

Tinker board

IoT를 위한 최선의 엣지 단말기

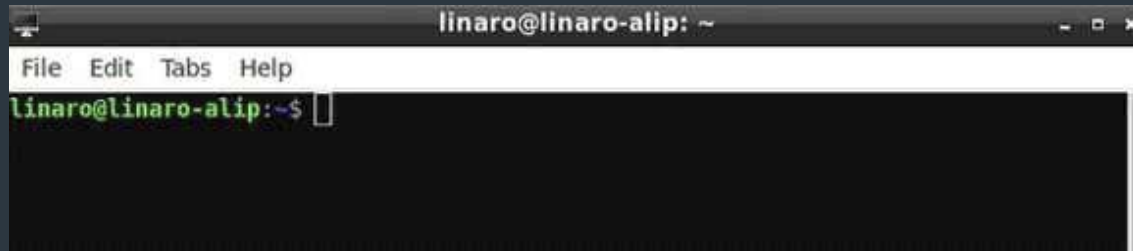
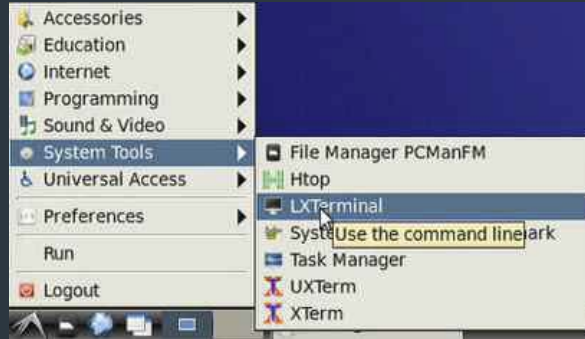


MAKER SPACE
G·CAMP

Contents

- Debian 패키지 갱신 및 기본 편집기 설치
- Swapfile
- SSH
- VNC

Debian 패키지 갱신 및 기본 편집기 설치



```
ldconfig: file /usr/lib/mali/libmali.so is truncated
ldconfig: file /usr/lib/mali/libmali.so.1.9.0 is truncated
ldconfig: file /usr/lib/mali/libmali.so.1 is truncated
ldconfig: file /usr/lib/mali/libmali-midgard-t86x-rl8p0-xml.so is truncated
ldconfig: file /usr/lib/libmali.so.1 is truncated
```

- 다음과 같이 하단 맨 좌측에 있는 아이콘을 눌러 [System Tools]-- [LXTerminal] 프로그램을 실행
- `$ sudo apt update`
- `$ sudo apt upgrade -y`
- `$ sudo apt install nano -y`
- 두 번째 명령 수행 후에는 다음과 같은 메시지가 뜨는데 무시

기본 nano 명령어

- 몇 가지 주요한 nano 명령어, 기억하여 python 파일 편집 시 활용

nano 명령	동작
Alt + #	줄 번호 표시
Ctrl + _	줄 이동
Ctrl + y	맨 처음 줄로 이동
Ctrl + v	맨 마지막 줄로 이동
Alt + t	커서 위치부터 파일 끝까지 지우기
Ctrl + k	한 줄 지우기
Alt + 6	한 줄 복사하기

Swap file 생성

```
linaro@linaro-alip:~$ free -h
```

	total	used	free	shared	buff/cache	available
Mem:	1.9Gi	255Mi	824Mi	127Mi	878Mi	1.5Gi
Swap:	0B	0B	0B			

- 메모리가 부족한 경우 메모리의 일부 내용을 디스크로 스왑(Swap)
- Windows의 가상 메모리와 같이 디스크의 일부를 메모리처럼 사용
- \$ free -h
- 현재 스왑 공간이 없음 확인

Swap file 생성

- 스왑 패키지를 설치
`sudo apt install dphys-swapfile -y`
- nano 편집기를 이용하여 [/etc/dphys-swapfile] 파일 Open
`sudo nano /etc/dphys-swapfile`
- 아래 그림의 왼쪽 부분을 찾아 오른쪽과 같이 수정



```
#CONF_SWAPSIZE= CONF_SWAPSIZE=10240
```

- [ctrl+x], y, 엔터키를 차례대로 눌러 파일 저장
- `$ sudo reboot` 으로 재부팅

Swap file 생성

```
linaro@linaro-alip:~$ free -h
```

	total	used	free	shared	buff/cache	available
Mem:	1.9Gi	225Mi	1.4Gi	107Mi	340Mi	1.6Gi
Swap:	2.0Gi	0B	2.0Gi			

