

Creality Sonic Pad

自定义机型固件编译指导书

- 1、下载安装 VirtualBox
- 2、下载 Ubuntu
- 3、新建虚拟机并安装 Ubuntu 系统
- 4、系统升级及安装 git 环境
- 5、下载 Klipper 固件
- 6、配置打印机固件
- 7、通过 SD/TF 卡升级打印机固件
- 8、配置 USB 升级打印机固件

1、下载安装 VirtualBox

在 www.VirtualBox.org 下载 VirtualBox

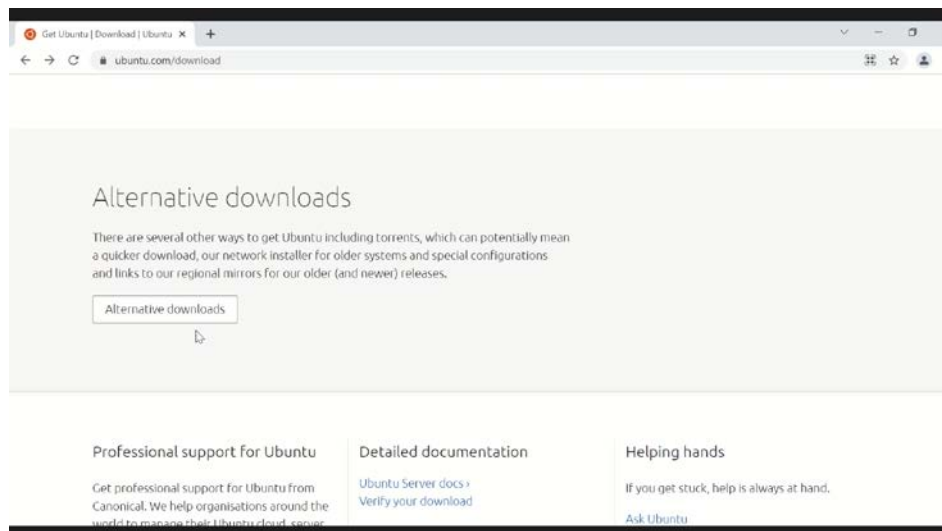
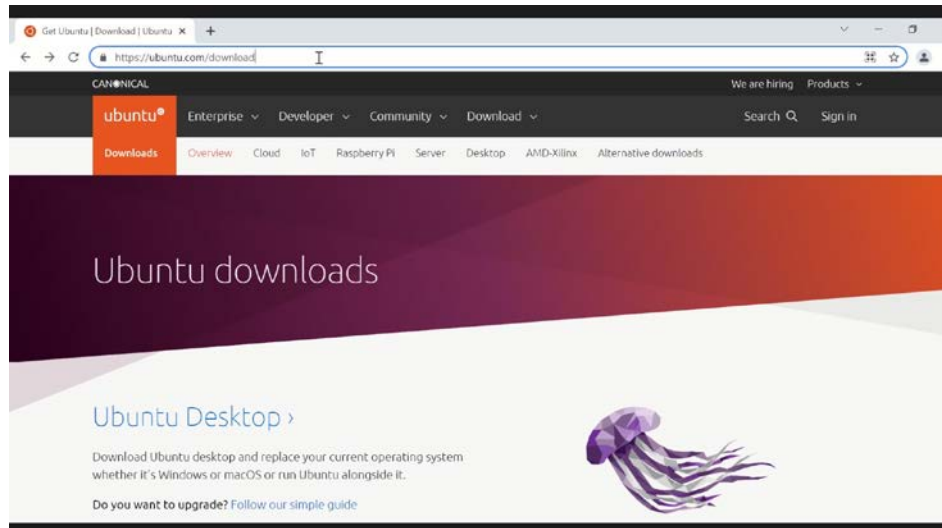


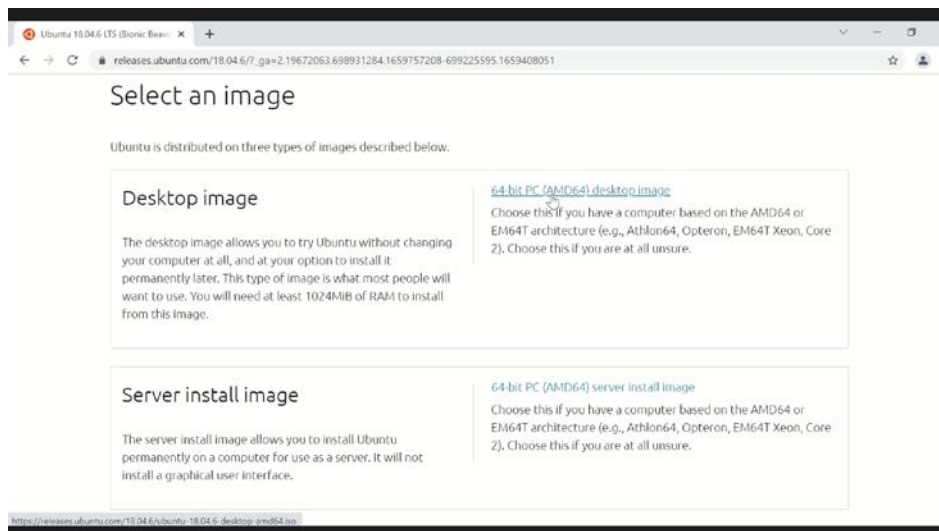
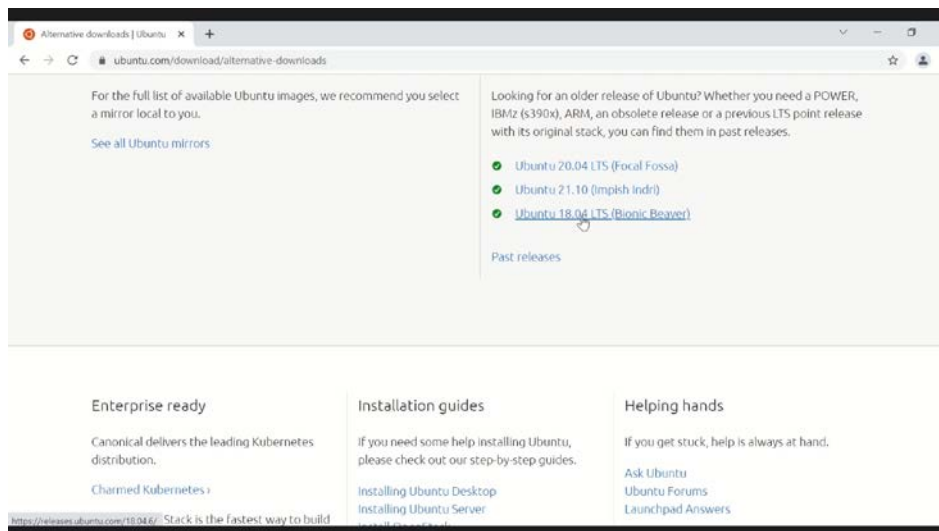
下载完成后安装 VirtualBox（安装完成后先不要打开）



2、下载 Ubuntu

在 [www.ubuntu.com/download](https://ubuntu.com/download) 下载 Ubuntu 18.04.LTS



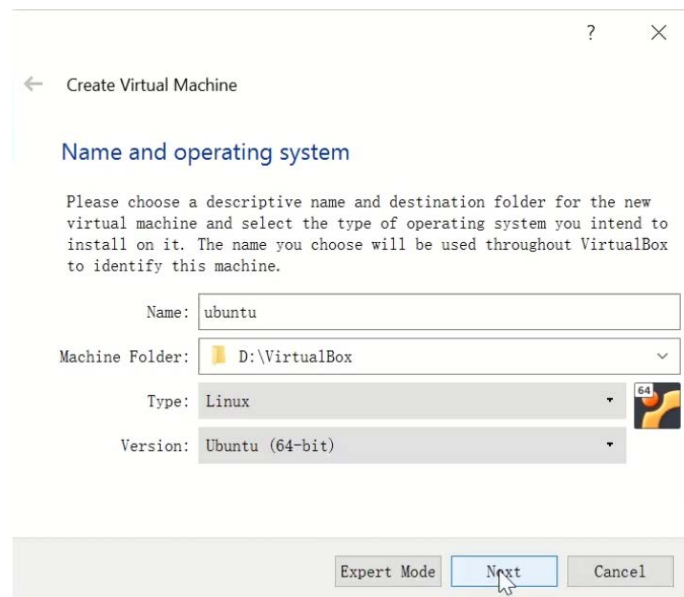


3、新建虚拟机并安装 Ubuntu 系统

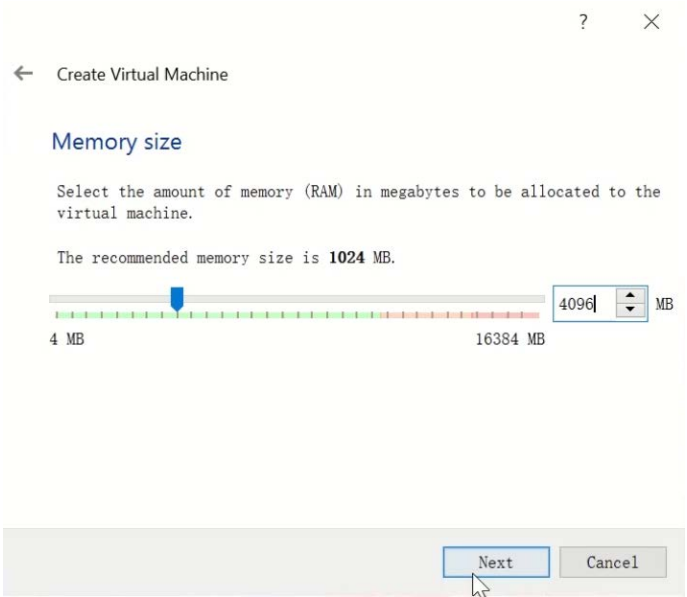
打开 VirtualBox 并新建虚拟机



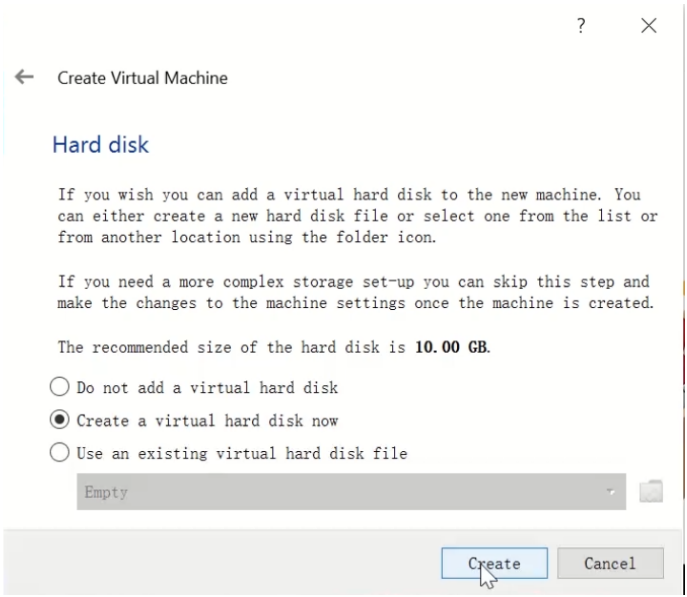
选择工程文件存放位置、输入名称,
类型选择 Linux、版本选择 Ubuntu (64-bit)



内存大小调整为 4096MB



创建虚拟硬盘



选择 VDI

← Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

☒ VDI (VirtualBox Disk Image)
☐ VHD (Virtual Hard Disk)
☐ VMDK (Virtual Machine Disk)

