

The University of Texas at Dallas
Dept. of Electrical Engineering
EEDG/CE 6301: Advanced Digital Logic
EEDG/CE 6303: Testing and Testable Design
Instructor: Mehrdad Nourani

Basic Unix Commands & Tools

DISCLAIMER: *This document aims at your becoming familiar with basic UNIX commands. UNIX is a large and complicated operating system with many commands and features that are not explained here. What we explain here is the minimum to use a UNIX workstation. Nothing is guaranteed here. With the new versions of UNIX, there might be problems in this document and some commands may not work exactly as explained. Proceed cautiously and use it at your own risk. I encourage you to assign enough time to familiarize yourself with UNIX to be able to use it efficiently. Please report problems, corrections and suggestions about this document to nourani@utdallas.edu.*

File Management

ls	LiSt current directory.
ls -R	LiSt current directory and all subdirectories.
ls -l	List current directory with user permission/time/date.
ls *.c	List all files in the current directory that end in dot c.
ls ~/*.c	List all *.c files in the user's home directory.
ls -l cpu*.def	List all files that start with cpu and end in dot def.
ls -a	List all hidden files (files that begin with a dot).
man ls	Gives the all MANual information on the ls command.
more hello.c	Display the 'contents' of the file hello.c at 24 lines at a time. This prevents the text from scrolling by the screen. Press the space bar to proceed. Press q to quit.
man more	Gives the all the MANual information on the more command.
cp source dest	CoPy a file from source to destination. Example, copy aaa.c to bbb.c: cp aaa.c bbb.c
rm filename	ReMove a filename from the current working directory.
rm -v *	ReMove files selectively (verbose) with y or n (yes/no).
mv old new	Rename (MoVe) the old filename with a new filename

quota -v Display your disk space quota.

chmod a+x file CHange the access MODe for All users (User,Group,Others) to eXecutable.

chmod g+rw file CHange the access MODe for group to Read, Write, & eXecute.

chmod u-rw file CHange to NOT allow user to read or write.

chmod g=rx,o=rx directory Will make your directory public
 (g=r,o=r directory read only will not do it)

Directory Management

pwd Display your Present Working Directory

cd If lost then return to original login directory path

mkdir xyz MaKe a DIRectory xyz.

cd xyz Go down the directory to xyz (Change Directory).

cd .. Go relative up the directory path.

rmdir xyz ReMove DIRectory xyz.

cd /path Go absolute path addressing from the root directory.

Large scale file management commands (Finding text & files when you forget)

tar cvf <tarfile> <files|directories to be saved save>
 Tape ARchiver: copy all files/directories into a tarfile
 Example: "tar cvf zzz.tar *" creates a tar file called
 "zzz.tar" and then copies all files "*" into it.

tar xvf <tarfile> e(x)tract all files from the tarfile.

compress <file|tarfile> Compress a file or tarfile in order to save space.
 Example: "compress zzz.tar" will compress zzz.tar
 and rename it as "zzz.tar.Z".
 Example: "compress *" compress all files and
 append ".Z" to each file.

uncompress file.Z Uncompress the file.

zcat file.Z Read a compressed file to standard out
 zcat file.Z | more ..and pipe to more.

egrep "include" *.c Show the lines of all files (which end in dot c)
 which contain the string include.

egrep "(fopen|include)" *.c *.h Show the lines of all files (*.c & *.h only) which contain the string, fopen, or include.

find . -name "xyz*" Search current directory and all sub-directories for any filenames which begin with xyz.

vi text editor at a Glance

This is the traditional screen unix editor. If you intend to be a unix user this is a must to learn and use.

vilearn This program will teach you the vi editor

vi filename if the file exists, it will be opened
if the file does NOT exist, then a NEW file will be created

vi consists of three modes:

navigate mode: move around the file

insert mode: insert text into the file

command mode: execute shell commands and ed commands

Text input

a Append text AFTER the current text line.
To end the input text mode: press ESC key.

i Insert text BEFORE the current text line.
To end the input text mode: press ESC key.

line addressing after you start vi by DEFAULT you are in navigate mode

l --> Character Forward: move cursor right one char.

w Word forward: move cursor to next word

h <-- Character Backward: move cursor left one char.

b Word backward: move cursor to last word

x Character delete

0 Point to beginning of the line.

\$ Point to end of the line.

k Previous line

j Next line down

Ctrl-b Back full page

Ctrl-f Forward full page

Ctrl-d Back half page (down)

Ctrl-u	Forward half page (up)
:=.	Print current line number.
#G	Goto line number where # is line number
1G	Goto first line in the file
25G	Goto 25th line in the file
G	Goto last line in the file
/pattern	Search for a "pattern"
n	Find next pattern

Command mode (each command ends with a "return" or "enter" key)

:w	Save a file, and return key but do not exit vi.
:w filename	Save under a different filename
:r filename	Include a file in your current file.
:wq	Write file and Quit vi.
:x	eXit: same as :wq
:q!	Quit vi and DO NOT save the file

Printing & finding text

:set nu	Set line numbering on
:1,20p	Print lines 1 to 20 on the screen.
:1,\$p	Print lines 1 to the end of the file on the screen.
:.,.+20p	Print 20 lines from the current line.
:.=p	Print the current line number & Print it.
:g/xyz/.=p	Print all line numbers & text which contain the string xyz.

