

OUTLINE FOR BCI-2100 INSTALLATION AND VERIFICATION

1. Install the BCI-2100 module. Refer to the BCI-2100 Owners Manual.
2. Install the BCI-2100 TREX diagnostic, **MED-0028-A** or **MED-0028-B**

The distribution media contains a save-set with the appropriate directories in OpenVMS backup format. Position at any directory and set privileges for **BYPA CMKR**, and **PFNM**, and load the tape via the following command.

```
$ backup/log/verify mka500:bci2100.sav/label=bci/select=[deeter...] [...]
```

The directory tree will be:
[.your_dir.trex.bci2100]

3. Determine the adapter and node for the BCI-2100.

Run the utility program "show_pci" located in [your_dir.trex] and record the adapter number and the node number of the BCI-2100 module for configuring the BCI-2100 TREX diagnostic environment.

4. Run the diagnostic "bci2100" located in [your_dir.trex.bci2100] and answer the adapter number and node number questions. At this point we are in a "shell" with a prompt of ">>", which is capable of executing BCI-2100 tests and performing examine/deposit operations on BCI-2100 registers. The TREX diagnostic environment is patterned after the console program (REX) of the DecStation 5000 system. The following commands will provide an introduction to TREX.

>>?	lists TREX commands
>>e?	lists examine/deposit commands
>>t 1/?	lists the tests available for the device
>>t 1/reg 100	execute the register test 100 times
>>sh.-e -v 1/test	run the non-loopback test suite
	-e is stop on error; -v is print each test command

Note: The slot # (1) specified in the t command is ignored by TREX and can be any value. It may be appropriate for operators to use the slot number displayed by the show_pci utility.

5. These programs use model specific privileged resources and may require rebuilding if they do not operate properly. To rebuild the diagnostic environment execute the command file bci2100.com in [your_dir.trex.bci2100]. To rebuild the show_pci utility execute the command file show_pci.com in [your_dir.trex].

OUTLINE FOR BCI-2100 INSTALLATION AND VERIFICATION

1. Install the BCI-2100 module. Refer to the BCI-2100 Owners Manual.
2. Install the BCI-2100 TREX diagnostic, **MED-0028-C**

The distribution media contains a save-set with the appropriate directories in OpenVMS backup format. Position at any directory and set privileges for BYPA CMKR, and PFMN, and load the tape via the following command.

```
$ mount dva0: bci
$ backup/log/verify dva0:[000000]bci2100.sav/save_set/select=[deeter...] [...]
$ dismount dva0:
```

The directory tree will be:
[.your_dir.trex.bci2100]

3. Determine the adapter and node for the BCI-2100.

Run the utility program "show_pci" located in [.your_dir.trex] and record the adapter number and the node number of the BCI-2100 module for configuring the BCI-2100 TREX diagnostic environment.

4. Run the diagnostic "bci2100" located in [.your_dir.trex.bci2100] and answer the adapter number and node number questions. At this point we are in a "shell" with a prompt of ">>", which is capable of executing BCI-2100 tests and performing examine/deposit operations on BCI-2100 registers. The TREX diagnostic environment is patterned after the console program (REX) of the DecStation 5000 system. The following commands will provide an introduction to TREX.

>>?	lists TREX commands
>>e?	lists examine/deposit commands
>>t 1/?	lists the tests available for the device
>>t 1/reg 100	execute the register test 100 times
>>sh -e -v 1/test	run the non-loopback test suite
	-e is stop on error, -v is print each test command

Note: The slot # (1) specified in the t command is ignored by TREX and can be any value. It may be appropriate for operators to use the slot number displayed by the show_pci utility.

5. These programs use model specific privileged resources and may require rebuilding if they do not operate properly. To rebuild the diagnostic environment execute the command file bci2100.com in [.your_dir.trex.bci2100]. To rebuild the show_pci utility execute the command file show_pci.com in [.your_dir.trex].

BCI-2100 Diagnostics

regtest -- Register test for BCI-2100

format reg passes

This test checks that all read/write register bits operate properly. A floating one, floating zero pattern is used. The pattern ends with all ones, all zeros words.

