



AUBO-i5

机械臂 Python 接口文档

Original Version 1.0

遨博（北京）智能科技有限公司

目录

1	导入接口库并且实例化机械臂控制类	3
2	初始化（此方法为静态方法）	3
3	反初始化（此方法为静态方法）	3
4	创建控制上下文句柄	3
5	删除控制上下文句柄	4
6	登陆系统	4
7	注销登陆	4
8	初始化机械臂控制全局属性	4
9	设置六个关节的最大加速度	5
10	获取机械臂最大关节加速度	5
11	设置六个关节的最大速度	5
12	获取机械臂最大关节速度	5
13	设置机械臂末端最大线加速度	6
14	获取机械臂末端最大线加速度	6
15	设置机械臂末端最大线速度	6
16	获取机械臂末端最大线速度	6
17	设置机械臂末端最大角加速度	7
18	获取机械臂末端最大角加速度	7
19	设置机械臂末端最大角速度	7
20	获取机械臂末端最大角速度	7
21	机械臂轴动	8
22	机械臂直线运动	8
23	清除所有已经设置的全局路点	8
24	添加全局路点用于轨迹运动	8
25	机械臂轨迹运动	9
26	设置交融半径	9
27	设置圆运动次数	10
28	设置用户坐标系	10
29	设置基座坐标系	11

30	检查用户坐标系是否合理.....	11
31	设置基座坐标系下的相对位移.....	12
32	设置用户坐标系下的相对位移.....	12
33	正解.....	13
34	逆解.....	13
35	基座坐标系转用户坐标系.....	14
36	用户坐标系转基座坐标系.....	15
37	启动机械臂.....	16
38	关闭机械臂.....	16
39	停止机械臂运动.....	17
40	暂停机械臂运动.....	17
41	继续机械臂运动.....	17
42	机械臂碰撞后恢复.....	17
43	获取机械臂服务状态.....	18
44	设置机械臂工作模式.....	18
45	获取机械臂当前工作模式.....	18
46	设置机械臂碰撞等级.....	19
47	判断当前是否已经链接真实机械臂.....	19
48	获取真实机械臂的关节状态.....	19
49	获取机械臂接口板 IO 配置.....	20
50	设置机械臂接口板 IO 状态.....	21
51	获取机械臂接口板 IO 状态.....	22
52	获取机械臂工具端电源状态.....	22
53	设置机械臂工具端电源状态.....	23
54	设置机械臂工具端 IO 类型.....	23
55	获取机械臂工具端电源数值.....	23
56	设置机械臂工具端 IO 状态.....	24
57	获取机械臂工具端 IO 状态.....	24
58	机械臂事件回调函数.....	25

1 导入接口库并且实例化机械臂控制类

```
from robotcontrol import *  
robot = Auboi5Robot()
```

2 初始化（此方法为静态方法）

函数名称:	Auboi5Robot.initialize()
功能描述:	动态库初始化
参数说明:	无
返回值:	成功: 0
	失败: 其他

3 反初始化（此方法为静态方法）

函数名称:	Auboi5Robot.uninitialize()
功能描述:	动态库反初始化
参数说明:	无
返回值:	成功: 0
	失败: 其他

4 创建控制上下文句柄

函数名称:	create_context()
功能描述:	创建控制上下文句柄
参数说明:	无
返回值:	成功: 上下文句柄 RSHD >= 0
	失败: -1

5 删除控制上下文句柄

函数名称:	destory_context(RSHD)
功能描述:	删除控制上下文句柄
参数说明:	RSHD 上下文句柄
返回值:	成功: 0
	失败: 其他

6 登陆系统

函数名称:	login(IP, PORT)
功能描述:	登陆系统
参数说明:	IP 机械臂 IP 地址 PORT 机械臂端口号
返回值:	成功: 0
	失败: 其他

