



WLKATA  
开塔机器人

# Python SDK

## 用户手册

(技术开发文档)

文档版本: V1.009

发布日期: 2024/04/16

**版权所有© 勤牛创智科技有限公司 2023。保留一切权利。**

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 免责声明

在法律允许的最大范围内，本手册所描述的产品（含其硬件、软件、固件等）均“按照现状”提供，可能存在瑕疵、错误或故障，勤牛创智不提供任何形式的明示或默示保证，亦不对使用本手册或使用本公司产品导致的任何特殊、偶然或间接的损害进行赔偿。在使用本产品前详细阅读本使用手册及网上发布的相关技术文档并了解相关信息，确保在充分了解产品相关知识的前提下使用本产品。

本产品的使用者有责任确保遵循相关国家的切实可行的法律法规，确保在勤牛创智机械臂的使用中不存在任何重大危险。

## 版本修订说明

时间	版本号	修订记录
2023.6.21	V1.000 版	初稿
2023.12.05	V1.007 版	增加 E4 和传送带控制
2024.04.16	V1.009 版	修改舵机夹爪控制方式，增加 Mirobot 机械臂调试 GUI 插件，该插件可使用串口或 RS485 调试机械臂笛卡尔坐标，并保存成 TXT 或 Gcode 文件。

## 北京勤牛创智科技有限公司

地址：北京市海淀区清华东路16号3号楼中关村能源与安全科技园1603室

网址：cn.wlkata.com

# 目录

1. Python SDK 简介 .....	- 5 -
1.1. Python SDK 概述 .....	- 5 -
1.2. PIP 安装 .....	- 6 -
● wlkatapython 模块安装 .....	- 6 -
● pyserial 模块安装 .....	- 7 -
2. Python SDK 控制函数介绍 .....	- 8 -
2.1. 调用库 .....	- 8 -
2.2. 创建机械臂对象(Mirobot) .....	- 8 -
2.3. 设置机械臂地址 .....	- 8 -
2.4. 多功能控制器重启 .....	- 9 -
2.5. 回零 .....	- 9 -
2.6. 机械臂回到初始位置 .....	- 10 -
2.7. 执行离线文件 .....	- 10 -
2.8. 机械臂停止 .....	- 11 -
2.9. 气泵控制 .....	- 11 -
2.10. 夹爪控制 .....	- 12 -
2.11. PWM 控制 .....	- 13 -
2.12. 透传指令 .....	- 13 -
2.13. 固件版本查询 .....	- 13 -
2.14. 获取机械臂全部状态 .....	- 14 -
2.15. 获取机械臂运动状态 .....	- 15 -

2.16. 获取机械臂各轴角度 .....	- 15 -
2.17. 获取机械臂坐标 .....	- 16 -
2.18. 获取机械臂运动模式 .....	- 17 -
2.19. 笛卡尔坐标控制机械臂(Mirobot) .....	- 17 -
2.20. 角度控制机械臂(Mirobot) .....	- 19 -
2.21. 机械臂速度 .....	- 21 -
2.22. 机械臂拓展轴控制 .....	- 21 -
2.23. 创建机械臂对象(E4) .....	- 22 -
2.24. 笛卡尔坐标控制机械臂(E4) .....	- 22 -
2.25. 角度控制机械臂(E4) .....	- 24 -
2.26. 创建传送带对象 .....	- 25 -
2.27. 设置传送带地址 .....	- 25 -
2.28. 传送带透传指令 .....	- 25 -
2.29. 传送带速度 .....	- 26 -
2.30. 创建 Mirobot_Serial_GUI 对象 .....	- 26 -
2.31. 调取 Mirobot_GUI .....	- 26 -

# 必 读

如实现本手册所示的各种功能，需将多功能控制器固件升级至 **20230710 及以上版本**，Mirobot 机械臂固件升级至 **20231129 及以上版本**，E4 机械臂升级至 **20231031 及以上版本**，可下载 **WkataStudio-Windows-V3.X.X** 配合使用，并对机械臂及多功能控制器进行升级。

# 1. Python SDK 简介

## 1.1. Python SDK 概述

本模块主要功能为 Mirobot 机械臂以及 E4 机械臂控制，**需要注意的是控制端需要连接到多功能控制器进行通信，不能控制端直接连接到机械臂本体。**控制端与多功能控制器之间可采用 RS485、UART、进行通信。