Expert Mode **Next** Cancel

选择动态分配

← Create Virtual Hard Disk

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

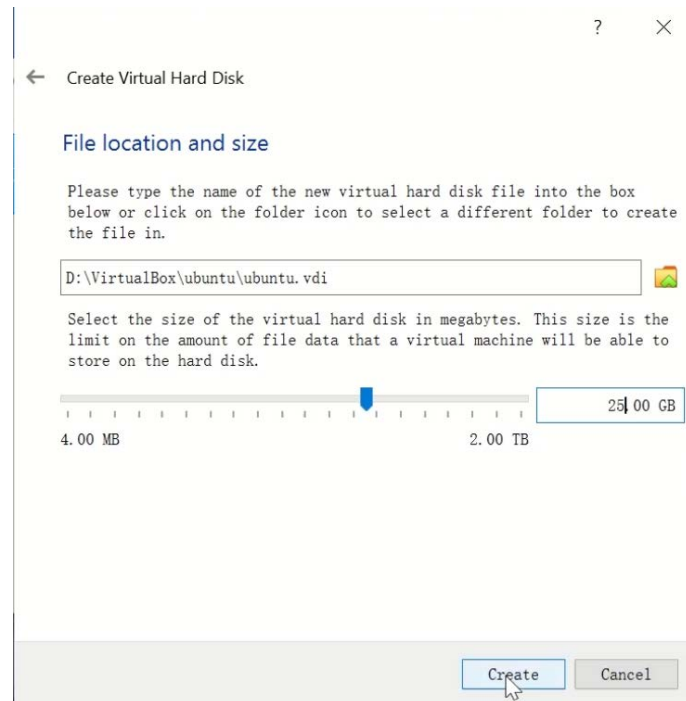
A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

