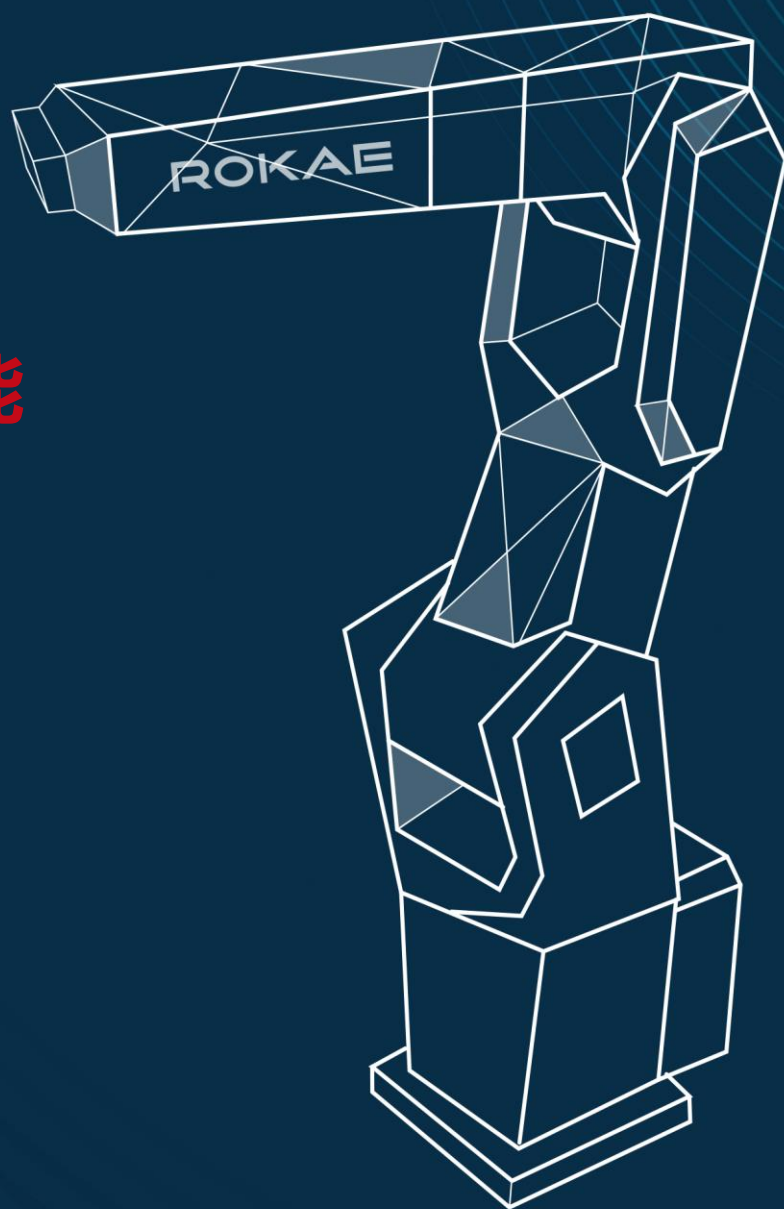


ROKAE 珞石

电镀线跟踪功能
使用手册



让智造更高效

电镀线跟踪功能

使用手册

控制系统版本: xCore V1.5

文档编号: 20220000006

文档版本: 1.2

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 2015-2022 ROKAE 保留所有权利

珞石（北京）科技有限公司

中国.北京

目录

1 概述	1
1.1 使用场景	1
1.2 使用限制	2
2 系统结构	3
3 硬件安装	5
3.1 硬件规格	5
3.2 触发信号接线	5
3.3 编码器接线	6
3.4 指示灯	7
3.5 示例	7
4 功能配置	9
4.1 编码器	9
4.1.1 新建编码器	9
4.1.2 配置编码器	9
4.1.3 验证	10
4.2 触发信号	10
4.2.1 新建	10
4.2.2 配置	10
4.2.3 验证	10
4.3 传送带	11
4.3.1 创建传送带	11
4.3.2 标定传送带	12
4.3.3 配置传送带	13
4.3.4 验证标定结果	13
4.3.5 提高标定精度	13
5 编程与调试	15
5.1 指令	15
5.1.1 ActUnit	15
5.1.2 DeactUnit	15
5.1.3 WaitWobj	16
5.1.4 DropWobj	16
5.1.5 GetCnvSpeed	17
5.1.6 GetCnvPulse	18
5.1.7 GetConnctObjPos	18

5.1.8 Wait（跟踪指令中间使用时）	19
5.1.9 MoveL（跟踪指令中间使用时）	19
5.1.10 MoveC（跟踪指令中间使用时）	20
5.2 工件坐标系定义	20
5.3 示教	21
5.3.1 示教点位	21
5.3.2 验证结果	22
5.4 运行	22
5.5 调试	22
5.5.1 跟踪过程中运动滞后	23
5.5.2 沿传送带运动方向上的误差	23
5.5.3 沿非传送带运动方向上的位置偏差	23
5.6 错误处理	23
5.6.1 错误“工件超出启动窗口”	24
5.6.2 错误“工件超出跟踪区”	24
5.6.3 错误“WaitWobj 时，超过某时间依然没有关联工件”	24
5.6.4 错误“未 DropWobj 时，再次 WaitWobj”	24
5.7 样本程序	24
修订记录	27