- \$ free -h
[스왑 공간]이 생성된 것을 확인

Python 실습 환경 구성

- Tinker Board Debian OS의 python 명령은 기본 상태에서 Python 2.7 버전으로 설정되어 있어 주로 사용 되는 Python 3.7 버전으로 변경 필요
- \$ python --version

```
linaro@linaro-alip:~$ python --version
Python 2.7.16
```

- \$ cd /usr/bin
- \$ sudo rm python
- \$ sudo ln -s python3 python

```
linaro@linaro-alip:/usr/bin$ python --version
Python 3.7.3
```




SSH

- SSH는 Secure SHell의 줄임말로 보안이 강화된 서버 원격 접속 및 제어 도구
기존 리눅스 사용자(Clinet)는 물리적으로 떨어진 리눅스 서버를 사용하기 위해서 원격접속 도구인 텔넷(Telnet)을 사용
- 텔넷의 치명적인 단점은 텔넷 클라이언트를 통해 서버를 조작할 때 주고 받는 데이터가 암호화 되지 않음
- SSH는 텔넷과 마찬가지로 원격지에서 서버에 접속하고 관리할 수 있는 도구
이지만 통신을 할때 암호화된 상태로 데이터를 주고 받으므로 보안에 더욱 뛰어나

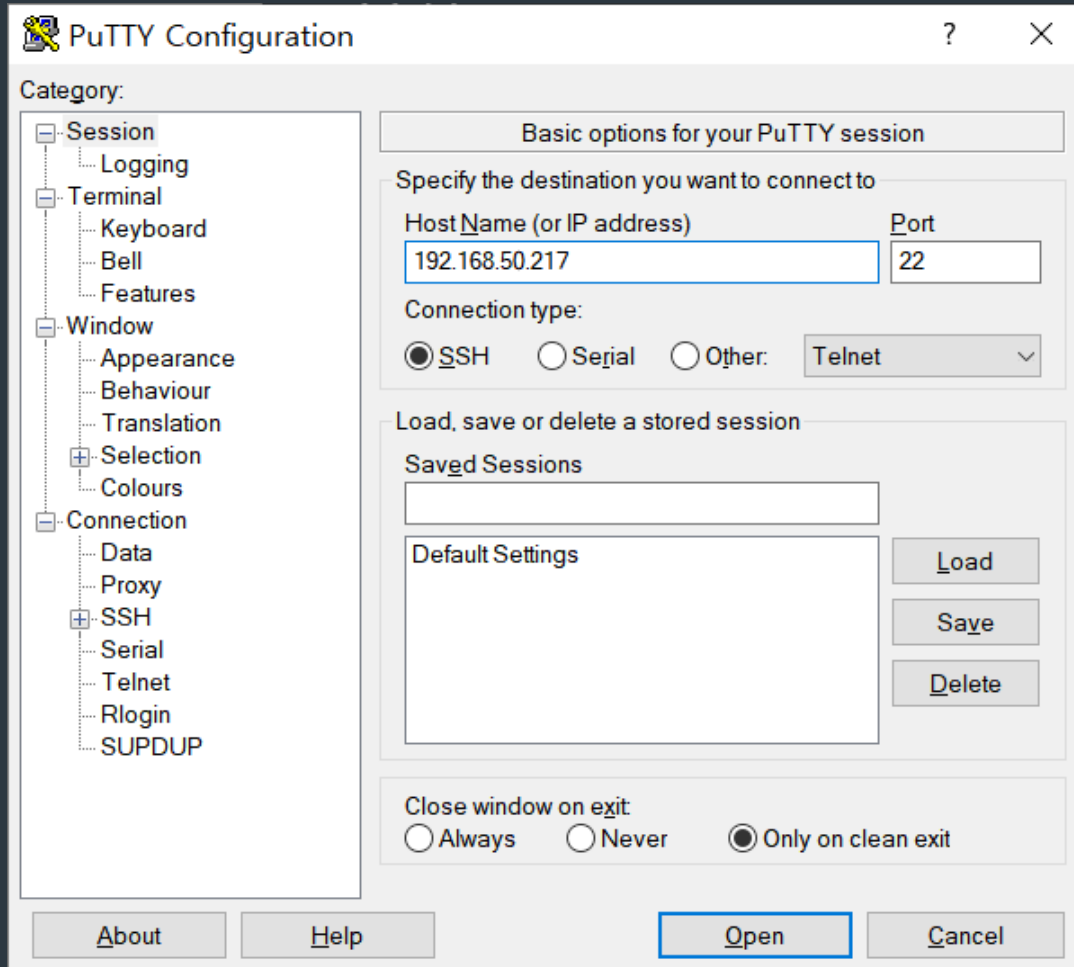
SSH



- Tinker Board 2에는 기본적으로 SSH가 설치되어 있음
- Tinker Board에서 IP 주소를 다음과 같이 확인
- WiFi 아이콘 상에서 마우스 오른쪽 버튼을 누르고,
- [Connection Information] 메뉴를 선택한 후
- Tinker Board 의 IP 주소를 확인

