

**MADE IN USA**



BORTZRBBCSEQ PAGE 1 00010000 780223 PDP10E010001 CZR6BCO RK611 DSKLS CTRL PRT2 MACY11 30(1046) 02-D  
 CZR6BC.P11 02-DEC-77 09:22

### IDENTIFICATION

PRODUCT CODE: AC-9100C-MC  
PRODUCT NAME: CZR6BCD RK611 DSKLS CTRL PRT 2  
DATE: FEB 1978  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHOR: ROY SPITZER

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A  
 LICENCE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE  
 TERMS OF SUCH LICENSE.

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

COPYRIGHT (C) 1976, 1978 BY DIGITAL EQUIPMENT CORPORATION

CO1

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 2

SEQ 0002

## TABLE OF CONTENTS

- 1.0 ABSTRACT
- 2.0 REQUIREMENTS
  - 2.1 EQUIPMENT
  - 2.2 PRELIMINARY PROGRAMS
- 3.0 OPERATING PROGRAMS
  - 3.1 LOADING PROCEDURE
  - 3.2 STARTING PROCEDURE
  - 3.3 OPTIONAL SWITCH SETTING
  - 3.4 RUN TIME
- 4.0 OPERATING PROCEDURES
- 5.0 PROGRAM DESCRIPTION
- 6.0 ERROR REPORTING

[illegible]

## 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  
CZR6ACD

## 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

52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107



FIRST PASS 7 SECONDS  
SUBSEQUENT PASSES 2 MINUTES

## 4.0 OPERATING PROCEDURES

THE PROGRAM IS EXECUTED BY STARTING AT THE APPROPRIATE 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 CSI 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

108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163

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 WITH A DRIVE CLEAR. CLOCK MESSAGE A AND B INTO SHIFT REGISTER. 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 WITH 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 WITH 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 WITH A RECALIBRATE. CLOCK MESSAGES A AND B INTO SHIFT REGISTER. 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 WITH A START SPINDLE. CLOCK MESSAGES A AND B INTO SHIFT REGISTER. 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.

164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219



220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275

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.

276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331



332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387

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  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
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  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000

#### 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 WITH A SELECT COMMAND. CLOCK ALL 16 BITS THROUGH THE DRIVE MESSAGE LOOPBACK. VERIFY PARITY HAS BEEN PRECONDITIONED 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 1 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 1 WITH A 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



444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499

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  
AVAILABLE 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 AVAILABLE, 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 AVAILABLE  
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

500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555



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.

556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611

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 AVAILABLE 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 AVAILABLE 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.

612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667

668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723

## 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 (PART 4)

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 DIAGNOSTIC 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

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 16

SEW 0016

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	WWWWW	AAAAAA

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

%

780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798



```

799      ; *** REV 003 ***
800      ; TITLE CZR6BCD RK611 DSKLS CTRL PRT2
801      ; *COPYRIGHT (C) 1976, 1977
802      ; *DIGITAL EQUIPMENT CORP.
803      ; *MAYNARD, MASS. 01754
804      ; *
805      ; *PROGRAM BY ROY SPITZER
806      ; *
807      ; *THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
808      ; *PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
809      ; *
810      ; SBTTL OPERATIONAL SWITCH SETTINGS
811      ; *
812      ; *      SWITCH      USE
813      ; *      -----
814      ; *      15      HALT ON ERROR
815      ; *      14      LOOP ON TEST
816      ; *      13      INHIBIT ERROR TYPEOUTS
817      ; *      12      ABORT PROGRAM AFTER 20 ERRORS
818      ; *      11      INHIBIT ITERATIONS
819      ; *      10      BELL ON ERROR
820      ; *      9       LOOP ON ERROR
821      ; *      8       LOOP ON TEST IN SWR<7:0>
822      ; SBTTL BASIC DEFINITIONS
823      ; *
824      ; *INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
825      ; STACK= 1100
826      ; EQUIV EMT,ERROR      ;;BASIC DEFINITION OF ERROR CALL
827      ; EQUIV IOT,SCOPE      ;;BASIC DEFINITION OF SCOPE CALL
828      ; *
829      ; *MISCELLANEOUS DEFINITIONS
830      ; *
831      ; *      11      CODE FOR HORIZONTAL TAB
832      ; *      12      CODE FOR LINE FEED
833      ; *      15      CODE FOR CARRIAGE RETURN
834      ; *      200     CODE FOR CARRIAGE RETURN-LINE FEED
835      ; *      177776  PROCESSOR STATUS WORD
836      ; *
837      ; *      177774  STACK LIMIT REGISTER
838      ; *      177772  PROGRAM INTERRUPT REQUEST REGISTER
839      ; *      177570  HARDWARE SWITCH REGISTER
840      ; *      177570  HARDWARE DISPLAY REGISTER
841      ; *
842      ; *GENERAL PURPOSE REGISTER DEFINITIONS
843      ; *
844      ; *      R0= %0      ;;GENERAL REGISTER
845      ; *      R1= %1      ;;GENERAL REGISTER
846      ; *      R2= %2      ;;GENERAL REGISTER
847      ; *      R3= %3      ;;GENERAL REGISTER
848      ; *      R4= %4      ;;GENERAL REGISTER
849      ; *      R5= %5      ;;GENERAL REGISTER
850      ; *      R6= %6      ;;GENERAL REGISTER
851      ; *      R7= %7      ;;GENERAL REGISTER
852      ; *      SP= %6      ;;STACK POINTER
853      ; *      PC= %7      ;;PROGRAM COUNTER
854      ; *
855      ; *PRIORITY LEVEL DEFINITIONS
856      ; *
857      ; *      PRO= 0      ;;PRIORITY LEVEL 0

```

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 18  
BASIC DEFINITIONS

SEQ 0018

855	000040	PR1=	40	:: PRIORITY LEVEL 1
856	000100	PR2=	100	:: PRIORITY LEVEL 2
857	000140	PR3=	140	:: PRIORITY LEVEL 3
858	000200	PR4=	200	:: PRIORITY LEVEL 4
859	000240	PR5=	240	:: PRIORITY LEVEL 5
860	000300	PR6=	300	:: PRIORITY LEVEL 6
861	000340	PR7=	340	:: PRIORITY LEVEL 7
862		: * "SWITCH REGISTER" SWITCH DEFINITIONS		
863		SW15=	100000	
864	100000	SW14=	40000	
865	040000	SW13=	20000	
866	020000	SW12=	10000	
867	010000	SW11=	4000	
868	004000	SW10=	2000	
869	002000	SW09=	1000	
870	001000	SW08=	400	
871	000400	SW07=	200	
872	000200	SW06=	100	
873	000100	SW05=	40	
874	000040	SW04=	20	
875	000020	SW03=	10	
876	000010	SW02=	4	
877	000004	SW01=	2	
878	000002	SW00=	1	
879	000001	. EQUIV	SW09, SW9	
880		. EQUIV	SW08, SW8	
881		. EQUIV	SW07, SW7	
882		. EQUIV	SW06, SW6	
883		. EQUIV	SW05, SW5	
884		. EQUIV	SW04, SW4	
885		. EQUIV	SW03, SW3	
886		. EQUIV	SW02, SW2	
887		. EQUIV	SW01, SW1	
888		. EQUIV	SW00, SW0	
889				
890		: * DATA BIT DEFINITIONS (BIT00 TO BIT15)		
891		BIT15=	100000	
892	100000	BIT14=	40000	
893	040000	BIT13=	20000	
894	020000	BIT12=	10000	
895	010000	BIT11=	4000	
896	004000	BIT10=	2000	
897	002000	BIT09=	1000	
898	001000	BIT08=	400	
899	000400	BIT07=	200	
900	000200	BIT06=	100	
901	000100	BIT05=	40	
902	000040	BIT04=	20	
903	000020	BIT03=	10	
904	000010	BIT02=	4	
905	000004	BIT01=	2	
906	000002	BIT00=	1	
907	000001	. EQUIV	BIT09, BIT9	
908		. EQUIV	BIT08, BIT8	
909		. EQUIV	BIT07, BIT7	
910				

```

911      .EQUIV BIT06,BIT6
912      .EQUIV BIT05,BIT5
913      .EQUIV BIT04,BIT4
914      .EQUIV BIT03,BIT3
915      .EQUIV BIT02,BIT2
916      .EQUIV BIT01,BIT1
917      .EQUIV BIT00,BIT0
918
919      .*BASIC "CPU" TRAP VECTOR ADDRESSES
920      ERRVEC= 4      ; TIME OUT AND OTHER ERRORS
921      RESVEC= 10     ; RESERVED AND ILLEGAL INSTRUCTIONS
922      TBITVEC= 14    ; "T" BIT
923      TRIVEC= 14     ; TRACE TRAP
924      BPTVEC= 14     ; BREAKPOINT TRAP (BPT)
925      IOTVEC= 20     ; INPUT/OUTPUT TRAP (IOT) **SCOPE**
926      PWRVEC= 24     ; POWER FAIL
927      EMTVEC= 30     ; EMULATOR TRAP (EMT) **ERROR**
928      TRAPVEC= 34    ; "TRAP" TRAP
929      TKVEC= 60      ; TTY KEYBOARD VECTOR
930      TPVEC= 64      ; TTY PRINTER VECTOR
931      PIRQVEC= 240   ; PROGRAM INTERRUPT REQUEST VECTOR
932      MEMVEC= 114    ; VECTOR FOR MEMORY CHECK ENABLE
933      MEMBAS= 172100 ; BUS ADDRESS FOR MEMORY CHECK ENABLE
934      PAR.EN= 1      ; MEMORY ENABLE PARITY CHECKING
935      AVECT1= 120210 ; DEFINE RK611 VECTOR ADDRESS
936      APRIOR= 5      ; DEFINE RK611 PRIORITY
937      ABASE= 177440  ; DEFINE BASE OF RK611 REGISTERS
938
939      .SBTTL RK611 CONTROLLER REGISTER DEFINITION
940
941      RKCS1= 0        ; CONTROL AND STATUS REGISTER 1
942      RKWC= 2         ; WORD COUNT REGISTER
943      RKBA= 4         ; BUS ADDRESS REGISTER
944      RKDA= 6         ; DESIRED TRACK SECTOR REGISTER
945      RKCS2= 10      ; CONTROL AND STATUS REGISTER 2
946      RKDS= 12       ; DRIVE STATUS REGISTER
947      RKER= 14       ; ERROR REGISTER
948      RKASOF= 16     ; ATTENTION SUMMARY AND OFFSET REGISTER
949      RKDCYL= 20      ; DESIRED CYLINDER REGISTER
950      RKDB= 24       ; DATA BUFFER
951      RKMR1= 26      ; MAINTENANCE REGISTER 1
952      RKMR2= 34      ; MAINTENANCE REGISTER 2
953      RKMR3= 36      ; MAINTENANCE REGISTER 3
954      RKECPS= 30     ; ECC POSITION INFORMATION
955      RKECPT= 32     ; ECC PATTERN INFORMATION
956      RKSPAR= 22     ; SPARE REGISTER
957
958      .SBTTL DRIVE COMMANDS
959
960      SELDRV= 01      ; SELECT DRIVE
961      PACK= 03       ; PACK ACKNOWLEDGE
962      CLEAR= 05      ; DRIVE CLEAR
963      UNLOAD= 07     ; UNLOAD
964      SRTSPL= 11     ; START SPINDLE
965      RECAL= 13      ; RECALIBRATE
966      OFFSET= 15     ; OFFSET

```

967	000017	SEEK= 17	; SEEK
968	000021	RODATA= 21	; READ DATA
969	000023	WRDATA= 23	; WRITE DATA
970	000025	RDHEAD= 25	; READ HEADER
971	000027	WRHEAD= 27	; WRITE HEADER AND DATA
972	000031	WRTCHK= 31	; WRITE CHECK
973	000300	INTR= 300	; GENERATE INTERRUPT TO CPU
974			
975		.SBTTL CONTROL AND STATUS REGISTER 1 BITS	
976			
977	000001	GO= BIT0	; GO BIT
978	000100	IE= BIT6	; INTERRUPT ENABLE
979	000200	ROY= BIT7	; CONTROLLER READY
980	000400	BA16= BIT8	; BUS ADDRESS BIT 16
981	001000	BA17= BIT9	; BUS ADDRESS BIT 17
982	002000	CDT= BIT10	; CONTROLLER DRIVE TYPE (0=RK06)
983	004000	CTO= BIT11	; CONTROLLER TIMED OUT WAITING FOR
984			; DRIVE RESPONSE
985	010000	CFMT= BIT12	; CONTROLLER DRIVE FORMAT (0=26 SECTOR, 1=24 SECTOR)
986	020000	SPAR= BIT13	; DRIVE BUS PARITY ERROR? DETECTED BY CONTROLLER
987	040000	DI= BIT14	; DRIVE INTERRUPT
988	100000	CERR= BIT15	; CONTROLLER ERROR
989	100000	CCLR= BIT15	; CONTROLLER CLEAR
990			
991		.SBTTL CONTROL AND STATUS REGISTER 2 BITS	
992			
993	000007	DRVMSK= 7	; MASK FOR DRIVE SELECTION CODE
994	000010	RLS= BIT3	; DESELECT OR RELEASE DRIVE IN BITS 0-2
995	000020	BAI= BIT4	; BUS ADDRESS INCREMENT INHIBIT
996	000040	SCLR= BIT5	; CLEAR CONTROLLER AND ALL DRIVES
997	000100	IR= BIT6	; INPUT READY
998	000200	OR= BIT7	; OUTPUT READY
999	000400	UFE= BIT8	; UNIT FIELD ERROR
1000	001000	MDS= BIT9	; MULTIPLE DRIVE SELECT
1001	002000	PGE= BIT10	; PROGRAMMING ERROR
1002	004000	NEM= BIT11	; NON-EXISTENT MEMORY
1003	010000	NED= BIT12	; NON-EXISTENT DRIVE
1004	020000	UPE= BIT13	; UNIBUS PARITY ERROR
1005	040000	WCE= BIT14	; WRITE CHECK ERROR
1006	100000	DLT= BIT15	; DATA LATE ERROR
1007			
1008		.SBTTL ERROR REGISTER BIT DEFINITION	
1009			
1010	000001	ILF= BIT0	; ILLEGAL FUNCTION CODE
1011	000002	SKI= BIT1	; SEEK INCOMPLETE
1012	000004	NXF= BIT2	; NON-EXECUTABLE DRIVE FUNCTION
1013	000010	DRPAR= BIT3	; DRIVE DETECTED DRIVE BUS PARITY ERROR
1014	000020	FMTE= BIT4	; FORMAT ERROR
1015	000040	DTYE= BIT5	; DRIVE TYPE ERROR
1016	000100	ECH= BIT6	; ECC HARD
1017	000200	BSE= BIT7	; BAD SECTOR ERROR
1018	000400	HVRC= BIT8	; HEADER VRC ERROR
1019	001000	COE= BIT9	; CYLINDER ADDRESS OVERFLOW ERROR
1020	002000	IOAE= BIT10	; INVALID DISK ADDRESS ERROR
1021	004000	WLE= BIT11	; WRITE LOCK ERROR
1022	010000	DTE= BIT12	; DRIVE TIMING ERROR

1023	020000	OPI=	BIT13	; OPERATION (SEARCH) INCOMPLETE
1024	040000	UNS=	BIT14	; DRIVE UNSAFE
1025	100000	DCK=	BIT15	; DATA CHECK
1026		.SBTTL STATUS REGISTER BIT DEFINITION		
1027				
1028				
1029	000001	DRA=	BIT0	; DRIVE AVAILABLE (CONTROLLER IS SET IF ; THIS BIT IS RESET)
1030				
1031	000004	OFST=	BIT2	; DRIVE OFFSET
1032	000010	ACLO=	BIT3	; AC LOW
1033	000020	SPDLSS=	BIT4	; SPEED LOSS
1034	000040	DROT=	BIT5	; DRIVE OFF TRACK
1035	000100	VV=	BIT6	; VOLUME VALID
1036	000200	DRDY=	BIT7	; DRIVE READY
1037	000400	DDT=	BIT8	; DRIVE TYPE (0=RK06)
1038	004000	WRL=	BIT11	; WRITE LOCK
1039	020000	PIP=	BIT13	; POSITIONING IN PROGRESS
1040	040000	DSC=	BIT14	; DRIVE STATUS CHANGE
1041	100000	SVAL=	BIT15	; STATUS VALID
1042		.SBTTL MAINTENANCE REGISTER 1 BIT DEFINITION		
1043				
1044				
1045	000017	MESMSK=	17	; MESSAGE MASK
1046				
1047	000020	PAT=	BIT4	; FORCE EVEN PARITY ON DRIVE MESSAGE LINES
1048	000040	DMD=	BIT5	; DIAGNOSTIC MODE
1049	000100	MSP=	BIT6	; MAINTENANCE SECTOR PULSE
1050	000200	MIND=	BIT7	; MAINTENANCE INDEX
1051	000400	MCLK=	BIT8	; MAINTENANCE CLOCK
1052	001000	MERD=	BIT9	; MAINTENANCE ENCODED READ DATA
1053	002000	MEWD=	BIT10	; MAINTENANCE ENCODED WRITE DATA
1054	004000	PCA=	BIT11	; PRECOMPENSATION ADVANCE
1055	010000	PCD=	BIT12	; PRECOMPENSATION DELAY
1056	020000	ECCW=	BIT13	; ECC WORD IS BEING READ OR WRITTEN
1057	040000	WRTGAT=	BIT14	; WRITE GATE
1058	100000	RDGATE=	BIT15	; READ GATE
1059		.SBTTL TRANSMITTED MESSAGE A		
1060				
1061				
1062	000020	S.SEEK=	BIT4	; SEEK COMMAND
1063	000040	S.RECL=	BIT5	; RECALIBRATE COMMAND
1064	000100	S.STSP=	BIT6	; START SPINDLE COMMAND
1065	000200	S.RTC=	BIT7	; DRIVE RETURN TO CENTERLINE COMMAND
1066	000400	S.CLR=	BIT8	; CLEAR ERROR AND DSC
1067	001000	S.FMT=	BIT9	; FORMAT
1068	002000	S.UNLD=	BIT10	; UNLOAD
1069	004000	S.PACK=	BIT11	; SET VOLUME VALID (PACK ACKNOWLEDGE)
1070		.SBTTL TRAP CATCHER		
1071				
1072	000000	;=0		
1073		; *ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"		
1074		; *SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS		
1075		; *LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS		
1076	000174	;=174		
1077	000174 000000	DISPREG:	.WORD 0	; SOFTWARE DISPLAY REGISTER
1078	000176 000000	SWREG:	.WORD 0	; SOFTWARE SWITCH REGISTER



J02

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 22  
STARTING ADDRESS(ES)

SEQ 0022

```

1079
1080 000200 000137 004316
1081 000204 000137 004306
1082
1083 000214 000137 004276
1084
1085
1086
1087
1088 000220
1089 000046 000046
1090 000046 042340
1091 000052 000052
1092 000052 000000
1093 000052 000220
1094 000052 001000
1095
1096
1097
1098
1099
1100 001000
1101 000024 000024
1102 000024 000200
1103 000044 000044
1104 000044 001000
1105 000044 001000
1106
1107
1108
1109
1110 001000
1111 001000 000000
1112 001002 001214
1113 001004 000001
1114 001006 000007
1115 001010 000007
1116 001012 000032

```

```

.SBTTL STARTING ADDRESS(ES)
JMP 2*START ;;JUMP TO STARTING ADDRESS OF PROGRAM
JMP RESTRT ;;JUMP TO RESTART ROUTINE
.=214
JMP PARM ;;JUMP TO OPERATOR ASSIGNED PARMETERS
.SBTTL ACT11 HOOKS
;*****
;HOOKS REQUIRED BY ACT11
$SVPC=.;SAVE PC
.=46
$ENDAD ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .SEOP
.=52
;WORD 0 ;;2)SET LOC.52 TO ZERO
.=1000
$SVPC ;;RESTORE PC
.SBTTL APT PARAMETER BLOCK
;*****
;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
;*****
.$X=.;SAVE CURRENT LOCATION
.=24 ;;SET POWER FAIL TO POINT TO START OF PROGRAM
200 ;;FOR APT START UP
.=44 ;;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR ;;POINT TO APT HEADER BLOCK
.=.$X ;;RESET LOCATION COUNTER
;*****
;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
;INTERFACE SPEC.
$APTHD:
$HIBTS: .WORD 0 ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBADR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
$TSTM: .WORD 1 ;;RUN TIM OF LONGEST TEST
$PASTM: .WORD 7 ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD 7 ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
.SBTTL .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)

```

1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124 001100 001100  
1125 001100 000000  
1126 001102 000  
1127 001103 000  
1128 001104 000000  
1129 001106 000000  
1130 001110 000000  
1131 001112 000000  
1132 001114 000  
1133 001115 001  
1134 001116 000000  
1135 001120 000000  
1136 001122 000000  
1137 001124 000000  
1138 001126 000000  
1139 001130 000000  
1140 001132 000000  
1141 001134 000  
1142 001135 000  
1143 001136 000000  
1144 001140 177570  
1145 001142 177570  
1146 001144 177560  
1147 001146 177562  
1148 001150 177564  
1149 001152 177566  
1150 001154 000  
1151 001155 002  
1152 001156 012  
1153 001157 000  
1154 001160 000000  
1155 001162 000000  
1156 001164 000000  
1157 001166 000000  
1158 001170 000000  
1159 001172 000000  
1160 001174 000000  
1161 001176 000000  
1162 001200 000000  
1163 001202 000000  
1164 001204 177607 000377  
1165 001210 077  
1166 001211 015  
1167 001212 000012  
1168  
1169  
1170  
1171  
1172

.SBTTL COMMON TAGS

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

SCMTAG: . =1100

STSTNM: .WORD 0  
SERFLG: .BYTE 0  
SICNT: .WORD 0  
SLPADR: .WORD 0  
SLPERR: .WORD 0  
SERTTL: .WORD 0  
SITEMB: .BYTE 0  
SERMAX: .BYTE 1  
SERRPC: .WORD 0  
SGDADR: .WORD 0  
SBDADR: .WORD 0  
SGDDAT: .WORD 0  
SBDDAT: .WORD 0  
SAUTOB: .BYTE 0  
SINTAG: .BYTE 0  
SWR: .WORD DSWR  
DISPLAY: .WORD DDISP  
STKS: 177560  
STKB: 177562  
STPS: 177564  
STPB: 177566  
SNUL: .BYTE 0  
SFILLS: .BYTE 2  
SFILLC: .BYTE 12  
STPFLG: .BYTE 0  
STMP0: .WORD 0  
STMP1: .WORD 0  
STMP2: .WORD 0  
STMP3: .WORD 0  
STMP4: .WORD 0  
STMP5: .WORD 0  
STMP6: .WORD 0  
STMP7: .WORD 0  
STIMES: 0  
SESCAPE: 0  
SBELL: .ASCIZ <207><377><377>  
SQUES: .ASCII /?/  
SCRLF: .ASCII <15>  
SLF: .ASCIZ <12>

;; START OF COMMON TAGS

;; CONTAINS THE TEST NUMBER  
;; CONTAINS ERROR FLAG  
;; CONTAINS SUBTEST ITERATION COUNT  
;; CONTAINS SCOPE LOOP ADDRESS  
;; CONTAINS SCOPE RETURN FOR ERRORS  
;; CONTAINS TOTAL ERRORS DETECTED  
;; CONTAINS ITEM CONTROL BYTE  
;; CONTAINS MAX. ERRORS PER TEST  
;; CONTAINS PC OF LAST ERROR INSTRUCTION  
;; CONTAINS ADDRESS OF 'GOOD' DATA  
;; CONTAINS ADDRESS OF 'BAD' DATA  
;; CONTAINS 'GOOD' DATA  
;; CONTAINS 'BAD' DATA  
;; RESERVED--NOT TO BE USED  
;; AUTOMATIC MODE INDICATOR  
;; INTERRUPT MODE INDICATOR  
;; ADDRESS OF SWITCH REGISTER  
;; ADDRESS OF DISPLAY REGISTER  
;; TTY KBD STATUS  
;; TTY KBD BUFFER  
;; TTY PRINTER STATUS REG. ADDRESS  
;; TTY PRINTER BUFFER REG. ADDRESS  
;; CONTAINS NULL CHARACTER FOR FILLS  
;; CONTAINS # OF FILLER CHARACTERS REQUIRED  
;; INSERT FILL CHARS. AFTER A "LINE FEED"  
;; "TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; USER DEFINED  
;; MAX. NUMBER OF ITERATIONS  
;; ESCAPE ON ERROR ADDRESS  
;; CODE FOR BELL  
;; QUESTION MARK  
;; CARRIAGE RETURN  
;; LINE FEED

.SBTTL APT MAILBOX-ETABLE

\*\*\*\*\*  
; EVEN

1173	001214		\$MAIL:		:: APT MAILBOX
1174	001214	000000	\$MSGTY:	.WORD	AMSGTY
1175	001216	000000	\$FATAL:	.WORD	AFATAL
1176	001220	000000	\$TESTN:	.WORD	ATESTN
1177	001222	000000	\$PASS:	.WORD	APASS
1178	001224	000000	\$DEVCT:	.WORD	ADEVCT
1179	001226	000000	\$UNIT:	.WORD	AUNIT
1180	001230	000000	\$MSGAD:	.WORD	AMSGAD
1181	001232	000000	\$MSGLG:	.WORD	AMSGLG
1182	001234		\$ETABLE:		:: APT ENVIRONMENT TABLE
1183	001234	000	\$ENV:	.BYTE	AENV
1184	001235	000	\$ENVN:	.BYTE	AENVN
1185	001236	000000	\$SWREG:	.WORD	ASWREG
1186	001240	000000	\$USWR:	.WORD	AUSWR
1187	001242	000000	\$CPUOP:	.WORD	ACPUOP
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, PQQ=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
1195	001245	000	\$MTYP1:	.BYTE	AMTYP1
1196			::		MEM. TYPE BLK#1
1197			::		MEM. TYPE BYTE -- (HIGH BYTE)
1198			::		900 NSEC CORE=001
1199			::		300 NSEC BIPOLAR=002
1200	001246	000000	\$MADR1:	.WORD	AMADR1
1201			::		;; HIGH ADDRESS, BLK#1
1202	001250	000	\$MAMS2:	.BYTE	AMAMS2
1203	001251	000	\$MTYP2:	.BYTE	AMTYP2
1204	001252	000000	\$MADR2:	.WORD	AMADR2
1205	001254	000	\$MAMS3:	.BYTE	AMAMS3
1206	001255	000	\$MTYP3:	.BYTE	AMTYP3
1207	001256	000000	\$MADR3:	.WORD	AMADR3
1208	001260	000	\$MAMS4:	.BYTE	AMAMS4
1209	001261	000	\$MTYP4:	.BYTE	AMTYP4
1210	001262	000000	\$MADR4:	.WORD	AMADR4
1211	001264	120210	\$VECT1:	.WORD	AVECT1
1212	001266	000000	\$VECT2:	.WORD	AVECT2
1213	001270	177440	\$BASE:	.WORD	ABASE
1214	001272	000000	\$DEVN:	.WORD	ADEVN
1215	001274	000000	\$CDW1:	.WORD	ACDW1
1216	001276	000000	\$CDW2:	.WORD	ACDW2
1217	001300		\$ETEND:		:: CONTROLLER DESCRIPTION WORD#1
1218			.MEXIT		:: CONTROLLER DESCRIPTION WORD#2

## .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

## \$ERRTB:

; ERROR 1: ATTEMPTING TO SET CMD BIT DRIVE MESS A  
 ; EMIN: 0

0  
 DT001  
 DF001

; ERROR 2: ATTEMPTING A SELECT OF DRIVE NUM - CS1 INCORRECT

EM106  
 EM2003  
 DT002  
 DF002

; ERROR 3: ATTEMPTING A SELECT OF DRIVE NUM - DRIVE NUM INCORRECT

EM106  
 EM2004  
 DT002  
 DF002

; ERROR 4: ATTEMPTING A SELECT OF DRIVE NUM - MESSAGE A INCORRECT

EM106  
 EM2001  
 DT002  
 DF002

; ERROR 5: ATTEMPTING A SELECT OF DRIVE NUM - MESSAGE B INCORRECT

EM106  
 EM2002  
 DT002  
 DF002

; ERROR 6: ATTEMPTING A SELECT WITH HEAD ADD - CS1 INCORRECT

EM107  
 EM2003  
 DT006  
 DF006

; ERROR 7: ATTEMPTING A SELECT WITH HEAD ADD - HEAD INCORRECT

EM107  
 EM2005  
 DT006  
 DF006

; ERROR 10: ATTEMPTING A SELECT WITH HEAD ADD - MESSAGE A INCORRECT

EM107  
 EM2001  
 DT006  
 DF006

; ERROR 11: ATTEMPTING A SELECT WITH HEAD ADD - MESSAGE B INCORRECT

EM107  
 EM2001  
 DT006  
 DF006

1219  
 1220  
 1221  
 1222  
 1223  
 1224  
 1225  
 1226  
 1227  
 1228  
 1229  
 1230  
 1231  
 1232  
 1233 001300  
 1234  
 1235 001300 000000  
 1236 001302 000000  
 1237 001304 046520  
 1238 001306 047154  
 1239  
 1240 001310 052410  
 1241 001312 057141  
 1242 001314 046540  
 1243 001316 047210  
 1244  
 1245 001320 052410  
 1246 001322 057204  
 1247 001324 046540  
 1248 001326 047210  
 1249  
 1250 001330 052410  
 1251 001332 057063  
 1252 001334 046540  
 1253 001336 047210  
 1254  
 1255 001340 052410  
 1256 001342 057112  
 1257 001344 046540  
 1258 001346 047210  
 1259  
 1260 001350 052514  
 1261 001352 057141  
 1262 001354 046562  
 1263 001356 047244  
 1264  
 1265 001360 052514  
 1266 001362 057255  
 1267 001364 046562  
 1268 001366 047244  
 1269  
 1270 001370 052514  
 1271 001372 057063  
 1272 001374 046562  
 1273 001376 047244  
 1274

N02

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 26  
ERROR POINTER TABLE

SEQ 0026

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 - CSI 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 - CSI 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 - CSI 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 RKK07 - 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

C03

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 28  
ERROR POINTER TABLE

SEQ 0028

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

D03

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 29  
ERROR POINTER TABLE

SEQ 0029

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	

E03

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 30  
ERROR POINTER TABLE

SEQ 0030

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 COMPLETE 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 SELECT 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	002210	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	



G03

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 32  
ERROR POINTER TABLE

SEQ 0032

1611	002326	047540	DF062		
1612			ERROR 104:	ATTEMPTING EXECUTION OF DESELECT DRIVE WITH IE SET	
1613			:	INTERRJPT 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		

H03

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 33  
ERROR POINTER TABLE

SEQ 0033

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	

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P.1 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 34  
ERROR POINTER TABLE

SEQ 0034

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

J03

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 35  
ERROR POINTER TABLE

SEQ 0035

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 AVAIVABLE
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 INCORRECT
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	

K03

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 36  
ERROR POINTER TABLE

SEQ 0036

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	

L03

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 37  
ERROR POINTER TABLE

SEQ 0037

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	055442	:	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

## M03

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 38  
ERROR POINTER TABLE

SEQ 0038

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

N03

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 39  
ERROR POINTER TABLE

SEQ 0039

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 FORCE 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 LOW 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	EM137	
2062	003452	EM2025	
2063	003454	DT062	
2064	003456	DF062	
2065	:	ERROR 217:	ATTEMPTING TO FORCE AC LOW AND C-D PARITY ERROR
2066	:		ERROR REG INCORRECT
2067	003460	EM137	
2068	003462	EM2013	
2069	003464	DT062	
2070	003466	DF062	
2071	:	ERROR 220:	ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2072	:		CS1 INCORRECT
2073	003470	EM138	
2074	003472	EM2003	
2075	003474	DT062	
2076	003476	DF062	
2077	:	ERROR 221:	ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2078	:		CS2 INCORRECT
2079	003500	EM138	
2080	003502	EM2012	
2081	003504	DT062	
2082	003506	DF062	
2083	:	ERROR 222:	ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2084	:		DRIVE STATUS REG INCORRECT
2085	003510	EM138	
2086	003512	EM2025	
2087	003514	DT062	
2088	003516	DF062	
2089	:	ERROR 223:	ATTEMPTING TO FORCE ILLEGAL DISK ADDRESS ERROR (DRIVE)
2090	:		ERROR REG INCORRECT
2091	003520	EM138	
2092	003522	EM2013	
2093	003524	DT062	
2094	003526	DF062	
2095	:	ERROR 224:	ATTEMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2096	:		CS1 INCORRECT
2097	003530	EM139	
2098	003532	EM2003	
2099	003534	DT224	
2100	003536	DF224	
2101	:	ERROR 225:	ATTEMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2102	:		CS2 INCORRECT
2103	003540	EM139	
2104	003542	EM2012	
2105	003544	DT224	
2106	003546	DF224	
2107	:	ERROR 226:	ATTEMPTING TO CLEAR RK611 WITH A CONTROLLER CLEAR
2108	:		DRIVE STATUS REG INCORRECT
2109	003550	EM139	
2110	003552	EM2025	
2111	003554	DT224	
2112	003556	DF224	
2113	:	ERROR 227:	ATTEMPTING 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			CSI 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	003624	046760	DT062
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	:	CSI 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 - CSI 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	DM000C
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 REGISTER
2284 004130 000000 T.CS2: .WORD 0 ;CONTROL AND STATUS REGISTER 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

```

G04

C2R6BCD RK611 DSKLS CTRL PRT2  
C2R6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 45  
PROGRAM DEFINED VARIABLES

SEQ 0045

2324			.SBTTL	PROGRAM DEFINED VARIABLES	
2325					
2326	004234	000210	RKVEC:	.WORD	210
2327	004236	000240	RKPRI:	.WORD	PR5
2328	004240	000000	SRTFLG:	.WORD	0
2329					
2330					
2331					
2332	004242	000000	ERRCNT:	.WORD	0
2333	004244	000000	DRVCOD:	.WORD	0
2334	004246	000000	MSGCOD:	.WORD	0
2335	004250	000000	HDCODE:	.WORD	0
2336	004252	000000	CYLIN:	.WORD	0
2337	004254	000000	OFFVAL:	.WORD	0
2338	004256	000000	SFTCNT:	.WORD	0
2339	004260	000000	PARBIT:	.WORD	0
2340	004262	000015	WAITIM:	.WORD	15
2341	004264	000144	STALL:	.WORD	100.
2342	004266	000000	DRVTYP:	.WORD	0
2343	004270	000000	ILLFUN:	.WORD	0
2344	004272	000000	TRAPPC:	.WORD	0
2345	004274	000000	SAVSWR:	.WORD	0

```

:RK611 VECTOR
:RK611 PRIORITY
:START FLAG
: 0 = 200
: 1 = 214
:-1 = 204
:ERROR COUNT FOR SWITCH 12 ABORT
:DRIVE SELECT CODE
:MESSAGE SELECT CODE
:HEAD SELECT CODE
:CYLINDER ADD VALUE
:OFFSET VALUE
:SHIFT COUNT FOR DRIVE MESSAGE SHIFTING
:PARITY BIT FOR SHIFT
:WAITING FOR DESELECT COMMAND
:STALL TIME FOR MESSAGE TIME OUT (NED)
:DRIVE TYPE INDICATOR
:ILLEGAL FUNCTION CODE
:ADDRESS OF TRAP FROM MEMORY CHECK
:SAVED SWITCH REG FOR POWER FAIL

```