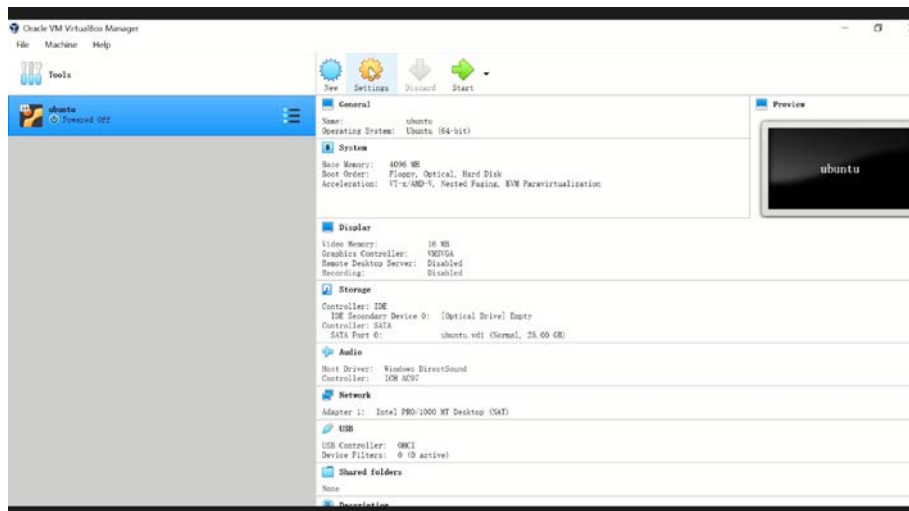
☒ Dynamically allocated
☐ Fixed size

Next Cancel

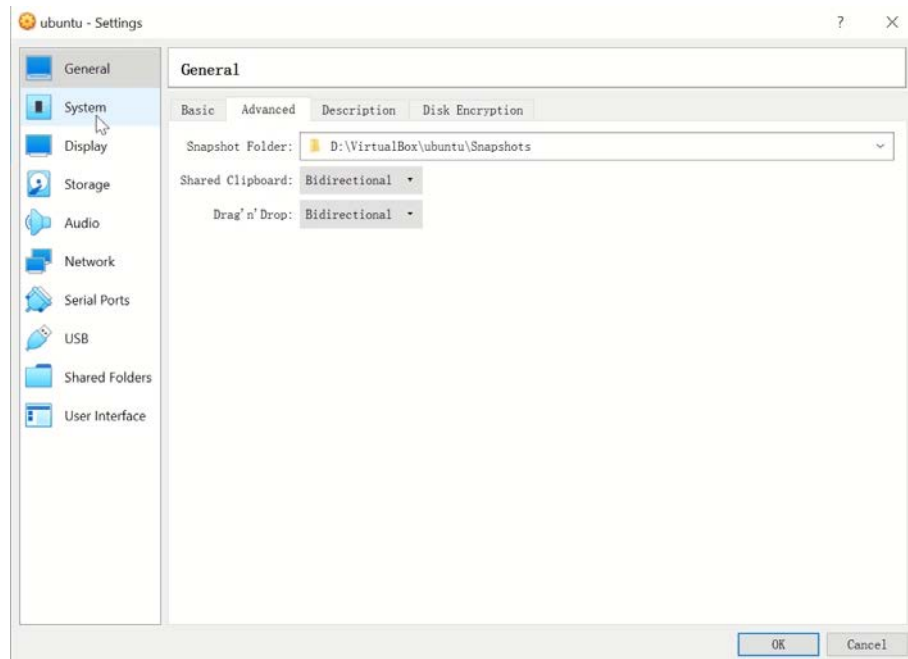
虚拟硬盘大小调整为 25GB



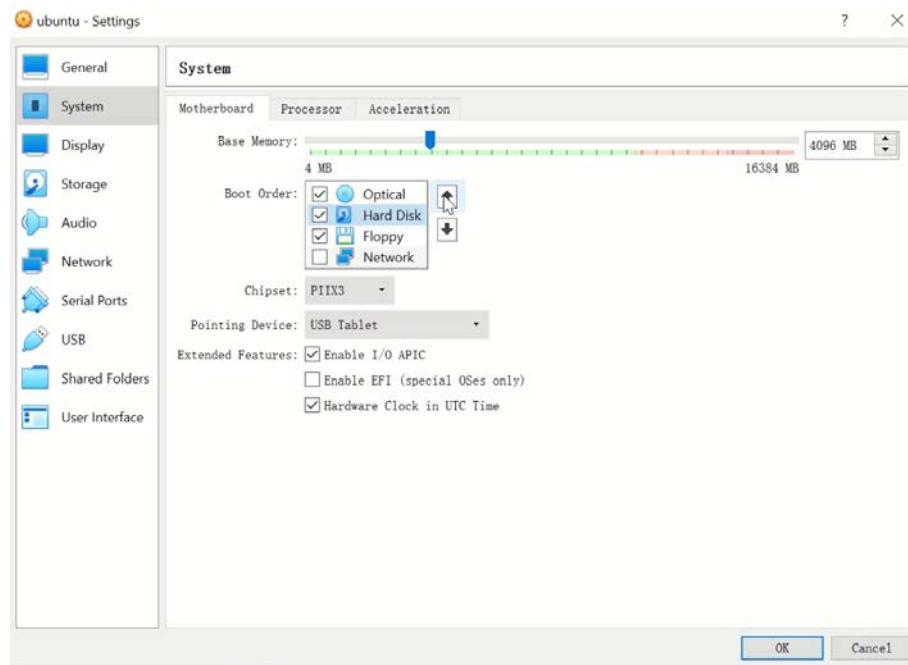
打开设置



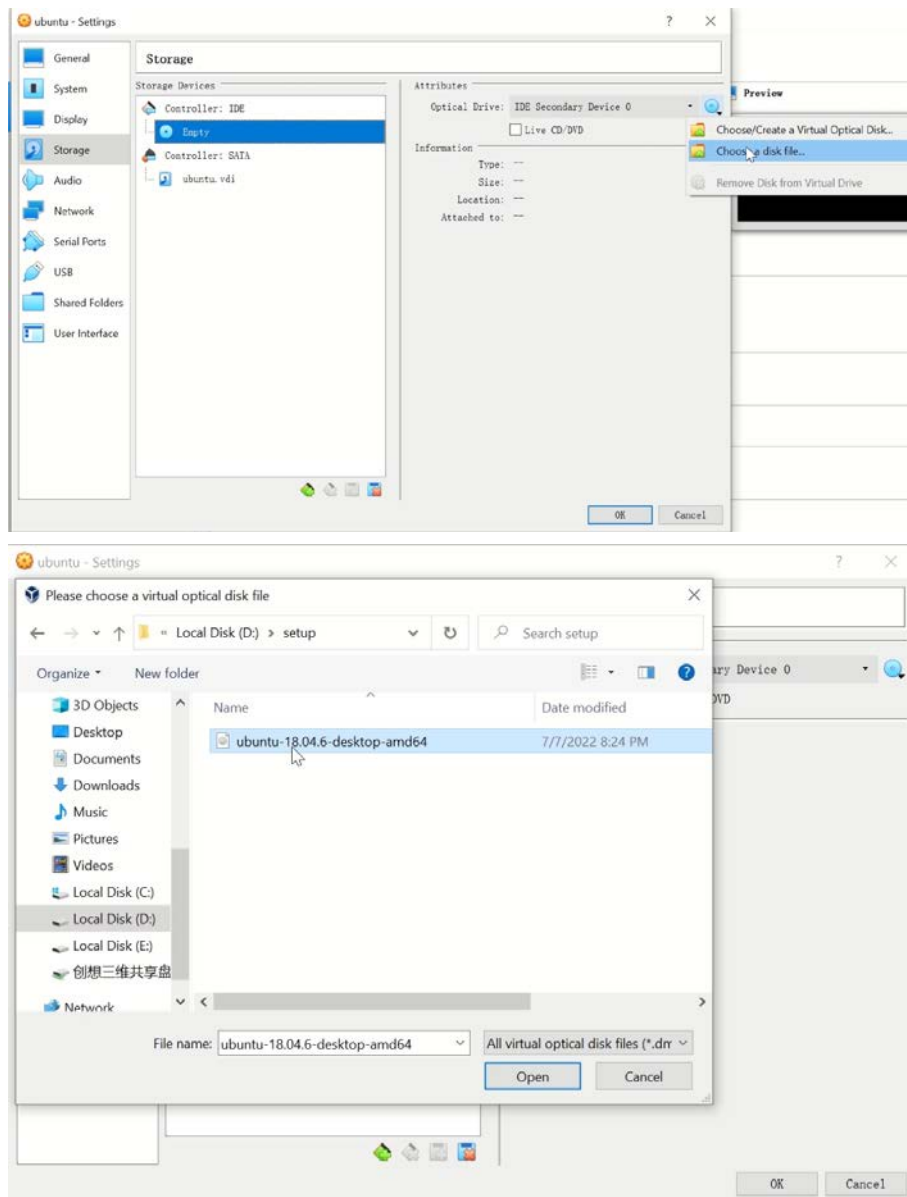
常规设置--高级 将共享粘贴板、拖放改为双向



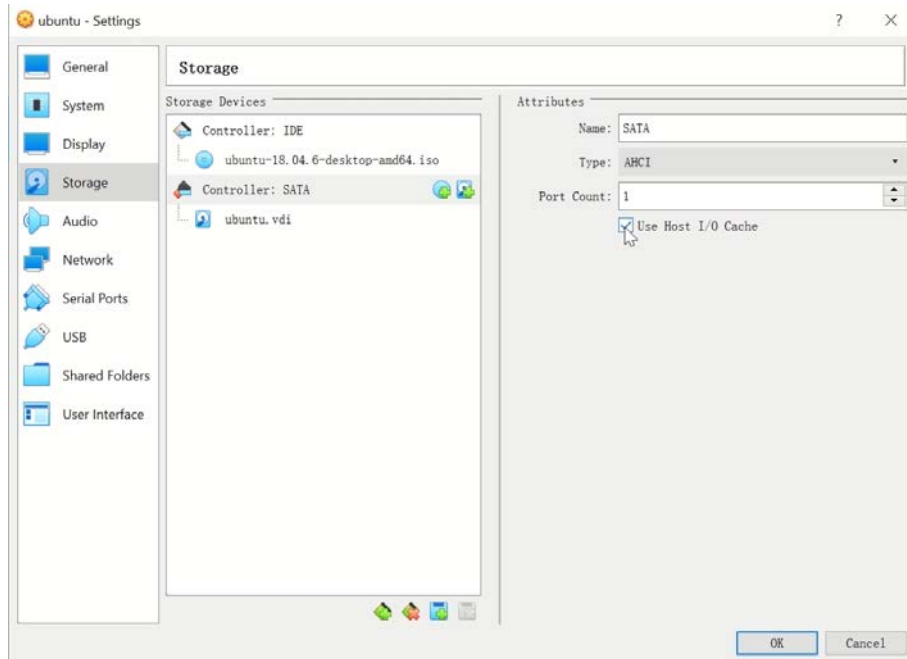
系统设置--调整启动顺序为光驱-硬盘-软驱



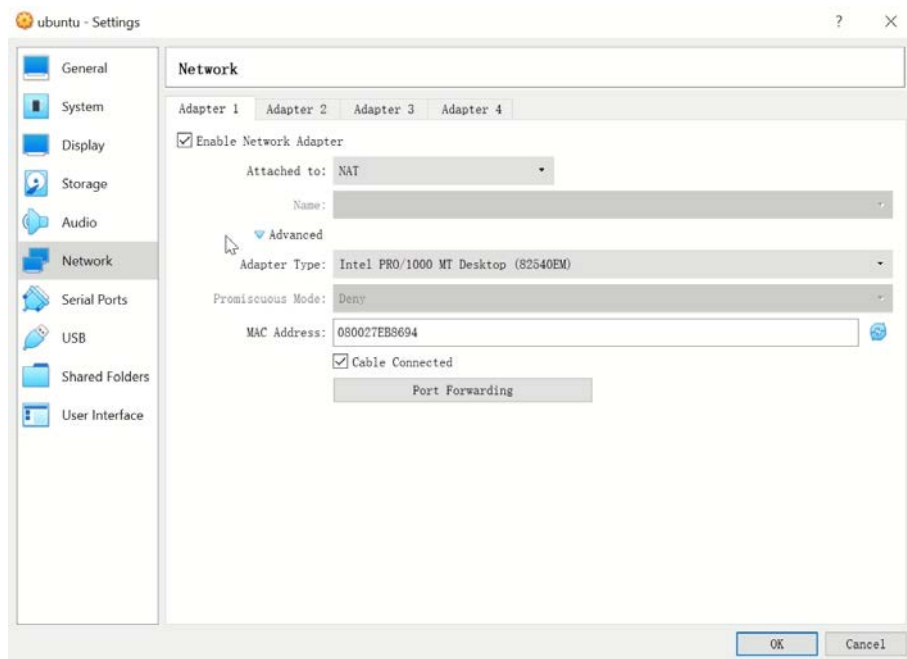
存储设置--IDE--选择下载的 Ubuntu 18.04.6 LTS



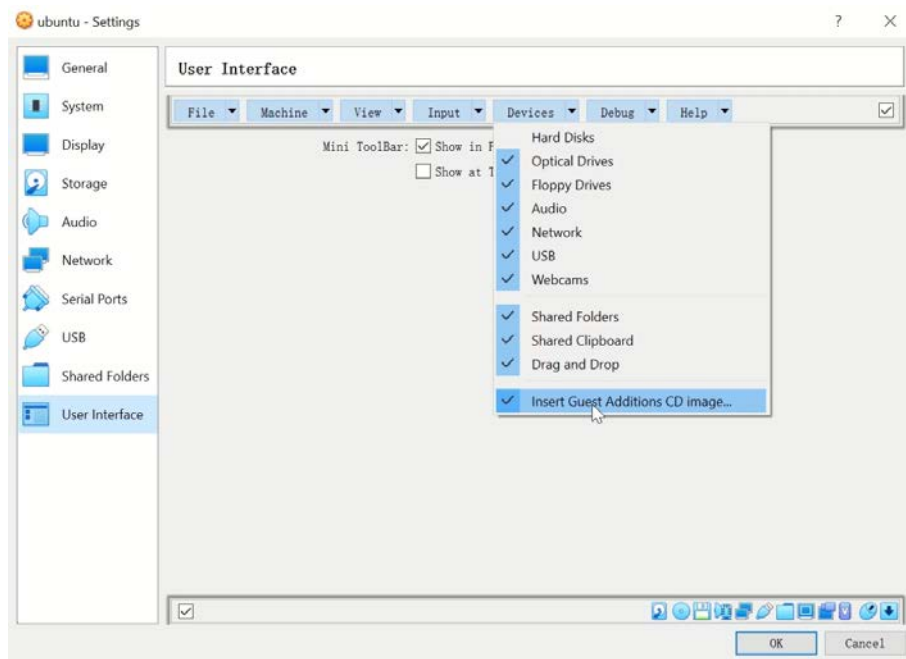
SATA--勾选使用 I/O 口



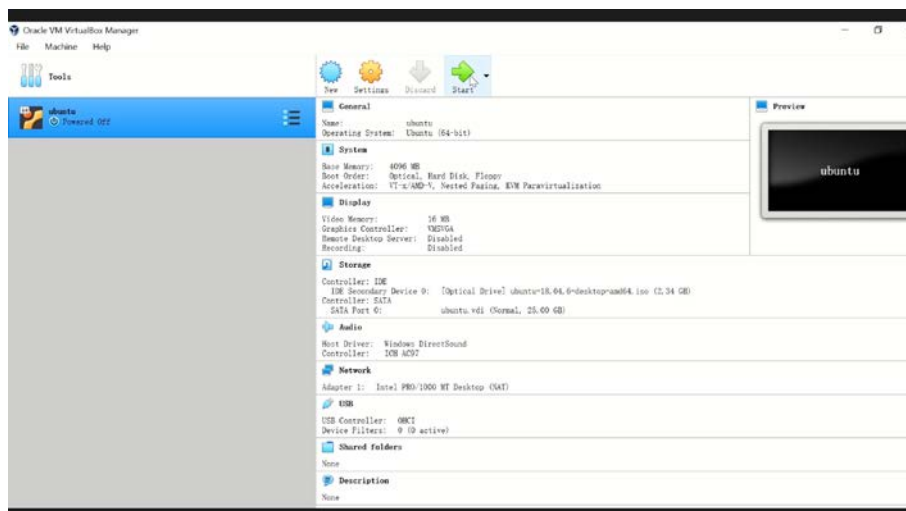
网络设置--高级--确认网络已连接