本模块是基于 pyserial 库进行通信的，故使用本模块的同时需要加载 pyserial 库。

使用本模块请使用 Python3.9 以上版本，且多功能控制器升级到 20230710 以上版本。

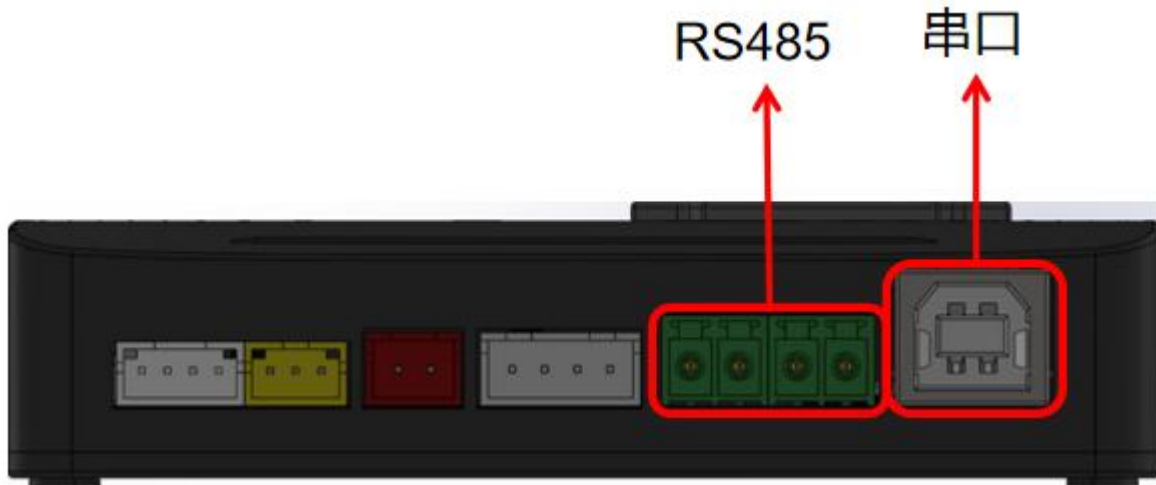
本手册所描述内容对应的模块版本为：

wlkatapython	0.0.9
pyserial	3.5

函数所给出的示例程序都是针对控制端与多功能控制器 RS485 通信进行的。在本文末尾有针对控制端与多功能控制器串口通信的示例程序。

Wlkata\_UART 类主要控制的是 Mirobot 机械臂及其拓展轴。

E4 类主要控制的是 E4 机械臂及其拓展轴。E4 类是继承自 Wlkata\_UART 类，故 Wlkata\_UART 中的控制函数可用于控制 E4 机械臂，其中笛卡尔坐标控制、角度控制，笛卡尔坐标获取、角度获取等函数用法有所区别，详细用法见下文。

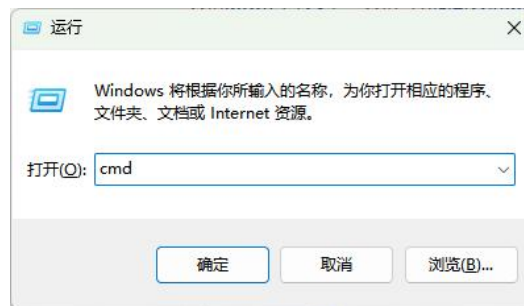


多功能控制器 RS485 设置方法请参照：[多功能控制器用户手册](#)

