



Digital IC Design

Week2: Basic Linux Operation needed for IC EDA Solution Operations

2024 | Nineplus Infotech X Anseong Polytechnic University
Sr Research Engineer, William Woo



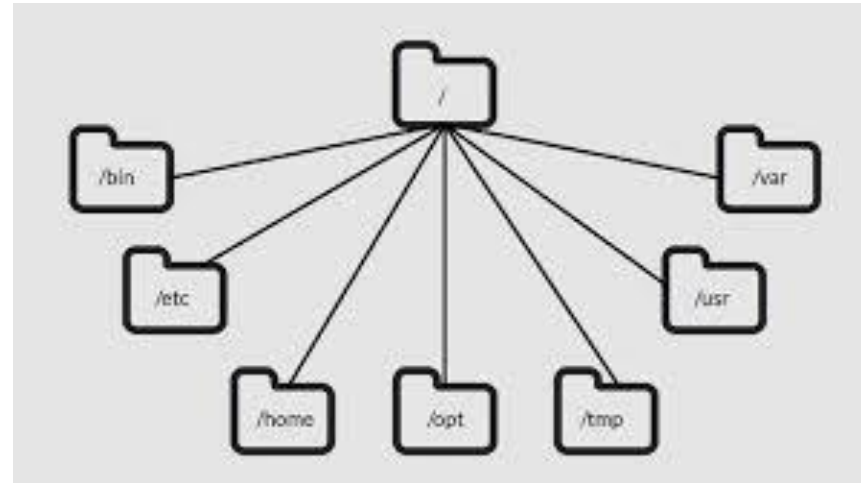
1. Basic Linux Operation

Directory



```
cd {directory}  
pwd
```

- Use Absolute PATH
- Use Relative PATH
- Use user's home



Manual



man {command}

- Search Keyword
- Moving Back & Forth
- Quit Manual

```
LSOF(8)                                System Manager's Manual                                LSOF(8)

NAME
    lsof - list open files

SYNOPSIS
    lsof [ -?abChlnNOPRtUvVX ] [ -A A ] [ -c c ] [ +c c ] [ +|-d d ] [
    +|-D D ] [ +|-e e ] [ +|-f [cfgGn] ] [ -F [fl] ] [ -g [sl] ] [ -i [il]
    ] [ -k k ] [ -K k ] [ +|-L [ll] ] [ +|-m m ] [ +|-M ] [ -o [ol] ] [ -p
    s ] [ +|-r [t[m<fmt>]] ] [ -s [p:sl] ] [ -S [tl] ] [ -T [tl] ] [ -u s ]
    [ +|-w ] [ -x [fil] ] [ -z [zl] ] [ -Z [Zl] ] [ -- ] [names]

DESCRIPTION
    Lsof revision 4.87 lists on its standard output file information
    about files opened by processes for the following UNIX dialects:

        Apple Darwin 9 and Mac OS X 10.[567]
        FreeBSD 4.9 and 6.4 for x86-based systems
        FreeBSD 8.2, 9.0 and 10.0 for AMD64-based systems
        Linux 2.1.72 and above for x86-based systems
        Solaris 9, 10 and 11

    (See the DISTRIBUTION section of this manual page for information on
    how to obtain the latest lsof revision.)

    An open file may be a regular file, a directory, a block special
    file, a character special file, an executing text reference, a
    library, a stream or a network file (Internet socket, NFS file or
    UNIX domain socket.) A specific file or all the files in a file
    system may be selected by path.

    Instead of a formatted display, lsof will produce output that can be
    parsed by other programs. See the -F, option description, and the
OUTPUT FOR OTHER PROGRAMS section for more information.

    In addition to producing a single output list, lsof will run in
    repeat mode. In repeat mode it will produce output, delay, then
    repeat the output operation until stopped with an interrupt or quit
    signal. See the +|-r [t[m<fmt>]] option description for more infor-
    mation.

OPTIONS
    In the absence of any options, lsof lists all open files belonging
    to all active processes.

Manual page lsof(8) line 1 (press h for help or q to quit)
```

