

从零开始玩转 K230

1. 环境搭建与镜像烧写

1.1 准备工作

k230 sdk 是使用 ubuntu 20.04 编译的。第一步是获得一个 ubuntu 20.04 的环境，该步骤我们可以用 windows 的 wsl 功能，也可以使用虚拟机软件实现。

1.2 虚拟机下载

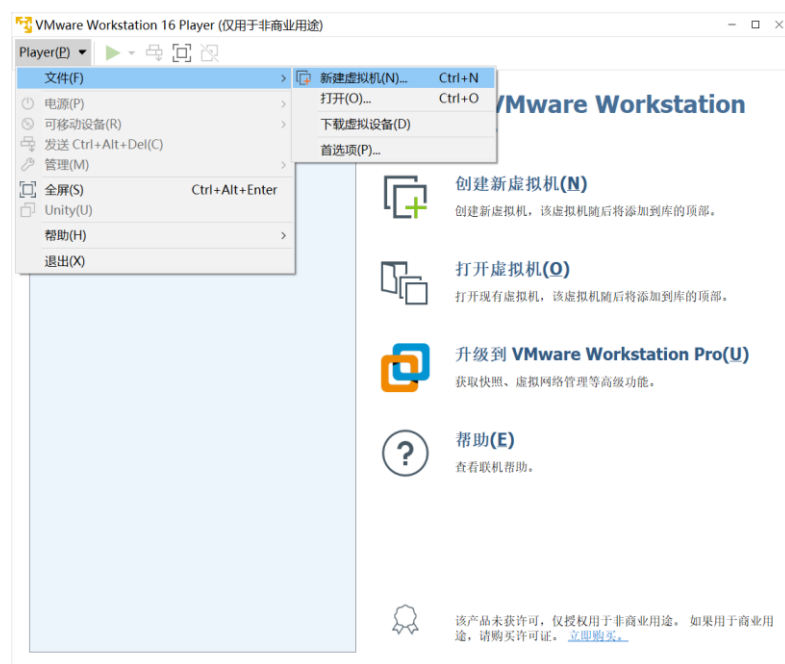
对于学校组织或者个人推荐 [vmware workstation player](#)。

1.3 ubuntu 镜像下载

下载 ubuntu 镜像，如果使用 docker 环境的话，版本不需要必须是 20.04，但是为了避免不必要的麻烦，还是推荐使用我们 20.04. [下载地址](#)

1.4 安装 ubuntu

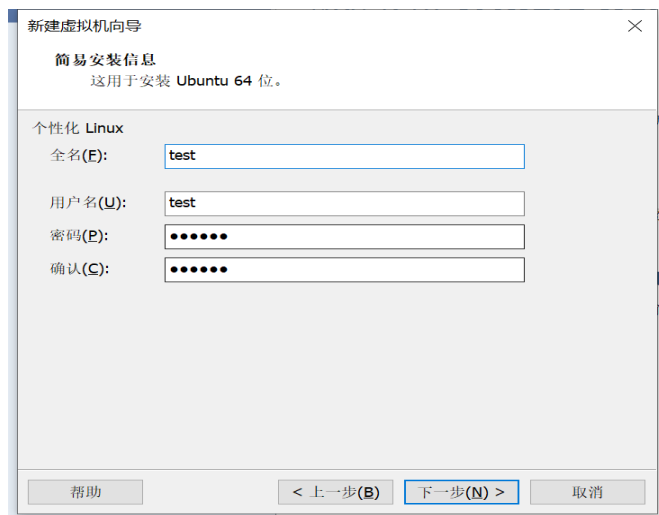
vmware workstation player 选择左上角 player->文件->新建虚拟机选项



