众为创造
UFACTORY

UFACTORY

机械爪 G2 用户 手 册



使用前请仔细阅读本手册

V 2.7.1

1.介绍

1.1 机械爪简要介绍

机械爪G2是机器人末端工具，能够动态抓取物体。

机械爪G2的位置数值范围是：0到840，数值越大，机械爪G2张开程度就越大，反之，数值越小，机械爪G2张开程度就越小。

1.2设置与控制

UFACTORY机械爪G2通过触点式接触或单根线缆与UFACTORY机械臂建立电源和通信。末端为机械爪G2提供24V电源，并实现了机械臂控制器的串行RS485通信。

UFACTORY机械爪G2的默认波特率为2000000。

1.3.安全

警告

操作员在使用机械爪G2之前必须已阅读并理解手册中的所有说明。

注意

术语“操作员”是指负责在机械爪G2上进行以下任何操作的任何人：

- 安装
- 控制
- 维护
- 检查
- 标定
- 编程
- 退役

本文档说明了机械爪G2从安装到运行再到使用的整个生命周期的一般操作。 本文档中的图形和照片是代表性的示例，它们与交付的产品之间可能存在差异。

1.3.1.警告

注意

不遵守警告而使用抓取工具，可能导致操作人员受伤或设备损坏。

警告

- 在操作机器人之前，必须正确固定好夹具。
- 请勿安装或操作已损坏或缺少零件的机械爪。
- 切勿为机械爪通交流电。
- 确保所有接线端子稳定连接在机械臂和机械爪两端。
- 请始终使用建议的电气连接。
- 在初始化机械臂程序之前，请确保没有人在机械臂和机械爪路径中。
- 始终不要超过机械爪的有效载荷。
- 根据您的应用情况，相应设置机械爪的速度。
- 接通电源时，手指和衣服应远离机械爪。
- 请勿在人或动物身上使用机械爪。

1.3.2.风险评估和最终应用

机械爪G2用于工业机器人，最终应用中使用的机器人、抓取器和任何其他设备必须进行风险评估。机器人集成商的责任是确保遵守所有本地安全措施和规定。根据不同的应用，可能存在需要采取额外保护/安全措施的风险，例如，机械爪G2操作的工件可能对操作员具有固有的危险。

1.3.3.用途

机械爪G2用于抓取并临时固定或保持物体。

警告

机械爪G2不适用于对物体或表面施加力。

该产品旨在安装在机器人或其他自动化设备上。

信息

- 始终遵守有关自动化安全和通用机器安全的本地和国家法律，法规和指令。
- 本设备只能在其技术数据范围内使用。产品的任何其他使用均被视为不当和意外使用。
- 对于因任何不当使用或不当使用引起的任何损坏，UFACTORY将不承担任何责任。

2.安装

警告

安装之前：

- 阅读并理解与机械爪G2有关的安全说明。
- 根据发货清单和订单验证包裹。
- 备有需求中列出的所需零件。

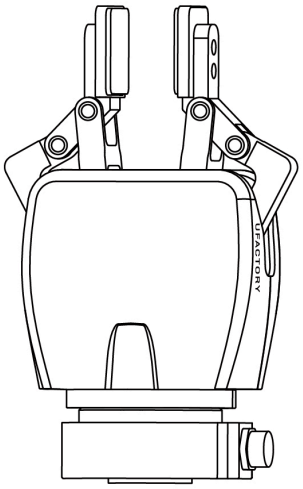
安装时：

- 满足环境条件。
- 在牢固地固定住机械爪并清除危险区域之前，请勿操作机械爪或打开电源。 机械爪的手指可能会移动并造成伤害或损坏。

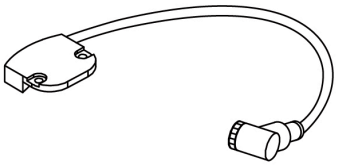
2.1. 发货清单

机械爪G2套件通常包括以下物品（如下图所示）：

- UFACTORY机械爪G2（1个）
- 机械爪G2转接线（备用）
- 十字沉头螺丝M6*12（2个）
- 十字沉头螺丝M6*16（2个）
- 十字沉头螺丝M6*22（2个）



UFACTORY 机械爪 G2



机械爪 G2转接线(备用)



十字沉头螺丝 M6*12(2)

十字沉头螺丝 M6*16(2)

十字沉头螺丝 M6*22(2)