Clear the screen



clear

Listing Directory



ls -option {directory}

- Listing Directory
- Read Output
- Options
 - Hidden File / Directory
 - Verbose

```
dr-xr-xr-x. 17 root root 4096 Aug 13 10:31 .
dr-xr-xr-x. 17 root root 4096 Aug 13 10:31 ..
-rw-r--r--. 1 root root 439505 Aug 9 16:36 .readahead
lrwxrwxrwx. 1 root root 7 Aug 9 16:20 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Aug 10 13:32 boot
drwxr-xr-x. 21 root root 3560 Aug 9 16:34 dev
drwxr-xr-x. 148 root root 8192 Aug 17 02:01 etc
drwxr-xr-x. 5 root root 66 Aug 16 00:12 home
lrwxrwxrwx. 1 root root 7 Aug 9 16:20 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Aug 9 16:20 lib64 -> usr/lib64
drwxr-xr-x. 2 root root 6 Dec 15 2017 media
drwxr-xr-x. 2 root root 6 Dec 15 2017 mnt
drwxr-xr-x. 4 root root 41 Aug 9 16:56 opt
dr-xr-xr-x. 866 root root 0 Aug 9 16:34 proc
dr-xr-xr-x. 18 root root 4096 Aug 17 02:01 root
drwxr-xr-x. 45 root root 1360 Aug 18 06:54 run
lrwxrwxrwx. 1 root root 8 Aug 9 16:20 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Dec 15 2017 srv
dr-xr-xr-x. 13 root root 0 Aug 9 16:34 sys
drwxrwxrwt. 29 root root 20480 Aug 18 16:01 tmp
drwxr-xr-x. 13 root root 207 Aug 9 16:20 usr
drwxr-xr-x. 20 root root 4096 Aug 9 16:34 var
```