1 概述

1.1 使用场景

电镀线跟踪功能是为 PCB 生产的电镀制程开发的。在该制程中，输送带悬挂 PCB 板经过电镀环境，一系列化学处理工序将在 PCB 板表面镀上一定厚度的铜。为了保证镀铜的厚度，必须保证固定的化学反应时间。因此输送带必须保持匀速运动，不能停止，即使在上下料时也要匀速运动。电镀线跟踪功能可以实现机器人与电镀线输送带同步时上下 PCB 板。

1.2 使用限制

由于电镀线跟踪功能是为了电镀线场景定制开发的，因此有一些使用上的限制。包括：

项目	说明
速度	最高可同步的输送带速度不超过 6m/min
重复定位精度	±1mm
指令	在该功能中，不支持在某点附近定时触发 IO 的 Trigger 类指令
传感器	仅支持编码器同步，不支持视觉同步
输送带	仅支持直线型编码器
重复路径精度	跟踪路径误差高于 2mm
转弯区	支持转弯区

2 系统结构

CTM 模块（跟踪的硬件模块）采集触发传感器（光电开关等）的信号，并实时获取编码器的数据，传输到 xCore 控制系统，因此该功能的系统结构如图：

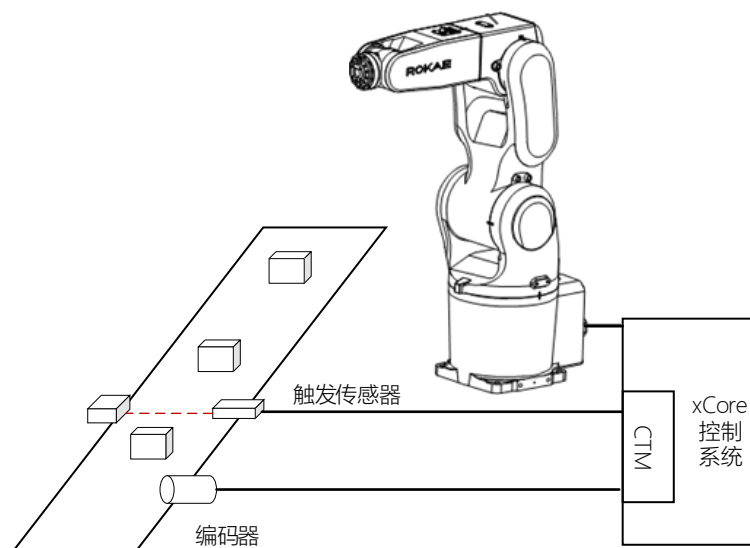


图 1 电镀线跟踪的系统结构

3 硬件安装

3.1 硬件规格

用于支持机器人末端实现对电镀线进行跟踪功能的模块称为 CTM 模块,它采用 24V 直流供电,能够读取 2 路光电开关信号及 2 路增量式编码器信号,并通过 EtherCAT 总线,将读取到的信号以一定的时间周期,发送给控制器,从而为控制器执行电镀线上的工件跟踪算法提供必要的信号采集和输入支持。

CTM 模块安装在控制柜内部,并在出厂前完成配置,其对外的接线口位于控制柜的面板窗,端子分布如下:

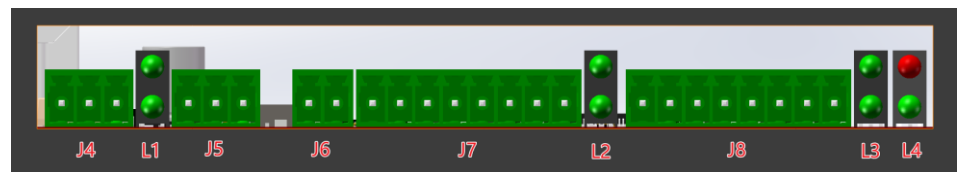


图 2 CTM 模块的端子分布

3.2 触发信号接线

CTM 模块可以支持具有以下参数的光电开关:

项目	参数
供电	DC10~30V
线数	三线制
输出类型	NPN 或 PNP

已完成对以下型号的光电开关的测试:

台邦: TB12J-D15N1、E3Z-R81。

具体的接线方式如下:

