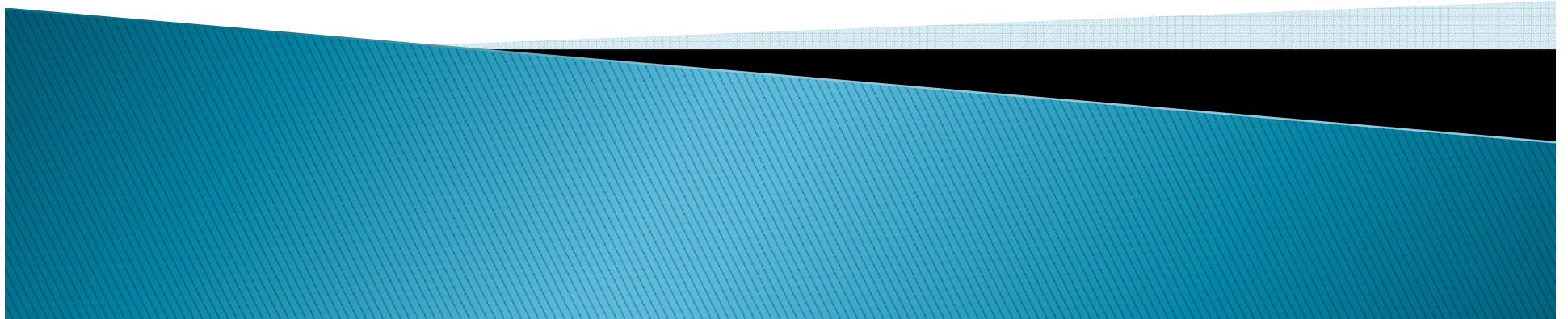


EMULATION

RAHUL WAGH

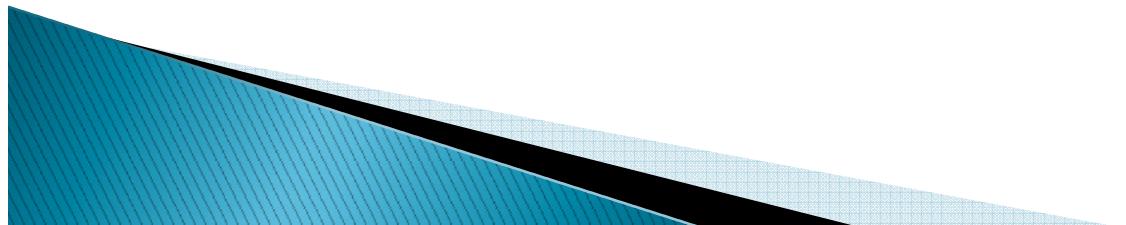
Department of Electrical & Computer Engineering
Portland State University



FUNCTIONAL VERIFICATION

► Review:

- What?
- Why?
- How?



FUNCTIONAL VERIFICATION: THE PLAYERS

- ▶ “*Big Three*”

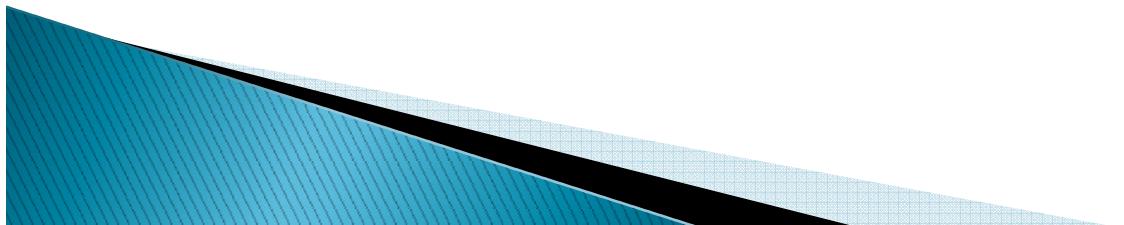


- ▶ Others:



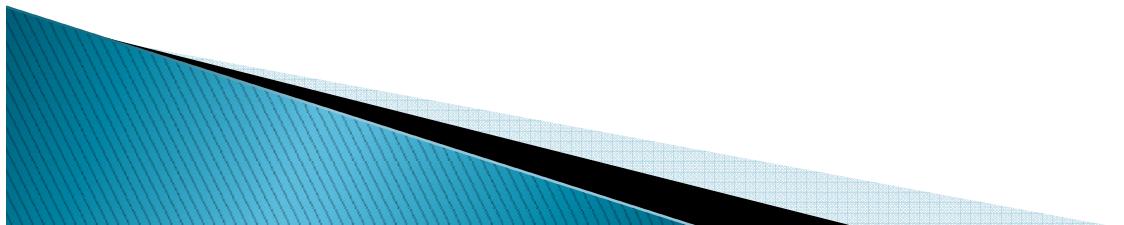
SOFTWARE SIMULATORS

- ▶ A software program that simulates the abstract model of a system by taking an input representation of the circuit and by processing the HDL and compiling it
- ▶ Eg: ModelSim, VCS, NC-Sim, VeriLogger Extreme



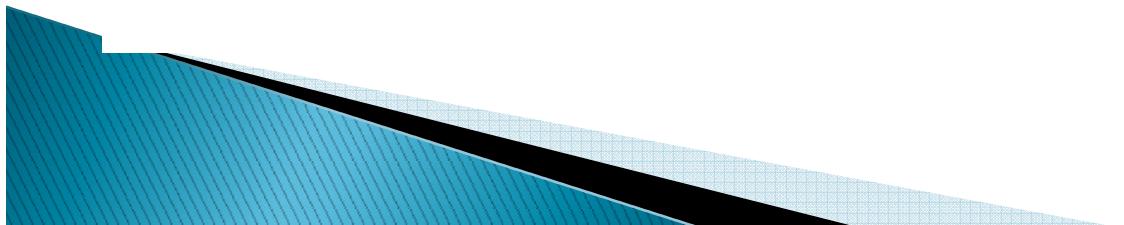
SOFTWARE SIMULATORS: ADVANTAGES

- ▶ Great for preliminary testing and in early stages of verification
- ▶ Simplicity
- ▶ Full visibility



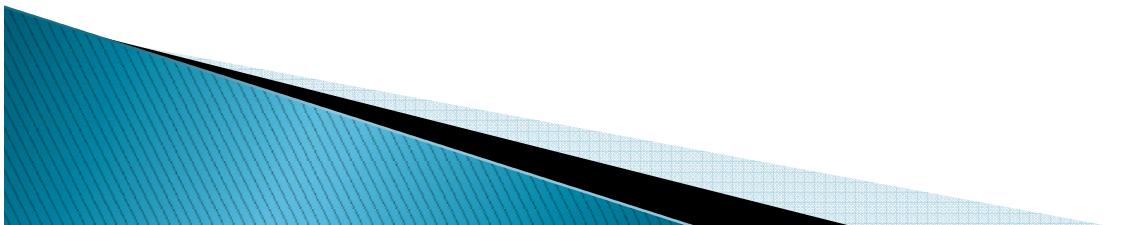
SIMULATION: SHORTCOMINGS

- ▶ The hardware underneath is an x86 processor, hence performance scaling has flattened out
- ▶ The software tool runs at Hz/KHz – way too slow to capture events that run for just a short time on actual hardware



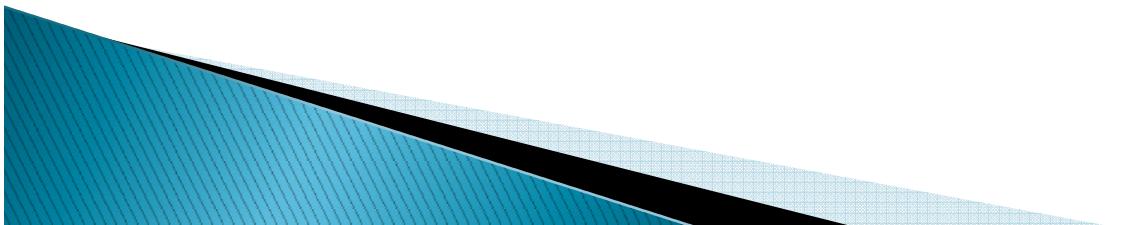
SOLUTION I: SIMULATION ACCELERATION

- ▶ The synthesizable elements of the DUT are mapped on to a hardware such a GPU or an FPGA
- ▶ Remaining part still runs on the software simulator on a PC/workstation
 - Eg: Nvidia Kepler GPU



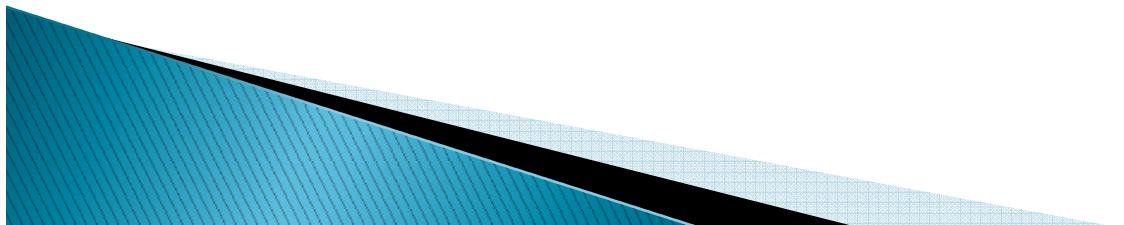
SOLUTION II: FPGA PROTOTYPING

- ▶ Implementation of the entire SoC on one/more FPGAs
- ▶ Real-time input and output
- ▶ Limitations:
 - FPGA Partitioning
 - Limited verification support provided by tools



SOLUTION III: EMULATORS!

- ▶ Map an entire design into gates which are executed on the emulator
- ▶ 10x to 1000x times faster than conventional simulation
- ▶ Verification-centric EDA tools



EMULATORS

- ▶ Mentor Graphics: Veloce



- ▶ Synopsys/EVE: Zebu

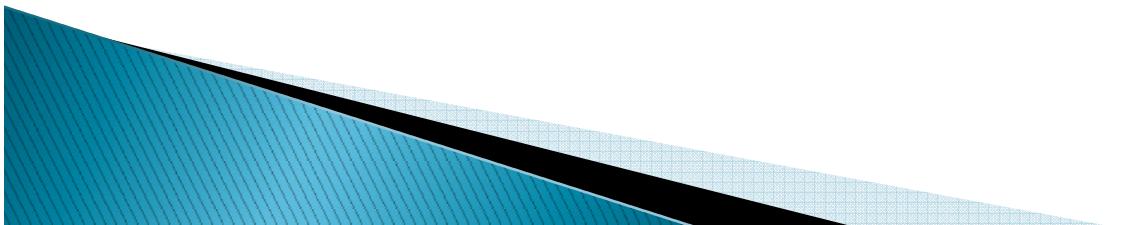


- ▶ Cadence: Palladium



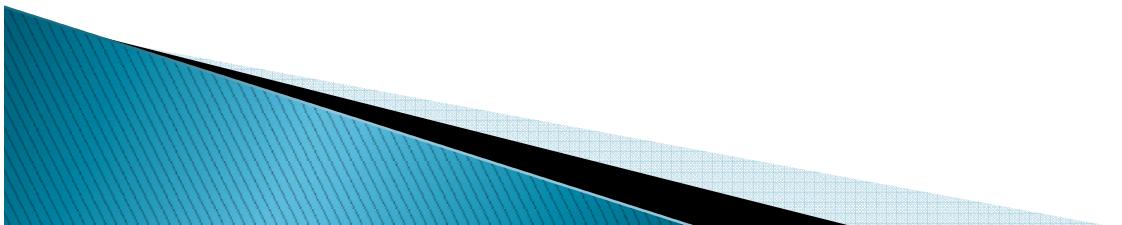
EMULATORS: SELECTION CRITERIA

- ▶ 1. Price/Gate
- ▶ 2. Initialization and Dedicated Support
- ▶ 3. Capacity
- ▶ 4. Primary Target Designs
- ▶ 5. Speed Range
- ▶ 6. Partitioning
- ▶ 7. Compile Time



