE. GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
 5737   054346   042524   020122   044103
 5738   054354   047101   042507   020056
 5739   054362   047507   042117   042040
 5740   054370   052101   020101   044507
 5741   054376   042526   020123   044127
 5742   054404   052101   051440   047510
 5743   054412   046125   020104   042502
 5744   054420   052040   042510   042522
 5745   054426   005015
 5746   054430   042522   042503   053111          .ASCII  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION WAS ATTEMPTED/<15
 5747   054436   042105   042040   052101
 5748   054444   020101   044507   042526
 5749   054452   020123   042522   044507
 5750   054460   052123   051105   041440
 5751   054466   047117   042524   052116
 5752   054474   020123   043101   042524
 5753   054502   020122   050117   051105
 5754   054510   052101   047511   020116
 5755   054516   040527   020123   052101
 5756   054524   042524   050115   042524
 5757   054532   006504      012
 5758   054535      115   042117   044506          .ASCIZ  /MODFING REG GIVES ADDRESS OF REGISTER BEING MODIFIED WHICH CAUSED ERROR
 5759   054542   043516   051040   043505
 5760   054550   043440   053111   051505
 5761   054556   040440   042104   042522
 5762   054564   051523   047440   020106
 5763   054572   042522   044507   052123
 5764   054600   051105   041040   044505
 5765   054606   043516   046440   042117
 5766   054614   043111   042511   020104
 5767   054622   044127   041511   020110
 5768   054630   040503   051525   042105
 5769   054636   042440   051122   051117
 5770   054644      000
 5771
 5772   054645      104   053105   041511  EM61:   .ASCIZ  /DEVICE NOT AVAILABLE BEFORE COMMAND WAS TO BE GIVEN/
 5773   054652   020105   047516   020124
 5774   054660   053101   044501   040514
 5775   054666   046102   020105   042502
 5776   054674   047506   042522   041440
 5777   054702   046517   040515   042116
 5778   054710   053440   051501   052040
 5779   054716   020117   042502   043440
 5780   054724   053111   047105      000
 5781   054731      122   042110   030523  EM63:   .ASCIZ  /RHDS1 CONTENTS DURING COMMAND WAS IN ERROR/
 5782   054736   041440   047117   042524
 5783   054744   052116   020123   052504
 5784   054752   044522   043516   041440
 5785   054760   046517   040515   042116
 5786   054766   053440   051501   044440
 5787   054774   020116   051105   047522
 5788   055002   000122
 5789   055004   042522   040503   044514  EM64:   .ASCII  /RECALIBRATE COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
```

```
5790  055012  051102  052101  020105
5791  055020  047503  046515  047101
5792  055026  020104  040503  051525
5793  055034  042105  044440  050115
5794  055042  047522  042520  020122
5795  055050  042522  044507  052123
5796  055056  051105  041440  040510
5797  055064  043516  006505     012
5798  055071     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5799  055076  040504  040524  043440
5800  055104  053111  051505  053440
5801  055112  040510  020124  044123
5802  055120  052517  042114  041040
5803  055126  020105  044124  051105
5804  055134  006505     012
5805  055137     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
5806  055144  042526  020104  040504
5807  055152  040524  043440  053111
5808  055160  051505  051040  043505
5809  055166  051511  042524  020122
5810  055174  047503  052116  047105
5811  055202  051524  040440  052106
5812  055210  051105  041440  046517
5813  055216  040515  042116     000
5814  055223     111  052116  051105  EM65:   .ASCIZ  /INTERRUPT FAILING/
5815  055230  052522  052120  043040
5816  055236  044501  044514  043516
5817  055244     000
5818  055245     110  040505  042504  EM66:   .ASCII  /HEADER AND DATA COMMAND FOR HEAD SELECTION TEST/<15><12>
5819  055252  020122  047101  020104
5820  055260  040504  040524  041440
5821  055266  046517  040515  042116
5822  055274  043040  051117  044040
5823  055302  040505  020104  042523
5824  055310  042514  052103  047511
5825  055316  020116  042524  052123
5826  055324  005015
5827  055326  040503  051525  042105          .ASCII  /CAUSED ERROR/<15><12>
5828  055334  042440  051122  051117
5829  055342  005015
5830  055344  044122  051504  020124          .ASCII  /RHDST GIVES WHAT TRACK WAS BEING WRITTEN OR READ/<15><12>
5831  055352  044507  042526  020123
5832  055360  044127  052101  052040
5833  055366  040522  045503  053440
5834  055374  051501  041040  044505
5835  055402  043516  053440  044522
5836  055410  052124  047105  047440
5837  055416  020122  042522  042101
5838  055424  005015
5839  055426  047117  041440  046131          .ASCIZ  /ON CYLINDER 0, SECTOR 0/
5840  055434  047111  042504  020122
5841  055442  026060  051440  041505
5842  055450  047524  020122  000060
5843  055456  042522  042101  044040  EM67:   .ASCII  /READ HEADER AND DATA ERROR IN HEAD SELECTION TEST/<12><15>
5844  055464  040505  042504  020122
5845  055472  047101  020104  040504
```

```
5846   055500   040524   042440   051122
5847   055506   051117   044440   020116
5848   055514   042510   042101   051440
5849   055522   046105   041505   044524
5850   055530   047117   052040   051505
5851   055536   005124      015
5852   055541      106   051111   052123          .ASCII  /FIRST FOUR WORD NUMBERS ARE HEADER/<12><15>
5853   055546   043040   052517   020122
5854   055554   047527   042122   047040
5855   055562   046525   042502   051522
5856   055570   040440   042522   044040
5857   055576   040505   042504   005122
5858   055604      015
5859   055605      127   051117   020104          .ASCII  /WORD NUMBERS 5 TO 260 ARE DATA WORDS/<12><15>
5860   055612   052516   041115   051105
5861   055620   020123   020065   047524
5862   055626   031040   030066   040440
5863   055634   042522   042040   052101
5864   055642   020101   047527   042122
5865   055650   005123      015
5866   055653      111   020116   040504          .ASCIZ  /IN DATA WORDS BITS 4,5,6,7,8 GIVE TRACK NUMBER/
5867   055660   040524   053440   051117
5868   055666   051504   041040   052111
5869   055674   020123   026064   026065
5870   055702   026066   026067   020070
5871   055710   044507   042526   052040
5872   055716   040522   045503   047040
5873   055724   046525   042502   000122
5874
5875   055732   042522   042101   044040  EM70:   .ASCII  /READ HEADER AND DATA ERROR IN/<15><12>
5876   055740   040505   042504   020122
5877   055746   047101   020104   040504
5878   055754   040524   042440   051122
5879   055762   051117   044440   006516
5880   055770      012
5881   055771      104   043111   042506          .ASCII  /DIFFERENCE LINE TEST/<15><12>
5882   055776   042522   041516   020105
5883   056004   044514   042516   052040
5884   056012   051505   006524      012
5885   056017      127   051117   020104          .ASCII  /WORD NOS 1-4 GIVE HEADER/<15><12>
5886   056024   047516   020123   026461
5887   056032   020064   044507   042526
5888   056040   044040   040505   042504
5889   056046   006522      012
5890   056051      127   051117   020104          .ASCIZ  /WORD NOS 5-260 GIVE DATA WHICH IS THE CYLINDER ADDRESS/
5891   056056   047516   020123   026465
5892   056064   033062   020060   044507
5893   056072   042526   042040   052101
5894   056100   020101   044127   041511
5895   056106   020110   051511   052040
5896   056114   042510   041440   046131
5897   056122   047111   042504   020122
5898   056130   042101   051104   051505
5899   056136   000123
5900   056140   047506   041522   047111  EM71:   .ASCII  /FORCING OPI BY 3 INDEX PULSES/<15><12>
5901   056146   020107   050117   020111
```

```
5902   056154   054502   031440   044440
5903   056162   042116   054105   050040
5904   056170   046125   042523   006523
5905   056176      012
5906   056177      103   052501   042523          .ASCII  /CAUSED IMPROPER REGISTER CHANGE/<15><12>
5907   056204   020104   046511   051120
5908   056212   050117   051105   051040
5909   056220   043505   051511   042524
5910   056226   020122   044103   047101
5911   056234   042507   005015
5912   056240   047507   042117   042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5913   056246   052101   020101   044507
5914   056254   042526   020123   044127
5915   056262   052101   051440   047510
5916   056270   046125   020104   042502
5917   056276   052040   042510   042522
5918   056304   005015
5919   056306   042522   042503   053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER 3 INDEX PULSES/
5920   056314   042105   042040   052101
5921   056322   020101   044507   042526
5922   056330   020123   042522   044507
5923   056336   052123   051105   041440
5924   056344   047117   042524   052116
5925   056352   020123   043101   042524
5926   056360   020122   020063   047111
5927   056366   042504   020130   052520
5928   056374   051514   051505      000
5929   056401      124   042510   042522  EM72:   .ASCII  /THERE WAS A SETUP ERROR DURING MULTIPLE WRITE/<15><12>
5930   056406   053440   051501   040440
5931   056414   051440   052105   050125
5932   056422   042440   051122   051117
5933   056430   042040   051125   047111
5934   056436   020107   052515   052114
5935   056444   050111   042514   053440
5936   056452   044522   042524   005015
5937   056460   042510   042101   051105          .ASCII  /HEADER AND DATA COMMANDS RESULTING IN AN ABORT/<15><12>
5938   056466   040440   042116   042040
5939   056474   052101   020101   047503
5940   056502   046515   047101   051504
5941   056510   051040   051505   046125
5942   056516   044524   043516   044440
5943   056524   020116   047101   040440
5944   056532   047502   052122   005015
5945   056540   043117   052040   044510          .ASCII  /OF THIS 'OPI' TEST./<15><12><15><12>
5946   056546   020123   047447   044520
5947   056554   020047   042524   052123
5948   056562   006456   006412      012
5949   056567      124   020117   051124          .ASCIZ  /TO TROUBLE SHOOT SETUP ERROR, LOOP ON THIS TEST/
5950   056574   052517   046102   020105
5951   056602   044123   047517   020124
5952   056610   042523   052524   020120
5953   056616   051105   047522   026122
5954   056624   046040   047517   020120
5955   056632   047117   052040   044510
5956   056640   020123   042524   052123
5957   056646      000
```