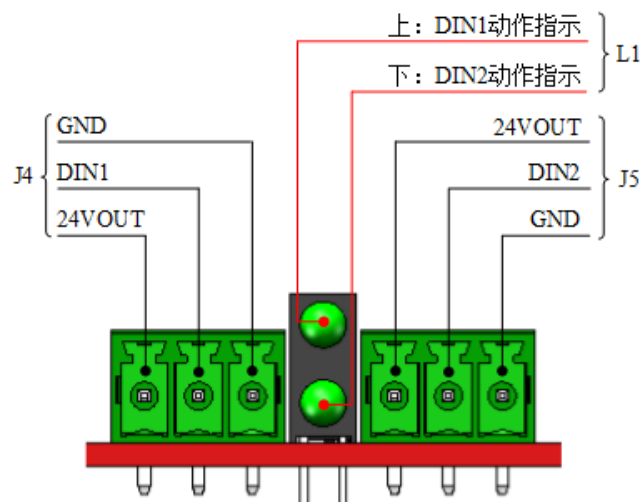


图 3 触发传感器 (光电开关) 的接线方式

位置	序号	标识	描述
J4	1	24VOUT	光电开关 24V 供电端

	2	DIN1	1 号光电开关信号输入端
	3	GND	电源负
J5	1	24VOUT	光电开关 24V 供电端
	2	DIN2	2 号光电开关信号输入端
	3	GND	电源负

与 J4、J5 配合的连接器的品牌为 DEGSON (高松), 型号为 15EDGKN-3.5-03P-14-00A(H)。

3.3 编码器接线

CTM 模块可以支持具有以下参数的编码器：

供电电压 DC5V±5%，负载电流≤200mA；

增量式；

具有 A/B 两相输出信号，其相位差为 90°，信号输出类型为线性驱动器输出 (AM26LS31 相当)；

注：对于带 Z 相信号的编码器，Z 相信号不接。

已完成对以下编码器型号的测试：

欧姆龙：E6B2-CWZ1X、E6C2-CWZ1X

多摩川：TS5312N512-2000C/T (多摩川)。

具体的接线方式如下：

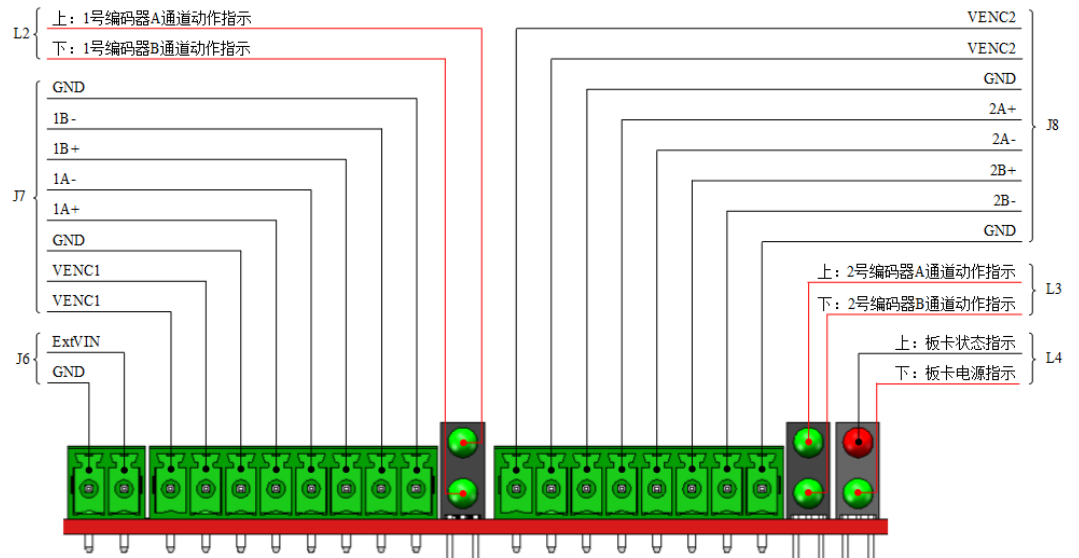


图 4 编码器接线方式

J6 ⁽¹⁾	1	GND	电源负
	2	ExtVIN	外部输入电源正
J7	1	VENC1 ⁽²⁾	1 号编码器供电输出
	2	VENC1 ⁽²⁾	1 号编码器供电输出
	3	GND	电源负
	4	1A+	1 号编码器 A 相正相端
	5	1A-	1 号编码器 A 相反相端
	6	1B+	1 号编码器 B 相正相端

	7	1B-	1 号编码器 B 相反相端
	8	GND	电源负
J8	1	VENC2 ⁽²⁾	2 号编码器供电输出
	2	VENC2 ⁽²⁾	2 号编码器供电输出
	3	GND	电源负
	4	2A+	2 号编码器 A 相正相端
	5	2A-	2 号编码器 A 相反相端
	6	2B+	2 号编码器 B 相正相端
	7	2B-	2 号编码器 B 相反相端
	8	GND	电源负

注：

(1) 若用户所使用的编码器的供电电压与板载供电 5V 不同，则用户可通过此端口自行接入所使用的编码器的供电电压；否则，将该端口悬空即可。

(2) 相同的引脚定义，用户接线时可任选其一进行连接。

与 J6 配合的连接器的品牌为 DEGSON（高松），型号为 15EDGKN-3.5-02P-14-00A(H)；与 J7、J8 配合的连接器的品牌为 DEGSON（高松），型号为 15EDGKN-3.5-08P-14-00A(H)。

