```
000001                          TITLE CMCS1,'REV B'
000002                  *       CACHE MEMORY TEST
000003                  *       PART NO.
000004                  * CMCX1      60134336-002
000005                  * CMCS1      60134337-002
000006                  * CMCL1      60134338-002
000007                  *
000008                  * DESCRIPTION
000009                  * -----------
000010                  *       THIS T & V PROGRAM VERIFIES PROPER OPERATION OF THE LEVEL-6 CACHE MEMORY
000011                  *       SUBSYSTEM.  PROVIDES A FIRST LEVEL OF DIAGNOSIS WHEN FAILURES ARE DETECTED
000012                  *       AND MAKES FACILITIES AVAILABLE TO SUPPORT EXTENSIVE PROBLEM INVESTIGATIONS
000013                  *
000014                  *
000015                  *       NOTE:  FOR MAIN MEMORY SIZE GREATER THAN 64K WORDS, THE CMCL1(LAF)
000016                  *              PROGRAM MUST BE USED.  THE SYSTEM SHOULD BE SET TO LAF MODE.
000017                  *              OTHERWISE, CMCS1 WILL ONLY USE UP TO 64K LOCATIONS.
000018                  *
000019                  *
000020                  *       THE SUBSYSTEM OPTIONS SUPPORTED BY THIS PROGRAM ARE:
000021                  *
000022                  *       BCHE006A  CACHE MEMORY CONTROLLER
000023                  *       BCHE002A  CACHE ARRAY PAC (2K WORDS)
000024                  *       BCHE004A  CACHE ARRAY PAC (4K WORDS)
000025                  *
000026                  * REVISION HISTORY
000027                  * -------- -------
000028                  *       A     DEC  1977              ORIGINAL RELEASE
000029                  *       B     APR  1978
000030                  *
000031                  *       THIS DOCUMENT AND THE INFORMATION CONTAINED THEREIN IS CONFIDENTIAL AND
000032                  *  PROPRIETARY TO AND THE EXCLUSIVE PROPERTY OF HONEYWELL INFORMATION SYSTEMS
000033                  *  INC.   IT  IS MADE AVAILABLE ONLY  TO  HONEYWELL AUTHORIZED RECIPIENTS FOR
000034                  *  THEIR USE SOLELY  IN  THE MAINTENANCE AND OPERATION OF HONEYWELL PRODUCTS.
000035                  *  THIS DOCUMENT AND INFORMATION MUST BE MAINTAINED  IN STRICTEST CONFIDENCE;
000036                  *  IT MUST NOT BE REPRODUCED IN  WHOLE OR IN PART;   AND IT SHALL NOT BE DIS-
000037                  *  CLOSED TO ANY OTHER PARTY WITHOUT THE PRIOR WRITTEN CONSENT OF HONEYWELL.
000038                  ************************************************************************
```

```
000039                          / PROGRAM PREPARATION:
000040                          *  ------- -----------
000041                          *      THE ROOT SOURCE OF THIS PROGRAM, AFTER THE ADDITION OF THE APPROPRIATE
000042                          *      TITLE AND END STATEMENTS, WAS PROCESSED BY THE HOST RESIDENT ASSEMBLER
000043                          *      TO CREATE EITHER SHORT OR LONG ADDRESS FORM ( SAF OR LAF ) OBJECT TEXT
000044                          *      AND LISTING.   THE  OBJECT TEXT  WAS  FURTHER PROCESSED  BY  THE  HOST
000045                          *      RESIDENT LINKER USING THE APPROPRIATE CONSOLE ZV$LIB LIBRARY TO CREATE
000046                          *      A PUNCH SEGMENT CONTAINING AN EXECUTABLE MODULE.  THE ASSEMBLY LISTING
000047                          *      WAS AUGMENTED WITH CROSS REFERENCE DATA,  PLUS  THE LOAD MAP FROM  THE
000048                          *      LINKER TO CREATE A LIST SEGMENT.
000049                          *              ROOT            SAF             LAF
000050                          *              ----            ---             ---
000051                          *      NAME    CMCX1           CMCS1           CMCL1
000052                          *      DOCUMENT 60134336-002   60134337-002   60134338-002
000053                          *
000054                          *  DISTRIBUTION
000055                          *  ------------
000056                          *      THE ELEMENTARY ITEMS SUBMITTED TO THE T & V PROGRAM DISTRIBUTION CENTER
000057                          *      WERE THE EXECUTABLE LINKED IMAGES, ON DISKETTE, OF PPPS1 AND PPPL1, AND
000058                          *      MAGNETIC TAPE IMAGES OF THE AUGMENTED LISTINGS.
000059                          *
000060                          *      REPRODUCTIONS OF THE EXECUTABLE LINKED IMAGES MAY BE  AS DUPLICATE CARD
000061                          *      DECKS OR  AS A MEMBER OF A MULTIPLE MEMBER FILE.   IN THE MOST FREQUENT
000062                          *      CASE, IT WILL BE FOUND  AS  MEMBER "SW" (SAF) OR "LW" (LAF) WITHIN FILE
000063                          *      "PROGFILE" OF A DISKETTE VOLUME ENTITLED "DIAGS".
000064                          *
000065                          *      DISTRIBUTION OF THE LISTINGS,  WHICH SHOULD BE AVAILABLE IF ANY COMPLEX
000066                          *      MAINTENANCE OR REPAIR IS TO BE PERFORMED, IS NORMALLY AS A PRINTED COPY.
000067                          *
000068                          *  ROUTINE DEMONSTRATION
000069                          *  ------- -------------
000070                          *          A MINIMUM SATISFACTORY TEST FOR NORMAL OPERATION MAY BE OBTAINED  BY
000071                          *      RUNNING A SINGLE PASS OF THE T&V PROGRAM WITH ALL THE AVAILABLE MAIN
000072                          *      MEMORY LOCATIONS.
000073                          *
000074                          *  STORAGE
000075                          *  -------
000076                          *      THIS PROGRAM REQUIRES AT LEAST  8 K  WORDS OF MAIN MEMORY AND  WILL  USE
000077                          *      ALL AVAILABLE MEMORY UP THROUGH MEMORY SIZE.
000078                          *
000079                          *  OPERATION
000080                          *  ---------
000081                          *      LOAD AND START (OR RESTART) THE PROGRAM. THE PROGRAM IDENTIFICATION WILL
000082                          *      BE DISPLAYED ON THE CONSOLE.  THE INITIAL START WILL ALSO DISPLAY:
000083                          *
000084                          *        THE ZV$LIB REVISION NUMBER
000085                          *        THE ADDRESS FORM (SAF OR LAF)
000086                          *        I/O  EQUIPMENT DETECTED IN THE SYSTEM
000087                          *        MEMORY SIZE
000088                          *
000089                          *      THIS DISPLAY MUST BE VERIFIED BY THE OPERATOR.   THIS DISPLAY IS OMITTED
000090                          *      ON RESTARTS.
000091                          *
000092                          *      THE CONSOLE SEARCH RULES ARE:  FIND THE CONSOLE WITH THE LOWEST  CHANNEL
000093                          *      NUMBER CONNECTED THRU AN  MDC  CONTROLLER.  IF THERE IS NO CONSOLE ON AN
000094                          *      MDC, THEN SEARCH FOR A TERMINAL WITH THE HIGHEST CHANNEL NUMBER ASSIGNED
000095                          *      TO AN ACLA ADAPTER ON AN MLC CONTROLLER.  IF NO ASYNC ADAPTER  IS FOUND,
000096                          *      THEN GO TO THE FULL CONTROL PANEL.
000097                          *
000098                          *      THERE ARE THREE CONSOLE CHANNEL OPTIONS DETERMINED BY THE VALUE  OF  LO-
000099                          *      CATION "ZV$TTY".
000100                          *
000101                          *      IF ZV$TTY EQUALS (0000), SEARCH FOR A CONSOLE.
000102                          *      IF ZV$TTY EQUALS (FFFF), ASSUME THERE IS NO CONSOLE.
000103                          *      IF ZV$TTY EQUALS NEITHER (0000), NOR (FFFF), THEN IT IS THE CONSOLE CHAN-
000104                          *      NEL NUMBER.    NOTE:  DEFAULT IS TO SEARCH FOR A CONSOLE.
000105                          *
000106                          *      ALL CONSOLE I/O IS EVEN PARITY.  IF CONSOLE IS ON MLC,  IT MUST BE ASYNC
000107                          *      AND THE BAUD RATE SET AT 1200 TO MATCH THE PROGRAM SUPPLIED RATE.  IF IT
000108                          *      IS NECESSARY TO CHANGE THE PROGRAM BAUD RATE,   THEN THE  NEW  BAUD RATE
000109                          *      CODE SHOULD BE PUT INTO LOCATION "ZV$BUD" IN HEX. THE TERMINAL BAUD RATE
000110                          *      MUST BE  SET  TO MATCH THIS NEW BAUD RATE.  THE CORRECT HEX VALUE MAY BE
000111                          *      OBTAINED FROM THE FOLLOWING TABLE.
000112                          *
000113                          *-----------------------------------------*
000114                          *                                         *
000115                          *        BAUD  RATE  TABLE                *
000116                          *                                         *
000117                          *-----------------------------------------*
000118                          *  ACLA I.D.          (2118)(2110)         (2108)
000119                          *  BAUD-RATE
000120                          *     50                 0                 1
000121                          *     75                 1                 2
000122                          *    110                 2                 3
000123                          *    134                 3                 4
000124                          *    150                 4                 5
000125                          *    200                 5                ---
000126                          *    300                 6                 6
000127                          *    600                 7                 7
000128                          *    900                ---                8
000129                          *   1050                 8                ---
000130                          *   1200                 9                 9
000131                          *   1800                10 (A)             10 (A)
000132                          *   2000                11 (B)            ---
000133                          *   2400                12 (C)             11 (B)
000134                          *   3600                ---                12 (C)
000135                          *   4800                13 (D)             13 (D)
000136                          *   7200                ---                14 (E)
000137                          *   9600                14 (E)             15 (F)
000138                          *  19200                15 (F)            ---
000139                          *
000140                          *      TO MAKE ANY OF THE ABOVE CHANGES,  LOAD AND HALT THE PROGRAM BEFORE EX-
000141                          *      ECUTION.  INSERT CHANGE THEN EXECUTE.  MEMORY LOCATIONS OF "ZV$TTY" AND
000142                          *      "ZV$BUD" MAY BE FOUND IN MAP AT END OF LISTING.
000143                          *      CONSULT LEVEL-6 T&V MANUAL "AW94" FOR DETAILS ON HOW TO LOAD THE TESTS.
000144                          *
000145                          *      THE FOLLOWING IS A TYPICAL RESULT OF LOADING  AND  STARTING TO  RUN THE
000146                          *      PROGRAM.
000147                          *
000148                          *      CMCS1   CACHE MEMORY TEST
000149                          *      APR 1978    REV. B
000150                          *      ZV$LIB REV. 6.0
000151                          *      ZV$AF= 1  <2>
```

```
000152                  *      WDT
000153                  *      CHAN  DEVC  ID
000154                  *      0400  DSKT  2010
000155                  *      0480  DSKT  2010
000156                  *      0580  CDR   2008
000157                  *      1200  DISC  2330
000158                  *      1280  DISC  2330
000159                  *      1300  LPT   2000
000160                  *      1380  CONS  2019
000161                  *      MEMORY LOW  00002B2D
000162                  *      MEMORY HIGH 00003FFF  16K
000163                  *
000164                  *
000165                  *
000166                  * USE
000167                  * ---
000168                  *      THE PROGRAM WILL FIRST CHECK THE SIZE OF THE MAIN MEMORY.  IF THE MAIN
000169                  *      MEMORY SIZE IS GREATER THAN 64K WORDS AND A SHORT ADDRESS FORM (SAF)
000170                  *      PROGRAM IS USED, THE PROGRAM WILL PRINT:
000171                  *
000172                  *         MEMORY GREATER THAN 64K, CHANGE TO CMCL1 PROGRAM, OTHERWISE, WILL
000173                  *         ONLY USE UP TO 64K LOCATIONS.