Init File[s] / Directorie[s]



```
touch {file_name}
```

- Make multiple files



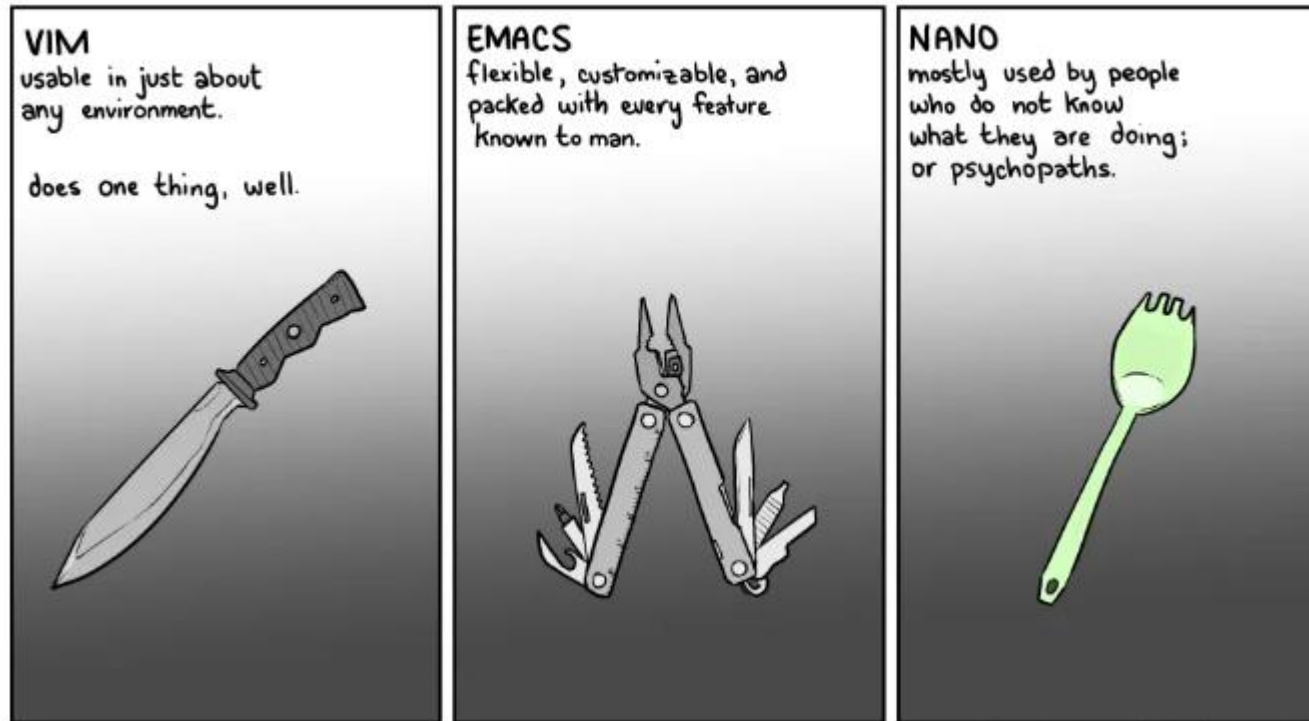
```
mkdir -option {directory_name}
```

- Nested directory

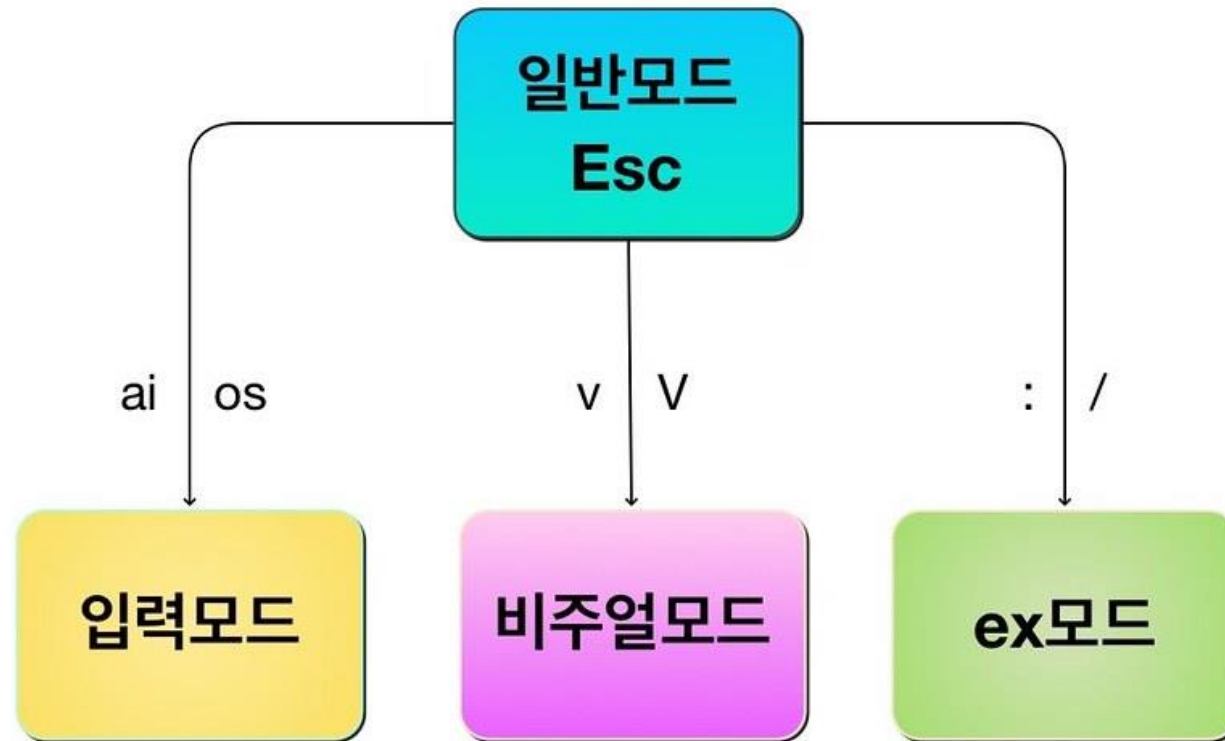
VI(M)



Unix/Linux 자체에 내장된 문서 편집기 프로그램.
강력한 문서 편집과 Customization을 통한 사용확장성을 보여준다.



