IDENTIFICATION

PRODUCT CODE: NAINDEC-1080-BNIT
PRODUCT NAME: DESYSTEM-1080 BASIC MEMORY TESTING
DATE RELEASED: JUNE 20, 1975
MAINTAINED BY: DIAGNOSTIC ENGINEERING
AUTHOR: JAMES KELLY

COPYRIGHT (C) 1975
DIGITAL EQUIPMENT CORPORATION
PAULSBORO, N.J. 08062

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

ACTUAL DISTRIBUTION OF THE SOFTWARE DESCRIBED IN THIS DOCUMENT WILL BE SUBJECT TO TERMS AND CONDITIONS TO BE ANNOUNCED ON SOME FUTURE DATE BY DIGITAL EQUIPMENT CORPORATION.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE OR EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.
## Table of Contents

1.0 ABSTRACTS

2.0 REQUIREMENTS
   2.1 EQUIPMENT
   2.2 STORAGE
   2.3 PRELIMINARY PROGRAMS

3.0 PROGRAM PROCEDURES
   3.1 LOADING PROCEDURES
   3.2 STARTING PROCEDURES

4.0 ERRORS
   4.1 ERROR PC RANGE

5.0 CYCLE TIME

6.0 PROGRAM DESCRIPTION

7.0 LISTING
1.0 ABSTRACT

The basic memory timing test (BMNT) is designed to verify that the basic memory cycle time of core memory is within tolerance. This is accomplished by comparing the speed of the KRL-1 real time clock versus the time required for a small set of instructions to be executed. No attempt is made to compute actual machine timing. A short verification of the KRL-1 is done prior to the timing test.

2.0 REQUIREMENTS

2.1 EQUIPMENT

A PDP-11/40 equipped with at least 4K of memory.
A DECtape drive for loading the program.
A console teletype to connect the monitor.
A KRL-1 real time clock.

2.2 STORAGE

This program occupies memory locations 1000(8) thru 1522(8). In addition, the stack pointer is set to 3600, and the appropriate interrupt vectors will be set up by the software prior to use.

2.3 PRELIMINARY PROGRAMS

All basic CPU instruction set diagnostic's must have been successfully run prior to attempting to run this program.
3.0 PROGRAM PROCEDURES

3.1 LOADING PROCEDURES

1. MOUNT THE DECTAPE CONTAINING THE BTTL DIAGNOSTIC TEST ONTO A WORKING DECTAPE DRIVE.
2. SET THE WRITE LOCK SWITCH TO WRITE LOCK.
3. SET THE DRIVES SELECT TO UNIT #0.
4. SET THE MODE SWITCH TO ONLINE.
5. SET 177072 INTO THE PDP-11/40 SWITCH REGISTER.
7. DEPRESS THE START KEY ON THE PDP-11/40 SEVERAL TIMES. THIS INITIALIZES THE CPU.
8. SET THE HALT KEY TO THE RUN POSITION (UP).
9. DEPRESS THE START KEY AND RELEASE ONE TIME.
10. THE SELECTED DECTAPE DRIVE WILL BEGIN TO TURN.
11. THE CONSOLE WILL TYPE THE MONITOR'S NAME FOLLOWED BY A "#".
12. TYPE OKNOWA.BIN <CR>.

3.2 STARTING PROCEDURES

TO START THE BTTL PROGRAM PROCEED AS FOLLOWS:

1. SET THE PDP-11/40 ADDRESS TO 1000.
2. DEPRESS AND RELEASE THE START KEY.
3. THE PROCESSOR WILL RUN FOR LESS THAN A SECOND AND HALT WITH ADDRESS 1614 DISPLAYED IN THE ADDRESS REGISTER.
4. IF THE PROCESSOR HALTS AT ANY ADDRESS OTHER THAN THE ABOVE, IT IS AN ERROR. REFER TO THE LISTING FOR A DETAILED DESCRIPTION OF THE SYMPTOM.
4.0 ERRORS

Any halt other than with 1514 displayed is an error and will have to be repaired before attempting any further testing. To troubleshoot the error, the error halt should be replaced with a "nop" and the program can then be restarted at the beginning of the failing subtest.

