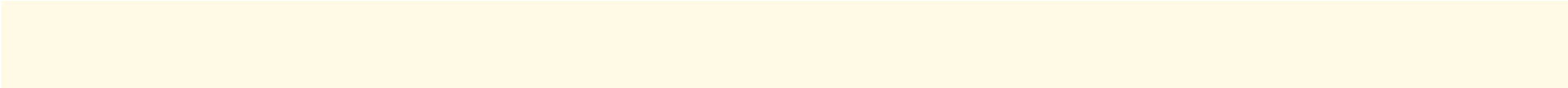


- KUKA机器人产品
- 产品特点
- 技术数据及系统构成方法
- 操作面板
- 座标系统
- 零点校正
- 工具测量
- 菜单
- 运动编程
- SPS编程

KUKA机器人产品



1985

- 第一台关节—非平行四边形—机器人



1996

- 首台PC界面



1998

- 首台达7M直径的大范围工作机器人
- 首台350KG 重载机器人



1999

- 首台INTERNET远程机器人



IR 600



IR 100



IR 300



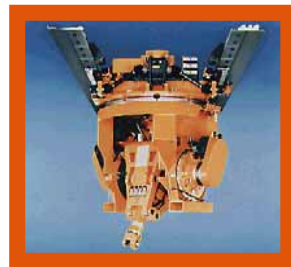
KR 6-350



1980

1990

2000



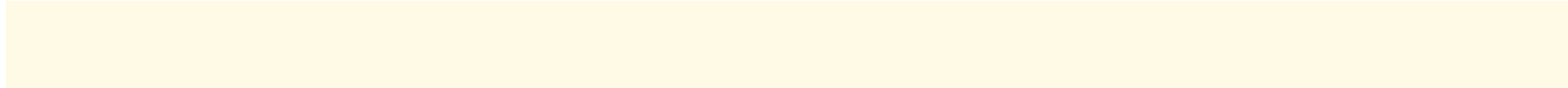
IR 200



IR 400



IR 700



KUKA robots are flexible

Application Automotive Industry



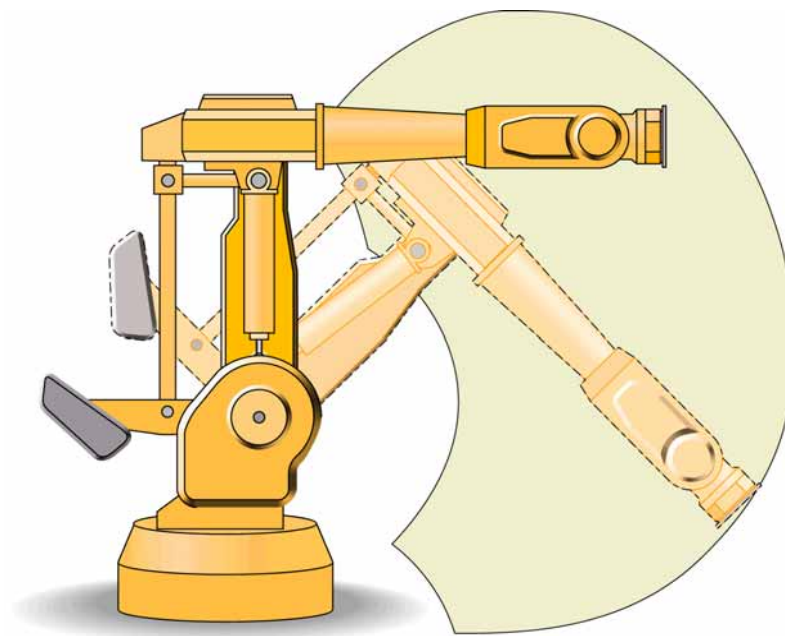
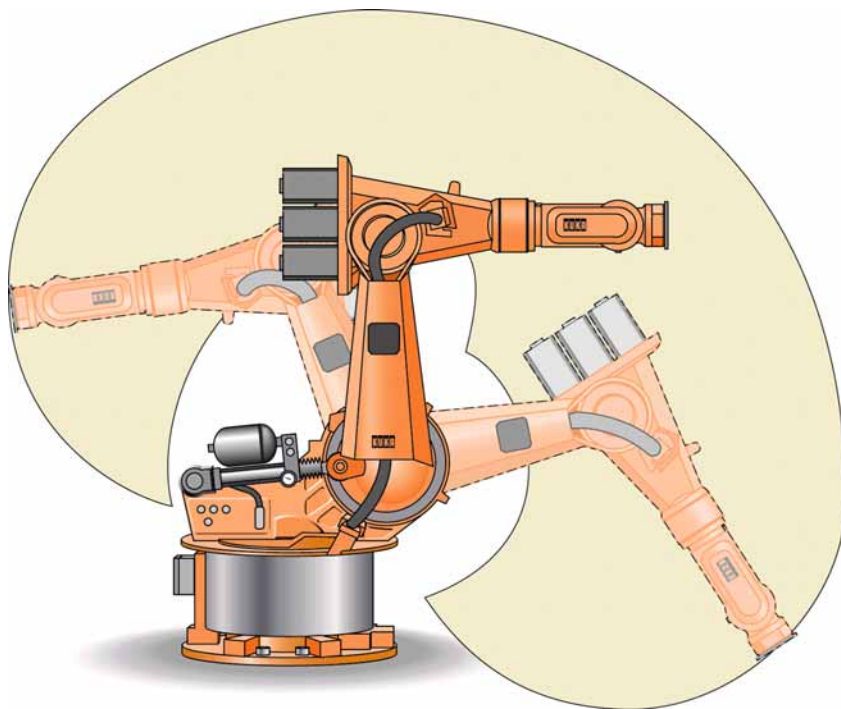
Application General Industry



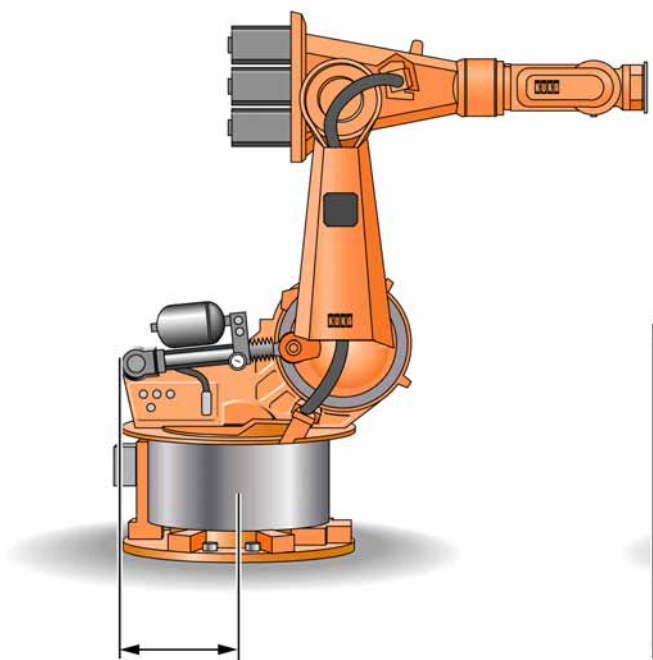
产品特点

优越的动力学性能

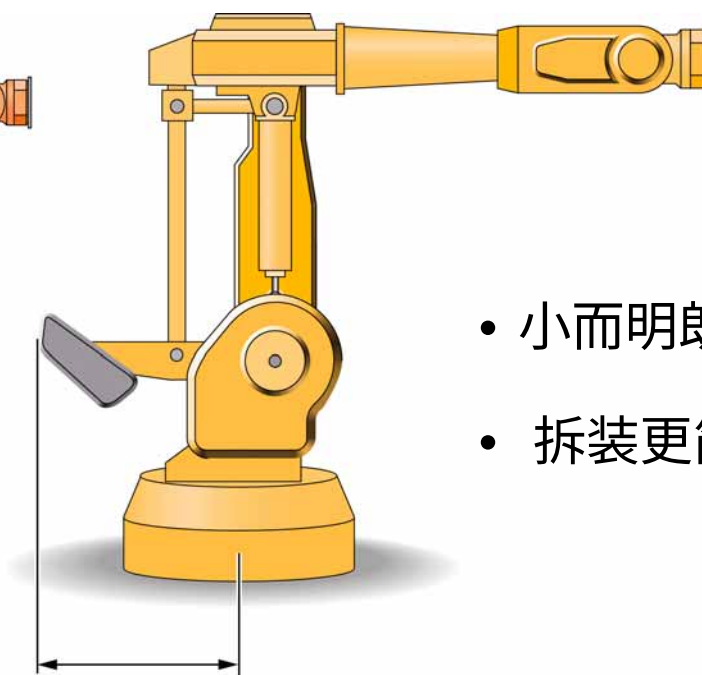
- 集中于上部运行
- 更大的运行空间



KUKA Robot



平行四边形 Robot



- 小而明朗的机械布置
- 拆装更简单

更广泛的通用性KR 125, KR 150 or KR 200

- 易于规划
- 柔性化
- 更多相同的机件



KR 125



KR 150



KR 200

KUKA 机器人系列



Lower Payload	Medium Payload	High Payload	Heavy Payload	Special Series
(V)KR 6/2 (V)KR 15/2	(V)KR 30/2 (V)KR 30 L15/2 (V)KR 45/2	(V)KR 125/2 (V)KR 150/2 (V)KR 200/2	(V)KR 350/2 (V)KR 350L280/2 (V)KR 350L240/2	(V)KR 60P/1 (V)KR 100P/1 (V)KR 100PA/1 (V)KR 160PA/1

重载机器人 (V)KR 350/2



(V)KR 60 P/2, (V)KR 100 P/2



KR 180 PA



架上安装robot (V)KR 125 K/1, (V)KR 150 K/1



架上安装 robot KR 30 K



墙上安装 robot (V)KR 125 W/2



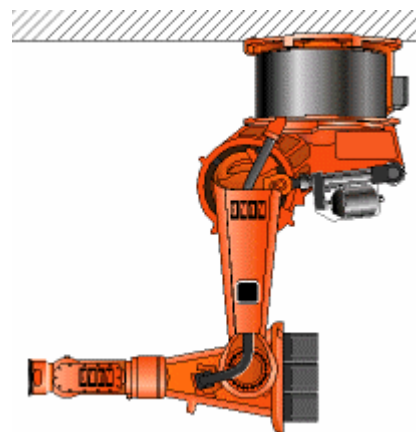
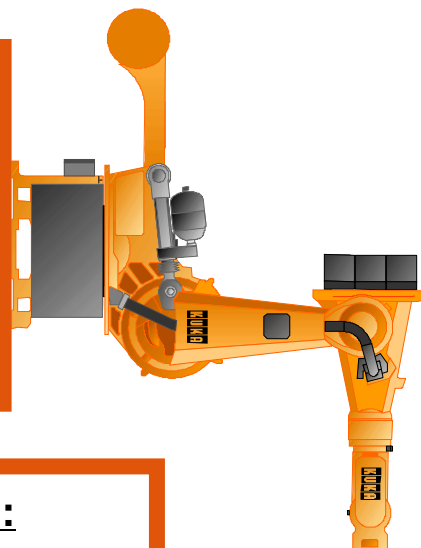
MINI robot KR 3



柔性化的安装

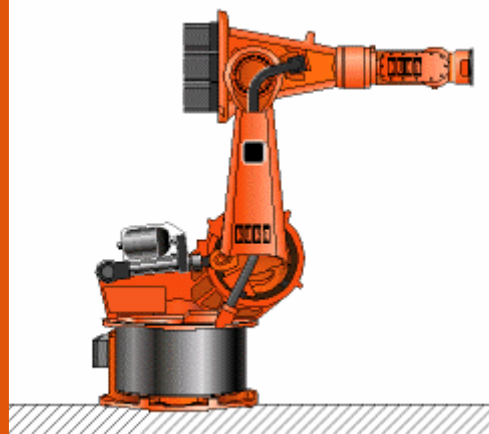
Wall mounting:

-  (V)KR 6/2
-  (V)KR 15/2
-  (V)KR 30/2
-  (V)KR 30L15/2
-  (V)KR 125 W/2



Floor mounting:

-  (V)KR 6/2
-  (V)KR 15/2
-  (V)KR 15 L2/2
-  (V)KR 30/2
-  (V)KR 30 L15/2
-  (V)KR 45/2
-  (V)KR 125/2
-  (V)KR 150/2, 200/2
-  (V)KR 350/2
-  KL 250, KL 1500



Ceiling mounting:

-  (V)KR 6/2
-  (V)KR 15/2
-  (V)KR 15 L2/2
-  (V)KR 30/2
-  (V)KR 30 L15/2
-  (V)KR 45/2
-  (V)KR 125/2
-  (V)KR 150/2, 200/2
-  (V)KR 350/2
-  KL 250, KL 1500

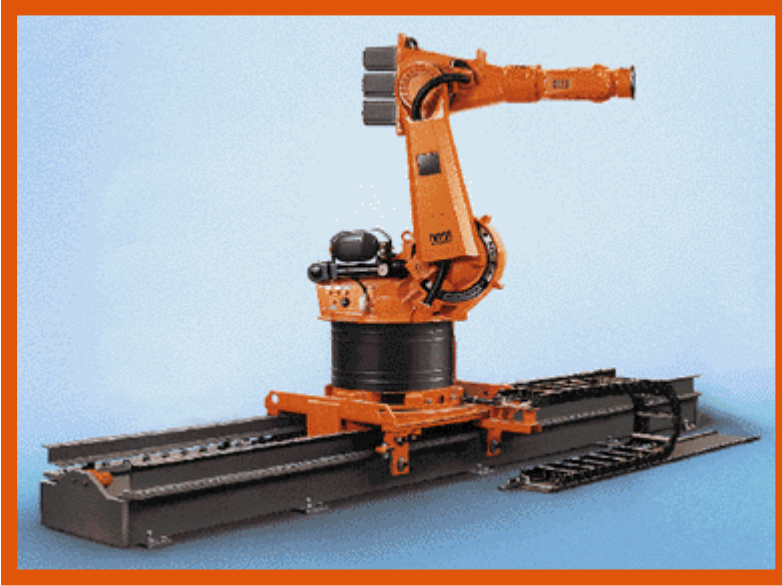
行走单元 (KL)

提高了自由度



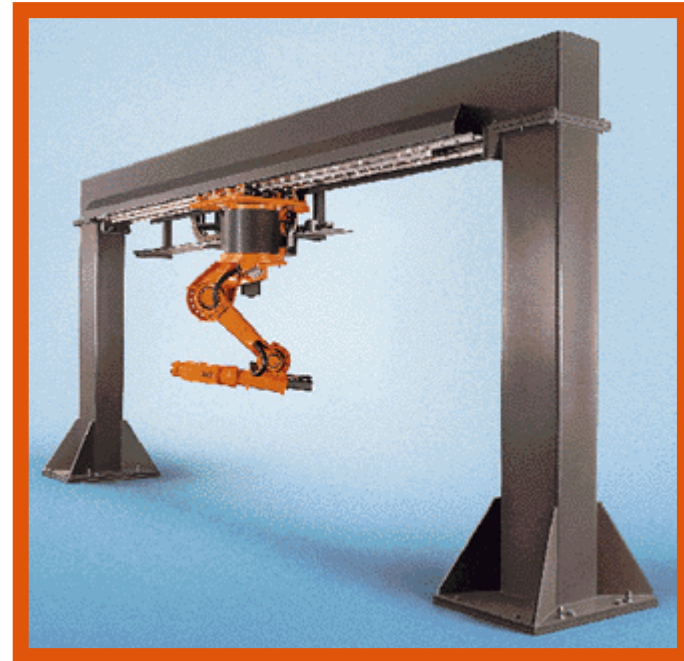
KL 250 (max. weight 250kg) 适合于KR15及以下的机器人
KL 1500 (max. weight 1500kg) 适合于KR30及以上机器人

KL Mounting positions

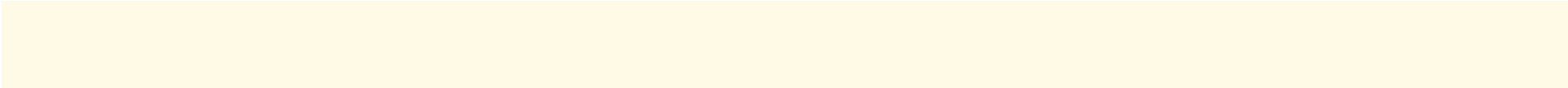


地面安装

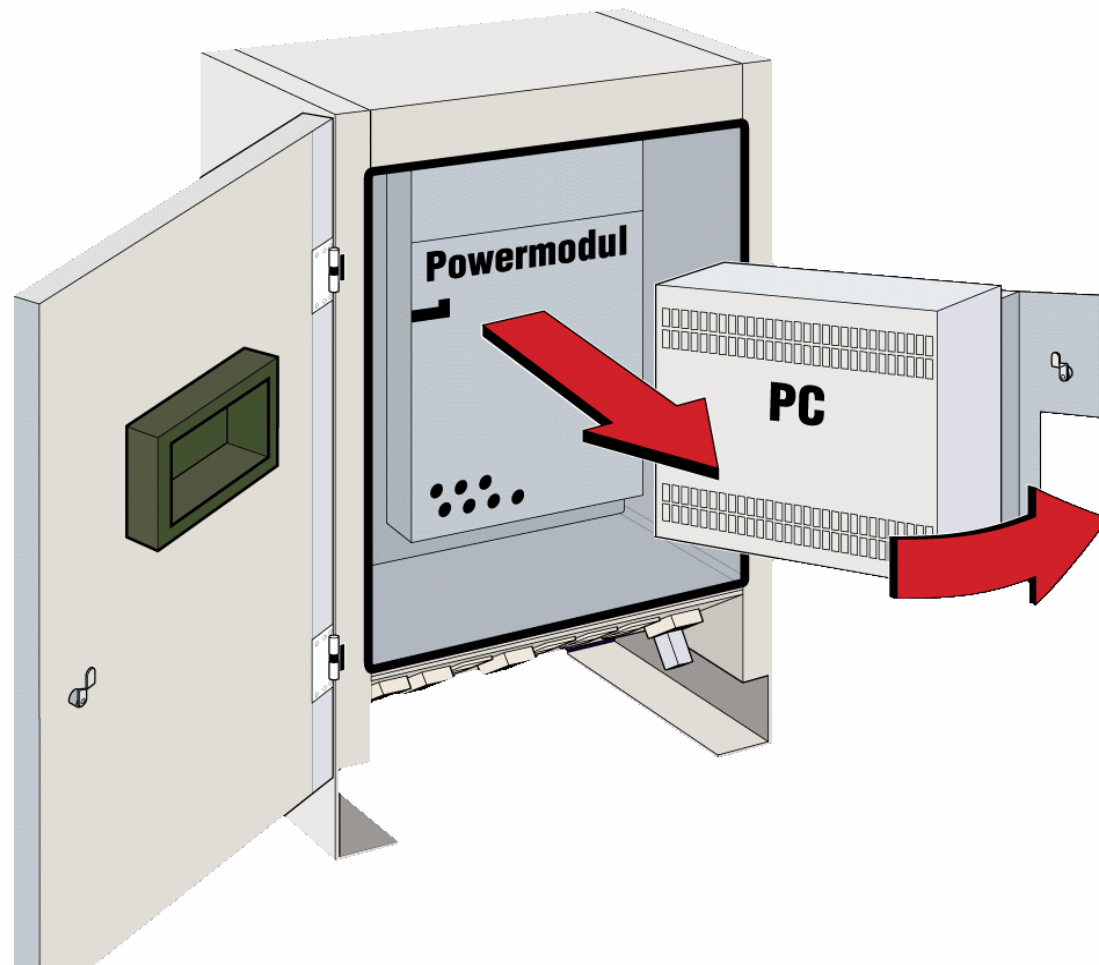
天花板安装



技术数据系系统构成方法



控制柜



技术数据 - KR C1

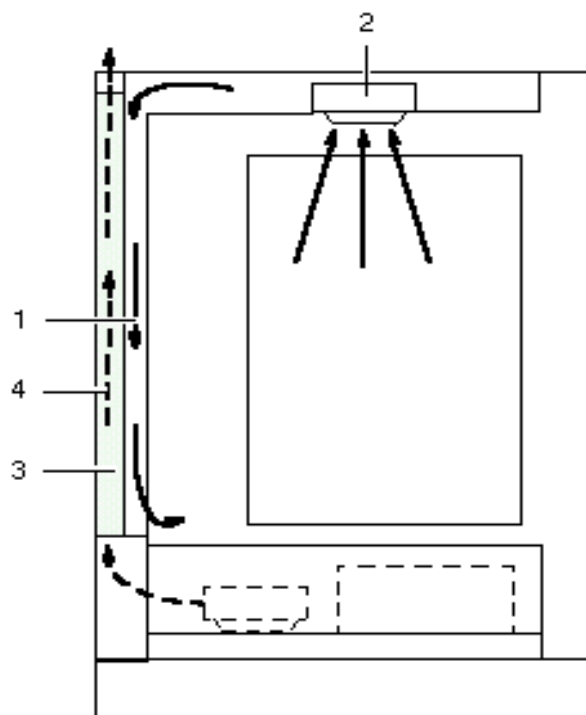


- 类型: KR C1
 - 最多6轴
- 环境 :
 - 无冷却系统: max. 45°C
 - 有冷却系统: max. 55°C
- 重量: 136kg
- 电源: 3x400V
- 处理器: Pentium
- RAM: 64MB

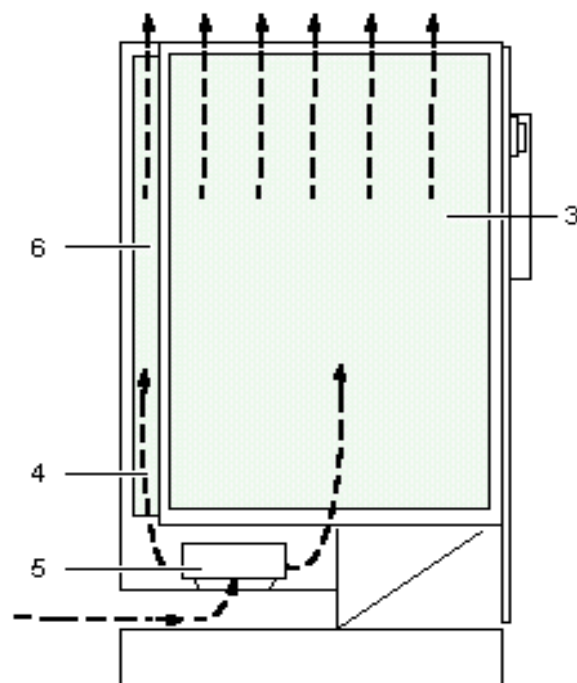
安装位置 KRC1



控制柜冷却



内/外冷却系统前视



外冷却系统侧视

1 内部冷却

2 内部冷却风扇

3 风道

4 外部冷却通道

5 外部冷却风扇

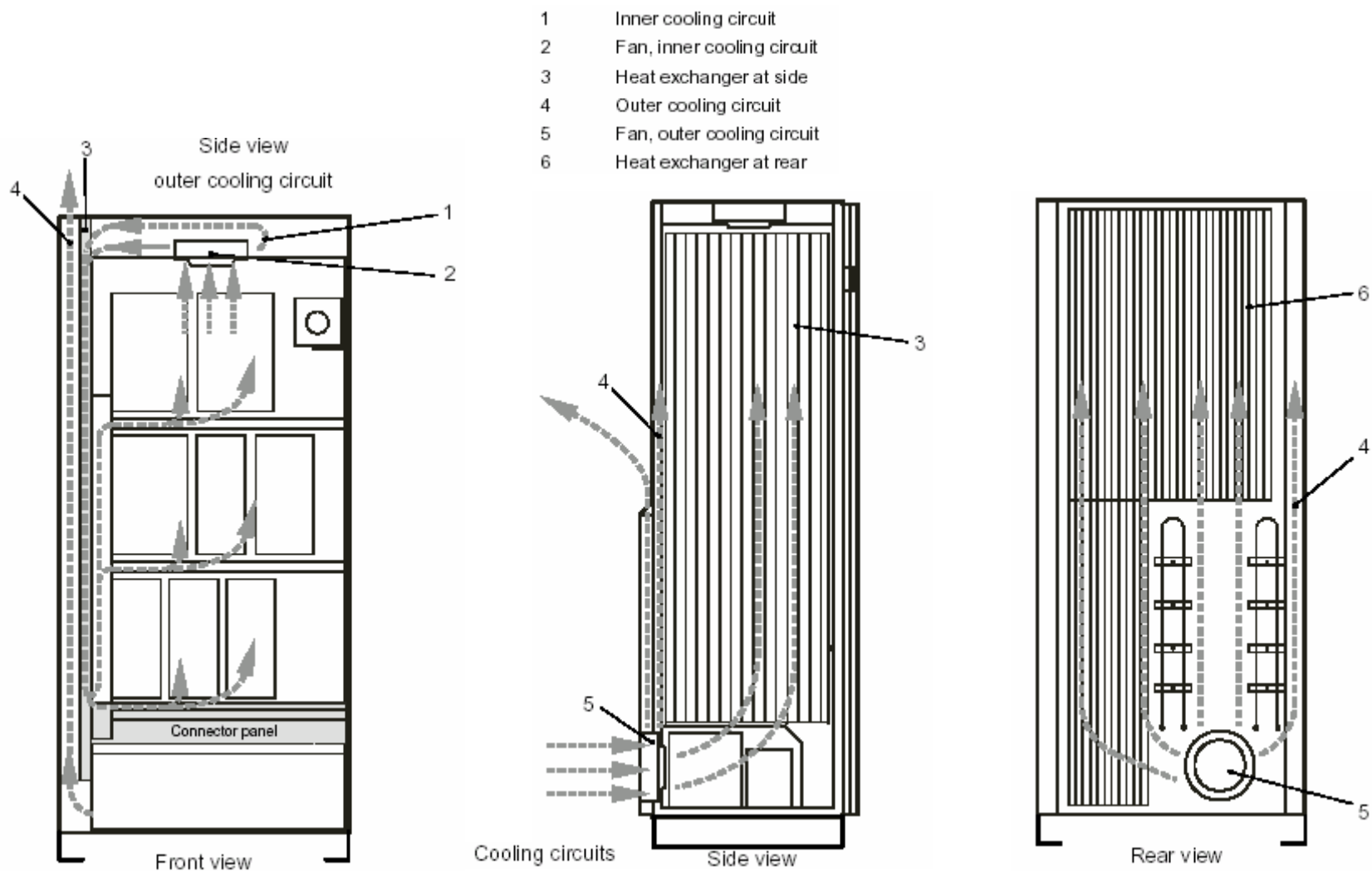
6 风道

KR C1A – 技术数据

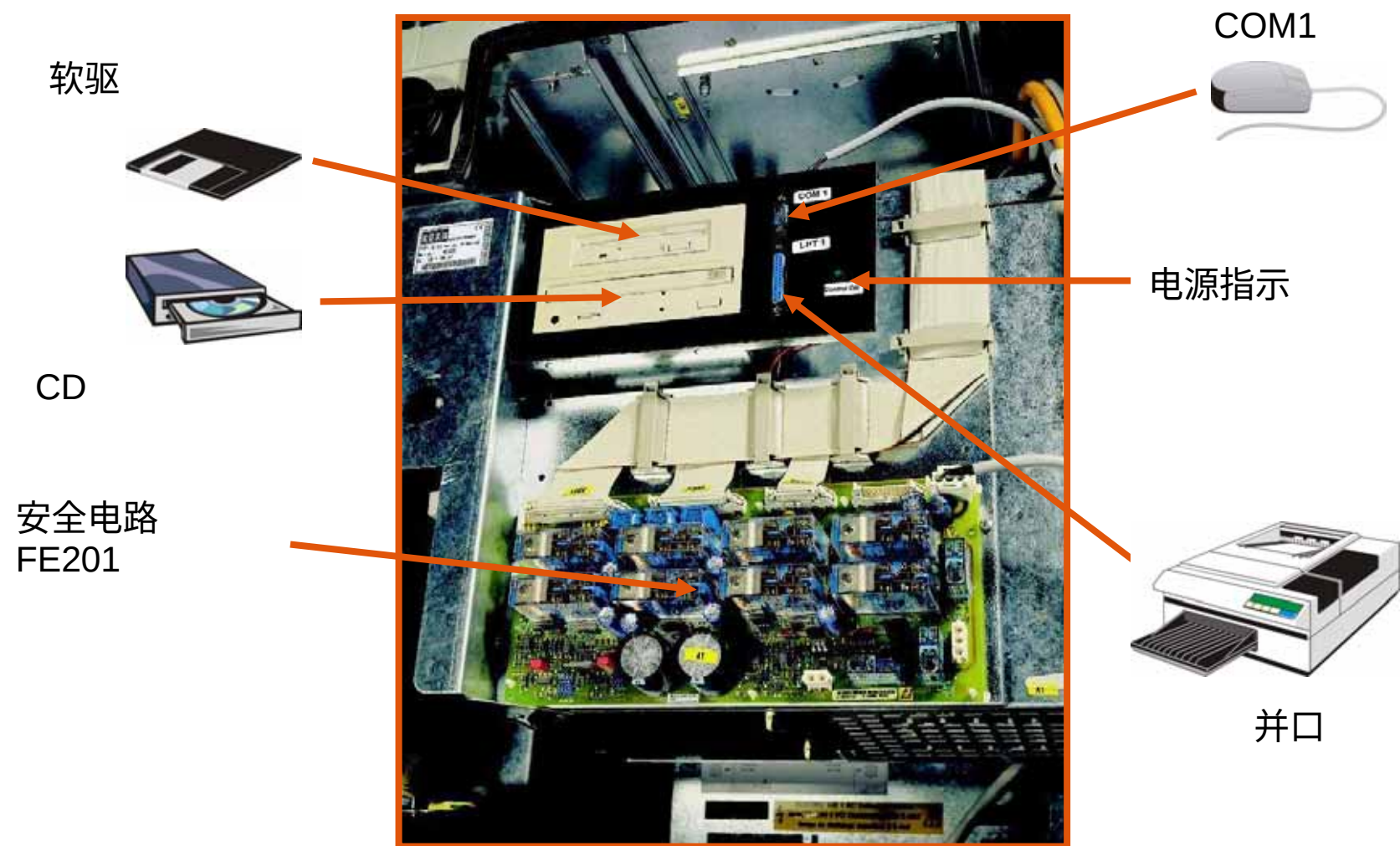


- 类型: KR C1
 - 最多8轴
- 环境:
 - 无冷却: max. 45°C
 - 有冷却: max. 55°C
- 重量: 323kg
- 电源: 3x400V
- 处理器: Pentium
- RAM: 64MB

KR C1A冷却系统



PC单元前视



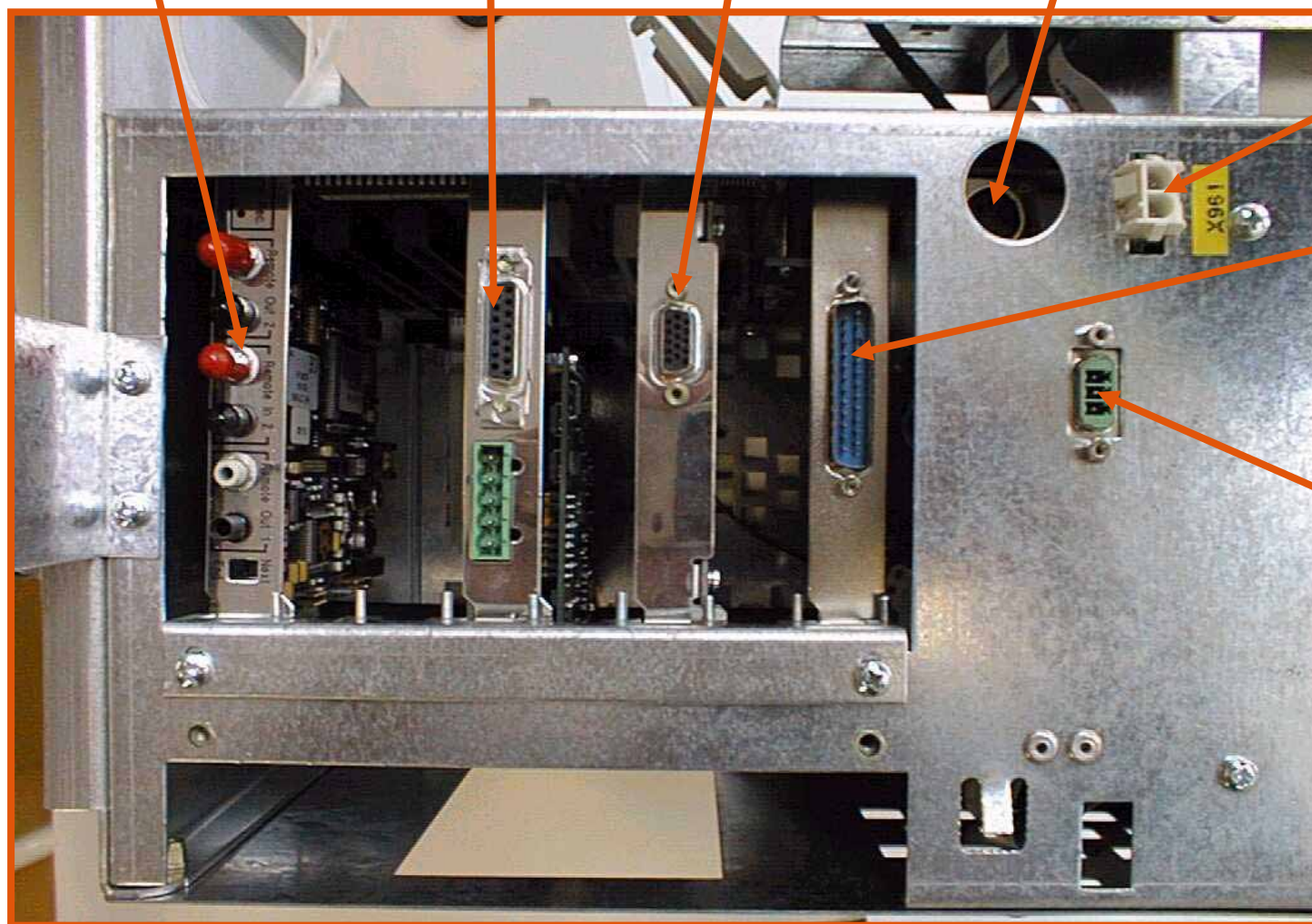
PC单元上视

总线

MFC 卡

KVGA 卡

键盘接口

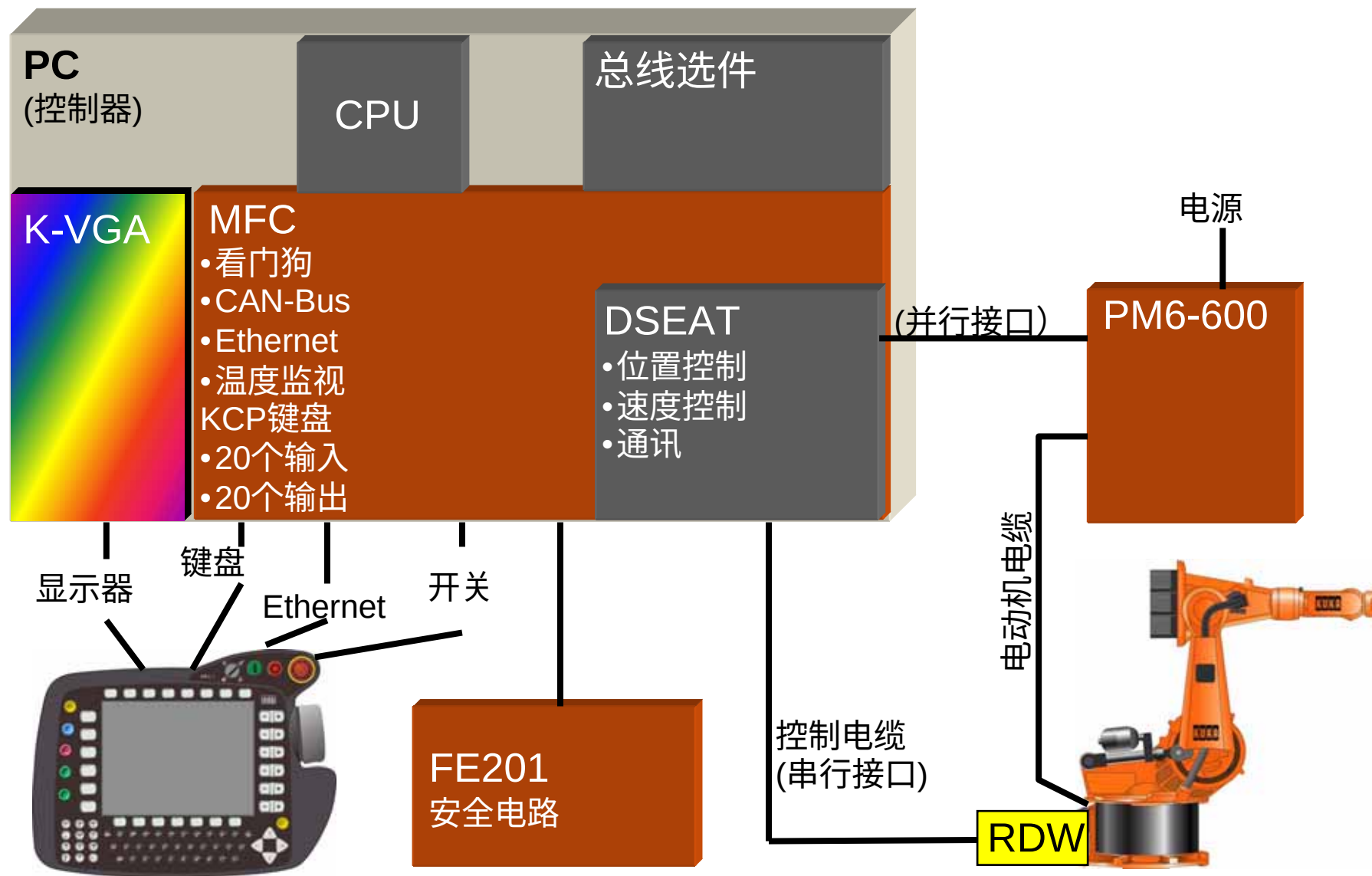


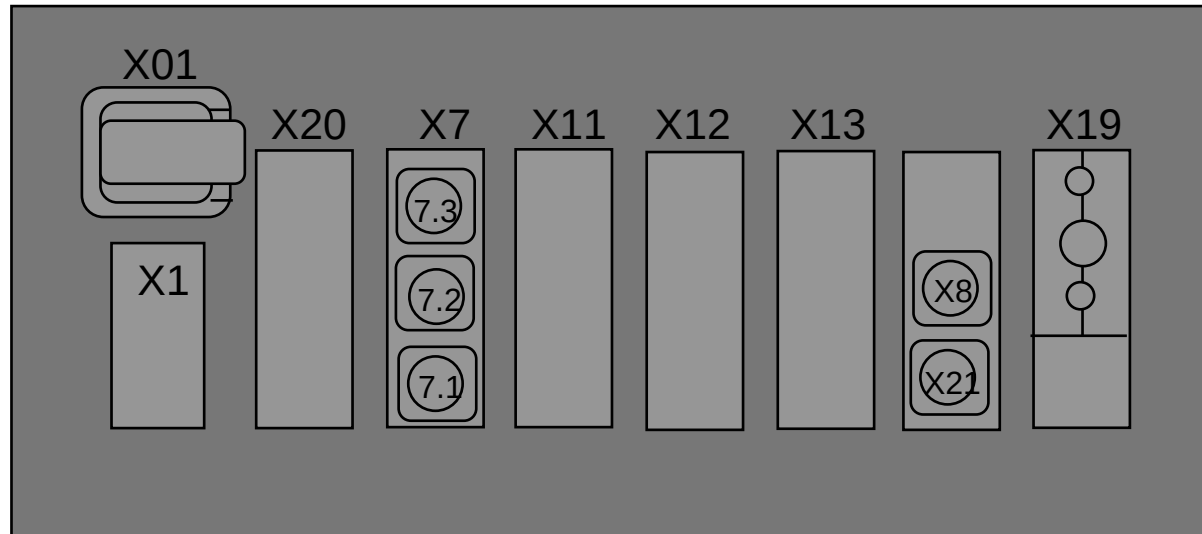
X961
PC电源

COM2

INTER外部电源

KUKA 硬件





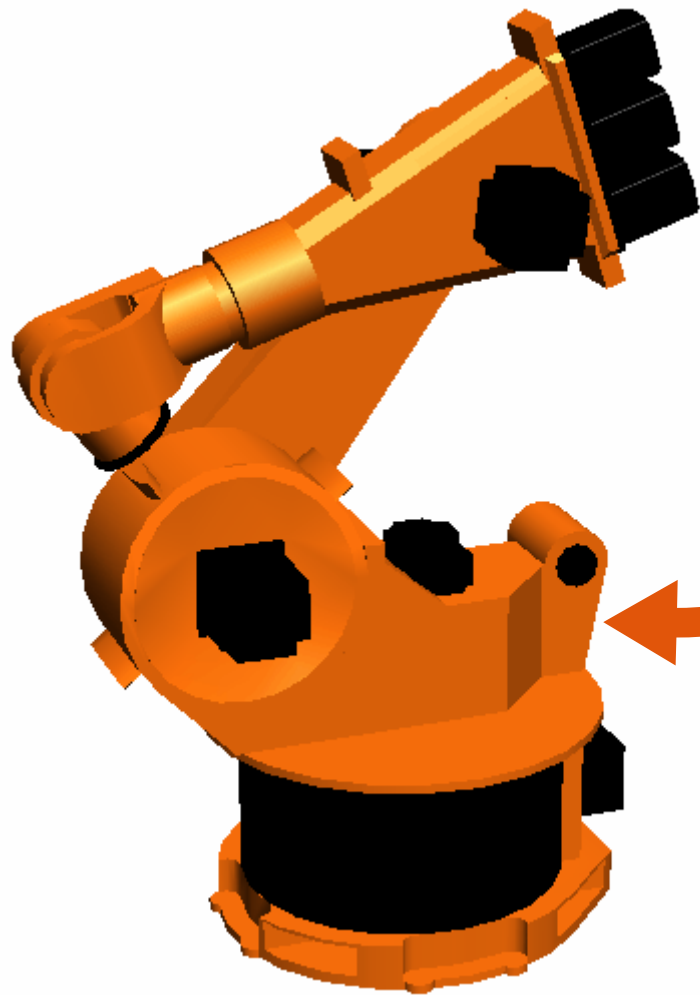
X1:	供电电源	X11:	外围接口
X01:	维修电源	X12:	外围接口 (选件)
X20:	电动机电缆 A1-A6	X13:	外围接口 (选件)
X7.1:	电动机电缆 A7 (Option)	X21:	反馈电缆 A1-A8
X7.2:	电动机电缆 A8 (Option)	X8:	反馈电缆 A7-A12
X7.3:	电动机电缆 A9 (Option)	X19:	KCP 接口

控制柜系列号



Serial number

机器人系列号

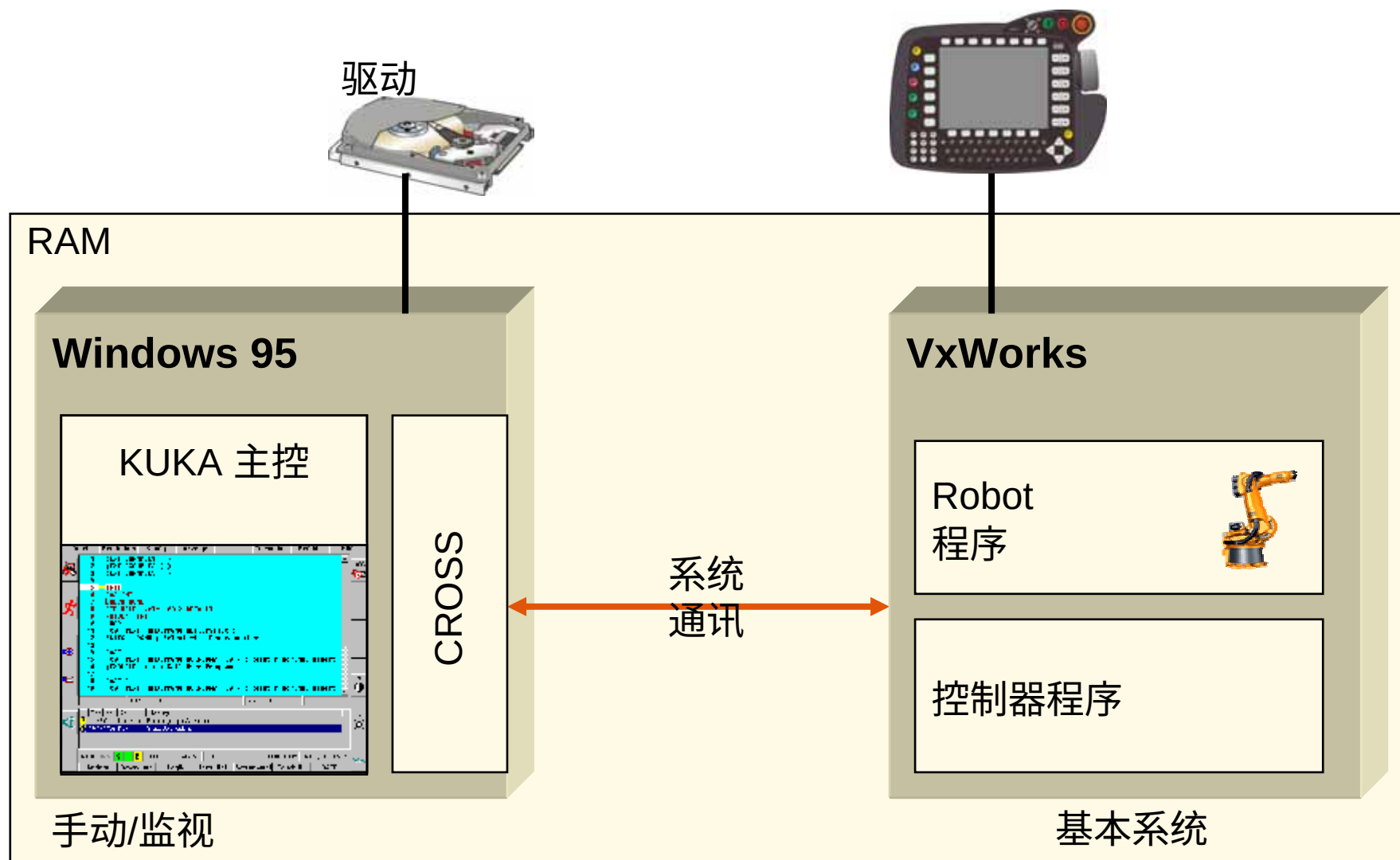


KUKA Roboter GmbH Augsburg / Germany			KUKA
Typ	Type	Type	VKR125/1
Werk-Nr.	Serial No.	No. Série	722025
Baujahr	Date	Année de Fab.	1997
Zeichn.-Nr.	Draw. No.	Plan No.	391.984
Gewicht	Weight	Poids	975/1120 kg

