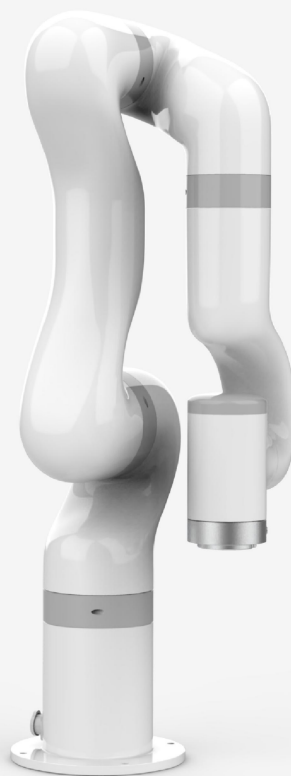




UFACTORY 850

用户手册



使用前请仔细阅读本手册

V 2.3.0

目录

前言	9
产品组成	9
手册主要内容	9
机械臂运动参数	10
单位使用说明	10
术语定义	11
更多信息	13
安全说明	13
UFactory 850 用户手册-硬件部分	18
1. UFactory 850 硬件安装	18
1.1. UFactory 850 硬件组成	18
硬件组成	18
紧急停止按钮	19
控制器说明	20
1.2. UFactory 850 安装	20

UFactory 850 安装环境安全说明	20
安装 UFactory 850	20
1.3. 机械臂系统上电	26
上电前准备	26
系统上电	26
机械臂系统关机	26
2. 电气接口	28
2.1. 交流控制器	28
交流控制器与机械臂连接	28
交流控制器外部电源连接	28
机械臂航空接头定义	29
2.2. 电气警告和注意事项	29
2.3. 末端执行器 I/O	30
工具数字输出	32
工具数字输入端	33
工具模拟输入	33
2.4. 控制器 I/O	34

所有数字 I/O 的通用规范	35
专用安全 I/O.....	37
通用数字 I/O 功能	40
通用模拟 I/O.....	41
2.5. 通信接口	43
以太网.....	43
3. 末端执行器	45
3.1. 机械爪	45
机械爪安装.....	45
机械爪控制过程	46
注意事项	46
3.2. 真空吸头.....	47
真空吸头安装.....	47
设置开关真空吸头	48
UFactory 850 用户手册-软件部分	49
1. UFactory studio 使用说明	49
1.1 硬件准备	49

1.2 连接机械臂	50
1.2.1 机械臂网络设置	50
1.2.2 PC 端 IP 设置	52
1.2.3 连接机械臂	53
1.2.4 回到搜索界面	55
1.3 UFactory studio 介绍	55
1.3.1 UFactory studio 的 3 个主要功能模块	55
1.3.2 工具栏	55
1.4 设置机械臂	56
1.4.1 运动参数	56
1.4.2 外接设备	65
1.4.3 安全设置	69
1.4.4 通用设置	70
1.4.5 我的设备	74
1.5 实时控制	79
1.5.1 页面介绍	79
1.5.2 位置控制	80

1.5.3 关节控制	85
1.5.4 轨迹录制	87
1.5.5 末端工具	88
1.6 Blockly 图形化编程	90
1.6.1 界面简介	90
1.6.2 Blockly 工作区	91
1.6.3 Blockly 编程	93
1.7 Python IDE	105
1.8 GCode 编程页面	106
2. UFactory 850 运动学特性	109
2.1 机械臂的运动方式	110
2.1.1 关节运动	110
2.1.2 直线运动及圆弧直线运动	113
2.1.3 圆及圆弧运动	116
2.3 奇异点	119
3. 典型示例	121
3.1 UFactory 850 真空吸头的使用	121

3.2 UFactory 850 机械爪的使用	122
3.3 数字 I/O 的使用	123
4. 机械臂运动模式和状态解析	124
4.1 机械臂运动模式解析	124
4.2 机械臂运动状态解析	127
附录	129
附录 1-报错及处理	129
1.1 关节报警信息和常规处理方式	129
1.2 控制器报警信息及常规处理方式	130
1.2.1 控制器错误代码	130
1.2.2 控制器警告代码	132
1.3 机械爪报警信息及常规处理方式	132
1.4 Python SDK 报警信息及常规处理方式	133
附录 2-技术规格	135
2.1 UFactory 850 规格	135
附录 3-UFactory 850 软件/固件升级的方法	137
3.1 在线升级：当 PC 端有网络连接时	137

3.2 离线升级：当 PC 端没有网络连接时	138
附录 4-维护与点检	139
附录 5-售后服务	139
附录 6-产品信息	140
1.1 产品标签	140
1.2 电磁兼容	141
1.3 使用环境	141
1.4 运输、存储和搬运	142
1.5 控制器放置高度	142
1.6 电源连接	142
1.7 特殊耗材	142
1.8 停机类别	142
1.9 停止时间和停止距离	143
1.10 最大速度	143
1.11 参数	144
附录 7-xArm 系列的运动学参数	145

前言

产品组成

一套完整的 UFactory 850 机器人出库清单如下所示：

- 机械臂 x 1
- 控制器 x 1
- 控制器电源电缆 x 1
- 机械臂供电电缆 x 1
- 机械臂通信电缆 x 1
- 网线 x1
- 机械臂末端转接线 x1
- 机械臂末端转网口线 x1
- UFactory 850 快速安装使用教程 x 1

手册主要内容

UFactory 850 用户手册-硬件部分

- (1) UFactory 850 硬件安装
- (2) UFactory 850 末端执行器

UFactory 850 用户手册-软件部分

- (1) UFactory studio 使用说明
- (2) UFactory 850 运动特性
- (3) 典型示例

附录

- (1) UFactory 850 报错及处理
- (2) UFactory 850 技术规格
- (3) UFactory 850 软件/固件升级方法
- (4) 维护与点检
- (5) 售后政策

机械臂运动参数

机械臂的运动参数见表 1.1 和表 1.2。

表 1.1 机械臂各关节工作范围

	机械臂型号	850
最大速度		180° /s
工作范围	轴 1	±360°
	轴 2	±132°
	轴 3	-242° ~3.5°
	轴 4	±360°
	轴 5	±124°
	轴 6	±360°

表 1.2 机械臂各运动参数范围

	TCP 运动	Joint 运动
速度 (speed)	0~1000mm/s	0~180° /s
加速度 (acc)	0~50000mm/s ²	0~1145° /s ²
加加速度 (jerk)	0~100000mm/s ³	0~28647° /s ³

注：（1）在 TCP 运动（即笛卡尔空间运动）指令（SDK 的 set_position() 函数）中，如果同时包含位置变化和姿态变化，一般情况下姿态旋转速度由系统自动算出。此时指定的速度参数为最大位置线速度，范围为：0~1000mm/s。

（2）当期望的 TCP 运动仅限于姿态 (roll, pitch, yaw) 变化，而位置(x, y, z)保持不变时，此时指定的速度参数为姿态旋转速度，所以范围 0~1000mm/s 对应 0~180° /s。

单位使用说明

该手册中 Python/Blockly 示例及通信协议中使用的单位见表 1.3。

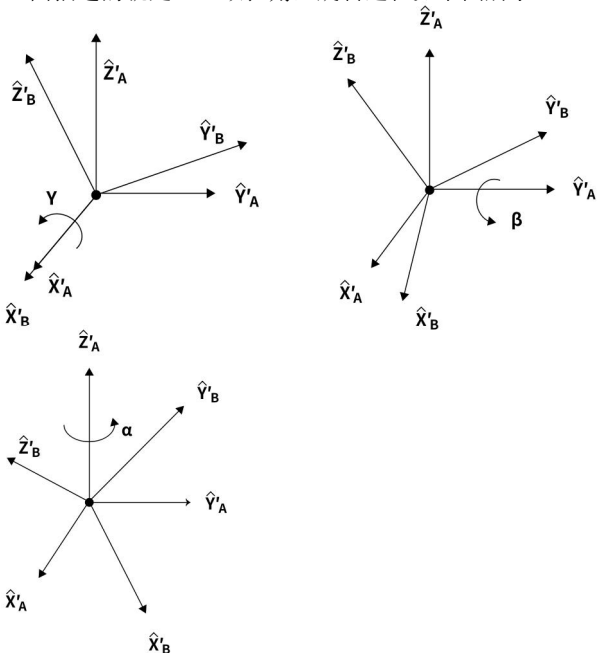
表 1.3. Python/Blockly 示例及通信协议中各参数单位

参数	Python-SDK	Blockly	通信协议
X (Y/Z)	毫米 (mm)	毫米 (mm)	毫米 (mm)
Roll (Pitch/Yaw)	度 (°)	度 (°)	弧度 (rad)
J ₁ (J ₂ /J ₃ /J ₄ /J ₅ /J ₆ /J ₇)	度 (°)	度 (°)	弧度 (rad)
TCP 速度	毫米/秒 (mm/s)	毫米/秒 (mm/s)	毫米/秒 (mm/s)
TCP 加速度	毫米/秒 ² (mm/s ²)	毫米/秒 ² (mm/s ²)	毫米/秒 ² (mm/s ²)

TCP 加加速度	毫米/秒 ³ (mm/s ³)	毫米/秒 ³ (mm/s ³)	毫米/秒 ³ (mm/s ³)
关节速度	度/秒 (° /s)	度/秒 (° /s)	弧度/秒 (rad/s)
关节加速度	度/秒 ² (° /s ²)	度/秒 ² (° /s ²)	弧度/秒 ² (rad/s ²)
关节加加速度	度/秒 ³ (° /s ³)	度/秒 ³ (° /s ³)	弧度/秒 ³ (rad/s ³)

术语定义

下列术语和定义适用于本手册。

术语	定义
控制器	为机械臂的核心部分，它是机械臂控制系统的集成。
末端执行器	末端执行器安装在机械臂手腕的前端，用来安装夹持器和专用工具（如机械爪、真空吸头等），可以直接执行工作任务。
使能机械臂	给机械臂上电，且开启机械臂电机，机械臂使能后，可正常开始运动。
TCP	工具中心点。（未设置末端执行器偏移时，为法兰盘中心）
TCP 运动	目标位置为笛卡尔空间坐标点的运动，末端在运动中遵循指定的轨迹（圆弧，直线等）。
TCP 负载 (末端负载)	负载重量是指实际的（末端执行器+托运外物）的重量，单位是 kg；X/Y/Z 轴表示 TCP 的重心相对于默认工具坐标系（位于法兰中心）的位置，单位是 mm。
TCP 偏移 (末端执行器偏移)	设置 TCP（末端执行器）坐标系与定义在法兰中心的工具坐标系之间的相对偏移量，单位是 mm。
Roll/Pitch/Yaw	<p>Roll /Pitch/Yaw 按顺序依次绕选定坐标系（基坐标系）的 X/Y/Z 旋转。 下面举例描述坐标系 {B} 姿态的一种方法： 例如首先将坐标系 {B} 和一个已知参考坐标系 {A} 重合。先将 {B} 绕 \hat{X}_A 旋转 γ，再绕 \hat{Y}_A 旋转 β，最后绕 \hat{Z}_A 旋转 α。 每个旋转都是绕着固定的参考坐标系 {A} 的轴，这种方法叫 XYZ 固定角坐标系，有时把他们定义为回转角 (roll)、俯仰角 (pitch) 和偏转角 (yaw)。 上面描述的就是 XYZ 欧拉角，旋转过程如下图所示：</p>  <p>等价旋转矩阵为： ${}^A_B R_{XYZ}(\gamma, \beta, \alpha) = R_Z(\alpha)R_Y(\beta)R_X(\gamma)$ 注： γ 对应 roll； β 对应 pitch； α 对应 yaw。</p>

轴角	<p>Rx/Ry/Rz 与 Roll/Pitch/Yaw 一样，使用 3 个值表示姿态，但不是三个旋转角度，而是一个三维旋转向量 $[x, y, z]$ 和一个旋转角度 ϕ (标量) 的乘积。</p> <p>轴角表示的性质： 假设旋转轴为 $[x, y, z]$，旋转角度为 ϕ。则轴角表示即为 $[R_x, R_y, R_z] = [x*\phi, y*\phi, z*\phi]$，其中 $[x, y, z]$ 为单位向量，ϕ 为非负值，因而 $[R_x, R_y, R_z]$ 的向量长度(模)即可推算旋转角度，向量方向即为旋转方向。 如果想表示逆向旋转，则将旋转向量 $[x, y, z]$ 取反，ϕ 值不变。 使用 ϕ 和 $[x, y, z]$ 同样可以推导出单位四元数的姿态表示 $q = [\cos(\phi/2), \sin(\phi/2)*x, \sin(\phi/2)*y, \sin(\phi/2)*z]$。</p> <p>举例： 当前 TCP 坐标系的姿态是基坐标系围绕某个空间向量旋转某个角度得到的。比如用基坐标系表示的旋转轴的向量为 $[1, 0, 0]$，旋转角度为 180 度 (π 弧度)，则这个姿态的轴角表示即为 $[\pi, 0, 0]$。如果旋转轴为 $[0.707, 0.707, 0]$，旋转角度为 90 度 ($\pi/2$ 弧度)，则轴角姿态为 $[0.707*(\pi/2), 0.707*(\pi/2), 0]$。</p>
基坐标系 (参考图 1)	<p>基坐标系是以机器人安装基座为基准、用来描述机器人本体运动的笛卡尔坐标系。</p> <p>任何机器人都离不开基坐标系，也是机器人 TCP 在三维空间运动所必须的基本坐标系 (面对机器人前后: X 轴，左右: Y 轴，上下: Z 轴)</p>
工具坐标系 (参考图 1)	<p>由工具中心点 TCP 与坐标方位组成。</p> <p>如果没有设置 TCP 偏移，那么默认工具坐标系位于法兰中心。 是以工具中心点作为零点，机器人的轨迹参照工具坐标系。</p>
用户坐标系 (参考图 1)	<p>用户坐标系可定义为机器人运动范围内的任意位置，设定任意角度的 X、Y、Z 轴，坐标系的方向根据客户需要任意定义。</p>
手动模式	<p>即示教模式或力矩模式，在该模式下，操作人员可直接手动控制机械臂。</p>
示教灵敏度	<p>示教灵敏度范围 1~5 个等级。设定的指越大，示教灵敏度等级越高，开启示教模式拖拽关节所需的力越小。</p>
碰撞灵敏度	<p>碰撞灵敏度范围 0~5 个等级，设置为 0 时表示不开启碰撞检测。设定的值越大，碰撞灵敏度等级越高，机械臂碰撞检测后所需的力越小。</p>
GPIO	<p>通用型之输入输出。</p> <p>对于输入，可以通过读取某个寄存器来确定引脚电位的高低； 对于输出，可以通过写入某个寄存器来让这个引脚输出高电位或者低电位；</p>
安全边界	<p>该模式被激活后，可以限制机械臂笛卡尔空间的边界范围，如果工具法兰中心 (TCP 偏移点) 超出设置的安全边界，机械臂将停止运动。</p>
缩减模式	<p>该模式被激活后，机械臂的笛卡尔运动的最大运动线速度、关节运动的最大关节速度和关节范围将受到限制。</p>

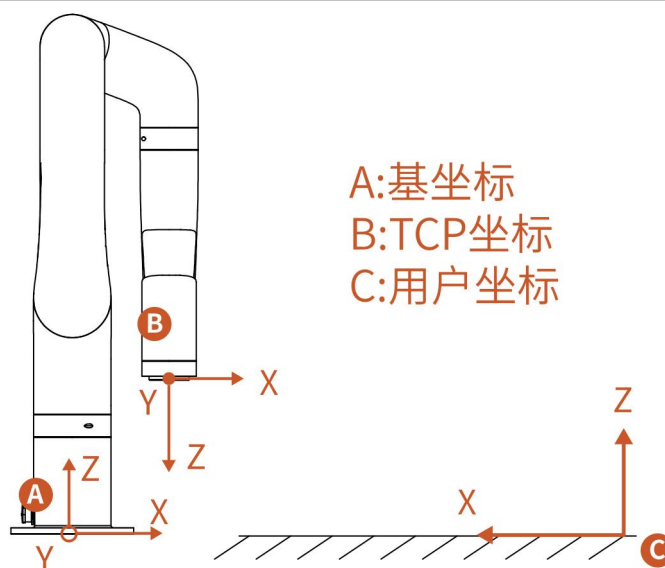


图 1

更多信息

更多的产品介绍，UFactory studio 软件下载，本手册的英文版本，请留意我们的 UFACTORY 官网 (<https://www.ufactory.cc/download/>)

安全说明

● 简介

本章节介绍了机器人应用规划应该遵守的安全原则和规范，集成商和用户必须阅读并理解，带有警示标志的内容需重点掌握并严格遵守。由于机器人系统复杂而且危险性较大，使用人员应充分了解操作的风险性，严格遵守并执行本手册中的规范和要求。操作人员需具备充分的安全意识。使用 SDK (Python/ROS/C++) 和图形化界面 UFactory studio 时，必须阅读相应的接口说明和操作手册。

UFACTORY 致力于提供可靠的安全信息，但不对此承担责任。即使一切操作都按照安全操作说明进行，也不能确保机械臂不会造成人身和财产方面的损失。

● 有效性和责任

本手册不包含设计、安装和操作一个完整的机器人应用规划，也不包含可能对应用规划的安全造成影响的周边设备。该应用规划应符合机器人安装所在国的标准和规范中确立的安全要求。集成商和用户有责任确保遵守国家相关的切实可行的法律法规，确保完整的一套机器人应用规划在安全情况下：

这包括但不限于以下内容：

- 对完整的应用规划做安全评估，与人交互时，确保路径规划与人的安全距离；
- 将风险评估定义的其他机械和附加安全设备连接在一起；
- 在使用软件编程时，认真阅读理解接口说明，建立适当的安全设置；
- 明确使用说明，确保不会因操作不当造成不必要的财产损失和人身安全；

● 责任限制

该手册所包含的所有安全方面的信息都不得视为 UFACTORY 的保证，即使遵守所有的安全指示，作业操作者所造成的伤害或损害依然有可能发生。

(1) 该手册中的警告标志

	<p>危险：可能引起的危险用电情况，如不避免，可导致人员伤亡或设备严重损害</p>
<p>危险</p>	
	<p>警告：可能引发的危险情况，如不避免，可导致人员伤亡或设备严重损害</p>
<p>警告</p>	
	<p>高温：可能引发危险的热表面，如果触碰，可造成人员伤害</p>
<p>高温</p>	
	<p>小心：如不避免，可能导致人员伤害或设备损坏</p>
<p>小心</p>	
	<p>注意：如不避免，可能导致人员伤害或设备损坏</p>
<p>注意</p>	

● 安全注意事项


(1) 概述

该章节包含安装机器人和机器人应用规划时的注意事项，预防机器及相关设备损坏的安全设

施，用户需要阅读说明书里的相关描述并完全熟知安全事项。相关情况描述无法面面俱到，具体问题需要具体分析。


(2) 一般警告和提醒

	<p>危险</p>	<ol style="list-style-type: none"> 1. 务必按照本手册指示和要求，正确安装机器人和所有电气设备。 2. 需要有专业的人员按照标准对机器人进行安装和测试。 3. 确保机械臂的活动空间不会碰撞到其他设备或者周边人群。 4. 首次启动系统和设备前，必须检查设备和系统是否完整，操作是否安全，检查机器人和其他设备系统是否遭到损坏。 5. 在使用机器人及投入生产前需要对机器人及周边防护系统进行初步测试和检查。 6. 在使用 SDK (Python/ROS/C++) 和图形化界面 UFactory studio 时，操作人员必须要经过相应培训，必须确保输入的参数和操作流程是正确的。 7. 机器人重新安装和调试后，需要再次进行全面的安全评估并保留文件记录。 8. 机器人在运行期间发生意外或运行不正常的情况下，可以按下紧急停止按钮，机械臂姿态会轻微刹车下坠。 9. 在机器人作业的时候，切勿有人或者其他设备出现在作业范围内。 10. 在释放机器人刹车时，请注意做好机械臂保护措施，避免高处掉落造成人员伤亡或设备损坏。 11. 将不同的机械连接在一起可能会加重危险或者引发新危险。应始终对安装系统做全面的安全评估。
	<p>警告</p>	<ol style="list-style-type: none"> 1. 机器人和控制器在运行的过程中会产生热量，机器人在工作时或刚停止工作时，请不要触摸机械臂和控制器。 2. 切勿将手指伸到末端执行器连接处。
	<p>注意</p>	<ol style="list-style-type: none"> 1. 确保正确安装机器人，检查所有的电路情况 2. 确保安装的环境，机械臂有足够的活动空间。 3. 确保机械臂工作空间没有障碍物。 4. 控制器必须放置在机械臂运动范围之外，以防出现紧急情况不能及时按下急停按钮 5. 运行过程中，如果紧急停止，务必确保在该姿态下的机器人，重新启动或复位零点时不会碰撞到障碍物。

	<ol style="list-style-type: none"> 切勿改动机器人，对机器人的改动可能造成的集成商无法预料的危险，机器人授权重组需依照最新版的所有相关服务手册。如果机器人以任何方式被改变或改动，我司拒绝承担一切责任。 在机器人运输或搬运过程中，做好防撞防水措施。
<p>小心</p>	<p>当机器人与其他机械协作时，应对整个协助系统做全面的安全评估。建议在应用规划时，对可能造成机械损坏的设备放置在机械臂工作范围以外的地方。</p>

(3) 人员安全

在操作或者运行机器人时，首先必须要确保作业人员的安全，下面列出一般性注意事项，请妥善采取确保作业人员安全的相应措施。

	<ol style="list-style-type: none"> 使用机器人系统的作业人员，应仔细阅读产品用户手册。需确保其充分掌握安全、规范的操作流程，以及机器人运行报错的处理方式。 在设备运转之中，即使机器人看上去已经停止，也有可能是因为机器人在等待启动信号而处在即将动作的状态。即使在这样的状态下，也应该将机器人视为正在动作中。 标清机器人的工作空间，使操作者了解机械臂及包含末端执行器（机械爪、真空吸头等）的动作范围。 定期检查机器人，防止固定机械臂或工具的螺栓松动而导致不良后果。 在机器人运行速度过快时，务必小心谨慎。 机器人意外断电，或因夹持不稳，导致物品掉落。
<p>注意</p>	

● 机器人操作的注意事项

操作机器人时必须遵循以下安全程序和警告事项：

<ol style="list-style-type: none"> 不要改变控制器安全配置中的任何信息。如果配置文件参数变更，整个机器人系统应被视为新系统，这就意味着所有安全审核过程，比如风险评估，都必须更新。 使用部件号相同的新部件或 UFACTORY 批准的相当部件替换故障部



警告

件。

3. 书面记录所有维修操作，并将其保存在整个机器人系统相关的技术文档中。



危险

1. 从控制器移除主输入电缆以确保其完全断电，采取必要的预防措施以避免其他人在维修期间重新接通系统电源。
2. 重新开启系统前请检查接地连接。
3. 拆分机械臂或控制器时请遵守 ESD（静电释放）法规。
4. 避免拆分控制器内的供电系统。控制器关闭后其供电系统仍可留存高压达数小时。
5. 避免水或粉尘进入机械臂或控制器。

UFactory 850 用户手册-硬件部分

1. UFactory 850 硬件安装

1.1. UFactory 850 硬件组成

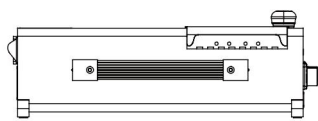
硬件组成

机械臂硬件组成包含如下图：

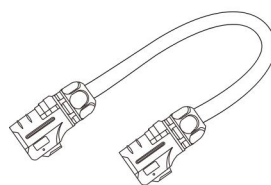
- 机械臂 (图-1)
- 控制器 (图-2)
- 机械臂信号与供电电缆 (图-3)
- 控制器电源电缆 (图-4)
- 末端转接线 (图-5)
- 网线 (图-6)
- 末端网线转接线 (图-7)



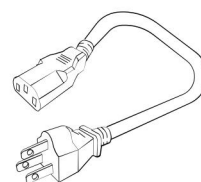
1



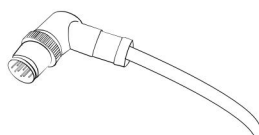
2



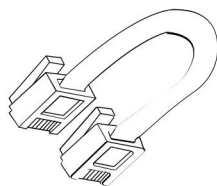
3



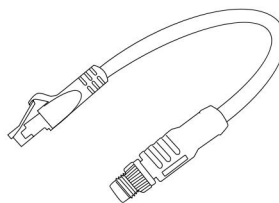
4



5



6

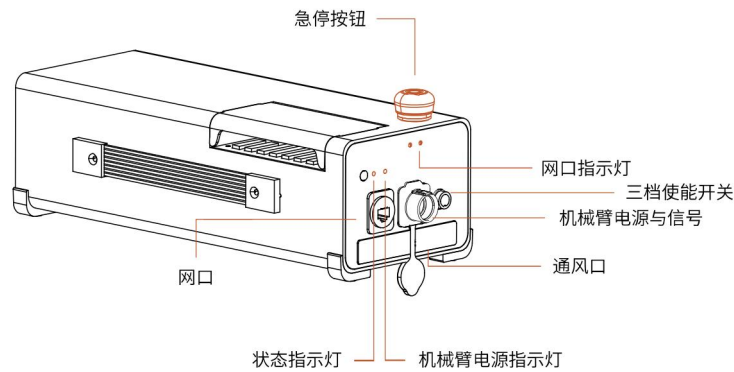


7

机械臂由底座和旋转关节组成，每个关节表示一个自由度，从底座往上依次表示关节 1、关节 2、关节 3...。最后一个关节也叫工具端，用于连接末端执行器（机械爪，真空吸头等）。
 关节图片介绍见技术规格。

紧急停止按钮

按下控制器的急停按钮，会发送指令到控制器进行软件减速停止，停止机械臂的一切活动，且清空控制器里面所有的缓存指令，随后 300ms 左右断开机械臂供电电源。紧急停止不可作为风险降低措施，机械臂运行过程中出现紧急情况时，可按下紧急停止按钮，机械臂姿态会轻微刹车下坠。紧急停止按钮如下图所示：



急停：按下急停按钮，机械臂电源指示灯熄灭，机械臂不在供电状态。

上电：按箭头指示方向旋转后按钮呈弹起态，机械臂电源指示灯亮起，机械臂为供电状态。

注意：

按下急停按钮后，再次使用机械臂时要进行以下操作：

1. 重新给机械臂上电（按箭头指示方向旋转急停按钮）
2. 使能机械臂（使能伺服电机）

使能机械臂：UFactory studio：点击首页按钮 **【使能机械臂】**



Python-SDK：motion_enable(true)

控制器说明

控制器按钮、指示灯	参数名称	功能
机械臂电源指示灯	ROBOT PWR	灯亮，表示机械臂已上电
状态指示灯	STATE	灯闪烁，表示控制器已开机
网口指示灯	LAN	灯亮，表示机械臂通信正常
急停按钮	EMERGENCY STOP	按下按钮，机械臂断电 旋转按钮，机械臂上电，机械臂电源指示灯亮起
通风口	通风口	为控制器通风，防止过热，用户可以拆卸清理通风口处的灰尘，避免堵塞。

1.2. UFactory 850 安装

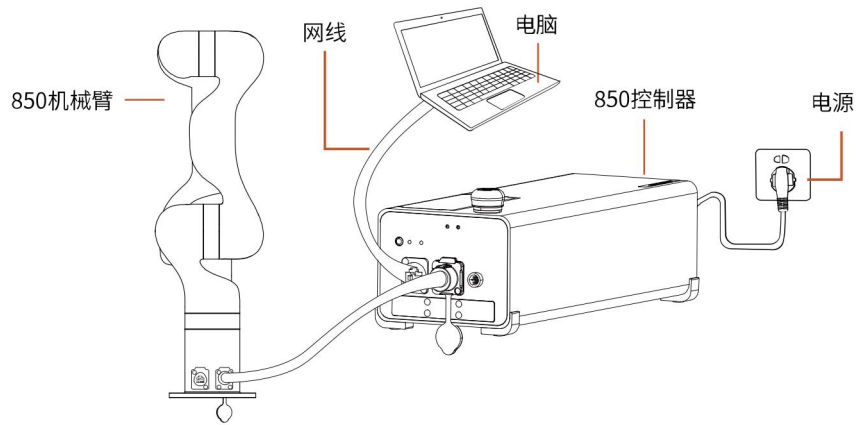
UFactory 850 安装环境安全说明

 危险	<ol style="list-style-type: none"> 1. 确保机器人正确并安全地安装到位。安装表面必须是防震坚固的。 2. 安装机器人必须检查螺栓是否拧紧。 3. 将机械臂安装在一个坚固的表面，该表面应当足以承受至少 10 倍的机座关节的完全扭转力，以及至少 5 倍的机械臂的重量。
 警告	<ol style="list-style-type: none"> 1. 机器人切勿直接接触液体，不应长期放置在潮湿环境。 2. 每次安装完都需要进行安全评估。在连接、断开机器人电缆时，确保已断开外部交流电的连接。切勿在连接了外部交流电的情况下去连接或者断开机器人电缆，以免触电发生危险。

安装 UFactory 850

机器人安装简要步骤：

- (1) 确定机械臂工作空间
- (2) 固定机械臂底座，安装机械臂
- (3) 机械臂与控制器连接
- (4) 控制器连接电缆、网线
- (5) 安装末端执行器

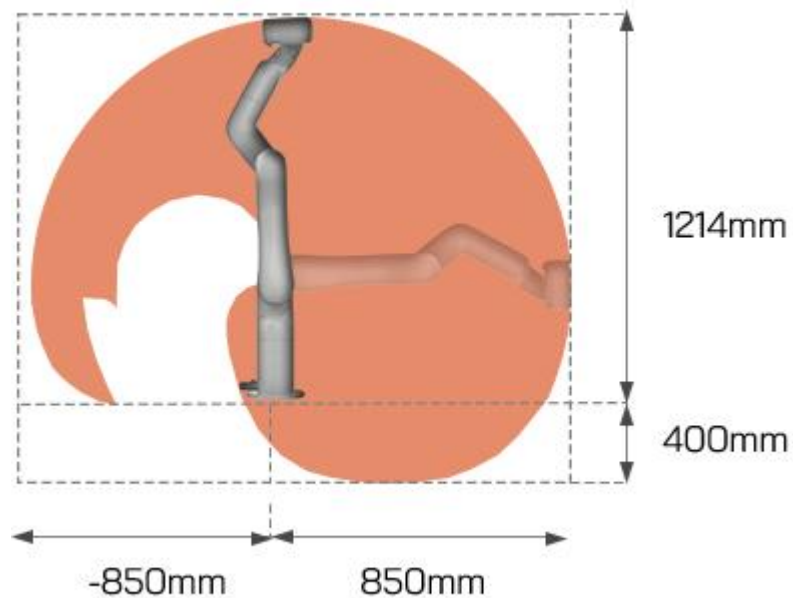


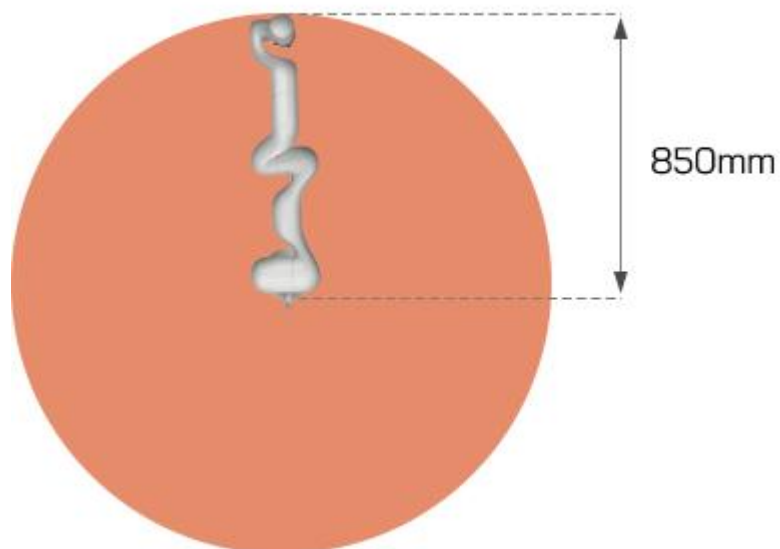
1.2.1.1. 确定机械臂工作空间

机械臂工作空间是指在关节延伸范围内的区域。下图为机械臂的尺寸图和工作范围图。在安装时，务必考虑机械臂的运动范围，以免磕碰到周围人员和设备。（以下工作范围不包括末端执行器）

UFactory 850 机械臂工作空间(单位：mm)

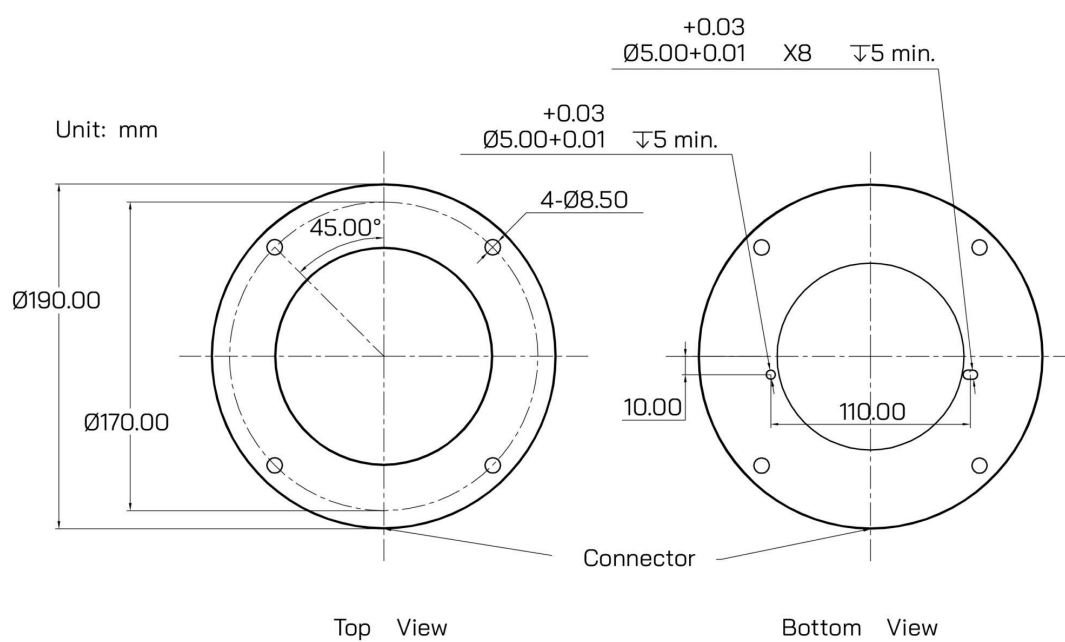
注意：以下工作范围图仅用于做安全评估。





1.2.1.2. 安装机械臂

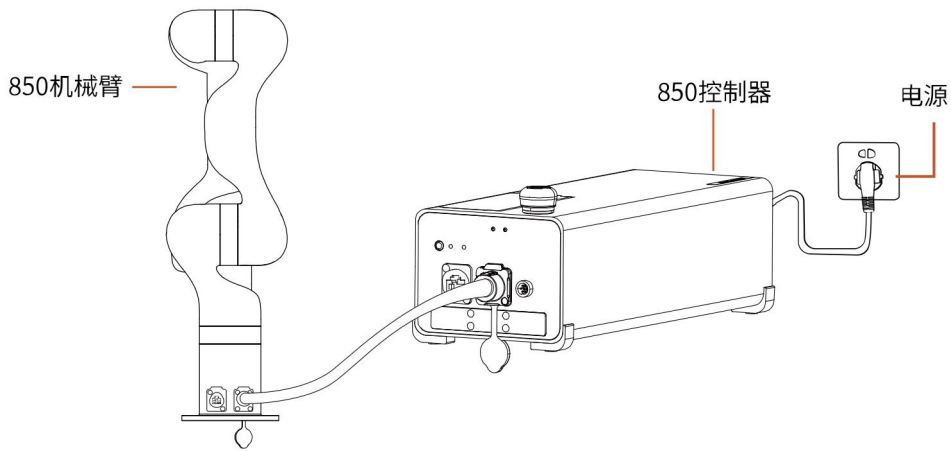
机械臂使用 4 颗 M8 螺栓，通过机械臂基座上的 4 个 $\Phi 8.5$ 孔用安装工具夹来安装机械臂。建议以 20Nm 扭矩紧固这些螺栓。



机械臂底座安装图，单位 mm

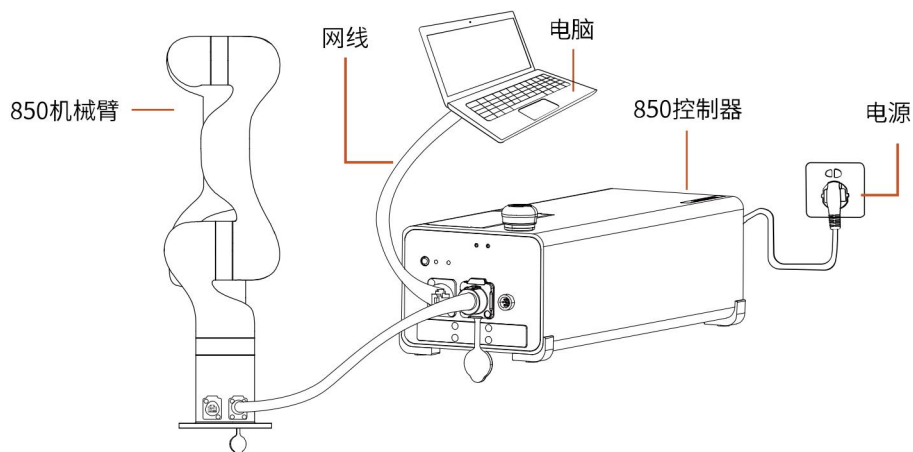
1.2.1.3. 机械臂与控制器连接

- 将机械臂供电电缆和机械臂通信电缆接头插入机械臂接口，接头具备防呆功能（每个接头都具有固定卡槽，请对准后再进行接线），已做限位处理，请勿暴力拆装；
- 将机械臂供电电缆和机械臂通信电缆另一端接头插入控制器一侧的接口；
- 将控制器电源电缆接头插入 AC（100V-240V）接口，另一端插入插座插口；