VI(M): Operation Mode



VI(M): insert mode



입력모드는 텍스트를 입력(편집)하는 모드이다

- a : 커서위치 다음 칸에 입력
- i : 커서의 위치에 입력
- o : 커서의 다음 행에 입력
- s : 커서 위치의 한글자를 삭제 후
입력

VI(M): visual mode



비주얼 모드는 블록 단위를 선택하여 작업하는 모드이다.

- v : Normal Visual Mode
- V : Visual Line Mode
- d: 선택된 내용 삭제 -> Normal Mode
- c: 선택된 내용 삭제 -> Insert Mode
- y: 선택된 내용의 복사 -> Normal Mode
- J: 선택된 모든 라인을 순서대로 결합 -> Normal Mode
- ~ : 선택된 내용의 case를 변경 -> Normal Mode
- u : 선택된 내용을 소문자로 변경 -> Normal Mode
- U : 선택된 내용을 대문자로 변경 -> Normal Mode

VI(M): ex mode



ex 모드는 작업을 명령하는 모드이다

- :w - 저장
- :w file.txt – file.txt로 저장
- :w >> file.txt – file.txt에 append하여 저장
- :q – VIM 종료
- :q! – 강제 종료
- :wq! – 저장하고 강제 종료
- :e file.txt – file.txt 를 불러옴

VI(M) 단축키

version 1.1
April 1st, 06

vi / vim 단축키 모음

Esc
명령 모드

~ 대소문자 전환	! 외부 명령	@ 매크로 실행	# 이전 검색	\$ 줄 끝으로 이동	% 일치하는 것끼리	^ 줄의 첫 문자	& 줄 반복	* 다음 줄로	(문장 시작) 문장 끝	_ 아래 줄로 이동	+ 다음 줄
· 매크로 이동	1	2	3	4	5	6	7	8	9	0 줄의 처음	- 이전 줄	= 자동 들여쓰기
Q 실행 모드	W 다음 WORD	E WORD 끝	R 수정 모드	T 뒤로 검색	Y 줄단위 복사	U 줄 단위 실행 취소	I 줄 시작에서 삽입	O 행 위에 삽입	P 커서 이전에 붙여넣기	{ 문단 시작	}	문단 끝
q 매크로 기록	w 다음 단어	e 단어 끝	r 한 문자 교체	t 한 문자 검색	y 복사	u 실행 취소	i 편집 모드	o 행 아래에 삽입	P 커서 이후에 붙여넣기	[기타]	기타
A 줄 끝에 덧붙이기	S 줄 삭제 후 편집 모드	D 줄 끝까지 삭제	F 뒤로 검색	G 파일 끝으로 이동	H 화면 상단	J 줄 합치기	K 다음 줄	L 화면 하단	: 명령줄	" 레지스터 지정	열 이동	
a 매크로 이동	s 단어 삭제 후 편집 모드	d 삭제	f 한 문자 찾기	g 확장 명령	h ←	j ↓	k ↑	l →	· /T/H/F 명령 반복	' 매크로 이동	\ 사용 안함	
Z 종료	X 백스페이스	C 줄 끝까지 바꾸기	V 줄단위 비주얼 모드	B 이전 WORD	N 이전 (찾기)	M 화면 가운데	< 들여쓰기	> 들여쓰기	? 찾기 (뒤로)			
Z 확장 명령	X 글자 삭제	c 바꾸기	v 비주얼 모드	b 이전 단어	n 다음 (찾기)	m 마크 설정	< 역순 검색	· 명령 반복	/ 찾기			

동작 커서를 이동하거나, 연산자가 동작할 범위를 지정합니다.

명령 바로 동작하는 명령.
빨간색은 편집 모드로 변경됩니다.

연산자 이동 관련 문자(숫자나 커서 이동)와 함께 사용하여 하며, 커서의 위치부터 목적지까지 연산합니다.