4.1 ERROR PC RANGE

Address(es) 1000 thru 1516 inclusive are legal error halts with the single exception of address 1512 displayed as 1514.

5.0 CYCLE TIME

The speed of execution is practically instantly.

6.0 PROGRAM DESCRIPTION

BRITT consists of five basic tests. The first four are a cursory check of the KILL-L REAL TIME CLOCK, the fifth being the actual timing test. The first four tests determine that the clock exists, that it will set the monitor bit, cause interrupts and that it can be locked out by clearing the interrupt enable bit. The final test waits for the clock to set the monitor bit thus giving us a complete cycle for our test. Then starting a simple time out loop, when the KILL-L causes an interrupt the number of iterations in the timeout loop are compared to a previously calculated normal range to determine that the machine speed is normal.

Basic memory timing tests flows
BASIC MEMORY TIMING TESTS FLOWS

ENTER

TO1:

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

* REGISTER SETUP
*
*******************************************************************************

* SET "HOT TRAP TO
* EDIA
*
*******************************************************************************

* SET "ILLEGAL COMMAND
* TRAP" TO
* EDIB
*
*******************************************************************************

* SET "KULL-L VECTOR
* TO
* EDIC
*
*******************************************************************************

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

SET SP TO 3200

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

SET PS TO PS7

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

NO

DOES LISSR EXIST?

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

I YES

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

I YES

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

NOTE

TOI DETERMINES IF THE KULL-L REAL TIME CSR REGIST
CAUSING A TIME OUT.
THE RESERVED INSTRUCTION AND NON-EXISTANT MEMORY SHOWN IN THIS TEST SHOULD NOT BE GIVEN IN THIS DIAGNOSTIC TEST. IT WILL BE ON THIS PAGE IT WILL BE NECESSARY TO STAMP TO DETERMINE WHERE THE TRAPED WAS CAUSED.
BASIC MEMORY TIMING TESTS FLOWS

TO3:  

REGISTER SETUP  

SET SP TO 3200  

I

IS KOHL-L MONITOR BIT SET  

NO

I YES

CLEAR KOHL-L MONITOR BIT

I

SET PS TO 90D

I

KOHL-L MONITOR BIT SET

I NO

TIMEOUT

I YES

ED3: HALT

MONITOR BIT FAILED

ED3A: HALT

UNEXPECTED KOHL-L
TITLE X
:XTIM

001000

:IO11-L REAL TIME CLOCK BIT DEFINITIONS

000200
:ALMON=200

000100
:KALEN=100

:LINE CLOCK INTERRUPT ENABLE

REGISTER ADDRESSES

177546
:LJSCR=177546

DEVICE VECTOR ASSIGNMENTS

000100
:LJVEC=100

000300
:LJSPRT=300

:COMMAND STATUS REGISTER ADDRESS

PRIORITY LEVELS EQUATES

177776
:P=177776

000200
:STACK=3200

:INTERRUPT VECTOR ADDRESS

:PROCESSOR STATUS.

:STACK PESET.

PRIORIY 6

000004
:RESVEC=4

000010
:RESVEC=10

:RESERVED INTRACTION TRAP.

BUS TIMEOUT TRAP (ILLEGAL OR NONEXISTANT MEMORY).
SETTL  IOUII-L DEVICE RESPONSE

THIS ROUTINE MAKES A BASIC CHECK OF THE IOUII-L REAL TIME CLOCK.
THE TESTS TEST AS FOLLOWS:

(T01) THAT WE CAN ACCESS THE "LKSCSR" WITHOUT A HCM TIMEOUT.

(T02) THAT THE MONITOR BIT IS SET BY THE LINE CURRENT AFTER
A REASONABLE PERIOD OF TIME.

(T03) THAT THE MONITOR BIT SET WITHOUT INTERRUPT ENABLE DOES NOT CAUSE
AN INTERRUPT.

(T04) THAT THE INTERRUPT ENABLE BIT WHEN SET WILL ALLOW INTERRUPTS.

(T05) THIS TEST DETERMINES THAT THE CPU CYCLE TIME IS WITHIN LIMITS.