Deleting text

dd	Delete the current line
2dd	Delete 2 lines
3dd	Delete 3 lines
dw	Delete the current word
J	Join two lines: delete carriage return
u	Undo last command
:dp	Delete the current line & Print it.
:15,23dp	Delete lines 15 through 23.

Copying text

yy	Copy current line into clipboard
2yy	Copy 2 lines
p	paste clipboard into text (any time)

Substituting Text

:s/old/new/p Substitute the string 'old' with 'new' in the current line
 & Print the substituted line.
:2,10s/old/new/gp Global Substitute all strings 'old' with 'new' between
 lines 2 through 10 & Print them.

Using Unix Shell commands inside the editor

:!ls List files in the current directory.
:!pwd Present Working Directory.
:!rm file ReMove a file
:!cp src dest CoPy a file from Source to destination.
:!command Any Unix command.

Emacs text editor at a Glance - full screen editor

This is a full screen editor which is used under UNIX or Xwindows compared
to a line editor (ed) which only uses a line at a time.
(For remote users type: setenv TERM vt100)

Note 1: Ctrl-f (1 press) means press the control key simultaneously with
the lower case f key.
Esc v (2 press) means first press the escape key and then the
the lower case v key.

Note 2:)}] When ever a left parenthesis is typed, the cursor will temporarily
jump to the corresponding right parenthesis, to show the user
the matched set.

Ctrl-g Abort current command, use this whenever you are lost.
 It will not exit from the editor or hurt anything.
 This will be one of your most important commands.

Ctrl-f --> Character Forward: move cursor right one char.
Ctrl-b <-- Character Backward: move cursor left one char.
Ctrl-d Character delete
Ctrl-a Point to beginning of the line.
Ctrl-e Point to end of the line.
Ctrl-x = Print current line number.
Ctrl-x g Goto line number: <enter number>

Ctrl-p Previous line up
Ctrl-n Next line down

Esc v Back page

Ctrl-v	Next page
Ctrl-x Ctrl-s	Save a file, type name if new, else just return key.
Ctrl-x Ctrl-f	Find (load) a file, type filename after Ctrl-f.
Ctrl-x i	Include a file in your current file.
Ctrl-x Ctrl-c	Exit EMACS. EMACS will ask if you need a file to be saved.
Ctrl-space	Mark a position in the file.

Copying Text	Step
-----	----
Ctrl-space	1) START: Mark the starting point of what you want to move.
	2) END: Then move your cursor to the END of what you are moving.
Ctrl-w Ctrl-y	3) This will copy the marked text into a paste buffer.
	4) Now move your cursor to your destination.
Ctrl-y	5) COPY: This will YANK your text into current cursor position.

Moving Text	Step
-----	----
Ctrl-space	1) START: Mark the starting point of what you want to move.
	2) END: Then move your cursor to the END of what you are moving.
Ctrl-w	3) This will move the marked text into a paste buffer.
	4) Now move your cursor to your destination.
Ctrl-y	5) MOVE: This will YANK your text into current cursor position.

Deleting Text	Step
-----	----
Ctrl-space	1) START: Mark the starting point of what you want to move.
	2) END: Then move your cursor to the END of what you are moving.
Ctrl-w	3) DELETE: This will delete the marked text, and maintain a copy of it in the paste buffer.

Searching Text	Step
-----	----
Ctrl-s	1) This will start the search.
	2) As you type each character on the string you are searching for, Emacs will move the cursor to the new position.
	3) If you made a mistake just backspace or Ctrl-g to abort and start over.
Ctrl-s	4) If this string is not the one you are interested in and want to find another one just like it, then type Ctrl-s as many times to find the next instance.
Ctrl-r	5) (Optional) if you accidentally missed the string and want to reverse and look at the previous string match then type Ctrl-r as many times as you like.

Esc	6) If this is were you want to stop, then press the Escape key.
Ctrl-g	7) If you have not found what you are interested in and want to return to your original position, type Ctrl-g to abort the search.

Esc x overwrite-mode This turns off the auto-insert mode which is good for drawing stick pictures.

