

**LAPORAN PRAKTIKUM SISTEM OPERASI
MODUL 3
“MENGENAL CARA DEBUGGING
PROGRAM BOOTSTRAP-LOADER”**

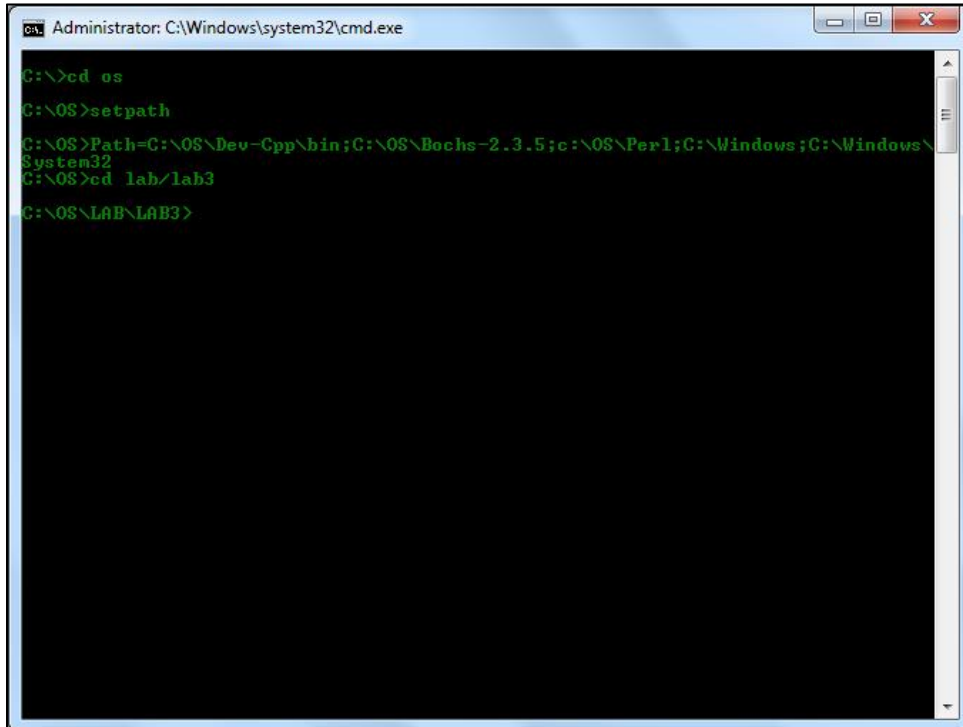


**Oleh:
Daffa Putra Alwansyah
L200190031
Informatika**

**Fakultas Komunikasi dan Informatika Universitas
Muhammadiyah Surakarta**

Langkah - Langkah :

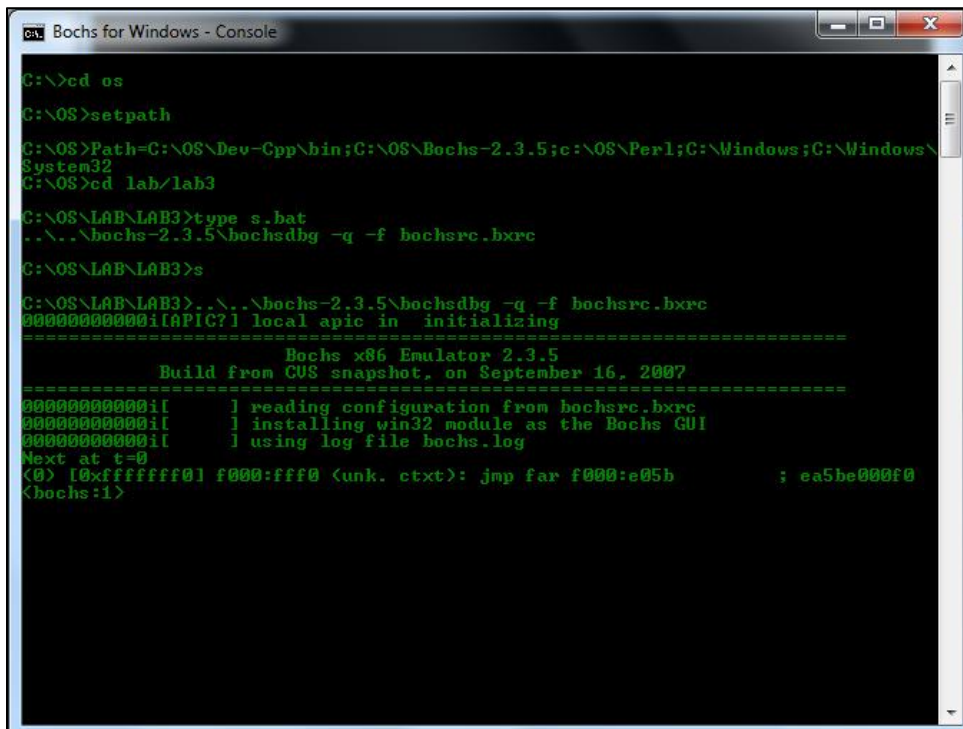
1. Membuka CMD lalu ketika cd OS, "Setpath" (jika sudah disetpath maka tidak usah).
Ketikan cd lab/lab3



```
Administrator: C:\Windows\system32\cmd.exe

C:\>cd os
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>
```

