Install Arch Linux

Check UEFI

```
ls /sys/firmware/efi/efivars
```

Check network

```
ping google.com
```

Check interface status

```
ip link
```

set network interface up

```
ip link set INTERFACE up
```

Connect Wi-Fi via netctl

create network profile in /etc/netctl profile sample can be copied from /etc/netctl/examples

```
netctl start PROFILE
```

Connnect Wi-Fi via wpa_supplicant

create profile via **wpa_passphrase**

```
wpa_passphrase ESSID PASSWD >> /etc/wpa_supplicant/wpa_supplicant.conf
wpa_supplicant -B -i INTERFACE -c /etc/wpa_supplicant/wpa_supplicant.conf
dhcpcp
```

Connect Wi-Fi via wifi-menu

```
wifi-menu
```

Set time

```
timedatectl set-ntp true
```

if dual boot with MS Windows, use local time

```
timedatectl set-local-rtc true
```

Partition

GPT use **gdisk**

```
gdisk /dev/sdX
```

MBR use **fdisk**

```
fdisk /dev/sdX
```

UEFI need EFI partition, code is ef00

Format disk

for linux system file

```
mkfs.ext4 /dev/sdXN
```

for EFI partition

```
mkfs.fat -F32 /dev/sdXN
```

Mount partition

```
mount /dev/sdXN /mnt
```

if addition partitions or mount points, e.g. /boot, /home...

```
mkdir /mnt/boot
mount /dev/sdXN /mnt/boot
```

Check mirrors

```
cat /etc/pacman.d/mirrorlist
```

if want to rank disk, use **rankmirrors** (from **pacman-contrib** package)

```
pacman -S pacman-contrib
cp /etc/pacman.d/mirrorlist /etc/pacman.d/mirrorlist.backup
rankmirrors -n 5 /etc/pacman.d/mirrorlist.backup > /etc/pacman.d/mirrorlist
```

Install base packages to new system

```
pacstrap /mnt base base-devel linux linux-firmware
```

other kernel can be added as well

Generate fstab file and check

```
genfstab -U /mnt >> /mnt/etc/fstab
cat /mnt/etc/fstab
```

Change root to new system

```
arch-chroot /mnt
```

Set time

```
ln -sf /usr/share/zoneinfo/Asia/Hong_Kong /etc/localtime
timedatectl set-ntp true
hwclock --systohc
```

if dual boot with MS Windows, don't use ntp

```
timedatectl set-local-rtc true
hwclock --localtime --systohc
```

Localisation

edit *locale.gen* file

```
echo "en_US.UTF-8" >> /etc/locale.gen
echo "en_HK.UTF-8" >> /etc/locale.gen
echo "zh_HK.UTF-8" >> /etc/locale.gen
```

```
locale-gen
```

if need to edit *locale.conf*

```
locale
vim locale.conf
```

Hosts and hostname

```
echo "HOSTNAME" >> /etc/hostname
vim /etc/hosts
```

/etc/hosts

```
127.0.0.1   localhost
::1         localhost
127.0.1.1   HOSTNAME.localdomain    HOSTNAME
```

Create new initramfs (usually not needed)

```
mkinitcpio -P
```

Update

```
pacman -Syu
```

Root password

```
passwd
```

Install boot loader

Install GRUB

```
grub-install --target=i386-pc /dev/sdX
```

for UEFI (ESP whould be in EFI partition)

```
grub-install --target=x86_64-efi --efi-directory=ESP --bootloader-id=GRUB
```

Install rEFInd (UEFI only)

```
refind-install
```

if fail to detect ESP, assign directory and create *refind_linux.conf*

```
refind-install --usedefault /dev/sdXN
mkrlconf
```

since refind-install is run in chroot, need to edit the path in *refind_linux.conf*

```
vim /boot/refind_linux.conf
```

change the UUID (or PARTUUID) to the **root** partition

Install neccessary package

Network

```
netctl
wpa_supplicant
dhcpcd
dialog (if want to use wifi-menu)
```

```
net-tools (deprecated, use iproute2)
networkmanager (if want to use desktop environment)
```

Drivers

graphics

```
xf86-video-intel
xf86-video-amdgpu
xf86-video-ati
xf86-video-nouveau
nvidia
nvidia-390xx
```

OpenGL

```
mesa, lib32-mesa (except NVIDIA)
nvidia-utils, lib32-nvidia-utils (NVIDIA)
nvidia-390xx-utils, lib32-nvidia-390xx-utils (NVIDIA)
```

touchpad

```
xf86-input-synaptics (for touchpad)
```

The drivers should be included as dependencies when installing **Xorg**, check (https://wiki.archlinux.org/index.php/Xorg)

