[份](https://fairino-doc-zhs.readthedocs.io/latest/index.html) / [SDK](https://fairino-doc-zhs.readthedocs.io/latest/SDKManual/index.html) / [Python](https://fairino-doc-zhs.readthedocs.io/latest/SDKManual/python_intro.html) / 2. 机器⼈运动



**2.** 机器⼈运动

**2.1.** 机器⼈点动

**2.1.1.** **jog**点动 [、](#bookmark1)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |  |  |
| StartJOG(ref,nb,dir,max\_dis,vel=20.0,acc=100.0) | | | | |
|  | | | | | |
| 描述 | jog点动 | | | | | | |
| 必选参数 |  | |  | | ：0-关节点动,2-基坐标系点动,4-⼯具坐标系点动,8-⼯件坐标系点动； | | |
| ref | |
| nb | ：1-1关节(x轴),2-2关节(y轴),3-3关节(z轴),4-4关节(rx),5-5关节(ry),6-6关节(rz) | | | |
| dir ：0-负⽅向，1-正⽅向;  max\_dis ：单次点动最⼤角度/距离，单位 ° 或 mm; | | | | |
| 默认参数 |  | |  | | ：速度百分⽐ ，[0~100] 默认20;  ：加速度百分⽐ ，[0~100] 默认100; | | |
| vel | |
|  |  |
| acc | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | | | |

**2.1.2.** **jog**点动减速停⽌

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |
| StopJOG(ref) | | |
|  |  |  |
| 描述 | jog点动减速停⽌ | | | | |
| 选参数 | . ref ：1-关节点动停⽌ ,3-基坐标系点动停⽌ ,5-⼯具坐标系点动停⽌ ,9-⼯件坐标系点动停 | | | | |
| 认参数 | ⽆ | | | | |
| 回值 | 错误码 成功-0 失败- errcode | | | | |

**2.1.3.** **jog**点动⽴即停⽌

 latest 

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| ImmStopJOG() |
|  |
| 描述 | jog点动⽴即停⽌ | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.1.3.1.** 代码示例



|  |  |  |
| --- | --- | --- |
| 1  **from fairino import** Robot  2  **import time**  3  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  4  robot = Robot.RPC( I192.168.58.2I)  5  *# 机器⼈单轴点动*  6  robot.StartJOG(0,1,0,20.0,20.0,30.0) *# 单关节运动,StartJOG为⾮阻塞指令，* *运动状态下接* *收其他运动指令（* *包含StartJOG）* *会被丢弃* | | |
| 7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62 | time.sleep(1)  *#机器⼈单轴点动减速停⽌*  ret = robot.StopJOG(1)  print(ret)  *#机器⼈单轴点动⽴即停⽌*  robot.ImmStopJOG()  robot.StartJOG(0,2,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(0,3,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(0,4,1,20.0,vel=40)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(0,5,1,20.0,acc=50)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(0,6,1,20.0,20.0,30.0)  time.sleep(1)  robot.ImmStopJOG()  *# 基坐标*  robot.StartJOG(2,1,0,20.0) *#基坐标系下点动* time.sleep(1)  *# #机器⼈单轴点动⽴即停⽌*  robot.ImmStopJOG()  robot.StartJOG(2,1,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(2,2,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(2,3,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(2,4,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(2,5,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(2,6,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  *# ⼯具坐标*  robot.StartJOG(4,1,0,20.0,20.0,100.0) *#⼯具坐标系下点动* time.sleep(1)  *# #机器⼈单轴点动⽴即停⽌*  robot.ImmStopJOG()  robot.StartJOG(4,1,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(4,2,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(4,3,1,20.0) | latest |

|  |  |
| --- | --- |
| 63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96 | time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(4,4,1,20.0,20.0,100.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(4,5,1,20.0,vel=10.0,acc=20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(4,6,1,20.0,acc=40.0)  time.sleep(1)  robot.ImmStopJOG()  *# ⼯件坐标*  robot.StartJOG(8,1,0,20.0,20.0,100.0) *#⼯件坐标系下点动* time.sleep(1)  *# #机器⼈单轴点动⽴即停⽌*  robot.ImmStopJOG()  robot.StartJOG(8,1,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(8,2,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(8,3,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(8,4,1,20.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(8,5,1,20.0,vel=30.0)  time.sleep(1)  robot.ImmStopJOG()  robot.StartJOG(8,6,1,20.0,20.0,acc=90.0)  time.sleep(1)  robot.ImmStopJOG() |

**2.2.** 关节空间运动

|  |  |
| --- | --- |
| MoveJ(joint\_pos, tool, user, desc\_pos = [0.0,0.0,0.0,0.0,0.0,0.0], vel = 20.0, acc = 0.0, ovl = 100 | |
| = [0.0,0.0,0.0,0.0], blendT = -1.0, offset\_flag = 0, offset\_pos = [0.0,0.0,0.0,0.0,0.0,0.0]) |  |
|  |
| 关节空间运动 | |

