RK611

DISKLESS CONTROL PART 2
CZR6BC0

AH-9102C-MC

FICHE 1 OF 1

MAR 1978
digital
MADE IN USA

```
1  1
                .REM %
                                IDENTIFICATION

                PRODUCT CODE:   AC-9100C-MC

                PRODUCT NAME:   CZR68C0 RK611 DSKLS CTRL PRT 2

                DATE:           FEB 1978

                MAINTAINER:     DIAGNOSTIC GROUP

                AUTHOR:         ROY SPITZER
```

TABLE OF CONTENTS

```
                                1.0 ABSTRACT

THE RK611 DISKLESS CONTROLLER DIAGNOSTIC: PART 2 TEST THE LOADING
OF THE DRIVE BUS MESSAGES BY EXECUTING CLASS A COMMANDS.  SOME
TESTS EXECUTE COMMANDS PARTIALLY MAINTENANCE MODE AND PARTIALLY
AT NORMAL SPEED TO FOOL THE CONTROLLER AND FORCE ERRORS.  THIS
PROGRAM DOES NOT REQUIRE THE PRESENCE OF AN RK06 DRIVE.


                                2.0 REQUIREMENTS

                                2.1 EQUIPMENT

                                        PDP-11 SYSTEM (16K CORE MEMORY)
                                        CONSOLE TERMINAL
                                        DECTAPE, PAPER TAPE READER, OR DECDISK
                                        RK611 CONTROLLER


                                2.2 PRELIMINARY PROGRAMS

                                        RK611 DISKLESS CONTROLLER DIAGNOSTIC: PART 1
                                                CZR6AC0


                                3.0 OPERATING PROCEDURES

                                3.1 LOADING PROCEDURE

THE PROGRAM CAN BE LOADED FROM PAPER TAPE USING ABSOLUTE LOADER
OR FROM ANY MEDIA SUPPORTED BY XXDP.


                                3.2 STARTING PROCEDURE

                                        LOCATION 200    - START PROGRAM

                                        LOCATION 204    - RESTART PROGRAM

                                        LOCATION 214    - REQUEST BUS ADDRESS, VECTOR ADDRESS, AND
                                                          PRIORITY MODIFICATION
                                3.3 OPTIONAL SWITCH SETTINGS

                                        SW15 -  HALT PROGRAM
                                        SW14 -  LOOP ON TEST
                                        SW13 -  INHIBIT ERROR TYPE OUT
                                        SW12 -  ABORT AFTER 20 ERRORS
                                        SW11 -  INHIBIT ITERATION COUNT
                                        SW10 -  BELL ON ERROR
                                        SW9  -  LOOP ON ERROR
                                        SW8  -  LOOP ON TEST IN SWITCHES 0-7


                                3.5 RUN TIME
```

```
                        FIRST PASS              7 SECONDS
                        SUBSEQUENT PASSES       2 MINUTES


                4.0     OPERATING PROCEDURES

                THE PROGRAM IS EXECUTED BY STARTING AT THE APPROPIATE ADDRESS.


                5.0 PROGRAM DESCRIPTION

                            **DRIVE MESSAGE LOADING

                                    TEST 1  FIRST COMMAND IN MAINT MODE

                                            INITIALIZE RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER I
                                            MODE. ISSUE SELECT DRIVE. WAIT AND MAKE SURE CS1 REMAINS
                                            THE SAME. CLOCK IN MESSAGES A AND B. MAKE SURE
                                            CORRECT MSG ARE LOADED. CHECKING IS DONE A FIELD AT A
                                            TIME.


                                    TEST 2  DRIVE SELECT BITS LOADING FOR DRIVE MESS.

                                            INITIALIZE RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER
                                            DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 2 WIT
                                            ZERO.  LOAD COMMAND AND STATUS REGISTER WITH A SELECT
                                            COMMAND.  CLOCK IN MESSAGES A AND B INTO SHIFT REGISTER.
                                            MAKE SURE CORRECT MESSAGES ARE LOADED.  REPEAT FOR DRIVE
                                            SELECT = 1-17.


                                    TEST 3  FORMAT BIT LOADING TO FOR DRIVE MESS.

                                            CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                            DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                                            A SELECT COMMAND AND 24 SECTOR MODE FORMAT.  MAKE SURE
                                            CORRECT MESSAGE IS LOADED.


                                    TEST 4  HEAD SELECT BITS LOADING FOR DRIVE MESS.

                                            CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                            DIAGNOSTIC MODE.  LOAD TRACK ADDRESS WITH ZERO.  LOAD
                                            COMMAND AND STATUS REGISTER 2 WITH ZERO.  LOAD COMMAND
                                            AND STATUS REGISTER WITH SELECT COMMAND.  CLOCK IN
                                            MESSAGE A AND B INTO SHIFT REGISTER.  MAKE SURE CORRECT
                                            MESSAGE IS LOADED.  REPEAT FOR TRACK ADDRESS = 1-7.


                                    TEST 5  MESSAGE SELECT BITS LOADING FOR DRIVE MESS.

                                            CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                            DIAGNOSTIC MODE AND ZERO IN MESSAGE SELECT BITS.  LOAD
                                            COMMAND AND STATUS REGISTER 1 WITH A SELECT COMMAND.  CL
                                            IN MESSAGE A AND B INTO SHIFT REGISTER.   MAKE SURE
```

```
                        CORRECT MESSAGE IS LOADED.  REPEAT FOR MESSAGE SELECT =


TEST 6  CLEAR DRIVE COMMAND LOADING FOR DRIVE MESS

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                        A DRIVE CLEAR.  CLOCK MESSAGE A AND B INTO SHIFT REGISTE
                        MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.  REPEAT
                        FOR 24 SECTOR FORMAT.


TEST 7  UNLOAD COMMAND LOADING FOR DRIVE MESS.

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                        AN UNLOAD COMMAND.  CLOCK IN MESSAGES A AND B INTO SHIFT
                        REGISTERS.  MAKE SURE SHIFT REGISTERS ARE LOADED CORRECT
                        REPEAT FOR 24 SECTOR FORMAT.


TEST 10 PACK ACKNOWLEDGE COMMAND LOADING FOR DRIVE MESS.

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                        A PACK ACKNOWLEDGE.  CLOCK MESSAGES A AND B INTO SHIFT
                        REGISTERS.  MAKE SURE SHIFT REGISTERS ARE LOADED CORRECT
                        REPEAT FOR 24 SECTOR FORMAT.


TEST 11 RECALIBRATE COMMAND LOADING FOR DRIVE MESS.

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                        A RECALIBRATE.  CLOCK MESSAGES A AND B INTO SHIFT REGIST
                        MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.


TEST 12 START SPINDLE COMMAND LOADING FOR DRIVE MESS.

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WIT
                        A START SPINDLE.  CLOCK MESSAGES A AND B INTO SHIFT REGI
                        MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.


TEST 13 SEEK AND CYLINDER ADD 0-777 LOADING FOR DRIVE MESS

                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        DIAGNOSTIC MODE.  LOAD ZERO IN CYLINDER ADDRESS.  LOAD
                        COMMAND AND STATUS REGISTER 1 WITH A SEEK COMMAND.
                        CLOCK IN MESSAGE A AND B INTO SHIFT REGISTER.  MAKE SURE
                        CORRECT MESSAGE IS LOADED.  REPEAT FOR CYLINDER = 1-777.


TEST 14 SEEK AND CYLINDER BIT 9 AND RK06 FOR DRIVE MESS.
```

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD 1000 IN CYLINDER ADDRESS. LOAD COMMAND AND STATUS REGISTER 1 WITH A SEEK COMMAND. CLOCK IN MESSAGE A AND B INTO SHIFT REGISTERS. MAKE SURE CYLINDER BIT 9 IN MESSAGE IN RESET. REPEAT FOR CYLINDER = 1400.

TEST 15 SEEK AND CYLINDER ADD 0,777-1777 LOADING FOR DRIVE MESS

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD 0 IN CYLINDER ADDRESS. LOAD COMMAND AND STATUS REGISTER 1 WITH SEEK COMMAND AND CDT SET. CLOCK IN MESSAGE A AND B INTO SHIFT REGISTER. MAKE SURE CYLINDER CORRECT. REPEAT FOR CYLINDER = 777-1

TEST 16 OFFSET COMMAND LOADING FOR DRIVE MESS.

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD OFFSET REGISTER WITH 0. LOAD COMMAND AND STATUS REGISTER 1 WITH AN OFFSET. CLOCK MESSAGES A AND B INTO SHIFT REGISTERS. MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY. REPEAT FOR OFFSET REGISTER = 1-377.

TEST 17 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 1)

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD CYLINDER ADDRESS REGISTER WITH 777. LOAD THE OFFSET REG TO 52. LOAD COMMAND AND STATUS REGISTER 1 WITH A SELECT. CLOCK MESSAGES A AND B INTO SHIFT REGISTERS. MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.

TEST 20 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 2)

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD CYLINDER ADDRESS REGISTER WITH 777. LOAD THE OFFSET REG TO 52. LOAD COMMAND AND STATUS REGISTER 1 WITH A PACK ACKNOWLEDGE. CLOCK MESSAGES A AND B INTO SHIFT REGISTERS. MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.

TEST 21 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 3)

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD CYLINDER ADDRESS REGISTER WITH 777. LOAD THE OFFSET REG TO 52. LOAD COMMAND

```
                                        AND STATUS REGISTER 1 WITH A CLEAR DRIVE.  CLOCK
                                        MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
                                        SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
                                        ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.


                                TEST 22 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 4)

                                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
                                        WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
                                        AND STATUS REGISTER 1 WITH AN UNLOAD.  CLOCK
                                        MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
                                        SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
                                        ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.


                                TEST 23 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 5)

                                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
                                        WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
                                        AND STATUS REGISTER 1 WITH A START SPINDLE.  CLOCK
                                        MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
                                        SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
                                        ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.


                                TEST 24 CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 6)

                                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
                                        WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
                                        AND STATUS REGISTER 1 WITH A RECALIBRATE.  CLOCK
                                        MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
                                        SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
                                        ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.


                                TEST 25 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 1)

                                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                        COMMAND AND STATUS REGISTER 1 WITH A PACK ACKNOWLEDGE.
                                        CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                        MESSAGE SELECT BITS ARE CLEARED.


                                TEST 26 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 2)

                                        CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                        COMMAND AND STATUS REGISTER 1 WITH A DRIVE CLEAR.
                                        CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                        MESSAGE SELECT BITS ARE CLEARED.
```

```
                        TEST 27 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 3)

                                CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                COMMAND AND STATUS REGISTER 1 WITH AN UNLOAD.
                                CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                MESSAGE SELECT BITS ARE CLEARED.


                        TEST 30 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 4)

                                CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                COMMAND AND STATUS REGISTER 1 WITH A START SPINDLE.
                                CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                MESSAGE SELECT BITS ARE CLEARED.


                        TEST 31 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 5)

                                CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                COMMAND AND STATUS REGISTER 1 WITH A RECALIBRATE.
                                CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                MESSAGE SELECT BITS ARE CLEARED.


                        TEST 32 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 6)

                                CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                COMMAND AND STATUS REGISTER 1 WITH A OFFSET.
                                CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                MESSAGE SELECT BITS ARE CLEARED.


                        TEST 33 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 7)

                                CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                COMMAND AND STATUS REGISTER 1 WITH A SEEK.
                                CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                MESSAGE SELECT BITS ARE CLEARED.


                **DRIVE MESSAGE LOOPBACK AND PARITY GENERATION TESTS

                        TEST 34 DRIVE MESSAGE LOOPBACK

                                CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER
                                IN DIAGNOSTIC MODE INDICATING MESSAGE 3. LOAD COMMAND
                                STATUS REGISTER FOR DRIVE 5. LOAD COMMAND AND STATUS
                                REGISTER 1 WITH A SELECT COMMAND.  CLOCK 4 BITS
                                THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY THAT BITS
                                ARE INDEED LOOPED BACK.
```

```
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
```

TEST 35 DRIVE MESSAGE SHIFT

CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD CYLINDER ADDRESS WITH 441. LOAD HEAD ADDRESS WITH 1. LOAD COMMAND AND STATUS REGISTER 1 WITH A SEEK IN 24 SECTOR MODE. CLOCK 8 BITS THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY THAT BITS ARE SHIFTED PROPERLY.

TEST 36 DRIVE MESSAGE PARITY PRECONDITIONING

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 W A SELECT COMMAND. CLOCK ALL 16 BITS THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY PARITY HAS BEEN PRECONDIT PROPERLY. REPEAT FOR BAD PARITY GENERATION.

TEST 37 ODD DRIVE MESSAGE PARITY GENERATION

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE AND MESSAGE SELECT = 1. LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE SELECT = 1. LOAD COMMAND AND STATUS REGISTER 1 WITH A SELECT COMMAND. VERIFY THAT PARITY HAS BEEN GENERATED CORRECTLY. REPEAT FOR MESSAGE SELECT = DRIVE SELECT = 2-17.

TEST 40 DRIVE MESSAGE PARITY INTERACTION

CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE SELECT = 1. LOAD COMMAND AND STATUS REGISTER WITH A SELECT COMMAND. VERIFY THAT THE CORRECT PARITY IS GENERATED FOR BOTH MESSAGES. REPEAT FOR MESSAGE SELECT = 1 AND DRIVE SELECT = 0.

TEST 41 EVEN DRIVE MESSAGE PARITY GENERATION

CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN DIAGNOSTIC MODE AND MESSAGE SELECT = 1 AND BAD PARITY SET. LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE SELECT = 1. LOAD COMMAND AND STATUS REGISTER SELECT COMMAND. VERIFY THAT EVEN PARITY IS GENERATED. REPEAT FOR MESSAGE SELECT = DRIVE SELECT = 2-17.

**CLASS A COMMAND EXECUTION

TEST 42 RELEASE COMMAND IN DIAGNOSTIC MODE

```
                        CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                        PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND
                        STATUS REGISTER 2 WITH DRIVE SELECT = 10. LOAD
                        COMMAND AND STATUS REGISTER 1 WITH A SELECT.
                        CLOCK COMMAND TO COMPLETION. MAKE SURE UNIT
                        FIELD ERROR DOES NOT SET (SACK HIGH). REPEAT FOR
                        DRIVE SELECT = 11-17.


TEST 43 SELECT COMMAND IN DIAGNOSTIC MODE

                        CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                        PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND
                        STATUS REGISTER 2 WITH DRIVE SELECT = 0. LOAD
                        COMMAND AND STATUS REGISTER 1 WITH A SELECT.
                        CLOCK COMMAND TO COMPLETION. MAKE SURE MESSAGE SHIFT IS
                        NOT DONE DURING THE RECEIVE CYCLE OF DRIVE MESSAGE.
                        MAKE SURE NO ERRORS SET. REPEAT FOR DRIVE SELECT = 1-7.


TEST 44 RELEASE COMMAND IN NORMAL MODE

                        CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                        LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE SELECT = 1
                        LOAD COMMAND AND STATUS REGISTER 1 WITH A SELECT.
                        MAKE SURE NO ERRORS OCCUR. REPEAT FOR DRIVE
                        SELECT = 11-17


TEST 45 INTERRUPT AT COMMAND COMPLETION

                        CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                        LOWER PROCESSOR PRIORITY TO ZERO. ISSUE A RELEASE
                        COMMAND WITH INTERRUPT ENABLE SET. MAKE SURE
                        INTERRUPT OCCURS. LOWER PRIORITY AFTER INTERRUPT
                        AND MAKE SURE INTERRUPT HAS CLEARED.

                        LOWER PROCESSOR PRIORITY TO ZERO.  REISSUE RELEASE
                        WITH INTERRUPT ENABLE RESET.  MAKE SURE NO INTERRUPT
                        OCCURS.  SET INTERRUPT ENABLE AND MAKE SURE NO
                        INTERRUPT OCCURS.


TEST 46 GO CLEAR OF SILO

                        CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                        WRITE ONE WORD INTO THE SILO.  ISSUE A RELEASE COMMAND
                        WITH INTERRUPT ENABLE RESET.  WAIT FOR READY.
                        READ THE DATA BUFFER TO MAKE SURE THE SILO HAS BEEN
                        CLEARED. (DATA LATE SET AFTER READ OF DATA BUFFER)


TEST 47 SEEK COMMAND IN DIAGNOSTIC MODE

                        CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
```

```
                                        PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT
                                        24 SECTOR FORMAT TO CYLINDER 1714, HEAD 7, DRIVE 0.
                                        MAKE SURE NO STATUS BITS ARE SET AND NO ERROR
                                        BITS ARE SET.


                                **ERROR AND STATUS BIT FORCING WITH DRIVE MESSAGES

                                        TEST 50 DRIVE STATUS FROM SHIFT REGISTER

                                                CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
                                                RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
                                                TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 757, HEAD 1,
                                                DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS
                                                6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE SPEED LOSS,
                                                DRIVE AVAILABLE, VOLUME VALID, OFFSET, DRIVE READY,
                                                AND WRITE LOCK ARE SET.


                                        TEST 51 DRIVE AVAILABLE SETTING

                                                CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                                PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK0
                                                26 SECTOR FORMAT TO CYLINDER 2, HEAD 0, DRIVE 0.
                                                CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
                                                TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE
                                                AVAILIABLE SETS.


                                        TEST 52 DRIVE BUS PARITY ERROR

                                                CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                                PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
                                                TO A RK06, 26 SECTOR FORMAT TO CYLINDER 3, HEAD 0,
                                                DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
                                                TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE BUS
                                                PARITY, DRIVE AVAILIABLE, AND CONTROLLER ERROR ARE SET.

                                TEST 53 DRIVE AVAILABLE RESET ERROR

                                                CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                                PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SELECT
                                                TO A RK06, 26 SECTOR FORMAT, AND DRIVE 0.
                                                CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
                                                TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE
                                                IS RESET AND CONTROLLER ERROR IS SET.


                                        TEST 54 CDT SET DRIVE TYPE

                                                CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                                PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
                                                WITH CDT SET. 26 SECTOR FORMAT, TO CYLINDER 23,
                                                HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
                                                UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE
```

AND MAKE SURE ONLY DRIVE AVAILIABLE SETS.

TEST 55 CDT SET AND DRIVE TYPE ERROR

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
WITH CDT SET, 26 SECTOR FORMAT, TO CYLINDER 2,
HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE
AND MAKE SURE DRIVE AVAILIABLE, DRIVE TYPE ERROR,
AND CONTROLLER ERROR SET.

TEST 56 RK06 AND DRIVE TYPE ERROR

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
TO A RK06, 26 SECTOR FORMAT, TO CYLINDER 23,
HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC
MODE AND MAKE SURE DRIVE AVAILIABLE, DRIVE TYPE ERROR,
AND CONTROLLER ERROR SETS.

TEST 57 SPEED LOSS FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK0
26 SECTOR FORMAT, TO CYLINDER 3, HEAD 1, DRIVE 0.
CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN
OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE AND
SPEED LOSS ARE SET.

TEST 60 DRIVE OFF TRACK FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK0
26 SECTOR FORMAT, TO CYLINDER 3, HEAD 2, DRIVE 0.
CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE
AND DRIVE OFF TRACK ARE SET.

TEST 61 WRITE LOCK ERROR FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A PACK ACKNOWLE
TO A RK06, 26 SECTOR FORMAT, WITH CYLINDER 0,
HEAD 1, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL
PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
SURE SPEED LOSS, WRITE LOCK ERROR AND CONTROLLER ERROR
ARE SET WITH DRIVE AVAILIABLE RESET.

TEST 62 SEEK INCOMPLETE

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE AN UNLOAD
TO A RK06, 26 SECTOR FORMAT, WITH CYLINDER 0,
HEAD 1, DRIVE 0, CLOCK IN DIAGNOSTIC MODE UNTIL
PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
SURE SPEED LOSS, SEEK INCOMPLETE, AND CONTROLLER ERROR
ARE SET WITH DRIVE AVAILIABLE RESET.

TEST 63 NON-EXECUTABLE DRIVE FUNCTION FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR
PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE
A DRIVE CLEAR TO A RK06, 26 SECTOR FORMAT,
WITH CYLINDER 0, HEAD 1, DRIVE 0. CLOCK IN DIAGNOSTIC
MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC
MODE AND MAKE SURE SPEED LOSS, NON-EXECUTABLE DRIVE FUNC
CONTROLLER ERROR ARE SET WITH DRIVE AVAILIABLE RESET.

TEST 64 AC LOW AND C-D PARITY FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR, PUT RK611
CONTROLLER IN DIAGNOSTIC MODE. ISSUE A START SPINDLE
TO AN RK06, IN 24 SECTOR FORMAT, CYLINDER 0, HEAD 0,
DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6
TURN OFF DIAGNOSTIC MODE AND MAKE SURE AC LOW, DRIVE

DETECTED SERCOM PARITY, AND CONTROLLER ERROR SET WITH
DRIVE AVAILABLE RESET.

TEST 65 ILLEGAL DISK ADDRESS ERROR FROM SHIFT REG.

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A RECALIBRAT
TO AN RK06, IN 26 SECTOR FORMAT, CYLINDER 0, HEAD 1,
DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE
ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE
SPEED LOSS, ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER
ERROR ARE SET WITH DRIVE AVAILABLE RESET.

TEST 66 IDAE DETECTION IN RK611 CONTROLLER (PART 1)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A
SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 1003,
HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL
PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
SURE DRIVE AVAILABLE, ILLEGAL DISK ADDRESS ERROR,
AND CONTROLLER ERROR ARE SET.

TEST 67 IDAE DETECTION IN RK611 CONTROLLER (PART 2)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 1022, HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILABLE AND POSITIONING IN PROGRESS ARE SET WITH ILLEGAL DISK ADDRESS ERROR RESET.

TEST 70 IDAE DETECTION IN RK611 CONTROLLER (PART 3)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2, HEAD 3, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILABLE, DRIVE OFF TRACK, SPEED LOSS, ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER ERROR ARE SET.

TEST 71 IDAE DETECTION IN RK611 CONTROLLER (PART4)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 3, HEAD 4, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILABLE, UNSAFE, ILLEGAL DISK ADDRESS ERROR AND CONTROLLER ERROR ARE SET.

TEST 72 IDAE DETECTION IN RK611 CONTROLLER (PART 5)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 23, HEAD 5, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILABLE, UNSAFE, SPEED LOSS, ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER ERROR ARE SET.

TEST 73 IDAE DETECTION IN RK611 CONTROLLER (PART 6)

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 23, HEAD 6, DRIVE 0. CLOCK IN DIAGNOS IC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILABLE, UNSAFE, DRIVE OFF TRACK, ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER CLEAR ARE SET.

```
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
```

TEST 74 NON-STANDARD MESSAGE RECEIVING

CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET IN 24 SECTOR FORMAT, CYLINDER 1757, HEAD 7, DRIVE 1. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6 TURN OFF DIAGNOSTIC MODE AND MAKE SURE NO ERRORS SET AND DRIVE STATUS IS NOT REPORTED. REPEAT FOR DRIVES 2 AND 4.

TEST 75 DRIVE BUS PARITY ON NON-STANDARD MESSAGE

CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2, HEAD 0, DRIVE 1. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE BUS PARITY ERROR AND CONTROLLER ERROR SETS.

TEST 76 NON-EXISTENT DRIVE (DRIVE MESSAGE TIME OUT)

CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SELECT TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 0, HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 5. TURN OFF DIAGNOSTIC MODE AND MAKE SURE NON-EXISTENT DRIVE AND CONTROLLER ERROR ARE SET. THIS TEST CHECKS NON-EXISTENT DRIVE DUE TO DRIVE MESSAGE TIME OUT.

TEST 77 NON-EXISTENT DRIVE AND NO SACK

CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SELECT TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 0, HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 4. TURN OFF DIAGNOSTIC MODE AND MAKE SURE NON-EXISTENT DRIVE AND CONTROLLER ERROR ARE SET.

THIS TEST EXERCISES THE NON-EXISTENT DRIVE LOGIC DUE TO RELEASE BIT RESET AND SACK RESET BUT THE PASSING

OF THIS TEST DOES GUARENTEE THAT THIS SITUATION DID INDEED CAUSE A NON-EXISTENT DRIVE.