-------------------------------------------------------------

T01:

681E08 681E34 000001

MOV E1, R0  ; AD-TEST NUMBER.
MOV E1, R1  ; E1-DEVICE (CLOCK E1).
MOV E2, R1  ; E2-TEST ADDRESS.
CLR R3  ; CLEAR BUS ADDRESS.
CLR R4  ; CLEAR SHOULD BE ADDRESS.

681E28 681E2C 000002

MOV E12, 000000  ; E12, SHREDIENT.
MOV F12, 000000  ; SET MON TRAP.
MOV 8F7, 000000  ; SET HIGN TRAP.
MOV 8F7, 000000  ; SET ILLEGAL COMMAND TRAP.

681E2C 681E30 000002

MOV E12, 000000  ; E12, SHREDIENT.
MOV F12, 000000  ; SET MON TRAP.
MOV 8F7, 000000  ; SET HIGN TRAP.
MOV 8F7, 000000  ; SET ILLEGAL COMMAND TRAP.

681E30 681E34 000003

MOV E12, 000000  ; E12, SHREDIENT.
MOV F12, 000000  ; SET MON TRAP.
MOV 8F7, 000000  ; SET HIGN TRAP.
MOV 8F7, 000000  ; SET ILLEGAL COMMAND TRAP.

681E34 681E34 000000

L01:  MOV 83, 0000  ; SET STACK TO 3000.
MOV 83, 0000  ; SET CPU PRIORITY TO 07.

681E34 681E38 000000

T01:  BR  L02  ; TEST FOR THE CLOCK.
JOTO NEXT TEST.

-------------------------------------------------------------

TEST E1.

(FC) (PS) (SP) TEST DEV6 ADDRESS HAS S/0
(R7) (PSH) (RS) (RO) (R1) (R2) (R3) (R4)
11D0 34H 31H 4 1 1 177546 0 0

-------------------------------------------------------------

E01A:  MHT  BR  L01  ; SHOWN EXISTANT MEMORY TRAP (NO CLOCK).
LOCK ON FAULT ERROR.
THIS ERROR SHOULD BE CAUSED
BY A TEST OTHER THAN "TEST 80".
CHECK THE CONTENTS OF THE STACK
TO DETERMINE IT'S ORIGIN.

TEST 01
(PC) (PS) (SP) TEST DEV0 ADDRESS HAS S/0
(R7) (PSH) (R6) (R0) (R1) (R2) (R3) (R4)
1110 340 3174 1 1 177546 0 0

ED1B: HALT BR LO1
; ILLEGAL COMMAND;
; LOCK ON FATAL ERROR;
; SEE COMMENT FOR PREVIOUS ERROR.

TEST 01
(PC) (PS) (SP) TEST DEV0 ADDRESS HAS S/0
(R7) (PSH) (R6) (R0) (R1) (R2) (R3) (R4)
1114 340 3174 1 1 177546 0 0

ED1C: HALT BR LO1
; UNEXPECTED IO41-L INTERRUPT;
; LOCK ON FATAL ERROR.
.TEXT 10211-L MONITOR BIT.

TO TEST WHETHER THE LINE CURRENT EVER SETS THE MONITOR BIT. THIS IS ACCOMPLISHED AS FOLLOWS:

1. TEST TO SEE IF THE MONITOR BIT IS ALREADY SET, IT PROBABLY WILL BE.
2. IF THE MONITOR BIT WAS SET CLEAR IT.
3. CLEAR A TIMER REGISTER FOR MAXIMUM WAITING TIME.
4. WAIT AROUND IN A TIME LOOP FOR THE FLAG TO SET.
5. IF IT FAILS TO SET PRIOR TO TIMEOUT THEN HALT.

.102: MOV #0, R0 ; NO-TEST NUMBER
       MOV R7, R0 ; ADDRESS TO R7.
       MOV R8, R0 ; SHOULD BE.

.L02: MOV @R7, @R8 ; SET IO11-L INTERRUPT VECTOR.
       MOV R4, @R8+2 ; SET Priority to 07.

.L02: MOV @SP, @R4+2 ; SET STACK TO 3200.
       BRA R5 ; CLEAR TIMER.

.R02: BRA R3, (R2) ; MONITOR BIT SET?
       BRA R5 ; YES!

.R02: BRA R3, (R2) ; CLEAR MONITOR BIT.

