

AELOS SMART 基础使用手册

目录

1 Aelos 桌面软件介绍	3
1.1 Aelos 桌面软件下载	3
1.2 软件功能介绍	3
1.3 软件使用须知	4
2. 网络配置及远程连接	7
2.1 下载 ssh 连接软件	7
2.2 操作步骤	9
2.3 注意事项	13
3. 执行 python 代码进行运动控制	14
3.1 编辑动作	14
3.2 下载动作	18
3.3 执行动作	18
4. 机器人 ROS 启动	22
4.1 ROS 简介	22
4.2 运行 ROS 节点的操作步骤	22

1 Aelos 桌面软件介绍

通过 Aelos 软件，可以实现对机器人的调试、编程、手柄控制等功能，是使用机器人必备软件。因此，本部分将对 Aelos 软件的使用作系统性的介绍，帮助您更简单地使用软件。

1.1 Aelos 桌面软件下载

- 点击下载链接：<http://www.lejurobot.com/support-cn/>，进入网站。
- 选择与电脑系统匹配的安裝程序，点击“下载支持”，并完成下载及安装。

教育版本		
Aelos lite机器人PC端教育版使用说明书 Aelos lite机器人PC端教育版使用说明书	01 DEC 2019	下载支持
Aelos Pro机器人PC端教育版使用说明书 Aelos Pro机器人PC端教育版使用说明书	01 DEC 2019	下载支持
Aelos 机器人PC端教育版安装程序 Windows 1.13.1版本 (Windows OS)	08 Apr 2021	下载支持
Aelos 机器人PC端教育版安装程序 MAC 1.13.1版本 (Mac OS)	08 Apr 2021	下载支持
Aelos手柄使用指导 Aelos手柄使用指导	01 JUNE 2019	下载支持
Aelos手爪装配说明 Aelos手爪装配说明	01 JUNE 2019	下载支持

1.2 软件功能介绍

Aelos 桌面软件可分为五个部分，分别是菜单栏、指令栏、编辑区、动作视图、机值视图区。



- 菜单栏提供多种类的软件功能，可与机器人进行联动使用。
- 指令栏中包括多种类型的指令，都是已经编写好的内容。可以将积木块按住拖动到编辑区，编写程序。
- 编辑区是编写程序的主要阵地，指令的添加、删除，程序的整体设计都在编辑区中进行，在这里能看到程序的整体情况。
- 动作视图可以显示每个动作的详细信息，例如各舵机角度值、速度、刚度、搭配的音乐等。这些信息以条状记录进行显示，可以显示单一动作或一个动作指令里的一组动作。动作视图中也可以对所显示的动作进行预览、修改、删除或者将整组动作打包成一个新的模块。
- 机值视图区是显示当前机器人身上各个舵机的旋转数值的区域。在机值视图中，机器人身体的各个关节处都标有舵机的编号，每个标号下方所显示的就是该舵机的数值。我们可以在机值视图对这些舵机值进行调整。

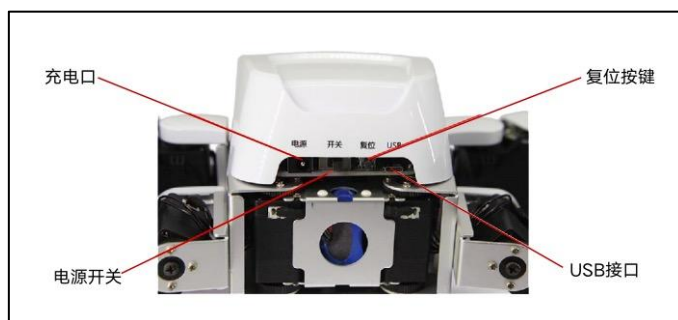
1.3 软件使用须知

1.3.1 新建文件需选择机器人型号

- 若是新建文件，打开 Aelos 桌面软件后，请点击菜单栏的“新建”按钮。弹出选择机器人型号的窗口，选择“Aelos Pro”。
- 若是打开之前的文件夹，则直接点击菜单栏的“打开”并进行选择即可。

1.3.2 串口连接

- 在使用“下载”、“动作下载”、“导入动作”等功能时，都会提示“请先连接机器人”，连接机器人即表示串口连接。
- 连接机器人需用 USB 数据线将机器人与电脑进行串口连接。连接后，需在在菜单栏的“串口”下拉菜单中选择对应的设备，即可连接。（请让电脑同时只连接一台机器人，这样方便选择对应机器人串口）



1.3.3 设置信道

- 若想使用手柄控制机器人，需将机器人与手柄的信道值设置一致。若不一致，需重新设置。信道设置方法如下：
- 机器人信道设置步骤：

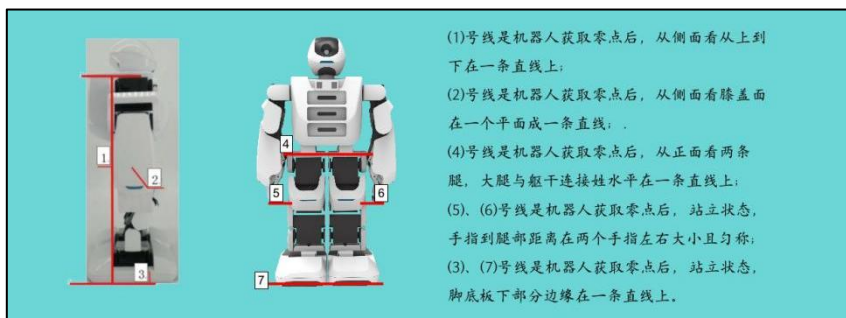


- 遥控器信道设置步骤：

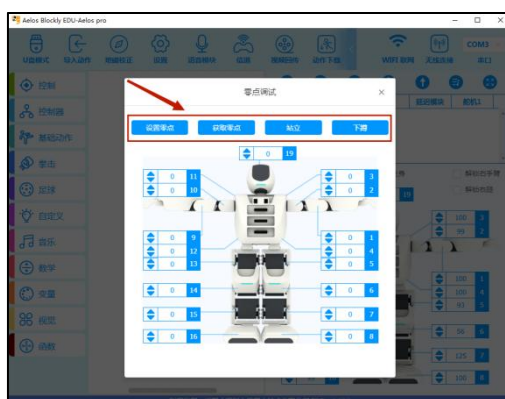


1.3.4 零点调试

- 若是初次使用机器人，推荐使用“零点调试”功能，检查机器人是否为标准姿态。
- 若机器人本身已达到零点标准姿态，则无需调试。若未达到标准姿态，则根据实际情况进行调试。