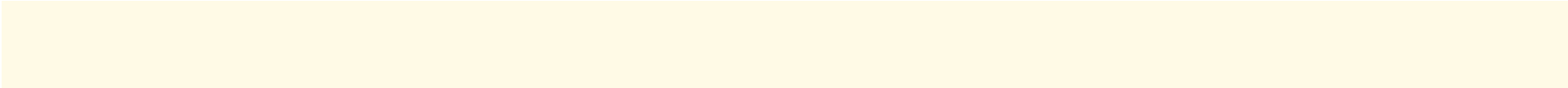
D/GB/F

Serial number

软件框架



操作面板



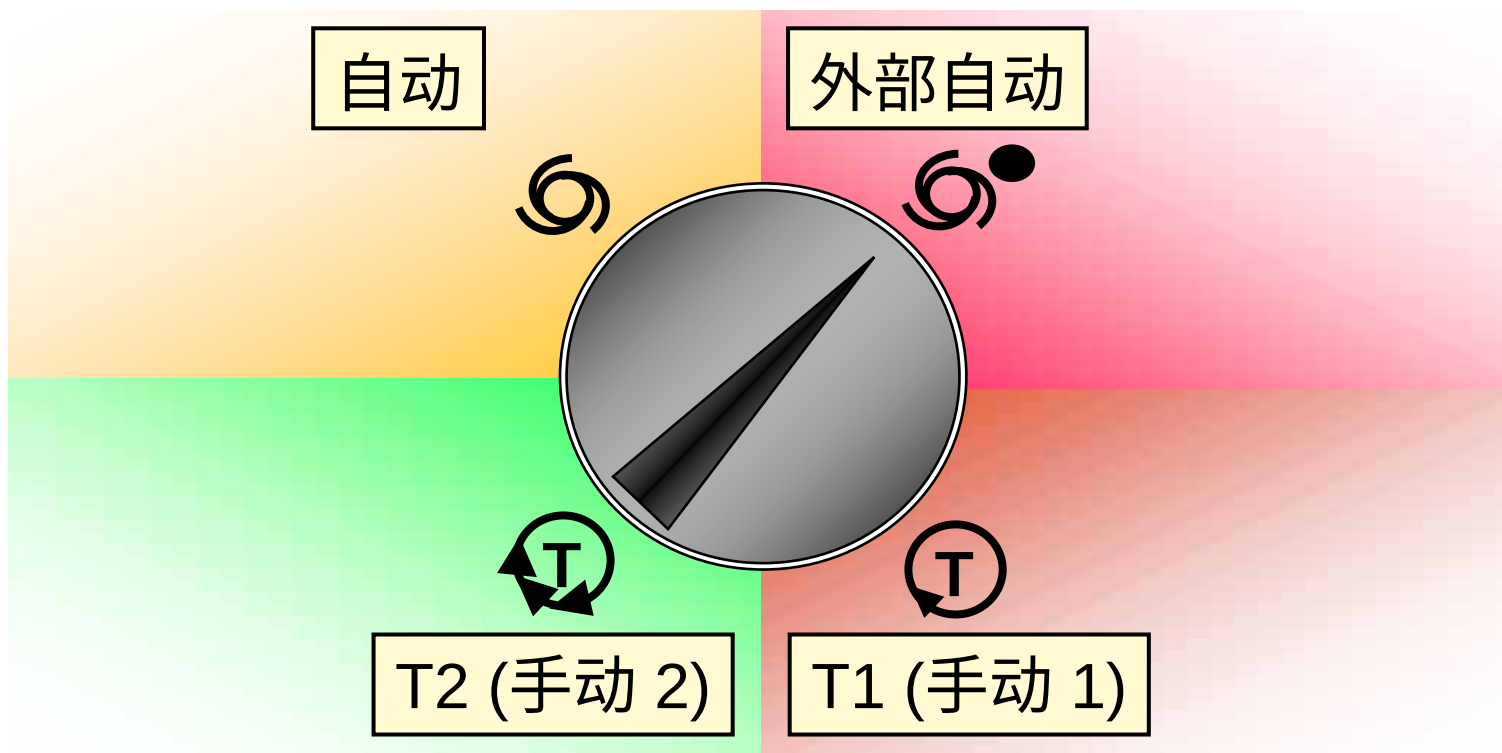
KUKA 操作面板(KCP)



基本操作条件



操作模式



操作模式

Operation mode switch	T1	T2	AUTOMATIC	AUTOMATIC EXTERN
软键操作  鼠标操作  HOV	250 mm/s 使能 (手动)	250 mm/s 使能 (手动)	无效	无效
执行程序 POV	250 mm/s 使能 (手动) 按起动键	编程速度 使能 (手动) 按起动键	编程速度 内部模拟 驱动ON	编程速度 外部控制



Esc key

窗口选择键

显示窗口

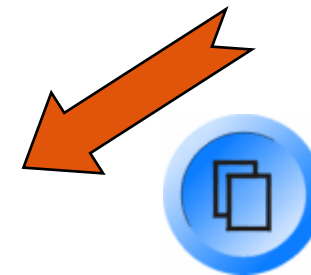
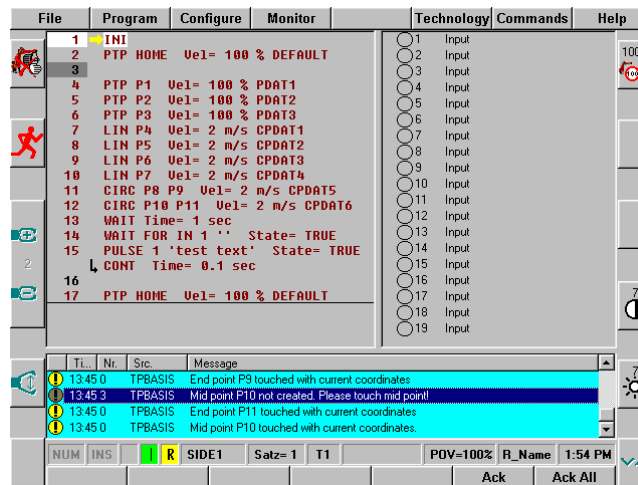
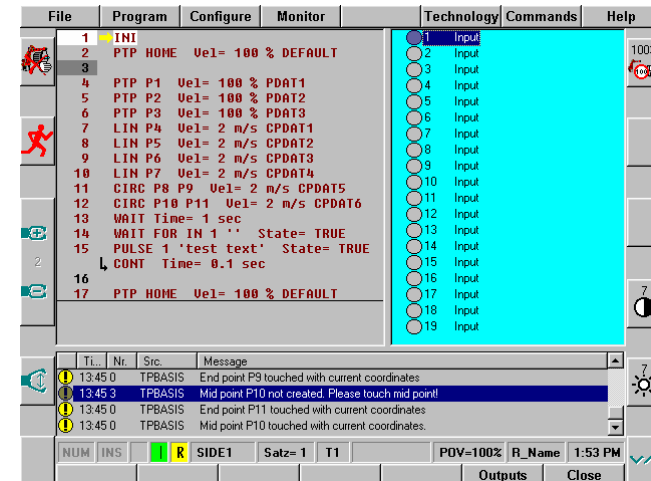
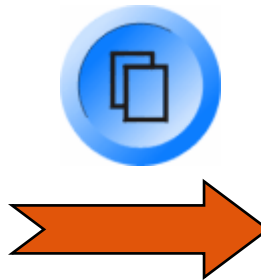
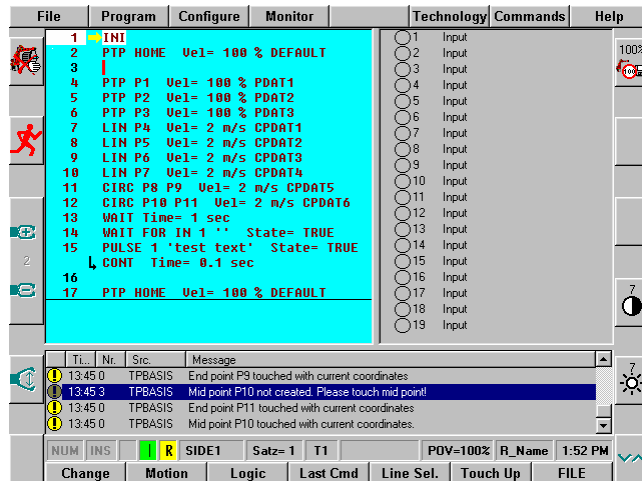
The screenshot shows a robotic control interface with a menu bar at the top: Datei, Bearbeiten, Konfigurier., Anzeige, Technolog., Befehle, and Hilfe. The main area is divided into three sections:

- Program Window (程序窗口):** A green grid area containing a list of program steps. The first step is highlighted with a yellow arrow.
- Status Window (状态窗口):** A blue area on the right showing a list of 19 outputs (Ausgang) with corresponding status indicators (circles). The 19th output is highlighted with a red circle.
- Information Window (信息窗口):** A yellow area at the bottom showing a table of messages (Meldung) with columns for time (Zeit), number (Nr.), absolute position (Abs.), and message (Meldung).

At the bottom of the interface, there is a status bar with various indicators and buttons:

- Buttons: NUM, INS, S, R, QUADRAT2, Satz= 1, T1, POV=100%, Robi 10, 14:37, Ändern, Bewegung, Logik, letzter Bef., Satzanwahl, Touch Up, DATEI.
- Icons: A green checkmark and a blue arrow.

窗口选择键



程序窗口

程序执行行提示

光标

行号

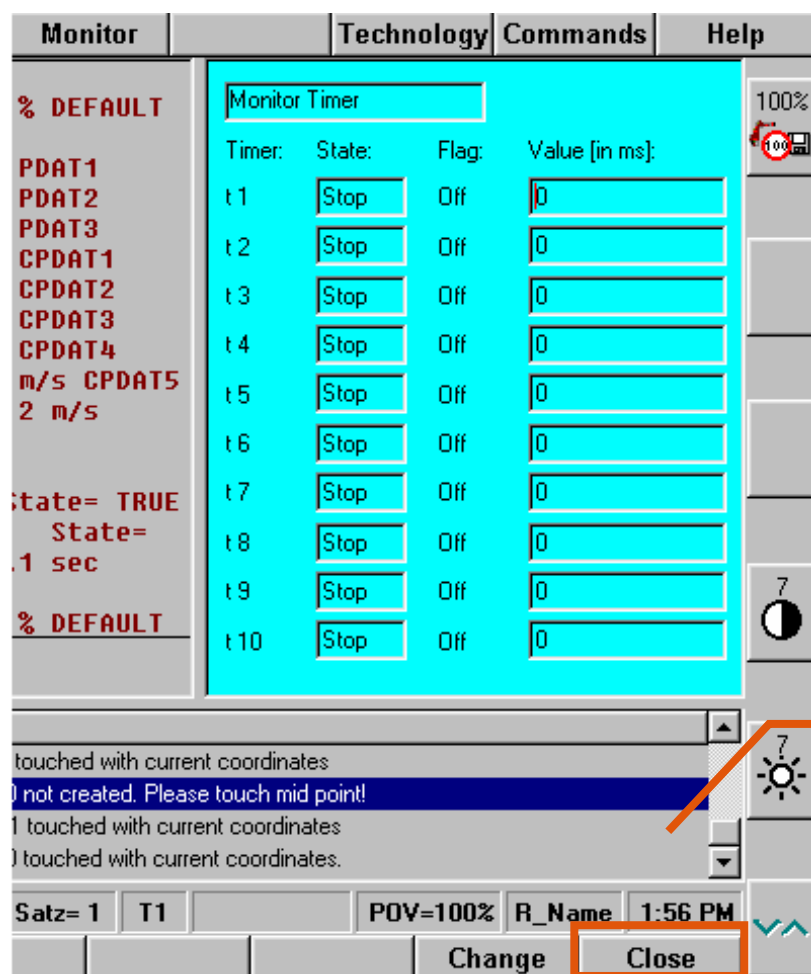
File	Program	Configuration	Log	Commands	Help
1	INI				
2	PTP HOME Vel= 100 % DEFAULT				
3					
4	PTP P1 Vel= 100 % PDAT1				
5	PTP P2 Vel= 100 % PDAT2				
6	PTP P3 Vel= 100 % PDAT3				
7	LIN P4 Vel= 2 m/s CPDAT1				
8	LIN P5 Vel= 2 m/s CPDAT2				
9	LIN P6 Vel= 2 m/s CPDAT3				
10	LIN P7 Vel= 2 m/s CPDAT4				
11	CIRC P8 P9 Vel= 100 % PDAT1				
12	CIRC P10 P11 Vel= 100 % PDAT2				
13	WAIT Time= 1 sec				
14	WAIT FOR IN 1 ' ' State= TRUE				
15	PULSE 1 'test text' State= TRUE				
16	CONT Time= 0.1 sec				
17	PTP HOME Vel= 100 % DEFAULT				

Ti...	Nr.	Src.	Message
13:45 0	TPBASIS	End point P9 touched with current coordinates	
13:45 3	TPBASIS	Mid point P10 not created. Please touch mid point!	
13:45 0	TPBASIS	End point P11 touched with current coordinates	
13:45 0	TPBASIS	Mid point P10 touched with current coordinates.	

NUM INS SIDE1 Satz= 1 T1 POV=100% R_Name 1:52 PM

Change Motion Logic Last Cmd Line Sel. Touch Up FILE

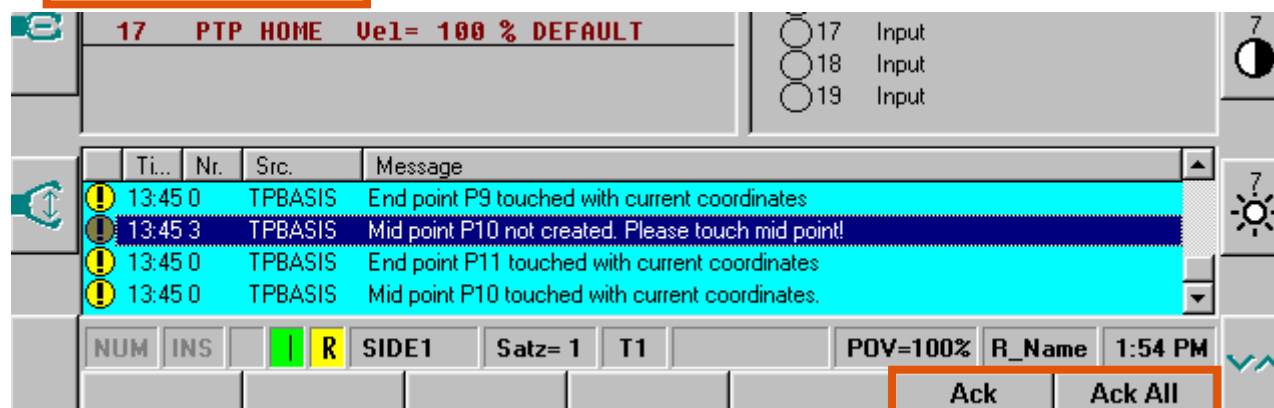
状态窗口



状态窗口内容选择.

状态窗口关闭.

窗口信息.



信息应答

Marks of the messages



提示

- e.g. “请求起动”



指导

- e.g. “急停”



必须应答

- e.g. “应答 E.-Stop”



等待

- e.g. “等待 SRB ”

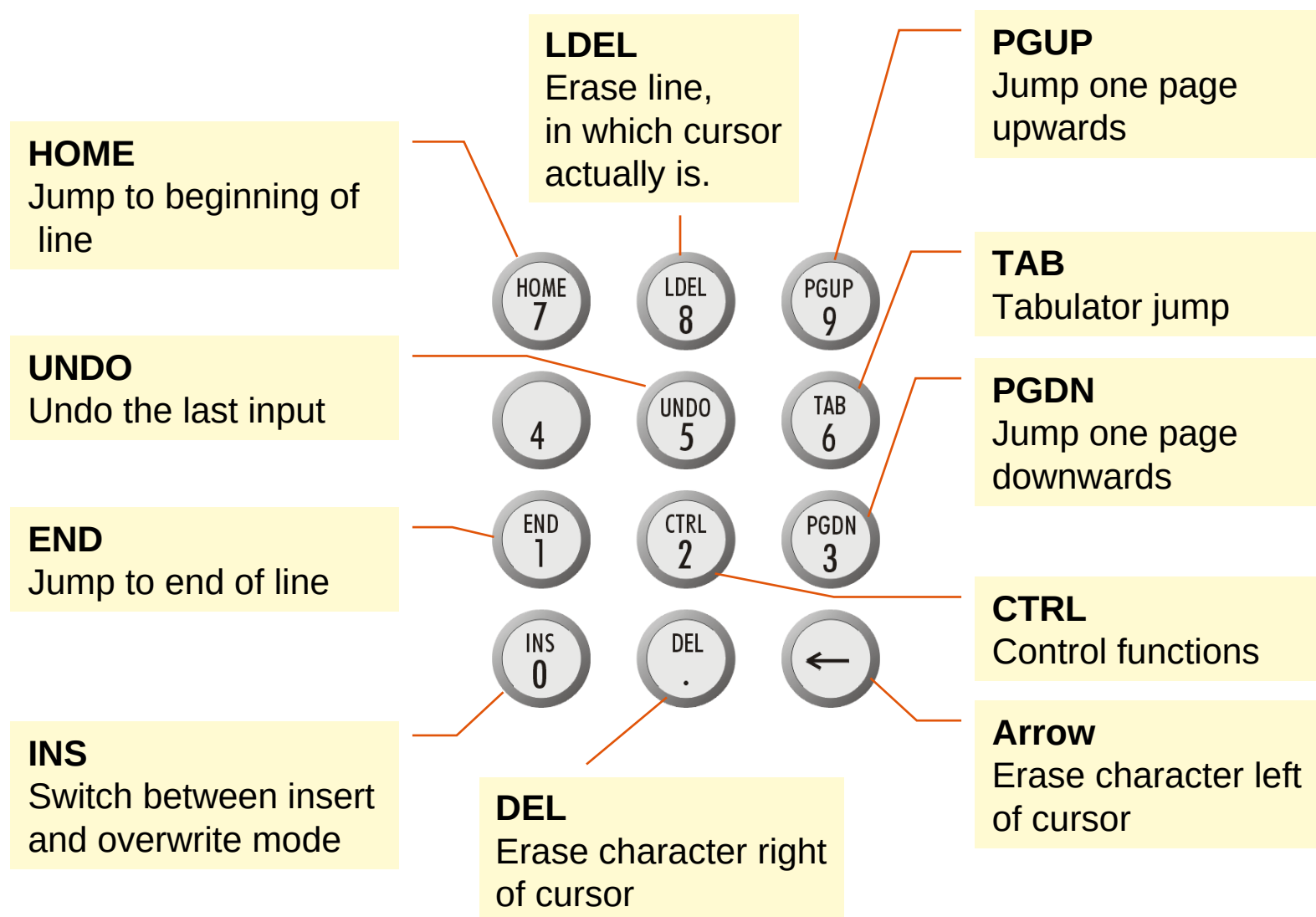


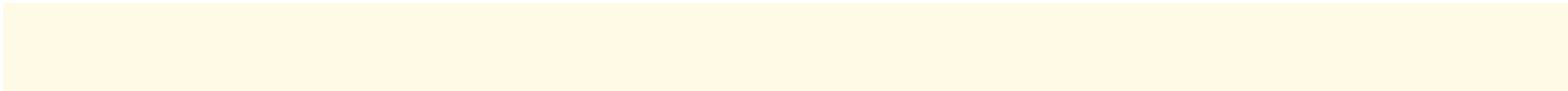
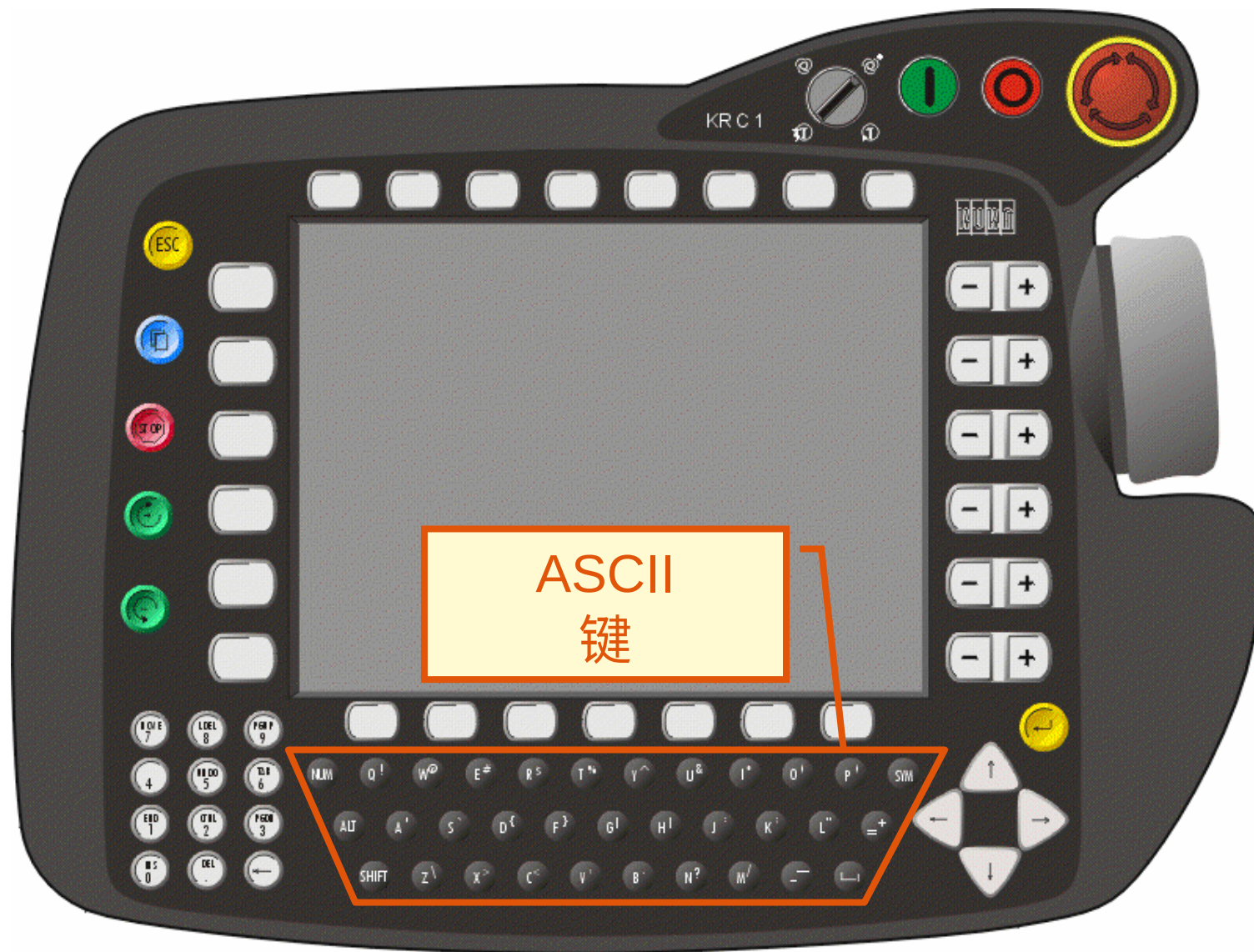
提问

- e.g. “要备份示教点吗 ?”

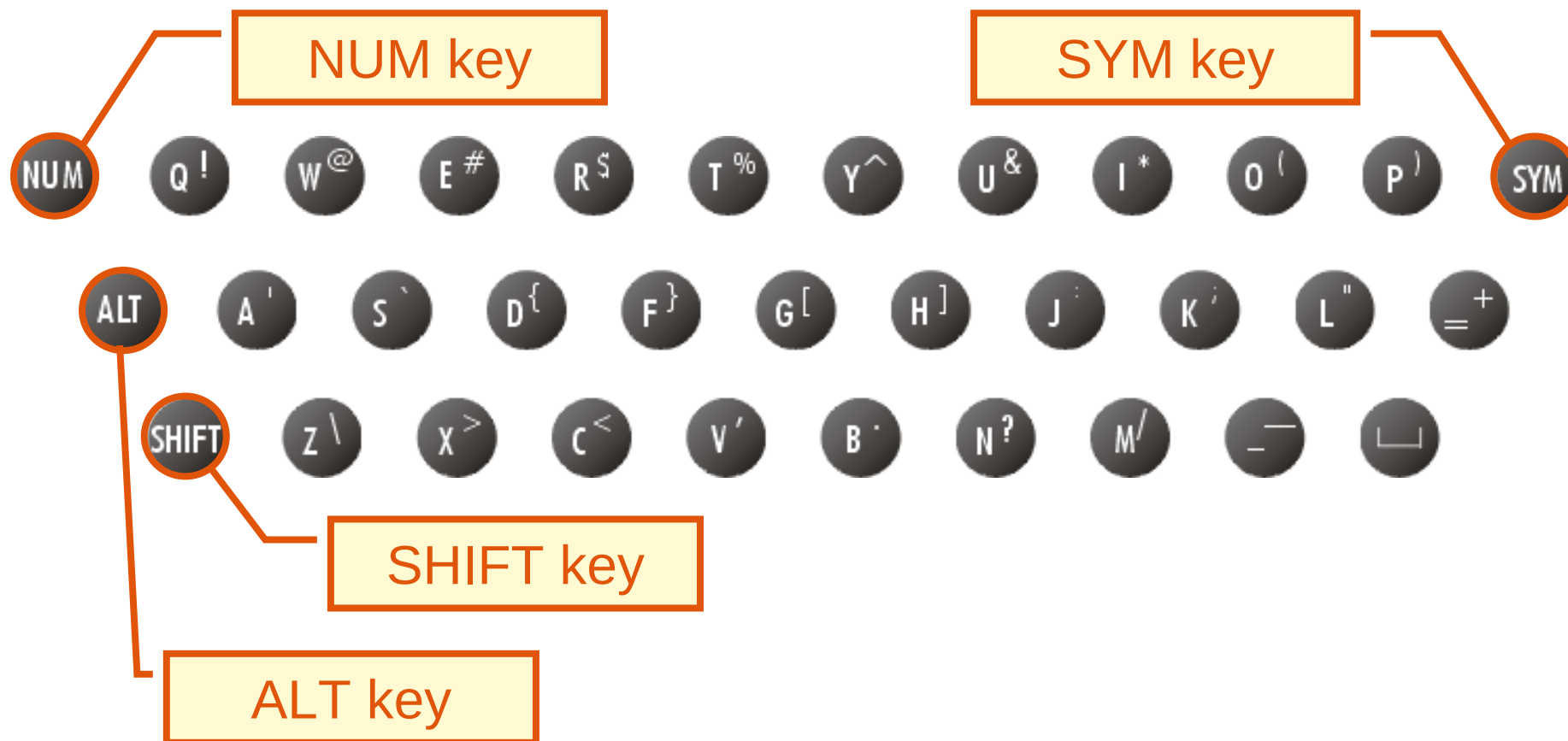


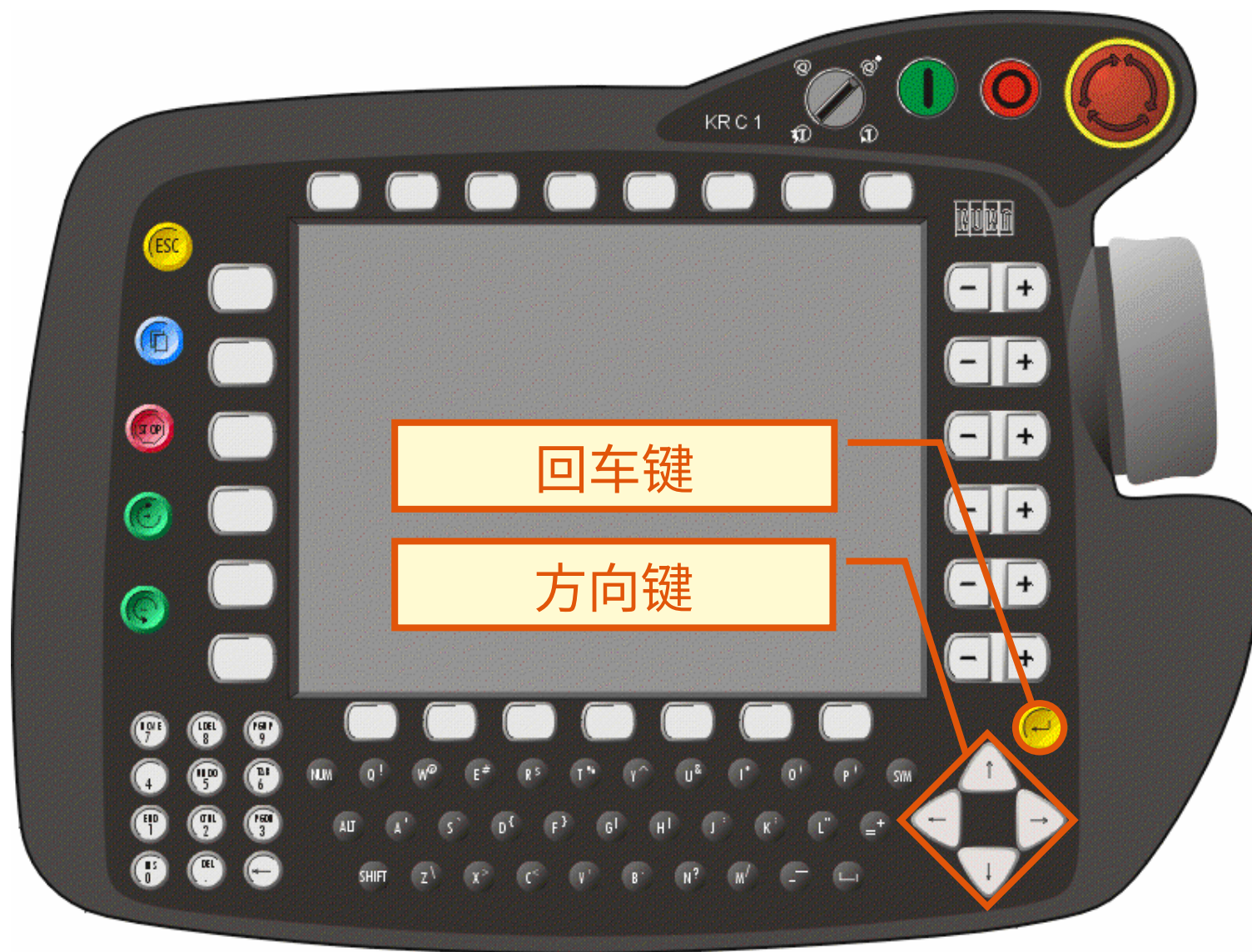
数字键

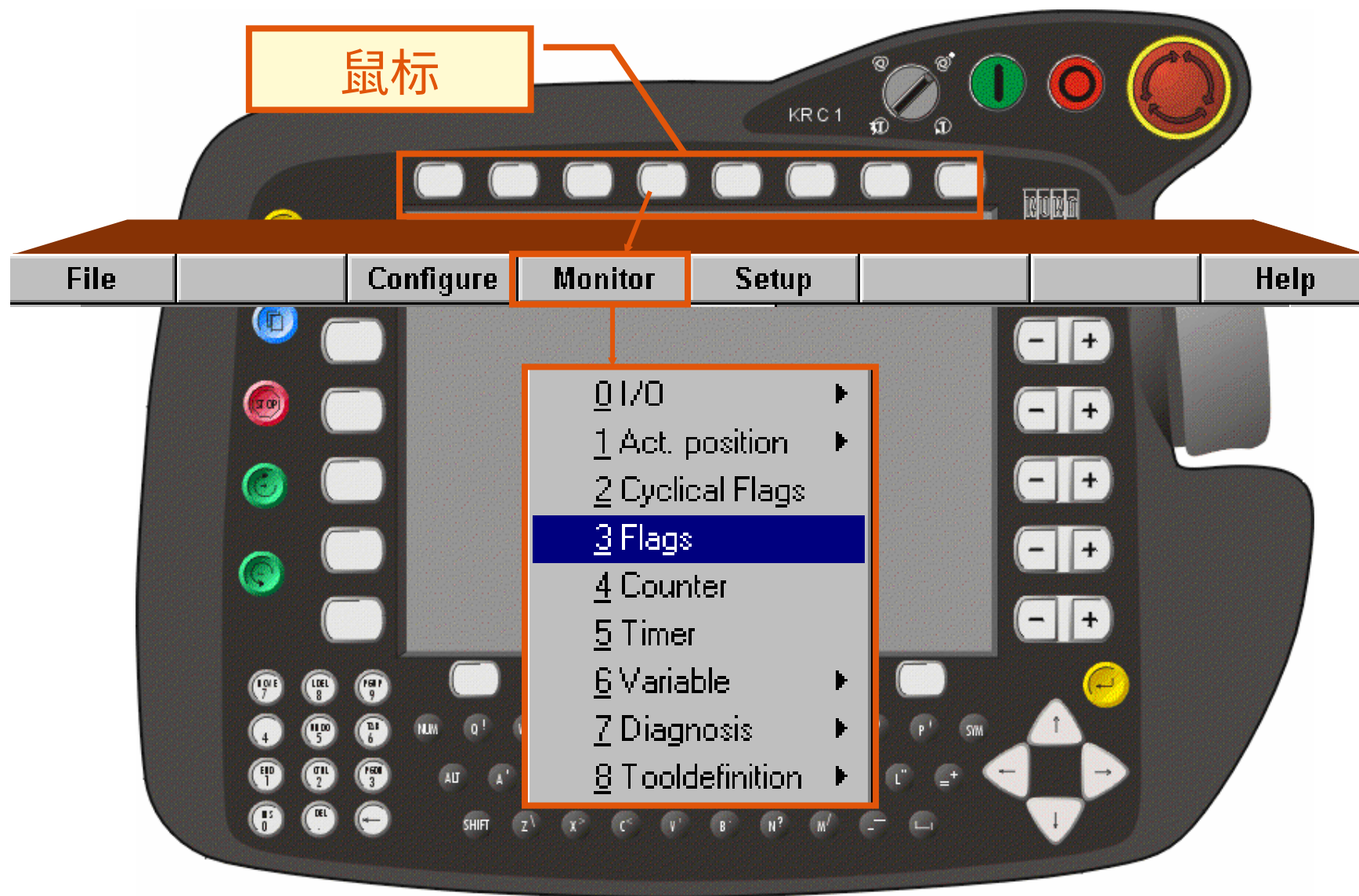




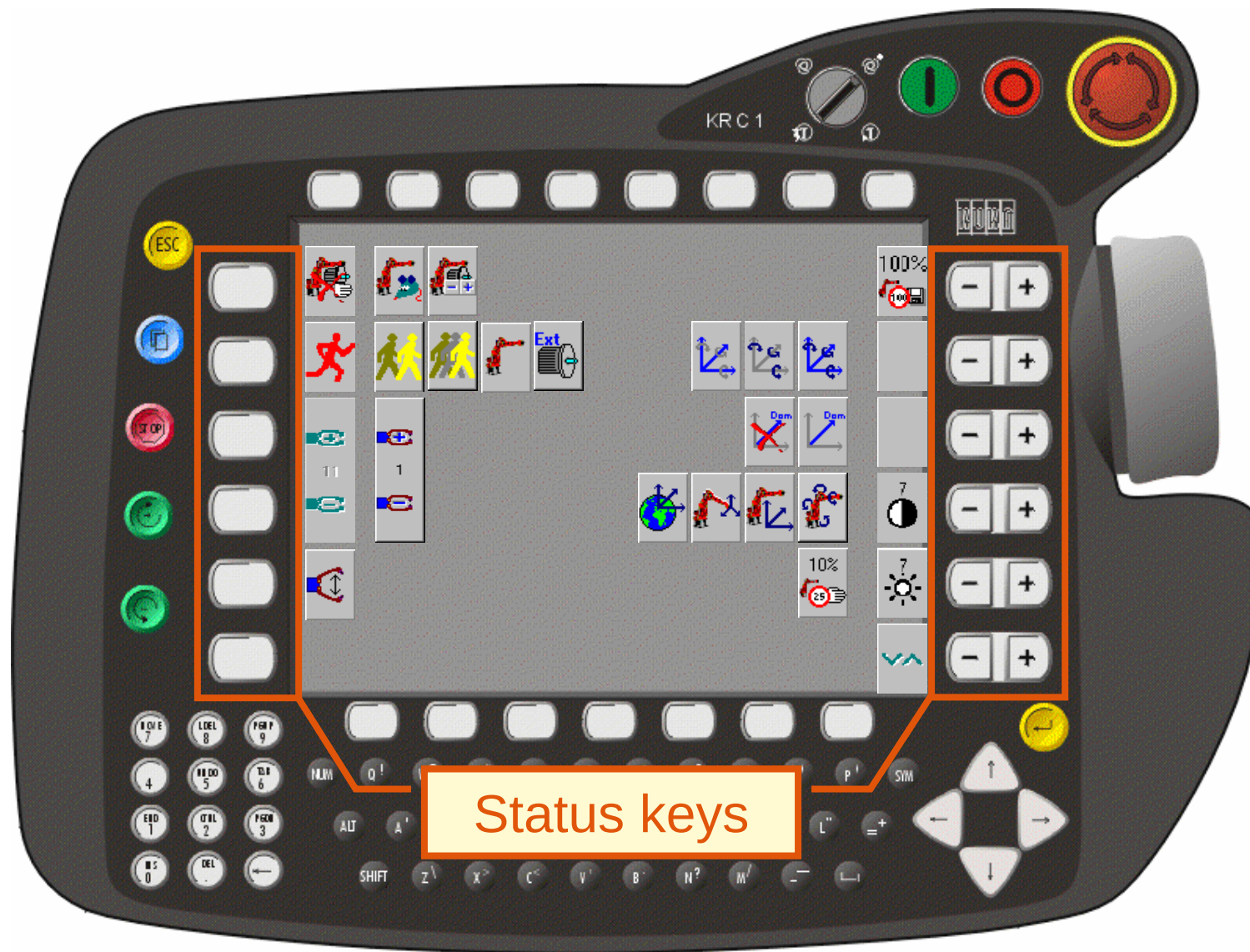
ASCII 键盘



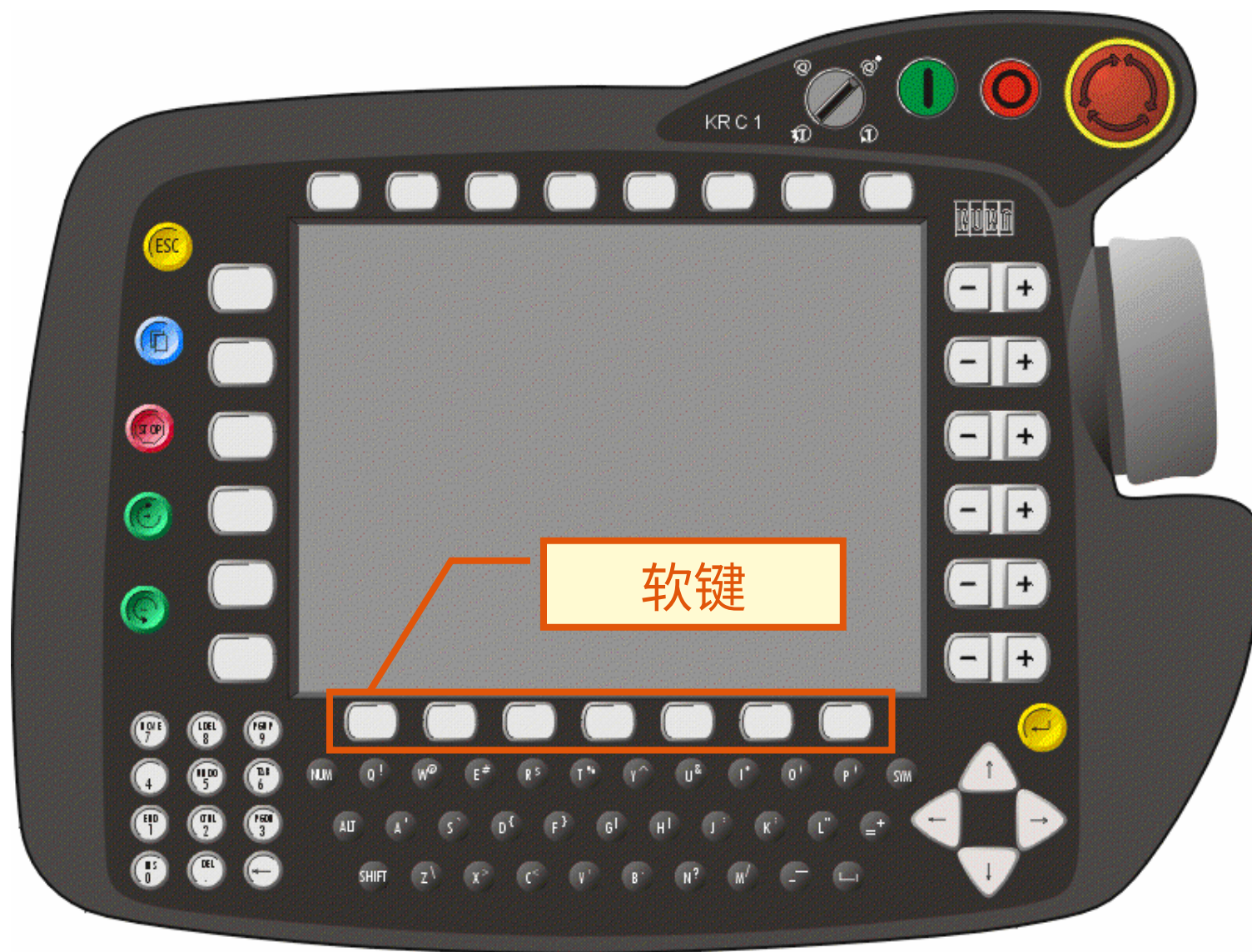




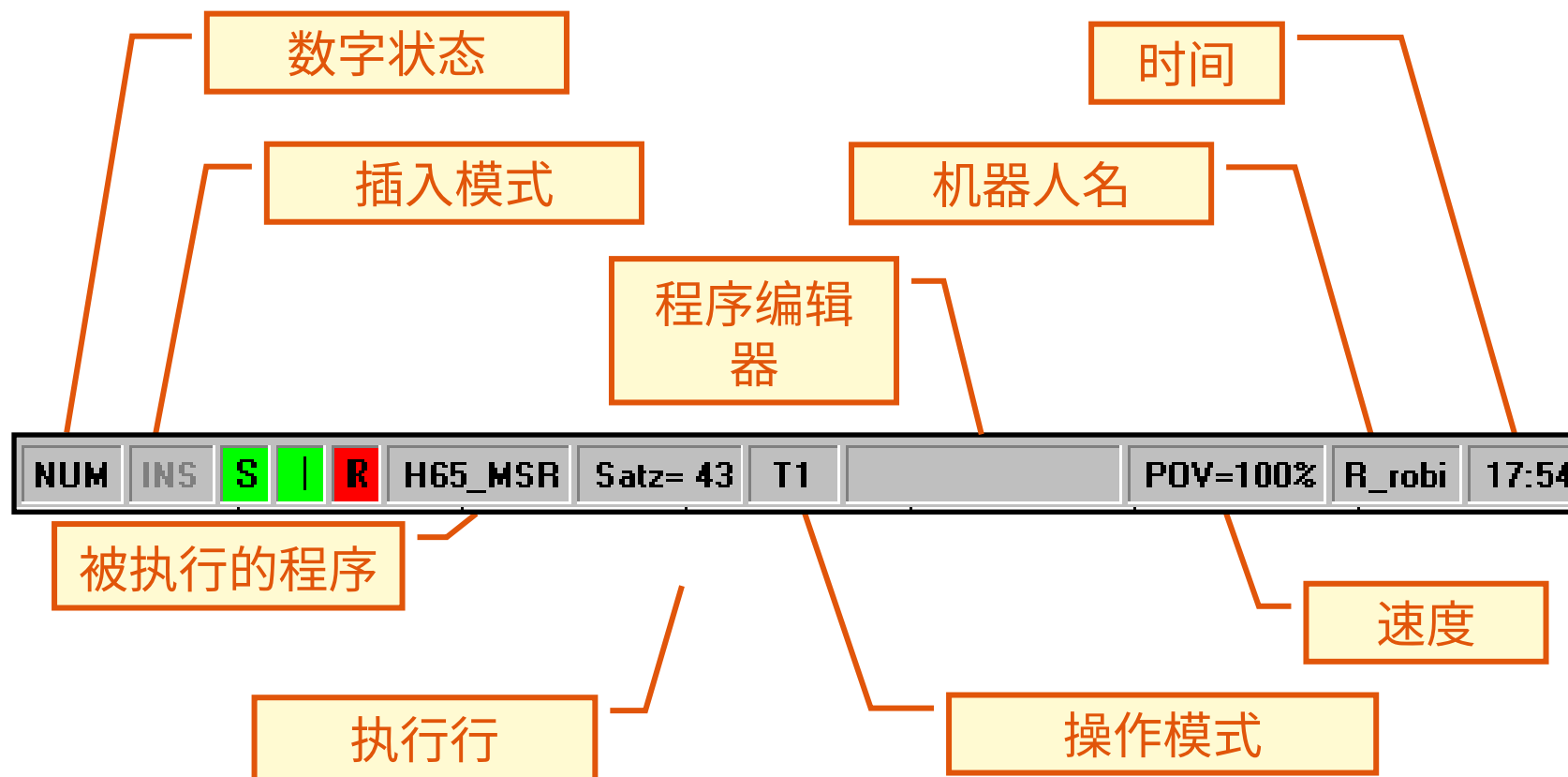
状态键



软键



状态行



状态行

NUM	INS	S	I	R	H65_MSR	Satz= 43	T1		POV=100%	R_robi	17:54
-----	-----	---	---	---	---------	----------	----	--	----------	--------	-------



INTERBUS停止



INTERBUS运行



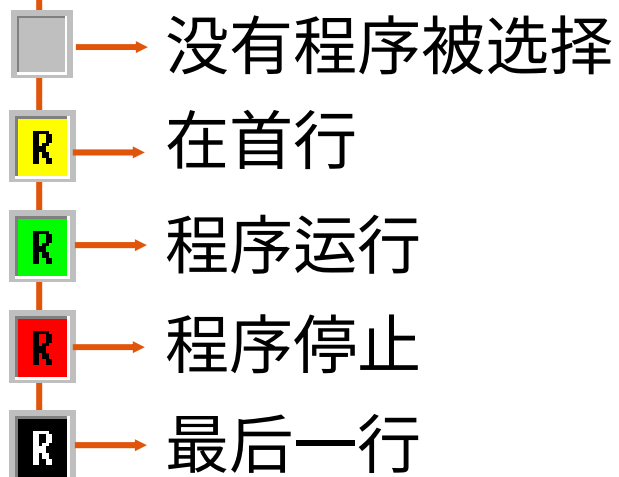
起动 OFF



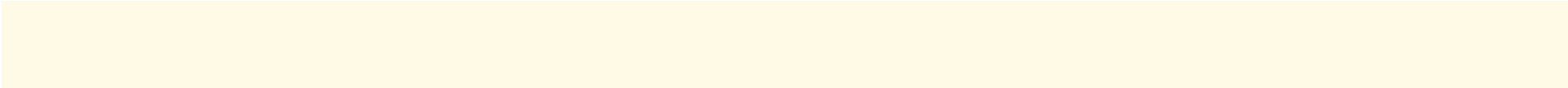
驱动 ON

Status line

NUM	INS	S	I	R	H65_MSR	Satz= 43	T1		POV=100%	R_robi	17:54
-----	-----	---	---	---	---------	----------	----	--	----------	--------	-------

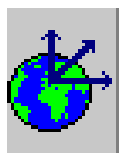


坐标系统





- **关节坐标**
每轴单独操作，可正反两个方向运行.



- **WORLD 坐标**
使用固定的，笛卡儿坐标，原点在机器人底座.



- **TOOL 坐标**
固定的，笛卡儿坐标，原点在机器人6轴发兰盘中心点.