1.2.1.4. 控制器连网

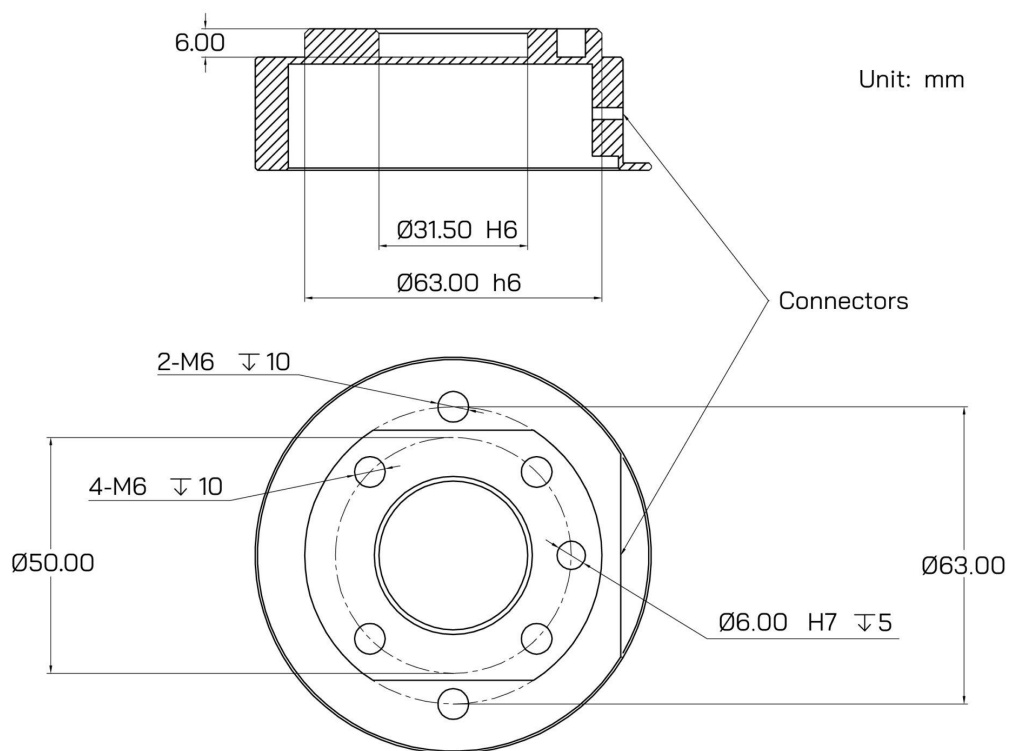
将网线接头插入控制器标为 LAN 的接口，网线另一端接头插入 PC 端（电脑）的网口或者局域网网口。



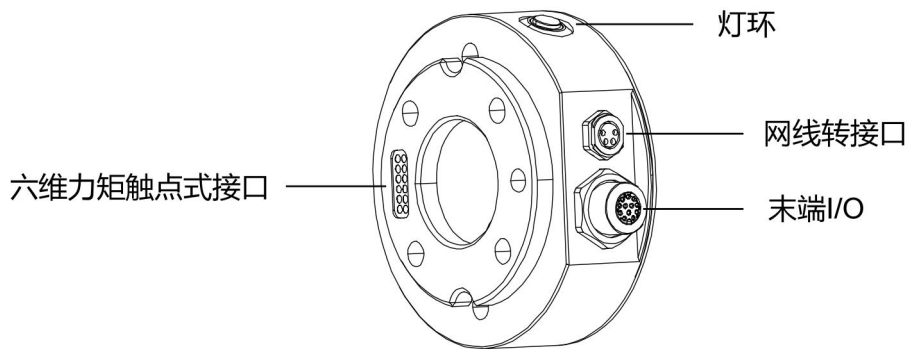
1.2.1.5. 安装末端执行器

UFactory 850 机械臂末端工具法兰有 6 个 M6 螺纹孔和一个 $\Phi 6$ 的定位孔，设计符合 ISO 9409-1-50-4-M6 标准。

若您要安装的末端执行器没有定位孔，安装末端执行器的方向务必以文件形式存档，避免因为换人重新安装末端执行器时方向出现错误，导致运行结果完全出乎意料。



工具法兰机械尺寸图，单位为 mm



工具法兰接口图

灯环

实际上是这个按钮，按下按钮可以显示不同颜色的环形灯，目前暂未定义，后期我们会开发相关的功能，集成到此按钮，用灯环颜色代表某种含义。

网线转接口

手臂内部由一根 100M 标准 CAT5 的以太网电缆连接到机械臂底座的网口，与大多数第 3 方视觉设备兼容。如果你不想把它当做是网线，也可以尝试将其作为物理信号线。包装附赠送一根转接线。

末端 I/O

有一个航空插座 12 芯的母工业连接器，为特定机械臂工具上使用的夹持器和传感器提供电源和控制信号。用于连接末端治具，可以连接 xArm 机械爪，xArm 真空吸头，BIO 机械爪，以及 Robotiq-2F-85 机械爪, Robotiq-2F-140 机械爪。和一些支持标准 modbus rtu 协议的机械爪。

六维力矩触点式接口

用于连接六轴力矩，如果没有购买此配件，则目前这个接口暂时没有其它的功能，此接口后续会开发和迭代，以支持更多的配件。



危险

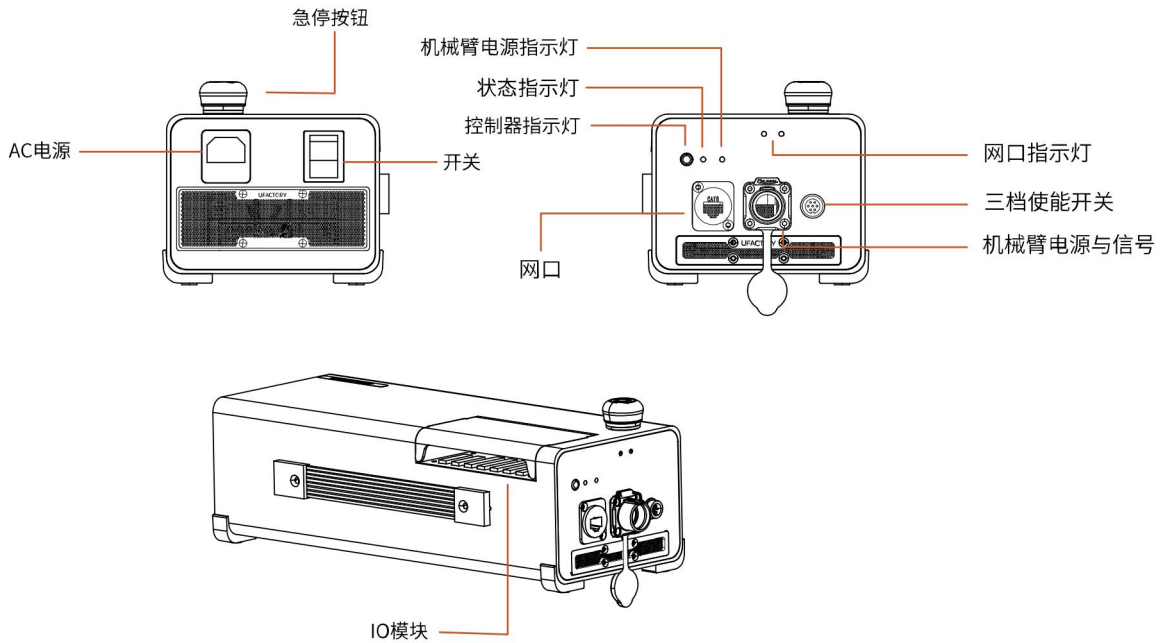
1. 确保末端执行器正确并安全地安装到位。
2. 如果末端执行器没有定位孔时，安装末端执行器的方向务必以文件形式存档。
3. 确保末端执行器安全架构，不会有零件意外坠落造成危险。
4. 若末端执行器尖锐，务必遵守操作规范。
5. 如果安装的末端执行器在机械臂零点姿态时已超出安装表面，此时回零点操作必须进行安全评估。

1.3. 机械臂系统上电

上电前准备

- 检查控制器与机械臂的电源电缆，通信线是否连接完好。
- 检查网络电缆是否连接完好。
- 检查控制器电源电缆是否连接完好。
- 确保机械臂在工作范围内不会碰到周围人员或设备。

系统上电



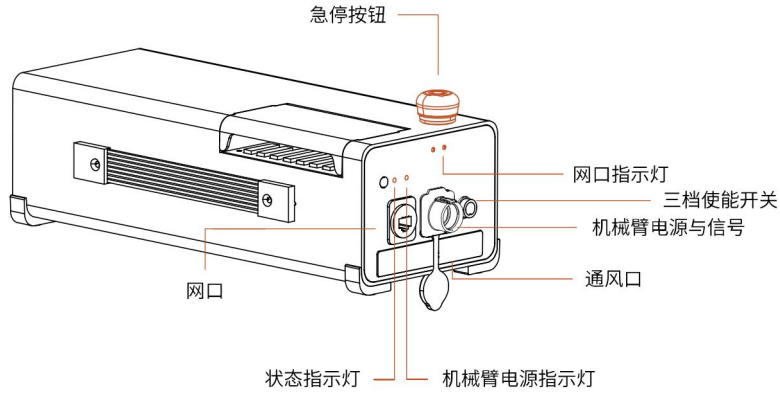
1. 打开控制器开关键，开关键 ON/OFF 灯亮起。
2. 检查状态指示灯是否亮起，若亮起表示控制器已开机。否则需手动按控制器电源按钮开机。
3. 急停按钮按照箭头指示方向旋转且向上拔起，此时机械臂电源指示灯亮起，机械臂上电。
4. 通过 UFactory Studio/SDK 命令完成使能机械臂的操作。（使能伺服电机）

机械臂系统关机

机械臂关机顺序

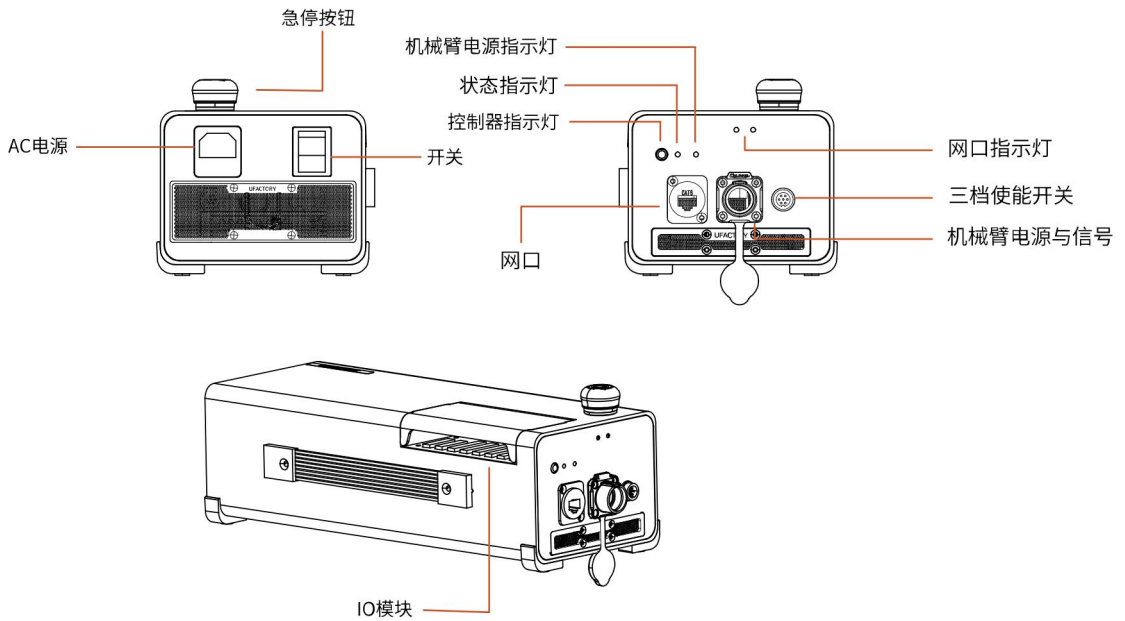
1. 断开机械臂供电

(1) 按下急停按钮，停止给机械臂通电，机械臂电源指示灯熄灭。



2. 控制器关机:

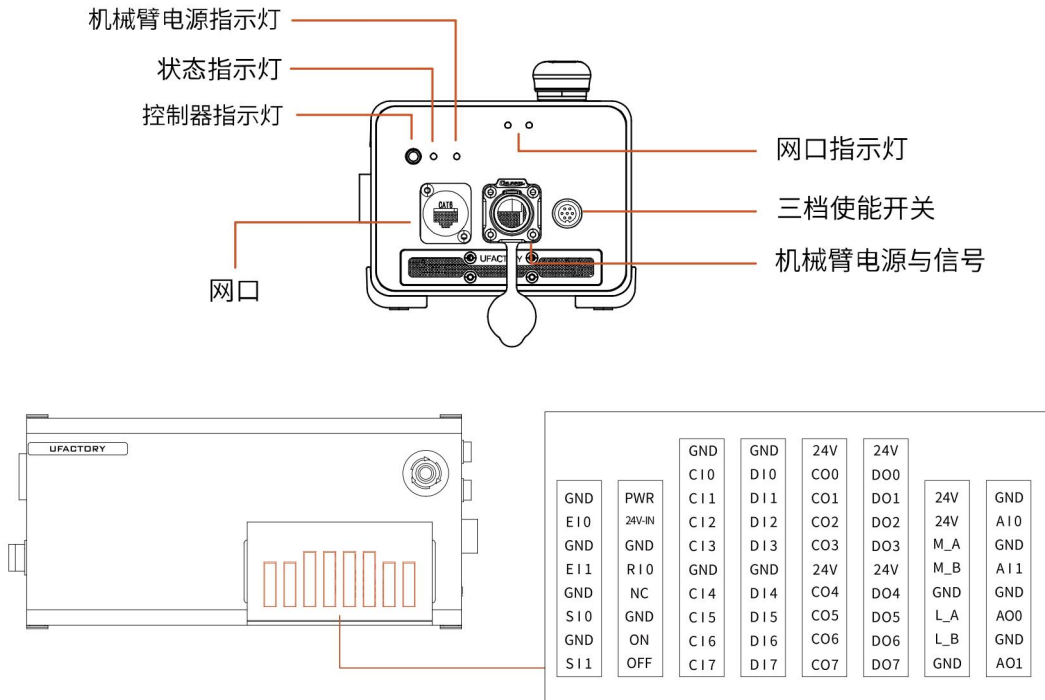
(1) 关闭控制器电源开关（关闭控制器电源开关需要 5s 左右的时间来释放电源。关闭电源后请不要立马开机。）



2. 电气接口

2.1. 交流控制器

交流控制器与机械臂连接



1. 机械臂供电电缆连接机械臂的电源接口和控制器的机械臂电源接口。
2. 机械臂信号电缆连接机械臂的信号接口和控制器的机械臂信号接口。

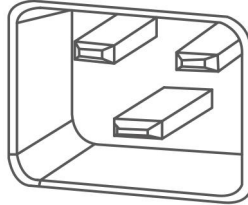
交流控制器外部电源连接

控制器市电电缆末端有一个标准 IEC 插头。将当地专用的市电插座或电缆连接到 IEC 插头。

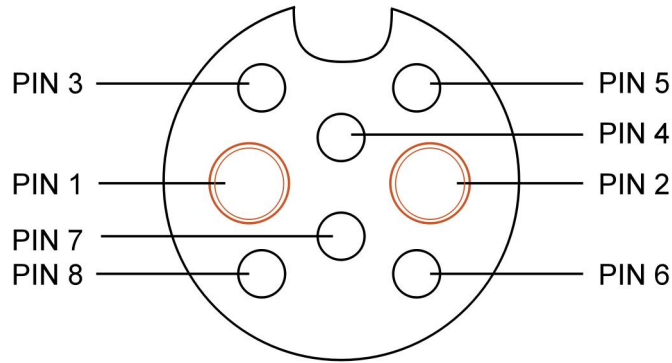
控制器由 100V-240V 交流电供电，输入频率为 47-63HZ，其内部的开关电源把 100V-240V 交流电转化为 12V、48V 直流电，为控制器内负载和机械臂供电。所以使用前必须检查机械臂与控制器的连线是否牢靠。控制器中有硬件防护和软件防护，最大程度保证在使用时的安全。

控制器的急停按钮，用户可在最短时间内切断机械臂电源，保护人员和设备的安全。

要使机械臂通电，控制器必须与电源相连。这个过程必须使用相应的 IEC C19 电线连接控制器的标准 IEC C20 插头来完成，请参见下图。



机械臂航空接头定义



6 芯航空插头(机械臂信号)	
航空接头线序	功能定义
1	48V
2	GND
3	485-A 绿 手臂
4	屏蔽
5	485-B 绿白 手臂
6	485-B 蓝白 末端工具
7	屏蔽
8	485-A 蓝 末端工具

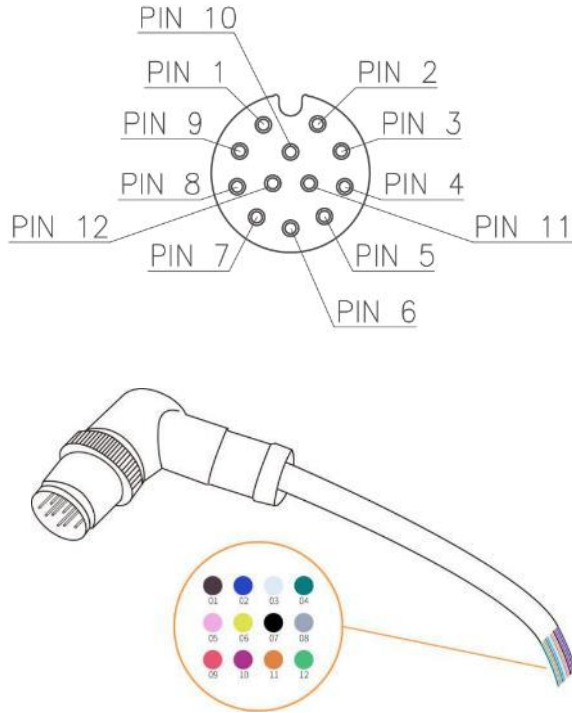
2.2. 电气警告和注意事项

在设计和安装机器人应用时，务必遵循以下警告和注意事项。实施维护作业同样要遵循这些警告和注意事项。

 <p>危险</p>	<ol style="list-style-type: none"> 切勿将安全信号连接到安全等级不合适的非安全型 PLC。如不遵守该警告，可能会因某项安全停止功能失效而导致人员严重受伤甚至死亡。
 <p>小心</p>	<ol style="list-style-type: none"> 请确保所有不得沾水的设备都保持干燥。如果有水进入产品，请切断电源，然后联系您的供应商。 仅使用该机械臂的原装电缆。请不要在那些电缆需要弯折的应用中使用机械臂。如果需要更长的电缆或柔性电缆，可以联系您的供应商。 本文提到的所有 GND 接头只适用于供电和传送信号。 当向机械臂的 I/O 安装接口电缆时，务必小心。
 <p>注意</p>	<ol style="list-style-type: none"> 高于 IEC 标准中规定电平的干扰信号将会造成机械臂的异常行为。信号电平极高或过度暴露将会对机械臂造成永久性的损害。由 EMC 问题造成的任何损失，深圳市众为创造科技有限公司概不负责。 用于连接控制器与其他机械和工厂设备的 I/O 电缆长度不得超过 30 米，除非进行延长测试后表明可行。
 <p>警告</p>	<ol style="list-style-type: none"> 控制器电气接口接线时，控制器必须断电。

2.3. 末端执行器 I/O

在机械臂的工具端，有一个航空插座 12 芯的母工业连接器，为特定机械臂工具上使用的夹持器和传感器提供电源和控制信号。请参见下图。




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（T00）
8	灰	工具输出 1（T01）
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）

电气规范如下所示：

参数	最小值	典型值	最大值	单位
24V 模式下的电源电压	20	24	30	V
电源电流 *	-	-	1800	mA

注意：* 强烈推荐为电感性负载使用保护二极管



危险


连接工具和夹持器保证中断电源时不会导致任何危险，例如工件从工具上掉落。

工具数字输出

数字输出以 NPN 的形式实现，集电极开路。数字输出激活后，相应的接头即会被驱动接通 GND，数字输出端禁用后，相应的接头将处于开路（开集/开漏）。

电气规范如下所示：

参数	最小值	典型值	最大值	单位
开路时的电压	-0.5	-	30	V
灌入 50mA 电流时的电压	-	0.05	0.20	V
灌电流	0	-	100	mA
通过 GND 的电流	0	-	100	mA

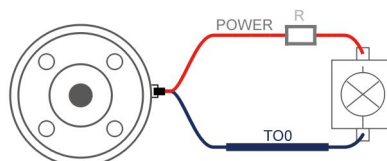


小心

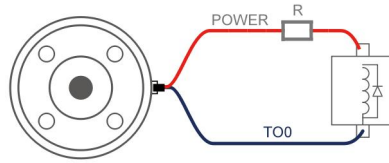
工具中的数字输出端没有电流限制，若超过所规定的数值，可能会导致永久性损坏。

2.3.1.1. 使用工具数字输出

下例说明了如何使用数字输出，因为内部输出为集电极开路，所以需要根据负载上接电阻到电源。电阻的大小及功率视具体情况。



注意：强烈推荐为电感性负载使用保护二极管，如下所示。



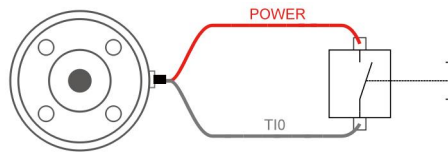
工具数字输入端

数字输入已配有下拉电阻器。这意味着浮置输入的读数始终为低。电气规范如下所示。

参数	最小值	典型值	最大值	单位
输入电压	-0.5	-	30	V
逻辑低电压	-	-	1.0	V
逻辑高电压	1.6	-	-	V
输入电阻	-	47K	-	Ω

2.4.2.1 使用工具数字输入

下例显示了简单按钮的连接方法。



工具模拟输入

工具模拟输入为非差分输入。电气规范如下所示。

参数	最小值	典型值	最大值	单位
电压模式下的输入电压	-0.5	-	3.3	V
分辨率	-	12	-	位
电流模式下的输入电流	-	-	-	mA
4mA 至 20mA 电流范围内的下拉电阻	-	-	165	Ω
分辨率	-	12	-	位

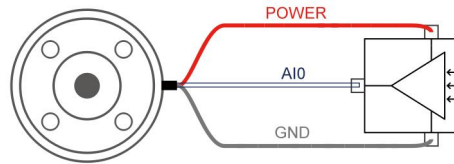


小心

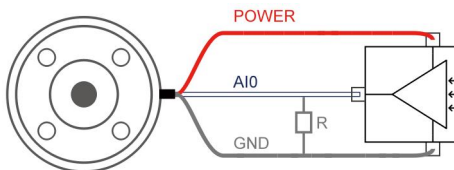
1. 在电流电压模式下，模拟输入没有提供过电压保护。超过电气规范中的极限可能导致输入端永久损坏。
2. 电流模式下，下拉电阻大小取决于输入电流的范围。

2.4.3.1 使用工具模拟输入，非差分

下例显示了模拟传感器与非差分输出的连接方式。



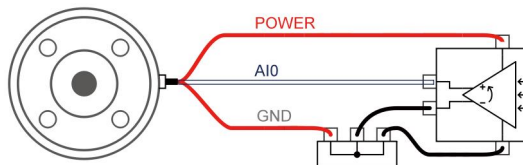
电压模式



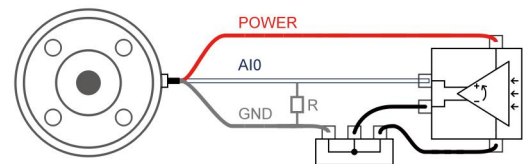
电流模式

2.4.3.2 使用工具模拟输入，差分

下例显示了模拟传感器与差分输出的连接方式。将负输出端连接至 GND（0V），即可像非差分传感器一样工作。



电压模式



电流模式

2.4. 控制器 I/O

本章说明了如何连接设备与控制器外部的 I/O。此 I/O 极其灵活，可用于多种不同的设备，其中包括气动继电器、PLC 和紧急停止按钮。

下图显示了控制器内部的电气接口布局。



所有数字 I/O 的通用规范

本节阐述了控制器的下列 24V 数字 I/O 的电气规范。

- 专用安全 I/O。
- 可配置通用 I/O

关于 IO 的具体配置功能见下表：

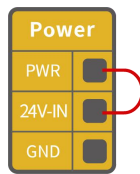
可配置功能	CI0-CI7 (可配置输入)	DI 0-DI 7 (数字输入)
通用输入	有	有
停止运动	有	无
防护重置	有	无
离线任务	有	有
手动模式	有	有
缩减模式	有	无
使能机械臂	有	有

可配置功能	CO 0-CO 7 (可配置输出)	DO 0-DO 7 (数字输出)
通用输出	有	有
运动停止	有	有
运动中	有	有
有错误	有	有
有警告	有	有
发生碰撞	有	有
手动模式生效	有	有
缩减模式生效	有	有

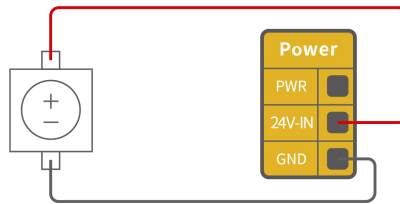
离线任务运行中	有	有
机械臂已使能	有	有
紧急停止按钮被按下	有	有

按照电气规范安装 UFactory 850 机械臂是非常重要的，这二类不同的输入都要做到这一点。

数字 I/O 可由内部 24V 电源供电，也可通过配置电源接线盒由外部电源供电。上面（PWR）为内部的 24V 电源输出。下面的终端（24V-IN）为供应 I/O 的 24V 输入外部电源输入。默认配置为使用内部电源，参见下文。



如果需要更大的电流，可如下图所示连接外部电源。




内部和外部电源的电气规范如下所示。

终端	参数	最小值	典型值	最大值	单位
内置 24V 电源					
[PWR - GND]	电压	23	24	30	V
[PWR - GND]	电流	0	-	1.8	A
外部 24V 输入要求					
[24V - 0V]	电压	20	24	30	V
[24V - 0V]	电流	0	-	3	A

数字 I/O 电气规范如下所示。

终端	参数	最小值	典型值	最大值	单位
数字输出					
[COx]	电流*	0	-	100	mA
[COx]	电压降	0	-	0.5	V
[COx]	漏电流	0	-	0.1	mA
[COx]	功能	-	NPN (OC)	-	类型
数字输入					
[EIx/SIx/CIx/RIx]	电压	0	-	30	V

[EIx/SIx/CIx/RIx]	OFF 区域	15	-	30	V
[EIx/SIx/CIx/RIx]	ON 区域(低电平)	0	-	5	V
[EIx/SIx/CIx/RIx]	电流 (0-0.5)	3	-	8	mA
[EIx/SIx/CIx/RIx]	功能	-	-	-	类型
注意: ** 对于最大 1H 的电阻负载或电感性负载。					



1. 控制器的数字输出端没有电流限制，若超过所规定的数值，可能会导致永久性损坏。

小心

专用安全 I/O

本节介绍了专用安全输入和配置为安全 I/O 时的可配置，务必遵循第 8 章的通用规范。

安全装置和设备必须按照安全说明和风险评估安装（参见第 1 章）。

所有安全 I/O 成对存在（冗余），必须保留成两个独立的分支。单一故障应不会导致丧失安全功能。固定的安全输入有两个：机械臂紧急停止和防护停止。机械臂紧急停止输入仅用于紧急停止设备。防护停止输入用于所有类型的安全型保护设备。功能差异如下所示。

	紧急停止	防护停止
机械臂停止运动	是	是
程序执行	停止	暂停
重置	手动	自动或手动
使用频率	不常使用	没有限制
需要重新初始化	仅释放制动器	否

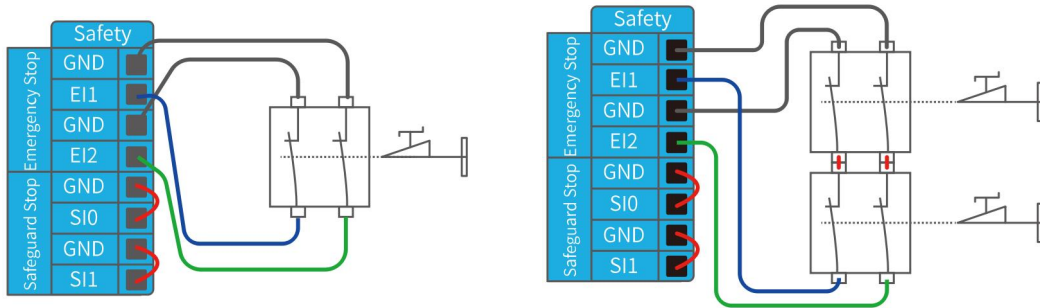
默认安全配置

所交付的机械臂进行了默认配置，可在没有任何附加安全设备的情况下进行操作，请参阅下图，若手臂出现问题，请第一时间检查下图连线是否正确。



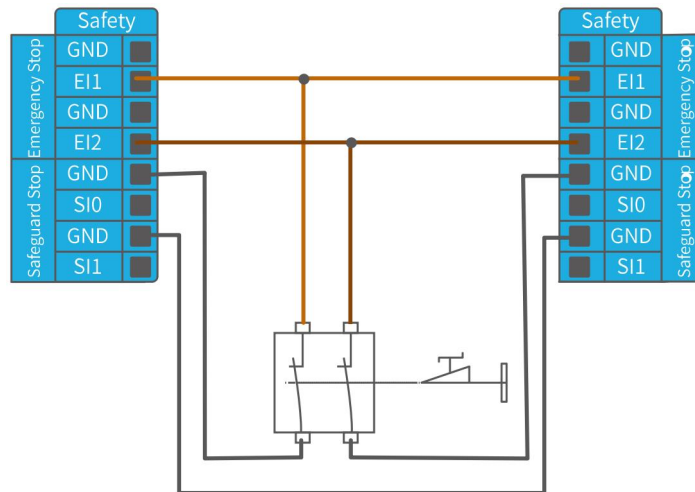
2.5.2.1 连接紧急停止按钮

在大多数应用中，需要使用一个或多个额外的紧急停止按钮。下图显示了如何连接一个或多个紧急停止按钮。



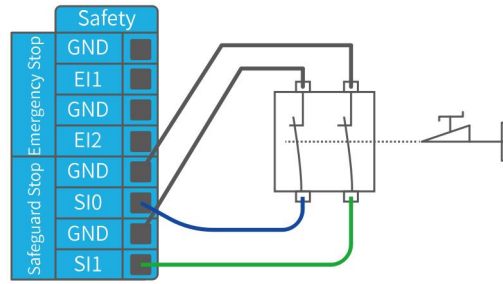
2.5.2.2 与其他机器共享紧急停止

当机械臂与其他机器搭配使用时，往往需要设置一条公共的紧急停止电路。下图显示了两台机械臂共享一个急停按钮（下图所示的连接方法同样适用于多台机械臂共享一个急停按钮）。

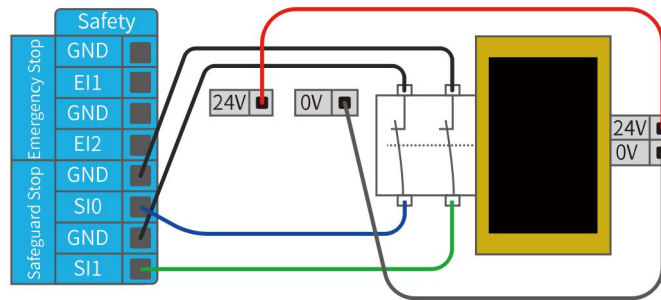


2.5.2.3 可自动恢复的防护停止

门开关就是基本防护停止设备的一个例子，门打开时，机器人停止，请参见下图。

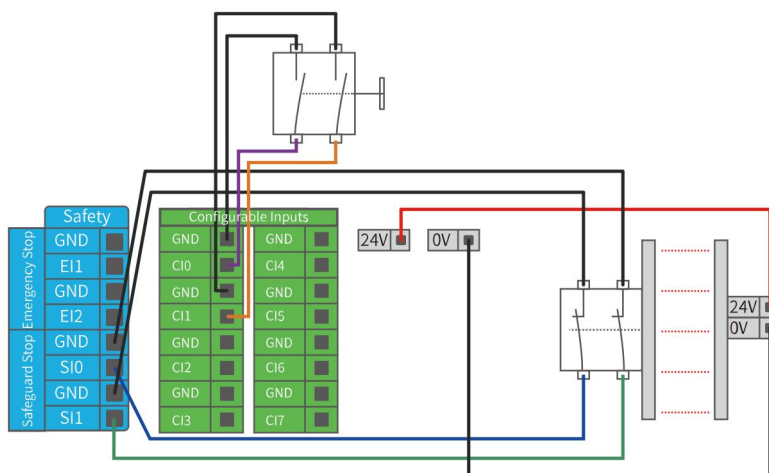


此配置仅针对操作员不能通过门并在身后关上门的应用。可配置的 I/O 可以用于设置门外的重置按钮，以重新激活机器人运动。适合进行自动恢复的另外一个例子是使用安全垫或安全型激光扫描仪，参见下图。



2.5.2.4 带重置按钮的防护停止

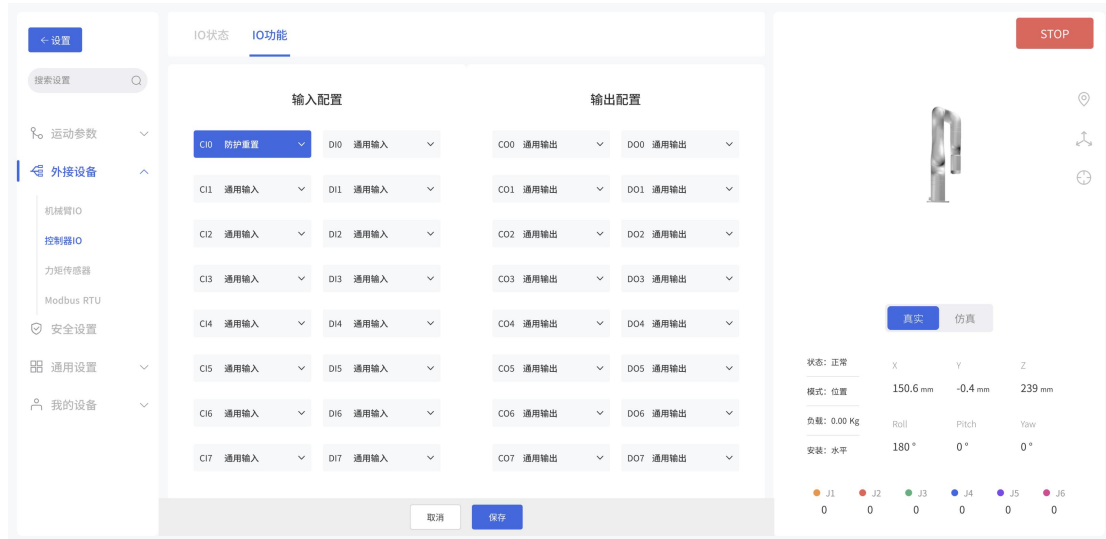
如果使用防护接口与光幕交互，需要从安全地带外部进行重置。重置按钮必须为双通道型按钮。在本例中，将 I/O 输入口 “CI0-CI1” 配置为防护停止（在 UFactory studio 中也要做相应的配置），参见下图。



如何实现带重置按钮的防护停止功能：

1. 在 UFactory studio 里将 “CI0” 配置为防护重置，具体步骤如下图：

进入 “设置” - “外接设备” - “控制器 I/O” - 将 CI0 配置为防护重置 - 点击 “保存”。



2. 若需要机械臂恢复运动，则将 S10 和 S11 接地，并通过将 C10 接地来触发机械臂运动；若需要机械臂暂停运动，则将 S10 和 S11 悬空。

注意：

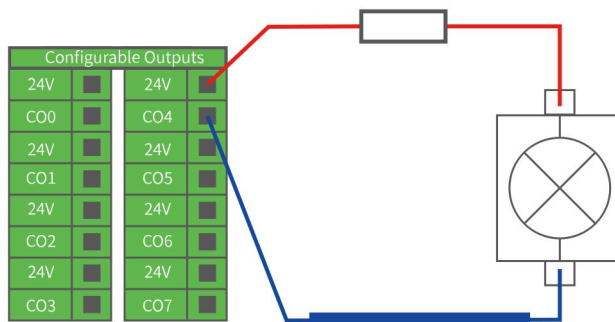
DI0-DI7 不能配置以下功能：停止运动、缩减模式、防护重置。

通用数字 I/O 功能

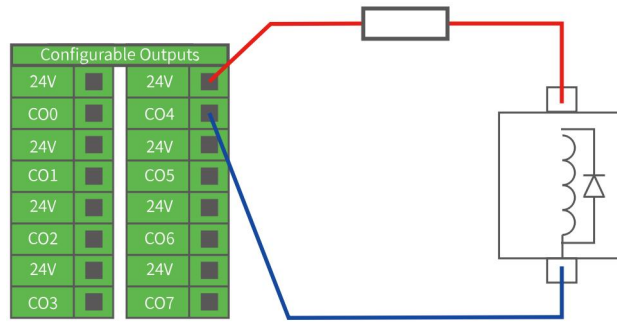
2.5.3.1 可配置数字输出

数字输出以 NPN 的形式实现。数字输出激活后，相应的接头即会被驱动接通 GND，数字输出端禁用后，相应的接头将处于开路（开集/开漏）。

下例说明了如何使用数字输出，因为内部输出为开漏输出，所以需要根据负载上接电阻到电源。电阻的大小及功率视具体情况。



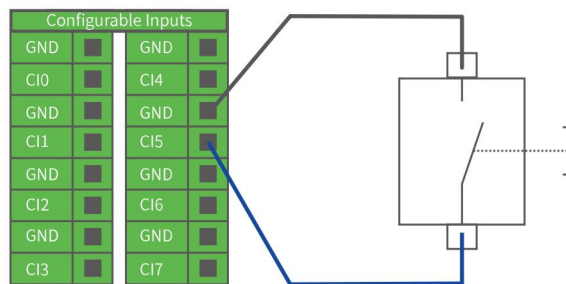
注意：强烈推荐为感性负载使用保护二极管，如下所示。



2.5.3.2 可配置数字输入

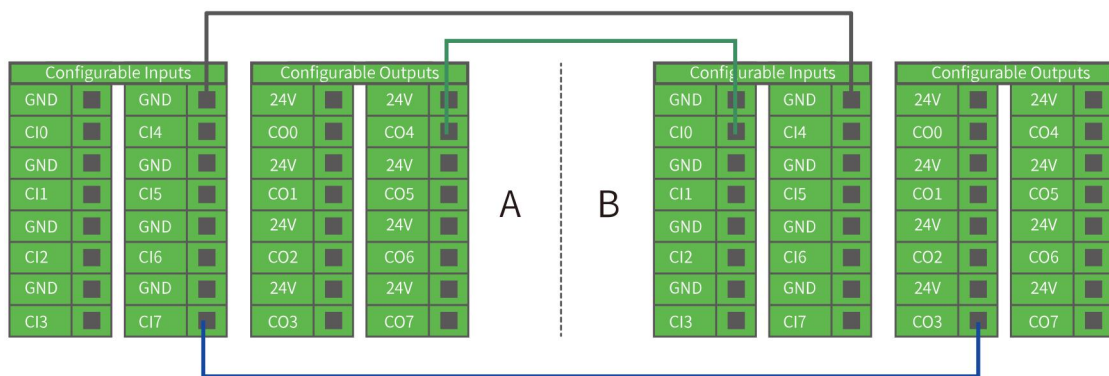
数字输入以配有弱上拉电阻器的形式实现。这意味着浮置输入的读数始终为高。

本例显示了简单按钮与数字输入的连接方式。



2.5.3.3 与其他机器或 PLC 通信

如果建立了通用 GND (0V) 并且机器采用开漏输出技术，则可使用数字 I/O 与其他设备通信，参见下图。



通用模拟 I/O

此类接口可用于设置或测量进出其他设备的电压 (0-10V)。

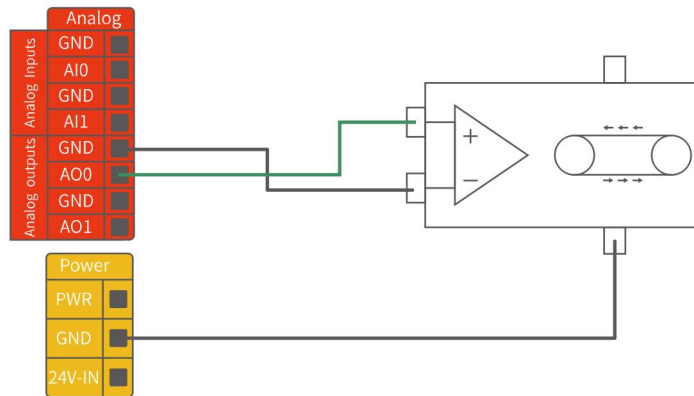
为获得最高准确度，建议遵循以下说明：

- 使用最靠近此 I/O 的 GND 终端。
- 设备和控制器使用相同的接地（GND）。模拟 I/O 与控制器不进行电位隔离。
- 使用屏蔽电缆或双绞线。将屏蔽线与“电源”端子上的“GND”终端相连。

