

```
      ,TITLE RB09 SYSTEM DIAGNOSTIC
/8-17-71, SMR
/PLEASE REPORT ANY PROBLEMS TO
/ STANLEY M, ROSE
/ LABORATORY OF COMPUTER SCIENCE
/ MASSACHUSETTS GENERAL HOSPITAL
/ 32 FRUIT STREET
/ BOSTON, MASSACHUSETTS 02114
/ (617) 726-3937
/
/USES EAE , BUT SAVES MQ, STEP COUNTER
/
/ERRORS:
/ 1: HARDWARE ERROR, LEGITIMATE BCD ADDRESS
/     STATUS REGISTER
/     DISK SIDE (0 OR 1)
/     TRACK ADDRESS (IN OCTAL)
/     SECTOR ADDRESS (IN OCTAL)
/
/ 2: HARDWARE ERROR, ILLEGAL BCD ADDRESS
/     STATUS REGISTER
/     ADDRESS LOADED
/
/ 3: MORE TYPE 1 OR 2 ERRORS THAN SPECIFIED BY [UODSW+1]
/     WILL NOT RE-TRY THIS BLOCK
/
/ 4: COMPARE ERROR
/     DISK SIDE
/     TRACK ADDRESS (OCTAL)
/     SECTOR ADDRESS (OCTAL)
/     CORE ADDRESS
/     OFFSET
/     EXPECTED
/     RECEIVED
/
/ 5: MORE ERRORS/BLOCK THAN SPECIFIED BY [UODSW+2]
/     WILL RE-READ AND TRY AGAIN
/
/ 6: RETRIED AS MANY TIMES AS SPECIFIED BY [UODSW+3]
/     WILL GO ON TO NEXT BLOCK
/
/ 7: ILLEGAL API ENTRY, NO FLAG SET
/     STATUS REGISTER
/     IORS
/     .EJECT
```

/ 10: AFTER ERRORS 7, 10 OR 11 THE PROGRAM SHOULD REINIALIZE AND RESTART
/ HOWEVER IT REENTERED ILLEGALLY
/ STATUS REGISTER
/ IORS

/ 11: INTERRUPT ENTRY AFTER CLEARING STATUS REGISTER AND STARTING
/ NO NEW IO OPERATION
/ STATUS REGISTER
/ IORS

707702 A EEM=707702
707764 A EBA=707764 /FOR PDP9 OR 15
641000 A CLAC=641000

703304 A DBK=703304
703344 A DBR=703344
705501 A SPI=705501
705504 A ISA=705504
705512 A RPL=705512
EJECT

```

,EBREL
00000 R 600007 A UODSW 600007 /IO,API CHANNEL 7 (BIT 2 SET TESTS MAINT.)
/((BIT 3 SET IS TEST WITH ONLY 64 WORD TESTS)
00001 R 000002 A HRDMAX 2 /MAX, NUMBER OF RETRIES ON HARDWARE ERRORS
00002 R 000005 A BLKCMP 5 /MAXIMUM NUM, OF REPORTED COMP, ERRORS / BLOCK
00003 R 000001 A CMPMAX 1 /MAXIMUM NUM, OF REREADS ON COMP, ERRORS
00004 R 000076 R ,DSA RBSERV /NORMAL ENTRY ADDRESS
00005 R 000032 R ,DSA RBINIT /INITIALIZE ADDRESS
00006 R 220260 A ,SIXBT "RB09 "
00007 R 714040 A
00010 R 000004 A SWITCH 4 /HOLD ON SWITCH 15
00011 R 000000 A ,BLOCK 7
00020 R 000000 A SYSCOM 0 /SYSTEM COMMUNICATION WORD
00021 R 000000 A ERWC 0 /TWO'S COMPLEMENT NUMBER OF WORDS TO PRINT
00022 R 000000 A ERCODE 0 /FIRST WORD OF MESSAGE
,REPT 7
00023 R 000000 A 0 /REST OF PRINT BUFFER
00024 R 000000 A #R
00025 R 000000 A #R
00026 R 000000 A #R
00027 R 000000 A #R
00030 R 000000 A #R
00031 R 000000 A #R
,EJECT