PRINTING

lpr -P<printer> file	PRint a file to a printer: Crawford building
	----- "lpr -Pljsol" for a printer called ljsol

lpq -P<printer>	Line Print Queue Status: shows all the print jobs on the printer. "lpq -Pljsol" for a printer called ljsol
-----------------	---

lprm -P<printer> <number>	ReMove print job from Line Printer.
---------------------------	-------------------------------------

print file	PRINT file to the Micro-VAX printer in the SMITH LAB.
------------	---

cat -n file lpr	Add LINE NUMBERS to each line in an ascii text file. Also, "cat -n file lpr -Pljsol".
-------------------	---

Spelling

look <word>	Unix command to check on how to spell a word, type spell and the word. The word does not have to be complete, it can even be the first letter. Spell will find the best matches.
-------------	--

ispell -t hello.tex	ispell is used to spell check .tex files. The speller stops at each unknown word and waits for an input command.
---------------------	--

command	{ explanation }
-----	-----
x	{ exit ispell }
?	{ help menu }
 <space>	 { accept word as correct this time only }

```

a          { accept word for rest of file }

i          { accept word as correct and insert in }
           { your private dictionary}

r          {replace the word}

l <word>    { look up a word in the dictionary,      }
           { type number IF you want to replace,    }
           { otherwise type another ispell command }

^l         { Control-L redraw screen }

!<unix command> { execute a unix command }
                { inside ispell }

```

X-window applications

xdvi sample.dvi will display a "latex sample.tex" file on the
graphics display.

xfig Graphics software to draw figures and import into
Latex files.

xmath Mathematica software.

xgif picture.gif will display a color or black & white "picture.gif".

xview view bit images.

To dump an image of an X window

In some other window, type:

```
xwd > <filename>
```

Now, click the rightmost mouse button on top of the
window you'd like to capture. You will hear three beeps.

The created file is in a special window dump format.

To convert this format to postscript, type:

```
xpr -device ps -output <filename>.ps -portrait -compact <filename>
```

This command has many other options, e.g. "-landscape" for landscape
mode instead of portrait mode, "-rv" for reverse video.

This postscript file can now be viewed using any postscript viewer, e.g.:

```
gs <filename>.ps
```

or printed:

```
lpr -Pljsol <filename>.ps
```

How to kill X-windows when you are locked up

- (1) Suppose your X-windows systems is locked up. Say machine, beryl.
- (2) login: First login on another workstation, say diamond, using the same login name & password
- (3) rsh beryl Remote login to the machine where you had trouble
- (4) ps This displays all the processes running under your login account on the workstation beryl.
- (5) kill -9 <process> Look at the numbers from the "ps" and start killing all the processes until the X-windows shuts down. Also check ps. Be careful not to kill yourself.
- (6) ^D logout, of the remote shell & logout of diamond.
- (7) ^D go to the first workstation, beryl, and logout there.

Common file types

- | | |
|-------|---|
| *.c | C Language file: use the unix "gcc", "g++", or "cc" command. |
| *.dvi | Device Independent file: generated by the "latex" or "tex" program. |
| *.gif | Graphics file: use the "xgif" file. |
| *.ps | Postscript file |
| *.pl | prolog file |
| *.tar | Tape Archived file: unix "tar" command. |
| *.tex | Tex file: used by "latex" and "tex" |

```

*.z          Compressed file by the unix "pack" command. Use "unpack" to
             restore.

*.zip        Compressed file. Use "unzip" to decompress.

*.Z          Compressed file by the unix "compress" command.
             use "uncompress *.Z" to restore.

```

file transfer notes

Using the Unix on-line Manual

```

man <unix_command>    { This command will give details on any Unix command }
man man               { This explains about the manual itself }

```

The manual is for more details of the following topics to be discussed here.

```

man mail              { The manual on electronic mail }
man uuencode          { The manual on encoding/decoding binary files }
man ftp               { The manual on File Transfer Program }
man compress          { The manual for compressing/uncompressing files }
man tar               { The manual for archived files }
man telnet            { The manual on Telnet }
man script            { The manual on saving text going to the display }

```

Printing the manual to the printer

```

man uuencode > zzztemp { This will copy the manual information of }
                        { "uuencode" to a file for example "zzztemp" }
lpr zzztemp            { This will print the file zzztemp }
rm zzztemp             { This will remove the file zzztemp }

```

ftp: file transfer program

This program allows a user to connect to computers around the world and transfer files between computers.

To get a file from a remote machine

```

ftp
open info.umd.edu      { open a connection to the University of }
                        { Maryland. }
login: anonymous        { login: is display once a connection is }
                        { made. The user types "anonymous". }

```

password: anonymous	{ Depending on the message from the remote }	
	{ system: type "anonymous" or "ident" }	
ls	{ This will list the directory of the }	
	{ of the remote system. }	
cd /pub	{ This will change the remote directory }	
	{ to "/pub" }	
get zzz	{ This will copy exactly one file from the }	
	{ remote system called "zzz" into your }	
	{ local directory with the same name. }	
quit	{ This will close the connection and exit ftp }	

Other ftp commands of interest

ls -l	{ list long format of remote directory }
ls p*	{ list all programs beginning with "p" }
mget *	{ Multiple Get: will copy more than one }
	{ file(s) to your local system. }
mget *.c	{ Multiple Get all files ending in ".c" }
lcd	{ Local Change Directory (i.e. your system) }
!<command>	{ execute local commands: }
	{ "!cd" does not work, always use "lcd" }
?	{ list all ftp commands }