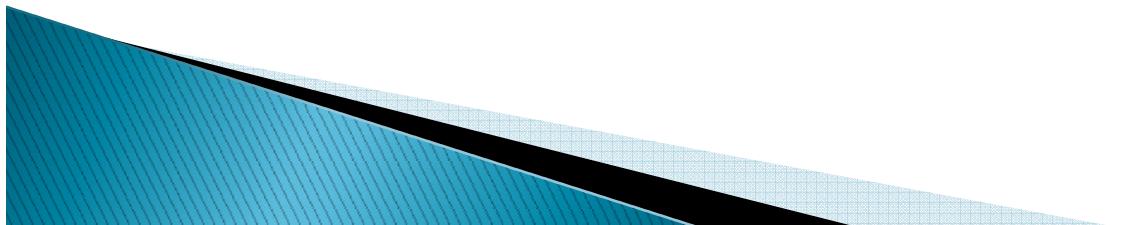
EMULATORS: SELECTION CRITERIA

- ▶ 8. Visibility
- ▶ 9. Debug
- ▶ 10. Virtual Platform API
- ▶ 11. Transactor Availability
- ▶ 12. Verification Language and Native Support
- ▶ 13. Number of Users
- ▶ 14. Memory Capacity



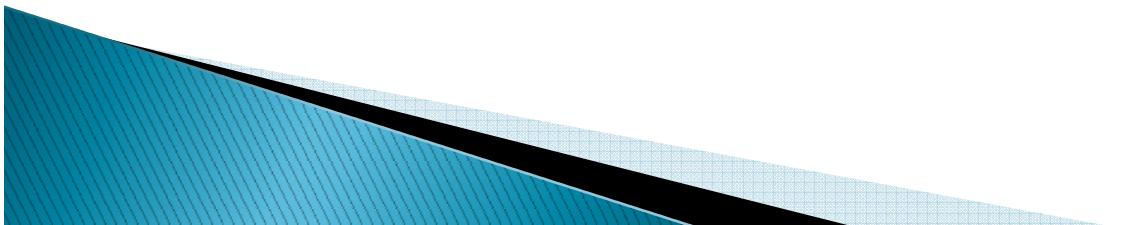
EMULATORS: COMPARISON

Vendor	Cadence, Mentor Graphics	Synopsys/EVE	Aldec, Bluespec
Architecture	Processor-based	FPGA-based	FPGA-based
Price/Gate	2 – 5 cents	0.5 – 2 cents	0.25 – 1 cents
Dedicated Support	Yes	Mixed	No
Speed Range (cycles/sec)	100K – 2M	500K – 5M	500K – 20M
Compile time	10M –30M gates/hr	25M–100M gates/hr	1M–15M gates/hr
Partitioning	Automated	Automated	Semi-automated



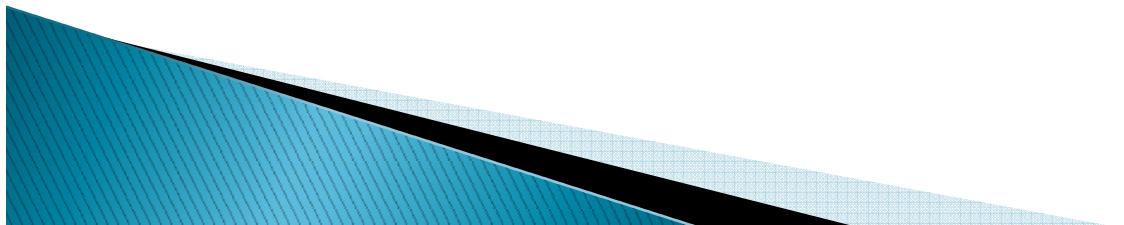
EMULATORS: COMPARISON

Vendor	Cadence, Mentor Graphics	Synopsys/EVE	Aldec, Bluespec
Visibility	Full visibility	Static/dynamic	Static/dynamic
Verification language native support	C++, SystemC, Specman e, SystemVerilog, OVM, SVA, PSL, OVL	Synthesizable Verilog, VHDL, SystemVerilog	Synthesizable Verilog, VHDL, SystemVerilog
Memory	Up to 1TB	Up to 200GB	Up to 32GB
Users	1 - 512 users	1 - 49 users	1 user



EMULATION: SHORTCOMINGS?

- ▶ Cost! – of the order of a million dollars
- ▶ Maintenance Cost
- ▶ High Compile time
- ▶ Not feasible for smaller designs

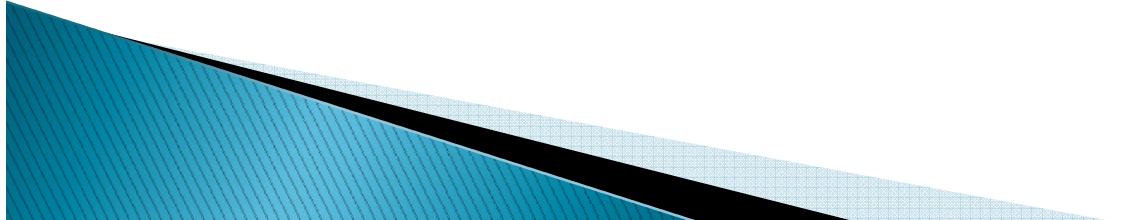


VELOCE EMULATORS



Your ticket to speed.

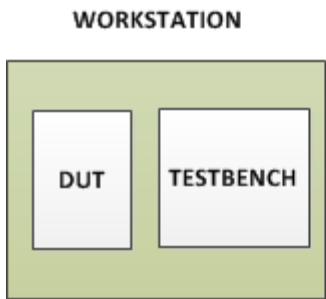
- ▶ 2007: Veloce Solo, Trio, Quattro, Grande
- ▶ 2008: Veloce Maximus
- ▶ 2012: Veloce2 Quattro, Veloce2 Maximus,
Veloce2 Double Maximus



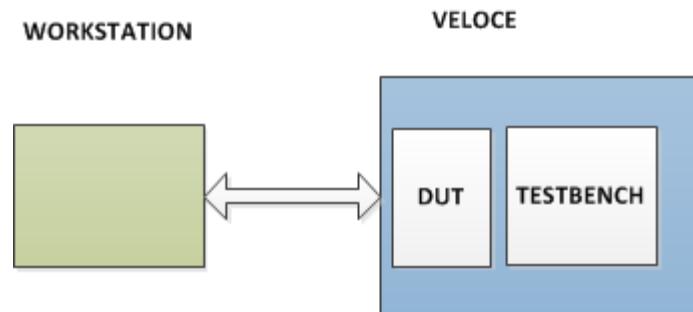
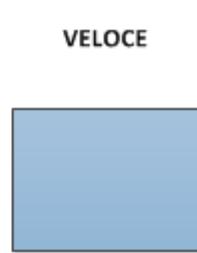
VELOCE2 EMULATORS: SPECS

Specification	Description
Capacity	Up to 2 billion gates
Number of users	Up to 128
Design Languages	Verilog, VHDL, SystemVerilog, gate-level netlist, and encrypted IP
Testbench Languages	C, C++, SystemC, SystemVerilog
Methodologies	UVM, OVM, TLM, VMM
Assertion Languages	SVA, PSL, OVL, QVL
Fast Compile	Up to 40 MG per hour
Asynchronous clocks	Can be supplied using functional generators or external PLLs
OS support	RH 4 & 5, SUSE 10 & 11
Power	About 11KW for a fully loaded system

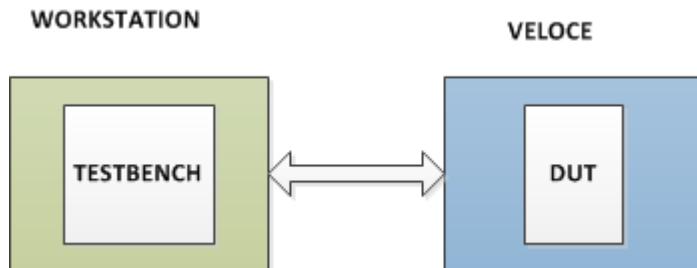
VELOCE: MODES



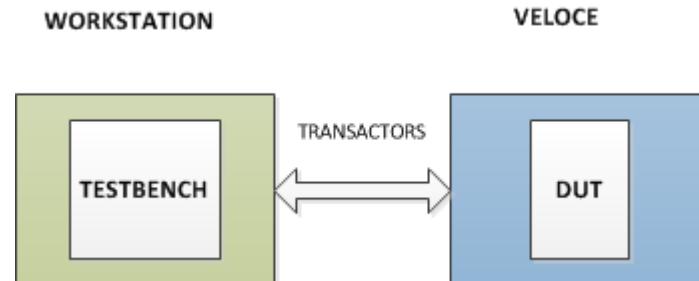
Conventional Simulation



In-Circuit Emulation (ICE)



Co-Simulation (HDL-Link)



Co-Modeling (TBX)



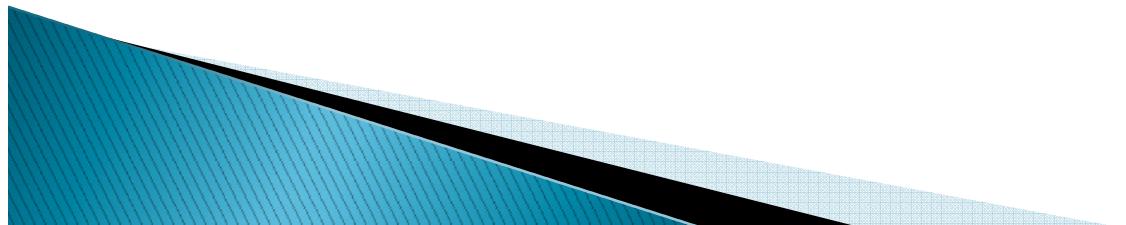
VELOCE AT PSU

- ▶ VeloceSolo
- ▶ Donated by Mentor Graphics Emulation Division (MED) to PSU in 2010
- ▶ Videos, seminar slides and tutorials available at <https://veloce.ece.pdx.edu/>
- ▶ Must be on-campus or VP Ned to CECS; must have an MCECS account