SSH

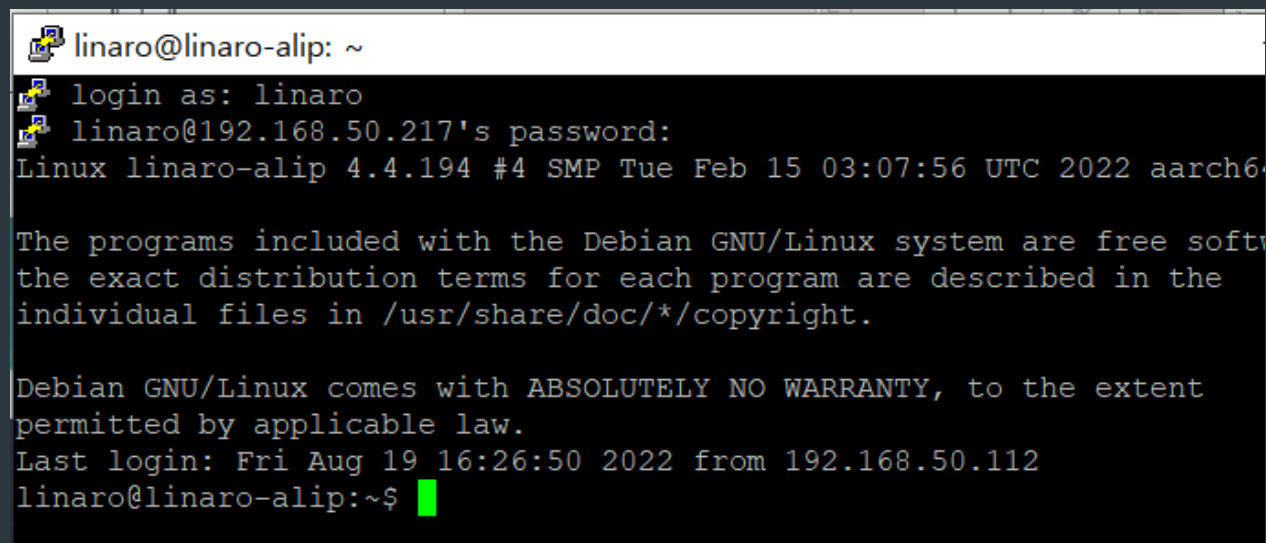
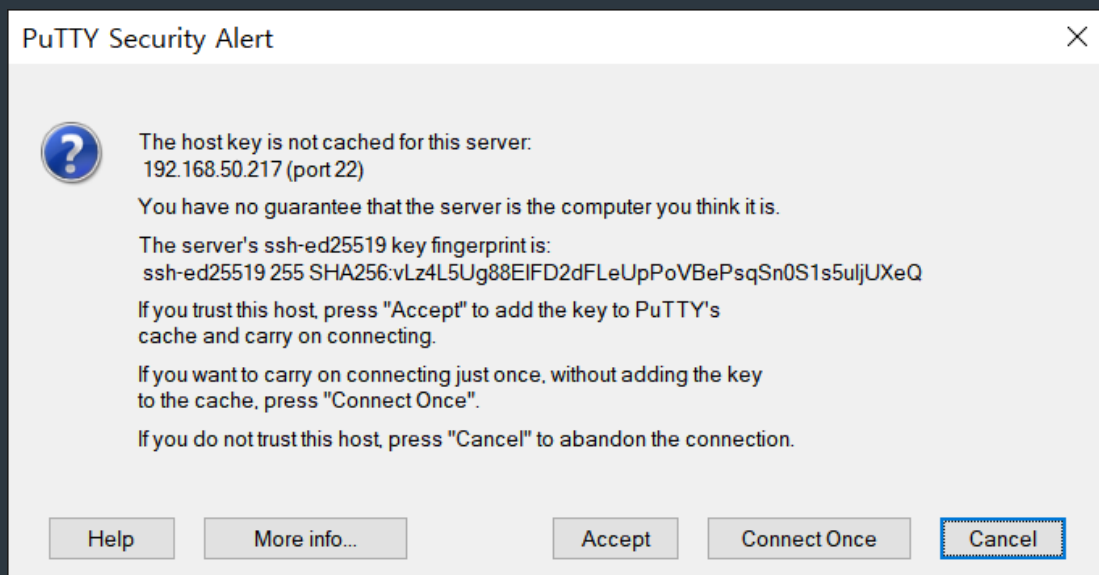
<putty>