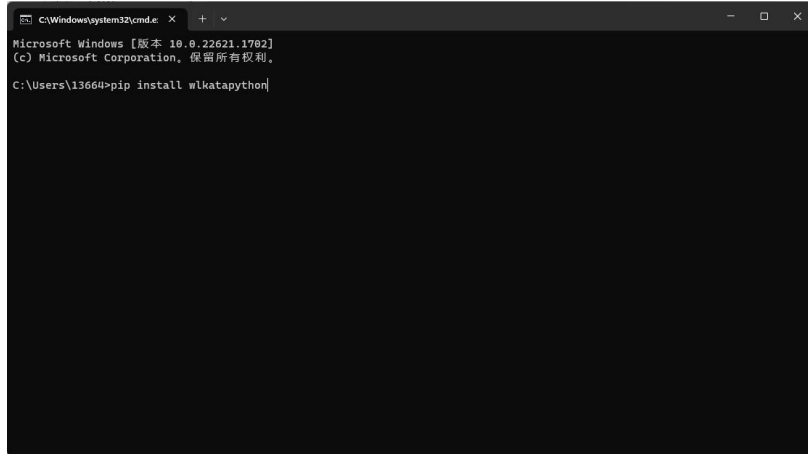
## 1.2. PIP 安装

- wlkatapython **模块安装**

在 Windows 下使用 win+r 键打开运行窗口，输入 cmd 后点击确认。



在弹出的命令提示符窗口中输入：`pip install wlkatapython`



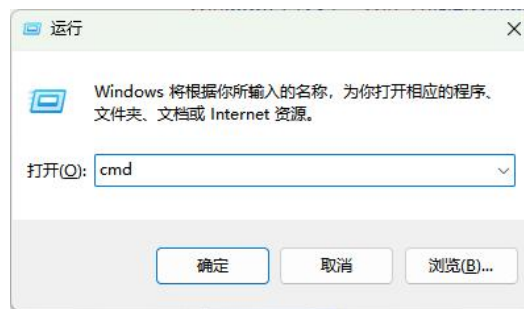
```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.22621.1702]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\13664>pip install wlkatapython
```

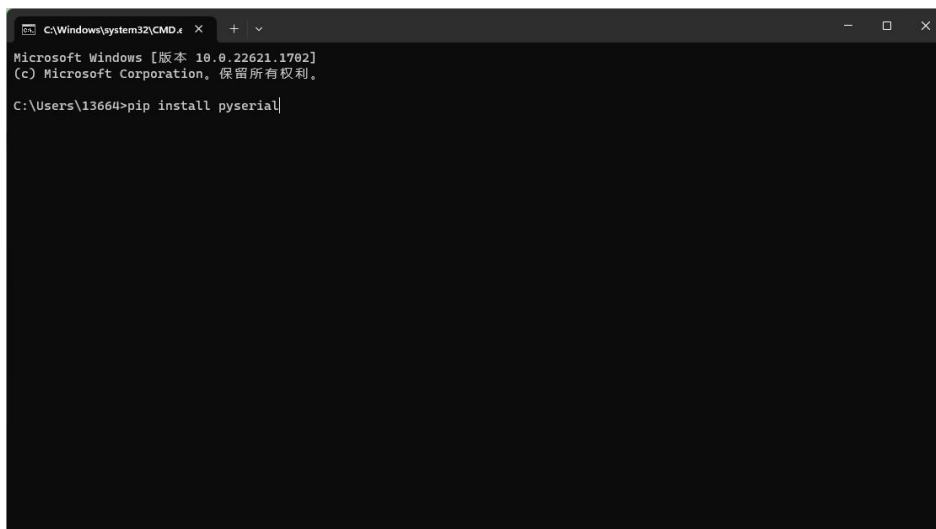
点击回车即可自动安装（需电脑联网）。

- **pyserial 模块安装**

在 Windows 下使用 win+r 键打开运行窗口，输入 cmd 后点击确认



在弹出的命令提示符窗口中输入： pip install pyserial



```
C:\Windows\system32\CMD.exe
Microsoft Windows [版本 10.0.22621.1702]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\13664>pip install pyserial
```

点击回车即可自动安装（需电脑联网）。



## 2. Python SDK 控制函数介绍

### 2.1. 调用库

<b>函数</b>	<code>import wlkatapython</code>
<b>函数说明</b>	导入 WLKATA 库文件。
<b>参数说明</b>	无
<b>返回值</b>	无
<b>备注</b>	无

### 2.2. 创建机械臂对象(Mirobot)

<b>函数</b>	<code>Wlkata_UART();</code>
<b>函数说明</b>	创建机械臂对象。
<b>参数说明</b>	无

### 2.3. 设置机械臂地址

<b>函数</b>	<code>init( p, adr);</code>
<b>函数说明</b>	将串口和多功能控制器地址进行绑定。
<b>参数说明</b>	P:串口号; adr:多功能控制器地址。

## 2.4. 多功能控制器重启

<b>函数</b>	restart();
<b>函数说明</b>	控制多功能控制器重启。
<b>参数说明</b>	无
<b>示例</b>	<pre>import wlkatapython import serial  ''' 多功能控制器重启'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 mirobot1.restart() #多功能控制器重启 serial1.close() #关闭串口</pre>