3.4 指示灯

CTM 模块上共有 4 组 LED 指示灯，每组指示灯上下各一个，其状态描述如下：

L1	上：DIN1 动作指示	连接到 J4 的 1 号光电开关动作时点亮，不动作时熄灭
	下：DIN2 动作指示	连接到 J5 的 2 号光电开关动作时点亮，不动作时熄灭
L2	上：1 号编码器 A 通道动作指示	连接到 J7 的 1 号编码器 A 通道动作时点亮，不动作时熄灭
	下：1 号编码器 B 通道动作指示	连接到 J7 的 1 号编码器 B 通道动作时点亮，不动作时熄灭
L3	上：2 号编码器 A 通道动作指示	连接到 J8 的 1 号编码器 A 通道动作时点亮，不动作时熄灭
	下：2 号编码器 B 通道动作指示	连接到 J8 的 1 号编码器 B 通道动作时点亮，不动作时熄灭
L4	上：板卡状态指示	板卡正常工作时，该指示灯熄灭；板卡有故障时，该指示灯常亮。
	下：板卡电源指示	板卡上电时点亮，板卡断电时熄灭。

3.5 示例

下图示意了 1 路光电开关和 1 路编码器的接线方式：

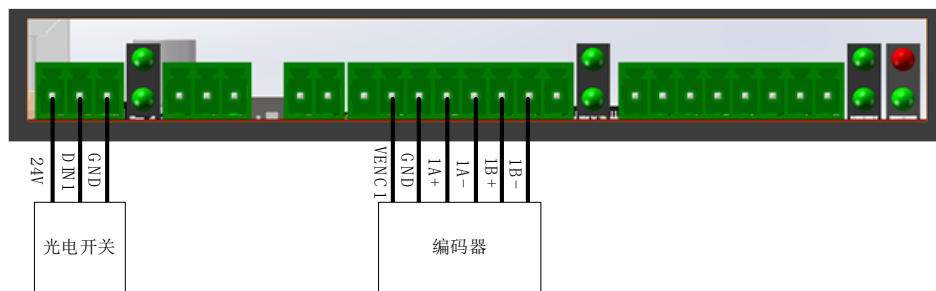


图 5 接线示例

4 功能配置

本章介绍如何在控制系统软件上配置相关参数。

4.1 编码器

4.1.1 新建编码器

新建编码器界面功能介绍。

您在创建任何传送带之前，必须首先在编码器界面新建一个编码器信息。每个实体传送带必须配备一个编码器。

在编码器界面点击“+”新建编码器，HMI 界面显示如下图。

选择对应实体编码器通道，点击“下一步”完成创建编码器。

← 编辑编码器：encoder0

基本信息

名称 encoder0

描述

基本设置

通道 0

类型 相对式

方向 正向

编码器分辨率 1000

取消
上一步
下一步

图 6 新建编码器

4.1.2 配置编码器

用户可以修改或者编辑编码器的参数，编码器的参数含义见下表。

您可以删除列表中的编码器。选中它，然后点击“删除”按钮。

注意：当有对应传送带绑定要删除的编码器时，无法删除此编码器。

编码器的相关参数如下：

序号	项目	含义
1	名称	输入您要命名的编码器名称，默认为 encoder0；创建传送带会根据编码器名称进行绑定；
2	描述	对此编码器进行描述；
3	通道	对应的实体编码器通道，与实际编码器接到 ATM 模块上的通道相关，不可为空，电镀线功能最多支持两路编码器接入；
4	类型	定义编码器类型，相对式编码器、绝对式编码器；
5	方向	正向定义编码器为增量的方式计算编码器脉冲值，反向定义编码器为减量的方式计算编码器脉冲值；

6	编码器分辨率	默认 1000，保持默认即可；
---	--------	-----------------

4.1.3 验证

如何验证编码器是否配置成功？

- 1) 根据已经创建的编码器创建一条传送带（参考下一节创建一条传送带）
- 2) 移动实体传送带
- 3) 打开状态监控，进入传送带界面

当移动传送带时，可以在状态监控中实时监控编码器脉冲值



图 7 编码器状态监控

另外，可使用一个简单的程序，如下所示。运行程序并移动传送带。当传送带开始移动时，可以打印编码器脉冲值

```

1 GLOBAL PROC main()
2 ActUnit "conveyor0"
3 double cnv = GetCnvPulse conveyor0
4 print(cnv)
5 ENDPROC

```

图 8 编码器脉冲输出程序

4.2 触发信号

本节介绍如何配置触发信号。

触发信号支持光电开关触发，电镀线功能最多支持两路光电开关接入。

4.2.1 新建

4.2.2 配置

4.2.3 验证

电镀线硬件板正常通信时，HMI 状态监控中观察触发信号状态：

当有物体触发光电开关信号后，在状态监控中可以看到光电状态为 true，反之当未触发信号时，

光电状态为 false;