7 注销登陆

函数名称:	logout()
功能描述:	注销登陆
参数说明:	RSHD 上下文句柄
返回值:	成功: 0
	失败: 其他

8 初始化机械臂控制全局属性

函数名称:	init_profile()
功能描述:	初始化机械臂控制全局属性
参数说明:	
返回值:	成功: 0
	失败: 其他

9 设置六个关节的最大加速度

函数名称:	set_joint_maxacc(joint_maxacc=(1.0,1.0,1.0,1.0,1.0,1.0))
功能描述:	设置机械臂最大关节加速度
参数说明:	Acc 六个关节的加速度,单位是弧度/秒
返回值:	成功:0 失败:其他

10 获取机械臂最大关节加速度

函数名称:	get_joint_maxacc()
功能描述:	获取机械臂最大关节加速度
参数说明:	Acc 六个关节的加速度,单位是弧度/秒
返回值:	成功: 六个关节的最大加速度单位(rad/s) 失败: None

11 设置六个关节的最大速度

函数名称:	set_joint_maxvelc(joint_maxvelc=(1.0,1.0,1.0,1.0,1.0,1.0))
功能描述:	设置机械臂最大关节速度
参数说明:	Acc 六个关节的速度,单位是弧度/秒
返回值:	成功:0 失败:其他

12 获取机械臂最大关节速度

函数名称:	get_joint_maxvelc()
功能描述:	获取机械臂最大关节速度
参数说明:	velc 六个关节的速度,单位是弧度/秒
返回值:	成功返回:六个关节的最大速度(rad/s) 失败返回:None

13 设置机械臂末端最大线加速度

函数名称:	set_end_max_line_acc(end_maxacc=0.1)
功能描述:	设置机械臂末端加速度
参数说明:	end_maxacc 末端加速度,单位是米/秒
返回值:	成功:0
	失败:其他

14 获取机械臂末端最大线加速度

函数名称:	get_end_max_line_acc()
功能描述:	获取机械臂最大末端加速度
参数说明:	
返回值:	成功返回:机械臂末端最大加速度, 单位(m/s)成功:0
	失败返回:None

15 设置机械臂末端最大线速度

函数名称:	set_end_max_line_velc(end_maxvelc=0.1)
功能描述:	设置机械臂末端最大线速度
参数说明:	end_maxacc:末端最大速度, 单位(m/s)
返回值:	成功返回:0
	失败:其他

16 获取机械臂末端最大线速度

函数名称:	get_end_max_line_velc()
功能描述:	获取机械臂末端最大线速度
参数说明:	
返回值:	成功返回:机械臂末端最大速度, 单位(m/s)
	失败返回:None

17 设置机械臂末端最大角加速度

函数名称:	set_end_max_angle_acc(end_maxacc=0.1)
功能描述:	设置机械臂末端加速度
参数说明:	end_maxacc:末端最大加速度, 单位(rad/s)
返回值:	成功:0
	失败:其他

18 获取机械臂末端最大角加速度

函数名称:	get_end_max_angle_acc()
功能描述:	获取机械臂最大末端加速度
参数说明:	
返回值:	成功返回:成功返回:机械臂末端最大加速度, 单位(rad/s)
	失败返回:None

19 设置机械臂末端最大角速度

函数名称:	set_end_max_angle_velc(end_maxvelc=0.1)
功能描述:	设置机械臂末端最大角速度
参数说明:	end_maxacc:end_maxacc:末端最大速度, 单位(rad/s)
返回值:	成功返回:0
	失败:其他

20 获取机械臂末端最大角速度

函数名称:	get_end_max_angle_velc()
功能描述:	获取机械臂末端最大角速度
参数说明:	
返回值:	成功返回:机械臂末端最大角速度, 单位(rad/s)
	失败返回:None

21 机械臂轴动

函数名称:	move_joint(joint_radian=(0.000000,0.000000,0.000000,0.000000,0.000000,0.000000))
功能描述:	机械臂轴动
参数说明:	Joint_radian:六个关节的关节角单位:弧度
返回值:	操作结果:result:
	成功:RobotErrorType.RobotError_SUCC

22 机械臂直线运动

函数名称:	move_line(joint_radian(1.0,1.0,1.0,1.0,1.0,1.0))
功能描述:	从当前位置机械臂直线运动到目标位置
参数说明:	RSHD 上下文句柄
返回值:	Joint_radian:六个关节的关节角单位:弧度
	成功返回:0

