IDENTIFICATION

PRODUCT CODE: AC-E1108-MC
PRODUCT NAME: CFRKABO PDP11/34 CACHE
DATE CREATED: MAY 1979
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: SCOTT GORDON
REVISED BY JOHN W. CIUKAJ
REV. DATE JAN 1979

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

COPYRIGHT (C) 1978, 1979 BY DIGITAL EQUIPMENT CORPORATION
TABLE OF CONTENTS

1.0 ABSTRACT

2.0 SYSTEM REQUIREMENTS
   2.1 HARDWARE
   2.2 SOFTWARE
   2.3 ACT & APT SETUP
   2.4 EXECUTION TIME

3.0 DIAGNOSTIC HIERARCHY PREREQUISITES

4.0 STARTING ADDRESS

5.0 PROGRAM CONTROL AND OPERATOR ACTION

6.0 PROGRAM DESCRIPTION

7.0 ERROR REPORTING

8.0 HANDLERS AND COMMON ROUTINES

9.0 REV B CHANGES

1.0 ABSTRACT

THE 11/34 CACHE DIAGNOSTIC IS COMPRIS ED OF A SERIES OF TESTS
WHICH WERE DESIGNED TO CHECK THE CACHE'S DATA PATHS AND ITS
CONTROL LOGIC. THE TESTS ARE ARRANGED IN A LOGICAL ORDER SUCH THAT
THEY BUILD ON ONE ANOTHER. THAT IS, THE CURRENTLY RUNNING
TEST WILL DEPEND ON LOGIC EXERCISED BY PREVIOUS TESTS. THOSE TESTS
REQUIREING EXTENSIVE AMOUNTS OF CACHE FUNCTIONING ARE DONE
NEAR THE END OF THE PROGRAM. THIS TESTING PROCEDURE SHOULD PROVIDE
AN EFFECTIVE DEGREE OF FAULT ISOLATION.

2.0 SYSTEM REQUIREMENTS

2.1 HARDWARE

1. A WORKING 11/34 CPU
2. A MINIMUM OF 16K TO A MAX OF 124K OF MEMORY. 124K IS
   NEEDED FOR COMPLETE CHECK OF TAG MEMORY.
3. A console terminal
4. A UNIBUS EXERCISER IF NPR DATOS ARE TO BE TESTED.
   HARDWARE SETTINGS: ADDRESS = 770000 (ALL SWITCHES ON)
   VECTOR NOT USED

2.2 SOFTWARE

THIS DIAGNOSTIC WILL RUN UNDER ACT, XDP AND STAND ALONE.
IT CAN ALSO BE RUN UNDER APT IN ACT MODE.

2.3 ACT & APT SETUP

2.3.1 RUN TIMES

FIRST PASS RUN TIME = 10
MAXIMUM PASS RUN TIME = 10

2.3.2 50Hz/60Hz SYSTEM CONFIGURATION

PROPER OPERATION OF FORCE MISS TESTS ARE DEPENDANT ON WHETHER LINE FREQUENCY IS 50Hz OR 60Hz. THE DIAGNOSTIC OPERATES IN THE FOLLOWING MANNER UNDER THE SPECIFIED ENVIRONMENTS:

1. ACT, XDP

   A. QUICK VERIFY, AUTO ACCEPT, XDP CHAIN

      DIAGNOSTIC RUNS FORCE MISS TESTS FOR 60Hz CONFIGURATION ONLY.

   B. DUMP MODE

      1ST PASS OF DIAGNOSTIC FOLLOWING LOADING OF PROGRAM WILL PROMPT USER FOR 60Hz OR 50Hz CONFIGURATION.

2. APT

   A. QUICK VERIFY, RUN TIME, STANDALONE

      APT SCRIPT MUST USE BIT 0 IN SWITCH 1 (CONSOLE SWITCH REGISTERS)

      1 = 50Hz CONFIGURATION
      0 = 60Hz CONFIGURATION

2.4 EXECUTION TIME

FOR AN ERROR FREE, FIRST RUN PASS ON A 11/34 WITH CORE MEMORY.
IT TAKES APPROXIMATELY 10 SECONDS.
3.0 DIAGNOSTIC HIERARCHY PREREQUISITES
-----------------------------

IT IS ASSUMED THAT CPU, MEMORY, MEMORY MANAGEMENT AND TTY ARE WORKING
PROPERLY FOR THIS PROGRAM TO GIVE CORRECT ERROR REPORTS. IF NOT, THEIR RESPECTIVE DIAGNOSTIC SHOULD BE RUN BEFORE THE
CACHE DIAGNOSTIC.

4.0 STARTING ADDRESS
---------------------

200 FOR NORMAL STARTUP

5.0 PROGRAM CONTROL AND OPERATOR ACTION
-----------------------------------------

5.1 THE STANDARD DIAGNOSTIC LOADING PROCEDURES ARE TO BE FOLLOWED.

5.2 LOAD ADDRESS 200

5.3 START

5.4 DIAG. WILL THEN PRINT ITS NAME AND EXPECTED RUN TIME
AFTER WHICH PROGRAM ENTERS COMMAND MODE AND PROMPTS
USER WITH "CACHE=>".

5.5 THE USER THEN HAS THE OPTION OF USING THE FOLLOWING COMMANDS

5.5.1 'LOT' ENTER LOOP ON ERRORING TEST MODE
PROGRAM WILL LOOP ON ANY TEST IN WHICH AN ERROR HAS OCCURRED

5.5.2 'CLOT' (DEFAULT) CANCELS EFFECT OF 'LOT'

5.5.3 'LOE' LOOP ON ERROR
PROGRAM WILL LOOP ON CURRENT ERROR

5.5.4 'CLOE' (DEFAULT) CANCELS EFFECT OF 'LOE'

5.5.5 'MOE' HALT ON ERROR
PROGRAM WILL PRINT TEST NUMBER OF FAILING TEST
THEN HALT. TEST MAY THEN BE REENTERED BY USING
THE CONSOLE CONTINUE SWITCH.

5.5.6 'CHOE' (DEFAULT) CANCELS EFFECT OF 'MOE'

5.5.7 'CER' (DEFAULT) ENABLE ERROR PRINTOUT
ENABLES THE PRINTING OF ERROR MESSAGES.

5.5.8 'IER' DISABLE PRINTING ERROR MESSAGES
5.5.9 'LST XXX' LOOP ON SELECTED TEST
WHERE XXX = TEST TO BE LOOPED ON
EXAMPLE:
LST 121
WILL EXECUTE ALL TESTS BEFORE TEST 121
THEN HOLD AT TEST 121 IN A LOOP
5.5.10 'CLST' (DEFAULT) CANCELS EFFECT OF 'LST XXX'

5.6 AFTERT DESIRED OPTIONS HAVE BEEN SELECTED THE
PROGRAM MAY THEN BE RUN BY TYPING 'RUN'
AFTER WHICH TESTING WILL BEGIN.

5.7 TYPING 'C' AT ANY TIME WILL STOP TESTING
AND RETURN TO COMMAND MODE.

6.0 PROGRAM DESCRIPTION

UPON START OF THE PROGRAM, THE CACHE IS IMMEDIATELY TURND OFF
(FORCE MISS IS ON FOR BOTH HALVES OF CACHE, INTERRUPTS ARE DISABLED
AND CACHE IS IN BYPASS MODE). THE TESTS THEN PROCEED TO SELECTIVELY
TURN ON ONLY THE HALF OF CACHE THAT IS TO BE EXERCISED.
THIS IS TO ENSURE THAT THE INSTRUCTIONS ARE NOT EXECUTED OUT
OF A POSSIBLY BAD CACHE. IN ORDER TO IMPLEMENT THIS SCHEME
THE TESTS THAT ENABLE CACHE ARE RELOCATED TO AREAS OF CACHE
THAT ARE NOT ENABLED. THE TESTS ARE STRUCTURED ON A HALF CACHE
BASIS. THAT IS A TEST MAY BE RUN IN LOW CACHE WHILE TESTING
HIGH CACHE AFTER WHICH AN IDENTICAL TEST WILL RUN IN HIGH CACHE
WHILE TESTING LOW CACHE.

TO FACILITATE THE TESTING OF CACHE, A 2K BUFFER IS RESERVED AT THE
END OF THE PROGRAM FOR READ WRITE OPERATIONS AND RELOCATION OF TESTS.

IMMEDIATELY AFTER THE PROGRAM IS STARTED THE PROGRAM
IDENTIFIES ITSELF AND THEN PROMPTS USER TO ENTER COMMANDS
THAT WILL SET CONDITIONS FOR TESTING (SEE SEC. 5.4).
THIS IS ONLY DONE ON PROGRAM START AND NOT REPEATED
FOR SUBSEQUENT PROGRAM LOOPS.

7.0 ERROR REPORTING

THE CONTENTS OF THE ERROR REPORTS IDENTIFIES THE HARDWARE
UNDER TEST AT THE TIME OF FAILURE. OTHER PERTINENT INFORMATION
SUCH AS CONTENTS OF CACHE CONTROL FIELDS AND FAILING ADDRESS
GOOD DATA, BAD DATA ARE ALSO REPORTED. EACH ERROR REPORT
THAT USES THE (ADDRESS, GOOD DATA, BAD DATA) FORMAT
WILL BE PRECEDED WITH AN EXPLANATION OF WHO'S ADDRESS
AND WHAT DATA IS BEING REPORTED.
IF THE PROGRAM SHOULD HANG OR HALT WITHOUT
PRINTING AN ERROR MESSAGE THE NUMBER OF THE LAST
TEST EXECUTED CAN BE FOUND AT $STESTN; LOCATION 612.
8.0 HANDLERS AND COMMON ROUTINES

8.1 THE FOLLOWING SECTION OFFERS EXPLANATION OF THE
UTILITY ROUTINES USED BY THE PROGRAM. THESE ROUTINES ARE
LOCATED ON THE FIRST 16 PAGES OF THE LISTING

8.1.1 "START:" PREPARES PROGRAM FOR EXECUTION
8.1.2 "PREPARE:" PREPARES 11/34 AND CACHE FOR
EXECUTION OF TESTS
8.1.3 "HALT:" HALT ON ERROR HANDLER
PRINTS HALT ON ERROR MESSAGE, THEN HALTS.
8.1.3 "PRINT:" PRINT PROGRAM TITLE
8.1.4 "LP1:" LOOP ON TEST COMMAND HANDLER
8.1.5 "LP2:" NO LOOP ON TEST COMMAND HANDLER
8.1.6 "LP3:" LOOP ON ERROR COMMAND HANDLER
8.1.7 "LP4:" NO LOOP ON ERROR COMMAND HANDLER
8.1.8 "ML1:" HALT ON ERROR COMMAND HANDLER
8.1.9 "ML2:" NO HALT ON ERROR COMMAND HANDLER
8.1.10 "DIS1:" DISABLE ERROR PRINTOUT COMMAND HANDLER
8.1.11 "DIS2:" ENABLE ERROR PRINTOUT COMMAND HANDLER
8.1.12 "LP5:" LOOP ON SPECIFIED TEST COMMAND HANDLER
8.1.13 "LP6:" DISABLE LOOP ON SPECIFIED TEST COMMAND HANDLER
8.1.14 "P10:" USED TO PRINT TEST I.D. ON CURRENT TEST RUNNING
8.1.15 "TSTFLG:" LOOKS FOR KEYBOARD INPUT
8.1.16 "GETCHA:" INPUTS CHAR. FORM KEYBOARD, PREFORMS
LOWER TO UPPER CASE CONVERSION AND CHECKS FOR SPECIAL
CONDITIONS SUCH AS RUBOUT AND CARRIAGE RETURN, THEN
ENTERS CHAR. INTO INPUT BUFFER.
8.1.17 "ECHO:" ECHO'S CHAR. JUST INPUT FROM KEYBOARD
8.1.18 "TRAP:" TIMEOUT TRAP HANDLER
SETS TRAP FLAG AND RETURNS
8.1.19 "PARITY:" PARITY TRAP HANDLER
DISABLES CACHE, SAVES CONTENTS OF ERROR REGISTER,
CLEARS ERROR REGISTER AND RETURNS.
8.1.20 "ERTSH:" ERROR LOOP HANDLER
HANDLER IS CALLED BY
8.1.21 "LOOPN:" LOOP ON TEST HANDLER
CHECKS LOOP ON TEST MODE FLAG
IF IN LOOP MODE (LOPERR) WILL BE CHECKED TO SEE IF TEST HAD ANY FAILURE
IF IT DID TEST WILL BE RESTARTED.

8.1.22 "DECOD:" PROGRAM COMMAND DECODER
READS COMMAND FROM INPUT BUFFER AND COMPARES IT
TO COMMAND LIST. IF FOUND IN LIST THE SELECTED COMMAND
HANDLER WILL BE ENTERED. IF NOT FOUND THE MESSAGE
"ILLEGAL COMMAND!" WILL BE PRINTED.

8.1.23 "SETE:" PRINT SELECTED ERROR MESSAGE
CALLED BY
JSR RO,SETE
.WORD "D4"
.WORD SENO01
.WORD SENO02
WHERE THE FIRST WORD AFTER THE CALL IS USED AS AN OFFSET
FOR RETURN.
THE SECOND WORD IS THE NUMBER OF THE FIRST SENTENCE TO BE PRINTED
THE THIRD WORD IS THE SECOND SENTENCE TO BE PRINTED.
THERE ARE NO LIMITS ON THE NUMBER OF SENTENCES THAT CAN
BE PRINTED BY THIS CALL.

8.1.24 "RELC:" AND "RELC:" TEST RELLOCATION HANDLERS
THESE HANDLERS ARE USED TO RELOCATE TESTS TO A
HIGH OR LOW CACHE AREA LOCATED AT THE END OF PROGRAM.
AFTER RELLOCATION IS COMPLETE A JUMP WILL BE MADE TO THE RELOCATED
AREA AND TESTING WILL BEGIN.

8.1.25 "GDB:" GOOD BAD DATA PRINTER
PRINTS THE OCTAL CONTENTS OF LOCATIONS (ADD), (GOOD) AND (BAD).

8.1.26 "BITNM:" BIT NAME PRINTER
PRINTS THE NAME OF ANY BIT LEFT SET IN LOCATION (ERROR).

9.0 REV B CHANGES
REASON: 1. ALLOW FORCE MISS TESTS TO OPERATE UNDER VARIOUS CONFIGURATIONS
OF LINE FREQUENCY/MAIN MEMORY SPEED
2. ALLOW CTRL-C TO ABORT ERROR PRINTOUTS AND RETURN
TO SOFTWARE MONITOR.

FOR EACH OF THE FOLLOWING CODING BLOCKS SPECIFIED, CHECK APPROPRIATE
COMMENTS FOR INDICATION OF CHANGES.
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>6952</td>
<td>000000</td>
</tr>
<tr>
<td>6953</td>
<td>000000</td>
</tr>
<tr>
<td>6954</td>
<td>000000</td>
</tr>
<tr>
<td>6955</td>
<td>000000</td>
</tr>
<tr>
<td>6956</td>
<td>000000</td>
</tr>
<tr>
<td>6957</td>
<td>000000</td>
</tr>
<tr>
<td>6958</td>
<td>000100</td>
</tr>
<tr>
<td>6959</td>
<td>000112</td>
</tr>
<tr>
<td>6960</td>
<td>000112</td>
</tr>
<tr>
<td>6961</td>
<td>000116</td>
</tr>
<tr>
<td>6962</td>
<td>000200</td>
</tr>
<tr>
<td>6963</td>
<td>000222</td>
</tr>
<tr>
<td>6964</td>
<td>000240</td>
</tr>
<tr>
<td>6965</td>
<td>000260</td>
</tr>
<tr>
<td>6966</td>
<td>000260</td>
</tr>
<tr>
<td>6967</td>
<td>000320</td>
</tr>
<tr>
<td>6968</td>
<td>000320</td>
</tr>
<tr>
<td>6969</td>
<td>000420</td>
</tr>
<tr>
<td>6970</td>
<td>000440</td>
</tr>
<tr>
<td>6971</td>
<td>000460</td>
</tr>
<tr>
<td>6972</td>
<td>000460</td>
</tr>
<tr>
<td>6973</td>
<td>000520</td>
</tr>
<tr>
<td>6974</td>
<td>000520</td>
</tr>
<tr>
<td>6975</td>
<td>001140</td>
</tr>
<tr>
<td>6976</td>
<td>001140</td>
</tr>
<tr>
<td>6977</td>
<td>001160</td>
</tr>
<tr>
<td>6978</td>
<td>001160</td>
</tr>
<tr>
<td>6979</td>
<td>001760</td>
</tr>
<tr>
<td>6980</td>
<td>002000</td>
</tr>
<tr>
<td>6981</td>
<td>002000</td>
</tr>
<tr>
<td>6982</td>
<td>002000</td>
</tr>
<tr>
<td>6983</td>
<td>012767</td>
</tr>
<tr>
<td>6984</td>
<td>012767</td>
</tr>
<tr>
<td>6985</td>
<td>012767</td>
</tr>
<tr>
<td>6986</td>
<td>020000</td>
</tr>
<tr>
<td>6987</td>
<td>020000</td>
</tr>
<tr>
<td>6988</td>
<td>020000</td>
</tr>
</tbody>
</table>

```
START:
PREPARE:
TRANS:
CMDST:
SMOTE:
SETEN:
DICT1:
FORCE MISS TESTS
```

```
.START:
.END
```
; REV B
BIS $SWREG, CONWRD
TSTB $ENV
BNE 3$
TST $@42
BEQ 2$
BIS $BIT15, CONWRD
JSR PC, PNTNAM
JMP PREPARE
; SET ANY BITS FROM APT LOADING
; IS THIS APT MODE
; YES
; IS THIS DUMP MODE
; YES
; SET HALT ON ERROR MODE
; PRINT TITLE
; START TEST

; REV B
INC $-1
BNE TITLE
JSR PC, LINFOQ
; PROMPT LINE FREQ. ON FIRST PASS ONLY
; PRINT LINE FREQ. PROMPT MESSAGE
; GET ANSWER
BR 4$

; REV B

; TITLE:
JSR PC, PNTNAM
; PRINT TITLE
.SBTL  APT PARAMETER BLOCK

**********************************************************************

;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT

**********************************************************************

;S$X=.; ;SAVE CURRENT LOCATION
.;=24 ; ;SET POWER FAIL TO POINT TO START OF PROGRAM
200 ; ;FOR APT START UP
.;=44 ; ;POINT TO APT INDIRECT ADDRESS PNTR.
$APTHDR ; ;POINT TO APT HEADER BLOCK
.;=$X ; ;RESET LOCATION COUNTER

**********************************************************************

;SET UP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC

INTERFACE SPEC.

**********************************************************************

$APTHD:
$SHBTS: .WORD 0 ; ;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
$MBADDR: .WORD $MAIL ; ;ADDRESS OF APT MAILBOX (BITS 0-15)
$TSTM: .WORD ; ;RUN TIME OF LONGEST TEST
$PSSTM: .WORD ; ;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
$UNITM: .WORD ; ;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT

;WORD $TEND=$MAIL/2 ; ;LENGTH MAILBOX-ETABLE (WORDS)
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000746</td>
<td>000000</td>
<td>$DDW8: .WORD ADDW8 :DEVICE DESCRIPTOR WORD#8</td>
</tr>
<tr>
<td>000750</td>
<td>000000</td>
<td>$DDW9: .WORD ADDW9 :DEVICE DESCRIPTOR WORD#9</td>
</tr>
<tr>
<td>000752</td>
<td>000000</td>
<td>$DDW10: .WORD ADDW10 :DEVICE DESCRIPTOR WORD#10</td>
</tr>
<tr>
<td>000754</td>
<td>000000</td>
<td>$DDW11: .WORD ADDW11 :DEVICE DESCRIPTOR WORD#11</td>
</tr>
<tr>
<td>000756</td>
<td>000000</td>
<td>$DDW12: .WORD ADDW12 :DEVICE DESCRIPTOR WORD#12</td>
</tr>
<tr>
<td>000760</td>
<td>000000</td>
<td>$DDW13: .WORD ADDW13 :DEVICE DESCRIPTOR WORD#13</td>
</tr>
<tr>
<td>000762</td>
<td>000000</td>
<td>$DDW14: .WORD ADDW14 :DEVICE DESCRIPTOR WORD#14</td>
</tr>
<tr>
<td>000764</td>
<td>000000</td>
<td>$DDW15: .WORD ADDW15 :DEVICE DESCRIPTOR WORD#15</td>
</tr>
<tr>
<td>000766</td>
<td></td>
<td>$ETEND:</td>
</tr>
</tbody>
</table>
.SBTL REGISTER DEFINITIONS

CMPE = 177744  ; CACHE MEMORY PARITY ERROR REGISTER
CCR = 177746  ; CACHE CONTROL REGISTER
CMR = 177750  ; CACHE MAINTENANCE REGISTER
CHR = 177752  ; CACHE HIT REGISTER
KRB = 177760  ; KEYBOARD READER STATUS
KBB = 177762  ; KEYBOARD READER BUFFER
PPS = 177764  ; PRINTER PUNCH STATUS
PPB = 177766  ; PRINTER PUNCH BUFFER
PSW = 177776  ; PROCESSOR STATUS WORD

R0 = %0
R1 = %1
R2 = %2
R3 = %3
R4 = %4
R5 = %5
SP = %6
PC = %7

BIT0 = 1
BIT1 = 2
BIT2 = 4
BIT3 = 8
BIT4 = 16
BIT5 = 32
BIT6 = 64
BIT7 = 128
BIT8 = 256
BIT9 = 512
BIT10 = 1024
BIT11 = 2048
BIT12 = 4096
BIT13 = 8192
BIT14 = 16384
BIT15 = 32768

KDD0 = 177546

OFF = 1015

KPDR0 = 172300
KPDR1 = 172302
KPDR2 = 172304
KPDR3 = 172306
KPDR4 = 172308
KPDR5 = 172310
KPDR6 = 172312
KPDR7 = 172314
KPAR0 = 172340
KPAR1 = 172342
KPAR2 = 172344
KPAR3 = 172346
KPAR4 = 172350
KPAR5 = 172352
KPAR6 = 172354
KPAR7 = 172356
SR0 = 177572
BECC = 170002
<table>
<thead>
<tr>
<th>Register</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEBA</td>
<td>170004</td>
<td></td>
</tr>
<tr>
<td>BEDA</td>
<td>170000</td>
<td></td>
</tr>
<tr>
<td>BCR1</td>
<td>170006</td>
<td></td>
</tr>
<tr>
<td>BCR2</td>
<td>170016</td>
<td></td>
</tr>
<tr>
<td>TID</td>
<td>$FESTN</td>
<td></td>
</tr>
</tbody>
</table>
CACHE DIAG.  MACH1  30A(1052)  31-OCT-79  15:29  PAGE 59-14
CPKKB.PT1  25-JUN-79  13:31  REGISTER DEFINITIONS

SEQ 0015

6083  001000
6084  001000  012701  044000  PREPARE:  ==1000
6085  001004  012102  5%:  MOV  (R1)+,R2
6086  001006  020127  050000  CMP  R1,ALOW1
6087  001012  001386  BNE  5%  ;TAGGING COMPLETE
6088  001014  012706  060000  MOV  #60000,SP
6089  001020  005067  003662  CLR  ACTION
6090  001024  005067  177616  CLR  TID
6091  ;REV B  :TEST REST ID

6099  001030  056767  177630  177140  BIS  $SWREG,CONWRD
6099  001034  012737  004650  000114  :SET BITS LOADED FROM APT
6099  001038  012737  000340  000116  MOV  #PARITY,  #114
6099  001042  012737  000340  000116  MOV  #340,  #116
6099  001046  012767  001415  176666  MOV  #OFF+BIT08.CCR
6099  001050  032767  040000  177110  BIT  #114,CONWRD
6100  001066  014035  NO  :IS THIS LOOP ON TEST MODE?
6101  001070  012767  000004  003610  BEQ  15
6102  001074  032767  011000  177072  NO  :IS THIS LOOP ON ERROR MODE?
6103  001078  010104  014035  NO
6104  001086  012767  000002  003572  MOV  #6,ACTION
6105  001094  001114  005576  003610  3%:  SET ACTION TO LOOP ON TEST
6106  001102  010246  :FREE R2 FOR USE
6107  001112  012702  001154  AHALT:  MOV  R2,-(SP)
6108  001116  047627  004732  :PRINT HALT ERROR MESSAGE
6109  001120  047627  002652  JSR  PC.TYPE
6110  001124  056334  :PRINT CURRENT TEST I.D.
6111  001128  012702  001154  MOV  #CRCLF,R2
6112  001132  047627  004716  JSR  PC.TYPE
6113  001136  012602  :ADD <CR>,<LF> TO END OF LINE
6114  001140  000000  MOV  (SP)+,R2
6115  001144  000000  :RESTORE R2
6116  001148  000000  RTS  PC
6117  001152  002007  :ERROR HALT
6118  001156  052114  047440  :RETURN
6119  001160  040510  HLONER:  .ASCII  "HALT ON ERROR AT /
6120  001164  001210  .EVEN
6121  001176  012702  LINFRQ:  MOV  #QUESHZ,R2
6122  001202  004767  004636  JSR  PC.TYPE
6123  001206  000207  RTS  PC
6124  001210  015  012  QUESHZ:  177  .BYTE  15,12,177,177,177
6125  001214  001111  020123  044514  .ASCII  "IS LINE FREQUENCY OF THIS SYSTEM 60HZ?"
6126  001263  015  012  177
6127  001270  044503  020106  047516  .ASCII  "IS TYPE 'Y' OR 'N': *
6128  001323  015  012  177
6129  001330  040250  042500  023440  .BYTE  40,40,40,40,40,40,177,177,177,0
6130  001352  040  040  040
6131  001356  001411  .PRINT TITLE
6132  001360  105767  177272  .NAME:  TSTB $ENV
6133  001364  000207  00042  :IS THIS APT MODE
6134  001370  001411  NO
6135  001372  000207  YES
6136  001374  000042  TST #A42
6137  .EVEN
6138  .EVEN
BEQ $20, $30F
CMP $32, $31F
BNE $20
RTS PC
TST $20
REO $3
MOV #PROM, R2
JSR PC, TYPE
MOV #NAME, R2
JSR PC, TYPE
RTS PC
NAME: .ASCII *CFKKABO PDB 11/34 CACHE DIAGNOSTIC*
.BYTE 15, 12, 177, 177, 177
.ASCIIZ 'EXPECTED RUN TIME APROX. 10 SECONDS'
.BYTE 12, 177, 177
.ASCIIZ 'ENTER 'H' FOR HELP OR 'RUN' TO START DIAGNOSTIC'
.BYTE 12, 177, 177
.ASCIIZ 'FOLLOW ALL COMMANDS WITH A CARRIAGE RETURN'
.BYTE 12, 177, 177
.ASCIIZ 'CACHE=2'
.BYTE 40, 177, 177, 177, 0
.EVEN
BIS #BIT14, CONWRD
RTS PC
BIC #BIT14, CONWRD
RTS PC
BIS #BIT09, CONWRD
RTS PC
BIC #BIT09, CONWRD
RTS PC
BIS #BIT15, CONWRD
RTS PC
BIS #BIT15, CONWRD
RTS PC
BIS #BIT13, CONWRD
RTS PC
BIC #BIT13, CONWRD
RTS PC
BIC #BIT13, CONWRD
RTS PC
CLR R5
LP5: BLO $3
CMPB (R1), #0
BLO 25
BLO 3
INC R1
BLO 5
BR 15
ASL R5
ASL R5
ASL R5
MOVB (R1), R2
BIC $17700, R2
SUB $60, R2
ADD R2, R5
INC R1
CMPB (R1), #0
BLO 6
<table>
<thead>
<tr>
<th>Address</th>
<th>OP Code</th>
<th>Function or Instruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>6195 002120</td>
<td>121127</td>
<td>000070 (\text{CMPB } (R1), #18)</td>
</tr>
<tr>
<td>6196 002124</td>
<td>103760</td>
<td>(\text{BLD } 3)</td>
</tr>
<tr>
<td>6197 002126</td>
<td>052767</td>
<td>000040 (\text{BIS } #100, \text{CONW} )</td>
</tr>
<tr>
<td>6198 002134</td>
<td>110567</td>
<td>176036 (\text{MOV R5, CONW} )</td>
</tr>
<tr>
<td>6199 002140</td>
<td>000207</td>
<td>(\text{RTS PC} )</td>
</tr>
<tr>
<td>6200 002142</td>
<td>062767</td>
<td>000077 (\text{BIC } #100+377, \text{CONW} )</td>
</tr>
<tr>
<td>6201 002150</td>
<td>000207</td>
<td>(\text{RTS PC} )</td>
</tr>
<tr>
<td>6202 002152</td>
<td>012702</td>
<td>002210 (\text{HELP: MOV } #\text{LOST}, R2 )</td>
</tr>
</tbody>
</table>

\(\text{SET LOOP ON TEST MODE} \)
\(\text{EXIT LOOP ON TEST MODE} \)
\(\text{PRINT HELP MESSAGE} \)
6204 002156 004767 003702 JSR PC, TYPE
6205 002162 000207 RTS PC
6206 002164 052767 0000001 176004 FRQ50: BIS #1, CONWRD ; SPECIFY 50HZ
6207 002172 000167 176410 JMP TITLE
6208 002176 042767 000001 175772 FRQ60: BIC #1, CONWRD ; SPECIFY 60HZ
6209 002204 000167 176376 JMP TITLE
6210
6211 002210 047105 042524 020122 LOST:
6212 002307 015 012 177 .ASCII /ENTER ANY OF THE FOLLOWING COMMANDS AFTER THE 'CACHE >' PROMPT/
6213 002314 044042 042517 020042 .BYTE 15, 12, 177, 177, 177
6214 002361 015 012 177 .ASCII *'HOB' = HALT ON ERROR (SW/SWR BIT 15)*
6215 002366 046042 052117 020042 .BYTE 15, 12, 177, 177, 177
6216 002442 015 012 177 .ASCII *'LOE' = LOOP ON FAILING TEST (SW/SWR BIT 14)*
6217 002447 042 042511 021122 .BYTE 15, 12, 177, 177, 177
6218 002595 015 012 177 .ASCII *'FER' = INHIBIT ERROR TYPEOUTS (SW/SWR BIT '3'*)
6219 002532 046042 042517 020042 .BYTE 15, 12, 177, 177, 177
6220 002574 015 012 177 .ASCII *'LOE' LOOP ON ERROR (SW/SWR BIT '9'*)
6221 002601 042 051514 021122 .BYTE 15, 12, 177, 177, 177
6222 002646 015 012 177 .ASCII *'LST' LOOP ON TEST XXX (SW/SWR BIT '8'*)
6223 002653 124 042510 041440 .ASCII *THE COMMAND MUST BE FOLLOWED BY A CARRIDGE RETURN. THE PROGRAM WII...
BYTE 15,12,177,177,177

ASCII *THEN RESPOND ANOTHER PROMPT. THE USER MAY ENTER ANOTHER COMMAND*

BYTE 15,12,177,177,177

ASCII / OR 'RUN' TO START DIAGNOSTIC /

BYTE 15,12,177,177,177

ASCII ANY COMMAND MAY BE CANCELLED BY TYPING 'C' PLUS THE COMMAND

BYTE 15,12,177,177,177

ASCII *EX. CHOE REMOVES HALTING ON ERROR. *

BYTE 15,12,177,177,177

ASCII *TYPING 'C' AT ANY TIME WILL STOP TESTING AND RETURN TO COMMAND MODE.