```

```

00032 R 740040 A RBINIT XX /INITALIZE AND START UP
00033 R 707764 A EBA
00034 R 707702 A EEM /9-15 COMPATIBLE ALTHOUGH NO 15'S WITH RB09!
00035 R 140021 R DZM ERWC
00036 R 140022 R DZM ERCODE
00037 R 201754 R LAC (BUFF
00040 R 040752 R DAC BUFFER /MOVING BUFFER INITIAL LOCATION
00041 R 777700 A LAW -100
00042 R 040570 R DAC DSKWC
00043 R 200000 R LAC UODSW /CHECK IF MAINTENANCE SECTORS
00044 R 501755 R AND (100000 /SHOULD BE USED (BIT 2)
00045 R 741200 A SNA
00046 R 600057 R JMP RBINT1
00047 R 201756 R LAC (120 /YES
00050 R 040757 R DAC PRIME
00051 R 201757 R LAC (200
00052 R 040720 R DAC MAITSC
00053 R 777766 A LAW -12
00054 R 040750 R DAC RUNNMB
00055 R 140751 R DZM START /MUST START ON 0 FOR MAINTENANCE
00056 R 600064 R JMP RBINT2

/
00057 R 140720 R RBINT1 DZM MAITSC
00060 R 201760 R LAC (127 /INCREMENT TRACK BY 1,SECTOR BY 7
00061 R 040737 R DAC PRIME
00062 R 777775 A LAW -3
00063 R 040750 R DAC RUNNMB /ONLY USED WHEN 'STAY IN 64 WORD/TRANSFER' MODE

/
00064 R 777773 A RBINT2 LAW -5
00065 R 040020 R DAC SYSCOM /NEXT ENTRY THROUGH RBSERV
00066 R 201761 R LAC (WRITE
00067 R 040174 R DAC EXIT
00070 R 040745 R DAC ARTFLG /NEXT ENTRY ARTIFICIAL
00071 R 200751 R LAC START
00072 R 040740 R DAC BLOCK /INITIAL BLOCK
00073 R 703344 A DBR
00074 R 400075 R XCT ,+1 /NEC. ON 9
00075 R 620032 R JMP# RBINIT
,EJECT

```

00076	R	740040	A	RBSERV	XX	/NORMAL ENTRY POINTR
00077	R	707764	A		EBA	
00100	R	707702	A		EEM	/FOR 9-15
00101	R	040730	R		DAC ACSAVE	
00102	R	641002	A		LACQ	
00103	R	040733	R		DAC MQSAVE	
00104	R	641001	A		LACS	
00105	R	040732	R		DAC SCSAVE	
00106	R	201762	R		LAC (400000)	
00107	R	705501	A		SPI	
00110	R	600122	R		JMP APIOFF	/NON-API ENTRY
00111	R	400722	R		XCT DSSF	
00112	R	600137	R		JMP ERROR7	/ILLEGAL API
00113	R	400724	R	RBSER1	XCT DSRS	
00114	R	040743	R		DAC STAT	
00115	R	400725	R		XCT DSCS	
00116	R	200745	R		LAC ARTFLG	/WAS ARTIFICIAL ENTRANCE EXPECTED?
00117	R	740200	A		SZA	
00120	R	600163	R		JMP ILLINT	/YES, BUT FLAG IS SET
00121	R	620174	R		JMP* EXIT	/CONTINUE FROM LAST EXIT
00122	R	400722	R	APIOFF	XCT DSSF	/FLAG UP?
00123	R	600127	R		JMP APIOF1	/NO
00124	R	750000	A		CLA	
00125	R	040730	R		DAC ACSAVE	
00126	R	600113	R		JMP RBSER1	
00127	R	200745	R	APIOF1	LAC ARTFLG	/ARTIFICIAL ENTRY EXPECTED?
00130	R	755201	A		SNA!CLCICLL	
00131	R	600134	R		JMP NOFLAG	/NO
00132	R	200020	R		LAC SYSCOM	/READY FOR ARTIFICIAL ENTRY?
00133	R	750201	A		SZA!CLC	/SYSCOM=0 MEANS READY
00134	R	620076	R	NOFLAG	JMP* RBSERV	/NOT THIS DEVICE
00135	R	140745	R		DZM ARTFLG	/ARTIFICIAL ENTRANCE ACHIEVED
00136	R	620174	R		JMP* EXIT	/CONTINUE ON
					EJECT	

```
00137 R 100225 R ERROR7 JMS PUTCHK
00140 R 201763 R LAC (7)
00141 R 040022 R DAC ERCODE
/
00142 R 400724 R ER7,1 XCT DSRS
00143 R 040023 R ER7,2 DAC ERCODE+1
00144 R 777775 A LAW -3
00145 R 040021 R DAC ERWC
00146 R 400724 R XCT DSRS
00147 R 040023 R DAC ERCODE+1
00150 R 700314 A IORS
00151 R 040024 R DAC ERCODE+2
00152 R 777775 A LAW -3
00153 R 040020 R DAC SYSCOM
00154 R 040745 R DAC ARTFLG
00155 R 400725 R XCT DSCS
00156 R 100174 R JMS EXIT
00157 R 100225 R JMS PUTCHK
00160 R 201764 R LAC (10)
00161 R 040022 R DAC ERCODE
00162 R 600142 R JMP ER7,1
/
00163 R 100225 R ILLINT JMS PUTCHK
00164 R 201765 R LAC (11)
00165 R 040022 R DAC ERCODE
00166 R 200743 R LAC STAT
00167 R 600143 R JMP ER7,2
,EJECT
```