23 清除所有已经设置的全局路点

函数名称:	remove_all_waypoint()
功能描述:	清除所有已经设置的全局路点
参数说明:	
返回值:	Joint_radian:六个关节的关节角 单位:弧度

24 添加全局路点用于轨迹运动

函数名称:	add_waypoint(joint_radian(1.0,1.0,1.0,1.0,1.0,1.0))
功能描述:	添加全局路点用于轨迹运动
参数说明:	Joint_radian:六个关节的关节角单位:弧度
返回值:	成功:0
	失败:其他

25 机械臂轨迹运动

函数名称:	move_track(track_type)
功能描述:	使用 add_waypoint 添加路点后,进行轨迹运动
参数说明:	track_type:轨迹运动类型 圆弧运动 RobotMoveTrackType.ARC_CIR 轨迹运动 RobotMoveTrackType.CARTESIAN_MOVEP
轨迹运动类型:	2:ARC_CIR 至少 3 个点 整圆或圆弧,通过 circular 数区分 0 为圆弧 大于 0 为圆 3:CARTESIAN_MOVEP 至少三个点,必须设置交融半径, 交融半径必须小于最短线段长度的 1/2,速度不宜高 以下方法暂不提供: 以下四种三阶样条插值曲线都有起始和结束点加速度不连续的情况, 不适合与新关节驱动版本 4:CARTESIAN_CUBICSPLINE 三次样条插值(过控制点),自动优化轨迹运行时间,目前不支持姿态变化; 5:CARTESIAN_UBSPLINEINTP 需要设定三次均匀 B 样条插值(过控制点)的时间间隔,目前不支持姿态变化 6:JOINT_CUBICSPLINE 7:JOINT_UBSPLINEINTP 用于轨迹回放
返回值:	成功: 0 失败: 其他

26 设置交融半径

函数名称:	set_blend_radius(blend_radius=0.01)
功能描述:	设置交融半径单位:米
参数说明:	blend_radius:交融半径, 单位(m)
返回值:	操作结果:result: 成功:0 失败: 其他

27 设置圆运动次数

函数名称:	set_circular_loop_times(circular_count=1)
功能描述:	设置圆运动次数
参数说明:	当 circular_count 大于 0 时, 机械臂进行圆运动 circular_count 次 当 circular_count 等于 0 时, 机械臂进行圆弧轨迹运动
返回值:	操作结果:result:
	成功:0
	失败:其他

28 设置用户坐标系

函数名称:	set_user_coord(user_coord)
功能描述:	设置用户坐标系
参数说明:	<p>RSHD 上下文句柄</p> <p>user_coord 用户坐标系</p> <p>user_coord = {'coord_type': 2, 坐标系类型: 必须是 2, 指定用户坐标系</p> <p>'calibrate_method': 0, 坐标系标定方法</p> <p>'calibrate_points': 坐标系标定点</p> <p>{ "point1": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0),</p> <p>"point2": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0),</p> <p>"point3": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0)},</p> <p>'tool_desc': 工具描述</p> <p>{ "pos": (0.0, 0.0, 0.0), 工具相对末端坐标系的位置</p> <p>"ori": (1.0, 0.0, 0.0, 0.0)} 工具相对末端坐标系的姿态</p> <p>}</p>
返回值:	操作结果:result:
	成功:0
	失败:其他

29 设置基座坐标系

函数名称:	set_base_coord()
功能描述:	设置基座坐标系
参数说明:	
返回值:	操作结果: result:
	成功:0
	失败:其他

30 检查用户坐标系是否合理

函数名称:	check_user_coord(user_coord)
功能描述:	设置用户坐标系
参数说明:	<p>user_coord 用户坐标系</p> <p>user_coord = {'coord_type': 2, 坐标系类型: 必须是 2, 指定用户坐标系 'calibrate_method': 0, 坐标系标定方法 'calibrate_points': 坐标系标定点 {"point1": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point2": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point3": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0)}, 'tool_desc': 工具描述 {"pos": (0.0, 0.0, 0.0), 工具相对末端坐标系的位置 "ori": (1.0, 0.0, 0.0, 0.0)} 工具相对末端坐标系的姿态 }</p>
返回值:	操作结果: result:
	成功:0
	失败:其他