```

2346 .SBTTL PROGRAM SETUP
2347
2348 004276 012737 000001 004240 PARM: MOV #1,SRTFLG ;LOAD START FLAG FOR PARMETER START
2349 004304 000406 BR START1
2350
2351 004306 012737 177777 004240 RESTRT: MOV #-1,SRTFLG ;LOAD START FLAG FOR RESTART
2352 004314 000402 BR START1
2353
2354 004316 005037 004240 START: CLR SRTFLG ;CLEAR START FLAG
2355 004322 000005 START1: RESET ;RESET THE WHOLE SYSTEM
2356 004324 012706 001100 MOV #STACK,SP ;INITIALIZE STACK POINTER
2357 004330 012746 000340 MOV #PR7,-(SP) ;LOAD STACK TO LOCK OUT ALL INTERRUPTS
2358 004334 012746 004342 MOV #1$,-(SP) ;LOAD START OF PROGRAM
2359 004340 000002 RTI ;LOAD PSW
2360
2361 004342 1$:
2362 .SBTTL INITIALIZE THE COMMON TAGS
2363 ;;CLEAR THE COMMON TAGS ($CMTAG) AREA
2364 004342 012706 001100 MOV #SCMTAG,R6 ;;FIRST LOCATION TO BE CLEARED
2365 004346 005026 CLR (R6)+ ;;CLEAR MEMORY LOCATION
2366 004350 022706 001140 CMP #SWR,R6 ;;DONE?
2367 004354 001374 BNE -6 ;;LOOP BACK IF NO
2368 004356 012706 001100 MOV #STACK,SP ;;SETUP THE STACK POINTER
2369 ;;INITIALIZE A FEW VECTORS
2370 004362 012737 042512 000020 MOV #SCOPE,2#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
2371 004370 012737 000340 000022 MOV #340,2#IOTVEC+2 ;;LEVEL 7
2372 004376 012737 043516 000030 MOV #ERROR,2#EMTVEC ;;EMI VECTOR FOR ERROR ROUTINE
2373 004404 012737 000340 000032 MOV #340,2#EMTVEC+2 ;;LEVEL 7
2374 004412 012737 046424 000034 MOV #STRAP,2#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
2375 004420 012737 000340 000036 MOV #340,2#TRAPVEC+2 ;;LEVEL 7
2376 004426 012737 046272 000024 MOV #SPWRDN,2#PWRVEC ;;POWER FAILURE VECTOR
2377 004434 012737 000340 000026 MOV #340,2#PWRVEC+2 ;;LEVEL 7
2378 004442 013737 042204 042176 MOV #ENDCT,SEOPCT ;;SETUP END-OF-PROGRAM COUNTER
2379 004450 005037 001200 CLR $TIMES ;;INITIALIZE NUMBER OF ITERATIONS
2380 004454 005037 001202 CLR $ESCAPE ;;CLEAR THE ESCAPE ON ERROR ADDRESS
2381 004460 112737 000001 001115 MOV #1,$ERMAX ;;ALLOW ONE ERROR PER TEST
2382 004466 012737 004466 001106 MOV #.,$LPADR ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
2383 004474 012737 004474 001110 MOV #.,$LPERR ;;SETUP THE ERROR LOOP ADDRESS
2384 ;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
2385 ;;EQUAL TO A "-1" SETUP FOR A SOFTWARE SWITCH REGISTER.
2386 004502 013746 000004 000004 MOV 2#ERRVEC,-(SP) ;;SAVE ERROR VECTOR
2387 004506 012737 004542 000004 MOV #64$,2#ERRVEC ;;SET UP ERROR VECTOR
2388 004514 012737 177570 001140 MOV #DSWR,SWR ;;SETUP FOR A HARDWARE SWICH REGISTER
2389 004522 012737 177570 001142 MOV #DDISP,DISPLAY ;;AND A HARDWARE DISPLAY REGISTER
2390 004530 022777 177777 174402 CMP #-1,2$SWR ;;TRY TO REFERENCE HARDWARE SWR
2391 004536 001012 BNE 66$ ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
2392 ;;AND THE HARDWARE SWR IS NOT = -1
2393 004540 000403 BR 65$ ;;BRANCH IF NO TIMEOUT
2394 004542 012716 004550 64$: MOV #65$, (SP) ;;SET UP FOR TRAF TURN
2395 004546 000002 RTI
2396 004550 012737 000176 001140 65$: MOV #SWREG,SWR ;;POINT TO SOFTWARE SWR
2397 004556 012737 000174 001142 65$: MOV #DISPREG,DISPLAY
2398 004564 012637 000004 66$: MOV (SP)+,2#ERRVEC ;;RESTORE ERROR VECTOR
2399
2400 004570 005037 001222 CLR $PASS ;;CLEAR PASS COUNT
2401 004574 132737 000200 001235 BITB #APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT

```

```

2402 004602 001403      BEQ      F7$      ;;YES,USE NON-APT SWITCH
2403 004604 012737 001236 001140      MOV      #SSWREG,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      #SENDAD,2#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      2#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$:
2424 004700 000426      BR      68$
2425      ;;69$:      .ASCIZ      <CRLF>/RK611 DISKLESS DIAGNOSTIC: PART 2 CZR6BCD/<CRLF>
2426 004756      68$:
2427 004756 022737 000001 004240      CMP      #1,SRTFLG      ;CHECK IF PARAMETER START
2428 004764 001122      BNE      15$      ;NO,CONTINUE SETUP
2429 004766 104401 050010      TYPE      OPRO01      ;TYPE "RK611 BUS ADDRESS ( ) ="
2430 004772 013746 001270      MOV      $BASE,-(SP)      ;;SAVE $BASE FOR TYPEOUT
2431 004776 104402      TYPOC      ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
2432 005000 104401 050037      TYPE      ,OPR002
2433 005004 104412      RDOCT      ;GET VALUE
2434 005006 012637 001160      MOV      (SP)+,$TMPD
2435 005012 001407      BEQ      7$      ;CHECK IF <CR>
2436 005014 022737 160000 001160      CMP      #160000,$TMPD      ;CHECK IF IN I/O PAGE
2437 005022 101361      BHI      5$
2438 005024 013737 001160 001270      MOV      $TMPD,$BASE      ;LOAD NEW BUS ADDRESS
2439 005032 104401 050045      TYPE      OPR003      ;TYPE "RK611 VECTOR ADDRESS ( ) ="
2440 005036 013746 001264      MOV      $VECT1,-(SP)
2441 005042 042716 160000      BIC      #160000,(SP)
2442 005046 104402      TYPOC
2443 005050 104401 050037      TYPE      ,OPR002
2444 005054 104412      RDOCT      ;GET VALUE
2445 005056 012637 001160      MOV      (SP)+,$TMPD
2446 005062 001412      BEQ      10$      ;CHECK IF <CR>
2447 005064 022737 001000 001160      CMP      #1000,$TMPD      ;CHECK IF LEGAL
2448 005072 101757      BLOS     7$
2449 005074 042737 017777 001264      BIC      #17777,$VECT1      ;LOAD NEW VECTOR ADDRESS
2450 005102 053737 001160 001264      BIS      $TMPD,$VECT1
2451 005110 104401 050075      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)

```



CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 48  
GET VALUE FOR SOFTWARE SWITCH REGISTER

SEQ 0048

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	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	#PA7,-(SP)	;LOCK OUT INTERRUPTS
2481	005264	012746	005272	MOV	#TST1,-(SP)	
2482	005270	000002		RTI		

K04

.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.  
\*\*\*\*\*

\*\*\*\*\*

TST1: SCOPE  
MOV #100, \$TIMES ;DO 100. ITERATIONS  
MOV \$BASE, R2 ;LOAD RK611 BASE  
MOV #CCLR, RKCS1(R2) ;CLEAR RK611  
MOV #DMD, RKMR1(R2) ;PUT RK611 IN DIAGNOSTIC MODE  
MOV #SELDIV, RKCS1(R2) ;LOAD CS1 WITH SELECT DRIVE  
MOV #15, R0 ;WAIT FOR READY TO SET  
1\$: DEC R0  
BNE 1\$  
MOV RKCS1(R2), T.CS1 ;STORE COMMAND AND STATUS REG. 1  
MOV #SELDIV, E.CS1 ;LOAD EXPECT CS1  
CMP E.CS1, T.CS1 ;CHECK IF CS1 CHANGED  
BEQ 2\$ ;NO CONTINUE  
ERROR 77 ;CS1 INCORRECT  
BR TST2 ;GO ON TO NEXT TEST2\$: MOV #3\*4+2, R0 ;CLOCK IN DRIVE MESSAGE  
3\$: MOV #DMD, MCLK, RKMR1(R2)  
MOV #DMD, RKMR1(R2)  
DEC R0  
BNE 3\$  
MOV RKCS1(R2), T.CS1 ;STORE COMMAND AND STATUS REG. 1  
MOV RKMR2(R2), T.MR2 ;STORE MAINT REG. 2  
MOV RKMR3(R2), T.MR3 ;STORE MAINT REG. 3  
MOV #SELDIV, E.CS1 ;LOAD EXPECTED CS1  
CLR E.MR2 ;LOAD EXPECTED MAINT REG. 2  
CLR E.MR3 ;LOAD EXPECTED MAINT REG. 3  
CMP E.CS1, T.CS1 ;CHECK COMMAND AND STATUS REG. 1 CORRECT  
BEQ 4\$ ;YES, CHECK MESSAGES A & B  
ERROR 116 ;CS1 INCORRECT  
BR TS12 ;GO ON TO NEXT TEST4\$: BIT #17, T.MR2 ;CHECK IF DRIVE SELECT BITS ZERO  
BEQ 5\$ ;YES, CONTINUE  
ERROR 117 ;MESSAGE SELECT BITS NOT ZERO  
5\$: BIT #7760, T.MR2 ;CHECK IF COMMAND BITS ZERO  
BEQ 6\$ ;YES, CONTINUE6\$: ERROR 120 ;COMMAND BITS NOT ZERO  
BIT #70000, T.MR2 ;CHECK IF HEAD SELECT BITS ZERO  
BEQ 7\$ ;YES, CONTINUE  
ERROR 121 ;HEAD SELECT NOT ZERO  
7\$: BIT #BIT15, T.MR2 ;CHECK PARITY BIT ON MESS A ZERO  
BEQ 8\$ ;YES, CONTINUE2483  
2484  
2485  
2486  
2487  
2488  
2489  
2490  
2491  
2492  
2493  
2494  
2495  
2496  
2497  
2498  
2499  
2500  
2501  
2502  
2503  
2504  
2505  
2506  
2507  
2508  
2509  
2510  
2511  
2512  
2513  
2514  
2515  
2516  
2517  
2518  
2519  
2520  
2521  
2522  
2523  
2524  
2525  
2526  
2527  
2528  
2529  
2530  
2531  
2532  
2533  
2534  
2535  
2536  
2537  
2538005272 000004  
005274 012737 000144 001200  
005302 013702 001270  
005306 012762 100000 000000  
005314 012762 000040 000026  
005322 012762 000001 000000  
005330 012700 000015  
005334 005300  
005336 001376  
005340 016237 000000 004120  
005346 012737 000001 004160  
005354 023737 004160 004120  
005362 001402  
005364 104077  
005366 000503  
005370 012700 000016  
005374 012762 000440 000026  
005402 012762 000040 000026  
005410 005300  
005412 001370  
005414 016237 000000 004120  
005422 016237 000034 004146  
005430 016237 000036 004150  
005436 012737 000001 004160  
005444 005037 004206  
005450 005037 004210  
005454 023737 004160 004120  
005462 001402  
005464 104116  
005466 000443  
005470 032737 000017 004146  
005476 001401  
005500 104117  
005502 032737 007760 004146  
005510 001401  
005512 104120  
005514 032737 070000 004146  
005522 001401  
005524 104121  
005526 032737 100000 004146  
005534 001401

L04

CZR6BC0 RK611 DSKLS CTRL PRT2 MACY11 30(1046) 02-DEC-77 09:31 PAGE 50  
 CZR6BC.P11 02-DEC-77 09:22 T1 FIRST COMMAND IN MAINT MODE

SEQ 0050

```

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$: BIT #BIT15,T.MR3 ; CHECK PARITY BIT ON MESSAGE B
2547 005572 001401          BEQ TST2         ; YES, GO ON TO NEXT TEST
2548 005574 104125          ERROR 125          ; PARITY ON MESS. B NOT ZERO
2549
2550 *****
2551 *TEST 2 DRIVE SELECT BITS LOADING FOR DRIVE MESS.
2552 *
2553 * INITIALIZE RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
2554 * DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 2 WITH
2555 * ZERO. LOAD COMMAND AND STATUS REGISTER WITH A SELECT
2556 * COMMAND. CLOCK IN MESSAGES A AND B INTO SHIFT REGISTER.
2557 * MAKE SURE CORRECT MESSAGES ARE LOADED. REPEAT FOR DRIVE
2558 * SELECT = 1-17.
2559 *
2560 *****
2561 TST2: SCOPE
2562 005600 012737 000144 001200 MOV #100,$TIMES ; DO 100. ITERATIONS
2563 005606 013702 001270 001200 MOV $BASE,R2 ; LOAD RK611 BASE
2564 005612 005037 004244 001200 CLR DRVC00 ; INITIALIZE DRIVE SELECT CODE
2565 005616 012737 000001 004160 MOV #SELDV,E.CS1 ; LOAD EXPECTED CS1
2566 005624 012737 005632 001110 MOV #1$,SLPERR ; LOAD LOOP ON ERROR LOCATION FOR
2567 ; SUBTEST LOOP
2568
2569 005632          1$: MOV #CCLR,RKCS1(R2) ; CLEAR RK611
2570 005632 012762 100000 000000 MOV #DMD,RKMR1(R2) ; PUT RK611 IN DIAGNOSTIC MODE
2571 005640 012762 000040 000026 MOV DRVC00,RKCS2(R2) ; LOAD DRIVE NUMBER
2572 005646 013762 004244 000010 MOV #SELDV,RKCS1(R2) ; LOAD SELECT COMMAND
2573 005654 012762 000001 000000 MOV #3*4+2,R0 ; CLOCK IN DRIVE MESSAGE
2574 005662 012700 000016 000000 2$: MOV #DMD!MCLK,RKMR1(R2)
2575 005666 012762 000440 000026 MOV #DMD,RKMR1(R2)
2576 005674 012762 000040 000026 DEC R0
2577 005702 005300 000000 000000 BNE 2$
2578 005704 001370 000000 000000 MOV RKCS1(R2),T.CS1 ; STORE COMMAND AND STATUS REG. 1
2579 005706 016237 000000 004120 MOV RKMR2(R2),T.MR2 ; STORE MAINT REG. 2
2580 005714 016237 000034 004146 MOV RKMR3(R2),T.MR3 ; STORE MAINT REG. 3
2581 005722 016237 000036 004150 MOV DRVC00,E.MR2 ; LOAD EXPECTED MAINT REG. 2
2582 005730 013737 004244 004206 CLR E.MR3 ; LOAD EXPECTED MAINT REG. 3
2583 005736 005037 004210 004120 CMP E.CS1,T.CS1 ; CHECK IF CS1 CORRECT
2584 005742 023737 004160 004120 BEQ 3$ ; YES, CHECK MESSAGE A&B
2585 005750 001405 000000 000000 ERROR 2
2586 005752 104002 000000 000000 MOV #CCLR,RKCS1(R2) ; CLEAN UP FOR NEXT CONFIGURATION
2587 005754 012762 100000 000000 BR 25$ ; CHECK IF LOOP ON ERROR
2588 005762 000426 000000 000000
2589
2590 005764 013737 004146 001160 3$: MOV T.MR2,$TMP0 ; MASK BITS NOT UNDER TEST
2591 005772 042737 177760 001160 BIC #177760,$TMP0
2592 006000 023737 004244 001160 CMP DRVC00,$TMP0 ; CHECK IF DRIVE SELECT BITS CORRECT
2593 006006 001402 000000 000000 BEQ 4$ ; YES, CHECK MESSAGES A&B
2594 006010 104003 000000 000000 ERROR 3 ; DRIVE SELECT BITS INCORRECT

```

M04

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 51  
T2 DRIVE SELECT BITS LOADING FOR DRIVE MESS.

SEQ 0051

```

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 ST3: SCOPE
2618 MOV #100,$TIMES ;DO 100. ITERATIONS
2619 MOV $BASE,R2 ;LOAD RK611 BASE
2620 MOV #EM100,EM1N ;LOAD ERROR MESSAGE
2621 MOV #CCLR,RKCS1(R2) ;CLEAR RK611
2622 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
2623 MOV #CFMT:SELDV,RKCS1(R2) ;LOAD CFMT:SELDV INTO COMMAND AND STATUS REG.
2624 MOV #CFMT:SELDV,E.CS1 ;LOAD EXPECT CS1
2625 MOV #3*4+2,R0 ;CLOCK IN DRIVE MESSAGES
2626 006134 012762 000440 000026 1$: MOV #DMD:MCLK,RKMR1(R2)
2627 006142 012762 000040 000026 MOV #DMD,RKMR1(R2)
2628 006150 005300 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

```

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 52  
T3 FORMAT BIT LOADING TO FOR DRIVE MESS.

SEQ 0052

C  
C

```

2651 006300 104001          ERROR 1
2652 006302 023737 004210 004150 4$: CMP E.MR3,T.MR3 ;CHECK IF DRIVE MESSAGE B CORRECT
2653 006310 001404          BEQ TST4 ;YES, GO ON TO NEXT TEST
2654 006312 012737 057112 001302 MOV #EM2002,EM1N+2 ;LOAD ERROR MESSAGE
2655 006320 104001          ERROR 1
2656
2657 *****
2658 *TEST 4 HEAD SELECT BITS LOADING FOR DRIVE MESS.
2659 *
2660 * CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
2661 * DIAGNOSTIC MODE. LOAD TRACK ADDRESS WITH ZERO. LOAD
2662 * COMMAND AND STATUS REGISTER 2 WITH ZERO. LOAD COMMAND
2663 * AND STATUS REGISTER WITH SELECT COMMAND. CLOCK IN
2664 * MESSAGE A AND B INTO SHIFT REGISTER. MAKE SURE CORRECT
2665 * MESSAGE IS LOADED. REPEAT FOR TRACK ADDRESS = 1-7.
2666 *
2667 *****
2668 TST4: SCOPE
2669 MOV #100,$TIMES ;DO 100. ITERATIONS
2670 MOV $BASE,R2 ;LOAD RK611 BASE
2671 HOCODE ;CLEAR HEAD SELECT CODE
2672 MOV #SELDIV,E.CS1 ;LOAD EXPECTED CS1
2673 MOV #1$,$LPERR ;LOAD LOOP ON ERROR LOCATION FOR
2674 ; SUBTEST LOOP
2675
2676 1$: MOV #CCLR,RKCS1(R2) ;CLEAR RK611
2677 MOV #DMD,RKMR1(R2) ;PUT RK611 IN DIAGNOSTIC MODE
2678 CLR -(SP) ;MAKE ROOM ON STACK
2679 MOVB HOCODE,1(SP) ;LOAD HEAD ADDRESS
2680 MOV (SP)+,RKDC(R2)
2681 MOV #SELDIV,RKCS1(R2) ;LOAD SELECT COMMAND
2682 MOV #3*4+2,R0 ;CLOCK IN DRIVE MESSAGE
2683 2$: MOV #DMD!MCLK,RKMR1(R2)
2684 MOV #DMD,RKMR1(R2)
2685 DEC R0
2686 BNE 2$
2687 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2688 MOV RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
2689 MOV RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
2690 CLR E.MR2
2691 MOVB HOCODE,E.MR2+1 ;GENERATE EXPECTED MAINT REG. 2
2692 ASL E.MR2
2693 ASL E.MR2
2694 ASL E.MR2
2695 ASL E.MR2
2696 CLR E.MR3 ;LOAD EXPECTED MAINT REG. 3
2697 CMP E.CS1,T.CS1 ;CHECK IF CS1 CORRECT
2698 BEQ 3$ ;YES, CHECK MESSAGE A&B
2699 ERROR 6
2700 MOV #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2701 BR 25$ ;CHECK IF LOOP ON ERROR
2702
2703 3$: MOV T.MR2,$TMP0 ;MASK BITS NOT UNDER TEST
2704 BIC #103777,$TMP0
2705 CMP E.MR2,$TMP0 ;CHECK IF HEAD SELECT BITS CORRECT
2706

```

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 53  
T4 HEAD SELECT BITS LOADING FOR DRIVE MESS.

SEQ 0053

```

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 A INCORRECT
2717 006616 104415      SCOPI      ;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      STS:    SCOPE
2733 006636 012737 000144 001200      MOV      #100,$TIMES    ;DO 100. ITERATIONS
2734 006644 013702 001270      MOV      $BASE,R2    ;LOAD RK611 BASE
2735 006650 005037 004246      CLR      MSGCOD    ;INITIALIZE MESSAGE SELECT
2736 006654 012737 000001 004160      MOV      #SELDIV,E.CS1    ;LOAD EXPECTED CSI
2737 006662 012737 006670 001110      MOV      #1$,SLPERR    ;LOAD LOOP ON ERROR LOCATION FOR
2738 ;      SUBTEST LOOP
2739
2740 006670      1$:    MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
2741 006670 012762 100000 000000      MOV      MSGCOD,RKMR1(R2) ;LOAD MESSAGE SELECT BITS
2742 006676 013762 004246 000026      BIS      #0MD,RKMR1(R2) ;PUT RK611 IN DIAGNOSTIC MODE
2743 006704 052762 000040 000026      MOV      #SELDIV,RKCS1(R2) ;LOAD SELECT COMMAND
2744 006712 012762 000001 000000      MOV      #3*4+2,R0    ;CLOCK IF DRIVE MESSAGE
2745 006720 012700 000016      2$:    BIS      #MCLK,RKMR1(R2)
2746 006724 052762 000400 000026      BIC      #MCLK,RKMR1(R2)
2747 006732 042762 000400 000026      DEC      R0
2748 006740 005300      BNE      2$
2749 006742 001370      MOV      RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2750 006744 016237 000000 004120      MOV      RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
2751 006752 016237 000026 004144      MOV      RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2752 006760 016237 000034 004146      MOV      RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2753 006766 016237 000036 004150      MOV      MSGCOD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
2754 006774 013737 004246 004204      BIS      #MEWD:0MD,E.MR1
2755 007002 052737 002040 004204      BIT      #ECCW,T.MR1
2756 007010 032737 020000 004144      BEQ      10$
2757 007016 001403      BIS      #ECCW,E.MR1
2758 007020 052737 020000 004204      CLR      E.MR2 ;LOAD EXPECTED MAINT REG. 2
2759 007026 005037 004206      MOV      MSGCOD,E.MR3 ;LOAD EXPECTED MAINT REG. 3
2760 007032 013737 004246 004210      CMP      E.CS1,T.CS1 ;CHECK IF CSI CORRECT
2761 007040 023737 004160 004120      BEQ      3$      ;YES, CHECK MAINT REG. 1
2762 007046 001405

```

```

2763 007050 104012          ERROR 12          ;CS1 INCORRECT
2764 007052 012762 100000 000000  MOV      #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2765 007060 000437          BR      25$      ;CHECK IF LOOP ON ERROR
2766
2767 007062 023737 004204 004144 3$:  CMP      E.MR1,T.MR1      ;CHECK IF MAINT REG. 1 CORRECT
2768 007070 001405          BEQ      4$      ;YES, CHECK MESSAGE A&B
2769 007072 104013          ERROR 13          ;MR1 INCORRECT
2770 007074 012762 100000 000000  MOV      #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2771 007102 000426          BR      25$      ;CHECK IF LOOP ON ERROR
2772
2773 007104 013737 004150 001160 4$:  MOV      T.MR3,$TMPD      ;MASK BITS NOT UNDER TEST
2774 007112 042737 177760 001160  BIC      #177760,$TMPD
2775 007120 023737 004246 001160  CMP      MSGCOD,$TMPD      ;CHECK IF MESSAGE SELECT CODE CORRECT
2776 007126 001402          BEQ      5$      ;YES, CHECK MESSAGES A&B
2777 007130 104014          ERROR 14          ;MESSAGE SELECT CODE INCORRECT
2778 007132 000412          BR      25$
2779
2780 007134 023737 004206 004146 5$:  CMP      E.MR2,T.MR2      ;CHECK IF MESSAGE A CORRECT
2781 007142 001401          BEQ      6$      ;YES, CHECK MESSAGE B
2782 007144 104015          ERROR 15          ;MESSAGE A INCORRECT
2783 007146 023737 004210 004150 6$:  CMP      E.MR3,T.MR3      ;CHECK IF MESSAGE B CORRECT
2784 007154 001401          BEQ      25$      ;YES, CHECK IF LOOP ON ERROR
2785 007156 104016          ERROR 16          ;MESSAGE B INCORRECT
2786 007160 104415          SCOP1      ;CHECK IF LOOP ON ERROR
2787 007162 005237 004246          INC      MSGCOD      ;INCREMENT MESSAGE SELECT CODE
2788 007166 022737 000017 004246  CMP      #17,MSGCOD      ;CHECK IF FINISHED
2789 007174 103235          BHIS     1$      ;NO, TRY NEXT CONFIGURATION
2790
2791 *****
2792 *TEST 6 CLEAR DRIVE COMMAND LOADING FOR DRIVE MESS
2793 *
2794 * CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
2795 * DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 WITH
2796 * A DRIVE CLEAR. CLOCK MESSAGE A AND B INTO SHIFT REGISTERS.
2797 * MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY. REPEAT
2798 * FOR 24 SECTOR FORMAT.
2799 *
2800 *****
2801 007176 000004          ST6:  SCOPE
2802 007200 012737 000144 001200  MOV      #100,$TIMES      ;DO 100. ITERATIONS
2803 007206 013702 001270          MOV      $BASE,R2      ;LOAD RK611 BASE
2804 007212 012737 052121 001300  MOV      #EM101,EM1N      ;LOAD ERROR MESSAGE
2805 007220 012737 000005 004160  MOV      #CLEAR,E.CS1      ;LOAD EXPECTED COMMAND AND STATUS REG. 1
2806 007226 012737 000400 004206  MOV      #5,CLR,E.MR2      ;LOAD EXPECTED MAINT. REG. 2
2807 007234 012737 007242 001110  MOV      #1$,SLPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2808                                     ; SUBTEST LOOP
2809
2810 007242          1$:  MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
2811 007242 012762 100000 000000  MOV      #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
2812 007250 012762 000040 000026  MOV      E.CS1,RKCS1(R2) ;LOAD CLEAR INTO COMMAND AND STATUS REG. 1
2813 007256 013762 004160 000000  MOV      #3*4+2,R0      ;CLOCK IN DRIVE MESSAGE
2814 007264 012700 000016          MOV      #DMD,MCLK,RKMR1(R2)
2815 007270 012762 000040 000026  MOV      #DMD,RKMR1(R2)
2816 007276 012762 000040 000026  DEC      R0
2817 007304 005300          BNE      2$
2818 007306 001370          2$:

```

CZR68C0 RK611 DSKLS CTRL PRT2  
CZR68C.P11 02-DEC-77 09:22

MACY11 30(1046)  
T6

02-DEC-77 09:31 PAGE 55  
CLEAR DRIVE COMMAND LOADING FOR DRIVE MESS

SEQ 0055

```

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,EMIN+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$:      MOV      T.MR2,$TMPD ;MASK BITS NOT UNDER TEST
2831 007374 042737 176377 001160      BIC      #C<S.FMT!S.CLR>,$TMPD
2832 007402 023737 004206 001160      CMP      E.MR2,$TMPD ;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,EMIN+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,EMIN+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 ERROR
2845 007454 012737 057112 001302      MOV      #EM2002,EMIN+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                                ;*
2858                                ;* CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
2859                                ;* DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 WITH
2860                                ;* AN UNLOAD COMMAND. CLOCK IN MESSAGES A AND B INTO SHIFT
2861                                ;* REGISTERS. MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
2862                                ;* REPEAT FOR 24 SECTOR FORMAT.
2863                                ;*****
2864 007514 000004      TST7: SCOPE
2865 007516 012737 000144 001200      MOV      #100,$TIMES ;DO 100. ITERATIONS
2866 007524 013702 001270              MOV      $BASE,R2 ;LOAD RK611 BASE
2867 007530 012737 052170 001300      MOV      #EM102,EMIN ;LOAD ERROR MESSAGE
2868 007536 012737 000007 004160      MOV      #UNLOAD,E.CS1 ;LOAD EXPECTED COMMAND AND STATUS REG. 1
2869 007544 012737 002000 004206      MOV      #S.UNLD,E.MR2 ;LOAD EXPECTED MAINT. REG. 2
2870 007552 012737 007560 001110      MOV      #1$,$LPERR ;LOAD LOOP ON ERROR LOCATION FOR
2871                                ; SUBTEST LOOP
2872
2873 007560 1$:      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
2874 007560 012762 100000 000000

```



## E05

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T702-DEC-77 09:31 PAGE 56  
UNLOAD COMMAND LOADING FOR DRIVE MESS.

SEQ 0056

```

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,PO ;CLOCK IN DRIVE MESSAGE
2878 007606 012762 000440 000026 2$:    MOV      #DMD!MCLK,RKMR1(R2)
2879 007614 012762 000040 000026      MOV      #DMD,RKMR1(R2)
2880 007622 005300                DEC      RO
2881 007624 001370                BNE      2$
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$:    MOV      T.MR2,$TMPD ;MASK BITS NOT UNDER TEST
2894 007712 042737 174777 001160      BIC      #C<S.FMT!S.UNLD,$TMPD
2895 007720 023737 004206 001160      CMP      E.MR2,$TMPD ;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 000427                BR       25$ ;CHECK IF LOOP ON ERROR
2901
2902 007742 023737 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 ERROR
2908 007772 012737 057112 001302      MOV      #EM2002,EMIN+2 ;LOAD ERROR MESSAGE
2909 010000 104001                ERROR    1
2910 010002 104415                SCOP1    ;CHECK IF LOOP ON ERROR
2911 010004 032737 010000 004160 25$:    BIT      #CFMT,E.CS1 ;CHECK IF ISSUED IN 24 SECTOR FORMAT
2912 010012 001007                BNE      TST10 ;YES, GO ON TO NEXT TEST
2913 010014 052737 010000 004160      BIS      #CFMT,E.CS1 ;INDICATE COMMAND IN 24 SECTOR FORMAT
2914 010022 052737 001000 004206      BIS      #S.FMT,E.MR2
2915 010030 000653                BR       1$ ;REISSUE IN 24 SECTOR FORMAT
2916
2917                                     ;*****
2918                                     ;TEST 10 PACK ACKNOWLEDGE COMMAND LOADING FOR DRIVE MESS.
2919                                     ;*
2920                                     ;*
2921                                     ;* CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
2922                                     ;* DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 WITH
2923                                     ;* A PACK ACKNOWLEDGE. CLOCK MESSAGES A AND B INTO SHIFT
2924                                     ;* REGISTERS. MAKE SURE SHIFT REGISTERS ARE LOADED CORRECTLY.
2925                                     ;* REPEAT FOR 24 SECTOR FORMAT.
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

```

## F05

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T10

02-DEC-77 09:31 PAGE 57

PACK ACKNOWLEDGE COMMAND LOADING FOR DRIVE MESS.

SEQ 0057

```

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$,SLPERR      ;LOAD LOOP ON ERROR LOCATION FOR
2934                                     ; SUBTEST LOOP
2935
2936 010076                                     1$:      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
2937 010076 012762 100000 000000      MOV      #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
2938 010104 012762 000040 000026      MOV      E.CS1,RKCS1(R2) ;LOAD PACK INTO COMMAND AND STATUS REG. 1
2939 010112 013762 004160 000000      MOV      #3*4+2,R0      ;CLOCK IN DRIVE MESSAGE
2940 010120 012700 000016 000000      MOV      #DMD!MCLK,RKMR1(R2)
2941 010124 012762 000440 000026      2$:      MOV      #DMD,RKMR1(R2)
2942 010132 012762 000040 000026      DEC      R0
2943 010140 005300                                     BNE      2$
2944 010142 001370      MOV      RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
2945 010144 016237 000000 004120      MOV      RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG. 2
2946 010152 016237 000034 004146      MOV      RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
2947 010160 016237 000036 004150      CLR      E.MR3          ;STORE EXPECTED MAINT REG. 3
2948 010166 005037 004210 000000      CMP      E.CS1,T.CS1    ;CHECK IF CS1 CORRECT
2949 010172 023737 004160 004120      BEQ      3$            ;YES, CHECK MESSAGE A&B
2950 010200 001410      MOV      #EM2003,EM1N+2 ;LOAD ERROR MESSAGE
2951 010202 012737 057141 001302      ERROR   1
2952 010210 104001      MOV      #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
2953 010212 012762 100000 000000      MOV      25$          ;CHECK IF LOOP ON ERROR
2954 010220 000437      BR
2955
2956 010222 013737 004146 001160      3$:      MOV      T.MR2,$TMPD ;MASK BITS NOT UNDER TEST
2957 010230 042737 172777 001160      BIC      #C<S.FMT!S.PACK>,$TMPD
2958 010236 023737 004206 001160      CMP      E.MR2,$TMPD    ;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,EM1N+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,EM1N+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 ERROR
2971 010310 012737 057112 001302      MOV      #EM2002,EM1N+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.

```

SEQ 0058

Line	Address	Hex	Hex	Hex	Label	Instruction	Comment
2987						*****	
2988						ST11: SCOPE	
2989	010350	000004				MOV #100, \$TIMES	;; DO 100. ITERATIONS
2990	010352	012737	000144	001200		MOV \$BASE, R2	;; LOAD RK611 BASE
2991	010360	013702	001270			MOV #EM104, EMIN	;; LOAD ERROR MESSAGE
2992	010364	012737	052306	001300		MOV #CCLR, RKCS1(R2)	;; CLEAR RK611
2993	010372	012762	100000	000000		MOV #DMD, RKMR1(R2)	;; PUT RK611 IN MAINTENANCE MODE
2994	010400	012762	000040	000026		MOV #RECAL, RKCS1(R2)	;; LOAD RECAL INT. COMMAND AND STATUS REG. 1
2995	010406	012762	000013	000000		MOV #RECAL, E.CS1	;; LOAD EXPECT CS1
2996	010414	012737	000013	004160		MOV #3*4+2, R0	;; CLOCK IN DRIVE MESSAGES
2997	010422	012700	000016			MOV #DMD!MCLK, RKMR1(R2)	
2998	010426	012762	000440	000026	1\$:	MOV #DMD, RKMR1(R2)	
2999	010434	012762	000040	000026		DEC R0	
3000	010442	005300				BNE 1\$	
3001	010444	001370				MOV RKCS1(R2), T.CS1	;; STORE COMMAND AND STATUS REG. 1
3002	010446	016237	000000	004120		MOV RKMR2(R2), T.MR2	;; STORE MAINT REG. 2
3003	010454	016237	000034	004146		MOV RKMR3(R2), T.MR3	;; STORE MAINT REG. 3
3004	010462	016237	000036	004150		MOV #S.RECL, E.MR2	;; LOAD EXPECTED MAINT REG. 2
3005	010470	012737	000040	004206		CLR E.MR3	;; LOAD EXPECTED MAINT REG. 3
3006	010476	005037	004210			CMP E.CS1, T.CS1	;; CHECK IF CS1 CORRECT
3007	010502	023737	004160	004120		BEQ 2\$	;; YES, CHECK MESSAGE A&B
3008	010510	001410				MOV #EM2003, EMIN+2	;; LOAD ERROR MESSAGE
3009	010512	012737	057141	001302		ERROR 1	
3010	010520	104001				MOV #CCLR, RKCS1(R2)	;; CLEAN UP FOR NEXT TEST
3011	010522	012762	100000	000000		BR TST12	;; GO ON TO NEXT TEST
3012	010530	000431					
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, EMIN+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, EMIN+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, EMIN+2	;; LOAD ERROR MESSAGE
3027	010612	104001				ERROR 1	
3028							
3029							
3030						*****	
3031						TEST 12	START SPINDLE COMMAND LOADING FOR DRIVE MESS.
3032							
3033							
3034							
3035							
3036							
3037							
3038	010614	000004				*****	
3039	010616	012737	000144	001200		ST12: SCOPE	
3040	010624	013702	001270			MOV #100, \$TIMES	;; DO 100. ITERATIONS
3041	010630	012737	052355	001300		MOV \$BASE, R2	;; LOAD RK611 BASE
3042	010636	012762	100000	000000		MOV #EM105, EMIN	;; LOAD ERROR MESSAGE
</							

## H05

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T1202-DEC-77 09:31 PAGE 59  
START SPINDLE COMMAND LOADING FOR DRIVE MESS.

SEQ 0059

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								
3080								
3081								
3082								
3083								
3084								
3085								
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\$,\$LPEHR	;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

\*\*\*\*\*  
 \*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.  
 \*\*\*\*\*

CZR68C0 RK611 DSKLS CTRL PRT2  
CZR69C.P11 02-DEC-77 09:22

MACY11 30(1046)  
T13

02-DEC-77 09:31 PAGE 67  
SEEK AND CYLINDER ADD 0-777 LOADING FOR DRIVE MESS

SEQ 0060

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,RO	:CLOCK IN DRIVE MESSAGE
3102	011150	012762	00044C	000026	2\$:	MOV	#DMD:MCLK,RKMR1(R2)	
3103	011156	012762	000040	000026		MOV	#DMD,RKMR1(R2)	
3104	011164	005300				DEC	RO	
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 MESSAGE 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

```

*****
*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.
*****

```

3143  
3144  
3145  
3146  
3147  
3148  
3149  
3150  
3151  
3152  
3153  
3154