VELOCE TUTORIALS

- ▶ Using Mentor Veloce at PSU
- ▶ Veloce Stand-alone Mode Tutorial



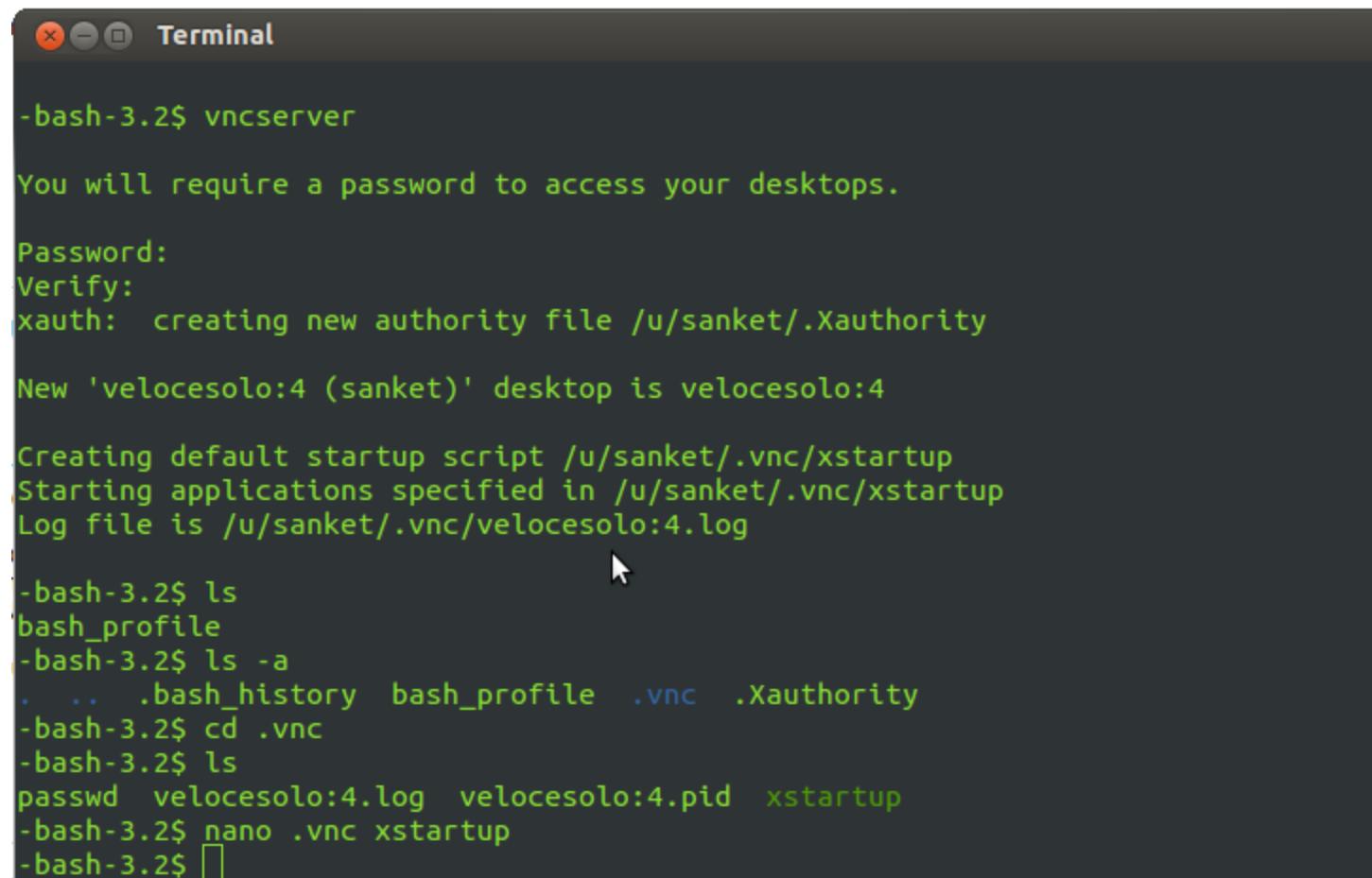
USING MENTOR VELOCE AT PSU

```
Terminal
-bash-3.2$ ls
bash_profile
-bash-3.2$ ls -a
. .. .bash_history bash_profile .vnc .Xauthority
-bash-3.2$ cd .vnc
-bash-3.2$ ls
passwd velocesolo:4.log velocesolo:4.pid xstartup
-bash-3.2$ nano .vnc xstartup
-bash-3.2$ vncviewer velocesolo:4
-bash: vncviewer: command not found
-bash-3.2$ exit
logout
Connection to velocesolo closed.
sanket@fab04:~$ vncviewer velocesolo:4

VNC Viewer Free Edition 4.1.1 for X - built Feb  5 2012 20:01:21
Copyright (C) 2002-2005 RealVNC Ltd.
See http://www.realvnc.com for information on VNC.

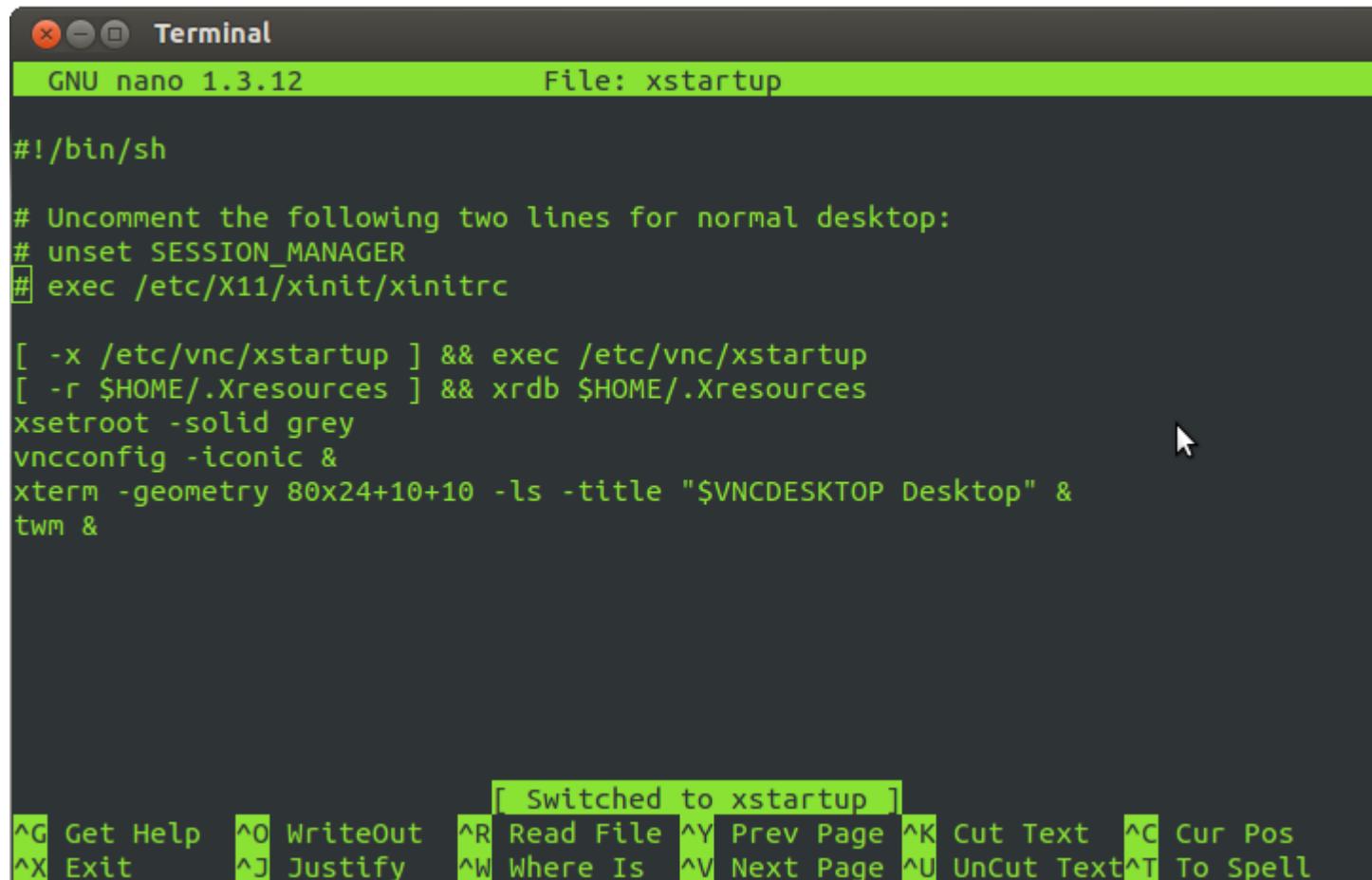
Mon Apr 28 17:12:51 2014
CConn:      connected to host velocesolo port 5904
CConnection: Server supports RFB protocol version 3.8
CConnection: Using RFB protocol version 3.8
Password: 
```

USING MENTOR VELOCE AT PSU



```
-bash-3.2$ vncserver  
  
You will require a password to access your desktops.  
  
Password:  
Verify:  
xauth: creating new authority file /u/sanket/.Xauthority  
  
New 'velocesolo:4 (sanket)' desktop is velocesolo:4  
  
Creating default startup script /u/sanket/.vnc/xstartup  
Starting applications specified in /u/sanket/.vnc/xstartup  
Log file is /u/sanket/.vnc/velocesolo:4.log  
  
-bash-3.2$ ls  
bash_profile  
-bash-3.2$ ls -a  
. .. .bash_history bash_profile .vnc .Xauthority  
-bash-3.2$ cd .vnc  
-bash-3.2$ ls  
passwd velocesolo:4.log velocesolo:4.pid xstartup  
-bash-3.2$ nano .vnc xstartup  
-bash-3.2$ █
```

USING MENTOR VELOCE AT PSU



```
GNU nano 1.3.12          File: xstartup

#!/bin/sh

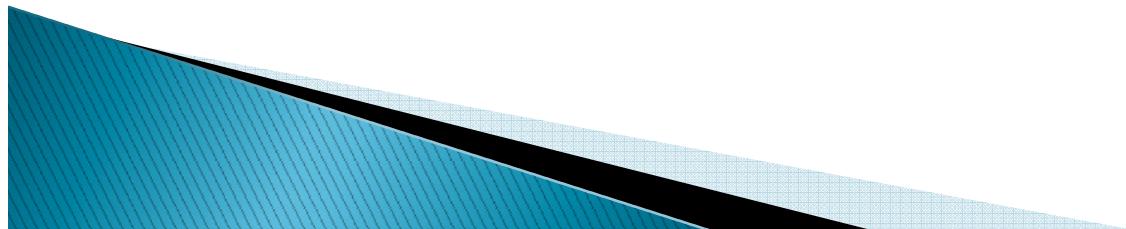
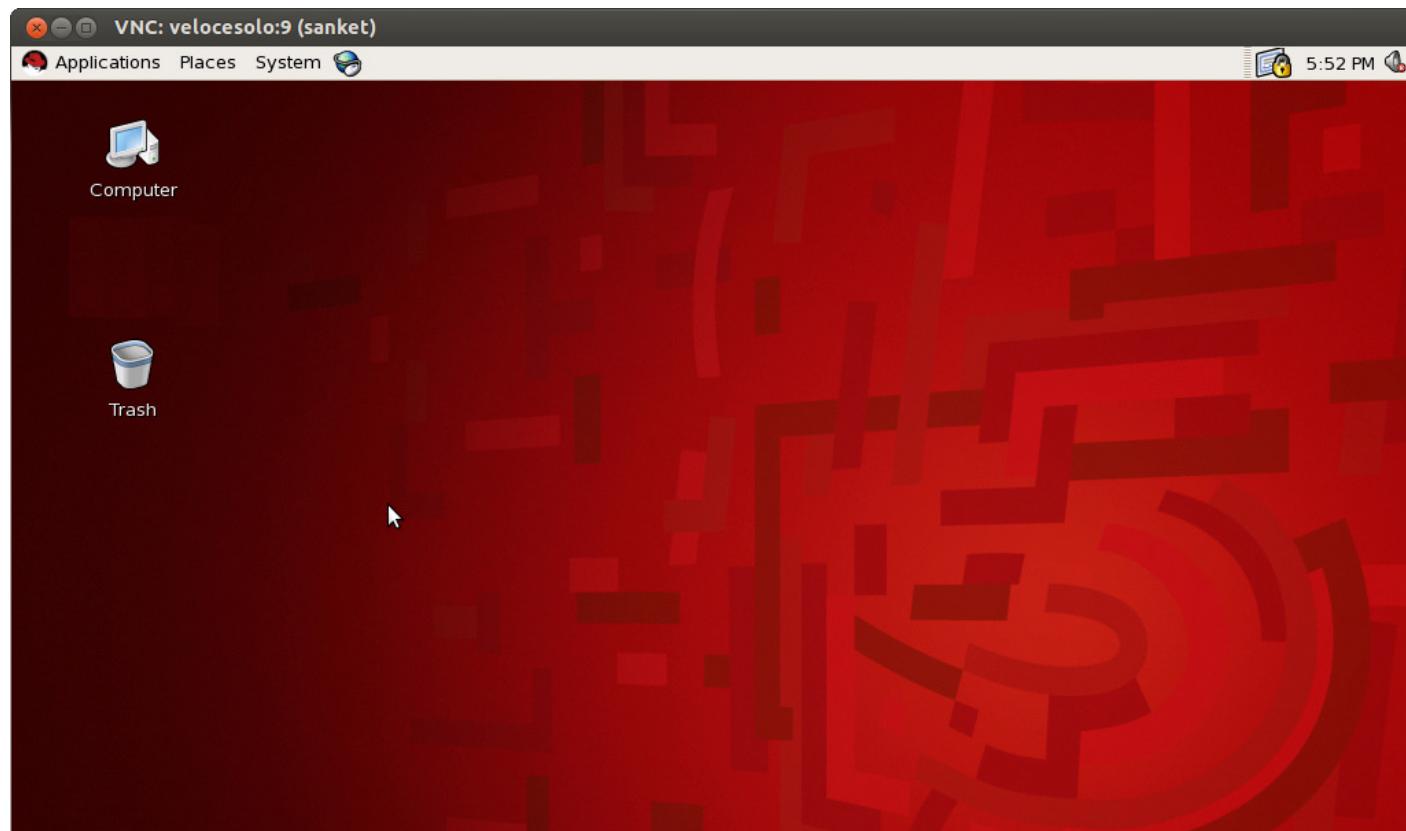
# Uncomment the following two lines for normal desktop:
# unset SESSION_MANAGER
# exec /etc/X11/xinit/xinitrc

[ -x /etc/vnc/xstartup ] && exec /etc/vnc/xstartup
[ -r $HOME/.Xresources ] && xrdb $HOME/.Xresources
xsetroot -solid grey
vncconfig -iconic &
xterm -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
twm &
```