- 调试步骤：点击菜单栏“设置”按钮，再点击弹出窗口中的“零点调试”，进入零点调试界面。在零点调试界面中点击获取零点可以读取当前机器人的零点值。可以通过调节各舵机框中的数值来调整对应舵机的零点值，调节结束后点击设置零点即可下载到机器人中。可以通过改变机器人姿势（站立、下蹲）来观察调试效果。



2. 网络配置及远程连接

- 镜像基于 Ubuntu MATE 20 系统制作
- 程序基于 ROS 和 Python3 实现，镜像中已经内置基本环境

2.1 下载 ssh 连接软件

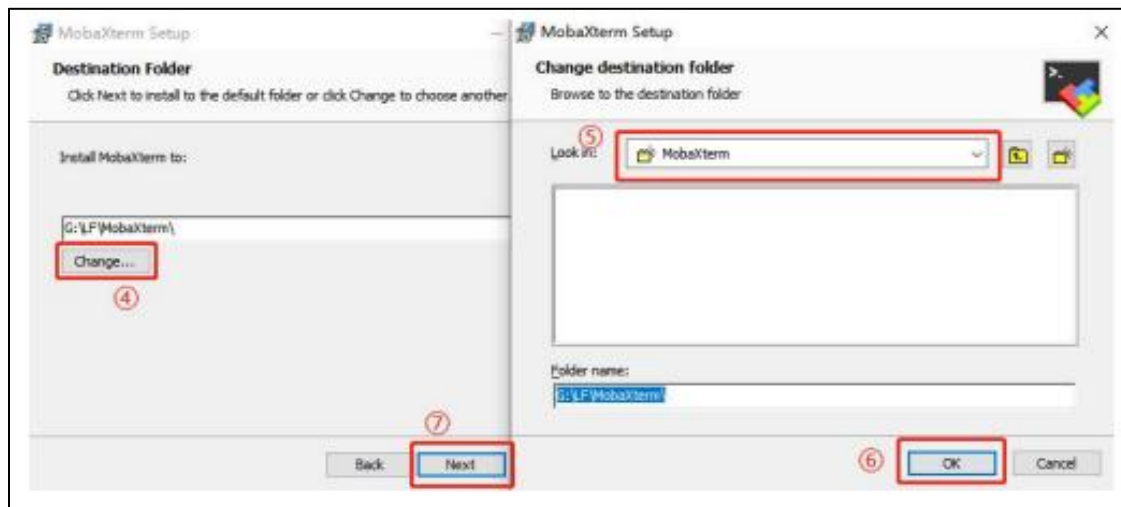
- MAC 用户可以使用系统自带的“终端”软件。
- Windows 用户可以下载 MobaXterm，用该软件进行连接。

（下载链接：<https://mobaxterm.mobatek.net/>）

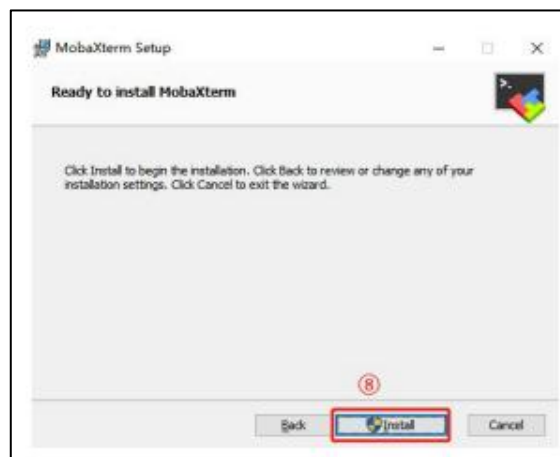
- 打开下载完毕的安装包：



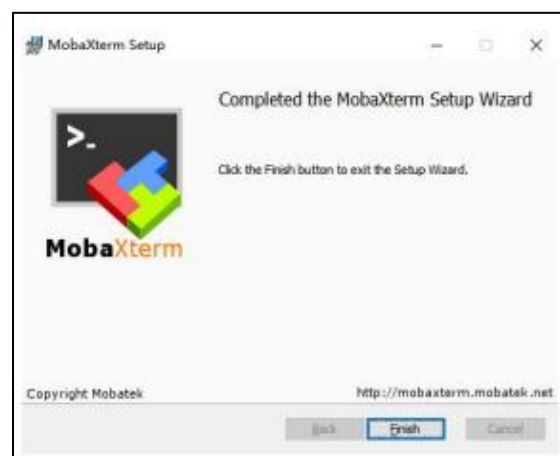
- 选择安装路径:



- 点击“安装”:

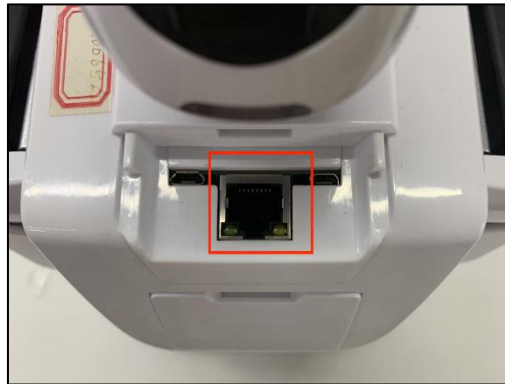


- 安装完成:



2.2 操作步骤

- 启动机器人，使用网线连接树莓派网口。将此网线与电脑连接至同一个网络。



（备注 1：在网络配置过程中，应全程保持机器人开机状态。为避免机器人断电关机，请提前为机器人充满电）

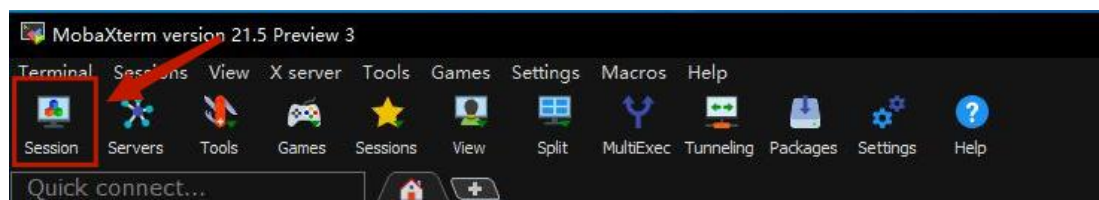
（备注 2：请使用一个路由器，同时连接电脑及机器人）

（备注 3：树莓派网口位于机器人下巴下方，用力下压滑盖同时向外抽出即可打开。）

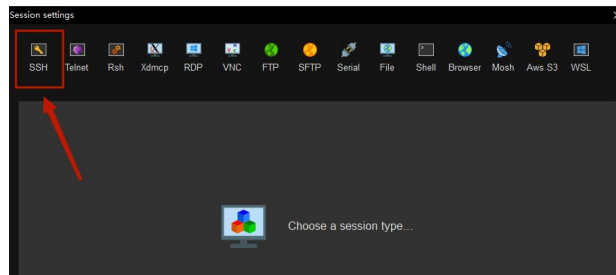
- 进入路由器管理界面，找到名称为 lemon 的设备，该设备就是机器人。请记住机器人 IP 地址，之后进行 ssh 连接需要使用。



- 打开之前下载好的 MobaXterm，点击菜单栏的“session”。

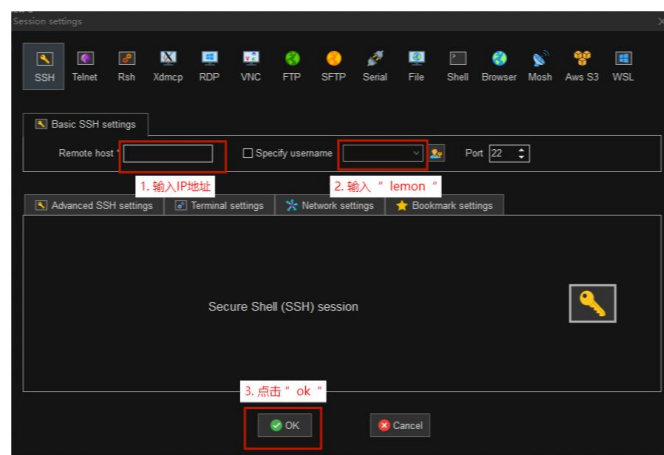


- 弹出 setting 窗口，点击 “SSH”。

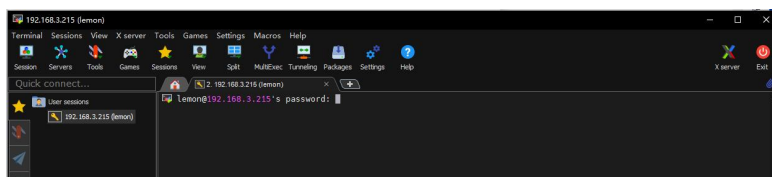


- 在 SSH 窗口中输入 “IP 地址” 及 “lemon”

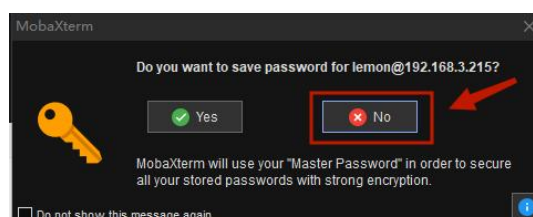
（备注：IP 地址在之前的路由器管理界面可以找到）



- 输入密码： **leju123** 。（注意：输入过程中窗口不会显示密码，请仔细输入）



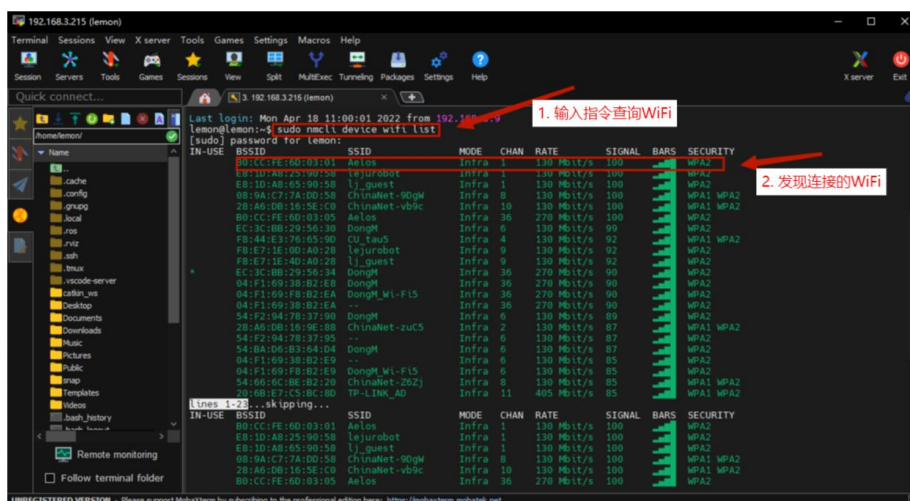
- 密码输入成功后，弹出是否储存密码的窗口，推荐选择 “No”，之后每次登录的时候需输入密码： **leju123** 。若想记住密码，可选择 “Yes”，再按流程操作。