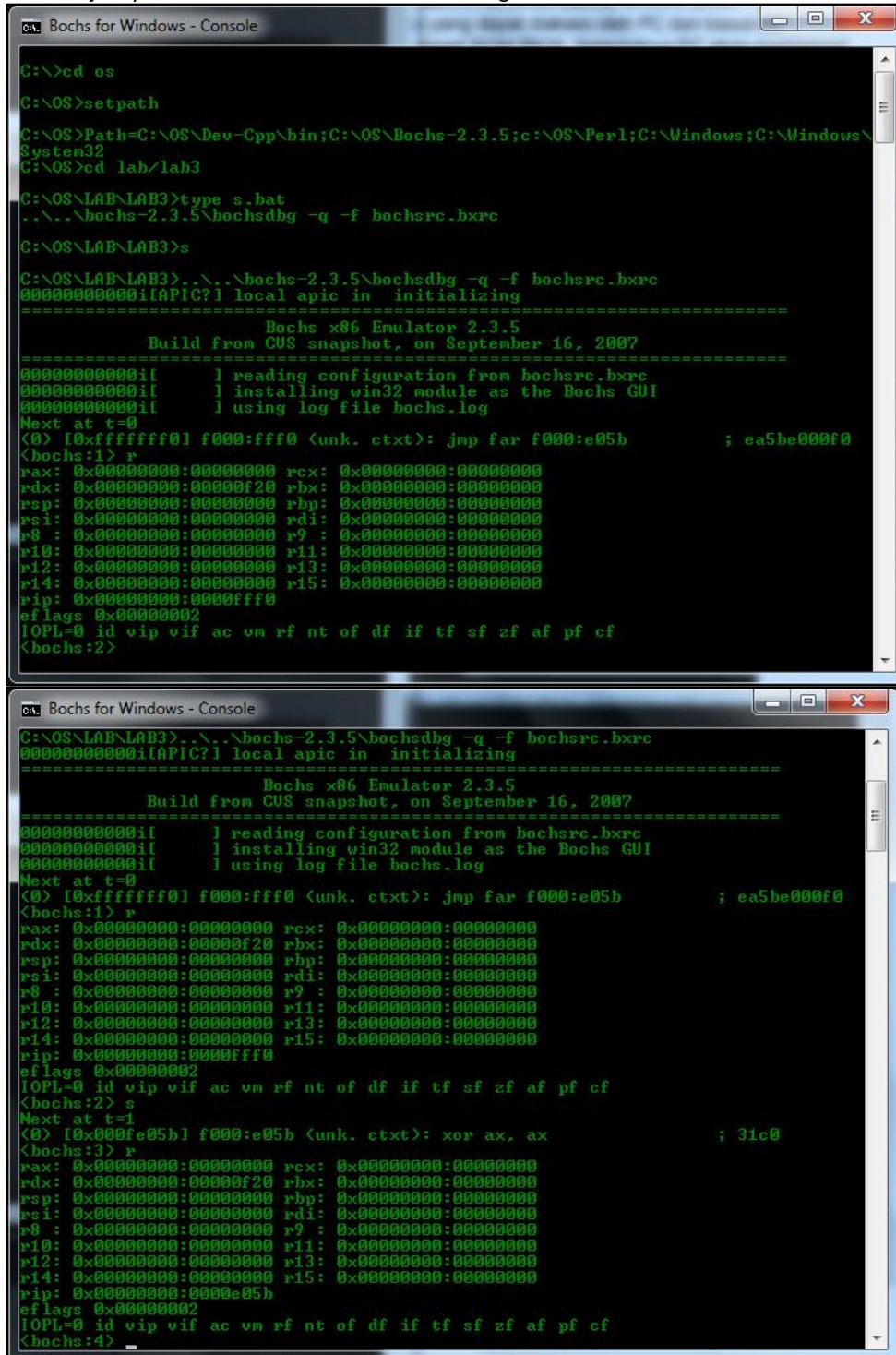
2. Ketikan "type s.bat", lalu mulai debugging dengan masukan perintah "s" Kondisi pada gambar di bawah menjelaskan kondisi PC pada mode 'Real-Mode' yang sedang akan menjalankan program yang pertama kali (0), yaitu program yang terdapat pada alamat 'F000:FFF0',



```
Bochs for Windows - Console

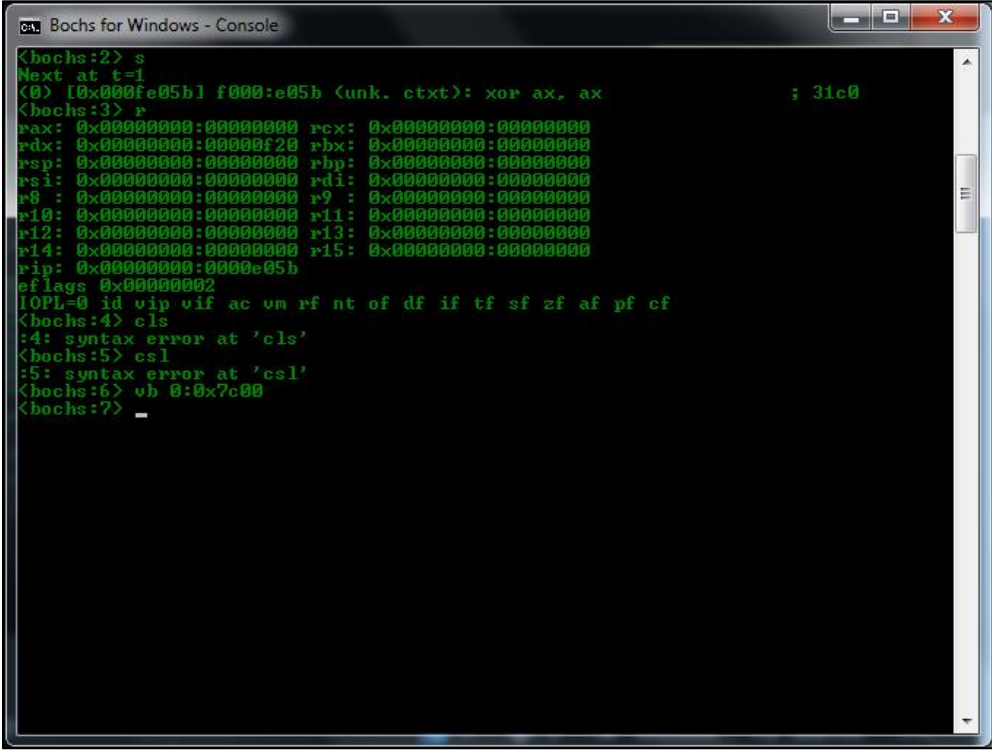
C:\>cd os
C:\OS>setpath
C:\OS>Path=C:\OS\Dev-Cpp\bin;C:\OS\Bochs-2.3.5;c:\OS\Perl;C:\Windows;C:\Windows\System32
C:\OS>cd lab/lab3
C:\OS\LAB\LAB3>type s.bat
..\\..\\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
C:\OS\LAB\LAB3>s
C:\OS\LAB\LAB3>..\\..\\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
000000000000i[aPIC?] local apic in  initializing
=====
                        Bochs x86 Emulator 2.3.5
                        Build from CVS snapshot, on September 16, 2007
=====
000000000000i[      ] reading configuration from bochsrc.bxrc
000000000000i[      ] installing win32 module as the Bochs GUI
000000000000i[      ] using log file bochs.log
Next at t=0
(0) [0xfffff0] f000:fff0 <unk. ctxt>: jmp far f000:e05b          ; ea5be00ff0
<bochs:1>
```