. joint\_pos :⽬标关节位置，单位[°]；

. tool :⼯具号，[0~14]；

. user :⼯件号，[0~14]；

 desc\_pos :⽬标笛卡尔位姿，单位 [mm][°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤正运动

vel :速度百分⽐ ，[0~100] 默认20.0;

 acc :加速度百分⽐ ，[0~100] ，暂不开放；

ovl :速度缩放因⼦ ，[0~100] 默认100.0;

 exaxis\_pos :外部轴 1 位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0];

blendT :[-1.0]-运动到位 (阻塞) ，[0~500.0]-平滑时间 (⾮阻塞) ，单位 [ms] 默认-1.0;

 offset\_flag :[0]-不偏移，[1]-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默认 0

 offset\_pos :位姿偏移量，单位 [mm][°] 默认[0.0,0.0,0.0,0.0,0.0,0.0];

 latest 

错误码 成功-0 失败- errcode

**2.2.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **from fairino import** Robot  **import time**  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  joint\_pos4 = [-83.24, -96.476, 93.688, -114.079, -62, -100]  joint\_pos5 = [-43.24, -70.476, 93.688, -114.079, -62, -80]  joint\_pos6 = [-83.24, -96.416, 43.188, -74.079, -80, -10]  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  ret = robot.MoveJ(joint\_pos4, tool, user, vel=30) *#关节空间运动* print("关节空间运动点4:错误码", ret)  ret = robot.MoveJ(joint\_pos5, tool, user) print("关节空间运动点5:错误码", ret)  robot.MoveJ(joint\_pos6, tool, user, offset\_flag=1, offset\_pos= [10,10,10,0,0,0]) print("关节空间运动点6:错误码", ret) |

**2.3.** 笛卡尔空间直线运动

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | MoveL(desc\_pos, tool, user, joint\_pos = [0.0,0.0,0.0,0.0,0.0,0.0], vel = 20.0, acc = 0.0 , ovl = |   blendR = -1.0, exaxis\_pos = [0.0,0.0,0.0,0.0], search = 0, offset\_flag = 0, offset\_pos = [0.0,0.0,0.0,0.0,0.0,0.0],overSpeedStrategy=0,speedPercent=10) | | | | | | | | | | |
| 笛卡尔空间直线运动 | | | | | | | | | | |
| . desc\_pos :⽬标笛卡尔位姿，单位[mm][°]；  . tool :⼯具号，[0~14]；  . user :⼯件号，[0~14]； | | | | | | | | | | |
|  | joint\_pos :⽬标关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学求        vel :速度百分⽐ ，[0~100] 默认20.0；  acc :加速度百分⽐ ，[0~100] ，暂不开放 默认0.0；  ovl :速度缩放因⼦ ，[0~100] 默认100.0；  blendR :blendR:[-1.0]-运动到位 (阻塞) ，[0~1000]-平滑半径 (⾮阻塞) ，单位 [mm] 默认-1.0; | | | | | | | | | |
| exaxis\_pos | | | | | | :外部轴 1 位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0]; | | | |
|  |  |  | :[0]-不焊丝寻位，[1]-焊丝寻位； | | | | | | |
| search | | |
| offset\_flag :ofset\_ﬂag:[0]-不偏移，[1]-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默认 0  offset\_pos :位姿偏移量，单位 [mm][°] 默认[0.0,0.0,0.0,0.0,0.0,0.0] | | | | | | | | | |
| overSpeedStrategy | | | | | | | | | :超速处理策略，0-策略关闭；1-标准；2-超速时报错停⽌； 3-⾃适应降速， |
|  |  |  |  |  |  |  |  | :允许降速阈值百分⽐[0-100] ，默认10% | |
| speedPercent | | | | | | | |
|  |  |  |  |  |  |  |  |
| 错误码 成功-0 失败- errcode | | | | | | | | | | |

**2.3.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **from fairino import** Robot  **import time**  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos1 = [36.794,-475.119, 65.379, -176.938, 2.535, -179.829]  desc\_pos2 = [136.794,-475.119, 65.379, -176.938, 2.535, -179.829]  desc\_pos3 = [236.794,-475.119, 65.379, -176.938, 2.535, -179.829]  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  ret = robot.MoveL(desc\_pos1, tool, user) *#笛卡尔空间直线运动* print("笛卡尔空间直线运动点1:错误码", ret)  robot.MoveL(desc\_pos2, tool, user, vel=20, acc=100) print("笛卡尔空间直线运动点2:错误码", ret)  robot.MoveL(desc\_pos3, tool, user, offset\_flag=1, offset\_pos= [10,10,10,0,0,0]) print("笛卡尔空间直线运动点3:错误码", ret) |

**2.4.** 笛卡尔空间圆弧运动





MoveC(desc\_pos\_p, tool\_p, user\_p, desc\_pos\_t, tool\_t, user\_t, joint\_pos\_p = [0.0,0.0,0.0,

0.0,0.0,0.0],joint\_pos\_t= [0.0,0.0,0.0,0.0,0.0,0.0], vel\_p = 20.0,acc\_p=100.0, exaxis\_pos\_p = [0.0,0. offset\_flag\_p = 0, offset\_pos\_p = [0.0,0.0,0.0,0.0,0.0,0.0], vel\_t= 20.0, acc\_t=100.0,exaxis\_pos\_t=

 [0.0,0.0,0.0,0.0], offset\_flag\_t = 0, offset\_pos\_t = [0.0,0.0,0.0, 0.0,0.0,0.0], ovl = 100.0, blend

笛卡尔空间圆弧运动

. desc\_pos\_p :路径点笛卡尔位姿，单位[mm][°]；

. tool\_p :路径点⼯具号，[0~14];

