目录

1. SDK 文档说明

Python 版本说明:

通过钧舵官网下载 python 版本压缩包后并解压，然后对应安装其中的 whl 文件(pip install JodellTool-0.1.2-py3-none-any.whl），安装完成后，可在 py 文件中直接导入，导入方式如下:

```
from jodellSdk.jodellSdkDemo import EpgClawControl # 导入EPG类型
from jodellSdk.jodellSdkDemo import RgClawControl # 导入RG类型
from jodellSdk.jodellSdkDemo import EpgHpClawControl # 导入EPG-HP类型
from jodellSdk.jodellSdkDemo import LepgClawControl # 导入LEPG类型
from jodellSdk.jodellSdkDemo import EvsClawControl_01 # 导入EVS01类型
from jodellSdk.jodellSdkDemo import EvsClawControl_08 # 导入EVS08类型
from jodellSdk.jodellSdkDemo import ErgClawControl # 导入ERG32类型
from jodellSdk.jodellSdkDemo import ElsClawControl # 导入ELS类型
```

选择对应机型进行导入并创建操作对象，例如导入 EPG 类型:

```
from jodellSdk.jodellSdkDemo import EpgClawControl # 导入EPG类型
clawControl = EpgClawControl()
```

2. 通用函数说明

2.1. 串口搜索

函数原形:

```
def searchCom(self):
    """
        查询可用的串口
    """
```

使用示例:

comList = clawControl.searchCom()

2.2. 通讯连接

函数原形:

```
def serialOperation(self, com, baudRate, status):
    """
        串口操作
        参数说明:
            com: 串口号
            baudRate: 波特率
            status: 开关状态（打开: True, 关闭: False）
    """
```

使用示例：

flag = clawControl. serialOperation(com, baudRate, True) # 连接

flag = clawControl.serialOperation(com, baudRate, False) # 断开连接

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.3. 读取寄存器

函数原形：

```
def readRegisterData(self, salveId, readMode, address, count):
    """
        读取寄存器中的数据
        参数说明:
            salveId:夹爪站点id
            address:modbus寄存器地址
            readMode:modbus读取状态指令编号03或者04
            count:读取寄存器的个数
    """
```

使用示例：

data = clawControl.readRegisterData(9, 4, 2000, 2) # 从站点为 9 的设备中以 0x04 模式读取首地址为 2000 的两个寄存器数据

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

2.4. 写入寄存器

函数原形：

```
def writeDataToRegister(self, salveId, address, sendCmdBuf):
    """
        将数据写入寄存器
        参数说明:
            salveId: 夹爪站点id
            address: modbus寄存器起始地址
            sendCmdBuf: 写入的数据（格式为列表，列表长度代表写入寄存器个数）
    """
```

使用示例：

flag = clawControl.writeDataToRegister (9, 1000, [1, 2]) # 向站点为 9 号的夹爪，地址为 1000 的

寄存器中写入数据 1，地址为 1001 的寄存器中写入数据 2（[ ]中数据个数代表要写入的寄存器个数）

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.5. 无参模式运行

函数原形：

```
def runWithoutParam(self, salveId, cmdId):
    """
        无参数运行指令
        参数说明:
            salveId: 夹爪站点id
            cmdId: 无参数指令编号
    """
```

使用示例：

flag = clawControl.runWithoutParam(9, 1) # id 为 9 的夹爪以预设命令编号 1 执行

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.6. 有参模式运行

函数原形：

```
def runWithParam(self, salveId, pos, speed, torque):
    """
        有参数运行指令
        参数说明:
            salveId: 夹爪站点id
            pos: 运行目标位置
            speed: 运行中最大速度
            torque: 抓取物体时的力矩
    """
```

使用示例：

flag = clawControl.runWithParam(9, 255, 255, 255) # id 为 9 的夹爪以 255 的力矩，最大 255 的速度运动到 255 的位置

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.7. 获取设备当前温度

函数原形：

```
def getDeviceCurrentTemperature(self, salveId):
    """
        获取设备当前温度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

temp = clawControl.getDeviceCurrentTemperature(9) # 获取站点为 9 号的设备温度
若 temp 的数据格式为列表则操作正常，否则操作异常，异常为 temp 的值

2.8. 获取设备当前电压

函数原形：

```
def getDeviceCurrentVoltage(self, salveId):
    """
        获取设备当前电压
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

voltage = clawControl.getDeviceCurrentVoltage(9) # 获取站点为 9 号的设备电压
若 voltage 的数据格式为列表则操作正常，否则操作异常，异常为 voltage 的值