/RESTART FROM SCRATCH

```

00170 R 740040 A MONEXT XX /NORMAL INTERRUPT EXIT
00171 R 140745 R DZM ARTFLG
00172 R 100174 R JMS EXIT
00173 R 620170 R JMP* MONEXT

/
00174 R 740040 A EXIT XX /ALL EXITS THROUGH HERE
00175 R 200732 R LAC SCSAVE /RESTORE STEP COUNTER
00176 R 241766 R XOR (77 /CMA
00177 R 341767 R TAD (640402
00200 R 501770 R AND (640477 /CREATE SHIFT
00201 R 040202 R DAC ,+1
00202 R 740040 A XX /EXECUTE CREATED SHIFT
00203 R 200733 R LAC MQSAVE
00204 R 652000 A LMQ
00205 R 200730 R LAC ACSAVE
00206 R 703344 A DBR
00207 R 400210 R XCT ,+1
00210 R 620076 R JMP* RBSERV /RETURN TO MONITOR

/
00211 R 740040 A ARTXIT XX /LEAVE EITHOUT STARTING ANY IO
00212 R 200020 R LAC SYSCOM /NEXT ENTRANCE WITHOUT FLAG SET
00213 R 740200 A SZA /ANY ERRORS IN TABLE?
00214 R 600220 R JMP ,+4 /YES, CHANGE TO -2
00215 R 140021 R DZM ERWC
00216 R 777773 A LAW -5
00217 R 741000 A SKP
00220 R 777776 A LAW -2
00221 R 040020 R DAC SYSCOM
00222 R 040745 R DAC ARTFLG
00223 R 100174 R JMS EXIT
00224 R 620211 R JMP* ARTXIT
.EJECT

```

```

00225 R 740040 A PUTCHK XX /CHECK IF ANYTHING IN OUTPUT TABLE
00226 R 200020 R LAC SYSCOM
00227 R 741200 A SNA /OK?
00230 R 600232 R JMP PUTCK1 /YES, TABLE EMPTY
00231 R 100211 R JMS ARTXIT

```

```

/
00232 R 777777 A PUTCK1 LAW -1
00233 R 040020 R DAC SYSCOM
00234 R 620225 R JMP* PUTCHK

```

```

/
/OUTPUT AN RB09 BCD DISK ADDRESS INTO ERRTABLE
/ STARTING AT [BBTMP2]
/SIDE 0 OR 1
/OCTAL OF TRACK (0-99 DEC.)
/OCTAL OF SECTOR (0-70 DEC.)
/

