

**ROKAE** 珞石

# 六维力传感器力控 使用手册



人类生产生活的得力伙伴







六维力传感器力控  
使用手册

控制系统版本: V3.1.1  
[备注]  
文档版本: [状态]

本手册中记载的内容如有变更，恕不事先通告。本公司对手册中可能出现的错误均不承担任何责任。

本公司对因使用本手册及其中所述产品而引起的意外或间接伤害均不承担任何责任，敬请谅解。

本公司不可能预见所有的危险和后果，因此本手册不能警告用户所有可能的危险。

禁止擅自复印或转载本手册的部分或全部内容。

如您发现本手册的内容有误或需要改进抑或补充之处，请不吝指正。

本手册的原始语言为中文，所有其他语言版本均翻译自中文版本。

©版权所有 2015-2024 ROKAE 保留所有权利  
珞石（山东）机器人集团有限公司  
中国·山东

## 目录

1 导纳力控概述.....	1
1.1 功能概述.....	1
1.2 导纳力控所支持的运动.....	1
1.2.1 传感器引导的运动.....	1
1.2.2 叠加运动指令的运动.....	1
1.3 使用场景与限制.....	1
2 导纳力控平台.....	2
2.1 硬件组成.....	2
2.2 导纳软件界面.....	3
2.2.1 开启导纳.....	3
2.2.2 传感器型号.....	3
2.2.3 截止频率.....	3
2.2.4 传感器安装位置.....	4
3 导纳指令.....	5
3.1 FcInitE.....	5
3.2 FcStartE.....	6
3.3 FcStopE.....	6
3.4 FcSetParaE.....	7
3.5 FcRefForceE.....	8
3.6 FcResetBiasE.....	9
3.7 FcCondForceE.....	10
3.8 FcCondWaitWhileE.....	11
3.9 FcSupvForceE.....	12
3.10 FcSupvTCPSpeedE.....	13
3.11 FcSupvPosBoxE.....	14
4 示例程序.....	15

## 1 导纳力控概述

### 1.1 功能概述

六维力传感器力控又称为导纳力控，是通过外接六维力传感器来获取与环境间的接触力（FX，FY，FZ，MZ，MY，MX），从而改变机器人的位姿达到力控效果。开启导纳力控时可实现 6 个自由度（X,Y,Z,A,B,C）的力控效果，配合设置的期望力和力控参数，可使工业机器人实现恒力跟踪的效果。目前导纳力控通过六维力传感器力控的工艺包搭配导纳力控指令实现，主要包括以下功能：

- 根据六维力传感器反馈的力/力矩进行传感器引导运动。
- 根据示教的轨迹叠加非运动方向的力控运动。
- 监控运动过程中的力、速度、位置，使其限定在一定范围内。

### 1.2 导纳力控所支持的运动

#### 1.2.1 传感器引导的运动

指机器人运动轨迹完全由实时力反馈生成，如果机器人执行传感器引导的运动，则在 RL 程序中不需要用运动指令，机器人会朝着设定的力控方向移动，直到满足配置的力/力矩控制参数。力和力矩可以设置 6 个自由度（Fx, Fy, Fz, Mx, My, Mz），力控参考坐标系可选用工具或工件坐标系。

#### 1.2.2 叠加运动指令的运动

机器人通过 MoveL 或 MoveAbsJ 等运动指令沿着编程的路径移动。机器人在非运动方向额外叠加位置补偿量，以达到配置的力/力矩控制参数。力和力矩设定值可设置 6 个自由度（Fx, Fy, Fz, Mx, My, Mz）。力控参考坐标系可选用工具或工件坐标系。