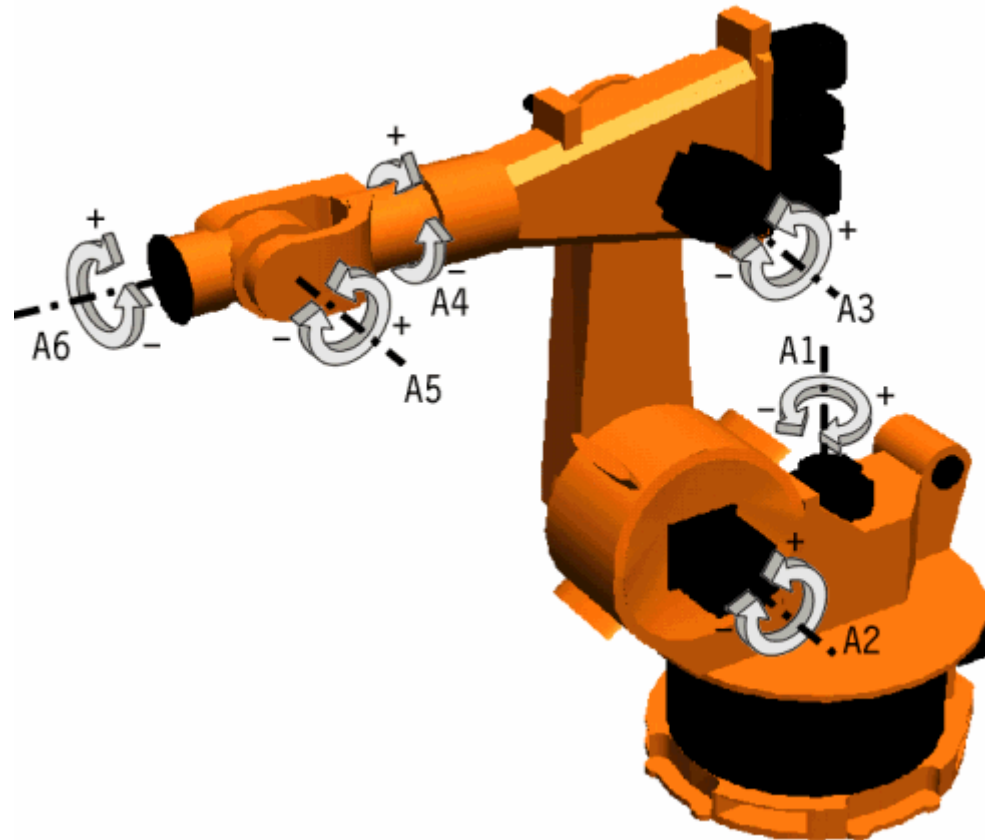


- **BASE 坐标**
固定的，笛卡儿坐标，原点在工件上.

关节坐标



每个轴都可正反两个方向操作.

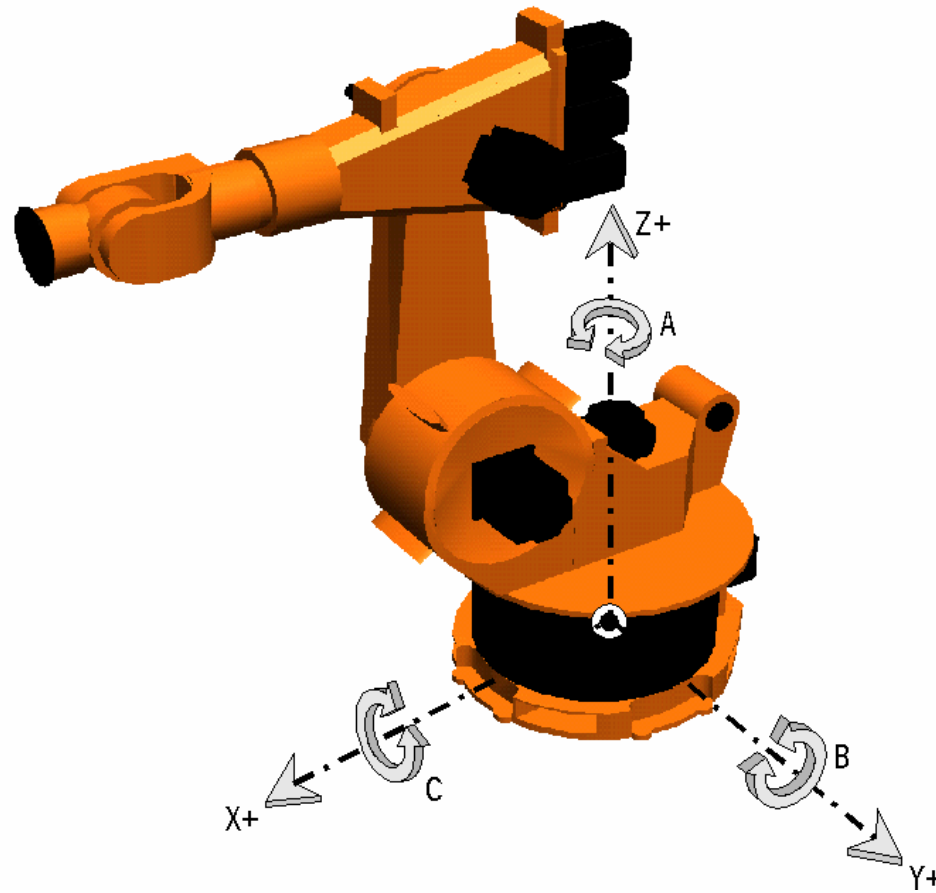


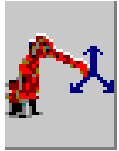
WORLD 坐标



- **WORLD 坐标**

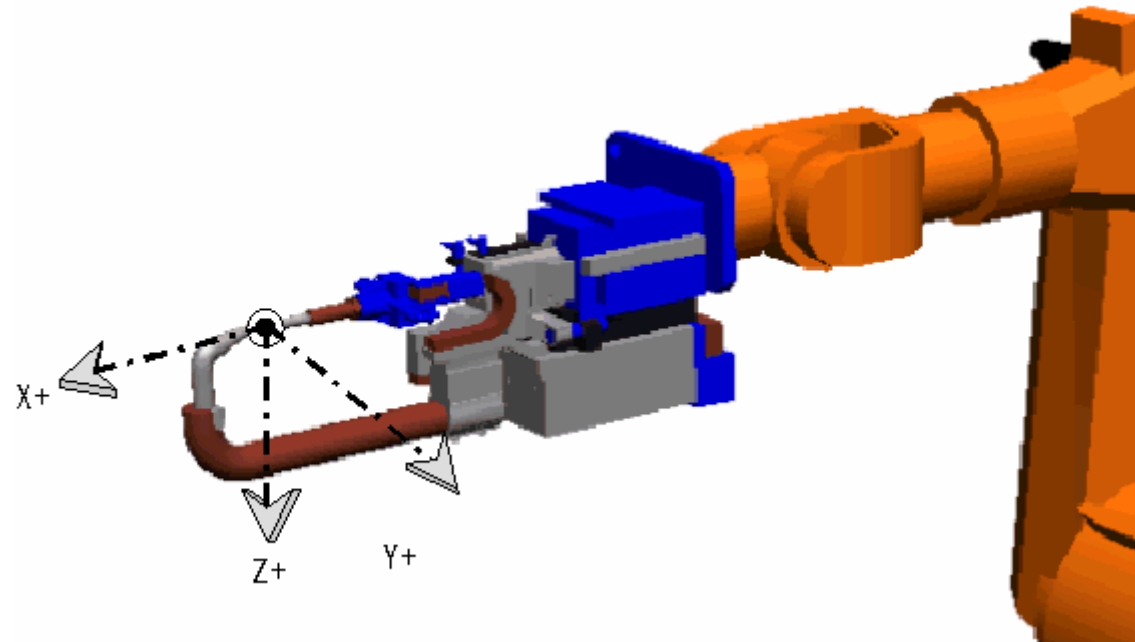
使用固定的，笛卡儿坐标，原点在机器人底座.





- **TOOL 坐标**

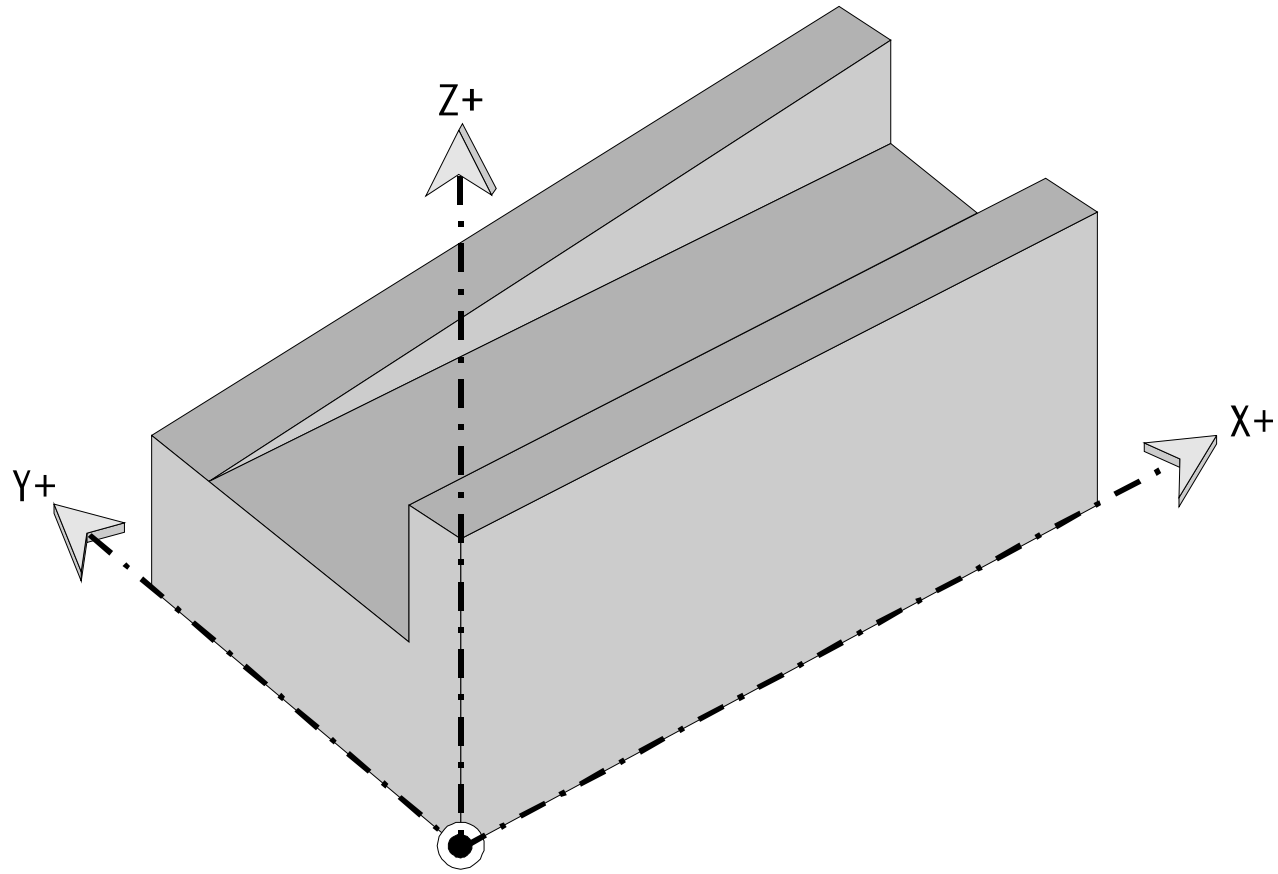
固定的，笛卡儿坐标，原点在机器人6轴发兰盘中心点.





- **BASE 坐标**

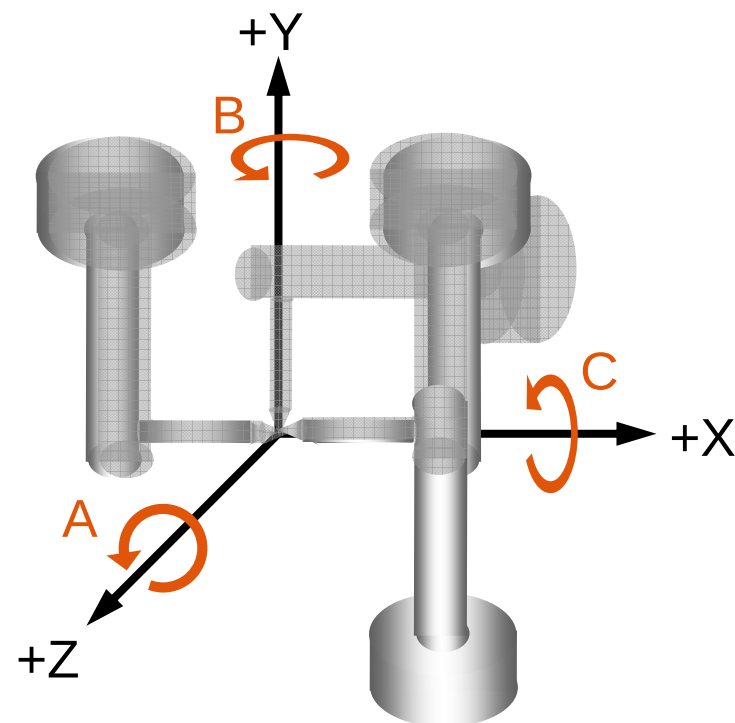
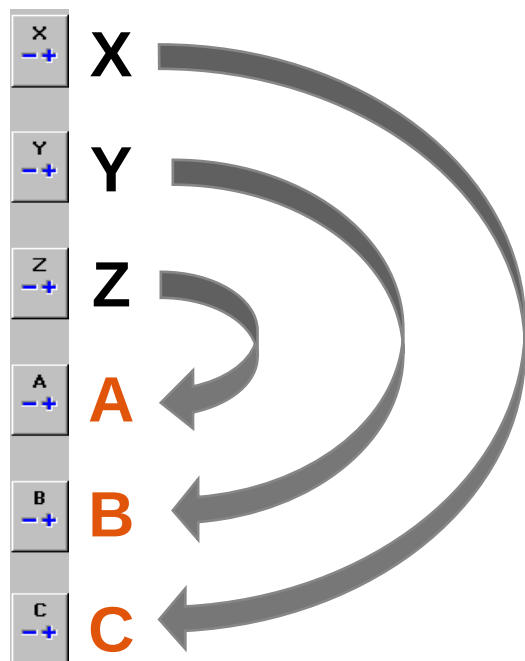
固定的，笛卡儿坐标，原点在工件上.



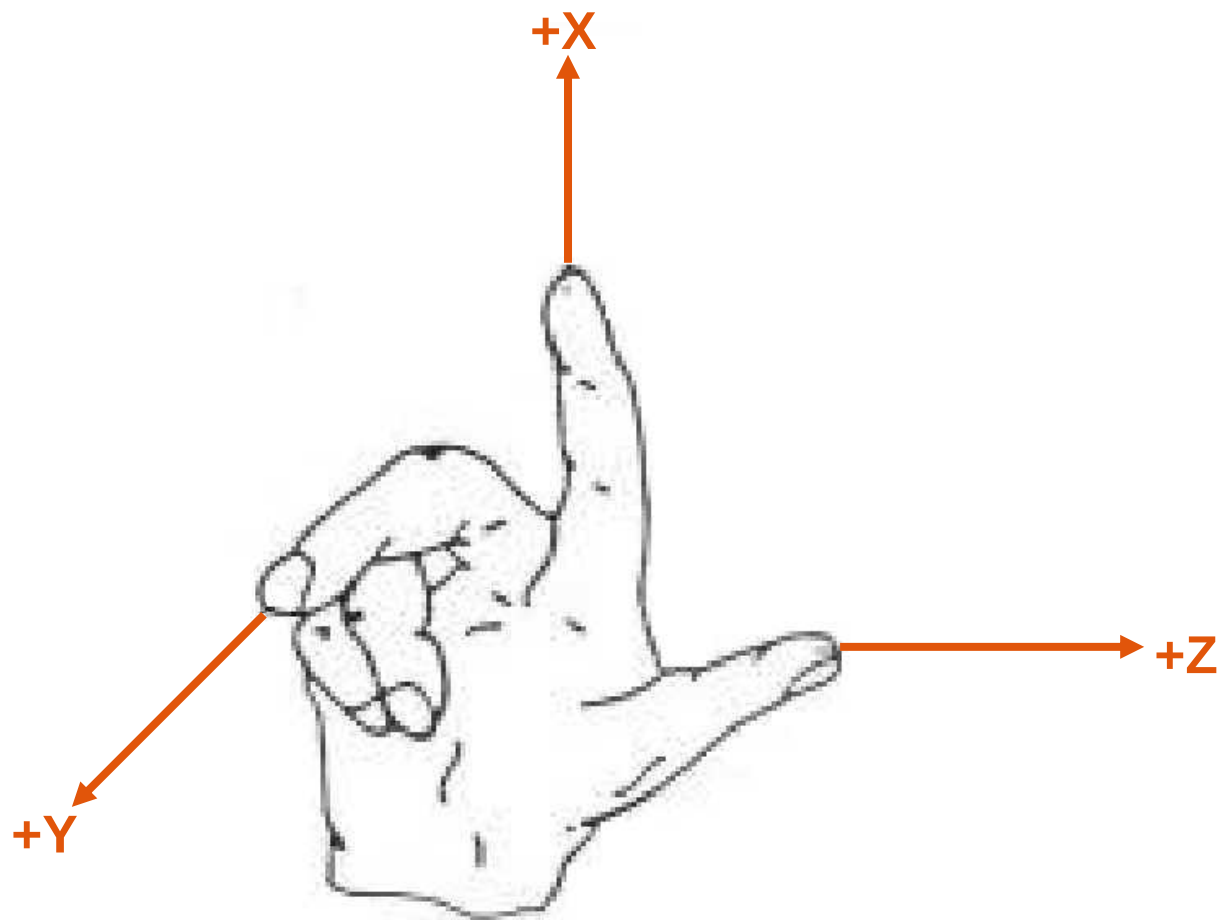
笛卡儿坐标的旋转

旋转 **A** →
旋转 **B** →
旋转 **C** →

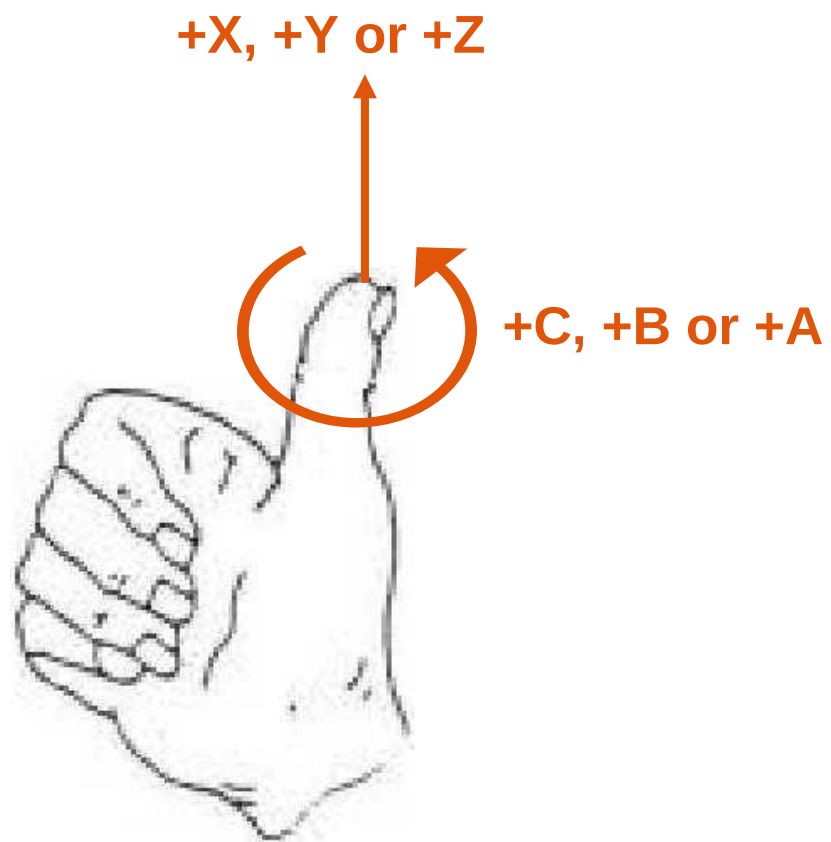
绕Z轴旋转
绕B轴旋转
绕C轴旋转



右手法



右手法



选择座标系统

- 旋转操作方式



鼠标操作/或软键操作



- 选择座标系统



关节坐标



WORLD 坐标



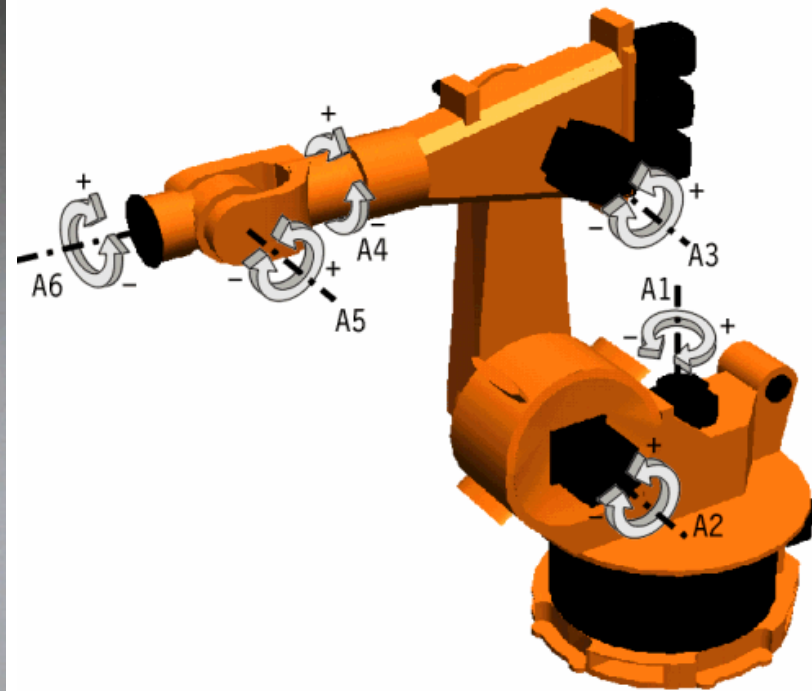
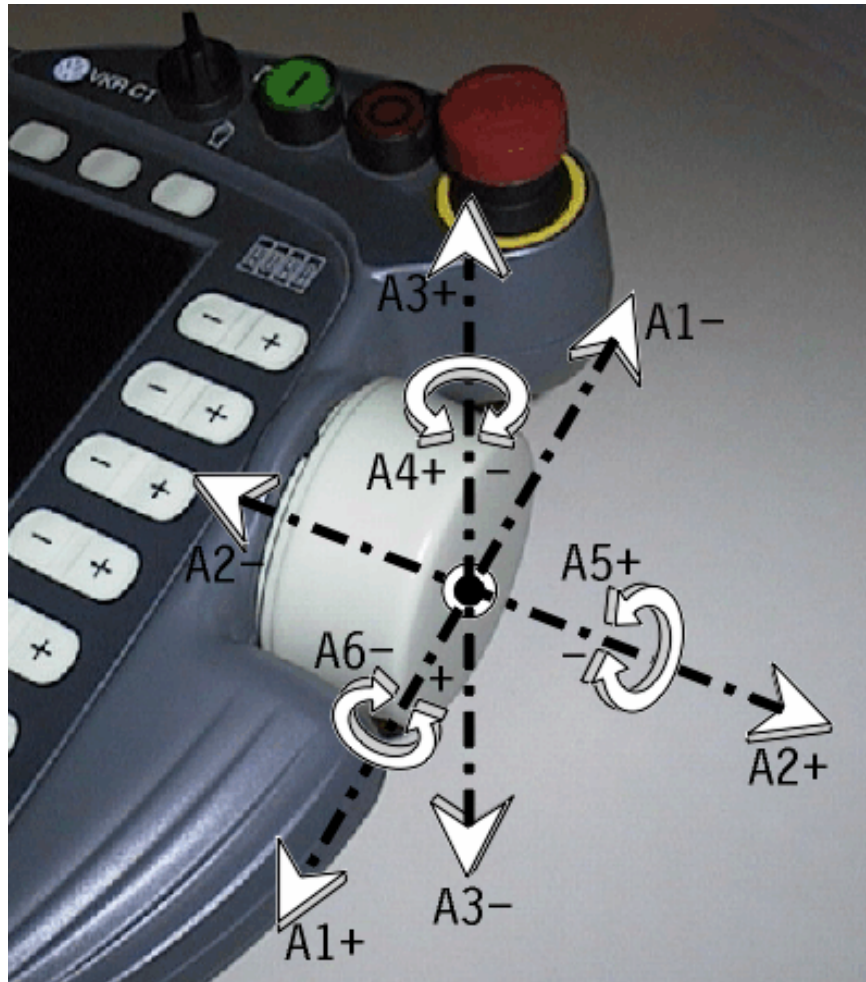
TOOL 坐标



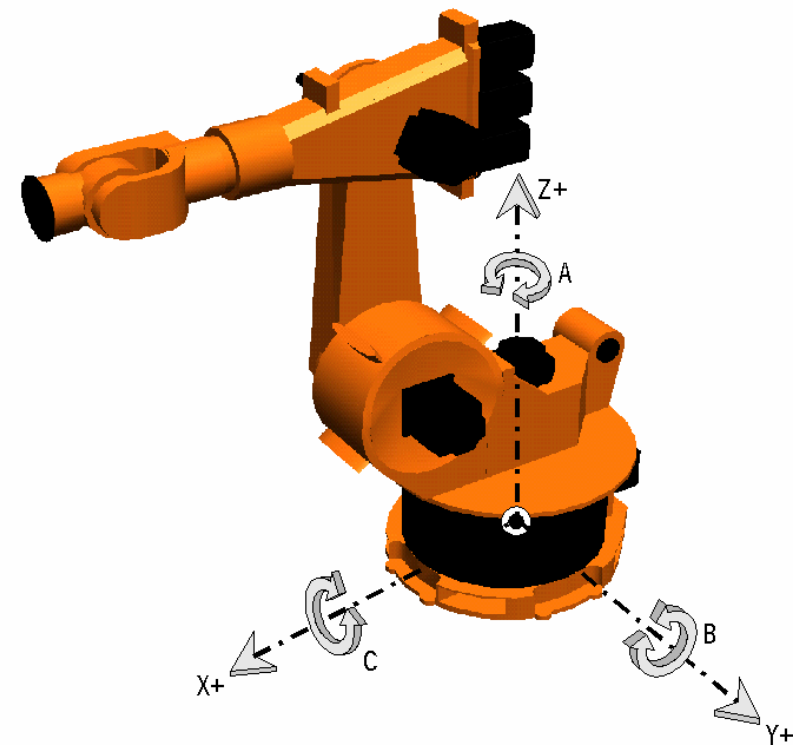
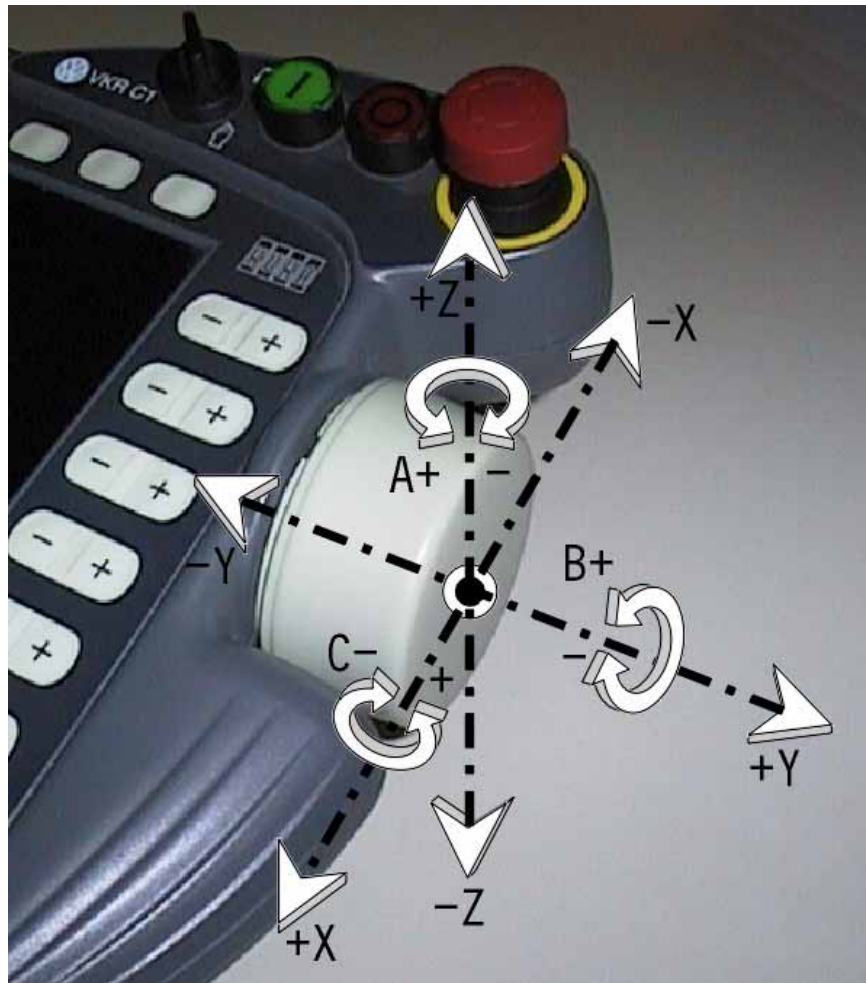
BASE 坐标



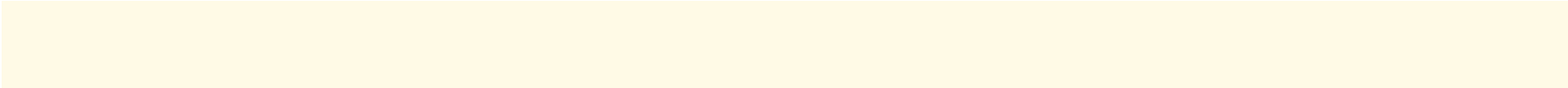
用鼠标操作关节坐标



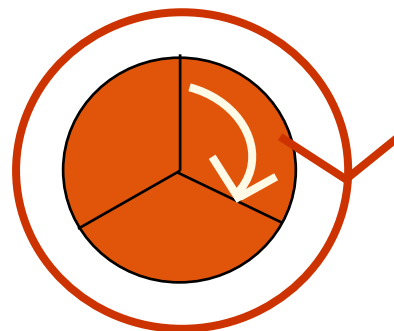
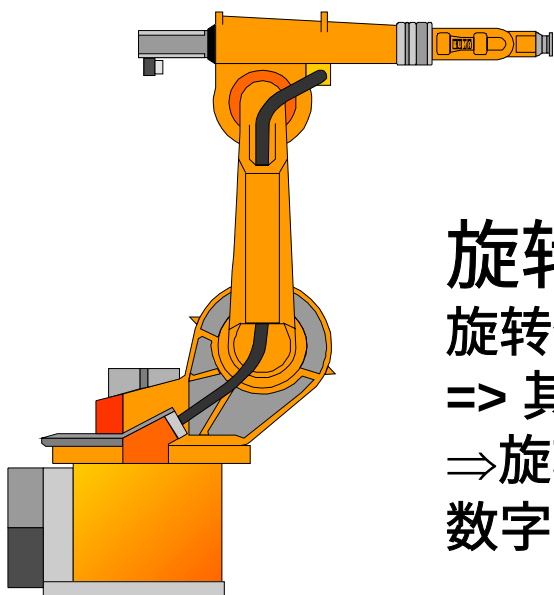
用鼠标操作笛卡儿坐标



零点校正



为什么要进行零点调整？



旋转位置编码器:

旋转位置编码器安装在电动机尾端.

=> 其机械角度是电角度的1/3.

⇒ 旋转位置编码器输出模拟量信号，在RDW中转换成数字量。

机器人在零点位置时，各轴视值如下:

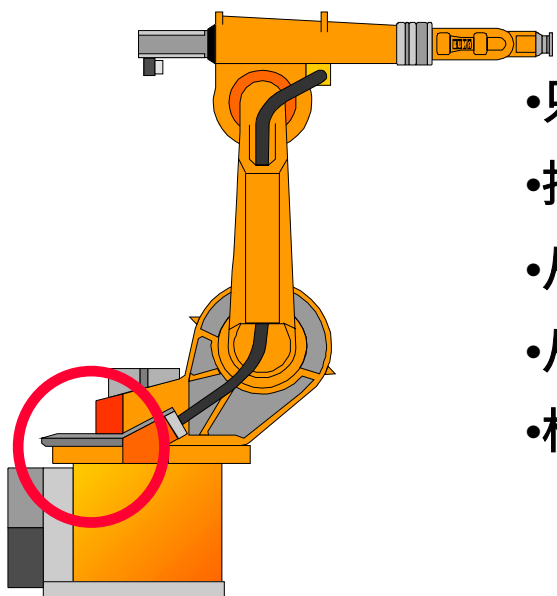
$$\begin{aligned} A1 &= 0^\circ, A2 = -90^\circ, A3 = 90^\circ \\ A4 &= 0^\circ, A5 = 0^\circ, A6 = 0^\circ \end{aligned}$$

零点

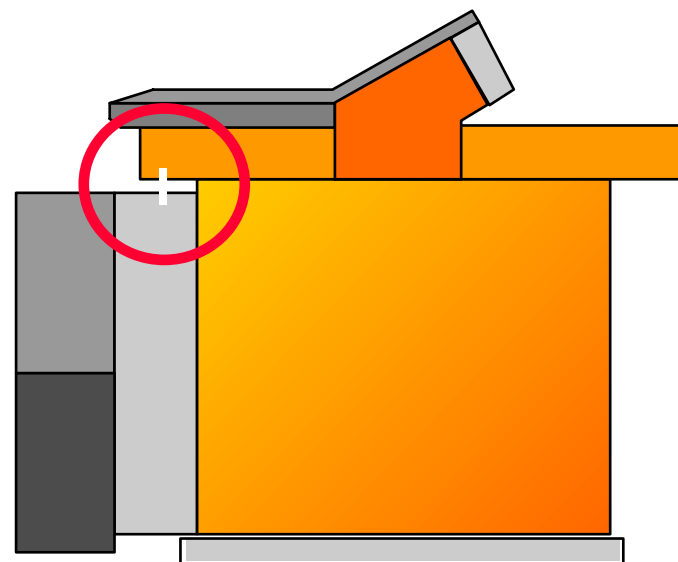
在下述情况发生时，需进行零点设定 ...	零点丢失原因...
... 维修后	...在引导期间自动丢失
... 没有连接控制器移动机器人的轴	...在引导期间自动丢失
...高于手动速度碰撞	...手动操作
...机器人与工件碰撞	...手动操作

