14. 其他接口

14.1. 下载点位表数据库

在 python 版本加入: SDK-v2.0.1

原型	`PointTableDownLoad(point_table_name, save_file_path)`
描述	下载点位表数据库
必选参数	• `point_table_name`： 要下载的点位表名称 pointTable1.db; • `save_file_path`:下载点位表的存储路径 C://test/;
默认参数	无
返回值	错误码 成功-0 失败- errcode

14.1.1. 代码示例

```
1  from fairino import Robot
2
3  # 与机器人控制器建立连接，连接成功返回一个机器人对象
4  robot = Robot.RPC('192.168.58.2')
5  error =
robot.PointTableDownLoad("point_table_a.db","D://Desktop/testPoint/download/")
6  print("PointTableDownLoad错误码:",error)
```

14.2. 上传点位表数据库

在 python 版本加入: SDK-v2.0.1

原型	`PointTableUpLoad(point_table_file_path)`
描述	上传点位表数据库
必选参数	• `point_table_file_path`： 上传点位表的全路径名 C://test/pointTable1.db
默认参数	无
返回值	错误码 成功-0 失败- errcode

14.2.1. 代码示例

```python
from fairino import Robot

# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
error = robot.PointTableUpLoad("D://Desktop/testPoint/point_table_a.db")
print("PointTableUpLoad错误码:",error)
```

14.3. 点位表切换

在 python 版本加入: SDK–v2.0.1

`ableSwitch(point_table_name)`

切换

`int_table_name`：要切换的点位表名称 pointTable1.db,当点位表为空，即”时，表示将lua程序更新为未应

成功–0 失败– errcode

14.3.1. 代码示例

```python
from fairino import Robot

# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
error = robot.PointTableSwitch("point_table_a.db")
print("PointTableSwitch:",error)
```

14.4. 点位表更新lua文件

在 python 版本加入: SDK–v2.0.1

`ableUpdateLua(point_table_name, lua_file_name)`

更新lua文件

`int_table_name`：要切换的点位表名称pointTable1.db,当点位表为空，即”时，表示将lua程序更新为未应

`a_file_name`：要更新的lua文件名称 testPointTable.lua

成功–0 失败– errcode

14.4.1. 代码示例

```
from fairino import Robot
# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
error = robot.PointTableUpdateLua("point_table_a.db","testpoint.lua")
print("PointTableUpdateLua:",error)
```

14.5. 初始化日志参数

在 python 版本加入: SDK–v2.0.2

`h=", file_num=5)`

功能)

接输出；1–缓冲输出；2–异步输出，默认1;
名称必须是xxx.log的形式，如D://Desktop /fairino.log。默认执行程序所在路径，默认名称fairino_ye
1~20个，默认值为5。单个文件上限50M。

14.5.1. 代码示例

```
from fairino import Robot
# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
robot.LoggerInit(output_model=0,file_path="D://Desktop/fairino.log",file_num=3)
robot.SetLoggerLevel(3)
```

14.6. 设置日志过滤等级

在 python 版本加入: SDK–v2.0.2

型	`SetLoggerLevel(lvl=1)`
述	设置日志过滤等级
参数	无
参数	• `lvl`：过滤等级值，值越小输出日志越少, 1–error, 2–warnning, 3–inform, 4–debug,默认
值	错误码 成功–0 失败– errcode

14.6.1. 代码示例

```
from fairino import Robot
# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
robot.LoggerInit(output_model=0,file_path="D://Desktop/fairino.log",file_num=3)
robot.SetLoggerLevel(3)
```

14.7. 设置机器人外设协议

在 python 版本加入: SDK-v2.0.3

	`SetExDevProtocol(protocol)`
	设置机器人外设协议
数	• `protocol`：机器人外设协议号 4096-扩展轴控制卡；4097-ModbusSlave；4098-Modbus
数	无
i	错误码 成功-0 失败- errcode

14.7.1. 代码示例

```
from fairino import Robot
# 与机器人控制器建立连接，连接成功返回一个机器人对象
robot = Robot.RPC('192.168.58.2')
ret =robot.SetExDevProtocol(4098)
print("SetExDevProtocol",ret)
ret =robot.GetExDevProtocol()
print("GetExDevProtocol",ret)
```

14.8. 获取机器人外设协议

在 python 版本加入: SDK-v2.0.3

	`GetExDevProtocol()`
	获取机器人外设协议
数	无
数	无
值	• 错误码 成功-0 失败- errcode; • `protocol`：机器人外设协议号 4096-扩展轴控制卡；4097-ModbusSlav