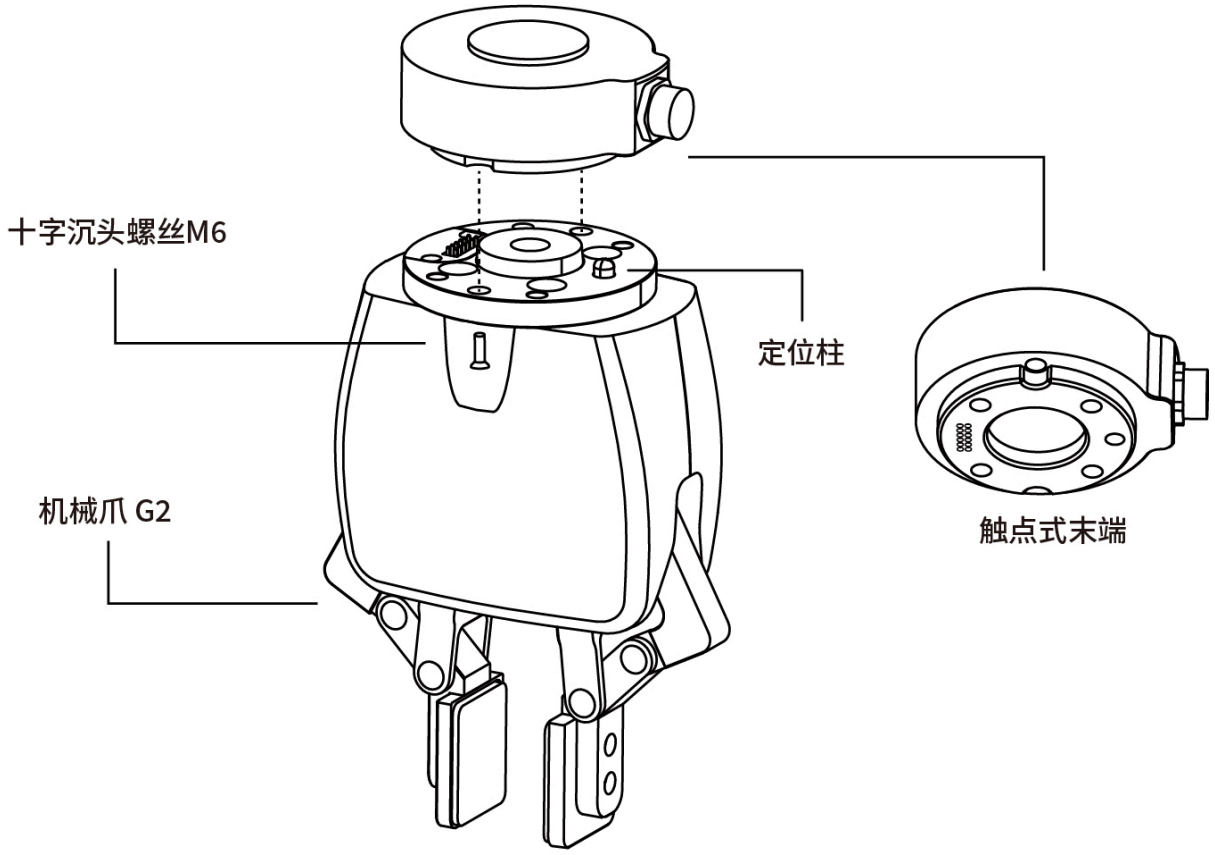
2.2.机械安装

2.2.1 安装准备

1. 使机械臂运动到安全位置（避免碰到机械臂安装表面或者其他设备）；
2. 机械臂断电（按下控制器急停按钮）；

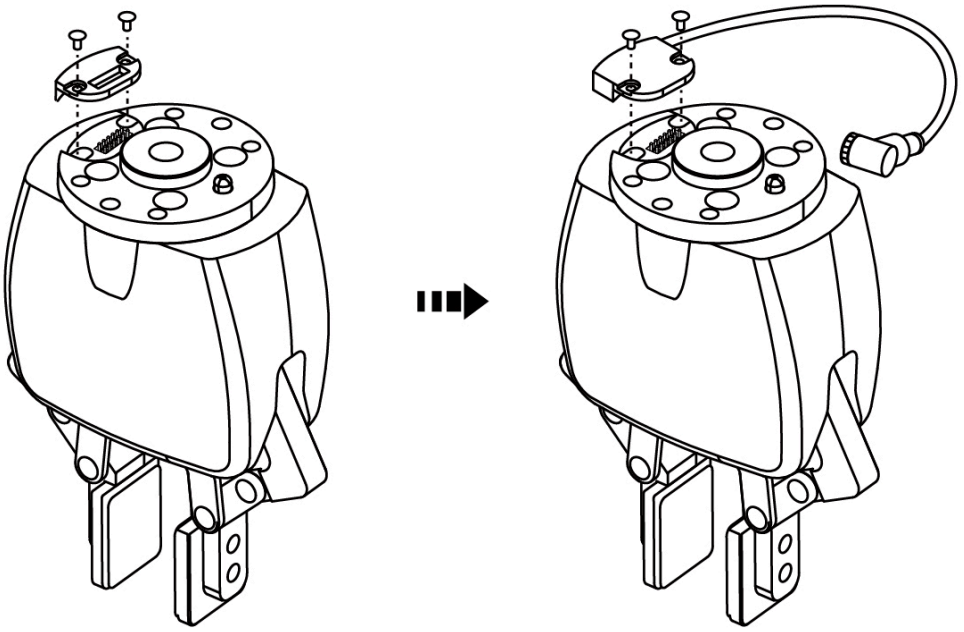
2.2.2 安装

- 若手臂末端是触点式：（UF850,XX1305） 用2颗M6螺丝把机械爪固定在机械臂末端；



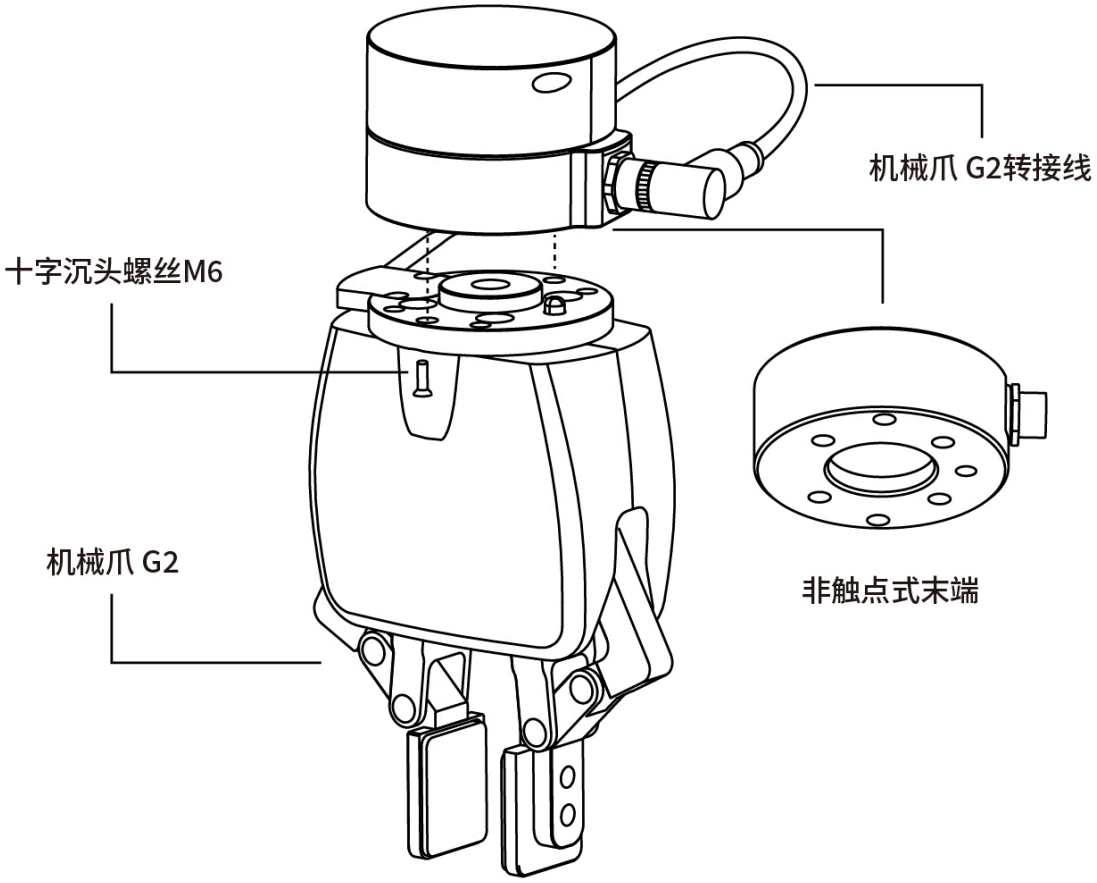
- 若手臂末端不是触点式：（XX1304或以下）

1. 请将机械爪G2法兰面两颗螺丝拧开，取下黑色盖板，替换银色盖板和机械爪G2转接线；



2. 用2颗M6螺丝把机械爪G2固定在机械臂末端；

3. 连接机械臂G2转接线到机械臂末端；



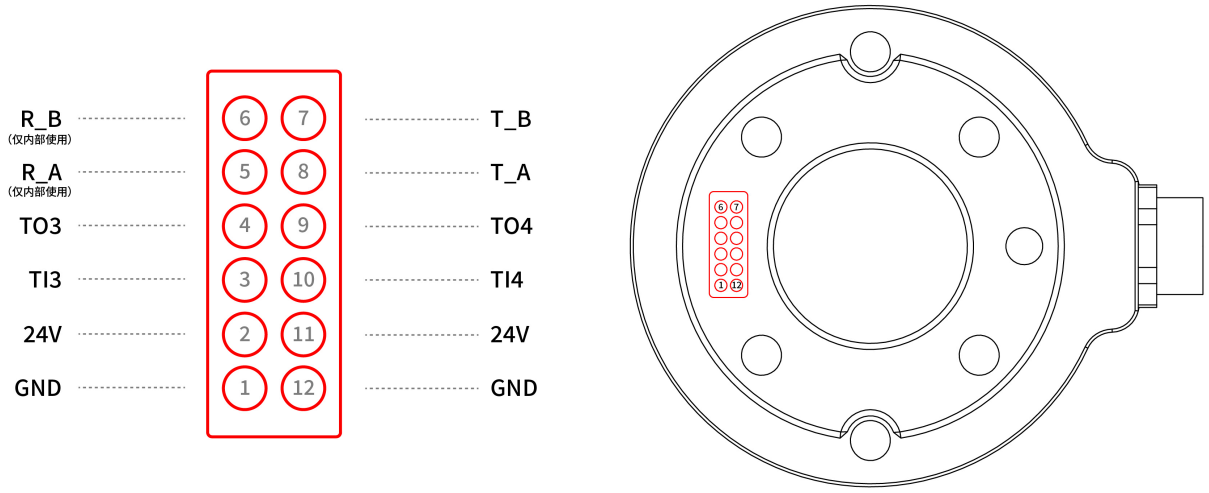
注意：

1. 安装机械爪G2时一定要使机械臂断电，急停开关处于按下状态，机械臂电源指示灯熄灭，避免热插拔引起机械臂故障。