终端	参数	最小值	典型值	最大值	单位
电压模式下的模拟输入					
[AIx - AG]	电压	0	-	10	V
[AIx - AG]	电阻	-	10K	-	Ω
[AIx - AG]	分辨率	-	12	12	位
电压模式下的模拟输出					
[AOx - AG]	电压	0	-	10	V
[AOx - AG]	电流	0	-	20	mA
[AOx - AG]	电阻	-	100K	-	Ω
[AOx - AG]	分辨率	-	12	-	位

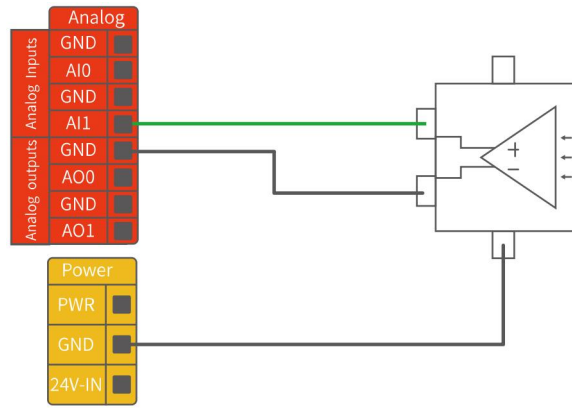
2.5.4.1 使用模拟输出

下例说明了如何利用模拟速度控制输入来控制传送带(接 A00 或者 A01)。



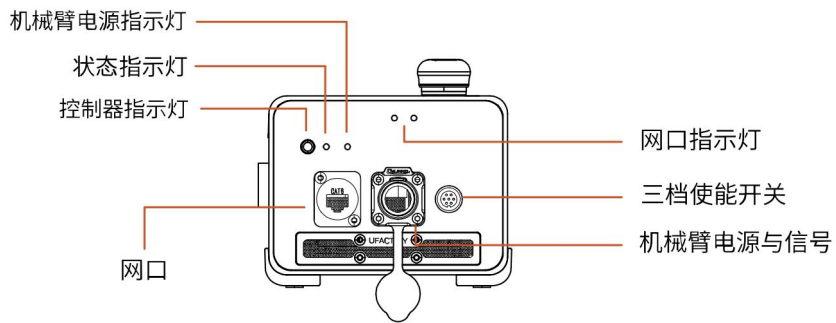
2.5.4.2 使用模拟输入

下例说明了如何连接模拟传感器(接 AI0 或者 AI1)。



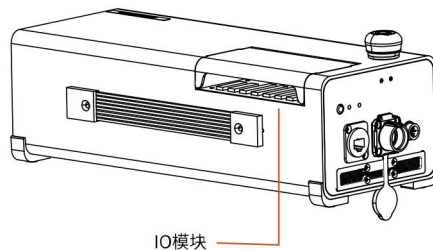
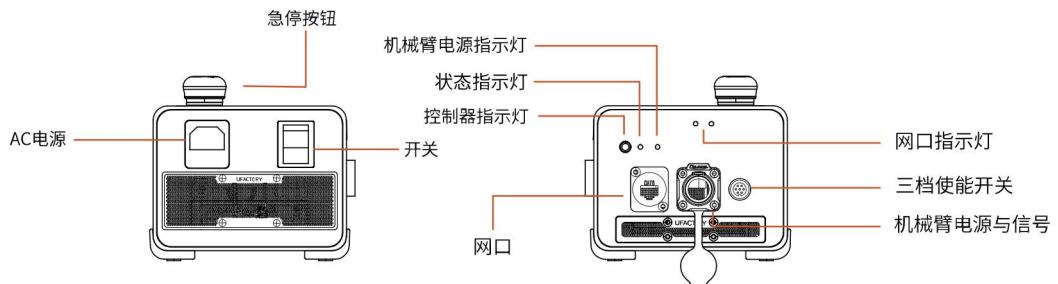
2.5. 通信接口

控制器提供以太网接口，详见下图。



以太网

控制器提供千兆以太网接口。



以太网连接步骤：控制器与电脑通过以太网连接，用网线一端连接控制器网络接口，另一端连接电脑或者局域网网络接口，此时若连接成功，网口指示灯会频繁闪烁。

UFactory 850 机械臂出厂化时控制器的 IP 地址默认网段为 192.168.1.*(2~254)，具体 IP 地址请查看控制器标签。与机器人通信，所用电脑 IP 地址要和 UFactory 850 控制器在同一网段。

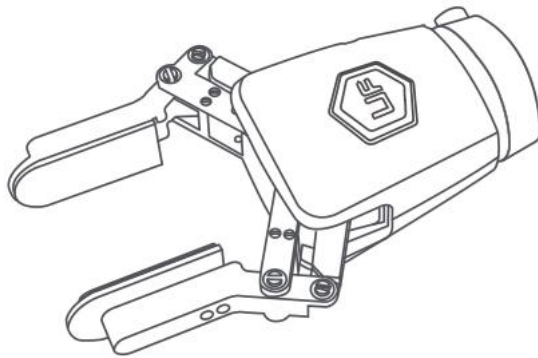
注意：用以太网连接，首先要检查所用电脑 IP 地址是否为 192.168.1.*字段，检查是否开启了网络代理，检查机械臂 IP 地址是否与局域网内的其他电器 IP 地址冲突。如果不是该字段，请自行更改电脑 IP 地址，且关闭电脑网络代理。测试电脑是否能与机械臂通信，电脑打开命令行终端，输入 ping 192.168.1.*(机械臂控制器的 IP 地址)，如果能 ping 成功，表示电脑与机械臂通信成功。

3. 末端执行器

3.1. 机械爪

机械爪是机械臂的末端执行器，能够动态抓取物体。

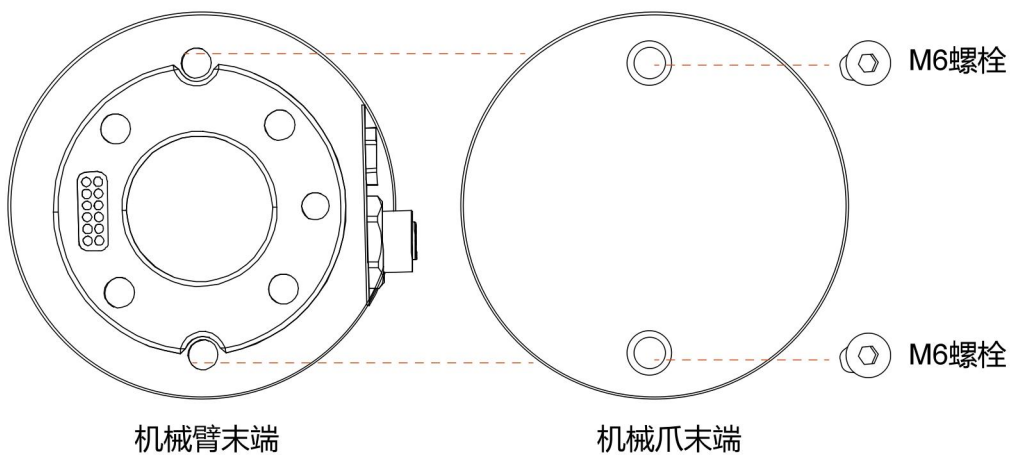
机械爪的行程范围是：-10 到 850，数值越大，机械爪张开程度就越大，反之，数值越小，机械爪张开程度就越小。



机械爪安装

安装机械爪流程：

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）
3. 用 2 颗 M6 螺栓把机械爪固定在机械臂末端
4. 用机械爪连接线连接机械臂和机械爪



注意：

1. 接通机械爪连接线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. 因机械爪连接线长度限制，机械爪接口与末端接口需在相同的方向；
3. 机械爪连接线接通机械爪跟机械臂时注意务必对齐两端接口的定位孔，连接线的公针较为纤细，避免在拆装时使公针弯曲。

机械爪控制过程

1. 使能机械爪
2. 发送位置进行夹合
3. 目前位置的数值范围是：-10 到 850。

注意事项



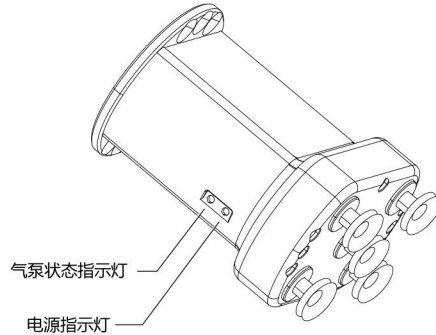
危险

1. 机械臂处于零点姿态时，机械爪会超过安装表面，安装时请把机械臂调整为适合安装机械爪的姿态。
2. 在安装有机械爪的机械臂做轨迹规划时，位置指令间必须进行安全评估，避免碰撞到附近设备。
3. 在安装有机械爪的机械臂做轨迹规划时，回零点操作必须进行安全评估。

机械爪的详细使用说明，请参考 UFactory 850 机械爪用户手册，下载地址：

<https://www.cn.ufactory.cc/xArm-download>

3.2. 真空吸头



真空吸头可动态吸放光滑的平面物体，有效载荷 $\leq 5\text{kg}$ 。真空吸头有 5 个吸盘，可根据吸放物体平面的大小相应减少吸盘，但要将其余不用的吸盘封住。

注意：如果是非光滑的平面，真空吸头漏气会导致不能将物体稳固地吸起。

指示灯状态：当真空吸头上电时，靠近真空吸头的电源指示灯常亮为红色。当气泵处于开启状态时，气泵状态指示灯常亮为绿色。

真空吸头安装

安装真空吸头流程：

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）；
3. 用 2 颗 M6 螺栓把真空吸头固定在机械臂末端；
4. 用连接线连接机械臂和真空吸头；

注意：

1. 接通真空吸头连接线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. 因真空吸头连接线长度限制，真空吸头接口与末端接口需在相同的方向；
3. 且真空吸头连接线接通真空吸头跟机械臂时注意务必对齐两端接口的定位孔，连接线的公针较为纤细，避免在拆装时使公针弯曲。

设置开关真空吸头

示例：

Blockly:



Python-SDK:

```
arm.set_vacuum_gripper(True, wait=False) #开启真空吸头
```

```
arm.set_vacuum_gripper(False, wait=False) #关闭真空吸头
```

Python-SDK 和 UFactory studio 有已封装好的上层函数可调用来开关真空吸头。查看 [UFactory studio- Blockly 命令-末端执行器-真空吸头](#)

真空吸头的详细使用说明，请参考 UFactory 850 真空吸头用户手册，下载地址：

https://www.cn.ufactory.cc/UFactory_850-download

UFactory 850 用户手册-软件部分

1. UFactory studio 使用说明

UFactory studio 是一个控制机械臂的图形化用户应用程序，可以用来设置机械臂参数，实时控制机械臂位置姿态，并且可以通过拖拽 Blockly 命令来创建机械臂的运动程序。

UFactory studio 提供了完善的机器人系统状态信息、故障提示与故障处理方案，即使您通过 UFactory xArm Python SDK /UFactory xArm C++ SDK/ UFactory xArm ROS 或者 UFactory 850 自定义 TCP 协议来创建您的程序，我们仍然建议您在编程、调试阶段，始终保持 UFactory studio 处于运行状态。

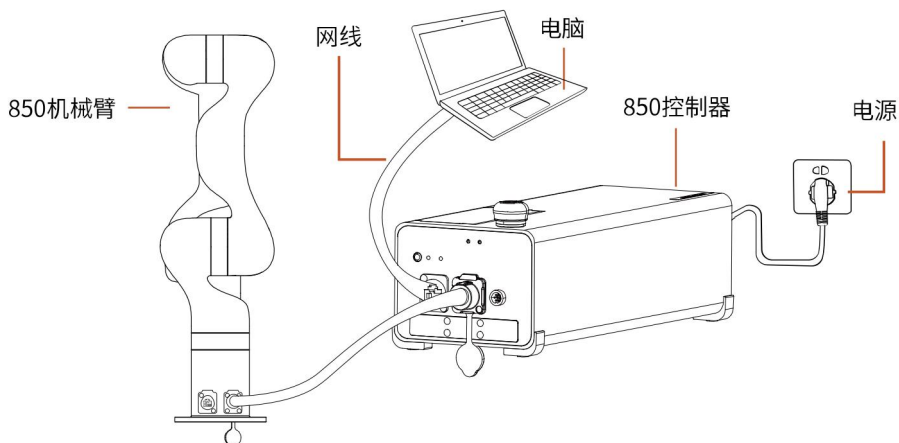
注意：

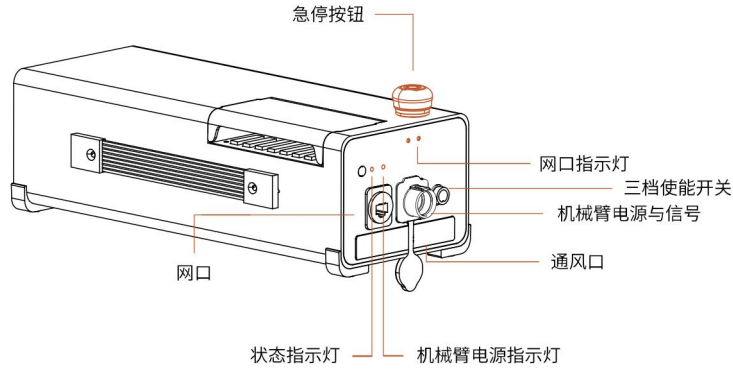
(1) UFactory studio 客户端支持的安装系统有：Windows10、Mac、Linux (Ubuntu16.04/Ubuntu 20.04)

(2) UFactory studio 还支持通过浏览器访问，现已兼容的浏览器有：Google Chrome / Firefox / Safari / Microsoft Edge(Chromium 内核)

(3) iPad、Android 平板和树莓派 4B 等基于 ARM 内核的设备，无法安装 UFactory studio，可通过浏览器访问。使用方法参考下文“[1.2.3 连接机械臂-第 2 节-用浏览器访问 UFactory studio](#)”

1.1 硬件准备





使用 UFactory studio 前，必须先确保硬件安装完好和做好机械臂周边的防护措施：

1. 机械臂已固定在平面上，机械臂在运动范围内做好防护措施。
2. 检查控制器与机械臂，电源，网线是否连接完好。
3. 检查控制器总电源是否开启，ON/OFF 灯亮起表示已供电。
4. 检查控制器是否开机，控制器开关旁边的状态指示灯亮起表示已开机。
5. 检查网络是否连接，控制器正面中间的网口指示灯频繁闪烁表示网络通信正常。
6. 检查机械臂是否供电，急停按钮松开，机械臂电源指示灯亮起表示已供电。

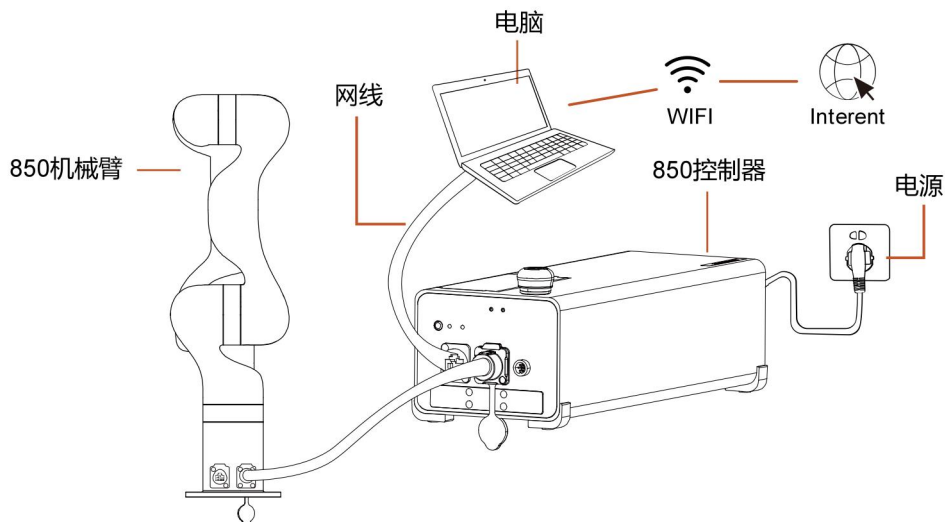
1.2 连接机械臂

1.2.1 机械臂网络设置

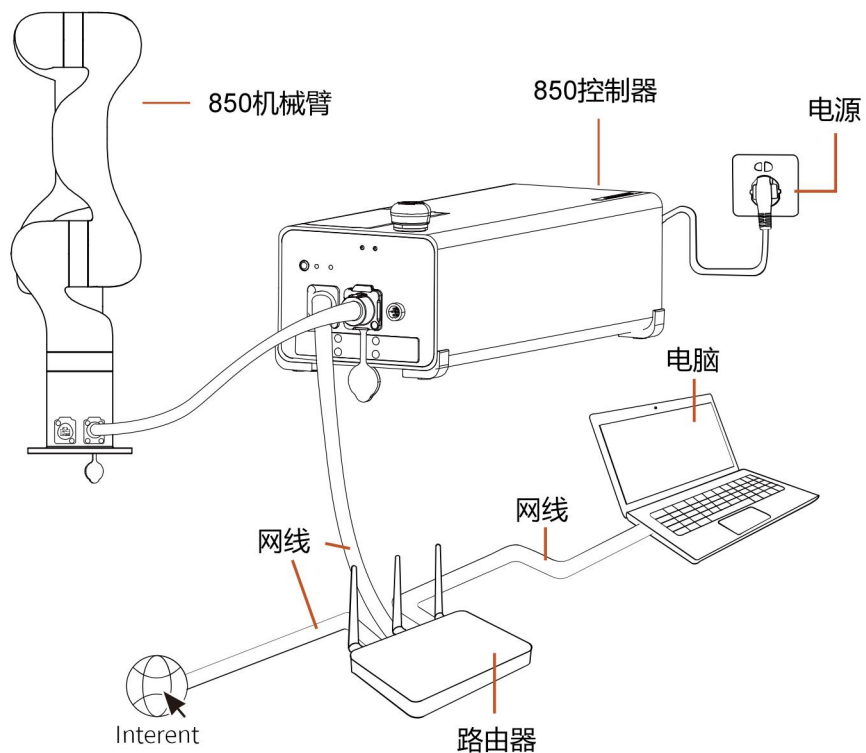
机械臂的网络设置方式有以下四种，您可以根据自己的实际情况选择合适的网络设置方式：

(1) 控制器与 PC 端直连，见下图：

注：推荐使用的连接方法

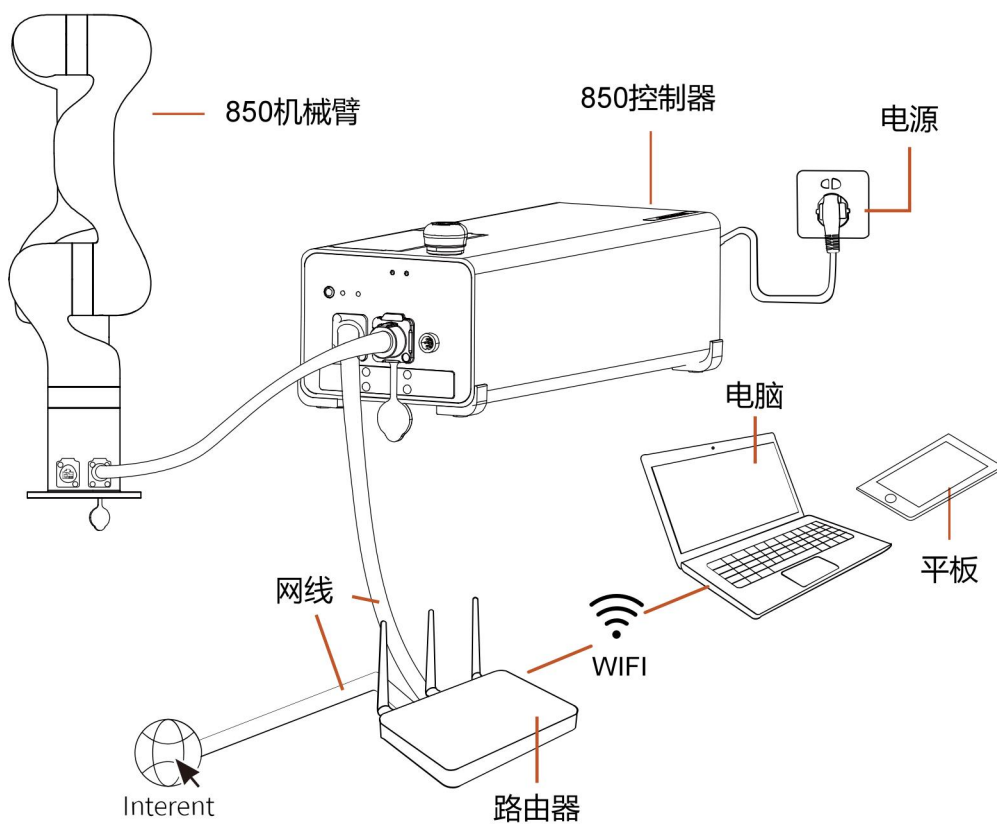


(2) 控制器、PC 与路由器通过网线连接，见下图：

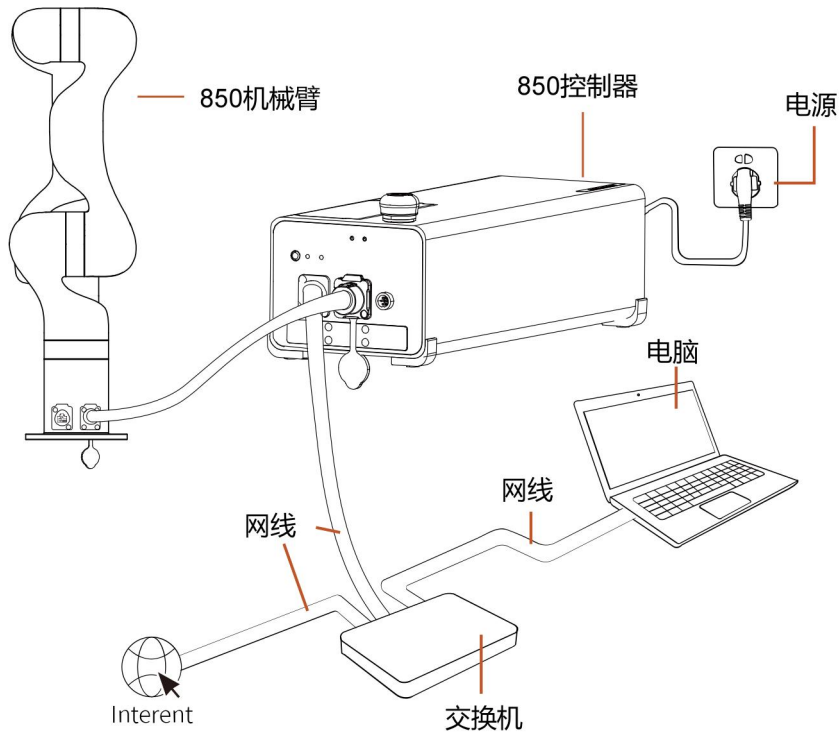


(3) PC（平板电脑）与路由器通过无线网络连接，控制器与路由器通过网线连接，见下图：

注：该连接方法有可能会网络延时、丢包的现象。



(4) 控制器、PC 与交换机通过网线连接，见下图：



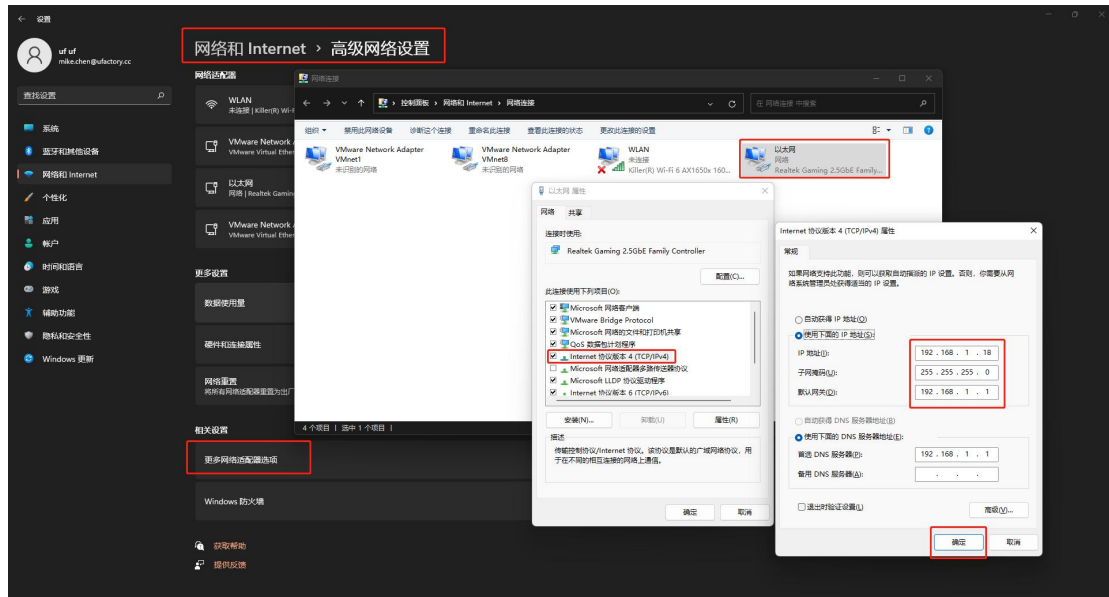
1.2.2 PC 端 IP 设置

在用 UFactory studio 连接机械臂（与机械臂通信）之前，要确保 PC 端的 IP 地址与控制器的 IP 地址为同一网段，控制器出厂化时默认 IP 地址是 192.168.1.xxx（控制器侧面已标示本台设备的出厂 IP 地址），所以要与控制器通信成功，PC 端的 IPV4 网段必须在 192.168.1.1-192.168.1.255 之间（不能与控制器的 IP 地址尾号相同）

PC 端 IP 设置方法：

1. 确保前序硬件连接步骤已经全部完成；
2. 配置 PC 端的 IP 地址

（1）设置 PC 端的 IP 地址：使 PC 端与控制器在同一局域网环境（UFactory 850 控制器的网段在 192.168.1.xxx（x 是 1-255 之间的数字）），所以 PC 端的 IPV4 网段必须在 192.168.1.1-192.168.1.255 之间（不能与控制器的 IP 地址尾号相同），PC 端配置步骤见下图：



(2) 验证 PC 端的 IP 地址：在搜索框内输入 cmd（见下图），打开命令提示符，在 doc 命令行直接 ping 机械臂的 IP 地址：**ping 192.168.1.XXX**（见下图），可成功收发数据包，此时 PC 端 IP 地址以配置成功；



1.2.3 连接机械臂

与机械臂的通信的方式有以下两种，下面对两种方式的具体通信步骤进行讲解。

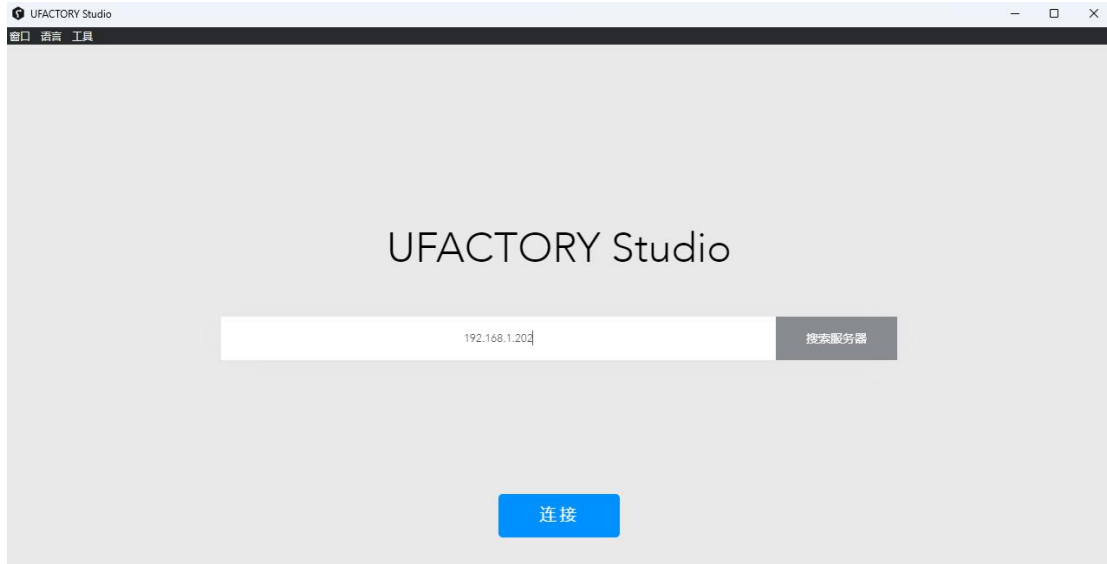
1. 用 UFactory studio 软件访问，你可以通过以下步骤来与机械臂通信：

(1) 下载 UFactory studio

UFactory studio 下载地址：https://www.cn.ufactory.cc/UFactory_850-download

(2) 安装 UFactory studio 软件

(3) 打开 UFactory studio 软件，在搜索框内输入控制器的 IP 地址（控制器侧面已标示本台设备的出厂 IP 地址）

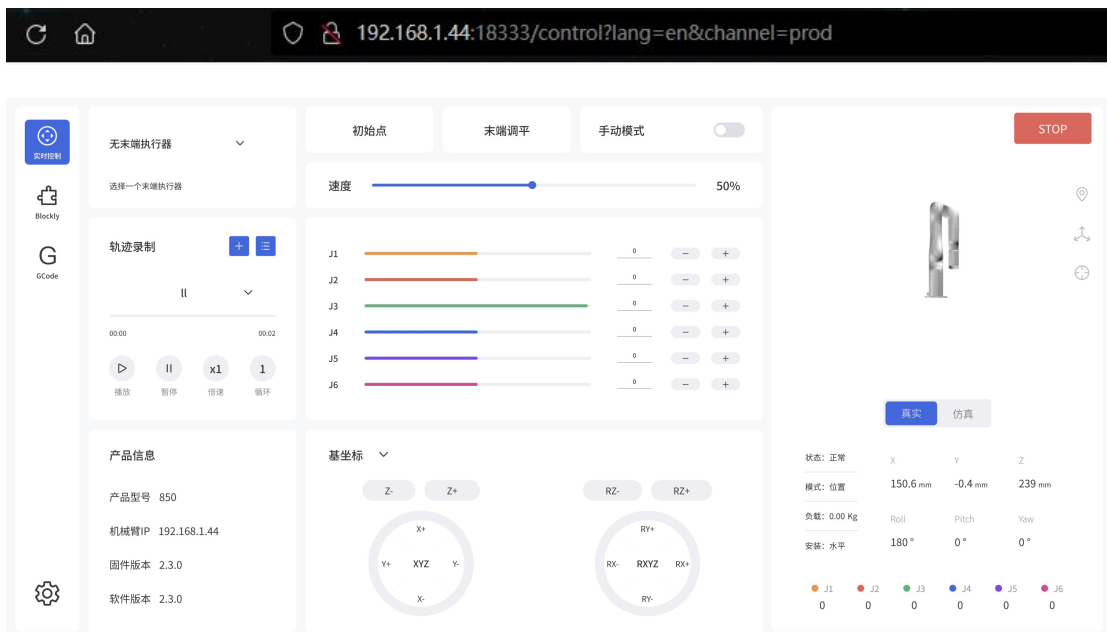


2. 用浏览器访问 UFactory studio:

(1) 打开浏览器

(2) 在搜索栏里输入：控制器 IP:18333

例如：控制器 IP 为 192.168.1.44，则搜索栏里输入 192.168.1.44:18333 来访问 UFactory studio。



1.2.4 回到搜索界面



PC 端：点击【工具】-【搜索】即可回到搜索界面。

1.3 UFactory studio 介绍

1.3.1 UFactory studio 的 3 个主要功能模块

UFactory studio 有 3 个功能模块：



- 实时控制：用于实时控制机械臂的位置，调整机械臂的运动姿态，末端工具，轨迹录制。

- Blockly：是一个图形化编程工具，无需手动编写代码即可通过拖拽代码块的方式来实现编程，进而控制机械臂运动。

- GCode：进入到 GCode 编程页面，用户可以使用 GCode 来控制机械臂运动



- 设置：用于设置机械臂的参数，进行通过设置，安全设置，系统软件升级

1.3.2 工具栏

窗口 语言 工具 帮助

窗口：调整窗口大小可在【窗口】下拉选择，或者采用拖拉边框的形式调整边框大小。

语言：切换语言在右上角工具栏-【语言】，可设置简体中文/英文。

工具：【工具】-【搜索】可回到搜索机械臂界面。

【工具】-【检查更新】可检查软件更新版本。

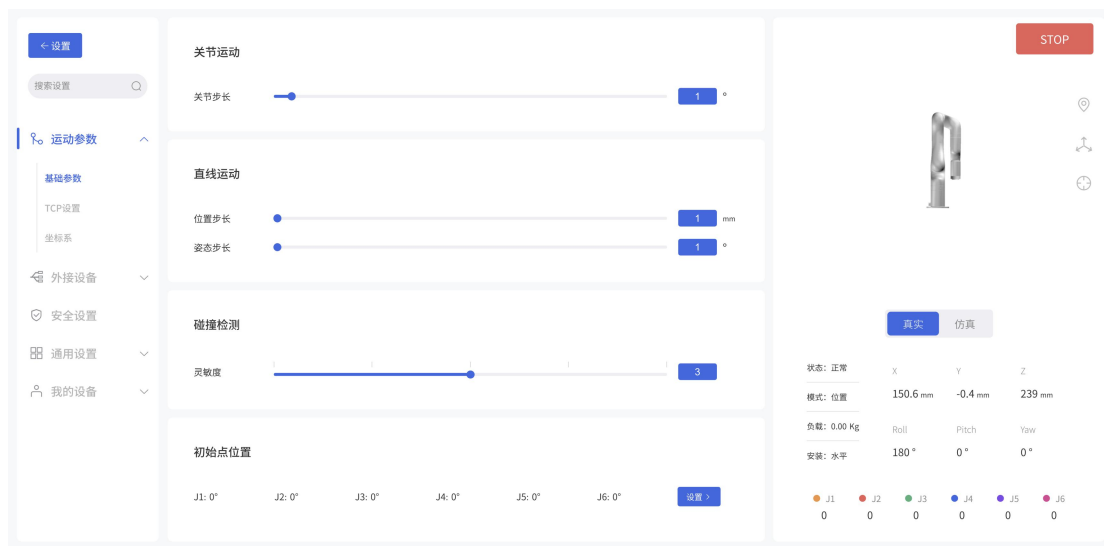
帮助：【帮助】下拉窗可下载机械臂相关手册，链接技术支持，打开论坛和 Github。

1.4 设置机械臂

点击主页的设置按钮进入机械臂设置界面，请根据实际情况，设置机械臂参数。

1.4.1 运动参数

1.4.1.1 基础参数



直线运动

位置步长: 设置笛卡尔空间坐标系 X/Y/Z 轴的步进值，在实时控制界面可以点击对应功能按钮来调节位置步长。

姿态步长: 设置笛卡尔空间坐标系 Roll/Pitch/Yaw 的步进值，在实时控制界面可以点击对应功能按钮来调节姿态步长。

关节运动

关节步长: 设置单关节旋转的步进值，在实时控制界面可以点击对应功能按钮来调节关节步长。

碰撞检测

碰撞灵敏度: 当机械臂运动过程中，关节检测到的力矩偏差超过一定正常范围时，机械臂会自动停止，以防机械臂或操作人员碰撞受伤。碰撞灵敏度范围 1~5 个等级，设定的值越大，碰撞灵敏度等级越高，机械臂触发碰撞保护所需的额外力矩越小。如果

