lixinrui@lixinruis-MacBook-Pro ~/o/D/大/2/o/test> **make bochs (进入bochs)**

nasm lab1.asm -o lab1

dd if=/dev/zero of=floppy3.img bs=1024 count=1440

1440+0 records in

1440+0 records out

1474560 bytes (1.5 MB, 1.4 MiB) copied, 0.004223 s, 349 MB/s

dd if=lab1 of=floppy3.img conv=notrunc

1+0 records in

1+0 records out

512 bytes copied, 6.8e-05 s, 7.5 MB/s

export FLOPPYFILE=floppy3.img && bochs -f bochsrc

========================================================================

Bochs x86 Emulator 2.6.9

Built from SVN snapshot on April 9, 2017

Compiled on May 1 2017 at 22:12:04

========================================================================

00000000000i[ ] LTDL\_LIBRARY\_PATH not set. using compile time default '/usr/local/Cellar/bochs/2.6.9\_1/lib/bochs/plugins'

00000000000i[ ] BXSHARE not set. using compile time default '/usr/local/Cellar/bochs/2.6.9\_1/share/bochs'

00000000000i[ ] lt\_dlhandle is 0x7ff8f272b8d0

00000000000i[PLUGIN] loaded plugin libbx\_usb\_common.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f2401db0

00000000000i[PLUGIN] loaded plugin libbx\_unmapped.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f260a9a0

00000000000i[PLUGIN] loaded plugin libbx\_biosdev.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f2402220

00000000000i[PLUGIN] loaded plugin libbx\_speaker.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f260b170

00000000000i[PLUGIN] loaded plugin libbx\_extfpuirq.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f260b390

00000000000i[PLUGIN] loaded plugin libbx\_parallel.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f260c640

00000000000i[PLUGIN] loaded plugin libbx\_serial.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f272e7d0

00000000000i[PLUGIN] loaded plugin libbx\_iodebug.so

00000000000i[ ] reading configuration from bochsrc

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

Bochs Configuration: Main Menu

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

This is the Bochs Configuration Interface, where you can describe the

machine that you want to simulate. Bochs has already searched for a

configuration file (typically called bochsrc.txt) and loaded it if it

could be found. When you are satisfied with the configuration, go

ahead and start the simulation.

You can also start bochs with the -q option to skip these menus.

1. Restore factory default configuration

2. Read options from...

3. Edit options

4. Save options to...

5. Restore the Bochs state from...

6. Begin simulation

7. Quit now

Please choose one: **[6]（进入模拟）**

00000000000i[ ] lt\_dlhandle is 0x7ff8f272eaf0

00000000000i[PLUGIN] loaded plugin libbx\_sdl2.so

00000000000i[ ] installing sdl2 module as the Bochs GUI

00000000000i[SDL2 ] maximum host resolution: x=2880 y=1800

00000000000i[ ] Bochs x86 Emulator 2.6.9

00000000000i[ ] Built from SVN snapshot on April 9, 2017

00000000000i[ ] Compiled on May 1 2017 at 22:12:04

00000000000i[ ] System configuration

00000000000i[ ] processors: 1 (cores=1, HT threads=1)

00000000000i[ ] A20 line support: yes

00000000000i[ ] IPS is set to 4000000

00000000000i[ ] CPU configuration

00000000000i[ ] SMP support: yes, quantum=16

00000000000i[ ] level: 6

00000000000i[ ] APIC support: xapic

00000000000i[ ] FPU support: yes

00000000000i[ ] MMX support: yes

00000000000i[ ] 3dnow! support: no

00000000000i[ ] SEP support: yes

00000000000i[ ] SIMD support: sse2

00000000000i[ ] XSAVE support: no

00000000000i[ ] AES support: no

00000000000i[ ] SHA support: no

00000000000i[ ] MOVBE support: no

00000000000i[ ] ADX support: no

00000000000i[ ] x86-64 support: yes

00000000000i[ ] 1G paging support: no

00000000000i[ ] MWAIT support: yes

00000000000i[ ] VMX support: 1

00000000000i[ ] Optimization configuration

00000000000i[ ] RepeatSpeedups support: yes

00000000000i[ ] Fast function calls: yes

00000000000i[ ] Handlers Chaining speedups: no

00000000000i[ ] Devices configuration

00000000000i[ ] PCI support: i440FX i430FX

00000000000i[ ] Networking: no

00000000000i[ ] Sound support: no

00000000000i[ ] USB support: UHCI

00000000000i[ ] VGA extension support: vbe cirrus

00000000000i[MEM0 ] allocated memory at 0x1086b2000. after alignment, vector=0x1086b2000