2. 机械爪G2和机械臂接通时注意务必对齐两端接口的定位孔，机械爪G2连接线的公针较为纤细，避免在拆装时使公针弯曲。

2.3.电气设置

2.3.1 触点式接口

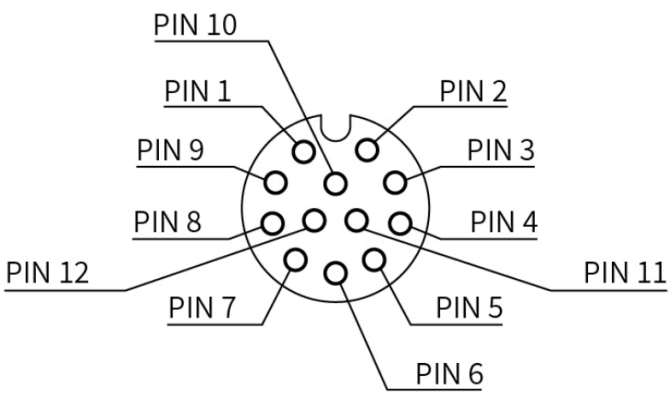
触点式接口定义：

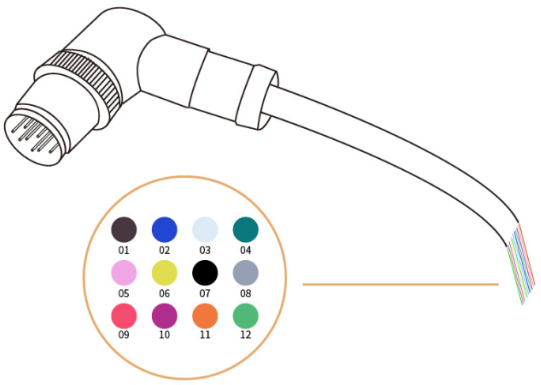


机械爪G2使用的信号有：两个24V，两个GND，T_A，T_B。

2.3.2.航插接口

机械爪G2航插接口如下图所示。





电缆内部的12条线有不同颜色，不同颜色代表不同功能，请参见下表：

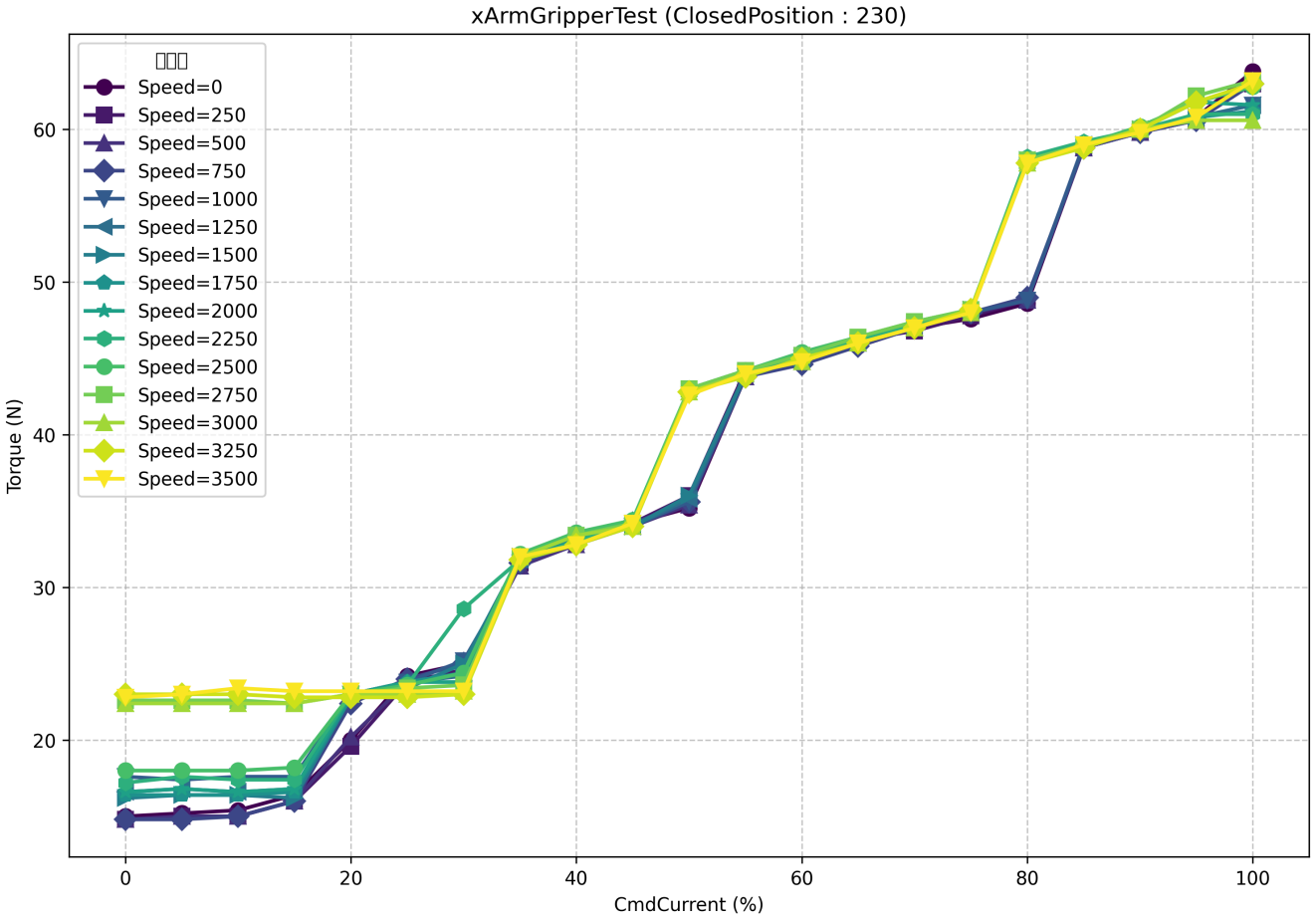
线序	颜色	信号
1	棕	+24V（电源）
2	蓝	+24V（电源）
3	白	0V （GND）
4	绿	0V （GND）
5	粉	用户485-A
6	黄	用户485-B
7	黑	工具输出 0 （TO0）
8	灰	工具输出 1 （TO1）
9	红	工具输入 0 （TI0）
10	紫	工具输入 1 （TI1）
11	橙	模拟输入 0 （AI0）
12	浅绿	模拟输入 1 （AI1）

