RP04/5/6

FUNCTIONAL CONTROL 1

CZRJIB0

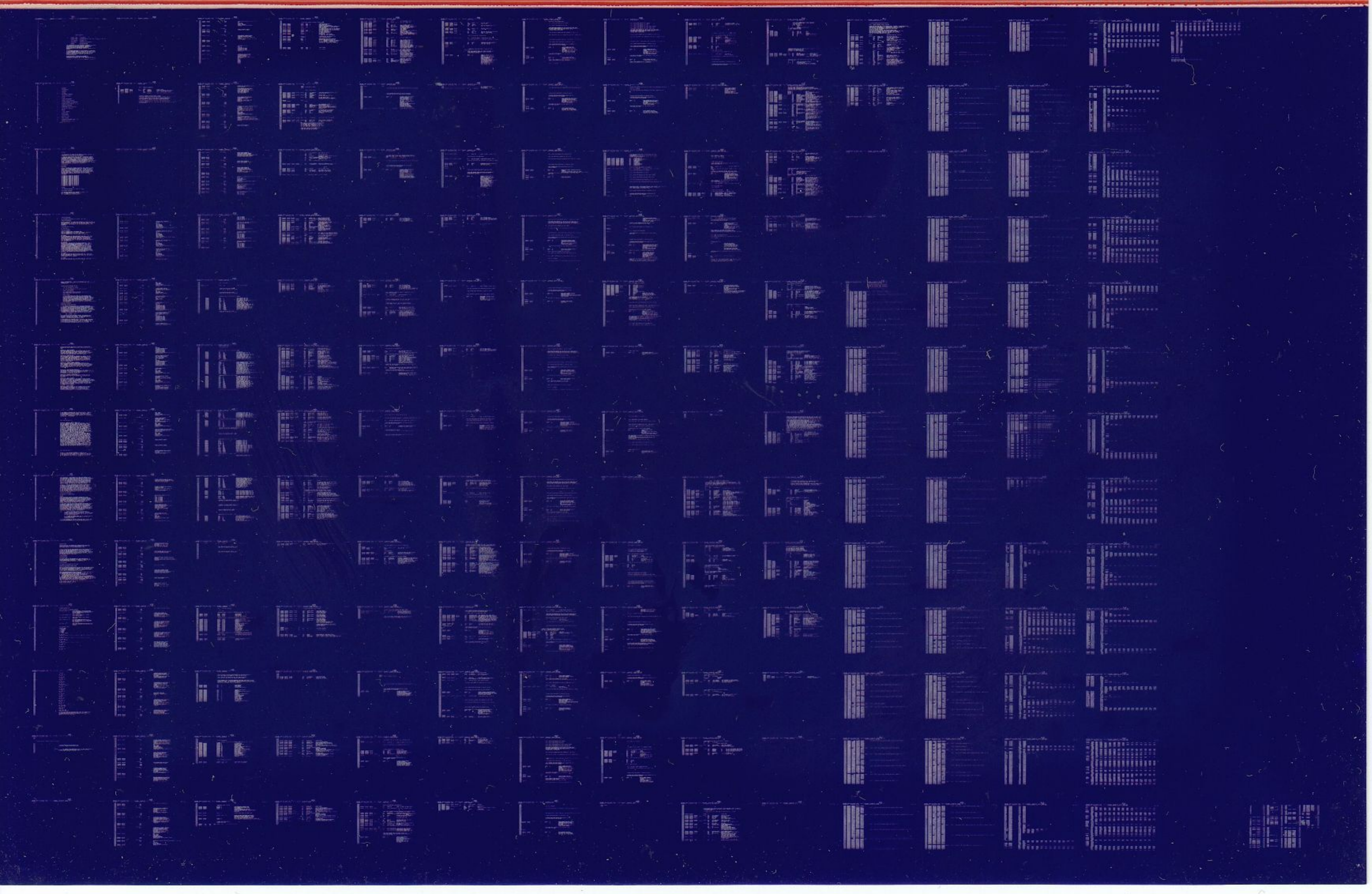
AH-9220B-MC

FICHE 1 OF 1

JAN 1978

digital

MADE IN USA



.REM @

IDENTIFICATION

PRODUCT CODE: AC-92188-MC
PRODUCT NAME: CZRJIB0 RP04/5/6 FUNCTIONHL CONTROLLER TEST PART I
DATE CREATED: DECEMBER 1977
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: PETE BLACKSTONE

CONTENTS

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 IS USED TO TEST RP04/5/6 DEVICE CONTROL LOGIC
                        CONNECTED TO 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 PROGRAMS
                        DZRJGA AND DZRJHA HAVE 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.
                        AND IT ASSUMES THAT MAINDEC-11-DZRJH-(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 DESIRES.  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 204---TO SELECT NON-DEFAULT ADDRESSES
                                START AT ADDRESS 210---FOR UNIT SELECTION
                                START AT ADDRESS 220---FOR NO MANUAL INTERVENTION

                                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 START 200 WITH THE FOLLOWING EXCEPTION:  THE
                                PROGRAM WILL INTERROGATE THE OPERATOR FOR A NON-STANDARD
                                C.S.R AND VECTOR ADDRESS BEFORE STARTING.  ONCE THE
                                QUESTIONS HAVE BEEN CORRECTLY ANSWERED, AND IT IS ALSO
                                NECESSARY TO SELECT A PARTICULAR UNIT FOR TEST (TYPICAL
                                PROGRAM EXECUTION FROM ADDRESS 210), OR IT IS NECESSARY
                                TO RUN THE PROGRAM WITHOUT MANUAL INTERVENTION (TYPICAL
                                PROGRAM EXECUTION FROM ADDRESS 220). THE PROCESSOR MAY
                                BE HALTED AND RESTARTED FROM THE DESIRED RESTART ADDRESS.
                                IF ALL UNITS ARE TO BE CHECKED, THE PROCESSOR NEED NOT BE
                                TOUCHED.  THE PROGRAM WILL AUTOMATICALLY RESTART AT ADDRESS
                                200 AFTER RECEIVING THE NEW DEVICE PARAMETERS.

                                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.

                                220 START
                                ALL SWITCHES MUST BE DOWN FOR WORST CASE RUN. WITH THIS STARTING
                                ADDRESS THE PROGRAM WILL NOT RUN THOSE TESTS THAT NEED
```

MANUAL INTERVENTION. THIS IS RECOMMENDED ONLY FOR DEBUGGING WHERE THE ERROR IS NOT IN A TEST THAT REQUIRES MANUAL INTERVENTION

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 BEIDNG RUN ON A SWITCHLES PROCESSOR (I. E. AN 11/34) IT WILL DETERMINE THAT A HARWARE SWITCH REGISTER IS NOT PRESENT, AND WILL USE "SOFTWARE" SWITCH REGISTER. THE SETTINGS OF THE SWITCHES ARE CONTROLLED THROUGH A KEYBOARD ROUTINE WHICH IS CALED BY TYPING A 'COBNTROL G'. THE PROGRAM WILL RECOGNIZE A 'CONTROL G' AT ANY TIME EXCEPT WHEN IT IS AR A HIGHER PRIORITY PROCESSING AN RP04/5/6 INTERRUPT. THE "SOFTWARE" SWITCH VALUE S 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 REGISTER, 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 BE 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

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

IN THE EVENT THAT THE DISK DRIVE BECOMES UNAVAILABLE TO THE
CONTROLLER, POWERS DOWN, OR CERTAIN CRITICAL STATUS BITS
CANNOT BE CLEARED PRIOR TO THE START OF A TEST SEQUENCE -
THIS INFORMATION WILL BE COMMUNICATED TO THE OPERATIOR. 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 FAILURE CONTINUES TO OCCUR,
THERE ARE TWO OPTIONS OPEN TO 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 DZRPS, 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 FOLLOWISNG 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 SWITCH REGISTER FEATURE WHILE 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 1 MINUTES PROVIDED AN OPERATOR IS PRESENT TO CARRY OUT THE TYPED INSTRUCTIONS IMMEDIATELY. SUBSEQUENT PASSES WILL TAKE 30 SECONDS WHETHER AN OPERATOR IS THERE OR NOT.

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.
                        @
```

```
603
604

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


610
611
612                                   ;*INTERNAL PROGRAM MACROS BEGIN HERE
613                                   ;************************************
614
615
616                                ;**
617                                ;**NOTE:        ALL MACRO CALLS BEGINNING WITH ".$" ARE SUPPLIED FROM AN
618                                ;*       EXTERNAL SYSMAC.SML PACKAGE WHICH MUST BE MADE AVAILABLE
619                                ;*       TO THE SOURCE PROGRAM AT ASSEMBLY TIME.
620                                ;*
621
622
```

623

```
624
```

```
                                .SBTTL  STARTING ADDRESSES

                        000200          .=200
000200  000137  005012  RA:     JMP     @#BEGIN         ;NORMAL START
000204  000137  043606  ADDMOD: JMP     @#BASECH        ;START FOR ADDRESS-MODIFICATION
000210  000137  004776          JMP     @#BEGIN2        ;JUMP TO SELECT DRIVE START
                        000220          .=220
000220  000137  004762          JMP     @#BEGIN1        ;JUMP TO NO OPERATOR TESTS START


                        ;*STARTING ADDRESS 200 FOR NORMAL STARTS
                        ;*THIS WILL TEST ALL DRIVES ON THE SYSTEM A SINGLE DRIVE AT A TIME

                        ;*STARTING ADDRESS 204 FOR NON-DEFAULT ADDRESS PARAMETERS
                        ;*AUTO RESTART AT ADDRESS 200 AFTER LOADING PARAMETERS

                        ;*STARTING ADDRESS 210 WILL TEST ONLY ONE SPECIFIED DRIVE

                        ;*STARTING ADDRESS 220 WILL JUMP OVER THE TESTS REQUIRING AN OPERATOR
                        ;*AT THE DRIVE
```

```
   649          001110                          .=1110
```

```
                                ;*ITEM1
001226  051114                          EM1             ;RP04 DID NOT INTERRUPT
                                                        ;WAITED ON BIT DID NOT OCCUR
001230  067154                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;REG CONTENTS
                                                        ;RHCS1 CONTENTS
001232  071502                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001234  072010                          DF1             ;0,0,0,0,0,0

                                ;*ITEM2
001236  051143                          EM2             ;INTERRUPT ENABLE BIT DOWN BUT
                                                        ;WAITED ON BIT DID NOT OCCUR
001240  067154                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;REG CONTENTS
                                                        ;RHCS1 CONTENTS
001242  071502                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001244  072010                          DF1             ;0,0,0,0,0,0

                                ;*ITEM3
001246  051232                          EM3             ;RP04 DID NOT INTERRUPT WHEN
                                                        ;WAITED ON BIT DID SET
001250  067154                          DH1             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;RHCS1 CONTENTS
001252  071502                          DT1             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,CS1
001254  072010                          DF1             ;0,0,0,0,0,0

                                ;*ITEM4
001256  051313                          EM4             ;WAITED ON BIT DID SET BUT
                                                        ;TIME IS IN ERROR
                                                        ;TIME IS GIVEN IN 10 MICRO SEC.
                                                        ;(DECIMAL)
001260  067333                          DH4             ;PC
                                                        ;WAT PC
                                                        ;BIT WAITED
                                                        ;REG ADDRESS
                                                        ;TIME IN 10 MSEC
001262  071522                          DT4             ;$ERRPC,WAITPC,WAITBT,WAITRE,$BDDAT,WAITIM
001264  072017                          DF4             ;0,0,0,0,0,1

                                ;*ITEM5
001266  051424                          EM5             ;RHAS DOES NOT CLEAR BY
                                                        ;MOVING IN ALL ONES
```

```
706  001270  067474                    DH5                 ;PC
707                                                        ;REG. ADDR.
708                                                        ;GOOD DATA
709                                                        ;RECEIVED DATA
710  001272  071542                    DT5                 ;$ERRPC,REGADR,$GDDAT,$BDDAT
711  001274  072026                    DF5                 ;0,0,0,0
712
713                              ;*ITEM6
714  001276  051476                    EM6                 ;LOADING RHER1 FOR ALL
715                                                        ;UNITS DID NOT SET ANY BITS
716                                                        ;IN RHAS-NO UNITS PRESENT
717  001300  067613                    DH6                 ;PC
718                                                        ;REG ADDR
719                                                        ;RECEIVED DATA
720  001302  071556                    DT6                 ;$ERRPC,REGADR,$BDDAT
721  001304  072033                    DF6                 ;0,0,0
722
723                              ;*ITEM7
724  001306  051564                    EM7                 ;SPECIFIED REGISTER NONEXISTANT
725                                                        ;SO ABORT PROGRAM
726  001310  067712                    DH7                 ;PC
727                                                        ;ADDR. OF REG.
728  001312  071570                    DT7                 ;$ERRPC,TEMP1
729  001314  072037                    DF7                 ;0,0
730
731                              ;*ITEM10
732  001316  051634                    EM10                ;STOPED DRIVE HAS MOL BIT
733                                                        ;IN RHDS1 = 1
734  001320  067752                    DH10                ;PC
735                                                        ;TEST NO
736                                                        ;FAILING REG ADDR
737                                                        ;CONTENTS OF RHCS1
738                                                        ;CONTENTS OF RHCS2
739                                                        ;CONTENTS OF RHDS1
740                                                        ;CONTENTS OF RHER1
741  001322  071600                    DT10                ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
742  001324  072042                    DF10                ;0,0,0,0,0,0,0
743
744                              ;*ITEM11
745  001326  051703                    EM11                ;WITH SPINDLE POWERED DOWN
746                                                        ;RHCS2 SHOULD HAVE ONLY
747                                                        ;UNIT NUMBER AND IR HIGH
748  001330  067752                    DH10                ;PC
749                                                        ;TEST NO
750                                                        ;FAILING REG. ADR
751                                                        ;CONTENTS OF RHCS1
752                                                        ;CONTENTS OF RHCS2
753                                                        ;CONTENTS OF RHDS1
754                                                        ;CONTENTS OF RHER1
755  001332  071600                    DT10                ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
756  001334  072042                    DF10                ;0,0,0,0,0,0,0
757
758                              ;*ITEM12
759  001336  052010                    EM12                ;AFTER A POWER UP WITH
760                                                        ;NO PACK ACKNOWLEDGE COMMAND
761                                                        ;RHDS1 SHOULD HAVE MOL=1, VV=0
```

```
                                        DH10                    ;PC
                                                                ;TEST NO
                                                                ;FAILING REGISTER ADDR.
                                                                ;CONTENTS OF RHCS1
                                                                ;CONTENTS OF RHCS2
                                                                ;CONTETNS OF RHDS1
                                                                ;CONTENTS OF RHER1
                                        DT10                    ;$ERRPC,$TSTNM,$BDADR,CS1,CS2,DS1,ER1
                                        DF10                    ;0,0,0,0,0,0,0

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

                                ;*ITEM14
                                        EM14                    ;AFTER POWER UP RHCC
                                                                ;SHOULD BE=0
                                        DH5                     ;PC
                                                                ;REG. ADDR.
                                                                ;GOOD DATA
                                                                ;RECEIVED DATA
                                        DT5                     ;$ERRPC,REGADR,$GDDAT,$BDDAT
                                        DF5                     ;0,0,0,0

                                ;*ITEM15
                                        EM15                    ;PACK ACKNOWLEDGE CAUSED
                                                                ;AN ERROR
                                                                ;GOOD DATA IS BEFORE COMMAND
                                                                ;RECEIVED DATA IS AFTER COMMAND
                                        DH5                     ;PC
                                                                ;REG. ADDR.
                                                                ;GOOD DATA
                                                                ;RECEIVED DATA
                                        DT5                     ;$ERRPC,REGADR,$GDDAT,$BDDAT
                                        DF5                     ;0,0,0,0

                                ;*ITEM16
                                        EM16                    ;GIVING A NO-OP COMMAND CAUSED
                                                                ;AN ERROR
                                                                ;GOOD DATA GIVES REGISTER
                                                                ;CONTENTS BEFORE COMMAND
                                                                ;RECEIVED DATA GIVES REGISTER
                                                                ;CONTENTS AFTER COMMAND
                                        DH5                     ;PC
```

```
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001402  071542                      DTS                 ;$ERRPC,REGADR,$GDDAT,$BDDAT
001404  072026                      DFS                 ;0,0,0,0

                            ;*ITEM17
001406  052602                      EM17                ;DRIVE CLEAR COMMAND
                                                        ;CAUSED AN ERROR
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS
                                                        ;THERE AFTER COMMAND
001410  067474                      DHS                 ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001412  071542                      DTS                 ;$ERRPC,REGADR,$GDDAT,$BDDAT
001414  072026                      DFS                 ;0,0,0,0

                            ;*ITEM20
001416  052737                      EM20                ;READ-IN COMMAND GAVE AN ERROR
                                                        ;GOOD DATA HAS WHAT SHOULD BE THERE
                                                        ;RECEIVED DATA HAS WHAT WAS
                                                        ;AFTER COMMAND
001420  067474                      DHS                 ;PC
                                                        ;REG. ADDR.
                                                        ;GOOD DATA
                                                        ;RECEIVED DATA
001422  071542                      DTS                 ;$ERRPC,REGADR,$GDDAT,$BDDAT
001424  072026                      DFS                 ;0,0,0,0

                            ;*ITEM 21
001426  053103                      EM21                ;RHCS1 CONTENTS DURING
                                                        ;COMMAND WAS IN ERROR
001430  067474                      DHS
001432  071542                      DTS
001434  072026                      DFS

                            ;*ITEM 22
001436  053156                      EM22                ;RHDS1 CONTENTS DURING
                                                        ;COMM ANS WAS IN ERROR
001440  067474                      DHS
001442  071542                      DTS
001444  072026                      DFS

                            ;*ITEM 23
001446  053231                      EM23                ;UNLOAD COMMAND GAVE AN ERROR
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS
                                                        ;THERE AFTER COMMAND
001450  067474                      DHS
001452  071542                      DTS
001454  072026                      DFS
```

```
                                ;*ITEM 24
001456  053374                        EM24            ;OFFSET COMMAND CAUSED AN ERROR
                                                        ;GOOD DATA IS WHAT SHOULD BE THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS THERE
                                                        ;AFTER AN OFFSET COMMAND
001460  067474                        DH5
001462  071542                        DT5
001464  072026                        DF5

                                ;*ITEM 25
001466  053537                        EM25            ;RETURN TO CENTER LINE COMMAND
                                                        ;CAUSED AN ERROR
                                                        ;GOOD DATA GIVES WHAT SHOULD BE
                                                        ;THERE
                                                        ;RECEIVED DATA GIVES WHAT WAS
                                                        ;THERE AFTER COMMAND
001470  067474                        DH5
001472  071542                        DT5
001474  072026                        DF5

                                ;*ITEM 26
001476  053721                        EM26            ;500 OFFSETS CAUSED AN ERROR
001500  070131                        DH26            ;PC
                                                        ;CONT. OF RHCS1
                                                        ;CONT. OF RHCS2
                                                        ;CONT. OF RHDS1
                                                        ;CONT. OF RHER1
                                                        ;CONT. OF RHER2
                                                        ;CONT. OF RHER3
001502  071620                        DT26            ;$ERRPC,CS1,CS2,DS1,ER1,ER2,ER3
001504  072051                        DF26            ;0,0,0,0,0,0,0

                                ;*ITEM 27
001506  054011                        EM27            ;WRITE HEADER AND DATA
                                                        ;CAUSED IMPROPER REGISTER CHANGE
                                                        ;GOOD DATA GIVES WHAT
                                                        ;SHOULD BE THERE
                                                        ;RECEIVED DATA GIVES WHAT
                                                        ;WAS THERE AFTER COMMAND
001510  067474                        DH5
001512  071542                        DT5
001514  072026                        DF5

                                ;*ITEM 30
001516  054227                        EM30            ;WRITE HEADER AND DATA
                                                        ;CHANGED WRITE FROM BUFFER
001520  070330                        DH30            ;PC
                                                        ;WORD NO
                                                        ;GOOD DATA
                                                        ;BAD DATA
001522  071642                        DT30            ;$ERRPC,ERWORD,$GDDAT,$BDDAT
001524  072061                        DF30            ;0,0,0,0

                                ;*ITEM 31
001526  054307                        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  067474                          DH5
001532  071542                          DT5
001534  072026                          DF5

                        ;*ITEM 32
001536  054524                          EM32                   ;WRITE HEADER AND DATA FOLLOWED
                                                                ;BY A READ HEADER AND DATA
                                                                ;CAUSED A READ/WRITE ERROR
001540  070330                          DH30
001542  071642                          DT30
001544  072061                          DF30

                        ;*ITEM 33
001546  054633                          EM33                   ;READ DATA CAUSED IMPROPER REGISTER
                                                                ;CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS THERE AFTER
                                                                ;COMMAND
001550  067474                          DH5
001552  071542                          DT5
001554  072026                          DF5

                        ;*ITEM 34
001556  055035                          EM34                   ;READ DATA INCORRECT
001560  070330                          DH30
001562  071642                          DT30
001564  072061                          DF30

                        ;*ITEM 35
001566  055061                          EM35                   ;WRITE DATA COMMAND CAUSED
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD BE THERE
                                                                ;RECEIVED DATA GIVES REGISTER
                                                                ;CONTENTS AFTER WRITE DATA
001570  067474                          DH5
001572  071542                          DT5
001574  072026                          DF5

                        ;*ITEM 36
001576  055277                          EM36                   ;WRITE DATA COMMAND CHANGED
                                                                ;WRITE FROM BUFFER
001600  070330                          DH30
001602  071642                          DT30
001604  072061                          DF30

                        ;*ITEM 37
001606  055354                          EM37                   ;SEEK COMMAND CAUSED AN
                                                                ;ERROR
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
```

```
                                                                ;WAS THERE AFTER SEEK COMMAND
001610  067474                          DH5                     ;
001612  071542                          DT5                     ;
001614  072026                          DF5                     ;

                        ;*ITEM 40
001616  055571                          EM40                    ;WRITE CHECK CAUSED AN
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS
                                                                ;THERE AFTER COMMAND
001620  067474                          DH5
001622  071542                          DT5
001624  072026                          DF5

                        ;*ITEM 41
001626  056000                          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  067474                          DH5
001632  071542                          DT5
001634  072026                          DF5

                        ;*ITEM 42
001636  056261                          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  067474                          DH5
001642  071542                          DT5
001644  072026                          DF5

                        ;*ITEM 43
001646  056537                          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  070330                          DH30
001652  071642                          DT30
001654  072061                          DF30

                        ;*ITEM 44
```

```
1042  001656  057101                          EM44                    ;ENABLING WRITES BY WRITE LOCK
1043                                                                  ;BUTTON CAUSED AN ERROR
1044                                                                  ;GOOD DATA GIVES WHAT SHOULD
1045                                                                  ;BE THERE
1046                                                                  ;RECEIVED DATA GIVES WHAT WAS
1047                                                                  ;THERE AFTER WRITE LOCK
1048                                                                  ;BUTTON ENABLED WRITES
1049  001660  067474                            DH5                  ;
1050  001662  071542                            DT5                  ;
1051  001664  072026                            DF5                  ;
1052
1053                                  ;*ITEM 45
1054  001666  057373                          EM45                    ;TRANSFERRING ON LAST BLOCK IE. CYLINDER
1055                                                                  ;410, SECTOR 21, TRACK 18
1056                                                                  ;CAUSED IMPROPER REGISTER
1057                                                                  ;CHANGE
1058                                                                  ;GOOD DATA GIVES WHAT SHOULD
1059                                                                  ;BE THERE
1060                                                                  ;RECEIVED DATA GIVES WHAT WAS
1061                                                                  ;THERE AFTER TRANSFER
1062  001670  067474                            DH5
1063  001672  071542                            DT5
1064  001674  072026                            DF5
1065
1066                                  ;*ITEM 46
1067  001676  057667                          EM46                    ;DATA READ FROM LAST
1068                                                                  ;BLOCK IE. CYLINDER 410
1069                                                                  ;SECTOR 21, TRACK 18 IS IN
1070                                                                  ;ERROR
1071  001700  070330                            DH30
1072  001702  071642                            DT30
1073  001704  072061                            DF30
1074
1075                                  ;*ITEM 47
1076  001706  060002                          EM47                    ;TRANSFERRING FROM NONEXISTANT
1077                                                                  ;SECTOR CAUSED IMPROPER
1078                                                                  ;REGISTER CHANGE
1079                                                                  ;GOOD DATA GIVES WHAT SHOULD
1080                                                                  ;BE THERE
1081                                                                  ;RECEIVED DATA GIVES WHAT WAS
1082                                                                  ;THERE AFTER ATTEMPTED
1083                                                                  ;TRANSFER
1084  001710  067474                            DH5
1085  001712  071542                            DT5
1086  001714  072026                            DF5
1087
1088                                  ;*ITEM 50
1089  001716  060264                          EM50                    ;TRANSFERRING FROM NONEXISTANT
1090                                                                  ;SECTOR CAUSED DATA ERROR
1091                                                                  ;GOOD DATA GIVES WHAT
1092                                                                  ;SHOULD BE IN BUFFER
1093                                                                  ;RECEIVED DATA GIVES WHAT WAS
1094                                                                  ;IN BUFFER AFTER TRANSFER
1095  001720  070330                            DH30
1096  001722  071642                            DT30
1097  001724  072061                            DF30
```

```
1098
1099                                    ;*ITEM 51
1100  001726  060503                           EM51          ;GIVING ILLEGAL FUNCTION CAUSED
1101                                                          ;IMPROPER REGISTER CHANGE
1102                                                          ;GOOD DATA GIVES WHAT SHOULD BE
1103                                                          ;THERE
1104                                                          ;RECEIVED DATA GIVES REGISTER
1105                                                          ;CONTENTS AFTER ILLEGAL FUNCTION
1106  001730  070444                           DH51          ;PC
1107                                                          ;REG. ADDR.
1108                                                          ;GOOD DATA
1109                                                          ;RECEIVED DATA
1110                                                          ;ILLEGAL FUNCTION
1111  001732  071656                           DT51          ;$ERRPC,REGADR,$GDDAT,$BDDAT,ILLEGL
1112  001734  072066                           DF51          ;0,0,0,0,0
1113
1114
1115                                    ;*ITEM 52
1116  001736  060750                           EM52          ;WRITE DATA ON NONEXISTANT
1117                                                          ;SECTOR CAUSED IMPROPER
1118                                                          ;REGISTER CHANGE
1119                                                          ;GOOD DATA GIVES WHAT SHOULD
1120                                                          ;BE THERE
1121                                                          ;RECEIVED DATA GIVES WHAT
1122                                                          ;WAS THERE AFTER ATTEMPTED
1123                                                          ;WRITE DATA
1124  001740  067474                           DH5
1125  001742  071542                           DT5
1126  001744  072026                           DF5
1127
1128                                    ;*ITEM 53
1129  001746  061225                           EM53          ;READ HEADER AND DATA AFTER
1130                                                          ;A SEARCH CAUSED AN ERROR
1131  001750  070330                           DH30
1132  001752  071642                           DT30
1133  001754  072061                           DF30
1134
1135                                    ;*ITEM 54
1136  001756  061313                           EM54          ;ATTEMPTED OPERATION WITH
1137                                                          ;INVALID ADDRESS CAUSED
1138                                                          ;IMPROPER REGISTER CHANGE
1139                                                          ;GOOD DATA GIVES WHAT SHOULD
1140                                                          ;BE THERE
1141                                                          ;RECEIVED DATA GIVES WHAT WAS
1142                                                          ;THERE AFTER OPERATION
1143  001760  067474                           DH5
1144  001762  071542                           DT5
1145  001764  072026                           DF5
1146
1147                                    ;*ITEM 55
1148  001766  061560                           EM55          ;WRITING/READING WITH EXPECTED
1149                                                          ;ADDRESS OVERFLOW ERROR CAUSED
1150                                                          ;IMPROPER REGISTER CHANGE
1151                                                          ;GOOD DATA GIVES WHAT SHOULD
1152                                                          ;BE THERE
1153                                                          ;RECEIVED DATA GIVES WHAT
```

```
1154
1155  001770  067474                    DH5                     ;WAS THERE AFTER OPERATION
1156  001772  071542                    DT5
1157  001774  072026                    DF5
1158
1159                              ;*ITEM 56
1160  001776  062046                    EM56                    ;DATA READ WITH AN EXPECTED
1161                                                            ;ADDRESS OVERFLOW ERROR IS
1162                                                            ;INCORRECT
1163                                                            ;WORD NO 1 TO 260 SHOULD
1164                                                            ;BE READ
1165                                                            ;WORD NOS 261 TO 266 SHOULD
1166                                                            ;NOT CHANGE DUE TO READ
1167  002000  070330                    DH30
1168  002002  071642                    DT30
1169  002004  072061                    DF30
1170
1171                              ;*ITEM 57
1172  002006  062256                    EM57                    ;ATTEMPTING DATA COMMAND
1173                                                            ;WITH WRONG FORMAT BIT CAUSED
1174                                                            ;IMPROPER REGISTER CHANGE
1175                                                            ;GOOD DATA GIVES WHAT SHOULD BE
1176                                                            ;THERE
1177                                                            ;RECEIVED DATA GIVES WHAT WAS
1178                                                            ;THERE AFTER ATTEMPTED DATA
1179                                                            ;TRANSFER
1180  002010  067474                    DH5
1181  002012  071542                    DT5
1182  002014  072026                    DF5
1183
1184                              ;*ITEM 60
1185  002016  062550                    EM60                    ;ATTEMPTING TO MODIFY REGISTER
1186                                                            ;DURING AN OPERATION CAUSED
1187                                                            ;IMPROPER REGISTER CHANGE
1188                                                            ;GOOD DATA GIVES WHAT SHOULD
1189                                                            ;BE THERE
1190                                                            ;RECEIVED DATA GIVES WHAT WAS
1191                                                            ;THERE AFTER OPERATION
1192                                                            ;WAS COMPLETE
1193  002020  070603                    DH60                    ;PC
1194                                                            ;REG. ADDR.
1195                                                            ;GOOD DATA
1196                                                            ;RECEIVED DATA
1197                                                            ;MODFING REGISTER
1198  002022  071674                    DT60                    ;$ERRPC,REGADR,$GDDAT,$BDDAT,$BDADR
1199  002024  072074                    DF60                    ;0,0,0,0,0
1200
1201                              ;*ITEM 61
1202  002026  063161                    EM61                    ;DEVICE NOT AVAILBLE BEFORE COMMAND WAS TO BE GIVEN
1203  002030  070740                    DH61                    ;PC
1204                                                            ;TEST NO.
1205                                                            ;PC OF JSR
1206                                                            ;RHCS1 CONTENTS
1207  002032  071712                    DT61                    ;$ERRPC,$TSTNM,PCJSR,$BDADR
1208  002034  072102                    DF61                    ;0,0,0,0
1209
```

```
1210                                    ;*ITEM 62
1211  002036  063242                           EM62            ;RHDS1 HAS STATUS BITS STUCK AT ONE
1212  002040  071033                           DH62            ;PC
1213                                                            ;TEST NO.
1214                                                            ;PC OF JSR
1215                                                            ;RHDS1 CONTENTS
1216  002042  071724                           DT62            ;$ERRPC,$TSTNM,PCJSR,$BDADR
1217  002044  072106                           DF62            ;0,0,0,0
1218
1219
1220                                    ;*ITEM 63
1221  002046  063323                           EM63            ;RHDS1 CONTENTS DURING
1222                                                            ;COMMAND WAS IN ERROR
1223  002050  067474                           DH5
1224  002052  071542                           DT5
1225  002054  072026                           DF5
1226
1227
1228                                    ;*ITEM 64
1229  002056  063377                           EM64            ;RECALIBRATE COMMAND CAUSED
1230                                                            ;IMPROPER REGISTER CHANGE.
1231                                                            ;GOOD DATA GIVES WHAT SHOULD BE
1232                                                            ;THERE.
1233                                                            ;RECEIVED DATA GIVES WHAT WAS THERE
1234                                                            ;AFTER COMMAND
1235  002060  067474                           DH5
1236  002062  071542                           DT5
1237  002064  072026                           DF5
1238
1239
1240                                    ;*ITEM65
1241
1242  002066  063616                           EM65            ;INTERRUPT FAILING
1243  002070  071126                           DH65            ;PC
1244                                                            ;TEST NO
1245                                                            ;CONTENTS OF RHCS1
1246                                                            ;CONTENTS OF RHAS
1247                                                            ;CONTENTS OF RHDS1
1248  002072  071736                           DT65            ;$ERRPC,TSTNM,CS1,AS,DS1
1249  002074  072112                           DF65            ;0,0,0,0,0
1250
1251
1252                                    ;*ITEM66
1253  002076  063640                           EM66            ;HEADER AND DATA COMMAND
1254                                                            ;FOR HEAD SELECTION TEST
1255                                                            ;CAUSED AN ERROR
1256                                                            ;RHDST GIVES WHAT TRACK
1257                                                            ;WAS BEING WRITTEN ON CYLINDER 0
1258                                                            ;SECTOR 0
1259  002100  071251                           DH66            ;PC
1260                                                            ;RHDST
1261                                                            ;RHER1
1262                                                            ;RHER2
1263                                                            ;RHER3
1264                                                            ;RHCS1
1265                                                            ;RHCS2
```

```
1266  002102  071752                    DT66          ;SERRPC,DST,ER1,ER2,ER3,CS1,CS2
1267  002104  072117                    DF66          ;0,0,0,0,0,0,0
1268                              ;*ITEM67
1269  002106  064032                    EM67          ;READ HEADER AND DATA ERROR
1270                                                   ;IN HEAD SELECTION TEST
1271                                                   ;FIRST FOUR WORDS GIVE HEADER
1272                                                   ;NEXT WORDS ARE DATA
1273                                                   ;GOOD DATA WORDS GIVE
1274                                                   ;THE TRACK NUMBER IN
1275                                                   ;BITS 4,5,6,7,8
1276  002110  070330                    DH30
1277  002112  071642                    DT30
1278  002114  072061                    DF30
1279                              ;*ITEM70
1280  002116  064322                    EM70          ;READ HEADER AND DATA ERROR
1281                                                   ;IN DIFFERENCE LINE TEST
1282                                                   ;WORD NOS. 1-4 GIVE
1283                                                   ;HEADER
1284                                                   ;WORD NOS. 5-260 GIVE DATA
1285                                                   ;WHICH IS THE CYLINDER
1286                                                   ;ADDRESS
1287  002120  070330                    DH30
1288  002122  071642                    DT30
1289  002124  072061                    DF30
1290
1291                              ;*ITEM 71
1292  002126  064530                    EM71          ;FORCING OPI CAUSED IMPROPER REGISTER
1293                                                   ; CHANGE
1294                                                   ;GOOD DATA GIVES WHAT SHOULD
1295                                                   ;BE THERE
1296                                                   ;RECEIVED DATA GIVES WHAT WAS
1297                                                   ;THERE AFTER 3 INDEX PULSES
1298  002130  067474                    DH5           ;PC
1299                                                   ;REG. ADDR.
1300                                                   ;GOOD DATA
1301                                                   ;RECEIVED DATA
1302  002132  071542                    DT5           ;$ERRPC,REGADR,$GDDAT,$BDDAT
1303  002134  072026                    DF5           ;0,0,0,0
1304                              ;*ITEM 74
1305  002136  065273                    EM74          ;WHILE USING UNIBUS B
1306                                                   ;READ DATA CAUSED IMPROPER REGISTER
1307                                                   ;CHANGE
1308                                                   ;GOOD DATA GIVES WHAT SHOULD BE THERE
1309                                                   ;RECEIVED DATA GIVES WHAT WAS THERE AFTER
1310                                                   ;COMMAND
1311  002140  067474                    DH5
1312  002142  071542                    DT5
1313  002144  072026                    DF5
1314
1315                              ;*ITEM 73
1316  002146  065221                    EM73          ;WHILE USING UNIBUS B
1317                                                   ;READ DATA INCORRECT
1318  002150  070330                    DH30
1319  002152  071642                    DT30
1320  002154  072061                    DF30
1321
```

```
1322                                    ;*ITEM 74
1323  002156  065273                            EM74            ;WHILE USING UNIBUS B
1324                                                            ;WRITE DATA COMMAND CAUSED
1325                                                            ;IMPROPER REGISTER CHANGE
1326                                                            ;GOOD DATA GIVES WHAT SHOULD BE THERE
1327                                                            ;RECEIVED DATA GIVES REGISTER
1328                                                            ;CONTENTS AFTER WRITE DATA
1329  002160  067474                            DH5
1330  002162  071542                            DT5
1331  002164  072026                            DF5
1332
1333                                    ;*ITEM 75
1334  002166  065537                            EM75            ;WHILE USING UNIBUS B
1335                                                            ;WRITE DATA COMMAND CHANGED
1336                                                            ;WRITE FROM BUFFER
1337  002170  070330                            DH30
1338  002172  071642                            DT30
1339  002174  072061                            DF30
1340
1341                                    ;*ITEM 76
1342  002176  065642                            EM76            ;WHILE USING UNIBUS B
1343                                                            ;WRITE CHECK CAUSED AN
1344                                                            ;IMPROPER REGISTER CHANGE
1345                                                            ;GOOD DATA GIVES WHAT SHOULD
1346                                                            ;BE THERE
1347                                                            ;RECEIVED DATA GIVES WHAT WAS
1348                                                            ;THERE AFTER COMMAND
1349  002200  067474                            DH5
1350  002202  071542                            DT5
1351  002204  072026                            DF5
1352
1353                                    ;*ITEM 77
1354  002206  066077                            EM77            ;CURRENT CYLINDER DOES NOT REFLECT DESIRED 'RHCC'
1355  002210  071347                            DH77            ;PC
1356                                                            ;PC OF JSR
1357                                                            ;REGISTER ADDRESS
1358                                                            ;GOOD DATA
1359                                                            ;BAD DATA
1360  002212  071772                            DT77            ;$ERRPC,PCJSR,REGADR,$GDDAT,$BDDAT
1361  002214  072127                            DF77            ;0,0,0,0,0
1362
1363                                    ;*ITEM 100
1364  002216  066322                            EM100           ;ERROR AFTER ADDRESS PLUG CHANGE
1365  002220  067474                            DH5             ;PC
1366                                                            ;REGISTER ADDRESS
1367                                                            ;GOOD DATA
1368                                                            ;RECEIVED DATA
1369  002222  071542                            DT5             ;$ERRPC,REGADR,$GDDAT,$BDDAT
1370  002224  072026                            DF5             ;0,0,0,0
1371
1372                                    ;*ITEM 101
1373  002226  066404                            EM101           ;UNIT DID NOT GO OFFLINE WHEN ADDR
1374                                                            ;PLUG WAS REMOVED
1375  002230  070131                            DH26            ;PC
1376                                                            ;CONT OF RHCS1
1377                                                            ;CONT OF RHCS2
```

```
1378
1379
1380
1381  002232  071620                    DT26        ;$ERRPC,CS1,CS2,DS1,ER2,ER3
1382  002234  072051                    DF26        ;0,0,0,0,0,0,0
1383
1384                              ;*ITEM 102
1385  002236  066466                    EM102       ;UNIT DID NOT COME BACK ONLINE WHEN
1386                                                 ;ADDR PLUG WAS REPLACED
1387  002240  070131                    DH26        ;PC
1388                                                 ;CONT OF RHCS1
1389                                                 ;CONT OF RHCS2
1390                                                 ;CONT OF RHDS1
1391                                                 ;CONT OF RHER2
1392                                                 ;CONT OF RHER3
1393  002242  071620                    DT26        ;$ERRPC,CS1,CS2,DS1,ER2,ER3
1394  002244  072051                    DF26        ;0,0,0,0,0,0,0
1395
1396                              ;*ITEM  103
1397  002246  066545                    EM103       ;REGISTER CONTENTS INCORRECT BEFORE A
1398                                                 ;DIAGNOSTIC SEEK
1399  002250  070131                    DH26        ;PC
1400                                                 ;CONT OF RHCS1
1401                                                 ;CONT OF RHCS2
1402                                                 ;CONT OF RHDS1
1403                                                 ;CONT OF RHER2
1404                                                 ;CONT OF RHER3
1405  002252  071620                    DT26        ;$ERRPC,CS1,CS2,DS1,ER2,ER3
1406  002254  072051                    DF26        ;0,0,0,0,0,0,0
1407
1408                              ;*ITEM  104
1409  002256  066631                    EM104       ;REGISTER CONTENTS INCORRECT AFTER A
1410                                                 ;DIAGNOSTIC SEEK
1411  002260  070131                    DH26        ;PC
1412                                                 ;CONT OF RHCS1
1413                                                 ;CONT OF RHCS2
1414                                                 ;CONT OF RHDS1
1415                                                 ;CONT OF RHER2
1416                                                 ;CONT OF RHER3
1417  002262  071620                    DT26        ;$ERRPC,CS1,CS2,DS1,ER2,ER3
1418  002264  072051                    DF26        ;0,0,0,0,0,0,0
1419
```

```
1420
1421                                    ;*RH11 REGISTERS
1422
1423
1424  002266  000254                    RPVEC:  254                     ;RP04 VECTOR ADDRESS
1425
1426
1427
1428                                    ;*WORD COUNT REGISTER (RHWC,
1429                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1430
1431
1432
1433                                    ;*BUS ADDRESS REGISTER (RHBA)
1434                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1435
1436
1437
1438                                    ;*CONTROL AND STATUS REGISTER 2 (RHCS2)
1439
1440          000001                    US1=    1                       ;UNIT SELECT (BIT #0)
1441          000002                    US2=    2                       ;UNIT SELECT (BIT #1)
1442          000004                    US4=    4                       ;UNIT SELECT (BIT #2)
1443          000010                    BAI=    10                      ;BUS ADDRESS INCREMENT INHIBIT (BIT #3)
1444          000020                    UNIB=   20                      ;UNIBUS B DC LO (BIT #4)
1445          000040                    CLR=    40                      ;CLEAR (BIT #5)
1446          000100                    IR=     100                     ;INPUT READY (BIT #6)
1447          000200                    OR=     200                     ;OUTPUT READY (BIT #7)
1448          000400                    MPE=    400                     ;MASS BUS PARITY ERROR (BIT #8)
1449          001000                    MXF=    1000                    ;MISSED TRANSFER ERROR (BIT #9)
1450          002000                    PGE=    2000                    ;PROGRAM ERROR (BIT #10)
1451          004000                    NEM=    4000                    ;NON EXISTANT MEMORY (BIT #11)
1452          010000                    NED=    10000                   ;NON EXISTANT DRIVE (BIT #12)
1453          020000                    UPE=    20000                   ;UNIBUS PARITY ERROR (BIT #13)
1454          040000                    WCE=    40000                   ;WRITE CHECK ERROR (BIT #14)
1455          100000                    DLT=    100000                  ;DATA LATE (BIT #15)
1456
1457                                    ;*DATA BUFFER REGISTER (RHDB)
1458                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1459
1460
```

```
                                ;*RP04 REGISTERS



                                ;*CONTROL AND STATUS 1 REGISTER. (#00)

000001                          GO=     1               ;GO (BIT #0)
000100                          IE=     100             ;INTERRUPT ENABLE (BIT #6)
000200                          RDY=    200             ;READY (BIT #7)
000400                          A16=    400             ;HIGH ORDER UNIBUS BITS (BIT #8)
001000                          A17=    1000            ;HIGH ORDER UNIBUS BITS (BIT #9)
002000                          PSEL=   2000            ;PORT SELECT (BIT #10)
004000                          DVA=    4000            ;DEVICE AVAILABLE (BIT #11)
020000                          MCPE=   20000           ;MASSBUSS PARITY ERROR (BIT #13)
040000                          TRE=    40000           ;TRANSFER ERROR (BIT #14)
100000                          SC=     100000          ;SPECIAL CONDITION (BIT #15)

                                ;*STATUS REGISTER (RHDS1) (#01)

000001                          DFF5=   1               ;DRIVE FORWARD 5"/SEC.  (BIT #0)
000002                          DFF20=  2               ;DRIVE FORWARD 20"/SEC.  (BIT #1)
000004                          DIGB=   4               ;DRIVE TO INNER GAVRD BAND (BIT #2)
000010                          GRV=    10              ;GO REVERSE (BIT #3)
000020                          DL64=   20              ;DIFFERENCE LESS THAN 64 (BIT #4)
000040                          DE1=    40              ;DIFFERENCE EQUALS 1 (BIT #5)
000100                          VV=     100             ;VOLUME VALID (BIT #6)
000200                          DRY=    200             ;DRIVE READY (BIT #7)
000400                          DPR=    400             ;DRIVE PRESENT (BIT #8)
001000                          PROG=   1000            ;PROGRAMABLE (BIT #9)
002000                          LBT=    2000            ;LAST SECTOR TRANSFERRED (BIT #10)
004000                          WRL=    4000            ;WRITE LOCK (BIT #11)
010000                          MOL=    10000           ;MEDIUM ON-LINE (BIT #12)
020000                          PIP=    20000           ;POSITIONING OPERATION IN PROGRESS (BIT #13)
040000                          ERR=    40000           ;COMPOSIT ERROR. (BIT #14)
100000                          ATA=    100000          ;ATTENTION ACTIVE (BIT #15)

                                ;*ERROR REGISTER #01 (RHER1) (#02)
000001                          ILF=    1               ;ILLEGAL FUNCTION (BIT #0)
000002                          ILR=    2               ;ILLEGAL REGISTER (BIT #1)
000004                          RMR=    4               ;REGISTER MODIFICATION REFUSED (BIT #2)
000010                          PAR=    10              ;PARITY ERROR (BIT #3)
000020                          FER=    20              ;FORMAT ERROR (BIT #4)
000040                          WCF=    40              ;WRITE CLOCK FAIL (BIT #5)
000100                          ECH=    100             ;ECC HARD ERROR (BIT #6)
000200                          HCE=    200             ;HEADER COMPARE ERROR (BIT #7)
000400                          HCRC=   400             ;HEADER CRC ERROR (BIT #8)
001000                          AOE=    1000            ;ADDRESS OVERFLOW ERROR (BIT #9)
002000                          IAE=    2000            ;INVALID ADDRESS ERROR (BIT #10)
004000                          WLE=    4000            ;WRITE LOCK ERROR (BIT #11)
010000                          DTE=    10000           ;DRIVE TIMING ERROR (BIT #12)
020000                          OPI=    20000           ;OPERATION INCOMPLETE (BIT #13)
040000                          UNS=    40000           ;DRIVE UNSAFE (BIT #14)
100000                          DCK=    100000          ;DATA CHECK ERROR (BIT 15)

                                ;*MAINTAINABILITY REGISTER (RHMR)(#03)
```

```
                        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)

                        ;*ERROR REGISTER #2 (RHER2) (#10)

                        WCU=    1               ;WRITE CURRENT UNSAFE (BIT #0)
                        CSF=    2               ;CURRENT SINK FAILURE (BIT #1)
```

```
1573            000004                  WSU=    4                       ;WRITE SELECT UNSAFE (BIT #2)
1574            000010                  CSU=    10                      ;CURRENT SWITCH UNSAFE (BIT #3)
1575            000020                  MSE=    20                      ;MOTOR SEQUENCE ERROR (BIT #4)
1576            000040                  TDF=    40                      ;TRANSITIONS DETECTOR FAILURE (BIT #5)
1577            000100                  TUF=    100                     ;TRANSITIONS UNSAFE (BIT #6)
1578            000200                  FEN=    200                     ;FAILSAFE ENABLED (BIT #7)
1579            000400                  WRU=    400                     ;WRITE READY UNSAFE (BIT #8)
1580            001000                  MHS=    1000                    ;MULTIPLE HEAD SELECT (BIT #9)
1581            002000                  NHS=    2000                    ;NO HEAD SELECTION (BIT #10)
1582            004000                  IXE=    4000                    ;INDEX ERROR (BIT #11)
1583            010000                  VU30=   10000                   ;30VOLT UNSAFE (BIT #12)
1584            020000                  PLU=    20000                   ;PLO UNSAFE (BIT #13)
1585            100000                  ACU=    100000                  ;ACUNSAFE (BIT #15)
1586
1587                                    ;*OFFSET REGISTER (RHOF) (#11)
1588
1589            000001                  OF25=   1                       ;OFFSET 25 MICRO INCHES (BIT #0)
1590            000002                  OF50=   2                       ;OFFSET 50 MICRO INCHES (BIT #1)
1591            000004                  OF100=  4                       ;OFFSET 100 MICRO INCHES (BIT #2)
1592            000010                  OF200=  10                      ;OFFSET 200 MICRO INCHES (BIT #3)
1593            000020                  OF400=  20                      ;OFFSET 400 MICRO INCHES (BIT #4)
1594            000040                  OF800=  40                      ;OFFSET 800 MICRO INCHES (BIT #5)
1595
1596            000200                  OFREV=  200                     ;OFFSET NEGATIVE (REVERSE) (BIT #5)
1597            002000                  HCI=    2000                    ;HEADER COMPARE INHIBIT (BIT #10)
1598            004000                  ECI=    4000                    ;ERROR CORRECTION CODE INHIBIT (BIT #11)
1599            010000                  FMT22=  10000                   ;FORMAT BIT (BIT #12
1600
1601                                    ;*DESIRED CYLINDER ADDRESS (RHCA) (#12)
1602                                    ;*EACH BIT IS CALLED BY BIT NUMBER.
1603
1604
1605
1606
1607                                    ;*CURRENT CYLINDER ADDRESS (RHCC) (#13)
1608                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1609
1610
1611
1612
1613                                    ;*SERIAL NUMBER REGISTER (RHSN) (#14)
1614                                    ;*EACH IS CALLED BY BIT NUMBER
1615
1616
1617
1618
1619                                    ;*ERROR REGISTER #03 (RHER3) (#15)
1620
1621            000001                  PSU=    1                       ;PACK SPEED UNSAFE (BIT #0)
1622            000002                  VUF=    2                       ;VELOCITY UNSAFE (BIT #1)
1623            000010                  UWR=    10                      ;ANY UNSAFE EXCEPT READ/WRITE (BIT #3)
1624            000020                  PRE=    20                      ;DISK PACK ROTATION ERROR (BIT #4)
1625            000040                  ACL=    40                      ;AC LOW (BIT #5)
1626            000100                  DCL=    100                     ;DC LOW (BIT #6)
1627            020000                  ACE=    20000                   ;ADDRESS CHANGE ERROR (BIT #13)
1628            040000                  SKI=    40000                   ;SEEK INCOMPLETE (BIT #14)
```

```
1629            100000                  OCYL=   100000                  ;OFF CYLINDER (BIT #15)
1630
1631
1632
1633                                    ;*ECC POSITION REGISTER (RHEC1) (#16)
1634                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1635
1636
1637
1638
1639                                    ;*ECC PATTERN REGISTER (RHEC2) (#17)
1640                                    ;*EACH BIT IS CALLED BY BIT NUMBER
1641
1642
1643
1644
1645
1646
1647
1648
1649
```

```
                        .SBTTL  REGISTER ADDRESSES



                        ;*RP04/5/6 DISK I/O REGISTER LOCATED IN THE RH11 CONTROLLER

002270  176722          RHDB:   176722          ;DATA BUFFER
002272  176702          RHWC:   176702          ;WORD COUNT
002274  176704          RHBA:   176704          ;BUS ADDRESS
002276  176710          RHCS2:  176710          ;CONTROL AND STATUS 2

                        ;*RP04/5/6 DISK I/O REGISTERS LOCATED IN THE DEVICE CONTROL LOGIC (DCL)

002300  176700          RHCS1:  176700          ;CONTROL AND STATUS 1
002302  176714          RHER1:  176714          ;ERROR #1
002304  176706          RHDST:  176706          ;DESIRED SECTOR/TRACK ADDRESS
002306  176740          RHER2:  176740          ;ERROR #2
002310  176732          RHOF:   176732          ;OFFSET
002312  176734          RHCA:   176734          ;DESIRED CYLINDER ADDRESS
002314  176742          RHER3:  176742          ;ERROR #3
002316  176716          RHAS:   176716          ;ATTENTION SUMMARY
002320  176724          RHMR:   176724          ;MAINTAINABILITY
002322  176712          RHDS1:  176712          ;DRIVE STATUS
002324  176726          RHOT:   176726          ;DRIVE TYPE
002326  176730          RHSN:   176730          ;SERIAL NUMBER
002330  176744          RHEC1:  176744          ;ECC POSITION
002332  176746          RHEC2:  176746          ;ECC PATTERN
002334  176736          RHCC:   176736          ;CURRENT CYLINDER ADDRESS
002336  176720          RHLA:   176720          ;LOOK-AHEAD

                        ;*ADDITIONAL I/O REGISTERS LOCATED IN THE RH70 CONTROLLER LOGIC

002340  176750          RHBAE:  176750          ;BUS ADDRESS EXTENSION REGISTER
002342  176752          RHCS3:  176752          ;CONTROL AND STATUS REGISTER #3


                        ;*P-CLOCK (KW11-P) I/O REGISTERS

002344  172540          PCLCSR: 172540          ;CONTROL AND STATUS REGISTER
002346  172542          PCLBUF: 172542          ;COUNT SET BUFFER
002350  172544          PCLCTR: 172544          ;COUNTER

