|  |  |  |
| --- | --- | --- |
| Bochs Developers Guide | | |
| [Prev](http://docs.google.com/add-keymapping.html) |  | [Next](http://docs.google.com/instrumentation.html) |

Chapter 3. Advanced debugger usage

3.1. I/O Interface to Bochs Debugger

This device was added by Dave Poirier (eks@void-core.2y.net).

Compiling Bochs with iodebug support

./configure --enable-iodebug  
make

Other optional fields may be added to the ./configure line, see Bochs documentation for all the information. To enable the iodebug plugin at runtime, it must be loaded with the 'plugin\_ctrl' bochsrc option.

Using the I/O Interface to the debugger  
  
port range: 0x8A00 - 0x8A01  
  
Port 0x8A00 servers as command register. You can use it to enable the i/o interface,  
change which data register is active, etc.  
  
Port 0x8A01 is used as data register for the memory monitoring.

3.1.1. Commands supported by port 0x8A00

0x8A00  
  
 Used to enable the device. Any I/O to the debug module before this command is sent  
 is sent will simply be ignored.  
  
  
0x8A01  
  
 Selects register 0: Memory monitoring range start address (inclusive)  
  
  
0x8A02  
  
 Selects register 1: Memory monitoring range end address (exclusive)  
  
  
0x8A80  
  
 Enable address range memory monitoring as indicated by register 0 and 1 and  
 clears both registers  
  
  
0x8AE0 - Return to Debugger Prompt  
  
 If the debugger is enabled (via --enable-debugger), sending 0x8AE0 to port 0x8A00  
 after the device has been enabled will return the Bochs to the debugger prompt.  
 Basically the same as doing CTRL+C.  
  
  
0x8AE2 - Instruction Trace Disable  
  
 If the debugger is enabled (via --enable-debugger), sending 0x8AE2 to port 0x8A00  
 after the device has been enabled will disable instruction tracing  
  
  
0x8AE3 - Instruction Trace Enable  
  
 If the debugger is enabled (via --enable-debugger), sending 0x8AE3 to port 0x8A00  
 after the device has been enabled will enable instruction tracing  
  
  
0x8AE4 - Register Trace Disable  
  
 If the debugger is enabled (via --enable-debugger), sending 0x8AE4 to port 0x8A00  
 after the device has been enabled will disable register tracing.  
  
  
0x8AE5 - Register Trace Enable  
  
 If the debugger is enabled (via --enable-debugger), sending 0x8AE5 to port 0x8A00  
 after the device has been enabled will enable register tracing. This currently  
 output the value of all the registers for each instruction traced.  
 Note: instruction tracing must be enabled to view the register tracing  
  
  
0x8AFF  
  
 Disable the I/O interface to the debugger and the memory monitoring functions.

**Note:** all accesses must be done using word

**Note:** reading this register will return 0x8A00 if currently activated, otherwise 0

3.1.2. Access to port 0x8A01 (write-only)

All accesses to this port must be done using words. Writing to this port will shift to the left by 16 the current value of the register and add the provided value to it.

Sample:  
  
reg0 = 0x01234567  
  
out port: 0x8A01 data: 0xABCD  
  
reg0 = 0x4567ABCD

3.1.3. Sample

Enable memory monitoring on first page of text screen (0xb8000-0xb8fa0): add in bochrc file: **optromimage1: file="asmio.rom", address=0xd0000**

/\*  
 \* Make asmio ROM file:  
 \* gcc -c asmio.S  
 \* objcopy -O binary asmio.o asmio.rom  
 \*/  
 .text  
 .global start  
 .code16  
  
/\* ROM Header \*/  
 .byte 0x55  
 .byte 0xAA  
 .byte 1 /\* 512 bytes long \*/  
  
start:  
/\* Monitor memory access on first page of text screen \*/  
 mov $0x8A00,%dx /\* Enable iodebug (0x8A00->0x8A00) \*/  
 mov %dx,%ax  
 out %ax,%dx  
 mov $0x8A01,%ax /\* Select register 0 start addr (0x8A01->0x8A00) \*/  
 out %ax,%dx  
 mov $0x8A01,%dx /\* Write start addr 0xB8000 (high word first) \*/  
 mov $0xB,%ax  
 out %ax,%dx  
 mov $0x8000,%ax /\* Write start addr (low word) \*/  
 out %ax,%dx  
  
 mov $0x8A02,%ax /\* Select register 1 end addr (0x8A02->0x8A00) \*/  
 mov $0x8A00,%dx  
 out %ax,%dx  
 mov $0x8A01,%dx /\* Write end addr 0xB8FA0 (high word first) \*/  
 mov $0xB,%ax  
 out %ax,%dx  
 mov $0x8FA0,%ax /\* Write end addr (low word) \*/  
 out %ax,%dx  
  
 mov $0x8A00,%dx /\* Enable addr range memory monitoring (0x8A80->0x8A00) \*/  
 mov $0x8A80,%ax  
 out %ax,%dx  
  
 mov $0x8A00,%dx /\* Return to Bochs Debugger Prompt (0x8AE0->0x8A00) \*/  
 mov $0x8AE0,%ax  
 out %ax,%dx  
 lret  
  
 .byte 0x6b /\* Checksum (code dependent!, update it as needed) \*/  
 .align 512 /\* NOP follow \*/

|  |  |  |
| --- | --- | --- |
| [Prev](http://docs.google.com/add-keymapping.html) | [Home](http://docs.google.com/index.html) | [Next](http://docs.google.com/instrumentation.html) |
| How to add keymapping in a GUI client |  | The instrumentation feature |