**ILLEGAL FUNCTION CODE TEST

TEST 100 ILLEGAL FUNCTION CODE

CLEAR RK611 WITH A CONTROLLER CLEAR. ISSUE AN ILLEGAL COMMAND IN NORMAL MODE AND MAKE SURE COMMAND FINISHES

```
                                                    SETTING CONTROLLER READY WITH PROPER ERROR CONDITIONS.


                        6.0 ERROR REPORTING

                        THE GENERAL FORMAT OF ERROR REPORTS IS:

                        OPERATION DESCRIPTION AND ERROR DESCRIPTION
                        TEST    ERROR
                        NUM     PC
                        XXXXXX  YYYYYY
                          EXPECT  ACTUAL  OTHER PERTENANT
                          REG     REG     INFORMATION
                          ZZZZZZ  WWWWWW  AAAAAA

                        NOTE:   MOVE THAN ONE SET OF EXPECT/ACTUAL REGISTERS MAY BE
                                PRINTED OUT. OTHER PERTENANT INFORMATION MAY CONSIST
                                OF MORE THAN ONE WORD.
                        %
```

```
                                ; *** REV 003 ***
                                .TITLE  CZR68CO RK611 DSKLS CTRL PRT2
                                ;*COPYRIGHT (C) 1976,1977
                                ;*DIGITAL EQUIPMENT CORP.
                                ;*MAYNARD, MASS. 01754
                                ;*
                                ;*PROGRAM BY ROY SPITZER
                                ;*
                                ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                                ;*PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
                                ;*
                                .SBTTL  OPERATIONAL SWITCH SETTINGS
                                ;*
                                ;*      SWITCH                  USE
                                ;*      ------          --------------------
                                ;*        15            HALT ON ERROR
                                ;*        14            LOOP ON TEST
                                ;*        13            INHIBIT ERROR TYPEOUTS
                                ;*        12            ABORT PROGRAM AFTER 20 ERRORS
                                ;*        11            INHIBIT ITERATIONS
                                ;*        10            BELL ON ERROR
                                ;*         9            LOOP ON ERROR
                                ;*         8            LOOP ON TEST IN SWR<7:0>
                                .SBTTL  BASIC DEFINITIONS

                                ;*INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
001100                          STACK=  1100
                                .EQUIV  EMT,ERROR       ;;BASIC DEFINITION OF ERROR CALL
                                .EQUIV  IOT,SCOPE       ;;BASIC DEFINITION OF SCOPE CALL

                                ;*MISCELLANEOUS DEFINITIONS
000011                          HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                          LF=     12              ;;CODE FOR LINE FEED
000015                          CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                          CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                          PS=     177776          ;;PROCESSOR STATUS WORD
                                .EQUIV  PS,PSW
177774                          STKLMT= 177774          ;;STACK LIMIT REGISTER
177772                          PIRQ=   177772          ;;PROGRAM INTERRUPT REQUEST REGISTER
177570                          DSWR=   177570          ;;HARDWARE SWITCH REGISTER
177570                          DDISP=  177570          ;;HARDWARE DISPLAY REGISTER

                                ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                          R0=     %0              ;;GENERAL REGISTER
000001                          R1=     %1              ;;GENERAL REGISTER
000002                          R2=     %2              ;;GENERAL REGISTER
000003                          R3=     %3              ;;GENERAL REGISTER
000004                          R4=     %4              ;;GENERAL REGISTER
000005                          R5=     %5              ;;GENERAL REGISTER
000006                          R6=     %6              ;;GENERAL REGISTER
000007                          R7=     %7              ;;GENERAL REGISTER
000006                          SP=     %6              ;;STACK POINTER
000007                          PC=     %7              ;;PROGRAM COUNTER

                                ;*PRIORITY LEVEL DEFINITIONS
000000                          PR0=    0               ;;PRIORITY LEVEL 0
```

```
                        
000040                  PR1=    40              ;;PRIORITY LEVEL 1
000100                  PR2=    100             ;;PRIORITY LEVEL 2
000140                  PR3=    140             ;;PRIORITY LEVEL 3
000200                  PR4=    200             ;;PRIORITY LEVEL 4
000240                  PR5=    240             ;;PRIORITY LEVEL 5
000300                  PR6=    300             ;;PRIORITY LEVEL 6
000340                  PR7=    340             ;;PRIORITY LEVEL 7
                        
                        ;*"SWITCH REGISTER" SWITCH DEFINITIONS
100000                  SW15=   100000
040000                  SW14=   40000
020000                  SW13=   20000
010000                  SW12=   10000
004000                  SW11=   4000
002000                  SW10=   2000
001000                  SW09=   1000
000400                  SW08=   400
000200                  SW07=   200
000100                  SW06=   100
000040                  SW05=   40
000020                  SW04=   20
000010                  SW03=   10
000004                  SW02=   4
000002                  SW01=   2
000001                  SW00=   1
                        .EQUIV  SW09,SW9
                        .EQUIV  SW08,SW8
                        .EQUIV  SW07,SW7
                        .EQUIV  SW06,SW6
                        .EQUIV  SW05,SW5
                        .EQUIV  SW04,SW4
                        .EQUIV  SW03,SW3
                        .EQUIV  SW02,SW2
                        .EQUIV  SW01,SW1
                        .EQUIV  SW00,SW0
                        
                        ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
100000                  BIT15=  100000
040000                  BIT14=  40000
020000                  BIT13=  20000
010000                  BIT12=  10000
004000                  BIT11=  4000
002000                  BIT10=  2000
001000                  BIT09=  1000
000400                  BIT08=  400
000200                  BIT07=  200
000100                  BIT06=  100
000040                  BIT05=  40
000020                  BIT04=  20
000010                  BIT03=  10
000004                  BIT02=  4
000002                  BIT01=  2
000001                  BIT00=  1
                        .EQUIV  BIT09,BIT9
                        .EQUIV  BIT08,BIT8
                        .EQUIV  BIT07,BIT7
```

```
                        .EQUIV  BIT06,BIT6
                        .EQUIV  BIT05,BIT5
                        .EQUIV  BIT04,BIT4
                        .EQUIV  BIT03,BIT3
                        .EQUIV  BIT02,BIT2
                        .EQUIV  BIT01,BIT1
                        .EQUIV  BIT00,BIT0

                        ;*BASIC "CPU" TRAP VECTOR ADDRESSES
000004                  ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
000010                  RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
000014                  TBITVEC=14              ;;"T" BIT
000014                  TRTVEC= 14              ;;TRACE TRAP
000014                  BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
000020                  IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
000024                  PWRVEC= 24              ;;POWER FAIL
000030                  EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
000034                  TRAPVEC=34              ;;"TRAP" TRAP
000060                  TKVEC=  60              ;;TTY KEYBOARD VECTOR
000064                  TPVEC=  64              ;;TTY PRINTER VECTOR
000240                  PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
000114                  MEMVEC= 114                     ;VECTOR FOR MEMORY CHECK ENABLE
172100                  MEMBAS= 172100                  ;BUS ADDRESS FOR MEMORY CHECK ENABLE
000001                  PAR.EN= 1                       ;MEMORY ENABLE PARITY CHECKING
120210                  AVECT1= 120210                  ;DEFINE RK611 VECTOR ADDRESS
000005                  APRIOR= 5                       ;DEFINE RK611 PRIORITY
177440                  ABASE=  177440                  ;DEFINE BASE OF RK611 REGISTERS

                        .SBTTL  RK611 CONTROLLER REGISTER DEFINITION

000000                  RKCS1=  0               ;CONTROL AND STATUS REGISTER 1
000002                  RKWC=   2               ;WORD COUNT REGISTER
000004                  RKBA=   4               ;BUS ADDRESS REGISTER
000006                  RKDA=   6               ;DESIRED TRACK SECTOR REGISTER
000010                  RKCS2=  10              ;CONTROL AND STATUS REGISTER 2
000012                  RKDS=   12              ;DRIVE STATUS REGISTER
000014                  RKER=   14              ;ERROR REGISTER
000016                  RKASOF= 16              ;ATTENTION SUMMARY AND OFFSET REGISTER
000020                  RKDCYL= 20              ;DESIRED CYLINDER REGISTER
000024                  RKDB=   24              ;DATA BUFFER
000026                  RKMR1=  26              ;MAINTENANCE REGISTER 1
000034                  RKMR2=  34              ;MAINTENANCE REGISTER 2
000036                  RKMR3=  36              ;MAINTENANCE REGISTER 3
000030                  RKECPS= 30              ;ECC POSITION INFORMATION
000032                  RKECPT= 32              ;ECC PATTERN INFORMATION
000022                  RKSPAR= 22              ;SPARE REGISTER

                        .SBTTL  DRIVE COMMANDS

000001                  SELDRV= 01              ;SELECT DRIVE
000003                  PACK=   03              ;PACK ACKNOWLEDGE
000005                  CLEAR=  05              ;DRIVE CLEAR
000007                  UNLOAD= 07              ;UNLOAD
000011                  SRTSPL= 11              ;START SPINDLE
000013                  RECAL=  13              ;RECALIBRATE
000015                  OFFSET= 15              ;OFFSET
```

```
                        DRIVE COMMANDS

000017          SEEK=   17                      ;SEEK
000021          RDDATA= 21                      ;READ DATA
000023          WRDATA= 23                      ;WRITE DATA
000025          RDHEAD= 25                      ;READ HEADER
000027          WRHEAD= 27                      ;WRITE HEADER AND DATA
000031          WRTCHK= 31                      ;WRITE CHECK
000300          INTR=   300                     ;GENERATE INTERRUPT TO CPU

                .SBTTL  CONTROL AND STATUS REGISTER 1 BITS

000001          GO=     BIT0                    ;GO BIT
000100          IE=     BIT6                    ;INTERRUPT ENABLE
000200          RDY=    BIT7                    ;CONTROLLER READY
000400          BA16=   BIT8                    ;BUS ADDRESS BIT 16
001000          BA17=   BIT9                    ;BUS ADDRESS BIT 17
002000          CDT=    BIT10                   ;CONTROLLER DRIVE TYPE (0=RK06)
004000          CTO=    BIT11                   ;CONTROLLER TIMED OUT WAITING FOR
                                                ;   DRIVE RESPONSE
010000          CFMT=   BIT12                   ;CONTROLLER DRIVE FORMAT (0=26 SECTOR, 1=24 SECTOR)
020000          SPAR=   BIT13                   ;DRIVE BUS PARITY ERRO? DETECTED BY CONTROLLER
040000          DI=     BIT14                   ;DRIVE INTERRUPT
100000          CERR=   BIT15                   ;CONTROLLER ERROR
100000          CCLR=   BIT15                   ;CONTROLLER CLEAR

                .SBTTL  CONTROL AND STATUS REGISTER 2 BITS

000007          DRVMSK= 7                       ;MASK FOR DRIVE SELECTION CODE
000010          RLS=    BIT3                    ;DESELECT OR RELEASE DRIVE IN BITS 0-2
000020          BAI=    BIT4                    ;BUS ADDRESS INCREMENT INHIBIT
000040          SCLR=   BIT5                    ;CLEAR CONTROLLER AND ALL DRIVES
000100          IR=     BIT6                    ;INPUT READY
000200          OR=     BIT7                    ;OUTPUT READY
000400          UFE=    BIT8                    ;UNIT FIELD ERROR
001000          MDS=    BIT9                    ;MULTIPLE DRIVE SELECT
002000          PGE=    BIT10                   ;PROGRAMMING ERROR
004000          NEM=    BIT11                   ;NON-EXISTENT MEMORY
010000          NED=    BIT12                   ;NON-EXISTENT DRIVE
020000          UPE=    BIT13                   ;UNIBUS PARITY ERROR
040000          WCE=    BIT14                   ;WRITE CHECK ERROR
100000          DLT=    BIT15                   ;DATA LATE ERROR

                .SBTTL  ERROR REGISTER BIT DEFINITION

000001          ILF=    BIT0                    ;ILLEGAL FUNCTION CODE
000002          SKI=    BIT1                    ;SEEK INCOMPLETE
000004          NXF=    BIT2                    ;NON-EXECUTABLE DRIVE FUNCTION
000010          DRPAR=  BIT3                    ;DRIVE DETECTED DRIVE BUS PARITY ERROR
000020          FMTE=   BIT4                    ;FORMAT ERROR
000040          DTYE=   BIT5                    ;DRIVE TYPE ERROR
000100          ECH=    BIT6                    ;ECC HARD
000200          BSE=    BIT7                    ;BAD SECTOR ERROR
000400          HVRC=   BIT8                    ;HEADER VRC ERRROR
001000          COE=    BIT9                    ;CYLINDER ADDRESS OVERFLOW ERROR
002000          IDAE=   BIT10                   ;INVALID DISK ADDRESS ERROR
004000          WLE=    BIT11                   ;WRITE LOCK ERROR
010000          DTE=    BIT12                   ;DRIVE TIMING ERROR
```

```
                020000          OPI=    BIT13                   ;OPERATION (SEARCH) INCOMPLETE
                040000          UNS=    BIT14                   ;DRIVE UNSAFE
                100000          DCK=    BIT15                   ;DATA CHECK

                                .SBTTL  STATUS REGISTER BIT DEFINITION

                000001          DRA=    BIT0                    ;DRIVE AVAILABLE (CONTROLLER IS SET IF
                                                                ;  THIS BIT IS RESET)
                000004          OFST=   BIT2                    ;DRIVE OFFSET
                000010          ACLO=   BIT3                    ;AC LOW
                000020          SPDLSS= BIT4                    ;SPEED LOSS
                000040          DROT=   BIT5                    ;DRIVE OFF TRACK
                000100          VV=     BIT6                    ;VOLUME VALID
                000200          DRDY=   BIT7                    ;DRIVE READY
                000400          DDT=    BIT8                    ;DRIVE TYPE (0=RK06)
                004000          WRL=    BIT11                   ;WRITE LOCK
                020000          PIP=    BIT13                   ;POSITIONING IN PROGRESS
                040000          DSC=    BIT14                   ;DRIVE STATUS CHANGE
                100000          SVAL=   BIT15                   ;STATUS VALID

                                .SBTTL  MAINTENANCE REGISTER 1 BIT DEFINITION

                000017          MESMSK= 17                      ;MESSAGE MASK

                000020          PAT=    BIT4                    ;FORCE EVEN PARITY ON DRIVE MESSAGE LINES
                000040          DMD=    BIT5                    ;DIAGNOSTIC MODE
                000100          MSP=    BIT6                    ;MAINTENANCE SECTOR PULSE
                000200          MIND=   BIT7                    ;MAINTENANCE INDEX
                000400          MCLK=   BIT8                    ;MAINTENANCE CLOCK
                001000          MERD=   BIT9                    ;MAINTENANCE ENCODED READ DATA
                002000          MEWD=   BIT10                   ;MAINTENACNE ENCODED WRITE DATA
                004000          PCA=    BIT11                   ;PRECOMPENSATION ADVANCE
                010000          PCD=    BIT12                   ;PRECOMPENSATION DELAY
                020000          ECCW=   BIT13                   ;ECC WORD IS BEING READ OR WRITTEN
                040000          WRTGAT= BIT14                   ;WRITE GATE
                100000          RDGATE= BIT15                   ;READ GATE

                                .SBTTL  TRANSMITTED MESSAGE A

                000020          S.SEEK= BIT4                    ;SEEK COMMAND
                000040          S.RECL= BIT5                    ;RECALIBRATE COMMAND
                000100          S.STSP= BIT6                    ;START SPINDLE COMMAND
                000200          S.RTC=  BIT7                    ;DRIVE RETURN TO CENTERLINE COMMAND
                000400          S.CLR=  BIT8                    ;CLEAR ERROR AND DSC
                001000          S.FMT=  BIT9                    ;FORMAT
                002000          S.UNLD= BIT10                   ;UNLOAD
                004000          S.PACK= BIT11                   ;SET VOLUME VALID (PACK ACKNOWNLEDGE)
                                .SBTTL  TRAP CATCHER

                000000                  .=0
                                ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
                                ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
                                ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
                000174                  .=174
000174  000000  DISPREG:  .WORD  0       ;;SOFTWARE DISPLAY REGISTER
000176  000000  SWREG:    .WORD  0       ;;SOFTWARE SWITCH REGISTER
```

```
1079
1080  000200  000137  004316          .SBTTL  STARTING ADDRESS(ES)
1081  000204  000137  004306                  JMP     @#START ;;JUMP TO STARTING ADDRESS OF PROGRAM
1082          000214                          JMP     RESTRT          ;JUMP TO RESTART ROUTINE
1083  000214  000137  004276                  .=214
1084                                          JMP     PARM            ;JUMP TO OPERATOR ASSIGNED PARMETERS
1085                                  .SBTTL  ACT11 HOOKS
1086
1087                                  ;;******************************************************************
1088          000220                  ;HOOKS REQUIRED BY ACT11
1089          000046                          $SVPC=.                 ;SAVE PC
1090  000046  042340                          .=46
1091          000052                          $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
1092  000052  000000                          .=52
1093          000220                          .WORD   0               ;;2)SET LOC.52 TO ZERO
1094          001000                          .=$SVPC                 ;; RESTORE PC
1095                                          .=1000
1096                                  .SBTTL  APT PARAMETER BLOCK
1097
1098                                  ;;******************************************************************
1099                                  ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
1100          001000                  ;;******************************************************************
1101          000024                          .$X=.   ;;SAVE CURRENT LOCATION
1102  000024  000200                          .=24    ;;SET POWER FAIL TO POINT TO START OF PROGRAM
1103          000044                          200     ;;FOR APT START UP
1104  000044  001000                          .=44    ;;POINT TO APT INDIRECT ADDRESS PNTR.
1105          001000                          $APTHDR ;;POINT TO APT HEADER BLOCK
1106                                          .=.$X   ;;RESET LOCATION COUNTER
1107                                  ;;******************************************************************
1108                                  ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
1109                                  ;INTERFACE SPEC.
1110  001000
1111  001000  000000                  $APTHD:
1112  001002  001214                  $HIBTS: .WORD   0       ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
1113  001004  000001                  $MBADR: .WORD   $MAIL   ;;ADDRESS OF APT MAILBOX (BITS 0-15)
1114  001006  000007                  $TSTM:  .WORD   1       ;;RUN TIM OF LONGEST TEST
1115  001010  000007                  $PASTM: .WORD   7       ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
1116  001012  000032                  $UNITM: .WORD   7       ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
                                          .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
                                .SBTTL  COMMON TAGS

                                ;;**********************************************************************
                                ;*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
                                ;*USED IN THE PROGRAM.

                        001100          .=1100
001100                  $CMTAG:                         ;;START OF COMMON TAGS
001100  000000                  .WORD   0
001102  000                     $TSTNM: .BYTE   0       ;;CONTAINS THE TEST NUMBER
001103  000                     $ERFLG: .BYTE   0       ;;CONTAINS ERROR FLAG
001104  000000                  $ICNT:  .WORD   0       ;;CONTAINS SUBTEST ITERATION COUNT
001106  000000                  $LPADR: .WORD   0       ;;CONTAINS SCOPE LOOP ADDRESS
001110  000000                  $LPERR: .WORD   0       ;;CONTAINS SCOPE RETURN FOR ERRORS
001112  000000                  $ERTTL: .WORD   0       ;;CONTAINS TOTAL ERRORS DETECTED
001114  000                     $ITEMB: .BYTE   0       ;;CONTAINS ITEM CONTROL BYTE
001115  001                     $ERMAX: .BYTE   1       ;;CONTAINS MAX. ERRORS PER TEST
001116  000000                  $ERRPC: .WORD   0       ;;CONTAINS PC OF LAST ERROR INSTRUCTION
001120  000000                  $GDADR: .WORD   0       ;;CONTAINS ADDRESS OF 'GOOD' DATA
001122  000000                  $BDADR: .WORD   0       ;;CONTAINS ADDRESS OF 'BAD' DATA
001124  000000                  $GDDAT: .WORD   0       ;;CONTAINS 'GOOD' DATA
001126  000000                  $BDDAT: .WORD   0       ;;CONTAINS 'BAD' DATA
001130  000000                          .WORD   0       ;;RESERVED--NOT TO BE USED
001132  000000                          .WORD   0
001134  000                     $AUTOB: .BYTE   0       ;;AUTOMATIC MODE INDICATOR
001135  000                     $INTAG: .BYTE   0       ;;INTERRUPT MODE INDICATOR
001136  000000                          .WORD   0
001140  177570                  SWR:    .WORD   DSWR    ;;ADDRESS OF SWITCH REGISTER
001142  177570                  DISPLAY: .WORD  DDISP   ;;ADDRESS OF DISPLAY REGISTER
001144  177560                  $TKS:   177560          ;;TTY KBD STATUS
001146  177562                  $TKB:   177562          ;;TTY KBD BUFFER
001150  177564                  $TPS:   177564          ;;TTY PRINTER STATUS REG. ADDRESS
001152  177566                  $TPB:   177566          ;;TTY PRINTER BUFFER REG. ADDRESS
001154  000                     $NULL:  .BYTE   0       ;;CONTAINS NULL CHARACTER FOR FILLS
001155  002                     $FILLS: .BYTE   2       ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
001156  012                     $FILLC: .BYTE   12      ;;INSERT FILL CHARS. AFTER A "LINE FEED"
001157  000                     $TPFLG: .BYTE   0       ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
001160  000000                  $TMP0:  .WORD   0       ;;USER DEFINED
001162  000000                  $TMP1:  .WORD   0       ;;USER DEFINED
001164  000000                  $TMP2:  .WORD   0       ;;USER DEFINED
001166  000000                  $TMP3:  .WORD   0       ;;USER DEFINED
001170  000000                  $TMP4:  .WORD   0       ;;USER DEFINED
001172  000000                  $TMP5:  .WORD   0       ;;USER DEFINED
001174  000000                  $TMP6:  .WORD   0       ;;USER DEFINED
001176  000000                  $TMP7:  .WORD   0       ;;USER DEFINED
001200  000000                  $TIMES: 0               ;;MAX. NUMBER OF ITERATIONS
001202  000000                  $ESCAPE:0               ;;ESCAPE ON ERROR ADDRESS
001204  177607  000377        $BELL:  .ASCIZ  <207><377><377> ;;CODE FOR BELL
001210  077                     $QUES:  .ASCII  /?/     ;;QUESTION MARK
001211  015                     $CRLF:  .ASCII  <15>    ;;CARRIAGE RETURN
001212  000012                  $LF:    .ASCIZ  <12>    ;;LINE FEED
                                ;;**********************************************************************
                                .SBTTL  APT MAILBOX-ETABLE

                                ;;**********************************************************************
                                .EVEN
```

```
1173  001214                        $MAIL:                          ;;APT MAILBOX
1174  001214  000000               $MSGTY: .WORD   AMSGTY          ;;MESS E TYPE CODE
1175  001216  000000               $FATAL: .WORD   AFATAL          ;;FATA RROR NUMBER
1176  001220  000000               $TESTN: .WORD   ATESTN          ;;TEST  JMBER
1177  001222  000000               $PASS:  .WORD   APASS           ;;PASS COUNT
1178  001224  000000               $DEVCT: .WORD   ADEVCT          ;;DEVICE COUNT
1179  001226  000000               $UNIT:  .WORD   AUNIT           ;;I/O UNIT NUMBER
1180  001230  000000               $MSGAD: .WORD   AMSGAD          ;;MESSAGE ADDRESS
1181  001232  000000               $MSGLG: .WORD   AMSGLG          ;;MESSAGE LENGTH
1182  001234                        $ETABLE:                        ;;APT ENVIRONMENT TABLE
1183  001234     000                $ENV:   .BYTE   AENV            ;;ENVIRONMENT BYTE
1184  001235     000                $ENVM:  .BYTE   AENVM           ;;ENVIRONMENT MODE BITS
1185  001236  000000               $SWREG: .WORD   ASWREG          ;;APT SWITCH REGISTER
1186  001240  000000               $USWR:  .WORD   AUSWR           ;;USER SWITCHES
1187  001242  000000               $CPUOP: .WORD   ACPUOP          ;;CPU TYPE,OPTIONS
1188                                ;*                              BITS 15-11=CPU TYPE
1189                                ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
1190                                ;*                                      11/70=06,PDQ=07,Q=10
1191                                ;*                              BIT 10=REAL TIME CLOCK
1192                                ;*                              BIT  9=FLOATING POINT PROCESSOR
1193                                ;*                              BIT  8=MEMORY MANAGEMENT
1194  001244     000                $MAMS1: .BYTE   AMAMS1          ;;HIGH ADDRESS,M.S. BYTE
1195  001245     000                $MTYP1: .BYTE   AMTYP1          ;;MEM. TYPE,BLK#1
1196                                ;*                              MEM.TYPE BYTE   --  (HIGH BYTE)
1197                                ;*                                      900 NSEC CORE=001
1198                                ;*                                      300 NSEC BIPOLAR=002
1199                                ;*                                      500 NSEC MOS=003
1200  001246  000000               $MADR1: .WORD   AMADR1          ;;HIGH ADDRESS,BLK#1
1201                                ;*                                MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF "TYPE" ABOVE
1202  001250     000                $MAMS2: .BYTE   AMAMS2          ;;HIGH ADDRESS,M.S. BYTE
1203  001251     000                $MTYP2: .BYTE   AMTYP2          ;;MEM.TYPE,BLK#2
1204  001252  000000               $MADR2: .WORD   AMADR2          ;;MEM.LAST ADDRESS,BLK#2
1205  001254     000                $MAMS3: .BYTE   A MS3           ;;HIGH ADDRESS,M.S.BYTE
1206  001255     000                $MTYP3: .BYTE   AMTYP3          ;;MEM.TYPE,BLK#3
1207  001256  000000               $MADR3: .WORD   AMADR3          ;;MEM.LAST ADDRESS,BLK#3
1208  001260     000                $MAMS4: .BYTE   AMAMS4          ;;HIGH ADDRESS,M.S.BYTE
1209  001261     000                $MTYP4: .BYTE   AMTYP4          ;;MEM.TYPE,BLK#4
1210  001262  000000               $MADR4: .WORD   AMADR4          ;;MEM.LAST ADDRESS,BLK#4
1211  001264  120210               $VECT1: .WORD   AVECT1          ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
1212  001266  000000               $VECT2: .WORD   AVECT2          ;;INTERRUPT VECTOR#2BUS PRIORITY#2
1213  001270  177440               $BASE:  .WORD   ABASE           ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
1214  001272  000000               $DEVM:  .WORD   ADEVM           ;;DEVICE MAP
1215  001274  000000               $CDW1:  .WORD   ACDW1           ;;CONTROLLER DESCRIPTION WORD#1
1216  001276  000000               $CDW2:  .WORD   ACDW2           ;;CONTROLLER DESCRIPTION WORD#2
1217  001300                        $ETEND:
1218                                .MEXIT
```

```
                                        .SBTTL  ERROR POINTER TABLE

                                        ;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
                                        ;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
                                        ;*LOCATION $ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
                                        ;*NOTE1:        IF $ITEMB IS 0 THE ONLY PERTINENT DATA IS ($ERRPC).
                                        ;*NOTE2:        EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

                                        ;*      EM              ;;POINTS TO THE ERROR MESSAGE
                                        ;*      DH              ;;POINTS TO THE DATA HEADER
                                        ;*      DT              ;;POINTS TO THE DATA
                                        ;*      DF              ;;POINTS TO THE DATA FORMAT


001300                                  $ERRTB:
                                        ;       ERROR 1: ATTEMPTING TO SET CMD BIT DRIVE MESS A
001300  000000                          EM1N:   0
001302  000000                                  0
001304  046520                                  DT001
001306  047154                                  DF001
                                        ;       ERROR 2: ATTEMPTING A SELECT OF DRIVE NUM - CS1 INCORRECT
001310  052410                                  EM106
001312  057141                                  EM2003
001314  046540                                  DT002
001316  047210                                  DF002
                                        ;       ERROR 3: ATTEMPTING A SELECT  OF DRIVE NUM - DRIVE NUM INCORRECT
001320  052410                                  EM106
001322  057204                                  EM2004
001324  046540                                  DT002
001326  047210                                  DF002
                                        ;       ERROR 4: ATTEMPTING A SELECT OF DRIVE NUM - MESSAGE A INCORRECT
001330  052410                                  EM106
001332  057063                                  EM2001
001334  046540                                  DT002
001336  047210                                  DF002
                                        ;       ERROR 5: ATTEMPTING A SELECT OF DRIVE NUM - MESSAGE B INCORRECT
001340  052410                                  EM106
001342  057112                                  EM2002
001344  046540                                  DT002
001346  047210                                  DF002
                                        ;       ERROR 6: ATTEMPTING A SELECT WITH HEAD ADD - CS1 INCORRECT
001350  052514                                  EM107
001352  057141                                  EM2003
001354  046562                                  DT006
001356  047244                                  DF006
                                        ;       ERROR 7: ATTEMPTING A SELECT WITH HEAD ADD - HEAD INCORRECT
001360  052514                                  EM107
001362  057255                                  EM2005
001364  046562                                  DT006
001366  047244                                  DF006
                                        ;       ERROR 10: ATTEMPTING A SELECT WITH HEAD ADD - MESSAGE A INCORRECT
001370  052514                                  EM107
001372  057063                                  EM2001
001374  046562                                  DT006
001376  047244                                  DF006
                                        ;       ERROR 11: ATTEMPTING A SELECT WITH HEAD ADD - MESSAGE B INCORRECT
```

```
1275  001400  052514                        EM107
1276  001402  057112                        EM2002
1277  001404  046562                        DT006
1278  001406  047244                        DF006
1279                          ;       ERROR 12: ATTEMPTING A SELECT WITH MESS SELECT BITS - CS1 INCORRECT
1280  001410  052611                        EM108
1281  001412  057141                        EM2003
1282  001414  046604                        DT012
1283  001416  047300                        DF012
1284                          ;       ERROR 13: ATTEMPTING A SELECT WITH MESS SELECT BITS - MR1 INCORRECT
1285  001420  052611                        EM108
1286  001422  057322                        EM2006
1287  001424  046604                        DT012
1288  001426  047300                        DF012
1289                          ;       ERROR 14: ATTEMPTING A SELECT WITH MESS SELECT BITS - MESS SELECT CODE INCORRECT
1290  001430  052611                        EM108
1291  001432  057351                        EM2007
1292  001434  046604                        DT012
1293  001436  047300                        DF012
1294                          ;       ERROR 15: ATTEMPTING A SELECT WITH MESS SELECT BITS - MESS A INCORRECT
1295  001440  052611                        EM108
1296  001442  057063                        EM2001
1297  001444  046604                        DT012
1298  001446  047300                        DF012
1299                          ;       ERROR 16: ATTEMPTING A SELECT WITH MESS SELECT BITS - MESS B INCORRECT
1300  001450  052611                        EM108
1301  001452  057112                        EM2002
1302  001454  046604                        DT012
1303  001456  047300                        DF012
1304                          ;       ERROR 17: ATTEMPTING A SEEK TO AN RK06 - CS1 INCORRECT
1305  001460  052711                        EM109
1306  001462  057141                        EM2003
1307  001464  046632                        DT017
1308  001466  047334                        DF017
1309                          ;       ERROR 20: ATTEMPTING A SEEK TO AN RK06 - SEEK BIT IN MESS A NOT SET
1310  001470  052711                        EM109
1311  001472  057005                        EM2000
1312  001474  046632                        DT017
1313  001476  047334                        DF017
1314                          ;       ERROR 21: ATTEMPTING A SEEK TO AN RK06 - CYLINDER ADD INCORRECT IN MESS B
1315  001500  052711                        EM109
1316  001502  057421                        EM2008
1317  001504  046632                        DT017
1318  001506  047334                        DF017
1319                          ;       ERROR 22: ATTEMPTING A SEEK TO AN RK06 - MESSAGE A INCORRECT
1320  001510  052711                        EM109
1321  001512  057063                        EM2001
1322  001514  046632                        DT017
1323  001516  047334                        DF017
1324                          ;       ERROR 23: ATTEMPTING A SEEK TO AN RK06 - MESSAGE B INCORRECT
1325  001520  052711                        EM109
1326  001522  057112                        EM2002
1327  001524  046632                        DT017
1328  001526  046632                        DT017
1329                          ;       ERROR 24: ATTEMPTING A SEEK WITH CDT SET - CS1 INCORRECT
1330  001530  052764                        EM110
```

```
1331  001532  057141                    EM2003
1332  001534  046632                    DT017
1333  001536  047334                    DF017
1334                          ;         ERROR 25: ATTEMPTING A SEEK TO AN RK607 - SEEK BIT IN MESS A NOT SET
1335  001540  052764                    EM110
1336  001542  057005                    EM2000
1337  001544  046632                    DT017
1338  001546  047334                    DF017
1339                          ;         ERROR 26: ATTEMPTING A SEEK WITH CDT SET
1340                          ;         CYLINDER ADD INCORRECT IN MESS B
1341  001550  052764                    EM110
1342  001552  057421                    EM2008
1343  001554  046632                    DT017
1344  001556  047334                    DF017
1345                          ;         ERROR 27: ATTEMPTING A SEEK WITH CDT SET - MESSAGE A INCORRECT
1346  001560  052764                    EM110
1347  001562  057063                    EM2001
1348  001564  046632                    DT017
1349  001566  047334                    DF017
1350                          ;         ERROR 30: ATTEMPTING A SEEK WITH CDT SET - MESSAGE B INCORRECT
1351  001570  052764                    EM110
1352  001572  057112                    EM2002
1353  001574  046632                    DT017
1354  001576  047334                    DF017
1355                          ;         ERROR 31: ATTEMPTING OFFSET - CS1 INCORRECT
1356  001600  053041                    EM111
1357  001602  057141                    EM2003
1358  001604  046654                    DT031
1359  001606  047370                    DF031
1360                          ;         ERROR 32: ATTEMPTING OFFSET - OFFSET BITS INCORRECT
1361  001610  053041                    EM111
1362  001612  057472                    EM2009
1363  001614  046654                    DT031
1364  001616  047370                    DF031
1365                          ;         ERROR 33: ATTEMPTING OFFSET - MESS A INCORRECT
1366  001620  053041                    EM111
1367  001622  057063                    EM2001
1368  001624  046654                    DT031
1369  001626  047370                    DF031
1370                          ;         ERROR 34: ATTEMPTING OFFSET - MESS B INCORRECT
1371  001630  053041                    EM111
1372  001632  057063                    EM2001
1373  001634  046654                    DT031
1374  001636  047370                    DF031
1375                          ;         ERROR 35: ATTEMPTING COMMAND WITH NON-ZERO CYL ADD AND OFFSET-
1376                          ;           CS1 INCORRECT
1377  001640  053104                    EM112
1378  001642  057141                    EM2003
1379  001644  046676                    DT035
1380  001646  047424                    DF035
1381                          ;         ERROR 36: ATTEMPTING COMMAND WITH NON-ZERO CYL ADD AND OFFSET-
1382                          ;           DRIVE COMMAND BIT NOT SET IN MESS A
1383  001650  053104                    EM112
1384  001652  057005                    EM2000
1385  001654  046676                    DT035
1386  001656  047424                    DF035
```

```
1387                                    ;       ERROR 37: ATTEMPTING COMMAND WITH NON-ZERO CYL ADD AND OFFSET-
1388                                    ;                 CYLINDER ADDRESS BITS INCORRECT IN MESS B
1389  001660  053104                            EM112
1390  001662  057421                            EM2008
1391  001664  046676                            DT035
1392  001666  047424                            DF035
1393                                    ;       ERROR 40: ATTEMPTING COMMAND WITH NON-ZERO CYL ADD AND OFFSET-
1394                                    ;                 MESS A INCORRECT
1395  001670  053104                            EM112
1396  001672  057063                            EM2001
1397  001674  046676                            DT035
1398  001676  047424                            DF035
1399                                    ;       ERROR 41: ATTEMPTING COMMAND WITH NON-ZERO CYL ADD AND OFFSET-
1400                                    ;                 MESS B INCORRECT
1401  001700  053104                            EM112
1402  001702  057112                            EM2002
1403  001704  046676                            DT035
1404  001706  047424                            DF035
1405                                    ;       ERROR 42: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1406                                    ;                 CS1 INCORRECT
1407  001710  053237                            EM113
1408  001712  057141                            EM2003
1409  001714  046604                            DT012
1410  001716  047300                            DF012
1411                                    ;       ERROR 43: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1412                                    ;                 MAINT REG 1 INCORRECT
1413  001720  053237                            EM113
1414  001722  057322                            EM2006
1415  001724  046604                            DT012
1416  001726  047300                            DF012
1417                                    ;       ERROR 44: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1418                                    ;                 DRIVE COMMAND BIT INCORRECT
1419  001730  053237                            EM113
1420  001732  057005                            EM2000
1421  001734  046604                            DT012
1422  001736  046604                            DT012
1423                                    ;       ERROR 45: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1424                                    ;                 MESSAGE SELECT SELECT CODE IN MESSAGE B INCORRECT
1425  001740  053237                            EM113
1426  001742  057351                            EM2007
1427  001744  046604                            DT012
1428  001746  047300                            DF012
1429                                    ;       ERROR 46: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1430                                    ;                 MESS A INCORRECT
1431  001750  053237                            EM113
1432  001752  057063                            EM2001
1433  001754  046604                            DT012
1434  001756  047300                            DF012
1435                                    ;       ERROR 47: ATTEMPTING COMMAND WITH NON-ZERO MSG SELECT
1436                                    ;                 MESS B INCORRECT
1437  001760  053237                            EM113
1438  001762  057112                            EM2002
1439  001764  046604                            DT012
1440  001766  047300                            DF012
1441                                    ;       ERROR 50:       ATTEMPTING TO SHIFT DRIVE MESSAGE
1442                                    ;                       SHIFT REG A INCORRECT
```

```
1443  001770  053351                  EM114
1444  001772  057063                  EM2001
1445  001774  046722                  DT050
1446  001776  047460                  DF050
1447                          ;       ERROR 51:       ATTEMPTING TO SHIFT DRIVE MESSAGE
1448                          ;                       SHIFT REG B INCORRECT
1449  002000  053351                  EM114
1450  002002  057112                  EM2002
1451  002004  046722                  DT050
1452  002006  047460                  DF050
1453                          ;       ERROR 52:       ATTEMPTING TO GENERATE ODD PARITY ON SELECT DRIVE MESSAGE
1454                          ;                       PARITY ON MESSAGE A INCORRECT
1455  002010  053414                  EM115
1456  002012  057543                  EM2010
1457  002014  046744                  DT052
1458  002016  047514                  DF052
1459                          ;       ERROR 53:       ATTEMPTING TO GENERATE ODD PARITY ON SELECT DRIVE MESSAGE
1460                          ;                       PARITY ON MESSAGE B INCORRECT
1461  002020  053414                  EM115
1462  002022  057605                  EM2011
1463  002024  046744                  DT052
1464  002026  047514                  DF052
1465                          ;       ERROR 54:       ATTEMPTING TO GENERATE ODD PARITY ON SELECT DRIVE MESSAGE
1466                          ;                       MESSAGE A INCORRECT
1467  002030  053414                  EM115
1468  002032  057063                  EM2001
1469  002034  046744                  DT052
1470  002036  047514                  DF052
1471                          ;       ERROR 55:       ATTEMPTING TO GENERATE ODD PARITY ON SELECT DRIVE MESSAGE
1472                          ;                       MESSAGE B INCORRECT
1473  002040  053414                  EM115
1474  002042  057112                  EM2002
1475  002044  046744                  DT052
1476  002046  047514                  DF052
1477                          ;       ERROR 56:       ATTEMPTING TO GENERATE EVEN PARITY ON SELECT DRIVE MESSAGE
1478                          ;                       PARITY ON MESSAGE A INCORRECT
1479  002050  053506                  EM116
1480  002052  057543                  EM2010
1481  002054  046744                  DT052
1482  002056  047514                  DF052
1483                          ;       ERROR 57:       ATTEMPTING TO GENERATE EVEN PARITY ON SELECT DRIVE MESSAGE
1484                          ;                       PARITY ON MESSAGE IS INCORRECT
1485  002060  053506                  EM116
1486  002062  057605                  EM2011
1487  002064  046744                  DT052
1488  002066  047514                  DF052
1489                          ;       ERROR 60:       ATTEMPTING TO GENERATE EVEN PARITY ON SELECT DRIVE MESSAGE
1490                          ;                       MESSAGE A INCORRECT
1491  002070  053506                  EM116
1492  002072  057063                  EM2001
1493  002074  046744                  DT052
1494  002076  047514                  DF052
1495                          ;       ERROR 61:       ATTEMPTING TO GENERATE EVEN PARITY ON SELECT DRIVE MESSAGE
1496                          ;                       MESSAGE B INCORRECT
1497  002100  053506                  EM116
1498  002102  057112                  EM2002
```

```
1499  002104  046744                    DT052
1500  002106  047514                    DF052
1501                          ;         ERROR 62:       ATTEMPTING COMPLETE EXECUTION OF DESELECT DRIVE IN
1502                          ;                         MAINT MODE - COMMAND AND STATUS REG 1 INCORRECT.
1503  002110  053601                    EM117
1504  002112  057141                    EM2003
1505  002114  046760                    DT062
1506  002116  047540                    DF062
1507                          ;         ERROR 63:       ATTEMPTING COMPLETE EXECUTION OF DESELECT DRIVE IN
1508                          ;                         MAINT MODE - COMMAND AND STATUS REG 2 INCORRECT.
1509  002120  053601                    EM117
1510  002122  057647                    EM2012
1511  002124  046760                    DT062
1512  002126  047540                    DF062
1513                          ;         ERROR 64:       ATTEMPTING COMPELTE EXECUTION OF DESELECT DRIVE IN
1514                          ;                         MAINT MODE - ERROR REG. INCORRECT.
1515  002130  053601                    EM117
1516  002132  057712                    EM2013
1517  002134  046760                    DT062
1518  002136  047540                    DF062
1519                          ;         ERROR 65:       ATTEMPTING COMPLETE EXECUTION OF SE.ECT DRIVE IN
1520                          ;                         MAINT MODE - COMMAND AND STATUS REGISTER 1 INCORRECT AT
1521                          ;                         PHASE ADDRESS 4
1522  002140  053716                    EM118
1523  002142  057736                    EM2014
1524  002144  047004                    DT065
1525  002146  047564                    DF065
1526                          ;         ERROR 66:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1527                          ;                         MAINT MODE - COMMAND AND STATUS REG 1 INVALID DURING
1528                          ;                         COMMAND EXECUTION.
1529  002150  053716                    EM118
1530  002152  060024                    EM2015
1531  002154  047004                    DT065
1532  002156  047564                    DF065
1533                          ;         ERROR 67:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1534                          ;                         MAINT MODE - MAINTENANCE REG 2 UNEXPECTEDLY CHANGED
1535                          ;                         DURING COMMAND EXECUTION.
1536  002160  053716                    EM118
1537  002162  060116                    EM2016
1538  002164  047014                    DT067
1539  002166  047610                    DF067
1540                          ;         ERROR 70:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1541                          ;                         MAINT MODE - MAINTENANCE REG 3 UNEXPECTEDLY CHANGED
1542                          ;                         DURING COMMAND EXECUTION.
1543  002170  053716                    EM118
1544  002172  060216                    EM2017
1545  002174  047014                    DT067
1546  002176  047610                    DF067
1547                          ;         ERROR 71:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1548                          ;                         MAINT MODE - COMMAND AND STATUS REG 1 INCORRECT.
1549  002200  053716                    EM118
1550  002202  057141                    EM2003
1551  002204  046760                    DT062
1552  002206  047540                    DF062
1553                          ;         ERROR 72:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1554                          ;                         MAINT MODE - COMMAND AND STATUS REG. 2 INCORRECT.
```

```
1555  002210  053716                    EM118
1556  002212  057647                    EM2012
1557  002214  046760                    DT062
1558  002216  047540                    DF062
1559                          ;         ERROR 73:       ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE IN
1560                          ;                         MAINT MODE - ERROR REGISTER INCORRECT.
1561  002220  053716                    EM118
1562  002222  057712                    EM2013
1563  002224  046760                    DT062
1564  002226  047540                    DF062
1565                          ;         ERROR 74:       ATTEMPTING EXECUTION OF DESELECT DRIVE AT NORMAL
1566                          ;                         SPEED - COMMAND AND STATUS REG. 1 INCORRECT.
1567  002230  054031                    EM119
1568  002232  057063                    EM2001
1569  002234  046760                    DT062
1570  002236  047540                    DF062
1571                          ;         ERROR 75:       ATTEMPTING EXECUTION OF DESELECT DRIVE AT NORMAL
1572                          ;                         SPEED - COMMAND AND STATUS REG. 2 INCORRECT.
1573  002240  054031                    EM119
1574  002242  057647                    EM2012
1575  002244  046760                    DT062
1576  002246  047540                    DF062
1577                          ;         ERROR 76:       ATTEMPTING EXECUTION OF DESELECT DRIVE AT NORMAL
1578                          ;                         SPEED - ERROR REG INCORRECT.
1579  002250  054031                    EM119
1580  002252  057712                    EM2013
1581  002254  046760                    DT062
1582  002256  047540                    DF062
1583                          ;         ERROR 77:       ATTEMPTING TO WRITE CS1 IN MAINT MODE - CS1 INCORRECT
1584  002260  054120                    EM120
1585  002262  057141                    EM2003
1586  002264  047004                    DT065
1587  002266  047564                    DF065
1588                          ;         ERROR 100:      ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET
1589                          ;                         INTERRUPT DID NOT OCCUR.
1590  002270  054214                    EM121
1591  002272  060316                    EM2018
1592  002274  047030                    DT100
1593  002276  047634                    DF100
1594                          ;         ERROR 101:      ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET
1595                          ;                         CS1 INCORRECT AFTER INTERRUPT.
1596  002300  054214                    EM121
1597  002302  060346                    EM2019
1598  002304  046760                    DT062
1599  002306  047540                    DF062
1600                          ;         ERROR 102:      ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET
1601                          ;                         CS2 INCORRECT AFTER INTERRUPT.
1602  002310  054214                    EM121
1603  002312  060431                    EM2020
1604  002314  046760                    DT062
1605  002316  047540                    DF062
1606                          ;         ERROR 103:      ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET
1607                          ;                         ERROR REGISTER IN CORRECT AFTER INTERRUPT
1608  002320  054214                    EM121
1609  002322  060514                    EM2021
1610  002324  046760                    DT062
```

```
1611  002326  047540                    DF062
1612                          ;         ERROR 104:      ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET
1613                          ;                         INTERRUPT DID NOT CLEAR IN RK611
1614  002330  054214                    EM121
1615  002332  060565                    EM2022
1616  002334  047030                    DT100
1617  002336  047634                    DF100
1618                          ;         ERROR 105:      ATTEMPTING DESELECT COMMAND AFTER WRITING SILO
1619                          ;                         TO CHECK GO CLEAR-CS2 INCORRECT
1620
1621  002340  054315                    EM122
1622  002342  057647                    EM2012
1623  002344  046760                    DT062
1624  002346  047540                    DF062
1625                          ;         ERROR 106:      ATTEMPTING DESELECT COMMAND AFTER WRITING SILO
1626                          ;                         TO CHECK GO CLEAR-DATA LATE DID NOT OCCUR WHEN
1627                          ;                         READING SILO
1628  002350  054315                    EM122
1629  002352  060626                    EM2023
1630  002354  046760                    DT062
1631  002356  047540                    DF062
1632                          ;         ERROR 107:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1633                          ;                         COMMAND AND STATUS REG 1 INCORRECT AT PHASE ADDRESS 4
1634  002360  054416                    EM123
1635  002362  057736                    EM2014
1636  002364  047004                    DT065
1637  002366  047564                    DF065
1638                          ;         ERROR 110:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1639                          ;                         COMMAND AND STATUS REG 1 INVALID DURING COMMAND EXECUTION
1640  002370  054416                    EM123
1641  002372  060024                    EM2015
1642  002374  047004                    DT065
1643  002376  047564                    DF065
1644
1645                          ;         ERROR 111:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1646                          ;                         MAINTENANCE REG 2 UNEXPECTEDLY CHANGED DURING COMMAND EXECUTION
1647  002400  054416                    EM123
1648  002402  060116                    EM2016
1649  002404  047014                    DT067
1650  002406  047610                    DF067
1651                          ;         ERROR 112:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1652                          ;                         MAINTENANCE REG 3 UNEXPECTEDLY CHANGED DURING COMMAND EXECUTION
1653  002410  054416                    EM123
1654  002412  060216                    EM2017
1655  002414  047014                    DT067
1656  002416  047610                    DF067
1657                          ;         ERROR 113:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1658                          ;                         COMMAND AND STATUS REG. 1 INCORRECT
1659  002420  054416                    EM123
1660  002422  057141                    EM2003
1661  002424  046760                    DT062
1662  002426  047540                    DF062
1663                          ;         ERROR 114:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1664                          ;                         COMMAND AND STATUS REG. 2 INCORRECT
1665  002430  054416                    EM123
1666  002432  057647                    EM2012
```

```
1667  002434  046760                    DI062
1668  002436  047540                    DF062
1669                          ;         ERROR 115:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1670                          ;                         ERROR REGISTER INCORRECT
1671  002440  054416                    EM123
1672  002442  057712                    EM2013
1673  002444  046760                    DT062
1674  002446  047540                    DF062
1675                          ;         ERROR 116:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1676                          ;                         COMMAND AND STATUS REG. 1 INCORRECT
1677  002450  054502                    EM124
1678  002452  057141                    EM2003
1679  002454  046520                    DT001
1680  002456  047154                    DF001
1681                          ;         ERROR 117:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1682                          ;                         DRIVE SELECT CODE IN MESSAGE INCORRECT
1683  002460  054502                    EM124
1684  002462  057204                    EM2004
1685  002464  046520                    DT001
1686  002466  047154                    DF001
1687                          ;         ERROR 120:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1688                          ;                         DRIVE COMMAND BITS IN MESSAGE INCORRECT
1689  002470  054502                    EM124
1690  002472  060700                    EM2024
1691  002474  046520                    DT001
1692  002476  047154                    DF001
1693                          ;         ERROR 121:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1694                          ;                         HEAD ADD CODE IN MESSAGE A INCORRECT
1695  002500  054502                    EM124
1696  002502  057255                    EM2005
1697  002504  046520                    DT001
1698  002506  047154                    DF001
1699                          ;         ERROR 122:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1700                          ;                         PARITY BIT IN MESSAGE INCORRECT
1701  002510  054502                    EM124
1702  002512  057543                    EM2010
1703  002514  046520                    DT001
1704  002516  047154                    DF001
1705                          ;         ERROR 123:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1706                          ;                         MESS SELECT CODE IN MESSAGE IN CORRECT
1707  002520  054502                    EM124
1708  002522  057351                    EM2007
1709  002524  046520                    DT001
1710  002526  047154                    DF001
1711                          ;         ERROR 124:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1712                          ;                         CYLINDER AND BITS IN MESSAGE IS INCORRECT
1713  002530  054502                    EM124
1714  002532  057421                    EM2008
1715  002534  046520                    DT001
1716  002536  047154                    DF001
1717                          ;         ERROR 125:      ATTEMPTING SELECT DRIVE IN MAINT MODE
1718                          ;                         PARITY BIT IN MESSAGE IS INCORRECT
1719  002540  054502                    EM124
1720  002542  057605                    EM2011
1721  002544  046520                    DT001
1722  002546  047154                    DF001
```

```
1723                                    ;       ERROR 126:      ATTEMPTING COMPLETE EXECUTION OF DESELECT DRIVE IN
1724                                    ;                       MAINT MODE - DRIVE STATUS REG INCORRECT
1725  002550  053601                            EM117
1726  002552  060750                            EM2025
1727  002554  046760                            DT062
1728  002556  047540                            DF062
1729                                    ;       ERROR 127:      ATTEMPTING EXECUTION OF SELECT DRIVE IN
1730                                    ;                       MAINT MODE - DRIVE STATUS REG INCORRECT
1731  002560  053716                            EM118
1732  002562  060750                            EM2025
1733  002564  046760                            DT062
1734  002566  047540                            DF062
1735                                    ;       ERROR 130:      ATTEMPTING EXECUTION OF DESELECT DRIVE AT NORMAL
1736                                    ;                       SPEED - DRIVE STATUS REG INCORRECT
1737  002570  054031                            EM119
1738  002572  060750                            EM2025
1739  002574  046760                            DT062
1740  002576  047540                            DF062
1741                                    ;       ERROR 131:      ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
1742                                    ;                       DRIVE STATUS REG INCORRECT
1743  002600  054416                            EM123
1744  002602  060750                            EM2025
1745  002604  046760                            DT062
1746  002606  047540                            DF062
1747                                    ;       ERROR 132:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WRL
1748                                    ;                   CONTROLLER READY DID NOT SET
1749  002610  054550                            EM125
1750  002612  061010                            EM2026
1751  002614  047030                            DT100
1752  002616  047634                            DF100
1753                                    ;       ERROR 133:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WRL
1754                                    ;                   LOAD STATUS DID NOT LOAD DRIVE STATUS REF
1755  002620  054550                            EM125
1756  002622  061045                            EM2027
1757  002624  046760                            DT062
1758  002626  047540                            DF062
1759                                    ;       ERROR 134:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WRL
1760                                    ;                   CS1 INCORRECT
1761  002630  054550                            EM125
1762  002632  057141                            EM2003
1763  002634  046760                            DT062
1764  002636  047540                            DF062
1765                                    ;       ERROR 135:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WRL
1766                                    ;                   CS2 INCORRECT
1767  002640  054550                            EM125
1768  002642  057647                            EM2012
1769  002644  046760                            DT062
1770  002646  047540                            DF062
1771                                    ;       ERROR 136:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WL
1772                                    ;                   ERROR REG. INCORRECT
1773  002650  054550                            EM125
1774  002652  057712                            EM2013
1775  002654  046760                            DT062
1776  002656  047540                            DF062
1777                                    ;       ERROR 137:  ATTEMPTING TO FORCE DRA,SPDLSS,VV,OFST,DRDY,WL
1778                                    ;                   DRIVE STATUS REG. INCORRECT
```

```
1779  002660  054550                    EM125
1780  002662  060750                    EM2025
1781  002664  046760                    DT062
1782  002666  047540                    DF062
1783                          ;         ERROR 140:  ATTEMPTING TO FORCE DRIVE AVAILIABLE
1784                          ;                     CS1 INCORRECT
1785  002670  054767                    EM126
1786  002672  057141                    EM2003
1787  002674  046760                    DT062
1788  002676  047540                    DF062
1789                          ;         ERROR 141:  ATTEMPTING TO FORCE DRIVE AVAILABLE
1790                          ;                     CS2 INCORRECT
1791  002700  054767                    EM126
1792  002702  057647                    EM2012
1793  002704  046760                    DT062
1794  002706  047540                    DF062
1795                          ;         ERROR 142:  ATTEMPTING TO FORCE DRIVE AVAILIABLE
1796                          ;                     DRIVE STATUS REC INCORRECT
1797  002710  054767                    EM126
1798  002712  060750                    EM2025
1799  002714  046760                    DT062
1800  002716  047540                    DF062
1801                          ;         ERROR 143:  ATTEMPTING TO FORCE DRIVE AVAIVIABLE
1802                          ;                     ERROR REGISTER INCORRECT
1803  002720  054767                    EM126
1804  002722  057712                    EM2013
1805  002724  046760                    DT062
1806  002726  047540                    DF062
1807                          ;         ERROR 144:  ATTEMPTING TO FORCE DRIVE BUS PARITY ERROR DETECTED BY RK611
1808                          ;                     CS1 INCORRECT
1809  002730  055034                    EM127
1810  002732  057141                    EM2003
1811  002734  046760                    DT062
1812  002736  047540                    DF062
1813                          ;         ERROR 145:  ATTEMPTING TO FORCE DRIVE BUS PARITY ERROR DETECTED BY RK611
1814                          ;                     CS2 INCORRECT
1815  002740  055034                    EM127
1816  002742  057647                    EM2012
1817  002744  046760                    DT062
1818  002746  047540                    DF062
1819                          ;         ERROR 146:  ATTEMPTING TO FORCE DRIVE BUS PARITY ERROR DETECTED BY RK611
1820                          ;                     DRIVE STATUS REG INCORRECT
1821  002750  055034                    EM127
1822  002752  060750                    EM2025
1823  002754  046760                    DT062
1824  002756  047540                    DF062
1825                          ;         ERROR 147:  ATTEMPTING TO FORCE DRIVE BUS PARITY ERROR DETECTED BY RK611
1826                          ;                     ERROR REC INCORRRECT
1827  002760  055034                    EM127
1828  002762  057712                    EM2013
1829  002764  046760                    DT062
1830  002766  047540                    DF062
1831                          ;         ERROR 150:  ATTEMPTING TO FORCE DRIVE AVAILIABLE RESET ERROR
1832                          ;                     CS1 INCORRECT
1833  002770  055132                    EM128
1834  002772  057141                    EM2003
```

```
1835  002774  046760                    DT062
1836  002776  047540                    DF062
1837                          ;         ERROR 151:  ATTEMPTING TO FORCE DRIVE AVAILABLE RESET ERROR
1838                          ;                     CS2 INCORRECT
1839  003000  055132                    EM128
1840  003002  057647                    EM2012
1841  003004  046760                    DT062
1842  003006  047540                    DF062
1843                          ;         ERROR 152:  ATTEMPTING TO FORCE DRIVE AVAILABLE RESET ERROR
1844                          ;                     DRIVE STATUS REG. INCORRECT
1845  003010  055132                    EM128
1846  003012  060750                    EM2025
1847  003014  046760                    DT062
1848  003016  047540                    DF062
1849                          ;         ERROR 153:  ATTEMPTING TO FORCE DRIVE AVAILABLE RESET ERROR
1850                          ;                     ERROR REG. INCORRECT
1851  003020  055132                    EM128
1852  003022  057712                    EM2013
1853  003024  046760                    DT062
1854  003026  047540                    DF062
1855                          ;         ERROR 154:  TESTING CDT SET DRIVE TYPE DETECTION
1856                          ;                     CS1 INCORRECT
1857  003030  055213                    EM129
1858  003032  057141                    EM2003
1859  003034  046760                    DT062
1860  003036  047540                    DF062
1861                          ;         ERROR 155:  TESTING CDT SET DRIVE TYPE DETECTION
1862                          ;                     CS2 INCORRECT
1863  003040  055213                    EM129
1864  003042  057647                    EM2012
1865  003044  046760                    DT062
1866  003046  047540                    DF062
1867                          ;         ERROR 156:  TESTING CDT SET DRIVE TYPE DETECTION
1868                          ;                     DRIVE STATUS REG INCORRECT
1869  003050  055213                    EM129
1870  003052  060750                    EM2025
1871  003054  046760                    DT062
1872  003056  047540                    DF062
1873                          ;         ERROR 157:  TESTING CDT SET DRIVE TYPE DETECTION
1874                          ;                     ERROR REG INCORRECT
1875  003060  055213                    EM129
1876  003062  057712                    EM2013
1877  003064  046760                    DT062
1878  003066  047540                    DF062
1879                          ;         ERROR 160:  ATTEMPTING TO FORCE DRIVE TYPE ERROR ADDRESSING RK06
1880                          ;                     CS1 INCORRECT
1881  003070  055260                    EM130
1882  003072  057141                    EM2003
1883  003074  046760                    DT062
1884  003076  047540                    DF062
1885                          ;         ERROR 161:  ATTEMPTING TO FORCE DRIVE TYPE ERROR ADDRESSING RK06
1886                          ;                     CS2 INCORRECT
1887  003100  055260                    EM130
1888  003102  057647                    EM2012
1889  003104  046760                    DT062
1890  003106  047540                    DF062
```

```
1891                                    ;       ERROR 162:  ATTEMPTING TO FORCE DRIVE TYPE ERROR ADDRESSING RK06
1892                                    ;                   DRIVE STATUS REG INCORRECT
1893  003110  055260                            EM130
1894  003112  060750                            EM2025
1895  003114  046760                            DT062
1896  003116  047540                            DF062
1897                                    ;       ERROR 163:  ATTEMPTING TO FORCE DRIVE TYPE ERROR ADDRESSING RK06
1898                                    ;                   ERROR REG INCORRECT
1899  003120  055260                            EM130
1900  003122  057712                            EM2013
1901  003124  046760                            DT062
1902  003126  047540                            DF062
1903                                    ;       ERROR 164:  ATTEMPTING TO FORCE DRIVE TYPE ERROR WITH CTD SET
1904                                    ;                   CS1 INCORRECT
1905  003130  055342                            EM131
1906  003132  057141                            EM2003
1907  003134  046760                            DT062
1908  003136  047540                            DF062
1909                                    ;       ERROR 165:  ATTEMPTING TO FORCE DRIVE TYPE ERROR WITH CDT SET
1910                                    ;                   CS2 INCORRECT
1911  003140  055342                            EM131
1912  003142  057647                            EM2012
1913  003144  046760                            DT062
1914  003146  047540                            DF062
1915                                    ;       ERROR 166:  ATTEMPTING TO FORCE DRIVE TYPE ERROR WITH CDT SET
1916                                    ;                   DRIVE STATUS REG INCORRECT
1917  003150  055342                            EM131
1918  003152  060750                            EM2025
1919  003154  046760                            DT062
1920  003156  047540                            DF062
1921                                    ;       ERROR 167:  ATTEMPTING TO FORCE DRIVE TYPE ERROR WITH CDT SET
1922                                    ;                   ERROR REG INCORRECT
1923  003160  055342                            EM131
1924  003162  057712                            EM2013
1925  003164  046760                            DT062
1926  003166  047540                            DF062
1927                                    ;       ERROR 170:  ATTEMPTING TO FORCE SPEED LOSS
1928                                    ;                   CS1 INCORRECT
1929  003170  055427                            EM132
1930  003172  057141                            EM2003
1931  003174  046760                            DT062
1932  003176  047540                            DF062
1933                                    ;       ERROR 171:  ATTEMPTING TO FORCE SPEED LOSS
1934                                    ;                   CS2 INCORRECT
1935  003200  055427                            EM132
1936  003202  057647                            EM2012
1937  003204  046760                            DT062
1938  003206  047540                            DF062
1939                                    ;       ERROR 172:  ATTEMPTING TO FORCE SPEED LOSS
1940                                    ;                   DRIVE STATUS REG INCORRECT
1941  003210  055427                            EM132
1942  003212  060750                            EM2025
1943  003214  046760                            DT062
1944  003216  047540                            DF062
1945                                    ;       ERROR 173:  ATTEMPTING TO FORCE SPEED LOSS
1946                                    ;                   ERROR REG. INCORRECT
```

```
1947  003220  055427                          EM132
1948  003222  057712                          EM2013
1949  003224  046760                          DT062
1950  003226  047540                          DF062
1951                          ;               ERROR 174:  ATTEMPTING TO FORCE DRIVE OFF TRACK
1952                          ;                           CS1 INCORRECT
1953  003230  055466                          EM133
1954  003232  057141                          EM2003
1955  003234  046760                          DT062
1956  003236  047540                          DF062
1957                          ;               ERROR 175:  ATTEMPTING TO FORCE DRIVE OFF TRACK
1958                          ;                           CS2 INCORRECT
1959  003240  055466                          EM133
1960  003242  057647                          EM2012
1961  003244  046760                          DT062
1962  003246  047540                          DF062
1963                          ;               ERROR 176:  ATTEMPTING TO FORCE DRIVE OFF TRACK
1964                          ;                           DRIVE STATUS REG INCORRECT
1965  003250  055466                          EM133
1966  003252  060750                          EM2025
1967  003254  046760                          DT062
1968  003256  047540                          DF062
1969                          ;               ERROR 177:  ATTEMPTING TO FORCE DRIVE OFF TRACK
1970                          ;                           ERROR REG INCORRECT
1971  003260  055466                          EM133
1972  003262  057712                          EM2013
1973  003264  046760                          DT062
1974  003266  047540                          DF062
1975                          ;               ERROR 200:  ATTEMPTING TO FORCE WRITE LOCK ERROR
1976                          ;                           CS1 INCORRECT
1977  003270  055532                          EM134
1978  003272  057141                          EM2003
1979  003274  046760                          DT062
1980  003276  047540                          DF062
1981                          ;               ERROR 201:  ATTEMPTING TO FORCE WRITE LOCK ERROR
1982                          ;                           CS2 INCORRECT
1983  003300  055532                          EM134
1984  003302  057647                          EM2012
1985  003304  046760                          DT062
1986  003306  047540                          DF062
1987                          ;               ERROR 202:  ATTEMPTING TO FORCE WRITE LOCK ERROR
1988                          ;                           DRIVE STATUS REG INCORRECT
1989  003310  055532                          EM134
1990  003312  060750                          EM2025
1991  003314  046760                          DT062
1992  003316  047540                          DF062
1993                          ;               ERROR 203:  ATTEMPTING TO FORCE WRITE LOCK ERROR
1994                          ;                           ERROR REG INCORRECT
1995  003320  055532                          EM134
1996  003322  057712                          EM2013
1997  003324  046760                          DT062
1998  003326  047540                          DF062
1999                          ;               ERROR 204:  ATTEMPTING TO FORCE SEEK INCOMPLETE
2000                          ;                           CS1 INCORRECT
2001  003330  055577                          EM135
2002  003332  057141                          EM2003
```

```
 2003  003334  046760                    DT062
 2004  003336  047540                    DF062
 2005                          ;         ERROR 205:  ATTEMPTING TO FORCE SEEK INCOMPLETE
 2006                          ;                     CS2 INCORRECT
 2007  003340  055577                    EM135
 2008  003342  057647                    EM2012
 2009  003344  046760                    DT062
 2010  003346  047540                    DF062
 2011                          ;         ERROR 206:  ATTEMPTING TO FORCE SEEK INCOMPLETE
 2012                          ;                     DRIVE STATUS REG INCORRECT
 2013  003350  055577                    EM135
 2014  003352  060750                    EM2025
 2015  003354  046760                    DT062
 2016  003356  047540                    DF062
 2017                          ;         ERROR 207:  ATTEMPTING TO FORCE SEEK INCOMPLETE
 2018                          ;                     ERROR REG INCORRECT
 2019  003360  055577                    EM135
 2020  003362  057712                    EM2013
 2021  003364  046760                    DT062
 2022  003366  047540                    DF062
 2023                          ;         ERROR 210:  ATTEMPTING TO FORCE NON-EXECUTABLE FUNCTION
 2024                          ;                     CS1 INCORRECT
 2025  003370  055643                    EM136
 2026  003372  057141                    EM2003
 2027  003374  046760                    DT062
 2028  003376  047540                    DF062
 2029                          ;         ERROR 211:  ATTEMPTING TO FORCE NON-EXECUTABLE FUNCTION
 2030                          ;                     CS2 INCORRECT
 2031  003400  055643                    EM136
 2032  003402  057647                    EM2012
 2033  003404  046760                    DT062
 2034  003406  047540                    DF062
 2035                          ;         ERROR 212:  ATTEMPTING TO FORCE NON-EXECUTABLE FUNCTION
 2036                          ;                     DRIVE STATUS REG INCORRECT
 2037  003410  055643                    EM136
 2038  003412  060750                    EM2025
 2039  003414  046760                    DT062
 2040  003416  047540                    DF062
 2041                          ;         ERROR 213:  ATTEMPTING TO FROCE NON-EXECUTABLE FUNCTION
 2042                          ;                     ERROR REG INCORRECT
 2043  003420  055643                    EM136
 2044  003422  057712                    EM2013
 2045  003424  046760                    DT062
 2046  003426  047540                    DF062
 2047                          ;         ERROR 214:  ATTEMPTING TO FORCE AC  JW AND C-D PARITY ERROR
 2048                          ;                     CS1 INCORRECT
 2049  003430  055717                    EM137
 2050  003432  057141                    EM2003
 2051  003434  046760                    DT062
 2052  003436  047540                    DF062
 2053                          ;         ERROR 215:  ATTEMPTING TO FORCE AC LOW AND C-D PARITY ERROR
 2054                          ;                     CS2 INCORRECT
 2055  003440  055717                    EM137
 2056  003442  057647                    EM2012
 2057  003444  046760                    DT062
 2058  003446  047540                    DF062
```

```
2059                                    ;       ERROR 216:  ATTEMPTING TO FORCE AC LOW AND C-D PARITY ERROR
2060                                    ;                   DRIVE STATUS REG INCORRECT
2061  003450  055717                           EM137
2062  003452  060750                           EM2025
2063  003454  046760                           DT062
2064  003456  047540                           DF062
2065                                    ;       ERROR 217:  ATTEMPTING TO FORCE AC LOW AND C-D PARITY ERROR
2066                                    ;                   ERROR REG INCORRECT
2067  003460  055717                           EM137
2068  003462  057712                           EM2013
2069  003464  046760                           DT062
2070  003466  047540                           DF062
2071                                    ;       ERROR 220:  ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2072                                    ;                   CS1 INCORRECT
2073  003470  055777                           EM138
2074  003472  057141                           EM2003
2075  003474  046760                           DT062
2076  003476  047540                           DF062
2077                                    ;       ERROR 221:  ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2078                                    ;                   CS2 INCORRECT
2079  003500  055777                           EM138
2080  003502  057647                           EM2012
2081  003504  046760                           DT062
2082  003506  047540                           DF062
2083                                    ;       ERROR 222:  ATTEMPTING TO FORCE ILLEGAL DISK ADDRRESS ERROR (DRIVE)
2084                                    ;                   DRIVE STATUS REG INCORRECT
2085  003510  055777                           EM138
2086  003512  060750                           EM2025
2087  003514  046760                           DT062
2088  003516  047540                           DF062
2089                                    ;       ERROR 223:  ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2090                                    ;                   ERROR REG INCORRECT
2091  003520  055777                           EM138
2092  003522  057712                           EM2013
2093  003524  046760                           DT062
2094  003526  047540                           DF062
2095                                    ;       ERROR 224:  ATTERMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2096                                    ;                   CS1 INCORRECT
2097  003530  056107                           EM139
2098  003532  057141                           EM2003
2099  003534  047044                           DT224
2100  003536  047674                           DF224
2101                                    ;       ERROR 225:  ATTERMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2102                                    ;                   CS2 INCORRECT
2103  003540  056107                           EM139
2104  003542  057647                           EM2012
2105  003544  047044                           DT224
2106  003546  047674                           DF224
2107                                    ;       ERROR 226:  ATTERMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2108                                    ;                   DRIVE STATUS REG INCORRECT
2109  003550  056107                           EM139
2110  003552  060750                           EM2025
2111  003554  047044                           DT224
2112  003556  047674                           DF224
2113                                    ;       ERROR 227:  ATTERMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2114                                    ;                   ERROR REG INCORRECT
```

```
2115  003560  056107                    EM139
2116  003562  057712                    EM2013
2117  003564  047044                    DT224
2118  003566  047674                    DF224
2119                          ;         ERROR 230: TESTING ILLEGAL DISK ADDRESS ERROR LOGIC IN RK611
2120                          ;                    CSI INCORRECT
2121  003570  056171                    EM140
2122  003572  057141                    EM2003
2123  003574  047100                    DT230
2124  003576  047730                    DF230
2125                          ;         ERROR 231: TESTING ILLEGAL DISK ADDRESS ERROR LOGIC IN RK611
2126                          ;                    CS2 INCORRECT
2127  003600  056171                    EM140
2128  003602  057647                    EM2012
2129  003604  047100                    DT230
2130  003606  047730                    DF230
2131                          ;         ERROR 232: TESTING ILLEGAL DISK ADDRESS ERROR LOGIC IN RK611
2132                          ;                    DRIVE STATUS REG INCORRECT
2133  003610  056171                    EM140
2134  003612  060750                    EM2025
2135  003614  047100                    DT230
2136  003616  047730                    DF230
2137                          ;         ERROR 233: TESTING ILLEGAL DISK ADDRESS ERROR LOGIC IN RK611
2138                          ;                    ERROR REGISTER
2139  003620  056171                    EM140
2140  003622  057712                    EM2013
2141  003624  047100                    DT230
2142  003626  047730                    DF230
2143                          ;         ERROR 234: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES
2144                          ;                    CS1 INCORRECT
2145  003630  056253                    EM141
2146  003632  057141                    EM2003
2147  003634  046760                    DT062
2148  003636  047540                    DF062
2149                          ;         ERROR 235: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES
2150                          ;                    CS2 INCORRECT
2151  003640  056253                    EM141
2152  003642  057647                    EM2012
2153  003644  046760                    DT062
2154  003646  047540                    DF062
2155                          ;         ERROR 236: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES
2156                          ;                    DRIVE STATUS REG. INCORRECT
2157  003650  056253                    EM141
2158  003652  060750                    EM2025
2159  003654  046760                    DT062
2160  003656  047540                    DF062
2161                          ;         ERROR 237: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES
2162                          ;                    ERROR REG. INCORRECT
2163  003660  056253                    EM141
2164  003662  057712                    EM2013
2165  003664  046760                    DT062
2166  003666  047540                    DF062
2167                          ;         ERROR 240: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES WITH
2168                          ;                    BAD PARITY - CSI INCORRECT
2169  003670  056327                    EM142
2170  003672  057141                    EM2003
```

```
2171  003674  046760                    DI062
2172  003676  047540                    DF062
2173                          ;         ERROR 241: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES WITH
2174                          ;                    BAD PARITY - CS2 INCORRECT
2175  003700  056327                    EM142
2176  003702  057647                    EM2012
2177  003704  046760                    DT062
2178  003706  047540                    DF062
2179                          ;         ERROR 242: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES WITH
2180                          ;                    BAD PARITY - DRIVE STATUS REG. INCORRECT
2181  003710  056327                    EM142
2182  003712  060750                    EM2025
2183  003714  046760                    DT062
2184  003716  047540                    DF062
2185                          ;         ERROR 243: ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES WITH
2186                          ;                    BAD PARITY - ERROR ERROR INCOMPLETE
2187  003720  056327                    EM142
2188  003722  057712                    EM2013
2189  003724  046760                    DT062
2190  003726  047540                    DF062
2191                          ;         ERROR 244: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (DRIVE BUS TIMEOUT)
2192                          ;                    CS1 INCORRECT
2193  003730  056425                    EM143
2194  003732  057141                    EM2003
2195  003734  046760                    DT062
2196  003736  047540                    DF062
2197                          ;         ERROR 245: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (DRIVE BUS TIMEOUT)
2198                          ;                    CS2 INCORRECT
2199  003740  056425                    EM143
2200  003742  057647                    EM2012
2201  003744  046760                    DT062
2202  003746  047540                    DF062
2203                          ;         ERROR 246: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (DRIVE BUS TIMEOUT)
2204                          ;                    DRIVE STATUS REG INCORRECT
2205  003750  056425                    EM143
2206  003752  060750                    EM2025
2207  003754  046760                    DT062
2208  003756  047540                    DF062
2209                          ;         ERROR 247: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (DRIVE BUS TIMEOUT)
2210                          ;                    ERROR REG INCORRECT
2211  003760  056425                    EM143
2212  003762  057712                    EM2013
2213  003764  046760                    DT062
2214  003766  047540                    DF062
2215                          ;         ERROR 250: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (NO SACK)
2216                          ;                    CS1 INCORRECT
2217  003770  056520                    EM144
2218  003772  057141                    EM2003
2219  003774  046760                    DT062
2220  003776  047540                    DF062
2221                          ;         ERROR 251: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (NO SACK)
2222                          ;                    CS2 INCORRECT
2223  004000  056520                    EM144
2224  004002  057647                    EM2012
2225  004004  046760                    DT062
2226  004006  047540                    DF062
```

```
2227                                    ;       ERROR 252: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (NO SACK)
2228                                    ;                  DRIVE STATUS REG INCORRECT
2229  004010  056520                           EM144
2230  004012  060750                           EM2025
2231  004014  046760                           DT062
2232  004016  047540                           DF062
2233                                    ;       ERROR 253: ATTEMPTING TO FORCE NON-EXISTENT DRIVE (NO SACK)
2234                                    ;                  ERROR REG INCORRECT
2235  004020  056520                           EM144
2236  004022  057712                           EM2013
2237  004024  046760                           DT062
2238  004026  047540                           DF062
2239                                    ;       ERROR 254: ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE RESET
2240                                    ;                  UNEXPECTED INTERRUPT OCCURRED
2241  004030  056601                           EM145
2242  004032  061120                           EM2028
2243  004034  047030                           DT100
2244  004036  047634                           DF100
2245                                    ;       ERROR 255: ATTEMPTING EXECUTION FO DESELECT DRIVE WITH IE RESET
2246                                    ;                  INTERRUPT OCCURRED WHEN INTERRUPT ENABLE SET
2247  004040  056601                           EM145
2248  004042  061156                           EM2029
2249  004044  047030                           DT100
2250  004046  047634                           DF100
2251                                    ;       ERROR 256:      ATTEMPTING TO EXECUTE AN ILLEGAL FUNCTION
2252                                    ;                       CS1 INCORRECT
2253  004050  056666                           EM146
2254  004052  057141                           EM2003
2255  004054  047132                           DT256
2256  004056  047764                           DF256
2257                                    ;       ERROR 257:      ATTEMPTING TO EXECUTE AN ILLEGAL FUNCTION
2258                                    ;                       ERROR REG INCORRECT
2259  004060  056666                           EM146
2260  004062  057712                           EM2013
2261  004064  047132                           DT256
2262  004066  047764                           DF256
2263                                    ;       ERROR 260:      ATTEMPTING TO CLEAR ILLEGAL FUNCTION - CS1 INCORRECT
2264  004070  056740                           EM147
2265  004072  057141                           EM2003
2266  004074  047132                           DT256
2267  004076  047764                           DF256
2268                                    ;       ERROR 261:      ATTEMPTING TO CLEAR ILLEGAL FUNCTION - ERROR REG INCORRECT
2269  004100  056740                           EM147
2270  004102  057712                           EM2013
2271  004104  047132                           DT256
2272  004106  047764                           DF256
2273                                    ;       ERROR 262:  UNEXPECTED MEMORY PARITY ERROR TRAP
2274  004110  051766                           EM000
2275  004112  050240                           DH000C
2276  004114  046514                           DT000
2277  004116  047150                           DF000
```

```
2278                                    .SBTTL  TEMPORARY STORAGE FOR RK611 CONTROLLER REGISTER
2279
2280  004120  000000                    T.CS1:  .WORD   0               ;CONTROL AND STATUS REGISTER 1
2281  004122  000000                    T.WC:   .WORD   0               ;WORD COUNT REGISTER
2282  004124  000000                    T.BA:   .WORD   0               ;BUS ADDRESS REGISTER
2283  004126  000000                    T.DA:   .WORD   0               ;DESIRED TRACK SECTOR  EGISTER
2284  004130  000000                    T.CS2:  .WORD   0               ;CONTROL AND STATUS R  ISTER 2
2285  004132  000000                    T.DS:   .WORD   0               ;DRIVE STATUS REGISTER
2286  004134  000000                    T.ER:   .WORD   0               ;ERROR REGISTER
2287  004136  000000                    T.ASOF: .WORD   0               ;ATTENTION SUMMARY AND OFFSET REGISTER
2288  004140  000000                    T.DCYL: .WORD   0               ;DESIRED CYLINDER REGISTER
2289  004142  000000                    T.DB:   .WORD   0               ;DATA BUFFER
2290  004144  000000                    T.MR1:  .WORD   0               ;MAINTENANCE REGISTER 1
2291  004146  000000                    T.MR2:  .WORD   0               ;MAINTENANCE REGISTER 2
2292  004150  000000                    T.MR3:  .WORD   0               ;MAINTENANCE REGISTER 3
2293  004152  000000                    T.ECPS: .WORD   0               ;ECC POSITION INFORMATION
2294  004154  000000                    T.ECPT: .WORD   0               ;ECC PATTERN INFORMATION
2295  004156  000000                    T.SPAR: .WORD   0               ;SPARE REGISTER
2296
2297                                    .SBTTL  EXPECTED RK611 CONTROLLER REGISTERS
2298
2299  004160  000000                    E.CS1:  .WORD   0               ;CONTROL AND STATUS REGISTER 1
2300  004162  000000                    E.WC:   .WORD   0               ;WORD COUNT REGISTER
2301  004164  000000                    E.BA:   .WORD   0               ;BUS ADDRESS REGISTER
2302  004166  000000                    E.DA:   .WORD   0               ;DESIRED TRACK SECTOR REGISTER
2303  004170  000000                    E.CS2:  .WORD   0               ;CONTROL AND STATUS REGISTER 2
2304  004172  000000                    E.DS:   .WORD   0               ;DRIVE STATUS REGISTER
2305  004174  000000                    E.ER:   .WORD   0               ;ERROR REGISTER
2306  004176  000000                    E.ASOF: .WORD   0               ;ATTENTION SUMMARY AND OFFSET REGISTER
2307  004200  000000                    E.DCYL: .WORD   0               ;DESIRED CYLINDER REGISTER
2308  004202  000000                    E.DB:   .WORD   0               ;DATA BUFFER
2309  004204  000000                    E.MR1:  .WORD   0               ;MAINTENANCE REGISTER 1
2310  004206  000000                    E.MR2:  .WORD   0               ;MAINTENANCE REGISTER 2
2311  004210  000000                    E.MR3:  .WORD   0               ;MAINTENANCE REGISTER 3
2312  004212  000000                    E.ECPS: .WORD   0               ;ECC POSITION INFORMATION
2313  004214  000000                    E.ECPT: .WORD   0               ;ECC PATTERN INFORMATION
2314  004216  000000                    E.SPAR: .WORD   0               ;SPARE REGISTER
2315
2316                                    .SBTTL  PREVIOUS RK611 CONTROLLER REGISTERS
2317
2318  004220  000000                    P.CS1:  .WORD   0               ;PREVIOUS COMMAND AND STATUS REG 1
2319  004222  000000                    P.CS2:  .WORD   0               ;PREVIOUS COMMAND AND STATUS REG 2
2320  004224  000000                    P.DS:   .WORD   0               ;PREVIOUS DRIVE STATUS REG
2321  004226  000000                    P.ER:   .WORD   0               ;PREVIOUS ERROR REG
2322  004230  000000                    U.MR2:  .WORD   0               ;UNSHIFTED MAINTENANCE REG 2
2323  004232  000000                    U.MR3:  .WORD   0               ;UNSHIFTED MAINTENANCE REG 3
```

```
2324                                    .SBTTL  PROGRAM DEFINED VARIABLES
2325
2326   004234  000210                   RKVEC:  .WORD   210             ;RK611 VECTOR
2327   004236  000240                   RKPRI:  .WORD   PR5             ;RK611 PRIORITY
2328   004240  000000                   SRTFLG: .WORD   0               ;START FLAG
2329                                                                    ;   0 = 200
2330                                                                    ;   1 = 214
2331                                                                    ;  -1 = 204
2332   004242  000000                   ERRCNT: .WORD   0               ;ERROR COUNT FOR SWITCH 12 ABORT
2333   004244  000000                   DRVCOD: .WORD   0               ;DRIVE SELECT CODE
2334   004246  000000                   MSGCOD: .WORD   0               ;MESSAGE SELECT CODE
2335   004250  000000                   HDCODE: .WORD   0               ;HEAD SELECT CODE
2336   004252  000000                   CYLIN:  .WORD   0               ;CYLINDER ADD VALUE
2337   004254  000000                   OFFVAL: .WORD   0               ;OFFSET VALUE
2338   004256  000000                   SFTCNT: .WORD   0               ;SHIFT COUNT FOR DRIVE MESSAGE SHIFTING
2339   004260  000000                   PARBIT: .WORD   0               ;PARITY BIT FOR SHIFT
2340   004262  000015                   WAITIM: .WORD   15              ;WAITING FOR DESELECT COMMAND
2341   004264  000144                   STALL:  .WORD   100.            ;STALL TIME FOR MESSAGE TIME OUT (NED)
2342   004266  000000                   DRVTYP: .WORD   0               ;DRIVE TYPE INDICATOR
2343   004270  000000                   ILLFUN: .WORD   0               ;ILLEGAL FUNCTION CODE
2344   004272  000000                   TRAPPC: .WORD   0               ;ADDRESS OF TRAP FROM MEMORY CHECK
2345   004274  000000                   SAVSWR: .WORD   0               ;SAVED SWITCH REG FOR POWER FAIL
```

```
                                        .SBTTL  PROGRAM SETUP
                                
004276  012737  000001  004240  PARM:   MOV     #1,SRTFLG       ;LOAD START FLAG FOR PARMETER START
004304  000406                  BR      START1
                                
004306  012737  177777  004240  RESTRT: MOV     #-1,SRTFLG      ;LOAD START FLAG FOR RESTART
004314  000402                  BR      START1
                                
004316  005037  004240          START:  CLR     SRTFLG          ;CLEAR START FLAG
004322  000005                  START1: RESET                   ;RESET THE WHOLE SYSTEM
004324  012706  001100                  MOV     #STACK,SP       ;INITIALIZE STACK POINTER
004330  012746  000340                  MOV     #PR7,-(SP)      ;LOAD STACK TO LOCK OUT ALL INTERRUPTS
004334  012746  004342                  MOV     #1$,-(SP)       ;LOAD START OF PROGRAM
004340  000002                  RTI                             ;LOAD PSW
                                
004342                          1$:
                                .SBTTL  INITIALIZE THE COMMON TAGS
                                ;;CLEAR THE COMMON TAGS ($CMTAG) AREA
004342  012706  001100                  MOV     #$CMTAG,R6      ;;FIRST LOCATION TO BE CLEARED
004346  005026                  CLR     (R6)+                   ;;CLEAR MEMORY LOCATION
004350  022706  001140                  CMP     #SWR,R6 ;;DONE?
004354  001374                  BNE     .-6                     ;;LOOP BACK IF NO
004356  012706  001100                  MOV     #STACK,SP       ;;SETUP THE STACK POINTER
                                ;;INITIALIZE A FEW VECTORS
004362  012737  042512  000020          MOV     #$SCOPE,@#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
004370  012737  000340  000022          MOV     #340,@#IOTVEC+2 ;;LEVEL 7
004376  012737  043516  000030          MOV     #$ERROR,@#EMIVEC ;;EMI VECTOR FOR ERROR ROUTINE
004404  012737  000340  000032          MOV     #340,@#EMTVEC+2 ;;LEVEL 7
004412  012737  046424  000034          MOV     #$TRAP,@#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
004420  012737  000340  000036          MOV     #340,@#TRAPVEC+2;LEVEL 7
004426  012737  046272  000024          MOV     #$PWRDN,@#PWRVEC ;;POWER FAILURE VECTOR
004434  012737  000340  000026          MOV     #340,@#PWRVEC+2 ;;LEVEL 7
004442  013737  042204  042176          MOV     $ENDCT,$EOPCT   ;;SETUP END-OF-PROGRAM COUNTER
004450  005037  001200                  CLR     $TIMES          ;;INITIALIZE NUMBER OF ITERATIONS
004454  005037  001202                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE ON ERROR ADDRESS
004460  112737  000001  001115          MOVB    #1,$ERMAX       ;;ALLOW ONE ERROR PER TEST
004466  012737  004466  001106          MOV     #.,$LPADR       ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
004474  012737  004474  001110          MOV     #.,$LPERR       ;;SETUP THE ERROR LOOP ADDRESS
                                ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
                                ;;EQUAL TO A "-1"  SETUP FOR A SOFTWARE SWITCH REGISTER.
004502  013746  000004                  MOV     @#ERRVEC,-(SP)  ;;SAVE ERROR VECTOR
004506  012737  004542  000004          MOV     #64$,@#ERRVEC   ;;SET UP ERROR VECTOR
004514  012737  177570  001140          MOV     #DSWR,SWR       ;;SETUP FOR A HARDWARE SWICH REGISTER
004522  012737  177570  001142          MOV     #DDISP,DISPLAY  ;;AND A HARDWARE DISPLAY REGISTER
004530  022777  177777  174402          CMP     #-1,@SWR        ;;TRY TO REFERENCE HARDWARE SWR
004536  001012                  BNE     66$                     ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
                                                                ;;AND  THE HARDWARE SWR IS NOT = -1
004540  000403                  BR      65$                     ;;BRANCH IF NO TIMEOUT
004542  012716  004550  64$:    MOV     #65$,(SP)               ;;SET UP FOR TRAP RETURN
004546  000002                  RTI
004550  012737  000176  001140  65$:    MOV     #SWREG,SWR      ;;POINT TO SOFTWARE SWR
004556  012737  000174  001142          MOV     #DISPREG,DISPLAY
004564  012637  000004          66$:    MOV     (SP)+,@#ERRVEC  ;;RESTORE ERROR VECTOR
                                
004570  005037  001222                  CLR     $PASS           ;;CLEAR PASS COUNT
004574  132737  000200  001235          BITB    #APTSIZE,$ENVM  ;;TEST USER SIZE UNDER APT
```

```
2402  004602  001403                        BEQ     67$                ;;YES,USE NON-APT SWITCH
2403  004604  012737  001236  001140        MOV     #$SWREG,SWR        ;;NO,USE APT SWITCH REGISTER
2404  004612                          67$:
2405  004612  005037  004242                CLR     ERRCNT             ;CLEAR ERROR COUNT FOR SWITCH 12 ABORT
2406                                  .SBTTL  TYPE PROGRAM NAME
2407                                  ;;TYPE THE NAME OF THE PROGRAM IF FIRST PASS
2408  004616  005227  177777                INC     #-1                ;;FIRST TIME?
2409  004622  001055                        BNE     68$                ;;BRANCH IF NO
2410  004624  022737  042340  000042        CMP     #$ENDAD,@#42       ;;ACT-11?
2411  004632  001451                        BEQ     68$                ;;BRANCH IF YES
2412  004634  104401  004702                TYPE    ,69$               ;;TYPE ASCIZ STRING
2413                                  .SBTTL  GET VALUE FOR SOFTWARE SWITCH REGISTER
2414  004640  005737  000042                TST     @#42               ;;ARE WE RUNNING UNDER XXDP/ACT?
2415  004644  001012                        BNE     70$                ;;BRANCH IF YES
2416  004646  123727  001234  000001        CMPB    $ENV,#1            ;;ARE WE RUNNING UNDER APT?
2417  004654  001406                        BEQ     70$                ;;BRANCH IF YES
2418  004656  023727  001140  000176        CMP     SWR,#SWREG         ;;SOFTWARE SWITCH REG SELECTED?
2419  004664  001005                        BNE     71$                ;;BRANCH IF NO
2420  004666  104406                        GTSWR                      ;;GET SOFT-SWR SETTINGS
2421  004670  000403                        BR      71$
2422  004672  112737  000001  001134  70$:  MOVB    #1,$AUTOB          ;;SET AUTO-MODE INDICATOR
2423  004700                          71$:
2424  004700  000426                        BR      68$                ;;GET OVER THE ASCIZ
2425                                  ;;69$:  .ASCIZ  <CRLF>/Rk611 DISKLESS DIAGNOSTIC: PART 2 CZR68C0/<CRLF>
2426  004756                          68$:
2427  004756  022737  000001  004240        CMP     #1,SRTFLG          ;CHECK IF PARAMETER START
2428  004764  001122                        BNE     15$                ;NO, CONTINUE SETUP
2429  004766  104401  050010          5$:   TYPE    ,OPR001            ;TYPE "RK611 BUS ADDRESS (   ) ="
2430  004772  013746  001270                MOV     $BASE,-(SP)        ;;SAVE $BASE FOR TYPEOUT
2431  004776  104402                        TYPOC                      ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
2432  005000  104401  050037                TYPE    ,OPR002
2433  005004  104412                        RDOCT                      ;GET VALUE
2434  005006  012637  001160                MOV     (SP)+,$TMP0
2435  005012  001407                        BEQ     7$                 ;CHECK IF <CR>
2436  005014  022737  160000  001160        CMP     #160000,$TMP0      ;CHECK IF IN I/O PAGE
2437  005022  101361                        BHI     5$
2438  005024  013737  001160  001270        MOV     $TMP0,$BASE        ;LOAD NEW BUS ADDRESS
2439  005032  104401  050045          7$:   TYPE    ,OPR003            ;TYPE "RK611 VECTOR ADDRESS (   ) ="
2440  005036  013746  001264                MOV     $VECT1,-(SP)
2441  005042  042716  160000                BIC     #160000,(SP)
2442  005046  104402                        TYPOC
2443  005050  104401  050037                TYPE    ,OPR002
2444  005054  104412                        RDOCT                      ;GET VALUE
2445  005056  012637  001160                MOV     (SP)+,$TMP0
2446  005062  001412                        BEQ     10$                ;CHECK IF <CR>
2447  005064  022737  001000  001160        CMP     #1000,$TMP0        ;CHECK IF LEGAL
2448  005072  101757                        BLOS    7$
2449  005074  042737  017777  001264        BIC     #17777,$VECT1      ;LOAD NEW VECTOR ADDRESS
2450  005102  053737  001160  001264        BIS     $TMP0,$VECT1
2451  005110  104401  050075          10$:  TYPE    ,OPR004            ;TYPE "RK611 PRIORITY (   ) ="
2452  005114  005046                        CLR     -(SP)              ;MAKE ROOM ON THE STACK
2453  005116  113716  001265                MOVB    $VECT1+1,(SP)
2454  005122  006216                        ASR     (SP)               ;SHIFT 5 BITS RIGHT
2455  005124  006216                        ASR     (SP)
2456  005126  006216                        ASR     (SP)
2457  005130  006216                        ASR     (SP)
```

```
 2458  005132  006216                          ASR     (SP)
 2459  005134  104402                          TYPOC
 2460  005136  104401  050037                  TYPE    ,OPR002
 2461  005142  104412                          RDOCT                   ;GET VALUE
 2462  005144  012637  001160                  MOV     (SP)+,$TMP0
 2463  005150  001430                          BEQ     15$             ;CHECK FOR DEFAULT
 2464  005152  022737  000007  001160          CMP     #7,$TMP0        ;CHECK IF LEGAL
 2465  005160  103753                          BLO     10$
 2466  005162  022737  000004  001160          CMP     #4,$TMP0
 2467  005170  101347                          BHI     10$
 2468  005172  006337  001160                  ASL     $TMP0           ;SHIFT 5 BITS LEFT
 2469  005176  006337  001160                  ASL     $TMP0
 2470  005202  006337  001160                  ASL     $TMP0
 2471  005206  006337  001160                  ASL     $TMP0
 2472  005212  006337  001160                  ASL     $TMP0
 2473  005216  042737  160000  001264          BIC     #160000,$VECT1  ;STORE NEW PRIORITY
 2474  005224  153737  001160  001265          BISB    $TMP0,$VECT1+1
 2475  005232  013737  001264  004234  15$:    MOV     $VECT1,RKVEC    ;STORE RK611 VECTOR
 2476  005240  042737  160000  004234          BIC     #160000,RKVEC
 2477  005246  113737  001265  004236          MOVB    $VECT1+1,RKPRI  ;STORE RK611 PRIORITY
 2478
 2479  005254  004737  042360          NEWPAS: JSR     PC,CHKPAR       ;CHECK FOR MEMORY CHECK ENABLE
 2480  005260  012746  000340                  MOV     #PR7,-(SP)      ;LOCK OUT INTERRUPTS
 2481  005264  012746  005272                  MOV     #TST1,-(SP)
 2482  005270  000002                          RTI
```

```
                                .SBTTL  **DRIVE MESSAGE LOADING
                                ;;*****************************************************************
                                ;*TEST 1         FIRST COMMAND IN MAINT MODE
                                ;*
                                ;*       INITIALIZE RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
                                ;*       MODE. ISSUE SELECT DRIVE. WAIT AND MAKE SURE CS1 REMAINS
                                ;*       THE SAME. CLOCK IN MESSAGES A AND B. MAKE SURE
                                ;*       CORRECT MSG ARE LOADED. CHECKING IS DONE A FIELD AT A
                                ;*       TIME.
                                ;*
                                ;;*****************************************************************
005272  000004                TST1:   SCOPE
005274  012737  000144  001200        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
005302  013702  001270        MOV     $BASE,R2        ;LOAD RK611 BASE
005306  012762  100000  000000        MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
005314  012762  000040  000026        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN DIAGNOSTIC MODE
005322  012762  000001  000000        MOV     #SELDRV,RKCS1(R2)       ;LOAD CS1 WITH SELECT DRIVE
005330  012700  000015                MOV     #15,R0          ;WAIT FOR READY TO SET
005334  005300                1$:     DEC     R0
005336  001376                        BNE     1$
005340  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
005346  012737  000001  004160        MOV     #SELDRV,E.CS1   ;LOAD EXPECT CS1
005354  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK IF CS1 CHANGED
005362  001402                        BEQ     2$              ;NO, CONTINUE
005364  104077                        ERROR   77              ;CS1 INCORRECT
005366  000503                        BR      TST2            ;;GO ON TO NEXT TEST
                                
005370  012700  000016        2$:     MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
005374  012762  000440  000026  3$:     MOV     #DMD!MCLK,RKMR1(R2)
005402  012762  000040  000026        MOV     #DMD,RKMR1(R2)
005410  005300                        DEC     R0
005412  001370                        BNE     3$
005414  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
005422  016237  000034  004146        MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
005430  016237  000036  004150        MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
005436  012737  000001  004160        MOV     #SELDRV,E.CS1   ;LOAD EXPECTED CS1
005444  005037  004206                CLR     E.MR2           ;LOAD EXPECTED MAINT REG. 2
005450  005037  004210                CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
005454  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG. 1 CORRECT
005462  001402                        BEQ     4$              ;YES, CHECK MESSAGES A & B
005464  104116                        ERROR   116             ;CS1 INCORRECT
005466  000443                        BR      TST2            ;;GO ON TO NEXT TEST
                                
005470  032737  000017  004146  4$:   BIT     #17,T.MR2       ;CHECK IF DRIVE SELECT BITS ZERO
005476  001401                        BEQ     5$              ;YES, CONTINUE
005500  104117                        ERROR   117             ;MESSAGE SELECT BITS NOT ZERO
005502  032737  007760  004146  5$:   BIT     #7760,T.MR2     ;CHECK IF COMMAND BITS ZERO
005510  001401                        BEQ     6$              ;YES, CONTINUE
                                
005512  104120                        ERROR   120             ;COMMAND BITS NOT ZERO
005514  032737  070000  004146  6$:   BIT     #70000,T.MR2    ;CHECK IF HEAD SELECT BITS ZERO
005522  001401                        BEQ     7$              ;YES, CONTINUE
005524  104121                        ERROR   121             ;HEAD SELECT NOT ZERO
005526  032737  100000  004146  7$:   BIT     #BIT15,T.MR2    ;CHECK PARITY BIT ON MESS A ZERO
005534  001401                        BEQ     8$              ;YES, CONTINUE
```

```
2539  005536  104122                          ERROR   122             ;PARITY ON MESS A NOT ZERO
2540  005540  032737  000017  004150  8$:     BIT     #17,T.MR3       ;CHECK MESS SELECT BITS ZERO
2541  005546  001401                          BEQ     9$              ;YES, CONTINUE
2542  005550  104123                          ERROR   123             ;MESSAGE SELECT BITS NOT ZERO
2543  005552  032737  077760  004150  9$:     BIT     #77760,T.MR3    ;CHECK CYLINDER ADDRESS BUFFER
2544  005560  001401                          BEQ     10$             ;YES, CONTINUE
2545  005562  104124                          ERROR   124             ;CYLINDER ADD BITS NOT ZERO
2546  005564  032737  100000  004150  10$:    BIT     #BIT15,T.MR3    ;CHECK PARITY BIT ON MESSAGE B
2547  005572  001401                          BEQ     TST2            ;YES, GO ON TO NEXT TEST
2548  005574  104125                          ERROR   125             ;PARITY ON MESS. B NOT ZERO
2549
2550                                  ;*****************************************************************
2551                                  ;*TEST 2        DRIVE SELECT BITS LOADING FOR DRIVE MESS.
2552                                  ;*
2553                                  ;*      INITIALIZE RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2554                                  ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 2 WITH
2555                                  ;*      ZERO.  LOAD COMMAND AND STATUS REGISTER WITH A SELECT
2556                                  ;*      COMMAND.  CLOCK IN MESSAGES A AND B INTO SHIFT REGISTER.
2557                                  ;*      MAKE SURE CORRECT MESSAGES ARE LOADED.  REPEAT FOR DRIVE
2558                                  ;*      SELECT = 1-17.
2559                                  ;*
2560                                  ;*****************************************************************
2561  005576  000004                  TST2:   SCOPE
2562  005600  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2563  005606  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
2564  005612  005037  004244                  CLR     DRVCOD          ;INITIALIZE DRIVE SELECT CODE
2565  005616  012737  000001  004160          MOV     #SELDRV,E.CS1   ;LOAD EXPECTED CS1
2566  005624  012737  005632  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2567                                                                  ;  SUBTEST LOOP
2568
2569  005632                          1$:
2570  005632  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
2571  005640  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN DIAGNOSTIC MODE
2572  005646  013762  004244  000010          MOV     DRVCOD,RKCS2(R2) ;LOAD DRIVE NUMBER
2573  005654  012762  000001  000000          MOV     #SELDRV,RKCS1(R2) ;LOAD SELECT COMMAND
2574  005662  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
2575  005666  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
2576  005674  012762  000040  000026          MOV     #DMD,RKMR1(R2)
2577  005702  005300                          DEC     R0
2578  005704  001370                          BNE     2$
2579  005706  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2580  005714  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
2581  005722  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
2582  005730  013737  004244  004206          MOV     DRVCOD,E.MR2    ;LOAD EXPECTED MAINT REG. 2
2583  005736  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
2584  005742  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2585  005750  001405                          BEQ     3$              ;YES, CHECK MESSAGE A&B
2586  005752  104002                          ERROR   2
2587  005754  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2588  005762  000426                          BR      25$             ;CHECK IF LOOP ON ERROR
2589
2590  005764  013737  004146  001160  3$:     MOV     T.MR2,$TMP0     ;MASK BITS NOT UNDER TEST
2591  005772  042737  177760  001160          BIC     #177760,$TMP0
2592  006000  023737  004244  001160          CMP     DRVCOD,$TMP0    ;CHECK IF DRIVE SELECT BITS CORRECT
2593  006006  001402                          BEQ     4$              ;YES, CHECK MESSAGES A&B
2594  006010  104003                          ERROR   3               ;DRIVE SELECT BITS INCORRECT
```

```
2595  006012  000412                          BR      25$             ;CHECK IF LOOP ON ERROR
2596
2597  006014  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
2598  006022  001401                          BEQ     5$              ;YES, CHECK MESSAGE B
2599  006024  104004                          ERROR   4               ;MESSAGE A INCORRECT
2600  006026  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
2601  006034  001401                          BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
2602  006036  104005                          ERROR   5               ;MESSAGE B INCORRECT
2603  006040  104415                  25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
2604  006042  005237  004244                  INC     DRVCOD          ;INCREMENT DRIVE SELECT CODE
2605  006046  022737  000017  004244          CMP     #17,DRVCOD      ;CHECK IF FINISHED
2606  006054  103266                          BHIS    1$              ;NO, TRY NEXT CONFIGURATION
2607
2608                                  ;;*****************************************************************
2609                                  ;*TEST 3         FORMAT BIT LOADING TO FOR DRIVE MESS.
2610                                  ;*
2611                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2612                                  ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
2613                                  ;*      A SELECT COMMAND AND 24 SECTOR MODE FORMAT.  MAKE SURE
2614                                  ;*      CORRECT MESSAGE IS LOADED.
2615                                  ;*
2616                                  ;;*****************************************************************
2617  006056  000004                  TST3:   SCOPE
2618  006060  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2619  006066  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
2620  006072  012737  052033  001300          MOV     #EM100,EM1N     ;LOAD ERROR MESSAGE
2621  006100  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
2622  006106  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
2623  006114  012762  010001  000000          MOV     #CFMT!SELDRV,RKCS1(R2)  ;LOAD CFMT!SELDRV INTO COMMAND AND STATUS REG.
2624  006122  012737  010001  004160          MOV     #CFMT!SELDRV,E.CS1      ;LOAD EXPECT CS1
2625  006130  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGES
2626  006134  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
2627  006142  012762  000040  000026          MOV     #DMD,RKMR1(R2)
2628  006150  005300                          DEC     R0
2629  006152  001370                          BNE     1$
2630  006154  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2631  006162  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
2632  006170  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
2633  006176  012737  001000  004206          MOV     #S.FMT,E.MR2    ;LOAD EXPECTED MAINT REG. 2
2634  006204  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
2635  006210  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2636  006216  001410                          BEQ     2$              ;YES, CHECK MESSAGE A&B
2637  006220  012737  057141  001302          MOV     #EM2003,EM1N+2  ;LOAD ERROR MESSAGE
2638  006226  104001                          ERROR   1
2639  006230  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT TEST
2640  006236  000431                          BR      TST4            ;;GO ON TO NEXT TEST
2641
2642  006240  032737  001000  004146  2$:     BIT     #S.FMT,T.MR2    ;CHECK IF S.FMT SET IN MESSAGE A
2643  006246  001005                          BNE     3$              ;YES, CHECK MESSAGES A&B
2644  006250  012737  057005  001302          MOV     #EM2000,EM1N+2  ;LOAD ERROR MESSAGE
2645  006256  104001                          ERROR   1
2646  006260  000420                          BR      TST4            ;;GO ON TO NEXT TEST
2647
2648  006262  023737  004206  004146  3$:     CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
2649  006270  001404                          BEQ     4$              ;YES, CHECK MESSAGE B
2650  006272  012737  057063  001302          MOV     #EM2001,EM1N+2  ;LOAD ERROR MESSAGE
```

```
                                        ERROR   1
006302  023737  004210  004150  4$:     CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
006310  001404                          BEQ     TST4            ;;YES, GO ON TO NEXT TEST
006312  012737  057112  001302          MOV     #EM2002,EM1N+2  ;LOAD ERROR MESSAGE
006320  104001                          ERROR   1

                                ;;*****************************************************************
                                ;*TEST 4         HEAD SELECT BITS LOADING FOR DRIVE MESS.
                                ;*
                                ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                ;*      DIAGNOSTIC MODE.  LOAD TRACK ADDRESS WITH ZERO.  LOAD
                                ;*      COMMAND AND STATUS REGISTER 2 WITH ZERO.  LOAD COMMAND
                                ;*      AND STATUS REGISTER WITH SELECT COMMAND.  CLOCK IN
                                ;*      MESSAGE A AND B INTO SHIFT REGISTER.  MAKE SURE CORRECT
                                ;*      MESSAGE IS LOADED.  REPEAT FOR TRACK ADDRESS = 1-7.
                                ;*
                                ;*****************************************************************
006322  000004                  TST4:   SCOPE
006324  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
006332  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
006336  005037  004250                  CLR     HDCODE          ;CLEAR HEAD SELECT CODE
006342  012737  000001  004160          MOV     #SELDRV,E.CS1   ;LOAD EXPECTED CS1
006350  012737  006356  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
                                                                ;  SUBTEST LOOP

006356                          1$:
006356  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
006364  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN DIAGNOSTIC MODE
006372  005046                          CLR     -(SP)           ;MAKE ROOM ON STACK
006374  113766  004250  000001          MOVB    HDCODE,1(SP)    ;LOAD HEAD ADDRESS
006402  012662  000006                  MOV     (SP)+,RKDA(R2)
006406  012762  000001  000000          MOV     #SELDRV,RKCS1(R2) ;LOAD SELECT COMMAND
006414  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
006420  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
006426  012762  000040  000026          MOV     #DMD,RKMR1(R2)
006434  005300                          DEC     R0
006436  001370                          BNE     2$
006440  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMMAND AND STATUS REG. 1
006446  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
006454  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
006462  005037  004206                  CLR     E.MR2
006466  113737  004250  004207          MOVB    HDCODE,E.MR2+1  ;GENERATE EXPECTED MAINT REG. 2
006474  006337  004206                  ASL     E.MR2
006500  006337  004206                  ASL     E.MR2
006504  006337  004206                  ASL     E.MR2
006510  006337  004206                  ASL     E.MR2
006514  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
006520  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
006526  001405                          BEQ     3$              ;YES, CHECK MESSAGE A&B
006530  104006                          ERROR   6
006532  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
006540  000426                          BR      25$             ;CHECK IF LOOP ON ERROR

006542  013737  004146  001160  3$:     MOV     T.MR2,$TMP0     ;MASK BITS NOT UNDER TEST
006550  042737  103777  001160          BIC     #103777,$TMP0
006556  023737  004206  001160          CMP     E.MR2,$TMP0     ;CHECK IF HEAD SELECT BITS CORRECT
```

```
2707  006564  001402                          BEQ     4$              ;YES, CHECK MESSAGES A&B
2708  006566  104007                          ERROR   7               ;HEAD SELECT BITS INCORRECT
2709  006570  000412                          BR      25$             ;CHECK IF LOOP ON ERROR
2710
2711  006572  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
2712  006600  001401                          BEQ     5$              ;YES, CHECK MESSAGE B
2713  006602  104010                          ERROR   10              ;MESSAGE A INCORRECT
2714  006604  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
2715  006612  001401                          BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
2716  006614  104011                          ERROR   11              ;MESSAGE B INCORRECT
2717  006616  104415                  25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
2718  006620  005237  004250                  INC     HDCODE          ;INCREMENT HEAD SELECT CODE F
2719  006624  022737  000007  004250          CMP     #7,HDCODE       ;CHECK IF FINISHED
2720  006632  103251                          BHIS    1$              ;NO, TRY NEXT CONFIGURATION
2721
2722                                  ;;*****************************************************************
2723                                  ;*TEST 5        MESSAGE SELECT BITS LOADING FOR DRIVE MESS.
2724                                  ;*
2725                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2726                                  ;*      DIAGNOSTIC MODE AND ZERO IN MESSAGE SELECT BITS.  LOAD
2727                                  ;*      COMMAND AND STATUS REGISTER 1 WITH A SELECT COMMAND.  CLOCK
2728                                  ;*      IN MESSAGE A AND B INTO SHIFT REGISTER.   MAKE SURE
2729                                  ;*      CORRECT MESSAGE IS LOADED.  REPEAT FOR MESSAGE SELECT = 1-17.
2730                                  ;*
2731                                  ;;*****************************************************************
2732  006634  000004                  TST5:   SCOPE
2733  006636  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2734  006644  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
2735  006650  005037  004246                  CLR     MSGCOD          ;INITIALIZE MESSAGE SELECT
2736  006654  012737  000001  004160          MOV     #SELDRV,E.CS1   ;LOAD EXPECTED CS1
2737  006662  012737  006670  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2738                                                                  ;  SUBTEST LOOP
2739
2740  006670                          1$:
2741  006670  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
2742  006676  013762  004246  000026          MOV     MSGCOD,RKMR1(R2) ;LOAD MESSAGE SELECT BITS
2743  006704  052762  000040  000026          BIS     #DMD,RKMR1(R2)  ;PUT RK611 IN DIAGNOSTIC MODE
2744  006712  012762  000001  000000          MOV     #SELDRV,RKCS1(R2) ;LOAD SELECT COMMAND
2745  006720  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IF DRIVE MESSAGE
2746  006724  052762  000400  000026  2$:     BIS     #MCLK,RKMR1(R2)
2747  006732  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
2748  006740  005300                          DEC     R0
2749  006742  001370                          BNE     2$
2750  006744  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2751  006752  016237  000026  004144          MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
2752  006760  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2753  006766  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2754  006774  013737  004246  004204          MOV     MSGCOD,E.MR1    ;LOAD EXPECTED MAINT REG. 1
2755  007002  052737  002040  004204          BIS     #MEWD!DMD,E.MR1
2756  007010  032737  020000  004144          BIT     #ECCW,T.MR1
2757  007016  001403                          BEQ     10$
2758  007020  052737  020000  004204          BIS     #ECCW,E.MR1
2759  007026  005037  004206          10$:    CLR     E.MR2           ;LOAD EXPECTED MAINT REG. 2
2760  007032  013737  004246  004210          MOV     MSGCOD,E.MR3    ;LOAD EXPECTED MAINT REG. 3
2761  007040  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2762  007046  001405                          BEQ     3$              ;YES, CHECK MAINT REG. 1
```

```
                                        ERROR   12                      ;CS1 INCORRECT
007050  104012
007052  012762  100000  000000          MOV     #CCLR,RKCS1(R2)         ;CLEAN UP FOR NEXT CONFIGURATION
007060  000437                          BR      25$                     ;CHECK IF LOOP ON ERROR

007062  023737  004204  004144  3$:     CMP     E.MR1,T.MR1             ;CHECK IF MAINT REG. 1 CORRECT
007070  001405                          BEQ     4$                      ;YES, CHECK MESSAGE A&B
007072  104013                          ERROR   13                      ;MR1 INCORRECT
007074  012762  100000  000000          MOV     #CCLR,RKCS1(R2)         ;CLEAN UP FOR NEXT CONFIGURATION
007102  000426                          BR      25$                     ;CHECK IF LOOP ON ERROR

007104  013737  004150  001160  4$:     MOV     T.MR3,$TMPO             ;MASK BITS NOT UNDER TEST
007112  042737  177760  001160          BIC     #177760,$TMPO
007120  023737  004246  001160          CMP     MSGCOD,$TMPO            ;CHECK IF MESSAGE SELECT CODE CORRECT
007126  001402                          BEQ     5$                      ;YES, CHECK MESSAGES A&B
007130  104014                          ERROR   14                      ;MESSAGE SELECT CODE INCORRECT
007132  000412                          BR      25$

007134  023737  004206  004146  5$:     CMP     E.MR2,T.MR2             ;CHECK IF MESSAGE A CORRECT
007142  001401                          BEQ     6$                      ;YES, CHECK MESSAGE B
007144  104015                          ERROR   15                      ;MESSAGE A INCORRECT
007146  023737  004210  004150  6$:     CMP     E.MR3,T.MR3             ;CHECK IF MESSAGE B CORRECT
007154  001401                          BEQ     25$                     ;YES, CHECK IF LOOP ON ERROR
007156  104016                          ERROR   16                      ;MESSAGE B INCORRECT
007160  104415                  25$:    SCOPE                           ;CHECK IF LOOP ON ERROR
007162  005237  004246                  INC     MSGCOD                  ;INCREMENT MESSAGE SELECT CODE
007166  022737  000017  004246          CMP     #17,MSGCOD              ;CHECK IF FINISHED
007174  103235                          BHIS    1$                      ;NO, TRY NEXT CONFIGURATION

                        ;*****************************************************************
                        ;*TEST 6        CLEAR DRIVE COMMAND LOADING FOR DRIVE MESS
                        ;*
                        ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                        ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
                        ;*      A DRIVE CLEAR.  CLOCK MESSAGE A AND B INTO SHIFT REGISTERS.
                        ;*      MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.  REPEAT
                        ;*      FOR 24 SECTOR FORMAT.
                        ;*
                        ;*****************************************************************
007176  000004         TST6:   SCOPE
007200  012737  000144  001200          MOV     #100.,$TIMES            ;;DO 100. ITERATIONS
007206  013702  001270                  MOV     $BASE,R2                ;LOAD RK611 BASE
007212  012737  052121  001300          MOV     #EM101,EMIN             ;LOAD ERROR MESSAGE
007220  012737  000005  004160          MOV     #CLEAR,E.CS1            ;LOAD EXPECTED COMMAND AND STATUS REG. 1
007226  012737  000400  004206          MOV     #S.CLR,E.MR2            ;LOAD EXPECTED MAINT. REG. 2
007234  012737  007242  001110          MOV     #1$,$LPERR              ;LOAD LOOP ON ERROR LOCATION FOR
                                                                        ;  SUBTEST LOOP

007242                          1$:
007242  012762  100000  000000          MOV     #CCLR,RKCS1(R2)         ;CLEAR RK611
007250  012762  000040  000026          MOV     #DMD,RKMR1(R2)          ;PUT RK611 IN MAINTENANCE MODE
007256  013762  004160  000000          MOV     E.CS1,RKCS1(R2)         ;LOAD CLEAR INTO COMMAND AND STATUS REG. 1
007264  012700  000016                  MOV     #3*4+2,R0               ;CLOCK IN DRIVE MESSAGE
007270  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
007276  012762  000040  000026          MOV     #DMD,RKMR1(R2)
007304  005300                          DEC     R0
007306  001370                          BNE     2$
```

```
2819  007310  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2820  007316  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2821  007324  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2822  007332  005037  004210                  CLR     E.MR3           ;STORE EXPECTED MAINT REG. 3
2823  007336  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2824  007344  001410                          BEQ     3$              ;YES, CHECK MESSAGE A&B
2825  007346  012737  057141  001302          MOV     #EM2003,EM1N+2  ;LOAD ERROR MESSAGE
2826  007354  104001                          ERROR   1
2827  007356  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2828  007364  000437                          BR      25$             ;CHECK IF LOOP ON ERROR
2829
2830  007366  013737  004146  001160  3$:     MOV     T.MR2,$TMP0     ;MASK BITS NOT UNDER TEST
2831  007374  042737  176377  001160          BIC     #^C<S.FMT!S.CLR>,$TMP0
2832  007402  023737  004206  001160          CMP     E.MR2,$TMP0     ;CHECK IF S.CLR AND FORMAT
2833                                                                  ;  BITS IN MESSAGE CORRECT
2834  007410  001405                          BEQ     4$              ;YES, CHECK MESSAGE A&B
2835  007412  012737  057005  001302          MOV     #EM2000,EM1N+2  ;LOAD ERROR MESSAGE
2836  007420  104001                          ERROR   1
2837  007422  000420                          BR      25$             ;CHECK IF LOOP ON ERROR
2838
2839  007424  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
2840  007432  001404                          BEQ     5$              ;YES, CHECK MESSAGE B
2841  007434  012737  057063  001302          MOV     #EM2001,EM1N+2  ;LOAD ERROR MESSAGE
2842  007442  104001                          ERROR   1
2843  007444  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
2844  007452  001404                          BEQ     25$             ;YES, CHECK IF LOOP ON EROR
2845  007454  012737  057112  001302          MOV     #EM2002,EM1N+2  ;LOAD ERROR MESSAGE
2846  007462  104001                          ERROR   1
2847  007464  104415          25$:            SCOP1                   ;CHECK IF LOOP ON ERROR
2848  007466  032737  010000  004160          BIT     #CFMT,E.CS1     ;CHECK IF ISSUED IN 24 SECTOR FORMAT
2849  007474  001007                          BNE     TST7            ;;YES, GO ON TO NEXT TEST
2850  007476  052737  010000  004160          BIS     #CFMT,E.CS1     ;INDICATE COMMAND IN 24 SECTOR FORMAT
2851  007504  052737  001000  004206          BIS     #S.FMT,E.MR2
2852  007512  000653                          BR      1$              ;REISSUE IN 24 SECTOR FORMAT
2853
2854                                  ;*****************************************************************
2855                                  ;*TEST 7        UNLOAD COMMAND LOADING FOR DRIVE MESS.
2856                                  ;*
2857                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2858                                  ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
2859                                  ;*      AN UNLOAD COMMAND.  CLOCK IN MESSAGES A AND B INTO SHIFT
2860                                  ;*      REGISTERS.  MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
2861                                  ;*      REPEAT FOR 24 SECTOR FORMAT.
2862                                  ;*
2863                                  ;*****************************************************************
2864  007514  000004          TST7:   SCOPE
2865  007516  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2866  007524  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
2867  007530  012737  052170  001300          MOV     #EM102,EM1N     ;LOAD ERROR MESSAGE
2868  007536  012737  000007  004160          MOV     #UNLCAD,E.CS1   ;LOAD EXPECTED COMMAND AND STATUS REG. 1
2869  007544  012737  002000  004206          MOV     #S.UNLD,E.MR2   ;LOAD EXPECTED MAINT. REG. 2
2870  007552  012737  007560  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2871                                                                  ;  SUBTEST LOOP
2872
2873  007560                          1$:
2874  007560  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
```

```
2875  007566  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
2876  007574  013762  004160  000000          MOV     E.CS1,RKCS1(R2) ;LOAD UNLOAD INTO COMMAND AND STATUS REG. 1
2877  007602  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
2878  007606  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
2879  007614  012762  000040  000026          MOV     #DMD,RKMR1(R2)
2880  007622  005300                          DEC     R0
2881  007624  001370                          BNE     2$
2882  007626  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2883  007634  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2884  007642  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2885  007650  005037  004210                  CLR     E.MR3           ;STORE EXPECTED MAINT REG. 3
2886  007654  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2887  007662  001410                          BEQ     3$              ;YES, CHECK MESSAGE A&B
2888  007664  012737  057141  001302          MOV     #EM2003,EMIN+2  ;LOAD ERROR MESSAGE
2889  007672  104001                          ERROR   1
2890  007674  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2891  007702  000437                          BR      25$             ;CHECK IF LOOP ON ERROR
2892
2893  007704  013737  004146  001160  3$:     MOV     T.MR2,$TMP0     ;MASK BITS NOT UNDER TEST
2894  007712  042737  174777  001160          BIC     #^C<S.FMT!S.UNLD>,$TMP0
2895  007720  023737  004206  001160          CMP     E.MR2,$TMP0     ;CHECK IF S.UNLD AND FORMAT
2896                                                                  ;  BITS IN MESSAGE CORRECT
2897  007726  001405                          BEQ     4$              ;YES, CHECK MESSAGE A&B
2898  007730  012737  057005  001302          MOV     #EM2000,EMIN+2  ;LOAD ERROR MESSAGE
2899  007736  104001                          ERROR   1
2900  007740  000420                          BR      25$             ;CHECK IF LOOP ON ERROR
2901
2902  007742  023[illegible]  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
2903  007750  001404                          BEQ     5$              ;YES, CHECK MESSAGE B
2904  007752  012737  057063  001302          MOV     #EM2001,EMIN+2  ;LOAD ERROR MESSAGE
2905  007760  104001                          ERROR   1
2906  007762  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
2907  007770  001404                          BEQ     25$             ;YES, CHECK IF LOOP ON EROR
2908  007772  012737  057112  001302          MOV     #EM2002,EMIN+2  ;LOAD ERROR MESSAGE
2909  010000  104001                          ERROR   1
2910  010002  104415                  25$:    SCOPE                   ;CHECK IF LOOP ON ERROR
2911  010004  032737  010000  004160          BIT     #CFMT,E.CS1     ;CHECK IF ISSUED IN 24 SECTOR FORMAT
2912  010012  001007                          BNE     TST10           ;;YES, GO ON TO NEXT TEST
2913  010014  052737  010000  004160          BIS     #CFMT,E.CS1     ;INDICATE COMMAND IN 24 SECTOR FORMAT
2914  010022  052737  001000  004206          BIS     #S.FMT,E.MR2
2915  010030  000653                          BR      1$              ;REISSUE IN 24 SECTOR FORMAT
2916
2917                                  ;;*************************************************************
2918                                  ;*TEST 10       PACK ACKNOWLEDGE COMMAND LOADING FOR DRIVE MESS.
2919                                  ;*
2920                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2921                                  ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
2922                                  ;*      A PACK ACKNOWLEDGE.  CLOCK MESSAGES A AND B INTO SHIFT
2923                                  ;*      REGISTERS.  MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
2924                                  ;*      REPEAT FOR 24 SECTOR FORMAT.
2925                                  ;*
2926                                  ;;*************************************************************
2927  010032  000004                  TST10:  SCOPE
2928  010034  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2929  010042  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
2930  010046  012737  052232  001300          MOV     #EM103,EMIN     ;LOAD ERROR MESSAGE
```

```
2931  010054  012737  000003  004160          MOV     #PACK,E.CS1     ;LOAD EXPECTED COMMAND AND STATUS REG. 1
2932  010062  012737  004000  004206          MOV     #S.PACK,E.MR2   ;LOAD EXPECTED MAINT. REG. 2
2933  010070  012737  010076  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2934                                                                  ;  SUBTEST LOOP
2935
2936  010076                          1$:
2937  010076  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
2938  010104  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
2939  010112  013762  004160  000000          MOV     E.CS1,RKCS1(R2) ;LOAD PACK INTO COMMAND AND STATUS REG. 1
2940  010120  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
2941  010124  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
2942  010132  012762  000040  000026          MOV     #DMD,RKMR1(R2)
2943  010140  005300                          DEC     R0
2944  010142  001370                          BNE     2$
2945  010144  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2946  010152  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2947  010160  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2948  010166  005037  004210                  CLR     E.MR3           ;STORE EXPECTED MAINT REG. 3
2949  010172  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
2950  010200  001410                          BEQ     3$              ;YES, CHECK MESSAGE A&B
2951  010202  012737  057141  001302          MOV     #EM2003,EMIN+2  ;LOAD ERROR MESSAGE
2952  010210  104001                          ERROR   1
2953  010212  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2954  010220  000437                          BR      25$             ;CHECK IF LOOP ON ERROR
2955
2956  010222  013737  004146  001160  3$:     MOV     T.MR2,$TMP0     ;MASK BITS NOT UNDER TEST
2957  010230  042737  172777  001160          BIC     #^C<S.FMT!S.PACK>,$TMP0
2958  010236  023737  004206  001160          CMP     E.MR2,$TMP0     ;CHECK IF S.PACK AND FORMAT
2959                                                                  ;  BITS IN MESSAGE CORRECT
2960  010244  001405                          BEQ     4$              ;YES, CHECK MESSAGE A&B
2961  010246  012737  057005  001302          MOV     #EM2000,EMIN+2  ;LOAD ERROR MESSAGE
2962  010254  104001                          ERROR   1
2963  010256  000420                          BR      25$             ;CHECK IF LOOP ON ERROR
2964
2965  010260  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
2966  010266  001404                          BEQ     5$              ;YES, CHECK MESSAGE B
2967  010270  012737  057063  001302          MOV     #EM2001,EMIN+2  ;LOAD ERROR MESSAGE
2968  010276  104001                          ERROR   1
2969  010300  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
2970  010306  001404                          BEQ     25$             ;YES, CHECK IF LOOP ON EROR
2971  010310  012737  057112  001302          MOV     #EM2002,EMIN+2  ;LOAD ERROR MESSAGE
2972  010316  104001                          ERROR   1
2973  010320  104415                  25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
2974  010322  032737  010000  004160          BIT     #CFMT,E.CS1     ;CHECK IF ISSUED IN 24 SECTOR FORMAT
2975  010330  001007                          BNE     TST11           ;;YES, GO ON TO NEXT TEST
2976  010332  052737  010000  004160          BIS     #CFMT,E.CS1     ;INDICATE COMMAND IN 24 SECTOR FORMAT
2977  010340  052737  001000  004206          BIS     #S.FMT,E.MR2
2978  010346  000653                          BR      1$              ;REISSUE IN 24 SECTOR FORMAT
2979
2980                                  ;;*****************************************************************
2981                                  ;*TEST 11        RECALIBRATE COMMAND LOADING FOR DRIVE MESS.
2982                                  ;*
2983                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
2984                                  ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
2985                                  ;*      A RECALIBRATE.  CLOCK MESSAGES A AND B INTO SHIFT REGISTERS.
2986                                  ;*      MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
```

```
2987                                    ;*
2988                                    ;**********************************************************************
2989  010350  000004                    TST11:  SCOPE
2990  010352  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
2991  010360  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
2992  010364  012737  052306  001300            MOV     #EM104,EM1N     ;LOAD ERROR MESSAGE
2993  010372  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
2994  010400  012762  000040  000026            MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
2995  010406  012762  000013  000000            MOV     #RECAL,RKCS1(R2)        ;LOAD RECAL INT. COMMAND AND STATUS REG.  1
2996  010414  012737  000013  004160            MOV     #RECAL,E.CS1    ;LOAD EXPECT CS1
2997  010422  012700  000016                    MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGES
2998  010426  012762  000440  000026  1$:       MOV     #DMD!MCLK,RKMR1(R2)
2999  010434  012762  000040  000026            MOV     #DMD,RKMR1(R2)
3000  010442  005300                            DEC     R0
3001  010444  001370                            BNE     1$
3002  010446  016237  000000  004120            MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3003  010454  016237  000034  004146            MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3004  010462  016237  000036  004150            MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3005  010470  012737  000040  004206            MOV     #S.RECL,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3006  010476  005037  004210                    CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
3007  010502  023737  004160  004120            CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3008  010510  001410                            BEQ     2$              ;YES, CHECK MESSAGE A&B
3009  010512  012737  057141  001302            MOV     #EM2003,EM1N+2  ;LOAD ERROR MESSAGE
3010  010520  104001                            ERROR   1
3011  010522  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT TEST
3012  010530  000431                            BR      TST12           ;;GO ON TO NEXT TEST
3013
3014  010532  032737  000040  004146  2$:       BIT     #S.RECL,T.MR2   ;CHECK IF S.RECL SET IN MESSAGE A
3015  010540  001005                            BNE     3$              ;YES, CHECK MESSAGES A&B
3016  010542  012737  057005  001302            MOV     #EM2000,EM1N+2  ;LOAD ERROR MESSAGE
3017  010550  104001                            ERROR   1
3018  010552  000420                            BR      TST12           ;;GO ON TO NEXT TEST
3019
3020  010554  023737  004206  004146  3$:       CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
3021  010562  001404                            BEQ     4$              ;YES, CHECK MESSAGE B
3022  010564  012737  057063  001302            MOV     #EM2001,EM1N+2  ;LOAD ERROR MESSAGE
3023  010572  104001                            ERROR   1
3024  010574  023737  004210  004150  4$:       CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
3025  010602  001404                            BEQ     TST12           ;;YES, GO ON TO NEXT TEST
3026  010604  012737  057112  001302            MOV     #EM2002,EM1N+2  ;LOAD ERROR MESSAGE
3027  010612  104001                            ERROR   1
3028
3029                                    ;;**********************************************************************
3030                                    ;*TEST 12       START SPINDLE COMMAND LOADING FOR DRIVE MESS.
3031                                    ;*
3032                                    ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3033                                    ;*      DIAGNOSTIC MODE.  LOAD COMMAND AND STATUS REGISTER 1 WITH
3034                                    ;*      A START SPINDLE.  CLOCK MESSAGES A AND B INTO SHIFT REGISTERS.
3035                                    ;*      MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
3036                                    ;*
3037                                    ;**********************************************************************
3038  010614  000004                    TST12:  SCOPE
3039  010616  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3040  010624  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
3041  010630  012737  052355  001300            MOV     #EM105,EM1N     ;LOAD ERROR MESSAGE
3042  010636  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
```

```
 3043  010644  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
 3044  010652  012762  000011  000000          MOV     #SRTSPL,RKCS1(R2)       ;LOAD SRTSPL INTO COMMAND AND STATUS REG.  1
 3045  010660  012737  000011  004160          MOV     #SRTSPL,E.CS1   ;LOAD EXPECT CS1
 3046  010666  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGES
 3047  010672  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 3048  010700  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 3049  010706  005300                          DEC     R0
 3050  010710  001370                          BNE     1$
 3051  010712  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
 3052  010720  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
 3053  010726  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
 3054  010734  012737  000100  004206          MOV     #S.STSP,E.MR2   ;LOAD EXPECTED MAINT REG. 2
 3055  010742  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
 3056  010746  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
 3057  010754  001410                          BEQ     2$              ;YES, CHECK MESSAGE A&B
 3058  010756  012737  057141  001302          MOV     #EM2003,EMIN+2  ;LOAD ERROR MESSAGE
 3059  010764  104001                          ERROR   1
 3060  010766  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT TEST
 3061  010774  000431                          BR      TST13           ;;GO ON TO NEXT TEST
 3062
 3063  010776  032737  000100  004146  2$:     BIT     #S.STSP,T.MR2   ;CHECK IF S.STSP SET IN MESSAGE A
 3064  011004  001005                          BNE     3$              ;YES, CHECK MESSAGES A&B
 3065  011006  012737  057005  001302          MOV     #EM2000,EMIN+2  ;LOAD ERROR MESSAGE
 3066  011014  104001                          ERROR   1
 3067  011016  000420                          BR      TST13           ;;GO ON TO NEXT TEST
 3068
 3069  011020  023737  004206  004146  3$:     CMP     E.MR2,T.MR2     ;CHECK IF DRIVE MESSAGE A CORRECT
 3070  011026  001404                          BEQ     4$              ;YES, CHECK MESSAGE B
 3071  011030  012737  057063  001302          MOV     #EM2001,EMIN+2  ;LOAD ERROR MESSAGE
 3072  011036  104001                          ERROR   1
 3073  011040  023737  004210  004150  4$:     CMP     E.MR3,T.MR3     ;CHECK IF DRIVE MESSAGE B CORRECT
 3074  011046  001404                          BEQ     TST13           ;;YES, GO ON TO NEXT TEST
 3075  011050  012737  057112  001302          MOV     #EM2002,EMIN+2  ;LOAD ERROR MESSAGE
 3076  011056  104001                          ERROR   1
 3077
 3078                                          ;*****************************************************************
 3079                                          ;*TEST 13       SEEK AND CYLINDER ADD 0-777 LOADING FOR DRIVE MESS
 3080                                          ;*
 3081                                          ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
 3082                                          ;*      DIAGNOSTIC MODE.  LOAD ZERO IN CYLINDER ADDRESS.  LOAD
 3083                                          ;*      COMMAND AND STATUS REGISTER 1 WITH A SEEK COMMAND.
 3084                                          ;*      CLOCK IN MESSAGE A AND B INTO SHIFT REGISTER.  MAKE SURE
 3085                                          ;*      CORRECT MESSAGE IS LOADED.  REPEAT FOR CYLINDER = 1-777.
 3086                                          ;*
 3087                                          ;*****************************************************************
 3088  011060  000004                  TST13:  SCOPE
 3089  011062  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 3090  011070  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 3091  011074  005037  004252                  CLR     CYLIN           ;INITIALIZE CYLINDER
 3092  011100  012737  000017  004160          MOV     #SEEK,E.CS1     ;LOAD EXPECTED CS1
 3093  011106  012737  011114  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
 3094                                                                  ;  SUBTEST LOOP
 3095
 3096  011114                           1$:
 3097  011114  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
 3098  011122  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
```

```
3099  011130  013762  004252  000020          MOV     CYLIN,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
3100  011136  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
3101  011144  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3102  011150  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
3103  011156  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3104  011164  005300                          DEC     R0
3105  011166  001370                          BNE     2$
3106  011170  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3107  011176  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3108  011204  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3109  011212  012737  000020  004206          MOV     #S.SEEK,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3110  011220  013737  004252  004210          MOV     CYLIN,E.MR3     ;GENERATE EXPECTED MAINT REG. 3
3111  011226  006337  004210                  ASL     E.MR3
3112  011232  006337  004210                  ASL     E.MR3
3113  011236  006337  004210                  ASL     E.MR3
3114  011242  006337  004210                  ASL     E.MR3
3115  011246  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3116  011254  001405                          BEQ     3$              ;YES, CHECK MESSAGE A&B
3117  011256  104017                          ERROR   17
3118  011260  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
3119  011266  000434                          BR      25$             ;CHECK IF LOOP ON ERROR
3120
3121  011270  032737  000020  004146  3$:     BIT     #S.SEEK,T.MR2   ;CHECK IF SEEK COMMAND BIT SET
3122  011276  001002                          BNE     4$              ;YES, CHECK CYLINDER ADDRESS BITS
3123  011300  104020                          ERROR   20              ;SEEK BIT NOT SET
3124  011302  000426                          BR      25$             ;CHECK IF LOOP ON ERROR
3125
3126  011304  013737  004150  001160  4$:     MOV     T.MR3,$TMP0     ;MASK BITS NOT UNDER TEST
3127  011312  042737  140017  001160          BIC     #140017,$TMP0
3128  011320  023737  004210  001160          CMP     E.MR3,$TMP0     ;CHECK IF CYLINDER ADDRESS BITS CORRECT
3129  011326  001402                          BEQ     5$              ;YES, CHECK MESSAGES A&B
3130  011330  104021                          ERROR   21              ;CYLINDER ADDRESS BITS INCORRECT
3131  011332  000412                          BR      25$             ;CHECK IF LOOP ON ERROR
3132
3133  011334  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSSAGE A CORRECT
3134  011342  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
3135  011344  104022                          ERROR   22              ;MESSAGE A INCORRECT
3136  011346  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3137  011354  001401                          BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
3138  011356  104023                          ERROR   23              ;MESSAGE B INCORRECT
3139  011360  104415                  25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
3140  011362  005237  004252                  INC     CYLIN           ;INCREMENT CYLINDER NUMBER
3141  011366  022737  000777  004252          CMP     #777,CYLIN      ;CHECK IF FINISHED
3142  011374  103247                          BHIS    1$              ;NO, TRY NEXT CONFIGURATION
3143
3144                                  ;;*****************************************************************
3145                                  ;*TEST 14       SEEK AND CYLINDER BIT 9 AND RK06 FOR DRIVE MESS.
3146                                  ;*
3147                                  ;*       CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3148                                  ;*       DIAGNOSTIC MODE.  LOAD 1000 IN CYLINDER ADDRESS.  LOAD
3149                                  ;*       COMMAND AND STATUS REGISTER 1 WITH A SEEK COMMAND.
3150                                  ;*       CLOCK IN MESSAGE A AND B INTO SHIFT REGISTERS.  MAKE
3151                                  ;*       SURE CYLINDER BIT 9 IN MESSAGE IN RESET.  REPEAT FOR
3152                                  ;*       CYLINDER = 1400.
3153                                  ;*
3154                                  ;;*****************************************************************
```

```
                                TST14:  SCOPE
3155  011376  000004                    
3156  011400  012737  000144  001200      MOV     #100.,$TIMES      ;;DO 100. ITERATIONS
3157  011406  013702  001270              MOV     $BASE,R2          ;LOAD RK611 BASE
3158  011412  012737  001000  004252      MOV     #1000,CYLIN       ;INITIALIZE CYLINDER
3159  011420  005037  004210              CLR     E.MR3             ;LOAD EXPECTED
3160  011424  012737  000017  004160      MOV     #SEEK,E.CS1       ;LOAD EXPECTED CS1
3161  011432  012737  011440  001110      MOV     #1$,$LPERR        ;LOAD LOOP ON ERROR LOCATION FOR
3162                                                                ;  SUBTEST LOOP
3163
3164  011440                          1$:
3165  011440  012762  100000  000000      MOV     #CCLR,RKCS1(R2)   ;CLEAR RK611
3166  011446  012762  000040  000026      MOV     #DMD,RKMR1(R2)    ;PUT RK611 IN MAINT MODE
3167  011454  013762  004252  000020      MOV     CYLIN,RKDCYL(R2)  ;LOAD CYLINDER ADDRESS
3168  011462  012762  000017  000000      MOV     #SEEK,RKCS1(R2)   ;ISSUE SEEK
3169  011470  012700  000016              MOV     #3*4+2,R0         ;CLOCK IN DRIVE MESSAGE
3170  011474  012762  000440  000026  2$: MOV     #DMD!MCLK,RKMR1(R2)
3171  011502  012762  000040  000026      MOV     #DMD,RKMR1(R2)
3172  011510  005300                      DEC     R0
3173  011512  001370                      BNE     2$
3174  011514  016237  000000  004120      MOV     RKCS1(R2),T.CS1   ;STORE COMMAND AND STATUS REG. 1
3175  011522  016237  000034  004146      MOV     RKMR2(R2),T.MR2   ;STORE MAINT REG.2
3176  011530  016237  000036  004150      MOV     RKMR3(R2),T.MR3   ;STORE MAINT REG.3
3177  011536  012737  000020  004206      MOV     #S.SEEK,E.MR2     ;LOAD EXPECTED MAINT REG. 2
3178  011544  023737  004160  004120      CMP     E.CS1,T.CS1       ;CHECK IF CS1 CORRECT
3179  011552  001405                      BEQ     3$                ;YES, CHECK MESSAGE A&B
3180  011554  104017                      ERROR   17
3181  011556  012762  100000  000000      MOV     #CCLR,RKCS1(R2)   ;CLEAN UP FOR NEXT CONFIGURATION
3182  011564  000434                      BR      25$               ;CHECK IF LOOP ON ERROR
3183
3184  011566  032737  000020  004146  3$: BIT     #S.SEEK,T.MR2     ;CHECK IF SEEK COMMAND BIT SEEK
3185  011574  001002                      BNE     4$                ;YES, CHECK CYLINDER ADDRESS BITS
3186  011576  104020                      ERROR   20                ;SEEK BIT NOT SET
3187  011600  000426                      BR      25$               ;CHECK IF LOOP ON ERROR
3188
3189  011602  013737  004150  001160  4$: MOV     T.MR3,$TMP0       ;MASK BITS NOT UNDER TEST
3190  011610  042737  140017  001160      BIC     #140017,$TMP0
3191  011616  023737  004210  001160      CMP     E.MR3,$TMP0       ;CHECK IF CYLINDER ADDRESS BITS CORRECT
3192  011624  001402                      BEQ     5$                ;YES, CHECK MESSAGES A&B
3193  011626  104021                      ERROR   21                ;CYLINDER ADDRESS BITS INCORRECT
3194  011630  000412                      BR      25$               ;CHECK IF LOOP ON ERROR
3195
3196  011632  023737  004206  004146  5$: CMP     E.MR2,T.MR2       ;CHECK IF MESSAGE A CORRECT
3197  011640  001401                      BEQ     6$                ;YES, CHECK MESSAGE B
3198  011642  104022                      ERROR   22                ;MESSAGE B INCORRECT
3199  011644  023737  004210  004150  6$: CMP     E.MR3,T.MR3       ;CHECK IF MESSAGE IS CORRECT
3200  011652  001401                      BEQ     25$               ;YES, CHECK IF LOOP ON ERROR
3201  011654  104023                      ERROR   23                ;MESSAGE INCORRECT
3202  011656  104415                 25$: SCOP1                     ;CHECK IF LOOP ON ERROR
3203  011660  022737  001400  004252      CMP     #1400,CYLIN       ;CHECK IF CYLINDER 1400
3204  011666  001407                      BEQ     TST15             ;;YES, GO ON TO NEXT TEST
3205  011670  012737  001400  004252      MOV     #1400,CYLIN       ;SET CYLINDER=1400
3206  011676  012737  010000  004210      MOV     #10000,E.MR3      ;LOAD EXPECTED CONFIGUR
3207  011704  000655                      BR      1$                ;TRY NEXT CONFIGURATION
3208
3209                                ;;*****************************************************************
3210                                ;*TEST 15        SEEK AND CYLINDER ADD 0,777-1777 LOADING FOR DRIVE MESS
```

```
                                        ;*
                                        ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
                                        ;*      DIAGNOSTIC MODE.  LOAD 0 IN CYLINDER ADDRESS.  LOAD
                                        ;*      COMMAND AND STATUS REGISTER 1 WITH SEEK COMMAND AND
                                        ;*      CDT SET.  CLOCK IN MESSAGE A AND B INTO SHIFT REGISTER.
                                        ;*      MAKE SURE CYLINDER CORRECT.  REPEAT FOR CYLINDER = 777-1777.
                                        ;*
                                        ;**************************************************************
011706  000004                        T15ST15:  SCOPE
011710  012737  000144  001200                  MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
011716  013702  001270                          MOV     $BASE,R2        ;LOAD RK611 BASE
011722  005037  004252                          CLR     CYLIN           ;INITIALIZE CYLINDER
011726  012737  002017  004160                  MOV     #CDT!SEEK,E.CS1 ;LOAD EXPECTED CS1
011734  012737  011742  001110                  MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
                                                                        ;  SUBTEST LOOP
011742                                  1$:
011742  012762  100000  000000                  MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
011750  012762  000040  000026                  MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
011756  013762  004252  000020                  MOV     CYLIN,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
011764  012762  002017  000000                  MOV     #CDT!SEEK,RKCS1(R2) ;ISSUE SEEK WITH CDT SET
011772  012700  000016                          MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
011776  012762  000440  000026          2$:     MOV     #DMD!MCLK,RKMR1(R2)
012004  012762  000040  000026                  MOV     #DMD,RKMR1(R2)
012012  005300                                  DEC     R0
012014  001370                                  BNE     2$
012016  016237  000000  004120                  MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
012024  016237  000034  004146                  MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
012032  016237  000036  004150                  MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
012040  012737  000020  004206                  MOV     #S.SEEK,E.MR2   ;LOAD EXPECTED MAINT REG. 2
012046  013737  004252  004210                  MOV     CYLIN,E.MR3     ;GENERATE EXPECTED MAINT REG. 3
012054  006337  004210                          ASL     E.MR3
012060  006337  004210                          ASL     E.MR3
012064  006337  004210                          ASL     E.MR3
012070  006337  004210                          ASL     E.MR3
012074  023737  004160  004120                  CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
012102  001405                                  BEQ     3$              ;YES, CHECK MESSAGE A&B
012104  104024                                  ERROR   24
012106  012762  100000  000000                  MOV     #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGRURATION
012114  000434                                  BR      25$             ;CHECK IF LOOP ON ERROR

012116  032737  000020  004146  3$:     BIT     #S.SEEK,T.MR2   ;CHECK IF SEEK COMMAND BIT SET
012124  001002                                  BNE     4$              ;YES, CHECK CYLINDER ADDRESS BITS
012126  104025                                  ERROR   25              ;SEEK BIT NOT SEEK
012130  000426                                  BR      25$             ;CHECK IF LOOP ON ERROR

012132  013737  004150  001160  4$:     MOV     T.MR3,$TMP0     ;MASK BITS NOT UNDER TEST
012140  042737  140017  001160                  BIC     #140017,$TMP0
012146  023737  004210  001160                  CMP     E.MR3,$TMP0     ;CHECK IF CYLINDER ADDRESS BITS CORRECT
012154  001402                                  BEQ     5$              ;YES, CHECK MESSAGES A&B
012156  104026                                  ERROR   26              ;CYLINDER ADDRESS BIT INCORRECT
012160  000412                                  BR      25$             ;CHECK IF LOOP ON ERROR

012162  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
012170  001401                                  BEQ     6$              ;YES, CHECK M MESSAGE B
012172  104027                                  ERROR   27
```

```
3267  012174  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B
3268  012202  001401                          BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
3269  012204  104030                          ERROR   30              ;MESSAGE B INCORRECT
3270  012206  104415                  25$:    SCOPE                   ;CHECK IF LOOP ON ERROR
3271  012210  005737  004252                  TST     CYLIN           ;CHECK IF ZERO
3272  012214  001003                          BNE     26$             ;NO, INCREMENT CYLINDER
3273  012216  012737  000776  004252          MOV     #776,CYLIN      ;NEXT CYLINDER=777
3274  012224  005237  004252          26$:    INC     CYLIN           ;INCREMENT CYLINDER NUMBER
3275  012230  022737  001777  004252          CMP     #1777,CYLIN     ;CHECK IF FINISHED
3276  012236  103241                          BHIS    1$              ;NO, TRY NEXT CONFIGURATION
3277
3278                                  ;;*****************************************************************
3279                                  ;*TEST 16       OFFSET COMMAND LOADING FOR DRIVE MESS.
3280                                  ;*
3281                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3282                                  ;*      DIAGNOSTIC MODE.  LOAD OFFSET REGISTER WITH 0.  LOAD
3283                                  ;*      COMMAND AND STATUS REGISTER 1 WITH AN OFFSET.  CLOCK
3284                                  ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE SHIFT
3285                                  ;*      REGISTERS ARE LOADED CORRECTLY.  REPEAT FOR OFFSET
3286                                  ;*      REGISTER = 1-377.
3287                                  ;*
3288                                  ;*****************************************************************
3289  012240  000004                  TST16:  SCOPE
3290  012242  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3291  012250  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3292  012254  005037  004254                  CLR     OFFVAL          ;INITIALIZE OFFSET VALUE
3293  012260  012737  000015  004160          MOV     #OFFSET,E.CS1   ;LOAD EXPECTED CS1
3294  012266  005037  004206                  CLR     E.MR2           ;LOAD EXPECT MAINT REG 2
3295  012272  012737  012300  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
3296                                                                  ;  SUBTEST LOOP
3297
3298  012300                          1$:
3299  012300  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3300  012306  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
3301  012314  013762  004254  000016          MOV     OFFVAL,RKASOF(R2);LOAD OFFSET VALUE
3302  012322  012762  000015  000000          MOV     #OFFSET,RKCS1(R2);ISSUE OFFSET
3303  012330  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3304  012334  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
3305  012342  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3306  012350  005300                          DEC     R0
3307  012352  001370                          BNE     2$
3308  012354  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3309  012362  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3310  012370  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3311  012376  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 2
3312  012402  013737  004254  004210          MOV     OFFVAL,E.MR3    ;GENERATE EXPECTED MR3
3313  012410  005137  004210                  COM     E.MR3
3314  012414  042737  177700  004210          BIC     #177700,E.MR3
3315  012422  006337  004210                  ASL     E.MR3
3316  012426  006337  004210                  ASL     E.MR3
3317  012432  006337  004210                  ASL     E.MR3
3318  012436  006337  004210                  ASL     E.MR3
3319  012442  052737  014000  004210          BIS     #14000,E.MR3
3320  012450  032737  000200  004254          BIT     #BIT7,OFFVAL    ;DETERMINE SIGN
3321  012456  001003                          BNE     10$
3322  012460  052737  002000  004210          BIS     #BIT10,E.MR3
```

```
3323  012466  023737  004160  004120  10$:    CMP     E.CS1,T.CS1       ;CHECK IF CS1 CORRECT
3324  012474  001405                          BEQ     4$                ;YES, CHECK MESSAGE A&B
3325  012476  104031                          ERROR   31
3326  012500  012762  100000  000000          MOV     #CCLR,RKCS1(R2)   ;CLEAN UP FOR NEXT CONFIGURATION
3327  012506  000426                          BR      25$               ;CHECK IF LOOP ON ERROR
3328
3329  012510  013737  004150  001160  4$:     MOV     T.MR3,$TMPO       ;MASK BITS NOT UNDER TEST
3330  012516  042737  140017  001160          BIC     #140017,$TMPO
3331  012524  023737  004210  001160          CMP     E.MR3,$TMPO       ;CHECK IF OFFSET VALUE CORRECT
3332  012532  001402                          BEQ     5$                ;YES, CHECK MESSAGES A&B
3333  012534  104032                          ERROR   32                ;OFFSET VALUE INCORRECT
3334  012536  000412                          BR      25$               ;CHECK IF LOOP ON ERROR
3335
3336  012540  023737  004206  004146  5$:     CMP     E.MR2,T.MR2       ;CHECK IF MESSAGE A CORRECT
3337  012546  001401                          BEQ     6$                ;YES, CHECK MESSAGE B
3338  012550  104033                          ERROR   33                ;MESSAGE A INCORRECT
3339  012552  023737  004210  004150  6$:     CMP     E.MR3,T.MR3       ;CHECK IF MESSAGE B CORRECT
3340  012560  001401                          BEQ     25$               ;YES, CHECK IF LOOP ON ERROR
3341  012562  104034                          ERROR   34                ;MESSAGE B INCORRECT
3342  012564  104415                  25$:    SCOP1                     ;CHECK IF LOOP ON ERROR
3343  012566  005237  004254                  INC     OFFVAL            ;INCREMENT OFFSET VALUE
3344  012572  022737  000377  004254          CMP     #377,OFFVAL       ;CHECK IF FINISHED
3345  012600  103237                          BHIS    1$                ;NO, TRY NEXT CONFIGURATION
3346
3347                          ;;*****************************************************************
3348                          ;*TEST 17       CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 1)
3349                          ;*
3350                          ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3351                          ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3352                          ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3353                          ;*      AND STATUS REGISTER 1 WITH A SELECT.  CLOCK
3354                          ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3355                          ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3356                          ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3357                          ;*
3358                          ;*****************************************************************
3359  012602  000004          ;ST17:  SCOPE
3360  012604  012737  000144  001200          MOV     #100.,$TIMES      ;;DO 100. ITERATIONS
3361  012612  013702  001270                  MOV     $BASE,R2          ;LOAD RK611 BASE
3362  012616  012737  001777  004252          MOV     #1777,CYLIN       ;LOAD CYLINDER VALUE
3363  012624  012737  000052  004254          MOV     #52,OFFVAL        ;LOAD OFFSET VALUE
3364  012632  012737  000001  004160          MOV     #SELDRV,E.CS1     ;LOAD EXPECTED CS1
3365  012640  012762  100000  000000          MOV     #CCLR,RKCS1(R2)   ;CLEAR RK611
3366  012646  012762  000040  000026          MOV     #DMD,RKMR1(R2)    ;PUT RK611 IN MAINTENANCE MODE
3367  012654  012762  001777  000020          MOV     #1777,RKDCYL(R2)  ;LOAD CYLINDER VALUE
3368  012662  012762  000052  000016          MOV     #52,RKASOF(R2)  ;LOAD OFFSET VALUE
3369  012670  012762  000001  000000          MOV     #SELDRV,RKCS1(R2) ;ISSUE SELDRV
3370  012676  012700  000016                  MOV     #3*4+2,R0         ;CLOCK IN DRIVE MESSAGE
3371  012702  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
3372  012710  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3373  012716  005300                          DEC     R0
3374  012720  001370                          BNE     1$
3375  012722  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3376  012730  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3377  012736  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3378  012744  012737  000000  004206          MOV     #0,E.MR2          ;LOAD EXPECTED MAINT REG. 2
```

```
3379  012752  005037  004210                    CLR     E.MR3               ;LOAD EXPECTED MAINTENANCE REG. 3
3380  012756  023737  004160  004120            CMP     E.CS1,T.CS1         ;CHECK IF CS1 CORRECT
3381  012764  001405                            BEQ     2$                  ;YES, CHECK MESSAGES A&B
3382  012766  104035                            ERROR   35
3383  012770  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3384  012776  000423                            BR      TST20               ;;GO ON TO NEXT TEST
3385
3386  013000                          2$:
3387  013000  013737  004150  001160            MOV     T.MR3,$TMP0         ;MASK OUT BITS NOT UNDER TEST
3388  013006  042737  140017  001160            BIC     #140017,$TMP0
3389  013014  001402                            BEQ     4$                  ;CHECK IF CYLINDER ADDRESS ZERO
3390  013016  104037                            ERROR   37                  ;CYLINDER ADDRESS BITS INCORRECT
3391  013020  000412                            BR      TST20               ;;GO ON TO NEXT TEST
3392
3393  013022  023737  004206  004146  4$:       CMP     E.MR2,T.MR2         ;CHECK IF MESSAGE A CORRECT
3394  013030  001401                            BEQ     5$                  ;YES, CHECK MESSAGE B
3395  013032  104040                            ERROR   40                  ;MESS A INCORRECT
3396  013034  023737  004210  004150  5$:       CMP     E.MR3,T.MR3         ;CHECK IF MESSAGE B CORRECT
3397  013042  001401                            BEQ     TST20               ;;YES, GO ON TO NEXT TEST
3398  013044  104041                            ERROR   41                  ;MESS B INCORRECT
3399
3400                                  ;;*****************************************************************
3401                                  ;*TEST 20       CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 2)
3402                                  ;*
3403                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3404                                  ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3405                                  ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3406                                  ;*      AND STATUS REGISTER 1 WITH A PACK ACKNOWLEDGE.  CLOCK
3407                                  ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3408                                  ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3409                                  ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3410                                  ;*
3411                                  ;*****************************************************************
3412  013046  000004                  TST20:  SCOPE
3413  013050  012737  000144  001200            MOV     #100.,$TIMES        ;;DO 100. ITERATIONS
3414  013056  013702  001270                    MOV     $BASE,R2            ;LOAD RK611 BASE
3415  013062  012737  001777  004252            MOV     #1777,CYLIN         ;LOAD CYLINDER VALUE
3416  013070  012737  000052  004254            MOV     #52,OFFVAL          ;LOAD OFFSET VALUE
3417  013076  012737  000003  004160            MOV     #PACK,E.CS1         ;LOAD EXPECTED CS1
3418  013104  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3419  013112  012762  000040  000026            MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
3420  013120  012762  001777  000020            MOV     #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3421  013126  012762  000052  000016            MOV     #52,RKASOF(R2) ;LOAD OFFSET VALUE
3422  013134  012762  000003  000000            MOV     #PACK,RKCS1(R2) ;ISSUE PACK
3423  013142  012700  000016                    MOV     #3*4+2,R0           ;CLOCK IN DRIVE MESSAGE
3424  013146  012762  000440  000026  1$:       MOV     #DMD!MCLK,RKMR1(R2)
3425  013154  012762  000040  000026            MOV     #DMD,RKMR1(R2)
3426  013162  005300                            DEC     R0
3427  013164  001370                            BNE     1$
3428  013166  016237  000000  004120            MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3429  013174  016237  000034  004146            MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3430  013202  016237  000036  004150            MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3431  013210  012737  004000  004206            MOV     #S.PACK,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3432  013216  005037  004210                    CLR     E.MR3               ;LOAD EXPECTED MAINTENANCE REG. 3
3433  013222  023737  004160  004120            CMP     E.CS1,T.CS1         ;CHECK IF CS1 CORRECT
3434  013230  001405                            BEQ     2$                  ;YES, CHECK MESSAGES A&B
```

```
3435  013232  104035                          ERROR   35
3436  013234  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3437  013242  000431                          BR      TST21           ;;GO ON TO NEXT TEST
3438
3439  013244                          2$:
3440  013244  032737  004000  004146          BIT     #S.PACK,T.MR2   ;CHECK IF PACK COMMAND
3441                                                                  ;  BIT SET
3442  013252  001002                          BNE     3$              ;YES, CHECK CYLINDER ADDRESS BITS
3443  013254  104036                          ERROR   36              ;S.PACK BIT NOT SET
3444  013256  000423                          BR      TST21           ;;GO ON TO NEXT TEST
3445
3446  013260                          3$:
3447  013260  013737  004150  001160          MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
3448  013266  042737  140017  001160          BIC     #140017,$TMP0
3449  013274  001402                          BEQ     4$              ;CHECK IF CYLINDER ADDRESS ZERO
3450  013276  104037                          ERROR   37              ;CYLINDER ADDRESS BITS INCORRECT
3451  013300  000412                          BR      TST21           ;;GO ON TO NEXT TEST
3452
3453  013302  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3454  013310  001401                          BEQ     5$              ;YES, CHECK MESSAGE B
3455  013312  104040                          ERROR   40              ;MESS A INCORRECT
3456  013314  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3457  013322  001401                          BEQ     TST21           ;YES, GO ON TO NEXT TEST
3458  013324  104041                          ERROR   41              ;MESS B INCORRECT
3459
3460                                  ;;*****************************************************************
3461                                  ;*TEST 21       CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 3)
3462                                  ;*
3463                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3464                                  ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3465                                  ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3466                                  ;*      AND STATUS REGISTER 1 WITH A CLEAR DRIVE.  CLOCK
3467                                  ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3468                                  ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3469                                  ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3470                                  ;*
3471                                  ;;*****************************************************************
3472  013326  000004                  TST21:  SCOPE
3473  013330  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3474  013336  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3475  013342  012737  001777  004252          MOV     #1777,CYLIN     ;LOAD CYLINDER VALUE
3476  013350  012737  000052  004254          MOV     #52,OFFVAL      ;LOAD OFFSET VALUE
3477  013356  012737  000005  004160          MOV     #CLEAR,E.CS1    ;LOAD EXPECTED CS1
3478  013364  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3479  013372  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
3480  013400  012762  001777  000020          MOV     #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3481  013406  012762  000052  000016          MOV     #52,RKASOF(R2) ;LOAD OFFSET VALUE
3482  013414  012762  000005  000000          MOV     #CLEAR,RKCS1(R2) ;ISSUE CLEAR
3483  013422  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3484  013426  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
3485  013434  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3486  013442  005300                          DEC     R0
3487  013444  001370                          BNE     1$
3488  013446  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3489  013454  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3490  013462  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
```

```
3491  013470  012737  000400  004206          MOV     #S.CLR,E.MR2    ;LOAD EXPECTED MAINT REG. 2
3492  013476  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINTENANCE REG. 3
3493  013502  023737  004160  00412C          CMP     E.CSI,T.CS1     ;CHECK IF CSI CORRECT
3494  013510  001405                          BEQ     2$              ;YES, CHECK MESSAGES A&B
3495  013512  104035                          ERROR   35
3496  013514  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3497  013522  000431                          BR      TST22           ;;GO ON TO NEXT TEST
3498
3499  013524                          2$:
3500  013524  032737  000400  004146          BIT     #S.CLR.T.MR2    ;CHECK IF CLEAR COMMAND
3501                                                                  ;  BIT SET
3502  013532  001002                          BNE     3$              ;YES, CHECK CYLINDER ADDRESS BITS
3503  013534  104036                          ERROR   36              ;S.CLR BIT NOT SET
3504  013536  000423                          BR      TST22           ;;GO ON TO NEXT TEST
3505
3506  013540                          3$:
3507  013540  013737  004150  001160          MOV     T.MR3,$TMPO     ;MASK OUT BITS NOT UNDER TEST
3508  013546  042737  140017  001160          BIC     #140017,$TMPO
3509  013554  001402                          BEQ     4$              ;CHECK IF CYLINDER ADDRESS ZERO
3510  013556  104037                          ERROR   37              ;CYLINDER ADDRESS BITS INCORRECT
3511  013560  000412                          BR      TST22           ;;GO ON TO NEXT TEST
3512
3513  013562  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3514  013570  001401                          BEQ     5$              ;YES, CHECK MESSAGE B
3515  013572  104040                          ERROR   40              ;MESS A INCORRECT
3516  013574  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3517  013602  001401                          BEQ     TST22           ;;YES, GO ON TO NEXT TEST
3518  013604  104041                          ERROR   41              ;MESS B INCORRECT
3519
3520                                  ;;*****************************************************************
3521                                  ;*TEST 22        CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 4)
3522                                  ;*
3523                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3524                                  ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3525                                  ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3526                                  ;*      AND STATUS REGISTER 1 WITH AN UNLOAD.  CLOCK
3527                                  ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3528                                  ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3529                                  ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3530                                  ;*
3531                                  ;*****************************************************************
3532  013606  000004                  TST22:  SCOPE
3533  013610  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3534  013616  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3535  013622  012737  001777  004252          MOV     #1777,CYLIN     ;LOAD CYLINDER VALUE
3536  013630  012737  000052  004254          MOV     #52,OFFVAL      ;LOAD OFFSET VALUE
3537  013636  012737  000007  004160          MOV     #UNLOAD,E.CS1   ;LOAD EXPECTED CS1
3538  013644  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3539  013652  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
3540  013660  012762  001777  000020          MOV     #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3541  013666  012762  000052  000016          MOV     #52,RKASOF(R2) ;LOAD OFFSET VALUE
3542  013674  012762  000007  000000          MOV     #UNLOAD,RKCS1(R2) ;ISSUE UNLOAD
3543  013702  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3544  013706  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
3545  013714  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3546  013722  005300                          DEC     R0
```

```
3547  013724  001370                        BNE     1$
3548  013726  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3549  013734  016237  000034  004146        MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3550  013742  016237  000036  004150        MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3551  013750  012737  002000  004206        MOV     #S.UNLD,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3552  013756  005037  004210                CLR     E.MR3           ;LOAD EXPECTED MAINTENANCE REG. 3
3553  013762  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3554  013770  001405                        BEQ     2$              ;YES, CHECK MESSAGES A&B
3555  013772  104035                        ERROR   35
3556  013774  012762  100000  000000        MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3557  014002  000431                        BR      TST23           ;;GO ON TO NEXT TEST
3558
3559  014004                          2$:
3560  014004  032737  002000  004146        BIT     #S.UNLD,T.MR2   ;CHECK IF UNLOAD COMMAND
3561                                                                ;   BIT SET
3562  014012  001002                        BNE     3$              ;YES, CHECK CYLINDER ADDRESS BITS
3563  014014  104036                        ERROR   36              ;S.UNLD BIT NOT SET
3564  014016  000423                        BR      TST23           ;;GO ON TO NEXT TEST
3565
3566  014020                          3$:
3567  014020  013737  004150  001160        MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
3568  014026  042737  140017  001160        BIC     #140017,$TMP0
3569  014034  001402                        BEQ     4$              ;CHECK IF CYLINDER ADDRESS ZERO
3570  014036  104037                        ERROR   37              ;CYLINDER ADDRESS BITS INCORRECT
3571  014040  000412                        BR      TST23           ;;GO ON TO NEXT TEST
3572
3573  014042  023737  004206  004146  4$:   CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3574  014050  001401                        BEQ     5$              ;YES, CHECK MESSAGE B
3575  014052  104040                        ERROR   40              ;MESS A INCORRECT
3576  014054  023737  004210  004150  5$:   CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3577  014062  001401                        BEQ     TST23           ;;YES, GO ON TO NEXT TEST
3578  014064  104041                        ERROR   41              ;MESS B INCORRECT
3579
3580                          ;*****************************************************************
3581                          ;TEST 23        CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 5)
3582                          ;*
3583                          ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3584                          ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3585                          ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3586                          ;*      AND STATUS REGISTER 1 WITH A START SPINDLE.  CLOCK
3587                          ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3588                          ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3589                          ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3590                          ;*
3591                          ;*****************************************************************
3592  014066  000004          TST23:  SCOPE
3593  014070  012737  000144  001200        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3594  014076  013702  001270                MOV     $BASE,R2        ;LOAD RK611 BASE
3595  014102  012737  001777  004252        MOV     #1777,CYLIN     ;LOAD CYLINDER VALUE
3596  014110  012737  000052  004254        MOV     #52,OFFVAL      ;LOAD OFFSET VALUE
3597  014116  012737  000011  004160        MOV     #SRTSPL,E.CS1   ;LOAD EXPECTED CS1
3598  014124  012762  100000  000000        MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3599  014132  012762  000040  000026        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
3600  014140  012762  001777  000020        MOV     #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3601  014146  012762  000052  000016        MOV     #52,RKASOF(R2) ;LOAD OFFSET VALUE
3602  014154  012762  000011  000000        MOV     #SRTSPL,RKCS1(R2) ;ISSUE SRTSPL
```

```
3603  014162  012700  000016              MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3604  014166  012762  000440  000026  1$: MOV     #DMD!MCLK,RKMR1(R2)
3605  014174  012762  000040  000026      MOV     #DMD,RKMR1(R2)
3606  014202  005300                      DEC     R0
3607  014204  001370                      BNE     1$
3608  014206  016237  000000  004120      MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3609  014214  016237  000034  004146      MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3610  014222  016237  000036  004150      MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3611  014230  012737  000100  004206      MOV     #S.STSP,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3612  014236  005037  004210              CLR     E.MR3           ;LOAD EXPECTED MAINTENANCE REG. 3
3613  014242  023737  004160  004120      CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3614  014250  001405                      BEQ     2$              ;YES, CHECK MESSAGES A&B
3615  014252  104035                      ERROR   35
3616  014254  012762  100000  000000      MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3617  014262  000431                      BR      TST24           ;;GO ON TO NEXT TEST
3618
3619  014264                          2$: 
3620  014264  032737  000100  004146      BIT     #S.STSP,T.MR2   ;CHECK IF SRTSPL COMMAND
3621                                                              ;   BIT SET
3622  014272  001002                      BNE     3$              ;YES, CHECK CYLINDER ADDRESS BITS
3623  014274  104036                      ERROR   36              ;S.STSP BIT NOT SET
3624  014276  000423                      BR      TST24           ;;GO ON TO NEXT TEST
3625
3626  014300                          3$: 
3627  014300  013737  004150  001160      MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
3628  014306  042737  140017  001160      BIC     #140017,$TMP0
3629  014314  001402                      BEQ     4$              ;CHECK IF CYLINDER ADDRESS ZERO
3630  014316  104037                      ERROR   37              ;CYLINDER ADDRESS BITS INCORRECT
3631  014320  000412                      BR      TST24           ;;GO ON TO NEXT TEST
3632
3633  014322  023737  004206  004146  4$: CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3634  014330  001401                      BEQ     5$              ;YES, CHECK MESSAGE B
3635  014332  104040                      ERROR   40              ;MESS A INCORRECT
3636  014334  023737  004210  004150  5$: CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3637  014342  001401                      BEQ     TST24           ;;YES, GO ON TO NEXT TEST
3638  014344  104041                      ERROR   41              ;MESS B INCORRECT
3639
3640                                  ;;*****************************************************************
3641                                  ;*TEST 24       CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 6)
3642                                  ;*
3643                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3644                                  ;*      DIAGNOSTIC MODE.  LOAD CYLINDER ADDRESS REGISTER
3645                                  ;*      WITH 777.  LOAD THE OFFSET REG TO 52.  LOAD COMMAND
3646                                  ;*      AND STATUS REGISTER 1 WITH A RECALIBRATE.  CLOCK
3647                                  ;*      MESSAGES A AND B INTO SHIFT REGISTERS.  MAKE SURE
3648                                  ;*      SHIFT REGISTERS ARE LOADED CORRECTLY AND THE CYLINDER
3649                                  ;*      ADDRESS FIELD IS ZERO IN DRIVE MESSAGE.
3650                                  ;*
3651                                  ;*****************************************************************
3652  014346  000004                  TST24:  SCOPE
3653  014350  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3654  014356  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3655  014362  012737  001777  004252          MOV     #1777,CYLIN     ;LOAD CYLINDER VALUE
3656  014370  012737  000052  004254          MOV     #52,OFFVAL      ;LOAD OFFSET VALUE
3657  014376  012737  000013  004160          MOV     #RECAL,E.CS1    ;LOAD EXPECTED CS1
3658  014404  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
```

```
3659  014412  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
3660  014420  012762  001777  000020          MOV     #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3661  014426  012762  000052  000016          MOV     #52,RKASOF(R2) ;LOAD OFFSET VALUE
3662  014434  012762  000013  000000          MOV     #RECAL,RKCS1(R2) ;ISSUE RECAL
3663  014442  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3664  014446  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
3665  014454  012762  000040  000026          MOV     #DMD,RKMR1(R2)
3666  014462  005300                          DEC     R0
3667  014464  001370                          BNE     1$
3668  014466  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3669  014474  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3670  014502  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3671  014510  012737  000040  004206          MOV     #S.RECL,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3672  014516  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINTENANCE REG. 3
3673  014522  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3674  014530  001405                          BEQ     2$              ;YES, CHECK MESSAGES A&B
3675  014532  104035                          ERROR   35
3676  014534  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR CONTROLLER FOR NEXT TEST
3677  014542  000431                          BR      TST25           ;;GO ON TO NEXT TEST
3678
3679  014544                          2$:
3680  014544  032737  000040  004146          BIT     #S.RECL,T.MR2   ;CHECK IF RECAL COMMAND
3681                                                                  ;  BIT SET
3682  014552  001002                          BNE     3$              ;YES, CHECK CYLINDER ADDRESS BITS
3683  014554  104036                          ERROR   36               ;S.RECL BIT NOT SET
3684  014556  000423                          BR      TST25           ;;GO ON TO NEXT TEST
3685
3686  014560                          3$:
3687  014560  013737  004150  001160          MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
3688  014566  042737  140017  001160          BIC     #140017,$TMP0
3689  014574  001402                          BEQ     4$              ;CHECK IF CYLINDER ADDRESS ZERO
3690  014576  104037                          ERROR   37              ;CYLINDER ADDRESS BITS INCORRECT
3691  014600  000412                          BR      TST25           ;;GO ON TO NEXT TEST
3692
3693  014602  023737  004206  004146  4$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3694  014610  001401                          BEQ     5$              ;YES, CHECK MESSAGE B
3695  014612  104040                          ERROR   40              ;MESS A INCORRECT
3696  014614  023737  004210  004150  5$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3697  014622  001401                          BEQ     TST25           ;;YES, GO ON TO NEXT TEST
3698  014624  104041                          ERROR   41              ;MESS B INCORRECT
3699
3700                                  ;;*****************************************************************
3701                                  ;*TEST 25        MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 1)
3702                                  ;*
3703                                  ;*       CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3704                                  ;*       DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
3705                                  ;*       COMMAND AND STATUS REGISTER 1 WITH A PACK ACKNOWLEDGE.
3706                                  ;*       CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
3707                                  ;*       MESSAGE SELECT BITS ARE CLEARED.
3708                                  ;*
3709                                  ;;*****************************************************************
3710  014626  000004                  TST25:  SCOPE
3711  014630  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3712  014636  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3713  014642  012737  000017  004246          MOV     #17,MSGCOD      ;LOAD MESSAGE CODE FOR PRINT OUT
3714  014650  012737  000003  004160          MOV     #PACK,E.CS1     ;LOAD EXPECTED CS1
```

```
3715  014656  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3716  014664  012762  000057  000026          MOV     #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3717                                                                    ;  SELECT MESSAGE 17
3718  014672  012762  000003  000000          MOV     #PACK,RKCS1(R2) ;ISSUE PACK
3719  014700  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3720  014704  052762  000400  000026  1$:     BIS     #MCLK,RKMR1(R2)
3721  014712  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
3722  014720  005300                          DEC     R0
3723  014722  001370                          BNE     1$
3724  014724  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3725  014732  016237  000026  004144          MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
3726  014740  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
3727  014746  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
3728  014754  012737  002040  004204          MOV     #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
3729  014762  032737  020000  004144          BIT     #ECCW,T.MR1
3730  014770  001403                          BEQ     10$
3731  014772  052737  020000  004204          BIS     #ECCW,E.MR1
3732  015000  012737  004000  004206  10$:     MOV     #S.PACK,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3733  015006  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
3734  015012  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3735  015020  001405                          BEQ     2$              ;YES, CHECK MAINT REG. 1
3736  015022  104042                          ERROR   42
3737  015024  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
3738  015032  000442                          BR      TST26           ;;GO ON TO NEXT TEST
3739
3740  015034  023737  004204  004144  2$:     CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
3741  015042  001405                          BEQ     3$              ;YES, CHECK MESSAGES A&B
3742  015044  104043                          ERROR   43              ;MAINT REG. 1 INCORRECT
3743  015046  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
3744  015054  000431                          BR      TST26           ;;GO ON TO NEXT TEST
3745
3746  015056                          3$:
3747  015056  032737  004000  004146          BIT     #S.PACK,T.MR2   ;CHECK IF PACK COMMAND
3748                                                                  ;   BIT SET
3749  015064  001002                          BNE     4$              ;YES, CHECK MESSAGE SELECT BITS
3750  015066  104044                          ERROR   44              ;S.PACK BIT NOT SET
3751  015070  000423                          BR      TST26           ;;GO ON TO NEXT TEST
3752
3753  015072                          4$:
3754  015072  013737  004150  001160          MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
3755  015100  042737  177760  001160          BIC     #177760,$TMP0
3756  015106  001402                          BEQ     5$              ;CHECK IF MESSAGE SELECT ZERO
3757  015110  104045                          ERROR   45              ;MESSAGE SELECT BITS NOT ZERO
3758  015112  000412                          BR      TST26           ;;GO ON TO NEXT TEST
3759
3760  015114  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3761  015122  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
3762  015124  104046                          ERROR   46              ;MESSAGE A INCORRECT
3763  015126  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3764  015134  001401                          BEQ     TST26           ;;YES, GO ON TO NEXT TEST
3765  015136  104047                          ERROR   47              ;MESS B INCORRECT
3766
3767                                  ;;*****************************************************************
3768                                  ;*TEST 26        MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 2)
3769                                  ;*
3770                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
```

```
                                        ;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
                                        ;*      COMMAND AND STATUS REGISTER 1 WITH A DRIVE CLEAR.
                                        ;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
                                        ;*      MESSAGE SELECT BITS ARE CLEARED.
                                        ;*
                                        ;;*****************************************************************
015140  000004                          TST26:  SCOPE
015142  012737  000144  001200                  MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
015150  013702  001270                          MOV     $BASE,R2        ;LOAD RK611 BASE
015154  012737  000017  004246                  MOV     #17,MSGCOD      ;LOAD MESSAGE CODE FOR PRINT OUT
015162  012737  000005  004160                  MOV     #CLEAR,E.CS1    ;LOAD EXPECTED CS1
015170  012762  100000  000000                  MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
015176  012762  000057  000026                  MOV     #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
                                                                        ;  SELECT MESSAGE 17
015204  012762  000005  000000                  MOV     #CLEAR,RKCS1(R2)        ;ISSUE CLEAR
015212  012700  000016                          MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
015216  052762  000400  000026  1$:             BIS     #MCLK,RKMR1(R2)
015224  042762  000400  000026                  BIC     #MCLK,RKMR1(R2)
015232  005300                                  DEC     R0
015234  001370                                  BNE     1$
015236  016237  000000  004120                  MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
015244  016237  000026  004144                  MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
015252  016237  000034  004146                  MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
015260  016237  000036  004150                  MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
015266  012737  002040  004204                  MOV     #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
015274  032737  020000  004144                  BIT     #ECCW,T.MR1
015302  001403                                  BEQ     10$
015304  052737  020000  004204                  BIS     #ECCW,E.MR1
015312  012737  000400  004206  10$:            MOV     #S.CLR,E.MR2    ;LOAD EXPECTED MAINT REG. 2
015320  005037  004210                          CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
015324  023737  004160  004120                  CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
015332  001405                                  BEQ     2$              ;YES, CHECK MAINT REG. 1
015334  104042                                  ERROR   42
015336  012762  100000  000000                  MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
015344  000442                                  BR      TST27           ;;GO ON TO NEXT TEST
015346  023737  004204  004144  2$:             CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
015354  001405                                  BEQ     3$              ;YES, CHECK MESSAGES A&B
015356  104043                                  ERROR   43              ;MAINT REG. 1 INCORRECT
015360  012762  100000  000000                  MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
015366  000431                                  BR      TST27           ;;GO ON TO NEXT TEST
015370                                  3$:
015370  032737  000400  004146                  BIT     #S.CLR,T.MR2    ;CHECK IF CLEAR COMMAND
                                                                        ;  BIT SET
015376  001002                                  BNE     4$              ;YES, CHECK MESSAGE SELECT BITS
015400  104044                                  ERROR   44              ;S.CLR BIT NOT SET
015402  000423                                  BR      TST27           ;;GO ON TO NEXT TEST
015404                                  4$:
015404  013737  004150  001160                  MOV     T.MR3,$TMPO     ;MASK OUT BITS NOT UNDER TEST
015412  042737  177760  001160                  BIC     #177760,$TMPO
015420  001402                                  BEQ     5$              ;CHECK IF MESSAGE SELECT ZERO
015422  104045                                  ERROR   45              ;MESSAGE SELECT BITS NOT ZERO
015424  000412                                  BR      TST27           ;;GO ON TO NEXT TEST
```

```
3827  015426  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3828  015434  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
3829  015436  104046                          ERROR   46              ;MESSAGE A INCORRECT
3830  015440  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3831  015446  001401                          BEQ     TST27           ;;YES, GO ON TO NEXT TEST
3832  015450  104047                          ERROR   47              ;MESS B INCORRECT
3833
3834                                  ;;*****************************************************************
3835                                  ;*TEST 27        MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 3)
3836                                  ;*
3837                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3838                                  ;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
3839                                  ;*      COMMAND AND STATUS REGISTER 1 WITH AN UNLOAD.
3840                                  ;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
3841                                  ;*      MESSAGE SELECT BITS ARE CLEARED.
3842                                  ;*
3843                                  ;*****************************************************************
3844  015452  000004                  TST27:  SCOPE
3845  015454  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3846  015462  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3847  015466  012737  000017  004246          MOV     #17,MSGCOD      ;LOAD MESSAGE CODE FOR PRINT OUT
3848  015474  012737  000007  004160          MOV     #UNLOAD,E.CS1   ;LOAD EXPECTED CS1
3849  015502  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3850  015510  012762  000057  000026          MOV     #DMD!17,RKMR1(R2) ;PUT Rk611 IN MAINTENANCE MODE
3851                                                                  ;  SELECT MESSAGE 17
3852  015516  012762  000007  000000          MOV     #UNLOAD,RKCS1(R2)       ;ISSUE UNLOAD
3853  015524  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3854  015530  052762  000400  000026  1$:     BIS     #MCLK,RKMR1(R2)
3855  015536  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
3856  015544  005300                          DEC     R0
3857  015546  001370                          BNE     1$
3858  015550  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3859  015556  016237  000026  004144          MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
3860  015564  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
3861  015572  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
3862  015600  012737  002040  004204          MOV     #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
3863  015606  032737  020000  004144          BIT     #ECCW,T.MR1
3864  015614  001403                          BEQ     10$
3865  015616  052737  020000  004204          BIS     #ECCW,E.MR1
3866  015624  012737  002000  004206  10$:    MOV     #S.UNLD,E.MR2   ;LOAD EXPECTED MAINT REG. 2
3867  015632  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
3868  015636  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
3869  015644  001405                          BEQ     2$              ;YES, CHECK MAINT REG. 1
3870  015646  104042                          ERROR   42
3871  015650  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
3872  015656  000442                          BR      TST30           ;;GO ON TO NEXT TEST
3873
3874  015660  023737  004204  004144  2$:     CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
3875  015666  001405                          BEQ     3$              ;YES, CHECK MESSAGES A&B
3876  015670  104043                          ERROR   43              ;MAINT REG. 1 INCORRECT
3877  015672  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
3878  015700  000431                          BR      TST30           ;;GO ON TO NEXT TEST
3879
3880  015702                          3$:
3881  015702  032737  002000  004146          BIT     #S.UNLD,T.MR2   ;CHECK IF UNLOAD COMMAND
3882                                                                  ;  BIT SET
```

```
                                        BNE     4$                  ;YES, CHECK MESSAGE SELECT BITS
                                        ERROR   44                  ;S.UNLD BIT NOT SET
                                        BR      TST30               ;;GO ON TO NEXT TEST

                                4$:
                                        MOV     T.MR3,$TMP0         ;MASK OUT BITS NOT UNDER TEST
                                        BIC     #177760,$TMP0
                                        BEQ     5$                  ;CHECK IF MESSAGE SELECT ZERO
                                        ERROR   45                  ;MESSAGE SELECT BITS NOT ZERO
                                        BR      TST30               ;;GO ON TO NEXT TEST

                                5$:     CMP     E.MR2,T.MR2         ;CHECK IF MESSAGE A CORRECT
                                        BEQ     6$                  ;YES, CHECK MESSAGE B
                                        ERROR   46                  ;MESSAGE A INCORRECT
                                6$:     CMP     E.MR3,T.MR3         ;CHECK IF MESSAGE B CORRECT
                                        BEQ     TST30               ;;YES, GO ON TO NEXT TEST
                                        ERROR   47                  ;MESS B INCORRECT

;;*****************************************************************
;*TEST 30        MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 4)
;*
;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
;*      COMMAND AND STATUS REGISTER 1 WITH A START SPINDLE.
;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
;*      MESSAGE SELECT BITS ARE CLEARED.
;*
;;*****************************************************************
TST30:  SCOPE
                                        MOV     #100.,$TIMES        ;;DO 100. ITERATIONS
                                        MOV     $BASE,R2            ;LOAD RK611 BASE
                                        MOV     #17,MSGCOD          ;LOAD MESSAGE CODE FOR PRINT OUT
                                        MOV     #SRTSPL,E.CS1       ;LOAD EXPECTED CS1
                                        MOV     #CCLR,RKCS1(R2)     ;CLEAR RK611
                                        MOV     #DMD!17,RKMR1(R2)   ;PUT RK611 IN MAINTENANCE MODE
                                                                    ;   SELECT MESSAGE 17
                                        MOV     #SRTSPL,RKCS1(R2)       ;ISSUE SRTSPL
                                        MOV     #3*4+2,R0           ;CLOCK IN DRIVE MESSAGE
                                1$:     BIS     #MCLK,RKMR1(R2)
                                        BIC     #MCLK,RKMR1(R2)
                                        DEC     R0
                                        BNE     1$
                                        MOV     RKCS1(R2),T.CS1     ;STORE COMMAND AND STATUS REG. 1
                                        MOV     RKMR1(R2),T.MR1     ;STORE MAINTENANCE REG. 1
                                        MOV     RKMR2(R2),T.MR2     ;STORE MAINTENANCE REG.2
                                        MOV     RKMR3(R2),T.MR3     ;STORE MAINTENANCE REG. 3
                                        MOV     #MEWD!DMD,E.MR1     ;LOAD EXPECTED MAINT REG. 1
                                        BIT     #ECCW,T.MR1
                                        BEQ     10$
                                        BIS     #ECCW,E.MR1
                                10$:    MOV     #S.STSP,E.MR2       ;LOAD EXPECTED MAINT REG. 2
                                        CLR     E.MR3               ;LOAD EXPECTED MAINT REG. 3
                                        CMP     E.CS1,T.CS1         ;CHECK IF CS1 CORRECT
                                        BEQ     2$                  ;YES, CHECK MAINT REG. 1
                                        ERROR   42
                                        MOV     #CCLR,RKCS1(R2)     ;CLEAR RK611 CONTROLLER FOR NEXT TEST
```

```
3939  016170  000442                          BR      TST31           ;;GO ON TO NEXT TEST
3940
3941  016172  023737  004204  004144  2$:     CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
3942  016200  001405                          BEQ     3$              ;YES, CHECK MESSAGES A&B
3943  016202  104043                          ERROR   43              ;MAINT REG. 1 INCORRECT
3944  016204  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
3945  016212  000431                          BR      TST31           ;;GO ON TO NEXT TEST
3946
3947  016214                          3$:
3948  016214  032737  000100  004146          BIT     #S.STSP,T.MR2   ;CHECK IF SRTSPL COMMAND
3949                                                                  ;   BIT SET
3950  016222  001002                          BNE     4$              ;YES, CHECK MESSAGE SELECT BITS
3951  016224  104044                          ERROR   44              ;S.STSP BIT NOT SET
3952  016226  000423                          BR      TST31           ;;GO ON TO NEXT TEST
3953
3954  016230                          4$:
3955  016230  013737  004150  001160          MOV     T.MR3,$TMPO     ;MASK OUT BITS NOT UNDER TEST
3956  016236  042737  177760  001160          BIC     #177760,$TMPO
 957  016244  001402                          BEQ     5$              ;CHECK IF MESSAGE SELECT ZERO
 958  016246  104045                          ERROR   45              ;MESSAGE SELECT BITS NOT ZERO
3959  016250  000412                          BR      TST31           ;;GO ON TO NEXT TEST
3960
3961  016252  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
3962  016260  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
3963  016262  104046                          ERROR   46              ;MESSAGE A INCORRECT
3964  016264  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
3965  016272  001401                          BEQ     TST31           ;;YES, GO ON TO NEXT TEST
3966  016274  104047                          ERROR   47              ;MESS B INCORRECT
3967
3968                                  ;;*****************************************************************
3969                                  ;*TEST 31       MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 5)
3970                                  ;*
3971                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
3972                                  ;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
3973                                  ;*      COMMAND AND STATUS REGISTER 1 WITH A RECALIBRATE.
3974                                  ;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
3975                                  ;*      MESSAGE SELECT BITS ARE CLEARED.
3976                                  ;*
3977                                  ;;*****************************************************************
3978  016276  000004                  TST31:  SCOPE
3979  016300  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
3980  016306  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
3981  016312  012737  000017  004246          MOV     #17,MSGCOD      ;LOAD MESSAGE CODE FOR PRINT OUT
3982  016320  012737  000013  004160          MOV     #RECAL,E.CS1    ;LOAD EXPECTED CS1
3983  016326  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
3984  016334  012762  000057  000026          MOV     #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3985                                                                  ;   SELECT MESSAGE 17
3986  016342  012762  000013  000000          MOV     #RECAL,RKCS1(R2)        ;ISSUE RECAL
3987  016350  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
3988  016354  052762  000400  000026  1$:     BIS     #MCLK,RKMR1(R2)
3989  016362  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
3990  016370  005300                          DEC     R0
3991  016372  001370                          BNE     1$
3992  016374  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3993  016402  016237  000026  004144          MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
3994  016410  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
```

```
 3995  016416  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
 3996  016424  012737  002040  004204          MOV     #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
 3997  016432  032737  020000  004144          BIT     #ECCW,T.MR1
 3998  016440  001403                          BEQ     10$
 3999  016442  052737  020000  004204          BIS     #ECCW,E.MR1
 4000  016450  012737  000040  004206  10$:    MOV     #S.RECL,E.MR2   ;LOAD EXPECTED MAINT REG. 2
 4001  016456  005037  004210                  CLR     E.MR3           ;LOAD EXPECTED MAINT REG. 3
 4002  016462  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
 4003  016470  001405                          BEQ     2$              ;YES, CHECK MAINT REG. 1
 4004  016472  104042                          ERROR   42
 4005  016474  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
 4006  016502  000442                          BR      TST32           ;;GO ON TO NEXT TEST
 4007
 4008  016504  023737  004204  004144  2$:     CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
 4009  016512  001405                          BEQ     3$              ;YES, CHECK MESSAGES A&B
 4010  016514  104043                          ERROR   43              ;MAINT REG. 1 INCORRECT
 4011  016516  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
 4012  016524  000431                          BR      TST32           ;;GO ON TO NEXT TEST
 4013
 4014  016526                          3$:
 4015  016526  032737  000040  004146          BIT     #S.RECL,T.MR2   ;CHECK IF RECAL COMMAND
 4016                                                                  ;   BIT SET
 4017  016534  001002                          BNE     4$              ;YES, CHECK MESSAGE SELECT BITS
 4018  016536  104044                          ERROR   44              ;S.RECL BIT NOT SET
 4019  016540  000423                          BR      TST32           ;;GO ON TO NEXT TEST
 4020
 4021  016542                          4$:
 4022  016542  013737  004150  001160          MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
 4023  016550  042737  177760  001160          BIC     #177760,$TMP0
 4024  016556  001402                          BEQ     5$              ;CHECK IF MESSAGE SELECT ZERO
 4025  016560  104045                          ERROR   45              ;MESSAGE SELECT BITS NOT ZERO
 4026  016562  000412                          BR      TST32           ;;GO ON TO NEXT TEST
 4027
 4028  016564  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
 4029  016572  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
 4030  016574  104046                          ERROR   46              ;MESSAGE A INCORRECT
 4031  016576  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
 4032  016604  001401                          BEQ     TST32           ;;YES, GO ON TO NEXT TEST
 4033  016606  104047                          ERROR   47              ;MESS'B INCORRECT
 4034
 4035                                  ;;*****************************************************************
 4036                                  ;*TEST 32        MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 6)
 4037                                  ;*
 4038                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
 4039                                  ;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
 4040                                  ;*      COMMAND AND STATUS REGISTER 1 WITH A OFFSET.
 4041                                  ;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
 4042                                  ;*      MESSAGE SELECT BITS ARE CLEARED.
 4043                                  ;*
 4044                                  ;*****************************************************************
 4045  016610  000004                  TST32.  SCOPE
 4046  016612  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 4047  016620  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 4048  016624  012737  000017  004246          MOV     #17,MSGCOD      ;LOAD MESSAGE CODE FOR PRINT OUT
 4049  016632  012737  000015  004160          MOV     #OFFSET,E.CS1   ;LOAD EXPECTED CS1
 4050  016640  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
```

```
4051  016646  012762  000057  000026          MOV     #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
4052                                                                    ;  SELECT MESSAGE 17
4053  016654  012762  000015  000000          MOV     #OFFSET,RKCS1(R2)       ;ISSUE OFFSET
4054  016662  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN DRIVE MESSAGE
4055  016666  052762  000400  000026  1$:     BIS     #MCLK,RKMR1(R2)
4056  016674  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
4057  016702  005300                          DEC     R0
4058  016704  001370                          BNE     1$
4059  016706  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4060  016714  016237  000026  004144          MOV     RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
4061  016722  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
4062  016730  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
4063  016736  012737  002040  004204          MOV     #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
4064  016744  032737  020000  004144          BIT     #ECCW,T.MR1
4065  016752  001403                          BEQ     10$
4066  016754  052737  020000  004204          BIS     #ECCW,E.MR1
4067  016762  005037  004206          10$:    CLR     E.MR2           ;LOAD EXPECTED MAINT REG 2
4068  016766  012737  017760  004210          MOV     #17760,E.MR3    ;LOAD EXPECTED MAINT REG 3
4069  016774  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
4070  017002  001405                          BEQ     2$              ;YES, CHECK MAINT REG. 1
4071  017004  104042                          ERROR   42
4072  017006  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
4073  017014  000434                          BR      TST33           ;;GO ON TO NEXT TEST
4074
4075  017016  023737  004204  004144  2$:     CMP     E.MR1,T.MR1     ;CHECK IF MAINT REG. 1 CORRECT
4076  017024  001405                          BEQ     3$              ;YES, CHECK MESSAGES A&B
4077  017026  104043                          ERROR   43              ;MAINT REG. 1 INCORRECT
4078  017030  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST
4079  017036  000423                          BR      TST33           ;;GO ON TO NEXT TEST
4080
4081  017040                          3$:
4082  017040  013737  004150  001160          MOV     T.MR3,$TMP0     ;MASK OUT BITS NOT UNDER TEST
4083  017046  042737  177760  001160          BIC     #177760,$TMP0
4084  017054  001402                          BEQ     5$              ;CHECK IF MESSAGE SELECT ZERO
4085  017056  104045                          ERROR   45              ;MESSAGE SELECT BITS NOT ZERO
4086  017060  000412                          BR      TST33           ;;GO ON TO NEXT TEST
4087
4088  017062  023737  004206  004146  5$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE A CORRECT
4089  017070  001401                          BEQ     6$              ;YES, CHECK MESSAGE B
4090  017072  104046                          ERROR   46              ;MESSAGE A INCORRECT
4091  017074  023737  004210  004150  6$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE B CORRECT
4092  017102  001401                          BEQ     TST33           ;;YES, GO ON TO NEXT TEST
4093  017104  104047                          ERROR   47              ;MESS'B INCORRECT
4094
4095                                  ;;*****************************************************************
4096                                  ;*TEST 33       MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 7)
4097                                  ;*
4098                                  ;*      CLEAR RK611 WITH CONTROLLER CLEAR.  PUT CONTROLLER IN
4099                                  ;*      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17.  LOAD
4100                                  ;*      COMMAND AND STATUS REGISTER 1 WITH A SEEK.
4101                                  ;*      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME.  MAKE SURE
4102                                  ;*      MESSAGE SELECT BITS ARE CLEARED.
4103                                  ;*
4104                                  ;*****************************************************************
4105  017106  000004                  TST33:  SCOPE
4106  017110  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
```

```
 4107  017116  013702  001270                    MOV     $BASE,R2          ;LOAD RK611 BASE
 4108  017122  012737  000017  004246            MOV     #17,MSGCOD        ;LOAD MESSAGE CODE FOR PRINT OUT
 4109  017130  012737  000017  004160            MOV     #SEEK,E.CS1       ;LOAD EXPECTED CS1
 4110  017136  012762  100000  000000            MOV     #CCLR,RKCS1(R2)   ;CLEAR RK611
 4111  017144  012762  000057  000026            MOV     #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
 4112                                                                      ;  SELECT MESSAGE 17
 4113  017152  012762  000017  000000            MOV     #SEEK,RKCS1(R2)   ;ISSUE SEEK
 4114  017160  012700  000016                    MOV     #3*4+2,R0         ;CLOCK IN DRIVE MESSAGE
 4115  017164  052762  000400  000026  1$:       BIS     #MCLK,RKMR1(R2)
 4116  017172  042762  000400  000026            BIC     #MCLK,RKMR1(R2)
 4117  017200  005300                            DEC     R0
 4118  017202  001370                            BNE     1$
 4119  017204  016237  000000  004120            MOV     RKCS1(R2),T.CS1   ;STORE COMMAND AND STATUS REG. 1
 4120  017212  016237  000026  004144            MOV     RKMR1(R2),T.MR1   ;STORE MAINTENANCE REG. 1
 4121  017220  016237  000034  004146            MOV     RKMR2(R2),T.MR2   ;STORE MAINTENANCE REG.2
 4122  017226  016237  000036  004150            MOV     RKMR3(R2),T.MR3   ;STORE MAINTENANCE REG. 3
 4123  017234  012737  002040  004204            MOV     #MEWD!DMD,E.MR1   ;LOAD EXPECTED MAINT REG. 1
 4124  017242  032737  020000  004144            BIT     #ECCW,T.MR1
 4125  017250  001403                            BEQ     10$
 4126  017252  052737  020000  004204            BIS     #ECCW,E.MR1
 4127  017260  012737  000020  004206  10$:      MOV     #S.SEEK,E.MR2     ;LOAD EXPECTED MAINT REG. 2
 4128  017266  005037  004210                    CLR     E.MR3             ;LOAD EXPECTED MAINT REG. 3
 4129  017272  023737  004160  004120            CMP     E.CS1,T.CS1       ;CHECK IF CS1 CORRECT
 4130  017300  001405                            BEQ     2$                ;YES, CHECK MAINT REG. 1
 4131  017302  104042                            ERROR   42
 4132  017304  012762  100000  000000            MOV     #CCLR,RKCS1(R2)   ;CLEAR RK611 CONTROLLER FOR NEXT TEST
 4133  017312  000442                            BR      TST34             ;;GO ON TO NEXT TEST
 4134
 4135  017314  023737  004204  004144  2$:       CMP     E.MR1,T.MR1       ;CHECK IF MAINT REG. 1 CORRECT
 4136  017322  001405                            BEQ     3$                ;YES, CHECK MESSAGES A&B
 4137  017324  104043                            ERROR   43                ;MAINT REG. 1 INCORRECT
 4138  017326  012762  100000  000000            MOV     #CCLR,RKCS1(R2)   ;CLEAR RK611 CONTROLLER FOR NEXT TEST
 4139  017334  000431                            BR      TST34             ;;GO ON TO NEXT TEST
 4140
 4141  017336                            3$:
 4142  017336  032737  000020  004146            BIT     #S.SEEK,T.MR      ;CHECK IF SEEK COMMAND
 4143                                                                      ;  BIT SET
 4144  017344  001002                            BNE     4$                ;YES, CHECK MESSAGE SELECT BITS
 4145  017346  104044                            ERROR   44                ;S.SEEK BIT NOT SET
 4146  017350  000423                            BR      TST34             ;;GO ON TO NEXT TEST
 4147
 4148  017352                            4$:
 4149  017352  013737  004150  001160            MOV     T.MR3,$TMP0       ;MASK OUT BITS NOT UNDER TEST
 4150  017360  042737  177760  001160            BIC     #177760,$TMP0
 4151  017366  001402                            BEQ     5$                ;CHECK IF MESSAGE SELECT ZERO
 4152  017370  104045                            ERROR   45                ;MESSAGE SELECT BITS NOT ZERO
 4153  017372  000412                            BR      TST34             ;;GO ON TO NEXT TEST
 4154
 4155  017374  023737  004206  004146  5$:       CMP     E.MR2,T.MR2       ;CHECK IF MESSAGE A CORRECT
 4156  017402  001401                            BEQ     6$                ;YES, CHECK MESSAGE B
 4157  017404  104046                            ERROR   46                ;MESSAGE A INCORRECT
 4158  017406  023737  004210  004150  6$:       CMP     E.MR3,T.MR3       ;CHECK IF MESSAGE B CORRECT
 4159  017414  001401                            BEQ     TST34             ;;YES, GO ON TO NEXT TEST
 4160  017416  104047                            ERROR   47                ;MESS B INCORRECT
 4161
 4162
```

```
                                .SBTTL  **DRIVE MESSAGE LOOPBACK AND PARITY GENERATION TESTS

                                ;;*****************************************************************
                                ;*TEST 34       DRIVE MESSAGE LOOPBACK
                                ;*
                                ;*      CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER
                                ;*      IN DIAGNOSTIC MODE INDICATING MESSAGE 3. LOAD COMMAND
                                ;*      STATUS REGISTER FOR DRIVE 5. LOAD COMMAND AND STATUS
                                ;*      REGISTER 1 WITH A SELECT COMMAND.  CLOCK 4 BITS
                                ;*      THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY THAT BITS
                                ;*      ARE INDEED LOOPED BACK.
                                ;*
                                ;;*****************************************************************
017420  000004                  TST34:  SCOPE
017422  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
017430  013702  001270          1$:     MOV     $BASE,R2        ;LOAD RK611 BASE
017434  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
017442  005037  004256                  CLR     SFTCNT          ;INITIALIZE SHIFT COUNT
017446  012737  000005  004206          MOV     #5,E.MR2        ;LOAD EXPECTED SHIFT REG. A
017454  012737  000005  004230          MOV     #5,U.MR2        ;LOAD UNSHIFTED SHIFT REG. A
017462  012737  000003  004210          MOV     #3,E.MR3        ;LOAD EXPECTED SHIFT REG.B
017470  012737  000003  004232          MOV     #3,U.MR3        ;LOAD UNSHIFTED SHIFT REG.B
017476  012762  000043  000026          MOV     #DMD!3,RKMR1(R2)        ;PUT RK611 IN MAINT. MODE
                                                                ;  MESSAGE SELECT = 3
017504  012762  000005  000010          MOV     #5,RKCS2(R2)    ;LOAD DRIVE NUMBER = 5
017512  012762  000001  000000          MOV     #SELDRV,RKCS1(R2)       ;ISSUE SELECT DRIVE
017520  012700  000016                  MOV     #3*4+2,R0       ;CLOCK IN MESSAGE
017524  052762  000400  000026  1$:     BIS     #MCLK,RKMR1(R2) ;ISSUE CLOCKS
017532  042762  000400  000026          BIC     #MCLK,RKMR1(R2)
017540  005300                          DEC     R0
017542  001370                          BNE     1$
017544  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE SHIFT REG. A
017552  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE SHIFT REG. B
017560  023737  004206  004146          CMP     E.MR2,T.MR2     ;CHECK SHIFT REG A CORRECT
017566  001402                          BEQ     2$              ;YES, CHECK SHIFT REG. B
017570  104050                          ERROR   50              ;SHIFT REG A INCORRECT
017572  000431                          BR      TST35           ;;GO ON TO NEXT TEST

017574  023737  004210  004150  2$:     CMP     E.MR3,T.MR3     ;CHECK SHIFT REG B CORRECT
017602  001402                          BEQ     3$              ;YES, SHIFT A BIT
017604  104051                          ERROR   51              ;SHIFT REG B INCORRECT
017606  000423                          BR      TST35           ;;GO ON TO NEXT TEST

017610  032737  000001  004210  3$:     BIT     #BIT0,E.MR3     ;CHECK IF SHIFT BIT = 1
017616  001402                          BEQ     4$              ;NO, CLEAR SHIFT BIT
017620  000261                          SEC                     ;SET SHIFT BIT
017622  000401                          BR      5$              ;GENERATE EXPECTED SHIFT
                                                                ;  REGISTERS A & B

017624  000241                  4$:     CLC                     ;CLEAR SHIFT BIT
017626  006037  004206          5$:     ROR     E.MR2           ;GENERATE EXPECTED SHIFT REG A
017632  006037  004210                  ROR     E.MR3           ;GENERATE EXPECTED SHIFT REG B
017636  012700  000004                  MOV     #4,R0           ;LOAD COUNT FOR 1 BIT SHIFT
017642  005237  004256                  INC     SFTCNT          ;INCREMENT SHIFT BIT COUNT
017646  022737  000004  004256          CMP     #4,SFTCNT       ;CHECK IF FINISHED
017654  103323                          BHIS    1$              ;NO, SHIFT IN NEXT BIT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 35       DRIVE MESSAGE SHIFT
                                        ;*
                                        ;*      CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER
                                        ;*      IN DIAGNOSTIC MODE. LOAD CYLINDER ADDRESS WITH 441.
                                        ;*      LOAD HEAD ADDRESS WITH 1. LOAD COMMAND AND STATUS
                                        ;*      REGISTER 1 WITH A SEEK IN 24 SECTOR MODE. CLOCK 8 BITS
                                        ;*      THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY THAT BITS ARE
                                        ;*      SHIFTED PROPERLY.
                                        ;*
                                        ;;*****************************************************************
017656  000004                        TST35:  SCOPE
017660  012737  000144  001200                MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
017666  013702  001270                        MOV     $BASE,R2        ;LOAD RK611 BASE
017672  012762  100000  000000                MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
017700  005037  004256                        CLR     SFTCNT          ;INITIALIZE SHIFT COUNT
017704  012737  011020  004206                MOV     #S.FMT!S.SEEK!BIT12,E.MR2       ;LOAD EXPECTED SHIFT REG. A
017712  012737  011020  004230                MOV     #S.FMT!S.SEEK!BIT12,U.MR2       ;LOAD UNSHIFTED SHIFT REG. A
017720  012737  011020  004210                MOV     #11020,E.MR3    ;LOAD EXPECTED SHIFT REG. B
017726  012737  011020  004232                MOV     #11020,U.MR3    ;LOAD UNSHIFTED SHIFT REG. B
017734  012762  000040  000026                MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT. MODE
017742  012762  000441  000020                MOV     #441,RKDCYL(R2) ;LOAD CYLINDER ADD. REG.
017750  012762  000400  000006                MOV     #400,RKDA(R2)   ;LOAD DISK ADDRESS REG.
017756  012762  010017  000000                MOV     #SEEK!CFMT,RKCS1(R2)    ;ISSUE SEEK

017764  012700  000016                        MOV     #3*4+2,R0       ;CLOCK IN MESSAGE
017770  052762  000400  000026  1$:   BIS     #MCLK,RKMR1(R2) ;ISSUE CLOCKS
017776  042762  000400  000026        BIC     #MCLK,RKMR1(R2)
020004  005300                        DEC     R0
020006  001370                        BNE     1$
020010  016237  000034  004146                MOV     RKMR2(R2),T.MR2 ;STORE SHIFT REG. A
020016  016237  000036  004150                MOV     RKMR3(R2),T.MR3 ;STORE SHIFT REG. B
020024  023737  004206  004146                CMP     E.MR2,T.MR2     ;CHECK SHIFT REG A CORRECT
020032  001402                                BEQ     2$              ;YES, CHECK SHIFT REG. B
020034  104050                                ERROR   50              ;SHIFT REG A INCORRECT
020036  000431                                BR      TST36           ;;GO ON TO NEXT TEST

020040  023737  004210  004150  2$:   CMP     E.MR3,T.MR3     ;CHECK SHIFT REG B CORRECT
020046  001402                        BEQ     3$              ;YES, SHIFT A BIT
020050  104051                        ERROR   51              ;SHIFT REG B INCORRECT
020052  000423                        BR      TST36           ;,GO ON TO NEXT TEST

020054  032737  000001  004210  3$:   BIT     #BIT0,E.MR3     ;CHECK IF SHIFT BIT = 1
020062  001402                        BEQ     4$              ;NO, CLEAR SHIFT BIT
020064  000261                        SEC                     ;SET SHIFT BIT
020066  000401                        BR      5$              ;GENERATE EXPECTED SHIFT
                                                              ;  REGISTERS A & B

020070  000241                4$:     CLC                     ;CLEAR SHIFT BIT
020072  006037  004206        5$:     ROR     E.MR2           ;GENERATE EXPECTED SHIFT REG A
020076  006037  004210                ROR     E.MR3           ;GENERATE EXPECTED SHIFT REG B
020102  012700  000004                MOV     #4,R0           ;LOAD COUNT FOR 1 BIT SHIFT
020106  005237  004256                INC     SFTCNT          ;INCREMENT SHIFT BIT COUNT
020112  022737  000010  004256        CMP     #8.,SFTCNT      ;CHECK IF FINISHED
020120  103323                        BHIS    1$              ;NO, SHIFT IN NEXT BIT
```

```
4275
4276                                    ;;*****************************************************************
4277                                    ;*TEST 36       DRIVE MESSAGE PARITY PRECONDITIONING
4278                                    ;*
4279                                    ;*      CLEAR RK611 WITH CONTROLLER CLEAR, PUT CONTROLLER
4280                                    ;*      IN DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 WITH
4281                                    ;*      A SELECT COMMAND. CLOCK ALL 16 BITS THROUGH THE
4282                                    ;*      DRIVE MESSAGE LOOPBACK. VERIFY PARITY HAS BEEN PRECONDITIONED
4283                                    ;*      PROPERLY. REPEAT FOR BAD PARITY GENERATION.
4284                                    ;*
4285                                    ;*****************************************************************
4286  020122  000004                    T$T36:  SCOPE
4287  020124  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
4288  020132  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
4289  020136  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
4290  020144  012762  000040  000026            MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
4291  020152  012762  000001  000000            MOV     #SELDRV,RKCS1(R2) ;ISSUE SELECT DRIVE
4292  020160  012700  000116                    MOV     #19.*4+2,R0     ;LOAD DRIVE MESSAGE AND SHIFT
4293  020164  012762  000440  000026  1$:       MOV     #DMD!MCLK,RKMR1(R2) ;  ALL 16 BITS
4294  020172  012762  000040  000026            MOV     #DMD,RKMR1(R2)
4295  020200  005300                            DEC     R0
4296  020202  001370                            BNE     1$
4297  020204  016237  000034  004146            MOV     RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4298  020212  016237  000036  004150            MOV     RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4299  020220  012737  100000  004206            MOV     #100000,E.MR2   ;LOAD EXPECTED MESSAGE B
4300  020226  012737  100000  004210            MOV     #100000,E.MR3   ;LOAD EXPECTED MESSAGE A
4301  020234  032737  100000  004150            BIT     #BIT15,T.MR3    ;CHECK IF PARITY ON MESSAGE A CORRECT
4302
4303  020242  001002                            BNE     2$              ;YES, CHECK PARITY ON MESSAGE B
4304  020244  104052                            ERROR   52              ;PARITY ON MESSAGE A INCORRECT
4305  020246  000420                            BR      5$              ;TRY EVEN PARITY
4306
4307  020250  032737  100000  004146  2$:       BIT     #BIT15,T.MR2    ;CHECK IF PARITY ON MESS B CORRECT
4308  020256  001002                            BNE     3$              ;YES, CHECK MESSAGE A AND B
4309  020260  104053                            ERROR   53              ;PARITY ON MESSAGE B INCORRECT
4310  020262  000412                            BR      5$              ;TRY EVEN PARITY
4311
4312  020264  023737  004210  004150  3$:       CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE A CORRECT
4313  020272  001401                            BEQ     4$              ;YES, CHECK MESSAGE B
4314  020274  104054                            ERROR   54              ;MESSAGE A INCORRECT
4315  020276  023737  004206  004146  4$:       CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE B CORRECT
4316  020304  001401                            BEQ     5$              ;YES, TRY EVEN PARITY
4317  020306  104055                            ERROR   55              ;MESSAGE B INCORRECT
4318  020310  012762  100000  000000  5$:       MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
4319  020316  012762  000060  000026            MOV     #DMD!PAT,RKMR1(R2) ;PUT RK611 MAINTENANCE MODE
4320                                            ;   AND EVEN PARITY
4321  020324  012762  000001  000000            MOV     #SELDRV,RKCS1(R2) ;ISSUE SELECT DRIVE
4322  020332  012700  000116                    MOV     #19.*4+2,R0     ;LOAD DRIVE MESSAGE AND SHIFT
4323  020336  012762  000460  000026  6$:       MOV     #DMD!PAT!MCLK,RKMR1(R2) ;  ALL 16 BITS
4324  020344  012762  000060  000026            MOV     #DMD!PAT,RKMR1(R2)
4325  020352  005300                            DEC     R0
4326
4327  020354  001370                            BNE     6$
4328  020356  016237  000034  004146            MOV     RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4329  020364  016237  000036  004150            MOV     RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4330  020372  005037  004206                    CLR     E.MR2           ;LOAD EXPECTED MESSAGE B
```

```
4331  020376  005037  004210                    CLR     E.MR3              ;LOAD EXPECTED MESSAGE A
4332  020402  032737  100000  004150            BIT     #BIT15,T.MR3       ;CHECK IF PARITY ON MESSAGE A CORRECT
4333  020410  001402                            BEQ     7$                 ;YES, CHECK PARITY ON MESSAGE B
4334  020412  104056                            ERROR   56                 ;PARITY ON MESSAGE A INCORRECT
4335  020414  000420                            BR      TST37              ;;GO ON TO NEXT TEST
4336
4337  020416  032737  100000  004146  7$:       BIT     #BIT15,T.MR2       ;CHECK IF PARITY ON MESS B CORRECT
4338  020424  001402                            BEQ     8$                 ;YES, CHECK MESSAGE A AND B
4339  020426  104057                            ERROR   57                 ;PARITY ON MESSAGE B INCORRECT
4340  020430  000412                            BR      TST37              ;;GO ON TO NEXT TEST
4341
4342  020432  023737  004210  004150  8$:       CMP     E.MR3,T.MR3        ;CHECK IF MESSAGE A CORRECT
4343  020440  001401                            BEQ     9$                 ;YES, CHECK MESSAGE B
4344  020442  104060                            ERROR   60                 ;MESSAGE A INCORRECT
4345  020444  023737  004206  004146  9$:       CMP     E.MR2,T.MR2        ;CHECK IF MESSAGE B CORRECT
4346  020452  001401                            BEQ     TST37              ;;YES, GO ON TO NEXT TEST
4347  020454  104061                            ERROR   61                 ;MESSAGE B INCORRECT
4348
4349                                  ;;*****************************************************************
4350                                  ;*TEST 37       ODD DRIVE MESSAGE PARITY GENERATION
4351                                  ;*
4352                                  ;*       CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER
4353                                  ;*       IN DIAGNOSTIC MODE AND MESSAGE SELECT = 1.
4354                                  ;*       LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE
4355                                  ;*       SELECT = 1. LOAD COMMAND AND STATUS REGISTER 1 WITH
4356                                  ;*       A SELECT COMMAND. VERIFY THAT PARITY HAS BEEN
4357                                  ;*       GENERATED CORRECTLY. REPEAT FOR MESSAGE SELECT =
4358                                  ;*       DRIVE SELECT = 2-17.
4359                                  ;*
4360                                  ;*****************************************************************
4361  020456  000004                  TST37:  SCOPE
4362  020460  012737  000144  001200            MOV     #100.,$TIMES       ;;DO 100. ITERATIONS
4363  020466  013702  001270                    MOV     $BASE,R2           ;LOAD RK611 BASE
4364  020472  012737  000001  004244            MOV     #1,DRVCOD          ;LOAD DRIVE CODE
4365  020500  012737  020506  001110            MOV     #1$,$LPERR         ;LOAD LOOP ON ERROR LOCATION FOR
4366                                                                       ;  SUBTEST LOOP
4367
4368  020506                          1$:
4369  020506  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
4370  020514  013762  004244  000026            MOV     DRVCOD,RKMR1(R2)           ;LOAD MESSAGE SELECT CODE
4371  020522  052762  000040  000026            BIS     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
4372  020530  013762  004244  000010            MOV     DRVCOD,RKCS2(R2)           ;LOAD DRIVE SELECT CODE
4373  020536  012762  000001  000000            MOV     #SELDRV,RKCS1(R2)          ;ISSUE SELECT DRIVE
4374  020544  012700  000116                    MOV     #19.*4+2,R0        ;LOAD DRIVE MESSAGE AND SHIFT
4375  020550  052762  000400  000026  2$:       BIS     #MCLK,RKMR1(R2) ;  ALL 16 BITS
4376  020556  042762  000400  000026            BIC     #MCLK,RKMR1(R2)
4377  020564  005300                            DEC     R0
4378  020566  001370                            BNE     2$
4379  020570  016237  000034  004146            MOV     RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4380  020576  016237  000036  004150            MOV     RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4381  020604  013701  004244                    MOV     DRVCOD,R1          ;DETERMINE PARITY
4382  020610  012703  000004                    MOV     #4,R3
4383  020614  005004                            CLR     R4
4384  020616  006001                  3$:       ROR     R1
4385  020620  103001                            BCC     4$
4386  020622  005204                            INC     R4
```

```
4387  020624  005303                    4$:     DEC     R3
4388  020626  001373                            BNE     3$
4389  020630  013737  004244  004206            MOV     DRVCOD,E.MR2    ;LOAD EXPECTED SHIFTED REG. B
4390  020636  013737  004244  004210            MOV     DRVCOD,E.MR3    ;LOAD EXPECTED SHIFTED REG. A
4391  020644  005037  004260                    CLR     PARBIT
4392  020650  032704  000001                    BIT     #BIT0,R4        ;CHECK FOR PARITY ON WORD
4393  020654  001011                            BNE     5$              ;PARITY ALREADY ODD
4394  020656  012737  100000  004260            MOV     #BIT15,PARBIT   ;SET PARITY BIT
4395  020664  052737  100000  004206            BIS     #BIT15,E.MR2
4396  020672  052737  100000  004210            BIS     #BIT15,E.MR3
4397  020700  013737  004150  001160  5$:       MOV     T.MR3,$TMPO     ;MASK ALL BITS EXCEPT PARITY
4398  020706  042737  077777  001160            BIC     #77777,$TMPO
4399  020714  023737  004260  001160            CMP     PARBIT,$TMPO    ;CHECK IF PARITY CORRECT
4400  020722  001402                            BEQ     6$              ;   ON MESSAGE A
4401  020724  104052                            ERROR   52              ;PARITY ON MESSAGE A INCORRECT
4402  020726  000426                            BR      25$             ;CHECK IF LOOP ON ERROR
4403
4404  020730  013737  004146  001160  6$:       MOV     T.MR2,$TMPO     ;MASK ALL BITS EXCEPT PARITY
4405  020736  042737  077777  001160            BIC     #77777,$TMPO
4406  020744  023737  004260  001160            CMP     PARBIT,$TMPO    ;CHECK IF PARITY CORRECT
4407  020752  001402                            BEQ     7$              ;   ON MESSAGE B
4408  020754  104053                            ERROR   53              ;PARITY ON MESSAGE B INCORRECT
4409  020756  000412                            BR      25$             ;CHECK IF LOOP ON ERROR
4410
4411  020760  023737  004210  004150  7$:       CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE A CORRECT
4412  020766  001401                            BEQ     8$              ;YES, CHECK MESSAGE B
4413  020770  104054                            ERROR   54              ;MESSAGE A INCORRECT
4414  020772  023737  004206  004146  8$:       CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE B CORRECT
4415  021000  001401                            BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
4416  021002  104055                            ERROR   55              ;MESSAGE B INCORRECT
4417  021004  104415                    25$:    SCOPE                   ;CHECK IF LOOP ON ERROR
4418  021006  005237  004244                    INC     DRVCOD          ;INCREMENT DRIVE SELECT CODE
4419  021012  022737  000017  004244            CMP     #17,DRVCOD      ;CHECK IF FINISHED
4420  021020  103232                            BHIS    1$              ;NO, TRY NEXT CONFIGURATION
4421
4422                                    ;*****************************************************************
4423                                    ;*TEST 40       DRIVE MESSAGE PARITY INTERACTION
4424                                    ;*
4425                                    ;*      CLEAR THE RK611 WITH A CONTROLLER CLEAR. PUT CONTROLLER
4426                                    ;*      IN DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 2
4427                                    ;*      WITH DRIVE SELECT = 1. LOAD COMMAND AND STATUS REGISTER 1
4428                                    ;*      WITH A SELECT COMMAND. VERIFY THAT THE CORRECT PARITY
4429                                    ;*      IS GENERATED FOR BOTH MESSAGES. REPEAT FOR MESSAGE
4430                                    ;*      SELECT = 1 AND DRIVE SELECT = 0.
4431                                    ;*
4432                                    ;*****************************************************************
4433  021022  000004                    TST40:  SCOPE
4434  021024  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
4435  021032  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
4436  021036  012737  000001  004244            MOV     #1,DRVCOD       ;SET INITIAL DRIVE SELECT CODE
4437  021044  005037  004246                    CLR     MESSCOD         ;SET INITIAL MESSAGE SELECT CODE
4438  021050  012737  100000  004206            MOV     #BIT15,E.MR2    ;LOAD EXPECTED MAINT. REG. 2 (MESS B)
4439  021056  012737  000001  004210            MOV     #BIT0,E.MR3     ;LOAD EXPECTED MAINT. REG. 3 (MESS A)
4440  021064  012737  100000  004260            MOV     #BIT15,PARBIT   ;LOAD PARITY FOR MESSAGE B
4441  021072  012737  021100  001110            MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
4442                                                                    ;   SUBTEST LOOP
```

```
4443
4444  021100                                  1$:
4445  021100  012762  100000  000000                MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
4446  021106  013762  004246  000026                MOV     MSGCOD,RKMR1(R2) ;LOAD MESSAGE SELECT CODE
4447  021114  052762  000040  000026                BIS     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
4448  021122  013762  004244  000010                MOV     DRVCOD,RKCS2(R2) ;LOAD DRIVE SELECT CODE
4449  021130  012762  000001  000000                MOV     #SELDRV,RKCS1(R2) ;ISSUE DRIVE SELECT
4450  021136  012700  000116                       MOV     #19.*4+2,R0     ;LOAD DRIVE MESSAGE AND SHIFT
4451  021142  052762  000400  000026  2$:          BIS     #MCLK,RKMR1(R2) ;  ALL 16 BITS
4452  021150  042762  000400  000026                BIC     #MCLK,RKMR1(R2)
4453  021156  005300                                DEC     R0
4454  021160  001370                                BNE     2$
4455  021162  016237  000034  004146                MOV     RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4456  021170  016237  000036  004150                MOV     RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4457  021176  013737  004150  001160                MOV     T.MR3,$TMP0     ;MASK ALL BITS EXCEPT PARITY
4458  021204  042737  077777  001160                BIC     #77777,$TMP0
4459  021212  023737  004260  001160                CMP     PARBIT,$TMP0    ;CHECK IF PARITY BIT CORRECT
4460  021220  001002                                BNE     3$              ;  ON MESSAGE A
4461  021222  104052                                ERROR   52              ;NO, PARITY ON MESSAGE INCORRECT
4462  021224  000426                                BR      25$             ;CHECK IF LOOP ON ERROR
4463
4464  021226  013737  004146  001160  3$:          MOV     T.MR2,$TMP0     ;MASK ALL BITS EXCEPT PARITY
4465  021234  042737  077777  001160                BIC     #77777,$TMP0
4466  021242  023737  004260  001160                CMP     PARBIT,$TMP0    ;CHECK IF PARITY CORRECT
4467  021250  001402                                BEQ     4$              ;  MESSAGE B
4468  021252  104053                                ERROR   53              ;PARITY ON MESSAGE B INCORRECT
4469  021254  000412                                BR      25$             ;CHECK IF LOOP ON ERROR
4470
4471  021256  023737  004210  004150  4$:          CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE A CORRECT
4472  021264  001401                                BEQ     5$              ;YES, CHECK IN MESSAGE B CORRECT
4473  021266  104054                                ERROR   54              ;MESSAGE A INCORRECT
4474  021270  023737  004206  004146  5$:          CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE B CORRECT
4475  021276  001401                                BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
4476  021300  104055                                ERROR   55              ;MESSAGE B INCORRECT
4477  021302  104415                        25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
4478  021304  005737  004244                       TST     DRVCOD          ;CHECK IF DRIVE SELECT = 0 (FINISHED)
4479  021310  001416                                BEQ     TST41           ;YES,GO ON TO NEXT TEST
4480  021312  005037  004244                       CLR     DRVCOD          ;SET DRIVE SELECT CODE = 0
4481  021316  012737  000001  004246                MOV     #1,MSGCOD       ;SET MESSAGE SELECT CODE
4482  021324  012737  000001  004206                MOV     #BIT0,E.MR2     ;LOAD EXPECTED MAINT REG 2 (MESS B)
4483  021332  012737  100000  004210                MOV     #BIT15,E.MR3    ;LOAD EXPECTED MAINT REG 3 (MESS A)
4484  021340  005037  004260                       CLR     PARBIT          ;LOAD PARITY FOR MESSAGE B
4485  021344  000655                                BR      1$              ;TRY SECOND CONFIGURATION
4486
4487                          ;*****************************************************************
4488                          ;*TEST 41       EVEN DRIVE MESSAGE PARITY GENERATION
4489                          ;*
4490                          ;*       CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER
4491                          ;*       IN DIAGNOSTIC MODE AND MESSAGE SELECT = 1
4492                          ;*       AND BAD PARITY SET. LOAD COMMAND AND STATUS
4493                          ;*       REGISTER 2 WITH DRIVE SELECT = 1. LOAD COMMAND
4494                          ;*       AND STATUS REGISTER SELECT COMMAND. VERIFY THAT
4495                          ;*       EVEN PARITY IS GENERATED.  REPEAT FOR MESSAGE SELECT =
4496                          ;*       DRIVE SELECT = 2-17.
4497                          ;*
4498                          ;*****************************************************************
```

```
4499  021346  000004                        TST41:  SCOPE
4500  021350  012737  000144  001200                MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
4501  021356  013702  001270                        MOV     $BASE,R2        ;LOAD RK611 BASE
4502  021362  012737  000001  004244                MOV     #1,DRVCOD       ;LOAD DRIVE CODE
4503  021370  012737  021376  001110                MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
4504                                                                        ;  SUBTEST LOOP
4505
4506  021376                                1$:
4507  021376  012762  100000  000000                MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
4508  021404  013762  004244  000026                MOV     DRVCOD,RKMR1(R2)        ;LOAD MESSAGE SELECT CODE
4509  021412  052762  000060  000026                BIS     #DMD!PAT,RKMR1(R2)      ;PUT RK611 IN MAINTENANCE MODE
4510                                                                        ;  AND SET BAD PARITY
4511  021420  013762  004244  000010                MOV     DRVCOD,RKCS2(R2)        ;LOAD DRIVE SELECT CODE
4512  021426  012762  000001  000000                MOV     #SELDRV,RKCS1(R2)       ;ISSUE SELECT DRIVE
4513  021434  012700  000116                       MOV     #19.*4+2,R0     ;LOAD DRIVE MESSAGE AND SHIFT
4514  021440  052762  000400  000026        2$:     BIS     #MCLK,RKMR1(R2) ;  ALL 16 BITS
4515  021446  042762  000400  000026                BIC     #MCLK,RKMR1(R2)
4516  021454  005300                                DEC     R0
4517  021456  001370                                BNE     2$
4518  021460  016237  000034  004146                MOV     RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4519  021466  016237  000036  004150                MOV     RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4520  021474  013701  004244                       MOV     DRVCOD,R1       ;DETERMINE PARITY
4521  021500  012703  000004                       MOV     #4,R3
4522  021504  005004                                CLR     R4
4523  021506  006001                        3$:     ROR     R1
4524  021510  103001                                BCC     4$
4525  021512  005204                                INC     R4
4526  021514  005303                        4$:     DEC     R3
4527  021516  001373                                BNE     3$
4528  021520  013737  004244  004206                MOV     DRVCOD,E.MR2    ;LOAD EXPECTED SHIFTED REG. B
4529  021526  013737  004244  004210                MOV     DRVCOD,E.MR3    ;LOAD EXPECTED SHIFTED REG. A
4530  021534  005037  004260                       CLR     PARBIT
4531  021540  032704  000001                       BIT     #BIT0,R4        ;CHECK FOR PARITY ON WORD
4532  021544  001411                                BEQ     5$              ;PARITY ALREADY EVEN
4533  021546  012737  100000  004260                MOV     #BIT15,PARBIT   ;SET PARITY BIT
4534  021554  052737  100000  004206                BIS     #BIT15,E.MR2
4535  021562  052737  100000  004210                BIS     #BIT15,E.MR3
4536  021570  013737  004150  001160        5$:     MOV     T.MR3,$TMP0     ;MASK ALL BITS EXCEPT PARITY
4537  021576  042737  077777  001160                BIC     #77777,$TMP0
4538  021604  023737  004260  001160                CMP     PARBIT,$TMP0    ;CHECK IF PARITY CORRECT
4539  021612  001402                                BEQ     6$              ;  ON MESSAGE A
4540  021614  104056                                ERROR   56              ;PARITY ON MESSAGE A INCORRECT
4541  021616  000426                                BR      25$             ;CHECK IF LOOP ON ERROR
4542
4543  021620  013737  004146  001160        6$:     MOV     T.MR2,$TMP0     ;MASK ALL BITS EXCEPT PARITY
4544  021626  042737  077777  001160                BIC     #77777,$TMP0
4545  021634  023737  004260  001160                CMP     PARBIT,$TMP0    ;CHECK IF PARITY CORRECT
4546  021642  001402                                BEQ     7$              ;  ON MESSAGE B
4547  021644  104057                                ERROR   57              ;PARITY ON MESSAGE B INCORRECT
4548  021646  000412                                BR      25$             ;CHECK IF LOOP ON ERROR
4549
4550  021650  023737  004210  004150        7$:     CMP     E.MR3,T.MR3     ;CHECK IF MESSAGE A CORRECT
4551  021656  001401                                BEQ     8$              ;YES, CHECK MESSAGE B
4552  021660  104060                                ERROR   60              ;MESSAGE A INCORRECT
4553  021662  023737  004206  004146        8$:     CMP     E.MR2,T.MR2     ;CHECK IF MESSAGE B CORRECT
4554  021670  001401                                BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
```

```
                                        ERROR   61              ;MESSAGE B INCORRECT
                                        SCOPE                   ;CHECK IF LOOP ON ERROR
                                25$:    INC     DRVCOD          ;INCREMENT DRIVE SELECT CODE
                                        CMP     #17,DRVCOD      ;CHECK IF FINISHED
                                        BHIS    1$              ;NO, TRY NEXT CONFIGURATION


                                .SBTTL  **CLASS A COMMAND EXECUTION

                                ;;*****************************************************************
                                ;*TEST 42        RELEASE COMMAND IN DIAGNOSTIC MODE
                                ;*
                                ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND
                                ;*      STATUS REGISTER 2 WITH DRIVE SELECT = 10. LOAD
                                ;*      COMMAND AND STATUS REGISTER 1 WITH A SELECT.
                                ;*      CLOCK COMMAND TO COMPLETION. MAKE SURE UNIT
                                ;*      FIELD ERROR DOES NOT SET (SACK HIGH). REPEAT FOR
                                ;*      DRIVE SELECT = 11-17.
                                ;*
                                ;;*****************************************************************
                                ST42:   SCOPE
                                        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
                                        MOV     $BASE,R2        ;LOAD RK611 BASE
                                        MOV     #10,DRVCOD      ;INITIALIZE FOR DESELECT OF DRIVE 0
                                        MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
                                                                ;  SUBTEST LOOP

                                1$:
                                        MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
                                        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
                                        MOV     DRVCOD,RKCS2(R2) ;LOAD DRIVE SELECTION
                                        MOV     #SELDRV,RKCS1(R2) ;ISSUE DESELECT
                                        MOV     #20.*4,R0       ;LOAD COUNT TO COMPLETE COMMAND
                                2$:     MOV     #DMD!MCLK,RKMR1(R2) ;CLOCK THRU COMMAND
                                        MOV     #DMD,RKMR1(R2)
                                        DEC     R0
                                        BNE     2$
                                        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
                                        MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
                                        MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
                                        MOV     RKER(R2),T.ER   ;STORE ERROR REGISTER
                                        MOV     #RDY,E.CS1      ;LOAD EXPECTED COMMAND AND STATUS REG. 1
                                        MOV     DRVCOD,E.CS2    ;GENERATE EXPECTED COMMAND AND
                                        BIS     #IR,E.CS2       ;  STATUS REG. 2
                                        CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REGISTER
                                        CLR     E.ER            ;LOAD EXPECTED ERROR REGISTER
                                        CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
                                        BEQ     3$              ;YES, CHECK CS2
                                        ERROR   62              ;COMMAND AND STATUS REG. 1 INCORRECT
                                3$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
                                        BEQ     4$              ;YES, CHECK ERROR REGISTER
                                        ERROR   63              ;COMMAND AND STATUS REG. 2 INCORRECT
                                4$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
                                        BEQ     5$              ;YES, CHECK DRIVE STATUS REG
                                        ERROR   64              ;ERROR REGISTER INCORRECT
```

```
4611  022136  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
4612  022144  001401                          BEQ     6$              ;YES, CHECK IF LOOP ON ERROR
4613  022146  104126                          ERROR   126             ;DRIVE STATUS REG INCORRECT
4614  022150  104415                  6$:     SCOP1                   ;CHECK IF LOOP ON ERROR
4615  022152  005237  004244                  INC     DRVCOD          ;INCREMENT DRIVE NUMBER
4616  022156  022737  000017  004244          CMP     #17,DRVCOD      ;CHECK IF ALL DRIVE NUMBERS TESTED
4617  022164  103266                          BHIS    1$              ;NO, DO IT FOR NEXT DRIVE NUMBER
4618
4619                                  ;;*****************************************************************
4620                                  ;*TEST 43        SELECT COMMAND IN DIAGNOSTIC MODE
4621                                  ;*
4622                                  ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
4623                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND
4624                                  ;*      STATUS REGISTER 2 WITH DRIVE SELECT = 0. LOAD
4625                                  ;*      COMMAND AND STATUS REGISTER 1 WITH A SELECT.
4626                                  ;*      CLOCK COMMAND TO COMPLETION. MAKE SURE MESSAGE SHIFT IS
4627                                  ;*      NOT DONE DURING THE RECEIVE CYCLE OF DRIVE MESSAGE.
4628                                  ;*      MAKE SURE NO ERRORS SET. REPEAT FOR DRIVE SELECT = 1-7.
4629                                  ;*
4630                                  ;*****************************************************************
4631  022166  000004                  TST43:  SCOPE
4632  022170  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
4633  022176  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
4634  022202  005037  004244                  CLR     DRVCOD          ;INITIALIZE FOR SELECT OF DRIVE 0
4635  022206  012737  022214  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
4636                                                                  ;  SUBTEST LOOP
4637
4638  022214                          1$:
4639  022214  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
4640  022222  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
4641  022230  013762  004244  000010          MOV     DRVCOD,RKCS2(R2)        ;LOAD DRIVE SELECT
4642  022236  012762  000001  000000          MOV     #SELDRV,RKCS1(R2)       ;ISSUE DRIVE SELECT
4643  022244  012700  000120                  MOV     #20.*4,R0       ;LOAD COUNT TO DESELECT COMPLETE
4644  022250  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)     ;CLOCK UNTIL DESELECT FINISHED
4645  022256  012762  000040  000026          MOV     #DMD,RKMR1(R2)
4646  022264  005300                          DEC     R0
4647  022266  001370                          BNE     2$
4648  022270  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4649  022276  012737  000001  004160          MOV     #SELDRV,E.CS1   ;LOAD EXPECTED COMMAND AND STATUS REG. 1
4650  022304  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF READY RESET
4651  022312  001402                          BEQ     3$              ;YES, CONTINUE COMMAND
4652  022314  104065                          ERROR   65              ;COMMAND AND STATUS REG. 1 INCORRECT
4653  022316  000566                          BR      25$             ;GO CHECK IF LOOP ON ERROR
4654
4655  022320  013703  004244          3$:     MOV     DRVCOD,R3       ;GENERATE EXPECTED MAINT REG 3
4656  022324  012701  000003                  MOV     #3,R1
4657  022330  005000                          CLR     R0
4658  022332  006003                  4$:     ROR     R3
4659  022334  103001                          BCC     5$
4660  022336  005200                          INC     R0
4661  022340  005301                  5$:     DEC     R1
4662  022342  001373                          BNE     4$
4663  022344  013737  004244  004210          MOV     DRVCOD,E.MR3
4664  022352  032700  000001                  BIT     #BIT0,R0
4665  022356  001003                          BNE     6$
4666  022360  052737  100000  004210          BIS     #BIT15,E.MR3
```

```
 4667  022366  012737  100000  004206  6$:     MOV     #BIT15,E.MR2    ;STORE EXPECTED MAINT REG 2
 4668  022374  012701  000003          7$:     MOV     #3,R1           ;ISSUE 3 CONTROL CLOCKS
 4669  022400  012700  000004                  MOV     #4,R0
 4670  022404  012762  000440  000026  8$:     MOV     #DMD!MCLK,RKMR1(R2)
 4671  022412  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 4672  022420  005300                          DEC     R0
 4673  022422  001370                          BNE     8$
 4674  022424  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
 4675  022432  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG 2
 4676  022440  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG 3
 4677  022446  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
 4678  022454  001402                          BEQ     9$              ;YES, CHECK MAINTENANCE REG. 2
 4679  022456  104066                          ERROR   66              ;CS1 INCORRECT
 4680  022460  000505                          BR      25$             ;CHECK IF LOOP ON ERROR
 4681
 4682  022462  023737  004206  004146  9$:     CMP     E.MR2,T.MR2     ;CHECK MAINT REG 2 CORRECT
 4683  022470  001402                          BEQ     10$             ;YES, CHECK MAINTENANCE REG 3
 4684  022472  104067                          ERROR   67              ;MR2 INCORRECT
 4685  022474  000477                          BR      25$             ;CHECK IF LOOP ON ERROR
 4686
 4687  022476  023737  004210  004150  10$:    CMP     E.MR3,T.MR3     ;CHECK IF MAINT REG 3 CORRECT
 4688  022504  001402                          BEQ     11$             ;YES, CHECK COMMAND COMPLETE
 4689  022506  104070                          ERROR   70              ;MR3 INCORRECT
 4690  022510  000471                          BR      25$             ;CHECK IF LOOP ON ERROR
 4691
 4692  022512  005301                  11$:    DEC     R1              ;CHECK IF COMMAND FINISHED
 4693  022514  001331                          BNE     7$              ;NO, ISSUE ANOTHER CONTROL CLOCK
 4694  022516  012700  000004                  MOV     #4,R0           ;ISSUE LAST CONTROL CLOCK FOR READY
 4695  022522  012762  000440  000026  12$:    MOV     #DMD!MCLK,RKMR1(R2)
 4696  022530  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 4697  022536  005300                          DEC     R0
 4698  022540  001370                          BNE     12$
 4699  022542  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
 4700  022550  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
 4701  022556  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
 4702  022564  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REGISTER
 4703  022572  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED COMMAND AND STATUS REG 1
 4704  022600  013737  004244  004170          MOV     DRVCOD,E.CS2    ;GENERATE EXPECTED COMMAND AND STATUS REG. 2
 4705  022606  052737  000100  004170          BIS     #IR,E.CS2
 4706  022614  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REGISTER
 4707  022620  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REGISTER
 4708  022624  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
 4709  022632  001401                          BEQ     13$             ;YES, CHECK CS2
 4710  022634  104071                          ERROR   71              ;CS1 INCORRECT
 4711  022636  023737  004170  004130  13$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
 4712  022644  001401                          BEQ     14$             ;YES, CHECK ERROR REG
 4713  022646  104072                          ERROR   72              ;CS2 INCORRECT
 4714  022650  023737  004174  004134  14$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
 4715  022656  001401                          BEQ     15$             ;YES, CHECK DRIVE STATUS REG CORRECT
 4716  022660  104073                          ERROR   73              ;ERROR REG INCORRECT
 4717  022662  023737  004172  004132  15$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
 4718  022670  001401                          BEQ     25$             ;YES, CHECK IF LOOP ON ERROR
 4719  022672  104127                          ERROR   127             ;DRIVE STATUS REGISTER INCORRECT
 4720  022674  104415                  25$:    SCOP1                   ;CHECK IF LOOP ON ERROR
 4721  022676  005237  004244                  INC     DRVCOD          ;INCREMENT DRIVE NUMBER
 4722  022702  022737  000007  004244          CMP     #7,DRVCOD       ;CHECK IF ALL DRIVES TESTED
```

```
                                        BLO     TST44           ;;YES, GO TO NEXT TEST
                                        JMP     1$              ;TRY NEXT DRIVE

                                ;;******************************************************************
                                ;*TEST 44        RELEASE COMMAND IN NORMAL MODE
                                ;*
                                ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                ;*      LOAD COMMAND AND STATUS REGISTER 2 WITH DRIVE SELECT = 10.
                                ;*      LOAD COMMAND AND STATUS REGISTER 1 WITH A SELECT.
                                ;*      MAKE SURE NO ERRORS OCCUR. REPEAT FOR DRIVE
                                ;*      SELECT = 11-17
                                ;*
                                ;******************************************************************
                                TST44:  SCOPE
                                        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
                                        MOV     $BASE, R2       ;LOAD RK611 BASE
                                        MOV     #10,DRVCOD      ;INITIALIZE FOR DESELECT OF DRIVE 0
                                        MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
                                                                ;  SUBTEST LOOP

                                1$:
                                        MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
                                        MOV     DRVCOD,RKCS2(R2)        ;LOAD DRIVE SELECTION
                                        MOV     #SELDRV,RKCS1(R2)       ;ISSUE DESELECT
                                        MOV     WAITIM,R0       ;WAIT FOR READY
                                2$:     TSTB    RKCS1(R2)
                                        BMI     3$
                                        DEC     R0
                                        BNE     2$
                                3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
                                        MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
                                        MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
                                        MOV     RKER(R2),T.ER   ;STORE ERROR REG.
                                        MOV     #RDY,E.CS1      ;LOAD EXPECTED COMMAND AND STATUS REG. 1
                                        MOV     DRVCOD,E.CS2    ;GENERATE EXPECTED COMMAND AND STATUS REG. 2
                                        BIS     #IR,E.CS2
                                        CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
                                        CLR     E.ER            ;LOAD EXPECTED ERROR REG.
                                        CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
                                        BEQ     4$              ;YES, CHECK CS2
                                        ERROR   74              ;CS1 INCORRECT
                                4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
                                        BEQ     5$              ;YES, CHECK ERROR REGISTER
                                        ERROR   75              ;CS2 INCORRECT
                                5$:     CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
                                        BEQ     6$              ;YES, CHECK DRIVE STATUS REG CORRECT
                                        ERROR   76              ;ERROR REG INCORRECT
                                6$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
                                        BEQ     7$              ;YES, CHECK IF LOOP ON ERROR
                                        ERROR   130             ;DRIVE STATUS REGISTER INCORRECT
                                7$:     SCOP1                   ;CHECK IF LOOP ON ERROR
                                        INC     DRVCOD          ;INCREMENT DRIVE NUMBER
                                        CMP     #17,DRVCOD      ;CHECK IF ALL DRIVE NUMBERS TESTED
                                        BHIS    1$              ;NO, DO IT FOR NEXT DRIVE

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