확장 특별한 키 합수로, 추가적인 키 입력이 필요합니다.

q· 입력후 (숫자를 제외 한 으로 끝낼수 있는) 글자를 입력하여야 합니다.

words: 구분자로 공백, 특수기호 모두 사용
WORDS: 구분자로 공백 문자만 사용

words: quux(foo, bar, baz);
WORDS: quux(foo, bar, baz);

주요 명령행 명령 ('ex'):
:w (저장), :q (종료), :q! (저장하지 않고 종료)
:e f (파일 f 열기),
:%s/x/y/g (파일 전체에서 'x'를 'y'로 교체),
:h (vim 도움말), :new (새 파일)

그외 중요한 명령들:
CTRL-R: 재실행 (vim),
CTRL-F/-B: 페이지 위로/아래로,
CTRL-E/-Y: 줄 스크롤 위로/아래로,
CTRL-V: 블록-비주얼 모드 (vim 전용)

비주얼 모드:
커서를 움직여 지정한 범위에 연산자를 적용합니다. (vim 전용)

참고:

- (1) 복사/붙여넣기/지우기 명령어를 사용하기 전에 "x"를 입력하여 레지스터(클립보드)를 지정하세요. (x는 a에서 z 또는 "을 사용할 수 있음) (예: "ay\$를 입력하면 현재 커서에서 라인 끝까지의 내용을 레지스터 'a'에 저장합니다.)
- (2) 어떤 명령을 입력하기 전에 횡수를 지정하면, 횡수만큼 반복하게 됩니다. (예: 2p, d2w, 5i, d4j)
- (3) 연속으로 입력하는 명령은 현재의 라인에 반영됩니다. 예시: dd(현재 라인 지우기), >>(들여쓰기)
- (4) ZZ는 저장후 종료, ZQ는 저장하지 않고 종료.
- (5) zt: 커서가 위치한 곳을 제일위로 올리기, zb: 바닥으로, zz: 가운데로
- (6) gg: 파일의 처음으로 (Vim 전용), gf: 커서가 위치한 곳의 파일 열기 (Vim 전용)

• VIM 연습

Copy and Paste



```
cp -option {src} {dst}  
rm -option {src} {dst}
```

- Directory
- Recursive Operation

Duplicity of “mv” command



`mv [src] [dst]`
`mv [src1] [src2]`

- Moving directory
- Change directory's name

```
dr-xr-xr-x. 17 root root 4096 Aug 13 10:31 .
dr-xr-xr-x. 17 root root 4096 Aug 13 10:31 ..
-rw-r--r--. 1 root root 439505 Aug 9 16:36 .readahead
lrwxrwxrwx. 1 root root 7 Aug 9 16:20 bin -> usr/bin
dr-xr-xr-x. 4 root root 4096 Aug 10 13:32 boot
drwxr-xr-x. 21 root root 3560 Aug 9 16:34 dev
drwxr-xr-x. 148 root root 8192 Aug 17 02:01 etc
drwxr-xr-x. 5 root root 66 Aug 16 00:12 home
lrwxrwxrwx. 1 root root 7 Aug 9 16:20 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Aug 9 16:20 lib64 -> usr/lib64
drwxr-xr-x. 2 root root 6 Dec 15 2017 media
drwxr-xr-x. 2 root root 6 Dec 15 2017 mnt
drwxr-xr-x. 4 root root 41 Aug 9 16:56 opt
dr-xr-xr-x. 866 root root 0 Aug 9 16:34 proc
dr-xr-xr-x. 18 root root 4096 Aug 17 02:01 root
drwxr-xr-x. 45 root root 1360 Aug 18 06:54 run
lrwxrwxrwx. 1 root root 8 Aug 9 16:20 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Dec 15 2017 srv
dr-xr-xr-x. 13 root root 0 Aug 9 16:34 sys
drwxrwxrwt. 29 root root 20480 Aug 18 16:01 tmp
drwxr-xr-x. 13 root root 207 Aug 9 16:20 usr
drwxr-xr-x. 20 root root 4096 Aug 9 16:34 var
```

Printing the contents in the file



```
cat {file}  
more / less {file}
```

- Quitting printed page
- Move back and forth through pages



```
head {file}  
tail {file}
```




2. Intermediate Linux Operation

Shell(1)