```

```
1693
1694
1695                            ;*THE FOLLOWING LOCATIONS ARE RESERVED FOR REGISTER SNAPSHOTS
1696                            ;*ANY TIME THERE IS AN ERROR ALL THESE WILL BE FILLED
1697
1698                            ;*ONLY SOME MAY BE PRINTED BUT ALL WILL BE FILLED TRUE
1699                            ;*FOR THE TIME JUST AFTER THE "ERROR" ERROR COMMAND
1700
1701
1702                            ;*THIS ASSUMES THAT A REGISTER SNAPSHOT HAS BEEN TAKEN WHICH IS NOT
1703                            ;*ALWAYS THE CASE - IF QUESTIONABLE CONTENTS APPEAR IN THE REGISTER
1704                            ;*PRINTOUTS, CHECK THE INLINE TEST CODE TO SEE IF THE REGISTER SNAPSHOT
1705                            ;*REFLECTS THE CURRENT STATE OF THE MACHINE
1706
1707
1708  002352  000000            DB:     0                       ;DATA BUFFER
1709  002354  000000            WC:     0                       ;WORD COUNT
1710  002356  000000            BA:     0                       ;BUS ADDRESS
1711  002360  000000            CS2:    0                       ;CONTROL AND STATUS 2
1712
1713
1714  002362  000000            CS1:    0                       ;CONTROL AND STATUS 1
1715  002364  000000            ER1:    0                       ;ERROR #1
1716  002366  000000            DST:    0                       ;DESIRED SECTOR/TRACK ADDRESS
1717  002370  000000            ER2:    0                       ;ERROR #2
1718  002372  000000            OF:     0                       ;OFFSET
1719  002374  000000            CA:     0                       ;DESIRED CYLINDER ADDRESS
1720  002376  000000            ER3:    0                       ;ERROR #3
1721  002400  000000            AS:     0                       ;ATTENTION SUMMARY
1722  002402  000000            MR:     0                       ;MAINTAINABILITY
1723  002404  000000            DS1:    0                       ;DRIVE STATUS
1724  002406  000000            DT:     0                       ;DRIVE TYPE
1725  002410  000000            SN:     0                       ;SERIAL NUMBER
1726  002412  000000            EC1:    0                       ;ECC POSITION
1727  002414  000000            EC2:    0                       ;ECC PATTERN
1728  002416  000000            CC:     0                       ;CURRENT CYLINDER ADDRESS
1729  002420  000000            LA:     0                       ;LOOK-AHEAD
1730
```

```
                                ;*FUNCTION EQUATES

                                        ;*TABLE OF FUNCTIONS FOR RHCS1 THEN "GO" BIT HAS TO BE SET
002422                          FUTABL:
002422  000000                  NOPERA: 0                       ;NO OPERATION
002424  000002                  UNLOAD: 2                       ;UNLOAD (STAND BY)
002426  000006                  RECALI: 6                       ;RECALIBRATE
002430  000010                  DCLEAR: 10                      ;DRIVE CLEAR
002432  000012                  RELEAS: 12                      ;RELEASE (DUAL-PORT OPERATION)
002434  000030                  SERCH:  30                      ;SEARCH COMMAND
002436  000050                  WRCHEK: 50                      ;WRITE CHECK DATA
002440  000052                  WRCHOT: 52                      ;WRITE CHECK HEADER AND DATA
002442  000060                  WRIDAT: 60                      ;WRITE DATA
002444  000062                  WRIFOR: 62                      ;WRITE HEADER AND DATA (FORMAT)
002446  000070                  READAT: 70                      ;READ DATA
002450  000072                  REFOR:  72                      ;READ HEADER AND DATA
002452  000004                  SEECOM: 4                       ;SEEK COMMAND
002454  000014                  OFSETC: 14                      ;OFFSET COMMAND
002456  000016                  RETCL:  16                      ;RETURNTO CENTERLINE
002460  000022                  PKACK:  22                      ;PACK ACKNOWLEDGE
002462  000020                  READIN: 20                      ;READ IN
002464  000000                  ILLEGL: .WORD   0               ;COMPUTED ILLEGAL FUNCTION


                                ;*DATA BUFFERS FOR READ/WRITE


002470  000422                  WRFROM: .BLKW   274.            ;WRITE FROM THIS BUFFER
003534  000422                  REINTO: .BLKW   274.            ;READ INTO THIS BUFFER
```

```
1764
1765
1766                                    ;*RESERVED CORE LOCATIONS
1767
1768  004600  000000                    REGADR: 0                       ;SAVE REGISTER ADDRESS HERE
1769  004602  000000                    ERWORD: 0                       ;SAVE ERROR WORD NUMBER HERE
1770  004604  000000                    TSTNM:  0                       ;TEST NUMBER
1771  004606  000000                    RP4VEC: 0                       ;CONTAINS ADDRESS OF LOCATION
1772                                                                    ;WHERE AN RP04 INTERRUPT IS TO VECTOR TO
1773                                                                    ;THIS MUST BE MOVED INTO 'RPVEC' TO BE
1774                                                                    ;EFFECTIVE.
1775
1776  004610  000000                    OFSTVL: 0                       ;OFFSET VALUE USED IN OFFSET TEST
1777
1778
1779  004612  000024                    SAVERE: .BLKW   20.             ;BLOCK TO SAVE REGISTERS FOR PRETEST
1780                                                                    ;HARDWARE REGISTER SNAPSHOTS - THESE
1781                                                                    ;ARE USUALLY THEN CHANGED TO REFLECT
1782                                                                    ;EXPECTED CONDITIONS AFTER THE TEST
1783  004662  000000                    FINALA: 0                       ;SAVE LOOK AHEAD REGISTER AT END OF OPERATION
1784  004664  000000                    FINACC: 0                       ;SAVE CURRENT CYLINDER REGISTER AT END OF OPERATION
1785
1786
1787                                    ;*TABLE FOR ATTENTION BITS
1788                                    ;*ATTENTION TABLE
1789
1790  004666    001     002     004     ATABLE: .BYTE   1,2,4,10,20,40,100,200
1791  004671    010     020     040
1792  004674    100     200
1793
```

```
1794
1795
1796                                    ;*FLAGS AND INTERNAL PROGRAM CONTROL WORDS
1797
1798  004676  000010                    UNITS:  .BLKW   8.              ;THIS IS FILLED WITH -1
1799  004716  000000                    UNIT:   .WORD   0               ;UNIT UNDER TEST
1800  004720  000000                    NOUNIT: .WORD   0               ;NUMBER OF UNITS PRESENT
1801                                                                    ;USED TO KEEP TRACK OF UNIT UNDER TEST
1802  004722  000000                    NUNIT:  .WORD   0               ;USED TO DETERMIN IF THERE ARE MORE
1803                                                                    ;THAN ONE UNIT
1804  004724  000000                    NOPUSH: 0                       ;ALL ONES INDICATE NONE OF THE OPERATOR
1805                                                                    ;INTERVENTION TESTS WILL BE PERFORMED
1806  004726  000000                    SELECT: .WORD   0               ;ALL ONES INDICATE UNIT TO BE SELECTED
1807  004730  000000                    UNITSL: .WORD   0               ;UNIT NO. SELECTED
1808  004732  000000                    UBUSB:  0                       ;IF ZERO UNIBUS PRESENT
1809                                                                    ;IF ONES NO UNIBUS B
1810  004734  000000                    ERFLGS: 0                       ;ERROR FLAG
1811  004736  000000                    FIRST:  0                       ;IF ZERO WILL TYPE HEADER
1812                                                                    ;IF ONES WILL NOT TYPE HEADER
1813
1814  004740  000000                    ATTENT: 0                       ;ATTENTION BIT FOR PRESENT UNIT
1815  004742  000000                    TOTALAT:0                       ;TOTAL ATTENTION BITS
1816
1817  004744  000000                    RP06:   0                       ;RP06 DEVICE TYPE FLAG LOCATION
1818  004746  000000                    RP05:   0                       ;MEMOREX RP04 DEVICE TYPE FLAG
1819  004750  000000                    RH70:   0                       ;IF = 1, PROGRAM IS RUNNING ON RWP04 SYSTEM
1820                                                                    ;IF = 0, PROGRAM IS RUNNING ON RJP04
1821  004752  000000                    INUNIT: 0                       ;INITIAL UNIT NO. - USED DURING
1822                                                                    ;CHECKING ALL ADDRESS PLUG ADDRESSES
1823
1824
1825
1826
1827
1828  004754  000000                    TMP0:   .WORD   0               ;TEMP STORAGE
1829  004756  000000                    TMP1:   .WORD   0
1830  004760  000000                    TMP4:   .WORD   0               ;TEMP STORAGE
```

```
                                        .SBTTL
                                        .SBTTL  **DIAGNOSTIC CODE**
                                        .SBTTL


                                                .SBTTL  SETUP TESTS


004762  012737  177777  004724  BEGIN1: MOV     #-1,@#NOPUSH    ;JUMP OVER OPERATOR REQUIRED TESTS
004770  005037  004726          CLR     @#SELECT        ;DO NOT SELECT UNIT
004774  000412                  BR      START
004776  012737  177777  004726  BEGIN2: MOV     #-1,@#SELECT    ;SELECT UNIT
005004  005037  004724          CLR     @#NOPUSH        ;DO NOT JUMP OVER ANY TEST
005010  000404                  BR      START
005012  005037  004726  BEGIN:  CLR     @#SELECT        ;DO NOT SELECT UNIT
005016  005037  004724          CLR     @#NOPUSH        ;DO NOT JUMP OVER ANY OPERATOR
                                                        ;INTERVENTION TESTS - NORMAL RUN

005022                  START:
005022  000005                  RESET

005244  012737  000000  177776          MOV     #0,PS           ;SET PROCESSOR STATUS TO 0
005252  012737  000200  000036          MOV     #200,@#TRAPVEC+2;TRAP PRIORITY = 4
005260  013700  002266                  MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
005264  012720  044662                  MOV     #RPVECT,(R0)+   ;THIS IS FOR UNTIMELY INTERRUPTS
005270  012710  000340                  MOV     #340,(R0)       ;DRIVE INTERRUPT SERVICE ROUTINE
                                                        ;PRIORITY = 7

005274  004737  045662                  JSR     PC,@#STKINT     ;INITIALIZE THE TTY KEYBOARD
005300  005737  004736                  TST     @#FIRST         ;IS THIS FIRST TIME ROUND ?
005304  001001                          BNE     1$              ;DON'T GIVE HEADER IF NOT
005306  000402                          BR      2$              ;HEADER 1ST TIME THROUGH
005310  000137  006110          1$:     JMP     @#SND1          ;NO HEADER

005314                  2$:


006110  012737  177777  004736  SND1:   MOV     #-1,@#FIRST     ;NEXT TIME DO NOT GIVE HEADER


006146  032777  010000  172764  RH70CK: BIT     #SW12,@SWR      ;LOOK TO SEE IF USING RH70
006154  001403                          BEQ     3$              ;IF SW12 = 0, SKIP NEXT
006156  012737  000001  004750          MOV     #1,@#RH70       ;IF SW12 = 1, CU IS AN RH70
006164                  3$:
                                        ;*IS THERE A P-CLOCK (KW11-P) ON THE SYSTEM
                                        ;*IF SO MAKE 'WAT' TRAPS GO TO 'WAIT.P'
                                        ;*IF SO MAKE RP04 INTERRUPTS GO TO 'TIME 1'
                                        ;*IF NOT MAKE 'WAT' TRAPS GO TO 'WAIT.T'
                                        ;*IF NOT MAKE RP04 INTERRUPTS GO TO 'TIME 2'

                                        ;*THE NEXT LINE IS TO BE ADDED LATER
                                        ;*AND THE JUMP AND NOP REMOVED
                                        ;*FOR NOW NO CLOCK WILL BE USED
```

```
1887
1888                                    ;       MOV     @#1$,@#ERRVEC   ;SET TIME-OUT VECTOR
1889                                    ;
1890                                    ;       JMP     @#1$            ;DO NOT USE CLOCK
1891                                    ;       NOP
1892                                    ;       TST     @#PCLCSR        ;REFERENCE P-CLOCK STATUS REGISTER
1893                                    ;                               ;ADDRESS = 172540
1894                                    ;       MOV     #WAIT.P,@#STRPAD+20 ;THERE IS A P-CLOCK
1895                                    ;       MOV     #TIME1,@#RP4VEC ;THERE IS A P CLOCK SO
1896                                    ;                               ;VECTOR TO TIME1
1897                                    ;       BR      2$
1898                                    ;1$:    MOV     #WAIT.T,@#STRPAD+20 ;THERE IS NO P-CLOCK
1899
1900
1901
1902  006164  012737  041664  004606          MOV     #TIME2,@#RP4VEC ;RP04/5/6 INTERRUPTS GO TO 'TIME 2'
1903  006172  012737  177777  047270  2$:     MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
1904
1905
1906
1907
1908  006200  005737  004726                  TST     @#SELECT        ;WAS IT A 200 START
1909  006264  104412                          RDOCT
1910  006266  042716  177770                  BIC     #177770,(SP)    ;ONLY KEEP LAST 3 BITS
1911  006272  011637  004716                  MOV     (SP),@#UNIT     ;SAVE UNIT TO BE TESTED
1912  006276  012637  004730                  MOV     (SP)+,@#UNITSL  ;SAVE UNIT TO BE TESTED
1913
1914
1915
1916  006304  013737  004730  004716          MOV     @#UNITSL,@#UNIT ;SET UNIT NUMBER
1917
```

```
1918
1919  006330  012706  001000                  MOV     #STACK, SP      ;SET UP STACK POINTER
1920  006334  012737  047132  000030          MOV     #REGSA1,@#EMTVEC;ERROR VECTOR SO THAT
1921                                                                  ;NO REGISTERS ARE SAVED
1922  006342  012737  006370  000004          MOV     #2$,@#ERRVEC    ;SET UP FOR BUS TIMEOUT
1923
1924  006350  012700  000024                  MOV     #24,R0          ;THERE ARE 24 REG TO TEST
1925  006354  012701  002270                  MOV     #RHDB,R1        ;R1 NOW HAS ADDR OF ADDR OF FIRST REG.
1926  006360  013102                  1$:     MOV     @(R1)+,R2       ;READ HARDWARE REG.
1927  006362  005300                          DEC     R0              ;COUNT DOWN
1928  006364  001375                          BNE     1$              ;BRANCH IF 24 NOT DONE
1929  006366  000454                          BR      3$              ;BRANCH IF 24 DONE
1930  006370  012737  000006  000004  2$:     MOV     #ERRVEC+2,@#ERRVEC ;RESTORE TRAP CATCHER
1931  006376  022626                          CMP     (SP)+,(SP)+     ;CLEAN STACK
1932  006400  016137  177776  001200          MOV     -2(R1),$TMP1    ;STORE FAILING REG ADDR
1933  006406  104007                          ERROR   7               ;REGISTER NON EXISTANT
1934  006410  032777  020000  172522          BIT     #SW13,@SWR      ;INHIBIT ERROR PRINTOUT ?
1935  006416  001036                          BNE     4$              ;BRANCH IF YES
1936
1937
1938  006504  012746  000204                  MOV     #ADDMOD,-(SP)   ;GET READY TO TYPE STARTING ADDRESS
1939                                                                  ;OF "CHANGE OF BASE ADDRESS" ROUTINE
1940  006510  104402                          TYPOC
1941  006512  000000                          HALT                    ;STOP TO FORCE THE CORRECT RESTART
1942  006514  000137  041000          4$:     JMP     @#$EOP          ;GO TO END OF PROGRAM ------------>
1943
1944  006520  012737  006574  000004  3$:     MOV     #TRP,@#4        ;INITIALIZE VECTOR
1945  006526  005737  C02340                  TST     @#RHBAE         ;ADDRESS RPBAE(RH11/RH70?)
1946  006532  005237  004750                  INC     @#RH70          ;FOUND AN RH70-SET MASK
1947  006572  000420                          BR      RTN             ;AND GET OUT
1948  006574  005726                  TRP:    TST     (SP)+
1949  006576  005726                          TST     (SP)+           ;RESTORE THE STACK
1950  006634  012737  047122  000030  RTN:    MOV     #$ERROR,@#EMTVEC;RESTORE ERROR VECTOR
1951                                                                  ;SO THAT REGISTERS ARE SAVED
1952  006642  012737  000006  000004          MOV     #ERRVEC+2,@#ERRVEC ;RESTORE TRAP CATCHER
1953
```

```
1954
1955
1956  006660  012706  001000                  MOV     #STACK,SP       ;SET STACK POINTER
1957
1958  006672  013701  002316                  MOV     @#RHAS,R1       ;R1 HAS ADDRESS OF RHAS
1959  006676  012711  177777                  MOV     #-1,@R1         ;THIS WRITES ALL ONES INTO RHAS
1960  006702  105711                          TSTB    @R1             ;TEST RHAS FOR ALL 0'S
1961  006706  011137  001126                  MOV     @R1,@#$BDDAT    ;BAD DATA
1962  006712  005037  001124                  CLR     @#$GDDAT        ;GOOD DATA
1963  006716  010137  004600                  MOV     R1,@#REGADR     ;FAILING REG. (RHAS)
1964
1965  006722  104005                          ERROR   5               ;RHAS DOES NOT CLEAR BY WRITING
1966                                                                  ;ALL ONES INTO IT
1967
```

```
1968
1969  006742  000005                          RESET                   ;START WITH AN INIT
1970  006744  004737  045662                  JSR     PC,@#$TKINT     ;INITIALIZE TTY KEYBOARD
1971
1972  006750  032777  020000  172162          BIT     #SW13,@SWR      ;INHIBIT ERROR TYPEOUT?
1973  006756  001026                          BNE     4$              ;BRANCH IF YES
1974  007034  013701  002316          4$:     MOV     @#RHAS,R1       ;R1 HAS ADDR. OF RHAS
1975  007040  013702  002276                  MOV     @#RHCS2,R2      ;R2 HAS ADDR. OF RHCS2
1976  007044  005012                          CLR     @R2             ;CLEAR RHCS2
1977  007046  012700  000010                  MOV     #8.,R0          ;COUNT
1978  007052  013704  002302                  MOV     @#RHER1,R4      ;R4 HAS ADDR. OF RHER1
1979  007056  012714  177777          1$:     MOV     #-1,@R4         ;MOVE ERRORS INTO RHER1
1980  007062  005212                          INC     @R2             ;INCREMENT UNIT NO.
1981  007064  005300                          DEC     R0              ;COUNT DOWN DRIVE COUNTER
1982  007066  001373                          BNE     1$              ;DO NEXT UNIT IF 8 NOT DONE
1983
1984  007070  111137  004742                  MOVB    @R1,@#TOTALAT   ;SAVE TOTAL ATTENTION
1985                                                                  ;USED IN DRIVE CLEAR TEST
1986  007074  105037  004743                  CLRB    @#TOTALAT+1     ;CLEAR UPPER BYTE
1987  007100  105711                          TSTB    @R1             ;TEST 'RHAS' FOR ANY DRIVES PRESENT
1988  007102  001402                          BEQ     2$              ;NONE RESPONDING - TYPE THE MESSAGE
1989  007104  000137  007456                  JMP     XE2             ;SOME THERE - GO FILL "UNITS" TABLE
1990
1991  007110  032777  020000  172022  2$:     BIT     #SW13,@SWR      ;INHIBIT ERROR TYPE OUT?
1992  007116  001402                          BEQ     3$              ;"NO DRIVES" MESSAGE IF NOT
1993  007120  000137  010014                  JMP     SELTST          ;CHECK FOR SELECTED UNIT START AND LOAD
1994                                                                  ;"UNITS" TABLE WITH SELECTED ONE IF SO
1995
1996  007124                          3$:
1997
1998  007452  000137  041000                  JMP     @#$EOP          ;GO OUT------------------------------>
1999
2000
2001
2002                                          ;*SET UP UNITS TABLE
2003
2004  007456                          XE2:
2005  007456  012700  000010          2$:     MOV     #8.,R0          ;COUNTER
2006  007462  012703  004676                  MOV     #UNITS,R3       ;POINTER
2007  007466  012723  177777          3$:     MOV     #-1,(R3)+       ;PRESET BLOCK TO ALL ONES
2008  007472  005300                          DEC     R0              ;COUNT
2009  007474  001374                          BNE     3$              ;BRANCH IF 8 NOT DONE
2010  007476  012703  004676                  MOV     #UNITS,R3       ;POINTER
2011  007502  005005                          CLR     R5              ;INITIALIZE UNIT NO. TO 0
2012  007504  005037  004720                  CLR     @#NOUNIT        ;NO. OF UNITS PRESENT
2013  007510  012700  000010                  MOV     #8.,R0          ;COUNTER
2014  007514  011137  001176                  MOV     @R1,@#$TMP0     ;TEMPORARY STORAGE
2015  007520  006037  001176          4$:     ROR     @#$TMP0         ;SET CARRY IF ONE IN  0 BIT
2016  007524  103120                          BCC     5$              ;CHECK NEXT UNIT IF ONE NOT IN BIT 0
2017
2018  007526  010577  172544                  MOV     R5,@RHCS2       ;INSERT UNIT NUMBER INTO RHCS2 UA BITS
2019  007532  022777  024020  172564          CMP     #24020,@RHDT    ;IS THIS A DUAL PORT RP04 ?
2020  007540  001503                          BEQ     6$              ;TYPE THE UNIT NO. IF YES
2021  007542  022777  020020  172554          CMP     #20020,@RHDT    ;IS THIS A SINGLE PORT RP04 ?
2022  007550  001477                          BEQ     6$              ;TYPE UNIT NO. IF YES
2023
```

```
                                                    CMP     #24021,@RHDT    ;DUAL PORT RP05 ?
                                                    BEQ     6$              ;TYPE UNIT NO. IF SO
                                                    CMP     #20021,@RHDT    ;SINGLE PORT RP05 ?
                                                    BEQ     6$              ;TYPE UNIT NO. IF SO

                                                    CMP     #24022,@RHDT    ;IS THIS A DUAL PORT RP06 ?
                                                    BEQ     6$              ;TYPE THE UNIT NO. IF SO
                                                    CMP     #20022,@RHDT    ;IS THIS A SINGLE PORT RP06 ?
                                                    BEQ     6$              ;TYPE UNIT NO. IF SO


                                                    ;*NO...IT'S NOT AN RP04/RP05/RP06 DEVICE SO TYPE
                                                    ;*OUT THE DEVICE TYPE

                                                    MOV     R5,-(SP)        ;GET READY TO TYPE UNIT NUMBER
                                                    TYPDS
                                                    MOV     @RHDT,-(SP)     ;GET READY TO TYPE RHDT
                                                    TYPOC
                                                    BR      5$              ;NO RP04/RP05/RP06 FOUND SO TEST NEXT UNIT

                                            6$:     MOV     R5,(R3)+        ;LOAD TABLE POSITION AND INCR IT
                                                    TYPE    ,$CRLF
                                                    MOV     R5,-(SP)        ;PUT DRIVE NO. ON STACK
                                                    TYPDS                   ;TYPE DRIVE NO.
                                                    INC     @#NOUNIT        ;INCR THE TOTAL NO. OF UNITS

                                            5$:     INC     R5              ;'RHCS2' UNIT ADDRESS
                                                    DEC     R0              ;DRIVE COUNTER DOWN ONE
                                                    BNE     4$              ;TEST AND DO NEXT UNIT IF 8 NOT DONE

                                                    MOV     @#UNITS,@#UNIT  ;SET UNIT NO. TO FIRST ONE FOUND OR 0
                                                    MOV     @#NOUNIT,@#NUNIT;SAVE NO. OF UNITS
                                                    DEC     @#NUNIT         ;IF NUNIT = 0 THEN ONLY ONE UNIT
                                                                            ;IF NUNIT > 0 THEN MORE THAN ONE UNIT

                                            SELTST: TST     @#SELECT        ;STARTING ADDRESS 200
                                                    MOV     @#UNITSL,@#UNIT ;CHANGE UNIT NUMBER TO SELECTED ONE
```

```
2061
2062
2063
2064  010064  005037  004740                CLR     @#ATTENT        ;CLEAR
2065
2066  010070  005737  004716                TST     @#UNIT          ;IS "UNIT 0" NEXT IN THE UNITS TABLE ?
2067  010074  001107                        BNE     9$              ;IF NOT TEST THIS UNIT
2068  010076  012700  000041                MOV     #41,R0          ;IF SO, CHECK THE LOAD MEDIA LOCATION
2069  010102  122710  000011                CMPB    #11,(R0)        ;WAS IT AN RP04/5/6 ?
2070  010106  001102                        BNE     9$              ;NO - GO AHEAD WITH TESTING UNIT #0
2071
2072  010110  005737  004726                TST     @#SELECT        ;WAS UNIT #0 SELECTED ?
2073                                                                ;(IE. WAS IT A 210 START ?)
2074  010114  001006                        BNE     12$             ;IF SO, CHANGE PACKS
2075
2076                                        ;*INCREMENT THE UNITS TABLE TO NEXT DRIVE (IF ANY)
2077                                        ;*& DECREMENT THE "NOUNITS" PRESENT (TO BE TESTED)
2078
2079  010116  012700  004676                MOV     #UNITS,R0       ;LOAD UNITS TABLE POINTER
2080  010122  005720                        TST     (R0)+           ;SELECT THE NEXT UNIT IN THE TABLE
2081                                                                ;(DOUBLE INCREMENT THE POINTER, R0)
2082  010124  022710  177777                CMP     #-1,(R0)        ;IS THERE ANOTHER TABLE ENTRY PRESENT ?
2083  010130  001065                        BNE     11$             ;IF SO, USE THE NEXT DRIVE & DEC "NOUNITS"
2084                                                                ;IF NOT, MUST USE DRIVE #0 & CHANGE PACK
2085  010132                        12$:
2086  010300  000000                        HALT
2087  010302  000404                        BR      9$              ;CONTINUE, USING SCRATCH PACK ON UNIT #0
2088
2089  010304  011037  004716        11$:    MOV     (R0),@#UNIT     ;SET UP TO BE THE UNIT UNDER TEST
2090  010310  005337  004720                DEC     @#NOUNITS       ;DECREMENT BECAUSE UNIT #0 WON'T BE TESTED
2091
2092  010314  013700  004716        9$:     MOV     @#UNIT,R0       ;R0 CONTAINS UNIT UNDER TEST
2093
2094
2095
2096  010320  005037  004744                CLR     @#RP06          ;CLEAR RP06 DEVICE TYPE FLAG
2097  010324  010077  171746                MOV     R0,@RHCS2       ;SET UP UNIT ADDRESSING
2098  010330  022777  024022  171766        CMP     #24022,@RHDT    ;IS IT A DUAL PORT RP06 ?
2099  010336  001405                        BEQ     2$              ;YES...SET THE FLAG
2100  010340  022777  020022  171756        CMP     #20022,@RHDT    ;IS IT A SINGLE PORT RP06 ?
2101  010346  001401                        BEQ     2$              ;YES...SET FLAG
2102  010350  000404                        BR      3$              ;DON'T SET FLAG - CHECK FOR RP04
2103  010352  012737  177777  004744  2$:   MOV     #-1,@#RP06      ;SET THE FLAG
2104  010360  000416                        BR      8$              ;DON'T CHECK FOR RP04, IT WAS RP06
2105
2106  010362  005037  004746          3$:   CLR     @#RP05          ;CLEAR MEMOREX RP04 DEVICE FLAG
2107  010366  022777  024021  171730        CMP     #24021,@RHDT    ;IS IT A DUAL PORT MEMOREX RP04 ?
2108  010374  001405                        BEQ     7$              ;YES..SET THE FLAG FOR ADDR PLUG TESTS
2109  010376  022777  020021  171720        CMP     #20021,@RHDT    ;IS IT A SINGLE PORT MEMOREX RP04 ?
2110  010404  001401                        BEQ     7$              ;YES..SET THE FLAG FOR ADDR PLUG TESTS
2111  010406  000403                        BR      8$              ;DON'T SET FLAG - NOT MEMOREX DRIVE
2112  010410  012737  177777  004746  7$:   MOV     #-1,@#RP05      ;SET THE FLAG
2113  010416                          8$:                           ;ASSUME THE NEXT UNIT IS AN RP04
2114
2115
2116
```

```
  2117  010416  116037  004666  004740          MOVB    ATABLE(R0),@#ATTENT ;SET APPROPRIATE ATTENTION BIT
  2118  010462  013746  004716                  MOV     @#UNIT,-(SP)    ;UNIT NO. TO STACK
  2119  010466  104405                          TYPDS                   ;TYPE  DRIVE NO.
  2120
```

```
                                                ;TYPE OUT THE DRIVE TYPE IN ASCII
                                                
010560  022777  024020  171536        CMP     #24020,@RHDT    ;DUAL PORT RP04 ?
010566  001425                        BEQ     4$              ;TYPE ASCII MSG OUT
010570  022777  020020  171526        CMP     #20020,@RHDT    ;SINGLE PORT RP04 ?
010576  001421                        BEQ     4$              ;TYPE THE MESSAGE
                                                
010600  022777  024021  171516        CMP     #24021,@RHDT    ;DUAL PORT RP05 ?
010606  001453                        BEQ     6$              ;TYPE THE MESSAGE
010610  022777  020021  171506        CMP     #20021,@RHDT    ;SINGLE PORT RP05 ?
010616  001447                        BEQ     6$              ;TYPE THE MESSAGE
                                                
010620  022777  024022  171476        CMP     #24022,@RHDT    ;DUAL PORT RP06 ?
010626  001424                        BEQ     5$              ;TYPE THE MESSAGE
010630  022777  020022  171466        CMP     #20022,@RHDT    ;SINGLE PORT RP06 ?
010636  001420                        BEQ     5$              ;TYPE THE MESSAGE
010640  000454                        BR      1$              ;DRIVE IS NOT AN RP04/RP05/RP06
                                                            ;DON'T TYPE ASCII MESSAGE OUT
                                                
                                                ;-SHOULD NEVER HAPPEN AT THIS POINT
                                                ;UNLESS DRIVE GOT SICK WHILE TESTING
                                                ;WAS IN PROGRESS
                                                
010642                          4$:
010676  000435                        BR      1$              ;SKIP NEXT MESSAGE
010700                          5$:
010734  000416                        BR      1$              ;SKIP NEXT MESSAGE
010736                          6$:
                                                
                                                
                                                
010772  005777  171330          1$:   TST     @RHSN           ;READ SERIAL NO. AND DRIVE TYPE
010776  005777  171322                TST     @RHDT           ;THESE TWO ARE TO HELP SCOPE LOOPS
011002  017737  171320  002410        MOV     @RHSN,@#SN      ;SAVE TO CHECK IF DRIVE CLEAR CLEARS ANY BITS
011010  017737  171310  002406        MOV     @RHDT,@#DT      ;SAVE TO CHECK IF DRIVE CLEAR CLEARS ANY BITS
```

```
2158
2159
2160
2161
2162  011026  004737  041452          JSR     PC,@#CLDISK     ;GIVE INITIALIZE
2163  011032  032713  010000          BIT     #MOL,@R3        ;CHECK MOL IN RHDS1
2164  011254  032713  010000    1$:   BIT     #MOL,@R3        ;CHECK MOL IN RHDS1
2165  011260  001775                  BEQ     1$              ;BRANCH IF MOL IS HIGH
```

```
2166
2167
2168  011372  012706  001000                MOV     #STACK,SP       ;RESET STACK
2169
2170  011402  013700  002266                MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
2171  011406  012720  011454                MOV     #RPTRP1,(R0)+   ;THIS IS FOR TIMELY INTERRUPTS
2172  011412  012710  000340                MOV     #340,(R0)       ;RP04 INTERRUPT SERVICE ROUTINE
2173                                                                 ;PRIORITY = 7
2174  011416  012737  000200  177776        MOV     #200,PS         ;SET PROCESSOR PRIORITY @ 4 (DISK @ 5)
2175  011424  012711  000300                MOV     #RDY!IE,@R1     ;RDY, IE IN RHSC1 SHOULD CAUSE INTERRUPT
2176
2177  011430  013737  042150  001200        MOV     @#TIMCNT,@#$TMP1;COUNTER
2178  011436  005337  001200        1$:     DEC     @#$TMP1         ;WAIT FOR INTERRUPT
2179  011442  001375                BNE     1$              ;BRANCH IF NOT ZERO
2180                                                                 ;BEFORE THIS IS ZERO INTERRUPT SHOULD OCCUR
2181  011444  104065                ERROR   65              ;INTERRUPT DID NOT OCCUR
2182  011446  012712  000040                MOV     #40,@R2         ;CLEAR CONTROLLER VIA CS2 CLR BIT
2183
2184  011454  022626                RPTRP1: CMP     (SP)+,(SP)+     ;RESTORE STACK
2185  011456  022711  004200                CMP     #DVA!RDY,@R1    ;IE SHOULD BE LOW AS RUPT OCCURRED
2186  011464  104065                ERROR   65              ;INTERRUPT OCCURED BUT
2187                                                                 ;IE FAILED TO RESET
2188  011466  012712  000040                MOV     #40,@R2         ;CLEAR CONTROLLER VIA CS2 CLR BIT
```

```
2189
2190
2191
2192  011502  012706  001000                  MOV     #STACK,SP       ;RESET STACK
2193
2194  011512  013700  002266                  MOV     @#RPVEC,R0      ;GET RP VECTOR ADDRESS
2195  011516  012720  011562                  MOV     #RPTRP2,(R0)+   ;THIS IS FOR UNTIMELY INTERRUPTS
2196  011522  012710  000340                  MOV     #340,(R0)       ;RP04 INTERRUPT SERVICE ROUTINE
2197                                                                  ;PRIORITY = 7
2198  011526  012737  000240  177776          MOV     #240,PS         ;SET PROCESSOR PRIORITY @ 5
2199  011534  012711  000300                  MOV     #RDY!IE,@R1     ;RDY, IE IN RHSC1 WHOULD CAUSE INTERRUPT
2200
2201  011540  013737  042150  001200          MOV     @#TIMCNT,@#$TMP1;COUNTER
2202  011546  005337  001200          1$:     DEC     @#$TMP1         ;WAIT FOR INTERRUPT
2203  011552  001375                  BNE     1$                      ;BRANCH IF NOT ZERO
2204                                                                  ;BEFORE THIS IS ZERO INTERRUPT WHOULD
2205                                                                  ;OCCUR
2206  011554  012712  000040                  MOV     #40,@R2         ;CLEAR THE CONTROLLER VIA CS2 CLR BIT
2207
2208  011562  022626          RPTRP2: CMP     (SP)+,(SP)+     ;RESTORE STACK
2209  011564  104065                  ERROR   65              ;INTERRUPT OCCURRED WITH PROCESSOR
2210                                                          ;PROCESSOR STATUS SAME AS DISK
2211  011566  012712  000040                  MOV     #40,@R2         ;CLEAR THE CONTROLLER VIA CS2 CLR BIT
2212
2213
2214
2215
2216
```

```
2217  011612  012737  000000  177776          MOV     #0,PS           ;SET PROCESSER STATUS TO 0
2218
2219  011674  013746  004716                  MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
2220  011700  104405                          TYPDS
2221  011736  000000                          HALT
2222  011740  032713  010000                  BIT     #MOL,@R3        ;MOL IN RHDS1 SHOULD BE = 0
2223  011744  001403                          BEQ     3$              ;BRANCH IF MOL=0
2224  011746  010337  001122                  MOV     R3,@#$BDADR     ;FAILING REGISTER ADDRESS-RHDS1
2225  011752  104010                          ERROR   10              ;ON SPINPLE POWERED DOWN
2226                                                                  ;MOL SHOULD BE 0
2227  011754  013746  004716          3$:     MOV     @#UNIT,-(SP)    ;UNIT NUMBER
2228  011760  052716  000100                  BIS     #IR,(SP)        ;INCLUDE IR
2229  011764  022612                          CMP     (SP)+,@R2       ;ONLY UNIT NO. AND IR SHOULD BE
2230                                                                  ;HIGH IN RHCS2
2231  011766  001403                          BEQ     4$              ;BRANCH IF RHCS2 GOOD
2232  011770  010237  001122                  MOV     R2,@#$BDADR     ;FAILING REGISTER ADDRESS-RHCS2
2233  011774  104011                          ERROR   11              ;WITH SPINDLE POWERED DOWN
2234                                                                  ;ONLY UNIT NO. AND IR SHOULD BE
2235                                                                  ;HIGH
2236
2237  011776                          4$:
2238  012026  013746  004716                  MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
2239  012032  104405                          TYPDS
2240  012102  000000                          HALT
2241  012104  032713  010000                  BIT     #MOL,@R3        ;MOL IN RHDS1 SHOULD BE = 1
2242  012110  001411                          BEQ     5$              ;BRANCH IF MOL = 0
2243  012112  032713  000400                  BIT     #DPR,@R3        ;DPR IN RHDS1 SHOULD BE = 1
2244  012116  001406                          BEQ     5$              ;BRANCH IF DPR = 0
2245  012120  032713  000200                  BIT     #DRY,@R3        ;DRY IN RHDS1 SHOULD BE = 1
2246  012124  001403                          BEQ     5$              ;BRANCH IF DRY =0
2247  012126  032713  000100                  BIT     #VV,@R3         ;VV IN RHDS1 SHOULD BE = 0
2248  012132  001403                          BEQ     6$              ;BRANCH IF VV = 0 (GOOD)
2249  012134  010337  001122          5$:     MOV     R3,@#$BDADR     ;FAILING REGISTER ADDRESS - RHDS1
2250  012140  104012                          ERROR   12              ;WITH SPINDLE POWERED UP
2251                                                                  ;RHDS1 SHOULD HAVE VV = 0, MOL = 1
2252
2253  012142  011100                  6$:     MOV     @R1,R0          ;GET RHCS1 CONTENTS
2254  012144  042700  160076                  BIC     #SC!TRE!MCPE!76,R0 ;CLEAR SC,TRE,MCPE AND
2255                                                                  ;ALL FUNCTION BITS
2256  012150  022700  004200                  CMP     #DVA!RDY,R0     ;RHCS1 SHOULD HAVE
2257                                                                  ;GO = 0, DVA = 1, RDY = 1
2258  012154  001403                          BEQ     7$              ;BRANCH IF RHCS1 IS GOOD
2259  012156  010137  001122                  MOV     R1,@#$BDADR     ;FAILING REGISTER RHCS1
2260  012162  104013                          ERROR   13              ;AFTER A POWER UP WITHOUT ANY
2261                                                                  ;INIT RHCS1 SHOULD HAVE
2262                                                                  ;GO = 0, DVA = 1, RDY = 1, IE = 0
2263                                                                  ;DISREGARD ALL OTHER BITS
2264
2265  012164  005777  170144          7$:     TST     @RHCC           ;TEST RHCC, IT SHOULD = 0
2266  012170  001411                          BEQ     10$             ;BRANCH IF RHCC = 0
2267  012172  013737  002334  004600          MOV     @#RHCC,@#REGADR ;FAILING REGISTER RHCC
2268  012200  005037  001124                  CLR     @#$GDDAT        ;RHCC SHOULD BE = 0
2269  012204  017737  170124  001126          MOV     @RHCC,@#$BDDAT  ;BAD DATA
2270  012212  104014                          ERROR   14              ;AFTER POWER UP RHCC
2271                                                                  ;SHOULD BE 0
2272
```

```
2273  012214                          10$:
2274
2275                                          ;*NOW SAVE REGISTERS FOR COMPARISON AFTER PACK ACKNOWLEDGE
2276
2277
2278
2279
2280
2281
2282                                          ;*NOW COMPARE REGISTERS SO THAT NO REGISTERS
2283                                          ;*CHANGED EXCEPT VV IN RHDS1 AND IE IN RHCS1
2284
2285
2286  012340  104015                  11$:    ERROR   15              ;GIVING A PACK ACKNOWLEDGE
2287  012342  000207                          RTS     PC              ;CAUSED AN ERROR
2288                                                                  ;PACK ACKNOWLEDGE SHOULD
2289                                                                  ;SET VV IN RHDS1
2290                                                                  ;INTERRUPT SHOULD MAKE
2291                                                                  ;IE = 0
2292                                                                  ;NO OTHER REGISTERS SHOULD
2293                                                                  ;CHANGE
2294                                                                  ;GOOD DATA GIVES
2295                                                                  ;CONTENTS OF REGISTER BEFORE
2296                                                                  ;PACK ACKNOWLEDGE
2297                                                                  ;RECEIVED DATA GIVES
2298                                                                  ;CONTENTS OF REGISTER
2299                                                                  ;AFTER PACK ACKNOWLEDGE
2300
2301  012344  012737  177777  047270  12$:    MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
2302
```

```
                                        ;*IN CASE THERE IS ANY DRIVE ERRORS DURING POWER UP
                                        ;*OR POWER DOWN OR ANY PARITY ERRORS A RESET IS GIVEN
                                        RESET
012354  000005
012356  004737  045662                  JSR     PC,@#$TKINT     ;INITILIZE TK


                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER PACK ACKNOWLEDGE




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


012604  104015                  1$:     ERROR   15              ;GIVING A PACK ACKNOWLEDGE
012606  000207                          RTS     PC              ;CAUSED AN ERROR
                                                                ;PACK ACKNOWLEDGE SHOULD
                                                                ;SET VV IN RHDS1
                                                                ;INTERRUPT SHOULD MAKE
                                                                ;IE = 0
                                                                ;NO OTHER REGISTERS SHOULD
                                                                ;CHANGE
                                                                ;GOOD DATA GIVES CONTENTS
                                                                ;OF REGISTER BEFORE COMMAND
                                                                ;RECEIVED DATA GIVES CONTENTS
                                                                ;OF REGISTER AFTER COMMAND
012610                          2$:
```

```
2338
2339  012610  005737  004744          TST     @#RP06          ;TEST FOR RP06 DRIVE
2340  012614  001005                  BNE     3$              ;IF = 1 DO "MAKECYL" 777
2341  012616  005737  004746          TST     @#RP05          ;TEST FOR RP05 DRIVE
2342  012622  001004                  BNE     4$              ;IF = 1 DO "MAKECYL" 377
2343                                                          ;OR THE ADDRESS PLUG TESTS
2344  012630                      3$:
2345  012634                      4$:
2346
2347
2348
2349
2350
2351
```

```
                                                TST     @#RP06          ;TEST FOR RP06 DRIVE
                                                BNE     4$              ;IF = 1, DO THIS TEST
                                                TST     @#RP05          ;TEST FOR MEMOREX RP04
                                                BNE     4$              ;IF = 1, DO THIS TEST
                                                                        ;IF NEITHER FLAG IS UP, ASSUME THE
                                                                        ;DRIVE IS AN ISS RP04 AND SKIP TEST
                                        4$:
                                                ;*SET DIAGNOSTIC MODE TO ENABLE A COMMAND ACTIVE WHILE
                                                ;*THE PLUG IS PULLED

                                                BIS     #DMD,@RHMR      ;SET UP DIAGNOSTIC MODE

                                                ;*TAKE AN INITIAL REGISTER SNAPSHOT


                                                ;*ISSUE A COMMAND AND THE 'GO' BIT (NOT POSITIONING COMMAND)
                                                ;*TO VERIFY COMMAND ABORT IF PLUG IS PULLED


                                                ;*ISSUE SOME CLOCKS TO GET THE COMMAND STARTED
                                                ;*(USE "SEARCH" WITH "DTETST" FLAG UP TO STOP CLOCKING ?)


                                                MOV     @#UNIT,-(SP)    ;GET THE UNIT NO. UNDER TEST
                                                TYPOS                   ;TYPE IT OUT
                                                HALT                    ;WAIT FOR OPERATOR PLUG CHANGE

                                                ;*CHECK THAT THE UNIT NO. UNDER TEST HAS GONE OFFLINE

                                                MOV     @RHDST,R0       ;ATTEMPT TO ADDRESS THE DRIVE
                                                JSR     PC,@#PUTREG     ;TAKE REGISTER SNAPSHOTS
                                                BIT     #NED,@#CS2      ;TEST FOR NON EXISTENT DRIVE
                                                BNE     7$              ;CONTINUE IF 'NED' BIT SET (UNIT
                                                                        ;IS OFFLINE AS IT SHOULD BE)
                                                ERROR   101             ;UNIT DID NOT GO OFFLINE WHEN ADDRESS
                                                                        ;PLUG WAS REMOVED
                                        7$:
```

```
2397
2398                                            ;*NOW REPLACE THE ADDRESS PLUG
2399
2400
2401  013266  013746  004716                    MOV     @#UNIT,-(SP)    ;GET THE UNIT UNDER TEST
2402  013272  104405                            TYPOS                   ;TYPE IT OUT
2403  013336  000000                            HALT                    ;WAIT FOR OPERATOR PLUG REPLACEMENT
2404
2405                                            ;*CHECK THAT THE ORIGINAL UNIT HAS COME BACK ON LINE
2406
2407  013340  004737  043364                    JSR     PC,@#PUTREG     ;TAKE NEW REGISTER SNAPSHOTS
2408  013344  032737  000400  002404            BIT     #DPR,@#DS1      ;TEST THAT 'DPR' = 1
2409  013352  001411                            BEQ     9$              ;ERROR - DRIVE SHOULD BE PRESENT
2410  013354  032737  004000  002362            BIT     #DVA,@#CS1      ;TEST THAT DEVICE IS NOW AVAILABLE
2411  013362  001405                            BEQ     9$              ;ERROR - 'DVA' SHOULD = 1
2412  013364  032737  000200  002404            BIT     #DRY,@#DS1      ;TEST THAT DRIVE READY IS = 1
2413  013372  001401                            BEQ     9$              ;ERROR - 'DRY' SHOULD = 1
2414  013374  000403                            BR      8$              ;A-OK: 'DPR' = 1, 'DVA' = 1, & 'DRY' =1
2415  013376  104102                    9$:     ERROR   102             ;UNIT NOT AVAILABLE AFTER
2416                                                                    ;ADDRESS PLUG WAS REPLACED
2417
2418  013404                            8$:     ;CHANGE THE INITIAL REGISTER SNAPSHOT TO EXPECTED VALUES
2419                                            ;*AFTER THE PLUG CHANGE
2420
2421
2422  013510  053737  004740  004636            BIS     @#ATTENT,@#SAVERE+24 ;SET UNIT UNDER TEST ATTENTION
2423  013516  005037  004654                    CLR     @#SAVERE+42     ;SET RHCC REGISTER IMAGE TO ALL 0'S
2424
2425                                            ;*TAKE A NEW REGISTER SNAPSHOT AND
2426                                            ;*COMPARE THE REGISTER CONTENTS WITH EXPECTED VALUES
2427
2428
2429  013540  104100                    5$:     ERROR   100             ;ADDRESS PLUG CHANGE CAUSED SOME
2430                                                                    ;REGISTER ERROR
2431  013542  000207                            RTS     PC              ;GO BACK AND CHECK THE NEXT REGISTER
2432                                                                    ;UNTIL ALL 18. HAVE BEEN DONE
```

```
2433
2434
2435
2436
2437  013544                                  65$:
2438
2439                                          ;*CHANGE THE REGISTER SNAPSHOT TO EXPECTED VALUES AFTER THE CLEAR
2440
2441
2442  013660  043737  004740  004636          BIC     @#ATTENT,@#SAVERE+24 ;UNIT UNDER TEST ATTENTION BIT
2443
2444
2445                                          ;*TAKE ANOTHER REGISTER SNAPSHOT AND COMPARE RESULTS
2446                                          ;*WITH THE EXPECTED VALUES
2447
2448
2449  013704  104100                  105$:   ERROR   100             ;ADDRESS PLUG CHANGE CAUSED SOME
2450  013706  000207                          RTS     PC              ;INCORRECT REGISTER RESULT
2451
2452  013710                          115$:
2453
2454                                          ;*(USE NED METHOD TO VERIFY
2455                                          ;*THAT ATTENTION BIT COMES UP IN THE PROPER LOCATION ??)
2456
2457
2458
2459
2460                                          ;*SET 'VV' IN RHDS1 AFTER RESET FROM THE RECALIBRATE
2461                                          ;*CAUSED BY PULLING THE PLUG
2462
2463
```

```
2464
2465
2466
2467
2468  013752  005737  004744          TST     @#RP06          ;TEST FOR RP06 DRIVE
2469  013756  001005                  BNE     4$              ;IF = 1, OK TO DO THIS TEST
2470  013760  005737  004746          TST     @#RP05          ;TEST FOR MEMOREX RP04
2471  013764  001002                  BNE     4$              ;IF = 1, OK TO DO THIS TEST
2472                                                          ;NOT AN RP06 OR MEMOREX RP04 SO
2473                                                          ;ASSUME AN ISS RP04 AND SKIP TEST
2474  013772                      4$:
2475
2476                                  ;*CHECK TO SEE IF THIS TEST HAS BEEN SELECTED WITH SW5
2477
2478  013772  032777  000040  165140  BIT     #SW5,@SWR       ;TEST THE SWITCH
2479  014000  001002                  BNE     5$              ;IF 0, TEST HAS NOT BEEN SELECTED
2480  014006                      5$:                         ;TEST SELECTED, CONTINUE IT
2481
2482
2483
2484
2485
2486
2487  014214  013737  004716  004752  MOV     @#UNIT,@#INUNIT ;MAKE THE INITIAL UNIT NO. = "UNIT"
2488
```

```
2489
2490
2491                                                ;*CHANGE ADDRESS PLUG ON THE UNIT UNDER TEST
2492
2493  014222                                  6$:
2494  014270  013746  004716                      MOV     @#UNIT,-(SP)    ;GET THE UNIT UNDER TEST
2495  014274  104405                              TYPDS                   ;TYPE IT OUT
2496  014412  000000                              HALT
2497
2498                                                ;*HOUSEKEEPING
2499
2500  014414  005037  047270                      CLR     @#PRITEM        ;CLEAR THE PREVIOUS ERROR NUMBER
2501  014420  005237  004716                      INC     @#UNIT          ;ADD ONE TO THE UNIT NO.
2502  014424  042737  177770  004716              BIC     #^C7,@#UNIT     ;TRUNCATE TO LOW ORDER 3 BITS
2503
2504
2505                                                ;*ATTEMPT TO ADDRESS THE NEW UNIT NUMBER
2506
2507  014436  017700  165642                      MOV     @RHDST,R0       ;ATTEMPT TO ADDRESS THE NEW DRIVE NO.
2508  014442  004737  043364                      JSR     PC,@#PUTREG     ;TAKE REG. SNAPSHOT IN CASE OF ERROR
2509  014446  032737  010000  002360              BIT     #NED,@#CS2      ;TEST FOR NON EXISTENT DRIVE
2510  014454  001423                              BEQ     7$              ;CONTINUE IF 'NED' IS NOT SET - DRIVE
2511                                                                      ;SHOULD BE EXISTENT ON THE BUSS
2512  014456  104102                              ERROR   102             ;UNIT NOT AVAILABLE AFTER ADDRESS
2513                                                                      ;PLUG REPLACED
2514  014504  013746  004716                      MOV     @#UNIT,-(SP)    ;GET THE BAD UNIT NUMBER
2515  014510  104405                              TYPDS                   ;TYPE IT OUT
2516
```

```
2517
2518                                        ;*CHECK IF ALL UNIT NUMBERS HAVE BEEN TRIED
2519
2520  014524  023737  004716  004752  7$:   CMP     @#UNIT,@#INUNIT ;HAVE WE INCREMENTED BACK TO THE
2521                                        ;ORIGINAL UNIT NO. YET ?
2522  014532  001233                        BNE     6$              ;NO..DO NEXT ADDRESS PLUG
2523                                        ;YES..CONTINUE WITH TESTS
2524
2525
2526
2527                                        ;*SET 'VV'IN RHDS1 AFTER RESET FROM THE RECALIBRATE
2528                                        ;*CAUSED BY PULLING THE ADDRESS PLUGS OUT
2529
2530
```

```
                                .SBTTL
                                .SBTTL  **DRIVE COMMAND TESTS**
                                .SBTTL


                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER NO OPERATION


                                        ;*NOW COMPARE REGISTERS BEFORE NO-OP COMMAND
                                        ;*WITH AFTER NO-OP COMMAND

015074  104016                  1$:     ERROR   16              ;GIVING A NO-OP COMMAND
015076  000207                          RTS     PC              ;CAUSED AN ERROR
                                                                ;NO REGISTERS SHOULD CHANGE
                                                                ;GOOD DATA GIVES REGISTER
                                                                ;CONTENTS BEFORE COMMAND
                                                                ;RECEIVED DATA GIVES REGISTER
                                                                ;CONTENTS AFTER COMMAND
```

```
2561
2562                                        ;*NOW REPEAT TEST BY MOVING IN ALL POSSIBLE ONES
2563
2564  015100                          2$:
2565  015104  012700  002272                MOV     #RHWC,R0        ;ADDR. OF ADDR OF RHWC IN R0
2566
2567
2568                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER A NO-OP
2569
2570
2571
2572
2573
2574                                        ;*CHANGE REGISTERS TO EXPECTED VALUES
2575
2576  015320  005737  004750                TST     @#RH70          ;RUNNING ON AN RH70 ?
2577  015324  001406                        BEQ     5$              ;IF NOT, SKIP NEXT
2578  015326  005737  004722                TST     @#NUNIT         ;TESTING MORE THAN ONE DRIVE ?
2579  015332  001003                        BNE     5$              ;SKIP NEXT IF SO
2580  015334  042737  100000  004620        BIC     #SC,@#SAVERE+6  ;CLEAR 'SC' IN RHCS1
2581
2582  015342                          5$:
2583
2584
2585                                        ;*NOW COMPARE REGISTERS BEFORE NO-OP WITH
2586                                        ;*AFTER NO-OP COMMAND
2587
2588
2589  015402  104016                  3$:   ERROR   16              ;GIVING A NO-OP COMMAND
2590  015404  000207                        RTS     PC              ;CAUSED AN ERROR
2591                                                                ;NO REGISTERS SHOULD CHANGE
2592                                                                ;GOOD DATA GIVES REGISTER
2593                                                                ;CONTENTS BEFORE COMMAND
2594                                                                ;RECEIVED DATA GIVES REGISTER
2595                                                                ;CONTENTS AFTER COMMAND
2596  015406                          4$:
2597
2598
```

```
                                        ;*WRITE ALL WRITABLE REGISTER BITS
                                        ;
015440  012700  002272                  MOV     #RHWC,R0        ;ADDR. OF ADDR. OF RHWC IN R0
                                        ;
015566  017737  164542  004654          MOV     @RHCC,@#SAVERE+42 ;SAVE RHCC IN SAVERE TABLE
                                        ;
                                        ;
                                        ;
                                        ;*NOW LOAD 'SAVERE' REGISTER SNAPSHOT WITH EXPECTED VALUES
                                        ;
015650  005037  004616                  CLR     @#SAVERE+4      ;CLEAR LOCATION FOR RHCS2
015654  053737  004716  004616          BIS     @#UNIT,@#SAVERE+4;PUT UNIT # BACK IN THE SAVED RHCS2
                                        ;
015662  005737  004750                  TST     @#RH70          ;RUNNING ON AN RH70 CONTROLLER ?
015666  001021                          BNE     8$              ;IF SO SKIP NEXT RH11 CODE
                                        ;
015730  000416                          BR      9$              ;SKIP NEXT RH70 CODE
                                        ;
015732                          8$:
015752  005737  004722                  TST     @#NUNIT         ;TESTING MORE THAN ONE DRIVE ?
015756  001003                          BNE     9$              ;SKIP NEXT IF SO
015760  042737  100000  004620          BIC     #SC,@#SAVERE+6  ;CLEAR 'SC' IF NOT
015766                          9$:
016046  013746  004742                  MOV     @#TOTALAT,-(SP) ;GET ALL BITS OF DRIVE & PRESENT
                                                                ;IN RHAS
016052  043716  004740                  BIC     @#ATTENT,(SP)   ;CLEAR WORKING DRIVE BIT
016056  012637  004636                  MOV     (SP)+,@#SAVERE+24 ;MOVE THIS INTO RHAS POSITION
                                        ;
016072                          3$:
                                        ;
016102  013737  002406  004644  4$:     MOV     @#DT,@#SAVERE+32 ;MOVE DRIVE TYPE INTO RHDT
                                                                ;POSITION OF SAVRE TABLE
016110  013737  002410  004646          MOV     @#SN,@#SAVERE+34 ;MOVE SERIAL NUMBER INTO RHSN
                                                                ;POSITION OF SAVERE TABLE
                                        ;
                                        ;*NOW THAT SAVERE TABLE HAS BEEN LOADED WITH
                                        ;*EXPECTED VALUES, THE REGISTERS WILL BE COMPARED
                                        ;*WITH SAVERE TABLE
                                        ;
016170  104017                  5$:     ERROR   17              ;DRIVE CLEAR COMMAND
016172  000207                          RTS     PC              ;GAVE AN ERROR
                                                                ;GOOD DATA HAS WHAT SHOULD
                                                                ;BE IN REGISTER AFTER A
                                                                ;DRIVE CLEAR
                                                                ;RECEIVED DATA HAS WHAT
                                                                ;THE REGISTER ACTUALLY
                                                                ;CONTAINED
016174                          6$:
                                        ;
```

```
2651
2652
2653  016270  013746  004716                    MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
2654  016274  104405                            TYPOS                   ;TYPE IT OUT
2655  016276  104401  001223                    TYPE    ,$CRLF
2656  016302  032713  010000          3$:       BIT     #MOL,@R3        ;MOL WILL BE HIGH TILL STOP IS HIT
2657  016306  001375                            BNE     3$              ;WAIT TILL STOP IS HIT
2658
2659
2660  016332  013746  004716                    MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
2661  016336  104405                            TYPOS                   ;TYPE IT
2662  016340  104401  001223                    TYPE    ,$CRLF
2663
2664  016344  032713  010000          4$:       BIT     #MOL,@R3        ;MOL WILL BE LOW TILL FILE READY
2665  016350  001775                            BEQ     4$              ;WAIT TILL FILE READY
2666  016370  012700  002272                    MOV     #RHWC,R0        ;ADDR. OF ADDR. OF RHWC IN R0
2667
2668
2669                                            ;*NOW INITIALIZE ALL THE REGISTERS
2670
2671
2672  016516  013777  004740  163572            MOV     @#ATTENT,@RHAS  ;CLEAR WORKING DRIVE 'ATA'
2673
2674                                            ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ-IN COMMAND
2675
2676
2677
2678  016552  005737  004750                    TST     @#RH70          ;RUNNING ON AN RH70 CONTROLLER ?
2679  016556  001411                            BEQ     7$              ;SKIP NEXT FOR RH70 IF NOT
2680  016600  000406                            BR      8$              ;SKIP NEXT FOR RH11
2681
2682  016602                            7$:
2683
2684  016616                            8$:
2685
2686                                            ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
2687
2688
2689                                            ;*NOW THAT SAVERE TABLE WITH SAVED REGISTERS HAVE
2690                                            ;*THE EXPECTED VALUE AFTER A READ-IN COMMAND
2691                                            ;*COMPARISONS ARE MADE
2692
2693
2694  016726  104020                    5$:     ERROR   20              ;READ IN COMMAND GAVE AN
2695  016730  000207                            RTS     PC              ;ERROR
2696                                                                    ;GOOD DATA HAS WHAT SHOULD
2697                                                                    ;BE IN REGISTER AFTER A
2698                                                                    ;READ-IN COMMAND
2699                                                                    ;RECEIVED DATA HAS WHAT
2700                                                                    ;THE REGISTER ACTUALLY CONTAINED
2701                                                                    ;THE FOLLOWING SHOULD
2702                                                                    ;BE THE REGISTER CONTENTS
2703                                                                    ;RHCA=0, RHDST = 0
2704                                                                    ;RHOF SHOULD HAVE FMT22 = 0,
2705                                                                    ;HCI = 0, ECI = 0,
2706                                                                    ;RHDS1 SHOULD HAVE VV = 1
```

```
 2707                                                          ;ALL OTHER BITS SHOULD
 2708                                                          ;BE UNCHANGED
 2709  016732  012737  177777  047270  6$:    MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
```

```
2710
2711
2712  016772  012700  002272                MOV     #RHWC,R0        ;ADDR. OF ADDR. OF RHWC IN R0
2713
2714
2715                                          ;*INITIALIZE ALL THE REGISTERS
2716
2717
2718  017056  013777  004740  163232        MOV     @#ATTENT,@RHAS  ;CLEAR WORKING DRIVE 'ATA'
2719
2720                                          ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ-IN COMMAND
2721
2722
2723
2724  017112  005737  004750                TST     @#RH70          ;RUNNING ON AN RH70 CONTROLLER ?
2725  017116  001411                        BEQ     9$              ;SKIP NEXT IF NOT
2726  017140  000406                        BR      10$             ;SKIP NEXT RH11 CODE
2727
2728  017142                          9$:
2729
2730  017156                          10$:
2731
2732                                          ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
2733
2734
2735                                          ;*NOW THAT SAVERE TABLE WITH SAVED REGISTERS HAVE
2736                                          ;*THE EXPECTED VALUE AFTER A READ-IN COMMAND
2737                                          ;*COMPARISONS ARE MADE
2738
2739
2740  017266  104020                  5$:     ERROR   20              ;READ-IN COMMAND GAVE AN
2741  017270  000207                          RTS     PC              ;ERROR
2742                                                                  ;GOOD DATA HAS WHAT SHOULD
2743                                                                  ;BE IN REGISTER AFTER A
2744                                                                  ;READ-IN COMMAND
2745                                                                  ;RECEIVED DATA HAS WHAT
2746                                                                  ;THE REGISTER ACTUALLY CONTAINE
2747                                                                  ;THE FOLLOWING SHOULD
2748                                                                  ;BE THE REGISTER CONTENTS
2749                                                                  ;RHCA = 0, RHDST = 0
2750                                                                  ;RHOF SHOULD HAVE FMT22 = 0,
2751                                                                  ;HCI = 0, ECI = 0,
2752                                                                  ;RHDS1 SHOULD HAVE VV = 1
2753                                                                  ;ALL OTHER BITS SHOULD
2754                                                                  ;BE UNCHANGED
2755  017272                          6$:
```

```
2756  017272  005737  004744          TST     @#RP06          ;TEST FOR RP06 DRIVE
2757  017276  001401                  BEQ     7$              ;IF = 0, TREAT DRIVE AS AN RP04
2758  017300  000402                  BR      8$              ;TREAT AS RP06 - DO NEXT "MAKECL"
2759  017302  000137  017346    7$:   JMP     @#DOG           ;DO SECOND FOLLOWING "MAKECL"
2760  017306              8$:
2761
2762
2763
2764  017342  000137  017402          JMP     @#FISH          ;DON'T DO NEXT "MAKECL"
2765
2766
2767  017346              DOG:
2768
```

```
2769
2770
2771
2772  017402                                FISH:
2773  017434  012700  002272                      MOV     #RHWC,R0        ;ADDR. OF ADDR. OF RHWC IN R0
2774
2775
2776                                              ;*NOW SAVE REGISTERS FOR COMPARISON AFTER RECALIBRATE
2777
2778
2779                                              ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
2780
2781
2782
2783                                              ;*NOW COMPARE REGISTERS AFTER A RECALIBRATE COMMAND
2784
2785
2786  020042  104064                      1$:     ERROR   64              ;RECALIBRATE COMMAND CAUSED
2787  020044  000207                              RTS     PC              ;AN ERROR
2788                                                                      ;GOOD DATA GIVES WHAT SHOULD
2789                                                                      ;BE THERE
2790                                                                      ;RECEIVED DATA GIVES WHAT WAS
2791                                                                      ;THERE AFTER COMMAND
2792
2793  020046                              2$:
```

```
2794
2795  020046  005737  004744          TST     @#RP06          ;TEST FOR RP06 DRIVE
2796  020052  001401                  BEQ     3$              ;IF = 0, TREAT DRIVE AS AN RP04
2797  020054  000402                  BR      4$              ;TREAT AS RP06 - DO NEXT "MAKECL"
2798  020056  000137  020122    3$:   JMP     @#CAT           ;DO SECOND FOLLOWING "MAKECL"
2799  020062                    4$:
2800
2801
2802  020116  000137  020156          JMP     @#BIRD          ;DON'T DO NEXT "MAKECL"
2803
2804
2805  020122                    CAT:
```

```
2806
2807
2808
2809  020156                          BIRD:
2810  020210  012700  002272                MOV     #RHWC,R0        ;ADDR. OF ADDR OF RHWC IN R0
2811
2812
2813                                          ;*NOW SAVE REGISTERS FOR COMPARISON AFTER RECALIBRATE
2814
2815                                          ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
2816
2817
2818                                          ;*NOW COMPARE REGISTERS AFTER A RECALIBRATE COMMAND
2819
2820
2821  020520  104064                  1$:     ERROR   64              ;RECALIBRATE COMMAND CAUSED
2822  020522  000207                          RTS     PC              ;AN ERROR
2823                                                                  ;GOOD DATA GIVES WHAT SHOULD BE
2824                                                                  ;THERE
2825                                                                  ;RECEIVED DATA GIVES WHAT WAS
2826  020524                          2$:                             ;THERE AFTER A RECALIBRATE
2827
```

```
2828
2829
2830                                        ;*THIS SETTING OF VV IS FOR LOOP ON ERROR ONLY
2831                                        ;*WHERE UNLOAD TAKES EFFECT AND CYCLE UP BRINGS VV DOWN
2832
2833  020606  017746  161510                MOV     @RHDS1,-(SP)    ;PUSH RHDS1 ONTO STACK
2834  020612  042716  167677                BIC     #167677,(SP)    ;CLEAR EVERYTHING EXCEPT VV AND MOL
2835  020616  022726  010100                CMP     #VV!MOL,(SP)+   ;ARE VV AND MOL SET ?
2836  020622  001504                        BEQ     6$              ;CONTINUE IF YES
2837  020766  000000                        HALT                    ;WAIT FOR CONTINUE
2838
2839
2840                                        ;*SET VV IN RHDS1 WITH PACK ACKNOWLEDGE
2841
2842
2843
2844  021034                        6$:
2845
2846  021046  012700  002272                MOV     #RHWC,R0        ;ADDR. OF ADDR OF RHWC IN R0
2847
2848                                        ;*LOAD ALL POSSIBLE REGISTERS WITH ONES
2849
2850
2851                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER UNLOAD
2852
2853
2854
2855
2856  021242  104413                        WAT                     ;WAIT FOR BIT TO SET
2857  021244  002300                        RHCS1                   ;IN RHCS1 REGISTER
2858  021246  000200                        RDY                     ;'RDY' BIT
2859  021250  000001                        1.                      ;ALLOW 10 MICRO SECONDS
2860  021252  000001                        1.                      ;ANOTHER 10 - 'RDY' MUST SET BETWEEN
2861                                                                ;00 AND 20 MICRO SECONDS
```

```
2862
2863                                        ;*COMPARE CONTENTS OF RHCS1 AND RHDS1, WHICH WERE SAVED
2864                                        ;*DURING THE UNLOAD COMMAND, WITH THE EXPECTED RESULTS
2865
2866  021254  013746  002424                MOV     @#UNLOAD,-(SP)   ;PUSH COMMAND ON STACK
2867  021260  052716  004201                BIS     #DVA!GO!RDY,(SP) ;INCLUDE THESE BITS SET
2868  021264  005737  004722                TST     @#NUNIT          ;IS THERE MORE THAN ONE UNIT ?
2869  021270  001413                        BEQ     9$               ;SKIP NEXT IF ONLY ONE UNIT
2870  021272  010037  004760                MOV     R0,@#TMP4        ;PUT SAVED RHCS1 INTO TMP4
2871  021276  042737  177677  004760        BIC     #^C<IE>,@#TMP4   ;MASK ALL BUT THE 'IE' BIT IN RHCS1
2872  021304  042716  000100                BIC     #IE,(SP)         ;CLEAR 'IE' IN EXPECTED DATA
2873  021310  053716  004760                BIS     @#TMP4,(SP)      ;SET 'IE' STATE FROM ACTUAL RHCS1 DATA
2874  021314  052716  100000                BIS     #SC,(SP)         ;SET 'SC' IN RHCS1 SAVED DATA
2875
2876  021320  011637  001124        9$:     MOV     (SP),@#$GDDAT    ;SAVE EXPECTED DATA FOR PRINTOUT
2877  021324  022600                        CMP     (SP)+,R0         ;COMPARE EXPECTED DATA WITH SAVED
2878                                                                 ;RHCS1 DATA AND RESET THE STACK
2879  021326  001405                        BEQ     10$              ;CHECK NEXT BITS IF THESE OK
2880  021330  010037  001126                MOV     R0,@#$BDDAT      ;RHCS1 IS BAD - PRINT IT OUT
2881  021334  010137  004600                MOV     R1,@#REGADR      ;REGISTER ADDRESS
2882  021340  104021                        ERROR   21               ;DURING ABOVE OPERATION ONLY THE
2883                                                                 ;'DVA'  'GO'  'RDY' AND COMMAND BITS
2884                                                                 ;SHOULD BE SET
2885
2886  021342  012746  020400        10$:    MOV     #PIP!DPR,-(SP)   ;PUT SOME EXPECTED RHDS1 BITS ON STACK
2887  021346  010537  004760                MOV     R5,@#TMP4        ;PUT SAVED RHDS1 INTO TMP4
2888  021352  042737  167677  004760        BIC     #^C<MOL!VV>,@#TMP4 ;MASK ALL BUT 'MOL' & 'VV' IN RHDS1
2889  021360  042716  010100                BIC     #MOL!VV,(SP)     ;CLEAR 'MOL' & 'VV' IN EXPECTED RHDS1
2890  021364  053716  004760                BIS     @#TMP4,(SP)      ;SET EXPECTED 'MOL' & 'VV' BIT STATES
2891                                                                 ;FROM THE ACTUAL DATA (DON'T CARE)
2892  021370  011637  001124                MOV     (SP),@#$GDDAT    ;SAVE EXPECTED DATA FOR PRINTOUT
2893  021374  022605                        CMP     (SP)+,R5         ;COMPARE EXPECTED DATA WITH SAVED
2894                                                                 ;RHDS1 DATA AND RESET THE STACK
2895  021376  001405                        BEQ     11$              ;CONTINUE IF EXPECTED=SAVED
2896  021400  010537  001126                MOV     R5,@#$BDDAT      ;RHDS1 IS BAD - PRINT IT OUT
2897  021404  010337  004600                MOV     R3,@#REGADR      ;REGISTER ADDRESS
2898  021410  104063                        ERROR   63               ;DURING THE ABOVE OPERATION, ONLY 'PIP
2899                                                                 ;AND 'DPR' SHOULD BE SET
2900                                                                 ;'MOL' & 'VV' ARE DON'T CARES
2901  021412                        11$:
2902
2903                                        ;THIS PROVIDES A 1 SECOND "STALL"
2904
```

```
2905
2906
2907                                        ;*NOW CHANGE REGISTERS SAVED BEFORE UNLOAD COMMAND
2908                                        ;*TO EXPECTED VALUES AFTER UNLOAD COMMAND
2909
2910                                        ;* - AGAIN 'MOL' $ 'VV' ARE DON'T CARES
2911
2912
2913  021472  012746  020400                MOV     #PIP!DPR,-(SP)  ;SET EXPECTED FINAL RHDS1 BITS
2914  021476  017737  160620  004760        MOV     @RHDS1,@#TMP4   ;GET PRESENT ACTUAL RHDS1 CONTENTS
2915  021504  042737  167677  004760        BIC     #^C<MOL!VV>,@#TMP4 ;MASK OUT ALL BUT 'MOL' & 'VV'
2916  021512  042716  010100                BIC     #MOL!VV,(SP)    ;CLEAR 'MOL' & 'VV' IN EXPECTED RHDS1
2917  021516  053716  004760                BIS     @#TMP4,(SP)     ;SET EXPECTED 'MOL' & 'VV' STATES
2918                                                                ;FROM THE ACTUAL (DON'T CARE COND.)
2919  021522  042716  100200                BIC     #ATA!DRY,(SP)   ;CLEAR THESE ADDITIONAL RHDS1 BITS
2920  021526  012637  004642                MOV     (SP)+,@#SAVERE+30 ;CHANGE THE SAVED RHDS1 REGISTER
2921                                                                ;AND ADJUST THE STACK
2922
2923
2924  021546  005737  004722                TST     @#NUNIT         ;IS THERE MORE THAN ONE UNIT ?
2925  021552  001006                        BNE     7$              ;SKIP NEXT IF MORE THAN ONE UNIT
2926  021570                          7$:
2927
2928                                        ;*NOW COMPARE REGISTERS AFTER THE UNLOAD COMMAND
2929                                        ;*WITH EXPECTED VALUES
2930
2931
2932  021614  104023                  3$:   ERROR   23              ;UNLOAD COMMAND GAVE
2933  021616  000207                        RTS     PC              ;AN ERROR
2934                                                                ;GOOD DATA GIVES WHAT SHOULD
2935                                                                ;BE THERE
2936                                                                ;RECEIVED DATA GIVES WHAT WAS
2937                                                                ;THERE AFTER UNLOAD COMMAND
2938
2939  021620
2940  021642  013746  004716            4$:   MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
2941  021646  104405                        TYPDS
2942
2943  021766  005037  047270                CLR     @#PRITEM        ;CLEAR PREVIOUS ERROR NUMBER
2944  021772  000000                        HALT                    ;WAIT FOR CONTINUE
2945
2946
2947                                        ;*SET VV IN RHDS1 AFTER RESET FROM ACTUATING
2948                                        ;*THE STANDBY SWITCH AND CYCLING UP (MOL = 1)
2949
2950
```

```
2951  022100  112737  000001  004610          MOVB    #1,@#OFSTVL      ;SET OFFSET VALUE TO 1
2952  022106  112737  000034  004611          MOVB    #34,@#OFSTVL+1 ;SET HCI,ECI,FMT22
2953
2954
2955  022114                          1$:
2956  022132  012700  002272                  MOV     #RHWC,R0         ;ADDR. OF ADDR OF RHWC IN R0
2957                                          ;*THE OFFSET REGISTER WILL BE INCREMENTED FROM 0 TO 377
2958
2959  022172  013730  004610                  MOV     @#OFSTVL,@(R0)+ ;SET OFFSET REGISTER
2960
2961
2962                                          ;*NOW SAVE REGISTERS FOR COMPARISON AFTER OFFSET
2963
2964
2965
2966
2967
2968                                          ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
2969
2970
2971                                          ;*NOW COMPARE REGISTERS AFTER AN OFSET COMMAND
2972
2973
2974  022530  104024                  2$:     ERROR   24               ;OFFSET COMMAND CAUSED AN ERROR
2975  022532  000207                          RTS     PC               ;GOOD DATA IS WHAT SHOULD BE THERE
2976                                                                   ;RECEIVED DATA GIVES WHAT WAS THERE
2977                                                                   ;AFTER AN OFFSET COMMAND
2978
2979  022534  013777  004740  157554  3$:     MOV     @#ATTENT,@RHAS   ;CLEAR WORKING DRIVE ATTENTION
2980
2981
2982
2983
2984                                          ;*NOW A RETURN TO CENTER LINE COMMAND WILL BE GIVEN
2985
2986
2987                                          ;*NOW REGISTERS ARE SAVED FOR COMPARISON AFTER COMMAND
2988
2989
2990
2991
2992
2993                                          ;*NOW CHANGE SAVED REGISTER TO EXPECTED VALUE
2994
2995
2996                                          ;*NOW COMPARE REGISTERS AFTER RETURN-TO-CENTER-LINE
2997  023034  104025                  4$:     ERROR   25               ;RETURN TO CENTER-LINE
2998  023036  000207                          RTS     PC               ;COMMAND CAUSED AN ERROR
2999                                                                   ;GOOD DATA HAS WHAT SHOULD
3000                                                                   ;BE THERE
3001                                                                   ;RECEIVED DATA HAS WHAT WAS
3002                                                                   ;THERE AFTER COMMAND
3003
3004  023040                          5$:
3005  023044  105237  004610                  INCB    @#OFSTVL         ;GET NEXT OFFSET VALUE
3006  023050  132737  000100  004610          BITB    #100,@#OFSTVL    ;SEE IF UNUSED BIT 6  IS ON
```

