Cheat sheet

Linux commands

The terminal commands ECE students use most: files, searching, permissions, processes, remote machines and talking to boards.

1 Files and folders

List, with hidden files	`ls -la`
Move around	`cd ~/projects  cd ..  pwd`
Make folders	`mkdir -p src/rtl/tb`

-p creates parents too

Copy, move, delete	`cp -r src backup  mv a.v b.v` `rm file`

rm -r deletes a folder for good: there is no bin

Find files	`find . -name "*.v"`
Disk use	`du -sh *  df -h`

2 Reading files

Whole file, or paged	`cat file.txt  less file.txt`

In less: / to search, q to quit

Start or end	`head -n 20 log.txt  tail -n` `20 log.txt`
Follow a growing log	`tail -f sim.log`

Ctrl+C to stop

Count lines	`wc -l *.v`

3 Searching

Text in files	`grep -rn "always" src/`

-r folders, -n line numbers, -i ignore case

Filter output	`ps aux | grep python`
Count matches	`grep -c "ERROR" run.log`

4 Pipes and redirection

Write, append	`make > build.log  echo hi >>` `notes.txt`
Errors too	`make > build.log 2>&1`
Chain commands	`grep VIOLATED timing.rpt |` `sort | uniq -c`
Run if the last one worked	`make && ./sim`

5 Permissions

Make a script runnable	`chmod +x run.sh  ./run.sh`
Numeric modes	`chmod 644 file  chmod 755` `dir`

6 = read+write, 4 = read, 7 = all

Run as administrator	`sudo apt install gtkwave`

6 Processes

What is running	`ps aux  top  htop`
Stop a process	`kill 1234  kill -9 1234`

Try plain kill first

Run in background	`long_sim &  jobs  fg %1`
Keep running after logout	`nohup ./run.sh > out.log 2>&1` `&`

7 Remote machines

Log in	`ssh user@192.168.1.20`

Copy files	`scp design.v user@host:~/work/` `scp -r user@host:~/results .`
Download	`curl -LO https://example.com/file.zip`
Archives	`tar -czf work.tar.gz work/` `tar -xzf work.tar.gz`

8 Boards and hardware

USB devices	`lsusb`
What just connected	`sudo dmesg | tail`
Serial ports	`ls /dev/ttyUSB* /dev/ttyACM*`
Serial terminal	`screen /dev/ttyUSB0 115200`

Quit: Ctrl+A, then K

Serial port access	`sudo usermod -aG dialout $USER`

Log out and back in afterwards

I²C devices (Raspberry Pi)	`i2cdetect -y 1`