3. Selanjutnya ketikkan "s" dan "r" secara berulang.



```
C:\>cd os
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>type s.bat
..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
C:\OS\LAB\LAB3>s
C:\OS\LAB\LAB3>..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
000000000000i[APIC?] local apic in  initializing
=====
Bochs x86 Emulator 2.3.5
Build from CVS snapshot, on September 16, 2007
=====
000000000000i|      | reading configuration from bochsrc.bxrc
000000000000i|      | installing win32 module as the Bochs GUI
000000000000i|      | using log file bochs.log
Next at t=0
<0> [0xffffffff] f000:fff0 <unk. ctxt>: jmp far f000:e05b          ; ea5be000f0
<bochs:1> 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: 0x00000000:0000fff0
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<bochs:2>
C:\OS\LAB\LAB3>..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
000000000000i[APIC?] local apic in  initializing
=====
Bochs x86 Emulator 2.3.5
Build from CVS snapshot, on September 16, 2007
=====
000000000000i|      | reading configuration from bochsrc.bxrc
000000000000i|      | installing win32 module as the Bochs GUI
000000000000i|      | using log file bochs.log
Next at t=0
<0> [0xffffffff] f000:fff0 <unk. ctxt>: jmp far f000:e05b          ; ea5be000f0
<bochs:1> 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: 0x00000000:0000fff0
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<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>
```