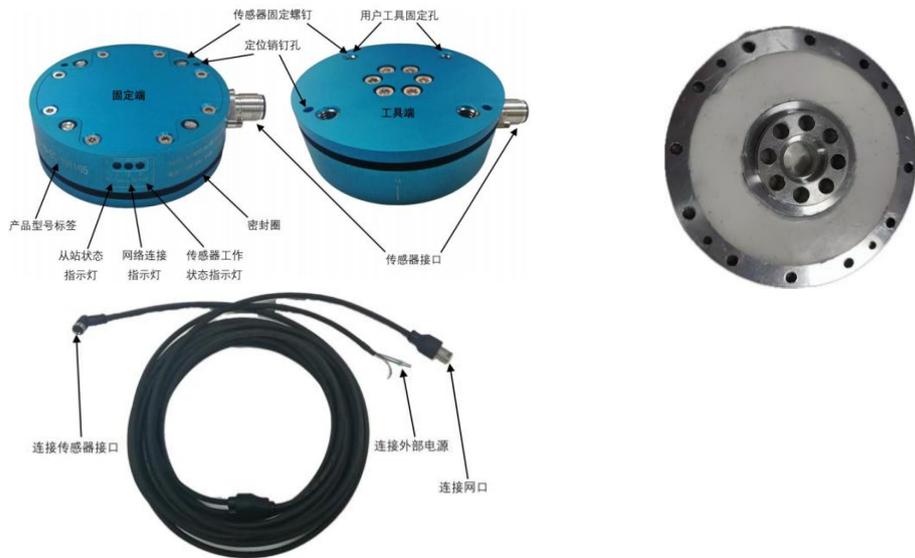
### 1.3 使用场景与限制

项目	说明
传感器类型	六维力传感器
运动指令	所有运动指令
支持机型	仅支持工业六轴机型
转弯区	支持转弯区
碰撞检测	不支持碰撞检测

## 2 导纳力控平台

### 2.1 硬件组成

如下图所示为安装传感器所需的硬件设备，分别为传感器本体、法兰转接板、传感器连接线缆，传感器通过 EtherCat 网口接入电柜。



安装完成图如下图所示。



## 2.2 导纳软件界面

## 2.2.1 开启导纳

在连接好传感器以及更新 ENI 文件后，可在工艺包界面打开导纳力控，导纳界面位于 HMI--工艺包--六维力传感器力控下，点击启用按钮即可开始使用导纳力控（确保可以扫描到传感器从站才能开启，否则报错），点击保存可以保存设置的参数。每次开启与关闭都会重启控制器。



在开启导纳后，可在监控界面那个看到传感器的反馈值，如下图所示，可人为对传感器施加力来验证。



## 2.2.2 传感器型号

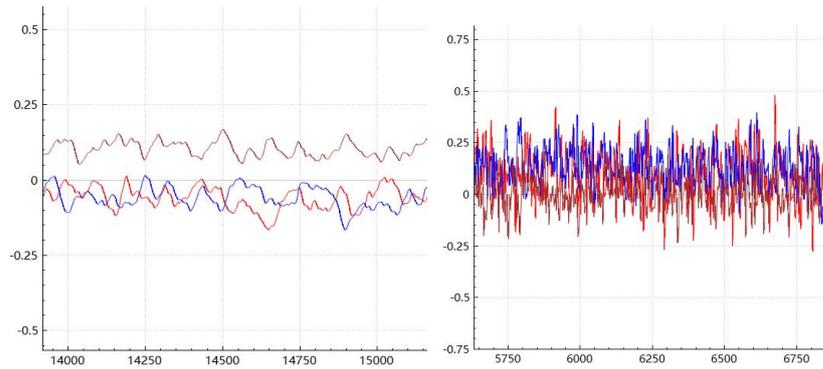
传感器型号：在传感器型号一栏中，可以选择传感器型号，目前支持的传感器型号有以下几种：