机械爪G2使用的信号有：24V（PIN1和PIN2），GND（PIN3和PIN4），485A（PIN5），485B（PIN6）。

3.控制方式

可编程参数：

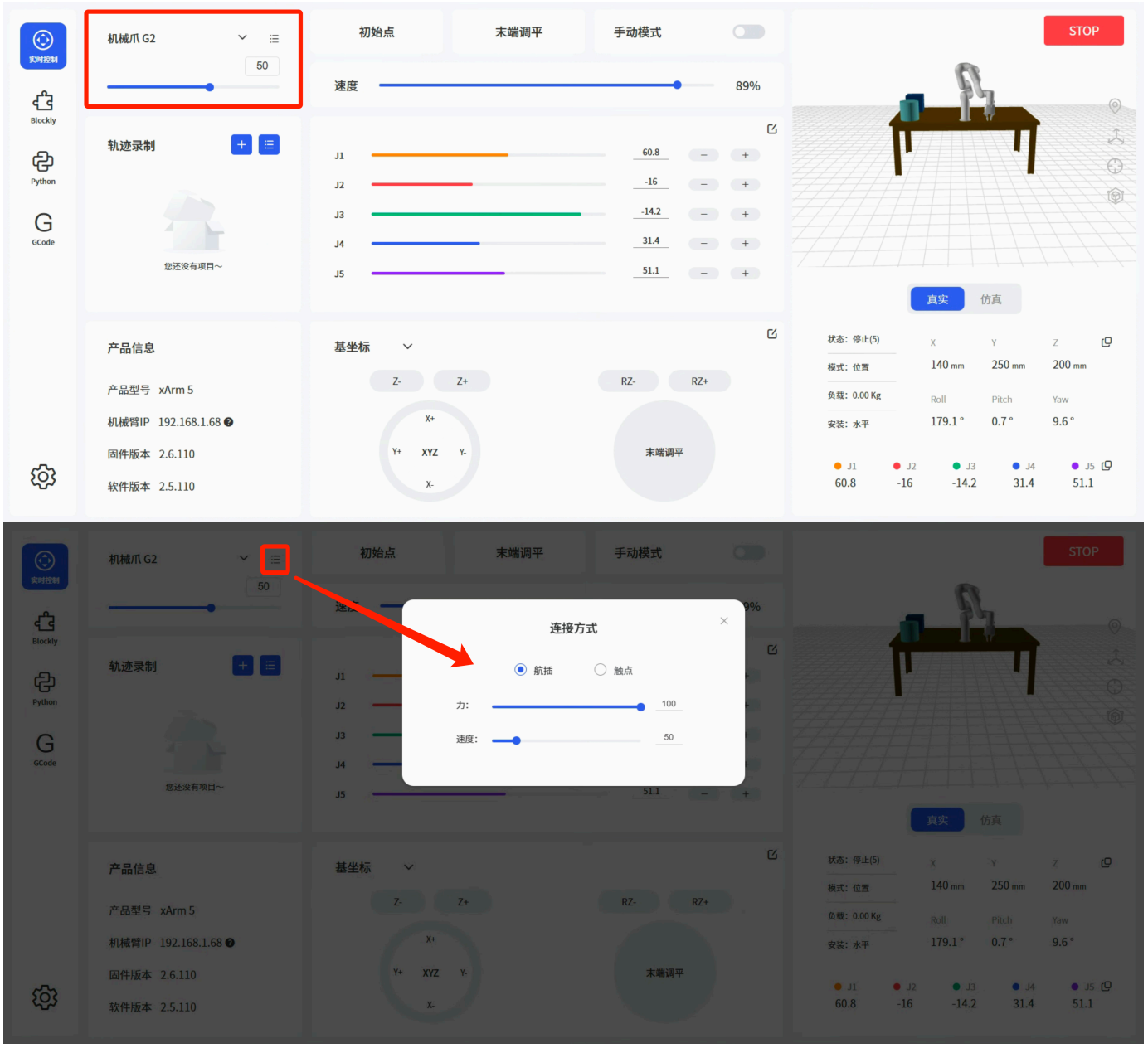
- 位置：0-840mm；
- 速度：15-225mm/s；
- 开合力：1-100%（百分比）；力设定值以百分比表示，对应实际力的大小参考下图：