4. Masukan perintah berikut 'vb 0:0x7C00' <ENTER> Maksud perintah ini adalah membuat titik pemberhentian (halte) pada alamat 0000:7C00.

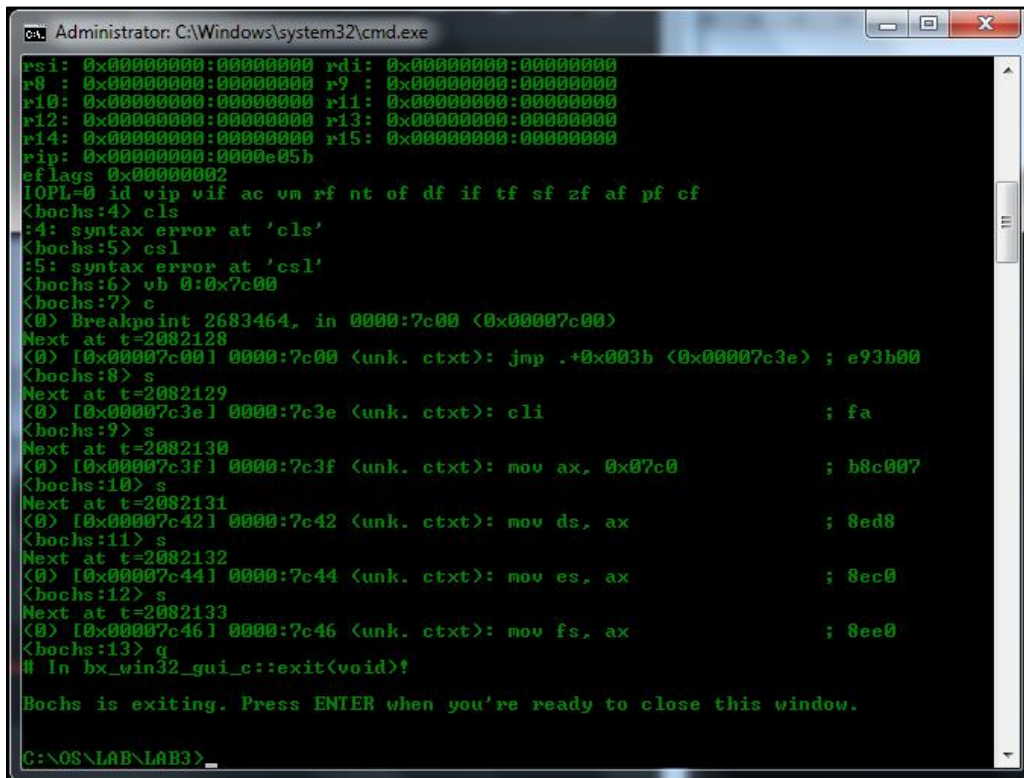


```
Bochs for Windows - Console
<bochs:2> s
Next at t=1
<0> [0x000fe05b] f000:e05b (unk. ctxt): xor ax, ax          ; 31c0
<bochs:3> r
rax: 0x00000000:00000000 rcx: 0x00000000:00000000
rdx: 0x00000000:00000f20 rbx: 0x00000000:00000000
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000
rsi: 0x00000000:00000000 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> cls
:4: syntax error at 'cls'
<bochs:5> csl
:5: syntax error at 'csl'
<bochs:6> vb 0:0x7c00
<bochs:7> _
```

5. Masukkan perintah “c”. Maksud perintah ini adalah teruskan (Continue) prosesnya sampai ke titik pemberhentian. Dalam sekejap PC sudah sampai pada pemberhentian yang kita buat di atas yaitu pada alamat 0000:7C00.