2.9. 读取软件版本

函数原形：

```
def readSoftwareVersion(self, salveId):
    """
        读取软件版本
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

softwareVersion = clawControl.readSoftwareVersion(9) # 查询 id 为 9 的夹爪的软件版本信息
若 softwareVersion 的数据格式为列表则操作正常，否则操作异常，异常为 softwareVersion 的值

2.10. 从站地址修改

函数原形：

```
def changeSalveId(self, oldId, newId):
    """
        修改夹爪站点ID
        参数说明:
            oldId: 当前的id
            newId: 修改后的id
    """
```

使用示例：

flag = clawControl.changeSalveId(9, 8) # 将夹爪的从站从 9 改为 8
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.11. 从站地址扫描

函数原形：

```
def scanSalveId(self, startId, stopId):
    """
        扫描连接的夹爪ID
        参数说明:
            startId: 起始ID
            stopId: 终止ID
    """
```

使用示例：

salveIdList = clawControl.scanSalveId(1, 10) # 轮询从站地址 1-10

2.12. 修改波特率

函数原形：

```
def changeBaudRate(self, salveId, baudRate):
    """
        修改夹爪波特率
        参数说明:
            salveId: 夹爪站点ID
            baudRate: 目标波特率编号（0:115200, 1: 57600, 2: 38400, 3: 19200, 4: 9600, 5: 4800）
    """
```

使用示例：（ERG26 机型暂不支持修改）

flag = clawControl.changeBaudRate(9, 0) # 将 id 为 9 的夹爪的波特率改为 115200
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

2.13. 写入自定义参数

函数原形：

```
def writeCustomParam(self, salveId, address, pos, speed, torque):
    """
        写入自定义参数
        参数说明:
            salveId:夹爪站点ID
            address:modbus寄存器起始地址
            pos:预设位置
            speed:预设速度
            torque:预设力矩
    """
```

使用示例：

flag = clawControl.writeCustomParam(9，0x03E9，255，255，255)# 将位置 255 速度 255 力 255 的参数写入到从站地址为 9 的设备中，写入数据寄存器的起始地址为 0x03E9。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3. EPG 机型函数说明

兼容所有通用函数，并根据自身实际衍生多个专属函数，具体如下。

3.1. 夹持使能

函数原形：

```
def enableClamp(self, salveId, state):
    """
        夹持使能
        参数说明:
            salveId:夹爪站点id
            state:是否使能(使能:True, 去使能:False)
    """
```

使用示例：

flag = clawControl.enableClamp(9, True)# 使能 id 为 9 的夹爪

flag = clawControl.enableClamp(9, False)# 去使能 id 为 9 的夹爪

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.2. 停止夹爪运动

函数原形：

```
def stopClaw(self, salveId):
    """
        停止夹爪运动
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.stopClaw(9) # 停止从站地址为 9 的夹爪的夹持运动。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.3. 强制打开抱闸

函数原形：

```
def forceOpenBrake(self, salveId, state):
    """
        强制打开抱闸
        参数说明:
            salveId:夹爪站点ID
            state:开关状态(开:True,关:False)
    """
```

使用示例：

flag = clawControl.forceOpenBrake(9，True) # 将从站地址为 9 的夹爪的抱闸强制打开。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.4. 选择模式

函数原形：

```
def switchMode(self, salveId, modeIndex):
    """
        选择模式
        参数说明:
            salveId:夹爪站点ID
            modeIndex:模式编号(0:串口通讯模式, 1:IO模式)
    """
```

使用示例：

flag = clawControl.switchMode(9，1) # 将从站地址为 9 的夹爪的控制模式切换为 IO 控制模式。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.5. 选择 IO 输入状态

函数原形：

```
def switchInputState(self, salveId, state):
    """
        选择IO输入状态
        参数说明:
            salveId:夹爪站点ID
            state:IO输入正反逻辑(False:正逻辑, True:反逻辑)
    """
```

使用示例：

flag = clawControl.switchInputState(9，True) # 将从站地址为 9 的夹爪的 IO 输入状态切换为反逻辑。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.6. 选择 IO 输出状态

函数原形：

```
def switchOutputState(self, salveId, state):
    """
        选择IO输出状态
        参数说明:
            salveId:夹爪站点ID
            state:IO输出正反逻辑(False:正逻辑, True:反逻辑)
    """
```

使用示例：

flag = clawControl.switchOutputState(9，True) # 将从站地址为 9 的夹爪的 IO 输出状态切换为反逻辑。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