点击浏览选中我们刚才下载的 ubuntu 镜像



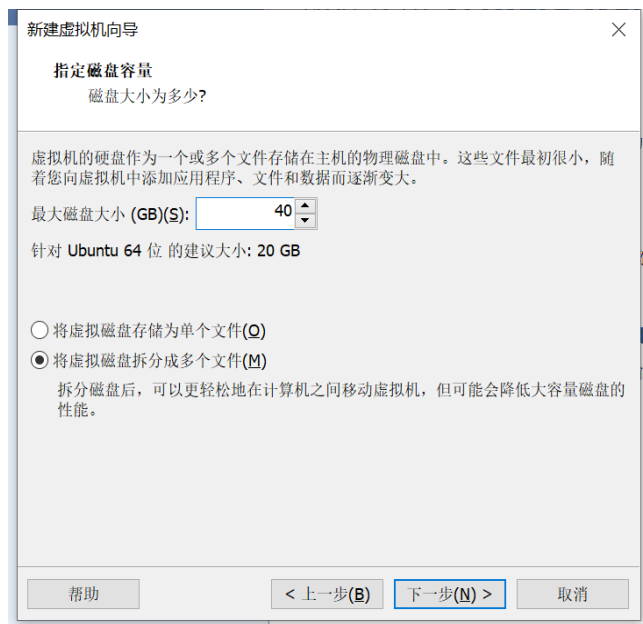
之后点击下一步，填入用户信息，名字可以随便起



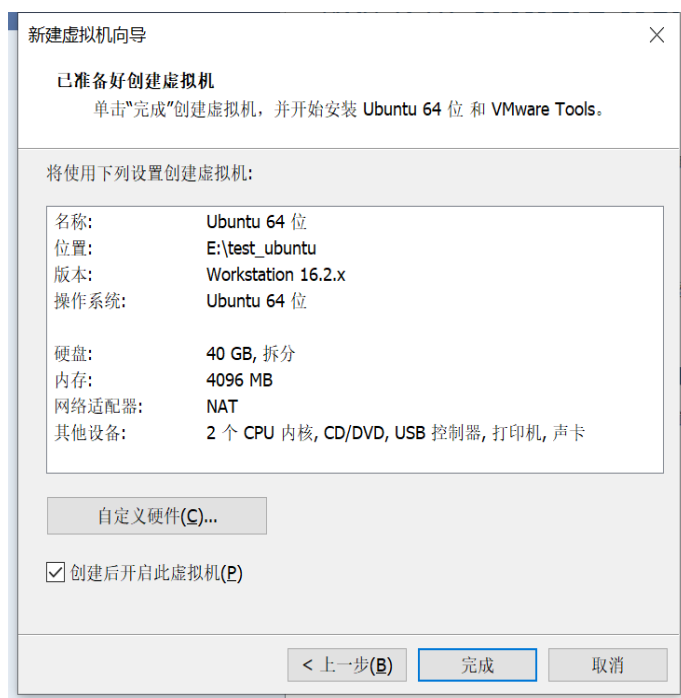
点击下一步选择 ubuntu 使用的磁盘空间位置



点击下一步根据电脑配置和使用需求选择虚拟机的各项配置参数，考虑到我们 sdk 占用的空间大小，这里可以把磁盘容量配置的大一些。



点击下一步，若没有特殊需求，可以直接点击完成开始安装虚拟机



1.5 配置 ubuntu

在虚拟机中输入 `ctrl + alt + T` 打开命令行界面。运行如下命令安装一些必要软件

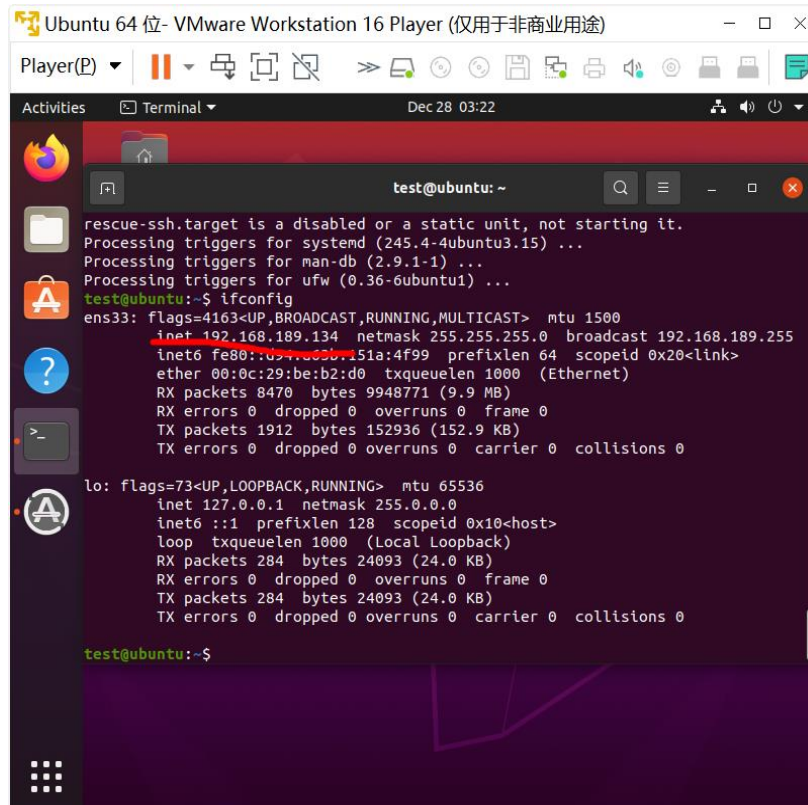
```
sudo apt-get install git docker.io net-tools openssh-server make curl -y
```

- git 用于下载代码
- docker.io 用于使用 sdk 中的 docker 镜像
- net-tools 用于查看虚拟机网络

- openssh-server 用于使用终端工具连接 ubuntu
- make 使用 makefile 工具
- curl 客户端的 URL 工具，我们使用他下载部分 sdk 内容

之后可以使用类似 xshell 或者 mobaxterm 这样的工具通过 ssh 远程连接访问 ubuntu 了，如果电脑上没有安装可以使用 windows 自带的 powershell

- 在 ubuntu 中运行 ifconfig 查看本机 ip

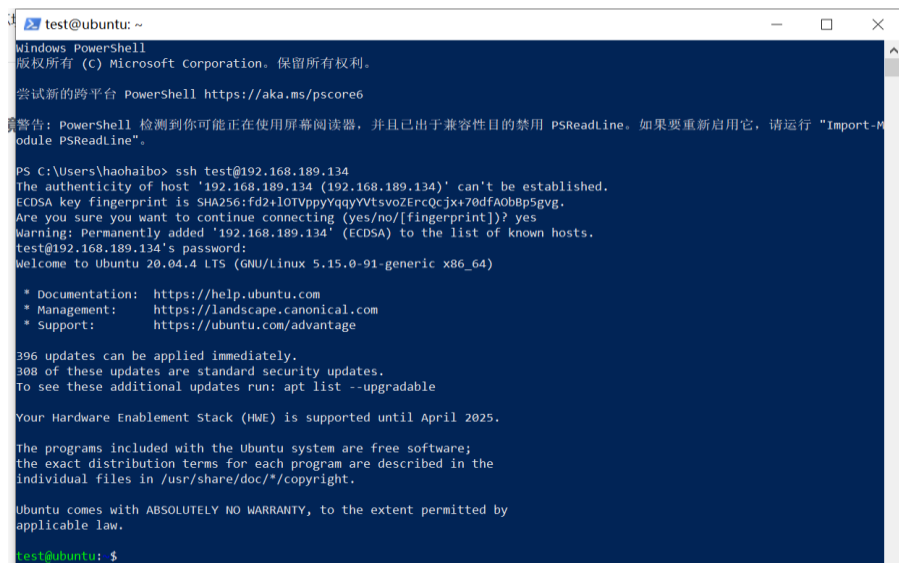


```
Ubuntu 64 位- VMware Workstation 16 Player (仅用于非商业用途)
Player(P)
Activities Terminal Dec 28 03:22
test@ubuntu: ~
rescue-ssh.target is a disabled or a static unit, not starting it.
Processing triggers for systemd (245.4-4ubuntu3.15) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ufw (0.36-6ubuntu1) ...
test@ubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.189.134 netmask 255.255.255.0 broadcast 192.168.189.255
    inet6 fe80::d274:355b:151a:4f99 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:be:b2:d0 txqueuelen 1000 (Ethernet)
    RX packets 8470 bytes 9948771 (9.9 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1912 bytes 152936 (152.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 284 bytes 24093 (24.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 284 bytes 24093 (24.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

test@ubuntu:~$
```