```

3155 011376 000004          TST14: SCOPE
3156 011400 012737 000144 001200      MOV    #100.,$TIMES      ;DO 100. ITERATIONS
3157 011406 013702 001270          MOV    $BASE,R2          ;LOAD RK611 BASE
3158 011412 012737 001000 004252      MOV    #1000,CYLIN      ;INITIALIZE CYLINDER
3159 011420 005037 004210          CLR    E.MR3            ;LOAD EXPECTED
3160 011424 012737 000017 004160      MOV    #SEEK,E.CS1      ;LOAD EXPECTED CS1
3161 011432 012737 011440 001110      MOV    #15,$LPERR      ;LOAD LOOP ON ERROR LOCATION FOR
3162                                     ; SUBTEST LOOP
3163
3164 011440 012762 100000 000000      1$:  MOV    #CCLR,RKCS1(R2) ;CLEAR RK611
3165 011446 012762 000040 000026      MOV    #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
3166 011454 013762 004252 000020      MOV    CYLIN,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
3167 011462 012762 000017 000000      MOV    #SEEK,RKCS1(R2) ;ISSUE SEEK
3168 011470 012700 000016          MOV    #3*4+2,R0          ;CLOCK IN DRIVE MESSAGE
3169 011474 012762 000440 000026      2$:  MOV    #DMD!MCLK,RKMR1(R2)
3170 011482 012762 000040 000026      MOV    #DMD,RKMR1(R2)
3171 011490 005300          DEC    R0
3172 011510 001370          BNE    2$
3173 011514 016237 000000 004120      MOV    RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3174 011522 016237 000034 004146      MOV    RKMR2(R2),T.MR2 ;STORE MAINT REG.2
3175 011530 016237 000036 004150      MOV    RKMR3(R2),T.MR3 ;STORE MAINT REG.3
3176 011536 012737 000020 004206      MOV    #S.SEEK,E.MR2    ;LOAD EXPECTED MAINT REG. 2
3177 011544 023737 004160 004120      CMP    E.CS1,T.CS1      ;CHECK IF CS1 CORRECT
3178 011552 001405          BEQ    3$                      ;YES, CHECK MESSAGE A&B
3179 011554 104017          ERROR 17
3180 011556 012762 100000 000000      MOV    #CCLR,RKCS1(R2) ;CLEAN UP FOR NEXT CONFIGURATION
3181 011564 000434          BR     25$                      ;CHECK IF LOOP ON ERROR
3182
3183 011566 032737 000020 004146      3$:  BIT    #S.SEEK,T.MR2    ;CHECK IF SEEK COMMAND BIT SEEK
3184 011574 001002          BNE    4$                      ;YES, CHECK CYLINDER ADDRESS BITS
3185 011576 104020          ERROR 20
3186 011600 000426          BR     25$                      ;SEEK BIT NOT SET
3187                                     ;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
3194 011630 000412          BR     25$                      ;CYLINDER ADDRESS BITS INCORRECT
3195                                     ;CHECK IF LOOP ON ERROR
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
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
3202 011656 104415          25$:  SCOP1                      ;MESSAGE INCORRECT
3203 011660 022737 001400 004252      CMP    #1400,CYLIN      ;CHECK IF LOOP ON ERROR
3204 011666 001407          BEQ    TST15                     ;CHECK IF CYLINDER 1400
3205 011670 012737 001400 004252      MOV    #1400,CYLIN      ;YES, GO ON TO NEXT TEST
3206 011676 012737 010000 004210      MOV    #10000,E.MR3     ;SET CYLINDER=1400
3207 011704 000655          BR     1$                      ;LOAD EXPECTED CONFIGUR
3208                                     ;TRY NEXT CONFIGURATION
3209
3210 ;*****
; *TEST 15      SEEK AND CYLINDER ADD 0,777-1777 LOADING FOR DRIVE MESS

```

## K05

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 62  
T15 SEEK AND CYLINDER ADD 0,777-1777 LOADING FOR DRIVE MESS

SEQ 0062

```

3211
3212
3213
3214
3215
3216
3217
3218
3219 011706 000004
3220 011710 012737 000144 001200
3221 011716 013702 001270
3222 011722 005037 004252
3223 011726 012737 002017 004160
3224 011734 012737 011742 001110
3225
3226
3227 011742
3228 011742 012762 100000 000000
3229 011750 012762 000040 000026
3230 011756 013762 004252 000020
3231 011764 012762 002017 000000
3232 011772 012700 000016
3233 011776 012762 000440 000026
3234 012004 012762 000040 000026
3235 012012 005300
3236 012014 001370
3237 012016 016237 000000 004120
3238 012024 016237 000034 004146
3239 012032 016237 000036 004150
3240 012040 012737 000020 004206
3241 012046 013737 004252 004210
3242 012054 006337 004210
3243 012060 006337 004210
3244 012064 006337 004210
3245 012070 006337 004210
3246 012074 023737 004160 004120
3247 012102 001405
3248 012104 104024
3249 012106 012762 100000 000000
3250 012114 000434
3251
3252 012116 032737 000020 004146
3253 012124 001002
3254 012126 104025
3255 012130 000426
3256
3257 012132 013737 004150 001160
3258 012140 042737 140017 001160
3259 012146 023737 004210 001160
3260 012154 001402
3261 012156 104026
3262 012160 000412
3263
3264 012162 023737 004206 004146
3265 012170 001401
3266 012172 104027

```

\*\*\*\*\*

15: SCOPE

MOV #100, \$TIMES ; DO 100 ITERATIONS

MOV \$BASE, R2 ; LOAD RK611 BASE

CLR CYLIN ; INITIALIZE CYLINDER

MOV #CDT!SEEK, E.CS1 ; LOAD EXPECTED CS1

MOV #1\$, \$LPERA ; LOAD LOOP ON ERROR LOCATION FOR

; SUBTEST LOOP

1\$:

MOV #CCLR, RKCS1(R2) ; CLEAR RK611

MOV #DMD, RKMR1(R2) ; PUT RK611 IN MAINTENANCE MODE

MOV CYLIN, RKDCYL(R2) ; LOAD CYLINDER ADDRESS

MOV #CDT!SEEK, RKCS1(R2) ; ISSUE SEEK WITH CDT SET

MOV #3\*4+2, R0 ; CLOCK IN DRIVE MESSAGE

2\$:

MOV #DMD!MCLK, RKMR1(R2)

MOV #DMD, RKMR1(R2)

DEC R0

BNE 2\$

MOV RKCS1(R2), T.CS1 ; STORE COMMAND AND STATUS REG. 1

MOV RKMR2(R2), T.MR2 ; STORE MAINT REG. 2

MOV RKMR3(R2), T.MR3 ; STORE MAINT REG. 3

MOV #S.SEEK, E.MR2 ; LOAD EXPECTED MAINT REG. 2

MOV CYLIN, E.MR3 ; GENERATE EXPECTED MAINT REG. 3

ASL E.MR3

ASL E.MR3

ASL E.MR3

ASL E.MR3

CMP E.CS1, T.CS1 ; CHECK IF CS1 CORRECT

BEQ 3\$ ; YES, CHECK MESSAGE A&B

24

ERROR

MOV #CCLR, RKCS1(R2) ; CLEAN UP FOR NEXT CONFIGURATION

BR 25\$ ; CHECK IF LOOP ON ERROR

3\$:

BIT #S.SEEK, T.MR2 ; CHECK IF SEEK COMMAND BIT SET

BNE 4\$ ; YES, CHECK CYLINDER ADDRESS BITS

25

ERROR

BR 25\$ ; SEEK BIT NOT SEEK

25\$

; CHECK IF LOOP ON ERROR

4\$:

MOV T.MR3, \$TMP0 ; MASK BITS NOT UNDER TEST

BIC #140017, \$TMP0

CMP E.MR3, \$TMP0 ; CHECK IF CYLINDER ADDRESS BITS CORRECT

BEQ 5\$ ; YES, CHECK MESSAGES A&B

26

ERROR

BR 25\$ ; CYLINDER ADDRESS BIT INCORRECT

25\$

; CHECK IF LOOP ON ERROR

5\$:

CMP E.MR2, T.MR2 ; CHECK IF MESSAGE A CORRECT

BEQ 6\$ ; YES, CHECK M MESSAGE B

27

ERROR

# L05

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T15

02-DEC-77 09:31 PAGE 63  
SEEK AND CYLINDER ADD 0,777-1777 LOADING FOR DRIVE MESS

SEQ 0063

```

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$: SCOP1 ;CHECK IF LOOP ON ERROR
3271 012210 005737 004252 TST CYLIN ;CHECK IF ZERO
3272 012214 001003 BNE 26$ ;NO, INCREMENT CYLINDER
3273 012216 012737 00C776 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 1$TEST16: SCOPE
3290 012242 012737 000144 001200 MOV #100,$TIMES ;DO 100. ITERATIONS
3291 012250 013702 001270 MOV $BASE,R2 ;LOAD RK611 BASE
3292 012254 005037 004254 CLR OFFVAL ;INITIALIZE OFFSET VALUE
3293 012260 012737 000015 004160 MOV #OFFSET,E.CS1 ;LOAD EXPECTED CS1
3294 012266 005037 004206 CLR E.MR2 ;LOAD EXPECT MAINT REG 2
3295 012272 012737 012300 001110 MOV #1$,SLPERR ;LOAD LOOP ON ERROR LOCATION FOR
3296 ; SUBTEST LOOP
3297
3298 012300 1$: MOV #CCLR,RKCS1(R2) ;CLEAR RK611
3299 012300 012762 100000 000000 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
3300 012306 012762 000040 000026 MOV OFFVAL,RKASOF(R2) ;LOAD OFFSET VALUE
3301 012314 013762 004254 000016 MOV #OFFSET,RKCS1(R2) ;ISSUE OFFSET
3302 012322 012762 000015 000000 MOV #3*4+2,R0 ;CLOCK IN DRIVE MESSAGE
3303 012330 012700 000016 2$: MOV #DMD!MCLK,RKMR1(R2)
3304 012334 012762 000440 000026 MOV #DMD,RKMR1(R2)
3305 012342 012762 000040 000026 DEC R0
3306 012350 005300 BNE 2$
3307 012352 001370 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3308 012354 016237 000000 004120 MOV RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
3309 012362 016237 000034 004146 MOV RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
3310 012370 016237 000036 004150 CLR E.MR3 ;LOAD EXPECTED MAINT REG. 2
3311 012376 005037 004210 004210 MOV OFFVAL,E.MR3 ;GENERATE EXPECTED MR3
3312 012402 013737 004254 004210 COM E.MR3
3313 012410 005137 004210 BIC #177700,E.MR3
3314 012414 042737 177700 004210 ASL 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 BIS #14000,E.MR3
3319 012442 052737 014000 004210 BIT #BIT7,OFFVAL ;DETERMINE SIGN
3320 012450 032737 000200 004254 BIT 10$
3321 012456 001003 BNE 10$
3322 012460 052737 002000 004210 BIS #BIT10,E.MR3

```



# M05

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T16

02-DEC-77 09:31 PAGE 64  
OFFSET COMMAND LOADING FOR DRIVE MESS.

SEQ 0064

```

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$: MOV T.MR3,$TMPD ;MASK BITS NOT UNDER TEST
3330 012516 042737 140017 001160 BIC #140017,$TMPD
3331 012524 023737 004210 001160 CMP E.MR3,$TMPD ;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$: CMP E.MR3,T.MR3 ;CHECK IF MESSAGE B CORRECT
3340 012556 001401 BEQ 25$ ;YES, CHECK IF LOOP ON ERROR
3341 012562 104034 ERROR 34 ;MESSAGE B INCORRECT
3342 012564 104415 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
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358

```

```

*****
*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.
*****

```

```

*****
*TEST17: SCOPE
*
* MOV #100,$TIMES ;DO 100. ITERATIONS
* MOV $BASE,R2 ;LOAD RK611 BASE
* MOV #1777,CYLIN ;LOAD CYLINDER VALUE
* MOV #52,OFFVAL ;LOAD OFFSET VALUE
* MOV #SELDV,E.CS1 ;LOAD EXPECTED CS1
* MOV #CCLR,RKCS1(R2) ;CLEAR RK611
* MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
* MOV #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
* MOV #52,RKASOF(R2) ;LOAD OFFSET VALUE
* MOV #SELDV,RKCS1(R2) ;ISSUE SELDRV
* MOV #3*4+2,R0 ;CLOCK IN DRIVE MESSAGE
* MOV #DMD,MCLK,RKMR1(R2)
* MOV #DMD,RKMR1(R2)
* DEC R0
* BNE 1$
* MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
* MOV RKMR2(R2),T.MR2 ;STORE MAINT REG. 2
* MOV RKMR3(R2),T.MR3 ;STORE MAINT REG. 3
* MOV #0,E.MR2 ;LOAD EXPECTED MAINT REG. 2

```

# NOS

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T17

02-DEC-77 09:31 PAGE 65  
CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 1)

SEQ 0065

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
3394	013030	001401			BEQ	5\$	;CHECK IF MESSAGE A CORRECT
3395	013032	104040			ERROR	40	;YES, CHECK MESSAGE B
3396	013034	023737	004210	004150	5\$:	CMP	E.MR3,T.MR3
3397	013042	001401			BEQ	TST20	;MESS A INCORRECT
3398	013044	104041			ERROR	41	;CHECK IF MESSAGE B CORRECT
3399							;YES, GO ON TO NEXT TEST
3400							;MESS B INCORRECT
3401							
3402							
3403							
3404							
3405							
3406							
3407							
3408							
3409							
3410							
3411							
3412	013046	000004					
3413	013050	012737	000144	001200	TST20:	SCOPE	
3414	013056	013702	001270		MOV	#100,\$TIMES	;DO 100. ITERATIONS
3415	013062	012737	001777	004252	MOV	\$BASE,R2	;LOAD RK611 BASE
3416	013070	012737	000052	004254	MOV	#1777,CYLIN	;LOAD CYLINDER VALUE
3417	013076	012737	000003	004160	MOV	#52,OFFVAL	;LOAD OFFSET VALUE
3418	013104	012762	100000	000000	MOV	#PACK,E.CS1	;LOAD EXPECTED CS1
3419	013112	012762	000040	000026	MOV	#CCLR,RKCS1(R2)	;CLEAR RK611
3420	013120	012762	001777	000020	MOV	#DMD,RKMR1(R2)	;PUT RK611 IN MAINTENANCE MODE
3421	013126	012762	000052	000016	MOV	#1777,RKDCYL(R2)	;LOAD CYLINDER VALUE
3422	013134	012762	000003	000000	MOV	#52,RKASOF(R2)	;LOAD OFFSET VALUE
3423	013142	012700	000016		MOV	#PACK,RKCS1(R2)	;ISSUE PACK
3424	013146	012762	000440	000026	1\$:	MOV	#3*4+2,R0
3425	013154	012762	000040	000026	MOV	#DMD!MCLK,RKMR1(R2)	;CLOCK IN DRIVE MESSAGE
3426	013162	005300			MOV	#DMD,RKMR1(R2)	
3427	013164	001370			DEC	R0	
3428	013166	016237	000000	004120	BNE	1\$	
3429	013174	016237	000034	004146	MOV	RKCS1(R2),T.CS1	;STORE COMMAND AND STATUS REG. 1
3430	013202	016237	000036	004150	MOV	RKMR2(R2),T.MR2	;STORE MAINT REG. 2
3431	013210	012737	004000	004206	MOV	RKMR3(R2),T.MR3	;STORE MAINT REG. 3
3432	013216	005037	004210		MOV	#S.PACK,E.MR2	;LOAD EXPECTED MAINT REG. 2
3433	013222	023737	004160	004120	CLR	E.MR3	;LOAD EXPECTED MAINTENANCE REG. 3
3434	013230	001405			CMP	E.CS1,T.CS1	;CHECK IF CS1 CORRECT
					BEQ	2\$	;YES, CHECK MESSAGES A&B

\*\*\*\*\*  
\*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.  
\*\*\*\*\*

\*\*\*\*\*  
\*ST20: SCOPE  
\*DO 100. ITERATIONS  
\*LOAD RK611 BASE  
\*LOAD CYLINDER VALUE  
\*LOAD OFFSET VALUE  
\*LOAD EXPECTED CS1  
\*CLEAR RK611  
\*PUT RK611 IN MAINTENANCE MODE  
\*LOAD CYLINDER VALUE  
\*LOAD OFFSET VALUE  
\*ISSUE PACK  
\*CLOCK IN DRIVE MESSAGE  
\*STORE COMMAND AND STATUS REG. 1  
\*STORE MAINT REG. 2  
\*STORE MAINT REG. 3  
\*LOAD EXPECTED MAINT REG. 2  
\*LOAD EXPECTED MAINTENANCE REG. 3  
\*CHECK IF CS1 CORRECT  
\*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          25:
3440 013244 032737 004000 004146 BIT  #S.PACK,T.MR2 ; CHECK IF PACK COMMAND
3441                                ; BIT SET
3442 013252 001002          BNE  35 ; 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          35:
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  45 ; 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 45: CMP  E.MR2,T.MR2 ; CHECK IF MESSAGE A CORRECT
3454 013310 001401          BEQ  55 ; YES, CHECK MESSAGE B
3455 013312 104040          ERROR 40 ; MESS A INCORRECT
3456 013314 023737 004210 004150 55: 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 ; *****
3473 ; TST21: SCOPE
3474 ; MOV  #100,$TIMES ; DO 100. ITERATIONS
3475 ; MOV  $BASE,R2 ; LOAD RK611 BASE
3476 ; MOV  #1777,CYLIN ; LOAD CYLINDER VALUE
3477 ; MOV  #52,OFFVAL ; LOAD OFFSET VALUE
3478 ; MOV  #CLEAR,E.CS1 ; LOAD EXPECTED CS1
3479 ; MOV  #CCLR,RKCS1(R2) ; CLEAR RK611
3480 ; MOV  #DMD,RKMR1(R2) ; PUT RK611 IN MAINTENANCE MODE
3481 ; MOV  #1777,RKDCYL(R2) ; LOAD CYLINDER VALUE
3482 ; MOV  #52,RKASOF(R2) ; LOAD OFFSET VALUE
3483 ; MOV  #CLEAR,RKCS1(R2) ; ISSUE CLEAR
3484 ; MOV  #3*4+2,R0 ; CLOCK IN DRIVE MESSAGE
3485 ; MOV  #DMD,MCLK,RKMR1(R2)
3486 ; MOV  #DMD,RKMR1(R2)
3487 ; DEC  R0
3488 ; BNE  15
3489 ; MOV  RKCS1(R2),T.CS1 ; STORE COMMAND AND STATUS REG. 1
3490 ; MOV  RKMR2(R2),T.MR2 ; STORE MAINT REG. 2
3491 ; MOV  RKMR3(R2),T.MR3 ; STORE MAINT REG. 3

```

C06

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
12102-DEC-77 09:31 PAGE 67  
CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 3)

SEQ 0067

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.CS1,T.CS1	;CHECK IF CS1 CORRECT
3494	013510	001405			BEQ	25	;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				25:		
3500	013524	032737	000400	004146	BIT	#S.CLR.T.MR2	;CHECK IF CLEAR COMMAND
3501							;BIT SET
3502	013532	001002			BNE	35	;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				35:		
3507	013540	013737	004150	001160	MOV	T.MR3,\$TMP0	;MASK OUT BITS NOT UNDER TEST
3508	013546	042737	140017	001160	BIC	#140017,\$TMP0	
3509	013554	001402			BEQ	45	;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	45:	CMP	E.MR2,T.MR2
3514	013570	001401			BEQ	55	;CHECK IF MESSAGE A CORRECT
3515	013572	104040			ERROR	40	;YES, CHECK MESSAGE B
3516	013574	023737	004210	004150	55:	CMP	E.MR3,T.MR3
3517	013602	001401			BEQ	TST22	;CHECK IF MESSAGE B CORRECT
3518	013604	104041			ERROR	41	;YES, GO ON TO NEXT TEST
3519							;MESS B INCORRECT

\*\*\*\*\*  
\*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.

\*\*\*\*\*  
\*ST22: SCOPE  
\*\*\*\*\*

3532	013606	000004			MOV	#100,\$TIMES	;DO 100. ITERATIONS
3533	013610	012737	000144	001200	MOV	\$BASE,R2	;LOAD RK611 BASE
3534	013616	013702	001270		MOV	#1777,CYLIN	;LOAD CYLINDER VALUE
3535	013622	012737	001777	004252	MOV	#52,OFFVAL	;LOAD OFFSET VALUE
3536	013630	012737	000052	004254	MOV	#UNLOAD,E.CS1	;LOAD EXPECTED CS1
3537	013636	012737	000007	004160	MOV	#CCLR,RKCS1(R2)	;CLEAR RK611
3538	013644	012762	100000	000000	MOV	#DMD,RKMR1(R2)	;PUT RK611 IN MAINTENANCE MODE
3539	013652	012762	000040	000026	MOV	#1777,RKDCYL(R2)	;LOAD CYLINDER VALUE
3540	013660	012762	001777	000020	MOV	#52,RKASOF(R2)	;LOAD OFFSET VALUE
3541	013666	012762	000052	000016	MOV	#UNLOAD,RKCS1(R2)	;ISSUE UNLOAD
3542	013674	012762	000007	000000	MOV	#3*4+2,R0	;CLOCK IN DRIVE MESSAGE
3543	013702	012700	000016		MOV	#DMD!MCLK,RKMR1(R2)	
3544	013706	012762	000440	000026	MOV	#DMD,RKMR1(R2)	
3545	013714	012762	000040	000026	DEC	R0	
3546	013722	005300					

```

3547 013724 001370 BNE 15
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 25 ;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 032737 002000 004146 25: BIT #S.UNLD,T.MR2 ;CHECK IF UNLOAD COMMAND
3560 014004 032737 002000 004146 ; BIT SET
3561 014012 001002 BNE 35 ;YES, CHECK CYLINDER ADDRESS BITS
3562 014014 104036 ERROR 36 ;S.UNLD BIT NOT SET
3563 014016 000423 BR TST23 ;GO ON TO NEXT TEST
3564
3565 014020 013737 004150 001160 35: MOV T.MR3,$TMP0 ;MASK OUT BITS NOT UNDER TEST
3566 014020 042737 140017 001160 BIC #140017,$TMP0
3567 014026 001402 BEQ 45 ;CHECK IF CYLINDER ADDRESS ZERO
3568 014034 104037 ERROR 37 ;CYLINDER ADDRESS BITS INCORRECT
3569 014036 000412 BR TST23 ;GO ON TO NEXT TEST
3570
3571 014042 023737 004206 004146 45: CMP E.MR2,T.MR2 ;CHECK IF MESSAGE A CORRECT
3572 014050 001401 BEQ 55 ;YES, CHECK MESSAGE B
3573 014052 104040 ERROR 40 ;MESS A INCORRECT
3574 014054 023737 004210 004150 55: CMP E.MR3,T.MR3 ;CHECK IF MESSAGE B CORRECT
3575 014062 001401 BEQ TST23 ;YES, GO ON TO NEXT TEST
3576 014064 104041 ERROR 41 ;MESS B INCORRECT
3577
3578
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 TST23: SCOPE
3593 014066 000004 MOV #100,$TIMES ;DO 100. ITERATIONS
3594 014070 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
3595 014076 013702 001270 MOV #1777,CYLIN ;LOAD CYLINDER VALUE
3596 014102 012737 001777 004252 MOV #52,OFFVAL ;LOAD OFFSET VALUE
3597 014110 012737 000052 004254 MOV #SRTSPL,E.CS1 ;LOAD EXPECTED CS1
3598 014116 012737 000011 004160 MOV #CCLR,RKCS1(R2) ;CLEAR RK611
3599 014124 012762 100000 000000 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3600 014132 012762 000040 000026 MOV #1777,RKDCYL(R2) ;LOAD CYLINDER VALUE
3601 014140 012762 001777 000020 MOV #52,RKASOF(R2) ;LOAD OFFSET VALUE
3602 014146 012762 000052 000016 MOV #SRTSPL,RKCS1(R2) ;ISSUE SRTSPL
3603 014154 012762 000011 000000

```

```

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    3$
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$:
3620 014264 032737 000100 004146      BIT      #S.STSP,T.MR2  ;CHECK IF SRTSPL COMMAND
3621      BNE      3$      ;BIT SET
3622 014272 001002      ERROR    36      ;YES, CHECK CYLINDER ADDRESS BITS
3623 014274 104036      BR       TST24        ;S.STSP BIT NOT SET
3624 014276 000423      ;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

```

F06

CZR6BCD RK611: DSKLS CTRL FRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T2402-DEC-77 09:31 PAGE 70  
CYLINDER ADDRESS LOADING OF DRIVE MESS (PART 6)

SEQ 0070

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\$: BIT	#S.RECL,T.MR2	;CHECK IF RECAL COMMAND
3680	014544	032737	000040	004146			;BIT SET
3681					BNE	3\$	;YES, CHECK CYLINDER ADDRESS BITS
3682	014552	001002			ERROR	36	;S.RECL BIT NOT SET
3683	014554	104036			BR	TST25	;GO ON TO NEXT TEST
3684	014556	000423					
3685							
3686	014560				3\$: MOV	T.MR3,\$TMP0	;MASK OUT BITS NOT UNDER TEST
3687	014560	013737	004150	001160	BIC	#140017,\$TMP0	
3688	014566	042737	140017	001160	BEQ	4\$	;CHECK IF CYLINDER ADDRESS ZERO
3689	014574	001402			ERROR	37	;CYLINDER ADDRESS BITS INCORRECT
3690	014576	104037			BR	TST25	;GO ON TO NEXT TEST
3691	014600	000412					
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							
3702							
3703							
3704							
3705							
3706							
3707							
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

```

*****
*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.
*****

```

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	0 2737	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,\$TMPD	;MASK OUT BITS NOT UNDER TEST
3755	015100	042737	177760	001160		BIC	#177760,\$TMPD	
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								
3769								
3770								

\*\*\*\*\*  
\*TEST 26 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 2)  
\*  
\* CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN



## H06

CZRBBCD RK611 DSKLS CTRL PRT2  
CZRBBC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 72  
T26 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 2)

SEQ 0072

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.

\*\*\*\*\*

```

T26: SCOPE
MOV #100., $TIMES ; DO 100. ITERATIONS
MOV $BASE, R2 ; LOAD RK611 BASE
MOV #17, MSGCOD ; LOAD MESSAGE CODE FOR PRINT OUT
MOV #CLEAR, E.CS1 ; LOAD EXPECTED CS1
MOV #CCLR, RKCS1(R2) ; CLEAR RK611
MOV #DMD!17, RKMR1(R2) ; PUT RK611 IN MAINTENANCE MODE
; SELECT MESSAGE 17
; ISSUE CLEAR
; CLOCK IN DRIVE MESSAGE
1$: MOV #3*4+2, R0
BIS #MCLK, RKMR1(R2)
BIC #MCLK, RKMR1(R2)
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
10$: BEQ 10$
BIS #ECCW, E.MR1
MOV #5, CLR, 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
42 ERROR
MOV #CCLR, RKCS1(R2) ; CLEAR RK611 CONTROLLER FOR NEXT TEST
BR TST27 ; GO ON TO NEXT TEST
2$: CMP E.MR1, T.MR1 ; CHECK IF MAINT REG. 1 CORRECT
BEQ 3$ ; YES, CHECK MESSAGES A&B
43 ERROR ; MAINT REG. 1 INCORRECT
MOV #CCLR, RKCS1(R2) ; CLEAR RK611 CONTROLLER FOR NEXT TEST
BR TST27 ; GO ON TO NEXT TEST
3$: BIT #5, CLR, T.MR2 ; CHECK IF CLEAR COMMAND
; BIT SET
; YES, CHECK MESSAGE SELECT BITS
; S.CLR BIT NOT SET
; GO ON TO NEXT TEST
4$: MOV T.MR3, $TMP0 ; MASK OUT BITS NOT UNDER TEST
BIC #177760, $TMP0
BEQ 5$ ; CHECK IF MESSAGE SELECT ZERO
45 ERROR ; MESSAGE SELECT BITS NOT ZERO
BR TST27 ; GO ON TO NEXT TEST

```

3771  
3772  
3773  
3774  
3775  
3776  
3777 015140 000004  
3778 015142 012737 000144 001200  
3779 015150 013702 001270  
3780 015154 012737 000017 004246  
3781 015162 012737 000005 004160  
3782 015170 012762 100000 000000  
3783 015176 012762 000057 000026  
3784  
3785 015204 012762 000005 000000  
3786 015212 012700 000016  
3787 015216 052762 000400 000026  
3788 015224 042762 000400 000026  
3789 015232 005300  
3790 015234 001370  
3791 015236 016237 000000 004120  
3792 015244 016237 000026 004144  
3793 015252 016237 000034 004146  
3794 015260 016237 000036 004150  
3795 015266 012737 002040 004204  
3796 015274 032737 020000 004144  
3797 015302 001403  
3798 015304 052737 020000 004204  
3799 015312 012737 000400 004206  
3800 015320 005037 004210  
3801 015324 023737 004160 004120  
3802 015332 001405  
3803 015334 104042  
3804 015336 012762 100000 000000  
3805 015344 000442  
3806  
3807 015346 023737 004204 004144  
3808 015354 001405  
3809 015356 104043  
3810 015360 012762 100000 000000  
3811 015366 000431  
3812  
3813 015370  
3814 015370 032737 000400 004146  
3815  
3816 015376 001002  
3817 015400 104044  
3818 015402 000423  
3819  
3820 015404  
3821 015404 013737 004150 001160  
3822 015412 042737 177760 001160  
3823 015420 001402  
3824 015422 104045  
3825 015424 000412  
3826

C2P6BC0 RK611 DSKLS CTRL PRT2  
C2P6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T26

02-DEC-77 09:31 PAGE 73  
MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 2)

SEQ 0073

```

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 ;MESSAGE 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 TST27: SCOPE
3845 MOV #100, $TIMES ;DO 100. ITERATIONS
3846 MOV $BASE, R2 ;LOAD RK611 BASE
3847 MOV #17, MSGCOD ;LOAD MESSAGE CODE FOR PRINT OUT
3848 MOV #UNLOAD, E.CS1 ;LOAD EXPECTED CS1
3849 MOV #CCLR, RKCS1(R2) ;CLEAR RK611
3850 MOV #DMC!17, RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3851 ;SELECT MESSAGE 17
3852 015516 012762 000007 000000 10V #UNLOAD, RKCS1(R2) ;ISSUE UNLOAD
3853 015524 012700 000016 MOV #3*4+2, R0 ;CLOCK IN DRIVE MESSAGE
3854 015530 052762 000400 000026 1$: 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!DMC, 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 10$: BIS #ECCW, E.MR1
3866 015624 012737 002000 004206 MOV #S.UNLD, E.MR2 ;LOAD EXPECTED MAINT REG. 2
3867 015632 005037 004210 004120 CLR E.MR3 ;LOAD EXPECTED MAINT REG. 3
3868 015636 023737 004160 004120 CMP E.CS1, T.CS1 ;CHECK IF CS1 CORRECT
3869 015644 001405 BEQ 2$ ;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 032737 002000 004146 3$: BIT #S.UNLD, T.MR2 ;CHECK IF UNLOAD COMMAND
3881 015702 BIT ;BIT SET
3882

```

```

3883 015710 001002      BNE      4$      ;YES, CHECK MESSAGE SELECT BITS
3884 015712 104044      ERROR    44      ;S.UNLD BIT NOT SET
3885 015714 000423      BR       TST30    ;GO ON TO NEXT TEST
3886
3887 015716      4$:      MOV      T.MR3,$TMP0 ;MASK OUT BITS NOT UNDER TEST
3888 015716 013737 004150 001160      BIC      #177760,$TMP0
3889 015724 042737 177760 001160      BEQ      5$      ;CHECK IF MESSAGE SELECT ZERO
3890 015732 001402      ERROR    45      ;MESSAGE SELECT BITS NOT ZERO
3891 015734 104045      BR       TST30    ;GO ON TO NEXT TEST
3892 015736 000412
3893
3894 015740 023737 004206 004146 5$:      CMP      E.MR2,T.MR2 ;CHECK IF MESSAGE A CORRECT
3895 015746 001401      BEQ      6$      ;YES, CHECK MESSAGE B
3896 015750 104046      ERROR    46      ;MESSAGE A INCORRECT
3897 015752 023737 004210 004150 6$:      CMP      E.MR3,T.MR3 ;CHECK IF MESSAGE B CORRECT
3898 015760 001401      BEQ      TST30    ;YES, GO ON TO NEXT TEST
3899 015762 104047      ERROR    47      ;MESS B INCORRECT
3900
3901 *****
3902 *TEST 30 MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 4)
3903 *
3904 *      CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER IN
3905 *      DIAGNOSTIC MODE WITH MESSAGE SELECT BITS = 17. LOAD
3906 *      COMMAND AND STATUS REGISTER 1 WITH A START SPINDLE.
3907 *      CLOCK MESSAGE TO LOAD B SHIFT REG. TIME. MAKE SURE
3908 *      MESSAGE SELECT BITS ARE CLEARED.
3909 *
3910 *****
3911 015764 000004      TST30:  SCOPE
3912 015766 012737 000144 001200      MOV      #100,$TIMES ;DO 100. ITERATIONS
3913 015774 013702 001270      MOV      $BASE,R2 ;LOAD RK611 BASE
3914 016000 012737 000017 004246      MOV      #17,$MSGCOD ;LOAD MESSAGE CODE FOR PRINT OUT
3915 016006 012737 000011 004160      MOV      #SRTSPL,E.CS1 ;LOAD EXPECTED CS1
3916 016014 012762 100000 000000      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
3917 016022 012762 000057 000026      MOV      #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3918                                     ;SELECT MESSAGE 17
3919 016030 012762 000011 000000      MOV      #SRTSPL,RKCS1(R2) ;ISSUE SRTSPL
3920 016036 012700 000016      MOV      #3*4+2,R0 ;CLOCK IN DRIVE MESSAGE
3921 016042 052762 000400 000026 1$:      BIS      #MCLK,RKMR1(R2)
3922 016050 042762 000400 000026      BIC      #MCLK,RKMR1(R2)
3923 016056 005300      DEC      R0
3924 016060 001370      BNE      1$
3925 016062 016237 000000 004120      MOV      RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
3926 016070 016237 000026 004144      MOV      RKMR1(R2),T.MR1 ;STORE MAINTENANCE REG. 1
3927 016076 016237 000034 004146      MOV      RKMR2(R2),T.MR2 ;STORE MAINTENANCE REG.2
3928 016104 016237 000036 004150      MOV      RKMR3(R2),T.MR3 ;STORE MAINTENANCE REG. 3
3929 016112 012737 002040 004204      MOV      #MEWD!DMD,E.MR1 ;LOAD EXPECTED MAINT REG. 1
3930 016120 032737 020000 004144      BIT      #ECCW,T.MR1
3931 016126 001403      BEQ      10$
3932 016130 052737 020000 004204      BIS      #ECCW,E.MR1
3933 016136 012737 000100 004206 10$:      MOV      #S.STSP,E.MR2 ;LOAD EXPECTED MAINT REG. 2
3934 016144 005037 004210      CLR      E.MR3 ;LOAD EXPECTED MAINT REG. 3
3935 016150 023737 004160 004120      CMP      E.CS1,T.CS1 ;CHECK IF CS1 CORRECT
3936 016156 001405      BEQ      2$      ;YES, CHECK MAINT REG. 1
3937 016160 104042      ERROR    42
3938 016162 012762 100000 000000      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER FOR NEXT TEST

```

K06

CZR6BCD RK611 DSKLS CTRL FRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T3002-DEC-77 09:31 PAGE 75  
MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 4)