3.7. 获取夹持端当前状态

函数原形：

```
def getClampCurrentState(self, salveId):
    """
        获取夹持端当前状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

status = clawControl.getClampCurrentState(9) # 获取站点为 9 号的夹爪夹持端状态

若 status 的数据格式为列表则操作正常，否则操作异常，异常为 status 的值

3.8. 获取夹持端当前位置

函数原形：

```
def getClampCurrentLocation(self, salveId):
    """
        获取夹持端当前位置
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

pos = clawControl.getClampCurrentLocation(9) # 获取站点为 9 号的夹爪位置
若 pos 的数据格式为列表则操作正常，否则操作异常，异常为 pos 的值

3.9. 获取夹持端当前速度

函数原形：

```
def getClampCurrentSpeed(self, salveId):
    """
        获取夹持端当前速度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

speed = clawControl.getClawCurrentSpeed(9) # 获取站点为 9 号的夹爪速度
若 speed 的数据格式为列表则操作正常，否则操作异常，异常为 speed 的值

3.10. 获取夹持端当前力矩

函数原形：

```
def getClampCurrentTorque(self, salveId):
    """
        获取夹持端当前力矩
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

torque = clawControl.getClawCurrentTorque(9) # 获取站点为 9 号的夹爪力矩
若 torque 的数据格式为列表则操作正常，否则操作异常，异常为 torque 的值

4. RG 机型函数说明

同 EPG 机型

5. EPG-HP 机型函数说明

同 EPG 机型

6. LEPG 机型函数说明

兼容 EPG 机型函数，并衍生出自身特有的部分函数，以及重构了部分函数，具体如下：

switchInputState，switchOutputState 在此机型中属于无效函数。

6.1. 碰撞使能

函数原形：

```
def enableCollision(self, salveId, state):
    """
        碰撞使能
        参数说明:
            salveId:夹爪站点ID
            state:使能状态(使能:True, 去使能:False)
    """
```

使用示例：

flag = clawControl.enableCollision(9，True) # 将从站地址为 9 的夹爪进行碰撞使能操作。若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.2. 设置碰撞阈值 1

函数原形：

```
def setCollisionThreshold_1(self, salveId, data):
    """
        设置碰撞阈值1
        参数说明:
            salveId:夹爪站点ID
            data:碰撞阈值
    """
```

使用示例：

flag = clawControl.setCollisionThreshold_1(9，100) # 将从站地址为 9 的夹爪碰撞阈值 1 设置为

100。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.3. 设置碰撞阈值 2

函数原形：

```
def setCollisionThreshold_2(self, salveId, data):
    """
        设置碰撞阈值2
        参数说明:
            salveId:夹爪站点ID
            data:碰撞阈值
    """
```

使用示例：

flag = clawControl.setCollisionThreshold_2(9，100) # 将从站地址为 9 的夹爪碰撞阈值 2 设置为 100。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.4. 设置碰撞阈值 3

函数原形：

```
def setCollisionThreshold_3(self, salveId, data):
    """
        设置碰撞阈值3
        参数说明:
            salveId:夹爪站点ID
            data:碰撞阈值
    """
```

使用示例：

flag = clawControl.setCollisionThreshold_3(9，100) # 将从站地址为 9 的夹爪碰撞阈值 3 设置为 100。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.5. 设置掉落检测阈值

函数原形：

```
def setDropThreshold(self, salveId, data):
    """
        掉落检测阈值设置
        参数说明:
            salveId:夹爪站点ID
            data:掉落检测阈值
    """
```

使用示例：

flag = clawControl.setDropThreshold(9，100) # 将从站地址为 9 的夹爪掉落检测阈值设置为 100。若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.6. 搜索行程模式

函数原形：

```
def searchRange(self, salveId):
    """
        搜索行程
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.searchRange(9) # 将从站地址为 9 的夹爪的使能模式设置为搜索行程模式，在夹爪使能时生效。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.7. 手动保存位置

函数原形：

```
def manualSavePosition(self, salveId):
    """
        手动保存位置
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.manualSavePosition(9) # 将从站地址为 9 的夹爪搜索到的行程手动进行保存。若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.8. 设置预撞击点位置

函数原形：

```
def setPreImpact(self, salveId, data):
    """
        设定预撞击点
        参数说明:
            salveId:夹爪站点ID
            data:设定预撞击点位置
    """
```

使用示例：

flag = clawControl.setPreImpact(9， 10) # 将从站地址为 9 的夹爪预撞击点位置设置为 10。
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

6.9. 设置预撞击点速度

函数原形：

```
def setPreImpactSpeed(self, salveId, data):
    """
        设定预撞击点速度
        参数说明:
            salveId:夹爪站点ID
            data:预撞击点速度
    """
```