```
5958  056647    122  040505  020104  EM73:   .ASCII  /READ HEADER AND DATA FOR 11960 WORDS /<15><12>
5959  056654 042510  042101  051105
5960  056662 040440  042116  042040
5961  056670 052101  020101  047506
5962  056676 020122  030461  033071
5963  056704 020060  047527  042122
5964  056712 020123  005015
5965  056716 044124  052101  044440          .ASCII  /THAT IS 46 SECTORS /<15><12>
5966  056724 020123  033064  051440
5967  056732 041505  047524  051522
5968  056740 006440     012
5969  056743    124  040510  020124          .ASCIZ  /THAT IS OVER 3 INDEX PULSES CAUSED AN ERROR/
5970  056750 051511  047440  042526
5971  056756 020122  020063  047111
5972  056764 042504  020130  052520
5973  056772 051514  051505  041440
5974  057000 052501  042523  020104
5975  057006 047101  042440  051122
5976  057014 051117     000
5977  057017    122  040505  020104  EM74:   .ASCII  /READ HEADER AND DATA FOR 11960 WORDS /<15><12>
5978  057024 042510  042101  051105
5979  057032 040440  042116  042040
5980  057040 052101  020101  047506
5981  057046 020122  030461  033071
5982  057054 020060  047527  042122
5983  057062 020123  005015
5984  057066 044124  052101  044440          .ASCII  /THAT IS 46 SECTORS, THAT IS OVER 3 INDEX /<15><12>
5985  057074 020123  033064  051440
5986  057102 041505  047524  051522
5987  057110 020054  044124  052101
5988  057116 044440  020123  053117
5989  057124 051105  031440  044440
5990  057132 042116  054105  006440
5991  057140     012
5992  057141    120  046125  042523          .ASCIZ  /PULSES CAUSED OPI TO SET/
5993  057146 020123  040503  051525
5994  057154 042105  047440  044520
5995  057162 052040  020117  042523
5996  057170 000124
```

```
5997
5998  057172  040504  040524  020114  CPHALT: .ASCII  /FATAL ERROR - SEE DOCUMENT LISTING/<15><12>
5999  057200  051105  047522  020122
6000  057206  020055  042523  020105
6001  057214  047504  052503  042515
6002  057222  052116  046040  051511
6003  057230  044524  043516  005015
6004  057236  006440  103412  177777          .ASCII  / /<15><12><207><377><377><207><377><377><207><377><377>
6005  057244  177607  103777  177777
6006  057252  044124  020105  047503          .ASCII  /THE CONTROLLER OR DEVICE HAS GONE OFFLINE, LOST/<15><12>
6007  057260  052116  047522  046114
6008  057266  051105  047440  020122
6009  057274  042504  044526  042503
6010  057302  044040  051501  043440
6011  057310  047117  020105  043117
6012  057316  046106  047111  026105
6013  057324  046040  051517  006524
6014  057332     012
6015  057333     047  042522  042101          .ASCII  /'READY', BECOME UNAVAILABLE, OR HAS STATUS BITS/<15><12>
6016  057340  023531  020054  042502
6017  057346  047503  042515  052440
6018  057354  040516  040526  046111
6019  057362  041101  042514  020054
6020  057370  051117  044040  051501
6021  057376  051440  040524  052524
6022  057404  020123  044502  051524
6023  057412  005015
6024  057414  044127  041511  020110          .ASCIZ  /WHICH CANNOT BE CLEARED/
6025  057422  040503  047116  052117
6026  057430  041040  020105  046103
6027  057436  040505  042522  000104
6028
6029
6030
6031  057444  041520  020040  020040  DH1:    .ASCII  /PC      TEST    WAIT    BIT     REG     REG     RHCS1/<15><12>
6032  057452  020040  042524  052123
6033  057460  020040  020040  040527
6034  057466  052111  020040  020040
6035  057474  044502  020124  020040
6036  057502  020040  042522  020107
6037  057510  020040  020040  042522
6038  057516  020107  020040  020040
6039  057524  044122  051503  006461
6040  057532     012
6041  057533     040  020040  020040          .ASCIZ  /       NO      PC      EXPCTD  ADDRESS CONTENT CONTENT         /
6042  057540  020040  047040  020117
6043  057546  020040  020040  050040
6044  057554  020103  020040  020040
6045  057562  042440  050130  052103
6046  057570  020104  040440  042104
6047  057576  042522  051523  041440
6048  057604  047117  042524  052116
6049  057612  041440  047117  042524
6050  057620  052116  000011
6051  057624  041520  020040  020040  DH4:    .ASCII  /PC      TEST    WAIT    BIT     REG     TIME IN/<15><12>
6052  057632  020040  042524  052123
```

```
6053   057640   020040   020040   040527
6054   057646   052111   020040   020040
6055   057654   044502   020124   020040
6056   057662   020040   042522   020107
6057   057670   020040   020040   044524
6058   057676   042515   044440   006516
6059   057704      012
6060   057705      040   020040   020040          .ASCIZ  /       NO      PC      EXPCTD  ADDRESS 10 MSEC/
6061   057712   020040   047040   020117
6062   057720   020040   020040   050040
6063   057726   020103   020040   020040
6064   057734   042440   050130   052103
6065   057742   020104   040440   042104
6066   057750   042522   051523   030440
6067   057756   020060   051515   041505
6068   057764      000
6069   057765      120   020103   020040  DH5:    .ASCII  /PC      TEST    REG     GOOD    RECEIVED/<15><12>
6070   057772   020040   052040   051505
6071   060000   020124   020040   051040
6072   060006   043505   020040   020040
6073   060014   043440   047517   020104
6074   060022   020040   051040   041505
6075   060030   044505   042526   006504
6076   060036      012
6077   060037      040   020040   020040          .ASCIZ  /       NO      ADDRESS DATA    DATA/
6078   060044   020040   047040   020117
6079   060052   020040   020040   040440
6080   060060   042104   042522   051523
6081   060066   042040   052101   020101
6082   060074   020040   042040   052101
6083   060102   000101
6084   060104   041520   020040   020040  DH6:    .ASCII  /PC      TEST    REG     RECEIVED/<15><12>
6085   060112   020040   042524   052123
6086   060120   020040   020040   042522
6087   060126   020107   020040   020040
6088   060134   042522   042503   053111
6089   060142   042105   005015
6090   060146   020040   020040   020040          .ASCIZ  /       NO      ADDRESS DATA/
6091   060154   020040   047516   020040
6092   060162   020040   020040   042101
6093   060170   051104   051505   020123
6094   060176   040504   040524      000
6095   060203      120   020103   020040  DH7:    .ASCIZ  /PC      TEST    REG     ADDRESS/
6096   060210   020040   052040   051505
6097   060216   020124   020040   051040
6098   060224   043505   020040   020040
6099   060232   040440   042104   042522
6100   060240   051523      000
6101
6102   060243      120   020103   020040  DH10:   .ASCII  /PC      TEST    FAILING CONTENT CONTENT CONTENT CONTENT/<15><12>
6103   060250   020040   052040   051505
6104   060256   020124   020040   043040
6105   060264   044501   044514   043516
6106   060272   041440   047117   042524
6107   060300   052116   041440   047117
6108   060306   042524   052116   041440
```

```
6109   060314  047117  042524  052116
6110   060322  041440  047117  042524
6111   060330  052116  005015
6112   060334  020040  020040  020040          .ASCIZ  /       NO      REG.    RHCS1   RHCS2   RHDS1   RHER1/
6113   060342  020040  047516  020040
6114   060350  020040  020040  042522
6115   060356  027107  020040  020040
6116   060364  044122  051503  020061
6117   060372  020040  044122  051503
6118   060400  020062  020040  044122
6119   060406  051504  020061  020040
6120   060414  044122  051105  000061
6121
6122   060422  041520  020040  020040  DH26:   .ASCII  /PC      TEST    CONT OF CONT OF CONT OF CONT OF CONT OF CONT OF/<15><12
6123   060430  020040  042524  052123
6124   060436  020040  020040  047503
6125   060444  052116  047440  020106
6126   060452  047503  052116  047440
6127   060460  020106  047503  052116
6128   060466  047440  020106  047503
6129   060474  052116  047440  020106
6130   060502  047503  052116  047440
6131   060510  020106  047503  052116
6132   060516  047440  006506    012
6133   060523     040  020040  020040          .ASCIZ  /       NO      RHCS1   RHCS2   RHDS1   RHER1   RHER2   RHER3   /
6134   060530  020040  047040  020117
6135   060536  020040  020040  051040
6136   060544  041510  030523  020040
6137   060552  051040  041510  031123
6138   060560  020040  051040  042110
6139   060566  030523  020040  051040
6140   060574  042510  030522  020040
6141   060602  051040  042510  031122
6142   060610  020040  051040  042510
6143   060616  031522  000011
6144
6145   060622  041520  020040  020040  DH30:   .ASCII  /PC      TEST    WORD    GOOD    BAD/<15><12>
6146   060630  020040  042524  052123
6147   060636  020040  020040  047527
6148   060644  042122  020040  020040
6149   060652  047507  042117  020040
6150   060660  020040  040502  006504
6151   060666     012
6152   060667     040  020040  020040          .ASCIZ  /       NO      NO      DATA    DATA/
6153   060674  020040  047040  020117
6154   060702  020040  020040  047040
6155   060710  020117  020040  020040
6156   060716  042040  052101  020101
6157   060724  020040  042040  052101
6158   060732  000101
6159
6160   060734  041520  020040  020040  DH51:   .ASCII  /PC      TEST    REG     GOOD    RECVD   ILLEGL/<15><12>
6161   060742  020040  042524  052123
6162   060750  020040  020040  042522
6163   060756  020107  020040  020040
6164   060764  047507  042117  020040
```