000174                  *
000175                  *      THE OPERATOR SHOULD SET THE SYSTEM TO LAF MODE AND RELOAD WITH THE CMCL1
000176                  *      TEST PROGRAM.
000177                  *
000178                  *
000179                  *      THE PROGRAM THAN WILL CHECK WHETHER THE CACHE IS PRESENT OR NOT.  IF THE
000180                  *      CACHE IS PRESENT, THE PROGRAM WILL PRINT:
000181                  *
000182                  *         CACHE PRESENT
000183                  *
000184                  *      IF THE CACHE IS NOT CONFIGURED, THE PROGRAM WILL PRINT:
000185                  *
000186                  *         CACHE NOT PRESENT
000187                  *
000188                  *      THE OPERATOR SHOULD VERIFY THE SYSTEM IF THIS MESSAGE IS PRINTED.
000189                  *
000190                  *
000191                  *      THE PROGRAM WILL THAN CHECK THE SIZE OF THE CACHE ARRAY.  THE CACHE SIZE
000192                  *      CAN BE EITHER 2K (2048) OR 4K (4096) WORDS.  IF THE CACHE SIZE IS 2K,
000193                  *      THE PROGRAM WILL PRINT:
000194                  *
000195                  *         CACHE SIZE IS 2K
000196                  *
000197                  *      IF THE CACHE SIZE IS 4K, THE PROGRAM WILL PRINT;
000198                  *
000199                  *         CACHE SIZE IS 4K
000200                  *
000201                  *
000202                  *
000203                  * ERROR REPORTING
000204                  * ----- ---------
000205                  *      DETECTED ERRORS ARE REPORTED AS FOLLOWS:
000206                  *
000207                  *          ERR CODE AT AAAA
000208                  *          REPLACE BBBB
000209                  *
000210                  *      WHERE:
000211                  *      CODE = ERROR CODE WITH UNIQUE FIRST FOUR CHARACTERS
000212                  *      AAAA = LOCATION OF TEST IN PROGRAM WHERE ERROR WAS.
000213                  *      BBBB = MOTHER BOARD OR DAUGHTER BOARD
000214                  *
000215                  *      NOTE:  THIS PROGRAM CAN BE OPERATED WITHOUT A CONSOLE.  ERROR-DATA IS
000216                  *      LIMITED TO "CODE" IN REGISTERS "R1" AND "R2", AND "AAAA" IN REGISTER "B2"
000217                  *      ALL ENTRIES MUST BE MADE VIA REGISTER "R1". THE PROCESS IS FURTHER EXPLAIN
000218                  *      ED IN MANUAL "AW94" ENTITL  ED "LEVEL-6 SYSTEM CHECKOUT AND T&V MANUAL".
000219                  *
000220                  *
000221                  * END OF PASS REPORT
000222                  * --- -- ---- ------
000223                  *      WHILE RUNNING, PASS COUNTS WILL BE REPORTED ON THE CONSOLE (IF PRESENT)
000224                  *    FOR EXAMPLE:
000225                  *
000226                  *         CMCS1  PS  1
000227                  *         CMCS1  PS  2
000228                  *         CMCS1  PS  3
000229                  *              .
000230                  *              .
000231                  *              .
000232                  *              .
000233                  *         CMCS1  PS  15
000234                  *         3 MIN DONE
000235                  *         CMCS1  PS  16
000236                  *              .
000237                  *              .
000238                  *              .
000239                  *              .
000240                  *
000241                  *
000242                  *      THE PASS COUNT MAY ALSO BE OBSERVED IN B6 (IN DECIMAL).
000243                  *
000244                  *      THE PROGRAM WILL PRINT "3 MIN DONE" AFTER 3 MINUTES TEST IS COMPLETED.
000245                  *      THE TEST WILL BE CONTINUED UNTILL MANUALLY STOPPED.
000246                  *
000247                  *
000248                  *      WHILE THE PROGRAM IS RUNNING, THE OPERATOR CAN STOP THE TEST ANY
000249                  *      TIME BY PUSHING THE STOP BUTTON ON THE OPERATOR'S PANNEL.  THE
000250                  *      TEST CAN BE RESTARTED AT E0 = 100 AND HIT THE EXECUTE BUTTON.
000251                  *
000252                  *
000253                  *
000254                  *
```

```
000255                          /  THIS PROGRAM RUNS AT LEVEL 4
000256                          *
000257                                CTRL  LINK ZV$TH
000258                                CTRL  LINK ZV$IH
000259                                XLOC  ZV$AF
000260                                XLOC  ZV$HK             HIGHEST MEMORY LOCATION
000261                                XLOC  ZV$HKU            HIGH ORDER BITS OF HIGH MEMORY ADDR
000262                                XLOC  ZV$LK             LOWEST MEMORY LOCATION
000263                                XLOC  ZV$TTY            =0 IF NO CONSOLE
000264                                XLOC  ZV$UIH            COMMON INTERRUPT HANDLER
000265                                XLOC  ZHCOMM            ABSOLUTE LOC 0
000266                                XLOC  ZHTSA             TRAP SAVE AREA - LOC 2
000267                                XLOC  ZHNTSA            NEXT AVAILABLE TSA PTR - LOC 10
000268                                XLOC  ZHRTCI            REAL TIME CLOCK INITIAL VALUE - LOC 14
000269                                XLOC  ZHRTCC            REAL TIME CLOCK CURRENT VALUE - LOC 15
000270                                XLOC  ZHRTCL            REAL TIME CLOCK RUPT LEVEL - LOC 16
000271                                XLOC  ZHTH15            TRAP HANDLER POINTER NO. 15 REFERENCE TO
000272                          *                             UNAVAILABLE RESOURCES - LOC 71(SAF), 62(LAF)
000273                                XLOC  ZHTH17            TRAP HANDLER POINTER NO. 17 - LOC 6F(SAF)
000274                          *                             LOC 5E(LAF)
000275                                XLOC  ZHISAZ            ISA0 POINTER - LOC 80
000276          0004          X     ZHISA4 EQU  ZHISAZ+4*$AF  ISA POINTER - LOC 84(SAF)
000277                          *                             LOC 88(LAF)
000278          0007          X     ZHTSA7 EQU  ZHTSA+4+2+$AF P-REG SAVED IN TSA - LOC 8(SAF), A(LAF)
000279                          *
000280          0000                ZERO   EQU  $
000281    0100                        ORG   ZERO+X'100'
000282                          *
000283                          *
000284                          *
000285    0100  0F00 0000             START  NOP   <ZERO            SAF, LAF CHECK
000286    0102  0F00 0000             NOP   <ZERO
000287    0104  7C02                  LDV   $R7,=2              DISABLE CACHE
000288    0105  E870 0300             LDR   $R6,=Z'0300'        CHANNEL NO.
000289    0107  0011                  DC    X'11'
000290    0108  8740 045F             CL    PASCNT+$AF-1        CLEAR PASS COUNT
000291    010A  0F84                  B     >START1
000292                          *
000293                          *
000294                          *
000295    010B  0F7F                  DONE  NOP   >$-1          3 MINUTE TESTING COMPLETE
000296                          *
000297    010C  0F81 0101             B     NXTPAS              CONTINUE TEST
000298                          *
000299                          *
000300    010E  0F86                  START1 B    >INSTRT       BECOMES "NOP" AFTER 1ST EXECUTION
000301                                START2 CALL ZV$RD         INITIALIZE
          010F  FBF0 0001
          0111  D380 0000       X
000302    0113  0F8C                  B     >START3             CONTINUE
000303                          *
000304                          *
000305                                INSTRT CALL ZV$RD,MSG1    INITIAL START, INITIALIZE AND PRINT
          0114  FBC0 0003
          0116  D380 0000       X
          0118  0F80
          0119  0491
000306    011A  9840 0023             LDR   $R1,NOP             PROGRAME NAME AND SYSTEM RESOURCES.
000307    011C  9F40 FFF1             STR   $R1,START1          INSERT "NOP" SO ONLY DO ONCE.
000308    011E  0FF1                  B     >START2
000309                          *
000310    011F  9BC0 0421             START3 LAB  $B1,SECTSA    PUT ADDRESS OF SECONDARY TRAP SAVE
000311    0121  9F80 0000       X           STB   $B1,<ZHTSA    AREA IN LC X'0002'.  SEE COMMENT
000312                          *                             AT "SECISA".
000313                          *
000314    0123  8740 0417             CL    ISA4                CLEAR TSA POINTER
000315    0125  8740 0415             CL    ISA4+$AF-1
000316                          *
000317                          * CHANGE TO LEVEL 4
000318                          *
000319    0127  BBC0 0414             LAB   $B3,ISA4+$AF        STORE POINTER TO INTERRUPT SAVE
000320    0129  BF80 0004       X     STB   $B3,<ZHISA4         AREA FOR LEVEL 4.
000321    012B  BBC0 0007             LAB   $B3,T4A1            ADDRESS FOR LEVEL 4 INTERRUPT.
000322    012D  BFC0 0411             STB   $B3,ISA4P
000323    012F  8E70 8004             LEV   =Z'8004'            SCHEDULE LEVEL 4,SUSPEND LEVEL (0),
000324                          *                             SCAN AND DISPACH.  FIRST SAVE CONTEXT
000325                          *                             OF CURRENT RUNNING LEVEL (0) AND
000326                          *                             RESTORE CONTEXT OF HIGHEST ACTIVE
000327                          *                             LEVEL (4).
000328    0131  0F80 0000       X     B     <ZV$UIH             ADDRESS OF THIS LOCATION GOES TO
000329                          *                             P-REG WORD OF ZV$ISA.
000330                          *
000331                          * COMES HERE ON LEVEL CHANGE FROM 0 TO 4
000332                          *
000333                          *
000334                          *
000335                          * INITIALIZE PASS COUNT
000336                          *
000337    0133  EC80 056E             T4A1  LDB   $B6,<NULL     **  B6 RESERVED FOR PASS COUNT DISPLAY  **
000338                          *
000339                          * INITIALIZE REAL TIME CLOCK FOR LATER TURN ON
000340                          *
000341    0135  9870 FFFF             LDR   $R1,=Z'FFFF'        MAX. VALUE
000342    0137  9F00 0000       X     STR   $R1,<ZHRTCC         RTC CURRENT
000343    0139  9F00 0000       X     STR   $R1,<ZHRTCI         RTC INITIAL
000344    013B  1C3F                  LDV   $R1,=63             LOWEST PRIORITY.  WON'T INTERRUPT
000345    013C  9F00 0000       X     STR   $R1,<ZHRTCL         RTC LEVEL
000346                          *
000347    013E  0F7F                  NOP   NOP   >$-1
000348                          *
000349                          *
000350    013F  9800 0000       X           LDR   $R1,<ZV$AF
000351    0141  1D02                  CMV   $R1,=2              LAF?
000352    0142  0911                  BE    >CLOCK              YES
000353    0143  8980 0000       X     CMZ   <ZV$HKU             NO.  MEMORY SIZE GREATER THAN 64K?
000354    0145  090E                  BE    >CLOCK              NO
000355                                CALL  ZV$TC,MSG2          YES.  MEMORY SIZE GREATER THAN 64K.
          0146  FBC0 0003
          0148  D380 0000       X
          014A  0F80
          014B  04A9
000356                                CALL  ZV$TC,MSG2B
          014C  FBC0 0003
```

```
                0014E  D380 0000      X
                0150   0F80
                0151   04D2
000357                                *                               PRINT MESSAGE:
000358  0152    0F7F                  NOP   >$-1                      CHANGE TO CALI(LAF) TEST PROGRAM
000359                                *                               AND SET THE SYSTEM TO LAF MODE
000360                                *                               OTHERWISE, THE PROGRAM WILL
000361                                *                               ONLY USE UP TO 64K LOCATIONS
000362                                *
000363
000364  0153  0004                    CLOCK RTCN                      TURN ON CLOCK
000365                                *
000366                                *
000367
000368                                *  TEST TO SEE IF CACHE IS PRESENT
000369                                **********************************
000370                                *
000371                                *
000372  0154  AB80 0171               TCP10 LAB   $B2,<TCP20          SET UP IV15
000373  0156  AF80 0000      X              STB   $B2,<ZHTH15
000374  0158  8740 040D                     CL    CAFLG               CLEAR CACHE FLAG
000375                                *
000376  015A  7C08                          LDV   $R7,=8              INITIALIZE CACHE
000377  015B  E870 0300                     LDR   $R6,=Z'0300'        CHANNEL NO.