M06


```
 3007  023056  001403                    BEQ     7$              ;NO SO DO SOME MORE
 3008  023060  062737  000100  004610    ADD     #100,@#OFSTVL   ;YES SO BY-PASS IT
 3009  023066  105737  004610    7$:     TSTB    @#OFSTVL        ;IF ZERO ALL COMBINATIONS ARE
 3010                                                            ;COMPLETE
 3011  023072  001001                    BNE     6$              ;BRANCH IF 377 NOT DONE
 3012  023076  000137  022114    6$:     JMP     @#1$            ;JUMP BECAUSE 377 NOT DONE
 3013
```

```
 3014  023104  012737  000764  004756          MOV     # 500.,@#TMP1   ;COUNTER
 3015  023112                          1$:
 3016  023176  032777  040000  157116          BIT     #ERR,@RHDS1     ;IS ERR SET?
 3017  023204  001417                          BEQ     2$              ;NO
 3018  023206  104026                          ERROR   26              ;REPEATED OFFSETS CAUSED AN ERROR
 3019  023210  000004                          SCOPE
 3020  023244  005337  004756          2$:     DEC     @#TMP1          ;COUNT DOWN
 3021  023252  000717                          BR      1$              ;GO BACK AND DO IT AGIEN
 3022
```

```
                        .SBTTL  READ/WRITE TESTS USING MEDIA

                                ;*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

023614  104027          1$:   ERROR   27              ;WRITE HEADER AND DATA
023616  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
                                ;*NOTHING GOT CHANGED

023620                  2$:

023636  104030          3$:   ERROR   30              ;WRITE HEADER AND DATA
023640  000207                RTS     PC              ;CHANGED WRITE FROM BUFFER


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

023642                  4$:

                                ;*NOW FILL COMMAND
```

```
3079
3080                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ HEADER AND DATA
3081
3082
3083
3084
3085
3086
3087                                    ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3088
3089
3090                                    ;*COMPARE REGISTER BEFORE READ HEADER AND DATA
3091                                    ;*WITH REGISTERS AFTER COMMAND
3092
3093
3094  024114  104031            5$:   ERROR   31              ;READ HEADER AND DATA CAUSED
3095  024116  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
3096                                                          ;GOOD DATA GIVES WHAT SHOULD
3097                                                          ;BE THERE
3098                                                          ;RECEIVED DATA GIVES WHAT WAS
3099                                                          ;THERE AFTER COMMAND
3100
3101                                    ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
3102                                    ;*THE READ WAS GOOD
3103
3104  024120                    6$:
3105
3106  024136  104032            7$:   ERROR   32              ;WRITE HEADER AND DATA
3107  024140  000207                  RTS     PC              ;FOLLOWED BY A READ HEADER
3108                                                          ;AND DATA GAVE A READ ERROR
3109                                                          ;ERROR MAY BE IN READ OR WRITE
3110  024142                    10$:
3111
```

```
3112
3113
3114                                    ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3115
3116
3117                                    ;*NOW THE READ DATA COMMAND WILL BE FILLED
3118
3119
3120                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
3121
3122
3123
3124
3125
3126
3127                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
3128
3129
3130                                    ;*NOW COMPARE REGISTERS BEFORE READ DATA WITH
3131                                    ;*AFTER COMMAND
3132
3133
3134  024430  104033          1$:       ERROR   33              ;READ DATA CAUSED IMPROPER REGISTER
3135  024432  000207                    RTS     PC              ;CHANGE
3136                                                            ;GOOD DATA GIVES WHAT SHOULD BE THERE
3137                                                            ;RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND
3138                                    ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE THAT READ
3139                                    ;*WAS GOOD
3140
3141  024434                    2$:
3142
3143  024452  104034          3$:       ERROR   34              ;READ DATA COMMAND
3144  024454  000207                    RTS     PC              ;READ INCORRECTLY
3145
3146  024456                    4$:
```

```
3147
3148
3149                                    ;*NOW FILL WRITE FROM BUFFER -200 OF 1'S AND 56 OF 125252
3150
3151
3152                                    ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA AS
3153                                    ;*WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
3154                                    ;*CAN BE MADE TO DETERMINE THAT WRITE DID NOT CHANGE BUFFER
3155
3156
3157                                    ;*NOW WRITE DATA COMMAND WILL BE LOADED
3158
3159
3160                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE DATA
3161
3162
3163
3164
3165                                    ;*ONE REVOLUTION = 16670 MICRO SEC, ONE SECTOR=760 MICRO SEC
3166
3167
3168                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3169
3170
3171                                    ;*NOW COMPARE REGISTERS BEFORE WRITE DATA WITH REGISTERS
3172                                    ;*AFTER COMMAND
3173
3174
3175
3176  025002  104035          1$:     ERROR   35              ;WRITE DATA COMMAND CAUSED
3177  025004  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
3178                                                          ;GOOD DATA GIVES WHAT SHOULD
3179                                                          ;BE
3180                                                          ;RECEIVED DATA GIVES WHAT WAS
3181                                                          ;THERE AFTER COMMAND
3182
3183                                    ;*NOW WRITE FROM BUFFER WILL BE CHECKED FOR NO CHANGE
3184
3185  025006                  2$:
3186
3187  025024  104036          3$:     ERROR   36              ;WRITE DATA COMMAND CHANGED
3188  025026  000207                  RTS     PC              ;WRITE FROM BUFFER
3189
3190                                    ;*NOW A READ DATA COMMAND WILL BE GIVEN
3191
3192                                    ;*FILL READ INTO BUFFER WITH 200 ZEROS AND 56 OF 377
3193
3194  025030                  4$:
3195
3196                                    ;*FILL WRITE FROM BUFFER WITH 200 ONES AND 56 OF 377
3197
3198
3199                                    ;*NOW FILL COMMAND
3200
3201
3202                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
```

```
3203
3204
3205
3206
3207
3208
3209                                            ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3210
3211
3212                                            ;*COMPARE REGISTERS BEFORE READ DATA COMMAND
3213                                            ;*WITH REGISTERS AFTER COMMAND
3214
3215
3216  025340  104033                5$:       ERROR   33              ;READ DATA CAUSED IMPROPER
3217  025342  000207                          RTS     PC              ;REGISTER CHANGE
3218                                                                    ;GOOD DATA GIVES WHAT SHOULD BE THERE
3219                                                                    ;RECEIVED DATA GIVES WHAT WAS THERE
3220                                                                    ;AFTER COMMAND
3221
3222                                            ;*NOW READ INTO BUFFER IS CHECKED FOR GOOD READ
3223
3224  025344                        6$:
3225
3226  025362  104034                7$:       ERROR   34              ;INCORRECT DATA AFTER
3227  025364  000207                          RTS     PC              ;WRITE DATA FOLLOWED BY A
3228                                                                    ;READ DATA
3229  025366                        10$:
3230
3231
3232
```

```
3233
3234                                    ;*NOW FILL WRITE FROM BUFFER - 256 OF 125252
3235
3236
3237                                    ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA AS
3238                                    ;*WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
3239                                    ;*CAN BE MADE TO DETERMINE THAT WRITE DID NOT CHANGE BUFFER
3240
3241
3242                                    ;*NOW WRITE DATA COMMAND WILL BE LOADED
3243
3244
3245                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE DATA
3246
3247
3248
3249
3250
3251                                    ;*ONE REVOLUTION=16670 MICROSEC, ONE SECTOR=760 MICROSEC
3252
3253
3254                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3255
3256
3257                                    ;*NOW COMPARE REGISTERS BEFORE WRITE DATA WITH REGISTERS
3258                                    ;*AFTER COMMAND
3259
3260
3261  025666  104035          1$:     ERROR   35              ;WRITE DATA COMMAND CAUSED
3262  025670  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
3263                                                          ;GOOD DATA GIVES WHAT SHOULD
3264                                                          ;BE
3265                                                          ;RECEIVED DATA GIVES WHAT WAS
3266                                                          ;THERE AFTER COMMAND
3267
3268                                    ;*NOW WRITE FROM BUFFER WILL CHECKED FOR NO CHANGE
3269
3270  025672                    2$:
3271
3272  025710  104036          3$:     ERROR   36              ;WRITE DATA COMMAND CHANGED
3273  025712  000207                  RTS     PC              ;WRITE FROM BUFFER
3274
3275                                    ;*NOW A READ DATA COMMAND WILL BE GIVEN
3276                                    ;*FILL READ INTO BUFFER WITH 256 ZEROS
3277
3278  025714                    4$:
3279
3280                                    ;*FILL WRITE FROM BUFFER WITH 256 OF 125252
3281
3282
3283                                    ;*NOW FILL COMMAND
3284
3285
3286                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
3287
3288
```

```
3289
3290
3291
3292
3293                                    ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3294
3295
3296                                    ;*COMPARE REGISTERS BEFORE READ DATA COMMAND
3297                                    ;*WITH REGISTERS AFTER COMMAND
3298
3299
3300  026200  104033          5$:     ERROR   33              ;READ DATA CAUSED IMPROPER
3301  026202  000207                  RTS     PC              ;REGISTER CHANGE
3302                                                          ;GOOD DATA GIVES WHAT SHOULD BE THE
3303                                                          ;RECEIVED DATA GIVES WHAT WAS THERE
3304                                                          ;AFTER COMMAND
3305
3306                                    ;*NOW READ INTO BUFFER IS CHECKED FOR GOOD READ
3307
3308  026204                  6$:
3309
3310  026222  104034          7$:     ERROR   34              ;INCORRECT DATA AFTER
3311  026224  000207                  RTS     PC              ;WRITE DATA FOLLOWED BY A
3312                                                          ;READ DATA
3313  026226                  10$:
3314
```

```
3315
3316                                    ;*NOW FILL WRITE FROM BUFFER-200 OF 52525 AND 56 OF 377
3317
3318
3319                                    ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA AS
3320                                    ;*WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
3321                                    ;*CAN BE MADE TO DETERMINE THAT WRITE DID NOT CHANGE BUFFER
3322
3323
3324                                    ;*NOW WRITE DATA COMMAND WILL BE LOADED
3325
3326
3327                                    ;*NOW SAVE REGISTER FOR COMPARISON AFTER WRITE DATA
3328
3329
3330
3331
3332
3333                                    ;*ONE REVOLUTION=16670 MICROSEC, ONE SECTOR=760 MICROSEC
3334
3335
3336                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3337
3338
3339                                    ;*NOW COMPARE REGISTERS BEFORE WRITE DATA WITH REGISTERS
3340                                    ;*AFTER COMMAND
3341
3342
3343  026552  104035          1$:     ERROR   35              ;WRITE DATA COMMAND CAUSED
3344  026554  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
3345                                                          ;GOOD DATA GIVES WHAT SHOULD
3346                                                          ;BE
3347                                                          ;RECEIVED DATA GIVES WHAT WAS
3348                                                          ;THERE AFTER COMMAND
3349
3350                                    ;*NOW WRITE FROM BUFFER WILL BE CHECKED FOR NO CHANGE
3351
3352  026556                  2$:
3353
3354  026574  104036          3$:     ERROR   36              ;WRITE DATA COMMAND CHANGED
3355  026576  000207                  RTS     PC              ;WRITE FROM BUFFER
3356
3357                                    ;*NOW A READ DATA COMMAND WILL BE GIVEN
3358
3359                                    ;*FILL READ INTO BUFFER WITH 200 ZEROS AND 56 OF ALL ONES
3360
3361  026600                  4$:
3362
3363                                    ;*FILL WRITE FROM BUFFER WITH 200 OF 52525 AND 56 OF 0
3364
3365
3366                                    ;*NOW FILL COMMAND
3367
3368
3369                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND
3370
```

```
3371
3372
3373
3374
3375
3376                                    ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3377
3378
3379                                    ;*COMPARE REGISTERS BEFORE READ DATA COMMAND
3380                                    ;*WITH REGISTERS AFTER COMMAND
3381
3382
3383  027110  104033          5$:       ERROR   33              ;READ DATA CAUSED IMPROPER
3384  027112  000207                    RTS     PC              ;REGISTER CHANGE
3385                                                            ;GOOD DATA GIVES WHAT SHOULD BE THE
3386                                                            ;RECEIVED DATA GIVES WHAT WAS THERE
3387                                                            ;AFTER COMMAND
3388
3389                                    ;*NOW READ INTO BUFFER 'S CHECKED FOR GOOD READ
3390
3391  027114                    6$:
3392
3393  027132  104034          7$:       ERROR   34              ;INCORRECT DATA AFTER
3394  027134  000207                    RTS     PC              ;WRITE DATA FOLLOWED BY A
3395                                                            ;READ DATA
3396  027136                    10$:
3397
3398
3399
3400
```

```
3401
3402  027176  005037  004732                  CLR     @#UBUSB         ;CLEAR UNIBUS INDICATOR
3403
3404                                          ;*NOW FILL WRITE FROM BUFFER-200 OF 52525 AND 56 OF 377
3405
3406
3407                                          ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA AS
3408                                          ;*WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
3409                                          ;*CAN BE MADE TO DETERMINE THAT WRITE DID NOT CHANGE BUFFER
3410
3411
3412                                          ;*NOW WRITE DATA COMMAND WILL BE LOADED
3413
3414  027274  052777  002000  152776          BIS     #PSEL,@RHCS1    ;SET PORT B
3415                                                                  ;THAT IS UNIBUS B
3416
3417                                          ;*NOW SAVE REGISTER FOR COMPARISON AFTER WRITE DATA
3418
3419
3420
3421
3422                                          ;*ONE REVOLUTION=16670 MICROSEC, ONE SECTOR=760 MICROSEC
3423
3424
3425
3426                                  ;*CHECK IF NEM NON EXISTANT MEMORY IS SET
3427                                  ;*IF SET IT MEANS UNIBUS B IS NOT CONNECTED
3428                                  ;*SO THIS TEST IS NOT PERFORMED
3429
3430  027446  032777  004000  152622          BIT     #NEM,@RHCS2     ;TEST NEM
3431  027454  001441                          BEQ     11$             ;BRANCH IF UNIBUS B THERE
3432  027456  012737  177777  004732          MOV     #-1,@#UBUSB     ;UNIBUS B NOT THERE
3433  027544  104401  001223                  TYPE    ,$CRLF
3434  027550  104401  001223                  TYPE    ,$CRLF
3435  027554  000137  030306                  JMP     @#10$           ;JUMP TO NEXT TEST - NO UNIBUS B
3436  027560                          11$:
3437  027636  104401  001223                  TYPE    ,$CRLF
3438  027642  104401  001223                  TYPE    ,$CRLF
3439
3440                                          ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3441
3442
3443                                          ;*NOW COMPARE REGISTERS BEFORE WRITE DATA WITH REGISTERS
3444                                          ;*AFTER COMMAND
3445
3446
3447  027714  104074                  1$:     ERROR   74              ;WHILE USING UNIBUS B
3448                                                                  ;WRITE DATA COMMAND CAUSED
3449  027716  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
3450                                                                  ;GOOD DATA GIVES WHAT SHOULD
3451                                                                  ;BE
3452                                                                  ;RECEIVED DATA GIVES WHAT WAS
3453                                                                  ;THERE AFTER COMMAND
3454
3455                                          ;*NOW WRITE FROM BUFFER WILL BE CHECKED FOR NO CHANGE
3456
```

```
027720                          2$:

027736  104075                  3$:     ERROR   75              ;WHILE USING UNIBUS B
                                        ;WRITE DATA COMMAND CHANGED
027740  000207                          RTS     PC              ;WRITE FROM BUFFER

                                ;*NOW A READ DATA COMMAND WILL BE GIVEN
                                ;*FILL READ INTO BUFFER WITH 200 ZEROS AND 56 OF ALL ONES

027742                          4$:

                                ;*FILL WRITE FROM BUFFER WITH 200 OF 52525 AND 56 OF 0

                                ;*NOW FILL COMMAND

030040  052777  002000  152232          BIS     #PSEL,@RHCS1    ;SET PORT B
                                        ;THAT IS UNIBUS B

                                ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ DATA COMMAND

                                ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES

                                ;*COMPARE REGISTERS BEFORE READ DATA COMMAND
                                ;*WITH REGISTERS AFTER COMMAND

030260  104072                  5$:     ERROR   72              ;WHILE USING UNIBUS B
                                        ;READ DATA CAUSED IMPROPER
030262  000207                          RTS     PC              ;REGISTER CHANGE
                                        ;GOOD DATA GIVES WHAT SHOULD BE THE
                                        ;RECEIVED DATA GIVES WHAT WAS THERE
                                        ;AFTER COMMAND

                                ;*NOW READ INTO BUFFER IS CHECKED FOR GOOD READ

030264                          6$:

030302  104073                  7$:     ERROR   73              ;WHILE USING UNIBUS B
                                        ;INCORRECT DATA AFTER
030304  000207                          RTS     PC              ;WRITE DATA FOLLOWED BY A
                                        ;READ DATA
030306                          10$:
```

```
                                        ;*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

                                        ;*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 DAT1



                                        ;*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

030732  104027            1$:   ERROR   27              ;WRITE HEADER AND DATA
030734  000207                  RTS     PC              ;CAUSED IMPROPER REGISTER
                                                        ;CHANGE
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE
                                                        ;RECEIVED DATA GIVES WHAT
                                                        ;WAS THERE AFTER COMMANT

                                        ;*NOW WRITE FROM BUFFER WILL BE CHECKED TO SEE THAT
                                        ;*NOTHING GOT CHANGED

030736                      2$:

030754  104030              3$:   ERROR   30              ;WRITE HEADER AND DATA
030756  000207                  RTS     PC              ;CHANGED WRITE FROM BUFFER

                                        ;*NOW A READ HEADER AND DATA COMMAND WILL BE GIVEN
                                        ;*FOR SECTOR 1, 256 WORDS
                                        ;*READ INTO BUFFER IS FILLED WITH ONES
030760                      4$:

                                        ;*WRITE FROM BUFFER IS FILLED WITH EXPECTED DATA
```

```
3567
3568
3569                                    ;*NOW FILL COMMAND
3570
3571
3572                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ HEADER AND DATA
3573
3574
3575
3576
3577
3578                                    ;*ONE REVOLUTION = 16670 MICRO SEC, ONE SECTOR 760 MICRO SECONDS
3579
3580
3581                                    ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3582
3583
3584                                    ;*COMPARE REGISTERS BEFORE READ HEADER AND DATA
3585                                    ;*WITH REGISTERS AFTER COMMAND
3586
3587
3588
3589  031270  104031          5$:     ERROR   31              ;READ HEADER AND DATA CAUSED
3590  031272  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
3591                                                          ;GOOD DATA GIVES WHAT SHOULD
3592                                                          ;BE THERE
3593                                                          ;RECEIVED DATA GIVES WHAT WAS
3594                                                          ;THERE AFTER COMMAND
3595
3596                                    ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
3597                                    ;*THAT READ WAS GOOD
3598
3599  031274                    6$:
3600
3601
3602  031312  104032          7$:     ERROR   32              ;WRITE HEADER AND DATA
3603  031314  000207                  RTS     PC              ;FOLLOWED BY A READ HEADER
3604                                                          ;AND DATA GAVE A READ ERROR
3605                                                          ;ERROR MAY BE IN READ OR WRITE
3606  031316                    10$:
```

```
3607
3608
3609
3610
3611
3612
3613
3614
3615
3616
3617                                    ;*FILL WRITE FROM BUFFER WITH HEADER
3618                                    ;*FILL WRITE FROM BUFFER WITH DATA
3619                                    ;*FILL WRITE FROM BUFFER WITH NEXT TRACK HEADER
3620                                    ;*FILL WRITE FROM BUFFER WITH NEXT TRACK DATA
3621                                    ;*NOW READ INTO BUFFER WILL BE FILLED WITH SAME DATA
3622                                    ;*AS WRITE FROM BUFFER SO THAT AFTER A WRITE COMPARISONS
3623                                    ;*CAN BE MADE TO MAKE SURE THAT WRITE DID NOT
3624                                    ;*CHANGE WRITE FROM BUFFER.
3625
3626                                    ;*NOW THE WRITE HEADER AND DATA COMMAND WILL BE FILLED
3627
3628                                    ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE HEADER AND DATA1
3629
3630
3631
3632                                    ;*ONE REVOLUTION = 16670 MICRO1 SEC, ONE SECTOR = 760 MICRO1 SEC
3633
3634                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
3635
3636
3637                                    ;*NOW COMARE REGISTERS BEFORE WRITE HEADER AND DATA
3638                                    ;*WITH REGISTERS AFTER COMMAND
3639
3640
3641
3642  031762  104027          1$:     ERROR   27              ;WRITE HEADER AND DATA
3643  031764  000207                  RTS     PC              ;CAUSED IMPROPER REGISTER
3644                                                          ;CHANGE
3645                                                          ;GOOD DATA GIVES WHAT SHOULD
3646                                                          ;BE THERE
3647                                                          ;RECEIVED DATA GIVES WHAT
3648                                                          ;WAS THERE AFTER COMMANT
3649
3650                                    ;*NOW WRITE FROM BUFFER WILL BE CHECKED TO SEE THAT
3651                                    ;*NOTHING GOT CHANGED
3652
3653  031766                    2$:
3654
3655  032004  104030          3$:     ERROR   30              ;WRITE HEADER AND DATA
3656  032006  000207                  RTS     PC              ;CHANGED WRITE FROM BUFFER
3657
3658                                    ;*NOW A READ HEADER AND DATA COMMAND WILL BE GIVEN
3659                                    ;*READ INTO BUFFER IS FILLED WITH ONES
3660
3661  032010                    4$:
3662
```

```
                                        ;*WRITE FROM BUFFER IS FILLED WITH 10001,0,0,0,2000,2000, AND 254 OF 0
                                        ;*NOW FILL COMMAND
                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER READ HEADER AND DATA
                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
                                        ;*COMPARE REGISTERS BEFORE READ HEADER AND DATA
                                        ;*WITH REGISTERS AFTER COMMAND
032320  104031                  5$:     ERROR   31              ;READ HEADER AND DATA CAUSED
032322  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT WAS
                                                                ;THERE AFTER COMMAND
                                        ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
                                        ;*THAT READ WAS GOOD
032324                          6$:
032342  104032                  7$:     ERROR   32              ;WRITE HEADER AND DATA
032344  000207                          RTS     PC              ;FOLLOWED BY A READ HEADER
                                                                ;AND DATA GAVE A READ ERROR
                                                                ;ERROR MAY BE IN READ OR WRITE
032346                          10$:
```

```
3699
3700
3701                                        ;*THE FOLLOWING MOVES ARE TO INITIALIZE TEST FROM
3702                                        ;*CYLINDER 0
3703                                        ;*THESE LOCATIONS ARE CHANGED DURING TEST TO ENABLE
3704                                        ;*GOING TO NEXT CYLINDER
3705
3706  032366  012737  010000  032544        MOV     #10000,@#ST1+10
3707  032374  012737  001125  032564        MOV     #<<18.*40>!21.>,@#ST2+10
3708  032402  012737  010001  032576        MOV     #10001,@#ST3+10
3709  032410  012737  002000  032616        MOV     #2000,@#ST4+10
3710  032416  012737  000000  032624        MOV     #0,@#ST5+4
3711  032424  012737  000001  033034        MOV     #1,@#ST6+6
3712  032432  012737  000001  033044        MOV     #1,@#ST6+16
3713  032440  012737  010001  033126        MOV     #10001,@#ST9+10
3714  032446  012737  002000  033146        MOV     #2000,@#ST10+10
3715  032454  012737  000001  033166        MOV     #1,@#ST11+4
3716
3717                                        ;*THIS IS TO GET THE HEADS TO CYLINDER 0
3718
3719  032522  012737  000012  001200        MOV     #10.,@#STMP1    ;TEN COUNT TO GET TO CYLINDER 10
3720
3721                                        ;*FILL WRITE FROM BUFFER WITH HEADER
3722  032534                          ST1:
3723
3724                                        ;*FILL WRITE FROM BUFFER WITH DATA
3725  032554                          ST2:
3726
3727                                        ;*FILL WRITE FROM BUFFER WITH NEXT TRACK HEADER
3728  032566                          ST3:
3729
3730                                        ;*FILL WRITE FROM BUFFER WITH NEXT TRACK DATA
3731  032606                          ST4:
3732
3733                                        ;*THE WRITE HEADER AND DATA COMMAND WILL BE FILLED
3734  032620                          ST5:
3735
3736                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE HEADER AND DATA
3737
3738
3739
3740
3741
3742                                        ;*ONE REVOLUTION = 16670 MICRO SECONDS, ONE SECTOR = 760 MICRO SEC.
3743                                        ;*MAX TIME ALLOWED = ONE REVOLUTION + SEEK + 2 SECTORS
3744                                        ;*MIN TIME ALLOWED = 2 SECTORS + SEEK
3745
3746
3747                                        ;*NOW CHANGES SAVED REGISTERS TO EXPECTED VALUES
3748
3749  033026                          ST6:
3750
3751                                        ;*COMPARE REGISTERS BEFORE WRITE HEADER AND DATA
3752                                        ;*WITH REGISTERS AFTER COMMAND
3753
3754  033074  104027                ST7:    ERROR   27              ;WRITE HEADER AND DATA CAUSED
```

```
3755  033076  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
3756                                                                  ;GOOD DATA GIVES WHAT SHOULD BE
3757                                                                  ;THERE
3758                                                                  ;RECEIVED DATA GIVES WHAT WAS BE
3759                                                                  ;THERE AFTER COMMAND
3760
3761                                          ;*SETUP TO READ HEADER AND DATA FOR NEXT TRACK
3762                                          ;*FILL READ INTO BUFFER WITH ALL ONES
3763
3764  033100                          ST8:
3765
3766                                          ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3767
3768  033116                          ST9:
3769  033136                          ST10:
3770
3771                                          ;*FILL COMMAND INTO REGISTERS
3772  033162                          ST11:
3773
3774                                          ;*SAVE REGISTERS FOR COMPARISON AFTER READ HEADER
3775                                          ;*AND DATA
3776
3777
3778
3779
3780
3781
3782                                          ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
3783
3784
3785                                          ;*COMPARE REGISTERS BEFORE READ HEADER AND DATA WITH
3786                                          ;*REGISTERS AFTER COMMAND
3787
3788
3789  033416  104031                  ST12:   ERROR   31              ;READ HEADER AND DATA CAUSED
3790  033420  000207                          RTS     PC              ;IMPROPER REGISTER CHANGE
3791                                                                  ;GOOD DATA GIVES WHAT SHOULD
3792                                                                  ;BE THERE
3793                                                                  ;RECEIVED DATA GIVES WHAT
3794                                                                  ;WAS THERE AFTER COMMAND
3795
3796                                          ;*READ INTO BUFFER IS CHECKED FOR PROPER READ
3797  033422                          ST13:
3798
3799  033440  104032                  ST14:   ERROR   32              ;WRITE HEADER AND DATA
3800  033442  000207                          RTS     PC              ;WITH AN IMPLIED SEEK
3801                                                                  ;FOLLOWED BY A READ
3802                                                                  ;HEADER AND DATA ON THE
3803                                                                  ;NEXT TRACK GAVE A
3804                                                                  ;READ ERROR
3805                                                                  ;ERROR MAY BE READ OR WRITE
3806
3807                                          ;*THE HEADS HAVE ADVANCED ONE CYLINDER BY AN IMPLIED
3808                                          ;*SEEK
3809                                          ;*CHANGES WILL BE MADE TO ENABLE GOING TO THE NEXT
3810                                          ;*CYLINDER AND THEN THE ABOVE WILL BE REPEATED
```

```
3811                                                    ;*TILL CYLINDER 10 IS REACHED
3812
3813  033444  005237  032544          ST15:   INC     @#ST1+10
3814  033450  062737  002000  032564          ADD     #<1.*2000>,@#ST2+10
3815  033456  005237  032576                  INC     @#ST3+10
3816  033462  062737  002000  032616          ADD     #<1.*2000>,@#ST4+10
3817  033470  005237  032624                  INC     @#ST5+4
3818  033474  005237  033034                  INC     @#ST6+6
3819  033500  005237  033044                  INC     @#ST6+16
3820  033504  005237  033126                  INC     @#ST9+10
3821  033510  062737  002000  033146          ADD     #<1.*2000>,@#ST10+10
3822  033516  005237  033166                  INC     @#ST11+4
3823  033522  005337  001200                  DEC     @#$TMP1         ;COUNT FOR TEN TIMES
3824  033526  001001                          BNE     ST16            ;BRANCH IF 10 NOT DONE
3825  033530  000402                          BR      ST17            ;10 COMPLETED SO CONTINUE
3826  033532  000137  032534          ST16:   JMP     ST1             ;JUMP AS 10 NOT DONE
3827
3828                                                    ;*THE HEADS ARE NOW AT CYLINDER 10
3829                                                    ;*ALL REGISTERS WILL BE SAVED AND A SEEK WILL BE GIVEN
3830                                                    ;*TO CYLINDER 0
3831                                                    ;*FILL REGISTERS FOR A SEEK COMMAND
3832
3833  033536                          ST17:
3834
3835                                                    ;*SAVE REGISTERS FOR COMPARISON AFTER SEEK COMMAND
3836
3837
3838
3839                                                    ;*SEEK FOR ONE CYLINDER=7MILI SEC., FOR TEN=70 MILI SEC
3840
3841                                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3842
3843
3844                                                    ;*COMPARE REGISTERS AFTER A SEEK COMMAND
3845
3846  033766  104037          ST18:   ERROR   37              ;SEEK COMMAND CAUSED AN
3847  033770  000207                  RTS     PC              ;ERROR
3848                                                          ;GOOD DATA GIVES WHAT SHOULD
3849                                                          ;BE THERE
3850                                                          ;RECEIVED DATA GIVES WHAT WAS
3851                                                          ;THERE AFTER A SEEK COMMAND
3852
3853                                                    ;*AT THIS POINT THE CURRENT CYLINDER IS GOOD AND THERE ARE
3854                                                    ;*NO ERROR BITS
3855                                                    ;*A READ HEADER AND DATA WILL BE DONE ON CYLINDER 0
3856                                                    ;*SECTOR 21 TRACK 18. EXPECTED DATA IS 1125
3857                                                    ;*FOR 10 WORDS
3858                                                    ;*CLEAR READ INTO BUFFER WITH ALL ONES
3859
3860  033772                          ST19:
3861
3862                                                    ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA
3863
3864                                                    ;*FILL READ HEADER AND DATA COMMAND FOR 10 WORDS
3865
3866
```

```
3867
3868
3869
3870                                        ;*CHECK READ WORDS
3871
3872  034156  104032            ST26:     ERROR   32              ;READ HEADER AND DATA
3873  034160  000207                      RTS     PC              ;FOLLOWING A SEEK TO CYLINDER 0
3874                                                              ;FROM CYLINDER 10 GAVE AN
3875                                                              ;ERROR
3876
3877  034162                    ST27:
3878
```

```
3879
3880
3881
3882                                    ;*THIS GETS HEADS TO CYLINDER 0
3883
3884
3885
3886                                    ;*FILL REGISTERS FOR A SEEK COMMAND
3887
3888                                    ;*SAVE REGISTERS FOR COMPARISON AFTER SEEK COMMAND
3889
3890
3891
3892                                    ;*SEEK FOR ONE CYLINDER=7 MILI SEC., FOR TEN=70 MILI SEC
3893
3894                                    ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUE
3895
3896
3897                                    ;*COMPARE REGISTERS AFTER A SEEK COMMAND
3898
3899  034466  104037          1$:       ERROR   37              ;SEEK COMMAND CAUSED
3900  034470  000207                    RTS     PC              ;ERROR
3901                                                            ;GOOD DATA GIVES WHAT SHOULD
3902                                                            ;BE THERE
3903                                                            ;RECEIVED DATA GIVES WHAT WAS
3904                                                            ;THERE AFTER A SEEK COMMAND
3905
3906                                    ;*AT THIS POINT THE CURRENT CYLINDER IS GOOD AND THERE ARE
3907                                    ;*NO ERROR BITS
3908                                    ;*A READ HEADER AND DATA WILL BE DONE ON CYLINDER 9
3909                                    ;*SECTOR 21 TRACK 18, EXPECTED DATA IS 23125
3910                                    ;*FOR 20 WORDS
3911                                    ;*CLEAR READ INTO BUFFER WITH ALL ONES
3912
3913  034472                    2$:
3914
3915                                    ;*FILL  WRITE FROM BUFFER WITH EXPECTED DATA
3916
3917
3918                                    ;*FILL READ HEADER AND DATA COMMAND FOR 10 WORDS
3919
3920
3921
3922
3923
3924                                    ;*CHECK READ WORDS
3925
3926  034662  104032          3$:       ERROR   32              ;READ HEADER AND DATA
3927  034664  000207                    RTS     PC              ;FOLLOWING A SEEK TO CYLINDER 9
3928                                                            ;FROM CYLINDER 0 GAVE AN
3929                                                            ;ERROR
3930  034666                    4$:
3931
3932
3933
3934
```

```
                        .SBTTL  WRITE CHECK DATA & WRITE PROTECT TESTS
```

```
3937
3938                                        ;*GET HEADS TO CYLINDER 5
3939
3940
3941
3942
3943                                        ;*FILL WRITE FROM BUFFER WITH HEADER
3944
3945                                        ;*10 WORDS OF OF THE FOLLOWING DATA
3946                                        ;* 12344,17777,0,52525,125252
3947
3948
3949                                        ;*FILL LEFT ROTATING ZEROS FROM WRFROM+<54.*2>
3950
3951  035076  012700  177776                MOV     #177776,R0      ;DATA
3952  035102  012705  000020                MOV     #16.,R5         ;COUNT
3953  035106  012701  002644                MOV     #WRFROM+<54.*2>,R1 ;WHERE DATA GOES
3954  035112  000261                        SEC
3955  035114  010021              1$:       MOV     R0,(R1)+        ;STORE DATA
3956  035116  006100                        ROL     R0              ;GET ZERO ONE BIT LEFT
3957  035120  005305                        DEC     R5              ;COUNT 16
3958  035122  001374                        BNE     1$              ;BRANCH IF 16 NOT DONE
3959
3960                                        ;*FILL LEFT ROTATING ONE INTO WRFROM+<65.*2>
3961
3962  035124  000241                        CLC
3963  035126  012700  000001                MOV     #1,R0
3964  035132  010021              2$:       MOV     R0,(R1)+
3965  035134  006300                        ASL     R0
3966  035136  103375                        BCC     2$
3967
3968
3969                                        ;*FILL REST OF DATA
3970
3971
3972
3973                                        ;*READ INTO BUFFER WILL BE CLEARED
3974
3975                                        ;*THE WRITE HEADER AND DATA COMMAND WILL BE LOADED
3976
3977                                        ;*NOW SAVE REGISTERS FOR COMPARISON AFTER WRITE HEADER AND DATA
3978
3979
3980
3981
3982
3983
3984                                        ;*NOW CHANGE SAVED REGISTERS TO EXPECTED VALUES
3985
3986                                        ;*NOW COMPARE REGISTERS BEFORE WRITE HEADER AND DATA
3987                                        ;*WITH REGISTERS AFTER COMMAND
3988
3989
3990  035370  104027              3$:       ERROR   27              ;WRITE HEADER AND DATA
3991  035372  000207                        RTS     PC              ;CAUSED IMPROPER REGISTER
3992                                                                ;CHANGE
```

```
                                        ;GOOD DATA GIVES WHAT SHOULD
                                        ;BE THERE
                                        ;RECEIVED DATA GIVES WHAT
                                        ;WAS THERE AFTER COMMANT

                        ;*NOW FILL COMMAND FOR READ

035374                  4$:

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




                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES


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


035550  104031          5$:     ERROR   31              ;READ HEADER AND DATA CAUSED
035552  000207                  RTS     PC              ;IMPROPER REGISTER CHANGE
                                                        ;GOOD DATA GIVES WHAT SHOULD
                                                        ;BE THERE RECEIVED DATA GIVES WHAT WAS
                                                        ;RECEIVED DATA GIVES WHAT WAS
                                                        ;THERE AFTER COMMAND

                        ;*NOW READ INTO BUFFER WILL BE CHECKED TO SEE
                        ;*THAT READ WAS GOOD

035554                  6$:


035572  104032          7$:     ERROR   32              ;WRITE HEADER AND DATA
035574  000207                  RTS     PC              ;FOLLOWED BY A READ HEADER
                                                        ;AND DATA GAVE A READ ERROR
                                                        ;ERROR MAY BE IN READ OR WRITE

                        ;*A WRITE,READ HAS BEEN SUCCESSFULLY COMPLETED
                        ;*NOW A WRITE CHECK HEADER AND DATA WILL BE GIVEN
                        ;*FILL THE WRITE CHECK HEADER AND DATA

035576                  10$:

                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE CHECK

035624                  ST25:
```

```
4049
4050                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
4051
4052
4053                                        ;*COMPARE REGISTERS BEFORE WRITE CHECK HEADER AND DATA
4054                                        ;*WITH REGISTERS AFTER COMMAND
4055
4056
4057  036036  104040            8$:       ERROR   40              ;WRITE CHECK CAUSED
4058  036040  000207                      RTS     PC              ;AN IMPROPER REGISTER
4059                                                              ;CHANGE
4060                                                              ;GOOD DATA GIVES WHAT
4061                                                              ;SHOULD BE THERE
4062                                                              ;RECEIVED DATA GIVES WHAT
4063                                                              ;WAS THERE AFTER COMMAND
4064
4065
4066  036042                    9$:
```

```
4067
4068
4069                                        ;*GET HEADS TO CYLINDER 5
4070
4071
4072
4073
4074
4075
4076                                        ;*10 WORDS OF EACH 12344,17777,0,52525,125252
4077
4078
4079                                        ;*FILL LEFT ROTATING ZEROS FROM WRFROM+<50.*2>
4080
4081  036226  012700  177776              MOV     #177776,R0      ;DATA
4082  036232  012705  000020              MOV     #16.,R5         ;COUNT
4083  036236  012701  002634              MOV     #WRFROM+<50.*2>,R1 ;WHERE DATA GOES
4084  036242  000261                      SEC
4085  036244  010021              1$:     MOV     R0,(R1)+        ;STORE DATA
4086  036246  006100                      ROL     R0              ;GET ZERO ONE BIT LEFT
4087  036250  005305                      DEC     R5              ;COUNT 16
4088  036252  001374                      BNE     1$              ;BRANCH IF 16 NOT DONE
4089
4090                                        ;*FILL LEFT ROTATING ONE INTO WRFROM+<65.*2>
4091
4092  036254  000241                      CLC
4093  036256  012700  000001              MOV     #1,R0
4094  036262  010021              2$:     MOV     R0,(R1)+
4095  036264  006300                      ASL     R0
4096  036266  103375                      BCC     2$
4097
4098                                        ;*FILL REST OF DATA
4099
4100
4101                                        ;*FILL THE WRITE CHECK HEADER AND DATA
4102
4103                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE CHECK
4104
4105
4106
4107
4108
4109                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
4110
4111
4112                                        ;*COMPARE REGISTERS BEFORE WRITE CHECK HEADER AND DATA
4113                                        ;*WITH REGISTER AFTER COMMAND
4114
4115  036552  104040              8$:     ERROR   40              ;WRITE CHECK CAUSED
4116  036554  000207                      RTS     PC              ;AN IMPROPER REGISTER
4117                                                              ;CHANGE
4118                                                              ;GOOD DATA GIVES WHAT
4119                                                              ;SHOULD BE THERE
4120                                                              ;RECEIVED DATA GIVES WHAT
4121                                                              ;WAS THERE AFTER COMMANDS
4122
```

```
 4123  036556                          9$:
 4124
 4125
```

```
                                                ;*GET HEADS TO CYLINDER 5
036634                                  11$:
                                                ;*10 WORDS OF EACH 12344,17777,0,52525,125252
                                                ;*FILL LEFT ROTATING ZEROS FROM WRFROM+<50.*2>
036766  012700  177776                          MOV     #177776,R0      ;DATA
036772  012705  000020                          MOV     #16.,R5         ;COUNT
036776  012701  002634                          MOV     #WRFROM+<50.*2>,R1 ;WHERE DATA GOES
037002  000261                                  SEC
037004  010021                          1$:     MOV     R0,(R1)+        ;STORE DATA
037006  006100                                  ROL     R0              ;GET ZERO ONE BIT LEFT
037010  005305                                  DEC     R5              ;COUNT 16
037012  001374                                  BNE     1$              ;BRANCH IF 16 NOT DONE
                                                ;*FILL LEFT ROTATING ONE INTO WRFROM+<65.*2>
037014  000241                                  CLC
037016  012700  000001                          MOV     #1,R0
037022  010021                          2$:     MOV     R0,(R1)+
037024  006300                                  ASL     R0
037026  103375                                  BCC     2$
                                                ;*FILL REST OF DATA
                                                ;*FILL THE WRITE CHECK HEADER AND DATA
037100  052777  002000  143172                  BIS     #PSEL,@RHCS1    ;SET PORT B
                                                                        ;THAT IS UNIBUS B