```
                                        ;*TEST 45       INTERRUPT AT COMMAND COMPLETION
                                        ;*
                                        ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                        ;*      LOWER PROCESSOR PRIORITY TO ZERO. ISSUE A RELEASE
                                        ;*      COMMAND WITH INTERRUPT ENABLE SET. MAKE SURE
                                        ;*      INTERRUPT OCCURS. LOWER PRIORITY AFTER INTERRUPT
                                        ;*      AND MAKE SURE INTERRUPT HAS CLEARED.
                                        ;*
                                        ;*      LOWER PROCESSOR PRIORITY TO ZERO.  REISSUE RELEASE
                                        ;*      WITH INTERRUPT ENABLE RESET.  MAKE SURE NO INTERRUPT
                                        ;*      OCCURS.  SET INTERRUPT ENABLE AND MAKE SURE NO
                                        ;*      INTERRUPT OCCURS.
                                        ;*
                                        ;*****************************************************************
023156  000004                  TST45:  SCOPE
023160  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
023166  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
023172  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
023200  012762  000010  000010          MOV     #10,RKCS2(R2)   ;SET DESELECT BIT
023206  013701  004234                  MOV     RKVEC,R1        ;LOAD INTERRUPT VECTOR
023212  012721  023274                  MOV     #5$,(R1)+
023216  012711  000340                  MOV     #PR7,(R1)
023222  005046                          CLR     -(SP)           ;LOAD STACK TO ALLOW ALL INTERRUPTS
023224  012746  023232                  MOV     #64$,-(SP)      ;LOAD NEXT ADDRESS
023230  000002                          RTI                     ;CLEAR PSW
                                64$:
023232  012762  000101  000000          MOV     #SELDRV!IE,RKCS1(R2)    ;ISSUE SELECT DRIVE
023240  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
023244  105762  000000          2$:     TSTB    RKCS1(R2)
023250  100402                          BMI     3$
023252  005300                          DEC     R0
023254  001373                          BNE     2$
023256  012746  000340          3$:     MOV     #PR7,-(SP)      ;LOCK OUT INTERRUPTS
023262  012746  023270                  MOV     #4$,-(SP)
023266  000002                          RTI