重新零点设定..	零点丢失...
...若存储的零点丢失	...手动操作

零点校正程序



- 只能在T1方式设定零点
- 按顺序分别设定每一个轴
- 从1轴开始
- 从正向负方向逼近预校正点
- 标尺处是预校正位置



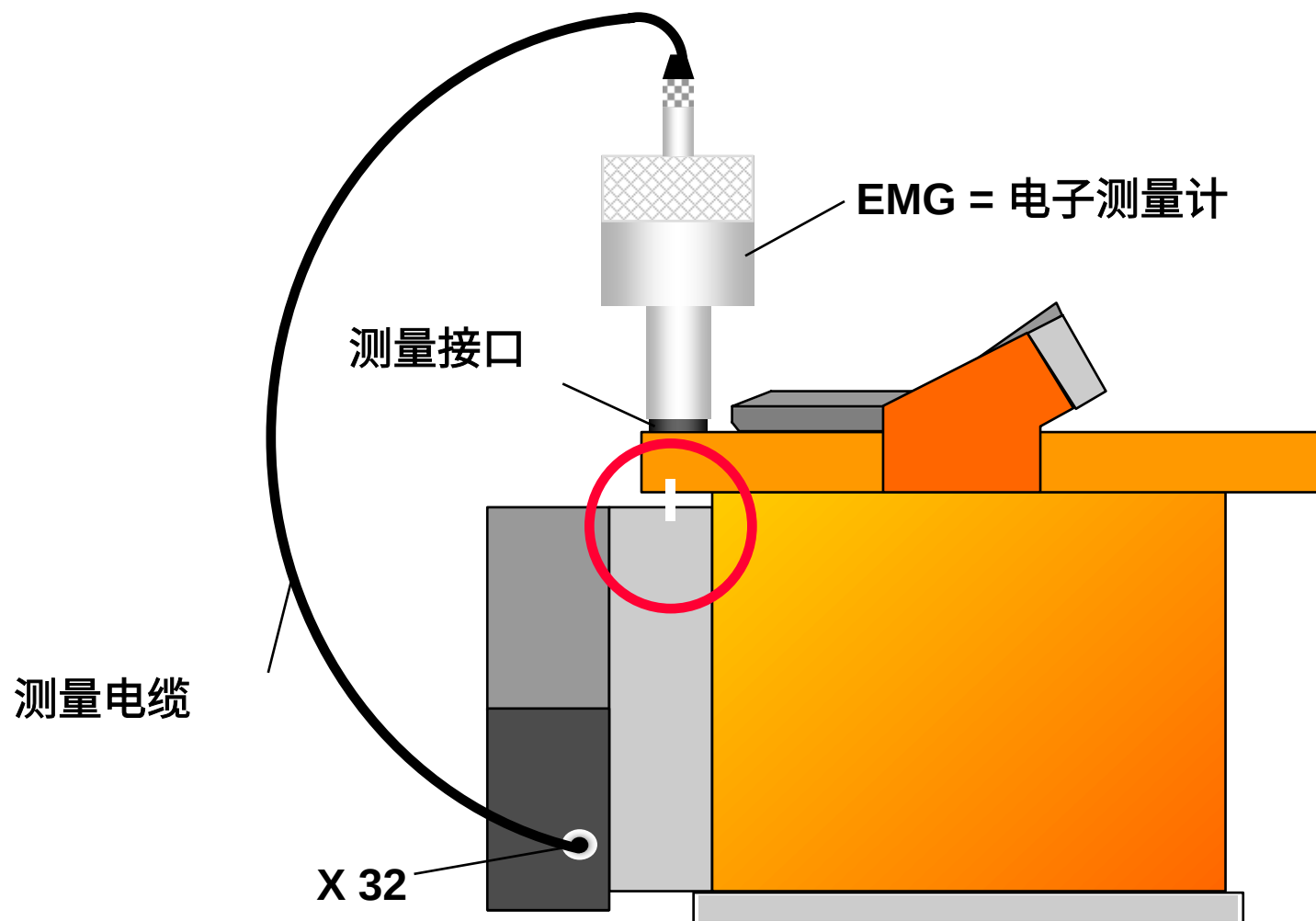
千分尺

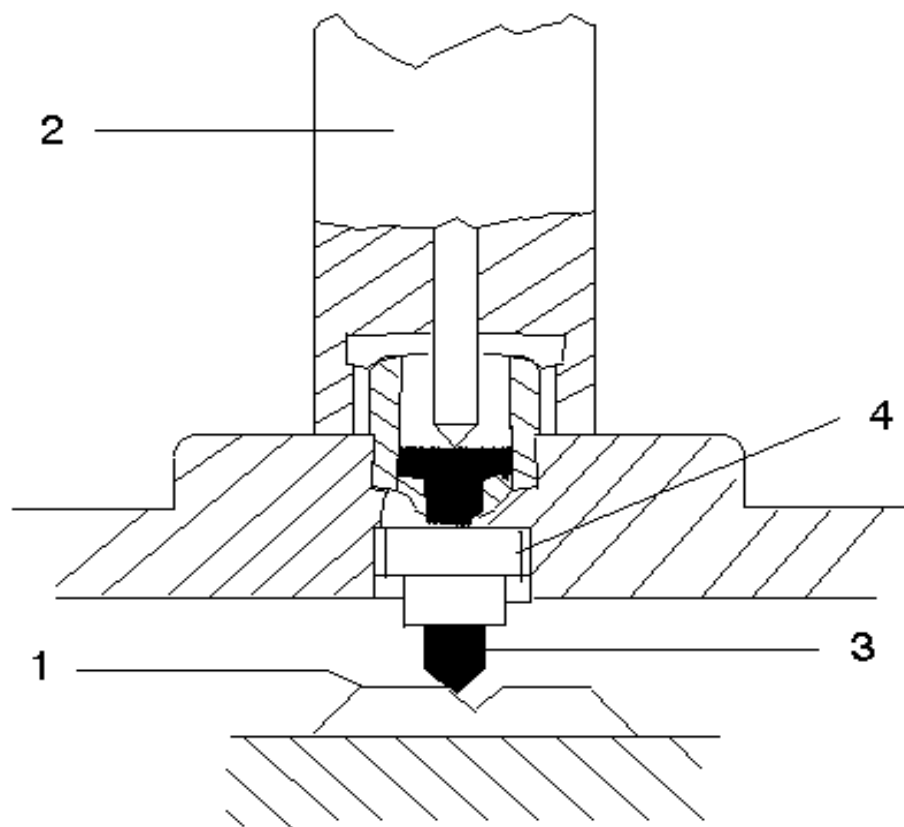


EMG/EMT



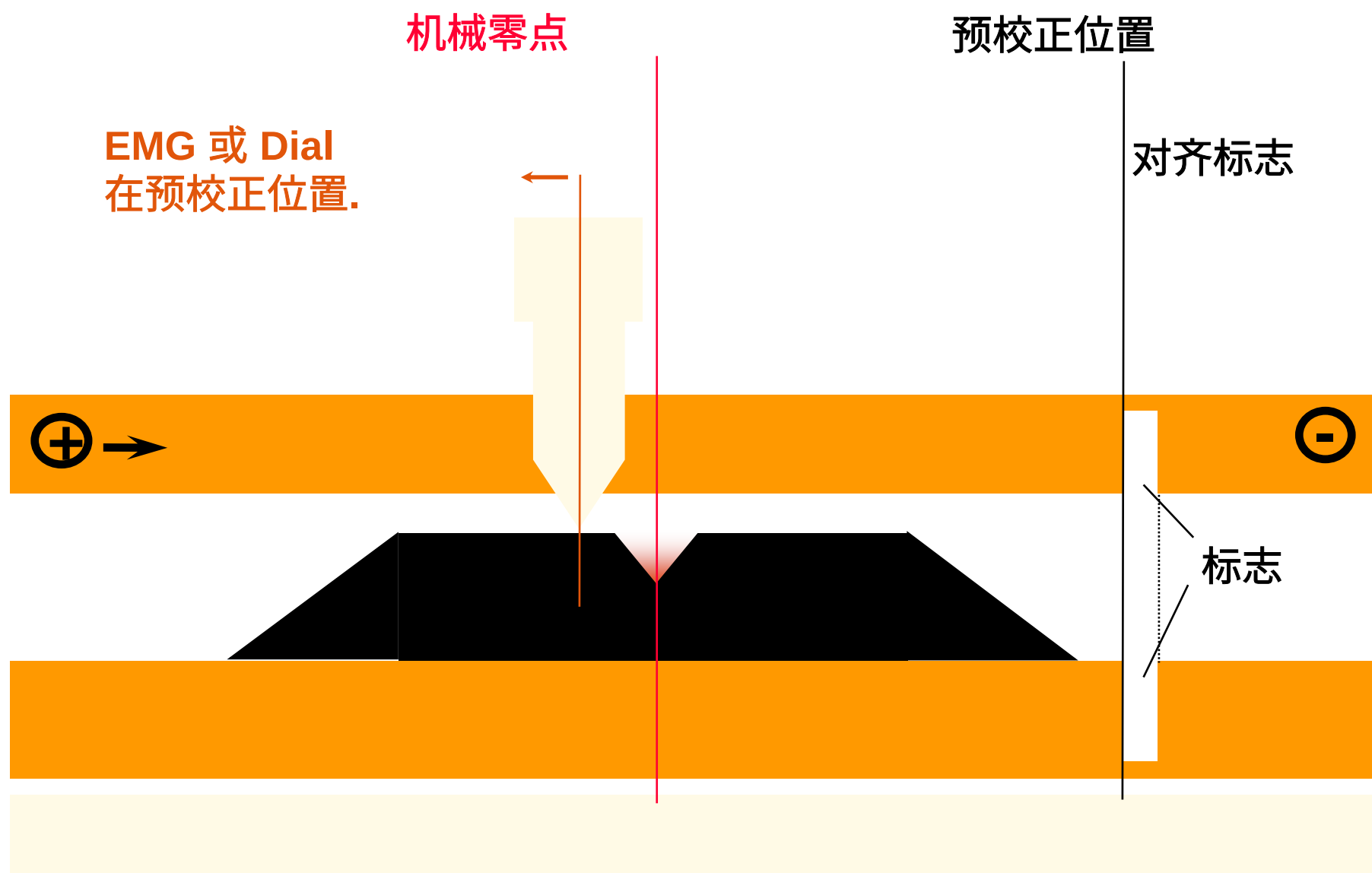
EMT 连接





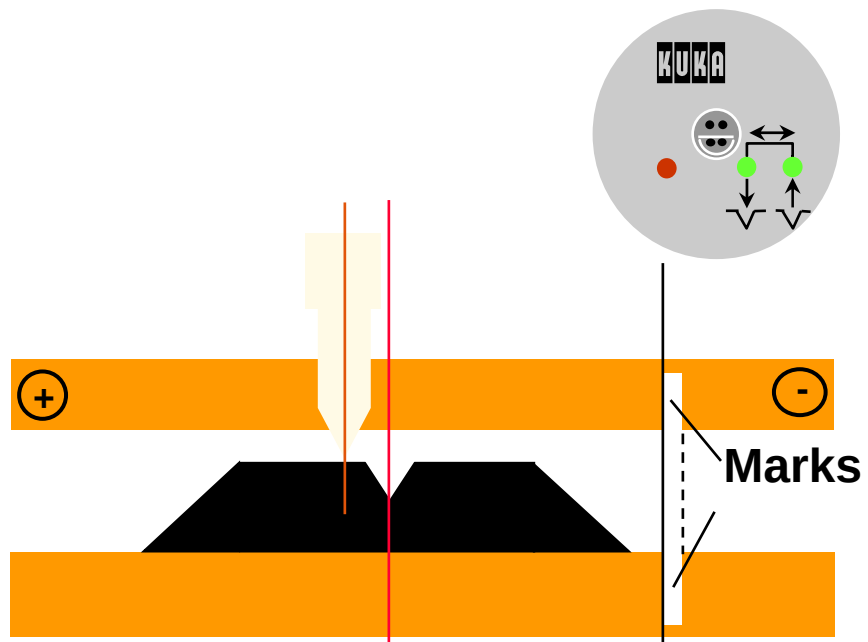
- 1 凹槽
- 2 零点测量计
- 3 顶针
- 4 顶针座

Measure pin with notch



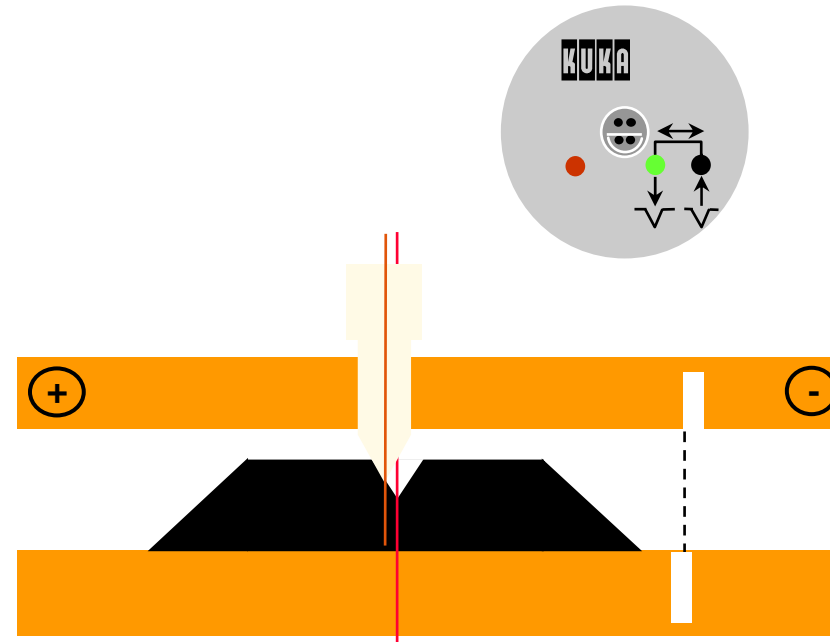
1

- 预校正位置
- 红灯灭
- 两个绿灯亮
- 对齐标志



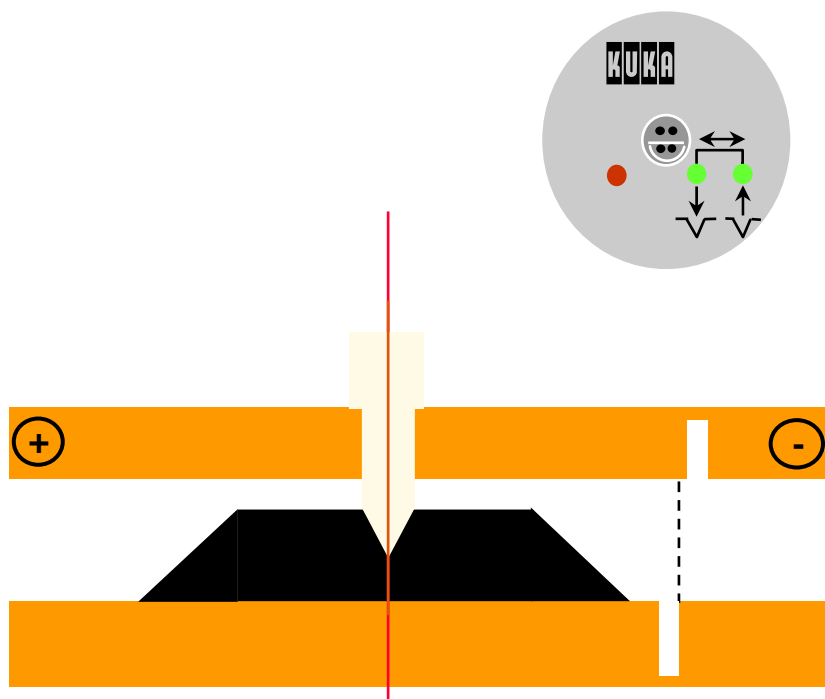
2

- EMG 在凹槽下滑
- 相应绿灯亮



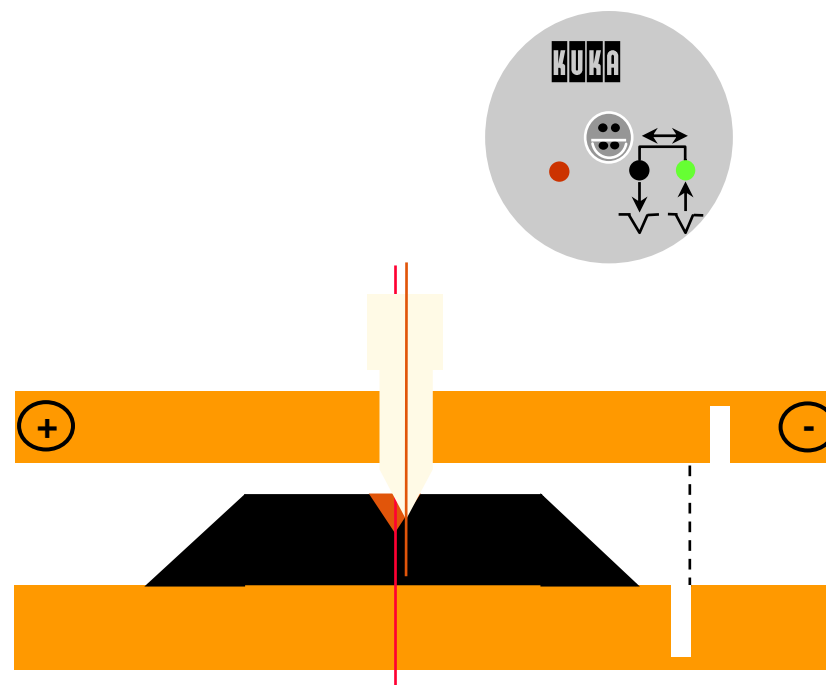
3

- 在零点位置
- 位置校验
- 绿灯都亮

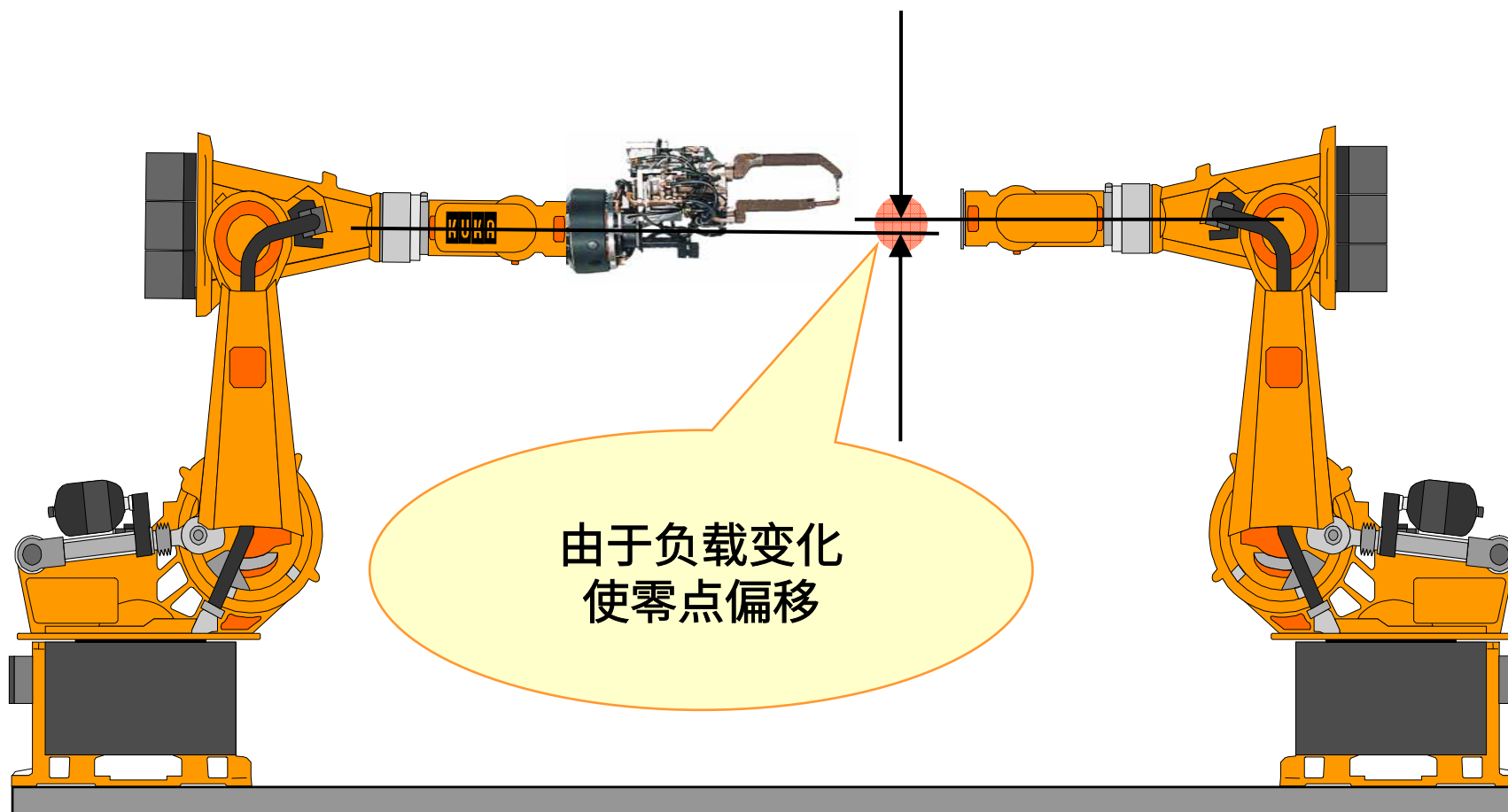


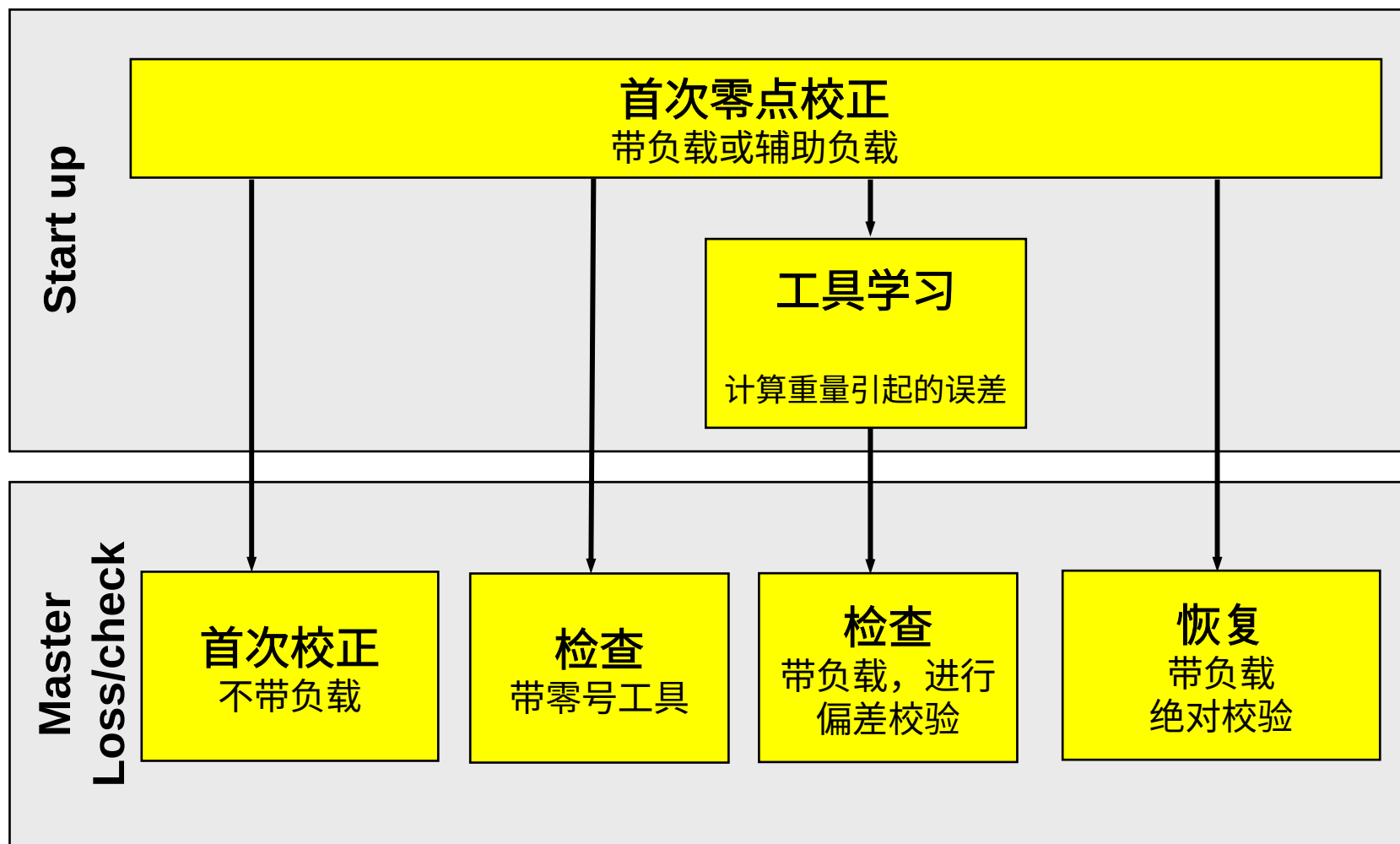
4

- 顶针向上
- 相应绿灯亮
- 红灯不亮

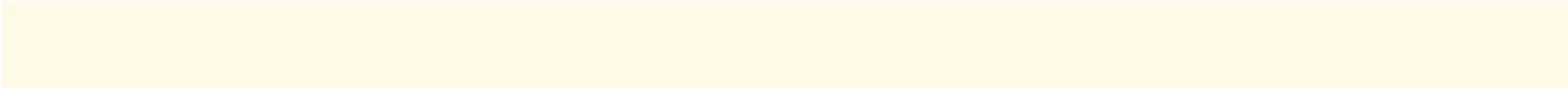


工具负载的影响





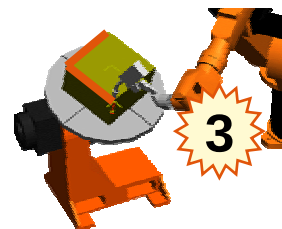
• 工具測量





用户可以进行工具测量.
工具原点由用户决定.

为什么要进行工具测量？



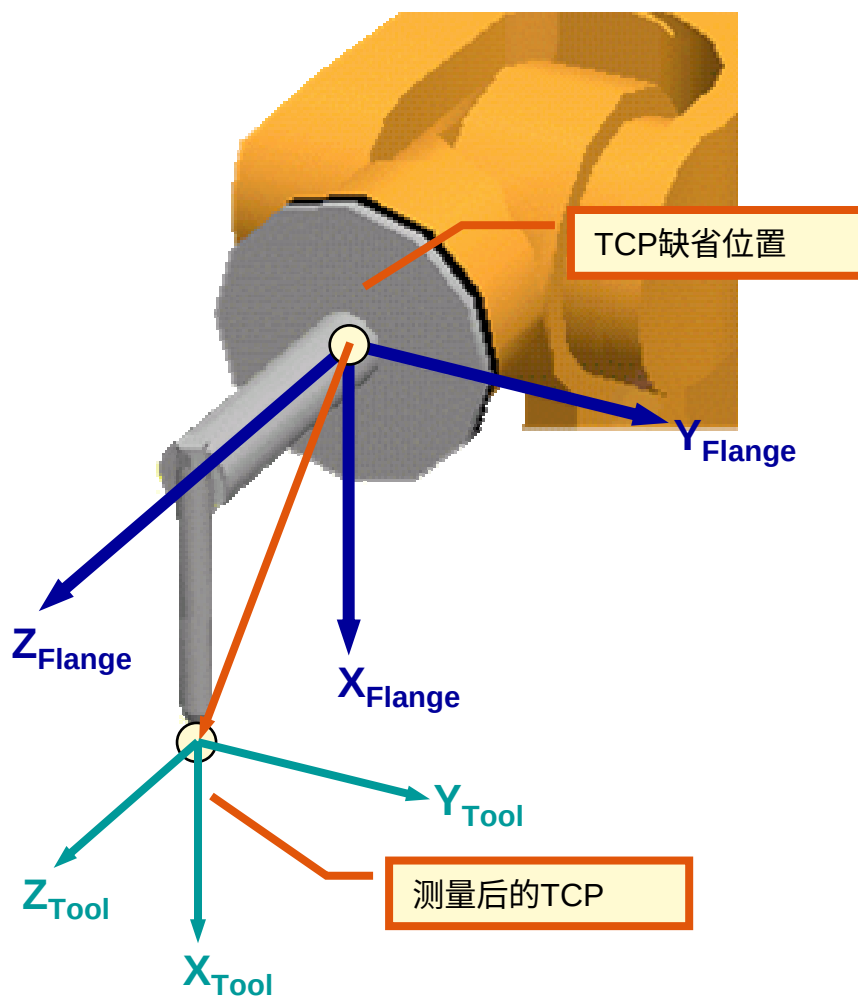
工具测量步骤

1.Step:

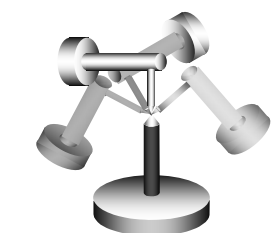
确定TCP中心点

2. Step:

确定工具的方向

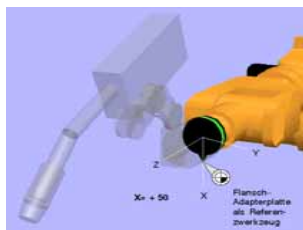


1. 确定TCP点



XYZ-4 点

or



XYZ-参考

2. 确定旋转方向



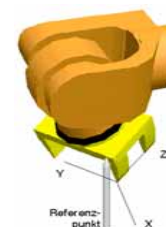
ABC-World 5D

or



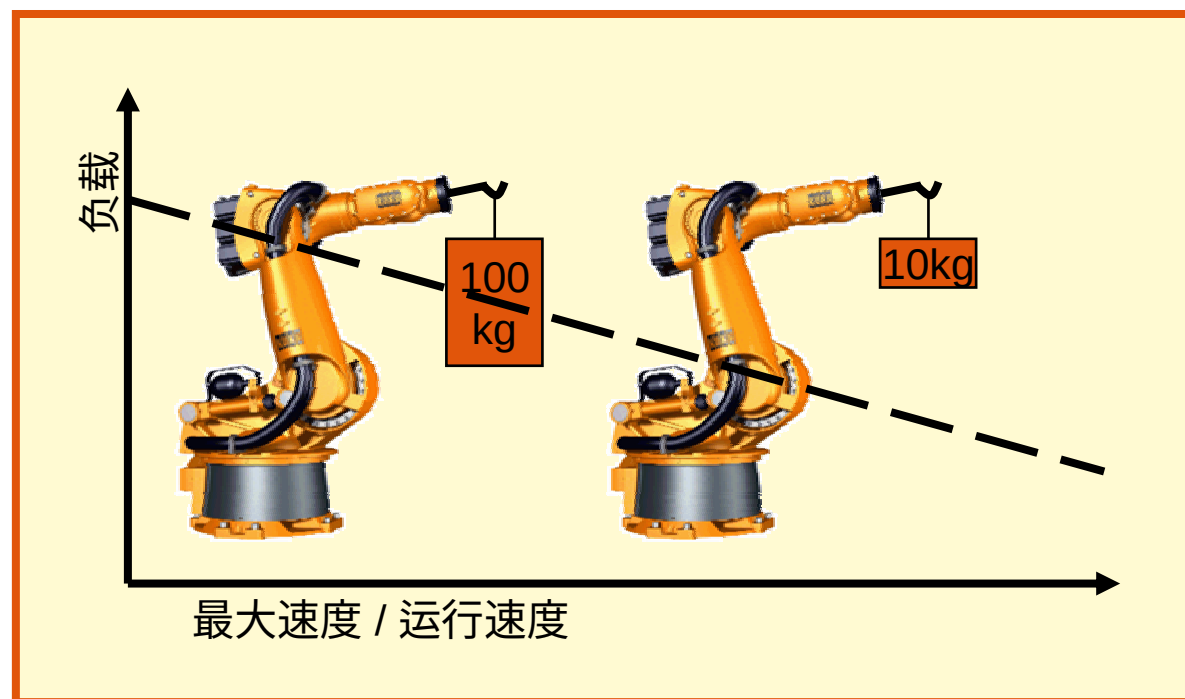
ABC-World 6D

or



ABC-2 Point

为了最快地运行机器人，要进行
负载数据优化

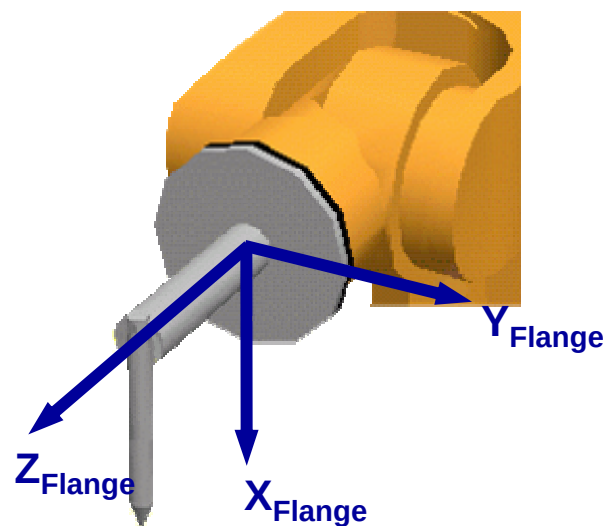


不要忘了外部负载!

payload data

M 负载重量.

X, Y, Z 负载中心至法兰盘中心点的距离

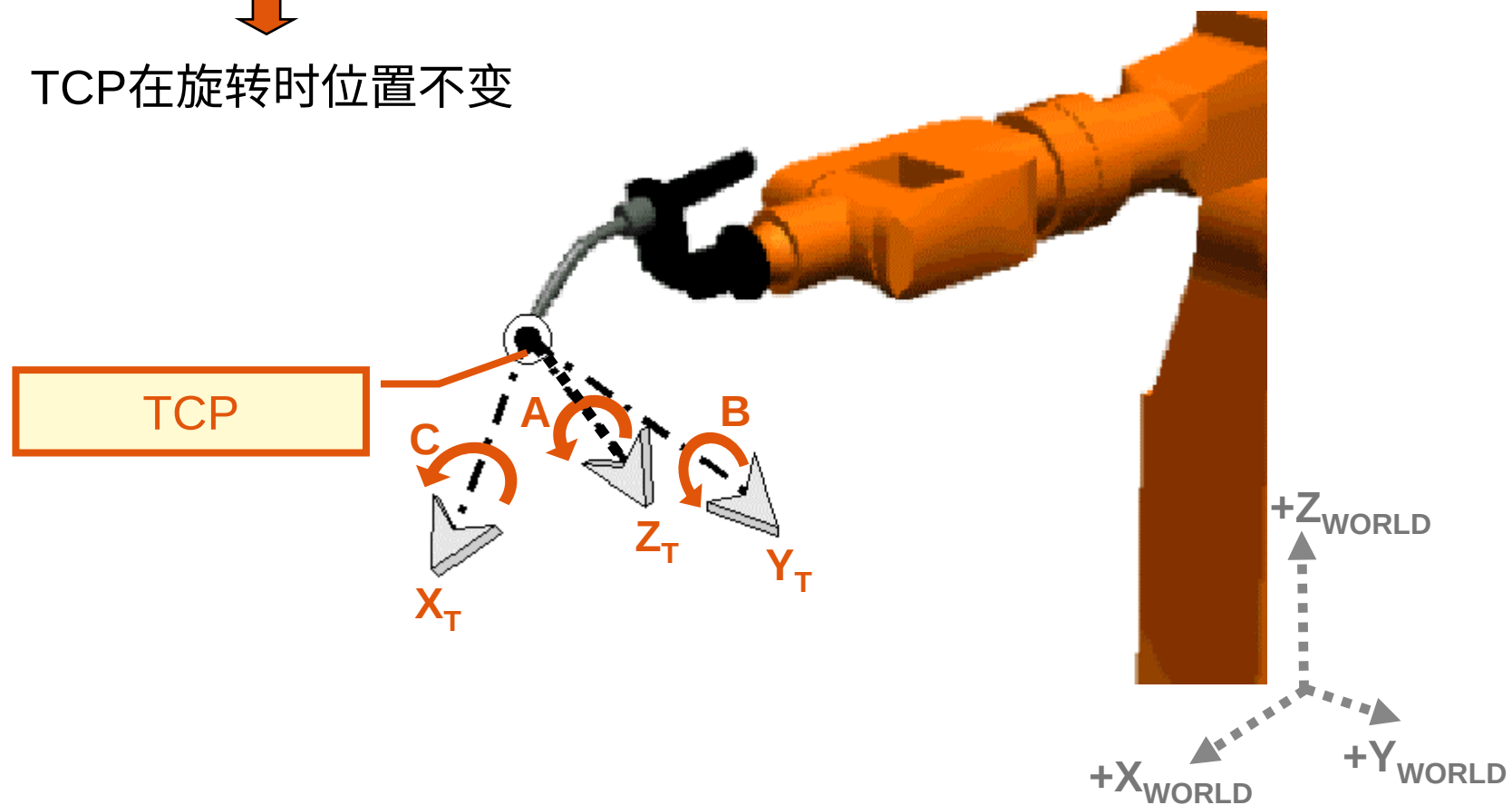


TCP点在旋转时的...

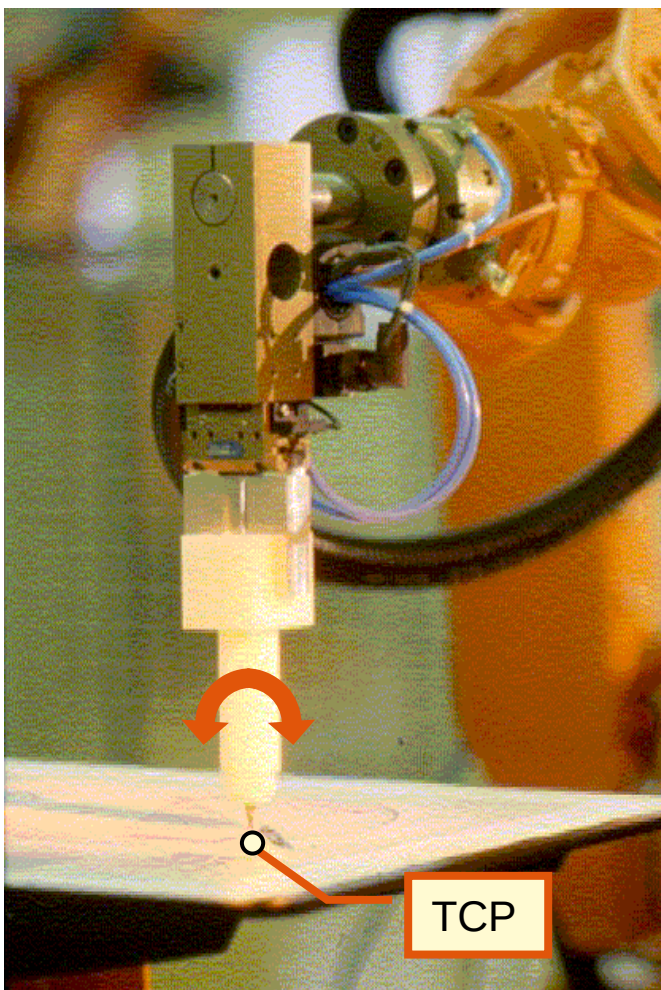
操作



TCP在旋转时位置不变



TCP示教 例



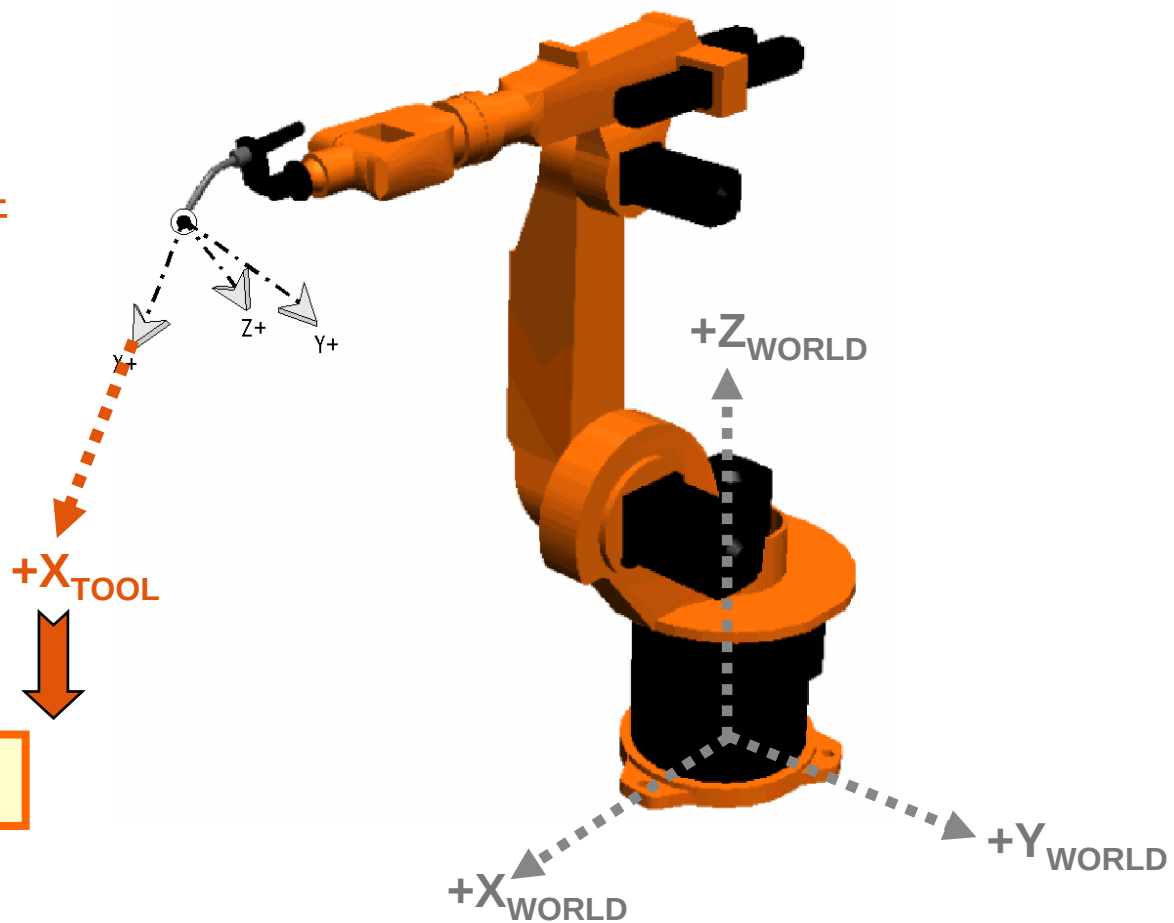
- 无论在哪个点，旋转时TCP不变.

TCP示教例

示教

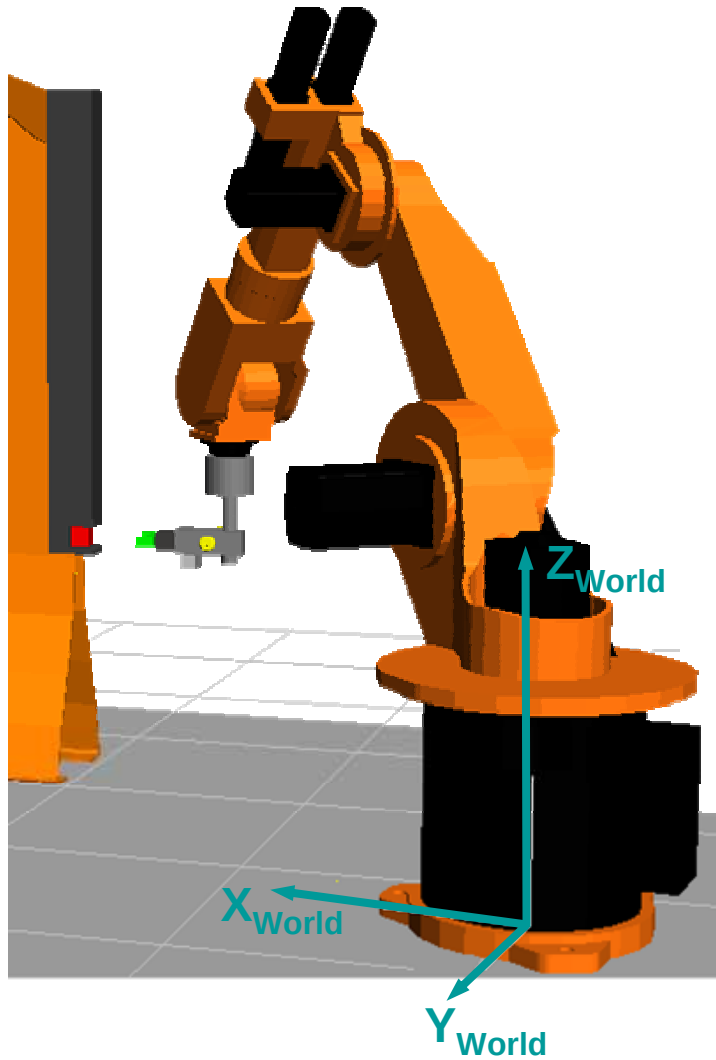


向工件方向直接运行



工件方向

Example of jogging in tool working direction



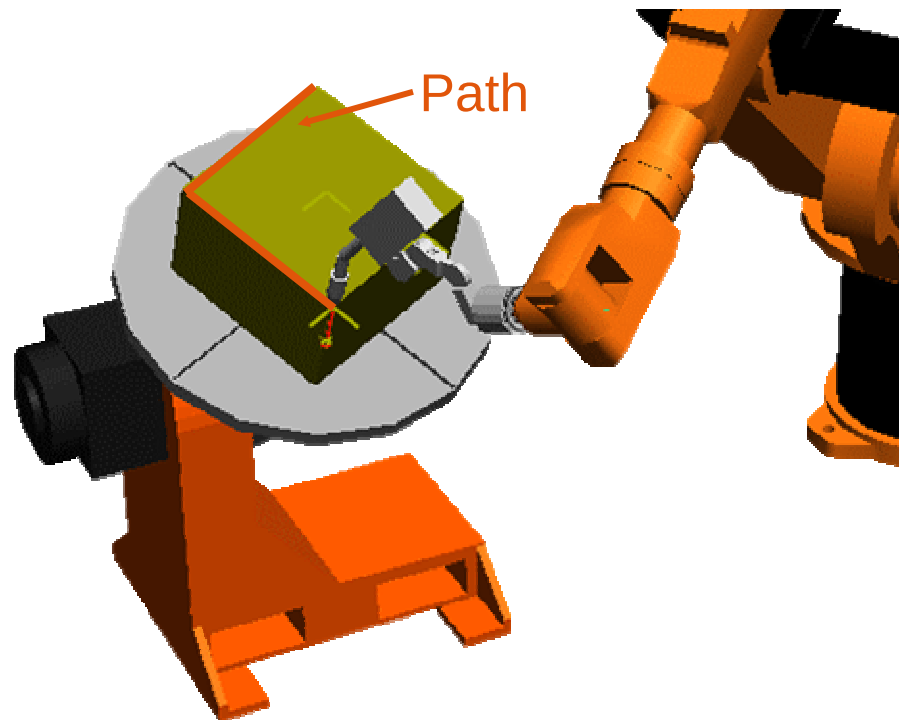
- Magazine is not orientated in WORLD coordinate system.
- If you want to jog to pick position in WORLD system, it's necessary to use different direction.
- Using a measured tool, you can jog in tool working direction to pick a cube.

Sense of tool measuring

Program execution

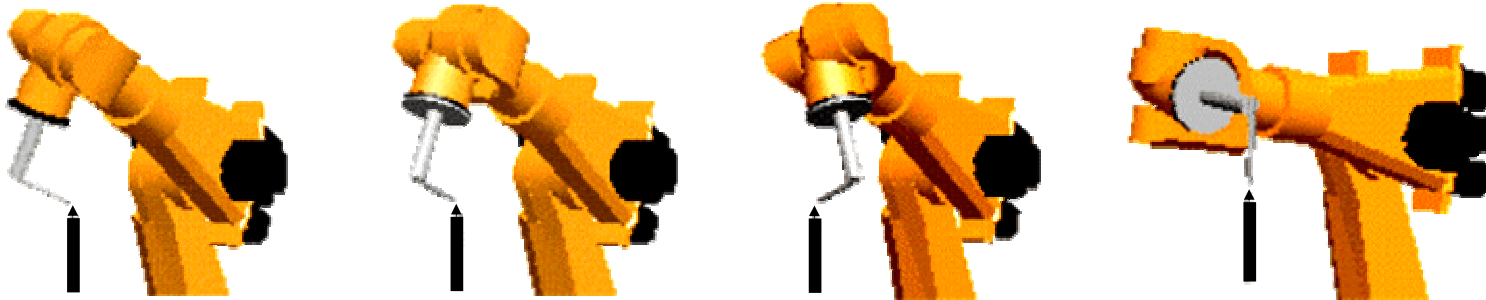


The programmed velocity
is kept during
the programmed path.



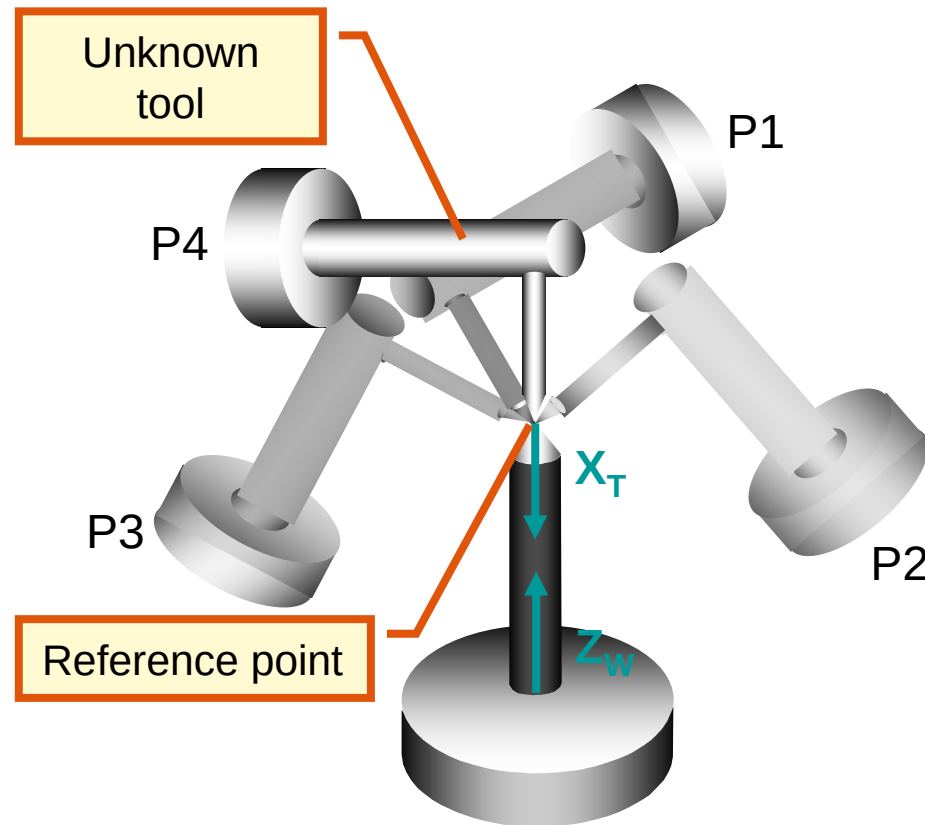
XYZ 4 point method

By means of XYZ 4 point method the TOOL CENTER POINT is jogged to a reference position out of 4 different directions.



The TOOL CENTER POINT will be calculated by the different positions of the flange and the robot axes.

4 point method



- Jog to reference point out of 4 different directions. (P1 to P4)
- **Advice** : Choose the last orientation (P4) , so that $+X_t$ has the same direction as $-Z_w$
- **Important**: The different Orientations of the tool must differ a minimum amount.



Reduce the HOV (hand override) near to the reference point in order to avoid a collision.

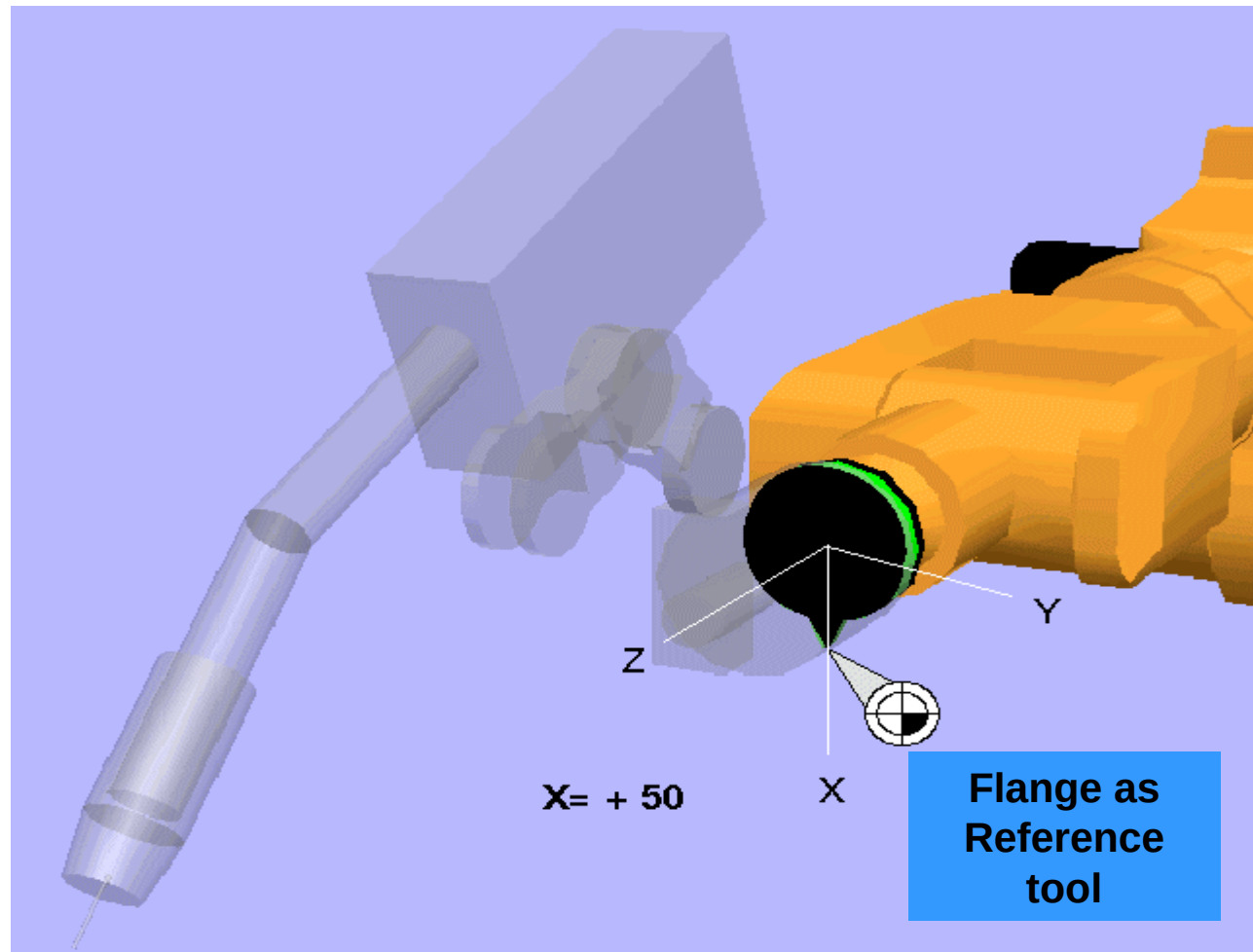
XYZ reference method

By means XYZ reference method the data of the TOOL CENTER POINT is calculated by comparison with a already known tool.



The new TOOL CENTER POINT will be calculated by the different flange and axis positions.

Example of XYZ Reference method



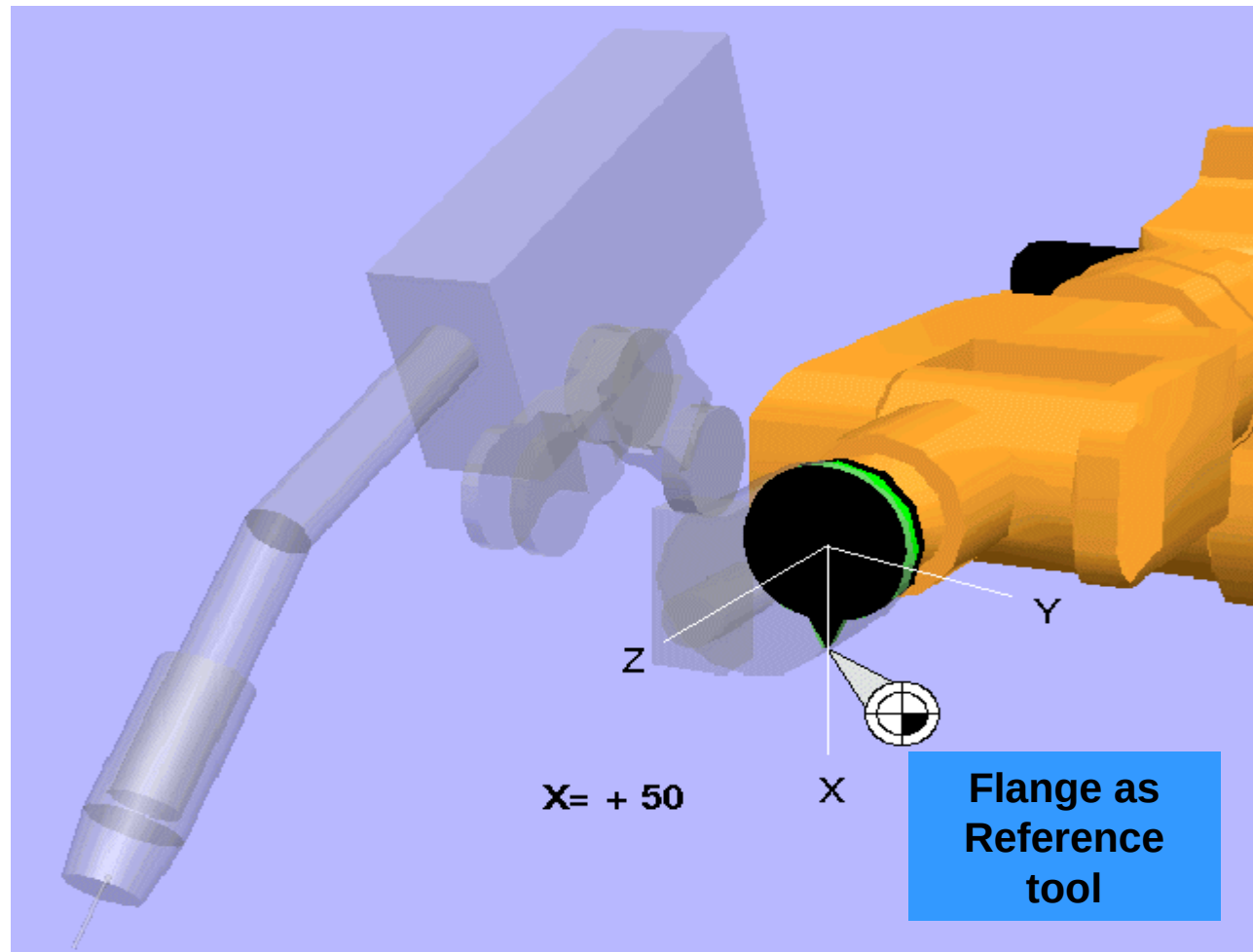
XYZ reference method

By means XYZ reference method the data of the TOOL CENTER POINT is calculated by comparison with a already known tool.

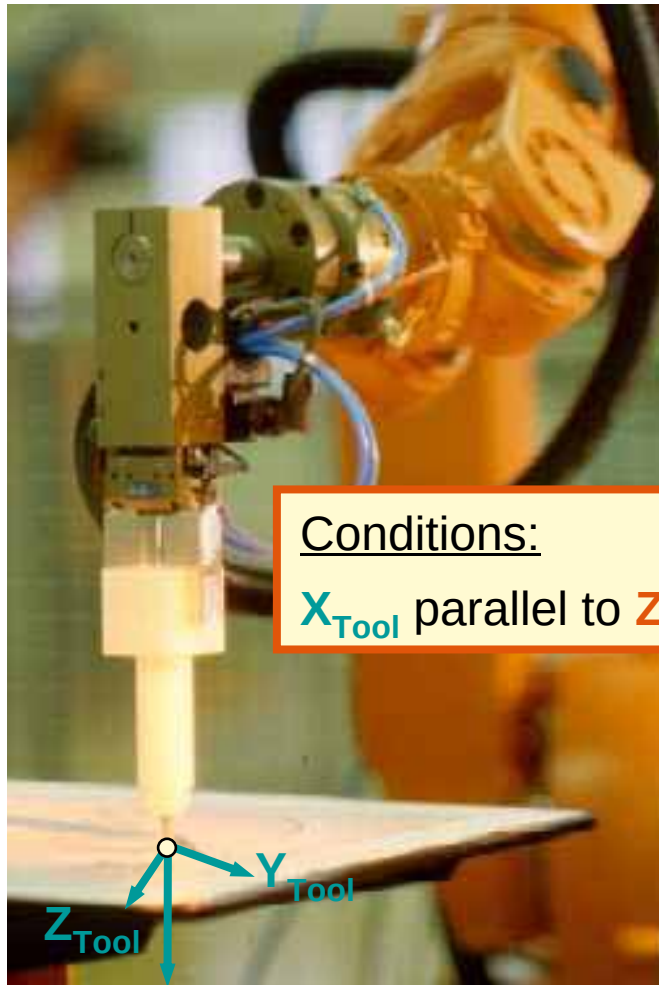


The new TOOL CENTER POINT will be calculated by the different flange and axis positions.