                                                ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE CHECK
                                                ;*SET PORT SELECT
                                                ;*CHANGE SAVED REGISTERS TO EXPECTED VALUES
                                                ;*COMPARE REGISTERS BEFORE WRITE CHECK HEADER AND DATA
                                                ;*WITH REGISTER AFTER COMMAND
```

```
4182
4183  037320  104076          8$:   ERROR   76          ;WHILE USING UNIBUS B
4184                                                    ;WRITE CHECK CAUSED
4185  037322  000207                RTS     PC          ;AN IMPROPER REGISTER
4186                                                    ;CHANGE
4187                                                    ;GOOD DATA GIVES WHAT
4188                                                    ;SHOULD BE THERE
4189                                                    ;RECEIVED DATA GIVES WHAT
4190                                                    ;WAS THERE AFTER COMMANDS
4191
4192  037324                  9$:
4193
4194
4195
4196
4197
```

```
                                        ;*FILL SECTOR 0, TRACK 0, CYL 0 WITH ONES
                                        ;*FILL WRITE FROM BUFFER

                                        ;*FILL WRITE DATA COMMAND



                                        ;*TIME IS NOT CRITICAL

                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE PROTECT
                                        ;*BUTTON HAS BEEN HIT

037534  013746  004716                  MOV     @#UNIT,-(SP)            ;GET UNIT UNDER TEST
037540  104405                          TYPDS
037654  000000                          HALT
                                        ;*THE ONLY REGISTER THAT SHOULD  CHANGE IS RHDS1 - BIT #11
                                        ;*-WRL

                                        ;*COMPARE ALL REGISTERS BEFORE WRITE WAS LOCKED
                                        ;*OUT WITH REGISTER VALUES AFTER WRITE WAS LOCKED OUT

037710  104041                  3$:     ERROR   41              ;LOCKING OUT WRITE BY
037712  000207                          RTS     PC              ;WRITE LOCK BUTTON CAUSED
                                                                ;IMPROPER REGISTER CHANGE
                                                                ;GOOD DATA GIVES WHAT SHOULD
                                                                ;BE THERE
                                                                ;RECEIVED DATA GIVES WHAT
                                                                ;WAS THERE AFTER WRITE
                                                                ;WAS LOCKED OUT BY BUTTON

                                        ;*NOW A WRITE WILL BE ATTEMPTED WITH WRITE LOCKED
                                        ;*OUT BY BUTTON
                                        ;*FILL WRITE FROM BUFFER WITH 377

037714  013737  037714  001110  4$:     MOV     @#4$,@#$LPERR   ;SCOPE LOOP STARTS FROM HERE


                                        ;*TRY A ONE WORD WRITE

                                        ;*SAVE REGISTERS


                                        ;*TIME IS NOT CRITICAL

                                        ;*CHANGE SAVED REGISTERS TO EXPECTED VALUE

040030  017737  142236  004612          MOV     @RHWC,@#SAVERE  ;RHWC IS UNPREDICTABLE
040036  017737  142232  004614          MOV     @RHBA,@#SAVERE+2;RHBA IS UNPREDICTABLE
040044  017746  142226                  MOV     @RHCS2,-(SP)    ;GET RHCS2
040050  042716  177477                  BIC     #^C<IR!OR>,(SP) ;KEEP IR AND OR
040054  042737  000300  004616          BIC     #IR!OR,@#SAVERE+4;CLEAR SAVED IR OR
```

```
4254  040062  052637  004616                BIS     (SP)+,@#SAVERE+4;SET OR IR AS REQUIRED
4255
4256                                        ;*COMPARE REGISTERS BEFORE WRITE WAS ATTEMPTED
4257                                        ;*WITH REGISTERS AFTER ATTEMPT
4258
4259
4260  040176  104042              5$:     ERROR   42              ;ATTEMPTING TO WRITE WITH
4261  040200  000207                      RTS     PC              ;WRITE LOCKED OUT
4262                                                              ;CAUSED IMPROPER REGISTER
4263                                                              ;CHANGE
4264                                                              ;GOOD DATA GIVES WHAT SHOULD
4265                                                              ;BE THERE
4266                                                              ;RECEIVED DATA GIVES WHAT WAS
4267                                                              ;THERE AFTER ATTEMPT
4268
4269                                        ;*NOW A READ WILL BE DONE TO DETERMIN THAT
4270                                        ;*READS CAN BE DONE WITH WRITE LOCKED OUT AND
4271                                        ;*THAT NO DATA ON DISK GOT CHANGED, BUT FIRST CLEAR ERROR
4272
4273  040202              6$:
4274
4275                                        ;*FILL READ INTO BUFFER WITH 0
4276
4277                                        ;*FILL WRITE FROM BUFFER WITH EXPECTED DATA FROM READ
4278
4279                                        ;*FILL COMMAND
4280
4281
4282
4283
4284                                        ;*COMPARE READ DATA
4285
4286  040326  104043              7$:     ERROR   43              ;WRITING WITH WRITE
4287                                                              ;LOCKED CHANGED DISK
4288                                                              ;GOOD DATA GIVES WHAT WAS
4289                                                              ;ON DISK BEFORE WRITE WITH
4290                                                              ;WRITE LOCK WAS ATTEPTED
4291                                                              ;
4292                                                              ;RECEIVED DATA GIVES WHAT
4293                                                              ;WAS READ BACK AFTER WRITE
4294                                                              ;WITH WRITE LOCKED WAS ATTEPTED
4295
4296                                        ;*SAVE REGISTERS FOR COMPARISON AFTER WRITE LOCK HAS BEEN
4297                                        ;*UNLOCKED
4298
4299  040330              8$:
4300
4301  040342              ST20:
4302  040366  013746  004716                MOV     @#UNIT,-(SP)    ;GET UNIT UNDER TEST
4303  040372  104405                       TYPDS
4304  040502  000000                       HALT
4305
4306                                        ;*THE ONLY BIT THAT SHOULD CHANGE IS WRL-BIT #11 IN RHDS1
4307
4308                                        ;*COMPARE ALL REGISTERS BEFORE WRITE LOCK WAS UNLOCKED
4309                                        ;*WITH REGISTERS AFTER WRITE WAS UNLOCKED
```

```
4310
4311
4312  040536  104044                  9$:     ERROR   44              ;UNLOCKING WRITES BY WRITE
4313  040540  000207                          RTS     PC              ;LOCK BUTTON CAUSED AN ERROR
4314                                                                  ;GOOD DATA GIVES WHAT SHOULD
4315                                                                  ;BE THERE
4316                                                                  ;RECEIVED DATA GIVES WHAT WAS
4317                                                                  ;THERE AFTER WRITES WERE
4318                                                                  ;UNLOCKED
4319                                                                  ;ON THIS ERROR NO LOOPING IS RECOMMENDED
4320                                                                  ;JUST A HALT ON ERROR WILL DO THE SAME
4321                                                                  ;THING AS ONLY THE REGISTERS ARE READ
4322  040542  012737  177777  047270  10$:     MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
```

```
4323
4324
4325  040560  012737  000000  177776          MOV     #0,PS               ;REINSTATE PS TO 0
4326
4327  040644  013746  004716                  MOV     @#UNIT,-(SP)        ;GET READY TO TYPE UNIT NUMBER
4328  040650  104405                          TYPDS
4329  040664  013746  001112                  MOV     @#$ERTTL,-(SP)      ;GET READY TO TYPE NUMBER OF ERRORS
4330  040670  104405                          TYPDS
4331  040672  005037  001112                  CLR     @#$ERTTL            ;CLEAR TOTAL NUMBER OF ERRORS
4332  040676  005037  001102                  CLR     @#$TSTNM            ;CLEAR TEST NUMBER
4333  040702  005737  004726                  TST     @#SELECT            ;STARTING FROM 200 ?
4334  040706  001413                          BEQ     3$                  ;CHECK NEXT DRIVE IF SO
4335                                                                      ;CONTINUE WITH THIS ONE IF NOT
4336
4337  040710  005237  001100                  INC     @#$PASS             ;INCREASE PASS COUNT
4338  040714  104401  041105                  TYPE    ,$ENDMG             ;TYPE "END PASS #"
4339  040720  013746  001100                  MOV     @#$PASS,-(SP)
4340  040724  104405                          TYPDS
4341  040726  104401  041102                  TYPE    ,$ENULL
4342  040732  000137  010030                  JMP     @#TST4              ;JUMP TEST 4 ---------------------)
4343
4344  040736  012737  177777  047270  3$:     MOV     #-1,@#PRITEM        ;CLEAR PREVIOUS ITEM NUMBER
4345  040744  005337  004720                  DEC     @#NOUNITS           ;NO. OF UNITS PRESENT
4346  040750  001413                          BEQ     $EOP                ;BRANCH IF ALL DRIVES COMPLETE
4347  040752  013700  004716                  MOV     @#UNIT,R0           ;UNIT UNDER TEST
4348  040756  012701  004676                  MOV     #UNITS,R1           ;TABLE POINTER
4349  040762  022100          1$:     CMP     (R1)+,R0            ;IS THIS UNIT JUST TESTED ?
4350  040764  001401                          BEQ     2$                  ;BRANCH IF YES
4351  040766  000775                          BR      1$                  ;BRANCH IF NO
4352  040770  011137  004716          2$:     MOV     (R1),@#UNIT         ;MAKE THIS NEXT UNIT
4353  040774  000137  010030                  JMP     @#TST4              ;TEST THE NEXT DRIVE --------------)
4354
```

```
4355                                    .SBTTL
4356                                    .SBTTL  **SUBROUTINES**
4357                                    .SBTTL
4358
4359
4360
```

```
                                        .SBTTL  JAM CURRENT CYLINDER ROUTINE

                                ;*THIS ROUTINE WILL CHANGE THE CURRENT CYLINDER REGISTER - 'RHCC'
                                ;*BY GIVING A 'SEEK' COMMAND FOLLOWED BY AN INIT WHICH WILL LOAD
                                ;*'RHCC' WITH THE DESIRED CYLINDER VALUE.  THE ROUTINE THEN CHECKS
                                ;*THAT THE LOADED VALUE IS CORRECT.
                                ;*
                                ;*CALL IS:
                                ;*      JSR     R0,@#MAKECYL
                                ;*      XC                              ;DESIRED VALUE OF CURRENT CYLINDER
                                
041122                          MAKECYL:
041124  010037  041506                  MOV     R0,@#PCJSR              ;PC OF JSR+4
041130  162737  000004  041506          SUB     #4,@#PCJSR              ;SAVE PC OF JSR
041136  012005                          MOV     (R0)+,R5                ;GETTING READY TO FILL DESIRED CYLINDER
041140  010577  141146                  MOV     R5,@RHCA                ;FILL DESIRED CYLINDER REGISTER
041144  005077  141134                  CLR     @RHDST                  ;MAKE SURE DESIRED SECTOR TRACK IS NOT ILLEGAL
041150  013777  002452  141122          MOV     @#SEECOM,@RHCS1         ;FILL SEEK COMMAND
041156  012777  000001  141134          MOV     #DMD,@RHMR              ;SET DIAGNOSTIC MODE

041164  004737  043364                  JSR     PC,@#PUTREG             ;TAKE A REGISTER SNAPSHOT
041170  032737  040000  002404          BIT     #ERR,@#DS1              ;CHECK FOR COMPOSITE ERROR
041176  001401                          BEQ     2$                      ;NOT = 1, A-OK
041200  104103                          ERROR   103                     ;REGISTER CONTENTS INCORRECT BEFORE A
                                                                        ;DIAGNOSTIC SEEK

041202  052777  000001  141070  2$:     BIS     #GO,@RHCS1              ;ISSUE 'GO' TO SEEK COMMAND
041210  000240                          NOP                             ;ALLOW TIME FOR SEEK TO HANG UP
041212  000240                          NOP                             ;ALLOW TIME FOR SEEK TO HANG UP
041214  000240                          NOP                             ;ALLOW TIME FOR SEEK TO HANG UP
041216  000240                          NOP                             ;ALLOW TIME FOR SEEK TO HANG UP

041220  004737  043364                  JSR     PC,@#PUTREG             ;TAKE A 2ND REGISTER SNAPSHOT
041224  032737  040000  002404          BIT     #ERR,@#DS1              ;CHECK FOR ERRORS
041232  001401                          BEQ     3$                      ;NOT = 1, A-OK
041234  104104                          ERROR   104                     ;REGISTER CONTENTS INCORRECT AFTER
                                                                        ;A DIAGNOSTIC SEEK

041236  004737  041452          3$:     JSR     PC,@#CLDISK             ;GIVE INIT TO FORCE THE TRANSFER
041242  017737  141066  001126          MOV     @RHCC,@#$BDDAT          ;TEST DATA
041250  020537  001126                  CMP     R5,@#$BDDAT             ;COMPARE CURRENT CYLINDER
041254  001406                          BEQ     1$                      ;BRANCH IF GOOD
041256  010537  001124                  MOV     R5,@#$GDDAT             ;GOOD VALUE OF RHCC
041262  013737  002334  004600          MOV     @#RHCC,@#REGADR         ;FAILING REGISTER ADDRESS
041270  104077                          ERROR   77                      ;CURRENT CYLINDER DOES NOT MATCH DESIRED CYLINDER
                                                                        ;REGISTER AFTER A SEEK AND AN INIT

041272                          1$:
041274  000200                          RTS     R0
```

```
                                ;*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
                                ;*
                                ;*
                                ;*

041276                          FLHEAD:
041302  012001                          MOV     (R0)+,R1        ;R1 HAS ADDRESS OF WHERE TO SAVE
041304  012002                          MOV     (R0)+,R2        ;R2 HAS NUMBER OF WORDS

                                        ;*NOW FILL DATA

041306  012021                  1$:     MOV     (R0)+,(R1)+     ;SAVE DATA
041310  005302                          DEC     R2              ;DECREMENT COUNT
041312  001375                          BNE     1$              ;BRANCH IF INCOMPLETE
041320  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

041322                          CLAREA:
041330  012001                          MOV     (R0)+,R1        ;FROM
041332  012002                          MOV     (R0)+,R2        ;NUMBER
041334  012003                          MOV     (R0)+,R3        ;DATA
041336  010321                  1$:     MOV     R3,(R1)+        ;MOVE DATA
041340  005302                          DEC     R2              ;COUNT
041342  001375                          BNE     1$              ;BRANCH IF NOT COMPLETE
041352  000200                          RTS     R0              ;RETURN TO MAIN PROGRAM



                                ;*THIS IS A SUBROUTINE TO FILL SAVED REGISTER LOCATION
                                ;*WITH GIVEN VALUE
                                ;*CALL IS
                                ;*      JSR     R0,@#FILLRE
```

```
4467                                    ;*      RHXX            ;REGISTER NAME
4468                                    ;*      D               ;DATA
4469                                    ;*
4470
4471  041354                            FILLRE:
4472  041360  012001                           MOV     (R0)+,R1        ;ADDRESS OF ADDRESS OF REGISTER
4473  041362  012002                           MOV     (R0)+,R2        ;DATA
4474  041364  162701  002272                  SUB     #RHWC,R1        ;OFSET
4475  041370  010261  004612                  MOV     R2,SAVERE(R1)   ;DATA IS MOVED IN
4476  041400  000200                           RTS     R0              ;RETURN TO MAIN PROGRAM
4477
4478
4479
```

```
4480                                    ;*THIS SUBROUTINE SETS UP FOR SEARCH
4481                                    ;*CALL IS
4482                                    ;*      JSR     R0,@#SRCH
4483                                    ;*      C                       ;CYLINDER
4484                                    ;*.BYTE S                       ;SECTOR
4485                                    ;*.BYTE T                       ;TRACK
4486
4487  041402  012077  140704           SRCH:   MOV     (R0)+,@RHCA     ;SET DESIRED CYLINDER ADDRESS
4488  041406  012077  140672                   MOV     (R0)+,@RHDST    ;SET DESIRED SECTOR/TRACK ADDRESS
4489  041412  013777  002434  140660           MOV     @#SERCH,@RHCS1  ;GET READY FOR SEARCH
4490                                                                   ;WITH 30 IN RHCS1
4491  041420  000200                         RTS     R0
4492
4493
4494
4495
4496
4497
4498
4499                                    ;*THIS SUBROUTINE SETS UP FOR SEEK COMMANDS
4500                                    ;*CALL IS
4501                                    ;*      JSR     R0,@#SEEKCY
4502                                    ;*      C                       ;CYLINDER
4503                                    ;*
4504
4505  041422  012077  140664           SEEKCY: MOV     (R0)+,@RHCA     ;SET DESIRED CYLINDER ADDRESS
4506  041426  013777  002452  140644           MOV     @#SEECOM,@RHCS1 ;MOV 4 INTO RHCS1
4507  041434  000200                         RTS     R0              ;RETURN TO MAIN PROGRAM
```

```
4508
4509                                    ;*THIS SUBROUTINE SETS UP FOR OFFSET COMMANDS
4510                                    ;*CALL IS
4511                                    ;*      JSR     R0,@#OFSET
4512                                    ;*      0               ;MICRO INCHES OFSET
4513
4514  041436  052077  140646          OFSET:  BIS     (R0)+,@RHOF     ;SET OFSET REGISTER
4515  041442  013777  002454  140630          MOV     @#OFSETC,@RHCS1 ;MOV14 INTO RHCS1
4516  041450  000200                  RTS     R0              ;RETURN TO MAIN PROGRAM
4517
4518
4519  041452  013701  002300          CLDISK: MOV     @#RHCS1,        R1      ;R1 WILL BE CONTROL AND STATUS1
4520  041456  013702  002276                  MOV     @#RHCS2,        R2      ;R2 WILL BE CONTROL AND STATUS2
4521  041462  013703  002322                  MOV     @#RHDS1,        R3      ;R3 WILL BE DISK STATUS REGISTER1
4522  041466  013704  002302                  MOV     @#RHER1,        R4      ;R4 WILL BE ERROR REGISTER #1
4523
4524  041472  012712  000040                  MOV     #CLR,@R2                ;CLEAR ALL REG.
4525  041476  013712  004716                  MOV     @#UNIT,@R2      ;REINSTATE UNIT NO.
4526  041502  005011                  CLR     @R1                     ;CLEAR FUNCTION BITS
4527  041504  000207                  RTS     PC
```

```
                                        ;*THIS CHECKS THAT DEVICE AVAILABLE (DVA) AND READY (RDY) IN RHCS1 = 1
                                        ;*AND CHECKS MEDIUM ON LINE (MOL), DEVICE PRESENT (DPR), DEVICE READY
                                        ;* (DRY) IN RHDS1 = 1

                                        ;*IT ALSO CHECKS THAT THERE ARE NO BITS STUCK AT 1 IN RHDS1

041506  000000                  °CJSR:  0                       ;PC OF JSR

041510  011637  041506          CHECK:  MOV     (SP),@#PCJSR    ;SAVE PC OF JSR+4
041514  162737  000004  041506          SUB     #4,@#PCJSR      ;GET PC OF JSR
041522  011346                          MOV     @R3,-(SP)       ;GET RHDS1
041524  052716  000100                  BIS     #VV,(SP)        ;DONT CHECK VV BIT
041530  000406                          BR      CHECKC          ;GOTO COMMON CHECK ROUTINE

041532  011637  041506          CHECKT: MOV     (SP),@#PCJSR    ;SAVE PC OF JSR+4
041536  162737  000004  041506          SUB     #4,@#PCJSR      ;GET PC OF JSR
041544  011346                          MOV     @R3,-(SP)       ;GET RHDS1 & DO VV CHECK AT 3$

041546  011146                  CHECKC: MOV     @R1,-(SP)       ;GET CS1
041550  042716  173577                  BIC     #173577,(SP)    ;CLEAR UNWANTED BITS
041554  022726  004200                  CMP     #DVA!RDY,(SP)+  ;RHCS1 SHOULD HAVE DEVICE AVAILABLE
                                                                ;AND BE READY
041560  001403                          BEQ     3$              ;BRANCH IF IT DOES
041562  011137  001122                  MOV     @R1,@#$BDADR    ;BAD DATA REGISTER (RHCS1)
041566  104061                          ERROR   61              ;RHCS1 DID NOT HAVE DEVICE
                                                                ;AVAILABLE RIGHT AT THE START
                                                                ;ALL OTHER BITS SHOULD BE 0

041570  042716  102000          3$:     BIC     #ATA!LBT,(SP)   ;CLEAR UNWANTED BITS
041574  022726  010700                  CMP     #MOL!DPR!DRY!VV,(SP)+ ;RHDS1 SHOULD HAVE THESE SET
041600  001404                          BEQ     7$              ;BRANCH IF GOOD
041602  011337  001122                  MOV     @R3,@#$BDADR    ;BAD DATA IN  REGISTER (RHDS1)
041606  104062                          ERROR   62              ;RHDS1 HAS SOME BITS OTHER
                                                                ;THAN MOL, DRY, DPR,VV SET
                                                                ;ALL OTHER BITS SHOULD BE 0
041610  000207                          RTS     PC              ;RETURN TO TEST AND HALT/CONTINUE
                                                                ;DEPENDING ON WHETHER THIS IS A
                                                                ;"FATAL" ERROR

041612  062716  000006          7$:     ADD     #6,(SP)         ;ADJUST STACK TO JUMP OVER HALT IN TEST
041616  000207                          RTS     PC              ;RETURN TO THE 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

041620                          SAVEP:
041626  012001                          MOV     (R0)+,R1        ;FROM
041630  012002                          MOV     (R0)+,R2        ;TO
041632  012003                          MOV     (R0)+,R3        ;NUMBER
041634  013122                  1$:     MOV     @(R1)+,(R2)+    ;SAVE REGISTER CONTENTS
041636  005303                          DEC     R3              ;COUNT
041640  001375                          BNE     1$              ;BRANCH IF NOT DONE
041650  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
                                ;*

041652  005077  140466          TIME1:  CLR     @PCLCSR                 ;STOP THE CLOCK
041656  017737  140466  041710          MOV     @PCLCTR,@#WAITTM ;GET TIME ON CLOCK
041664  017737  140444  004664  TIME2:  MOV     @RHCC,@#FINACC          ;GET CURRENT CYLINDER
041672  017737  140440  004662          MOV     @RHLA,@#FINALA          ;GET LOOK AHEAD
041700  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
                                ;*
```

```
4631                                    ;*R1-WILL HAVE TIME ALLOWED IN 10 MICRO SECONDS
4632                                    ;*R2-WILL HAVE TOLERANCE PLUS/MINUS IN 10 MICRO SECONDS
4633                                    ;*MINIMUM TIME THAT CAN BE MEASURED IS ABOUT 12 MICRO SECONDS
4634                                    ;*FOR THE SLOWEST PROCESSOR
4635
4636  041702  000000                    WAITPC: 0                       ;WAT PC
4637  041704  000000                    WAITRE: 0                       ;WAIT ON REGISTER ADDRESS
4638  041706  000000                    WAITBT: 0                       ;WAIT ON BIT
4639  041710  000000                    WAITTM: 0                       ;WAITED TIME
4640  041712  005037  041710            WAIT.P: CLR     @#WAITTM        ;CLEAR WAITED TIME
4641  041716  005077  140424                    CLR     @PCLBUF ;CLEAR COUNT SET BUFFER
4642  041722  012777  000021  140414            MOV     #GO!BIT4,@PCLCSR ;COUNT UP, 100 KHZ, START CLOCK
4643  041730  016600  000010                    MOV     10(SP),R0       ;R0 HAS ADDRESS OF NEXT LOCATION
4644  041734  010037  041702                    MOV     R0,@#WAITPC     ;NOW WAITPC HAS WAT PC + 2
4645  041750  162737  000002  041702            SUB     #2,@#WAITPC     ;WAT PC IS IN WAITPC
4646  041756  013037  041704                    MOV     @(R0)+,@#WAITRE ;WAIT ON REGISTER ADDRESS
4647  041762  012037  041706                    MOV     (R0)+,@#WAITBT  ;WAIT ON BIT
4648  041766  012001                            MOV     (R0)+,R1        ;R1 HAS TIME IN 10 MSEC
4649  041770  012002                            MOV     (R0)+,R2        ;R2 HAS TOLERANCE IN 10 MSEC
4650  041772  010066  000010                    MOV     R0,10(SP)       ;RESTORE RETURN ON STACK
4651
4652                                    ;*THIS SECTION WAITS FOR BIT, THROUGH TWO COUNT DOWNS
4653  041776  013703  042150            MOV     @#TIMCNT,R3     ;R3 IS A TEMPORARY COUNTER
4654  042002  033777  041706  177674  1$:       BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
4655  042010  001025                            BNE     4$              ;BRANCH IF YES
4656  042012  005303                            DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
4657  042014  001372                            BNE     1$
4658  042016  013703  042150            MOV     @#TIMCNT,R3     ;TEMPORARY COUNTER
4659  042022  033777  041706  177654  2$:       BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
4660  042030  001015                            BNE     4$              ;BRANCH IF YES
4661  042032  005303                            DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
4662  042034  001372                            BNE     2$
4663  042036  017737  177642  001126            MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
4664  042044  032777  000100  140226            BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
4665  042052  001402                            BEQ     3$              ;BRANCH IF YES
4666  042054  104001                            ERROR   1               ;RP04 DID NOT INTERRUPT
4667  042056  000427                            BR      7$              ;OUT
4668  042060  104002                    3$:     ERROR   2               ;RP04 INTERRUPTED BUT WAITED
4669                                                                    ;ON BIT DID NOT OCCUR
4670                                                                    ;EVEN AFTER TWO COUNT DOWNS
4671                                                                    ;FROM 177777 TO 0
4672  042062  000425                            BR      7$              ;OUT
4673
4674                                    ;*NOW TIME AND TOLERANCE WILL BE CHECKED
4675  042064  017737  177614  001126  4$:       MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
4676  042072  032777  000100  140200            BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
4677  042100  001402                            BEQ     5$              ;BRANCH IF YES
4678  042102  104003                            ERROR   3               ;INTERRUPT DID NOT OCCUR EVEN
4679                                                                    ;AFTER ONE BNE AND ONE MOV
4680                                                                    ;OF THE WAITED ON BIT SETTING
4681  042104  000414                            BR      7$              ;OUT
4682  042106  160201                    5$:     SUB     R2,R1           ;R1 NOW HAS LOWER LIMIT OF TIME
4683  042110  023701  041710                    CMP     @#WAITTM,R1     ;FOR GOOD RESULTS, WAITTM
4684                                                                    ;MUST BE GREATER OR EQUAL
4685                                                                    ;TOR1
4686  042114  103002                            BHIS    6$              ;BRANCH IF GOOD
```

```
4687  042116  104004                        ERROR   4               ;BIT DID OCCUR BUT TIME
4688                                                                  ;TAKEN IS BELOW LOWER LIMIT
4689  042120  000406                        BR      7$              ;OUT
4690
4691  042122  060202                6$:     ADD     R2,R2           ;DOUBLE TOLERANCE
4692  042124  060201                        ADD     R2,R1           ;R1 NOW HAS UPPER LIMIT OF TIME
4693  042126  020137  041710                CMP     R1,@#WAITTM     ;FOR GOOD RESULTS, WAITTM
4694                                                                  ;MUST BE LESS OR EQUAL TO R1
4695  042132  103001                        BHIS    7$              ;BRANCH IF GOOD
4696  042134  104004                        ERROR   4               ;BIT DID OCCUR BUT TIME TAKEN
4697                                                                  ;IS ABOVE UPPER LIMIT
4698  042136                        7$:
4699  042146  000002                        RTI                     ;RETURN TO MAIN TEST
4700
4701
4702
4703
4704
4705
4706                                ;*THIS IS A WAIT LOOP WHEN NO P-CLOCK IS AVAILABLE
4707                                ;*NO TIMING IS DONE
4708                                ;*CALL IS
4709                                ;*      WAT
4710                                ;*      A       ;ABSOLUTE REGISTER ADDRESS
4711                                ;*      B       ;BIT WAITE) FOR
4712                                ;*      TA      ;TIME-NOT USED HERE
4713                                ;*      TO      ;TIME-NOT USED HERE
4714                                ;*R3-IS A TEMPORARY COUNTER
4715  042150  177777                TIMCNT: 177777                  ;COUNT FOR WAIT LOOP
4716
4717  042152                        WAIT.T:
4718  042156  016600  000004                MOV     4(SP),R0        ;R0 HAS ADDRESS OF NEXT LOCATION
4719  042162  010037  041702                MOV     R0,@#WAITPC     ;WAT PC +2 IS IN WAITPC
4720  042166  162737  000002  041702        SUB     #2,@#WAITPC     ;WAT PC IS IN WAITPC
4721  042174  013037  041704                MOV     @(R0)+,@#WAITRE ;WAIT ON REGISTER ADDRESS
4722  042200  012037  041706                MOV     (R0)+,@#WAITBT  ;WAIT ON BIT
4723  042204  022020                        CMP     (R0)+,(R0)+     ;DUMP NEXT TWO WORDS-TA, TO
4724  042206  010066  000004                MOV     R0,4(SP)        ;RESTORE RETURN ON STACK
4725
4726                                ;*THIS HAS THE TWO COUNT DOWNS FROM 177777
4727  042212  013703  042150                MOV     @#TIMCNT,R3     ;R3 HAS TEMPORARY COUNT
4728  042216  033777  041706  177460  1$:   BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
4729  042224  001025                        BNE     4$              ;BRANCH IF YES
4730  042226  005303                        DEC     R3              ;COUNT IF REQUIRED BIT NOT THERE
4731  042230  001372                        BNE     1$
4732  042232  013703  042150                MOV     @#TIMCNT,R3     ;SECOND COUNT DOWN FROM 177777
4733  042236  033777  041706  177440  2$:   BIT     @#WAITBT,@WAITRE ;IS REQUIRED BIT THERE
4734  042244  001015                        BNE     4$              ;BRANCH IF YES
4735  042246  005303                        DEC     R3              ;COUNT IF REQUIRED BIT NOOT THERE
4736  042250  001372                        BNE     2$
4737  042252  017737  177426  001126        MOV     @WAITRE,@#$BCDAT ;REGISTER CONTENTS FOR TYPEOUT
4738  042260  032777  000100  140012        BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
4739  042266  001402                        BEQ     3$              ;BRANCH IF YES
4740  042270  104001                        ERROR   1               ;RP04 DID NOT INTERRUPT
4741                                                                  ;BIT DID NOT OCCUR
4742  042272  000414                        BR      5$              ;OUT
```

```
4743  042274  104002                3$:     ERROR   2               ;RP04 INTERRUPTED BUT
4744                                                                  ;WAITED ON BIT DID NOT OCCUR
4745                                                                  ;EVEN AFTER TWO COUNT DOWNS
4746                                                                  ;FROM 177777 TO 0
4747  042276  000412                        BR      5$              ;OUT
4748
4749                                        ;*BIT DID SET SO CHECK IF INTERRUPT OCCURED
4750  042300  000240                4$:     NOP                     ;ALLOW TIME FOR INTERRUPT
4751  042302  032777  000100  137770        BIT     #IE,@RHCS1      ;DID ANY INTERRUPT OCCUR
4752  042310  001405                        BEQ     5$              ;BRANCH IF YES
4753  042312  017737  177366  001126        MOV     @WAITRE,@#$BDDAT ;REGISTER CONTENTS FOR TYPEOUT
4754  042320  104003                        ERROR   3               ;INTERRUPT DID NOT OCCUR
4755                                                                  ;EVEN AFTER ONE BNE OF
4756                                                                  ;THE WAITED ON BIT OCCURING
4757  042322  000400                        BR      5$              ;OUT
4758  042324                        5$:
4759  042330  000002                        RTI                     ;RETURN TO MAIN TEST
```

```
                                        ;*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
042332                                  CHREG:
042336  012001                                  MOV     (R0)+,R1        ;R1 HAS ADDRESS OF ADDRESS OF REGISTER
042340  012002                                  MOV     (R0)+,R2        ;R2 HAS NUMBER OF CHANGES
042342  162701  002272                        SUB     #RHWC,R1        ;R1 HAS OFSET OF REQUIRED REGISTER
042346  005720                          1$:     TST     (R0)+           ;IS A BIC OR A BIS TO BE DONE
042350  001403                                  BEQ     2$              ;BRANCH IF A BIC IS REQUIRED
042352  052061  004612                        BIS     (R0)+,SAVERE(R1) ;SET REQUIRED BIT
042356  000402                                  BR      3$              ;BRANCH TO DECREMENT COUNT
042360  042061  004612                  2$:     BIC     (R0)+,SAVERE(R1) ;CLEAR REQUIRED BIT
042364  005302                          3$:     DEC     R2              ;DECREMENT NUMBER OF CHANGES
042366  001367                                  BNE     1$              ;BRANCH IF NOT COMPLETE
042374  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

042376                                  FILL:
042406  012001                                  MOV     (R0)+,R1        ;R1 HAS ADDRESS WHERE DATA IS TO GO
042410  012002                                  MOV     (R0)+,R2        ;R2 HAS NUMBER OF WORDS TO BE FILLED
042412  012003                                  MOV     (R0)+,R3        ;STARTING VALUE OF DATA
042414  012004                                  MOV     (R0)+,R4        ;R4 HAS INCREMENT
                                                ;*NOW DATA WILL BE FILLED
042416  010321                          1$:     MOV     R3,(R1)+        ;FILL DATA
042420  060403                                  ADD     R4,R3           ;GET NEXT VALUE OF DATA
042422  005302                                  DEC     R2              ;DECREMENT COUNT
042424  001374                                  BNE     1$              ;BRANCH IF ALL NOT DONE
042436  000200                                  RTS     R0              ;RETURN TO MAIN PROGRAM
```

```
                                        ;*THIS IS A SUBROUTINE TO COMPARE REGISTERS
                                        ;*GOOD DATA IS ALREADY SAVED IN 'SAVERE'
                                        ;*TEST DATA IS IN THE REGISTERS
                                        ;*CALL IS
                                        ;*      JSR     R0,@#COMREG
                                        ;*      SAVERE                  ;GOOD DATA
                                        ;*      RHCS1                   ;ADDRESS OF ADDRESS TEST DATA
                                        ;*      N.                      ;RETURN FOR ERROR
                                        ;*      RG                      ;RETURN FOR GOOD COMPARISON
                                        ;*ON RETURN WITH ERROR '$GDDAT' HAS GOOD DATA, '$BDDAT' HAS BAD DATA
                                        ;*'REGADR' HAS REGISTER ADDRESS

042440                                  COMREG:
042452  012001                                  MOV     (R0)+,R1        ;R1 HAS ADDRESS OF GOOD DATA
042454  012002                                  MOV     (R0)+,R2        ;R2 HAS ADDRESS OF ADDRESS OF TEST DATA
042456  012003                                  MOV     (R0)+,R3        ;R3 HAS NUMBER OF WORDS
042460  012004                                  MOV     (R0)+,R4        ;R4 HAS RETURN FOR ERROR
042462  011000                                  MOV     (R0),R0         ;R0 HAS RETURN ON NO ERROR
                                                ;*NOW SAVE REGISTERS
042464  004737  043364                          JSR     PC,@#PUTREG     ;SAVE REGISTERS
042470  113737  004637  002401                  MOVB    @#$AVERE+25,@#AS+1;MAKE UPPER BYTE OF RHAS SAME
042476  012705  177776                          MOV     #-2,R5          ;PRESET R5 TO -2
                                                ;*NOW COMPARES WILL MADE
042502  062705  000002                  1$:     ADD     #2,R5           ;INCREMENT TO INDEX
042506  022122                                  CMP     (R1)+,(R2)+     ;COMPARE REGISTER CONTENTS
042510  001420                                  BEQ     2$              ;BRANCH IF GOOD
042512  014137  001124                          MOV     -(R1),@#$GDDAT  ;SAVE GOOD DATA
042516  014237  001126                          MOV     -(R2),@#$BDDAT  ;SAVE BAD DATA
042522  016537  002272  004600                  MOV     RHWC(R5),@#REGADR       ;SAVE ADDRESS OF FAILING REGISTER
042530  004714                                  JSR     PC,@R4          ;RETURN TO MAIN PROGRAM
                                                                        ;TO PRINT ERROR
042532  022122                                  CMP     (R1)+,(R2)+     ;UNDO -(R1) AND -(R2) FOR ERRORS
042534  017746  136400                          MOV     @SWR,-(SP)      ;GET SWITCH SETTING
042540  042716  177177                          BIC     #^C600,(SP)     ;KEEP ONLY SWITCH 7 AND 8
042544  022726  000200                          CMP     #SW07,(SP)+     ;IS 7 SET AND 8 DOWN
042550  001402                                  BEQ     3$              ;BRANCH OUT IF YES
042552  005303                          2$:     DEC     R3              ;ARE ALL COMPARES DONE
042554  001352                                  BNE     1$              ;BRANCH IF NOT COMPLETE

042556                                  3$:
042570  000200                                  RTS     R0              ;RETURN TO MAIN PROGRAM
042572  000000                          4$:     .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.

042574  000000                  TESTAD: 0                       ;FIRST ADDRESS OF TEST
042576                          OPERSEL:
042576  005037  177776                  CLR     PS              ;MAKE PROCESSOR STATUS ZERO
042602  012737  177777  047270          MOV     #-1,@#PRITEM    ;CLEAR PREVIOUS ITEM NUMBER
042660  013746  004604                  MOV     @#TSTNM,-(SP)   ;GET READY TO TYPE TEST
042664  104402                          TYPOC                   ;NUMBER
042724  013746  001110                  MOV     @#$LPERR,-(SP)  ;GET READY TO TYPE LOOP BACK PC
042730  104402                          TYPOC
042732  104401  001223                  TYPE    ,$CRLF
043164  104412                          RDOCT
043166  062716  000002                  ADD     #2,(SP)         ;GET LPADR
043172  012637  001106                  MOV     (SP)+,@#$LPADR
043350  104412                          RDOCT
043352  012637  001110                  MOV     (SP)+,@#$LPERR  ;GET LPERR
043356  013746  001106                  MOV     @#$LPADR,-(SP)
043362  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


043364                          PUTREG:
043372  012700  002272                  MOV     #RHWC,R0        ;STARTING ADDRESS OF REG
043376  012701  002354                  MOV     #WC,R1          ;STARTING ADDRESS OF WERE SAVED
043402  012702  000022                  MOV     #RHCC-RHWC+2/2,R2 ;NUMBER OF REG. INTO R2
043406  013021                  10$:     MOV     @(R0)+,(R1)+    ;SAVE HARDWARE REG.
043410  005302                          DEC     R2
043412  001375                          BNE     10$
043422  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
043424  012C77  136662          RUN:    MOV     (R0)+,@RHCA     ;CYLINDER
043430  012077  136650                  MOV     (R0)+,@RHDST    ;DESIRED SECTOR/TRACK
043434  012077  136632                  MOV     (R0)+,@RHWC     ;WORD COUNT
043440  012077  136630                  MOV     (R0)+,@RHBA     ;BUS ADDRESS
043444  013746  004716                  MOV     @#UNIT,-(SP)    ;GET UNIT NO
043450  052016                          BIS     (R0)+,(SP)      ;SET BUS ADDRESS INHIBIT
043452  012677  136620                  MOV     (SP)+,@RHCS2    ;UNIT NO AND BAI TO RHCS2
043456  012077  136626                  MOV     (R0)+,@RHOF     ;FORMAT, ECC INHIBIT, HEADER
                                                                ;COMPARE, IF THERE
043462  013077  136612                  MOV     @(R0)+,@RHCS1   ;COMMAND IN RHCS1
043466  000200                          RTS     R0              ;RETURN TO MAIN PROGRAM
```

```
4942
4943
4944                                    ;*THIS IS A SUBROUTINE TO COMPARE TWO BLOCKS IN MEMORY
4945
4946                                    ;*R1 HAS GOOD DATA BUFFER ADDRESS
4947                                    ;*R2 HAS TEST DATA BUFFER ADDRESS
4948                                    ;*R5 HAS ADDRESS OF RETURN ON ERROR
4949                                    ;*R3 HAS NUMBER OF WORDS TO BE COMPARED
4950                                    ;*R4 HAS ONE MORE THAN NUMBER OF WORDS TO BE COMPARED
4951
4952                                    ;*CALL IS:
4953                                    ;*      JSR     R0,@#COMPAR
4954                                    ;*      G                       ;ADDRESS OF GOOD DATA
4955                                    ;*      T                       ;ADDRESS OF TEST DATA
4956                                    ;*      N                       ;NUMBER OF WORDS TO BE COMPARED
4957                                    ;*      RE                      ;RETURN ON ERROR
4958                                    ;*      RG                      ;RETURN ON NO ERROR
4959
4960
4961
4962  043470                            COMPAR:
4963  043502  012001                            MOV     (R0)+,R1        ;ADDRESS OF GOOD DATA BUFFER
4964  043504  012002                            MOV     (R0)+,R2        ;ADDRESS OF TEST DATA BUFFER
4965  043506  012003                            MOV     (R0)+,R3        ;NO OF WORDS TO BE COMPARED
4966  043510  012005                            MOV     (R0)+,R5        ;RETURN ON ERROR
4967  043512  011000                            MOV     (R0),R0         ;RETURN ON NO ERROR
4968  043514  010304                            MOV     R3,R4           ;NO OF WORDS TO BE COMPARED
4969  043516  005204                            INC     R4
4970  043520  010437  004602           1$:     MOV     R4,@#ERWORD     ;FOR ERROR WORD NO
4971  043524  022122                            CMP     (R1)+,(R2)+     ;COMPARE GOOD WITH TEST DATA
4972  043526  001417                            BEQ     2$              ;BRANCH IF GOOD
4973
4974  043530  014137  001124                    MOV     -(R1),@#$GDDAT  ;GOOD DATA
4975  043534  014237  001126                    MOV     -(R2),@#$BDDAT  ;BAD DATA
4976  043540  160337  004602                    SUB     R3,@#ERWORD     ;ERROR WORD NO.
4977  043544  004715                            JSR     PC,@R5          ;RETURN TO PRINT ERROR
4978  043546  022122                            CMP     (R1)+,(R2)+     ;UNDO -(R1) AND -(R2) FOR ERRORS
4979  043550  017746  135364                    MOV     @SWR,-(SP)      ;GET SWITCH SETTING
4980  043554  042716  177177                    BIC     #^C600,(SP)     ;KEEP ONLY SWITCH 7 AND 8
4981  043560  022726  000200                    CMP     #SW07,(SP)+     ;IS 7 SET AND 8 RESET
4982  043564  001402                            BEQ     3$              ;BRANCH OUT IF YES
4983  043566  005303                    2$:     DEC     R3              ;COUNT
4984  043570  001353                            BNE     1$              ;BRANCH IF ALL NOT DEVICE
4985  043572                            3$:
4986  043604  000200                            RTS     R0              ;RETURN TO MAIN PROGRAM
```

```
                                        ;*      THIS ROUTINE WILL ALLOW THE CHANGE OF THE BASE
                                        ;*      ADDRESS FROM 176700 TO ANY TYPED VALUE

043606                                  BASECH:
043666  013746  002300                          MOV     @#RHCS1,-(SP)   ;GET READY TO TYPE OLD BASE
043672  104402                                  TYPOC
043754  004737  045662                          JSR     PC,@#$TKINT     ;INITIALIZE THE TTY KEYBOARD
043760  104412                                  RDOCT
043762  012700  002270                          MOV     #RHDB,R0        ;GET STARTING ADDRESS OF REGISTERS
043766  012701  000026                          MOV     #22.,R1         ;NUMBER OF REGISTERS
043772  012737  044572  000004                  MOV     #ADTIMO,@#4     ;SET TRAP CATCHER TO CHECK THIS ADDRESS
044000  021637  002300                          CMP     @SP,@#RHCS1     ;NEW ADDRESS?
044004  001407                                  BEQ     1$              ;NO, OLD ONE JUST RETYPED.
044006  005776  000000                          TST     @0(SP)          ;OK, SO ACCESS THIS NEW ADDRESS
044012  163716  002300                          SUB     @#RHCS1,@SP     ;GET THE ADDRESS OFFSET
044016  061620                          2$:     ADD     @SP,(R0)+       ;AND PLUG IT IN.
044020  005301                                  DEC     R1              ;ONE LESS REGISTER TO GO
044022  001375                                  BNE     2$              ;BUT WE'RE NOT DONE YET.
044024                                  1$:
044070  013746  002266                          MOV     @#RPVEC,-(SP)   ;GET READY TO TYPE OLD VECTOR ADDRESS
044074  104402                                  TYPOC
044202  104412                                  RDOCT
044204  012637  002266                          MOV     (SP)+,@#RPVEC   ;SETUP VECTOR ADDRESS
044252  013746  002300                          MOV     @#RHCS1,-(SP)
044256  104402                                  TYPOC
044322  013746  002266                          MOV     @#RPVEC,-(SP)
044326  104402                                  TYPOC
044546  012746  000200                          MOV     #RA,-(SP)
044552  104402                                  TYPOC
044566  000137  005012                          JMP     @#BEGIN         ;RESTART, TO RUN ALL DRIVES
044572                                  ADTIMO:
044654  022626                                  CMP     (SP)+,(SP)+     ;RESTORE THE STACK
044656  000137  043606                          JMP     @#BASECH        ;AND DO THE WHOLE THING AGAIN!
```

```
5024
5025  044662                          RPVECT:
5026  044712  104402                          TYPOC                   ;TYPE FROM PC
5027  044714  012777  044662  135344          MOV     #RPVECT,@RPVEC  ;RESTORE TRAP RP04 VECTOR
5028  044722  000000                          HALT                    ;CHANGE TO CONTINUE
5029
5030
5031
5032
```

```
5033                                                    ;FROM THE TTY
```