023270  104100                  4$:     ERROR   100             ;INTERRUPT DID NOT OCCUR
023272  000522                          BR      25$

023274  062706  000004          5$:     ADD     #4,SP           ;ADJUST STACK
023300  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
023306  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
023314  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG.
023322  012737  000300  004160          MOV     #RDY!IE,E.CS1   ;LOAD EXPECTED CS1
023330  012737  000110  004170          MOV     #IR!10,E.CS2    ;LOAD EXPECTED CS2
023336  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR
023342  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
023350  001401                          BEQ     6$              ;YES, CHECK CS2
023352  104101                          ERROR   101             ;CS1 INCORRECT
023354  023737  004170  004130  6$:     CMP     E.CS2,T.CS2     ;CHECK IF CS2 INCORRECT
023362  001401                          BEQ     7$              ;YES, CHECK IF ERROR REG CORRECT
023364  104102                          ERROR   102             ;CS2 INCORRECT
023366  023737  004174  004134  7$:     CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
023374  001401                          BEQ     8$              ;YES, CHECK IF INTERRUPT CLEARED
023376  104103                          ERROR   103             ;ERROR REG. INCORRECT
```

```
 4835  023400  012777  023512  160626  8$:     MOV     #10$,@RKVEC     ;LOAD VECTOR FOR UNEXPECTED INTERRUPT
 4836  023406  005046                          CLR     -(SP)           ;LOAD STACK TO ALLOW ALL INTERRUPTS
 4837  023410  012746  023416                  MOV     #65$,-(SP)      ;LOAD NEXT ADDRESS
 4838  023414  000002                          RTI                     ;CLEAR PSW
 4839
 4840  023416                          65$:
 4841  023416  000240                          NOP                     ;WAIT FOR INTERRUPT
 4842  023420  012777  023522  160606          MOV     #15$,@RKVEC     ;LOAD VECTOR ADDRESS FOR UNEXPECTED INTERRUPT
 4843  023426  012762  000010  000010          MOV     #10,RKCS2(R2)   ;ISSUE DESELECT
 4844  023434  012762  000001  000000          MOV     #SELDRV,RKCS1(R2)
 4845  023442  013700  004262                  MOV     WAITIM,R0
 4846  023446  105762  000000          9$:     TSTB    RKCS1(R2)
 4847  023452  100402                          BMI     11$
 4848  023454  005300                          DEC     R0
 4849  023456  001373                          BNE     9$
 4850  023460  000240                  11$:    NOP                     ;WAIT FOR INTERRUPT
 4851  023462  012777  023532  160544          MOV     #20$,@RKVEC     ;LOAD VECTOR ADDRESS FOR UNEXPECTED INTERRUPT
 4852  023470  012762  000100  000000          MOV     #IE,RKCS1(R2)   ;SET INTERRUPT ENABLE
 4853  023476  000240                          NOP                     ;ALLOW INTERRUPT TO OCCUR
 4854  023500  012746  000340                  MOV     #PR7,-(SP)      ;LOCK OUT INTERRUPT
 4855  023504  012746  023540                  MOV     #25$,-(SP)      ;RESTORE TRAP CATCHER
 4856  023510  000002                          RTI
 4857
 4858  023512  062706  000004          10$:    ADD     #4,SP           ;ADJUST STACK
 4859  023516  104104                          ERROR   104             ;UNEXPECTED INTERRUPT
 4860  023520  000407                          BR      25$             ;RESTORE TRAP CATCHER
 4861
 4862  023522  062706  000004          15$:    ADD     #4,SP           ;ADJUST STACK
 4863  023526  104254                          ERROR   254             ;UNEXPECTED INTERRUPT ON DESELECT
 4864  023530  000403                          BR      25$             ;RESTORE TRAP CATCHER
 4865
 4866  023532  062706  000004          20$:    ADD     #4,SP           ;ADJUST STACK
 4867  023536  104255                          ERROR   255             ;UNEXPECTED INTERRUPT WHEN SETTING
 4868                                                                  ;   INTERRUPT ENABLE
 4869  023540  012762  000040  000010  25$:    MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
 4870  023546  013701  004234                  MOV     RKVEC,R1        ;RESTORE TRAP CATCHER
 4871  023552  010111                          MOV     R1,(R1)
 4872  023554  062721  000002                  ADD     #2,(R1)+
 4873  023560  005011                          CLR     (R1)
 4874
 4875                                  ;;*****************************************************************
 4876                                  ;*TEST 46       GO CLEAR OF SILO
 4877                                  ;*
 4878                                  ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
 4879                                  ;*      WRITE ONE WORD INTO THE SILO.  ISSUE A RELEASE COMMAND
 4880                                  ;*      WITH INTERRUPT ENABLE RESET.  WAIT FOR READY.
 4881                                  ;*      READ THE DATA BUFFER TO MAKE SURE THE SILO HAS BEEN
 4882                                  ;*      CLEARED. (DATA LATE SET AFTER READ OF DATA BUFFER)
 4883                                  ;*
 4884                                  ;;*****************************************************************
 4885  023562  000004                  TST46:  SCOPE
 4886  023564  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 4887  023572  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 4888  023576  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
 4889  023604  005062  000024                  CLR     RKDB(R2)        ;LOAD 1 WORD IN SILO
 4890  023610  012762  000010  000010          MOV     #10,RKCS2(R2)   ;LOAD DESELECT DRIVE 0
```

```
4891  023616  012762  000001  000000          MOV     #SELDRV,RKCS1(R2)       ;ISSUE DESELECT
4892  023624  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
4893  023630  105762  000000          2$:     TSTB    RKCS1(R2)
4894  023634  100402                          BMI     3$
4895  023636  005300                          DEC     R0
4896  023640  001373                          BNE     2$
4897  023642  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4898  023650  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
4899  023656  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
4900  023664  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REGISTER
4901  023672  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
4902  023700  012737  000110  004170          MOV     #IR!10,E.CS2    ;LOAD EXPECTED CS2
4903  023706  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
4904  023712  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REGISTER
4905  023716  023737  004170  004130          CMP     E.CS2,T.CS2     ;CHECK IF CS1 CORRECT
4906  023724  001401                          BEQ     10$             ;YES, READ WORD FROM SILO
4907  023726  104105                          ERROR   105             ;CS2 INCORRECT
4908  023730  005762  000024          10$:    TST     RKDB(R2)        ;READ SILO TO MAKE IT IS CLEAR
4909  023734  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4910  023742  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
4911  023750  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG.
4912  023756  012737  100200  004160          MOV     #CERR!RDY,E.CS1 ;LOAD EXPECTED CS1
4913  023764  012737  100110  004170          MOV     #DLT!IR!10,E.CS2 ;LOAD EXPECTED CS2
4914  023772  023737  004170  004130          CMP     E.CS2,T.CS2     ;CHECK IF DATA LATE SET
4915  024000  001401                          BEQ     11$             ;YES, CLEAR CONTROLLER REG. 1
4916  024002  104106                          ERROR   106             ;DATA LATE NOT SET
4917  024004  012762  100000  000000  11$:    MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
4918
4919                                  ;;*****************************************************************
4920                                  ;*TEST 47       SEEK COMMAND IN DIAGNOSTIC MODE
4921                                  ;*
4922                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
4923                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET
4924                                  ;*      24 SECTOR FORMAT TO CYLINDER 1714, HEAD 7, DRIVE 0.
4925                                  ;*      MAKE SURE NO STATUS BITS ARE SET AND NO ERROR
4926                                  ;*      BITS ARE SET.
4927                                  ;*
4928                                  ;*****************************************************************
4929  024012  000004                  ;ST47:  SCOPE
4930  024014  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
4931  024022  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
4932  024026  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
4933  024034  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
4934  024042  012762  001714  000020          MOV     #1714,RKDCYL(R2)        ;LOAD CYLINDER ADDRESS
4935  024050  012762  003400  000006          MOV     #3400,RKDA(R2)  ;LOAD HEAD 7
4936  024056  012762  012017  000000          MOV     #SEEK!CFMT!CDT,RKCS1(R2)        ;ISSUE SEEK CDT SET,24 SECTOR
4937  024064  012700  000120                  MOV     #20.*4,R0       ;LOAD COUNT TO DESELECT DECISION
4938  024070  012762  000440  000026  2$:     MOV     #DMD!MCLK,RKMR1(R2)
4939  024076  012762  000040  000026          MOV     #DMD,RKMR1(R2)
4940  024104  005300                          DEC     R0
4941  024106  001370                          BNE     2$
4942  024110  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4943  024116  012737  012017  004160          MOV     #SEEK!CFMT!CDT,E.CS1    ;LOAD EXPECTED COMMAND AND STATUS REG 1
4944  024124  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF READY RESET
4945  024132  001402                          BEQ     3$              ;YES, CONTINUE COMMAND
4946  024134  104107                          ERROR   107
```

```
4947  024136  000543                                    BR      TST50           ;;GO ON TO NEXT TEST
4948
4949  024140  012737  071020  004210  3$:       MOV     #S.SEEK!S.EMI!70000,E.MR3       ;LOAD EXPECTED MAINT REG. 3
4950  024146  012737  136300  004206            MOV     #136300,E.MR2   ;LOAD EXPECTED MAINT REG. 2
4951  024154  012701  000003                    MOV     #3,R1           ;ISSUE 3 CONTROL CLOCKS
4952  024160  012700  000004                4$: MOV     #4,R0
4953  024164  012762  000440  000026    5$:   MOV     #DMD!MCLK,RKMR1(R2)
4954  024172  012762  000040  000026          MOV     #DMD,RKMR1(R2)
4955  024200  005300                           DEC     R0
4956  024202  001370                           BNE     5$
4957  024204  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
4958  024212  016237  000034  004146          MOV     RKMR2(R2),T.MR2 ;STORE MAINT REG 2
4959  024220  016237  000036  004150          MOV     RKMR3(R2),T.MR3 ;STORE MAINT REG 3
4960  024226  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG. 1 CORRECT
4961  024234  001402                           BEQ     6$              ;YES, CHECK MAINTENANCE REG. 2
4962  024236  104110                           ERROR   110             ;CS1 INCORRECT
4963  024240  000502                           BR      TST50           ;;GO TO NEXT TEST
4964
4965  024242  023737  004206  004146  6$:     CMP     E.MR2,T.MR2     ;CHECK MAINT REG 2 CORRECT
4966  024250  001402                           BEQ     7$              ;YES, CHECK MAINTENANCE REG 3
4967  024252  104111                           ERROR   111             ;MAINT REG 2 INCORRECT
4968  024254  000474                           BR      TST50           ;;GO TO NEXT TEST
4969
4970  024256  023737  004210  004150  7$:     CMP     E.MR3,T.MR3     ;CHECK IF MAINT REG 3 CORRECT
4971  024264  001402                           BEQ     8$              ;YES, CHECK COMMAND COMPLETE
4972  024266  104112                           ERROR   112             ;MR3 INCORRECT
4973  024270  000466                           BR      TST50           ;;GO TO NEXT TEST
4974
4975  024272  005301                       8$: DEC     R1              ;CHECK IF COMMAND FINISHED
4976  024274  001331                           BNE     4$              ;NO, ISSUE ANOTHER CONTROL CLOCK
4977
4978  024276  012700  000004                   MOV     #4,R0           ;ISSUE LAST CONTROL CLOCK FOR READY
4979  024302  012762  000440  000026    9$:   MOV     #DMD!MCLK,RKMR1(R2)
4980  024310  012762  000040  000026          MOV     #DMD,RKMR1(R2)
4981  024316  005300                           DEC     R0
4982  024320  001370                           BNE     9$
4983  024322  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4984  024330  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
4985  024336  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
4986  024344  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REGISTER
4987  024352  012737  012216  004160          MOV     #RDY!CFMT!CDT!<SEEK&^C<GO>>,E.CS1       ;LOAD EXPECTED CS1
4988  024360  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
4989  024366  005037  004172                   CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REGISTER
4990  024372  005037  004174                   CLR     E.ER            ;LOAD EXPECTED ERROR REGISTER
4991  024376  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK IF COMMAND AND STATUS REG. 2
4992  024404  001401                           BEQ     10$             ;YES, CHECK CS2
4993  024406  104113                           ERROR   113             ;CS1 INCORRECT
4994  024410  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
4995  024416  001401                           BEQ     11$             ;YES, CHECK ERROR REG
4996  024420  104114                           ERROR   114             ;CS2 INCORRECT
4997  024422  023737  004174  004134  11$:    CMP     E.ER,T.ER       ;CHECK ERROR REGISTER
4998  024430  001401                           BEQ     12$             ;YES, CHECK DRIVE STATUS REG
4999  024432  104115                           ERROR   115             ;ERROR REG. INCORRECT
5000  024434  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REGISTER CORRECT
5001  024442  001401                           BEQ     TST50           ;;YES, GO ON TO NEXT TEST
5002  024444  104131                           ERROR   131             ;DRIVE STATUS REGISTER INCORRECT
```

```
                                        .SBTTL  **ERROR AND STATUS BIT FORCING WITH DRIVE MESSAGES

                                        ;;*****************************************************************
                                        ;*TEST 50       DRIVE STATUS FROM SHIFT REGISTER
                                        ;*
                                        ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
                                        ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
                                        ;*      TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 757, HEAD 1,
                                        ;*      DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS
                                        ;*      6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE SPEED LOSS,
                                        ;*      DRIVE AVAILABLE, VOLUME VALID, OFFSET, DRIVE READY,
                                        ;*      AND WRITE LOCK ARE SET.
                                        ;*
                                        ;;*****************************************************************
024446  000004                  TST50:  SCOPE
024450  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
024456  013702  001270                  MOV     $PASE,R2        ;LOAD RK611 BASE
024462  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
024470  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
024476  012762  000757  000020          MOV     #757,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
024504  012762  000400  000006          MOV     #400,RKDA(R2)   ;LOAD HEAD ADD =1
024512  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
024520  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
024524  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
024532  012762  000040  000026          MOV     #DMD,RKMR1(R2)
024540  005300                          DEC     R0
024542  001370                          BNE     1$
024544  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
024550  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR FOR READY
024554  105762  000000          2$:     TSTB    RKCS1 R2)
024560  100412                          BMI     3$
024562  005300                          DEC     R0
024564  001373                          BNE     2$
024566  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
024574  012737  000216  004160          MOV     #RDY!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
024602  104132                          ERROR   132             ;READY NOT SET
024604  000460                          BR      10$             ;CLEAR RK06 SUBSYSTEM

024606  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
024614  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
024622  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REF
024630  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG.
024636  012737  000216  004160          MOV     #RDY!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECT CS1
024644  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
024652  012737  104325  004172          MOV     #DRA!OFST!SPDLSS!VV!DRDY!WRL!SVAL,E.DS ;LOAD EXPECTED DRIVE STATUS
024660  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REGESTER
024664  032737  004325  004132          BIT     #DRA!OFST!SPDLSS!VV!DRDY!WRL,T.DS ;CHECK LOAD STATUS SET
024672  001001                          BNE     4$              ;YES, CHECK REGISTERS
024674  104133                          ERROR   133             ;NO BIT IN DRIVE STATUS SET
024676  023737  004160  004120  4$:     CMP     E.CS1,T.CS1     ;CHECK CS1 CORRECT
024704  001401                          BEQ     5$              ;YES, CONTINUE
024706  104134                          ERROR   134             ;CS1 INCORRECT
024710  023737  004170  004130  5$:     CMP     E.CS2,T.CS2     ;CHECK CS2 CORRECT
024716  001401                          BEQ     6$              ;YES, CONTINUE
024720  104135                          ERROR   135             ;CS2, INCORRECT
```

```
5059  024722  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
5060  024730  001401                          BEQ     7$              ;YES CONTINUE
5061  024732  104136                          ERROR   136             ;ERROR REG INCORRECT
5062  024734  023737  004172  004132  7$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS CORRECT
5063  024742  001401                          BEQ     10$             ;CLEAR RK611
5064  024744  104137                          ERROR   137             ;DRIVE STATUS INCORRECT
5065  024746  013737  004120  004220  10$:    MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5066  024754  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG 1
5067  024762  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG 2
5068  024770  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG
5069                                          ;   AND ERROR REG
5070  024776  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5071  025004  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5072  025012  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5073  025020  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5074  025026  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5075  025034  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5076  025042  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5077  025050  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5078  025054  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5079  025060  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5080  025066  001401                          BEQ     11$             ;YES, CHECK CS2
5081  025070  104224                          ERROR   224             ;CS1 INCORRECT
5082  025072  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5083  025100  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5084  025102  104225                          ERROR   225             ;CS2 INCORRECT
5085  025104  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5086  025112  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5087  025114  104226                          ERROR   226             ;ERROR REG INCORRECT
5088  025116  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5089  025124  001401                          BEQ     TSTS1           ;YES, GO ON TO NEXT TEST
5090  025126  104227                          ERROR   227             ;ERROR REG INCORRECT
5091
5092                                          ;;*****************************************************************
5093                                          ;*TEST 51        DRIVE AVAILABLE SETTING
5094                                          ;*
5095                                          ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5096                                          ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK06.
5097                                          ;*      26 SECTOR FORMAT TO CYLINDER 2, HEAD 0, DRIVE 0.
5098                                          ;*      CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
5099                                          ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE
5100                                          ;*      AVAILIABLE SETS.
5101                                          ;*
5102                                          ;;*****************************************************************
5103  025130  000004                  TST51:  SCOPE
5104  025132  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5105  025140  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
5106  025144  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5107  025152  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5108  025160  012762  000002  000020          MOV     #2,RKDCYL(R2)   ;LOAD CYLINDER AND
5109  025166  012762  000000  000006          MOV     #0,RKDA(R2)     ;LOAD HEAD ADDRESS
5110  025174  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
5111  025202  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5112  025206  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5113  025214  012762  000040  000026          MOV     #DMD,RKMR1(R2)
5114  025222  005300                          DEC     R0
```

```
5115  025224  001370                          BNE     1$
5116  025226  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5117  025232  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5118  025236  105762  000000          2$:     TSTB    RKCS1(R2)
5119  025242  100402                          BMI     3$
5120  025244  005300                          DEC     R0
5121  025246  001373                          BNE     2$
5122  025250  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5123  025256  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5124  025264  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5125  025272  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5126  025300  012737  000216  004160          MOV     #RDY!SEEK&^C<GO>,E.CS1 ;LOAD EXPECTED CS1
5127  025306  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5128  025314  012737  100001  004172          MOV     #SVAL!DRA,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
5129  025322  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
5130  025330  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
5131  025336  001401                          BEQ     4$              ;YES, CONTINUE
5132  025340  104140                          ERROR   140
5133  025342  023737  004170  004130  4$      CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5134  025350  001401                          BEQ     5$              ;YES, CONTINUE
5135  025352  104141                          ERROR   141
5136  025354  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
5137  025362  001401                          BEQ     6$              ;YES, CONTINUE
5138  025364  104142                          ERROR   142
5139  025366  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
5140  025374  001401                          BEQ     7$              ;YES, CLEAR RK611
5141  025376  104143                          ERROR   143
5142  025400  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5143  025406  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
5144  025414  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
5145  025422  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
5146                                                                  ;  AND ERROR REG
5147  025430  042762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5148  025436  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5149  025444  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5150  025452  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5151  025460  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5152  025466  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5153  025474  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5154  025502  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5155  025506  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5156  025512  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5157  025520  001401                          BEQ     11$             ;YES, CHECK CS2
5158  025522  104224                          ERROR   224             ;CS1 INCORRECT
5159  025524  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5160  025532  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5161  025534  104225                          ERROR   225             ;CS2 INCORRECT
5162  025536  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5163  025544  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5164  025546  104226                          ERROR   226             ;ERROR REG INCORRECT
5165  025550  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5166  025556  001401                          BEQ     TST52           ;YES, GO ON TO NEXT TEST
5167  025560  104227                          ERROR   227             ;ERROR REG INCORRECT
5168
5169                                  ;;*****************************************************************
5170                                  ;*TEST 52        DRIVE BUS PARITY ERROR
```

```
                                        ;*
                                        ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                        ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
                                        ;*      TO A RK06, 26 SECTOR FORMAT TO CYLINDER 3, HEAD 0,
                                        ;*      DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
                                        ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE BUS
                                        ;*      PARITY, DRIVE AVAILIABLE, AND CONTROLLER ERROR ARE SET.
                                        ;*
                                        ;**************************************************************
025562  000004                  TST52:  SCOPE
025564  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
025572  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
025576  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
025604  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
025612  012762  000003  000020          MOV     #3,RKDCYL(R2)   ;LOAD CYLINDER AND
025620  012762  000000  000006          MOV     #0,RKDA(R2)     ;LOAD HEAD ADDRESS
025626  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
025634  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
025640  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
025646  012762  000040  000026          MOV     #DMD,RKMR!(R2),
025654  005300                          DEC     R0              .
025656  001370                          BNE     1$              .
025660  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
025664  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
025670  105762  000000          2$:     TSTB    RKCS1(R2)       .
025674  100402                          BMI     3$              .
025676  005300                          DEC     R0              .
025700  001373                          BNE     2$              .
025702  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
025710  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
025716  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
025724  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
025732  012737  120216  004160          MOV     #CERR!SPAR!RDY!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
025740  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
025746  012737  100001  004172          MOV     #SVAL!DRA,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
025754  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
025762  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
025770  001401                          BEQ     4$              ;YES, CONTINUE
025772  104144                          ERROR   144
025774  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
026002  001401                          BEQ     5$              ;YES, CONTINUE
026004  104145                          ERROR   145
026006  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
026014  001401                          BEQ     6$              ;YES, CONTINUE
026016  104146                          ERROR   146
026020  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
026026  001401                          BEQ     7$              ;YES, CLEAR RK611
026030  104147                          ERROR   147
026032  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
026040  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG 1
026046  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG 2
026054  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG
                                        ;   AND ERROR REG
026062  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
026070  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
026076  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
```

```
5227  026104  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5228  026112  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5229  026120  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5230  026126  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5231  026134  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5232  026140  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5233  026144  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5234  026152  001401                          BEQ     11$             ;YES, CHECK CS2
5235  026154  104224                          ERROR   224             ;CS1 INCORRECT
5236  026156  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5237  026164  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5238  026166  104225                          ERROR   225             ;CS2 INCORRECT
5239  026170  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5240  026176  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5241  026200  104226                          ERROR   226             ;ERROR REG INCORRECT
5242  026202  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5243  026210  001401                          BEQ     TST53           ;YES, GO ON TO NEXT TEST
5244  026212  104227                          ERROR   227             ;ERROR REG INCORRECT
5245
5246                                  ;*****************************************************************
5247                                  ;*TEST 53       DRIVE AVAILABLE RESET ERROR
5248                                  ;*
5249                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5250                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SELECT
5251                                  ;*      TO A RK06, 26 SECTOR FORMAT, AND DRIVE 0.
5252                                  ;*      CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
5253                                  ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE
5254                                  ;*      IS RESET AND CONTROLLER ERROR IS SET.
5255                                  ;*
5256                                  ;*****************************************************************
5257  026214  000004                  TST53:  SCOPE
5258  026216  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5259  026224  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
5260  026230  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5261  026236  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5262  026244  012762  000001  000000          MOV     #SELDRV,RKCS1(R2)       ;ISSUE SELDRV
5263  026252  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5264  026256  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5265  026264  012762  000040  000026          MOV     #DMD,RKMR1(R2)
5266  026272  005300                          DEC     R0
5267  026274  001370                          BNE     1$
5268  026276  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5269  026302  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5270  026306  105762  000000          2$:     TSTB    RKCS1(R2)
5271  026312  100402                          BMI     3$
5272  026314  005300                          DEC     R0
5273  026316  001373                          BNE     2$
5274  026320  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5275  026326  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5276  026334  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5277  026342  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5278  026350  012737  100200  004160          MOV     #CERR!RDY!SELDRV&^C<GO>,E.CS1 ;LOAD EXPECTED CS1
5279  026356  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5280  026364  012737  100000  004172          MOV     #SVAL!0,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5281  026372  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
5282  026400  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
```

```
5283   026406  001401                          BEQ     4$                      ;YES, CONTINUE
5284   026410  104150                          ERROR   150
5285   026412  023737  004170  004130  4$:     CMP     E.CS2,T.CS2             ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5286   026420  001401                          BEQ     5$                      ;YES, CONTINUE
5287   026422  104151                          ERROR   151
5288   026424  023737  004172  004132  5$:     CMP     E.DS,T.DS               ;CHECK DRIVE STATUS REG. CORRECT
5289   026432  001401                          BEQ     6$                      ;YES, CONTINUE
5290   026434  104152                          ERROR   152
5291   026436  023737  004174  004134  6$:     CMP     E.ER,T.ER               ;CHECK ERROR REGISTER CORRECT
5292   026444  001401                          BEQ     7$                      ;YES, CLEAR RK611
5293   026446  104153                          ERROR   153
5294   026450  013737  004120  004220  7$:     MOV     T.CS1,P.CS1             ;STORE PREVIOUS CONTENTS OF
5295   026456  013737  004130  004222          MOV     T.CS2,P.CS2             ;   COMMAND AND STATUS REG 1
5296   026464  013737  004132  004224          MOV     T.DS,P.DS               ;   COMMAND AND STATUS REG 2
5297   026472  013737  004134  004226          MOV     T.ER,P.ER               ;   DRIVE STATUS REG
5298                                                                           ;   AND ERROR REG
5299   026500  012762  100000  000000          MOV     #CCLR,RKCS1(R2)         ;CLEAR RK611
5300   026506  016237  000000  004120          MOV     RKCS1(R2),T.CS1         ;STORE COMMAND AND STATUS REG 1
5301   026514  016237  000010  004130          MOV     RKCS2(R2),T.CS2         ;STORE COMMAND AND STATUS REG 2
5302   026522  016237  000012  004132          MOV     RKDS(R2),T.DS           ;STORE DRIVE STATUS REG
5303   026530  016237  000014  004134          MOV     RKER(R2),T.ER           ;STORE ERROR REG
5304   026536  012737  000200  004160          MOV     #RDY,E.CS1              ;LOAD EXFECTED CS1
5305   026544  012737  000100  004170          MOV     #IR,E.CS2               ;LOAD EXPECTED CS2
5306   026552  005037  004172                  CLR     E.DS                    ;LOAD EXPECTED DRIVE STATUS REG
5307   026556  005037  004174                  CLR     E.ER                    ;LOAD EXPECTED ERROR REG
5308   026562  023737  004160  004120          CMP     E.CS1,T.CS1             ;CHECK COMMAND AND STATUS REG 1 CORRECT
5309   026570  001401                          BEQ     11$                     ;YES, CHECK CS2
5310   026572  104224                          ERROR   224                     ;CS1 INCORRECT
5311   026574  023737  004170  004130  11$:    CMP     E.CS2,T.CS2             ;CHECK COMMAND AND STATUS REG 2 CORRECT
5312   026602  001401                          BEQ     12$                     ;YES, CHECK DRIVE STATUS REG
5313   026604  104225                          ERROR   225                     ;CS2 INCORRECT
5314   026606  023737  004172  004132  12$:    CMP     E.DS,T.DS               ;CHECK IF DRIVE STATUS REG CORRECT
5315   026614  001401                          BEQ     13$                     ;YES, CHECK ERROR REG
5316   026616  104226                          ERROR   226                     ;ERROR REG INCORRECT
5317   026620  023737  004174  004134  13$:    CMP     E.ER,T.ER               ;CHECK IF ERROR REG CORRECT
5318   026626  001401                          BEQ     TST54                   ;YES, GO ON TO NEXT TEST
5319   026630  10422                           ERROR   227                     ;ERROR REG INCORRECT
5320
5321                                           ;*****************************************************************
5322                                           ;*TEST 54       CDT SET DRIVE TYPE
5323                                           ;*
5324                                           ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5325                                           ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
5326                                           ;*      WITH CDT SET, 26 SECTOR FORMAT, TO CYLINDER 23,
5327                                           ;*      HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
5328                                           ;*      UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE
5329                                           ;*      AND MAKE SURE ONLY DRIVE AVAILIABLE SETS.
5330                                           ;*
5331                                           ;*****************************************************************
5332   026632  000004                  TST54:  SCOPE
5333   026634  012737  000144  001200          MOV     #100.,$TIMES            ;;DO 100. ITERATIONS
5334   026642  013702  001270                  MOV     $BASE,R2                ;LOAD RK611 BASE
5335   026646  012762  000040  000010          MOV     #SCLR,RKCS2(R2)         ;CLEAR RK06 SUBSYSTEM
5336   026654  012762  000040  000026          MOV     #DMD,RKMR1(R2)          ;PUT RK611 IN MAINT MODE
5337   026662  012762  000023  000020          MOV     #23,RKDCYL(R2)          ;LOAD CYLINDER AND
5338   026670  012762  000000  000006          MOV     #0,RKDA(R2)             ;LOAD HEAD ADDRESS
```

```
5339  026676  012762  002017  000000          MOV     #CDT!SEEK,RKCS1(R2)     ;ISSUE CDT!SEEK
5340  026704  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5341  026710  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5342  026716  012762  000040  000026          MOV     #DMD,RKMR1(R2)
5343  026724  005300                          DEC     R0
5344  026726  001370                          BNE     1$
5345  026730  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5346  026734  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5347  026740  105762  000000          2$:     TSTB    RKCS1(R2)
5348  026744  100402                          BMI     3$
5349  026746  005300                          DEC     R0
5350  026750  001373                          BNE     2$
5351  026752  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5352  026760  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5353  026766  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5354  026774  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5355  027002  012737  002216  004160          MOV     #CDT!PDY!CDT!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
5356  027010  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5357  027016  012737  100401  004172          MOV     #SVAL!DRA!DDT,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5358  027024  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
5359  027032  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
5360  027040  001401                          BEQ     4$              ;YES, CONTINUE
5361  027042  104154                          ERROR   154
5362  027044  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5363  027052  001401                          BEQ     5$              ;YES, CONTINUE
5364  027054  104155                          ERROR   155
5365  027056  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
5366  027064  001401                          BEQ     6$              ;YES, CONTINUE
5367  027066  104156                          ERROR   156
5368  027070  023737  004174  004134  6$:     CMP     E.ER,T ER       ;CHECK ERROR REGISTER CORRECT
5369  027076  001401                          BEQ     7$              ;YES, CLEAR RK611
5370  027100  104157                          ERROR   157
5371  027102  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5372  027110  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
5373  027116  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
5374  027124  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
5375                                                                  ;  AND ERROR REG
5376  027132  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5377  027140  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5378  027146  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5379  027154  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5380  027162  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5381  027170  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5382  027176  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5383  027204  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5384  027210  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5385  027214  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5386  027222  001401                          BEQ     11$             ;YES, CHECK CS2
5387  027224  104224                          ERROR   224             ;CS1 INCORRECT
5388  027226  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5389  027234  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5390  027236  104225                          ERROR   225             ;CS2 INCORRECT
5391  027240  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5392  027246  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5393  027250  104226                          ERROR   226             ;ERROR REG INCORRECT
5394  027252  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
```

```
                        BEQ     TST55           ;;YES, GO ON TO NEXT TEST
                        ERROR   227             ;ERROR REG INCORRECT

                ;;*****************************************************************
                ;*TEST 55       CDT SET AND DRIVE TYPE ERROR
                ;*
                ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR
                ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
                ;*      WITH CDT SET, 26 SECTOR FORMAT, TO CYLINDER 2,
                ;*      HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
                ;*      UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE
                ;*      AND MAKE SURE DRIVE AVAILIABLE, DRIVE TYPE ERROR,
                ;*      AND CONTROLLER ERROR SET.
                ;*
                ;*****************************************************************
                TST55:  SCOPE
                        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
                        MOV     $BASE,R2        ;LOAD RK611 BASE
                        MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
                        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
                        MOV     #2,RKDCYL(R2)   ;LOAD CYLINDER AND
                        MOV     #0,RKDA(R2)     ;LOAD HEAD ADDRESS
                        MOV     #CDT!SEEK,RKCS1(R2)     ;ISSUE CDT!SEEK
                        MOV     #22 *4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
                1$:     MOV     #DMD!MCLK,RKMR1(R2)
                        MOV     #DMD,RKMR1(R2)
                        DEC     R0
                        BNE     1$
                        CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
                        MOV     WAITIM,R0       ;WAIT FOR READY
                2$:     TSTB    RKCS1(R2)
                        BMI     3$
                        DEC     R0
                        BNE     2$
                3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
                        MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
                        MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
                        MOV     RKER(R2),T.ER   ;STORE ERROR REG
                        MOV     #CDT!CERR!RDY!CDT!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
                        MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
                        MOV     #SVAL!DRA,E.DS ;LOAD EXPECTED DRIVE STATUS REG
                        MOV     #DTYE,E.ER      ;LOAD EXPECTED ERROR REG
                        CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
                        BEQ     4$              ;YES, CONTINUE
                        ERROR   160
                4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
                        BEQ     5$              ;YES, CONTINUE
                        ERROR   161
                5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
                        BEQ     6$              ;YES, CONTINUE
                        ERROR   162
                6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
                        BEQ     7$              ;YES, CLEAR RK611
                        ERROR   163
                7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
                        MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
```

```
5451  027550  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
5452  027556  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
5453                                                                  ;  AND ERROR REG
5454  027564  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5455  027572  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5456  027600  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5457  027606  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5458  027614  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5459  027622  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5460  027630  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5461  027636  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5462  027642  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5463  027646  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5464  027654  001401                          BEQ     11$             ;YES, CHECK CS2
5465  027656  104224                          ERROR   224             ;CS1 INCORRECT
5466  027660  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5467  027666  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5468  027670  104225                          ERROR   225             ;CS2 INCORRECT
5469  027672  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5470  027700  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5471  027702  104226                          ERROR   226             ;ERROR REG INCORRECT
5472  027704  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5473  027712  001401                          BEQ     TST56           ;YES, GO ON TO NEXT TEST
5474  027714  104227                          ERROR   227             ;ERROR REG INCORRECT
5475
5476                                  ;;*****************************************************************
5477                                  ;*TEST 56       RK06 AND DRIVE TYPE ERROR
5478                                  ;*
5479                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR
5480                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
5481                                  ;*      TO A RK06, 26 SECTOR FORMAT, TO CYLINDER 23,
5482                                  ;*      HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
5483                                  ;*      UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC
5484                                  ;*      MODE AND MAKE SURE DRIVE AVAILIABLE, DRIVE TYPE ERROR,
5485                                  ;*      AND CONTROLLER ERROR SETS.
5486                                  ;*
5487                                  ;*****************************************************************
5488  027716  000004                  TST56:  SCOPE
5489  027720  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5490  027726  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
5491  027732  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5492  027740  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5493  027746  012762  000023  000020          MOV     #23,RKDCYL(R2)  ;LOAD CYLINDER ENO
5494  027754  012762  000000  000006          MOV     #0,RKDA(R2)     ;LOAD HEAD ADDRESS
5495  027762  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
5496  027770  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5497  027774  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5498  030002  012762  000040  000026          MOV     #DMD,RKMR1(R2)
5499  030010  005300                          DEC     R0
5500  030012  001370                          BNE     1$
5501  030014  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5502  030020  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5503  030024  105762  000000          2$:     TSTB    RKCS1(R2)
5504  030030  100402                          BMI     3$
5505  030032  005300                          DEC     R0
5506  030034  001373                          BNE     2$
```