- 成功登录后，输入以下指令查询周围可用 Wi-Fi：

`sudo nmcli device wifi list`

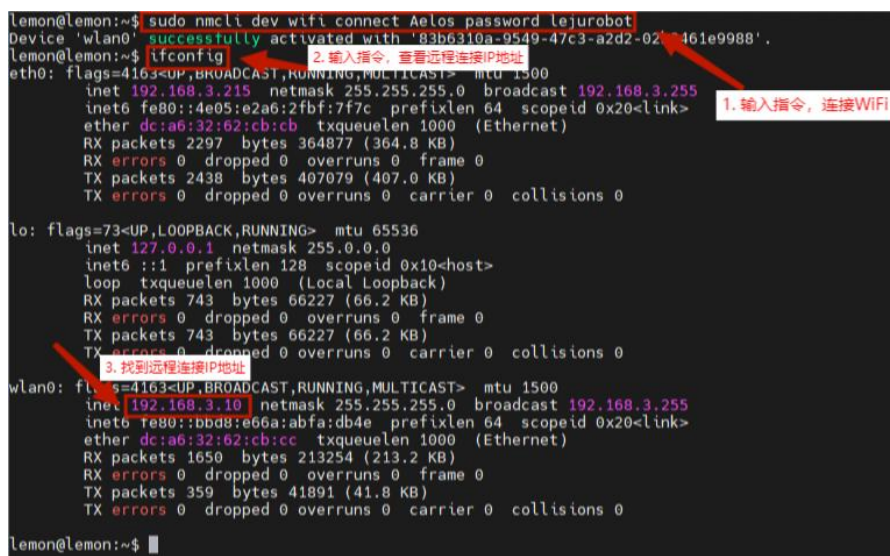
- 之后，弹出 WiFi 列表，可以发现之前电脑和机器人连接的 WiFi 名，如图中示例。



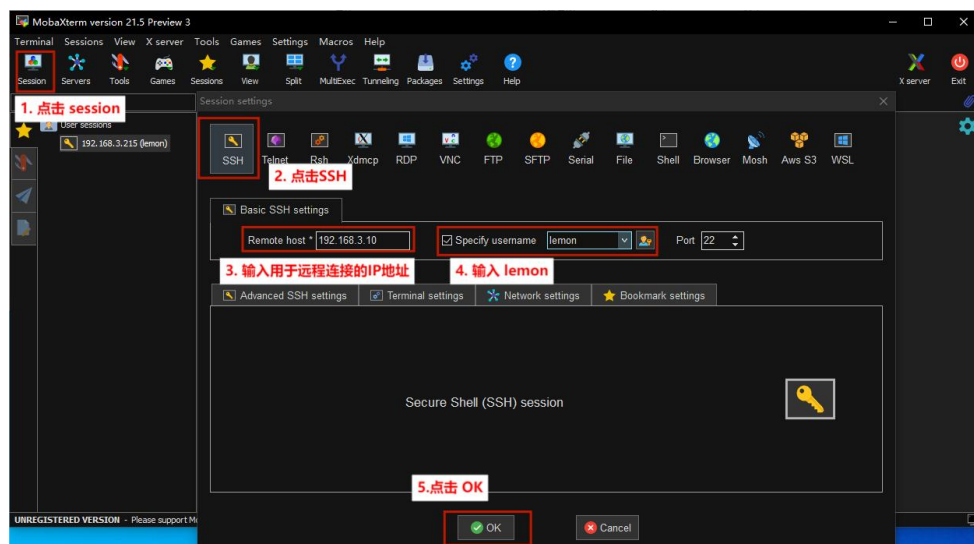
- 使用以下指令连接该 WiFi：

`sudo nmcli dev wifi connect` 无线网络名字 `password` 密码

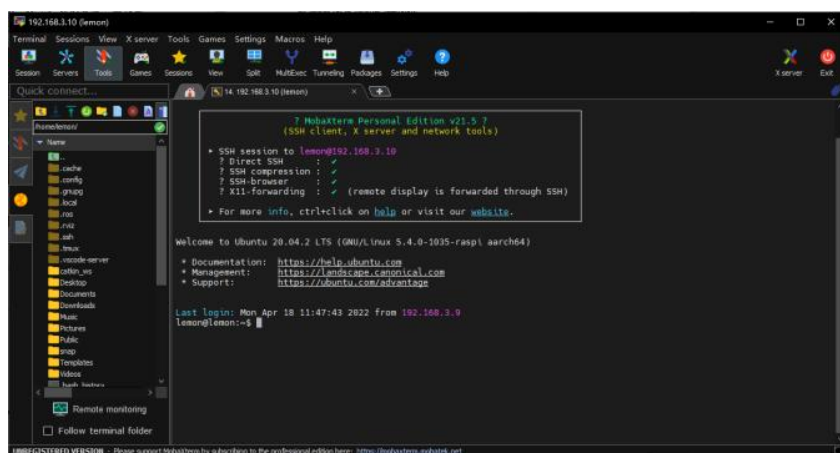
- 连接成功后，可以使用指令：`ifconfig`，查看远程连接的 IP 地址。请记住此地址，远程登陆时，需要使用此 IP 地址。



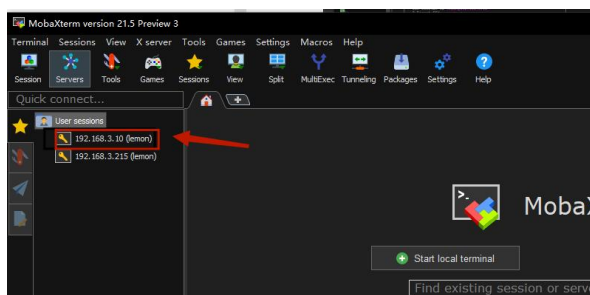
- 重点：记住这个远程连接 IP 地址后，就可以拔掉机器人连接的网线。之后可以使用这个 IP 地址来远程连接机器人。
- 按照下图步骤，点击菜单栏“session”，重新建立一个用于远程连接的 session。



- 连接成功后，输入密码：**leju123**。密码输入成功后，弹出是否储存密码的窗口，推荐选择“No”，之后每次登录的时候需输入密码：**leju123**。若想记住密码，可选择“Yes”，再按流程操作。
- 输入密码，成功登录。至此，为机器人作网络配置的任务就完成了。