```
                                                
```

```
6221  061456  020040  020040  051040
6222  061464  041510  030523  020040
6223  061472  051040  040510  020123
6224  061500  020040  051040  042110
6225  061506  030523    000
6226  061511    120   020103  020040  DH66:   .ASCIZ  /PC      TEST    RHDST   RHER1   RHER2   RHER3   RHCS1   RHCS2/
6227  061516  020040  052040  051505
6228  061524  020124  020040  051040
6229  061532  042110  052123  020040
6230  061540  051040  042510  030522
6231  061546  020040  051040  042510
6232  061554  031122  020040  051040
6233  061562  042510  031522  020040
6234  061570  051040  041510  030523
6235  061576  020040  051040  041510
6236  061604  031123    000
6237
6238  061607    120   020103  020040  DH72:   .ASCIZ  /PC      TEST    RHCSI   RHCS2   RHDS1   RHDST   RHCA    RHER1   RHWC/
6239  061614  020040  052040  051505
6240  061622  020124  020040  051040
6241  061630  041510  044523  020040
6242  061636  051040  041510  031123
6243  061644  020040  051040  042110
6244  061652  030523  020040  051040
6245  061660  042110  052123  020040
6246  061666  051040  041510  020101
6247  061674  020040  051040  042510
6248  061702  030522  020040  051040
6249  061710  053510  000103
6250
6251
6252                                  .EVEN
6253
6254  061714  001116  004504  033316  DT1:    .WORD   $ERRPC,TSTNM,WAITPC,WAITBT,WAITRE,$BDDAT,CS1,0
6255  061722  033322  033320  001126
6256  061730  002262  000000
6257  061734  001116  004504  004504  DT4:    .WORD   $ERRPC,TSTNM,TSTNM,WAITPC,WAITBT,WAITRE,$BDDAT,WAITTM,0
6258  061742  033316  033322  033320
6259  061750  001126  033324  000000
6260  061756  001116  004504  004500  DT5:    .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,0
6261  061764  001124  001126  000000
6262  061772  001116  004504  004500  DT6:    .WORD   $ERRPC,TSTNM,REGADR,$BDDAT,0
6263  062000  001126  000000
6264  062004  001116  004504  001200  DT7:    .WORD   $ERRPC,TSTNM,$TMP1,0
6265  062012  000000
6266  062014  001116  004504  001122  DT10:   .WORD   $ERRPC,TSTNM,$BDADR,CS1,CS2,DS1,ER1,0
6267  062022  002262  002260  002304
6268  062030  002264  000000
6269  062034  001116  004504  002262  DT26:   .WORD   $ERRPC,TSTNM,CS1,CS2,DS1,ER1,ER2,ER3,0
6270  062042  002260  002304  002264
6271  062050  002270  002276  000000
6272  062056  001116  004504  004502  DT30:   .WORD   $ERRPC,TSTNM,ERWORD,$GDDAT,$BDDAT,0
6273  062064  001124  001126  000000
6274  062072  001116  004504  004500  DT51:   .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,ILLEGL,0
6275  062100  001124  001126  002364
6276  062106  000000
```

```
6277  062110  001116  004504  004500  DT60:   .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,$BDADR,0
6278  062116  001124  001126  001122
6279  062124  000000
6280  062126  001116  004504  033122  DT61:   .WORD   $ERRPC,TSTNM,PCJSR,$BDADR,0
6281  062134  001122  000000
6282  062140  001116  004504  033122  DT62:   .WORD   $ERRPC,TSTNM,PCJSR,$BDADR,0
6283  062146  001122  000000
6284  062152  001116  004504  002262  DT65:   .WORD   $ERRPC,TSTNM,CS1,AS,DS1,0
6285  062160  002300  002304  000000
6286  062166  001116  004504  002266  DT66:   .WORD   $ERRPC,TSTNM,DST,ER1,ER2,ER3,CS1,CS2,0
6287  062174  002264  002270  002276
6288  062202  002262  002260  000000
6289  062210  001116  004504  002262  DT72:   .WORD   $ERRPC,TSTNM,CS1,CS2,DS1,DST,CA,ER1,WC,0
6290  062216  002260  002304  002266
6291  062224  002274  002264  002254
6292  062232  000000
6293
6294  062234     000     000     000  DF1:    .BYTE   0,0,0,0,0,0,0
6295  062237     000     000     000
6296  062242     000
6297  062243     000     000     000  DF4:    .BYTE   0,0,0,0,0,1,0
6298  062246     000     000     001
6299  062251     000
6300  062252     000     000     000  DF5:    .BYTE   0,0,0,0,0
6301  062255     000     000
6302  062257     000     000     000  DF6:    .BYTE   0,0,0,0
6303  062262     000
6304  062263     000     000     000  DF7:    .BYTE   0,0,0
6305  062266     000     000     000  DF10:   .BYTE   0,0,0,0,0,0,0
6306  062271     000     000     000
6307  062274     000
6308
6309  062275     000     000     000  DF26:   .BYTE   0,0,0,0,0,0,0,0
6310  062300     000     000     000
6311  062303     000     000
6312
6313  062305     000     000     000  DF30:   .BYTE   0,0,0,0,0
6314  062310     000     000
6315
6316  062312     000     000     000  DF51:   .BYTE   0,0,0,0,0,0
6317  062315     000     000     000
6318
6319  062320     000     000     000  DF60:   .BYTE   0,0,0,0,0,0
6320  062323     000     000     000
6321  062326     000     000     000  DF61:   .BYTE   0,0,0,0
6322  062331     000
6323  062332     000     000     000  DF62:   .BYTE   0,0,0,0
6324  062335     000
6325  062336     000     000     000  DF65:   .BYTE   0,0,0,0,0
6326  062341     000     000
6327  062343     000     000     000  DF66:   .BYTE   0,0,0,0,0,0,0,0,0
6328  062346     000     000     000
6329  062351     000     000     000
6330
6331  062354     000     000     000  DF72:   .BYTE   0,0,0,0,0,0,0,0,0
6332  062357     000     000     000
```

```
 6333  062362      000      000      000
 6334
 6335            062366                          .EVEN
 6336
 6337            000001                  .END
```

```
ACL    = 000040            1541#
ACU    = 100000            1501#
ADDMOD   000204             627#  1829
ADTIMO   036222            4721   4741#
AOE    = 001000            1418#  2991   3065
AS       002300            1620#  4560*  4781   6284
ATA    = 100000            1406#  2480   2549   2578   2629   2776   2821   2871   2903   2991   3065   3196   3265
                           3338   3408   3566   3652   4272
ATABLE   004566            1689#  1993
ATTENT   004644            1723#  1948*  1993*  2480   2549   2578   2629   2776   2823   2873   2903   2991   3076
                           3196   3265   3338   3408   3566   3652
AT0    = 000001            1438#
AT1    = 000002            1439#
AT2    = 000004            1440#
AT3    = 000010            1441#
AT4    = 000020            1442#
AT5    = 000040            1443#
AT6    = 000100            1444#
AT7    = 000200            1445#
A16    = 000400            1381#
A17    = 001000            1382#
BA       002256            1609#  4781
BAI    = 000010            1349#  3782
BASECH   035232             627   4714#  4743
BEGIN    004710             626   1744#  4740   4837
BEGIN1   004660             630   1738#
BEGIN2   004674             628   1741#
BIT0   = 000001             620#  2163   2201   3338   3408
BIT00  = 000001             620#
BIT01  = 000002             620#
BIT02  = 000004             620#
BIT03  = 000010             620#
BIT04  = 000020             620#
BIT05  = 000040             620#
BIT06  = 000100             620#
BIT07  = 000200             620#
BIT08  = 000400             620#  4754
BIT09  = 001000             620#  4754   4755
BIT1   = 000002             620#  3329   3399
BIT10  = 002000             620#  3329   3399   4755
BIT11  = 004000             620#  4754
BIT12  = 010000             620#
BIT13  = 020000             620#  4755
BIT14  = 040000             620#  4754
BIT15  = 100000             620#
BIT2   = 000004             620#  3329
BIT3   = 000010             620#
BIT4   = 000020             620#  4350
BIT5   = 000040             620#
BIT6   = 000100             620#
BIT7   = 000200             620#
BIT8   = 000400             620#
BIT9   = 001000             620#
BPTVEC=  000014             620#
CA       002274            1618#  4781   6289
CC       002316            1627#  4781
```

```
CHECK   033124           2096  4253#
CHECKC  033162           4257  4263#
CHECKT  033146           2159  2196  2225  2243  2289  2332  2339  2446  2474  2508  2534  2572  2614
                         2624  2658  2691  2756  2801  2840  2852  2897  2929  2986  3060  3115  3166
                         3181  3220  3259  3303  3310  3325  3381  3395  3431  3464  3561  3612  3684
                         3734  3782  3820  3835  3892  3940  3975  4021  4259#
CHREG   033756           2102  2163  2201  2480  2543  2546  2549  2578  2629  2776  2821  2823  2871
                         2873  2902  2903  2991  3065  3196  3265  3338  3408  3566  3569  3573  3652
                         4496#
CKSWR = 104407           4754  4755  4837#
CLAREA  032736           2142  2183  2232  2236  2238  2307  2389  2440  2469  2501  2529  2565  2567
                         2647  2653  2686  2736  2796  2847  2892  2924  2951  2970  3039  3041  3043
                         3108  3110  3163  3175  3213  3215  3252  3254  3301  3318  3320  3374  3378
                         3389  3426  3460  3461  3831  3881  3888  3971  4013  4017  4162#
CLDISK  033066           1948  2032  2042  2066  2096  2128  2181  2224  2228  2273  2330  2334  2435
                         2469  2497  2563  2612  2618  2643  2686  2722  2794  2837  2845  2890  2920
                         2932  3025  3044  3101  3111  3158  3170  3212  3250  3292  3299  3362  3369
                         3424  3458  3491  3520  3594  3672  3674  3705  3724  3757  3779  3810  3824
                         3875  3940  3965  4008  4232#
CLR   = 000040           1351#  2055  2061  2078  2083  4237
COMPAR  035114           2215  2323  2406  2493  2595  2679  2712  3092  3144  3243  3280  3354  3448
                         3485  3918  4035  4686#
COMREG  034064           2107  2165  2203  2252  2313  2395  2455  2482  2516  2551  2581  2631  2667
                         2700  2783  2826  2876  2906  3007  3080  3122  3200  3230  3269  3341  3411
                         3440  3477  3577  3664  4552#
CPHALT  057172           2096  2159  2196  2225  2243  2289  2332  2339  2446  2474  2508  2534  2572
                         2614  2624  2658  2691  2756  2801  2840  2852  2897  2929  2986  3060  3115
                         3166  3181  3220  3259  3303  3310  3325  3381  3395  3431  3464  3561  3612
                         3684  3734  3782  3820  3835  3892  3940  3975  4021  5998#
CR    = 000015            620#  4754
CRLF  = 000200            620#  4754
CSF   = 000002           1488#
CSU   = 000010           1490#
CS1     002262           1613#  3693*  3694  3743*  3744  3799*  3800  3854  3904  4781  6254  6266  6269
                         6284   6286   6289
CS2     002260           1610#  4781   6266  6269   6286  6289
DB      002252           1607#
DCK   = 100000           1424#
DCL   = 000100           1542#
DCLEAR  002330           1642#
DDISP = 177570            620#   642   1750
DE1   = 000040           1396#
DFF20 = 000002           1392#
DFF5  = 000001           1391#
DF1     062234            655   667   678   6294#
DF10    062266            732   746   760   776   6305#
DF26    062275            895  6309#
DF30    062305            916   935   951   968   1029  1063  1087  1123  1159  1266  1277  6313#
DF4     062243            691  6297#
DF5     062252            701   786   798   812   826   838   846   853   863   872   883   906   927
                          945   961   979   990  1004  1016  1041  1054  1076  1116  1135  1147  1172
                         1213  1225  1291  6300#
DF51    062312           1102  6316#
DF6     062257            711  6302#
DF60    062320           1189  6319#
DF61    062326           1197  6321#
```