使用示例：

flag = clawControl.setPreImpactSpeed(9， 10) # 将从站地址为 9 的夹爪预撞击点速度设置为 10。
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7. EVS01 机型函数说明

兼容所有通用函数，并根据自身实际衍生多个专属函数，具体如下。

runWithoutParam，runWithParam 在此机型中属于无效函数。

7.1. 设备使能

函数原形：

```
def enableDevice(self, salveId, state):
    """
        设备使能
        参数说明:
            salveId:夹爪站点id
            state:是否使能(使能:True, 去使能:False)
    """
```

使用示例：

flag = clawControl.enableDevice(9, True) # 使能 id 为 9 的吸盘

flag = clawControl.enableDevice(9, False) # 去使能 id 为 9 的吸盘

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.2. 无参模式运行

函数原形：

```
def runWithoutParam(self, salveId, state):
    """
        无参数运行指令
        参数说明:
            salveId:夹爪站点id
            state:启动或者停止(启动:True,停止:False)
    """
```

使用示例：

flag = clawControl.runWithoutParam(9, True) # id 为 9 的吸盘开始运行

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.3. 有参模式运行

函数原形：

```
def runWithParam(self, salveId, maxData, minData, timeout, state):
    """
        有参数运行指令
        参数说明:
            salveId:夹爪站点id
            maxData:气压设置上限
            minData:气压设置下限
            timeout:加压超时时间
            state:启动或者停止(启动:True,停止:False)
    """
```

使用示例：

flag = clawControl.runWithParam(9，50，40，5，True) # id 为 9 的吸盘以最大真空度 50，最小真空度 40，超时时间 5s 的参数运行。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.4. 写入自定义参数

函数原形：

```
def writeCustomParam(self, salveId, maxPressure, minPressure, timeout):
    """
        写入自定义参数
        参数说明:
            salveId:夹爪站点ID
            maxPressure:最大气压值
            minPressure:最小气压值
            timeout:超时时间
    """
```

使用示例：

flag = clawControl.writeCustomParam(9，50，40，5) # id 为 9 的吸盘设置自定义参数为最大真空度 50，最小真空度 40，超时时间 5s。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.5. 获取设备当前状态

函数原形：

```
def getDeviceCurrentState(self, salveId):
    """
        获取设备当前状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

state = clawControl.getDeviceCurrentState(9) # 获取站点为 9 号的设备状态

若 state 的数据格式为列表则操作正常，否则操作异常，异常为 state 的值

7.6. 获取设备当前真空度

函数原形：

```
def getDeviceCurrentVacuumDegree(self, salveId):
    """
        获取设备当前真空度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

data = clawControl.getDeviceCurrentVacuumDegree(9) # 获取站点为 9 号的设备的当前真空度

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

7.7. 启动或停止设备

函数原形：

```
def startOrStopDevice(self, salveId, mode, runFlag, breakState):
    """
        启动或停止设备
        参数说明:
            salveId: 夹爪站点id
            mode: 运行模式,0为自动控制,1为高级模式
            runFlag: 启停标志, 0为停止, 1为运行
            breakState: 破真空标志, True为破真空, False为不破真空
    """
```

使用示例：

flag = clawControl.startOrStopDevice(9，1，1，True) # id 为 9 的吸盘以破真空与高级模式启动运行。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.8. 选择模式

函数原形：

```
def switchMode(self, salveId, modeIndex):
    """
        选择模式
        参数说明:
            salveId:夹爪站点ID
            modeIndex:模式编号(0:串口通讯模式, 1:IO模式)
    """
```

使用示例：

flag = clawControl.switchMode(9，1) # 将从站地址为 9 的设备的控制模式切换为 IO 控制模式。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

7.9. 获取设备当前温度

函数原形：

```
def getDeviceCurrentTemperature(self, salveId):
    """
        获取设备当前温度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

temp = clawControl.getDeviceCurrentTemperature(9) # 获取站点为 9 号的设备温度

若 temp 的数据格式为列表则操作正常，否则操作异常，异常为 temp 的值

7.10. 获取设备当前电压

函数原形：

```
def getDeviceCurrentVoltage(self, salveId):
    """
        获取设备当前电压
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

voltage = clawControl.getDeviceCurrentVoltage(9) # 获取站点为 9 号的设备电压

若 voltage 的数据格式为列表则操作正常，否则操作异常，异常为 voltage 的值

8. EVS08 机型函数说明

兼容 EVS01 机型函数，并根据自身实际衍生多个专属函数，具体如下。

runWithoutParam，runWithParam 在此机型中属于无效函数。

8.1. 写入自定义参数

函数原形：

```
def writeCustomParam(self, salveId, channelNo, maxPressure, minPressure, timeout):
    """
        写入自定义参数
        参数说明:
            salveId:夹爪站点ID
            channelNo: 通道号(1: 通道1, 2:通道2)
            maxPressure:最大气压值
            minPressure:最小气压值
            timeout:超时时间
    """
```