BYTE 15,12,177,177,177

ASCII WARNING !!! THE HARDWARE SWITCHES ON THE CACHE MODULE MUST BE*

BYTE 15,12,177,177,177

ASCII *IN THE ON POSITION I.E. BOTH SWITCHES FACING TOWARDS THE CONSOLE*

BYTE 15,12,177,177,177

ASCII *NOTE: EACH OF THE ABOVE COMMANDS SETS THE DESIGNATED BIT IN THE*

BYTE 15,12,177,177,177

ASCII *SOFTWARE SWITCH REGISTER LOCATED AT 000176*

BYTE 15,12,177,177,177,0,0,0

.EVEN

PRINT THE CURRENT TID

MOV R2, -(SP)

SAVE REGISTERS R2, R3

MOV R3, -(SP)

PLACE CURRENT TEST I.D. FOR DISEMBERING

MOV TID, R2

SET UP TO GENERATE THE HUNDREDS DIGIT

MOV $7, R3

SUB #100, R2

DIVISION BY SUBTRACTION

BPL 1$

MOV R3, STID+3

ADD DIGIT TO STRING

ADD #100, R2

CORRECT THE REMAINDER OF NUMBER

MOV $7, R3

ADD #100, R2

FIND TENS ASCII DIGIT

SUB #10, R2

FIND NUMBER OF TENS IN NUMBER

MOV R3, STID+4

ADD DIGIT TO NUMBER

ADD #100, R2

CORRECT NUMBER FOR ONES DIGIT

MOV R2, STID+5

ADD DIGIT TO STRING

CORRECT DONES DIGIT

JSR PC TYPE

MOV (SP)+, R3

RESTORE R3

MOV (SP)+, R2

RESTORE R2

RTS PC

ASCII /STID /

BYTE 15,12,11,1,0

.EVEN

.6225 002756 015 012 177
.6226 002763 124 042510 020116
.6227 003062 015 012 177
.6228 003067 117 020122 051047
.6229 003124 015 012 177
.6230 003131 101 054516 041440
.6231 003224 015 012 177
.6232 003231 050 054105 020356
.6233 003275 015 012 177
.6234 003302 054516 044520 043516
.6235 003450 015 012 177
.6236 003457 045027 047122 047111
.6237 003507 015 012 177
.6238 003514 047111 052040 042510
.6239 003614 015 012 177
.6240 003621 116 052117 035105
.6241 003720 015 012 177
.6242 003725 123 043117 053524
.6243 003777 015 012 177
.6244 004010
.6245
.6246
.6247
.6248
.6249
.6250 004010 010246
.6251 004012 010346
.6252 004014 016702 174626
.6253 004020 012703 000057
.6254 004024 067202 000100
.6255 004026 162702 000100
.6256 004032 100374
.6257 004034 110367 000055
.6258 004040 062702 000100
.6259 004044 012703 000057
.6260 004050 050203
.6261 004052 162702 000010
.6262 004056 100374
.6263 004060 110367 000032
.6264 004064 062702 000070
.6265 004070 110267 000023
.6266 004074 012702 004112
.6267 004100 004767 001760
.6268 004104 012603
.6269 004106 012603
.6270 004110 000207
.6271 004112 051524 020124 020040
.6272 004120 015 012 001
.6273 004126
SB9TL KEYBOARD HANDLER

; TEST TO SEE IF THE KEYBOARD WANTS US FOR ANYTHING

TSTLG:   TSTB KRS
         BPL $1
         JSR PC, GETCHA
         RTS PC

; INPUT CHAR, AND TEST FOR SPECIAL CONDITIONS

GETCHA:  MOV R2, -(SP)   ; FREE R2 FOR USE
          MOVB KRS, ABUF PNT
          BJC B #200, ABUF PNT
          CMPB #3, ABUF PNT
          BNE 10$   ; NOT THIS TIME
          JMP START
          CMPB ABUF PNT, #177
          10$:       ; THIS A RUBOUT, WE'Ll IS IT!
ASCII / UNIBUS EXERCISER PRESENT AT ADDRESS 770000/

.BYTE 15, 12, 177, 177, 177, 0

.EVEN
SBTL HIGHEST CACHE ERROR LOOP HANDLER

BIS ERROR, LOPERR
TSTB SENV
BNE 5$ ; IS THIS APPT MODE

JSR PC, TSTFLAG ; LOOK FOR KEYBOARD INPUT

ADD #4, RO ; NORMAL RETURN IF NO ERROR

BIT #1108, CONWRD ; IS THIS LOOP ON TEST MODE
BNE 1$ ; NO

CMPB CONWRD, TID ; IS THIS THE TEST TO LOOP ON
BNE 1$ ; YES

TST -(RO) ; SET LOOP FLAG IF LOOPING
BNE 2$ ; YES

MOV (RO), RO ; SET RETURN ADDRESS

RTS RO ; RETURN

TST ERROR ; ANY ERRORS
BNE 2$ ; NO

SUB ACTION, RO ; TAKE SELECTED ACTION
BR 3$ ;

BIT #81708, CONWRD ; IS THIS LOOP ON TEST MODE
BNE 1$ ; NO

CMPB CONWRD, TID ; IS THIS TEST TO LOOP ON
BNE 2$ ; NO

MOV (RO), RO ; SET RETURN ADDRESS
CLR LOPERR ; RESET LOOP FLAG

RTS RO ; RETURN

TST ACTION, #4 ; IS THIS LOOP ON ERRORING TEST MODE
BNE 2$ ; NO

TST LOPERR ; DID TEST FAIL
BNE 2$ ; YES, RESTART TEST

RTS RO ; RETURN ADDRESS NO LOOP

LDOPERR ; WORD 0
<table>
<thead>
<tr>
<th>Offset</th>
<th>Value 1</th>
<th>Value 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0000</td>
<td>005324</td>
<td>001764</td>
</tr>
<tr>
<td>0x0004</td>
<td>005326</td>
<td>001774</td>
</tr>
<tr>
<td>0x0008</td>
<td>005330</td>
<td>002004</td>
</tr>
<tr>
<td>0x000C</td>
<td>005332</td>
<td>002014</td>
</tr>
<tr>
<td>0x0010</td>
<td>005334</td>
<td>002024</td>
</tr>
<tr>
<td>0x0014</td>
<td>005336</td>
<td>002034</td>
</tr>
<tr>
<td>0x0018</td>
<td>005340</td>
<td>002044</td>
</tr>
<tr>
<td>0x001C</td>
<td>005342</td>
<td>002142</td>
</tr>
<tr>
<td>0x0020</td>
<td>005344</td>
<td>002152</td>
</tr>
</tbody>
</table>

Note: The values are in hexadecimal format.
.SBTL1 ERROR MESSAGE HANDLERS

SETEN:    MOV (RO), LAST ; FIND RETURN ADDRESS
          ADD RO, LAST

JSR PC, TSIFLG ; REV B

TSTB $ENV ; IS THIS APT MODE
 BEQ 11%
 TST ERROR ; ANY ERROR
 BEQ 11% ; Halt
 INC MSGTY ; SET APT ERROR FLAG
 JSR PC, AHALT ; Halt
 BR 1S

BIT #8113, CONNRD ; ERROR PRINT MODE
 BEQ 4S
 MOV LAST, RO ; SO RETURN
 CLR GOODBD ; RESET DATA PRINT FLAG
 CLR BITFLG ; RESET BIT PRINT FLAG
 R1S RO

4$ : TST ERROR ; ANY ERRORS
 BEQ 1%
 JSR PC, PID ; NO SO RETURN
 MOV R1, -(SP) ; PRINT CURRENT TEST ID
 FREE R1 FOR USE
 JSR PC, PRINT ; PRINT SENTENCE
 MOV R0, -(SP) ; ALL SENTENCES PRINTED YET
 CMP RO, LAST
 BEQ 3S ; NO

11S : BIT #81113, CONNRD ; ERROR PRINT MODE
 BEQ 6S
 CLR GOODBD ; IS THIS DATA PRINT MODE
 BEQ 5S ; NO
 JSR PC, GOBD ; ENTER DATA PRINTER

3$ : MOV LAST, RO ; GET THIS BIT PRINT MODE
 BEQ 6S ; NO
 JSR PC, BITNAM ; ENTER BIT PRINTER
 MOV R2, -(SP) ; FREE R2 FOR USE
 ADD LINE FEEDS TO END OF MESSAGE
 JSR PC, TYPE

5$ : BIT #8115, CONNRD ; IS THIS HALT ON ERROR MODE
 BEQ 10S ; NO
 JSR PC, AHALT ; YES!! SO HALT

6$ : MOV (SP), R2 ; RESTORE R2
 MOV (SP), R1 ; RESTORE R1
 BR 1S ; EXIT

10% ; FREE R3 FOR USE
 MOV R3, -(SP) ; STRING POINTER
 MOV #LINE R3
 MOV R2, -(SP) ; SAVE R2
 MOV (R1), R2 ; WORD POINTER
 MOV #RFLF, R2 ; PRINT CARIDGE RETURN, LINE FEED AND FILL
 JSR PC, AWDRD ; ADD WORD TO STRING
 JSR PC, LINE R2 ; ADDRESS OF LINE TO PRINT
 JSR PC, TYPE ; PRINT STRING

PRINT:    MOV R3, -(SP) ; FREE R3 FOR USE
          MOV #LINE R3
          MOV R2, -(SP) ; STRING POINTER
          MOV (R1), R2 ; WORD POINTER
          MOV #RFLF, R2 ; PRINT CARIDGE RETURN, LINE FEED AND FILL
          JSR PC, AWDRD ; ADD WORD TO STRING
          JSR PC, LINE R2 ; ADDRESS OF LINE TO PRINT
          JSR PC, TYPE ; PRINT STRING

5564 005564 010346 005657

PRINT:    MOV R3, -(SP) ; FREE R3 FOR USE
          MOV #LINE R3
          MOV R2, -(SP) ; STRING POINTER
          MOV (R1), R2 ; WORD POINTER
          MOV #RFLF, R2 ; PRINT CARIDGE RETURN, LINE FEED AND FILL
          JSR PC, AWDRD ; ADD WORD TO STRING
          JSR PC, LINE R2 ; ADDRESS OF LINE TO PRINT
          JSR PC, TYPE ; PRINT STRING

5564 005564 010346 005657

PRINT:    MOV R3, -(SP) ; FREE R3 FOR USE
          MOV #LINE R3
          MOV R2, -(SP) ; STRING POINTER
          MOV (R1), R2 ; WORD POINTER
          MOV #RFLF, R2 ; PRINT CARIDGE RETURN, LINE FEED AND FILL
          JSR PC, AWDRD ; ADD WORD TO STRING
          JSR PC, LINE R2 ; ADDRESS OF LINE TO PRINT
          JSR PC, TYPE ; PRINT STRING

5564 005564 010346 005657

PRINT:    MOV R3, -(SP) ; FREE R3 FOR USE
          MOV #LINE R3
          MOV R2, -(SP) ; STRING POINTER
          MOV (R1), R2 ; WORD POINTER
          MOV #RFLF, R2 ; PRINT CARIDGE RETURN, LINE FEED AND FILL
          JSR PC, AWDRD ; ADD WORD TO STRING
          JSR PC, LINE R2 ; ADDRESS OF LINE TO PRINT
          JSR PC, TYPE ; PRINT STRING

5564 005564 010346 005657
CACHE DIAG.  MACY11 30A(1052)  31-OCT-79  15:29  PAGE 62-7
ERROR MESSAGE HANDLERS

6539 005620 012602
6540 005622 012603
6541 005624 000207
6542 005626 004767 000010
1S:  JSR PC,ADWRD
6543 005628 000760
6544 005632 013 012 001 CRLF:
6545
6546
6547
6548
6549 005642 112763 000040 177777 ADWRD:
6550 005650 112293
6551 005652 001376
6552 005654 000207
6553 005656 000
6554 005658 000204
6555 005660 0000
6556
6557
6558
6559 006064 105767 172572 TYPE:
6560 006070 001401
6561 006072 000207
6562 006074 105767 171464
6563 006100 100375
6564 006102 112267 171460
1S:  TSTB $ENV
6565 006106 105762
6566 006110 001365
6567 006112 000207
MOV (SP)+,R2
MOV (SP)+,R3
RTS PC
JSR PC,ADWRD
BR 2S
.BYTE 15,12,1,1,1,0
.EVEN
MOV #40,-1(R3)
MOV (R2)+,(R3)+
BNE 1S
RTS PC
.LINE:
.BYTE 0
.EVEN
.TSTB $ENV
.REQ 1S
.RTS PC
.TSTB PPS
.DPL 1S
.MOV (R2)+,PPB
.TSTB (R2)
.BNE TYPE
.RTS PC
<table>
<thead>
<tr>
<th>Address</th>
<th>Offset</th>
<th>Segment</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6681</td>
<td>006600</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT03/</td>
</tr>
<tr>
<td>6682</td>
<td>006606</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT04/</td>
</tr>
<tr>
<td>6683</td>
<td>006614</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT05/</td>
</tr>
<tr>
<td>6684</td>
<td>006622</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT06/</td>
</tr>
<tr>
<td>6685</td>
<td>006630</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT07/</td>
</tr>
<tr>
<td>6686</td>
<td>006636</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT08/</td>
</tr>
<tr>
<td>6687</td>
<td>006644</td>
<td>044502</td>
<td>030124</td>
<td>ASCIZ /BIT09/</td>
</tr>
<tr>
<td>6688</td>
<td>006652</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT10/</td>
</tr>
<tr>
<td>6689</td>
<td>006660</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT11/</td>
</tr>
<tr>
<td>6690</td>
<td>006666</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT12/</td>
</tr>
<tr>
<td>6691</td>
<td>006674</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT13/</td>
</tr>
<tr>
<td>6692</td>
<td>006702</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT14/</td>
</tr>
<tr>
<td>6693</td>
<td>006710</td>
<td>044502</td>
<td>030524</td>
<td>ASCIZ /BIT15/</td>
</tr>
</tbody>
</table>
ATTEMPT READ INTO CMPE TO TEST ADDRESS SELECT LOGIC
IF TIME OUT OCCURS THEN LOGIC IN ERROR

TST001:

- INC TID
- MOV #OFF,CCR
- MOV #TRAP,4
- MOV #360.6
- MOV #ERTSHI_RO
- CLR ITTRAP
- MOV CMPE,R1
- MOV ITTRAP,ERROR
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +D4
- WORD SEN1
- JSR RO,(RO)
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +S
- LOOP ON TEST

ATTEMPT READ INTO CMPE TO CHECK ADDRESS SELECT LOGIC
IF TIME OUT OCCURS THEN LOGIC IN ERROR

TST002:

- INC TID
- MOV #OFF,CCR
- CLR ITTRAP
- MOV CCR,R1
- MOV ITTRAP,ERROR
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +D4
- WORD SEN1
- JSR RO,(RO)
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +S
- LOOP ON TEST

ATTEMPT READ INTO CMR TO CHECK ADDRESS SELECT LOGIC
IF TIME OCCURS THEN LOGIC IN ERROR

TST003:

- INC TID
- MOV #OFF,CCR
- CLR ITTRAP
- MOV CMR,R1
- MOV ITTRAP,ERROR
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +D4
- WORD SEN1
- JSR RO,(RO)
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +S
- LOOP ON TEST

- WORD SEN1
- JSR RO,(RO)
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +S
- LOOP ON TEST

- WORD SEN1
- JSR RO,(RO)
- JSR RO,SETEN
- PRINT LIST OF SENTENCES
- WORD +S
- LOOP ON TEST
6751 007116 035772
6752 007120 0046010
6753 007122 007062
6754 007124 007062
6755
6756
6757
6758
6759
6760
6761
6762
6763 007126 005267 171514
6764 007132 012767 001015 170606
6765 007140 005067 175374
6766 007144 016701 170602
6767 007150 016767 175364 175364
6768 007156 004067 176170
6769 007162 000004
6770 007164 035722
6771 007166 036010
6772 007170 004010
6773 007172 007132
6774 007174 007132
6775
6776
6777
6778
6779
6780
6781
6782
6783 007176 005267 171444
6784 007202 012767 001015 170536
6785 007210 005067 175324
6786 007214 016701 170522
6787 007220 016767 175314 175314
6788 007226 003567 175310
6789 007232 004067 176114
6790 007236 000004
6791 007240 035722
6792 007242 036026
6793 007244 004010
6794 007246 007192
6795 007250 007202
6796
6797
6798
6799
6800
6801
6802
6803
6804 007252 005267 171370
6805 007256 012767 001015 170462
6806 007264 005067 175250

WORD SEN4
JSR RO,(RO)
WORD 15
WORD 15

READING MAINTENANCE REGISTER CAUSED TIME OUT
TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST

ATTEMPT READ INTO CHR TO CHECK ADDRESS SELECT LOGIC
IF TIME OUT OCCURES THEN LOGIC IN ERROR

1ST004:
INC TID
; UPDATE TEST ID
1S:
MOV #OFF,CCR
; DISABLE CACHE
CLR ITTRAP
; RESET TRAP FLAG
MOV CHR,R1
; READ HIT REGISTER
MOV ITTRAP,ERROR
; SET ERROR FLAG IF TRAP OCCURED
JSR RO,SETEN
; PRINT LIST OF SENTENCES
WORD *D4
WORD SEN1
WORD SEN5
WORD SEN6
JSR RO,(RO)
; TAKE SELECTED ACTION ON ERROR
WORD 15
; LOOP ON ERROR
WORD 15
; LOOP ON TEST

READ ADDRESS BELOW ERROR REGISTER TO CHECK THAT CACHE
WILL NOT RESPOND TO THAT LOCATION

1ST005:
INC TID
; UPDATE TEST ID
1S:
MOV #OFF,CCR
; DISABLE CACHE
CLR ITTRAP
; RESET TRAP FLAG
MOV CMPE-2,R1
; READ ADDRESS BELOW ERROR REGISTER
MOV ITTRAP,ERROR
; ERROR IF NO TRAP
DEC ERROR
JSR RO,SETEN
; PRINT LIST OF SENTENCES
WORD *D4
WORD SEN1
WORD SEN6
JSR RO,(RO)
; TAKE SELECTED ACTION ON ERROR
WORD 15
; LOOP ON ERROR
WORD 15
; LOOP ON TEST

READ LOCATION ABOVE HIT REGISTER TO CHECK THAT CACHE WILL NOT RESPOND

1ST006:
INC TID
; UPDATE TEST ID
1S:
MOV #OFF,CCR
; DISABLE CACHE
CLR ITTRAP
; RESET TRAP FLAG
MOV CHR+, A1
MOV ITTRAP, ERROR
DEC ERROR
JSR RO, SETEN
"WORD "D4"
"WORD SEN1"
"WORD SEN1"
JSR RO, (RO)
"WORD 19"
"WORD 19"

TESTING ADDRESS SELECTION LOGIC BY WRITING ONE INTO UNUSED CMPE REGISTER BITOO THEN READ CONTENTS OF REGISTER BACK LOOKING TO SEE IF BITOO IS SET.

IF BITOO IS SET IT IS POSSIBLE WE ARE ADDRESSING THE WRONG REGISTER

TST007:
INC TID
1S:
MOV #OFF, CCR
DISABLE CACHE
CLR ITTRAP
RESET TRAP FLAG
MOV #1, CMPE
WRITE 1 INTO ERROR REGISTER
MOV CMPE, ERROR
ERROR IF BIT0 SET
BIC #177776, ERROR
DON'T LOOK AT UPPER BITS
JSR RO, SETEN
PRINT LIST OF SENTENCES
"WORD "D4"
"WORD SEN1"
"WORD SEN8"
"WORD SEN10"
JSR RO, (RO)
"WORD 19"
"WORD 19"

TESTING ADDRESS SELECTION LOGIC BY WRITING A ONE INTO CCR BITCC THEN READ REGISTER BACK.

IF BITOO READ AS ZERO THEN IT IS POSSIBLE WE ARE ADDRESSING WRONG REGISTER

TST010:
INC TID
1S:
MOV #OFF, CCR
DISABLE CACHE
CLR ITTRAP
WRITE 1 INTO CONTROL REGISTER
MOV CCR, ERROR
ERROR IF BITOO NOT SET
BIC #177776, ERROR
DEC ERROR
JSR RO, SETEN
PRINT LIST OF SENTENCES
"WORD "D4"
"WORD SEN1"
"WORD SEN10"
WRITING ONE INTO CCR BITCC READ BACK ZERO
<table>
<thead>
<tr>
<th>Address</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6863</td>
<td>007462</td>
<td>036114</td>
<td>.WORD SENC9</td>
</tr>
<tr>
<td>6864</td>
<td>007464</td>
<td>004010</td>
<td>JSR RO, (RO)</td>
</tr>
<tr>
<td>6865</td>
<td>007466</td>
<td>007414</td>
<td>.WORD 1$</td>
</tr>
<tr>
<td>6866</td>
<td>007470</td>
<td>007414</td>
<td>.WORD 1$</td>
</tr>
<tr>
<td>6867</td>
<td></td>
<td></td>
<td>LOOP ON TEST</td>
</tr>
<tr>
<td>6868</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6869</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6870</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
.SBTTL CONTROL REGISTER DATA TEST

WRITE ZERO INTO CCR BIT00 THEN READ CCR
IF CCR IS READ AS ONE THEN CACHE CCR REGISTER MAY BE BAD
OR CACHE REGISTER DATA PATH COULD BE IN ERROR

TST011: INC TID ;UPDATE TEST ID
1$: MOV #OFF+BIT08.CCR;DISABLE CACHE
       BIC #BIT00.CCR;WRITE ZERO TO BIT00
       MOV CCR,ERROR;ERROR IF BIT00 = 1
       MOV #OFF.CCR;DISABLE CACHE
       BIC #BIT00-1,ERROR
       JSR RO.SETEN ;PRINT LIST OF SENTENCES
          .WORD "D4"
          .WORD SEN1 ;CONTROL REGISTER DATA TEST
          .WORD SEN2 ;READ ZERO INTO CCR BIT00 READ BACK ONE
          JSR RO.(RO);TAKE SELECTED ACTION ON ERROR
          .WORD 1$ ;LOOP ON ERROR
          .WORD 1$ ;LOOP ON TEST

WRITE ZERO INTO CCR BIT02 THEN READ CCR
IF BIT02 IS READ AS ONE THEN CCR REGISTER MAY BE BAD
OR CACHE REGISTER DATA PATH MAY BE AT FAULT

TST012: INC TID ;UPDATE TEST ID
          JSR R4.RELCH ;RELOCATE TEST TO HIGH CACHE
          MOV TST013 ;WRITE O INTO BIT02
1$: BIC #BIT02,CCR ;ERROR IF BIT02 = 1
       MOV #0.CCR,ERROR ;DISABLE CACHE
       MOV #OFF.CCR
       BIC #BIT02-1,ERROR
       JSR RO.SETEN ;PRINT LIST OF SENTENCES
          .WORD "D6"
          .WORD SEN1 ;CONTROL REGISTER DATA TEST
          .WORD SEN3 ;WRITE ZERO INTO CCR BIT02 READ ONE
          JSR RO.(RO);TAKE SELECTED ACTION ON ERROR
          .WORD HIGHSP;LOOP ON ERROR
          .WORD HIGHSP;LOOP ON TEST

WRITE ONE INTO CCR BIT02 THEN READ CCR
IF CCR BIT02 READ BACK AS ZERO THEN CCR REGISTER BIT MAY BE BAD
OR CACHE REGISTER DATA PATH MAY BE AT FAULT

TST013: INC TID ;UPDATE TEST ID
1$: BIS #BIT02,CCR ;WRITE 4 INTO CONTROL REGISTER
       MOV #OFF.CCR,ERROR ;ERROR IF BIT02 <> 1
       MOV #OFF.+BIT08.CCR ;DISABLE CACHE
BIC #-BIT02-1, #ERROR
SUB #4, #ERROR
JSR RO, #SETEN
;PRINT LIST OF SENTENCES

;CONTROL REGISTER DATA TEST
WORD #D4
WORD SNTI
WORD SNT14
WROTE ONE INTO CCR BIT02 READ ZERO

28: JSR RO, (RO)
;TAKE SELECTED ACTION ON ERROR

;LOOP ON ERROR
WORD 1$)
;LOOP ON TEST
WORD 1$)

MAYBE:
ASCII /POSSIBLE FORCE MISS SWITCH ERROR!! VERIFY SWITCH POSITIONS/
BYTE 15,12,12,12,177,177,177,0
EVEN

WRITE ZERO INTO CCR BIT03 THEN READ CCR
IF BIT03 READ BACK AS ONE THEN CCR REGISTER BIT MAY BE BAD
OR CACHE REGISTER DATA PATH MAY BE AT FAULT

INC TID
;UPDATE TEST ID
6952 010034 004467 176054  JSR R4.RELCTRL  ;RELOCATE TEST TO LOW CACHE
6953 010040 010114  .WORD TST015
6956 010042 042737 000010 177746 1S:  BIC #BIT03,#CCR  ;WRITE 0 TO BIT03
6955 010050 013737 177746 004542  MOV #CCR,#ERROR  ;ERROR IF BIT = 1
6956 010056 012737 001015 177746  MOV #OFF,#CCR  ;DISABLE CACHE
6957 010064 042737 177767 004542  BIC #BIT03-1,#ERROR
6958 010072 004037 005352  JSR RO,#SETEN  ;PRINT LIST OF SENTENCES
6959 010076 000006  .WORD *D6
6960 010100 036150  .WORD SENT11  ;CONTROL REGISTER DATA TEST
6961 010102 036244  .WORD SENT15  ;WROTE ZERO INTO CCR BIT03 READ ONE
6962 010106 042166  .WORD SENT168
6963 010106 004010  JSR RO,(RO)  ;TAKE SELECTED ACTION ON ERROR
6964 010110 044000  .WORD LOWSP  ;LOOP ON ERROR
6965 010112 044000  .WORD LOWSP  ;LOOP ON TEST
WRITE ZERO INTO CCR BIT07 THEN READ CCR
IF CCR BIT07 READ AS ONE THEN CCR REGISTER BIT MAY BE BAD
OR CACHE REGISTER DATA PATH MAY BE AT FAULT

WRITE ONE INTO CCR BIT07 THEN READ CCR
IF CCR BIT07 READ AS ZERO THEN CCR REGISTER BIT MAY BE BAD
OR CACHE REGISTER DATA PATH MAY BE AT FAULT
MOV #OFF,CCR ;DISABLE CACHE
BIC #BIT10-1,ERROR ;MASK BIT10
SUB #BIT10,ERROR
JSR R0,SETEN ;REPORT ERROR IF ANY
JSR R0,SETEN ;CONTROL REGISTER DATA TEST
WORD #04
WORD SEN11
WORD SEN24
JSR R0,(RO) ;WRITE ONE INTO CCR BIT10 READ ZERO
JSR R0,(RO) ;TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST
WRITE INTO UNUSED CCR REGISTER BIT01 THEN READ CCR

IF CCR BIT01 READ AS ONE THEN CACHE DATA PATH ERROR

ST026:

INC TID : UPDATE TEST ID
MOV #OFF.CCR : DISABLE CACHE
BIS #0,T01.CCR : WRITE 1 INTO CONTROL REGISTER BIT01
MOV CCR,ERROR : ERROR IF BIT01 = 1
BIC #~BIT01-1,ERROR
JSR RO,SETEN

WORD 4D
WORD $25
JSR RO,(RD)
WORD 1S
LOOP ON ERROR

WRITE ONE INTO UNUSED CCR BIT04 THEN READ CCR

IF CCR BIT04 READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

ST027:

INC TID : UPDATE TEST ID
MOV #OFF.CCR : DISABLE CACHE
BIS #BIT04.CCR : WRITE 1 INTO CONTROL REGISTER BIT04
MOV CCR,ERROR : ERROR IF BIT04 = 1
BIC #~BIT04-1,ERROR
JSR RO,SETEN

WORD 4D
WORD $25
JSR RO,(RD)
WORD 1S
LOOP ON ERROR

WRITE ONE INTO UNUSED CCR BIT05 THEN READ CCR

IF CCR BIT05 READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

ST030:

INC TID : UPDATE TEST ID
MOV #OFF.CCR : DISABLE CACHE
BIS #BIT05.CCR : WRITE 1 INTO CONTROL REGISTER BIT05
MOV CCR,ERROR : ERROR IF BIT05 = 1
BIC #~BIT05-1,ERROR
JSR RO,SETEN

WORD 4D
WORD $25
JSR RO,(RD)
WORD 1S
LOOP ON ERROR
WRITE ONE INTO UNUSED CCR BIT0B THEN READ CCR
IF CCR BIT0B READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

ST031:
1S:
INC TID
UPDATE TEST ID
MOV IOFF,CCR
DISABLE Cache
BIS #BIT0B,CCR
WRITE 1 INTO CONTROL REGISTER BIT0B
MOV CCR,ERROR
ERROR IF BIT0B = 1
BIC #BIT0B-1,ERROR
JSR RO,SETEN
PRINT LIST OF SENTENCES
WORD *D4
JSR RO,(RO)
CONTROL REGISTER UNUSED BIT TEST
WORD SEN25
READ ONE FROM UNUSED CCR BIT0B
INC TID
UPDATE TEST ID
WRITE ONE INTO UNUSED CCR BIT11 THEN READ CCR
IF CCR BIT11 READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