SEQ 0129

5034

```
                                ;*;**********************************

                                .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.

047270  000000                  PRITEM: 0                       ;PREVIOUS ITEM NO. LOCATION

047272  017746  131642          $ERRTYP: MOV    @SWR,-(SP)      ;GET SWITCH SETTING
047276  042716  177277                   BIC    #^C500,(SP)     ;KEEP ONLY SWITCH 8 AND 6
047302  022726  000100                   CMP    #SW06,(SP)+     ;IS 6 SET AND 8 RESET
047306  001001                          BNE    1$              ;IF NOT BRANCH
047310  000402                          BR     2$              ;BRANCH IF SW 6 IS SET AND 8 RESET
047312  000137  050232          1$:     JMP    @#TYPERR        ;JUMP IF SW 8 IS SET
                                                                ;OR IF SW 8 IS RESET AND SW 6 IS RESET
047316                          2$:
050226  005037  047270                   CLR    @#PRITEM        ;CLEAR PREVIOUS ERROR ITEM
050232                          TYPERR:
050232  104401  001223                   TYPE    $CRLF          ;"CARRIAGE RETURN" & "LINE FEED"
050236  010046                          MOV    R0,-(SP)        ;SAVE R0
050240  005000                          CLR    R0              ;PICKUP THE ITEM INDEX
050242  153700  001114                   BISB   @#$ITEMB,R0
050246  001004                          BNE    1$              ;IF ITEM NUMBER IS ZERO, JUST
                                                                ;TYPE THE PC OF THE ERROR
050250  013746  001116                   MOV    $ERRPC,-(SP)    ;SAVE $ERRPC FOR TYPEOUT
                                                                ;ERROR ADDRESS
050254  104402                          TYPOC                  ;GO TYPE--OCTAL ASCII(ALL DIGITS)
050256  000454                          BR     10$             ;GET OUT
050260  005300                  1$:     DEC    R0              ;ADJUST THE INDEX SO THAT IT WILL
050262  006300                          ASL    R0              ;       WORK FOR THE ERROR TABLE
050264  006300                          ASL    R0
050266  006300                          ASL    R0
050270  062700  001226                   ADD    #$ERRTB,R0      ;FORM TABLE POINTER
050274  020037  047270                   CMP    R0,@#PRITEM     ;WAS PREVIOUS ERROR SAME
050300  001002                          BNE    13$             ;BRANCH IF NOT
050302  022020                          CMP    (R0)+,(R0)+     ;POP R0 OVER EM AND DH
050304  000420                          BR     5$
050306  010037  047270          13$:    MOV    R0,@#PRITEM     ;SAVE NEW ERROR ITEM
050312  012037  050322                   MOV    (R0)+,2$        ;PICKUP "ERROR MESSAGE" POINTER
050316  001404                          BEQ    3$              ;SKIP TYPEOUT IF NO POINTER
050320  104401                          TYPE                   ;TYPE THE "ERROR MESSAGE"
050322  000000                  2$:     .WORD  0               ;"ERROR MESSAGE" POINTER GOES HERE
050324  104401  001223                   TYPE    $CRLF          ;"CARRIAGE RETURN" & "LINE FEED"
050330  012037  050340          3$:     MOV    (R0)+,4$        ;PICKUP "DATA HEADER" POINTER
050334  001404                          BEQ    5$              ;SKIP TYPEOUT IF 0
050336  104401                          TYPE                   ;TYPE THE "DATA HEADER"
```

```
5091  050340  000000            ..      4$:     .WORD   0               ;"DATA HEADER" POINTER GOES HERE
5092  050342  104401  001223                    TYPE    $CRLF           ;"CARRIAGE RETURN" & "LINE FEED"
5093  050346  010146            5$:     MOV     R1,-(SP)        ;SAVE R1
5094  050350  012001                    MOV     (R0)+,R1        ;PICKUP "DATA TABLE" POINTER
5095  050352  001415                    BEQ     9$              ;BR IF NO DATA TO BE TYPED
5096  050354  012000                    MOV     (R0)+,R0        ;PICKUP "DATA FORMAT" POINTER
5097  050356  105720            6$:     TSTB    (R0)+           ;"OCTAL" OR "DECIMAL"
5098  050360  001003                    BNE     7$              ;BR IF DECIMAL
5099  050362  013146                    MOV     @(R1)+,-(SP)    ;SAVE @(R1)+ FOR TYPEOUT
5100  050364  104402                    TYPOC                   ;GO TYPE--OCTAL ASCII(ALL DIGITS)
5101  050366  000402                    BR      8$
5102  050370                    7$:
5103  050370  013146                    MOV     @(R1)+,-(SP)    ;SAVE @(R1)+ FOR TYPEOUT
5104  050372  104405                    TYPDS                   ;GO TYPE--DECIMAL ASCII WITH SIGN
5105  050374  005711            8$:     TST     (R1)            ;IS THERE ANOTHER NUMBER?
5106  050376  001403                    BEQ     9$              ;BR IF NO
5107  050400  104401  050414            TYPE    ,11$            ;TYPE TWO(2) SPACES
5108  050404  000764                    BR      6$              ;LOOP
5109
5110  050406  012601            9$:     MOV     (SP)+,R1        ;RESTORE R1
5111  050410  012600            10$:    MOV     (SP)+,R0        ;"CARRIAGE RETURN" & "LINE FEED"
5112  050412  000207                    RTS     PC              ;RETURN
5113  050414  020040  000       11$:    .ASCIZ  /  /            ;TWO(2) SPACES
5114          050420                    .EVEN
5115
```

```
5116
```

```
5117
```