使用示例：

flag = clawControl.writeCustomParam(9，1，50，40，5)# id 为 9 的吸盘设置通道 1 的自定义参数为最大真空度 50，最小真空度 40，超时时间 5s。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

8.2. 启动或停止设备

函数原形：

```
def startOrStopDevice(self, salveId, mode, runFlag, channelIndex, breakState):
    """
        启动或停止设备
        参数说明:
            salveId: 夹爪站点id
            mode: 运行模式,0为自动控制,1为高级模式
            runFlag: 启停标志, 0为停止, 1为运行
            channelIndex: 通道个数, 0为所有通道, 1为通道1, 2为通道2
            breakState: 破真空标志, True为破真空, False为不破真空
    """
```

使用示例：

flag = clawControl.startOrStopDevice(9，1，1，1，True)＃id 为 9 的吸盘以破真空与高级模式启动通道 1 运行。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

8.3. 获取设备当前状态

函数原形：

```
def getDeviceCurrentState(self, salveId, channelNo):
    """
        获取设备当前状态
        参数说明:
            salveId:夹爪站点id
            channelNo: 通道号(1: 通道1, 2:通道2)
    """
```

使用示例：

state = clawControl.getDeviceCurrentState(9，1)＃ 获取站点为 9 号的设备通道 1 的状态

若 state 的数据格式为列表则操作正常，否则操作异常，异常为 state 的值

8.4. 获取设备当前真空度

函数原形：

```
def getDeviceCurrentVacuumDegree(self, salveId, channelNo):
    """
        获取设备当前真空度
        参数说明:
            salveId:夹爪站点id
            channelNo: 通道号(1: 通道1, 2:通道2)
    """
```

使用示例：

data = clawControl.getDeviceCurrentVacuumDegree(9，1)# 获取站点为 9 号的设备通道 1 的当前真空度

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

9. ERG32 机型函数说明

兼容所有通用函数，并根据自身实际衍生多个专属函数，具体如下。

9.1. 夹持使能

函数原形：

```
def enableClamp(self, salveId, state):
    """
        夹持使能
        参数说明:
            salveId:夹爪站点id
            state:是否使能(使能:True, 去使能:False)
    """
```

使用示例：

flag = clawControl.enableClamp(9, True) # 使能 id 为 9 的夹爪

flag = clawControl.enableClamp(9, False) # 去使能 id 为 9 的夹爪

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.2. 旋转使能

函数原形：

```
def enableRotate(self, salveId, state):
    """
        夹爪旋转使能
        参数说明:
            salveId:夹爪站点id
            state:是否使能(使能:True,去使能:False)
    """
```

使用示例：

flag = clawControl.enableRotate(9, True) # id 为 9 的夹爪进行旋转使能

flag = clawControl.enableRotate(9, False) # id 为 9 的夹爪的进行旋转去使能

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.3. 无参模式旋转运动

函数原形：

```
def rotateWithoutParam(self, salveId, cmdId):
    """
        无参数旋转指令
        参数说明:
            salveId:夹爪站点id
            cmdId:无参数指令编号
    """
```

使用示例：

flag = clawControl.rotateWithoutParam(9, 1) # id 为 9 的夹爪以预设命令编号 1 执行旋转

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.4. 有参模式旋转运动

函数原形：

```
def rotateWithParam(self, salveId, angle, speed, torque, absStatus, cycleNum=0):
    """
        有参数旋转指令
        参数说明:
            salveId:夹爪站点id
            angle:旋转角度
            speed:旋转时最大速度
            torque:旋转时的力矩
            absState:是否按绝对位置进行旋转(是:True, 否:False)
            cycleNum:圈数（配合绝对位置旋转使用）
    """
```

使用示例：

flag = clawControl.enableRotate(9, 360, 255, 255，False，0) # id 为 9 的夹爪以最大 255 的力矩，最大 255 的速度以相对运动，运动 360°（圈数只在绝对运动时生效）

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.5. 获取夹持端当前状态

函数原形：

```
def getClampCurrentState(self, salveId):
    """
        获取夹持端当前状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

status = clawControl.getClampCurrentStatus(9) # 获取站点为 9 号的夹爪夹持端状态
若 status 的数据格式为列表则操作正常，否则操作异常，异常为 status 的值

9.6. 获取夹持端当前位置

函数原形：