- [Putty](#) 다운 후 실행
- IP 입력 및 22 Port, SSH 설정
- 보안 지문 Accept
- ID / PW 입력 후 접속
- ID -> linaro
- PW -> linaro
- Putty 연결 후 HDMI 및 USB 연결 해제 가능

SSH

<putty>



VNC

- VNC란 Virture Network Computing의 약자로 가상 네트워크 컴퓨팅이란 뜻
- Tinker Board나 라즈베리파이는 기본적으로 VNC가 설치되어 있지만 Tinker Edge R은 없어 설치 및 세팅을 해야함
- Tinker 및 라즈베리파이는 GUI 데스크톱으로 LXDE "Lightweight X11 Desktop Environment " 를 사용
- LXDE와는 RealVNC 및 x11vnc가 궁합이 좋음
- X11vnc를 통한 VNC 세팅 강의

VNC

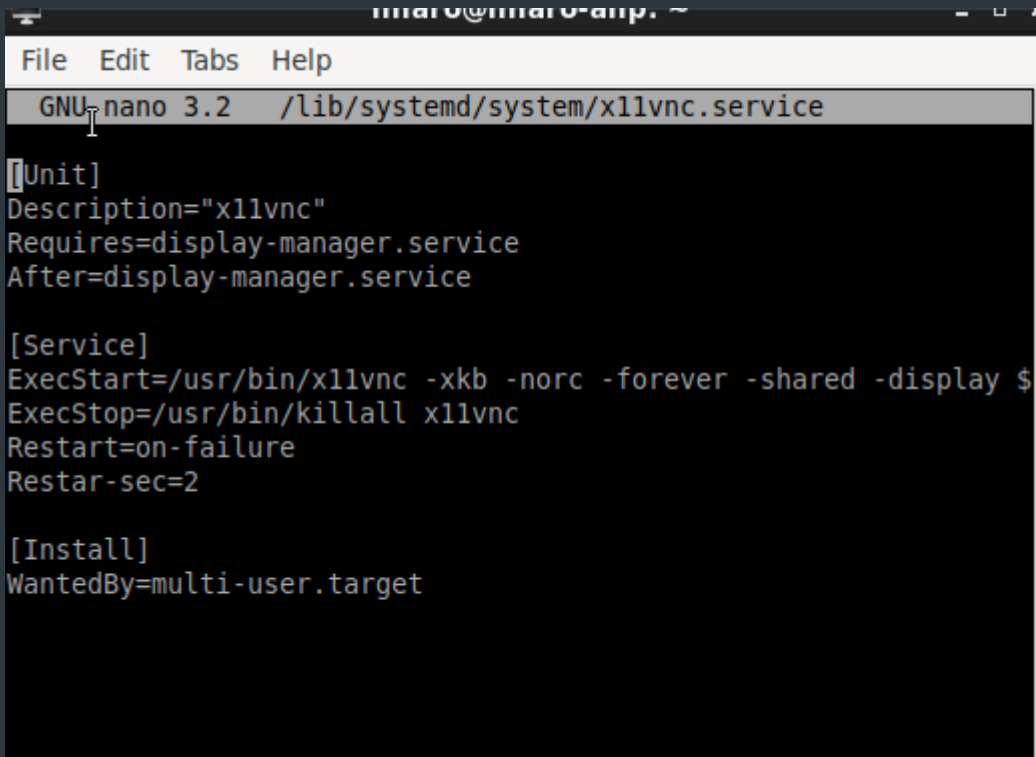
<x11vnc 설치>

```
linaro@linaro-alip:~$ sudo apt-get update
Get:1 http://security.debian.org buster/updates InRelease [34.8
Get:2 http://security.debian.org buster/updates/main Sources [24
Get:3 http://security.debian.org buster/updates/main arm64 Packa
Get:4 http://security.debian.org buster/updates/main Translation
Hit:5 http://cdn-fastly.deb.debian.org/debian buster InRelease
Get:6 http://cdn-fastly.deb.debian.org/debian buster-updates InF
Get:7 http://cdn-fastly.deb.debian.org/debian buster-updates/mai
Get:8 http://cdn-fastly.deb.debian.org/debian buster-updates/mai
kB]
Get:9 http://cdn-fastly.deb.debian.org/debian buster-updates/mai
[397 B]
Get:9 http://cdn-fastly.deb.debian.org/debian buster-updates/mai
[397 B]
Get:10 http://cdn-fastly.deb.debian.org/debian buster-updates/ma
35.pdiff [283 B]
Get:10 http://cdn-fastly.deb.debian.org/debian buster-updates/ma
35.pdiff [283 B]
Fetched 882 kB in 9s (99.4 kB/s)
Reading package lists... Done
linaro@linaro-alip:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

- `sudo apt-get update`
- `sudo apt upgrade`
- `sudo apt-get install x11vnc`

VNC

<x11vnc daemon 서비스 등록>