负载或安装方向设置不够准确，可能会引起误报。在某些高负载或高速运动时，如果确认负载或安装方向设置准确，可以尽量将碰撞灵敏度调低一点，但不建议将其调到 3 以下。

初始点位置

设置机械臂初始点可帮助用户在规划轨迹时，使机械臂回到相对安全的位置。

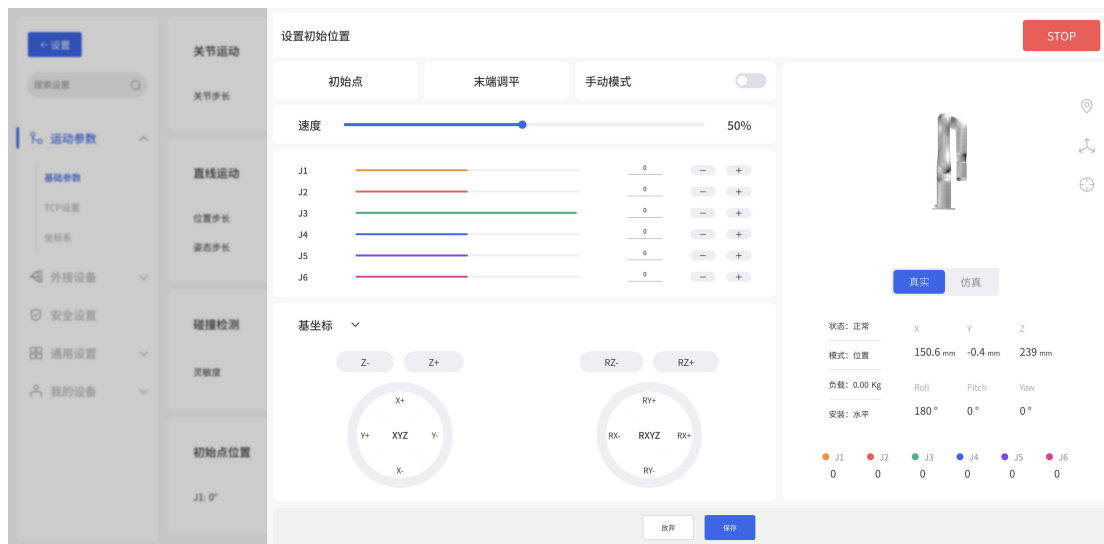
初始点设置步骤：

1. 进入【设置】-【运动设置】

2. 点击初始点旁边的【设置】按钮，弹出控制面板，如下图。其设置的初始点就是控制面板里的【初始点】。

【确定】：保存所做的更改。

【取消】：取消所做的更改。



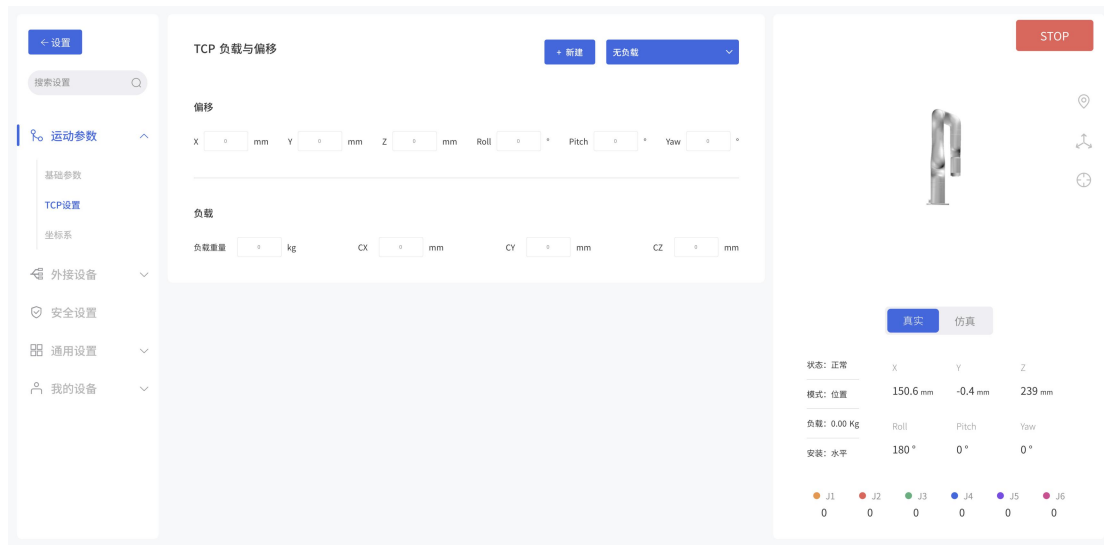
1.4.1.2 TCP 负载与偏移

查看当前手臂的 TCP 负载和偏移情况，并根据实际情况设置 TCP 负载，TCP 重心偏移。

【TCP 负载】：负载重量是指实际的（末端执行器+工件）的重量，单位是 kg；X/Y/Z 轴表示 TCP 的重心位置，单位是 mm。如果末端实际上没有任何负载的时候，TCP 负载和重心都要设置为 0。

【TCP 偏移】：设置 TCP（末端执行器）坐标系与定义在法兰中心的工具坐标系之间的相对偏移量，单位是 mm。如上图 B 点 TCP 坐标系，位置坐标 X、Y 和 Z 决定了 TCP 的位置，而 Roll、Pitch 和 Yaw 决定了其方向。当指定的值均为零时，TCP 与

工具法兰的中心点重合。



在该页面可设置机械臂当前负载和记录备用负载数据。备用负载数据可在 Blockly 编程时引用。

【新建】：可新建负载数据。

【删除】：删除选中的负载数据。 注意：当前默认负载数据不能删除。

【保存】：可保存新添加的负载记录、设为默认负载，和确认删除负载记录。

【放弃】：可放弃保存新添加的负载记录、设为默认负载，和删除负载记录。

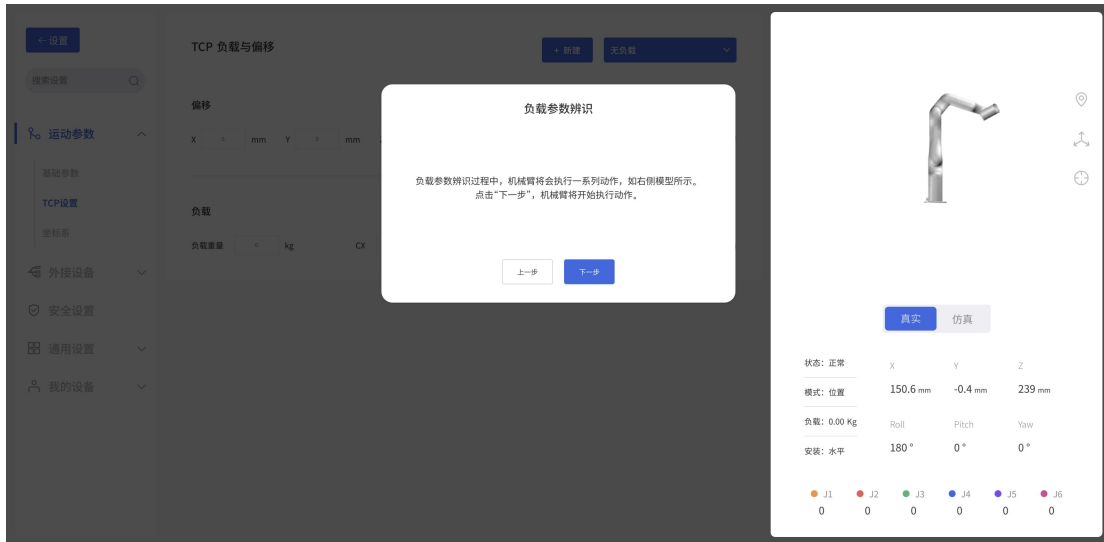
新建负载与 TCP 偏移

新建负载记录有两种方式，手动输入负载和重心和机械臂自动辨识。知道负载重量和大概的负载重心可选择手动输入。负载重心以工具坐标（如上图所示的 B 点坐标）为基点设置。

采用自动辨识的方式，当前机械臂必须是水平安装。机械臂需要运行一系列动作指令计算出 TCP 负载参数。此外请务必确保机械臂附近设备和人员的安全。

注意：新建负载的名称一旦确定后，不能更改名称。





在完成 TCP 负载设置之后，点击【下一步】，即可开始 TCP 偏移设置

当新建 TCP 偏移时，有以下两种方式来设置新的 TCP 偏移参数，如下图：

设置偏移参数

点击“示教获取”，
通过示教5个点来获取位置和方向。
如果已知末端执行器的偏移参数，请选择
“手动输入”。



1) 手动输入

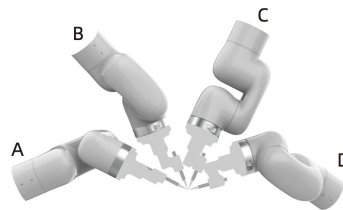
当已知末端执行器的偏移参数时，可选择手动输入其 TCP 偏移参数。

2) 自动辨识

当未知末端执行器的偏移参数时，可点击【自动辨识】按钮，通过示教 4 个点，来获取 TCP 偏移参数。

示教工具中心点位置

接下来，你需要从4个不同的角度，
将机械臂的工具中心点移动到同一个点。
尽量保证4个姿态具有多样性且不在同一平面，例如下图。
如果操作成功，机械臂将自动计算出偏移的位置参数。



示教偏移

注意事项：

1. 示教偏移前，需要确保当前设置的偏移值为[0,0,0,0,0,0]
2. 示教偏移的过程中，不要更改偏移或坐标系偏移参数。

我已仔细阅读并理解以上信息

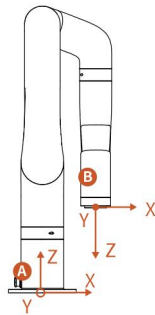
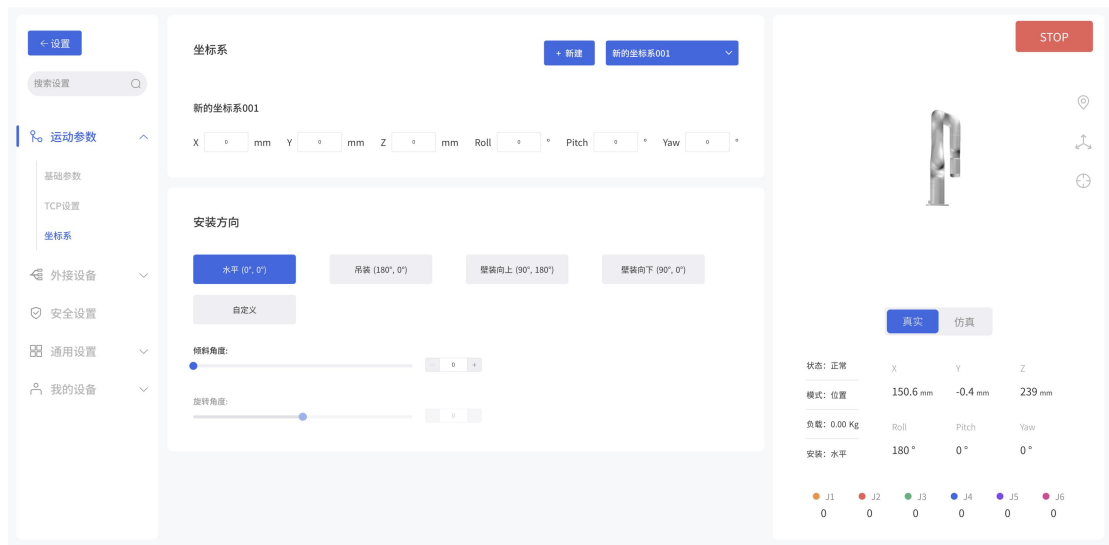




【保存】可保存新添加的偏移记录、设为默认偏移，和确认删除偏移记录。

【放弃】可放弃保存新添加的偏移记录、设为默认偏移，和删除偏移记录。

1.4.1.3 坐标系

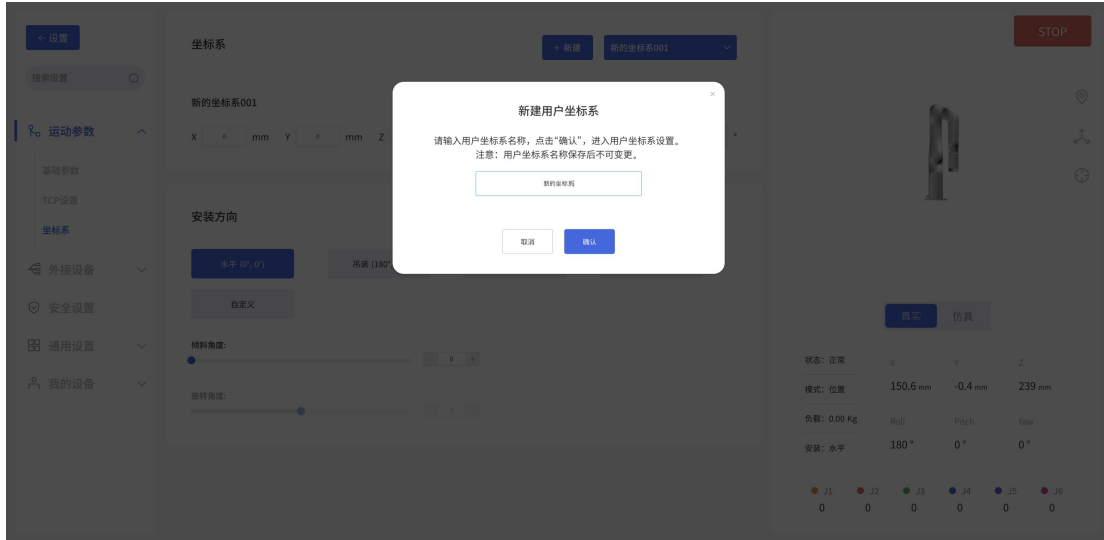


A: 基坐标系
B: 工具坐标系

在该界面，可设置基坐标偏移来自定义用户坐标系。如上图 A 点基坐标，X, Y, Z 是相对于自定义用户坐标系依次偏移的坐标值。Roll, Pitch, Yaw 表示相对于自定义用户坐标系旋转的角度值，是以一定的顺序绕自定义用户坐标系旋转三次得到的方位描述。

【新建】：新建基坐标偏移。

当新建基坐标偏移时，有以下两种方式来设置新的基坐标偏移参数，如下图：



3) 手动输入

当已知用户坐标系参数时，可选择手动输入其用户坐标系参数。

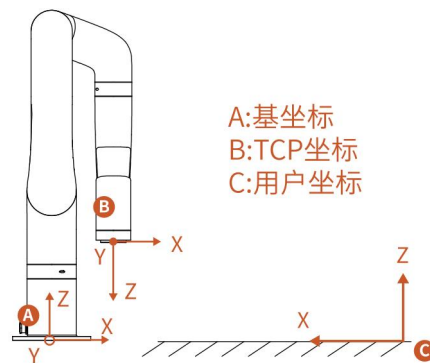
4) 示教获取

当未知用户坐标系参数时，可通过点击**【示教获取】**按钮，通过示教 4 个点，来获取用户坐标系参数。

【保存】：保存修改后的数据。

【放弃】：放弃修改的数据。





示例:

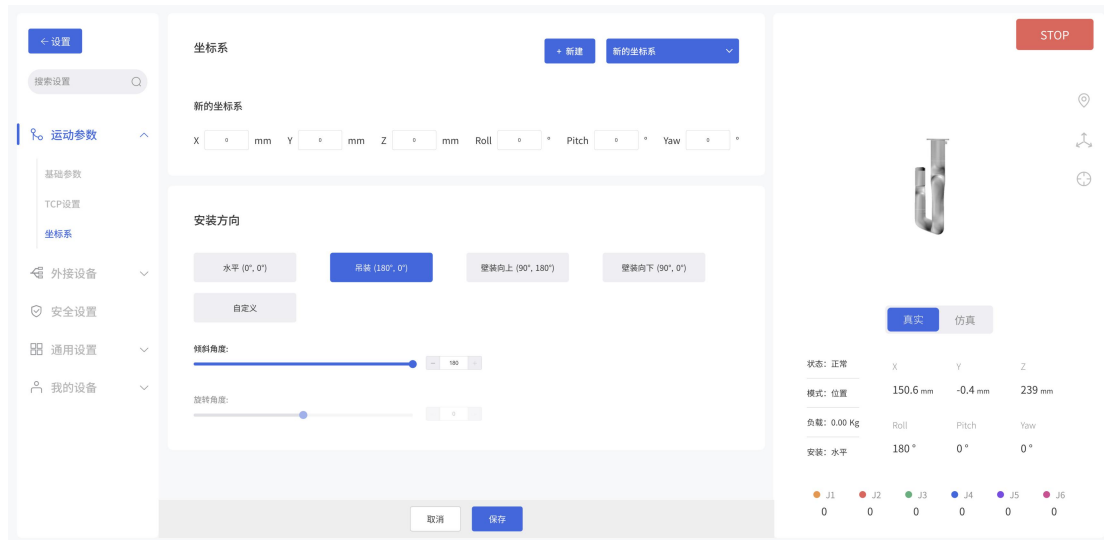
在坐标系 {A} 下: $B(207, 0, 112, 180, 0, 0)$ $D_{AC}=1000\text{mm}$

如果用户想要设置坐标系 {C} 为参考坐标系, 则需要在用户坐标系 {C} 下来表示坐标系 {A} 的方位。

则: 如图所示, 基坐标系的偏移应为 $(1000, 0, 0, 0, 0, 180)$

TCP 坐标由: $B(207, 0, 112, 180, 0, 0)$ 变为: $B'(793, 0, 112, 180, 0, 180)$

1.4.1.4 安装方向

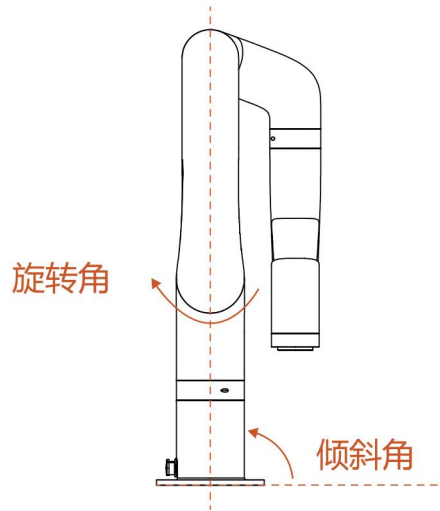


设置机械臂的安装方式，主要是告知控制器当前机械臂的实际安装方向与重力方向的关系。如果机械臂安装方向设置不正确，机械臂将无法准确识别重力方向，会导致机械臂频繁触发碰撞检测，还将导致机械臂进入手动模式后触发“手动模式运动异常”的报错。

机械臂内置的 IMU（惯性测量单元）会检测重力方向，当您设置的安装方向与 IMU 检测的安装方向偏差超过 10° 时，软件会弹出提示。

- **【水平 (0° , 0°)】**：水平面安装 (0°)，机械臂默认为水平安装，水平安装的机械臂不必设置安装倾斜角和旋转角。
- **【吊装 (180° , 0°)】**：机械臂吊装，只需设置机械臂安装方式为吊装，不必设置旋转角。
- **【壁装向上 (90° , 180°)】**：表示机械臂壁装，且机械臂末端朝上。
- **【壁装向下 (90° , 0°)】**：表示机械臂壁装，且机械臂末端朝下。
- **【自定义】**：其他角度安装，机械臂呈一定角度安装，需要根据实际情况同时设置倾斜角和旋转角。

如何确定倾斜角与旋转角？

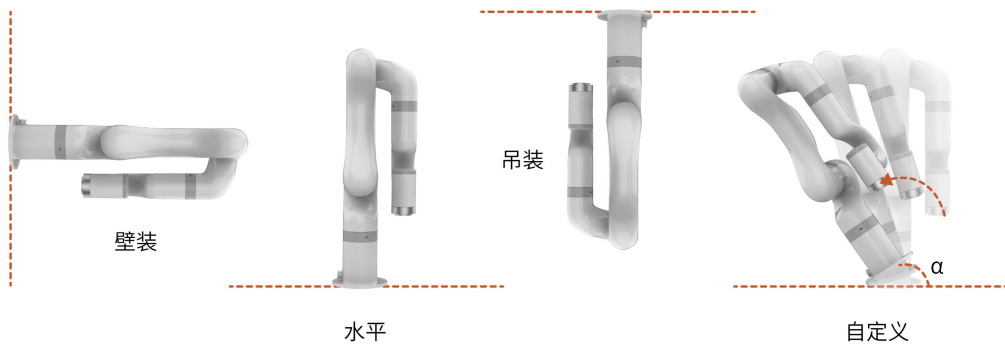


机械臂的初始位置点：在水平线上，把机械臂零点姿态下，人面向机械臂时，机械臂末端在人的左手边且指向斜面向下方向为初始位置点。

倾斜角：机械臂的初始位置点与实际要安装的机械臂的底座夹角作为倾斜角，其范围是 $0^{\circ} \sim 180^{\circ}$ 。

旋转角：机械臂初始位置点到机械臂实际要安装的末端方向间的旋转角度作为旋转角。

旋转角±方向的确定方法：右手握紧，拇指指向垂直于安装斜面机械臂机身的方向，四指指向为正方向，反之为负方向。旋转角范围是 $\pm 180^{\circ}$ 。

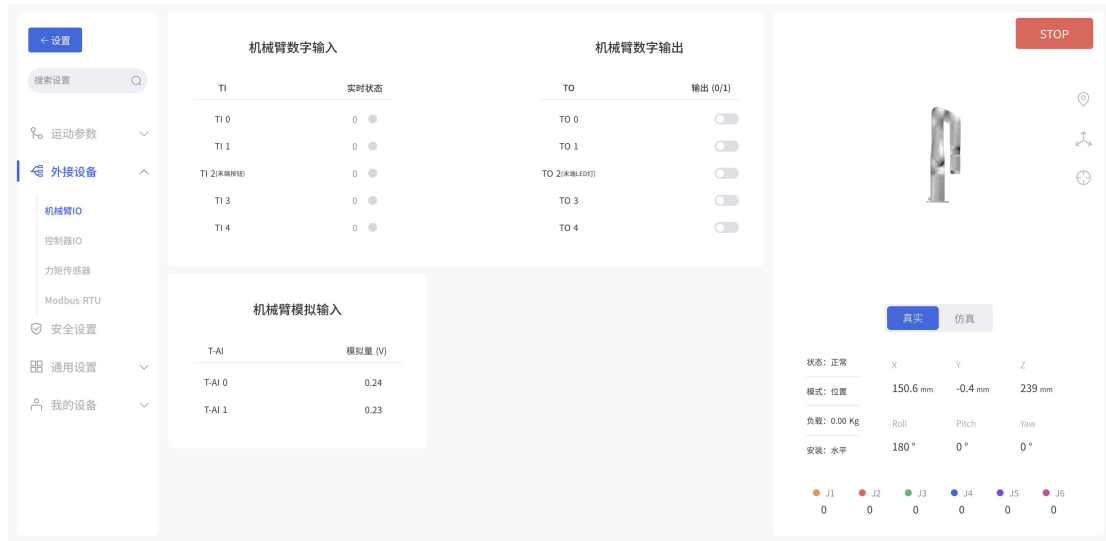


危险

1. 确保机械臂根据实际情况，使用正确的安装方式。
2. 必须安装在坚固防震的表面，避免机械臂出现侧翻的危险。

1.4.2 外接设备

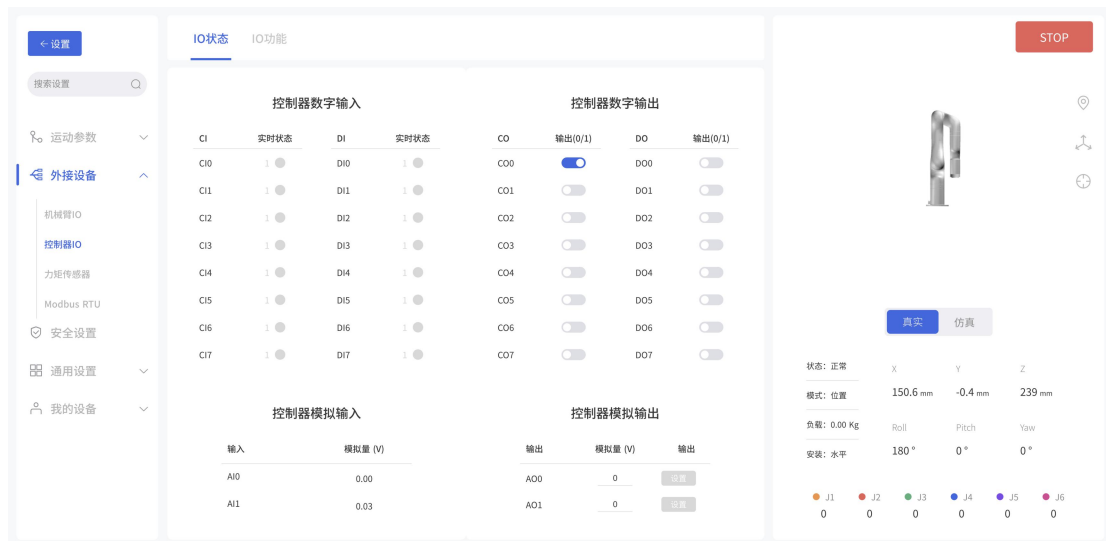
1.4.2.1 机械臂 I/O



通过此页面可以监测到外部设备对机械臂的数字输入和模拟输入状态，并设置机械臂末端的数字输出，在 850 上还可以设置 LED 灯的亮灭以及监控末端按钮的状态。

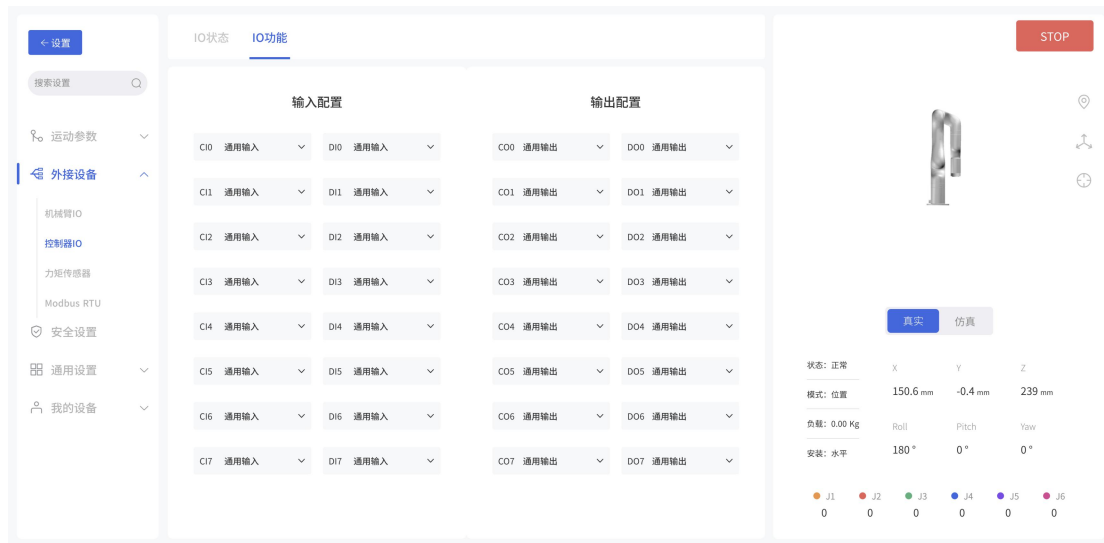
1.4.2.2 控制器 I/O

I/O 状态



在该界面，可监测控制器 I/O 的输入状态, 设置控制器 I/O 的输出状态。也可以设置模拟量输出以及检测控制器模拟量输入。

IO 功能



机械臂控制器共有 32 路数字输入输出信号，只有当 IO 设置为通用输入/输出时，才可在 Blockly 项目和 SDK 中做自定义设置，否则该自定义设置不生效。


输入信号为低电平时触发所选择的下列功能。

【通用输入】：控制器输入只有被设置为通用输入时，用户才可以在 Blockly 或 SDK 程序中自由使用该 IO，否则会导致功能冲突。例如，CI 0 如果被配置为离线任务，则在任何程序中，都不应再使用 CI 0。

【停止运动】：触发 IO，机械臂停止运动。

【防护重置】：触发 IO，使处于防护停止状态的机械臂恢复运动。需配合 SI 使用，使用方法参考 [2.5.2.4 带重置按钮的防护停止](#)

【离线任务】：离线任务可添加多个 Blockly 项目，实现脱离电脑与网络连接，通过触发 IO 运行项目。如上图所示，CI0 设置为离线任务功能，并添加了 Blockly 项目。点

击 **【添加】** 添加 Blockly 项目，点击 **【】** 删除该项目。

【手动模式】 设置为示教模式功能，输入信号持续为低电平时可自由拖拽机械臂。

【缩减模式】 触发 IO，机械臂进入缩减模式。

【使能机械臂】 通过触发 IO 来使能机械臂。

【保存】：保存所做修改。

【取消】：取消保存所做修改。

注意：

DI0-DI7 不能配置以下功能：停止运动、缩减模式、防护重置。

输出信号可选择下列功能：

【通用输出】 控制器输出只有被设置为通用输出时，用户才可以在 Blockly 或 SDK 程序中自由使用该 IO，否则会导致功能冲突。例如，CO 0 如果被配置为运动停止，则在任何程序中，都不应再使用 CO 0。

【运动停止】 系统进入紧急停止状态，输出高电平。符合该紧急停止的动作有：

- 控制器急停按钮按下，机械臂电源切断。
- 通过控制器输入（CI）进入紧急停止。
- UFactory studio 的 STOP 按钮和 Blockly 的 Emergency stop 代码块。
- SDK 的紧急停止 API。

【运动中】 机械臂在运动状态时，输出高电平。

【有错误】 机械臂报错时，输出高电平。

【有警告】 机械臂有警告时，输出高电平。

【发生碰撞】 机械臂报碰撞错误时，输出高电平。

【手动模式生效】 手动模式开启时，输出高电平。

【缩减模式生效】 缩减模式开启时，输出高电平。

【离线任务运行中】 机械臂在运行离线任务时，输出高电平。

【机械臂已使能】 当机械臂处于已使能状态，输出高电平。

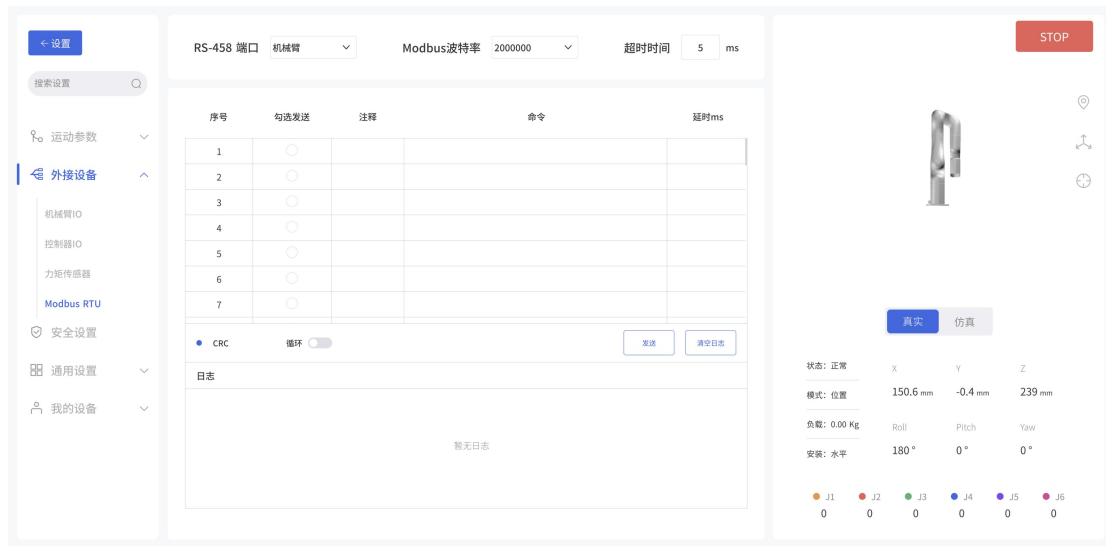
【紧急停止按钮被按下】 机械臂控制器上的紧急停止按钮被按下时，输出高电平

1.4.2.4 力矩传感器



此页面可以对力矩传感器做负载辨识，并设置力矩传感器手动模式方向。

1.4.2.3 Modbus RTU



在 Modbus RTU 界面，用户可以调试机械臂末端 Modbus RTU 和控制盒 Modbus RTU。以调试机械臂末端 Modbus RTU 获取机械爪的位置信息为例：

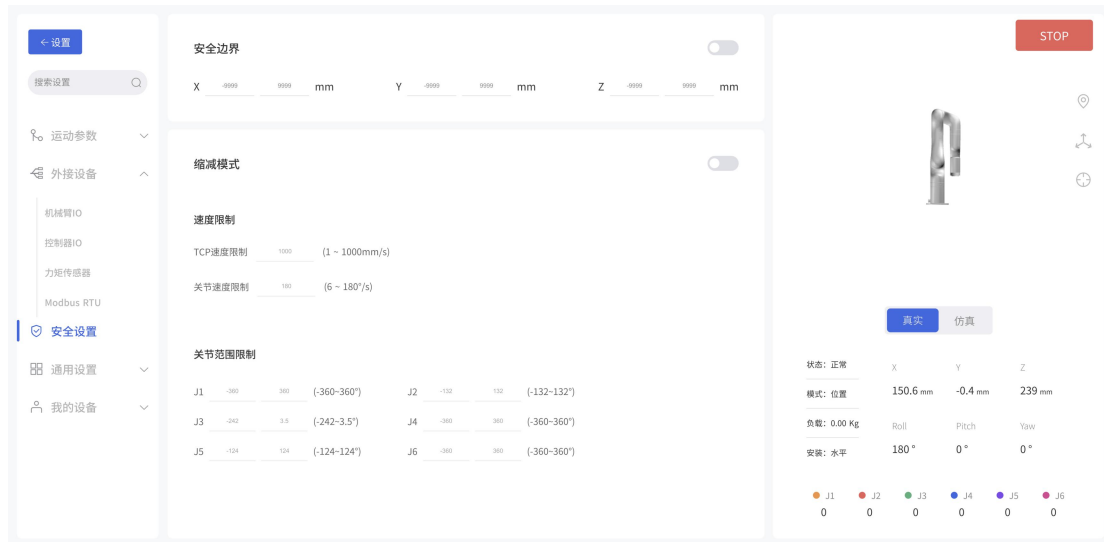
- 选择机械臂 Modbus
- 设置波特率，出厂默认波特率是 2000000。
- 在“命令”框中输入指令，例如：0x08, 0x03, 0x07, 0x02, 0x00, 0x02，注意程序会

自动做 CRC 校验。

- 点击发送，在左侧的调试框可看到收发的信息。

如需循环发送，需设置延时，开启循环功能后点击发送。

1.4.3 安全设置



安全边界

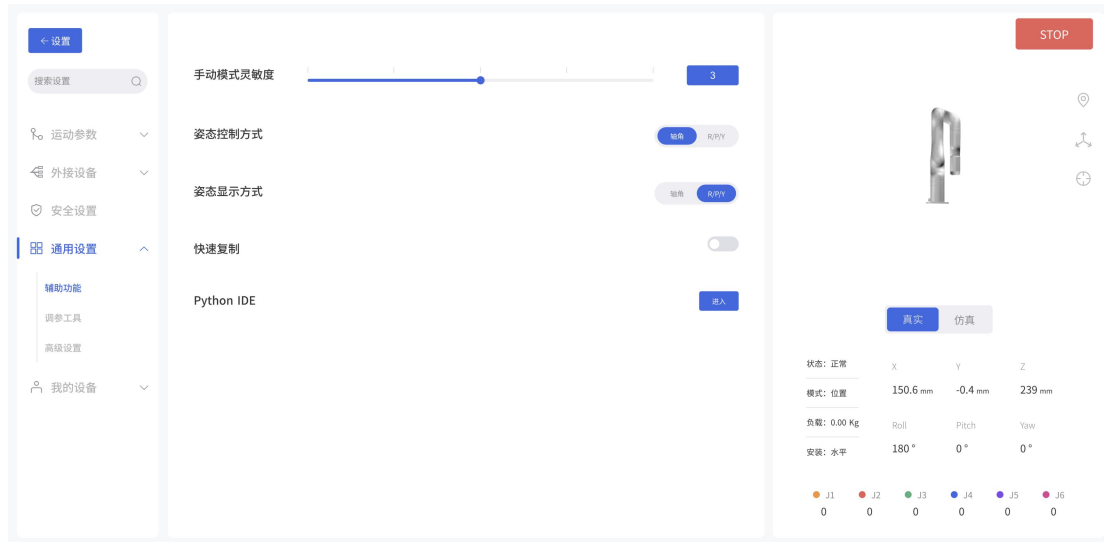
该模式被激活后，可以限制机械臂笛卡尔空间的边界范围，如果工具中心点（TCP）超出设置的安全边界，机械臂将停止运动。此时用户可以将机械臂回归初始点或者移动到其他位置。

缩减模式

该模式被激活后，机械臂在笛卡尔空间最大运动速度、最大关节速度和关节范围将受到限制。

1.4.4 通用设置

辅助功能



在此页面可以调整机械臂的姿态控制方式是以轴角或者 Roll/Pitch/Yaw 显示，并且可以开启机械臂位置的快速复制。

【手动模式灵敏度】：可以调整手动模式下的灵敏度，灵敏度越高，手动模式下拖动手臂所需要的力越小。

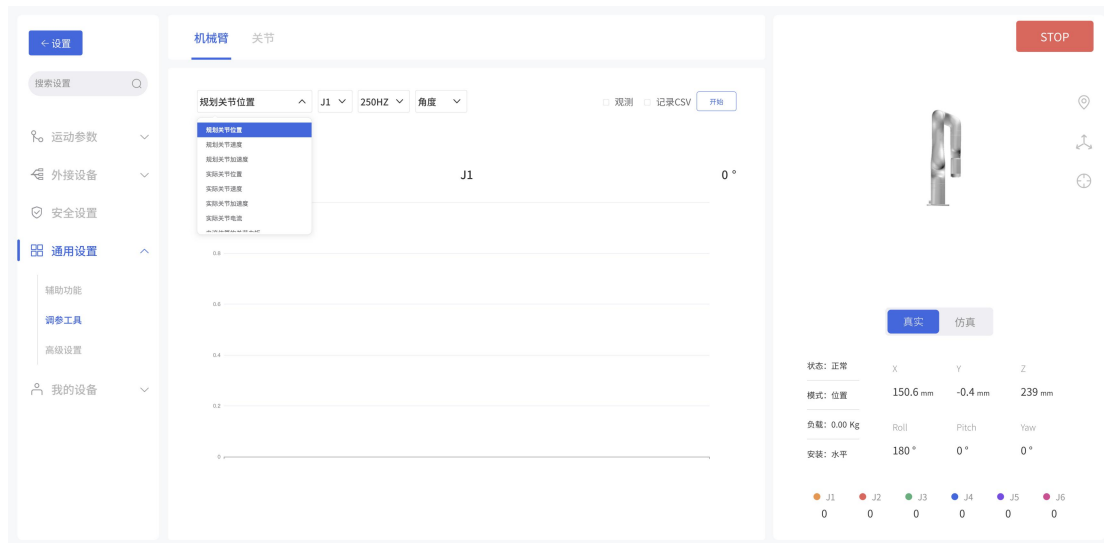
【姿态控制方式】：UFactory 850 机械臂支持通过轴角和 R/P/Y 来调节机械臂的末端旋转，一般建议使用轴角，轴角控制更加直观。这里的选择决定了 UFactory studio 实时控制界面的 TCP 控制方式。下图左侧为轴角控制方法，按钮显示为 Rx/Ry/Rz；右侧为 R/P/Y 控制方法，按钮显示为 R/P/Y。



