

OmniHand 灵巧手 产品使用说明书

版本号 1.9

目录

版本号 1.8	1
目录	1
1. 重要安全须知	2
1.1 使用注意事项	3
1.2 风险评估	4
1.3 售后服务承诺书	4
1.4 免责声明	5
2. 产品简介	6
2.1 产品概述	6
2.2 主要特性	7
2.3 产品参数	7
2.4 关节角度范围	9
2.5 产品尺寸	10
3. 安装说明	12
3.1 装箱清单	12
3.2 机械接口	13
3.3 电气和通讯接口	15
3.4 通信方式	17
4. 通讯协议说明	18
4.1 CAN-FD 通信帧格式	18
4.2 USB/串口 通信帧格式	20
4.3 通信协议内容	23
4.4 固件/上位机版本	28
4.5 控制注意事项	29
4.6 指示灯功能说明	29
4.7 整手不支持热插拔	错误!未定义书签。
5. 上位机使用说明	29
5.1 运行环境	29
5.2 下载安装	30
5.3 登录界面	30
5.4 主界面（上位机版本 2.6.33）	30
5.5 手部控制	32
5.5.1 自定义动作播放/循环	33
5.5.2 保存和加载自定义动作文件	33
5.6 压力矩阵	34
5.7 固件升级	35

1. 重要须知

本章节介绍了安装和使用 OmniHand 灵动款/灵动触觉款 灵巧手时必须遵守的安全要求，OmniHand 灵巧手是一款半成品机械，因此，在首次使用之前，必须仔细阅读本说明书，全面了解相关安全功能及潜在危险。相关人员在每次安装 OmniHand 灵巧手之前，必须进行风险评估，并在使用过程中严格遵守本说明书中所有说明和指导信息。

1.1 安全警示与使用注意事项

- 严禁将 OmniHand 灵巧手置于高温、高湿、易燃、易爆的环境中，禁止靠近火源，严禁在强电磁场环境中使用（如高压电线附近、大功率机器附近等）。
- 严禁在手指施加过大的力，或用力拉扯灵巧手，请勿让 OmniHand 灵巧手承受过大的负载和冲击。
- 严禁用 OmniHand 灵巧手抓取过重、过热、尖锐、表面粗糙、有腐蚀性的物体。
- 严禁用 OmniHand 灵巧手操作危险机械。由于此类行为导致的人身伤害及财产损失，我司不承担任何责任。
- 严禁未成年人单独操作 OmniHand 灵巧手。确需使用时，须在具备操作资质的成年人员全程监护和指导下进行。
- 未接受过产品安全操作培训、不熟悉设备运行逻辑与安全规范的人员，严禁独立进行通电运行与调试操作。
- 严禁酒后、疲劳、服用影响判断或反应能力药物后操作 OmniHand 灵巧手，严禁行动或认知能力受限人员独立操作 OmniHand 灵巧手。
- 操作前确保所有电缆、插头、插座完好无损。遇到异常情况(如短路、过热等)时立即断开电源。

1.2 风险评估

潜在危害清单

机械危害：因接触 OmniHand 灵巧手或其夹持的尖锐物体而导致压伤、擦伤等。

电气危害：因接触带电零件而导致电烧伤或触电。

高温危害：因长时间接触 OmniHand 灵巧手的高温表面而导致皮肤烧伤等。

噪音危害：无。

振动危害：无。

辐射危害：无。

材料/物质危害：无。

人体工程学危害：因接触组件缝隙而导致夹伤等。

环境危害：无。

1.3 质保与售后服务范围

OmniHand 灵巧手具有 6 个月保修期。质保期内，若在投入使用后，出现因制造或材料不良所致的缺陷，公司提供必要的备用部件予以更换或维修。若设备缺陷是由处理不当或未遵循产品使用说明书中所述的相关信息或私自拆卸产品所致，则本产品质量保证即告失效。在不违背本产品质量保证的原则下，若产品已经超出保修期，公司保留向客户收取更换或维修费用的权利。在保修期外，如果设备呈现缺陷，公司不承担由此引起的任何损害或损失，包括但不限于生产损失或对其他生产设备造成的损坏。

严禁私自拆解 OmniHand 灵巧手，否则将导致《销售协议》中质保相关条款失效。如发生任何故障，请及时联系售后维修人员。

客户可另行购买延保或 OmniCare+ 服务，请联系销售或客户服务团队了解具体内容。

1.4 免责声明

公司致力于不断提高产品可靠性和性能，并因此保留升级产品的权利，恕不另行通知。公司力求确保本手册内容的准确性和可靠性，但不对其中的任何错误或遗漏信息负责。以下情况导致的故障不在本保修范围内：

- 1) 未按用户手册要求安装、接线、连接其他控制设备；
- 2) 未经允许，私自拆装灵巧手；
- 3) 使用时超出用户手册所示规格或标准；
- 4) 由于运输不当导致的产品损坏；
- 5) 事故或碰撞导致的损坏；
- 6) 火灾、地震、海啸、雷击、大风和洪水等自然灾害。
- 7) 整手在带电状态下，不支持电源及信号线热插拔功能，请使用者规范使用；如若未按要求进行热插拔出现故障，客户承担相应责任。

本公司对于因客户违反本章节内的免责声明而造成的任何损失或伤害不承担任何责任。请客户在购买和使用本产品前，仔细阅读并同意本免责声明。

1.5 维护与保养说明

为保证 OmniHand 灵巧手长期稳定运行，请按照以下要求进行保养与存放。

如长期存放，建议将本产品存放在干燥、温度适宜、无强光、无腐蚀性气体、无强电磁干扰、远离火源和热源的环境中。

长期存放时，应断开电源及信号线，避免灵巧手长期处于受力、弯折、挤压或悬挂状态。建议使用原包装或防尘、防震包装进行保护。

在没有保护的情况下，严禁让 OmniHand 灵巧手接触液体（酒、水、饮料等）、沙尘和腐蚀物，且应避免雨淋。如不慎接触，请立即切断电源，及时检查接口、关节和外壳状态。

请勿直接用水冲洗、浸泡或喷洒清洁液。请勿使用酒精、强酸、强碱、腐蚀性清洁剂、挥发性溶剂或研磨性材料清洁灵巧手。可以使用干燥洁净的软布擦拭灵巧手的硬质外壳表面去除浮尘。

请定期进行维护和检查，确保机器的组件如电机、电源等正常运行，避免因老化或损坏导致的危险。

灵巧手发生故障后，请勿自行拆卸或强行继续使用。应记录故障现象、发生时间、使用环境、负载情况及异常提示，并及时联系售后维修人员。非授权维修、私自拆解、改装或使用非原厂配件造成的损坏，可能导致保修条款失效，相关风险由使用者自行承担。

2. 产品简介

2.1 产品概述

OmniHand 灵动款 2025 是一款紧凑型高自由度交互灵巧手，小身形，真百搭。

OmniHand 灵动款 2025 用途多样，支持手势交互、触摸交互、轻作业等场景应用，支持多

种常见交互手势，覆盖手心、手背和五指达 300+ 点位触觉感应。采用手感漆面，防夹手安全设计。轻巧灵动，适合小尺寸人形机器人(身高 110~130cm)。



2.2 主要特性

- **紧凑轻巧：**总长 180 mm、轻至 510g，尺寸百搭，各种型号人形机器人理想双手
- **灵动交互：**16 自由度，手势交互更拟人，常见交互手势全覆盖，独创手背触摸交互
- **安全亲和：**全手 300+ 触点感应力控，防夹手设计，交互更安全

2.3 产品基础信息

产品型号	OmniHand 灵动款	OmniHand 灵动触觉款
CE-EMC 认证情况	满足 EN IEC 61000-6-1/6-2/6-3/6-4	满足 EN IEC 61000-6-1/6-2/6-3/6-4
ROHS 认证情况	满足 RoHS 认证	满足 RoHS 认证