图 9 查看光电开关状态

4.3 传送带

4.3.1 创建传送带

您在创建任何传送带之前，必须首先创建一条编码器信息，可以创建多条传送带信息对应一个编码器。但每个实体传送带必须配备一个编码器。

在传送带界面点击“+”新建传送带，HMI 界面显示如下图：

← 新建传送带

基本信息

名称

描述

编码器

触发行为

信号类型

位姿标定

标定坐标系 手动输入

图 10 传送带创建界面

各项参数如下：

类别	选项	说明
触发行为方式	光电	通过光电触发信号的方式来创建传送带信息，目前仅支持光电信号错发。
	视觉	预留

	软件	预留
信号类型	上升沿	当信号有上升沿时的开关动作，当电位由低变高而触发输出变化的叫上升沿触发
	下降沿	当信号有下降沿时的开关动作，当电位由高变低而触发输出变化的叫下降沿触发
位姿标定	标定坐标系	通过标定的方式确定传送带，可以准确的得到传动比、传送带基座标系、跟踪区域等参数信息
	手动输入	可以手动的方式输入传送带的传动比、传送带基座标系、跟踪区域等参数信息

4.3.2 标定传送带

标定传送带时，需要使用标定好的工具，工具的标定精度将决定传送带的标定精度，因此尽量使用刚性尖端，安装到机器人法兰上，进行高精度的工具标定。

若担心由于工具标定引入误差，可以直接使用未经标定的刚性尖端，但在标定传送带的各点时，工具只能平移，不能旋转，否则将导致标定错误！

具体标定流程包括：

- 1) 选择对应实体编码器，触发方式选择光电，信号类型选择上升沿或下降沿；
- 2) 选择“标定坐标系”，点击“下一步”进入标定界面；

← 标定传送带坐标系



3) 实体传送带上放置一物块，物块在传送带上可以稳固放置，避免物块与传送带的相对移动，移动传送带使其经过光电开关，停止传送带；

4) 移动机器人工具，使其 TCP 对准工件上一特征点 A，点击“确认第一点”（此点做为工作区域起点）；

5) 继续移动传送带，使工件到达合适位置，停止传送带；

6) 沿传送带方向移动机器人使其对准工件上一特征点 A，点击“确认第二点”；

7) 继续移动传送带，使工件停止在工作区域终点位置，停止传送带；

8) 沿传送带方向移动机器人使其对准工件上一特征点 A，点击“确认第三点”（此点作为工作区域终点）；

9) 把工件放在传送带上，沿垂直于传送带的方向移动一段距离，移动机器人使其对准工件上一特征点 A，点击“确认第四点”；

10) 点击“下一步”完成标定传送带信息；

4.3.3 配置传送带

传送带标定完成后，用户可以对传送带的参数进行修改和配置，可配置的参数和含义如下表：

序号	配置参数	说明
1	名称	传送带的名称（不与 RL 规则冲突的合法字符）
2	描述	对传送带的描述
3	编码器	选择使用的的编码器
4	触发行为	工件到来的触发信号，当前仅支持光电触发
5	信号类型	可选择“上升沿”或“下降沿”作为工件到来的光电触发信号
6	光电偏移（单位：mm）	光电开关的位置，相对于传送带基坐标系 x 方向定义
7	传动比（单位：count/m）	传送带移动 1 米对应的编码器数据变化量
8	传送带基坐标系 X/Y/Z/A/B/C（单位：mm/deg）	传送带基坐标系的位姿，相对于世界坐标系定义。X/Y/Z 为平移部分；A/B/C 为旋转部分，用欧拉角表示
9	工作区开始（单位：mm）	传送带跟踪范围的起点，工件到达此位置后加入跟踪队列。相对于传送带基坐标系 x 方向定义。
10	启动窗口（单位：mm）	可以被跟踪的工件区域，工件到达此位置时若还未被跟踪，即从跟踪队列中删除。相对于工作区开始位置定义。
11	工作区结束（单位：mm）	传送带跟踪范围的终点，跟踪中到达此位置时停止跟踪。相对于传送带基坐标系 x 方向定义。
12	跟踪位置补偿（单位：mm）	用于对跟踪过程中机械臂沿传送带方向上的位置滞后进行补偿
13	跟踪时间补偿（单位：ms）	作用同上。若滞后距离与传送带速度相关，则选用时间补偿；否则选用位置补偿。

4.3.4 验证标定结果

如何验证标定是否成功？

- 1) 在传送带上放置一工件，移动传送带，工件触发光电开关后，暂停传送带，移动机器人使 tcp 对准工件上一特征点，状态监控中读取此时编码器值和机器人位置值；
- 2) 继续移动传送带一段距离，暂停传送带，移动机器人使 tcp 对准工件上一特征点，状态监控中读取此时编码器值和机器人位置值；
- 3) 手动计算出每米的脉冲差值与传动比值的精度差值比；
- 4) 与标定后计算出的传动比进行对比验证，结果是否准确；

4.3.5 提高标定精度

传送带的标定精度直接决定了最终的跟踪精度，在发现传送带的标定精度较差时，通过以下方法可以提高标定精度：

- 1) 尽量使用刚性尖端和明显特征点的物块标定。使用刚度大的尖端标定时，确定点位的精度更高。
- 2) 标定时，只进行工具平移，不进行旋转，避免由于工具标定和旋转带来的误差。
- 3) 标定前 3 点间的距离越远，标定精度越高，第 4 点在垂直传送带方向移动地越远，标定精度越高；