ST032:
1S:
MOV IOFF,CCR
DISABLE Cache
BIS #BIT11,CCR
WRITE 1 INTO CONTROL REGISTER BIT11
MOV CCR,ERROR
ERROR IF BIT11 = 1
BIC #BIT11-1,ERROR
JSR RO,SETEN
PRINT LIST OF SENTENCES
WORD *D4
JSR RO,(RO)
CONTROL REGISTER UNUSED BIT TEST
WORD SEN25
READ ONE FROM UNUSED CCR BIT11
WORD SEN30
READ ONE FROM UNUSED CCR BIT11
JSR RO,(RO)
CONTROL REGISTER UNUSED BIT TEST
WORD SEN25
READ ONE FROM UNUSED CCR BIT11
INC TID
UPDATE TEST ID
WRITE ONE INTO UNUSED CCR BIT14 THEN READ CCR
IF CCR BIT14 READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

ST033:
1S:
MOV IOFF,CCR
DISABLE Cache
BIS #BIT14,CCR
WRITE 1 INTO CONTROL REGISTER BIT14
MOV CCR,ERROR
ERROR IF BIT14 = 1
BIC #BIT14-1,ERROR
JSR RO,SETEN
PRINT LIST OF SENTENCES
CONTROL REGISTER UNUSED BIT TEST

```
7264 011320 000004
7265 011322 036504
7266 011324 036626
7267 011326 004010
7268 011400 011334
7269 011402 011334

; CONTROL REGISTER UNUSED BIT TEST
; READ ONE FROM UNUSED CCR BIT 16
; TAKE SELECTED ACTION ON ERROR
; LOOP ON ERROR
; LOOP ON TEST

; WRITE ONE INTO UNUSED CCR BIT 15 THEN READ CCR
; IF CCR BIT 15 READ AS ONE THEN CACHE REGISTER DATA PATH ERROR

; INC TID
; MOV #OFF, CCR
; BIS #BIT15, CCR
; MOV CCR, ERROR
; BIC #BIT15-1, ERROR
; JSR RO, SETEN
```

; PRINT LIST OF SENTENCES
;

; WORD 04
; WORD 025
; CONTROL REGISTER UNUSED BIT TEST
; TAKE SELECTED ACTION ON ERROR
; LOOP ON ERROR
; LOOP ON TEST

; WORD 15
; WORD 15
; LOOP ON TEST
SBTTL CACHE CONTROL REGISTER BYTE TESTS

REGISTER BYTE SELECTION LOGIC TEST

WRITE ZERO INTO LOW BYTE WRITE ONE INTO HIGH BYTE

VERIFY THAT LOW BYTE DATA IS NOT EFFECTED BY WRITE TO HIGH BYTE

; STT035:
; INC TID:
; $:
; MOV OFF,CCR
; DISABLE CACHE

; BICB #B701,CCR
; WRITE 0 INTO CONTROL REGISTER BIT01

; BISB #B701,CCR+1
; WRITE 1 INTO CONTROL REGISTER BIT09

; MOV CCR,ERROR
; ERROR IF BIT01 = 1

; BIC #=-B701+B09>1,ERROR
; OR BIT09 = 0

; SUB #1000,ERROR

; JR R0,SETEN

; PRINT LIST OF SENTENCES

; .WORD +DB

; .WORD SEN33
; CACHE CONTROL REGISTER BYTE TESTS

; .WORD SEN36
; WRITE ZERO INTO LOW BYTE BIT01

; .WORD SEN35
; WRITE ONE INTO HIGH BYTE BIT09

; .WORD SEN36
; READ ZERO FROM BIT09

; JSR R0,(RO)
; TAKE SELECTED ACTION ON ERROR

; .WORD +IS
; LOOP ON ERROR

; .WORD +IS
; LOOP ON TEST

; WRITE ZERO INTO HIGH BYTE WRITE ONE INTO LOW BYTE

; VERIFY HIGH BYTE NOT EFFECTED BY WRITE INTO LOW BYTE

; STT036:
; INC TID:
; $:
; MOV OFF,CCR
; DISABLE CACHE

; BICB #B702,CCR+1
; WRITE 0 INTO CONTROL REGISTER BIT10

; BISB #B702,CCR
; WRITE 1 INTO CONTROL REGISTER BIT12

; MOV CCR,ERROR
; ERROR IF BIT01 = 0 OR IF BIT09 = 1

; BIC #=-B702-1,ERROR
; SUB #B702,ERROR

; JSR R0,SETEN

; PRINT LIST OF SENTENCES

; .WORD +DB

; .WORD SEN33
; WRITE ZERO INTO HIGH BYTE BIT10

; .WORD SEN37
; WRITE ONE INTO LOW BYTE BIT02

; .WORD SEN38
; READ ZERO FROM BIT02 OR READ ONE FROM BIT10

; JSR R0,(RO)
; TAKE SELECTED ACTION ON ERROR

; .WORD +IS
; LOOP ON ERROR

; .WORD +IS
; LOOP ON TEST
SBTTL MAINTENANCE REGISTER DATA TEST

:STO37:

; VERIFY CMR BIT00 CAN BE WRITTEN TO A ONE

:STO38:

; VERIFY CMR BIT00 CAN BE WRITTEN TO A ZERO

:STO41:

; ATTEMPT TO WRITE ONE INTO UNUSED MAINT REGISTER BIT01

; READ INTO CMR SHOULD RESULT IN ZERO FROM BIT01
ATTEMPT TO WRITE ONE INTO UNUSED MAINT. REGISTER BIT02
READ INTO CMR SHOULD RESULT IN ZERO FROM BIT02

TST042: INC TID ;UPDATE TEST ID
1$: BIS #BIT02,CMR ;WRITE 1 INTO MAINTENANCE REGISTER BIT02
MOV CMR,ERROR ;ERROR IF BIT02 = 1
BIC #-BIT02-1,ERROR
JSR RO.SETEN ;PRINT LIST OF SENTENCES
WORD *D4

WORD SEN40 ;MAINTENANCE REGISTER DATA TEST
WORD SEN44 ;READ ONE FROM UNUSED CMR BIT02
JSR RO,(RO) ;TAKE SELECTED ACTION ON ERROR

WORD 1$ ;LOOP ON ERROR
WORD 1$ ;LOOP ON TEST
SBTTL DATA PATH TEST

; TEST WRITES ALL ONES TO HIGH CACHE THEN READS
; SAME LOCATIONS, IF READ WAS A HIT FORM CACHE
; THEN TEST DATA FOR ALL ONES, ANY BIT READ AS ZERO
; CAUSES ERROR REPORT.

TSTO43: INC TID : UPDATE TEST ID
JSR R4, RELCTL : RELOCATE TEST TO LOW CACHE
.word TSTO44
MOV #HIGHSR, R1 : POINT TO HIGH CACHE
MOV #5, @CCR : ENABLE HIGH CACHE
1$: MOV (R1)+, R2 : TAG ALL HIGH CACHE LOCATIONS
CMP R1, #HIGHSR+2000
BNE 1$

2$: MOV R2, -(R1) : DATA FOR TEST
CMP R1, #HIGHSR
BNE 2$
CLR @ERROR
CLR R5
3$: MOV (R1)+, R2 : READ DATA
BIT #BIT03, @CHR
BEQ 4$
INC R3

4$: MOV #OFF, @CCR
TST R5
BNE 5$
MOV #HIGHSR+2000
BNE 3$

5$: CLR @ERROR
INC #BITFLG
JSR R0, @SETEN
.word #6
.word SEN45
3$: DATA PATH TEST
.word SEN46
3$: WRITE ALL ONES TO HIGH CACHE
.word SENK7
3$: NO HITS ON DATA READ

6$: JSR RO, (R0)
.word LOWSP
3$: TAKE SELECTED ACTION ON ERROR
.word LOWSP
3$: LOOP ON TEST

WRITE ALL ZEROS INTO LOW CACHE, READ AND VERIFY
; ZEROS READ FORM CACHE, IF READ HIT AND BIT READ AS ONE THEN ERROR
TSTO44: INC TID : UPDATE TEST ID
JSR R4, RELCTL : RELOCATE TEST TO LOW CACHE
CACHE DIAG.  MARY1 30A(1052)  31-OCT-79  15:29  PAGE 65-11
(FKKB1.P11)  25-JUN-79  13:31  DATA PATH TEST

7481  012320  012502
7482  012322  012701  060000
7483  012326  012737  000005  177746
7484  012334  012102
7485  012336  020127  050000
7486  012342  001374
7487  012344  005002
7488  012346  010241
7489  012350  020127  046000
7490  012354  001374
7491  012356  005037  004542
7492  012362  005037
7493  012364  012102
7494  012366  032737  000004  177744
7495  012374  001404
7496  012376  000203
7497  012400  005102
7498  012402  005037  004542
7499  012406  020127  050000
7500  012412  001364
7501  012414  012737  001015  177746
7502  012422  005137  004542
7503  012426  005703
7504  012430  010111
7505  012432  005237  004542
7506  012436  004037  005352
7507  012442  000006
7508  012444  037126
7509  012446  037204
7510  012450  037154
7511  012452  000410
7512  012456  005237  004560
7513  012460  004037  005352
7514  012464  000006
7515  012466  037126
7516  012470  037204
7517  012472  037222
7518  012474  004010
7519  012476  044000
7520  012500  044000

WORD TST04S
MOV #HIGHSP,R7
MOV $5,#CCR

1$: MOV (R1)+,R2
CMP R1,#HIGHSP+2000
BNE 19
CLR R2

2$: MOV R2,-(R1)
CMP R1,#HIGHSP
BNE 29
CLR #ERROR

3$: MOV (R1)+,R2
BIT #BIT02,#CMPE
BEQ 49
INC R3

4$: CMP R1,#HIGHSP+2000
BNE 38
TST R3

5$: INC #BITFLG
JSR RO,#SETEN

WORD 46

6$: JSR RO,(RO)
WORD LOWSP
WORD LOWSP

DATA PATH TEST
POINT TO HIGH CACHE
ENABLE HIGH CACHE
TAG ALL HIGH CACHE LOCATIONS
DATA FOR TEST
WRITE ALL ZEROS TO HIGH CACHE
DATA READ FROM CACHE .FLAG
READ DATA
VERIFY DATA READ FROM CACHE
DATA READ FROM CACHE INDICATOR
OR ALL READ DATA
DISABLE CACHE
SEE IF ANY DATA READ FROM CACHE
ERROR FLAG
PRINT LIST OF SENTENCES
DATA PATH TEST
WRITE ALL ZEROS TO HIGH CACHE
NO HITS ON DATA READ
PRINT ERRORING BIT(S)
DATA PATH TEST
WRITE ALL ZEROS TO HIGH CACHE
DATA BIT(S) READ AS ONE
TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST
CACHE DIAG: MACY11 30A(1052) 31-OCT-79 15:29 PAGE 65-12
CKK11 P11 25-JUN-79 13:31 DATA PATH BIT SHORT TEST

SBTTL DATA PATH BIT SHORT TEST

ROTATE A ONE ACROSS THE DATA PATH TO VERIFY THAT EACH BIT CAN BE WRITTEN TO A ONE INDIVIDUALLY.

TST045: INC TID :UPDATE TEST ID
CLR ERROR :RESET ERROR FLAG
JSR R4,RELCTH :RELOCATE TEST TO HIGH CACHE
.WORD TST046
MOV @R1, R3 :DATA FOR TEST
MOV #11, @MCCR :ENABLE LOW CACHE
MOV @L0SP,R2 :TAG LOW CACHE BLOCK #2
MOV @L0SP,R2 :TAG LOW CACHE LOCATION
MOV R3,#L0SP :WRITE DATA
CLR @CME :RESET ERROR REGISTER
MOV @L0SP,R2 :READ DATA
BIT #BIT05,#CHR :VERIFY A HIT
BNE 2$ :NO HIT
CMP R2, R3 :VERIFY DATA
BNE 7$ :BAD DATA

10$: ASL R3
11$: BNE 1$

MOV #OFF, @MCCR :DISABLE CACHE
BR TST046

2$: BIS #BIT07,#MCCR :ENABLE ABORT FOR ERROR READ
MOV @M0PE,R5 :SAVE ERROR IN R5
BIT #BIT05,R5 :DISABLE ABORT
BIT #BIT05,R5 :ANY TAG FAILURE
BEQ 3$
MOV @LOW1,R2 :TAG LOW BLOCK #2
BIS #BIT10,#MCCR :WRITE WRONG TAG
MOV @L0SP,R2 :WRITE WRONG TAG
BIT #BIT05,R5 :ANY LOW BYTE ERROR
BEQ 4$

3$: BIT #BIT06,R5 :SET WRITE WRONG DATA
MOV R3,#L0SP :WRITE LOW BYTE
BIT #BIT07,#MCCR :DISABLE WRITE WRONG DATA
BIT #BIT05,R5 :DID BYTE FAIL
BEQ 5$
BIS #BIT06,#MCCR :ENABLE WRITE WRONG DATA

4$: MOV #BIT06,#MCCR :SET WRITE WRONG DATA
MOV R3,#L0SP :WRITE LOW BYTE
BIT #BIT07,#MCCR :DISABLE WRITE WRONG DATA
BIT #BIT05,R5 :DID BYTE FAIL
BEQ 6$

5$: SWAB R3 :POS DATA FOR MOVB
MOV R3,#L0SP :WRITE HIGH BYTE
SWAB R3 :RESTORE DATA
BIT #BIT06,#BIT10,#MCCR :DISABLE WRITE WRONG
MOV @L0SP,R2 :READ DATA
MOV PLC,R5 :CORRECT WRONG PARITY

6$: CMP R3,#3$-TST045-16+HIGHSP
BNE 6$

7$: MOV #OFF,#MCCR :DISABLE CACHE
MOV #1,#MERROR :SET ERROR FLAG
MOV R3,#GOOD :GOOD DATA
CLR @ADD :CACHE ADDRESS
MOV R2,#BAD :BAD DATA
INC @GOODR :BIT PRINT MODE
<table>
<thead>
<tr>
<th>Instruction</th>
<th>Address</th>
<th>Data 1</th>
<th>Data 2</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIS #BIT07, ACCR</td>
<td>7685 013500</td>
<td>00200</td>
<td>177746</td>
<td>ENABLE ABORT FOR ERROR READ</td>
</tr>
<tr>
<td>MOV A,CMPE,R5</td>
<td>7686 013506</td>
<td>00200</td>
<td>177746</td>
<td>SAVE ERROR IN R5</td>
</tr>
<tr>
<td>BIC #BIT07, ACCR</td>
<td>7687 013512</td>
<td>00200</td>
<td>177746</td>
<td>DISABLE ABORT</td>
</tr>
<tr>
<td>BIT #BIT05,R5</td>
<td>7688 013520</td>
<td>000040</td>
<td>177746</td>
<td>ANY TAG FAILURE</td>
</tr>
<tr>
<td>BEQ 3$</td>
<td>7689 013524</td>
<td>000040</td>
<td>177746</td>
<td>NO</td>
</tr>
<tr>
<td>MOV A,LOW1,R2</td>
<td>7690 013526</td>
<td>00200</td>
<td>177746</td>
<td>TAG LOW BLOCK #2</td>
</tr>
<tr>
<td>BIS #BIT10, ACCR</td>
<td>7691 013532</td>
<td>00200</td>
<td>177746</td>
<td>WRITE WRONG TAG</td>
</tr>
<tr>
<td>MOV A,LMOSP,R2</td>
<td>7692 013540</td>
<td>00400</td>
<td>177746</td>
<td>WRITE WRONG DATA</td>
</tr>
<tr>
<td>BIT #BIT06,R5</td>
<td>7693 013544</td>
<td>000100</td>
<td>177746</td>
<td>ANY LOW BYTE ERRORS</td>
</tr>
<tr>
<td>BEQ 4$</td>
<td>7694 013550</td>
<td>001403</td>
<td>177746</td>
<td>NO</td>
</tr>
<tr>
<td>BIS #BIT06, ACCR</td>
<td>7695 013552</td>
<td>000100</td>
<td>177746</td>
<td>WRITE WRONG DATA</td>
</tr>
<tr>
<td>MOV B,R5,LMOSP</td>
<td>7696 013560</td>
<td>110337</td>
<td>040000</td>
<td>WRITE LOW BYTE</td>
</tr>
<tr>
<td>BIS #BIT06, ACCR</td>
<td>7697 013564</td>
<td>000100</td>
<td>177746</td>
<td>DISABLE WRITE WRONG DATA</td>
</tr>
<tr>
<td>BIT #BIT07,R5</td>
<td>7698 013572</td>
<td>00200</td>
<td>177746</td>
<td>DID HIGH BYTE FAIL</td>
</tr>
<tr>
<td>BEQ 5$</td>
<td>7699 013576</td>
<td>001603</td>
<td>177746</td>
<td>NO</td>
</tr>
<tr>
<td>BIS #BIT06, ACCR</td>
<td>7700 013600</td>
<td>000100</td>
<td>177746</td>
<td>ENABLE WRITE WRONG DATA</td>
</tr>
<tr>
<td>MOV B,R5,LMOSP+1</td>
<td>7701 013606</td>
<td>000300</td>
<td>177746</td>
<td>POS. DATA FOR HIGH WRITE</td>
</tr>
<tr>
<td>BIC #BIT06,BIT10, ACCR</td>
<td>7702 013610</td>
<td>110337</td>
<td>040001</td>
<td>WRITE HIGH BYTE</td>
</tr>
<tr>
<td>MOV B,R5,LMOSP+1</td>
<td>7703 013614</td>
<td>002100</td>
<td>177746</td>
<td>DISABLE WRITE WRONG</td>
</tr>
<tr>
<td>MOV B,R5,LMOSP</td>
<td>7704 013622</td>
<td>000300</td>
<td>177746</td>
<td>READ DATA FROM CACHE</td>
</tr>
<tr>
<td>MOV B,R5,LMOSP,R2</td>
<td>7705 013624</td>
<td>000300</td>
<td>177746</td>
<td>CORRECT WRONG WRITTEN PARITY</td>
</tr>
<tr>
<td>MOV PC,R5</td>
<td>7706 013630</td>
<td>010705</td>
<td>040000</td>
<td>CAUSE WRITE TAG AND DATA</td>
</tr>
<tr>
<td>MOV -1(R5), (R5)</td>
<td>7707 013632</td>
<td>015151</td>
<td>044140</td>
<td>CLEAR DATA</td>
</tr>
<tr>
<td>CMP R5, #32-TS047-16+LMOSP</td>
<td>7708 013634</td>
<td>020527</td>
<td>044140</td>
<td>VERIFY DATA</td>
</tr>
<tr>
<td>BEQ 10$</td>
<td>7709 013640</td>
<td>001374</td>
<td>044140</td>
<td>VERIFY DATA</td>
</tr>
<tr>
<td>CMP R2,R3</td>
<td>7710 013642</td>
<td>020203</td>
<td>044140</td>
<td>VERIFY DATA</td>
</tr>
<tr>
<td>BEQ 10$</td>
<td>7711 013644</td>
<td>001074</td>
<td>044140</td>
<td>VERIFY DATA</td>
</tr>
<tr>
<td>MOV #OFF, ACCR</td>
<td>7712 013646</td>
<td>012737</td>
<td>00215</td>
<td>177746</td>
</tr>
<tr>
<td>CLR BADD</td>
<td>7713 013654</td>
<td>005037</td>
<td>00544</td>
<td>CACHE ADDRESS</td>
</tr>
<tr>
<td>MOV #1, #ERROR</td>
<td>7714 013660</td>
<td>012737</td>
<td>000001</td>
<td>004542</td>
</tr>
<tr>
<td>MOV R3, #GOOD</td>
<td>7715 013666</td>
<td>010550</td>
<td>04552</td>
<td>GOOD DATA</td>
</tr>
<tr>
<td>MOV R2, #BAD</td>
<td>7716 013672</td>
<td>010237</td>
<td>04552</td>
<td>BAD DATA</td>
</tr>
<tr>
<td>INC #GOODBR</td>
<td>7717 013676</td>
<td>005237</td>
<td>04552</td>
<td>SET DATA PRINT MODE</td>
</tr>
<tr>
<td>JSR R0, #SETEN</td>
<td>7718 013679</td>
<td>060307</td>
<td>05352</td>
<td>REPORT ERROR</td>
</tr>
<tr>
<td>.WORD &quot;66&quot;</td>
<td>7719 013680</td>
<td>000006</td>
<td>04400</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>.WORD SEN51</td>
<td>7720 013682</td>
<td>032556</td>
<td>046004</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>.WORD SEN55</td>
<td>7721 013684</td>
<td>032556</td>
<td>046004</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>.WORD SEN53</td>
<td>7722 013686</td>
<td>032556</td>
<td>046004</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>JSR R0, #RO</td>
<td>7723 013688</td>
<td>004010</td>
<td>046004</td>
<td>TAKE SELECTED ACTION ON ERROR</td>
</tr>
<tr>
<td>.WORD 1S-TS047-16+LMOSP</td>
<td>7724 013690</td>
<td>046004</td>
<td>046004</td>
<td>LOOP ON ERROR</td>
</tr>
<tr>
<td>.WORD 10S-TS047-16+LMOSP</td>
<td>7725 013692</td>
<td>046004</td>
<td>046004</td>
<td>LOOP ON ERROR</td>
</tr>
</tbody>
</table>

**#STO50:**

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Address</th>
<th>Data 1</th>
<th>Data 2</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INC TID</td>
<td>7726 013694</td>
<td>005267</td>
<td>164716</td>
<td>UPDATE TEST ID</td>
</tr>
<tr>
<td>CLR ERROR</td>
<td>7727 013696</td>
<td>005067</td>
<td>17060</td>
<td>RESET ERROR FLAG</td>
</tr>
<tr>
<td>JSR R4, RELTIL</td>
<td>7728 013698</td>
<td>004467</td>
<td>172154</td>
<td>RELocate TO LOW CACHE</td>
</tr>
<tr>
<td>JSR R4, #TS051</td>
<td>7729 01369A</td>
<td>014342</td>
<td>177776</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>MOV #177776,R3</td>
<td>772A 01369C</td>
<td>012737</td>
<td>0000005</td>
<td>177746</td>
</tr>
<tr>
<td>MOV #5, ACCR</td>
<td>772B 01369E</td>
<td>012737</td>
<td>0000005</td>
<td>177746</td>
</tr>
<tr>
<td>MOV #HIG1,R2</td>
<td>772C 0136A0</td>
<td>015302</td>
<td>052000</td>
<td>TAG HIGH BLOCK #2</td>
</tr>
<tr>
<td>MOV #HIG2,R2</td>
<td>772D 0136A2</td>
<td>015302</td>
<td>052000</td>
<td>TAG HIGH BLOCK #1</td>
</tr>
<tr>
<td>MOV R3, #HIGSP</td>
<td>772E 0136A4</td>
<td>010337</td>
<td>046000</td>
<td>WRITE TEST DATA INTO CACHE</td>
</tr>
</tbody>
</table>

**#STO50:**

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Address</th>
<th>Data 1</th>
<th>Data 2</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INC TID</td>
<td>772F 0136A6</td>
<td>005267</td>
<td>164716</td>
<td>UPDATE TEST ID</td>
</tr>
<tr>
<td>CLR ERROR</td>
<td>7730 0136A8</td>
<td>005067</td>
<td>17060</td>
<td>RESET ERROR FLAG</td>
</tr>
<tr>
<td>JSR R4, RELTIL</td>
<td>7731 0136AA</td>
<td>004467</td>
<td>172154</td>
<td>RELocate TO LOW CACHE</td>
</tr>
<tr>
<td>JSR R4, #TS051</td>
<td>7732 0136AC</td>
<td>014342</td>
<td>177776</td>
<td>DATA PATH BIT SHORT TEST</td>
</tr>
<tr>
<td>MOV #177776,R3</td>
<td>7733 0136AE</td>
<td>012737</td>
<td>0000005</td>
<td>177746</td>
</tr>
<tr>
<td>MOV #5, ACCR</td>
<td>7734 0136B0</td>
<td>012737</td>
<td>0000005</td>
<td>177746</td>
</tr>
<tr>
<td>MOV #HIG1,R2</td>
<td>7735 0136B2</td>
<td>015302</td>
<td>052000</td>
<td>TAG HIGH BLOCK #2</td>
</tr>
<tr>
<td>MOV #HIG2,R2</td>
<td>7736 0136B4</td>
<td>015302</td>
<td>052000</td>
<td>TAG HIGH BLOCK #1</td>
</tr>
<tr>
<td>MOV R3, #HIGSP</td>
<td>7737 0136B6</td>
<td>010337</td>
<td>046000</td>
<td>WRITE TEST DATA INTO CACHE</td>
</tr>
</tbody>
</table>
SIBTTL CACHE FLUSH TESTS

VERIFY FLUSH IN PROGRESS BIT WILL SET AS RESULT OF FLUSH

ST051:
INC TID
UPDATE TEST ID
BIS &BIT0B,CCR
CAUSE CACHE FLUSH
BIT &BIT12,CCR
VERIFY FLUSH IN PROGRESS
BEQ 1S
CLR ERROR
RESET ERROR FLAG
3S: CLR R2
WAIT FOR FLUSH TO COMPLETE
2S: DEC R2
WAIT LOOP
BNE 2S
JSR RO,(RO)
TAKE SELECTED ACTION ON ERROR
WORD 4S
LOOP ON ERROR
WORD 4S
LOOP ON TEST
CLR LOPERR
RESET LOOP ON TEST FLAG
BR ST052
NEXT TEST

VERIFY FLUSH IN PROGRESS BIT WILL RESET ON COMPLETION OF FLUSH.

ST052:
INC TID
UPDATE TEST ID
CLR R2
RESET DELAY COUNTER
CLR ERROR
RESET ERROR FLAG
BIS &BIT0B,CCR
START FLUSH CYCLE
BIT &BIT12,CCR
SEE IF FLUSH COMPLETE
BEQ 1S
FLUSH COMPLETE
DEC R2
SEE IF TIME HAS RUN OUT
BNE 2S
NOT YET
MOV #1,ERROR
SET ERROR FLAG
JSR RO,SETEN
REPORT ERROR
WORD #6
WORD SENS?
CACHE FLUSH TESTS
WORD SENS0
FLUSH IN PROGRESS FAILED TO CLEAR
WORD SENS1
TIME FOR FLUSH TO COMPLETE RUN OUT
1S: JSR RO,(RO)
TAKE SELECTED ACTION ON ERROR
WORD 8S
LOOP ON ERROR
WORD 8S
LOOP ON TEST
CLR LOPERR
RESET LOOP ON ERROR FLAG

VERIFY VALID SET IN USE BIT WILL CHANGE AS RESULT OF CACHE FLUSH

ST053:
INC TID
UPDATE TEST ID FLAG
CLR ERROR
RESET ERROR FLAG
6S: BIT &BIT13,CCR
SELECT VALID BITS SET A
7854 014466 001412
7855 014466 005002
7856 014450 052765 000400 163250
7857 014476 032767 010000 163242
7858 014504 001402
7859 014506 005302
7860 014510 001372
7861 014512 052765 000400 163226
7862 014520 005002
7863 014522 032767 010000 163216
7864 014530 001402
7865 014532 005302
7866 014534 001372
7867 014536 016767 163204 167776
7868 014544 032767 020000 167770
7869 014552 001406
7870 014554 005067 167762
7871 014560 006010
7872 014562 014452
7873 014564 014452
7874 014566 000413
7875 014570 005267 167746
7876 014574 004067 170552
7877 014600 000006
7878 014602 037342
7879 014604 037450
7880 014606 037474
7881 014610 005067 170250
7882 014614 007671

7883
7884
7886
7887
7888

7889 014616 005267 164024
7890 014622 005002
7891 014624 005002
7892 014630 052767 000400 163170
7893 014636 032767 010000 163170
7894 014644 001402
7895 014646 005302
7896 014650 001372
7897 014652 032767 020000 163066
7898 014660 001410
7899 014662 005267 167654
7900 014666 004067 170460
7901 014672 000006
7902 014674 000006
7903 014676 037504
7904 014700 037524
7905 014702 004010
7906 014704 014712
7907 014706 014712
7908 014710 000416
7909 014712 032767 020000 163026

VERIFY VALID SET IN USE BIT WILL CHANGE AS RESULT OF CACHE FLUSH.