00000000000i[MEM0 ] 32.00MB

00000000000i[MEM0 ] mem block size = 0x00020000, blocks=256

00000000000i[MEM0 ] rom at 0xfffe0000/131072 ('/usr/local/Cellar/bochs/2.6.9\_1/share/bochs/BIOS-bochs-latest')

00000000000i[ ] lt\_dlhandle is 0x7ff8f269fba0

00000000000i[PLUGIN] loaded plugin libbx\_hdimage.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f2427ea0

00000000000i[PLUGIN] loaded plugin libbx\_pci.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f4878780

00000000000i[PLUGIN] loaded plugin libbx\_pci2isa.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f26a0050

00000000000i[PLUGIN] loaded plugin libbx\_usb\_uhci.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f26a1960

00000000000i[PLUGIN] loaded plugin libbx\_acpi.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f4878ca0

00000000000i[PLUGIN] loaded plugin libbx\_cmos.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f26a1fc0

00000000000i[PLUGIN] loaded plugin libbx\_dma.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f2428640

00000000000i[PLUGIN] loaded plugin libbx\_pic.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f24286d0

00000000000i[PLUGIN] loaded plugin libbx\_pit.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f2428d30

00000000000i[PLUGIN] loaded plugin libbx\_vga.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f26a2540

00000000000i[PLUGIN] loaded plugin libbx\_floppy.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f26a2c50

00000000000i[PLUGIN] loaded plugin libbx\_ioapic.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f48790d0

00000000000i[PLUGIN] loaded plugin libbx\_keyboard.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f4879520

00000000000i[PLUGIN] loaded plugin libbx\_harddrv.so

00000000000i[ ] lt\_dlhandle is 0x7ff8f250bd90

00000000000i[PLUGIN] loaded plugin libbx\_pci\_ide.so

00000000000i[PLUGIN] init\_dev of 'pci' plugin device by virtual method

00000000000i[DEV ] i440FX PMC present at device 0, function 0

00000000000i[PLUGIN] init\_dev of 'pci2isa' plugin device by virtual method

00000000000i[DEV ] PIIX3 PCI-to-ISA bridge present at device 1, function 0

00000000000i[PLUGIN] init\_dev of 'cmos' plugin device by virtual method

00000000000i[CMOS ] Using local time for initial clock

00000000000i[CMOS ] Setting initial clock to: Fri Mar 16 16:04:47 2018 (time0=1521187487)

00000000000i[PLUGIN] init\_dev of 'dma' plugin device by virtual method

00000000000i[DMA ] channel 4 used by cascade

00000000000i[PLUGIN] init\_dev of 'pic' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'pit' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'vga' plugin device by virtual method

00000000000i[MEM0 ] Register memory access handlers: 0x0000000a0000 - 0x0000000bffff

00000000000i[VGA ] interval=200000, mode=realtime

00000000000i[MEM0 ] Register memory access handlers: 0x0000e0000000 - 0x0000e0ffffff

00000000000i[BXVGA ] VBE Bochs Display Extension Enabled

00000000000i[KEYMAP] Loading keymap from '/usr/local/Cellar/bochs/2.6.9\_1/share/bochs/keymaps/sdl2-pc-us.map'

00000000000i[KEYMAP] Loaded 147 symbols

00000000000i[MEM0 ] rom at 0xc0000/41472 ('/usr/local/Cellar/bochs/2.6.9\_1/share/bochs/VGABIOS-lgpl-latest')

00000000000i[PLUGIN] init\_dev of 'floppy' plugin device by virtual method

00000000000i[DMA ] channel 2 used by Floppy Drive

00000000000i[FLOPPY] fd0: 'floppy3.img' ro=0, h=2,t=80,spt=18

00000000000i[FLOPPY] Using boot sequence floppy, none, none

00000000000i[FLOPPY] Floppy boot signature check is enabled

00000000000i[PLUGIN] init\_dev of 'acpi' plugin device by virtual method

00000000000i[DEV ] ACPI Controller present at device 1, function 3

00000000000i[PLUGIN] init\_dev of 'ioapic' plugin device by virtual method

00000000000i[IOAPIC] initializing I/O APIC

00000000000i[MEM0 ] Register memory access handlers: 0x0000fec00000 - 0x0000fec00fff