Example of XYZ Reference method

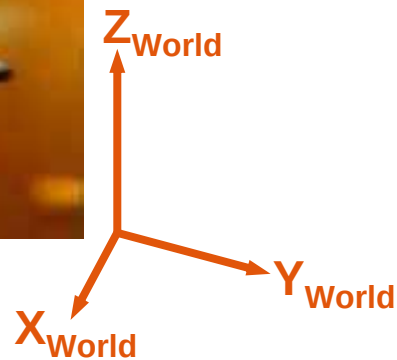


ABC world 5D method



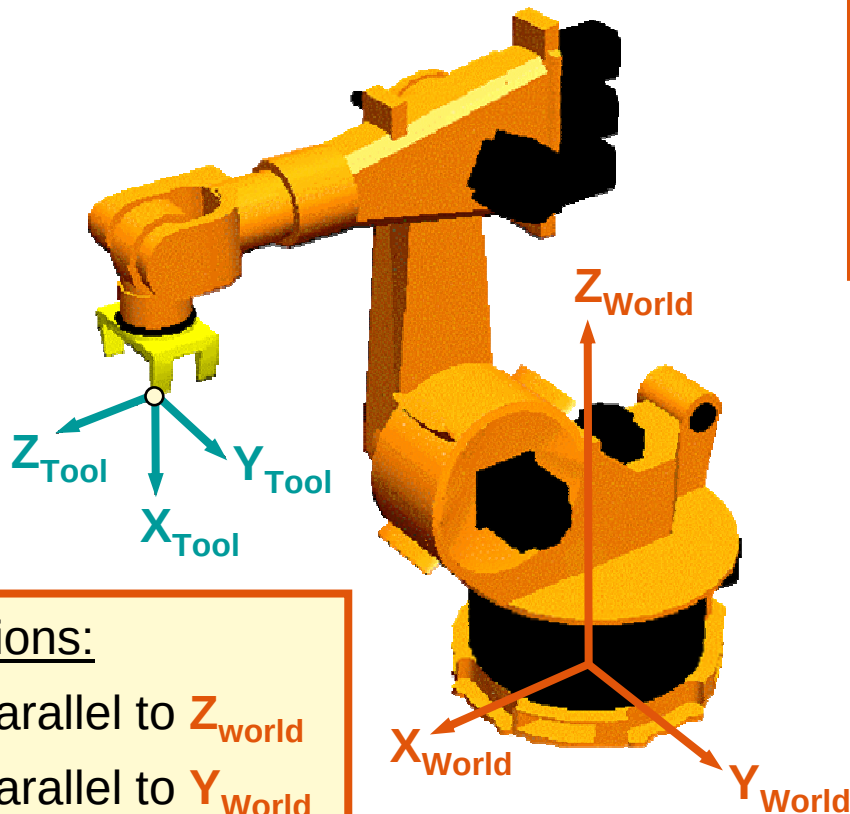
Conditions:

X_{Tool} parallel to Z_{World}



In this method the working direction of the tool has to be parallel to the Z-Axis of WORLD coordinate system. The orientation of the other axes will be determined by the controller.

ABC world 6D method



In this method the tool must be orientated in alignment with the WORLD coordinate system.

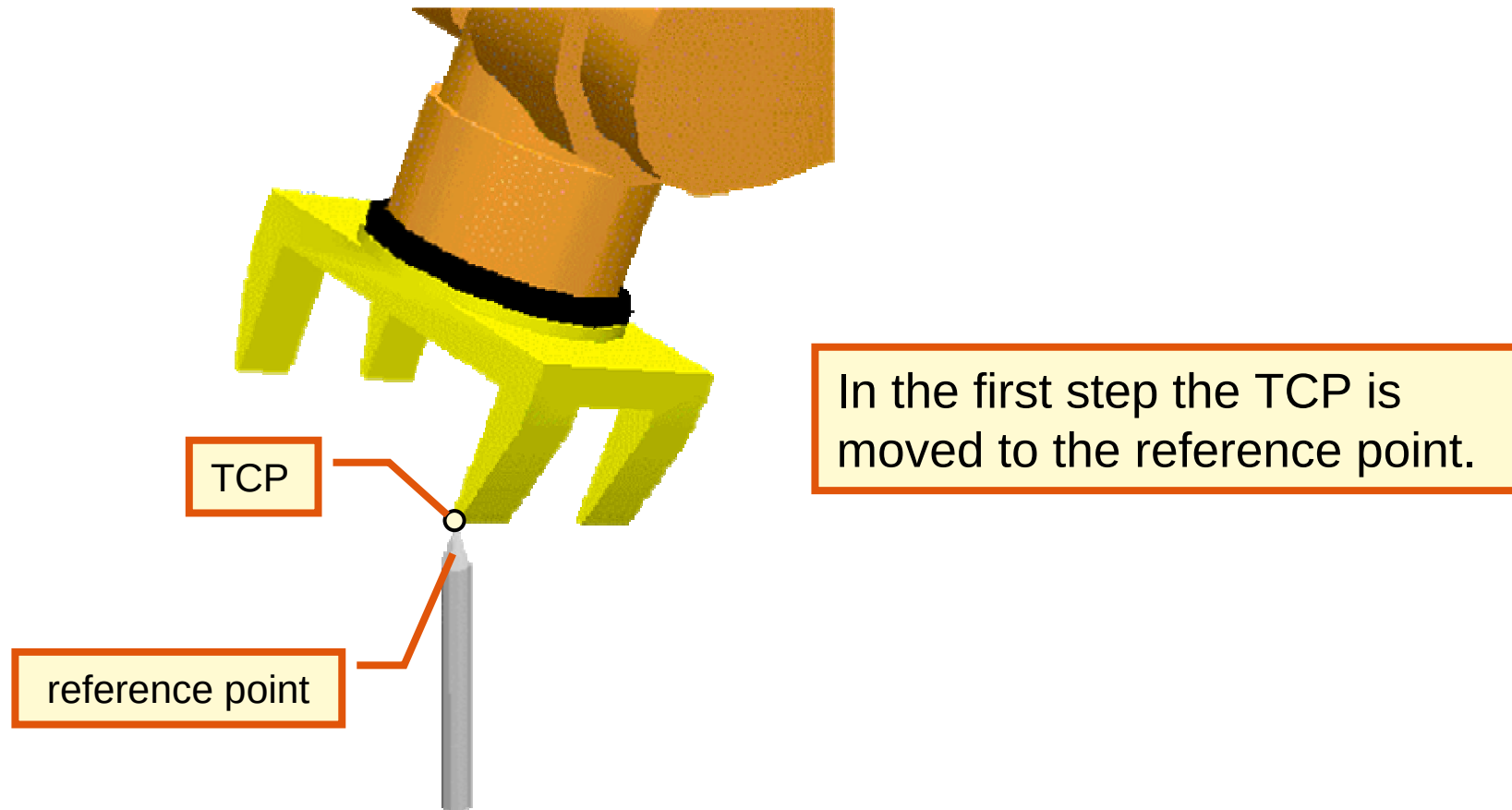
Conditions:

X_{Tool} parallel to Z_{World}

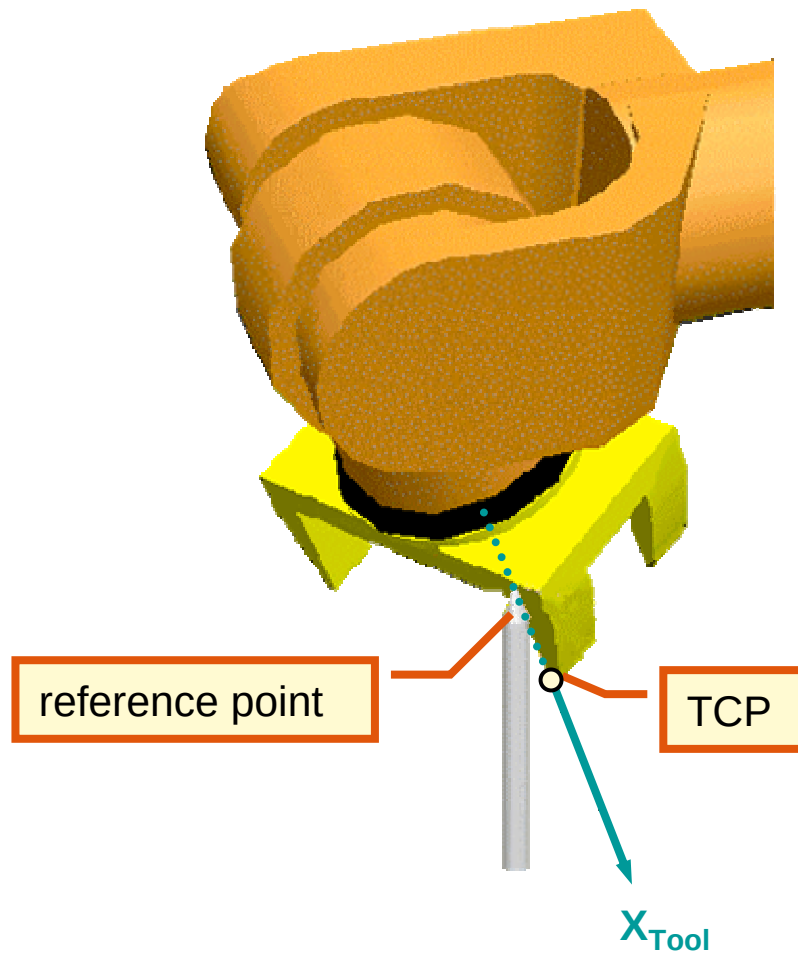
Y_{Tool} parallel to Y_{World}

Z_{Tool} parallel to X_{World}

ABC 2 Point method 1. Step

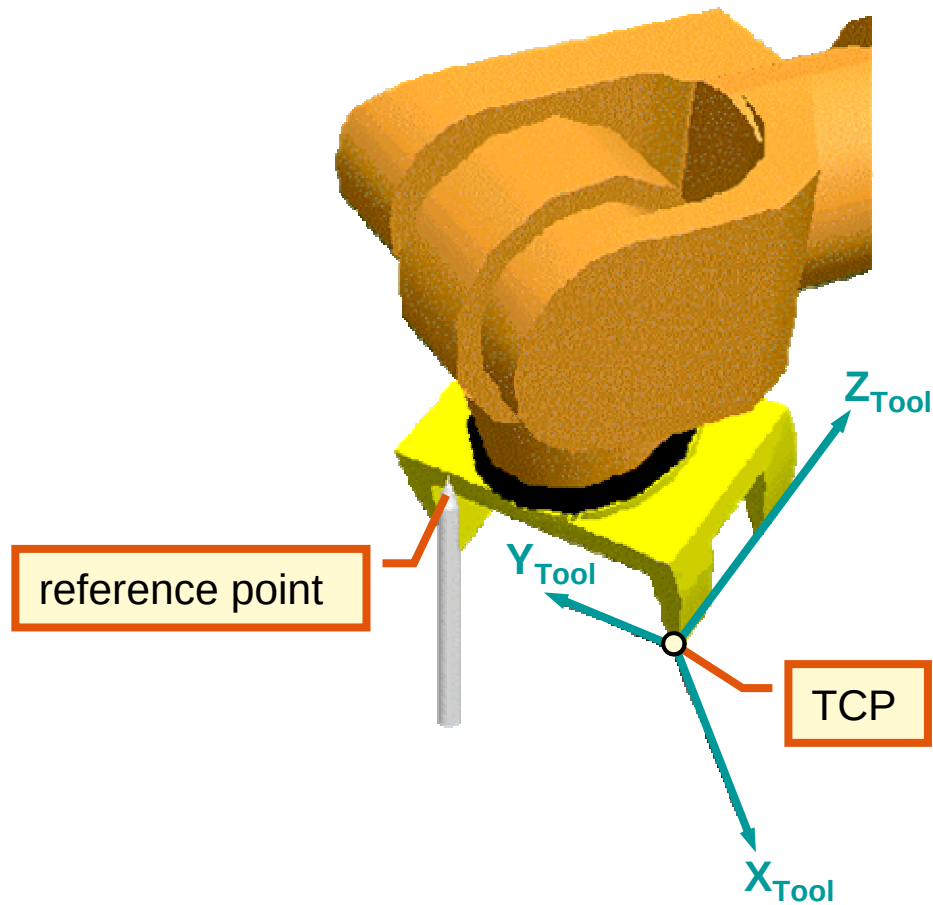


ABC 2 Point method 2. Step



In the second step a point in the reverse working direction is moved to the reference point.
Now the working direction is defined.

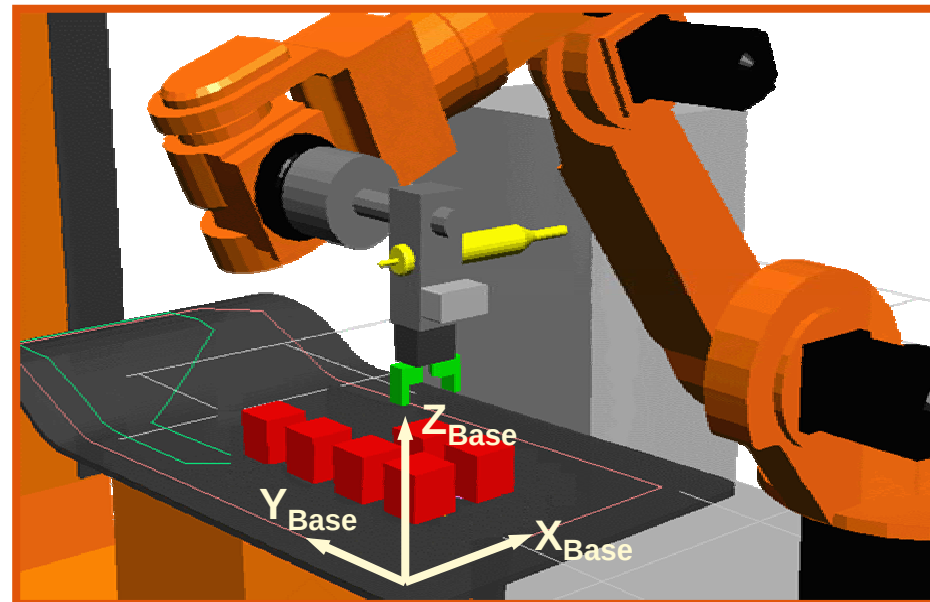
ABC 2 Point method 3. Step



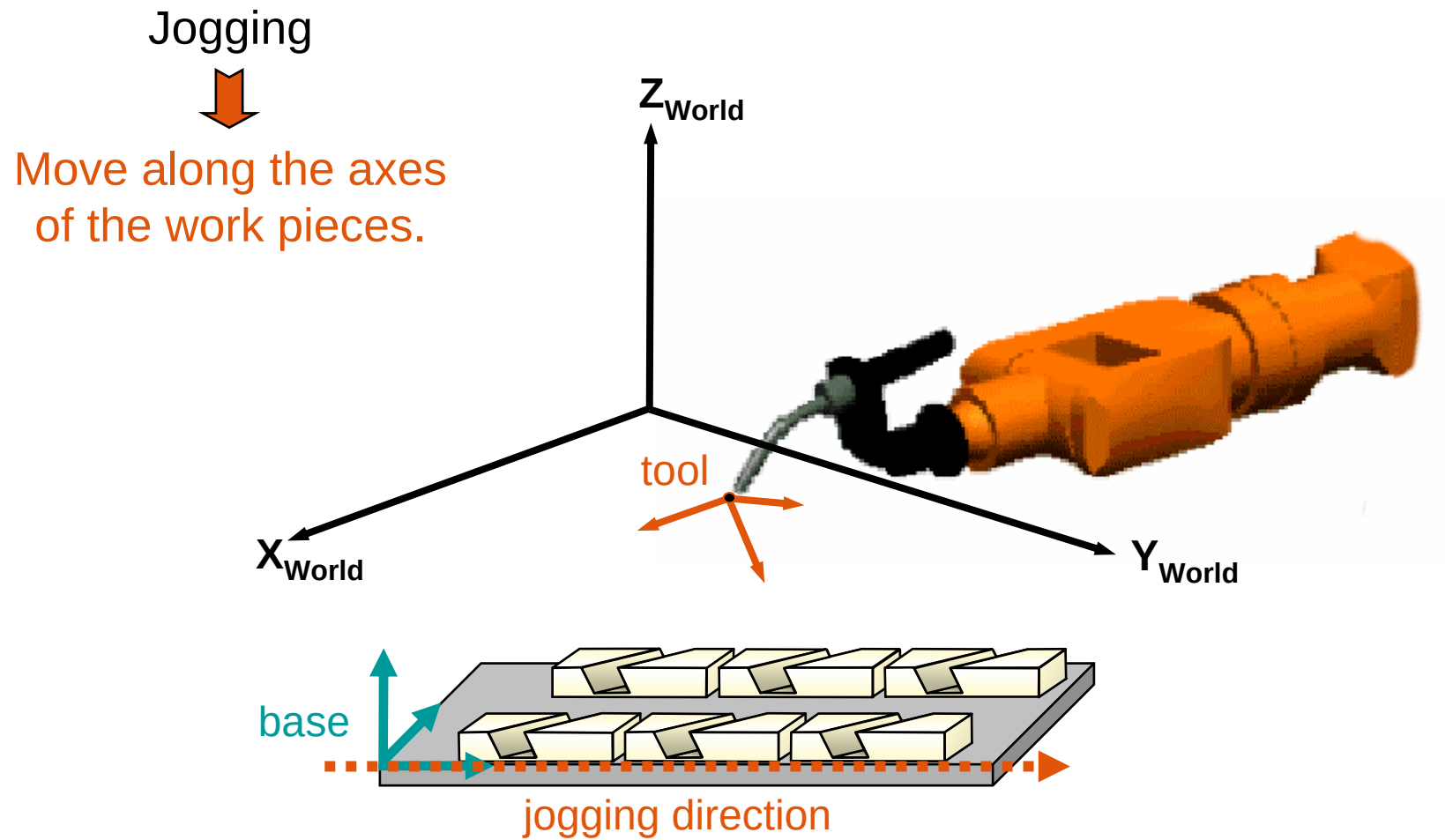
In the third step a point in the XY plane of the future tool coordinate system is moved to the reference point.

What happens during the base calibration ?

A **Cartesian coordinate system** which is linked to the work piece is defined. The origin (position) and orientation has to be defined by the user.



Sense of base calibration

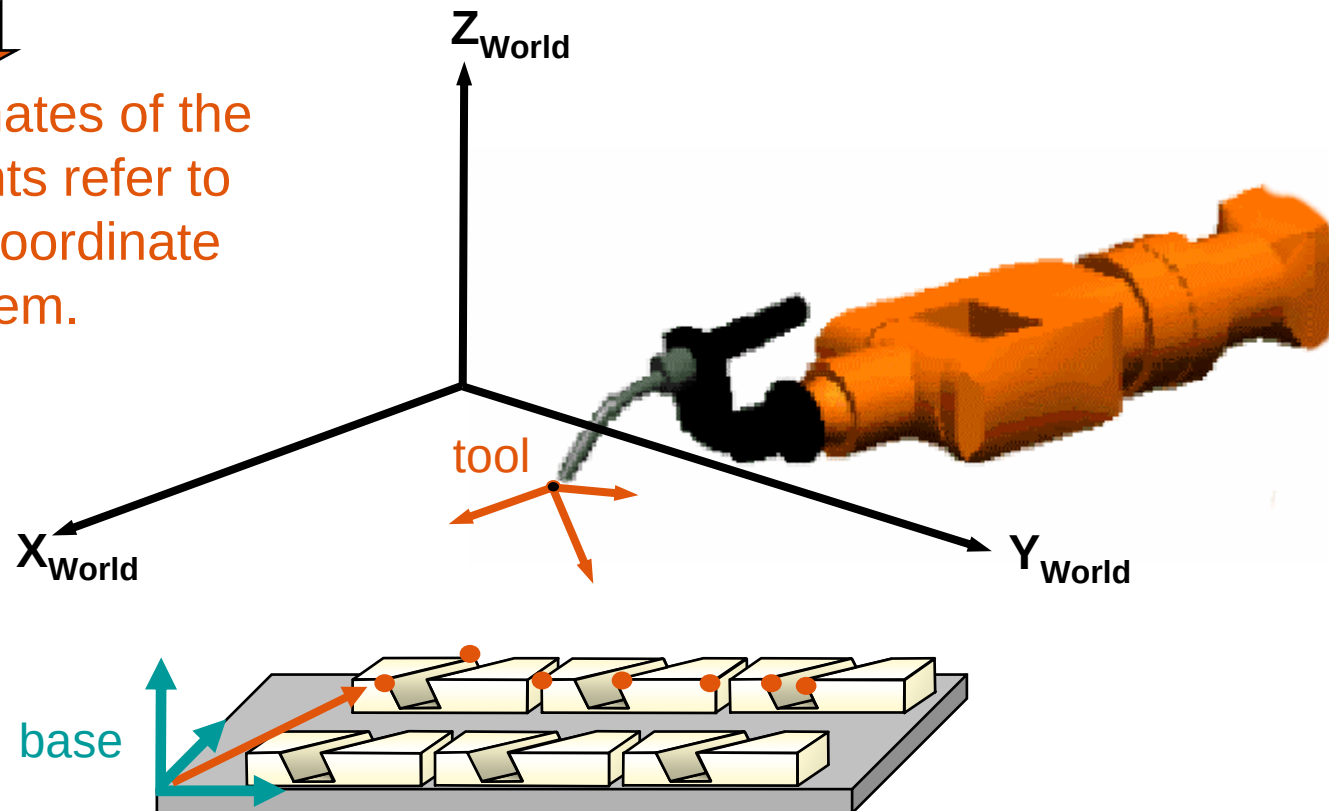


Sense of base calibration

Teaching points



The coordinates of the taught points refer to the base coordinate system.

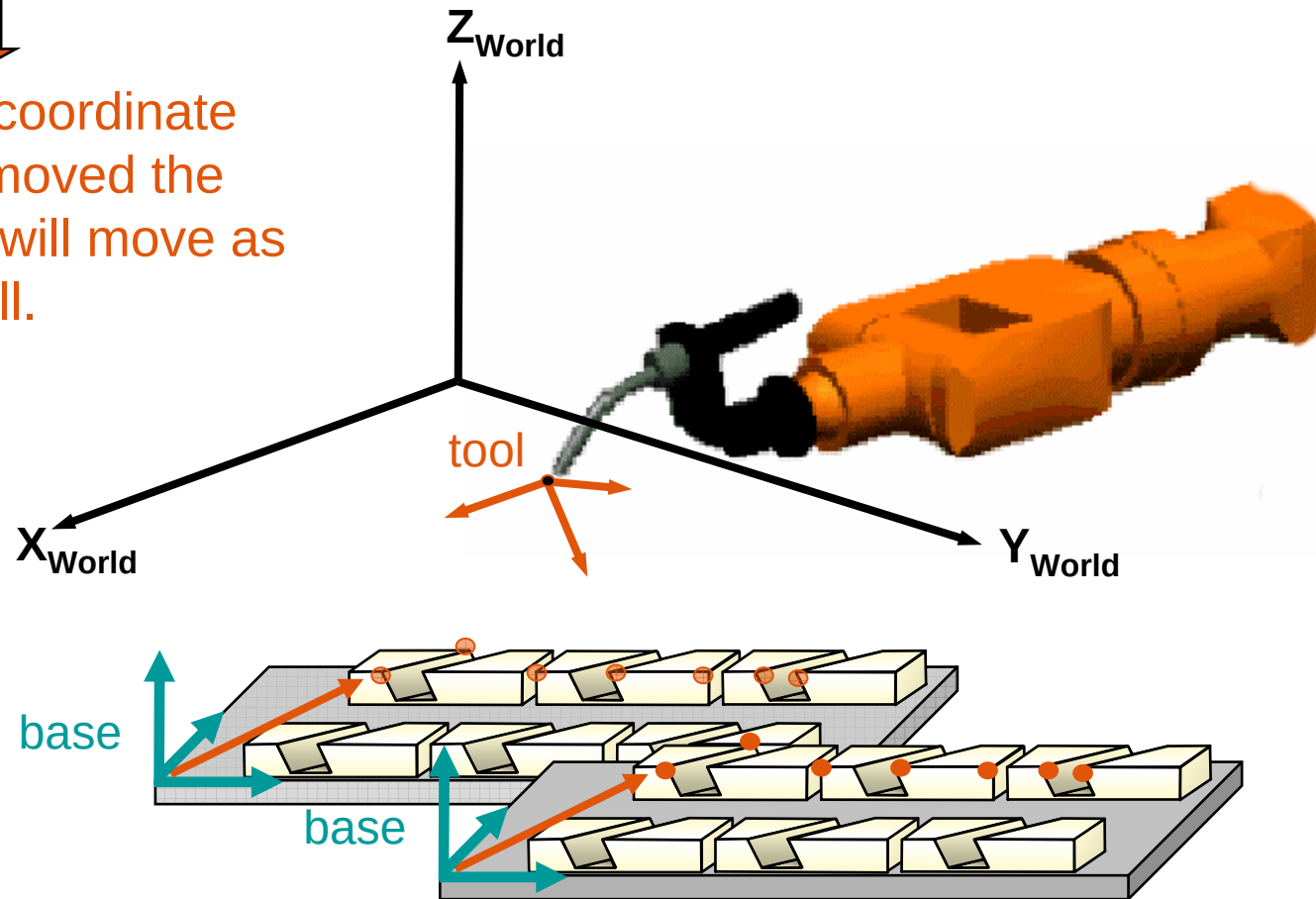


Sense of base calibration

Program operation



If the base coordinate system is moved the taught points will move as well.

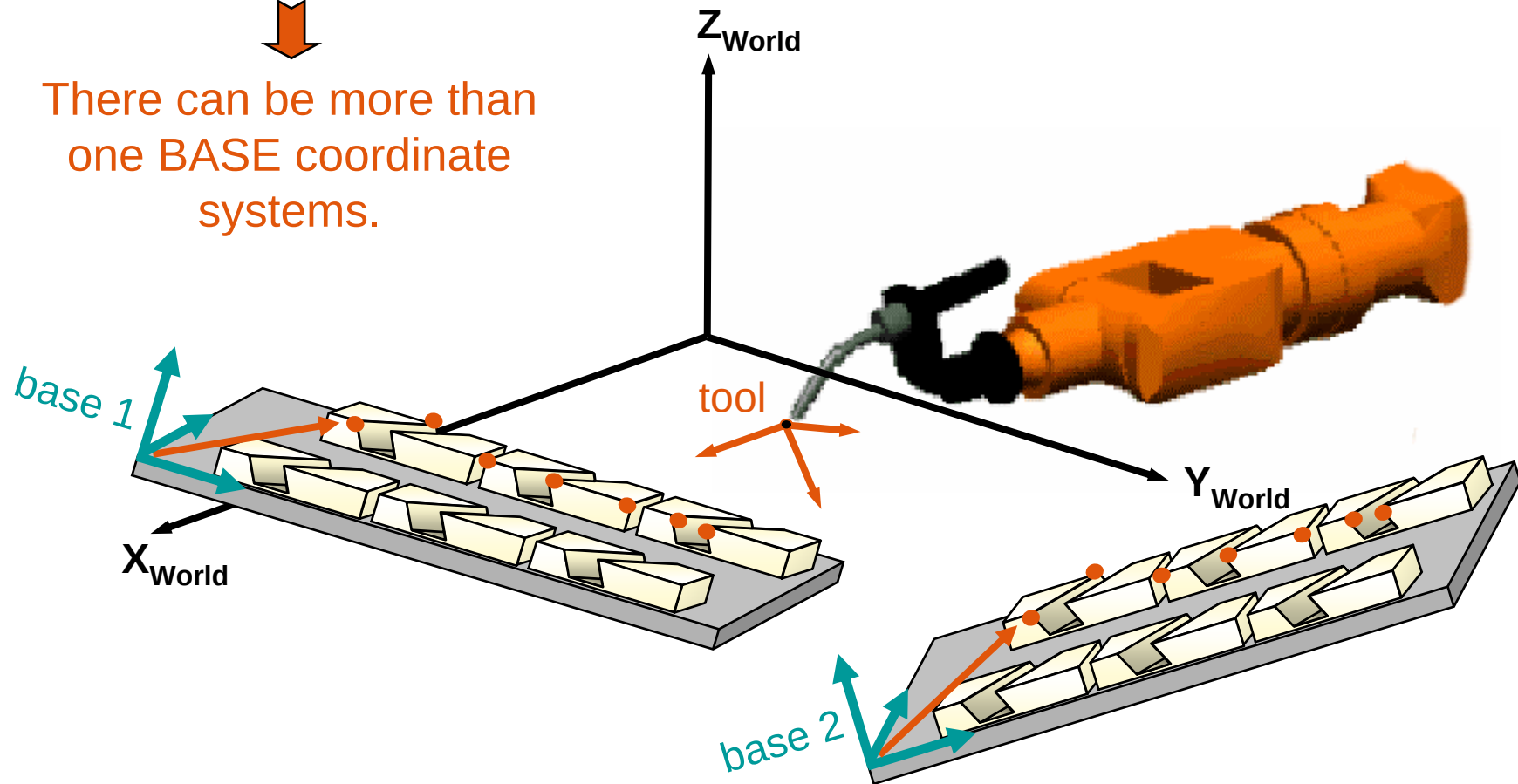


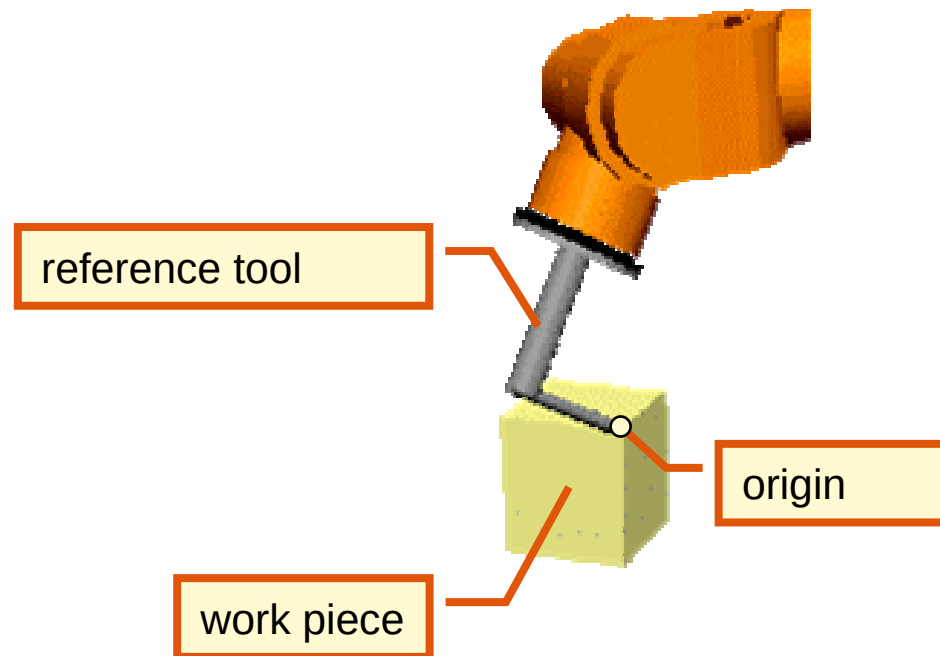
Sense of base calibration

Program operation

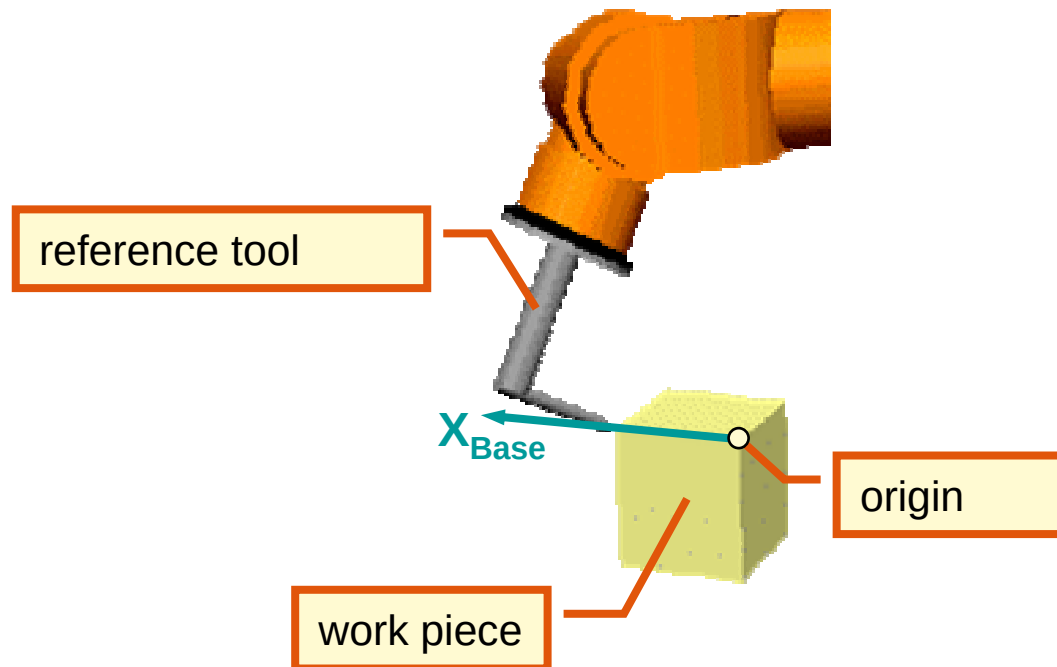


There can be more than
one BASE coordinate
systems.

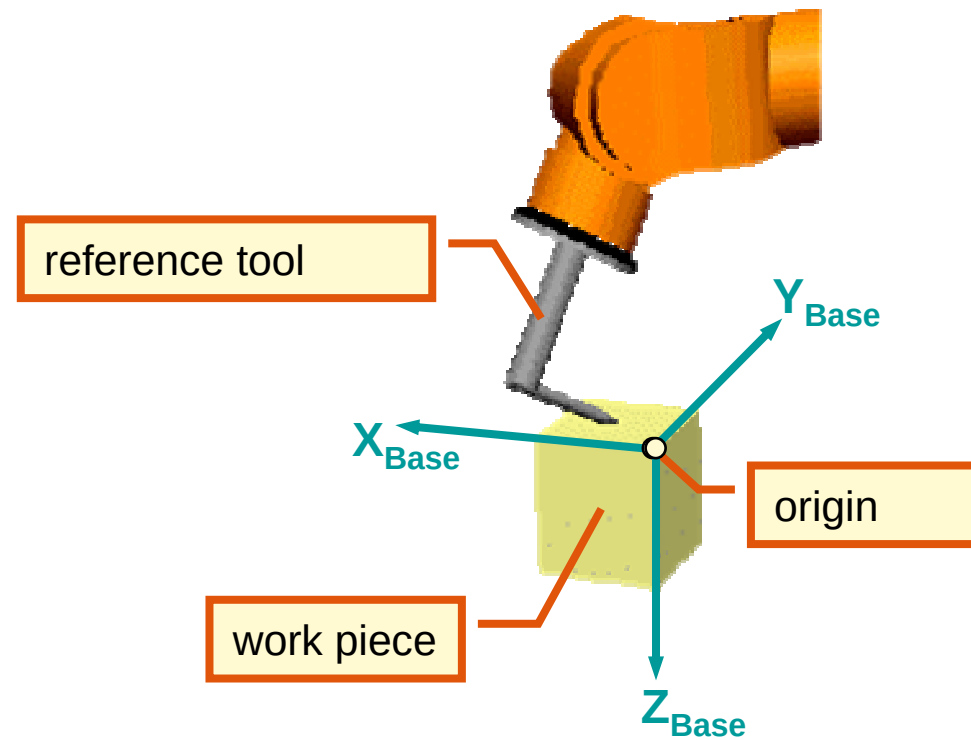




- In the first step the TCP is moved to the origin of the new base coordinate system. (position)



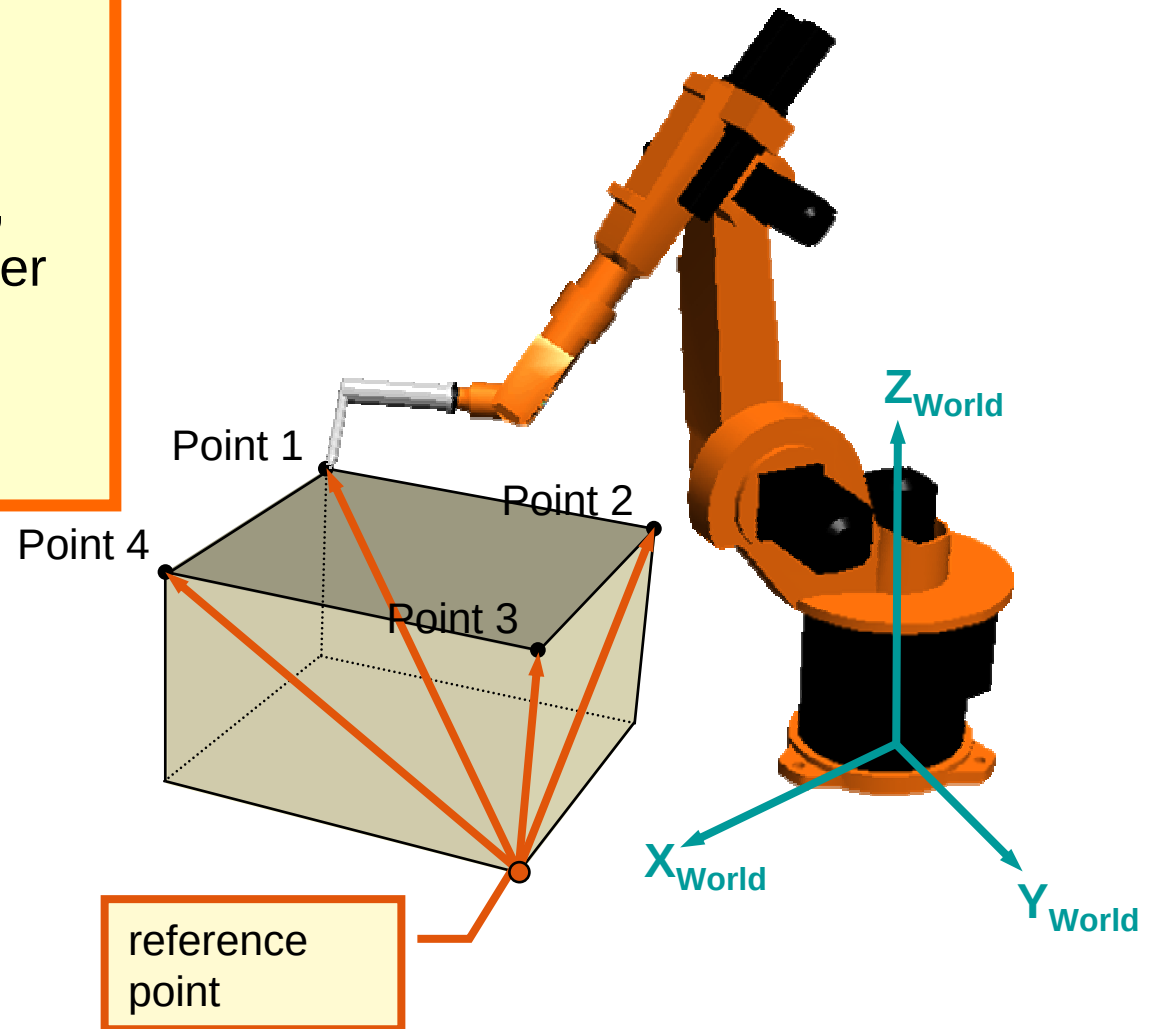
- In the second step the TCP is moved to a point on the positive X-axis of the new base coordinate system.



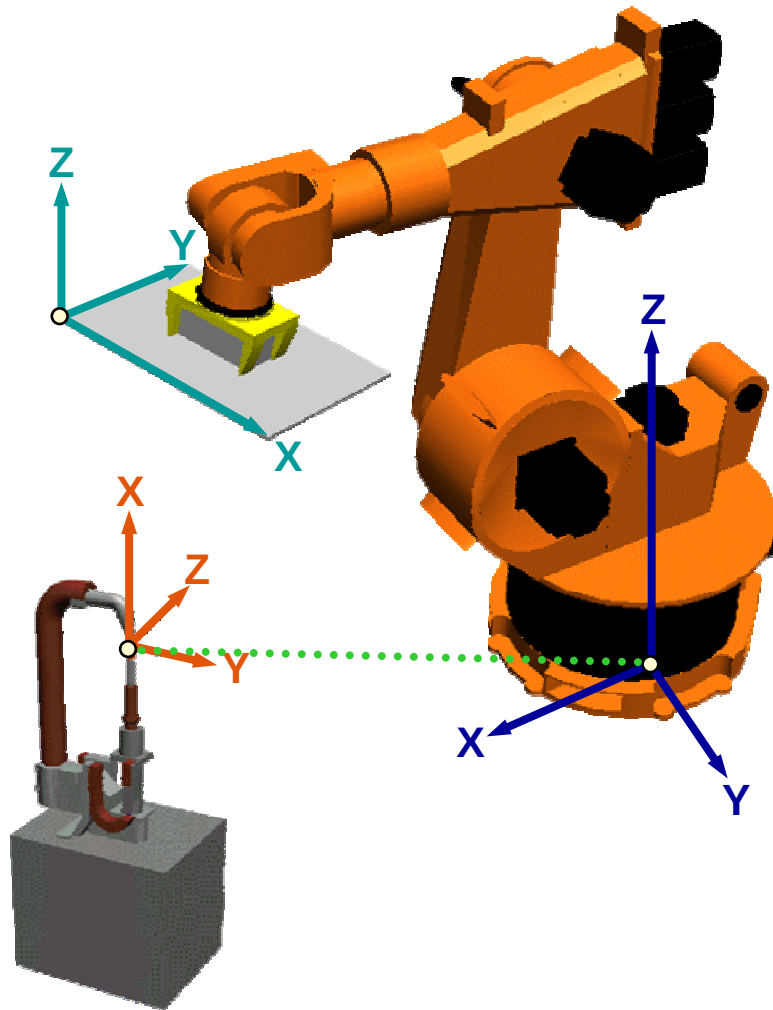
- In the third step the TCP is moved to a point with positive Y value on the XY plane of the new base coordinate system.

Indirect determination of the BASE coordinate system

In this method the TCP is moved to four points whose positions are known (e.g. CAD data, manufacturing drawing, etc.). The robot controller calculates the base coordinate system automatically.



Fixed tool

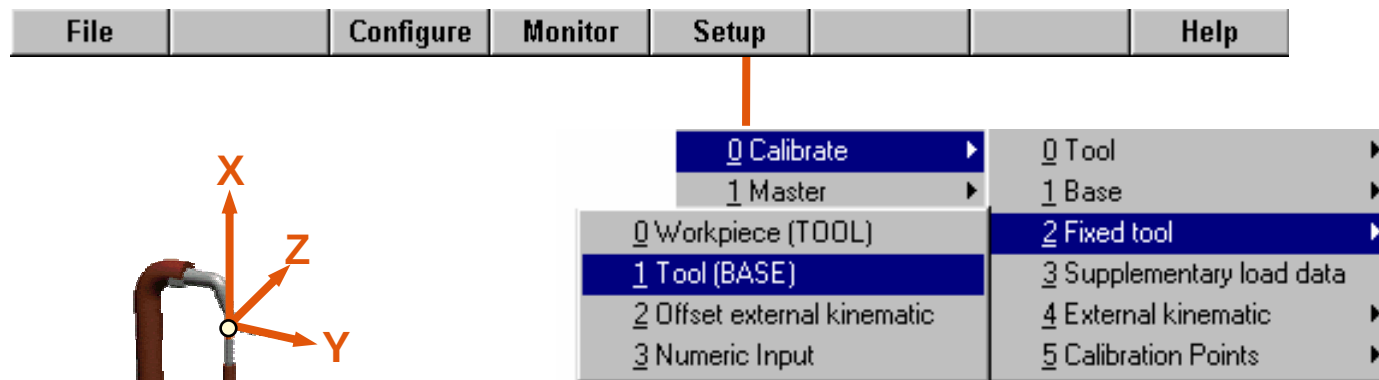


Using a “fixed tool” means, that the robot feeds one or more tools, which are steadily integrated in the work cell, with a work piece. The measuring consist of two parts:

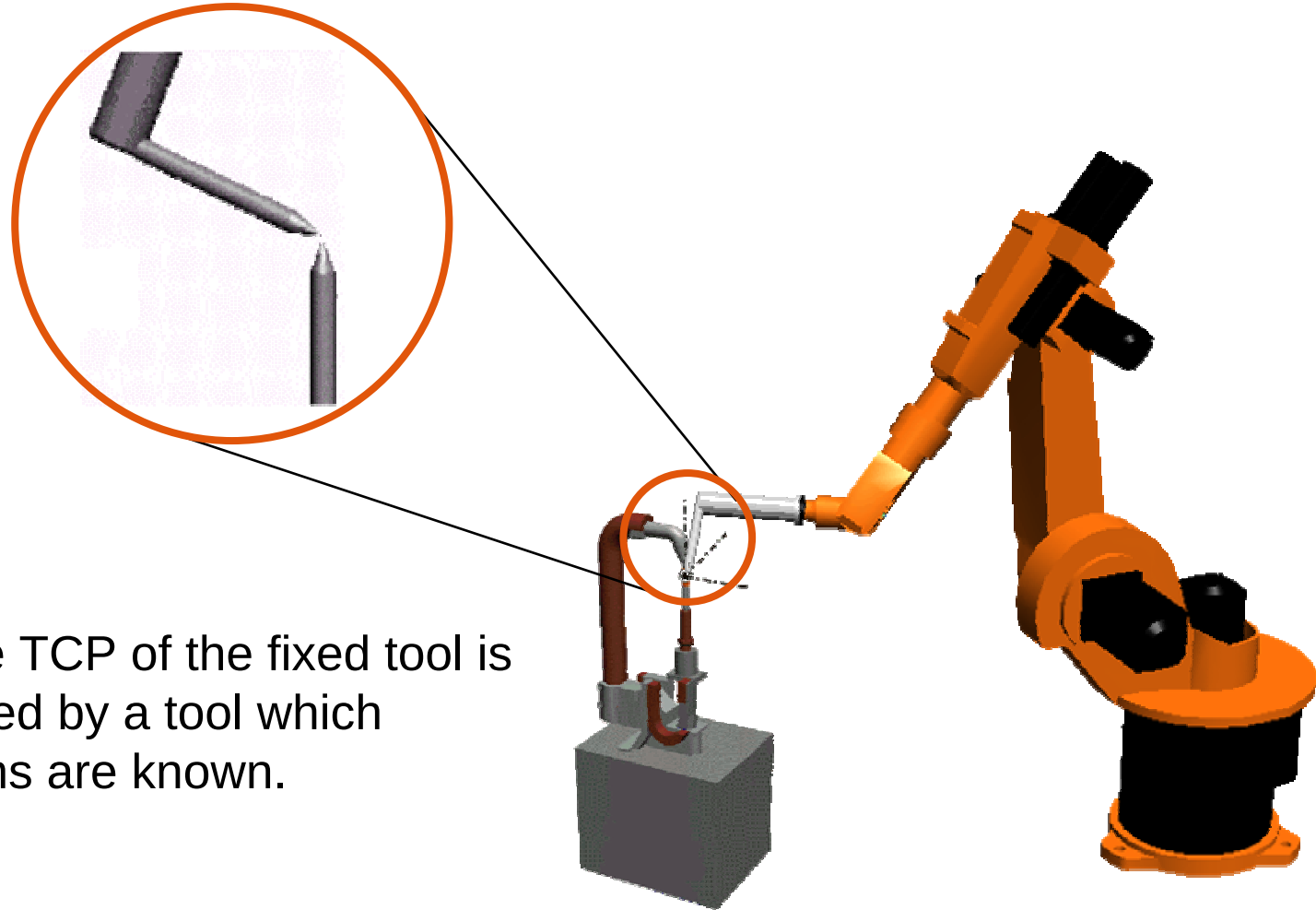
- Determination of the **distance** between the TCP of the fixed tool and the origin of the world coordinate system.
- The **work piece** is mounted to the flange of the robot and is calibrated as BASE.