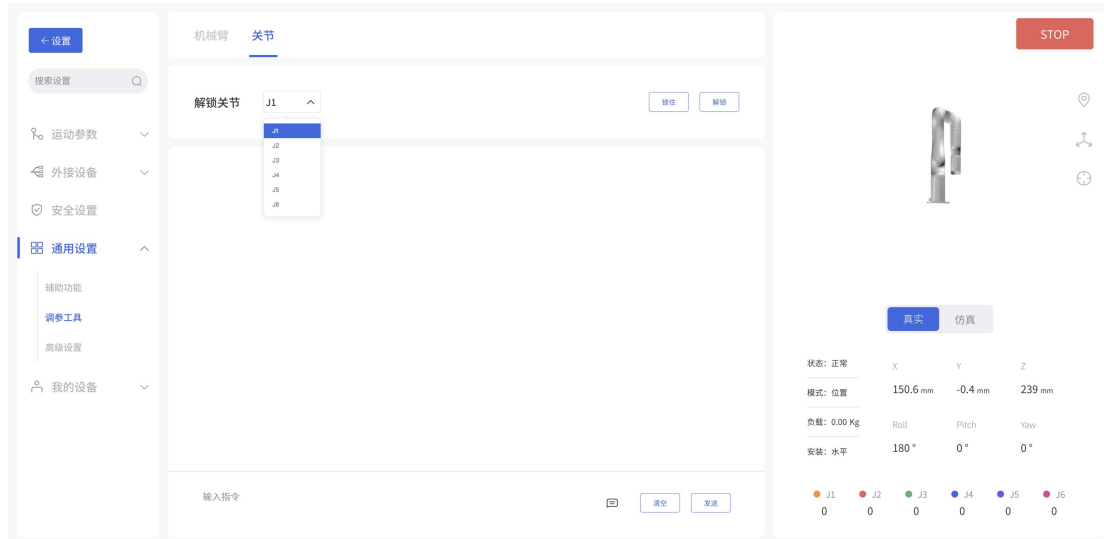
【快速复制】 打开该按钮后，可在实时控制界面复制机械臂的 TCP 坐标和关节角度值。

【Python IDE】：进入到 Python IDE 编程页面，可以在此页面使用 Python 控制手臂运动。

调参工具



此页面可以通过观测机械臂的一些参数值的变化，并绘制图形给到技术支持，协助技术支持远程分析和解决问题，同时还可以勾选记录 CSV，观测完下载下来一并发送技术支持。可以选择观测的值有：规划关节位置，规划关节速度，规划关节加速度，实际关节位置，实际关节速度，实际关节加速度，实际关节电流，电流估算的关节力矩等。



此页面可以通过输入指令的方式对于机械臂进行调参或者查询当前机械臂的相关参数，该功能需要在技术支持的指引下进行操作，此外还能查看到各个关节的电流和电压状况，并可以通过解锁页面对机械臂关节进行解锁，使其能够手动转动关节。



注意：

该功能要在技术支持的指导下操作。（联系技术支持：support@ufactory.cc）

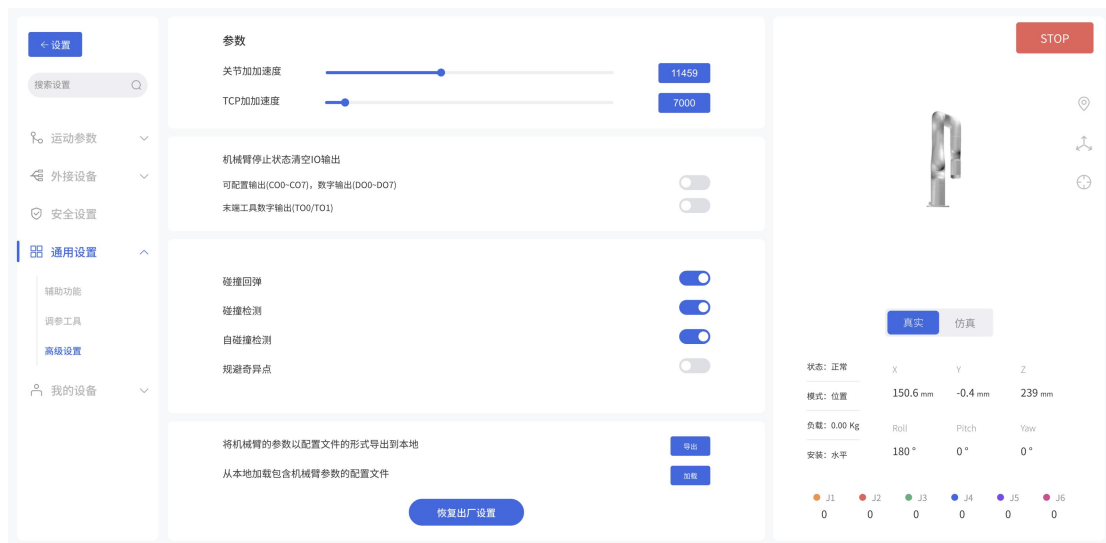
点击【解锁】可解锁单个关节，解锁后的关节没有任何力提供，需要外力扶持，此时可用手拖拽关节转动，确认好位置点后请重新手动锁住全部关节。

注意：

1. 为防止解锁后的关节没有力提供而掉落，解锁关节时请务必用手托住机械臂，做好保护机械臂和周边设施的措施。
2. 解锁关节操作主要用于当机械臂报错时，可手动把机械臂姿态调整到相对安全的位置。注意超过关节范围时，需手动调整关节到范围内。
3. 在“仿真机械臂模式”时，点击解锁关节按钮，也会解锁机械臂的真实关节。

	<p>1. 解锁关节时，必须要有人扶持机械臂姿态，避免机械臂没有外力支撑而坠落，损坏机械臂和周围设备。</p>
	<p>1. 解锁关节后，手动拖动机械臂时，要时刻注意关节转动度数，避免超出机械臂关节转动范围，损坏机械臂内部结构。</p>

高级设置



状态: 正常	X	Y	Z
模式: 位置	150.6 mm	-0.4 mm	239 mm
负载: 0.00 Kg	Roll	Pitch	Yaw
安装: 水平	180°	0°	0°

此页面需要输入密码才能进行访问，默认密码是：admin

请妥善保管密码，当密码遗忘时，请联系技术支持进行密码重置。

若想要修改机械臂的关节加加速度和 TCP 加加速度，可在这里进行修改。

注：

1. 加加速度影响机械臂的加速性能，一般情况下，我们不建议修改该参数。
2. 机械臂未使能情况下，无法修改加加速度。
3. 机械臂有错误的情况下，无法修改加加速度。
4. 机械臂运动时，无法修改加加速度。

当打开【机械臂停止状态清空 IO 输出】后，如果机械臂收到停止指令，则对应的【控制器数字输出】/【末端工具数字输出】将被设置为无效状态。

示例：

当开启“机械臂停止状态清空 IO 输出”后，如果机械臂收到停止指令，“控制器数字输出”或“末端工具数字输出”将被设置为无效状态。

当关闭“机械臂停止状态清空 IO 输出”后，如果机械臂收到停止指令，“控制器数字输出”或“末端工具数字输出”状态将不受停止指令的影响。

打开【碰撞回弹】功能，当机械臂碰撞障碍物时，机械臂会反向回弹一定的距离。如果碰撞灵敏度不为 0，该模式关闭后，机械臂会停留在检测到碰撞的位置。

打开【碰撞检测】功能，会检测手臂是否发生了碰撞。

打开【自碰撞检测】功能，会防止机械臂造成自碰撞。

打开【规避奇异点】功能，会避免手臂经过奇异点而产生超速

点击【导出】按钮，可以将机械臂的参数以配置文件的形式导出。

可导出的机械臂参数主要有：运动参数，TCP 偏移，TCP 负载，IO 设置，安全边界，安装方式，设置的坐标系，高级参数。

点击【加载】按钮，可以导入包含机械臂参数的配置文件。

点击【恢复出厂设置】按钮，机械臂将恢复出厂设置的模式。

注意：

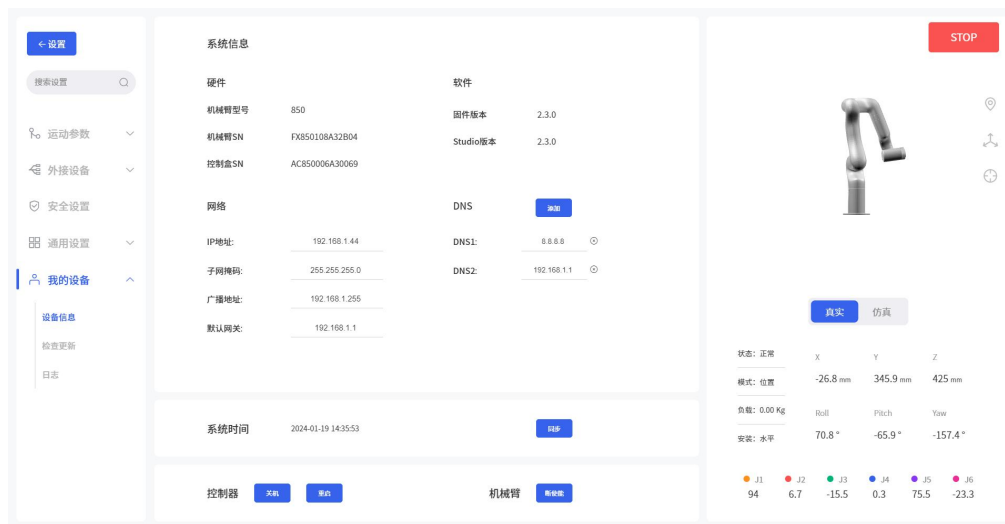
1. 当需要多台机械臂共享一套配置参数时，可通过点击【导出】按钮导出一台已经设置完好的机械臂的参数的配置文件，再将该配置文件通过点击【加载】的按钮导入其他机械臂即可。

2. 当控制器出现故障需要返修时，您可将机械臂的配置文件导出并保存，以免控制器在维修过程中出现原始数据丢失或变更的情况。

3. 机械臂在恢复出厂设置后，其参数会发生变化，为了避免恢复出厂设置后，机械臂参数发生变化，请在机械臂恢复出厂设置前，导出机械臂的配置文件参数

1.4.5 我的设备

设备信息



在设备信息页面，可对控制器进行重启或者关节操作，查看当前机械臂的轴数，IP 地址，固件版本，机械臂 SN、控制器 SN 和软件版本。

系统信息：显示连接机械臂的地址，机械臂固件版本，UFactory studio 软件版本。

网络设置：显示机械臂的 IP 地址，子网掩码，广播地址，默认网关。

可修改和添加 DNS 地址。

注意：若更改了 IP 地址，务必在控制器上做好标注，以免出现不知道控制器 IP 地址的情况，

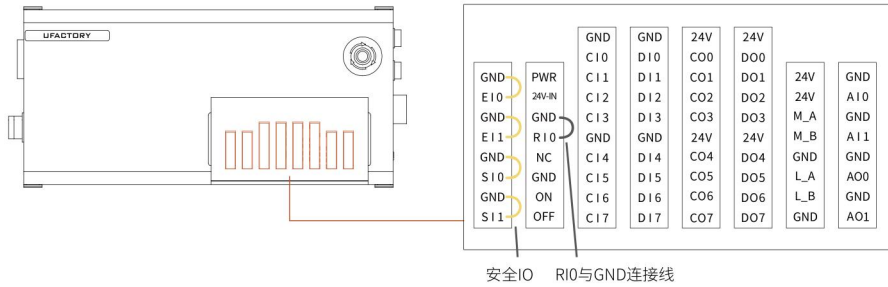
如果忘记或丢失修改后的 IP 地址，可用下面的方法重置 IP。

重置 IP

重置 IP 步骤:

1. 按下急停开关, 关闭 UFactory 850 控制器电源。

2. 用导线将 RI0 与 GND 连接。



3. 开启 UFactory 850 控制器电源, 听到“滴”的一声之后, 表示控制器 IP 已经重置成功。

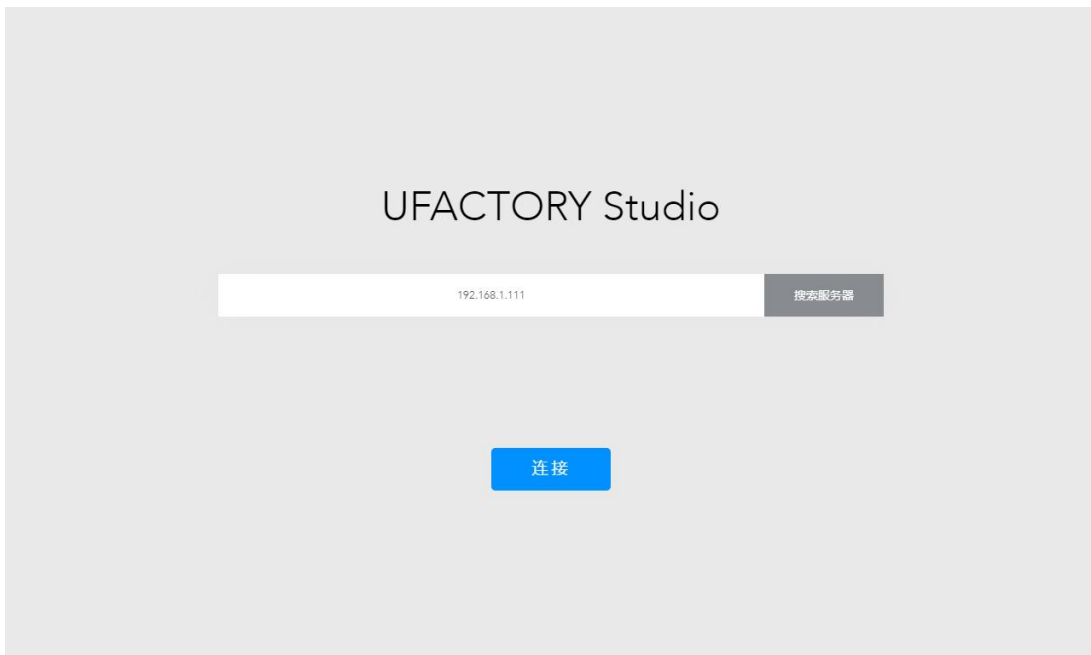
重置后的 IP 为: 192.168.1.111。

4. 请拔掉连接 RI0 和 GND 的导线, 等待控制器启动完成 (60 秒) 即可。

注意:

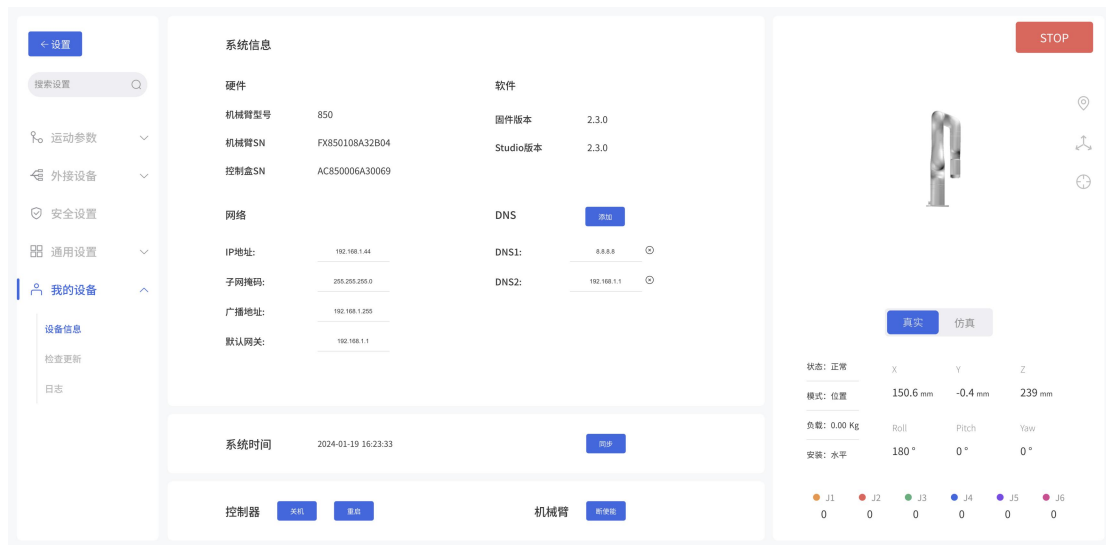
如果不拔掉连接 RI0 和 GND 的导线, 会导致下次重启控制器的时候, 无论你修改的 IP 地址是多少, 控制器 IP 都会被自动修改为 192.168.1.111, 所以重置 IP 以后, 请务必拔掉连接 RI0 和 GND 的导线, 然后再修改 IP。

5. 在 UFactory studio 搜索框内输入 192.168.1.111, 连接机械臂。

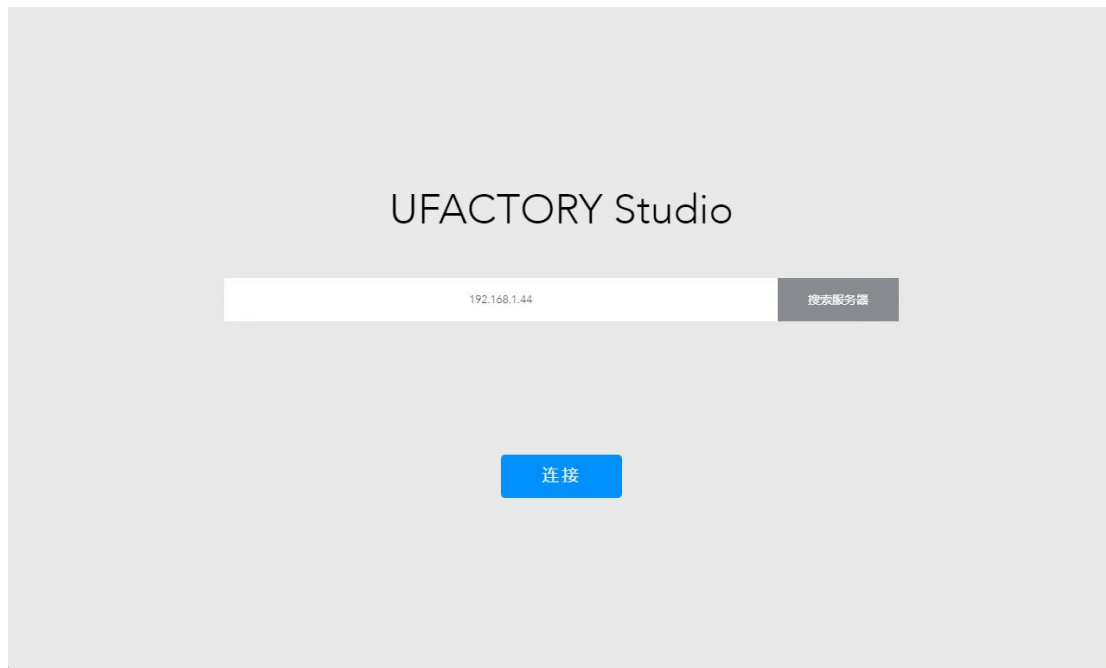


6. 如果你需要修改 IP, 在【设置】→【我的设备】→【设备信息】中修改 IP 即可。(例如:

修改的 IP 为：192.168.1.44)



7. 重启控制器，在 UFactory studio 搜索框内输入你修改的 IP，连接机械臂。



系统时间：显示机械臂系统的当前时间，点击【同步】可以同步上位机的时间

关闭控制：可以进行控制器的【关机】，【重启】

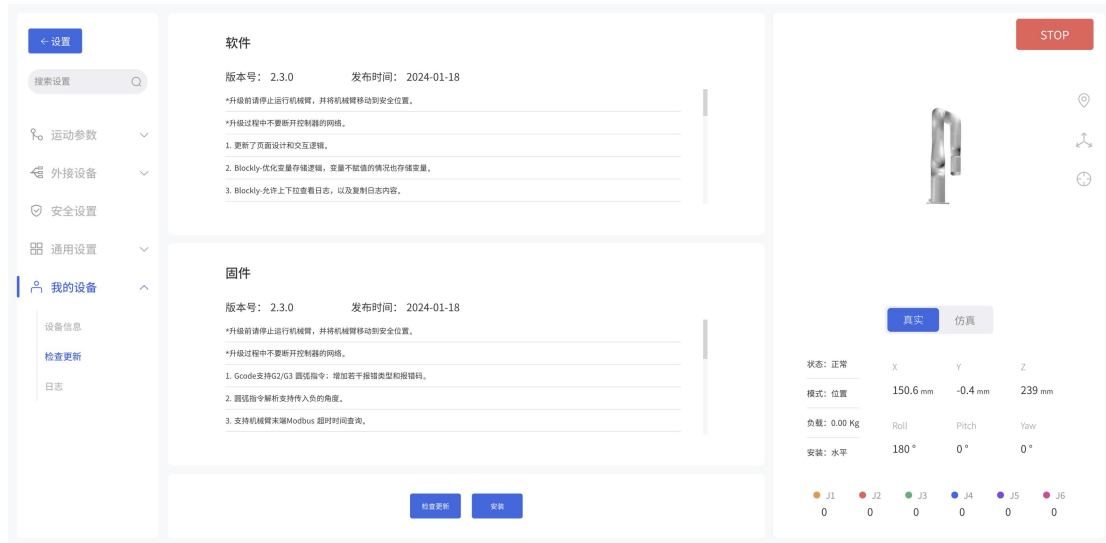
关机/重启：用于关机或重启机械臂的控制器。注意关机/重启按钮不会关闭机械臂的供电电源和总电源。

【关机】：点击此按钮，页面重新跳回到【搜索机械臂控制器地址】页面，控制器会关机。此操作等同于长按控制器的电源按钮，关机过程需要等待 2-3s。

【重启】：点击此按钮，机械臂控制器会重新启动，重启动作完成后，弹窗关闭，会自动重新连接机械臂。此操作等同于在控制器里的控制器关机-开机过程，重启过程需要等待 3-4 分钟。

注意：UFactory 850 关机与 UFactory 850 重启按钮均不影响控制器的总电源和机械臂的供电电源。

检查更新

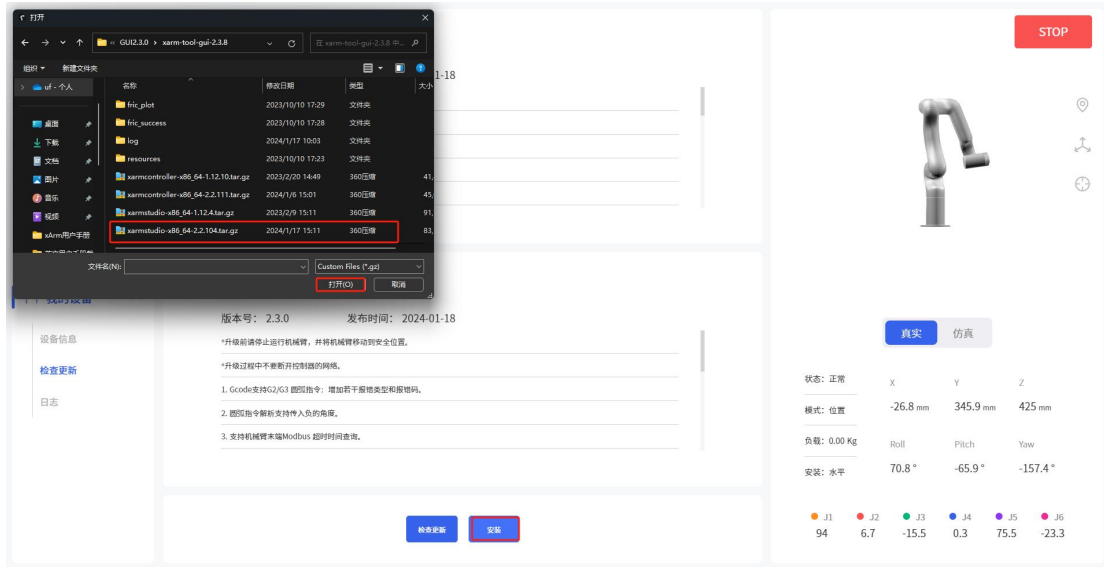


更新软件和固件需确保电脑、控制器所在的局域网能跟外部网络通信。此外，控制器需接入到局域网，保证控制器能与外部网络通信。

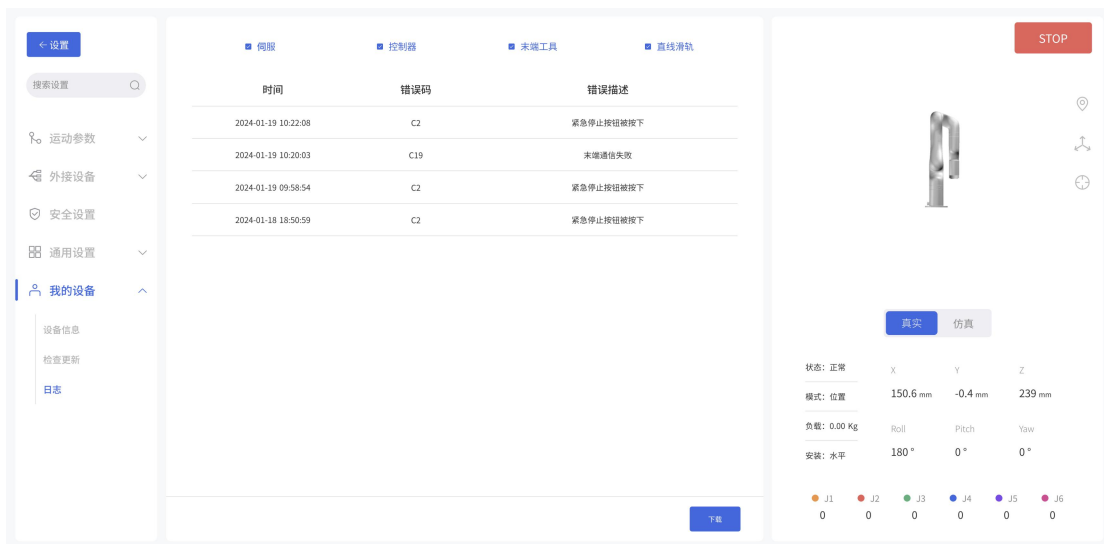
【检查更新】按钮来获取控制器最新的 UFactory studio 和 UFactory 850 固件版本信息。检查到有更新版本，点击**【下载】**可自动完成升级更新。

【安装】：点击进入到离线安装 UFactory studio 和 UFactory 850 固件窗口，选中提前下载好的升级包，即可更新固件和 studio 到最新。

[点击下载最新离线升级安装包](#)



日志



可查看控制器报错日志、伺服报错日志和末端执行器报错日志。

点击【下载】按钮，可下载报错日志。

1.5 实时控制



1.5.1 页面介绍

- IP 地址
 - 显示当前链接的机械臂控制器的 IP 地址。
- 负载
 - 显示手臂当前默认负载参数
- 机械臂状态
 - 正常：机械臂没有在运动，但是准备好进入运动状态。
 - 运动：机械臂正在运动。
 - 停止：机械臂处于运动停止状态。
 - 暂停：机械臂处于运动暂停状态。
- 模式
 - 机械臂当前所处的模式，默认为【位置模式】
- 位置信息
 - 机械臂当前所处的位置信息，包含 $[X, Y, Z, R, P, Y]$ ，若选择轴角显示方式，则为 $[X, Y, Z, R_x, R_y, R_z]$
- 真实机械臂/仿真机械臂
 - 仿真机械臂：表示机械臂已连接，当前为仿真机械臂。可通过 UFactory studio 界面

控制虚拟机械臂运动。

- 真实机械臂：表示机械臂已连接，当前为真实机械臂。可通过 UFactory studio 界面控制真实机械臂运动，该虚拟机械臂会实时反应真实机械臂的位置姿态

注意：一台机械臂只能处于一种模式下（真实机械臂模式/仿真机械臂模式）。

- 紧急停止

- **【STOP】** 点击紧急停止按钮可使机械臂立即停止当前运动并清除所有缓存指令。

注意：

UFactory studio 的“STOP”按钮与控制器上的紧急停止按钮不同。

UFactory studio 的“STOP”按钮：会使机械臂立即停止运动并清除所有缓存指令。

控制器上的紧急停止按钮：发送停止命令后会断开机械臂供电，机械臂姿态会轻微刹车下坠。

- 末端调平

- 点击这个按钮后，工具法兰将调至水平，即将 pitch 和 roll 调至固定值 0° 和 180° 。

- 初始点

- **【初始点】**：如果初始点设置为所有关节角度值为 0，长按初始点按钮，可使机械臂回到零点姿态。可在主页点击 **【设置】-【运动参数】-【基础参数】-【初始点位置】**，设置初始点。

- **【初始点】** 点按为步进运动，长按为连续运动。



危险

机械臂安装了末端执行器的情况下，如果要进行回零点操作时，请务必评估回零点路径会不会碰撞障碍物或者安装机械臂的固定面。

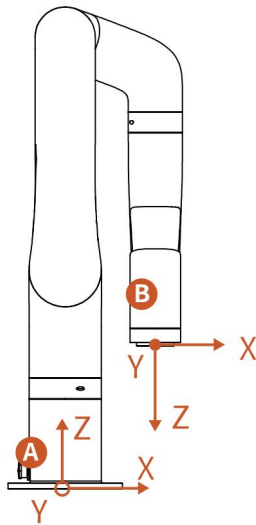
1.5.2 位置控制

1.5.2.1 简介

用户可以基于基坐标系、TCP 坐标系来控制机械臂的运动状态。笛卡尔空间运动的机械臂使工具在路点之间进行线性移动。这意味着每个关节都会执行更为复杂的移动，以使工具保持在直线路径上。目标点确认后，末端路径是唯一的，轨迹过程中对应的姿态是随机的。基坐标系的 X、Y 和 Z 控制工具的位置，箭头指向为+方向，单位是 mm。TCP 坐标系的 Roll /Pitch/Yaw 控制工具的方向，单位是度（°）。

直线运动和圆弧直线运动属于笛卡尔空间轨迹规划，需要做逆运动学求解。因而可能存在无解、多解、逼近解的情况；而且由于关节空间和笛卡尔空间的非线性关系，可能会出现轴运动超出其最大速度和加速度限制的情况。

1.5.2.2 TCP 坐标系



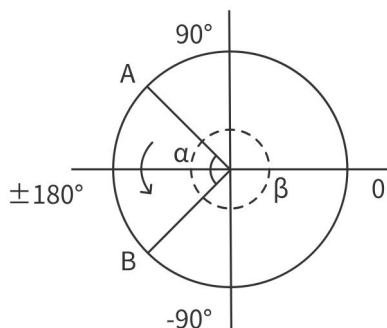
A: 基坐标系

B: 工具坐标系

机械臂末端执行器法兰中心点（TCP）来定义的 TCP 坐标系，是依次绕基坐标系的 X/Y/Z 轴旋转 $[180^\circ, 0^\circ, 0^\circ]$ 的结果。TCP 坐标系的 X/Y/Z 空间指向会根据旋转角度的改变而改变。

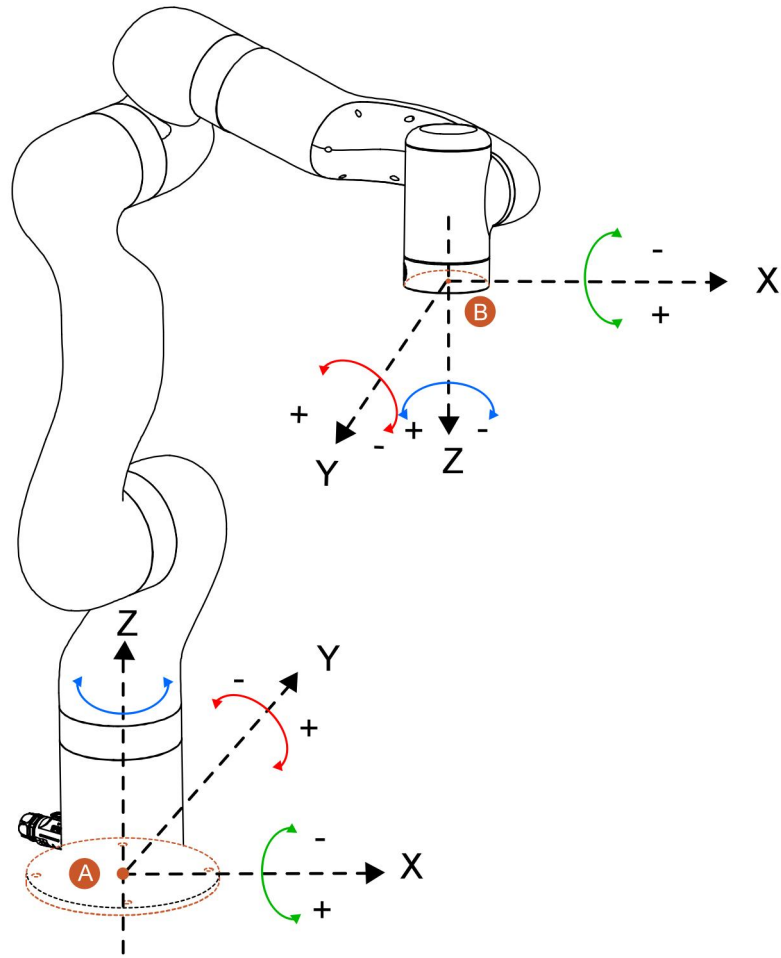
- Roll /Pitch/Yaw 分别绕选定坐标系的 X/Y/Z 旋转，设置数值的±表示在旋转角度范围内圆的数值，其旋转方向会根据两点之间较小的角度方向旋转，这一点尤其要注意，严格控制两点之间的偏转角度大小来控制旋转方向，必要时在两点之间插入第三个点。如下图所

示，如果要从位置点 A 偏转到位置点 B，机械臂走的是 α 夹角方向，如果需要走 β 夹角方向，需要在 β 夹角之间插入新的位置点，而且要保证插入的位置点与 A 形成的夹角小于 α 。



- Roll/Pitch/Yaw 三者的 $\pm 180^\circ$ 在空间上的点是重合的，取值范围都为 $\pm 180^\circ$ ，所以机械臂上报位置时， $\pm 180^\circ$ 都有可能。
- RPY [rad] 滚动角、俯仰角和偏航角 (RPY)。RPY 旋转矩阵 (X、Y、Z 旋转) 由以下公式确定：

$$R_{\text{rpy}}(r, p, y) = R_Z(y) \cdot R_Y(p) \cdot R_X(r)$$



A: 基坐标系

B: TCP 坐标系



危险

1. 机械臂在记录笛卡尔位置点前必须检查 TCP 偏移量。

1.5.2.3 速度设置

用于调节实时控制界面 UFactory 850 的运动速度。（注意实时控制界面的速度最大值不是机械臂真实的最大运动速度，如需让程序高速运行，可在 Blockly 运动程序中添加速度指令。）

【关节运行速度】

- 范围是 $1^\circ / s \sim 180^\circ / s$ ，此速度的最大值还会受机械臂运行时的负载、速度和姿态等因

素的影响有所变化，最大速度并非是绝对的。

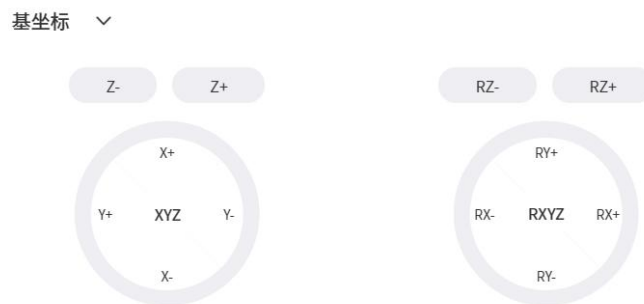
注意：关节运动指令间的速度不连续，机器人在运行关节指令时会有短暂的停顿。

【TCP 运行速度】

- TCP 运行速度范围为 1mm/s ~ 1000mm/s，速度的最大值还会受机械臂运行时的负载、速度和姿态等因素的影响，如果设置的速度接近极限速度，机械臂将减速或引发报错机制。当一条指令包含有位移和旋转时，计算位移运动和旋转运动所需要的时间时，速度取决于用时多的一方。但是原则上最好将位移指令和旋转指令分开。

1.5.2.4 操作方式

UFactory 850 操作界面



基坐标

【工具坐标】：可实现基坐标系和工具坐标系控制功能的切换。

【复制图标】：可通过点击该按钮，复制机械臂的 TCP 坐标值。

【位置/姿态实时状态显示】：X/Y/Z 表示机械臂的工具中心点（TCP）位置在基坐标偏移下的坐标。姿态下的 Roll/Pitch/Yaw 表示在基坐标偏移下旋转的角度值，是以一定顺序绕选定坐标系旋转三次得到的方位的描述。

【位置实时控制】：X/Y/Z 分别控制选中坐标系的 X/Y/Z 轴，点按为步进运动，长按为连续运动。

【姿态实时控制】：Roll/Pitch/Yaw 分别控制选中坐标系的 Roll/Pitch/Yaw，点按步进，长按连续运动。

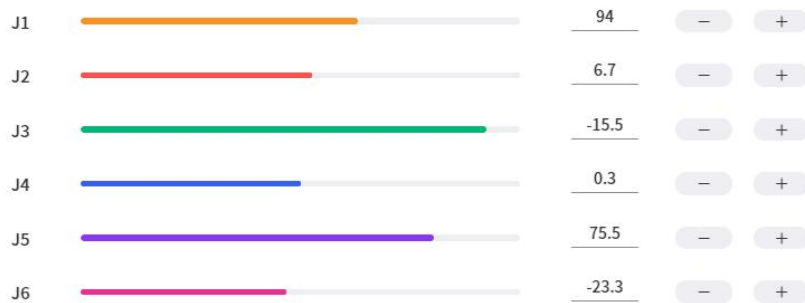
可在主页点击**【设置】-【运动参数】-【基础参数】-【姿态步长】**中设置点按步长。

【末端调平】：点击这个按钮后，工具法兰将调至水平，即将 pitch 和 roll 调至固定值 0° 和 180°。

1.5.3 关节控制

UFactory 850 机械臂由关节模块组成。通过协调每一个关节的旋转角度来控制末端执行器的位置。

关节运动是以最快的路径到达目标点，末端轨迹不是直线，速度单位是°/s。目标点确定之后，末端路径和轨迹过程中对应的姿态都是唯一的。目标点有多个关节联动时，完成这条指令每个关节用时一样。