```
def getClampCurrentLocation(self, salveId):
    """
        获取夹持端当前位置
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

pos = clawControl.getClampCurrentLocation(9) # 获取站点为 9 号的夹爪位置
若 pos 的数据格式为列表则操作正常，否则操作异常，异常为 pos 的值

9.7. 获取夹持端当前速度

函数原形：

```
def getClampCurrentSpeed(self, salveId):
    """
        获取夹持端当前速度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

speed = clawControl.getClawCurrentSpeed(9) # 获取站点为 9 号的夹爪速度
若 speed 的数据格式为列表则操作正常，否则操作异常，异常为 speed 的值

9.8. 获取夹持端当前力矩

函数原形：

```
def getClampCurrentTorque(self, salveId):
    """
        获取夹持端当前力矩
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

torque = clawControl.getClawCurrentTorque(9) # 获取站点为 9 号的夹爪力矩
若 torque 的数据格式为列表则操作正常，否则操作异常，异常为 torque 的值

9.9. 停止夹持运动

函数原形：

```
def stopClampMotion(self, salveId):
    """
        停止夹持运动
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.stopClampMotion(9) # id 为 9 的夹爪停止夹持运动

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.10. 获取旋转端当前状态

函数原形：

```
def getRotateCurrentState(self, salveId):
    """
        获取旋转端当前状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

status = clawControl.getRotateCurrentState(9) # 获取站点为 9 号的夹爪旋转端状态

若 status 的数据格式为列表则操作正常，否则操作异常，异常为 status 的值

9.11. 获取旋转端绝对角度

函数原形：

```
def getAbsoluteAngle(self, salveId):
    """
        获取旋转端绝对角度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

angle= clawControl.getAbsoluteAngle(9) # 获取站点为 9 号的夹爪旋转端绝对角度

若 angle 的数据格式为列表则操作正常，否则操作异常，异常为 angle 的值

9.12. 获取旋转端相对角度

函数原形：

```
def getRelativeAngle(self, salveId):
    """
        获取旋转端相对角度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

angle = clawControl.getRelativeAngle(9) # 获取站点为 9 号的夹爪旋转端相对对角度
若 angle 的数据格式为列表则操作正常，否则操作异常，异常为 angle 的值

9.13. 获取旋转端绝对圈数

函数原形：

```
def getAbsoluteNumberOfTurns(self, salveId):
    """
        获取旋转端绝对圈数
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

number = clawControl.getAbsoluteAngle(9) # 获取站点为 9 号的夹爪旋转端绝对圈数
若 number 的数据格式为列表则操作正常，否则操作异常，异常为 number 的值

9.14. 获取旋转端当前速度

函数原形：

```
def getRotateCurrentSpeed(self, salveId):
    """
        获取旋转端当前速度
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

speed = clawControl.getRotateCurrentSpeed(9) # 获取站点为 9 号的夹爪旋转端运动速度
若 speed 的数据格式为列表则操作正常，否则操作异常，异常为 speed 的值

9.15. 获取旋转端当前力矩

函数原形：

```
def getRotateCurrentTorque(self, salveId):
    """
        获取旋转端当前力矩
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

torque = clawControl.getClawCurrentTorque(9) # 获取站点为 9 号的夹爪旋转端力矩
若 torque 的数据格式为列表则操作正常，否则操作异常，异常为 torque 的值

9.16. 停止旋转运动

函数原形：

```
def stopRotateMotion(self, salveId):
    """
        停止旋转运动
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.stopRotateMotion(9) # id 为 9 的夹爪停止旋转运动
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.17. 强制打开抱闸

函数原形：

```
def forceOpenBrake(self, salveId, state):
    """
        强制打开抱闸
        参数说明:
            salveId:夹爪站点ID
            state:开关状态(开:True,关:False)
    """
```

使用示例：

flag = clawControl.forceOpenBrake(9，True) # 将从站地址为 9 的夹爪的抱闸强制打开。
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.18. 旋转端设置零点偏置

函数原形：

```
def setRotateToZero(self, salveId, angle):
    """
        旋转端设置零点偏置
        参数说明:
            salveId:夹爪站点ID
            angle:对零角度
    """
```

使用示例：

flag = clawControl.setRotateToZero(9，60) # 设置从站地址为 9 的夹爪的零点偏置为 60°。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

9.19. 选择模式

函数原形：

```
def switchMode(self, salveId, modeIndex):
    """
        选择模式
        参数说明:
            salveId:夹爪站点ID
            modeIndex:模式编号(0:串口通讯模式, 1:IO模式)
    """
```