Calibrating the fixed tool

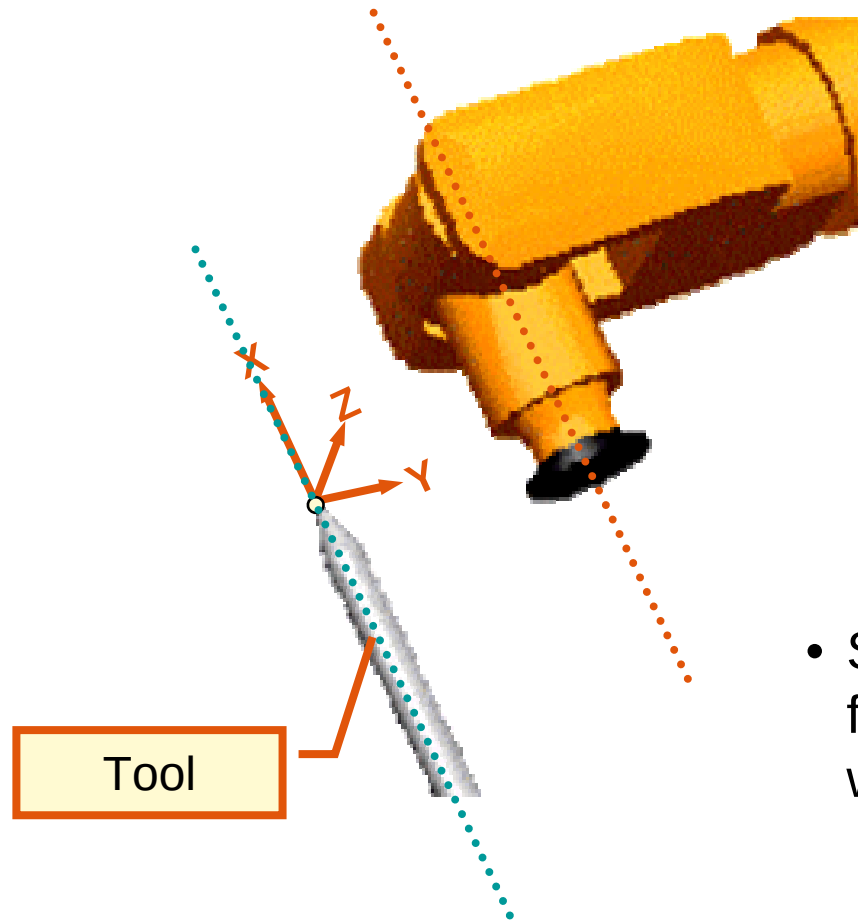
1. calibrate the fixed tool



The coordinates are saved as
BASE_DATA[1]....[16].



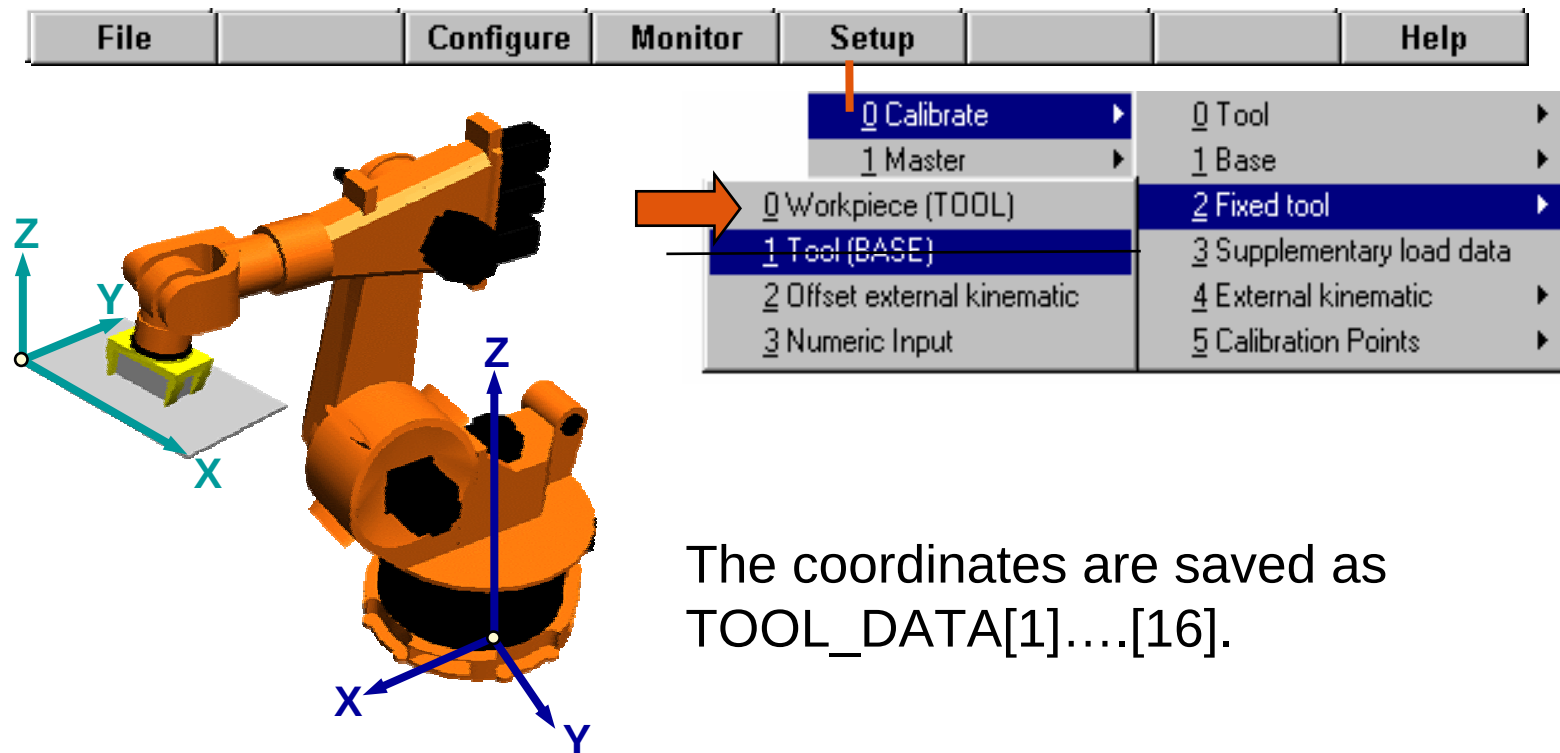
- At first the TCP of the fixed tool is approached by a tool which dimensions are known.



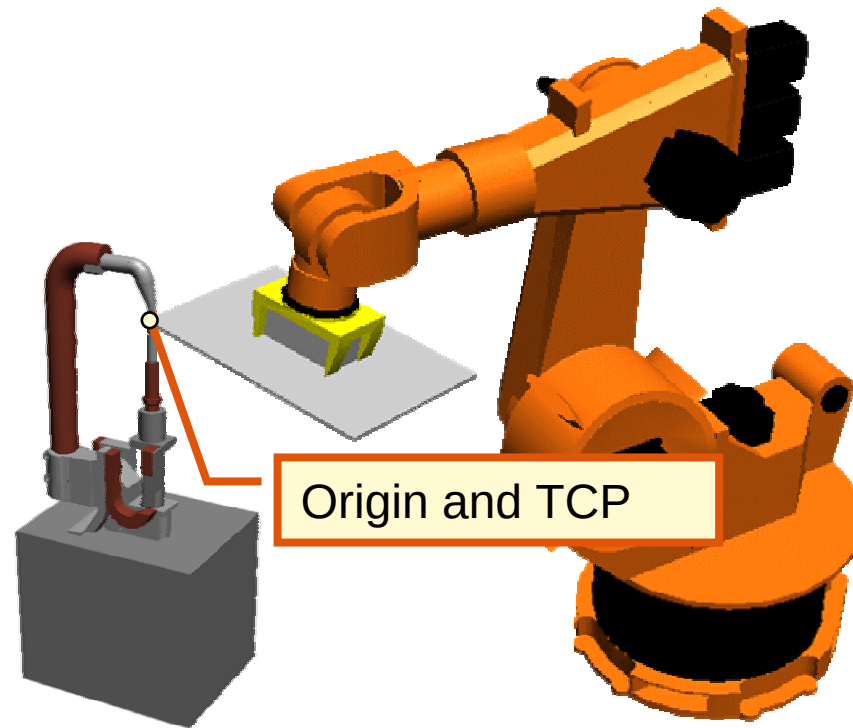
- Second step is to orientate the flange of the robot in the working direction of the tool.

Calibrating of the moveable work piece

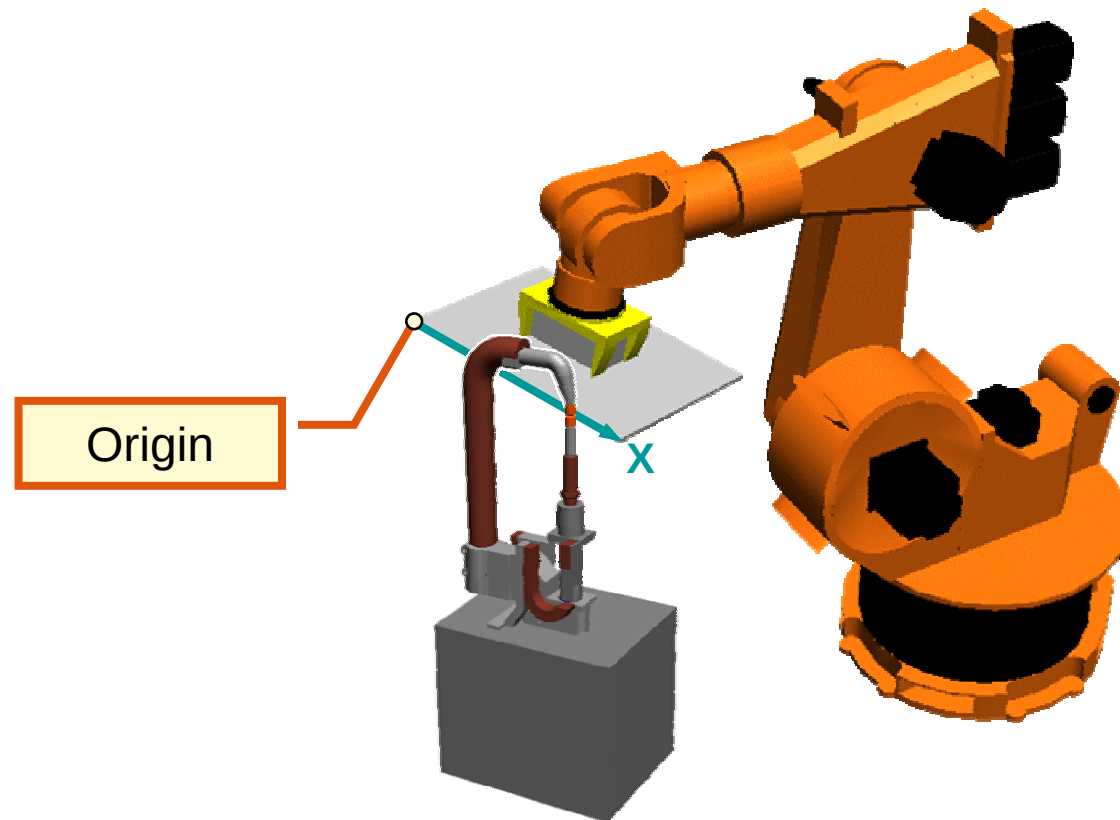
2. calibrate the work piece



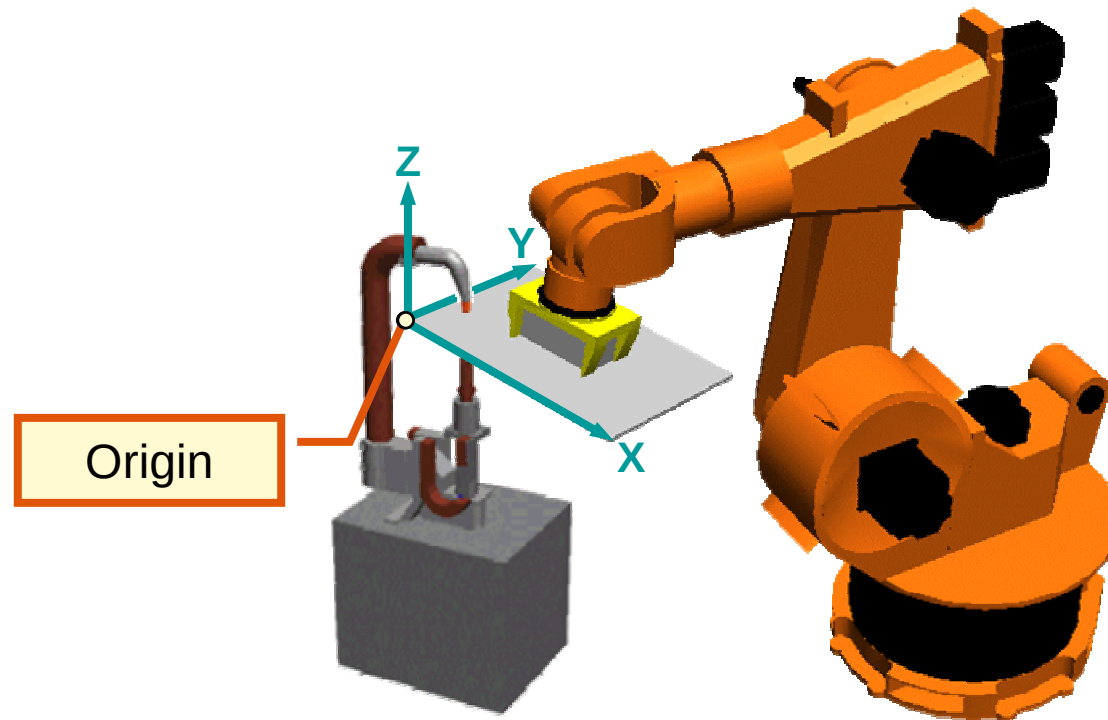
- The origin of the coordinate system of the work piece has to be moved to the TCP of the fixed tool.



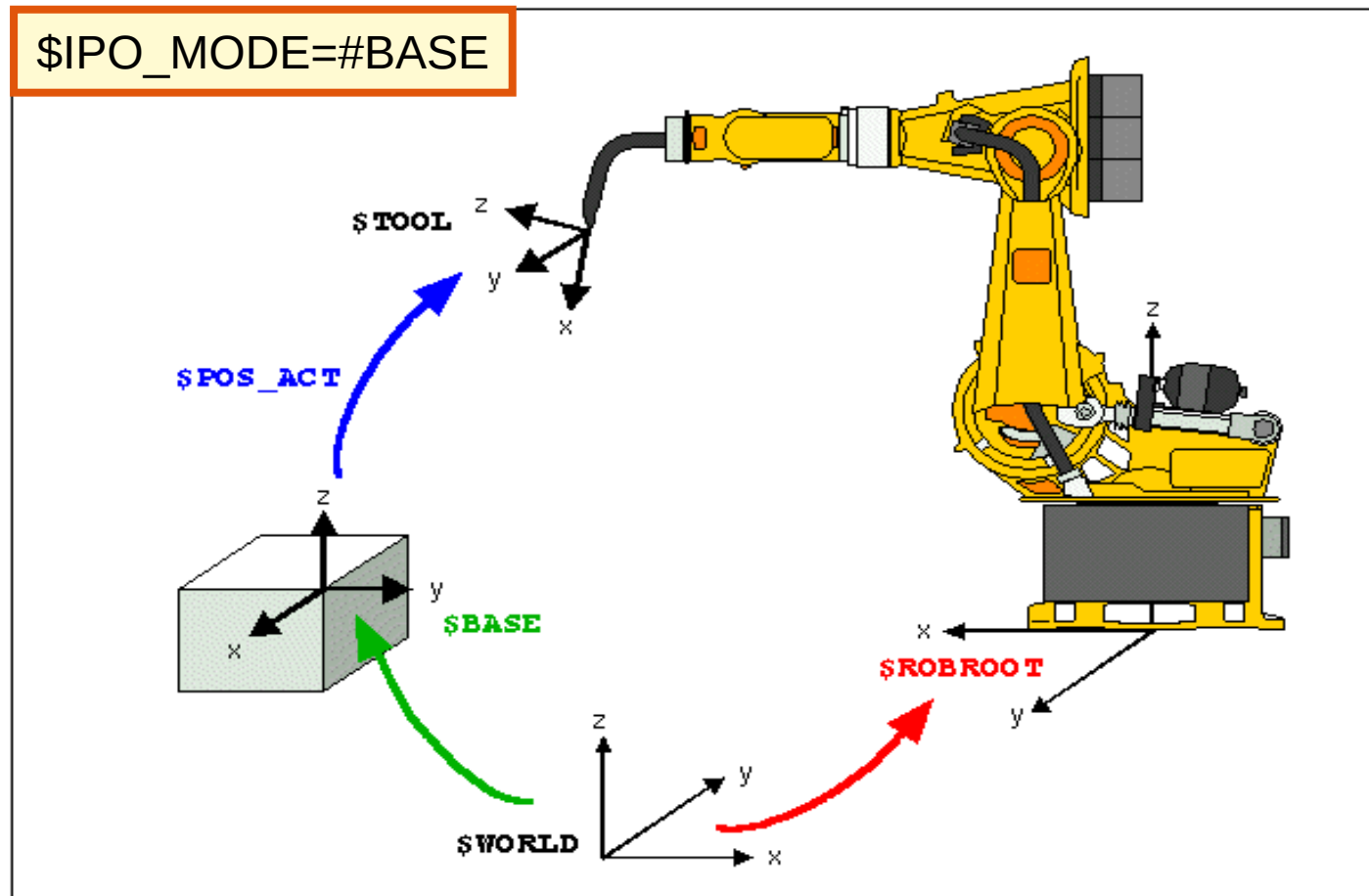
- Second step is to save a point, which is located on the positive x-axis of the coordinate system of the work piece.



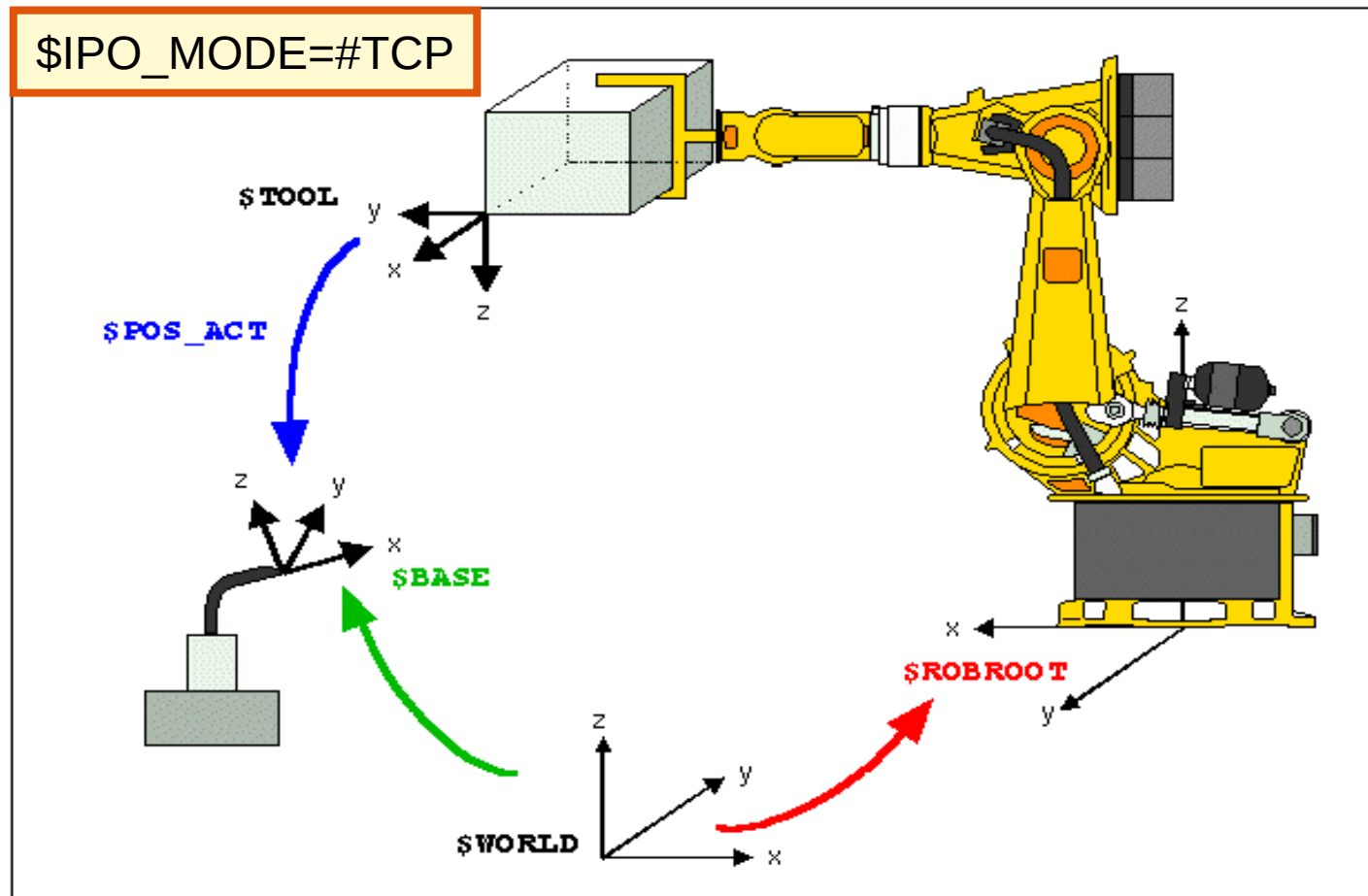
- Last step is to move to a point on the XY-plane of the coordinate system of the work piece, which has positive Y value.



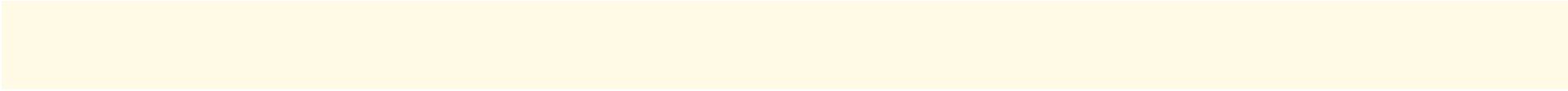
Kinematics chain in base-oriented interpolation



Kinematics chain in external TCP interpolation



运动编程



0	上 一个 指 令	
1	标 准 移 动	▶
2	工 艺 移 动	▶
3	宏 / 子 程 序 / 钳	▶
4	SPS ==>	▶
5	等 候 /FB	▶
6	ANA/BS/摆 动	▶
7	USER	
8	注 解	

总览

- 上一个指令 用来输入上一个被执行指令的命令；
- 标准移动 可以对 PTP 移动、LIN 移动和 CIRC 移动进行编程；
- 工艺移动 KLIN 移动、KCIRC 移动以及查找命令的编程。
- USER (用户) VW 用户在专家编程中编程；

指令

通过按菜单键“指令”，可以打开选择菜单及其下属菜单中的“标准移动”或者“工艺移动”。



移动方式

标准移动	
PTP (点到点)	工具沿着最快的轨迹运行至目标点。
LIN (线性)	工具以设定的速度沿一条直线移动。
CIRC (圆周)	工具以设定的速度沿圆周轨迹移动。
工艺移动	
KLIN (线性)	用于粘结应用场合，沿直线移动
KCIRC (圆周)	同样用于粘结场合，但是沿圆周轨迹运行
查找运行	传感器监视下的线性移动

 VB= 100 % VE= 0 % ACC= 100 % Wzg= 1 SPSTrig= 0 [1/100s]

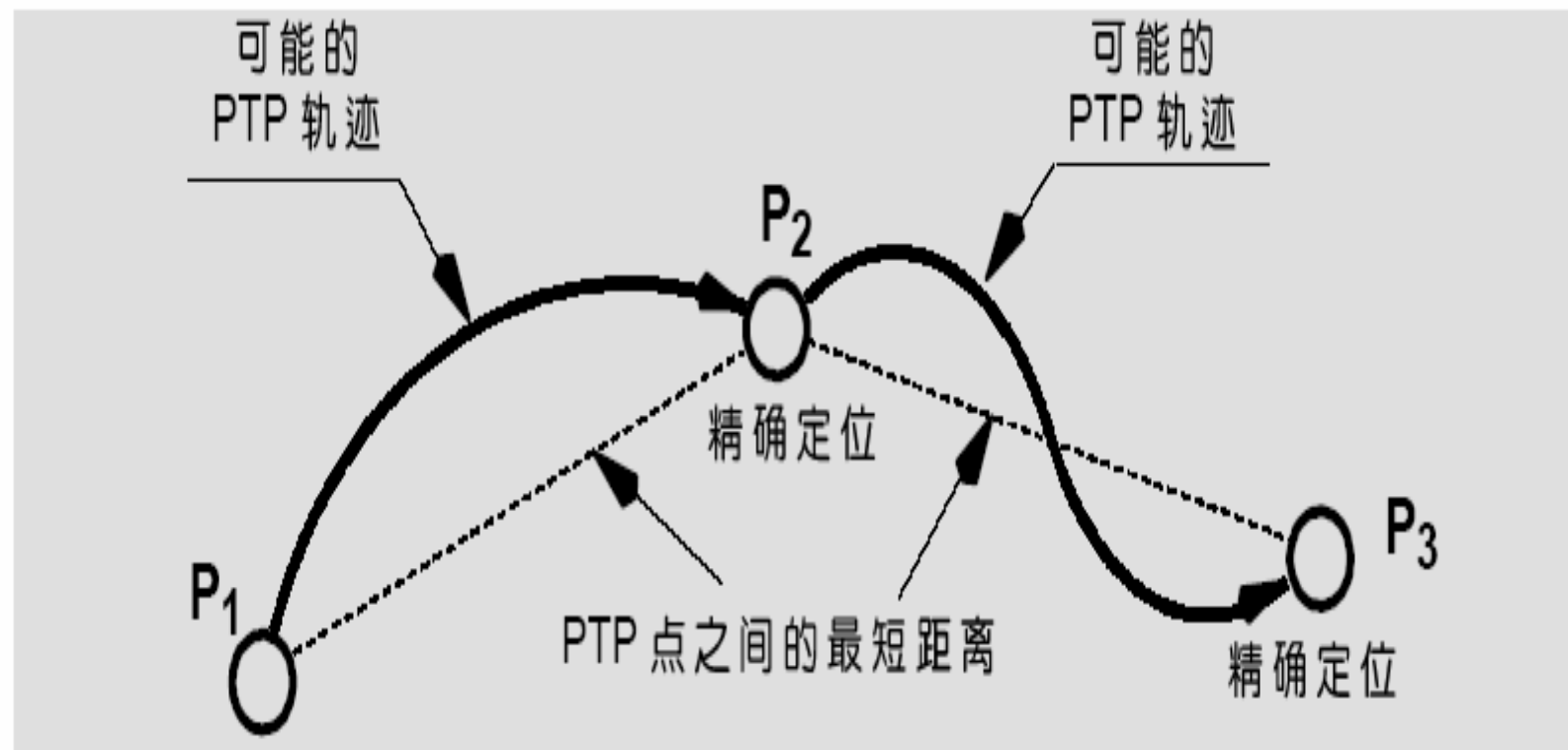
与此同时，软键条也发生变化。

指令退出			LIN/CIRC		坐标	指令认可
------	--	--	----------	--	----	------

呈深蓝底色的区域就是所谓的“反显”区，这表示：您可以在这个区域中进行改动。按住 Shift 键，同时移动光标键“←”及“→”，或者光标键“↑”及“↓”，可以从一个反显区移动到下一个反显区。

精确定位的 PTP 移动

在精确定位的 PTP 移动中，每个目标点都将精确地抵达各自的位置。



区域描述	功能	数值范围
PTP	移动方式	PTP、LIN、CIRC、KLIN、KCIRC
VB	移动速度	最大值的 1 % 至 100 % (预设值为 100 %)
VE	逼近区域	指令长度的一半的 0 % 至 100 % (预设值为 0 % = 无轨迹逼近)
ACC	加速度	最大值的 1 % 至 100 % (预设值为 100 %)
Wzg	所使用的工具的号码	1 至 16 (预设号码为 1)
SPSTrig	SPS 触发的时间点	0 至 100 ¹ / ₁₀₀ s (预设 为 0)



“PTP”输入窗中的值，可以通过相应状态键“+”及“-”，或者通过软键“LIN/CIRC”来进行改动。此外，您还可以使用（键盘的）数字区直接输入数值。

指令退出



通过软键“指令退出”，可以随时退出移动指令的程序编制状态，并且不存储数据。同样，Escape 键也具有这一功能。

LIN/CIRC

通过使用软键“LIN/CIRC”，您可以随时设置另一种移动方式，此操作与您在哪个输入窗中无关。您可以选择移动方式 PTP、LIN 或 CIRC。此软键上总是显示出尚未被使用的移动方式的名称。

坐标

如果您希望改动一个已经存在的 PTP 指令中的机器人坐标，则请按软键“坐标”。接下来在确认了软键“是”以后，当前的机器人坐标就被取用。

LIN 移动的编程

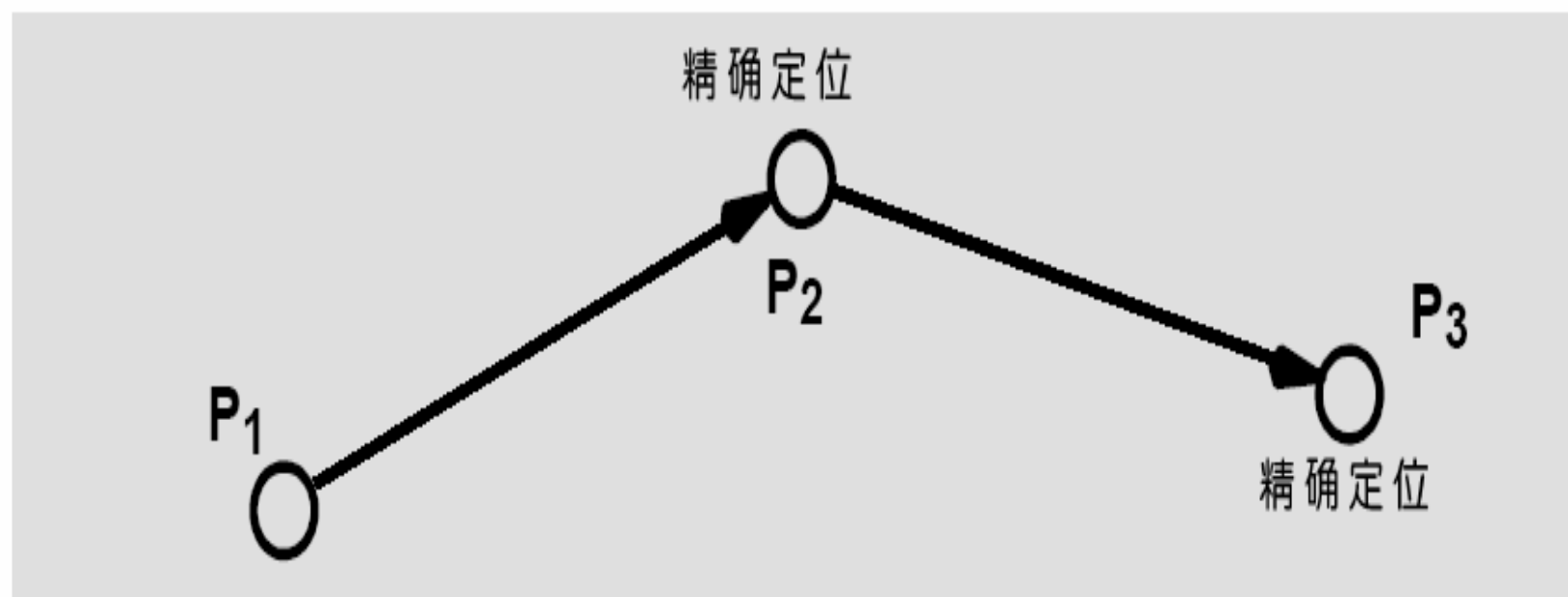
LIN ▾	VB=	2000	[mm/s]	VE=	0	%	ACC=	100	%	Wzg=	1	SPSTrig=	0
[1/100s]													

同时，软键条的状态也会发生变化。

指令退出			CIRC/PTP		坐标	指令认可
------	--	--	----------	--	----	------

精确定位的 LIN 移动

在精确定位的 LIN 移动中，每个目标点的位置都将被精确地触及。



区域描述	功能	数值范围
LIN	移动方式	PTP、LIN、CIRC、KLIN、KCIRC
VB	轨迹速度	1 至 2000 mm/s (预设值为 1750 mm/s)
VE	逼近区域	指令长度的一半的 0 % 至 100 % (预设值为 0 % = 无轨迹逼近)
ACC	加速度	最大值的 1 % 至 100 % (预设值为 100 %)
Wzg	所使用的工具的号码	1 至 16 (预设号码为 1)
SPSTrig	SPS 触发的时间点	0 至 100 ¹ / ₁₀₀ s (预设值为 0)

线性移动 (LIN)

在做 LIN 移动时，工具尖端从起点到目标点做直线运动。因为两点确定一条直线，所以只要给出目标点就可以了。在这种移动方式下，所有参加移动的轴均处在一条直线上。

LIN 移动总是应用于：当需要在两点之间进行一个已经定义了轨迹速度的精确的轨迹移动时，或者当如果做 PTP 移动，则抵达目标位置时会有碰撞危险的情况下。

0 上一个指令

- 1 标准移动 ▶
- 2 工艺移动 ▶
- 3 宏/子程序/钳 ▶
- 4 SPS ==> ▶
- 5 等候 /FB ▶
- 6 ANA/BS/摆动 ▶
- 7 USER
- 8 注解

圆周移动（CIRC）

在做 CIRC 移动时，工具尖端从起点到目标点沿着圆周轨迹运动。此圆周轨迹由三个点来确定：起始点、辅助点和结束点。起始点是上一个指令的精确定位点。对于辅助点来说，只是坐标 X、Y 和 Z 起决定作用。在移动过程中，工具尖端取向的变化顺应于持续的移动轨迹。

CIRC 移动总是应用于：当已经定义了速度的工作进程，需要在圆周轨迹上完成的时候。

精确定位的 CIRC 移动

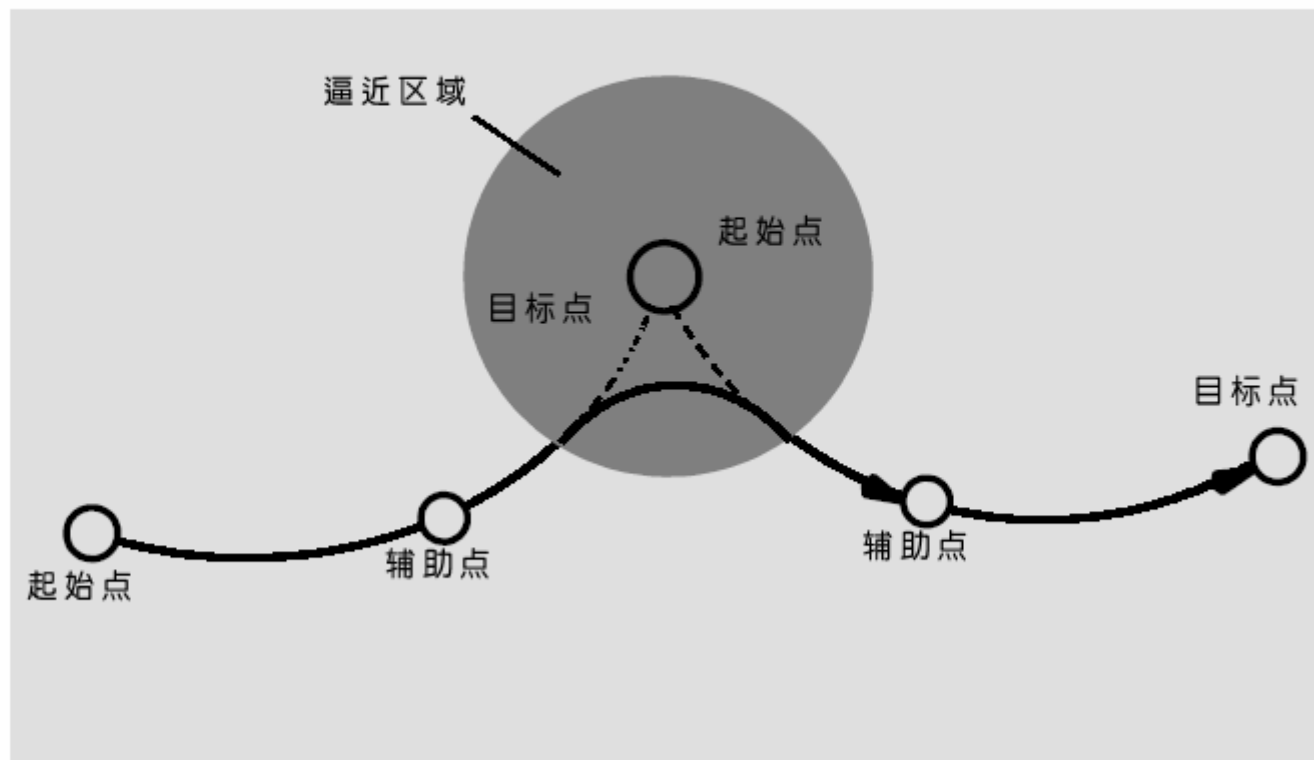
在精确定位的 CIRC 移动中，每个目标点的位置都将被精确地触及。

CIRC 移动轨迹通过起始点、辅助点和目标点来确定。



轨迹逼近的 CIRC 移动

当两个 CIRC 移动通过轨迹逼近互相连接时，控制装置在逼近区域内将两个抛物线部分组合为一个移动轨迹。



CIRC 移动的编程

选择菜单项“CIRC”之后，在程序窗中会出现 InLine 表格，以备填写 CIRC 指令的参数。

CIR ▾	VB=	2000	[mm/s]	VE=	0	%	ACC=	100	%	Wzg=	1	SPSTrig=	0
[1/100s]													

同时，软键条的状态也会发生变化。

指令退出			PTP/LIN	坐标 HP	坐标 ZP	指令认可
------	--	--	---------	-------	-------	------

区域描述	功能	数值范围
CIP	移动方式	PTP、LIN、CIRC、KLIN、KCIRC
VB	轨迹速度	1 至 2000 mm/s (预设值为 1750 mm/s)
VE	逼近区域	指令长度的一半的 0 % 至 100 % (预设值为 0 % = 无轨迹逼近)
ACC	加速度	最大值的 1 % 至 100 % (缺省值为 100 %)
Wzg	所使用的工具的号码	1 至 16 (缺省值为 1)
SPSTrig	SPS 触发的时间点	0 至 100 ¹ / ₁₀₀ s (缺省值为 0)

PTP/LIN

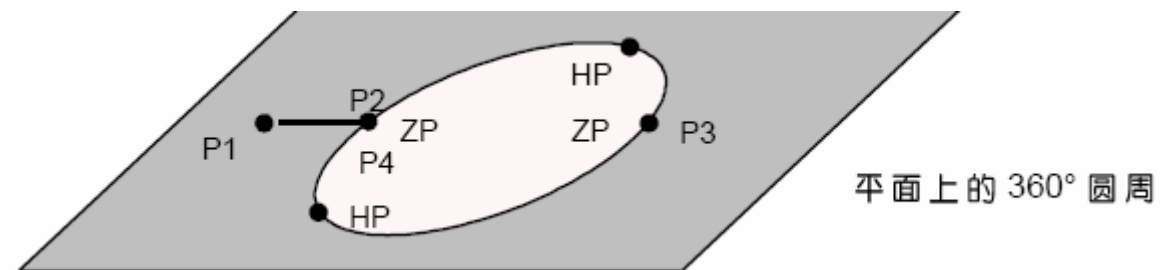
通过使用软键“PTP/LIN”，您可以随时设置另一种移动方式，此操作与您在哪个输入窗中无关。您可以选择移动方式 PTP、LIN 或 CIRC。此软键上总是显示出尚未被使用的移动方式的名称。

坐标 HP

通过按软键“坐标 HP”，可以将当前坐标作为辅助点（简称：HP）的坐标储存。

坐标 ZP

通过按软键“坐标 ZP”，可以将当前坐标作为目标点（简称：ZP）的坐标储存。



平面上的 360° 圆周

工艺移动

在执行 KLIN 和 KCIRC 功能时，由于起始斜面和制动斜面的坡度较大，并且加速度过高，所以不允许使用重工具或者焊钳一类的工作。这两种功能可以应用于粘结或者气体保护焊等方面。

对于 KLIN 功能来说，最大可能的加速度为 2.3 m/s^2 ，而对于 KCIRC 来说，则可以达到 4.6 m/s^2 。

线性移动 (KLIN)



KLIN 功能的选择，只能通过右下方的状态键 “+” 及 “-” 来实现。

与普通的线性移动 LIN 不同，在做 KLIN 移动的时候，以 mm 为单位给出逼近标准。因此，在做高速的轨迹移动的时候，也可以实现轨迹逼近，从而来完成粘结任务。

如果输入窗 “Genau (精确)” 处于反显状态，则您可以输入数值，此值表示出：在距离到达目标点多少 mm 处应该开始轨迹逼近。当您填入 0 的时候，将不发生逼近移动。

其他方面，指令 “KLIN” 与移动指令 “LIN” 相同。

KLIN ▾ VB= 2000 [mm/s] Genau= 0 [mm] ACC= 100 % Wzg= 1 SPSTrig= 0
[1/100s]

圆周移动 (KCIRC)



KCIRC 功能的选择，也只能通过右下方的状态键“+”及“-”来实现。

与普通的圆周移动 CIRC 不同，在做 KCIRC 移动的时候，以 mm 为单位给出逼近标准。因此，在做高速的轨迹移动的时候，也可以实现轨迹逼近，从而来完成粘结任务。

如果输入窗“Genau (精确)”处于反显状态，则您可以输入数值，此值表示出：在距离到达目标点多少 mm 处应该开始轨迹逼近。当您填入 0 的时候，将不发生逼近移动。

其他方面，指令“KCIRC”与移动指令“CIRC”相同。

KCIR ▾ VB= 2000 [mm/s] Genau= 0 [mm] ACC= 100 % Wzg= 1 SPSTrig= 0
[1/100s]

查找运行程序的编制

选择菜单键“指令”，及菜单项“查找运行”之后，在程序窗中会出现 InLine 表格，以备填写查找运行指令的参数。



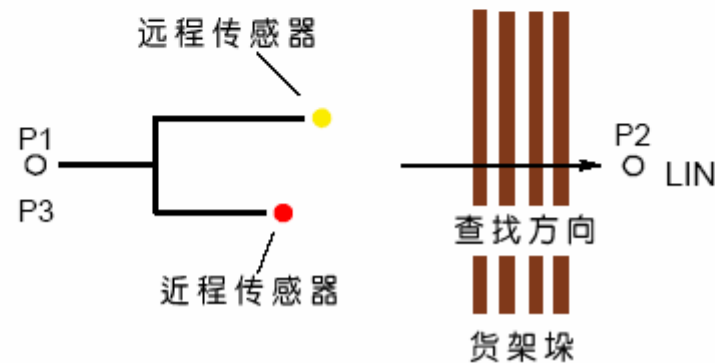
同时，软键条的状态也会发生变化。



通过此功能，可以完成例如搬放整理之类的任务。

在查找运行时，机器人向一点做线性移动。此时，您应该设置查找运行的目标点，而不是起始点。

在下面的例子中给出了点 P2。查找从点 P1 开始进行。如果远程传感输入端（“Fern”区）被置位，那么查找速度降低为“Vred”区中的值。“Vred”的值是程序编定的轨迹速度 VB（单位 [mm/s]）的百分比例值。如果远程传感输入端未被置位，或者“Vred”区中的值为100，则查找速度将不会降低。在近程传感输入端（“Nah”区）被置位的情况下，查找运行以及与此有关的机器人移动将停止，并且机器人向最近的点开行。在此例中为点 P3。

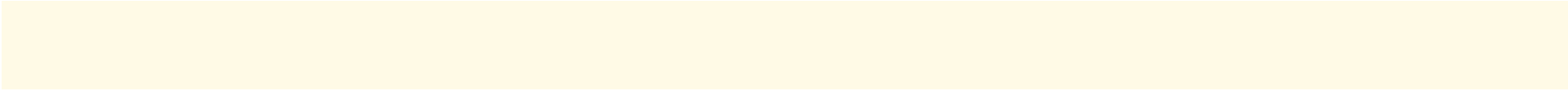


区域描述	功能	数值范围
VB	轨迹速度	1 至 2000 mm/s (预设值为 1750 mm/s)
Wzg	所使用的工具的号码	1 至 16 (预设号码为 1)
Fern	远程传感输入端	1 至 1024 (预设值为 1)
VRED	速度降低	轨迹速度 (VB) 的最大值的 1 % 至 100 % (预设值为 10 %)
Nah	近程传感输入端	1 至 1024 (预设值为 1)



数字输入窗的值可以通过数字区或者状态键“+/-”来改变。

SPS编程



在一个工作程序中，您可以为某个移动命令添加所谓的 SPS 指令。这些指令将根据 SPS 的触发情况而执行。

选定工作程序

选择



在为一个移动命令添加 SPS 指令之前，必须首先打开这个工作程序。通过软键“选定”，程序显示于显示屏上。接下来您可以使用方向键“↓”及“↑”，将编辑光标移动至欲编辑的行，即：您希望在其下插入新的控制命令的行。