14.9. 末端传感器配置

在 python 版本加入: SDK–v2.0.5

原型	`AxleSensorConfig(idCompany, idDevice, idSoftware, idBus)`
描述	末端传感器配置
必选参数	• `idCompany`: 厂商，18–JUNKONG；25–HUIDE • `idDevice`: 类型，0–JUNKONG/RYR6T.V1.0 • `idSoftware`: 软件版本，0–J1.0/HuiDe1.0(暂未开放) • `idBus`: 挂载位置，1–末端1号口；2–末端2号口...8–末端8号口(暂未开放)
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.9.1. 代码示例

```python
from fairino import Robot
import time
# 与机器人控制器建立连接，连接成功返回一个机器人对象

robot = Robot.RPC('192.168.58.2')
error = robot.AxleSensorConfig(18,0,0,0)
print("AxleSensorConfig return:", error)

error = robot.AxleSensorConfigGet()
print("AxleSensorConfigGet return:", error)

error = robot.AxleSensorActivate(0)
print("AxleSensorActivate return:", error)
time.sleep(1)
error = robot.AxleSensorActivate(1)
print("AxleSensorActivate return:", error)

while(1):
    error = robot.AxleSensorRegWrite(1, 4, 6, 1, 0, 0, 0)
    print("AxleSensorRegWrite return:", error)
```

14.10. 获取末端传感器配置

在 python 版本加入: SDK–v2.0.5

原型	`AxleSensorConfigGet()`
描述	获取末端传感器配置
必选参数	无
默认参数	无

返回值	• 错误码 成功-0 失败- errcode • `idCompany`：厂商，18-JUNKONG；25-HUIDE • `idDevice`：类型，0-JUNKONG/RYR6T.V1.0

14.11. 末端传感器激活

在 python 版本加入： SDK-v2.0.5

原型	`AxleSensorActivate(actFlag)`
描述	末端传感器激活
必选参数	`actFlag`： 0-复位；1-激活
默认参数	无
返回值	• 错误码 成功-0 失败- errcode • `coord`：坐标系值[x,y,z,rx,ry,rz]

14.12. 末端传感器寄存器写入

在 python 版本加入： SDK-v2.0.5

原型	`AxleSensorRegWrite(devAddr, regHAddr, regLAddr, regNum, data1, data2, isNoBlock)`
描述	末端传感器寄存器写入
必选参数	• `devAddr`：设备地址编号 0-255 • `regHAddr`：寄存器地址高8位 • `regLAddr`：寄存器地址低8位 • `regNum`：寄存器个数 0-255 • `data1`：写入寄存器数值1 • `data2`：写入寄存器数值2 • `isNoBlock`：是否阻塞 0-阻塞；1-非阻塞
默认参数	无
返回值	错误码 成功-0 失败- errcode

14.13. 设置SmartTool停止/暂停后输出是否复位

在 python 版本加入： SDK-v2.0.5

原型	`SetOutputResetSmartToolDO(resetFlag)`
描述	设置SmartTool停止/暂停后输出是否复位

必选参数	• `resetFlag`：是否复位，0–不复位，1–复位
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.14. 获取末端通讯参数

在 python 版本加入: SDK–v2.0.5

`GetAxleCommunicationParam()`

获取末端通讯参数

无

无

- 错误码 成功–0 失败– errcode
- `baudRate`：波特率:支持1–9600，2–14400，3–19200，4–38400，5–56000，6–67600，7–1152
- `dataBit`：数据位:数据位支持（8,9），目前常用为 8
- `stopBit`：停止位:1–1，2–0.5，3–2，4–1.5，目前常用为 1
- `verify`：校验位:0–None，1–Odd，2–Even,目前常用为 0
- `timeout`：超时时间:1~1000ms，此值需要结合外设搭配设置合理的时间参数
- `timeoutTimes`：超时次数:1~10，主要进行超时重发，减少偶发异常提高用户体验
- `period`：周期性指令时间间隔:1~1000ms，主要用于周期性指令每次下发的时间间隔

14.15. 设置末端通讯参数

在 python 版本加入: SDK–v2.0.5

`SetAxleCommunicationParam(baudRate, dataBit, stopBit, verify, timeout, timeoutTimes, period)`

设置末端通讯参数

- `baudRate`：波特率:支持1–9600，2–14400，3–19200，4–38400，5–56000，6–67600，7–1152
- `dataBit`：数据位:数据位支持（8,9），目前常用为 8
- `stopBit`：停止位:1–1，2–0.5，3–2，4–1.5，目前常用为 1
- `verify`：校验位:0–None，1–Odd，2–Even,目前常用为 0
- `timeout`：超时时间:1~1000ms，此值需要结合外设搭配设置合理的时间参数
- `timeoutTimes`：超时次数:1~10，主要进行超时重发，减少偶发异常提高用户体验
- `period`：周期性指令时间间隔:1~1000ms，主要用于周期性指令每次下发的时间间隔