31 设置基座坐标系下的相对位移

函数名称:	set_relative_offset_on_base(relative(x,y,z))
功能描述:	设置基座坐标系下的相对位移
参数说明:	Relative x,y,z 相对位移量 单位米
返回值:	操作结果: result:
	成功:0
	失败:其他

32 设置用户坐标系下的相对位移

函数名称:	set_relative_offset_on_user(relative(x,y,z),user_coord)
功能描述:	设置用户坐标系下的相对位移
参数说明:	<p>RSHD 上下文句柄</p> <p>Relative x,y,z 相对位移量 单位米</p> <p>user_coord 用户坐标系</p> <p>user_coord = {'coord_type': 2, 坐标系类型: 必须是 2, 指定用户坐标系 'calibrate_method': 0, 坐标系标定方法 'calibrate_points': 坐标系标定点 {"point1": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point2": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point3": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0)}, 'tool_desc': 工具描述 {"pos": (0.0, 0.0, 0.0), 工具相对末端坐标系的位置 "ori": (1.0, 0.0, 0.0, 0.0)} 工具相对末端坐标系的姿态 }</p>
返回值:	操作结果:result:
	成功:0
	失败:其他

33 正解

函数名称:	forward_kin(joint_radian(1.0,1.0,1.0,1.0,1.0,1.0))
功能描述:	正解
参数说明:	Joint_radian:六个关节的关节角 单位:弧度
返回值:	正解结果:waypoint {'joint': [1.0, 1.0, 1.0, 1.0, 1.0, 1.0], 'pos':[-0.06403157614989634,-0.4185973810159096, 0.816883228463401], 'ori':[-0.11863209307193756,0.3820514380931854,0.0,0.9164950251579285]}

34 逆解

函数名称:	inverse_kin(joint_radian(1.0, 1.0, 1.0, 1.0, 1.0, 1.0), pos(x,y,z), ori(w,x,y,z))
功能描述:	逆解
参数说明:	joint_radian 起始点六个关节的关节角, 单位(rad) Pos 位置信息 Ori 姿态信息
返回值:	操作结果: result:
	成功:0
	失败:其他
逆解结果:	waypoint {'joint':[1.0,1.0,1.0,1.0,1.0,1.0], 'pos':[-0.06403157614989634,-0.4185973810159096,0.816883228463401], 'ori':[-0.11863209307193756,0.3820514380931854,0.0,0.9164950251579285]}

35 基坐标系转用户坐标系

函数名称:	base_to_user(pos(x,y,z), ori(w,x,y,z), user_coord, tool(x,y,z))
功能描述:	基坐标系转用户坐标系
参数说明:	<p>Pos (x,y,z)基座坐标系下的位置信息 Ori (w,x,y,z)基座坐标系下的姿态信息 user_coord 用户坐标系 user_coord = {'coord_type': 2, 坐标系类型: 必须是 2, 指定用户坐标系 'calibrate_method': 0, 坐标系标定方法 'calibrate_points': 坐标系标定点 {"point1": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point2": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point3": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0)}, 'tool_desc': 工具描述 {"pos": (0.0, 0.0, 0.0), 工具相对末端坐标系的位置 "ori": (1.0, 0.0, 0.0, 0.0)} 工具相对末端坐标系的姿态 }</p> <p>Tool (pos(x,y,z), ori(w,x,y,z))用户工具末端位置和姿态</p>
返回值:	<p>用户坐标系下的位置和位姿: {"pos":(1,2,3), "ori":(1,2,3,4)}</p>