2.4 产品参数

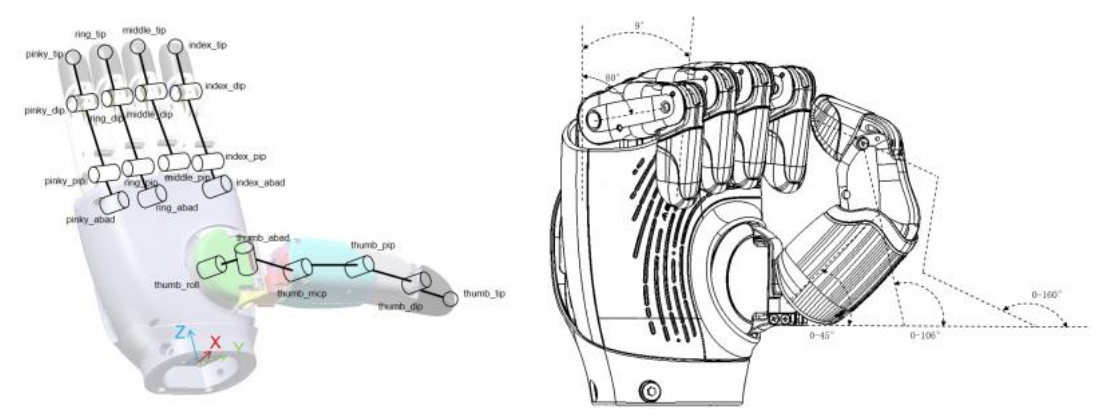
		OmniHand 灵动款	OmniHand 灵动触觉款
本体参数	重量（不含手腕板护盖及螺钉）	≤510g	≤520g
	尺寸	180*85*38.5mm	
	主动自由度	10（大拇指：3；食指：2；中指：1；无名指：2；小指：2）	
	总自由度	16（大拇指：5;食指：3；中指：2；无名指：3；小指：3）	
	最小张开/闭合时间	0.5s / 0.5s	
	指尖重复定位精度	0.3mm	
	负载能力	五指抓握 2kg; 整手负载 5kg	
	工作电压	18-27V	
	通讯接口	CAN-FD / RS485	
	工作温度范围	0~45℃	
	在线升级	支持 OTA 在线升级	
触觉传感器	力感知类型	/	指尖+手掌+手背：一维力
	阵列分辨率	/	0.1N
	感知范围	/	0~20N
	最大接受力(不损坏)	/	200N

2.4 关节角度范围

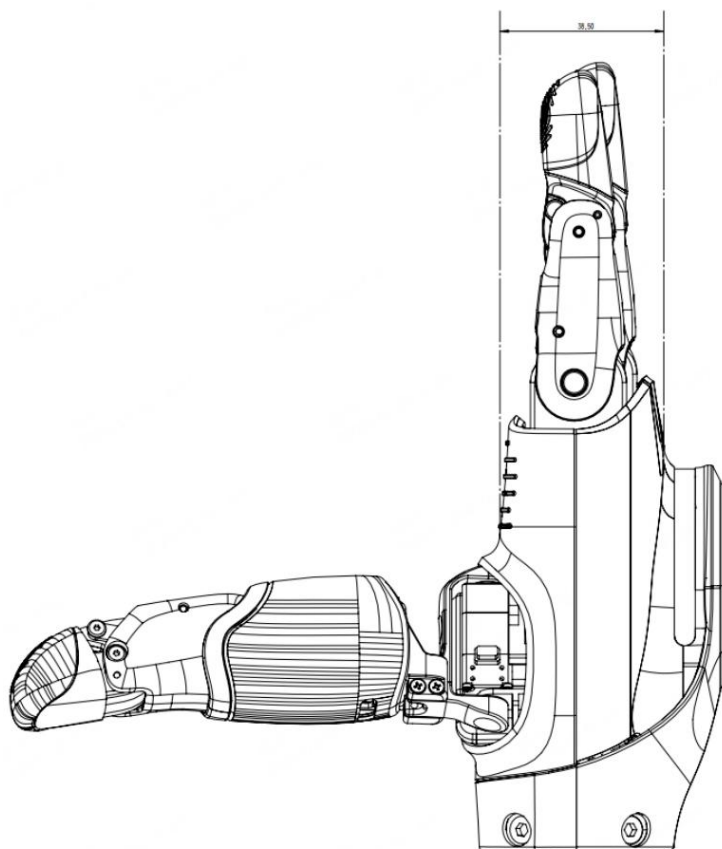
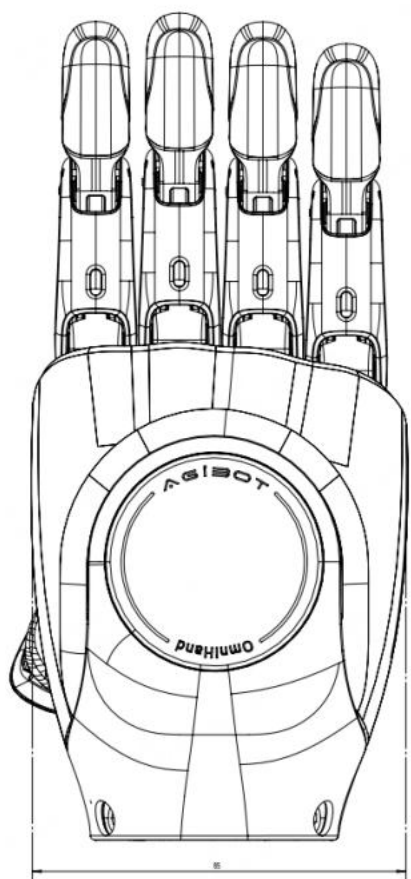
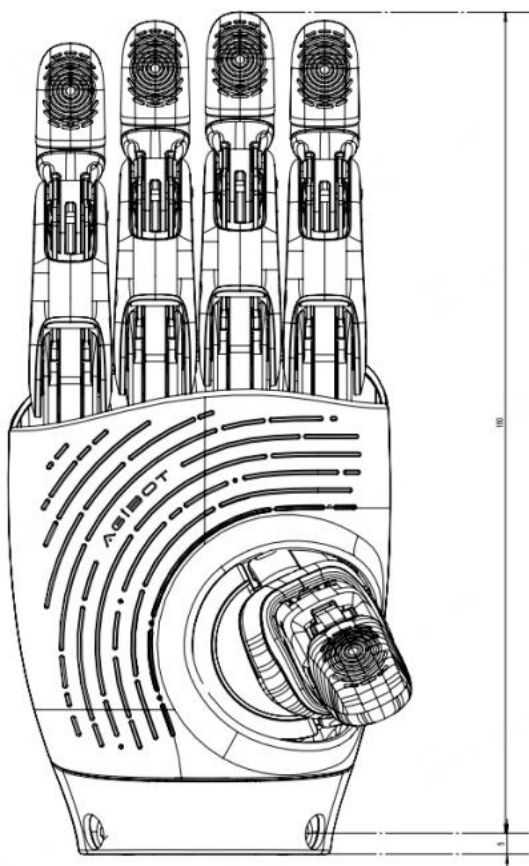
		右手		左手	
手指	关节名称	最小值 (度)	最大值 (度)	最小值 (度)	最大值 (度)
大拇指	thumb_roll_joint	-2	64	-64	2
	thumb_abad_joi nt	-94	3	-3	94
	thumb_mcp_joi nt	0	48	-48	0
	thumb_pip_joint	NA / 耦合关节，耦合关系参考 SDK			
	thumb_dip_joint	NA / 耦合关节，耦合关系参考 SDK			
食指	index_abad_join t	-9	0	0	9
	index_pip_joint	0	85	0	85
	index_dip_joint	NA / 耦合关节，耦合关系参考 SDK			
中指	middle_pip_joint	0	85	0	85
	middle_dip_joint	NA / 耦合关节，耦合关系参考 SDK			
无名指	ring_abad_joint	0	10	-10	0
	ring_pip_joint	NA / 耦合关节，耦合关系参考 SDK			
	ring_dip_joint	0	85	0	85