无

错误码 成功–0 失败– errcode

14.16. 设置末端文件传输类型

在 python 版本加入: SDK–v2.0.5

原型	`SetAxleFileType(type)`
描述	设置末端文件传输类型
必选参数	• `type`：1–MCU升级文件,2–LUA文件
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.17. 设置启用末端LUA执行

在 python 版本加入: SDK–v2.0.5

原型	`SetAxleLuaEnable(enable)`
描述	设置启用末端LUA执行
必选参数	• `enable`：0–不启用；1–启用
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.18. 末端LUA文件异常错误恢复

在 python 版本加入: SDK–v2.0.5

原型	`SetRecoverAxleLuaErr(enable)`
描述	末端LUA文件异常错误恢复
必选参数	• `status`：0–不恢复；1–恢复
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.19. 获取末端LUA执行使能状态

在 python 版本加入: SDK–v2.0.5

原型	`GetAxleLuaEnableStatus()`

描述	获取末端LUA执行使能状态
必选参数	无
默认参数	无
返回值	• 错误码 成功–0 失败– errcode • `enable`：0–不启用；1–启用

14.20. 设置末端LUA末端设备启用类型

在 python 版本加入: SDK–v2.0.5

原型	`SetAxleLuaEnableDeviceType(forceSensorEnable, gripperEnable, IOEnable)`
描述	设置末端LUA末端设备启用类型
必选参数	• `forceSensorEnable`：力传感器启用状态，0–不启用；1–启用 • `gripperEnable`：夹爪启用状态，0–不启用；1–启用 • `IOEnable`：IO设备启用状态，0–不启用；1–启用
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.21. 获取末端LUA末端设备启用类型

在 python 版本加入: SDK–v2.0.5

原型	`GetAxleLuaEnableDeviceType()`
描述	获取末端LUA末端设备启用类型
必选参数	无
默认参数	无
返回值	• 错误码 成功–0 失败– errcode • `forceSensorEnable`：力传感器启用状态，0–不启用；1–启用 • `gripperEnable`：夹爪启用状态，0–不启用；1–启用 • `IOEnable`：IO设备启用状态，0–不启用；1–启用

14.22. 获取当前配置的末端设备

在 python 版本加入: SDK–v2.0.5

原型	`GetAxleLuaEnableDevice()`

描述	获取当前配置的末端设备
必选参数	无
默认参数	无
返回值	• 错误码 成功–0 失败– errcode • `forceSensorEnable[8]`：力传感器启用状态，0–不启用；1–启用 • `gripperEnable[8]`：夹爪启用状态，0–不启用；1–启用 • `IOEnable[8]`：IO设备启用状态，0–不启用；1–启用

14.23. 控制器日志下载

在 python 版本加入: SDK–v2.1.1

原型	`RbLogDownload(savePath)`
描述	控制器日志下载
必选参数	• `savePath`：保存文件路径D://zDown/
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.24. 所有数据源下载

在 python 版本加入: SDK–v2.1.1

原型	`AllDataSourceDownload(savePath)`
描述	所有数据源下载
必选参数	• `savePath`：保存文件路径D://zDown/
默认参数	无
返回值	错误码 成功–0 失败– errcode

14.25. 数据备份包下载

在 python 版本加入: SDK–v2.1.1

原型	`DataPackageDownload(savePath)`
描述	数据备份包下载
必选参数	• `savePath`：保存文件路径D://zDown/
默认参数	无

返回值	错误码 成功–0 失败– errcode

14.26. 获取机器人状态

在 python 版本加入: SDK–v2.1.1

原型	`GetRobotRealTimeState()`
描述	获取机器人状态
必选参数	无
默认参数	无
返回值	• 错误码 成功–0 失败– errcode • `robot_state_pkg`：机器人状态结构体

14.27. 获取控制箱SN码

在 python 版本加入: SDK–v2.1.1

原型	`GetRobotSN()`
描述	获取控制箱SN码
必选参数	无
默认参数	无
返回值	• 错误码 成功–0 失败– errcode • `SNCode`：控制箱SN码

14.28. 关闭机器人操作系统

在 python 版本加入: SDK–v2.1.1

原型	`ShutDownRobotOS()`
描述	关闭机器人操作系统
必选参数	无
默认参数	无
返回值	错误码 成功–0 失败– errcode