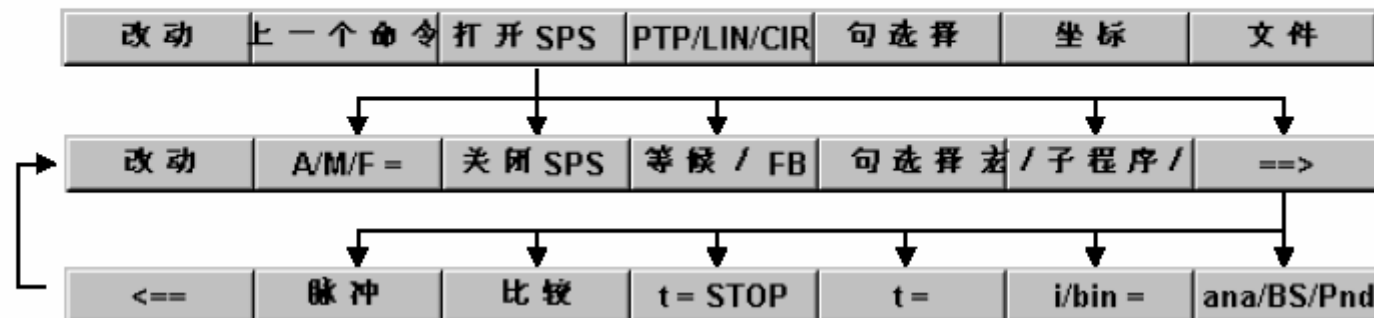
新的程序行总是插入在编辑光标下面的行中。



打开 SPS

打开 SPS

您已经位于选定的程序之后，必须通过位于显示屏下方边缘的软键“打开 SPS”，来打开编程点所属的折合。然后您可以调用相应的点 SPS 指令：



SPS 指令的含义如下：

A/M/F = 输出端，状态寄存器及标帜器

关闭 SPS 用软键“关闭 SPS”关闭属于编程点的折合，以及把软键条又转换回标准占用。
由此，SPS 指令不再处于可供使用的状态；

等候 / FB 等候及开行条件

宏 / 子程序 / 宏、子程序、焊钳功能、宏／子程序循环

==> 使用此键，可以调用更进一步的 SPS 指令；

<== 操作此键，可以退回到上一个软键条；

脉冲

脉冲输出端

比较

算术比较

t = STOP

计时器停止

t =

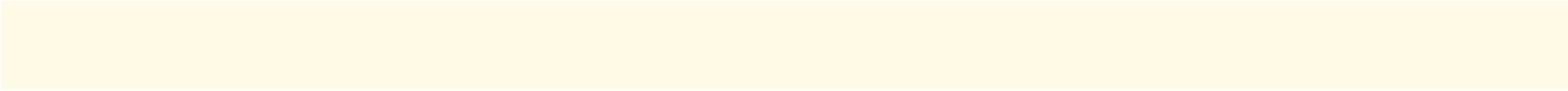
计时器启动

i/bin =

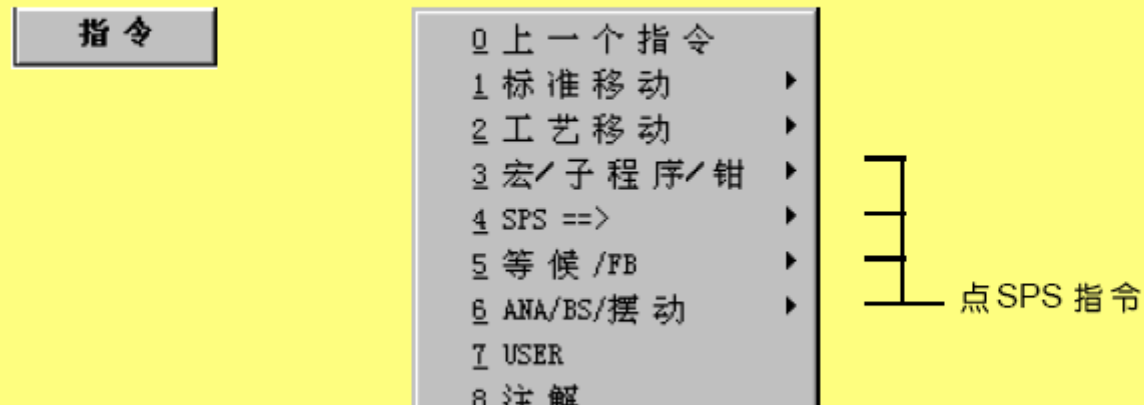
计数器和二进制输出端

ana/BS/Pnd

模拟输出、轨迹开关功能、摆动



同样，您也可以通过菜单来选定 SPS 指令，具体方法如下：按菜单“指令”，然后从它的下属菜单中选择：“宏/子程序/钳”、“SPS==>”、“等候/FB”或“ANA/BS/摆动”。



若是从菜单“指令”中选择点 SPS 指令，而并未打开 SPS 指令的折合，则会出现出错提示“命令不被允许”。

所有注明行号的SPS点指令，将在指定的点执行。通过对SPS触发情况的控制，可以推迟执行一个向定点运行的指令。

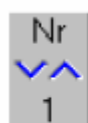
如果未编号的点 SPS 指令处在已编号的指令之前，则在通往程序设定的点的过程中执行未编号的指令。此执行过程不同步于SPS点的触发。例如：“BS”和“FB_Onl”就属于此类命令。

如果未编号的点 SPS 指令处在已编号的指令之后，则也是首先在指定的点执行已编号的指令，然后才可以执行未编号的指令。例如：“子程序”就属于此类命令。

调用相应的功能块，并且填写它的 InLine 表格，由此可以给出所希望的指令。下面是指令“SPSMakro”的 InLine 表格：

SPSMAKRO	1	=		EIN		
----------	---	---	--	-----	--	--

您可以使用如下方法在区域中进行输入：



- 通过操作状态键“高/低”。此键的状态与在激活区域中有可能进行的输入有关；
- 通过使用软键条上的软键。它们的状态也会根据可能进行的输入发生相应的变化；
- 通过在 VKCP 的键盘上，输入选项名称的第一个字母。如果多次按此键，则以相同字母为开始的选项被相继打开。

通过软键“新建 OP”和“删除 OP”，可以新建或者删除 InLine 表格中的附加条件（操作符和操作数）。只有当反显（深蓝底色）处于一个操作符区时，这两个软键才能显示在软键条上。

新的 OP

在操作软键“新建 OP”之后，会出现一个扩展的 InLine 表格，在其中可以添加新的条件。新的操作符总是被添加在反显区之后。下面的例子中，只有当两个输入端分别都出现信号时，才能执行此指令。

扩展的 InLine 表格

SPSMAKRO	1	=		EIN		+		EIN	
----------	---	---	--	-----	--	---	--	-----	--

SPSMAKRO	1	=		EIN		+		E	1
----------	---	---	--	-----	--	---	--	---	---

删除 OP

通过指令“删除 OP”，可以删除呈反显状态的操作符及操作数。



通过按“回车键”，可以关闭 InLine 表格，并且储存已经输入的设置。



而通过“ESC”键，您可以随时退出输入，并且不储存已经输入的内容，同时关闭 InLine 表格。

其它



对于包含一个或多个 SPS 指令的移动指令，可以在其末尾用下列记号注明：

- F**， 当含有 Online 开行条件时；
- P**， 当含有带行号的点 SPS 指令时
（例如：“A1 = 开”、“等候至/时间”、“宏”，等等）。
- U**， 当含有子程序时。

概览

这一节将简要介绍允许的元素，数值范围，操作符及一般的术语定义。

简单元素、操作符、优先级

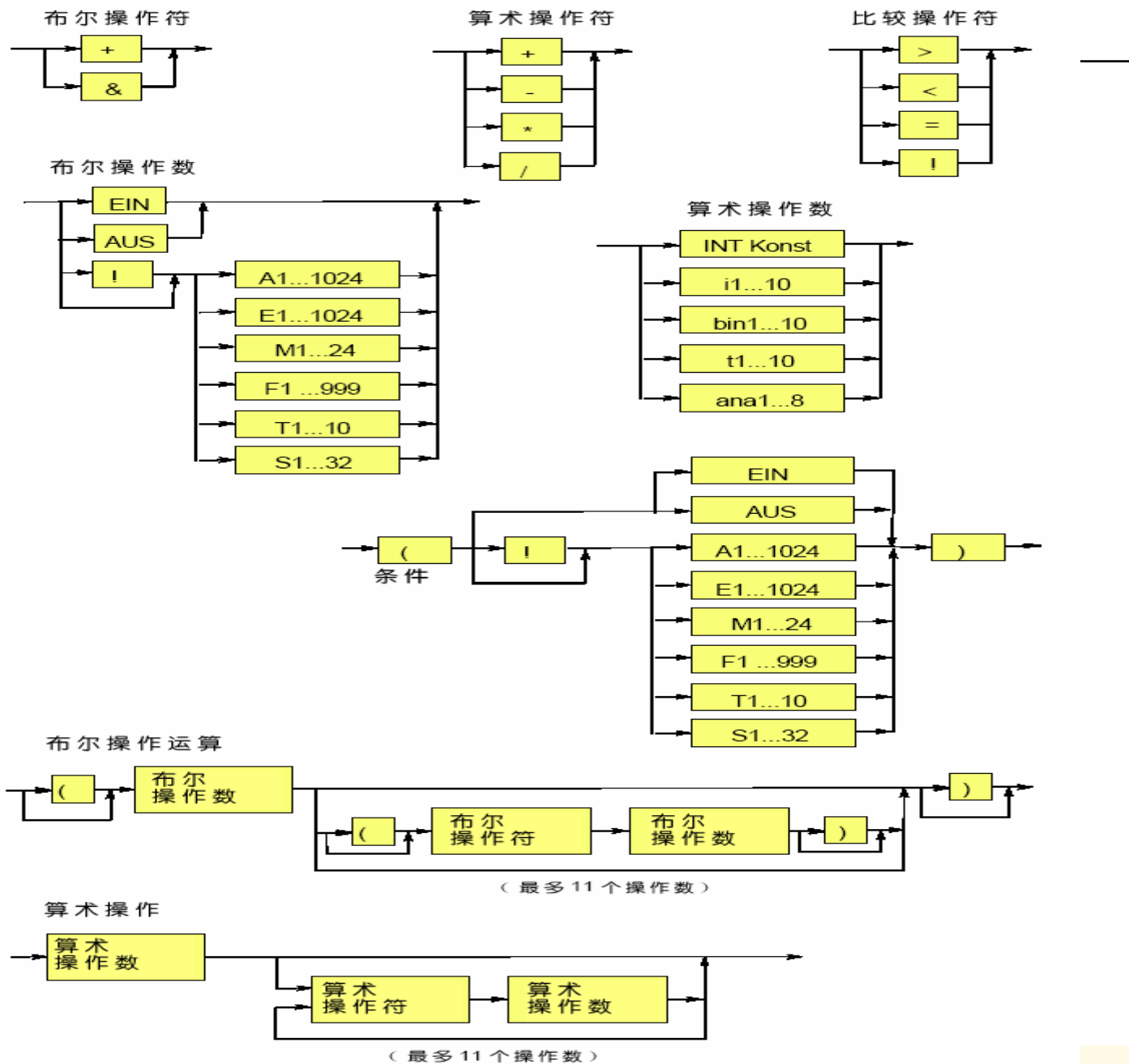
简单元素		
E1 ... E1024	输入端	布尔操作数
A1 ... A1024	输出端	布尔操作数
M1 ... M24	状态寄存器	布尔操作数
F1 ... F999	标帜器	布尔操作数
T1 ... T10	计时器的标帜器	布尔操作数
S1 ... S32	传感器的标帜器	布尔操作数
num	数字	算术操作数
i1 ... i10	整数计数器	算术操作数
bin1 ... bin10	二进制输出	算术操作数
t1 ... t10	计时器	算术操作数
ana1 ... ana8	模拟输出端	算术操作数

操作符		
+	或	布尔操作符
&	与	布尔操作符
(	括号开始	操作符
)	括号结束	操作符
+	加	算术操作符
-	减	算术操作符
*	乘	算术操作符
/	除	算术操作符
>	大于	比较操作符
<	小于	比较操作符
=	等于	比较操作符
!	非	比较操作符
开及 ?		布尔常量
关		布尔常量

操作符的优先级

!	非	比较操作符
*、/	乘, 除	算术操作符
+、-	加, 减	算术操作符
&	与	布尔操作符
+	或	布尔操作符
==、<、>、<=、>=、<>	等于、小于、大于、不等于	比较操作符

术语定义



算术操作数 num、i、bin、t 及 ana 的数值范围

操作数	值	输出
num	-99999 ... +99999	数字 例如：“12”
i1 ... i10	-9999 ... +9999	整数 例如：“15”
bin1 ... bin10	-9999 ... +9999	二进制数 例如： “10011010”
t1 ... t10	-99999 ... +99999	单位为 [1/10s] 例如：“300”
ana1 ... ana8	-9999 ... +9999	单位为 [mV] 例如：“3500”

num、i、t 及 ana 的比较

num = 5

F1 = (num = i1) => “ TRUE ”

i1 = 5

num = 100

F1 = (num = t1) => “ FALSE ”

num = “ 1000 ”

T1 = 1000 ms = 10 * 1/10s

t1 = “ 10 ” 1/10s

n = 1000

F1 = (num = ana1) => “ TRUE ”

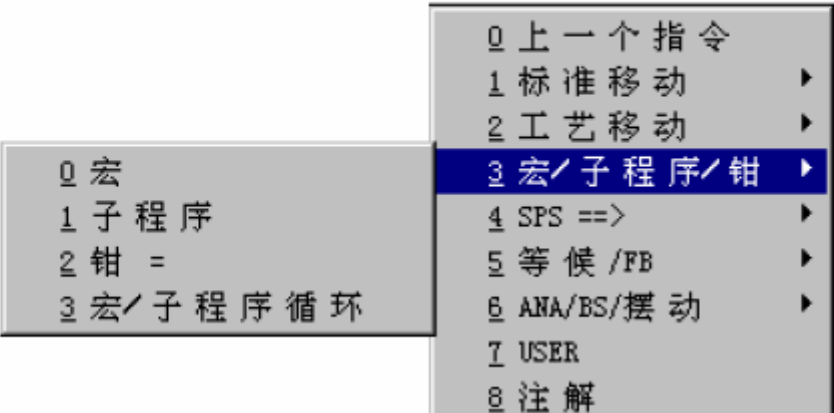
ana1 = 1000 mV

选定

宏/子程序/焊钳的功能只有在选定程序和打开折合的情形下才可使用

调用宏、子程序及钳命令

指令 按菜单键“指令”，并选择菜单“宏/子程序/钳”，然后选出所希望的菜单项：



或者

宏 / 子程序 / 操作软键“宏/子程序/钳”，然后在软键条上选出所希望的选项：



老

选定软键“宏”之后，会出现如下的 InLine 表格：

SPSMAKRO 1 = EIN

SPSMAKRO 1 = E 32

区域描述	功能	数值范围
SPSMakro	宏指令的号码	0...99
=	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	开、关、E、A、M、F、T、S
32	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)
*1) => 当操作数类型为“开”、“关”时，不显示		



举例

SPSMAKRO 1 = E 32

当输入端 32 有信号时，即：此输入端的值为“逻辑1”或者“TRUE”的时候，SPS 宏指令 1 将起动作。

MakroSaw（语句选择宏）

当操作者中止惯常的程序运行时，例如：需要改动此工作程序时，则语句选择宏处于待用状态。操作者只需按软键“语句选择”，就可以运行宏指令“MakroSaw”。在这个宏指令中，临界的输出端可以转化为非临界状态。例如，在机器人及装置的闭锁机构中，此功能特别重要。

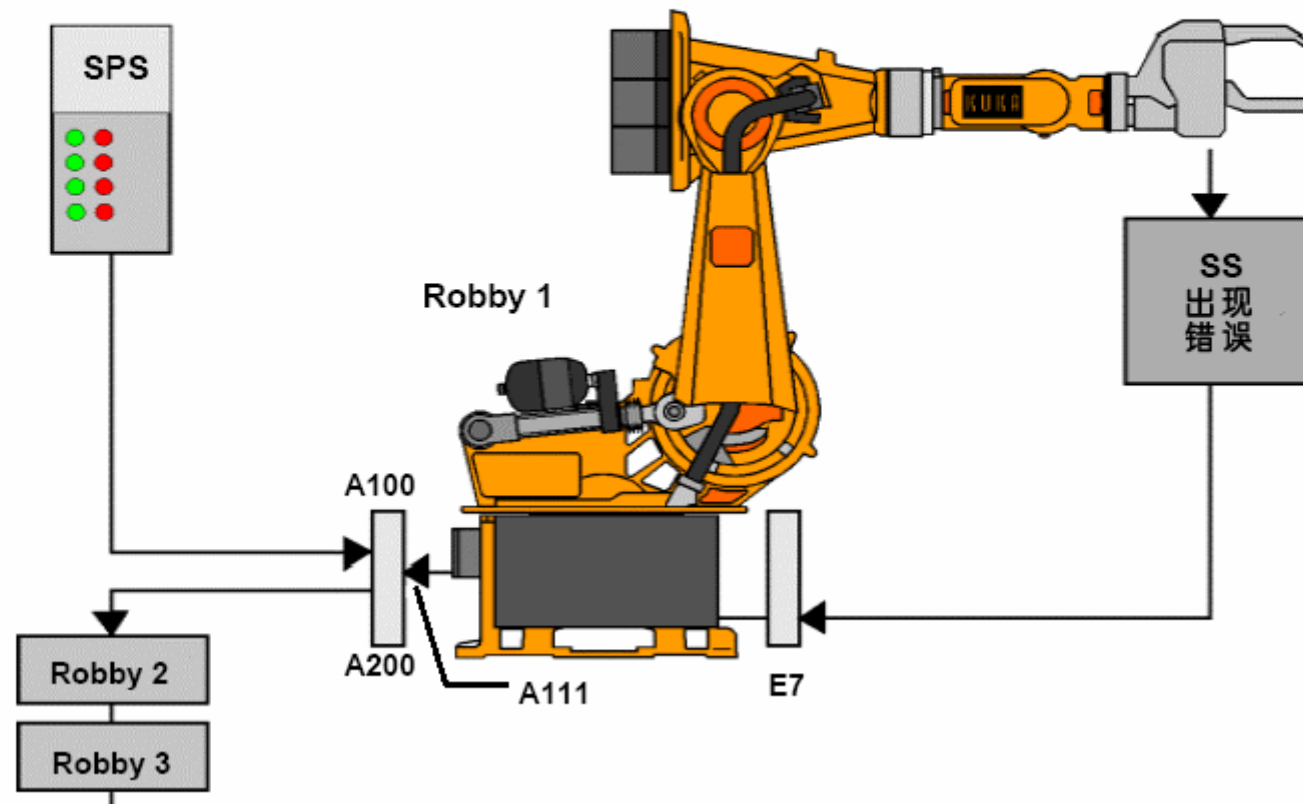
为此，必须首先将工作程序选定，或者使其处于编辑模式。在将编辑光标置于所希望的行之后，可以通过按软键“语句选择”来执行宏指令“MakroSaw”，并且黄色的程序指针也将被置于同一行。

以编辑文件“Makrosaw.src”的方法可以改变语句选择宏本身。在语句选择宏中，只能对输出端或标帜器进行布尔赋值。最多只允许 30 条指令。

MakroSPS



宏指令“MakroSPS”动态运行在背景中，它可以在出现错误时，起动一个设定的操作。以下图为例，在焊接控制装置发出出错提示后，输入端E7被置位，也就是说，它取值逻辑1或者TRUE。这个宏指令同时也将输出端111置位，SPS可以处理该输出端。



在标准情况下，宏指令“MakroSPS”位于目录“C:\KRC\Roboter\KRC\R1”中，其文件名称为“Makrosps.src”。

对该宏指令将动态地（即持续地）询问，它的性能如同独立SPS，在使用时必须进行参数设定。请您注意建立MakroSPS的有关规定。

在“MakroSPS”中不允许出现PTP、LIN、CIRC、KLIN及KCIRC指令。否则会出现出错提示：“指令在此宏中非法”。

同样，某些特定的点SPS指令也不被允许。尤其是下列指令：“宏/子程序/钳”、“等候/FB”、“SPS==>脉冲”和“ANA/BS/摆动”等。如果试图输入这些指令，会出现出错提示：“指令在MAKROSPS中非法”。

子程序（调用子程序）

使用子程序可以使工作程序及序列条理清楚，并且可以降低工作程序运行的时间消耗。子程序可以被任意多次调用。

子程序（调用子程序）

子程序 选定软键“子程序”之后，会出现如下的 InLine 表格：

UP **1** = EIN

UP **1** = E **1**

区域描述	功能	数值范围
UP（子程序）	子程序的号码	1...99
=	操作符	_，（
	操作符 *1)	_，！
EIN,E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E（1...1024）、A（1...1024）、 M（1...24）、F（1...999）、T（1...10）、 S（1...32）
	操作符	_，）
*1) => 当操作数类型为“开”、“关”时，不显示		



举例

UP **1** = E **1**

当输入端1收到信号时（即此输入端的值为“逻辑1”或者“TRUE”时），启动子程序1。

钳（钳指令）

通过此命令，输出端执行钳触发的指定功能。

钳（钳指令）

—— 钳 = 选定软键“钳”后，将打开如下的 InLine 表格：

ZANGE 1 AUF = EIN

ZANGE 1 AUF = E 1

区域描述	功能	数值范围
ZANGE (焊钳)	钳的号码	1...16
AUF (打开)	状态	打开、关闭
=	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)
*1) => 当操作数类型为“开”、“关”时，不显示		



举例

ZANGE 1 AUF = ! E 25 & E 26

当在输入端 25 没有信号（即它的值为“逻辑0”或“FALSE”），而输入端 26 有信号（即取值“逻辑1”或“TRUE”）时，钳1将打开。

宏/子程序运行/运行命令、

REPEAT N= STOP=

REPEAT N= STOP=

区域描述	功能	数值范围
REPEAT	种类	子程序、宏指令
1	子程序或者宏指令的号码	子程序 (1...99) 、宏指令 (0...99)
N=	算术操作数 (重复)	num、i、bin、t、ana
1	重复次数	num (1...99999) 、i (1...10) 、 bin (1...10) 、t (1...10) 、ana (1...8)
STOP=	操作符 *1)	_, !
EIN, E	操作数类型	开、关、E、A、M、F
1	操作数的号码 *1)	E (1...1024) 、A (1...1024) 、 M (1...24) 、F (1...999) 、T (1...10) 、 S (1...32)
*1) => 当操作数类型为“开”、“关”时，不显示		

举例:

REPEAT N= STOP=

在上面的例子中，子程序“5”将循环运行 12 次。当然，此循环只能运行到输入端 1 出现信号为止（即“TRUE”或“逻辑1”）。

退出指令的检验每一次都在重新执行子程序之前进行。

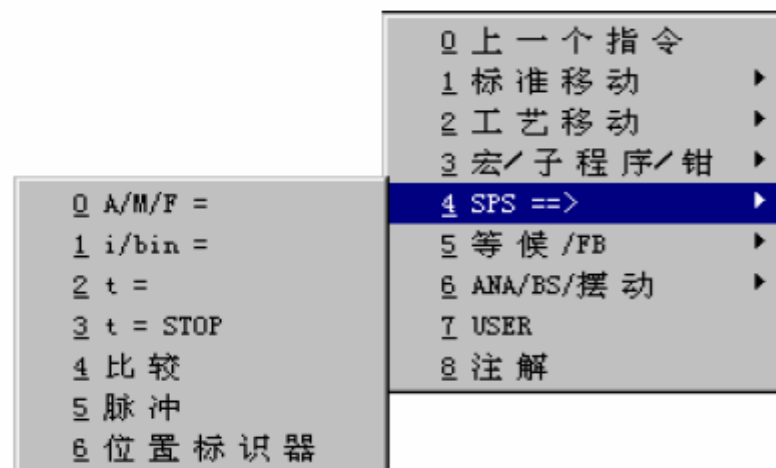
SPS 指令

选定

只有在选定了程序后，才可以使用SPS指令。而后，您可以通过两种方式来选择SPS指令：

指令

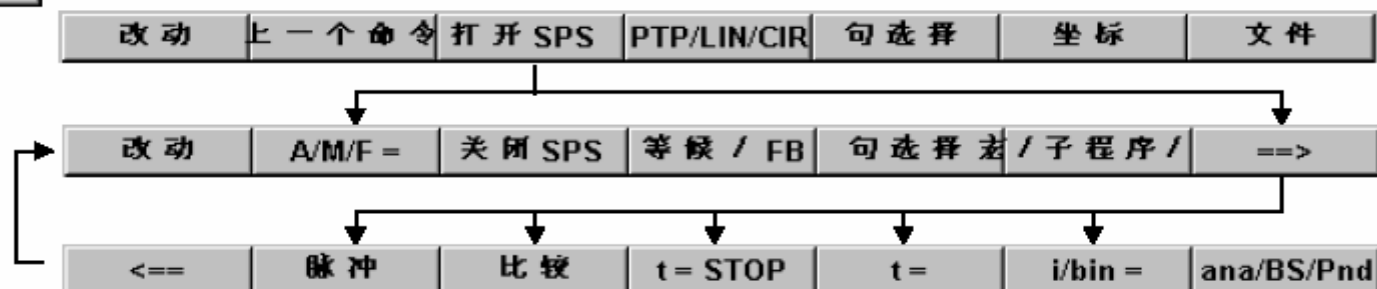
按菜单键“指令”，并选择菜单“SPS==>”，然后从出现的下属菜单中选出所希望的SPS指令：



或者

打开 SPS

依次操作所希望的软键：



A/M/F (输出端 / 状态寄存器 / 标帜器)

A/M/F =

选定此软键后，出现如下的 InLine 表格：

A 1 = EIN

A 1 = E 1

区域描述	功能	数值范围
A	种类	A、M、F
1	号码	A (1...1024)、M (1...24)、F (1...999)
=	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)

*1) => 当操作数类型为“开”、“关”时，不显示



举例

A 3 = E 3 & ! A 7 + A 11

在此例中，输出端 3 在下列情况被置位

- 当输入端 3 及输出端 7 均有信号 (“ TRUE ”) 时，或者
- 当输入端 3 及输出端 11 均有信号 (“ TRUE ”) 时。

i/bin (计数器和二进制输出端)

通过这个 SPS 指令，您可以将整数变量作为计数器来编程，或者将算术表达式的值，按顺序从指定的输出端以二进制形式输出。

此指令可用于统计焊接点的数量，或者把程序号输出至其它仪器或控制器。

i/bin =

选定此软键后，出现如下的 InLine 表格：

i 1 (?) = 1

i 1 (E) - 1

区域描述	功能	数值范围
i	算术操作符	i、bin
1	号码	i (1...10)、bin (1...10)
	操作符 *1)	_、!
?, E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	算术操作数	num、i、bin、t、ana
1	号码	num (1...9999)、i (1...10)、 bin (1...10)、t (1...10)、ana (1...8)
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		



举例

i 4 (M 15) = 1

此处表明：当在状态寄存器 15 上出现信号（“逻辑 1”或“TRUE”）时，整数计数器 4 被置位。

t= (计时器启动)

通过 “ t= ” 这一功能，您可以给出计时器的初始值，并在满足其条件的同时启动计时器。

t=

选定此软键后，出现下列InLine 表格：

t 1 (?) = 1 [1/10 sec]

t 1 (E) = 1 [1/10 sec]

区域描述	功能	数值范围
t	计时器的号码	1...10
	操作符 *1)	_、!
?, E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
=	算术操作数	num、i、bin、t、ana
1	号码	num (1...99999)、i (1...10)、 bin (1...10)、t (1...10)、ana (1...8)
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		



举例

t 5 (!) F 25 = 300 [1/10 sec]

当标帜器 25 的值为“FALSE”或“逻辑0”时，计数器 5 以 30 秒为初始值而启动。它将一直运行，直到开始执行指令：“t = STOP”的时候为止。

t=STOP（计时器停止）

通过“t=STOP”这一功能，您可以编制一个程序，使计时器在满足给定条件的情况下停止。

t = STOP

选定此软键后，出现下列InLine 表格：

t [?] = STOP

t [] [E] [1] = STOP

区域描述	功能	数值范围
t	计时器停止的号码	1...10
	操作符 *1)	_、!
?, E	操作数类型	?. 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		



举例

t [] [E] [1] = STOP

如果在输入端 1 出现信号（“逻辑 1”或“TRUE”）时，计时器 1 停下。

比较（算术比较）

通过“比较”功能，您可以编程序来进行算术比较。比较结果将储存于标帜器中，此标帜器的状况可以随时查询。

比较

选定此软键后，出现下列InLine 表格：

F 1 = 1 > 1

区域描述	功能	数值范围
F	标帜器的号码	1...999
=	算术操作数	num、i、bin、t、ana
1	号码	num (1...9999)、i (1...10)、 bin (1...10)、t (1...10)、ana (1...8)
>	比较操作符	>、<、=、!
	算术操作数	num、i、bin、t、ana
1	号码	num (1...9999)、i (1...10)、 bin (1...10)、t (1...10)、ana (1...8)



举例

F 3 = i 3 > i 2

当整数计数器 3 的值大于整数计数器 2 的值时，标帜器 3 被置位。

脉冲（脉冲输出端）

脉冲输出端的操作执行伴随着移动而进行，它们将按照设定的时间保持其逻辑状态。可以进行非同步的置位及复位，以及对设定的输出端进行后续触发。

脉冲

选定此软键后，出现下列InLine 表格：

Puls A **1** =Pegel: **EIN** Zeit: **1** [1/10Sek]

Puls A **1** =Pegel: **E** Zeit: **1** [1/10Sek]

区域描述	功能	数值范围
Puls A (脉冲)	可置位的脉冲输出	1...1024
	操作符 *1)	_, !
EIN	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
Zeit: 1 (时间)	停止逻辑状态的时间	0...300 ¹ / ₁₀ 秒
*1) => 当操作数类型为“开”、“关”时，不显示		



例如在焊接销子时，希望同步于机器人的移动输送后续的螺栓，则可以使用这项功能。

取决于位置的标帜器 (MakroSPS)

只能通过选择菜单键“指令”，及其下属菜单“SPS==>”，才能进入指令“位置标帜器”。此指令可以用来设置标帜器。当轴到达指定的位置时，可以通过该标帜器来触发某个动作。

通过选定菜单键“位置标帜器”，将打开下列Inline 表格：

F1 = AXIS 1 > 20

区域描述	功能	数值范围
F	标帜器	1...999
=	轴	AXIS_1...6、EXAX_1...6
>	比较操作符	<、>、=、!
	算术操作数	num、i、bin、t、ana
20	与轴相关的实际值， 单位为度	num (1...9999)、i (1...10)、 bin (1...10)、t (1...10)、ana (1...8)

举例

F1 = AXIS 1 > 20

在此例中，一旦轴1达到或超过与轴有关的值20°时，标帜器1被置位。

等候指令和开行指令

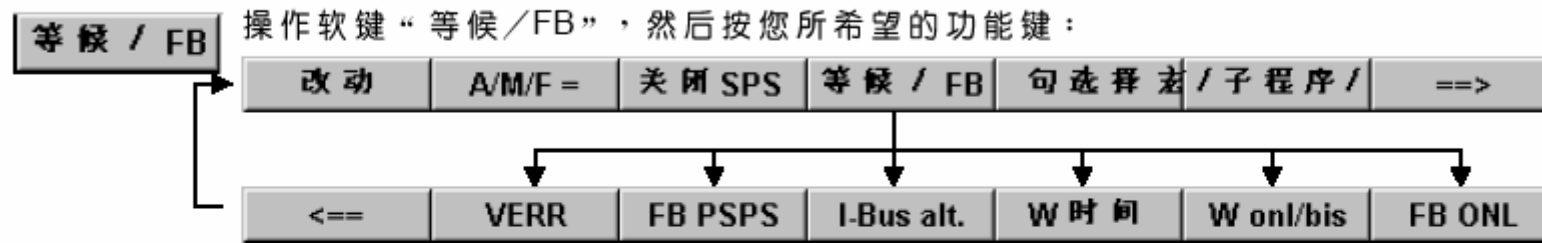
指令

操作菜单键“指令”，并选择菜单“等候/FB”，然后再选出您所需要的下属菜单：

- 0 FB ONL
- 1 等候 onl/bis
- 2 等候时间
- 3 闭锁
- 4 FB PPS
- 5 I-Bus 可替换

- 0 上一个指令
- 1 标准移动 ▶
- 2 工艺移动 ▶
- 3 宏/子程序/钳 ▶
- 4 SPS ==> ▶
- 5 等候 /FB ▶
- 6 ANA/BS/摆动 ▶
- 7 USER
- 8 注解

或者



通过等候指令的执行，可以等候：

- 确定的时间，
- 等候条件或者
- 开行条件(FB)。

此功能可以用于点焊，或者用在与其它机器人及仪器之间的连锁等场合。
该等候指令将在对它编程设置过的点被执行。

FB ONL (开行条件)

机器人只在当为其设定的开行条件被满足时才运行。如果不再满足指定的外部条件，此开行条件也可以使机器人有控制地停下来。

FB ONL

选定软键“FB ONL”后，将打开下列InLine 表格：

FB ONL=

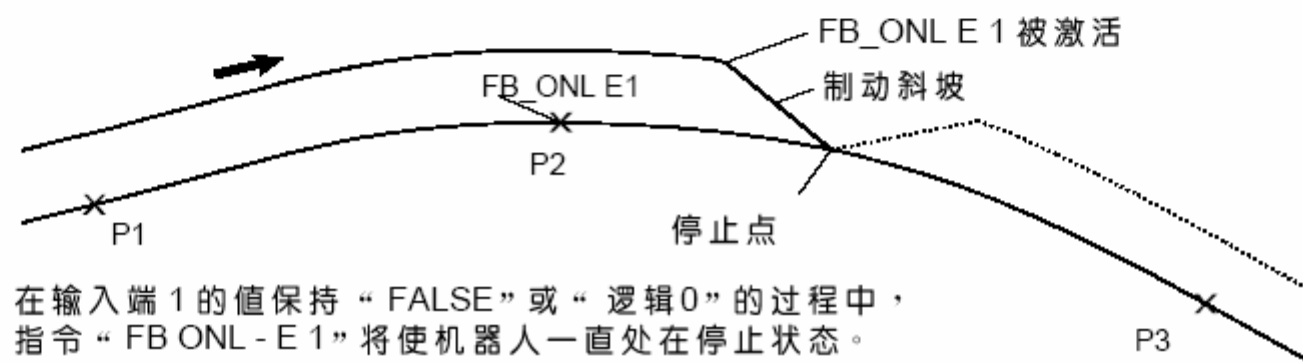
FB ONL=

区域描述	功能	数值范围
FB ONL	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)
*1) => 当操作数类型为“开”、“关”时，不显示		



举例

FB ONL=



W onl/bis（等候条件）

通过这个指令，您可以编程设置等候条件。

W onl/bis

选定软键 “W onl/bis”（等候 onl/至）后，将打开下列 InLine 表格：

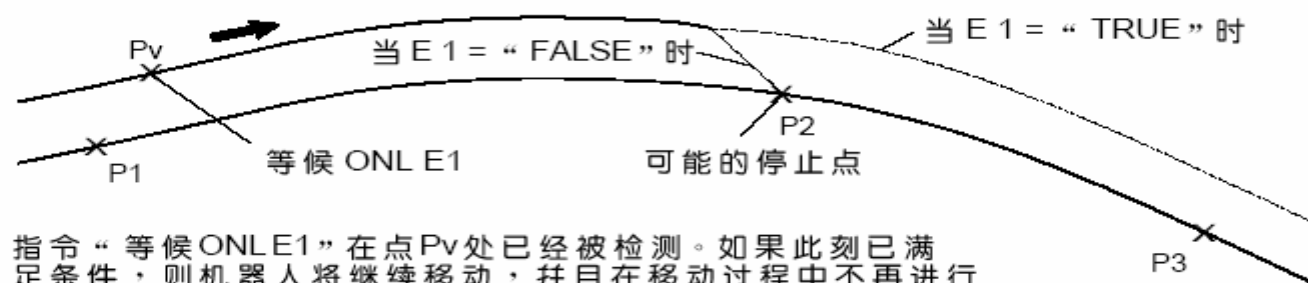
WARTE **ONL** EIN

WARTE **ONL** E 1

区域描述	功能	数值范围
WARTE	种类	onl、bis
	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)
*1) => 当操作数类型为“开”、“关”时，不显示		

**举例 1**

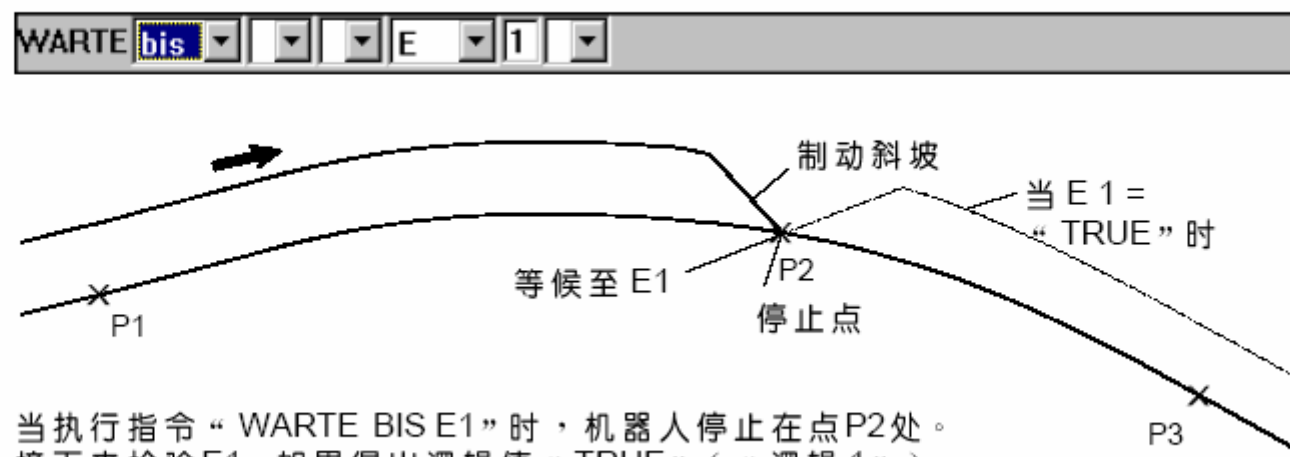
WARTE **ONL** E 1



指令“等候 ONL E1”在点 Pv 处已经被检测。如果此刻已满足条件，则机器人将继续移动，并且在移动过程中不再进行检测。否则，机器人将在点 P2 处停止。



举例 2



当执行指令“WARTE BIS E1”时，机器人停止在点P2处。接下来检验E1。如果得出逻辑值“TRUE”（“逻辑1”），则机器人继续运动。

W 时间（确定的时间）

通过此功能，您可以编程设定确定的等候时间。

（注：此功能需要软件包“W”支持。）

W 时间

选定软键“W 时间”后，将打开下列 InLine 表格：

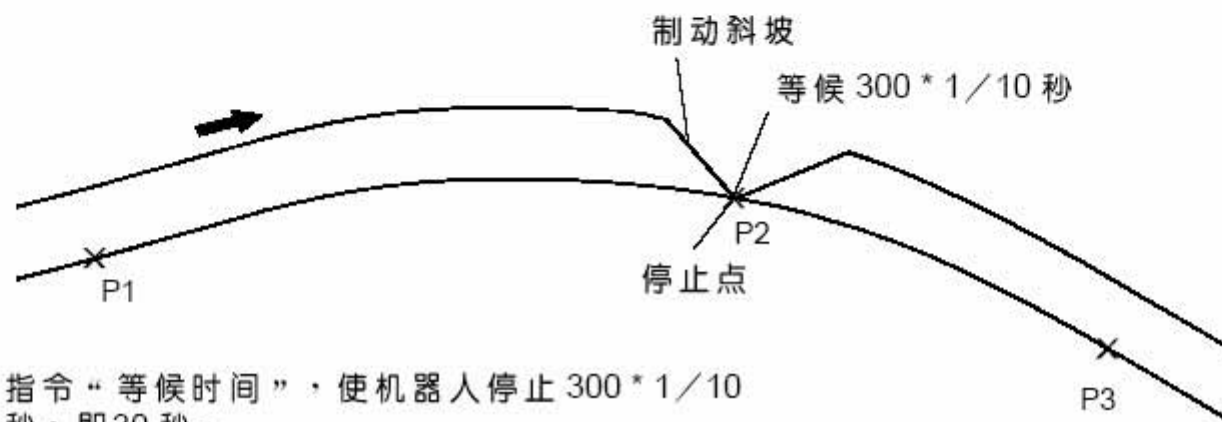
WARTEZEIT  1 [1/10Sek]

区域描述	功能	数值范围
Wartezeit (等候时间)	算术操作数	num、i、bin、t、ana
	时间	$\frac{1}{10}$ 秒



举例

WARTEZEIT  300 [1/10Sek]



指令“等候时间”，使机器人停止 $300 * \frac{1}{10}$ 秒，即 30 秒。

VERR (闭锁)

多个机器人系统的工作空间重叠的时候，可以使用闭锁命令。

VERR

在按软键“VERR”（闭锁）后，将打开一个用来输入参数的 InLine 表格，这些参数是执行闭锁命令所必须的：

VERR. **1** : ROB : EIN ▾ Werte bis E **5** dann A **7** =AUS Prio: **1**



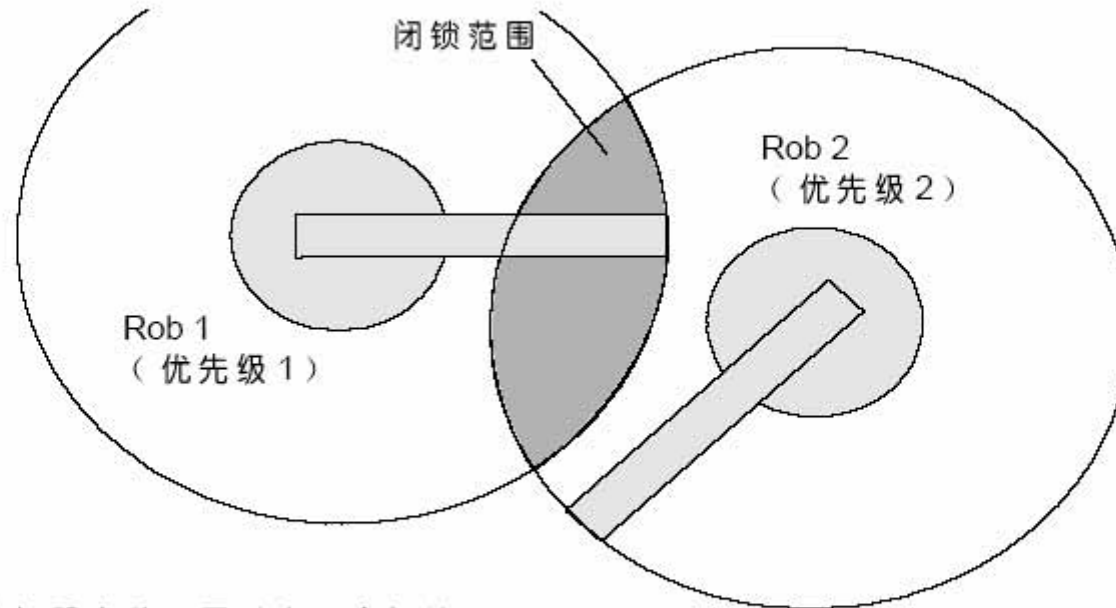
闭锁号码相同的闭锁命令不允许发生嵌套！

区域描述	功能	数值范围
VERR. (闭锁)	闭锁号码	1...8
ROB	机器人名称	最多 8 位字符
EIN	操作数类型	开、关
Werte bis E	输入端号码	E (1...1024)
dann A	输出端号码	A (1...1024)
Prio	优先级	1、2



举例

VERR. **1** : ROB 2 : EIN Warte bis E **1** dann A **1** =AUS Prio: **2**



如果这两个机器人装置同时向一个闭锁区域内移动，则 Rob 2 被闭锁，因为它的优先级是 2。

当在 Rob 2 的输入端 1 处，出现信号“TRUE”或“逻辑 1”时，闭锁装置则被激活。接着在输出端 1 处给出信号“FALSE”或“逻辑 0”。

6 **FB PSPS**（开行条件）

机器人只在当为其设定的开行条件被满足时才运行。如果不再满足指定的外部条件，此开行条件也可以使机器人有控制地停下来。

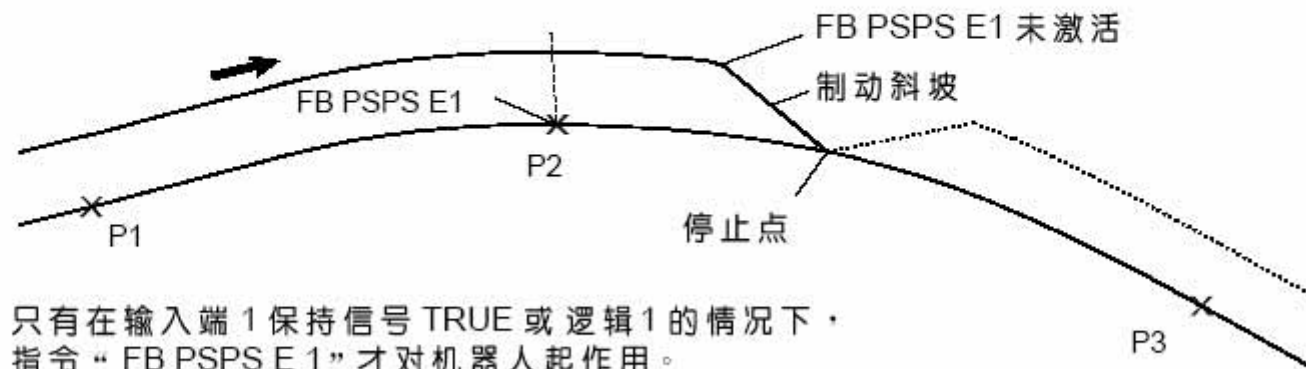
例如，此功能可用闭锁机器人，但也可以在出现故障时使机器人停止。

FB PSPS 功能在移动过程中被检测。



举例

FB PSPS =



只有在输入端 1 保持信号 TRUE 或 逻辑 1 的情况下，指令“FB PSPS E 1”才对机器人起作用。

I-总线的抉择

“I-总线的抉择”功能可以暂时关断Interbus区段或者它的用户。例如当机器人带有自动调换装置，并且某个Interbus用户处于欲调换的工具上时，必须用到此指令。否则，在工具调换时，可能会发生总线错误。

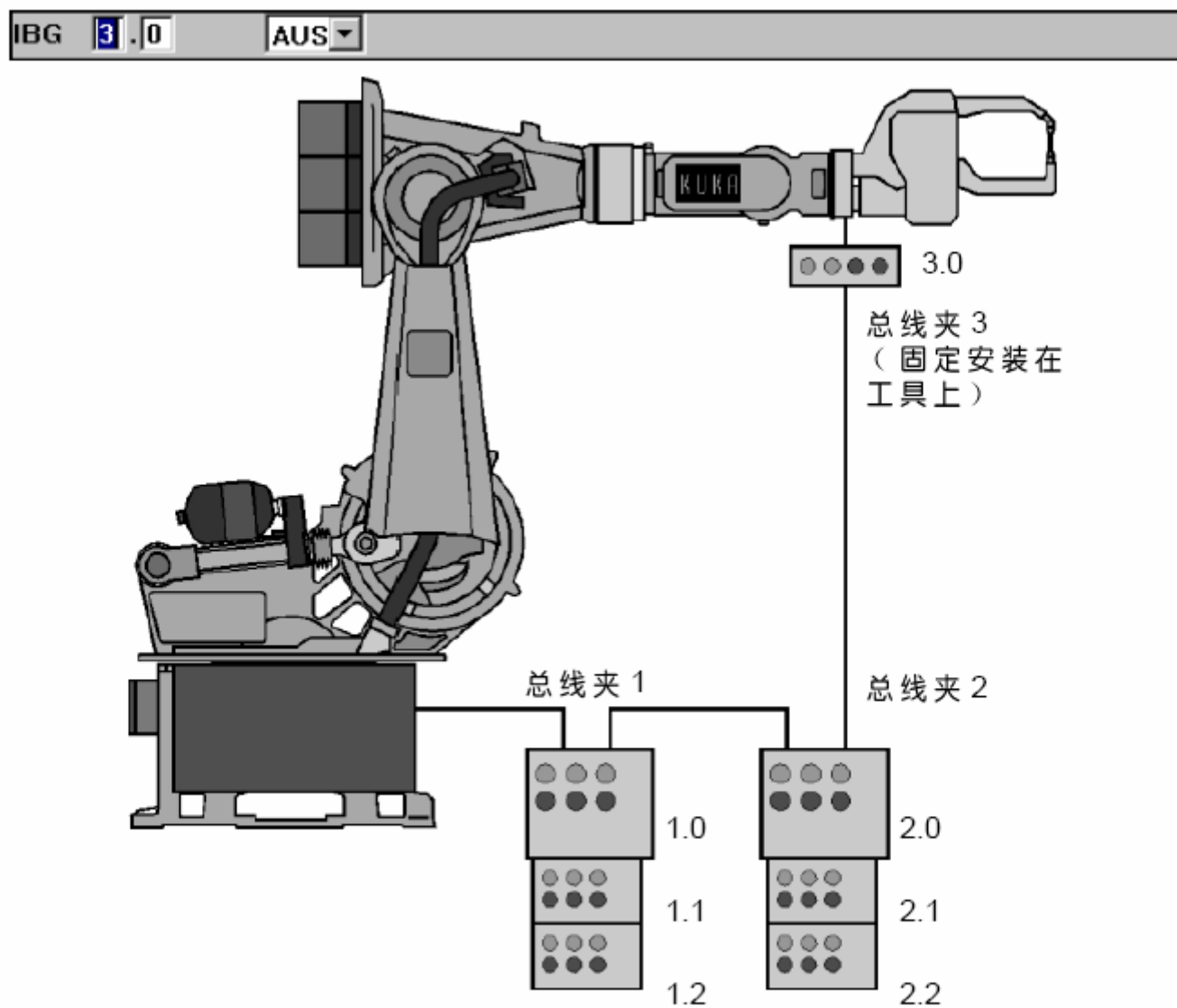
IBG .