厂家	型号
蓝点	ST-6-200-10-RS
鑫精诚	XJC-6F-D80-H25
宇立	SRI-M8126f32

## 2.2.3 截止频率

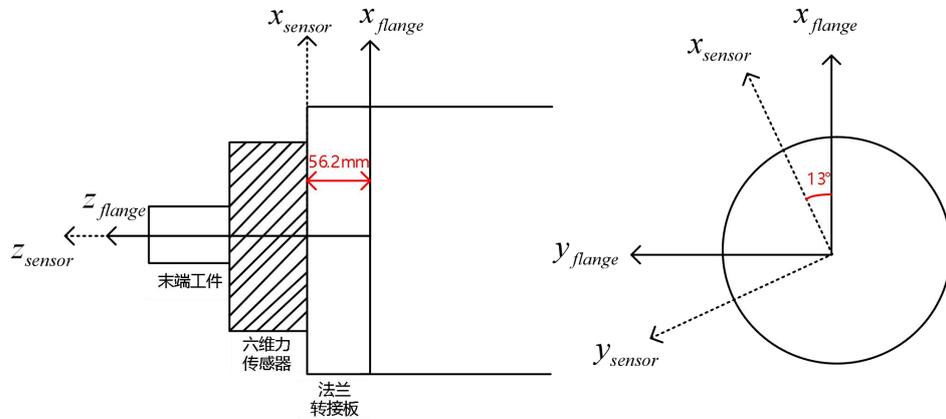
在导纳力控中，能否精准的控制机器人进行力/力矩控制很大程度上依赖于六维力传感器的数据是否准确，往往实际运行过程中会产生大量噪声干扰，影响传感器的读数，因此需要对传感器的数据进行滤波处理，而截止频率是滤波器中允许通过的最大信号频率，增大截止频率会引入更大的噪声，但可以提升响应速度，减小截止频率会使力信号变得平滑，但会降低响应速度。

通常默认传感器的截止频率为 100Hz。如图所示分别为截止频率为 30Hz 和 100Hz 时采集的传感器数据：



2.2.4 传感器安装位置

传感器安装位置是指传感器坐标系相对于法兰坐标系的位姿。如图所示



其中，Z 的值填 56.2，C 的值填 13。其余均填 0。

**传感器位置**

获取传感器坐标系相对于法兰坐标系的位姿关系。XYZ为传感器坐标系原点在法兰坐标系中的位置，ABC为传感器坐标系相对于法兰坐标系的旋转角度。详细情况请参考对应的手册内容。

X (mm) : 0.00	A (°) : 0.00
Y (mm) : 0.00	B (°) : 0.00
Z (mm) : 56.20	C (°) : 13.00

## 3 导纳指令

## 3.1 FcInitE

说明	用于力控开启前的一些初始化工作，如设置工件、工具和力控坐标系。
定义	<p>FcInitE (Tool, Wobj, ForceFrameRef)</p> <p>Tool 数据类型: pose 力控所使用的工具，力控坐标系的原点是该工具的 TCP(姿态与第三个参数中所选择的坐标系姿态相同)。需要注意的是，所有使用的转接法兰都需要包含在工具的定义中。该参数默认为 Tool0。</p> <p>Wobj 数据类型: pose 力控所使用的工件，很多力控功能的定义是相对于工件坐标系来的，例如力控坐标系的姿态、搜索模式和终止条件等。该参数默认为 Wobj0。</p> <p>ForceFrameRef 数据类型: int 定义力控坐标系相对于哪个坐标系定义，支持： 1.工件坐标系 2.工具坐标系 默认为工具坐标系 2</p>
示例	<p>例 1</p> <p>FcInitE(tool1,wobj0,2);</p> <p>力控初始化，并且定义力控所使用的工具 Tool1 和工件 Wobj0，以及力控坐标系相对于工具坐标系 Tool1 定义。</p>
注意	<ul style="list-style-type: none"> <li>● FcInitE 之后，FcStartE 之前，不能插入运动指令，否则运动指令不生效。</li> <li>● 不支持外部工具和手持工件。</li> </ul>

## 3.2 FcStartE

说明	开启力控
定义	FcStartE ()  没有参数直接使用
示例	例 1 FclnitE (Tool1, Wobj0, 1); FcStartE (); FclnitE 之后通过 FcStartE 开启力控，此时机器人处于力控模式。
注意	<ul style="list-style-type: none"> <li>● FclnitE 之后，FcStartE 之前，不能插入运动指令，否则运动指令不生效。</li> <li>● FclnitE 之后调用</li> </ul>

## 3.3 FcStopE

说明	关闭力控
定义	FcStopE()  没有参数，直接使用
示例	例 1 FclnitE (Tool1, Wobj0, 1); FcStartE(); Wait 5 FcStopE(); 停止力控，将机器人从力控制切换到位置控制。执行该指令会清空所有的力控状态
注意	<ul style="list-style-type: none"> <li>● FcStartE 之后调用此接口，调用此接口时会清空力控状态，如需再次开启力控，则需重新 FclnitE。</li> </ul>