使用示例：

flag = clawControl.switchMode(9，1) # 将从站地址为 9 的夹爪的控制模式切换为 IO 控制模式。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10. ELS 机型函数说明

兼容所有通用函数，并根据自身实际衍生多个专属函数，具体如下。

10.1. 设备重启

函数原形：

```
def restartDevice(self, salveId):
    """
        设备重启
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

flag = clawControl.restartDevice(9) # 重启 ID 为 9 的设备
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.2. 参数保存

函数原形：

```
def saveParam(self, salveId):
    """
        参数保存
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

flag = clawControl.saveParam(9) # 将 id 为 9 的设备配置参数进行保存
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.3. 设备使能

函数原形：

```
def enableDevice(self, salveId, state):
    """
        设备使能
        参数说明:
            salveId:夹爪站点id
            state:使能状态(1:使能 0:失能)
    """
```

使用示例：

flag = clawControl.enableDevice(9, 1) # 使能 id 为 9 的设备
若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.4. 动作控制

函数原形：

```
def controlAction(self, salveId, state):
    """
        动作控制
        参数说明:
            salveId:夹爪站点id
            state:动作状态(1:推  0:拉)
    """
```

使用示例：

flag = clawControl.controlAction(9，1) # id 为 9 的设备执行推动作

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.5. 设备复位

函数原形：

```
def resetDevice(self, salveId):
    """
        设备复位
        参数说明:
            salveId:夹爪站点ID
    """
```

使用示例：

flag = clawControl.resetDevice(9) # 复位 id 为 9 的设备

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.6. 示教模式

函数原形：

```
def switchTeachingMode(self, salveId, state):
    """
        示教模式
        参数说明:
            salveId:夹爪站点ID
            state:模式编号(0:关闭示教模式, 1:打开示教模式)
    """
```

使用示例：

flag = clawControl.switchTeachingMode(9，1) # 将 id 为 9 的设备切换成示教模式。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.7. 点动正向

函数原形：

```
def joggingForward(self, salveId, state):
    """
        点动正向
        参数说明:
            salveId:夹爪站点ID
            state:执行状态(1:开始  0:拉停止)
    """
```

使用示例：

flag = clawControl.joggingForward(9，1) # 让 id 为 9 的设备执行正向点动。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.8. 点动反向

函数原形：

```
def joggingReverse(self, salveId, state):
    """
        点动反向
        参数说明:
            salveId:夹爪站点ID
        state:执行状态(1:开始  0:拉停止)
    """
```

使用示例：

flag = clawControl.joggingReverse(9，1) # 让 id 为 9 的设备执行反向点动。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.9. 推速度设置

函数原形：

```
def setPushSpeed(self, salveId, data):
    """
        推速度设置
        参数说明:
            salveId:夹爪站点ID
            data:推速度数值，输入范围0~65535，单位0.01mm/s
    """
```

使用示例：

flag = clawControl.setPushSpeed(9，500) # 设置 id 为 9 的设备的推速度为 5mm/s

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.10. 拉速度设置

函数原形：

```
def setPullSpeed(self, salveId, data):
    """
        拉速度设置
        参数说明:
            salveId:夹爪站点ID
            data:拉速度数值，输入范围0~65535，单位0.01mm/s
    """
```

使用示例：

flag = clawControl.setPullSpeed(9，500) # 设置 id 为 9 的设备的拉速度为 5mm/s。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.11. 推力设置

函数原形：

```
def setPushPower(self, salveId, data):
    """
        推力设置
        参数说明:
            salveId:夹爪站点ID
            data:推力数值，输入范围30~100，单位N
    """
```

使用示例：

flag = clawControl.setPushPower(9，30) # 设置 id 为 9 的设备的推力为 30N。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.12. 拉力设置

函数原形：

```
def setPullPower(self, salveId, data):
    """
        拉力设置
        参数说明:
            salveId:夹爪站点ID
            data:拉力数值，输入范围30~100，单位N
    """
```

使用示例：

flag = clawControl.setPullPower(9，30) # 设置 id 为 9 的设备的拉力为 30N。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.13. 推位置设置

函数原形：

```
def setPushPosition(self, salveId, data):
    """
        推位置设置
        参数说明:
            salveId:夹爪站点ID
            data:推位置数值，输入范围0~65535，单位0.01mm
    """
```

使用示例：

flag = clawControl.setPushPosition(9，3000) # 设置 id 为 9 的设备的推位置设置为 30mm。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.14. 拉位置设置

函数原形：

```
def setPullPosition(self, salveId, data):
    """
        拉位置设置
        参数说明:
            salveId:夹爪站点ID
            data:拉位置数值，输入范围0~65535，单位0.01mm
    """
```