SEQ 0075

```

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 032737 000100 004146 3$: BIT #S.STSP,T.MR2 ;CHECK IF SRTSPL COMMAND
3948 016214 032737 000100 004146 ; BIT SET
3949
3950 016222 001002 BNE 4$ ;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 013737 004150 001160 4$: MOV T.MR3,$TMPD ;MASK OUT BITS NOT UNDER TEST
3955 016230 042737 177760 001160 BIC #177760,$TMPD
3956 016236 001402 BEQ 5$ ;CHECK IF MESSAGE SELECT ZERO
3957 016244 104045 ERROR 45 ;MESSAGE SELECT BITS NOT ZERO
3958 016246 000412 BR TST31 ;;GO ON TO NEXT TEST
3959
3960 016252 023737 004206 004146 5$: CMP E.MR2,T.MR2 ;CHECK IF MESSAGE A CORRECT
3961 016260 001401 BEQ 6$ ;YES, CHECK MESSAGE B
3962 016262 104046 ERROR 46 ;MESSAGE A INCORRECT
3963 016264 023737 004210 004150 6$: CMP E.MR3,T.MR3 ;CHECK IF MESSAGE B CORRECT
3964 016272 001401 BEQ TST31 ;YES, GO ON TO NEXT TEST
3965 016274 104047 ERROR 47 ;MESS B INCORRECT
3966
3967
3968
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978 016276 000004 TST31: SCOPE
3979 016300 012737 000144 001200 MOV #100,$TIMES ;DO 100. ITERATIONS
3980 016306 013702 001270 004246 MOV $BASE,R2 ;LOAD RK611 BASE
3981 016312 012737 000017 004246 MOV #17,MSGCOD ;LOAD MESSAGE CODE FOR PRINT OUT
3982 016320 012737 000013 004160 MOV #RECAL,E.CS1 ;LOAD EXPECTED CS1
3983 016326 012762 100000 000000 MOV #CCLR,RKCS1(R2) ;CLEAR RK611
3984 016334 012762 000057 000026 MOV #DMD!17,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
3985
3986 016342 012762 000013 000000 MOV #RECAL,RKCS1(R2) ;SELECT MESSAGE 17
3987 016350 012700 000016 000026 1$: MOV #3*4+2,R0 ;ISSUE RECAL
3988 016354 052762 000400 000026 BIS #MCLK,RKMR1(R2) ;CLOCK IN DRIVE MESSAGE
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

```

```

*****
*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.
*****

```

L06

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T3102-DEC-77 09:31 PAGE 76  
MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 5)

SEQ 0076

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\$: BIT	#S.RECL,T.MR2	;CHECK IF RECAL COMMAND
4015	016526	032737	000040	004146			BIT SET
4016					BNE	4\$	;YES, CHECK MESSAGE SELECT BITS
4017	016534	001002			ERROR	44	;S.RECL BIT NOT SET
4018	016536	104044			BR	TST32	;GO ON TO NEXT TEST
4019	016540	000423					
4020							
4021	016542				4\$: MOV	T.MR3,\$TMP0	;MASK OUT BITS NOT UNDER TEST
4022	016542	013737	004150	001160	BIC	#177760,\$TMP0	
4023	016550	042737	177760	001160	BEQ	5\$	;CHECK IF MESSAGE SELECT ZERO
4024	016556	001402			ERROR	45	;MESSAGE SELECT BITS NOT ZERO
4025	016560	104045			BR	TST32	;GO ON TO NEXT TEST
4026	016562	000412					
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							
4037							
4038							
4039							
4040							
4041							
4042							
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

```

*****
*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.
*****

```

## MO6

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T3202-DEC-77 09:31 PAGE 77  
MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 6)

SEQ 0077

```

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

```

# NO6

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046)  
T33

02-DEC-77 09:31 PAGE 78  
MESSAGE SELECT BIT CLEARING FOR CLASS A (PART 7)

SEQ 0078

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	022737	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	000442			BR	TST34	;GO ON TO NEXT TEST
4140							
4141	017336				3\$: BIT	#S.SEEK,T.MR	;CHECK IF SEEK COMMAND
4142	017336	032737	000020	004146			;BIT SET
4143							
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\$: MOV	T.MR3,\$TMP0	;MASK OUT BITS NOT UNDER TEST
4149	017352	013737	004150	001160	BIC	#177760,\$TMP0	
4150	017360	042737	177760	001160	BEQ	5\$	;CHECK IF MESSAGE SELECT ZERO
4151	017366	001402			ERROR	45	;MESSAGE SELECT BITS NOT ZERO
4152	017370	104045			BR	TST34	;GO ON TO NEXT TEST
4153	017372	000412					
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							

```

15:  SCOPE
      MOV      #DOO, $TIMES          ; DO 100. ITERATIONS
      MOV      $BASE, R2             ; LOAD RK611 BASE
      MOV      #CCLR, RKCS1(R2)      ; CLEAR RK611
      CLR      SFTCNT                ; INITIALIZE SHIFT COUNT
      MOV      #5, E.MR2             ; LOAD EXPECTED SHIFT REG. A
      MOV      #5, U.MR2             ; LOAD UNSHIFTED SHIFT REG. A
      MOV      #3, E.MR3             ; LOAD EXPECTED SHIFT REG. B
      MOV      #3, U.MR3             ; LOAD UNSHIFTED SHIFT REG. B
      MOV      #DMD!3, RKMR1(R2)     ; PUT RK611 IN MAINT. MODE
      ; MESSAGE SELECT = 3
      MOV      #5, RKCS2(R2)         ; LOAD DRIVE NUMBER = 5
      MOV      #SELDIV, RKCS1(R2)    ; ISSUE SELECT DRIVE
      MOV      #3*4+2, RO            ; CLOCK IN MESSAGE
      BIS      #MCLK, RKMR1(R2)      ; ISSUE CLOCKS
      BIC      #MCLK, RKMR1(R2)
      DEC      RO
      BNE      1$
      MOV      RKMR2(R2), T.MR2      ; STORE SHIFT REG. A
      MOV      RKMR3(R2), T.MR3      ; STORE SHIFT REG. B
      CMP      E.MR2, T.MR2          ; CHECK SHIFT REG A CORRECT
      BEQ      2$                    ; YES, CHECK SHIFT REG. B
      ERROR    50                     ; SHIFT REG A INCORRECT
      BR       TST35                 ; GO ON TO NEXT TEST

25:  CMP      E.MR3, T.MR3           ; CHECK SHIFT REG B CORRECT
      BEQ      3$                    ; YES, SHIFT A BIT
      ERROR    51                     ; SHIFT REG B INCORRECT
      BR       TST35                 ; GO ON TO NEXT TEST

35:  BIT      #BIT0, E.MR3           ; CHECK IF SHIFT BIT = 1
      BEQ      4$                    ; NO, CLEAR SHIFT BIT
      SEC      ; SET SHIFT BIT
      BR       5$                    ; GENERATE EXPECTED SHIFT
      ; REGISTERS A & B

45:  CLC
55:  ROR      E.MR2                   ; CLEAR SHIFT BIT
      ROR      E.MR3                   ; GENERATE EXPECTED SHIFT REG A
      MOV      #4, RO                 ; GENERATE EXPECTED SHIFT REG B
      INC      SFTCNT                 ; LOAD COUNT FOR 1 BIT SHIFT
      CMP      #4, SFTCNT             ; INCREMENT SHIFT BIT COUNT
      BHS      1$                     ; CHECK IF FINISHED
      ; NO, SHIFT IN NEXT BIT

```



4219  
4220  
4221  
4222  
4223  
4224  
4225  
4226  
4227  
4228  
4229  
4230  
4231  
4232  
4233  
4234  
4235  
4236  
4237  
4238  
4239  
4240  
4241  
4242  
4243  
4244  
4245  
4246  
4247  
4248  
4249  
4250  
4251  
4252  
4253  
4254  
4255  
4256  
4257  
4258  
4259  
4260  
4261  
4262  
4263  
4264  
4265  
4266  
4267  
4268  
4269  
4270  
4271  
4272  
4273  
4274

017656	000004		
017660	012737	000144	001200
017666	013702	001270	
017672	012762	100000	000000
017700	005037	004256	
017704	012737	011020	004206
017712	012737	011020	004230
017720	012737	011020	004210
017726	012737	011020	004232
017734	012762	000040	000026
017742	012762	000441	000020
017750	012762	000400	000006
017756	012762	010017	000000
017764	012700	000016	
017770	052762	000400	000026
017776	042762	000400	000026
020004	005300		
020006	001370		
020010	016237	000034	004146
020016	016237	000036	004150
020024	023737	004206	004146
020032	001402		
020034	104050		
020036	000431		
020040	023737	004210	004150
020046	001402		
020050	104051		
020052	000423		
020054	032737	000001	004210
020062	001402		
020064	000261		
020066	000401		
020070	000241		
020072	006037	004206	
020076	006037	004210	
020102	012700	000004	
020106	005237	004256	
020112	022737	000010	004256
020120	103323		

```

*****
*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.
*****
TST35:  SCOPE
        MOV      #100,STIMES      ;;DO 100. ITERATIONS
        MOV      $BASE,R2        ;;LOAD RK611 BASE
        MOV      @CCL,RKCS1(R2)  ;;CLEAR RK611
        CLR      SFTCNT          ;;INITIALIZE SHIFT COUNT
        MOV      #S.FMT!S.SEEK!BIT12,E.MR2  ;;LOAD EXPECTED SHIFT REG. A
        MOV      #S.FMT!S.SEEK!BIT12,U.MR2  ;;LOAD UNSHIFTED SHIFT REG. A
        MOV      #11020,E.MR3     ;;LOAD EXPECTED SHIFT REG. B
        MOV      #11020,U.MR3     ;;LOAD UNSHIFTED SHIFT REG. B
        MOV      @DMO,RKMR1(R2)   ;;PUT RK611 IN MAINT. MODE
        MOV      #441,RKDCYL(R2)  ;;LOAD CYLINDER ADD. REG.
        MOV      #400,RKDA(R2)    ;;LOAD DISK ADDRESS REG.
        MOV      @SEEK!CFMT,RKCS1(R2) ;;ISSUE SEEK
        MOV      #3*4+2,R0        ;;CLOCK IN MESSAGE
        BIS      @MCLK,RKMR1(R2)  ;;ISSUE CLOCKS
        BIC      @MCLK,RKMR1(R2)
        DEC      R0
        BNE      1$
        MOV      RKMR2(R2),T.MR2  ;;STORE SHIFT REG. A
        MOV      RKMR3(R2),T.MR3  ;;STORE SHIFT REG. B
        CM?      E.MR2,T.MR2      ;;CHECK SHIFT REG. A CORRECT
        BEQ      2$               ;;YES CHECK SHIFT REG. B
        ERROR    50               ;;SHIFT REG A INCORRECT
        BR       TST36            ;;GO ON TO NEXT TEST
        CMP      E.MR3,T.MR3      ;;CHECK SHIFT REG B CORRECT
        BEQ      3$               ;;YES SHIFT A BIT
        ERROR    51               ;;SHIFT REG B INCORRECT
        BR       TST36            ;;GO ON TO NEXT TEST
        BIT      @BIT0,E.MR3      ;;CHECK IF SHIFT BIT = 1
        BEQ      4$               ;;NO CLEAR SHIFT BIT
        SEC      SEC              ;;SET SHIFT BIT
        BR       5$               ;;GENERATE EXPECTED SHIFT
                                   ;;REGISTERS A & B
        CLC
        ROR      E.MR2            ;;CLEAR SHIFT BIT
        ROR      E.MR3            ;;GENERATE EXPECTED SHIFT REG A
        ROR      E.MR3            ;;GENERATE EXPECTED SHIFT REG B
        MOV      #4,R0            ;;LOAD COUNT FOR 1 BIT SHIFT
        INC      SFTCNT           ;;INCREMENT SHIFT BIT COUNT
        CMP      @B.,SFTCNT       ;;CHECK IF FINISHED
        BHS      1$              ;;NO, SHIFT IN NEXT BIT

```

```

4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286 020122 000004
4287 020124 012737 000144 001200
4288 020132 013702 001270
4289 020136 012762 100000 000000
4290 020144 012762 000040 000026
4291 020152 012762 000001 000000
4292 020160 012700 000116
4293 020164 012762 000440 000026
4294 020172 012762 000040 000026
4295 020200 005300
4296 020202 00170
4297 020204 016237 000034 004146
4298 020212 016237 000036 004150
4299 020220 012737 100000 004206
4300 020226 012737 100000 004210
4301 020234 032737 100000 004150
4302
4303 020242 001002
4304 020244 104052
4305 020246 000420
4306
4307 020250 032737 100000 004146 2$:
4308 020256 001002
4309 020260 104053
4310 020262 000412
4311
4312 020264 023737 004210 004150 3$:
4313 020272 001401
4314 020274 104054
4315 020276 023737 004206 004146 4$:
4316 020304 001401
4317 020306 104055
4318 020310 012762 100000 000000 5$:
4319 020316 012762 000060 000026
4320
4321 020324 012762 000001 000000
4322 020332 012700 000116
4323 020336 012762 000460 000026 6$:
4324 020344 012762 000060 000026
4325 020352 005300
4326
4327 020354 001370
4328 020356 016237 000034 004146
4329 020364 016237 000036 004150
4330 020372 005037 004206

```

\*\*\*\*\*  
TEST 36 DRIVE MESSAGE PARITY PRECONDITIONING  
\*\*\*\*\*  
CLEAR RK611 WITH CONTROLLER CLEAR. PUT CONTROLLER  
IN DIAGNOSTIC MODE. LOAD COMMAND AND STATUS REGISTER 1 WITH  
A SELECT COMMAND. CLOCK ALL 16 BITS THROUGH THE  
DRIVE MESSAGE LOOPBACK. VERIFY PARITY HAS BEEN PRECONDITIONED  
PROPERLY. REPEAT FOR BAD PARITY GENERATION.  
\*\*\*\*\*  
↑ST36: SCOPE  
MOV #100, \$TIMES ; DO 100 ITERATIONS  
MOV \$BASE, R2 ; LOAD RK611 BASE  
MOV #CCLR, RKCS1(R2) ; CLEAR RK611  
MOV #DMD, RKMR1(R2) ; PUT RK611 IN MAINTENANCE MODE  
MOV #SELD, RKCS1(R2) ; ISSUE SELECT DRIVE  
MOV #19.\*4+2, R0 ; LOAD DRIVE MESSAGE AND SHIFT  
1\$: MOV #DMD!MCLK, RKMR1(R2) ; ALL 16 BITS  
MOV #DMD, RKMR1(R2)  
DEC R0  
BNE 1\$  
MOV RKMR2(R2), T.MR2 ; STORE SHIFTED MESSAGE B  
MOV RKMR3(R2), T.MR3 ; STORE SHIFTED MESSAGE A  
MOV #100000, E.MR2 ; LOAD EXPECTED MESSAGE B  
MOV #100000, E.MR3 ; LOAD EXPECTED MESSAGE A  
BIT #BIT15, T.MR3 ; CHECK IF PARITY ON MESSAGE A CORRECT  
BNE 2\$ ; YES, CHECK PARITY ON MESSAGE B  
ERROR 52 ; PARITY ON MESSAGE A INCORRECT  
BR 5\$ ; TRY EVEN PARITY  
2\$: BIT #BIT15, T.MR2 ; CHECK IF PARITY ON MESS B CORRECT  
BNE 3\$ ; YES, CHECK MESSAGE A AND B  
ERROR 53 ; PARITY ON MESSAGE B INCORRECT  
BR 5\$ ; TRY EVEN PARITY  
3\$: CMP E.MR3, T.MR3 ; CHECK IF MESSAGE A CORRECT  
BEQ 4\$ ; YES, CHECK MESSAGE B  
ERROR 54 ; MESSAGE A INCORRECT  
4\$: CMP E.MR2, T.MR2 ; CHECK IF MESSAGE B CORRECT  
BEQ 5\$ ; YES, TRY EVEN PARITY  
ERROR 55 ; MESSAGE B INCORRECT  
5\$: MOV #CCLR, RKCS1(R2) ; CLEAR RK611  
MOV #DMD!PAT, RKMR1(R2) ; PUT RK611 MAINTENANCE MODE  
; AND EVEN PARITY  
MOV #SELD, RKCS1(R2) ; ISSUE SELECT DRIVE  
MOV #19.\*4+2, R0 ; LOAD DRIVE MESSAGE AND SHIFT  
6\$: MOV #DMD!PAT!MCLK, RKMR1(R2) ; ALL 16 BITS  
MOV #DMD!PAT, RKMR1(R2)  
DEC R0  
BNE 6\$  
MOV RKMR2(R2), T.MR2 ; STORE SHIFTED MESSAGE B  
MOV RKMR3(R2), T.MR3 ; STORE SHIFTED MESSAGE A  
CLR E.MR2 ; LOAD EXPECTED MESSAGE B

E07

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T3602-DEC-77 09:31 PAGE 82  
DRIVE MESSAGE PARITY PRECONDITIONING

SEQ 0082

```

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
4351
4352
4353
4354
4355
4356
4357
4358
4359
4360
4361 020456 000004          ;*****
4362 020460 012737 000144 001200  ;TEST 37      ODD DRIVE MESSAGE PARITY GENERATION
4363 020466 013702 001270          ;
4364 020472 012737 000001 004244  ;
4365 020500 012737 020506 001110  ;
4366
4367
4368
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4380
4381
4382
4383
4384
4385
4386

```

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.

```

;*****
TST37:  SCOPE
        MOV      #100,$TIMES      ;DO 100. ITERATIONS
        MOV      $BASE,R2        ;LOAD RK611 BASE
        MOV      #1,DRVCO        ;LOAD DRIVE CODE
        MOV      #1,$SLPERR       ;LOAD LOOP ON ERROR LOCATION FOR
        ; SUBTEST LOOP

1$:     MOV      #CCLR,RKCS1(R2) ;CLEAR RK611
        MOV      DRVCO,RKMR1(R2) ;LOAD MESSAGE SELECT CODE
        BIS      #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
        MOV      DRVCO,RKCS2(R2) ;LOAD DRIVE SELECT CODE
        MOV      #SELDV,RKCS1(R2) ;ISSUE SELECT DRIVE
        MOV      #19,*4+2,R0      ;LOAD DRIVE MESSAGE AND SHIFT
        BIS      #MCLK,RKMR1(R2) ; ALL 16 BITS
        BIC      #MCLK,RKMR1(R2)
        DEC      R0
        BNE      2$
        MOV      RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
        MOV      RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
        MOV      DRVCO,R1        ;DETERMINE PARITY
        MOV      #4,R3
        CLR      R4
        ROR      R1
        BCC      4$
        INC      R4
        3$:

```

```

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$: MOV     T.MR3,$TMP0 ;MASK ALL BITS EXCEPT PARITY
4398 020706 042737 077777 001160 BIC     #77777,$TMP0
4399 020714 023737 004260 001160 CMP     PARBIT,$TMP0 ;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$: MOV     T.MR2,$TMP0 ;MASK ALL BITS EXCEPT PARITY
4405 020736 042737 077777 001160 BIC     #77777,$TMP0
4406 020744 023737 004260 001160 CMP     PARBIT,$TMP0 ;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$: CMP     E.MR3,T.MR3 ;CHECK IF MESSAGE A CORRECT
4412 020766 001401          BEQ      6$ ;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          SCOPE1    ;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 *ST40: SCOPE
4434 021022 000004          MOV     #100,$TIMES ;DO 100. ITERATIONS
4435 021024 012737 000144 001200 MOV     $BASE,R2 ;LOAD RK611 BASE
4436 021032 013702 001270          MOV     #1,DRVCOD ;SET INITIAL DRIVE SELECT CODE
4437 021036 012737 000001 004244 MOV     MSGCOD ;SET INITIAL MESSAGE SELECT CODE
4438 021044 005037 004246          CLR     MSGCOD
4439 021050 012737 100000 004206 MOV     #BIT15,E.MR2 ;LOAD EXPECTED MAINT. REG. 2 (MESS B)
4440 021056 012737 000001 004210 MOV     #BIT0,E.MR3 ;LOAD EXPECTED MAINT. REG. 3 (MESS A)
4441 021064 012737 100000 004260 MOV     #BIT15,PARBIT ;LOAD PARITY FOR MESSAGE B
4442 021072 012737 021100 001110 MOV     #1,$SLPERR ;LOAD LOOP ON ERROR LOCATION FOR
; SUBTEST LOOP

```

```

4443
4444 021107 012762 100000 000000 1$: MOV #CCLR,RKCS1(R2) ;CLEAR RK611
4445 021108 013762 004246 000026 MOV MSGCOD,RKMR1(R2) ;LOAD MESSAGE SELECT CODE
4446 021109 052762 000040 000026 BIS #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
4447 021114 013762 004244 000010 MOV DPVCOD,RKCS2(R2) ;LOAD DRIVE SELECT CODE
4448 021122 012762 000001 000000 MOV #SELDIV,RKCS1(R2) ;ISSUE DRIVE SELECT
4449 021130 012700 000116 000026 2$: MOV #19.*4+2,R0 ;LOAD DRIVE MESSAGE AND SHIFT
4450 021142 052762 000400 000026 BIS #MCLK,RKMR1(R2) ; ALL 16 BITS
4451 021150 042762 000400 000026 BIC #MCLK,RKMR1(R2)
4452 021156 005300 DEC R0
4453 021160 001370 BNE 2$
4454 021162 016237 000034 004146 MOV RKMR2(R2),T.MR2 ;STORE SHIFTED MESSAGE B
4455 021170 016237 000036 004150 MOV RKMR3(R2),T.MR3 ;STORE SHIFTED MESSAGE A
4456 021176 013737 004150 001160 MOV T.MR3,$TMP0 ;MASK ALL BITS EXCEPT PARITY
4457 021204 042737 077777 001160 BIC #7777,$TMP0
4458 021212 023737 004260 001160 CMP PARBIT,$TMP0 ;CHECK IF PARITY BIT CORRECT
4459 021220 001002 3$ ; ON MESSAGE A
4460 021222 104052 52 ;NO PARITY ON MESSAGE INCORRECT
4461 021224 000426 BR 25$ ;CHECK IF LOOP ON ERROR
4462
4463
4464 021226 013737 004146 001160 3$: MOV T.MR2,$TMP0 ;MASK ALL BITS EXCEPT PARITY
4465 021234 042737 077777 001160 BIC #7777,$TMP0
4466 021242 023737 004260 001160 CMP PARBIT,$TMP0 ;CHECK IF PARITY CORRECT
4467 021250 001402 4$ ; MESSAGE B
4468 021252 104053 53 ;PARITY ON MESSAGE B INCORRECT
4469 021254 000412 BR 25$ ;CHECK IF LOOP ON ERROR
4470
4471 021256 023737 004210 004150 4$: CMP E.MR3,T.MR3 ;CHECK IF MESSAGE A CORRECT
4472 021264 001401 5$ ;YES CHECK IN MESSAGE B CORRECT
4473 021266 104054 54 ;MESSAGE A INCORRECT
4474 021270 023737 004206 004146 5$: CMP E.MR2,T.MR2 ;CHECK IF MESSAGE B CORRECT
4475 021276 001401 25$ ;YES CHECK IF LOOP ON ERROR
4476 021300 104055 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 *****

```

H07

CZP6BC0 RK611 DSKLS CTRL PR\*2  
CZP6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T4102-DEC-77 09:31 PAGE 85  
EVEN DRIVE MESSAGE PARITY GENERATION

SEQ 0085

```

4499 021346 000004 TST41: SCOPE
4500 021350 012737 000144 001200 MOV #100, $TIMES ; DO 100. ITERATIONS
4501 021356 013702 001270 004244 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 #SELD, RKCS1(R2) ; ISSUE SELECT DRIVE
4513 021434 012700 000116 000000 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 000004 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

```

```

4555 021672 104061          ERROR 61          :MESSAGE B INCORRECT
4556 021674 104415          SCOP1          :CHECK IF LOOP ON ERROR
4557 021676 005237 004244    INC  DRVCOD      :INCREMENT DRIVE SELECT CODE
4558 021702 022737 000017 004244  CMP  #17,DRVCOD :CHECK IF FINISHED
4559 021710 103232          BHIS  1$          :NO, TRY NEXT CONFIGURATION
4560
4561
4562 .SBTTL **CLASS A COMMAND EXECUTION
4563
4564 :*****
4565 :*TEST 42          RELEASE COMMAND IN DIAGNOSTIC MODE
4566 :*
4567 :*          CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
4568 :*          PUT CONTROLLER IN DIAGNOSTIC MODE. LOAD COMMAND AND
4569 :*          STATUS REGISTER 2 WITH DRIVE SELECT = 10. LOAD
4570 :*          COMMAND AND STATUS REGISTER 1 WITH A SELECT.
4571 :*          CLOCK COMMAND TO COMPLETION. MAKE SURE UNIT
4572 :*          FIELD ERROR DOES NOT SET (SACK HIGH). REPEAT FOR
4573 :*          DRIVE SELECT = 11-17.
4574 :*
4575 :*****
4576 021712 000004          ST42: SCOPE
4577 021714 012737 000144 001200    MOV  #100.,$TIMES          ;;DO 100. ITERATIONS
4578 021722 013702 001270          MOV  $BASE,R2          ;;LOAD RK611 BASE
4579 021726 012737 000010 004244    MOV  #10,DRVCOD        ;;INITIALIZE FOR DESELECT OF DRIVE 0
4580 021734 012737 021742 001110    MOV  #1$, $LPERR        ;;LOAD LOOP ON ERROR LOCATION FOR
4581 :          SUBTEST LOOP
4582
4583 021742          1$:
4584 021742 012762 000040 000010    MOV  #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
4585 021750 012762 000040 000026    MOV  #OMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
4586 021756 013762 004244 000010    MOV  DRVCOD,RKCS2(R2) ;LOAD DRIVE SELECTION
4587 021764 012762 000001 000000    MOV  #SELDV,RKCS1(R2) ;ISSUE DESELECT
4588 021772 012700 000120          MOV  #20.*4,R0          ;;LOAD COUNT TO COMPLETE COMMAND
4589 021776 012762 000440 000026    MOV  #OMD:MCLK,RKMR1(R2) ;CLOCK THRU COMMAND
4590 022004 012762 000040 000026    MOV  #OMD,RKMR1(R2)
4591 022012 005300          DEC  R0
4592 022014 001370          BNE  2$
4593 022016 016237 000000 004120    MOV  RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4594 022024 016237 000010 004130    MOV  RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
4595 022032 016237 000012 004132    MOV  RKDS(R2),T.DS   ;STORE DRIVE STATUS REGISTER
4596 022040 016237 000014 004134    MOV  RKER(R2),T.ER   ;STORE ERROR REGISTER
4597 022046 012737 000200 004160    MOV  #RDY,E.CS1     ;LOAD EXPECTED COMMAND AND STATUS REG. 1
4598 022054 013737 004244 004170    MOV  DRVCOD,E.CS2     ;GENERATE EXPECTED COMMAND AND
4599 022062 052737 000100 004170    BIS  #IR,E.CS2        ;STATUS REG. 2
4600 022070 005037 004172          CLR  E.DS             ;LOAD EXPECTED DRIVE STATUS REGISTER
4601 022074 005037 004174          CLR  E.ER             ;LOAD EXPECTED ERROR REGISTER
4602 022100 023737 004160 004120    CMP  E.CS1,T.CS1     ;CHECK COMMAND AND STATUS REG 1 CORRECT
4603 022106 001401          BEQ  3$          ;YES, CHECK CS2
4604 022110 104062          ERROR 62          ;COMMAND AND STATUS REG. 1 INCORRECT
4605 022112 023737 004170 004130    CMP  E.CS2,T.CS2     ;CHECK COMMAND AND STATUS REG. 2 CORRECT
4606 022120 001401          BEQ  4$          ;YES, CHECK ERROR REGISTER
4607 022122 104063          ERROR 63          ;COMMAND AND STATUS REG. 2 INCORRECT
4608 022124 023737 004174 004134    CMP  E.ER,T.ER       ;CHECK ERROR REGISTER CORRECT
4609 022132 001401          BEQ  5$          ;YES, CHECK DRIVE STATUS REG
4610 022134 104064          ERROR 64          ;ERROR REGISTER INCORRECT

```

CZR6BC0 RK611 DSKLS CTRL PRT2 MACY11 30(1046) 02-DEC-77 09:31 PAGE 87  
 CZR6BC.P11 02-DEC-77 09:22 T42 RELEASE COMMAND IN DIAGNOSTIC MODE

SEQ 0087

```

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 1$T43: SCOPE
4632 022170 012737 000144 001200 MOV #100,$TIMES ;DO 100. ITERATIONS
4633 022176 013702 001270 MOV $BASE,R2 ;LOAD RK611 BASE
4634 022202 005037 004244 CLR DRVCOD ;INITIALIZE FOR SELECT OF DRIVE 0
4635 022206 012737 022214 001110 MOV #1$,SLPERR ;LOAD LOOP ON ERROR LOCATION FOR
4636 : SUBTEST LOOP
4637
4638 022214 1$: MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
4639 022214 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
4640 022222 012762 000040 000026 MOV DRVCOD,RKCS2(R2) ;LOAD DRIVE SELECT
4641 022230 013762 004244 000010 MOV #SELDV,RKCS1(R2) ;ISSUE DRIVE SELECT
4642 022236 012762 000001 000000 MOV #20,*4,R0 ;LOAD COUNT TO DESELECT COMPLETE
4643 022244 012700 000120 2$: MOV #DMD!MCLK,RKMR1(R2) ;CLOCK UNTIL DESELECT FINISHED
4644 022250 012762 000440 000026 MOV #DMD,RKMR1(R2)
4645 022256 012762 000040 000026 DEC R0
4646 022264 005300 BNE 2$
4647 022266 001370 2$: MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4648 022270 016237 000000 004120 MOV #SELDV,E.CS1 ;LOAD EXPECTED COMMAND AND STATUS REG. 1
4649 022276 012737 000001 004160 CMP E.CS1,T.CS1 ;CHECK IF READY RESET
4650 022304 023737 004160 004120 BEQ 3$ ;YES, CONTINUE COMMAND
4651 022312 001402 ERROR 6$ ;COMMAND AND STATUS REG. 1 INCORRECT
4652 022314 104065 BR 25$ ;GO CHECK IF LOOP ON ERROR
4653 022316 000566
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

```



K07

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T4302-DEC-77 09:31 PAGE 88  
SELECT COMMAND IN DIAGNOSTIC MODE

SEQ 0088

4667	022366	012737	100000	004206	6\$:	MOV	#BIT15,E.MR2	;STORE EXPECTED MAINT REG 2
4668	022374	012701	000003			MOV	#3,R1	;ISSUE 3 CONTR. CLOCKS
4669	022400	012700	000004		7\$:	MOV	#4,RO	
4670	022404	012762	000440	000026	8\$:	MOV	#DMD!MCLK,RKMR1(R2)	
4671	022412	012762	000040	000026		MOV	#DMD,RKMR1(R2)	
4672	022420	005300				DEC	RO	
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,RO	;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	RO	
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

4723 022710 103402  
4724 022712 000137 022214BLO 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

:\*\*\*\*\*

4735  
4736 022716 000004  
4737 022720 012737 000144 001200  
4738 022726 013702 001270  
4739 022732 012737 000010 004244  
4740 022740 012737 022746 001110TST44: 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 LOOP4741  
4742  
4743 022746  
4744 022746 012762 000040 000010  
4745 022754 013762 004244 000010  
4746 022762 012762 000001 000000  
4747 022770 013700 004262  
4748 022774 105762 0000001\$: MOV #SCLR, RKCS2(R2) ;:CLEAR RK06 SUBSYSTEM  
MOV DRVCOD, RKCS2(R2) ;:LOAD DRIVE SELECTION  
MOV #SELDIV, RKCS1(R2) ;:ISSUE DESELECT  
MOV WAITIM, R0 ;:WAIT FOR READY4749 023000 100402  
4750 023002 005300  
4751 023004 0013732\$: TSTB RKCS1(R2)  
BMI 3\$  
DEC R0  
BNE 2\$4752 023006 016237 000000 004120  
4753 023014 016237 000010 004130  
4754 023022 016237 000012 004132  
4755 023030 016237 000014 004134  
4756 023036 012737 000200 004160  
4757 023044 013737 004244 004170  
4758 023052 052737 000100 004170  
4759 023060 005037 004172  
4760 023064 005037 004174  
4761 023070 023737 004160 0041203\$: 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 RKR(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.CS24762 023076 001401  
4763 023100 104074  
4764 023102 023737 004170 004130  
4765 023110 001401  
4766 023112 104075  
4767 023114 023737 004174 004134  
4768 023122 001401  
4769 023124 104076  
4770 023126 023737 004172 004132  
4771 023134 001401  
4772 023136 104130  
4773 023140 104415;:LOAD EXPECTED DRIVE STATUS REG  
;: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  
CMP E.CS2, T.CS2 ;:CHECK COMMAND AND STATUS REG 2 CORRECT  
BEQ 5\$ ;:YES, CHECK ERROR REGISTER  
ERROR 75 ;:CS2 INCORRECT  
CMP E.ER, T.ER ;:CHECK ERROR REG CORRECT  
BEQ 6\$ ;:YES, CHECK DRIVE STATUS REG CORRECT  
ERROR 76 ;:ERROR REG INCORRECT  
CMP E.DS, T.DS ;:CHECK DRIVE STATUS REG CORRECT  
BEQ 7\$ ;:YES, CHECK IF LOOP ON ERROR  
ERROR 130 ;:DRIVE STATUS REGISTER INCORRECT4774 023142 005237 004244  
4775 023146 022737 000017 004244  
4776 023154 1032747\$: SCOP1  
INC DRVCOD  
CMP #17, DRVCOD ;:CHECK IF ALL DRIVE NUMBERS TESTED  
BHS 1\$ ;:NO, DO IT FOR NEXT DRIVE

:\*\*\*\*\*

MO7

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 90  
T45 INTERRUPT AT COMMAND COMPLETION

SEQ 0090

```

4779          ;*TEST 45          INTERRUPT AT COMMAND COMPLETION
4780          ;*
4781          ;* CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
4782          ;* LOWER PROCESSOR PRIORITY TO ZERO. ISSUE A RELEASE
4783          ;* COMMAND WITH INTERRUPT ENABLE SET. MAKE SURE
4784          ;* INTERRUPT OCCURS. LOWER PRIORITY AFTER INTERRUPT
4785          ;* AND MAKE SURE INTERRUPT HAS CLEARED.
4786          ;*
4787          ;* LOWER PROCESSOR PRIORITY TO ZERO. REISSUE RELEASE
4788          ;* WITH INTERRUPT ENABLE RESET. MAKE SURE NO INTERRUPT
4789          ;* OCCURS. SET INTERRUPT ENABLE AND MAKE SURE NO
4790          ;* INTERRUPT OCCURS.
4791          ;*
4792          ;*****
4793          ;ST45: SCOPE
4794          023156 000004          MOV      #100.,$TIMES          ;DO 100. ITERATIONS
4795          023160 012737 000144 001270          MOV      $BASE,R2          ;LOAD RK611 BASE
4796          023166 013702 001270          MOV      $SCLR,RKCS2(R2)          ;CLEAR RK06 SUBSYSTEM
4797          023172 012762 000040 000010          MOV      #10,RKCS2(R2)          ;SET DESELECT BIT
4798          023200 012762 000010 000010          MOV      RKVEC,R1          ;LOAD INTERRUPT VECTOR
4799          023206 013701 004234          MOV      #5,$(R1)+
4800          023212 012721 023274          MOV      #PR7,(R1)
4801          023216 012711 000340          MOV      CLR      -(SP)          ;LOAD STACK TO ALLOW ALL INTERRUPTS
4802          023222 005046          MOV      #64,$-(SP)          ;LOAD NEXT ADDRESS
4803          023224 012746 023232          RTI          ;CLEAR PSW
4804          023230 000002
4805          023232
4806          023232 012762 000101 000000          64$: MOV      #SELDV!IE,RKCS1(R2)          ;ISSUE SELECT DRIVE
4807          023240 013700 004262          MOV      WAITIM,R0          ;WAIT FOR READY
4808          023244 105762 000000          2$: TSTB      RKCS1(R2)
4809          023250 100402          BMI      3$
4810          023252 005300          DEC      R0
4811          023254 001373          BNE      2$
4812          023256 012746 000340          3$: MOV      #PR7,-(SP)          ;LOCK OUT INTERRUPTS
4813          023262 012746 023270          MOV      #4,$-(SP)
4814          023266 000002          RTI
4815
4816          023270 104100          4$: ERROR      100          ;INTERRUPT DID NOT OCCUR
4817          023272 000522          BR      25$
4818
4819          023274 062706 000004          5$: ADD      #4,SP          ;ADJUST STACK
4820          023300 016237 000000 004120          MOV      RKCS1(R2),T.CS1          ;STORE COMMAND AND STATUS REG. 1
4821          023306 016237 000010 004130          MOV      RKCS2(R2),T.CS2          ;STORE COMMAND AND STATUS REG. 2
4822          023314 016237 000014 004134          MOV      RKER(R2),T.ER          ;STORE ERROR REG.
4823          023322 012737 000300 004160          MOV      #RDY!IE,E.CS1          ;LOAD EXPECTED CS1
4824          023330 012737 000110 004170          MOV      #IR!10,E.CS2          ;LOAD EXPECTED CS2
4825          023336 005037 004174          CLR      E.ER          ;LOAD EXPECTED ERROR
4826          023342 023737 004160 004120          CMP      E.CS1,T.CS1          ;CHECK IF CS1 CORRECT
4827          023350 001401          BEQ      6$          ;YES, CHECK CS2
4828          023352 104101          ERROR      101          ;CS1 INCORRECT
4829          023354 023737 004170 004130          6$: CMP      E.CS2,T.CS2          ;CHECK IF CS2 INCORRECT
4830          023362 001401          BEQ      7$          ;YES, CHECK IF ERROR REG CORRECT
4831          023364 104102          ERROR      102          ;CS2 INCORRECT
4832          023366 023737 004174 004134          7$: CMP      E.ER,T.ER          ;CHECK IF ERROR REG CORRECT
4833          023374 001401          BEQ      8$          ;YES, CHECK IF INTERRUPT CLEARED
4834          023376 104103          ERROR      103          ;ERROR REG. INCORRECT