00000000000i[IOAPIC] IOAPIC enabled (base address = 0xfec00000)

00000000000i[PLUGIN] init\_dev of 'keyboard' plugin device by virtual method

00000000000i[KBD ] will paste characters every 400 keyboard ticks

00000000000i[PLUGIN] init\_dev of 'harddrv' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'pci\_ide' plugin device by virtual method

00000000000i[DEV ] PIIX3 PCI IDE controller present at device 1, function 1

00000000000i[PLUGIN] init\_dev of 'unmapped' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'biosdev' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'speaker' plugin device by virtual method

00000000000i[PCSPK ] Forwarding beep to gui

00000000000i[PLUGIN] init\_dev of 'extfpuirq' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'parallel' plugin device by virtual method

00000000000i[PAR ] parallel port 1 at 0x0378 irq 7

00000000000i[PLUGIN] init\_dev of 'serial' plugin device by virtual method

00000000000i[SER ] com1 at 0x03f8 irq 4 (mode: null)

00000000000i[PLUGIN] init\_dev of 'iodebug' plugin device by virtual method

00000000000i[PLUGIN] init\_dev of 'usb\_uhci' plugin device by virtual method

00000000000i[DEV ] USB UHCI present at device 1, function 2

00000000000i[UHCI ] USB UHCI initialized

00000000000i[PLUGIN] register state of 'pci' plugin device by virtual method

00000000000i[PLUGIN] register state of 'pci2isa' plugin device by virtual method

00000000000i[PLUGIN] register state of 'cmos' plugin device by virtual method

00000000000i[PLUGIN] register state of 'dma' plugin device by virtual method

00000000000i[PLUGIN] register state of 'pic' plugin device by virtual method

00000000000i[PLUGIN] register state of 'pit' plugin device by virtual method

00000000000i[PLUGIN] register state of 'vga' plugin device by virtual method

00000000000i[PLUGIN] register state of 'floppy' plugin device by virtual method

00000000000i[PLUGIN] register state of 'unmapped' plugin device by virtual method

00000000000i[PLUGIN] register state of 'biosdev' plugin device by virtual method

00000000000i[PLUGIN] register state of 'speaker' plugin device by virtual method

00000000000i[PLUGIN] register state of 'extfpuirq' plugin device by virtual method

00000000000i[PLUGIN] register state of 'parallel' plugin device by virtual method

00000000000i[PLUGIN] register state of 'serial' plugin device by virtual method

00000000000i[PLUGIN] register state of 'iodebug' plugin device by virtual method

00000000000i[PLUGIN] register state of 'usb\_uhci' plugin device by virtual method

00000000000i[PLUGIN] register state of 'acpi' plugin device by virtual method

00000000000i[PLUGIN] register state of 'ioapic' plugin device by virtual method

00000000000i[PLUGIN] register state of 'keyboard' plugin device by virtual method

00000000000i[PLUGIN] register state of 'harddrv' plugin device by virtual method

00000000000i[PLUGIN] register state of 'pci\_ide' plugin device by virtual method