```

```

00235 R 740040 A BCDOUT XX
00236 R 200665 R LAC BINBCD
00237 R 673602 A LMO!CLAC!LLS+2
00240 R 060716 R DAC* BBTMP2
00241 R 440716 R ISZ BBTMP2
00242 R 641604 A CLAC!LLS+4
00243 R 744010 A RCL /MULT, BY 2
00244 R 040715 R DAC BBTMP1
00245 R 742010 A RTL /MULT, BY 4 (*2)
00246 R 340715 R TAD BBTMP1
00247 R 040715 R DAC BBTMP1
00250 R 641604 A CLAC!LLS+4
00251 R 340715 R TAD BBTMP1
00252 R 060716 R DAC* BBTMP2
00253 R 440716 R ISZ BBTMP2
00254 R 641604 A CLAC!LLS+4
00255 R 744010 A RCL /S1*2
00256 R 040715 R DAC BBTMP1
00257 R 742010 A RTL /S1*8
00260 R 340715 R TAD BBTMP1 /S1*10
00261 R 040715 R DAC BBTMP1
00262 R 641604 A CLAC!LLS+4
00263 R 340715 R TAD BBTMP1
00264 R 060716 R DAC* BBTMP2
00265 R 620235 R JMP* BCDOUT
      .EJECT

```



```

/ RB09 HANDLER
00266 R 200752 R WRITE LAC BUFFER
00267 R 040746 R DAC POINTR
00270 R 200570 R LAC DSKWC
00271 R 744022 A STL:RAR
00272 R 744022 A STL:RAR
00273 R 040744 R DAC LOOP /CUT LOOP BY 4 FOR INCREASED SPEED
00274 R 200740 R LAC BLOCK
00275 R 240736 R XOR HASH
00276 R 060746 R WRITE1 DAC* POINTR
00277 R 440746 R ISZ POINTR
00300 R 341771 R TAD (1
00301 R 060746 R DAC* POINTR
00302 R 440746 R ISZ POINTR
00303 R 341771 R TAD (1
00304 R 060746 R DAC* POINTR
00305 R 440746 R ISZ POINTR
00306 R 341771 R TAD (1
00307 R 060746 R DAC* POINTR
00310 R 440746 R ISZ POINTR
00311 R 341771 R TAD (1
00312 R 440744 R ISZ LOOP
00313 R 600276 R JMP WRITE1
00314 R 200740 R LAC BLOCK
00315 R 100665 R JMS BINBCD
00316 R 040665 R DAC BINBCD
00317 R 201772 R LAC (7000 /WRITE, INT, EN
00320 R 100553 R JMS DISK
00321 R 100537 R JMS MOVBUF /MOVE THE BUFFER
00322 R 200740 R LAC BLOCK
00323 R 340737 R TAD PRIME
,DEC
00324 R 341773 R TAD (-16000 /NO GREATER THAN 15999
00325 R 741100 A SPA
00326 R 341774 R TAD (16000
,OCT
00327 R 040740 R DAC BLOCK
00330 R 540751 R SAD START
00331 R 741000 A SKP /DONE WITH WRITING DISK, NOW READ
00332 R 600266 R JMP WRITE
,EJECT