TST054: INC TIP

4$: CLR R2
5$: MOVE CCR,ERROR
6$: MOVE CCR,ERROR
7$: JSR RO.(RO)
8$: JSR RO.SETEN
9$: JSR RO.(RO)

ON SET A NOW
RESET TIME OUT COUNTER
CAUSE FLUSH
WAIT FOR FLUSH TO COMPLETE
FLUSH COMPLETE
WAIT LOOP
CAUSE FLUSH
RESET WAIT LOOP COUNTER
WAIT FOR FLUSH TO COMPLETE
OR TIME TO RUN OUT
VALID SET B SELECTED
SHOULD BE SET
ERROR
TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST
NEXT TEST
INC ERROR
SET ERROR FLAG
REPORT ERROR
Cache Flush Tests
Cache Flush Tests
Valid Bit In Use Bit Did Not Set As Result Of Flush
Reset Loop On Test Flag

BR TST054
BR 78
VERIFY THAT ALL TAGGED LOCATIONS WILL BE INVALIDATED
AS RESULT OF CACHE FLUSH.
TST055: INC TID
UPDATE TEST ID
1%: JSR R4,RELTEST
RELOCATE TEST TO HIGH CACHE
WORD TST056
CLR @ERROR
RESET ERROR FLAG
CLR @BIT13, @WCCR
SELECT VALID BITS SET A
BEQ $3
SET A IN USE NOW
BIS @BIT08, @WCCR
FLUSH CACHE
2%: BIS @BIT12, @WCCR
WAIT TILL COMPLETE
BNE $5

3%: MOV @11, @WCCR
ENABLE LOW CACHE
MOV @LOWSP,R2
LOW BLOCK #1 POINTER
4%: MOV (R2)+,R3
TAG LOCATION
CMP R2, @LOWSP+2000
TAG ALL OF LOW BLOCK
BNE $5
COMPLETE

BIS @BIT08, @WCCR
FLUSH CACHE SELECT VALID B
5%: BIS @BIT12, @WCCR
WAIT TILL COMPLETE
BNE $5

6%: BIS @BIT08, @WCCR
FLUSH CACHE SELECT VALID A
BIS @BIT12, @WCCR
WAIT TILL COMPLETE
BNE $6

7%: MOV @HIGH1,R2
ERROR LOG BLOCK
MOV @LOWSP,R3
PREV. TAGGED LOW BLOCK
CLR @GOOD
SUCCESSFUL VALID BIT CLEAR COUNT
CLR @BAD
UNSUCCESSFUL VALID BIT CLEAR COUNT
7%: MOV (R3)+, R5
READ FROM LOW BLOCK #1
BIT @BIT03, @WCHR
LOOK FOR READ HIT
BEQ $8
NO HIT
INC @BAD
READ HIT FROM FLUSHED CACHE
MOV #1,(R2)+
SET ERROR FLAG FOR LOCATION
9%: CMP R5, @LOWSP+2000
REPEAT FOR ALL OF LOW CACHE
BNE $7
NOT COMPLETE
BNE $7

10$: MOV @OFF, @WCCR
DISABLE CACHE
TS7 @BAD
ANY ERRORS
TST @ERROR
NO
IS IT FLUSH COUNTER ERROR
BIS @ERROR
SET ERROR FLAG
JSR R0, @SETFN
WORD "06"
VERIFY FLUSH COUNTER LOGIC BY EXAMINING FLUSH FAILURE

IS'060:

INC TID
JSR R4, RELCTL
WORD TST061
CLR @WERROR
BIT #11T3, @WCCR
BNE 3%
BIS @BIT0B, @WCCR
; FLUSH CACHE
BIT #1IT1, @WCCR
; SELECT VALID BITS SET B
BNE 2%
BIS #1IT2, @WCCR
; WAIT TILL COMPLETE
BNE 3%
MOV #5, @WCCR
; ENABLE HIGH CACHE
MOV #HIGHSP,R2
; HIGH BLOCK #1 POINTER
4$: MOV (R2)+, R3
; TAG LOCATION
CMP R2, #HIGHSP+2000
; TAG ALL OF HIGH BLOCK
BNE 4%
BIS @BIT0B, @WCCR
; FLUSH CACHE SELECT VALID A
BIT #1IT2, @WCCR
; WAIT TILL COMPLETE
BNE 5%
BIS @BIT0B, @WCCR
; FLUSH CACHE SELECT VALID B
BIT #1IT2, @WCCR
; WAIT TILL COMPLETE
BNE 6%
MOV #LOW1, R2
; ERROR LOG BLOCK
MOV #HIGHSP,R3
; PREV TAGGED LOW BLOCK
CLR @GOOD
; SUCCESSFUL VALID BIT CLEAR COUNT
CLR @BAD
; UNSUCCESSFUL VALID BIT CLEAR COUNT
7$: MOV (R3)+, R5
; READ FROM HIGH BLOCK #1
BIT #1IT03, @WCHR
; LOOK FOR READ HIT
BEQ 8%
; NO HIT
INC @BAD
; READ HIT FROM FLUSHED CACHE
MOV #1, (R2)+
; SET ERROR FLAG FOR LOCATION
9$: CMP R3, #HIGHSP+2000
; REPEAT FOR ALL OF HIGH BLOCK
BNE 7%
; NOT COMPLETE
BR 10%
; COMPLETE
8$: INC @GOOD
; NO READ HIT FROM FLUSHED CACHE
CLR (R2)+
; CLEAR ERROR FLAGS FOR LOCATION
BR 9%
MOV #OFF, @WCCR
; DISABLE Cache
TST @BAD
; ANY ERRORS
; IS IT FLUSH IN ERROR
BEQ 7%
; REPORT ERROR
INC @ERROR
JSR R0, @WSETEN
; WORD +W6
WORD SENS7
; CACHE FLUSH TESTS
WORD SENS6
; FLUSH FAILED TO INVALIDATE CACHE
WORD SENS72
; TESTING HIGH CACHE USING VALID BITS SET B
14$: JSR R0, (R0)
; LOOP ON ERROR
WORD LOWSP
; LOOP ON TEST
BR TST051
; ERROR FLAGS POINTER
11$: MOV #LOW1, R2
12$: TST (R2)+
BEQ 12%
13$: CLR @ERROR
; FIND ERRORS BIT
CACHE DIAG.  MACY11 30A1052 31-0CT-79 15:29 PAGE 66-10
CFKKAB.P11  25-JUN-79 13:31  CACHE FLUSH TESTS

```
8190 016470 005722 TST (R2)+ ; FIND NO. OF BITS
8191 016471 001374  BNE 138
8192 016474 006337 004342  ASL @ERROR
8193 016500 005257 004360  INC @BITFLG
8194 016504 004037 005332  JSR R0,@MSETEN
8195 016510 000010  ; REPORT ERROR
8196 016512 037342  ; CACHE FLUSH TESTS
8197 016514 037532  ; FLUSH FAILED TO INVALIDATE CACHE
8198 016516 037672  ; TESTING HIGH CACHE USING VALID BITS SET 8
8199 016520 037370  ; POSSIBLE FLUSH COUNTER BIT FAILUR
8200 016522 000746  
8201
8202
8203
8204
```
FORCE MISS TESTS

TEST ENABLES LOW CACHE END CAUSES MULTIPLE HEADS TO LOW CACHE WHILE KEEPING TRACK OF HOW LONG ITS TAKING TO READ BY USING THE SYSTEM CLOCK.

IF THE READ LOOP TAKES TO LONG TO COMPLETE IT IS ASSUMED THAT LOW CACHE DID NOT ENABLE.

8206 INC TID
8207 JSR R4, RELCH
8208 .WORD TSTO62
8209 3$: CLR @COUNT
8210 CLR @TIME
8211 MOV #LOWSP.R2
8212 CLR RT
8213 CLR @KOD00
8214 JST @KOD00
8215 JSTB @KOD00
8216 4$: BIC #BIT02+B0T09, @CCR
8217 ENABLE LOW CACHE
8218 1$: MOV @OFF, @CCR
8219 :DISABLE CACHE
8220 :SAVE TIME
8221 :REV B
8222 BIT #1, @COMWID
8223 :LINE 60HZ OR 50HZ
8224 BEQ 5$: BIT 0 CLEARED MEANS 60HZ
8225 CMP @ERROR, #103
8226 :IT IS 50HZ
8227 \DID LOOP COMPLETE IN TIME (50HZ)
8228 BHI 2$: 
8229 BR 6$: 
8230 CMP @ERROR, #117
8231 BHIS 2$: 
8232 :DID LOOP COMPLETE IN TIME (60HZ)
8233 :REV B
8234 6$: CLR @ERROR
8235 :RESET ERROR FLAG
8236 2$: JSR R0, @SETEN
8237 :PRINT LIST OF SENTENCES
8238 .WORD *04
8239 .WORD SEN73
8240 .WORD SEN74
8241 JSR R0 (R0)
8242 \FORCE MISS TESTS
8243 \LOW CACHE LOOKS DISABLED WHEN ENABLED
8244 \TAKE SELECTED ACTION ON ERROR
8245 \WORD 35-TSTO61-12+HIGHSP
8246 \LOOP ON TEST
8247 \WORD 35-TSTO61-12+HIGHSP
8248 \LOOP ON TEST
8249 ...
TEST ENABLES LOW CACHE AND CAUSES READS TO HIGH CACHE
HIGH CACHE READS IS TIMED USING THE SYSTEM CLOCK
IF READ LOOP COMPLETES TO FAST THEN IT IS ASSUMED THAT HIGH CACHE IS ENABLED

ST062: INC TID : UPDATE TEST ID
        JSR R4,RELCTH : RELOCATE THIS TEST TO HIGH ACHE
        WORD ST063
3$: CLR @COUNT : RESET LOOP COUNTER
    CLR @TIME : RESET CLOCK
    MOV @HIGH?,R2 : HIGH CACHE ADDRESS
    CLR R1 : LOOP COUNTER
    CLR @K00K00 : RESET TICK
7$: BIC @BIT02+BIT09,#CCR : ENABLE LOW CACHE
    JSB @K00K00 : LOOK FOR TICK
    BEC 10$: NO TICK
7$: INC @TIME : CLOCK TICK
        CLR @K00K00 : RESET TICK
10$: CMP (R2),#0 : READ FROM CACHE
    CMP (R2),#0
INC R1
BNE 45
MOV #OFF,#CCR
MOV @TIME,#ERROR
;DISABLE CACHE
;SAVE TIME TO COMPLETE LOOP
8290
8291
8292
8293  017022  032737  000001  000176
8294  017030  001405
8295  017032  023727  004542  000103
8296
8297  017040  103407
8298  017042  000404
8299  017044  023727  004542  000117  S$:  CMP  @ERROR, #117

; REV B

BIT #1, @COMWRD
BEQ  S$
CMP @ERROR, #103

; LINE 60Hz OR 50Hz
; BIT O' CLEARED MEANS 60Hz
; IT IS 50Hz

BLO 2$
BR 6$

; DID LOOP COMPLETE IN TIME (50Hz)
; NO

; DID LOOP COMPLETE IN TIME (60Hz)
<table>
<thead>
<tr>
<th>Address</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>8301</td>
<td>017052</td>
<td>103402  BLO 2%; NO</td>
</tr>
<tr>
<td>8302</td>
<td></td>
<td>:REV B</td>
</tr>
<tr>
<td>8303</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8304</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8305</td>
<td>017054</td>
<td>005037 004542  6%; CLR #ERROR; RESET ERROR FLAG</td>
</tr>
<tr>
<td>8306</td>
<td>017060</td>
<td>004037 005352  2%; JSR RO,#SETEN; PRINT LIST OF SENTENCES</td>
</tr>
<tr>
<td>8307</td>
<td>017064</td>
<td>000004  :WORD &quot;D4&quot;</td>
</tr>
<tr>
<td>8308</td>
<td>017066</td>
<td>037774  :FORCE MISS TESTS</td>
</tr>
<tr>
<td>8309</td>
<td>017070</td>
<td>037742  :WORD SEN73; HIGH CACHE LOOKS ENABLED WHEN DISABLED</td>
</tr>
<tr>
<td>8310</td>
<td>017072</td>
<td>004010  JSR RO,(RO); TAKE SELECTED ACTION ON ERROR</td>
</tr>
</tbody>
</table>
 CACHE DIAG.  MACY11 30A(1052)  31-OCT-79  15:20  PAGE 70

.FORCE MISS TESTS

8312  017074  046000
8313  017076  046000
8314
8315
8316
8317
8318
8319
8320
8321
8322
8323
8324  017100  005267  161542
8325  017104  004467  167004
8326  017110  017266
8327  017112  005037  004704
8328  017116  005037  004702
8329  017122  012702  050000
8330  017126  005001
8331  017130  005037  177546
8332  017134  042737  001010  177746
8333  017142  105737  177546
8334  017146  001404
8335  017150  005267  004702
8336  017154  005037  177546
8337  017160  021212
8338  017162  021212
8339  017164  021212
8340  017166  021212
8341  017170  005201
8342  017172  001363
8343  017174  012737  001075  177746
8344  017202  013737  004702  004542
8345
8346
8347
8348  017210  032737  000001  000176
8349  017216  001405
8350  017220  032372  004542  000103
8351
8352  017236  103407
8353  017250  000404
8354  017252  032372  004542  000117
8355  017260  103402
8356
8357
8358
8359  017262  005037  004542
8360  017266  004037  005352
8361  017252  000004
8362  017254  037214
8363  017256  037760
8364  017260  004010
8365  017262  044000
8366  017264  044000
8367

.CHASE S8-TS062-12-HI MSPL
:LOOP ON ERROR
.CHASE S8-TS062-12-HI MSPL
:LOOP ON TEST

.TEST ENABLES HIGH CACHE AND causes TO LOW CACHE

.SYSTEM CLOCK IS USED TO TIME READ LOOP

.IF READ LOOP COMPLETES TO FAST
.

.THEN IT IS ASSUMED THAT LOW CACHE IS ENABLED

.TS063:

.INC TID
:UPDATE TEST ID

.JSR R4,RECLCL
:RELOCATE TEST TO LOW CACHE

.WORD TS064

3S:

CLR @RCOUNT
:RESET COUNTER

CLR @RTIME
:RESET CLOCK

MOV #LOW1,R2
:LOW CACHE ADDRESS

CLR R1

CLR @K6000
:RESET TCK

BIC @BIT03+BIT09,A,CCR
:ENABLE HIGH CACHE

4S:

TSB @K6000
:LOOK FOR TCK

BEQ 10S
:N0 TCK

INC @RTIME
:CLOCK TCK

CLR @K6000
:RESET TCK

10S:

CMP (R2),(R2)

CMP (R2),(R2)

CMP (R2),(R2)

INC R1

BNE 4S

MOV #OFF,A,CCR
:DISABLE CACHE

MOV @RTIME,A,ERROR
:SAVE TIME TO COMPLETE LOOP

:REV B

.BIT #1,CONMND
:LINE 60Hz OR 50Hz

.BEQ 5S
:BIT 0 CLEANED MEANS 60Hz

CMP @ERROR,#103
:IT IS 50 Hz

.BLO 28
:DO LOOP COMPLETE IN TIME(50Hz)

BR 6S

5S:

CMP @ERROR,#117

.BLO 28
:DO LOOP COMPLETE IN TIME(60Hz)

:REV B

6S:

CLR @ERROR
:RESET ERROR

.JSR RO,ASETEN
:PRINT LIST OF SENTENCES

.WORD "4"

.WORD S073
:LOW CACHE LOOKS ENABLED WHEN DISABLED

.JSR RO,(R0)
:TAKE SELECTED ACTION ON ERROR

.WORD S8-TS063-12+LOWSP
:LOOP ON ERROR

.WORD S8-TS062-12+LOWSP
:LOOP ON TEST
TEST ENABES HIGH CACHE AND CAUSES MULTIPLE READS TO HIGH CACHE.
HIGH LOOP IS TIMED USING SYSTEM CLOCK IF READ LOOP COMPLETES TO SLOW THEN IT IS ASSUMED THAT HIGH CACHE DID NOT ENABLE.

TST064:
INC TIC
CLR @COUNT
CLR @TIME
MOV #HIGHSP,R2
MOV #LOWSP,R2
LOOP COUNTER
CLR @KICKOUT
BIC B2:0,0
BIC B1:0,0
BIC B0,0
Enable Low Cache
TSTB @KICKOUT
BIC
LOCK FOR JACK
BEQ 10%
INC @TIME
LOCK COUNTER
BEC 4%
MOV #OFF,B0
DISABLE CACHE
MOV #OFF,B0
SAVE LOOP TIME

BIT 0, CONRA
: LINE 60HZ OR 50HZ
BIT 0 CLEARED 60HZ
BIT 5
BIT 5
BIT 5
BIT 5
BIT 5
SBTTL WRITE WRONG PARITY TESTS

ENABLE WRITE WRONG PARITY TO TAG AND DATA, ENABLE HIGH CACHE.
VERIFY PARITY ERROR TO TAG AND DATA.

TST065:
INC TID
JMP R4, REL 30CL
COR Test ID
RELOCATE TEST TO LOW CACHE

MOV @OFF+BIT06, @CCR
FLUSH CACHE

10%:
DEC R1
CHECK FOR TIME OUT

BIT @R12, @CCR
WAIT TILL COMPLETE

BNE 10%:
MOV @HIGHSP, R1
WRITE TO HIGH CACHE LOCATIONS

1%:
MOV @HIGHSP, R1
WRITE WRONG ALL TO HIGH CACHE

2%:
MOV @R10+R1, @CCR
CAUSE WRITES TO 6 HIGH CACHE LOCATIONS

CMP R1, #HIGHSP+14
LOOP

BNE 2%:
NOT COMPLETE

MOV @BIT02+BIT00, @CCR
ENABLE HIGH CACHE ONLY

CLR @CPPE
RESET PARITY ERRORS

CLR @ERROR
RESET ERROR FLAG

3%:
MOV -(R1), R5
READ TO CAUSE ERROR

BIS #215, @CCR
SET ABORT FOR ERROR READ

BIS @CPPE, @ERROR
SAVE ERROR INFO

CLR @CPPE
RESET ERROR REGISTER

MOV @5, @CCR
DISABLE ABORT

CMP R1, #HIGHSP
:LOOP SIX TIMES

BNE 3%:
MOV PC, R2
CORRECT ANY WRONG PARITY TO LOW CACHE

11%:
MOV -(R2), R1
READ LOW CACHE

CMP R2, #LONSP
CONTINUE TO START OF BLOCK

BNE 11%:
COM @ERROR
PARITY ERRORS HIGH LOW AND TAP

BIC #177437, @ERROR
RESET ERROR REGISTER

BIC @CPPE
DISABLED CACHE

MOV @OFF, @CCR
IF NOT SET THEN ERROR

BEQ 4%:
MOV #BIT06, @ERROR
WRITE PARITY ERROR BIT06 NOT SET IN CPPE

BNE 5%:
MOV #BIT07, @ERROR
WRITE PARITY ERROR BIT07 NOT SET IN CPPE

BNE 5%:
JSR R0, @SETEN
PRINT LIST OF SENTENCES

JSR @D6
WRITE PARITY TESTS

WORD SEN78
WRITE WRONG PARITY TO LOW BYTE HIGH BYTE AND TA

WORD SEN80
LOW BYTE PARITY ERROR BIT06 NOT SET IN CPPE

BNE 5%:
JSR R0, @SETEN
PRINT LIST OF SENTENCES

JSR @D6
WRITE PARITY TESTS

WORD SEN79
WRITE WRONG PARITY TO LOW BYTE HIGH BYTE AND TA

WORD SEN78
WRITE WRONG PARITY TO LOW BYTE HIGH BYTE AND TA

WORD SEN81
HIGH BYTE PARITY ERROR BIT07 NOT SET IN CPPE

BNE 5%:
REO 68
JSR R0, @SETEN : PRINT LIST OF SENTENCES

WORD *D6

WORD SEN78 : WRITE WRONG PARITY TESTS

WORD SEN99 : WRITE WRONG PARITY TO LOW BYTE HIGH BYTE AND TH

WORD SNNB2 : TAG PARITY ERROR BIT 0S NOT SET IN CMPE

68: JSR R0, (RO) : TAKE SELECTED ACTION ON ERROR

WORD 1S-TST065-12+LOWSP : LOOP ON ERROR

WORD 1S-TST065-12+LOWSP : LOOP ON TEST
SATTL HIT REGISTER TESTS

; CHECK THAT ALL SIX HIT BITS CAN CONTAIN ZEROS

TST066: INC TID ; UPDATE TEST ID
          JSR R4,RELCTL ; RELOCATE TEST TO LOW CACHE
          .WORD TST067
          MOV PC,R1 ; LOW TEST AREA
          CMP R1,(R1)+
          BNE 2%

2%: MOV R1,#18-TST066-12+LOWSP
          NOP
          NOP
          NOP
          NOP
          NOP
          NOP
          NOP
          NOP
          MOV R0,(R0)+
          CMP R0,(R0)+
          BNE 1%

1%: MOV @CHR,#ERROR ; READ AND SAVE HIT REGISTER CONTENTS
          BIC #177700,#ERROR ; MASK FOR HIT BITS
          INC @BITFLG ; BIT PRINT MODE
          JSR RO,#SETEN ; REPORT ERROR IF ANY
          .WORD #6
          .WORD SEN83 ; HIT REGISTER TESTS
          .WORD SEN84 ; CACHE HIT REGISTER BIT(S) STUCK HIGH
          .WORD LOWSP ; TAKE SELECTED ACTION ON ERROR
          .WORD LOWSP

; CHECK THAT HIT BITS CAN CONTAIN A ONE

TST067: INC TID ; UPDATE TEST ID
          JSR R4,RELCTL ; RELOCATE TEST TO LOW CACHE
          .WORD TST070
          MOV #5,#CCR ; ENABLE HIGH CACHE
          MOV @HIGHSP,R2 ; CAUSE READ MISS
          MOV @HIGHSP,R2 ; CAUSE READ HIT
          NOP
          NOP
          NOP
          NOP
          MOV @CHR,#ERROR ; READ AND SAVE HIT REGISTER
          BIC #177797,#ERROR ; MASK FOR BIT 5
          SUB #40,#ERROR ; BIT 5 SHOULD BE SET
          MOV #OFF,#CCR ; DISABLE CACHE
          JSR RO,#SETEN ; REPORT ERROR IF ANY
          .WORD #5B
          .WORD SEN85 ; HIT REGISTER TESTS
          .WORD SEN86 ; HIT BIT ERROR
          .WORD SEN87 ; ATTEMPT TO WRITE HIT REGISTER BITS
          .WORD LOWSP ; TO A ONE , VAI READ HIT, FAILED
          .WORD LOWSP ; LOOP ON ERROR
          .WORD LOWSP ; LOOP ON TEST
8549 020160 005267 160462 TST07: LSBTL CACHE DATA WRITE BYTE TEST
8550 020164 004467 165724 INCL TID: UPDATE TEST ID
8551 020170 020374 JSR R4:RELCIL: RELOCATE TEST TO LOW CACHE
8552 020172 012737 000005 177746 MOV #5, @WCCR: ENABLE HIGH CACHE
8553 020200 013701 046000 MOV @HIGHSP, R1: READ MISS
8554 020204 012737 000000 046000 MOV @HIGHSP: WRITE WORD HIT
8555 020212 112737 000023 046000 MOV #23, @HIGHSP: WRITE BYTE HIT
8556 020220 112737 000023 046001 MOVB #23, @HIGHSP+1: WRITE BYTE HIT
8557 020224 005037 177744 CLR @CMPE: RESET ERROR REGISTER
8558 020232 013703 046000 MOV @HIGHSP, R3: READ AND LOOK FOR ERROR
8559 020236 012727 000215 177746 MOV #215, @WCCR: SET ABORT FOR ERROR READ
8560 020244 013737 177744 004542 MOV @CMPE, @ERROR: SAVE ERROR
8561 020252 012737 001015 177746 MOV @OFF, @WCCR: ENABLE CACHE
8562 020260 032737 000100 004542 BIT @BIT06, @ERROR: ANY LOW BYTE ERROR
8563 020266 001407 BEQ 1$: REPORT ERROR
8564 020270 004037 005352 3$: JSR R0, @WSETEN: CACHE DATA WRITE BYTE TEST
8565 020274 000006 .WORD "D6: CACHE DATA WRITE BYTE ERROP
8566 020276 040246 .WORD SENB8: CANT WRITE LOW BYTE
8567 020300 040262 .WORD SENB9: CANT WRITE LOW BYTE
8568 020302 040276 .WORD SEN90: CANT WRITE HIGH BYTE
8569 020304 000430 .WORD SEN91: CANT WRITE HIGH BYTE
8569 020306 032737 000200 004542 1$: BIT @BIT07, @ERROR: ANY HIGH BYTE ERROR
8570 020314 001407 BEQ 5$: REPORT ERROR
8571 020316 004037 005352 5$: JSR R0, @WSETEN: CACHE DATA WRITE BYTE TEST
8572 020322 000006 .WORD "D6: CACHE DATA WRITE BYTE ERROP
8573 020324 040246 .WORD SENB8: CANT WRITE LOW BYTE
8574 020326 040262 .WORD SENB9: CANT WRITE LOW BYTE
8575 020330 040210 .WORD SEN91: CANT WRITE HIGH BYTE
8576 020332 000435 .WORD SEN91: CANT WRITE HIGH BYTE
8577 020334 010337 004542 5$: MOV R3, @ERROR: WAS DATA READ BACK OK
8578 020340 012703 011423 MOV #11423, R3: PREVENT LOOP
8579 020344 162737 011423 004542 SUB #11423, @ERROR: DATA THAT SHOULD BE
8580 020352 105737 004542 TSTB @ERROR: ANY LOW BYTE ERROR
8581 020356 001354 TSTB @ERROR+1: ANY HIGH BYTE ERROR
8582 020360 001350 BNE 1$: YES
8583 020364 105737 004543 BNE 1$: YES
8584 020366 001350 2$: JSR R0, (R0): TAKE SELECTED ACTION ON ERROR
8585 020370 044000 .WORD LOWSP
8586 020372 044000 .WORD LOWSP
CACHE DIAG. \[ MACY1130A(1052) \] 31-OCT-79 15:29 PAGE 70-6
CFKAB, P11 25-JUN-79 13:31
PARITY ERROR REGISTER TESTS

.SBTL PARITY ERROR REGISTER TESTS

: VERIFY PARITY ERROR REGISTER BIT15 SETS AS RESULT
: OF TAG ERROR

8589
8590
8591
8592
8593
8594
8595
8596

8597 020376 005267 160266
8598 020400 004467 165510
8599 020404 020560
8600 020406 005037 177746
8601 020412 012737 000005 177746
8602 020420 052000
8603 020424 013702 046000
8604 020430 052737 002000 177746
8605 020436 013702 052000
8606 020442 042737 002000 177746
8607 020448 013702 052000
8608 020454 052737 000215 177746
8609 020460 013702 177746
8610 020466 005037 001015 177746
8611 020472 0010701
8612 020502 014111
8613 020504 020127 044000
8614 020510 001374
8615 020512 032702 000040
8616 020516 001415
8617 020520 010237 004542
8618 020524 100412
8619 020526 004037 005352
8620 020532 00006
8621 020534 003222
8622 020536 040334
8623 020540 040356
8624 020542 040100
8625 020544 044000
8626 020546 044000
8627 020550 000403
8628 020552 005037 004542
8629 020556 000771

3%: MOV -(R1), (R1) ;CAUSE CACHE WRITE IN BYPASS MODE
8630
8631
8632
8633
8634
8635
8636
8637
8638
8639
8640
8641
8642
8643
8644

8645 020560 005267 160062
8646 020564 004467 165324
8647 020570 020746
8648 020572 005037 177746
8649 020576 012737 000005 177746
8650 020580 013702 052000
8651 020584 013702 046000

TST071: INC TID ;UPDATE TEST ID
JSR R4, RELCTL ;RELOCATE TEST TO LOW CACHE
.WORD TST072
CLR @CMPE ;RESET CACHE ERROR REGISTER
MOV #5, @CCCR ;ENABLE HIGH CACHE
MOV @HIGH, R2 ;TAG HIGH LOCATION BLOCK #2
MOV @HIGHSP, R2 ;TAG HIGH LOCATION BLOCK #1
TST072: INC TID ;UPDATE TEST ID
JSR R4, RELCTL ;RELOCATE TEST TO LOW CACHE
.WORD TST073
CLR @CMPE ;RESET CACHE PARITY ERROR REGISTER
MOV #5, @CCCR ;ENABLE HIGH CACHE
MOV @HIGH, R2 ;TAG HIGH LOCATION BLOCK #2
MOV @HIGHSP, R2 ;TAG HIGH LOCATION BLOCK #1

: VERIFY PARITY ERROR REGISTER BIT15 SETS AS RESULT
: OF LOW BYTE PARITY ERROR

8652
8653
8654
8655
8656
8657
8658
8659
8660
8661
8662
8663
8664
CACHE DIAG.  MACY11 30A(1052)  31-OCt-79  15:29  PAGE 70-7
(TKKAB.P11)  25-JUN-79 13:31

PARITY ERROR REGISTER TESTS

BIS #BIT0..MCCR ; ENABLE WRITE WRONG DATA
MOV #3, #HIGHSR ; WRITE WRONG PARITY LOW BYTE
BIC #BIT0, #MCCR ; DISABLE WRITE WRONG
MOV #5, #HIGHSR,R2 ; READ AND CAUSE LOW BYTE PARITY ERROR
BIS #215, #MCCR ; ENABLE ABORT FOR ERROR READ
MOV @CMPE,R2 ; READ AND SAVE ERROR REGISTER
MOV #OFF, #MCCR ; DISABLE CACHE
CLR #ERROR ; RESET ERROR REGISTER
MOV P,C,R3 ; UNTAG LOW CACHE-parity
MOV -#R3, #R3 ; CAUSE WRITE IN BYPASS MODE
CMP R3,#LOWSP
BNE 5:
BIT #BIT0,R2 ; ANY LOW BYTE PARITY ERROR
BEQ 1:
MOV R2,#ERROR ; WAS BIT15 SET AS RESULT OF ERROR
BMI 1:
JSR RO,#SETEN
WORD #6
WORD SEN92
WORD SEN95
BIT15 OF CMPE DID NOT SET
WORD SEN96
WORD #6
AS RESULT OF LOW BYTE PARITY ERROR
JSR RO,(RO) ; TAKE SELECTED ACTION ON ERROR
WORD #4
WORD LOWSP
LOOP ON ERROR
LOWSP
LOOP ON TEST
BR TS073:
1:
CLR #ERROR ; RESET ERROR FLAG
BR 2:

: VERIFY PARITY ERROR REGISTER BIT15 SETS AS RESULT
: OF HIGH BYTE PARITY ERROR

8645 020614 005273 000100 177746
8646 020622 112737 000000 046000
8647 020630 042737 000000 177746
8648 020636 013702 006000 177746
8649 020642 052737 000000 217746
8650 020650 013702 177746
8651 020654 012737 000105 177746
8652 020662 005037 004542
8653 020666 010703
8654 020670 014313
8655 020672 020327 044000
8656 020676 001374
8657 020700 032702 000100
8658 020704 001415
8659 020706 010237 004542
8660 020712 100412
8661 020714 004037 003532
8662 020720 000006
8663 020722 040322
8664 020724 040370
8665 020726 040406
8666 020730 004010
8667 020732 044000
8668 020734 044000
8669 020736 000403
8670 020740 005037 004542
8671 020744 000771

8678
8679
8680 020746 005267 157674
8681 020752 004467 165136
8682 020756 021134
8683 020760 005037 177746
8684 020764 012737 000000 177746
8685 020772 013702 052000
8686 020776 013702 046000
8687 021002 052737 000100 177746
8688 021010 112737 000000 046001
8689 021016 042737 000000 177746
8690 021024 013702 046000
8691 021030 052737 000215 177746
8692 021036 013702 177746
8693 021042 013702 001015 177746
8694 021050 005037 004542
8695 021054 010703
8696 021056 014313
8697 021060 020327 044000
8698 021064 001374
8699 021066 032702 000200
8700 021072 001415