00000000000i[SYS ] bx\_pc\_system\_c::Reset(HARDWARE) called

00000000000i[CPU0 ] cpu hardware reset

00000000000i[APIC0 ] allocate APIC id=0 (MMIO enabled) to 0x0000fee00000

00000000000i[CPU0 ] CPU[0] is the bootstrap processor

00000000000i[CPU0 ] CPUID[0x00000000]: 00000005 756e6547 6c65746e 49656e69

00000000000i[CPU0 ] CPUID[0x00000001]: 00000633 00010800 00002028 1fcbfbff

00000000000i[CPU0 ] CPUID[0x00000002]: 00410601 00000000 00000000 00000000

00000000000i[CPU0 ] CPUID[0x00000003]: 00000000 00000000 00000000 00000000

00000000000i[CPU0 ] CPUID[0x00000004]: 00000000 00000000 00000000 00000000

00000000000i[CPU0 ] CPUID[0x00000005]: 00000040 00000040 00000003 00000020

00000000000i[CPU0 ] CPUID[0x80000000]: 80000008 00000000 00000000 00000000

00000000000i[CPU0 ] CPUID[0x80000001]: 00000000 00000000 00000101 2a100000

00000000000i[CPU0 ] CPUID[0x80000002]: 20202020 20202020 20202020 6e492020

00000000000i[CPU0 ] CPUID[0x80000003]: 286c6574 50202952 69746e65 52286d75

00000000000i[CPU0 ] CPUID[0x80000004]: 20342029 20555043 20202020 00202020

00000000000i[CPU0 ] CPUID[0x80000005]: 01ff01ff 01ff01ff 40020140 40020140

00000000000i[CPU0 ] CPUID[0x80000006]: 00000000 42004200 02008140 00000000

00000000000i[CPU0 ] CPUID[0x80000007]: 00000000 00000000 00000000 00000000

00000000000i[CPU0 ] CPUID[0x80000008]: 00003028 00000000 00000000 00000000

00000000000i[PLUGIN] reset of 'pci' plugin device by virtual method

00000000000i[PLUGIN] reset of 'pci2isa' plugin device by virtual method

00000000000i[PLUGIN] reset of 'cmos' plugin device by virtual method

00000000000i[PLUGIN] reset of 'dma' plugin device by virtual method

00000000000i[PLUGIN] reset of 'pic' plugin device by virtual method

00000000000i[PLUGIN] reset of 'pit' plugin device by virtual method

00000000000i[PLUGIN] reset of 'vga' plugin device by virtual method

00000000000i[PLUGIN] reset of 'floppy' plugin device by virtual method

00000000000i[PLUGIN] reset of 'acpi' plugin device by virtual method

00000000000i[PLUGIN] reset of 'ioapic' plugin device by virtual method

00000000000i[PLUGIN] reset of 'keyboard' plugin device by virtual method

00000000000i[PLUGIN] reset of 'harddrv' plugin device by virtual method

00000000000i[PLUGIN] reset of 'pci\_ide' plugin device by virtual method

00000000000i[PLUGIN] reset of 'unmapped' plugin device by virtual method

00000000000i[PLUGIN] reset of 'biosdev' plugin device by virtual method

00000000000i[PLUGIN] reset of 'speaker' plugin device by virtual method

00000000000i[PLUGIN] reset of 'extfpuirq' plugin device by virtual method

00000000000i[PLUGIN] reset of 'parallel' plugin device by virtual method

00000000000i[PLUGIN] reset of 'serial' plugin device by virtual method

00000000000i[PLUGIN] reset of 'iodebug' plugin device by virtual method

00000000000i[PLUGIN] reset of 'usb\_uhci' plugin device by virtual method

00000000000i[ ] set SIGINT handler to bx\_debug\_ctrlc\_handler

Next at t=0

(0) [0x0000fffffff0] f000:fff0 (unk. ctxt): jmpf 0xf000:e05b ; ea5be000f0

<bochs:1> **b 0x7c00（在0x7c00处设置断点）**

<bochs:2> **c（继续运行）**

00000004662i[BIOS ] $Revision: 13073 $ $Date: 2017-02-16 22:43:52 +0100 (Do, 16. Feb 2017) $

00000318050i[KBD ] reset-disable command received

00000320819i[BIOS ] Starting rombios32

00000321257i[BIOS ] Shutdown flag 0

00000321840i[BIOS ] ram\_size=0x02000000

00000322261i[BIOS ] ram\_end=32MB

00000362829i[BIOS ] Found 1 cpu(s)

00000376413i[BIOS ] bios\_table\_addr: 0x000f9cd8 end=0x000fcc00

00000704208i[PCI ] i440FX PMC write to PAM register 59 (TLB Flush)

00001032137i[P2ISA ] PCI IRQ routing: PIRQA# set to 0x0b

00001032156i[P2ISA ] PCI IRQ routing: PIRQB# set to 0x09

00001032175i[P2ISA ] PCI IRQ routing: PIRQC# set to 0x0b

00001032194i[P2ISA ] PCI IRQ routing: PIRQD# set to 0x09

00001032204i[P2ISA ] write: ELCR2 = 0x0a

00001032974i[BIOS ] PIIX3/PIIX4 init: elcr=00 0a

00001040697i[BIOS ] PCI: bus=0 devfn=0x00: vendor\_id=0x8086 device\_id=0x1237 class=0x0600