5 编程与调试

5.1 指令

本节介绍所有与电镀线跟踪的相关指令。

5.1.1 ActUnit

说明

机械单元接货指令；激活之后，程序中开始使用采集的编码器脉冲数据和光电开关信息，且与传送带跟踪的指令在激活之后才有效。

返回值

无。

定义

ActUnit(name)
name
数据类型：string
传送带的名称

示例

```
ActUnit(“conveyor1”) //激活传送带  
WaitWObj wobj_cnv //阻塞等待工件来临，开始跟踪  
wait 2  
DropWobj //断开对工件的连接，停止跟踪  
DeactUnit(“conveyor1”) //失效传动带跟踪单元
```

5.1.2 DeactUnit

说明

用于失效传送带跟踪单元；失效之后，信号采集将停止。

返回值

无。

定义

DeactUnit(name)

name

数据类型: string

传送带名称

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2
DropWObj                //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.3 WaitWobj

说明

等待工件“连接”，即工件被关电感知之后，进入跟踪区域即被填到跟踪队列里面。
该指令可以将跟踪队列第一个工件与之前定义好的随动工件坐标系建立连接。如果队列为空，那么 RL 程序将持续在该指令处等待，直到出现第一个可供连接的工件。

返回值

无。

定义

WaitWobj Wobj

Wobj

数据类型: wobj

与传送带单元绑定的工件

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2
DropWObj                //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.4 DropWobj

说明

该指令用来断开 RL 程序与当前工件之间的连接关系，用于结束跟踪。意味着此工件，跟踪完毕。

可以进行下一个跟踪循环(WaitWobj → 跟踪操作 → DropWobj 的循环)。

返回值

无。

定义

DropWobj Wobj
Wobj
数据类型：wobj
与传送带单元绑定的工件

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv //阻塞等待工件来临，开始跟踪
wait 2
DropWobj //断开对工件的连接，停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.5 GetCnvSpeed

说明

获取传送带当前速度

返回值

传送带当前速度，单位：mm/s
数据类型：double

定义

```
double cnv_speed = GetCnvSpeed (name)
name
数据类型：string
```

传送带名称

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2
double cnv_speed = GetCnvSpeed( "conveyor1" )
DropWobj              //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.6 GetCnvPulse

说明

获取传送带当前位置的编码器脉冲值

返回值

传送带当前编码器脉冲值
数据类型: int

定义

```
double cnv_pulse = GetCnvPulse (name)
```

name

数据类型: string
传送带名称

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2
double cnv_pulse = GetCnvPulse( "conveyor1" )
DropWobj              //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.7 GetConnctObjPos

说明

获取关联工件在传送带基坐标系的 X 坐标

返回值

数据类型: double

定义

```
double obj_pos = GetConnectObjPos (name)
```

name

数据类型: string

传送带名称

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2
double obj_pos = GetConnectObjPos ( "conveyor1" )
DropWobj               //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.8 Wait (跟踪指令中间使用时)

说明

程序等待一段时间, 范围 0~2147484 秒。

若在传送带跟踪过程中使用 Wait, 机器人跟随工件沿传送带运动方向同步运动。

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv      //阻塞等待工件来临, 开始跟踪
wait 2                 //机器人跟随工件运动
DropWobj               //断开对工件的连接, 停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.9 MoveL (跟踪指令中间使用时)

说明

在机器人进行传送带跟踪过程中，通过 MoveL 指令，机器人的 TCP 直线运动至传送带上跟踪工件的特定位置。

定义

MoveL p1,v1000,fine,tool1,wobj_cnv
wobj_cnv

数据类型：wobj
与传送带绑定的工件

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv //阻塞等待工件来临，开始跟踪
MoveL p1,v1000,fine,tool1,wobj_cnv
DropWobj //断开对工件的连接，停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.1.10 MoveC（跟踪指令中间使用时）

说明

在机器人进行传送带跟踪过程中，通过 MoveC 指令，将工具中心点 TCP 沿圆弧经过中心辅助点运动至传送带跟踪工件的特定位置。

定义

MoveC p1,v1000,fine,tool1,wobj_cnv
wobj_cnv

数据类型：wobj
与传送带绑定的工件

示例

```
ActUnit( "conveyor1" ) //激活传送带
WaitWObj wobj_cnv //阻塞等待工件来临，开始跟踪
MoveC p1,v1000,fine,tool1,wobj_cnv
DropWobj //断开对工件的连接，停止跟踪
DeactUnit( "conveyor1" ) //失效传动带跟踪单元
```

5.2 工件坐标系定义

本节描述了如何定义与传送带关联的工件，关于基本的工件创建方法，详见《xCore 机器人控制系统使用手册》7.1.7 工件列表。

编辑工件: wobj_cnv1

图 11 创建工作件

在定义传送带上的工件时，注意如下参数：

参数	描述
可移动对象	工件是否为可移动的。这里选择可移动，即与传送带协调
相关用户坐标系	用于协调机器人移动的机械单元作为工件用户坐标系。这里选择对应的传送带名称，例如 conveyor1
位姿标定	一般选择不标定即可