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

051114 050122 032060 042040  EM1:    .ASCIZ  /RPO4 DID NOT INTERRUPT/
051122 042111 047040 052117
051130 044440 052116 051105
051136 052522 052120    000
051143    111 052116 051105  EM2:    .ASCIZ  /INTERRUPT ENABLE BIT DOWN BUT EXPECTED BIT DID NOT SET/
051150 052522 052120 042440
051156 040516 046102 020105
051164 044502 020124 047504
051172 047127 041040 052125
051200 042440 050130 041505
051206 042524 020104 044502
051214 020124 044504 020104
051222 047516 020124 042523
051230 000124
051232 050122 032060 042040  EM3:    .ASCIZ  /RPO4 DID NOT INTERRUPT WHEN EXPECTED BIT DID SET/
051240 042111 047040 052117
051246 044440 052116 051105
051254 052522 052120 053440
051262 042510 020116 054105
051270 042520 052103 042105
051276 041040 052111 042040
051304 042111 051440 052105
051312    000
051313    105 050130 041505  EM4:    .ASCIZ  /EXPECTED BIT DID SET BUT TIME IS IN ERROR (TIME IN 10 MICROSEC, DECIMAL
051320 042524 020104 044502
051326 020124 044504 020104
051334 042523 020124 052502
051342 020124 044524 042515
051350 044440 020123 047111
051356 042440 051122 051117
051364 024040 044524 042515
051372 044440 020116 030061
051400 046440 041511 047522
051406 042523 026103 042040
051414 041505 046511 046101
051422 000051
051424 044122 051501 042040  EM5:    .ASCIZ  /RHAS DOES NOT CLEAR BY MOVING IN ALL ONES/
051432 042517 020123 047516
051440 020124 046103 040505
051446 020122 054502 046440
051454 053117 047111 020107
051462 047111 040440 046114
051470 047440 042516 000123
051476 047514 042101 047111  EM6:    .ASCIZ  /LOADING RHER1 FOR ALL UNITS DID NOT SET ANY RHAS BITS.
051504 020107 044122 051105
051512 020061 047506 020122
051520 046101 020114 047125
```

```
 5174  051526  052111  020123  044504
 5175  051534  020104  047516  020124
 5176  051542  042523  020124  047101
 5177  051550  020131  044122  051501
 5178  051556  041040  052111  000123
 5179  051564  047516  020116  054105  EM7:    .ASCIZ  /NON EXISTENT REGISTER, PROGRAM ABORTED./
 5180  051572  051511  042524  052116
 5181  051600  051040  043505  051511
 5182  051606  042524  026122  050040
 5183  051614  047522  051107  046501
 5184  051622  040440  047502  052122
 5185  051630  042105  000056
 5186
 5187  051634  052123  050117  042520  EM10:   .ASCIZ  /STOPPED DRIVE HAS MOL BIT IN RHDS1 SET/
 5188  051642  020104  051104  053111
 5189  051650  020105  040510  020123
 5190  051656  047515  020114  044502
 5191  051664  020124  047111  051040
 5192  051672  042110  030523  051440
 5193  051700  052105  000
 5194  051703  127  052111  020110  EM11:   .ASCIZ  /WITH SPINDLE POWERED DOWN RHCS2 SHOULD ONLY HAVE UNIT NO. AND IR SET/
 5195  051710  050123  047111  046104
 5196  051716  020105  047520  042527
 5197  051724  042522  020104  047504
 5198  051732  047127  051040  041510
 5199  051740  031123  051440  047510
 5200  051746  046125  020104  047117
 5201  051754  054514  044040  053101
 5202  051762  020105  047125  052111
 5203  051770  047040  027117  040440
 5204  051776  042116  044440  020122
 5205  052004  042523  000124
 5206  052010  043101  042524  020122  EM12:   .ASCIZ  /AFTER SPINDLE POWERED UP, NO PACK ACKN. RHDS1 SHOULD HAVE MOL=1, VV=0/
 5207  052016  050123  047111  046104
 5208  052024  020105  047520  042527
 5209  052032  042522  020104  050125
 5210  052040  020054  047516  050040
 5211  052046  041501  020113  041501
 5212  052054  047113  020056  044122
 5213  052062  051504  020061  044123
 5214  052070  052517  042114  044040
 5215  052076  053101  020105  047515
 5216  052104  036514  026061  053040
 5217  052112  036526  000060
 5218  052116  044527  044124  051440  EM13:   .ASCIZ  /WITH SPINDLE POWERED UP, NO INTIALIZE, RHCS1 SHOULD HAVE GO=0, DVA=1, R
 5219  052124  044520  042116  042514
 5220  052132  050040  053517  051105
 5221  052140  042105  052440  026120
 5222  052146  047040  020117  047111
 5223  052154  044524  046101  055111
 5224  052162  026105  051040  041510
 5225  052170  030523  051440  047510
 5226  052176  046125  020104  040510
 5227  052204  042526  043440  036517
 5228  052212  026060  042040  040526
 5229  052220  030475  020054  042122
```

```
5230  052226  036531  026061  044440
5231  052234  036505  000060
5232  052240  043101  042524  020122  EM14:   .ASCIZ  /AFTER SPINDLE POWERED UP, RHCC SHOULD BE=0/
5233  052246  050123  047111  046104
5234  052254  020105  047520  042527
5235  052262  042522  020104  050125
5236  052270  020054  044122  041503
5237  052276  051440  047510  046125
5238  052304  020104  042502  030075
5239  052312     000
5240  052313     120  041501  020113  EM15:   .ASCII  /PACK ACKNOWLEDGE COMMAND CAUSED AN ERROR/<15><12>
5241  052320  041501  047113  053517
5242  052326  042514  043504  020105
5243  052334  047503  046515  047101
5244  052342  020104  040503  051525
5245  052350  042105  040440  020116
5246  052356  051105  047522  006522
5247  052364     012
5248  052365     107  047517  020104          .ASCIZ  /GOOD DATA IS BEFORE COMMAND, REC DATA IS AFTER COMMAND/
5249  052372  040504  040524  044440
5250  052400  020123  042502  047506
5251  052406  042522  041440  046517
5252  052414  040515  042116  020054
5253  052422  042522  020103  040504
5254  052430  040524  044440  020123
5255  052436  043101  042524  020122
5256  052444  047503  046515  047101
5257  052452  000104
5258  052454  047516  047455  020120  EM16:   .ASCII  /NO-OP COMMAND CAUSED AN ERROR/<15><12>
5259  052462  047503  046515  047101
5260  052470  020104  040503  051525
5261  052476  042105  040440  020116
5262  052504  051105  047522  006522
5263  052512     012
5264  052513     107  047517  020104          .ASCIZ  /GOOD DATA IS BEFORE COMMAND, REC DATA IS AFTER COMMAND/
5265  052520  040504  040524  044440
5266  052526  020123  042502  047506
5267  052534  042522  041440  046517
5268  052542  040515  042116  020054
5269  052550  042522  020103  040504
5270  052556  040524  044440  020123
5271  052564  043101  042524  020122
5272  052572  047503  046515  047101
5273  052600  000104
5274  052602  051104  053111  020105  EM17:   .ASCII  /DRIVE CLEAR COMMAND CAUSED AN ERROR/<15><12>
5275  052610  046103  040505  020122
5276  052616  047503  046515  047101
5277  052624  020104  040503  051525
5278  052632  042105  040440  020116
5279  052640  051105  047522  006522
5280  052646     012
5281  052647     107  047517  020104          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES AFTER COMMAND
5282  052654  040504  040524  043440
5283  052662  053111  051505  051440
5284  052670  047510  046125  020104
5285  052676  042502  020054  042522
```

```
5286  052704  020103  040504  040524
5287  052712  043440  053111  051505
5288  052720  040440  052106  051105
5289  052726  041440  046517  040515
5290  052734  042116     000
5291
5292  052737     122  040505  026504  EM20:   .ASCII  /READ-IN COMMAND CAUSED AN ERROR/<15><12>
5293  052744  047111  041440  046517
5294  052752  040515  042116  041440
5295  052760  052501  042523  020104
5296  052766  047101  042440  051122
5297  052774  051117  005015
5298  053000  047507  042117  042040          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5299  053006  052101  020101  044507
5300  053014  042526  020123  044123
5301  053022  052517  042114  041040
5302  053030  026105  051040  041505
5303  053036  042040  052101  020101
5304  053044  044507  042526  020123
5305  053052  042522  027107  041440
5306  053060  047117  027124  040440
5307  053066  052106  051105  041440
5308  053074  046517  040515  042116
5309  053102     000
5310  053103     122  041510  030523  EM21:   .ASCIZ  /RHCS1 CONTENTS DURING COMMAND WAS IN ERROR/
5311  053110  041440  047117  042524
5312  053116  052116  020123  052504
5313  053124  044522  043516  041440
5314  053132  046517  040515  042116
5315  053140  053440  051501  044440
5316  053146  020116  051105  047522
5317  053154  000122
5318  053156  044122  051504  020061  EM22:   .ASCIZ  /RHDS1 CONTENTS DURING COMMAND WAS IN ERROR/
5319  053164  047503  052116  047105
5320  053172  051524  042040  051125
5321  053200  047111  020107  047503
5322  053206  046515  047101  020104
5323  053214  040527  020123  047111
5324  053222  042440  051122  051117
5325  053230     000
5326  053231     125  046116  040517  EM23:   .ASCII  /UNLOAD COMMAND CAUSED AN ERROR/<15><12>
5327  053236  020104  047503  046515
5328  053244  047101  020104  040503
5329  053252  051525  042105  040440
5330  053260  020116  051105  047522
5331  053266  006522     012
5332  053271     107  047517  020104          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5333  053276  040504  040524  043440
5334  053304  053111  051505  051440
5335  053312  047510  046125  020104
5336  053320  042502  020054  042522
5337  053326  020103  040504  040524
5338  053334  043440  053111  051505
5339  053342  051040  043505  020056
5340  053350  047503  052116  020056
5341  053356  043101  042524  020122
```

```
5342  053364  047503  046515  047101
5343  053372  000104
5344  053374  043117  051506  052105  EM24:   .ASCII  /OFFSET COMMAND CAUSED AN ERROR/<15><12>
5345  053402  041440  046517  040515
5346  053410  042116  041440  052501
5347  053416  042523  020104  047101
5348  053424  042440  051122  051117
5349  053432  005015
5350  053434  047507  042117  042040          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5351  053442  052101  020101  044507
5352  053450  042526  020123  044123
5353  053456  052517  042114  041040
5354  053464  026105  051040  041505
5355  053472  042040  052101  020101
5356  053500  044507  042526  020123
5357  053506  042522  027107  041440
5358  053514  047117  027124  040440
5359  053522  052106  051105  041440
5360  053530  046517  040515  042116
5361  053536     000
5362  053537     122  052105  051125  EM25:   .ASCII  /RETURN TO CENTER LINE COMMAND CAUSED AN ERROR/<15><12>
5363  053544  020116  047524  041440
5364  053552  047105  042524  020122
5365  053560  044514  042516  041440
5366  053566  046517  040515  042116
5367  053574  041440  052501  042523
5368  053602  020104  047101  042440
5369  053610  051122  051117  005015
5370  053616  047507  042117  042040          .ASCIZ  /GOOD DATA GIVES SHOULD BE, REC DATA GIVES REG. CONT. AFTER COMMAND/
5371  053624  052101  020101  044507
5372  053632  042526  020123  044123
5373  053640  052517  042114  041040
5374  053646  026105  051040  041505
5375  053654  042040  052101  020101
5376  053662  044507  042526  020123
5377  053670  042522  027107  041440
5378  053676  047117  027124  040440
5379  053704  052106  051105  041440
5380  053712  046517  040515  042116
5381  053720     000
5382  053721     065  030060  047440  EM26:   .ASCIZ  /500 OFFSET COMMANDS ONE AFTER THE OTHER CAUSED AN ERROR/
5383  053726  043106  042523  020124
5384  053734  047503  046515  047101
5385  053742  051504  047440  042516
5386  053750  040440  052106  051105
5387  053756  052040  042510  047440
5388  053764  044124  051105  041440
5389  053772  052501  042523  020104
5390  054000  047101  042440  051122
5391  054006  051117     000
5392  054011     127  044522  042524  EM27.   .ASCII  /WRITE HEADER AND DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5393  054016  044040  040505  042504
5394  054024  020122  047101  020104
5395  054032  040504  040524  041440
5396  054040  052501  042523  020104
5397  054046  046511  051120  050117
```

```
5398  054054  051105  051040  043505
5399  054062  051511  042524  020122
5400  054070  044103  047101  042507
5401  054076  005015
5402  054100  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5403  054106  052101  020101  044507
5404  054114  042526  020123  044127
5405  054122  052101  051440  047510
5406  054130  046125  020104  042502
5407  054136  052040  042510  042522
5408  054144  005015
5409  054146  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5410  054154  042105  042040  052101
5411  054162  020101  044507  042526
5412  054170  020123  044127  052101
5413  054176  053440  051501  052040
5414  054204  042510  042522  040440
5415  054212  052106  051105  041440
5416  054220  046517  040515  042116
5417  054226     000
5418
5419  054227     127  044522  042524  EM30:   .ASCIZ  /WRITE HEADER AND DATA CHANGED WRITE FROM BUFFER/
5420  054234  044040  040505  042504
5421  054242  020122  047101  020104
5422  054250  040504  040524  041440
5423  054256  040510  043516  042105
5424  054264  053440  044522  042524
5425  054272  043040  047522  020115
5426  054300  052502  043106  051105
5427  054306     000
5428  054307     122  040505  020104  EM31:   .ASCII  /READ HEADER AND DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5429  054314  042510  042101  051105
5430  054322  040440  042116  042040
5431  054330  052101  020101  040503
5432  054336  051525  042105  044440
5433  054344  050115  047522  042520
5434  054352  020122  042522  044507
5435  054360  052123  051105  041440
5436  054366  040510  043516  006505
5437  054374     012
5438  054375     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5439  054402  040504  040524  043440
5440  054410  053111  051505  053440
5441  054416  040510  020124  044123
5442  054424  052517  042114  041040
5443  054432  020105  044124  051105
5444  054440  006505     012
5445  054443     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5446  054450  042526  020104  040504
5447  054456  040524  043440  053111
5448  054464  051505  053440  040510
5449  054472  020124  040527  020123
5450  054500  044124  051105  020105
5451  054506  043101  042524  020122
5452  054514  047503  046515  047101
5453  054522  000104
```

```
5454  054524  051127  052111  020105  EM32:   .ASCIZ  /WRITE HEADER DATA, FOLLOWED BY READ HEADER AND DATA, CAUSED DATA ERROR/
5455  054532  042510  042101  051105
5456  054540  042040  052101  026101
5457  054546  043040  046117  047514
5458  054554  042527  020104  054502
5459  054562  051040  040505  020104
5460  054570  042510  042101  051105
5461  054576  040440  042116  042040
5462  054604  052101  026101  041440
5463  054612  052501  042523  020104
5464  054620  040504  040524  042440
5465  054626  051122  051117     000
5466  054633     122  040505  020104  EM33:   .ASCII  /READ DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
5467  054640  040504  040524  041440
5468  054646  052501  042523  020104
5469  054654  046511  051120  050117
5470  054662  051105  051040  043505
5471  054670  051511  042524  020122
5472  054676  044103  047101  042507
5473  054704  005015
5474  054706  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5475  054714  052101  020101  044507
5476  054722  042526  020123  044127
5477  054730  052101  051440  047510
5478  054736  046125  020104  042502
5479  054744  051040  042510  042522
5480  054752  005015
5481  054754  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
5482  054762  042105  042040  052101
5483  054770  020101  044507  042526
5484  054776  020123  044127  052101
5485  055004  053440  051501  052040
5486  055012  042510  042522  040440
5487  055020  052106  051105  041440
5488  055026  046517  040515  042116
5489  055034     000
5490  055035     122  040505  020104  EM34:   .ASCIZ  /READ DATA INCORRECT/
5491  055042  040504  040524  044440
5492  055050  041516  051117  042522
5493  055056  052103     000
5494  055061     127  044522  042524  EM35:   .ASCII  /WRITE DATA COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
5495  055066  042040  052101  020101
5496  055074  047503  046515  047101
5497  055102  020104  040503  051525
5498  055110  042105  044440  050115
5499  055116  047522  042520  020122
5500  055124  042522  044507  052123
5501  055132  051105  041440  040510
5502  055140  043516  006505     012
5503  055145     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5504  055152  040504  040524  043440
5505  055160  053111  051505  053440
5506  055166  040510  020124  044123
5507  055174  052517  042114  041040
5508  055202  020105  044124  051105
5509  055210  006505     012
```

```
5510  055213     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
5511  055220  042526  020104  040504
5512  055226  040524  043440  053111
5513  055234  051505  051040  043505
5514  055242  051511  042524  020122
5515  055250  047503  052116  047105
5516  055256  051524  040440  052106
5517  055264  051105  041440  046517
5518  055272  040515  042116     000
5519  055277     127  044522  042524  EM36:   .ASCIZ  /WRITE DATA COMMAND CHANGED WRITE FROM BUFFER/
5520  055304  042040  052101  020101
5521  055312  047503  046515  047101
5522  055320  020104  044103  047101
5523  055326  042507  020104  051127
5524  055334  052111  020105  051106
5525  055342  046517  041040  043125
5526  055350  042506  000122
5527  055354  042523  045505  041440  EM37:   .ASCII  /SEEK COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
5528  055362  046517  040515  042116
5529  055370  041440  052501  042523
5530  055376  020104  046511  051120
5531  055404  050117  051105  051040
5532  055412  043505  051511  042524
5533  055420  020122  044103  047101
5534  055426  042507  005015
5535  055432  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5536  055440  052101  020101  044507
5537  055446  042526  020123  044127
5538  055454  052101  051440  047510
5539  055462  046125  020104  042502
5540  055470  052040  042510  042522
5541  055476  005015
5542  055500  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER SEEK COMMAND/
5543  055506  042105  042040  052101
5544  055514  020101  044507  042526
5545  055522  020123  042522  044507
5546  055530  052123  051105  041440
5547  055536  047117  042524  052116
5548  055544  020123  043101  042524
5549  055552  020122  042523  045505
5550  055560  041440  046517  040515
5551  055566  042116     000
5552
5553  055571     127  044522  042524  EM40:   .ASCII  /WRITE CHECK CAUSED IMPROPER REGISTER CHANGE/<15><12>
5554  055576  041440  042510  045503
5555  055604  041440  052501  042523
5556  055612  020104  046511  051120
5557  055620  050117  051105  051040
5558  055626  043505  051511  042524
5559  055634  020122  044103  047101
5560  055642  042507  005015
5561  055646  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5562  055654  052101  020101  044507
5563  055662  042526  020123  044127
5564  055670  052101  051440  047510
5565  055676  046125  020104  042502
```

```
5566  055704  052040  042510  042522
5567  055712  005015
5568  055714  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
5569  055722  042105  042040  052101
5570  055730  020101  044507  042526
5571  055736  020123  042522  044507
5572  055744  052123  051105  041440
5573  055752  047117  042524  052116
5574  055760  020123  043101  042524
5575  055766  020122  047503  046515
5576  055774  047101  000104
5577  056000  047514  045503  047111  EM41:   .ASCII  /LOCKING OUT WRITE BY WRITE LOCK BUTTON CAUSED IMPROPER REGISTER CHANGE/
5578  056006  020107  052517  020124
5579  056014  051127  052111  020105
5580  056022  054502  053440  044522
5581  056030  042524  046040  041517
5582  056036  020113  052502  052124
5583  056044  047117  041440  052501
5584  056052  042523  020104  046511
5585  056060  051120  050117  051105
5586  056066  051040  043505  051511
5587  056074  042524  020122  044103
5588  056102  047101  042507  005015
5589  056110  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5590  056116  052101  020101  044507
5591  056124  042526  020123  044127
5592  056132  052101  051440  047510
5593  056140  046125  020104  042502
5594  056146  052040  042510  042522
5595  056154  005015
5596  056156  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER WRITES WERE LOCKED OUT/
5597  056164  042105  042040  052101
5598  056172  020101  044507  042526
5599  056200  020123  042522  044507
5600  056206  052123  051105  041440
5601  056214  047117  042524  052116
5602  056222  020123  043101  042524
5603  056230  020122  051127  052111
5604  056236  051505  053440  051105
5605  056244  020105  047514  045503
5606  056252  042105  047440  052125
5607  056260     000
5608  056261     101  052124  046505  EM42:   .ASCII  /ATTEMPTING TO WRITE WITH WRITES LOCKED OUT CAUSED IMPROPER REGISTER CHA
5609  056266  052120  047111  020107
5610  056274  047524  053440  044522
5611  056302  042524  053440  052111
5612  056310  020110  051127  052111
5613  056316  051505  046040  041517
5614  056324  042513  020104  052517
5615  056332  020124  040503  051525
5616  056340  042105  044440  050115
5617  056346  047522  042520  020122
5618  056354  042522  044507  052123
5619  056362  051105  041440  040510
5620  056370  043516  006505     012
5621  056375     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
```

```
5622  056402  040504  040524  043440
5623  056410  053111  051505  053440
5624  056416  040510  020124  044123
5625  056424  040517  042114  041040
5626  056430  020105  044124  051105
5627  056436  006505      012
5628  056443      122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED WRITE/
5629  056450  040524  020104  040504
5630  056456  040524  043440  053111
5631  056464  051505  051040  043505
5632  056472  051511  042524  020122
5633  056500  047503  052116  047105
5634  056506  051524  040440  052106
5635  056514  051105  040440  052124
5636  056522  046505  052120  042105
5637  056530  053440  044522  042524
5638  056536      000
5639  056537      127  044522  044524  EM43:   .ASCII  /WRITING WITH WRITES LOCKED OUT CHANGED DISK DATA/<15><12>
5640  056544  043516  053440  052111
5641  056552  020110  051127  052111
5642  056560  051505  046040  041517
5643  056566  042513  020104  052517
5644  056574  020124  044103  047101
5645  056602  042507  020104  044504
5646  056610  045523  042040  052101
5647  056616  006501      012
5648  056621      107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT WAS ON DISK BEFORE WRITE (WITH WRITE LOCKED OUT)/<
5649  056626  040504  040524  043440
5650  056634  053111  051505  053440
5651  056642  040510  020124  040527
5652  056650  020123  047117  042040
5653  056656  051511  020113  042502
5654  056664  047506  042522  053440
5655  056672  044522  042524  020040
5656  056700  044527  044124  053440
5657  056706  044522  042524  046040
5658  056714  041517  042513  020104
5659  056722  052517  024524  005015
5660  056730  040527  020123  052101          .ASCII  /WAS ATTEMPTED/<15><12>
5661  056736  042524  050115  042524
5662  056744  006504      012
5663  056747      122  041505  044505          .ASCII  /RECEIVED DATA GIVES WHAT WAS READ BACK AFTER WRITE/<15><12>
5664  056754  042526  020104  040504
5665  056762  040524  043440  053111
5666  056770  051505  053440  040510
5667  056776  020124  040527  020123
5668  057004  042522  042101  041040
5669  057012  041501  020113  043101
5670  057020  042524  020122  051127
5671  057026  052111  006505      012
5672  057033      050  044527  044124          .ASCIZ  /(WITH WRITE LOCKED OUT) WAS ATTEMPTED/
5673  057040  053440  044522  042524
5674  057046  046040  041517  042513
5675  057054  020104  052517  024524
5676  057062  053440  051501  040440
5677  057070  052124  046505  052120
```

```
5678  057076  042105  000
5679  057101  105  040516  046102  EM44:   .ASCII  /ENABLING WRITES BY WRITE LOCK BUTTON CAUSED IMPROPER REGISTER CHANGE/<1
5680  057106  047111  020107  051127
5681  057114  052111  051505  041040
5682  057122  020131  051127  052111
5683  057130  020105  047514  045503
5684  057136  041040  052125  047524
5685  057144  020116  040503  051525
5686  057152  042105  044440  050115
5687  057160  047522  042520  020122
5688  057166  042522  044507  052123
5689  057174  051105  041440  040510
5690  057202  043616  006505  012
5691  057207  107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5692  057215  040504  040524  043440
5693  057223  053111  051505  053440
5694  057231  040510  020124  044123
5695  057237  042517  042114  041040
5696  057245  020105  044124  051105
5697  057253  043616  006505  012
5698  057260  122  041505  044505          .ASCII  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER WRITE LOCK BUTTON/<15><12>
5699  057266  042522  020104  040504
5700  057274  040524  043440  053111
5701  057302  051505  051040  043505
5702  057310  051511  042524  020122
5703  057316  047503  052116  047105
5704  057324  051524  040440  052106
5705  057332  051105  053440  044522
5706  057340  042524  046040  041517
5707  057346  020113  052502  052124
5708  057354  047117  005015
5709  057360  047105  041101  042514          .ASCIZ  /ENABLED WRITES/
5710  057366  020104  051127  052111
5711  057374  051505  000
5712  057377  124  040522  051516  EM45:   .ASCII  /TRANSFERRING ON LAST BLOCK - CYLINDER 410, SECTOR 21, TRACK 18/<15><12>
5713  057404  042506  051122  047111
5714  057412  020107  047117  046040
5715  057420  051501  020124  046102
5716  057426  041517  020113  020055
5717  057434  054503  044514  042116
5718  057442  051105  032040  030061
5719  057450  020054  042523  052103
5720  057456  051117  031040  026061
5721  057464  052040  040522  045503
5722  057472  030440  006470  012
5723  057477  103  052501  042523          .ASCII  /CAUSED IMPROPER REGISTER CHANGE/<15><12>
5724  057504  020104  046511  051120
5725  057512  050117  051105  051040
5726  057520  043505  051511  042524
5727  057526  020122  044103  047101
5728  057534  042507  005015
5729  057540  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5730  057546  052101  020101  044507
5731  057554  042526  020123  044127
5732  057562  052101  051440  047510
5733  057570  046125  020104  042502
```

```
5734  057572  052040  042510  042522
5735  057600  005015
5736  057602  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER TRANSFER/
5737  057610  042105  042040  052101
5738  057616  020101  044507  042526
5739  057624  020123  042522  044507
5740  057632  052123  051105  041440
5741  057640  047117  042524  052116
5742  057646  020123  043101  042524
5743  057654  020122  051124  047101
5744  057662  043123  051105     000
5745  057667     104  052101  020101  EM46:   .ASCII  /DATA READ FROM LAST BLOCK - CYLINDER 410, SECTOR 21, TRACK 18/<15><12>
5746  057674  042522  042101  043040
5747  057702  047522  020115  040514
5748  057710  052123  041040  047514
5749  057716  045503  026440  041440
5750  057724  046131  047111  042504
5751  057732  020122  030464  026060
5752  057740  051440  041505  047524
5753  057746  020122  030462  020054
5754  057754  051124  041501  020113
5755  057762  034061  005015
5756  057766  051511  044440  020116          .ASCIZ  /IS IN ERROR/
5757  057774  051105  047522  000122
5758  060002  051124  047101  043123  EM47:   .ASCII  /TRANSFERRING DATA FROM NONEXISTANT SECTOR CAUSED IMPROPER /<15><12>
5759  060010  051105  044522  043516
5760  060016  042040  052101  020101
5761  060024  051106  046517  047040
5762  060032  047117  054105  051511
5763  060040  040524  052116  051440
5764  060046  041505  047524  020122
5765  060054  040503  051525  042105
5766  060062  044440  050115  047522
5767  060070  042520  020122  005015
5768  060076  042522  044507  052123          .ASCII  /REGISTER CHANGE, GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5769  060104  051105  041440  040510
5770  060112  043516  026105  043440
5771  060120  047517  020104  040504
5772  060126  040524  043440  053111
5773  060134  051505  053440  040510
5774  060142  020124  044123  052517
5775  060150  042114  041040  020105
5776  060156  044124  051105  006505
5777  060164     012
5778  060165     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED TRANSFER/
5779  060172  042526  020104  040504
5780  060200  040524  043440  053111
5781  060206  051505  051040  043505
5782  060214  051511  042524  020122
5783  060222  047503  052116  047105
5784  060230  051524  040440  052106
5785  060236  051105  040440  052124
5786  060244  046505  052120  042105
5787  060252  052040  040522  051516
5788  060260  042506  000122
5789
```

```
 5790  060264  051124  047101  043123  EM50:   .ASCII  /TRANSFERRING FROM NONEXISTANT SECTOR CAUSED DATA ERROR/<15><12>
 5791  060272  051105  044522  043516
 5792  060300  043040  047522  020115
 5793  060306  047516  042516  044530
 5794  060314  052123  047101  020124
 5795  060322  042523  052103  051117
 5796  060330  041440  052501  042523
 5797  060336  020104  040504  040524
 5798  060344  042440  051122  051117
 5799  060352  005015
 5800  060354  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
 5801  060362  052101  020101  044507
 5802  060370  042526  020123  044127
 5803  060376  052101  051440  047510
 5804  060404  046125  020104  042502
 5805  060412  052040  042510  042522
 5806  060420  005015
 5807  060422  040502  020104  040504          .ASCIZ  /BAD DATA GIVES WHAT WAS IN BUFFER AFTER TRANSFER/
 5808  060430  040524  043440  053111
 5809  060436  051505  053440  040510
 5810  060444  020124  040527  020123
 5811  060452  047111  041040  043125
 5812  060460  042506  020122  043101
 5813  060466  042524  020122  051124
 5814  060474  047101  043123  051105
 5815  060502     000
 5816  060503     107  053111  047111  EM51:   .ASCII  /GIVING ILLEGAL FUNCTION CAUSED IMPROPER REGISTER CHANGE/<15><12>
 5817  060510  020107  046111  042514
 5818  060516  040507  020114  052506
 5819  060524  041516  044524  047117
 5820  060532  041440  052501  042523
 5821  060540  020104  046511  051120
 5822  060546  050117  051105  051040
 5823  060554  043505  051511  042524
 5824  060562  020122  044103  047101
 5825  060570  042507  005015
 5826  060574  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
 5827  060602  052101  020101  044507
 5828  060610  042526  020123  044127
 5829  060616  052101  051440  047510
 5830  060624  046125  020104  042502
 5831  060632  052040  042510  042522
 5832  060640  005015
 5833  060642  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ILLEGAL FUNCTION IS GIVEN/
 5834  060650  042105  042040  052101
 5835  060656  020101  044507  042526
 5836  060664  020123  042522  044507
 5837  060672  052123  051105  041440
 5838  060700  047117  042524  052116
 5839  060706  020123  043101  042524
 5840  060714  020122  046111  042514
 5841  060722  040507  020114  052506
 5842  060730  041516  044524  047117
 5843  060736  044440  020123  044507
 5844  060744  042526  000116
 5845  060750  051127  052111  020105  EM52:   .ASCII  /WRITE DATA COMMAND ON NONEXISTANT SECTOR CAUSED IMPROPER REG. CHANGE/<1
```

```
5846  060756  040504  040524  041440
5847  060764  046517  040515  042116
5848  060772  047440  020116  047516
5849  061000  042516  044530  052123
5850  061006  047101  020124  042523
5851  061014  052103  051117  041440
5852  061022  052501  042523  020104
5853  061030  046511  051120  050117
5854  061036  051105  051040  043505
5855  061044  020056  044103  047101
5856  061052  042507  005015
5857  061056  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5858  061064  052101  020101  044507
5859  061072  042526  020123  044127
5860  061100  052101  051440  047510
5861  061106  046125  020104  042502
5862  061114  052040  042510  042522
5863  061122  005015
5864  061124  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED WRITE DATA/
5865  061132  042105  042040  052101
5866  061140  020101  044507  042526
5867  061146  020123  042522  044507
5868  061154  052123  051105  041440
5869  061162  047117  042524  052116
5870  061170  020123  043101  042524
5871  061176  020122  052101  042524
5872  061204  050115  042524  020104
5873  061212  051127  052111  020105
5874  061220  040504  040524     000
5875  061225     122  040505  020104  EM53:   .ASCIZ  /READ HEADER AND DATA AFTER A SEARCH CAUSED DATA ERROR/
5876  061232  042510  042101  051105
5877  061240  040440  042116  042040
5878  061246  052101  020101  043101
5879  061254  042524  020122  020101
5880  061262  042523  051101  044103
5881  061270  041440  052501  042523
5882  061276  020104  040504  040524
5883  061304  042440  051122  051117
5884  061312     000
5885  061313     101  052124  046505  EM54:   .ASCII  /ATTEMPTING COMMAND WITH INVALID ADDRESS CAUSED IMPROPER REGISTER CHANGE
5886  061320  052120  047111  020107
5887  061326  047503  046515  047101
5888  061334  020104  044527  044124
5889  061342  044440  053116  046101
5890  061350  042111  040440  042104
5891  061356  042522  051523  041440
5892  061364  052501  042523  020104
5893  061372  046511  051120  050117
5894  061400  051105  051040  043505
5895  061406  051511  042524  020122
5896  061414  044103  047101  042507
5897  061422  005015
5898  061424  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5899  061432  052101  020101  044507
5900  061440  042526  020123  044127
5901  061446  052101  051440  047510
```

```
5902  061454  046125  020104  042502
5903  061462  052040  042510  042522
5904  061470  005015
5905  061472  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION/
5906  061500  042105  042040  052101
5907  061506  020101  044507  042526
5908  061514  020123  042522  044507
5909  061522  052123  051105  041440
5910  061530  047117  042524  052116
5911  061536  020123  043101  042524
5912  061544  020122  050117  051105
5913  061552  052101  047511  000116
5914  061560  051127  052111  047111  EM55:   .ASCII  /WRITING OR READING WITH EXPECTED ADDRESS OVERFLOW ERROR/<15><12>
5915  061566  020107  051117  051040
5916  061574  040505  044504  043516
5917  061602  053440  052111  020110
5918  061610  054105  042520  052103
5919  061616  042105  040440  042104
5920  061624  042522  051523  047440
5921  061632  042526  043122  047514
5922  061640  020127  051105  047522
5923  061646  006522     012
5924  061651     103  052501  042523          .ASCII  /CAUSED IMPROPER REGISTER CHANGE/<15><12>
5925  061656  020104  046511  051120
5926  061664  050117  051105  051040
5927  061672  043505  051511  042524
5928  061700  020122  044103  047101
5929  061706  042507  005015
5930  061712  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5931  061720  052101  020101  044507
5932  061726  042526  020123  044127
5933  061734  052101  051440  047510
5934  061742  046125  020104  042502
5935  061750  052040  042510  042522
5936  061756  005015
5937  061760  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION/
5938  061766  042105  042040  052101
5939  061774  020101  044507  042526
5940  062002  020123  042522  044507
5941  062010  052123  051105  041440
5942  062016  047117  042524  052116
5943  062024  020123  043101  042524
5944  062032  020122  050117  051105
5945  062040  052101  047511  000116
5946  062046  040504  040524  051040  EM56:   .ASCII  /DATA READ WITH AN EXPECTED ADDRESS OVERFLOW ERROR IS INCORRECT/<15><12>
5947  062054  040505  020104  044527
5948  062062  044124  040440  020116
5949  062070  054105  042520  052103
5950  062076  042105  040440  042104
5951  062104  042522  051523  047440
5952  062112  042526  043122  047514
5953  062120  020127  051105  047522
5954  062126  020122  051511  044440
5955  062134  041516  051117  042522
5956  062142  052103  005015
5957  062146  047527  042122  047040          .ASCII  /WORD NO. 1 TO 260 SHOULD BE READ, WORD NO 261 TO 266 SHOULD/<15><12>
```

```
5958  062154  027117  030440  052040
5959  062162  020117  033062  020060
5960  062170  044123  052517  042114
5961  062176  041040  020105  042522
5962  062204  042101  020054  047527
5963  062212  042122  047040  020117
5964  062220  033062  020061  047524
5965  062226  031040  033066  051440
5966  062234  047510  046125  006504
5967  062242     012
5968  062243     102  020105  044103          .ASCIZ  /BE CHANGED/
5969  062250  047101  042507  000104
5970  062256  052101  042524  050115  EM57:   .ASCII  /ATTEMPTING DATA COMMAND WITH WRONG FORMAT BIT CAUSED/<15><12>
5971  062264  044524  043516  042040
5972  062272  052101  020101  047503
5973  062300  046515  047101  020104
5974  062306  044527  044124  053440
5975  062314  047522  043516  043040
5976  062322  051117  040515  020124
5977  062330  044502  020124  040503
5978  062336  051525  042105  005015
5979  062344  046511  051120  050117          .ASCII  /IMPROPER REGISTER CHANGE/<15><12>
5980  062352  051105  051040  043505
5981  062360  051511  042524  020122
5982  062366  044103  047101  042507
5983  062374  005015
5984  062376  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
5985  062404  052101  020101  044507
5986  062412  042526  020123  044127
5987  062420  052101  051440  047510
5988  062426  046125  020104  042502
5989  062434  052040  042510  042522
5990  062442  005015
5991  062444  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER ATTEMPTED DATA TRANSFER/
5992  062452  042105  042040  052101
5993  062460  020101  044507  042526
5994  062466  020123  042522  044507
5995  062474  052123  051105  041440
5996  062502  047117  042524  052116
5997  062510  020123  043101  042524
5998  062516  020122  052101  042524
5999  062524  050115  042524  020104
6000  062532  040504  040524  052040
6001  062540  040522  051516  042506
6002  062546  000122
6003
6004  062550  052101  042524  050115  EM60:   .ASCII  /ATTEMPTING TO MODIFY REGISTER DURING AN OPERATION CAUSED IMPROPER <15><
6005  062556  044524  043516  052040
6006  062564  020117  047515  044504
6007  062572  054506  051040  043505
6008  062600  051511  042524  020122
6009  062606  052504  044522  043516
6010  062614  040440  020116  050117
6011  062622  051105  052101  047511
6012  062630  020116  040503  051525
6013  062636  042105  044440  050115
```

```
6014  062644  047522  042520  006522
6015  062652     012
6016  062653     122  043505  051511          .ASCII  /REGISTER CHANGE. GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
6017  062660  042524  020122  044103
6018  062666  047101  042507  020056
6019  062674  047507  042117  042040
6020  062702  052101  020101  044507
6021  062710  042526  020123  044127
6022  062716  052101  051440  047510
6023  062724  046125  020104  042502
6024  062732  052040  042510  042522
6025  062740  005015
6026  062742  042522  042503  053111          .ASCII  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER OPERATION WAS ATTEMPTED/<15
6027  062750  042105  042040  052101
6028  062756  020101  044507  042526
6029  062764  020123  042522  044507
6030  062772  052123  051105  041440
6031  063000  047117  042524  052116
6032  063006  020123  043101  042524
6033  063014  020122  050117  051105
6034  063022  052101  047511  020116
6035  063030  040527  020123  052101
6036  063036  042524  050115  042524
6037  063044  006504     012
6038  063047     115  042117  043111          .ASCIZ  /MODIFYING REG GIVES ADDRESS OF REGISTER BEING MODIFIED WHICH CAUSED ERR
6039  063054  044531  043516  051040
6040  063062  043505  043440  053111
6041  063070  051505  040440  042104
6042  063076  042522  051523  047440
6043  063104  020106  042522  044507
6044  063112  052123  051105  041040
6045  063120  044505  043516  046440
6046  063126  042117  043111  042511
6047  063134  020104  044127  041511
6048  063142  020110  040503  051525
6049  063150  042105  042440  051122
6050  063156  051117     000
6051  063161     122  041510  030523  EM61:   .ASCIZ  /RHCS1 HAS SOME INCORRECT STATUS BITS = 1, OR = 0/
6052  063166  044040  051501  051440
6053  063174  046517  020105  047111
6054  063202  047503  051122  041505
6055  063210  020124  052123  052101
6056  063216  051525  041040  052111
6057  063224  020123  020075  026061
6058  063232  047440  020122  020075
6059  063240  000060
6060  063242  044122  051504  020061  EM62:   .ASCIZ  /RHDS1 HAS SOME INCORRECT STATUS BITS = 1, OR = 0/
6061  063250  040510  020123  047523
6062  063256  042515  044440  041516
6063  063264  051117  042522  052103
6064  063272  051440  040524  052524
6065  063300  020123  044502  051524
6066  063306  036440  030440  020054
6067  063314  051117  036440  030040
6068  063322     000
6069  063323     122  042110  030523  EM63:   .ASCIZ  /RHDS1 CONTENTS DURING COMMAND WERE IN ERROR.
```

```
6070  063330  041440  047117  042524
6071  063336  052116  020123  052504
6072  063344  044522  043516  041440
6073  063352  046517  040515  042116
6074  063360  053440  051105  020105
6075  063366  047111  042440  051122
6076  063374  051117     000
6077  063377     122  041505  046101  EM64:   .ASCII  /RECALIBRATE COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
6078  063404  041111  040522  042524
6079  063412  041440  046517  040515
6080  063420  042116  041440  052501
6081  063426  042523  020104  046511
6082  063434  051120  050117  051105
6083  063442  051040  043505  051511
6084  063450  042524  020122  044103
6085  063456  047101  042507  005015
6086  063464  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
6087  063472  052101  020101  044507
6088  063500  042526  020123  044127
6089  063506  052101  051440  047510
6090  063514  046125  020104  042502
6091  063522  052040  042510  042522
6092  063530  005015
6093  063532  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
6094  063540  042105  042040  052101
6095  063546  020101  044507  042526
6096  063554  020123  042522  044507
6097  063562  052123  051105  041440
6098  063570  047117  042524  052116
6099  063576  020123  043101  042524
6100  063604  020122  047503  046515
6101  063612  047101  000104
6102  063616  047111  042524  051122  EM65:   .ASCIZ  /INTERRUPT FAILING/
6103  063624  050125  020124  040506
6104  063632  046111  047111  000107
6105  063640  042510  042101  051105  EM66:   .ASCII  /HEADER AND DATA COMMAND FOR HEAD SELECTION TEST CAUSED/<15><12>
6106  063646  040440  042116  042040
6107  063654  052101  020101  047503
6108  063662  046515  047101  020104
6109  063670  047506  020122  042510
6110  063676  042101  051440  046105
6111  063704  041505  044524  047117
6112  063712  052040  051505  020124
6113  063720  040503  051525  042105
6114  063726  005015
6115  063730  051105  047522  020122          .ASCII  /ERROR - RHDST GIVES TRACK BEING WRITTEN OR READ ON CYL 0, SCTR 0/<15><1
6116  063736  020055  044122  051504
6117  063744  020124  044507  042526
6118  063752  020123  051124  041501
6119  063760  020113  042502  047111
6120  063766  020107  051127  052111
6121  063774  042524  020116  051117
6122  064002  051040  040505  020104
6123  064010  047117  041440  046131
6124  064016  030040  020054  041523
6125  064024  051124  030040  005015
```

```
6126  064032  042522  042101  044040  EM67:   .ASCII  /READ HEADER AND DATA ERROR IN HEAD SELECTION TEST./<12><15>
6127  064040  040505  042504  020122
6128  064046  047101  020104  040504
6129  064054  040524  042440  051122
6130  064062  051117  044440  020116
6131  064070  042510  042101  051440
6132  064076  046105  041505  044524
6133  064104  047117  052040  051505
6134  064112  027124  006412
6135  064116  044506  051522  020124          .ASCII  /FIRST FOUR WORD NUMBERS ARE THE HEADER./<12><15>
6136  064124  047506  051125  053440
6137  064132  051117  020104  052516
6138  064140  041115  051105  020123
6139  064146  051101  020105  044124
6140  064154  020105  042510  042101
6141  064162  051105  005056     015
6142  064167     127  051117  020104          .ASCII  /WORD NUMBERS 5 TO 260 ARE DATA WORDS,/<12><15>
6143  064174  052516  041115  051105
6144  064202  020123  020065  047524
6145  064210  031040  030066  040440
6146  064216  042522  042040  052101
6147  064224  020101  047527  042122
6148  064232  026123  006412
6149  064236  047101  020104  047111          .ASCIZ  /AND IN DATA WORDS BITS 4,5,6,7,8 GIVE TRACK NUMBER./
6150  064244  042040  052101  020101
6151  064252  047527  042122  020123
6152  064260  044502  051524  032040
6153  064266  032454  033054  033454
6154  064274  034054  043440  053111
6155  064302  020105  051124  041501
6156  064310  020113  052516  041115
6157  064316  051105  000056
6158
6159  064322  042522  042101  044040  EM70:   .ASCII  /READ HEADER AND DATA ERROR IN/<15><12>
6160  064330  040505  042504  020122
6161  064336  047101  020104  040504
6162  064344  040524  042440  051122
6163  064352  051117  044440  006516
6164  064360     012
6165  064361     104  043111  042506          .ASCII  /DIFFERENCE LINE TEST/<15><12>
6166  064366  042522  041516  020105
6167  064374  044514  042516  052040
6168  064402  051505  006524     012
6169  064407     127  051117  020104          .ASCII  /WORD NOS 1-4 GIVE HEADER/<15><12>
6170  064414  047516  020123  026461
6171  064422  020064  044507  042526
6172  064430  044040  040505  042504
6173  064436  006522     012
6174  064441     127  051117  020104          .ASCIZ  /WORD NOS 5-260 GIVE DATA WHICH IS THE CYLINDER ADDRESS/
6175  064446  047516  020123  026465
6176  064454  033062  020060  044507
6177  064462  042526  042040  052101
6178  064470  020101  044127  041511
6179  064476  020110  051511  052040
6180  064504  042510  041440  046131
6181  064512  047111  042504  020122
```