小指	pinky_abad_join t	0	11	-11	0
	pinky_pip_joint	0	85	0	85
	pinky_dip_joint	NA / 耦合关节，耦合关系参考 SDK			

- 以表格数值为准



2.5 产品尺寸



2.7 环保与废弃处理声明

本产品属于电子电器废弃物范畴，请依据 WEEE（废旧电子电气设备）指令相关要求。产品废弃时禁止与普通生活垃圾混合投放，请在产品报废后将其送至当地具备资质的电子废弃物回收网点，通过专业分类处置实现资源回收与无害化处理，降低对生态环境的潜在影响。

本产品严格遵循 RoHS 环保法规要求，产品各部件有害物质含量均符合法定限值标准。

2.8 生产厂家信息

厂家名称	上海临界点创新智能科技有限公司
厂家地址	上海市徐汇区天平路 185 号 11 层 1107 室
联系方式	021-20960883

3. 产品操作指南

本章节介绍了如何安装 OmniHand 灵巧手，以及电气和通讯接口的相关信息。

3.1 装箱清单

- OmniHand 灵动款 × 1
- (电源+CANFD 通讯线) × 1--电源线端子类型：XT30 母头
- RS485 通讯线 × 1
- USB Type-C 通讯线 × 1

- 产品合格证 × 1
- 快速使用指南 x1

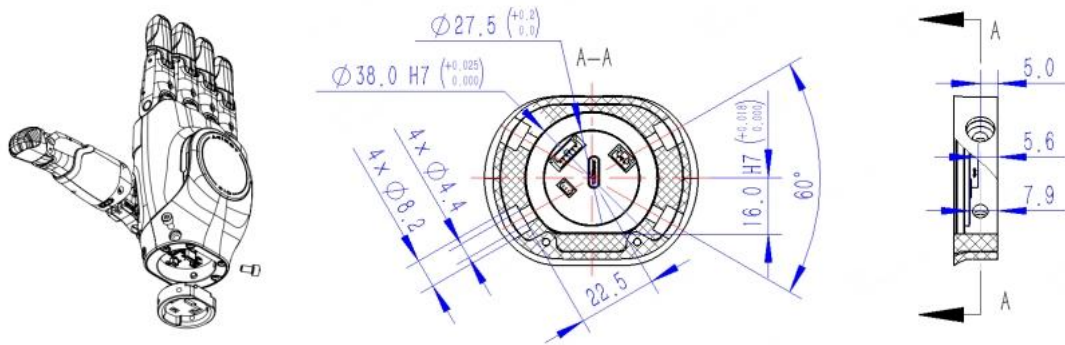
3.2 安装和拆卸指引

3.2.1 安装工具清单

序号	物料名称	数量	是否自带
1	OmniHand 灵巧手	1	是
2	超短头内六角螺栓 M4x8	4	否
3	手腕后端盖	1	是
4	H2 内六角扳手	1	否，需自备
5	RS485 通讯线	1	是
6	USB Type-C 通讯线	1	是
7	电源+CANFD 通讯线	1	是

3.2.2 安装指引

1. 拧掉手腕处 4 颗 M4 螺钉，拆掉手腕后端盖；
2. 连接通讯线和电源线（需要用户自备 24V 电源供电）；
3. 将灵巧手装到机器人手臂处（手臂机械接口设计参考下图），固定 步骤 1 拆下的 M4 螺钉，即完成安装。



3.2.3 拆卸指引

STEP1: 拧掉手腕处 4 颗 M4 螺钉，将灵巧手从机械臂手腕法兰（如有）处脱离；

STEP2: 拔掉通信线（RS485 或 USB Type-C ）和电源线；

STEP3: 将手腕后端盖安装到手腕处，拧紧 4 颗 M4×8 螺钉。

3.2.4 使用其他机械臂

● 转接法兰设计

参考上图和三维数模完成转接法兰设计，注意配合面仿形，螺钉孔位置和深度。需注意一般情况下，机械臂仅可以供电，不支持通讯。需注意机械臂末端是否有供电接口，如 6pin 8pin 接口（图），因此转接法兰设计时，需要留走线位置。（如图示）

转接法兰参考 - 机械臂末端法兰符合 ISO9409-1-50-4-M6 标准

[法兰文件](#)

● 连接机械臂

参考 3.2.2 拆卸手腕后端盖 连接灵巧手的供电及通讯线束， 然后连接灵巧手与转接法兰后，
再将转接法兰与机械臂连接在一起

3.3 接口及通信

3.3.1 电气和通讯接口 - 引脚定义

连接器采用 WAFER-GH1.25-6PLB (位号: U1), 支持 CAN-FD 。具体引脚定义如下:

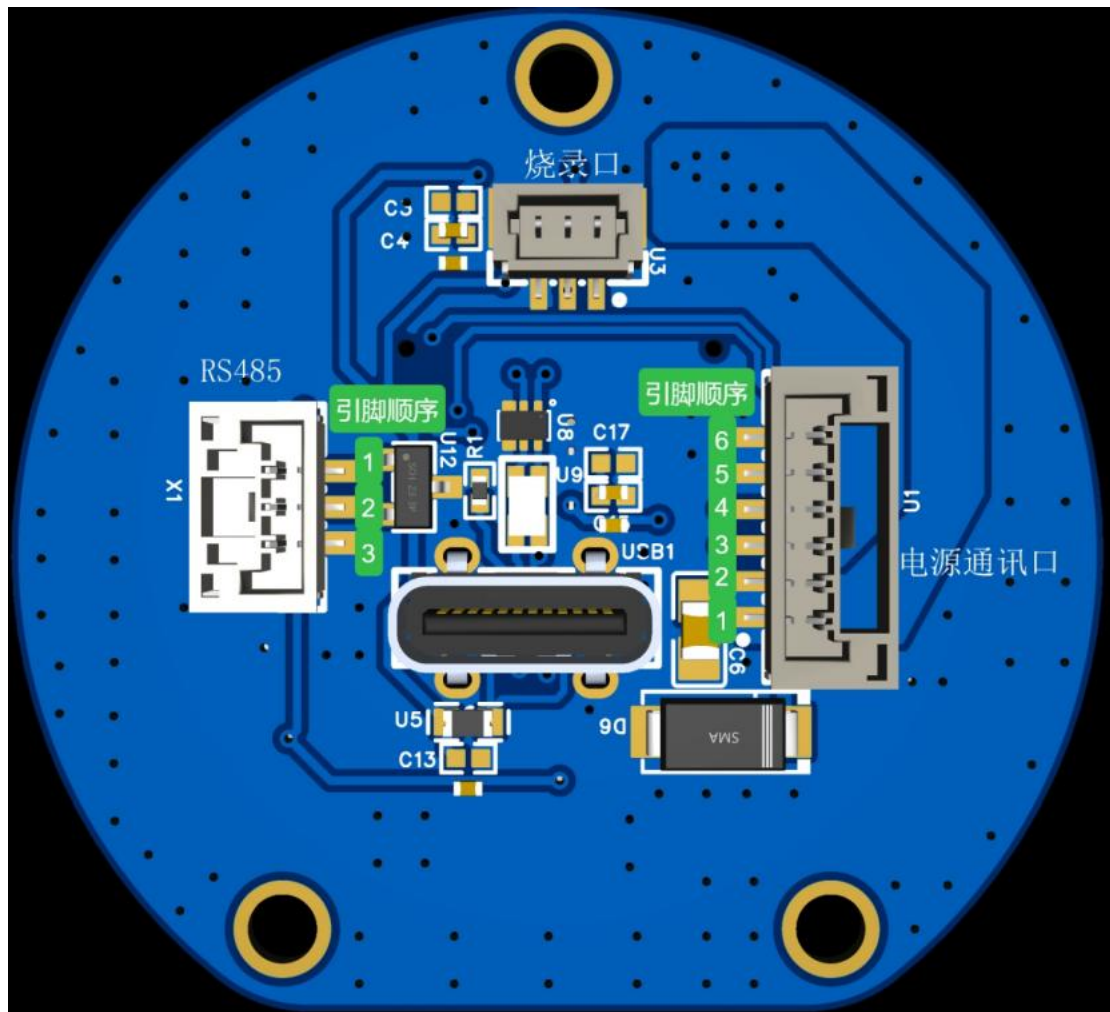
引脚	引脚定义
1	24V
2	24V
3	CAN_H
4	CAN_L
5	GND
6	GND