```

NO7

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 91  
T45 INTERRUPT AT COMMAND COMPLETION

SEQ 0091

```

4835 023400 012777 023512 160626 8$: MOV #10$,ARKVEC ;LOAD VECTOR FOR UNEXPECTED INTERRUPT
4836 023406 005046 CLR -(SP) ;LOAD STACK TO ALLOW ALL INTERRUPTS
4837 023410 012746 023416 MOV #65$,-(SP) ;LOAD NEXT ADDRESS
4838 023414 000002 RTI ;CLEAR PSW
4839
4840 023416 65$: NOP ;WAIT FOR INTERRUPT
4841 023416 000240 MOV #15$,ARKVEC ;LOAD VECTOR ADDRESS FOR UNEXPECTED INTERRUPT
4842 023420 012777 023522 160606 MOV #10$,RKCS2(R2) ;ISSUE DESELECT
4843 023426 012762 000010 000010 MOV #SELDRV,RKCS1(R2)
4844 023434 012762 000001 000000 MOV WAITIM,R0
4845 023442 013700 004262 MOV RKCS1(R2)
4846 023446 105762 000000 9$: TSTB
4847 023452 100402 BMI 11$
4848 023454 005300 DEC R0
4849 023456 001373 BNE 9$
4850 023460 000240 11$: NOP ;WAIT FOR INTERRUPT
4851 023462 012777 023532 160544 MOV #20$,ARKVEC ;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 #CLR,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 $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 #SELDV,RKCS1(R2) ;ISSUE DESELECT
4892 023624 013700 004262 000000 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$: TST RKDS(R2) ;READ SILO TO MAKE IT IS CLEAR
4909 023734 016237 000000 004120 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
4910 023742 016237 000010 004130 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG. 2
4911 023750 016237 000014 004134 MOV RKER(R2),T.ER ;STORE ERROR REG.
4912 023756 012737 100200 004160 MOV #CERR!RDY,E.CS1 ;LOAD EXPECTED CS1
4913 023764 012737 100110 004170 MOV #DLT!IR!10,E.CS2 ;LOAD EXPECTED CS2
4914 023772 023737 004170 004130 CMP E.CS2,T.CS2 ;CHECK IF DATA LATE SET
4915 024000 001401 BEQ 11$ ;YES, CLEAR CONTROLLER REG. 1
4916 024002 104106 ERROR 106 ;DATA LATE NOT SET
4917 024004 012762 100000 000000 11$: MOV #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
4918
4919 *****
4920 *TEST 47 SEEK COMMAND IN DIAGNOSTIC MODE
4921 *
4922 * CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
4923 * PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK WITH CDT SET
4924 * 24 SECTOR FORMAT TO CYLINDER 1714, HEAD 7, DRIVE 0.
4925 * MAKE SURE NO STATUS BITS ARE SET AND NO ERROR
4926 * BITS ARE SET.
4927 *
4928 *****
4929 *ST47: SCOPE
4930 024012 000004 MOV #100,$TIMES ;DO 100. ITERATIONS
4931 024014 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
4932 024022 013702 001270 MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
4933 024026 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
4934 024034 012762 000040 000026 MOV #1714,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
4935 024042 012762 001714 000020 MOV #3400,RKDA(R2) ;LOAD HEAD 7
4936 024050 012762 003400 000006 MOV #SEEK!CFMT!CDT,RKCS1(R2) ;ISSUE SEEK CDT SET,24 SECTOR
4937 024056 012762 012017 000000 MOV #20,*4,R0 ;LOAD COUNT TO DESELECT DECISION
4938 024064 012700 000120 2$: MOV #DMD!MCLK,RKMR1(R2)
4939 024070 012762 000440 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$: MOV #5, SEEK!S.FMT!70000 E.MR3 ;LOAD EXPECTED MAINT REG. 3
4950 024146 012737 136300 004206 MOV #136300, E.MR2 ;LOAD EXPECTED MAINT REG. 2
4951 024154 012701 000003 MOV #3, R1 ;ISSUE 3 CONTROL CLOCKS
4952 024160 012700 000004 4$: 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 BLS 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$: DEC R1 ;CHECK IF C01 1AND FINISHED
4976 024274 001331 BNE 4$ ;NO, ISSUE AN THER 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.  
\*\*\*\*\*

TST50: SCOPE

```

MOV      #100, $TIMES      ; DO 100 ITERATIONS
$PASE, R2      ; LOAD RK611 BASE
MOV      #SCLR, RKCS2(R2) ; CLEAR RK06 SUBSYSTEM
MOV      #DMD, RKMR1(R2)  ; PUT RK611 IN MAINT MODE
MOV      #757, RKDCYL(R2) ; LOAD CYLINDER ADDRESS
MOV      #400, RKDA(R2)   ; LOAD HEAD ADD =1
MOV      #SEEK, RKCS1(R2) ; ISSUE SEEK
MOV      #22, #4+2, R0    ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
15:      MOV      #DMD!MCLK, RKMR1(R2)
MOV      #DMD, RKMR1(R2)
DEC      R0
BNE      15
CLR      RKMR1(R2)        ; FINISH COMMAND IN NORMAL MODE
MOV      WAITIM, R0       ; WAIT FOR FOR READY
25:      TSTB     RKCS1(R2)
BMI      35
DEC      R0
BNE      25
MOV      RKCS1(R2), T.CS1 ; STORE COMMAND AND STATUS REG 1
MOV      #RDY!SEEK<IC<GO>>, E.CS1 ; LOAD EXPECTED CS1
ERROR    132              ; READY NOT SET
BR       105              ; CLEAR RK06 SUBSYSTEM
35:      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 REF
MOV      RKER(R2), T.ER   ; STORE ERROR REG.
MOV      #RDY!SEEK<IC<GO>>, E.CS1 ; LOAD EXPECT CS1
MOV      #IR, E.CS2       ; LOAD EXPECTED CS2
MOV      #DRA!OFST!SPDLSS!VV!DRDY!WRL!SVAL, E.DS ; LOAD EXPECTED DRIVE STATUS
CLR      E.ER             ; LOAD EXPECTED ERROR REGISTER
BIT      #DRA!OFST!SPDLSS!VV!DRDY!WRL, T.DS ; CHECK LOAD STATUS SET
BNE      45
ERROR    133              ; YES, CHECK REGISTERS
45:      ERROR    133      ; NO BIT IN DRIVE STATUS SET
CMP      E.CS1, T.CS1      ; CHECK CS1 CORRECT
BEQ      55
ERROR    134              ; YES, CONTINUE
55:      ERROR    134      ; CS1 INCORRECT
CMP      E.CS2, T.CS2      ; CHECK CS2 CORRECT
BEQ      65
ERROR    135              ; YES, CONTINUE
65:      ERROR    135      ; CS2 INCORRECT

```

5003  
5004  
5005  
5006  
5007  
5008  
5009  
5010  
5011  
5012  
5013  
5014  
5015  
5016  
5017  
5018 024446 000004  
5019 024450 012737 000144 001200  
5020 024456 013702 001270  
5021 024462 012762 000040 000010  
5022 024470 012762 000040 000026  
5023 024476 012762 000757 000020  
5024 024504 012762 000400 000006  
5025 024512 012762 000017 000000  
5026 024520 012700 000132  
5027 024524 012762 000440 000026 15:  
5028 024532 012762 000040 000026  
5029 024540 005300  
5030 024542 001370  
5031 024544 005062 000026  
5032 024550 013700 004262  
5033 024554 105762 000000 25:  
5034 024560 100412  
5035 024562 005300  
5036 024564 001373  
5037 024566 016237 000000 004120  
5038 024574 012737 000216 004160  
5039 024602 104132  
5040 024604 000460  
5041  
5042 024606 016237 000000 004120 35:  
5043 024614 016237 000010 004130  
5044 024622 016237 000012 004132  
5045 024630 016237 000014 004134  
5046 024636 012737 000216 004160  
5047 024644 012737 000100 004170  
5048 024652 012737 104325 004172  
5049 024660 005037 004174  
5050 024664 032737 004325 004132  
5051 024672 001001  
5052 024674 104133  
5053 024676 023737 004160 004120 45:  
5054 024704 001401  
5055 024706 104134  
5056 024710 023737 004170 004130 55:  
5057 024716 001401  
5058 024720 104135

```

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 TST51 ;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 TST51: SCOPE
5104 025130 000004 MOV #100,$TIMES ;DO 100. ITERATIONS
5105 025132 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
5106 025140 013702 001270 MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5107 025144 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
5108 025152 012762 000040 000026 MOV #2,RKDCYL(R2) ;LOAD CYLINDER AND
5109 025160 012762 000002 000020 MOV #0,RKDA(R2) ;LOAD HEAD ADDRESS
5110 025166 012762 000000 000006 MOV #SEEK,RKCS1(R2) ;ISSUE SEEK
5111 025174 012762 000017 000000 MOV #22,*4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5112 025206 012762 000440 000026 1$: 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,P0	;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,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	#D.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				EEQ	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	012762	100000	000000		MOV	#CLR,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

```

S171
S172
S173
S174
S175
S176
S177
S178
S179
S180 025562 000004
S181 025564 012737 000144 001200
S182 025572 013702 001270
S183 025576 012762 000040 000010
S184 025604 012762 000040 000026
S185 025612 012762 000003 000020
S186 025620 012762 000000 000006
S187 025626 012762 000017 000000
S188 025634 012700 000132
S189 025640 012762 000440 000026 1S:
S190 025646 012762 000040 000026
S191 025654 005300
S192 025656 001370
S193 025660 005062 000026
S194 025664 013700 004262
S195 025670 105762 000000 2S:
S196 025674 100402
S197 025676 005300
S198 025700 001373
S199 025702 016237 000000 004120 3S:
S200 025710 016237 000010 004130
S201 025716 016237 000012 004132
S202 025724 016237 000014 004134
S203 025732 012737 120216 004160
S204 025740 012737 000100 004170
S205 025746 012737 100001 004172
S206 025754 012737 000000 004174
S207 025762 023737 004160 004120
S208 025770 001401
S209 025772 104144
S210 025774 023737 004170 004130 4S:
S211 026002 001401
S212 026004 104145
S213 026006 023737 004172 004132 5S:
S214 026014 001401
S215 026016 104146
S216 026020 023737 004174 004134 6S:
S217 026026 001401
S218 026030 104147
S219 026032 013737 004120 004220 7S:
S220 026040 013737 004130 004222
S221 026046 013737 004132 004224
S222 026054 013737 004134 004226
S223
S224 026062 012762 100000 000000
S225 026070 016237 000000 004120
S226 026076 016237 000010 004130

: *
: * 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.
: *
: * *****
TS2: SCOPE
MOV #100,STIMES ;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 #3,RKDCYL(R2) ;LOAD CYLINDER AND
MOV #0,RKDA(R2) ;LOAD HEAD ADDRESS
MOV #SEEK,RKCS1(R2) ;ISSUE SEEK
MOV #22,*4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
MOV #DMD!MCLK,RKMR1(R2)
MOV #DMD,RKMR1(R2),
R0
DEC R0
BNE 1S
CLR RKMR1(R2) ;FINISH COMMAND IN NORMAL MODE
MOV WAITIM,R0 ;WAIT FOR READY
TSTB RKCS1(R2)
BMI 3S
R0
DEC R0
BNE 2S
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 RKR(R2),T.ER ;STORE ERROR REG
MOV #CERR!SPAR!RDY!SEEK<1C<GO>>,E.CS1 ;LOAD EXPECTED CS1
MOV #IR,E.CS2 ;LOAD EXPECTED CS2
MOV #SVAL!DRA,E.DS ;LOAD EXPECTED DRIVE STATUS REG
MOV #0,E.ER ;LOAD EXPECTED ERROR REG
CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG.1 CORRECT
BEQ 4S ;YES, CONTINUE
ERROR 144
CMP E.CS2,T.CS2 ;CHECK COMMAND AND STATUS REG. 2 CORRECT
BEQ 5S ;YES, CONTINUE
ERROR 145
CMP E.DS,T.DS ;CHECK DRIVE STATUS REG. CORRECT
BEQ 6S ;YES, CONTINUE
ERROR 146
CMP E.ER,T.ER ;CHECK ERROR REGISTER CORRECT
BEQ 7S ;YES, CLEAR RK611
ERROR 147
MOV T.CS1,P.CS1 ;STORE PREVIOUS CONTENTS OF
MOV T.CS2,P.CS2 ;COMMAND AND STATUS REG 1
MOV T.DS,P.DS ;COMMAND AND STATUS REG 2
MOV T.ER,P.ER ;DRIVE STATUS REG
;AND ERROR REG
MOV #CCLR,RKCS1(R2) ;CLEAR RK611
MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
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 T53 ;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 AVAILABLE
5254 * IS RESET AND CONTROLLER ERROR IS SET.
5255 *
5256 *****
5257 *ST53: SCOPE
5258 026214 000004 MOV #100, $TIMES ;DO 100. ITERATIONS
5259 026216 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
5260 026224 013702 001270 MOV #CLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5261 026230 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
5262 026236 012762 000040 000026 MOV #SELDRV,RKCS1(R2) ;ISSUE SELDRV
5263 026244 012762 000001 000000 MOV #22,*4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5264 026252 012700 000132 1$: MOV #DMD,MCLK,RKMR1(R2)
5265 026256 012762 000440 000026 MOV #DMD,RKMR1(R2)
5266 026264 012762 000040 000026 DEC R0
5267 026272 005300 BNE 1$
5268 026274 001370 CLR RKMR1(R2) ;FINISH COMMAND IN NORMAL MODE
5269 026276 005062 000026 MOV WAITIM,R0 ;WAIT FOR READY
5270 026302 013700 004262 2$: TSTB RKCS1(R2)
5271 026306 105762 000000 BMI 3$
5272 026312 100402 DEC R0
5273 026314 005300 BNE 2$
5274 026316 001373 3$: MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5275 026320 016237 000000 004120 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5276 026326 016237 000010 004130 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG
5277 026334 016237 000012 004132 MOV RKER(R2),T.ER ;STORE ERROR REG
5278 026342 016237 000014 004134 MOV #CERR!RDY!SELDRV,<T<GO>>,E.CS1 ;LOAD EXPECTED CS1
5279 026350 012737 100200 004160 MOV #IR,E.CS2 ;LOAD EXPECTED CS2
5280 026356 012737 000100 004170 MOV #SVAL!0,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5281 026364 012737 100000 004172 MOV #0,E.ER ;LOAD EXPECTED ERROR REG
5282 026372 012737 000000 004174 MOV #0,E.ER ;LOAD EXPECTED ERROR REG
5283 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 ERR 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$: MOV T.CS1,P.CS1 ;STORE PREVIOUS CONTENTS OF
5295 026456 013737 004130 004222 MOV T.CS2,P.CS2 ;COMMAND AND STATUS REG 1
5296 026464 013737 004132 004224 MOV T.DS,P.DS ;COMMAND AND STATUS REG 2
5297 026472 013737 004134 004226 MOV T.ER,P.ER ;DRIVE STATUS REG
5298 ;AND ERROR REG
5299 026500 012762 100000 000000 MOV #CCLR,RKCS1(R2) ;CLEAR RK611
5300 026506 016237 000000 004120 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5301 026514 016237 000010 004130 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5302 026522 016237 000012 004132 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG
5303 026530 016237 000014 074134 MOV RKER(R2),T.ER ;STORE ERROR REG
5304 026536 012737 000200 004160 MOV #RDY,E.CS1 ;LOAD EXPECTED CS1
5305 026544 012737 000100 004170 MOV #IR,E.CS2 ;LOAD EXPECTED CS2
5306 026552 005037 004172 CLR E.DS ;LOAD EXPECTED DRIVE STATUS REG
5307 026556 005037 004174 CLR E.ER ;LOAD EXPECTED ERROR REG
5308 026562 023737 004160 004120 CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG 1 CORRECT
5309 026570 001401 BEQ 11$ ;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$: CMP E.ER,T.ER ;CHECK IF ERROR REG CORRECT
5318 026626 001401 BEQ TST54 ;YES, GO ON TO NEXT TEST
5319 026630 104227 ERROR 227 ;ERROR REG INCORRECT
5320
5321 *****
5322 *TEST 54 CDT SET DRIVE TYPE
5323 *
5324 *
5325 * CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5326 * PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
5327 * WITH CDT SET, 26 SECTOR FORMAT, TO CYLINDER 23.
5328 * HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE
5329 * UNTIL PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE
5330 * AND MAKE SURE ONLY DRIVE AVAILABLE SETS.
5331 *****
5332 *TST54: SCOPE
5333 026632 000004 MOV #100,$TIMES ;DO 100. ITERATIONS
5334 026634 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
5335 026642 013702 001270 MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5336 026646 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
5337 026654 012762 000040 000026 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<ICGO>,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!DOT,E.DS	;LOAD EXPECTED DRIVE STATUS REG
5358	027024	012737	000000	004174	MOV	#D.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	#CLR,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

K08

CZR6JCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 101  
T54 CDT SET DRIVE TYPE

SEQ 0101

5395 027260 001401  
5396 027262 104227BEQ 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 AVAILABLE, DRIVE TYPE ERROR,  
\* AND CONTROLLER ERROR SET.  
\*\*\*\*\*

↑TST55: SCOPE

5410	027264	000004				MOV	#100.,\$TIMES	;:DO 100. ITERATIONS
5411	027266	012737	000144	001200		MOV	\$BASE,R2	;:LOAD RK611 BASE
5412	027274	013702	001270			MOV	\$SCLR,RKCS2(R2)	;:CLEAR RK06 SUBSYSTEM
5413	027300	012762	000040	000010		MOV	#DMD,RKMR1(R2)	;:PUT RK611 IN MAINT MODE
5414	027306	012762	000040	000026		MOV	#2,RKDCYL(R2)	;:LOAD CYLINDER AND
5415	027314	012762	000002	000020		MOV	#0,RKDA(R2)	;:LOAD HEAD ADDRESS
5416	027322	012762	000000	000006		MOV	#CDT!SEEK,RKCS1(R2)	;:ISSUE CDT!SEEK
5417	027330	012762	002017	000000		MOV	#22,*4+2,R0	;:ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5418	027336	012700	000132			MOV	#DMD!MCLK,RKMR1(R2)	
5419	027342	012762	000440	000026	1\$:	MOV	#DMD,RKMR1(R2)	
5420	027350	012762	000040	000026		DEC	R0	
5421	027356	005300				BNE	1\$	
5422	027360	001370				CLR	RKMR1(R2)	;:FINISH COMMAND IN NORMAL MODE
5423	027362	005062	000026			MOV	WAITIM,R0	;:WAIT FOR READY
5424	027366	013700	004262			TSTB	RKCS1(R2)	
5425	027372	105762	000000		2\$:	BMI	3\$	
5426	027376	1J0402				DEC	R0	
5427	027400	005300				BNE	2\$	
5428	027402	001373				MOV	RKCS1(R2),T.CS1	;:STORE COMMAND AND STATUS REG 1
5429	027404	016237	000000	004120	3\$:	MOV	RKCS2(R2),T.CS2	;:STORE COMMAND AND STATUS REG 2
5430	027412	016237	000010	004130		MOV	RKDS(R2),T.DS	;:STORE DRIVE STATUS REG
5431	027420	016237	000012	004132		MOV	RKER(R2),T.ER	;:STORE ERROR REG
5432	027426	016237	000014	004134		MOV	#CDT!CERR!RDY!CDT!SEEK<↑C<GO>>,E.CS1	;:LOAD EXPECTED CS1
5433	027434	012737	102216	004160		MOV	#IR,E.CS2	;:LOAD EXPECTED CS2
5434	027442	012737	000100	004170		MOV	#SVAL!DRA,E.DS	;:LOAD EXPECTED DRIVE STATUS REG
5435	027450	012737	100001	004172		MOV	#DTYE,E.ER	;:LOAD EXPECTED ERROR REG
5436	027456	012737	000040	004174		CMP	E.CS1,T.CS1	;:CHECK COMMAND AND STATUS REG.1 CORRECT
5437	027464	023737	004160	004120		BEQ	4\$	;:YES, CONTINUE
5438	027472	001401				ERROR	160	
5439	027474	104160				CMP	E.CS2,T.CS2	;:CHECK COMMAND AND STATUS REG. 2 CORRECT
5440	027476	023737	004170	004130	4\$:	BEQ	5\$	;:YES, CONTINUE
5441	027504	001401				ERROR	161	
5442	027506	104161				CMP	E.DS,T.DS	;:CHECK DRIVE STATUS REG. CORRECT
5443	027510	023737	004172	004132	5\$:	BEQ	6\$	;:YES, CONTINUE
5444	027516	001401				ERROR	162	
5445	027520	104162				CMP	E.ER,T.ER	;:CHECK ERROR REGISTER CORRECT
5446	027522	023737	004174	004134	6\$:	BEQ	7\$	;:YES, CLEAR RK611
5447	027530	001401				ERROR	163	
5448	027532	104163				MOV	T.CS1,P.CS1	;:STORE PREVIOUS CONTENTS OF
5449	027534	013737	004120	004220	7\$:	MOV	T.CS2,P.CS2	;:COMMAND AND STATUS REG 1
5450	027542	013737	004130	004222				

```

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

```

```

*****
*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.

```

```

*****
TST56: SCOPE
MOV #100,$TIMES ; DO 100. ITERATIONS
MOV $BASE,R2 ; LOAD RK611 BASE
MOV #SCLR,RKCS2(R2) ; CLEAR RK06 SUBSYSTEM
MOV #DMD,RKMR1(R2) ; PUT RK611 IN MAINT MODE
MOV #23,RKDCYL(R2) ; LOAD CYLINDER FND
MOV #0,RKDA(R2) ; LOAD HEAD ADDRESS
MOV #SEEK,RKCS1(R2) ; ISSUE SEEK
MOV #22,*4+2,R0 ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
1$: 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$

```

```

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 TS157 ;YES, GO ON TO NEXT TEST
5552 030346 104227 ERROR 227 ;ERROR REG INCORRECT

```

```

*****
*TEST 57 SPEED LOSS FROM SHIFT REG.
*
* 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 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.
*

```

5553  
5554  
5555  
5556  
5557  
5558  
5559  
5560  
5561  
5562



# N08

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 104  
TS7 SPEED LOSS FROM SHIFT REG.

SEQ 0104

```

5563
5564
5565 030350 000004
5566 030352 012737 000144 001200
5567 030360 013702 001270
5568 030364 012762 000040 000010
5569 030372 012762 000040 000026
5570 030400 012762 000003 000020
5571 030406 012762 000400 000006
5572 030414 012762 000017 000000
5573 030422 012700 000132
5574 030426 012762 000440 000026 1$:
5575 030434 012762 000040 000026
5576 030442 005300
5577 030444 001370
5578 030446 005062 000026
5579 030452 013700 004262
5580 030456 105762 000000 2$:
5581 030462 100402
5582 030464 005300
5583 030466 001373
5584 030470 016237 000000 004120 3$:
5585 030476 016237 000010 004130
5586 030504 016237 000012 004132
5587 030512 016237 000014 004134
5588 030520 012737 000216 004160
5589 030526 012737 000100 004170
5590 030534 012737 100021 004172
5591 030542 012737 000000 004174
5592 030550 023737 004160 004120
5593 030556 001401
5594 030560 104170
5595 030562 023737 004170 004130 4$:
5596 030570 001401
5597 030572 104171
5598 030574 023737 004172 004132 5$:
5599 030602 001401
5600 030604 104172
5601 030606 023737 004174 004134 6$:
5602 030614 001401
5603 030616 104173
5604 030620 013737 004120 004220 7$:
5605 030626 013737 004130 004222
5606 030634 013737 004132 004224
5607 030642 013737 004134 004226
5608
5609 030650 012762 100000 000000
5610 030656 016237 000000 004120
5611 030664 016237 000010 004130
5612 030672 016237 000012 004132
5613 030700 016237 000014 004134
5614 030706 012737 000200 004160
5615 030714 012737 000100 004170
5616 030722 005037 004172
5617 030726 005037 004174
5618 030732 023737 004160 004120

; *
; *****
; TS7: SCOPE
; MOV #100, $TIMES ; DO 100. ITERATIONS
; MOV $BASE, R2 ; LOAD RK611 BASE
; MOV #SCLR, RKCS2(R2) ; CLEAR RK06 SUBSYSTEM
; MOV #DMD, RKMR1(R2) ; PUT RK611 IN MAINT MODE
; MOV #3, RKDCYL(R2) ; LOAD CYLINDER AND
; MOV #400, RKDA(R2) ; LOAD HEAD ADDRESS
; MOV #SEEK, RKCS1(R2) ; ISSUE SEEK
; MOV #22, *4+2, RO ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
; MOV #DMD!MCLK, RKMR1(R2)
; MOV #DMD, RKMR1(R2)
; DEC RO
; BNE 1$
; CLR RKMR1(R2) ; FINISH COMMAND IN NORMAL MODE
; MOV WAITIM, RO ; WAIT FOR READY
; TSTB RKCS1(R2)
; BMI 3$
; DEC RO
; 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 REG
; MOV RKER(R2), T.ER ; STORE ERROR REG
; MOV #RDY!SEEK&C<GO>, E.CS1 ; LOAD EXPECTED CS1
; MOV #IR, E.CS2 ; LOAD EXPECTED CS2
; MOV #SVAL!DRA!SPDLSS, E.DS ; LOAD EXPECTED DRIVE STATUS REG
; MOV #J, E.ER ; LOAD EXPECTED ERROR REG
; CMP E.CS1, T.CS1 ; CHECK COMMAND AND STATUS REG.1 CORRECT
; BEQ 4$ ; YES, CONTINUE
; ERROR 170
; CMP E.CS2, T.CS2 ; CHECK COMMAND AND STATUS REG. 2 CORRECT
; BEQ 5$ ; YES, CONTINUE
; ERROR 171
; CMP E.DS, T.DS ; CHECK DRIVE STATUS REG. CORRECT
; BEQ 6$ ; YES, CONTINUE
; ERROR 172
; CMP E.ER, T.ER ; CHECK ERROR REGISTER CORRECT
; BEQ 7$ ; YES, CLEAR RK611
; ERROR 173
; MOV T.CS1, P.CS1 ; STORE PREVIOUS CONTENTS OF
; MOV T.CS2, P.CS2 ; COMMAND AND STATUS REG 1
; MOV T.DS, P.DS ; COMMAND AND STATUS REG 2
; MOV T.ER, P.ER ; DRIVE STATUS REG
; ; AND ERROR REG
; MOV #CCLR, RKCS1(R2) ; CLEAR RK611
; 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 #RDY, E.CS1 ; LOAD EXPECTED CS1
; MOV #IR, E.CS2 ; LOAD EXPECTED 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

```

```

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$: CMP E.ER,T.ER ;CHECK IF ERROR REG CORRECT
5628 030776 001401 BEQ TST60 ;YES, GO ON TO NEXT TEST
5629 031000 104227 ERROR 227 ;ERROR REG INCORRECT
5630
5631 ;*****
5632 ;*TEST 60 DRIVE OFF TRACK FROM SHIFT REG.
5633 ;*
5634 ;* CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5635 ;* PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK TO A RK06.
5636 ;* 26 SECTOR FORMAT TO CYLINDER 3, HEAD 2, DRIVE 0.
5637 ;* CLOCK IN DIAGNOSTIC MODE UNTIL PHASE ADDRESS 6.
5638 ;* TURN OFF DIAGNOSTIC MODE AND MAKE SURE DRIVE AVAILIABLE
5639 ;* AND DRIVE OFF TRACK ARE SET.
5640 ;*
5641 ;*****
5642 ;*****
5643 031002 000004 TST60: SCOPE
5644 031004 012737 000144 001200 MOV #100,$TIMES ;DO 100. ITERATIONS
5645 031012 013702 001270 MOV $BASE,R2 ;LOAD RK611 BASE
5646 031016 012762 000040 000010 MOV $SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5647 031024 012762 000040 000026 MOV $DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
5648 031032 012762 000003 000020 MOV #3,RKDCYL(R2) ;LOAD CYLINDER AND
5649 031040 012762 001000 000006 MOV #1000,RKDR(R2) ;LOAD HEAD ADDRESS
5650 031046 012762 000017 000000 MOV $SEEK,RKCS1(R2) ;ISSUE SEEK
5651 031054 012700 000132 MOV #22,*4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5652 031060 012762 000440 000026 1$: MOV $DMD!MCLK,RKMR1(R2)
5653 031066 012762 000040 000026 MOV $DMD,RKMR1(R2)
5654 031074 005300 DEC R0
5655 031076 001370 BNE 1$
5656 031100 005062 000026 CLR RKMR1(R2) ;FINISH COMMAND IN NORMAL MODE
5657 031104 013700 004262 MOV WAITIM,R0 ;WAIT FOR READY
5658 031110 105762 000000 2$: TSTB RKCS1(R2)
5659 031114 100402 BMI 3$
5660 031116 005300 DEC R0
5661 031120 001373 BNE 2$
5662 031122 016237 000000 004120 3$: MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5663 031130 016237 000010 004130 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5664 031136 016237 000012 004132 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG
5665 031144 016237 000014 004134 MOV RKER(R2),T.ER ;STORE ERROR REG
5666 031152 012737 000216 004160 MOV $RDY!SEEK&<T.CS1>>,E.CS1 ;LOAD EXPECTED CS1
5667 031160 012737 000100 004170 MOV $IR,E.CS2 ;LOAD EXPECTED CS2
5668 031166 012737 100041 004172 MOV $SVAL!DRA!DROT,E.DS ;LOAD EXPECTED DRIVE STATUS REG
5669 031174 012737 000000 004174 MOV $D.E.ER ;LOAD EXPECTED ERROR REG
5670 031202 023737 004160 004120 CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG.1 CORRECT
5671 031210 001401 BEQ 4$ ;YES, CONTINUE
5672 031212 104174 ERROR 174
5673 031214 023737 004170 004130 4$: CMP E.CS2,T.CS2 ;CHECK COMMAND AND STATUS REG. 2 CORRECT
5674 031222 001401 BEQ 5$ ;YES, CONTINUE
5675 031224 104175 ERROR 175

```

```

5675 031226 023737 004172 004132 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

```

```

*****
;TEST 61 WRITE LOCK ERROR FROM SHIFT REG.

```

```

;
; CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
; PUT CONTROLLER IN DIAGNOSTIC MODE. ISSUE A PACK ACKNOWLEDGE
; 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.

```

```

*****

```

```

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	RO	
5732	031530	001370				BNE	1\$	
5733	031532	005062	000026			CLR	RKMR1(R2)	;FINISH COMMAND IN NORMAL MODE
5734	031536	013700	004262			MOV	WAITIM,RO	;WAIT FOR READY
5735	031542	105762	000000		2\$:	TSTB	RKCS1(R2)	
5736	031546	100402				BMI	3\$	
5737	031550	005300				DEC	RO	
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, CONTINUE
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	#CLR,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								