000378  015D  0011                          DC    X'11'               WILL TRAP IF CACHE IS NOT PRESENT
000379                                *
000380  015E  7C02                          LDV   $R7,=2              DISABLE CACHE
000381  015F  0011                          DC    X'11'
000382                                *
000383  0160  89C0 0405                     CMZ   CAFLG               CACHE PRESENT?
000384  0162  0988                          BNE   >TCP15              NO
000385                                *
000386                                *
000387                                      CALL  ZV$TC,MSG2A         YES.  PRINT "CACHE PRESENT"
                0163  FBC0 0003
                0165  D380 0000      X
                0167  0F80
                0168  04E2
000388  0169  0F8B                          B     >DCS5
000389                                *
000390                                *
000391                                TCP15 CALL  ZV$TC,MSG3          PRINT "CACHE NOT PRESENT"
                016A  FBC0 0003
                016C  D380 0000      X
                016E  0F80
                016F  04E9
000392  0170  0000                          HLT
000393                                *
000394  0171  8AC0 03F4               TCP20 INC   CAFLG               CACHE IS NOT PRESENT.  SET FLAG
000395  0173  0003                          RTT                       RETURN
000396                                *
000397                                *
000398                                *
000399                                *  DETERMINE CACHE SIZE
000400                                ************************
000401                                *
000402                                *
000403                                *  CHECK NO HIT FAULT TEST MODE
000404                                *
000405  0174  AB80 0192               DCS5  LAB   $B2,<DCS10          SET UP IV15 FOR UNAVAILABLE RESOURCES
000406  0176  AF80 0000      X              STB   $B2,<ZHTH15
000407                                *
000408  0178  8740 03E1                     CL    TEMPA               CLEAR NO HIT FLAG
000409  017A  BCC0 001A                     LDB   $B3,DCS20           TEST ADDR
000410                                *
000411  017C  7C08                          LDV   $R7,=8              INITIALIZE CACHE
000412  017D  E870 0300                     LDR   $R6,=Z'0300'        CHANNEL NO.
000413  017F  0011                          DC    X'11'
000414                                *
000415  0180  F870 00C0                     LDR   $R7,=Z'00C0'        SET TEST MODE ON
000416  0182  0011                          DC    X'11'
000417  0183  C803                          LDR   $R4,$B3             READ.  WILL CAUSE NO HIT FAULT TRAP
000418                                *
000419  0184  89C0 03D5                     CMZ   TEMPA               TRAP?  = 0 NO TRAP, = 1 TRAP
000420  0186  0990                          BNE   >DCS30              YES
000421                                      CALL  ZV$ER,DC1,MSGDC1    NO.  ERROR "DC1"
                0187  FBC0 0003
                0189  D380 0000      X
                018B  0F80
                018C  0190
                018D  050F
000422  018E  D3C0 02B4                     LNJ   $B5,PR2             PRINT "REPLACE ARRAY PAC"
000423  0190  0F7F                  DC1   NOP   >$-1                  **  ERROR  **
000424                                *                               COMPARATOR LOGIC FAULT
000425  0191  0F85                          B     >DCS30
000426                                *
000427                                *
000428  0192  8AC0 03C7               DCS10 INC   TEMPA               TRAP.  SET NO HIT FAULT FLAG
000429  0194  0003                          RTT
000430  0195        1FFE      X       DCS20 DC    <ZHCOMM+X'1FFE'       TEST ADDR
000431                                *
000432                                *
000433                                *  NOW CHECK THE CACHE SIZE
000434                                *
000435                                *
000436  0196  AB80 01CD               DCS30 LAB   $B2,<DCS40          SET UP IV15
000437  0198  AF80 0000      X              STB   $B2,<ZHTH15
000438                                *
000439  019A  8740 03D1                     CL    CASIZE
000440  019C  A870 0FFF                     LDR   $R2,=Z'0FFF'        PRESTORE 4K AS LAST LOCATION ON CACHE
000441  019E  AF40 03CE                     STR   $R2,CASIZE+1        WILL MODIFY IF CACHE SIZE IS NOT 4K
000442  01A0  8740 03CA                     CL    SIZFLG              CLEAR CACHE SIZE FLAG
000443  01A2  7C08                          LDV   $R7,=8              INITIALIZE CACHE
000444  01A3  E870 0300                     LDR   $R6,=Z'0300'        CHANNEL NO.
000445  01A5  0011                          DC    X'11'
000446  01A6  F870 00C0                     LDR   $R7,=Z'00C0'        SET NO-HIT FAULT MODE ON
000447  01A8  0011                          DC    X'11'
000448  01A9  9CC0 0026                     LDB   $B1,DCS50           TEST ADDR TO CHECK WHETHER THE
000449                                *                               CACHE SIZE IS 4K
000450  01AB  C801                          LDR   $R4,$B1             READ.  WILL TRAP IF CACHE SIZE
000451                                *                               IS LESS THAN 4K (4095)
000452                                *
000453                                *
```

```
000454  01AC  89C0 03BE          CMZ   SIZFLG           TRAP?
000455  01AE  0924               BE    >DCS70           NO TRAP.  CACHE SIZE IS 4K
000456  01AF  8740 03BB          CL    SIZFLG           TRAP.  RESET SIZE FLAG
000457  01B1  9CC0 001F          LDB   $B1,DCS60        TEST ADDR TO CHECK WHETHER THE
000458                       *                         CACHE SIZE IS 2K
000459  01B3  C801               LDR   $R4,$B1          READ.  WILL TRAP IF CACHE SIZE
000460                       *                         IS LESS THAN 2K (2047)
000461  01B4  89C0 03B6          CMZ   SIZFLG           TRAP?
000462  01B6  0923               BE    >DCS80           NO TRAP.  CACHE SIZE IS 2K
000463                       *
000464  01B7  8740 03B3          CL    SIZFLG           TRAP.  CACHE SIZE IS LESS THAN 2K
000465  01B9  8740 03A3          CL    TEMP2            SET ADDR TO 0
000466  01BB  8740 03A2          CL    TEMP2+1
000467  01BD  9CC0 039B          LDB   $B1,TEMP1-$AF+2
000468  01BF  C801               LDR   $R4,$B1          READ LOCATION 0.  WILL TRAP  IF
000469                       *                         CACHE DAUGHTER BOARD IS NOT PRESENT
000470  01C0  89C0 03AA          CMZ   SIZFLG           TRAP?
000471  01C2  0921               BE    >CPP10           NO TRAP.  GO TO NEXT TEST
000472                       *
000473                       *
000474                           CALL  ZV$ER,DC2,MSGDC2 TRAP.  ERROR "DC2"
        01C3  FBC0 0003
        01C5  D380 0000      X
        01C7  0F80
        01C8  01CA
        01C9  0511
000475  01CA  0F7F           DC2   NOP   >$-1           **  ERROR  **
000476                       *                         CHECK WHETHER DAUGHTER BOARD IS
000477                       *                         PROPERLY CONNECTED.
000478  01CB  0F81 0017          B     CPP10
000479                       *
000480                       *
000481                       *
000482                       *
000483  01CD  8AC0 039D      DCS40 INC   SIZFLG           TRAPS.  SET FLAG FOR CACHE SIZE
000484                       *                         LESS THAN 4K
000485  01CF  0003               RTT
000486                       *
000487  01D0        0FFF     X  DCS50 DC    <ZHCOMM+X'0FFF'  TEST ADDR (4095) TO CHECK WHETHER
000488                       *                         CACHE SIZE IS 4K
000489                       *
000490  01D1        07FF     X  DCS60 DC    <ZHCOMM+X'07FF'  TEST ADDR (2047) TO CHECK WEITHER
000491                       *                         CACHE SIZE IS 2K
000492                       *
000493                       DCS70 CALL  ZV$TC,MSG3A      PRINT "CACHE SIZE IS 4K"
        01D2  FBC0 0003
        01D4  D380 0000      X
        01D6  0F80
        01D7  04F2
000494  01D8  0F8B               B     >CPP10           GO TO NEXT TEST
000495                       *
000496                       *
000497                       DCS80 CALL  ZV$TC,MSG3B      PRINT "CACHE SIZE IS 2K"
        01D9  FBC0 0003
        01DB  D380 0000      X
        01DD  0F80
        01DE  04FB
000498  01DF  A870 07FF           LDR   $R2,=Z'07FF'     LAST LOCATION ON CACHE
000499  01E1  AF40 038B           STR   $R2,CASIZE+1
000500                       *
000501                       *
000502                       *
000503                       *
000504                       *  CHECK WHETHER PROM IS PRESENT
000505                       ********************************
000506                       *
000507                       *
000508  01E3  AB80 020D      CPP10 LAB   $B2,<CPP60       SET UP IV15 FOR UNAVAILABLE RESOURCES
000509  01E5  AF80 0000      X     STB   $B2,<ZHTH15
000510  01E7  AB80 020D           LAB   $B2,<CPP60       SET UP IV17 FOR PARITY TRAP
000511  01E9  AFC0 0000      P     STB   $B2,ZHTH17
000512  01EB  ACC0 0000      P     LDB   $B2,ZV$HR        MEMORY HIGH
000513  01ED  9CC0 0380           LDB   $B1,NULL         CLEAR B1
000514  01EF  9C80 0000      X     LDB   $B1,<ZV$LR       MEMORY
000515                       *
000516  01F1  C801           CPP20 LDR   $R4,$B1          READ ORIGIN
000517  01F2  CF40 0367           STR   $R4,TEMPA        SAVEORIGIN
000518  01F4  8601                CPL   $B1              READ ORIGIN, CPL, WRITE
000519  01F5  C671                XOR   $R4,+$B1         XOR ORIGUN WITH CPL DATA
000520  01F6  4DFF                CMV   $R4,-1           ORIGIN COMPLEMENTED?
000521  01F7  0989                BNE   >CPP40           NO
000522  01F8  9DD2                CMB   $B1,=$B2         ADDR GREATER THAN MEMORY HIGH?
000523  01F9  0278                BL    >CPP20           NO.  CHECK NEXT LOCATION
000524                       *
000525                       *
000526  01FA  9C80 0000      X     LDB   $B1,<ZV$HR       PROM NOT PRESENT
000527  01FC  9FC0 0373           STB   $B1,MEMHI-$AF+2  MEMORY HIGH
000528  01FE  0F81 0010           B     RRT10
000529                       *
000530                       *
000531  0200  C861           CPP40 LDR   $R4,-$B1        DEC. ADDR, READ LOCATION THAT DON'T CPL
000532  0201  9854                LDR   $R1,=$R4
000533  0202  1048                SOR   $R1,8            SHIFT DATA TO CHECK FOR 0FXX
000534  0203  1D0F                CMV   $R1,=X'0F'       PROM 1ST WORD?
000535  0204  0903                BE    >CPP50           YES.  PROM PRESENT
000536  0205  C971                CMR   $R4,+$B1         NO.  DUMMY TO BUMP ADDR
000537  0206  0FEB                B     >CPP20           CHECK NEXT LOCATION
000538                       *
000539                       *
000540  0207  9FC0 036A      CPP50 STB   $B1,PLOW-$AF+2   STORE PROM 1ST LOCATION
000541  0209  C961                CMR   $R4,-$B1         DUMMY TO DEC. ADDR
000542  020A  9FC0 0365           STB   $B1,MEMHI-$AF+2  STORE RAM LAST LOCATION (MEMORY HIGH)
000543  020C  0F83                B     >RRT10
000544                       *
000545                       *
000546  020D  0003           CPP60 RTT                    RETURN
000547                       *
000548                       *
000549                       *
000550                       *
000551                       *
000552                       *
000553                       *  COMES HERE FOR START OF NEXT PASS
```

```
000554                                  ************************************
000555                                  *
000556                                  *
000557  020E  0F7F                      NXTPAS NOP   >$-1
000558                                  *
000559                                  *
000560                                  *
000561                                  *
000562                                  *  ROUND ROBIN TEST
000563                                  ********************
000564                                  *
000565                                  *
000566  020F  AB80 023C                 RRT10  LAB   $B2,<RRT40          SET UP IV 15 FOR UNAVAILABLE RESOURCES
000567  0211  AF80 0000       X                STB   $B2,<ZHTH15
000568                                  *
000569  0213  8740 0353                       CL    TRPFLG               CLEAR TRAP FLAG
000570  0215  ACC0 0357                       LDB   $B2,CASIZE-$AF+2     LAST LOCATION ON CACHE
000571  0217  8740 0340                       CL    TEMP1
000572  0219  8740 033F                       CL    TEMP1+1
000573  021B  9CC0 033D                       LDB   $B1,TEMP1-$AF+2      INITIALIZE ADDR TO 0
000574                                  *
000575  021D  E870 0300                       LDR   $R6,=Z'0300'         CHANNEL NO.