## 2.5. 回零

<b>函数</b>	homing();
<b>函数说明</b>	机械臂回零操作。
<b>参数说明</b>	无
<b>示例</b>	<pre>import wlkatapython import serial  ''' 机械臂回零'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 mirobot1.homing() #机械臂回零 serial1.close() #关闭串口</pre>

## 2.6. 机械臂回到初始位置

<b>函数</b>	zero();
<b>函数说明</b>	控制机械臂回到初始位置。
<b>参数说明</b>	无
<b>示例</b>	<pre>import wlkatapython import serial  ''' 机械臂回到初始位置'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 mirobot1.zero() #机械臂回到初始位置 serial1.close() #关闭串口</pre>

## 2.7. 执行离线文件

<b>函数</b>	runFile(fileName, num=False);
<b>函数说明</b>	执行多功能控制器中的离线文件;
<b>参数说明</b>	<p>fileName,: 离线文件名称;</p> <p>num: 是否循环执行, False 为只执行 1 次, True 为循环执行该离线文件。该参数为选填, 不填写默认为 False。</p>
<b>示例</b>	<pre>import wlkatapython import serial  ''' 执行离线文件'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象</pre>

	<pre>mirobot1.init(serial1, 1) #设置机械臂地址 mirobot1.runFile("ceshi", False) #循环执行某一文件, True 为循环执行文件, False 为单次执行文件 serial1.close() #关闭串口</pre>
--	---

## 2.8. 机械臂停止

<b>函数</b>	cancellation();
<b>函数说明</b>	机械臂以减速形式停止当前动作，并清理掉缓存动作。
<b>参数说明</b>	无
<b>示例</b>	<pre>import wlkatapython import serial  ''' 停止当前程序'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 mirobot1.runFile("ceshi", True) #循环执行某一文件, True 为循环执行文件, False 为单次执行文件 time.sleep(5) #等待 5 秒钟 mirobot1.cancellation() #停止当前运行的程序 serial1.close() #关闭串口</pre>

## 2.9. 气泵控制

<b>函数</b>	pump(num);
<b>函数说明</b>	控制气泵吸气、吹气、关闭。
<b>参数说明</b>	num:0—关闭;

	1—吸气; 2—吹气。
<b>示例</b>	<pre> import wlkatapython import serial import time     '''控制气泵''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 for i in range(0, 3): #气泵吸气吹气     mirobot1.pump(1) #气泵吸气     time.sleep(3)     mirobot1.pump(2) #气泵吹气     time.sleep(3) mirobot1.pump(0) #气泵关闭 serial1.close() #关闭串口                 </pre>

## 2.10. 夹爪控制

<b>函数</b>	gripper(num);
<b>函数说明</b>	控制夹爪张开以及闭合。
<b>参数说明</b>	num:夹爪状态——0 夹爪断电, 1 夹爪打开, 2 夹爪闭合。
<b>示例</b>	<pre> import wlkatapython import serial import time     '''夹爪控制''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率                 </pre>

```

mirobot1 =wllkatapython.Wlkata_UART() #新建 mirobot1 对象
mirobot1.init(serial1, 1) #设置机械臂地址
for i in range(0,3): #电动夹爪
    mirobot1.gripper(0) #夹爪断电
    time.sleep(1)
    mirobot1.gripper(1) #夹爪打开
    time.sleep(1)
    mirobot1.gripper(2) #夹爪闭合
    time.sleep(1)
serial1.close() #关闭串口
    
```

### 2.11. PWM 控制

<b>函数</b>	pwmWrite(num);
<b>函数说明</b>	向 PWM 口输出信号。
<b>参数说明</b>	num:PWM 输出值。

### 2.12. 透传指令

<b>函数</b>	sendMsg(string);
<b>函数说明</b>	向机械臂传输指令。
<b>参数说明</b>	String: 需要传输的指令。

### 2.13. 固件版本查询

<b>函数</b>	version();
-----------	------------

<b>函数说明</b>	查询机械臂及多功能控制器固件版本。(当前只支持 UART 查询, RS485 暂不支持)
<b>参数说明</b>	无
<b>返回值</b>	机械臂及多功能控制器版本号。
<b>示例</b>	<pre>import wlkatapython import serial  ''' 固件版本查询'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 print(mirobot1.version()) #查询机械臂及多功能控制器信息 serial1.close() #关闭串口</pre>