- 之后想远程连接机器人时，只需要在机器人开机的状态下，打开“MobaXterm”软件，双击用于远程连接的 session 就可以了。



（备注：若忘记 IP 地址，可以重新登录路由器获取）

2.3 注意事项

- 电量过低可能影响机器人的正常使用，例如执行动作不正常或无法连接机器人，请尽量在电量充足的情况下进行调试。
- 在关机时请先通过以下指令关闭树莓派，然后再切断机器人电源：

sudo shutdown -P now

- 使用关闭机器人电源的方式关闭树莓派系统可能会导致文件受损，出现 read-only file system 的问题。当出现此问题时可以通过执行下列指令解决。

```
? MobaXterm 20.3 ?
(SSH client, X-server and networking tools)

> SSH session to lemon@192.168.3.221
? SSH compression : ✓
? SSH-browser      : ✓
? X11-forwarding   : ✓ (remote display is forwarded through SSH)
? DISPLAY          : ✓ (automatically set on remote server)
> For more info, ctrl+click on help or visit our website

Last login: Mon Jun  7 19:24:48 2021 from 192.168.3.199
/usr/bin/xaauth: error in locking authority file /home/lemon/.Xauthority
mktemp: failed to create file via template '/tmp/setup.sh.XXXXXXXXXX': Read-only file system
Could not create temporary file:
mktemp: failed to create file via template '/tmp/setup.sh.XXXXXXXXXX': Read-only file system
Could not create temporary file:
lemon@lemon:~$
```

```
sudo fsck /dev/mmcblk0p1
sudo fsck /dev/mmcblk0p2
sudo reboot
```

3. 执行 python 代码进行运动控制

3.1 编辑动作

3.1.1 编辑动作方式总结

调节机器人舵机值→点击“增加动作”→生成一个动作帧→ 重复步骤，生成多个动作帧→点击“生成模块”→生成动作

3.1.2 调节机器人舵机的方法

调节机器人的舵机值共有两种方法：第一种是手工扭转法，通过点击舵机 ID 或接触框，对舵机进行加解锁。解除舵机后，直接用手扭动机器人关节处的舵机，使得机器人形态发生变化，舵机加锁后，会读取舵机当前角度值，最终形成既定的动作。第二种是舵值调整法，通过直接输入舵机值来改变机器人动作形态。

3.1.3 例子：举手

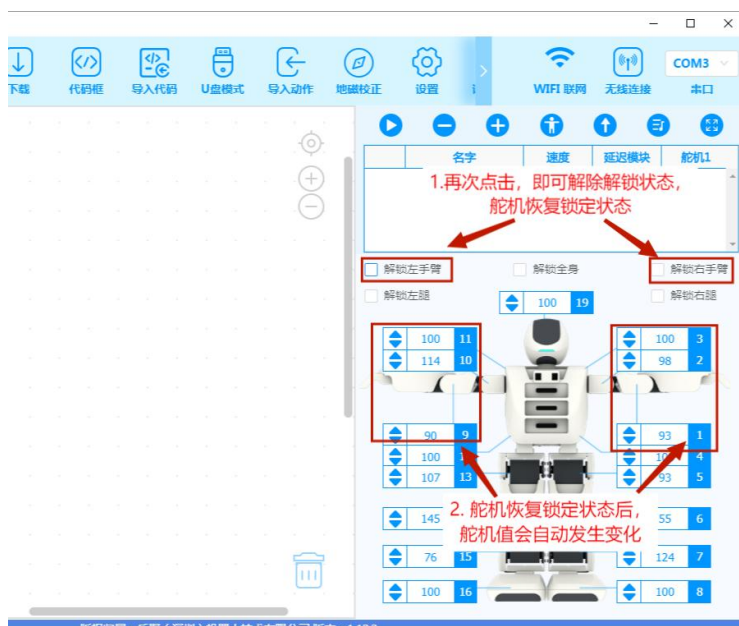
- 以下将通过一个编“举手”动作的例子来介绍如何编辑动作：
- 点击“恢复站立”按钮，让机器人恢复站立状态。之后点击解锁手臂的选择框，让机器人手臂解锁。



- 手臂解锁后，机器人手臂可自由扳动，本例子中，让机器人平举。



- 扳动好机器人手臂后，取消解锁两个手臂。此时，机器人视图区的舵机已显示为当前手部平举动作的舵机值。



- 点击“增加动作”，刚刚平举手的动作将记录在动作视图列表中。至此，就完成一个动作帧。之后可以重复：点击“解锁舵机”→扳动机器人→取消“解锁舵机”→点击“增加动作”，这样的过程来完成一个动作帧的动作。



- 也可以通过直接修改舵机框内的数值，来调节舵机值。



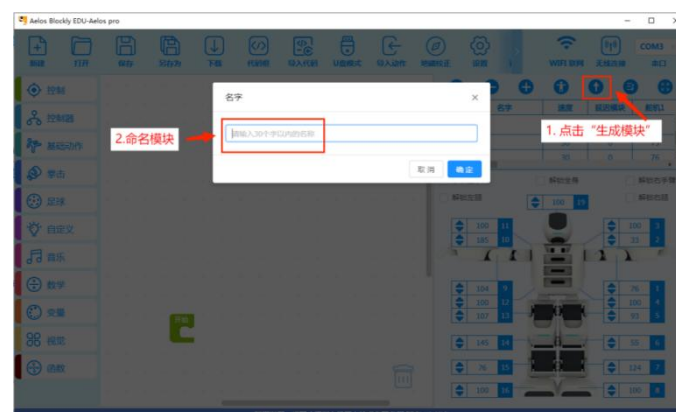
- 列表中可以为动作设置下列属性：
 - 速度：机器人执行该动作帧时的运动速度，越大则运动越快。
 - 延迟：动作帧间的停顿时间。
 - 刚度：改变舵机的柔韧性。例如当执行动作时发现舵机有点软无法很好的支撑，则可以适当加大刚度。



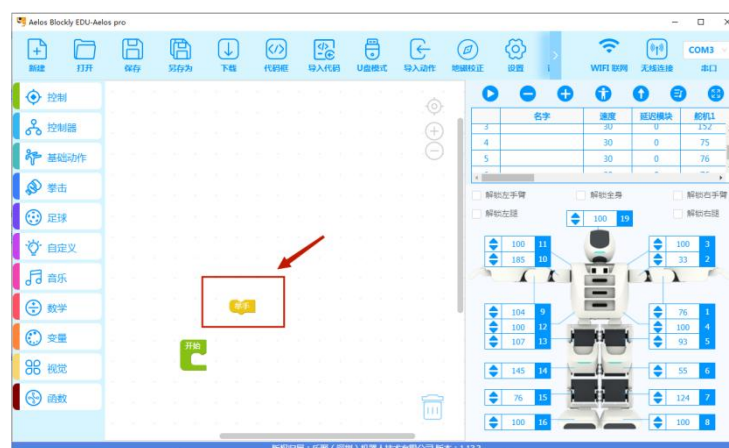
- 完成多个动作帧后，可以点击“动作预览”，预览我们刚刚编辑的动作。也可以点击单独的某一列，预览某个动作帧。