000576  021F  7C08                            LDV   $R7,=8               INITIALIZE CACHE
000577  0220  0011                            DC    X'11'
000578  0221  F870 00C0                       LDR   $R7,=Z'00C0'         SET TO NO HIT MODE
000579  0223  0011                            DC    X'11'
000580  0224  C871                 RRT20  LDR   $R4,+$B1              READ.  BUMP ADDR.  WILL TRAP IF
000581                                  *                                ROUND ROBIN FAULT IS DETECTED
000582                                  *
000583  0225  89C0 0341                       CMZ   TRPFLG               TRAP?
000584  0227  0901 FFFC                       BE    RRT20                NO TRAP.  CONTINUE TEST
000585  0229  8740 033D                       CL    TRPFLG               TRAP
000586  022B  9DD2                            CMB   $B1,=$B2             ADDR LESS THAN CACHE SIZE?
000587  022C  0383                            BLE   >RRT30               NO.  REPORT ERROR
000588                                  *
000589                                  *
000590                                  *
000591  022D  0F81 0011                       B     CLT10                GO TO NEXT TEST
000592                                  *
000593                                  *
000594                                  RRT30  CALL  ZV$ER,RR1,MSGRR1     ERROR "RR1"
       022F  FBC0 0003
       0231  D380 0000       X
       0233  0F80
       0234  0238
       0235  0523
000595  0236  D3C0 0202                       LNJ   $B5,PR1              PRINT "REPLACE MOTHER BOARD"
000596  0238  0F7F                      RR1    NOP   >$-1                 **  ERROR  **
000597                                  *                                ROUND ROBIN RAM ERROR
000598                                  *                                REPLACE MOTHER BOARD
000599  0239  7C08                            LDV   $R7,=8               INITIALIZE CACHE
000600  023A  0011                            DC    X'11'
000601  023B  0FE9                            B     >RRT20               CONTINUE TEST
000602                                  *
000603                                  *
000604  023C  8AC0 032A                 RRT40  INC   TRPFLG               SET TRAP FLAG
000605  023E  0003                            RTT                        RETURN
000606                                  *
000607                                  *
000608                                  *
000609                                  *
000610                                  *  COMPARATOR LOGIC TEST
000611                                  ************************
000612                                  *
000613                                  *
000614  023F  AB80 0275                 CLT10  LAB   $B2,<CLT50           SET IV15
000615  0241  AF80 0000       X                STB   $B2,<ZHTH15
000616  0243  9CC0 0329                       LDB   $B1,CASIZE-$AF+2
000617  0245  89F1                            CMZ   +$B1                 INITIALIZE ADDR TO CACHE SIZE + 1
000618  0246  0005                            RTCF                       TURN OFF CLOCK
000619  0247  7C08                            LDV   $R7,=8               INITIALIZE CACHE
000620  0248  E870 0300                       LDR   $R6,=Z'0300'         CHANNEL NO.
000621  024A  0011                            DC    X'11'
000622  024B  F870 00C0                 CLT20  LDR   $R7,=Z'00C0'         SET NO HIT TEST MODE ON
000623  024D  0011                            DC    X'11'
000624  024E  C801                            LDR   $R4,$B1              READ.  WILL CAUSE NO HIT FAULT TRAP
000625                                  *
000626  024F  7C02                            LDV   $R7,=2               DISABLE CACHE
000627  0250  0011                            DC    X'11'
000628  0251  0004                            RTCN                       TURN ON CLOCK
000629  0252  89C0 0307                       CMZ   TEMPA                TRAP?  = 0 - NO TRAP,  = 1 - TRAP
000630  0254  0924                            BE    >CLT60               NO TRAP.  REPORT ERROR
000631  0255  8740 0304                       CL    TEMPA                CLEAR TRAP FLAG
000632  0257  9FC0 0301                 CLT30  STB   $B1,TEMP1-$AF+2      INC. ADDR BY 1K TO NEXT ROW IN THE DIRECTORY
000633  0259  E840 02FE                       LDR   $R6,TEMP1
000634  025B  F840 02FD                       LDR   $R7,TEMP1+1
000635  025D  FA70 0400                       ADD   $R7,=Z'0400'
000636  025F  8ED6                            CAD   =$R6
000637  0260  EF40 02F7                       STR   $R6,TEMP1
000638  0262  FF40 02F6                       STR   $R7,TEMP1+1
000639  0264  9CC0 02F4                       LDB   $B1,TEMP1-$AF+2
000640  0266  9D80 0000       X                CMB   $B1,<ZV$LR           ADDR REGISTER OVERFLOW?
000641  0268  0201 001B                       BL    DPT10                YES
000642  026A  9DC0 0305                       CMB   $B1,MEMHI-$AF+2      ADDR GREATER THAN MAIN MEMORY HIGH?
000643  026C  0301 0017                       BG    DPT10                YES
000644  026E  0005                            RTCF                       NO.  TURN OFF CLOCK
000645  026F  E870 0300                       LDR   $R6,=Z'0300'         CHANNEL NO.
000646  0271  7C03                            LDV   $R7,=3               ENABLE CACHE
000647  0272  0011                            DC    X'11'
000648  0273  0F81 FFD7                       B     CLT20                CONTINUE TEST
000649                                  *
000650                                  *
000651  0275  8AC0 02E4                 CLT50  INC   TEMPA                TRAP.  SET NO HIT FAULT FAULT FLAG
000652  0277  0003                            RTT                        RETURN
000653                                  *
000654                                  CLT60  CALL  ZV$ER,CL1,MSGCL1     ERROR "CL1"
       0278  FBC0 0003
       027A  D380 0000       X
       027C  0F80
       027D  0281
       027E  0513
000655  027F  D3C0 01C3                       LNJ   $B5,PR2              PRINT "REPLACE ARRAY PAC"
000656  0281  0F7F                      CL1    NOP   >$-1                 **  ERROR  **
```

```
000657                                  *                               COMPARATOR LOGIC FAULT
000658                                  *                               REPLACE ARRAY PAC
000659  0282  0F81 FFD4                        B       CLT30           CONTINUE TEST
000660                                  *
000661                                  *
000662                                  *
000663                                  *  DATA PATH TEST
000664                                  ****************
000665                                  *
000666                                  *
000667                                  *  FIRST TEST DATA PATH IN CACHE REPLACEMENT CYCLE TO ISOLATE THE DATA PATH
000668                                  *  FAULT FROM CACHE SERVICE CYCLE AND CACHE UPDATE CYCLE.
000669                                  *
000670                                  *
000671  0284  9870 1000                  DPT10   LDR     $R1,=Z'1000'    INITIALIZE ADDR TO 4K+1
000672  0286  8740 02D1                        CL      TEMP1
000673  0288  9F40 02D0                        STR     $R1,TEMP1+1
000674  028A  9CC0 02CE                        LDB     $B1,TEMP1-$AF+2
000675  028C  5C01                            LDV     $R5,=1          DATA = 0001
000676  028D  DF01                    DPT20   STR     $R5,$B1         WRITE
000677  028E  E870 0300                        LDR     $R6,=Z'0300'    CHANNEL NO.
000678  0290  7C08                            LDV     $R7,=8          INITIALIZE CACHE
000679  0291  0011                            DC      X'11'
000680  0292  C871                            LDR     $R4,+$B1        READ.  BUMP ADDR
000681  0293  7C02                            LDV     $R7,=2          DISABLE CACHE
000682  0294  0011                            DC      X'11'
000683  0295  C955                            CMR     $R4,=$R5        COMPARE DATA
000684  0296  0994                            BNE     >DPT80          REPORT ERROR IF DATA DON'T MATCH
000685  0297  5011                    DPT30   SCL     $R5,1           SHIFT DATA  1 BIT LEFT
000686  0298  5D01                            CMV     $R5,=1          ALL 16 BIT DATA PATH TESTED?
000687  0299  09F4                            BNE     >DPT20          BRANCH TO DPT20 IF NOT
000688                                  *
000689                                  *
000690                                  *
000691                                  *  DATA PATH TEST IN CACHE SERVICE CYCLE
000692                                  *
000693                                  *
000694  029A  9CC0 0000          P             LDB     $B1,ZV$LR       USE MAIN MEMORY LOW AS TEST LOCATION
000695  029C  E870 0300                        LDR     $R6,=Z'0300'    CHANNEL NO.
000696  029E  7C08                            LDV     $R7,=8          INITIALIZE CACHE
000697  029F  0011                            DC      X'11'
000698  02A0  DF01                    DPT50   STR     $R5,$B1         WRITE
000699  02A1  C801                            LDR     $R4,$B1         READ
000700  02A2  7C02                            LDV     $R7,=2          DISABLE CACHE
000701  02A3  0011                            DC      X'11'
000702  02A4  C955                            CMR     $R4,=$R5        COMPARE DATA
000703  02A5  0990                            BNE     >DPT90          REPORT ERROR IF DATA DON'T MATCH
000704  02A6  5011                    DPT60   SCL     $R5,1           SHIFT DATA 1 BIT LEFT
000705  02A7  5D01                            CMV     $R5,=1          ALL 16 BITS DATA PATH TESTED?
000706  02A8  09F8                            BNE     >DPT50          BRANCH TO DPT50 IF NOT
000707  02A9  0F97                            B       >CRT10          DATA PATH TEST DONE
000708                                  *
000709                                  *
000710                                  *
000711                                  DPT80   CALL    ZV$ER,DP1,MSGDP1        ERROR "DP1"
        02AA  FBC0 0003
        02AC  D380 0000           X
        02AE  0F80
        02AF  02B3
        02B0  0515
000712  02B1  D3C0 0187                        LNJ     $B5,PR1         PRINT "REPLACE MOTHER BOARD"
000713  02B3  0F7F                    DP1     NOP     >$-1            **  ERROR  **
000714                                  *                               DATA PATH ERROR (CACHE REPLACEMENT CYCLE)
000715                                  *                               REPLACE MOTHER BOARD
000716                                  *
000717  02B4  0FE3                            B       >DPT30          CONTINUE NEXT BIT (TOTAL 16 BITS) TEST
000718                                  *
000719                                  *
000720                                  DPT90   CALL    ZV$ER,DP2,MSGDP2        ERROR "DP2"
        02B5  FBC0 0003
        02B7  D380 0000           X
        02B9  0F80
        02BA  02BE
        02BB  0517
000721  02BC  D3C0 017C                        LNJ     $B5,PR1         PRINT "REPLACE MOTHER BOARD"
000722  02BE  0F7F                    DP2     NOP     >$-1            **  ERROR  **
000723                                  *                               DATA PATH ERROR (CACHE SERVICE CYCLE)
000724                                  *                               REPLACE MOTHER BOARD
000725  02BF  0FE7                            B       >DPT60          CONTINUE NEXT BIT (TOTAL 16 BIT) TEST
000726                                  *
000727                                  *
000728                                  *
000729                                  *
000730                                  *
000731                                  *  CACHE RAM TEST
000732                                  ****************
000733                                  *
000734                                  *
000735  02C0  E870 0300                 CRT10   LDR     $R6,=Z'0300'    CHANNEL NO.