00001042976i[BIOS ] PCI: bus=0 devfn=0x08: vendor\_id=0x8086 device\_id=0x7000 class=0x0601

00001045094i[BIOS ] PCI: bus=0 devfn=0x09: vendor\_id=0x8086 device\_id=0x7010 class=0x0101

00001045323i[PIDE ] new BM-DMA address: 0xc000

00001045939i[BIOS ] region 4: 0x0000c000

00001047953i[BIOS ] PCI: bus=0 devfn=0x0a: vendor\_id=0x8086 device\_id=0x7020 class=0x0c03

00001048157i[UHCI ] new base address: 0xc020

00001048773i[BIOS ] region 4: 0x0000c020

00001048901i[UHCI ] new irq line = 9

00001050796i[BIOS ] PCI: bus=0 devfn=0x0b: vendor\_id=0x8086 device\_id=0x7113 class=0x0680

00001051028i[ACPI ] new irq line = 11

00001051040i[ACPI ] new irq line = 9

00001051065i[ACPI ] new PM base address: 0xb000

00001051079i[ACPI ] new SM base address: 0xb100

00001051107i[PCI ] setting SMRAM control register to 0x4a

00001215200i[CPU0 ] Enter to System Management Mode

00001215200i[CPU0 ] enter\_system\_management\_mode: temporary disable VMX while in SMM mode

00001215210i[CPU0 ] RSM: Resuming from System Management Mode

00001379231i[PCI ] setting SMRAM control register to 0x0a

00001394138i[BIOS ] MP table addr=0x000f9db0 MPC table addr=0x000f9ce0 size=0xc8

00001395960i[BIOS ] SMBIOS table addr=0x000f9dc0

00001398141i[BIOS ] ACPI tables: RSDP addr=0x000f9ee0 ACPI DATA addr=0x01ff0000 size=0xf72

00001401353i[BIOS ] Firmware waking vector 0x1ff00cc

00001403148i[PCI ] i440FX PMC write to PAM register 59 (TLB Flush)

00001403871i[BIOS ] bios\_table\_cur\_addr: 0x000f9f04

00001531488i[VBIOS ] VGABios $Id: vgabios.c,v 1.76 2013/02/10 08:07:03 vruppert Exp $

00001531559i[BXVGA ] VBE known Display Interface b0c0

00001531591i[BXVGA ] VBE known Display Interface b0c5

00001534516i[VBIOS ] VBE Bios $Id: vbe.c,v 1.65 2014/07/08 18:02:25 vruppert Exp $

00014040189i[BIOS ] Booting from 0000:7c00

(0) Breakpoint 1, 0x0000000000007c00 in ?? ()