```
6182  064520  042101  051104  051505
6183  064526  000123
6184  064530  047506  041522  047111  EM71:   .ASCII  /FORCING OPI BY 3 INDEX PULSES/<15><12>
6185  064546  020107  050117  020111
6186  064544  054502  031440  044440
6187  064552  042116  054105  050040
6188  064560  046125  042523  006523
6189  064566     012
6190  064567     103  052501  042523          .ASCII  /CAUSED IMPROPER REGISTER CHANGE/<15><12>
6191  064574  020104  046511  051120
6192  064602  050117  051105  051040
6193  064610  043505  051511  042524
6194  064616  020122  044103  047101
6195  064624  042507  005015
6196  064630  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
6197  064636  052101  020101  044507
6198  064644  042526  020123  044127
6199  064652  052101  051440  047510
6200  064660  046125  020104  042502
6201  064666  052040  042510  042522
6202  064674  005015
6203  064676  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER 3 INDEX PULSES/
6204  064704  042105  042040  052101
6205  064712  020101  044507  042526
6206  064720  020123  042522  044507
6207  064726  052123  051105  041440
6208  064734  047117  042524  052116
6209  064742  020123  043101  042524
6210  064750  020122  020063  047111
6211  064756  042504  020130  052520
6212  064764  051514  051505     000
6213  064771     127  044510  042514  EM72:   .ASCII  /WHILE USING UNIBUS B/<15><12>
6214  064776  052440  044523  043516
6215  065004  052440  044516  052502
6216  065012  020123  006502     012
6217  065017     122  040505  020104          .ASCII  /READ DATA CAUSED IMPROPER REGISTER CHANGE/<15><12>
6218  065024  040504  040524  041440
6219  065032  052501  042523  020104
6220  065040  046511  051120  050117
6221  065046  051105  051040  043505
6222  065054  051511  042524  020122
6223  065062  044103  047101  042507
6224  065070  005015
6225  065072  047507  042117  042040          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
6226  065100  052101  020101  044507
6227  065106  042526  020123  044127
6228  065114  052101  051440  047510
6229  065122  046125  020104  042502
6230  065130  052040  042510  042522
6231  065136  005015
6232  065140  042522  042503  053111          .ASCIZ  /RECEIVED DATA GIVES WHAT WAS THERE AFTER COMMAND/
6233  065146  042105  042040  052101
6234  065154  020101  044507  042526
6235  065162  020123  044127  052101
6236  065170  053440  051501  052040
6237  065176  042510  042522  040440
```

```
  6238  065204  052106  051105  041440
  6239  065212  046517  040515  042116
  6240  065220     000
  6241  065221     127  044510  042514  EM73:   .ASCII  /WHILE USING UNIBUS B/<15><12>
  6242  065236  052440  044523  043516
  6243  065244  052440  044516  052502
  6244  065252  020123  006502     012
  6245  065257     122  044505  020104          .ASCIZ  /READ DATA INCORRECT/
  6246  065264  040504  040524  044440
  6247  065272  041516  051117  042522
  6248  065300  052103     000
  6249  065303     127  044510  042514  EM74:   .ASCII  /WHILE USING UNIBUS B/<15><12>
  6250  065320  052440  044523  043516
  6251  065326  052440  044516  052502
  6252  065334  020123  006502     012
  6253  065341     127  044522  042524          .ASCII  /WRITE DATA COMMAND CAUSED IMPROPER REGISTER CHANGE/<15><12>
  6254  065346  042040  052101  020101
  6255  065354  047503  046515  047101
  6256  065362  020104  040503  051525
  6257  065370  042105  044440  050115
  6258  065376  047522  042520  020122
  6259  065404  042522  044507  052123
  6260  065412  051105  041440  040510
  6261  065420  043516  006505     012
  6262  065425     107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
  6263  065432  040504  040524  043440
  6264  065440  053111  051505  053440
  6265  065446  040510  020124  044123
  6266  065454  052517  042114  041040
  6267  065462  020105  044124  051105
  6268  065470  006505     012
  6269  065473     122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
  6270  065500  042526  020104  040504
  6271  065506  040524  043440  053111
  6272  065514  051505  051040  043505
  6273  065522  051511  042524  020122
  6274  065530  047503  052116  047105
  6275  065536  051524  040440  052106
  6276  065544  051105  041440  046517
  6277  065552  040515  042116     000
  6278  065557     127  044510  042514  EM75:   .ASCII  /WHILE USING UNIBUS B/<15><12>
  6279  065564  052440  044523  043516
  6280  065572  052440  044516  052502
  6281  065600  020123  006502     012
  6282  065605     127  044522  042524          .ASCIZ  /WRITE DATA COMMAND CHANGED WRITE FROM BUFFER/
  6283  065622  042040  052101  020101
  6284  065630  047503  046515  047101
  6285  065636  020104  044103  047101
  6286  065644  042507  020104  051127
  6287  065652  052111  020105  051106
  6288  065660  046517  041040  043125
  6289  065666  042506  000122
  6290  065672  044127  046111  020105  EM76:   .ASCII  /WHILE USING UNIBUS B/<15><12>
  6291  065700  051525  047111  020107
  6292  065706  047125  041111  051525
  6293  065714  041040  005015
```

```
6294  066570  051127  052111  020105          .ASCII  /WRITE CHECK CAUSED IMPROPER REGISTER CHANGE/<15><12>
6295  066576  044103  041505  020113
6296  066704  040503  051525  042105
6297  066712  044440  040115  047522
6298  066720  042440  040122  042522
6299  066726  044507  052123  051105
6300  066734  041440  040510  043516
6301  066742  006505      012
6302  066745      107  047517  020104          .ASCII  /GOOD DATA GIVES WHAT SHOULD BE THERE/<15><12>
6303  066752  040504  040524  043440
6304  066760  053111  051505  053440
6305  066766  040510  020124  044123
6306  065774  052517  042114  041040
6307  066002  020105  044124  051105
6308  066010  006505      012
6309  066013      122  041505  044505          .ASCIZ  /RECEIVED DATA GIVES REGISTER CONTENTS AFTER COMMAND/
6310  066020  042526  020104  040504
6311  066026  040524  043440  053111
6312  066034  051505  051040  043505
6313  066042  051511  042524  020122
6314  066050  047503  052116  047105
6315  066056  051524  040440  052106
6316  066064  051105  041440  046517
6317  066072  040515  042116      000
6318  066077      120  042522  047514  EM77:   .ASCII  /PRELOADING 'RHCC' PRIOR TO DOING NEXT TEST DOES NOT PRODUCE CORRECT RES
6319  066104  042101  047111  020107
6320  066112  051047  041510  023503
6321  066120  020040  044522  051117
6322  066126  052040  020117  047504
6323  066134  047111  020107  042516
6324  066142  052130  052040  051505
6325  066150  020124  047504  051505
6326  066156  047040  052117  050040
6327  066164  047522  052504  042503
6328  066172  041440  051117  042522
6329  066200  052103  051040  051505
6330  066206  046125  006524      012
6331  066213      124  042510  042522          .ASCIZ  /THEREFORE NEXT TEST RESULTS ARE SUSPECT WITH REGARD TO 'RHCC' CONTENTS/
6332  066220  047506  042522  047040
6333  066226  054105  020124  042524
6334  066234  052123  051040  051505
6335  066242  046125  051524  040440
6336  066250  042522  051440  051525
6337  066256  042520  052103  053440
6338  066264  052111  020110  042522
6339  066272  040507  042122  052040
6340  066300  020117  051047  041510
6341  066306  023503  041440  047117
6342  066314  042524  052116  000123
6343
6344  066322  042101  051104  051505  EM100:  .ASCIZ  /ADDRESS PLUG CHANGE RESULTED IN BAD REGISTER DATA/
6345  066330  020123  046120  043525
6346  066336  041440  040510  043516
6347  066344  020105  042522  052523
6348  066352  052114  042105  044440
6349  066360  020116  040502  020104
```

```
6350  066436  042522  044507  052123
6351  066444  051105  042040  052101
6352  066452  040101
6353  066460  047125  052111  042040  EM101:  .ASCIZ  /UNIT DID NOT GO OFFLINE WHEN ADDRESS PLUG REMOVED/
6354  066466  042111  047040  052117
6355  066474  043440  020117  043117
6356  066502  046105  047111  020105
6357  066510  044127  047105  040440
6358  066516  042104  042522  051523
6359  066524  050040  052514  020107
6360  066532  042522  047515  042526
6361  066540  000104
6362  066542  047125  052111  047040  EM102:  .ASCIZ  /UNIT NOT AVAILABLE AFTER ADDRESS PLUG REPLACED/
6363  066550  052117  040440  040526
6364  066556  046111  041101  042514
6365  066564  040440  052106  051105
6366  066572  040440  042104  042522
6367  066600  051523  050040  052514
6368  066606  020107  042522  046120
6369  066614  041501  042105  000     EM103:  .ASCIZ  /REGISTER CONTENTS INCORRECT BEFORE A DIAG MODE SEEK/
6370  066621  122     043505  051511
6371  066626  042524  020122  047503
6372  066634  052116  047105  051524
6373  066642  044440  041516  051117
6374  066650  042522  052103  041040
6375  066656  043105  051117  020105
6376  066664  020101  044504  043501
6377  066672  046440  042117  020105
6378  066700  042523  045505  000
6379  066705  122     043505  051511  EM104:  .ASCIZ  /REGISTER CONTENTS INCORRECT AFTER A DIAG MODE SEEK/
6380  066712  042524  020122  047503
6381  066720  052116  047105  051524
6382  066726  044440  041516  051117
6383  066734  042522  052103  040440
6384  066742  052106  051105  040440
6385  066750  042040  040511  020107
6386  066756  047515  042504  051440
6387  066764  042505  000113
```

```
6388
6389
6390  066714  040506  040524  020114  CPHALT: .ASCII  /FATAL ERROR - SEE DOCUMENT LISTING/<15><12><15><12>
6391  066722  051105  047522  020122
6392  066730  020055  042523  020105
6393  066736  047504  052503  042515
6394  066744  052116  046040  051511
6395  066752  044524  043516  005015
6396  066760  005015
6397  066762  044124  020105  047503          .ASCII  /THE CONTROLLER OR DEVICE HAS GONE OFFLINE, LOST/<15><12>
6398  066770  052116  047522  046114
6399  066776  051105  047440  020122
6400  067004  042504  044526  042503
6401  067012  044040  051501  043440
6402  067020  047117  020105  043117
6403  067026  046106  047111  026105
6404  067034  046040  051517  006524
6405  067042     012
6406  067043     047  042522  042101          .ASCII  /'READY', BECOME UNAVAILABLE, OR HAS STATUS BITS/<15><12>
6407  067050  023531  020054  042502
6408  067056  047503  042515  052440
6409  067064  040516  040526  046111
6410  067072  041101  042514  020054
6411  067100  051117  044040  051501
6412  067106  051440  040524  052524
6413  067114  020123  044502  051524
6414  067122  005015
6415  067124  044127  041511  020110          .ASCIZ  /WHICH CANNOT BE CLEARED/
6416  067132  040503  047116  052117
6417  067140  041040  020105  046103
6418  067146  040505  042522  000104
6419
6420
6421
6422
6423  067154  041520  020040  020040  DH1:    .ASCII  /PC      TEST    WAT     BIT     REG     REG     RHCS1/<15><12>
6424  067162  020040  042524  052123
6425  067170  020040  020040  040527
6426  067176  020124  020040  020040
6427  067204  044502  020124  020040
6428  067212  020040  042522  020107
6429  067220  020040  020040  042522
6430  067226  020107  020040  020040
6431  067234  044122  051503  006461
6432  067242     012
6433  067243     040  020040  020040          .ASCIZ  /       NO      PC      EXPECT  ADDRESS CONTENT CONTENT/
6434  067250  020040  047040  020117
6435  067256  020040  020040  050040
6436  067264  020103  020040  020040
6437  067272  042440  050130  041505
6438  067300  020124  040440  042104
6439  067306  042522  051523  041440
6440  067314  047117  042524  052116
6441  067322  041440  047117  042524
6442  067330  052116     000
6443  067333     120  020103  020040  DH4:    .ASCII  /PC      TEST    WAT     BIT     REG     TIME IN/<15><12>
```

```
6444  067340  020040  052040  051505
6445  067346  020124  020040  053440
6446  067354  052101  020040  020040
6447  067362  041040  052111  020040
6448  067370  020040  051040  043505
6449  067376  020040  020040  052040
6450  067404  046511  020105  047111
6451  067412  005015
6452  067414  020040  020040  020040          .ASCIZ  /       NO      PC      EXPECT  ADDRESS 10 MSEC/
6453  067422  020040  047516  020040
6454  067430  020040  020040  041520
6455  067436  020040  020040  020040
6456  067444  054105  042520  052103
6457  067452  020040  042101  051104
6458  067460  051505  020123  030061
6459  067466  046440  042523  000103
6460  067474  041520  020040  020040  DH5:    .ASCII  /PC     TEST    REG     GOOD    RECEIVED/<15><12>
6461  067502  020040  042524  052123
6462  067510  020040  020040  042522
6463  067516  020107  020040  020040
6464  067524  047507  042117  020040
6465  067532  020040  042522  042503
6466  067540  053111  042105  005015
6467  067546  020040  020040  020040          .ASCIZ  /       NO      ADDRESS DATA    DATA/
6468  067554  020040  047516  020040
6469  067562  020040  020040  042101
6470  067570  051104  051505  020123
6471  067576  040504  040524  020040
6472  067604  020040  040504  040524
6473  067612     000
6474  067613     120  020103  020040  DH6:    .ASCII  /PC     TEST    REG     RECEIVED/<15><12>
6475  067620  020040  052040  051505
6476  067626  020124  020040  051040
6477  067634  043505  020040  020040
6478  067642  051040  041505  044505
6479  067650  042526  004504     012
6480  067655     040  020040  020040          .ASCIZ  /       NO      ADDRESS DATA/
6481  067662  020040  047040  020117
6482  067670  020040  020040  040440
6483  067676  042104  042522  051523
6484  067704  042040  052101  000101
6485  067712  041520  020040  020040  DH7:    .ASCIZ  /PC     TEST    REG     ADDRESS/
6486  067720  020040  042524  052123
6487  067726  020040  020040  042522
6488  067734  020107  020040  020040
6489  067742  042101  051104  051505
6490  067750  000123
6491
6492  067752  041520  020040  020040  DH10:   .ASCII  /PC     TEST    FAILING CONTENT CONTENT CONTENT CONTENT/<15><12>
6493  067760  020040  042524  052123
6494  067766  020040  020040  040506
6495  067774  046111  047111  020107
6496  070002  047503  052116  047105
6497  070010  020124  047503  052116
6498  070016  047105  020124  047503
6499  070024  052116  047105  020124
```

```
 6500  070032  047503  052116  047105
 6501  070040  006524     012
 6502  070043     040  020040  020040          .ASCIZ  /       NO      REG.    RHCS1   RHCS1   RHDS1   RHER1/
 6503  070050  020040  047040  020117
 6504  070056  020040  020040  051040
 6505  070064  043505  020056  020040
 6506  070072  051040  041510  030523
 6507  070100  020040  051040  041510
 6508  070106  030523  020040  051040
 6509  070114  042110  030523  020040
 6510  070122  051040  042510  030522
 6511  070130     000
 6512  070131     120  020103  020040  DH26:   .ASCII  /PC      TEST    CONT OF CONT OF CONT OF CONT OF CONT OF CONT OF/<15><12
 6513  070136  020040  052040  051505
 6514  070144  020124  020040  041440
 6515  070152  047117  020124  043117
 6516  070160  041440  047117  020124
 6517  070166  043117  041440  047117
 6518  070174  020124  043117  041440
 6519  070202  047117  020124  043117
 6520  070210  041440  047117  020124
 6521  070216  043117  041440  047117
 6522  070224  020124  043117  005015
 6523  070232  020040  020040  020040          .ASCIZ  /       NO      RHCS1   RHCS2   RHDS1   RHER1   RHER2   RHER3/
 6524  070240  020040  047516  020040
 6525  070246  020040  020040  044122
 6526  070254  051503  020061  020040
 6527  070262  044122  051503  020062
 6528  070270  020040  044122  051504
 6529  070276  020061  020040  044122
 6530  070304  051105  020061  020040
 6531  070312  044122  051105  020062
 6532  070320  020040  044122  051105
 6533  070326  000063
 6534  070330  041520  020040  020040  DH30:   .ASCII  /PC      TEST    WORD    GOOD    BAD/<15><12>
 6535  070336  020040  042524  052123
 6536  070344  020040  020040  047527
 6537  070352  042122  020040  020040
 6538  070360  047507  042117  020040
 6539  070366  020040  040502  006504
 6540  070374     012
 6541  070375     040  020040  020040          .ASCIZ  /       NO      NO      DATA    DATA/<15><12>
 6542  070402  020040  047040  020117
 6543  070410  020040  020040  047040
 6544  070416  020117  020040  020040
 6545  070424  020040  052101  020101
 6546  070432  020040  042040  052101
 6547  070440  006501  000012
 6548
 6549  070444  041520  020040  020040  DH51:   .ASCII  /PC      TEST    REG     GOOD    RECVD   ILLEGL/<15><12>
 6550  070452  020040  042524  052123
 6551  070460  020040  020040  042522
 6552  070466  020107  020040  020040
 6553  070474  047507  042117  020040
 6554  070502  020040  042522  053103
 6555  070510  020104  020040  046111
```

```
6556  070516  042514  046107  005015
6557  070524  020040  020040  020040          .ASCIZ  /       NO      ADDRESS DATA    DATA    FUNCTN/
6558  070532  020040  047516  020040
6559  070540  020040  020040  042101
6560  070546  051104  051505  020123
6561  070554  040504  040524  020040
6562  070562  020040  040504  040524
6563  070570  020040  020040  052506
6564  070576  041516  047124     000
6565
6566  070603     120  020103  020040  DH60:   .ASCII  /PC     TEST    REG     GOOD    RECVD   MODFING/<15><12>
6567  070610  020040  052040  051505
6568  070616  020124  020040  051040
6569  070624  043505  020040  020040
6570  070632  043440  047517  020104
6571  070640  020040  051040  041505
6572  070646  042126  020040  046440
6573  070654  042117  044506  043516
6574  070662  005015
6575  070664  020040  020040  020040          .ASCIZ  /       NO      ADDRESS DATA    DATA    REG/
6576  070672  020040  047516  020040
6577  070700  020040  020040  042101
6578  070706  051104  051505  020123
6579  070714  040504  040524  020040
6580  070722  020040  040504  040524
6581  070730  020040  020040  042522
6582  070736  000107
6583  070740  041520  020040  020040  DH61:   .ASCII  /PC     TEST    PC OF   RHCS1/<15><12>
6584  070746  020040  042524  052123
6585  070754  020040  020040  041520
6586  070762  047440  020106  020040
6587  070770  044122  051503  006461
6588  070776     012
6589  070777     040  020040  020040          .ASCIZ  /       NO      JSR     WAS/
6590  071004  020040  047040  020117
6591  071012  020040  020040  045040
6592  071020  051123  020040  020040
6593  071026  053440  051501     000
6594  071033     120  020103  020040  DH62:   .ASCII  /PC     TEST    PC OF   RHDS1/<15><12>
6595  071040  020040  052040  051505
6596  071046  020124  020040  050040
6597  071054  020103  043117  020040
6598  071062  051040  042110  030523
6599  071070  005015
6600  071072  020040  020040  020040          .ASCIZ  /       NO      JSR     WAS/
6601  071100  020040  047516  020040
6602  071106  020040  020040  051512
6603  071114  020122  020040  020040
6604  071122  040527  000123
6605  071126  041520  020040  020040  DH65:   .ASCII  /PC     TEST    CONT    CONT    CONT    /<15><12>
6606  071134  020040  042524  052123
6607  071142  020040  020040  047503
6608  071150  052116  020040  020040
6609  071156  047503  052116  020040
6610  071164  020040  047503  052116
6611  071172  020040  020040  005015
```

```
 6612  071200  020040  020040  020040          .ASCIZ  /       NO      RHCS1   RHAS    RHDS1   /
 6613  071206  020040  047516  020040
 6614  071214  020040  020040  044122
 6615  071222  051503  020061  020040
 6616  071230  044122  051501  020040
 6617  071236  020040  044122  051504
 6618  071244  020061  020040     000
 6619  071251     120  020103  020040  DH66:   .ASCIZ  /PC      TEST    RHDST   RHER1   RHER2   RHER3   RHCS1   RHCS2/
 6620  071256  020040  052040  051505
 6621  071264  020124  020040  051040
 6622  071272  042110  052123  020040
 6623  071300  051040  042510  030522
 6624  071306  020040  051040  042510
 6625  071314  031122  020040  051040
 6626  071322  042510  031522  020040
 6627  071330  051040  041510  030523
 6628  071336  020040  051040  041510
 6629  071344  031123     000
 6630
 6631  071347     120  020103  020040  DH77:   .ASCII  /PC      TEST    PC OF   REG.    GOOD    BAD/<15><12>
 6632  071354  020040  052040  051505
 6633  071362  020124  020040  050040
 6634  071370  020103  043117  020040
 6635  071376  051040  043505  020056
 6636  071404  020040  043440  047517
 6637  071412  020104  020040  041040
 6638  071420  042101  005015
 6639  071424  020040  020040  020040          .ASCIZ  /       NO      JSR     ADDR    DATA    DATA/
 6640  071432  020040  047516  020040
 6641  071440  020040  020040  051512
 6642  071446  020122  020040  020040
 6643  071454  042101  051104  020040
 6644  071462  020040  040504  040524
 6645  071470  020040  020040  040504
 6646  071476  040524     000
 6647
 6648          071502                  .EVEN
 6649
 6650  071502  001116  004604  041702  DT1:    .WORD   $ERRPC,TSTNM,WAITPC,WAITBT,WAITRE,$BDDAT,CS1,0
 6651  071510  041706  041704  001126
 6652  071516  002362  000000
 6653  071522  001116  004604  041702  DT4:    .WORD   $ERRPC,TSTNM,WAITPC,WAITBT,WAITRE,$BDDAT,WAITTM,0
 6654  071530  041706  041704  001126
 6655  071536  041710  000000
 6656  071542  001116  004604  004600  DT5:    .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,0
 6657  071550  001124  001126  000000
 6658  071556  001116  004604  004600  DT6:    .WORD   $ERRPC,TSTNM,REGADR,$BDDAT,0
 6659  071564  001126  000000
 6660  071570  001116  004604  001200  DT7:    .WORD   $ERRPC,TSTNM,$TMP1,0
 6661  071576  000000
 6662
 6663  071600  001116  004604  001122  DT10:   .WORD   $ERRPC,TSTNM,$BDADR,CS1,CS2,DS1,ER1,0
 6664  071606  002362  002360  002404
 6665  071614  002364  000000
 6666  071620  001116  004604  002362  DT26:   .WORD   $ERRPC,TSTNM,CS1,CS2,DS1,ER1,ER2,ER3,0
 6667  071626  002360  002404  002364
```

```
 6668  071634  002370  002376  000000
 6669  071642  001116  004604  004602  DT30:   .WORD   $ERRPC,TSTNM,ERWORD,$GDDAT,$BDDAT,0
 6670  071650  001124  001126  000000
 6671
 6672  071656  001116  004604  004600  DT51:   .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,ILLEGL,0
 6673  071664  001124  001126  002464
 6674  071672  000000
 6675
 6676  071674  001116  004604  004600  DT60:   .WORD   $ERRPC,TSTNM,REGADR,$GDDAT,$BDDAT,$BDADR,0
 6677  071702  001124  001126  001122
 6678  071710  000000
 6679  071712  001116  004604  041506  DT61:   .WORD   $ERRPC,TSTNM,PCJSR,$BDADR,0
 6680  071720  001122  000000
 6681  071724  001116  004604  041506  DT62:   .WORD   $ERRPC,TSTNM,PCJSR,$BDADR,0
 6682  071732  001122  000000
 6683  071736  001116  004604  002362  DT65:   .WORD   $ERRPC,TSTNM,CS1,AS,DS1,0
 6684  071744  002400  002404  000000
 6685  071752  001116  004604  002364  DT66:   .WORD   $ERRPC,TSTNM,ER1,ER2,ER3,CS1,CS2,0
 6686  071760  002370  002376  002362
 6687  071766  002360  000000
 6688
 6689  071772  001116  004604  041506  DT77:   .WORD   $ERRPC,TSTNM,PCJSR,REGADR,$GDDAT,$BDDAT,0
 6690  072000  004600  001124  001126
 6691  072006  000000
 6692
 6693  072010     000     000     000  DF1:    .BYTE   0,0,0,0,0,0,0
 6694  072013     000     000     000
 6695  072016     000
 6696  072017     000     000     000  DF4:    .BYTE   0,0,0,0,0,1,0
 6697  072022     000     000     001
 6698  072025     000
 6699  072026     000     000     000  DF5:    .BYTE   0,0,0,0,0
 6700  072031     000     000
 6701  072033     000     000     000  DF6:    .BYTE   0,0,0,0
 6702  072036     000
 6703  072037     000     000     000  DF7:    .BYTE   0,0,0
 6704
 6705  072042     000     000     000  DF10:   .BYTE   0,0,0,0,0,0,0
 6706  072045     000     000     000
 6707  072050     000
 6708
 6709  072051     000     000     000  DF26:   .BYTE   0,0,0,0,0,0,0,0
 6710  072054     000     000     000
 6711  072057     000     000
 6712
 6713  072061     000     000     000  DF30:   .BYTE   0,0,0,0,0
 6714  072064     000     000
 6715
 6716  072066     000     000     000  DF51:   .BYTE   0,0,0,0,0,0
 6717  072071     000     000     000
 6718
 6719  072074     000     000     000  DF60:   .BYTE   0,0,0,0,0,0
 6720  072077     000     000     000
 6721  072102     000     000     000  DF61:   .BYTE   0,0,0,0
 6722  072105     000
 6723  072106     000     000     000  DF62:   .BYTE   0,0,0,0
```

```
 6724  072111      000
 6725  072112      000       000       000  DF65:    .BYTE   0,0,0,0,0
 6726  072115      000       000
 6727  072117      000       000       000  DF66:    .BYTE   0,0,0,0,0,0,0,0
 6728  072122      000       000       000
 6729  072125      000       000
 6730
 6731  072127      000       000       000  DF77:    .BYTE   0,0,0,0,0,0
 6732  072132      000       000       000
 6733
 6734            072136                     .EVEN
 6735
 6736
 6737            000001                     .END
```

```
ACE    = 020000          1627#   2421    2441
ACL    = 000040          1625#
ACU    = 100000          1585#
ADDMOD   000204           631#   1938
ADTIMO   044572          4999    5019#
AOE    = 001000          1507#
AS       002400          1721#   4831#   5061    6683
ATA    = 100000          1495#   2421    2441    2583    2817    2919    2970    2995    3843    3896    4255    4561
ATABLE   004666          1790#   2117
ATTENT   004740          1814#   2064#   2117#   2422    2442    2583    2625    2672    2718    2781    2817    2927    2970
                         2979    2995    3843    3896    4255

ATO    = 000001          1527#
AT1    = 000002          1528#
AT2    = 000004          1529#
AT3    = 000010          1530#
AT4    = 000020          1531#
AT5    = 000040          1532#
AT6    = 000100          1533#
AT7    = 000200          1534#
A16    = 000400          1470#
A17    = 001000          1471#
BA       002356          1710#   5061
BAI    = 000010          1443#   2616    2619
BASECH   043606           631    4992#   5021
BEGIN    005012           630    1845#   5018    5118
BEGIN1   004762           634    1839#
BEGIN2   004776           632    1842#
BIRD     020156          2802    2809#
BIT0   = 000001           624#
BIT00  = 000001           624#
BIT01  = 000002           624#
BIT02  = 000004           624#
BIT03  = 000010           624#
BIT04  = 000020           624#
BIT05  = 000040           624#
BIT06  = 000100           624#
BIT07  = 000200           624#
BIT08  = 000400           624#    5033
BIT09  = 001000           624#    5033    5034
BIT1   = 000002           624#
BIT10  = 002000           624#    5034
BIT11  = 004000           624#    5033
BIT12  = 010000           624#
BIT13  = 020000           624#    5034
BIT14  = 040000           624#    5033
BIT15  = 100000           624#    2995
BIT2   = 000004           624#
BIT3   = 000010           624#
BIT4   = 000020           624#    4642
BIT5   = 000040           624#
BIT6   = 000100           624#
BIT7   = 000200           624#
BIT8   = 000400           624#
BIT9   = 001000           624#
BPTVEC= 000014            624#
CA       002374          1719#   5061
```

```
CAT     020122  2798  2805#
CC      002416  1728#  5061
CHECK   041510  2274  2313  2666  2712  2839  2945  2951  4541#
CHECKC  041546  4545  4551#
CHECKT  041532  2363  2482  2542  2567  2599  2777  2810  2829  2845  2956  2986  3016  3043
                3083  3123  3163  3205  3248  3293  3330  3372  3420  3479  3532  3575  3629
                3670  3739  3778  3866  3920  3937  3980  4005  4046  4068  4105  4127  4172
                4206  4547#
CHREG   042332  2281  2319  2421  2441  2583  2616  2619  2635  2688  2734  2817  2923  2926
                2970  2995  3843  3896  4220  4255  4307  4768#
CKSWR = 104407  5033  5034  5117#
CLAREA  041322  3030  3036  3076  3116  3151  3156  3195  3198  3236  3241  3279  3282  3318
                3323  3362  3365  3406  3411  3467  3470  3516  3520  3526  3565  3568  3612
                3616  3622  3662  3664  3726  3732  3765  3770  3861  3863  3914  3917  3948
                3971  3974  4078  4100  4142  4163  4203  4239  4276  4278  4449#
CLDISK  041452  2064  2162  2169  2193  2217  2238  2274  2313  2348  2351  2363  2438  2482
                2503  2542  2565  2599  2652  2666  2712  2764  2768  2773  2802  2806  2810
                2829  2839  2951  2956  3005  3016  3020  3024  3076  3113  3148  3195  3233
                3279  3315  3362  3401  3467  3512  3565  3608  3662  3700  3720  3765  3834
                3881  3914  3937  3942  4001  4041  4068  4075  4127  4139  4200  4239  4274
                4401  4519#
CLR   = 000040  1445#  4524
COMPAR  043470  3068  3105  3142  3186  3225  3271  3309  3353  3392  3458  3499  3557  3600
                3654  3691  3798  3871  3925  4029  4285  4962#
COMREG  042440  2285  2324  2428  2448  2552  2588  2640  2693  2739  2785  2820  2931  2973
                2997  3055  3093  3133  3175  3215  3260  3299  3342  3382  3446  3488  3544
                3587  3641  3678  3754  3788  3845  3898  3989  4016  4056  4115  4182  4223
                4259  4311  4823#
CPHALT  066714  2274  2313  2363  2482  2542  2567  2599  2666  2712  2773  2810  2829  2839
                2951  3016  3043  3083  3123  3163  3205  3248  3289  3330  3372  3420  3479
                3532  3575  3627  3670  3739  3778  3866  3920  3937  3980  4005  4046  4068
                4105  4127  4172  4206  6390#
CR    = 000015   624#  5033
CRLF  = 000200   624#  5033
CSF   = 000002  1572#
CSU   = 000010  1574#
CS1     002362  1714#  2410  5061  6650  6663  6666  6683  6685
CS2     002360  1711#  2391  2509  5061  6663  6666  6685
DB      002352  1708#
DCK   = 100000  1513#
DCL   = 000100  1626#
DCLEAR  002430  1740#  2606
DDISP = 177570   624#  650  1851
DE1   = 000040  1485#
DFF20 = 000002  1481#
DFF5  = 000001  1480#
DF1     072010   665  677  688  6693#
DF10    072042   742  756  770  786  6705#
DF26    072051   905  1382  1394  1406  1418  6709#
DF30    072061   926  945  961  978  1039  1073  1097  1133  1169  1278  1289  1320  1339
                6713#
DF4     072017   701  6696#
DF5     072026   711  796  808  822  836  848  856  863  873  882  893  916  937
                 955  971  989  1000  1014  1026  1051  1064  1086  1126  1145  1157  1182
                1225  1237  1303  1313  1331  1351  1370  6699#
DF51    072066  1112  6716#
```

```
DF6      072033            721    6701#
DF60     072074           1199    6719#
DF61     072102           1208    6721#
DF62     072106           1217    6723#
DF65     072112           1249    6725#
DF66     072117           1267    6727#
DF7      072037            729    6703#
DF77     072127           1361    6731#
DH1      067154            658     670     682    6423#
DH10     067752            734     748     762     778    6492#
DH26     070131            897    1375    1387    1399    1411    6512#
DH30     070330            921     943     959     976    1037    1071    1095    1131    1167    1276    1287    1318    1337
                          6534#
DH4      067333            695    6443#
DH5      067474            706     791     803     817     831     843     854     861     871     880     891     914     935
                           953     969     987     998    1012    1024    1049    1062    1084    1124    1143    1155    1180
                          1223    1235    1298    1311    1329    1349    1365    6460#
DH51     070444           1106    6549#
DH6      067613            717    6474#
DH60     070603           1193    6566#
DH61     070740           1203    6583#
DH62     071033           1212    6594#
DH65     071126           1243    6605#
DH66     071251           1259    6619#
DH7      067712            726    6485#
DH77     071347           1355    6631#
DIGB   = 000004           1482#
DISPLA   001142            650#   1851#   5033#   5034#
DISPRE   000174            625#   1851
DLT    = 100000           1455#
DL64   = 000020           1484#
DMD    = 000001           1517#   2348    2351    2367    2441    2764    2768    2802    2806    4381
DOG      017346           2759    2767#
DPR    = 000400           1488#   2243    2319    2408    2778    2815    2886    2913    2967    2992    3048    3086    3126
                          3168    3208    3253    3292    3335    3375    3424    3481    3537    3580    3634    3673    3746
                          3781    3840    3893    4049    4108    4176    4562
DRY    = 000200           1487#   2245    2319    2412    2778    2815    2919    2967    2992    3016    3020    3719    3840
                          3885    3893    3942    4074    4138    4562
DST      002366           1716#   5061
DSWR   = 177570            624#    650    1851
DS1      002404           1723#   2408    2412    4384    4396    5061    6663    6666    6683
DT       002406           1724#   2157#   2630    5061
DTE    = 010000           1510#
DTSY   = 001000           1523#
DT1      071502            664     676     687    6650#
DT10     071600            741     755     769     785    6663#
DT26     071620            904    1381    1393    1405    1417    6666#
DT30     071642            925     944     960     977    1038    1072    1096    1132    1168    1277    1288    1319    1338
                          6669#
DT4      071522            700    6653#
DT5      071542            710     795     807     821     835     847     855     862     872     881     892     915     936
                           954     970     988     999    1013    1025    1050    1063    1085    1125    1144    1156    1181
                          1224    1236    1302    1312    1330    1350    1369    6656#
DT51     071656           1111    6672#
DT6      071556            720    6658#
DT60     071674           1198    6676#
```

DT61	071712	1207	6679#										
DT62	071724	1216	6681#										
DT65	071736	1248	6683#										
DT66	071752	1266	6685#										
DT7	071570	728	6660#										
DT77	071772	1360	6689#										
DVA =	004000	1473#	2185	2256	2319	2410	2778	2815	2867	2967	2992	3048	3086	3126
		3168	3208	3253	3292	3335	3375	3424	3481	3537	3580	3634	3673	3746
		3781	3840	3893	4049	4108	4176	4553						
ECH =	000100	1504#												
ECI =	004000	1598#	2688	2734	2995	3039	3079	3119	3159	3201	3244	3285	3326	3368
		3414	3473	3529	3571	3625	3666	3735	3773	3865	3919	3976	4001	4041
		4102	4165	4205	4241	4280								
EC1	002412	1726#	5061											
EC2	002414	1727#	5061											
EMTVEC=	000030	624#	1851*	1920*	1950*									
EM1	051114	656	5127#											
EM10	051634	732	5187#											
EM100	066322	1364	6344#											
EM101	066404	1373	6353#											
EM102	066466	1385	6362#											
EM103	066545	1397	6370#											
EM104	066631	1409	6379#											
EM11	051703	745	5194#											
EM12	052010	759	5206#											
EM13	052116	773	5218#											
EM14	052240	789	5232#											
EM15	052313	799	5240#											
EM16	052454	811	5258#											
EM17	052602	825	5274#											
EM2	051143	668	5131#											
EM20	052737	839	5292#											
EM21	053103	852	5310#											
EM22	053156	859	5318#											
EM23	053231	866	5326#											
EM24	053374	876	5344#											
EM25	053537	885	5362#											
EM26	053721	896	5382#											
EM27	054011	908	5392#											
EM3	051232	680	5141#											
EM30	054227	919	5419#											
EM31	054307	929	5428#											
EM32	054524	940	5454#											
EM33	054633	948	5466#											
EM34	055035	958	5490#											
EM35	055061	964	5494#											
EM36	055277	974	5519#											
EM37	055354	981	5527#											
EM4	051313	691	5150#											
EM40	055571	992	5553#											
EM41	056000	1003	5577#											
EM42	056261	1017	5608#											
EM43	056537	1029	5639#											
EM44	057101	1042	5679#											
EM45	057373	1054	5712#											
EM46	057667	1067	5745#											

```
EM47    060002          1076    5758#
EM5     051424           704    5163#
EM50    060264          1089    5790#
EM51    060503          1100    5816#
EM52    060750          1116    5845#
EM53    061225          1129    5875#
EM54    061313          1136    5885#
EM55    061560          1148    5914#
EM56    062046          1160    5946#
EM57    062256          1172    5970#
EM6     051476           714    5170#
EM60    062550          1185    6004#
EM61    063161          1202    6051#
EM62    063242          1211    6060#
EM63    063323          1221    6069#
EM64    063377          1229    6077#
EM65    063616          1242    6102#
EM66    063640          1253    6105#
EM67    064032          1269    6126#
EM7     051564           724    5179#
EM70    064322          1280    6159#
EM71    064530          1292    6184#
EM72    064771         6213#
EM73    065221          1316    6241#
EM74    065273          1305    1323    6249#
EM75    065537          1334    6278#
EM76    065642          1342    6290#
EM77    066077          1354    6318#
ERFLG$  004734          1810#   5034#
ERR     = 040000        1494#   2421    2441    3016    4255    4384    4396
ERRVEC= 000004           624#   1851#   1922#   1930#   1952#   5033#
ERWORD  004602          1769#   4970#   4976#   6669
ER1     002364          1715#   5061    6663    6666    6685
ER2     002370          1717#   5061    6666    6685
ER3     002376          1720#   5061    6666    6685
EXT1    = 000001        1552#
EXT10   = 000010        1555#
EXT2    = 000002        1553#
EXT20   = 000020        1556#
EXT4    = 000004        1554#
EXT40   = 000040        1557#
FEN     = 000200        1578#
FER     = 000020        1502#
FILL    042376          4794#
FILLRE  041354          2610    2623    2627    2629    2635    2688    2734    2781    2817    2995    3051    3089    3129
                        3171    3211    3256    3295    3338    3378    3442    3484    3540    3583    3637    3675    3749
                        3750    3784    3843    3896    3985    4011    4052    4111    4179    4255    4471#
FINACC  004664          1784#   4617#
FINALA  004662          1783#   4618#
FIRST   004736          1811#   1862    1871#
FISH    017402          2764    2772#
FLHEAD  041276          3027    3036    3514    3518    3526    3568    3610    3614    3622    3664    3723    3729    3769
                        3863    3917    3944    3971    4424#
FMT22 = 010000          1599#   2688    2734    2995    3039    3079    3119    3159    3201    3244    3285    3326    3368
                        3414    3473    3529    3571    3625    3666    3735    3773    3865    3919    3976    4001    4041
                        4102    4165    4205    4241    4280
```

```
FUTABL  002422          1736#
GNS     = ******        6#      1868    1869    1870    1909    1937    1947    1950    1974    1997    2038    2040    2042
                        2086    2118    2120    2146    2148    2150    2164    2166    2219    2221    2238    2240    2383
                        2385    2401    2403    2484    2494    2496    2514    2516    2531    2652    2659    2837    2903
                        2940    2942    3433    3437    4215    42.7    4302    4304    4327    4329    4879    4881    4884
                        4887    4933    4995    5008    5010    5012    5014    5016    5018    5020    5026    5061    5117
GO      = 000000:       1467#   2279    2318    2463    2530    2548    2572    2606    2680    2683    2726    2729    2778
                        2815    2842    2855    2867    2923    2950    2966    2967    2991    2992    3016    3020    3045
                        3048    3085    3086    3125    3126    3165    3168    3207    3208    3250    3253    3291    3292
                        3332    3335    3374    3375    3421    3424    3480    3481    3534    3537    3577    3580    3631
                        3634    3672    3673    3719    3741    3746    3780    3781    3838    3840    3868    3884    3891
                        3893    3922    3941    3982    4007    4048    4049    4073    4107    4108    4137    4175    4176
                        4208    4245    4282    4389    4642
GPV     = 000010        1483#
GTSWP   = 104406        1873    5117#
HCE     = 000200        1505#
HCI     = 002000        1597#   2686    2734    2995    3039    3079    3119    3159    3201    3244    3285    3326    3368
                        3414    3473    3529    3571    3625    3666    3735    3773    3865    3919    3978    4001    4041
                        4102    4165    4205    4241    4280
HCRC    = 000400        1506#
HT      = 000011        624#    5033
HTT.    = 003323        2280#   2319#   2463#   2530#   2549#   2573#   2607#   2685#   2731#   2778#   2815#   2843#   2950#
                        2967#   2992#   3016#   3020#   3040#   3086#   3126#   3168#   3208#   3253#   3292#   3335#   3375#
                        3424#   3481#   3537#   3580#   3634#   3673#   3719#   3746#   3781#   3840#   3869#   3885#   3893#
                        3923#   3942#   3983#   4008#   4049#   4074#   4108#   4138#   4176#   4210#   4246#   4283#
IAE     = 002000        1508#
IE      = 000100        1468#   2175    2199    2683    2729    2778    2815    2871    2872    2966    2967    2991    2992
                        3016    3020    3045    3048    3085    3086    3125    3126    3165    3168    3207    3208    3250
                        3253    3291    3292    3332    3335    3374    3375    3421    3424    3480    3481    3534    3537
                        3577    3580    3631    3634    3672    3673    3719    3741    3746    3780    3781    3838    3840
                        3868    3884    3891    3893    3922    3941    3982    4007    4048    4049    4073    4107    4108
                        4137    4175    4176    4208    4245    4282    4664    4676    4738    4751
ILF     = 000001        1498#
ILLEGL  002464          1754#   6672
ILR     = 000002        1499#
INUMIT  004752          1821#   2487#   2520
IOTVEC= 000020          624#    1851#
IR      = 000100        1446#   2228    2616    2619    4252    4253
IXE     = 004000        1582#
KIPAR0= 172340          649#
KIPAR1= 172342          649#
KIPAR2= 172344          649#
KIPAR3= 172346          649#
KIPAR4= 172350          649#
KIPAR5= 172352          649#
KIPAR6= 172354          649#
KIPAR7= 172356          649#
KIPDR0= 172300          649#
KIPDR1= 172302          649#
KIPDR2= 172304          649#
KIPDR3= 172306          649#
KIPDR4= 172310          649#
KIPDR5= 172312          649#
KIPDR6= 172314          649#
KIPDR7= 172316          649#
LA      002420          1729#   5061
```