IBG .

区域描述	功能	数值范围
IBG	I 总线区段号	1...256
.	I 总线用户号	1...512
	操作符 *1)	_, !
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
*1) => 当操作数类型为“开”、“关”时，不显示		



举例



Interbus 区段 3.0 将暂时不起作用，以便进行工具调换。否则在调换工具时，分开 Interbus 会造成总线错误。

模拟输出、轨迹开关功能、摆动

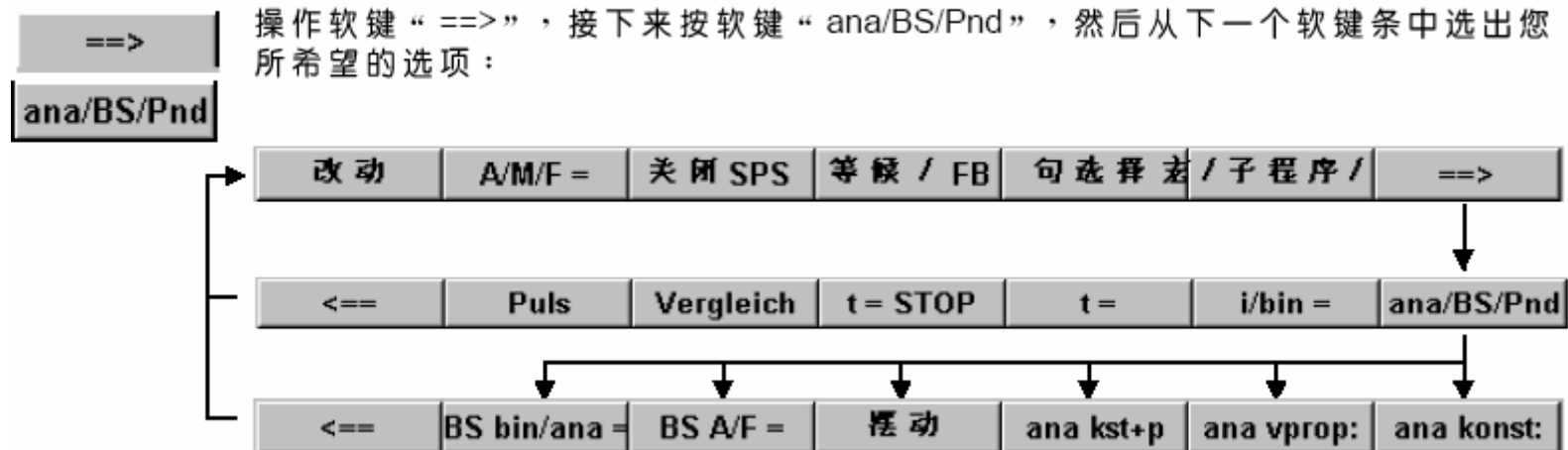
指令

按菜单键“指令”，并选择菜单“ANA/BS/摆动”，然后选出您所希望的菜单项：

0 ana konst:
1 ana vprop:
2 ana ksttp
3 摆动
4 BS A/F =
5 BS bin/ana =

0 上一个指令
1 标准移动 ▶
2 工艺移动 ▶
3 宏/子程序/钳 ▶
4 SPS ==> ▶
5 等候 /FB ▶
6 ANA/BS/摆动 ▶
7 USER
8 注解

或者



模拟输出

ana/BS/Pnd

在按了软键“ana/BS/Pnd”之后，有下面三个选项可以供您选择：



通过模拟输出，您可以通过控制器输出模拟电压。此模拟电压可以为：

- 时间恒定，
- 与速度相关或者
- 与摆动偏移量相关。

此功能可以用于例如：粘结剂的剂量控制、气体保护焊的参数设定或者点焊钳上的压力级别控制等。

最多可以同时控制8条不同的模拟通道，这些也通道可以设定为与某些条件相关。

ana konst : (恒定的模拟电压)

通过这一功能，您可以输出一个在一段时间内恒定的模拟电压。

ana konst:

选定软键 “ ana konst : ” 后，将打开下列 InLine 表格：

ana 1 () = KONST: [mV]

ana 1 () = KONST: [mV]

区域描述	功能	数值范围
ana	模拟输出	1...8
	操作符 *1)	_, !
EIN, E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
KONST:	算术操作数	num、i、bin、t、ana
3500	输入端电压	-10000 ... +10000 [mV]
*1) => 当操作数类型为 “ ? ”、“开”、“关”时，不显示		



举例

ana 3 (21) = KONST: [mV]

当状态寄存器 21 没有置位（ “ 逻辑0” 或 “ FALSE ” ）时，在模拟输出端 3 上的输出电压为 10000 mV。

ana vprop : (同速度成正比的模拟电压)

使用此功能可以实现与速度相关的模拟输出。

ana vprop:

选定软键“ana vprop:”后，将打开下列InLine 表格：

ana 1 [E IN] = Vprop: [] 1 [mm/Sek] Offs: [] 3500 [mV]

ana 1 [] [E] [] = Vprop: [] 1 [mm/Sek] Offs: [] 3500 [mV]

延迟

当“ana”区为反显状态时，软键“延迟”处于可供支配的状态。通过此功能，可补偿受过程制约的死区时间，例如在涂抹粘结剂时。延迟时间可以为正，也可以为负。

区域描述	功能	数值范围
ana	模拟输出端	1...8
	操作符 *1)	_, !
开、E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
Vprop:	种类	num、i、bin、t、ana
	速度	-9999 ... +9999 [mm/Sek]
Offs	算术操作数	num、i、bin、t、ana
	输入端电压	-10000 ... +10000 [mV]
延迟	持续时间 *2)	-576 ... +576 [ms]
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		
*2) => 只有当软键“延迟”被按之后，才可以使用此功能		



在Vprop ($V_{比例}$) 区中，填入以[mm/s]为单位的的速度值，此速度值相应于最大的可能输出电压 (10000 mV)。

同速度成正比的模拟输出，只可以使用通道 1 至 4。

带有延迟功能处于激活状态的模拟输出是可以逼近的。

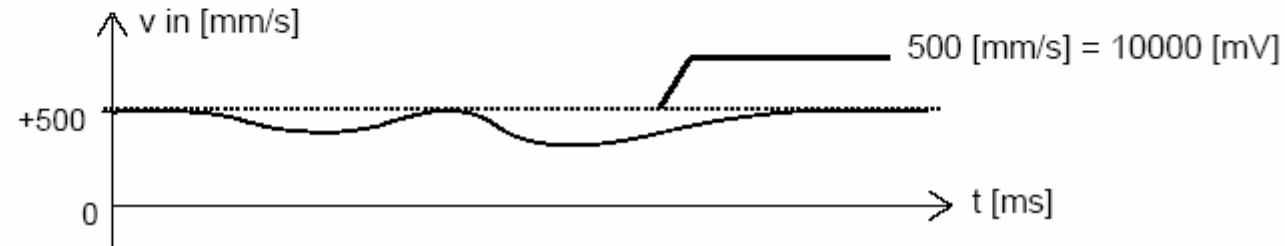


举例 1 (模拟输出)

ana4 [! E]=Vprop: [mm/Sek] Offs: [mV]

同速度成正比的模拟输出，只在当输入端 16 置 0 (即：“FALSE”或者“逻辑 0”)的情况下，才被激活。它将一直处于激活状态，直到它被恒定模拟输出关断为止。此处的模拟输出量同当前的机器人的 TCP 速度成正比。

同速度成正比的模拟运动只在轨迹移动的情况下才有可能，而在点到点的移动过程中则不可能。



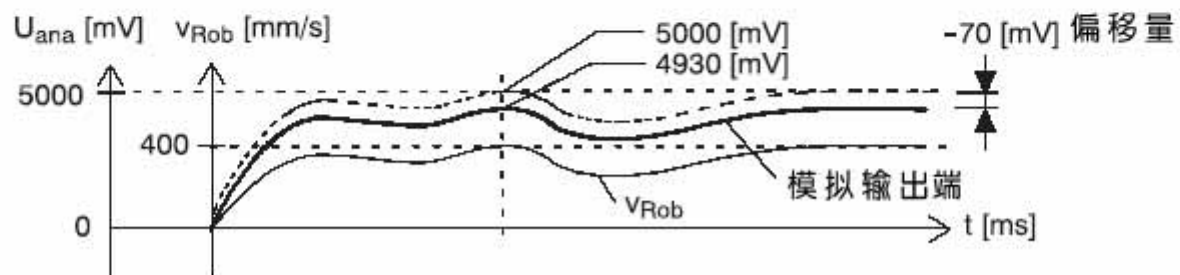


举例 2 (偏移量)

当机器人速度为 400 mm/s 时，欲最佳地涂抹粘结剂需要 5000 mV 的模拟电压。试验表明，只有在机器人的模拟电压为 -70 mV 时，粘结控制装置的内部模拟值才为 0 V。

ana 4 [!] E 16]=Vprop: 800 [mm/Sek] Offs: -70 [mV]

在此例中，因为当速度为 400 mm/s 时需要 5000 mV，所以机器人必须以 800 mm/s 的速度移动，才有可能达到 10000 mV 的最大值。此值将被记录在 InLine 表格中。

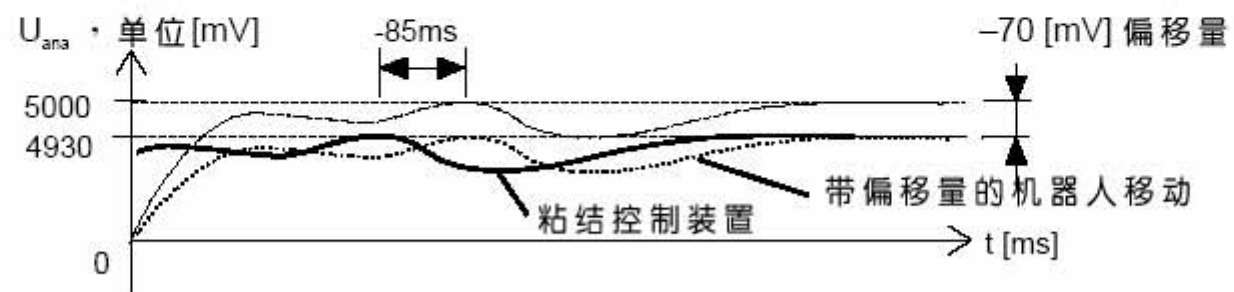




举例 3 (模拟延迟)

ana 4 [EIN] =Vprop: 800 [mm/Sek] Offs: -70 [mV] Delay: 85 [ms]

在此例中，粘结控制装置先于带相应偏移量的机器人85ms起动。





如果机器人的移动速度值大于在区域“Vprop”和“偏移量”中的数值之和，则在提示窗中会给出相应的信息提示。在这种情况下机器人不会被停止。



“Vprop”表示机器人的运行速度，以此速度运行时，偏移量为0的模拟输出可达到的最大值为 10000 mV。

ana kst+p : (恒定的电压及摆动偏移)

通过此功能，可以输出一个恒定的电压，及一个与摆动位置相关的电压。

ana kst+p

选定软键“ana kst+p”之后，会出现下列InLine 表格：

ana 1 [EIN] = KST+P: [] 750 [mV] Pendel: [] 3500 [mV]

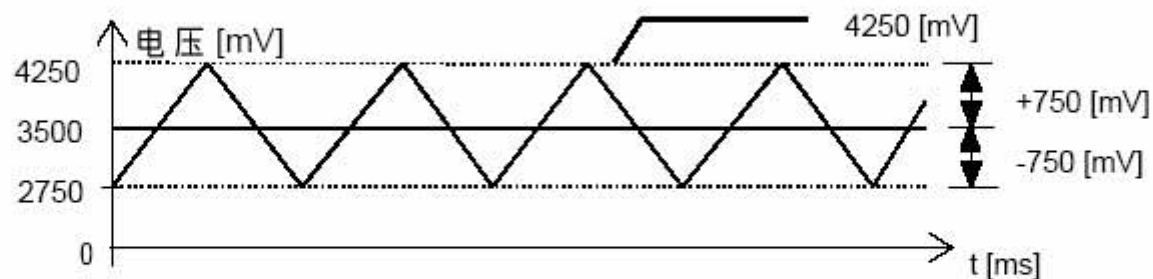
ana 1 [] [E] 1 = KST+P: [] 750 [mV] Pendel: [] 3500 [mV]

区域描述	功能	数值范围
ana	模拟输出端	1...8
	操作符 *1)	_, !
Ein, E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
KST+P	算术操作数	num、i、bin、t、ana
	电压	-10000 ... +10000 [mV]
Pendel: (摆动)	算术操作数	num、i、bin、t、ana
	输入端电压	-10000 ... +10000 [mV]
*1) => 当操作数类型为“？”、“开”、“关”时，不显示		



举例（摆型：“三角形”）

ana1 [E 1] = KST+P: 3500 [mV] Pendel: 750 [mV]



只有当输入端1的值为“TRUE”或者“逻辑1”时，模拟输出才被激活。此输出电压介于 2750 mV 至 4250 mV 之间。它的大小与摆型的瞬时偏转有关。



与摆动相关的模拟输出，只可以使用通道 1 至 4。

BS 轨迹开关功能

ana/BS/Pnd

通过按软键“ana/BS/Pnd”，您可以打开轨迹开关功能。



利用轨迹开关功能，您可以离下一个空间点一定间距，或者在到达下一个空间点之前的某个时刻进行

- 二进制/模拟输出，或者
- 对输出端及标帜器的置位。

这些功能可用于粘结等应用场合。

以下 4 种移动指令可以用于轨迹开关功能：

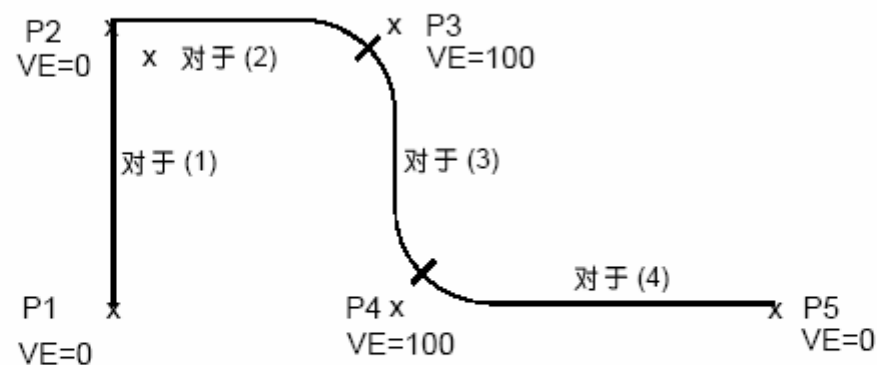
段	停止	VE
对于 (1)	精确定位 -> 精确定位	0 -> 0 *1)
对于 (2)	精确定位 -> 轨迹逼近点	0 -> 100 *1)
对于 (3)	轨迹逼近点 -> 轨迹逼近点	100 -> 100 *1)
对于 (4)	轨迹逼近点 -> 精确定位	100 -> 0 *1)

*1) 第二个VE值 将填入每个相应的 移动指令的 InLine 表格

轨迹开关功能的 InLine 表格中有一选项为“开关点：”。它的第一个值表示与触发点之间的距离，而第二个值表示一个附加的延迟时间，它可以为正，也可以为负。延迟时间在粘结应用场合特别重要，因为用它可以补偿粘结控制的死区时间。



做 PTP 移动时的轨迹开关功能



在做PTP移动的时候，选项“开关点***[mm]”处的值目前总为0。这表明，在“精确定位”的轨迹开关功能情况下，触发总是针对目标点而言（Distance=1），而对于“逼近”而言，起决定作用的是逼近区域的中心。

用下面两个例子可以做更进一步的阐明：

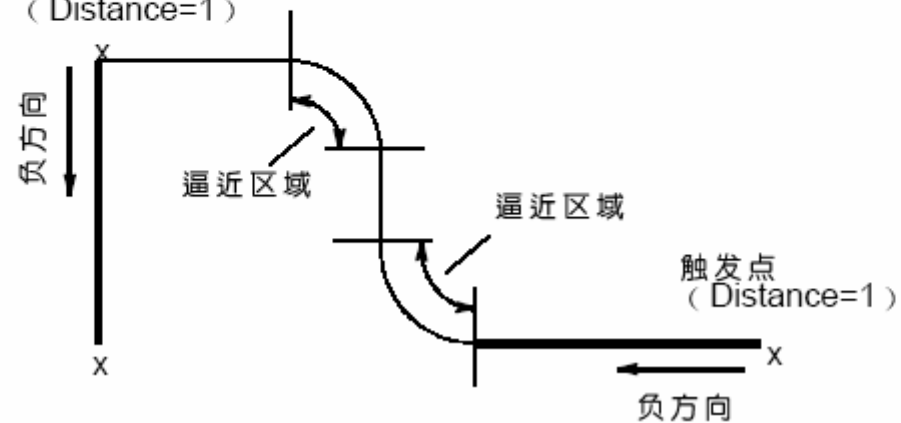


精确定位

在精确定位（触发点）的情况下，触发时间总是作用在触发点的负方向。其极限为上一个逼近点或逼近区域的终止处。

触发点

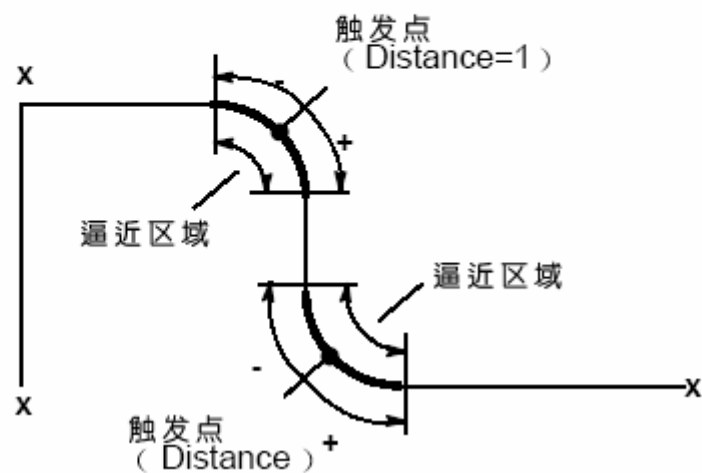
(Distance=1)



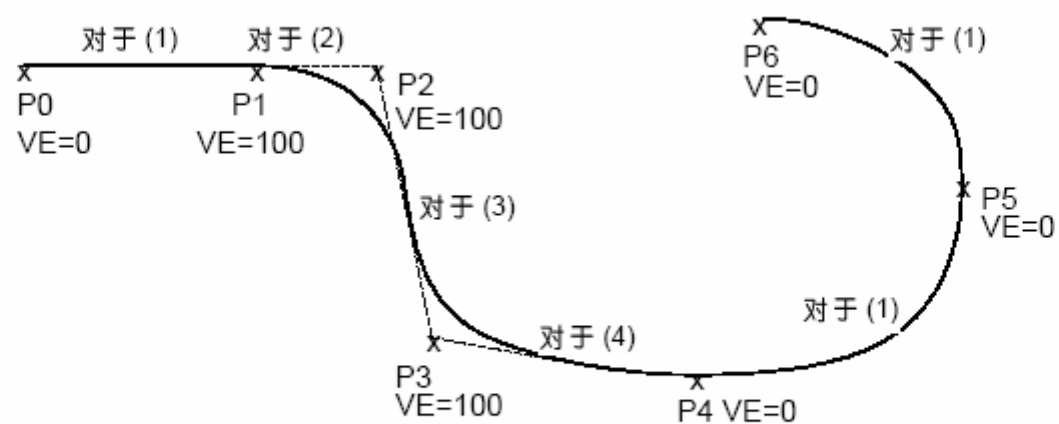


逼近

在逼近的情况下，触发点位于逼近区域的中心。触发时间可以为负值，也可以为正值。其极限为该逼近区域的开始及终止处。



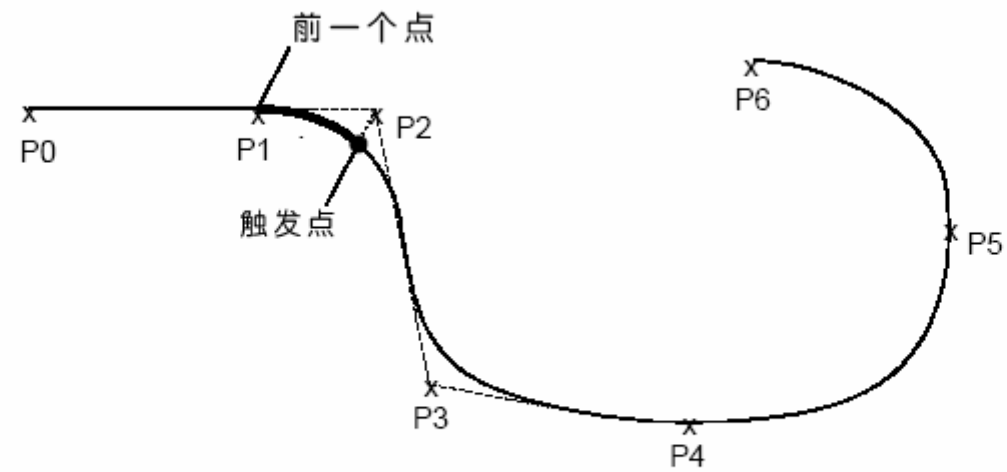
做轨迹移动（LIN、CIR、KLIN、KCIR）时的轨迹开关功能



与做PTP移动时相反，做轨迹移动时，轨迹触发点可以推移到下一个精确定位点。
下面的例子可以做更进一步的阐明：



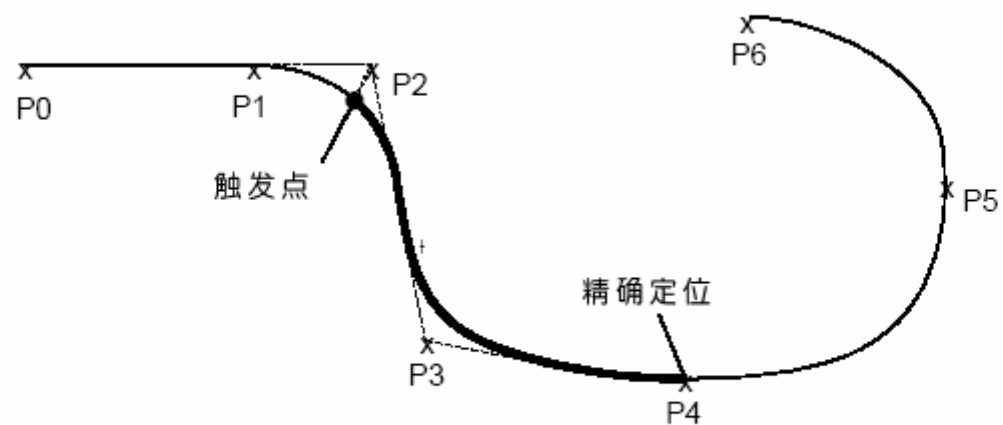
负方向推移



轨迹移动的开关点可以从程序编定的点（触发点）做负方向推移，直到回到上一个点。它也相应于 InLine 表格中的“Schaltpkt:”（开关点）区中的负值。



正方向推移



相反，开关点在正方向可以推移越过移动指令边界。此推移的极限是下一个精确定位的点，或者是 PTP 移动指令。

BS A/F (输出端)

BS A/F =

选定软键“BS A/F =”后，将打开下列InLine 表格：

BS A 1 (EIN) = E Schaltpkt: 1 [mm]: 1 [ms]

BS A 1 (E 1) = E 1 Schaltpkt: 1 [mm]: 1 [ms]

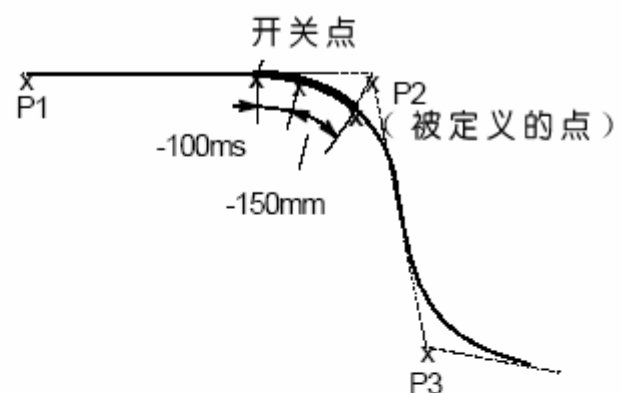
区域描述	功能	数值范围
BS	种类	A、F
1	号码	A (1...1024)、F (1...999)
	操作符 *1)	_、!
EIN, E	操作数类型	?、开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
=	操作符	_、(
	操作符 *1)	_、!
EIN, E	操作数类型	?、开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
	操作符	_、)
Schaltpkt: (开关点)	到触发点的距离	-9999 ... +9999 [mm]
	延迟时间	-9999 ... +9999 [ms]
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		



举例

BS A 10 (EIN) = E 5 Schaltpkt: -150 [mm]: -100 [ms]

输入端5的值被赋予输出端10。真正的开关点将提前100ms，在到达定义点前的150 mm 处被触发。



进行 PTP 移动时，开关点的值必须为零。



在 PTP 指令中，开关点只可能直接重合于被定义的点。如果仍然需要补偿死区时间（例如在粘结工艺），则只可能通过延迟时间 [ms] 来完成。

BS bin/ana (二进制 / 模拟输出)

BS bin/ana=

选定软键“BS bin/ana=”之后，会出现下列InLine 表格：

BS bin [1] [EIN] [=] [3500] Schaltpkt: 0 [mm]: 0 [ms]

BS bin [1] [E] [1] [=] [3500] Schaltpkt: 0 [mm]: 0 [ms]

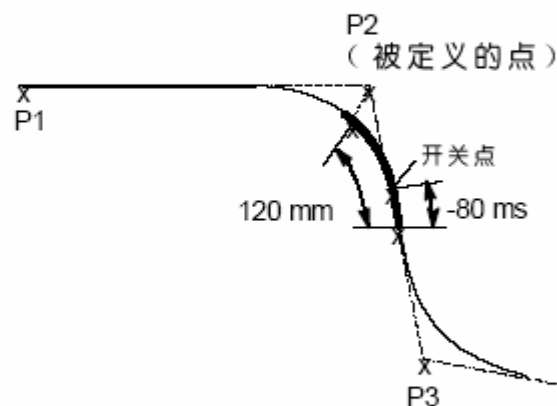
区域描述	功能	数值范围
BS	种类	bin (作用于二进制输出)、 ana (作用于恒定的模拟输出)、 ana_vprop (作用于同速度成正比的模拟输出)、 ana_offs (作用于同速度成正比的模拟输出)
1	号码	bin (1...8)、ana (1...10)、 ana_vprop (1...8)、ana_offs (1...8)
	操作符 *1)	_, !
EIN, E	操作数类型	?, 开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、 M (1...24)、F (1...999)、T (1...10)、 S (1...32)
=	算术操作数	num、i、bin、t、ana
		-9999 ... +9999 [mm]
Schaltpkt: (开关点)	到触发点的距离	-9999 ... +9999 [mm]
	延迟时间	-9999 ... +9999 [ms]
*1) => 当操作数类型为“?”、“开”、“关”时，不显示		



举例

BS **ana** 3 (**E** 1) - 250 [mV] Schaltpkt: 120 [mm]: -80 [ms]

当输入端 1 的信号为“TRUE”或“逻辑 1”时，模拟输出端 3 上的输出电压为 250 mV。开关点将提前 80 ms，在到达定义点后面的 120 mm 距离之前被触发。

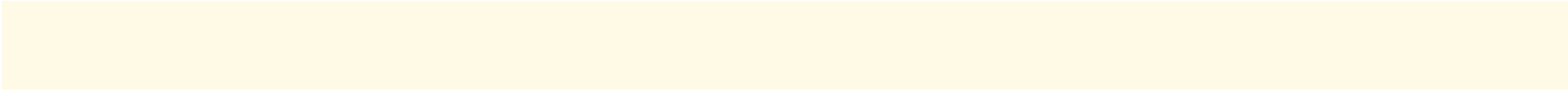


进行 PTP 移动时，开关点的值必须为零。



在 PTP 指令中，开关点只可能直接重合于被定义的点。如果仍然需要补偿死区时间（例如在粘结工艺），则只可能通过延迟时间 [ms] 来完成。

Pnd (摆动)



ana/BS/Pnd “摆动”功能在轨迹移动上叠加一个可以选择的摆型。



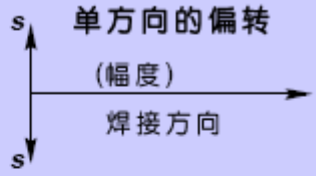
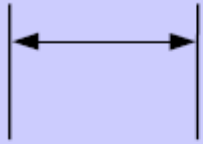
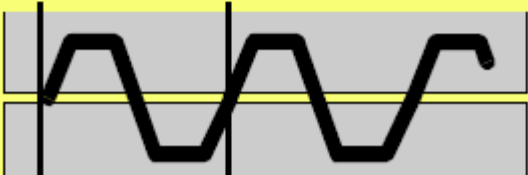
<==	BS bin/ana =	BS A/F =	摆动	ana kst+p	ana vprop:	ana konst:
-----	--------------	----------	-----------	-----------	------------	------------

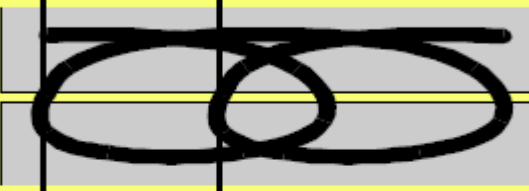
您可以通过按软键“ana/BS/Pnd”来选择此功能，并且软在变化的键条中选择“摆动”功能。选定之后，会出现下面的 InLine 表格：

Pendeln	EIN	Figur:	1	Amplitude:	1 [mm]	Periode:	1 [mm]	Ebene:	0 [°]
---------	------------	--------	----------	------------	---------------	----------	---------------	--------	--------------

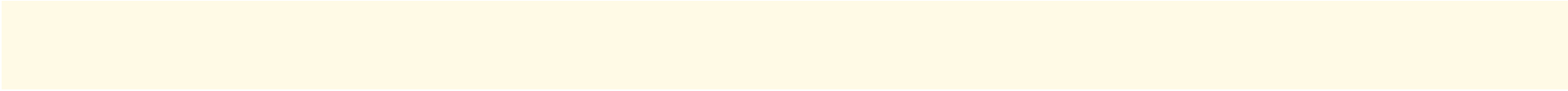
区域描述	功能	数值范围
Pendeln (摆动)	打开摆动或者关断摆动	开、关
Figur (摆型)	摆型选择	1 = 三角形 2 = 梯形 3 = 不对称的梯形 4 = 螺旋形 5 = 斜置的 8 字形 6 = 用户自定义 (专家平面) 7 = 用户自定义 (专家平面)
Amplitude (幅度)	单方向的偏转 (相当于摆宽的一半)	1 至 30 mm
Periode (周期)	摆长	1 至 50 mm
Ebene (平面)	摆动平面的转动	-90° 至 +90°

下列摆型可供选择：

摆型	 单方向的偏转 (幅度) 焊接方向	 摆长
无摆动		
1 三角形		
2 梯形		

3	不对称的梯形		
4	螺旋形		
5	斜置的8字形		
6	用户自定义 只在专家平面上		
7	用户自定义 只在专家平面上		

USER (KRL 功能块调用及参数传递)





在选择菜单键“指令”之后，通过其下属菜单“USER”，可以调用“VW_USR_R”工作程序。通过这个模块您可以使用 KRL 语言编制各种功能。为此，有许多局部子程序可供支配，您可以根据需要和它们的功能来使用。输入 InLine 表格中的参数将被传递给 VW_USR 模块。

指令

操作菜单键“指令”之后，选出菜单项“USER”：

- 0 上一个指令
- 1 标准移动 ▶
- 2 工艺移动 ▶
- 3 宏/子程序/钳 ▶
- 4 SPS ==> ▶
- 5 等候 /FB ▶
- 6 ANA/BS/摆动 ▶
- 7 USER**
- 8 注解



此功能只能通过菜单键“指令”来执行。它只允许在点 SPS 指令的内部进行。否则，将会出现出错提示。



用户可以调用 InLine 表格，只能进行参数传递（数值大小）的更改。这也只有在专家事先经过配置才可进行。USER 调用在宏 SPS 和其它的宏中都不允许。只有在点 SPS 指令中才可以使用。

接下来将打开下面的 InLine 表格，在表格中最多可以传递 7 个参数。

VW_USER P1= P2= P3= P4= P5=
P6= P7= EIN

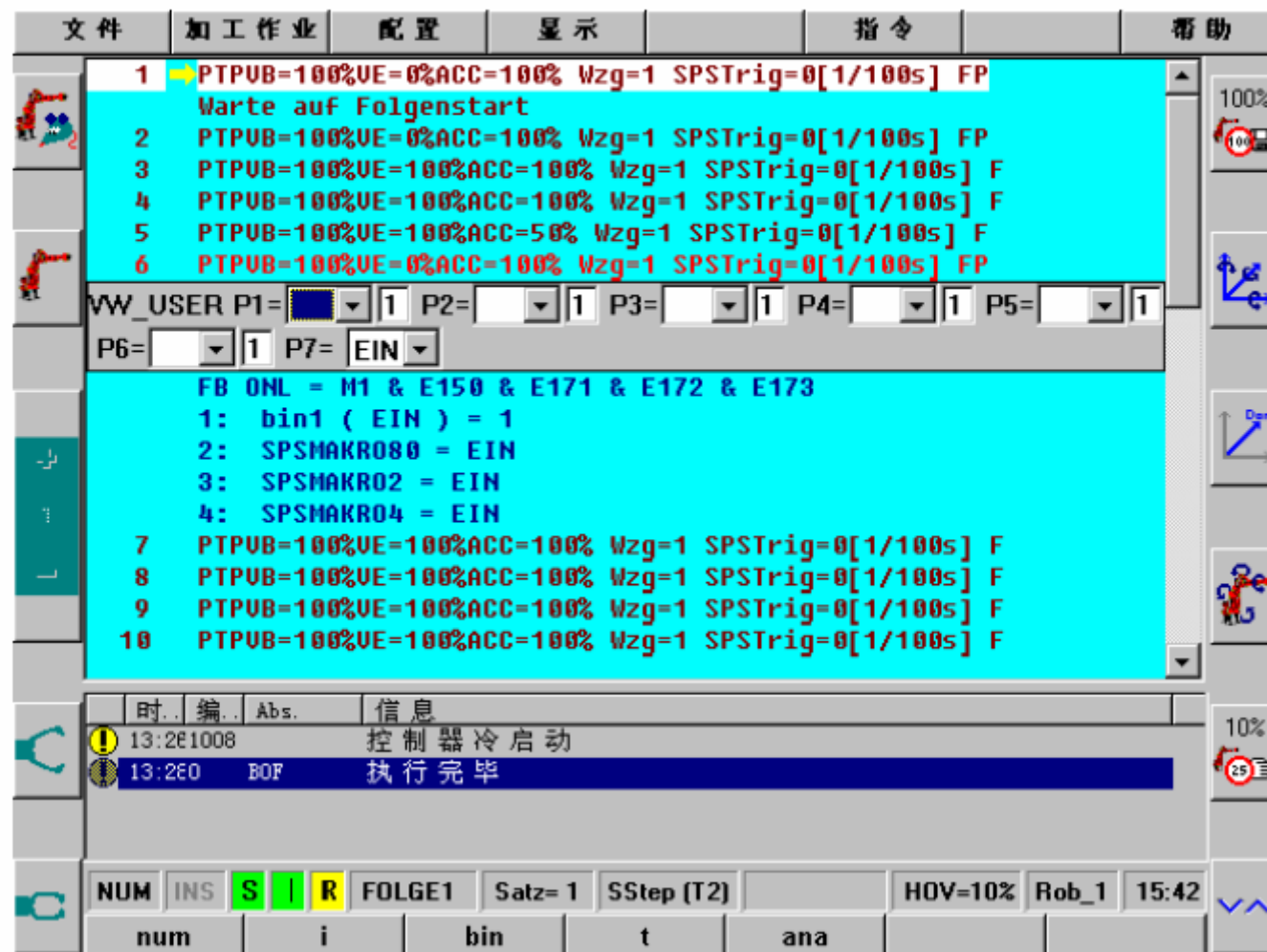
VW_USER P1= P2= P3= P4= P5=
P6= P7= 1

区域描述	功能	数值范围
P1...6=	算术操作数	i、bin、t、ana
1	号码	i (1...10)、bin (1...10)、t (1...10)、ana (1...8)
P7=	操作符 *1)	_, !
EIN, E	操作数类型	开、关、E、A、M、F、T、S
1	操作数的号码 *1)	E (1...1024)、A (1...1024)、M (1...24)、F (1...999)、T (1...10)、S (1...32)
*1) => 当逻辑状态为“开”、“关”时，不显示		

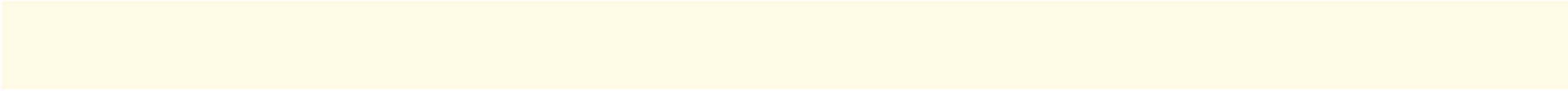
通过 **BOF**（用户界面）调用 **VW_USER** 指令

指令

按下菜单键“指令”和选定菜单项“USER”之后，可以输入所需的数值，前提是USER已经过“配置”：

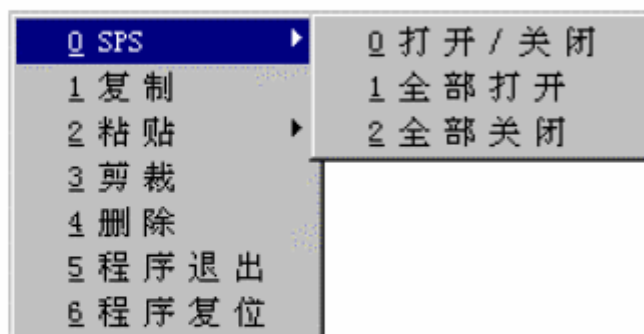


程序处理



编辑

为使程序结构更加直观明了，可以把 SPS 指令隐藏起来，仅在需要参考信息或者进行处理时才显示它们。通过选定菜单键“编辑”和其中的菜单“SPS”，您就可以利用此选项。



菜单“SPS”提供如下的下属菜单：

- 打开 / 关闭 打开或关闭程序中编辑光标所在之指令行的折合。
- 全部打开 打开程序中所有的折合。
- 全部关闭 关闭程序中所有的折合。

SPS 开

打开或关闭 SPS 的功能也可以通过带有软键的软键条进行。

SPS 关

```
1  PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
   Warte auf Folgenstart
2  PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] FP
   FB ONL = EIN
   1: WARTE BIS E18 & M1
   2: -- geht noch --
3  PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
4  PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
5  PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
6  PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
7  PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
8  PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
9  PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
10 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
11 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
12 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
13 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
```

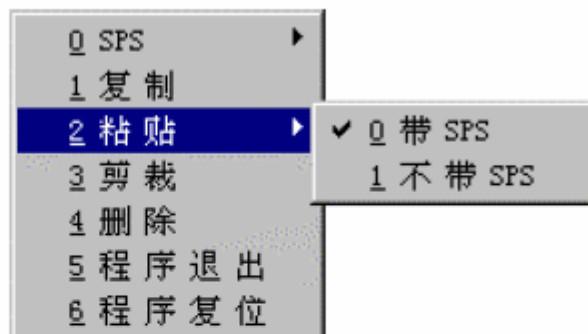
如图所示为第二个程序点上打开的折合。

块处理功能

编辑

■ 编辑

下列菜单功能皆可用于编辑处理，它们可以作为块处理功能一次性用于多个程序项或者也可以用于个别程序项：



可以一下子标识若干行。为此请按下 Shift 键和 PGUP “↑” 或 PGDN “↓” 键。程序的最后一个语句可以用“向右”的光标键标识。被标识的程序行为红色。标识时可以往右，也可以往左进行。欲取消标识，可以按任意一个键、按 Esc 键或者 PGUP “↑” 或 PGDN “↓” 键。然后可以从菜单键“编辑”中，选定块处理的标准功能。这些标准功能有：复制、粘贴、剪裁和删除。



一般情况下块处理功能无法实现的项目有：

1. 序列的第一点不能被删除和剪裁。
2. 宏的开头、等候序列启动和子程序的开头，不能被复制、删除或者剪裁。

复制

标识的文字范围将复制在中间寄存器中。而其原样保持不变。

粘贴，带／不带 SPS

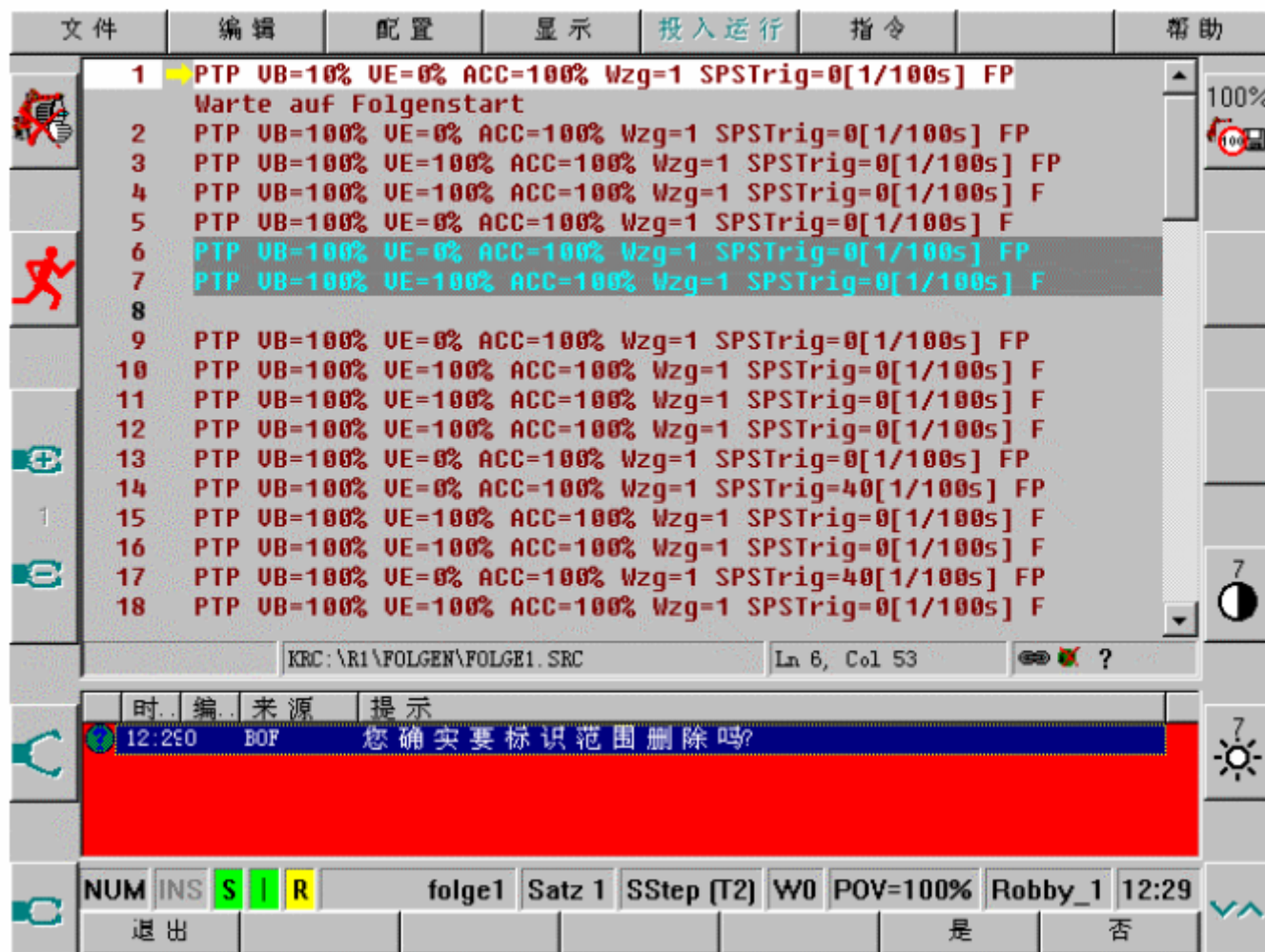
中间寄存器中的文字块将被粘贴在标识的行后面。可以选择粘贴时是否带 SPS。

剪裁

被标识的文字内容将被复制到中间寄存器，并且原文将从文件中被去除。

删除

标识的文字内容将从程序中删除。激活此功能后，跟着将出现安全应答。



使用注释

通过插入注释或注释行，可以使您的程序更加清楚明了，并使他人容易理解。

指令



按下菜单键“指令”可进入下属菜单“注释”。按光标键“↑”或“↓”便可进行选定，选定后的确认则按回车键，或者通过处于激活状态的数字区输入带下划线的数字。

- 0 最后一个指令
- 1 标准移动 ▶
- 2 工艺移动 ▶
- 3 宏 / 子程序 / 钳 ▶
- 4 SPS ==> ▶
- 5 等候 / FB ▶
- 6 ANA/BS/摆动 ▶
- 7 用户
- 8 注解

选定此功能时，程序窗中打开下列Inline 表格，软键条改变为退出和确认指令。

1	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
Warte auf Folgenstart	
2	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] FP
Comment:	
3	PTP UB=100% UE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
4	PTP UB=100% UE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
5	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
6	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
7	PTP UB=100% UE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
8	PTP UB=100% UE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s]
9	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
10	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
11	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
12	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
13	PTP UB=100% UE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]

指令退出

指令认可

在您输入完注释内容后，按软键“指令认可”或者回车键。

```
1 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
  Warte auf Folgenstart
2 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s] FP
  -- schwpkt.125 --
3 PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s] F
4 PTP VB=100% VE=100% ACC=100% Wzg=1 SPSTrig=0[1/100s] FP
  FB ONL = M1
  → i: -- ab hier Verriegelung --
5 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
6 PTP VB=100% VE=0% ACC=100% Wzg=1 SPSTrig=0[1/100s]
```



在重新调用“注释”指令时，上次编辑过的字符串将作为参考出现在 InLine 表格中。

...改动

改动

直接把程序光标移到注释行上，不管注释行是位于折合内还是界于两个程序项之间，然后按下软键“改动”。接着将打开 InLine 表格，现有的文字便可供改动。



现在您可以根据需要更改文字内容。最后请用软键“指令认可”或回车键接纳文字输入，或是以软键“指令退出”或者 ESC 键退出此编辑过程。

指令认可

指令退出



...删除

± ¼-



请直接把程序光标移到注释行上，不管注释行是位于折合内还是界于两个程序项之间，按下软键“编辑”，然后把光标键移到菜单项“删除”上。

- 0 SPS ▶
- 1 复制
- 2 粘贴 ▶
- 3 剪裁
- 4 删除**
- 5 程序退出
- 6 程序复位



按下回车键会出现安全应答和相应的软键条。

时	编	Abs.	信息
11:450	BOF		确实要删除本行？

NUM	INS	S	I	R	FOLGE1	Satz= 1	T1		POV=100%	R_Name	11:46
退出									是	否	

您如果确定应该删除此行，请按软键“是”，该注释行将从程序中消失。若您按软键“否”或“退出”，则将退出删除功能，注释行在程序中维持原样，句子指针和程序标记将停留在注释行上。