TS073:
INC TID : UPDATE TEST ID
JSR R4, RELCTL : RELOCATE TEST TO LOW CACHE
.WORD TST074
CLR #CMPE
MOV #5, #MCCR
MOV #HIGHSR,R2
MOV #HIGHSR,R2
BIT #BIT0..MCCR
MOV #HIGHSR,R2
MOV #HIGHSR,R2
MOV #HIGHSR,R2
BIT #215, #MCCR
MOV @CMPE,R2
MOV #OFF, #MCCR
CLR #ERROR
MOV P,C,R3
MOV -#R3, #R3
CMP R3,#LOWSP
BNE 5:
BIT #BIT0,R2
BEQ 1:
MOV -#R3, #R3
CMP R3,#LOWSP
BNE 5:
BIT #BIT0,R2
BEQ 1:
MOV -#R3, #R3
CMP R3,#LOWSP
BNE 5:
BIT #BIT0,R2
BEQ 1:
DATA DIAG 31-OCT-79 15:29 PAGE 70-9
CFKBAP-PT 25-JUN-79 13:31
PARITY ERROR REGISTER TESTS

8701 021074 010237 004542
8702 021100 100412
8703 021102 004307 005352
8704 021106 000000
8705 021110 040322
8706 021112 040426
8707 021114 040444
8708 021116 004010
8709 021120 040000
8710 021122 040000
8711 021124 000403
8712 021126 005037 004542
8713 021132 000771
8714 021134 005267 157506
8715 021140 004467 164750
8716 021144 021276
8717 021146 005037 177744
8718 021152 021273 001000 177746
8719 021160 013022 052000
8720 021164 012737 002105 177746
8721 021172 013762 046000
8722 021176 012737 000000 177746
8723 021204 013702 046000
8724 021210 005037 177744
8725 021214 012737 001213 177746
8726 021222 013737 177744 004542
8727 021230 012737 000000 177746
8728 021236 010702
8729 021242 012122 044000
8730 021246 001227 044000
8731 021247 013747 001015 177746
8732 021250 014377 001015 005352
8733 021256 004307 005352
8734 021262 000000
8735 021266 004322
8736 021269 040464
8737 021270 004010
8738 021272 040000
8739 021274 040000
8740 021276 040000
8741 021278 040000
8742 021280 040000
8743 021282 040000
8744 021284 040000
8745 021286 040000
8746 021288 040000
8747 021290 040000
8748 021292 040000
8749 021294 040000

MOV R2, #ERROR
BMI 1%
JSR RO, #SETEN
WORD '06
WORD SEN92
WORD SEN97
WORD SEN98
PARITY ERROR REGISTER TESTS
BIT15 OF CMPE WAS NOT SET
AS RESULT OF HIGH BYTE PARITY ERROR
TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST

2S: JSR RO, (R0)
LOWSP
BR TST074
LOWSP
BR 2S:

1S: CLR #ERROR
SET ERROR FLAG

TST074:
INC TID
UPDATE TEST ID
JSR R4, #RECLT
RELOCATE TEST TO LOW CACHE

1S: CLR #CMPE
RESET ERROR REGISTER
MOV R5, #CMCR
ENABLE HIGH CACHE
MOV #HIGH,R2
POINT TAG TO HIGH BLOCK #2
MOV #HIGHSP,R2
WRITE WRONG TO HIGH CACHE
MOV #CMCR
DISABLE WRITE WRONG
MOV #CMHIGHSP,R2
DISABLE READ FROM CACHE
CLR #CMPE
RESET ERROR REGISTER
MOV #OFF+BIT07,#CMCR
ENABLE ABORT FOR ERROR READ
MOV #CMPE,#ERROR
SAVE ERROR INFO.
DISABLE CACHE
MOV PC,R2
CORRECT ANY WRONG PARITY IN LOWER
MOV -(R2),(R2)
CAUSE READ TO LOW CACHE
CMP R2,#LOWSP
FOR ALL LOW CACHE USED
BNE 2S

MOV #OFF,#CMCR
DISABLE CACHE
JSR RO, #SETEN
PRINT LIST OF SENTENCES
WORD '04
WORD SEN92
PARITY ERROR REGISTER TESTS
WRITE TO CMPE FAILED TO CLEAR REGISTER
TAKE SELECTED ACTION ON ERROR
LOOP ON ERROR
LOOP ON TEST

...
SBTL PARITY ERROR LOGIC TESTS

VERIFY PARITY ERROR LOGIC BY EVEN AND ODD DATA INTO CACHE
TAG AND DATA. IF A PARITY ERROR OCCURS ON EVEN DATA
BUT NOT ON ODD THEN LOGIC IS ASSUMED GOOD.
ONLY IF BOTH ODD AND EVEN FAIL IS THE LOGIC ASSUMED BAD.

TST075:   INC TID   JSR R4, RELCTL   ; RELOCATE TEST TO LOW CACHE
          .WORD TST076
          .WORD 3
          .JSR R4, RELCTL   ; ENABLE HIGH CACHE
          MOV #5, @CCR   ; RESET ERROR REGISTER
          CLR @CMPE
          MOV @HIGH, R2   ; POINT TAG TO HIGH BLOCK #2
          MOV @HIGHSP, R2   ; POINT TAG TO HIGH BLOCK #1
          MOV #6, @HIGHSP   ; WRITE DATA INTO IT
          MOV @HIGHSP, R2   ; READ EVEN DATA WITH EVEN TAG
          MOV @CMPE, @ERROR   ; SET ABORT TO READ ERROR
          MOV @CMPE, @ERROR   ; SAVE ERROR INFO
          MOV #5, @CCR   ; DISABLE ABORT
          MOV @HIGH, R2   ; POINT TAG TO HIGH BLOCK #2
          MOV #4, @HIGH   ; WRITE ODD DATA INTO IT
          CLR @CMPE   ; READ ODD DATA AND TAG
          MOV #5, @CCR   ; RELOCATE TEST TO LOW CACHE
          MOV @HIGH, R2   ; ENABLE ABORT FOR ERROR READ
          MOV @HIGHSP, R2   ; READ ERROR REGISTER
          MOV @CMPE, @ERROR   ; DISABLE ABORT
          MOV @CMPE, @ERROR   ; DISABLE CACHE
          MOV #5, @CCR   ; AND RESULTS OF ODD AND EVEN
          MOV R2, @ERROR   ; ANY BITS SET = ERROR
          BIT #0, @ERROR   ; WAS LOW PARITY SET
          BEQ #7
          JSR R0, @SETEN
          .WORD 00000000, 00000000
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .WORD #06
          .JSR R0, @SETEN
8863   022046  040570
8864   022050  040612
8865   022052  032737  000040  004542
8866   022060  001407
8867   022062  004037  005352
8868   022066  000010
8869   022070  040504
8870   022072  040560
8871   022074  040570
8872   022076  040612
8873   022100  042737  000140  177746
8874   022106  052737  001010  177746
8875   022114  004010
8876   022116  040000
8877   022120  040000

...
.SB TTL READ WRITE HIT TESTS

TST077: INC TID

JMP R4, RELCH

: UPDATE TEST ID

: RELOCATE THIS TEST TO HIGH CACHE

CLR @ERROR

: RESET ERROR REGISTER

MOV #LOW1, R1

: POINTER TO LOW BLOCK #2

MOV #11, @CCR

: ENABLE LOW CACHE

MOV 1, R2

: TAG SIX LOW CACHE LOCATIONS

CMP R1, #LOW1+14

BEQ 18

: RESET LOW CACHE POINTER

MOV #LOW1, R1

: CAUSE WRITES TO 6 LOW CACHE LOCATIONS

2%

CMP R1, #LOW1+14

BNE 28

: COMPLETE YET

: NO

3%

MOV -(R1), R2

: READ PREV. WRITTEN LOCATIONS

MOV @CHR, R3

: SAVE HIT REGISTER

MOV #BIT02, R3

: WAS LAST ACCESS A HIT

BEQ 30%

: NO

INC @ERROR

: YES

5%

CMP #LOW1, R1

: LOOP TILL COMPLETE

BNE 35

: DISABLE CACHE

MOV #OF, @CCR

: LESS THAN 6 HITS = ERROR

SUB #5, @ERROR

: PRINT LIST OF SENTENCES

JMP RQ, @SETEN

: READ WRITE HIT TESTS

WORD SENO7

: NO HITS AS RESULT OF READS TO SIX TAGGED LOCATIONS

WORD SENO8

: TAKE SELECTED ACTION ON ERROR

JMP RQ, RO1

: LOOP ON ERROR

WORD HIGSP

: LOOP ON TEST
SBTTL I/O PAGE TESTS

: VERIFY THAT I/O PAGE CAN NOT BE TAGGED OR CAUSE A READ HIT. OPERATING IN MAINTENANCE MODE.

8920 8921 8922
8923 022262 005267 156360
8924 022266 004467 163622
8925 022266 00452 00452
8926 022266 00452 00452
8927 022266 00452 00452
8928 022266 00452 00452
8929 022266 00452 00452
8930 022266 00452 00452
8931 022266 00452 00452
8932 022266 00452 00452
8933 022266 00452 00452
8934 022266 00452 00452
8935 022266 00452 00452
8936 022266 00452 00452
8937 022266 00452 00452
8938 022266 00452 00452
8939 022266 00452 00452
8940 022266 00452 00452
8941 022266 00452 00452
8942 022266 00452 00452
8943 022266 00452 00452
8944 022266 00452 00452
8945 022266 00452 00452
8946 022266 00452 00452
8947 022266 00452 00452
8948 022266 00452 00452
8949 022266 00452 00452
8950 022266 00452 00452
8951 022266 00452 00452
8952 022266 00452 00452
8953 022266 00452 00452
8954 022266 00452 00452
8955 022266 00452 00452
8956 022266 00452 00452
8957
8958
8959
8960
8961
8962
8963 022452 005267 156170
8964 022452 005267 156170
8965 022452 005267 156170
8966 022452 005267 156170

TST101:

INC TID
JSR R4, RELCTL
;
RELOCATE TEST TO LOW CACHE

CLR @ERROR
;
RESET ERROR FLAG

MOV @R5, @R6
;
PASS COUNTER

MOV @RHG, @R5
;
ENABLE HIGH CACHE

MOV @RHSP+1776.35:ATTEMPT TO TAG I/O PAGE

MOV @RHSP, @R5
;
ATTEMPT HIT FROM I/O PAGE

BIT #BIT03, @CHR
;
CHECK FOR HIT

BEO.18
;
NO HIT

MOV #OFF, @R5
;
DISABLE CACHE

MOV @R5, @R6
;
SET ERROR FLAG

JSR RO, @SETEN
;
REPORT ERROR

WORD "D4"
;
I/O PAGE TESTS

WORD SEN10
;
READ INTO I/O PAGE CAUSED HIT

BR 2S

MOV @RHG, @R5
;
READ PREG TAGGED LOCATION

BIT #BIT03, @CHR
;
VERIFY HIT

BNE 2S
;
THEIR WAS A HIT

MOV #OFF, @R5
;
DISABLE CACHE

MOV @R5, @R6
;
SET ERROR FLAG

JSR RO, @SETEN
;
REPORT ERROR

WORD "D4"
;
I/O PAGE TESTS

WORD SEN110
;
READ INTO I/O PAGE INVALIDATED TAGGED LOCATION

MOV #OFF, @R5
;
DISABLE CACHE

JSR RO, (RD)
;
TAKE SELECTED ACTION ON ERROR

WORD 3S-TST100-12+LOWSP
;
LOOP ON ERROR

WORD 3S-TST100-12+LOWSP
;
LOOP ON ERROR

JSR RO, (RD)
;
LOOP ON TEST CHECK

WORD 5S-TST100-12+LOWSP
;
RESTART TEST

VERIFY THAT I/O PAGE CAN NOT BE TAGGED OR CAUSE A READ HIT.
8968 022470 012702 000006
8969 022474 012737 000001 177750
8970 022502 012737 000005 177746
8971 022510 013703 047776
8972 022514 013703 177776
8973 022520 013703 177776
8974 022524 032737 000010 177752
8975 022532 001417
8976 022534 012737 000001 004542
8977 022542 050537 177750
8978 022546 012737 001015 177746
8979 022554 060437 005352
8980 022560 000006
8981 022562 040664
8982 022564 040674
8983 022566 040732
8984 022570 000424
8985 022572 013703 047776
8986 022576 032737 000010 177752
8987 022604 003106
8988 022606 005037 177750
8989 022612 012737 001015 177746
8990 022620 012737 000001 004542
8991 022626 004037 005352
8992 022632 000006
8993 022634 040664
8994 022636 040712
8995 022640 040726
8996 022642 005037 177750
8997 022646 012737 001015 177746
8998 022654 040604
8999 022656 040410
9000 022660 04176
9001 022662 004037 005012
9002 022666 044004
9003
9004
9005
9006
VERIFIED ADDRESS LINES TO CACHE MEMORY CHIPS
BY WRITING AND READING DATA PATTERNS

: REPORT ERROR
: VALID BIT STORAGE TESTS
: OPERATING IN LOW CACHE WITH VALID SET A
: ADDRESS BIT ERROR

: VERIFY ADDRESS LINES TO CACHE MEMORY CHIPS
: BY WRITING AND READING DATA PATTERNS

: UPDATE TEST ID
: RELOCATE TEST TO LOW CACHE
: PATTERN SELECTOR

: JSR R0, #SETEN
: WORD #D6
: WORD #E113
: WORD #E114
: WORD #E115
: BR 12B

9064 023152 004037 005352
9065 023156 000006
9066 023160 040744
9067 023162 040756
9068 023164 041000
9069 023166 000753
9070
9071
9072
9073
9074
9075
9076 023170 005267 155452
9077 023174 004467 162714
9078 023200 023470
9079 023202 012702 000001
```asm
9081 023206 005037 005064 11S: CLR 01706  
9082 023212 012703 046000  MOV #HIGHSP,R3  
9083 023216 005037 004550  CLR #GOOD  
9084 023222 005037 004552  CLR #BAD  
9085 023226 052757 000000 177746  BIS #BIT09,#ACCR  
9086 023234 032757 010000 177746  BIT #BIT12,#ACCR  
9087 023242 001374  BNE 1S  
9088 023244 032757 020000 177746  BIT #BIT13,#ACCR  
9089 023252 001765  BEO 3S  
9090 023254 012757 000005 177746  MOV #5,#ACCR  
9091 023262 010201 6S: MOV R2,R1  
9092 023264 012305  MOV (R3)+,R5  
9093 023266 005301  DEC R1  
9094 023270 001375  BNE 2S  
9095 023272 020327 050000  CMP R3,#HIGHSP+2000  
9096 023276 001410  BEO 7S  
9097 023300 010201 5S: MOV R2,R1  
9098 023302 062703 000002  ADD #2,R3  
9099 023306 005301  DEC R1  
9100 023310 001374  BNE 5S  
9101 023312 020327 050000  CMP R3,#HIGHSP+2000  
9102 023316 001361  BEO 6S  
9103 023320 012703 046000  MOV #HIGHSP,R3  
9104 023324 012305  MOV (R3)+,R5  
9105 023326 032757 000010 177752  BIT #BIT03,#ACCR  
9106 023334 001006  BNE 8S  
9107 023336 005237 004552  CMP R3,#HIGHSP+2000  
9108 023342 020327 050000  BNE 9S  
9109 023346 001366  BEO 3S  
9110 023350 000405  BNE 2S  
9111 023352 005237 004550  INC #BAD  
9112 023356 020327 050000  CMP R3,#HIGHSP+2000  
9113 023362 001360  BEO 9S  
9114 023364 012737 001015 177746  INC #GOOD  
9115 023372 023757 004550 004552  CMP R3,#HIGHSP+2000  
9116 023400 001406  BEO 12S  
9117 023402 005737 004552  TST #BAD  
9118 023406 001413  BEQ 13S  
9119 023410 005737 004550  TST #GOOD  
9120 023414 001410  BEQ 13S  
9121 023416 063020 12S: ASL R2  
9122 023420 032702 001000  BIT #BIT09,R2  
9123 023424 001672  BEQ 11S  
9124 023426 004037 005012  JSR R0,#LPOINTS  
9125 023432 004000  NEXT TEST  
9126 023434 000015  BR 1ST106  
9127 023436 010237 004542 13S: MOV R2,#ERROR  
9128 023442 006337 004542  ASL #ERROR  
9129 023446 005237 004560  INC #BITTLG  
9130 023452 004037 003532  JSR R0,#SEN16  
9131 023456 000006 006  
9132 023460 047044  .WORD SEN13  
9133 023462 041010  .WORD SEN16  
9134 023464 041000  .WORD SEN15  
9135 023466 000753  BR 12S  
9136 000000 000000  BR 12S  
```
CATCH EASIOG. MACYI13 30A(1052) 31-OCT-79 15:29 PAGE 72-2
(FKXAK_A. P/ 25-JAN-79 13:31) BYPASS MODE TESTS

9141
9142
9143
9144
9145
9146
9147
9148
9149
9150
9151
9152
9153
9154
9155
9156
9157
9158
9159
9160
9161
9162
9163
9164
9165
9166
9167
9168
9169
9170
9171
9172
9173
9174
9175
9176
9177
9178
9179
9180
9181
9182
9183
9184
9185
9186
9187
9188
9189
9190
9191
9192
9193
9194
9195
9196

9141
9142
9143
9144
9145
9146
9147
9148
9149
9150
9151
9152
9153
9154
9155
9156
9157
9158
9159
9160
9161
9162
9163
9164
9165
9166
9167
9168
9169
9170
9171
9172
9173
9174
9175
9176
9177
9178
9179
9180
9181
9182
9183
9184
9185
9186
9187
9188
9189
9190
9191
9192
9193
9194
9195
9196

\$LTLL BYPASS MODE TESTS

: VERIFY THAT A WRITE TO MEMORY WHILE CACHE IS IN
BYPASS MODE WILL INVALIDATE THAT LOCATION

TST104: INC TID ;UPDATE TEST ID
JSR R4, RELCTL ;RELOCATE TEST TO LOW CACHE
WORD TST110
BIT #BIT13, A, CCR ;SELECT VALID SET A
BEQ 18 ;SET A IN USE NOW
BIT #BIT12, A, CCR ;SELECT VALID SET B
BNE 26 ;WAIT TILL COMPLETE
26: BIT #BIT12, A, CCR ;SELECT VALID SET B
BNE 26 ;WAIT TILL COMPLETE
18: MOV #R5, A, CCR ;ENABLE HIGH CACHE
MOV #HIGHSP, R3 ;TAG HIGH LOCATION
MOV #HIGHSP, R3 ;ENTER BYPASS MODE
MOV #HIGHSP, R3 ;INVALIDATE LOCATION
MOV #HIGHSP, R3 ;VERIFY LOCATION INVALIDATED
MOV #OFF, A, CCR ;DISABLE CACHE
CLR #ERROR ;RESET ERROR FLAG

TST105: INC TID ;UPDATE TEST ID
JSR R4, RELCTL ;RELOCATE TEST TO LOW CACHE
WORD TST110
BIT #BIT13, A, CCR ;SELECT Valid BITS SET B
BNE 18 ;IN USE NOW
BIT #BIT12, A, CCR ;SELECT Valid BITS B
BNE 26 ;WAIT TILL COMPLETE
26: BIT #BIT12, A, CCR ;SELECT Valid BITS B
BNE 26 ;WAIT TILL COMPLETE
18: MOV #R5, A, CCR ;Enable HIGH CACHE
MOV #HIGHSP, R3 ;TAG HIGH LOCATION
MOV #HIGHSP, R3 ;ENTER BYPASS MODE
MOV #HIGHSP, R3 ;INVALIDATE LOCATION
MOV #HIGHSP, R3 ;VERIFY LOCATION INVALIDATED
MOV #OFF, A, CCR ;DISABLE CACHE
CLR #ERROR ;RESET ERROR FLAG
CACHE DIAG. MACY11 30A(1052) 31-OCT-79 15:29 PAGE 72-3
CFKAB.P11 25-JUN-79 13:31 BYPASS MODE TESTS

9197 023764 032703 000004 BIT #BIT02,R3 LOOK FOR HIT
9198 023770 001411 BEQ 3% NO HIT MEANS NO ERRORS
9199 023772 004542 INC #ERROR SET ERROR FLAG
9200 023776 004037 005352 JSR RO,#SETEN REPORT ERROR
9201 024002 000010 .WORD #D8
9202 024004 041030 .WORD #SEN17 BYPASS MODE TESTS
9203 024006 041040 .WORD #SEN18 HIGH CACHE LOCATION NOT INVALIDATED
9204 024010 041054 .WORD #SEN19 BY WRITE TO LOCATION IN BYPASS MODE
9205 024012 041114 .WORD #SEN121 USING VALID BITS SET B
9206 024014 004010 3% JSR RO,(RO) TAKE SELECTED ACTION ON ERROR
9207 024016 044000 .WORD LOWSP LOOP ON ERROR
9208 024020 044000 .WORD LOWSP LOOP ON TEST
9209 024022 012737 001015 177746 MOV #OFF,#CCCH DISABLE CACHE
9210
9211
9212
9213
9214
9215
9216 024030 005267 154612 VERIFY THAT A READ TO MEMORY WHILE IN
9217 024034 004467 162054 BYPASS MODE WILL INVALIDATE CACHE
9218 024040 024162 TST106: INC TID UPDATE TEST ID
9219 024042 012737 000005 JSR RO,#RELCTL RELOCATE TEST TO LOW CACHE
9220 024050 013703 046000 .WORD #TST107 BYPASS MODE TESTS
9221 024054 052737 001000 .WORD #SEN17 HIGH CACHE LOCATION NOT INVALIDATED
9222 024062 013703 046000 MOV #OFF,#CCCH BY WRITE TO LOCATION IN BYPASS MODE
9223 024066 042737 001000 MOV #OFF,#CCCH USING VALID BITS SET B
9224 024074 013703 046000 MOV #OFF,#CCCH HIGH CACHE LOCATION NOT INVALIDATED
9225 024100 013703 177752 MOV #OFF,#CCCH BYPASS MODE TESTS
9226 024104 012737 001015 177746 MOV #OFF,#CCCH HIGH CACHE LOCATION NOT INVALIDATED
9227 024112 020537 004542 MOV #OFF,#CCCH BYWRITE TO LOCATION IN BYPASS MODE
9228 024116 032703 000004 MOV #OFF,#CCCH USING VALID BITS SET B
9229 024122 001411 MOV #OFF,#CCCH HIGH CACHE LOCATION NOT INVALIDATED
9230 024124 012737 000001 004542 BIT #BIT02,R3 TAKE SELECTED ACTION ON ERROR
9231 024132 004037 005352 JSR RO,#SETEN REPORT ERROR
9232 024136 000006 .WORD #D6 NO HIT MEANS NO ERRORS
9233 024140 041030 .WORD #SEN117 BYPASS MODE TESTS
9234 024142 041130 .WORD #SEN122 HIGH CACHE LOCATION NOT INVALIDATED
9235 024144 041146 .WORD #SEN123 BYWRITE TO LOCATION IN BYPASS MODE
9236 024146 040100 15: JSR RO,(RO) TAKE SELECTED ACTION ON ERROR
9237 024150 044000 .WORD LOWSP LOOP ON ERROR
9238 024152 044000 .WORD LOWSP LOOP ON TEST
9239 024154 012737 001015 177746 MOV #OFF,#CCCH DISABLE CACHE
9240
9241
9242
9243
CACHE DIAG.  
MAY11 30A(1052) 31-OCT-79 15:29 PAGE 72-4

TAG BIT BASIC READ WRITE TESTS

TST07:  INC TID  
        JSR R4,RELCH  
        ;UPDATE TEST ID
        ;RELOCATE TEST TO HIGH CACHE
        .WORD TST07
        MOV #11, @WCR  
        ;ENABLE LOW CACHE
        MOV $1000,$R3  
        ;TAG LOW CACHE LOCATION
        MOV $A0,$R3  
        ;TAG LOW 1K
        MOV $A0,$R3  
        ;READ HIT TO LOW 1K
        MOV @CHR,$ERROR  
        ;SAVE HIT REGISTER CONTENTS
        MOV #OFF, @CCR  
        ;DISABLE CACHE
        BI MOVI, @ERROR  
        ;VERIFY HIT
        BNE 1S  
        ;LOCATION WAS HIT
        MOV #1, @ERROR  
        ;SET ERROR FLAG
        JSR R0, @SENTEN  
        ;REPORT ERROR
        .WORD *6
        .WORD SENT1  
        ;TAG BIT BASIC READ WRITE TESTS
        .WORD SENT124  
        ;NO HIT FROM READ INTO MEMORY LOCATION
        .WORD SENT125  
        ;ADDRESS 000000
        .WORD HIGHS  
        ;TAKE SELECTED ACTION ON ERROR
        .WORD HIGHS+  
        ;LOOP ON ERROR
        .WORD HIGHS+  
        ;LOOP ON TEST
        BR TST110

TST110:  BIC #777, @ERROR  
         ;MASK OFF HIT BITS
        MOV @0, @ADD  
         ;FAILING ADDRESS
        MOV @ERROR, @BAD  
         ;BAD DATA
        MOV #0, @BAD  
         ;GOOD DATA
        INC @GOODDD  
         ;ERROR PRINT MODE
        JSR R0, @SENTEN  
         ;REPORT ERROR
        .WORD *6
        .WORD SENT124  
         ;TAG BIT BASIC READ WRITE TESTS
        .WORD SENT127  
         ;WRONG TAG READ FROM HIT REGISTER
        .WORD SENT128  
         ;AS RESULT OF MEMORY READ
        BR 2S

TST124:  .VERIFY TAG BIT11 CAN BE WRITTEN A ONE
        ;WITH ALL OTHER TAG BITS WRITTEN ZERO

TST110:   INC TID  
           JSR R4,RELCH  
           ;UPDATE TEST ID
           ;RELOCATE TEST TO HIGH CACHE
           .WORD TST110
           MOV #11, @WCR  
           ;ENABLE LOW CACHE
           MOV $1000,$R3  
           ;TAG LOW CACHE LOCATION
           MOV $A000,$R3  
           ;TAG LOW 2K
           MOV $A000,$R3  
           ;READ HIT TO LOW 2K
           MOV @CHR,$ERROR  
           ;SAVE HIT REGISTER CONTENTS
           MOV #OFF, @CCR  
           ;DISABLE CACHE
           BI MOVI, @ERROR  
           ;VERIFY HIT
           BNE 1S  
           ;LOCATION WAS HIT
           MOV #1, @ERROR  
           ;SET ERROR FLAG
           JSR R0, @SENTEN  
           ;REPORT ERROR
           .WORD *6
           .WORD SENT1  
           ;TAG BIT BASIC READ WRITE TESTS
           .WORD SENT124  
           ;NO HIT FROM READ INTO MEMORY LOCATION
9413 025204 012737 000001 004542
9414 025212 004637 003352
9415 025216 000006
9416 025220 041160
9417 025222 041176
9418 025226 047516
9419 025226 004010
9420 025230 046000

MOV #1,B#ERROR
MOV #1,B#SETEN

call *D6

.JSR RO.RO

.:SET ERROR FLAG
.:REPORT ERROR
.:TAG BIT BASIC READ WRITE TESTS
.:NO HIT FROM READ INTO MEMORY LOCATION
.:ADDRESS 040000
.:TAKE SELECTED ACTION ON ERROR
.:LOOP ON ERROR

.word HIGHSP
THIS TEST WILL ABORT IF LESS THAN 20K OF MEMORY
VERIFY TAG BITS CAN BE WRITTEN A ONE
WITH ALL OTHER TAG BIT WRITTEN TO ZERO
CACHE DIAG. MCRY11 30A(1052) 31-OCT-79 15:29 PAGE 73-3
CFKAB.P11 25-JUN-79 13:31 TAG BIT BASIC READ WRITE TESTS

9590 026276 013703 044000 MOV @LOWSP, R3 :TAG LOW CACHE
9591 026276 013703 100000 MOV @100000, R3 :TAG LOW 64K
9592 026306 013703 100000 MOV @0100000, R3 :READ HIT TO LOW 64K
9593 026312 013737 177752 004542 MOV @CHR, #ERROR :SAVE HIT REGISTER CONTENTS
9594 026320 012737 001013 177746 MOV #OFF, #CCR :DISABLE CACHE
9595 026326 032737 000004 004542 BIT @BIT02, #ERROR :VERIFY HIT
9596 026334 000107 $E 13
9597 026336 012737 000001 004542 MOV #1, #ERROR :SET ERROR FLAG
9598 026344 004037 005352 JSR RO, @SETEN :REPORT ERROR
9599 026350 000006 .WORD "06"
9600 026352 041160 .WORD SEND124 :TAG BIT BASIC READ WRITE TESTS
9601 026354 041176 .WORD SEND125 :NO HIT FROM READ INTO MEMORY LOCATION
9602 026356 041340 .WORD SEND136 :ADDRESS 400000
9603 026360 004010 2$: JSR RO, (RO) :TAKE SELECTED ACTION ON ERROR
9604 026362 046000 .WORD HIGHSP :LOOP ON ERROR
9605 026364 046000 .WORD HIGHSP :LOOP ON TEST
9606 026366 005037 177572 CLR @#SRO :DISABLE MEMORY MANAGEMENT
9607 026372 000427 BR TST117
9608 026374 042737 000777 004542 8$: BIC #777, #ERROR :MASK OFF HIT BITS
9609 026402 012737 000000 004554 MOV #0, #ADD :MEMORY ADDRESS
9610 026410 013737 004542 004552 MOV @ERROR, #BAD :BAD DATA
9611 026467 012737 100000 004550 MOV @100000, #GOOD :GOOD DATA
9612 026474 012737 100000 004542 SUB @100000, #ERROR :IF NAY BITS LEFT SET THEN ERROR
9613 026482 004037 005352 JSR RO, @SETEN :REPORT ERROR
9614 026496 000010 .WORD "08"
9615 0264A2 041160 .WORD SEND124 :TAG BIT BASIC READ WRITE TESTS
9616 0264A4 041224 .WORD SEND127 :WRONG TAG READ FROM HIT REGISTER
9617 0264A6 041264 .WORD SEND130 :MEMORY ADDRESS AND HIT REGISTER DATA
9618 0264A8 041340 .WORD SEND136 :ADDRESS 400000
9619 0264B0 000743 BR 2$
CACHE DIAG. \( \text{MACY} \text{1130A(1052)} \) 31-OCT-79 15:29 PAGE 73-6