- 在 powershell 中使用 ssh 连接该 ip 地址



```
test@ubuntu: ~
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

警告: PowerShell 检测到你可能正在使用屏幕阅读器，并且已出于兼容性目的禁用 PSReadLine。如果要重新启用它，请运行 "Import-Module PSReadLine"。

PS C:\Users\haohaibo> ssh test@192.168.189.134
The authenticity of host '192.168.189.134 (192.168.189.134)' can't be established.
ECDSA key fingerprint is SHA256:fd2+10TVppyYqyYVtsvoZErCQcJx+70dFA0bBp5gvg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.189.134' (ECDSA) to the list of known hosts.
test@192.168.189.134's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.15.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

396 updates can be applied immediately.
308 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

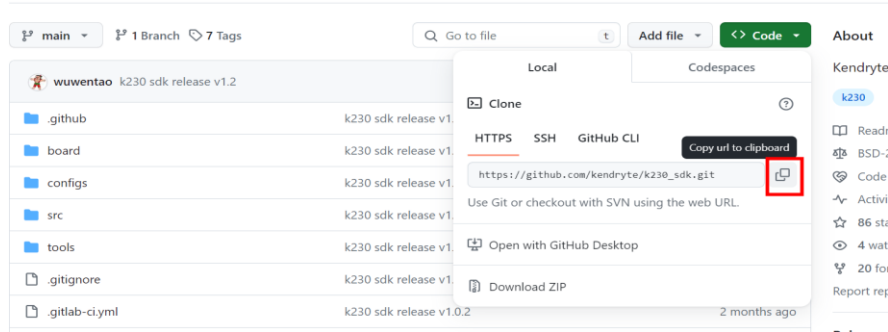
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

test@ubuntu:~$
```

```
ssh test@192.168.189.134
```

1.6 下载 sdk

登录 github 网站，点击红框标注的位置复制代码链接



回到 powershell 界面输入以下命令下载 sdk

```
test@ubuntu:~$ git clone https://github.com/kendryte/k230_sdk.git
```

```
test@ubuntu:~$ git clone https://github.com/kendryte/k230_sdk.git
Cloning into 'k230_sdk'...
remote: Enumerating objects: 109987, done.
remote: Counting objects: 100% (491/491), done.
remote: Compressing objects: 100% (419/419), done.
Receiving objects: 8% (9089/109987), 6.32 MiB | 1.66 MiB/s
```

等待代码全部拉取完毕后开始配置 k230 的编译环境

1.7 编译环境配置

根据 https://github.com/kendryte/k230_sdk 的 Readme 文件说明配置 docker 环境。

- 获取 docker 镜像

```
sudo docker pull ghcr.io/kendryte/k230_sdk
```

```
test@ubuntu:~$ sudo docker pull ghcr.io/kendryte/k230_sdk
Using default tag: latest
latest: Pulling from kendryte/k230_sdk
30ecab32a3b6: Pull complete
013d1247f63b: Pull complete
20d31a1c3162: Pull complete
ed14d53bcc92: Pull complete
fb346cd221c2: Pull complete
2c351203f2d0: Pull complete
9b58527ce7e5: Pull complete
a859c95137f0: Pull complete
921a0f332238: Pull complete
03d77e3eeb0f: Pull complete
4f4fb700ef54: Pull complete
ed273b09c244: Pull complete
217f10d73712: Pull complete
cae42310c443: Pull complete
ac59987d74fe: Pull complete
cf738dba42d1: Pull complete
Digest: sha256:ea828ec75ad448c46aaa4afa222912ec97619b51ac81779ba550f26266a07836
Status: Downloaded newer image for ghcr.io/kendryte/k230_sdk:latest
ghcr.io/kendryte/k230_sdk:latest
```