- 动作编辑完成后，点击“生成模块”，命名模块，就可以保存刚刚编辑的动作。



- 生成的动作模块将展示至编程面板内



- 将模块放置在“开始”积木块内，成为完整的程序。点击“下载”，将动作下载至机器人。重启机器人，机器人开机后将自动执行“下蹲举手”的动作。



3.2 下载动作

- 在执行动作前，需点击菜单栏的“动作下载”。此步骤可以将系统内置的动作、音乐及自定义的动作下载至机器人。



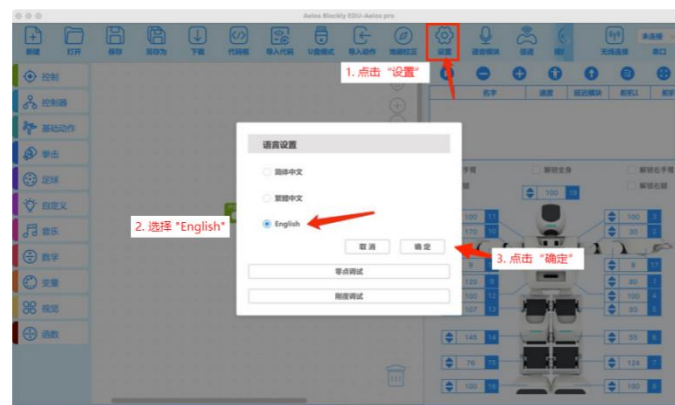
- 请注意，下载完毕后，需再重启机器人，才可以按照下面的步骤执行动作。

3.3 执行动作

- 打开“MobaXterm”软件，远程连接 ssh。
- 进入 `"/home/lemon/catkin_ws/src/robot_demo/scripts/CMDcontrol.py"` 此路径，再输入对应的动作名，即可执行。需要注意的是，在 Aelos 机器人中，动作可被分为三类：第一类是机器人系统内置的动作，第二类是用户自定义编辑的动作，第三类是音乐。对于每一类动作，在执行时会有稍许不一样的地方，需要仔细区分，下面将会对每一类进行介绍。

3.3.1 机器人系统内置动作

- 对于系统内置的动作，需输入动作的英文名。因此需要先在设置中切换语言为英语，然后在对应分类下找到动作的英文名。例如下蹲对应 Squat。





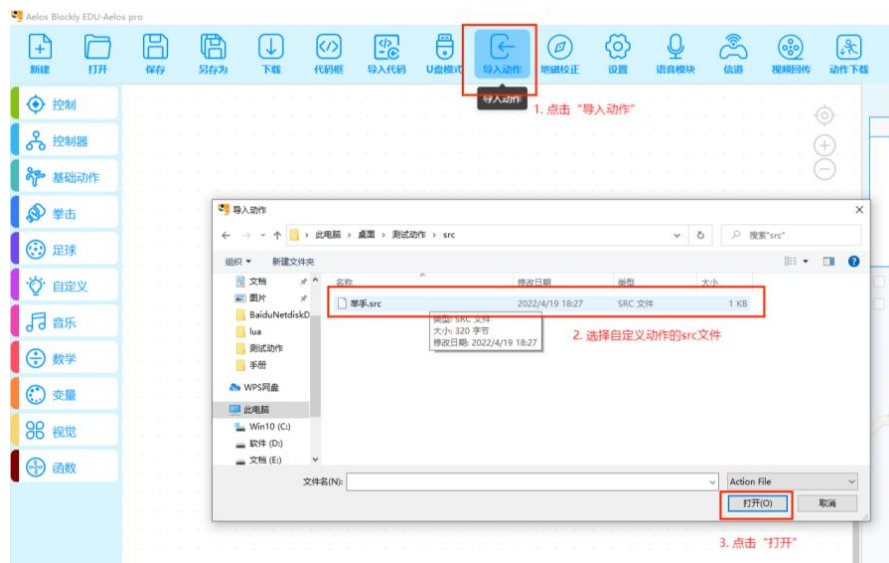
- 得到系统内置动作的英文名后，再进入路径，输入动作的英文名字，机器人开始执行动作。

```

lemon@lemon:~$ "/home/lemon/catkin_ws/src/robot_demo/scripts/CMDcontrol.py"
this is MDS action
please act_name123:提示...
Squat
haha: Squat
act_name
lej_u_f0b381df91e3f7af9b88235fb80e2f99
action_wait done
please_act_name123:
  
```

3.3.2 自定义编辑的动作

- 对于自定义编辑动作，需先将动作导入。具体的操作步骤如下图所示：



- 动作导入后，再点击动作下载。下载完毕后，重启机器人。

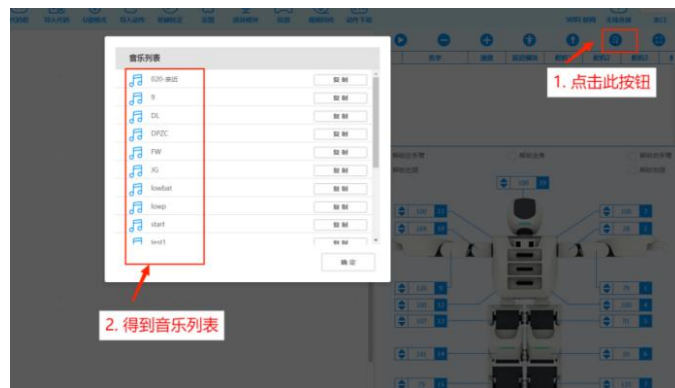


- 按照下图步骤，进入路径，输入自定义的动作名字，机器人开始执行动作。

```
lemon@lemon:~$ "/home/lemon/catkin_ws/src/robot_demo/scripts/CMDcontrol.py"
this is MD5_action
please act_name123:提示...
举手
haha: 举手
act_name
leju_c5ab749f1113956d563578e41f3c8936
action_wait done
please act_name123:
```