UNIBUS EXERCISER DMA TEST

; SBTL UNIBUS EXERCISER DMA TEST
; CHECK THAT DMA WRITE INVALIDATES CACHE

9738 027204 005267 151436
9741 027210 004667 156700
9743 027214 027442
9744 027216 005037 004540
9745 027222 005757 170006
9746 027226 004537 004540
9747 027232 001402
9748 027234 000167 000202
9749 027240 005737 000650
9750 027244 001004
9751 027254 012702 004562
9752 027256 000237 005066
9753 027256 102737 000000 177746
9754 027264 012701 052000
9755 027270 012102
9756 027272 021027 054000
9757 027276 001374
9758 027300 012701 046000
9759 027304 012102
9760 027306 201270 050000
9761 027312 001374
9762 027316 012737 046000 170004
9763 027322 012737 177000 170002
9764 027330 012737 177777 170000
9765 027336 012737 000000 170016
9766 027336 012737 003045 170006
9767 027352 102737 170006
9768 027356 100375
9769 027360 012701 046000
9770 027364 012102
9771 027366 053757 177752 004542
9772 027374 020127 050000
9773 027400 0011371
9774 027402 042737 177700 004542
9775 027410 012737 001015 177746
9776 027416 004637 005352
9777 027422 000040
9778 027424 041412
9779 027426 041424
9780 027430 037330
9781 027432 041446
9782 027436 006010
9783 027436 044000
9784 027440 044000
9785
9786
9787
9788
9789
9790
9791
9792
9793

SBTL DATA BIT MARCH PATTERN TEST
; TEST OPERATES IN HIGH CACHE FOR TESTING LOW CACHE
; TEST WRITES BACKGROUND OF ALL ZEROS
; READ A LOCATION STARTING AT CACHE ADDRESS 0000
; COMPLEMENTS DATA
WRITES THE COMPLEMENT TO CACHE
READS COMPLEMENTED DATA AND ERROR CHECKS
PROCEED TO NEXT CACHE LOCATION
AFTER ALL HIGH CACHE HAS BEEN WRITTEN TO COMPLEMENT TEST THEN STARTS READING FROM LAST LOCATION
COMPLEMENTS DATA
WRITES COMPLEMENTED DATA TO CACHE
READS AND ERROR CHECKS DATA
PROCEEDS TO NEXT LOWER LOCATION
UNTIL START OF HIGH CACHE

TST121: INC TID ;UPDATE TEST ID
          JSR R6, RELTH ;RELOCATE TEST TO HIGH CACHE
          .WORD TST122
          CLR @ERROR ;RESET ERROR FLAG
          CLR @LOPERR ;RESET LOOP ON TEST FLAG
          MOV #11, W, CR ;ENABLE LOW CACHE
          MOV #LOW, R1 ;TAG LOW BLOCK #1
          MOV (R1)+, R3 ;CAUSE READ TO LOW BLOCK
          CMP (R1)+,2000, R1
          BNE 18
          MOV @LOW, R1 ;WRITE BACKGROUND TO LOW CACHE
          CMP R1, LOW+2000
          BNE 28
          CLR (R1)+ ;CAUSE READ TO TAG, THEN WRITE BACKGROUND
          MOV @LOW, R1
          MOV #LOW, R1 ;ADDRESS OF TEST BLOCK
          MOV #HIGH, R2 ;ADDRESS OF ERROR BLOCK
          CLR @CRPE ;RESET ERROR REGISTER
          MOV (R1)+, @LOW ;READ BACKGROUND SAVE ERROR INFO
          BIS #215, @CCR ;SET ABORT FOR ERROR READ
          MOV @CRPE, (R1)+ ;SAVE ERROR INFO.
          BNE 38 ;DISABLE ABORT
          MOV @LOW, R1 ;WRITE BACKGROUND
          MOV #HIGH, R2 ;
          MOV TST (R2) ;ANY ERROR FOR LOCATION
          BNE 48 ;YES
          ADD #2, R2 ;POINT TO NEXT
          ADD #2, R1
          CMP R1, LOW+2000
          BNE 33
          BR 10S
          BIT #BIT05, (R2) ;ANY TAG ERROR
          BEQ 58
          MOV 4000(R1), R3 ;UNITAG LOCATION
          BIS #BIT10, @CCR ;ENABLE WRITE WRONG TAG
          MOV (R1)+, R3 ;TAG LOCATION WRONG
          BIS #BIT10, @CCR ;DISABLE WRITE WRONG TAG
          BIT #BIT06, (R2) ;ANY LOW BYTE FAILURE
          BEQ 68 ;NO
          BIS #BIT06, (R2) ;ENABLE WRITE WRONG LOW BYTE
          MOV #0, (R1) ;WRITE WRONG DATA LOW BYTE
          BIT #BIT06, (R2) ;DISABLE WRITE DATA LOW BYTE
          BIT #BIT07, (R2) ;ANY HIGH BYTE FAILURE
<table>
<thead>
<tr>
<th>Address</th>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10018</td>
<td>030750 011411</td>
<td>BEQ 58</td>
</tr>
<tr>
<td>10019</td>
<td>030752 016103</td>
<td>MOV 4000(R1),R3, R3</td>
</tr>
<tr>
<td>10020</td>
<td>030756 052737</td>
<td>BIS #1B10, &amp;CCR</td>
</tr>
<tr>
<td>10021</td>
<td>030764 011103</td>
<td>MOV (R1), R3</td>
</tr>
<tr>
<td>10022</td>
<td>030766 042737</td>
<td>MOV #0, (R1)</td>
</tr>
<tr>
<td>10023</td>
<td>030767 032712</td>
<td>MOV (R1), (R2)</td>
</tr>
<tr>
<td>10024</td>
<td>031000 001610</td>
<td>ANY LOW BYTE FAILURE</td>
</tr>
<tr>
<td>10025</td>
<td>031002 052737</td>
<td>BIT #1B106, (R2)</td>
</tr>
<tr>
<td>10026</td>
<td>031010 011211</td>
<td>ANY HIGH BYTE FAILURE</td>
</tr>
<tr>
<td>10027</td>
<td>031014 042737</td>
<td>MOV #0, (R1)</td>
</tr>
<tr>
<td>10028</td>
<td>031022 032712</td>
<td>MOV #0, (R1)</td>
</tr>
<tr>
<td>10029</td>
<td>031036 001313</td>
<td>WRITE WRONG DATA</td>
</tr>
<tr>
<td>10030</td>
<td>031036 011261</td>
<td>WRITE WRONG DATA HIGH BYTES</td>
</tr>
<tr>
<td>10031</td>
<td>031036 011261</td>
<td>WRITE WRONG DATA LOW BYTES</td>
</tr>
<tr>
<td>10032</td>
<td>031044 011273</td>
<td>MOV #0, (R1)</td>
</tr>
<tr>
<td>10033</td>
<td>031052 011003</td>
<td>MOV #OFF, &amp;CCCR</td>
</tr>
<tr>
<td>10034</td>
<td>031054 014331</td>
<td>MOV PC,R3</td>
</tr>
<tr>
<td>10035</td>
<td>031056 020237</td>
<td>MOV -(R3), (R3)</td>
</tr>
<tr>
<td>10036</td>
<td>031062 001374</td>
<td>CORRECT WRONG PARITY LOW</td>
</tr>
<tr>
<td>10037</td>
<td>031064 021373</td>
<td>CMP R3,#4+$-TST122-12+$LOWSP</td>
</tr>
<tr>
<td>10038</td>
<td>031072 000714</td>
<td>BNE 85</td>
</tr>
<tr>
<td>10039</td>
<td>031074 012700</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10040</td>
<td>031100 012702</td>
<td>MOV # renovations</td>
</tr>
<tr>
<td>10041</td>
<td>031104 011103</td>
<td>MOV (R1), R3</td>
</tr>
<tr>
<td>10042</td>
<td>031106 005103</td>
<td>MOV R3,(R1)</td>
</tr>
<tr>
<td>10043</td>
<td>031110 010311</td>
<td>MOV R3,(R1)</td>
</tr>
<tr>
<td>10044</td>
<td>031112 005013</td>
<td>CLR #ACME</td>
</tr>
<tr>
<td>10045</td>
<td>031116 011112</td>
<td>MOV (R1),(R2)</td>
</tr>
<tr>
<td>10046</td>
<td>031120 005273</td>
<td>MOV #1ACME</td>
</tr>
<tr>
<td>10047</td>
<td>031124 001010</td>
<td>TST #ACME</td>
</tr>
<tr>
<td>10048</td>
<td>031126 020279</td>
<td>BNE 125</td>
</tr>
<tr>
<td>10049</td>
<td>031136 005000</td>
<td>ADD #2,R1</td>
</tr>
<tr>
<td>10050</td>
<td>031156 020127</td>
<td>ADD #2,R2</td>
</tr>
<tr>
<td>10051</td>
<td>031159 007360</td>
<td>CMP R1,#HIGHSP+2000</td>
</tr>
<tr>
<td>10052</td>
<td>031144 004533</td>
<td>BNE 115</td>
</tr>
<tr>
<td>10053</td>
<td>031146 052737</td>
<td>MOV #15, &amp;CCCR</td>
</tr>
<tr>
<td>10054</td>
<td>031154 013705</td>
<td>MOV #15, &amp;CCCR</td>
</tr>
<tr>
<td>10055</td>
<td>031160 027373</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10056</td>
<td>031166 032705</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10057</td>
<td>031172 001374</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10058</td>
<td>031174 052737</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10059</td>
<td>031202 011273</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10060</td>
<td>031204 027373</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10061</td>
<td>031212 032705</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10062</td>
<td>031216 001374</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10063</td>
<td>031220 000303</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10064</td>
<td>031222 052737</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10065</td>
<td>031239 011361</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10066</td>
<td>031242 012173</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10067</td>
<td>031242 027373</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10068</td>
<td>031250 002705</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10069</td>
<td>031252 014515</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10070</td>
<td>031254 020279</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10071</td>
<td>031260 001374</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10072</td>
<td>031262 012707</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
<tr>
<td>10073</td>
<td>031270 011112</td>
<td>MOV #5, &amp;CCCR</td>
</tr>
</tbody>
</table>
BR 16S
MOV #OFF, @MCCR
MOV #LOW1, R1
; DISABLE CACHE
; READ AND REPORT ERROR
MOV @ERROR, @BAD
INC @GOODBD
SUB @ERROR, @ADD
JSR RO, @SETEN
"WORD D4
"WORD SEN143
"WORD SEN144
BIS @ERROR, @LOPERR
CMP R1, @LOW
BNE 29$5
MOV @LOPERR, @ERROR
JSR RD, (RO)
"WORD CWSP
"WORD CWSP
:BAD DATA
:SET DATA PRINT MODE
:CACHE ADDRESS
:REPORT ERROR IF ANY
:DATA BIT MARCH PATTERN TEST
:CACHE ADDRESS, DATA EXPECTED, DATA NEAC
:DEL ERROR LOCATION
:TAKE SELECTFED ACTION ON ERROR
:LOOP ON ERROR
:LOOP ON TEST
SFBTL DATA PARITY MARCH PATTERN TEST

PATTERN RUNNING IN LOW CACHE PARITY

10150 INC TID : UPDATE TEST ID

10151 JSR R4,RELCHR : RELOCATE TEST TO HIGH CACHE

10152 LDR TST124

10153 CLR @ERROR : RESET ERROR FLAG

10154 MOV #LOWSP,R1 : FIRST ADDRESS OF TEST BLOCK

10155 MOV #11,#ACC : ENABLE LOW CACHE

10156 MOV #LOWSP,R1 : TEST BLOCK

10157 CLR (R1)+ : WRITE ZERO TO PARITY

10158 CMP R1,#LOWSP+2000

10159 BNE 1S

10160 MOV #LOWSP,R1 : FIRST ADDRESS OF TEST BLOCK

10161 MOV #HIGH1,R3 : ERROR BLOCK POINTER

10162 CLR @CMPE : RESET ERROR REGISTER

10163 MOV (R1),R2 : READ PARITY

10164 TST @CMPE : ANY ERROR

10165 BNE 3S : YES

10166 CLR (R3) : WRITE CMP, PARITY

10167 MOV #GOOD1,(R1) : WRITE CMP, PARITY

10168 MOV (R1),R2 : READ CMP

10169 TST @CMPE : ANY ERROR

10170 BNE 3S : YES

10171 ADD #2,R1 : ADD J2, R3

10172 ADD #2,R3 : ADD J2, R3

10173 MOV #OFF,#ACC : INVALIDATE HIGH CACHE

10174 MOV PC,R2 : INVALDATE HIGH CACHE

10175 CMP R2,#25-TST123-12+HIGHSP

10176 BNE 30S

10177 MOV #11,#ACC : DATA ADDRESS

10178 MOV #LOWSP,R1 : ERROR REGISTER SAVE BLOCK

10179 MOV #HIGH1,R3 : DISABLE CACHE

10180 MOV #OFF,#ACC : DISABLE CACHE

10181 TST (R3) : ANY ERROR FOR LOCATION

10182 BNE 21$ : YES

10183 ADD #2,R1 : UPDATE POINTERS

10184 ADD #2,R3 : UPDATE POINTERS

10185 CMP R1,#LOWSP+2000

10186 BNE 23$: YES

10187 BR 10$: ENABLE ABORT FOR ERROR READ

10188 MOV #CMPE,(R3) : SAVE ERROR

10189 MOV #11,#ACC : DISABLE ABORT

10190 BR 4$: ENABLE ABORT FOR ERROR READ

10191 CLR @GOOD

10192 CLR @BAD

10193 TST (R1) : SHOULD PARITY BIT BE SET

10194 BNE 5$: NO

10195 MOV #1,#GOOD : PARITY BIT SHOULD HAVE BEEN SET

10196 BR 6$: ENABLE ABORT FOR ERROR READ
ACME DIAG. MACH 72 304-1052 31-JUL-79 15:29 PAGE 74-3

FXKAB.P1 25-JUN-79 13:31

DATA PARITY MARCH PATTERN TEST

1026 032174 012737 000001 004552 5$: MOV #1, @BAD
1027 032202 010137 004554
1028 032206 040000 004554
1029 032214 005237 004542
1030 032220 005527 004556
1031 032224 032713 000100
1032 032230 000115
1033 032232 032713 000200
1034 032236 000160
1035 032240 004037 005352
1036 032244 000006
1037 032266 041516
1038 032260 041516
1039 032252 041544
1040 032254 053737 004542 005064
1041 032262 000711
1042 032266 004037 005352
1043 032270 000006
1044 032272 041536
1045 032274 041566
1046 032276 041544
1047 032300 053737 004542 005064
1048 032306 000677
1049 032310 013737 005064 004542
1050 032316 004010
1051 032320 046000
1052 032322 046000
1053 032324 052671 146316
1054 032328 004467 153560
1055 032334 032742
1056 032336 005037 005064
1057 032342 005037 004542
1058 032346 012701 046000
1059 032352 012737 000005 177746
1060 032360 012701 046000
1061 032364 005021
1062 032366 020157 050000
1063 032372 000134
1064 032374 012701 046000
1065 032400 012703 050000
1066 032404 005037 177744
1067 032410 014717 177744
1068 032412 005537 177744
1069 032416 000102
1070 032420 005013
1071 032422 012711 004001
1072 032426 010102
1073 032430 005537 177744
1074 032434 001063
1075 032436 062701 000002
1076 032442 062703 000002
1077 032446 012737 001015 177746
1078 032450 010702

TST124: INC TID
JSR R4.RELCTL
.word TST125
CLR @LOPERR
MOV #HIGHSP,R1
MOV $5,#CCR
MOV #HIGHSP,R1
MOV $5,#CCR
1$: CLR (R1)+
cmp R1,#HIGHSP+2000
bne 15
MOV #HIGHSP,R1
MOV #LOW1,R3
MOV #LOW1,R3
2$: CLR @CMP
MOV (R1),A2
TST @CMP
bne 15
 CLR (R3)
MOV #G001,(R1)
MOV (R1),A2
TST @CMP
bne 15
 ADC #2,R3
MOV @OFF,#CCR
MOV PC,R2

PATTERN RUNNING IN HIGH CACHE

UPDATE TEST ID
RELOCATE TEST TO LOW CACHE
RESET ERROR FLAG
FIRST ADDRESS OF TEST BLOCK
ENABLE HIGH CACHE
TEST BLOCK
WRITE ZERO TO PARITY
FIRST ADDRESS OF TEST BLOCK
ERROR BLOCK POINTER
RESET ERROR REGISTER
READ PARITY
ANY ERROR
YES
YES
ADJ. POINTER
INVALIDATE LOW CACHE
LACME DIAG.  MACY1: 30A (1052)  31-OCT-79  15:29  PAGE 74-4

DATA PARITY MARCH PATTERN TEST

30$:  MOV -(R2), (R2)
       CMP R2, #25 - TST 124 - 12 + LOWSP
       BNE 50$

25$:  MOV #5, @MCCR
       CMP R1, #HIGHSP + 2000
       BNE 25$

22$:  BIS #215, @MCCR
       MOV @MPE, (R3)
       MOV #215, @MCCR
       MOV #5, @MCCR
       MOV R1, #HIGHSP + 2000

21$:  TST (R3)
       BNE 22$

20$:  ADD #2, R1
       ADD #2, R3

19$:  CMP R1, #HIGHSP + 2000
       BNE 18$

18$:  BR 10$

17$:  BIS #215, @MCCR
       MOV @MPE, (R3)
       MOV #215, @MCCR
       MOV #5, @MCCR
       MOV R1, #HIGHSP + 2000

16$:  CLR @MGOOD
       CLR @MBAD
       TST (R1)
       BEO 5$

14$:  MOV #1, @MGOOD
       PARITY BIT SHOULD HAVE Bin SET

13$:  BR 6$

12$:  MOV #1, @MBAD
       PARITY BIT SHOULDN'T BE SET

11$:  MOV R1, @MADD
       MEMORY ADDRESS

10$:  SUB #40000, @MADD
       CACHE ADDRESS

9$:  INC @ERROR
       ERROR FLAG

8$:  INC @GOODBAD
       DATA PRINT MODE

7$:  BIT @BIT06, (R3)
       WAS IT LOW BYTE ERROR

6$:  BNE 7$

5$:  BIT @BIT170, (R3)
       WAS IT HIGH BYTE

4$:  BEO 4$

3$:  JSR R0, @MSETEN
       REPORT ERROR

2$:  JSR R0, @MSETEN
       REPORT ERROR

0$:  JSR R0, @MSETEN
       REPORT ERROR
TESTING LOW CACHE USING BOTH SETS OF VALID BITS

TST125: INC TID :UPDATE TEST ID
          JSR R4, RELCNH :RELOCATE TEST TO HIGH CACHE
          LDR TST126 :WORD TST126
          CLR R5
          20$: CLR @LPOPERR :RESET HAD ERROR FLAG
          CLR @LERROR :RESET ERROR FLAG
          MOV #LWSP, R1 :WRITE VALID BIT TO ZERO
          1$: MOV #0, (R1) :READ VALID WRITE COMP.
          CMP R1, #LWSP+2000 :LOOK FOR HIT
          BNE 1$: MOV #HIGH1, R3 :READ DID CAUSE HIT, IT SHOULDN'T HAVE
          MOV #LWSP, R1 :READ COMP.
          LDR #11, @WCR
          MOV R1, R2 :IS VALID SET
          BIT #BIT03, R1, @CHR
          BNE 7$: BIT #BIT03, @CHR
          BEQ 5$: BIT #BIT03, @CHR
          5$: CMP R1, #LWSP+2000 :IS VALID SET
          BNE 5$: CMP R1, #LWSP+2000
          BR 1$: CMP R1, #LWSP+2000
          MOV #CCCR+1, (R3) :SAVE VALID SET IN USE BIT
          MOV #0, (R3) :SAVE DATA WRITTEN
          TST (R3) :ADD POINTER
          BR 5$: CONTINUE
          3$: MOV #CCCR+1, (R3) :SAVE VALID SET IN USE
          MOV #1, (R3) :SAVE DATA WRITTEN
          TST (R3) :ADD POINTER
          BR 5$: CONTINUE
          11$: MOV #OFF, #CCCR :DISABLE CACHE
          MOV #HIGH1, R3 :ERROR DATA BLOCK
          12$: TST (R3) :ANY ERROR
          BNE 13$: TST (R3)
          16$: CMP R3, #HIGH1+2000 :ANY ERROR
          BNE 12$: CMP R3, #HIGH1+2000
          BR 10$: CMP R3, #HIGH1+2000
          13$: CLR @GOOD :FIND WHAT WENT WRONG
          CLR @BAD :FIND DATA WRITTEN
          TSTB -2(R3) :DATA WRITTEN WAS ONE
          BNE 14$: TSTB -2(R3)
          MOV #1, @BAD :DATA READ WAS ONE
          MOV #1, @BAD :DATA READ WAS ONE
          BR 15$: MOV #1, @BAD
          14$: MOV #GOOD :MEMORY ADDRESS
          MOV #GOOD :MEMORY ADDRESS
          15$: MOV R3, @ADD :CACHE ADDRESS
          MOV R3, @ADD :CACHE ADDRESS
          SUB #HIGH1+2, @ADD :SET ERROR FLAG
          INC @GOOD :SET DATA PRINT MODE
          MOV #1, @ERROR :SET DELAY ERROR FLAG
          INC @ERROR :SET DELAY ERROR FLAG
          BIT #BIT1, 2(R3) :IS THIS VALID SET B
          BEQ 4$: BIT #BIT1, 2(R3)
          4$: 


CACH DIAG MAC 31 30A(1052) 31-OCT-79 15:29 PAGE 74-6
CPKAB.P-11 25-JUN-79 13:31
VALID BIT MARCH PATTERN TEST

SEQ 0114

10372 033240 004037 005352  JSR RO,#SETEN :REPORT ERROR
10375 033244 000006  JSR RO,$D6 :VALID BIT MARCH PATTERN TEST
10376 033246 001600  .WORD SEN149 :VALID BIT SET B INUSE
10376 033250 001614  .WORD SEN70 :VALID BIT SET IN USE
10376 033252 001630  .WORD SEN151 :CACHE ADDRESS, VALID BIT WRITTEN, VALID BIT R
10377 033254 000072  BR 16$: :REPORT ERROR
10378 033256 004037 005352 4$: JSR RO,#SETEN :VALID BIT MARCH PATTERN TEST
10379 033262 000000  JSR RO,$D6 :VALID BIT MARCH PATTERN TEST
10380 033264 001600  .WORD SEN149 :VALID BIT SET B INUSE
10381 033266 001656  .WORD SEN70 :VALID BIT SET IN USE
10382 033270 001672  .WORD SEN151 :CACHE ADDRESS, VALID BIT WRITTEN, VALID BIT R
10383 033272 000072  BR 16$: :REPORT ERROR
10384 033274 012757 001015 177746 10$: MOV #OFF,#CCR :DISABLE CACHE
10385 033302 013737 000064 004542  MOV $LOPERR,#ERROR :JSR RO,RO :TAKE SELECTED ACTION ON ERROR
10386 033310 004010  :LOOP ON ERROR
10387 033312 046000  :LOOP ON TEST
10388 033314 046000  :JS THIS FIRST PASS
10389 033316 005705  TST R5 :BNE TST126 :NO CONTINUE TO NEXT TEST
10390 033320 001005  BIS #II107,#CCR :FLUSH CACHE SELECT OTHER VALID SET
10391 033322 052737 000400 177746  INC R5
10392 033330 005205  BR 20$: :TESTING HIGH CACHE USING BOTH SETS OF VALID Bits
10393 033332 006611 :UPDATE TEST ID
10394
10395
10396
10397
10398
10399
10400
10401
10402 033334 005267 165306  TST126:   INC TID :RELOCATE TEST TO LOW CACHE
10403 033340 004467 152550  JSR R4,RELCTL :VALID CHANGE FLAG
10404 033344 033330  .WORD TST127
10405 033346 005005  CLR R5 :CLR $LOPERR :RESET BAD ERROR FLAG
10406 033350 005037 005064 :CLR #ERROR :RESET ERROR FLAG
10407 033354 005037 004542 :MOV #HICHP,R1 :WRITE VALID BIT TO ZERO
10408 033360 012701 064000 1$: MOV #O,(R1)+
10409 033364 012721 000000  CMP R1,#HICHP+2000
10410 033370 020127 050000  BNE 15$:
10411 033374 001373  MOV #LW1,R3
10412 033376 012703 050000 :MOV #HICHP,R1
10413 033402 012701 064000 :MOV #S,#CCR
10414 033406 012737 000000 177746 2$: MOV (R1),R2
10415 033416 011102 000000 :BIT #BIT03,#CCR
10416 033416 032737 000010 177746 :BNE 7$:
10417 033424 001012  MOV (R1),R2
10418 033426 012102 :BIT #BIT03,#CHR
10419 033430 032737 000010 177752 :IS VALID SET
10420 033436 001614  BEQ 3$: :NO
10421 033440 005023  CLR (R3)+
10422 033442 020127 050000 5$: CMP R1,#HICHP+2000
10423 033446 001357  BNE 2$: :BNE 11$:
10424 033450 006416  BR 11$: :MOVB #CCR+1,(R3)
10425 033452 113763 177747 000001 :SAVE VALID SET INUSE BIT
10426 033460 112713 000000 :SAVE DATA WRITTEN
CACHE DIAG. MACT11 304(1052) 31-OCT-79 15:29 PAGE 75
(FUKAS.P1) 25-JUN-79 13:31 VALID BIT MARCH PATTERN TEST

01428 033464 005723
01429 033466 000765
01430 033470 117363 177747 000001
01431 033476 112713 000001
01432 033502 005723
01433 033504 000756
01434 033506 010737 001015 177746
01435 033514 012203 050000
01436 033520 005723
01437 033522 001004
01438 033524 020327 052000
01439 033530 000173
01440 033532 000456
01441 033534 005037 004550
01442 033540 000537 004552
01443 033544 105763 177776
01444 033550 001004
01445 033552 012737 000001 004552
01446 033560 000403
01447 033562 012737 000001 004550
01448 033570 010337 004554
01449 033574 162737 046002 004554
01450 033602 012757 000001 004542
01451 033610 005237 004556
01452 033614 053737 004542 005064
01453 033622 032763 020000 177776
01454 033630 001410
01455 033632 004037 005352
01456 033636 000000
01457 033640 041600
01458 033642 041614
01459 033644 041630
01460 033646 005205
01461 033650 000725
01462 033652 004037 005352
01463 033656 000006
01464 033660 041600
01465 033662 041656
01466 033664 041756
01467 033666 000716
01468 033670 012737 001015 177746
01469 033676 013537 005064 004542
01470 033704 004010
01471 033706 004000
01472 033710 004000
01473 033714 005705
01474 033714 005705
01475 033716 052737 000400 177746
01476 033726 000526
01477 033726 000610

TST (R3)+
MOV @CCCR+1,1(R3)
MOV @R1, (R3)
TST (R3)+
BR 5$
MOV @OFF, @CCCR
MOV @LLOW, @R3
TST (R3)+
BNE 13$
CMP @R3, @LLOW+1000
BNE 12$
BR 10$
CLR @GOOD
CLR @BAD
TSTB -2(R3)
BNE 14$
MOV @R1, @BAD
BR 15$
MOV @R1, @GOOD
MOV R3, @ADD
SUB @HIGHSP+2, @ADD
MOV @R1, @ERROR
INC @GOODBD
BIS @ERROR, @LOPERR
BIT @BIT, 13,-2(R3)
BEQ 48
JSR R0, @SETEN
:WORD "D6"
:WORD SENT149
:WORD SENT150
:WORD SENT151
INC R5
BR 16$
JSR R0, @SETEN
:WORD "D6"
:WORD SENT149
:WORD SENT152
:WORD SENT153
BR 16$
MOV @OFF, @CCCR
MOV @LOPERR, @ERROR
JSR R0, (R0)
:WORD LOWSP
:WORD LOWSP
:WORD LOWSP
TST R5
BNE TST127
BIS @BITOB, @CCCR
INC R5
BR 20$

:SAVE VALID SET INUSE BIT
:SAVE DATA WRitten
:DISABLE CACHE
:ERROR DATA BLOCK
:ANY ERROR
:YES
:FINd WHAT WENT WrONG
:FINd DATA WRitten
:DATA WRitten WAS A ONE
:DATA READ WAS A ONE
:DATA READ WAS ZERO
:MEMORY ADDRESS
:CACHE ADDRESS
:Set ERROR FLAG
:SET DATA PRINT MODE
:SET DELAY ERROR FLAG
:IS THIS VALID BIT B
:REPORT ERROR
:VALID BIT MARCH PATTERN TEST
:VALID BIT SET B INUSE
:CACHE ADDRESS , VALID BIT WRITTEN , VALID BIT R
:REPORT ERROR
:VALID BIT MARCH PATTERN TEST
:VALID BIT SET A INUSE
:CACHE ADDRESS , VALID BIT WRITTEN , VALID BIT RE
:DISABLE CACHE
:TAKE SELECTED ACTION ON ERROR
:LOOP ON ERROR
:LOOP ON TEST
:IS THIS FIRST PASS
:NO CONTINUE TO NEXT TEST
:FLUSH CACHE SELECT OTHER VALID SET
<table>
<thead>
<tr>
<th>Address</th>
<th>Byte</th>
<th>Operation 1</th>
<th>Operation 2</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>10535</td>
<td>03</td>
<td>4220</td>
<td>012737</td>
<td>001015</td>
</tr>
<tr>
<td>10536</td>
<td>03</td>
<td>4226</td>
<td>005037</td>
<td>004550</td>
</tr>
<tr>
<td>10537</td>
<td>03</td>
<td>4232</td>
<td>005037</td>
<td>004552</td>
</tr>
<tr>
<td>10538</td>
<td>03</td>
<td>4236</td>
<td>010137</td>
<td>004554</td>
</tr>
<tr>
<td>10539</td>
<td>03</td>
<td>4242</td>
<td>012737</td>
<td>000001</td>
</tr>
<tr>
<td>10540</td>
<td>03</td>
<td>4250</td>
<td>005237</td>
<td>004556</td>
</tr>
<tr>
<td>10541</td>
<td>03</td>
<td>4254</td>
<td>004037</td>
<td>005352</td>
</tr>
<tr>
<td>10542</td>
<td>03</td>
<td>4260</td>
<td>000006</td>
<td></td>
</tr>
<tr>
<td>10543</td>
<td>03</td>
<td>4262</td>
<td>041720</td>
<td></td>
</tr>
<tr>
<td>10544</td>
<td>03</td>
<td>4266</td>
<td>041732</td>
<td></td>
</tr>
<tr>
<td>10545</td>
<td>03</td>
<td>4266</td>
<td>041744</td>
<td></td>
</tr>
<tr>
<td>10546</td>
<td>03</td>
<td>4270</td>
<td>012737</td>
<td>000011</td>
</tr>
<tr>
<td>10547</td>
<td>03</td>
<td>4276</td>
<td>053737</td>
<td>004542</td>
</tr>
<tr>
<td>10548</td>
<td>03</td>
<td>4304</td>
<td>000742</td>
<td></td>
</tr>
<tr>
<td>10549</td>
<td>03</td>
<td>4306</td>
<td>012737</td>
<td>001015</td>
</tr>
<tr>
<td>10550</td>
<td>03</td>
<td>4314</td>
<td>013737</td>
<td>005064</td>
</tr>
<tr>
<td>10551</td>
<td>03</td>
<td>4322</td>
<td>004010</td>
<td></td>
</tr>
<tr>
<td>10552</td>
<td>03</td>
<td>4324</td>
<td>046000</td>
<td></td>
</tr>
<tr>
<td>10553</td>
<td>03</td>
<td>4326</td>
<td>046000</td>
<td></td>
</tr>
<tr>
<td>10554</td>
<td>03</td>
<td>4328</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10555</td>
<td>03</td>
<td>432A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10556</td>
<td>03</td>
<td>4338</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10557</td>
<td>03</td>
<td>433A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10558</td>
<td>03</td>
<td>433C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10559</td>
<td>03</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SDBTL TAG MARCH PATTERN TESTS
 OPERATING IN LOW CACHE TESTING HIGH CACHE

TST130: INC TID ; UPDATE TEST ID

JSR R4,RELOC

;RELOCATE TEST TO LOW CACHE

WORD TST131

CLR @IOPERR ; RESET DELAY ERROR

CLR @ERROR ; RESET ERROR FLAG

MOV #35, @MCR ; ENABLE HIGH CACHE

MOV #2000,R1 ; TAGGING START ADDRESS

15: MOV (R1), @R2 ; WRITE ZERO TO TAG

CMP R1, #4000 ; FOR ALL HIGH LOCATIONS

BNE 16

MOV #2000,R1 ; READ TAG, START ADDRESS

25: MOV (R1), @R2 ; READ BACKGROUND

BIT #BIT03, @CHR ; SHOULD HAVE HIT

BEQ 35 ; NO HIT

MOV #1, @MCR ; ENABLE MAINT. MODE

MOV (R1), @R2 ; WRITE COMP TAG

MOV (R1), @R2 ; READ COMP TAG

BIT #BIT03, @CHR ; SHOULD HAVE HIT

BEQ 35

MOV #0, @MCR ; DISABLE MAINT MODE

MOV R2, R1 ; NEXT ADDRESS

CMP R1, #4000 ; LOOP COMPLETE YET

BNE 28

BR 63

35: MOV #0, @MCR ; DISABLE MAINT MODE

MOV #0, @MCR ; DISABLE CACHE

CLR @BAD ; DATA UNKNOWN

INC @GOODBD

INC @ERROR ; SET ERROR FLAG

MOV R1, @ADD ; CACHE ADDRESS

JSR R0, @SETEN ; REPORT ERROR

WORD #06

WORD SENT154 ; TAG MARCH PATTERN TESTS

WORD SENT157 ; HIGH CACHE TAG FAILURE

WORD SENT156 ; CACHE ADDRESS, DATA UNKNOWN

51: MOV #35, @MCR ; ENABLE CACHE

65: MOV #4000,R1 ; END ADDRESS

75: MOV #1, @MCR ; ENABLE MAINT. MODE

MOV -(R1), @R2 ; READ BACKGROUND

BIT #BIT03, @CHR ; ANY HIT?