. user\_p :路径点⼯件号，[0~14];

. desc\_pos\_t :⽬标点笛卡尔位姿，单位 [mm][°];

. tool\_t :⼯具号，[0~14]；

. user\_t :⼯件号，[0~14]；

 joint\_pos\_p :路径点关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学

 joint\_pos\_t :⽬标点关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学

vel\_p :路径点速度百分⽐ ，[0~100] 默认20.0;

acc\_p :路径点加速度百分⽐ ，[0~100] 暂不开放,默认0.0;

exaxis\_pos\_p :路径点外部轴 1位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0];

offset\_flag\_p :路径点是否偏移[0]-不偏移，[1]-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默

 vel\_t :⽬标点速度百分⽐ ，[0~100] 默认20.0;

 acc\_t :⽬标点加速度百分⽐ ，[0~100] 暂不开放 默认0.0;

 exaxis\_pos\_t :⽬标点外部轴 1 位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0];

 offset\_flag\_t :⽬标点是否偏移[0]-不偏移，[1]-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默

 offset\_pos\_t :⽬标点位姿偏移量，单位 [mm][°] 默认[0.0,0.0,0.0,0.0,0.0,0.0];

 ovl: :速度缩放因⼦ ，[0~100] 默认100.0;

blendR :[-1.0]-运动到位 (阻塞) ，[0~1000]-平滑半径 (⾮阻塞) ，单位 [mm] 默认-1.0;

|  |  |  |
| --- | --- | --- |
| 错误码 成功-0 失败- errcode |  | latest |

**2.4.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  *动*  12 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos1 = [236.794,-475.119, 65.379, -176.938, 2.535, -179.829]  desc\_posc1 = [266.794,-455.119, 65.379, -176.938, 2.535, -179.829] *#MoveC过渡点*  desc\_posc2 = [286.794,-475.119, 65.379, -176.938, 2.535, -179.829] *#MoveC⽬标点* tool = 0*#⼯具坐标系编号*  user = 0 *#⼯件坐标系编号*  ret = robot.MoveL(desc\_pos1, tool, user, vel=30, acc=100) print("笛卡尔空间直线运动 :错误码", ret)  ret = robot.MoveC(desc\_posc1, tool, user, desc\_posc2,tool, user) *#笛卡尔空间圆弧运* print("笛卡尔空间圆弧运动 :错误码", ret) |

**2.5.** 笛卡尔空间整圆运动

|  |
| --- |
| Circle(desc\_pos\_p,tool\_p,user\_p,desc\_pos\_t,tool\_t,user\_t,joint\_pos\_p= [0.0,0.0,0.0,0.0,0.0,0.0], jo |

[0.0,0.0,0.0,0.0,0.0,0.0], vel\_p = 20.0, acc\_p=0.0, exaxis\_pos\_p= [0.0,0.0, 0.0,0.0], vel\_t=20.0, a

|  |  |
| --- | --- |
| exaxis\_pos\_t = [0.0,0.0,0.0,0.0], ovl=100.0, offset\_flag=0, offset\_pos= [0.0,0.0,0.0,0.0,0.0,0.0]) |  |

笛卡尔空间整圆运动

. desc\_pos\_p :路径点笛卡尔位姿，单位[mm][°]；

. tool\_p :⼯具号，[0~14]；

. user\_p :⼯件号，[0~14]；

. desc\_pos\_t :⽬标点笛卡尔位姿，单位[mm][°]；

. tool\_t :⼯具号，[0~14]；

. user\_t :⼯件号，[0~14]；

. joint\_pos\_p :路径点关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学

. joint\_pos\_t :⽬标点关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学

. vel\_p :速度百分⽐ ，[0~100] 默认20.0;

. acc\_p :路径点加速度百分⽐ ，[0~100] 暂不开放 默认0.0;

. exaxis\_pos\_p :路径点外部轴 1 位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0];

. vel\_t :⽬标点速度百分⽐ ，[0~100] 默认20.0;

. acc\_t :⽬标点加速度百分⽐ ，[0~100] 暂不开放 默认0.0;

. exaxis\_pos\_t :标点外部轴 1 位置 ~ 外部轴 4 位置 默认[0.0,0.0,0.0,0.0]

. ovl :速度缩放因⼦ ，[0~100] 默认100.0;

. offset\_flag :是否偏移[0]-不偏移，[1]-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默认 0;

. offset\_pos :位姿偏移量，单位 [mm][°] 默认[0.0,0.0,0.0,0.0,0.0,0.0]

错误码 成功-0 失败- errcode

**2.5.1.** 代码示例

|  |
| --- |
| **from fairino import** Robot  1  2  3  4  5  6  7  8  9  10  11  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos2 = [236.794,-475.119, 65.379, -176.938, 2.535, -179.829]  desc\_posc3 = [256.794,-435.119, 65.379, -176.938, 2.535, -179.829] *#Circle路径点*  desc\_posc4 = [286.794,-475.119, 65.379, -176.938, 2.535, -179.829] *#Circle⽬标点* tool = 0*#⼯具坐标系编号*  user = 0 *#⼯件坐标系编号*  robot.MoveL(desc\_pos2, tool, user, vel=40, acc=100) print("笛卡尔空间直线运动 :错误码", ret)  ret = robot.Circle(desc\_posc3, tool, user, desc\_posc4, tool, user, vel\_t=40,  offset\_flag=1, offset\_pos= [5,10,15,0,0,1]) *#笛卡尔空间圆弧运动* 12  print("笛卡尔空间圆弧运动 :错误码", ret) *#笛卡尔空间整圆运动* |