使用示例：

flag = clawControl.setPullPosition(9，3000) # 设置 id 为 9 的设备的拉位置设置为 30mm。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.15. IO 强制输出

函数原形：

```
def ioForcedOutput(self, salveId, data):
    """
        IO强制输出
        参数说明:
            salveId:夹爪站点ID
            data:写入的数据
    """
```

使用示例：

flag = clawControl.ioForcedOutput(9，1) # 设置 id 为 9 的设备 IO 强制输出为 1。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.16. 抱闸强制控制

函数原形：

```
def holdingBrakeForcedControl(self, salveId, data):
    """
        抱闸强制控制
        参数说明:
            salveId:夹爪站点ID
            data:写入的数据(1:强制开, 0:关闭)
    """
```

使用示例：

flag = clawControl.holdingBrakeForcedControl(9，1) # 设置 id 为 9 的设备的抱闸强制打开。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.17. 选择命令源

函数原形：

```
def selectCommandSource(self, salveId, data):
    """
        选择命令源
        参数说明:
            salveId:夹爪站点ID
            data:写入的数据(1:485控制 0:IO控制)
    """
```

使用示例：

flag = clawControl.selectCommandSource(9，1) # 选择 id 为 9 的设备控制命令源为 485 控制。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.18. 修改设备从站地址

函数原形：

```
def changeSalveId(self, oldId, newId):
    """
        修改设备从站地址
        参数说明:
            oldId:当前的id
            newId:修改后的id(输入范围:1~247)
    """
```

使用示例：

flag = clawControl.changeSalveId(9，1) # 将 id 为 9 的设备的从站地址修改为 1。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.19. 修改设备波特率

函数原形：

```
def changeBaudRate(self, salveId, baudRate):
    """
        修改设备波特率
        参数说明:
            salveId:夹爪站点ID
            baudRate:目标波特率编号(0:9600, 1:19200, 2:38400, 3:115200)
    """
```

使用示例：

flag = clawControl.changeBaudRate(9，3) # 将 id 为 9 的设备的波特率修改为 115200。

若 flag 等于 1 则操作正常，否则操作异常，异常为 flag 的值

10.20. 获取设备当前状态

函数原形：

```
def getDeviceCurrentState(self, salveId):
    """
        获取设备当前状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

data = clawControl.getDeviceCurrentState(9) # 读取站点为 9 号的设备的当前状态

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.21. 获取设备动作状态

函数原形：

```
def getDeviceActionState(self, salveId):
    """
        获取设备动作状态
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

data = clawControl.getDeviceActionState(9) # 读取站点为 9 号的设备的动作状态

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.22. 获取设备故障状态

函数原形：

```
def getDeviceErrorState(self, salveId):
    """
        获取设备故障状态
        参数说明:
            salveId:夹爪站点id
        返回值说明:
            bit0: 电机故障
            bit1: 行程错误
            bit2: 搜索超时
            bit3: 推超时
            bit4: 拉超时
    """
```

使用示例：

data = clawControl.getDeviceErrorState(9) # 获取站点为 9 号的设备的故障状态

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.23. 获取设备当前位置

函数原形：

```
def getDeciveCurrentLocation(self, salveId):
    """
        获取设备当前位置
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

data = clawControl.getDeciveCurrentLocation(9) # 获取站点为 9 号的设备当前位置

若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.24. 获取设备当前力矩

函数原形：

```
def getDeviceCurrentTorque(self, salveId):
    """
        获取设备当前力矩
        参数说明:
            salveId:夹爪站点id
    """
```

使用示例：

data = clawControl.getDeviceCurrentTorque(9) # 获取站点为 9 号的设备当前力矩
若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.25. 获取 IO 输入监测

函数原形：

```
def getIoInputMonitoring(self, salveId):
    """
        获取IO输入监测
        参数说明:
            salveId:夹爪站点id
        返回值说明:
            转为2进制，检测IO输入(1bit对应1信号)
    """
```

使用示例：

data = clawControl.getIoInputMonitoring(9) # 获取站点为 9 号的设备 IO 输入监测
若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值

10.26. 获取 IO 输出监测

函数原形：

```
def getIoOutputMonitoring(self, salveId):
    """
        获取IO输出监测
        参数说明:
            salveId:夹爪站点id
        返回值说明:
            转为2进制，检测IO输出(1bit对应1信号)
    """
```

使用示例：

data = clawControl.getIoOutputMonitoring(9) # 获取站点为 9 号的设备 IO 输出监测
若 data 的数据格式为列表则操作正常，否则操作异常，异常为 data 的值