```

```

/ RB09 READ ROUTINE
00333 R 200003 R READ LAC CMPMAX
00334 R 740001 A CMA /NUMBER OF REREADS ON COMPARE ERRORS
00335 R 040742 R DAC RDFTRY
00336 R 200740 R READ2 LAC BLOCK
00337 R 100665 R JMS BINBCD
00340 R 040665 R DAC BINBCD
00341 R 201775 R LAC (6000 /READ, INT, EN
00342 R 100553 R JMS DISK
00343 R 200752 R LAC BUFFER
00344 R 040746 R DAC POINTR
00345 R 200570 R LAC DSKWC
00346 R 744022 A STL:RAR
00347 R 744022 A STL:RAR
00350 R 040744 R DAC LOOP
00351 R 200002 R LAC BLKCMP /NUMBER OF COMPARE ERRORS
00352 R 740001 A CMA /PER BLOCK WHICH ARE REPORTED
00353 R 341771 R TAD (1
00354 R 040731 R DAC REPLOP
00355 R 200740 R LAC BLOCK
00356 R 240736 R XOR HASH
00357 R 560746 R READ3 SAD* POINTR
00360 R 741000 A SKP
00361 R 100471 R JMS RDEROR
00362 R 440746 R ISZ POINTR
00363 R 341771 R TAD (1
00364 R 560746 R SAD* POINTR
00365 R 741000 A SKP
00366 R 100471 R JMS RDEROR
00367 R 440746 R ISZ POINTR
00370 R 341771 R TAD (1
00371 R 560746 R SAD* POINTR
00372 R 741000 A SKP
00373 R 100471 R JMS RDEROR
00374 R 440746 R ISZ POINTR
00375 R 341771 R TAD (1
00376 R 560746 R SAD* POINTR
00377 R 741000 A SKP
00400 R 100471 R JMS RDEROR
00401 R 440746 R ISZ POINTR
00402 R 341771 R TAD (1
00403 R 440744 R ISZ LOOP
00404 R 600357 R JMP READ3
00405 R 100537 R READ1 JMS MOVBUF /MOVE THE BUFFER
00406 R 200740 R LAC BLOCK
00407 R 340737 R TAD PRIME
,DEC
00410 R 341773 R TAD (-16000
00411 R 741100 A SPA
00412 R 341774 R TAD (16000
,OCT
00413 R 040740 R DAC BLOCK
00414 R 540751 R SAD START /DONE READING?

```

```

00415 R 741000 A SKP /YES
00416 R 600333 R JMP READ
00417 R 200736 R LAC HASH /CALCULATE NEW HASH FOR NEXT PASS
00420 R 301776 R ADD (677773
00421 R 040736 R DAC HASH
00422 R 200000 R LAC UODSW /MAINTENANCE MODE?
00423 R 501755 R AND (100000
00424 R 740200 A SZA
00425 R 600460 R JMP DONE1 /YES
00426 R 200751 R LAC START
00427 R 341771 R TAD (1 /INCREMENT FIRST BLOCK
,DEC
00430 R 341773 R TAD (-16000 /RAP AROUND?
00431 R 741100 A SPA
00432 R 341774 R TAD (16000 /NO
,OCT
00433 R 040751 R DAC START
00434 R 040740 R DAC BLOCK
00435 R 200000 R LAC UODSW /STAY IN 64 WORD/BLOCK MODE?
00436 R 501777 R AND (040000
00437 R 740200 A SZA
00440 R 600460 R JMP DONE1 /YES, 6 PASSES PER COMPLETE PASS
00441 R 200570 R LAC DSKWC /CALCULATE NEW WORD COUNT
00442 R 744010 A RCL /MULT, BY 2
00443 R 542000 R SAD (-2000 /RAP AROUND?
00444 R 600462 R JMP DONE
00445 R 040570 R DAC DSKWC
00446 R 542001 R SAD (-200 /CALCULATE NEW APPROPRIATE PRIME
00447 R 202002 R LAC (136 /TRACK BY 1, SECTOR BY 6
00450 R 542003 R SAD (-400
00451 R 202004 R LAC (134 /TRACK BY 1, SECTOR BY 12
00452 R 542005 R SAD (-1000
00453 R 202006 R LAC (150 /TRACK BY 1, SECTOR BY 24
00454 R 040737 R DAC PRIME
00455 R 201754 R LAC (BUFF
00456 R 040752 R DAC BUFFER /RESET BUFFER TO START OF LARGE BUFFER
00457 R 600266 R JMP WRITE

/
00460 R 440750 R DONE1 ISZ RUNNMB /FINISHED?
00461 R 600266 R JMP WRITE
00462 R 100225 R DONE JMS PUTCHK /YES
00463 R 140021 R DZM ERWC
00464 R 140022 R DZM ERCODE
00465 R 777774 A LAW -4
00466 R 040020 R DAC SYSCOM
00467 R 100170 R JMS MONEXT
00470 R 740040 A HLT /SHOULD NOT RETURN HERE
,EJECT