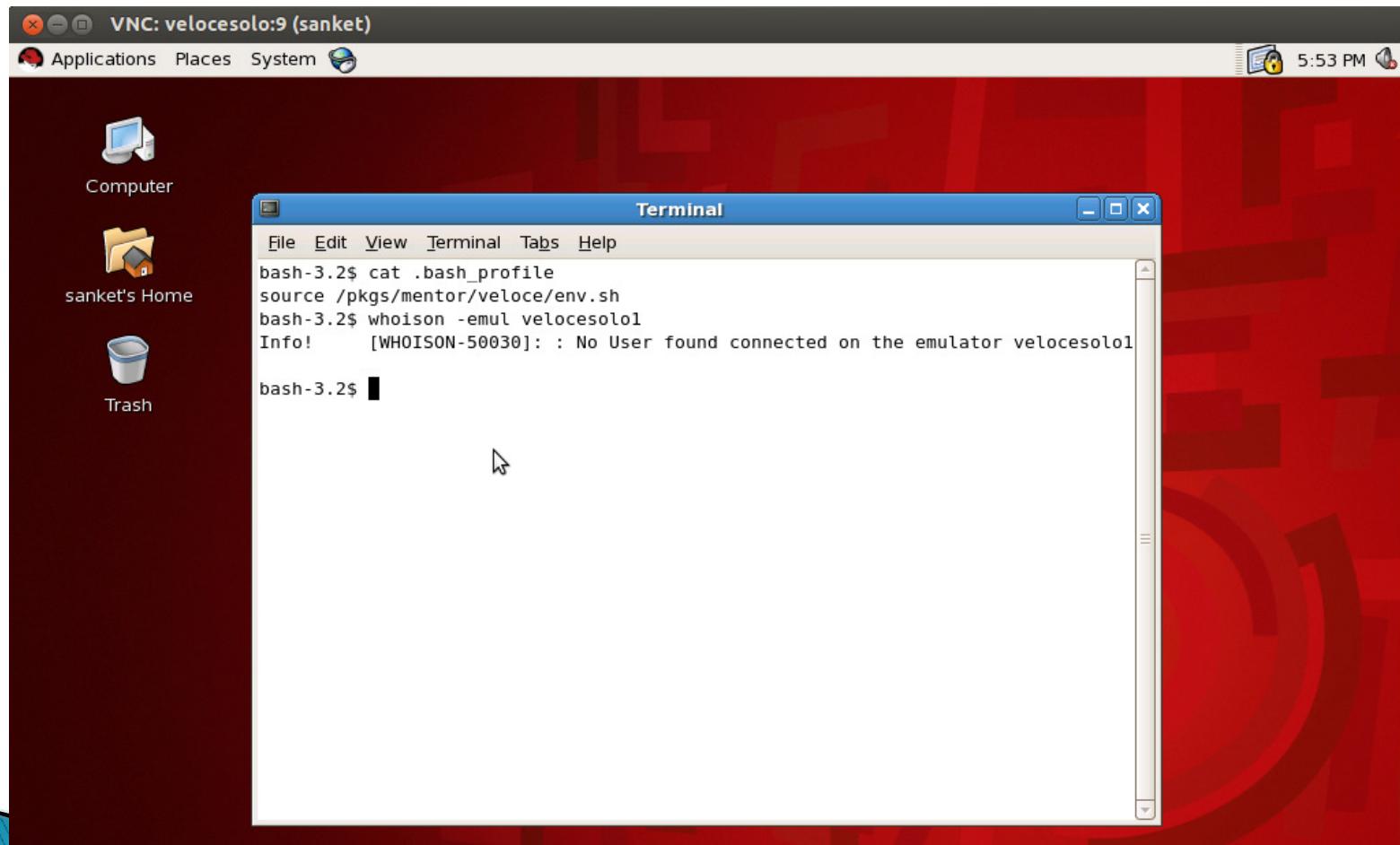
Switched to xstartup]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