连接器采用 GH125-S03CCA-00 (位号: X1), 支持 RS485。具体引脚定义如下:

引脚	引脚定义
1	RS485_A
2	RS485_B
3	GND

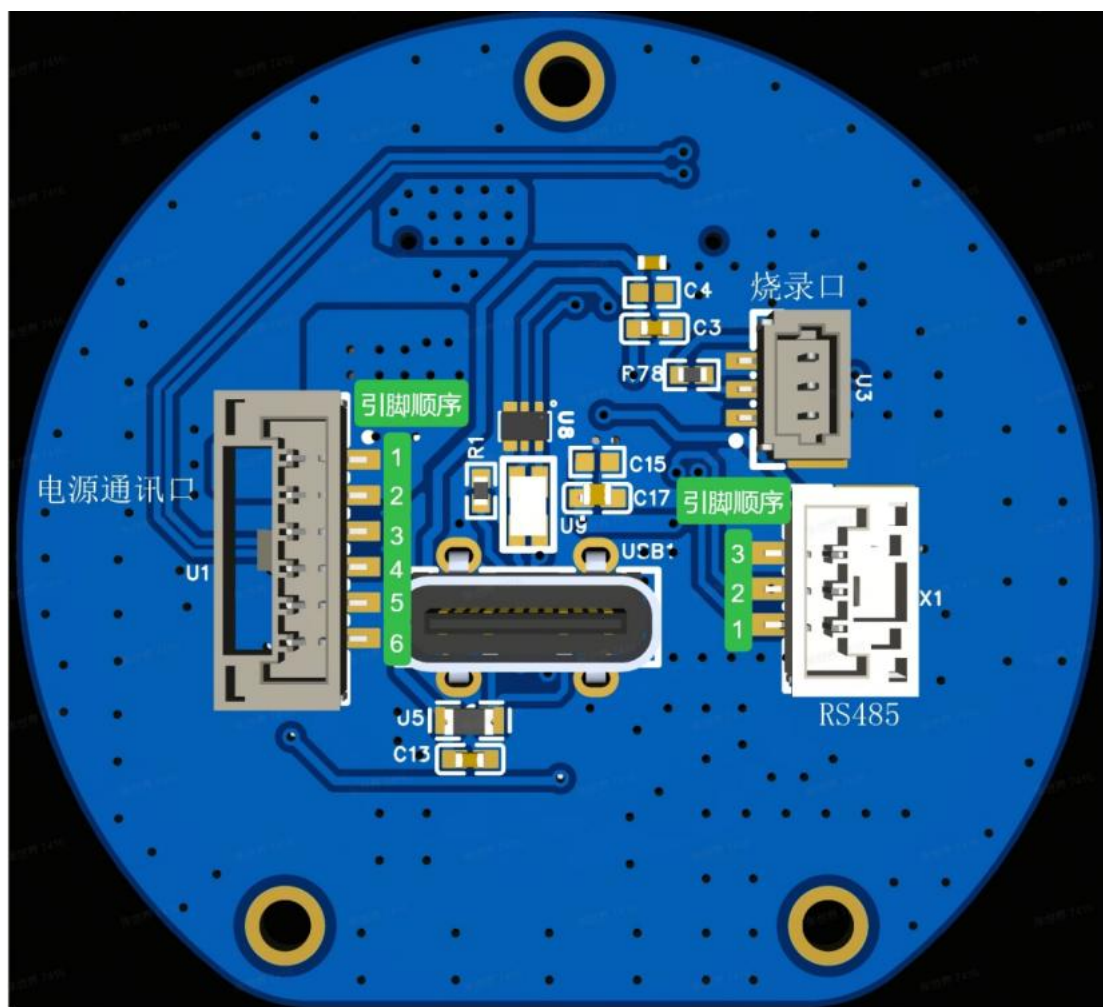
T 版本具体引脚顺序和对应引脚定义如下图 (判断标准: SN 码标记红色的 2 位小于 12 的都是

T 版本 AHB6RUUKX2509200038)



P 版本具体引脚顺序和对应引脚定义如下图（判断标准：SN 码标记红色的 2 位大于等于 12 的

都是 P 版本 AHB6RUUKX25¹²120035)



3.3.2 通信方式

支持用户通过 USB 通讯方式，连接上位机界面操作灵巧手；

支持基于 CAN-FD/RS485 通讯方式，进行灵巧手应用二次开发。

开发文档 请查阅 https://github.com/AgibotTech/agillink_omnihand_sdk/tree/main

3.4 常见故障排查

常见故障	解决措施
指套软胶和手掌软胶磨损	更换软胶

常见故障	解决措施
螺丝拆卸再拧紧后松脱	每个螺丝拆卸后不再二次使用，更换新螺丝
本体硬件排障	禁止客户自行拆解或更换配件，如遇本体硬件类问题请联系售后人员。
SDK 排障	使用 SDK 驱动 OmniHand 设备常见问题解决办法详见 https://github.com/AgibotTech/agillink_omnihand_sdk/blob/main/doc/zh_cn/TROUBLESHOOTING.md
上位机排障	固件升级失败等上位机类故障排查，请参照 5. 上位机使用说明。

3.5 操作类注意事项

- 配件更换：除手腕后端盖，禁止客户自行拆解或更换配件，如遇问题请联系售后人员。
- 调试、使用、启停等其他操作：可配合上位机实现，操作指南详见 5.上位机使用说明。
- 禁忌操作：禁止在灵巧手达到限位点时掰动手指，禁止在灵巧手上电状态下掰动手指。

4. 通讯协议说明

4.1 CAN-FD 通信帧格式

CAN-FD 通信参数：仲裁域波特率 1M（80%采样点），数据域波特率 5M（75%采样点），本设备仅支持 CAN-FD 帧格式。对于经典 CAN2.0 帧，设备将：

- 不解析数据
- 不返回应答
- 不产生错误帧

以下为 CAN-FD 数据域中的通信帧布局：

ID	DLC	D0	D1 - D63
节点 ID	1-64	CMD	数据段

- 节点 ID：0 ~ 0x7FF，标准帧。默认 ID 为 0x09，当节点 ID 未知时，可以利用寻呼地址 **0x7FF** 来查询节点 ID（任意指令即可，比如查询使能状态、查询版本号、查询序列号等），节点在收到此 ID 数据后依然按照自身 ID 进行回复。

- CMD：控制指令，占据数据域中的首字节，具体定义参见 4.3 通信协议内容
- 数据段：与控制指令对应的数据，具体定义参见 4.3 通信协议内容 上下行数据定义，所有数据按照小端排布。
- 指令样例：（十六进制格式）

- A. 使能指令：0x09 0x02 0x01 0x01
- B. 失能指令：0x09 0x02 0x01 0x00
- C. 设置 1 号电机到 2000 位置： 0x09 0x04 0x06 0x01 0x00 0x7D
- D. 设置所有电机到 2000 位置：0x09 0x18 0x08 0x00 0x7D 0x00 0x7D 0x00 0x7D 0x00

0x7D 0x00 0x7D 0x00 0x7D 0x00 0x7D 0x00 0x7D 0x00 0x7D 0x00 0x7D

4.2 USB/串口 通信帧格式

使用 USB 连接时，驱动板会虚拟出一个串口，串口参数设置要求如下。

串口通信参数设置：

- 波特率：460800
- 字节总数：8
- 校验位：无
- 停止位：1

字段	帧头	ID	数据长度	数据段（CMD+DATA）	crc
字节占用	2	2	1	N	2

- 帧头 0xAAEE，小端模式。
- ID: 设备节点 ID，0~0x7FF。默认 ID 为 0x01，0x7FF 为广播地址，各节点收到广播数据后会使用自身 ID 回复应答帧。当节点 ID 未知时，可以利用广播地址查询节点 ID（任意指令即可，比如查询使能状态、查询版本号、查询序列号等），小端模式，节点在收到此 ID 数据后依然按照自身 ID 进行回复。
- 数据长度：数据段字节长度
- 数据段：长度为 1-64，首字节为控制指令，其后跟随对应数据，与 CAN-FD 中的数据域保持一致，小端模式
- crc 校验说明：数据校验，校验范围为**从帧头至数据段结束**，计算方法如下。预留