## 2.14. 获取机械臂全部状态

<b>函数</b>	getStatus();
<b>函数说明</b>	查询机械臂状态。
<b>参数说明</b>	无
<b>返回值</b>	以字典的形式返回机械臂运动状态、角度模式下各轴角度、坐标模式下的坐标及末端工具状态 (气泵+夹爪)、及当前运动模式。
<b>示例</b>	<pre>import wlkatapython import serial  ''' 获取机械臂全部状态'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址</pre>

	<pre>print(mirobot1.getStatus())#获取机械臂全部信息 serial1.close()#关闭串口</pre>
--	---

## 2.15. 获取机械臂运动状态

<b>函数</b>	getState();
<b>函数说明</b>	查询机械臂状态。
<b>参数说明</b>	无
<b>返回值</b>	返回机械臂运动状态。
<b>示例</b>	<pre>import wlkatapython import serial ''' 获取机械臂状态''' serial1 = serial.Serial("COM3", 38400)#设置串口及波特率 mirobot1 =wlkatapython.Wlkata_UART()#新建 mirobot1 对象 mirobot1.init(serial1, 1)#设置机械臂地址 print(mirobot1.getState())#获取机械臂状态 serial1.close()#关闭串口</pre>

## 2.16. 获取机械臂各轴角度

<b>函数</b>	getAngle(num);
<b>函数说明</b>	获取机械臂各轴角度。
<b>参数说明</b>	Num:1/2/3/4/5/6/7, 表示 1-6 轴的角度, 7 表示传送带的角度。
<b>返回值</b>	返回机械臂各轴角度及传送带的角度。
<b>示例</b>	<pre>import wlkatapython import serial</pre>



	<pre> ''' 获取机械臂角度''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 =wllkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 for i in range(1,8):     print(mirobot1.getAngle(i)) #获取机械臂角度, 1 到 6     对应 1 到 6 轴角度, 7 对应传送带 serial1.close() #关闭串口                 </pre>
--	--

## 2.17. 获取机械臂坐标

<b>函数</b>	getcoordinate(num);
<b>函数说明</b>	获取机械臂坐标。
<b>参数说明</b>	Num:1/2/3/4/5/6, 分别表示机械臂 X//Y/Z/RX/RY/RZ 坐标。
<b>返回值</b>	返回机械臂 X//Y/Z/RX/RY/RZ 坐标。
<b>示例</b>	<pre> import wllkatapython import serial ''' 获取机械臂坐标''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 =wllkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 for i in range(1,7):     print(mirobot1.getcoordinate(i)) #获取机械臂坐标, 1     到 6 对应 x/y/z/rx/ry/rz 坐标 serial1.close() #关闭串口                 </pre>

## 2.18. 获取机械臂运动模式

<b>函数</b>	getmooe();
<b>函数说明</b>	获取机械臂运动模式。
<b>参数说明</b>	无
<b>返回值</b>	0 为角度模式, 1 为坐标模式。
<b>示例</b>	<pre>import wlkatapython import serial  ''' 运动模式查询'''  serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 print(mirobot1.getmooe()) #运动模式查询 serial1.close() #关闭串口</pre>

## 2.19. 笛卡尔坐标控制机械臂(Mirobot)

<b>函数</b>	writecoordinate(motion,position,x,y,z,a,b,c);
<b>函数说明</b>	笛卡尔坐标控制机械臂。
<b>参数说明</b>	<p>Motion:0—快速运动;</p> <p>1—直线运动;</p> <p>2—门型运动。</p> <p>Position: 0—绝对值运动;</p> <p>1—增量值运动。</p> <p>x: x 点坐标;</p>