5.3 示教

本节描述了如何对该工件上的作业点位进行示教。

5.3.1 示教点位

1) 用以下列指令创建一段程序：

```
ActUnit "conveyor1" // 激活传送带
ConfL Off // 取消构型监控
MoveJ pt_prepare, v1000, fine, tool1 // 移动到准备位置
WaitWObj wobj_cnv1 // 阻塞等待工件到来
```

图 12 示教程序

2) 单步执行到 WaitWObj 指令

该指令会一直等待，直到传送带上的工件进入启动窗口。

3) 运行传送带，直至工件通过触发传感器

当工件进入启动窗口时，状态监控界面会显示该工件的位置，在工作空间的合适位置，停止传送带。

4) 新建点位或更新已有点位

关于基本的点位操作方法，参见《xCore 机器人控制系统使用手册》7.1.4 点位列表。

这里注意，在创建点位或更新点位位置的时候，软件界面顶部状态栏选择具体使用的工具，并选择 [5.2](#) 中与传送带关联的工件。

5.3.2 验证结果

1) 用下列指令创建一段程序

```
ActUnit "conveyor1" //激活传送带
ConfL Off //取消构型监控
MoveJ pt_prepare, v1000, fine, tool1 //移动到准备位置
WaitWObj wobj_cnv1 //阻塞等待工件到来
MoveL pt_cnv1, v1000, fine, tool1, wobj_cnv1 //移动到工件上的pt_cnv1点位
```

图 13 验证程序

2) 单步执行到 WaitWObj 指令

3) 运行传送带，直至创建了工作对象

当工件进入启动窗口时，状态监控界面会显示该工件的位置，在工作空间的合适位置，停止传送带。

4) 运行 MoveL 指令

若机械臂会运动到 [5.3.1](#) 节中工件上示教的相对位置，则证明示教成功。

5.4 运行

本节描述了如何运行一段包含有传送带跟踪的程序。

1) 用下列指令创建一段程序

```
ActUnit "conveyor1" //激活传送带
ConfL Off //取消构型监控
MoveJ pt_prepare, v1000, fine, tool1 //移动到准备位置
WaitWObj wobj_cnv1 //阻塞等待工件到来
MoveL pt_cnv1, v1000, fine, tool1, wobj_cnv1 //移动到工件上的pt_cnv1点位
Wait 0.5 //跟踪中，保持与工件的相对静止
DropWObj wobj_cnv1 //机械臂停止跟踪
MoveJ pt_prepare, v1000, fine, tool1 //移动到准备位置
DeactUnit "conveyor1"
```

图 14 运行程序

2) 上电并自动运行程序，执行到 WaitWObj 指令

如果传送带上的启动窗口中已经有一个工件，则指令将立即运行完成；否则会一直等待，直到传送带上的工件进入启动窗口。

注意：程序运行速度太低，容易导致机械臂跟踪作业过程中超出工作区域。

3) 运行传送带，直至创建了工作对象

当工件进入启动窗口时，状态监控界面会显示该工件的位置，同时 WaitWObj 会运行完成。

4) 机械臂执行在传送带的协调运动，直到 DropWObj 指令。

在上述示例程序中，机械臂会运动到传送带上关联工件的 pt_cnv1 点位，并保持与工件相对静止 0.5 秒。然后运行 DropWObj 指令，机械臂停止运动。

5.5 调试

本节描述了传送带跟踪应用中如何调试以提升跟踪精度。

5.5.1 跟踪过程中运动滞后

现象：自动运行时，跟踪过程中工件上的点位 pt_cnv1 沿传送带运动方向上有一定滞后；而在手动单步运行时，单步运行 MoveL 指令，pt_cnv1 点位无滞后现象。

解决方案：可适当修改传送带参数中的“跟踪位置补偿”和“跟踪时间补偿”参数。观察该滞后距离是否与传送带速度相关：若与传送带速度无关，估测其滞后距离，并修改“跟踪位置补偿”参数；若与传送带速度无关，估测其滞后时间（可估测距离，并除以传送带速度），并修改“跟踪时间补偿”参数。