36 用户坐标系转基坐标系

函数名称:	user_to_base(pos(x,y,z), ori(w,x,y,z), user_coord, tool(x,y,z))
功能描述:	用户坐标系转基坐标系
参数说明:	<p>Pos (x,y,z)用户坐标系下的位置信息</p> <p>Ori (w,x,y,z)用户坐标系下的姿态信息</p> <p>user_coord 用户坐标系</p> <p>user_coord = {'coord_type': 2, 坐标系类型: 必须是 2, 指定用户坐标系 'calibrate_method': 0, 坐标系标定方法 'calibrate_points': 坐标系标定点 {"point1": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point2": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), "point3": (0.0, 0.0, 0.0, 0.0, 0.0, 0.0)}, 'tool_desc': 工具描述 {"pos": (0.0, 0.0, 0.0), 工具相对末端坐标系的位置 "ori": (1.0, 0.0, 0.0, 0.0)} 工具相对末端坐标系的姿态 }</p> <p>Tool (pos(x,y,z), ori(w,x,y,z))用户工具末端位置</p>
返回值:	<p>基座坐标系下的位置和位姿:</p> <p>{"pos":(1,2,3),</p> <p>"ori":(1,2,3,4)}</p>

37 启动机械臂

函数名称:	robot_startup(collision, tool_dynamics)																
功能描述:	启动机械臂																
参数说明:	Collision 碰撞等级(0~10) tool_dynamics 动力学参数 { "position": (x, y, z),工具重心 "payload": 1.5 负载,(0~5)公斤 "inertia": (xx,xy,xz,yy,yz,zz)惯量 }																
返回值:	操作结果:result:																
	成功:0																
	失败:其他																
机械臂启动状态:	<p>state</p> <table> <tr> <td>0 ROBOT_SERVICE_READY</td><td>准备</td></tr> <tr> <td>1 ROBOT_SERVICE_STARTING</td><td>正在启动</td></tr> <tr> <td>2 ROBOT_SERVICE_WORKING</td><td>工作</td></tr> <tr> <td>3 ROBOT_SERVICE_CLOSING</td><td>正在关闭</td></tr> <tr> <td>4 ROBOT_SERVICE_CLOSED</td><td>关闭</td></tr> <tr> <td>5 ROBOT_SETVICE_FAULT_POWER</td><td>电源故障</td></tr> <tr> <td>6 ROBOT_SETVICE_FAULT_BRAKE,</td><td>刹车故障</td></tr> <tr> <td>7 ROBOT_SETVICE_FAULT_NO_ROBOT</td><td>没有链接机械臂</td></tr> </table>	0 ROBOT_SERVICE_READY	准备	1 ROBOT_SERVICE_STARTING	正在启动	2 ROBOT_SERVICE_WORKING	工作	3 ROBOT_SERVICE_CLOSING	正在关闭	4 ROBOT_SERVICE_CLOSED	关闭	5 ROBOT_SETVICE_FAULT_POWER	电源故障	6 ROBOT_SETVICE_FAULT_BRAKE,	刹车故障	7 ROBOT_SETVICE_FAULT_NO_ROBOT	没有链接机械臂
0 ROBOT_SERVICE_READY	准备																
1 ROBOT_SERVICE_STARTING	正在启动																
2 ROBOT_SERVICE_WORKING	工作																
3 ROBOT_SERVICE_CLOSING	正在关闭																
4 ROBOT_SERVICE_CLOSED	关闭																
5 ROBOT_SETVICE_FAULT_POWER	电源故障																
6 ROBOT_SETVICE_FAULT_BRAKE,	刹车故障																
7 ROBOT_SETVICE_FAULT_NO_ROBOT	没有链接机械臂																