.R02: BRA R3, (R2) ; CONTENTS OF LISR TO R3.
       BRA R3, (R2) ; MONITOR BIT SET?
       BRA R5 ; YES! GOTO "103"
       BRA R5 ; NO!

.INC R5 ; TIMEOUT OVER?
       BRA R5 ; NO! CHECK MONITOR BIT AGAIN.

; TEST R2

INC R7 ; (PC) (PS) (SP) TEST DEV0 ADDRESS HAS SHD BE
INC R6 ; (R7) (PSH) (R5) (R2) (R1) (R2) (R3) (R4)

1202 240 3200 2 1 177546 ????? 200

; THIS MONITOR BIT FAILED TO SET.
; LOCK ON FATAL ERROR.
<table>
<thead>
<tr>
<th>(PC)</th>
<th>(PS)</th>
<th>(SP)</th>
<th>TEST</th>
<th>DEV#</th>
<th>ADDRESS WAS</th>
<th>S/B</th>
</tr>
</thead>
<tbody>
<tr>
<td>(R7)</td>
<td>(PSH)</td>
<td>(R6)</td>
<td>(R0)</td>
<td>(R1)</td>
<td>(R2)</td>
<td>(R3)</td>
</tr>
<tr>
<td>1206</td>
<td>340</td>
<td>3174</td>
<td>2</td>
<td>1</td>
<td>177546</td>
<td>???</td>
</tr>
</tbody>
</table>

**ED2A:** HALT

**BR 102**

;**UNEXPECTED IO11 - L INTERRUPT.**

;**LOCK ON FATAL ERROR.**
J02

SETTL, IO411-L INTERRUPT ENABLE BIT.

1) VERIFY THAT THE MONITOR ENABLE BIT BECOMES SET WITH
THE CPU PRIORITY AT 0. IT WILL NOT CAUSE AN INTERRUPT UNLESS INTERRUPT
ENABLE IS ALSO SET, AS FOLLOWS:

1). TEST TO SEE IF MONITOR BIT IS SET.
2). IF THE MONITOR BIT WAS SET THEN CLEAR IT.
3). SET CPU PRIORITY TO #0.
4). CLEAR THE TIMER FOR MAXIMUM WAITING TIME.
5). WAIT FOR THE MONITOR BIT TO SET IF IT DOESNT SET
PRIOR TO THE TIMEOUT THAN HALT.

C03: MOV R3, R0 ; RC-TEST NUMBER.
C03: MOV R1, R1 ; DEVICE (START 80).
C03: MOV AS16, R2 ; ADDRESS TO REG.
C03: MOV B16, RIN ; SHOULD BE.
C03: ED16, R16, 1566 ; SET IO411 - L INTERRUPT VECTOR.
001236 012706 003200
C03: MOV & stack, SP ; SET THE STACK POINTER TO 3200.

001246 0040412
C03: MOV R4, (R2) ; MONITOR BIT SET?
C03: MOV R03 ; RC.
C03: MOV R4, (R2) ; CLEAR MONITOR BIT.

C03: CLR R5 ; SET CPU PRIORITY TO PND.
C03: CLR R5 ; CLEAR TIMER.

RD3: MOV (R2), R3 ; CONTENTS OF LPSGVR TO R3.
RD3: MOV R4, (R2) ; MONITOR BIT SET?
RD3: MOV R04 ; "GO! GOTO "TOF"
RD3: MOV R5 ; "TIMEOUT OVERT"
RD3: MOV R03 ; "NO! CHECK MONITOR BIT AGAIN.

TEST R0

(PC) (PS) (SP) TEST DEVB ADDRESS WAS SHOULD BE

(R7) (PSM) (R6) (R0) (R1) (R2) (R3) (R4)

1272 0 3200 3 1 177546 ?????? 200

001270 000000
EO0: HALT ;### MONITOR BIT FAILED TO SET.
K02

BR LO3 ;LOCK ON FATAL ERROR.

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

TEST 03

(PC) (PS) (SP) TEST DEV# ADDRESS WAS S/0

(R7) (R6) (R5) (R4) (R3) (R2) (R1)

1276 340 3174 1 1 17754 ????? 200

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

EO2A: HLT LO3 ;UNEXPECTED IOU1 - L INTERRUPT.