	y: y 点坐标;  z: z 点坐标;  a: a 点坐标;  b: b 点坐标;  c: c 点坐标;
<b>返回值</b>	无
<b>示例</b>	<pre> import wlkatapython import serial import time  ''' 机械臂使用三指柔爪抓取木块'''  coordinate_g=[[198.6 , 0 , 230 , 0 , 0 , 0],               [198.6 , 0 , 90.7 , 0 , 0 , 0],               [198.6 , 0 , 70.7 , 0 , 0 , 0],               [198.6 , 0 , 90.7 , 0 , 0 , 0],               [152.1 , -127.6 , 90.4 , 0 , 0 , -39.9],               [152.1 , -127.6 , 70.4 , 0 , 0 , -39.9],               [152.1 , -127.6 , 90.4 , 0 , 0 , -39.9],               ]  serial1 = serial.Serial("COM4", 38400) #设置串口及波特率 mirobot1 = wlkatapython.Wlkata_UART() #新建 mirobot1 对象 mirobot1.init(serial1, 1) #设置机械臂地址 if mirobot1.getState()=="Alarm": #如果机械臂处于 Alarm 状态则进行回 零     mirobot1.homing()                 </pre>

```

for i in range(0,7):
    while mirobot1.getState()!="Idle":
        print("机械臂 1 状态: ",mirobot1.getState())

mirobot1.writecoordinate(0,0,coordinate_g[i][0],coordinate_g[i][1],
coordinate_g[i][2],coordinate_g[i][3],coordinate_g[i][4],coordinate_g[i][5])#快速/直线/门型,绝对/相对,x,y,z,a,b,c,
    if i==1:
        mirobot1.pump(1) # 气泵吸气
        time.sleep(3)
    elif i==2:
        mirobot1.pump(2) # 气泵吹气
        time.sleep(3)
    elif i==5:
        mirobot1.pump(1) # 气泵吸气
        time.sleep(3)
mirobot1.pump(0) # 气泵关闭
mirobot1.zero()#机械臂回到初始位置
serial1.close()#关闭串口
    
```

## 2.20. 角度控制机械臂(Mirobot)

<b>函数</b>	writeangle(position,x,y,z,a,b,c);
<b>函数说明</b>	笛卡尔坐标控制机械臂。
<b>参数说明</b>	Position: 0—绝对值运动; 1—增量值运动。  x: 1 轴角度;

	y: 2 轴角度; z: 3 轴角度; a: 4 轴角度; b: 5 轴角度; c: 6 轴角度;
<b>返回值</b>	无
<b>示例</b>	<pre> import wlkatapython import serial  coordinate_g=[[30 ,0 ,0 ,0 ,0 ,0],               [-30 ,0 ,0 ,0 ,0 ,0],                [0 ,30 ,0 ,0 ,0 ,0],               [0 ,-30 ,0 ,0 ,0 ,0],                [0 ,0 ,30 ,0 ,0 ,0],               [0 ,0 ,-30 ,0 ,0 ,0],                [0 ,0 ,0 ,30 ,0 ,0],               [0 ,0 ,0 ,-30 ,0 ,0],                [0 ,0 ,0 ,0 ,30 ,0],               [0 ,0 ,0 ,0 ,-30 ,0],                [0 ,0 ,0 ,0 ,0 ,30],               [0 ,0 ,0 ,0 ,0 ,-30],             </pre>

```

]
serial1 = serial.Serial("COM4", 115200) #设置串口及波特率
mirobot1 =wltkapython.Wlkata_UART() #新建 mirobot1 对象
mirobot1.init(serial1, -1) #设置机械臂地址
if mirobot1.getState()=="Alarm": #如果机械臂处于 Alarm 状态则进行回
零
    mirobot1.homing()
for i in range(0,12):
    while mirobot1.getState()!="Idle":
        print("机械臂 1 状态: ",mirobot1.getState())
mirobot1.writeangle(0,coordinate_g[i][0],coordinate_g[i][1],coord
inate_g[i][2],coordinate_g[i][3],coordinate_g[i][4],coordinate_g[
i][5]) #快速/直线/门型, 绝对/相对, x, y, z, a, b, c,
mirobot1.zero() #机械臂回到初始位置
serial1.close() #关闭串口
    
```

## 2.21. 机械臂速度

函数	speed(num);
函数说明	机械臂速度设置。
参数说明	num: 机械臂速度, 取值范围 0~2000。
返回值	无

## 2.22. 机械臂拓展轴控制

函数	writeexpand(motion, position, d);
函数说明	控制机械臂拓展轴运动。

<b>参数说明</b>	Motion:0—快速运动; 1—直线运动。 Position: 0—绝对值运动; 1—增量值运动。 d:移动多少距离。
<b>返回值</b>	无