38 关闭机械臂

函数名称:	robot_shutdown()
功能描述:	关闭机械臂
参数说明:	
返回值:	操作结果:result:
	成功:0
	失败:其他

39 停止机械臂运动

函数名称:	move_stop()
功能描述:	停止机械臂运动
参数说明:	
返回值:	操作结果:result:
	成功:0
	失败:其他

40 暂停机械臂运动

函数名称:	move_pause()
功能描述:	暂停机械臂运动
参数说明:	
返回值:	操作结果:result:
	成功:0
	失败:其他

41 继续机械臂运动

函数名称:	move_continue()
功能描述:	机械臂暂停后继续机械臂运动
参数说明:	
返回值:	操作结果:result:
	成功:0
	失败:其他

42 机械臂碰撞后恢复

函数名称:	collision_recover()
功能描述:	机械臂碰撞后恢复
参数说明:	
返回值:	操作结果:result:
	成功:0
	失败:其他

43 获取机械臂服务状态

函数名称:	get_robot_state()																
功能描述:	获取机械臂服务状态																
参数说明:																	
返回值:	操作结果:result:																
	成功:0																
	失败:其他																
机械臂启动状态:	<p>state</p> <table> <tr> <td>0 ROBOT_SERVICE_READY</td><td>准备</td></tr> <tr> <td>1 ROBOT_SERVICE_STARTING</td><td>正在启动</td></tr> <tr> <td>2 ROBOT_SERVICE_WORKING</td><td>工作</td></tr> <tr> <td>3 ROBOT_SERVICE_CLOSING</td><td>正在关闭</td></tr> <tr> <td>4 ROBOT_SERVICE_CLOSED</td><td>关闭</td></tr> <tr> <td>5 ROBOT_SETVICE_FAULT_POWER</td><td>电源故障</td></tr> <tr> <td>6 ROBOT_SETVICE_FAULT_BRAKE,</td><td>刹车故障</td></tr> <tr> <td>7 ROBOT_SETVICE_FAULT_NO_ROBOT</td><td>没有链接机械臂</td></tr> </table>	0 ROBOT_SERVICE_READY	准备	1 ROBOT_SERVICE_STARTING	正在启动	2 ROBOT_SERVICE_WORKING	工作	3 ROBOT_SERVICE_CLOSING	正在关闭	4 ROBOT_SERVICE_CLOSED	关闭	5 ROBOT_SETVICE_FAULT_POWER	电源故障	6 ROBOT_SETVICE_FAULT_BRAKE,	刹车故障	7 ROBOT_SETVICE_FAULT_NO_ROBOT	没有链接机械臂
0 ROBOT_SERVICE_READY	准备																
1 ROBOT_SERVICE_STARTING	正在启动																
2 ROBOT_SERVICE_WORKING	工作																
3 ROBOT_SERVICE_CLOSING	正在关闭																
4 ROBOT_SERVICE_CLOSED	关闭																
5 ROBOT_SETVICE_FAULT_POWER	电源故障																
6 ROBOT_SETVICE_FAULT_BRAKE,	刹车故障																
7 ROBOT_SETVICE_FAULT_NO_ROBOT	没有链接机械臂																

44 设置机械臂工作模式

函数名称:	set_work_mode(mode)
功能描述:	设置机械臂工作模式
参数说明:	<p>Mode 0 仿真模式</p> <p>1 真实机械臂</p>
返回值:	操作结果:result:
	成功:0
	失败:其他

45 获取机械臂当前工作模式

函数名称:	get_work_mode()
功能描述:	获取机械臂当前工作模式
参数说明:	
返回值:	<p>Mode 0 仿真模式</p> <p>1 真实机械臂</p>

46 设置机械臂碰撞等级

函数名称:	set_collision_class(collision)
功能描述:	设置机械臂碰撞等级
参数说明:	Collision 碰撞等级(0~10)
返回值:	操作结果: result:
	成功: 0

47 判断当前是否已经链接真实机械臂

函数名称:	is_have_real_robot()
功能描述:	判断当前是否已经链接真实机械臂
参数说明:	
返回值:	真实机械臂是否存在
	不存在:0
	存在:1

48 获取真实机械臂的关节状态

函数名称:	get_joint_status()
功能描述:	获取真实机械臂的关节状态
参数说明:	
返回值:	关节状态 {'joint1':{'current':电流(毫安),'voltage':电压(伏特),'temperature':温度(摄氏度)}, 'joint2':{'current':0,'voltage':0.0,'temperature':0}, 'joint3':{'current':0,'voltage':0.0,'temperature':0}, 'joint4':{'current':0,'voltage':0.0,'temperature':0}, 'joint5':{'current':0,'voltage':0.0,'temperature':0}, 'joint6':{'current':0,'voltage':0.0,'temperature':0}}