: 进度条表示关节范围，数字表示当前关节和关节度数。

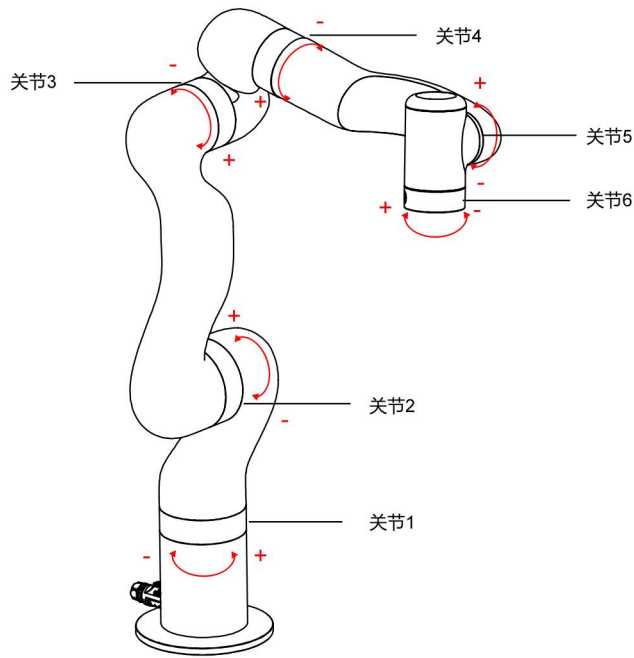


: 可通过点击该按钮，复制机械臂的关节角度值。

操作方式:

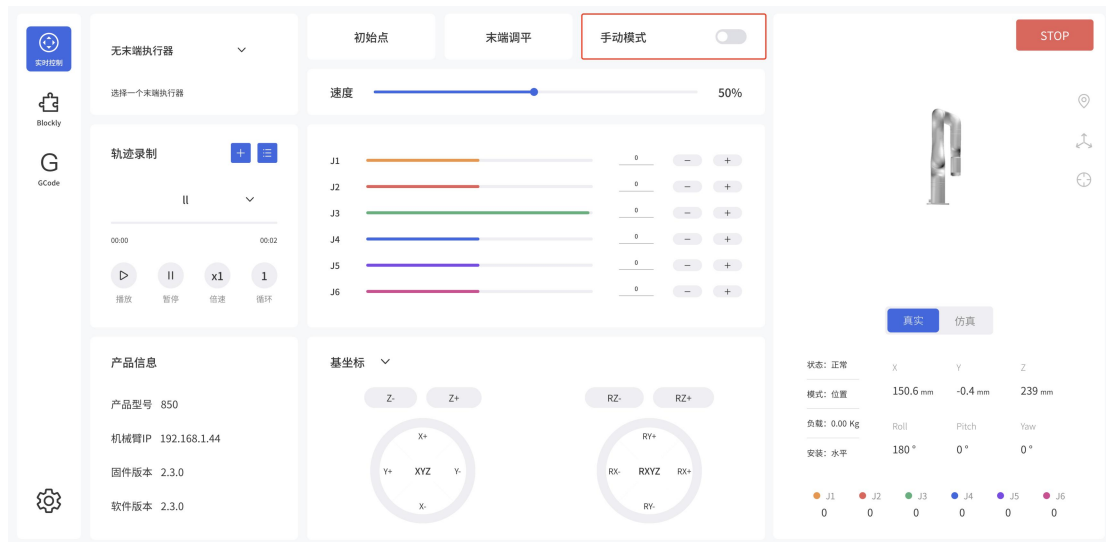
- 1) 点按【+】或【-】为关节角度步进，可在【设置】-【运动参数】-【关节运动】-【步长】中设置步进角度。
- 2) 长按【+】或【-】为关节连续正向或负向运动，松开后运动停止。

确定关节旋转方向，可参照下图:



UFactory 850

1.5.3.1 手动模式



- 开启手动模式，可手动拖拽机器人关节到达目标位置点，使录制机器人运动轨迹更加方便，从而减轻开发工作量。当发生危险时，也可以运用手动模式手动拖拽机器人快速离开危险区域。

注意：（1）开启手动模式前，需确保机械臂的安装方式和机械臂的负载设置与实际符合，否则


会有危险。


- (2) 机械臂与控制器需一一对应，才能开启手动模式，可在【设置】-【我的设备】-【设备信息】查看控制器地址与对应的机械臂地址。
- (3) 机械臂的 SN 地址可在底座的电源信号接口旁查看。

1.5.4 轨迹录制



轨迹录制，可以记录在一段时间内对自由驱动机械臂时的运动轨迹，最长录制时间为 5 分钟。回放会完全重复录制时的运动轨迹，可设置轨迹回放的倍速（1 倍、2 倍、4 倍）。录制好的轨迹可导入 Blockly 的项目里。

 可新建录制文件。

 点击该按钮，示教模式相应开启，可直接拖拽机械臂进行轨迹录制。在开启录制时，务必要注意机械臂的负载状态，避免机械臂实际负载与设置的负载相差太大，导致机械臂自运动。

 显示录制时间。

 可停止录制。

【循环次数】 可设置回放次数。

【倍速】 可设置回放倍速。

我的项目

- ↓ 导出
🗑 删除
🚫 取消
☰ 取消全选

选择	序号	名字	时长
<input checked="" type="checkbox"/>	1	ll	00:02

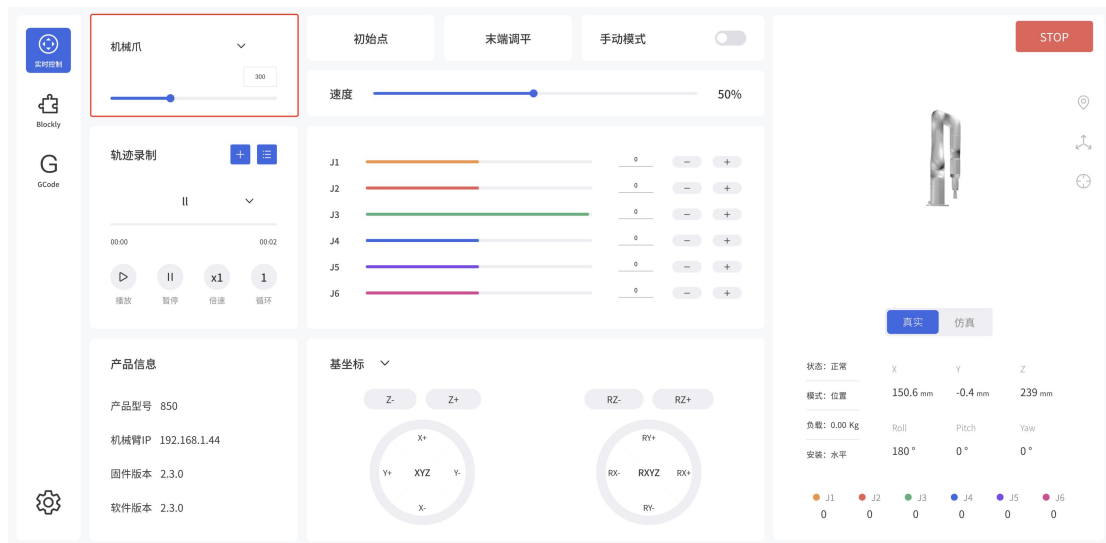
关闭

1.5.5 末端工具

UFactory 850 目前可支持连接的末端执行器有：UFactory 850 机械爪、UFactory 850 真空吸头、UFactory 850 BIO 机械爪、Robotiq-2F-85 机械爪、Robotiq-2F-140 机械爪。

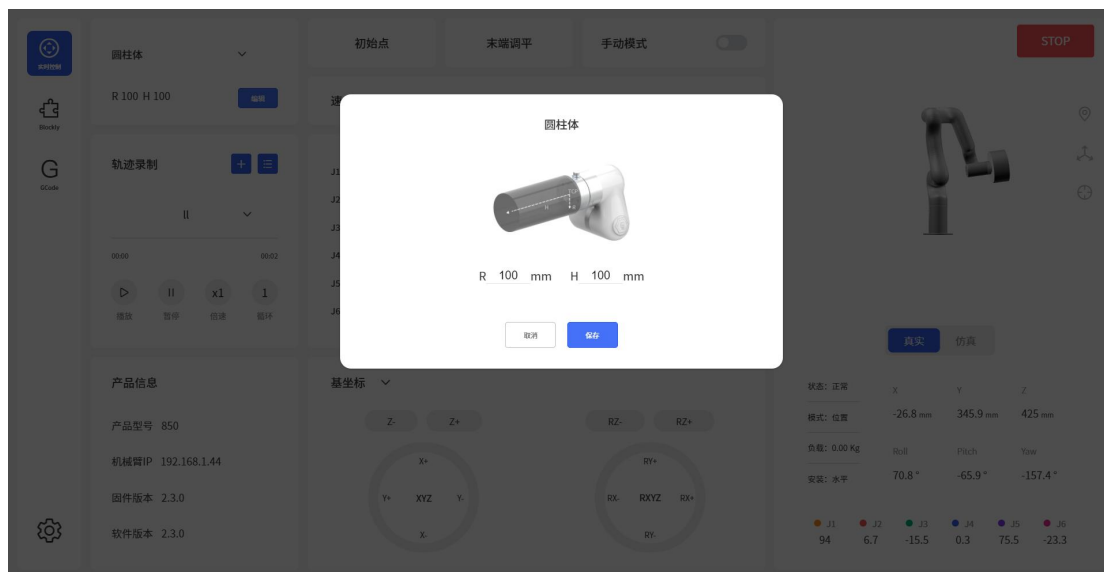
- 当机械臂末端已安装选项里所提供的末端执行器时，选择相应的末端执行器

以机械爪为例子，在 3D 显示界面，通过修改数值的大小或者拖动进度条，可控制机械爪的开合。



- 当机械臂末端安装的末端执行器不在可提供的选项里时，选择【其他】

可以选择一个可包裹住末端执行器的 3D 模型（圆柱体/长方体），将其作为末端执行器的防自碰撞模型



- 当机械臂末端无末端执行器时，选择【无末端执行器】

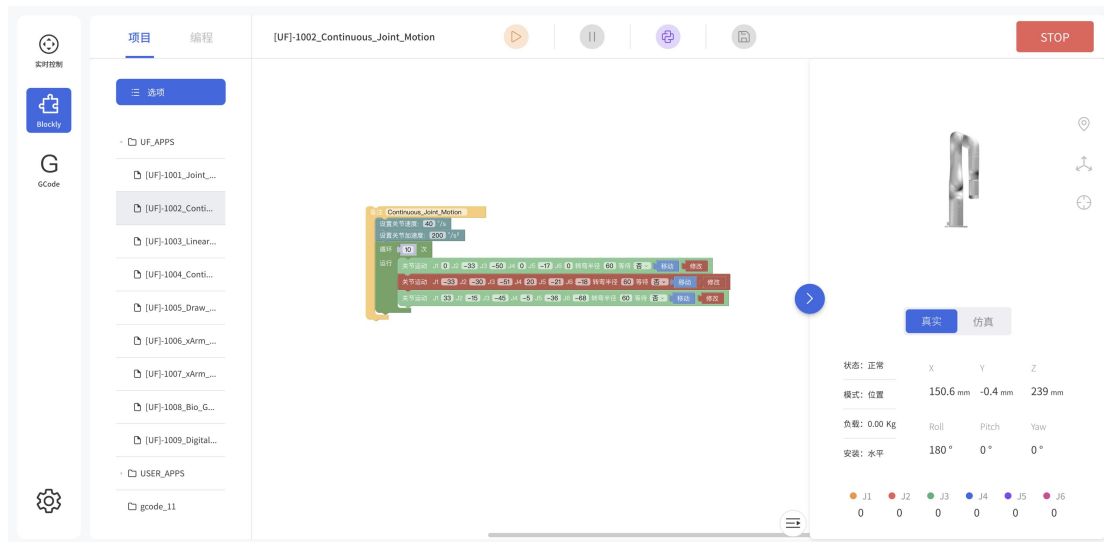


选择【无末端执行器】时，会将默认 TCP 负载设为 0。

1.6 Blockly 图形化编程

Blockly 是一个图形化编程工具，可以通过拖拽代码块的方式编程来控制机械臂，无需手动编写代码。


1.6.1 界面简介




【项目】查看所有的 Blockly 项目。

【编程】进入到编程页面，从中可以选取到你需要的代码块。

- 新建文件夹
- 新建文件
- 导入
- 下载
- 重命名
- 删除

【】鼠标右键单击 Blockly 文件名，可以新建文件夹，新建文件。下载，导入，重命名或者删除文件。

【】运行已经写好的 Blockly 程序。

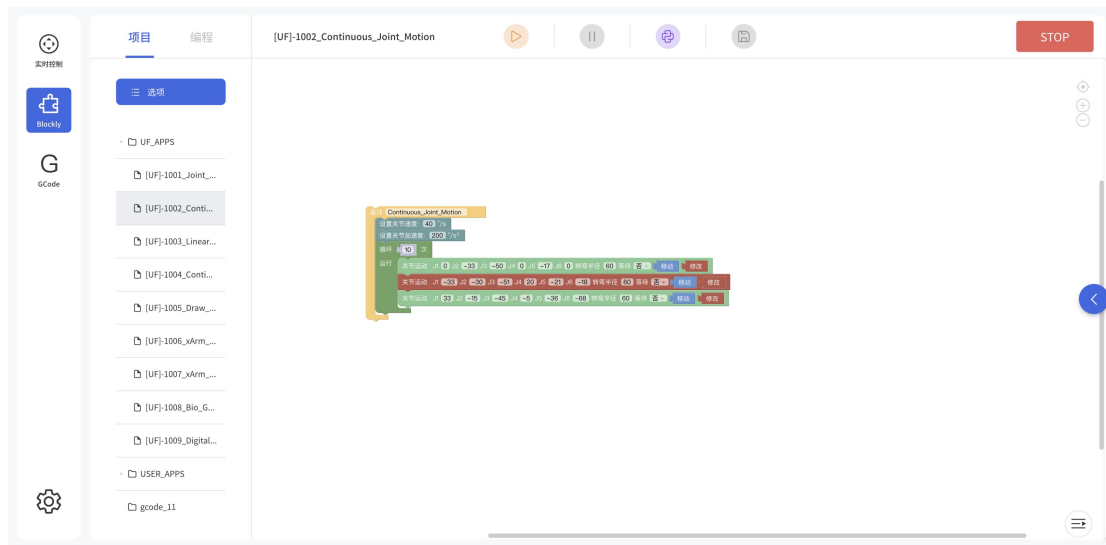
【】点击可将 Blockly 工程转化为 Python 代码。

【】保存对 Blockly 代码的更改。


【】可以收起或者展开 机械臂的 3D 页面和基础的位置信息

注：机械臂处于仿真模式时，也可以运行 Blockly 的运动程序来观察虚拟机械臂的运动。

1.6.2 Blockly 工作区



将代码块拖入到操作面板，代码执行的顺序是自上而下，可同时拖拽该代码块后面连着的代码。

【】回到默认大小和代码块居中位置。

【】放大代码块。

【】缩小代码块。

【】拖拽代码块到垃圾桶删除，或者可以选中代码块后按键盘【enter】/【delete】键。

1.6.2.1 工作区鼠标右键功能



在非代码块的空白工作区右键弹窗，功能主要针对所有的代码块：

【撤销】：撤销上一次操作。

【重做】：还原上一次撤销的操作。

【折叠命令】：折叠所有代码块。

【显示完整命令】：显示所有折叠的命令。


【删除 14 个命令】：删除所有的的代码块。

1.6.2.2 代码块鼠标右键功能



在代码块里单击鼠标右键，弹窗各模块功能：

【复制】：复制当前包裹的所有代码块。键盘复制/剪切快捷键可粘贴到其他文件里。（鼠标 Ctrl+c, Ctrl+x, Ctrl+v 可跨文件复制剪切粘贴指令）

【添加描述】：可针对该代码块添加描述，由符号标识 ，点击该符号可打开/关闭描述弹窗，如下图显示。



【外部输入】：设置文本框的位置在最右边显示。

【内部输入】：设置文本框的位置在默认中间位置显示。

【折叠命令】：折叠当前包裹的代码块。

【禁止运行命令】：禁止运行当前包裹的代码块的命令，与此相反的是【恢复运行命令】。

【删除 82 个命令】：删除当前鼠标单击选中的代码块。

【帮助】：相应代码块的‘帮助页面’跳转。

1.6.2.3 移动/Wait/修改的说明

这三个按钮为一些运动指令的公共功能按钮，点击【移动】机械臂会运动到当前位置点；点击【修改】，可弹出实时控制界面，在该界面可修改位置点；wait(true/false)，表示是否等待该命令执行完后再发送下一条命令。

1.6.3 Blockly 编程

设置：用于设置机械臂的运行速度、加速度、碰撞灵敏度、负载等。

运动：包含直线运动、关节运动、带圆弧的直线运动、停留时间、零点和紧急停止等常用运动指令。

应用：可导入 Blockly 其他项目。

GPIO：按键执行触发事件。

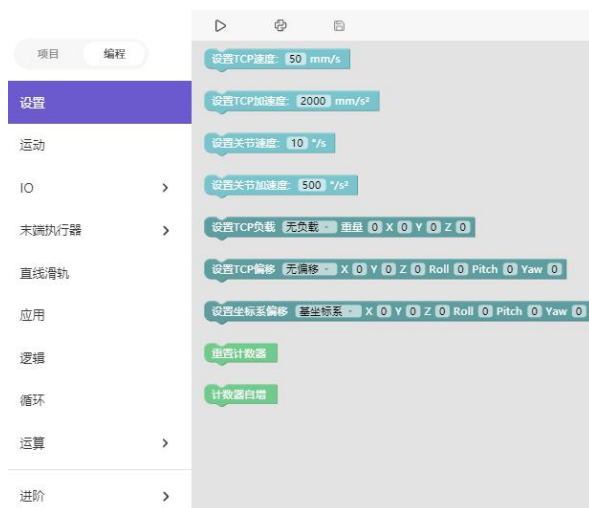
末端执行器：用于设置末端执行器。

逻辑：包含常用的逻辑指令。

循环：包含多次循环、无限循环、跳出循环等常见循环指令。

进阶：包含位置备注和消息提醒。

1.6.4 设置



【设置 TCP 速度 () mm/s】：设置直线运动的速度，单位是 mm/s。

【设置 TCP 加速度 () mm/s²】：设置直线运动的加速度，单位是 mm/s²。

【设置关节速度 () ° /s】：设置关节运动的速度，单位是° /s。

【设置关节加速度 () ° /s²】：设置关节运动的加速度，单位是° /s²。

代码块中默认的速度和加速度值是当前机械臂设置的速度和加速度值，可手动输入更改。

【设置负载 () 重量 () XYZ】：设置当前项目的负载，下拉列表引用设置-TCP 负载。

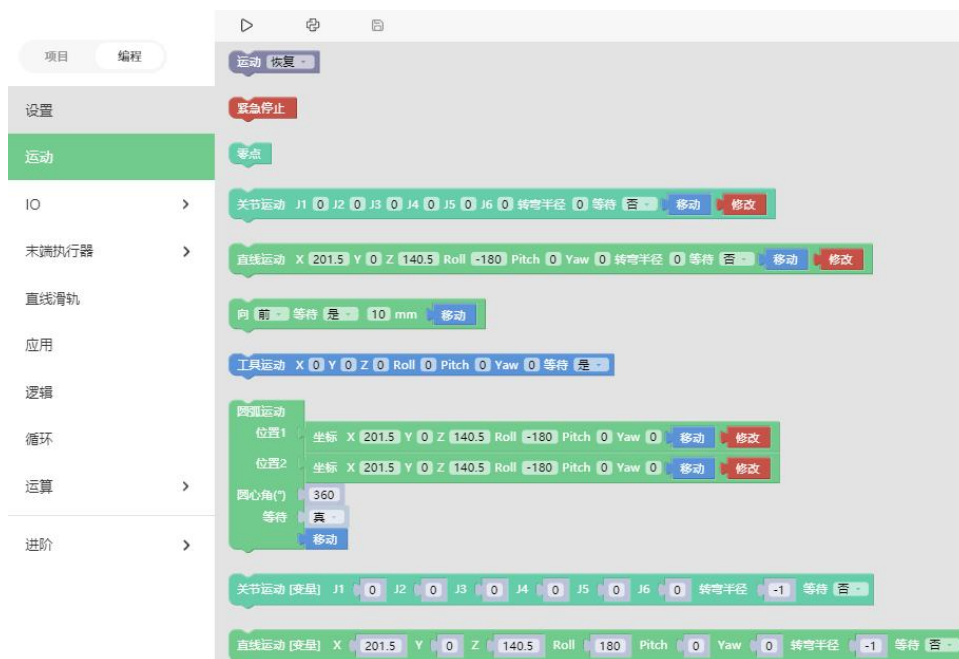
【设置末端偏移 () X Y Z Roll Pitch Yaw】：设置当前项目的末端偏移，下拉列表引用设置-TCP 偏移。

【设置基坐标偏移 () X Y Z Roll Pitch Yaw】：设置当前项目的基坐标偏移，下拉列表引用设置-基坐标偏移数据。

【重置计数器】：该指令将控制器内的计数器重置为 0。

【计数器自增】：每次执行到该指令时会使控制器的计数器自增 1。可用于计算机械臂实际循环执行的次数。

1.6.5 运动



【停留 () 秒】：收到该命令后，机械臂会停止运动，持续时间为所设置的时间，然后再继续执行后面的命令。主要用于需要做连续运动的运动程序中，放在连续运动指令之前，用来缓存更多的运动指令。

【运动 ()】：通过该命令，可设置机械臂的状态（运动，暂停，停止），用于控制机械臂的状态，主要用于条件触发的程序中。

【紧急停止】：机械臂立即停止运动，且清空命令缓存。

【零点】：机械臂回到关节值都为 0 的姿态。

【关节运动 J1() J2() J3 () J4() J5() J6() J7() Radius() Wait(true/false) [move] [edit]】：设置关节运动的各个关节值，单位是°。在真实机械臂模式下，从侧边拖出的关节运动指令数据是真实机械臂当前姿态数据。

【笛卡尔运动 X() Y()Z() Roll() Pitch() Yaw() Radius() Wait(true/false) [move] [edit]】：表示直线运动的笛卡尔坐标值与 TCP 旋转角度，单位是毫米（mm）和度（°）。

【向（前/后/左/右）（true/false）() mm】：表示机械臂基于当前位置向前/后/左/右做相对直线运动，单位是毫米（mm）。

【工具运动 X() Y()Z() Roll() Pitch() Yaw() Radius() Wait(true/false)】：该指令是相对于 TCP 坐标走的相对运动。

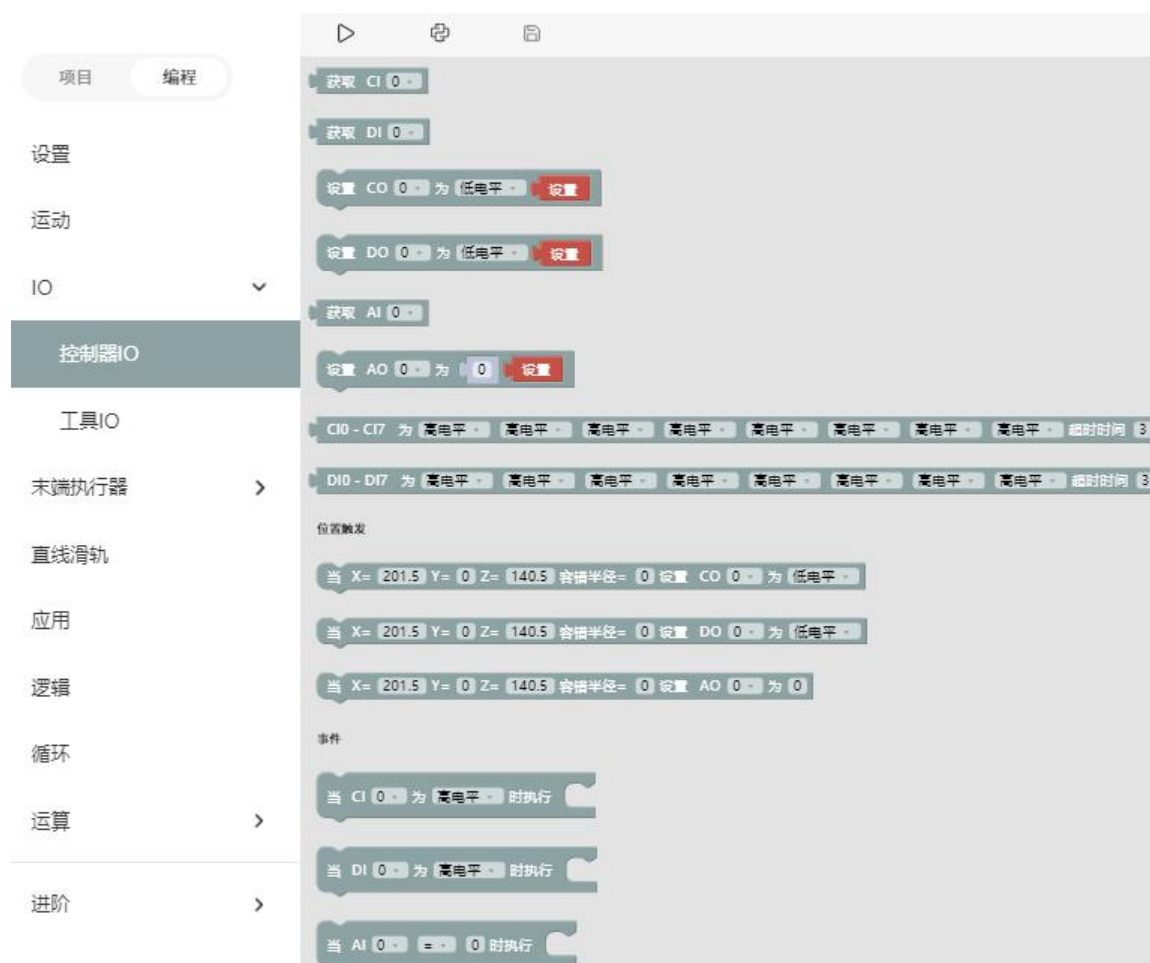
【圆弧运动 从位置 1 到位置 2】：表示位置 1 到位置 2 走圆弧运动，但该命令前面还需再加入一个位置点，三点确定一个圆；单位是毫米（mm）。

【圆心角（°）（）】：表示该圆弧的度数，设置 360° 时可以走完一整个圆弧，可大于小于 360°。（注意：要想实现平滑的轨迹运动，需要设置 Wait = false）

【关节运动[变量] J1() J2() J3 () J4() J5() J6() J7() Wait(true/false)】：该指令走关节运动，支持变量值。

【笛卡尔运动[变量] X() Y()Z() Roll() Pitch() Yaw() Radius() Wait(true/false)】：该指令走笛卡尔运动，支持变量值

1.6.6 IO（控制器、末端执行器接口）





I/O 接口分为控制器接口和末端执行器接口，均可做获取，设置，监听 I/O 接口操作。控制器有 8 个数字输入接口，8 个数字输出接口，2 个模拟输入接口，2 个模拟输出接口。末端执行器有 2 个数字输入接口，2 个数字输出接口。2 个模拟输入接口。控制器数字 I/O 为低电平时触发。末端执行器数字 I/O 为高电平时触发。

【获取 I/O】：获取代码块的 I/O 接口数据。

【设置 I/O】：设置代码块的 I/O 接口，点击【设置】可执行该条指令。

【设置 I/O, when (X, Y, Z, 容错半径)】

- 当机械臂到达指定位置时（以触发位置点 (X, Y, Z) 为中心指定的球面体区域（球面体半径为容错半径）），触发 I/O。该指令可用于在特定位置触发 I/O 的场景。
- X, Y, Z 代表机械臂要到达的指定位置的坐标值，单位毫米 (mm)。
- 容错半径是指，TCP 位置触发的数字 I/O 操作中，以触发位置点为中心指定一个球面体区域，只要系统检测 TCP 位置进入这个附近区域时即触发 I/O。如果不设置容错半径，则机械臂以非 0 的速度通过指定点时可能因无法准确检测到而导致漏触发。

【当数字 I/O 为(高电平/低电平)时执行】：当条件满足时执行该代码块包含的指令。

【当模拟 I/O 数值满足设置的条件时执行】：当监听的模拟 I/O 数值满足条件时执行代码块包含的指令，条件有 =, ≠, >, ≥, <, ≤。

UFactory studio 的 IO 触发逻辑：

1. 每 100ms 获取一次 IO 状态，将第一次获取到的状态作为初始值。
2. 从第二次获取到的 IO 状态与上一次获取的 IO 状态比较，有变化时触发满足条件的回调。

1.6.7 末端执行器



【设置 UFactory 850 机械爪 位置() 速度() Wait (true/false) [移动] [修改]】：设置 UFactory 850 机械爪的位置和张合速度。

【设置 Bio 机械爪 打开() 速度() Wait (true/false) [设置]】：设置 Bio 机械爪的开合和开合速度。

【设置 Robotiq 机械爪 位置() 速度() 力度() Wait (true/false) [设置]】：设置 Robotiq 机械爪的位置和张合速度以及抓握力度。

【初始化机械爪】：使能机械爪。

【获取真空吸头状态】：可获取真空吸头是否吸住物体的状态，真空吸头状态等于 1 时，表示

成功吸住物体，当真空吸头态为 0 时，表示没有吸住物体。

【检测真空吸头已经（吸住/释放）】：检测真空吸头是否吸住（释放）物体，若检测到真空吸头已吸住（释放）物体，则跳出这条指令，执行下一个指令，如果超过了超时时间后，真空吸头还未吸住（释放）物体，则也会跳出该指令，执行下一个指令。

【设置真空吸头（开/关） [设置]】：可设置真空吸头开启和关闭状态。

【是否拾取检测】=true，同时检测了是否吸住物体，如果没有吸住物体，将会跳出整个程序；

【是否拾取检测】=false，不检测是否吸住物体。

1.6.8 应用



【运行轨迹（路径）次数[1]】：可导入轨迹录制文件，并设置运行次数。

【导入其他 APP】：可导入其他项目的 Blockly。

1.6.9 逻辑



【等待 ()】：等待发送后面的指令，单位是秒 (s)。

【如果(条件 1) 运行 (指令 1)】：如果条件 1 成立，则会运行指令 1，否则就跳过。

如果/否则语句的设置：




1. 点击命令块上的设置按钮，然后命令块会弹出一个选择框，如下图：



2. 此时将上图中的【否则】代码块拖到右边【如果】代码块的下方，使两个代码块结合，如下图：



3. 此时点击设置按钮, 选择框收回，如果/否则语句设置完成，如下图：



1.6.10 循环



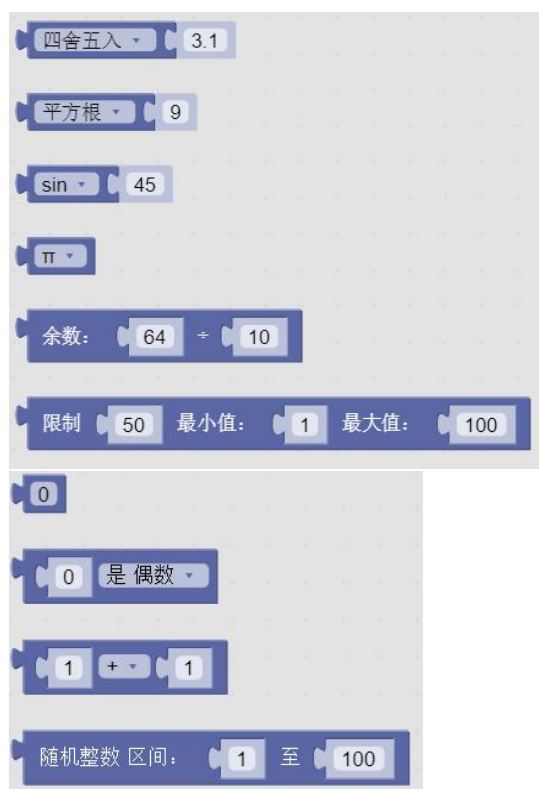
【一直循环】：里面包含的指令无限循环。

【循环() 运行】：循环（？）次里面包含的指令。

【循环直到】：当条件不成立时，跳出循环。

【跳出循环】：终止循环。

1.6.11 运算



可使用上面的代码块做加减乘除，指数运算等一些复杂的运算。

1.6.12 文字



【备注】：备注该代码块，起指示作用，可改变颜色。

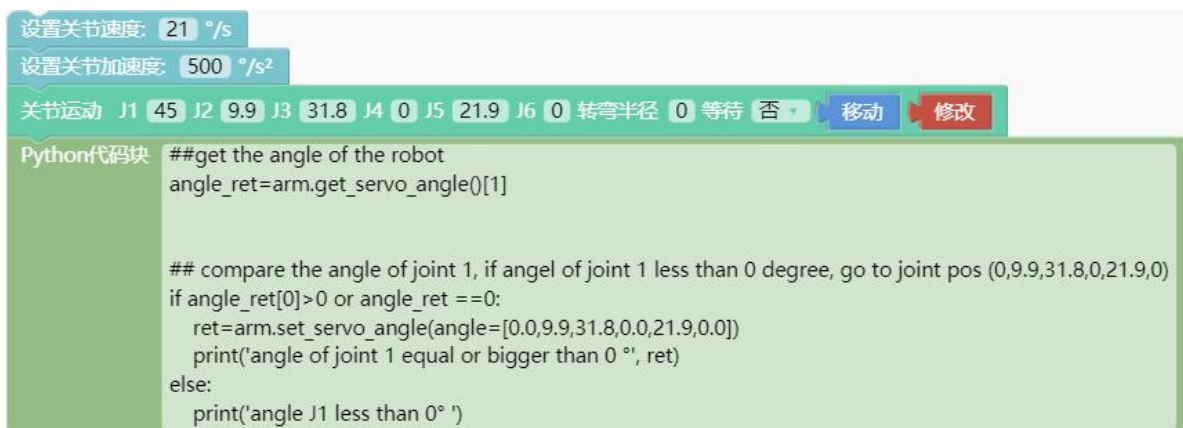
【消息提示】：类型（信息/成功/警告/错误），持续时间表示消息显示的时间，单位是秒（s）；message 表示提示的消息内容。

【字符串打印】：可在下方打印输入的字符串，并设置字体颜色。

【变量打印】：可打印添加的变量，并设置字体颜色。

【日期】：可输出运行该指令时的日期。

【Python Code】可以自定义编写 python 代码,使其变成 Blockly 中的一个块,并在程序中使用它



1.6.13 变量



【新建变量】：可新增变量。新增后变量默认有三条指令（设置变量值，通过加减操作改变变量值，变量）

【重命名变量】：可重新命名该变量。

【删除变量】：可删除变量。

1.6.14 函数



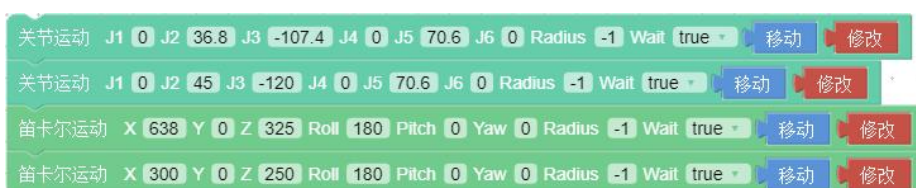
【定义函数（函数名）】：可定义新函数，不带返回值。

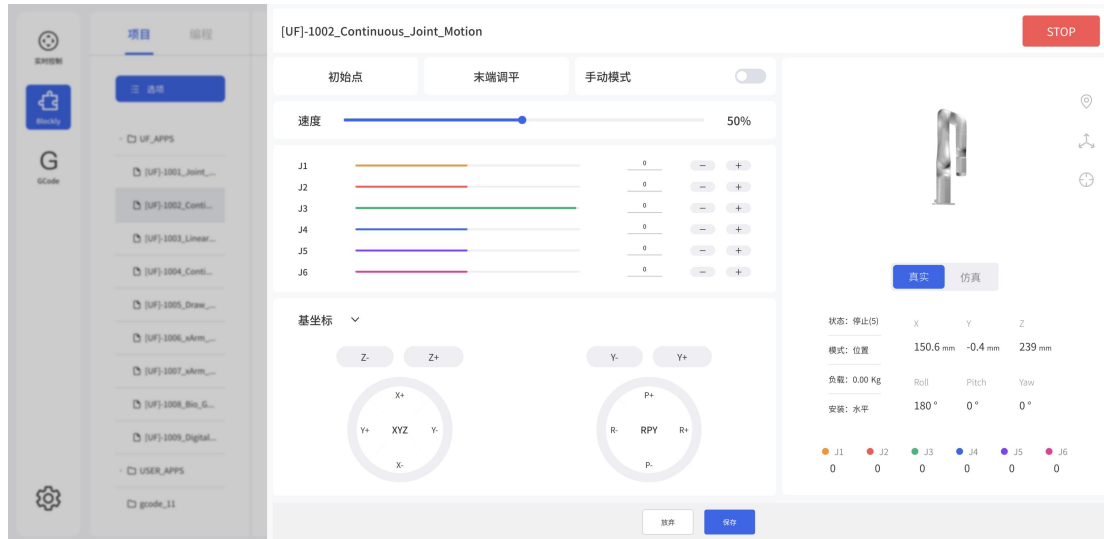
【定义函数（函数名）输出[]】：可定义新函数，带返回值。

【如果[]输出[]】：条件判断句，只能放在内置函数中。

注意：定义的函数要放在主程序的前面。

1.6.15 移动，修改运动指令坐标





点击【修改】，弹出实时控制界面，可重新编辑当前指令的运动坐标。



首先要先长按【移动】按钮，使机械臂移动到当前指令的位置点。

点击【保存】，保存更改，关闭弹窗；点击【放弃】，放弃更改，关闭弹窗。

注意：


1. 在一段指令里面，有顺序 A/B/C/D... 等多个点，如果点击【移动】让机械臂从 A 点位置跳过 B 点直接到 C 点或者 C 点以后的位置。为避免损害周边设施，必须要进行安全评估。
2. 因为笛卡尔指令的复杂性，笛卡尔空间轨迹规划，需要做逆运动学求解。因而可能存在无解、多解、逼近解的情况。从 A 点到 B 点走笛卡尔指令时的解不理想时，必要时在两点之间插入第三个关节指令。