85: BNE 95

MOV #0, @MCR ; DISABLE MAINT. MODE

MOV (R1), @R2 ; WRITE COMP

MOV (R1), @R2 ; READ COMP

BIT #BIT03, @CHR ; SHOULD HAVE HIT

BEQ 85

BNE 95 ; LOOP COMPLETE YET

95: CMP #2000,R1

BNE 75

100
10617 034626 012737 001015 177746 8$: MOV #OFF, #WCCR ;DISABLE CACHE
10618 034634 005037 004350 CLR #GOOD ;DATA UNKNOWN
10619 034640 005037 004552 CLR #BAD
10620 034644 010137 004554 MOV #1, #ADD ;CACHE ADDRESS
10621 034650 012737 000001 004542 MOV #1, #ERROR ;SET ERROR FLAG
10622 034656 005237 004556 INC #GOODDB ;SET BIT PRINT MODE
10623 034662 004037 005352 JSR R0, #SETEN ;REPORT ERROR
10624 034666 000006 .WORD "06" ;TAG MARCH PATTERN TESTS
10625 034670 041720 .WORD SEN1S4 ;HIGH CACHE TAG FAILURE
10626 034672 041760 .WORD SEN1S7 ;CACHE ADDRESS, DATA UNKNOWN
10627 034674 041744 .WORD SEN1S6 ;ENABLE CACHE
10628 034676 012737 000005 177746 MOV #5, #WCCR ;DISABLE CACHE
10629 034704 053737 004542 005064 BIS #ERROR, #WLOPERR ;SET DELAYED ERROR FLAG
10630 034712 000741 BR 9$: ;TAKE SELECTED ACTION ON ERROR
10631 034714 012737 001015 177746 10$: MOV #OFF, #WCCR
10632 034722 013737 005064 004542 MOV #WLOPERR, #ERROR
10633 034750 004010 JSR R0, (RD)
10634 034752 044000 .WORD LOWSP
10635 034734 044000 .WORD LOWSP
10636
10637
10638
10639
10640
10641
.SBTL PARITY INTERRUPT TESTS

WRITE WRONG PARITY TO CACHE
READ CACHE LOCATION, CAUSE PARITY ERROR
ENABLE PARITY INTERRUPT CAUSE TRAP

10643 034736 005267 143704 TST131:   INC TID
10654 034774 004467 151146   UPD TID
10655 034746 035126   RELCTL
10656 034750 012737 000000 177776   DLCT
10657 034758 005037 177744   LGT
10658 034762 012737 000005 177746   LGT
10659 034770 013702 052000   LGT
10660 034774 013702 046000   LGT
10661 035000 005273 000100 177746   LGT
10662 035006 012737 000077 046000   LGT
10663 035014 012737 001015 177746   LGT
10664 035022 010702   LGT
10665 035024 014212   LGT
10666 035026 020227 044000   LGT
BNE 16
10667 035032 001374   LGT
10668 035034 012737 000005 177746   LGT
10669 035042 013702 046000   LGT
10670 035046 042737 000001 177746   LGT
10671 035054 013737 177746 004542   LGT
10672 035062 012737 001015 177746   LGT
10673 035070 005137 004542   LGT
10674 035074 042737 177776 004542   LGT
10675 035102 004037 005352   LGT
10676 035106 000010   LGT
10677 035110 041772   LGT
10678 035112 042002   LGT
10679 035116 042034   LGT
10680 035116 042034   LGT
10681 035120 004010   LGT
10682 035122 044000   LGT
10683 035124 044000   LGT

WRITE WRONG PARITY TO CACHE
ENABLE PARITY ERROR ABORT FOR HIGH CACHE
READ AND CAUSE PARITY ERROR

10693 035126 005267 143514 TST132:   INC TID
10694 035132 004467 150756   UPD TID
10695 035136 035276   RELCTL
10696 035140 005037 177744   LGT
10697 035144 012737 000005 177746   LGT
10698 035152 013702 052000   LGT

:PARITY INTERRUPT TESTS
CACHE FAILED TO INTERRUPT ON PARITY ERROR
USING CACHE INTERRUPT LOGIC
BIT0 OF CCR USED
TAKE SELECTED ACTION ON ERROR

:READ AND CAUSE PARITY ERROR
<table>
<thead>
<tr>
<th>Location</th>
<th>Instruction(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>000000</td>
<td>\text{RESET}</td>
</tr>
<tr>
<td>000005</td>
<td>\text{ENDIT: JSR PC,(RO)}</td>
</tr>
<tr>
<td>004710</td>
<td>\text{NOP}</td>
</tr>
<tr>
<td>000240</td>
<td>\text{NOP}</td>
</tr>
<tr>
<td>000240</td>
<td>\text{NOP}</td>
</tr>
<tr>
<td>000240</td>
<td>\text{NOP}</td>
</tr>
<tr>
<td>000167</td>
<td>\text{JMP PREPARE}</td>
</tr>
<tr>
<td>143102</td>
<td>\text{RESTART DIAG.}</td>
</tr>
<tr>
<td>040520</td>
<td>\text{ASCII /PAS COMPLETE/}</td>
</tr>
<tr>
<td>051523</td>
<td>\text{BYTE 15,12,177,177,177,0}</td>
</tr>
<tr>
<td>041440</td>
<td>\text{EVEN}</td>
</tr>
<tr>
<td>0112</td>
<td></td>
</tr>
</tbody>
</table>
0989 042166 042455 043254 043262 SEN168: .WORD DIC27, DIC95, DIC96, DIC160, DIC161, DIC162, DIC160, DIC163, 0
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>11084</td>
<td>043236</td>
<td>ASCIZ</td>
<td><em>FAILURE</em></td>
</tr>
<tr>
<td>11085</td>
<td>043246</td>
<td>ASCIZ</td>
<td><em>LID</em></td>
</tr>
<tr>
<td>11086</td>
<td>043252</td>
<td>ASCIZ</td>
<td><em>LID</em></td>
</tr>
<tr>
<td>11087</td>
<td>043254</td>
<td>ASCIZ</td>
<td><em>FORCE</em></td>
</tr>
<tr>
<td>11088</td>
<td>043262</td>
<td>ASCIZ</td>
<td><em>MISS</em></td>
</tr>
<tr>
<td>11089</td>
<td>043267</td>
<td>ASCIZ</td>
<td><em>LOOKS</em></td>
</tr>
<tr>
<td>11090</td>
<td>043275</td>
<td>ASCIZ</td>
<td><em>DISABLED</em></td>
</tr>
<tr>
<td>11091</td>
<td>043282</td>
<td>ASCIZ</td>
<td><em>ENABLED</em></td>
</tr>
<tr>
<td>11092</td>
<td>043313</td>
<td>ASCIZ</td>
<td><em>ERROR</em></td>
</tr>
<tr>
<td>11093</td>
<td>043323</td>
<td>ASCIZ</td>
<td><em>ATTEMTP</em></td>
</tr>
<tr>
<td>11094</td>
<td>043331</td>
<td>ASCIZ</td>
<td><em>VIA</em></td>
</tr>
<tr>
<td>11095</td>
<td>043345</td>
<td>ASCIZ</td>
<td><em>AND</em></td>
</tr>
<tr>
<td>11096</td>
<td>043351</td>
<td>ASCIZ</td>
<td><em>TAG</em></td>
</tr>
<tr>
<td>11097</td>
<td>043371</td>
<td>ASCIZ</td>
<td><em>SHEK</em></td>
</tr>
<tr>
<td>11098</td>
<td>043377</td>
<td>ASCIZ</td>
<td><em>ATTEMTP</em></td>
</tr>
<tr>
<td>11099</td>
<td>043385</td>
<td>ASCIZ</td>
<td><em>CANT</em></td>
</tr>
<tr>
<td>11100</td>
<td>043391</td>
<td>ASCIZ</td>
<td><em>WAS</em></td>
</tr>
<tr>
<td>11101</td>
<td>043397</td>
<td>ASCIZ</td>
<td><em>LOGIC</em></td>
</tr>
<tr>
<td>11102</td>
<td>043403</td>
<td>ASCIZ</td>
<td><em>OUI</em></td>
</tr>
<tr>
<td>11103</td>
<td>043410</td>
<td>ASCIZ</td>
<td><em>EVEN</em></td>
</tr>
<tr>
<td>11104</td>
<td>043417</td>
<td>ASCIZ</td>
<td><em>READS</em></td>
</tr>
<tr>
<td>11105</td>
<td>043423</td>
<td>ASCIZ</td>
<td><em>SIX</em></td>
</tr>
<tr>
<td>11106</td>
<td>043429</td>
<td>ASCIZ</td>
<td><em>TAGGED</em></td>
</tr>
<tr>
<td>11107</td>
<td>043435</td>
<td>ASCIZ</td>
<td><em>LOCATIONS</em></td>
</tr>
<tr>
<td>11108</td>
<td>043440</td>
<td>ASCIZ</td>
<td><em>I/O</em></td>
</tr>
<tr>
<td>11109</td>
<td>043450</td>
<td>ASCIZ</td>
<td><em>PAGE</em></td>
</tr>
<tr>
<td>11110</td>
<td>043455</td>
<td>ASCIZ</td>
<td><em>INVALIATED</em></td>
</tr>
<tr>
<td>11111</td>
<td>043461</td>
<td>ASCIZ</td>
<td><em>LOCATION</em></td>
</tr>
<tr>
<td>11112</td>
<td>043467</td>
<td>ASCIZ</td>
<td><em>WHILE</em></td>
</tr>
<tr>
<td>11113</td>
<td>043473</td>
<td>ASCIZ</td>
<td><em>MODE</em></td>
</tr>
<tr>
<td>11114</td>
<td>043479</td>
<td>ASCIZ</td>
<td><em>STORAGE</em></td>
</tr>
<tr>
<td>11115</td>
<td>043485</td>
<td>ASCIZ</td>
<td><em>OPERATING</em></td>
</tr>
<tr>
<td>11116</td>
<td>043491</td>
<td>ASCIZ</td>
<td><em>WITH</em></td>
</tr>
<tr>
<td>11117</td>
<td>043497</td>
<td>ASCIZ</td>
<td><em>BYPASS</em></td>
</tr>
<tr>
<td>11118</td>
<td>043503</td>
<td>ASCIZ</td>
<td><em>BY</em></td>
</tr>
<tr>
<td>11119</td>
<td>043509</td>
<td>ASCIZ</td>
<td><em>RUN</em></td>
</tr>
<tr>
<td>11120</td>
<td>043515</td>
<td>ASCIZ</td>
<td><em>MAGIC</em></td>
</tr>
<tr>
<td>11121</td>
<td>043521</td>
<td>ASCIZ</td>
<td><em>MEMORY</em></td>
</tr>
<tr>
<td>11122</td>
<td>043527</td>
<td>ASCIZ</td>
<td><em>MEMORY</em></td>
</tr>
<tr>
<td>11123</td>
<td>043533</td>
<td>ASCIZ</td>
<td><em>MEMORY</em></td>
</tr>
<tr>
<td>11124</td>
<td>054000</td>
<td>ASCIZ</td>
<td>.-54000</td>
</tr>
<tr>
<td>11125</td>
<td>054006</td>
<td>ASCIZ</td>
<td>.REV B</td>
</tr>
<tr>
<td>11126</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11127</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11128</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11129</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11130</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11131</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11132</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11133</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11134</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11135</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11136</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11137</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11138</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>11139</td>
<td>054000</td>
<td>ASCIZ</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>Value 1</td>
<td>Value 2</td>
<td>Value 3</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>11140</td>
<td>054125</td>
<td>115</td>
<td>051101</td>
</tr>
<tr>
<td>11141</td>
<td>054133</td>
<td>120</td>
<td>052101</td>
</tr>
<tr>
<td>11142</td>
<td>054143</td>
<td>105</td>
<td>050130</td>
</tr>
<tr>
<td>11143</td>
<td>054154</td>
<td>051127</td>
<td>052111</td>
</tr>
<tr>
<td>11144</td>
<td>054164</td>
<td>051127</td>
<td>052111</td>
</tr>
<tr>
<td>11145</td>
<td>054173</td>
<td>125</td>
<td>043516</td>
</tr>
<tr>
<td>11146</td>
<td>054203</td>
<td>111</td>
<td>052116</td>
</tr>
<tr>
<td>11147</td>
<td>054215</td>
<td>125</td>
<td>042523</td>
</tr>
<tr>
<td>11148</td>
<td>054222</td>
<td>041101</td>
<td>051117</td>
</tr>
<tr>
<td>11149</td>
<td>054230</td>
<td>051124</td>
<td>050101</td>
</tr>
<tr>
<td>11150</td>
<td>054235</td>
<td>102</td>
<td>052117</td>
</tr>
<tr>
<td>11151</td>
<td>054242</td>
<td>041517</td>
<td>052503</td>
</tr>
<tr>
<td>11152</td>
<td>054242</td>
<td>051124</td>
<td>050101</td>
</tr>
<tr>
<td>11153</td>
<td>054242</td>
<td>041123</td>
<td>052517</td>
</tr>
<tr>
<td>11154</td>
<td>054271</td>
<td>110</td>
<td>053101</td>
</tr>
<tr>
<td>11155</td>
<td>054276</td>
<td>041101</td>
<td>051117</td>
</tr>
<tr>
<td>11156</td>
<td>054306</td>
<td>053523</td>
<td>052111</td>
</tr>
<tr>
<td>11157</td>
<td>054315</td>
<td>105</td>
<td>051122</td>
</tr>
<tr>
<td>11158</td>
<td>054325</td>
<td>126</td>
<td>051105</td>
</tr>
<tr>
<td>11159</td>
<td>054334</td>
<td>047520</td>
<td>044523</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>11161</td>
<td>044000</td>
<td>LOWSP:</td>
<td>.:=44000</td>
</tr>
<tr>
<td>11162</td>
<td>045000</td>
<td>WORD 0</td>
<td></td>
</tr>
<tr>
<td>11163</td>
<td>046000</td>
<td>.=46000</td>
<td></td>
</tr>
<tr>
<td>11164</td>
<td>046000</td>
<td>HIGHSP:</td>
<td>.:=50000</td>
</tr>
<tr>
<td>11165</td>
<td>050000</td>
<td>WORD 0</td>
<td></td>
</tr>
<tr>
<td>11166</td>
<td>050000</td>
<td>LOW1:</td>
<td>.:=52000</td>
</tr>
<tr>
<td>11167</td>
<td>052000</td>
<td>END</td>
<td></td>
</tr>
<tr>
<td>11168</td>
<td>052000</td>
<td>HIGH1:</td>
<td>.:=END</td>
</tr>
<tr>
<td>11169</td>
<td>000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbol</td>
<td>Value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ABAFE</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACDW1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACDW2</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACDW3</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACTION</td>
<td>6089&lt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADD</td>
<td>064554</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDST</td>
<td>066400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW0</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW2</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW3</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW4</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW5</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW6</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDW7</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADEVM</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDRD</td>
<td>005642</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AEMV</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFATAL</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AHALT</td>
<td>001120</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMADR1</td>
<td>6106&lt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMADR2</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMADR3</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMADR4</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG2</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG3</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG4</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG5</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG6</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG7</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG8</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG9</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSG10</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMY7</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMY8</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMY9</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>APASS</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>APRIOR</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASWREG</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATST1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUNIT</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATSW1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVECT1</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVECT2</td>
<td>6018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BAD</td>
<td>004552</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BADST</td>
<td>006434</td>
<td>6611</td>
<td>664-8#</td>
</tr>
<tr>
<td>BEBA</td>
<td>170004</td>
<td>60764</td>
<td>976-2#</td>
</tr>
<tr>
<td>BECC</td>
<td>170002</td>
<td>60754</td>
<td>9765#</td>
</tr>
<tr>
<td>BECR1</td>
<td>170006</td>
<td>60784</td>
<td>9745#</td>
</tr>
<tr>
<td>BECR2</td>
<td>170016</td>
<td>60794</td>
<td>9765#</td>
</tr>
<tr>
<td>BEDA</td>
<td>170000</td>
<td>60774</td>
<td>9764#</td>
</tr>
<tr>
<td>BIT</td>
<td>006536</td>
<td>6656</td>
<td>6678#</td>
</tr>
<tr>
<td>BITFLG</td>
<td>006560</td>
<td>65314</td>
<td>6500#</td>
</tr>
<tr>
<td>BITNAM</td>
<td>006454</td>
<td>6515</td>
<td>6544#</td>
</tr>
<tr>
<td>BIT00</td>
<td>000001</td>
<td>60404</td>
<td>6880</td>
</tr>
<tr>
<td>BIT01</td>
<td>000002</td>
<td>60414</td>
<td>7160</td>
</tr>
<tr>
<td>BIT02</td>
<td>000004</td>
<td>60424</td>
<td>6503</td>
</tr>
<tr>
<td>BIT03</td>
<td>000010</td>
<td>60434</td>
<td>6954</td>
</tr>
<tr>
<td>BIT04</td>
<td>000020</td>
<td>60444</td>
<td>7180</td>
</tr>
<tr>
<td>BIT05</td>
<td>000040</td>
<td>60454</td>
<td>7200</td>
</tr>
<tr>
<td>BIT06</td>
<td>000100</td>
<td>9838</td>
<td>10017</td>
</tr>
<tr>
<td>BIT07</td>
<td>000200</td>
<td>60474</td>
<td>7047</td>
</tr>
<tr>
<td>BIT08</td>
<td>000400</td>
<td>60484</td>
<td>6098</td>
</tr>
<tr>
<td>BIT09</td>
<td>001000</td>
<td>8149</td>
<td>8152</td>
</tr>
<tr>
<td>BIT10</td>
<td>002000</td>
<td>60504</td>
<td>7124</td>
</tr>
<tr>
<td>BIT11</td>
<td>004000</td>
<td>60514</td>
<td>7250</td>
</tr>
<tr>
<td>BIT12</td>
<td>010000</td>
<td>60524</td>
<td>7803</td>
</tr>
<tr>
<td>BIT13</td>
<td>020000</td>
<td>8072</td>
<td>8080</td>
</tr>
<tr>
<td>BIT14</td>
<td>040000</td>
<td>60544</td>
<td>6099</td>
</tr>
<tr>
<td>BIT15</td>
<td>100000</td>
<td>5996</td>
<td>6054</td>
</tr>
<tr>
<td>BLFINT</td>
<td>004424</td>
<td>60054</td>
<td>6012</td>
</tr>
<tr>
<td>CCR</td>
<td>177746</td>
<td>59834</td>
<td>60244</td>
</tr>
</tbody>
</table>

**CROSS REFERENCE TABLE -- USER SYMBOLS**

**SEQ 0134**
<table>
<thead>
<tr>
<th>CACHE DIAG.</th>
<th>MACY11 30A(C052)</th>
<th>31-DEC-79 15:29</th>
<th>PAGE 77-3</th>
<th>G71</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFFKAB.P11</td>
<td>25-JUN-79 13:31</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CROSS REFERENCE TABLE -- USER SYMBOLS**

<table>
<thead>
<tr>
<th>Code</th>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1C103</td>
<td>0.4335</td>
<td>10900, 10903</td>
</tr>
<tr>
<td>D1C104</td>
<td>0.4351</td>
<td>10905, 11096#</td>
</tr>
<tr>
<td>D1C105</td>
<td>0.4357</td>
<td>10907, 11097#</td>
</tr>
<tr>
<td>D1C106</td>
<td>0.4355</td>
<td>10908, 10965</td>
</tr>
<tr>
<td>D1C107</td>
<td>0.4361</td>
<td>10908, 11099#</td>
</tr>
<tr>
<td>D1C108</td>
<td>0.4365</td>
<td>10911, 10912, 11100#</td>
</tr>
<tr>
<td>D1C109</td>
<td>0.4372</td>
<td>10914, 10918, 11101#</td>
</tr>
<tr>
<td>D1C110</td>
<td>0.4373</td>
<td>10824, 10832, 10846, 10854, 11002#</td>
</tr>
<tr>
<td>D1C111</td>
<td>0.4376</td>
<td>10921, 10981, 10983, 10985, 11102#</td>
</tr>
<tr>
<td>D1C112</td>
<td>0.4304</td>
<td>10923, 10927, 11103#</td>
</tr>
<tr>
<td>D1C113</td>
<td>0.4310</td>
<td>10929, 11104#</td>
</tr>
<tr>
<td>D1C114</td>
<td>0.4315</td>
<td>10929, 11105#</td>
</tr>
<tr>
<td>D1C115</td>
<td>0.4323</td>
<td>10929, 11106#</td>
</tr>
<tr>
<td>D1C116</td>
<td>0.4327</td>
<td>10929, 11107#</td>
</tr>
<tr>
<td>D1C117</td>
<td>0.4336</td>
<td>10929, 11108#</td>
</tr>
<tr>
<td>D1C118</td>
<td>0.4336</td>
<td>10930, 10931, 10932, 11109#</td>
</tr>
<tr>
<td>D1C119</td>
<td>0.4345</td>
<td>10930, 10931, 10932, 11110#</td>
</tr>
<tr>
<td>D1C12</td>
<td>0.43461</td>
<td>10932, 10939, 11111#</td>
</tr>
<tr>
<td>D1C120</td>
<td>0.4357</td>
<td>10825, 10861, 10933, 11003#</td>
</tr>
<tr>
<td>D1C121</td>
<td>0.4357</td>
<td>10932, 10939, 10940, 10943, 10944, 10946, 10960, 11112#</td>
</tr>
<tr>
<td>D1C122</td>
<td>0.4357</td>
<td>10933, 11113#</td>
</tr>
<tr>
<td>D1C123</td>
<td>0.4357</td>
<td>10933, 10938, 10940, 10943, 11114#</td>
</tr>
<tr>
<td>D1C124</td>
<td>0.4357</td>
<td>10934, 11115#</td>
</tr>
<tr>
<td>D1C125</td>
<td>0.4357</td>
<td>10935, 11116#</td>
</tr>
<tr>
<td>D1C126</td>
<td>0.4357</td>
<td>10935, 10941, 11117#</td>
</tr>
<tr>
<td>D1C127</td>
<td>0.4357</td>
<td>10938, 10940, 10943, 11118#</td>
</tr>
<tr>
<td>D1C128</td>
<td>0.4357</td>
<td>10940, 10962, 11119#</td>
</tr>
<tr>
<td>D1C129</td>
<td>0.4357</td>
<td>10941, 11120#</td>
</tr>
<tr>
<td>D1C130</td>
<td>0.4357</td>
<td>10945, 11121#</td>
</tr>
<tr>
<td>D1C131</td>
<td>0.4357</td>
<td>10826, 10904, 10905, 10906, 10907, 10908, 10928, 10931, 10946, 10948, 10951, 11004#</td>
</tr>
<tr>
<td>D1C132</td>
<td>0.4357</td>
<td>10946, 10949, 10951, 10958, 10960, 11122#</td>
</tr>
<tr>
<td>D1C133</td>
<td>0.4357</td>
<td>10947, 11123#</td>
</tr>
<tr>
<td>D1C134</td>
<td>0.54000</td>
<td>10950, 11128#</td>
</tr>
<tr>
<td>D1C135</td>
<td>0.54007</td>
<td>10952, 11129#</td>
</tr>
<tr>
<td>D1C136</td>
<td>0.54016</td>
<td>10953, 11130#</td>
</tr>
<tr>
<td>D1C137</td>
<td>0.54016</td>
<td>10954, 11131#</td>
</tr>
<tr>
<td>D1C138</td>
<td>0.54025</td>
<td>10955, 11132#</td>
</tr>
<tr>
<td>D1C139</td>
<td>0.54034</td>
<td>10956, 11133#</td>
</tr>
<tr>
<td>D1C140</td>
<td>0.54043</td>
<td>10957, 11134#</td>
</tr>
<tr>
<td>D1C141</td>
<td>0.54052</td>
<td>10959, 11135#</td>
</tr>
<tr>
<td>D1C142</td>
<td>0.54061</td>
<td>10827, 10828, 10829, 11005#</td>
</tr>
<tr>
<td>D1C143</td>
<td>0.54071</td>
<td>1061, 10963, 11136#</td>
</tr>
<tr>
<td>D1C144</td>
<td>0.54080</td>
<td>1061, 10963, 11137#</td>
</tr>
<tr>
<td>D1C145</td>
<td>0.54090</td>
<td>1061, 10963, 11138#</td>
</tr>
<tr>
<td>D1C146</td>
<td>0.54110</td>
<td>1063, 11139#</td>
</tr>
<tr>
<td>D1C147</td>
<td>0.54120</td>
<td>1064, 10966, 10970, 10975, 11140#</td>
</tr>
<tr>
<td>D1C148</td>
<td>0.54130</td>
<td>1064, 10966, 10970, 10975, 11141#</td>
</tr>
<tr>
<td>D1C149</td>
<td>0.54140</td>
<td>1065, 11142#</td>
</tr>
<tr>
<td>D1C150</td>
<td>0.54150</td>
<td>1068, 10972, 11143#</td>
</tr>
<tr>
<td>D1C151</td>
<td>0.54160</td>
<td>1068, 10972, 11143#</td>
</tr>
<tr>
<td>D1C152</td>
<td>0.54160</td>
<td>10983, 10985, 10986, 11148#</td>
</tr>
<tr>
<td>GOODST</td>
<td>6606</td>
<td>6647#</td>
</tr>
<tr>
<td>HELP</td>
<td>6202#</td>
<td>6673</td>
</tr>
<tr>
<td>HIGHSP</td>
<td>046000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6586</td>
<td>6592</td>
</tr>
<tr>
<td></td>
<td>6788</td>
<td>6798</td>
</tr>
<tr>
<td></td>
<td>7588</td>
<td>7598</td>
</tr>
<tr>
<td></td>
<td>8727</td>
<td>8729</td>
</tr>
<tr>
<td></td>
<td>9191</td>
<td>9193</td>
</tr>
<tr>
<td></td>
<td>9420</td>
<td>9422</td>
</tr>
<tr>
<td></td>
<td>9663</td>
<td>9665</td>
</tr>
<tr>
<td></td>
<td>9996</td>
<td>9997</td>
</tr>
<tr>
<td>HIGH1</td>
<td>052000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7603</td>
<td>7621</td>
</tr>
<tr>
<td></td>
<td>8685</td>
<td>8725</td>
</tr>
<tr>
<td></td>
<td>9041</td>
<td>9043</td>
</tr>
<tr>
<td></td>
<td>9922</td>
<td>9950</td>
</tr>
<tr>
<td></td>
<td>10366</td>
<td>10659</td>
</tr>
<tr>
<td></td>
<td>6107</td>
<td>6113#</td>
</tr>
<tr>
<td>M1</td>
<td>002004</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6170#</td>
<td>6467</td>
</tr>
<tr>
<td>M2</td>
<td>002014</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6172#</td>
<td>6468</td>
</tr>
<tr>
<td>MLL</td>
<td>005174</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6432</td>
<td>6437#</td>
</tr>
<tr>
<td>INBUL</td>
<td>004430</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6005</td>
<td>6012</td>
</tr>
<tr>
<td></td>
<td>6808</td>
<td>6831</td>
</tr>
<tr>
<td></td>
<td>6056#</td>
<td>8221#</td>
</tr>
<tr>
<td></td>
<td>6070#</td>
<td>9572#</td>
</tr>
<tr>
<td>TRAP</td>
<td>004540</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6345#</td>
<td>6357*</td>
</tr>
<tr>
<td></td>
<td>6808</td>
<td>6831</td>
</tr>
<tr>
<td></td>
<td>6005#</td>
<td>8221#</td>
</tr>
<tr>
<td></td>
<td>6070#</td>
<td>9523#</td>
</tr>
<tr>
<td>ROODCO=177566</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KPAR0</td>
<td>172340</td>
<td></td>
</tr>
<tr>
<td>KPAR1</td>
<td>172342</td>
<td></td>
</tr>
<tr>
<td>KPAR2</td>
<td>172344</td>
<td></td>
</tr>
<tr>
<td>KPAR3</td>
<td>172346</td>
<td></td>
</tr>
<tr>
<td>KPAR4</td>
<td>172350</td>
<td></td>
</tr>
<tr>
<td>KPAR5</td>
<td>172352</td>
<td></td>
</tr>
<tr>
<td>KPAR6</td>
<td>172354</td>
<td></td>
</tr>
<tr>
<td>KPAR7</td>
<td>172356</td>
<td></td>
</tr>
<tr>
<td>KPAR8</td>
<td>172360</td>
<td></td>
</tr>
<tr>
<td>KPAR9</td>
<td>172364</td>
<td></td>
</tr>
<tr>
<td>KRB</td>
<td>177562</td>
<td></td>
</tr>
<tr>
<td>KRS</td>
<td>177560</td>
<td></td>
</tr>
<tr>
<td>LAST</td>
<td>004710</td>
<td></td>
</tr>
<tr>
<td>LFF1</td>
<td>005552</td>
<td></td>
</tr>
<tr>
<td>LINE</td>
<td>005625</td>
<td></td>
</tr>
<tr>
<td>LINFRQ</td>
<td>001176</td>
<td></td>
</tr>
<tr>
<td>LOPPR</td>
<td>005064</td>
<td></td>
</tr>
<tr>
<td>LOST</td>
<td>002210</td>
<td></td>
</tr>
<tr>
<td>LWSB</td>
<td>046000</td>
<td></td>
</tr>
</tbody>
</table>