0x5555 为调试专用 crc，可直接通过校验，小端模式。

- crc 计算算法说明：

- 初始值：0x0000
- 多项式：0x1021
- 异或值：0x0000
- 数据反转：不反转，按 msb_first 位序

- 指令样例：(十六进制格式)

A. 使能指令：0xEE 0xAA 0x01 0x00 0x02 0x01 0x01 0x60 0xAA

B. 失能指令：0xEE 0xAA 0x01 0x00 0x02 0x01 0x00 0x41 0xBA

C. 设置 1 号电机到 2000 位置： 0xEE 0xAA 0x01 0x00 0x04 0x06 0x01 0x00 0xD7 0x21
0xB9

D. 设置所有电机到 2000 位置：0xEE 0xAA 0x01 0x00 0x15 0x08 0x00 0xD7 0x00 0xD7
0x00 0xD7 0x00 0xD7 0x00 0xD7 0x00 0xD7 0x00 0xD7 0x00 0xD7 0x00 0xD7 0x00
0xD7 0x9B 0x65

```
C
/*
 * @Author: richie.li
 * @Date: 2023-08-21 21:46:17
 * @LastEditors: richie.li
 * @LastEditTime: 2023-12-26 17:07:10
 */

/* Private includes -----*/
```

```

#include "crc.h"

/* Private types -----*/
/* Private macro -----*/
/* Private constants -----*/
/* Private variables -----*/

static const uint16_t crc16Tab[256] = {
    0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50a5, 0x60c6, 0x70e7, 0x8108, 0x9129, 0xa14a, 0xb16b,
    0xc18c, 0xd1ad, 0xe1ce, 0xf1ef, 0x1231, 0x0210, 0x3273, 0x2252, 0x52b5, 0x4294, 0x72f7, 0x62d6,
    0x9339, 0x8318, 0xb37b, 0xa35a, 0xd3bd, 0xc39c, 0xf3ff, 0xe3de, 0x2462, 0x3443, 0x0420, 0x1401,
    0x64e6, 0x74c7, 0x44a4, 0x5485, 0xa56a, 0xb54b, 0x8528, 0x9509, 0xe5ee, 0xf5cf, 0xc5ac, 0xd58d,
    0x3653, 0x2672, 0x1611, 0x0630, 0x76d7, 0x66f6, 0x5695, 0x46b4, 0xb75b, 0xa77a, 0x9719, 0x8738,
    0xf7df, 0xe7fe, 0xd79d, 0xc7bc, 0x48c4, 0x58e5, 0x6886, 0x78a7, 0x0840, 0x1861, 0x2802, 0x3823,
    0xc9cc, 0xd9ed, 0xe98e, 0xf9af, 0x8948, 0x9969, 0xa90a, 0xb92b, 0x5af5, 0x4ad4, 0x7ab7, 0x6a96,
    0x1a71, 0x0a50, 0x3a33, 0x2a12, 0xdbfd, 0xcdbdc, 0xfbbf, 0xeb9e, 0x9b79, 0x8b58, 0xbb3b, 0xab1a,
    0x6ca6, 0x7c87, 0x4ce4, 0x5cc5, 0x2c22, 0x3c03, 0x0c60, 0x1c41, 0xedae, 0xfdf, 0xcdec, 0xddcd,
    0xad2a, 0xbd0b, 0x8d68, 0x9d49, 0x7e97, 0x6eb6, 0x5ed5, 0x4ef4, 0x3e13, 0x2e32, 0x1e51, 0x0e70,
    0xff9f, 0xefbe, 0xdfdd, 0xcffc, 0xbfb, 0xaf3a, 0x9f59, 0x8f78, 0x9188, 0x81a9, 0xb1ca, 0xa1eb,
    0xd10c, 0xc12d, 0xf14e, 0xe16f, 0x1080, 0x00a1, 0x30c2, 0x20e3, 0x5004, 0x4025, 0x7046, 0x6067,
    0x83b9, 0x9398, 0xa3fb, 0xb3da, 0xc33d, 0xd31c, 0xe37f, 0xf35e, 0x02b1, 0x1290, 0x22f3, 0x32d2,
    0x4235, 0x5214, 0x6277, 0x7256, 0xb5ea, 0xa5cb, 0x95a8, 0x8589, 0xf56e, 0xe54f, 0xd52c, 0xc50d,
    0x34e2, 0x24c3, 0x14a0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405, 0xa7db, 0xb7fa, 0x8799, 0x97b8,
    0xe75f, 0xf77e, 0xc71d, 0xd73c, 0x26d3, 0x36f2, 0x0691, 0x16b0, 0x6657, 0x7676, 0x4615, 0x5634,
    0xd94c, 0xc96d, 0xf90e, 0xe92f, 0x99c8, 0x89e9, 0xb98a, 0xa9ab, 0x5844, 0x4865, 0x7806, 0x6827,
    0x18c0, 0x08e1, 0x3882, 0x28a3, 0xcb7d, 0xdb5c, 0xeb3f, 0xfb1e, 0x8bf9, 0x9bd8, 0xabbb, 0xbb9a,
    0x4a75, 0x5a54, 0x6a37, 0x7a16, 0x0af1, 0x1ad0, 0x2ab3, 0x3a92, 0xfd2e, 0xed0f, 0xdd6c, 0xcd4d,
    0xbdaa, 0xad8b, 0x9de8, 0x8dc9, 0x7c26, 0x6c07, 0x5c64, 0x4c45, 0x3ca2, 0x2c83, 0x1ce0, 0x0cc1,
    0xef1f, 0xff3e, 0xcf5d, 0xdf7c, 0xaf9b, 0xbfba, 0x8fd9, 0x9ff8, 0x6e17, 0x7e36, 0x4e55, 0x5e74,
    0x2e93, 0x3eb2, 0x0ed1, 0x1ef0};

/* Private function prototypes -----*/
/* Private user code -----*/

uint16_t Crc16(const uint8_t* buf, uint32_t len) {
    uint16_t crc = 0;
    while (0 < len--) {
        crc = (crc << 8) ^ crc16Tab[((crc >> 8) ^ *buf++) & 0x00FF];
    }
}

```

```
return crc;  
}
```

4.3 通信协议内容

手内部的主动轴按照序号进行分布，索引为 1-10（如下图：红色）；触觉传感器序号从拇指开始依次分布，索引为 1-7（如下图：黑色）。



发送和回复的命令字段相同。

命令	定义	下行数据	上行数据	备注
0x01	设备使能或失能	1 字节 00: 失能 01: 使能 02: 校准模式	1 字节 00: 失败 01: 成功	
0x02	查询当前设备使能状态		1 字节	