## 3.4 FcSetParaE

说明	设置力控参数以及选择力控自由度
定义	<p>FcSetParaE(Kp, Ki, Kd, select_matrix)</p> <p>Kp 数据类型: Fccartnum xyzabc 方向上的比例系数</p> <p>Ki 数据类型: Fccartnum xyzabc 方向上的积分系数</p> <p>Kd 数据类型: Fccartnum xyzabc 方向上的微分系数</p> <p>select_matrix 数据类型: double[6] 选择笛卡尔空间的力控方向, 1 开启, 0 关闭</p>
示例	<p>例 1</p> <pre>VAR fccartnum Kp_z = fccart:{0.0,0.0,0.002,0.0,0.0,0.0}; VAR fccartnum Ki_z = fccart:{0.0,0.0,0.0,0.0,0.0,0.0}; VAR fccartnum Kd_z = fccart:{0.0,0.0,0.0,0.0,0.0,0.0}; VAR double select_z[6] = {0,0,1,0,0,0}; FclnitE(tool0,wobj1,1); FcSetParaE (Kp_z,Ki_z,Kd_z,select_z);</pre> <p>根据力控自由度选择向量 select_z 的值确定 z 方向开启力控, 并将 z 方向的力控参数参数值分别设为 0.002, 0.0, 0.0 (其中 Kp 影响了对期望力的响应快慢, 若设定期望力为 10N, 当 Kp 越大时, 达到期望力 10N 的时间越少, 反之, 当 Kp 越小时, 达到期望力 10N 的时间就越久, 可以理解为 KP 越大, 在力控方向可以更容易拖动; Ki 系数表示位置补偿量的积分; Kd 则影响对力响应的变化率, 主要减少机器人在与环境接触的瞬间产生的过大接触力。在实际调试过程中, 基本只需关注 Kp 的大小即可。)</p>
注意	<ul style="list-style-type: none"> <li>● FclnitE 之后, FcStartE 之前调用。</li> </ul>

## 3.5 FcRefForceE

说明	设置期望力，机器人会在力控方向上施加该设置的期望力
定义	<p>FcRefForceE (Fx, Fy, Fz, Tx, Ty, Tz)</p> <p>Fx 数据类型: double x 方向上的期望力, 期望力范围为: <math>\pm 200\text{N}</math></p> <p>Fy 数据类型: double y 方向上的期望力, 期望力范围为: <math>\pm 200\text{N}</math></p> <p>Fz 数据类型: double z 方向上的期望力, 期望力范围为: <math>\pm 200\text{N}</math></p> <p>Tx 数据类型: double A 方向上的期望力矩, 期望力矩范围为: <math>\pm 10\text{Nm}</math></p> <p>Ty 数据类型: double B 方向上的期望力矩, 期望力矩范围为: <math>\pm 10\text{Nm}</math></p> <p>Tz 数据类型: double C 方向上的期望力矩, 期望力矩范围为: <math>\pm 10\text{Nm}</math></p>
示例	<p>例 1</p> <pre>FcInitE(tool0,wobj1,1); FcRefForceE(0,0,30,0,0,0); FcStartE();</pre> <p>设置 z 方向的期望力为 30N</p>
注意	<ul style="list-style-type: none"> <li>● FcInitE 之后, FcStartE 之前调用。</li> </ul>

## 3.6 FcResetBiasE

说明	传感器清零
定义	FcResetBiasE() 传感器清零，无参数直接使用
示例	例 1 FcInitE(tool0,wobj1,1); FcRefForceE(0,0,30,0,0,0); wait (0.5); FcResetBiasE(); FcStartE();
注意	<ul style="list-style-type: none"><li>● 在接触前最好进行一次传感器清零。当前清零方式时暂时清掉传感器当前数据，变化姿态可能会对传感器数据有影响。</li></ul>