;:\*\*\*\*\*

```

5787      *TEST 62      SEEK INCOMPLETE
5788      *
5789      *      CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
5790      *      PUT CONTROLLER IN DIAGNOSTIC MODE.  ISSUE AN UNLOAD
5791      *      TO A RK06, 26 SECTOR FORMAT, WITH CYLINDER 0
5792      *      HEAD 1, DRIVE 0, CLOCK IN DIAGNOSTIC MODE UNTIL
5793      *      PHASE ADDRESS 6.  TURN OFF DIAGNOSTIC MODE AND MAKE
5794      *      SURE SPEED LOSS, SEEK INCOMPLETE, AND CONTROLLER ERROR
5795      *      ARE SET WITH DRIVE AVAILABLE RESET.
5796      *
5797      *
5798      *****
5799      032066 000004 1ST62: SCOPE
5800      032070 012737      MOV      #100, $TIMES      ; DO 100. ITERATIONS
5801      032076 013702      MOV      $BASE, R2      ; LOAD RK611 BASE
5802      032102 012762      MOV      #SCLR, RKCS2(R2) ; CLEAR RK06 SUBSYSTEM
5803      032110 012762      MOV      #DMD, RKMR1(R2) ; PUT RK611 IN MAINT MODE
5804      032116 012762      MOV      #0, RKDCYL(R2)  ; LOAD CYLINDER AND
5805      032124 012762      MOV      #400, RKDA(R2)   ; LOAD HEAD ADDRESS
5806      032132 012762      MOV      #UNLOAD, RKCS1(R2) ; ISSUE UNLOAD
5807      032140 012700      MOV      #22, *4+2, R0    ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5808      032144 012762      MOV      #DMD!MCLK, RKMR1(R2)
5809      032152 012762      MOV      #DMD, RKMR1(R2)
5810      032160 005300      DEC      R0
5811      032162 001370      BNE      1$
5812      032164 005062      CLR      RKMR1(R2)      ; FINISH COMMAND IN NORMAL MODE
5813      032170 013700      MOV      WAITIM, R0      ; WAIT FOR READY
5814      032174 105762      TSTB     RKCS1(R2)
5815      032200 100402      BMI      3$
5816      032202 005300      DE      R0
5817      032204 001373      BNE      2$
5818      032206 016237      MOV      RKCS1(R2), T.CS1 ; STORE COMMAND AND STATUS REG 1
5819      032214 016237      MOV      RKCS2(R2), T.CS2 ; STORE COMMAND AND STATUS REG 2
5820      032222 016237      MOV      RKDS(R2), T.DS  ; STORE DRIVE STATUS REG
5821      032230 016237      MOV      RKER(R2), T.ER  ; STORE ERROR REG
5822      032236 012737      MOV      #CERR!RDY!UNLOAD, <T.CS1> E.CS1 ; LOAD EXPECTED CS1
5823      032244 012737      MOV      #IR, E.CS2      ; LOAD EXPECTED CS2
5824      032252 012737      MOV      #SVAL!SPDLSS, E.DS ; LOAD EXPECTED DRIVE STATUS REG
5825      032260 012737      MOV      #SKI, E.ER      ; LOAD EXPECTED ERROR REG
5826      032266 023737      CMP      E.CS1, T.CS1    ; CHECK COMMAND AND STATUS REG.1 CORRECT
5827      032274 001401      BEQ      4$              ; YES, CONTINUE
5828      032276 104204      ERROR    204
5829      032300 023737      CMP      E.CS2, T.CS2    ; CHECK COMMAND AND STATUS REG. 2 CORRECT
5830      032306 001401      BEQ      5$              ; YES, CONTINUE
5831      032310 104205      ERROR    205
5832      032312 023737      CMP      E.DS, T.DS      ; CHECK DRIVE STATUS REG. CORRECT
5833      032320 001401      BEQ      6$              ; YES, CONTINUE
5834      032322 104206      ERROR    206
5835      032324 023737      CMP      E.ER, T.ER      ; CHECK ERROR REGISTER CORRECT
5836      032332 001401      BEQ      7$              ; YES, CLEAR RK611
5837      032334 104207      ERROR    207
5838      032336 013737      MOV      T.CS1, P.CS1    ; STORE PREVIOUS CONTENTS OF
5839      032344 013737      MOV      T.CS2, P.CS2    ; COMMAND AND STATUS REG 1
5840      032352 013737      MOV      T.DS, P.DS      ; COMMAND AND STATUS REG 2
5841      032360 013737      MOV      T.ER, P.ER      ; DRIVE STATUS REG
5842      032366 012762      MOV      #CCLR, RKCS1(R2) ; CLEAR RK611
5843      032370 012762      MOV      #0, RKDCYL(R2)  ; CLEAR CYLINDER
5844      032376 012762      MOV      #0, RKDA(R2)    ; CLEAR HEAD ADDRESS
5845      032384 012762      MOV      #UNLOAD, RKCS1(R2) ; ISSUE UNLOAD
5846      032392 012700      MOV      #22, *4+2, R0    ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5847      032394 012762      MOV      #DMD!MCLK, RKMR1(R2)
5848      032396 012762      MOV      #DMD, RKMR1(R2)
5849      032400 005300      DEC      R0
5850      032402 001370      BNE      1$
5851      032404 005062      CLR      RKMR1(R2)      ; FINISH COMMAND IN NORMAL MODE
5852      032410 013700      MOV      WAITIM, R0      ; WAIT FOR READY
5853      032414 105762      TSTB     RKCS1(R2)
5854      032420 100402      BMI      3$
5855      032422 005300      DE      R0
5856      032424 001373      BNE      2$
5857      032426 016237      MOV      RKCS1(R2), T.CS1 ; STORE COMMAND AND STATUS REG 1
5858      032434 016237      MOV      RKCS2(R2), T.CS2 ; STORE COMMAND AND STATUS REG 2
5859      032442 016237      MOV      RKDS(R2), T.DS  ; STORE DRIVE STATUS REG
5860      032450 016237      MOV      RKER(R2), T.ER  ; STORE ERROR REG
5861      032458 012737      MOV      #CERR!RDY!UNLOAD, <T.CS1> E.CS1 ; LOAD EXPECTED CS1
5862      032466 012737      MOV      #IR, E.CS2      ; LOAD EXPECTED CS2
5863      032474 012737      MOV      #SVAL!SPDLSS, E.DS ; LOAD EXPECTED DRIVE STATUS REG
5864      032482 012737      MOV      #SKI, E.ER      ; LOAD EXPECTED ERROR REG
5865      032490 023737      CMP      E.CS1, T.CS1    ; CHECK COMMAND AND STATUS REG.1 CORRECT
5866      032498 001401      BEQ      4$              ; YES, CONTINUE
5867      032500 104204      ERROR    204
5868      032502 023737      CMP      E.CS2, T.CS2    ; CHECK COMMAND AND STATUS REG. 2 CORRECT
5869      032510 001401      BEQ      5$              ; YES, CONTINUE
5870      032512 104205      ERROR    205
5871      032514 023737      CMP      E.DS, T.DS      ; CHECK DRIVE STATUS REG. CORRECT
5872      032522 001401      BEQ      6$              ; YES, CONTINUE
5873      032524 104206      ERROR    206
5874      032526 023737      CMP      E.ER, T.ER      ; CHECK ERROR REGISTER CORRECT
5875      032534 001401      BEQ      7$              ; YES, CLEAR RK611
5876      032536 104207      ERROR    207
5877      032538 013737      MOV      T.CS1, P.CS1    ; STORE PREVIOUS CONTENTS OF
5878      032546 013737      MOV      T.CS2, P.CS2    ; COMMAND AND STATUS REG 1
5879      032554 013737      MOV      T.DS, P.DS      ; COMMAND AND STATUS REG 2
5880      032562 013737      MOV      T.ER, P.ER      ; DRIVE STATUS REG
5881      032564 012762      MOV      #CCLR, RKCS1(R2) ; CLEAR RK611
5882      032566 012762      MOV      #0, RKDCYL(R2)  ; CLEAR CYLINDER
5883      032568 012762      MOV      #0, RKDA(R2)    ; CLEAR HEAD ADDRESS
5884      032570 012762      MOV      #UNLOAD, RKCS1(R2) ; ISSUE UNLOAD
5885      032572 012700      MOV      #22, *4+2, R0    ; ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5886      032574 012762      MOV      #DMD!MCLK, RKMR1(R2)
5887      032576 012762      MOV      #DMD, RKMR1(R2)
5888      032578 005300      DEC      R0
5889      032580 001370      BNE      1$
5890      032582 005062      CLR      RKMR1(R2)      ; FINISH COMMAND IN NORMAL MODE
5891      032584 013700      MOV      WAITIM, R0      ; WAIT FOR READY
5892      032586 105762      TSTB     RKCS1(R2)
5893      032588 100402      BMI      3$
5894      032590 005300      DE      R0
5895      032592 001373      BNE      2$
5896      032594 016237      MOV      RKCS1(R2), T.CS1 ; STORE COMMAND AND STATUS REG 1
5897      032596 016237      MOV      RKCS2(R2), T.CS2 ; STORE COMMAND AND STATUS REG 2
5898      032598 016237      MOV      RKDS(R2), T.DS  ; STORE DRIVE STATUS REG
5899      032600 016237      MOV      RKER(R2), T.ER  ; STORE ERROR REG
5900      032602 012737      MOV      #CERR!RDY!UNLOAD, <T.CS1> E.CS1 ; LOAD EXPECTED CS1
5901      032604 012737      MOV      #IR, E.CS2      ; LOAD EXPECTED CS2
5902      032606 012737      MOV      #SVAL!SPDLSS, E.DS ; LOAD EXPECTED DRIVE STATUS REG
5903      032608 012737      MOV      #SKI, E.ER      ; LOAD EXPECTED ERROR REG
5904      032610 023737      CMP      E.CS1, T.CS1    ; CHECK COMMAND AND STATUS REG.1 CORRECT
5905      032612 001401      BEQ      4$              ; YES, CONTINUE
5906      032614 104204      ERROR    204
5907      032616 023737      CMP      E.CS2, T.CS2    ; CHECK COMMAND AND STATUS REG. 2 CORRECT
5908      032618 001401      BEQ      5$             
```

```

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 AVAILABLE RESET.
5874 :
5875 *****
5876 TST63: SCOPE
5877 032520 000004 MOV #100, $TIMES ;DO 100. ITERATIONS
5878 032522 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
5879 032530 013702 001270 000010 MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
5880 032534 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
5881 032542 012762 000040 000026 MOV #0,RKDCYL(R2) ;LOAD CYLINDER AND
5882 032550 012762 000000 000020 MOV #400,RKDA(R2) ;LOAD HEAD ADDRESS
5883 032556 012762 000400 000006 MOV #CLEAR,RKCS1(R2) ;ISSUE CLEAR
5884 032564 012762 000005 000000 MOV #22,*4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
5885 032576 012762 000440 000026 1$: 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

```

```

5899 032670 012737 100204 004160 MOV #CERR!RDY!CLEAR&(<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!SPOLSS,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$: MOV T.CS1,P.CS1 ;STORE PREVIOUS CONTENTS OF
5916 032776 013737 004130 004222 MOV T.CS2,P.CS2 ;COMMAND AND STATUS REG 1
5917 033004 013737 004132 004224 MOV T.DS,P.DS ;COMMAND AND STATUS REG 2
5918 033012 013737 004134 004226 MOV T.ER,P.ER ;DRIVE STATUS REG
5919 ;AND ERROR REG
5920 033020 012762 100000 000000 MOV #CLR,RKCS1(R2) ;CLEAR RK611
5921 033026 016237 000000 004120 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
5922 033034 016237 000010 004130 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
5923 033042 016237 000012 004132 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG
5924 033050 016237 000014 004134 MOV RKER(R2),T.ER ;STORE ERROR REG
5925 033056 012737 000200 004160 MOV #RDY,E.CS1 ;LOAD EXPECTED CS1
5926 033064 012737 000100 004170 MOV #IR,E.CS2 ;LOAD EXPECTED CS2
5927 033072 005037 004172 CLR E.DS ;LOAD EXPECTED DRIVE STATUS REG
5928 033076 005037 004174 CLR E.ER ;LOAD EXPECTED ERROR REG
5929 033102 023737 004160 004120 CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG 1 CORRECT
5930 BEQ 11$ ;YES, CHECK CS2
5931 033112 104224 ERROR 224 ;CS1 INCORRECT
5932 033114 023737 004 70 004130 11$: CMP E.CS2,T.CS2 ;CHECK COMMAND AND STATUS REG 2 CORRECT
5933 BEQ 12$ ;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

```

```

*****
;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.

```

```

*****
;TST64: SCOPE

```

```

5953 033152 000004
5954

```



H09

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T6402-DEC-77 09:31 PAGE 111  
AC LOW AND C-D PARITY FROM SHIFT REG.

SEQ 0111

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		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 CORRECT
6009	033540	001401			BEQ	12\$	;;YES, CHECK DRIVE STATUS REG
6010	033542	104225			ERROR	225	;;CS2 INCORRECT



CZR6800 RK611 DSKLS CTRL PRT2  
CZR6800.P11 02-DEC-77 09:22

MACY11 30(1046)  
T64

02-DEC-77 09:31 PAGE 112  
AC LOW AND C-0 PARITY FROM SHIFT REG.

SEQ 0112

```

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 000040 MOV $BASE,R2 ;LOAD RK611 BASE
6033 033604 012762 000040 000010 MOV $SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6034 033612 012762 000040 000026 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINT MODE
6035 033620 012762 000000 000020 MOV #0,RKDCYL(R2) ;LOAD CYLINDER AND
6036 033626 012762 000400 000006 MOV #400,RKDA(R2) ;LOAD HEAD ADDRESS
6037 033634 012762 000013 000000 MOV #RECAL,RKCS1(R2) ;ISSUE RECAL
6038 033642 012700 000132 000026 1$: MOV #22, #4+2,R0 ;ISSUE CLOCKS UNTIL PHASE ADDRESS 6
6039 033646 012762 000440 000026 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 000000 MOV WAITIM,R0 ;WAIT FOR READY
6045 033676 105762 000000 23: TSTB RKCS1(R2)
6046 033702 100402 BMI 3$
6047 033704 005300 DEC R0
6048 033706 001373 BNE 2$
6049 033710 016237 000000 004120 3$: MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
6050 033716 016237 000010 004130 MO' RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG 2
6051 033724 016237 000012 004132 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG
6052 033732 016237 000014 004134 MOV RKER(R2),T.ER ;STORE ERROR REG
6053 033740 012737 100212 004160 MOV #CERR!RDY!RECAL<T.CGO>,E.CS1 ;LOAD EXPECTED CS1
6054 033746 012737 000100 004170 MOV #IR,E.CS2 ;LOAD EXPECTED CS2
6055 033754 012737 100020 004172 MOV #SVAL!SPDLSS,E.DS ;LOAD EXPECTED DRIVE STATUS REG
6056 033762 012737 002000 004174 MOV #IDAE,E.ER ;LOAD EXPECTED ERROR REG
6057 033770 023737 004160 004120 CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG.1 CORRECT
6058 033776 001401 BEQ 4$ ;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  
6098  
6099  
6100  
6101  
6102  
6103  
6104  
6105  
6106  
6107

\*\*\*\*\*  
\*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.  
\*\*\*\*\*

6108 034222 000004  
6109 034224 012737 000144 001200  
6110 034232 013702 001270  
6111 034236 012762 000040 000010  
6112 034244 012762 000040 000026  
6113 034252 012762 001002 000020  
6114 034260 012737 001002 004252  
6115 034266 012737 000000 004250  
6116 034274 005046  
6117 034276 113766 004250 000001  
6118 034304 012662 000006  
6119 034310 012737 000006 004266  
6120 034316 012762 000017 000000  
6121 034324 012700 000132  
6122 034330 012762 000440 000026 1\$:

TST66: 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 MAINTENANCE MODE  
MOV #1002,RKDCYL(R2) ;LOAD CYLINDER ADDRESS  
MOV #1002,CYLIN  
MOV #0,HDCODE ;LOAD HEAD ADDRESS  
CLR -(SP)  
MOVB HDCODE,1(SP)  
MOV (SP)+,RKDA(R2)  
MOV #6,DRVTYP ;LOAD DRIVE TYPE FOR PRINT OUT  
MOV #SEEK,RKCS1(R2) ;ISSUE SEEK TO RK06  
MOV #22,\*4+2,R0 ;ISSUE CLOCK TO GET THROUGH PHASE 6  
MOV #DMD:MCLK,RKMR1(R2)

K09

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T6602-DEC-77 09:31 PAGE 114  
IDAE DETECTION IN RK611 CONTROLLER (PART 1)

SEQ 0114

6123	034336	012762	000040	000026	MOV	#DMD,RKMR1(R2)	
6124	034344	005300			DEC	RO	
6125	034346	001370			BNE	1\$	
6126	034350	005062	000026		CLR	RKMR1(R2)	;ALLOW COMMAND TO FINISH
6127	034354	013700	004262		MOV	WAITIM,RO	;LOAD WAIT TIME
6128	034360	105762	000000		TSTB	RKCS1(R2)	;WAIT FOR READY
6129	034364	100402			BMI	3\$	
6130	034366	005300			DEC	RO	
6131	034370	001373			BNE	2\$	
6132	034372	016237	000000	004120	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&T.C<GO>>,E.CS1	;LOAD EXPECTED 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	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	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	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	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 CORRECT
6168	034642	001401			BEQ	10\$	;YES, CHECK CS2
6169	034644	104224			ERROR	224	;CS1 INCORRECT
6170	034646	023737	004170	004130	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	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	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

```
6179
6180
6181
6182
6183
6184
6185
6186
6187
6188
6189
6190
6191
6192 034704 000004
6193 034706 012737 000144 001200
6194 034714 013702 001270
6195 034720 012762 000040 000010
6196 034726 012762 000040 000026
6197 034734 012762 001022 000020
6198 034742 012737 001022 004252
6199 034750 012737 000000 004250
6200 034756 005046
6201 034760 113766 004250 000001
6202 034766 012662 000006
6203 034772 012737 000007 004266
6204 035000 012762 002017 000000
6205 035006 012700 000132
6206 035012 012762 000440 000026 1$:
6207 035020 012762 000040 000026
6208 035026 005300
6209 035030 001370
6210 035032 005062 000026
6211 035036 013700 004262
6212 035042 105762 000000 2$:
6213 035046 100402
6214 035050 005300
6215 035052 001373
6216 035054 016237 000000 004120 3$:
6217 035062 016237 000010 004130
6218 035070 016237 000012 004132
6219 035076 016237 000014 004134
6220 035104 012737 002216 004160
6221
6222 035112 012737 000100 004170
6223 035120 012737 120401 004172
6224 035126 012737 000000 004174
6225 035134 023737 004160 004120
6226 035142 001401
6227 035144 104230
6228 035146 023737 004170 004130 4$:
6229 035154 001401
6230 035156 104231
6231 035160 023737 004172 004132 5$:
6232 035166 001401
6233 035170 104232
6234 035172 023737 004174 004134 6$:
```

```
*****
*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.
*****
*ST67: SCOPE
MOV #100, $TIMES ;DO 100. ITERATIONS
MOV $BASE, R2 ;LOAD RK611 BASE
MOV #SCLR, RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
MOV #DMD, RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
MOV #1022, RKDCYL(R2) ;LOAD CYLINDER ADDRESS
MOV #1022, CYLIN
MOV #0, HOCODE ;LOAD HEAD ADDRESS
CLR -(SP)
MOVB HOCODE, 1(SP)
MOV (SP)+, RKDR(R2)
MOV #7, DRVTP ;LOAD DRIVE TYPE FOR PRINT OUT
MOV #CDT!SEEK, RKCS1(R2) ;ISSUE SEEK TO RK06
MOV #22, *4+2, RD ;ISSUE CLOCK TO GET THROUGH PHASE 6
MOV #DMD!MCLK, RKMR1(R2)
MOV #DMD, RKMR1(R2)
DEC RD
BNE 1$
CLR RKMR1(R2) ;ALLOW COMMAND TO FINISH
MOV WAITIM, RD ;LOAD WAIT TIME
TSTB RKCS1(R2) ;WAIT FOR READY
BMI 3$
DEC RD
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 REG
MOV RKER(R2), T.ER ;STORE ERROR REG
MOV #CDT!RDY!<SEEK&10<GO>>, E.CS1 ;LOAD EXPECTED CS1
MOV #IR, E.CS2 ;LOAD EXPECTED COMMAND AND STATUS REG.2
MOV #SVAL!DRA!PIP!DDT, E.DS ;LOAD EXPECTED DRIVE STATUS REG
MOV #0, E.ER ;LOAD EXPECTED ERROR REG
CMP E.CS1, T.CS1 ;CHECK COMMAND AND STATUS REG1 CORRECT
BEQ 4$ ;YES, CHECK CS2
ERROR 230 ;CS1 INCORRECT
CMP E.CS2, T.CS2 ;CHECK COMMAND AND STATUS REG2 CORRECT
BEQ 5$ ;YES, CHECK DRIVE STATUS REG.
ERROR 231 ;CS2 INCORRECT
CMP E.DS, T.DS ;CHECK DRIVE STATUS REG. CORRECT
BEQ 6$ ;YES, CHECK ERROR REG
ERROR 232 ;DRIVE STATUS REG. INCORRECT
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$: CMP      E.ER,T.ER ;CHECK ERROR REG CORRECT
6261 035362 001401      BEQ      TST70 ;YES, GO ON TO NEXT TEST
6262 035364 104227      ERROR    227      ;ERROR REG. INCORRECT
6263
6264 *****
6265 ;TEST 70 IDAE DETECTION IN RK611 CONTROLLER (PART 3)
6266 ;
6267 ;
6268 ; CLEAR RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
6269 ; RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A SEEK
6270 ; TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2,
6271 ; HEAD 3, DRIVE 0. CLOCK IN DIAGNOSTIC MODE, UNTIL
6272 ; PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
6273 ; SURE DRIVE AVAILABLE, DRIVE OFF TRACK, SPEED LOSS,
6274 ; ILLEGAL DISK ADDRESS ERROR, AND CONTROLLER ERROR ARE
6275 ; SET.
6276 *****
6277 ;TST70: SCOPE
6278 035366 000004      MOV      #100,$TIMES ;DO 100. ITERATIONS
6279 035370 012737 000144 001200      MOV      $BASE,R2 ;LOAD RK611 BASE
6280 035376 013702 001270      MOV      #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6281 035402 012762 000040 000010      MOV      #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
6282 035410 012762 000040 000026      MOV      #2,RKDCYL(R2) ;LOAD CYLINDER ADDRESS
6283 035416 012762 000002 000020      MOV      #2,CYLIN ;
6284 035424 012737 000002 004252      MOV      #3,HDCODE ;LOAD HEAD ADDRESS
6285 035432 012737 000003 004250      CLR      -(SP)
6286 035440 005046      MOV      HDCODE,1(SP)
6287 035442 113766 004250 000001      MOV      (SP)+,RKDA(R2)
6288 035450 012662 000006      MOV      #6,DRV TYP ;LOAD DRIVE TYPE FOR PRINT OUT
6289 035454 012737 000006 004266      MOV      #SEEK,RKCS1(R2) ;ISSUE SEEK TO RK06
6290 035462 012762 000017 000000      MOV      #22,*4+2,R0 ;ISSUE CLOCK TO GET THROUGH PHASE 6
6290 035470 012700 000132

```

6291	035474	012762	000440	000026	1\$:	MOV	#DMD:MCLK,RKMR1(R2)	
6292	035502	012762	000040	000026		MOV	#DMD,RKMR1(R2)	
6293	035510	005300				DEC	RD	
6294	035512	001370				BNE	1\$	
6295	035514	005062	000026			CLR	RKMR1(R2)	;ALLOW COMMAND TO FINISH
6296	035520	013700	004262			MOV	WAITIM,RO	;LOAD WAIT TIME
6297	035524	105762	000000		2\$:	TSTB	RKCS1(R2)	;WAIT FOR READY
6298	035530	100402				BMI	3\$	
6299	035532	005300				DEC	RD	
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&T.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	#CLR,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

\*\*\*\*\*  
 TEST 71 IDAE DETECTION IN RK611 CONTROLLER (PART 4)  
 \*\*\*\*\*

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.

\*\*\*\*\*  
 ST71: SCOPE  
 \*\*\*\*\*

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



6403	036336	023737	004174	004134	6S:	CMP	E.ER,T.ER	:CHECK ERROR REG. CORRECT
6404	036344	001401				BEG	7S	:YES CHECK CONTROLLER CLEAR
6405	036346	104233				ERROR	233	:ERROR REG. INCORRECT
6406	036350	013737	004120	004220	7S:	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	#ROY,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				BEG	10S	:YES, CHECK CS2
6422	036472	104224				ERROR	224	:CS1 INCORRECT
6423	036474	023737	004170	004130	10S:	CMP	F.CS2,T.CS2	:CHECK COMMAND AND STATUS REG2 CORRECT
6424	036502	001401				BEG	11S	:YES, CHECK DRIVE STATUS REG
6425	036504	104225				ERROR	225	:CS2 INCORRECT
6426	036506	023737	004172	004132	11S:	CMP	E.DS,T.DS	:CHECK DRIVE STATUS REG CORRECT
6427	036514	001401				BEG	12S	:YES, CHECK ERROR REGISTER
6428	036516	104226				ERROR	226	:DRIVE STATUS REG INCORRECT
6429	036520	023737	004174	004134	12S:	CMP	E.ER,T.ER	:CHECK ERROR REG CORRECT
6430	036526	001401				BEG	TCT72	:YES GO ON TO NEXT TEST
6431	036530	104227				ERROR	227	:ERROR REG. INCORRECT

\*\*\*\*\*  
\*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 COT 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.  
\*\*\*\*\*

6445	036532	000004			†ST72:	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		MOV	HDCODE,1(SP)	
6455	036614	012662	000006			MOV	(SP)+,RKDA(R2)	
6456	036620	012737	000007	004266		MOV	#7,DRVTYPE	:LOAD DRIVE TYPE FOR PRINT OUT
6457	036626	012762	002017	000000		MOV	#COT!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	RO	
6462	036656	001370				BNE	1\$	
6463	036660	005062	000026			CLR	RKMR1(R2)	;ALLOW COMMAND TO FINISH
6464	036664	013700	004262			MOV	WAITIM,RO	;LOAD WAIT TIME
6465	036670	105762	000000		2\$:	TSTB	RKCS1(R2)	;WAIT FOR READY
6466	036674	100402				BMI	3\$	
6467	036676	005300				DEC	RO	
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	;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	TS*73	;YES, GO ON TO NEXT TEST

6515 037212 104227

ERROR 227 ;ERROR REG. INCORRECT

```

*****
;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 DIAGNOSTIC 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.
;

```

```

*****
;ST73: SCOPE

```

6529	037214	C00004				MOV	#100, \$TIMES	;; DO 100. ITERATIONS
6530	037216	012737	000144	001200		MOV	\$BASE, R2	;; LOAD RK611 BASE
6531	037224	013702	001270			MOV	#SCLR, RKCS2(R2)	;; CLEAR RK06 SUBSYSTEM
6532	037230	012762	000040	000010		MOV	#DMD, RKMR1(R2)	;; PUT RK611 IN MAINTENANCE MODE
6533	037236	012762	000040	000026		MOV	#23, RKDCYL(R2)	;; LOAD CYLINDER ADDRESS
6534	037244	012762	000023	000020		MOV	#23, CYLIN	
6535	037252	012737	000023	004252		MOV	#F, HDCCODE	;; LOAD HEAD ADDRESS
6536	037260	012737	000006	004250		MOV	-(SP)	
6537	037266	005046				CLR	HDCCODE, 1(SP)	
6538	037270	013766	004250	000001		MOVB	(SP)+, RKDA(R2)	
6539	037276	012662	000006			MOV	#7, DRVTYP	;; LOAD DRIVE TYPE FOR PRINT OUT
6540	037302	012737	000007	004266		MOV	#CDT!SEEK, RKCS1(R2)	;; ISSUE SEEK TO RK06
6541	037310	012762	002017	000000		MOV	#22, #4+2, RO	;; ISSUE CLOCK TO GET THROUGH PHASE 6
6542	037316	012700	000132			MOV	#DMD!MCLK, RKMR1(R2)	
6543	037322	012762	000440	000026	1\$:	MOV	#DMD, RKMR1(R2)	
6544	037330	012762	000040	000026		DEC	RO	
6545	037336	005300				BNE	1\$	
6546	037340	013702				CLR	RKMR1(R2)	;; ALLOW COMMAND TO FINISH
6547	037342	005062	000026			MOV	WAITIM, RO	;; LOAD WAIT TIME
6548	037346	013700	004262			TSTB	RKCS1(R2)	;; WAIT FOR READY
6549	037352	105762	000000		2\$:	BMI	3\$	
6550	037356	100402				DEC	RO	
6551	037360	005300				BNE	2\$	
6552	037362	001373				MOV	RKCS1(R2), T.CS1	;; STORE COMMAND AND STATUS REG.1
6553	037364	016237	000000	004120	3\$:	MOV	RKCS2(R2), T.CS2	;; STORE COMMAND AND STATUS REG.2
6554	037372	016237	000010	004130		MOV	RKDS(R2), T.DS	;; STORE DRIVE STATUS REG
6555	037400	016237	000012	004132		MOV	RKER(R2), T.ER	;; STORE ERROR REG
6556	037406	016237	000014	004134		MOV	#CERR!CDT!RDY!<SEEK&+C<GO>>, E.CS1	;; LOAD EXPECTED CS1
6557	037414	012737	102216	004160		MOV	#IR, E.CS2	;; LOAD EXPECTED COMMAND AND STATUS REG.2
6558						MOV	#SVAL!DRA!DROT!DOT, E.DS	;; LOAD EXPECTED DRIVE STATUS REG
6559	037422	012737	000100	004170		MOV	#UNS!IDAE, E.ER	;; LOAD EXPECTED ERROR REG
6560	037430	012737	100441	004172		CMP	E.CS1, T.CS1	;; CHECK COMMAND AND STATUS REG1 CORRECT
6561	037436	012737	042000	004174		BEQ	4\$	;; YES, CHECK CS2
6562	037444	023737	004160	004120		ERROR	230	;; CS1 INCORRECT
6563	037452	001401				CMP	E.CS2, T.CS2	;; CHECK COMMAND AND STATUS REG2 CORRECT
6564	037454	104230			4\$:	BEQ	5\$	;; YES, CHECK DRIVE STATUS REG.
6565	037456	023737	004170	004130		ERROR	231	;; CS2 INCORRECT
6566	037464	001401				CMP	E.DS, T.DS	;; CHECK DRIVE STATUS REG. CORRECT
6567	037466	104231			5\$:	BEQ	6\$	;; YES, CHECK ERROR REG
6568	037470	023737	004172	004132		ERROR	232	;; DRIVE STATUS REG. INCORRECT
6569	037476	001401						
6570	037500	104232						

```

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 004244 MOV $BASE,R2 ;LOAD RK611 BASE
6616 037712 012737 000001 004244 MOV #1,DRVCD ;LOAD INITIAL DRIVE CODE
6617 037720 012737 037726 001110 MOV #1,$SLPERR ;LOAD LOOP ON ERROR LOCATION FOR
6618 ; SUBTEST LOOP
6619

```

```

6620 037726 1$: MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6621 037726 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
6622 037734 012762 000040 000026 MOV #1757,RKDCYL(R2) ;LOAD CYLINDER ADDRESS REG
6623 037742 012762 001757 000020 MOV #3400,RKDA(R2) ;LOAD HEAD 7
6624 037750 012762 003400 000006 MOV DRVCD,RKCS2(R2) ;LOAD DRIVE NUMBER
6625 037756 013762 004244 000010 MOV #CDT!SEEK,RKCS1(R2) ;ISSUE A SEEK WITH CDT SET
6626 037764 012762 002017 000000

```

## G10

CZP6BCD RK611 DSKLS CTRL PRT2  
CZP6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T7402-DEC-77 09:31 PAGE 123  
NON-STANDARD MESSAGE RECEIVING

SEQ 0123

```

6627 037772 012700 000132      MOV      #22, *4+2, R0      ;ISSUE CLOCKS THROUGH PHASE 6
6628 037776 012762 000440      MOV      #DMD!MCLK, RKMR1(R2)
6629 040004 012762 000040      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      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      SCOP1     ;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 *
6668 * CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR.
6669 * PUT THE RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE
6670 * A SEEK TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 2,
6671 * HEAD 0, DRIVE 1. CLOCK IN DIAGNOSTIC MODE UNTIL
6672 * PHASE ADDRESS 6. TURN OFF DIAGNOSTIC MODE AND MAKE
6673 * SURE DRIVE BUS PARITY ERROR AND CONTROLLER ERROR SETS.
6674 *
6675 *****
6675 040210 000004      ST75: 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

```

H10

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046)  
T7502-DEC-77 09:31 PAGE 124  
DRIVE BUS PARITY ON NON-STANDARD MESSAGE

SEQ 0124

6683	040262	012700	000132			MOV	#22,*4+2,R0	:ISSUE CLOCKS THROUGH PHASE 6
6684	040266	012762	000440	000026	1\$:	MOV	#0MD:MCLK,RKMR1(R2)	
6685	040274	012762	000040	000026		MOV	#0MD,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!SEK&1C<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. INCORRECT
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

6739  
6740  
6741  
6742  
6743  
6744  
6745  
6746  
6747  
6748  
6749  
6750  
6751  
6752

6753	040636	000004		
6754	040640	012737	000144	001200
6755	040646	013702	001270	
6756	040652	012762	000040	000010
6757	040660	012762	000040	000026
6758	040666	012762	000001	000000
6759	040674	012700	000124	
6760	040700	012762	000440	000026
6761	040706	012762	000040	000026
6762	040714	005300		
6763	040716	001370		
6764	040720	005062	000026	
6765	040724	013700	004262	
6766	040730	105762	000000	
6767	040734	100402		
6768	040736	005300		
6769	040740	001373		
6770	040742	013700	004264	
6771	040746	005300		
6772	040750	001376		
6773	040752	016237	000000	004120
6774	040760	016237	000010	004130
6775	040766	016237	000012	004132
6776	040774	016237	000014	004134
6777	041002	012737	100200	004160
6778	041010	032737	020000	004120
6779	041016	001403		
6780	041020	052737	020000	004160
6781	041026	012737	010100	004170
6782	041034	012737	100000	004172
6783	041042	005037	004174	
6784	041046	023737	004160	004120
6785	041054	001401		
6786	041056	104244		
6787	041060	023737	004170	004130
6788	041066	001401		
6789	041070	104245		
6790	041072	023737	004172	004132
6791	041100	001401		
6792	041102	104246		
6793	041104	023737	004174	004134
6794	041112	001401		

\*\*\*\*\*  
: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.  
\*\*\*\*\*

\*\*\*\*\*  
:ST76: SCOPE  
:DO 100. ITERATIONS  
:MOV \$BASE,R2 ;LOAD RK611 BASE  
:MOV \$SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM  
:MOV \$DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE  
:MOV \$SELDV,RKCS1(R2) ;ISSUE SELECT DRIVE  
:MOV \$21,\$4,R0 ;ISSUE CLOCKS THROUGH PHASE 4  
:MOV \$DMD,\$MCLK,RKMR1(R2)  
:MOV \$DMD,RKMR1(R2)  
:DEC R0  
:BNE 1\$  
:CLR RKMR1(R2) ;ALLOW COMMAND TO FINISH  
:MOV WAITIM,R0 ;LOAD WAIT TIME  
:TSTB RKCS1(R2) ;WAIT FOR READY  
:BMI 3\$  
:DEC R0  
:BNE 2\$  
:MOV STALL,R0 ;STALL 100 USEC FOR MESSAGE TIME OUT  
:DEC R0  
:BNE 4\$  
: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 RKR(R2),T.ER ;STORE ERROR REG  
:MOV \$CERR,\$RDY,E.CS1 ;LOAD EXPECTED CS1  
:BIT \$SPAR,T.CS1 ;CHECK FOR BUS PARITY ERROR  
:BEQ 5\$  
:BIS \$SPAR,E.CS1 ;PUT BUS PARITY ERROR IN EXPECTED CS1  
:MOV \$NED,\$IR,E.CS2 ;LOAD EXPECTED CS2  
:MOV \$SVAL,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 6\$ ;YES, CHECK CS2  
:ERROR 244 ;CS1 INCORRECT  
:CMP E.CS2,T.CS2 ;CHECK COMMAND AND STATUS REG.2 CORRECT  
:BEQ 7\$ ;YES, CHECK DRIVE STATUS REG  
:ERROR 245 ;CS2 INCORRECT  
:CMP E.DS,T.DS ;CHECK DRIVE STATUS REG CORRECT  
:BEQ 8\$ ;YES, CHECK ERROR REG.  
:ERROR 246 ;DRIVE STATUS INCORRECT  
:CMP E.ER,T.ER ;CHECK ERROR REG CORRECT  
:BEQ 9\$ ;YES, ISSUE CONTROLLER CLEAR  
\*\*\*\*\*

```

6795 041114 104247          ERROR 247          ;ERROR REG INCORRECT
6796 041116 013737 004120 004220 9$: MOV T.CS1,P.CS1 ;STORE PREVIOUS CS1,CS2
6797 041124 013737 004130 004222 MOV T.CS2,P.CS2 ;DRIVE STATUS REG.,
6798 041132 013737 004132 004224 MOV T.DS,P.DS ;AND ERROR REG.
6799 041140 013737 004134 004226 MOV T.ER,P.ER
6800 041146 012762 100000 000000 MOV #CCLAR,KCS1(R2) ;ISSUE CONTROLLER CLEAR
6801 041154 016237 000000 004120 MOV RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG.1
6802 041162 016237 000010 004130 MOV RKCS2(R2),T.CS2 ;STORE COMMAND AND STATUS REG.2
6803 041170 016237 000012 004132 MOV RKDS(R2),T.DS ;STORE DRIVE STATUS REG.
6804 041176 016237 000014 004134 MOV RKER(R2),T.ER ;STORE ERROR REG.
6805 041204 012737 000200 004160 MOV #RDY,E.CS1 ;LOAD EXPECTED CS1
6806 041212 012737 000100 004170 MOV #IR,E.CS2 ;LOAD EXPECTED CS2
6807 041220 005037 004172 CLR E.DS ;LOAD EXPECTED DRIVE STATUS REG.
6808 041224 005037 004174 CLR E.ER ;LOAD EXPECTED ERROR REG.
6809 041230 023737 004160 004120 CMP E.CS1,T.CS1 ;CHECK COMMAND AND STATUS REG1 CORRECT
6810 041236 001401 BEQ 10$ ;YES, CHECK CS2
6811 041240 104224 ERROR 224 ;CS1 INCORRECT
6812 041242 023737 004170 004130 10$: CMP E.CS2,T.CS2 ;CHECK COMMAND AND STATUS REG.2 CORRECT
6813 041250 001401 BEQ 11$ ;YES, CHECK DRIVE STATUS REG.
6814 041252 104225 ERROR 225 ;CS2 INCORRECT
6815 041254 023737 004172 004132 11$: CMP E.DS,T.DS ;CHECK DRIVE STATUS REG CORRECT
6816 041262 001401 BEQ 12$ ;YES, CHECK ERROR REG
6817 041264 104226 ERROR 226 ;DRIVE STATUS INCORRECT
6818 041266 023737 004174 004134 12$: CMP E.ER,T.ER ;CHECK ERROR REG CORRECT
6819 041274 001401 BEQ TST77 ;YES, GO ON TO NEXT TEST
6820 041276 104227 ERROR 227 ;ERROR MESSAGE INCORRECT
6821
6822 *****
6823 *TEST 77 NON-EXISTENT DRIVE AND NO SACK
6824 *
6825 * CLEAR THE RK06 SUBSYSTEM WITH A SUBSYSTEM CLEAR. PUT
6826 * THE RK611 CONTROLLER IN DIAGNOSTIC MODE. ISSUE A
6827 * SELECT TO AN RK06 IN 26 SECTOR FORMAT, CYLINDER 0,
6828 * HEAD 0, DRIVE 0. CLOCK IN DIAGNOSTIC MODE UNTIL
6829 * PHASE ADDRESS 4. TURN OFF DIAGNOSTIC MODE AND MAKE SURE
6830 * NON-EXISTENT DRIVE AND CONTROLLER ERROR ARE SET.
6831 *
6832 * THIS TEST EXERCISES THE NON-EXISTENT DRIVE LOGIC
6833 * DUE TO RELEASE BIT RESET AND SACK RESET BUT THE PASSING
6834 * OF THIS TEST DOES GUARENTEE THAT THIS SITUATION DID
6835 * INDEED CAUSE A NON-EXISTENT DRIVE.
6836 *
6837 *****
6838 TST77: SCOPE
6839 041300 000004 MOV #100,$TIMES ;DO 100. ITERATIONS
6840 041302 012737 000144 001200 MOV $BASE,R2 ;LOAD RK611 BASE
6841 041310 013702 001270 MOV #SCLR,RKCS2(R2) ;CLEAR RK06 SUBSYSTEM
6842 041314 012762 000040 000010 MOV #DMD,RKMR1(R2) ;PUT RK611 IN MAINTENANCE MODE
6843 041322 012762 000040 000026 MOV #SELDV,RKCS1(R2) ;ISSUE SELECT DRIVE
6844 041330 012762 000001 000000 MOV #19,*4+2,R0 ;ISSUE CLOCKS THROUGH PHASE 3
6845 041342 012762 000440 000026 1$: MOV #DMD,MCLK,RKMR1(R2)
6846 041350 012762 000040 000026 MOV #DMD,RKMR1(R2)
6847 041356 005300 DEC R0
6848 041360 001370 BNE 1$
6849 041362 005062 000026 CLR RKMR1(R2) ;ALLOW COMMAND TO FINISH
6850 041366 013700 004262 MOV WAITIM,R0 ;LOAD WAIT TIME

```



## K10

CZR6BCD RK611 DSKLS CTRL PRT2 MACY11 30(1046) 02-DEC-77 09:31 PAGE 127  
 CZR6BC.P11 02-DEC-77 09:22 T77 NON-EXISTENT DRIVE AND NO SACK

SEQ 0127

