

Nama : Aprinta Sewelastami
NIM : L200180088

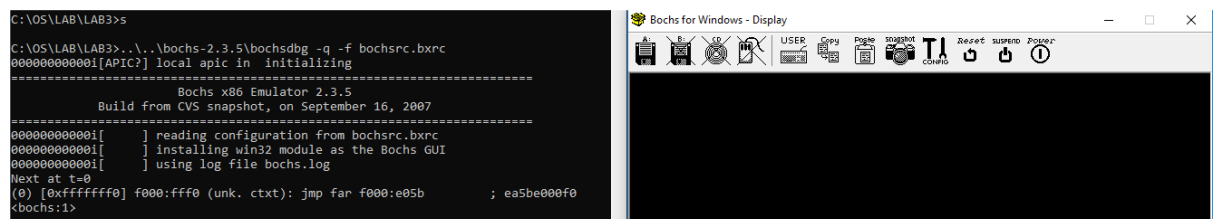
1. Masuk ke C, direktori OS, ketikkan setpath dan masuk ke direktori LAB\LAB3

```
C:\OS>setpath  
  
C:\OS>Path=C:\OS\Dev-Cpp\bin;C:\OS\Bochs-2.3.5;c:\OS\Perl;C:\Windows;C:\Windows\System32  
C:\OS>cd LAB/LAB3  
  
C:\OS\LAB\LAB3>
```

2. Ketik "type s.bat"

```
C:\OS\LAB\LAB3>type s.bat  
..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc  
  
C:\OS\LAB\LAB3>
```

3. Lakukan debugging dengan cara ketik 's'



```
C:\OS\LAB\LAB3>s  
C:\OS\LAB\LAB3>..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc  
00000000000i[APIC?] local apic in initializing  
===== Bochs x86 Emulator 2.3.5  
Build from CVS snapshot, on September 16, 2007  
===== 00000000000i[ ] reading configuration from bochsrc.bxrc  
00000000000i[ ] installing win32 module as the Bochs GUI  
00000000000i[ ] using log file bochs.log  
Next at t=0  
(0) [0xfffffff0] f000:ffff (unk. ctxt): jmp far f000:e05b ; ea5be000f0  
<bochs:1>
```

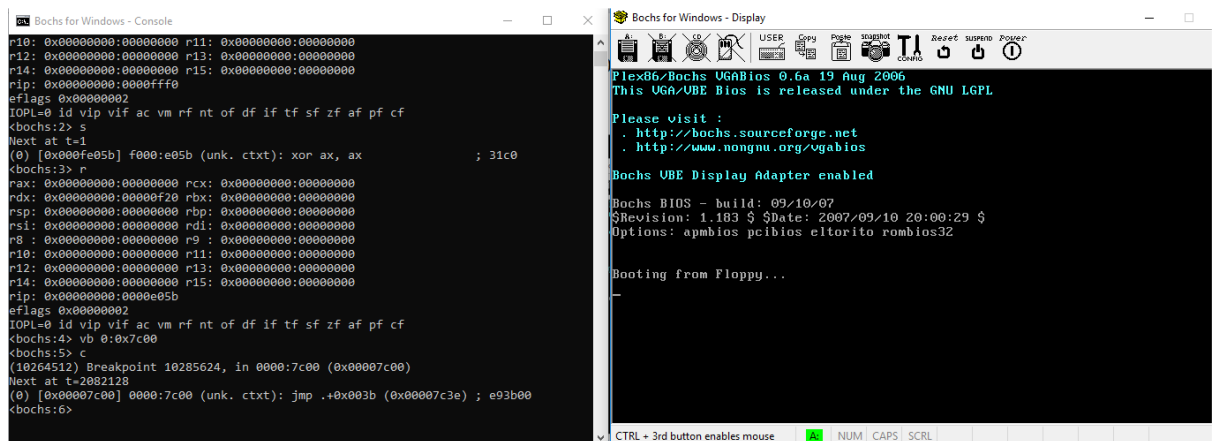
4. Ketikkan perintah 'r' untuk melihat isi register CS dan IP

```
<bochs:1> r  
rax: 0x00000000:00000000 rcx: 0x00000000:00000000  
rdx: 0x00000000:0000f20 rbx: 0x00000000:00000000  
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000  
rsi: 0x00000000:00000000 rdi: 0x00000000:00000000  
r8 : 0x00000000:00000000 r9 : 0x00000000:00000000  
r10: 0x00000000:00000000 r11: 0x00000000:00000000  
r12: 0x00000000:00000000 r13: 0x00000000:00000000  
r14: 0x00000000:00000000 r15: 0x00000000:00000000  
rip: 0x00000000:0000ffff  
eflags 0x00000002  
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf  
<bochs:2>
```