3.7 FcCondForceE

说明	设置接触力监控条件，达到该力是判断为接触
定义	<p>FcCondForceE (xmin, xmax, ymin, ymax, zmin, zmax, lsInside, TimeOut)</p> <p>xmin 数据类型: double x 方向上的力限制下限, 单位为 N</p> <p>xmax 数据类型: double x 方向上的力限制上限, 单位为 N</p> <p>ymin 数据类型: double y 方向上的力限制下限, 单位为 N</p> <p>ymax 数据类型: double y 方向上的力限制上限, 单位为 N</p> <p>zmin 数据类型: double z 方向上的力限制下限, 单位为 N</p> <p>zmax 数据类型: double z 方向上的力限制上限, 单位为 N</p> <p>lsInside 数据类型:bool 用于定义限制条件内部为 true 还是外部为 true</p> <p>TimeOut 数据类型:doulble 定义超时时间, 单位为秒</p>
示例	<p>例 1</p> <pre>FcInItE(tool0,wobj1,1); FcStartE(); FcCondForceE(-100,100,-100,100,-100,100,true,20); FcCondWaitWhileE();</pre> <p>定义一个终止条件，当接触力在力控坐标系 X/Y/Z 轴方向的大小在± 100N 的范围之内时，程序阻塞在 FcCondWaitWhileE，当超出 100N 范围时或超过 20s 时，跳出阻塞继续执行后面程序。</p>
注意	<ul style="list-style-type: none"> <li>● FcStartE 之后 FcStopE 之前可以调用</li> </ul>

## 3.8 FcCondWaitWhileE

说明	用于激活在之前定义的终止条件，并在当前行等待直到这些条件变为 False 或者超时
定义	FcCondWaitWhileE()  无参数直接使用
示例	例 1 FcInitE(tool0,wobj1,1); FcStartE(); FcCondForceE(-100,100,-100,100,-100,100,true,20); FcCondWaitWhileE();
注意	● 在 FcCondForceE 指令之后调用

## 3.9 FcSupvForceE

说明	设置接触力的安全范围监控
定义	<p>FcSupvForceE (xmin, xmax, ymin, ymax, zmin, zmax);</p> <p>xmin: 数据类型: double x 方向上的力限制下限, 单位 N</p> <p>xmax: 数据类型: double x 方向上的力限制上限, 单位 N</p> <p>ymin: 数据类型: double y 方向上的力限制下限, 单位 N</p> <p>ymax: 数据类型: double y 方向上的力限制上限, 单位 N</p> <p>zmin: 数据类型: double z 方向上的力限制下限, 单位 N</p> <p>zmax: 数据类型: double z 方向上的力限制上限, 单位 N</p>
示例	<p>例 1</p> <pre> FcInitE(tool0,wobj1,1); FcSupvForceE(-100,100,-150,150,-200,200); </pre> <p>在开启导纳力控后, 监控力控坐标系受到的接触力 <math>F_x</math>, <math>F_y</math>, <math>F_z</math>, 且 <math>F_x</math>, <math>F_y</math>, <math>F_z</math> 分别在 <math>[-100,100]</math>, <math>[-150,150]</math>, <math>[-200,200]</math> 范围内运行, 超出范围会下电报警。</p>
注意	<ul style="list-style-type: none"> <li>● FcInitE 之后, FcStartE 之前调用</li> </ul>

## 3.10 FcSupvTCPSpeedE

说明	设置末端 tcp 速度安全监控
定义	<p>FcSupvTCPSpeedE (xmin, xmax, ymin, ymax, zmin, zmax);</p> <p>xmin: 数据类型: double x 方向上的速度限制下限, 单位 mm/s, 范围±1000mm/s</p> <p>xmax: 数据类型: double x 方向上的速度限制上限, 单位 mm/s, 范围±1000mm/s</p> <p>ymin: 数据类型: double y 方向上的速度限制下限, 单位 mm/s, 范围±1000mm/s</p> <p>ymax: 数据类型: double y 方向上的速度限制上限, 单位 mm/s, 范围±1000mm/s</p> <p>zmin: 数据类型: double z 方向上的速度限制下限, 单位 mm/s, 范围±1000mm/s</p> <p>zmax: 数据类型: double z 方向上的速度限制上限, 单位 mm/s, 范围±1000mm/s</p>
示例	<p>例 1</p> <p>FcInitE (tool0,wobj1,1);</p> <p>FcSupvTCPSpeedE(-150,150,-500,500,-500,500);</p> <p>在开启导纳力控后, 监控 TCP 线速度, X 正负方向速度不允许超过 150mm/s, Y 正负方向和 Z 正负方向速度不允许超过 500mm/s。超出范围会下电报警。</p>
注意	<ul style="list-style-type: none"> <li>● FcInitE 之后, FcStartE 之前调用</li> </ul>