000736  02C2  7C08                            LDV     $R7,=8          INITIALIZE CACHE
000737  02C3  0011                            DC      X'11'
000738  02C4  7C02                            LDV     $R7,=2          DISABLE CACHE
000739  02C5  0011                            DC      X'11'
000740  02C6  AB80 0342                        LAB     $B2,<CRT150     SET UP FOR UNEXPECTED REG TRAP
000741  02C8  AF80 0000           X            STB     $B2,<ZHTH17
000742  02CA  9870 1000                        LDR     $R1,=Z'1000'    INITIAL ADDR
000743  02CC  8740 028B                        CL      TEMP1
000744  02CE  9F40 028A                        STR     $R1,TEMP1+1
000745  02D0  9CC0 0288                        LDB     $B1,TEMP1-$AF+2 MEMORY ADDR FOR DATA FROM BUFF1
000746  02D2  9FC0 0290                        STB     $B1,TEMP5-$AF+2 SAVE INITIAL ADDR
000747  02D4  AB80 054D                        LAB     $B2,<BUFF1      BUFF1 STORES DATA
000748  02D6  1C00                            LDV     $R1,=0          INDEX
000749  02D7  2CF5                            LDV     $R2,=-11        COUNT NO. OF LOCATIONS IN BUFF1
000750  02D8  E812                    CRT20   LDR     $R6,$B2.$R1     MOVE DATA FROM BUFF1 TO MEMORY
000751  02D9  EF5D                            STR     $R6,$B1.+$R1
000752  02DA  27FE                            BINC    $R2,>CRT20
000753  02DB  9870 1005                        LDR     $R1,=Z'1005'
000754  02DD  8740 027F                        CL      TEMP2
000755  02DF  9F40 027E                        STR     $R1,TEMP2+1     MMM SOURCE ADDR
000756  02E1  1E06                            ADV     $R1,=6
000757  02E2  9F40 0276                        STR     $R1,TEMP1+1
000758  02E4  CCC0 0274                        LDB     $B4,TEMP1-$AF+2 STARTING ADDR
000759  02E6  9CD4                            LDB     $B1,=$B4
```

```
000760  02E7  1EF9                    ADV   $R1,=-7
000761  02E8  9F40 027C               STR   $R1,TEMP6+1          "MMM" ADDR
000764  02EA  8740 0279               CL    TEMP6
000763  02EC  89C0 027E               CMZ   SIZFLG               CACHE SIZE = 4K?
000764  02EE  0981 003A               BNE   CRT110               NO
000765  02F0  E840 026C               LDR   $R6,TEMP2            SOURCE ADDR
000766  02F2  F840 026B               LDR   $R7,TEMP2+1
000767  02F4  FA70 0FFB               ADD   $R7,=Z'0FFB'         ADD 4091 TO SOURCE ADDR
000768  02F6  EF40 0268               STR   $R6,TEMP3            DEST. ADDR
000769  02F8  FF40 0267               STR   $R7,TEMP3+1
000770  02FA  9870 1FF6               LDR   $R1,=Z'1FF6'
000771  02FC  9F40 0264               STR   $R1,TEMP4            NO. OF  BYTES TO MOVE = 8182
000772  02FE  7EFF            CRT30   ADV   $R7,=-1
000773  02FF  FF40 0259               STR   $R7,TEMP1+1
000774  0301  ECC0 0257               LDB   $B6,TEMP1-$AF+2      ENDING ADDR
000775  0303  D870 CA5A               LDR   $R5,=Z'CA5A'         DATA
000776  0305  DF71            CRT40   STR   $R5,+$B1             WRITE, BUMP ADDR
000777  0306  5011                    SCL   $R5,1                CLOSE SHIFT DATA 1 BIT LEFT
000778  0307  9DD6                    CMB   $B1,=$B6             ADDR GREATER THAN ENDING ADDR?
000779  0308  03FD                    BLE   >CRT40               NO
000780                                *
000781                                *  SET UP FOR MEMORY TO MEMORY MOVE
000782                                *
000783  0309  ACC0 0254               LDB   $B2,TEMP2-$AF+2      SOURCE ADDR
000784  030B  2C00                    LDV   $R2,=0               DISPLACEMENT
000785  030C  BCC0 0253               LDB   $B3,TEMP3-$AF+2      DEST. ADDR
000786  030E  3C00                    LDV   $R3,=0               DISPLACEMENT
000787  030F  9840 0251               LDR   $R1,TEMP4            NO. OF BYTES TO MOVE
000788  0311  9CD4                    LDB   $B1,=$B4             STARTING ADDR
000789  0312  D870 CA5A               LDR   $R5,=Z'CA5A'         DATA
000790  0314  8740 0245               CL    TEMPA                CLEAR MMM FLAG
000791  0316  FCC0 024C               LDB   $B7,TEMP5-$AF+2
000792  0318  0005            CRT50   RTCF                       TURN OFF CLOCK
000793  0319  D387                    LNJ   $B5,$B7              JUMP TO CACHE ENABLE
000794                                *
000795                                *  CACHE IS ENABLED.  START MEMORY TO MEMORY MOVE OPERATION.  TEST
000796                                *  PATTERNS ARE WRITTEN INTO CACHE.  READ 1 LOCATION.  WILL TRAP
000797                                *  AND  BRANCH TO REPORT ERROR IF DATA ERROR OCCURES.  IF NO ERROR
000798                                *  IS DETECTED,  DISABLE CACHE AND RETURN.
000799                                *
000800  031A  0004                    RTCN                       TURN ON CLOCK
000801  031B  89C0 023E               CMZ   TEMPA                MMM FLAG = 0?
000802  031D  0901 001B               BE    CRT120               YES
000803  031F  C955            CRT60   CMR   $R4,=$R5             COMPARE DATA
000804  0320  0981 002D               BNE   CRT160               DATA DO NOT MATCH.  REPORT ERROR
000805  0322  9871            CRT70   LDR   $R1,+$B1             DUMMY TO BUMP ADDR
000806  0323  5011                    SCL   $R5,1                CLOSE SHIFT DATA 1 BIT LEFT
000807  0324  9DD6                    CMB   $B1,=$B6             ADDR GREATER THAN ENDING ADDR?
000808  0325  0381 FFF2               BLE   CRT50                NO
000809  0327  0F81 0032               B     RDT10                DONE
000810                                *
000811                                *
000812  0329  E840 0233        CRT110 LDR   $R6,TEMP2            SOURCE ADDR
000813  032B  F840 0232               LDR   $R7,TEMP2+1
000814  032D  FA70 0800               ADD   $R7,=Z'0800'         ADD 2K TO SOURCE ADDR
000815  032F  EF40 022F               STR   $R6,TEMP3            DEST. ADDR
000816  0331  FF40 022E               STR   $R7,TEMP3+1
000817  0333  9870 1000               LDR   $R1,=Z'1000'
000818  0335  9F40 022B               STR   $R1,TEMP4            NO. OF BYTES TO MOVE = 4K
000819  0337  0F81 FFC6               B     CRT30
000820                                *
000821  0339  ACC0 022B        CRT120 LDB   $B2,TEMP6-$AF+2      MMM ADDR
000822  033B  A840 FE02               LDR   $R2,NOP              OVERWRITE MMM WITH NOP
000823  033D  AF02                    STR   $R2,$B2
000824  033E  8AC0 021B               INC   TEMPA                SET MMM FLAG
000825  0340  0F81 FFDE               B     CRT60
000826                                *
000827                        CRT150  CALL  ZV$ER,CR1,MSGCR1     ERROR "CR1"
        0342  FBC0 0003
        0344  D380 0000       X
        0346  0F80
        0347  034B
        0348  0519
000828  0349  D3C0 00F9               LNJ   $B5,PR2              PRINT "REPLACE ARRAY PAC"
000829  034B  0F7F            CR1     NOP   >$-1                 ** ERROR **
000830                                *                          CACHE DATA PARITY ERROR
000831                                *                          SINGLE BIT ERROR DETECTED
000832  034C  0F81 000D               B     RDT10                CONTINUE TEST
000833                                *
000834                        CRT160  CALL  ZV$ER,CR2,MSGCR2     ERROR "CR2"
        034E  FBC0 0003
        0350  D380 0000       X
        0352  0F80
        0353  0357
        0354  051B
000835  0355  D3C0 00ED               LNJ   $B5,PR2              PRINT "REPLACE ARRAY PAC"
000836  0357  0F7F            CR2     NOP   >$-1
000837                                *                          **  ERROR  **
000838                                *                          CACHE DATA ERROR
000839                                *                          DATA FROM CACHE UNEQUAL DATA SHOULD BE
000840                                *                          R4 = DATA FROM CACHE
000841                                *                          R5 = DATA SHOULD BE
000842  0358  0F81 FFC9               B     CRT70                CONTINUE TEST
000843                                *
000844                                *
000845                                *
000846                                *
000847                                *  REPLACEMENT LOGIC AND DIRECTORY TEST
000848                                ****************************************
000849                                *
000850                                *
000851  035A  E870 0300        RDT10  LDR   $R6,=Z'0300'         CHANNEL NO.
000852  035C  7C02                    LDV   $R7,=2               DISABLE CACHE
000853  035D  0011                    DC    X'11'
000854  035E  9C80 0000       X       LDB   $B1,<ZV$LR           ADDR = MAIN MEMORY LOW
000855  0360  9FC0 01F8               STB   $B1,TEMP1-$AF+2
000856  0362  9840 01F6               LDR   $R1,TEMP1+1          DATA = ADDR
000857  0364  9F71            RDT20   STR   $R1,+$B1             WRITE, BUMP ADDR
000858  0365  1E01                    ADV   $R1,=1               BUMP DATA
000859  0366  9D80 0000       X       CMB   $B1,<ZV$LR           ADDR REGISTER OVERFLOW?
000860  0368  0204                    BL    >RDT25               YES
000861  0369  9DC0 0206               CMB   $B1,MEMHI-$AF+2      ADDR GREATER THAN MAIN MEMORY HIGH?
000862  036B  03F9                    BLE   >RDT20               NO
```

```
000863
000864  036C  ACC0 0203          *
000865  036E  AFC0 01EA  RDT25   LDB   $B2,MEMH1-$AF+2     CALCULATE LOCATION TO STORE
000866  0370  9840 01E8          STB   $B2,TEMP1-$AF+2     "CACHE ENABLE" INSTRUCTION
000867  0372  1EF5               LDR   $R1,TEMP1+1
000868  0373  9F40 01E5          ADV   $R1,=-11
000869  0375  ACC0 01E3          STR   $R1,TEMP1+1
000870  0377  BB80 054D          LDB   $B2,TEMP1-$AF+2
000871  0379  1C00               LAB   $B3,<BUFF1          BUFF1 STORES "CACHE ENABLE", ETC.
000872  037A  2CF5               LDV   $R1,=0              INDEX
000873  037B  B813               LDV   $R2,=-11            COUNT NO. OF LOCATIONS IN BUFF1
000874  037C  BF5E       RDT30   LDR   $R3,$B3,$R1         MOVE CONTENTS IN BUFF1 TO MEMORY
000875  037D  27FE               STR   $R3,$B2,+$R1
000876  037E  AFC0 01DA          BINC  $R2,>RDT30
000877  0380  9840 01D8          STB   $B2,TEMP1-$AF+2     OVERWRITE MMM WITH NOP
000878  0382  1E04               LDR   $R1,TEMP1+1
000879  0383  9F40 01D5          ADV   $R1,=4
000880  0385  BCC0 01D3          STR   $R1,TEMP1+1
000881  0387  9840 FDB6          LDB   $B3,TEMP1-$AF+2
000882  0389  9F03               LDR   $R1,NOP
000883  038A  8740 01D2          STR   $R1,$B3
000884  038C  8740 01D1          CL    TEMP2
000885                           CL    TEMP2+1
000886                    *
000887                    *  FIRST TEST THE ADDRESS PATH AND THE REPLACEMENT LOGIC
000888  038E  9CC0 01CF          *
000889  0390  BCD1               LDB   $B1,TEMP2-$AF+2     INITIALIZE ADDR TO 0
000890                           LDB   $B3,=$B1            FOR ADDR REGISTER OVERFLOW CHECK IN
000891  0391  89F1               *                         SAF MODE
000892                           CMZ   +$B1                DUMMY TO BUMP ADDR
000893  0392  E870 0300          *
000894  0394  7C08               LDR   $R6,=Z'0300'        CHANNEL NO.