编译 SDK

进入 sdk 目录，运行如下命令下载其他必要的工具和库文件等内容

```
cd k230_sdk
make prepare_sourcecode
```

```
test@ubuntu:~/k230_sdk$ make prepare_sourcecode
download toolchain
riscv64-unknown-linux-musl-rv64imafdcv-lp64d-20230420.tar.bz2 100%[=====>] 103.34M 2.72MB/s
in 32s
xuantie-900-gcc-linux-5.10.4-glibc-x86_64-V2.6.0.tar.bz2 100%[=====>] 429.04M 3.89MB/s
in 1m 59s
extract toolchain
./toolchain/xuantie-900-gcc-linux-5.10.4-glibc-x86_64-V2.6.0.tar.bz2
./toolchain/riscv64-unknown-linux-musl-rv64imafdcv-lp64d-20230420.tar.bz2
toolchain is ready
prepare source code
download nncase sdk
100%[=====>] 311.82M 3.74MB/s
in 85s
100%[=====>] 44.84M 2.94MB/s
in 15s
download big utils
100%[=====>] 73.55M 3.23MB/s
in 22s
/home/test/k230_sdk
download little firmware
./src/little/utlis/firmware/AIW4211L_demo_allinone.bin 100%[=====>] 679.54K 2.54MB/s
in 0.3s
download tuning server
src/little/buildroot-ext/package/tuning-server/tuning-server-pa 100%[=====>] 2.63M 2.92MB/s
in 0.9s
kendryte_ISP_Tool_TuningClient_RC22.5_Pre_596062-20221116.7z 100%[=====>] 48.27M 3.15MB/s
in 15s
download buildroot dl
100%[=====>] 378.06M 3.15MB/s
in 1m 58s
test@ubuntu:~/k230_sdk$
```

运行 docker 命令准备编译

```
sudo docker run -u root -it -v $(pwd):$(pwd) -v
$(pwd)/toolchain:/opt/toolchain -w $(pwd) ghcr.io/kendryte/k230_sdk
/bin/bash
```

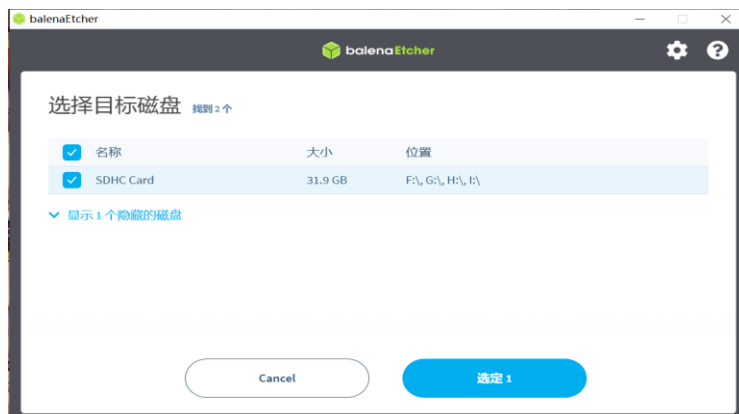
运行完该命令后会就进入新的命令行环境中

```
root@0c0b95c97d0f:/home/test/k230_sdk#
```