```

```

00471 R 740040 A RDEROR XX /COMPARE ERROR
00472 R 040747 R DAC RDCHAR
/
00473 R 100225 R RDFERR JMS PUTCHK
00474 R 202007 R LAC (4)
00475 R 040022 R DAC ERCODE
00476 R 777770 A LAW -10
00477 R 040021 R DAC ERWC
00500 R 202010 R LAC (ERCODE+1
00501 R 040716 R DAC BBTMP2 /POINTER FOR BCDOUT
00502 R 100235 R JMS BCDOUT
00503 R 200746 R LAC POINTR
00504 R 040026 R DAC ERCODE+4
00505 R 200752 R LAC BUFFER
00506 R 740001 A CMA
00507 R 341771 R TAD (1
00510 R 340746 R TAD POINTR
00511 R 040027 R DAC ERCODE+5
00512 R 220746 R LAC* POINTR
00513 R 040030 R DAC ERCODE+6
00514 R 200747 R LAC RDCHAR
00515 R 040031 R DAC ERCODE+7
/
00516 R 440731 R ISZ REPLOP /DONE REPORTING ERRORS IN THIS BLOCK?
00517 R 600535 R JMP RDREP
00520 R 100225 R JMS PUTCHK /YES
00521 R 777777 A LAW -1
00522 R 040021 R DAC ERWC
00523 R 202011 R LAC (5)
00524 R 040022 R DAC ERCODE
/
00525 R 440742 R ISZ RDFTRY /REREAD?
00526 R 600336 R JMP READ2 /YES
00527 R 100225 R RDFER1 JMS PUTCHK /NO
00530 R 777777 A LAW -1
00531 R 040021 R DAC ERWC
00532 R 202012 R LAC (6)
00533 R 040022 R DAC ERCODE
00534 R 600405 R JMP READ1
/
00535 R 200747 R RDREP LAC RDCHAR /CHECK MORE
00536 R 620471 R JMP* RDEROR
.EJECT

```

/ROUTINE TO MOVE BUFFER WITHIN LARGE AVAILABLE BUFFER

```

00537 R 740040 A   MOVBUF  XX
00540 R 200570 R   LAC DSKWC   /CALCULATE NEEDED BUFFER SIZE
00541 R 740001 A   CMA
00542 R 341771 R   TAD (1
00543 R 340752 R   TAD BUFFER  /IS THERE ROOM AVAILABLE TO MOVE?
00544 R 541753 R   SAD BUFEND
00545 R 600550 R   JMP MOVBF1  /NO, RESET IT TO START
00546 R 440752 R   ISZ BUFFER  /YES, INCREMENT IT
00547 R 620537 R   JMP* MOVBUF
00550 R 201754 R   MOVBF1  LAC (BUFF
00551 R 040752 R   DAC BUFFER  /RESET IT
00552 R 620537 R   JMP* MOVBUF

```

//
//
//

```

00553 R 740040 A   DISK    XX           /RB09 START UP ROUTINE
00554 R 040735 R   DAC FNSAVE
00555 R 200001 R   LAC HRDMAX  /NUMBER OF RETRIES ON HARDWARE ERRORS
00556 R 740001 A   CMA
00557 R 040741 R   DAC ERRCNT
00560 R 750004 A   DISKRP  LAS           /HOLD SWITCH SET
00561 R 500010 R   AND SWITCH
00562 R 741200 A   SNA
00563 R 600566 R   JMP ,+3     /NO
00564 R 100211 R   JMS ARTXIT /YES
00565 R 600560 R   JMP DISKRP
00566 R 200752 R   LAC BUFFER
00567 R 400726 R   XCT DSLM
00570 R 777700 A   DSKWC   LAW -100
00571 R 400723 R   XCT DSLW
00572 R 200665 R   LAC BINBCD
00573 R 400721 R   XCT DSLD
00574 R 200735 R   LAC FNSAVE
00575 R 400727 R   XCT DSLS
00576 R 100170 R   JMS MONEXT /GO AWAY UNTIL INTERRUPT
/RETURN HERE ON INTERRUPT
00577 R 200743 R   LAC STAT
00600 R 740100 A   SMA           /ERROR?
00601 R 620553 R   JMP* DISK   /NO
           ,EJECT