3.1.用UFACTORY Studio控制

3.1.1 实时控制界面控制

进入实时控制界面，选择机械爪G2，可进行位置控制，范围0-84。 点击右上角按钮，可选择连接方式，调整开合力和速度。



3.1.2 Blockly控制

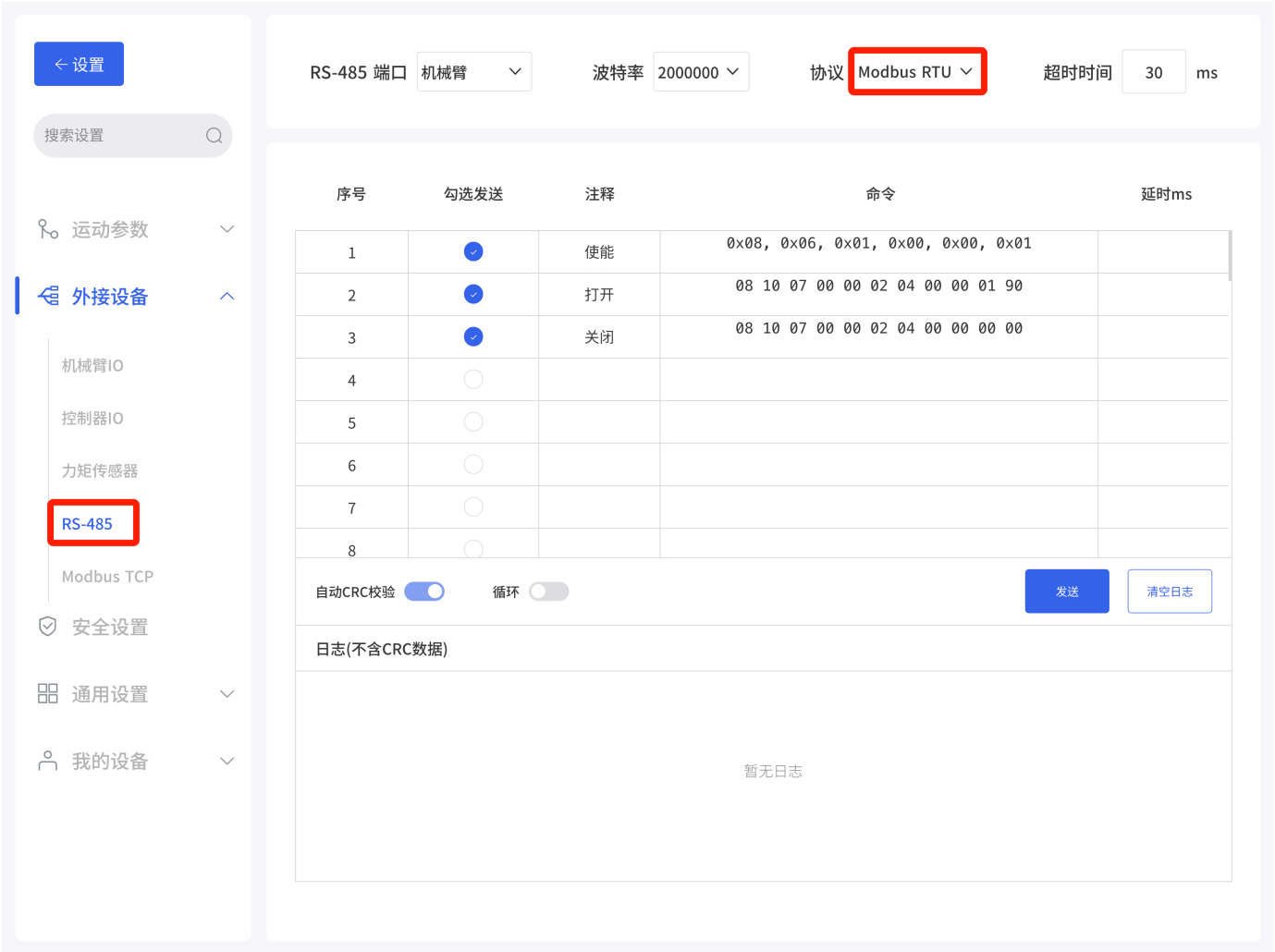
Blockly为机械爪G2提供3个块，如下：

- 设置机械爪G2 位置[] 速度[] 力度[] 等待[]
- 设置机械爪G2[变量] 位置[] 速度[] 力度[] 等待[]
- 物体已被机械爪G2夹住 超时时间[]



3.1.3 Modbus RTU控制

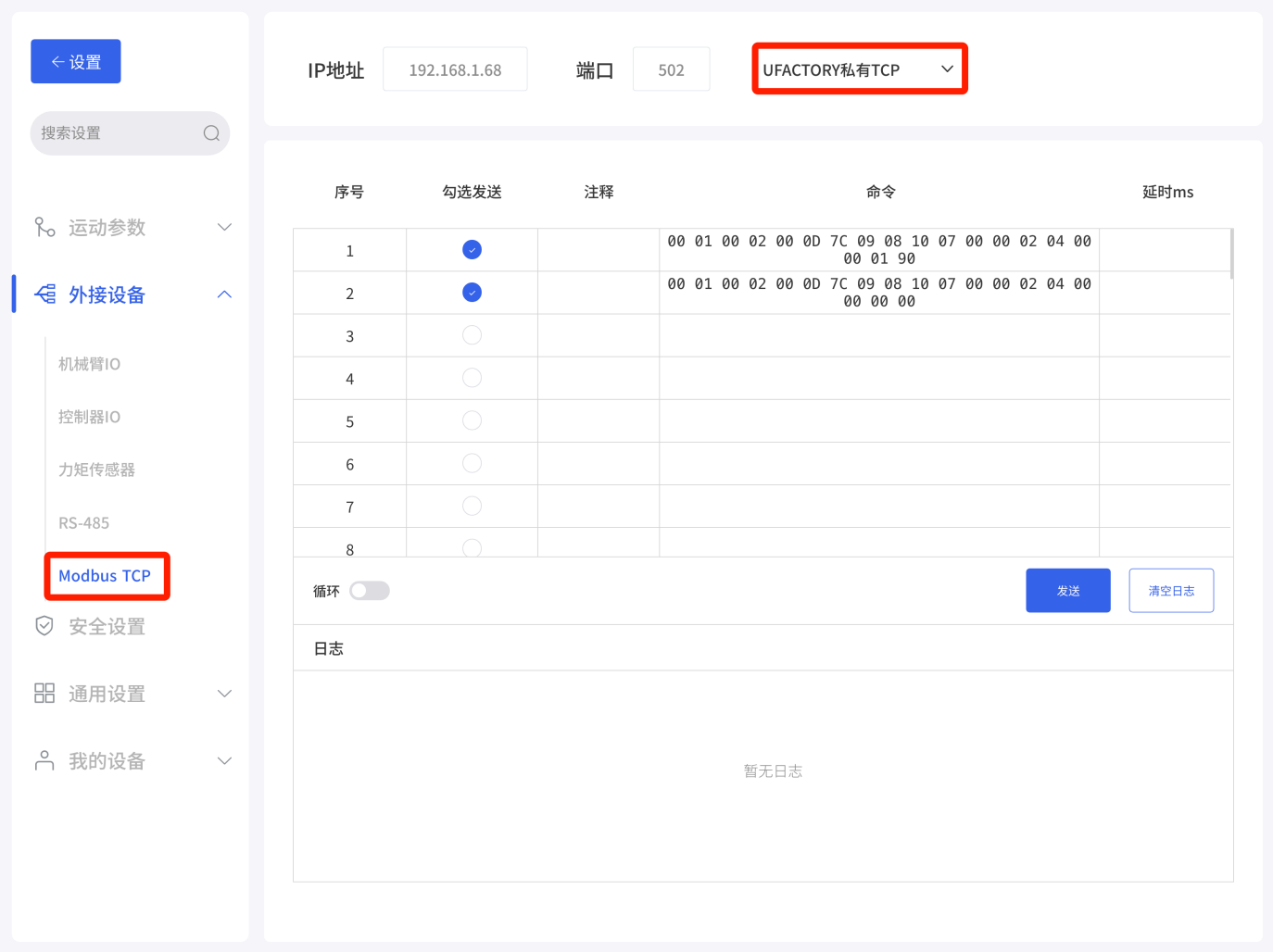
进入设置-外接设备-RS485页面，选择Modbus RTU协议，发送相应的指令进行控制。Modbus通讯协议请参考 第四章Modbus RTU通讯协议。



3.1.4 UFACTORY私有协议控制

进入设置-外接设备-Modbus TCP页面，选择'UFACTORY私有TCP'，发送相应的私有TCP指令进行控制。

UFACTORY私有协议请参考 附录。



3.2 用Python SDK控制

常用接口如下：(Python SDK ≥ 1.16.0) `set_gripper_g2_position`：设置机械爪G2的位置、力、速度； `get_gripper_g2_position`：获取机械爪G2位置。

Python示例：

```python
import os
import sys

sys.path.append(os.path.join(os.path.dirname(__file__), '../../..'))
```

```python
from xarm.wrapper import XArmAPI

arm = XArmAPI('192.168.1.68')
arm.motion_enable(enable=True)
arm.set_mode(0)
arm.set_state(state=0)

code = arm.set_gripper_enable(True)
print('set gripper enable, code={}'.format(code))

code = arm.set_gripper_g2_position(80, wait=True, speed=200, force=80)
print('set gripper, code={}'.format(code))

print('position=', arm.get_gripper_g2_position())
```

4.Modbus RTU通讯协议控制

4.1 寄存器地址说明

机械爪G2支持标准Modbus RTU协议，默认波特率2Mbps，机械爪ID为0x08。目前支持的功能码有：0x03/0x06/0x10。

- 0x03: 读取寄存器
- 0x06: 写单个寄存器
- 0x10: 写多个寄存器

4.1.1 监控组：

通讯地址	说明
0x0000	运行状态
0x0001	速度（r/min）
0x0002	q轴电流
0x0003	母线电流
0x0004	指令位置(p)
0x0006	电机位置(p)
0x0008	位置误差(p)
0x000F	当前报警代码