**2.6.** 笛卡尔空间螺旋线运动



param, joint\_pos = [0.0,0.0,0.0,0.0,0.0,0.0], vel = 20.0, acc = 0.0, exaxis\_pos = [0.0,0.0,0.0,0.0],





单 位

位[mm][°];



gle, rad\_init, rad\_add, rotaxis\_add,

螺旋圈数;circle\_angle: 螺旋倾⻆;rad\_init: 螺旋初始半径;rad\_add: 半径增量;rotaxis\_add: 转轴⽅向增量

位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学求解返回值; 认20.0;

默认100.0;

外部轴 4 位置 默认[0.0,0.0,0.0,0.0]; 默认100.0;

-⼯件/基坐标系下偏移，[2]-⼯具坐标系下偏移 默认 0; [mm][°] 默认[0.0,0.0,0.0,0.0,0.0,0.0]





**2.6.1.** 代码示例

|  |
| --- |
| 1  **from fairino import** Robot  2  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  3  robot = Robot.RPC( I192.168.58.2I)  4  desc\_pos\_spiral= [236.794,-475.119, -65.379, -176.938, 2.535, -179.829]*#Spiral ⽬* *标点*  5  *#螺旋线参数[circle\_num,circle\_angle,rad\_init,rad\_add,rotaxis\_add,rot\_direction]* 6  *# circle\_num:螺旋圈数，circle\_angle:螺旋倾角，rad\_init:螺旋初始半径，rad\_add:半径增量，* 7  *# rotaxis\_add:转轴⽅向增量，rot\_direction:旋转⽅向，0-顺时针，1-逆时针*  8  param = [5.0,10,30,10,5,0]  9  tool = 0*#⼯具坐标系编号*  10  user = 0 *#⼯件坐标系编号*  11  ret = robot.NewSpiral(desc\_pos\_spiral, tool, user, param,vel=40 ) *#笛卡尔空间螺旋* *线运动*  12  print("笛卡尔空间螺旋线运动 :错误码", ret) |

**2.7.** 伺服运动开始

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| ServoMoveStart() |
|  |
| 描述 | 伺服运动开始，配合ServoJ 、ServoCart指令使⽤ | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.8.** 伺服运动结束

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| ServoMoveEnd() |
|  |
| 描述 | 伺服运动结束，配合ServoJ 、ServoCart指令使⽤ | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.9.** 关节空间伺服模式运动

 latest

|  |  |
| --- | --- |
| 原型 | ServoJ(joint\_pos, axisPos, acc = 0.0, vel = 0.0, cmdT = 0.008, filterT = 0.0, gain = 0.0) |
| 描述 | 关节空间伺服模式运动 |
| 必选参数 | . joint\_pos :⽬标关节位置，单位[°]；  . axisPos :外部轴位置,单位mm； |

|  |  |
| --- | --- |
| 默认参数 | . acc :加速度，范围 [0~100] ，暂不开放，默认为 0.0;  . vel :速度，范围 [0~100] ，暂不开放，默认为 0.0;  . cmdT :指令下发周期，单位s ，建议范围[0.001~0.0016], 默认为0.008;  . filterT :滤波时间，单位 [s] ，暂不开放， 默认为0.0;  . gain :⽬标位置的⽐例放⼤器，暂不开放， 默认为0.0; |
| 返回值 | 错误码 成功-0 失败- errcode |

**2.10.** 笛卡尔空间伺服模式运动

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| ServoCart(mode, desc\_pos, pos\_gain = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0] , acc = 0.0, vel = 0.0, c | | | | | | |
| 0.008, filterT = 0.0, gain = 0.0) | | | | | |  |
|  |  |  |  |  |  |
|  | 笛卡尔空间伺服模式运动 | | | | | | | |
| 数 | . mode :[0]-绝对运动(基坐标系) ，[1]-增量运动(基坐标系) ，[2]-增量运动(⼯具坐标系)；  . desc\_pos :⽬标笛卡尔位置/⽬标笛卡尔位置增量； | | | | | | | |
| 数 | . pos\_gain :位姿增量⽐例系数，仅在增量运动下⽣效，范围 [0~1], 默认为 [1.0, 1.0, 1.0, 1.0, 1.  . acc :加速度，范围 [0~100] ，暂不开放，默认为 0.0;  . vel :速度，范围 [0~100] ，暂不开放，默认为 0.0;  . cmdT :指令下发周期，单位s ，建议范围[0.001~0.0016], 默认为0.008;  . filterT :滤波时间，单位 [s] ，暂不开放， 默认为0.0;  . gain :⽬标位置的⽐例放⼤器，暂不开放， 默认为0.0; | | | | | | | |
| 值 | 错误码 成功-0 失败- errcode | | | | | | | |