```
 5507  030036  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5508  030044  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5509  030052  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 5510  030060  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 5511  030066  012737  100216  004160          MOV     #CERR!RDY!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
 5512  030074  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 5513  030102  012737  100401  004172          MOV     #SVAL!DRA!DOT,E.DS ;LOAD EXPECTED DRIVE STATUS REG
 5514  030110  012737  000040  004174          MOV     #DTYE,E.ER      ;LOAD EXPECTED ERROR REG
 5515  030116  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
 5516  030124  001401                          BEQ     4$              ;YES, CONTINUE
 5517  030126  104164                          ERROR   164
 5518  030130  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
 5519  030136  001401                          BEQ     5$              ;YES, CONTINUE
 5520  030140  104165                          ERROR   165
 5521  030142  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 5522  030150  001401                          BEQ     6$              ;YES, CONTINUE
 5523  030152  104166                          ERROR   166
 5524  030154  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
 5525  030162  001401                          BEQ     7$              ;YES, CLEAR RK611
 5526  030164  104167                          ERROR   167
 5527  030166  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
 5528  030174  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG 1
 5529  030202  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG 2
 5530  030210  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG
 5531                                                                  ;   AND ERROR REG
 5532  030216  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
 5533  030224  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5534  030232  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5535  030240  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 5536  030246  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 5537  030254  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 5538  030262  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 5539  030270  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
 5540  030274  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
 5541  030300  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
 5542  030306  001401                          BEQ     11$             ;YES, CHECK CS2
 5543  030310  104224                          ERROR   224             ;CS1 INCORRECT
 5544  030312  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
 5545  030320  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
 5546  030322  104225                          ERROR   225             ;CS2 INCORRECT
 5547  030324  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
 5548  030332  001401                          BEQ     13$             ;YES, CHECK ERROR REG
 5549  030334  104226                          ERROR   226             ;ERROR REG INCORRECT
 5550  030336  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
 5551  030344  001401                          BEQ     TST57           ;YES, GO ON TO NEXT TEST
 5552  030346  104227                          ERROR   227             ;ERROR REG INCORRECT
 5553
 5554                                  ;*****************************************************************
 5555                                  ;*TEST 57       SPEED LOSS FROM SHIFT REG.
 5556                                  ;*
 5557                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
 5558                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK06,
 5559                                  ;*      26 SECTOR FORMAT, TO CYLINDER 3, HEAD 1, DRIVE 0.
 5560                                  ;*      CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6. TURN
 5561                                  ;*      OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE AND
 5562                                  ;*      SPEED LOSS ARE SET.
```

```
5563                                        ;*
5564                                        ;;*****************************************************************
5565  030350  000004                        TST57:  SCOPE
5566  030352  012737  000144  001200                MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5567  030360  013702  001270                        MOV     $BASE,R2        ;LOAD RK611 BASE
5568  030364  012762  000040  000010                MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5569  030372  012762  000040  000026                MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5570  030400  012762  000003  000020                MOV     #3,RKDCYL(R2)   ;LOAD CYLINDER AND
5571  030406  012762  000400  000006                MOV     #400,RKDA(R2)   ;LOAD HEAD ADDRESS
5572  030414  012762  000017  000000                MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
5573  030422  012700  000132                        MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5574  030426  012762  000440  000026  1$:           MOV     #DMD!MCLK,RKMR1(R2)
5575  030434  012762  000040  000026                MOV     #DMD,RKMR1(R2)
5576  030442  005300                                DEC     R0
5577  030444  001370                                BNE     1$
5578  030446  005062  000026                        CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5579  030452  013700  004262                        MOV     WAITIM,R0       ;WAIT FOR READY
5580  030456  105762  000000            2$:         TSTB    RKCS1(R2)
5581  030462  100402                                BMI     3$
5582  030464  005300                                DEC     R0
5583  030466  001373                                BNE     2$
5584  030470  016237  000000  004120  3$:           MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5585  030476  016237  000010  004130                MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5586  030504  016237  000012  004132                MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5587  030512  016237  000014  004134                MOV     RKER(R2),T.ER   ;STORE ERROR REG
5588  030520  012737  000216  004160                MOV     #RDY!SEEK&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
5589  030526  012737  000100  004170                MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5590  030534  012737  100021  004172                MOV     #SVAL!DRA!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5591  030542  012737  000000  004174                MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
5592  030550  023737  004160  004120                CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
5593  030556  001401                                BEQ     4$              ;YES, CONTINUE
5594  030560  104170                                ERROR   170
5595  030562  023737  004170  004130  4$:           CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5596  030570  001401                                BEQ     5$              ;YES, CONTINUE
5597  030572  104171                                ERROR   171
5598  030574  023737  004172  004132  5$:           CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
5599  030602  001401                                BEQ     6$              ;YES, CONTINUE
5600  030604  104172                                ERROR   172
5601  030606  023737  004174  004134  6$:           CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
5602  030614  001401                                BEQ     7$              ;YES, CLEAR RK611
5603  030616  104173                                ERROR   173
5604  030620  013737  004120  004220  7$:           MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5605  030626  013737  004130  004222                MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG 1
5606  030634  013737  004132  004224                MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG 2
5607  030642  013737  004134  004226                MOV     T.ER,P.ER       ;   DRIVE STATUS REG
5608                                                                        ;   AND ERROR REG
5609  030650  012762  100000  000000                MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5610  030656  016237  000000  004120                MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5611  030664  016237  000010  004130                MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5612  030672  016237  000012  004132                MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5613  030700  016237  000014  004134                MOV     RKER(R2),T.ER   ;STORE ERROR REG
5614  030706  012737  000200  004160                MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5615  030714  012737  000100  004170                MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5616  030722  005037  004172                        CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5617  030726  005037  004174                        CLR     E.ER            ;LOAD EXPECTED ERROR REG
5618  030732  023737  004160  004120                CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
```

```
5619  030740  001401                          BEQ     11$             ;YES, CHECK CS2
5620  030742  104224                          ERROR   224             ;CS1 INCORRECT
5621  030744  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5622  030752  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5623  030754  104225                          ERROR   225             ;CS2 INCORRECT
5624  030756  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5625  030764  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5626  030766  104226                          ERROR   226             ;ERROR REG INCORRECT
5627  030770  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5628  030776  001401                          BEQ     TST60           ;YES, GO ON TO NEXT TEST
5629  031000  104227                          ERROR   227             ;ERROR REG INCORRECT
5630
5631                                  ;;*****************************************************************
5632                                  ;*TEST 60       DRIVE OFF TRACK FROM SHIFT REG.
5633                                  ;*
5634                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5635                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK06,
5636                                  ;*      26 SECTOR FORMAT, TO CYLINDER 3, HEAD 2, DRIVE 0.
5637                                  ;*      CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
5638                                  ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE
5639                                  ;*      AND DRIVE OFF TRACK ARE SET.
5640                                  ;*
5641                                  ;*****************************************************************
5642  031002  000004                  TST60:  SCOPE
5643  031004  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5644  031012  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
5645  031016  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5646  031024  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5647  031032  012762  000003  000020          MOV     #3,RKDCYL(R2)   ;LOAD CYLINDER AND
5648  031040  012762  001000  000006          MOV     #1000,RKDA(R2)  ;LOAD HEAD ADDRESS
5649  031046  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK
5650  031054  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5651  031060  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5652  031066  012762  000040  000026          MOV     #DMD,RKMR1(R2)
5653  031074  005300                          DEC     R0
5654  031076  001370                          BNE     1$
5655  031100  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5656  031104  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5657  031110  105762  000000          2$:     TSTB    RKCS1(R2)
5658  031114  100402                          BMI     3$
5659  031116  005300                          DEC     R0
5660  031120  001373                          BNE     2$
5661  031122  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5662  031130  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5663  031136  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5664  031144  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5665  031152  012737  000216  004160          MOV     #RDY!SEEK&^C<GO>),E.CS1 ;LOAD EXPECTED CS1
5666  031160  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5667  031166  012737  100041  004172          MOV     #SVAL!DRA!DROT,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5668  031174  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
5669  031202  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
5670  031210  001401                          BEQ     4$              ;YES, CONTINUE
5671  031212  104174                          ERROR   174
5672  031214  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5673  031222  001401                          BEQ     5$              ;YES, CONTINUE
5674  031224  104175                          ERROR   175
```

```
5675  031226  023737  004172  004134  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
5676  031234  001401                          BEQ     6$              ;YES, CONTINUE
5677  031236  104176                          ERROR   176
5678  031240  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
5679  031246  001401                          BEQ     7$              ;YES, CLEAR RK611
5680  031250  104177                          ERROR   177
5681  031252  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5682  031260  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
5683  031266  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
5684  031274  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
5685                                                                  ;  AND ERROR REG
5686  031302  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5687  031310  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5688  031316  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5689  031324  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5690  031332  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5691  031340  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5692  031346  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5693  031354  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5694  031360  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5695  031364  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5696  031372  001401                          BEQ     11$             ;YES, CHECK CS2
5697  031374  104224                          ERROR   224             ;CS1 INCORRECT
5698  031376  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5699  031404  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5700  031406  104225                          ERROR   225             ;CS2 INCORRECT
5701  031410  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5702  031416  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5703  031420  104226                          ERROR   226             ;ERROR REG INCORRECT
5704  031422  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5705  031430  001401                          BEQ     TST61           ;YES, GO ON TO NEXT TEST
5706  031432  104227                          ERROR   227             ;ERROR REG INCORRECT
5707
5708                                  ;*****************************************************************
5709                                  ;*TEST 61       WRITE LOCK ERROR FROM SHIFT REG.
5710                                  ;*
5711                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5712                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A PACK ACKNOWLEDGE
5713                                  ;*      TO A RK06, 26 SECTOR FORMAT, WITH CYLINDER 0,
5714                                  ;*      HEAD 1, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL
5715                                  ;*      PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
5716                                  ;*      SURE SPEED LOSS, WRITE LOCK ERROR AND CONTROLLER ERROR
5717                                  ;*      ARE SET WITH DRIVE AVAILIABLE RESET.
5718                                  ;*
5719                                  ;*****************************************************************
5720  031434  000004                  TST61:  SCOPE
5721  031436  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
5722  031444  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
5723  031450  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5724  031456  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
5725  031464  012762  000000  000020          MOV     #0,RKDCYL(R2)   ;LOAD CYLINDER AND
5726  031472  012762  000400  000006          MOV     #400,RKDA(R2)   ;LOAD HEAD ADDRESS
5727  031500  012762  000003  000000          MOV     #PACK,RKCS1(R2) ;ISSUE PACK
5728  031506  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5729  031512  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
5730  031520  012762  000040  000026          MOV     #DMD,RKMR1(R2)
```

```
5731  031526  005300                          DEC     R0
5732  031530  001370                          BNE     1$
5733  031532  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
5734  031536  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
5735  031542  105762  000000          2$:     TSTB    RKCS1(R2)
5736  031546  100402                          BMI     3$
5737  031550  005300                          DEC     R0
5738  031552  001373                          BNE     2$
5739  031554  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5740  031562  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5741  031570  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5742  031576  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5743  031604  012737  100202  004160          MOV     #CERR!RDY!PACK&^C<GO>,E.CS1 ;LOAD EXPECTED CS1
5744  031612  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5745  031620  012737  100020  004172          MOV     #SVAL!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5746  031626  012737  004000  004174          MOV     #WLE,E.ER       ;LOAD EXPECTED ERROR REG
5747  031634  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
5748  031642  001401                          BEQ     4$              ;YES, CONTINUE
5749  031644  104200                          ERROR   200
5750  031646  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5751  031654  001401                          BEQ     5$              ;YES, CONTINLE
5752  031656  104201                          ERROR   201
5753  031660  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
5754  031666  001401                          BEQ     6$              ;YES, CONTINUE
5755  031670  104202                          ERROR   202
5756  031672  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
5757  031700  001401                          BEQ     7$              ;YES, CLEAR RK611
5758  031702  104203                          ERROR   203
5759  031704  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
5760  031712  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
5761  031720  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
5762  031726  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
5763                                                                  ;  AND ERROR REG
5764  031734  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5765  031742  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5766  031750  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5767  031756  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5768  031764  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5769  031772  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
5770  032000  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
5771  032006  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
5772  032012  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
5773  032016  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
5774  032024  001401                          BEQ     11$             ;YES, CHECK CS2
5775  032026  104224                          ERROR   224             ;CS1 INCORRECT
5776  032030  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
5777  032036  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
5778  032040  104225                          ERROR   225             ;CS2 INCORRECT
5779  032042  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
5780  032050  001401                          BEQ     13$             ;YES, CHECK ERROR REG
5781  032052  104226                          ERROR   226             ;ERROR REG INCORRECT
5782  032054  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
5783  032062  001401                          BEQ     TST62           ;YES, GO ON TO NEXT TEST
5784  032064  104227                          ERROR   227             ;ERROR REG INCORRECT
5785
5786                          ;;*****************************************************************
```

```
                                        ;*TEST 62       SEEK INCOMPLETE
                                        ;*
                                        ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                        ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE AN UNLOAD
                                        ;*      TO A RK06, 26 SECTOR FORMAT, WITH CYLINDER 0,
                                        ;*      HEAD 1, DRIVE 0, CLOCK IN DIAGNOSTIC MODE UNTIL
                                        ;*      PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
                                        ;*      SURE SPEED LOSS, SEEK INCOMPLETE, AND CONTROLLER ERROR
                                        ;*      ARE SET WITH DRIVE AVAILIABLE RESET.
                                        ;*
                                        ;**************************************************************
032066  000004                    TST62:  SCOPE
032070  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
032076  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
032102  012762  000040  000010            MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
032110  012762  000040  000026            MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
032116  012762  000000  000020            MOV     #0,RKDCYL(R2)   ;LOAD CYLINDER AND
032124  012762  000400  000006            MOV     #400,RKDA(R2)   ;LOAD HEAD ADDRESS
032132  012762  000007  000000            MOV     #UNLOAD,RKCS1(R2)       ;ISSUE UNLOAD
032140  012700  000132                    MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
032144  012762  000440  000026    1$:     MOV     #DMD!MCLK,RKMR1(R2)
032152  012762  000040  000026            MOV     #DMD,RKMR1(R2)
032160  005300                            DEC     R0
032162  001370                            BNE     1$
032164  005062  000026                    CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
032170  013700  004262                    MOV     WAITIM,R0       ;WAIT FOR READY
032174  105762  000000            2$:     TSTB    RKCS1(R2)
032200  100402                            BMI     3$
032202  005300                            DEC     R0
032204  001373                            BNE     2$
032206  016237  000000  004120    3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
032214  016237  000010  004130            MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
032222  016237  000012  004132            MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
032230  016237  000014  004134            MOV     RKER(R2),T.ER   ;STORE ERROR REG
032236  012737  '00206  004160            MOV     #CERR!RDY!UNLOAD&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
032244  012737  000100  004170            MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
032252  012737  100020  004172            MOV     #SVAL!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
032260  012737  000002  004174            MOV     #SKI,E.ER       ;LOAD EXPECTED ERROR REG
032266  023737  004160  004120            CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
032274  001401                            BEQ     4$              ;YES, CONTINUE
032276  104204                            ERROR   204
032300  023737  004170  004130    4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
032306  001401                            BEQ     5$              ;YES, CONTINUE
032310  104205                            ERROR   205
032312  023737  004172  004132    5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
032320  001401                            BEQ     6$              ;YES, CONTINUE
032322  104206                            ERROR   206
032324  023737  004174  004134    6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
032332  001401                            BEQ     7$              ;YES, CLEAR RK611
032334  104207                            ERROR   207
032336  013737  004120  004220    7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
032344  013737  004130  004222            MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
032352  013737  004132  004224            MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
032360  013737  004134  004226            MOV     T.ER,P.ER       ;  DRIVE STATUS REG
                                                                  ;  AND ERROR REG
032366  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
```

```
 5843  032374  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5844  032402  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5845  032410  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 5846  032416  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 5847  032424  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 5848  032432  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 5849  032440  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
 5850  032444  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
 5851  032450  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
 5852  032456  001401                          BEQ     11$             ;YES, CHECK CS2
 5853  032460  104224                          ERROR   224             ;CS1 INCORRECT
 5854  032462  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
 5855  032470  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
 5856  032472  104225                          ERROR   225             ;CS2 INCORRECT
 5857  032474  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
 5858  032502  001401                          BEQ     13$             ;YES, CHECK ERROR REG
 5859  032504  104226                          ERROR   226             ;ERROR REG INCORRECT
 5860  032506  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
 5861  032514  001401                          BEQ     TST63           ;YES, GO ON TO NEXT TEST
 5862  032516  104227                          ERROR   227             ;ERROR REG INCORRECT
 5863
 5864                                  ;;**********************************************************
 5865                                  ;*TEST 63       NON-EXECUTABLE DRIVE FUNCTION FROM SHIFT REG.
 5866                                  ;*
 5867                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR
 5868                                  ;*      PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE
 5869                                  ;*      A DRIVE CLEAR TO A RK06, 26 SECTOR FORMAT,
 5870                                  ;*      WITH CYLINDER 0, HEAD 1, DRIVE 0. CLOCK IN DIAGNOSTIC
 5871                                  ;*      MODE UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC
 5872                                  ;*      MODE AND MAKE SURE SPEED LOSS, NON-EXECUTABLE DRIVE FUNCTION, AND
 5873                                  ;*      CONTROLLER ERROR ARE SET WITH DRIVE AVAILIABLE RESET.
 5874                                  ;*
 5875                                  ;;**********************************************************
 5876  032520  000004                  TST63:  SCOPE
 5877  032522  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 5878  032530  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 5879  032534  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
 5880  032542  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
 5881  032550  012762  000000  000020          MOV     #0,RKDCYL(R2)   ;LOAD CYLINDER AND
 5882  032556  012762  000400  000006          MOV     #400,RKDA(R2)   ;LOAD HEAD ADDRESS
 5883  032564  012762  000005  000000          MOV     #CLEAR,RKCS1(R2)        ;ISSUE CLEAR
 5884  032572  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
 5885  032576  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 5886  032604  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 5887  032612  005300                          DEC     R0
 5888  032614  001370                          BNE     1$
 5889  032616  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
 5890  032622  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
 5891  032626  105762  000000          2$:     TSTB    RKCS1(R2)
 5892  032632  100402                          BMI     3$
 5893  032634  005300                          DEC     R0
 5894  032636  001373                          BNE     2$
 5895  032640  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5896  032646  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5897  032654  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 5898  032662  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
```

NON-EXECUTABLE DRIVE FUNCTION FROM SHIFT REG.

```
5899  032670  012737  100204  004160          MOV     #CERR!RDY!CLEAR&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
5900  032676  012737  000100  004170          MOV     #IR,E.CS2               ;LOAD EXPECTED CS2
5901  032704  012737  100020  004172          MOV     #SVAL!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5902  032712  012737  000004  004174          MOV     #NXF,E.ER               ;LOAD EXPECTED ERROR REG
5903  032720  023737  004160  004120          CMP     E.CS1,T.CS1             ;CHECK COMMAND AND STATUS REG.1 CORRECT
5904  032726  001401                          BEQ     4$                      ;YES, CONTINUE
5905  032730  104210                          ERROR   210
5906  032732  023737  004170  004130  4$:     CMP     E.CS2,T.CS2             ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5907  032740  001401                          BEQ     5$                      ;YES, CONTINUE
5908  032742  104211                          ERROR   211
5909  032744  023737  004172  004132  5$:     CMP     E.DS,T.DS               ;CHECK DRIVE STATUS REG. CORRECT
5910  032752  001401                          BEQ     6$                      ;YES, CONTINUE
5911  032754  104212                          ERROR   212
5912  032756  023737  004174  004134  6$:     CMP     E.ER,T.ER               ;CHECK ERROR REGISTER CORRECT
5913  032764  001401                          BEQ     7$                      ;YES, CLEAR RK611
5914  032766  104213                          ERROR   213
5915  032770  013737  004120  004220  7$:     MOV     T.CS1,P.CS1             ;STORE PREVIOUS CONTENTS OF
5916  032776  013737  004130  004222          MOV     T.CS2,P.CS2             ;   COMMAND AND STATUS REG 1
5917  033004  013737  004132  004224          MOV     T.DS,P.DS               ;   COMMAND AND STATUS REG 2
5918  033012  013737  004134  004226          MOV     T.ER,P.ER               ;   DRIVE STATUS REG
5919                                                                          ;   AND ERROR REG
5920  033020  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
5921  033026  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5922  033034  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5923  033042  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
5924  033050  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
5925  033056  012737  000200  004160          MOV     #RDY,E.CS1              ;LOAD EXPECTED CS1
5926  033064  012737  000100  004170          MOV     #IR,E.CS2               ;LOAD EXPECTED CS2
5927  033072  005037  004172                  CLR     E.DS                    ;LOAD EXPECTED DRIVE STATUS REG
5928  033076  005037  004174                  CLR     E.ER                    ;LOAD EXPECTED ERROR REG
5929  033102  023737  004160  004120          CMP     E.CS1,T.CS1             ;CHECK COMMAND AND STATUS REG 1 CORRECT
5930  033110  001401                          BEQ     11$                     ;YES, CHECK CS2
5931  033112  104224                          ERROR   224                     ;CS1 INCORRECT
5932  033114  023737  004170  004130  11$:    CMP     E.CS2,T.CS2             ;CHECK COMMAND AND STATUS REG 2 CORRECT
5933  033122  001401                          BEQ     12$                     ;YES, CHECK DRIVE STATUS REG
5934  033124  104225                          ERROR   225                     ;CS2 INCORRECT
5935  033126  023737  004172  004132  12$:    CMP     E.DS,T.DS               ;CHECK IF DRIVE STATUS REG CORRECT
5936  033134  001401                          BEQ     13$                     ;YES, CHECK ERROR REG
5937  033136  104226                          ERROR   226                     ;ERROR REG INCORRECT
5938  033140  023737  004174  004134  13$:    CMP     E.ER,T.ER               ;CHECK IF ERROR REG CORRECT
5939  033146  001401                          BEQ     TST64                   ;YES, GO ON TO NEXT TEST
5940  033150  104227                          ERROR   227                     ;ERROR REG INCORRECT
5941
5942                                  ;*****************************************************************
5943                                  ;TEST 64        AC LOW AND C-D PARITY FROM SHIFT REG.
5944                                  ;*
5945                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR, PUT RK611
5946                                  ;*      CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A START SPINDLE
5947                                  ;*      TO AN RK06, IN 24 SECTOR FORMAT, CYLINDER 0, HEAD 0,
5948                                  ;*      DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
5949                                  ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE AC LOW, DRIVE
5950                                  ;*      DETECTED SERCOM PARITY, AND CONTROLLER ERROR SET WITH
5951                                  ;*      DRIVE AVAILABLE RESET.
5952                                  ;*
5953                                  ;*****************************************************************
5954  033152  000004                  TST64:  SCOPE
```

```
 5955  033154  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 5956  033162  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 5957  033166  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
 5958  033174  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
 5959  033202  012762  010011  000000          MOV     #SRTSPL!CFMT,RKCS1(R2)  ;ISSUE SRTSPL!CFMT
 5960  033210  012700  000132                  MOV     #22,#4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
 5961  033214  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 5962  033222  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 5963  033230  005300                          DEC     R0
 5964  033232  001370                          BNE     1$
 5965  033234  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
 5966  033240  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
 5967  033244  105762  000000          2$:     TSTB    RKCS1(R2)
 5968  033250  100402                          BMI     3$
 5969  033252  005300                          DEC     R0
 5970  033254  001373                          BNE     2$
 5971  033256  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5972  033264  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5973  033272  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 5974  033300  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 5975  033306  012737  110210  004160          MOV     #CERR!CFMT!RDY!SRTSPL!CFMT&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
 5976  033314  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 5977  033322  012737  100010  004172          MOV     #SVAL!ACLO,E.DS ;LOAD EXPECTED DRIVE STATUS REG
 5978  033330  012737  000010  004174          MOV     #DRPAR,E.ER     ;LOAD EXPECTED ERROR REG
 5979  033336  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
 5980  033344  001401                          BEQ     4$              ;YES, CONTINUE
 5981  033346  104214                          ERROR   214
 5982  033350  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
 5983  033356  001401                          BEQ     5$              ;YES, CONTINUE
 5984  033360  104215                          ERROR   215
 5985  033362  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 5986  033370  001401                          BEQ     6$              ;YES, CONTINUE
 5987  033372  104216                          ERROR   216
 5988  033374  023737  004174  004134  6$:     CMP     E.ER,T ER       ;CHECK ERROR REGISTER CORRECT
 5989  033402  001401                          BEQ     7$              ;YES, CLEAR RK611
 5990  033404  104217                          ERROR   217
 5991  033406  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
 5992  033414  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG 1
 5993  033422  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG 2
 5994  033430  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG
 5995                                                                  ;   AND ERROR REG
 5996  033436  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
 5997  033444  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
 5998  033452  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
 5999  033460  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 6000  033466  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6001  033474  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 6002  033502  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 6003  033510  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
 6004  033514  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
 6005  033520  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
 6006  033526  001401                          BEQ     11$             ;YES, CHECK CS2
 6007  033530  104224                          ERROR   224             ;CS1 INCORRECT
 6008  033532  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORREC
 6009  033540  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
 6010  033542  104225                          ERROR   225             ;CS2 INCORRECT
```

```
6011  033544  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
6012  033552  001401                          BEQ     13$             ;YES, CHECK ERROR REG
6013  033554  104226                          ERROR   226             ;ERROR REG INCORRECT
6014  033556  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
6015  033564  001401                          BEQ     TST65           ;YES, GO ON TO NEXT TEST
6016  033566  104227                          ERROR   227             ;ERROR REG INCORRECT
6017
6018                                          ;;*****************************************************************
6019                                          ;*TEST 65       ILLEGAL DISK ADDRESS ERROR FROM SHIFT REG.
6020                                          ;*
6021                                          ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.   PUT
6022                                          ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A RECALIBRATE
6023                                          ;*      TO AN RK06, IN 26 SECTOR FORMAT, CYLINDER 0, HEAD 1,
6024                                          ;*      DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE
6025                                          ;*      ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE
6026                                          ;*      SPEED LOSS, ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER
6027                                          ;*      ERROR ARE SET WITH DRIVE AVAILABLE RESET.
6028                                          ;*
6029                                          ;;*****************************************************************
6030  033570  000004                  TST65:  SCOPE
6031  033572  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
6032  033600  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
6033  033604  012762  000040  000C10          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6034  033612  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINT MODE
6035  033620  012762  000000  000020          MOV     #0,RKDCYL(R2)   ;LOAD CYLINDER AND
6036  033626  012762  000400  000006          MOV     #400,RKDA(R2)   ;LOAD HEAD ADDRESS
6037  033634  012762  000013  000000          MOV     #RECAL,RKCS1(R2)        ;ISSUE RECAL
6038  033642  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
6039  033646  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
6040  033654  012762  000040  000026          MOV     #DMD,RKMR1(R2)
6041  033662  005300                          DEC     R0
6042  033664  001370                          BNE     1$
6043  033666  005062  000026                  CLR     RKMR1(R2)       ;FINISH COMMAND IN NORMAL MODE
6044  033672  013700  004262                  MOV     WAITIM,R0       ;WAIT FOR READY
6045  033676  105762  000000          2$:     TSTB    RKCS1(R2)
6046  033702  100402                          BMI     3$
6047  033704  005300                          DEC     R0
6048  033706  001373                          BNE     2$
6049  033710  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
6050  033716  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
6051  033724  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
6052  033732  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
6053  033740  012737  100212  004160          MOV     #CERR!RDY!RECAL&<^C<GO>>,E.CS1 ;LOAD EXPECTED CS1
6054  033746  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
6055  033754  012737  100020  004172          MOV     #SVAL!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
6056  033762  012737  002000  004174          MOV     #IDAE,E.ER      ;LOAD EXPECTED ERROR REG
6057  033770  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
6058  033776  001401                          BEQ     4$              ;YES, CONTINUE
6059  034000  104220                          ERROR   220
6060  034002  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
6061  034010  001401                          BEQ     5$              ;YES, CONTINUE
6062  034012  104221                          ERROR   221
6063  034014  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
6064  034022  001401                          BEQ     6$              ;YES, CONTINUE
6065  034024  104222                          ERROR   222
6066  034026  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
```

```
6067  034034  001401                          BEQ     7$              ;YES, CLEAR RK611
6068  034036  104223                          ERROR   223
6069  034040  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS CONTENTS OF
6070  034046  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG 1
6071  034054  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG 2
6072  034062  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG
6073                                                                  ;  AND ERROR REG
6074  034070  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
6075  034076  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
6076  034104  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
6077  034112  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
6078  034120  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
6079  034126  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
6080  034134  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
6081  034142  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG
6082  034146  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG
6083  034152  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
6084  034160  001401                          BEQ     11$             ;YES, CHECK CS2
6085  034162  104224                          ERROR   224             ;CS1 INCORRECT
6086  034164  023737  004170  004130  11$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG 2 CORRECT
6087  034172  001401                          BEQ     12$             ;YES, CHECK DRIVE STATUS REG
6088  034174  104225                          ERROR   225             ;CS2 INCORRECT
6089  034176  023737  004172  004132  12$:    CMP     E.DS,T.DS       ;CHECK IF DRIVE STATUS REG CORRECT
6090  034204  001401                          BEQ     13$             ;YES, CHECK ERROR REG
6091  034206  104226                          ERROR   226             ;ERROR REG INCORRECT
6092  034210  023737  004174  004134  13$:    CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
6093  034216  001401                          BEQ     TST66           ;YES, GO ON TO NEXT TEST
6094  034220  104227                          ERROR   227             ;ERROR REG INCORRECT
6095
6096                                  ;;*****************************************************************
6097                                  ;*TEST 66        IDAE DETECTION IN RK611 CONTROLLER (PART 1)
6098                                  ;*
6099                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
6100                                  ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A
6101                                  ;*      SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 1003.
6102                                  ;*      HEAD 0, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL
6103                                  ;*      PHASE ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE
6104                                  ;*      SURE DRIVE AVAILABLE, ILLEGAL DISK ADDRESS ERROR,
6105                                  ;*      AND CONTROLLER ERROR ARE SET.
6106                                  ;*
6107                                  ;;*****************************************************************
6108  034222  000004                  TST66:  SCOPE
6109  034224  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
6110  034232  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
6111  034236  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6112  034244  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
6113  034252  012762  001002  000020          MOV     #1002,RKDCYL(R2)        ;LOAD CYLINDER ADDRESS
6114  034260  012737  001002  004252          MOV     #1002,CYLIN
6115  034266  012737  000000  004250          MOV     #0,HDCODE       ;LOAD HEAD ADDRESS
6116  034274  005046                          CLR     -(SP)
6117  034276  113766  004250  000001          MOVB    HDCODE,1(SP)
6118  034304  012662  000006                  MOV     (SP)+,RKDA(R2)
6119  034310  012737  000006  004266          MOV     #6,DRVTYP       ;LOAD DRIVE TYPE FOR PRINT OUT
6120  034316  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK TO RK06
6121  034324  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCK TO GET THROUGH PHASE 6
6122  034330  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
```

```
 6123  034336  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 6124  034344  005300                          DEC     R0
 6125  034346  001370                          BNE     1$
 6126  034350  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
 6127  034354  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
 6128  034360  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
 6129  034364  100402                          BMI     3$
 6130  034366  005300                          DEC     R0
 6131  034370  001373                          BNE     2$
 6132  034372  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6133  034400  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6134  034406  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 6135  034414  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6136  034422  012737  100216  004160          MOV     #CERR!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD E:PECTED CS1
 6137
 6138  034430  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG.2
 6139  034436  012737  100001  004172          MOV     #SVAL!DRA,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
 6140  034444  012737  002000  004174          MOV     #IDAE,E.ER      ;LOAD EXPECTED ERROR REG
 6141  034452  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
 6142  034460  001401                          BEQ     4$              ;YES, CHECK CS2
 6143  034462  104230                          ERROR   230             ;CS1 INCORRECT
 6144  034464  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6145  034472  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
 6146  034474  104231                          ERROR   231             ;CS2 INCORRECT
 6147  034476  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 6148  034504  001401                          BEQ     6$              ;YES, CHECK ERROR REG
 6149  034506  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
 6150  034510  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
 6151  034516  001401                          BEQ     7$              ;YES, CHECK CONTROLLER CLEAR
 6152  034520  104233                          ERROR   233             ;ERROR REG. INCORRECT
 6153  034522  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS VALUES OF
 6154  034530  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG.1
 6155  034536  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG.2
 6156  034544  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG.
 6157                                          ;   ERROR REG.
 6158  034552  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
 6159  034560  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6160  034566  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6161  034574  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
 6162  034602  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6163  034610  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 6164  034616  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 6165  034624  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
 6166  034630  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG.
 6167  034634  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORPECT
 6168  034642  001401                          BEQ     10$             ;YES, CHECK CS2
 6169  034644  104224                          ERROR   224             ;CS1 INCORRECT
 6170  034646  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6171  034654  001401                          BEQ     11$             ;YES, CHECK DRIVE STATUS REG
 6172  034656  104225                          ERROR   225             ;CS2 INCORRECT
 6173  034660  023737  004172  004132  11$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
 6174  034666  001401                          BEQ     12$             ;YES, CHECK ERROR REGISTER
 6175  034670  104226                          ERROR   226             ;DRIVE STATUS REG INCORRECT
 6176  034672  023737  004174  004134  12$:    CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
 6177  034700  001401                          BEQ     TST67           ;YES,GO ON TO NEXT TEST
 6178  034702  104227                          ERROR   227             ;ERROR REG. INCORRECT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 67       IDAE DETECTION IN RK611 CONTROLLER (PART 2)
                                        ;*
                                        ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
                                        ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
                                        ;*      WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 1022, HEAD
                                        ;*      0, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE
                                        ;*      ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE
                                        ;*      DRIVE AVAILABLE AND POSITIONING IN PROGRESS ARE SET
                                        ;*      WITH ILLEGAL DISK ADDRESS ERROR RESET.
                                        ;*
                                        ;;*****************************************************************
034704  000004                  TST67:  SCOPE
034706  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
034714  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
034720  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
034726  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
034734  012762  001022  000020          MOV     #1022,RKDCYL(R2)        ;LOAD CYLINDER ADDRESS
034742  012737  001022  004252          MOV     #1022,CYLIN
034750  012737  000000  004250          MOV     #0,HDCODE       ;LOAD HEAD ADDRESS
034756  005046                          CLR     -(SP)
034760  113766  004250  000001          MOVB    HDCODE,1(SP)
034766  012662  000006                  MOV     (SP)+,RKDA(R2)
034772  012737  000007  004266          MOV     #7,DRVTYP       ;LOAD DRIVE TYPE FOR PRINT OUT
035000  012762  002017  000000          MOV     #CDT!SEEK,RKCS1(R2)  ;ISSUE SEEK TO RK06
035006  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCK TO GET THROUGH PHASE 6
035012  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
035020  012762  000040  000026          MOV     #DMD,RKMR1(R2)
035026  005300                          DEC     R0
035030  001370                          BNE     1$
035032  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
035036  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
035042  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
035046  100402                          BMI     3$
035050  005300                          DEC     R0
035052  001373                          BNE     2$
035054  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
035062  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
035070  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
035076  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
035104  012737  002216  004160          MOV     #CDT!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD EXPECTED CS1

035112  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG.2
035120  012737  120401  004172          MOV     #SVAL!DRA!PIP!DDT,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
035126  012737  000000  004174          MOV     #0,E.ER ;LOAD EXPECTED ERROR REG
035134  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
035142  001401                          BEQ     4$              ;YES, CHECK CS2
035144  104230                          ERROR   230             ;CS1 INCORRECT
035146  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
035154  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
035156  104231                          ERROR   231             ;CS2 INCORRECT
035160  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
035166  001401                          BEQ     6$              ;YES, CHECK ERROR REG
035170  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
035172  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
```

```
6235  035200  001401                          BEQ     7$                ;YES, CHECK CONTROLLER CLEAR
6236  035202  104233                          ERROR   233               ;ERROR REG. INCORRECT
6237  035204  013737  004120  004220  7$:     MOV     T.CS1,P.CS1       ;STORE PREVIOUS VALUES OF
6238  035212  013737  004130  004222          MOV     T.CS2,P.CS2       ;  COMMAND AND STATUS REG.1
6239  035220  013737  004132  004224          MOV     T.DS,P.DS         ;  COMMAND AND STATUS REG.2
6240  035226  013737  004134  004226          MOV     T.ER,P.ER         ;  DRIVE STATUS REG.
6241                                                                    ;  ERROR REG.
6242  035234  012762  100000  000000          MOV     #CCLR,RKCS1(R2)   ;ISSUE CONTROLLER CLEAR
6243  035242  016237  000000  004120          MOV     RKCS1(R2),T.CS1   ;STORE COMMAND AND STATUS REG.1
6244  035250  016237  000010  004130          MOV     RKCS2(R2),T.CS2   ;STORE COMMAND AND STATUS REG.2
6245  035256  016237  000012  004132          MOV     RKDS(R2),T.DS     ;STORE DRIVE STATUS REG.
6246  035264  016237  000014  004134          MOV     RKER(R2),T.ER     ;STORE ERROR REG
6247  035272  012737  000200  004160          MOV     #RDY,E.CS1        ;LOAD EXPECTED CS1
6248  035300  012737  000100  004170          MOV     #IR,E.CS2         ;LOAD EXPECTED CS2
6249  035306  005037  004172                  CLR     E.DS              ;LOAD EXPECTED DRIVE STATUS REG.
6250  035312  005037  004174                  CLR     E.ER              ;LOAD EXPECTED ERROR REG.
6251  035316  023737  004160  004120          CMP     E.CS1,T.CS1       ;CHECK COMMAND AND STATUS REG.1 CORRECT
6252  035324  001401                          BEQ     10$               ;YES, CHECK CS2
6253  035326  104224                          ERROR   224               ;CS1 INCORRECT
6254  035330  023737  004170  004130  10$:    CMP     E.CS2,T.CS2       ;CHECK COMMAND AND STATUS REG2 CORRECT
6255  035336  001401                          BEQ     11$               ;YES, CHECK DRIVE STATUS REG
6256  035340  104225                          ERROR   225               ;CS2 INCORRECT
6257  035342  023737  004172  004132  11$:    CMP     E.DS,T.DS         ;CHECK DRIVE STATUS REG CORRECT
6258  035350  001401                          BEQ     12$               ;YES, CHECK ERROR REGISTER
6259  035352  104226                          ERROR   226               ;DRIVE STATUS REG INCORRECT
6260  035354  023737  004174  004134  12$:    CMP     E.ER,T.ER         ;CHECK ERROR REG CORRECT
6261  035362  001401                          BEQ     TST70             ;;YES,GO ON TO NEXT TEST
6262  035364  104227                          ERROR   227               ;ERROR REG. INCORRECT
6263
6264                                  ;;*****************************************************************
6265                                  ;*TEST 70        IDAE DETECTION IN RK611 CONTROLLER (PART 3)
6266                                  ;*
6267                                  ;*       CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
6268                                  ;*       RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
6269                                  ;*       TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2,
6270                                  ;*       HEAD 3, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL
6271                                  ;*       PHASE ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE
6272                                  ;*       SURE DRIVE AVAILABLE, DRIVE OFF TRACK, SPEED LOSS,
6273                                  ;*       ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER ERROR ARE
6274                                  ;*       SET.
6275                                  ;*
6276                                  ;*****************************************************************
6277  035366  000004                  TST70:  SCOPE
6278  035370  012737  000144  001200          MOV     #100.,$TIMES      ;;DO 100. ITERATIONS
6279  035376  013702  001270                  MOV     $BASE,R2          ;LOAD RK611 BASE
6280  035402  012762  000040  000010          MOV     #SCLR,RKCS2(R2)   ;CLEAR RK06 SUBSYSTEM
6281  035410  012762  000040  000026          MOV     #DMD,RKMR1(R2)    ;PUT RK611 IN MAINTENANCE MODE
6282  035416  012762  000002  000020          MOV     #2,RKDCYL(R2)     ;LOAD CYLINDER ADDRESS
6283  035424  012737  000002  004252          MOV     #2,CYLIN
6284  035432  012737  000003  004250          MOV     #3,HDCODE         ;LOAD HEAD ADDRESS
6285  035440  005046                          CLR     -(SP)
6286  035442  113766  004250  000001          MOVB    HDCODE,1(SP)
6287  035450  012662  000006                  MOV     (SP)+,RKDA(R2)
6288  035454  012737  000006  004266          MOV     #6,DRVTYP         ;LOAD DRIVE TYPE FOR PRINT OUT
6289  035462  012762  000017  000000          MOV     #SEEK,RKCS1(R2)   ;ISSUE SEEK TO RK06
6290  035470  012700  000132                  MOV     #22.*4+2,R0       ;ISSUE CLOCK TO GET THROUGH PHASE 6
```

```
 6291  035474  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 6292  035502  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 6293  035510  005300                          DEC     R0
 6294  035512  001370                          BNE     1$
 6295  035514  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
 6296  035520  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
 6297  035524  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
 6298  035530  100402                          BMI     3$
 6299  035532  005300                          DEC     R0
 6300  035534  001373                          BNE     2$
 6301  035536  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6302  035544  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6303  035552  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 6304  035560  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6305  035566  012737  100216  004160          MOV     #CERR!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD EXPECTED CS1
 6306
 6307  035574  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG.2
 6308  035602  012737  100061  004172          MOV     #SVAL!DRA!DROT!SPDLSS,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
 6309  035610  012737  002000  004174          MOV     #IDAE,E.ER      ;LOAD EXPECTED ERROR REG
 6310  035616  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
 6311  035624  001401                          BEQ     4$              ;YES, CHECK CS2
 6312  035626  104230                          ERROR   230             ;CS1 INCORRECT
 6313  035630  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6314  035636  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
 6315  035640  104231                          ERROR   231             ;CS2 INCORRECT
 6316  035642  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 6317  035650  001401                          BEQ     6$              ;YES, CHECK ERROR REG
 6318  035652  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
 6319  035654  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
 6320  035662  001401                          BEQ     7$              ;YES, CHECK CONTROLLER CLEAR
 6321  035664  104233                          ERROR   233             ;ERROR REG. INCORRECT
 6322  035666  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS VALUES OF
 6323  035674  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG.1
 6324  035702  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG.2
 6325  035710  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG.
 6326                                          ;   ERROR REG.
 6327  035716  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
 6328  035724  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6329  035732  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6330  035740  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
 6331  035746  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6332  035754  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 6333  035762  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 6334  035770  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
 6335  035774  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG.
 6336  036000  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
 6337  036006  001401                          BEQ     10$             ;YES, CHECK CS2
 6338  036010  104224                          ERROR   224             ;CS1 INCORRECT
 6339  036012  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6340  036020  001401                          BEQ     11$             ;YES, CHECK DRIVE STATUS REG
 6341  036022  104225                          ERROR   225             ;CS2 INCORRECT
 6342  036024  023737  004172  004132  11$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
 6343  036032  001401                          BEQ     12$             ;YES, CHECK ERROR REGISTER
 6344  036034  104226                          ERROR   226             ;DRIVE STATUS REG INCORRECT
 6345  036036  023737  004174  004134  12$:    CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
 6346  036044  001401                          BEQ     TST71           ;;YES,GO ON TO NEXT TEST
```

```
 6347  036046  104227                          ERROR   227             ;ERROR REG. INCORRECT
 6348
 6349                                  ;;*****************************************************************
 6350                                  ;*TEST 71       IDAE DETECTION IN RK611 CONTROLLER (PART 4)
 6351                                  ;*
 6352                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
 6353                                  ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
 6354                                  ;*      TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 3, HEAD
 6355                                  ;*      4, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE
 6356                                  ;*      ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE
 6357                                  ;*      DRIVE AVAILABLE, UNSAFE, ILLEGAL DISK ADDRESS ERROR
 6358                                  ;*      AND CONTROLLER ERROR ARE SET.
 6359                                  ;*
 6360                                  ;;*****************************************************************
 6361  036050  000004                  TST71:  SCOPE
 6362  036052  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
 6363  036060  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
 6364  036064  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
 6365  036072  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
 6366  036100  012762  000003  000020          MOV     #3,RKDCYL(R2)   ;LOAD CYLINDER ADDRESS
 6367  036106  012737  000003  004252          MOV     #3,CYLIN
 6368  036114  012737  000004  004250          MOV     #4,HDCODE       ;LOAD HEAD ADDRESS
 6369  036122  005046                          CLR     -(SP)
 6370  036124  113766  004250  000001          MOVB    HDCODE,1(SP)
 6371  036132  012662  000006                  MOV     (SP)+,RKDA(R2)
 6372  036136  012737  000006  004266          MOV     #6,DRVTYP       ;LOAD DRIVE TYPE FOR PRINT OUT
 6373  036144  012762  000017  000000          MOV     #SEEK,RKCS1(R2) ;ISSUE SEEK TO RK06
 6374  036152  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCK TO GET THROUGH PHASE 6
 6375  036156  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 6376  036164  012762  000040  000026          MOV     #DMD,RKMR1(R2
 6377  036172  005300                          DEC     R0
 6378  036174  001370                          BNE     1$
 6379  036176  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
 6380  036202  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
 6381  036206  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
 6382  036212  100402                          BMI     3$
 6383  036214  005300                          DEC     R0
 6384  036216  001373                          BNE     2$
 6385  036220  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6386  036226  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6387  036234  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 6388  036242  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6389  036250  012737  100216  004160          MOV     #CERR!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD EXPECTED CS1
 6390
 6391  036256  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG 2
 6392  036264  012737  100001  004172          MOV     #SVAL!DRA,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
 6393  036272  012737  042000  004174          MOV     #UNS!IDAE,E.ER  ;LOAD EXPECTED ERROR REG
 6394  036300  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
 6395  036306  001401                          BEQ     4$              ;YES, CHECK CS2
 6396  036310  104230                          ERROR   230             ;CS1 INCORRECT
 6397  036312  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6398  036320  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
 6399  036322  104231                          ERROR   231             ;CS2 INCORRECT
 6400  036324  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 6401  036332  001401                          BEQ     6$              ;YES, CHECK ERROR REG
 6402  036334  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
```

```
6403  036336  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
6404  036344  001401                          BEQ     7$              ;YES, CHECK CONTROLLER CLEAR
6405  036346  104233                          ERROR   233             ;ERROR REG. INCORRECT
6406  036350  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS VALUES OF
6407  036356  013737  004130  004222          MOV     T.CS2,P.CS2     ;  COMMAND AND STATUS REG.1
6408  036364  013737  004132  004224          MOV     T.DS,P.DS       ;  COMMAND AND STATUS REG.2
6409  036372  013737  004134  004226          MOV     T.ER,P.ER       ;  DRIVE STATUS REG.
6410                                                                  ;  ERROR REG.
6411  036400  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
6412  036406  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6413  036414  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6414  036422  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
6415  036430  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
6416  036436  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
6417  036444  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
6418  036452  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
6419  036456  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG.
6420  036462  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
6421  036470  001401                          BEQ     10$             ;YES, CHECK CS2
6422  036472  104224                          ERROR   224             ;CS1 INCORRECT
6423  036474  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
6424  036502  001401                          BEQ     11$             ;YES, CHECK DRIVE STATUS REG
6425  036504  104225                          ERROR   225             ;CS2 INCORRECT
6426  036506  023737  004172  004132  11$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
6427  036514  001401                          BEQ     12$             ;YES, CHECK ERROR REGISTER
6428  036516  104226                          ERROR   226             ;DRIVE STATUS REG INCORRECT
6429  036520  023737  004174  004134  12$:    CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
6430  036526  001401                          BEQ     TST72           ;YES,GO ON TO NEXT TEST
6431  036530  104227                          ERROR   227             ;ERROR REG. INCORRECT
6432
6433                                  ;;*****************************************************************
6434                                  ;*TEST 72       IDAE DETECTION IN RK611 CONTROLLER (PART 5)
6435                                  ;*
6436                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
6437                                  ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
6438                                  ;*      WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 23, HEAD 5,
6439                                  ;*      DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS
6440                                  ;*      6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE
6441                                  ;*      AVAILABLE, UNSAFE, SPEED LOSS, ILLEGAL DISK ADDRESS
6442                                  ;*      ERROR, AND CONTROLLER ERROR ARE SET.
6443                                  ;*
6444                                  ;*****************************************************************
6445  036532  000004                  TST72:  SCOPE
6446  036534  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
6447  036542  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
6448  036546  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6449  036554  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
6450  036562  012762  000023  000020          MOV     #23,RKDCYL(R2)  ;LOAD CYLINDER ADDRESS
6451  036570  012737  000023  004252          MOV     #23,CYLIN
6452  036576  012737  000005  004250          MOV     #5,HDCODE       ;LOAD HEAD ADDRESS
6453  036604  005046                          CLR     -(SP)
6454  036606  113766  004250  000001          MOVB    HDCODE,1(SP)
6455  036614  012662  000006                  MOV     (SP)+,RKDA(R2)
6456  036620  012737  000007  004266          MOV     #7,DRVTYP       ;LOAD DRIVE TYPE FOR PRINT OUT
6457  036626  012762  002017  000000          MOV     #CDT!SEEK,RKCS1(R2)  ;ISSUE SEEK TO RK06
6458  036634  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCK TO GET THROUGH PHASE 6
```

```
 6459  036640  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
 6460  036646  012762  000040  000026          MOV     #DMD,RKMR1(R2)
 6461  036654  005300                          DEC     R0
 6462  036656  001370                          BNE     1$
 6463  036660  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
 6464  036664  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
 6465  036670  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
 6466  036674  100402                          BMI     3$
 6467  036676  005300                          DEC     R0
 6468  036700  001373                          BNE     2$
 6469  036702  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6470  036710  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6471  036716  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
 6472  036724  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6473  036732  012737  102216  004160          MOV     #CERR!CDT!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD EXPECTED CS1
 6474
 6475  036740  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG.2
 6476  036746  012737  100421  004172          MOV     #SVAL!DRA!SPDLSS!DDT,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
 6477  036754  012737  042000  004174          MOV     #UNS!IDAE,E.ER  ;LOAD EXPECTED ERROR REG
 6478  036762  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
 6479  036770  001401                          BEQ     4$              ;YES, CHECK CS2
 6480  036772  104230                          ERROR   230             ;CS1 INCORRECT
 6481  036774  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6482  037002  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
 6483  037004  104231                          ERROR   231             ;CS2 INCORRECT
 6484  037006  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
 6485  037014  001401                          BEQ     6$              ;YES, CHECK ERROR REG
 6486  037016  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
 6487  037020  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
 6488  037026  001401                          BEQ     7$              ;YES, CHECK CONTROLLER CLEAR
 6489  037030  104233                          ERROR   233             ;ERROR REG. INCORRECT
 6490  037032  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS VALUES OF
 6491  037040  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG.1
 6492  037046  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG.2
 6493  037054  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG.
 6494                                                                  ;   ERROR REG.
 6495  037062  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
 6496  037070  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
 6497  037076  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
 6498  037104  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
 6499  037112  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
 6500  037120  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
 6501  037126  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
 6502  037134  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
 6503  037140  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG.
 6504  037144  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
 6505  037152  001401                          BEQ     10$             ;YES, CHECK CS2
 6506  037154  104224                          ERROR   224             ;CS1 INCORRECT
 6507  037156  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
 6508  037164  001401                          BEQ     11$             ;YES, CHECK DRIVE STATUS REG
 6509  037166  104225                          ERROR   225             ;CS2 INCORRECT
 6510  037170  023737  004172  004132  11$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
 6511  037176  001401                          BEQ     12$             ;YES, CHECK ERROR REGISTER
 6512  037200  104226                          ERROR   226             ;DRIVE STATUS REG INCORRECT
 6513  037202  023737  004174  004134  12$:    CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
 6514  037210  001401                          BEQ     TST73           ;;YES,GO ON TO NEXT TEST
```

```
6515  037212  104227                          ERROR   227             ;ERROR REG. INCORRECT
6516
6517                                  ;*****************************************************************
6518                                  ;*TEST 73       IDAE DETECTION IN RK611 CONTROLLER (PART 6)
6519                                  ;*
6520                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
6521                                  ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
6522                                  ;*      WITH CDT SET IN 26 SECTOR FORMAT, CYLINDER 23, HEAD 6,
6523                                  ;*      DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS
6524                                  ;*      6.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE
6525                                  ;*      AVAILABLE, UNSAFE, DRIVE OFF TRACK, ILLEGAL
6526                                  ;*      DISK ADDRESS ERROR, AND CONTROLLER CLEAR ARE SET.
6527                                  ;*
6528                                  ;*****************************************************************
6529  037214  C00004                  TST73:  SCOPE
6530  037216  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
6531  037224  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
6532  037230  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6533  037236  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
6534  037244  012762  000023  000020          MOV     #23,RKDCYL(R2)  ;LOAD CYLINDER ADDRESS
6535  037252  012737  000023  004252          MOV     #23,CYLIN
6536  037260  012737  000006  004250          MOV     #F.HDCODE       ;LOAD HEAD ADDRESS
6537  037266  005046                          CLR     -(SP)
6538  037270  113766  004250  000001          MOVB    HDCODE,1(SP)
6539  037276  012662  000006                  MOV     (SP)+,RKDA(R2)
6540  037302  012737  000007  004266          MOV     #7,DRVTYP       ;LOAD DRIVE TYPE FOR PRINT OUT
6541  037310  012762  002017  000000          MOV     #CDT!SEEK,RKCS1(R2)  ;ISSUE SEEK TO RK06
6542  037316  012700  000132                  MOV     #22.*4+2,R0     ;ISSUE CLOCK TO GET THROUGH PHASE 6
6543  037322  012762  000440  000026  1$:     MOV     #DMD!MCLK,RKMR1(R2)
6544  037330  012762  000040  000026          MOV     #DMD,RKMR1(R2)
6545  037336  005300                          DEC     R0
6546  037340  001370                          BNE     1$
6547  037342  005062  000026                  CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
6548  037346  013700  004262                  MOV     WAITIM,R0       ;LOAD WAIT TIME
6549  037352  105762  000000          2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
6550  037356  100402                          BMI     3$
6551  037360  005300                          DEC     R0
6552  037362  001373                          BNE     2$
6553  037364  016237  000000  004120  3$:     MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6554  037372  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6555  037400  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
6556  037406  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
6557  037414  012737  102216  004160          MOV     #CERR!CDT!RDY!<SEEK&^C<GO>>,E.CS1  ;LOAD EXPECTED CS1
6558
6559  037422  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED COMMAND AND STATUS REG.2
6560  037430  012737  100441  004172          MOV     #SVAL!DRA!DROT!DDT,E.DS  ;LOAD EXPECTED DRIVE STATUS REG
6561  037436  012737  042000  004174          MOV     #UNS!IDAE,E.ER  ;LOAD EXPECTED ERROR REG
6562  037444  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
6563  037452  001401                          BEQ     4$              ;YES, CHECK CS2
6564  037454  104230                          ERROR   230             ;CS1 INCORRECT
6565  037456  023737  004170  004130  4$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
6566  037464  001401                          BEQ     5$              ;YES, CHECK DRIVE STATUS REG.
6567  037466  104231                          ERROR   231             ;CS2 INCORRECT
6568  037470  023737  004172  004132  5$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
6569  037476  001401                          BEQ     6$              ;YES, CHECK ERROR REG
6570  037500  104232                          ERROR   232             ;DRIVE STATUS REG. INCORRECT
```

```
6571  037502  023737  004174  004134  6$:     CMP     E.ER,T.ER       ;CHECK ERROR REG. CORRECT
6572  037510  001401                          BEQ     7$              ;YES, CHECK CONTROLLER CLEAR
6573  037512  104233                          ERROR   233             ;ERROR REG. INCORRECT
6574  037514  013737  004120  004220  7$:     MOV     T.CS1,P.CS1     ;STORE PREVIOUS VALUES OF
6575  037522  013737  004130  004222          MOV     T.CS2,P.CS2     ;   COMMAND AND STATUS REG.1
6576  037530  013737  004132  004224          MOV     T.DS,P.DS       ;   COMMAND AND STATUS REG.2
6577  037536  013737  004134  004226          MOV     T.ER,P.ER       ;   DRIVE STATUS REG.
6578                                                                  ;   ERROR REG.
6579  037544  012762  100000  000000          MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
6580  037552  016237  000000  004120          MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6581  037560  016237  000010  004130          MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6582  037566  016237  000012  004132          MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
6583  037574  016237  000014  004134          MOV     RKER(R2),T.ER   ;STORE ERROR REG
6584  037602  012737  000200  004160          MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
6585  037610  012737  000100  004170          MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
6586  037616  005037  004172                  CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
6587  037622  005037  004174                  CLR     E.ER            ;LOAD EXPECTED ERROR REG.
6588  037626  023737  004160  004120          CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
6589  037634  001401                          BEQ     10$             ;YES, CHECK CS2
6590  037636  104224                          ERROR   224             ;CS1 INCORRECT
6591  037640  023737  004170  004130  10$:    CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG2 CORRECT
6592  037646  001401                          BEQ     11$             ;YES, CHECK DRIVE STATUS REG
6593  037650  104225                          ERROR   225             ;CS2 INCORRECT
6594  037652  023737  004172  004132  11$:    CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
6595  037660  001401                          BEQ     12$             ;YES, CHECK ERROR REGISTER
6596  037662  104226                          ERROR   226             ;DRIVE STATUS REG INCORRECT
6597  037664  023737  004174  004134  12$:    CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
6598  037672  001401                          BEQ     TST74           ;YES,GO ON TO NEXT TEST
6599  037674  104227                          ERROR   227             ;ERROR REG. INCORRECT
6600
6601                                  ;*****************************************************************
6602                                  ;*TEST 74       NON-STANDARD MESSAGE RECEIVING
6603                                  ;*
6604                                  ;*      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
6605                                  ;*      RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A SEEK
6606                                  ;*      WITH CDT SET IN 24 SECTOR FORMAT, CYLINDER 1757, HEAD 7,
6607                                  ;*      DRIVE 1.  CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
6608                                  ;*      TURN OFF DIAGNOSTIC MODE AND MAKE SURE NO ERRORS SET
6609                                  ;*      AND DRIVE STATUS IS NOT REPORTED.  REPEAT FOR DRIVES
6610                                  ;*      2 AND 4.
6611                                  ;*
6612                                  ;*****************************************************************
6613  037676  000004                  TST74:  SCOPE
6614  037700  012737  000144  001200          MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
6615  037706  013702  001270                  MOV     $BASE,R2        ;LOAD RK611 BASE
6616  037712  012737  000001  004244          MOV     #1,DRVCOD       ;LOAD INITIAL DRIVE CODE
6617  037720  012737  037726  001110          MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
6618                                                                  ;   SUBTEST LOOP
6619
6620  037726                          1$:
6621  037726  012762  000040  000010          MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6622  037734  012762  000040  000026          MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
6623  037742  012762  001757  000020          MOV     #1757,RKDCYL(R2)  ;LOAD CYLINDER ADDRESS REG
6624  037750  012762  003400  000006          MOV     #3400,RKDA(R2)  ;LOAD HEAD 7
6625  037756  013762  004244  000010          MOV     DRVCOD,RKCS2(R2)  ;LOAD DRIVE NUMBER
6626  037764  012762  002017  000000          MOV     #CDT!SEEK,RKCS1(R2)  ;ISSUE A SEEK WITH CDT SET
```

```
6627  037772  012700  000132                 MOV     #22.*4+2,R0      ;ISSUE CLOCKS THROUGH PHASE 6
6628  037776  012762  000440  000026  2$:    MOV     #DMD!MCLK,RKMR1(R2)
6629  040004  012762  000040  000026         MOV     #DMD,RKMR1(R2)
6630  040012  005300                         DEC     R0
6631  040014  001370                         BNE     2$
6632  040016  005062  000026                 CLR     RKMR1(R2)        ;ALLOW COMMAND TO FINISH
6633  040022  013700  004262                 MOV     WAITIM,R0        ;LOAD WAIT TIME
6634  040026  105762  000000          3$:    TSTB    RKCS1(R2)        ;WAIT FOR READY
6635  040032  100402                         BMI     4$
6636  040034  005300                         DEC     R0
6637  040036  001373                         BNE     3$
6638  040040  016237  000000  004120  4$:    MOV     RKCS1(R2),T.CS1  ;STORE COMMAND AND STATUS REG.1
6639  040046  016237  000010  004130         MOV     RKCS2(R2),T.CS2  ;STORE COMMAND AND STATUS REG.2
6640  040054  016237  000012  004132         MOV     RKDS(R2),T.DS    ;STORE DRIVE STATUS REG.
6641  040062  016237  000014  004134         MOV     RKER(R2),T.ER    ;STORE ERROR REG
6642  040070  012737  002216  004160         MOV     #CDT!RDY!<SEEK&^C<GO>>,E.CS1   ;LOAD EXPECTED CS1
6643  040076  013737  004244  004170         MOV     DRVCOD,E.CS2     ;LOAD EXPECTED CS2
6644  040104  052737  000100  004170         BIS     #IR,E.CS2
6645  040112  005037  004172                 CLR     E.DS             ;LOAD EXPECTED DRIVE STATUS REG.
6646  040116  005037  004174                 CLR     E.ER             ;LOAD EXPECTED ERROR REG.
6647  040122  023737  004160  004120         CMP     E.CS1,T.CS1      ;CHECK COMMAND AND STATUS REG.1 CORRECT
6648  040130  001401                         BEQ     5$               ;YES, CHECK CS2
6649  040132  104234                         ERROR   234              ;CS1 INCORRECT
6650  040134  023737  004170  004130  5$:    CMP     E.CS2,T.CS2      ;CHECK COMMAND AND STATUS REG.2 CORRECT
6651  040142  001401                         BEQ     6$               ;YES, CHECK DRIVE STATUS REG.
6652  040144  104235                         ERROR   235              ;CS2 INCORRECT
6653  040146  023737  004172  004132  6$:    CMP     E.DS,T.DS        ;CHECK DRIVE STATUS REG CORRECT
6654  040154  001401                         BEQ     7$               ;YES, CHECK ERROR REG
6655  040156  104236                         ERROR   236              ;DRIVE STATUS REG INCORRECT
6656  040160  023737  004174  004134  7$:    CMP     E.ER,T.ER        ;CHECK IF ERROR CORRECT
6657  040166  001401                         BEQ     8$               ;YES, CHECK IF LOOP ON ERROR
6658  040170  104237                         ERROR   237              ;ERROR REG INCORRECT
6659  040172  104415                 8$:     SCOPE                    ;CHECK IF LOOP ON ERROR
6660  040174  006337  004244                 ASL     DRVCOD           ;GENERATE NEXT DRIVE COME
6661  040200  032737  000010  004244         BIT     #BIT3,DRVCOD     ;CHECK IF FINISHED
6662  040206  001647                         BEQ     1$               ;NO, TRY NEXT COME
6663
6664                                  ;*****************************************************************
6665                                  ;*TEST 75       DRIVE BUS PARITY ON NON-STANDARD MESSAGE
6666                                  ;*
6667                                  ;*       CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
6668                                  ;*       PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE
6669                                  ;*       A SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2,
6670                                  ;*       HEAD 0, DRIVE 1.  CLOCK IN DIAGNOSTIC MODE UNTIL
6671                                  ;*       PHASE ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE
6672                                  ;*       SURE DRIVE BUS PARITY ERROR AND CONTROLLER ERROR SETS.
6673                                  ;*
6674                                  ;*****************************************************************
6675  040210  000004                 TST75:  SCOPE
6676  040212  012737  000144  001200         MOV     #100.,$TIMES     ;;DO 100. ITERATIONS
6677  040220  013702  001270                 MOV     $BASE,R2         ;LOAD RK611 BASE
6678  040224  012762  000040  000010         MOV     #SCLR,RKCS2(R2)  ;CLEAR RK06 SUBSYSTEM
6679  040232  012762  000040  000026         MOV     #DMD,RKMR1(R2)   ;PUT RK611 IN MAINTENANCE MODE
6680  040240  012762  000002  000020         MOV     #2,RKDCYL(R2)    ;LOAD CYLINDER ADDRESS REG
6681  040246  012762  000001  000010         MOV     #1,RKCS2(R2)     ;LOAD DRIVE NUMBER 1
6682  040254  012762  000017  000000         MOV     #SEEK,RKCS1(R2)  ;ISSUE SEEK
```

```
6683  040262  012700  000132                    MOV     #22.*4+2,R0     ;ISSUE CLOCKS THROUGH PHASE 6
6684  040266  012762  000440  000026  1$:       MOV     #DMD!MCLK,RKMR1(R2)
6685  040274  012762  000040  000026            MOV     #DMD,RKMR1(R2)
6686  040302  005300                            DEC     R0
6687  040304  001370                            BNE     1$
6688  040306  005062  000026                    CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
6689  040312  013700  004262                    MOV     WAITIM,R0       ;LOAD WAIT TIME
6690  040316  105762  000000            3$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
6691  040322  100402                            BMI     4$
6692  040324  005300                            DEC     R0
6693  040326  001373                            BNE     3$
6694  040330  016237  000000  004120  4$:       MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6695  040336  016237  000010  004130            MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6696  040344  016237  000012  004132            MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
6697  040352  016237  000014  004134            MOV     RKER(R2),T.ER   ;STORE ERROR REG.
6698  040360  012737  120216  004160            MOV     #CERR!SPAR!RDY!SEEK&^C<GO>,E.CS1  ;LOAD EXPECTED CS1
6699  040366  012737  000101  004170            MOV     #IR!1,E.CS2     ;LOAD EXPECTED CS1
6700  040374  005037  004172                    CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
6701  040400  005037  004174                    CLR     E.ER            ;LOAD EXPECTED ERROR REG.
6702  040404  023737  004160  004120            CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
6703  040412  001401                            BEQ     5$              ;YES, CHECK CS2
6704  040414  104240                            ERROR   240             ;CS1 INCORRECT
6705  040416  023737  004170  004130  5$:       CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG.2 CORRECT
6706  040424  001401                            BEQ     6$              ;YES, CHECK DRIVE STATUS REG
6707  040426  104241                            ERROR   241             ;CS2 INCORRECT
6708  040430  023737  004172  004132  6$:       CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG. CORRECT
6709  040436  001401                            BEQ     7$              ;YES, CHECK ERROR REG.
6710  040440  104242                            ERROR   242             ;DRIVE STATUS REG. INCORRECT
6711  040442  023737  004174  004134  7$:       CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
6712  040450  001401                            BEQ     8$              ;YES, CLEAR RK611
6713  040452  104243                            ERROR   243             ;ERROR REG. INCORRECT
6714  040454  013737  004120  004220  8$:       MOV     T.CS1,P.CS1     ;STORE PREVIOUS CS1, CS2,
6715  040462  013737  004130  004222            MOV     T.CS2,P.CS2     ;  DRIVE STATUS REG.,
6716  040470  013737  004132  004224            MOV     T.DS,P.DS       ;  AND ERROR REG.
6717  040476  013737  004134  004226            MOV     T.ER,P.ER
6718  040504  012762  100000  000000            MOV     #CCLR,RKCS1(R2) ;CLEAR RK611
6719  040512  016237  000000  004120            MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6720  040520  016237  000010  004130            MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6721  040526  016237  000012  004132            MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
6722  040534  016237  000014  004134            MOV     RKER(R2),T.ER   ;STORE ERROR REG:
6723  040542  012737  000200  004160            MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
6724  040550  012737  000100  004170            MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
6725  040556  005037  004172                    CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
6726  040562  005037  004174                    CLR     E.ER            ;LOAD EXPECTED ERROR REG.
6727  040566  023737  004160  004120            CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
6728  040574  001401                            BEQ     10$             ;YES, CHECK CS2
6729  040576  104224                            ERROR   224             ;CS1 INCORRECT
6730  040600  023737  004170  004130  10$:      CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG.2 CORRECT
6731  040606  001401                            BEQ     11$             ;YES, CHECK DRIVE STATUS REG
6732  040610  104225                            ERROR   225             ;CS2 INCORRECT
6733  040612  023737  004172  004132  11$:      CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
6734  040620  001401                            BEQ     12$             ;YES, CHECK ERROR REG
6735  040622  104226                            ERROR   226             ;DRIVE STATUS REG. INCOPRECT
6736  040624  023737  004174  004134  12$:      CMP     E.ER,T.ER       ;CHECK ERROR CORRECT
6737  040632  001401                            BEQ     TST76           ;YES, GO ON TO NEXT TEST
6738  040634  104227                            ERROR   227             ;ERROR REG INCORRECT
```

```
                                        ;;*****************************************************************
                                        ;*TEST 76       NON-EXISTENT DRIVE (DRIVE MESSAGE TIME OUT)
                                        ;*
                                        ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
                                        ;*      PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE
                                        ;*      A SELECT TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 0,
                                        ;*      HEAD 0, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL
                                        ;*      PHASE ADDRESS 5.  TURN OFF DIAGNOSTIC MODE
                                        ;*      AND MAKE SURE NON-EXISTENT DRIVE AND CONTROLLER
                                        ;*      ERROR ARE SET.  THIS TEST CHECKS NON-EXISTENT DRIVE
                                        ;*      DUE TO DRIVE MESSAGE TIME OUT.
                                        ;*
                                        ;;*****************************************************************
040636  000004                    TST76:  SCOPE
040640  012737  000144  001200            MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
040646  013702  001270                    MOV     $BASE,R2        ;LOAD RK611 BASE
040652  012762  000040  000010            MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
040660  012762  000040  000026            MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
040666  012762  000001  000000            MOV     #SELDRV,RKCS1(R2)  ;ISSUE SELECT DRIVE
040674  012700  000124                    MOV     #21.*4,R0       ;ISSUE CLOCKS THROUGH PHASE 4
040700  012762  000440  000026    1$:     MOV     #DMD!MCLK,RKMR1(R2)
040706  012762  000040  000026            MOV     #DMD,RKMR1(R2)
040714  005300                            DEC     R0
040716  001370                            BNE     1$
040720  005062  000026                    CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
040724  013700  004262                    MOV     WAITIM,R0       ;LOAD WAIT TIME
040730  105762  000000            2$:     TSTB    RKCS1(R2)       ;WAIT FOR READY
040734  100402                            BMI     3$
040736  005300                            DEC     R0
040740  001373                            BNE     2$
040742  013700  004264            3$:     MOV     $TALL,R0        ;STALL 100 USEC FOR MESSAGE TIME OUT
040746  005300                    4$:     DEC     R0
040750  001376                            BNE     4$
040752  016237  000000  004120            MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
040760  016237  000010  004130            MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
040766  016237  000012  004132            MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG
040774  016237  000014  004134            MOV     RKER(R2),T.ER   ;STORE ERROR REG
041002  012737  100200  004160            MOV     #CERR!RDY,E.CS1 ;LOAD EXPECTED CS1
041010  032737  020000  004120            BIT     #SPAR,T.CS1     ;CHECK FOR BUS PARITY ERROR
041016  001403                            BEQ     5$
041020  052737  020000  004160            BIS     #SPAR,E.CS1     ;PUT BUS PARITY ERROR IN EXPECTED CS1
041026  012737  010100  004170    5$:     MOV     #NED!IR,E.CS2   ;LOAD EXPECTED CS2
041034  012737  100000  004172            MOV     #SVAL,E.DS      ;LOAD EXPECTED DRIVE STATUS REG.
041042  005037  004174                    CLR     E.ER            ;LOAD EXPECTED ERROR REG.
041046  023737  004160  004120            CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG.1 CORRECT
041054  001401                            BEQ     6$              ;YES, CHECK CS2
041056  104244                            ERROR   244             ;CS1 INCORRECT
041060  023737  004170  004130    6$:     CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG.2 CORRECT
041066  001401                            BEQ     7$              ;YES, CHECK DRIVE STATUS REG
041070  104245                            ERROR   245             ;CS2 INCORRECT
041072  023737  004172  004132    7$:     CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
041100  001401                            BEQ     8$              ;YES, CHECK ERROR REG.
041102  104246                            ERROR   246             ;DRIVE STATUS INCORRECT
041104  023737  004174  004134    8$:     CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
041112  001401                            BEQ     9$              ;YES, ISSUE CONTROLLER CLEAR
```

```
                                        ERROR   247             ;ERROR REG INCORRECT
041114  104247
041116  013737  004120  004220  9$:   MOV     T.CS1,P.CS1     ;STORE PREVIOUS CS1,CS2
041124  013737  004130  004222        MOV     T.CS2,P.CS2     ;  DRIVE STATUS REG.,
041132  013737  004132  004224        MOV     T.DS,P.DS       ;  AND ERROR REG.
041140  013737  004134  004226        MOV     T.ER,P.ER
041146  012762  100000  000000        MOV     #CCLR,RKCS1(R2) ;ISSUE CONTROLLER CLEAR
041154  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
041162  016237  000010  004130        MOV     RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
041170  016237  000012  004132        MOV     RKDS(R2),T.DS   ;STORE DRIVE STATUS REG.
041176  016237  000014  004134        MOV     RKER(R2),T.ER   ;STORE ERROR REG.
041204  012737  000200  004160        MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
041212  012737  000100  004170        MOV     #IR,E.CS2       ;LOAD EXPECTED CS2
041220  005037  004172                CLR     E.DS            ;LOAD EXPECTED DRIVE STATUS REG.
041224  005037  004174                CLR     E ER            ;LOAD EXPECTED ERROR REG.
041230  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG1 CORRECT
041236  001401                        BEQ     10$             ;YES, CHECK CS2
041240  104224                        ERROR   224             ;CS1 INCORRECT
041242  023737  004170  004130  10$:   CMP     E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG.2 CORRECT
041250  001401                        BEQ     11$             ;YES, CHECK DRIVE STATUS REG.
041252  104225                        ERROR   225             ;CS2 INCORRECT
041254  023737  004172  004132  11$:   CMP     E.DS,T.DS       ;CHECK DRIVE STATUS REG CORRECT
041262  001401                        BEQ     12$             ;YES, CHECK ERROR REG
041264  104226                        ERROR   226             ;DRIVE STATUS INCORRECT
041266  023737  004174  004134  12$:   CMP     E.ER,T.ER       ;CHECK ERROR REG CORRECT
041274  001401                        BEQ     TST77           ;YES, GO ON TO NEXT TEST
041276  104227                        ERROR   227             ;ERROR MESSAGE INCORRECT

                        ;*****************************************************************
                        ;*TEST 77       NON-EXISTENT DRIVE AND NO SACK
                        ;*
                        ;*      CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.  PUT
                        ;*      THE RK611 CONTROLLER IN DIAGNOSTIC MODE.  ISSUE A
                        ;*      SELECT TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 0,
                        ;*      HEAD 0, DRIVE 0.  CLOCK IN DIAGNOSTIC MODE UNTIL
                        ;*      PHASE ADDRESS 4.  TURN OFF DIAGNOSTIC MODE AND MAKE SURE
                        ;*      NON-EXISTENT DRIVE AND CONTROLLER ERROR ARE SET.
                        ;*
                        ;*      THIS TEST EXERCISES THE NON-EXISTENT DRIVE LOGIC
                        ;*      DUE TO RELEASE BIT RESET AND SACK RESET BUT THE PASSING
                        ;*      OF THIS TEST DOES GUARENTEE THAT THIS SITUATION DID
                        ;*      INDEED CAUSE A NON-EXISTENT DRIVE.
                        ;*
                        ;*****************************************************************
000004          TST77:  SCOPE
041302  012737  000144  001200        MOV     #100.,$TIMES    ;;DO 100. ITERATIONS
041310  013702  001270                MOV     $BASE,R2        ;LOAD RK611 BASE
041314  012762  000040  000010        MOV     #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
041322  012762  000040  000026        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN MAINTENANCE MODE
041330  012762  000001  000000        MOV     #SELDRV,RKCS1(R2)  ;ISSUE SELECT DRIVE
041336  012700  000116                MOV     #19.*4+2,R0     ;ISSUE CLOCKS THROUGH PHASE 3
041342  012762  000440  000026  1$:   MOV     #DMD!MCLK,RKMR1(R2)
041350  012762  000040  000026        MOV     #DMD,RKMR1(R2)
041356  005300                        DEC     R0
041360  001370                        BNE     1$
041362  005062  000026                CLR     RKMR1(R2)       ;ALLOW COMMAND TO FINISH
041366  013700  004262                MOV     WAITIM,R0       ;LOAD WAIT TIME
```