**USER SYMBOLS**

SEQ 0140
<table>
<thead>
<tr>
<th>L11</th>
<th>050000</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPOINTS</td>
<td>005012</td>
</tr>
<tr>
<td>LP1</td>
<td>001742</td>
</tr>
<tr>
<td>LP2</td>
<td>001754</td>
</tr>
<tr>
<td>LP3</td>
<td>001764</td>
</tr>
<tr>
<td>LP4</td>
<td>001774</td>
</tr>
<tr>
<td>LP5</td>
<td>002044</td>
</tr>
<tr>
<td>LP6</td>
<td>002742</td>
</tr>
<tr>
<td>MAPRE</td>
<td>025564</td>
</tr>
<tr>
<td>MAYBE</td>
<td>007726</td>
</tr>
<tr>
<td>NAME</td>
<td>001446</td>
</tr>
<tr>
<td>NOINC</td>
<td>006712</td>
</tr>
<tr>
<td>NOJBE</td>
<td>004562</td>
</tr>
<tr>
<td>OCTASC</td>
<td>006510</td>
</tr>
<tr>
<td>OFF</td>
<td>001015</td>
</tr>
<tr>
<td>PARITY</td>
<td>004650</td>
</tr>
<tr>
<td>PNTNAM</td>
<td>001364</td>
</tr>
<tr>
<td>PPB</td>
<td>177566</td>
</tr>
<tr>
<td>PPS</td>
<td>-177564</td>
</tr>
<tr>
<td>PREPAR</td>
<td>001000</td>
</tr>
<tr>
<td>PRINT</td>
<td>000564</td>
</tr>
<tr>
<td>PROM</td>
<td>001750</td>
</tr>
<tr>
<td>PSW</td>
<td>=177776</td>
</tr>
<tr>
<td>PT1D</td>
<td>004010</td>
</tr>
<tr>
<td>QUESHZ</td>
<td>001210</td>
</tr>
<tr>
<td>RELCTH</td>
<td>006412</td>
</tr>
<tr>
<td>RELCTL</td>
<td>006114</td>
</tr>
<tr>
<td>REST</td>
<td>035672</td>
</tr>
<tr>
<td>SAVTAT</td>
<td>006456</td>
</tr>
<tr>
<td>SENT</td>
<td>035722</td>
</tr>
</tbody>
</table>

**CACHE DIAG.**


**CROSS REFERENCE TABLE — USER SYMBOLS**

SEQ 0141
<p>| SEN149  | 041600 | 10374 | 10380 | 10457 | 10464 | 10970# |
| SEN15   | 036244 | 6961  | 10836#|
| SEN150  | 041614 | 10575 | 10568 | 10971#|
| SEN151  | 041630 | 10376 | 10459 | 10972#|
| SEN152  | 041656 | 10381 | 10465 | 10973#|
| SEN153  | 041672 | 10582 | 10466 | 10974#|
| SEN154  | 041740 | 10516 | 10543 | 10598 | 10625 | 10975#|
| SEN155  | 041732 | 10517 | 10544 | 10976#|
| SEN156  | 041744 | 10518 | 10545 | 10600 | 10627 | 10977#|
| SEN157  | 041760 | 10599 | 10626 | 10978#|
| SEN158  | 041772 | 10677 | 10714 | 10751 | 10797 | 10979#|
| SEN159  | 042002 | 10678 | 10715 | 10752 | 10980#|
| SEN16   | 036264 | 6996  | 10837#|
| SEN160  | 042022 | 10679 | 10981#|
| SEN161  | 042034 | 10680 | 10982#|
| SEN162  | 042046 | 10716 | 10983#|
| SEN163  | 042066 | 10717 | 10754 | 10984#|
| SEN164  | 042076 | 10753 | 10985#|
| SEN165  | 042116 | 10798 | 10986#|
| SEN166  | 042136 | 10799 | 10987#|
| SEN167  | 042154 | 10800 | 10988#|
| SEN168  | 042166 | 6911  | 6962  | 10989#|
| SEN17   | 036304 | 7007  | 10838#|
| SEN18   | 036324 | 7033  | 10839#|
| SEN19   | 036344 | 7054  | 10840#|
| SEN2    | 035734 | 6711  | 10823#|
| SEN20   | 036364 | 7076  | 10841#|
| SEN21   | 036404 | 7093  | 10842#|
| SEN22   | 036424 | 7115  | 10843#|
| SEN23   | 036444 | 7130  | 10844#|
| SEN24   | 036464 | 7147  | 10845#|
| SEN25   | 036504 | 7165  | 7185  | 7205  | 7225  | 7245  | 7265  | 7285  | 10846#|
| SEN26   | 036520 | 7166  | 10847#|
| SEN27   | 036536 | 7186  | 10848#|
| SEN28   | 036554 | 7206  | 10849#|
| SEN29   | 036572 | 7226  | 10850#|
| SEN3    | 035754 | 6731  | 10824#|
| SEN30   | 036610 | 7246  | 10851#|
| SEN31   | 036626 | 7266  | 10852#|
| SEN32   | 036644 | 7286  | 10853#|
| SEN33   | 036662 | 7310  | 7334  | 10854#|
| SEN34   | 036682 | 7311  | 10855#|
| SEN35   | 036714 | 7312  | 10856#|
| SEN36   | 036732 | 7313  | 10857#|
| SEN37   | 036744 | 7335  | 10858#|
| SEN38   | 036762 | 7336  | 10859#|
| SEN39   | 037000 | 7357  | 10860#|
| SEN4    | 035772 | 6751  | 10825#|
| SEN40   | 037024 | 7357  | 7377  | 7396  | 7415  | 10861#|
| SEN41   | 037036 | 7358  | 10862#|
| SEN42   | 037054 | 7378  | 10863#|
| SEN43   | 037072 | 7397  | 10864#|
| SEN44   | 037110 | 7416  | 10986#|
| SEN45   | 037126 | 7459  | 7467  | 7508  | 7515  | 10866#|
| SEN46   | 037136 | 7460  | 7468  | 10867#|
| SEN47   | 037154 | 7461  | 7510  | 10868#|</p>
<table>
<thead>
<tr>
<th>USER SYMBOL</th>
<th>LOCATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SENC8 037170</td>
<td>7469 10869#</td>
</tr>
<tr>
<td>SENC9 037204</td>
<td>7509 7516 10h70#</td>
</tr>
<tr>
<td>SEN5 036010</td>
<td>6771 10826#</td>
</tr>
<tr>
<td>SEN50 037222</td>
<td>7517 10871#</td>
</tr>
<tr>
<td>SEN51 037236</td>
<td>7584 7652 7720 7788 10872#</td>
</tr>
<tr>
<td>SEN52 037252</td>
<td>7585 7653 10873#</td>
</tr>
<tr>
<td>SEN53 037270</td>
<td>7586 7722 10874#</td>
</tr>
<tr>
<td>SEN54 037300</td>
<td>7654 10875#</td>
</tr>
<tr>
<td>SEN55 037310</td>
<td>7721 7789 10876#</td>
</tr>
<tr>
<td>SEN56 037330</td>
<td>7790 9/80 10877#</td>
</tr>
<tr>
<td>SEN57 037342</td>
<td>7817 7878 7878 7902 7966 7984 8037 8055 8108 8126 8178 8196 10878#</td>
</tr>
<tr>
<td>SEN58 037352</td>
<td>7818 10879#</td>
</tr>
<tr>
<td>SEN59 037372</td>
<td>7819 10880#</td>
</tr>
<tr>
<td>SEN6 036026</td>
<td>6792 10827#</td>
</tr>
<tr>
<td>SEN60 037412</td>
<td>7839 10881#</td>
</tr>
<tr>
<td>SEN61 037430</td>
<td>7840 10882#</td>
</tr>
<tr>
<td>SEN62 037450</td>
<td>7879 10883#</td>
</tr>
<tr>
<td>SEN63 037474</td>
<td>7880 10884#</td>
</tr>
<tr>
<td>SEN64 037504</td>
<td>7903 10885#</td>
</tr>
<tr>
<td>SEN65 037524</td>
<td>7904 10886#</td>
</tr>
<tr>
<td>SEN66 037532</td>
<td>7967 7985 8038 8056 8109 8127 8179 8197 10887#</td>
</tr>
<tr>
<td>SEN67 037546</td>
<td>7968 7986 10888#</td>
</tr>
<tr>
<td>SEN68 037570</td>
<td>7987 8058 8128 8199 10889#</td>
</tr>
<tr>
<td>SEN69 037604</td>
<td>8039 10890#</td>
</tr>
<tr>
<td>SEN7 036052</td>
<td>6814 10828#</td>
</tr>
<tr>
<td>SEN70 037626</td>
<td>8057 10891#</td>
</tr>
<tr>
<td>SEN71 037650</td>
<td>8110 10892#</td>
</tr>
<tr>
<td>SEN72 037672</td>
<td>8180 8198 10893#</td>
</tr>
<tr>
<td>SEN73 037714</td>
<td>8252 8308 8362 8416 10894#</td>
</tr>
<tr>
<td>SEN74 037724</td>
<td>8253 10895#</td>
</tr>
<tr>
<td>SEN75 037742</td>
<td>8309 10896#</td>
</tr>
<tr>
<td>SEN76 037760</td>
<td>8363 10897#</td>
</tr>
<tr>
<td>SEN77 037776</td>
<td>8417 10898#</td>
</tr>
<tr>
<td>SEN78 040014</td>
<td>8470 8477 8484 10899#</td>
</tr>
<tr>
<td>SEN79 040026</td>
<td>8471 8478 8485 10900#</td>
</tr>
<tr>
<td>SEN8 036076</td>
<td>6838 10829#</td>
</tr>
<tr>
<td>SEN80 040054</td>
<td>8472 10901#</td>
</tr>
<tr>
<td>SEN81 040100</td>
<td>8479 10902#</td>
</tr>
<tr>
<td>SEN82 040124</td>
<td>8486 10903#</td>
</tr>
<tr>
<td>SEN83 040146</td>
<td>8512 8540 10904#</td>
</tr>
<tr>
<td>SEN84 040156</td>
<td>8513 10905#</td>
</tr>
<tr>
<td>SEN85 040174</td>
<td>8541 10906#</td>
</tr>
<tr>
<td>SEN86 040204</td>
<td>8542 10907#</td>
</tr>
<tr>
<td>SEN87 040222</td>
<td>8543 10908#</td>
</tr>
<tr>
<td>SEN88 040246</td>
<td>8566 8574 10909#</td>
</tr>
<tr>
<td>SEN89 040262</td>
<td>8567 8575 10910#</td>
</tr>
<tr>
<td>SEN9 036114</td>
<td>6839 6863 10830#</td>
</tr>
<tr>
<td>SEN90 040276</td>
<td>8568 10911#</td>
</tr>
<tr>
<td>SEN91 040310</td>
<td>8576 10912#</td>
</tr>
<tr>
<td>SEN92 040322</td>
<td>8621 8663 8705 8741 10913#</td>
</tr>
<tr>
<td>SEN93 040334</td>
<td>8622 10914#</td>
</tr>
<tr>
<td>SEN94 040356</td>
<td>8623 10915#</td>
</tr>
<tr>
<td>SEN95 040370</td>
<td>8664 1076#</td>
</tr>
<tr>
<td>SEN96 040406</td>
<td>8665 10-17#</td>
</tr>
<tr>
<td>SEN97 040426</td>
<td>8706 10918#</td>
</tr>
<tr>
<td>SEN98 040444</td>
<td>8707 10919#</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>VALUE</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>SEN99</td>
<td>040464</td>
</tr>
<tr>
<td>SETEN</td>
<td>005352</td>
</tr>
<tr>
<td>SLASH</td>
<td>004426</td>
</tr>
<tr>
<td>SRO</td>
<td>177572</td>
</tr>
<tr>
<td>START</td>
<td>000500</td>
</tr>
<tr>
<td>STD</td>
<td>004112</td>
</tr>
<tr>
<td>TID</td>
<td>000646</td>
</tr>
<tr>
<td>TIME</td>
<td>004702</td>
</tr>
<tr>
<td>TITLE</td>
<td>000606</td>
</tr>
<tr>
<td>TRAP</td>
<td>004642</td>
</tr>
<tr>
<td>TSTFLG</td>
<td>004126</td>
</tr>
<tr>
<td>TST001</td>
<td>006716</td>
</tr>
<tr>
<td>TST002</td>
<td>007006</td>
</tr>
<tr>
<td>TST003</td>
<td>007056</td>
</tr>
<tr>
<td>TST004</td>
<td>007126</td>
</tr>
<tr>
<td>TST005</td>
<td>007176</td>
</tr>
<tr>
<td>TST006</td>
<td>007252</td>
</tr>
<tr>
<td>TST007</td>
<td>007326</td>
</tr>
<tr>
<td>TST010</td>
<td>007410</td>
</tr>
<tr>
<td>TST011</td>
<td>007472</td>
</tr>
<tr>
<td>TST012</td>
<td>007554</td>
</tr>
<tr>
<td>TST013</td>
<td>007640</td>
</tr>
<tr>
<td>TST014</td>
<td>010030</td>
</tr>
<tr>
<td>TST015</td>
<td>010114</td>
</tr>
<tr>
<td>TST016</td>
<td>010176</td>
</tr>
<tr>
<td>TST017</td>
<td>010252</td>
</tr>
<tr>
<td>TST020</td>
<td>010530</td>
</tr>
<tr>
<td>TST021</td>
<td>010624</td>
</tr>
<tr>
<td>TST022</td>
<td>010506</td>
</tr>
<tr>
<td>TST023</td>
<td>010562</td>
</tr>
<tr>
<td>TST024</td>
<td>010644</td>
</tr>
<tr>
<td>TST025</td>
<td>010712</td>
</tr>
<tr>
<td>TST026</td>
<td>010774</td>
</tr>
<tr>
<td>TST027</td>
<td>011050</td>
</tr>
<tr>
<td>TST030</td>
<td>011124</td>
</tr>
<tr>
<td>TST031</td>
<td>011200</td>
</tr>
<tr>
<td>TST032</td>
<td>011254</td>
</tr>
<tr>
<td>TST033</td>
<td>011350</td>
</tr>
<tr>
<td>TST034</td>
<td>011404</td>
</tr>
</tbody>
</table>

**Cross Reference Table -- User Symbols**

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEN99</td>
<td>040464</td>
</tr>
<tr>
<td>SETEN</td>
<td>005352</td>
</tr>
<tr>
<td>SLASH</td>
<td>004426</td>
</tr>
<tr>
<td>SRO</td>
<td>177572</td>
</tr>
<tr>
<td>START</td>
<td>000500</td>
</tr>
<tr>
<td>STD</td>
<td>004112</td>
</tr>
<tr>
<td>TID</td>
<td>000646</td>
</tr>
<tr>
<td>TIME</td>
<td>004702</td>
</tr>
<tr>
<td>TITLE</td>
<td>000606</td>
</tr>
<tr>
<td>TRAP</td>
<td>004642</td>
</tr>
<tr>
<td>TSTFLG</td>
<td>004126</td>
</tr>
<tr>
<td>TST001</td>
<td>006716</td>
</tr>
<tr>
<td>TST002</td>
<td>007006</td>
</tr>
<tr>
<td>TST003</td>
<td>007056</td>
</tr>
<tr>
<td>TST004</td>
<td>007126</td>
</tr>
<tr>
<td>TST005</td>
<td>007176</td>
</tr>
<tr>
<td>TST006</td>
<td>007252</td>
</tr>
<tr>
<td>TST007</td>
<td>007326</td>
</tr>
<tr>
<td>TST010</td>
<td>007410</td>
</tr>
<tr>
<td>TST011</td>
<td>007472</td>
</tr>
<tr>
<td>TST012</td>
<td>007554</td>
</tr>
<tr>
<td>TST013</td>
<td>007640</td>
</tr>
<tr>
<td>TST014</td>
<td>010030</td>
</tr>
<tr>
<td>TST015</td>
<td>010114</td>
</tr>
<tr>
<td>TST016</td>
<td>010176</td>
</tr>
<tr>
<td>TST017</td>
<td>010252</td>
</tr>
<tr>
<td>TST020</td>
<td>010530</td>
</tr>
<tr>
<td>TST021</td>
<td>010624</td>
</tr>
<tr>
<td>TST022</td>
<td>010506</td>
</tr>
<tr>
<td>TST023</td>
<td>010562</td>
</tr>
<tr>
<td>TST024</td>
<td>010644</td>
</tr>
<tr>
<td>TST025</td>
<td>010712</td>
</tr>
<tr>
<td>TST026</td>
<td>010774</td>
</tr>
<tr>
<td>TST027</td>
<td>011050</td>
</tr>
<tr>
<td>TST030</td>
<td>011124</td>
</tr>
<tr>
<td>TST031</td>
<td>011200</td>
</tr>
<tr>
<td>TST032</td>
<td>011254</td>
</tr>
<tr>
<td>TST033</td>
<td>011350</td>
</tr>
<tr>
<td>TST034</td>
<td>011404</td>
</tr>
<tr>
<td>TST035</td>
<td>017660</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>TST036</td>
<td>017554</td>
</tr>
<tr>
<td>TST037</td>
<td>016500</td>
</tr>
<tr>
<td>TST040</td>
<td>017174</td>
</tr>
<tr>
<td>TST041</td>
<td>012002</td>
</tr>
<tr>
<td>TST042</td>
<td>012050</td>
</tr>
<tr>
<td>TST043</td>
<td>012116</td>
</tr>
<tr>
<td>TST044</td>
<td>012310</td>
</tr>
<tr>
<td>TST045</td>
<td>012503</td>
</tr>
<tr>
<td>TST046</td>
<td>013034</td>
</tr>
<tr>
<td>TST047</td>
<td>013366</td>
</tr>
<tr>
<td>TST050</td>
<td>013724</td>
</tr>
<tr>
<td>TST051</td>
<td>014262</td>
</tr>
<tr>
<td>TST052</td>
<td>014356</td>
</tr>
<tr>
<td>TST055</td>
<td>014466</td>
</tr>
<tr>
<td>TST054</td>
<td>014616</td>
</tr>
<tr>
<td>TST055</td>
<td>014746</td>
</tr>
<tr>
<td>TST056</td>
<td>015302</td>
</tr>
<tr>
<td>TST057</td>
<td>015636</td>
</tr>
<tr>
<td>TST060</td>
<td>016710</td>
</tr>
<tr>
<td>TST061</td>
<td>016524</td>
</tr>
<tr>
<td>TST062</td>
<td>016712</td>
</tr>
<tr>
<td>TST063</td>
<td>017100</td>
</tr>
<tr>
<td>TST064</td>
<td>017266</td>
</tr>
<tr>
<td>TST065</td>
<td>017454</td>
</tr>
<tr>
<td>TST066</td>
<td>017750</td>
</tr>
<tr>
<td>TST067</td>
<td>020046</td>
</tr>
<tr>
<td>TST070</td>
<td>020160</td>
</tr>
<tr>
<td>TST071</td>
<td>020374</td>
</tr>
<tr>
<td>TST072</td>
<td>020560</td>
</tr>
<tr>
<td>TST073</td>
<td>020746</td>
</tr>
<tr>
<td>TST074</td>
<td>021134</td>
</tr>
<tr>
<td>TST075</td>
<td>021276</td>
</tr>
<tr>
<td>TST076</td>
<td>021546</td>
</tr>
<tr>
<td>TST077</td>
<td>022112</td>
</tr>
<tr>
<td>TST100</td>
<td>022265</td>
</tr>
<tr>
<td>TST101</td>
<td>022452</td>
</tr>
<tr>
<td>TST102</td>
<td>022670</td>
</tr>
<tr>
<td>TST103</td>
<td>023170</td>
</tr>
<tr>
<td>TST104</td>
<td>023470</td>
</tr>
<tr>
<td>TST105</td>
<td>023650</td>
</tr>
<tr>
<td>TST106</td>
<td>024030</td>
</tr>
<tr>
<td>TST107</td>
<td>024162</td>
</tr>
<tr>
<td>TST110</td>
<td>024346</td>
</tr>
<tr>
<td>TST111</td>
<td>024540</td>
</tr>
<tr>
<td>TST112</td>
<td>024732</td>
</tr>
<tr>
<td>TST113</td>
<td>025124</td>
</tr>
<tr>
<td>TST114</td>
<td>025316</td>
</tr>
<tr>
<td>TST115</td>
<td>025702</td>
</tr>
<tr>
<td>TST116</td>
<td>026166</td>
</tr>
<tr>
<td>TST117</td>
<td>026452</td>
</tr>
<tr>
<td>TST118</td>
<td>026644</td>
</tr>
<tr>
<td>TST121</td>
<td>027442</td>
</tr>
<tr>
<td>TST122</td>
<td>030564</td>
</tr>
<tr>
<td>TST123</td>
<td>031706</td>
</tr>
<tr>
<td>TST124</td>
<td>032324</td>
</tr>
<tr>
<td>Symbol</td>
<td>Value</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>TST125</td>
<td>032742</td>
</tr>
<tr>
<td>TST126</td>
<td>033334</td>
</tr>
<tr>
<td>TST127</td>
<td>033334</td>
</tr>
<tr>
<td>TST128</td>
<td>034330</td>
</tr>
<tr>
<td>TST130</td>
<td>034330</td>
</tr>
<tr>
<td>TST131</td>
<td>034736</td>
</tr>
<tr>
<td>TST132</td>
<td>035126</td>
</tr>
<tr>
<td>TST133</td>
<td>035276</td>
</tr>
<tr>
<td>TST134</td>
<td>035446</td>
</tr>
<tr>
<td>TST135</td>
<td>035656</td>
</tr>
<tr>
<td>TYPE</td>
<td>006064</td>
</tr>
<tr>
<td>SAPTHD</td>
<td>000626</td>
</tr>
<tr>
<td>SBASE</td>
<td>000716</td>
</tr>
<tr>
<td>$CDW1</td>
<td>000722</td>
</tr>
<tr>
<td>$CDW2</td>
<td>000724</td>
</tr>
<tr>
<td>$CRDP</td>
<td>00670</td>
</tr>
<tr>
<td>$DDW0</td>
<td>000726</td>
</tr>
<tr>
<td>$DDW1</td>
<td>000730</td>
</tr>
<tr>
<td>$DDW2</td>
<td>000752</td>
</tr>
<tr>
<td>$DDW3</td>
<td>000754</td>
</tr>
<tr>
<td>$DDW4</td>
<td>000756</td>
</tr>
<tr>
<td>$DDW5</td>
<td>000760</td>
</tr>
<tr>
<td>$DDW6</td>
<td>000762</td>
</tr>
<tr>
<td>$DDW7</td>
<td>000764</td>
</tr>
<tr>
<td>$DDW8</td>
<td>000772</td>
</tr>
<tr>
<td>$DDW9</td>
<td>000774</td>
</tr>
<tr>
<td>$DEVC</td>
<td>000652</td>
</tr>
<tr>
<td>$DEVN</td>
<td>000720</td>
</tr>
<tr>
<td>$ENV</td>
<td>000662</td>
</tr>
<tr>
<td>$ENVM</td>
<td>000663</td>
</tr>
<tr>
<td>$ETABL</td>
<td>000662</td>
</tr>
<tr>
<td>$ETEND</td>
<td>000766</td>
</tr>
<tr>
<td>$FATAL</td>
<td>000644</td>
</tr>
<tr>
<td>$HIBTS</td>
<td>000626</td>
</tr>
<tr>
<td>$HAPR1</td>
<td>000674</td>
</tr>
<tr>
<td>$HAPR2</td>
<td>000700</td>
</tr>
<tr>
<td>$HAPR3</td>
<td>000704</td>
</tr>
<tr>
<td>$HAPR4</td>
<td>000710</td>
</tr>
<tr>
<td>$HUAL</td>
<td>000642</td>
</tr>
<tr>
<td>$HAMS1</td>
<td>000672</td>
</tr>
<tr>
<td>$HAMS2</td>
<td>000676</td>
</tr>
<tr>
<td>$HAMS3</td>
<td>000700</td>
</tr>
<tr>
<td>$HAMS4</td>
<td>000706</td>
</tr>
<tr>
<td>$HAPRS</td>
<td>000630</td>
</tr>
<tr>
<td>$HSGA0</td>
<td>000656</td>
</tr>
<tr>
<td>$HSGL0</td>
<td>000660</td>
</tr>
<tr>
<td>$HSGT0</td>
<td>000642</td>
</tr>
<tr>
<td>$HTYP1</td>
<td>000673</td>
</tr>
<tr>
<td>$HTYP2</td>
<td>000677</td>
</tr>
<tr>
<td>$HTYP3</td>
<td>000703</td>
</tr>
<tr>
<td>MTYP4</td>
<td>000707</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>SPASS</td>
<td>000650</td>
</tr>
<tr>
<td>SPASTM</td>
<td>000654</td>
</tr>
<tr>
<td>SSNPREG</td>
<td>000664</td>
</tr>
<tr>
<td>STSTN</td>
<td>000666</td>
</tr>
<tr>
<td>STSTM</td>
<td>000632</td>
</tr>
<tr>
<td>SUNIT</td>
<td>000654</td>
</tr>
<tr>
<td>SUNITM</td>
<td>000636</td>
</tr>
<tr>
<td>SUSWE</td>
<td>000666</td>
</tr>
<tr>
<td>SVECT1</td>
<td>000712</td>
</tr>
<tr>
<td>SVECT2</td>
<td>000714</td>
</tr>
<tr>
<td>.</td>
<td>052002</td>
</tr>
<tr>
<td>.</td>
<td>000626</td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
</tbody>
</table>
MACRO NAMES

COMMON 1528#
ENDCN 1540#
ESCAPE 1656#
GETPRI 1270#
GETSME 1727#
MUL 43.87#
NEWST 1587#
POP 2105#
PUSH 2097#
REPORT 5346#
SETPRI 1238#
SETUP 1506#
SKIP 1690#
SLASH 1480#
STARS 1469#
SUMSU 1418#
TYPEIN 2041#
TYPDEC 2011#
TYPNAM 1781#
TYPNUM 1978#
TYPECS 1931#
TYPLOT 1894#
TYPTRT 1848#
SSESCA 1669#
SSEEUM 1625#
SSTKIP 1703#
EQUAT 168#
HEADE 42#
X11 311#
SETUP 1172#
SWRH 184#
SACT 695#
SAPTB 4999# 5951# 6016 6018
SAPTH 5255# 5951# 6016
SAPTY 5450#
BASTA 5301#
SCATC 897#
SCMTA 1008#
SDB2D 4585#
SDB2O 4708#
SDIV 4488#
SEOP 2164#
SEPER 2644#
SERTT 2839#
SMULT 4425#
SPOWE 4137#
SRAND 4212#
SRDDE 3815#
SRDOC 3724#
SRFAD 3329#
SR2AZ 4852#
SSAVE 3890#
SSB2D 4669#
SSB2O 4770#
SSCOP 2398#
SSIZE 4265#