			00: 失能 01: 使能 02: 校准模式	
0x06	设置单个轴位置	3 字节 d0: 轴索引, 对应电机 ID d1-d2: 目标位置, uint16 类型, 范围 0 - 4096; 将整个角度运行范围分为 4096 份, 比如手指运行 $\alpha=0.0^{\circ}\sim90.0^{\circ}$, 目标位置 = $4096 * \alpha / 3600$, 角度精确到 0.1°	3 字节 d0: 轴索引, 对应电机 ID d1-d2: 当前位置, uint16 类型, 范围 0 - 4096; 将整个角度运行范围分为 4096 份, 比如手指运行 $\alpha=0.0^{\circ}\sim90.0^{\circ}$, 目标位置 = $4096 * \alpha / 3600$, 角度精确到 0.1°	
0x07	读取单个轴位置	1 字节, 轴索引 范围 1-10	同 0x06 指令回复	
0x08	设置全部轴位置数据	20 字节 d0-d19, 所有轴目标位置数据 每个轴两字节位置数据, uint16 类型, 从 1 号轴开始依次排布, 每个轴位置范围 0 - 4096; 将整个角度运行范围分为 4096 份, 比如手指运行 $\alpha=0.0^{\circ}\sim90.0^{\circ}$, 目标位置 = $4096 * \alpha / 3600$, 角度精确到 0.1°	60 字节 d0-d19, 所有轴当前位置数据 每个轴两字节位置数据, uint16 类型, 从 1 号轴开始依次排布, 每个轴位置范围 0 - 4096 d20-d39: 所有轴速度反馈; d40-d49: 所有轴力矩反馈; d50-d59: 所有轴的故障状态;	

		d20-d23: 0, 预留自由度;		
0x09	读取全部轴位置数据		20 字节 d0-d19, 所有轴当前位置数据 每个轴两字节位置数据, uint16 类型, 从 1 号轴开始依次排布, 每个轴位置范围 0 - 4096	
0x0A	读取所有轴电流信息		20 字节 d0-d19, 所有轴当前电流数据, 单位: 0.01A; 每个轴两字节电流数据, uint16 类型, 从 1 号轴开始依次排布;	
0x0B	读取所有轴速度信息		20 字节 d0-d23, 所有轴当前速度数据 每个轴两字节速度数据, uint16 类型, 从 1 号轴开始依次排布;	
0x0C	读取全部轴温度数据		10 字节 d0-d9,代表 10 个轴的当前温度数据 每个轴 1 字节温度数据, int8 类型, 从 1 号轴开始依次排布, 单位为 度	
0x0D	读取错误码		2 字节, uint16 类型 0x0000: 无错误 1-10: 电机 1-10 堵转	

			21-30: 电机 1-10 过温 41-50: 电机 1-10 电流异常 61-70: 电机 1-10 编码器异常 81-90: 电机 1-10 通讯异常 101: 通讯异常 102: 电机初始化故障 103: 触觉传感器初始化故障 104: 自动校准故障 105: 参数错误	
0x0E	清除错误		1 字节 00: 失败 01: 成功	
0x11	读取单个指尖传感器的数据	1 字节，传感器索引 范围：0-6	17 字节（手指）/26 字节（手心手背） d0，代表传感器索引 d1-d16: 手指传感器的数据，4*4 点，每点 uint8，按行读取； d1-d25: 手心/手背传感器数据，5*5 点，每点 uint8，先按行再按列读取；	
0x12	读取所有指尖传感器的数据 1		48 字节，顺序为： 大拇指 4*4 + 食指 4*4 + 中指 4*4 每点 uint8，每个传感器先按行再按列读取；	
0x13	读取所有指尖传感器的数据 2		32 字节，顺序为：	

			无名指 4*4 + 小拇指 4*4 每点 uint8，每个传感器先按列再按行读取；	
0x14	读取所有指尖传感器的数据 3		50 字节，顺序为： 手心 5*5 + 手背 5*5 每点 uint8，每个传感器先按行再按列读取；	
0x32	设置全部轴位置（力位混合模式）	50 字节 d0-d19，所有轴目标位置数据 每个轴两字节位置数据，uint16 类型，从 1 号轴开始依次排布，每个轴位置范围 0 - 4096；将整个角度运行范围分为 4096 份，比如手指运行 $\alpha=0.0^{\circ}\sim 90.0^{\circ}$ ，目标位置 = $4096 * \alpha / 3600$ ，角度精确到 0.1° d20-d39：所有轴速度；每个轴两字节位置数据，uint16 类型，从 1 号轴开始依次排布，每个轴速度范围 -4095 - 4096；电机最大速度为 4096。 d40-d49：所有轴力矩；每个轴 1 字节，力矩范围 0~255；	60 字节 d0-d19，所有轴当前位置数据 每个轴两字节位置数据，uint16 类型，从 1 号轴开始依次排布，每个轴位置范围 0 - 4096 d20-d39：所有轴速度反馈； d40-d49：所有轴力矩反馈； d50-d59：所有轴的故障状态；	

0x80	控制源	1 字节，控制源设定： 0：机器人本体控制； (默认) 1：人机界面操作员控制。	返回：原数据	
0x81	控制源查询	0 字节	返回：控制源状态 0：机器人本体控制； 1：人机界面操作员控制。	
0xC2	读取产品系列号		19 字节： d0~d2：3 位供应商暗码 d3~d8：6 位物料暗码 d9~d14：YYMMDD，年月日 d15~d18：4 位流水号	
0xCD	查询设备型号以及软硬件版本		8 字节 d0：设备类型 2：O10 灵巧手 d1~d2：产品状态（ASCII 码， 如 T1、T2、P1 等） d3~d5：软件版本 (d1.d2.d3) d6~d8：硬件版本 (d4.d5.d6) d9：自由度	兼容 OmniPicker 指令

说明：上下行数据介绍中，d 代表 data，d0 即为第一个字节。

4.4 固件/上位机版本

固件和上位机最新版本可以从官网（<https://www.agibot.com.cn/DOCS/OS/Omnihand->

O10) 上下载。

4.5 控制注意事项

机器人本体控制灵巧手，建议使用位置+力矩的复合模式，灵巧手自适应调整力矩对物体进行抓握。**建议不要频繁抓握，抓取间隔时间建议 $\geq 2s$ 较为适宜。**

4.6 指示灯功能说明

当手背灯显示蓝色，表明整机正在初始化；蓝灯变为绿灯，表明初始化结束；当电压过低或者过高时，手背灯无法从蓝色变为绿色，请注意及时检查电压是否满足预设值电压 $24V \pm 10\%$ 的要求。

5. 上位机使用说明

5.1 运行环境

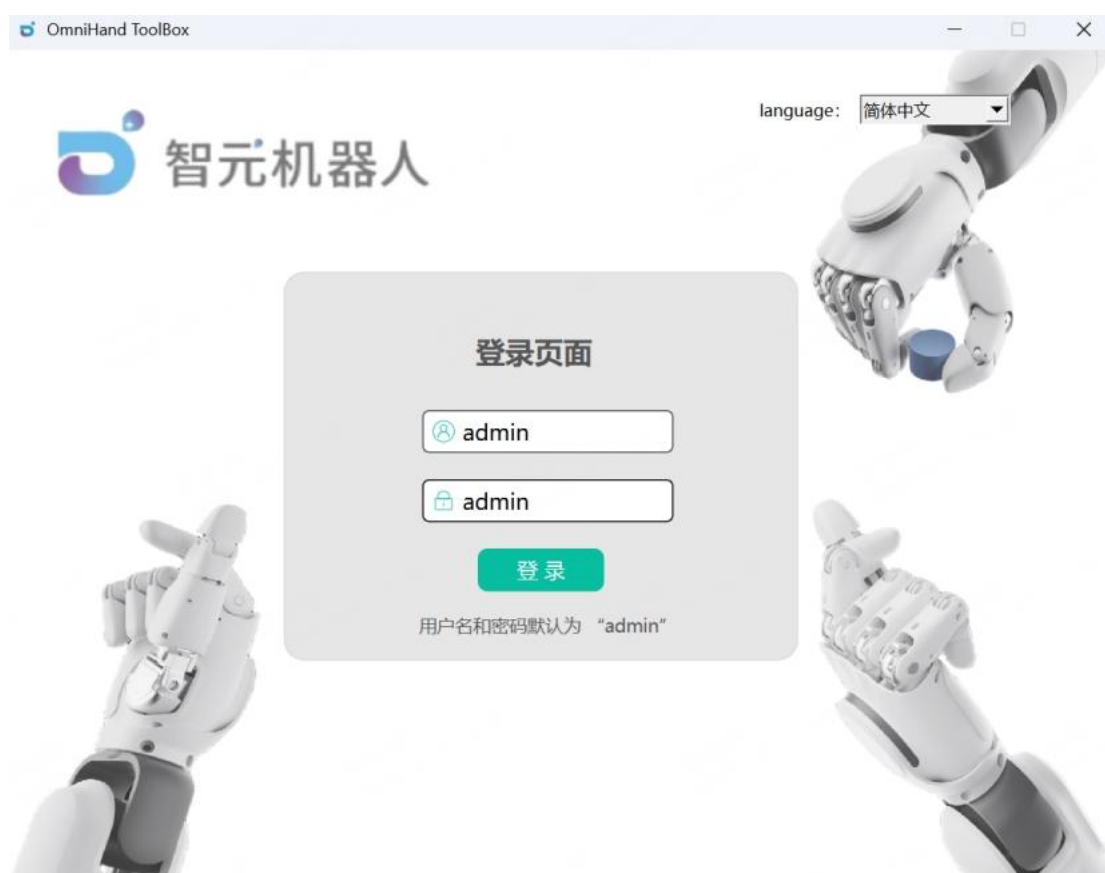
- CPU: Intel Core i5-8 代以上，双核 1.5GHz+
- 内存: 8GB+
- 显示: 支持 OpenGL 3.3+ (集成显卡即可)
- 操作系统: Windows 10/11
- 接口: USB 2.0+
- 软件依赖: VC++ 2015 Redistributable / https://aka.ms/vc14/vc_redist.x64.exe(x64 版本), .Net Framework 4.8