```
File Edit Tabs Help
GNU nano 3.2 /lib/systemd/system/x11vnc.service

[Unit]
Description="x11vnc"
Requires=display-manager.service
After=display-manager.service

[Service]
ExecStart=/usr/bin/x11vnc -xkb -norc -forever -shared -display $
ExecStop=/usr/bin/killall x11vnc
Restart=on-failure
RestartSec=2

[Install]
WantedBy=multi-user.target
```

- `sudo nano`
`/lib/systemd/system/x11vnc.service`
- [x11vnc.service](#) 파일 작성
- `sudo systemctl daemon-reload`
- `sudo systemctl start x11vnc.service`
- `sudo systemctl enable`
`x11vnc.service`

VNC

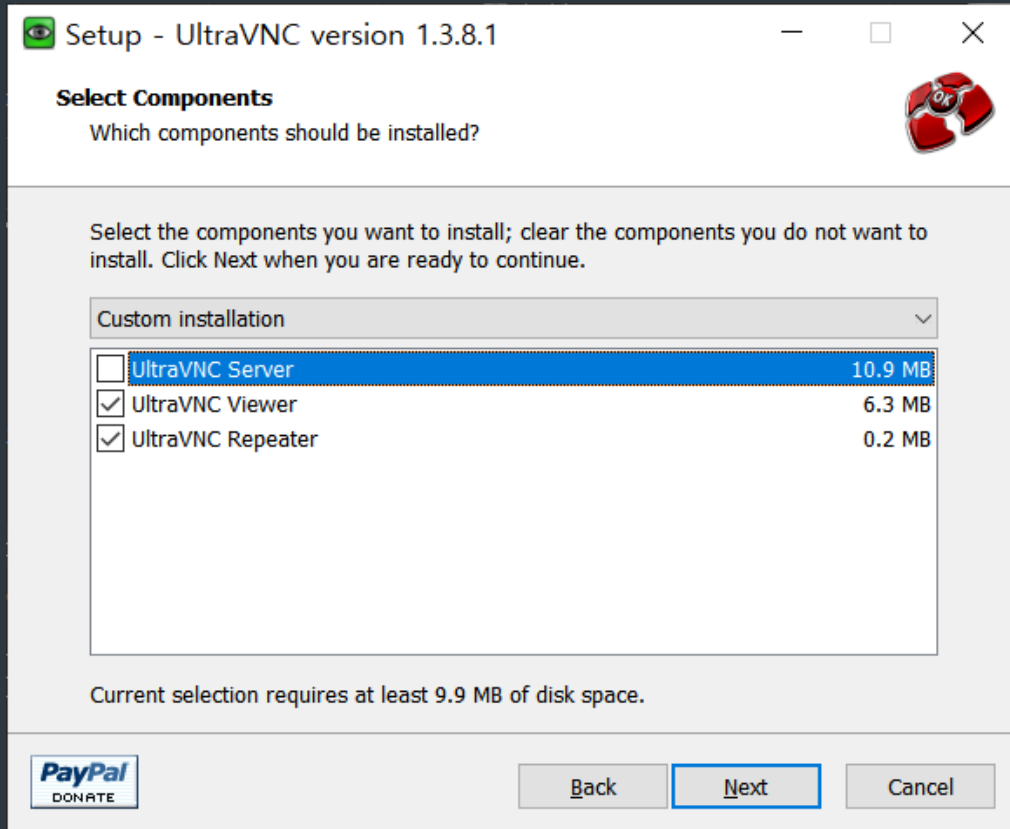
<x11vnc 암호 설정>