```
6851  041372  105762  000000          3$:     TSTB    RKCS1(R2)          ;WAIT FOR READY
6852  041376  100402                          BMI     4$
6853  041400  005300                          DEC     R0
6854  041402  001373                          BNE     3$
6855  041404  016237  000000  004120  4$:     MOV     RKCS1(R2),T.CS1    ;STORE COMMAND AND STATUS REG.1
6856  041412  016237  000010  004130          MOV     RKCS2(R2),T.CS2    ;STORE COMMAND AND STATUS REG.2
6857  041420  016237  000012  004132          MOV     RKDS(R2),T.DS      ;STORE DRIVE STATUS REG
6858  041426  016237  000014  004134          MOV     RKER(R2),T.ER      ;STORE ERROR REG
6859  041434  012737  100200  004160          MOV     #CERR!RDY,E.CS1    ;LOAD EXPECTED CS1
6860  041442  012737  010100  004170          MOV     #NED!IR,E.CS2      ;LOAD EXPECTED CS2
6861  041450  005037  004172                  CLR     E.DS               ;LOAD EXPECTED DRIVE STATUS REG.
6862  041454  005037  004174                  CLR     E.ER               ;LOAD EXPECTED ERROR REG.
6863  041460  023737  004160  004120          CMP     E.CS1,T.CS1        ;CHECK COMMAND AND STATUS REG.1 CORRECT
6864  041466  001401                          BEQ     5$                 ;YES, CHECK CS2
6865  041470  104250                          ERROR   250                ;CS1 INCORRECT
6866  041472  023737  004170  004130  5$:     CMP     E.CS2,T.CS2        ;CHECK COMMAND AND STATUS REG.2 CORRECT
6867  041500  001401                          BEQ     6$                 ;YES, CHECK DRIVE STATUS REG
6868  041502  104251                          ERROR   251                ;CS2 INCORRECT
6869  041504  023737  004172  004132  6$:     CMP     E.DS,T.DS          ;CHECK DRIVE STATUS REG CORRECT
6870  041512  001401                          BEQ     7$                 ;YES, CHECK ERROR REG.
6871  041514  104252                          ERROR   252                ;DRIVE STATUS INCORRECT
6872  041516  023737  004174  004134  7$:     CMP     E.ER,T.ER          ;CHECK ERROR REG CORRECT
6873  041524  001401                          BEQ     8$                 ;YES, ISSUE CONTROLLER CLEAR
6874  041526  104253                          ERROR   253                ;ERROR REG INCORRECT
6875  041530  013737  004120  004220  8$:     MOV     T.CS1,P.CS1        ;STORE PREVIOUS CS1,CS2
6876  041536  013737  004130  004222          MOV     T.CS2,P.CS2        ;  DRIVE STATUS REG.,
6877  041544  013737  004132  004224          MOV     T.DS,P.DS          ;  AND ERROR REG.
6878  041552  013737  004134  004226          MOV     T.ER,P.ER
6879  041560  012762  100000  000000          MOV     #CCLR,RKCS1(R2)    ;ISSUE CONTROLLER CLEAR
6880  041566  016237  000000  004120          MOV     RKCS1(R2),T.CS1    ;STORE COMMAND AND STATUS REG.1
6881  041574  016237  000010  004130          MOV     RKCS2(R2),T.CS2    ;STORE COMMAND AND STATUS REG.2
6882  041602  016237  000012  004132          MOV     RKDS(R2),T.DS      ;STORE DRIVE STATUS REG.
6883  041610  016237  000014  004134          MOV     RKER(R2),T.ER      ;STORE ERROR REG.
6884  041616  012737  000200  004160          MOV     #RDY,E.CS1         ;LOAD EXPECTED CS1
6885  041624  012737  000100  004170          MOV     #IR,E.CS2          ;LOAD EXPECTED CS2
6886  041632  005037  004172                  CLR     E.DS               ;LOAD EXPECTED DRIVE STATUS REG.
6887  041636  005037  004174                  CLR     E.ER               ;LOAD EXPECTED ERROR REG.
6888  041642  023737  004160  004120          CMP     E.CS1,T.CS1        ;CHECK COMMAND AND STATUS REG1 CORRECT
6889  041650  001401                          BEQ     10$                ;YES, CHECK CS2
6890  041652  104224                          ERROR   224                ;CS1 INCORRECT
6891  041654  023737  004170  004130  10$:    CMP     E.CS2,T.CS2        ;CHECK COMMAND AND STATUS REG.2 CORRECT
6892  041662  001401                          BEQ     11$                ;YES, CHECK DRIVE STATUS REG.
6893  041664  104225                          ERROR   225                ;CS2 INCORRECT
6894  041666  023737  004172  004132  11$:    CMP     E.DS,T.DS          ;CHECK DRIVE STATUS REG CORRECT
6895  041674  001401                          BEQ     12$                ;YES, CHECK ERROR REG
6896  041676  104226                          ERROR   226                ;DRIVE STATUS INCORRECT
6897  041700  023737  004174  004134  12$:    CMP     E.ER,T.ER          ;CHECK ERROR REG CORRECT
6898  041706  001401                          BEQ     TST100             ;YES, GO ON TO NEXT TEST
6899  041710  104227                          ERROR   227                ;ERROR MESSAGE INCORRECT
6900
6901                                  .SBTTL  **ILLEGAL FUNCTION CODE TEST
6902
6903                                  ;;*****************************************************************
6904                                  ;*TEST 100       ILLEGAL FUNCTION CODE
6905                                  ;*
6906                                  ;*      CLEAR RK611 WITH A CONTROLLER CLEAR.  ISSUE AN ILLEGAL
```

```
                                ;*      COMMAND IN NORMAL MODE AND MAKE SURE COMMAND FINISHES
                                ;*      SETTING CONTROLLER READY WITH PROPER ERROR CONDITIONS.
                                ;*
                                ;**************************************************************
041712  000004                TST100: SCOPE
041714  012737  000764  001200        MOV     #500.,$TIMES    ;;DO 500. ITERATIONS
041722  012737  000033  004270        MOV     #33,ILLFUN      ;SET ILLEGAL FUNCTION
041730  012737  041736  001110        MOV     #1$,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
                                                              ;  SUBTEST LOOP
041736                          1$:
041736  012762  100000  000000        MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
041744  013737  004270  004160        MOV     ILLFUN,E.CS1    ;GENERATE EXPECTED CS1
041752  042737  000001  004160        BIC     #GO,E.CS1
041760  052737  100200  004160        BIS     #CERR!RDY,E.CS1
041766  012737  000001  004174        MOV     #ILF,E.ER       ;LOAD EXPECTED ERROR REG
041774  012762  000040  000026        MOV     #DMD,RKMR1(R2)  ;PUT RK611 IN DIAGNOSTIC MODE
042002  013762  004270  000000        MOV     ILLFUN,RKCS1(R2) ;ISSUE ILLEGAL FUNCTION
042010  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
042016  016237  000014  004134        MOV     RKER(R2),T.ER   ;STORE ERROR REG
042024  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT
042032  001401                        BEQ     3$              ;YES, CHECK ERROR REG
042034  104256                        ERROR   256             ;CS1 INCORRECT AFTER ILL FUNCT
042036  023737  004174  004134  3$:   CMP     E.ER,T.ER       ;CHECK IF ERROR REG CORRECT
042044  001401                        BEQ     4$              ;YES, CLEAR CONTROLLER
042046  104257                        ERROR   257             ;ERROR REG INCORRECT AFTER ILL FUNCT
042050  012762  100000  000000  4$:   MOV     #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
042056  016237  000000  004120        MOV     RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
042064  016237  000014  004134        MOV     RKER(R2),T.ER   ;STORE ERROR REG
042072  012737  000200  004160        MOV     #RDY,E.CS1      ;LOAD EXPECTED CS1
042100  005037  004174                CLR     E.ER            ;LOAD EXPECTED ERROR REG
042104  023737  004160  004120        CMP     E.CS1,T.CS1     ;CHECK IF CS1 CORRECT (CERR CLEAR)
042112  001401                        BEQ     6$              ;YES, CHECK IF ERROR REG CORRECT
042114  104260                        ERROR   260             ;CONTROL CLEAR DID NOT CLEAR CERR
042116  023737  004174  004134  6$:   CMP     E.ER,T.ER       ;CHECK IF ILF CLEARED
042124  001401                        BEQ     7$              ;YES, GO ON TO NEXT CONFIGURATION
042126  104261                        ERROR   261             ;CONTROLLER CLEAR DID NOT CLEAR ILF
042130  104415                  7$:   SCOPE1                  ;CHECK IF LOOP ON ERROR
042132  062737  000002  004270        ADD     #2,ILLFUN       ;GENERATE NEXT ILLEGAL FUNCTION
042140  022737  000041  004270        CMP     #41,ILLFUN      ;CHECK IF FINISHED
042146  101273                        BHI     1$              ;NO, USE NEXT CONFIGURATION
```

```
                                        .SBTTL  END OF PASS ROUTINE

                                        ;;*****************************************************************
                                        ;*INCREMENT THE PASS NUMBER ($PASS)
                                        ;*TYPE "END PASS #XXXXX TOTAL NUMBER OF ERRORS SINCE LAST REPORT YYYYY"
                                        ;*WHERE XXXXX AND YYYYY ARE DECIMAL NUMBERS
                                        ;*IF THERE'S A MONITOR GO TO IT
                                        ;*IF THERE ISN'T JUMP TO NEWPAS

042150                                  $EOP:
042150  000004                                  SCOPE
042152  005037  001102                          CLR     $TSTNM          ;;ZERO THE TEST NUMBER
042156  005037  001200                          CLR     $TIMES          ;;ZERO THE NUMBER OF ITERATIONS
042162  005237  001222                          INC     $PASS           ;;INCREMENT THE PASS NUMBER
042166  042737  100000  001222                  BIC     #100000,$PASS   ;;DON'T ALLOW A NEG. NUMBER
042174  005327                                  DEC     (PC)+           ;;LOOP?
042176  000001                          $EOPCT: .WORD   1
042200  003063                                  BGT     $DOAGN          ;;YES
042202  012737                                  MOV     (PC)+,@(PC)+    ;;RESTORE COUNTER
042204  000001                          $ENDCT: .WORD   1
042206  042176                                  $EOPCT
042210  104401  042216                          TYPE    ,65$            ;;TYPE ASCIZ STRING
042214  000407                                  BR      64$             ;;GET OVER THE ASCIZ
                                        ;;65$:  .ASCIZ  <12><15>/END PASS #/
042234                                  64$:
042234  013746  001222                          MOV     $PASS,-(SP)     ;;SAVE $PASS FOR TYPEOUT
                                                                        ;;TYPE PASS NUMBER
042240  104405                                  TYPDS                   ;;GO TYPE--DECIMAL ASCII WITH SIGN
042242  104401  042250                          TYPE    ,67$            ;;TYPE ASCIZ STRING
042246  000421                                  BR      66$             ;;GET OVER THE ASCIZ
                                        ;;67$:  .ASCIZ  /  TOTAL ERRORS SINCE LAST REPORT /
042312                                  66$:
042312  013746  001112                          MOV     $ERTTL,-(SP)    ;;SAVE $ERTTL FOR TYPEOUT
                                                                        ;;TOTAL NUMBER OF ERRORS
042316  104405                                  TYPDS                   ;;GO TYPE--DECIMAL ASCII WITH SIGN
042320  104401  001211                          TYPE    ,$CRLF          ;;TYPE CARRIAGE RETURN, LINE FEED
042324  005037  001112                          CLR     $ERTTL          ;;CLEAR ERROR TOTAL
042330  013700  000042                  $GET42: MOV     @#42,R0         ;;GET MONITOR ADDRESS
042334  001405                                  BEQ     $DOAGN          ;;BRANCH IF NO MONITOR
042336  000005                                  RESET                   ;;CLEAR THE WORLD
042340  004710                          $ENDAD: JSR     PC,(R0)         ;;GO TO MONITOR
042342  000240                                  NOP                     ;;SAVE ROOM
042344  000240                                  NOP                     ;;FOR
042346  000240                                  NOP                     ;;ACT11
042350                                  $DOAGN:
042350  000137                                  JMP     @(PC)+          ;;RETURN
042352  005254                          $RTNAD: .WORD   NEWPAS
042354      377       377       000     $ENULL: .BYTE   -1,-1,0         ;;NULL CHARACTER STRING
042360                                          .EVEN

                                        .SBTTL  CHECK FOR MEMORY CHECK ENABLE OPTION

042360  012737  042432  000004  CHKPAR: MOV     #20$,ERRVEC     ;SET VECTOR FOR MEMORY PARITY CHECK
042366  012737  000340  000006          MOV     #PR7,ERRVEC+2
042374  012703  172100                  MOV     #MEMBAS,R3      ;LOAD REGISTER TO DETERMINE IF
                                                                ;   MEMORY CHECK ENABLE AVAILIABLE
```

```
7005  042400  012704  000020            MOV     #16.,R4             ;LOAD COUNT
7006  042404  012723  000001  16$:      MOV     #PAR.EN,(R3)+       ;ENABLE MEMORY CHECK
7007  042410  012737  042450  000114    MOV     #MEMERR,MEMVEC      ;LOAD MEMORY CHECK VECTOR
7008  042416  012737  000340  000116    MOV     #PR7,MEMVEC+2
7009  042424  005304                    DEC     R4                  ;CHECK IF FINISHED
7010  042426  001366                    BNE     16$                 ;NO, SET UP NEXT MEMORY PARITY MODULE
7011  042430  000401                    BR      22$                 ;RESTORE TRAP VECTOR
7012
7013  042432  022626            20$:    CMP     (SP)+,(SP)+         ;ADJUST STACK
7014  042434  012737  000006  000004  22$:    MOV     #ERRVEC+2,ERRVEC  ;RESTORE TRAP CATCHER
7015  042442  005037  000006            CLR     ERRVEC+2
7016  042446  000207                    RTS     PC                  ;RETURN
7017
7018                          .SBTTL  MEMORY CHECK ENABLE TRAP
7019
7020  042450  012737  042464  001202  MEMERR: MOV     #10$,$ESCAPE        ;LOAD ESCAPE
7021  042456  011637  004272            MOV     (SP),TRAPPC         ;STORE PC
7022  042462  104262                    ERROR   262                 ;REPORT MEM PARITY ERROR
7023  042464  005037  001202  10$:      CLR     $ESCAPE             ;CLEAR ESCAPE
7024  042470  032777  001000  136442    BIT     #SW9,@SWR           ;CHECK IF LOOP ON ERROR
7025  042476  001001                    BNE     15$                 ;YES, FORCE STACK AND TRY AGAIN
7026  042500  000002                    RTI                         ;NO, RETURN
7027
7028  042502  012706  001100  15$:      MOV     #STACK,SP           ;INITIALIZE STACK
7029  042506  000177  136376            JMP     @$LPERR             ;LOOP ON ERROR
7030
7031                          .SBTTL  SCOPE HANDLER ROUTINE
7032
7033                          ;;**********************************************************
7034                          ;*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
7035                          ;*AND LOAD THE TEST NUMBER($TSTNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
7036                          ;*AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
7037                          ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
7038                          ;*SW14=1        LOOP ON TEST
7039                          ;*SW11=1        INHIBIT ITERATIONS
7040                          ;*SW09=1        LOOP ON ERROR
7041                          ;*SW08=1        LOOP ON TEST IN SUB<7:0>
7042                          ;*CALL
7043                          ;*      SCOPE           ;;SCOPE=IOT
7044
7045  042512                  $SCOPE:
7046  042512  104407                    CKSWR                       ;;TEST  FOR CHANGE IN SOFT-SWR
7047  042514  032777  040000  136416  1$:      BIT     #BIT14,@SWR         ;;LOOP ON PRESENT TEST?
7048  042522  001131                    BNE     $OVER               ;;YES IF SW14=1
7049                          ;*****START OF CODE FOR THE XOR TESTER*****
7050  042524  000416          $XTSTR: BR      6$                  ;;IF RUNNING ON THE "XOR" TESTER CHANGE
7051                                                              ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
7052  042526  013746  000004            MOV     @#ERRVEC,-(SP)      ;;SAVE THE CONTENTS OF THE ERROR VECTOR
7053  042532  012737  042552  000004    MOV     #5$,@#ERRVEC        ;;SET FOR TIMEOUT
7054  042540  005737  177060            TST     @#177060            ;;TIME OUT ON XOR?
7055  042544  012637  000004            MOV     (SP)+,@#ERRVEC      ;;RESTORE THE ERROR VECTOR
7056  042550  000500                    BR      $SVLAD              ;;GO TO THE NEXT TEST
7057  042552  022626          5$:     CMP     (SP)+,(SP)+         ;;CLEAR THE STACK AFTER A TIME OUT
7058  042554  012637  000004            MOV     (SP)+,@#ERRVEC      ;;RESTORE THE ERROR VECTOR
7059  042560  000440                    BR      7$                  ;;LOOP ON THE PRESENT TEST
7060  042562          6$:;*****END OF CODE FOR THE XOR TESTER*****
```

```
7061  042562  032777  000400  136350          BIT     #BIT08,@SWR     ;;LOOP ON SPEC. TEST?
7062  042570  001421                          BEQ     2$              ;;BR IF NO
7063  042572  005046                          CLR     -(SP)           ;;CLEAR A TEMP. LOCATION
7064  042574  117716  136340                  MOVB    @SWR,(SP)       ;;PICKUP THE DESIRED TEST NUMBER
7065  042600  001414                          BEQ     8$              ;;BRANCH IF BAD TEST NUMBER IN SWR
7066  042602  022716  000100                  CMP     #100,(SP)       ;;CHECK THE NUMBER IN THE SWR
7067  042606  002411                          BLT     8$              ;;BRANCH IF TEST NUMBER IS OUT OF RANGE
7068  042610  011637  001102                  MOV     (SP),$TSTNM     ;;UPDATE THE TEST NUMBER
7069  042614  005316                          DEC     (SP)            ;;BACKUP BY ONE
7070  042616  006316                          ASL     (SP)            ;;SCALE THE TEST NUMBER AS AN INDEX
7071  042620  062716  043024                  ADD     #$SW08TBL,(SP)  ;;FORM THE ADDRESS OF TEST POINTER
7072  042624  013637  001106                  MOV     @(SP)+,$LPADR   ;;SET LOOP ADDRESS TO DESIRED TEST
7073  042630  000466                          BR      $OVER           ;;GO LOOP ON THE TEST
7074  042632  005726                  8$:     TST     (SP)+           ;;CLEAN THE BAD TEST NUMBER OFF OF THE STACK
7075  042634  105737  001103          2$:     TSTB    $ERFLG          ;;HAS AN ERROR OCCURRED?
7076  042640  001421                          BEQ     3$              ;;BR IF NO
7077  042642  123737  001115  001103          CMPB    $ERMAX,$ERFLG   ;;MAX. ERRORS FOR THIS TEST OCCURRED?
7078  042650  101015                          BHI     3$              ;;BR IF NO
7079  042652  032777  001000  136260          BIT     #BIT09,@SWR     ;;LOOP ON ERROR?
7080  042660  001404                          BEQ     4$              ;;BR IF NO
7081  042662  013737  001110  001106  7$:     MOV     $LPERR,$LPADR   ;;SET LOOP ADDRESS TO LAST SCOPE
7082  042670  000446                          BR      $OVER
7083  042672  105037  001103          4$:     CLRB    $ERFLG          ;;ZERO THE ERROR FLAG
7084  042676  005037  001200                  CLR     $TIMES          ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
7085  042702  000415                          BR      1$              ;;ESCAPE TO THE NEXT TEST
7086  042704  032777  004000  136226  3$:     BIT     #BIT11,@SWR     ;;INHIBIT ITERATIONS?
7087  042712  001011                          BNE     1$              ;;BR IF YES
7088  042714  005737  001222                  TST     $PASS           ;;IF FIRST PASS OF PROGRAM
7089  042720  001406                          BEQ     1$              ;;      INHIBIT ITERATIONS
7090  042722  005237  001104                  INC     $ICNT           ;;INCREMENT ITERATION COUNT
7091  042726  023737  001200  001104          CMP     $TIMES,$ICNT    ;;CHECK THE NUMBER OF ITERATIONS MADE
7092  042734  002024                          BGE     $OVER           ;;BR IF MORE ITERATION REQUIRED
7093  042736  012737  000001  001104  1$:     MOV     #1,$ICNT        ;;REINITIALIZE THE ITERATION COUNTER
7094  042744  013737  043022  001200          MOV     $MXCNT,$TIMES   ;;SET NUMBER OF ITERATIONS TO DO
7095  042752  105237  001102          $SVLAD: INCB    $TSTNM          ;;COUNT TEST NUMBERS
7096  042756  113737  001102  00.220          MOVB    $TSTNM,$TESTN   ;;SET TEST NUMBER IN APT MAILBOX
7097  042764  011637  001106                  MOV     (SP),$LPADR     ;;SAVE SCOPE LOOP ADDRESS
7098  042770  011637  001110                  MOV     (SP),$LPERR     ;;SAVE ERROR LOOP ADDRESS
7099  042774  005037  001202                  CLR     $ESCAPE         ;;CLEAR THE ESCAPE FROM ERROR ADDRESS
7100  043000  112737  000001  001115          MOVB    #1,$ERMAX       ;;ONLY ALLOW ONE(1) ERROR ON NEXT TEST
7101  043006  013777  001102  136126  $OVER:  MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER
7102  043014  013716  001106                  MOV     $LPADR,(SP)     ;;FUDGE RETURN ADDRESS
7103  043020  000002                          RTI                     ;;FIXES PS
7104  043022  003720                  $MXCNT: 2000.                   ;;MAX. NUMBER OF ITERATIONS
7105  043024                          $SW08TBL:
7106  043024  005274                          .WORD   TST1+2          ;;STARTING ADDRESS OF TEST 1
7107  043026  005600                          .WORD   TST2+2          ;;STARTING ADDRESS OF TEST 2
7108  043030  006060                          .WORD   TST3+2          ;;STARTING ADDRESS OF TEST 3
7109  043032  006324                          .WORD   TST4+2          ;;STARTING ADDRESS OF TEST 4
7110  043034  006636                          .WORD   TST5+2          ;;STARTING ADDRESS OF TEST 5
7111  043036  007200                          .WORD   TST6+2          ;;STARTING ADDRESS OF TEST 6
7112  043040  007516                          .WORD   TST7+2          ;;STARTING ADDRESS OF TEST 7
7113  043042  010034                          .WORD   TST10+2         ;;STARTING ADDRESS OF TEST 10
7114  043044  010352                          .WORD   TST11+2         ;;STARTING ADDRESS OF TEST 11
7115  043046  010616                          .WORD   TST12+2         ;;STARTING ADDRESS OF TEST 12
7116  043050  011062                          .WORD   TST13+2         ;;STARTING ADDRESS OF TEST 13
```

```
7117  043052  011400                .WORD   TST14+2          ;;STARTING ADDRESS OF TEST 14
7118  043054  011710                .WORD   TST15+2          ;;STARTING ADDRESS OF TEST 15
7119  043056  012242                .WORD   TST16+2          ;;STARTING ADDRESS OF TEST 16
7120  043060  012604                .WORD   TST17+2          ;;STARTING ADDRESS OF TEST 17
7121  043062  013050                .WORD   TST20+2          ;;STARTING ADDRESS OF TEST 20
7122  043064  013330                .WORD   TST21+2          ;;STARTING ADDRESS OF TEST 21
7123  043066  013610                .WORD   TST22+2          ;;STARTING ADDRESS OF TEST 22
7124  043070  014070                .WORD   TST23+2          ;;STARTING ADDRESS OF TEST 23
7125  043072  014350                .WORD   TST24+2          ;;STARTING ADDRESS OF TEST 24
7126  043074  014630                .WORD   TST25+2          ;;STARTING ADDRESS OF TEST 25
7127  043076  015142                .WORD   TST26+2          ;;STARTING ADDRESS OF TEST 26
7128  043100  015454                .WORD   TST27+2          ;;STARTING ADDRESS OF TEST 27
7129  043102  015766                .WORD   TST30+2          ;;STARTING ADDRESS OF TEST 30
7130  043104  016300                .WORD   TST31+2          ;;STARTING ADDRESS OF TEST 31
7131  043106  016612                .WORD   TST32+2          ;;STARTING ADDRESS OF TEST 32
7132  043110  017110                .WORD   TST33+2          ;;STARTING ADDRESS OF TEST 33
7133  043112  017422                .WORD   TST34+2          ;;STARTING ADDRESS OF TEST 34
7134  043114  017660                .WORD   TST35+2          ;;STARTING ADDRESS OF TEST 35
7135  043116  020124                .WORD   TST36+2          ;;STARTING ADDRESS OF TEST 36
7136  043120  020460                .WORD   TST37+2          ;;STARTING ADDRESS OF TEST 37
7137  043122  021024                .WORD   TST40+2          ;;STARTING ADDRESS OF TEST 40
7138  043124  021350                .WORD   TST41+2          ;;STARTING ADDRESS OF TEST 41
7139  043126  021714                .WORD   TST42+2          ;;STARTING ADDRESS OF TEST 42
7140  043130  022170                .WORD   TST43+2          ;;STARTING ADDRESS OF TEST 43
7141  043132  022720                .WORD   TST44+2          ;;STARTING ADDRESS OF TEST 44
7142  043134  023160                .WORD   TST45+2          ;;STARTING ADDRESS OF TEST 45
7143  043136  023564                .WORD   TST46+2          ;;STARTING ADDRESS OF TEST 46
7144  043140  024014                .WORD   TST47+2          ;;STARTING ADDRESS OF TEST 47
7145  043142  024450                .WORD   TST50+2          ;;STARTING ADDRESS OF TEST 50
7146  043144  025132                .WORD   TST51+2          ;;STARTING ADDRESS OF TEST 51
7147  043146  025564                .WORD   TST52+2          ;;STARTING ADDRESS OF TEST 52
7148  043150  026216                .WORD   TST53+2          ;;STARTING ADDRESS OF TEST 53
7149  043152  026634                .WORD   TST54+2          ;;STARTING ADDRESS OF TEST 54
7150  043154  027266                .WORD   TST55+2          ;;STARTING ADDRESS OF TEST 55
7151  043156  027720                .WORD   TST56+2          ;;STARTING ADDRESS OF TEST 56
7152  043160  030352                .WORD   TST57+2          ;;STARTING ADDRESS OF TEST 57
7153  043162  031004                .WORD   TST60+2          ;;STARTING ADDRESS OF TEST 60
7154  043164  031436                .WORD   TST61+2          ;;STARTING ADDRESS OF TEST 61
7155  043166  032070                .WORD   TST62+2          ;;STARTING ADDRESS OF TEST 62
7156  043170  032522                .WORD   TST63+2          ;;STARTING ADDRESS OF TEST 63
7157  043172  033154                .WORD   TST64+2          ;;STARTING ADDRESS OF TEST 64
7158  043174  033572                .WORD   TST65+2          ;;STARTING ADDRESS OF TEST 65
7159  043176  034224                .WORD   TST66+2          ;;STARTING ADDRESS OF TEST 66
7160  043200  034706                .WORD   TST67+2          ;;STARTING ADDRESS OF TEST 67
7161  043202  035370                .WORD   TST70+2          ;;STARTING ADDRESS OF TEST 70
7162  043204  036052                .WORD   TST71+2          ;;STARTING ADDRESS OF TEST 71
7163  043206  036534                .WORD   TST72+2          ;;STARTING ADDRESS OF TEST 72
7164  043210  037216                .WORD   TST73+2          ;;STARTING ADDRESS OF TEST 73
7165  043212  037700                .WORD   TST74+2          ;;STARTING ADDRESS OF TEST 74
7166  043214  040212                .WORD   TST75+2          ;;STARTING ADDRESS OF TEST 75
7167  043216  040640                .WORD   TST76+2          ;;STARTING ADDRESS OF TEST 76
7168  043220  041302                .WORD   TST77+2          ;;STARTING ADDRESS OF TEST 77
7169  043222  041714                .WORD   TST100+2                ;;STARTING ADDRESS OF TEST 100
7170                  ;*****************************************************************
7171                  ;SBTTL  LOOP ON INTERNAL ERROR
7172
```

```
7173  043224  032777  001000  135706  SCOP1$: BIT     #SW9,@SWR               ;CHECK IF LOOP ON ERROR
7174  043232  001405                          BEQ     5$                      ;NO, CONTINUE
7175  043234  105737  001103                  TSTB    $ERFLG                  ;CHECK IF ERROR OCCURRED
7176  043240  001402                          BEQ     5$                      ;NO, CONTINUE
7177  043242  013716  001110                  MOV     $LPERR,(SP)             ;LOAD ERROR RETURN
7178  043246  000002                  5$:     RTI                             ;RETURN
7179                                  .SBTTL  APT COMMUNICATIONS ROUTINE
7180
7181                                  ;;**************************************************************************
7182  043250  112737  000001  043514 $ATY1:  MOVB    #1,$FFLG        ;;IO REPORT FATAL ERROR
7183  043256  112737  000001  043512 $ATY3:  MOVB    #1,$MFLG        ;;TO TYPE A MESSAGE
7184  043264  000403                          BR      $ATYC
7185  043266  112737  000001  043514 $ATY4:  MOVB    #1,$FFLG        ;;TO ONLY REPORT FATAL ERROR
7186  043274                          $ATYC:
7187  043274  010046                          MOV     R0,-(SP)        ;;PUSH R0 ON STACK
7188  043276  010146                          MOV     R1,-(SP)        ;;PUSH R1 ON STACK
7189  043300  105737  043512                  TSTB    $MFLG           ;;SHOULD TYPE A MESSAGE?
7190  043304  001450                          BEQ     5$              ;;IF NOT:  BR
7191  043306  122737  000001  001234          CMPB    #APTENV,$ENV    ;;OPERATING UNDER APT?
7192  043314  001031                          BNE     3$              ;;IF NOT:  BR
7193  043316  132737  000100  001235          BITB    #APTSPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
7194  043324  001425                          BEQ     3$              ;;IF NOT:  BR
7195  043326  017600  000004                  MOV     @4(SP),R0       ;;GET MESSAGE ADDR.
7196  043332  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
7197  043340  005737  001214          1$:     TST     $MSGTYPE        ;;SEE IF DONE W/ LAST XMISSION?
7198  043344  001375                          BNE     1$              ;;IF NOT:  WAIT
7199  043346  010037  001230                  MOV     R0,$MSGAD       ;;PUT ADDR IN MAILBOX
7200  043352  105720                  2$:     TSTB    (R0)+           ;;FIND END OF MESSAGE
7201  043354  001376                          BNE     2$
7202  043356  163700  001230                  SUB     $MSGAD,R0       ;;SUB START OF MESSAGE
7203  043362  006200                          ASR     R0              ;;GET MESSAGE LNGTH IN WORDS
7204  043364  010037  001232                  MOV     R0,$MSGLGT      ;;PUT LENGTH IN MAILBOX
7205  043370  012737  000004  001214          MOV     #4,$MSGTYPE     ;;TELL APT TO TAKE MSG.
7206  043376  000413                          BR      5$
7207  043400  017637  000004  043424  3$:     MOV     @4(SP),4$       ;;PUT MSG ADDR IN JSR LINKAGE
7208  043406  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDRESS
7209  043414  013746  177776                  MOV     177776,-(SP)    ;;PUSH 177776 ON STACK
7210  043420  004737  044200                  JSR     PC,$TYPE        ;;CALL TYPE MACRO
7211  043424  000000                  4$:     .WORD   0
7212  043426                          5$:
7213  043426  105737  043514          10$:    TSTB    $FFLG           ;;SHOULD REPORT FATAL ERROR?
7214  043432  001416                          BEQ     12$             ;;IF NOT:  BR
7215  043434  005737  001234                  TST     $ENV            ;;RUNNING UNDER APT?
7216  043440  001413                          BEQ     12$             ;;IF NOT:  BR
7217  043442  005737  001214          11$:    TST     $MSGTYPE        ;;FINISHED LAST MESSAGE?
7218  043446  001375                          BNE     11$             ;;IF NOT:  WAIT
7219  043450  017637  000004  001216          MOV     @4(SP),$FATAL   ;;GET ERROR #
7220  043456  062766  000002  000004          ADD     #2,4(SP)                ;;BUMP RETURN ADDR.
7221  043464  005237  001214                  INC     $MSGTYPE        ;;TELL APT TO TAKE ERROR
7222  043470  105037  043514          12$:    CLRB    $FFLG           ;;CLEAR FATAL FLAG
7223  043474  105037  043513                  CLRB    $LFLG           ;;CLEAR LOG FLAG
7224  043500  105037  043512                  CLRB    $MFLG           ;;CLEAR MESSAGE FLAG
7225  043504  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
7226  043506  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
7227  043510  000207                          RTS     PC              ;;RETURN
7228  043512     000          $MFLG:  .BYTE   0               ;;MESSG. FLAG
```

```
7229  043513  000                    $LFLG:  .BYTE   0               ;;LOG FLAG
7230  043514  000                    $FFLG:  .BYTE   0               ;;FATAL F_AG
7231          043516                         .EVEN
7232          000200                 APTSIZE=200
7233          000001                 APTENV=001
7234          000100                 APTSPOOL=100
7235          000040                 APTCSUP=040
7236                                 .SBTTL  ERROR HANDLER ROUTINE
7237
7238                                 ;;*****************************************************************
7239                                 ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
7240                                 ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
7241                                 ;*AND GO TO TYPERR ON ERROR
7242                                 ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
7243                                 ;*SW15=1        HALT ON ERROR
7244                                 ;*SW13=1        INHIBIT ERROR TYPEOUTS
7245                                 ;*SW10=1        BELL ON ERROR
7246                                 ;*SW09=1        LOOP ON ERROR
7247                                 ;*CALL
7248                                 ;*      ERROR   N       ;;ERROR=EMT AND N=ERROR ITEM NUMBER
7249
7250  043516                         $ERROR:
7251  043516  104407                         CKSWR                   ;;TEST FOR CHANGE IN SOFT-SWR
7252  043520  105237  001103         7$:     INCB    $ERFLG          ;;SET THE ERROR FLAG
7253  043524  001775                         BEQ     7$              ;;DON'T LET THE FLAG GO TO ZERO
7254  043526  013777  001102  135406         MOV     $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
7255  043534  032777  002000  135376         BIT     #BIT10,@SWR     ;;BELL ON ERROR?
7256  043542  001402                         BEQ     1$              ;;NO - SKIP
7257  043544  104401  001204                 TYPE    ,$BELL          ;;RING BELL
7258  043550  005237  001112         1$:     INC     $ERTTL          ;;COUNT THE NUMBER OF ERRORS
7259  043554  011637  001116                 MOV     (SP),$ERRPC     ;;GET ADDRESS OF ERROR INSTRUCTIO'
7260  043560  162737  000002  001116         SUB     #2,$ERRPC
7261  043566  117737  135324  001114         MOVB    @$ERRPC,$ITEMB  ;;STRIP AND SAVE THE ERROR ITEM CODE
7262  043574  032777  020000  135336         BIT     #BIT13,@SWR     ;;SKIP TYPEOUT IF SET
7263  043602  001004                         BNE     20$             ;;SKIP TYPEOUTS
7264  043604  004737  043716                 JSR     PC,TYPERR       ;;GO TO USER ERROR ROUTINE
7265  043610  104401  001211                 TYPE    ,$CRLF
7266  043614                         20$:
7267  043614  122737  000001  001234         CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
7268  043622  001007                         BNE     2$              ;;NO,SKIP APT ERROR REPORT
7269  043624  113737  001114  043636         MOVB    $ITEMB,21$      ;;SET ITEM NUMBER AS ERROR NUMBER
7270  043632  004737  043266                 JSR     PC,$ATY4        ;;REPORT FATAL ERROR TO APT
7271  043636     000         21$:    .BYTE   0
7272  043637     000                 .BYTE   0
7273  043640  000777         22$:    BR      22$             ;;APT ERROR LOOP
7274  043642  005777  135272  2$:     TST     @SWR            ;;HALT ON ERROR
7275  043646  100002                 BPL     3$              ;;SKIP IF CONTINUE
7276  043650  000000                 HALT                    ;;HALT ON ERROR!
7277  043652  104407                 CKSWR                   ;;TEST FOR CHANGE IN SOFT-SWR
7278  043654  032777  001000  135256 3$:     BIT     #BIT09,@SWR     ;;LOOP ON ERROR SWITCH SET?
7279  043662  001402                 BEQ     4$              ;;BR IF NO
7280  043664  013716  001110                 MOV     $LPERR,(SP)     ;;FUDGE RETURN FOR LOOPING
7281  043670  005737  001202         4$:     TST     $ESCAPE         ;;CHECK FOR AN ESCAPE ADDRESS
7282  043674  001402                 BEQ     5$              ;;BR IF NONE
7283  043676  013716  001202                 MOV     $ESCAPE,(SP)    ;;FUDGE RETURN ADDRESS FOR ESCAPE
7284  043702                         5$:
```

```
7285  043702  022737  042340  000042          CMP     #$ENDAD,#42     ;;ACT-11 AUTO-ACCEPT?
7286  043710  001001                          BNE     6$              ;;BRANCH IF NO
7287  043712  000000                          HALT                    ;;YES
7288  043714                          6$:
7289  043714  000002                          RTI                     ;;RETURN
7290
7291                                  ;;*****************************************************************
7292                                  .SBTTL  TYPE ERROR ROUTINE
7293                                  ;*ENTRY JSR PC,TYPERR
7294                                  ;*RETURN RTS PC
7295                                  ;*
7296                                  ;*THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
7297                                  ;*ERROR IS TO BE REPORTED. IT THEN USES THE "ERROR TABLE" ($ERRTB)
7298                                  ;*ENTRY TO DEFINE WHAT INFORMATION IS TO BE REPORTED CONCERNING
7299                                  ;*THE ERROR.
7300                                  ;*****************************************************************
7301  043716  104413                  TYPERR: SAVREG
7302  043720  113700  001114                  MOVB    $ITEMB,R0       ;ENTER ERROR NUMBER
7303  043724  042700  177400                  BIC     #177400,R0      ;CLEAR UNUSED BITS
7304  043730  005300                          DEC     R0              ;FORM INDEX FOR ERROR TABLE
7305  043732  006300                          ASL     R0
7306  043734  006300                          ASL     R0
7307  043736  006300                          ASL     R0
7308  043740  062700  001300          1$:     ADD     #$ERRTB,R0      ;FORM ADDRESS OF ERROR ENTRY
7309  043744  012037  043760                  MOV     (R0)+,2$        ;GET EM POINTER
7310  043750  001404                          BEQ     3$              ;BRANCH IF THERE ISN'T ONE
7311  043752  104401  001211                  TYPE    ,$CRLF          ;TYPE CARRIAGE RETURN LINE FEED
7312  043756  104401                          TYPE                    ;TYPE ERROR MESSAGE (EM)
7313  043760  000000          2$:     .WORD   0                       ;EM POINTER GOES HERE
7314  043762  012037  043776          3$:     MOV     (R0)+,4$        ;GET DH POINTER
7315  043766  001404                          BEQ     5$              ;BRANCH IF THERE ISN'T ONE
7316  043770  104401  001211                  TYPE    ,$CRLF          ;TYPE CR-LF
7317  043774  104401                          TYPE                    ;TYPE DATA HEADER
7318  043776  000000          4$:     .WORD   0                       ;DH POINTER GOES HERE
7319  044000  012001          5$:     MOV     (R0)+,R1        ;GET DT POINTER
7320  044002  001445                  BEQ     20$             ;BRANCH IF THERE ARE NONE
7321  044004  005004                  CLR     R4              ;RESET INDENT SWITCH
7322  044006  012000                  MOV     (R0)+,R0        ;GET DF POINTER
7323  044010  012002                  MOV     (R0)+,R2        ;STORE NUMBER OF DH'S
7324  044012  104401  001211          TYPE    ,$CRLF          ;TYPE <CR><LF>
7325  044016  112003          10$:    MOVB    (R0)+,R3        ;GET & STORE NUMBER OF DATA WORDS
7326  044020  105720                  TSTB    (R0)+           ;BUMP PAST FORMAT WORD
7327  044022  005703                  TST     R3              ;TEST IF ANY DATA FOR THIS HEADER
7328  044024  001416                  BEQ     14$             ;NO - SKIP DATA PRINT
7329  044026  005704                  TST     R4              ;CHECK FOR INDENT
7330  044030  001004                  BNE     12$             ;YES, GO INDENT
7331  044032  013146          11$:    MOV     @(R1)+,-(SP)    ;PUT FIRST DATA WORD ON STACK
7332  044034  104402                  TYPOC                   ;TYPE IT
7333  044036  005303                  DEC     R3              ;MORE DATA WORDS
7334  044040  001403                  BEQ     13$             ;NO-BRANCH
7335  044042  104401  050117  12$:    TYPE    ,SPACE2         ;TYPE SEPARATORS
7336  044046  000771                  BR      11$             ;LOOP
7337  044050  104401  001211  13$:    TYPE    ,$CRLF          ;TYPE <CR><LF>
7338  044054  005710                  TST     (R0)            ;CHECK IF NEXT HEADER AVAILIBLE
7339  044056  001401                  BEQ     14$             ;NO, DO NOT CHANGE INDENT
7340  044060  005104                  COM     R4              ;CHANGE INDENT
```

```
7341  044062  005302                    14$:    DEC     R2              ;MORE DH'S?
7342  044064  003414                            BLE     20$             ;NO-BRANCH
7343  044066  012037  044106            15$:    MOV     (R0)+, @#       ;GET NEXT DH POINTER
7344  044072  001751                            BEQ     10$             ;IF NO HEADER GO GET DATA
7345  044074  005704                            TST     R4              ;INDENT?
7346  044076  001402                            BEQ     17$             ;NO-BRANCH
7347  044100  104401  050117                    TYPE    ,SPACE2         ;YES-TYPE SPACES
7348  044104  104401                    17$:    TYPE                    ;TYPE DH
7349  044106  000000                    18$:    .WORD   0               ;DH POINTER GOES HERE
7350  044110  104401  001211                    TYPE    $CRLF
7351  044114  000740                            BR      10$             ;GO TYPE OUT DATA
7352  044116  104414                    20$:    RESREG
7353  044120  005237  004242                    INC     ERRCNT          ;INCREMENT THE ERROR COUNT
7354  044124  032777  010000  135006            BIT     #SW12,@SWR      ;CHECK IF SWITCH 12 SET
7355  044132  001421                            BEQ     25$             ;NO, RETURN
7356  044134  022737  000024  004242            CMP     #20.,ERRCNT     ;CHECK IF ERROR THRESHOLD EXCEEDED
7357  044142  103015                            BHIS    25$             ;NO, RETURN
7358  044144  104401  050122                    TYPE    ,ABORT          ;TYPE "PROGRAM ABORTED BECAUSE ERROR
7359                                                                    ;  THRESHOLD EXCEEDED"
7360  044150  005737  000042                    TST     42              ;CHECK IF IN CHAIN MODE
7361  044154  001407                            BEQ     22$             ;NO, HALT PROCESSOR
7362  044156  012737  000001  042176            MOV     #1,$EOPCT       ;FOR PASS COUNT FOR ABORT
7363  044164  012706  001100                    MOV     #STACK,SP       ;INITIALIZE STACK
7364  044170  000137  042.50                    JMP     $EOP            ;BRING IN NEXT PROGRAM
7365
7366  044174  000000                    22$:    HALT
7367  044176  000207                    25$:    RTS     PC
7368                                    .SBTTL  TYPE ROUTINE
7369
7370                                    ;;**********************************************************************
7371                                    ;*ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
7372                                    ;*THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
7373                                    ;*NOTE1:        $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
7374                                    ;*NOTE2:        $FILLS CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
7375                                    ;*NOTE3:        $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
7376                                    ;*
7377                                    ;*CALL:
7378                                    ;*1) USING A TRAP INSTRUCTION
7379                                    ;*      TYPE    ,MESADR         ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
7380                                    ;*OR
7381                                    ;*      TYPE
7382                                    ;*      MESADR
7383                                    ;*
7384
7385  044200  105737  001157            $TYPE:  TSTB    $TPFLG          ;;IS THERE A TERMINAL?
7386  044204  100002                            BPL     1$              ;;BR IF YES
7387  044206  000000                            HALT                    ;;HALT HERE IF NO TERMINAL
7388  044210  000430                            BR      3$              ;;LEAVE
7389  044212  010046                    1$:     MOV     R0,-(SP)        ;;SAVE R0
7390  044214  017600  000002                    MOV     @2(SP),R0       ;;GET ADDRESS OF ASCIZ STRING
7391  044220  122737  000001  001234            CMPB    #APTENV,$ENV    ;;RUNNING IN APT MODE
7392  044226  001011                            BNE     62$             ;;NO,GO CHECK FOR APT CONSOLE
7393  044230  132737  000100  001235            BITB    #APTSPOOL,$ENVM ;;SPOOL MESSAGE TO APT
7394  044236  001405                            BEQ     62$             ;;NO,GO CHECK FOR CONSOLE
7395  044240  010037  044250                    MOV     R0,61$          ;;SETUP MESSAGE ADDRESS FOR APT
7396  044244  004737  043256                    JSR     PC,$ATY3        ;;SPOOL MESSAGE TO APT
```

```
 7397  044250  000000                  61$:    .WORD   0                ;;MESSAGE ADDRESS
 7398  044252  132737  000040  001235  62$:    BITB    #APTCSUP,$ENVM   ;;APT CONSOLE SUPPRESSED
 7399  044260  001003                          BNE     60$              ;;YES,SKIP TYPE OUT
 7400  044262  112046                  2$:     MOVB    (R0)+,-(SP)      ;;PUSH CHARACTER TO BE TYPED ONTO STACK
 7401  044264  001005                          BNE     4$               ;;BR IF IT ISN'T THE TERMINATOR
 7402  044266  005726                          TST     (SP)+            ;;IF TERMINATOR POP IT OFF THE STACK
 7403  044270  012600                  60$:    MOV     (SP)+,R0         ;;RESTORE R0
 7404  044272  062716  000002          3$:     ADD     #2,(SP)          ;;ADJUST RETURN PC
 7405  044276  000002                          RTI                      ;;RETURN
 7406  044300  122716  000011          4$:     CMPB    #HT,(SP)         ;;BRANCH IF <HT>
 7407  044304  001430                          BEQ     8$
 7408  044306  122716  000200                  CMPB    #CRLF,(SP)       ;;BRANCH IF NOT <CRLF>
 7409  044312  001006                          BNE     5$
 7410  044314  005726                          TST     (SP)+            ;;POP  <CR><LF> EQUIV
 7411  044316  104401                          TYPE                     ;;TYPE A CR AND LF
 7412  044320  001211                          $CRLF
 7413  044322  105037  044456                  CLRB    $CHARCNT         ;;CLEAR CHARACTER COUNT
 7414  044326  000755                          BR      2$               ;;GET NEXT CHARACTER
 7415  044330  004737  044412          5$:     JSR     PC,$TYPEC        ;;GO TYPE THIS CHARACTER
 7416  044334  123726  001156          6$:     CMPB    $FILLC,(SP)+     ;;IS IT TIME FOR FILLER CHARS.?
 7417  044340  001350                          BNE     2$               ;;IF NO GO GET NEXT CHAR.
 7418  044342  013746  001154                  MOV     $NULL,-(SP)      ;;GET # OF FILLER CHARS. NEEDED
 7419                                                                   ;;AND THE NULL CHAR.
 7420  044346  105366  000001          7$:     DECB    1(SP)            ;;DOES A NULL NEED TO BE TYPED?
 7421  044352  002770                          BLT     6$               ;;BR IF NO--GO POP THE NULL OFF OF STACK
 7422  044354  004737  044412                  JSR     PC,$TYPEC        ;;GO TYPE A NULL
 7423  044360  105337  044456                  DECB    $CHARCNT         ;;DO NOT COUNT AS A COUNT
 7424  044364  000770                          BR      7$               ;;LOOP
 7425
 7426                                  ;HORIZONTAL TAB PROCESSOR
 7427
 7428  044366  112716  000040          8$:     MOVB    #' ,(SP)         ;;REPLACE TAB WITH SPACE
 7429  044372  004737  044412          9$:     JSR     PC,$TYPEC        ;;TYPE A SPACE
 7430  044376  132737  000007  044456          BITB    #7,$CHARCNT      ;;BRANCH IF NOT AT
 7431  044404  001372                          BNE     9$               ;;TAB STOP
 7432  044406  005726                          TST     (SP)+            ;;POP SPACE OFF STACK
 7433  044410  000724                          BR      2$               ;;GET NEXT CHARACTER
 7434  044412  105777  134532  $TYPEC: TSTB    @$TPS                    ;;WAIT UNTIL PRINTER IS READY
 7435  044416  100375                          BPL     $TYPEC
 7436  044420  116677  000002  134524          MOVB    2(SP),@$TPB      ;;LOAD CHAR TO BE TYPED INTO DATA REG.
 7437  044426  122766  000015  000002          CMPB    #CR,2(SP)        ;;IS CHARACTER A CARRIAGE RETURN?
 7438  044434  001003                          BNE     1$               ;;BRANCH IF NO
 7439  044436  105037  044456                  CLRB    $CHARCNT         ;;YES--CLEAR CHARACTER COUNT
 7440  044442  000406                          BR      $TYPEX           ;;EXIT
 7441  044444  122766  000012  000002  1$:     CMPB    #LF,2(SP)        ;;IS CHARACTER A LINE FEED?
 7442  044452  001402                          BEQ     $TYPEX           ;;BRANCH IF YES
 7443  044454  105227                          INCB    (PC)+            ;;COUNT THE CHARACTER
 7444  044456  000000          $CHARCNT:.WORD  0                        ;;CHARACTER COUNT STORAGE
 7445  044460  000207          $TYPEX: RTS     PC
 7446
 7447                                  .SBTTL  BINARY TO OCTAL (ASCII) AND TYPE
 7448
 7449                                  ;;******************************************************************
 7450                                  ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
 7451                                  ;*OCTAL (ASCII) NUMBER AND TYPE IT.
 7452                                  ;*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
```

```
                                        ;*CALL:
                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
                                        ;*      TYPOS                   ;;CALL FOR TYPEOUT
                                        ;*      .BYTE   N               ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
                                        ;*      .BYTE   M               ;;M=1 OR 0
                                        ;*                                      ;;1=TYPE LEADING ZEROS
                                        ;*                                      ;;0=SUPPRESS LEADING ZEROS
                                        ;*
                                        ;*$TYPON----ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
                                        ;*$TYPOS OR $TYPOC
                                        ;*CALL:
                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
                                        ;*      TYPON                   ;;CALL FOR TYPEOUT
                                        ;*
                                        ;*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
                                        ;*CALL:
                                        ;*      MOV     NUM,-(SP)       ;;NUMBER TO BE TYPED
                                        ;*      TYPOC                   ;;CALL FOR TYPEOUT
                                        ;*
044462  017646  000000          $TYPOS: MOV     @(SP),-(SP)     ;;PICKUP THE MODE
044466  116637  000001  044705          MOVB    1(SP),$OFILL    ;;LOAD ZERO FILL SWITCH
044474  112637  044707                  MOVB    (SP)+,$OMODE+1  ;;NUMBER OF DIGITS TO TYPE
044500  062716  000002                  ADD     #2,(SP)         ;;ADJUST RETURN ADDRESS
044504  000406                          BR      $TYPON
044506  112737  000001  044705  $TYPOC: MOVB    #1,$OFILL       ;;SET THE ZERO FILL SWITCH
044514  112737  000006  044707          MOVB    #6,$OMODE+1     ;;SET FOR SIX(6) DIGITS
044522  112737  000005  044704  $TYPON: MOVB    #5,$OCNT        ;;SET THE ITERATION COUNT
044530  010346                          MOV     R3,-(SP)        ;;SAVE R3
044532  010446                          MOV     R4,-(SP)        ;;SAVE R4
044534  010546                          MOV     R5,-(SP)        ;;SAVE R5
044536  113704  044707                  MOVB    $OMODE+1,R4     ;;GET THE NUMBER OF DIGITS TO TYPE
044542  005404                          NEG     R4
044544  062704  000006                  ADD     #6,R4           ;;SUBTRACT IT FOR MAX. ALLOWED
044550  110437  044706                  MOVB    R4,$OMODE       ;;SAVE IT FOR USE
044554  113704  044705                  MOVB    $OFILL,R4       ;;GET THE ZERO FILL SWITCH
044560  016605  000012                  MOV     12(SP),R5       ;;PICKUP THE INPUT NUMBER
044564  005003                          CLR     R3              ;;CLEAR THE OUTPUT WORD
044566  006105                  1$:     ROL     R5              ;;ROTATE MSB INTO "C"
044570  000404                          BR      3$              ;;GO DO MSB
044572  006105                  2$:     ROL     R5              ;;FORM THIS DIGIT
044574  006105                          ROL     R5
044576  006105                          ROL     R5
044600  010503                          MOV     R5,R3
044602  006103                  3$:     ROL     R3              ;;GET LSB OF THIS DIGIT
044604  105337  044706                  DECB    $OMODE          ;;TYPE THIS DIGIT?
044610  100016                          BPL     7$              ;;BR IF NO
044612  042703  177770                  BIC     #177770,R3      ;;GET RID OF JUNK
044616  001002                          BNE     4$              ;;TEST FOR 0
044620  005704                          TST     R4              ;;SUPPRESS THIS 0?
044622  001403                          BEQ     5$              ;;BR IF YES
044624  005204                  4$:     INC     R4              ;;DON'T SUPPRESS ANYMORE 0'S
044626  052703  000060                  BIS     #'0,R3          ;;MAKE THIS DIGIT ASCII
044632  052703  000040          5$:     BIS     #' ,R3          ;;MAKE ASCII IF NOT ALREADY
044636  110337  044702                  MOVB    R3,8$           ;;SAVE FOR TYPING
044642  104401  044702                  TYPE    ,8$             ;;GO TYPE THIS DIGIT
044646  105337  044704          7$:     DECB    $OCNT           ;;COUNT BY 1
```

```
7509  044652  003347                        BGT     2$              ;;BR IF  ORE TO DO
7510  044654  002402                        BLT     6$              ;;BR IF DONE
7511  044656  005204                        INC     R4              ;;INSURE LAST DIGIT ISN'T A BLANK
7512  044660  000744                        BR      2$              ;;GO DO THE LAST DIGIT
7513  044662  012605                6$:     MOV     (SP)+,R5        ;;RESTORE R5
7514  044664  012604                        MOV     (SP)+,R4        ;;RESTORE R4
7515  044666  012603                        MOV     (SP)+,R3        ;;RESTORE R3
7516  044670  016666  000002  000004       MOV     2(SP),4(SP)     ;;SET THE STACK FOR RETURNING
7517  044676  012616                        MOV     (SP)+,(SP)
7518  044700  000002                        RTI                     ;;RETURN
7519  044702     000                8$:     .BYTE   0               ;;STORAGE FOR ASCII DIGIT
7520  044703     000                        .BYTE   0               ;;TERMINATOR FOR TYPE ROUTINE
7521  044704     000                $OCNT:  .BYTE   0               ;;OCTAL DIGIT COUNTER
7522  044705     000                $OFILL: .BYTE   0               ;;ZERO FILL SWITCH
7523  044706  000000                $OMODE: .WORD   0               ;;NUMBER OF DIGITS TO TYPE
7524                                .SBTTL  CONVERT BINARY TO DECIMAL AND TYPE ROUTINE
7525
7526                                ;;*****************************************************************
7527                                ;*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
7528                                ;*SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
7529                                ;*NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
7530                                ;*BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
7531                                ;*REPLACED WITH SPACES.
7532                                ;*CALL:
7533                                ;*      MOV     NUM,-(SP)       ;;PUT THE BINARY NUMBER ON THE STACK
7534                                ;*      TYPOS                   ;;GO TO THE ROUTINE
7535
7536  044710                        $TYPDS:
7537  044710  010046                        MOV     R0,-(SP)        ;;PUSH R0 ON STACK
7538  044712  010146                        MOV     R1,-(SP)        ;;PUSH R1 ON STACK
7539  044714  010246                        MOV     R2,-(SP)        ;;PUSH R2 ON STACK
7540  044716  010346                        MOV     R3,-(SP)        ;;PUSH R3 ON STACK
7541  044720  010546                        MOV     R5,-(SP)        ;;PUSH R5 ON STACK
7542  044722  012746  020200                MOV     #20200,-(SP)    ;;SET BLANK SWITCH AND SIGN
7543  044726  016605  000020                MOV     20(SP),R5       ;;GET THE INPUT NUMBER
7544  044732  100004                        BPL     1$              ;;BR IF INPUT IS POS.
7545  044734  005405                        NEG     R5              ;;MAKE THE BINARY NUMBER POS.
7546  044736  112766  000055  000001       MOVB    #'-,1(SP)       ;;MAKE THE ASCII NUMBER NEG.
7547  044744  005000                1$:     CLR     R0              ;;ZERO THE CONSTANTS INDEX
7548  044746  012703  045124                MOV     #$DBLK,R3       ;;SETUP THE OUTPUT POINTER
7549  044752  112723  000040                MOVB    #' ,(R3)+       ;;SET THE FIRST CHARACTER TO A BLANK
7550  044756  005002                2$:     CLR     R2              ;;CLEAR THE BCD NUMBER
7551  044760  016001  045114                MOV     $DTBL(R0),R1    ;;GET THE CONSTANT
7552  044764  160105                3$:     SUB     R1,R5           ;;FORM THIS BCD DIGIT
7553  044766  002402                        BLT     4$              ;;BR IF DONE
7554  044770  005202                        INC     R2              ;;INCREASE THE BCD DIGIT BY 1
7555  044772  000774                        BR      3$
7556  044774  060105                4$:     ADD     R1,R5           ;;ADD BACK THE CONSTANT
7557  044776  005702                        TST     R2              ;;CHECK IF BCD DIGIT=0
7558  045000  001002                        BNE     5$              ;;FALL THROUGH IF 0
7559  045002  105716                        TSTB    (SP)            ;;STILL DOING LEADING 0'S?
7560  045004  100407                        BMI     7$              ;;BR IF YES
7561  045006  106316                5$:     ASLB    (SP)            ;;MSD?
7562  045010  103003                        BCC     6$              ;;BR IF NO
7563  045012  116663  000001  177777       MOVB    1(SP),-1(R3)    ;;YES--SET THE SIGN
7564  045020  052702  000060                BIS     #'0,R2          ;;MAKE THE BCD DIGIT ASCII
```

```
7565  045024  052702  000040          7$:     BIS     #' ,R2          ;;MAKE IT A SPACE IF NOT ALREADY A DIGIT
7566  045030  110223                          MOVB    R2,(R3)+        ;;PUT THIS CHARACTER IN THE OUTPUT BUFFER
7567  045032  005720                          TST     (R0)+           ;;JUST INCREMENTING
7568  045034  020027  000010                 CMP     R0,#10          ;;CHECK THE TABLE INDEX
7569  045040  002746                          BLT     2$              ;;GO DO THE NEXT DIGIT
7570  045042  003002                          BGT     8$              ;;GO TO EXIT
7571  045044  010502                          MOV     R5,R2           ;;GET THE LSD
7572  045046  000764                          BR      6$              ;;GO CHANGE TO ASCII
7573  045050  105726                  8$:     TSTB    (SP)+           ;;WAS THE LSD THE FIRST NON-ZERO?
7574  045052  100003                          BPL     9$              ;;BR IF NO
7575  045054  116663  177777  177776        MOVB    -1(SP),-2(R3)   ;;YES--SET THE SIGN FOR TYPING
7576  045062  105013                  9$:     CLRB    (R3)            ;;SET THE TERMINATOR
7577  045064  012605                          MOV     (SP)+,R5        ;;POP STACK INTO R5
7578  045066  012603                          MOV     (SP)+,R3        ;;POP STACK INTO R3
7579  045070  012602                          MOV     (SP)+,R2        ;;POP STACK INTO R2
7580  045072  012601                          MOV     (SP)+,R1        ;;POP STACK INTO R1
7581  045074  012600                          MOV     (SP)+,R0        ;;POP STACK INTO R0
7582  045076  104401  045124                 TYPE    ,$DBLK          ;;NOW TYPE THE NUMBER
7583  045102  016666  000002  000004         MOV     2(SP),4(SP)     ;;ADJUST THE STACK
7584  045110  012616                          MOV     (SP)+,(SP)
7585  045112  000002                          RTI                     ;;RETURN TO USER
7586  045114  023420                  $DTBL:  10000.
7587  045116  001750                          1000.
7588  045120  000144                          100.
7589  045122  000012                          10.
7590  045124  000004                  $DBLK:  .BLKW   4
7591                                  .SBTTL  TTY INPUT ROUTINE
7592
7593                                  ;;******************************************************************
7594                                  .ENABL  LSB
7595
7596                                  ;;******************************************************************
7597                                  ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
7598                                  ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
7599                                  ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
7600                                  ;*WHEN OPERATING IN TTY FLAG MODE.
7601  045134  022737  000176  001140  $CKSWR: CMP     #SWREG,SWR      ;;IS THE SOFT-SWR SELECTED?
7602  045142  001074                          BNE     15$             ;;BRANCH IF NO
7603  045144  105777  133774                 TSTB    @$TKS           ;;CHAR THERE?
7604  045150  100071                          BPL     15$             ;;IF NO, DON'T WAIT AROUND
7605  045152  117746  133770                 MOVB    @$TKB,-(SP)     ;;SAVE THE CHAR
7606  045156  042716  177600                 BIC     #^C177,(SP)     ;;STRIP-OFF THE ASCII
7607  045162  022726  000007                 CMP     #7,(SP)+        ;;IS IT A CONTROL G?
7608  045166  001062                          BNE     15$             ;;NO, RETURN TO USER
7609  045170  123727  001134  000001         CMPB    $AUTOB,#1       ;;ARE WE RUNNING IN AUTO-MODE?
7610  045176  001456                          BEQ     15$             ;;BRANCH IF YES
7611
7612  045200  104401  046007                 TYPE    .$CNTLG         ;;ECHO THE CONTROL-G (^G)
7613  045204  104401  046014          $GTSWR: TYPE    ,$MSWR          ;;TYPE CURRENT CONTENTS
7614  045210  013746  000176                 MOV     SWREG,-(SP)     ;;SAVE SWREG FOR TYPEOUT
7615  045214  104402                          TYPOC                   ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
7616  045216  104401  046025                 TYPE    ,$MNEW          ;;PROMPT FOR NEW SWR
7617  045222  005046                  19$:    CLR     -(SP)           ;;CLEAR COUNTER
7618  045224  005046                          CLR     -(SP)           ;;THE NEW SWR
7619  045226  105777  133712          7$:     TSTB    @$TKS           ;;CHAR THERE?
7620  045232  100375                          BPL     7$              ;;IF NOT TRY AGAIN
```

```
7621
7622  045234  117746  133706                  MOVB    @$TKB,-(SP)     ;;PICK UP CHAR
7623  045240  042716  177600                  BIC     #^C177,(SP)     ;;MAKE IT 7-BIT ASCII
7624
7625
7626
7627  045244  021627  000025          9$:     CMP     (SP),#25        ;;IS IT A CONTROL-U?
7628  045250  001005                  BNE     10$                     ;;BRANCH IF NOT
7629  045252  104401  046002                  TYPE    ,$CNTLU         ;;YES, ECHO CONTROL-U (^U)
7630  045256  062706  000006          20$:    ADD     #6,SP           ;;IGNORE PREVIOUS INPUT
7631  045262  000757                  BR      19$                     ;;LET'S TRY IT AGAIN
7632
7633
7634  045264  021627  000015          10$:    CMP     (SP),#15        ;;IS IT A <CR>?
7635  045270  001022                  BNE     16$                     ;;BRANCH IF NO
7636  045272  005766  000004                  TST     4(SP)           ;;YES, IS IT THE FIRST CHAR?
7637  045276  001403                  BEQ     11$                     ;;BRANCH IF YES
7638  045300  016677  000002  133632          MOV     2(SP),@SWR      ;;SAVE NEW SWR
7639  045306  062706  000006          11$:    ADD     #6,SP           ;;CLEAR UP STACK
7640  045312  104401  001211          14$:    TYPE    ,$CRLF          ;;ECHO <CR> AND <LF>
7641  045316  123727  001135  000001          CMPB    $INTAG,#1       ;;RE-ENABLE TTY KBD INTERRUPTS?
7642  045324  001003                  BNE     15$                     ;;BRANCH IF NOT
7643  045326  012777  000100  133610          MOV     #100,@$TKS      ;;RE-ENABLE TTY KBD INTERRUPTS
7644  045334  000002          15$:    RTI                             ;;RETURN
7645  045336  004737  044412          16$:    JSR     PC,$TYPEC       ;;ECHO CHAR
7646  045342  021627  000060                  CMP     (SP),#60        ;;CHAR < 0?
7647  045346  002420                  BLT     18$                     ;;BRANCH IF YES
7648  045350  021627  000067                  CMP     (SP),#67        ;;CHAR > 7?
7649  045354  003015                  BGT     18$                     ;;BRANCH IF YES
7650  045356  042726  000060                  BIC     #60,(SP)+       ;;STRIP-OFF ASCII
7651  045362  005766  000002                  TST     2(SP)           ;;IS THIS THE FIRST CHAR
7652  045366  001403                  BEQ     17$                     ;;BRANCH IF YES
7653  045370  006316                  ASL     (SP)                    ;;NO, SHIFT PRESENT
7654  045372  006316                  ASL     (SP)                    ;;   CHAR OVER TO MAKE
7655  045374  006316                  ASL     (SP)                    ;;   ROOM FOR NEW ONE.
7656  045376  005266  000002          17$:    INC     2(SP)           ;;KEEP COUNT OF CHAR
7657  045402  056616  177776                  BIS     -2(SP),(SP)     ;;SET IN NEW CHAR
7658  045406  000707                  BR      7$                      ;;GET THE NEXT ONE
7659  045410  104401  001210          18$:    TYPE    ,$QUES          ;;TYPE ?<CR><LF>
7660  045414  000720                  BR      20$                     ;;SIMULATE CONTROL-U
7661                                  .DSABL  LSB
7662
7663
7664                                  ;;*****************************************************************
7665                                  ;*THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
7666                                  ;*CALL:
7667                                  ;*      RDCHR                   ;;INPUT A SINGLE CHARACTER FROM THE TTY
7668                                  ;*      RETURN HERE             ;;CHARACTER IS ON THE STACK
7669                                  ;*                              ;;WITH PARITY BIT STRIPPED OFF
7670                                  ;
7671
7672  045416  011646                  $RDCHR: MOV     (SP),-(SP)      ;;PUSH DOWN THE PC
7673  045420  016666  000004  000002          MOV     4(SP),2(SP)     ;;SAVE THE PS
7674  045426  105777  133512          1$:     TSTB    @$TKS           ;;WAIT FOR
7675  045432  100375                  BPL     1$                      ;;A CHARACTER
7676  045434  117766  133506  000004          MOVB    @$TKB,4(SP)     ;;READ THE TTY
```

```
 7677  045442  042766  177600  000004          BIC     #^C<177>,4(SP)  ;;GET RID OF JUNK IF ANY
 7678  045450  026627  000004  000023          CMP     4(SP),#23       ;;IS IT A CONTROL-S?
 7679  045456  001013                          BNE     3$              ;;BRANCH IF NO
 7680  045460  105777  133460          2$:     TSTB    @$TKS           ;;WAIT FOR A CHARACTER
 7681  045464  100375                          BPL     2$              ;;LOOP UNTIL ITS THERE
 7682  045466  117746  133454                  MOVB    @$TKB,-(SP)     ;;GET CHARACTER
 7683  045472  042716  177600                  BIC     #^C<177>,(SP)   ;;MAKE IT 7-BIT ASCII
 7684  045476  022627  000021                  CMP     (SP)+,#21       ;;IS IT A CONTROL-Q?
 7685  045502  001366                          BNE     2$              ;;IF NOT DISCARD IT
 7686  045504  000750                          BR      1$              ;;YES, RESUME
 7687  045506  026627  000004  000140  3$:     CMP     4(SP),#140      ;;IS IT UPPER CASE?
 7688  045514  002407                          BLT     4$              ;;BRANCH IF YES
 7689  045516  026627  000004  000175          CMP     4(SP),#175      ;;IS IT A SPECIAL CHAR?
 7690  045524  003003                          BGT     4$              ;;BRANCH IF YES
 7691  045526  042766  000040  000004          BIC     #40,4(SP)       ;;MAKE IT UPPER CASE
 7692  045534  000002                  4$:     RTI                     ;;GO BACK TO USER
 7693                                          ;;**************************************************************
 7694                                          ;*THIS ROUTINE WILL INPUT A STRING FROM THE TTY
 7695                                          ;*CALL:
 7696                                          ;*      RDLIN                   ;;INPUT A STRING FROM THE TTY
 7697                                          ;*      RETURN HERE             ;;ADDRESS OF FIRST CHARACTER WILL BE ON THE STACK
 7698                                          ;*                              ;;TERMINATOR WILL BE A BYTE OF ALL 0'S
 7699
 7700  045536  010346                  $RDLIN: MOV     R3,-(SP)        ;;SAVE R3
 7701  045540  005046                          CLR     -(SP)           ;;CLEAR THE RUBOUT KEY
 7702  045542  012703  045772          1$:     MOV     #$TTYIN,R3      ;;GET ADDRESS
 7703  045546  022703  046002          2$:     CMP     #$TTYIN+8.,R3   ;;BUFFER FULL?
 7704  045552  101456                          BLOS    4$              ;;BR IF YES
 7705  045554  104410                          RDCHR                   ;;GO READ ONE CHARACTER FROM THE TTY
 7706  045556  112613                          MOVB    (SP)+,(R3)      ;;GET CHARACTER
 7707  045560  122713  000177          10$:    CMPB    #177,(R3)       ;;IS IT A RUBOUT
 7708  045564  001022                          BNE     5$              ;;BR IF NO
 7709  045566  005716                          TST     (SP)            ;;IS THIS THE FIRST RUBOUT?
 7710  045570  001007                          BNE     6$              ;;BR IF NO
 7711  045572  112737  000134  045770          MOVB    #'\,9$          ;;TYPE A BACK SLASH
 7712  045600  104401  045770                  TYPE    ,9$
 7713  045604  012716  177777                  MOV     #-1,(SP)        ;;SET THE RUBOUT KEY
 7714  045610  005303                  6$:     DEC     R3              ;;BACKUP BY ONE
 7715  045612  020327  045772                  CMP     R3,#$TTYIN      ;;STACK EMPTY?
 7716  045616  103434                          BLO     4$              ;;BR IF YES
 7717  045620  111337  045770                  MOVB    (R3),9$         ;;SETUP TO TYPEOUT THE DELETED CHAR.
 7718  045624  104401  045770                  TYPE    ,9$             ;;GO TYPE
 7719  045630  000746                          BR      2$              ;;GO READ ANOTHER CHAR.
 7720  045632  005716                  5$:     TST     (SP)            ;;RUBOUT KEY SET?
 7721  045634  001406                          BEQ     7$              ;;BR IF NO
 7722  045636  112737  000134  045770          MOVB    #'\,9$          ;;TYPE A BACK SLASH
 7723  045644  104401  045770                  TYPE    ,9$
 7724  045650  005016                          CLR     (SP)            ;;CLEAR THE RUBOUT KEY
 7725  045652  122713  000025          7$:     CMPB    #25,(R3)        ;;IS CHARACTER A CTRL U?
 7726  045656  001003                          BNE     8$              ;;BR IF NO
 7727  045660  104401  046002                  TYPE    ,$CNTLU         ;;TYPE A CONTROL "U"
 7728  045664  000726                          BR      1$              ;;GO START OVER
 7729  045666  122713  000022          8$:     CMPB    #22,(R3)        ;;IS CHARACTER A "^R"?
 7730  045672  001011                          BNE     3$              ;;BRANCH IF NO
 7731  045674  105013                          CLRB    (R3)            ;;CLEAR THE CHARACTER
 7732  045676  104401  001211                  TYPE    ,$CRLF          ;;TYPE A "CR" & "LF"
```

```
7733  045702  104401  045772          TYPE    ,$TTYIN         ;;TYPE THE INPUT STRING
7734  045706  000717                  BR      2$              ;;GO PICKUP ANOTHER CHACTER
7735  045710  104401  001210  4$:     TYPE    ,$QUES          ;;TYPE A '?'
7736  045714  000712                  BR      1$              ;;CLEAR THE BUFFER AND LOOP
7737  045716  111337  045770  3$:     MOVB    (R3),9$         ;;ECHO THE CHARACTER
7738  045722  104401  045770          TYPE    ,9$
7739  045726  122723  000015          CMPB    #15,(R3)+       ;;CHECK FOR RETURN
7740  045732  001305                  BNE     2$              ;;LOOP IF NOT RETURN
7741  045734  105063  177777          CLRB    -1(R3)          ;;CLEAR RETURN (THE 15)
7742  045740  104401  001212          TYPE    ,$LF            ;;TYPE A LINE FEED
7743  045744  005726                  TST     (SP)+           ;;CLEAN RUBOUT KEY FROM THE STACK
7744  045746  012603                  MOV     (SP)+,R3        ;;RESTORE R3
7745  045750  011646                  MOV     (SP),-(SP)      ;;ADJUST THE STACK AND PUT ADDRESS OF THE
7746  045752  016666  000004  000002  MOV     4(SP),2(SP)     ;;      FIRST ASCII CHARACTER ON IT
7747  045760  012766  045772  000004  MOV     #$TTYIN,4(SP)
7748  045766  000002                  RTI                     ;;RETURN
7749  045770     000  9$:     .BYTE   0               ;;STORAGE FOR ASCII CHAR. TO TYPE
7750  045771     000          .BYTE   0               ;;TERMINATOR
7751  045772  000010  $TTYIN: .BLKB   8.              ;;RESERVE 8 BYTES FOR TTY INPUT
7752  046002  052536  005015     000  $CNTLU: .ASCIZ  /↑U/<15><12>    ;;CONTROL "U"
7753  046007     136  006507  000012  $CNTLG: .ASCIZ  /↑G/<15><12>    ;;CONTROL "G"
7754  046014  005015  053523  020122  $MSWR:  .ASCIZ  <15><12>/SWR = /
7755  046022  020075     000
7756  046025     040  047040  053505  $MNEW:  .ASCIZ  /  NEW = /
7757  046032  036440  000040
7758                                  .SBTTL  READ AN OCTAL NUMBER FROM THE TTY
7759
7760                                  ;;**************************************************************
7761                                  ;*THIS ROUTINE WILL READ AN OCTAL (ASCII) NUMBER FROM THE TTY AND
7762                                  ;*CHANGE IT TO BINARY.
7763                                  ;*THE INPUT CHARACTERS WILL BE CHECKED TO INSURED THEY ARE LEGAL
7764                                  ;*OCTAL DIGITS. IF AN ILLEGAL CHARACTER IS READ A "?" WILL BE TYPED
7765                                  ;*FOLLOWED BY A CARRIAGE RETURN-LINE FEED. THE COMPLETE NUMBER MUST
7766                                  ;*THEN BE RETYPED. THE INPUT IS TERMINATED BY TYPING A CARRIAGE RETURN.
7767                                  ;*CALL:
7768                                  ;*      RDOCT                   ;;READ AN OCTAL NUMBER
7769                                  ;*      RETURN HERE             ;;LOW ORDER BITS ARE ON TOP OF THE STACK
7770                                  ;*                              ;;HIGH ORDER BITS ARE IN $HIOCT
7771
7772  046036  011646          $RDOCT: MOV     (SP),-(SP)      ;;PROVIDE SPACE FOR THE
7773  046040  016666  000004  000002  MOV     4(SP),2(SP)     ;;INPUT NUMBER
7774  046046  010046                  MOV     R0,-(SP)        ;;PUSH R0 ON STACK
7775  046050  010146                  MOV     R1,-(SP)        ;;PUSH R1 ON STACK
7776  046052  010246                  MOV     R2,-(SP)        ;;PUSH R2 ON STACK
7777  046054  104411          1$:     RDLIN                   ;;READ AN ASCIZ LINE
7778  046056  012600                  MOV     (SP)+,R0        ;;GET ADDRESS OF 1ST CHARACTER
7779  046060  010037  046164          MOV     R0,5$           ;;AND SAVE IT
7780  046064  005001                  CLR     R1              ;;CLEAR DATA WORD
7781  046066  005002                  CLR     R2
7782  046070  112046          2$:     MOVB    (R0)+,-(SP)     ;;PICKUP THIS CHARACTER
7783  046072  001420                  BEQ     3$              ;;IF ZERO GET OUT
7784  046074  122716  000060          CMPB    #'0,(SP)        ;;MAKE SURE THIS CHARACTER
7785  046100  003026                  BGT     4$              ;;IS AN OCTAL DIGIT
7786  046102  122716  000067          CMPB    #'7,(SP)
7787  046106  002423                  BLT     4$
7788  046110  006301                  ASL     R1              ;;*2
```