Next at t=14040244

(0) [0x000000007c00] 0000:7c00 (unk. ctxt): jmp .+39 (0x00007c29) ; eb27

<bochs:3> **u/100（反编译，显示100行代码）**

00007c00: ( ): jmp .+39 ; eb27

00007c02: ( ): jo .+23 ; 7017

00007c04: ( ): call .+1795 ; e80307

00007c07: ( ): add byte ptr ds:[bx+si], al ; 0000

00007c09: ( ): add byte ptr ds:[bx+di], bl ; 0019

00007c0b: ( ): add bh, bh ; 00ff

00007c0d: ( ): call word ptr ds:[bx+si] ; ff5000

00007c10: ( ): (invalid) ; ffff

00007c12: ( ): add word ptr ds:[bx+si], ax ; 0100

00007c14: ( ): add word ptr ds:[bx+si], ax ; 0100

00007c16: ( ): inc cx ; 41

00007c17: ( ): add word ptr ds:[si+105], cx ; 014c69

00007c1a: ( ): js .+105 ; 7869

00007c1c: ( ): outsb dx, byte ptr ds:[si] ; 6e

00007c1d: ( ): jb .+117 ; 7275

00007c1f: ( ): imul sp, word ptr ds:[bx+si], 0x3531 ; 69203135

00007c23: ( ): xor si, word ptr ss:[bp+si] ; 3332

00007c25: ( ): xor si, word ptr ds:[bx+si] ; 3330

00007c27: ( ): xor si, word ptr ss:[bp+si] ; 3332

00007c29: ( ): call .+32 ; e82000

00007c2c: ( ): mov ax, cs ; 8cc8

00007c2e: ( ): mov es, ax ; 8ec0

00007c30: ( ): mov ds, ax ; 8ed8

00007c32: ( ): mov es, ax ; 8ec0

00007c34: ( ): mov ax, 0xb800 ; b800b8

00007c37: ( ): mov gs, ax ; 8ee8

00007c39: ( ): mov byte ptr ds:0x7c16, 0x41 ; c606167c41

00007c3e: ( ): call .+30 ; e81e00

00007c41: ( ): call .+41 ; e82900

00007c44: ( ): call .+101 ; e86500

00007c47: ( ): call .+133 ; e88500

00007c4a: ( ): jmp .-14 ; ebf2

00007c4c: ( ): pusha ; 60

00007c4d: ( ): mov ah, 0x06 ; b406

00007c4f: ( ): mov al, 0x00 ; b000

00007c51: ( ): mov bh, 0x07 ; b707

00007c53: ( ): mov ch, 0x00 ; b500

00007c55: ( ): mov cl, 0x00 ; b100

00007c57: ( ): mov dh, 0x18 ; b618

00007c59: ( ): mov dl, 0x4f ; b24f

00007c5b: ( ): int 0x10 ; cd10

00007c5d: ( ): popa ; 61

00007c5e: ( ): ret ; c3

00007c5f: ( ): pusha ; 60

00007c60: ( ): mov cx, 0x03e8 ; b9e803

00007c63: ( ): mov bx, 0x1770 ; bb7017

00007c66: ( ): dec bx ; 4b

00007c67: ( ): jnle .-3 ; 7ffd

00007c69: ( ): loop .-8 ; e2f8

00007c6b: ( ): popa ; 61

00007c6c: ( ): ret ; c3

00007c6d: ( ): pusha ; 60

00007c6e: ( ): xor ax, ax ; 31c0

00007c70: ( ): mov ax, word ptr ds:0x7c06 ; a1067c

00007c73: ( ): mov bx, 0x0050 ; bb5000

00007c76: ( ): mul ax, bx ; f7e3

00007c78: ( ): add ax, word ptr ds:0x7c08 ; 0306087c

00007c7c: ( ): mov bx, 0x0002 ; bb0200

00007c7f: ( ): mul ax, bx ; f7e3

00007c81: ( ): mov bp, ax ; 89c5

00007c83: ( ): mov ah, byte ptr ds:0x7c17 ; 8a26177c

00007c87: ( ): mov al, byte ptr ds:0x7c16 ; a0167c

00007c8a: ( ): call .-65 ; e8bfff

00007c8d: ( ): mov word ptr gs:[bp], ax ; 65894600

00007c91: ( ): call .+2 ; e80200

00007c94: ( ): popa ; 61

00007c95: ( ): ret ; c3

00007c96: ( ): pusha ; 60

00007c97: ( ): mov ax, 0x7c18 ; b8187c

00007c9a: ( ): mov bp, ax ; 89c5

00007c9c: ( ): mov cx, 0x0011 ; b91100

00007c9f: ( ): mov ax, 0x1300 ; b80013

00007ca2: ( ): mov bx, 0x00f1 ; bbf100

00007ca5: ( ): mov dx, 0x0920 ; ba2009

00007ca8: ( ): int 0x10 ; cd10

00007caa: ( ): popa ; 61

00007cab: ( ): ret ; c3

00007cac: ( ): pusha ; 60

00007cad: ( ): mov ax, word ptr ds:0x7c12 ; a1127c