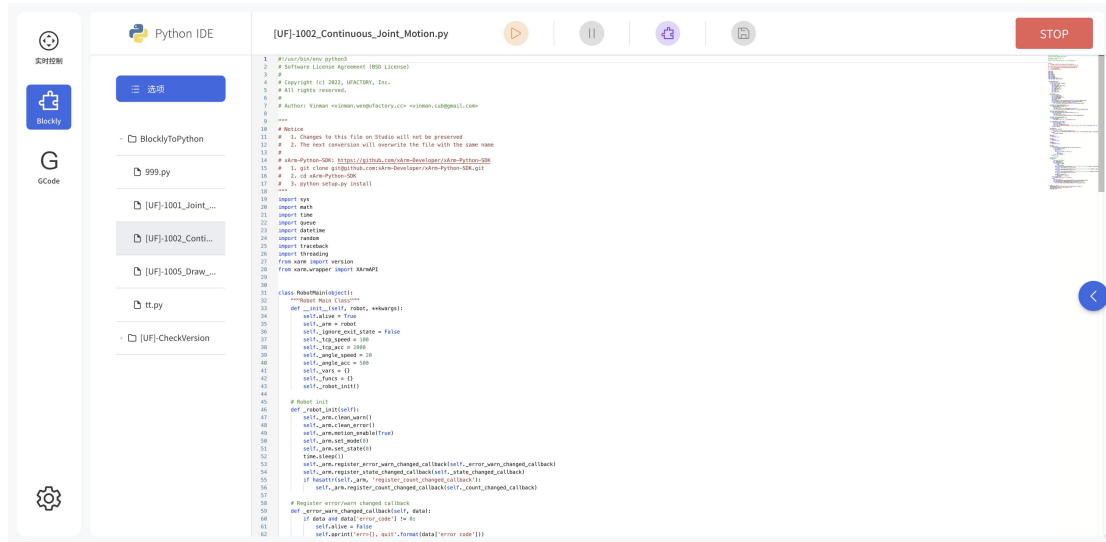
1.6.16 路径规划注意事项

 <p style="text-align: center;">危险</p>	<p>1. 如果机械臂运动过程中发生碰撞导致停机，此时机械臂会报错，需要清除错误后，才能正常使用。再次运动前请务必做安全评估，防止发生碰撞。</p>
 <p style="text-align: center;">注意</p>	<p>1. 机械臂在某些位置，可能存在直线运动无解的情况，需重新规划路线。详情请参考“UFactory 850 运动学特性-直线运动”。</p>

1.7 Python IDE



点击 Blockly 页面的小图标【】进入到 Python IDE，Python IDE 是 Python 开发集成环境，可直接使用 UFactory 850-Python-SDK API 和查看 Blockly 项目编译为的 Python 代码。




新建文件夹

新建文件


导入

下载

重命名

【】鼠标右键单击 Python 文件名，可以新建文件夹，新建文件，下载，导入，重命名或者删除文件。



【】运行已经写好的 Python 程序。

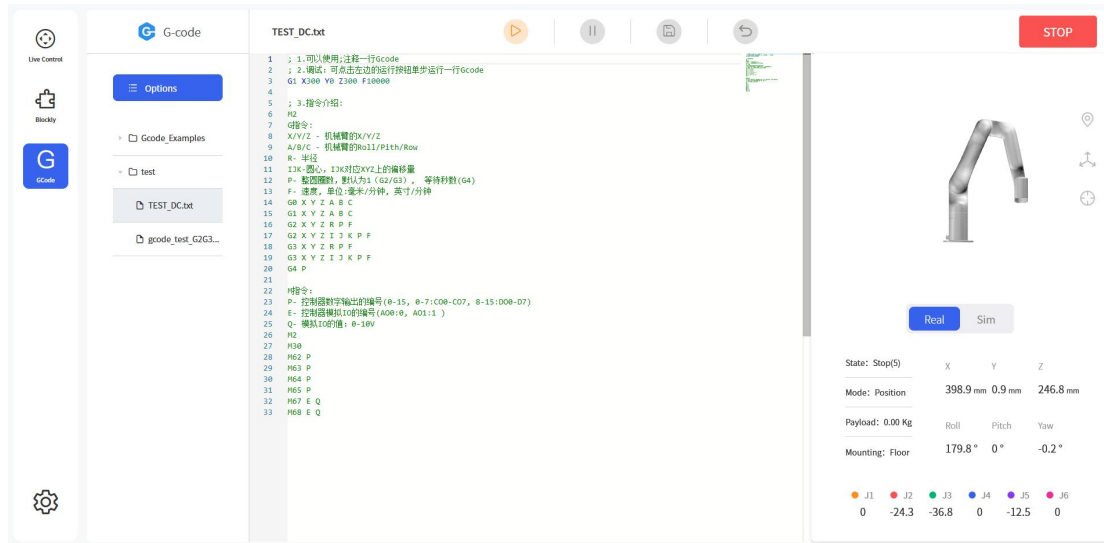


【】保存对 Python 代码的更改。

控制器指令缓存机制

当前控制器可缓存 2048 条指令，当控制器缓存指令达到 2048 条的时候会先等待控制器先执行当前指令，再继续发送指令。当控制器等待执行的指令超过最大缓存量，即 2048 条，如果继续发送指令会返回警告，警告码为十进制 11，并丢弃这条指令。下发的指令，控制在 256 条指令缓存之内是比较合适的。

1.8 GCode 编程页面



通过 GCode 编程控制手臂运动，下面是 UFACTORY Studio 支持的指令介绍：

指令兼容 Linux CNC

TCP 端口：504

支持指令

G 指令

G0 X Y Z A B C # 固定速度为 240mm/s

G1 X Y Z A B C F # XYZ 单位 mm，ABC 对应 roll/pitch/yaw（单位°），F 为速度（单位 mm/min 或英寸/min，缺省为上一次 G1 指令的速度或 100mm/s）（当有 XYZABC 参数时，G1 也可以省略）

G2 X Y Z R P F # （顺时针）半径格式圆，XYZ 表示目标位置，R 表示圆半径，P 表示整圆圈数，F 表示速度

G2 X Y Z I J K P F # （顺时针）中心格式圆弧，XYZ 表示目标位置，IJK 对应圆心，P 表示整圆圈数，F 表示速度

G3 X Y Z R P F # （逆时针）半径格式圆，XYZ 表示目标位置，R 表示圆半径，P 表示整圆圈数，F 表示速度

G3 X Y Z I J K P F # （逆时针）中心格式圆弧，XYZ 表示目标位置，IJK 对应圆心，P 表示整圆圈数，F 表示速度

G4 P # 休眠指令，P 为要休眠的秒数

- G17** # 圆弧指令设置 XY 平面, G2/G3 默认使用该平面
- G18** # 圆弧指令设置 XZ 平面
- G19** # 圆弧指令设置 YZ 平面
- G20** # 单位为英寸, 影响 G0/G1
- G21** # 单位为毫米, 影响 G0/G1
- G90** # 绝对定位, 影响 G0/G1/G2/G3, G0/G1 默认使用该模式, G2/G3 的 XYZ 参数默认使用该模式
- G90.1** # 圆弧指令使用绝对定位, 影响 G2/G3
- G91** # 相对定位, 影响 G0/G1/G2/G3
- G91.1** # 圆弧指令使用相对定位, 影响 G2/G3, G2/G3 的 IJK 参数默认使用该模式

M 指令

- M2/M30** # 停止程序(这个需要客户端控制该指令之后的指令不下发), 固件收到该指令仅重置参数(单位使用毫米(G21), 绝对定位(G90), 圆弧平面为 XY(G17), G1 的默认速度为 100mm/s)
- M62 P** # (设置 CGPIO 数字 IO 输出高电平, 队列执行), P 为 IONUM
- M63 P** # (设置 CGPIO 数字 IO 输出低电平, 队列执行), P 为 IONUM
- M64 P** # (设置 CGPIO 数字 IO 输出高电平, 立即执行), P 为 IONUM
- M65 P** # (设置 CGPIO 数字 IO 输出低电平, 立即执行), P 为 IONUM
- M67 E Q** # (设置 CGPIO 模拟 IO 输出, 队列执行), E 为 IONUM, Q 为要设置的值
- M68 E Q** # (设置 CGPIO 模拟 IO 输出, 立即执行), E 为 IONUM, Q 为要设置的值

注意

1. 固件版本需要高于 v2.1.102
2. 端口暂时使用 504
3. 回复数据(5 字节)

byte0: Gcode 解析返回值(非 0 表示格式不对或命令不支持)

0: 成功

- 1: 解析错误（一般是 Gcode 不规范）
- 2: 暂不支持的 Gcode 指令
- 11: 参数缺失（一般是缺少一些必须参数）
- 12: 参数类型错误（一般是需要整型却给了个浮点）
- 13: 参数值错误（一般是参数值不在范围内）
- 19: 其它参数错误（暂时未用到）

byte1: 高 4 位表示模式，低 4 位表示状态

byte2: 控制器错误码，Cxx

byte3: 和 byte4: 缓存数(大端字节序)

4. 建议每次发一行非空数据（带换行符），固件按行回复的 `sock.send(b'G0 X300\n')`
5. 要接收回复，不然久了缓冲区会满 `sock.recv(5)`
6. 最多同时缓存 2000 条队列指令（G0/G1/G2/G3/G4/M62/M63/M67），一旦缓存满了，处理进程会阻塞直到缓存指令被消耗至少 1 条，阻塞过程无法处理任何 Gcode 指令。

Python 示例:

```
import socket
sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
sock.setblocking(True)
sock.connect(('192.168.1.67', 504))
def send_and_recv(data):
    for line in data.split('\n'):
        line = line.strip()
        if not line:
            continue
        sock.send(line.encode('utf-8', 'replace') + b'\n')
        ret = sock.recv(5)
        code, mode_state, err = ret[0:3]
        state, mode = mode_state & 0x0F, mode_state >> 4
        cmdnum = ret[3] << 8 | ret[4]
        if code != 0 or state >= 4 or err > 0:
            print('code: {}, mode: {}, state: {}, err: {}, cmdnum: {}, cmd:
{}'.format(code, mode, state, err, cmdnum, line))
# move x to x=500mm, speed= 10000 mm/min
send_and_recv('G1 X500 F10000')
```

2. UFactory 850 运动学特性

这一章节我们主要用 Python/Blockly 示例对下面列表中的几个典型运动进行讲解。

- 关节运动
- 直线运动
- 圆弧直线运动
- 圆及圆弧运动

关于 Python-SDK:

所有带有 `is_radian` 的接口, `is_radian` 的默认值为实例化时的 `is_radian`。

所谓实例化就是在创建 `UFactory 850API()` 时, 所设置的参数 `is_radian` 的值, 下面举三个例子进行说明:

```
1. arm = xArmAPI('192.168.1.226',)
2. arm = xArmAPI('192.168.1.226', is_radian=False)
3. arm = xArmAPI('192.168.1.226', is_radian=True)
```

以方式 1 创建 `xArmAPI()` 接口时, `is_radian` 默认值为 `False`, 此时使用度 ($^{\circ}$) 为单位;

以方式 2 创建 `xArmAPI()` 接口时, 指定 `is_radian=False`, 此时使用度 ($^{\circ}$) 为单位;

以方式 3 创建 `xArmAPI()` 接口时, 指定 `is_radian=True`, 此时使用弧度 (`rad`) 为单位;

关于 Blockly:

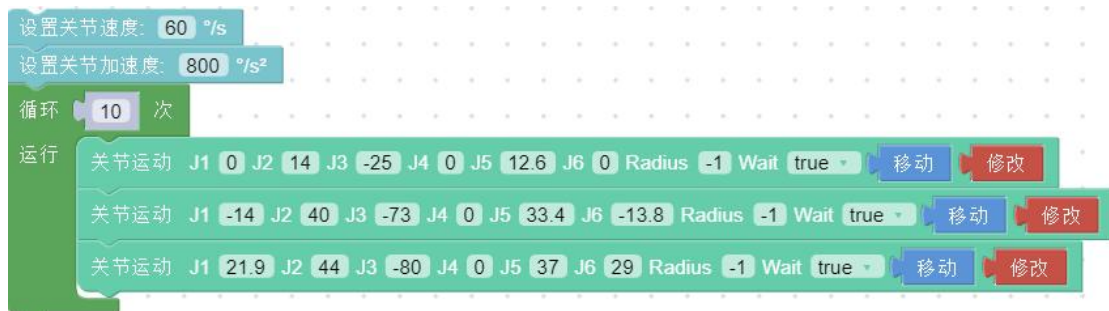
所有关于角度的单位都使用度 ($^{\circ}$)。

2.1 机械臂的运动方式

2.1.1 关节运动

实现关节空间的点对点运动（单位：度），每个指令之间速度不连续。

Blockly 示例：



【设置关节速度 () ° /s】：设置关节运动的速度，单位是 ° /s。

【设置关节加速度 () ° /s²】：设置关节运动的加速度，单位是 ° /s²。

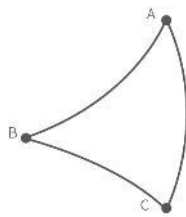
【J1(), J2(), J3 (), J4(), J5(), J6(), J7(), Radius()】：设置关节运动的各个关节值，单位是 ° 。

【Wait(true/false)】：表示是否等待该命令执行完后再发送下一条命令。

【移动】：机械臂会运动到当前位置点。

【修改】：打开实时控制界面，调节当前点的坐标。

上例中机械臂的运动轨迹如下：



Python 示例：

```
arm.set_servo_angle(angle=[0.0, 7.0, -71.2, 0.0, 0.0, 0.0], speed=8, mvacc=1145, wait=True)
arm.set_servo_angle(angle=[0.0, 7.0, -51.2, 0.0, 0.0, 0.0], speed=8, mvacc=1145, wait=True)
arm.set_servo_angle(angle=[0.0, 7.0, -91.2, 0.0, 0.0, 0.0], speed=8, mvacc=1145, wait=True)
```

接口 set_servo_angle 说明，见表 2.1：

表 2.1 set_servo_angle 说明

接口 set_servo_angle 说明		
说明	设置关节运动的关节角度	
参数	servo_id	关节 ID, 1-7, None 或 8 表示所有关节: a) 1-(Number of axes) 机械臂的关节号 ex: arm.set_servo_angle(servo_id=1, angle=45, is_radian=False) b) None(8) 代表所有关节 ex: arm.set_servo_angle(angle=[30, -45, 0, 0, 0, 0, 0], is_radian=False)
	angle	关节角度或关节角度列表 (默认关节角度的单位为 is_radian=False, 即度 (°)) a) 若 servo_id is 1- (关节序号) ex: arm.set_servo_angle(servo_id=1, angle=45, is_radian=False) b) 若 servo_id is None or 8, ex: arm.set_servo_angle(angle=[30, 45, 0, 0, 0, 0, 0], is_radian=False)
	speed	关节速度 (默认单位为 °/s): 单位: 若 is_radian=True, 则单位为 rad/s; 若 is_radian=False, 则单位为 °/s;
	mvacc	关节加速度 (默认单位为 °/s ²) 单位: 若 is_radian=True, 则单位为 rad/s ² ; 若 is_radian=False, 则单位为 °/s ² ;
	is_radian	roll/pitch/yaw 是否为弧度 (默认 is_radian=False) 若 is_radian=True, 则 roll/pitch/yaw 单位为弧度 (rad); 若 is_radian=False, 则 roll/pitch/yaw 单位为度 (°);
	wait	若 wait=True, 为等待当前指令发完再发下一条指令; 若 wait=False, 不等待, 直接发下一条指令;
	mvtime	0, 运动时间, 暂无意义;

注: (1) 若关节角度要设置为弧度, 则 is_radian=True;

 ex: code = arm.set_servo_angle(servo_id=1, angle=1.57, is_radian=True)

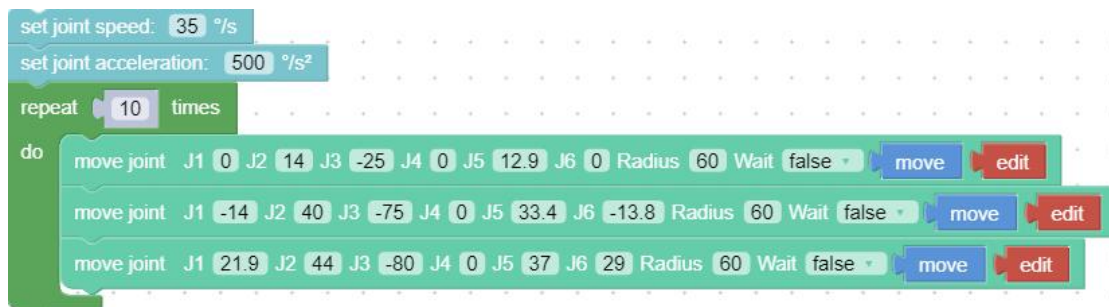
(2) 若要等待机械臂完成当前的指令再返回, 则 wait=True;

 ex: code = arm.set_servo_angle(servo_id=1, angle=45, is_radian=False, wait=True)

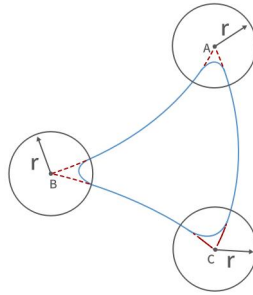
如何规划连续的关节运动

两条关节运动指令之间插入圆弧过渡, 是规划机械臂进行连续的关节运动的一种方式。

Blockly:



上例中机械臂的运动轨迹如下:



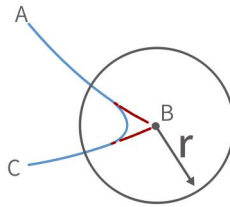
关键参数说明

Radius=60

“move joint” 指令中的 Radius=60，指设置的过渡圆弧的半径 $R=60\text{mm}$ ，用于实现关节运动转弯的平滑过渡。

Radius 的参数可设置为 $\text{Radius}>0$, $\text{Radius}=0$, $\text{Radius}=-1$ ，不同参数对应不同的轨迹。

(1) $\text{Radius}>0$ 。例如设置 $\text{Radius}=60$ ，转弯的轨迹如下图蓝色圆弧，可以实现平滑转弯效果。

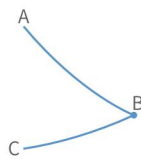


注：

1) 关节速度在交融区域是平滑过渡的，交融部分执行的时间是不交融两段轨迹时间的一半，所以如果关节运动交融半径比较小，速度提升感不会十分明显。

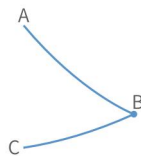
2) 设置的 Radius 的值应该小于 D_{AB} 和 D_{BC} （直线距离）。

(2) $\text{Radius}=0$ 。转弯处不设置圆弧过渡，尖角转弯，如下图所示。



(3) $\text{Radius}<0$ 。转弯处不设置圆弧过渡，尖角转弯且运动轨迹不连续，如下图所示。

注意： $\text{Radius}<0$ 不能实现连续运动，如果需要规划机械臂连续运动，请确保 $\text{Radius}\geq 0$ 。



Wait=false

“move joint” 指令中的 wait 表示是否需要等待本条指令执行完再发送下一条指令。

如果需要规划连续运动，请确保 $\text{wait}=\text{false}$ 。

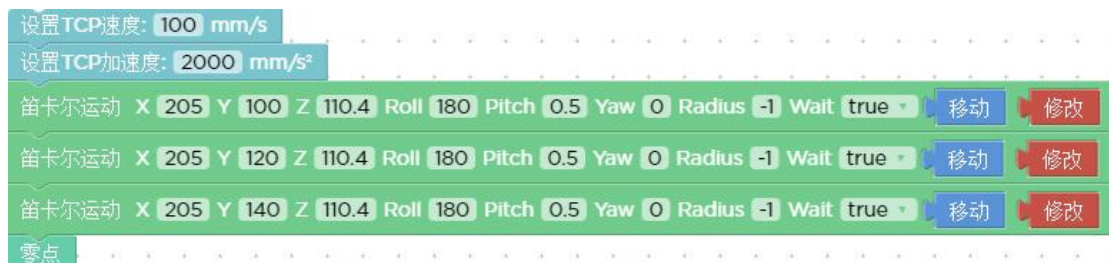
2.1.2 直线运动及圆弧直线运动

2.1.2.1 直线运动

直线运动的概念

- 实现笛卡尔坐标之间的直线运动（单位：mm），每个指令之间速度不连续。
- 用户可以基于基坐标系、TCP 坐标系来控制机器人运动状态。笛卡尔空间运动的机器人使工具在路点之间进行线性移动。这意味着每个关节都会执行更为复杂的移动，以使工具保持在直线路径上。目标点确认后，末端路径是唯一的，轨迹过程中对应的姿态是随机的。基坐标系的 X、Y 和 Z 控制工具的位置，箭头指向为+方向，单位是 mm。TCP 坐标系的 Roll /Pitch/Yaw 控制工具的方向，单位是度。
- 直线运动和圆弧直线运动属于笛卡尔空间轨迹规划，需要做逆运动学求解。因而可能存在无解、多解、逼近解的情况；而且由于关节空间和笛卡尔空间的非线性关系，可能会出现轴运动超出其最大速度和加速度限制的情况。

Blockly 示例：



【设置 TCP 运动速度 () mm/s】：设置直线运动的速度，单位是 mm/s。

【设置 TCP 运动加速度 () mm/s²】：设置直线运动的加速度，单位是 mm/s²。

【笛卡尔运动 X() Y()Z() Roll() Pitch() Yaw() Radius() Wait(true/false) [move] [edit]】：表示直线运动的笛卡尔坐标值与 TCP 旋转角度，单位是 mm 和°。

注：笛卡尔运动即 TCP 运动。

Python 示例：

```
arm.set_tcp_jerk(2000)

arm.set_position(x=205.0, y=100.0, z=110.4, roll=180.0, pitch=0.5, yaw=0.0, speed=100, radius=-1.0, wait=True)
```

```
arm.set_position(x=205.0, y=120.0, z=110.4, roll=180.0, pitch=0.5, yaw=0.0, speed=100, radius=-1.0, wait=True)
arm.set_position(x=205.0, y=140.0, z=110.4, roll=180.0, pitch=0.5, yaw=0.0, speed=100, radius=-1.0, wait=True)

arm.reset()
```

接口 set_position 说明，见表 2.2:

表 2.2 set_position 说明

接口 set_position 说明		
说明	设置直线运动的笛卡尔坐标值	
参数	X	坐标 X, (单位: mm)
	Y	坐标 Y, (单位: mm)
	Z	坐标 Z, (单位: mm)
	Roll	姿态 roll (默认单位为度) 单位: 若 is_radian=True, 则单位为 rad; 若 is_radian=False, 则单位为° ;
	Pitch	姿态 pitch (默认单位为度): 单位: 若 is_radian=True, 则单位为 rad; 若 is_radian=False, 则单位为° ;
	Yaw	姿态 yaw (默认单位为度) 单位: 若 is_radian=True, 则单位为 rad; 若 is_radian=False, 则单位为°);
	radius	若为直线运动, 则 radius<0/radius=None; 若为圆弧直线运动, 则 radius>0;
	is_radian	若 is_radian=True, 则 roll/pitch/yaw 单位为 rad; 若 is_radian=False, 则 roll/pitch/yaw 单位为° ;
	speed	TCP 运动速度 (mm/s, rad/s);
	mvacc	TCP 运动加速度 (mm/s ² , rad/s ²);
	mvertime	0, 运动时间, 暂无意义;
	relative	若 relative=True, 则为相对运动; 若 relative=False, 则不为相对运动;
	wait	若 wait=True, 为等待当前指令发完再发下一条指令; 若 wait=False, 则直接发下一条指令;

注: 若为 UFactory 8505, 则 roll 和 pitch 必须设置为 roll=±180°, pitch=0°。

2.1.2.2 圆弧直线运动

圆弧直线运动 (Lineb), 在两条直线之间插入圆弧过渡, 是规划机械臂进行连续运动的一种方式。下图是一个简单的利用圆弧直线运动规划机械臂进行连续运动的示例。

注意: 当固件版本 ≥1.6.0 时, 若需要规划 Lineb 运动, 则需要将 TCP 速度调至 200mm/s 以下进行调试, 否则会有很高的安全风险。

Blockly:

```

set TCP speed: 200 mm/s
set TCP acceleration: 2000 mm/s2
repeat 10 times
do
  move (arc) line X 300 Y -100 Z 100 Roll 180 Pitch 0 Yaw 0 Radius 5 Wait false
  move (arc) line X 300 Y -100 Z 300 Roll 180 Pitch 0 Yaw 0 Radius 5 Wait false
  move (arc) line X 300 Y 100 Z 300 Roll 180 Pitch 0 Yaw 0 Radius 5 Wait false
  move (arc) line X 300 Y 100 Z 100 Roll 180 Pitch 0 Yaw 0 Radius 5 Wait false
  
```

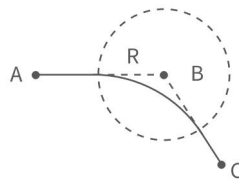
关键参数说明

Radius=5

“move (arc) line” 指令中的 Radius=5，指设置两条直线间的过渡圆弧的半径 R= 5mm，用于实现圆弧直线运动转弯的平滑过渡。

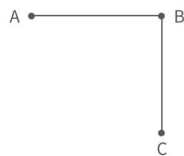
Radius 的参数可设置为 Radius>0, Radius=0, Radius=-1, 不同参数对应不同的轨迹。

(1) Radius>0。例如设置 Radius=5，转弯的轨迹如下图黑色圆弧，可以实现平滑转弯效果。



注意：设置的 Radius 的值应该小于 D_{AB} 和 D_{BC} 。

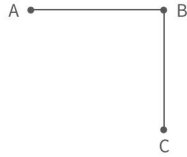
(2) Radius=0。转弯处不设置圆弧过渡，直角转弯，如下图所示。



注：如果机械臂的运动为往返的直线运动，则需设置 radius=0，若 radius>0 可能会导致机械臂报运动规划错误。

(3) Radius<0。转弯处不设置圆弧过渡，直角转弯且运动轨迹不连续，如下图所示。

注意：Radius<0 不能实现连续运动，如果需要规划机械臂连续运动，请确保 Radius≥0。



Wait=false

“move (arc) line” 指令中的 wait 表示是否需要等待本条指令执行完再发送下一条指令。

如果需要规划连续运动，请确保 wait=false。

Python 示例：

```
arm.reset(wait=True)
arm.set_pause_time(0.5)

while True:
arm.set_position(x=400, y=-100, z=250, roll=180, pitch=0, yaw=0, radius=50,speed=200, wait=False)
arm.set_position(x=400, y=100, z=250, roll=180, pitch=0, yaw=0, radius=50,speed=200, wait=False)
arm.set_position(x=300, y=0, z=250, roll=-180, pitch=0, yaw=0, radius=50,speed=200, wait=False)
```

set_position 接口：参考表 2.2，set_pause_time 接口说明，见表 2.3：

表 2.3 set_pause_time 说明

set_pause_time		
说明	设置机械臂暂停时间	
参数	sitime	暂停时间，单位：秒（s）；
	wait	是否等待，默认为 False；

2.1.3 圆及圆弧运动

圆弧运动根据三点坐标计算出空间圆的轨迹，三点坐标分别为（起点, 位置 1, 位置 2）。

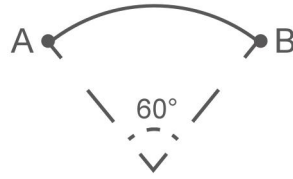
三点画圆的计算方式：

以当前点为起点，然后再设置两个位置点，三点确定一个圆。

圆心角的设置：

（1）若 $0 < \text{圆心角} (^{\circ}) < 360^{\circ}$ 或 $\text{圆心角} (^{\circ}) > 360^{\circ}$ ，则机械臂的运动路径为相应度数的圆弧；

若圆心角= 60° ，机械臂的运动轨迹为：



- (2) 圆心角 ($^{\circ}$) = 360° , 则机械臂可完整走完一整个圆弧;
- (3) 若想连续画多个圆弧 (例如, 连续画 10 段圆弧), 则将圆心角设置为所画圆弧度数的十倍。

Blockly 示例:



【圆弧运动 从位置 1 到位置 2】: 表示位置 1 到位置 2 走圆弧运动, 但该命令前面还需再加入一个位置点, 三点确定一个圆; 单位是 mm。

- 注: (1) 起点, 位置 1 和位置 2 确定一个整圆的三个参考点, 若机械臂的运动路径为一段圆弧, 则位置 1 和位置 2 不一定为终点或者经过点;
- (2) 若想让机械臂在运动过程中实现姿态变化, 请将位置 2 的 roll、pitch、yaw 设为完成轨迹时的期望姿态;

【圆心角 ($^{\circ}$)】: 表示该圆弧的度数, 设置 360° 时可以走完一整个圆弧, 可大于小于 360° ;

示例讲解:

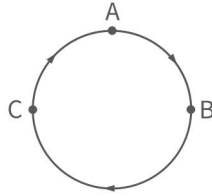
在此示例中, 圆心角设置为 3600° , 代表每次可连续画十个圆, 且机械臂每画完一个圆, 还是停留在起点的。

机械臂运动轨迹方向的判定

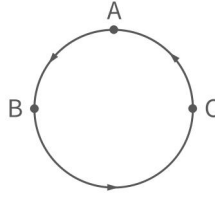
上例中起点, 位置 1 和位置 2 分别为:

A (300, 0, 400, 180, 0, 0) B (350, 50, 400, 180, 0, 0) C (350, -50, 400, 180, 0, 0)

- 机械臂以顺时针方向画圆，机械臂运动轨迹的方向如下图：



- 如果将 B 点和 C 的位置交换，B 点为 (350, -50, 400, 180, 0, 0) ， C 点为 (350, 50, 400, 180, 0, 0) ，则机械臂的将以逆时针方向画圆，机械臂运动轨迹如下：



Python 示例：

```
arm.set_servo_angle(angle=[0.0, -45.0, 0.0, 0.0, -45.0, 0.0], speed=20, mvacc=500, wait=True)
arm.set_position(*[300.0, 0.0, 400.0, 0.0, -90.0, 180.0], speed=300, mvacc=2000, radius=-1.0,
wait=True)
move_circle([350.0, 50.0, 400.0, 180.0, -90.0, 0.0], [350.0, -50.0, 400.0, 180.0, -90.0, 0.0],
1000.0, speed=300, mvacc=2000, wait=True)
```

arm.set_servo_angle 接口：查看表 2.1。

arm.set_position 接口：查看表 2.2。

arm.move_circle 接口说明，见表 2.4：

表 2.4 move_circle 说明

接口 move_circle 说明		
说明	该运动根据三点坐标计算空间圆的轨迹。三点坐标为（当前起点，位姿 1，位姿 2）	
参数	Pose1	笛卡尔坐标，[x(mm), y(mm), z(mm), roll(rad or °), pitch(rad or °), yaw(rad or °)]；
	Pose2	笛卡尔坐标，[x(mm), y(mm), z(mm), roll(rad or °), pitch(rad or °), yaw(rad or °)]；
	Percent	所移动圆弧的百分数；
	is_radian	若 is_radian=True, 则 roll/pitch/yaw 单位为 rad； 若 is_radian=False, 则 roll/pitch/yaw 单位为 ° ；
	speed	TCP 运动速度 (mm/s, rad/s)；
	mvacc	TCP 运动加速度 (mm/s ² , rad/s ²)；
	mvertime	0, 运动时间，暂无意义；
wait	若 wait=True, 为等待当前指令发完再发下一条指令； 若 wait=False, 则直接发下一条指令；	

2.3 奇异点

(1) 概念

当机械臂的任意两个关节的轴心处于同一直线时会出现奇异点。处于奇异点时, 机器人自由度将发生退化, 将会造成某些关节角速度趋于无穷大, 导致失控。常见情况是, 当腕关节(倒数第 2 个关节) 处于或接近第一关节轴线时, 也会出现奇异点(见图 2.1), 因而尽量避免直接经过靠近基座中心区域的笛卡尔轨迹, 容易造成 1 关节速度过大。



图 2.1 UFactory 8506 奇异点

(2) 特性

奇异点的特性为无法正确的进行规划运动。基于坐标的规划运动无法明确的反向转化为各轴的关节运动。机器人在奇异点附近进行规划运动(直线、圆弧等, 不包括关节运动)时会报警停止, 以避免通过奇异点时造成关节瞬间速度过快, 所以示教时应尽量避开奇异点或以关节运动通过奇异点。

(3) 针对奇异点的处理方法

情况一: 在机器人示教时遇到奇异点

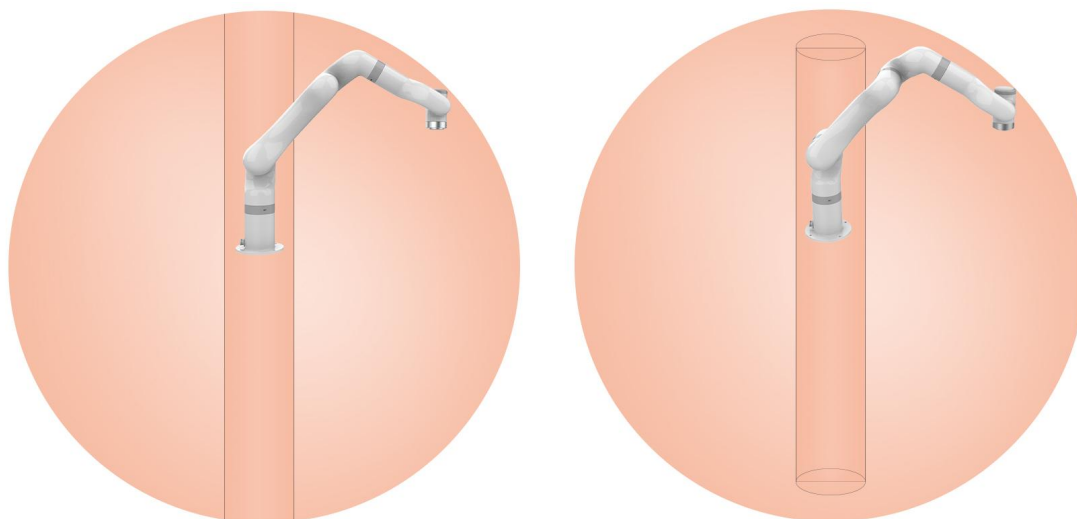
a) 将机器人的示教坐标系切换到关节。

情况二: 在程序运行时遇到奇异点

a) 当运行程序时遇到奇异点, 可以修改机器人的位置姿态, 以避开路径当中存在的奇异点, 重新进行到达目标点的路径规划。

注意：

选择机械臂安装位置时，务必考虑机械臂正上方和正下方的圆柱体空间（见下图）。尽可能避免将机械臂腕关节（J5）移向圆柱体空间，因为这样会造成机械臂慢速运动时关节运动超速，从而降低机械臂的工作效率，且风险评估难以进行。



3. 典型示例

在UFactory Studio中的Blockly里面有内置了10个例子，可以参考这十个例子进行创作编程。下面介绍其中几个较有代表性的例子。

3.1 UFactory 850 真空吸头的使用

该Blockly程序在示例程序中【UF】-1008_UFactory 850_Vacuum_Gripper



这段程序的作用：执行此程序，可控制真空吸头在指定位置吸取目标物，然后将目标物放到特定的位置。

注意：

1. 定义的函数要放在主程序的前面，如上图所示。

主要指令讲解：

- **【设置真空吸头（开/关） [设置]】**：可设置真空吸头开启和关闭状态。
【是否拾取检测】=true，同时检测了是否吸住物体，如果没有吸住物体，将会跳出整个程序；
【是否拾取检测】=false，不检测是否吸住物体。
- **【检测真空吸头已经（吸住/释放）】**：检测真空吸头是否吸住（释放）物体，若检测到真空吸头已吸住（释放）物体，则跳出这条指令，执行下一个指令，如果超过了超时时间后，真空吸头还未吸住（释放）物体，则也会跳出该指令，执行下一个指令。
- 循环运动计数：通过添加**【计数器自增】**，机械臂每次执行到该指令时会使控制器的计数器自增 1。可用于计算机械臂实际循环执行的次数。
- **【重置计数器】**：该指令将控制器内的计数器重置为 0。

3.2 UFactory 850 机械爪的使用

该 Blockly 程序在示例程序中**【UF】-1007_UFactory 850_Gripper**

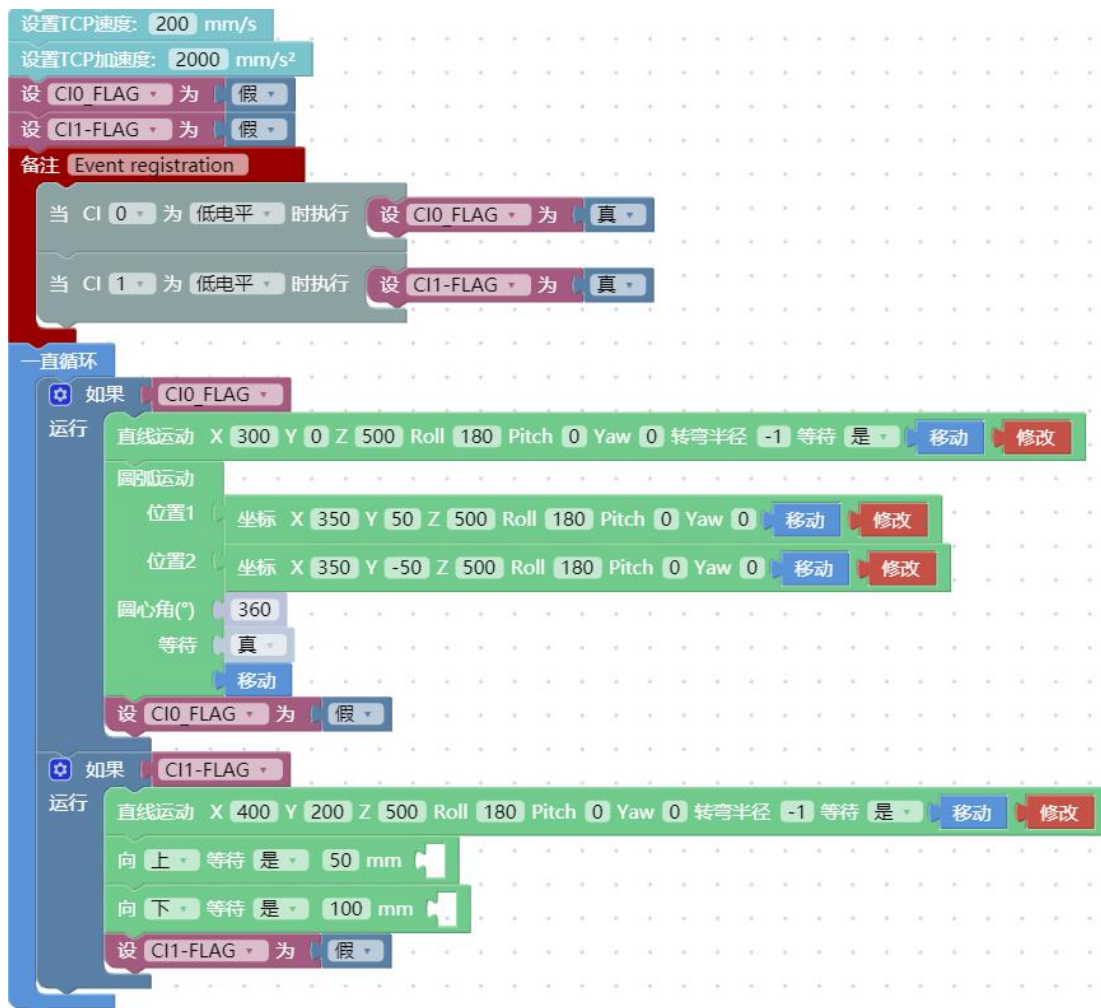


这段程序的作用：执行此程序，可控制机械爪在指定位置夹取目标物，然后将目标物放到特定的位置。

- 【设置 TCP 负载】：该指令在上述程序中使用多次，其主要目的是动态的改变负载，适应机械臂在夹取和释放物体的过程中 TCP 负载的变化。

3.3 数字 IO 的使用

该 Blockly 程序在示例程序中【UF】-1010_Digital_IO



这段程序的作用：如果需要使用数字 IO 来控制机械臂运动时，可以通过触发数字 IO，来执行相应的运动。

4. 机械臂运动模式和状态解析

4.1 机械臂运动模式解析

控制器的 7 种运动模式：（对应 SDK 中的 `mode()` 函数）

- 模式 0：位置控制模式。

控制器启动后默认进入的模式。

1. 关节运动：实现关节空间的点对点运动（单位：度/弧度），每个指令之间速度不连续。

[点击查看关节运动 `set_servo_angle\(\)` 代码示例。](#)