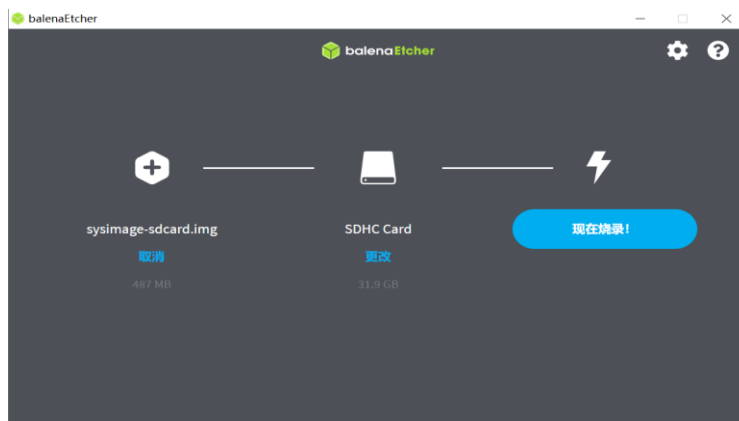
之后根据手中的开发板类型编译 sdk 即可，我们目前使用的是 k230 CanMV 开发板所以使用如下命令编译

```
CanMV
make CONF=k230_esp_defconfig
```

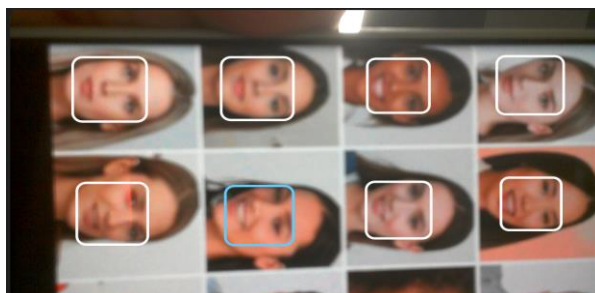
```
INFO: hdimage(sysimage-sdcard.img): adding partition 'uboot_spl_1' from 'little-core/uboot/fn_u-boot-spl.bin' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'uboot_spl_2' from 'little-core/uboot/fn_u-boot-spl.bin' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'uboot_env' from 'little-core/uboot/env.env' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'uboot' from 'little-core/uboot/fn_ug_u-boot.bin' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'rtt' (in MBR) from 'big-core/rtt_system.bin' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'linux' (in MBR) from 'little-core/linux_system.bin' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'rootfs' (in MBR) from 'little-core/rootfs.ext4' ...
INFO: hdimage(sysimage-sdcard.img): adding partition 'fat32appfs' (in MBR) from 'app.vfat' ...
INFO: hdimage(sysimage-sdcard.img): writing GPT
INFO: hdimage(sysimage-sdcard.img): writing hybrid MBR
root@0c0b95c97d0f:/home/test/k230_sdk#
```

选择现在烧录，等待烧录完成



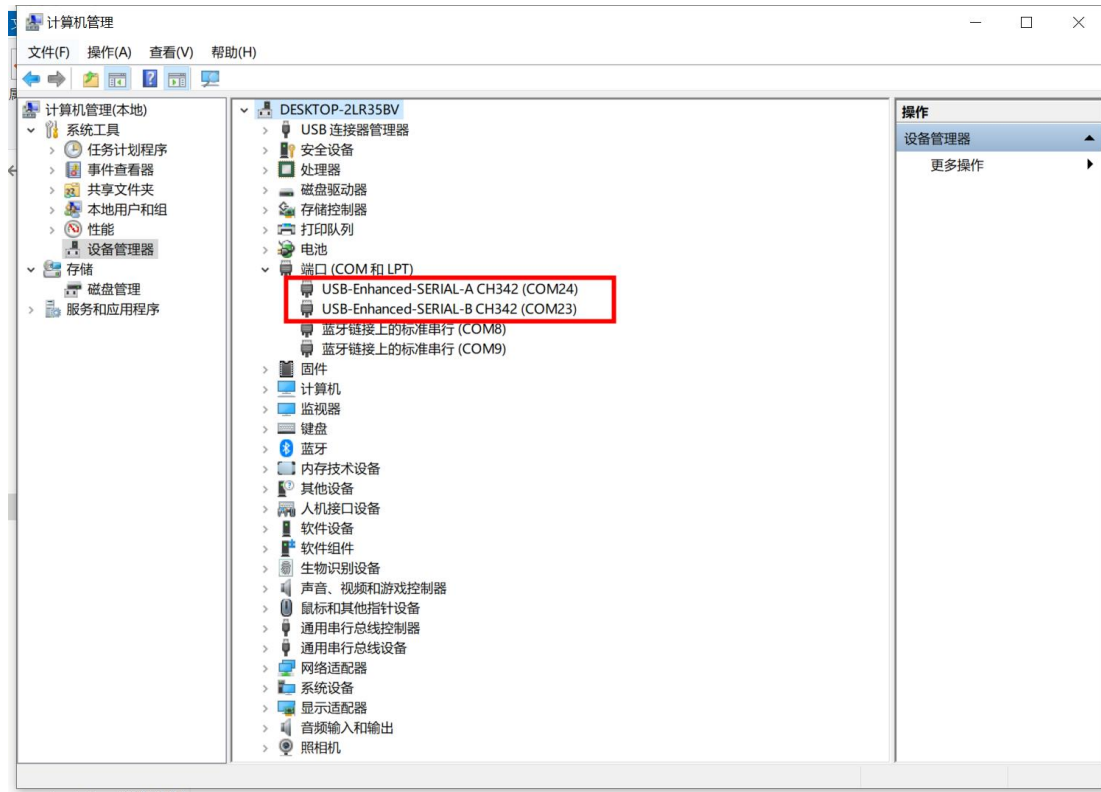
之后将 sd 卡插入 CanMV 开发板，上电重启。默认会运行一个人脸识别的 demo 程序。如下图所示