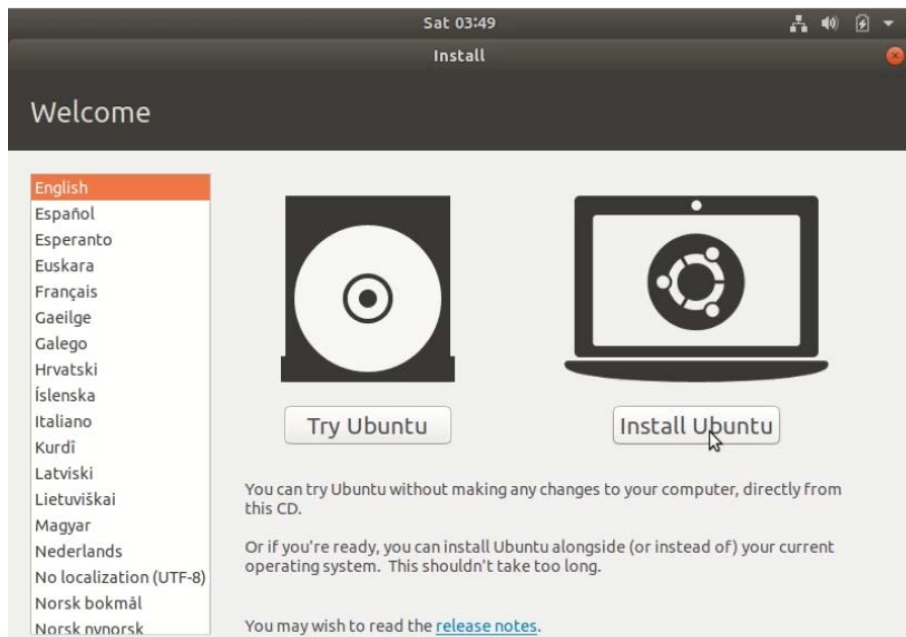
用户界面设置--设备--勾选安装增强功能



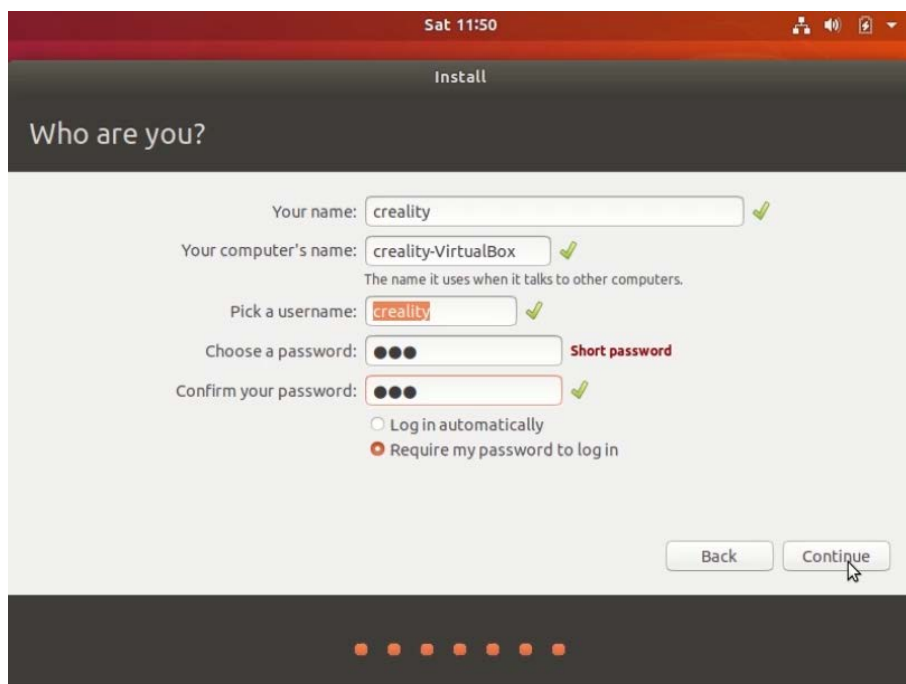
确认后启动虚拟机



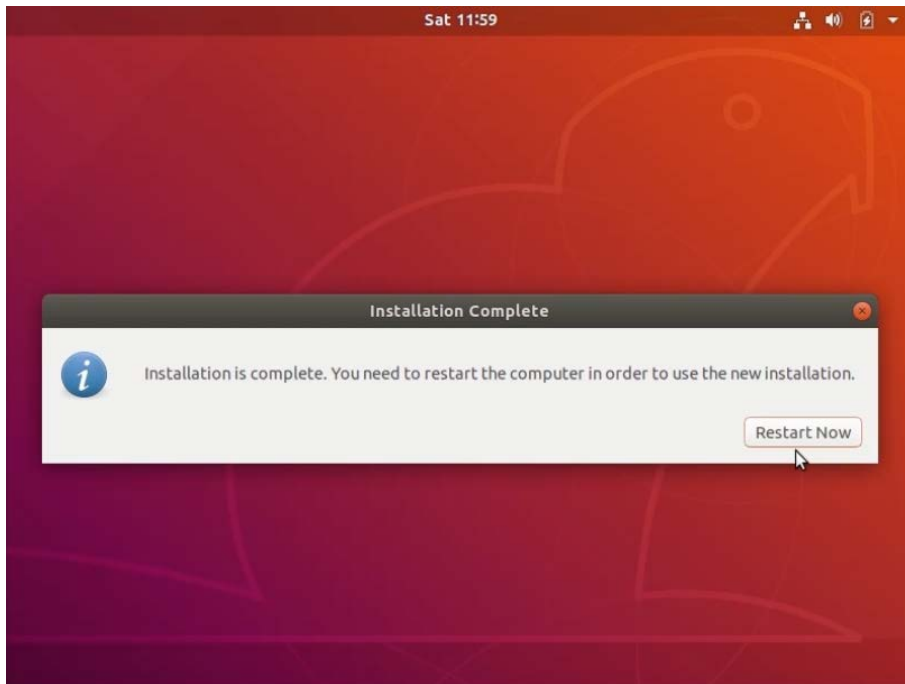
安装 Ubuntu



添加用户名及系统密码



确认后等待重启



4、系统升级及安装 git 环境

输入刚刚设置的系统密码进入系统

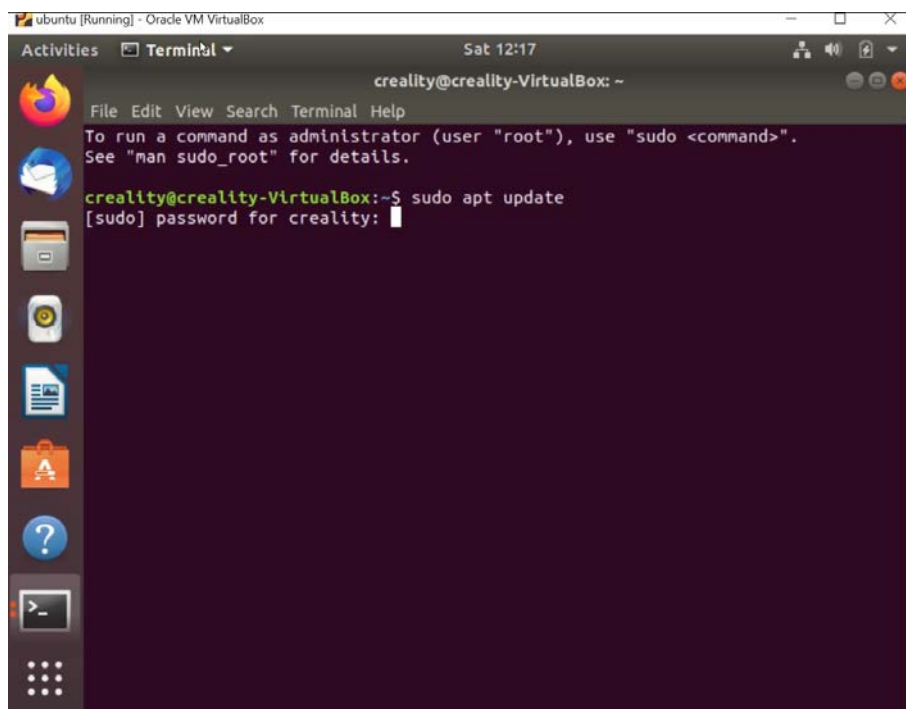


桌面--右键--打开终端

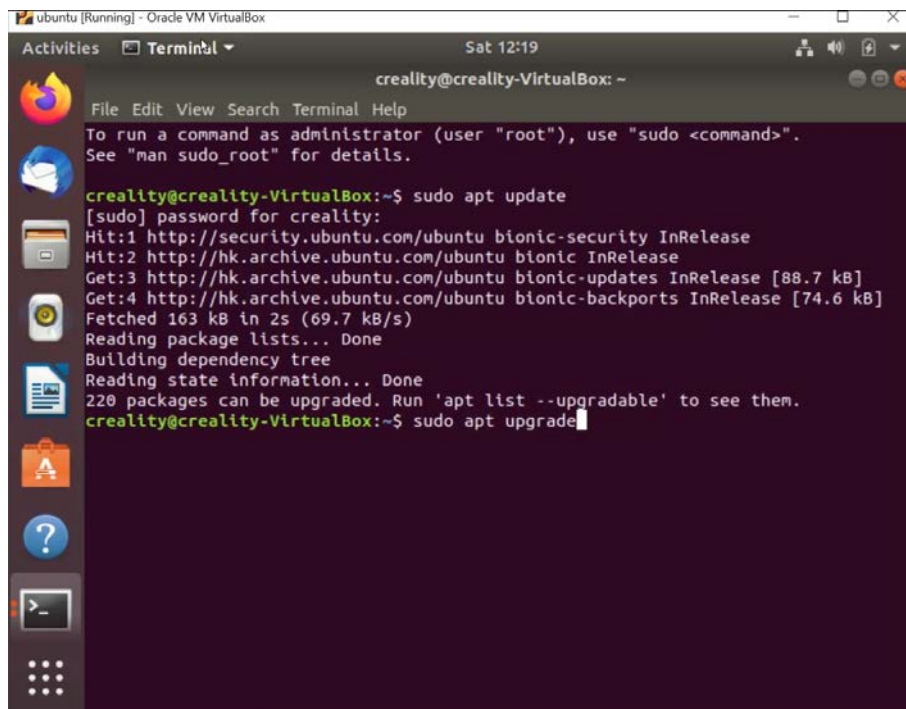


输入“sudo apt update”后回车（列出所有可更新的软件清单）

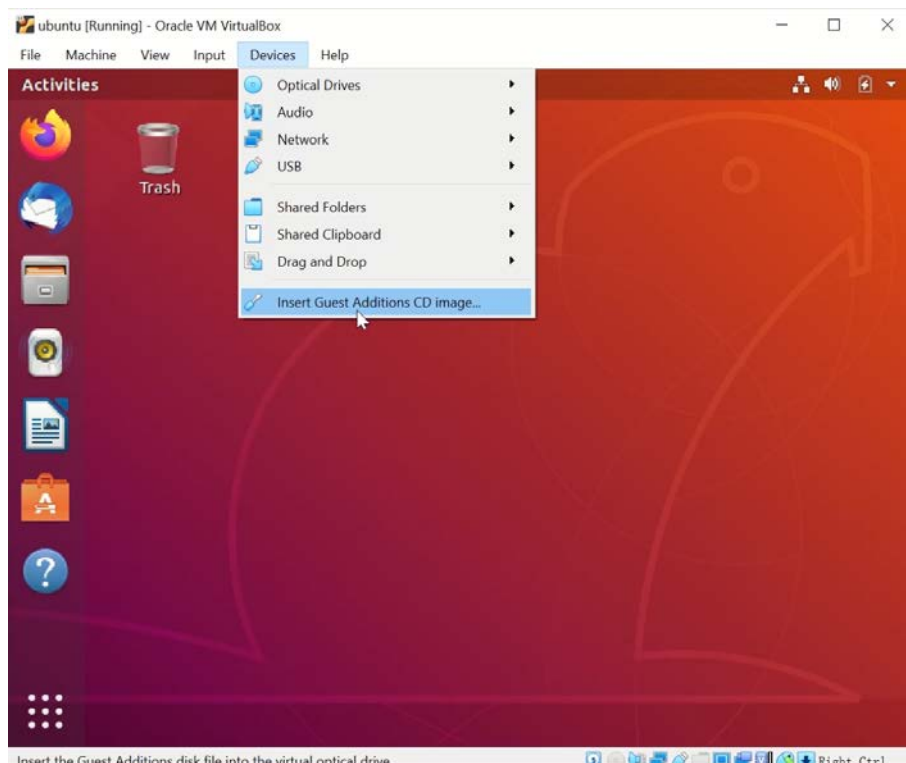
输入系统密码后回车（密码自动隐藏）



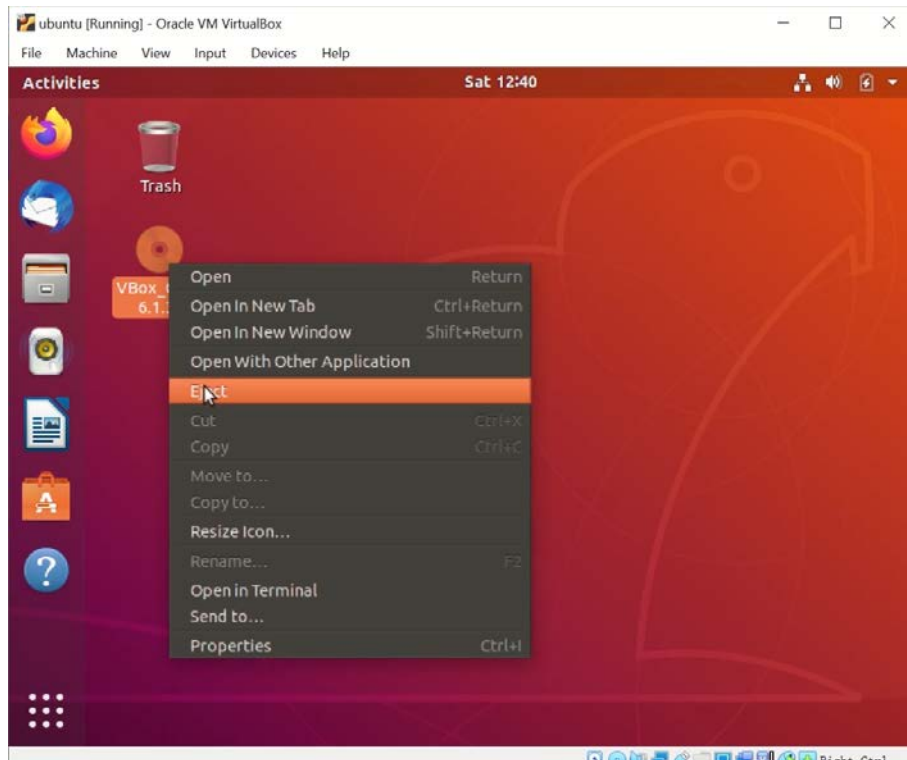
输入“sudo apt upgrade”后回车（升级系统软件包）



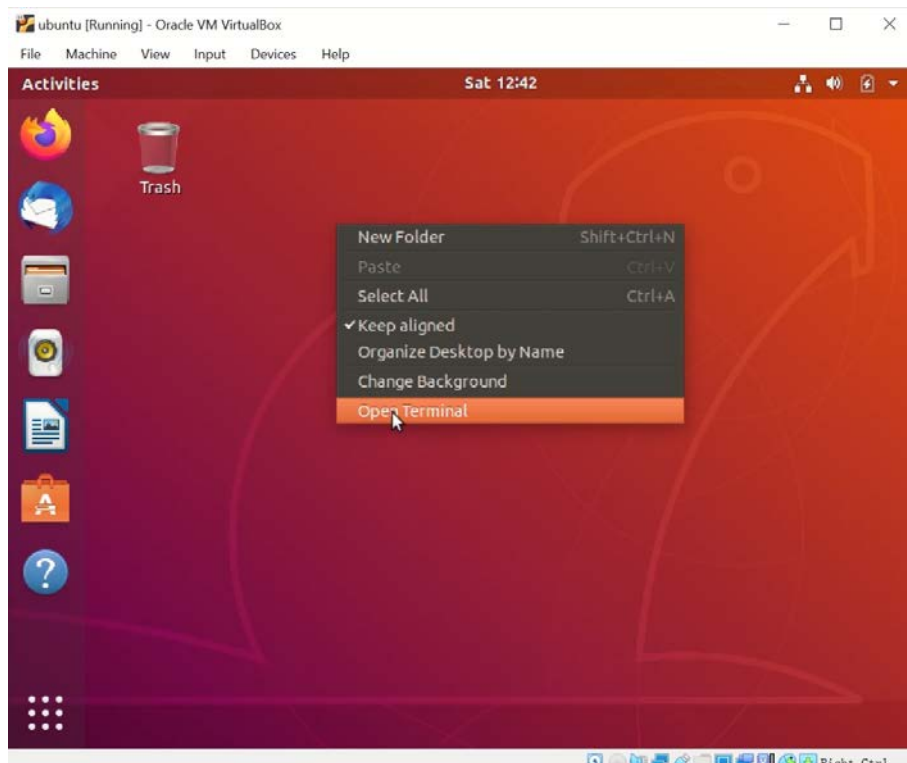
完成后关闭终端窗口，设备--安装增强功能