```
DF62     062332            1205    6323#
DF65     062336            1237    6325#
DF66     062343            1255    6327#
DF7      062263             719    6304#
DF72     062354            1307    1318    1325    6331#
DH1      057444             648     660     672    6031#
DH10     060243             724     738     752     768    6102#
DH26     060422             887    6122#
DH30     060622             911     933     949     966    1027    1061    1085    1121    1157    1264    1275    6145#
DH4      057624             685    6051#
DH5      057765             696     781     793     807     821     833     844     851     861     870     881     904     925
                            943     959     977     988    1002    1014    1039    1052    1074    1114    1133    1145    1170
                           1211    1223    1286    6069#
DH51     060734            1096    6160#
DH6      060104             707    6084#
DH60     061073            1183    6177#
DH61     061230            1193    6194#
DH62     061322            1201    6204#
DH65     061375            1231    6212#
DH66     061511            1247    6226#
DH7      060203             716    6095#
DH72     061607            1298    1316    1323    6238#
DIGB   = 000004            1393#
DISPLA   001142             642#   1750#   4754#   4755#
DISPRE   000174             621#   1750
DLT    = 100000            1361#
DL64   = 000020            1395#
DMD    = 000001            1428#   3626    3652
DPR    = 000400            1399#   2102    2247    2343    2627    2661    2694    3434    3472    4273
DRY    = 000200            1398#   2102    2291    2332    2617    2627    2844    3314    3823    3951    4011    4273
DST      002266            1615#   4781    6286    6289
DSWR   = 177570             620#    642    1750
DS1      002304            1622#   3852    3902    4781    6266    6269    6284    6289
DT       002306            1623#   2027#   4781
DTE    = 010000            1421#
DTSY   = 001000            1434#
DT1      061714             654     666     677    6254#
DT10     062014             731     745     759     775    6266#
DT26     062034             894    6269#
DT30     062056             915     934     950     967    1028    1062    1086    1122    1158    1265    1276    6272#
DT4      061734             690    6257#
DT5      061756             700     785     797     811     825     837     845     852     862     871     882     905     926
                            944     960     978     989    1003    1015    1040    1053    1075    1115    1134    1146    1171
                           1212    1224    1290    6260#
DT51     062072            1101    6274#
DT6      061772             710    6262#
DT60     062110            1188    6277#
DT61     062126            1196    6280#
DT62     062140            1204    6282#
DT65     062152            1236    6284#
DT66     062166            1254    6286#
DT7      062004             718    6264#
DT72     062210            1306    1317    1324    6289#
DVA    = 004000            1384#   2058    2102    2247    2342    2627    2661    2694    3434    3472    4265
ECH    = 000100            1415#
ECI    = 004000            1514#   2150    2153    2190    2193    2240    2273    2334    2442    2471    2503    2531    2568
```

```
                    2655    2688    2747    2749    2798    2849    2894    2928    2980    2982    3052    3054    3113
                    3165    3177    3217    3256    3302    3321    3380    3391    3428    3462    3683    3733    3782
                    3834    3891    3974    4020
EC1     002312      1625#   4781
EC2     002314      1626#   4781
EMTVEC= 000030       620#   1750*   1811*   1840*
EM1     042566       646    4848#
EM10    043306       722    4907#
EM11    043355       735    4915#
EM12    043462       749    4927#
EM13    043570       763    4939#
EM14    043707       779    4953#
EM15    043761       789    4961#
EM16    044122       801    4979#
EM17    044250       815    4995#
EM2     042615       658    4852#
EM20    044405       829    5012#
EM21    044554       842    5031#
EM22    044627       849    5039#
EM23    044702       856    5047#
EM24    045051       866    5065#
EM25    045214       875    5083#
EM26    045376       886    5103#
EM27    045466       898    5113#
EM3     042704       670    4862#
EM30    045704       909    5139#
EM31    045764       919    5148#
EM32    046201       930    5173#
EM33    046306       938    5185#
EM34    046510       948    5209#
EM35    046534       954    5213#
EM36    046752       964    5238#
EM37    047027       971    5246#
EM4     042765       681    4871#
EM40    047244       982    5271#
EM41    047453       993    5296#
EM42    047734      1007    5328#
EM43    050212      1019    5358#
EM44    050550      1032    5397#
EM45    051042      1044    5430#
EM46    051350      1057    5465#
EM47    051474      1066    5480#
EM5     043076       694    4884#
EM50    051756      1079    5511#
EM51    052175      1090    5538#
EM52    052442      1106    5567#
EM53    052713      1119    5596#
EM54    053001      1126    5606#
EM55    053246      1138    5635#
EM56    053534      1150    5667#
EM57    053744      1162    5691#
EM6     043150       704    4891#
EM60    054236      1175    5724#
EM61    054645      1192    1200    5772#
EM63    054731      1209    5781#
EM64    055004      1217    5789#
```

```
EM65     055223            1230    5814#
EM66     055245            1241    5818#
EM67     055456            1257    5843#
EM7      043236             714    4900#
EM70     055732            1268    5875#
EM71     056140            1280    5900#
EM72     056401            1294    5929#
EM73     056647            1314    5958#
EM74     057017            1321    5977#
ERFLG$   004632            1710#   3614#   4755#
ERR    = 040000            1405#   2480    2549    2578    2776    2821    2871    2903    2991    3065    3196    3265    3338
                           3408    3566    3652    3852    3902
ERRVEC= 000004             620#    1750#   1813#   1821#   1842#   4754#
ERWORD   004502            1671#   4694#   4700#   6272
ER1      002264            1614#   3789#   3790    4781    6266    6269    6286    6289
ER2      002270            1616#   4781    6269    6286
ER3      002276            1619#   4781    6269    6286
EXT1   = 000001            1468#
EXT10  = 000010            1471#
EXT2   = 000002            1469#
EXT20  = 000020            1472#
EXT4   = 000004            1470#
EXT40  = 000040            1473#
FEN    = 000200            1494#
FER    = 000020            1413#   3196    3265
FILL     034022            4522#
FILLRE   032770            2163    2201    2249    2309    2391    2452    2480    2514    2543    2546    2549    2578    2663
                           2696    2991    2998    3000    3065    3071    3073    3120    3189    3194    3196    3225    3265
                           3338    3408    3436    3437    3474    3574    3575    3652    4184#
FINACC   004564            1683#   4325#
FINALA   004562            1682#   4326#
FIRST    004634            1712#   1760    1769#
FLHEAD   032712            2135    2138    2183    2230    2234    2307    2389    2437    2499    2565    2645    2653    2731
                           2734    2922    2944    2948    2964    2968    3031    3035    3108    3161    3299    3372    3376
                           3680    3730    3828    3885    3968    4016    4137#
FMT22  = 010000            1515#   2150    2153    2190    2193    2240    2273    2334    2442    2471    2503    2531    2568
                           2655    2688    2747    2749    2798    2849    2894    2928    2980    2982    3052    3054    3113
                           3165    3177    3217    3256    3302    3321    3380    3391    3428    3462    3600    3615    3683
                           3733    3782    3834    3891    3974    4020    4061
FUTABL   002322            1638#   3499
GNS    = ****** U           621    1766    1767    1768    1803    1828    1838    1840    1863    1885    1922    1924    1926
                           1970    1994    1996    2016    2018    2020    2034    2036    4086    4088    4607    4609    4612
                           4615    4715    4717    4730    4732    4734    4736    4738    4740    4742    4746    4781    4837
GO     = 000001            1378#   2101    2159    2196    2227    2245    2247    2281    2289    2332    2342    2346    2371
                           2448    2476    2510    2536    2574    2616    2626    2627    2660    2661    2693    2694    2758
                           2803    2843    2853    2899    2931    2988    3062    3117    3168    3183    3222    3261    3305
                           3313    3327    3383    3397    3433    3434    3467    3472    3562    3631    3686    3736    3782
                           3822    3837    3894    3950    3977    4010    4028    4350
GRV    = 000010            1394#
GTSWR  = 104406            1771    4837#
HCE    = 000200            1416#   2480    2549    2578
HCI    = 002000            1513#   2150    2153    2190    2193    2240    2273    2334    2442    2471    2503    2531    2568
                           2655    2688    2747    2749    2798    2849    2894    2928    2980    2982    3052    3054    3113
                           3165    3177    3217    3256    3302    3321    3380    3391    3428    3462    3683    3733    3782
                           3834    3891    3974    4020
HCRC   = 000400            1417#
```