```
7789  046112  006102                        ROL     R2
7790  046114  006301                        ASL     R1              ;;*4
7791  046116  006102                        ROL     R2
7792  046120  006301                        ASL     R1              ;;*8
7793  046122  006102                        ROL     R2
7794  046124  042716  177770                BIC     #^C7,(SP)       ;;STRIP THE ASCII JUNK
7795  046130  062601                        ADD     (SP)+,R1        ;;ADD IN THIS DIGIT
7796  046132  000756                        BR      2$              ;;LOOP
7797  046134  005726                3$:     TST     (SP)+           ;;CLEAN TERMINATOR FROM STACK
7798  046136  010166  000012                MOV     R1,12(SP)       ;;SAVE THE RESULT
7799  046142  010237  046174                MOV     R2,$HIOCT
7800  046146  012602                        MOV     (SP)+,R2        ;;POP STACK INTO R2
7801  046150  012601                        MOV     (SP)+,R1        ;;POP STACK INTO R1
7802  046152  012600                        MOV     (SP)+,R0        ;;POP STACK INTO R0
7803  046154  000002                        RTI                     ;;RETURN
7804  046156  005726                4$:     TST     (SP)+           ;;CLEAN PARTIAL FROM STACK
7805  046160  105010                        CLRB    (R0)            ;;SET A TERMINATOR
7806  046162  104401                        TYPE                    ;;TYPE UP THRU THE BAD CHAR.
7807  046164  000000                5$:     .WORD   0
7808  046166  104401  001210                TYPE    ,$QUES          ;;"?" "CR" & "LF"
7809  046172  000730                        BR      1$              ;;TRY AGAIN
7810  046174  000000                $HIOCT: .WORD   0               ;;HIGH ORDER BITS GO HERE
7811                                        .SBTTL  SAVE AND RESTORE R0-R5 ROUTINES
7812
7813                                        ;;******************************************************************
7814                                        ;*SAVE R0-R5
7815                                        ;*CALL:
7816                                        ;*      SAVREG
7817                                        ;*UPON RETURN FROM $SAVREG THE STACK WILL LOOK LIKE:
7818                                        ;*
7819                                        ;*TOP---(+16)
7820                                        ;* +2---(+18)
7821                                        ;* +4---R5
7822                                        ;* +6---R4
7823                                        ;* +8---R3
7824                                        ;*+10---R2
7825                                        ;*+12---R1
7826                                        ;*+14---R0
7827
7828  046176                        $SAVREG:
7829  046176  010046                        MOV     R0,-(SP)        ;;PUSH R0 ON STACK
7830  046200  010146                        MOV     R1,-(SP)        ;;PUSH R1 ON STACK
7831  046202  010246                        MOV     R2,-(SP)        ;;PUSH R2 ON STACK
7832  046204  010346                        MOV     R3,-(SP)        ;;PUSH R3 ON STACK
7833  046206  010446                        MOV     R4,-(SP)        ;;PUSH R4 ON STACK
7834  046210  010546                        MOV     R5,-(SP)        ;;PUSH R5 ON STACK
7835  046212  016646  000022                MOV     22(SP),-(SP)    ;;SAVE PS OF MAIN FLOW
7836  046216  016646  000022                MOV     22(SP),-(SP)    ;;SAVE PC OF MAIN FLOW
7837  046222  016646  000022                MOV     22(SP),-(SP)    ;;SAVE PS OF CALL
7838  046226  016646  000022                MOV     22(SP),-(SP)    ;;SAVE PC OF CALL
7839  046232  000002                        RTI
7840
7841                                        ;*RESTORE R0-R5
7842                                        ;*CALL:
7843                                        ;*      RESREG
7844  046234                        $RESREG:
```

```
7845  046234  012666  000022          MOV     (SP)+,22(SP)    ;;RESTORE PC OF CALL
7846  046240  012666  000022          MOV     (SP)+,22(SP)    ;;RESTORE PS OF CALL
7847  046244  012666  000022          MOV     (SP)+,22 SP)    ;;RESTORE PC OF MAIN FLOW
7848  046250  012666  000022          MOV     (SP)+,22(SP)    ;;RESTORE PS OF MAIN FLOW
7849  046254  012605                  MOV     (SP)+,R5        ;;POP STACK INTO R5
7850  046256  012604                  MOV     (SP)+,R4        ;;POP STACK INTO R4
7851  046260  012603                  MOV     (SP)+,R3        ;;POP STACK INTO R3
7852  046262  012602                  MOV     (SP)+,R2        ;;POP STACK INTO R2
7853  046264  012601                  MOV     (SP)+,R1        ;;POP STACK INTO R1
7854  046266  012600                  MOV     (SP)+,R0        ;;POP STACK INTO R0
7855  046270  000002                  RTI
7856
7857                            .SBTTL  POWER DOWN AND UP ROUTINES
7858
7859                            ;;*****************************************************************
7860                            ;POWER DOWN ROUTINE
7861  046272  017737  132642  004274 $PWRDN: MOV     @SWR,SAVSWR     ;SAVE SWITCH REG
7862  046300  012737  046320  000024         MOV     #$PWRUP,PWRVEC  ;SET UP VECTOR
7863  046306  012737  000340  000026         MOV     #PR7,PWRVEC+2
7864  046314  000000                  HALT
7865  046316  000776                  BR      .-2             ;HANG UP
7866
7867                            ;;*****************************************************************
7868                            ;POWER UP ROUTINE
7869  046320  005037  046410       $PWRUP: CLR     $PWRCT          ;LOOP LOOP TIMER
7870  046324  012737  000144  046412         MOV     #100.,$PWRCT+2
7871  046332  005237  046410       1$:     INC     $PWRCT          ;WAIT FOR TELETYPE
7872  046336  001375                  BNE     1$
7873  046340  005337  046412          DEC     $PWRCT+2
7874  046344  001372                  BNE     1$
7875  046346  012737  046272  000024         MOV     #$PWRDN,PWRVEC  ;SET UP THE POWER DOWN VECTOR
7876  046354  012737  000340  000026         MOV     #PR7,PWRVEC+2
7877  046362  012706  001100          MOV     #STACK,SP       ;FORCE STACK POINTER
7878  046366  104401  046414          TYPE    ,$POWER         ;TYPE POWER
7879  046372  004737  042360          JSR     PC,CHKPAP       ;CHECK FOR MEMORY CHECK ENABLE OPTION
7880  046376  013777  004274  132534  MOV     SAVSWR,@SWR     ;RESTORE SWITCH REG
7881  046404  000177  132476          JMP     @$LPADR         ;START TEST AGAIN
7882
7883  04641C  000000  000000       $PWRCT: .WORD   0,0             ;COUNTER FOR TELETYPE
7884  046414  005015  047520  042527 $POWER: .ASCIZ  <15><12>/POWER/
7885  046422  000122
7886                                  .EVEN
7887                            .SBTTL  TRAP DECODER
7888
7889                            ;;*****************************************************************
7890                            ;*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
7891                            ;*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
7892                            ;*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
7893                            ;*GO TO THAT ROUTINE.
7894
7895  046424  010046             $TRAP:  MOV     R0,-(SP)        ;;SAVE R0
7896  046426  016600  000002          MOV     2(SP),R0        ;;GET TRAP ADDRESS
7897  046432  005740                  TST     -(R0)           ;;BACKUP BY 2
7898  046434  111000                  MOVB    (R0),R0         ;;GET RIGHT BYTE OF TRAP
7899  046436  006300                  ASL     R0              ;;POSITION FOR INDEXING
7900  046440  016000  046460          MOV     $TRPAD(R0),R0   ;;INDEX TO TABLE
```

```
 7901  046444  000200                          RTS     R0              ;;GO TO ROUTINE
 7902
 7903
 7904                                  ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
 7905
 7906  046446  011646                  $TRAP2: MOV     (SP),-(SP)      ;;MOVE THE PC DOWN
 7907  046450  016666  000004  000002          MOV     4(SP),2(SP)     ;;MOVE THE PSW DOWN
 7908  046456  000002                          RTI                     ;;RESTORE THE PSW
 7909
 7910                                  .SBTTL  TRAP TABLE
 7911
 7912                                  ;*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
 7913                                  ;*BY THE "TRAP" INSTRUCTION.
 7914
 7915                                  ;       ROUTINE
 7916                                  ;       -------
 7917  046460  046446                  $TRPAD: .WORD   $TRAP2
 7918  046462  044200                          $TYPE   ;;CALL=TYPE     TRAP+1(104401)  TTY TYPEOUT ROUTINE
 7919  046464  044506                          $TYPOC  ;;CALL=TYPOC    TRAP+2(104402)  TYPE OCTAL NUMBER (WITH LEADING ZEROS)
 7920  046466  044462                          $TYPOS  ;;CALL=TYPOS    TRAP+3(104403)  TYPE OCTAL NUMBER (NO LEADING ZEROS)
 7921  046470  044522                          $TYPON  ;;CALL=TYPON    TRAP+4(104404)  TYPE OCTAL NUMBER (AS PER LAST CALL)
 7922  046472  044710                          $TYPDS  ;;CALL=TYPDS    TRAP+5(104405)  TYPE DECIMAL NUMBER (WITH SIGN)
 7923
 7924  046474  045204                          $GTSWR  ;;CALL=GTSWR    TRAP+6(104406)  GET SOFT SWR SETTING
 7925
 7926  046476  045134                          $CKSWR  ;;CALL=CKSWR    TRAP+7(104407)  TEST FOR CHANGE IN SOFT-SWR
 7927  046500  045416                          $RDCHR  ;;CALL=RDCHR    TRAP+10(104410) TTY TYPEIN CHARACTER ROUTINE
 7928  046502  045536                          $RDLIN  ;;CALL=RDLIN    TRAP+11(104411) TTY TYPEIN STRING ROUTINE
 7929  046504  046036                          $RDOCT  ;;CALL=RDOCT    TRAP+12(104412) READ AN OCTAL NUMBER FROM TTY
 7930  046506  046176                          $SAVREG ;;CALL=SAVREG   TRAP+13(104413) SAVE R0-R5 ROUTINE
 7931  046510  046234                          $RESREG ;;CALL=RESREG   TRAP+14(104414) RESTORE R0-R5 ROUTINE
 7932  046512  043224                          SCOP1$  ;;CALL=SCOP1    TRAP+15(104415) INTERNAL LOOP ON ERROR
```

```
                                        SBTTL  DATA PRINTED BY ERROR ROUTINES

046514  001220  004272          DT000:  .WORD   $TESTN,TRAPPC
046520  001220  001116  004160  DT001:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,E.MR2,T.MR2,E.MR3,T.MR3
046526  004120  004206  004146
046534  004210  004150
046540  001220  001116  004160  DT002:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,DRVCOD,E.MR2,T.MR2,E.MR3,T.MR3
046546  004120  004244  004206
046554  004146  004210  004150
046562  001220  001116  004160  DT006:  .WORD   $TESTN,$ERRPC E.CS1,T.CS1,HDCODE,E.MR2,T.MR2,E.MR3,T.MR3
046570  004120  004250  004206
046576  004146  004210  004150
046604  001220  001116  004160  DT012:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,E.MR1,T.MR1,MSGCOD
046612  004120  004204  004144
046620  004246
046622  004206  004146  004210          .WORD   E.MR2,T.MR2,E.MR3,T.MR3
046630  004150
046632  001220  001116  004160  DT017:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,CYLIN,E.MR2,T.MR2,E.MR3,T.MR3
046640  004120  004252  004206
046646  004146  004210  004150
046654  001220  001116  004160  DT031:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,OFFVAL,E.MR2,T.MR2,E.MR3,T.MR3
046662  004120  004254  004206
046670  004146  004210  004150
046676  001220  001116  004160  DT035:   WORD   $TESTN,$ERRPC,E.CS1,T.CS1,CYLIN,OFFVAL
046704  004120  004252  004254
046712  004206  004146  004210          .WORD   E.MR2,T.MR2,E.MR3,T.MR3
046720  004150
046722  001220  001116  004230  DT050:  .WORD   $TESTN,$ERRPC,U.MR2,U.MR3,SFTCNT,E.MR2,T.MR2,E.MR3,T.MR3
046730  004232  004256  004206
046736  004146  004210  004150
046744  001220  001116  004210  DT052:  .WORD   $TESTN,$ERRPC,E.MR3,T.MR3,E.MR2,T.MR2
046752  004150  004206  004146
046760  001220  001116  004160  DT062:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,E.CS2,T CS2,E.DS,T.DS,E.ER,T.ER
046766  004120  004170  004130
046774  004172  004132  004174
047002  004134
047004  001220  001116  004160  DT065:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1
047012  004120
047014  001220  001116  004206  DT067:  .WORD   $TESTN,$ERRPC,E.MR2,T.MR2,E.MR3,T.MR3
047022  004146  004210  004150
047030  001220  001116          DT100:  .WORD   $TESTN,$ERRPC
047034  001220  001116  004160  DT126:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1
047042  004120
047044  001220  001116  004160  DT224:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,E.CS2,T.CS2,E.DS,T.DS
047052  004120  004170  004130
047060  004172  004132
047064  004174  004134  004220          .WORD   E.ER,T.ER,P.CS1,P.CS2,P.DS,P.ER
047072  004222  004224  004226
047100  001220  001116  004266  DT230:  .WORD   $TESTN,$ERRPC,DRVTYP,CYLIN,HDCODE,E.CS1,T.CS1,E.CS2,T.CS2
047106  004252  004250  004160
047114  004120  004170  004130
047122  004172  004132  004174          .WORD   E.DS,T.DS,E.ER,T.ER
047130  004134
047132  001220  001116  004160  DT256:  .WORD   $TESTN,$ERRPC,E.CS1,T.CS1,E.ER,T.ER,ILLFUN
047140  004120  004174  004134
047146  004270
```

```
                                        .SBTTL  DATA FORMATS
                                
                                DF000:  .WORD   1
047150  000001                          .BYTE   2,0
047152      002         000
                                DF001:  .WORD   7                       ;ERROR 1
047154  000007                          .BYTE   0,0
047156      000         000
047160  050207                          .WORD   DH0C0A
047162      000         000             .BYTE   0,0
047164  050225                          .WORD   DH000B
047166      002         000             .BYTE   2,0
047170  050271                          .WORD   DH001A
047172      000         000             .BYTE   0,0
047174  050310                          .WORD   DH001B
047176      002         000             .BYTE   2,0
047200  050326                          .WORD   DH001C
047202      000         000             .BYTE   0,0
047204  050365                          .WORD   DH001D
047206      004         000             .BYTE   4,0
047210  000007                  DF002:  .WORD   7                       ;ERRORS 2-5
047212      000         000             .BYTE   0,0
047214  050207                          .WORD   DH000A
047216      000         000             .BYTE   0,0
047220  050225                          .WORD   DH000B
047222      002         000             .BYTE   2,0
047224  050424                          .WORD   DH002A
047226      000         000             .BYTE   0,0
047230  050452                          .WORD   DH002B
047232      003         000             .BYTE   3,0
047234  050326                          .WORD   DH001C
047236      000         000             .BYTE   0,0
047240  050365                          .WORD   DH001D
047242      004         000             .BYTE   4,0
047244  000007                  DF006:  .WORD   7                       ;ERRORS 6-11
047246      000         000             .BYTE   0,0
047250  050207                          .WORD   DH000A
047252      000         000             .BYTE   0,0
047254  050225                          .WORD   DH000B
047256      002         000             .BYTE   2,0
047260  050501                          .WORD   DH006A
047262      000         000             .BYTE   0,0
047264  050526                          .WORD   DH006B
047266      003         000             .BYTE   3,0
047270  050326                          .WORD   DH001C
047272      000         000             .BYTE   0,0
047274  050365                          .WORD   DH001D
047276      004         000             .BYTE   4,0
047300  000007                  DF012:  .WORD   7                       ;ERRORS12-16
047302      000         000             .BYTE   0,0
047304  050207                          .WORD   DH000A
047306      000         000             .BYTE   0,0
047310  050225                          .WORD   DH000B
047312      002         000             .BYTE   2,0
047314  050552                          .WORD   DH012A
047316      000         000             .BYTE   0,0
047320  050617                          .WORD   DH012B
047322      005         000             .BYTE   5,0
```

```
8045  047324  050326                        .WORD   DH001C
8046  047326     000      000               .BYTE   0,0
8047  047330  050365                        .WORD   DH001D
8048  047332     004      000               .BYTE   4,0
8049  047334  000007              DF017:    .WORD   7                       ;ERRORS 17-30
8050  047336     000      000               .BYTE   0,0
8051  047340  050207                        .WORD   DH00DA
8052  047342     000      000               .BYTE   0,0
8053  047344  050225                        .WORD   DH0008
8054  047346     002      000               .BYTE   2,0
8055  047350  050665                        WORD    DH017A
8056  047352     000      000               .BYTE   0,0
8057  047354  050713                        .WORD   DH017B
8058  047356     003      000               .BYTE   3,0
8059  047360  050326                        .WORD   DH001C
8060  047362     000      000               .BYTE   0,0
8061  047364  050365                        .WORD   DH001D
8062  047366     004      000               .BYTE   4,0
8063  047370  000007              DF031:    .WORD   7                       ;ERRORS 31-34
8064  047372     000      000               .BYTE   0,0
8065  047374  050207                        .WORD   DH00DA
8066  047376     000      000               .BYTE   0,0
8067  047400  050225                        .WORD   DH0008
8068  047402     002      000               .BYTE   2,0
8069  047404  050737                        .WORD   DH031A
8070  047406     000      000               .BYTE   0,0
8071  047410  050766                        .WORD   DH031B
8072  047412     003      000               .BYTE   3,0
8073  047414  050326                        .WORD   DH001C
8074  047416     000      000               .BYTE   0,0
8075  047420  050365                        .WORD   DH001D
8076  047422     004      000               .BYTE   4,0
8077  047424  000007              DF035:    .WORD   7                       ;ERROR 35-41
8078  047426     000      000               .BYTE   0,0
8079  047430  050207                        .WORD   DH00DA
8080  047432     000      000               .BYTE   0,0
8081  047434  050225                        .WORD   DH0008
8082  047436     002      000               .BYTE   2,0
8083  047440  051014                        .WORD   DH035A
8084  047442     000      000               .BYTE   0,0
8085  047444  051053                        .WORD   DH035B
8086  047446     004      000               .BYTE   4,0
8087  047450  050326                        .WORD   DH001C
8088  047452     000      000               .BYTE   0,0
8089  047454  050365                        .WORD   DH001D
8090  047456     004      000               .BYTE   4,0
8091  047460  000007              DF050:    .WORD   7                       ;ERRORS 50 & 51
8092  047462     000      000               .BYTE   0,0
8093  047464  050207                        .WORD   DH00DA
8094  047466     000      000               .BYTE   0,0
8095  047470  050225                        .WORD   DH0008
8096  047472     002      000               .BYTE   2,0
8097  047474  051111                        .WORD   DH050A
8098  047476     000      000               .BYTE   0,0
8099  047500  051137                        .WORD   DH050B
8100  047502     003      000               .BYTE   3,0
```

```
8101  047504  050326          .WORD   DH001C
8102  047506  000     000     .BYTE   0,0
8103  047510  050365          .WORD   DH001D
8104  047512  004     000     .BYTE   4,0
8105  047514  000005  DF052:  .WORD   5               ;ERRORS 52-61
8106  047516  000     000     .BYTE   0,0
8107  047520  050207          .WORD   DH0G0A
8108  047522  000     000     .BYTE   0,0
8109  047524  050225          .WORD   DH0008
8110  047526  002     000     .BYTE   2,0
8111  047530  050326          WORD    DH001C
8112  047532  000     000     .BYTE   0,0
8113  047534  050365          .WORD   DH001D
8114  047536  004     000     .BYTE   4,0
8115  047540  000005  DF062:  .WORD   5               ;ERRORS 62-64
8116  047542  000     000     .BYTE   0,0
8117  047544  050207          .WORD   DH000A
8118  047546  000     000     .BYTE   0,0
8119  047550  050225          .WORD   DH0008
8120  047552  002     000     .BYTE   2,0
8121  047554  051165          .WORD   DH062A
8122  047556  000     000     .BYTE   0,0
8123  047560  051264          .WORD   DH062B
8124  047562  010     000     .BYTE   8..0
8125  047564  000005  DF065:  .WORD   5               ;ERRORS-65-66
8126  047566  000     000     .BYTE   0,0
8127  047570  050207          .WORD   DH000A
8128  047572  000     000     .BYTE   0,0
8129  047574  050225          .WORD   DH0008
8130  047576  002     000     .BYTE   2,0
8131  047600  050271          .WORD   DH001A
8132  047602  000     000     .BYTE   0,0
8133  047604  050310          .WORD   DH001B
8134  047606  002     000     .BYTE   2,0
8135  047610  000005  DF067:  .WORD   5               .ERRORS 67-70
8136  047612  000     000     .BYTE   0,0
8137  047614  050207          .WORD   DH000A
8138  047616  000     000     .BYTE   0,0
8139  047620  050225          .WORD   DH0008
8140  047622  002     000     .BYTE   2,0
8141  047624  051361          .WORD   DH067A
8142  047626  000     000     .BYTE   0,0
8143  047630  051420          .WORD   DH067B
8144  047632  004     000     .BYTE   4,0
8145  047634  000003  DF100:  .WORD   3               .ERROR 100
8146  047636  000     000     .BYTE   0,0
8147  047640  050207          .WORD   DH000A
8148  047642  000     000     .BYTE   0,0
8149  047644  050225          .WORD   DH0008
8150  047646  002     000     .BYTE   2,0
8151  047650  000005  DF126:  .WORD   5               ;ERROR 126
8152  047652  000     000     .BYTE   0,0
8153  047654  050207          .WORD   DH000A
8154  047656  000     000     .BYTE   0,0
8155  047660  050225          .WORD   DH0008
8156  047662  002     000     .BYTE   2,0
```

```
8157  047664  051456                        .WORD   DH126A
8158  047666     000      000               .BYTE   0,0
8159  047670  051475                        .WORD   DH126B
8160  047672     002      000               .BYTE   2,0
8161  047674  000007              DF224:    .WORD   7                       ;ERRORS 224-227
8162  047676     000      000               .BYTE   0,0
8163  047700  050207                        .WORD   DH060A
8164  047702     000      000               .BYTE   0,0
8165  047704  050225                        .WORD   DH000B
8166  047706     002      000               .BYTE   2,0
8167  047710  051165                        .WORD   DH062A
8168  047712     000      000               .BYTE   0,0
8169  047714  051264                        .WORD   DH062B
8170  047716     010      000               .BYTE   8.,0
8171  047720  051513                        .WORD   DH224A
8172  047722     000      000               .BYTE   0,0
8173  047724  051546                        .WORD   DH224B
8174  047726     004      000               .BYTE   4,0
8175  047730  000007              DF230:    .WORD   7                   ;ERRORS 230-233
8176  047732     000      000               .BYTE   0,0
8177  047734  050207                        .WORD   DH000A
8178  047736     000      000               .BYTE   0,0
8179  047740  050225                        .WORD   DH000B
8180  047742     002      000               .BYTE   2,0
8181  047744  051603                        .WORD   DH230A
8182  047746     000      000               .BYTE   0,0
8183  047750  051630                        .WORD   DH230B
8184  047752     003      000               .BYTE   3,0
8185  047754  051165                        .WORD   DH062A
8186  047756     000      000               .BYTE   0,0
8187  047760  051264                        .WORD   DH062B
8188  047762     010      000               .BYTE   8.,0
8189  047764  000005              DF256:    .WORD   5                       ;ERROR 256
8190  047766     000      000               .BYTE   0,0
8191  047770  050207                        .WORD   DH000A
8192  047772     000      000               .BYTE   0,0
8193  047774  050225                        .WORD   DH000B
8194  047776     002      000               .BYTE   2,0
8195  050000  051654                         WORD   DH256A
8196  050002     000      000               .BYTE   0,0
8197  050004  051720                        .WORD   DH256B
8198  050006     005      000               .BYTE   5,0
```

```
 8199                                                .SBTTL  ASCII MESSAGES
 8200
 8201  050010  005015  045522  030466  OPR001: .ASCIZ  <15><12>/RK611 BUS ADDRESS (
 8202  050016  020061  052502  020123
 8203  050024  042101  051104  051505
 8204  050032  020123  020050     000
 8205  050037     040  020051  020075  OPR002: .ASCIZ  / ) = /
 8206  050044     000
 8207  050045     122  033113  030461  OPR003: .ASCIZ  /RK611 VECTOR ADDRESS ( /
 8208  050052  053040  041505  047524
 8209  050060  020122  042101  051104
 8210  050066  051505  020123  020050
 8211  050074     000
 8212  050075     122  033113  030461  OPR004: .ASCIZ  /RK611 PRIORITY ( /
 8213  050102  050040  044522  051117
 8214  050110  052111  020131  020050
 8215  050116     000
 8216  050117     040  000040          SPACE2: .ASCIZ  /  /
 8217  050122  005015  051120  043517  ABORT:  .ASCIZ  <15><12>/PROGRAM ABORTED BECAUSE ERROR THRESHOLD EXCEEDED/<15><12>
 8218  050130  040522  020115  041101
 8219  050136  051117  042524  020104
 8220  050144  042502  040503  051525
 8221  050152  020105  051105  047522
 8222  050160  020122  044124  042522
 8223  050166  044123  046117  020104
 8224  050174  054105  042503  042105
 8225  050202  042105  005015     000
```

```
                                                .SBTTL  DATA HEADERS
                                                
 8228  050207     124  051505  020124  DH000A: .ASCIZ  /TEST    ERROR/
 8229  050214  020040  042440  051122
 8230  050222  051117     000
 8231  050225     116  046525  020040  DH000B: .ASCIZ  /NUM     PC/
 8232  050232  020040  050040  000103
 8233  050240  042524  052123  020040  DH000C: .ASCII  /TEST    TRAP/<15><12>
 8234  050246  020040  051124  050101
 8235  050254  005015
 8236  050256  052516  020115  020040          .ASCIZ  /NUM     PC/
 8237  050264  020040  041520     000
 8238  050271     105  050130  041505  DH001A: .ASCIZ  /EXPECT  ACTUAL/
 8239  050276  020124  040440  052103
 8240  050304  040525  000114
 8241  050310  045522  051503  020061  DH001B: .ASCIZ  /RKCS1   RKCS1/
 8242  050316  020040  045522  051503
 8243  050324  000061
 8244  050326  054105  042520  052103  DH001C: .ASCIZ  /EXPECT  ACTUAL  EXPECT  ACTUAL/
 8245  050334  020040  041501  052524
 8246  050342  046101  020040  054105
 8247  050350  042520  052103  020040
 8248  050356  041501  052524  046101
 8249  050364     000
 8250  050365     115  051505  020123  DH001D: .ASCIZ  /MESS A  MESS A  MESS B  MESS B/
 8251  050372  020101  046440  051505
 8252  050400  020123  020101  046440
 8253  050406  051505  020123  020102
 8254  050414  046440  051505  020123
 8255  050422  000102
 8256  050424  054105  042520  052103  DH002A: .ASCIZ  /EXPECT  ACTUAL  DRIVE/
 8257  050432  020040  041501  052524
 8258  050440  046101  020040  051104
 8259  050446  053111  000105
 8260  050452  045522  051503  020061  DH002B: .ASCIZ  /RKCS1   RKCS1   SELECT/
 8261  050460  020040  045522  051503
 8262  050466  020061  020040  042523
 8263  050474  042514  052103     000
 8264  050501     105  050130  041505  DH006A: .ASCIZ  /EXPECT  ACTUAL  HEAD/
 8265  050506  020124  040440  052103
 8266  050514  040525  020114  044040
 8267  050522  040505  000104
 8268  050526  045522  051503  020061  DH006B: .ASCIZ  /RKCS1   RKCS1   ADD/
 8269  050534  020040  045522  051503
 8270  050542  020061  020040  042101
 8271  050550  000104
 8272  050552  054105  042520  052103  DH012A: .ASCIZ  /EXPECT  ACTUAL  EXPECT  ACTUAL  MESS/
 8273  050560  020040  041501  052524
 8274  050566  046101  020040  054105
 8275  050574  042520  052103  020040
 8276  050602  041501  052524  046101
 8277  050610  020040  042515  051523
 8278  050616     000
 8279  050617     122  041513  030523  DH012B: .ASCIZ  /RKCS1   RKCS1   RKMR1   RKMR1   SELECT/
 8280  050624  020040  051040  041513
 8281  050632  030523  020040  051040
```

```
8282  050640  046513  030522  020040
8283  050646  051040  046513  030522
8284  050654  020040  042523  042514
8285  050662  052103     000
8286  050665     105  050130  041505  DH017A: .ASCIZ  /EXPECT  ACTUAL  CYLIN/
8287  050672  020124  040440  052103
8288  050700  040525  020114  041440
8289  050706  046131  047111     000
8290  050713     122  041513  030523  DH017B: .ASCIZ  /RKCS1   RKCS1   ADD/
8291  050720  020040  051040  041513
8292  050726  030523  020040  040440
8293  050734  042104     000
8294  050737     105  050130  041505  DH031A: .ASCIZ  /EXPECT  ACTUAL  OFFSET/
8295  050744  020124  040440  052103
8296  050752  040525  020114  042440
8297  050760  043106  042523  000124
8298  050766  045522  051503  020061  DH031B: .ASCIZ  /RKCS1   RKCS1   VALUE/
8299  050774  020040  045522  051503
8300  051002  020061  020040  040526
8301  051010  052514  000105
8302  051014  054105  042520  052103  DH035A: .ASCIZ  /EXPECT  ACTUAL  CYLIN   OFFSET/
8303  051022  020040  041501  052524
8304  051030  046101  020040  054503
8305  051036  044514  020116  020040
8306  051044  043117  051506  052105
8307  051052     000
8308  051053     122  041513  030523  DH035B: .ASCIZ  /RKCS1   RKCS1   ADD     VALUE/
8309  051060  020040  051040  041513
8310  051066  030523  020040  040440
8311  051074  042104  020040  020040
8312  051102  053040  046101  042525
8313  051110     000
8314  051111     102  043105  051117  DH050A: .ASCIZ  /BEFORE SHIFTING SHIFT/
8315  051116  020105  044123  043111
8316  051124  044524  043516  051440
8317  051132  044510  052106     000
8318  051137     115  051505  020123  DH050B: .ASCIZ  /MESS A  MESS B  COUNT/
8319  051144  020101  046440  051505
8320  051152  020123  020102  041440
8321  051160  052517  052116     000
8322  051165     105  050130  041505  DH062A: .ASCIZ  /EXPECT  ACTUAL  EXPECT  ACTUAL  EXPECT  ACTUAL  EXPECT  ACTUAL/
8323  051172  020124  040440  052103
8324  051200  040525  020114  042440
8325  051206  050130  041505  020124
8326  051214  040440  052103  040525
8327  051222  020114  042440  050130
8328  051230  041505  020124  040440
8329  051236  052103  040525  020114
8330  051244  042440  050130  041505
8331  051252  020124  040440  052103
8332  051260  040525  000114
8333  051264  045522  051503  020061  DH062B: .ASCIZ  /RKCS1   RKCS1   RKCS2   RKCS2   RKDS    RKDS    RKER    RkER
8334  051272  020040  045522  051503
8335  051300  020061  020040  045522
8336  051306  051503  020062  020040
8337  051314  045522  051503  020062
```

```
8338  051322  020040  045522  051504
8339  051330  020040  020040  045522
8340  051336  051504  020040  020040
8341  051344  045522  051105  020040
8342  051352  020040  045522  051105
8343  051360     000
8344  051361     105  050130  041505  DH067A: .ASCIZ  /EXFECT  ACTUAL  EXPECT  ACTUAL/
8345  051366  020124  040440  052103
8346  051374  040525  020114  042440
8347  051402  050130  041505  020124
8348  051410  040440  052103  040525
8349  051416  000114
8350  051420  045522  051115  020062  DH067B: .ASCIZ  /RKMR2   RKMR2   RKMR3   RKMR3/
8351  051426  020040  045522  051115
8352  051434  020062  020040  045522
8353  051442  051115  020063  020040
8354  051450  045522  051115  000063
8355  051456  054105  042520  052103  DH126A: .ASCIZ  /EXPECT  ACTUAL/
8356  051464  020040  041501  052524
8357  051472  046101     000
8358  051475     122  041513  030523  DH126B: .ASCIZ  /RKCS1   RKCS1/
8359  051502  020040  051040  041513
8360  051510  030523     000
8361  051513     120  042522  044526  DH224A: .ASCIZ  /PREVIOUS REGISTER CONTENTS/
8362  051520  052517  020123  042522
8363  051526  044507  052123  051105
8364  051534  041440  047117  042524
8365  051542  052116  000123
8366  051546  045522  051503  020061  DH224B: .ASCIZ  /RKCS1   RKCS2   RKDS    RKER/
8367  051554  020040  045522  051503
8368  051562  020062  020040  045522
8369  051570  051504  020040  020040
8370  051576  045522  051105     000
8371  051603     104  044522  042526  DH230A: .ASCIZ  /DRIVE   CYLIN   HEAD/
8372  051610  020040  041440  046131
8373  051616  047111  020040  044040
8374  051624  040505  000104
8375  051630  054524  042520  020040  DH230B: .ASCIZ  /TYPE    ADD     ADD/
8376  051636  020040  042101  020104
8377  051644  020040  020040  042101
8378  051652  000104
8379  051654  054105  042520  052103  DH256A: .ASCIZ  /EXPECT  ACTUAL  EXPECT  ACTUAL  ILL/
8380  051662  020040  041501  052524
8381  051670  046101  020040  054105
8382  051676  042520  052103  020040
8383  051704  041501  052524  046101
8384  051712  020040  046111  000114
8385  051720  045522  051503  020061  DH256B: .ASCIZ  /RKCS1   RKCS1   RKER    RKER    FUNCT/
8386  051726  020040  045522  051503
8387  051734  020061  020040  045522
8388  051742  051105  020040  020040
8389  051750  045522  051105  020040
8390  051756  020040  052506  041516
8391  051764  000124
```

```
                                        .SBTTL  ERROR MESSAGES
                                        
 052766  047125  054105  042520  EM000:  .ASCIZ  /UNEXPECTED MEMORY PARITY ENABLE TRAP/
 051774  052103  042105  046440
 052002  046505  051117  020131
 052010  040523  044522  054524
 052016  042440  040516  046102
 052024  020105  051124  050101
 052032     000
 052033     101  052124  046505  EM100:  .ASCIZ  /ATTEMPTING A SELECT IN 24 SECTOR FORMAT IN MAINT MODE/
 052040  052120  047111  020107
 052046  020101  042523  042514
 052054  052103  044440  020116
 052062  032062  051440  041505
 052070  047524  020122  047506
 052076  046522  052101  044440
 052104  020116  040515  047111
 052112  020124  047515  042504
 052120     000
 052121     101  052124  046505  EM101:  .ASCIZ  /ATTEMPTING A DRIVE CLEAR IN MAINT MODE/
 052126  052120  047111  020107
 052134  020101  051104  053111
 052142  020105  046103  040505
 052150  020122  047111  046440
 052156  044501  052116  046440
 052164  042117  000105
 052170  052101  042524  050115  EM102:  .ASCIZ  /ATTEMPTING A UNLOAD IN MAINT MODE/
 052176  044524  043516  040440
 052204  052440  046116  040517
 052212  020104  047111  046440
 052220  044501  052116  046440
 052226  042117  000105
 052232  052101  042524  050115  EM103:  .ASCIZ  /ATTEMPTING A PACK ACKNOWLEDGE IN MAINT MODE/
 052240  044524  043516  040440
 052246  050040  041501  020113
 052254  041501  047113  053517
 052262  042514  043504  020105
 052270  047111  046440  044501
 052276  052116  046440  042117
 052304  000105
 052306  052101  042524  050115  EM104:  .ASCIZ  /ATTEMPTING A RECALIBRATE IN MAINT MODE/
 052314  044524  043516  040440
 052322  051040  041505  046101
 052330  041111  040522  042524
 052336  044440  020116  040515
 052344  047111  020124  047515
 052352  042504     000
 052355     101  052124  046505  EM105:  .ASCIZ  /ATTEMPTING A START SPINDLE/
 052362  052120  047111  020107
 052370  020101  052123  051101
 052376  020124  050123  047111
 052404  046104  000105
 052410  052101  042524  050115  EM106:  .ASCIZ  /ATTEMPTING A SELECT USING ALL DRIVE SELECTION CONFIGS IN MAINT MODE/
 052416  044524  043516  040440
 052424  051440  046105  041505
 052432  020124  051525  047111
```

```
8448  052440  020107  046101  020114
8449  052446  051104  053111  020105
8450  052454  042523  042514  052103
8451  052462  047511  020116  047503
8452  052470  043116  043511  020123
8453  052476  047111  046440  044501
8454  052504  052116  046440  042117
8455  052512  000105
8456  052514  052101  042524  050115  EM107:  .ASCIZ  /ATTEMPTING A SELECT USING ALL HEAD ADD CONFIGS IN MAINT MODE/
8457  052522  044524  043516  040440
8458  052530  051440  046105  041505
8459  052536  020124  051525  047111
8460  052544  020107  046101  020114
8461  052552  042510  042101  040440
8462  052560  042104  041440  047117
8463  052566  044506  051507  044440
8464  052574  020116  040515  047111
8465  052602  020124  047515  042504
8466  052610      000
8467  052611      101  052124  046505  EM108:  .ASCIZ  /ATTEMPTING A SELECT USING ALL MESS SELECT CONFIGS IN MAINT MODE/
8468  052616  052120  047111  020107
8469  052624  020101  042523  042514
8470  052632  020103  052440  044523
8471  052640  043516  040440  046114
8472  052646  046440  051505  020123
8473  052654  042523  042514  052103
8474  052662  041440  047117  044506
8475  052670  051507  044440  020116
8476  052676  040515  047111  020124
8477  052704  047515  042504      000
8478  052711      101  052124  046505  EM109:  .ASCIZ  /ATTEMPTING A SEEK TO AN RK06 IN MAINT MODE/
8479  052716  052120  047111  020107
8480  052724  020101  042523  045505
8481  052732  052040  020117  047101
8482  052740  051040  030113  020066
8483  052746  047111  046440  044501
8484  052754  052116  046440  042117
8485  052762  000105
8486  052764  052101  042524  050115  EM110:  .ASCIZ  /ATTEMPTING A SEEK WITH CDT SET IN MAINT MODE/
8487  053772  044524  043516  040440
8488  053000  051440  042505  020113
8489  053006  044527  044124  041440
8490  053014  052104  051440  052105
8491  053022  044440  020116  040515
8492  053030  047111  020124  047515
8493  053036  042504      000
8494  053041      101  052124  046505  EM111:  .ASCIZ  /ATTEMPTING AN OFFSET IN MAINT MODE/
8495  053046  052120  047111  020107
8496  053054  047101  047440  043106
8497  053062  042523  020124  047111
8498  053070  046440  044501  052116
8499  053076  046440  042117  000105
8500  053104  052101  042524  050115  EM112:  .ASCII  /ATTEMPTING COMMAND WITH NON-ZERO CYLINDER ADDRESS AND/<15><12>
8501  053112  044524  043516  041440
8502  053120  046517  040515  042116
8503  053126  053440  052111  020110
```

```
8504  053134  047516  026516  042532
8505  053142  047522  041440  046131
8506  053150  047111  042504  020122
8507  053156  042101  051104  051505
8508  053164  020123  047101  006504
8509  053172     012
8510  053173     116  047117  055055          .ASCIZ  /NON-ZERO OFFSET IN MAINTENANCE MODE/
8511  053200  051105  020117  043117
8512  053206  051506  052105  044440
8513  053214  020116  040515  047111
8514  053222  042524  040516  041516
8515  053230  020105  047515  042504
8516  053236     000
8517  053237     101  052124  046505  EM113:  .ASCII  /ATTEMPTING COMMAND WITH NON-ZERO MESSAGE SELECT CODE/<15><12>
8518  053244  052120  047111  020107
8519  053252  047503  046515  047101
8520  053260  020104  044527  044124
8521  053266  047040  047117  055055
8522  053274  051105  020117  042515
8523  053302  051523  043501  020105
8524  053310  042523  042514  052103
8525  053316  041440  042117  006505
8526  053324     012
8527  053325     111  020116  040515          .ASCIZ  /IN MAINTENANCE MODE/
8528  053332  047111  042524  040516
8529  053340  041516  020105  047515
8530  053346  042504     000
8531  053351     101  052124  046505  EM114:  .ASCIZ  /ATTEMPTING TO SHIFT DRIVE MESSAGES/
8532  053356  052120  047111  020107
8533  053364  047524  051440  044510
8534  053372  052106  042040  044522
8535  053400  042526  046440  051505
8536  053406  040523  042507  000123
8537  053414  052101  042524  050115  EM115:  .ASCIZ  /ATTEMPTING TO GENERATE ODD PARITY ON SELECT DRIVE MESSAGE/
8538  053422  044524  043516  052040
8539  053430  020117  042507  042516
8540  053436  040522  042524  047440
8541  053444  042104  050040  051101
8542  053452  052111  020131  047117
8543  053460  051440  046105  041505
8544  053466  020124  051104  053111
8545  053474  020105  042515  051523
8546  053502  043501  000105
8547  053506  052101  042524  050115  EM116:  .ASCIZ  /ATTEMPTING TO GENERATE EVEN PARITY ON SELECT DRIVE MESSAGE/
8548  053514  044524  043516  052040
8549  053522  020117  042507  042516
8550  053530  040522  042524  042440
8551  053536  042526  020116  040520
8552  053544  044522  054524  047440
8553  053552  020116  042523  042514
8554  053560  052103  042040  044522
8555  053566  042526  046440  051505
8556  053574  040523  042507     000
8557  053601     101  052124  046505  EM117:  .ASCII  /ATTEMPTING COMPLETE EXECUTION OF DESELECT DRIVE COMMAND/
8558  053606  052120  047111  020107
8559  053614  047503  050115  042514
```

```
8560  053622  042524  042440  042530
8561  053630  052503  044524  047117
8562  053636  047440  020106  042504
8563  053644  042523  042514  052103
8564  053652  042040  044522  042526
8565  053660  041440  046517  040515
8566  053666  042116
8567  053670  005015  047111  046440          .ASCIZ  <15><12>/IN MAINTENANCE MODE/
8568  053676  044501  052116  047105
8569  053704  047101  042503  046440
8570  053712  042117  000105
8571  053716  052101  042524  050115  EM118:  .ASCII  /ATTEMPTING COMPLETE EXECUTION OF SELECT DRIVE COMMAND/
8572  053724  044524  043516  041440
8573  053732  046517  046120  052105
8574  053740  020105  054105  041505
8575  053746  052125  047511  020116
8576  053754  043117  051440  046105
8577  053762  041505  020124  051104
8578  053770  053111  020105  047503
8579  053776  046515  047101     104
8580  054003     015  044412  020116          .ASCIZ  <15><12>/IN MAINTENANCE MODE/
8581  054010  040515  047111  042524
8582  054016  040516  041516  020105
8583  054024  047515  042504     000
8584  054031     101  052124  046505  EM119:  .ASCIZ  /ATTEMPTING EXECUTION OF DESELECT DRIVE AT NORMAL SPEED/
8585  054036  052120  047111  020107
8586  054044  054105  041505  052125
8587  054052  047511  020116  043117
8588  054060  042040  051505  046105
8589  054066  041505  020124  051104
8590  054074  053111  020105  052101
8591  054102  047040  051117  040515
8592  054110  020114  050123  042505
8593  054116  000104
8594  054120  052101  042524  050115  EM120:  .ASCIZ   ATTEMPTING TO WRITE COMMAND AND STATUS REG. 1 IN MAINT MODE/
8595  054126  044524  043516  052040
8596  054134  020117  051127  052111
8597  054142  020105  047503  046515
8598  054150  047101  020104  047101
8599  054156  020104  052123  052101
8600  054164  051525  051040  043505
8601  054172  020056  020061  047111
8602  054200  046440  044501  052116
8603  054206  046440  042117  000105
8604  054214  052101  042524  050115  EM121:  .ASCIZ   ATTEMPTING EXECUTION OF DESELECT DRIVE WITH INTERRUPT ENABLE SET/
8605  054222  044524  043516  042440
8606  054230  042530  052503  044524
8607  054236  047117  047440  020106
8608  054244  042504  042523  042514
8609  054252  052103  042040  044522
8610  054260  042526  053440  052111
8611  054266  020110  047111  042524
8612  054274  051122  050125  020124
8613  054302  047105  041101  042514
8614  054310  051440  052105     000
8615  054315     101  052124  046505  EM122:  .ASCII   ATTEMPTING DESELECT COMMAND AFTER WRITING SILO
```

```
8616  054322  052120  047111  020107
8617  054330  042504  042523  042514
8618  054336  052103  041440  046517
8619  054344  040515  042116  040440
8620  054352  052106  051105  053440
8621  054360  044522  044524  043516
8622  054366  051440  046111  020117
8623  054374  047524  041440  042510          .ASCIZ  /TO CHECK GO CLEAR/
8624  054402  045503  043440  020117
8625  054410  046103  040505  000122
8626  054416  052101  042524  050115  EM123:  .ASCIZ  /ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE
8627  054424  044524  043516  041440
8628  054432  046517  046120  052105
8629  054440  020105  054105  041505
8630  054446  052125  047511  020116
8631  054454  043117  051440  042505
8632  054462  020113  047111  046440
8633  054470  044501  052116  046440
8634  054476  042117  000105
8635  054502  052101  042524  050115  EM124:  .ASCIZ  /ATTEMPTING SELECT DRIVE IN MAINT MODE
8636  054510  044524  043516  051440
8637  054516  046105  041505  020124
8638  054524  051104  053111  020105
8639  054532  047111  046440  044501
8640  054540  052116  046440  042117
8641  054546  000105
8642  054550  052101  042524  050115  EM125:  .ASCIZ  /ATTEMPTING CHECK "LOAD STATUS" BY FORCING/<15><12>
8643  054556  044524  043516  041440
8644  054564  042510  045503  021040
8645  054572  047514  042101  051440
8646  054600  040524  052524  021123
8647  054606  041040  020131  047506
8648  054614  041522  047111  006507
8649  054622     012
8650  054623     104  044522  042526          .ASCII  /DRIVE AVAILIABLE, SPEED LOSS, VOLUME VALID,/<15><12>
8651  054630  040440  040526  046111
8652  054636  040511  046102  026105
8653  054644  051440  042520  042105
8654  054652  046040  051517  026123
8655  054660  053040  046117  046525
8656  054666  020105  040526  044514
8657  054674  026104  005015
8658  054700  043117  051506  052105           ASCII  /OFFSET, DRIVE READY, AND WRITE LOCK/<15><12>
8659  054706  020054  051104  053111
8660  054714  020105  042522  042101
8661  054722  026131  040440  042116
8662  054730  053440  044522  042524
8663  054736  046040  041517  006513
8664  054744     012
8665  054745     104  044522  042526          .ASCIZ  /DRIVE STATUS REG./
8666  054752  051440  040524  052524
8667  054760  020123  042522  027107
8668  054766     000
8669  054767     101  052124  046505  EM126:  .ASCIZ  /ATTEMPTING TO FORCE DRIVE AVAILIABLE/
8670  054774  052120  047111  020107
8671  055002  047524  043040  051117
```

```
8672  055010  042503  042040  044522
8673  055016  042526  040440  040526
8674  055024  046111  040511  046102
8675  055032  000105
8676  055034  052101  042524  050115  EM127:  .ASCII  /ATTEMPTING TO FORCE DRIVE BUS PARITY ERROR/<15><12>
8677  055042  044524  043516  052040
8678  055050  020117  047506  041522
8679  055056  020105  051104  053111
8680  055064  020105  052502  020123
8681  055072  040520  044522  054524
8682  055100  042440  051122  051117
8683  055106  005015
8684  055110  042504  042524  052103          .ASCIZ  /DETECTED BY RK611/
8685  055116  042105  041040  020131
8686  055124  045522  030466  000061
8687  055132  052101  042524  050115  EM128:   ASCIZ  /ATTEMPTING TO FORCE DRIVE AVAILIABLE RESET ERROR
8688  055140  044524  043516  052040
8689  055146  020117  047506  041522
8690  055154  020105  051104  053111
8691  055162  020105  053101  044501
8692  055170  044514  041101  042514
8693  055176  051040  051505  052105
8694  055204  042440  051122  051117
8695  055212     000
8696  055213     124  051505  044524  EM129:   ASCIZ  /TESTING CDT SET DRIVE TYPE DETECTION/
8697  055220  043516  041440  052104
8698  055226  051440  052105  042040
8699  055234  044522  042526  052040
8700  055242  050131  020105  042504
8701  055250  042524  052103  047511
8702  055256  000116
8703  055260  052101  042524  050115  EM130:  .ASCIZ  /ATTEMPTING TO FORCE DRIVE TYPE ERROR WITH CDT SET/
8704  055266  044524  043516  052040
8705  055274  020117  047506  041522
8706  055302  020105  051104  053111
8707  055310  020105  054524  042520
8708  055316  042440  051122  051117
8709  055324  053440  052101  020110
8710  055332  042103  020124  042523
8711  055340  000124
8712  055342  052101  042524  050115  EM131:  .ASCIZ  /ATTEMPTING TO FORCE DRIVE TYPE ERROR ADDRESSING RK06
8713  055350  044524  043516  052040
8714  055356  020117  047506  041522
8715  055364  020105  051104  053111
8716  055372  020105  054524  042520
8717  055400  042440  051122  051117
8718  055406  040440  042104  042522
8719  055414  051523  047111  020107
8720  055422  045522  033060     000
8721  055427     101  052124  046505  EM132:  .ASCIZ  /ATTEMPTING TO FORCE SPEED LOSS/
8722  055434  052120  047111  020107
8723  055442  047524  043040  051117
8724  055450  042503  051440  042520
8725  055456  042105  046040  051517
8726  055464  000123
8727  055466  052101  042524  050115  EM133:  .ASCIZ  /ATTEMPTING TO FORCE DRIVE OFF TRACK/
```

```
8728  055474  044524  043516  052040
8729  055502  020117  047506  041522
8730  055510  020105  051104  053111
8731  055516  020105  043117  020106
8732  055524  051124  041501  000113
8733  055532  052101  042524  050115  EM134:  .ASCIZ  /ATTEMPTING TO FORCE WRITE LOCK ERROR/
8734  055540  044524  043516  052040
8735  055546  020117  047506  041522
8736  055554  020105  051127  052111
8737  055562  020105  047514  045503
8738  055570  042440  051122  051117
8739  055576     000
8740  055577     101  052124  046505  EM135:  .ASCIZ  /ATTEMPTING TO FORCE SEEK INCOMPLETE/
8741  055604  052120  047111  020107
8742  055612  047524  043040  051117
8743  055620  042503  051440  042505
8744  055626  020113  047111  047503
8745  055634  050115  042514  042524
8746  055642     000
8747  055643     101  052124  046505  EM136:  .ASCIZ  /ATTEMPTING TO FORCE NON-EXECUTABLE FUNCTION/
8748  055650  052120  047111  020107
8749  055656  047524  043040  051117
8750  055664  042503  047040  047117
8751  055672  042455  042530  052503
8752  055700  040524  046102  020105
8753  055706  052506  041516  044524
8754  055714  047117     000
8755  055717     101  052124  046505  EM137:  .ASCIZ  /ATTEMPTING TO FORCE AC LOW AND C-D PARITY ERROR/
8756  055724  052120  047111  020107
8757  055732  047524  043040  051117
8758  055740  042503  040440  020103
8759  055746  047514  020127  047101
8760  055754  020104  026503  020104
8761  055762  040520  044522  054524
8762  055770  042440  051122  051117
8763  055776     000
8764  055777     101  052124  046505  EM138:  .ASCII  /ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR
8765  056004  052120  047111  020107
8766  056012  047524  043040  051117
8767  056020  042503  044440  046114
8768  056026  043505  046101  042040
8769  056034  051511  020113  042101
8770  056042  051104  051505  020123
8771  056050  051105  047522     122
8772  056055     015  043012  047522          ASCIZ  <15><12>/FROM DRIVE MESSAGE BITS/
8773  056062  020115  051104  053111
8774  056070  020105  042515  051523
8775  056076  043501  020105  044502
8776  056104  051524     000
8777  056107     101  052124  046505  EM139:  .ASCIZ  /ATTEMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR/
8778  056114  052120  047111  020107
8779  056122  047524  041440  042514
8780  056130  051101  051040  033113
8781  056136  030461  053440  052111
8782  056144  020110  020101  047503
8783  056152  052116  047522  046114
```

```
8784  056160  051105  041440  042514
8785  056166  051101     000
8786  056171     124  051505  044524  EM140:  .ASCIZ  /TESTINC ILLEGAL DISK ADDRESS ERROR LOGIC IN RK611/
8787  056176  043516  044440  046114
8788  056204  043505  046101  042040
8789  056212  051511  020113  042101
8790  056220  051104  051505  020123
8791  056226  051105  047522  020122
8792  056234  047514  044507  020103
8793  056242  047111  051040  033113
8794  056250  030461     000
8795  056253     101  052124  046505  EM141:  .ASCIZ  /ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES/
8796  056260  052120  047111  020107
8797  056266  047524  051040  041505
8798  056274  044505  042526  047040
8799  056302  047117  051455  040524
8800  056310  042116  051101  020104
8801  056316  042515  051523  043501
8802  056324  051505     000
8803  056327     101  052124  046505  EM142:  .ASCII  /ATTEMPTING TO RECEIVE NON-STANDARD MESSAGES/
8804  056334  052120  047111  020107
8805  056342  047524  051040  041505
8806  056350  044505  042526  047040
8807  056356  047117  051455  040524
8808  056364  042116  051101  020104
8809  056372  042515  051523  043501
8810  056400  051505
8811  056402  053440  052111  020110          .ASCIZ  / WITH PARITY ERROR/
8812  056410  040520  044522  054524
8813  056416  042440  051122  051117
8814  056424     000
8815  056425     101  052124  046505  EM143:  .ASCIZ  /ATTEMPTING TO FORCE NON-EXISTENT DRIVE (DRIVE BUS TIMEOUT)/
8816  056432  052120  047111  020107
8817  056440  047524  043040  051117
8818  056446  042503  047040  047117
8819  056454  042455  044530  052123
8820  056462  047105  020124  051104
8821  056470  053111  020105  042050
8822  056476  044522  042526  041040
8823  056504  051525  052040  046511
8824  056512  047505  052125  000051
8825  056520  052101  042524  050115  EM144:  .ASCIZ  /ATTEMPTING TO FORCE NON-EXISTENT DRIVE (NO SACK)/
8826  056526  044524  043516  052040
8827  056534  020117  047506  041522
8828  056542  020105  047516  026516
8829  056550  054105  051511  042524
8830  056556  052116  042040  044522
8831  056564  042526  024040  047516
8832  056572  051440  041501  024513
8833  056600     000
8834  056601     101  052124  046505  EM145:  .ASCIZ  /ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE RESET/
8835  056606  052120  047111  020107
8836  056614  054105  041505  052125
8837  056622  047511  020116  043117
8838  056630  042040  051505  046105
8839  056636  041505  020124  051104
```

```
8840  056644  053111  020105  044527
8841  056652  044124  044440  020105
8842  056660  042522  042523  000124
8843  056666  052101  042524  050115  EM146:  .ASCIZ  /ATTEMPTING TO EXECUTE AN ILLEGAL FUNCTION/
8844  056674  044524  043516  052040
8845  056702  020117  054105  041505
8846  056710  052125  020105  047101
8847  056716  044440  046114  043505
8848  056724  046101  043040  047125
8849  056732  052103  047511  000116
8850  056740  052101  042524  050115  EM147:  .ASCIZ  /ATTEMPTING TO CLEAR ILLEGAL FUNCTION/
8851  056746  044524  043516  052040
8852  056754  020117  046103  040505
8853  056762  020122  046111  042514
8854  056770  040507  020114  052506
8855  056776  041516  044524  047117
8856  057004     000
8857  057005     104  044522  042526  EM2000: .ASCIZ   DRIVE COMMAND BIT DID NOT SET IN DRIVE MESS A/
8858  057012  041440  046517  040515
8859  057020  042116  041040  052111
8860  057026  042040  042111  047040
8861  057034  052117  051440  052105
8862  057042  044440  020116  051104
8863  057050  053111  020105  042515
8864  057056  051523  040440     000
8865  057063     104  044522  042526  EM2001: .ASCIZ  /DRIVE MESS A INCORRECT/
8866  057070  046440  051505  020123
8867  057076  020101  047111  047503
8868  057104  051122  041505  000124
8869  057112  051104  053111  020105  EM2002: .ASCIZ  /DRIVE MESS B INCORRECT/
8870  057120  042515  051523  041040
8871  057126  044440  041516  051117
8872  057134  042522  052103     000
8873  057141     103  046517  040515  EM2003: .ASCIZ  /COMMAND AND STAUS REG. 1 INCORRECT/
8874  057146  042116  040440  042116
8875  057154  051440  040524  051525
8876  057162  051040  043505  020056
8877  057170  020061  047111  047503
8878  057176  051122  041505  000124
8879  057204  051104  053111  020105  EM2004: .ASCIZ  /DRIVE SELECT CODE IN MESSAGE A INCORRECT/
8880  057212  042523  042514  052103
8881  057220  041440  042117  020105
8882  057226  047111  046440  051505
8883  057234  040523  042507  040440
8884  057242  044440  041516  051117
8885  057250  042522  052103     000
8886  057255     110  040505  020104  EM2005: .ASCIZ  /HEAD ADD CODE IN MESSAGE A INCORRECT/
8887  057262  042101  020104  047503
8888  057270  042504  044440  020116
8889  057276  042515  051523  043501
8890  057304  020105  020101  047111
8891  057312  047503  051122  041505
8892  057320  000124
8893  057322  040515  047111  020124  EM2006: .ASCIZ  /MAINT REG. 1 INCORRECT/
8894  057330  042522  027107  030440
8895  057336  044440  041516  051117
```

```
8896  057344  042522  052103     000
8897  057351     115  051505  020123  EM2007: .ASCIZ  /MESS SELECT CODE IN MESSAGE B INCORRECT/
8898  057356  042523  042514  052103
8899  057364  041440  042117  020105
8900  057372  047111  046440  051505
8901  057400  040523  042507  041040
8902  057406  044440  041516  051117
8903  057414  042522  052103     000
8904  057421     103  046131  047111  EM2008: .ASCIZ  /CYLINDER ADD BITS IN MESSAGE B INCORRECT/
8905  057426  042504  020122  042101
8906  057434  020104  044502  051524
8907  057442  044440  020116  042515
8908  057450  051523  043501  020105
8909  057456  020102  047111  047503
8910  057464  051122  041505  000124
8911  057472  043117  051506  052105  EM2009: .ASCIZ  /OFFSET VALUE BITS IN MESSAGE B INCORRECT/
8912  057500  053040  046101  042525
8913  057506  041040  052111  020123
8914  057514  047111  046440  051505
8915  057522  040523  042507  041040
8916  057530  044440  041516  051117
8917  057536  042522  052103     000
8918  057543     120  051101  052111  EM2010: .ASCIZ  /PARITY BIT IN MESSAGE A INCORRECT/
8919  057550  020131  044502  020124
8920  057556  047111  046440  051505
8921  057564  040523  042507  040440
8922  057572  044440  041516  051117
8923  057600  042522  052103     000
8924  057605     120  051101  052111  EM2011: .ASCIZ  /PARITY BIT IN MESSAGE B INCORRECT/
8925  057612  020131  044502  020124
8926  057620  047111  046440  051505
8927  057626  040523  042507  041040
8928  057634  044440  041516  051117
8929  057642  042522  052103     000
8930  057647     103  046517  040515  EM2012: .ASCIZ  /COMMAND AND STATUS REG 2 INCORRECT/
8931  057654  042116  040440  042116
8932  057662  051440  040524  052524
8933  057670  020123  042522  020107
8934  057676  020062  047111  047503
8935  057704  051122  041505  000124
8936  057712  051105  047522  020122  EM2013: .ASCIZ  /ERROR REG INCORRECT/
8937  057720  042522  020107  047111
8938  057726  047503  051122  041505
8939  057734  000124
8940  057736  047503  046515  047101  EM2014: .ASCIZ  /COMMAND AND STATUS REG 1 INCORRECT AT PHASE ADDRESS 4/
8941  057744  020104  047101  020104
8942  057752  052123  052101  051525
8943  057760  051040  043505  030440
8944  057766  044440  041516  051117
8945  057774  042522  052103  040440
8946  060002  020124  044120  051501
8947  060010  020105  042101  051104
8948  060016  051505  020123  000064
8949  060024  047503  046515  047101  EM2015: .ASCIZ  /COMMAND AND STATUS REG 1 INVALID DURING COMMAND EXECUTION/
8950  060032  020104  047101  020104
8951  060040  052123  052101  051525
```

```
8952  060046  051040  043505  030440
8953  060054  044440  053116  046101
8954  060062  042111  042040  051125
8955  060070  047111  020107  047503
8956  060076  046515  047101  020104
8957  060104  054105  041505  052125
8958  060112  047511  000116
8959  060116  040515  047111  042524  EM2016: .ASCIZ  /MAINTENANCE REG 2 UNEXPECTEDLY CHANGED DURING COMMAND EXECUTION/
8960  060124  040516  041516  020105
8961  060132  042522  020107  020062
8962  060140  047125  054105  042520
8963  060146  052103  042105  054514
8964  060154  041440  040510  043516
8965  060162  042105  042040  051125
8966  060170  047111  020107  047503
8967  060176  040515  047101  020104
8968  060204  054105  041505  052125
8969  060212  047511  000116
8970  060216  040515  047111  042524  EM2017: .ASCIZ  /MAINTENANCE REG 3 UNEXPECTEDLY CHANGED DURING COMMAND EXECUTION/
8971  060224  040516  041516  020105
8972  060232  042522  020107  020063
8973  060240  047125  054105  042520
8974  060246  052103  042105  054514
8975  060254  041440  040510  043516
8976  060262  042105  042040  051125
8977  060270  047111  020107  047503
8978  060276  046515  047101  020104
8979  060304  054105  041505  052125
8980  060312  047511  000116
8981  060316  047111  042524  051122  EM2018: .ASCIZ  /INTERRUPT DID NOT OCCUR/
8982  060324  050125  020124  044504
8983  060332  020104  047516  020124
8984  060340  041517  052503  000122
8985  060346  047503  046515  047101  EM2019: .ASCIZ  /COMMAND AND STATUS REG 1 INCORRECT AFTER INTERRUPT/
8986  060354  020104  047101  020104
8987  060362  052123  052101  051525
8988  060370  051040  043505  030440
8989  060376  044440  041516  051117
8990  060404  042522  052103  040440
8991  060412  052106  051105  044440
8992  060420  052116  051105  052522
8993  060426  052120     000
8994  060431     103  046517  040515  EM2020: .ASCIZ  /COMMAND AND STATUS REG 2 INCORRECT AFTER INTERRUPT/
8995  060436  042116  040440  042116
8996  060444  051440  040524  052524
8997  060452  020123  042522  020107
8998  060460  020062  047111  047503
8999  060466  051122  041505  020124
9000  060474  043101  042524  020122
9001  060502  047111  042524  051122
9002  060510  050125  000124
9003  060514  051105  047522  020122  EM2021: .ASCIZ  /ERROR REGISTER INCORRECT AFTER INTERRUPT/
9004  060522  042522  044507  052123
9005  060530  051105  044440  041516
9006  060536  051117  042522  052103
9007  060544  040440  052106  051105
```

```
 9008  060552  044440  052116  051105
 9009  060560  052522  052120     000
 9010  060565     111  052116  051105  EM2022: .ASCIZ  /INTERRUPT DID NOT CLEAR IN RK611/
 9011  060572  052522  052120  042040
 9012  060600  042111  047040  052117
 9013  060606  041440  042514  051101
 9014  060614  044440  020116  045522
 9015  060622  030466  000061
 9016  060626  040504  040524  046040  EM2023: .ASCIZ  /DATA LATE DID NOT OCCUR WHEN LEAVING SILO/
 9017  060634  052101  020105  044504
 9018  060642  020104  047516  020124
 9019  060650  041517  052503  020122
 9020  060656  044127  047105  046040
 9021  060664  040505  044526  043516
 9022  060672  051440  046111  000117
 9023  060700  051104  053111  020105  EM2024: .ASCIZ  /DRIVE COMMAND BITS IN MESSAGE INCORRECT/
 9024  060706  047503  046515  047101
 9025  060714  020104  044502  051524
 9026  060722  044440  020116  042515
 9027  060730  051523  043501  020105
 9028  060736  047111  047503  051122
 9029  060744  041505  000124
 9030  060750  051104  053111  020105  EM2025: .ASCIZ  /DRIVE STATUS REGISTER INCORRECT/
 9031  060756  052123  052101  051525
 9032  060764  051040  043505  051511
 9033  060772  042524  020122  047111
 9034  061000  047503  051122  041505
 9035  061006  000124
 9036  061010  047503  052116  047522  EM2026: .ASCIZ  /CONTROLLER READY DID NOT SET/
 9037  061016  046114  051105  051040
 9038  061024  040505  054504  042040
 9039  061032  042111  047040  052117
 9040  061040  051440  052105     000
 9041  061045     114  040517  020104  EM2027: .ASCIZ  /LOAD STATUS DID NOT LOAD DRIVE STATUS REG./
 9042  061052  052123  052101  051525
 9043  061060  042040  042111  047040
 9044  061066  052117  046040  040517
 9045  061074  020104  051104  053111
 9046  061102  020105  052123  052101
 9047  061110  051525  051040  043505
 9048  061116  000056
 9049  061120  047125  054105  042520  EM2028: .ASCIZ  /UNEXPECTED INTERRUPT OCCURRED/
 9050  061126  052103  042105  044440
 9051  061134  052116  051105  052522
 9052  061142  052120  047440  041503
 9053  061150  051125  042522  000104
 9054  061156  047111  042524  051122  EM2029: .ASCIZ  /INTERRUPT OCCURRED WHEN INTERRUPT ENABLE SET/
 9055  061164  050125  020124  041517
 9056  061172  052503  051122  042105
 9057  061200  053440  042510  020116
 9058  061206  047111  042524  051122
 9059  061214  050125  020124  047105
 9060  061222  041101  042514  051440
 9061  061230  052105     000
 9062          000001                  .END
```

```
ABASE = 177440            937*   1172    1213
ABORT   050122           7358    8217*
ACDW1 = 000000           1172    1215
ACDW2 = 000000           1172    1216
ACLO  = 000010           1032*   5977
ACPUOP= 000000           1172    1187
ADDW0 = 000000           1172
ADDW1 = 000000           1172
ADDW10= 000000           1172
ADDW11= 000000           1172
ADDW12= 000000           1172
ADDW13= 000000           1172
ADDW14= 000000           1172
ADDW15= 000000           1172
ADDW2 = 000000           1172
ADDW3 = 000000           1172
ADDW4 = 000000           1172
ADDW5 = 000000           1172
ADDW6 = 000000           1172
ADDW7 = 000000           1172
ADDW8 = 000000           1172
ADDW9 = 000000           1172
ADEVCT= 000000           1172    1178
ADEVM = 000000           1172    1214
AENV  = 000000           1172    1183
AENVM = 000000           1172    1184
AFATAL= 000000           1172    1175
AMADR1= 000000           1172    1200
AMADR2= 000000           1172    1204
AMADR3= 000000           1172    1207
AMADR4= 000000           1172    1210
AMAMS1= 000000           1172    1194
AMAMS2= 000000           1172    1202
AMAMS3= 000000           1172    1205
AMAMS4= 000000           1172    1208
AMSGAD= 000000           1172    1180
AMSGLG= 000000           1172    1181
AMSGTY= 000000           1172    1174
AMTYP1= 000000           1172    1195
AMTYP2= 000000           1172    1203
AMTYP3= 000000           1172    1206
AMTYP4= 000000           1172    1209
APASS = 000000           1172    1177
APRIOR= 000005            936*   1172
APTCSU= 000040           7235*   7398
APTENV= 000001           7191    7233*   7267    7391
APTSIZ= 000200           2401    7232*
APTSPO= 000100           7193    7234*   7393
ASWREG= 000000           1172    1185
ATESTN= 000000           1172    1176
AUNIT = 000000           1172    1179
AUSWR = 000000           1172    1186
AVECT1= 120210            935*   1172    1211
AVECT2= 000000           1172    1212
BA1   = 000020            995*
BA16  = 000400            980*
```

```
BA17  = 001000  981#
BIT0  = 000001  917#  977  1010  1029  4206  4262  4392  4439  4482  4531  4664
BIT00 = 000001  907#  917
BIT01 = 000002  906#  916
BIT02 = 000004  905#  915
BIT03 = 000010  904#  914
BIT04 = 000020  903#  913
BIT05 = 000040  902#  912
BIT06 = 000100  901#  911
BIT07 = 000200  900#  910
BIT08 = 000400  899#  909  7061
BIT09 = 001000  898#  908  7079  7278
BIT1  = 000002  916#  1011
BIT10 = 002000  897#  982  1001  1020  1053  1068  3322  7255
BIT11 = 004000  896#  983  1002  1021  1038  1054  1069  7086
BIT12 = 010000  895#  985  1003  1022  1055  4236  4237
BIT13 = 020000  894#  986  1004  1023  1039  1056  7262
BIT14 = 040000  893#  987  1005  1024  1040  1057  7047
BIT15 = 100000  892#  988  989  1006  1025  1041  1058  2537  2546  4301  4307  4332  4337
                4394  4395  4396  4438  4440  4483  4533  4534  4535  4666  4667
BIT2  = 000004  915#  1012  1031
BIT3  = 000010  914#  994  1013  1032  6661
BIT4  = 000020  913#  995  1014  1033  1047  1062
BIT5  = 000040  912#  996  1015  1034  1048  1063
BIT6  = 000100  911#  978  997  1016  1035  1049  1064
BIT7  = 000200  910#  979  998  1017  1036  1050  1065  3320
BIT8  = 000400  909#  980  999  1018  1037  1051  1066
BIT9  = 001000  908#  981  1000  1019  1052  1067
BPTVEC= 000014  924#
BSE   = 000200  1017#
CCLR  = 100000  989#  2498  2570  2587  2621  2639  2677  2701  2741  2764  2770  2811  2827
                2874  2890  2937  2953  2993  3011  3042  3060  3097  3118  3165  3181  3228
                3249  3299  3326  3365  3383  3418  3436  3478  3496  3538  3556  3598  3616
                3658  3676  3715  3737  3743  3782  3804  3810  3849  3871  3877  3916  3938
                3946  3983  4005  4011  4050  4072  4078  4110  4132  4138  4179  4234  4289
                4318  4369  4445  4507  4917  5070  5147  5224  5299  5376  5454  5532  5609
                5686  5764  5842  5920  5996  6074  6158  6242  6327  6411  6495  6579  6718
                6800  6879  6918  6933
CCT   = 002000  382#  3223  3231  4936  4943  4987  5339  5355  5417  5433  6204  6220  6457
                6473  6541  6557  6626  6642
CERR  = 100000  988#  4912  5203  5278  5433  5511  5743  5821  5899  5975  6053  6136  6305
                6389  6473  6557  6698  6777  6859  6921
CFMT  = 010000  985#  2623  2624  2848  2850  2911  2913  2974  2976  4243  4936  4943  4987
                5959  5975
CHKPAR  042360  2479  7001#  7879
CKSWR = 104407  7046  7251  7277  7926#
CLEAR = 000005  962#  2805  3477  3482  3781  3785  5883  5899
COE   = 001000  1019#
CR    = 000015  832#  7437  7447
CRLF  = 000200  833#  2426  7408  7447
CTO   = 004000  983#
CYLIN   004252  2336#  3091*  3099  3110  3140*  3141  3158*  3167  3203  3205*  3222*  3230  3241
                3271  3273*  3274*  3275  3362*  3415*  3475*  3535*  3595*  3655*  6114*  6198*  6283*
                6367*  6451*  6535*  7950  7956  7981
DCK   = 100000  1025#
DDISP = 177570  839#  1145  2389
```

```
DDT    = 000400  1037#  5357  5513  6223  6476  6560
DF000    047150  2277  7991#
DF001    047154  1238  1680  1686  1692  1698  1704  1710  1716  1722  7993#
DF002    047210  1243  1248  1253  1258  8007#
DF006    047244  1263  1268  1273  1278  8021#
DF012    047300  1283  1288  1293  1298  1303  1410  1416  1428  1434  1440  8035#
DF017    047334  1308  1313  1318  1323  1333  1338  1344  1349  1354  8049#
DF031    047370  1359  1364  1369  1374  8063#
DF035    047424  1380  1386  1392  1398  1404  8077#
DF050    047460  1446  1452  8091#
DF052    047514  1458  1464  1470  1476  1482  1488  1494  1500  8105#
DF062    047540  1506  1512  1518  1552  1558  1534  1570  1576  1582  1599  1605  1611  1624
                 1631  1662  1668  1674  1728  1734  1740  1746  1758  1764  1770  1776  1782
                 1788  1794  1800  1806  1812  1818  1824  1830  1836  1842  1848  1854  1860
                 1866  1872  1878  1884  1890  1896  1902  1908  1914  1920  1926  1932  1938
                 1944  1950  1956  1962  1968  1974  1980  1986  1992  1998  2004  2010  2016
                 2022  2028  2034  2040  2046  2052  2058  2064  2070  2076  2082  2088  2094
                 2148  2154  2160  2166  2172  2178  2184  2190  2196  2202  2208  2214  2220
                 2226  2232  2238  8115#
DF065    047564  1525  1532  1587  1637  1643  8125#
DF067    047610  1539  1546  1650  1656  8135#
DF100    047634  1593  1617  1752  2244  2250  8145#
DF126    047650  8151#
DF224    047674  2100  2106  2112  2118  8161#
DF230    047730  2124  2130  2136  2142  8175#
DF256    047764  2256  2262  2267  2272  3189#
DH000A   050207  7995  8009  8023  8037  8051  8065  8079  8093  8107  8117  8127  8137  8147
                 8153  8163  8177  8191  9228#
DH000B   050225  7997  8011  8025  8039  8053  C067  8081  8095  8109  8119  8129  8139  8149
                 8155  8165  8179  8193  8231#
DH000C   050240  2275  8233#
DH001A   050271  7999  8131  8238#
DH001B   050310  8001  8133  8241#
DH001C   050326  8003  8017  8031  8045  8059  8073  8087  8101  8111  8244#
DH001D   050365  8005  8019  8033  8047  8061  8075  8089  8103  8113  8250#
DH002A   050424  8013  8256#
DH002B   050452  8015  8260#
DH006A   050501  8027  8264#
DH006B   050526  8029  8268#
DH012A   050552  8041  8272#
DH012B   050617  8043  8279#
DH017A   050665  8055  8286#
DH017B   050713  8057  8290#
DH031A   050737  8069  8294#
DH031B   050766  8071  8298#
DH035A   051014  8083  8302#
DH035B   051053  8085  8308#
DH050A   051111  8097  8314#
DH050B   051137  8099  8318#
DH062A   051165  6121  8167  9185  8322#
DH062B   051264  8123  8169  8187  8333#
DH067A   051361  8141  8344#
DH067B   051420  8143  8350#
DH126A   051456  8157  8355#
DH126B   051475  8159  8358#
DH224A   051513  8171  8361#
```