安装完成后弹出 VBox_Gas

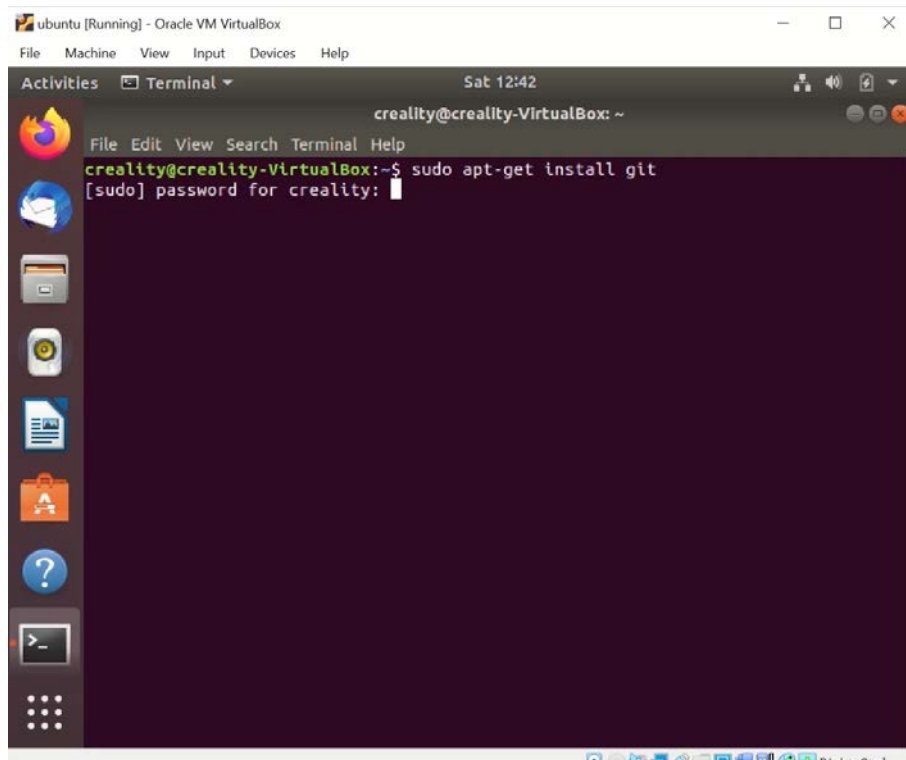


桌面--右键--打开终端

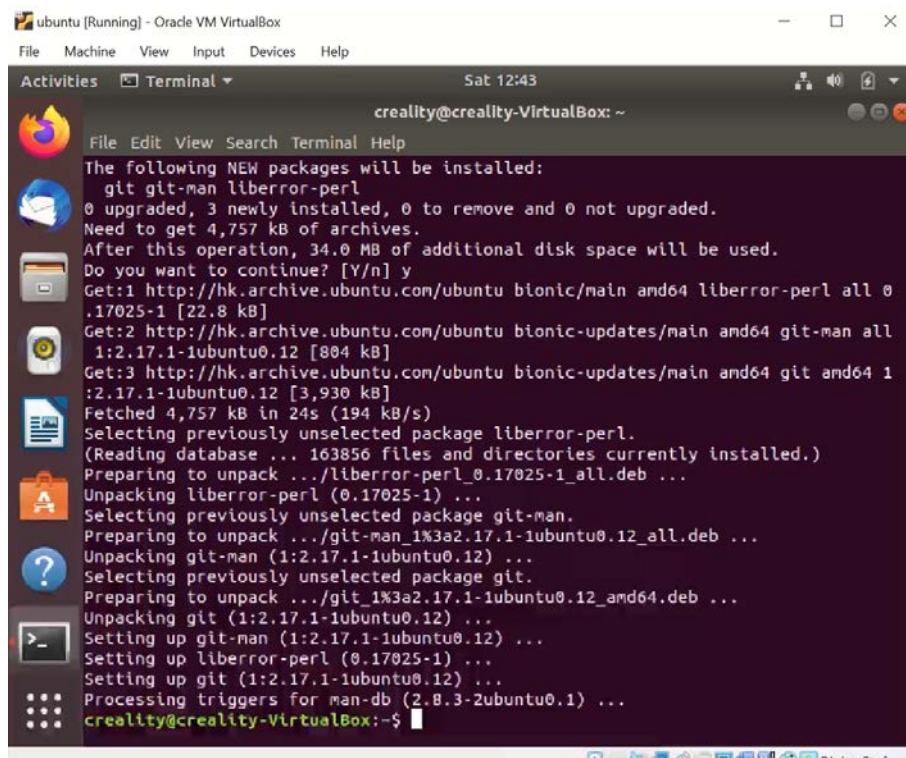


输入“sudo apt-get install git”后回车（安装 git）

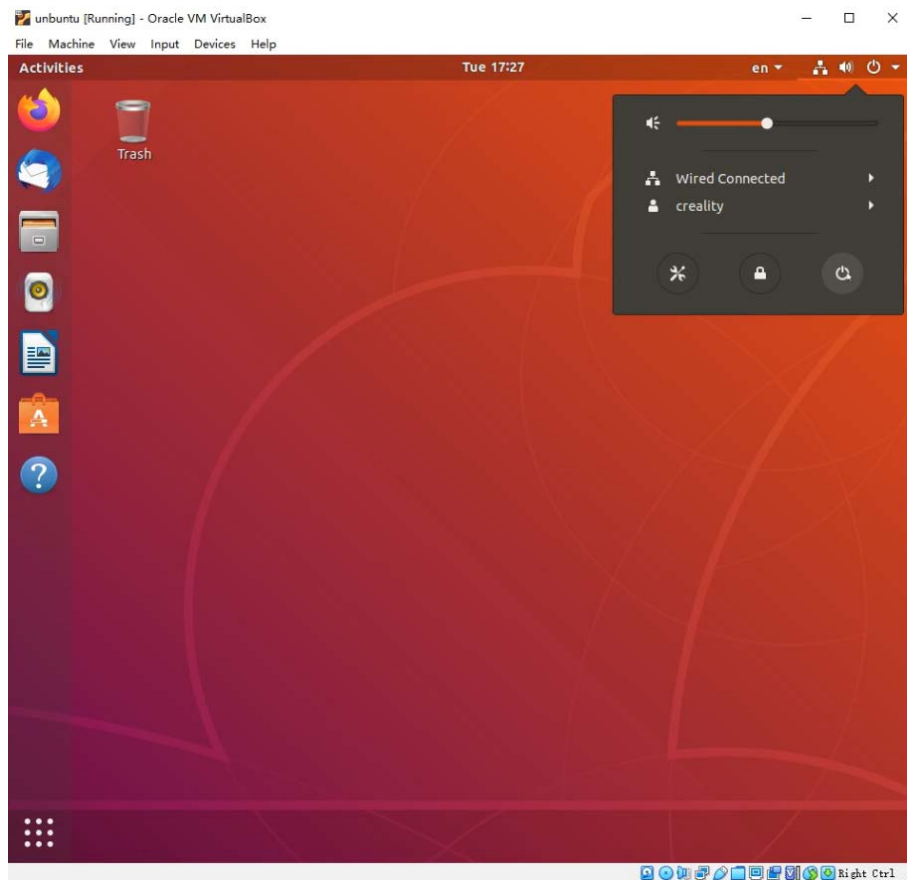
输入系统密码后回车（密码自动隐藏）



安装完成

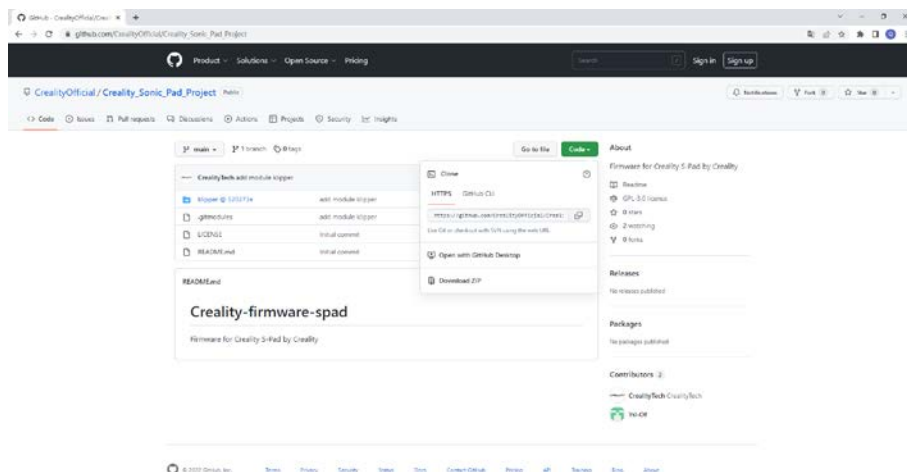


重启系统



5、下载 Klipper 固件

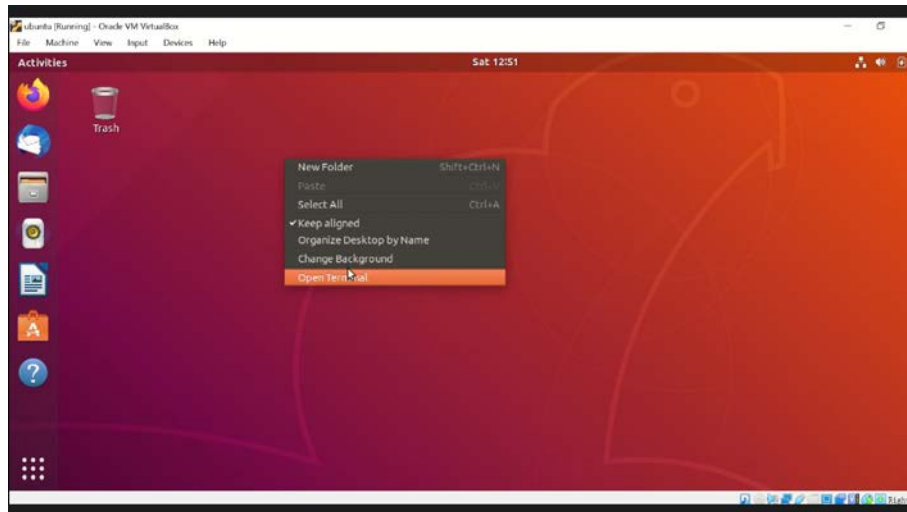
在 [www.github.com/ CrealityOfficial](https://github.com/CrealityOfficial) 复制 Klipper 固件地址链接



固件地址

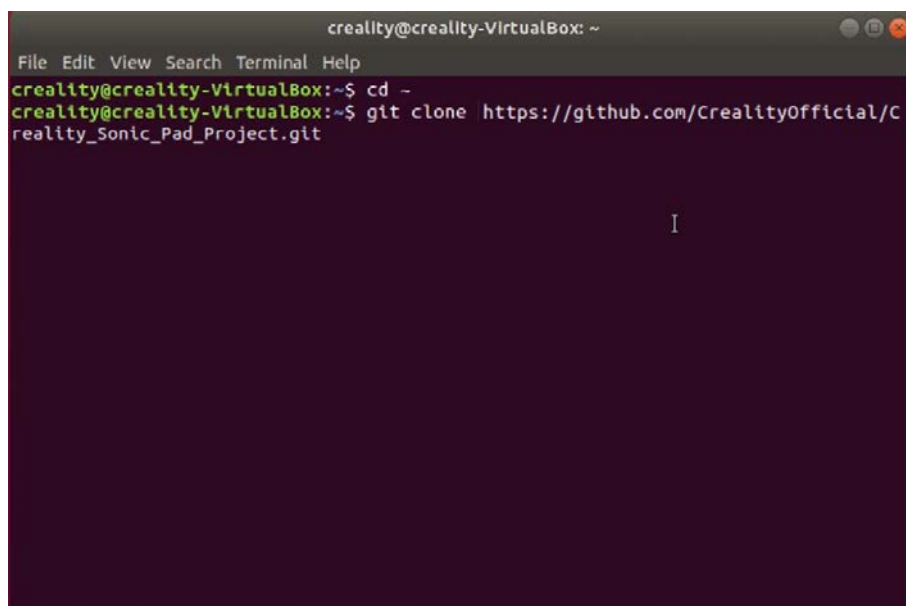
https://github.com/CrealityOfficial/Creality_Sonic_Pad_Project.git