2. 直线运动：实现笛卡尔坐标之间的直线运动（单位：mm），每个指令之间速度不连续。

（可以用 Roll/Pitch/Yaw 表示机械臂姿态，也可以用轴角 [Rx, Ry, Rz] 表示机械臂姿态。）

[点击查看直线运动 `set_position\(\)` 代码示例](#)

[点击查看直线运动 `set_position_aa\(\)` 代码示例](#)

3. 直线圆弧运动：实现笛卡尔坐标之间的直线运动（单位：mm），在两条直线之间插入圆弧过度，每个指令之间速度连续。

[点击查看直线圆弧运动 `move_arc_lines\(\)` 代码示例](#)

4. 圆及圆弧运动：圆弧运动根据三点坐标计算出空间圆的轨迹，三点坐标分别为（起点，参数 1，参数 2）。

[点击查看圆与圆弧运动 `move_circle\(\)` 代码示例](#)

- 模式 1：关节伺服 (servoj) 模式。

servoj 运动，机械臂可以接受以 250Hz 频率发送的关节位置指令（注意：不是笛卡尔指令）若发送指令的频率超过 250Hz，则多余的指令将会丢失，每收到一条指令机械臂就会立即响应，并以最大速度 (180° /s) 和加速度执行移动至给定的关节位置（单位：度/弧度），此命令不设置缓冲区，仅执行最新收到的目标点。我们提供的 UFactory 850-Python-SDK 接口函

数也预留了速度、加速度和时间的设定，但目前也是不起作用的。如果用户层有能力完成速度、加速度平滑的运动轨迹规划并映射到关节空间，servoj 模式下可以替代控制器的规划，而令 UFactory 850 执行用户自己的或者第三方（比如 ROS Moveit!）的规划算法。如果用户无法自己实现轨迹规划以及运动学逆解，则不要使用此模式。

点击查看代码示例：

1. [set_servo_angle_j\(\)](#)

servo_cartesian 运动：以最快的速度 and 加速度移动至给定的笛卡尔位置点（单位：毫米（mm）），此命令不设置缓冲区，仅执行最新收到的目标点，需要进入 servoj 模式才可以使用。servoj 模式下，控制器的最大接收频率为 250Hz（1.4.0 之前的版本的最大接收频率为 100Hz），若发送指令的频率超过 250Hz，则多余的指令将会丢失。我们提供的 UFactory 850-Python-SDK 接口函数也预留了速度、加速度和时间的设定，但目前也是不起作用的。建议使用方法：如果用户自己规划轨迹，使用此命令最好是以固定的频率（100Hz 或者 200Hz）发布经过插值的平滑轨迹点（两个点之间的单步距离不超过 10mm），类似位置伺服控制指令。（注意：执行此指令类似最高速度的阶跃响应，为安全性考虑，切勿一次性给出过远的目标位置）。用户发布指令的频率建议控制在 30Hz-250Hz 的范围内，若频率低于 30Hz，机械臂的运动可能会不连续。使用此模式需要规划每一个轴的精细位置和预测机械臂的运动行为，开发难度较大。

点击查看代码示例：

1. [set_servo_cartesian\(\)](#)

2. [set_servo_cartesian_aa\(\)](#)

● 模式 2：关节示教模式。

此模式下，机械臂会进入零重力模式，用户可以自由拖动机械臂各连杆完成示教功能。如果拖动示教完成，请切换回模式 0。

安全使用注意：开启关节示教模式之前，务必确认机械臂安装方向和 TCP 负载设置正确，否则手臂可能会由于重力补偿不准确而无法自我保持静止！

点击查看代码示例：

1. [start_record_trajectory\(\)](#)

2. [playback_trajectory\(\)](#)

- 模式 4: 关节速度控制模式。

此模式下, 可设置关节目标速度。

点击查看代码示例:

```
1. vc_set_joint_velocity()
```

- 模式 5: 笛卡尔速度控制模式。

此模式下, 可设置 3 维笛卡尔线速度和 3 维笛卡尔姿态角速度。

点击查看代码示例:

```
1. vc_set_cartesian_velocity()
```

- 模式 6: 关节在线轨迹规划模式。

通过 `set_servo_angle()` 发送的指令该模式下每收到一个运动指令, 都会打断当前正在进行的运动指令, 然后从当前位置开始规划执行该运动指令, 后一个运动指令可以打断正在进行的运动指令。

点击查看代码示例:

```
1. set_mode(6)
```

- 模式 7: 笛卡尔在线轨迹规划模式。

通过 `set_position()` 或者 `set_position_aa()` 发送的指令该模式下每收到一个运动指令, 都会打断当前正在进行的运动指令, 然后从当前位置开始规划执行该运动指令, 后一个运动指令可以打断正在进行的运动指令。

注意: 使用笛卡尔在线规划模式时, `set_position_aa()` 中的 `is_tool_coord` 必须为 `False`, 即笛卡尔在线规划模式只能是以基坐标系为参考坐标系, 不能以工具坐标系做相对运动。

点击查看代码示例:

```
1. set_mode(7)
```



注意

设计机器人运动规划的操作员必须要具备如下条件：

1. 良好的安全意识，充分的操作规范流程知识。
2. 对机器人有较深了解，明白关节运动方式与直线运动方式。
3. 具备应对紧急情况的安全知识。
4. 在做机器人运动规划时，必须做风险评估，请操作员务必小心。

4.2 机械臂运动状态解析

控制器可以设置的的 3 种状态：（对应 SDK 中的 `set_state()` 函数）

- 状态 0：开启运动。

可以理解为准备好运动或运动就绪状态。在此状态下，机械臂能够正常响应和执行运动指令。如果机械臂从错误、断电或停止状态（状态 4）中恢复，请记得在继续发送运动指令之前设置状态为 0，否则发送的指令会被丢弃。

- 状态 3：暂停状态。

暂停当前在执行的运动，并可以通过再次设置状态 0 在中断处继续恢复运行。

- 状态 4：停止状态。

终止当前的运动，并清空已缓存的后续指令。需要设置状态 0 之后方可继续运动。

控制器可以获取到的 6 种状态：（对应 SDK 中的 `get_state()` 函数）

- 状态 1：运动中。

机械臂正在执行运动指令，非静止状态。

- 状态 2：休眠中。

控制器已经在运动就绪状态，但没有运动指令可执行。

- 状态 3：暂停中。

机械臂被设置了暂停状态，运动指令缓存可能不为空。

- 状态 4：停止中。

控制器停止状态，在这个状态下机械臂不会接收和执行一切运动相关的指令，可以接收和执行系统设置和获取信息类的指令。开机默认进入的状态。

- 状态 5：系统重置。

用户刚刚进行模式切换或者变更了某些设置（比如 TCP 偏移，灵敏度等等）后自动进入的状态，以上操作会使正在进行的运动终止并清空指令缓存，表现与 STOP 状态相同。

- 状态 6：停止。

一般用 studio 会产生状态 6，状态 6 也是停止的一种。

附录

附录 1-报错及处理

1.1 关节报警信息和常规处理方式

报警处理方式可采用重新上电，步骤如下（重新上电需要走完以下所有步骤）：

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

UFactory studio 使能方式：点击报错弹窗的引导按钮或者首页的使能按钮。

UFactory 850-Python-SDK 使能方式：[参照报警处理方式](#)。

UFactory 850_ros 库：查看相关文档 https://github.com/UFactory_850-Developer/xArm_ros

若多次上电无效后请寻找 UFACTORY 团队支持，至邮件：support@ufactory.cc。

软件报错代码	报警处理
S0	关节通信失败 请通过控制器上的紧急停止按钮重启机械臂。如多次重启无效，请联系技术支持。
S10	电流检测异常 请通过控制器上的紧急停止按钮重启机械臂。
S11	关节电流过大 请通过控制器上的紧急停止按钮重启机械臂。
S12	关节速度过大 请通过控制器上的紧急停止按钮重启机械臂。
S14	位置指令过大 请通过控制器上的紧急停止按钮重启机械臂。
S15	关节过热 如果机械臂长时间运行温度过高，请停机，等待机械臂冷却后重启机械臂。
S16	编码器初始化异常 请确保机械臂通电时，无外力推动机械臂运动。请通过控制器上的紧急停止按钮重启机械臂。
S17	单圈编码器错误 请通过控制器上的紧急停止按钮重启机械臂，如多次重启无效，请联系技术支持。
S18	多圈编码器错误 请点击“清除错误”，随后将控制器电源开关推至 OFF 档，等待 5 秒后重新上电。如多次尝试无效，请联系技术支持。
S19	电池电压过低
S20	驱动 IC 硬件异常 请重新使能机械臂。

S21	驱动 IC 初始化异常 请通过控制器上的紧急停止按钮重启机械臂
S22	编码器配置错误
S23	电机位置偏差过大 请检查机械臂运动是否受阻，末端负载是否超过机械臂额定负载，机械臂加速度值是否设置过大。
S26	第 N 关节正向超限 请检测 N 关节角度值是否设置过大。
S27	第 N 关节负向超限 请检测第 N 关节角度值是否设置过大，如果是，请点击清除报错后，手动解锁该关节并转动该关节至其运动范围内。
S28	关节指令错误 机械臂未使能，请点击“使能机械臂”。
S33	驱动器过载， 请确保机械臂负载处于额定负载内。
S34	电机过载 请确保机械臂负载处于额定负载内。
S35	电机类型错误 请通过控制器上的紧急停止按钮重启机械臂。
S36	驱动器类型错误 请通过控制器上的紧急停止按钮重启机械臂。
S39	关节过压 请在运动设置中减少加速度值。
S40	关节欠压 请在运动设置中减少加速度值，请检查控制器紧急停止开关是否松开。
S49	EEPROM 读写错误 请通过控制器上的紧急停止按钮重启机械臂。
S52	电机角度初始化失败 请通过控制器上的紧急停止按钮重启机械臂。
上表中未出现的报警代码：重新上电。如多次上电无效，请联系技术支持。	

1.2 控制器报警信息及常规处理方式

1.2.1 控制器错误代码

机器人硬件出现错误、控制软件出现错误、下发指令错误，会发出错误或警告，这个错误/警告信号会在用户发送任意指令时反馈回去，也就是这个反馈是被动的，并非主动上报。

注意：

出现以下错误后，机器人会立即停止工作，并且丢弃控制器缓存指令。需手动清除错误后，机器人方可继续正常工作。根据上报的错误信息，重新调整机器人的运动规划。

软件报错代码	报警处理
C1	控制器上的紧急停止按钮被按下 请释放紧急停止按钮，然后点击“使能机械臂”。
C2	控制器上的紧急停止 IO 被触发 请将控制器的 2 组 EI 接地，然后点击“使能机械臂”。

C3	三态开关的紧急停止按钮被按下 请释放三态开关的紧急停止按钮，然后点击“使能机械臂”。
C11-C17	请通过控制器上的紧急停止按钮重启机械臂。
C19	末端通信失败 请检查机械爪是否安装，波特率设置是否正确。
C21	运动学错误 请重新规划路径。
C22	自碰撞错误 机械臂即将发生自碰撞，请重新规划路径。如果机械臂持续报自碰撞错误，请到“实时控制”界面开启“手动模式”，将机械臂拖回正常位置。
C23	关节角度超出限制 请点击“初始点”按钮回到关节零点。
C24	速度超出限制 请检查机械臂是否超出运动范围，或减小运动速度和加速度。
C25	规划错误 请重新规划路径或者减小运动速度。
C26	Linux RT 错误
C27	回复指令错误 请重试，或通过控制器上的紧急停止按钮重启机械臂。
C29	其他错误
C30	反馈速度超出限制
C31	机械臂电流异常 1. 检查机械臂是否发生碰撞； 2. 检查“设置”-“TCP 设置”-“TCP 负载”处设置的质量、质心是否与实际的负载匹配； 3. 检查“设置”-“安装方式”处设置的安装方向是否与实际匹配； 4. 检查程序中设置的 TCP 负载参数是否与实际匹配； 5. 降低机械臂运动速度； 6. 前往“设置”-“运动”-“灵敏度设置”，调低碰撞检测灵敏度。
C32	三点圆弧指令计算出错 请重新设置圆弧指令。
C33	控制器 IO 模块异常
C34	轨迹录制超时 轨迹录制时间超过最大限制 5 分钟，建议重新录制。
C35	机械臂到达安全边界 机械臂到达安全边界，请让机械臂在安全边界内运行。
C36	延时指令数量超限 1. 请检查位置检测或 IO 延时指令是否过多。 2. 增加位置检测指令的容错半径。
C37	手动模式运动异常（负载设置或安装方式设置错误） 请检查机械臂的 TCP 负载设置和机械臂安装方式设置是否与实际匹配。
C38	关节角度异常 请通过控制器上的紧急停止按钮停止机械臂，并联系技术支持。
C39	控制器电源板主从 IC 通信异常 请联系技术支持。
C50	六维力矩传感器错误 请查询六维力矩传感器用户手册错误码，定位问题并重新上电。
C51	六维力矩传感器模式设置错误 请确保机械臂没有处于手动模式，检查本指令给定值是否为 0/1/2
C52	六维力矩传感器零点设置错误 请检查传感器通信接线以及通电是否正常
C53	六维力矩传感器过载 请减小负载或施加的力
C110	机械臂底座板通信异常

C111	控制器外接 485 设备通信异常
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。	

1.2.2 控制器警告代码

警告是不影响机器人正常工作，但有可能会影响用户的程序运行，一旦发生警告，机器人会设置警告标志位，并在指令回复中一起返回，除此之外不会进行其他操作，机械臂依然会正常运行。

警告代码	描述	报警处理
11	缓存溢出	控制指令缓存
12	指令参数异常	检查发送指令
13	指令不存在	检查发送指令
14	指令无解	检查发送指令

1.3 机械爪报警信息及常规处理方式

报警处理方式可采用重新上电，步骤如下（重新上电需要走完以下所有步骤）：

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

UFactory studio 使能方式：点击报错弹窗的引导按钮或者首页的使能按钮。

UFactory 850-Python-SDK 使能方式：参照[报警处理方式](#)。

UFactory 850_ros 库：查看相关文档 https://github.com/UFactory_850-Developer/xArm_ros

3. 新使能机械爪：若多次上电无效后请联系 UFACTORY 团队。

软件报错代码	报警处理
G9	机械爪电流检测异常 请通过控制器上的紧急停止按钮重启机械臂。
G11	机械爪过流 请点击“确认”重新使能机械爪。
G12	机械爪过速 请点击“确认”重新使能机械爪。
G14	机械爪位置指令过大 请点击“确认”重新使能机械爪。
G15	机械爪 EEPROM 读写错误 请点击“确认”重新使能机械爪。
G20	机械爪驱动 IC 硬件异常 请点击“确认”重新使能机械爪。

G21	机械爪驱动 IC 初始化异常 请点击“确认”重新使能机械爪。
G23	机械爪电机位置偏差过大 请检查机械爪运动是否受阻，如机械爪运动未受阻，请点击“确认”重新使能机械爪。
G25	机械爪指令超软件限位 请检测机械爪指令是否设置超出软件限制。
G26	机械爪反馈位置超限软件限位
G33	机械爪驱动器过载
G34	机械爪电机过载
G36	机械爪驱动器类型错误 请点击“确认”重新使能机械爪。
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。	

1.4 Python SDK 报警信息及常规处理方式

软件报错代码	报警处理
A-9	紧急停止
A-8	TCP 位置指令超出机械臂运动范围，请调整 TCP 位置指令。
A-2	机械臂未准备好，请检查机械臂是否已经使能，机械臂状态是否设置正确。
A-1	未连接到 UFactory 850 控制器或连接中断，请检查网路连接。
A1	有错误未清除，请清除错误后重试。
A2	有警告未清除，请清除警告后重试。
A3	获取回复超时，请检查固件版本和网络连接。
A4	TCP 回复长度错误，请检查网络连接。
A5	TCP 回复序号错误，请检查网络连接。
A6	TCP 协议标志错误，请检查网络连接。
A7	TCP 回复指令与发送指令不匹配，请检查网络连接。
A8	发送指令错误，请检查网络连接。
A9	机械臂未准备好，请检查机械臂错误是否已清除，机械臂是否已经使能，机械臂状态是否设置正确。
A11	其他错误，请联系技术支持。
A12	参数错误。
A20	末端工具 IO ID 错误。
A22	末端工具 Modbus 波特率错误。
A23	末端工具 Modbus 回复长度错误。
A31	轨迹读写失败。
A32	轨迹读写超时。
A33	运行轨迹超时。
A41	真空吸头等待超时。

A100	等待完成超时。
A101	检测末端执行器状态失败次数过多。
A102	末端执行器有错误。
A103	末端执行器未使能。
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。	

UFactory 850-Python-SDK 报警处理方式：

在用 Python 库设计机器人运动规划时，如果机器人出现故障，需要手动清除错误。清除错误后，仍需重新给机器人使能，设置运动模式，方可使机器人正常运动。此时根据上报的错误信息，应重新调整机器人的路径规划。

Python 库清除错误步骤：（如下接口，详细说明请查看 GitHub）

1. 清除错误：`clean_error()`
2. 重新使能机械臂：`motion_enable(true)`
3. 设置运动状态：`set_state(0)`

附录 2-技术规格

2.1 UFactory 850 规格

UFactory 850		
笛卡尔范围	X	±850 mm
	Y	±850 mm
	Z	-400 mm~1214 mm
	Roll/Yaw/Pitch	± 180°
最大关节速度		180° /s
有效工作半径		850mm
重复定位精度		±0.02mm
最大末端速度		1m/s
环境温度		0-50 ° C
功耗		最低 20 W, 典型 240 W, 最高 1000W
输入电源		48 V DC, 20.8A
ISO 洁净室等级		5
机械臂安装		任意角度
编程方式		UFactory studio 图形界面 Python/C++/ROS 底层接口
机械臂通信协议		自定义 TCP 协议
末端工具 IO 接口		2 个数字输入, 2 个数字输出, 2 个模拟输入
末端工具 485 通信协议		Modbus TCP
占地面积(mm)		Ø 190 mm
重量 (仅机械臂)		20 kg
最大负载重量		5 kg
材料		铝、碳纤维
末端工具法兰		DIN ISO 9409-1-A50/63 (M6*6)
 <p>The diagram illustrates the six joints of the UFactory 850 robotic arm. Joint 1 (关节1) is the base rotation. Joint 2 (关节2) is the shoulder rotation. Joint 3 (关节3) is the elbow rotation. Joint 4 (关节4) is the wrist rotation. Joint 5 (关节5) and Joint 6 (关节6) are the end effector rotations. Red arrows and '+'/'- signs indicate the direction of rotation for each joint.</p>		

控制器		
	交流控制器	直流控制器
输入	100-240V AC 50/60Hz	48-72V DC
输出	48V DC 1000Wmax	48V DC 960Wmax
控制器通信协议	自定义 TCP 协议	
控制器通信方式	Ethernet (以太网)	
控制器 IO 接口	8*CI+8*DI (数字输入) 8*CO+8*DO (数字输出) 2*AI (模拟输入) 2*AO (模拟输出) 1*RS-485 主	8*CI+8*DI (数字输入) 8*CO+8*DO (数字输出) 2*AI (模拟输入) 2*AO (模拟输出) 1*RS-485 主
质量	4.8kg	2.8kg
尺寸(L*W*H)	345*135*101mm	262*160*76mm
机械爪		
额定电源电压	24V DC	
绝对最大电源电压	28V DC	
静态功耗 (最低功耗)	1.5W	
峰值电流	1.5A	
工作范围	84mm	
最大夹持力度	30N	
质量	802g	
通信方式	RS-485	
通信协议	Modbus RTU	
可编程参数	位置、速度	
反馈	位置	
真空吸头		
额定电源电压	24V DC	
最大电源电压	28V DC	
静态电流	30mA	
峰值电流	400mA	
真空度	78%	
真空流速(L/min)	>5.6L/min	
质量 (g)	610g	
尺寸 (L*W*H)	122.5*91.6*75mm	
有效载荷 (kg)	≤5kg	
噪音水平(30cm 内)	<60dB	
通信方式	数字 IO	
状态指示灯	电源, 工作状态	
反馈信号	气压 (低或常规)	
备注: *1. 机械臂的工作温度是 0-50℃, 关节连续高速操作时, 降低环境温度。		

附录 3-UFactory 850 软件/固件升级的方法

UFactory 850 软件/固件升级注意事项

- 建议将 UFactory 850 固件和 UFactory Studio 软件同时更新。
- 每次更新前，请仔细查看更新日志。已经部署到生产环境中的设备不建议更新。
- 每次更新后，请从官网下载最新的《UFactory 850 用户手册》和《UFactory 850 开发者手册》，以了解 UFactory 850 的最新功能。若使用 UFactory 850-Python-SDK（UFactory 850-C++ SDK 或者 UFactory 850 ROS）进行开发，请从 GitHub 获取最新 SDK 代码。

手册下载地址：<https://www.cn.ufactory.cc/xArm-download>

SDK 获取地址：<https://github.com/xArm-Developer>

3.1 在线升级：当 PC 端有网络连接时

方法一：使用 UFactory studio 进行在线升级。

进入【我的设备】→【检查更新】，点击“检查更新”，如有新的版本，点击“下载”，下载完成后点击“安装”加载下载好的安装包，等待系统重启。重启耗时约 2-3 分钟。

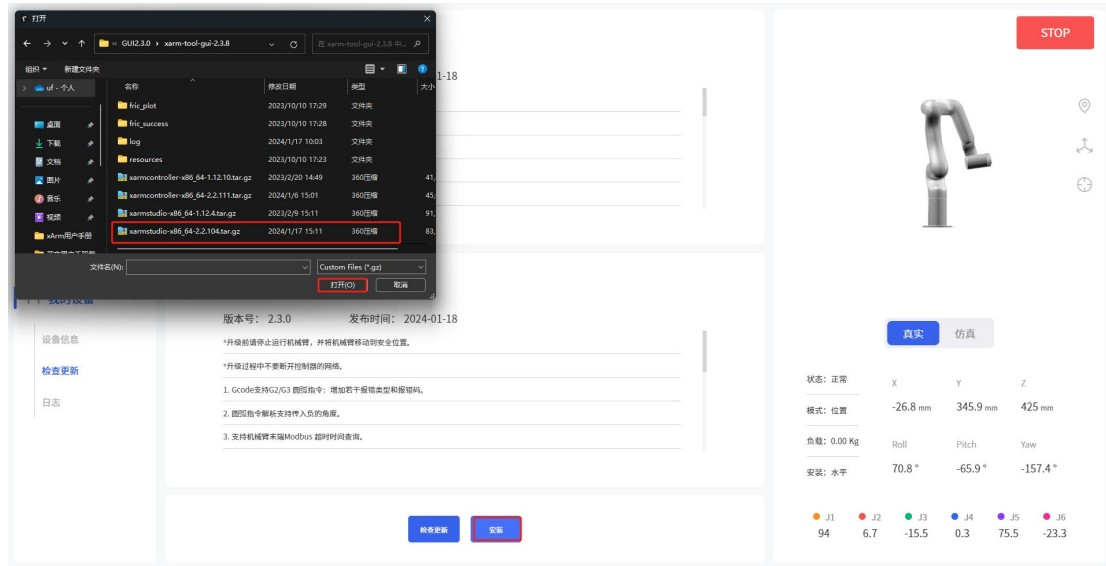


3.2 离线升级：当 PC 端没有网络连接时

方法一：使用 UFactory Studio 离线升级。

[点击下载](#)

进入【我的设备】-【检查更新】，点击“安装”加载提前下载好的离线包，等待安装完成系统重启即可。



注意：当在线升级不成功时，可以尝试离线升级。离线升级仍然不成功时，请联系技术支持

(support@ufactory.cc)。

附录 4-维护与点检

1. 长期放置

机械臂如果长时间（≥3 个月）不用，需要每 3 个月给 UFactory 850 机械臂通电 6 小时来给机械臂内置电池充电。

给机械臂通电时，控制器上的急停按钮需要松开，可以不使能机械臂。

2. 清洁

机械臂长期使用后，碳纤维外壳上可能会有污垢或油脂（在极少数情况下，可在关节处看到少量的润滑油脂溢出，这不会影响关节的正常使用或寿命），可以使用 95%酒精或 70% 的异丙醇擦拭碳纤维表面进行清洁。

注意：

清洁碳纤维表面时，注意不要让液体渗入关节。

附录 5-售后服务

1. 维修服务

机械臂维修只能由深圳市众为创造科技有限公司进行。

维修后的机械臂，在使用前必须进行测试，确保机械臂符合安全标准，并确认机械臂所有安全功能的正常运行。

2. 售后政策：

对于产品的质量保证金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>

3. 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. 请将产品的原包装材料保存好，当需要将产品寄回我司进行维修时，需要将产品用原装箱

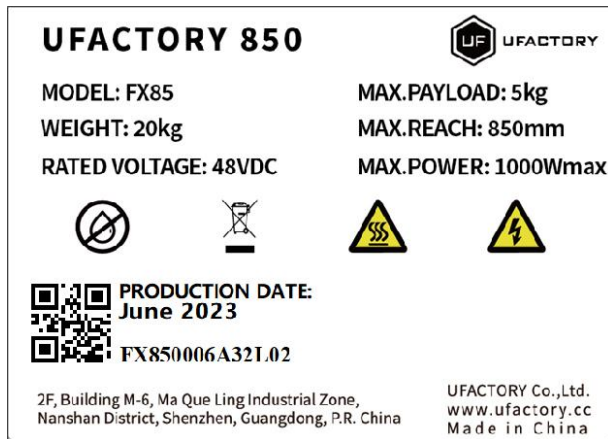
打包好，避免在运输过程中发生不必要的碰撞，导致机械臂受损。

2. 当需要寄回控制器时，您需要将机械臂的配置文件导出并保存（请参考的1.4.4 设置-通用设置-高级设置-配置文件），以免控制器在维修过程中出现原始数据丢失或变更的情况。

附录 6-产品信息

1.1 产品标签

机械臂标签



控制器标签



1.2 电磁兼容

- IEC 61000-6-2:2005
- IEC 61000-6-4/A1:2010
- EN 61000-6-2:2005 [2004/108/EC]
- EN 61000-6-4/A1:2011 [2004/108/EC]

电磁兼容性 (EMC)

第 6-2 部分：通用标准 - 工业环境的抗抗干扰

第 6-4 部分：通用标准 - 工业环境辐射标准

这些标准定义了对电气和电磁干扰的要求。符合这些标准可确保 UFactory 850 机器人在工业环境中表现良好，并且不会干扰其他设备。

- EN 61000-6-4:2019
- EN 61000-6-2:2019

测量、控制和实验室用电气设备 - EMC 要求

第 3-1 部分：安全相关系统和旨在执行安全相关功能（功能安全）的设备的抗干扰要求 - 一般工业应用。该标准定义了安全相关功能的扩展 EMC 抗扰度要求。符合此标准可确保 UFactory 850 机器人的安全功能提供安全，即使其他设备超过 IEC 61000 标准中定义的 EMC 辐射限制。

1.3 使用环境

- 低湿度（25%-85% 无冷凝）
- 海拔高度：<2000m
- 环境温度：0° C ~ 50° C
- 避免阳光直射（室内使用）
- 无腐蚀性气体或液体。
- 无易燃材料。
- 无油雾。
- 无盐雾。
- 无灰尘或金属粉末。
- 无机械冲击、振动。
- 无电磁噪声。
- 无放射性物质。

1.4 运输、存储和搬运

- 通过 UFactory studio 将机器人移动到零点，然后将 UFactory 850 机器人和控制器放入原包装中。
- 使用原包装运输机器人。
- 将机械臂从包装移至安装地点时，同时提起机械臂两侧的连杆，将机器人固定到位，直到所有安装螺栓都牢固地拧紧在机器人底座上。
- 控制器箱应通过手柄提起。
- 将包装材料保存在干燥的地方，以后可能需要打包和运输机器人。

1.5 控制器放置高度

控制器应放置在 0.6m 至 1.5m 的高度。

1.6 电源连接

本产品电源切断方式为插头/插座连接方式，所以在使用本产品的时候，建议配备适当的具有足够分断能力的开关电器（如空气开关 绝缘电压：400V AC 额定电流：10A）

1.7 特殊耗材

保险丝规格：15A 250V 5×20mm Time-Lag 玻璃体筒式保险丝

1.8 停机类别

停止类别 1 和停止类别 2 会在驱动器电源打开的情况下使机器人减速，这使机器人能够在不偏离当前路径的情况下停止。

安全输入	描述
控制器上的紧急停止按钮	1 类停机
控制器上的紧急停止输入（EI）	1 类停机
三档使能开关上的紧急停止按钮	1 类停机
三档使能开关	2 类停机
控制器上的防护停止输入（SI）	2 类停机

1.9 停止时间和停止距离

1 类停机的停止距离和时间。

下表包括触发 1 类停机时测量的停止距离和停止时间。 这些测量值对应于机器人的以下配置：

- 臂展： 100% (机器人手臂完全水平伸展) 。
- 速度： 100% (机器人的一般速度设为 100%，以 $180^\circ /s$ 的关节速度进行移动) 。
- 负载： 机械臂可以承受的最大有效载荷 (5 kg) 。

通过执行水平移动 (即旋转轴垂直于地面) 对关节1 进行测试。在对关节2 和关节3 的测试过程中，机器人遵循垂直轨迹，即旋转轴平行于地面，并在机器人向下移时停止。

	停止距离 (rad)	停止时间 (ms)
关节 1	0.62	521
关节 2	1.12	885
关节 3	0.67	577

1.10 最大速度

模式	典型场景	最大速度
示教模式	UFactory studio 实时控制界面	250 mm/s
自动模式	UFactory studio 的 Blockly/ IDE 编程界面	1000 mm/s

1.11 参数

机械臂型号	XF8500	
旋转范围	1, 4, 6	±360°
	2	-132° ~132°
	3	-242° ~3.5°
	5	-124° ~124°
负载	5kg	
臂展	850mm	
自由度	6	
重复定位精度	±0.02mm	
最大关节速度	180° /s	
重量 (不含控制器)	20kg	
占地面积	Ø 190 mm	
最大功率	1000W	
额定电压	48V DC	

控制器型号	AC15
尺寸	376mm*145mm*130mm
I/O 接口	8*CI 8*DI 8*CO 8*DO 2*AI 2*AO 2*RS-485
通信协议	自定义 TCP 协议
重量	4.8kg
环境温度	0-50°C
湿度	25%-85% (无冷凝)
短路电流	50A
输入	单相 AC 100-240V 47~63HZ 1050Wmax

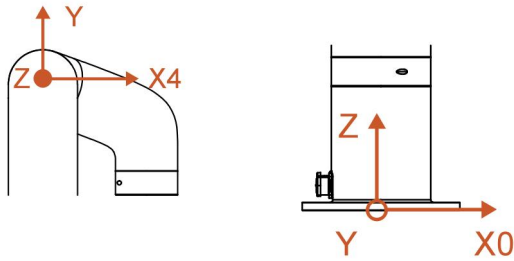
附录 7-xArm 系列的运动学参数

● UFACTORY 850 的运动学和动力学参数

坐标表示的注释：

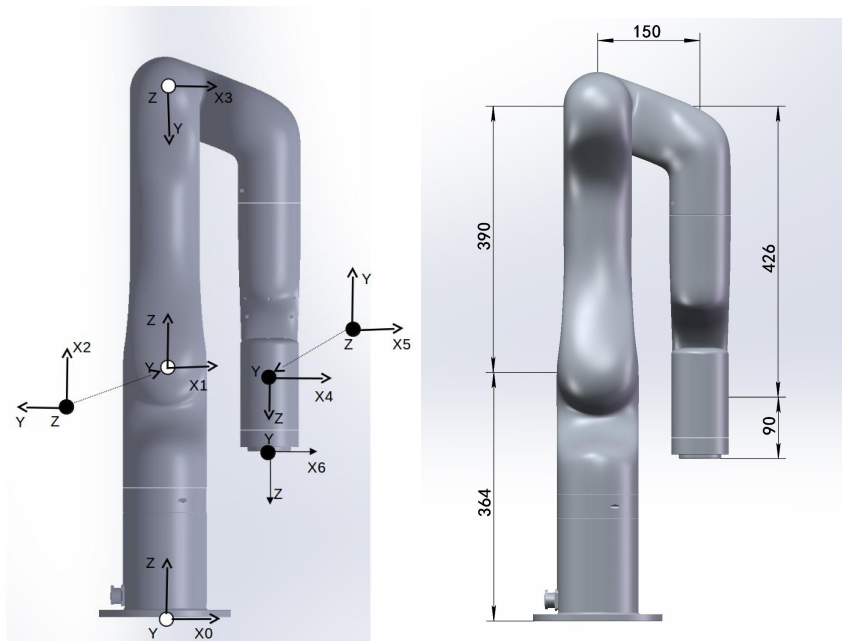
实心 (●)： 这意味着方向指向观察表面之外。

空心 (○)： 这意味着方向指向观察表面。



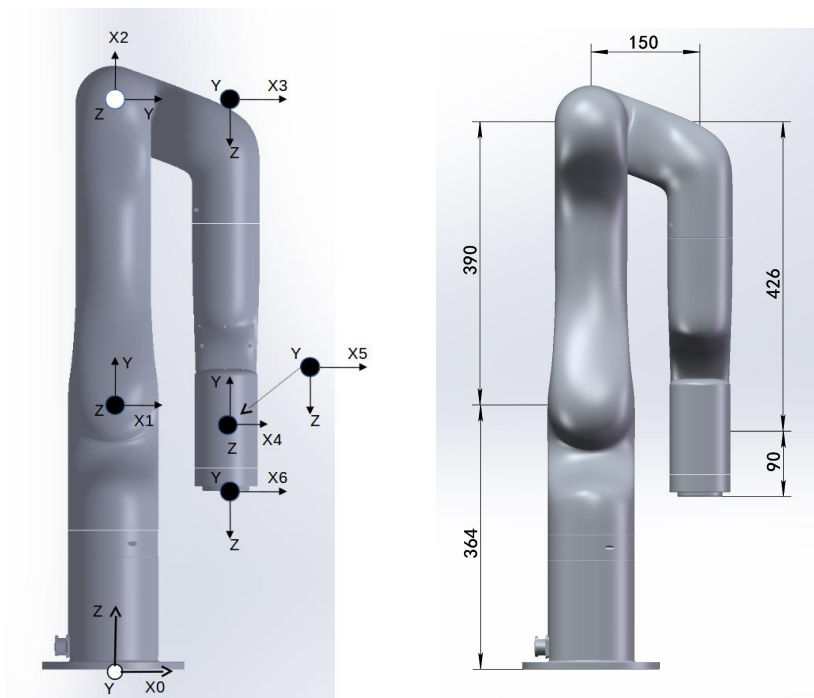
* 请注意提供的动力学参数仅供参考。

1) UFACTORY 850 修改 DH 参数



Kiematics	α_{i-1} (°)	a_{i-1} (mm)	d_i (mm)	θ_i (°)
Joint1	0	0	364	0
Joint2	90	0	0	90
Joint3	180	390	0	90
Joint4	-90	150	426	0
Joint5	-90	0	0	0
Joint6	90	0	90	0

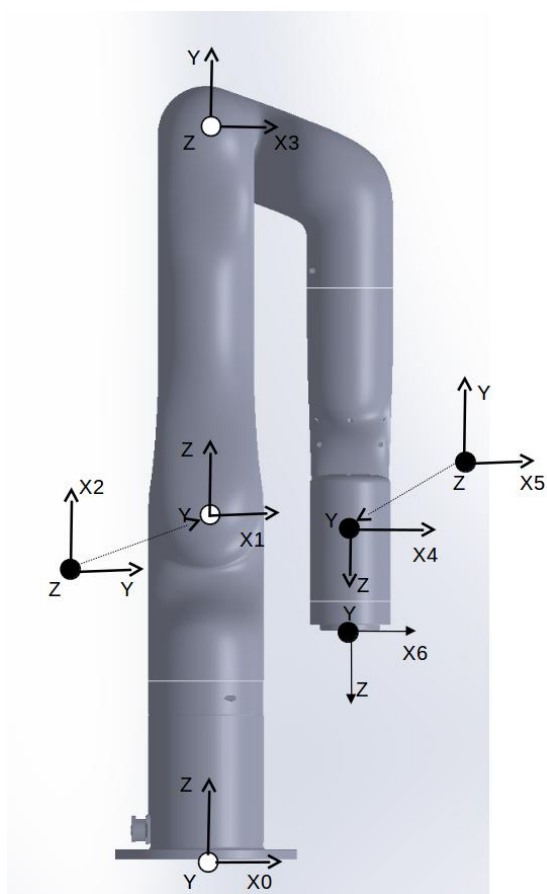
2) UFACTORY 850 标准 DH 参数



Kiematics	α_{i-1} (°)	a_{i-1} (mm)	d_i (mm)	θ_i (°)
Joint1	90	0	364	0
Joint2	180	390	0	90
Joint3	-90	150	0	90
Joint4	-90	0	426	0
Joint5	90	0	0	0
Joint6	0	0	90	0

3) UFACTORY 850 质量参数

关节坐标定义



UFACTORY 850 模型

Dynamics	Mass (Kg)	Center of Mass (mm)
Link1	4.881	[-0.049, 13.92, -20.31]
Link2	3.166	[327.98, 0.074, 6.48]
Link3	2.247	[130.23, 105.18, 8.97]
Link4	2.12	[0.016, -38.47, -98.3]
Link5	1.346	[-0.42, -17.6, -1.42]
Link6	0.225	[0.15, 1.38, -13.97]

制造商：深圳市众为创造科技有限公司

地址：广东省深圳市南山区麻雀岭工业区 M6 栋 2 楼

网址：www.ufactory.cc