3.3.3 机器人音乐

- 对于系统内置的音乐，可以在软件中通过音乐列表按键获取当前可用的音乐。

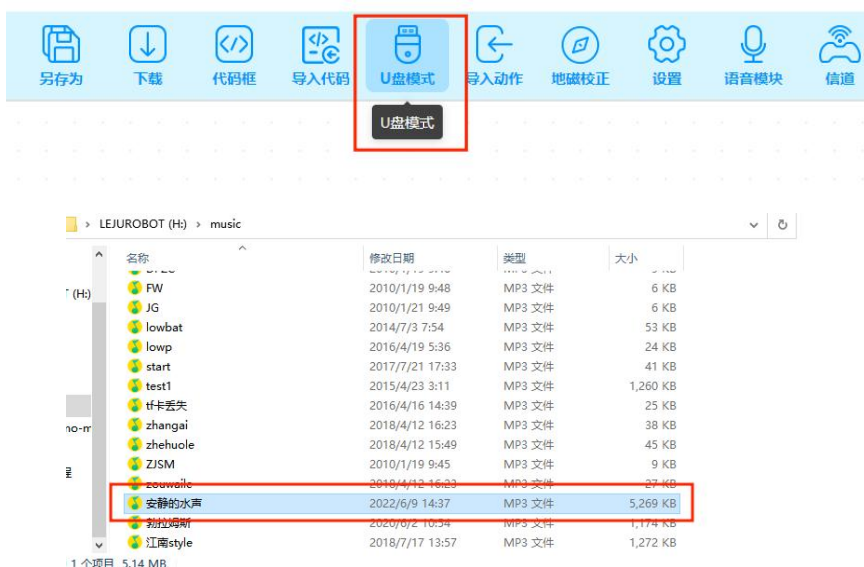


- 按照下图步骤，进入路径，输入动作名为‘play_’前缀加上音乐名，即可执行。

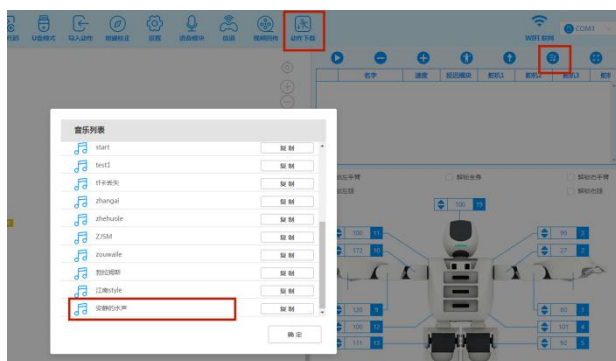
```
lemon@lemon:~$ "/home/lemon/catkin_ws/src/robot_demo/scripts/CMDcontrol.py"
this is MD5_action
please act_name123:提示...
play_tf卡丢失
haha: play_tf卡丢失
act_name
leju_02a613b34a84742ba9159e9d2c6bff6e
action_wait done
please act_name123:
```

- 若想要机器人播放其他音乐，可以用 U 盘模式添加音乐。具体步骤为：连接串口后，点击菜单栏的 U 盘模式。之后进入机器人 U 盘，将准备好的音乐文件拖入至 u 盘中。例如，下图中“安静的水声”为新导入的音乐。

（注意：音乐文件需为 MP3 格式；音乐文件的名字中不能有特殊符号，只能为中英文文字和数字。）



- 导入音乐后，请重启机器人。连接串口，在音乐列表中，能找到我们之前添加的音乐。再点击“动作下载”，将音乐下载至机器人。



- 下载完毕后。重启机器人，再进入路径，输入动作名为‘play_’前缀加上音乐名，即可执行。

```

Lemon@lemon:~$ "/home/Lemon/catkin_ws/src/robot_demo/scripts/CMDcontrol.py"
this is MD5_action
please act_name123:提示...
play_安静的水声
haha: play_安静的水声
act_name
leju_1907bb9a3bb63b833ebc6e108da8c14a
action_wait done
please act_name123:
  
```


4. 机器人 ROS 启动

4.1 ROS 简介

ROS (Robot Operating System, 下文简称“ROS”) 是一个适用于机器人的开源的元操作系统。它提供了操作系统应有的服务, 包括硬件抽象, 底层设备控制, 常用函数的实现, 进程间消息传递, 以及包管理。它也提供用于获取、编译、编写、和跨计算机运行代码所需的工具和库函数。

- 关于 ROS 相关资料可以访问官方网址获取: <http://wiki.ros.org/>
- 机器人中的 demo 程序位于 /home/lemon/catkin_ws/src/robot_demo

```
lemon@lemon:~/catkin_ws/src/robot_demo$ ls
Cm4_Test_manual  CMakeLists.txt  config  launch  package.xml  README.md  scripts
lemon@lemon:~/catkin_ws/src/robot_demo$
```