5.5.2 沿传送带运动方向上的误差

现象：自动或手动运行时，工件上的点位 pt_cnv1 沿传送带运动方向上有一定误差，且随着离工作区起点越远，误差越大。

解决方案：重新标定传送带传动比或对传送带传动比数值进行微调。此现象的原因在于标定的传动比可能存在误差，需要重新标定传送带传动比，提高标定精度的方法包括：

采用加工精度较高、刚度较好的标定尖端；

jog 工具标定不同点位时，只平移不旋转，避免旋转带来的末端误差；

标定过程中尽量避免上下电前后更新点位；

第 3 点尽量离第 1 点远一些，等传送带完全停止时再进行点位示教

5.5.3 沿非传送带运动方向上的位置偏差

现象：自动或手动运行时，工件上的点位 pt_cnv1 沿非传送带运动方向上有一定误差，且随着离工作区起点越远，误差越大。

解决方案：重新标定传送带基坐标系或对传送带基坐标系的方向进行微调。此现象的原因在于标定的传送基坐标系 x 方向（即传送带运动方向）可能存在误差。重新标定传送带基坐标系时，注意第 3 点尽量离第 1 点远一些，第 4 点、第 1 点和第 3 点形成的角度尽量大一些。若通过参数对传送带基坐标系的方向进行微调需要了解跟踪原理，可使用标定的传送带基坐标系位姿创建一个普通工件 wobj1，并使机械臂 TCP 沿 wobj1 的 x 方向运动，以估测标定的传送基坐标系 x 方向与实际的传送带运动方向的误差。

5.6 错误处理

xcore 控制系统支持异常捕获机制，捕获形式为通过 try-catch 进行异常捕获，RL 程序执行 try{...} 以内的程序遇到异常情况时，机器人停止 try 中正在执行程序，进入 catch(error e){...} 执行异常处理程序，可在异常处理程序中进行打印错误信息等操作。

示例

```
try{
  ActUnit( "conveyor1" )
  WaitWObj wobj_cnv      //如果在 WaitWobj 时超时，停止 try 内程序执行，进入 catch{}
  MoveL p1,v1000,fine,tool1,wobj_cnv
  DropWobj
```

```
    DeactUnit( "conveyor1" )
}
catch(error e){
    print(e)                //打印错误信息
}
endtry
```

5.6.1 错误“工件超出启动窗口”

在 WaitWobj 时，控制系统发现关联工件已经超出启动窗口时，发生此错误；
遇到该错误，弹窗报警“工件超出启动窗口”；直接等待下一个工件；

5.6.2 错误“工件超出跟踪区”

跟踪过程中，还未结束对工件的操作过程（未 DropWobj 关联工件），工件超出了跟踪区域时，
发生此错误；

遇到该错误，未进行异常捕捉时，弹窗报警“跟踪工件已超出最大跟踪区域”；机器人停止。

若进行了异常捕捉时，异常支持被捕捉，异常代码：“Out MaxDistance”。

5.6.3 错误“WaitWobj 时，超过某时间依然没有关联工件”

WaitWobj 在时间限制内未关联工件时，发生此错误。

若用户没有进行异常捕捉，弹窗报警“等待工件超时”。

若进行了异常捕捉，异常支持被捕捉，异常代码“OutWaitTime”。

时间限制：用户可于传送带标定中手动输入进行时间限制设置，默认值为 60s。

若用户不想设置等待时间限制，则可以将该参数设为 0，则机器人不做等待时间限制。

5.6.4 错误“未 DropWobj 时，再次 WaitWobj”

未 DropWobj 掉关联工件，再次 WaitWobj 关联工件时，发生次错误。

若发生该种情况，用户未进行异常捕获，则弹窗报错：“工件关联两次”

若用户进行了异常捕获，则抛出异常代码“Connected Twice”。

5.7 样本程序

本节给出一个机械臂从传送带上抓取工件的简单示例。

1) 场景描述

一系列物品通过传送带输送，当到达机械臂工作区域后，机械臂需把物品从传送带上取下来，
并放置到固定空间位置。

2) 示例代码

```
GLOBAL PROC main()
  print("开始任务")
  ActUnit "conveyor1" // 激活传送带
  Confl Off // 取消构型监控
  WHILE(true)
    MoveJ pt_prepare, v1000, fine, tool1 // 移动到准备位置
    SetDO jiazhua, true // 打开夹爪
    WaitWObj wobj_cnv1 // 阻塞等待工件到来
    MoveL Offs(pt_cnv1, 0, 0, 20), v1000, z50, tool1, wobj_cnv1 // 移动到工件上的pt_cnv1点位上方
    MoveL pt_cnv1, v1000, fine, tool1, wobj_cnv1 // 移动到工件上的pt_cnv1点位
    Wait 0.1 // 跟踪中, 保持与工件的相对静止
    SetDO jiazhua, false // 关闭夹爪, 夹住物品
    Wait 0.1 // 跟踪中, 保持与工件的相对静止
    MoveL Offs(pt_cnv1, 0, 0, 20), v1000, fine, tool1, wobj_cnv1 // 移动到工件上的pt_cnv1点位上方
    DropWObj wobj_cnv1 // 机械臂停止跟踪
    MoveJ pt_place, v1000, fine, tool1 // 工件放置到空间固定位置
    SetDO jiazhua, true // 打开夹爪
    print("完成一次物品抓取")
  ENDWHILE
  DeactUnit "conveyor1"
  print("结束任务")
ENDPROC
```

图 15 样本程序

修订记录

版本	修订内容	时间
1.0	初版发行	2020.09.30
1.1	修订部分文字错误	2021.01.26
1.2	增加 3.6 版本内容和修订错误	2021.11.03

ROKAE 珞石



珞石机器人

400-010-8700
www.rokae.com
sales@rokae.com