```

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

```

.SBTTL \*\*ILLEGAL FUNCTION CODE TEST

```

*****
;TEST 100 ILLEGAL FUNCTION CODE
;
;
; CLEAR RK611 WITH A CONTROLLER CLEAR. ISSUE AN ILLEGAL

```

6900  
6901  
6902  
6903  
6904  
6905  
6906



```

6907      ;*      COMMAND IN NORMAL MODE AND MAKE SURE COMMAND FINISHES
6908      ;*      SETTING CONTROLLER READY WITH PROPER ERROR CONDITIONS.
6909      ;*
6910      ;*****
6911      041712 000004      ST100: SCOPE
6912      041714 012737 000764 001200      MOV      #500.,$TIMES      ;DO 500. ITERATIONS
6913      041722 012737 000033 004270      MOV      #33,ILLFUN      ;SET ILLEGAL FUNCTION
6914      041730 012737 041736 001110      MOV      #1$,SLPERR      ;LOAD LOOP ON ERROR LOCATION FOR
6915      ; SUBTEST LOOP
6916
6917      041736      1$:
6918      041736 012762 100000 000000      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
6919      041744 013737 004270 004160      MOV      ILLFUN,E.CS1 ;GENERATE EXPECTED CS1
6920      041752 042737 000001 004160      BIC      #GO,E.CS1
6921      041760 052737 100200 004160      BIS      #CERR!RDY,E.CS1
6922      041766 012737 000001 004174      MOV      #ILF,E.ER ;LOAD EXPECTED ERROR REG
6923      041774 012762 000040 000026      MOV      #DMD,RKMR1(R2) ;PUT RK611 IN DIAGNOSTIC MODE
6924      042002 013762 004270 000000      MOV      ILLFUN,RKCS1(R2) ;ISSUE ILLEGAL FUNCTION
6925      042010 016237 000000 004120      MOV      RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG 1
6926      042016 016237 000014 004134      MOV      RKER(R2),↑.ER ;STORE ERROR REG
6927      042024 023737 004160 004120      CMP      E.CS1,T.CS1 ;CHECK IF CS1 CORRECT
6928      042032 001401      BEQ      3$ ;YES, CHECK ERROR REG
6929      042034 104256      ERROR 256 ;CS1 INCORRECT AFTER ILL FUNCT
6930      042036 023737 004174 004134 3$:      CMP      E.ER,T.ER ;CHECK IF ERROR REG CORRECT
6931      042044 001401      BEQ      4$ ;YES, CLEAR CONTROLLER
6932      042046 104257      ERROR 257 ;ERROR REG INCORRECT AFTER ILL FUNCT
6933      042050 012762 100000 000000 4$:      MOV      #CCLR,RKCS1(R2) ;CLEAR RK611 CONTROLLER
6934      042056 016237 000000 004120      MOV      RKCS1(R2),T.CS1 ;STORE COMMAND AND STATUS REG. 1
6935      042064 016237 000014 004134      MOV      RKER(R2),↑.ER ;STORE ERROR REG
6936      042072 012737 000200 004160      MOV      #RDY,E.CS1 ;LOAD EXPECTED CS1
6937      042100 005037 004174      CLR      E.ER ;LOAD EXPECTED ERROR REG
6938      042104 023737 004160 004120      CMP      E.CS1,T.CS1 ;CHECK IF CS1 CORRECT (CERR CLEAR)
6939      042112 001401      BEQ      6$ ;YES, CHECK IF ERROR REG CORRECT
6940      042114 104260      ERROR 260 ;CONTROL CLEAR DID NOT CLEAR CERR
6941      042116 023737 004174 004134 6$:      CMP      E.ER,T.ER ;CHECK IF ILF CLEARED
6942      042124 001401      BEQ      7$ ;YES, GO ON TO NEXT CONFIGURATION
6943      042126 104261      ERROR 261 ;CONTROLLER CLEAR DID NOT CLEAR ILF
6944      042130 104415 7$:      SCOP1 ;CHECK IF LOOP ON ERROR
6945      042132 062737 000002 004270      ADD      #2,ILLFUN ;GENERATE NEXT ILLEGAL FUNCTION
6946      042140 022737 000041 004270      CMP      #41,ILLFUN ;CHECK IF FINISHED
6947      042146 101273      BHI      1$ ;NO, USE NEXT CONFIGURATION
6948

```

M10

SEQ 0129

.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 THERES A MONITOR GO TO IT  
*IF THERE ISN'T JUMP TO NEWPAS
```

\$EOP:

```
SCOPE  
CLR $STNM ; ZERO THE TEST NUMBER  
CLR $TIMES ; ZERO THE NUMBER OF ITERATIONS  
INC $PASS ; INCREMENT THE PASS NUMBER  
BIC #100000,$PASS ; DON'T ALLOW A NEG. NUMBER  
DEC (PC)+ ; LOOP?  
$EOPCT: .WORD 1  
BGT $DOAGN ; YES  
MOV (PC)+,$(PC)+ ; RESTORE COUNTER  
$ENDCT: .WORD 1  
$EOPCT  
TYPE $65$ ; TYPE ASCIZ STRING  
BR $64$ ; GET OVER THE ASCIZ  
$65$: .ASCIZ <12><15>/END PASS #/  
$64$: MOV $PASS,-(SP) ; SAVE $PASS FOR TYPEOUT  
; TYPE PASS NUMBER  
; GO TYPE--DECIMAL ASCII WITH SIGN  
TYPDS $67$  
TYPE $66$ ; TYPE ASCIZ STRING  
BR $66$ ; GET OVER THE ASCIZ  
$67$: .ASCIZ / TOTAL ERRORS SINCE LAST REPORT /  
$66$: MOV $ERTTL,-(SP) ; SAVE $ERTTL FOR TYPEOUT  
; TOTAL NUMBER OF ERRORS  
; GO TYPE--DECIMAL ASCII WITH SIGN  
TYPDS $CRLF  
TYPE $ERTTL ; TYPE CARRIAGE RETURN, LINE FEED  
CLR $ERTTL ; CLEAR ERROR TOTAL  
$GET42: MOV $42,PC ; GET MONITOR ADDRESS  
BEQ $DOAGN ; BRANCH IF NO MONITOR  
RESET ; CLEAR THE WORLD  
$ENDAD: JSR PC,(R0) ; GO TO MONITOR  
NOP ; SAVE ROOM  
NOP ; FOR  
NOP ; ACT11  
$DOAGN: JMP $(PC)+ ; RETURN  
$RTNAD: .WORD NEWPAS  
$ENULL: .BYTE -1,-1,0 ; NULL CHARACTER STRING  
; .EVEN
```

.SBTTL CHECK FOR MEMORY CHECK ENABLE OPTION

```
CHKPAR: MOV #20,$ERRVEC ; SET VECTOR FOR MEMORY PARITY CHECK  
MOV #PR7,$ERRVEC+2  
MOV #MEMBAS,R3 ; LOAD REGISTER TO DETERMINE IF  
; MEMORY CHECK ENABLE AVAILIABLE
```

```
6949  
6950  
6951  
6952  
6953  
6954  
6955  
6956  
6957  
6958 042150  
6959 042150 000004  
6960 042152 005037 001102  
6961 042156 005037 001200  
6962 042162 005237 001222  
6963 042166 042737 100000 001222  
6964 042174 005327  
6965 042176 000001  
6966 042200 003063  
6967 042202 012737  
6968 042204 000001  
6969 042206 042176  
6970 042210 104401 042216  
6971 042214 000407  
6972  
6973 042234  
6974 042234 013746 001222  
6975  
6976 042240 104405  
6977 042242 104401 042250  
6978 042246 000421  
6979  
6980 042312  
6981 042312 013746 001112  
6982  
6983 042316 104405  
6984 042320 104401 001211  
6985 042324 005037 001112  
6986 042330 013700 000042  
6987 042334 001405  
6988 042336 000005  
6989 042340 004710  
6990 042342 000240  
6991 042344 000240  
6992 042346 000240  
6993 042350  
6994 042350 000137  
6995 042352 005254  
6996 042354 377 377 000  
6997 042360  
6998  
6999  
7000  
7001 042360 012737 042432 000004  
7002 042366 012737 000340 000006  
7003 042374 012703 172100  
7004
```

```

7005 042400 012704 000020      MOV    #16,R4      ;LOAD COUNT
7006 042404 012723 000001      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      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      CLR    $ESCAPE ;CLEAR ESCAPE
7024 042470 032777 001000 136442 10$: 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      MOV    #STACK, SP ;INITIALIZE STACK
7029 042506 000177 136376      JMP    $SLPERA ;LOOP ON ERROR
7030
7031      .SBTTL  SCOPE HANDLER ROUTINE
7032
7033      ;*****
7034      ;THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
7035      ;AND LOAD THE TEST NUMBER($STNM) 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      $SCOPE:
7046 042512 104407      CKSWR
7047 042514 032777 040000 13641E 1$: BIT    #BIT14, $SWR ;TEST FOR CHANGE IN SOFT-SWR
7048 042522 001131      BNE    $OVER ;LOOP ON PRESENT TEST?
7049      ;*****START OF CODE FOR THE XOR TESTER*****
7050 042524 000416      $XTSTR: BR     6$      ;YES IF SW14=1
7051      ;IF RUNNING ON THE "XOR" TESTER CHANGE
7052 042526 013746 000004      MOV    @#ERRVEC, -(SP) ;THIS INSTRUCTION TO A "NOP" (NOP=240)
7053 042532 012737 042552 000004      MOV    #5$, @#ERRVEC ;SAVE THE CONTENTS OF THE ERROR VECTOR
7054 042540 005737 177060      TST    @#177060 ;SET FOR TIMEOUT
7055 042544 012637 000004      MOV    (SP)+, @#ERRVEC ;TIME OUT ON XOR?
7056 042550 000500      BR     $SVLAD ;RESTORE THE ERROR VECTOR
7057 042552 022626      CMP    (SP)+,(SP)+ ;GO TO THE NEXT TEST
7058 042554 012637 000004      MOV    (SP)+, @#ERRVEC ;CLEAR THE STACK AFTER A TIME OUT
7059 042560 000440      BR     7$      ;RESTORE THE ERROR VECTOR
7060 042562      6$: ; *****END OF CODE FOR THE XOR TESTER*****

```

7061	042562	032777	000400	136350		BIT	#BIT08,2SWR	:: LOOP ON SPEC. TEST?
7062	042570	001421				BEQ	25	:: BR IF NO
7063	042572	005046				CLR	-(SP)	:: CLEAR A TEMP. LOCATION
7064	042574	117716	136340			MOVB	2SWR,(SP)	:: PICKUP THE DESIRED TEST NUMBER
7065	042600	001414				BEQ	85	:: BRANCH IF BAD TEST NUMBER IN SWR
7066	042602	022716	000100			CMP	#100,(SP)	:: CHECK THE NUMBER IN THE SWR
7067	042606	002411				BLT	85	:: BRANCH IF TEST NUMBER IS OUT OF RANGE
7068	042610	011637	001102			MOV	(SP), \$STNM	:: 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	2(SP)+,\$LPADR	:: SET LOOP ADDRESS TO DESIRED TEST
7073	042630	000466				BR	\$OVER	:: GO LOOP ON THE TEST
7074	042632	005726			85:	TST	(SP)+	:: CLEAN THE BAD TEST NUMBER OFF OF THE STACK
7075	042634	105737	001103		25:	TSTB	\$ERFLG	:: HAS AN ERROR OCCURRED?
7076	042640	001421				BEQ	35	:: BR IF NO
7077	042642	123737	001115	001103		CMPB	\$ERMAX,\$ERFLG	:: MAX. ERRORS FOR THIS TEST OCCURRED?
7078	042650	101015				BHI	35	:: BR IF NO
7079	042652	032777	001000	136260		BIT	#BIT09,2SWR	:: LOOP ON ERROR?
7080	042660	001404				BEQ	45	:: BR IF NO
7081	042662	013737	001110	001106	75:	MOV	\$LPERR,\$LPADR	:: SET LOOP ADDRESS TO LAST SCOPE
7082	042670	000446				BR	\$OVER	
7083	042672	105037	001103		45:	CLRB	\$ERFLG	:: ZERO THE ERROR FLAG
7084	042676	005037	001200			CLR	\$TIMES	:: CLEAR THE NUMBER OF ITERATIONS TO MAKE
7085	042702	000415				BR	15	:: ESCAPE TO THE NEXT TEST
7086	042704	032777	004000	136226	35:	BIT	#BIT11,2SWR	:: INHIBIT ITERATIONS?
7087	042712	001011				CNE	15	:: BR IF YES
7088	042714	005737	001222			TST	\$PASS	:: IF FIRST PASS OF PROGRAM
7089	042720	001406				BEQ	15	:: 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	15:	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	\$STNM	:: COUNT TEST NUMBERS
7096	042756	113737	001102	001220		MOVB	\$STNM,\$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	\$STNM,\$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      $SW9,2SWR      ;CHECK IF LOOP ON ERROR
7174 043232 001405          BEQ      $S          ;NO, CONTINUE
7175 043234 105737 001103          TSTB     $ERFLG      ;CHECK IF ERROR OCCURRED
7176 043240 001402          BEQ      $S          ;NO, CONTINUE
7177 043242 013716 001110          MOV      $LPERR,(SP)    ;LOAD ERROR RETURN
7178 043246 000002          $S:      RTI              ;RETURN
7179          .SBTTL   APT COMMUNICATIONS ROUTINE
7180
7181 *****
7182 043250 112737 000001 043514 $ATY1:  MOVB     #1,$FFLG      ;;TO 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      RO,-(SP)      ;;PUSH RO 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      $S          ;;IF NOT: BR
7191 043306 122737 000001 001234 CMPB     $APTENV,$ENV      ;;OPERATING UNDER APT?
7192 043314 001031          BNE      $S          ;;IF NOT: BR
7193 043316 132737 000100 001235 BITB     $APTPOOL,$ENVM    ;;SHOULD SPOOL MESSAGES?
7194 043324 001425          BEQ      $S          ;;IF NOT: BR
7195 043326 017600 000004          MOV      24(SP),RO      ;;GET MESSAGE ADDR.
7196 043332 062766 000002 000004 ADD      #2,4(SP)      ;;BUMP RETURN ADDR.
7197 043340 005737 001214          TST      $MSGTYPE      ;;SEE IF DONE W/ LAST XMISSION?
7198 043344 001375          BNE      $S          ;;IF NOT: WAIT
7199 043346 010037 001230          MOV      RO,$MSGAD      ;;PUT ADDR IN MAILBOX
7200 043352 105720 2$:          TSTB     (RO)+      ;;FIND END OF MESSAGE
7201 043354 001376          BNE      2$
7202 043356 163700 001230          SUB      $MSGAD,RO      ;;SUB START OF MESSAGE
7203 043362 006200          ASR      RO          ;;GET MESSAGE LGTH IN WORDS
7204 043364 010037 001232          MOV      RO,$MSGGLT      ;;PUT LENGTH IN MAILBOX
7205 043370 012737 000004 001214 MOV      #4,$MSGTYPE      ;;TELL APT TO TAKE MSG.
7206 043376 000413          BR         $S
7207 043400 017637 000004 043424 3$:  MOV      24(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      24(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)+,RO      ;;POP STACK INTO RO
7227 043510 000207          RTS      PC          ;;RETURN
7228 043512 000          $MFLG:  .BYTE      0      ;;MESSG. FLAG

```

E11

SEQ 0134

```

7229 043513 000 $LFLG: .BYTE 0 ::LOG FLAG
7230 043514 000 $FFLG: .BYTE 0 ::FATAL FLAG
7231 043516 EVEN
7232 000200 APTSIZE=200
7233 000001 APTENV=001
7234 000100 APTSPool=100
7235 00004C 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 $ERROR:
7251 043516 104407 001103 7$: CKSWR ::TEST FOR CHANGE IN SOFT-SWR
7252 043520 105237 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 $TSTNM,$DISP ::DISPLAY TEST NUMBER AND ERROR FLAG
7255 043534 032777 002000 135376 TIT $BIT10,$SWR BELL ON ERROR?
7256 043542 001402 IS 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 INSTRUCTION
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 , $CRLF
7266 C43614 20$: CMPB $APTENV, $ENV ::RUNNING IN APT MODE
7267 043614 122737 000001 001234 BNE 2$ NO SKIP APT ERROR REPORT
7268 043622 001007 MOV $ITEMB, 21$ SET ITEM NUMBER AS ERROR NUMBER
7269 043624 113737 001114 043636 JSR PC, $ATY4 REPORT FATAL ERROR TO APT
7270 043632 004737 043266 21$: .BYTE 0
7271 043636 000 .BYTE 0
7272 043637 000 22$: BR 22$ ::APT ERROR LOOP
7273 043640 000777 135272 2$: TST $SWR HALT ON ERROR
7274 043642 005777 BPL 3$ SKIP IF CONTINUE
7275 043646 100002 HALT ON ERROR!
7276 043650 000000 CKSWR ::TEST FOR CHANGE IN SOFT-SWR
7277 043652 104407 001000 135256 3$: BIT $BIT09,$SWR LOOP ON ERROR SWITCH SET?
7278 043654 032777 001402 BEQ 4$ BR IF NO
7279 043662 001402 MOV $LPERR, (SP) FUDGE RETURN FOR LOOPING
7280 043664 013716 001110 4$: TST $ESCAPE CHECK FOR AN ESCAPE ADDRESS
7281 043670 007737 001202 BEQ 5$ BR IF NONE
7282 043674 001402 5$: MOV $ESCAPE, (SP) FUDGE RETURN ADDRESS FOR ESCAPE
7283 043676 013716 001202
7284 043702

```

```

7285 043702 022737 042340 000042      CMP      #SENDAD,2#42      ;;ACT-11 AUTO-ACCEPT?
7286 043710 001001                      BNE      6$              ;;BRANCH IF NO
7287 043712 000000                      HALT                      ;;YES
7288 043714                      6$:      RTI                      ;;RETURN
7289 043714 000002
7290
7291
7292
7293
7294
7295
7296
7297
7298
7299
7300
7301 043716 104413
7302 043720 113700 001114
7303 043724 042700 177400
7304 043730 005300
7305 043732 006300
7306 043734 006300
7307 043736 006300
7308 043740 062700 001300
7309 043744 012037 043760
7310 043750 001404
7311 043752 104401 001211
7312 043756 104401
7313 043760 000000
7314 043762 012037 043776
7315 043766 001404
7316 043770 104401 001211
7317 043774 104401
7318 043776 000000
7319 044000 012001
7320 044002 001445
7321 044004 005004
7322 044006 012000
7323 044010 012002
7324 044012 104401 001211
7325 044016 112003
7326 044020 10572C
7327 044022 005703
7328 044024 001416
7329 044026 005704
7330 044030 001004
7331 044032 013146
7332 044034 104402
7333 044036 005303
7334 044040 001403
7335 044042 104401 050117
7336 044046 000771
7337 044050 104401 001211
7338 044054 005710
7339 044056 001401
7340 044060 005104

      CMP      #SENDAD,2#42      ;;ACT-11 AUTO-ACCEPT?
      BNE      6$              ;;BRANCH IF NO
      HALT                      ;;YES
6$:      RTI                      ;;RETURN

;*****
;SBTTL TYPE ERROR ROUTINE
;ENTRY JSR PC TYPERR
;RETURN RTS PC
;
;THIS ROUTINE USES THE "ITEM CONTROL BYTE" ($ITEMB) TO DETERMINE WHICH
;ERROR IS TO BE REPORTED. IT THEN USES THE "ERROR TABLE" ($ERRTB)
;ENTRY TO DEFINE WHAT INFORMATION IS TO BE REPORTED CONCERNING
;THE ERROR.
;*****
TYPERR: SAVREG
      MOV      $ITEMB,RO          ;ENTER ERROR NUMBER
      BIC      #177400,RO        ;CLEAR UNUSED BITS
      DEC      RO                ;FORM INDEX FOR ERROR TABLE
      ASL      RO
      ASL      RO
      ASL      RO
1$:      ADD      #ERRTB,RO        ;FORM ADDRESS OF ERROR ENTRY
      MOV      (RO)+,2$          ;GET EM POINTER
      BEQ      3$                ;BRANCH IF THERE ISN'T ONE
      TYPE     ,SCRLF            ;TYPE CARRIAGE RETURN LINE FEED
      TYPE     ,SCRLF            ;TYPE ERROR MESSAGE (EM)
2$:      .WORD    0                ;EM POINTER GOES HERE
3$:      MOV      (RO)+,4$          ;GET DH POINTER
      BEQ      5$                ;BRANCH IF THERE ISN'T ONE
      TYPE     ,SCRLF            ;TYPE CR-LF
      TYPE     ,SCRLF            ;TYPE DATA HEADER
4$:      .WORD    0                ;DH POINTER GOES HERE
5$:      MOV      (RO)+,R1          ;GET DT POINTER
      BEQ      20$               ;BRANCH IF THERE ARE NONE
      CLR      R4                ;RESET INDENT SWITCH
      MOV      (RO)+,R2          ;GET DF POINTER
      MOV      (RO)+,R2          ;STORE NUMBER OF DH'S
      TYPE     ,SCRLF            ;TYPE <CR><LF>
10$:     MOV      (RO)+,R3          ;GET & STORE NUMBER OF DATA WORDS
      TST      (RO)+             ;BUMP PAST FORMAT WORD
      TST      R3                ;TEST IF ANY DATA FOR THIS HEADER
      BEQ      14$               ;NO - SKIP DATA PRINT
      TST      R4                ;CHECK FOR INDENT
      BNE      12$               ;YES, GO INDENT
11$:     MOV      2(R1)+,-(SP)      ;PUT FIRST DATA WORD ON STACK
      TYPE     IT                ;TYPE IT
      DEC      R3                ;MORE DATA WORDS
      BEQ      13$               ;NO-BRANCH
      TYPE     ,SPACE2           ;TYPE SEPARATORS
      BR       11$              ;LOOP
13$:     TYPE     ,SCRLF            ;TYPE <CR><LF>
      TST      (RO)              ;CHECK IF NEXT HEADER AVAILBLE
      BEQ      14$               ;NO, DO NOT CHANGE INDENT
      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$: MOV      (R0)+, 0$    ;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      TYPE      ;TYPE DH
7349 044106 000000      .WORD      0      ;DH POINTER GOES HERE
7350 044110 104401 001211 TYPE      ,SCLRF
7351 044114 000740      BR       10$     ;GO TYPE OUT DATA
7352 044116 104414      RESREG
7353 044120 005237 004242 INC      ERRCNT ;INCREMENT THE ERROR COUNT
7354 044124 032777 010000 135006 BIT      #SW12,2SWR ;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      BHS      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,SEOPCT ;FOR PASS COUNT FOR ABORT
7363 044164 012706 001100 MOV      #STACK,SP ;INITIALIZE STACK
7364 044170 000137 042.50 JMP      SEOP ;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                                ;*
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      #APTPOOL,$ENVM ;SPOOL MESSAGE TO APT
7394 044236 001405      BEQ      62$     ;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 000040 001235 61S: WORD 0 MESSAGE ADDRESS
7398 044252 132737 000040 001235 62S: BITB 0 APTCSUP,$ENVN ; APT CONSOLE SUPPRESSED
7399 044260 001003 000040 001235 60S BNE 60S ; YES, SKIP TYPE OUT
7400 044262 112046 000040 001235 2S: MOVB (RO)+,-(SP) ; PUSH CHARACTER TO BE TYPED ONTO STACK
7401 044264 001005 000040 001235 BNE 4S ; BR IF IT ISN'T THE TERMINATOR
7402 044266 005726 000040 001235 TST (SP)+ ; IF TERMINATOR POP IT OFF THE STACK
7403 044270 012600 000040 001235 60S: MOV (SP)+,RO ; RESTORE RO
7404 044272 062716 000002 001235 3S: ADD #2,(SP) ; ADJUST RETURN PC
7405 044276 000002 000002 001235 RTI ; RETURN
7406 044300 122716 000011 001235 4S: CMPB #HT,(SP) ; BRANCH IF <HT>
7407 044304 001430 000011 001235 BEQ 8S ; BRANCH IF NOT <CRLF>
7408 044306 122716 000200 001235 CMPB #CRLF,(SP) ; BRANCH IF NOT <CRLF>
7409 044312 001006 000200 001235 BNE 5S ; BRANCH IF NOT <CRLF>
7410 044314 005726 000200 001235 TST (SP)+ ; POP <CR><LF> EQUIV
7411 044316 104401 000200 001235 TYPE ; TYPE A CR AND LF
7412 044320 001211 000200 001235 SCRLF ; CLEAR CHARACTER COUNT
7413 044322 105037 044456 001235 CLAB ; GET NEXT CHARACTER
7414 044326 000755 044456 001235 BR 2S ; GO TYPE THIS CHARACTER
7415 044330 004737 044412 001235 5S: JSR PC,$TYPEC ; GO TYPE THIS CHARACTER
7416 044334 123726 001156 001235 6S: CMPB $FILLC,(SP)+ ; IS IT TIME FOR FILLER CHARS.?
7417 044340 001350 001156 001235 BNE 2S ; IF NO GO GET NEXT CHAR.
7418 044342 013746 001156 001235 MOV $NULL,-(SP) ; GET # OF FILLER CHARS. NEEDED
7419 044346 105366 000001 001235 7S: DECB 1(SP) ; AND THE NULL CHAR.
7420 044352 002770 000001 001235 BLT 6S ; DOES A NULL NEED TO BE TYPED?
7421 044354 004737 044412 001235 JSR PC,$TYPEC ; BR IF NO--GO POP THE NULL OFF OF STACK
7422 044360 105337 044456 001235 DECB $CHARCNT ; GO TYPE A NULL
7423 044364 000770 044456 001235 BR 7S ; DO NOT COUNT AS A COUNT
7424 044364 000770 044456 001235 ; LOOP
7425 044364 000770 044456 001235 ;
7426 044364 000770 044456 001235 ;
7427 044364 000770 044456 001235 ;
7428 044366 112716 000040 001235 8S: MOVB #' (SP) ; REPLACE TAB WITH SPACE
7429 044372 004737 044412 001235 9S: JSR PC,$TYPEC ; TYPE A SPACE
7430 044376 132737 000007 044456 001235 BITB #7,$CHARCNT ; BRANCH IF NOT AT
7431 044404 001372 000007 044456 001235 BNE 9S ; TAB STOP
7432 044406 005726 000007 044456 001235 TST (SP)+ ; POP SPACE OFF STACK
7433 044410 000724 000007 044456 001235 BR 2S ; GET NEXT CHARACTER
7434 044412 105777 134532 001235 $TYPEC: TSTB 2STPS ; WAIT UNTIL PRINTER IS READY
7435 044416 100375 134532 001235 BPL $TYPEC ;
7436 044420 116677 000002 134524 001235 MOVB 2(SP),2STPB ; LOAD CHAR TO BE TYPED INTO DATA REG.
7437 044426 122766 000015 000002 001235 CMPB #CR,2(SP) ; IS CHARACTER A CARRIAGE RETURN?
7438 044434 001003 000015 000002 001235 BNE 1S ; BRANCH IF NO
7439 044436 105037 044456 000015 000002 001235 CLAB $CHARCNT ; YES--CLEAR CHARACTER COUNT
7440 044442 000406 000015 000002 001235 BR $TYPEX ; EXIT
7441 044444 122766 000012 000002 1S: CMPB #LF,2(SP) ; IS CHARACTER A LINE FEED?
7442 044452 001402 000012 000002 001235 BEQ $TYPEX ; BRANCH IF YES
7443 044454 105227 000012 000002 001235 INCB (PC)+ ; COUNT THE CHARACTER
7444 044456 000000 000012 000002 001235 $CHARCNT: WORD 0 ; CHARACTER COUNT STORAGE
7445 044460 000207 000012 000002 001235 $TYPEX: RTS PC ;
7446 044460 000207 000012 000002 001235 ;
7447 044460 000207 000012 000002 001235 ;
7448 044460 000207 000012 000002 001235 ;
7449 044460 000207 000012 000002 001235 ;
7450 044460 000207 000012 000002 001235 ;
7451 044460 000207 000012 000002 001235 ;
7452 044460 000207 000012 000002 001235 ;

```

7453					*CALL:	MOV	NUM,- SP)	:: NUMBER TO BE TYPED
7454					*	TYPOS		:: CALL FOR TYPEOUT
7455					*	.BYTE	N	:: N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
7456					*	.BYTE	M	:: M=1 OR 0
7457					*			:: 1=TYPE LEADING ZEROS
7458					*			:: 0=SUPPRESS LEADING ZEROS
7459					*			
7460					*\$TYPON----	ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST		
7461					*\$TYPOS OR \$TYPOC			
7462					*CALL:	MOV	NUM,-(SP)	:: NUMBER TO BE TYPED
7463					*	TYPON		:: CALL FOR TYPEOUT
7464					*			
7465					*\$TYPOC----	ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER		
7466					*CALL:	MOV	NUM,-(SP)	:: NUMBER TO BE TYPED
7467					*	TYPOC		:: CALL FOR TYPEOUT
7468					*			
7469					\$TYPOS:	MOV	2(SP),-(SP)	:: PICKUP THE MODE
7470						MOV	1(SP), \$OFILL	:: LOAD ZERO FILL SWITCH
7471	044462	017646	000000			MOV	(SP)+, \$OMODE+1	:: NUMBER OF DIGITS TO TYPE
7472	044466	116637	000001	044705		ADD	2, (SP)	:: ADJUST RETURN ADDRESS
7473	044474	112637	044707			BR	\$TYPON	
7474	044500	062716	000002		\$TYPOC:	MOV	1, \$OFILL	:: SET THE ZERO FILL SWITCH
7475	044504	000406				MOV	6, \$OMODE+1	:: SET FOR SIX(6) DIGITS
7476	044506	112737	000001	044705	\$TYPON:	MOV	5, \$OCNT	:: SET THE ITERATION COUNT
7477	044514	112737	000006	044707		MOV	R3,-(SP)	:: SAVE R3
7478	044522	112737	000005	044704		MOV	R4,-(SP)	:: SAVE R4
7479	044530	010346				MOV	R5,-(SP)	:: SAVE R5
7480	044532	010446				MOV	\$OMODE+1, R4	:: GET THE NUMBER OF DIGITS TO TYPE
7481	044534	010546				MOV	R4	
7482	044536	113704	044707			NEG	R4	
7483	044542	005404				ADD	6, R4	:: SUBTRACT IT FOR MAX. ALLOWED
7484	044544	062704	000006			MOV	R4, \$OMODE	:: SAVE IT FOR USE
7485	044550	110437	044706			MOV	\$OFILL, R4	:: GET THE ZERO FILL SWITCH
7486	044554	113704	044705			MOV	12(SP), R5	:: PICKUP THE INPUT NUMBER
7487	044560	016605	000012			CLR	R3	:: CLEAR THE OUTPUT WORD
7488	044564	005003			1\$:	ROL	R5	:: ROTATE MSB INTO "C"
7489	044566	006105				BR	3\$	:: GO DO MSB
7490	044570	000404			2\$:	ROL	R5	:: FORM THIS DIGIT
7491	044572	006105				ROL	R5	
7492	044574	006105				ROL	R5	
7493	044576	006105				MOV	R5, R3	
7494	044600	010503			3\$:	ROL	R3	:: GET LSB OF THIS DIGIT
7495	044602	006103				DECB	\$OMODE	:: TYPE THIS DIGIT?
7496	044604	105337	044706			BPL	7\$	:: BR IF NO
7497	044610	100016				BIC	177770, R3	:: GET RID OF JUNK
7498	044612	042703	177770			BNE	4\$	:: TEST FOR 0
7499	044616	001002				TST	R4	:: SUPPRESS THIS 0?
7500	044620	005704				BEQ	5\$	:: BR IF YES
7501	044622	001403			4\$:	INC	R4	:: DON'T SUPPRESS ANYMORE 0'S
7502	044624	005204				BIS	1'0, R3	:: MAKE THIS DIGIT ASCII
7503	044626	052703	000060		5\$:	BIS	1', R3	:: MAKE ASCII IF NOT ALREADY
7504	044632	052703	000040			MOV	R3, 8\$	:: SAVE FOR TYPING
7505	044636	110337	044702			TYPE	8\$	:: GO TYPE THIS DIGIT
7506	044642	104401	044702		7\$:	DECB	\$OCNT	:: COUNT BY 1
7507	044646	105337	044704					

```

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          ES: 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          BS: .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          ;*      TYPDS      ;;GO TO THE ROUTINE
7535
7536          $TYPDS:
7537          MOV      R0,-(SP)      ;;PUSH R0 ON STACK
7538          MOV      R1,-(SP)      ;;PUSH R1 ON STACK
7539          MOV      R2,-(SP)      ;;PUSH R2 ON STACK
7540          MOV      R3,-(SP)      ;;PUSH R3 ON STACK
7541          MOV      R5,-(SP)      ;;PUSH R5 ON STACK
7542          MOV      #20200,-(SP)  ;;SET BLANK SWITCH AND SIGN
7543          MOV      20(SP),R5     ;;GET THE INPUT NUMBER
7544          BPL      1$           ;;BR IF INPUT IS POS.
7545          NEG      R5           ;;MAKE THE BINARY NUMBER POS.
7546          MOVB     #'-,1(SP)    ;;MAKE THE ASCII NUMBER NEG.
7547          CLR      R0           ;;ZERO THE CONSTANTS INDEX
7548          MOV      #SDBLK,R3    ;;SETUP THE OUTPUT POINTER
7549          MOVB     #'',(R3)+    ;;SET THE FIRST CHARACTER TO A BLANK
7550          CLR      R2           ;;CLEAR THE BCD NUMBER
7551          MOV      $DTBL(R0),R1  ;;GET THE CONSTANT
7552          SUB      R1,R5        ;;FORM THIS BCD DIGIT
7553          BLT      4$           ;;BR IF DONE
7554          INC      R2          ;;INCREASE THE BCD DIGIT BY 1
7555          BR      3$
7556          4$: ADD      R1,R5    ;;ADD BACK THE CONSTANT
7557          TST      R2          ;;CHECK IF BCD DIGIT=0
7558          BNE      5$         ;;FALL THROUGH IF 0
7559          TSTB     (SP)       ;;STILL DOING LEADING 0'S?
7560          BMI      7$         ;;BR IF YES
7561          ASLB      (SP)       ;;MSD?
7562          BCC      6$         ;;BR IF NO
7563          MOVB     1(SP),-1(R3) ;;YES--SET THE SIGN
7564          BIS      #'0,R2     ;;MAKE THE BCD DIGIT ASCII

```

```

7565 045024 052702 000040      7$: BIS      #' R2      ;; MAKE IT A SPACE IF NOT ALREADY A DIGIT
7566 045030 110223              MOV      R2,(R3)+    ;; PUT THIS CHARACTER IN THE OUTPUT BUFFER
7567 045032 005720              TST      (R0)+      ;; JUST INCREMENTING
7568 045034 020027 000010      CMP      R0,#10      ;; CHECK THE TABLE INDEX
7569 045040 002746              BLT      2$          ;; 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  MOV      -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 000002 000004  TYPE      $DBLK      ;; NOW TYPE THE NUMBER
7583 045102 016666              MOV      2(SP),4(SP)  ;; ADJUST THE STACK
7584 045110 012616              MOV      (SP)+,(SP)
7585 045112 000002              RTI
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  ITTY 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 ITTY 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      0$TKS      ;; CHAR THERE?
7604 045150 100071              BPL      15$          ;; IF NO, DON'T WAIT AROUND
7605 045152 117746 133770              MOV      0$TKB,-(SP)  ;; SAVE THE CHAR
7606 045156 042716 177600              BIC      #1C177,(SP)  ;; STRIP-OFF THE ASCII
7607 045162 022726 000007              CMP      #7,(SP)+    ;; IS IT A CONTROL G?
7608 045166 001062              BNE      15$          ;; 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      $GTSWR: TYPE      .SCNTLG      ;; ECHO THE CONTROL-G (↑G)
7613 045204 104401 046014              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$: TSTB      0$TKS      ;; CHAR THERE?
7620 045232 100375              BPL      7$          ;; IF NOT TRY AGAIN

```