000895  0395  0011               LDV   $R7,=8              INITIALIZE CACHE
000896                           DC    X'11'
000897  0396  0005       RDT33   RTCF                      TURN OFF CLOCK
000898                    *
000899  0397  D382               LNJ   $B5,$B2             JUMP TO CACHE ENABLE
000900                    *
000901                    *
000902                    *  CACHE IS ENABLED.  READ CACHE.  DISABLE CACHE.  RETURN.
000903                    *
000904  0398  0004               RTCN                      TURN ON CLOCK
000905  0399  D801               LDR   $R5,$B1             READ MAIN MEMORY WITH CASH DISABLED
000906  039A  9DC0 0077          CMB   $B1,LOC15           LOCATION 15 IS THE REAL TIME CLOCK
000907  039C  0904               BE    >RDT35              CURRENT  VALUE
000908  039D  C955               CMR   $R4,=$R5
000909  039E  0981 0067          BNE   RDT180              REPORT ERROR
000910  03A0  9FC0 01B8  RDT35   STB   $B1,TEMP1-$AF+2     SHIFT ADDR 1 BIT LEFT
000911  03A2  E840 01B5          LDR   $R6,TEMP1
000912  03A4  F840 01B4          LDR   $R7,TEMP1+1
000913  03A6  7081               DOL   $R7,1
000914  03A7  EF40 01B0          STR   $R6,TEMP1
000915  03A9  FF40 01AF          STR   $R7,TEMP1+1
000916  03AB  9CC0 01AD          LDB   $B1,TEMP1-$AF+2
000917  03AD  9DD2               CMB   $B1,=$B2            ADDR GREATER THAN "CACHE ENABLE" ADDR?
000918  03AE  0301 0003          BG    RDT37               YES
000919  03B0  9DD3               CMB   $B1,=$B3            B1 OVERFLOW?
000920  03B1  09E5               BNE   >RDT33              NO
000921                    *
000922                    *  NOW TEST ALL LEVELS (2 OR 4) IN THE CACHE DIRECTORY
000923                    *
000924                    *
000925  03B2  9CC0 01AB  RDT37   LDB   $B1,TEMP2-$AF+2     REINITIALIZE ADDR TO 0
000926  03B4  8740 01AF          CL    TEMP6               INITIALIZE FLAG
000927  03B6  3C00               LDV   $R3,=0              INITIALIZE COUNTER A
000928                    *
000929                    *
000930                    *
000931  03B7  E870 0300          LDR   $R6,=Z'0300'        CHANNEL NO.
000932  03B9  7C08               LDV   $R7,=8              INITIALIZE CACHE
000933  03BA  0011               DC    X'11'
000934                    *
000935  03BB  0005       RDT40   RTCF                      TURN OFF CLOCK
000936  03BC  D382               LNJ   $B5,$B2             JUMP TO CACHE ENABLE
000937                    *
000938                    *
000939                    *  CACHE IS ENABLED.  READ CACHE.  DISABLE CACHE.  RETURN.
000940                    *
000941  03BD  0004               RTCN                      TURN ON CLOCK
000942  03BE  D871               LDR   $R5,+$B1            READ MAIN MEMORY WITH CACHE DISABLED
000943                    *                                POST INC. ADDR
000944  03BF  9DC0 0053          CMB   $B1,LOC16           LOCATION 15 IS THE REAL TIME CLOCK
000945  03C1  0904               BE    >RDT50              CURRENT VALUE
000946  03C2  C955               CMR   $R4,=$R5            COMPARE CACHE DATA TO MAIN MEMORY DATA
000947  03C3  0981 0036          BNE   RDT170              DATA DO NOT MATCH.  REPORT ERROR
000948  03C5  89C0 019E  RDT50   CMZ   TEMP6               FLAG = 0?
000949  03C7  0981 001C          BNE   RDT110              NO
000950  03C9  9DD2               CMB   $B1,=$B2            ADDR LESS THAN "CACHE ENABLE" ADDR?
000951  03CA  0201 FFF0          BL    RDT40               YES.  TEST NEXT LOCATION
000952  03CC  8AD3               INC   =$R3                NO.  INC. COUNTER A
000953  03CD  8AC0 0196          INC   TEMP6               SET FLAG
000954  03CF  3D01               CMV   $R3,=1              COUNTER A = 1?
000955  03D0  0901 0019          BE    RDT120
000956  03D2  8740 0185          CL    TEMP1
000957  03D4  8740 0184          CL    TEMP1+1
000958  03D6  9CF0 0559          LDB   $B1,=TEMP1-$AF+2    SET ADDR TO 0
000959  03D8  9853               LDR   $R1,=$R3            CALCULATE ENDING ADDR
000960  03D9  9B70 0400          MUL   $R1,=1024
000961  03DB  1EFF               ADV   $R1,=-1
000962  03DC  8740 017B          CL    TEMP1
000963  03DE  9F40 017A          STR   $R1,TEMP1+1
000964  03E0  CCC0 0178          LDB   $B4,TEMP1-$AF+2
000965  03E2  0F81 FFD8          B     RDT40
000966                    *
000967                    *
000968                    *
000969  03E4  9DD4       RDT110  CMB   $B1,=$B4            ADDR GREATTER THAN ENDING ADDR?
000970  03E5  0381 FFD5          BLE   RDT40               NO
000971  03E7  3D03               CMV   $R3,=3              COUNTER A
000972  03E8  0281 002B          BGE   ULT10               GO TO NEXT TEST IF DONE
000973                    *
000974                    *
000975                    *
```

```
000976  03EA  E840 0172          RDT120 LDR   $R6,TEMP2
000977  03EC  F840 0171                 LDR   $R7,TEMP2+1
000978  03EE  FA70 03FF                 ADD   $R7,=Z'03FF'
000979  03F0  EF40 016C                 STR   $R6,TEMP2
000980  03F2  FF40 016B                 STR   $R7,TEMP2+1          UPDATE INITIAL ADDR
000981  03F4  9CC0 0169                 LDB   $B1,TEMP2-$AF+2      REINITIALIZE ADDR
000982  03F6  8740 016D                 CL    TEMP6                RESET FLAG
000983  03F8  0F81 FFC2                 B     RDT40
000984                             *
000985                             RDT170 CALL  ZV$ER,RD1,MSGRD1     ERROR "RD1"
        03FA  FBC0 0003
        03FC  D380 0000      X
        03FE  0F80
        03FF  0403
        0400  0510
000986  0401  D3C0 0041                 LNJ   $B5,PR2              PRINT "REPLACE ARRAY PAC"
000987  0403  0F7F             RD1     NOP   >$-1                 **  ERROR  **
000988                             *                              DIRECTORY ERROR
000989                             *                              R4 = DATA IS
000990                             *                              R5 = DATA SHOULD BE
000991  0404  0F81 FFC0                 B     RDT50                CONTINUE TEST
000992                             *
000993                             *
000994                             RDT180 CALL  ZV$ER,RD2,MSGRD2     ERROR "RD2"
        0406  FBC0 0003
        0408  D380 0000      X
        040A  0F80
        040B  040F
        040C  051F
000995  040D  D3C0 002B                 LNJ   $B5,PR1              PRINT "REPLACE MOTHER BOARD"
000996  040F  0F7F             RD2     NOP   >$-1                 **  ERROR  **
000997                             *                              REPLACEMENT LOGIC ERROR
000998                             *
000999  0410  0F81 FF8F                 B     RDT35                CONTINUE TEST
001000                             *
001001                             *
001002                             *
001003  0412        0000      X    LOC15  DC    <ZHRTCC
001004                             *
001005  0413        0001      X    LOC16  DC    <ZHRTCC+1
001006                             *
001007                             *
001008                             *
001009                             *
001010                             *
001011                             *  UPDATE LOGIC TEST
001012                             ********************
001013                             *
001014                             *
001015  0414  9CC0 0000      P      ULT10  LDB   $B1,ZV$LR           ADDR = MAIN MEMORY LOW
001016  0416  9FC0 0142                 STB   $B1,TEMP1-$AF+2
001017  0418  D840 0140                 LDR   $R5,TEMP1+1          DATA = ADDR
001018  041A  E870 3000                 LDR   $R6,=Z'3000'         CHANNEL NO.
001019  041C  7C06                      LDV   $R7,=8               INITIALIZE CACHE
001020  041D  0011                      DC    X'11'
001021  041E  DF01                 ULT20  STR   $R5,$B1             WRITE
001022  041F  C871                      LDR   $R4,*$B1             READ.  BUMP ADDR
001023  0420  C955                      CMR   $R4,=$R5             COMPARE DATA
001024  0421  098D                      BNE   >ULT40               DATA DON'T MATCH.  REPORT ERROR
001025  0422  5E01                      ADV   $R5,=1               BUMP DATA
001026  0423  9D80 0000      X         CMB   $B1,<ZV$LR           ADDR REGISTER OVERFLOW?
001027  0425  0204                      BL    >ULT30               YES
001028  0426  9DC0 0149                 CMB   $B1,MEMHI-$AF+2      ADDR GREATER THAN MAIN MEMORY HIGH?
001029  0428  03F6                      BLE   >ULT20               NO.  TEST NEXT LOCATION
001030  0429  F870 0002           ULT30  LDR   $R7,=2               YES.  DISABLE CACHE
001031  042B  0011                      DC    X'11'
001032  042C  0F81 0027                 B     EOP                  DONE
001033                             *
001034                             *
001035                             ULT40  CALL  ZV$ER,ULT1,MSGULT1   ERROR "ULT1"
        042E  FBC0 0003
        0430  D380 0000      X
        0432  0F80
        0433  0437
        0434  0521
001036  0435  D3C0 0003                 LNJ   $B5,PR1              PRINT "REPLACE MOTHER BOARD"
001037  0437  0F7F             ULT1    NOP   >$-1                 **  ERROR  **
001038                             *                              UPDATE LOGIC FAULT
001039                             *                              REPLACE MOTHER BOARD
001040  0438  0FE6                      B     >ULT20               CONTINUE TEST
001041                             *
001042                             *
001043                             *
001044                             *
001045                             *
001046                             *
001047                             *  SUBROUTINES
001048                             **************
001049                             *
001050                             *
001051                             *
001052                             *  SUBROUTINE TO PRINT FAULTY BOARD
001053                             *
001054  0439  DFC0 0121            PR1    STB   $B5,TEMPC            SAVE B5
001055                                    CALL  ZV$TC,MSGCB          PRINT "REPLACE MOTHER BOARD"
        043B  FBC0 0003
        043D  D380 0000      X
        043F  0F80
        0440  0525
001056  0441  8388 055B                 JMP   *<TEMPC              RETURN
001057                             *
001058  0443  DFC0 0117            PR2    STB   $B5,TEMPC            SAVE B5
001059                                    CALL  ZV$TC,MSGAP          PRINT "REPLACE DAUGHTER BOARD"
        0445  FBC0 0003
        0447  D380 0000      X
        0449  0F80
        044A  0532
001060  044B  8388 055B                 JMP   *<TEMPC              RETURN
001061                             *
001062                             *
001063                             *
001064                             *
001065                             *
```

```
001066                                  *  SUBROUTINE TO SET UP FOR RED OR PARITY ERRORS
001067                                  *
001068                                  *
001069  044D  9BC0 0003                        LAB   $B1,TH17C           SET UP FOR UNEXPECTED RED OR PARITY
001070  044F  9F80 0000        X                STB   $B1,<ZHTH17         ERRORS
001071                                  *
001072                                  *
001073  0451  8AC0 0115                 TH17C  INC   TRPFLG              SET RED TRAP FLAG
001074  0453  0003                             RTT                       RETURN
001075                                  *
001076                                  *
001077                                  *
001078                                  *
001079                                  *  END OF PASS
001080                                  **************
001081                                  *
001082                                  *
001083                                  *
001084  0454  8AC0 0113                 EOP    INC   PASCNT+$AF-1         BUMP PASS COUNT
001085                                  *
001086                                  *  DISPLAY PASS COUNT IN B6 IN DECIMAL
001087                                  *
001088  0456  F840 0111                        LDR   $R7,PASCNT+$AF-1
001089  0458  A870 0003                        LDR   $R2,=3
001090  045A  8756                     DIGIT  CL    =$R6
001091  045B  F370 000A                        DIV   $R7,=10
001092  045D  EAD5                             SRM   $R6,=$R5,=Z'000F'
        045E  000F
001093  045F  5054                             SCR   $R5,4
001094  0460  277A                             BDEC  $R2,>DIGIT
001095  0461  DF40 0108                        STR   $R5,B6DCML+1
001096  0463  ECC0 0106                        LDB   $B6,B6DCML-$AF+2    **  B6 RESERVED FOR PASS COUNT DISPLAY  **
001097                                  *
001098                                  *  TYPE PASS COUNT AND "DONE" IF 3 MINUTES ARE UP
001099                                  *
001100                                  *
001101  0465  9800 0000        X               LDR   $R1,<ZHRTCC          NO, ARE 3 MINUTES UP?