5.2 下载安装

灵巧手上位机软件可在官方网站进行下载压缩包，链接见 4.4。

下载完成后，需要进行解压。安装提示完成后，双击 OmniHand-ToolBox 运行上位机软件。

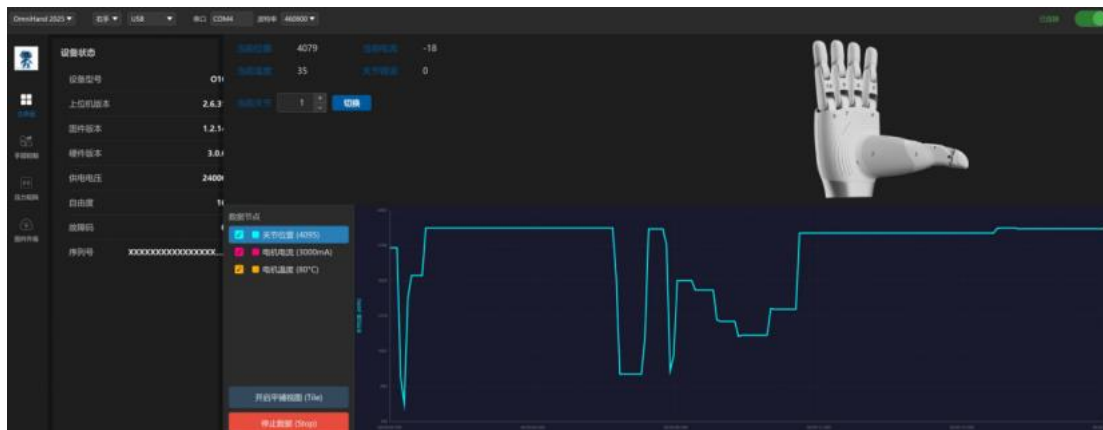
5.3 登录界面



点击“登录”按钮直接进入用户主界面，用户名和密码默认为“admin”

页面右上角可调整界面语言，目前支持中文，英文。

5.4 主界面（上位机版本 2.6.33）



OmniHand 灵巧手在接通电源后，使用 USB 线束与主控电脑进行连接，正确接线后，上位机软件会自动搜索端口号，选择正确的端口号并在左边栏的端口号中进行显示。

灵巧手与主控器的波特率固定为 460800，目前不支持进行波特率的调整。

滑动连接，等待握手成功。当电脑成功与灵巧手连接后，连接状态显示为“**已连接**”，“状态栏”会显示设备基础信息。

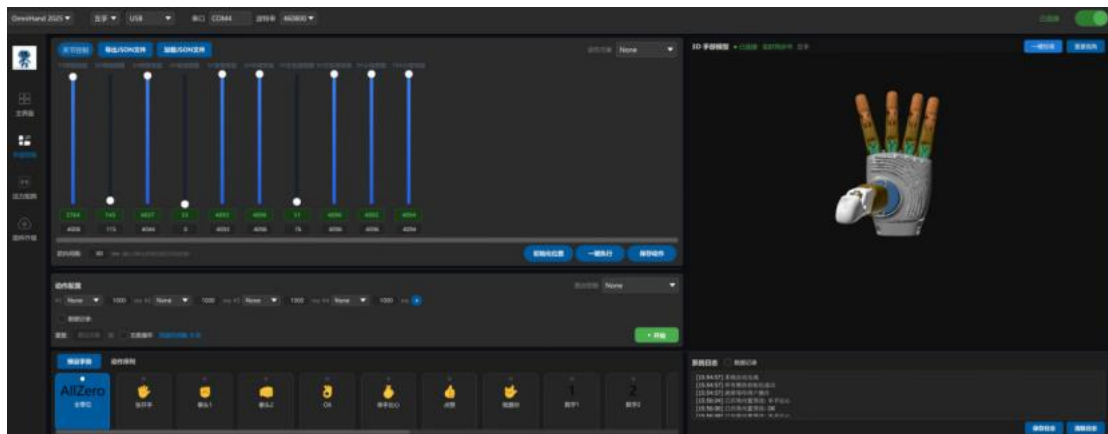
在主界面还可查看关节的详细信息，选择想查看的关节，可查看该关节的位置量程，力矩，转速等信息，在下方还可以查看实时的电流及温度等。

错误处理

- 若显示“**已连接**”，但是**无法操作灵巧手的运动**，请检查“端口号”中是否有其他端口，请重新选择端口号后进行连接。
- 如果发生连接失败，请确认软件左上角的“端口号”框内是否能检测到端口：
 - 若**已经显示端口号**，但是连接状态**显示未连接**，请将上位机软件关闭重新尝试。
 - 若**未找到端口号**，请从以下几个方面进行检查：
 - 检查电脑是否安装 USB 设备驱动，是否需要更新 USB 设备驱动

- 检查设备是否正在正常供电
- 检查是否在使用拓展坞进行连接（有可能会出现问题，请移除）
- 检查线束是否接触不良，USB 数据线需要支持数据传输功能。

5.5 手部控制



在手部控制面板，可对灵巧手进行操作控制。

动作面板中，**#1~#10** 分别对应 10 个关节，该款灵巧手支持 10 个关节控制，每个关节对应的位置显示在滑动条上方。其中大拇指有三个自由度，食指、无名指和小指各有两个自由度（侧摆和弯曲），中指只有一个自由度（侧摆不支持）。

若对手的姿态进行操作，用户可在每个关节的框内，通过拉动滑轨或输入具体数值进行关节控制，数值范围为 0~4096。若使用输入框进行控制，输入数据后，点击 **回车键** 控制生效，同时滑动条下方会显示关节的具体数值。绿色为当前实际关节位置参数，此处无法输入数值。右侧的灵巧手 3D 模型，将实时同步灵巧手的手势变化。

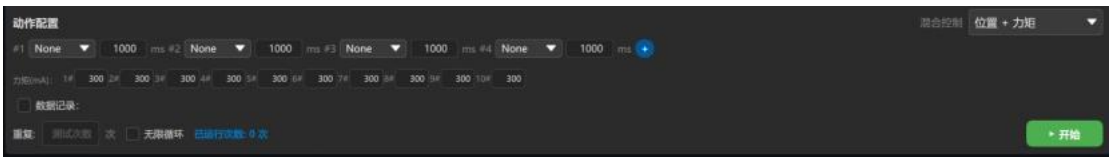
若想一次性调节所有关节数值，可在每个关节对应的输入框里输入目标数值，点击一键执行，每个关节都将运动到目标位置。