;LOCK ON FATAL ERROR.
SETTL 1011-L INTERRUPT.

This verifies that the 1011-L will cause an interrupt when the enable bit is set and the line current causes the monitor bit to be set, and the CPU priority is set at 05 as follows.

1. Set up 1011-L clock to trap to next test.
2. If monitor bit is set, clear it.
3. Set interrupt enable bit.
4. Clear the timer register for maximum waiting time.
5. Test to see if the monitor bit is set.
6. If we get interrupted we will go to next test.
Otherwise, error halt.

TOP:
MOV #M, R0
MOV #I, R1
MOV 11, R2
MOV #1000, R3
MOV #000100, R5

LOP:
MOV #STACK, SP
MOV #PR, R1
MOV #PRG, R1
MOV RH, (R2)
MVI R1, RH
MVI R2, RH
BIC RH, (R2)

NOV: MOV #GOALENA, (R2)

NOV: MOV (R2), R3

OUT RH, (R2)
OUT ED, (R3)

INC RS
ONE RN

TEST #M

(PC) (PS) (SP) TEST DEV# ADDRESS HAS SHO BE
(R7) (PSM) (R6) (R0) (R1) (R2) (R3) (R4)

1366 240 3200 4 1 177546 ???? 200
; FATAL ERROR
; HANG ON FATAL ERROR.

; SET PRIORITY TO 87.
**NO2.**

**KHIIL-FETCH CYCLE TIMING.**

`TOS` determines the processor cycle time. This is accomplished as follows:

1. The KHIIL is monitored until we get to the next monitor bit set.
2. Clear the monitor bit.
3. Clear the timer (R5).
4. Set CPU priority to R5.
5. Wait for the interrupt while in the timeout loop.
6. Check that the time spent waiting for the interrupt was within tolerance.
7. The code will run until we have serviced 1 interrupt.

---

```
001276  012700  000005
001426  012701  000001
001426  012702  177556
001426  012704  000020
001416  012706  003200
001424  012737  001476  000100
001426  014203  001004
001426  014203  001004
001426  003905  001379
```

---

```
TOS: MOV A6, R0; R0=TEST ADDRESS.
      MOV A1, R1; R1=DEVICE (CLOCK #1).
      MOV A2, R2; ADDRESS TO R2.
      SHL R4, R4; SHOULD BE.
      MOV A3, R3; CLEAR TIMER.
      MOV A5, 00; SET UP INTERRUPT VECTOR

LOG: MOV @STACK, SP; RESET STACK POINTER.
      CLR R5; CLEAR TIMER.

POS: MOV (R2), R3; CONTENTS OF LKSCSR TO R3.
      BNE R6; R6 (R2); YES
      INC R5; WAIT.
      BNE POS; NO!
```

---

**TEST #6**

(PC) (PS) (SP) TEST DEV# ADDRESS WAS SHD BE
(R7) (PSM) (R6) (R0) (R1) (R2) (R3) (R4)

```
01446  340  3200  5  1  1775% ?????  200
```

---

```
EO5: HALT
      BR LOG; #HOMONITOR BIT FAILED TO SET.
      BSET; LOCK ON FATAL ERROR.
```
BO3

BEGIN: 

001256 002000 000100 
NO5: JRS RN,(R2) 
| clear monitor bit 
| set interrupt enable bit.

001456 000100 
CLR R5 
| clear timer.

001460 012727 000840 177776 
MOV OP1,OP2 
| set CPU to priority 06.

15: INC R5 
| await an interrupt.

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

TEST 05 
(PC) (PS) (SP) TEST DEV# ADDRESS WAS SHD BE 
(R7) (PSM) (R6) (R5) (R1) (R2) (R3) (R4) 
1470 240 3200 5 1 177776 ?????? 200 
******************************************************************************

001746 000000 
OER: HALT B0 
| check no interrupt.

001756 000020 
OER: CRP RES,HALT 
| check for system error.

001760 000040 
CRP B0,FAST 
| check FAST.

001765 000060 
CRP B0,SL-O 
| check SL-OVM.

001769 000080 
HALT 
| check HALT.

001773 000080 
HALT 
| check HALT.

001777 000080 
HALT 
| check HALT.

001781 000080 
HALT 
| check HALT.

END