



# 快速开始

文档地址: <https://docs.ducorobots.cn/zh/develop/latest/index.html>

接口文件: DucoCobotAPI\_py/DucoCobot.py

推荐直接查看接口文件，更加直接和清晰

本文档介绍DucoCobot.py如何根据DucoCobotAPI\_py/DucoCobot.py提供的接口，进一步封装

## Demo流程

```
# 创建实例
robot = SiasunRobotPythonInterface()

# 获取当前位姿
xyz, deg = robot.get_xyz_eulerdeg()
print(xyz, deg)

# 移动到目标位姿z轴增加20mm
x, y, z = xyz
robot.moveL([x, y, z+20], deg)
```

## 创建连接

在创建SiasunRobotPythonInterface类时  
实际上触发了DucoCobot类的

- 实例创建DucoCobot
- 连接操作open
- 机器人上电power\_on
- 机器人上使能enable
- 停止任务stop

```
def init_robot(self):
    self.duco = DucoCobot(self.robot_ip, 7003)
    while self.duco.open() == -1:
        self.duco.close()
        time.sleep(1)
    self.duco.power_on(True)
    self.duco.enable(True)
    self.duco.stop(True)
    self.tcp_pose = None
```

## 获取当前位姿

实际触发了DucoCobot类的get\_tcp\_pose()方法，获取位姿信息（单位m和rad）  
转化为mm和度单位进行返回

```
# 获取当前位姿
def get_xyz_eulerdeg(self):
    """获取欧拉角格式的位姿信息
    Returns:
        list: 位置信息(x,y,z)
        list: 姿态信息(rx,ry,rz) 角度360度制
    """
    pose = self.duco.get_tcp_pose()
    trans = np.array(pose[:3]) * 1000
    deg = [c/math.pi* 180 for c in pose[3:]]

    return trans.tolist(), deg
```

## 移动到目标位姿（z轴增加20mm）

实际触发了DucoCobot类的move()方法，传入目标位姿信息，执行运动

```

def moveL(self, trans, deg, vel=0.4, acc=0.2):
    """直线运动
    Args:
        trans (list: [x,y,z]): 位姿信息 mm
        deg (list: [rx, ry, rz]): 角度信息 degree (360度)
    Returns:
        list: 返回码列表([0] 成功 [errcode]错误码)
    """
    trans_mm = np.array(trans)/1000.0 # convert from millimeter to meter
    deg_radian = np.radians(deg)
    target_P = trans_mm.tolist() + deg_radian.tolist() # concat the two lists
    block = True
    print("moveL target_P (m, radians): ", target_P)
    # moveL(self, p, v, a, r, q_near, tool, wobj, block, op=op_)
    ret = self.duco.moveL(target_P, vel, acc, 0, None, None, None, block)
    return ret

```

## 完整运行demo

首先确保机器人有安全的操作空间，以及紧急制动按钮在手边（遇到bug可停止），然后在当前目录（DucoCobotAPI）下运行

```
python SiasunRobot.py
```

可以看到机械臂沿着z轴抬高了20mm