Shell은 리눅스 커널과 사용자를 연결해 주는 인터페이스 명령어(프로그래밍 언어)이다



echo \$0
: 현재 사용 중인 사용자의 Shell을 출력한다

- Bash Shell
- C-Shell for EDA

Shell(2)



`source {shell_file}`
: Shell 파일의 내용을 적용한다

- Shell 적용의 우선순위

Alias



```
alias {shorten_customized_cmd} = "original_long_cmd"
```

- 단축 명령어를 설정하는 방법
- 즐겨찾기

Link



- Hard Link: `ln [src] [dst]`
 - Soft Link: `ln -s [src] [dst]`
-
- 복사와 링크의 다른점
 - 원본 파일 변경/삭제
 - Hard Link보다 Soft(Symbolic) Link가 선호되는 이유

grep



- `grep -option {search_pattern} {file or path}`

Compress using tar



- `tar -xzvf {compressed_file}`
- `tar -czvf {compressed file} {src_directory}`

- Process of Compressing
- Extension name of compressed file

Logical Concatenation of commands



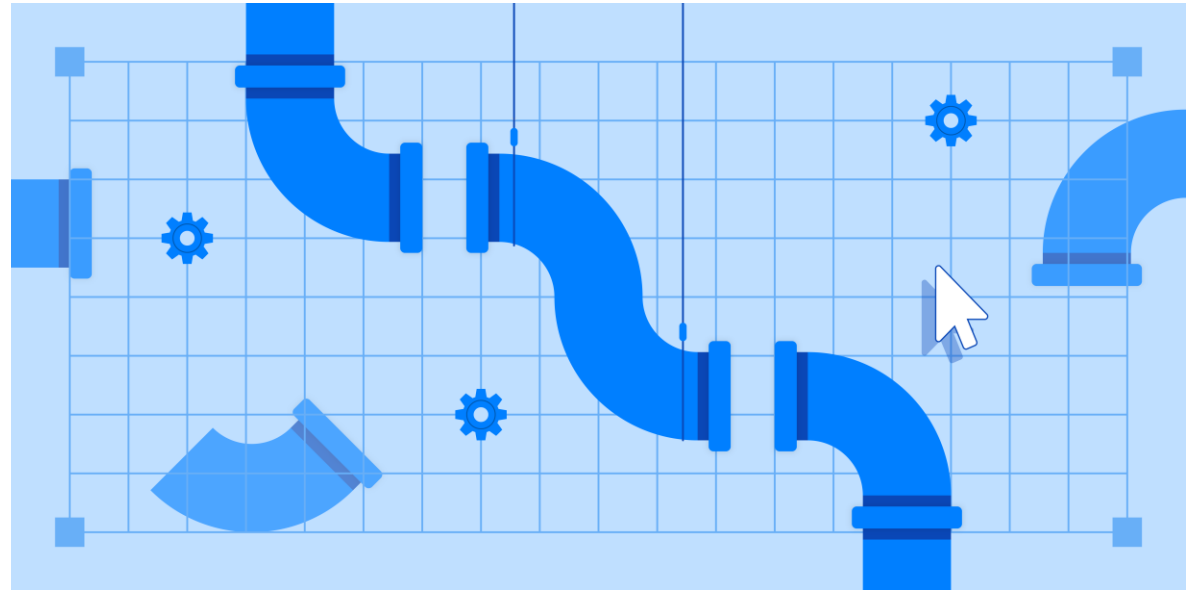
- `command1 ; command2`
- `command1 && command2`
- `command1 || command2`

Command Pipelining



command1 | command2

- Pipeline 개념의 공학적 응용



Redirection IO



command > file
command >> file

- Standard Input / Output

Control Process



ps : prints user's whole process
kill : terminate user's PID

- Process란?
- PID(Process ID)

Permission



rwX: read, write, execution
ugo: user, group, other[s]



`chmod -option permission_masking_bits {file or directory}`

- Octal 사용법
- +/- rwx 사용법
- Recursive Operation

LAB

1. Configure user account
2. Practice Command
3. Vi Editor: VIM 연습

Homework

- Chapter3: Gate Level Modeling(Preview)