### 2.23. 创建机械臂对象(E4)

<b>函数</b>	E4();
<b>函数说明</b>	创建机械臂对象。
<b>参数说明</b>	无

### 2.24. 笛卡尔坐标控制机械臂(E4)

<b>函数</b>	writecoordinate(motion,position,x,y,z,a);
<b>函数说明</b>	笛卡尔坐标控制机械臂。
<b>参数说明</b>	Motion:0—快速运动; 1—直线运动; 2—门型运动。 Position: 0—绝对值运动; 1—增量值运动。

	x: x 点坐标;  y: y 点坐标;  z: z 点坐标;  a: a 点坐标;
<b>返回值</b>	无
<b>示例</b>	<pre> import wlkatapython import serial import time  '''E4 机械臂笛卡尔坐标运动''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 E4_1 = wlkatapython.E4() #新建 E4_1 对象 E4_1.init(serial1, 1) #设置机械臂地址 E4_1.restart() time.sleep(3) E4_1.homing() while E4_1.getState()=="Home":     pass E4_1.writecoordinate(0, 1, 20, 20, 20) while E4_1.getState()=="Idle":     pass E4_1.writecoordinate(0, 1, -20, -20, -20) while E4_1.getState()=="Idle":     pass serial1.close() #关闭串口             </pre>



## 2.25. 角度控制机械臂(E4)

<b>函数</b>	writeangle(position,x,y,z,a);
<b>函数说明</b>	笛卡尔坐标控制机械臂。
<b>参数说明</b>	Position: 0—绝对值运动; 1—增量值运动。  x: 1 轴角度; y: 2 轴角度; z: 3 轴角度; a: 4 轴角度;
<b>返回值</b>	无
<b>示例</b>	<pre> import wlkatapython import serial import time  '''E4 机械臂角度坐标运动''' serial1 = serial.Serial("COM3", 38400) #设置串口及波特率 E4_1 = wlkatapython.E4() #新建 E4_1 对象 E4_1.init(serial1, 1) #设置机械臂地址 E4_1.restart() time.sleep(3) E4_1.homing() while E4_1.getState()!="Idle":     pass E4_1.writeangle(1, 20, 20, 20) while E4_1.getState()!="Idle":                     </pre>

	<pre>                 pass             E4_1.writeangle(1, -20, -20, -20)             while E4_1.getState() != "Idle":                 pass             serial1.close() #关闭串口         </pre>
--	---

## 2.26. 创建传送带对象

<b>函数</b>	MS4220_UART();
<b>函数说明</b>	创建传送带对象。
<b>参数说明</b>	无

## 2.27. 设置传送带地址

<b>函数</b>	init( p, adr);
<b>函数说明</b>	将串口和多功能控制器地址进行绑定。
<b>参数说明</b>	P:串口号; adr:多功能控制器地址。

## 2.28. 传送带透传指令

<b>函数</b>	sendMsg(string);
<b>函数说明</b>	向机械臂传输指令。

<b>参数说明</b>	String: 需要传输的指令。
-------------	------------------

## 2.29. 传送带速度

<b>函数</b>	speed(num);
<b>函数说明</b>	传送带速度设置。
<b>参数说明</b>	num: 传送带速度, 取值范围-100~100。负值为反方向转动, 正值为正方向转动, 数值应为整数。
<b>返回值</b>	无

## 2.30. 创建 Mirobot\_Serial\_GUI 对象

<b>函数</b>	Mirobot_Serial_GUI();
<b>函数说明</b>	创建 Mirobot_Serial_GUI 对象
<b>参数说明</b>	无
<b>返回值</b>	无

## 2.31. 调取 Mirobot\_GUI

<b>函数</b>	Mirobot_Serial_GUI();
<b>函数说明</b>	创建 Mirobot_Serial_GUI 对象
<b>参数说明</b>	无
<b>返回值</b>	无

<b>示例</b>	<pre>import wlkatapython  Mirobot_GUI=wlkatapython.Mirobot_Serial_GUI()  Mirobot_GUI.Mirobot_GUI()</pre>
-----------	--

如果需要使用多功能控制器串口进行控制则将波特率改为 115200，地址改为-1 即可，以下是使用多

功能控制器串口通信执行离线文件示例：

```
import wlkatapython
import serial
''' 执行离线文件'''
serial1 = serial.Serial("COM3", 115200) #设置串口及波特率
mirobot1 =wlkatapython.Wlkata_UART() #新建 mirobot1 对象
mirobot1.init(serial1, -1) #设置机械臂地址
mirobot1.runFile("ceshi", False) #循环执行某一文件，True 为循环执行文件，False 为单
次执行文件
serial1.close() #关闭串口
```



官方公众号