```
HT     = 000011          620#  4754
HT.    = 015440         2102#  2161#  2198#  2227#  2247#  2291#  2305#  2332#  2375#  2387#  2450#  2478#  2512#
                        2538#  2576#  2617#  2627#  2661#  2694#  2759#  2804#  2844#  2854#  2900#  2932#  2989#
                        3063#  3118#  3169#  3184#  3223#  3262#  3307#  3314#  3336#  3385#  3406#  3434#  3472#
                        3564#  3688#  3738#  3785#  3823#  3845#  3895#  3951#  3983#  4011#  4029#
IAE    = 002000         1419#  2759   2776   2804   2823   2854   2873   2900   2903
IE     = 000100         1379#  2048   2072   2159   2196   2227   2245   2247   2281   2289   2332   2342   2346
                        2371   2448   2476   2510   2536   2574   2616   2626   2627   2660   2661   2693   2694
                        2758   2803   2843   2853   2899   2931   2988   3062   3117   3168   3183   3222   3261
                        3305   3313   3327   3383   3397   3433   3434   3467   3472   3562   3686   3736   3782
                        3822   3837   3894   3950   3977   4010   4028   4373   4385   4452   4475
ILF    = 000001         1409#  3566
ILLEGL   002364         1656#  3511*  3522   3533   3541   3561   3569   6274
ILR    = 000002         1410#
IOTVEC= 000020           620#  1750*
IR     = 000100         1352#  2543   2991   3196   3225   3265   3539   3550   3573   3658   3659
IXE    = 004000         1498#
KIPAR0= 172340           640#
KIPAR1= 172342           640#
KIPAR2= 172344           640#
KIPAR3= 172346           640#
KIPAR4= 172350           640#
KIPAR5= 172352           640#
KIPAR6= 172354           640#
KIPAR7= 172356           640#
KIPDR0= 172300           640#
KIPDR1= 172302           640#
KIPDR2= 172304           640#
KIPDR3= 172306           640#
KIPDR4= 172310           640#
KIPDR5= 172312           640#
KIPDR6= 172314           640#
KIPDR7= 172316           640#
LA       002320         1628#  4781
LBT    = 002000         1401#  2161   2163   2201   2991   3065   4272
LF     = 000012          620#  4754
LT.    = 000000         2102#  2161#  2198#  2227#  2247#  2291#  2305#  2332#  2375#  2387#  2450#  2478#  2512#
                        2538#  2576#  2617#  2627#  2661#  2694#  2759#  2804#  2844#  2854#  2900#  2932#  2989#
                        3063#  3118#  3169#  3184#  3223#  3262#  3307#  3314#  3336#  3385#  3406#  3434#  3472#
                        3564#  3688#  3738#  3785#  3823#  3845#  3895#  3951#  3983#  4011#  4029#
MCLK   = 000002         1429#
MCPE   = 020000         1385#
MHS    = 001000         1496#
MINX   = 000004         1430#  3627   3628   3639   3640   3643   3644   3647   3648
MMVEC  = 000250          640#
MOL    = 010000         1403#  2033   2034   2102   2247   2343   2627   2661   2694   3434   3472   4273
MPE    = 000400         1354#
MR       002302         1621#  4781
MRD    = 000020         1432#
MSE    = 000020         1491#
MSTCK  = 000010         1431#
MWR    = 000040         1433#
MXF    = 001000         1355#  3538   3548   3557   3573
NED    = 010000         1358#
NEM    = 004000         1357#
NHS    = 002000         1497#
```

```
NOPERA  002322            1639#
NOPUSH  004624            1703#  1738*  1742*  1745*
NOUNIT  004620            1699#  1899*  1932*  1939   1974*  4104*
NUNIT   004622            1701#  1939*  1940*
OCYL  = 100000            1544#
OF      002272            1617#  4781
OFREV = 000200            1512#
OFSET   033052            4227#
OFSETC  002354            1652#  4228
OFSTVL  004510            1678#
OF100 = 000004            1507#
OF200 = 000010            1508#
OF25  = 000001            1505#
OF400 = 000020            1509#
OF50  = 000002            1506#
OF800 = 000040            1510#
OPERSE  034222            4604#  4754
OPI   = 020000            1422#  3652   3790
OR    = 000200            1353#  2543   2546   2991   3196   3225   3265   3538   3548   3658   3659
PAR   = 000010            1412#
PCJSR   033122            4251#  4253*  4254*  4259*  4260*  6280   6282
PCLBUF  002246            1597#  4349*
PCLCSR  002244            1596#  4323*  4350*
PCLCTR  002250            1598#  4324
PGE   = 002000            1356#
PIP   = 020000            1404#
PIRQ  = 177772             620#
PIRQVE= 000240             620#
PKACK   002360            1654#  2096   2101   2102
PLU   = 020000            1500#
PRE   = 000020            1540#
PRITEM  040742            1800*  4103*  4606*  4771#  4781*  4799   4803*
PROG  = 001000            1400#
PR0   = 000000             620#
PR1   = 000040             620#
PR2   = 000100             620#
PR3   = 000140             620#
PR4   = 000200             620#
PR5   = 000240             620#
PR6   = 000300             620#
PR7   = 000340             620#
PS    = 177776             620#  1752*  2047*  2071*  2094*  2292*  2377*  4085*  4605*
PSEL  = 002000            1383#
PSU   = 000001            1537#
PSW   = 177776             620#
PUTREG  035010            3696   3746   3792   3802   3850   3900   4559   4634#
PWRVEC= 000024             620#  1750*  4837*
RA      000200             626#  4738
RDCHR = 104410            4754   4837#
RDLIN = 104411            4755   4837#
RDOCT = 104412            1803   4612   4615   4718   4732   4837#
RDY   = 000200            1380#  2048   2058   2072   2102   2198   2227   2247   2305   2342   2375   2387   2450
                          2478   2512   2538   2576   2627   2661   2694   2932   2989   3063   3118   3169   3184
                          3223   3262   3307   3336   3385   3406   3434   3472   3564   3688   3738   3785   3845
                          3895   3983   4029   4265
READAT  002346            1649#  2190   2193   2196   2471   2476   2894   2899   3217   3222   3256   3261   3321
```

		3327	3428	3433	3434	3462	3467	3472						
READIN	002362	1655#												
RECALI	002326	1641#	2227	2616	3822	3950	4010							
REFOR	002350	1650#	2280	2370	2568	2574	2688	2693	2694	2849	2853	3052	3054	3062
		3113	3117	3782	3891	3894	4020	4028						
REGADR	004500	1670#	1852*	2102*	2247*	2350*	2364*	2627*	2661*	2694*	3434*	3472*	4568*	6260
		6262	6274	6277										
REGSAV	040576	4755#												
REGSA1	040604	1811	4755#											
REINTO	003434	1664#	2142	2190	2193	2201	2215	2238	2273	2309	2323	2334	2391	2406
		2440	2471	2493	2501	2567	2568	2578	2595	2653	2679	2686	2688	2696
		2712	2847	2849	2892	2894	3043	3052	3054	3065	3092	3110	3113	3120
		3144	3215	3217	3225	3243	3254	3256	3280	3320	3321	3338	3354	3428
		3437	3448	3461	3462	3474	3485	3782	3881	3891	3918	4013	4020	4035
RELEAS	002332	1643#												
RESVEC=	000010	620#												
RETCL	002356	1653#												
RHAS	002216	1578#	1847	1863	3468*									
RHBA	002174	1566#	2163	2201	2249	2309	2391	2452	2514	2543	2546	2578	2663	2696
		2770	2815	2865	2991	3065	3120	3189	3194	3225	3338	3408	3437	3474
		3521*	3544*	3553*	3574	3598*	3656	4658*						
RHBAE	002240	1590#	1836											
RHCA	002212	1576#	2163	2201	2998	3000	3071	3073	3616*	4200*	4218*	4655*		
RHCC	002234	1585#	2391	2998	3000	3338	3408	3474	4325	4637				
RHCS1	002200	1571#	2096*	2101*	2159*	2196*	2198	2227*	2245*	2247	2289*	2299*	2305	2309
		2332*	2346*	2387	2391	2448*	2450	2476*	2478	2480	2510*	2512	2536*	2538
		2549	2574*	2576	2578	2616*	2626*	2629	2660*	2661	2693*	2694	2758*	2774
		2803*	2819	2843*	2853*	2869	2899*	2902	2931*	2932	2988*	2989	2991	3062*
		3063	3065	3117*	3118	3168*	3169	3183*	3184	3196	3222*	3223	3261*	3262
		3265	3292	3305*	3307	3313*	3327*	3336	3338	3363	3383*	3385	3397*	3406
		3408	3433*	3434	3467*	3472	3564	3569	3631*	3652	3686*	3688	3693	3736*
		3738	3743	3782*	3785	3799	3822*	3837*	3845	3894*	3895	3950*	3977*	3983
		4010*	4028*	4029	4202*	4219*	4228*	4232	4373	4385	4452	4475	4664*	4715
		4722	4725	4734										
RHCS2	002176	1567#	1864	1905*	1982*	2543	2546	2772	2817	2867	2991	3196	3225	3265
		3573	3657	4233	4661*									
RHCS3	002242	1591#												
RHDB	002170	1564#	1816	4719										
RHDST	002204	1573#	2163	2201	2249	2296*	2309	2378*	2391	2452	2480	2514	2549	2578
		2663	2696	2776	2821	2871	2902	3004	3076	3120	3196	3225	3265	3338
		3408	3437	3474	3611*	3652	4201*	4656*						
RHDS1	002222	1580#	2102	2161	2163	2201	2227	2291	2332	2375	2480	2549	2578	2617
		2627	2629	2776	2821	2844	2871	2903	2991	3065	3196	3265	3314	3338
		3408	3566	3652	3823	3951	4011	4234						
RHDT	002224	1581#	1906	1908	1911	1913	1916	1918	1924	1983	1985	1996	2000	2002
		2005	2007	2010	2012	2025	2027							
RHEC1	002230	1583#												
RHEC2	002232	1584#												
RHER1	002202	1572#	1867	2480	2549	2578	2759	2776	2804	2823	2854	2873	2900	2903
		2991	3065	3196	3265	3338	3408	3566	3652	3789	4235			
RHER2	002206	1574#												
RHER3	002214	1577#												
RHLA	002236	1586#	4326											
RHMR	002220	1579#	3625	3652										
RHOF	002210	1575#	3615*	4227*	4662*									
RHSN	002226	1582#	1996	2024	2026									