**2.10.1.** 代码示例

|  |  |
| --- | --- |
| 1  **from fairino import** Robot  2  **import time**  3  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象* 4  robot = Robot.RPC( I192.168.58.2I)  5  error,joint\_pos = robot.GetActualJointPosDegree()  6  print("机器⼈当前关节位置",joint\_pos)  7  joint\_pos =  [joint\_pos [0],joint\_pos [1],joint\_pos [2],joint\_pos [3],joint\_pos [4],joint\_pos [5]]  8  error\_joint = 0  9  count =100  10  error = robot.ServoMoveStart() *#伺服运动开始* 11  print("伺服运动开始错误码",error)  12  **while**(count):  13  error = robot.ServoJ(joint\_pos=joint\_pos,axisPos= [0,0,0,0,0,0]) *#关节空间伺服*  *模式运动* | |
| 14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | **if** error!=0:  error\_joint =error  joint\_pos [0] = joint\_pos [0] + 0.1 *#每次1轴运动0.1度，* *运动100次*  count = count - 1  time.sleep(0.008)  print("关节空间伺服模式运动错误码",error\_joint) error = robot.ServoMoveEnd() *#伺服运动结束* print("伺服运动结束错误码",error)  mode = 2 *#[0]-绝对运动(基坐标系)，* *[1]-增量运动(基坐标系)，* *[2]-增量运动(⼯具坐标系)* n\_pos = [0.0,0.0,0.5,0.0,0.0,0.0] *#笛卡尔空间位姿增量*  error,desc\_pos = robot.GetActualTCPPose()  print("机器⼈当前笛卡尔位置",desc\_pos)  count = 100  error\_cart =0  error = robot.ServoMoveStart() *#伺服运动开始* print("伺服运动开始错误码",error)  **while**(count):  error = robot.ServoCart(mode, n\_pos, vel=40) *#笛卡尔空间伺服模式运动*  **if** error!=0:  error\_cart =error  count = count - 1  time.sleep(0.008)  print("笛卡尔空间伺服模式运动错误码", error\_cart) error = robot.ServoMoveEnd() *#伺服运动开始*  print("伺服运动结束错误码",error) |

**2.11.** 笛卡尔空间点到点运动

必选参数

|  |  |
| --- | --- |
| 原型 | MoveCart(desc\_pos, tool, user, vel = 20.0, acc = 0.0, ovl = 100.0, blendT = -1.0, config = -1) |
| 描述 | 笛卡尔空间点到点运动 |
|  | . desc\_pos :⽬标笛卡尔位置；  . tool :⼯具号，[0~14]；  . user :⼯件号，[0~14]； |

|  |  |
| --- | --- |
| 默认参数 | . vel :速度，范围 [0~100] ，默认为 20.0;  . acc :加速度，范围 [0~100] ，暂不开放,默认为 0.0;  . ovl :速度缩放因⼦ ，[0~100] ，默认为 100.0;  . blendT :[-1.0]-运动到位 (阻塞) ，[0~500]-平滑时间 (⾮阻塞) ，单位 [ms] 默认为 -1.0  . config :关节配置，[-1]-参考当前关节位置求解，[0~7]-依据关节配置求解 默认为 -1 |
| 返回值 | 错误码 成功-0 失败- errcode |

**2.11.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **from fairino import** Robot  **import time**  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos7 = [236.794,-475.119, 65.379, -176.938, 2.535, -179.829]  desc\_pos8 = [236.794,-575.119, 165.379, -176.938, 2.535, -179.829]  desc\_pos9 = [236.794,-475.119, 265.379, -176.938, 2.535, -179.829]  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  robot.MoveCart(desc\_pos7, tool, user)  print("笛卡尔空间点到点运动点7:错误码", ret)  robot.MoveCart(desc\_pos8, tool, user, vel=30) print("笛卡尔空间点到点运动点8:错误码", ret)  robot.MoveCart(desc\_pos9, tool, user,) print("笛卡尔空间点到点运动点9:错误码", ret) |

**2.12.** 机器⼈样条运动

**2.12.1.** 样条运动开始

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| SplineStart() |
|  | |
| 描述 | 样条运动开始 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.12.2.** 样条运动**PTP**

|  |
| --- |
|  |
| SplinePTP(joint\_pos, tool, user, desc\_pos = [0.0,0.0,0.0,0.0,0.0,0.0], vel = 20.0, acc = 100.0, o |
|  |
| 样条运动PTP |

. joint\_pos :⽬标关节位置，单位[°]；

. tool :⼯具号，[0~14]；

. user :⼯件号，[0~14]；



. desc\_pos :⽬标笛卡尔位姿，单位 [mm][°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤正运动

. vel :速度，范围 [0~100] ，默认为 20.0;

. acc :加速度，范围 [0~100] ，默认为 100.0;

. ovl :速度缩放因⼦ ，[0~100] ，默认为 100.0



错误码 成功-0 失败- errcode

**2.12.3.** 样条运动结束

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| SplineEnd() |
|  |
| 描述 | 样条运动结束 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.12.3.1.** 代码示例

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  joint\_pos1 = [116.489,-85.278,111.501,-112.486,-85.561,24.693]  joint\_pos2 = [86.489,-65.278,101.501,-112.486,-85.561,24.693] joint\_pos3 = [116.489,-45.278,91.501,-82.486,-85.561,24.693] ret = robot.SplineStart() *#样条运动开始* | |
| print("样条运动开始 :错误码", ret)  ret = robot.SplinePTP(joint\_pos1, tool, user) print("样条运动PTP运动点1:错误码", ret)  ret = robot.SplinePTP(joint\_pos2, tool, user) print("样条运动PTP运动点2:错误码", ret)  ret = robot.SplinePTP(joint\_pos3, tool, user) print("样条运动PTP运动点3:错误码", ret)  ret = robot.SplineEnd() *#样条运动结束*  print("样条运动结束 :错误码", ret) | *#样条运动PTP*  *#样条运动PTP*  *#样条运动PTP* |