Font

```
noto-fonts, noto-fonts-cjk, noto-fonts-emoji
dejavu
source code pro
fonts awesome
...
```

Others

```
tlp (power manager)
vim
zsh
git
firefox
pacman-contrib
...
```

Add new user

```
useradd -m -G wheel (-s /bin/zsh) USER
passwd USER
```

-m to create home path -G to add to group -s to assign shell

Edit sudoers, uncomment wheel

```
visudo
```

Unmount all mount points and reboot

```
umount -R /mnt
reboot
```

Install desktop environment and display manager

Install Xfce, Gnome, KDE5, E17

```
sudo pacman -S gnome
sudo pacman -S plasma sddm kde-applications
sudo pacman -S xfce4 xfce4-goodies lightdm lightdm-gtk-greeter
sudo pacman -S enlightenment lightdm terminology
```

Install display manager if needed

```
sudo pacman -S lightdm, sddm, xdm (select one or more)
```

Install i3wm (without display manager)

```
sudo pacman -S i3wm
```

Create config file and customize

```
mkdir ~/.config/i3
cp /etc/i3/config ~/.config/i3/config
```

create custom directory for i3status/i3blocks

```
mkdir ~/.config/i3ststus
mkdir ~/.config/i3blocks
mkdir ~/.scripts/i3blocks (to contain i3blocks scripts)
```

install **dmenu** if needed

```
sudo pacman -S dmenu
```

Install input method (ibus-rime)

```
sudo pacman -S ibus-rime
ibus-setup
```

add following lines to ~/.bashrc (or ~/.zshrc, ~/.xprofile, ~/.xinitrc)

```
export GTK_IM_MODULE=ibus
export XMODIFIERS=@im=ibus
export QT_IM_MODULE=ibus
ibus-daemon -drx (ibus-daemon --xim -d -r)
```

Install terminal emulator (urxvt)

```
sudo pacman -S rxvt-unicode
```

edit ~/.Xresources to customize

Configure firewall

by **iptables**

```
iptables-restore < /etc/iptables/empty.rules
iptables -N TCP
iptables -N UDP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
iptables -P INPUT DROP
```

```
iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -m conntrack --ctstate INVALID -j DROP
iptables -A INPUT -p icmp --icmp-type 8 -m conntrack --ctstate NEW -j ACCEPT
iptables -A INPUT -p udp -m conntrack --ctstate NEW -j UDP
iptables -A INPUT -p tcp --syn -m conntrack --ctstate NEW -j TCP
iptables -A INPUT -j REJECT --reject-with icmp-proto-unreachable
iptables -A TCP -p tcp --dport 22 -j ACCEPT
iptables -I TCP -p tcp -m recent --update --rsource --seconds 60 --name TCP-PORTSCAN -j REJECT --reject
iptables -A INPUT -p tcp -m recent --set --rsource --name TCP-PORTSCAN -j REJECT --reject-with tcp-rese
iptables -I UDP -p udp -m recent --update --rsource --seconds 60 --name UDP-PORTSCAN -j REJECT --reject
iptables -A INPUT -p udp -m recent --set --rsource --name UDP-PORTSCAN -j REJECT --reject-with icmp-por
iptables -A INPUT -j REJECT --reject-with icmp-proto-unreachable
iptables-save -f /etc/iptables/iptables.rules
```

check (https://wiki.archlinux.org/index.php/Simple_stateful_firewall)

Advance setting

Touchpad

use **libinput** instead of **xf86-input-synaptics**

```
cp /usr/share/X11/xorg.conf.d/70-synaptics.conf /etc/X11/xorg.conf.d/
```

Sample using synaptics (https://wiki.archlinux.org/index.php/Touchpad_Synaptics)

```
Section "InputClass"
    Identifier "touchpad"
    Driver "synaptics"
    MatchIsTouchpad "on"
        Option "TapButton1" "1"
        Option "TapButton2" "3"
        Option "TapButton3" "2"
        Option "VertEdgeScroll" "on"
        Option "VertTwoFingerScroll" "on"
        Option "HorizEdgeScroll" "on"
        Option "HorizTwoFingerScroll" "on"
        Option "CircularScrolling" "on"
        Option "CircScrollTrigger" "2"
        Option "EmulateTwoFingerMinZ" "40"
        Option "EmulateTwoFingerMinW" "8"
        Option "CoastingSpeed" "0"
        Option "FingerLow" "30"
        Option "FingerHigh" "50"
        Option "MaxTapTime" "125"
EndSection
```

Sample using libinput

since the file is *40-libinput.conf* which order lower then *70-synaptics.conf*, use **ln** to avoid

```
ln -s /usr/share/X11/xorg.conf.d/40-libinput.conf /etc/X11/xorg.conf.d/40-libinput.conf
```