## 3.11 FcSupvPosBoxE

说明	用于定义一个长方体区域，当机器人的 TCP 离开这片空间时，将会触发程序停止。
定义	<p>FcSupvPosBoxE(SupvFrame, Box)</p> <p>SupvFrame: 数据类型: Pose 用于选择监控的空间体相对哪个坐标系定义。该坐标系是从工件坐标系上叠加一个坐标系转换得来的，转换坐标系由 pose 定义，默认使用 pose0，即不使用任何转换，直接使用工件坐标系。</p> <p>Box: 数据类型: fcboxvol 定义一个长方体区域。</p>
示例	<p>例 1</p> <pre>VAR fcboxvol box1 = fcbv:{-100.0, 100.0, -200.0, 200.0, -300.0, 300.0}; VAR pose pose1 = pe:{0, 0, 0},{1, 0, 0, 0}; FCSupvPosBox (pose1, box1);</pre> <p>监控一个 X 方向-100~100, Y 方向-200~200, Z 方向-300~300 的长方体，该长方体相对于工件坐标系定义。TCP 超过该范围下电报警。</p>
注意	<ul style="list-style-type: none"> <li>● FcInitE 之后, FcStartE 之前调用</li> </ul>

## 4 示例程序

```

GLOBAL PROC main()
    var fccartnum Kp_z=fccart:{0.0,0.0,0.003,0.0,0.0,0.0};
    var fccartnum Ki_z=fccart:{0.0,0.0,0.0,0.0,0.0,0.0};
    var fccartnum Kd_z=fccart:{0.0,0.0,0.0,0.0,0.0,0.0};
//定义力控参数变量
    var double select_z[6]={0,0,1,0,0,0};
//选择力控自由度，开启力控坐标系 z 方向力控
    MoveAbsJ(J0,v1000,z50,tool0) qishidian,v1000,z50,tool0
    MoveAbsJ(J1,v500,z50,tool0);
//移动至打磨点，通常是与环境保持些许距离的点，如点 J1
    FcInitE(tool0,wobj1,1) ;
//力控初始化，选择工件坐标系 WOBJ1 为力控坐标系
    FcSetParaE(Kp_z,Ki_z,Kd_z,select_z) ;
//设置力控参数
    FcSupvForceE(-100,100,-100,100,-100,100);
//设置接触力安全范围监控，保证在此范围内进行作业
    FcSupvTCPspeedE(-1000,1000,-1000,1000,-1000,1000) ;
//设置 TCP 速度安全范围监控，保证在此范围内进行作业
    FcRefForceE(0,0,20,0,0,0);
//设置期望力
    wait (0.5);
    FcResetBiasE();
//传感器清零
    wait (0.5);
    FcStartE();
//开启力控
    FcCondForceE(-100,100,-100,100,-8,8,true,60);
    FcCondWaitWhileE();
设置接触力判断，在 z 方向达到 8N 的力时跳出 waitwhile，执行后面的程序，等待时间 60s
    MoveAbsJ( J2,v500,z50,tool0);
    MoveAbsJ( J3v500,z50,tool0);
    MoveAbsJ( J4,v500,z50,tool0);
    MoveAbsJ( J5,v500,z50,tool0);
    MoveAbsJ( J6,v500,z50,tool0);
    MoveAbsJ( J7,v500,z50,tool0);
//力控运行轨迹
    FcStopE();
//关闭力控
    MoveL(P0,v500,z50,tool0) ;
//回到安全位置
ENDPROC

```