回到初始位置可以让所有关节一次性回到初始位置。

保存动作，将目前动作面板上所有关节自由度的数值保存为一个自定义动作，点击保存，输入动作名称，保存成功后，可以在动作列表中找到。

在预设手势栏，可以选择预设好的手势，点击即执行。切换到动作序列，可以便捷地编辑动作序列，调整动作顺序等。

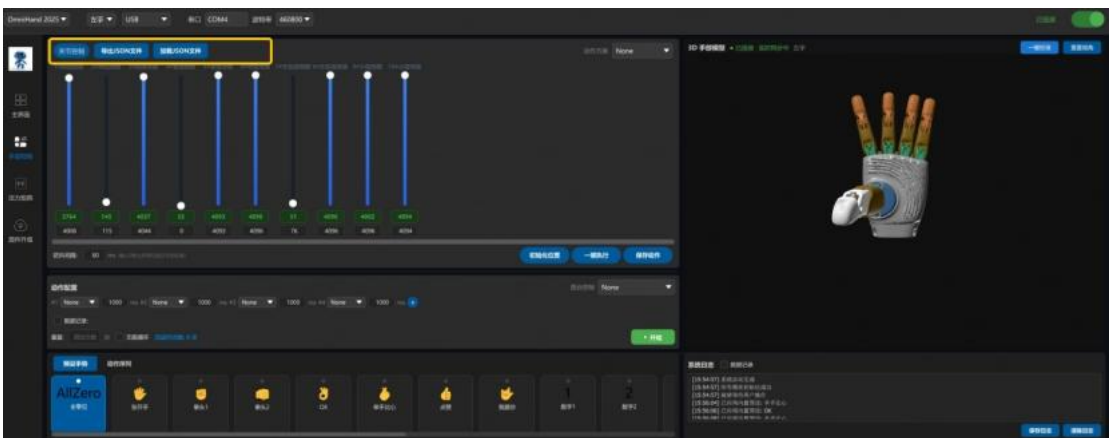
5.5.1 自定义动作播放/循环



在动作配置面板，可以选择已在动作列表中的动作，组成一组动作。一组最多可选择四个动作，选择好动作后，可在后面对应的框内输入动作间隔的时间。此处可以调节控制模式（位置+力矩，位置+速度+力矩）来方便地调节动作的执行效果。

在重复次数输入框，输入希望重复此组动作的次数，点击开始，系统将按序执行动作，并将执行情况记录到系统日志。点击停止将终止进程

5.5.2 保存和加载自定义动作文件



动作列表中的自定义动作，可以导出为动作文件，以便下次打开上位机时继续使用，点击导出

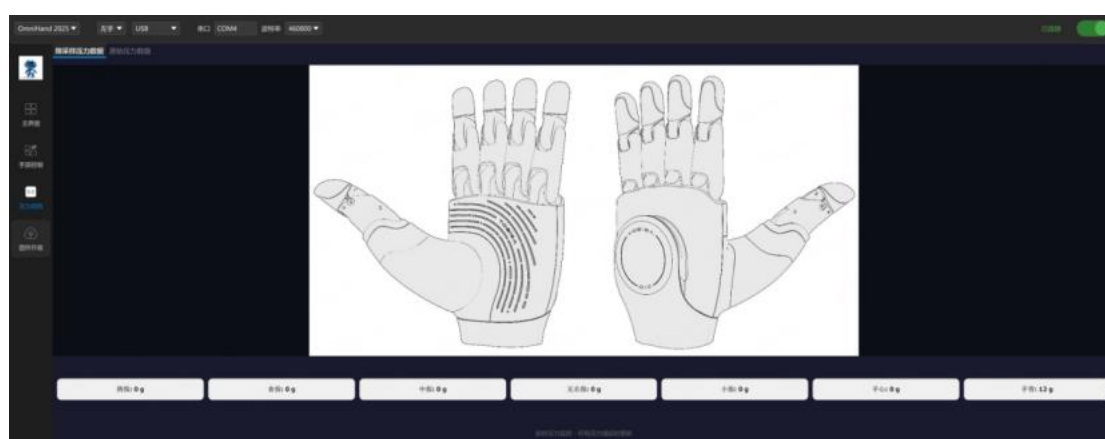
JSON 文件，可将动作列表中的所有动作一次性导出。

下次使用时，点击加载 JSON 文件，选择本地文件后，所有动作可导入动作列表。

对于手动添加动作格式，需要添加一个 JSON 对象，每个对象之间用 `,` 分隔，对象内容放在 `{}` 大括号里面。大括号里面需要填写 `name` 字段，为动作名称；`pos_axis_x` 字段是用户期望关节到达的位置，目前 `x` 只需要 1-10 即可。

```
{  
  "name": "左手半张",  
  "pos_axis_1": 4067,  
  "pos_axis_10": 4090,  
  "pos_axis_11": 0,  
  "pos_axis_12": 0,  
  "pos_axis_2": 3,  
  "pos_axis_3": 3996,  
  "pos_axis_4": 4096,  
  "pos_axis_5": 4090,  
  "pos_axis_6": 4093,  
  "pos_axis_7": 4083,  
  "pos_axis_8": 4088,  
  "pos_axis_9": 18  
},
```

5.6 压力矩阵

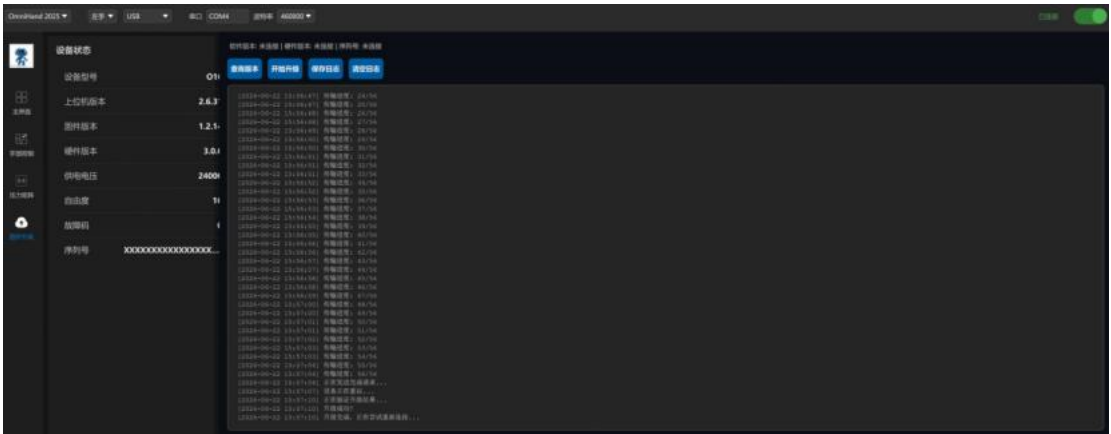


在压力矩阵面板，可以查看五个手指指尖、手心和手背的压力情况，其中在每个手指的指尖的压力测点合成了 3 个点，手心 9 个压力测点，手背 5 个压力测点。

下方显示每个区域的总压力。

原始压力数据仅支持通过 CAN-FD 连接方式，不支持 USB-C 连接方式。

5.7 固件升级



在固件升级面板，点击查询版本号按钮，可以查询当前设备的固件版本号等信息。

点击开始升级，选择下载好的新固件文件 (*.bin)，可以为灵巧手进行固件升级。

请先确认设备在已连接状态后再开始升级，升级开始后请耐心等待升级完成，升级过程通过日志打印，通过查看日志可以确定升级是否成功，成功后会显示 **升级成功** 等字样。

如果显示 **升级失败**，请再次尝试升级，点击 **开始升级** 然后重新操作一遍。若仍然失败，请查看日志中的提示，若出现连接失败，可能需要使用 ST-link 更新 MCU 固件后再重试。

本手册基于 OmniHand 灵动款 硬件 Ver. 3.0.0，固件 Ver. 1.2.12，上位机软件 Ver. 2.6.33 编写。

其他产品手册中未注明事项，请查阅官网，或联系客服团队获取信息。