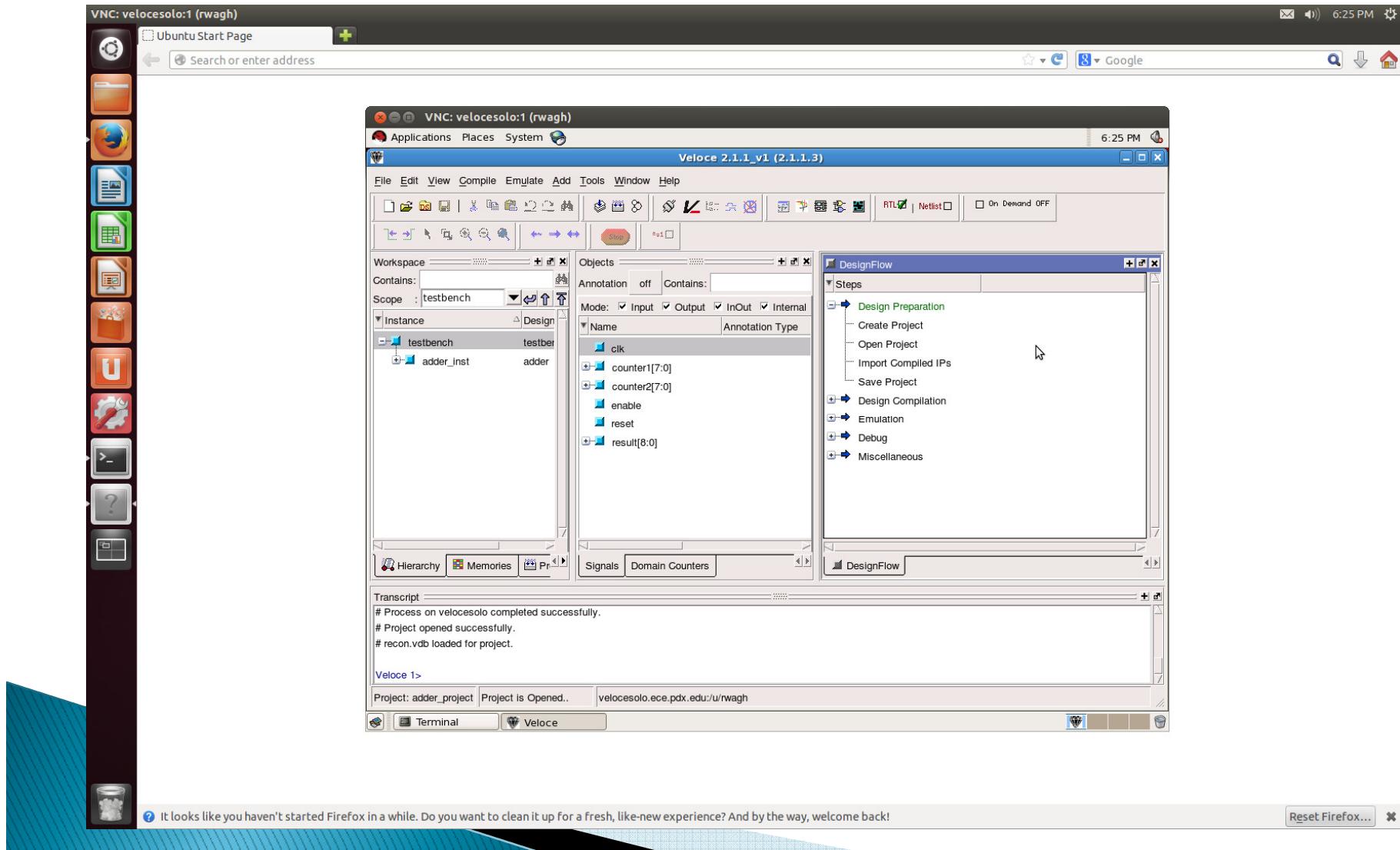
USING MENTOR VELOCE AT PSU



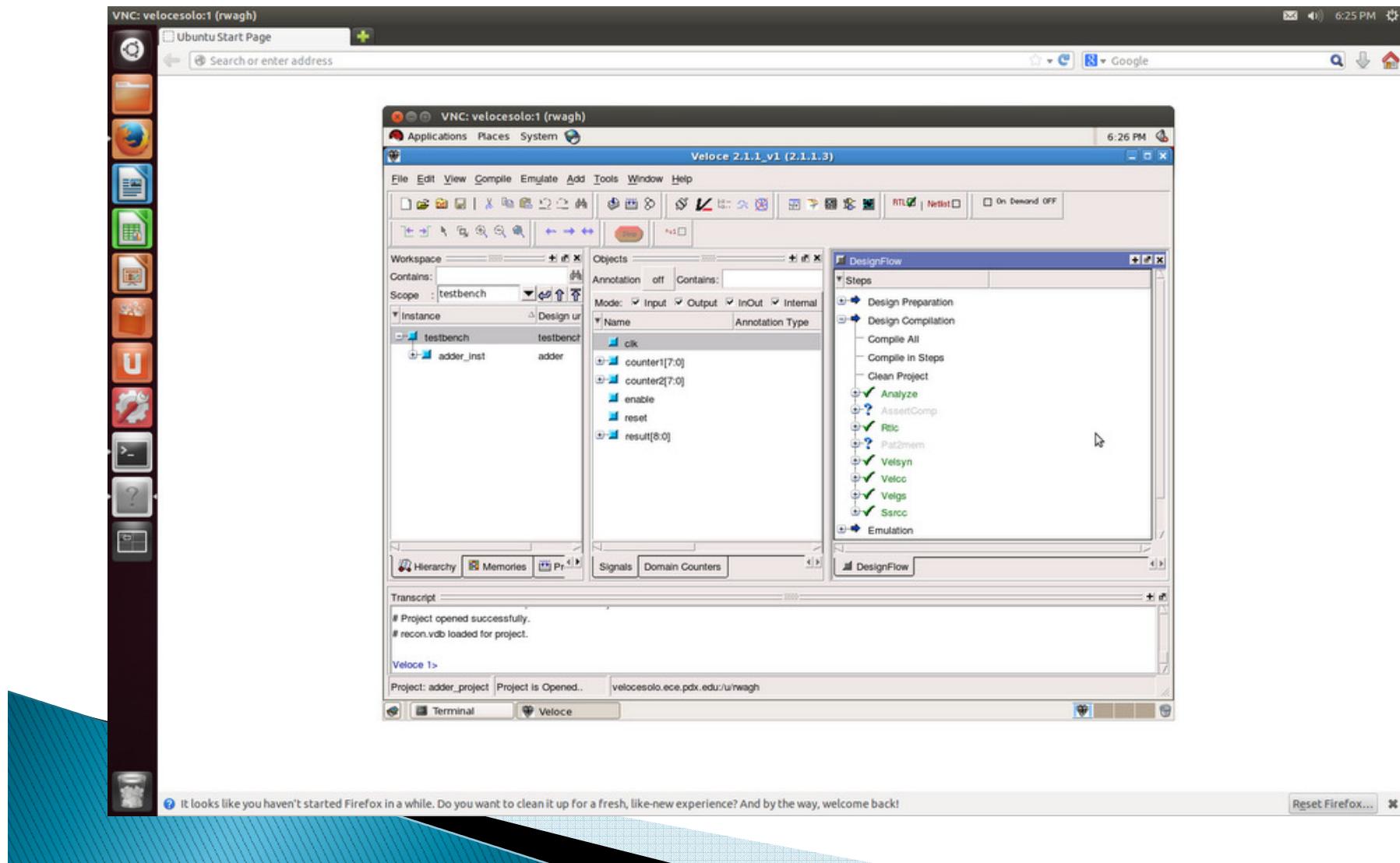
USING MENTOR VELOCE AT PSU



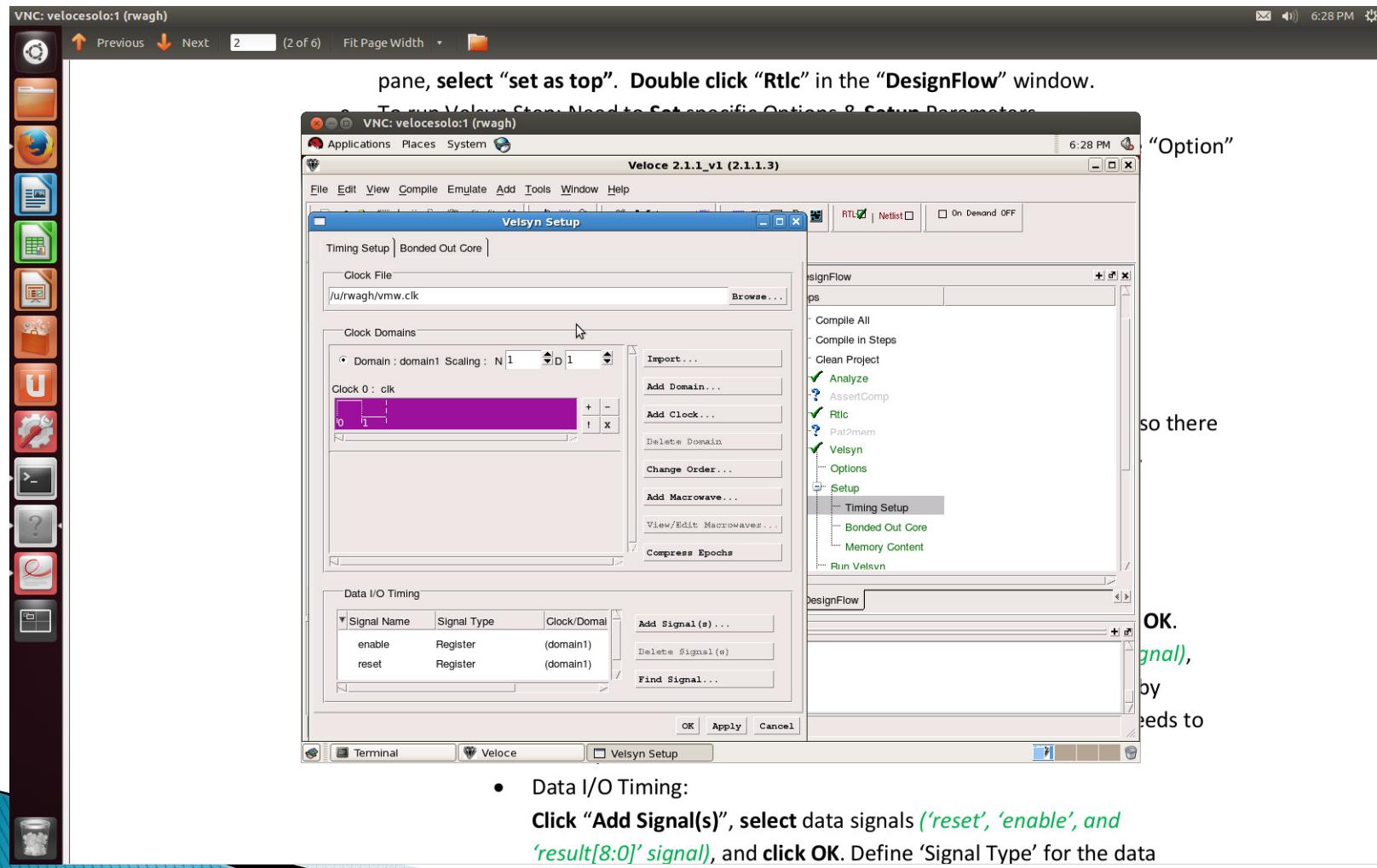
VELOCE STANDALONE MODE TUTORIAL



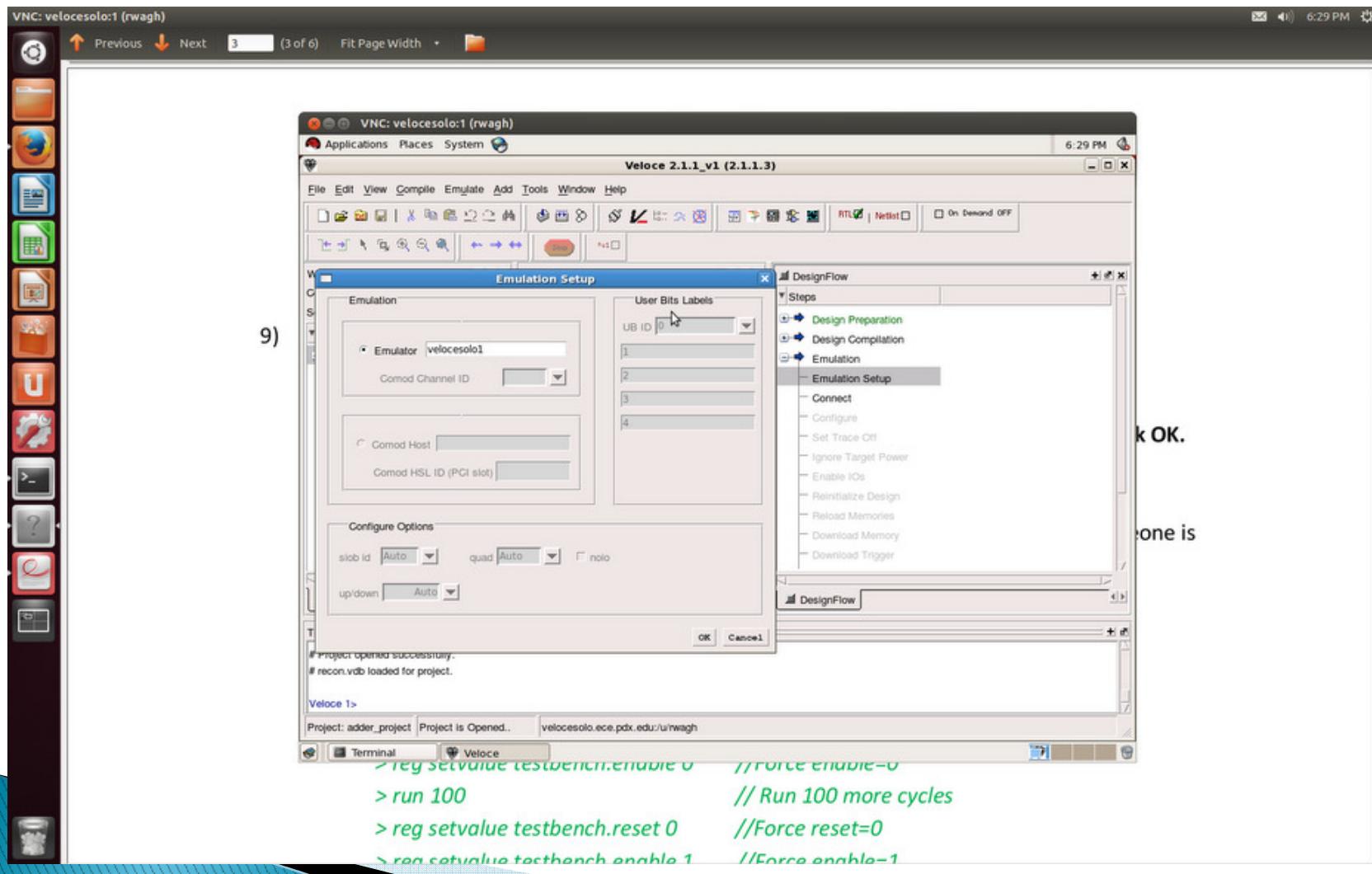
VELOCE STANDALONE MODE TUTORIAL



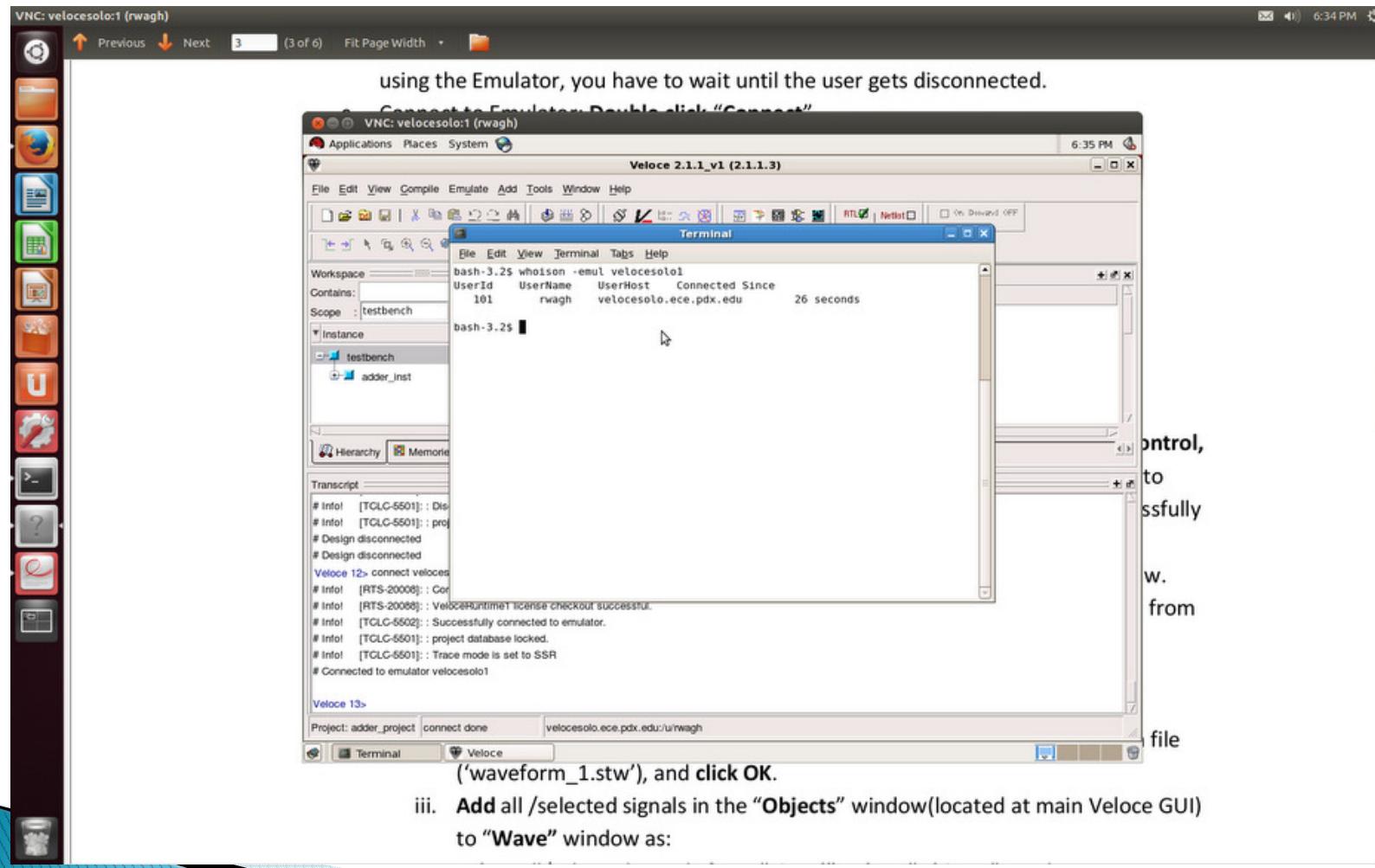
VELOCE STANDALONE MODE TUTORIAL



VELOCE STANDALONE MODE TUTORIAL



VELOCE STANDALONE MODE TUTORIAL



VELOCE STANDALONE MODE TUTORIAL

VNC: velocesolo:1 (rwagh) 6:32 PM

In the “DesignFlow” window, select “Emulation” and click “(+)”.
Double click “Emulation setup”, type Emulator name as “velocesolo1”, and click OK.