```
RHWC    002172  1565# 2099  2158  2163  2196  2201  2242  2249  2276  2309  2337  2391  2445
                2452  2473  2507  2514  2533  2543  2546  2571  2578  2622  2657  2663  2690
                2696  2755  2768  2800  2813  2851  2863  2896  2985  2991  3059  3065  3115
                3120  3180  3189  3194  3219  3225  3258  3324  3338  3394  3408  3430  3436
                3463  3474  3520* 3543* 3552* 3560  3575  3596* 3619  3655  4187  4499  4568
                4635  4637  4657*
RH70    004640  1717# 1774* 1837* 2540  3186  3492
RH70CK  006032  1772#
RMR     = 000004  1411# 3338  3408
RPTCTR  033566  4426# 4470
RPTRP1  011266  2044  2057#
RPTRP2  011374  2068  2080#
RPVEC   002166  1330# 1754  2043  2067  2100* 2159* 2196* 2226* 2244* 2288* 2298* 2332* 2339*
                2381* 2447* 2475* 2509* 2535* 2573* 2615* 2625* 2659* 2692* 2757* 2802* 2841*
                2853* 2898* 2930* 2987* 3061* 3116* 3167* 3182* 3221* 3260* 3304* 3312* 3326*
                3382* 3396* 3432* 3465* 3685* 3735* 3782* 3821* 3836* 3893* 3949* 3976* 4009*
                4022* 4730  4733* 4736  4747*
RPVECT  036306  1755  4745# 4747
RP06    004636  1715# 1981* 1988* 2131  2147  2187  2727  2742  2940  2960  2976  2994  3027
                3049  3068  3955  4048  4066
RP4VEC  004506  1673# 1799* 2100  2159  2196  2226  2244  2298  2332  2381  2447  2475  2509
                2535  2573  2615  2625  2659  2692  2757  2802  2841  2853  2898  2930  2987
                3061  3116  3167  3182  3221  3260  3304  3312  3326  3382  3396  3432  3465
                3685  3735  3782  3821  3836  3893  3949  3976  4009  4022
RTN     006516  1838  1840#
RUN     035050  2150  2153  2190  2193  2240  2273  2334  2442  2471  2503  2531  2568  2655
                2688  2747  2749  2798  2849  2894  2928  2980  2982  3052  3054  3113  3165
                3177  3217  3256  3302  3321  3380  3391  3428  3462  3683  3733  3782  3834
                3891  3974  4020  4655#
RUNCTR  004642  1720# 3632* 3635*
RUNWAT  027414  3632#
SAVER   033234  2099  2158  2196  2242  2276  2337  2445  2473  2507  2533  2571  2622  2657
                2690  2755  2800  2851  2896  2985  3059  3115  3180  3219  3258  3324  3394
                3430  3463  3560  3619  4294#
SAVERE  004512  1681# 2099  2107  2158  2165  2196  2203  2242  2252  2276  2313  2337  2395
                2445  2455  2473  2480* 2482  2507  2516  2533  2549* 2551  2571  2578* 2581
                2622  2629* 2631  2657  2667  2690  2700  2755  2768* 2770* 2772* 2774* 2776*
                2783  2800  2813* 2815* 2817* 2819* 2821* 2823* 2826  2851  2863* 2865* 2867*
                2869* 2871* 2873* 2876  2896  2902* 2903* 2906  2985  2991* 3004* 3007  3059
                3076* 3080  3115  3122  3180  3196* 3200  3219  3225* 3230  3258  3265* 3269
                3324  3338* 3341  3394  3408* 3411  3430  3440  3463  3477  3560  3566* 3569*
                3577  3619  3652* 3655* 3656* 3659* 3660* 3664  4188* 4502* 4504* 4560
SC      = 100000  1387# 2480  2549  2578  2629  2902  2991  3065  3196  3265  3338  3408  3525
                3535  3545  3554  3569  3652  3694  3744  3800
SC1     = 000100  1474#
SC10    = 001000  1477#
SC2     = 000200  1475#
SC20    = 002000  1478#
SC4     = 000400  1476#
SEECOM  002352  1651# 2332  2843  3313  4219
SEEKCY  033036  2332  2842  3311  4218#
SELECT  004626  1705# 1739* 1741* 1744* 1802  1943  1955  4092
SELTST  007700  1881  1943#
SERCH   002334  1644# 2273  2289  2334  2340  2346  2626  2627  4202
SKI     = 040000  1543#
SN      002310  1624# 2026* 4781
```

```
SND1     005774            1763    1769#
SRCH     033016            2340    2620    4200#
SR0    = 177572             640#
SR1    = 177574             640#
SR2    = 177576             640#
SR3    = 172516             640#
SST1     031126            3877#
SST2     031134            3880#   3881    3938
SST3     031154            3871#   3884#   3932#
SST4     031174            3872#   3887#   3933#
SST5     031206            3873#   3890#   3934#
STACK  = 001000             620#   1750    1810    1845    1948    2041    2065    2096    2128    2224    2330    2435    2497
                           2612    2643    2722    2794    2837    2890    2920    3158    3250    3292    3362    3458    3491
                           3594    3674    3724    3810    3940
START    004720            1740    1743    1748#
STKLMT=  177774             620#
ST22     024150            3298#   3360
ST22A    022546            3121#
ST23     024726            3354    3358#
ST24     025000            3368#   3455
ST28     026014            3448    3453#
SWR      001140             642#   1750#   1771    1772    1825    1861    1879    4572    4703    4754#   4755    4773    4837#
SWREG    000176             621#   1750    1771    4754
SW0    = 000001             620#
SW00   = 000001             620#
SW01   = 000002             620#
SW02   = 000004             620#
SW03   = 000010             620#
SW04   = 000020             620#
SW05   = 000040             620#
SW06   = 000100             620#   4775
SW07   = 000200             620#   4574    4705
SW08   = 000400             620#
SW09   = 001000             620#
SW1    = 000002             620#
SW10   = 002000             620#
SW11   = 004000             620#
SW12   = 010000             620#   1772
SW13   = 020000             620#   1825    1861    1879
SW14   = 040000             620#
SW15   = 100000             620#
SW2    = 000004             620#
SW3    = 000010             620#
SW4    = 000020             620#
SW5    = 000040             620#
SW6    = 000100             620#
SW7    = 000200             620#
SW8    = 000400             620#
SW9    = 001000             620#
TBITVE=  000014             620#
TDF    = 000040            1492#
TESTAD   034220            4603#
TIMCNT   033564            2050    2074    4362    4367    4425#   4441    4446
TIME1    033266            4323#
TIME2    033300            1799    4325#
TKVEC  = 000060             620#   4754#
```

```
TMP0     004650          1726# 2282# 2299  2370# 2371# 2381  3292# 3328  3331# 3343  3363# 3398  3401#
                         3413
TMP1     004652          1727# 3676# 3718# 3726# 3770#
TMP4     004654          1728#
TMP5     004656          1729# 2283# 2296  2372# 2378  3293# 3358# 3364# 3453#
TN     = 000037          1948# 2096# 2128# 2224# 2330# 2435# 2497# 2612# 2643# 2722# 2794# 2837# 2890#
                         2920# 3158# 3250# 3292# 3362# 3458# 3491# 3594# 3674# 3724# 3810# 3940#
TOTALA   004646          1724# 1872# 1874#
TPVEC  = 000064           620#
TRAPVE= 000034            620# 1750# 1753# 2351# 2365# 2376#
TRE    = 040000          1386# 2480  2549  2578  2902  2991  3065  3196  3265  3535  3545  3554  3652
                         3854  3904
TRK1   = 004000          1479#
TRK10  = 040000          1482#
TRK2   = 010000          1480#
TRK20  = 100000          1483#
TRK4   = 020000          1481#
TRP      006460          1835  1839#
TRTVEC= 000014            620#
TSTNM    004504          1672# 1810# 1858# 1948# 2029# 2041# 2065# 2096# 2128# 2224# 2330# 2435# 2497#
                         2612# 2643# 2722# 2794# 2837# 2890# 2920# 3158# 3250# 3292# 3362# 3458# 3491#
                         3594# 3674# 3724# 3810# 3940# 4607  6254  6257  6260  6262  6264  6266  6269
                         6272  6274  6277  6280  6282  6284  6286  6289
TST1     006176          1803  1807  1810# 4122
TST10    011404          2079  2089#
TST11    011650          2128#
TST12    012524          2224#
TST13    013504          2330#
TST14    014252          2435#
TST15    014760          2497#
TST16    016110          2612#
TST17    016442          2643#
TST2     006532          1845#
TST20    017330          2722#
TST21    017710          2794#
TST22    020166          2837#
TST23    020524          2890#
TST24    020772          2920#
TST25    022612          3158#
TST26    023566          3250#
TST27    024114          3292#
TST3     006600          1850  1857#
TST30    024744          3360  3362#
TST31    026032          3455  3458#
TST32    026376          3491#
TST33    027224          3494  3594#
TST34    027652          3674#
TST35    030134          3724#
TST36    030572          3750  3810#
TST37    031404          3938  3940#
TST4     007714          1944  1947#  4101  4112
TST40    032336          4085#
TST5     010642          2029#
TST6     011174          2034  2040#
TST7     011304          2056  2059   2064#
TUF    = 000100          1493#
```