```
DH2248  051546          8173    8366#
DH230A  051603          8181    8371#
DH230B  051630          8183    8375#
DH256A  051654          8195    8379#
DH256B  051720          8197    8385#
DI     = 040000          987#
DISPLA  001142          1145#   2389*   2397*   7101*   7254*
DISPRE  000174          1077#   2397
DLT    = 100000          1006#   4913
DMD    = 000040          1048#   2499    2512    2513    2571    2575    2576    2622    2626    2627    2678    2684    2685
                         2743    2755    2812    2815    2816    2875    2878    2879    2938    2941    2942    2994    2998
                         2999    3043    3047    3048    3098    3102    3103    3166    3170    3171    3223    3233    3234
                         3300    3304    3305    3366    3371    3372    3419    3424    3425    3479    3484    3485    3539
                         3544    3545    3599    3604    3605    3659    3664    3665    3716    3728    3783    3795    3850
                         3862    3917    3929    3984    3996    4051    4063    4111    4123    4185    4240    4290    4293
                         4294    4319    4323    4324    4371    4447    4509    4585    4589    4590    4640    4644    4645
                         4670    4671    4695    4696    4933    4938    4939    4953    4954    4979    4980    5022    5027
                         5028    5107    5112    5113    5184    5189    5190    5261    5264    5265    5336    5341    5342
                         5414    5419    5420    5492    5497    5498    5569    5574    5575    5646    5651    5652    5724
                         5729    5730    5802    5807    5808    5880    5885    5896    5958    5961    5962    6034    6039
                         6040    6112    6122    6123    6196    6206    6207    6281    6291    6292    6365    6375    6376
                         6449    6459    6460    6533    6543    6544    6622    6628    6629    6679    6684    6695    6757
                         6760    6761    6842    6845    6846    6923
DRA    = 000001          1029#   5048    5050    5128    5205    5357    5435    5513    5590    5667    6139    6223    6308
                         6392    6476    6560
DRDY   = 000200          1036#   5048    5050
DROT   = 000040          1034#   5667    6308    6560
DRPAR  = 000010          1013#   5978
DRVCOD  004244          2333#   2564*   2572    2582    2592    2604*   2605    4364*   4370    4372    4381    4389    4390
                         4418*   4419    4436*   4448    4478    4480*   4502*   4508    4511    4520    4528    4529    4557*
                         4558    4579*   4586    4598    4615*   4616    4634*   4641    4655    4663    4704    4721*   4722
                         4739*   4745    4757    4774*   4775    6616*   6625    6643    6660*   6661    7939
DRVMSK= 000007            993#
DRVTYP  004266          2342#   6119*   6203*   6288*   6372*   6456*   6540*   7981
DSC    = 040000          1040#
DSWR   = 177570           838#   1144    2388
DTE    = 010000          1022#
DTYE   = 000040          1015#   5436    5514
DT000   046514          2276    7935#
DT001   046520          1237    1679    1685    1691    1697    1703    1709    1715    1721    7936#
DT002   046540          1242    1247    1252    1257    7939#
DT006   046562          1262    1267    1272    1277    7942#
DT012   046604          1282    1287    1292    1297    1302    1409    1415    1421    1422    1427    1433    1439    7945#
DT017   046632          1307    1312    1317    1322    1327    1328    1332    1337    1343    1348    1353    7950#
DT031   046654          1358    1363    1368    1373    7953#
DT035   046676          1379    1385    1391    1397    1403    7956#
DT050   046722          1445    1451    7960#
DT052   046744          1457    1463    1469    1475    1481    1487    1493    1499    7963#
DT062   046760          1505    1511    1517    1551    1557    1563    1569    1575    1581    1598    1604    1610    1623
                         1630    1661    1667    1673    1727    1733    1739    1745    1757    1763    1769    1775    1781
                         1787    1793    1799    1805    1811    1817    1823    1829    1835    1841    1847    1853    1859
                         1865    1871    1877    1883    1889    1895    1901    1907    1913    1919    1925    1931    1937
                         1943    1949    1955    1961    1967    1973    1979    1985    1991    1997    2003    2009    2015
                         2021    2027    2033    2039    2045    2051    2057    2063    2069    2075    2081    2087    2093
                         2147    2153    2159    2165    2171    2177    2183    2189    2195    2201    2207    2213    2219
                         2225    2231    2237    7965#
```

```
DT065   047004          1524    1531    1586    1636    1642    7969#
DT067   047014          1538    1545    1649    1655    7971#
DT100   047030          1592    1616    1751    2243    2249    7973#
DT126   047034          7974#
DT224   047044          2099    2105    2111    2117    7976#
DT230   047100          2123    2129    2135    2141    7981#
DT256   047132          2255    2261    2266    2271    7966#
ECCW  = 020000          1056#   2756    2758    3729    3731    3796    3798    3863    3865    3930    3932    3997    3999
                        4064    4066    4124    4126
ECM   = 000100          1016#
EMTVEC= 000030           927#   2372#   2373#
EM000   051766          2274    8394#
EM1N    001300          1235#   2620#   2637#   2644#   2650#   2654#   2804#   2825#   2835#   2841#   2845#   2867#   2888#
                        2898#   2904#   2908#   2930#   2951#   2961#   2967#   2971#   2992#   3009#   3016#   3022#   3026#
                        3041#   3058#   3065#   3071#   3075#
EM100   052033          2620    8401#
EM101   052121          2804    8411#
EM102   052170          2867    8418#
EM103   052232          2930    8424#
EM104   052306          2992    3432#
EM105   052355          3041    8439#
EM106   052410          1240    1245    1250    1255    8444#
EM107   052514          1260    1265    1270    1275    845[illegible]#
EM108   052611          1280    1285    1290    1295    13[illegible]   8467#
EM109   052711          1305    1310    1315    1320    13[illegible]   8478#
EM110   052764          1330    1335    1341    1346    1351    8486#
EM111   053041          1356    1361    1366    1371    8494#
EM112   053104          1377    1383    1389    1395    1401    8500#
EM113   053237          1407    1413    1419    1425    1431    1437    8517#
EM114   053351          1443    1449    8531#
EM115   053414          1455    1461    1467    1473    8537#
EM116   053506          1479    1485    1491    1497    8547#
EM117   053601          1503    1509    1515    1725    8557#
EM118   053716          1522    1529    1536    1543    1549    1555    1561    1731    8571#
EM119   054031          1567    1573    1579    1737    8584#
EM120   054120          1584    8594#
EM121   054214          1590    1596    1602    1608    1614    8604#
EM122   054315          1621    1628    8615#
EM123   054416          1634    1640    1647    1653    1659    1665    167[illegible]  1743    8626#
EM124   054502          1677    1683    1689    1695    1701    1707    1713    1719    8635#
EM125   054550          1749    1755    1761    1767    1773    1779    8642#
EM126   054767          1785    1791    1797    1803    8669#
EM127   055034          1809    1815    1821    1827    8676#
EM128   055132          1833    1839    1845    1851    8687#
EM129   055213          1857    1863    1869    1875    8696#
EM130   055260          1881    1887    1893    1899    8703#
EM131   055342          1905    1911    1917    1923    8712#
EM132   055427          1929    1935    1941    1947    8721#
EM133   055466          1953    1959    1965    1971    8727#
EM134   055532          1977    1983    1989    1995    8733#
EM135   055577          2001    2007    2013    2019    8740#
EM136   055643          2025    2031    2037    2043    8747#
EM137   055717          2049    2055    2061    2067    8755#
EM138   055777          2073    2079    2085    2091    8764#
EM139   056107          2097    _103    2109    2115    8777#
EM140   056171          2121    212[illegible]  2133    2139    8786#
```

EM141	056253	2145	2151	2157	2163	8795#								
EM142	056327	2169	2175	2181	2187	8803#								
EM143	056425	2193	2199	2205	2211	8815#								
EM144	056520	2217	2223	2229	2235	8825#								
EM145	056601	2241	2247	8834#										
EM146	056666	2253	2259	8843#										
EM147	056740	2264	2269	8850#										
EM2000	057005	1311	1336	1384	1420	2644	2835	2898	2961	3016	3065	8857#		
EM2001	057063	1251	1271	1296	1321	1347	1367	1372	1396	1432	1444	1468	1492	1568
		2650	2841	2904	2967	3022	3071	8865#						
EM2002	057112	1256	1276	1301	1326	1352	1402	1438	1450	1474	1498	2654	2845	2908
		2971	3026	3075	8869#									
EM2003	057141	1241	1261	1281	1306	1331	1357	1378	1408	1504	1550	1585	1660	1678
		1762	1786	1810	1834	1858	1882	1906	1930	1954	1978	2002	2026	2050
		2074	2098	2122	2146	2170	2194	2218	2254	2265	2637	2825	2888	2951
		3009	3058	8873#										
EM2004	057204	1246	1684	8879#										
EM2005	057255	1266	1696	8886#										
EM2006	057322	1286	1414	8893#										
EM2007	057351	1291	1426	1708	8897#									
EM2008	057421	316	1342	1390	1714	8904#								
EM2009	057472	.362	8911#											
EM2010	057543	1456	1480	1702	8918#									
EM2011	057605	1462	1486	1720	8924#									
EM2012	057647	1510	1556	1574	1622	1666	1768	1792	1816	1840	1864	1888	1912	1936
		1960	1984	2008	2032	2056	2080	2104	2128	2152	2176	2200	2224	8930#
EM2013	057712	1516	1562	1580	1672	1774	1804	1828	1852	1876	1900	1924	1948	1972
		1936	2020	2044	2068	2092	2116	2140	2164	2188	2212	2236	2260	2270
		8936#												
EM2014	057736	1523	1635	8940#										
EM2015	060024	1530	1641	8949#										
EM2016	060116	1537	1648	8959#										
EM2017	060216	1544	1654	8970#										
EM2018	060316	1591	8981#											
EM2019	060346	1597	8985#											
EM2020	060431	1603	8994#											
EM2021	060514	1609	9003#											
EM2022	060565	1615	9010#											
EM2023	060626	1629	9016#											
EM2024	060700	1690	9023#											
EM2025	060750	1726	1732	1738	1744	1780	1798	1822	1846	1870	1894	1918	1942	1966
		1990	2014	2038	2062	2086	2110	2134	2158	2182	2206	2230	9030#	
EM2026	061010	1750	9036#											
EM2027	061045	1756	9041#											
EM2028	061120	2242	9049#											
EM2029	061156	2248	9054#											
ERRCNT	004242	2332#	2405#	7353#	7356									
ERRVEC=	000004	920#	2386	2387#	2398#	7001#	7002#	7014#	7015#	7052	7053#	7055#	7058#	
E ASOF	004176	2306#												
E BA	004164	2301#												
E CSI	004160	2299#	2505#	2506	2519#	2522	2565#	2584	2624#	2635	2672#	2698	2736#	2761
		2805#	2813	2823	2848	2850#	2868#	2876	2886	2911	2913#	2931#	2939	2949
		2974	2976#	2996#	3007	3045#	3056	3092#	3115	3160#	3178	3223#	3246	3293#
		3323	3364#	3380	3417#	3433	3477#	3493	3537#	3553	3597#	3613	3657#	3673
		3714#	3734	3781#	3801	3848#	3868	3915#	3935	3982#	4002	4049#	4069	4109#
		4129	4597#	4602	4649#	4650	4677	4703#	4708	4756#	4761	4823#	4826	4901#

```
                        4912*  4943*  4944   4960   4987*  4991   5038*  5046*  5053   5075*  5079   5126*  5130
                        5152*  5156   5203*  5207   5229*  5233   5258*  5282   5304*  5308   5355*  5359   5381*
                        5385   5433*  5437   5459*  5463   5511*  5515   5537*  5541   5588*  5592   5614*  5618
                        5665*  5669   5691*  5695   5743*  5747   5769*  5773   5821*  5825   5847*  5851   5899*
                        5903   5925*  5929   5975*  5979   6001*  6005   6053*  6057   6079*  6083   6136*  6141
                        6163*  6167   6220*  6225   6247*  6251   6305*  6310   6332*  6336   6389*  6394   6416*
                        6420   6473*  6478   6500*  6504   6557*  6562   6584*  6588   6642*  6647   6698*  6702
                        6723*  6727   6777*  6780*  6784   6805*  6809   6859*  6863   6884*  6888   6919*  6920*
                        6921*  6927   6936*  6938   7936   7939   7942   7945   7950   7953   7956   7965   7969
                        7974   7976   7981   7986
E.CS2   004170          2303*  4598*  4599*  4605   4704*  4705*  4711   4757*  4758*  4764   4824*  4829   4902*
                        4905   4913*  4914   4988*  4994   5047*  5056   5076*  5082   5127*  5133   5153*  5159
                        5204*  5210   5230*  5236   5279*  5285   5305*  5311   5356*  5362   5382*  5388   5434*
                        5440   5460*  5466   5512*  5518   5538*  5544   5589*  5595   5615*  5621   5666*  5672
                        5692*  5698   5744*  5750   5770*  5776   5822*  5828   5848*  5854   5900*  5906   5926*
                        5932   5976*  5982   6002*  6008   6054*  6060   6080*  6086   6138*  6144   6164*  6170
                        6222*  6228   6248*  6254   6307*  6313   6333*  6339   6391*  6397   6417*  6423   6475*
                        6481   6501*  6507   6559*  6565   6585*  6591   6643*  6644*  6650   6699*  6705   6724*
                        6730   6781*  6787   6806*  6812   6860*  6866   6885*  6891   7965   7976   7981
E.DA    004166          2302*
E.DB    004202          2308*
E.DCYL  004200          2307*
E.DS    004172          2304*  4600*  4611   4706*  4717   4759*  4770   4903*  4989*  5000   5048*  5062   5077*
                        5085   5128*  5136   5154*  5162   5205*  5213   5231*  5239   5280*  5288   5306*  5314
                        5357*  5365   5383*  5391   5435*  5443   5461*  5469   5513*  5521   5539*  5547   5590*
                        5598   5616*  5624   5667*  5675   5693*  5701   5745*  5753   5771*  5779   5823*  5831
                        5849*  5857   5901*  5909   5927*  5935   5977*  5985   6003*  6011   6055*  6063   6081*
                        6089   6139*  6147   6165*  6173   6223*  6231   6249*  6257   6308*  6316   6334*  6342
                        6392*  6400   6418*  6426   6476*  6484   6502*  6510   6560*  6568   6586*  6594   6645*
                        6653   6700*  6708   6725*  6733   6782*  6790   6807*  6815   6861*  6869   6886*  6894
                        7965   7976   7984
E.ECPS  004212          2312*
E.ECPT  004214          2313*
E.ER    004174          2305*  4601*  4608   4707*  4714   4760*  4767   4825*  4832   4904*  4990*  4997   5049*
                        5059   5078*  5088   5129*  5139   5155*  5165   5206*  5216   5232*  5242   5281*  5291
                        5307*  5317   5358*  5368   5384*  5394   5436*  5446   5462*  5472   5514*  5524   5540*
                        5550   5591*  5601   5617*  5627   5668*  5678   5694*  5704   5746*  5756   5772*  5782
                        5824*  5834   5850*  5860   5902*  5912   5928*  5938   5978*  5988   6004*  6014   6056*
                        6066   6082*  6092   6140*  6150   6166*  6176   6224*  6234   6250*  6260   6309*  6319
                        6335*  6345   6393*  6403   6419*  6429   6477*  6487   6503*  6513   6561*  6571   6587*
                        6597   6646*  6656   6701*  6711   6726*  6736   6783*  6793   6808*  6818   6862*  6872
                        6887*  6897   6922*  6930   6937*  6941   7965   7979   7984   7986
E.MR1   004204          2309*  2754*  2755*  2758*  2767   3728*  3731*  3740   3795*  3798*  3807   3862*  3865*
                        3874   3929*  3932*  3941   3996*  3999*  4008   4063*  4066*  4075   4123*  4126*  4135
                        7945
E.MR2   004206          2310*  2520*  2582*  2597   2633*  2648   2691*  2692*  2693*  2694*  2695*  2696*  2706
                        2711   2759*  2780   2806*  2832   2839   2851*  2869*  2895   2902   2914*  2932*  2958
                        2965   2977*  3005*  3020   3054*  3069   3109*  3133   3177*  3196   3240*  3264   3294*
                        3336   3378*  3393   3431*  3453   3491*  3513   3551*  3573   3611*  3633   3671*  3693
                        3732*  3760   3799*  3827   3866*  3894   3933*  3961   4000*  4028   4067*  4088   4127*
                        4155   4181*  4196   4213*  4236*  4252   4269*  4299*  4315   4330*  4345   4389*  4395*
                        4414   4438*  4474   4482*  4528*  4534*  4553   4667*  4682   4950*  4965   7936   7939
                        7942   7948   7950   7953   7958   7960   7963   7971
E.MR3   004210          2311*  2521*  2583*  2600   2634*  2652   2697*  2714   2760*  2783   2822*  2843   2885*
                        2906   2948*  2969   3006*  3024   3055*  3073   3110*  3111*  3112*  3113*  3114*  3128
                        3136   3159*  3191   3199   3206*  3241*  3242*  3243*  3244*  3245*  3259   3267   3311*
```

```
                    3312*  3313*  3314*  3315*  3316*  3317*  3318*  3319*  3322*  3331   3339   3379*  3396
                    3432*  3456   3492*  3516   3552*  3576   3612*  3636   3672*  3696   3733*  3763   3800*
                    3830   3867*  3897   3934*  3964   4001*  4031   4068*  4091   4128*  4158   4183*  4201
                    4206   4214*  4238*  4257   4262   4270*  4300*  4312   4331*  4342   4390*  4396*  4411
                    4439*  4471   4483*  4529*  4535*  4550   4663*  4666*  4687   4949*  4970   7936   7939
                    7942   7948   7950   7953   7958   7960   7963   7971
E.SPAR  004216      2314#
E.WC    004162      2300#
FMTE  = 000020      1014#
GNS   = ****** U    1076   2425   6972   6979   7918   7919   7920   7921   7922   7924   7926   7927   7928
                    7929   7930   7931   7932
GO    = 000001       977#  4987   5038   5046   5126   5203   5278   5355   5433   5511   5588   5665   5743
                    5821   5899   5975   6053   6136   6220   6305   6389   6473   6557   6642   6698   6920
GTSWR = 104406      2420   7924#
HDCODE  004250      2335#  2671#  2680   2692   2719*  2719   6115*  6117   6199*  6201   6284*  6286   6368*
                    6370   6452*  6454   6536*  6538   7942   7981
HT    = 000011       830#  7406   7447
HVRC  = 000400      1018#
IDAE  = 002000      1020#  6056   6140   6309   6393   6477   6561
IE    = 000100       978#  4806   4823   4852
ILF   = 000001      1010#  6922
ILLFUN  004270      2343#  6913#  6919   6924   6945*  6946   7986
INTR  = 000300       973#
IOTVEC= 000220       925#  2370*  2371*
IR    = 000200       997#  4599   4705   4758   4824   4902   4913   4988   5047   5076   5127   5153   5204
                    5230   5279   5305   5356   5382   5434   5460   5512   5538   5589   5615   5666   5692
                    5744   5770   5822   5848   5900   5926   5976   6002   6054   6080   6138   6164   6222
                    6248   6307   6333   6391   6417   6475   6501   6559   6585   6644   6699   6724   6781
                    6806   6860   6885
LF    = 000012       831#   441   7447
MCLK  = 000400      1051#  2512   2575   2626   2684   2746   2747   2815   2878   2941   2998   3047   3102
                    3170   3233   3304   3371   3424   3484   3544   3604   3664   3720   3721   3787   3788
                    3854   3855   3921   3922   3988   3989   4055   4056   4115   4116   4190   4191   4246
                    4247   4293   4323   4375   4376   4451   4452   4514   4515   4589   4644   4670   4695
                    4938   4953   4979   5027   5112   5189   5264   5341   5419   5497   5574   5651   5729
                    5807   5885   5961   6039   6122   6206   6291   6375   6459   6543   6628   6684   6760
                    6845
MDS   = 001000      1000#
MEMBAS= 172100       933#  7003
MEMERR  042450      7007   7020#
MEMVEC= 000114       932#  7007*  7008*
MERD  = 001000      1052#
MESMSK= 000017      1045#
MEWD  = 002000      1053#  2755   3728   3795   3862   3929   3996   4063   4123
MIND  = 000200      1050#
MSGCOD  004246      2334#  2735*  2742   2754   2760   2775   2787*  2788   3713*  3780*  3847*  3914*  3981*
                    4048*  4108*  4437   4446   4481*  7945
NSP   = 000100      1049#
NED   = 010000      1003#  6781   6860
NEM   = 004000      1002#
NEWPAS  005254      2479#  6995
NXF   = 000004      1012#  5902
OFFSET= 000015       966#  3293   3302   4049   4053
OFFVAL  004254      2337#  3292*  3301   3312   3320   3343*  3344   3363*  3416*  3476*  3536*  3596*  3656*
                    7953   7956
OFST  = 000004      1031#  5048   5050
```

```
OPI    = 020000          1023#
OPR001  050010          2429    8201#
OPR002  050037          2432    2443    2460    8205#
OPR003  050045          2439    8207#
OPR004  050075          2451    8212#
OR     = 000200           998#
PACK   = 000003           961#  2931    3417    3422    3714    3718    5727    5743
PARBIT  004260          2339#  4391*   4394*   4399    4406    4440*   4459    4466    4484*   4530*   4533*   4538    4545
PARM    004276          1083    2348#
PAR.EN= 000001           934#  7006
PAT    = 000020          1047#  4319    4323    4324    4509
PCA    = 004000          10_ #
PCD    = 010000          1055#
PGE    = 002000          1001#
PIE    = 020000          1039#  6223
PIRQ   = 177772           837#
PIRQVE= 000240            931#
PR0    = 000000           854#
PR1    = 000040           855#
PR2    = 000100           856#
PR3    = 000140           857#
PR4    = 000200           858#
PR5    = 000240           859#  2327
PR6    = 000300           860#
PR7    = 000340           861#  2357    2480    4800    4812    4854    7002    7008    7863    7876
PS     = 177776           834#   835
PSW    = 177776           835#
PWRVEC= 000024            926#  2376*   2377*   7862*   7863*   7875*   7876*
P.CS1   004220          2318#  5065*   5142*   5219*   5294*   5371*   5449*   5527*   5604*   5681*   5759*   5837*   5915*
                        5991*  6069*   6153*   6237*   6322*   6406*   6490*   6574*   6714*   6796*   6875*   7979
P.CS2   004222          2319#  5066*   5143*   5220*   5295*   5372*   5450*   5528*   5605*   5682*   5760*   5838*   5916*
                        5992*  6070*   6154*   6238*   6323*   6407*   6491*   6575*   6715*   6797*   6876*   7979
P.DS    004224          2320#  5067*   5144*   5221*   5296*   5373*   5451*   5529*   5606*   5683*   5761*   5839*   5917*
                        5993*  6071*   6155*   6239*   6324*   6408*   6492*   6576*   6716*   6798*   6877*   7979
P.ER    004226          2321#  5068*   5145*   5222*   5297*   5374*   5452*   5530*   5607*   5684*   5762*   5840*   5918*
                        5994*  6072*   6156*   6240*   6325*   6409*   6493*   6577*   6717*   6799*   6878*   7979
RDCHR  = 104410          7705   7927#
RDDATA= 000021            968#
RDGATE= 100000           1058#
RDHEAD= 000025            970#
RDLIN  = 104411          7777   7928#
RDOCT  = 104412          2433    2444    2461    7929#
RDY    = 000200           979#  4597    4703    4756    4823    4901    4912    4987    5038    5046    5075    5126    5152
                        5203    5229    5278    5304    5355    5381    5433    5459    5511    5537    5588    5614    5665
                        5691    5743    5769    5821    5847    5899    5925    5975    6001    6053    6079    6136    6163
                        6220    6247    6305    6332    6389    6416    6473    6500    6557    6584    6642    6698    6723
                        6777    6805    6859    6884    6921    6936
RECAL  = 000013           965#  2995    2996    3657    3662    3982    3986    6037    6053
RESREG= 104414          7352   7931#
RESTRT  004306          1081   2351#
RESVEC= 000010            921#
RKASOF= 000016            948#  3301*   3368*   3421*   3481*   3541*   3601*   3661*
RKBA   = 000004           943#
RKCS1  = 000000           941#  2498*   2500*   2504    2516    2570*   2573*   2579    2587*   2621*   2623*   2630    2639*
                        2677*  2682*   2688    2701*   2741*   2744*   2750    2764*   2770*   2811*   2813*   2819    2827*
                        2874*  2876*   2882    2890*   2937*   2939*   2945    2953*   2993*   2995*   3002    3011*   3042*
```

```
                    3044*  3051   3060*  3097*  3100*  3106   3118*  3165*  3168*  3174   3181*  3228*  3231*
                    3237   3249*  3299*  3302*  3308   3326*  3365*  3369*  3375   3383*  3418*  3422*  3428
                    3436*  3478*  3482*  3488   3496*  3538*  3542*  3548   3556*  3598*  3602*  3608   3616*
                    3658*  3662*  3668   3676*   715*  3718*  3724   3737*  3743*  3782*  3785*  3791   3804*
                    3810*  3849*  3852*  3858   3871*  3877*  3916*  3919*  3925   3938*  3944*  3983*  3986*
                    3992   4005*  4011*  4050*  4053*  4059   4072*  4078*  4110*  4113*  4119   4132*  4138*
                    4179*  4188*  4234*  4243*  4269*  4291*  4318*  4321*  4369*  4373*  4445*  4449*  4507*
                    4512*  4587*  4593   4642*  4648   4674   4699   4746*  4748   4752   4806*  4808   4820
                    4844*  4846   4852*  4891*  4893   4897   4909   4917*  4936*  4942   4957   4983   5025*
                    5033   5037   5042   5070*  5071   5110*  5118   5122   5147*  5148   5187*  5195   5199
                    5224*  5225   5262*  5270   5274   5299*  5300   5339*  5347   5351   5376*  5377   5417*
                    5425   5429   5454*  5455   5495*  5503   5507   5532*  5533   5572*  5580   5584   5609*
                    5610   5649*  5657   5661   5686*  5687   5727*  5735   5739   5764*  5765   5805*  5813
                    5817   5842*  5843   5883*  5891   5895   5920*  5921   5959*  5967   5971   5996*  5997
                    6037*  6045   6049   6074*  6075   6120*  6128   6132   6158*  6159   6204*  6212   6216
                    6242*  6243   6289*  6297   6301   6327*  6328   6373*  6381   6385   6411*  6412   6457*
                    6465   6469   6495*  6496   6541*  6549   6553   6579*  6580   6626*  6634   6638   6682*
                    6690   6694   6718*  6719   6758*  6766   6773   6800*  6801   6843*  6851   6855   6879*
                    6880   6918*  6924*  6925   6933*  6934
RkCS2 = 000010        945*  2572*  4187*  4372*  4448*  4511*  4584*  4586*  4594   4639*  4641*  4700   4744*
                    4745*  4753   4796*  4797*  4821   4843*  4869*  4888*  4890*  4898   4910   4932*  4984
                    5021*  5043   5072   5106*  5123   5149   5183*  5200   5226   5260*  5275   5301   5335*
                    5352   5378   5413*  5430   5456   5491*  5508   5534   5568*  5585   5611   5645*  5662
                    5688   5723*  5740   5766   5801*  5818   5844   5879*  5896   5922   5957*  5972   5998
                    6033*  6050   6076   6111*  6133   6160   6195*  6217   6244   6280*  6302   6329   6364*
                    6386   6413   6448*  6470   6497   6532*  6554   6581   6621*  6625*  6639   6678*  6681*
                    6695   6720   6756*  6774   6802   6841*  6856   6881
RKDA  = 000006         944*  2681*  4242*  4935*  5024*  5109*  5186*  5338*  5416*  5494*  5571*  5648*  5726*
                    5804*  5882*  6036*  6118*  6202*  6287*  6371*  6455*  6539*  6624*
RKDB  = 000024         950*  4889*  4908
RKDCYL= 000020         949*  3099*  3167*  3230*  3367*  3420*  3480*  3540*  3600*  3660*  4241*  4934*  5023*
                    5108*  5185*  5337*  5415*  5493*  5570*  5647*  5725*  5803*  5881*  6035*  6113*  6197*
                    6282*  6366*  6450*  6534*  6623*  6680*
RKDS  = 000012         946*  4595   4701   4754   4899   4985   5044   5073   5124   5150   5201   5227   5276
                    5302   5353   5379   5431   5457   5509   5535   5586   5612   5663   5689   5741   5767
                    5819   5845   5897   5923   5973   5999   6051   6077   6134   6161   6218   6245   6303
                    6330   6387   6414   6471   6498   6555   6582   6640   6696   6721   6775   6803   6857
                    6882
RKECPS= 000030         954*
RKECPT= 000032         955*
RKER  = 000014         947*  4596   4702   4755   4822   4900   4911   4986   5045   5074   5125   5151   5202
                    5228   5277   5303   5354   5380   5432   5459   5510   5536   5587   5613   5664   5690
                    5742   5768   5820   5846   5898   5924   5974   6000   6052   6078   6135   6162   6219
                    6246   6304   6331   6388   6415   6472   6499   6556   6583   6641   6697   6722   6776
                    6804   6858   6883   6926   6935
RKMR1 = 000026         951*  2499*  2512*  2513*  2571*  2575*  2576*  2622*  2626*  2627*  2678*  2684*  2685*
                    2742*  2743*  2746*  2747*  2751   2812*  2815*  2816*  2875*  2878*  2879*  2938*  2941*
                    2942*  2994*  2998*  2999*  3043*  3047*  3048*  3098*  3102*  3103*  3166*  3170*  3171*
                    3229*  3233*  3234*  3300*  3304*  3305*  3366*  3371*  3372*  3419*  3424*  3425*  3479*
                    3484*  3485*  3539*  3544*  3545*  3599*  3604*  3605*  3659*  3664*  3665*  3716*  3720*
                    3721*  3725   3783*  3787*  3788*  3792   3850*  3854*  3855*  3859   3917*  3921*  3922*
                    3926   3984*  3988*  3989*  3993   4051*  4055*  4056*  4060   4111*  4115*  4116*  4120
                    4185*  4190*  4191*  4240*  4246*  4247*  4290*  4293*  4294*  4319*  4323*  4324*  4370*
                    4371*  4375*  4376*  4446*  4447*  4451*  4452*  4508*  4509*  4514*  4515*  4585*  4589*
                    4590*  4640*  4644*  4645*  4670*  4671*  4695*  4696*  4933*  4938*  4939*  4953*  4954*
                    4979*  4980*  5022*  5027*  5028*  5031*  5107*  5112*  5113*  5116*  5184*  5189*  5190*
```

```
                    5193*  5261*  5264*  5265*  5268*  5336*  5341*  5342*  5345*  5414*  5419*  5420*  5423*
                    5492*  5497*  5498*  5501*  5569*  5574*  5575*  5578*  5646*  5651*  5652*  5655*  5724*
                    5729*  5730*  5733*  5802*  5807*  5808*  5811*  5880*  5885*  5886*  5889*  5958*  5961*
                    5962*  5965*  6034*  6039*  6040*  6043*  6112*  6122*  6123*  6126*  6196*  6206*  6207*
                    6210*  6281*  6291*  6292*  6295*  6365*  6375*  6376*  6379*  6449*  6459*  6460*  6463*
                    6533*  6543*  6544*  6547*  6622*  6628*  6629*  6632*  6679*  6684*  6685*  6688*  6757*
                    6760*  6761*  6764*  6842*  6845*  6846*  6849*  6923*
RKMR2 = 000034       952#  2517   2580   2631   2689   2752   2820   2883   2946   3003   3052   3107   3175
                    3238   3309   3376   3429   3489   3549   3609   3669   3726   3793   3860   3927   3994
                    4061   4121   4194   4250   4297   4328   4379   4455   4518   4675   4958
RKMR3 = 000036       953#  2518   2581   2632   2690   2753   2821   2884   2947   3004   3053   3108   3176
                    3239   3310   3377   3430   3490   3550   3610   3670   3727   3794   3861   3928   3995
                    4062   4122   4195   4251   4298   4329   4380   4456   4519   4676   4959
RKPRI   004236      2327#  2477*
RKSPAR= 000022       956#
RKVEC   004234      2326#  2475*  2476*  4798   4835*  4842*  4851*  4870
RKWC  = 000002       942#
RLS   = 000010       994#
SAVREG= 104413      7301   7930#
SAVSWR  004274      2345#  7861*  7880
SCLR  = 000040       996#  4584   4639   4744   4796   4869   4888   4932   5021   5106   5183   5260   5335
                    5413   5491   5568   5645   5723   5801   5879   5957   6033   6111   6195   6280   6364
                    6448   6532   6621   6678   6756   6841
SCOP1 = 104415      2603   2717   2786   2847   2910   2973   3139   3202   3270   3342   4417   4477   4556
                    4614   4720   4773   6659   6944   7932#
SCOP1$  043224      7173#  7932
SEEK  = 000017       967#  3092   3100   3160   3168   3223   3231   4109   4113   4243   4936   4943   4987
                    5025   5038   5046   5110   5126   5187   5203   5339   5355   5417   5433   5495   5511
                    5572   5588   5649   5665   6120   6136   6204   6220   6289   6305   6373   6389   6457
                    6473   6541   6557   6626   6642   6682   6698
SELDRV= 000001       960#  2500   2505   2519   2565   2573   2623   2624   2672   2682   2736   2744   3364
                    3369   4188   4291   4321   4373   4449   4512   4587   4642   4649   4746   4806   4844
                    4891   5262   5278   6758   6843
SFTCNT  004256      2338#  4180*  4216*  4217   4235*  4272*  4273   7960
SKI   = 000002      1011#  5824
SPACE2  050117      7335   7347   8216#
SPAR  = 020000       986#  5203   6698   6778   6780
SPDLSS= 000020      1033#  5048   5050   5590   5745   5823   5901   6055   6308   6476
SRTFLG  004240      2328#  2348*  2351*  2354*  2427
SRTSPL= 000011       964#  3044   3045   3597   3602   3915   3919   5959   5975
STACK = 001100       825#  2356   2368   7028   7363   7877
STALL   004264      2341#  6770
START   004316      1080   2354#
START1  004322      2349   2352   2355#
STKLMT= 177774       836#
SVAL  = 100000      1041#  5048   5128   5205   5280   5357   5435   5513   5590   5667   5745   5823   5901
                    5977   6055   6139   6223   6308   6392   6476   6560   6782
SWR     001140      1144#  2366   2388*  2390   2396*  2403*  2418   7024   7047   7061   7064   7079   7086
                    7173   7255   7262   7274   7278   7354   7601   7638*  7861   7880*
SWREG   000176      1078#  2396   2418   7601   7614
SW0   = 000001       889#
SW00  = 000001       879#   889
SW01  = 000002       878#   888
SW02  = 000004       877#   887
SW03  = 000010       876#   886
SW04  = 000020       875#   885
```

```
SW05   = 000040            874#    884
SW06   = 000100            873#    883
SW07   = 000200            872#    882
SW08   = 000400            871#    881
SW09   = 001000            870#    880
SW1    = 000002            888#
SW10   = 002000            869#
SW11   = 004000            868#
SW12   = 010000            867#   7354
SW13   = 020000            866#
SW14   = 040000            865#
SW15   = 100000            864#
SW2    = 000004            887#
SW3    = 000010            886#
SW4    = 000020            885#
SW5    = 000040            884#
SW6    = 000100            883#
SW7    = 000200            882#
SW8    = 000400            881#
SW9    = 001000            880#   7024   7173
S.CLR  = 000400           1066#   2806   2831   3491   3500   3799   3814
S.FMT  = 001000           1067#   2633   2642   2831   2851   2894   2914   2957   2977   4236   4237   4949
S.PACK= 004000            1069#   2932   2957   3431   3440   3732   3747
S.RECL= 000040            1063#   3005   3014   3671   3680   4000   4015
S.RTC  = 000200           1065#
S.SEEK= 000020            1062#   3109   3121   3177   3184   3240   3252   4127   4142   4236   4237   4949
S.STSP= 000100            1064#   3054   3063   3611   3620   3933   3948
S.UNLD= 002000            1068#   2869   2894   3551   3560   3866   3881
TBITVE= 000014             922#
TKVEC  = 000060            929#
TPVEC  = 000064            930#
TRAPPC  004272            2344#   7021*  7935
TRAPVE= 000034             928#   2374*  2375*
TRTVEC= 000014             923#
TST1    005272            2481    2495#  7106
TST10   010032            2912    2927#  7113
TST100  041712            6898    6911#  7169
TST11   010350            2975    2989#  7114
TST12   010614            3012    3018   3025   3038#  7115
TST13   011060            3061    3067   3074   3088#  7116
TST14   011376            3155#   7117
TST15   011706            3204    3219#  7118
TST16   012240            3289#   7119
TST17   012602            3359#   7120
TST2    005576            2509    2525   2547   2561#  7107
TST20   013046            3384    3391   3397   3412#  7121
TST21   013326            3437    3444   3451   3457   3472#  7122
TST22   013606            3497    3504   3511   3517   3532#  7123
TST23   014066            3557    3564   3571   3577   3592#  7124
TST24   014346            3617    3624   3631   3637   3652#  7125
TST25   014626            3677    3684   3691   3697   3710#  7126
TST26   015140            3738    3744   3751   3758   3764   3777#  7127
TST27   015452            3805    3811   3818   3825   3831   3844#  7128
TST3    006056            2617#   7108
TST30   015764            3872    3878   3885   3892   3898   3911#  7129
TST31   016276            3939    3945   3952   3959   3965   3978#  7130
```

```
TST32   016610          4006    4012    4019    4026    4032    4045#   7131
TST33   017106          4073    4079    4086    4092    4105#   7132
TST34   017420          4133    4139    4146    4153    4159    4176#   7133
TST35   017656          4199    4204    4231#   7134
TST36   020122          4255    4260    4286#   7135
TST37   020456          4335    4340    4346    4361#   7136
TST4    006322          2640    2646    2653    2668#   7109
TST40   021022          4433#   7137
TST41   021346          4479    4499#   7138
TST42   021712          4576#   7139
TST43   022166          4631#   7140
TST44   022716          4723    4736#   7141
TST45   023156          4793#   7142
TST46   023562          4885#   7143
TST47   024012          4929#   7144
TST5    006634          2732#   7110
TST50   024446          4947    4963    4968    4973    5001    5018#   7145
TST51   025130          5089    5103#   7146
TST52   025562          5166    5180#   7147
TST53   026214          5243    5257#   7148
TST54   026632          5318    5332#   7149
TST55   027264          5395    5410#   7150
TST56   027716          5473    5488#   7151
TST57   030350          5551    5565#   7152
TST6    007176          2801#   7111
TST60   031002          5628    5642#   7153
TST61   031434          5705    5720#   7154
TST62   032066          5783    5798#   7155
TST63   032520          5861    5876#   7156
TST64   033152          5939    5954#   7157
TST65   033570          6015    6030#   7158
TST66   034222          6093    6108#   7159
TST67   034704          6177    6192#   7160
TST7    007514          2849    2864#   7112
TST70   035366          6261    6277#   7161
TST71   036050          6346    6361#   7162
TST72   036532          6430    6445#   7163
TST73   037214          6514    6529#   7164
TST74   037676          6598    6613#   7165
TST75   040210          6675#   7166
TST76   040636          6737    6753#   7167
TST77   041300          6819    6838#   7168
TYPDS = 104405          6976    6983    7922#
TYPE  = 104401          2412    2429    2432    2439    2443    2451    2460    6970    6977    6984    7257    7265    7311
                        7312    7316    7317    7324    7335    7337    7347    7348    7350    7358    7411    7507    7582
                        7612    7613    7616    7629    7640    7659    7712    7718    7723    7727    7732    7733    7735
                        7738    7742    7806    7808    7878    7918#
TYPERR  043716          7264    7301#
TYPOC = 104402          2431    2442    2459    7332    7615    7919#
TYPON = 104404          7921#
TYPOS = 104403          7920#
T.ASOF  004136          2287#
T.BA    004124          2282#
T.CS1   004120          2280#   2504*   2506    2516*   2522    2579*   2584    2630*   2635    2688*   2698    2750*   2761
                        2819*   2823    2882*   2886    2945*   2949    3002*   3007    3051*   3056    3106*   3115    3174*
                        3178    3237*   3246    3308*   3323    3375*   3380    3428*   3433    3488*   3493    3548*   3553
```

```
                    3608*  3613   3668*  3673   3724*  3734   3791*  3801   3858*  3868   3925*  3935   3992*
                    4002   4059*  4069   4119*  4129   4593*  4602   4648*  4650   4674*  4677   4699*  4708
                    4752*  4761   4820*  4826   4897*  4909*  4942*  4944   4957*  4960   4983*  4991   5037*
                    5042*  5053   5065   5071*  5079   5122*  5130   5142   5148*  5156   5199*  5207   5219
                    5225*  5233   5274*  5282   5294   5300*  5308   5351*  5359   5371   5377*  5385   5429*
                    5437   5449   5455*  5463   5507*  5515   5527   5533*  5541   5584*  5592   5604   5610*
                    5618   5661*  5669   5681   5687*  5695   5739*  5747   5759   5765*  5773   5817*  5825
                    5837   5843*  5851   5895*  5903   5915   5921*  5929   5971*  5979   5991   5997*  6005
                    6049*  6057   6069   6075*  6083   6132*  6141   6153   6159*  6167   6216*  6225   6237
                    6243*  6251   6301*  6310   6322   6328*  6336   6385*  6394   6406   6412*  6420   6469*
                    6478   6490   6496*  6504   6553*  6562   6574   6580*  6588   6638*  6647   6694*  6702
                    6714   6719*  6727   6773*  6778   6784   6796   6801*  6809   6855*  6863   6875   6880*
                    6888   6925*  6927   6934*  6938   7936   7939   7942   7945   7950   7953   7956   7965
                    7969   7974   7976   7981   7986
T.CS2    004130     2284#  4594*  4605   4700*  4711   4753*  4764   4821*  4829   4898*  4905   4910*  4914
                    4984*  4994   5043*  5056   5066   5072*  5082   5123*  5133   5143   5149*  5159   5200*
                    5210   5220   5226*  5236   5275*  5285   5295   5301*  5311   5352*  5362   5372   5378*
                    5388   5430*  5440   5450   5456*  5466   5508*  5518   5528   5534*  5544   5585*  5595
                    5605   5611*  5621   5662*  5672   5682   5688*  5698   5740*  5750   5760   5766*  5776
                    5818*  5828   5838   5844*  5854   5896*  5906   5916   5922*  5932   5972*  5982   5992
                    5998*  6008   6050*  6060   6070   6076*  6086   6133*  6144   6154   6160*  6170   6217*
                    6228   6238   6244*  6254   6302*  6313   6323   6329*  6339   6386*  6397   6407   6413*
                    6423   6470*  6481   6491   6497*  6507   6554*  6565   6575   6581*  6591   6639*  6650
                    6695*  6705   6715   6720*  6730   6774*  6787   6797   6802*  6812   6856*  6866   6876
                    6881*  6891   7965   7976   7981
T.DA     004126     2283#
T.DB     004142     2289#
T.DCYL   004140     2288#
T.DS     004132     2285#  4595*  4611   4701*  4717   4754*  4770   4899*  4985*  5000   5044*  5050   5062
                    5067   5073*  5085   5124*  5136   5144   5150*  5162   5201*  5213   5221   5227*  5239
                    5276*  5288   5296   5302*  5314   5353*  5365   5373   5379*  5391   5431*  5443   5451
                    5457*  5469   5509*  5521   5529   5535*  5547   5586*  5598   5606   5612*  5624   5663*
                    5675   5683   5689*  5701   5741*  5753   5761   5767*  5779   5819*  5831   5839   5845*
                    5857   5897*  5909   5917   5923*  5935   5973*  5985   5993   5999*  6011   6051*  6063
                    6071   6077*  6089   6134*  6147   6155   6161*  6173   6218*  6231   6239   6245*  6257
                    6303*  6316   6324   6330*  6342   6387*  6400   6408   6414*  6426   6471*  6484   6492
                    6498*  6510   6555*  6568   6576   6582*  6594   6640*  6653   6696*  6708   6716   6721*
                    6733   6775*  6790   6798   6803*  6815   6857*  6869   6877   6882*  6894   7965   7976
                    7984
T.ECPS   004152     2293#
T.ECPT   004154     2294#
T.ER     004134     2286#  4596*  4608   4702*  4714   4755*  4767   4822*  4832   4900*  4911*  4986*  4997
                    5045*  5059   5068   5074*  5088   5125*  5139   5145   5151*  5165   5202*  5216   5222
                    5228*  5242   5277*  5291   5297   5303*  5317   5354*  5368   5374   5380*  5394   5432*
                    5446   5452   5458*  5472   5510*  5524   5530   5536*  5550   5587*  5601   5607   5613*
                    5627   5664*  5678   5684   5690*  5704   5742*  5756   5762   5768*  5782   5820*  5834
                    5840   5846*  5860   5898*  5912   5918   5924*  5938   5974*  5988   5994   6000*  6014
                    6052*  6066   6072   6078*  6092   6135*  6150   6156   6162*  6176   6219*  6234   6240
                    6246*  6260   6304*  6319   6325   6331*  6345   6388*  6403   6409   6415*  6429   6472*
                    6487   6493   6499*  6513   6556*  6571   6577   6583*  6597   6641*  6656   6697*  6711
                    6717   6722*  6736   6776*  6793   6799   6804*  6818   6858*  6872   6878   6883*  6897
                    6926*  6930   6935*  6941   7965   7979   7984   7986
T.MR1    004144     2290#  2751*  2756   2767   3725*  3729   3740   3792*  3796   3807   3859*  3863   3874
                    3926*  3930   3941   3993*  3997   4008   4060*  4064   4075   4120*  4124   4135   7945
T.MR2    004146     2291#  2517*  2527   2530   2534   2537   2580*  2590   2597   2631*  2642   2648   2689*
                    2704   2711   2752*  2780   2820*  2830   2839   2883*  2893   2902   2946*  2956   2965
```

```
                        3003*  3014   3020   3052*  3063   3069   3107*  3121   3133   3175*  3184   3196   3238*
                        3252   3264   3309*  3336   3376*  3393   3429*  3440   3453   3489*  3500   3513   3549*
                        3560   3573   3609*  3620   3633   3669*  3680   3693   3726*  3747   3760   3793*  3814
                        3827   3860*  3881   3894   3927*  3948   3961   3994*  4015   4028   4061*  4088   4121*
                        4142   4155   4194*  4196   4250*  4252   4297*  4307   4315   4328*  4337   4345   4379*
                        4404   4414   4455*  4464   4474   4518*  4543   4553   4675*  4682   4958*  4965   7936
                        7939   7942   7948   7950   7953   7958   7960   7963   7971
T.MR3   004150          2292*  2518*  2540   2543   2546   2581*  2600   2632*  2652   2690*  2714   2753*  2773
                        2783   2821*  2843   2884*  2906   2947*  2969   3004*  3024   3053*  3073   3108*  3126
                        3136   3176*  3189   3199   3239*  3257   3267   3310*  3329   3339   3377*  3387   3396
                        3430*  3447   3456   3490*  3507   3516   3550*  3567   3576   3610*  3627   3636   3670*
                        3687   3696   3727*  3754   3763   3794*  3821   3830   3861*  3888   3897   3928*  3955
                        3964   3995*  4022   4031   4062*  4082   4091   4122*  4149   4158   4195*  4201   4251*
                        4257   4298*  4301   4312   4329*  4332   4342   4380*  4397   4411   4456*  4457   4471
                        4519*  4536   4550   4676*  4687   4959*  4970   7936   7939   7942   7948   7950   7953
                        7958   7960   7963   7971
T.SPAR  004156          2295*
T.WC    004122          2281*
UFE     = 000400         999*
UNLOAD= 000007           963*  2868   3537   3542   3848   3852   5805   5821
UNS     = 040000        1024*  6393   6477   6561
UPE     = 020000        1004*
U.MR2   004230          2322*  4182*  4237*  7960
U.MR3   004232          2323*  4184*  4239*  7960
VV      = 000100        1035*  5048   5050
WAITIM  004262          2340*  4747   4807   4845   4892   5032   5117   5194   5269   5346   5424   5502   5579
                        5656   5734   5812   5890   5966   6044   6127   6211   6296   6380   6464   6548   6633
                        6689   6765   6850
WCE     = 040000        1005*
WLE     = 004000        1021*  5746
WRDATA= 000023           969*
WRHEAD= 000027           971*
WRL     = 004000        1038*  5048   5050
WRTCHK= 000031           972*
WRTGAT= 040000        1057*
$APTHD  001000        1104   1110*
$ASTAT= ****** U      7213   7228
$ATYC   043274        7184   7186*
$ATY1   043250        7182*
$ATY3   043256        7183*  7396
$ATY4   043266        7185*  7270
$AUTOB  001134        1141*  2422*  7609   7758
$BASE   001270        1213*  2430   2438*  2497   2563   2619   2670   2734   2803   2866   2929   2991   3040
                      3090   3157   3221   3291   3361   3414   3474   3534   3594   3654   3712   3779   3846
                      3913   3980   4047   4107   4178   4233   4288   4363   4435   4501   4578   4633   4738
                      4795   4887   4931   5020   5105   5182   5259   5334   5412   5490   5567   5644   5722
                      5800   5878   5956   6032   6110   6194   6279   6363   6447   6531   6615   6677   6755
                      6840
$BDADR  001122        1136*
$BDDAT  001126        1138*
$BELL   001204        1164*  7257   7290
$CDW1   001274        1215*
$CDW2   001276        1216*
$CHARC  044456        7413*  7423*  7430   7439*  7444*
$CKSWR  045134        7601*  7926
$CMTAG  001100        1124*  2363   2364   2372   2378   2379   2380
```

```
$CM3  = 000000  1154#
$CM4  = 000010  1154#  1155#  1156#  1157#  1158#  1159#  1160#  1161#  1162#
$CNTLG  046007  7612  7753#
$CNTLU  046002  7629  7727  7752#
$CPUOP  001242  1187#
$CRLF   001211  1166#  6984  7265  7290  7311  7316  7324  7337  7350  7412  7447  7640  7732
                7752  7811
$DBLK   045124  7548  7582  7590#
$DEVCT  001224  1178#
$DEVM   001272  1214#
$DOAGN  042350  6966  6987  6993#
$DTBL   045114  7551  7586#
$ENDAD  042340  1090  2410  6989#  7285
$ENDCT  042204  2378  6968#
$ENULL  042354  6996#
$ENV    001234  1183#  2416  7191  7215  7267  7391
$ENVM   001235  1184#  2401  7193  7393  7398
$EOP    042150  6958#  7364
$EOPCT  042176  2378*  6965#  6969  7362*
$ERFLG  001103  1127#  7036  7075  7077  7083*  7105  7175  7252*  7290
$ERMAX  001115  1133#  2381*  7077  7100*  7105
$ERROR  043516  2372  7250#
$ERRPC  001116  1134#  7259*  7260*  7261  7290  7936  7939  7942  7945  7950  7953  7956  7960
                7963  7965  7969  7971  7973  7974  7976  7981  7986
$ERRTB  001300  1233#  7308
$ERTTL  001112  1131#  6981  6985*  7258*  7290
$ESCAP  001202  1163#  2380*  7020*  7023*  7099*  7281  7283  7290
$ETABL  001234  1182#
$ETEND  001300  1116  1217#
$FATAL  001216  1175#  7219#
$FFLG   043514  7182#  7185#  7213  7222*  7230#
$FILLC  001156  1152#  7416  7447
$FILLS  001155  1151#  7447
$GDADR  001120  1135#
$GDDAT  001124  1137#
$GET42  042330  6986#
$GTSWR  045204  7613#  7924
$HD   = 000000  810
$HIBTS  001000  1111#
$HIOCT  046174  7799*  7810#
$ICNT   001104  1128#  7090*  7091  7093*  7104
$INTAG  001135  1142#  7641  7758
$ITEMB  001114  1132#  7261*  7269  7290  7302
$LF     001212  1167#  7290  7447  7742  7752  7811
$LFLG   043513  7223*  7229#
$LPADR  001106  1129#  2382*  7072*  7081*  7097*  7102  7104  7881
$LPERR  001110  1130#  2383*  2566*  2673*  2737*  2807*  2870*  2933*  3093*  3161*  3224*  3295*  4365*
                4441*  4503*  4580*  4635*  4740*  6617*  6914*  7029  7081  7098*  7104  7177  7280
$MADR1  001246  1200#
$MADR2  001252  1204#
$MADR3  001256  1207#
$MADR4  001262  1210#
$MAIL   001214  1112  1116  1173#  2400  2416  7096  7267  7391
$MAMS1  001244  1194#
$MAMS2  001250  1202#
$MAMS3  001254  1205#
```

```
$MAMS4  001260          1208#
$MBADR  001002          1112#
$MFLG   043512          7183*   7189    7224*   7228#
$MNEW   046025          7616    7756#
$MSGAD  001230          1180#   7199*   7202
$MSGLG  001232          1181#   7204#
$MSGTY  001214          1174#   7197    7205*   7217    7221*
$MSWR   046014          7613    7754#
$MTYP1  001245          1195#
$MTYP2  001251          1203#
$MTYP3  001255          1206#
$MTYP4  001261          1209#
$MXCNT  043022          7094    7104#
$NULL   001154          1150#   7418    7447
$NWTST= 000001          2485#   2487    2550#   2552    2608#   2610    2657#   2659    2722#   2724    2791#   2793    2854#
                        2856    2917#   2919    2980#   2982    3029#   3031    3078#   3080    3144#   3146    3209#   3211
                        3278#   3280    3347#   3349    3400#   3402    3460#   3462    3520#   3522    3580#   3582    3640#
                        3642    3700#   3702    3767#   3769    3834#   3836    3901#   3903    3968#   3970    4035#   4037
                        4095#   4097    4165#   4167    4220#   4222    4276#   4278    4349#   4351    4422#   4424    4487#
                        4489    4564#   4566    4619#   4621    4726#   4728    4778#   4780    4875#   4877    4919#   4921
                        5006#   5008    5092#   5094    5169#   5171    5246#   5248    5321#   5323    5398#   5400    5476#
                        5478    5554#   5556    5631#   5633    5708#   5710    5786#   5788    5864#   5866    5942#   5944
                        6018#   6020    6096#   6098    6180#   6182    6264#   6266    6349#   6351    6433#   6435    6517#
                        6519    6601#   6603    6664#   6666    6740#   6742    6822#   6824    6903#   6905
$OCNT   044704          7479*   7508*   7521#
$OMODE  044706          7474*   7478*   7483    7486*   7497*   7523#
$OVER   043006          7048    7073    7082    7092    7101#
$PASS   001222          1177#   2400*   6962*   6963*   6974    6996    7088    7105
$PASTM  001006          1114#
$POWER  046414          7878    7884#
$PWRCT  046410          7869*   7870*   7871*   7873*   7883#
$PWRDN  046272          2376    7861#   7875
$PWRUP  046320          7862    7869#
$QUES   001210          1165#   7290    7447    7659    7735    7752    7808    7811
$RDCHR  045416          7672#   7927
$RDDEC= ****** U        7930
$RDLIN  045536          7700#   7928
$RDOCT  046036          7772#   7929
$RDSZ = 000010          7693#
$RESRE  046234          7844#   7931
$RTNAD  042352          6995#
$R2A  = ****** U        7932
$SAVRE  046176          7828#   7930
$SCOPE  042512          2370    7045#
$SETUP= 000137          2348#   2369    2370    2372    2374    2376    2378    2379    2380    2382    2410    2413    6960
                        7046    7251    7277    7285    7596    7758
$STUP = 177777          2348#
$SVLAD  042752          7056    7095#
$SVPC = 000220          1088#   1093
$SWR  = 167400           800#    810     814     815     816     817     818     819     820     821    1162    1163    1164
                        2379    2380    2382    2383    2496    2562    2618    2669    2733    2802    2865    2928    2990
                        3039    3089    3156    3220    3290    3360    3413    3473    3533    3593    3653    3711    3778
                        3845    3912    3979    4046    4106    4177    4232    4287    4362    4434    4500    4577    4632
                        4737    4794    4886    4930    5019    5104    5181    5258    5333    5411    5489    5566    5643
                        5721    5799    5877    5955    6031    6109    6193    6278    6362    6446    6530    6614    6676
                        6754    6839    6912    6955    6961    6988    6994    6996    7037    7038    7039    7040    7041
```

```
                    7047   7059   7061   7062   7075   7076   7077   7084   7085   7086   7098   7101   7104
                    7242   7243   7244   7245   7246   7255   7262   7274   7278   7290
$SWREG  001236      1185#  2403
$SWRMK= 000000       821    822   7041   7042   7065
$SWO8T  043024      7071   7105#
$TESTN  001220      1176#  7096*  7935   7936   7939   7942   7945   7950   7953   7956   7960   7963   7965
                    7969   7971   7973   7974   7976   7981   7986
$TIMES  001200      1162#  2379*  2496*  2562*  2618*  2669*  2733*  2802*  2865*  2928*  2990*  3039*  3089*
                    3156*  3220*  3290*  3360*  3413*  3473*  3533*  3593*  3653*  3711*  3778*  3845*  3912*
                    3979*  4046*  4106*  4177*  4232*  4287*  4362*  4434*  4500*  4577*  4632*  4737*  4794*
                    4886*  4930*  5019*  5104*  5181*  5258*  5333*  5411*  5489*  5566*  5643*  5721*  5799*
                    5877*  5955*  6031*  6109*  6193*  6278*  6362*  6446*  6530*  6614*  6676*  6754*  6839*
                    6912*  6961*  7084*  7091   7094*  7104
$TKB    001146      1147#  7594   7605   7622   7676   7682
$TKS    001144      1146#  7594   7603   7619   7643*  7674   7680
$TMPO   001160      1154#  2434*  2436   2438   2445*  2447   2450   2462*  2464   2466   2468*  2469*  2470*
                    2471*  2472*  2474   2590*  2591*  2592   2704*  2705*  2706   2773*  2774*  2775   2830*
                    2831*  2832   2893*  2894*  2895   2956*  2957*  2958   3126*  3127*  3128   3189*  3190*
                    3191   3257*  3258*  3259   3329*  3330*  3331   3387*  3388*  3447*  3448*  3507*  3508*
                    3567*  3568*  3627*  3628*  3687*  3688*  3754*  3755*  3821*  3822*  3888*  3889*  3955*
                    3956*  4022*  4023*  4082*  4083*  4149*  4150*  4397*  4398*  4399   4404*  4405*  4406
                    4457*  4458*  4459   4464*  4465*  4466   4536*  4537*  4538   4543*  4544*  4545
$TMP1   001162      1155#
$TMP2   001164      1156#
$TMP3   001166      1157#
$TMP4   001170      1158#
$TMP5   001172      1159#
$TMP6   001174      1160#
$TMP7   001176      1161#
$TN   = 000101       800#   810   2485   2496#  2509   2525   2547   2550   2562#  2608   2618#  2640   2646
                    2653   2657   2669#  2722   2733#  2791   2802#  2849   2854   2865#  2912   2917   2928#
                    2975   2980   2990#  3012   3018   3025   3029   3039#  3061   3067   3074   3078   3089#
                    3144   3156#  3204   3209   3220#  3278   3290#  3342   3360#  3384   3391   3397   3400
                    3413#  3437   3444   3451   3457   3460   3473#  3497   3504   3511   3517   3520   3533#
                    3557   3564   3571   3577   3580   3593#  3617   3624   3631   3637   3640   3653#  3677
                    3684   3691   3697   3700   3711#  3738   3744   3751   3758   3764   3767   3778#  3805
                    3811   3818   3825   3831   3834   3845#  3872   3878   3885   3892   3898   3901   3912#
                    3939   3945   3952   3959   3965   3968   3979#  4006   4012   4019   4026   4032   4035
                    4046#  4073   4079   4086   4092   4095   4106#  4133   4139   4146   4153   4159   4165
                    4177#  4199   4204   4220   4232#  4255   4260   4276   4287#  4335   4340   4346   4349
                    4362#  4422   4434#  4479   4487   4500#  4564   4577#  4619   4632#  4723   4726   4737#
                    4778   4794#  4875   4886#  4919   4930#  4947   4963   4968   4973   5001   5006   5019#
                    5089   5092   5104#  5166   5169   5181#  5243   5246   5258#  5318   5321   5333#  5395
                    5398   5411#  5473   5476   5489#  5551   5554   5566#  5628   5631   5643#  5705   5708
                    5721#  5783   5786   5799#  5861   5864   5877#  5939   5942   5955#  6015   6018   6031#
                    6093   6096   6109#  6177   6180   6193#  6261   6264   6278#  6346   6349   6362#  6430
                    6433   6446#  6514   6517   6530#  6598   6601   6614#  6664   6676#  6737   6740   6754#
                    6819   6822   6839#  6898   6903   6312#  7066   7106
$TPB    001152      1149#  7436*  7447
$TPFLG  001157      1153#  7385   7447
$TPS    001150      1148#  7434   7447
$TRAP   046424      2374   7895#
$TRAP2  046446      7906#  7917
$TRP  = 000016      7910#  7919#  7920#  7921#  7922#  7923#  7924   7925#  7926   7927#  7928#  7929#  7930#
                    7931#  7932#  7933#
$TRPAD  046460      7900   7917#
```

```
$TSTM   001004          1113#
$TSTNM  001102          1126#  6360*  7036   7068*  7095*  7096   7101   7105   7254   7290
$TTYIN  045772          7702   7703   7715   7733   7747   7751#
$TYPBN= ****** U        7923
$TYPOS  044710          7536#  7922
$TYPE   044200          7210   7385#  7910   7918
$TYPEC  044412          7415   7422   7429   7434#  7455   7645
$TYPEX  044460          7440   7442   7445#
$TYPOC  044506          7477#  7919
$TYPON  044522          7476   7479#  7921
$TYPOS  044462          7472#  7920
$UNIT   001226          1179#
$UNITM  001010          1115#
$USWR   001240          1186#
$VECT1  001264          1211#  2440   2449#  2450#  2453   2473#  2474#  2475   2477
$VECT2  001266          1212#
$XTSTR  042524          7050#
$$GFT4= 000000          6988#
$$SW08= 000101          7105#  7106   7107#  7108#  7109#  7110#  7111#  7112#  7113#  7114#  7115#  7116#  7117#
                        7118#  7119#  7120#  7121#  7122#  7123#  7124#  7125#  7126#  7127#  7128#  7129#  7130#
                        7131#  7132#  7133#  7134#  7135#  7136#  7137#  7138#  7139#  7140#  7141#  7142#  7143#
                        7144#  7145#  7146#  7147#  7148#  7149#  7150#  7151#  7152#  7153#  7154#  7155#  7156#
                        7157#  7158#  7159#  7160#  7161#  7162#  7163#  7164#  7165#  7166#  7167#  7168#  7169#
                        7170#
$OFILL  044705          7473#  7477#  7487   7522#
$40CAT= ****** U        7047   7264
.       = 061233        1072#  1076#  1082#  1088   1089#  1091#  1093#  1094#  1100   1101#  1103#  1105#  1123#
                        1168   2367   2382   2383   6973#  6996   6997#  7104   7105   7231#  7290   7447   7590#
                        7594   7751#  7752   7758   7811   7865
.$ASTA= ****** U        7183   7186
.$X     = 001000        1100#  1105
```

```
CLRPSW  1117#  4801  4836
COMMEN   932#
CYLWRT  1117#  3361  3414  3474  3534  3594  3654
ENDCOM   932#
ERROR    826#  2508  2524  2529  2533  2536  2539  2542  2545  2548  2586  2594  2599  2602  2638
         2645  2651  2655  2700  2708  2713  2716  2763  2769  2777  2782  2785  2826  2836  2842
         2846  2889  2899  2905  2909  2952  2962  2968  2972  3010  3017  3023  3027  3059  3065
         3072  3076  3117  3123  3130  3135  3138  3180  3186  3193  3198  3201  3248  3254  3261
         3266  3269  3325  3333  3338  3341  3382  3390  3395  3398  3435  3443  3450  3455  3458
         3495  3503  3510  3515  3518  3555  3563  3570  3575  3578  3615  3623  3630  3635  3638
         3675  3683  3690  3695  3698  3736  3742  3750  3757  3762  3765  3803  3809  3817  3824
         3829  3832  3870  3876  3884  3891  3896  3899  3937  3943  3951  3958  3963  3966  4004
         4010  4018  4025  4030  4033  4071  4077  4085  4090  4093  4131  4137  4145  4152  4157
         4160  4198  4203  4254  4259  4304  4309  4314  4317  4334  4339  4344  4347  4401  4408
         4413  4416  4461  4468  4473  4476  4540  4547  4552  4555  4604  4607  4610  4613  4652
         4679  4684  4689  4710  4713  4716  4719  4763  4766  4769  4772  4816  4828  4831  4834
         4859  4863  4867  4907  4916  4946  4962  4967  4972  4993  4996  4999  5002  5039  5052
         5055  5058  5061  5064  5081  5084  5087  5090  5132  5135  5138  5141  5158  5161  5164
         5167  5209  5212  5215  5218  5235  5238  5241  5244  5284  5287  5290  5293  5310  5313
         5316  5319  5361  5364  5367  5370  5387  5390  5393  5396  5439  5442  5445  5448  5465
         5468  5471  5474  5517  5520  5523  5526  5543  5546  5549  5552  5594  5597  5600  5603
         5620  5623  5626  5629  5671  5674  5677  5680  5697  5700  5703  5706  5749  5752  5755
         5758  5775  5778  5781  5784  5827  5830  5833  5836  5853  5856  5859  5862  5905  5908
         5911  5914  5931  5934  5937  5940  5981  5984  5987  5990  6007  6010  6013  6016  6059
         6062  6065  6068  6085  6088  6091  6094  6143  6146  6149  6152  6169  6172  6175  6178
         6227  6230  6233  6236  6253  6256  6259  6262  6312  6315  6318  6321  6338  6341  6344
         6347  6396  6399  6402  6405  6422  6425  6428  6431  6480  6483  6486  6489  6506  6509
         6512  6515  6564  6567  6570  6573  6590  6593  6596  6599  6649  6652  6655  6658  6704
         6707  6710  6713  6729  6732  6735  6738  6786  6789  6792  6795  6811  6814  6817  6820
         6865  6868  6871  6874  6890  6893  6896  6899  6929  6932  6940  6943  7022
ESCAPE   932#
FORERR  1117#  5105  5182  5259  5334  5412  5490  5567  5644  5722  5800  5878  5956  6032
GETPRI   932#
GETSWR   932#  2413#
IDAETT  1117#  6110  6194  6279  6363  6447  6531
LDLPER  1117#  2566  2673  2737  2807  2870  2933  3093  3161  3224  3295  4365  4441  4503  4580
        4635  4740  6617  6914
MESCMD  1117#  2619  2991  3040
MESSFT  1117#  4178  4233
MSCMD1  1117#  2803  2866  2929
MSG     2485#  2487  2550#  2552  2608#  2610  2657#  2659  2722#  2724  2791#  2793  2854#  2856  2917#
        2919  2980#  2982  3029#  3031  3078#  3080  3144#  3146  3209#  3211  3278#  3280  3347#  3349
        3400#  3402  3460#  3462  3520#  3522  3580#  3582  3640#  3642  3700#  3702  3767#  3769  3834#
        3836  3901#  3903  3968#  3970  4035#  4037  4095#  4097  4165#  4167  4220#  4222  4276#  4278
        4349#  4351  4422#  4424  4487#  4489  4564#  4566  4619#  4621  4726#  4728  4778#  4780  4875#
        4877  4919#  4921  5006#  5009  5092#  5094  5169#  5171  5246#  5248  5321#  5323  5398#  5400
        5476#  5478  5554#  5556  5631#  5633  5708#  5710  5786#  5788  5864#  5866  5942#  5944  6018#
        6020  6096#  6098  6180#  6182  6264#  6266  6349#  6351  6433#  6435  6517#  6519  6601#  6603
        6664#  6666  6740#  6742  6822#  6824  6903#  6905
MSGSEL  1117#  3712  3779  3846  3913  3980  4047  4107
MULT     932#
NEWTST   932#  2485  2550  2608  2657  2722  2791  2854  2917  2980  3029  3078  3144  3209  3278
        3347  3400  3460  3520  3580  3640  3700  3767  3834  3901  3968  4035  4095  4165  4220
        4276  4349  4422  4487  4564  4619  4726  4778  4875  4919  5006  5092  5169  5246  5321
        5398  5476  5554  5631  5708  5786  5864  5942  6018  6096  6180  6264  6349  6433  6517
        6601  6664  6740  6822  6903
```

```
PARGEN  1117#  4363  4501
POP      932#  7225  7226  7577  7800  7849
PUSH     932#  7186  7188  7209  7536  7774  7829
REPORT   932#
SCOPE    827#  2495  2561  2617  2668  2732  2801  2864  2927  2989  3038  3088  3155  3219  3289
         3359  3412  3472  3532  3592  3652  3710  3777  3844  3911  3978  4045  4105  4176  4231
         4286  4361  4433  4499  4576  4631  4736  4793  4885  4929  5018  5103  5180  5257  5332
         5410  5488  5565  5642  5720  5798  5876  5954  6030  6108  6192  6277  6361  6445  6529
         6613  6675  6753  6838  6911  6959
SETPRI   932#
SETTRA  7910#  7919  7920  7921  7922  7924  7926  7927  7928  7929  7930  7931  7932
SETUP    932#  2361
SKIP     932#  2509  2525  2547  2640  2646  2653  2849  2912  2975  3012  3018  3025  3061  3067
         3074  3204  3384  3391  3397  3437  3444  3451  3457  3497  3504  3511  3517  3557  3564
         3571  3577  3617  3624  3631  3637  3677  3684  3691  3697  3738  3744  3751  3758  3764
         3805  3811  3818  3825  3831  3872  3878  3885  3892  3898  3939  3945  3952  3959  3965
         4006  4012  4019  4026  4032  4073  4079  4086  4092  4133  4139  4146  4153  4159  4199
         4204  4255  4260  4335  4340  4346  4479  4723  4947  4963  4968  4973  5001  5089  5166
         5243  5318  5395  5473  5551  5628  5705  5783  5861  5939  6015  6093  6177  6261  6346
         6430  6514  6598  6737  6819  6898
SLASH    932#
SPACE    932#
STARS    932#  1086  1037  1099  1106  1119  1168  1171  2485  2494  2550  2560  2608  2616  2657
         2667  2722  2731  2791  2800  2854  2863  2917  2926  2980  2988  3029  3037  3078  3087
         3144  3154  3209  3218  3278  3288  3347  3358  3400  3411  3460  3471  3520  3531  3580
         3591  3640  3651  3700  3709  3767  3776  3834  3843  3901  3910  3968  3977  4035  4044
         4095  4104  4165  4175  4220  4230  4276  4285  4349  4360  4422  4432  4487  4498  4564
         4575  4619  4630  4726  4735  4778  4792  4875  4884  4919  4928  5006  5017  5092  5102
         5169  5179  5246  5256  5321  5331  5398  5409  5476  5487  5554  5564  5631  5641  5708
         5719  5786  5797  5864  5875  5942  5953  6018  6029  6096  6107  6180  6191  6264  6276
         6349  6360  6433  6444  6517  6528  6601  6612  6664  6674  6740  6752  6822  6837  6903
         6910  6951  7033  7170  7181  7238  7291  7300  7370  7449  7526  7593  7596  7664  7693
         7760  7813  7859  7867  7889
SWRSU    932#  2384#
TPMTRP  7910#
TYPBIN   932#
TYPDEC   932#  6974  6981
TYPNAM   932#  2406
TYPNUM   932#
TYPOCS   932#
TYPOCT   932#  2430  7614
TYPTXT   932#  6970  6977
$$CMRE  1117#
$$CMTM  1117#  1154  1155  1156  1157  1158  1159  1160  1161
$$ESCA   932#
$$NEWT   932#  2485  2550  2608  2657  2722  2791  2854  2917  2980  3029  3078  3144  3209  3278
         3347  3400  3460  3520  3580  3640  3700  3767  3834  3901  3968  4035  4095  4165  4220
         4276  4349  4422  4487  4564  4619  4726  4778  4875  4919  5006  5092  5169  5246  5321
         5398  5476  5554  5631  5708  5786  5864  5942  6018  6096  6180  6264  6349  6433  6517
         6601  6664  6740  6822  6903
$$SET   7910#  7919  7920  7921  7922  7924  7926  7927  7928  7929  7930  7931  7932
$$SETM  2400#
$$SKIP   932#  2509  2525  2547  2640  2646  2653  2849  2912  2975  3012  3018  3025  3061  3067
         3074  3204  3384  3391  3397  3437  3444  3451  3457  3497  3504  3511  3517  3557  3564
         3571  3577  3617  3624  3631  3637  3677  3684  3691  3697  3738  3744  3751  3758  3764
         3805  3811  3818  3825  3831  3872  3878  3885  3892  3898  3939  3945  3952  3959  3965
```

```
          4006    4012    4019    4026    4032    4073    4079    4086    4092    4133    4139    4146    4153    4159    4199
          4204    4255    4260    4335    4340    4346    4479    4723    4947    4963    4968    4973    5001    5089    5166
          5243    5318    5395    5473    5551    5628    5705    5783    5861    5939    6015    6093    6177    6261    6346
          6430    6514    6598    6737    6819    6898
.EQUAT     800#     822
.HEADE     800#
.SETUP     800#    2348
.SWRHI     800#     810
.SWRLO     800#     822#
.$ACTI     800#    1084
.$APTB    1169#
.$APTH     800#    1095
.$APTY     800#    7179
.$CATC     800#    1070
.$CMTA     800#    1117
.$EOP      800#    6949
.$ERRO     800#    7236
.$ERRT     800#
.$POWE     800#
.$RDOC     800#    7758
.$READ     800#    7591
.$SAVE     800#    7811
.$SCOP     800#    7031
.$TRAP     800#    7887
.$TYPD     800#    7524
.$TYPE     800#    7368
.$TYPO     800#    7447

. ABS.   061233      000

 ERRORS DETECTED:  0

 RM03:CZR6BC,RM03:CZR6BC.SEQ/SOL/CRF/NL:TOC/DOC=RM03:CZR6BC.P11
 RUN-TIME:  33 29 3 SECONDS
 RUN-TIME RATIO:  1288/65=19.5
 CORE USED:  30K  (59 PAGES)

 DOCUMENT PAGES:  189
```