001102  0467  9970 D20F                        CMR   $R1,=Z'D20F'         AB9F = 43935(DEC) = 65535-21600(DEC).
001103                                  *                                 AB9F IS  THE NORMAL REAL TIME CLOCK VALUE
001104                                  *                                 D20F IS THE CORRECTED CLOCK VALUE
001105                                  *                                 BECAUSE THE CLOCK IS TURNED OFF WHEN
001106                                  *                                 THE CACHE IS ENABLED
001107                                  *
001108                                  *  CLOCK SET INITIALLY TO FFFF = 65535(DEC).  21600 TICKS = 3 MINUTES.
001109                                  *
001110  0469  0319                             BG    >GLEN                3 MINUTES NOT UP YET
001111                                  *
001112  046A  1CFF                             LDV   $R1,=-1              SET UP FOR NEXT 3 MIN INTERVAL
001113  046B  9F00 0000        X               STR   $R1,<ZHRTCC
001114                                          CALL  ZV$TC,MSG6           TYPE PASS MESSAGE
        046D  FBC0 0003
        046F  D380 0000        X
        0471  0F80
        0472  0504
001115                                          CALL  ZV$TD,PASCNT+$AF-1   TYPE THE COUNT
        0473  FBC0 0003
        0475  D380 0000        X
        0477  0F80
        0478  0568
001116                                          CALL  ZV$TC,MSG7           TYPE "3 MIN DONE"
        0479  FBC0 0003
        047B  D380 0000        X
        047D  0F80
        047E  0509
001117                                  *
001118  047F  0004                             RTCN
001119  0480  0F81 FC8A                        B     DONE                 **  3 MIN TESTING COMPLETE  **
001120                                  *
001121                                  *  TYPE PASS COUNT
001122                                  *
001123                                  GLEN   CALL  ZV$TC,MSG6           TYPE PASS MESSAGE
        0482  FBC0 0003
        0484  D380 0000        X
        0486  0F80
        0487  0504
001124                                          CALL  ZV$TD,PASCNT+$AF-1   TYPE THE COUNT
        0488  FBC0 0003
        048A  D380 0000        X
        048C  0F80
        048D  0568
001125                                  *
001126                                  *
001127  048E  0004                             RTCN                       TURN ON THE CLOCK
001128                                  *
001129                                  *
001130  048F  0F81 FD7E                        B     NXTPAS               REPEAT TEST FOR NEXT PASS
001131                                  *
001132                                  *
001133                                  *
001134                                  *  MESSAGES
001135                                  ************
001136                                  *
001137                                  *
001138                                  *
001139                                          IFZ   $AF-2,LAF1
001140  0491  636D 6373 3120            MSG1   TEXT  'CMC51  CACHE MEMORY TEST'
        0494  2063 6163 6865
              206D 656D 6F72
              7920 7465 7374
001141                                  LAF1   NULL
001144                                  SAF1   NULL
001145  049D  2020 2032 3120                  TEXT  '   21 APR 1978   REV. B$'
        04A0  4170 7220 3139
              3738 2020 2052
              4556 2E20 4224
001146  04A9  6D65 6D6F 7279            MSG2   TEXT  'MEMORY GREATER THAN 64K, CHANGE TO CMCL1 ,',Z'0D0A';
        04AC  2067 7265 6174
001147  04AF  6572 2074 6861                       '  SET TO LAF MODE.  OTHER WISE, WILL$'
        04B2  6E20 3634 6B2C
              2063 6861 6E67
              6520 746F 2063
              6D63 6C31 202C
              0D0A 2020 7365
```

```
                    7420 746F 206C
                    6166 206D 6F64
                    652E 2020 6F74
                    6865 7220 7769
                    7365 2C20 7769
                    6C6C 2400
001148  04D2  2020 6F6E 6C79        MSG2B   TEXT  '  ONLY USE UP TO 64K LOCATIONS!$'
        04D5  2075 7365 2075
              7020 746F 2036
              346B 206C 6F63
              6174 696F 6E73
              2124
001149  04E2  6361 6368 6520        MSG2A   TEXT  'CACHE PRESENT$'
        04E5  7072 6573 656E
              7424
001150  04E9  6361 6368 6520        MSG3    TEXT  'CACHE NOT PRESENT$'
        04EC  6E6F 7420 7072
              6573 656E 7424
001151  04F2  6361 6368 6520        MSG3A   TEXT  'CACHE SIZE IS 4K$'
        04F5  7369 7A65 2069
              7320 346B 2400
001152  04FB  6361 6368 6520        MSG3B   TEXT  'CACHE SIZE IS 2K$'
        04FE  7369 7A65 2069
              7320 326B 2400
001153                              IFZ   $AF-2,LAF2
001154  0504  636D 6373 3120        MSG6    TEXT  'CMCS1  P5$'
        0507  2070 7324
001155                              LAF2    NULL
001158                              SAF2    NULL
001159  0509  3320 6D69 6E20        MSG7    TEXT  '3 MIN DONE$'
        050C  646F 6E65 2400
001160  050F  6463 3124            MSGDC1  TEXT  'DC1$'
001161  0511  6463 3224            MSGDC2  TEXT  'DC2$'
001162  0513  636C 3124            MSGCL1  TEXT  'CL1$'
001163  0515  6470 3124            MSGDP1  TEXT  'DP1$'
001164  0517  6470 3224            MSGDP2  TEXT  'DP2$'
001165  0519  6372 3124            MSGCR1  TEXT  'CR1$'
001166  051B  6372 3224            MSGCR2  TEXT  'CR2$'
001167  051D  7264 3124            MSGRD1  TEXT  'RD1$'
001168  051F  7264 3224            MSGRD2  TEXT  'RD2$'
001169  0521  756C 3124            MSGUL1  TEXT  'UL1$'
001170  0523  7272 3124            MSGRR1  TEXT  'RR1$'
001171  0525  7265 706C 6163        MSGCB   TEXT  'REPLACE CONTROLLER BOARD$'
        0528  6520 636F 6E74
              726F 6C6C 6572
              2062 6F61 7264
              2400
001172  0532  7265 706C 6163        MSGAP   TEXT  'REPLACE ARRAY PAC$'
        0535  6520 6172 7261
              7920 7061 6324
001173                              *
001174                              *
001175                              *
001176                              *  RESERVED AREA
001177                              ****************
001178                              *
001179                              *
001180                              *
001181                              *
001182                              *  INTERRUPT SAVE/RESTORE AREA FOR LEVEL 4
001183                              *  (RESTORED ON LEVEL CHANGE FROM 0,1 OR 3 TO 4)
001184                              *
001185  053B  0000                  ISA4    RESV  $AF,0           TRAP SAVE AREA POINTER
001186  053C  0000                          RESV  1,0             I.D. OF INTERRUPTING DEVICE
001187  053D  0000                          RESV  1,0             KUPT SAVE MASK 1
001188  053E  0000                          RESV  1,0             KUPT SAVE MASK 2
001189  053F  0133                  ISA4P   DC    <T4A1           P-REG
001190  0540  4000                          DC    X'4000'         S-REG, SET PRIVILEGE MODE
001191                              *                             FOR INTERRUPT HANDLER
001192                              *
001193                              *
001194                              *  SECONDARY TRAP SAVE AREA
001195                              *
001196  0541  0000                  SECTSA RESV  12,0             LINKED TO TSA AT X'0002' TO PREVENT
001197                              *                             LEVEL 2 INTERRUPT WHEN TRAP AND
001198                              *                             RECORDS MACHINE STATE IN CASE OF
001199                              *                             DOUBLE (UNEXPECTED) TRAP.
001200                              *
001201                              *
001202                              *
001203                              *
001204  054D  E870                  BUFF1   DC    Z'E870'         CACHE ENABLE
001205  054E  0300                          DC    Z'0300'
001206  054F  7C03                          DC    Z'7C03'
001207  0550  0011                          DC    Z'0011'
001208  0551  0008                          DC    Z'0008'         MEMORY TO MEMORY MOVE
001209  0552  C801                          DC    Z'C801'         READ (DATA IN R4, ADDR IN B1)
001210  0553  E870                          DC    Z'E870'         CACHE DISABLE
001211  0554  0300                          DC    Z'0300'
001212  0555  7C02                          DC    Z'7C02'
001213  0556  0011                          DC    Z'0011'
001214  0557  8385                          DC    Z'8385'         RETURN
001215                              *
001216                              *
001217                              *
001218  0558  0000                  TEMP1   RESV  2,0
001219  055A  0000                  TEMPA   RESV  1,0
001220  055B  0000                  TEMPC   RESV  2,0
001221  055D  0000                  TEMP2   RESV  2,0
001222  055F  0000                  TEMP3   RESV  2,0
001223  0561  0000                  TEMP4   RESV  1,0
001224  0562  0000                  TEMP5   RESV  2,0
001225  0564  0000                  TEMP6   RESV  2,0
001226  0566  0000                  CAFLG   RESV  1,0             =0 IF CACHE  IS PRESENT
001227                              *                             =1 IF NOT
001228  0567  0000                  TRPFLG  RESV  1,0             =0 IF NO TRAP OCCURED
001229                              *                             =1 IF TRAPS
001230  0568  0000                  PASCNT  RESV  $AF,0           PASS COUNTER
001231                              *
001232  0569  0000                  B6DCML  RESV  2,0             FOR B6 (PASS COUNT) RESTORATION
001233                              *
001234                              *
001235  056B  0000                  SIZFLG  RESV  1,0             =0 IF CACHE SIZE IS 4K
```

```
001236                                  *                               =1 IF CACHE SIZE IS 2K
001237                                  *
001238  056C  0000                     CASIZE RESV  2,0                 STORES LAST CACHE ADDR
001239                                  *
001240  056E  0000                     NULL   RESV  $AF,0
001241                                  *
001242  056F  0000                     MEMHI  RESV  2,0                 MEMORY HIGH IF PROM IS PRESENT
001243                                  *
001244  0571  0000                     PLOW   RESV  2,0                 PROM 1ST LOCATION
001245                                  *
001246                                  *
001247                                  *
001248                                  *
001249                                  *
001250  0573  636F 7079 7269            TEXT  'COPYRIGHT 1977, BY HONEYWELL INC.$'
      0576  6768 7420 3139
            3737 2C20 6279
            2068 6F6E 6579
            7765 6C6C 2069
            6E63 2E24
001251                                  *
001252                                  *
001253  0584  0100                           END   CMCS1,START
0000  ERR COUNT
      TITLE CMCS1,'REV B'
      $AF       276   278   290C  315C  319   467   527C  540C  542C  570
                573   616   632C  639   642   674   745   746C  758   774
                783   785   791   821   855C  861   864   865C  869   876C
                880   888   910C  916   925   958   964   981   1016C 1028