```
TYPDS = 104405          1923  1931  1995  4087  4089  4099  4122  4825  4837#
TYPE  = 104401          1766  1767  1768  1803  1828  1838  1840  1863  1885  1922  1924  1926  1929
                        1970  1994  1996  2016  2018  2020  2034  2036  2096  2159  2196  2225  2243
                        2289  2332  2339  2446  2474  2508  2534  2572  2614  2624  2658  2691  2756
                        2801  2840  2852  2897  2929  2986  3060  3115  3166  3181  3220  3259  3303
                        3310  3325  3381  3395  3431  3464  3561  3612  3684  3734  3782  3820  3835
                        3892  3940  3975  4021  4086  4088  4097  4100  4122  4607  4609  4611  4612
                        4615  4715  4717  4730  4732  4734  4736  4738  4740  4742  4746  4754  4755
                        4781  4784  4806  4808  4811  4813  4828  4837#
TYPERR  041704          4778  4783#
TYPOC = 104402          1831  1925  1996  4608  4610  4716  4731  4735  4737  4739  4746  4754  4781
                        4792  4821  4837#
TYPON = 104404          4837#
TYPOS = 104403          4837#
UNIB  = 000020          1350#
UNIT    004616          1698#  1805*  1807*  1938*  1944*  1950  1973*  1976  1994  4086  4106  4111*  4238
                        4659
UNITS   004576          1697#  1893  1897  1938  1962  4107
UNITSL  004630          1706#  1806*  1807  1944
UNLOAD  002324          1640#
UNS   = 040000          1423#
UPE   = 020000          1359#
US1   = 000001          1346#
US2   = 000002          1347#
US4   = 000004          1348#
UWR   = 000010          1539#
VUF   = 000002          1538#
VU30  = 010000          1499#
VV    = 000100          1397#  2102  2247  2343  2627  2661  2694  3434  3472  4256  4273
WAITBT  033322          4346#  4355*  4363  4368  4435*  4442  4447  6254  6257
WAITPC  033316          4344#  4352*  4353*  4432*  4433*  6254  6257
WAITRE  033320          4345#  4354*  4363  4368  4372  4384  4434*  4442  4447  4451  4477  6254  6257
WAITTM  033324          4324*  4347#  4348*  4392  4402  6257
WAIT.P  033326          4348#
WAIT.T  033570          4429#  4837
WAT   = 104413          2102  2161  2198  2227  2247  2291  2305  2332  2375  2387  2450  2478  2512
                        2538  2576  2617  2627  2661  2694  2759  2804  2844  2854  2900  2932  2989
                        3063  3118  3169  3184  3223  3262  3307  3314  3336  3385  3406  3434  3472
                        3564  3688  3738  3785  3823  3845  3895  3951  3983  4011  4029  4837#
WC      002254          1608#  2107  2165  2203  2252  2313  2395  2455  2482  2516  2551  2581  2631
                        2667  2700  2783  2826  2876  2906  3007  3080  3122  3200  3230  3269  3341
                        3411  3440  3477  3577  3664  4636  4781  6289
WCE   = 040000          1360#
WCF   = 000040          1414#
WCU   = 000001          1487#
WLE   = 004000          1420#
WRCHDT  002340          1646#
WRCHEK  002336          1645#
WRFROM  002370          1663#  2135  2138  2150  2153  2163  2183  2215  2230  2232  2234  2236  2240
                        2249  2307  2323  2389  2406  2437  2442  2452  2469  2493  2499  2503  2514
                        2529  2531  2543  2546  2565  2595  2645  2647  2655  2663  2679  2712  2731
                        2734  2736  2747  2749  2796  2798  2922  2924  2928  2944  2948  2951  2964
                        2968  2970  2980  2982  2991  3031  3035  3039  3041  3092  3108  3144  3161
                        3163  3165  3175  3177  3189  3194  3213  3243  3252  3280  3299  3301  3302
                        3318  3354  3372  3374  3376  3378  3380  3389  3391  3408  3426  3448  3460
                        3485  3544  3546  3597  3680  3683  3730  3733  3828  3831  3834  3885  3888
```

		3918	3968	3971	3974	4016	4017	4035					
WRIDAT	002342	1647*	2531	2536	2798	2803	3177	3183	3391	3397			
WRIFOR	002344	1648*	2150	2153	2159	2240	2245	2247	2442	2448	2503	2510	2655	2660
		2661	2747	2749	2758	2928	2931	2980	2982	2988	3165	3168	3302	3305
		3380	3383	3612	3683	3686	3733	3736	3834	3837	3974	3977		
WRL =	004000	1402*												
WRU =	000400	1495*												
WSU =	000004	1489*												
XE2	007342	1877	1891*											
$AUTOB	001134	642*	1771*	4754										
$BDADR	001122	642*	3344*	3414*	4268*	4275*	6266	6277	6280	6282				
$BDDAT	001126	642*	1850*	2102*	2247*	2348*	2362*	2627*	2661*	2694*	3434*	3472*	4372*	4384*
		4451*	4477*	4567*	4699*	6254	6257	6260	6262	6272	6274	6277		
$BELL	001216	642*	4754	4755										
$CHARC	037310	4754**												
$CKSWR	037622	4754*	4837											
$CMTAG	001100	642*	1750											
$CM1 =	000006	642*												
$CM2 =	000014	642*												
$CM3 =	000006	642*												
$CM4 =	000006	642*												
$CNTLC	040373	4754*												
$CNTLG	040405	4754*												
$CNTLU	040400	4754*												
$CRLF	001223	642*	1929	4611	4754	4755	4784	4808	4813					
$DBLK	037064	4754*												
$DOAGN	032666	4122*												
$DTBL	037054	4754*												
$ENDAD	032656	622	4122*											
$ENDCT	032624	4122*												
$ENDMG	032675	4097	4122*											
$ENULL	032672	4100	4122*											
$EOP	032570	1833	1886	4105	4122*									
$EOPCT	032616	4122*												
$ERFLG	001103	642*	4754*	4755*										
$ERMAX	001115	642*	1750*	4754*										
$ERROR	040574	1750	1840	4755*										
$ERRPC	001116	642*	4755*	4790	6254	6257	6260	6262	6264	6266	6269	6272	6274	6277
		6280	6282	6284	6286	6289								
$ERRTB	001226	642*	4798											
$ERRTY	040744	4755	4773*											
$ERTTL	001112	642*	4088	4090*	4755*									
$ESCAP	001214	642*	1750*	4754*	4755									
$FILLC	001156	642*	4754											
$FILLS	001155	642*	4754											
$GDADR	001120	642*												
$GDDAT	001124	642*	1851*	2102*	2247*	2349*	2363*	2627*	2661*	2694*	3434*	3472*	4566*	4698*
		6260	6272	6274	6277									
$GET42	032646	4122*												
$GTSWR	037712	4754*	4837											
$HD =	000000	595												
$HIOCT	040572	4755**												
$ICNT	001104	642*	4754*											
$ILLUP	042550	4837*												
$INTAG	001135	642*	4754*											
$ITEMB	001114	642*	4755*	4787										

```
$LF     001224            642#   4754   4755
$LPADR  001106            642#   1750*  1947*  4614*  4617   4754*
$LPERR  001110            642#   1750*  3516*  3881*  3998*  4609   4616*  4754*  4755
$MAIL = ****** U         1750    1771   4754   4755
$MNEW   040423           4754#
$MSWR   040412           4754#
$MXCNT  036646           4754#
$NULL   001154            642#   4754
$NWTST= 000001           1810#   1845#  1857#  1947#  2029#  2040#  2064#  2089#  2128#  2224#  2330#  2435#  2497#
                         2612#   2643#  2722#  2794#  2837#  2890#  2920#  3158#  3250#  3292#  3362#  3458#  3491#
                         3594#   3674#  3724#  3810#  3940#  4085#
$OCNT   042314           4837#*
$OMODE  042316           4837#*
$OVER   036632           4754#
$PASS   001100            642#   4096*  4098   4122*  4754
$POWER  042556           4837#
$PWRAD  042544           4837#
$PWRDN  042404           1750    4837#
$PWRMG  042540           4837#
$PWRUP  042456           4837#
$QUES   001222            642#   4754   4755
$RDCHR  040164           4754#   4837
$RDDEC= ****** U         4837
$RDLIN  040254           4754#   4837
$RDOCT  040434           4755#   4837
$RDSZ = 000011           4754#
$REGAD  001160            642#
$REG0   001162            642#
$REG1   001164            642#
$REG2   001166            642#
$REG3   001170            642#
$REG4   001172            642#
$REG5   001174            642#
$RTNAD  032670           4122#
$R2A  = ****** U         4837
$SAVRE= ****** U         4837
$SAVR6  042554           4837#*
$SCOPE  036376           1750    4754#
$SETUP= 000117           1749#   1750   1771   4122   4754   4755
$SS1  = 000000           1751#
$STUP = 177777           1749#
$SVLAD  036604           4754#
$SVPC = 000200            622#
$SWR  = 167770            595#    619    642   1750   1810   1845   1857   1947   2029   2040   2064   2089   2128
                         2224    2330   2435   2497   2612   2643   2722   2794   2837   2890   2920   3158   3250
                         3292    3362   3458   3491   3594   3674   3724   3810   3940   4085   4122   4754   4755
                         4837
$SWRMK= 000000            619    4754
$TIMES  001212            642#   1750*  1810*  1845*  1857*  1947*  4085*  4122*  4754*
$TKB    001146            642#   4754
$TKCNT  037314           4754#*
$TKINT  037334           1759    1859   2093   4717   4754#
$TKQEN= 037333           4754#
$TKQIN  037316           4754#*
$TKQOU  037320           4754#*
$TKQSR  037322           4754#
```