```
linaro@linaro-alip:~$ sudo x11vnc -storepasswd
Enter VNC password:
Verify password:
Write password to /root/.vnc/passwd? [y]/n y
Password written to: /root/.vnc/passwd
linaro@linaro-alip:~$ sudo cp /root/.vnc/passwd /etc/x11vnc.pass
linaro@linaro-alip:~$ sudo service x11vnc restart
```

- VNC 보안을 위해 필수적으로 암호 생성 필요
- `sudo x11vnc -storepasswd`
- `sudo cp ~/.vnc/passwd /etc/x11vnc.pass`
- `sudo service x11vnc restart`

VNC

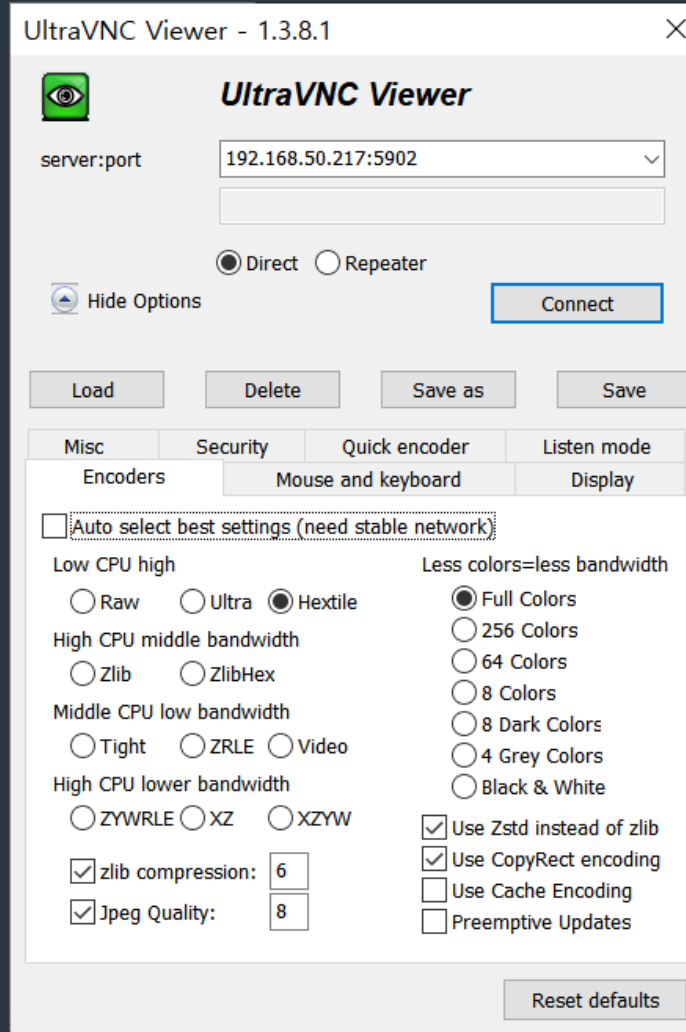
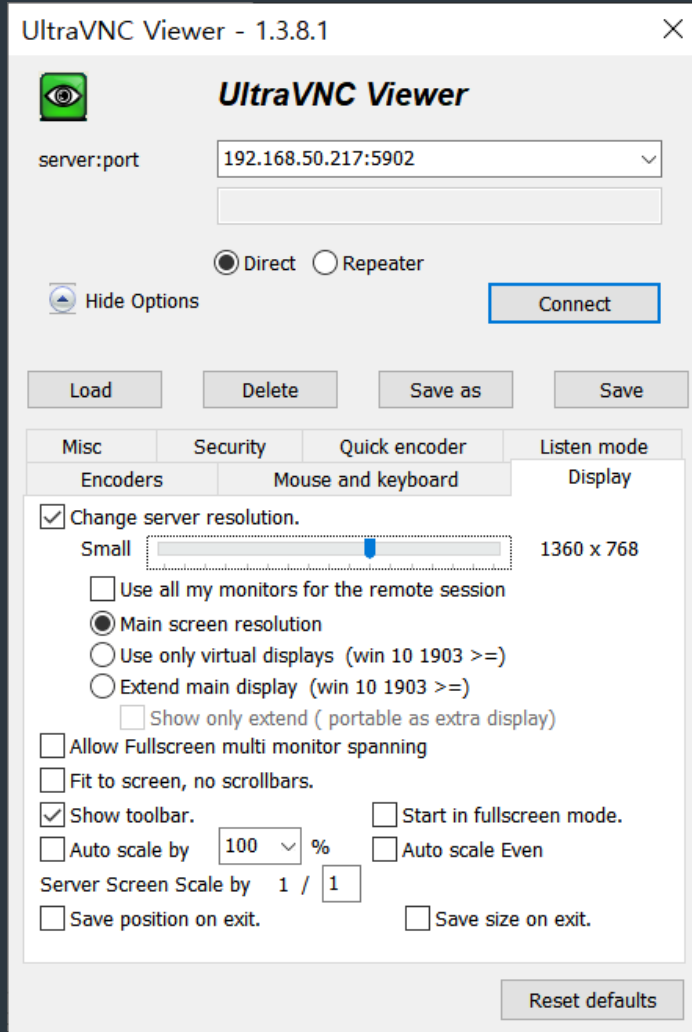
<UltraVNC 클라이언트>



- [UltraVNC](#)는 x11vnc와 궁합이 잘 맞는 VNC 클라이언트
- 홈페이지에서 [다운로드](#)
- Tinker Edge R 제어에는 Viewer만 필요
- UltraVNC Viewer 실행

VNC

<UltraVNC 클라이언트>

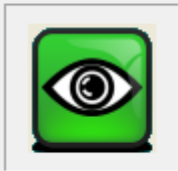


- 해상도 및 화질 선택을 함으로 네트워크 부하를 줄일수 있음
- Tinker Edge R의 IP 주소 입력 후 x11vnc-server 서비스 등록 시 적었던 5902 Port 사용

VNC

<UltraVNC 클라이언트>

VNC Authentication 192.168.50.217:5902



Password:

••••••

Log On

Cancel