5. Kemudian ketik 's'<ENTER> lalu ketik 'r'<ENTER>

```
<bochs:2> s
Next at t=1
(0) [0x000fe05b] f000:e05b (unk. ctxt): xor ax, ax          ; 31c0
<bochs:3> r
rax: 0x00000000:00000000 rcx: 0x00000000:00000000
rdx: 0x00000000:00000f20 rbx: 0x00000000:00000000
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000
rsi: 0x00000000:00000000 rdi: 0x00000000:00000000
r8 : 0x00000000:00000000 r9 : 0x00000000:00000000
r10: 0x00000000:00000000 r11: 0x00000000:00000000
r12: 0x00000000:00000000 r13: 0x00000000:00000000
r14: 0x00000000:00000000 r15: 0x00000000:00000000
rip: 0x00000000:0000e05b
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<bochs:4>
```

6. Masukkan perintah 'vb 0:0x7C00'<ENTER>



The screenshot shows two windows from the Bochs emulator. The 'Bochs for Windows - Console' window on the left displays the execution of the 'vb 0:0x7C00' command, which sets a breakpoint at address 0x0007c00. The 'Bochs for Windows - Display' window on the right shows the BIOS boot screen, including the version (0.6a), release date (19 Aug 2006), and the message 'Booting from Floppy...'. The console window also shows the state of registers and flags before the command.

```
Bochs for Windows - Console
r10: 0x00000000:00000000 r11: 0x00000000:00000000
r12: 0x00000000:00000000 r13: 0x00000000:00000000
r14: 0x00000000:00000000 r15: 0x00000000:00000000
rip: 0x00000000:0000ff00
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
Next at t=1
(0) [0x000fe05b] f000:e05b (unk. ctxt): xor ax, ax          ; 31c0
<bochs:2> s
Next at t=1
(0) [0x000fe05b] f000:e05b (unk. ctxt): xor ax, ax          ; 31c0
<bochs:3> r
rax: 0x00000000:00000000 rcx: 0x00000000:00000000
rdx: 0x00000000:00000f20 rbx: 0x00000000:00000000
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000
rsi: 0x00000000:00000000 rdi: 0x00000000:00000000
r8 : 0x00000000:00000000 r9 : 0x00000000:00000000
r10: 0x00000000:00000000 r11: 0x00000000:00000000
r12: 0x00000000:00000000 r13: 0x00000000:00000000
r14: 0x00000000:00000000 r15: 0x00000000:00000000
rip: 0x00000000:0000e05b
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<bochs:4> vb 0:0x7c00
<bochs:5> c
(10264512) Breakpoint 10285624, in 0000:7c00 (0x0007c00)
Next at t=2082128
(0) [0x00007c00] 0000:7c00 (unk. ctxt): jmp .+0x003b (0x0007c3e) ; e93b00
<bochs:6>
```

Bochs for Windows - Display

Plex86/Bochs UGA BIOS 0.6a 19 Aug 2006
This UGA/UE BIOS is released under the GNU LGPL

Please visit :
- <http://bochs.sourceforge.net>
- <http://www.nongnu.org/ugabios>

Bochs UBE Display Adapter enabled

Bochs BIOS - build: 09/10/07
\$Revision: 1.183 \$ \$Date: 2007/09/10 20:00:29 \$
Options: apmbios pcibios eltorito rombios32

Booting from Floppy...