```
7621
7622 045234 117746 133706          MOVB 2$TKB, -(SP)      ;; PICK UP CHAR
7623 045240 042716 177600          BIC  #1C177, (SP)    ;; MAKE IT 7-BIT ASCII
7624
7625
7626
7627 045244 021627 000025          9$: 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), 2$SWR      ;; SAVE NEW SWR
7639 045306 062706 000006          11$: 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, 2$TKS      ;; RE-ENABLE TTY KBD INTERRUPTS
7644 045334 000002                RTI                  ;; RETURN
7645 045336 004737 044412          16$: 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
7662
7663
7664
7665
7666
7667
7668
7669
7670
7671
7672 045416 011646 000004 000002  $RDCHR: MOV (SP), -(SP)  ;; PUSH DOWN THE PC
7673 045420 016666 000004 000002  MOV 4(SP), 2(SP)    ;; SAVE THE PS
7674 045426 105777 133512          1$: TSTB 2$TKS      ;; WAIT FOR
7675 045432 100375 133506 000004  BPL 1$              ;; A CHARACTER
7676 045434 117766 133506 000004  MOVB 2$TKB, 4(SP)    ;; READ THE TTY
```

```
;; *****
;; THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
;; *CALL:
;; * RDCHR
;; * RETURN HERE
;; * INPUT A SINGLE CHARACTER FROM THE TTY
;; * CHARACTER IS ON THE STACK
;; * WITH PARITY BIT STRIPPED OFF
```

```

7677 045442 042766 177600 000004      BIC      #1C<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$:      TSTB      2$TKS      ;; WAIT FOR A CHARACTER
7681 045464 100375          BPL          2$          ;; LOOP UNTIL ITS THERE
7682 045466 117746 133454      MOVB      2$TKB,-(SP)      ;; GET CHARACTER
7683 045472 042716 177600      BIC      #1C,77,(SP)      ;; MAKE IT 7-BIT ASCII
7684 045476 026627 000021      CMP      (SP)+,#21      ;; IS IT A CONTROL-Q?
7685 045502 001366          BNE          2$          ;; 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          $RDLIN: MOV      R3,-(SP)      ;; SAVE R3
7701 045540 005046          CLR      -(SP)      ;; CLEAR THE RUBOUT KEY
7702 045542 012703 045772      1$:      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$
7719 045630 000746          BR          2$          ;; GO TYPE
7720 045632 005716          TST      (SP)          ;; GO READ ANOTHER CHAR.
7721 045634 001406          BEQ          7$          ;; RUBOUT KEY SET?
7722 045636 112737 000134 045770      MOVB      #'\\,9$      ;; BR IF NO
7723 045644 104401 045770          TYPE      9$
7724 045650 005016          CLR      (SP)          ;; TYPE A BACK SLASH
7725 045652 122713 000025      7$:      CMPB      #25,(R3)      ;; CLEAR THE RUBOUT KEY
7726 045656 001003          BNE          8$          ;; IS CHARACTER A CTRL U?
7727 045660 104401 046002          TYPE      $CNTLU      ;; BR IF NO
7728 045664 000726          BR          1$          ;; TYPE A CONTROL "U"
7729 045666 122713 000022      8$:      CMPB      #22,(R3)      ;; GO START OVER
7730 045672 001011          BNE          3$          ;; IS CHARACTER A "↑R"?
7731 045674 105013          CLRB      (R3)          ;; BRANCH IF NO
7732 045676 104401 001211          TYPE      $CRLF      ;; CLEAR THE CHARACTER
                          ;; TYPE A "CR" & "LF"

```

```
7733 045702 104401 045772          TYPE      $TTYIN          ;;TYPE THE INPUT STRING
7734 045706 000717          BR      2$          ;;GO PICKUP ANOTHER CHARACTER
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          $MNEW: .ASCIZ  / NEW = /          ;;
7756 046025 040 053505          .SBTTL  READ AN OCTAL NUMBER FROM THE TTY
7757 046032 036440 000040          ;;*****
7758          ;;THIS ROUTINE WILL READ AN OCTAL (ASCII) NUMBER FROM THE TTY AND
7759          ;;CHANGE IT TO BINARY.
7760          ;;THE INPUT CHARACTERS WILL BE CHECKED TO INSURED THEY ARE LEGAL
7761          ;;OCTAL DIGITS. IF AN ILLEGAL CHARACTER IS READ A "?" WILL BE TYPED
7762          ;;FOLLOWED BY A CARRIAGE RETURN-LINE FEED. THE COMPLETE NUMBER MUST
7763          ;;THEN BE RETYPED. THE INPUT IS TERMINATED BY TYPING A CARRIAGE RETURN.
7764          ;;CALL:
7765          ;;      RDOCT          ;;READ AN OCTAL NUMBER
7766          ;;      RETURN HERE    ;;LOW ORDER BITS ARE ON TOP OF THE STACK
7767          ;;      *              ;;HIGH ORDER BITS ARE IN $HIOCT
7768          $RDOCT: MOV      (SP),-(SP)          ;;PROVIDE SPACE FOR THE
7769          MOV      4(SP),2(SP)          ;;INPUT NUMBER
7770          MOV      R0,-(SP)          ;;PUSH R0 ON STACK
7771          MOV      R1,-(SP)          ;;PUSH R1 ON STACK
7772          MOV      R2,-(SP)          ;;PUSH R2 ON STACK
7773          1$:      RDLIN          ;;READ AN ASCII LINE
7774          MOV      (SP)+,R0          ;;GET ADDRESS OF 1ST CHARACTER
7775          MOV      R0,5$          ;;AND SAVE IT
7776          CLR      R1          ;;CLEAR DATA WORD
7777          CLR      R2          ;;
7778          2$:      MOVB      (R0)+,-(SP)          ;;PICKUP THIS CHARACTER
7779          BEQ      3$          ;;IF ZERO GET OUT
7780          CMPB      #0,(SP)          ;;MAKE SURE THIS CHARACTER
7781          BGT      4$          ;;IS AN OCTAL DIGIT
7782          CMPB      #7,(SP)          ;;
7783          BLT      4$          ;;
7784          ASL      R1          ;;*2
7785          3$:      BEQ      3$          ;;
7786          4$:      BEQ      3$          ;;
7787          5$:      BEQ      3$          ;;
7788          6$:      BEQ      3$          ;;
```



```

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      #1C7, (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, $SHIOCT
7800 046146 012602      MOV      (SP)+, R2      ;; POP STACK INTO R2
7801 046150 012601      MOV      (SP)+, R1      ;; POP STACK INTO R1
7802 046152 012600      MOV      (SP)+, R0      ;; POP STACK INTO R0
7803 046154 000002      RTI      ;; RETURN
7804 046156 005726      4$: 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      $SHIOCT: .WORD 0      ;; HIGH ORDER BITS GO HERE
7811      .SBTTL  SAVE AND RESTORE R0-R5 ROUTINES
7812      ;*****
7813      ;SAVE R0-R5
7814      ;CALL:
7815      ;      SAVREG
7816      ;UPON RETURN FROM $SAVREG THE STACK WILL LOOK LIKE:
7817      ;
7818      ;TOP---(+16)
7819      ; +2---(+18)
7820      ; +4---R5
7821      ; +6---R4
7822      ; +8---R3
7823      ;+10---R2
7824      ;+12---R1
7825      ;+14---R0
7826
7827
7828 046176      $SAVREG: MOV      R0, -(SP)      ;; PUSH R0 ON STACK
7829 046176 010046      MOV      R1, -(SP)      ;; PUSH R1 ON STACK
7830 046200 011146      MOV      R2, -(SP)      ;; PUSH R2 ON STACK
7831 046202 011246      MOV      R3, -(SP)      ;; PUSH R3 ON STACK
7832 046204 010346      MOV      R4, -(SP)      ;; PUSH R4 ON STACK
7833 046206 010446      MOV      R5, -(SP)      ;; PUSH R5 ON STACK
7834 046210 010546      MOV      22(SP), -(SP)      ;; SAVE PS OF MAIN FLOW
7835 046212 016646 000022 MOV      22(SP), -(SP)      ;; SAVE PC OF MAIN FLOW
7836 046216 016646 000022 MOV      22(SP), -(SP)      ;; SAVE PS OF CALL
7837 046222 016646 000022 MOV      22(SP), -(SP)      ;; SAVE PC OF CALL
7838 046226 016646 000022 RTI
7839 046232 000002
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      JSWR,SAVSWR      ;SAVE SWITCH REG
7862 046300 012737 046320 000024      MOV      $PWRUP,PWRVEC      ;SET UP VECTOR
7863 046306 012737 000340 000026      MOV      $PR7,PWAVEC+2
7864 046314 000000              HALT
7865 046316 000776              BR      -2          ;HANG UP
7866
7867      ; *****
7868      ; POWER UP ROUTINE
7869 046320 005037 046410          $PWRUP: CLR      $PWACT      ;LOOP LOOP TIMER
7870 046324 012737 000144 046412      MOV      #100,$PWACT+2
7871 046332 005237 046410      1$: INC      $PWACT      ;WAIT FOR TELETYPE
7872 046336 001375              BNE      1$
7873 046340 005337 046412      DEC      $PWACT+2
7874 046344 001372              BNE      1$
7875 046346 012737 046272 000024      MOV      $PWRDN,PWRVEC      ;SET UP THE POWER DOWN VECTOR
7876 046354 012737 000340 000026      MOV      $PR7,PWAVEC+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,JSWR      ;RESTORE SWITCH REG
7881 046404 000177 132476      JMP      $SLPADR      ;START TEST AGAIN
7882
7883 046410 000000 000000          $PWACT: .WORD      0,0      ;COUNTER FOR TELETYPE
7884 046414 005015 047520 042527 $POWER: .ASCIIZ  <15><12>/POWER/
7885 046422 000122
7886
7887      .SBTTL  EVEN
7888      .SBTTL  TRAP DECODER
7889
7890      ; *****
7891      ; THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
7892      ; AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
7893      ; OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
7894      ; GO TO THAT ROUTINE.
7895 046424 010046              $TRAP: MOV      RD,-(SP)      ;; SAVE RD
7896 046426 016600 000002      MOV      2(SP),RD      ;; GET TRAP ADDRESS
7897 046432 005740              TST      -(RD)      ;; BACKUP BY 2
7898 046434 111000              MOVB     (RD),RD      ;; GET RIGHT BYTE OF TRAP
7899 046436 006300              ASL      RD      ;; POSITION FOR INDEXING
7900 046440 016000 046460      MOV      $TRPAD(RD),RD      ;; INDEX TO TABLE

```

```

7901 046444 000200          RTS      RO          ;;GO TO ROUTINE
7902
7903
7904          ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
7905
7906 046446 011646          STRAP2: MOV      (SP),-(SP)      ;;MOVE THE PC DOWN
7907 046450 016666 000004 000002      MOV      4(SF),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          $TRAPAD: .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 NUMBFR (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 TY
7930 046506 046176          $SAVREG     ;;CALL=SAVREG     TRAP+13(104413) SAVE RO-R5 ROUTINE
7931 046510 046234          $RESREG     ;;CALL=RESREG     TRAP+14(104414) RESTORE RO-R5 ROUTINE
7932 046512 043224          $SCOPI      ;;CALL=SCOPI     TRAP+15(104415) INTERNAL LOOP ON ERROR

```

E12

CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 147  
DATA PRINTED BY ERROR ROUTINES

SEQ 0147

## SBTTL DATA PRINTED BY ERROR ROUTINES

7933						
7934						
7935	046514	001220	004272		DT000:	.WORD \$TESTN,TRAPPC
7936	046520	001220	001116	004160	DT001:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,E.MR2,T.MR2,E.MR3,T.MR3
7937	046526	004120	004206	004146		
7938	046534	004210	004150			
7939	046540	001220	001116	004160	DT002:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,DRVCOD,E.MR2,T.MR2,E.MR3,T.MR3
7940	046546	004120	004244	004206		
7941	046554	004146	004210	004150		
7942	046562	001220	001116	004160	DT006:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,HDCODE,E.MR2,T.MR2,E.MR3,T.MR3
7943	046570	004120	004250	004206		
7944	046576	004146	004210	004150		
7945	046604	001220	001116	004160	DT012:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,E.MR1,T.MR1,MSGCOD
7946	046612	004120	004204	004144		
7947	046620	004246				
7948	046622	004206	004146	004210		.WORD E.MR2,T.MR2,E.MR3,T.MR3
7949	046630	004150				
7950	046632	001220	001116	004160	DT017:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,CYLIN,E.MR2,T.MR2,E.MR3,T.MR3
7951	046640	004120	004252	004206		
7952	046646	004146	004210	004150		
7953	046654	001220	001116	004160	DT031:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,OFFVAL,E.MR2,T.MR2,E.MR3,T.MR3
7954	046662	004120	004254	004206		
7955	046670	004146	004210	004150		
7956	046676	001220	001116	004160	DT035:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,CYLIN,OFFVAL
7957	046704	004120	004252	004254		
7958	046712	004206	004146	004210		.WORD E.MR2,T.MR2,E.MR3,T.MR3
7959	046720	004150				
7960	046722	001220	001116	004230	DT050:	.WORD \$TESTN,\$ERRPC,U.MR2,U.MR3,SFTCNT,E.MR2,T.MR2,E.MR3,T.MR3
7961	046730	004232	004256	004206		
7962	046736	004146	004210	004150		
7963	046744	001220	001116	004210	DT052:	.WORD \$TESTN,\$ERRPC,E.MR3,T.MR3,E.MR2,T.MR2
7964	046752	004150	004206	004146		
7965	046760	001220	001116	004160	DT062:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,E.CS2,T.CS2,E.DS,T.DS,E.ER,T.ER
7966	046766	004120	004170	004130		
7967	046774	004172	004132	004174		
7968	047002	004134				
7969	047004	001220	001116	004160	DT065:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1
7970	047012	004120				
7971	047014	001220	001116	004206	DT067:	.WORD \$TESTN,\$ERRPC,E.MR2,T.MR2,E.MR3,T.MR3
7972	047022	004146	004210	004150		
7973	047030	001220	001116		DT100:	.WORD \$TESTN,\$ERRPC
7974	047034	001220	001116	004160	DT126:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1
7975	047042	004120				
7976	047044	001220	001116	004160	DT224:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,E.CS2,T.CS2,E.DS,T.DS
7977	047052	004120	004170	004130		
7978	047060	004172	004132			
7979	047064	004174	004134	004220		.WORD E.ER,T.ER,P.CS1,P.CS2,P.DS,P.ER
7980	047072	004222	004224	004226		
7981	047100	001220	001116	004266	DT230:	.WORD \$TESTN,\$ERRPC,DRVTP,CYLIN,HDCODE,E.CS1,T.CS1,E.CS2,T.CS2
7982	047106	004252	004250	004160		
7983	047114	004120	004170	004130		
7984	047122	004172	004132	004174		.WORD E.DS,T.DS,E.ER,T.ER
7985	047130	004134				
7986	047132	001220	001116	004160	DT256:	.WORD \$TESTN,\$ERRPC,E.CS1,T.CS1,E.ER,T.ER,ILLFUN
7987	047140	004120	004174	004134		
7988	047146	004270				

				.SBTTL DATA FORMATS	
7989				DF000:	.WORD 1
7990					.BYTE 2,0
7991	047150	000001		DF001:	.WORD 7 ;ERROR 1
7992	047152	002	000		.BYTE 0,0
7993	047154	000007			.WORD 04000A
7994	047156	000	000		.BYTE 0,0
7995	047160	050207			.WORD 04000B
7996	047162	000	000		.BYTE 2,0
7997	047164	050225			.WORD 04001A
7998	047166	002	000		.BYTE 0,0
7999	047170	050271			.WORD 04001B
8000	047172	000	000		.BYTE 2,0
8001	047174	050310			.WORD 04001C
8002	047176	002	000		.BYTE 0,0
8003	047200	050326			.WORD 04001D
8004	047202	000	000		.BYTE 4,0
8005	047204	050365		DF002:	.WORD 7 ;ERRORS 2-5
8006	047206	004	000		.BYTE 0,0
8007	047210	000007			.WORD 04000A
8008	047212	000	000		.BYTE 0,0
8009	047214	050207			.WORD 04000B
8010	047216	000	000		.BYTE 2,0
8011	047220	050225			.WORD 04002A
8012	047222	002	000		.BYTE 0,0
8013	047224	050424			.WORD 04002B
8014	047226	000	000		.BYTE 3,0
8015	047230	050452			.WORD 04001C
8016	047232	003	000		.BYTE 0,0
8017	047234	050326			.WORD 04001D
8018	047236	000	000		.BYTE 4,0
8019	047240	050365		DF006:	.WORD 7 ;ERRORS 6-11
8020	047242	004	000		.BYTE 0,0
8021	047244	000007			.WORD 04000A
8022	047246	000	000		.BYTE 0,0
8023	047250	050207			.WORD 04000B
8024	047252	000	000		.BYTE 2,0
8025	047254	050225			.WORD 04006A
8026	047256	002	000		.BYTE 0,0
8027	047260	050501			.WORD 04006B
8028	047262	000	000		.BYTE 3,0
8029	047264	050526			.WORD 04001C
8030	047266	003	000		.BYTE 0,0
8031	047270	050326			.WORD 04001D
8032	047272	000	000		.BYTE 4,0
8033	047274	050365		DF012:	.WORD 7 ;ERRORS 12-16
8034	047276	004	000		.BYTE 0,0
8035	047300	000007			.WORD 04000A
8036	047302	000	000		.BYTE 0,0
8037	047304	050207			.WORD 04000B
8038	047306	000	000		.BYTE 2,0
8039	047310	050225			.WORD 04012A
8040	047312	002	000		.BYTE 0,0
8041	047314	050552			.WORD 04012B
8042	047316	000	000		.BYTE 5,0
8043	047320	050617			
8044	047322	005	000		

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		.WORD	7	;ERRORS 17-30
8050	047336	000	000	.BYTE	0,0	
8051	047340	050207		.WORD	DH000A	
8052	047342	000	000	.BYTE	0,0	
8053	047344	050225		.WORD	DH000B	
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		.WORD	7	;ERRORS 31-34
8064	047372	000	000	.BYTE	0,0	
8065	047374	050207		.WORD	DH000A	
8066	047376	000	000	.BYTE	0,0	
8067	047400	050225		.WORD	DH000B	
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		.WORD	7	;ERRORS 35-41
8078	047426	000	000	.BYTE	0,0	
8079	047430	050207		.WORD	DH000A	
8080	047432	000	000	.BYTE	0,0	
8081	047434	050225		.WORD	DH000B	
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		.WORD	7	;ERRORS 50 & 51
8092	047462	000	000	.BYTE	0,0	
8093	047464	050207		.WORD	DH000A	
8094	047466	000	000	.BYTE	0,0	
8095	047470	050225		.WORD	DH000B	
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	5,0	:ERRORS 52-61
8106	047516	000	000	.BYTE	0,0	
8107	047520	050207	.	WORD	DH000A	
8108	047522	000	000	.BYTE	0,0	
8109	047524	050225	.	WORD	DH000B	
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	5,0	: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	DH000B	
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	5,0	: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	DH000B	
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	5,0	: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	DH000B	
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	3,0	: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	DH000B	
8150	047646	002	000	.BYTE	2,0	
8151	047650	000005	DF126:	.WORD	5,0	: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	DH000B	
8156	047662	002	000	.BYTE	2,0	

CZR6800 RK611 DSKLS CTRL PRT2  
CZR680 P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 151  
DATA FORMATS

SEQ 0151

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		.WORD	7,0	: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		.WORD	7,0	: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		.WORD	5,0	: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	



J12

CZP6BCD RK611 DSKLS CTRL PR\*2  
CZP6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 152  
ASCII MESSAGES

SEQ 0152

8199

.SBTTL ASCII MESSAGES

8200  
8201 050010 005015 045522 030466  
8202 050016 020061 052502 020123  
8203 050024 042101 051104 051505  
8204 050032 020123 020050 000  
8205 050037 040 020051 020075  
8206 050044 000  
8207 050045 122 033113 030461  
8208 050052 053040 041505 047524  
8209 050060 020122 042101 051104  
8210 050066 051505 020123 020050  
8211 050074 000  
8212 050075 122 033113 030461  
8213 050102 050040 044522 051117  
8214 050110 052111 020131 020050  
8215 050116 000  
8216 050117 040 000040  
8217 050122 005015 051120 043517  
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

OPR001: .ASCIIZ &lt;15&gt;&lt;12&gt;/RK611 BUS ADDRESS (

OPR002: .ASCIIZ / ) = /

OPR003: .ASCIIZ /RK611 VECTOR ADDRESS ( /

OPR004: .ASCIIZ /RK611 PRIORITY ( /

SPACE2: .ASCIIZ / /

ABORT: .ASCIIZ &lt;15&gt;&lt;12&gt;/PROGRAM ABORTED BECAUSE ERROR THRESHOLD EXCEEDED/&lt;15&gt;&lt;12&gt;

.SBTTL DATA HEADERS			
8226			
8227			
8228	050207	124	051505 020124
8229	050214	020040	042440 051122
8230	050222	051117	000
8231	050225	116	046525 020040
8232	050232	020040	050040 000103
8233	050240	042524	052123 020040
8234	050246	020040	051124 050101
8235	050254	005015	
8236	050256	052516	020115 020040
8237	050264	020040	041520 000
8238	050271	105	050130 041505
8239	050276	020124	040440 052103
8240	050304	040525	000114
8241	050310	045522	051503 020061
8242	050316	020040	045522 051503
8243	050324	000061	
8244	050326	054105	042520 052103
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
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
8257	050432	020040	041501 052524
8258	050440	046101	020040 051104
8259	050446	053111	000105
8260	050452	045522	051503 020061
8261	050460	020040	045522 051503
8262	050466	020061	020040 042523
8263	050474	042514	052103 000
8264	050501	105	050130 041505
8265	050506	020124	040440 052103
8266	050514	040525	020114 044040
8267	050522	040505	000104
8268	050526	045522	051503 020061
8269	050534	020040	045522 051503
8270	050542	020061	020040 042101
8271	050550	000104	
8272	050552	054105	042520 052103
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
8280	050624	020040	051040 041513
8281	050632	030523	020040 051040

[illegible]

M12

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 155  
DATA HEADERS

SEQ 0155

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

8392					
8393					
8394	01.766	047125	054105	042520	EM000: .ASCIZ /UNEXPECTED MEMORY PARITY ENABLE TRAP/
8395	051774	052103	042105	046440	
8396	052002	046505	051117	020131	
8397	052010	040520	044522	054524	
8398	052016	042440	040516	046102	
8399	052024	020105	051124	050101	
8400	052032	000			
8401	052033	101	052124	046505	EM100: .ASCIZ /ATTEMPTING A SELECT IN 24 SECTOR FORMAT IN MAINT MODE/
8402	052040	052120	047111	020107	
8403	052046	020101	042523	042514	
8404	052054	052103	044440	020116	
8405	052062	032062	051440	041505	
8406	052070	047524	020122	047506	
8407	052076	046522	052101	044440	
8408	052104	020116	040515	047111	
8409	052112	020124	047515	042504	
8410	052120	000			
8411	052121	101	052124	046505	EM101: .ASCIZ /ATTEMPTING A DRIVE CLEAR IN MAINT MODE/
8412	052126	052120	047111	020107	
8413	052134	020101	051104	053111	
8414	052142	020105	046103	040505	
8415	052150	020122	047111	046440	
8416	052156	044501	052116	046440	
8417	052164	042117	000105		
8418	052170	052101	042524	050115	EM102: .ASCIZ /ATTEMPTING A UNLOAD IN MAINT MODE/
8419	052176	044524	043516	040440	
8420	052204	052440	046116	040517	
8421	052212	020104	047111	046440	
8422	052220	044501	052116	046440	
8423	052226	042117	000105		
8424	052232	052101	042524	050115	EM103: .ASCIZ /ATTEMPTING A PACK ACKNOWLEDGE IN MAINT MODE/
8425	052240	044524	043516	040440	
8426	052246	050040	041501	020113	
8427	052254	041501	047113	053517	
8428	052262	042514	043504	020105	
8429	052270	047111	046440	044501	
8430	052276	052116	046440	042117	
8431	052304	000105			
8432	052306	052101	042524	050115	EM104: .ASCIZ /ATTEMPTING A RECALIBRATE IN MAINT MODE/
8433	052314	044524	043516	040440	
8434	052322	051040	041505	046101	
8435	052330	041111	040522	042524	
8436	052336	044440	020116	040515	
8437	052344	047111	020124	047515	
8438	052352	042504	000		
8439	052355	101	052124	046505	EM105: .ASCIZ /ATTEMPTING A START SPINDLE/
8440	052362	052120	047111	020107	
8441	052370	020101	052123	051101	
8442	052376	020124	050123	047111	
8443	052404	046104	000105		
8444	052410	052101	042524	050115	EM106: .ASCIZ /ATTEMPTING A SELECT USING ALL DRIVE SELECTION CONFIGS IN MAINT MODE/
8445	052416	044524	043516	040440	
8446	052424	051440	046105	041505	
8447	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	C-2610	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	02103	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	052772	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	05312J	046517	040515	042116	
8503	053126	053440	052111	020110	

CZR6800 RK611 DSKLS CTRL PRT2  
CZR6800.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 158  
ERROR MESSAGES

SEQ 0158

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	

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 159  
ERROR MESSAGES

SEQ 0159

8560	053622	042524	042440	042530	
8561	053630	052503	044524	047117	
8562	053638	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



E13

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 160  
ERROR MESSAGES

SEQ 0160

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
8624	054402	045503	043440	020117
8625	054410	046103	040505	000122
8626	054416	052101	042524	050115
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
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
8643	054556	044524	043516	041440
8644	054564	042510	045503	021040
8645	054572	047514	042101	051440
8646	054600	040524	052524	021123
8647	054606	041040	020101	047506
8648	054614	041522	047111	006507
8649	054622	012		
8650	054623	104	044522	042526
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
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
8666	054752	051440	040524	052524
8667	054760	020123	042522	027107
8668	054766	000		
8669	054767	101	052124	046505
8670	054774	052120	047111	020107
8671	055002	047524	043040	051117

.ASCIZ /TO CHECK GO CLEAR/

EM123: .ASCIZ /ATTEMPTING COMPLETE EXECUTION OF SEEK IN MAINT MODE

EM124: .ASCIZ /ATTEMPTING SELECT DRIVE IN MAINT MODE

EM125: .ASCII /ATTEMPTING CHECK "LOAD STATUS" BY FORCING/1512/

.ASCII /DRIVE AVAILIABLE, SPEED LOSS, VOLUME VALID,1512/

ASCII /OFFSET, DRIVE READY, AND WRITE LOCK/1512/

.ASCIZ /DRIVE STATUS REG./

EM126: .ASCIZ /ATTEMPTING TO FORCE DRIVE AVAILIABLE/

F13

CZR6BC0 RK611 DSKLS CTRL PR2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 161  
ERROR MESSAGES

SEQ 0161

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	052111	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	

H13

CZR6BC0 RK611 DSKLS CTRL PR12  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 163  
ERROR MESSAGES

SEQ 0163

8784	056160	051105	041440	042514	
8785	056166	051101	000		
8786	056171	124	051505	044524	EM140: .ASCIZ /TESTING 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	

02R6800 RK611 DSKLS CTRL PRT2  
02R6800 P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 164  
ERROR MESSAGES

SEQ 0164

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 STATUS 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	

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR63C.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 165  
ERROR MESSAGES

SEQ 0165

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	EM2009: .ASCIZ /OFFSET VALUE BITS IN MESSAGE B INCORRECT/
8911	057472	043117	051506	052105	
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	

K13

CZR6BC0 PV611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 166  
ERROR MESSAGES

SEQ 0166

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	047101	020104		
8968	060204	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	

L13

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 167  
ERROR MESSAGES

SEQ 0167

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



CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 169  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0168

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	
ADWD0 = 000000	1172		
ADWD1 = 000000	1172		
ADWD10 = 000000	1172		
ADWD11 = 000000	1172		
ADWD12 = 000000	1172		
ADWD13 = 000000	1172		
ADWD14 = 000000	1172		
ADWD15 = 000000	1172		
ADWD2 = 000000	1172		
ADWD3 = 000000	1172		
ADWD4 = 000000	1172		
ADWD5 = 000000	1172		
ADWD6 = 000000	1172		
ADWD7 = 000000	1172		
ADWD8 = 000000	1172		
ADWD9 = 000000	1172		
ADEVCT = 000000	1172	1178	
ADEVN = 000000	1172	1214	
AENV = 000000	1172	1183	
AENVN = 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#	
APTSP0 = 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	
BAI = 000020	995#		
BA16 = 000400	980#		



[illegible]

945#

623



E14

CZP6BCD RK611 DSKLS CTRL PRT2  
CZP6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 174  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0173

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	7001#	7002#	7014#	7015#	7052	7053#	7055#	7058#	
ERRVEC=	000004	920#	2386	2387#	2398#									
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#

F14

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 175  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0174

		4912*	4943*	4944	4960	4987*	4991	5038*	5046*	5053	5075*	5079	5126*	5130
		5152*	5156	5203*	5207	5229*	5233	528*	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*

```

ISP      = 000100
NED      = 010000
NEM      = 004000
NEWPAS   005254
NXF      = 000004
OFFSET   000015
OFFVAL   004254

OFST     = 000004

```

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					
2314#												
2300#												
1014#												
1076	2425	6972	6979	7918	7919	7920	7921	7922	7924	7926	7927	7928
7929	7930	7931	7932									
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
2420	7924#											
2335#	2671*	2680	2692	2719*	2719	6115*	6117	6199*	6201	6284*	6286	6368*
6370	6452*	6454	6536*	6538	7942	7981						
830#	7406	7447										
1018#												
1020#	6056	6140	6309	6393	6477	6561						
978#	4806	4823	4852									
1010#	6922											
2343#	6913*	6919	6924	6945*	6946	7986						
973#												
925#	2370*	2371*										
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										
831#	441	7447										
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												
1000#												
933#	7003											
7007	7020#											
932#	7007*	7008*										
1052#												
1045#												
1053#	2755	3728	3795	3862	3929	3996	4063	4123				



CZR6BC0 RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 177  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0176

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	101*													
PCD = 010000	1055*													
PGE = 002000	1001*													
PIP = 020000	1039*	6223												
PIRQ = 177772	837*													
PIRQVE= 000240	931*													
PRO = 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*													
PWAVEC= 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	6594	6642	6698	6723	
	6777	6805	6859	6884	6921	6936								
	965*	2995	2996	3657	3662	3982	3986	6037	6053					
RECAL = 000013	7352	7931*												
RESREG= 104414	1081	2351*												
RESTAT 004306	921*													
RESVEC= 000010	948*	3301*	3368*	3421*	3481*	3541*	3601*	3661*						
RKASOF= 000016	943*													
RKBA = 000004	941*	2498*	2500*	2504	2516	2570*	2573*	2579	2587*	2621*	2623*	2630	2639*	
RKCS1 = 000000	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*	5889*	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	5452	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	6493	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*

J14

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*					
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		
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		
2327*	2477*											
956*												
2326*	2475*	2476*	4798	4835*	4842*	4851*	4870					
942*												
994*												
7301	7930*											
2345*	7861*	7880										
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							
2603	2717	2786	2847	2910	2973	3139	3202	3270	3342	4417	4477	4556
4614	4720	4773	6659	6944	7932*							
7173*	7932											
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						
960*	2500	2505	2519	2565	2573	2623	2624	2672	2682	2736	2744	3364
3369	4188	4291	4221	4373	4449	4512	4587	4642	4649	4746	4806	4844
4891	5262	5278	6758	6843								
2338*	4180*	4216*	4217	4235*	4272*	4273	7960					
1011*	5824											
7335	7347	8216*										
986*	5203	6698	6778	6780								
1033*	5048	5050	5590	5745	5823	5901	6055	6308	6476			
2328*	2348*	2351*	2354*	2427								
964*	3044	3045	3597	3602	3915	3919	5959	5975				
825*	2356	2368	7028	7363	7877							
2341*	6770											

SEQ 0179

[illegible]

L14

CZR6BCO RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 181  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0180

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#															
TYP0C =	104402	2431	2442	2459	7332	7615	7919#											
TYP0N =	104404	7921#																
TYP0S =	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				

M14

		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	5667*	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	583*	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

N14

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 183  
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0182

T.MR3 004150

T.SPAC 004156  
T.WC 004122  
UFE = 000400  
UNLOAD= 000007  
UNS = 040000  
UPE = 020000  
U.MR2 004230  
U.MR3 004232  
VV = 000100  
WAITIM 004262

WCE = 040000  
WLE = 004000  
WRDATA= 000023  
WRHEAD= 000027  
WRL = 004000  
WRTCHK= 000031  
WRTGAT= 040000  
\$APTHD 001000  
\$ASTAT= \*\*\*\*\* U  
\$ATYC 043274  
\$ATY1 043250  
\$ATY3 043256  
\$ATY4 043266  
\$AUTOB 001134  
\$BASE 001270

\$BDAOR 001122  
\$BDDAT 001126  
\$BELL 001204  
\$CDW1 001274  
\$CDW2 001276  
\$CHARC 044456  
\$CKSWR 045134  
\$CMTAG 001100

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				
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									
2295*												
2281*												
999*												
963*	2868	3537	3542	3848	3852	5805	5821					
1024*	6393	6477	6561									
1004*												
2322*	4182*	4237*	7960									
2323*	4184*	4239*	7960									
1035*	5048	5050										
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										
1005*												
1021*	5746											
969*												
971*												
1038*	5048	5050										
972*												
1057*												
1104	1110*											
7213	7228											
7184	7186*											
7182*												
7183*	7396											
7185*	7270											
1141*	2422*	7609	7758									
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												
1136*												
1138*												
1164*	7257	7290										
1215*												
1216*												
7413*	7423*	7430	7439*	7444*								
7601*	7926											
1124*	2363	2364	2372	2378	2379	2380						

[illegible]



1208#												
1112#												
7183#	7189	7224*	7228#									
7616	7756#											
1180#	7199*	7202										
1181#	7204*											
1174#	7197	7205*	7217	7221*								
7613	7754#											
1195#												
1203#												
1206#												
1209#												
7094	7104#											
1150#	7418	7447										
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		
7479#	7508#	7521#										
7474#	7478#	7483	7486#	7497#	7523#							
7048	7073	7082	7092	7101#								
1177#	2400#	6962#	6963#	6974	6996	7088	7105					
1114#												
7878	7884#											
7869#	7870#	7871#	7873#	7883#								
2376	7861#	7875										
7862	7869#											
1165#	7290	7447	7659	7735	7752	7808	7811					
7672#	7927											
7930												
7700#	7928											
7772#	7929											
7693#												
7844#	7931											
6995#												
7932												
7828#	7930											
2370	7045#											
2348#	2369	2370	2372	2374	2376	2378	2379	2380	2382	2410	2413	6960
7046	7251	7277	7285	7596	7758							
2348#												
7056	7095#											
1088#	1093											
800#	810	814	815									

		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								
\$SWOBT	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							
\$TK8	001146	1147#	7594	7605	7622	7676	7682							
\$TK5	001144	1146#	7594	7603	7619	7643#	7674	7680						
\$TMP0	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#	3347	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			

[illegible]

CLAPSW	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	4953	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#					</									

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22MACY11 30(1046) 02-DEC-77 09:31 PAGE 190  
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0188

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	2987	2988	3029	3037	3078	3087
	3144	3154	3209	3218	3278	3298	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												
SSCMRE	1117#														
SSCMTM	1117#	1154	1155	1156	1157	1158	1159	1160	1161						
SSESCA	932#														
SSNEWT	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										
SSSET	7910#	7919	7920	7921	7922	7924	7926	7927	7928	7929	7930	7931	7932		
SSSETM	2400#														
SSSKIP	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

H15

CZR6BCD RK611 DSKLS CTRL PRT2  
CZR6BC.P11 02-DEC-77 09:22

MACY11 30(1046) 02-DEC-77 09:31 PAGE 191  
CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0189

	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													
.HEAD	800#														
.SETUP	800#	2348													
.SWRHI	800#	810													
.SWRLO	800#	822#													
.SACT1	800#	1084													
.SAPT8	1169#														
.SAPTH	800#	1095													
.SAPTY	800#	7179													
.SCATC	800#	1070													
.SCMTA	800#	1117													
.SEOP	800#	6949													
.SERRO	800#	7236													
.SERRT	800#														
.SPOWE	800#														
.SROOC	800#	7758													
.SREAD	800#	7591													
.SSAVE	800#	7811													
.SSCOP	800#	7031													
.STRAP	800#	7887													
.STYPD	800#	7524													
.STYPE	800#	7368													
.STYPO	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