bochs调试FAQ:

1.       Q:如何从引导扇区开始设置断点?

A: BIOS被载入内存中运行,其引导扇区的地址一般都是在0x7c00,因为在实模式下,逻辑地址和物理地址是一一对应的, 所以我们可以在启动bochs调试的命令行下输入

*pb 0x7c00* 或者*vb 0:0x7c00*

就着输入

        c

bochs就会在0x7c00处停下

2.       Q: 我想在单步执行(s)后看到寄存器的变化情况, 该怎么设置?

A: 输入

               trace-reg on

   后, 再执行单步调试的时候都会显示寄存器的当前状态了.

3.       Q: 在单步调试的时候,遇到一个调用子程序的指令,输入s会把调试跳进子程序中,如何只执行子程序而不要陷进去调试?

A: 在调试子程序的指令处,不用s改为n或p

4.       Q: 查看当前堆栈的命令?

A: print-stack

Table 1. 部分Bochs调试指令

| **行为** | **指令** | **举例** |
| --- | --- | --- |
| 在某物理地址设置断点 | b addr | b 0x30400 |
| 显示当前所有断点信息 | info break | info break |
| 继续执行，直到遇上断点 | c | c |
| 单步执行 | s | s |
| 单步执行（遇到函数则跳过） | n | n |
| 查看寄存器信息 | info cpu  r  fp  sreg  creg | info cpu  r  fp  sreg  creg |
| 查看堆栈 | print-stack | print-stack |
| 查看内存物理地址内容 | xp /nuf addr | xp /40bx 0x9013e |
| 查看线性地址内容 | x /nuf addr | x /40bx 0x13e |
| 反汇编一段内存 | u start end | u 0x30400 0x3040D |
| 反汇编执行的每一条指令 | trace-on | trace-on |
| 每执行一条指令就打印CPU信息 | trace-reg | trace-reg on |

其中"xp /40bx 0x9013e"这样的格式可能显得有点复杂，读者可以用"help x"这一指令在Bochs中亲自看一下它代表的意义

[执行控制]

c|cont 向下执行，相当于WinDBG的“g”。

s|step|stepi [count] 单步执行，相当于WinDBG的“t”，count 默认为 1。

p|n|next 单步执行，类似于WinDBG的“p”。

q|quit|exit 退出调试，同时关闭虚拟机。

Ctrl-C 结束执行状态，返回调试器提示符。

Ctrl-D if at empty line on command line, exit

（至少在Windows版本中我没有发现Ctrl-D有什么功能）

[执行断点]

vb|vbreak [seg:off] 在虚拟地址上下断点。

lb|lbreak [addr] 在线性地址上下断点，相当于WinDBG的“bp”。

pb|pbreak|b|break [addr] 在物理地址上下断点。（为了兼容GDB的语法，地址前

可以加上一个“\*”）。

blist 显示断点状态，相当于WinDBG的“bl”。

bpd|bpe [num] 禁用/启用断点，WinDBG的“be”和“bd”。num是断

点号，可以用blist命令查询。

d|del|delete [num] 删除断点，相当于WinDBG的“bc”。mum是断点号，可

以用blist命令查询。

[读写断点]

watch read [addr] 设置读断点。

watch write [addr] 设置写断点。

unwatch read [addr] 清除读断点。

unwatch write [addr] 清除写断点。

watch 显示当前所有读写断点。

unwatch 清除当前所有读写断点。

watch stop|continue 开关选项，设置遇到读写断点时中断下来还是显示出来但

是继续运行。

[内存操作]

x /nuf [addr] 显示线性地址的内容

xp /nuf [addr] 显示物理地址的内容

n 显示的单元数

u 每个显示单元的大小，u可以是下列之一：

b BYTE

h WORD

w DWORD

g DWORD64

注意: 这种命名法是按照GDB习惯的，而并不是按照inter的规范。

f 显示格式，f可以是下列之一：

x 按照十六进制显示

d 十进制显示

u 按照无符号十进制显示

o 按照八进制显示

t 按照二进制显示

c 按照字符显示

n、f、u是可选参数，如果不指定，则u默认是w，f默认是x。如果前面使用过x或

者xp命令，会按照上一次的x或者xp命令所使用的值。n默认为1。addr 也是一个

可选参数，如果不指定，addr是0，如过前面使用过x或者xp命令，指定了n=i，

则再次执行时n默认为i+1。

setpmem [addr] [size] [val] 设置物理内存某地址的内容。

需要注意的是，每次最多只能设置一个DWORD：

这样是可以的：

<bochs:1> setpmem 0x00000000 0x4 0x11223344

<bochs:2> x /4 0x00000000

[bochs]:

0x00000000 <bogus+ 0>: 0x11223344 0x00000000 0x00000000 0x00000000

这样也可以：

<bochs:1> setpmem 0x00000000 0x2 0x11223344

<bochs:2> x /4 0x00000000

[bochs]:

0x00000000 <bogus+ 0>: 0x00003344 0x00000000 0x00000000 0x00000000

或者：

<bochs:1> setpmem 0x00000000 0x1 0x20

<bochs:2> x /4 0x00000000

[bochs]:

0x00000000 <bogus+ 0>: 0x00000020 0x00000000 0x00000000 0x00000000

下面的做法都会导致出错：

<bochs:1> setpmem 0x00000000 0x3 0x112233

Error: setpmem: bad length value = 3

<bochs:2> setpmem 0x00000000 0x8 0x11223344

Error: setpmem: bad length value = 8

crc [start] [end] 显示物理地址start到end之间数据的CRC。

[寄存器操作]

set $reg = val 设置寄存器的值。现在版本可以设置的寄存器包括：

eax ecx edx ebx esp ebp esi edi

暂时不能设置：

eflags cs ss ds es fs gs

r|reg|registers reg = val 同上。

dump\_cpu 显示完整的CPU信息。

set\_cpu 设置CPU状态，这里可以设置dump\_cpu所能显示出来的

所有CPU状态。

[反汇编命令]

u|disas|disassemble [/num] [start] [end]

反汇编物理地址start到end 之间的代码，如

果不指定参数则反汇编当前EIP指向的代码。

num是可选参数，指定处理的代码量。

set $disassemble\_size = 0|16|32 $disassemble\_size变量指定反汇编使用的段

大小。

set $auto\_disassemble = 0|1 $auto\_disassemble决定每次执行中断下来的

时候（例如遇到断点、Ctrl-C等）是否反汇

编当前指令。

[其他命令]

trace-on|trace-off Tracing开关打开后，每执行一条指令都会将反汇编的结果

显示出来。

ptime 显示Bochs自本次运行以来执行的指令条数。

sb [val] 再执行val条指令就中断。val是64-bit整数，以L结尾，形

如“1000L”

sba [val] 执行到Bochs自本次运行以来的第val条指令就中断。val是

64-bit整数，以L结尾，形如“1000L”

modebp 设置切换到v86模式时中断。

record ["filename"] 将输入的调试指令记录到文件中。文件名必须包含引号。

playback ["filename"] 回放record的记录文件。文件名必须包含引号。

print-stack [num] 显示堆栈，num默认为16，表示打印的条数。

?|calc 和WinDBG的“?”命令类似，计算表达式的值。

load-symbols [global] filename [offset]

载入符号文件。如果设定了“global”关键字，则符号针

对所有上下文都有效。offset会默认加到所有的symbol地

址上。symbol文件的格式为："%x %s"。

[info命令]

info program 显示程序执行的情况。

info registers|reg|r 显示寄存器的信息。

info pb|pbreak|b|break 相当于blist

info dirty 显示脏页的页地址。

info cpu 显示所有CPU寄存器的值。

info fpu 显示所有FPU寄存器的值。

info idt 显示IDT。

info gdt [num] 显示GDT。

info ldt 显示LDT。

info tss 显示TSS。

info pic 显示PIC。

info ivt [num] [num] 显示IVT。

info flags 显示状态寄存器。

info cr 显示CR系列寄存器。

info symbols 显示symbol信息。

info ne2k|ne2000 显示虚拟的ne2k网卡信息。