49 获取机械臂接口板 IO 配置

函数名称:	get_board_io_config(iotype)
功能描述:	获取机械臂接口板 IO 配置
参数说明:	<p>RSHD 上下文句柄</p> <p>Iotype IO 类型:</p> <p>0:ControllerDI, 1:ControllerDO, 2:ControllerAI, 3:ControllerAO, 4:UserDI, 5:UserDO, 6:UserAI, 7:UserAO, 8:ToolPower, 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO,</p>
返回值:	<p>IO 配置:</p> <p>({"id": ID "name": "IO 名字" "addr": IO 地址 "type": IO 类型 "valu": IO 当前值}, { "id": ID "name": "IO 名字" "addr": IO 地址 "type": IO 类型 "valu": IO 当前值}, ...}</p>

50 设置机械臂接口板 IO 状态

函数名称:	set_board_io_status(Iotype, name, value)
功能描述:	设置机械臂接口板 IO 状态
参数说明:	<p>IoType:</p> <p>0:ControllerDI, 1:ControllerDO, 2:ControllerAI, 3:ControllerAO, 4:UserDI, 5:UserDO, 6:UserAI, 7:UserAO, 8:ToolPower, 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO</p> <p>Name:IO 名称 Value:IO 状态</p>
返回值:	操作结果: result:
	成功: 0

51 获取机械臂接口板 IO 状态

函数名称:	get_board_io_status(Iotype, name)
功能描述:	获取机械臂接口板 IO 状态
参数说明:	IoType: 0:ControllerDI, 1:ControllerDO, 2:ControllerAI, 3:ControllerAO, 4:UserDI, 5:UserDO, 6:UserAI, 7:UserAO, 8:ToolPower, 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO Name:IO 名称
返回值:	Value:IO 状态

52 获取机械臂工具端电源状态

函数名称:	get_tool_power_type()
功能描述:	获取机械臂工具端电源状态
参数说明:	
返回值:	Power_type: OUT_0V = 0, OUT_12V = 1, OUT_24V = 2

53 设置机械臂工具端电源状态

函数名称:	set_tool_power_type(power_type)
功能描述:	设置机械臂工具端电源状态
参数说明:	Power_type: OUT_0V = 0, OUT_12V = 1, OUT_24V = 2
返回值:	操作结果: result: 成功: 0

54 设置机械臂工具端 IO 类型

函数名称:	set_tool_io_type(addr, iotype)
功能描述:	设置机械臂工具端 IO 类型
参数说明:	Addr:工具端 IO 地址 TOOL_DIGITAL_IO_0 = 0, TOOL_DIGITAL_IO_1 = 1, TOOL_DIGITAL_IO_2 = 2, TOOL_DIGITAL_IO_3 = 3 IoType:IO 类型 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO
返回值:	操作结果: result: 成功: 0

55 获取机械臂工具端电源数值

函数名称:	get_tool_power_voltage()
功能描述:	获取机械臂工具端电源数值
参数说明:	
返回值:	电压数值:voltage

56 设置机械臂工具端 IO 状态

函数名称:	set_tool_io_status(name, status)
功能描述:	设置机械臂工具端 IO 状态
参数说明:	Addr:工具端 IO 地址 TOOL_DIGITAL_IO_0 = 0, TOOL_DIGITAL_IO_1 = 1, TOOL_DIGITAL_IO_2 = 2, TOOL_DIGITAL_IO_3 = 3 IoType:IO 类型 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO status:IO 数据(0 或 1)
返回值:	操作结果: result: 成功: 0

57 获取机械臂工具端 IO 状态

函数名称:	get_tool_io_status(name)
功能描述:	获取机械臂工具端 IO 状态
参数说明:	Addr:工具端 IO 地址 TOOL_DIGITAL_IO_0=0, TOOL_DIGITAL_IO_1=1, TOOL_DIGITAL_IO_2=2, TOOL_DIGITAL_IO_3=3 IoType:IO 类型 9:ToolDI, 10:ToolDO, 11:ToolAI, 12:ToolAO
返回值:	status:IO 数据(0 或 1)