A table containing register names and read/write bit masks is used to control this the test. Only bits set in the masks are written or read by this test.

There is one option for this test, a pass count.
The default is to run one pass.

memtest -- Local buffer memory test for BCI-2100

format mem passes dwb

This test checks the 256kb or 1mb on-board local memory. If no local memory is found, a message is printed and the test is skipped. One subtest tests the first location with a data pattern. A second subtest test the entire memory with an address pattern. The third subtest uses the first location to test byte writes.

There are two options for this test. The first is the pass count
The second is a list of subtests to run, 'd' for the data test, 'w' for the address test, and 'b' for the byte test.

Note: The subtests may specified in any order.
The default is to run one pass with all three subtests.

inttest -- Interrupt test for BCI-2100

(Test board is only required for subtest)

format int passes b

This test checks that a PCI interrupt can be generated by writing the XER bit in the csr when in maintenance mode. A subtest uses the test board to generate a PCI interrupt via a qbus interrupt.

There are two options for this test. The first is a pass count.
The second is a 'b' for selecting the test board interrupt test.

The default is one pass without the test board interrupt.

ctltest -- Qbus control signal test for BCI-2100

format ctl passes

This test uses the maintenance register to check that DCLO, ACLO, MSYN, and SSYN can be set and cleared, and that a grant timeout will occur. It also addresses the beginning of I/O space to generate a bus timeout.

There is one option for this test, a pass count.
The default is to run one pass with both subtests.

piotest -- PIO test for BCI-2100 (Test board is required)

format pio passes

This test checks word and byte transfers on the qbus by writing and reading to the data register on the test board. A floating one, floating zero data pattern is used by the word test. The pattern ends with all ones, all zeros words. The byte test uses an AA 55 pattern.

There is one option for this test, a pass count.
The default is to run one pass with both subtests.

dmadtest -- DMA direct to PCI test for BCI-2100 (Test board is required)

format dmad passes byte_count zwrpb

This test checks that data can be transmitted and received using DMA's between system memory via the PCI bus and the test board on the qbus. A 8-bit or 16-bit incrementing data pattern is used by this test (byte or word transfers).

There are three options for this test. The first is a pass count. The second is the number of 8-bit bytes to transfer. The default is 4 bytes, the minimum is 2 bytes, and the maximum is 65535 bytes. The third is a list of test modifiers; 'z' will cause the system memory buffer to be zeroed, 'w' will cause the test board to do a DMA write to system memory, 'r' will cause the test board to do a DMA read/verify from system memory, 'b' will make all writes operate in byte mode, and 'p' will cause a test board status register polling for DMA termination rather than use an interrupt.

Note: The modifiers may be entered in any order, but the order of execution will be, zero, write, and then read/verify.
The default is one pass with write, read/verify and no (zero,byte,poll).

dmaltest -- DMA to local simm test for BCI-2100

(Test board is required)

format dmal passes byte_count zwrbps

This test checks that data can be transmitted and received using DMA's between the on-board buffer memory simm and the test board on the qbus. If no local memory is found, a message is printed and the test is skipped. An 8-bit or 16-bit (byte/word) incrementing data pattern is used by this test.

There are three options for this test. The first is a pass count. The second is the number of 8-bit bytes to transfer. The default is 4 bytes, the minimum is 2 bytes, and the maximum is 65535 bytes. The third is a list of test modifiers; 'z' will cause the local memory buffer to be zeroed, 'w' will cause the test board to do a DMA write to local memory, 'r' will cause the test board to do a DMA read/verify from local memory, 'b' will make all writes operate in byte mode, 'p' will cause a test board status register polling for DMA termination rather than use an interrupt, 's' will use the pass count, from 0 to n, as a 65kb page number to step the DMA buffer through the local memory by selecting and loading the appropriate page register.

Note: The modifiers may be entered in any order, but the order of execution will be, zero, write, and then read/verify. The default is one pass with write, read/verify and no (zero,byte,poll,step).