VNC: velocesolo:1 (rwagh)

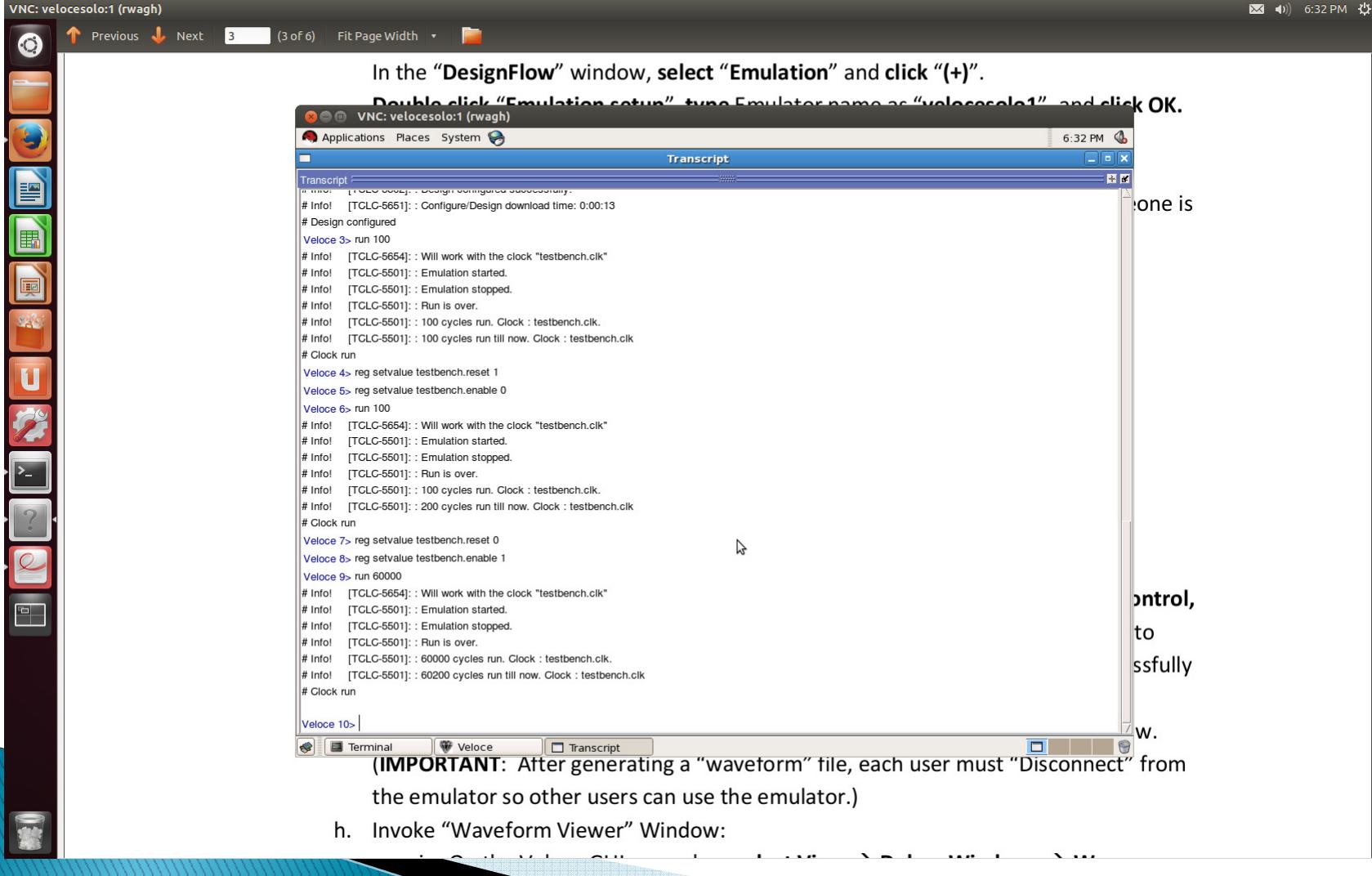
Applications Places System Transcript

Transcript:

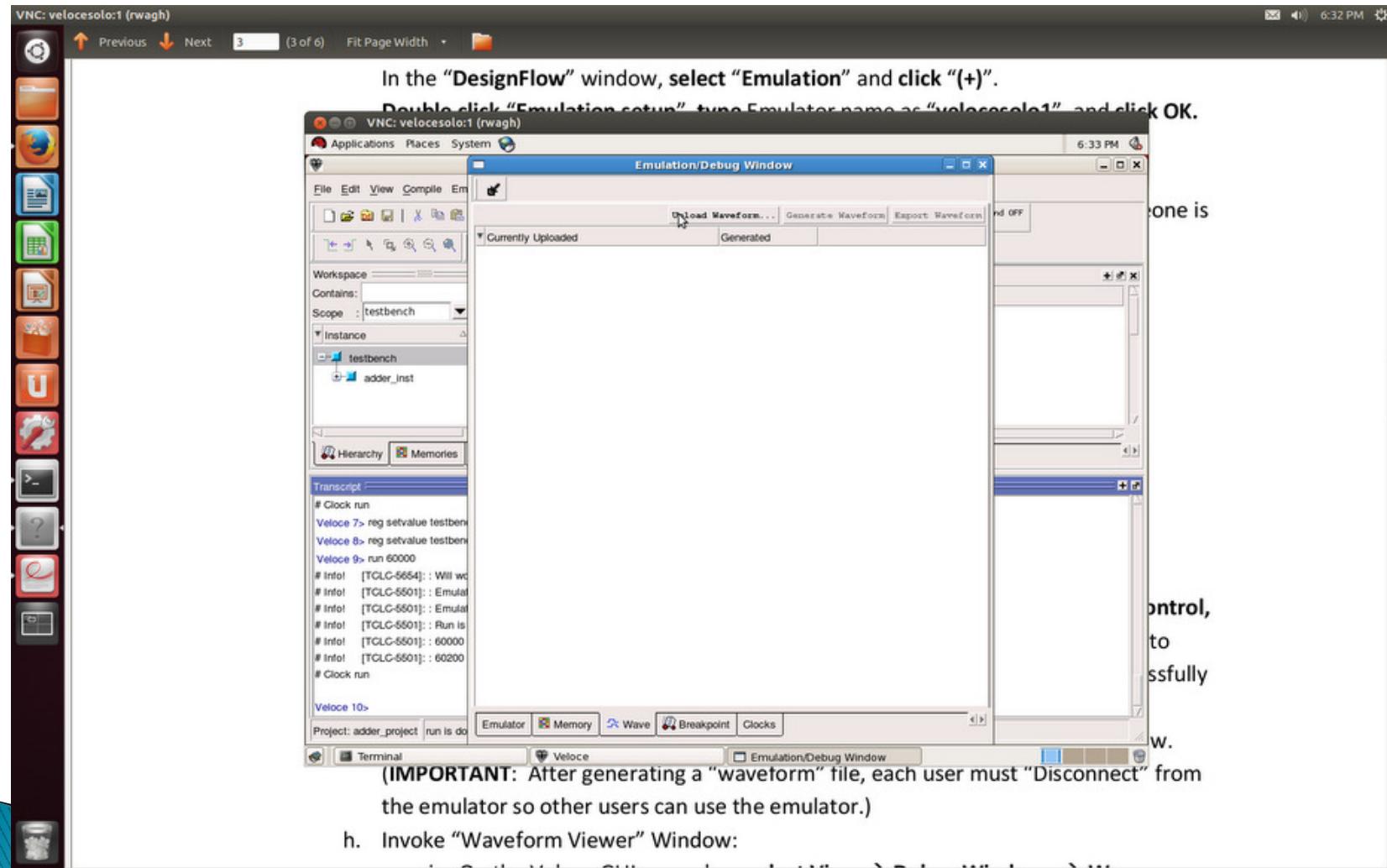
```
# Info! [TCLC-5654]: : Design configured successfully.
# Info! [TCLC-5651]: : Configure/Design download time: 0:00:13
# Design configured
Veloce 3> run 100
# Info! [TCLC-5654]: : Will work with the clock "testbench.clk"
# Info! [TCLC-5501]: : Emulation started.
# Info! [TCLC-5501]: : Emulation stopped.
# Info! [TCLC-5501]: : Run is over.
# Info! [TCLC-5501]: : 100 cycles run. Clock : testbench.clk.
# Info! [TCLC-5501]: : 100 cycles run till now. Clock : testbench.clk
# Clock run
Veloce 4> reg setvalue testbench.reset 1
Veloce 5> reg setvalue testbench.enable 0
Veloce 6> run 100
# Info! [TCLC-5654]: : Will work with the clock "testbench.clk"
# Info! [TCLC-5501]: : Emulation started.
# Info! [TCLC-5501]: : Emulation stopped.
# Info! [TCLC-5501]: : Run is over.
# Info! [TCLC-5501]: : 100 cycles run. Clock : testbench.clk.
# Info! [TCLC-5501]: : 200 cycles run till now. Clock : testbench.clk
# Clock run
Veloce 7> reg setvalue testbench.reset 0
Veloce 8> reg setvalue testbench.enable 1
Veloce 9> run 60000
# Info! [TCLC-5654]: : Will work with the clock "testbench.clk"
# Info! [TCLC-5501]: : Emulation started.
# Info! [TCLC-5501]: : Emulation stopped.
# Info! [TCLC-5501]: : Run is over.
# Info! [TCLC-5501]: : 60000 cycles run. Clock : testbench.clk.
# Info! [TCLC-5501]: : 60200 cycles run till now. Clock : testbench.clk
# Clock run
Veloce 10>
```

(IMPORTANT: After generating a “waveform” file, each user must “Disconnect” from the emulator so other users can use the emulator.)

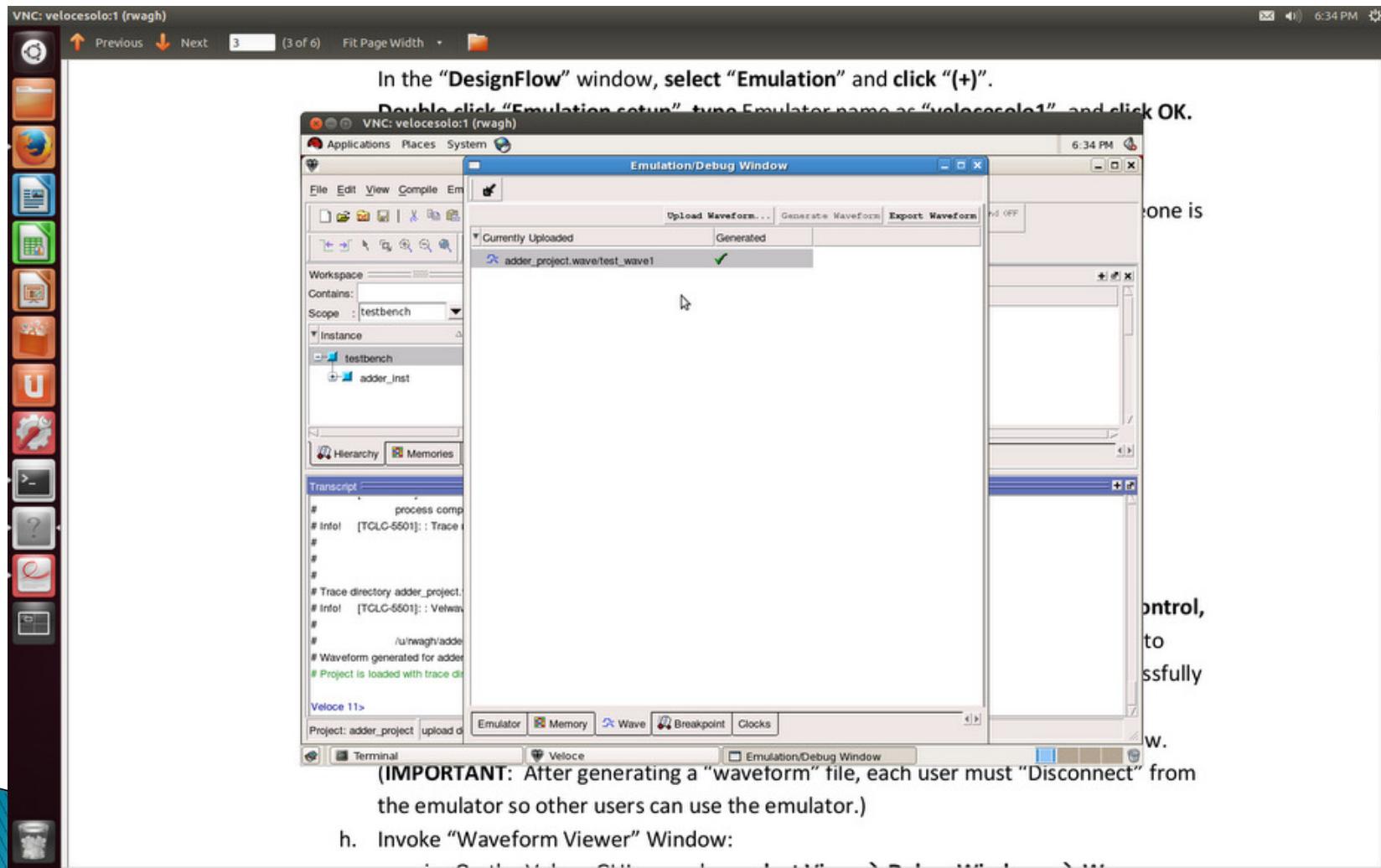
h. Invoke “Waveform Viewer” Window:



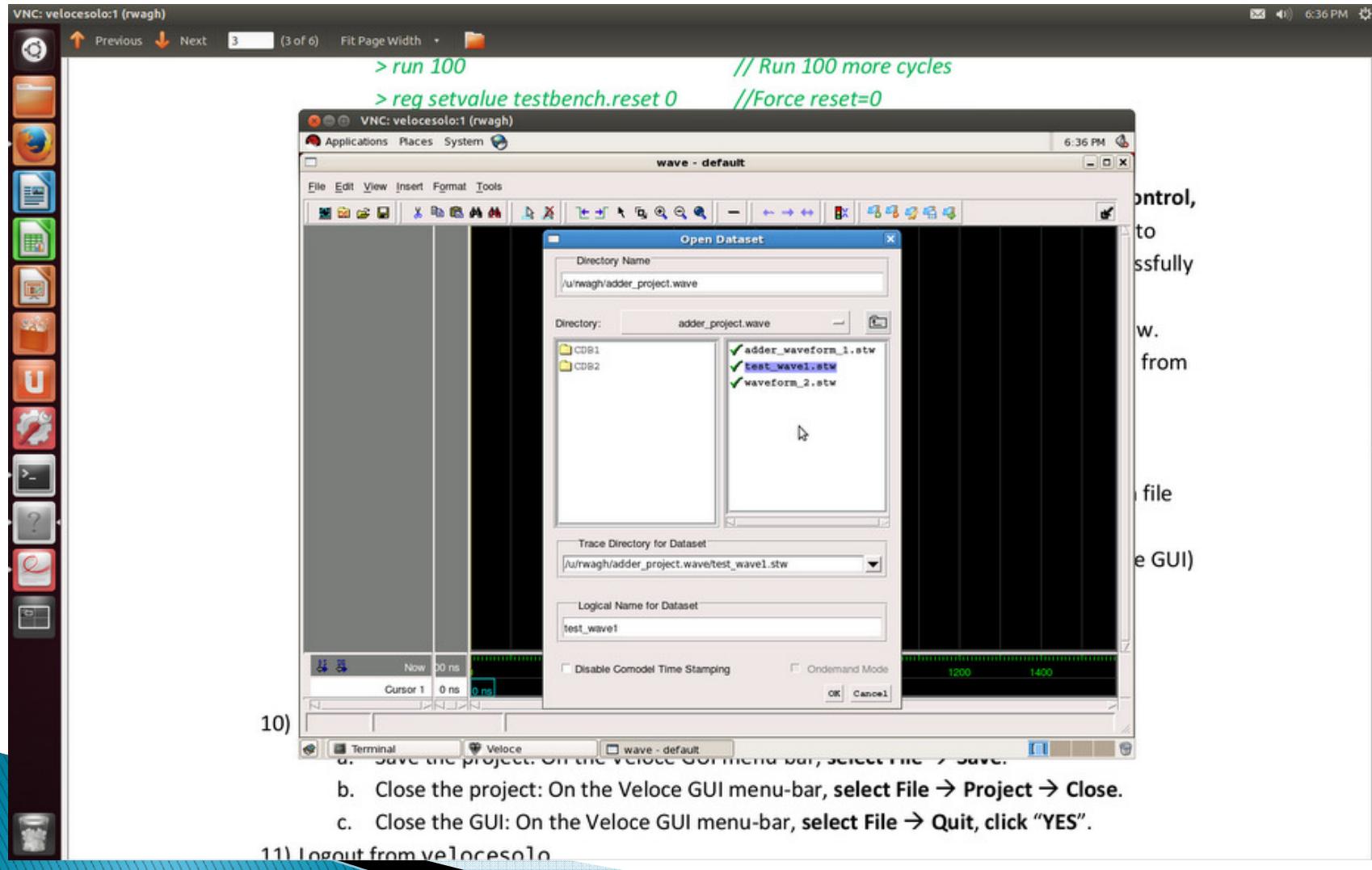
VELOCE STANDALONE MODE TUTORIAL



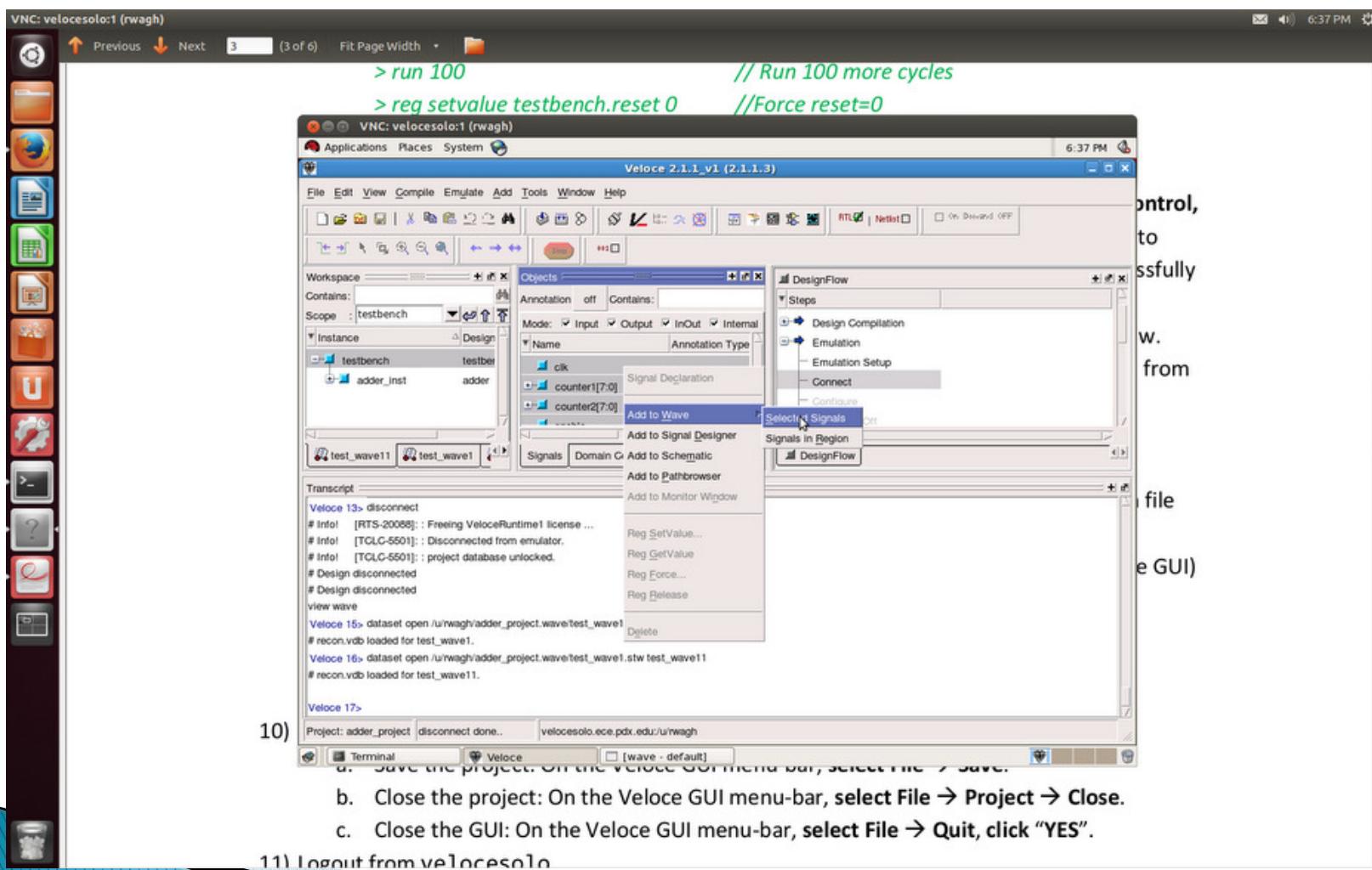
VELOCE STANDALONE MODE TUTORIAL



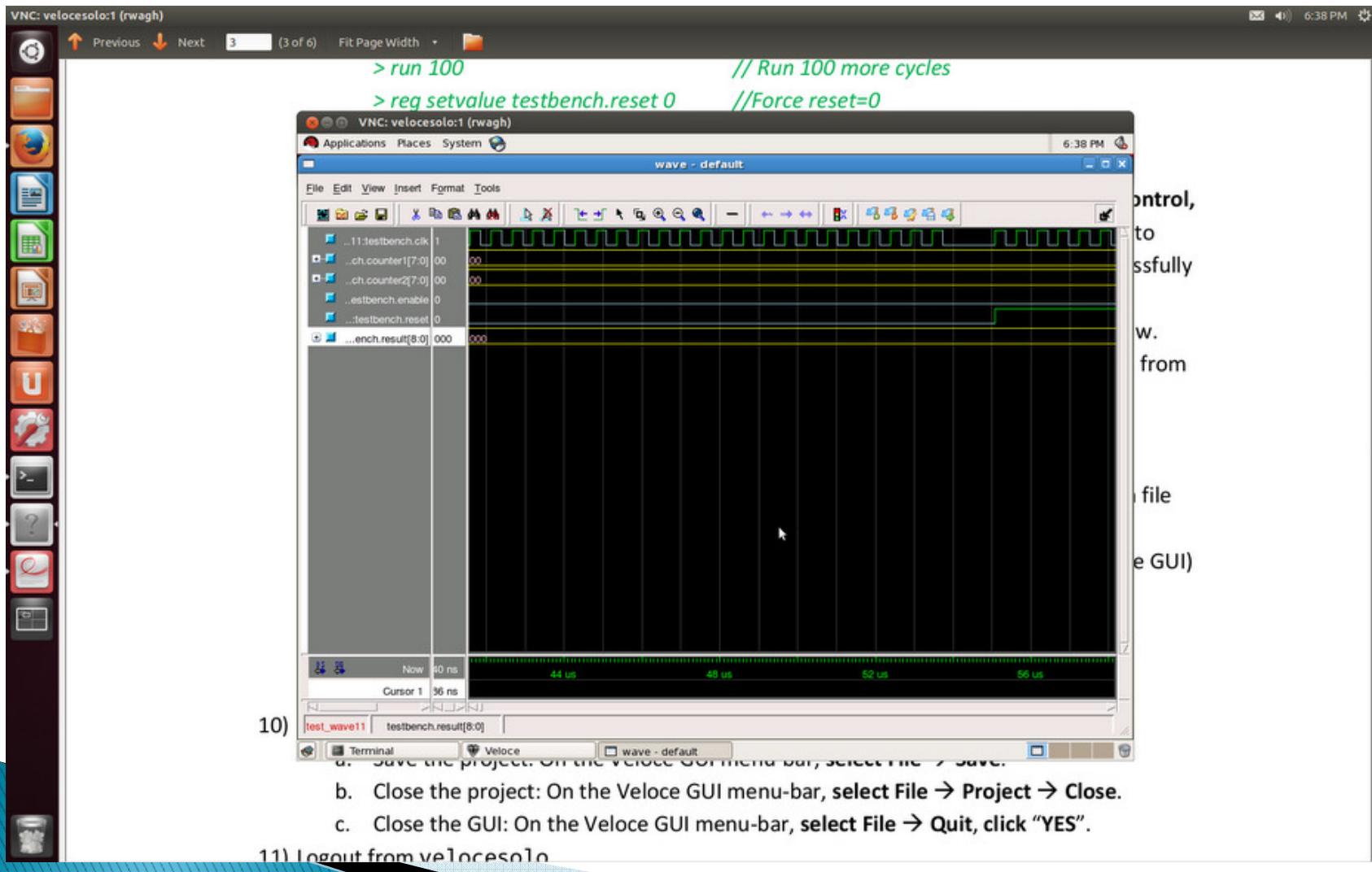
VELOCE STANDALONE MODE TUTORIAL



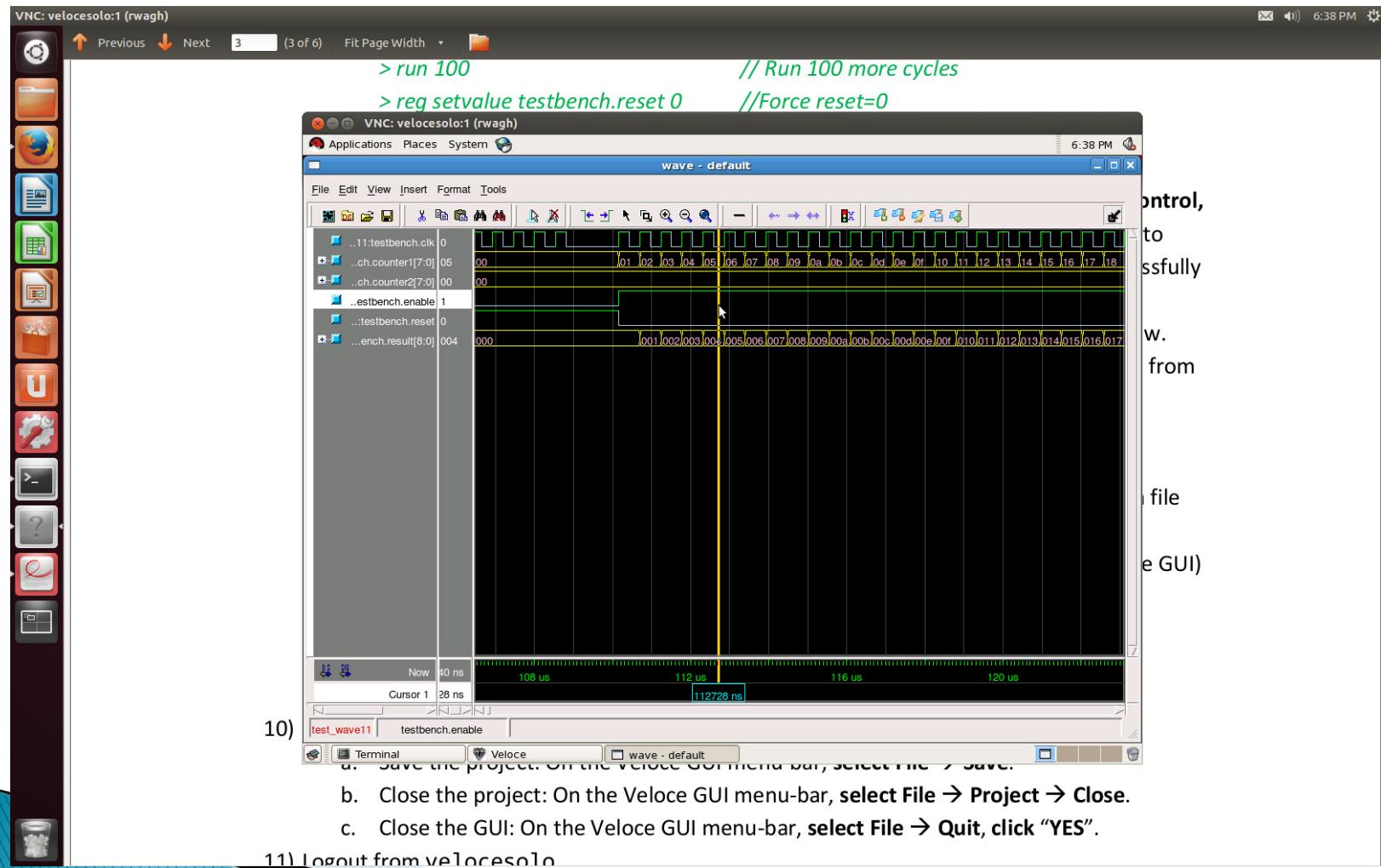
VELOCE STANDALONE MODE TUTORIAL



VELOCE STANDALONE MODE TUTORIAL



VELOCE STANDALONE MODE TUTORIAL



USING THE VELOCE AT PSU: GOTCHAS

- ▶ Change in the “Creating your environment” step in the Using Mentor Veloce at PSU (FIXED)

Creating your environment

linux.cec.pdx.edu - PuTTY

Welcome to VeloceSolo!

Recent changes to the veloceSolo environment:

To use the emulator tools on veloceSolo each user must set up their local environment by either sourcing /pkgs/mentor/veloce/env.sh "in place", or by creating and sourcing a local copy. **The /opt/veloce/env1415.sh script is no longer globally sourced on login.**

Bash shell users (the default shell) can set up their environment to source this file by running the following command. This command only needs to be run once.

echo "source /pkgs/mentor/veloce/env.sh" >> ~/.bash_profile

Running this command will add a line to your bash login script which sources the /pkgs/mentor/veloce/env.sh script, configuring your environment automatically upon login. After running the command, it is necessary to logout and log back in for the changes to take effect.

-bash: /opt/veloce/env1415.sh: No such file or directory
-bash: /opt/veloce/env1415.sh: No such file or directory
-bash-3.2\$

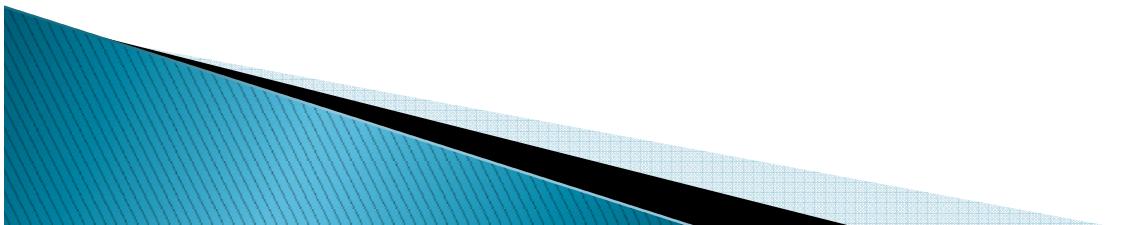
Now execute the following commands. This will ensure that you have all the necessary Veloce environment variables set each time you login. You will need to logout and login again for these to take effect.

**% echo ". /opt/veloce/env1415.sh" >> ~/.bash_profile
% logout**

Starting and Configuring a VNC server

USING THE VELOCE AT PSU: GOTCHAS

- ▶ Initial one-time setup: Logout wherever mentioned
- ▶ Check twice if your `.bash_profile` contains
“source /pkgs/mentor/veloce/env.sh”
- ▶ `.bash_profile` in `velocesolo` and not the MCECS host you are using



USING THE VELOCE AT PSU: GOTCHAS

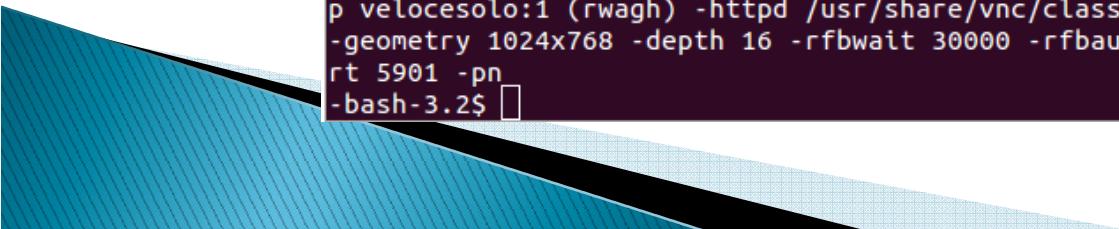
- ▶ Issue of multiple login instances
- ▶ Use: ps aux|grep vnc



```
Terminal
-bash-3.2$ ps aux|grep vnc
rwagh    16565  0.0  0.0  56948 17632 ?          S      2013   1:11 Xvnc :2 -deskto
p velocesolo:2 (rwagh) -httpd /usr/share/vnc/classes -auth /u/rwagh/.Xauthority