桌面--右键--打开终端



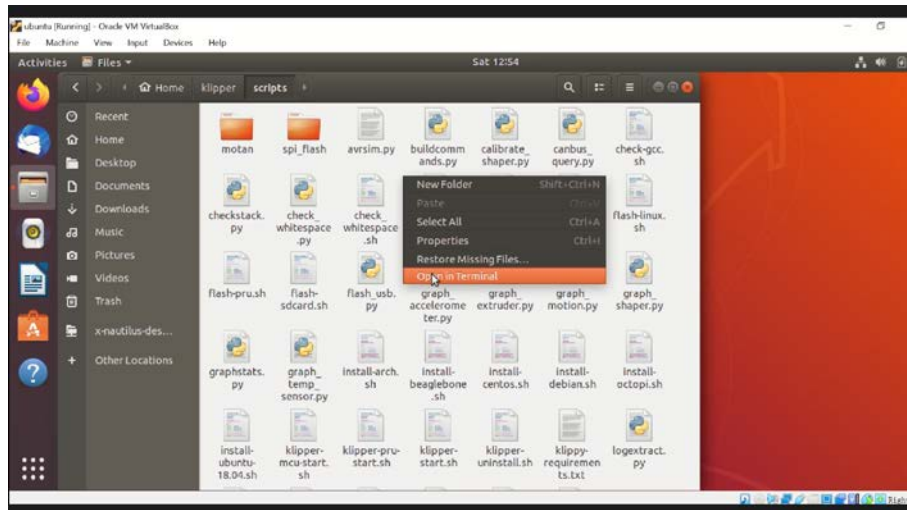
输入“cd ~”后回车（切换工作目录至 home 目录）

输入“git clone”后粘贴 Klipper 固件地址链接--回车



6、配置打印机固件

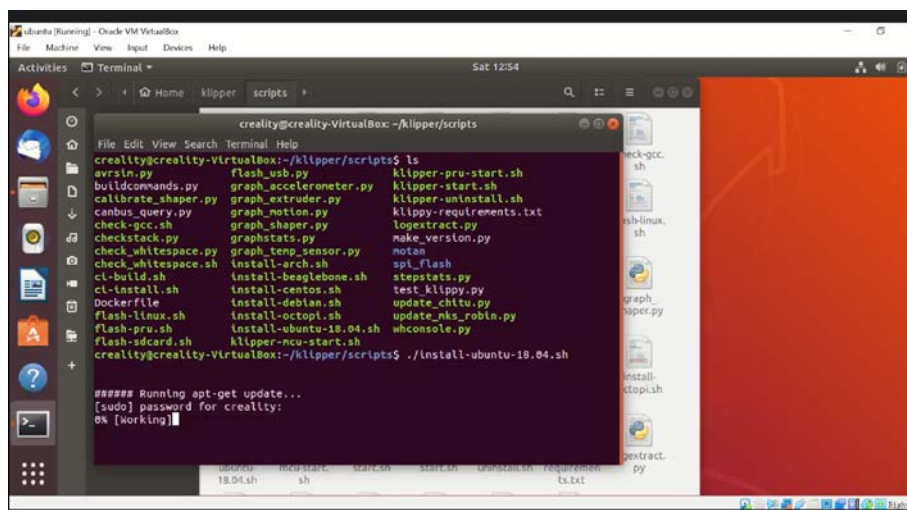
文件管理器--Klipper 文件夹--scripts 文件夹--右键--在终端打开



输入“ls”后回车（确认当前目录下内容与文件夹内一致）

输入“./install-ubuntu-18.04.sh”后回车（安装 ubuntu）

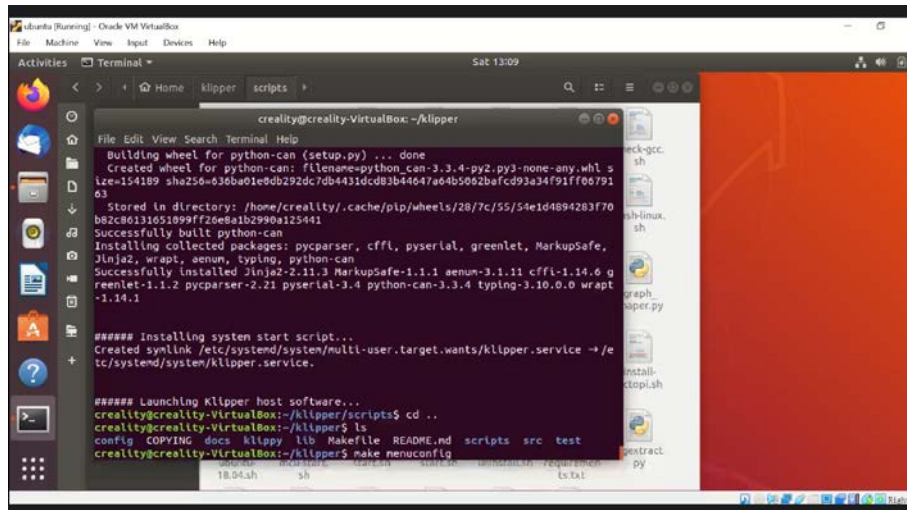
输入系统密码后回车（密码自动隐藏）



输入“cd ..”后回车(切换至上级目录)

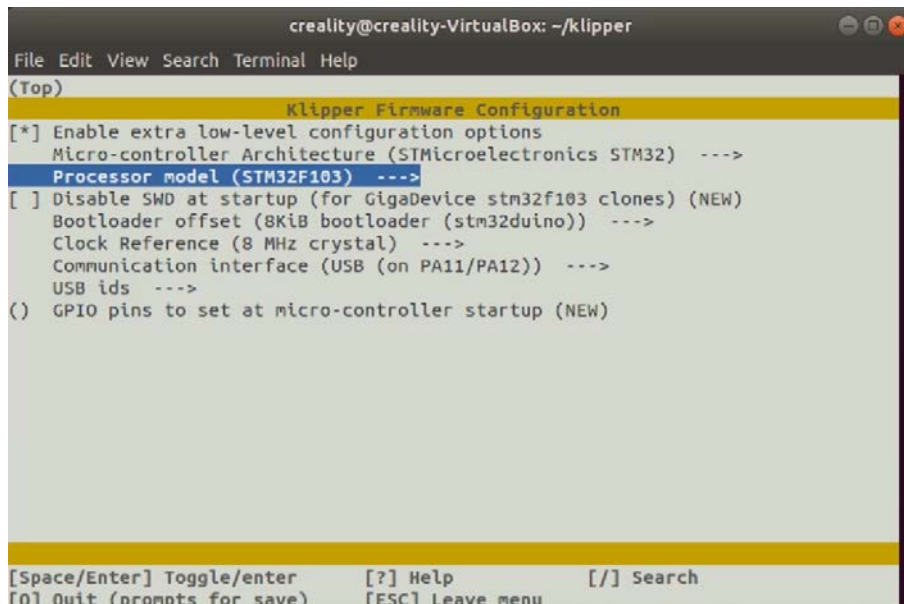
输入“ls”后回车（确认当前目录下内容与文件夹内一致）

输入“make menuconfig”后回车（配置打印机主板）

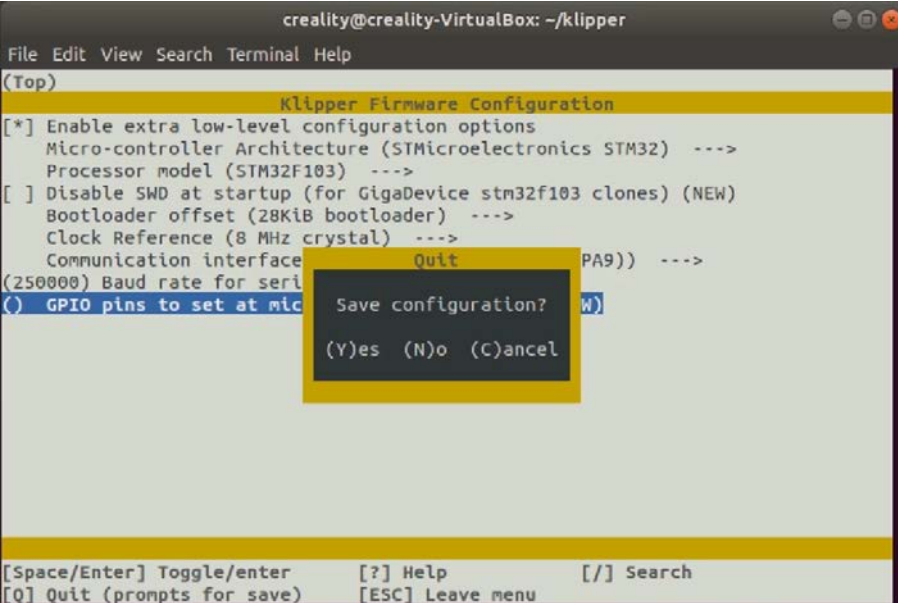


选择芯片结构、芯片型号、Bootloader 偏移、外部时钟、通讯口等参数

(示例，请按照主板配置)

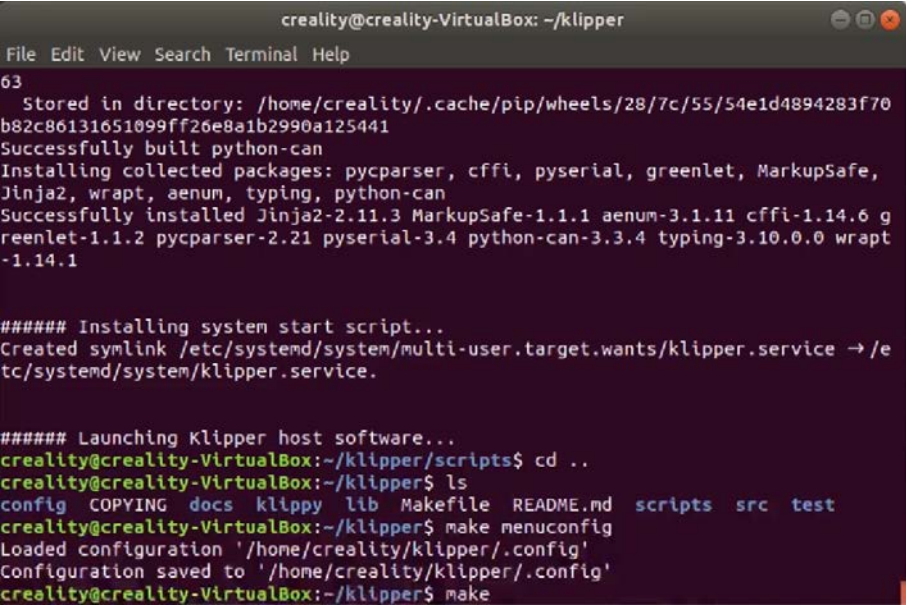


参数配置完成后按“Q”保存，再按“Y”确认



```
creality@creality-VirtualBox: ~/klipper
File Edit View Search Terminal Help
(Top)
Klipper Firmware Configuration
[*] Enable extra low-level configuration options
  Micro-controller Architecture (STMicroelectronics STM32) --->
  Processor model (STM32F103) --->
[ ] Disable SWD at startup (for GigaDevice stm32f103 clones) (NEW)
  Bootloader offset (28KiB bootloader) --->
  Clock Reference (8 MHz crystal) --->
  Communication interface (PA9)) --->
(250000) Baud rate for serial
(Q) GPIO pins to set at mic
Save configuration? (Y)
(Y)es (N)o (C)ancel
[Space/Enter] Toggle/enter    [?] Help    [/] Search
[Q] Quit (prompts for save)    [ESC] Leave menu
```