运动状态（0x0000）

位1:0	00：停止状态	位3:2	00：未使能
	01：运动状态		10：使能状态
	10：夹取状态		

4.1.2 Fn1xx控制相关

编号(地址)	说明	设定范围	单位	出厂设定
Fn100	机械爪使能	0-1		0
Fn109	故障复位	0-1		0

4.1.3 Fn3xx速度参数

编号(地址)	说明	设定范围	单位	出厂设定
Fn303	速度指令	0-0xFFFF	r/min	2000

4.1.4 Fn5xx转矩参数

编号(地址)	说明	设定范围	单位	出厂设定
Fn500	夹取电流指令	0-100	-	50

4.1.5 Fn6xx通讯参数

编号(地址)	说明	设定范围	单位	出厂设定
Fn600	通讯地址	1-255	-	8
Fn601	通讯波特率	0: 4800 1: 9600 2: 192008: 9216009: 1M10: 1.5M11: 2M	bps	11

4.1.6 Fn7xx位置命令组

编号(地址)	说明	设定范围	单位	出厂设定
Fn700	位置命令高位	0-0xFFFF	-	-
Fn701	位置命令低位	0-0xFFFF	-	-
Fn702	位置反馈高位	0-0xFFFF	-	0
Fn703	位置反馈低位	0-0xFFFF	-	0

4.1.7 FnCxx机械爪控制指令

编号(地址)	说明	设定范围	单位	出厂设定
FnC00	机械爪使能	0-1	-	0
FnC01	速度指令	0-0xFFFF	-	2000
FnC02	夹取电流指令	0-100	-	50
FnC03	位置命令高位	0-0xFFFF	-	
FnC04	位置命令低位	0-0xFFFF	-	

4.2 FnCxx寄存器使用说明

FnCxx控制器指令寄存器			
请求指令格式			
Modbus RTU 数据	机械爪 ID	1 Byte	0x08
	功能码	1 Byte	0x10
	寄存器起始地址	2 Bytes	0x0C, 0x00
	寄存器数量	2 Bytes	0x00, 0x05
	字节数	1 Byte	0x0A
	机械爪使能	2 Bytes	0x00, 0x01
	速度指令(3000)	2 Bytes	0x0B, 0xB8
	夹取电流指令(50)	2 Bytes	0x00, 0x32
	位置命令(850)	2 Bytes	0x03, 0x52
	Modbus CRC16	2 Bytes	0x17, 0x4E
响应指令格式			
Modbus RTU 数据	机械爪 ID	1 Byte	0x08

FnCxx控制器指令寄存器			
	功能码	1 Byte	0x10
	寄存器起始地址	2 Bytes	0x0C, 0x00
	寄存器数量	2 Bytes	0x00, 0x05
	Modbus CRC16	2 Bytes	0x03, 0xC3

4.3 通讯格式示例

1. 写机械爪使能。地址(0x0100)

```
发：08 06 01 00 00 01 49 6F
收：08 06 01 00 00 01 49 6F
```

2. 写机械爪位置为100。地址(0x0700)

```
发：08 10 07 00 00 02 04 00 00 00 64 FA E8
收：08 10 07 00 00 02 40 25
```

3. 读取报警代码。地址(0x000F)

```
发：08 03 00 0F 00 01 B4 90
收：08 03 02 00 00 64 45
```

4. 写机械爪使能，位置(850)，速度(3000)，力(50)。地址(0x0Cxx)

```
发：08 10 0C 00 00 05 0a 00 01 0b b8 00 32 03 52 17 4E
收：08 10 0C 00 00 05 03 C3
```

5. 报警与处理

5.1 错误代码

软件报错代码	报警代码	报警处理
G1	0x01	机械爪G2 FOC运行超时，请清除错误后重试。
G2	0x02	机械爪G2 过压。
G3	0x03	机械爪G2 欠压。
G4	0x04	机械爪G2 温度过高。
G6	0x06	机械爪G2 转速反馈故障。
G7	0x07	机械爪G2 电流过大 请点击“确认”重新使能机械爪。
G8	0x08	机械爪G2 MCSDK软件错误。
G9	0x09	机械爪G2 驱动器保护。

上表中未出现的报警代码：重新使能机械臂和机械爪。如频繁出现，请联系技术支持。

5.2 报警处理

5.2.1 通过UFACTORY Studio清除错误

1. 通过控制器上的紧急停止按钮重新对机械臂上电
2. 使能机械臂：点击报错弹窗的引导按钮或者首页的使能机械臂按钮。
3. 选择机械爪G2，并进行控制。



5.2.2 通过xArm-Python-SDK清理错误

清除错误步骤：(Python-SDK接口)

1. 通过控制器上的紧急停止按钮重新对机械臂上电
2. 清除错误：`clean_error()`
3. 重新使能机械臂：`motion_enable(enable=True)`
4. 设置运动模式：`set_mode(0)`
5. 设置运动状态：`set_state(0)`
6. 使能机械爪G2:`set_gripper_enable(enable=True)`

python示例：

```python
import os
import sys
import time

sys.path.append(os.path.join(os.path.dirname(__file__), '../../..'))

from xarm.wrapper import XArmAPI

arm = XArmAPI('192.168.1.68')
arm.clean_error() #清除错误
arm.motion_enable(enable=True) #重新使能机械臂
arm.set_mode(0) #设置运动模式
```

```
arm.set_state(0) #设置运动状态
arm.set_gripper_enable(True)  #使能机械爪G2
```

6.技术规格

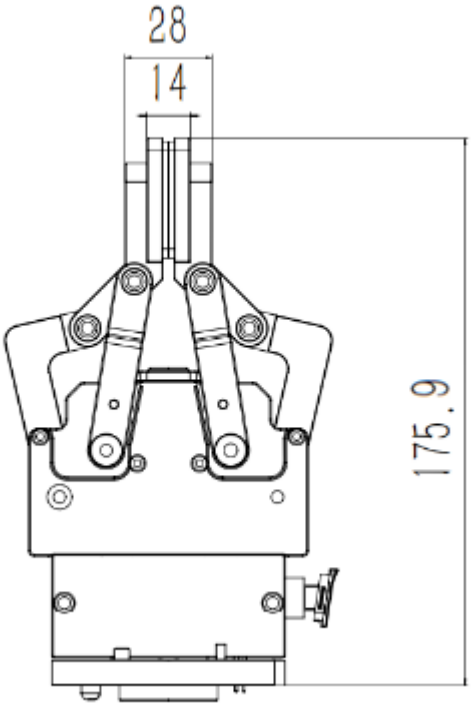
6.1 技术参数

机械爪G2（AG1200）	
额定电压	24V DC 1.5AMax
最大输入电压	28V DC
静态功耗（最低功耗）	1W
重量	820g
负载	5kg
行程	84±1mm
手指形态	可更换
夹持力	10-50N
额定扭矩	0.22 N·m
峰值扭矩	0.5 N·m
扭矩常数Kt	0.09 N·m/A
开合速度	15-225mm/s
寿命	＞2,000,000循环
通信协议	RS-485, Modbus RTU
可编程参数	速度、位置、力矩
反馈	位置
防护等级	IP40
使用环境	0-50℃