58 机械臂事件回调函数

函数名称:	setcallback_robot_event(robot_event_callback)
功能描述:	设置机械臂事件回调函数
参数说明:	robot_event_callback Python 下的回调函数名称
返回值:	操作结果: result: 成功: 0
Python 回调函数参数	<pre>{'code': 事件 ID, 'type': 事件类型} 事件类型包括如下: typedef enum{ RobotEvent_armCanbusError, //机械臂 CAN 总线错误 RobotEvent_remoteHalt, //机械臂停止 RobotEvent_remoteEmergencyStop, //机械臂远程急停 RobotEvent_jointError, //关节错误 RobotEvent_forceControl, //力控制 RobotEvent_exitForceControl, //退出力控制 RobotEvent_softEmergency, //软急停 RobotEvent_exitSoftEmergency, //退出软急停 RobotEvent_collision, //碰撞 RobotEvent_collisionStatusChanged, //碰撞状态改变 RobotEvent_tcpParametersSucc, //工具动力学参数设置成功 RobotEvent_powerChanged, //机械臂电源开关状态改变 RobotEvent_ArmPowerOff, //机械臂电源关闭 RobotEvent_mountingPoseChanged, //安装位置发生改变 RobotEvent_encoderError, //编码器错误 RobotEvent_encoderLinesError, //编码器线数不一致 RobotEvent_singularityOverspeed, //奇异点超速 RobotEvent_currentAlarm, //机械臂电流异常 RobotEvent_toolioError, //机械臂工具端错误 RobotEvent_robotStartupPhase, //机械臂启动阶段 RobotEvent_robotStartupDoneResult, //机械臂启动完成结果 RobotEvent_robotShutdownDone, //机械臂关机结果 RobotEvent_atTrackTargetPos, //机械臂轨迹运动到位信号通知 RobotSetPowerOnDone, //设置电源状态完成 RobotReleaseBrakeDone, //机械臂刹车释放完成 RobotEvent_robotControllerStateChaned, //机械臂控制状态改变 RobotEvent_robotControllerError, //机械臂控制错误----一般是算法规划出现问题时返回 RobotEvent_socketDisconnected, //socket 断开连接 RobotEvent_overSpeed, //超速 RobotEvent_algorithmException, //机械臂算法异常 //interface board io event }</pre>

	<pre> RobotEvent_boardIoPoweron, //外部上电信号 RobotEvent_boardIoRunmode, //联动/手动 RobotEvent_boardIoPause, //外部暂停信号 RobotEvent_boardIoStop, //外部停止信号 RobotEvent_boardIoHalt, //外部关机信号 RobotEvent_boardIoEmergency, //外部急停信号 RobotEvent_boardIoRelease_alarm, //外部报警解除信号 RobotEvent_boardIoOrigin_pose, //外部回原点信号 RobotEvent_boardIoAutorun, //外部自动运行信号 //interface board safety io event RobotEvent_safetyIoExternalEmergencyStope, //外部急停输入 01 RobotEvent_safetyIoExternalSafeguardStope, //外部保护停止输入 02 RobotEvent_safetyIoReduced_mode, //缩减模式输入 RobotEvent_safetyIoSafeguard_reset, //防护重置 RobotEvent_safetyIo3PositionSwitch, //三态开关 1 RobotEvent_safetyIoOperationalMode, //操作模式 RobotEvent_safetyIoManualEmergencyStop, //示教器急停 01 RobotEvent_safetyIoSystemStop, //系统停止输入 //robot working event RobotEvent_alreadySuspended, //机械臂暂停 RobotEvent_alreadyStopped, //机械臂停止 RobotEvent_alreadyRunning, //机械臂运行 RobotEvent_exceptEvent = 100, //unknown event robot_event_unknown, //user event RobotEvent_User = 1000, // first user event id RobotEvent_MaxUser = 65535 // last user event id }RobotEventType;</pre>
示例	<pre> import libpyauboi5 # 机械臂事件 def robot_event(event): print(event) # 设置机械臂事件回调函数 libpyauboi5.setcallback_robot_event(robot_event)</pre>