-geometry 1024x768 -depth 16 -rfbwait 30000 -rfbauth /u/rwagh/.vnc/passwd -rfbpo
rt 5902 -pn
jkerth   20858  0.0  0.0  58664 20660 ?          S     Mar31   0:18 Xvnc :3 -deskto
p velocesolo:3 (jkerth) -httpd /usr/share/vnc/classes -auth /u/jkerth/.Xauthorit
y -geometry 1280x1024 -depth 16 -rfbwait 30000 -rfbauth /u/jkerth/.vnc/passwd -r
fbport 5903 -pn
rwagh    20865  0.0  0.0  61220    780 pts/4      S+    18:21   0:00 grep vnc
rwagh    31478  0.0  0.0  52996 15516 ?          S      2013   1:16 Xvnc :1 -deskto
p velocesolo:1 (rwagh) -httpd /usr/share/vnc/classes -auth /u/rwagh/.Xauthority
-geometry 1024x768 -depth 16 -rfbwait 30000 -rfbauth /u/rwagh/.vnc/passwd -rfbpo
rt 5901 -pn
-bash-3.2$
```

USING THE VELOCE AT PSU: GOTCHAS

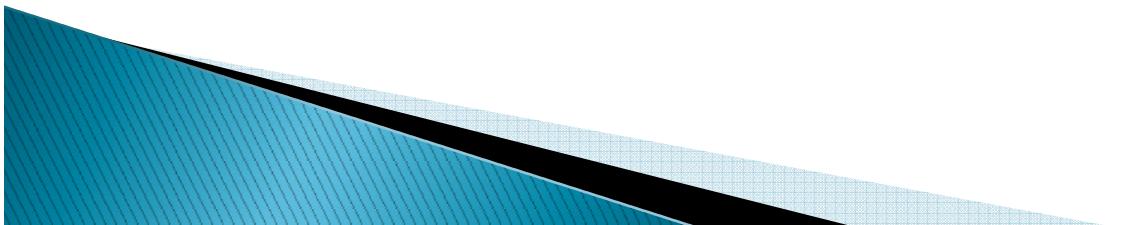
- ▶ Issue of multiple login instances
- ▶ Use: kill -port#



```
Terminal
p velocesolo:2 (rwagh) -httpd /usr/share/vnc/classes -auth /u/rwagh/.Xauthority
-geometry 1024x768 -depth 16 -rfbwait 30000 -rfbauth /u/rwagh/.vnc/passwd -rfbpo
rt 5902 -pn
jkerth 20858 0.0 0.0 58664 20660 ? S Mar31 0:18 Xvnc :3 -deskto
p velocesolo:3 (jkerth) -httpd /usr/share/vnc/classes -auth /u/jkerth/.Xauthorit
y -geometry 1280x1024 -depth 16 -rfbwait 30000 -rfbauth /u/jkerth/.vnc/passwd -r
fbport 5903 -pn
rwagh 20865 0.0 0.0 61220 780 pts/4 S+ 18:21 0:00 grep vnc
rwagh 31478 0.0 0.0 52996 15516 ? S 2013 1:16 Xvnc :1 -deskto
p velocesolo:1 (rwagh) -httpd /usr/share/vnc/classes -auth /u/rwagh/.Xauthority
-geometry 1024x768 -depth 16 -rfbwait 30000 -rfbauth /u/rwagh/.vnc/passwd -rfbpo
rt 5901 -pn
-bash-3.2$ kill -2 16565
-bash-3.2$ ps aux|grep vnc
jkerth 20858 0.0 0.0 58664 20660 ? S Mar31 0:18 Xvnc :3 -deskto
p velocesolo:3 (jkerth) -httpd /usr/share/vnc/classes -auth /u/jkerth/.Xauthorit
y -geometry 1280x1024 -depth 16 -rfbwait 30000 -rfbauth /u/jkerth/.vnc/passwd -r
fbport 5903 -pn
rwagh 20871 0.0 0.0 61216 764 pts/4 S+ 18:22 0:00 grep vnc
rwagh 31478 0.0 0.0 52996 15516 ? S 2013 1:16 Xvnc :1 -deskto
p velocesolo:1 (rwagh) -httpd /usr/share/vnc/classes -auth /u/rwagh/.Xauthority
-geometry 1024x768 -depth 16 -rfbwait 30000 -rfbauth /u/rwagh/.vnc/passwd -rfbpo
rt 5901 -pn
-bash-3.2$ 
```

REFERENCES

- ▶ www.mentor.com
- ▶ <http://www.deepchip.com/items/0522-01.html>
- ▶ Faust, M., “Using Mentor Veloce at PSU”, Aug 2010
- ▶ Chung, H., “Veloce Stand-Alone Mode Tutorial”, Sep 2010



THANK YOU!

► QUESTIONS??