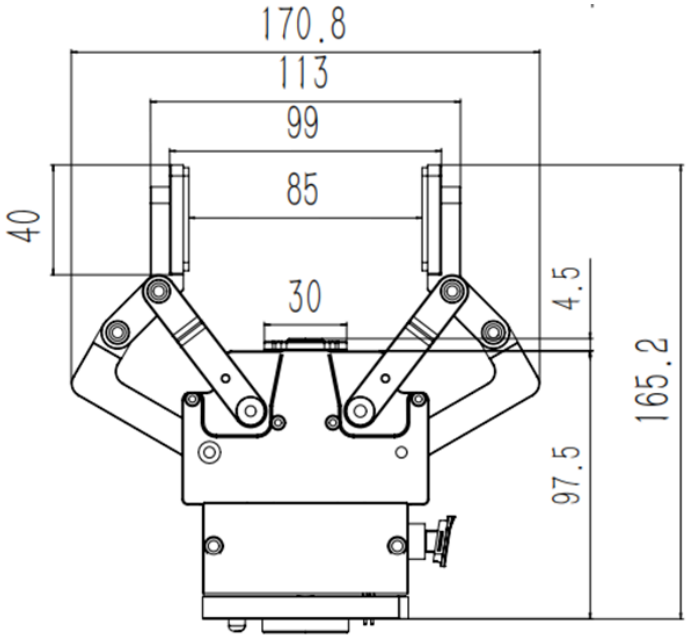
6.2 机械尺寸

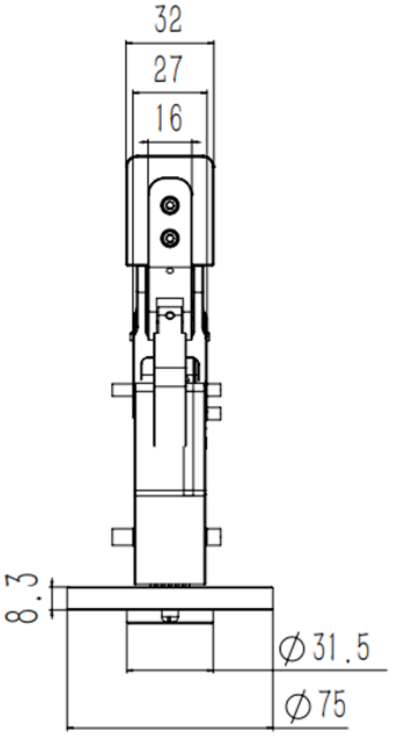
机械爪G2的机械尺寸如下图所示。图中 d 为两指张开距离，L 为机械爪长度。

闭合状态下的尺寸：

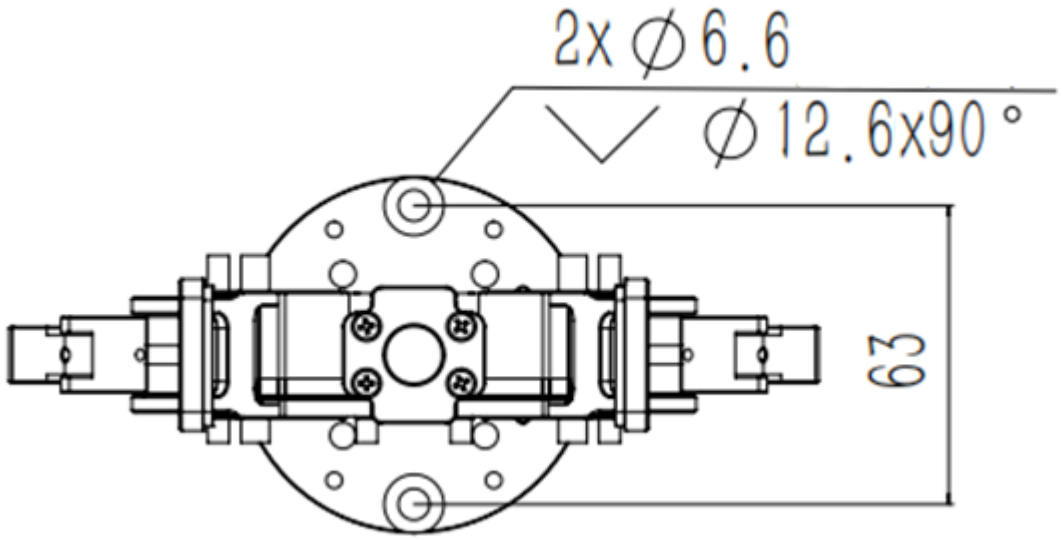


张开状态下的尺寸及侧视图：





底部安装尺寸：

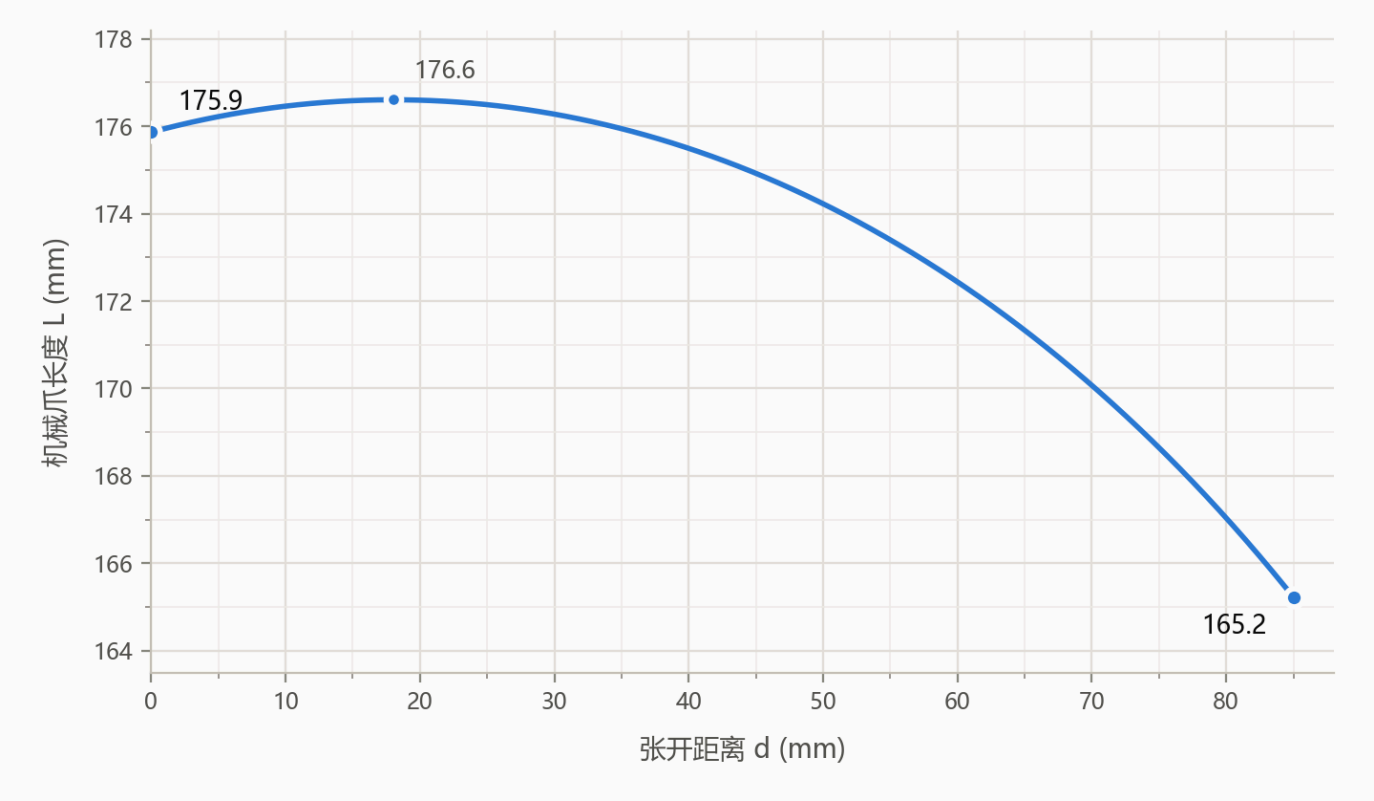


机械爪长度 L 与张开距离 d 的几何关系为：

$$L = 121.606 + \sqrt{55.000^2 - (0.5d - 8.997)^2}$$

式中，L 和 d 的单位均为 mm。例如，d = 0 时 L ≈ 175.9 mm，d = 85 mm 时 L ≈ 165.2 mm，与上图中闭合、张开状态的尺寸标注一致。

L 随 d 变化的曲线如下图所示：



7.售后服务

7.1 售后政策

对于产品的质量保证以及维修和退换货的详情，见官网的售后政策：

https://www.cn.ufactory.cc/warranty

7.2 售后服务流程

（1）联系技术支持（support@ufactory.cc），确认产品需要寄回维修，确定需要寄回的部件。

（2）我司根据售后政策，判定产品保修状况，付费或免费维修。

（3）维修、测试完成后，我们会将产品寄回，一般情况下，整个维修流程大约需要1-2周。

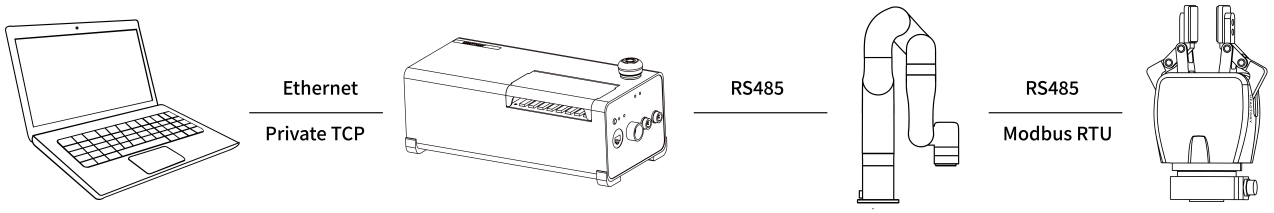
注意：

1.当需要将产品寄回我司进行维修时，需要将产品用包装箱打包好，避免在运输过程中发生不必要的碰撞，导致机械爪受损。

8. 附录-UFACTORY私有协议控制

Modbus协议是一项应用层报文传输协议，有ASCII、RTU、TCP三种报文类型。标准Modbus协议物理层接口有RS232、RS422、RS485和以太网接口，采用master/slave方式通信。

机械爪G2使用UFACTORY私有TCP协议，与标准Modbus TCP类似但不完全相同。



私有TCP通信过程： （1）建立TCP连接 （2）准备私有TCP报文 （3）使用send命令发送报文 （4）在同一连接下等待应答 （5）使用recv命令读取报文，完成一次数据交换 （6）通信任务结束时，关闭TCP连接

参数： 默认TCP端口：**502** 协议标识：0x00 0x02 控制(当前只有这一个) 在本章节中，数据解析均为大端解析。

8.1 读取寄存器

	读取寄存器		
	请求指令格式		
Modbus TCP 包头	事务标识	2 Bytes	0x00，0x01
	协议标识	2 Bytes	0x00，0x02
	长度	2 Bytes	0x00，0x08
	寄存器	1 Byte	0x7C
参数	状态	1 Byte	0x00
内部使用	主机 ID	1 Byte	0x09
Modbus RTU 数据	机械爪 ID	1 Byte	0x08
	功能码	1 Byte	0x03
	寄存器起始地址	2 Bytes	**Address**

	读取寄存器		
	寄存器数量	2 Bytes	**N***