```
LBT     = 002000          1490#  4561
LF      = 000012           624#  5033
LT.     = 000105          2280#  2319#  2463#  2530#  2549#  2573#  2607#  2685#  2731#  2778#  2815#  2843#  2950#
                          2967#  2992#  3016#  3020#  3048#  3086#  3126#  3168#  3208#  3253#  3292#  3335#  3375#
                          3424#  3481#  3537#  3580#  3634#  3673#  3719#  3746#  3781#  3840#  3869#  3885#  3893#
                          3923#  3942#  3983#  4008#  4049#  4074#  4108#  4138#  4176#  4210#  4246#  4283#
MAKECY  041122            2348   2351   2764   2768   2802   2806   4374#
MCLK    = 000002          1518#
MCPE    = 020000          1474#  2254   2421   2441
MHS     = 001000          1580#
MINX    = 000004          1519#
MMVEC   = 000250           649#
MOL     = 010000          1492#  2163   2164   2222   2241   2319   2656   2664   2778   2815   2835   2888   2889
                          2915   2916   2967   2992   3048   3086   3126   3168   3208   3253   3292   3335   3375
                          3424   3481   3537   3580   3634   3673   3746   3781   3840   3893   4049   4108   4176
                          4562
MPE     = 000400          1448#
MR      = 002402          1722#  5061
MRD     = 000020          1521#
MSE     = 000020          1575#
MSTCK   = 000010          1520#
MWR     = 000040          1522#
MXF     = 001000          1449#  2616
NED     = 010000          1452#  2391   2421   2441   2509
NEM     = 004000          1451#  3430
NHS     = 002000          1581#
NOPERA  002422            1737#  2543   2548   2567   2572
NOPUSH  004724            1804#  1839*  1843*  1846*  2219   2363   2482   2652   2829   4200
NOUNIT  004720            1800#  2012*  2048*  2055   2090*  4345*
NUNIT   004722            1802#  2055*  2056*  2578   2619   2778   2868   2924   2967   2992
OCYL    = 100000          1629#
OF      002372            1718#  5061
OFREV   = 000200          1596#
OFSET   041436            3016   4514#
OFSETC  002454            1750#  2961   2966   2967   3016   4515
OFSTVL  004610            1776#  2951*  2952*  2959   3005*  3006   3008*  3009
OF100   = 000004          1591#
OF200   = 000010          1592#
OF25    = 000001          1589#
OF400   = 000020          1593#
OF50    = 000002          1590#
OF800   = 000040          1594#
OPERSE  042576            4876#  5033
OPI     = 020000          1511#
OR      = 000200          1447#  4252   4253
PAR     = 000010          1501#  2635
PCJSR   041506            4375#  4376*  4539#  4541*  4542*  4547*  4548*  6679   6681   6689
PCLBUF  002346            1690#  4641#
PCLCSR  002344            1689#  4615*  4642*
PCLCTR  002350            1691#  4616
PGE     = 002000          1450#
PIP     = 020000          1493#  2778   2815   2886   2913   2967   2992   3840   3893
PIRQ    = 177772           624#
PIRQVE  = 000240           624#
PKACK   002460            1752#  2274   2279   2313   2318   2319   2463   2530   2842   2950
PLU     = 020000          1584#
```

```
PRE   = 000020  1624#
PRITEM  047270  1903#  2301#  2500#  2709#  2943#  4322#  4344#  4878#  5051#  5061#  5078  5082#
PROG  = 001000  1489#
PR0   = 000000   624#
PR1   = 000040   624#
PR2   = 000100   624#
PR3   = 000140   624#
PR4   = 000200   624#
PR5   = 000240   624#
PR6   = 000300   624#
PR7   = 000340   624#
PS    = 177776   624#  1854#  2174#  2198#  2217#  4325#  4677#
PSEL  = 002000  1472#  3414   3421   3424   3473   3480   3481   4167   4175   4176
PSU   = 000001  1621#
PSW   = 177776   624#
PUTREG  043364  2390   2407   2508   4383   4395   4830   4905#
PWRVEC= 000024   624#  1851#  5118#
RA      000200   630#  5016
RDCHR = 104410  5033   5117#
RDLIN = 104411  5034   5117#
RDOCT = 104412  1909   4884   4887   4996   5010   5117#
RDY   = 000200  1469#  2175   2185   2199   2256   2319   2421   2549   2573   2607   2731   2778   2815
                2858   2867   2967   2992   3048   3086   3126   3168   3208   3253   3292   3335   3375
                3424   3481   3537   3580   3634   3673   3746   3781   3840   3869   3893   3923   3983
                4008   4049   4108   4176   4210   4246   4283   4553
READAT  002446  1747#  3119   3125   3126   3201   3207   3208   3285   3291   3292   3368   3374   3375
                3473   3480   3481   4280   4282
READIN  002462  1753#  2673   2680   2693   2719   2726   2729
RECALI  002426  1739#  2775   2778   2812   2815   3020   3719   3884
REFOR   002450  1748#  3079   3085   3086   3571   3577   3580   3666   3672   3673   3773   3780   3781
                3865   3868   3919   3922   4001   4007
REGADR  004600  1768#  1963*  2267*  2319*  2778*  2815*  2881*  2897*  2967*  2992*  3048*  3086*  3126*
                3168#  3208#  3253#  3292#  3335#  3375#  3424#  3481#  3537#  3580#  3634#  3673#  3746#
                3781#  3840#  3893#  4049#  4108#  4176#  4406#  4839#  6656   6658   6672   6676   6689
REGSAV  047124  5034#
REGSA1  047132  1920   5034#
REINTO  003534  1762#  3036   3068   3076   3079   3089   3105   3119   3129   3142   3156   3186   3195
                3201   3211   3225   3241   3271   3279   3285   3295   3309   3323   3353   3362   3368
                3378   3392   3411   3458   3467   3473   3484   3499   3526   3557   3565   3571   3583
                3600   3622   3654   3662   3666   3675   3691   3765   3773   3784   3798   3861   3865
                3871   3914   3919   3925   3974   4001   4011   4029   4276   4280   4285
RELEAS  002432  1741#
RESVEC= 000010   624#
RETCL   002456  1751#  2986   2991   2992
RHAS    002316  1671#  1958   1974   2672#  2718#  2979#
RHBA    002274  1659#  2610   3051   3089   3129   3171   3211   3256   3295   3338   3378   3442   3484
                3540   3563   3637   3675   3749   3784   3985   4011   4052   4111   4179   4250   4932#
RHBAE   002340  1683#  1945
RHCA    002312  1669#  2623   2688   2734   3637   3750   4378#  4487#  4505#  4929#
RHCC    002334  1678#  2265   2267   2269   2604   2781   2817   3637   3750   3843   3896   4402   4406
                4617   4908
RHCS1   002300  1664#  2274#  2279#  2313#  2318#  2421   2441   2463#  2530#  2543#  2548#  2549   2567#
                2572#  2573   2606#  2607   2616   2635   2673#  2680#  2683#  2719#  2726#  2729#  2731
                2775#  2778#  2781   2812#  2815#  2817   2842#  2850#  2855#  2857   2923   2926   2950#
                2961#  2966#  2970   2986#  2991#  2995   3016#  3020#  3045#  3048   3085#  3086   3125#
                3126   3165#  3168   3207#  3208   3250#  3253   3291#  3292   3332#  3335   3374#  3375
```

		3414*	3421*	3424	3473*	3480*	3481	3534*	3537	3577*	3580	3631*	3634	3672*
		3673	3719*	3741*	3746	3780*	3781	3838*	3843	3868*	3869	3884*	3891*	3896
		3922*	3923	3941*	3982*	3983	4007*	4008	4048*	4049	4073*	4107*	4108	4137*
		4167*	4175*	4176	4208*	4210	4245*	4246	4255	4282*	4283	4380*	4389*	4489*
		4506*	4515*	4519	4664	4676	4738	4751	4938*	4993	5000	5003	5012	
RHCS2	002276	1660*	1975	2018*	2097*	2421	2441	2616	2619	3430	4251	4520	4935*	
RHCS3	002342	1684*												
RHDB	002270	1657*	1925	4997										
RHDST	002304	1666*	2389	2507	2623	2688	2734	3051	3089	3129	3171	3211	3256	3295
		3338	3378	3442	3484	3540	3583	3637	3675	3750	3784	3985	4011	4052
		4111	4179	4255	4379*	4488*	4930*							
RHDS1	002322	1673*	2280	2281	2319	2421	2441	2463	2530	2583	2629	2685	2688	2734
		2778	2781	2815	2817	2833	2843	2914	2950	2967	2970	2992	2995	3016
		3020	3719	3840	3843	3885	3893	3896	3942	4074	4138	4220	4255	4307
		4521												
RHDT	002324	1674*	2019	2021	2024	2026	2029	2031	2040	2098	2100	2107	2109	2120
		2124	2126	2129	2131	2134	2136	2155	2157					
RHEC1	002330	1676*	2635											
RHEC2	002332	1677*	2635											
RHER1	002302	1665*	1978	2623	4255	4522								
RHER2	002306	1667*	2623											
RHER3	002314	1670*	2421	2441	2623									
RHLA	002336	1679*	4618											
RHMR	002320	1672*	2348*	2351*	2367*	2441	2627	2764*	2768*	2802*	2806*	4381*		
RHOF	002310	1668*	2623	2688	2734	2781	2995	4514*	4936*					
RHSN	002326	1675*	2120	2154	2156									
RHWC	002272	1658*	2277	2316	2371	2546	2565	2570	2602	2610	2666	2676	2712	2722
		2773	2778	2810	2815	2846	2853	2956	2964	2989	3042	3051	3082	3089
		3122	3129	3162	3171	3204	3211	3247	3256	3288	3295	3329	3338	3371
		3378	3419	3442	3478	3484	3531	3540	3574	3583	3628	3637	3669	3675
		3738	3749	3777	3784	3836	3889	3979	3985	4004	4011	4045	4052	4104
		4111	4171	4179	4214	4243	4249	4300	4474	4771	4839	4906	4908	4931*
RH70	004750	1819*	1876*	1946*	2576	2613	2678	2724	3401	4127				
RH70CK	006146	1874*												
RMR =	000004	1500*												
RPTRP1	011454	2171	2184*											
RPTRP2	011562	2195	2208*											
RPVEC	002266	1424*	1856	2170	2194	2278*	2317*	2547*	2571*	2605*	2677*	2723*	2778*	2815*
		2854*	2965*	2990*	3044*	3084*	3124*	3164*	3206*	3249*	3290*	3331*	3373*	3421*
		3480*	3533*	3576*	3630*	3671*	3719*	3740*	3779*	3837*	3867*	3883*	3890*	3921*
		3940*	3981*	4006*	4047*	4072*	4106*	4136*	4173*	4207*	4244*	4281*	5008	5011*
		5014	5027*											
RPVECT	044662	1857	5025*	5027										
RP05	004746	1818*	2106*	2112*	2341	2357	2470							
RP06	004744	1817*	2096*	2103*	2339	2355	2468	2756	2795					
RP4VEC	004606	1771*	1902*	2278	2317	2547	2571	2605	2677	2723	2778	2815	2854	2965
		2990	3044	3084	3124	3164	3206	3249	3290	3331	3373	3421	3480	3533
		3576	3630	3671	3719	3740	3779	3837	3867	3883	3890	3921	3940	3981
		4006	4047	4072	4106	4136	4173	4207	4244	4281				
RTN	006634	1947	1950*											
RUN	043424	3039	3079	3119	3159	3201	3244	3285	3326	3368	3414	3473	3529	3571
		3625	3666	3735	3773	3865	3919	3976	4001	4041	4102	4165	4205	4241
		4280	4929*											
SAVER	041620	2277	2316	2371	2546	2570	2676	2722	2778	2815	2853	2964	2989	3042
		3082	3122	3162	3204	3247	3288	3329	3371	3419	3478	3531	3574	3628
		3669	3738	3777	3836	3889	3979	4004	4045	4104	4171	4214	4243	4300

```
                        4586#
SAVERE  004612          1779#  2277   2285   2316   2324   2371   2422#  2423#  2428   2442#  2448   2546   2552
                        2570   2580#  2583#  2588   2604#  2610#  2611#  2621#  2626#  2630#  2632#  2640   2676
                        2693   2722   2739   2778   2781#  2785   2815   2817#  2820   2853   2920#  2927#  2931
                        2964   2970#  2973   2989   2995#  2997   3042   3055   3082   3093   3122   3133   3162
                        3175   3204   3215   3247   3260   3288   3299   3329   3342   3371   3382   3419   3446
                        3478   3488   3531   3544   3574   3587   3628   3641   3669   3678   3738   3754   3777
                        3788   3836   3843#  3845   3889   3896#  3898   3979   3989   4004   4016   4045   4056
                        4104   4115   4171   4182   4214   4223   4243   4249#  4250#  4253#  4254#  4255#  4259
                        4300   4311   4475#  4774#  4776#  4831
SC      = 100000        1476#  2254   2421   2441   2580   2616   2621   2778   2817   2874   2926   2967   2970
                        2992   2995   3843   3896   4255
SC1     = 000100        1558#
SC10    = 001000        1561#
SC2     = 000200        1559#
SC20    = 002000        1562#
SC4     = 000400        1560#
SEECOM  002452          1749#  3838   3840   3891   3893   3941   4073   4137   4380   4506
SEEKCY  041422          3834   3887   3939   4071   4135   4505#
SELECT  004726          1806#  1840#  1842#  1845#  1908   2059   2072   4333
SELTST  010014          1993   2059#
SERCH   002434          1742#  4489
SKI     = 040000        1628#
SN      002410          1725#  2156#  2632   5061
SND1    006110          1865   1871#
SRCH    041402          4487#
SR0     = 177572         649#
SR1     = 177574         649#
SR2     = 177576         649#
SR3     = 172516         649#
STACK   = 001000         624#  1851   1919   1956   2064   2168   2192   2217   2313   2348   2351   2363   2482
                        2542   2599   2652   2712   2764   2768   2773   2802   2806   2810   2829   2951   3016
                        3024   3113   3148   3233   3315   3401   3512   3608   3700   3881   3937   4068   4127
                        4200
START   005022          1841   1844   1849#
STKLMT= 177774           624#
ST1     032534          3706#  3722#  3813#  3826
ST10    033136          3714#  3769#  3821#
ST11    033162          3715#  3772#  3822#
ST12    033416          3788   3789#
ST13    033422          3788   3797#
ST14    033440          3798   3799#
ST15    033444          3798   3813#
ST16    033532          3824   3826#
ST17    033536          3825   3833#
ST18    033766          3845   3846#
ST19    033772          3845   3860#
ST2     032554          3707#  3725#  3814#
ST20    040342          4301#
ST25    035624          4044#
ST26    034156          3871   3872#
ST27    034162          3871   3877#
ST3     032566          3708#  3728#  3815#
ST4     032606          3709#  3731#  3816#
ST5     032620          3710#  3734#  3817#
ST6     033026          3711#  3712#  3749#  3818#  3819#
```

```
ST7      033074            3754#
ST8      033100            3754   3764#
ST9      033116            3713#  3768#  3820#
SWR      001140             650#  1851#  1873   1874   1934   1972   1991   2478   4843   4979   5033#  5034   5053
                           5118#
SWREG    000176             625#  1851   1873   5033
SW0    = 000001             624#
SW00   = 000001             624#
SW01   = 000002             624#
SW02   = 000004             624#
SW03   = 000010             624#
SW04   = 000020             624#
SW05   = 000040             624#
SW06   = 000100             624#  5055
SW07   = 000200             624#  4845   4981
SW08   = 000400             624#
SW09   = 001000             624#
SW1    = 000002             624#
SW10   = 002000             624#
SW11   = 004000             624#
SW12   = 010000             624#  1874
SW13   = 020000             624#  1934   1972   1991
SW14   = 040000             624#
SW15   = 100000             624#
SW2    = 000004             624#
SW3    = 000010             624#
SW4    = 000020             624#
SW5    = 000040             624#  2478
SW6    = 000100             624#
SW7    = 000200             624#
SW8    = 000400             624#
SW9    = 001000             624#
TBITVE= 000014              624#
TDF    = 000040            1576#
TESTAD   042574            4875#
TIMCNT   042150            2177   2201   4653   4658   4715#  4727   4732
TIME1    041652            4615#
TIME2    041664            1902   4617#
TKVEC  = 000060             624#  5033#
TMP0     004754            1828#
TMP1     004756            1829#  3014#  3020#
TMP4     004760            1830#  2778#  2870#  2871#  2873   2887#  2888#  2890   2914#  2915#  2917   2967#  2992#
TN     = 000050            2064#  2217#  2313#  2363#  2482#  2542#  2599#  2652#  2712#  2773#  2810#  2829#  2951#
                           3016#  3024#  3113#  3148#  3233#  3315#  3401#  3512#  3608#  3700#  3881#  3937#  4068#
                           4127#  4200#
TOTALA   004742            1815#  1984#  1986#  2566   2603   2623   2671   2774   2850   2961
TPVEC  = 000064             624#
TRAPVE= 000034              624#  1851#  1855#
TRE    = 040000            1475#  2254   2421   2441   4255
TRK1   = 004000            1563#
TRK10  = 040000            1566#
TRK2   = 010000            1564#
TRK20  = 100000            1567#
TRK4   = 020000            1565#
TRP      006574            1944   1948#
TRTVEC= 000014               624#
```

```
TSTNM   004604          1770#  1919#  1957#  1969#  2064#  2162#  2168#  2192#  2217#  2313#  2348#  2351#  2363#
                        2482#  2542#  2599#  2652#  2712#  2764#  2768#  2773#  2802#  2806#  2810#  2829#  2951#
                        3016#  3024#  3113#  3148#  3233#  3315#  3401#  3512#  3608#  3700#  3881#  3937#  4068#
                        4127#  4200#  4879    6650   6653   6656   6658   6660   6663   6666   6669   6672   6676
                        6679   6681   6683   6685   6689
TST1    006312          1909   1916   1919#  4360
TST10   011572          2207   2217#
TST11   012352          2219   2306#
TST12   012634          2348#
TST13   012674          2345   2351#
TST14   012730          2349   2354#
TST15   013742          2361   2363   2417   2467#
TST16   014750          2343   2474   2480   2482   2542#
TST17   015406          2599#
TST2    006650          1956#
TST20   016174          2652#
TST21   016740          2652   2712#
TST22   017306          2764#
TST23   017346          2768#
TST24   017402          2773#
TST25   020062          2802#
TST26   020122          2806#
TST27   020156          2810#
TST3    006724          1961   1969#
TST30   020524          2828#
TST31   022040          2829   2951#
TST32   023102          3012   3014#
TST33   023254          3021   3024#
TST34   024142          3113#
TST35   024456          3148#
TST36   025366          3233#
TST37   026226          3315#
TST4    010030          2060   2064#  4342   4353
TST40   027136          3401#
TST41   030306          3401   3512#
TST42   031316          3608#
TST43   032346          3700#
TST44   034162          3881#
TST45   034666          3937#
TST46   036042          4068#
TST47   036556          4127#
TST5    011016          2162#
TST50   037324          4127   4199#
TST51   040550          4200   4325#
TST6    011362          2164   2167#
TST7    011472          2183   2186   2191#
TUF   = 000100          1577#
TYPDS = 104405          2039   2047   2119   2220   2239   2384   2402   2495   2515   2654   2661   2941   4216
                        4303   4328   4330   4340   4360   5104   5117#
TYPE  = 104401          1868   1869   1870   1909   1937   1947   1950   1974   1997   2038   2040   2042   2045
                        2086   2118   2120   2146   2148   2150   2164   2166   2219   2221   2238   2240   2274
                        2313   2363   2383   2385   2401   2403   2482   2484   2494   2496   2514   2516   2531
                        2542   2567   2599   2652   2655   2659   2662   2666   2712   2773   2810   2829   2837
                        2839   2903   2940   2942   2951   3016   3043   3083   3123   3163   3205   3248   3289
                        3330   3372   3420   3433   3434   3437   3438   3479   3532   3575   3629   3670   3739
                        3778   3866   3920   3937   3980   4005   4046   4068   4105   4127   4172   4206   4215
```

```
                    4217  4302  4304  4327  4329  4338  4341  4360  4879  4881  4883  4884  4887
                    4993  4995  5008  5010  5012  5014  5016  5018  5020  5026  5033  5034  5061
                    5063  5085  5087  5090  5092  5107  5117* 5118
TYPERR  050232      5058  5062*
TYPOC = 104402      1940  2041  2120  4880  4882  4994  5009  5013  5015  5017  5026  5033  5061
                    5071  5100  5117*
TYPON = 104404      5117*
TYPOS = 104403      5117*
UBUSB   004732      1808*  3402*  3432*  4128
UNIB  = 000020      1444*
UNIT    004716      1799*  1911*  1916*  2054*  2060*  2066  2089*  2092  2118  2219  2227  2238  2383
                    2401  2487  2494  2501*  2502*  2514  2520  2611  2653  2660  2940  4215  4302
                    4327  4347  4352*  4525  4933
UNITS   004676      1798*  2006  2010  2054  2079  4348
UNITSL  004730      1807*  1912*  1916  2060
UNLOAD  002424      1738*  2850  2855  2866
UNS   = 040000      1512*
UPE   = 020000      1453*
US1   = 000001      1440*
US2   = 000002      1441*
US4   = 000004      1442*
UWR   = 000010      1623*
VUF   = 000002      1622*
VJ30  = 010000      1583*
VV    = 000100      1486*  2247  2280  2281  2319  2421  2463  2530  2685  2688  2734  2778  2815
                    2835  2843  2888  2889  2915  2916  2950  2967  2992  3048  3086  3126  3168
                    3208  3253  3292  3335  3375  3424  3481  3537  3580  3634  3673  3746  3781
                    3840  3893  4049  4108  4176  4544  4562
WAITBT  041706      4638*  4647*  4654  4659  4722*  4728  4733  6650  6653
WAITPC  041702      4636*  4644*  4645*  4719*  4720*  6650  6653
WAITRE  041704      4637*  4646*  4654  4659  4663  4675  4721*  4728  4733  4737  4753  6650  6653
WAITTM  041710      4616*  4639*  4640*  4683  4693  6653
WAIT.P  041712      4640*
WAIT.T  042152      4717*  5117
WAT   = 104413      2280  2319  2463  2530  2549  2573  2607  2685  2731  2778  2815  2843  2856
                    2950  2967  2992  3016  3020  3048  3086  3126  3168  3208  3253  3292  3335
                    3375  3424  3481  3537  3580  3634  3673  3719  3746  3781  3840  3869  3885
                    3893  3923  3942  3983  4008  4049  4074  4108  4138  4176  4210  4246  4283
                    5117*
WC      002354      1709*  2285  2324  2428  2448  2552  2588  2640  2693  2739  2785  2820  2931
                    2973  2997  3055  3093  3133  3175  3215  3260  3299  3342  3382  3446  3488
                    3544  3587  3641  3678  3754  3788  3845  3898  3989  4016  4056  4115  4182
                    4223  4259  4311  4907  5061
WCE   = 040000      1454*
WCF   = 000040      1503*
WCU   = 000001      1571*
WLE   = 004000      1509*  4255
WRCHDT  002440      1744*  4041  4048  4049
WRCHEK  002436      1743*  4102  4107  4108  4165  4175  4176
WRFROM  002470      1761*  3027  3030  3039  3051  3068  3105  3116  3142  3151  3159  3171  3186
                    3198  3225  3236  3244  3256  3271  3282  3309  3318  3326  3338  3353  3365
                    3392  3406  3414  3442  3458  3470  3499  3514  3516  3518  3520  3529  3540
                    3557  3568  3600  3610  3612  3614  3616  3625  3637  3654  3664  3691  3723
                    3726  3729  3732  3735  3749  3769  3770  3798  3863  3871  3917  3925  3944
                    3948  3953  3971  3976  3985  4029  4041  4052  4078  4083  4100  4102  4111
                    4142  4147  4163  4165  4179  4203  4205  4239  4241  4278  4285
```

```
WRIDAT  002442          1745#  3159   3165   3168   3244   3250   3253   3326   3332   3335   3414   3421   3424
                        4205   4208   4241   4245
WRIFOR  002444          1746#  3039   3045   3048   3529   3534   3537   3625   3631   3634   3735   3741   3746
                        3976   3982
WRL    = 004000         1491#  4220   4307
WRU    = 000400         1579#
WSU    = 000004         1573#
XE2     007456          1989   2004#
$AUTOB  001134           650#  1873#  5033
$BDADR  001122           650#  2224#  2232#  2249#  2259#  4556#  4564#  6663   6676   6679   6681
$BDDAT  001126           650#  1961#  2269#  2319#  2778#  2815#  2880#  2896#  2967#  2992#  3048#  3086#  3126#
                        3168#  3208#  3253#  3292#  3335#  3375#  3424#  3481#  3537#  3580#  3634#  3673#  3746#
                        3781#  3840#  3893#  4049#  4108#  4176#  4402#  4403   4663#  4675#  4737#  4753#  4838#
                        4975#  6650   6653   6656   6658   6669   6672   6676   6689
$BELL   001216           650#  5033   5034
$CHARC  045636          5033#*
$CKSWR  046150          5033#  5117
$CMTAG  001100           650#  1851
$CM1   = 000006          650#
$CM2   = 000014          650#
$CM3   = 000006          650#
$CM4   = 000006          650#
$CNTLC  046721          5033#
$CNTLG  046733          5033#
$CNTLU  046726          5033#
$CRLF   001223           650#  2045   2655   2662   3433   3434   3437   3438   4883   5033   5034   5063   5087
                        5092
$DBLK   045412          5033#
$DOAGN  041076          4360#
$DTBL   045402          5033#
$ENDAD  041066           626   4360#
$ENDCT  041034          4360#
$ENDMG  041105          4338   4360#
$ENULL  041102          4341   4360#
$EOP    041000          1942   1998   4346   4360#
$EOPCT  041026          4360#
$ERFLG  001103           650#  5033#  5034#
$ERMAX  001115           650#  1851#  5033#
$ERROR  047122          1851   1950   5034#
$ERRPC  001116           650#  5034#  5069   6650   6653   6656   6658   6660   6663   6666   6669   6672   6676
                        6679   6681   6683   6685   6689
$ERRTB  001226           650#  5077
$ERRTY  047272          5034   5053#
$ERTTL  001112           650#  4329   4331#  5034#
$ESCAP  001214           650#  1851#  5033#  5034
$FILLC  001156           650#  5033
$FILLS  001155           650#  5033
$GDADR  001120           650#
$GDDAT  001124           650#  1962#  2268#  2319#  2778#  2815#  2876#  2892#  2967#  2992#  3048#  3086#  3126#
                        3168#  3208#  3253#  3292#  3335#  3375#  3424#  3481#  3537#  3580#  3634#  3673#  3746#
                        3781#  3840#  3893#  4049#  4108#  4176#  4405#  4837#  4974#  6656   6669   6672   6676
                        6689
$GET42  041056          4360#
$GTSWR  046240          5033#  5117
$HD    = 000000           604
$HIOCT  047120          5034#*
```

```
$ICNT   001104            650#  5033#
$ILLUP  051076           5118#
$INTAG  001135            650#  5033#
$ITEMB  001114            650#  5034#  5066
$LF     001224            650#  5033   5034
$LPADR  001106            650#  1851#  2064#  2951#  4886#  4889   5033#
$LPERR  001110            650#  1851#  4237#  4881   4888#  5033#  5034
$MAIL = ****** U         1851   1873   5033   5034
$MNEW   046751           5033#
$MSWR   046740           5033#
$MXCNT  045174           5033#
$NULL   001154            650#  5033
$NWTST= 000001           1919#  1956#  1969#  2064#  2162#  2167#  2191#  2217#  2306#  2348#  2351#  2354#  2467#
                         2542#  2599#  2652#  2712#  2764#  2768#  2773#  2802#  2806#  2810#  2828#  2951#  3014#
                         3024#  3113#  3148#  3233#  3315#  3401#  3512#  3608#  3700#  3881#  3937#  4068#  4127#
                         4199#  4325#
$OCNT   050642           5117#*
$OMODE  050644           5117#*
$OVER   045160           5033#
$PASS   001100            650#  2219   2363   2482   2652   2829   4200   4337#  4339   4360#  5033
$POWER  051104           5118#
$PWRAD  051072           5118#
$PWRDN  050732           1851   5118#
$PWRMG  051066           5118#
$PWRUP  051004           5118#
$QUES   001222            650#  5033   5034
$RDCHR  046512           5033#  5117
$RDDEC= ****** U         5117
$RDLIN  046602           5033#  5117
$RDOCT  046762           5034#  5117
$RDSZ = 000011           5033#
$REGAD  001160            650#
$REG0   001162            650#
$REG1   001164            650#
$REG2   001166            650#
$REG3   001170            650#
$REG4   001172            650#
$REG5   001174            650#
$RTNAD  041100           4360#
$R2A  = ****** U         5117
$SAVRE= ****** U         5117
$SAVR6  051102           5118#*
$SCOPE  044724           1851   5033#
$SETUP= 000117           1850#  1851   1873   4360   5033   5034
$SS1  = 000000           1853#
$STUP = 177777           1850#
$SVLAD  045132           5033#
$SVPC = 000200            626#
$SWR  = 167770            604#   622    650   1851   1919   1956   1969   2064   2162   2167   2191   2217   2306
                         2348   2351   2354   2467   2542   2599   2652   2712   2764   2768   2773   2802   2806
                         2810   2828   2951   3014   3024   3113   3148   3233   3315   3401   3512   3608   3700
                         3881   3937   4068   4127   4199   4325   4360   5033   5034   5118
$SWRMK= 000000            622   5033
$TIMES  001212            650#  1851#  1919#  1956#  1969#  2064#  2354#  2467#  3401#  4325#  4360#  5033#
$TKB    001146            650#  5033
$TKCNT  045642           5033#*
```

```
$TKINT  045662            1861    1970    2310    4995    5033#
$TKQEN= 045661            5033#
$TKQIN  045644            5033#*
$TKQOU  045646            5033#*
$TKQSR  045650            5033#
$TKS    001144             650#   5033#
$TKSRV  045732            5033#
$TMP0   001176             650#   2014#   2015#
$TMP1   001200             650#   1932#   2177#   2178#   2201#   2202#   3719#   3823#   6660
$TMP2   001202             650#
$TMP3   001204             650#
$TMP4   001206             650#
$TMP5   001210             650#
$TN   = 000052             604#   1909    1916    1919#   1956#   1957    1961    1969#   2060    2064#   2162#   2164    2167#
                          2168    2183    2186    2191#   2192    2207    2217#   2219    2306#   2313    2343    2344    2348#
                          2349    2351#   2354#   2361    2363    2417    2467#   2474    2480    2482    2542#   2599#   2652#
                          2712#   2764#   2768#   2773#   2802#   2806#   2810#   2828#   2829    2951#   3012    3014#   3016
                          3021    3024#   3113#   3148#   3233#   3315#   3401#   3512#   3608#   3700#   3881#   3937#   4068#
                          4127#   4199#   4200    4325#
$TPB    001152             650#   5033#
$TPFLG  001157             650#   5033
$TPS    001150             650#   5033
$TRAP   050646            1851    5117#
$TRAP2  050670            5117#
$TRP  = 000014            5117#
$TRPAD  050702            5117#
$TSTNM  001102             650#   4332#   4360#   5033#   5034
$TTYIN  046710            5033#
$TYPBN= ****** U          5117
$TYPDS  045176            5033#   5117
$TYPE   045422            5033#   5117
$TYPEC  045572            5033#
$TYPEX  045640            5033#
$TYPOC  050444            5117#
$TYPON  050460            5117#
$TYPOS  050420            5117#
$XTSTR  044736            5033#
$$GET4= 000000            4360#
$OFILL  050643            5117#*
$40CAT= ****** U          5033    5034
.     = 072136             625#    626#    629#    633#    649#    650#   1760#   1761#   1762#   1779#   1798#   1851    1868#
                          1869#   1870#   1937#   1947#   1950#   1997#   2038#   2040#   2042#   2086#   2118#   2146#   2148#
                          2150#   2164#   2166#   2221#   2484#   2494#   2496#   2531#   2837#   2942#   3437#   4215#   4217#
                          4302#   4329#   4360    4884#   4887#   5008#   5016#   5020#   5033#   5034    5061#   5114#   5118
                          6648#   6734#
```

```
ALLREG   622#  5060  5061
CHANGR   622#  2281  2319  2421  2441  2582  2583  2616  2618  2635  2688  2734  2781  2817  2923
        2926  2970  2995  3843  3896  4220  4255  4307
CHECKD   622#  2363  2482  2542  2567  2599  2773  2810  2829  3016  3043  3083  3123  3163  3205
        3248  3289  3330  3372  3420  3479  3532  3575  3629  3670  3739  3778  3866  3920  3937
        3980  4005  4046  4068  4105  4127  4172  4206
CHECKV   622#  2274  2313  2666  2712  2839  2951
CHKCNT   622#  2844  2956  2986
CKCNTV   622#  2945
CLEARA   622#  3030  3036  3076  3116  3151  3156  3195  3198  3236  3241  3279  3282  3318  3323
        3362  3365  3406  3411  3467  3470  3516  3520  3526  3565  3568  3612  3616  3622  3662
        3664  3725  3731  3765  3769  3770  3860  3863  3914  3917  3948  3971  3974  4078  4100
        4142  4163  4203  4239  4276  4278
CMPBLK   622#  3067  3104  3141  3185  3224  3270  3308  3352  3391  3457  3498  3556  3599  3653
        3690  3797  3871  3925  4028  4285
CMREGI   622#  2285  2324  2428  2448  2552  2588  2640  2693  2739  2785  2820  2931  2973  2997
        3055  3093  3133  3175  3215  3260  3299  3342  3382  3446  3488  3544  3587  3641  3678
        3754  3788  3845  3898  3989  4016  4056  4115  4182  4223  4259  4311
COMMEN   605   624#
DATACO   622#  3039  3079  3119  3159  3201  3244  3285  3326  3368  3414  3473  3529  3571  3625
        3666  3734  3772  3865  3919  3976  4001  4041  4102  4165  4205  4241  4280
DISREG   622#
DUM      622#  2274  2313  2543  2567  2673  2719  2775  2812  2850  2961  2986
ENDCOM   610   624#
ERROR    624#  1933  1965  2181  2186  2209  2225  2233  2250  2260  2270  2286  2319  2325  2394
        2415  2429  2449  2512  2553  2589  2641  2694  2740  2778  2786  2815  2821  2882  2898
        2932  2967  2974  2992  2997  3018  3048  3056  3069  3086  3094  3106  3126  3134  3143
        3168  3176  3187  3208  3216  3226  3253  3261  3272  3292  3300  3310  3335  3343  3354
        3375  3383  3393  3424  3447  3459  3481  3489  3500  3537  3545  3558  3580  3589  3602
        3634  3642  3655  3673  3680  3693  3746  3754  3781  3789  3799  3840  3846  3872  3893
        3899  3926  3990  4018  4031  4049  4057  4108  4115  4176  4183  4224  4260  4286  4312
        4386  4398  4407  4557  4565  4666  4668  4678  4687  4696  4740  4743  4754
ESCAPE   624#
FIHEAD   622#  3027  3036  3514  3518  3526  3568  3610  3614  3622  3664  3722  3728  3768  3863
        3917  3944  3971
FILLBL   622#
FLSVRE   622#  2610  2623  2627  2628  2635  2688  2734  2781  2817  2995  3051  3089  3129  3171
        3211  3256  3295  3338  3378  3442  3484  3540  3583  3637  3675  3749  3750  3784  3843
        3896  3985  4011  4052  4111  4179  4255
GETPRI   624#
GETSWR   604#   624#  1873
GOO      622#  2279  2318  2463  2530  2548  2572  2606  2680  2682  2726  2728  2778  2815  2842
        2855  2950  2966  2991  3016  3020  3045  3085  3125  3165  3207  3250  3291  3332  3374
        3421  3480  3534  3577  3631  3672  3719  3741  3780  3838  3868  3884  3891  3922  3941
        3982  4007  4048  4073  4107  4137  4175  4208  4245  4282
LOAD     622#  2566  2603  2671  2717  2774  2812  2850  2957  2961
MAKECL   622#  2348  2351  2764  2767  2802  2805
MANUAL   622#  2219  2363  2482  2652  2829  4200
MSG     1918#  1919  1955#  1956  1968#  1969  2063#  2064  2161#  2162  2166#  2167  2190#  2191  2217#
        2305#  2306  2353#  2354  2466#  2467  2541#  2542  2599#  2651#  2652  2710#  2712  2771#  2773
        2808#  2810  2828#  2951#  3014#  3024#  3112#  3113  3147#  3148  3233#  3315#  3401#  3511#  3512
        3607#  3608  3699#  3700  3880#  3881  3937#  4067#  4068  4126#  4127  4198#  4199  4325#
MULT     624#
NEWTST   624#  1919  1956  1969  2064  2162  2167  2191  2217  2306  2348  2351  2354  2467  2542
        2599  2652  2712  2764  2768  2772  2802  2806  2809  2828  2951  3014  3024  3113  3148
        3233  3315  3401  3512  3608  3700  3881  3937  4068  4127  4199  4325
```

```
OFFST    622#  3016
POP      624#  2566  2603  2671  2774  2850  2961  3967  4097  4161  4409  4433  4456  4476  4593
         4698  4758  4779  4804  4850  4912  4985  5033  5034  5118
PUSH     624#  2566  2603  2671  2774  2850  2961  3951  4081  4145  4374  4424  4449  4471  4586
         4643  4717  4768  4794  4823  4905  4962  5033  5034  5118
REPORT   624#
RFORGC   622#
RHCLEA   622#  2064  2169  2193  2217  2237  2273  2313  2363  2437  2482  2503  2542  2564  2599
         2652  2666  2712  2773  2810  2829  2839  2951  2955  3004  3016  3020  3024  3075  3113
         3148  3194  3233  3278  3315  3361  3401  3466  3512  3564  3608  3661  3700  3720  3764
         3833  3881  3913  3937  3942  4000  4040  4068  4075  4127  4139  4200  4239  4273
RH70CK   622#  3401  4127
SAVE     622#  5034
SAVTST   622#  1919  1957  1969  2162  2168  2192  2348  2351  2764  2768  2802  2806
SCH      622#
SCOPE    624#  1919  1956  1969  2064  2162  2167  2191  2217  2306  2348  2351  2354  2467  2542
         2599  2652  2712  2764  2768  2773  2802  2806  2810  2828  2951  3014  3019  3024  3113
         3148  3233  3315  3401  3512  3608  3700  3881  3937  4068  4127  4199  4325  4360
SEEKCO   622#  3834  3887  3939  4071  4134
SETPRI   624#  5033
SETTRA  5117#
SETUP    624#  1851
SKIP     622#   624#  1909  1916  1961  2060  2164  2183  2186  2207  3012  3021
SLASH    624#
SPACE    624#
SREGIS   622#  2277  2316  2371  2546  2570  2676  2722  2778  2815  2853  2964  2989  3042  3082
         3122  3162  3204  3247  3288  3329  3371  3419  3478  3531  3574  3628  3669  3738  3777
         3836  3889  3979  4004  4044  4104  4171  4214  4243  4299
STARS     616   621   624#  626   649   650  1421  1422  1461  1462  1702  1706  1878  1899  1919
         1956  1969  2024  2033  2064  2096  2114  2124  2150  2162  2167  2191  2217  2306  2339
         2346  2348  2351  2354  2355  2362  2467  2468  2475  2542  2599  2652  2712  2756  2761
         2764  2768  2773  2795  2800  2802  2806  2810  2828  2951  2985  3014  3024  3113  3148
         3233  3315  3401  3512  3608  3700  3881  3937  4068  4127  4199  4325  4360  5024  5029
         5033  5034  5117  5118
STARTT   622#  2064  2217  2313  2363  2482  2542  2599  2652  2712  2773  2810  2829  2951  3015
         3024  3113  3148  3233  3315  3401  3512  3608  3700  3881  3937  4068  4127  4200
SWRSU    624#  1851#
TJUMP    622#  2219  2343  2344  2349  2361  2363  2417  2474  2480  2482  2652  2829  3401  4127
         4230
TRMTRP  5117#
TSCLR2   622#
TSCLR5   622#
TTSTNO   622#  2064  2217  2313  2363  2482  2542  2599  2652  2712  2773  2810  2829  2951  3016
         3024  3113  3148  3233  3315  3401  3512  3608  3700  3881  3937  4068  4127  4200
TYPBIN   624#
TYPDEC   624#  4360
TYPNAM   624#
TYPNUM   624#
TYPOCS   624#
TYPOCT   624#  2120  5033
TYPTXT   624#  1867  1868  1869  1870  1909  1937  1947  1950  1974  1996  1997  2038  2040  2042
         2085  2086  2118  2120  2145  2147  2149  2164  2166  2219  2221  2238  2240  2383  2385
         2401  2403  2484  2493  2496  2514  2516  2531  2652  2659  2837  2903  2939  2942  3433
         3436  4215  4217  4301  4304  4327  4329  4879  4881  4884  4887  4992  4995  5007  5010
         5012  5014  5016  5018  5019  5025  5061
VECSET   622#  2278  2317  2547  2571  2605  2677  2723  2778  2815  2854  2965  2990  3044  3084
```

```
            3124    3164    3206    3249    3290    3331    3373    3421    3480    3533    3576    3630    3671    3719    3740
            3779    3837    3867    3883    3890    3921    3940    3981    4006    4047    4072    4106    4136    4173    4207
            4244    4281
MT          622#    2280    2319    2463    2530    2549    2573    2607    2684    2730    2778    2815    2843    2950    2967
            2992    3016    3020    3048    3086    3126    3168    3208    3253    3292    3335    3375    3424    3481    3537
            3580    3634    3673    3719    3746    3781    3840    3869    3885    3893    3923    3942    3983    4008    4049
            4074    4108    4138    4176    4210    4246    4283
MT1         622#    2280    2319    2463    2530    2549    2573    2607    2685    2731    2778    2815    2843    2950    2967
            2992    3016    3020    3048    3086    3126    3168    3208    3253    3292    3335    3375    3424    3481    3537
            3580    3634    3673    3719    3746    3781    3840    3869    3885    3893    3923    3942    3983    4008    4049
            4074    4108    4138    4176    4210    4246    4283
$$CMRE      650#
$$CMTM      650#
$$ESCA      624#
$$NEWT      624#    1919    1956    1969    2064    2162    2167    2191    2217    2306    2348    2351    2354    2467    2542
            2599    2652    2712    2764    2768    2773    2802    2806    2810    2828    2951    3014    3024    3113    3148
            3233    3315    3401    3512    3608    3700    3881    3937    4068    4127    4199    4325
$$SET       5117#
$$SKIP      624#
.EQUAT      604#     624
.HEADE      604#
.KT11       604#     649
.SETUP      604#    1849
.SWRHI      604#     622
.SWRLO      604#     622#    623
.$ACT1      604#     626
.$CATC      604#     625
.$CMTA      604#     650
.$EOP       604#    4360
.$ERRO      604#    5034
.$ERRT      604#
.$POWE      604#    5118
.$RDOC      604#    5034
.$READ      604#    5033
.$SCOP      604#    5033
.$TRAP      604#    5117
.$TYPD      604#    5033
.$TYPE      604#    5033
.$TYPO      604#    5117

. ABS.   072136      000

 ERRORS DETECTED:  0

 RMO3:CZRJIB,CZRJIB.SEQ/CRF/SOL/NL:MC:ME:CND=RMO3:CZRJIB.P11
 RUN-TIME: 44 36 2 SECONDS
 RUN-TIME RATIO: 573/83=6.9
 CORE USED:  29K  (57 PAGES)
```