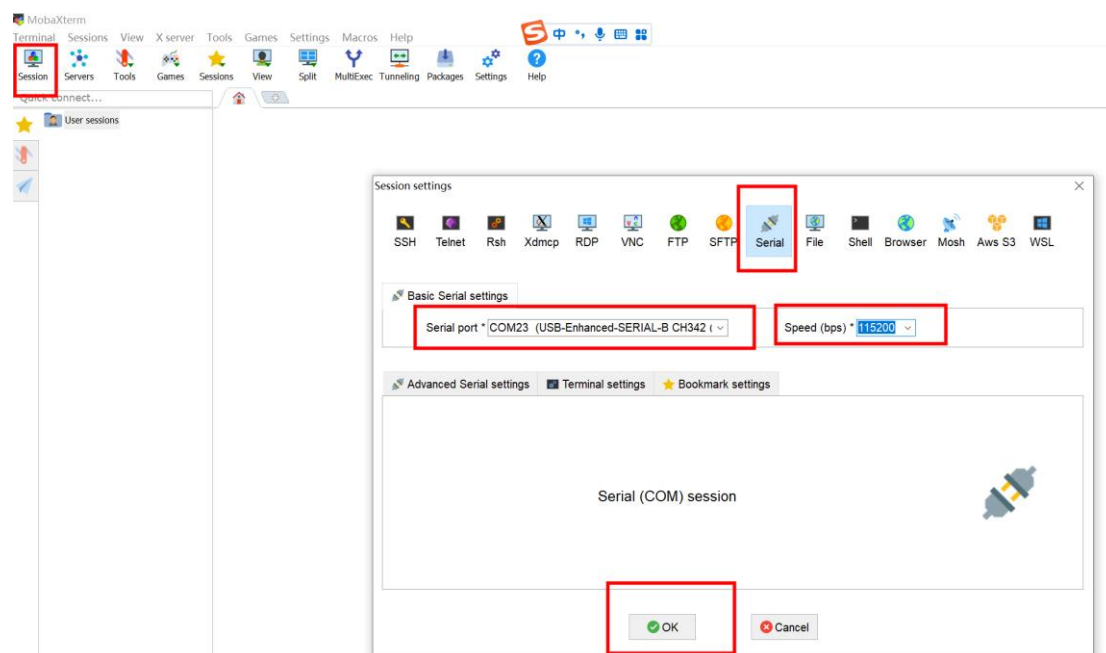
1.9 通过串口操作开发板

准备一个串口工具，推荐使用 [mobaxterm](#)

CamMV 上电运行后到设备管理器内找到查询串口号。



使用 mobaxterm 连接串口



之后就可以使用串口操作大小核的命令行了