**2.13.** 机器⼈新样条运动

**2.13.1.** 新样条运动开始

*在* *python 版本发⽣变更:* SDK-v2.0.3

 latest 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |  |
| NewSplineStart(type,averageTime=2000) | | | |
|  |  |  |  |
| 描述 | 新样条运动开始 | | | | | |
| 必选参数 |  | |  | :0-圆弧过渡，1-给定点位路径点 | | |
| type |
| 默认参数 |  | |  | | :全局平均衔接时间（ ms）默认为 2000 | |
| averageTime | |
|  |  |
| 返回值 | 错误码 成功-0 失败- errcode | | | | | |

**2.13.2.** 新样条运动结束

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| NewSplineEnd() |
|  | |
| 描述 | 新样条运动结束 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.13.3.** 新样条指令点

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| NewSplinePoint(desc\_pos,tool,user,lastFlag,joint\_pos= [0.0,0.0,0.0,0.0,0.0,0.0], vel = 0.0, acc = | | | | | | | |
| = 100.0 ,blendR = 0.0 ) | | | | | | |  |
|  | | | | | | |
| 新样条指令点 | | | | | | | | |
|  | | desc\_pos :⽬标笛卡尔位姿，单位 [mm][°];  tool :⼯具号，[0~14]；  user :⼯件号，[0~14]； | | | | | | |
| lastFlag | | | | :是否为最后—个点，0-否，1-是; | | |
| . joint\_pos :⽬标关节位置，单位 [°] 默认初值为[0.0,0.0,0.0,0.0,0.0,0.0] ，默认值调⽤逆运动学求  . vel :速度，范围 [0~100] ，暂不开放，默认为 0.0;；  . acc :加速度，范围 [0~100] ，暂不开放，默认为 0.0;  . ovl :速度缩放因⼦ ，[0~100] 默认为 100.0;  . blendR : [0~1000]-平滑半径，单位 [mm] 默认0.0; | | | | | | | | |
| 错误码 成功-0 失败- errcode | | | | | | | | |

**2.13.3.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  lastFlag= 0 *# 是否为最后—个点，0-否，1-是*  desc\_pos4 = [236.794,-375.119, 65.379, -176.938, 2.535, -179.829]  desc\_pos5 = [236.794,-275.119, 165.379, -176.938, 2.535, -179.829]  desc\_pos6 = [286.794,-375.119, 265.379, -176.938, 2.535, -179.829]  ret = robot.NewSplineStart(1) *#新样条运动开始* print("新样条运动开始 :错误码", ret)  ret = robot.NewSplinePoint(desc\_pos4, tool, user, lastFlag)*#新样条指令点* print("新样条指令点4:错误码", ret)  ret = robot.NewSplinePoint(desc\_pos5, tool, user, lastFlag, vel=30)*#新样条指令点* print("新样条指令点5:错误码", ret)  lastFlag = 1  ret = robot.NewSplinePoint(desc\_pos6, tool, user, lastFlag, vel=30)*#新样条指令点* print("新样条指令点6:错误码", ret)  ret = robot.NewSplineEnd() *#新样条运动结束* print("新样条运动结束 :错误码", ret) |

**2.14.** 机器⼈终⽌运动

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| StopMotion() |
|  |
| 描述 | 终⽌运动，使⽤终⽌运动需运动指令为⾮阻塞状态 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.14.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos1 = [-187.519, 319.248, 397, -157.278, -31.188, 107.199]  desc\_pos2 = [-187.519, 310.248, 297, -157.278, -31.188, 107.199]  joint\_pos1 = [-83.24, -96.476, 93.688, -114.079, -62, -100]  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  ret = robot.MoveL(desc\_pos1, tool, user, joint\_pos=joint\_pos1) *#笛卡尔空间直线运动* print("笛卡尔空间直线运动点1:错误码", ret)  ret = robot.StopMotion() *#终⽌运动* print("终⽌运动 :错误码", ret)  robot.MoveL(desc\_pos2, tool, user, vel=40, acc=100) print("笛卡尔空间直线运动点2:错误码", ret) |

**2.15.** 机器⼈暂停运动

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

 latest 

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| PauseMotion() |
|  |
| 描述 | 暂停运动，使⽤暂停运动需运动指令为⾮阻塞状态 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.16.** 机器⼈恢复运动

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| ResumeMotion() |
|  |
| 描述 | 恢复运动，使⽤恢复运动需运动指令为⾮阻塞状态 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.17.** 机器⼈点位整体偏移

**2.17.1.** 点位整体偏移开始

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |
| PointsOffsetEnable(flag,offset\_pos) | | |
|  |  |  |
| 描述 | 点位整体偏移开始 | | | | |
| 必选参数 | . flag :0-基坐标或⼯件坐标系下偏移， 2-⼯具坐标系下偏移；  . offset\_pos :偏移量，单位[mm][°]。 | | | | |
| 默认参数 | ⽆ | | | | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | |

**2.17.2.** 点位整体偏移结束

 latest

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| PointsOffsetDisable() |
|  |
| 描述 | 点位整体偏移结束 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.17.2.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  desc\_pos3 = [-127.519, 256.248, 312, -147.278, -51.588, 107.199]  desc\_pos4 = [-140.519, 219.248, 300, -137.278, -11.188, 127.199]  desc\_pos5 = [-187.519, 319.248, 397, -157.278, -31.188, 107.199]  desc\_pos6 = [-207.519, 229.248, 347, -157.278, -31.188, 107.199]  tool = 0 *#⼯具坐标系编号* user = 0 *#⼯件坐标系编号*  flag = 1 *#0-基坐标系下/⼯件坐标系下偏移，2-⼯具坐标系下偏移*  offset\_pos = [10,20,30,0,0,0] *#位姿偏移量*  ret = robot.PointsOffsetEnable(flag,offset\_pos) print("点位整体偏移开始 :错误码", ret)  robot.MoveL(desc\_pos3, tool, user, offset\_flag=1, offset\_pos= [10,10,10,0,0,0]) print("笛卡尔空间直线运动点3:错误码", ret)  robot.MoveL(desc\_pos4, tool, user, vel=30, acc=100)  print("笛卡尔空间直线运动点4:错误码", ret) robot.MoveL(desc\_pos5, tool, user)  print("笛卡尔空间直线运动点5:错误码", ret) ret = robot.PointsOffsetDisable()  print("点位整体偏移结束 :错误码", ret) |

**2.18.** 控制箱运动**AO**开始

*在* *python 版本加⼊:* SDK-v2.0.4

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |
| MoveAOStart(AONum,maxTCPSpeed=1000,maxAOPercent=100,zeroZoneCmp=20) | | |
|  |  |  |
| 描述 | 控制箱运动AO开始 | | | | |
| 必选参数 |  | | AONum :控制箱AO编号 | | |
| 默认参数 | . maxTCPSpeed :最⼤TCP速度值[1-5000mm/s] ，默认1000；  . maxAOPercent :最⼤TCP速度值对应的AO百分⽐ ，默认100%；  . zeroZoneCmp :死区补偿值AO百分⽐ ，整形，默认为20% ，范围[0-100]。 | | | | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | |



**2.18.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( '192.168.58.2')  *#控制箱运动AO开始*  error = robot.MoveAOStart(0,100,98,1)  print("MoveAOStart",error)  error,joint\_pos = robot.GetActualJointPosDegree()  print("GetActualJointPosDegree",error,joint\_pos)  joint\_pos [0] = joint\_pos [0]+10  *#机器⼈关节运动*  error = robot.MoveJ(joint\_pos,1,1)  print("MoveJ",error)  time.sleep(3)  *#控制箱运动AO停⽌*  error = robot.MoveAOStop()  print("MoveAOStop",error) |

**2.19.** 控制箱运动**AO**结束

*在* *python 版本加⼊:* SDK-v2.0.4

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| MoveAOStop() |
|  |
| 描述 | 控制箱运动AO结束 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.20.** 末端运动**AO**开始

*在* *python 版本加⼊:* SDK-v2.0.4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |  |
| MoveToolAOStart(AONum,maxTCPSpeed=1000,maxAOPercent=100,zeroZoneCmp =20) | | | |
|  |  |  |  |
| 描述 | 末端运动AO开始 | | | | | |
| 必选参数 |  | |  | :末端AO编号 | | |
| AONum |
| 默认参数 | . maxTCPSpeed :最⼤TCP速度值[1-5000mm/s] ，默认1000；  . maxAOPercent :最⼤TCP速度值对应的AO百分⽐ ，默认100%；  . zeroZoneCmp :死区补偿值AO百分⽐ ，整形，默认为20% ，范围[0-100]。 | | | | | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | | |

**2.20.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( '192.168.58.2')  *#末端运动AO开始*  error = robot.MoveToolAOStart(0,100,98,1)  print("MoveToolAOStart",error)  error,desc\_pos = robot.GetActualTCPPose()  print("GetActualTCPPose",error,desc\_pos)  desc\_pos [2] = desc\_pos [2]-50  *#笛卡尔空间直线运动*  error = robot.MoveL(desc\_pos,1,1)  print("MoveL",error)  time.sleep(3)  *#末端运动AO停⽌*  error = robot.MoveToolAOStop()  print("MoveToolAOStop",error) |

**2.21.** 末端运动**AO**结束

*在* *python 版本加⼊:* SDK-v2.0.4

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 原型 |  |  |  | | MoveToolAOStop() | |  | | | 描述 | 末端运动AO结束 | | | | 必选参数 | ⽆ | | | | 默认参数 | ⽆ | | | | 返回值 | 错误码 成功-0 失败- errcode | | |   **2.22.** 开始**Ptp**运动**FIR**滤波 *在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 原型 |  |  |  |  |  | | PtpFIRPlanningStart(maxAcc) | | | |  |  |  | | 描述 | 开始Ptp运动FIR滤波 | | | | | | 必选参数 | . maxAcc :最⼤加速度极值(deg/s2) | | | | | | 默认参数 | ⽆ | | | | | | 返回值 | 错误码 成功-0 失败- errcode | | | | | |  |
| **2.23.** 关闭**Ptp**运动**FIR**滤波 |  |
|  | latest |
| *在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| PtpFIRPlanningEnd() |
|  | |
| 描述 | 关闭Ptp运动FIR滤波 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.23.1.** 代码示例