```

00602	R	100225	R	JMS PUTCHK	/SYSCOM =1 NOW?
00603	R	200665	R	LAC BINBCD	/NO, CHECK IF LEGAL BCD FORMAT ADDRESS
00604	R	653601	A	LMQ:CLAC:LLS+1	/SHOULD BE ZERO
00605	R	744200	A	SZA:CLL	
00606	R	600633	R	JMP DISK2	
00607	R	660605	A	LLSS+5	
00610	R	502013	R	AND (17	
00611	R	342014	R	TAD (-12	/9 DEC, OR LESS
00612	R	740100	A	SMA	
00613	R	600633	R	JMP DISK2	
00614	R	640604	A	LLS+4	
00615	R	502013	R	AND (17	
00616	R	342014	R	TAD (-12	
00617	R	744100	A	SMA:CLL	
00620	R	600633	R	JMP DISK2	
00621	R	640604	A	LLS+4	
00622	R	502013	R	AND (17	
00623	R	342015	R	TAD (-10	/7 OR LESS
00624	R	744100	A	SMA:CLL	
00625	R	600633	R	JMP DISK2	
00626	R	640604	A	LLS+4	
00627	R	502013	R	AND (17	
00630	R	342014	R	TAD (-12	
00631	R	745100	A	SPA:CLL	
00632	R	600644	R	JMP DISK3	/GOOD ADDRESS
00633	R	777775	A	LAW =3	
00634	R	040021	R	DAC ERWC	
00635	R	202016	R	LAC (2)	
00636	R	040022	R	DAC ERCODE	
00637	R	200743	R	LAC STAT	
00640	R	040023	R	DAC ERCODE+1	
00641	R	200665	R	LAC BINBCD	
00642	R	040024	R	DAC ERCODE+2	
00643	R	600655	R	JMP DISK4	
/					
00644	R	201771	R	LAC (1)	
00645	R	040022	R	DAC ERCODE	
00646	R	777773	A	LAW =5	
00647	R	040021	R	DAC ERWC	
00650	R	200743	R	LAC STAT	
00651	R	040023	R	DAC ERCODE+1	
00652	R	202017	R	LAC (ERCODE+2	
00653	R	040716	R	DAC BBTMP2	
00654	R	100235	R	JMS BCDOUT	
/					
00655	R	440741	R	ISZ ERRCNT	/RETRY?
00656	R	600560	R	JMP DISKRP	/YES
00657	R	100225	R	JMS PUTCHK	
00660	R	202020	R	LAC (3)	/RETRIED AS MANY TIMES
00661	R	040022	R	DAC ERCODE	/AS SPECIFIED BY HRDMAX
00662	R	777777	A	LAW =1	
00663	R	040021	R	DAC ERWC	
00664	R	620553	R	JMP* DISK	

/TITLE RB09 BINARY TO BCD CONVERTER

```

00665 R 740040 A /BINBCD XX
00666 R 744000 A CLL
00667 R 653323 A IDIV
00670 R 000012 A .12 12
00671 R 040715 R DAC BBTMP1
00672 R 641323 A CLAC!DIV
00673 R 000010 A .10 10
00674 R 040716 R DAC BBTMP2
00675 R 641323 A CLAC!DIV; 12
00676 R 000012 A
00677 R 040717 R DAC BBTMP3
00700 R 641002 A LACQ
00701 R 342014 R TAD (-12
00702 R 740100 A SMA
00703 R 342012 R TAD (6
00704 R 340670 R TAD ,12
00705 R 660704 A ALSS+4
00706 R 300717 R ADD BBTMP3
00707 R 660704 A ALSS+4
00710 R 300716 R ADD BBTMP2
00711 R 660704 A ALSS+4
00712 R 300715 R ADD BBTMP1
00713 R 300720 R ADD MAITSC
00714 R 620665 R JMP* BINBCD