输入“make”后回车（生成烧录文件）



```
creality@creality-VirtualBox: ~/klipper
File Edit View Search Terminal Help
63
  Stored in directory: /home/creality/.cache/pip/wheels/28/7c/55/54e1d4894283f70
b82c86131651099ff26e8a1b2990a125441
Successfully built python-can
Installing collected packages: pycparser, cffi, pyserial, greenlet, MarkupSafe,
Jinja2, wrapt, aenum, typing, python-can
Successfully installed Jinja2-2.11.3 MarkupSafe-1.1.1 aenum-3.1.11 cffi-1.14.6 g
reenlet-1.1.2 pycparser-2.21 pyserial-3.4 python-can-3.3.4 typing-3.10.0.0 wrapt
-1.14.1

##### Installing system start script...
Created symlink /etc/systemd/system/multi-user.target.wants/klipper.service → /e
tc/systemd/system/klipper.service.

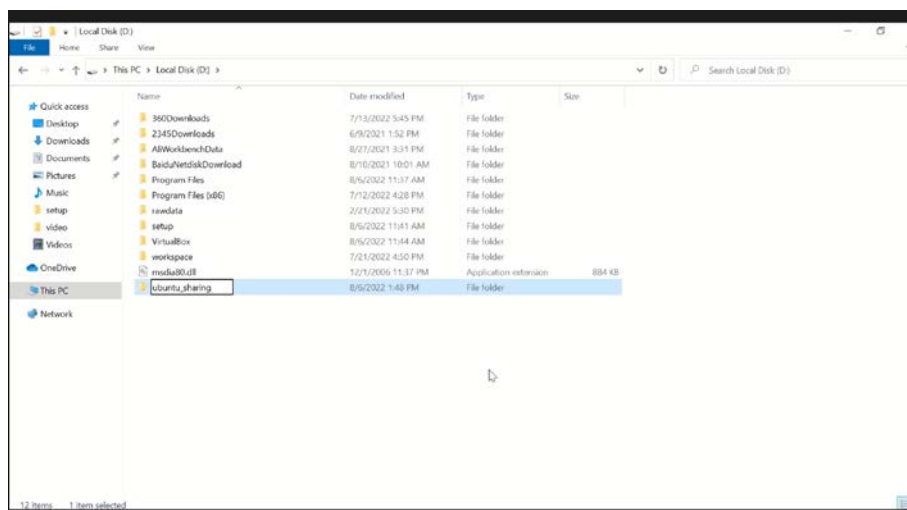
##### Launching Klipper host software...
creality@creality-VirtualBox:~/klipper/scripts$ cd ..
creality@creality-VirtualBox:~/klipper$ ls
config COPYING docs klippy lib Makefile README.md scripts src test
creality@creality-VirtualBox:~/klipper$ make menuconfig
Loaded configuration '/home/creality/klipper/.config'
Configuration saved to '/home/creality/klipper/.config'
creality@creality-VirtualBox:~/klipper$ make
```

找到固件存放位置

```
creality@creality-VirtualBox: ~/klipper
File Edit View Search Terminal Help
Compiling out/src/tmcuart.o
Compiling out/src/neopixel.o
Compiling out/src/pulse_counter.o
Compiling out/src/stm32/watchdog.o
Compiling out/src/stm32/gpio.o
Compiling out/src/generic/crc16_ccitt.o
Compiling out/src/generic/armcm_boot.o
Compiling out/src/generic/armcm_irq.o
Compiling out/src/generic/armcm_reset.o
Compiling out/src/./lib/stm32f1/system_stm32f1xx.o
Compiling out/src/stm32/stm32f1.o
Compiling out/src/generic/armcm_timer.o
Compiling out/src/stm32/adc.o
Compiling out/src/stm32/i2c.o
Compiling out/src/stm32/spi.o
Compiling out/src/stm32/serial.o
Compiling out/src/generic/serial_irq.o
Compiling out/src/stm32/hard_pwm.o
Building out/compile_time_request.o
Version: v0.10.0-169-g520273e5
Preprocessing out/src/generic/armcm_link.ld
Linking out/klipper.elf
Creating hex file out/klipper.bin
creality@creality-VirtualBox:~/klipper$
```

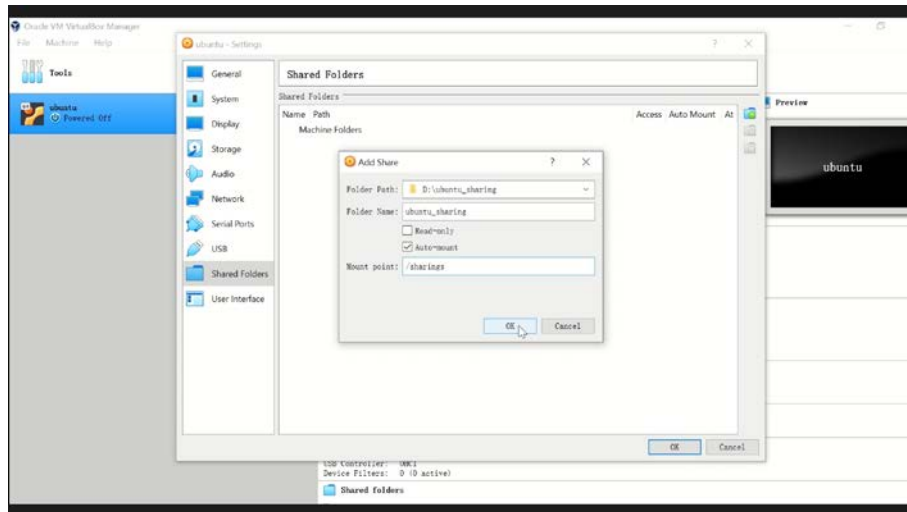
7、通过 SD/TF 卡升级打印机固件

在电脑盘新建“ubuntu_sharing”

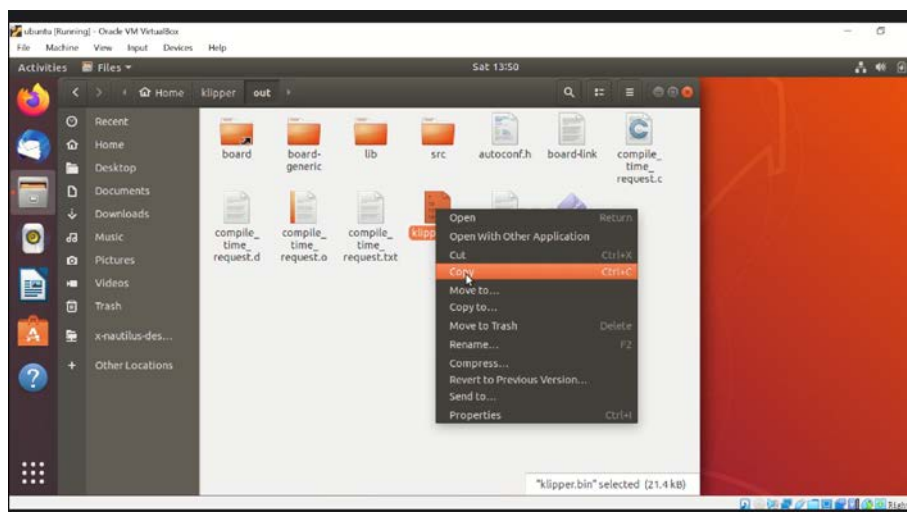


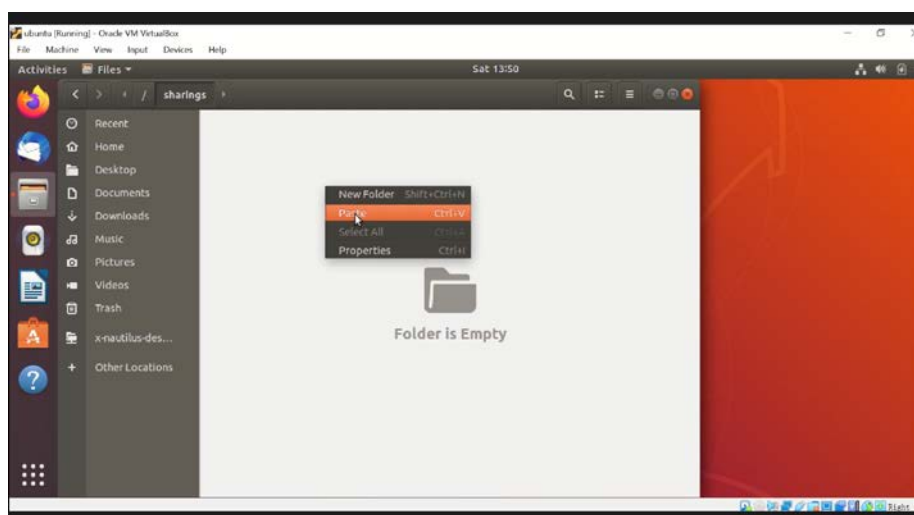
打开 VirtualBox--设置--共享文件夹

添加共享文件夹（勾选自动挂载），挂载点输入“/sharings”

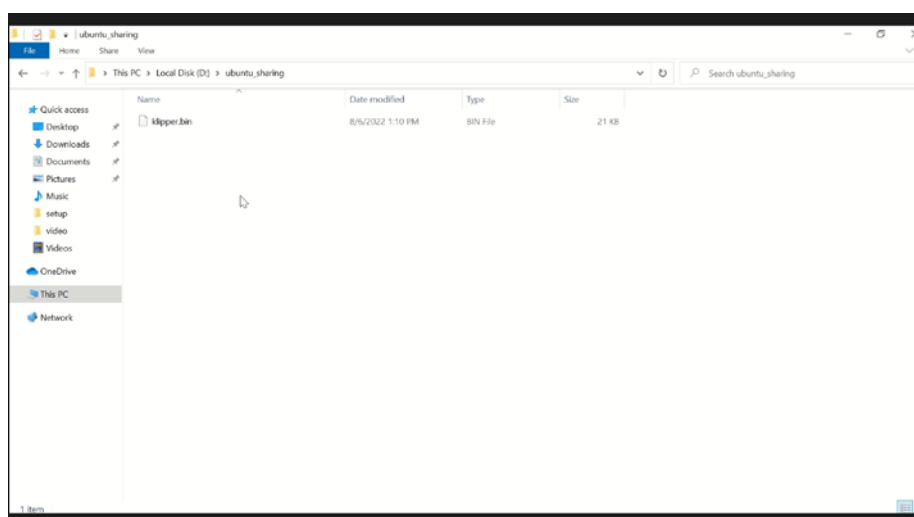


将刚刚生成的 klipper.bin 文件拷贝至 sharings 文件夹





在电脑盘 ubuntu_sharing 文件夹拷贝 klipper.bin 至 TF/SD 卡升级打印机固件



8、配置 USB 升级打印机固件

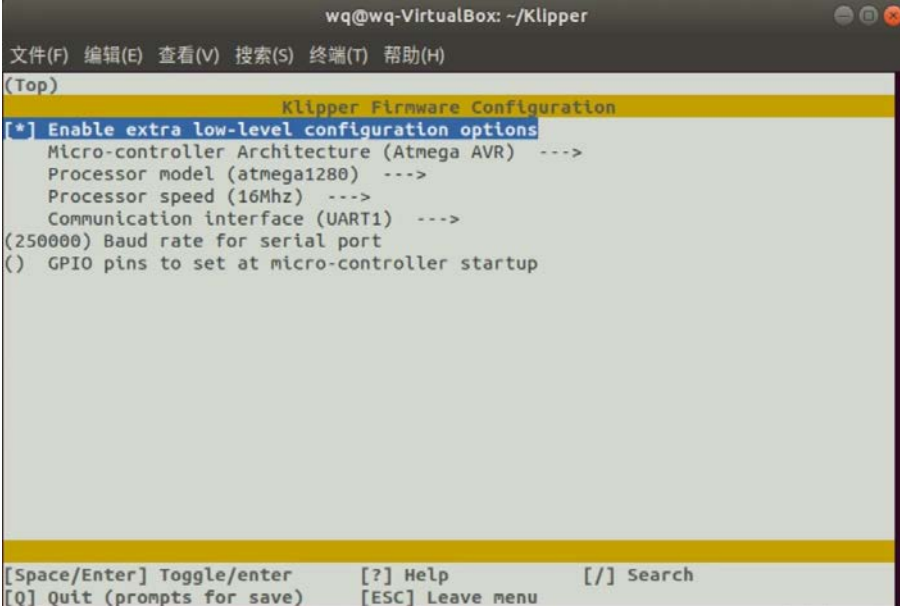
(主板无 Bootloader, 不支持 SD 卡升级)

请先用 USB 数据线将打印机与虚拟机所在的电脑连接,

升级固件过程中请保持连接

参考前面教程进入配置主板界面 (示例, 请按照主板配置)