PC mulai memasuki tahapan ‘BOOTSTRAPLOADER’, untuk sampai pada tahap ini PC sudah menghabiskan clock sebanyak ‘2082128’ (dapat dilihat di “Next at t=2082128”).

Lalu sekarang PC akan mulai menjalankan program ‘boot.asm’. ketikan “s” secara berulang dan “q” Untuk menghentikan debugging.



```
Administrator: C:\Windows\system32\cmd.exe
rax: 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> cls
:4: syntax error at 'cls'
<bochs:5> csl
:5: syntax error at 'csl'
<bochs:6> vb 0:0x7c00
<bochs:7> c
<0> Breakpoint 2683464, in 0000:7c00 <0x00007c00>
Next at t=2082128
<0> [0x00007c00] 0000:7c00 <unk. ctxt>: jmp .+0x003b <0x00007c3e> ; e93b00
<bochs:8> s
Next at t=2082129
<0> [0x00007c3e] 0000:7c3e <unk. ctxt>: cli ; fa
<bochs:9> s
Next at t=2082130
<0> [0x00007c3f] 0000:7c3f <unk. ctxt>: mov ax, 0x07c0 ; b8c007
<bochs:10> s
Next at t=2082131
<0> [0x00007c42] 0000:7c42 <unk. ctxt>: mov ds, ax ; 8ed8
<bochs:11> s
Next at t=2082132
<0> [0x00007c44] 0000:7c44 <unk. ctxt>: mov es, ax ; 8ec0
<bochs:12> s
Next at t=2082133
<0> [0x00007c46] 0000:7c46 <unk. ctxt>: mov fs, ax ; 8ee0
<bochs:13> 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>
```


6. Kemudian buatlah break-point, masukan perintah 'vb 0x0100:0x0000' untuk menghentikan langkah saat PC mulai mengeksekusi instruksi dari program 'kernel.bin' lalu perintah "c".

The screenshot shows two windows from the Bochs for Windows application. The 'Console' window on the left displays the command prompt where the user has entered 'C:\OS\LAB\LAB3>' and executed 'C:\OS\LAB\LAB3>..\..\bochs-2.3.5\bochsdhg -q -f bochs.rc.bxrc'. The output shows the Bochs x86 Emulator 2.3.5 build from CVS snapshot, on September 16, 2007. It lists the configuration files used: 'bochs.rc.bxrc' for reading configuration, 'win32' for installing the module, and 'bochs.log' for logging. The console also shows the execution of the 'cli' instruction at address 0x00007c3e, followed by 'mov ax, 0x07c0' at 0x00007c3f, 'mov ds, ax' at 0x00007c42, 'mov es, ax' at 0x00007c44, and 'mov fs, ax' at 0x00007c46. The 'Display' window on the right shows the Bochs BIOS boot screen, including the version (0.6a 19 Aug 2006), license (GNU LGPL), and the message 'Bochs VBE Display Adapter enabled'. It also shows the BIOS build date (09/10/07) and revision (1.183 \$ \$Date: 2007/09/10 20:00:29 \$). The boot process is shown as 'Booting from Floppy...' and 'Loading kernel ver 0.01'.

7. Selanjutnya teruskan langkah PC Simulator step-by-step minimal sebanyak 10x, ketik 's',

The screenshot shows the same two windows as before, but the 'Console' window now displays the execution of the 'c' instruction at address 0x00007c46, followed by 'mov ax, 0x0100' at 0x00007c47, 'mov ds, ax' at 0x00007c48, 'mov es, ax' at 0x00007c49, 'cli' at 0x00007c4a, 'mov ss, ax' at 0x00007c4b, 'mov sp, 0xffff' at 0x00007c4c, 'sti' at 0x00007c4d, 'push dx' at 0x00007c4e, 'push es' at 0x00007c4f, 'xor ax, ax' at 0x00007c50, and 'mov es, ax' at 0x00007c51. The 'Display' window remains the same, showing the BIOS boot screen and the message 'Booting from Floppy...' and 'Loading kernel ver 0.01'.