00007cb0: ( ): mov bx, word ptr ds:0x7c14 ; 8b1e147c

00007cb4: ( ): add word ptr ds:0x7c06, ax ; 0106067c

00007cb8: ( ): add word ptr ds:0x7c08, bx ; 011e087c

00007cbc: ( ): popa ; 61

00007cbd: ( ): ret ; c3

00007cbe: ( ): cmp byte ptr ds:0x7c17, 0x0f ; 803e177c0f

00007cc3: ( ): jnz .+5 ; 7505

00007cc5: ( ): mov byte ptr ds:0x7c17, 0x00 ; c606177c00

00007cca: ( ): inc byte ptr ds:0x7c17 ; fe06177c

00007cce: ( ): ret ; c3

00007ccf: ( ): pusha ; 60

00007cd0: ( ): mov ax, word ptr ds:0x7c06 ; a1067c

00007cd3: ( ): cmp ax, word ptr ds:0x7c0c ; 3b060c7c

00007cd7: ( ): jz .+8 ; 7408

00007cd9: ( ): cmp ax, word ptr ds:0x7c0a ; 3b060a7c

00007cdd: ( ): jz .+2 ; 7402

00007cdf: ( ): jmp .+7 ; eb07

00007ce1: ( ): call .-38 ; e8daff

00007ce4: ( ): neg word ptr ds:0x7c12 ; f71e127c

00007ce8: ( ): mov bx, word ptr ds:0x7c08 ; 8b1e087c

00007cec: ( ): cmp bx, word ptr ds:0x7c10 ; 3b1e107c

<bochs:4> **b 0x00007c3e（在0x00007c3e处设置断点）**

<bochs:5> **c（继续运行）**

(0) Breakpoint 2, 0x0000000000007c3e in ?? ()

Next at t=14042615

(0) [0x000000007c3e] 0000:7c3e (unk. ctxt): call .+30 (0x00007c5f) ; e81e00

<bochs:6> **u/10（显示10行代码）**

00007c3e: ( ): call .+30 ; e81e00

00007c41: ( ): call .+41 ; e82900

00007c44: ( ): call .+101 ; e86500

00007c47: ( ): call .+133 ; e88500

00007c4a: ( ): jmp .-14 ; ebf2

00007c4c: ( ): pusha ; 60

00007c4d: ( ): mov ah, 0x06 ; b406

00007c4f: ( ): mov al, 0x00 ; b000

00007c51: ( ): mov bh, 0x07 ; b707

00007c53: ( ): mov ch, 0x00 ; b500

<bochs:7> **b 0x00007c41（设置断点）**

<bochs:8> **c（继续运行）**

(0) Breakpoint 3, 0x0000000000007c41 in ?? ()

Next at t=26044620

(0) [0x000000007c41] 0000:7c41 (unk. ctxt): call .+41 (0x00007c6d) ; e82900

<bochs:9> s

Next at t=26044621

(0) [0x000000007c6d] 0000:7c6d (unk. ctxt): pusha ; 60

<bochs:10> **u/10（显示10行代码）**

00007c6d: ( ): pusha ; 60

00007c6e: ( ): xor ax, ax ; 31c0

00007c70: ( ): mov ax, word ptr ds:0x7c06 ; a1067c

00007c73: ( ): mov bx, 0x0050 ; bb5000

00007c76: ( ): mul ax, bx ; f7e3

00007c78: ( ): add ax, word ptr ds:0x7c08 ; 0306087c

00007c7c: ( ): mov bx, 0x0002 ; bb0200

00007c7f: ( ): mul ax, bx ; f7e3

00007c81: ( ): mov bp, ax ; 89c5

00007c83: ( ): mov ah, byte ptr ds:0x7c17 ; 8a26177c

<bochs:11> **s（单步执行一条指令）**

Next at t=26044622

(0) [0x000000007c6e] 0000:7c6e (unk. ctxt): xor ax, ax ; 31c0

<bochs:12>

Next at t=26044623

(0) [0x000000007c70] 0000:7c70 (unk. ctxt): mov ax, word ptr ds:0x7c06 ; a1067c

<bochs:13> **r（查看寄存器的值）**

CPU0:

rax: 00000000\_00000000 rcx: 00000000\_00090000

rdx: 00000000\_00000000 rbx: 00000000\_00000000

rsp: 00000000\_0000ffc4 rbp: 00000000\_00000000

rsi: 00000000\_000e0000 rdi: 00000000\_0000ffac

r8 : 00000000\_00000000 r9 : 00000000\_00000000

r10: 00000000\_00000000 r11: 00000000\_00000000

r12: 00000000\_00000000 r13: 00000000\_00000000

r14: 00000000\_00000000 r15: 00000000\_00000000

rip: 00000000\_00007c70

eflags 0x00000046: id vip vif ac vm rf nt IOPL=0 of df if tf sf ZF af PF cf

<bochs:14> **s（单步执行一条指令）**

Next at t=26044624

(0) [0x000000007c73] 0000:7c73 (unk. ctxt): mov bx, 0x0050 ; bb5000

<bochs:15> **r（查看寄存器的值）**

CPU0:

rax: 00000000\_00000007 rcx: 00000000\_00090000

rdx: 00000000\_00000000 rbx: 00000000\_00000000

rsp: 00000000\_0000ffc4 rbp: 00000000\_00000000

rsi: 00000000\_000e0000 rdi: 00000000\_0000ffac

r8 : 00000000\_00000000 r9 : 00000000\_00000000

r10: 00000000\_00000000 r11: 00000000\_00000000

r12: 00000000\_00000000 r13: 00000000\_00000000

r14: 00000000\_00000000 r15: 00000000\_00000000

rip: 00000000\_00007c73

eflags 0x00000046: id vip vif ac vm rf nt IOPL=0 of df if tf sf ZF af PF cf

<bochs:16> **x 0x7c06（查看内存0x7c06处的值）**

[bochs]:

0x0000000000007c06 <bogus+ 0>: 0x00000007

<bochs:17> x 0x00026044624i[ ] Ctrl-C detected in signal handler.

[bochs]:

0x0000000000007c06 <bogus+ 0>: 0x00000007

<bochs:18> **x/h 0x7c06（查看内存0x7c06处的值，只显示2个字节）**

[bochs]:

0x0000000000007c06 <bogus+ 0>: 0x0007

<bochs:19> **c（继续运行）**

(0) Breakpoint 2, 0x0000000000007c3e in ?? ()

Next at t=26059515

(0) [0x000000007c3e] 0000:7c3e (unk. ctxt): call .+30 (0x00007c5f) ; e81e00

<bochs:20>

(0) Breakpoint 3, 0x0000000000007c41 in ?? ()

Next at t=38061520

(0) [0x000000007c41] 0000:7c41 (unk. ctxt): call .+41 (0x00007c6d) ; e82900

<bochs:21>

(0) Breakpoint 2, 0x0000000000007c3e in ?? ()

Next at t=38076415

(0) [0x000000007c3e] 0000:7c3e (unk. ctxt): call .+30 (0x00007c5f) ; e81e00

<bochs:22> ^Dfgets() returned ERROR.

debugger interrupt request was 0

00038076415i[CPU0 ] CPU is in real mode (active)

00038076415i[CPU0 ] CS.mode = 16 bit

00038076415i[CPU0 ] SS.mode = 16 bit

00038076415i[CPU0 ] EFER = 0x00000000

00038076415i[CPU0 ] | EAX=0000b800 EBX=00000000 ECX=00090000 EDX=00000000

00038076415i[CPU0 ] | ESP=0000ffd6 EBP=00000000 ESI=000e0000 EDI=0000ffac

00038076415i[CPU0 ] | IOPL=0 id vip vif ac vm rf nt of df if tf SF zf af PF CF

00038076415i[CPU0 ] | SEG sltr(index|ti|rpl) base limit G D

00038076415i[CPU0 ] | CS:0000( 0004| 0| 0) 00000000 0000ffff 0 0

00038076415i[CPU0 ] | DS:0000( 0005| 0| 0) 00000000 0000ffff 0 0

00038076415i[CPU0 ] | SS:0000( 0005| 0| 0) 00000000 0000ffff 0 0

00038076415i[CPU0 ] | ES:0000( 0005| 0| 0) 00000000 0000ffff 0 0

00038076415i[CPU0 ] | FS:0000( 0005| 0| 0) 00000000 0000ffff 0 0

00038076415i[CPU0 ] | GS:b800( 0005| 0| 0) 000b8000 0000ffff 0 0

00038076415i[CPU0 ] | EIP=00007c3e (00007c3e)

00038076415i[CPU0 ] | CR0=0x60000010 CR2=0x00000000

00038076415i[CPU0 ] | CR3=0x00000000 CR4=0x00000000

(0).[38076415] [0x000000007c3e] 0000:7c3e (unk. ctxt): call .+30 (0x00007c5f) ; e81e00

00038076415i[CMOS ] Last time is 1521187496 (Fri Mar 16 16:04:56 2018)

00038076415i[SIM ] quit\_sim called with exit code 1

make: \*\*\* [bochs] Error 1