```
linaro-alip:0
192.168.50.217:5
system
linaro@linaro-alip: ~
File Edit Tabs Help
Get:2 http://security.debian.org buster/updates/main Sources [249 kB]
Get:3 http://security.debian.org buster/updates/main arm64 Packages [334 kB]
Get:4 http://security.debian.org buster/updates/main Translation-en [183 kB]
Hit:5 http://cdn-fastly.deb.debian.org/debian buster InRelease
Get:6 http://cdn-fastly.deb.debian.org/debian buster-updates InRelease [56.6 kB]
Get:7 http://cdn-fastly.deb.debian.org/debian buster-updates/main Sources.diff/Index [12.1 kB]
Get:8 http://cdn-fastly.deb.debian.org/debian buster-updates/main arm64 Packages.diff/Index [12.1 kB]
Get:9 http://cdn-fastly.deb.debian.org/debian buster-updates/main Sources 2022-08-18-2019.35.pdiff [397 B]
Get:9 http://cdn-fastly.deb.debian.org/debian buster-updates/main Sources 2022-08-18-2019.35.pdiff [397 B]
Get:10 http://cdn-fastly.deb.debian.org/debian buster-updates/main arm64 Packages 2022-08-18-2019.35.pdiff [283 B]
Get:10 http://cdn-fastly.deb.debian.org/debian buster-updates/main arm64 Packages 2022-08-18-2019.35.pdiff [283 B]
Fetched 882 kB in 9s (99.4 kB/s)
Reading package lists... Done
linaro@linaro-alip:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
linaro@linaro-alip:~$ sudo nano /lib/systemd/system/x11vnc.service
linaro@linaro-alip:~$ sudo nano /lib/systemd/system/x11vnc.service
linaro@linaro-alip:~$ ^C
linaro@linaro-alip:~$ sudo x11vnc -storepasswd
Enter VNC password:
Verify password:
Write password to /root/.vnc/passwd? [y]/n y
Password written to: /root/.vnc/passwd
linaro@linaro-alip:~$ sudo cp /root/.vnc/passwd /etc/x11vnc.pass
linaro@linaro-alip:~$ sudo service x11vnc restart
```

etc – AP 클럭 조정

<tinker-power-management>

```
linaro@linaro-alip:~$ tinker-power-management  
linaro@linaro-alip:~$ █
```

- tinker-power-management으로 AP 클럭 및 정책 설정 가능
- (C)PU와 (G)PU 키로 각각 조절 가능
- "auto", "manual", "powersave", "performance" 옵션 존재
- Manual로 A72 big 코어 및 A53 LITTLE 코어, GPU 클럭 세팅 가능
- AP 온도 및 AP 로드율 확인 가능

etc – AP 클럭 조정

<tinker-power-management>

Device Info

Name

I

= Tinker Edge R

Version

= 1.04

SN/PPID

= bala6d9045005afa

SoC/CPU

= RK3399PR0

Memory

= 4GB

Storage

= eMMC(mmcblk1) 14.6G

No SD card(mmcblk0)

OS

= Debian GNU/Linux

Version

= 10 (buster)

Platform

= aarch64

Build

= v2.0.5-20220217

Kernel:

Release

= Linux version 4.4.194

Version

= #4 SMP Tue Feb 15 03:07:56 UTC 2022

System Config

CPU:

Governor = auto manual powersave performance

	Min. freq.	Max. freq.
4 x Arm Cortex-A53	408	1416
2 x Arm Cortex-A72	408	1800

GPU:

Governor = auto

	Min. freq.	Max. freq.
Arm Mali-T86x	200	800

Monitor

CPU:

	Curr. freq.	Temperature
4 x Arm Cortex-A53	600	37.50°C
2 x Arm Cortex-A72	1608	37.50°C

CPU usage:

8%

[####

]

GPU:

	Curr. freq.	Temperature
Arm Mali-T86x	200	36.88°C

GPU usage:

0%

[

]

Memory usage:

447 / 3937 MB (11%)

[#####

]

Press Left or Right keys to select the governor and Space key to save. Press Q to quit and go back.

etc – DD 명령어로 Custom 이미지 제작

<DD>