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  startdescPose = [-569.710, -132.595, 395.147, 178.418, -1.893, 171.051]  startjointPos = [-2.334, -79.300, 108.196, -120.594, -91.790, -83.386]  enddescPose = [-366.397, -572.427, 418.339, -178.972, 1.829, -142.970]  endjointPos = [43.651, -70.284, 91.057, -109.075, -88.768, -83.382]  exaxisPos = [0, 0, 0, 0]  offdese = [0, 0, 0, 0, 0, 0]  *# Ptp运动FIR滤波开启*  robot.PtpFIRPlanningStart(maxAcc=1000)  robot.MoveJ(startjointPos, 0, 0,vel=50)  robot.MoveJ(endjointPos, 0, 0,vel=50)  robot.PtpFIRPlanningEnd() |

**2.24.** 开始**LIN**、**ARC**运动**FIR**滤波

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |  |
| LinArcFIRPlanningStart(maxAccLin,maxAccDeg,maxJerkLin,maxJerkDeg) | | | |
|  |  |  |  |
| 描述 | 开始LIN 、ARC运动FIR滤波 | | | | | |
| 必选参数 | . maxAccLin :线加速度极值(mm/s2)  . maxAccDeg :角加速度极值(deg/s2)  . maxJerkLin :线加加速度极值(mm/s3)  . maxJerkDeg :角加加速度极值(deg/s3) | | | | | |
| 默认参数 | ⽆ | | | | | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | | |

**2.25.** 关闭**LIN**、**ARC**运动**FIR**滤波

  latest 

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| LinArcFIRPlanningEnd() |
|  | |
| 描述 | 关闭LIN 、ARC运动FIR滤波 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.25.1.** 代码示例

*在* *Python 版本加⼊:* SDK-v2.0.8-3.7.8

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  startdescPose = [-569.710, -132.595, 395.147, 178.418, -1.893, 171.051]  startjointPos = [-2.334, -79.300, 108.196, -120.594, -91.790, -83.386]  enddescPose = [-366.397, -572.427, 418.339, -178.972, 1.829, -142.970]  endjointPos = [43.651, -70.284, 91.057, -109.075, -88.768, -83.382]  exaxisPos = [0, 0, 0, 0]  offdese = [0, 0, 0, 0, 0, 0]  *# LIN、ARC运动FIR滤波开启*  robot.LinArcFIRPlanningStart(5000, 5000, 5000, 5000)  robot.MoveL(startdescPose, 0, 0,vel=100)  robot.MoveL(enddescPose, 0, 0,vel=100)  robot.LinArcFIRPlanningEnd() |

**2.26.** 停⽌运动

*在* *python 版本加⼊:* SDK-v2.1.1

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 |  |  |  |
| StopMove() |
|  |
| 描述 | 停⽌运动 | | |
| 必选参数 | ⽆ | | |
| 默认参数 | ⽆ | | |
| 返回值 | 错误码 成功-0 失败- errcode | | |

**2.27.** 加速度平滑开启

*在* *python 版本加⼊:* SDK-v2.1.1

|  |  |  |  |
| --- | --- | --- | --- |
| 原型 | |  | | --- | | AccSmoothStart(saveFlag\_flag) | |  |
| 描述 | 加速度平滑开启 | |



 latest 

|  |  |
| --- | --- |
| 必选参数 | . saveFlag\_flag ：是否断电保存 |
| 默认参数 | ⽆ |
| 返回值 | 错误码 成功-0 失败- errcode |

**2.28.** 加速度平滑关闭

*在* *python 版本加⼊:* SDK-v2.1.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 原型 |  |  |  |  |  |
| AccSmoothEnd(saveFlag\_flag) | | |
|  |  |  |
| 描述 | 加速度平滑关闭 | | | | |
| 必选参数 |  | | |  | | --- | | saveFlag\_flag | | ：是否断电保存 | |
| 默认参数 | ⽆ | | | | |
| 返回值 | 错误码 成功-0 失败- errcode | | | | |

**2.28.1.** 代码示例

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | **from fairino import** Robot  *# 与机器⼈控制器建⽴连接，* *连接成功返回—个机器⼈对象*  robot = Robot.RPC( I192.168.58.2I)  JP1 = [88.927,-85.834,80.289,-85.561,-91.388,108.718]  DP1 = [88.739,-527.617,514.939,-179.039,1.494,70.209]  JP2 = [27.036,-83.909,80.284,-85.579,-90.027,108.604]  DP2 = [-433.125,-334.428,497.139,-179.723,-0.745,8.437]  error = robot.AccSmoothStart(saveFlag=0)  print("AccSmoothStart return:",error)  error = robot.MoveJ(JP1, tool=0, user=0, vel=100)  error = robot.MoveJ(JP2, tool=0, user=0, vel=100)  error = robot.MoveJ(JP1, tool=0, user=0, vel=100)  error = robot.MoveJ(JP2, tool=0, user=0, vel=100)  error = robot.AccSmoothEnd(saveFlag=0)  print("AccSmoothEnd return:", error) |