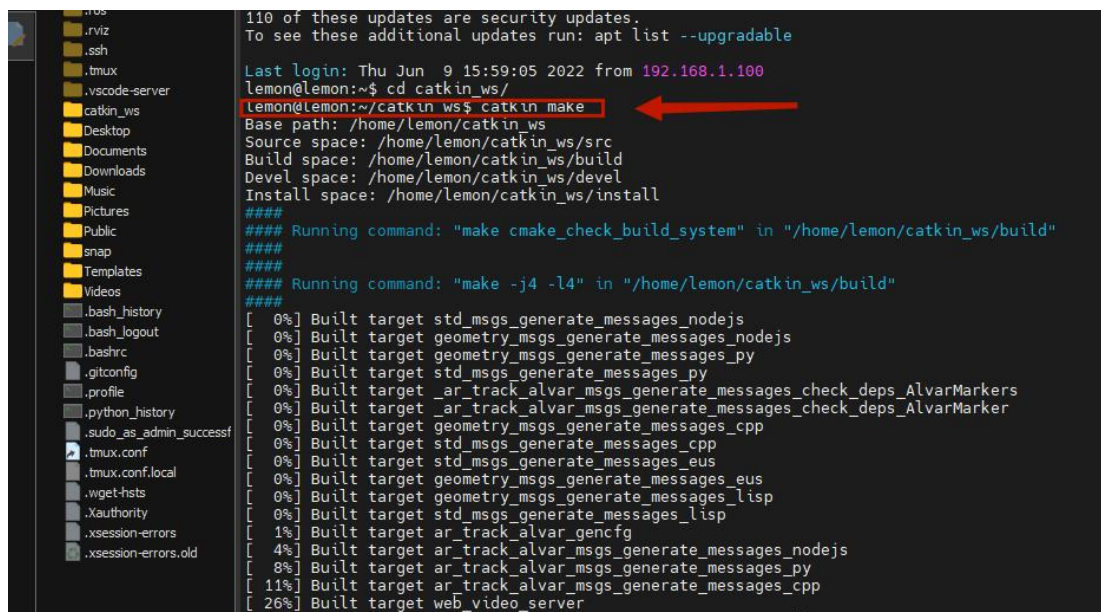
ros launch 文件 demo代码目录

4.2 运行 ROS 节点的操作步骤

步骤一: ssh 登录后, 进入 catkin_ws 文件夹

- 如果是第一次执行程序或者有过对节点和库的修改, 则需要执行以下编译指令:

catkin_make



```
110 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Last login: Thu Jun  9 15:59:05 2022 from 192.168.1.100
lemon@lemon:~$ cd catkin_ws/
lemon@lemon:~/catkin_ws$ catkin_make
Base path: /home/lemon/catkin_ws
Source space: /home/lemon/catkin_ws/src
Build space: /home/lemon/catkin_ws/build
Devel space: /home/lemon/catkin_ws/devel
Install space: /home/lemon/catkin_ws/install
####
#### Running command: "make cmake_check_build_system" in "/home/lemon/catkin_ws/build"
####
#### Running command: "make -j4 -l4" in "/home/lemon/catkin_ws/build"
####
[ 0%] Built target std_msgs_generate_messages_nodejs
[ 0%] Built target geometry_msgs_generate_messages_nodejs
[ 0%] Built target geometry_msgs_generate_messages_py
[ 0%] Built target std_msgs_generate_messages_py
[ 0%] Built target _ar_track_alvar_msgs_generate_messages_check_deps_AlvarMarkers
[ 0%] Built target _ar_track_alvar_msgs_generate_messages_check_deps_AlvarMarker
[ 0%] Built target geometry_msgs_generate_messages_cpp
[ 0%] Built target std_msgs_generate_messages_cpp
[ 0%] Built target std_msgs_generate_messages_eus
[ 0%] Built target geometry_msgs_generate_messages_eus
[ 0%] Built target geometry_msgs_generate_messages_lisp
[ 0%] Built target std_msgs_generate_messages_lisp
[ 1%] Built target ar_track_alvar_genmsg
[ 4%] Built target ar_track_alvar_msgs_generate_messages_nodejs
[ 8%] Built target ar_track_alvar_msgs_generate_messages_py
[ 11%] Built target ar_track_alvar_msgs_generate_messages_cpp
[ 26%] Built target web_video_server
```

- 如果之前编译过或编译成功后，则运行以下指令：

source devel/setup.bash

```
Last login: Thu Jun  9 16:15:50 2022 from 192.168.1.100
lemon@lemon:~$ cd catkin_ws/
lemon@lemon:~/catkin_ws$ source devel/setup.bash
lemon@lemon:~/catkin_ws$ roslaunch robot_demo ar_track.launch
... logging to /home/lemon/.ros/log/41c3210e-e7cd-11ec-a5c8-f712ac605a6f/roslaunch-lemon-22933.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://lemon:44931/

SUMMARY
=====

PARAMETERS
* /ar_track_alvar/marker_size: 4.4
* /ar_track_alvar/max_new_marker_error: 0.05
```

步骤二：执行 **roslaunch robot_demo ar_track.launch**，启动 ROS 节点。

```
lemon@lemon:~/catkin_ws$ roslaunch robot_demo ar_track.launch
... logging to /home/lemon/.ros/log/881eed1e-e/cc-11ec-a5c8-f712ac605a6f/roslaunch-lemon-20390.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

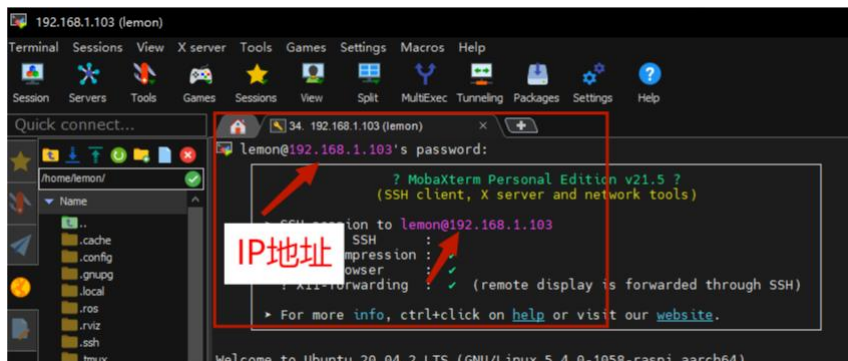
started roslaunch server http://lemon:42037/

SUMMARY
=====

PARAMETERS
* /ar_track_alvar/marker_size: 4.4
* /ar_track_alvar/max_new_marker_error: 0.05
* /ar_track_alvar/max_track_error: 0.05
* /ar_track_alvar/output_frame: /map
* /roscdistro: noetic
* /rosversion: 1.15.9
* /usb_cam_chest/camera_frame_id: usb_cam
* /usb_cam_chest/image_height: 480
* /usb_cam_chest/image_width: 640
* /usb_cam_chest/io_method: mmap
* /usb_cam_chest/pixel_format: yuyv
* /usb_cam_chest/video_device: /dev/video2
* /usb_cam_head/camera_frame_id: usb_cam
* /usb_cam_head/image_height: 480
* /usb_cam_head/image_width: 640
* /usb_cam_head/io_method: mmap
* /usb_cam_head/pixel_format: yuyv
```

步骤三：测试摄像头节点是否正常工作。

- 将电脑和机器人接在相同局域网中，打开浏览器。在网址处输入“IP 地址：8080”，连接机器人 8080 端口。



- 此时，点击对应链接即可显示摄像头画面。