```
linaro@linaro-alip:~$ sudo fdisk -l
Disk /dev/ram0: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/mmcblk1: 14.6 GiB, 15634268160 bytes, 30535680 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 4B7AFFDF-55D1-4C03-B3A5-071AA934F7AB
```

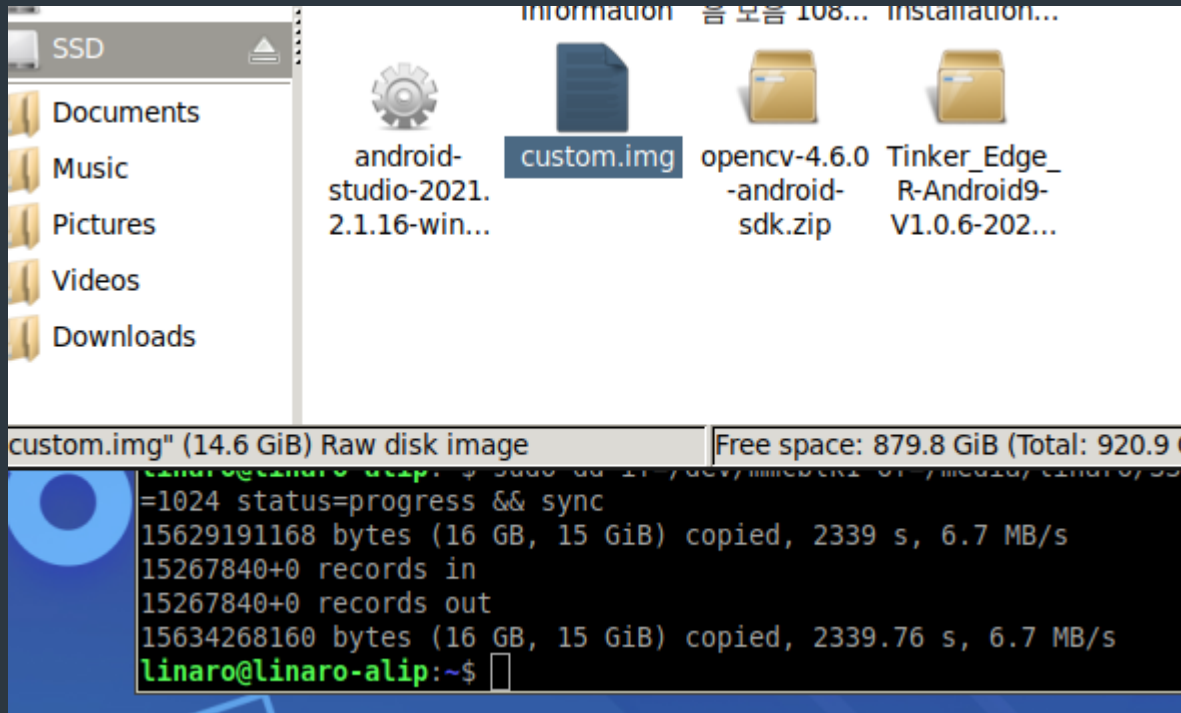
Device	Start	End	Sectors	Size	Type
/dev/mmcblk1p1	16384	24575	8192	4M	Linux filesystem
/dev/mmcblk1p2	24576	32767	8192	4M	Linux filesystem
/dev/mmcblk1p3	32768	40959	8192	4M	Linux filesystem
/dev/mmcblk1p4	40960	106495	65536	32M	Linux filesystem
/dev/mmcblk1p5	106496	303103	196608	96M	Linux filesystem
/dev/mmcblk1p6	303104	368639	65536	32M	Linux filesystem
/dev/mmcblk1p7	368640	499711	131072	64M	Linux filesystem
/dev/mmcblk1p8	499712	30535646	30035935	14.3G	Linux filesystem

- 이미지를 제작해 넣을 USB
메모리 연결
- `sudo fdisk -l` 로 disk 목록 확인
`/dev/mmcblk1 -> emmc`
`/dev/mmcblk0 -> sdcard`

etc – DD 명령어로 Custom 이미지 제작

<DD>

```
linaro@linaro-alip:~$ sudo dd if=/dev/mmcblk1 of=/media/linaro/SSD/custom.img bs=1024 status=progress && sync
786769920 bytes (787 MB, 750 MiB) copied, 110 s, 7.2 MB/s
```



- dd 명령어를 사용해 /dev/mmcblk1 emmc를 통째로 img화
- if는 이미지화할 드라이브
of는 *.img로 나올 드라이브
bs는 블록 크기 설정
progress && sync는 dd 명령어 진행 사항 확인