	响应指令格式		
Modbus TCP 包头	事务标识	2 Bytes	0x00，0x01
	协议	2 Bytes	0x00，0x02
	长度	2 Bytes	**6+N*x2**
	寄存器	1 Byte	0x7C
	状态	1 Byte	0x00
内部使用	主机 ID	1 Byte	0x09
Modbus RTU 数据	机械爪 ID	1 Byte	0x08
	功能码	1 Byte	0x03
	字节数	1 Byte	**N*x2**
	寄存器值	Nx2 Bytes	**Value**

注： N* = 寄存器数量

8.2 写入寄存器

	写入寄存器		
	请求指令格式		
Modbus TCP 包头	事务标识	2 Bytes	0x00，0x01
	协议	2 Bytes	0x00，0x02
	长度	2 Bytes	9+Nx2
	寄存器	1 Byte	0x7C
内部使用	主机 ID	1 Byte	0x09

	写入寄存器		
Modbus RTU 数据	机械爪 ID	1 Byte	0x08
	功能码	1 Byte	0x10
	寄存器起始地址	2 Bytes	**Address**
	寄存器数量	2 Bytes	N*
	字节数	1 Byte	**N*x2**
	寄存器	N*x2 Bytes	**Value**
	响应指令格式		
Modbus TCP 包头	事务标识	2 Bytes	0x00，0x01
	协议	2 Bytes	0x00，0x02
	长度	2 Bytes	0x00，0x09
	寄存器	1 Byte	0x7C
参数	状态	1 Byte	0x00
内部使用	主机 ID	1 Byte	0x09
Modbus RTU 数据	机械爪 ID	1 Byte	0x08
	功能码	1 Byte	0x10
	寄存器起始地址	2 Bytes	**Address**
	寄存器数量	2 Bytes	**N***

8.3 私有TCP协议示例

1. 写机械爪使能。地址(0x0100)

```
发：00 01 00 02 00 0B 7C 09 08 10 01 00 00 01 02 00 01
收：00 01 00 02 00 09 7C 00 09 08 10 01 00 00 01
```

2. 写机械爪位置为400。地址(0x0700)

```
发：00 01 00 02 00 0D 7C 09 08 10 07 00 00 02 04 00 00 01 90
收：00 01 00 02 00 09 7C 00 09 08 10 07 00 00 02
```

3. 读取报警代码。地址(0x000F)

```
发：00 01 00 02 00 08 7C 09 08 03 00 0F 00 01
收：00 01 00 02 00 08 7C 30 09 08 03 02 00 00
```

4. 写机械爪使能，位置(850)，速度(3000)，力(50)。地址(0x0Cxx)

```
发：00 01 00 02 00 11 7C 09 08 10 0C 00 00 05 0A 00 01 0B B8 00 32 03 52
收：00 01 00 02 00 09 7C 30 09 08 10 0C 00 00 05
```