                1084C 1088  1096  1115  1124  1139  1142  1153  1156  1185
                1230  1240
      $B1       310   311C  448   450   457   459   467   468   513   514
                516   518C  519   522   526   527C  531   536   540C  541
                542C  573   580   586   616   617   624   632C  639   640
                642   674   676C  680   694   698C  699   745   746C  751C
                759   776C  778   788   805   807   854   855C  857C  859
                861   888   889   891   905   906   910C  916   917   919
                925   942   944   950   958   969   981   1015  1016C 1021C
                1022  1026  1028  1069  1070C
      $B2       372   373C  405   406C  436   437C  508   509C  510   511C
                512   522   566   567C  570   586   614   615C  740   741C
                747   750   783   821   823C  864   865C  869   874C  876C
                899B  917   936B  950
      $B3       319   320C  321   322C  409   417   785   870   873   880
                882C  889   919
      $B4       758   759   788   964   969
      $B5       422B  595B  655B  712B  721B  793B  828B  835B  899B  936B
                986B  995B  1036B 1054C 1058C
      $B6       337   774   778   807   1096
      $B7       791   793B
      $R1       306   307C  341   342C  343C  344   345C  350   351   532
                533   534   671   673C  742   744C  748   750   751C  753
                755C  756   757C  760   761C  770   771C  787   805   817
                818C  856   857C  858   866   867   868C  871   873   874C
                877   878   879C  881   882C  959   960   961   963C  1101
                1102  1112  1113C
      $R2       440   441C  498   499C  749   752B  784   822   823C  872
                875B  1089  1094B
      $R3       786   873   874C  927   952C  954   959   971
      $R4       417   450   459   468   516   517C  519   520   531   532
                536   541   580   624   680   683   699   702   803   908
                946   1022  1023
      $R5       675   676C  683   685   686   698C  702   704   705   775
                776C  777   789   803   806   905   908   942   946   1017
                1021C 1023  1025  1092  1093  1095C
      $R6       288   377   412   444   575   620   633   636   637C  645
                677   695   735   750   751C  765   768C  812   815C  851
                893   911   914C  931   976   979C  1018  1090C 1092
      $R7       287   376   380   411   415   443   446   576   578   599
                619   622   626   634   635   638C  646   678   681   696
                700   736   738   766   767   769C  772   773C  813   814
                816C  852   894   912   913   915C  932   977   978   980C
                1019  1030  1088  1091
1232  B6DCML    1095C 1096
1204  BUFF1     747   870
1226  CAFLG     374C  383   394C
1238  CASIZE    439C  441C  499C  570   616
656   CL1       654
364   CLOCK     352B  354B
614   CLT10     591B
622   CLT20     648B
632   CLT30     659B
651   CLT50     614
654   CLT60     630B
      CMCS1       1   1253
508   CPP10     471B  478B  494B
516   CPP20     523B  537B
531   CPP40     521B
540   CPP50     535B
546   CPP60     508   510
829   CR1       827
836   CR2       834
735   CRT10     707B
812   CRT110    764B
821   CRT120    802B
827   CRT150    740
834   CRT160    804B
750   CRT20     752B
772   CRT30     819B
776   CRT40     779B
792   CRT50     808B
803   CRT60     825B
805   CRT70     842B
423   DC1       421
475   DC2       474
428   DCS10     405
430   DCS20     409
436   DCS30     420B  425B
483   DCS40     436
405   DCS5      388B
487   DCS50     448
490   DCS60     457
493   DCS70     455B
```

```
 497  DCS80      462B
1090  DIGIT     1094B
 295  DONE      1119B
 713  DP1        711
 722  DP2        720
 671  DPT10      641B     643B
 676  DPT20      687B
 685  DPT30      717B
 698  DPT50      706B
 704  DPT60      725B
 711  DPT80      684B
 720  DPT90      703B
1084  EOP       1032B
1123  GLEN      1110B
 305  INSTRT     300B
1185  ISA4       314C     315C     319
1189  ISA4P      322C
1141  LAF1      1139
1155  LAF2      1153
1003  LOC15      906
1005  LOC16      944
1242  MEMHI      527C     542C     642      861      864      1028
1140  MSG1       305
1143  MSG1
1146  MSG2       355
1149  MSG2A      387
1148  MSG2B      356
1150  MSG3       391
1151  MSG3A      493
1152  MSG3B      497
1154  MSG6      1114     1123
1157  MSG6
1159  MSG7      1116
1172  MSGAP     1059
1171  MSGCB     1055
1162  MSGCL1     654
1165  MSGCR1     827
1166  MSGCR2     834
1160  MSGDC1     421
1161  MSGDC2     474
1163  MSGDP1     711
1164  MSGDP2     720
1167  MSGKD1     985
1168  MSGKD2     994
1170  MSGRR1     594
1169  MSGUL1    1035
 347  NOP        306      822      881
1240  NULL       337      513
 557  NXTPAS     297B    1130B
1230  PASCNT     290C     1084C    1088     1115     1124
1244  PLOW       540C
1054  PR1        595B     712B     721B     995B     1036B
1058  PR2        422B     655B     828B     835B      986B
 987  RD1        985
 996  RD2        994
 851  RDT10      809B     832B
 969  RDT110     949B
 976  RDT120     955B
 985  RDT170     947B
 994  RDT180     909B
 857  RDT20      862B
 864  RDT25      860B
 873  RDT30      875B
 897  RDT33      920B
 910  RDT35      907B     999B
 925  RDT37      918B
 935  RDT40      951B     965B     970B     983B
 948  RDT50      945B     991B
 596  RR1        594
 566  RRT10      528B     543B
 580  RRT20      584B     601B
 594  RRT30      587B
 604  RRT40      566
1144  SAF1      1142
1158  SAF2      1156
1196  SECTSA     310
      SET       1147
1235  SIZFLG     442C     454      456C     461      464C     470      483C     763
 285  START      1253
 300  START1     291B     307C
 301  START2     308B
 310  START3     302B
 337  T4A1       321     1189
 372  TCP10
 391  TCP15      384B
 394  TCP20      372
1218  TEMP1      467      571C     572C     573      632C     633      634      637C     638C     639
                672C     673C     674      743C     744C     745      757C     758      773C     774
                855C     856      865C     866      868C     869      876C     877      879C     880
                910C     911      912      914C     915C     916      956C     957C     958      962C
                963C     964     1016C    1017
1221  TEMP2      465C     466C     754C     755C     765      766      783      812      813      883C
                884C     888      925      976      977      979C     980C     981
1222  TEMP3      768C     769C     785      815C     816C
1223  TEMP4      771C     787      818C
1224  TEMP5      746C     791
1225  TEMP6      761C     762C     821      926C     948      953C     982C
1219  TEMPA      408C     419      428C     517C     629      631C     651C     790C     801      824C
1220  TEMPC     1054C    1056B    1058C    1060B
1073  TH17C     1069
1228  TRPFLG     569C     583      585C     604C     1073C
1037  UL1       1035
1015  ULT10      972B
1021  ULT20     1029B    1040B
1030  ULT30     1027B
1035  ULT40     1024B
 280  ZERO       281      285      286
      ZHCOMM     265      430      487      490
 276  ZHISA4     320C
      ZHISAZ     275      276
      ZHNTSA     267
      ZHRTCC     269      342C     1003     1005     1101     1113C
      ZHRTCI     268      343C
```

```
          ZHRTCL    270     345C
          ZHTH15    271     373C    406C    437C    509C    567C    615C
          ZHTH17    273     511C    741C   1070C
          ZHTSA     266     278     311C
    278   ZHTSA7
          ZV$AF     259     350
          ZV$ER     421B    474B    594B    654B    711B    720B    827B    834B    985B    994B
                   1035B
          ZV$HR     260     512     526
          ZV$HRU    261     353
          ZV$LK     262     514     640     694     854     859    1015    1026
          ZV$RD     301B    305B
          ZV$TC     355B    356B    387B    391B    493B    497B   1055B   1059B   1114B   1116B
                   1123B
          ZV$TD     1115B   1124B
          ZV$TTY    263
          ZV$UIH    264     328B
 178 LABELS
 761 REFERENCES
1253 RECORDS
   1 U FLAGS
   2 M FLAGS
   4 N FLAGS
6 CROSS REF VERSION L - 24 SEPT, 1976
RS LINKER VERSION 5.00 04/27/78  1715.1 EST THU
LINK MAP FOR CMCS1
 START          0100
 LOW            0000
 HIGH           0CF2
 CURRENT        0CF3
*LOC DEFS
  ZHCOMM        0000
 *CMCS1         0000   REV B
  ZHPFR         0000
  ZHTSA         0002
  ZHNTSA        0010
  ZHRTC1        0014
  ZHRTCC        0015
  ZHRTCL        0016
  ZHWDTC        0017
  ZHMERC        001F
  ZHIAFB        0020
  ZHTH29        0063
  ZHTH28        0064
  ZHTH27        0065
  ZHTH26        0066
  ZHTH25        0067
  ZHTH24        0068
  ZHTH23        0069
  ZHTH22        006A
  ZHTH21        006B
  ZHTH20        006C
  ZHTH19        006D
  ZHTH18        006E
  ZHTH17        006F
  ZHMEMP        006F
  ZHTH16        0070
  ZHLERR        0070
  ZHTH15        0071
  ZHNRES        0071
  ZHTH14        0072
  ZHPMEM        0072
  ZHTH13        0073
  ZHP-OP        0073
  ZHTH12        0074
  ZHTH11        0075
  ZHTH10        0076
  ZHTH9         0077
  ZHTH8         0078
  ZHTH7         0079
  ZHTH6         007A
  ZHOVFL        007A
  ZHTH5         007B
  ZHOP-N        007B
  ZHTH4         007C
  ZHTH3         007D
  ZHSC-N        007D
  ZHTH2         007E
  ZHTRC         007E
  ZHTH1         007F
  ZHMCL         007F
  ZHISA2        0080
  ZHIVBS        0080
  ZHTVBS        0080
 *ZV$TH         0584
  ZV$TD         05B9
  ZV$TH         0584
  ZV$TH2        05AC
 *ZV$IH         05D4
  ZV$IH         05D4
  ZV$ID         05D9
  ZV$IAD        05DE
  ZV$--2        05F6
  ZV$--3        0608
 *ZV$ER         066D   REV. 5.0
  ZV$ER         066D
  ZV$TA         0699
  ZV$--0        0680
 *ZV$GP         06DD
  ZV$GP         06DD
  ZV$--4        06FD
 *ZV$HA         0709
  ZV$HA         0709
  ZV$H2         0713
  ZV$H5         070E
 *ZV$T          0742   REV. 5.0
  ZV$TC         074B
  ZV$T          0742
  ZV$UC         075F
  ZV$U          0754
 *ZV$HD         0773
  ZV$HD         0773
 *ZV$IA         07A5   REV. 7
```

```
  ZV$IA          07A8
  ZV$ARG         0857
  ZV$ABF         0859
  ZV$---1        0814
  ZV$IAV         07A6
 *ZV$RD          0864   REV. 7
  ZV$RD          0864
  ZV$UIH         0894
  ZV$AF          0875
  ZV$HRU         088D
  ZV$HR          0893
  ZV$LR          0890
  ZV$DAT         0873
  ZV$HM          08DA
  ZV$SV1         0A39
  ZV$SV3         0A59
  ZV$TTY         0877
  ZV$SV2         0A49
  ZV$OTP         090B
  ZV$TID         0876
  ZV$CF2         0880
  ZV$TK          087C
  ZV$KAR         087D
  ZV$ST1         0881
  ZV$RCC         0882
  ZV$BUD         0878
  ZV$OLB         0884
  ZV$RCB         0885
  ZV$NSR         0889
  ZV$STR         0887
  ZV$IZ          089E
  ZV$HRL         088E
  ZV$LRU         088F
  ZV$LRL         0890
  ZV$HBD         0891
  ZV$CF1         087F
  ZV$BKS         088B
  ZV$BKF         088C
  ZV$---5        0896
  ZV$RMD         0874
  ZV$MCP         0892
  HIBAUD         0891
  ZV$RAW         087E
  ZV$RDT         0A95
  ZV$CTL         087B
  ZV$B1          09B6
  ZV$TST         0AEB
  ZV$MDC         0ABF
  ZV$R99         0CBD
  ZV$ISA         0899
  ZV$ZRO         0918
  ZV$BSH         091A
  ZV$CPU         087A
  ZV$R50         08F8
  ZV$R60         0903
  ZV$RT          0BFA
  ZV$ALL         0879
 *MLCHPG         0CC2   T+V
  MLCHPG         0CC2
  ENDCHP         0CF3
*UNLINK MODULE(S)
 ZV$TC
 ZV$T
```