```
$TKS    001144            642#  4754*
$TKSRV  037404           4754#
$TMP0   001176            642#  1901*  1902*
$TMP1   001200            642#  1823*  2050*  2051*  2074*  2075*  3498*  3501   3503*  3507   3511   3512*  3816*
                         3867*  3877*  3936*  3959*  3962*  3990*  3999*  4043   4045*  4052   4054   4058*  4059
                         4060   4063   4070   4073   4077*  6264
$TMP2   001202            642#
$TMP3   001204            642#
$TMP4   001206            642#
$TMP5   001210            642#
$TN   = 000041            595#  1803   1807   1810#  1845#  1850   1857#  1858   1944   1947#  1948   2029#  2034
                         2040#  2041   2056   2059   2064#  2065   2079   2089#  2096   2128#  2224#  2330#  2435#
                         2497#  2612#  2643#  2722#  2794#  2837#  2890#  2920#  3158#  3250#  3292#  3360   3362#
                         3455   3458#  3491#  3494   3594#  3674#  3724#  3750   3810#  3938   3940#  4085#
$TPB    001152            642#  4754*
$TPFLG  001157            642#  4754
$TPS    001150            642#  4754
$TRAP   042320           1750   4837#
$TRAP2  042342           4837#
$TRP  = 000014           4837#
$TRPAD  042354           4837#
$TSTNM  001102            642#  4091*  4122*  4754*  4755
$TTYIN  040362           4754#
$TYPBN= ****** U         4837
$TYPDS  036650           4754#  4837
$TYPE   037074           4754#  4837
$TYPEC  037244           4754#
$TYPEX  037312           4754#
$TYPOC  042116           4837#
$TYPON  042132           4837#
$TYPOS  042072           4837#
$XTSTR  036410           4754#
$$GET4= 000000           4122#
$OFILL  042315           4837**
$40CAT= ****** U         4754   4755
.     = 062366            621#   622#   625#   629#   641#   642#  1662#  1663#  1664#  1681#  1697#  1750   1766#
                         1767#  1838#  1840#  1863#  1885#  1922#  1924#  1926#  1970#  1994#  2016#  2018#  2020#
                         2034#  2036#  4086#  4088#  4122   4612#  4615#  4730#  4734#  4736#  4738#  4746#  4754#
                         4755   4781#  4835#  4837   6335#
```

ALLREG 619# 4780 4781
CHANGR 619# 2102 2163 2201 2480 2543 2546 2548 2549 2578 2629 2776 2821 2823 2871
2873 2902 2903 2991 3065 3076 3195 3196 3265 3338 3408 3566 3568 3572 3652
CHECKD 619# 2159 2196 2225 2243 2289 2332 2339 2446 2474 2508 2534 2572 2614 2624
2658 2691 2756 2801 2840 2852 2897 2929 2986 3060 3115 3166 3181 3220 3259
3303 3310 3325 3381 3395 3431 3464 3561 3612 3684 3734 3782 3820 3835 3892
3940 3975 4021
CHECKV 619# 2096
CHKCNT 619#
CKCNTV 619#
CLEARA 619# 2142 2183 2232 2236 2238 2307 2389 2440 2469 2501 2529 2565 2567 2647
2653 2686 2736 2796 2847 2892 2924 2951 2970 3039 3041 3043 3108 3110 3163
3175 3213 3215 3252 3254 3301 3318 3320 3374 3378 3389 3426 3460 3461 3830
3880 3887 3970 4013 4016
CMPBLK 619# 2214 2322 2405 2492 2594 2678 2711 3091 3143 3242 3279 3353 3447 3484
3917 4035
CMREGI 619# 2107 2165 2203 2252 2313 2395 2455 2482 2516 2551 2581 2631 2667 2700
2783 2826 2876 2906 3007 3080 3122 3200 3230 3269 3341 3411 3440 3477 3577
3664
COMMEN 598 620#
DATACO 619# 2150 2152 2190 2192 2240 2273 2334 2442 2471 2503 2531 2568 2655 2688
2747 2748 2798 2849 2894 2928 2980 2981 3052 3053 3113 3165 3177 3217 3256
3302 3321 3380 3391 3428 3462 3682 3732 3782 3833 3890 3973 4019
DISREG 619# 3196 3225 3265
DUM 619# 2096
ENDCOM 603 620#
ERROR 620# 1824 1853 2054 2059 2081 2102 2108 2166 2204 2216 2247 2253 2313 2324
2352 2366 2396 2407 2455 2482 2493 2516 2551 2582 2595 2627 2632 2661 2668
2680 2694 2701 2713 2784 2827 2877 2907 3008 3081 3093 3123 3145 3201 3231
3244 3270 3281 3345 3355 3415 3434 3441 3449 3472 3478 3486 3578 3665 3697
3747 3793 3803 3857 3907 3920 4036 4269 4276 4375 4377 4387 4396 4405 4455
4459 4478
ESCAPE 620#
FIHEAD 619# 2135 2137 2183 2230 2234 2307 2389 2437 2499 2565 2645 2653 2731 2733
2922 2944 2947 2964 2967 3031 3034 3108 3161 3299 3372 3376 3680 3730 3827
3884 3967 4015
FILLBL 619#
FLSVRE 619# 2163 2201 2249 2309 2391 2452 2480 2514 2543 2545 2546 2549 2578 2663
2696 2991 2998 2999 3000 3065 3071 3072 3120 3189 3193 3194 3196 3225 3265
3338 3408 3436 3437 3474 3573 3574 3652
GETPRI 620#
GETSWR 595# 620# 1771
GOO 619# 2101 2159 2196 2227 2245 2289 2332 2346 2448 2476 2510 2536 2574 2616
2626 2660 2693 2758 2803 2843 2853 2899 2931 2988 3062 3117 3168 3183 3222
3261 3305 3313 3327 3383 3397 3433 3467 3686 3736 3782 3822 3837 3894 3950
3977 4010 4028
LOAD 619#
MSG 1810# 1844# 1845 1856# 1857 1946# 1947 2029# 2039# 2040 2063# 2064 2088# 2089 2127#
2128 2223# 2224 2329# 2330 2434# 2435 2497# 2611# 2612 2643# 2721# 2722 2793# 2794
2836# 2837 2889# 2890 2919# 2920 3157# 3158 3250# 3291# 3292 3362# 3457# 3458 3490#
3491 3593# 3594 3673# 3674 3723# 3724 3809# 3810 3939# 3940 4085#
MULT 620#
NEWTST 620# 1810 1845 1857 1947 2029 2040 2064 2089 2128 2224 2330 2435 2497 2612
2643 2722 2794 2837 2890 2920 3158 3250 3292 3362 3458 3491 3594 3674 3724
3810 3940 4085
OFFST 619#

```
POP      620#  3346  3416  4146  4169  4189  4301  4407  4483  4507  4533  4579  4641  4709  4754
        4755  4837
PUSH     620#  3342  3412  4137  4162  4184  4294  4351  4429  4496  4522  4552  4634  4686  4754
        4755  4837
REPORT   620#
RFORGC   619#
RHCLEA   619#  1948  2042  2066  2096  2128  2180  2224  2228  2272  2330  2334  2435  2468  2497
        2562  2612  2618  2643  2685  2722  2794  2837  2845  2890  2920  2932  3024  3044  3100
        3111  3158  3170  3211  3250  3292  3298  3362  3368  3423  3458  3491  3519  3594  3674
        3704  3724  3756  3779  3810  3824  3875  3940  3965  4007
SAVE     619#  4755
SAVTST   619#  1810  1858  2029  2041  2065
SCH      619#  2340  2620
SCOPE    620#  1810  1845  1857  1947  2029  2040  2064  2089  2128  2224  2330  2435  2497  2612
        2643  2722  2794  2837  2890  2920  3158  3250  3292  3362  3458  3491  3594  3674  3724
        3810  3940  4085  4122
SEEKCO   619#  2332  2842  3311
SETPRI   620#  4754
SETTRA  4837#
SETUP    620#  1750
SKIP     619#   620#  1803  1807  1850  1944  2034  2056  2059  2079  3938
SLASH    620#
SPACE    620#
SREGIS   619#  2099  2158  2196  2242  2276  2337  2445  2473  2507  2533  2571  2622  2657  2690
        2755  2800  2851  2896  2985  3059  3115  3180  3219  3258  3324  3394  3430  3463  3559
        3619
STARS    613   618   620#  622   642  1327  1328  1372  1373  1776  1797  1810  1845  1857  1911
        1920  1947  1981  1991  2000  2020  2029  2040  2064  2089  2128  2131  2134  2147  2149
        2187  2189  2224  2330  2416  2429  2435  2497  2612  2643  2722  2727  2730  2742  2745
        2782  2794  2837  2890  2920  2928  2938  2940  2943  2960  2963  2976  2979  2994  2997
        3024  3027  3030  3049  3052  3068  3071  3106  3158  3250  3292  3362  3458  3491  3594
        3674  3724  3810  3940  3955  3958  4002  4048  4051  4066  4069  4085  4122  4745  4749
        4754  4755  4837
STARTT   619#  1948  2096  2128  2224  2330  2435  2497  2612  2643  2722  2794  2837  2890  2920
        3158  3250  3292  3362  3458  3491  3594  3674  3724  3810  3940
SWRSU    620#  1750#
TJUMP    619#  3360  3455  3494  3750
TRMTRP  4837#
TSCLR2   619#
TSCLR5   619#
TTSTNO   619#  1948  2096  2128  2224  2330  2435  2497  2612  2643  2722  2794  2837  2890  2920
        3158  3250  3292  3362  3458  3491  3594  3674  3724  3810  3940
TYPBIN   620#
TYPDEC   620#  4122
TYPNAM   620#
TYPNUM   620#
TYPOCS   620#
TYPOCT   620#  1996  4754
TYPTXT   620#  1765  1766  1767  1768  1803  1828  1838  1840  1863  1884  1885  1922  1924  1926
        1969  1970  1994  1996  2015  2017  2019  2034  2036  4086  4088  4607  4609  4612  4615
        4714  4717  4729  4732  4734  4736  4738  4740  4741  4745  4781
VECSET   619#  2100  2159  2196  2226  2244  2298  2332  2381  2447  2475  2509  2535  2573  2615
        2625  2659  2692  2757  2802  2841  2853  2898  2930  2987  3061  3116  3167  3182  3221
        3260  3304  3312  3326  3382  3396  3432  3465  3685  3735  3782  3821  3836  3893  3949
        3976  4009  4022
WT       619#  2102  2161  2198  2227  2247  2291  2305  2332  2375  2387  2450  2478  2512  2538
```

```
                2576    2617    2627    2661    2694    2759    2804    2844    2854    2900    2932    2989    3063    3118    3169
                3184    3223    3262    3307    3314    3336    3385    3406    3434    3472    3564    3688    3738    3785    3823
                3845    3895    3951    3983    4011    4029
WTT              619#   2102    2161    2198    2227    2247    2291    2305    2332    2375    2387    2450    2478    2512    2538
                2576    2617    2627    2661    2694    2759    2804    2844    2854    2900    2932    2989    3063    3118    3169
                3184    3223    3262    3307    3314    3336    3385    3406    3434    3472    3564    3688    3738    3785    3823
                3845    3895    3951    3983    4011    4029
$$CMRE           642#
$$CMTM           642#
$$ESCA           620#
$$NEWT           620#   1810    1845    1857    1947    2029    2040    2064    2089    2128    2224    2330    2435    2497    2612
                2643    2722    2794    2837    2890    2920    3158    3250    3292    3362    3458    3491    3594    3674    3724
                3810    3940    4085
$$SET           4837#
$$SKIP           620#
.EQUAT           595#     620
.HEADE           595#
.KT11            595#     640
.SETUP           595#    1748
.SWRHI           595#     619
.SWRLO           595#     619#
.$ACTI           595#     622
.$CATC           595#     621
.$CMTA           595#     642
.$EOP            595#    4122
.$ERRO           595#    4755
.$ERRT           595#
.$POWE           595#    4837
.$RDOC           595#    4755
.$READ           595#    4754
.$SCOP           595#    4754
.$TRAP           595#    4837
.$TYPD           595#    4754
.$TYPE           595#    4754
.$TYPO           595#    4837

. ABS.   062366     000

 ERRORS DETECTED:  0

 RM03:CZRJJB,CZRJJB.SEQ/CRF/SOL/NL:MC:ME:CND=RM03:CZRJJB.P11
 RUN-TIME:  41  32  2 SECONDS
 RUN-TIME RATIO: 588/76=7.7
 CORE USED:  28K  (55 PAGES)
```