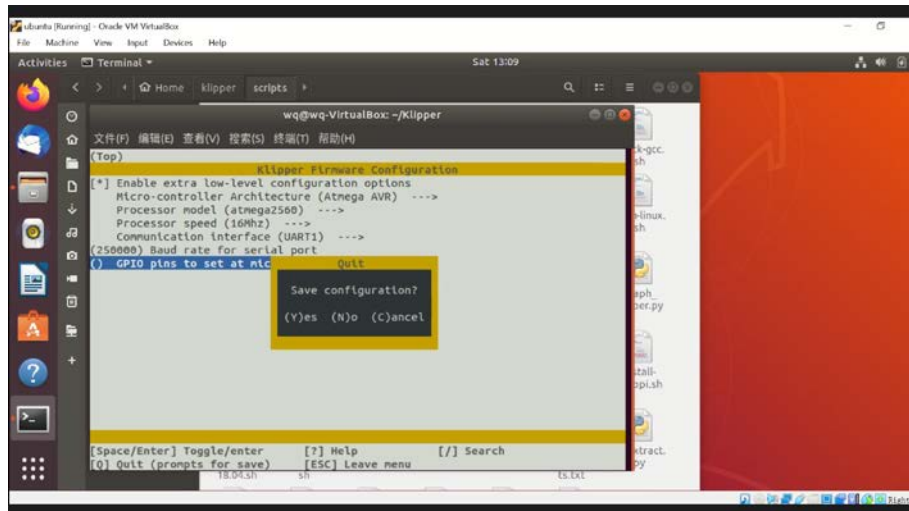
选择芯片结构、芯片型号、Bootloader 偏移、外部时钟、通讯口等参数



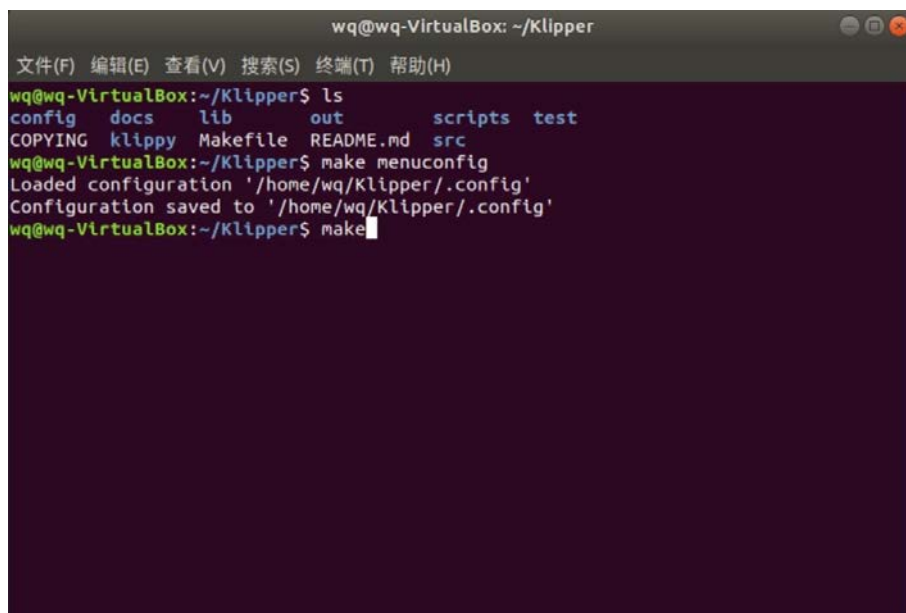
```
wq@wq-VirtualBox: ~/Klipper
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
(Top)
Klipper Firmware Configuration
[*] Enable extra low-level configuration options
  Micro-controller Architecture (Atmega AVR) --->
  Processor model (atmega1280) --->
  Processor speed (16Mhz) --->
  Communication interface (UART1) --->
(250000) Baud rate for serial port
() GPIO pins to set at micro-controller startup

[Space/Enter] Toggle/enter    [?] Help    [/] Search
[Q] Quit (prompts for save)    [ESC] Leave menu
```

参数配置完成后按“Q”保存，再按“Y”确认



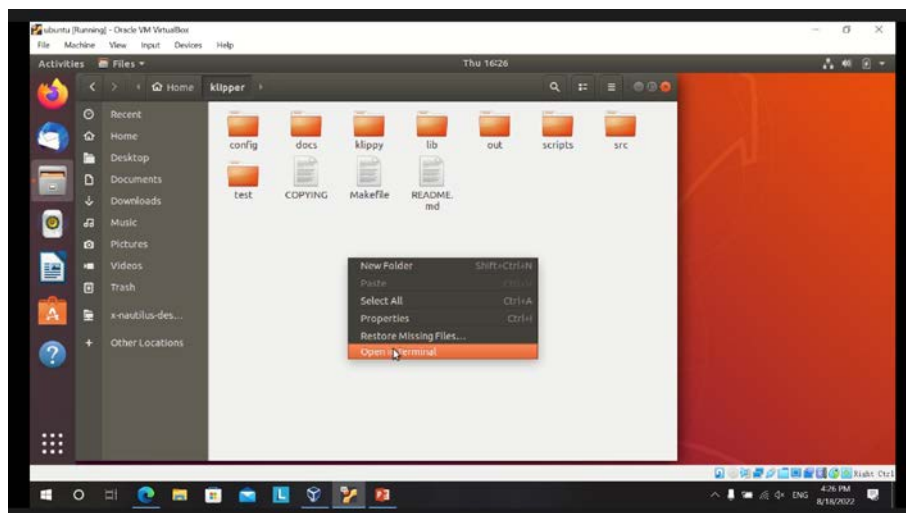
输入“make”后回车（生成烧录文件）



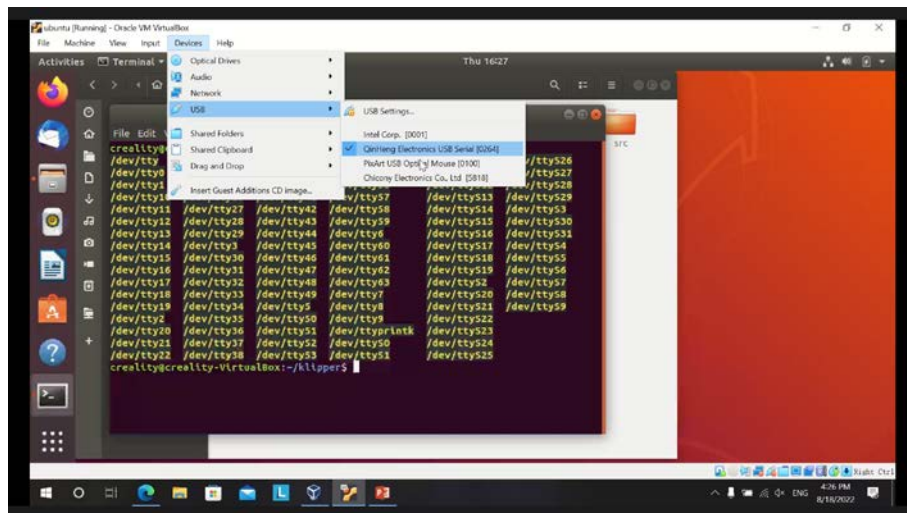
找到固件存放位置

```
wq@wq-VirtualBox: ~/Klipper
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
Compiling out/src/pwmcms.o
Compiling out/src/spi_software.o
Compiling out/src/sensor_adxl345.o
Compiling out/src/lcd_st7920.o
Compiling out/src/lcd_hd44780.o
Compiling out/src/buttons.o
Compiling out/src/tmcuart.o
Compiling out/src/neopixel.o
Compiling out/src/pulse_counter.o
Compiling out/src/avr/main.o
Compiling out/src/avr/timer.o
Compiling out/src/avr/gpio.o
Compiling out/src/avr/adc.o
Compiling out/src/avr/spi.o
Compiling out/src/avr/i2c.o
Compiling out/src/avr/hard_pwm.o
Compiling out/src/avr/watchdog.o
Compiling out/src/avr/serial.o
Compiling out/src/generic/serial_irq.o
Building out/compile_time_request.o
Version: v0.10.0-169-g520273e5-20220913_155730-wq-VirtualBox
Linking out/klipper.elf
Creating hex file out/klipper.elf.hex
wq@wq-VirtualBox:~/Klipper$
```

文件管理器--Klipper 文件夹--右键--在终端打开



设备--USB--勾选与打印机连接的 USB 选项

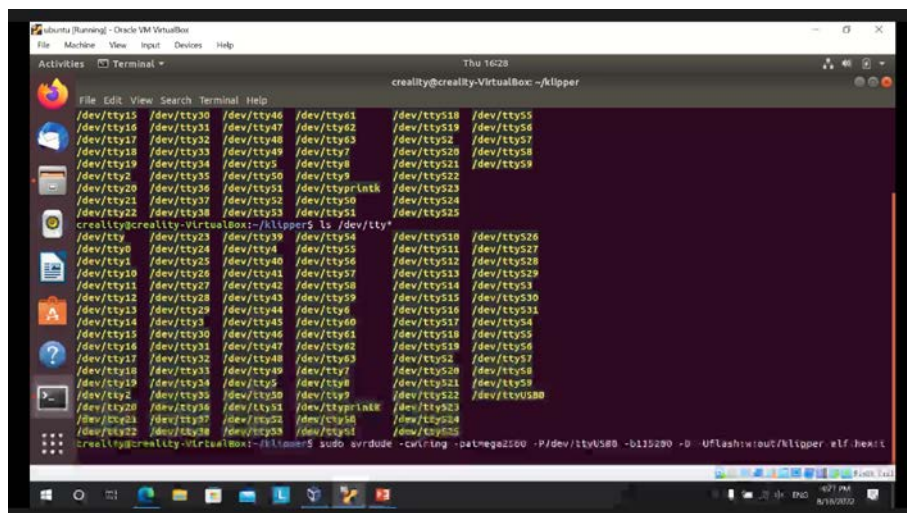


输入“ls /dev/tty*”后回车（确认目录内出现/dev/ttyUSB0）

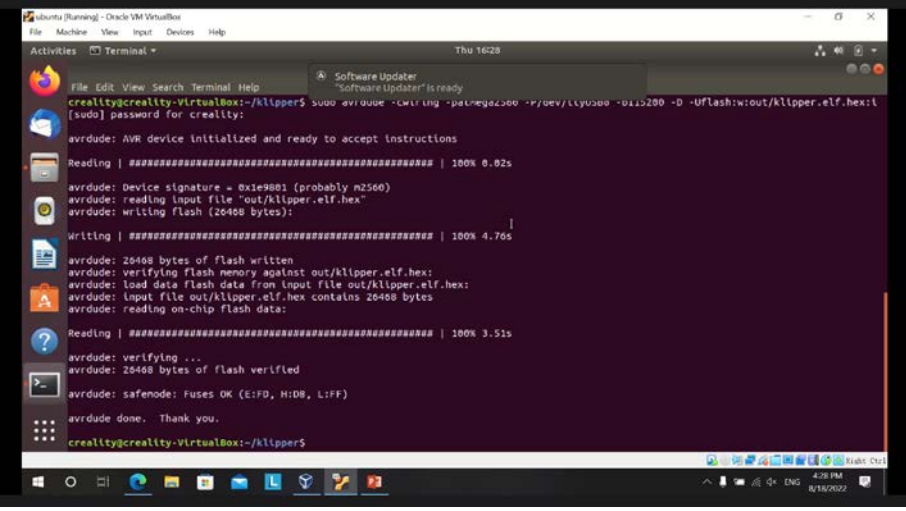
输入

“sudo avrdude -cwiring -patmega2560 -P/dev/ttyUSB0 -b115200 -D -Uflash:w:out/klipper.elf.hex:i”

后回车（将 klipper 固件传输到打印机进行升级，注意命令中“-”前有空格）



升级完成



```
creality@creality-VirtualBox:~/klipper$ sudo avrdude -c wiring -p atmega2560 -r /dev/ttyUSB0 -b 115200 -D -Uflash:w:out/klipper.elf.hex:l
[sudo] password for creality:
avrdude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.02s
avrdude: Device signature = 0x1e9801 (probably m2560)
avrdude: reading input file "out/klipper.elf.hex"
avrdude: writing flash (26468 bytes):
Writing | ##### | 100% 4.70s
avrdude: 26468 bytes of flash written
avrdude: verifying flash memory against out/klipper.elf.hex:
avrdude: load data flash data from input file out/klipper.elf.hex:
avrdude: input file out/klipper.elf.hex contains 26468 bytes
avrdude: reading on-chip flash data:
Reading | ##### | 100% 3.51s
avrdude: verifying ...
avrdude: 26468 bytes of flash verified
avrdude: safenode: Fuses OK (E:FD, H:00, L:FF)
avrdude done. Thank you.
creality@creality-VirtualBox:~/klipper$
```