CTRL + 3rd button enables mouse NUM CAPS SCRL

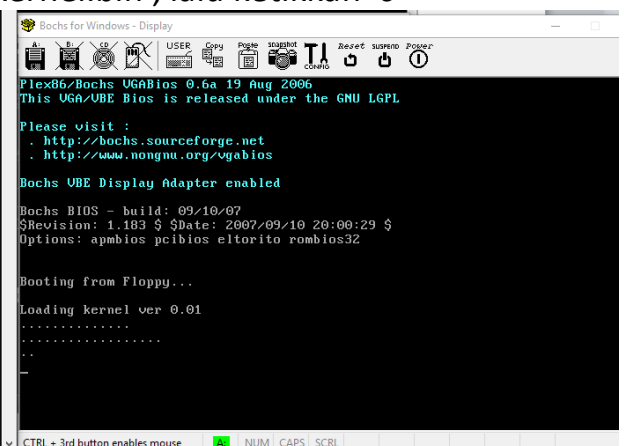
7. Ketik perintah 'c' untuk continue, lalu ketik 's' berulang sebanyak 10x

```
<bochs:5> c
(10264512) Breakpoint 10285624, in 0000:7c00 (0x00007c00)
Next at t=2082128
(0) [0x00007c00] 0000:7c00 (unk. ctxt): jmp .+0x003b (0x00007c3e) ; e93b00
<bochs:6> s
Next at t=2082129
(0) [0x00007c3e] 0000:7c3e (unk. ctxt): cli ; fa
<bochs:7> s
Next at t=2082130
(0) [0x00007c3f] 0000:7c3f (unk. ctxt): mov ax, 0x07c0 ; b8c007
<bochs:8> s
Next at t=2082131
(0) [0x00007c42] 0000:7c42 (unk. ctxt): mov ds, ax ; 8ed8
<bochs:9> s
Next at t=2082132
(0) [0x00007c44] 0000:7c44 (unk. ctxt): mov es, ax ; 8ec0
<bochs:10> s
Next at t=2082133
(0) [0x00007c46] 0000:7c46 (unk. ctxt): mov fs, ax ; 8ee0
<bochs:11> s
Next at t=2082134
(0) [0x00007c48] 0000:7c48 (unk. ctxt): mov gs, ax ; 8ee8
<bochs:12> s
Next at t=2082135
(0) [0x00007c4a] 0000:7c4a (unk. ctxt): mov ax, 0x0000 ; b80000
<bochs:13> s
Next at t=2082136
(0) [0x00007c4d] 0000:7c4d (unk. ctxt): mov ss, ax ; 8ed0
<bochs:14> s
Next at t=2082137
(0) [0x00007c4f] 0000:7c4f (unk. ctxt): mov sp, 0xffff ; bcf0ff
<bochs:15> s
Next at t=2082138
(0) [0x00007c52] 0000:7c52 (unk. ctxt): sti ; fb
<bochs:16>
```

8. Ketikkan 'q' untuk menghentikan debugging. Kemudian lakukan debugging lagi dengan cara ketikkan 's', kemudian ketikkan 'vb 0x0100:0x0000' untuk menghentikan langkah saat PC mulai mengeksekusi instruksi dari program 'kernel.bin', lalu ketikkan 'c'

```
<bochs:3> q
# In bx_win32_gui.c:exit(void)!
Bochs is exiting. Press ENTER when you're ready to close this window.

C:\OS\LAB\LAB3>s
C:\OS\LAB\LAB3>..\bochs-2.3.5\bochsrc -q -f bochsrc.bxrc
000000000001[APIC?] local apic in initializing
=====
Bochs x86 Emulator 2.3.5
Build from CVS snapshot, on September 16, 2007
=====
000000000001[ ] reading configuration from bochsrc.bxrc
000000000001[ ] installing win32 module as the Bochs GUI
000000000001[ ] using log file bochs.log
Next at t=0
(0) [0xfffffff0] f000:ffff (unk. ctxt): jmp far f000:e05b ; ea5be00f0
<bochs:1> vb 0x0100:0x0000
<bochs:2> c
(10264512) Breakpoint 10285624, in 0100:0000 (0x00001000)
Next at t=2945013
(0) [0x00001000] 0100:0000 (unk. ctxt): mov ax, 0x0100 ; b80001
<bochs:3>
```



9. Kemudian ketik 's' sebanyak 10x, lalu bandingkan hasilnya dengan isi file kernel.asm

```
<bochs:3> s
Next at t=2945014
(0) [0x00001003] 0100:0003 (unk. ctxt): mov ds, ax ; 8ed8
<bochs:4> s
Next at t=2945015
(0) [0x00001005] 0100:0005 (unk. ctxt): mov es, ax ; 8ec0
<bochs:5> s
Next at t=2945016
(0) [0x00001007] 0100:0007 (unk. ctxt): cli ; fa
<bochs:6> s
Next at t=2945017
(0) [0x00001008] 0100:0008 (unk. ctxt): mov ss, ax ; 8ed0
<bochs:7> s
Next at t=2945018
(0) [0x0000100a] 0100:000a (unk. ctxt): mov sp, 0xffff ; bcffff
<bochs:8> s
Next at t=2945019
(0) [0x0000100d] 0100:000d (unk. ctxt): sti ; fb
<bochs:9> s
Next at t=2945020
(0) [0x0000100e] 0100:000e (unk. ctxt): push dx ; 52
<bochs:10> s
Next at t=2945021
(0) [0x0000100f] 0100:000f (unk. ctxt): push es ; 06
<bochs:11> s
Next at t=2945022
(0) [0x00001010] 0100:0010 (unk. ctxt): xor ax, ax ; 31c0
<bochs:12> s
Next at t=2945023
(0) [0x00001012] 0100:0012 (unk. ctxt): mov es, ax ; 8ec0
<bochs:13>
```