```

/0 UNLESS RUNNING MAINT, TEST

```

00715 R 000000 A BBTMP1 0
00716 R 000000 A BBTMP2 0
00717 R 000000 A BBTMP3 0

```

```

00720 R 000000 A MAITSC 0
.EJECT

```

/200 IF RUNNING MAINTENANCE TEST

```

00721 R 707104 A   DSLD  707104   /LOAD DISK ADDRESS
00722 R 707121 A   DSSF  707121   /SKIP ON FLAG
00723 R 707124 A   DSLW  707124   /LOAD WORD COUNT
00724 R 707132 A   DSRS  707132   /READ STATUS
00725 R 707141 A   DSCS  707141   /CLEAR STATUS
00726 R 707142 A   DSLM  707142   /LOAD CORE ADDRESS
00727 R 707144 A   DSLS  707144   /LOAD STATUS
/
/TITLE RB09 IOT'S
00730 R 000000 A   ACSAVE 0   /CONTENTS OF AC IF API, 0 IF PI
00731 R 000000 A   REPLOP 0
00732 R 000000 A   SCSAVE 0   /STEP COUNTER
00733 R 000000 A   MQSAVE 0   /MQ
00734 R 000000 A   SVEXIT 0   /SAVE EXIT WHEN HOLDING ON SWITCH
00735 R 000000 A   FNSAVE 0   /FUNCTION, READ OR WRITE
00736 R 000000 A   HASH 0
00737 R 000127 A   PRIME 127  /INCREMENT TRACK BY 1, SEGMENT BY 7
00740 R 000000 A   BLOCK 0   /TRACK ADDRESS
00741 R 000000 A   ERRCNT 0   /NUMBER OF RB09 RETRIES
00742 R 000000 A   RDPTRY 0
00743 R 000000 A   STAT 0   /RB09 STATUS WORD
00744 R 000000 A   LOOP 0
00745 R 000000 A   ARTFLG 0   /0 IF EXPECTING FLAG ENTRY
/ /NON-ZERO IF EXPECTING ARTIFICIAL ENTRY
00746 R 000000 A   POINTR 0
00747 R 000000 A   RDCHAR 0   /EXPECTED CHARACTER ON COMPARE ERROR
00750 R 000000 A   RUNNMB 0   /PASS COUNTER
00751 R 000000 A   START 0   /FIRST BLOCK OF TEST
00752 R 000753 R   BUFFER BUFF   /FIRST WORD OF MOVING BUFFER
00753 R A   BUFF ,BLOCK 1000
01753 R 001753 R   BUFEND BUFEND
,EJECT

```


.L TORG

01754 R 000753 R #L
01755 R 100000 A #L
01756 R 000120 A #L
01757 R 000200 A #L
01760 R 000127 A #L
01761 R 000266 R #L
01762 R 400000 A #L
01763 R 000007 A #L
01764 R 000010 A #L
01765 R 000011 A #L
01766 R 000077 A #L
01767 R 640402 A #L
01770 R 640477 A #L
01771 R 000001 A #L
01772 R 007000 A #L
01773 R 740600 A #L
01774 R 037200 A #L
01775 R 006000 A #L
01776 R 677773 A #L
01777 R 040000 A #L
02000 R 776000 A #L
02001 R 777600 A #L
02002 R 000136 A #L
02003 R 777400 A #L
02004 R 000134 A #L
02005 R 777000 A #L
02006 R 000150 A #L
02007 R 000004 A #L
02010 R 000023 R #L
02011 R 000005 A #L
02012 R 000006 A #L
02013 R 000017 A #L
02014 R 777766 A #L
02015 R 777770 A #L
02016 R 000002 A #L
02017 R 000024 R #L
02020 R 000003 A #L
000000 R

SIZE=02023

.END UODSW
NO ERROR LINES

