

正确开/关机的操作

- 开机

- 在确认输入电压正常后，打开电源开关。

- 关机

- 1、在示教器的“重新启动”菜单中选择“关机”。
- 2、关闭电源开关

注意：关机后再次开启电源需要等2分钟。

机器人周期保养

- 控制柜的周期保养（ ABB 机器人随机光盘说明书中有详细的说明 ）
- 机器人本体的周期保养（ ABB 机器人随机光盘说明书中有详细的说明 ）
- 机器人工具的周期保养（ 根据工具供应商提供具体说明进行保养 ）

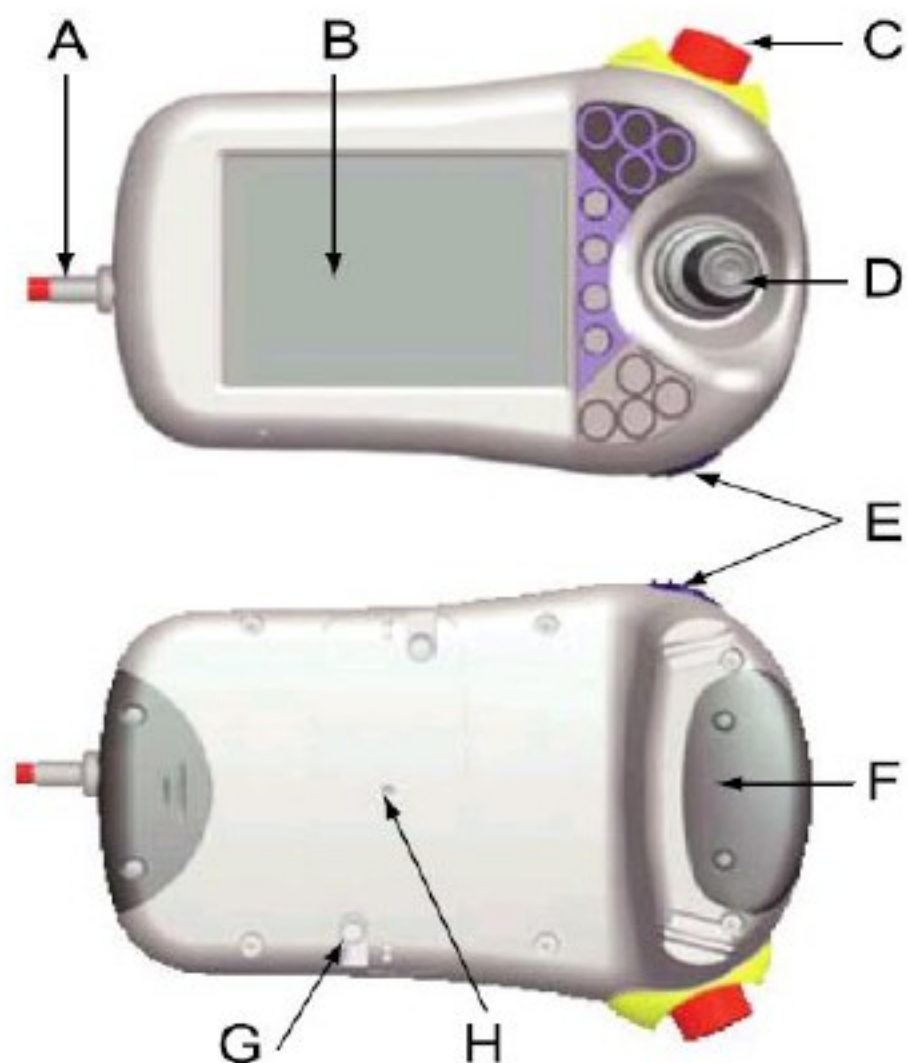
控制面板画面的说明

项目	说明
外观	外观自定义显示器亮度和左/右手操作习惯的设置。
监控	动作碰撞监控设置和执行设置。
FlexPendant	示教器操作特性的设置。
I/O	配置常用 I/O 列表，在“输入输出”菜单中第一页显示。
语言	机器人控制器当前语言的设置。
ProgKeys	为可编程按钮指定输入输出信号。
日期和时间	机器人控制器的日期和时间设置。
诊断	创建诊断文件以利于故障排除。
配置	配置系统参数设置。
触摸屏	触摸屏重新校准设置。

ABB机器人示教器

什么是示教器？

示教器是进行机器人的手动操纵、程序编写、参数配置以及监控用的手持装置，也是我们最常打交道的控制装置。



示教器解说：

- A 连接电缆
- B 触摸屏
- C 急停开关
- D 手动操作摇杆
- E 数据备份用USB接口
- F 使能器按钮
- H 示教器复位按钮
- G 触摸屏用笔

如何手持示教器



示教器的使能器



- 使能器按钮分了两档，在手动状态下第一档按下去，机器人将处于电机开启状态。
- 第二档按下去以后，机器人又处于防护装置停止状态。

使能器按钮的作用

- 使能器按钮是工业机器人为保证操作人员人身安全而设置。
- 只有在按下使能器按钮，并保持在“电机开启”的状态，才可对机器人进行手动的操作与程序的调试。
- 当发生危险时，人会本能地将使能器按钮松开或按紧，则机器人会马上停下来，保证安全。

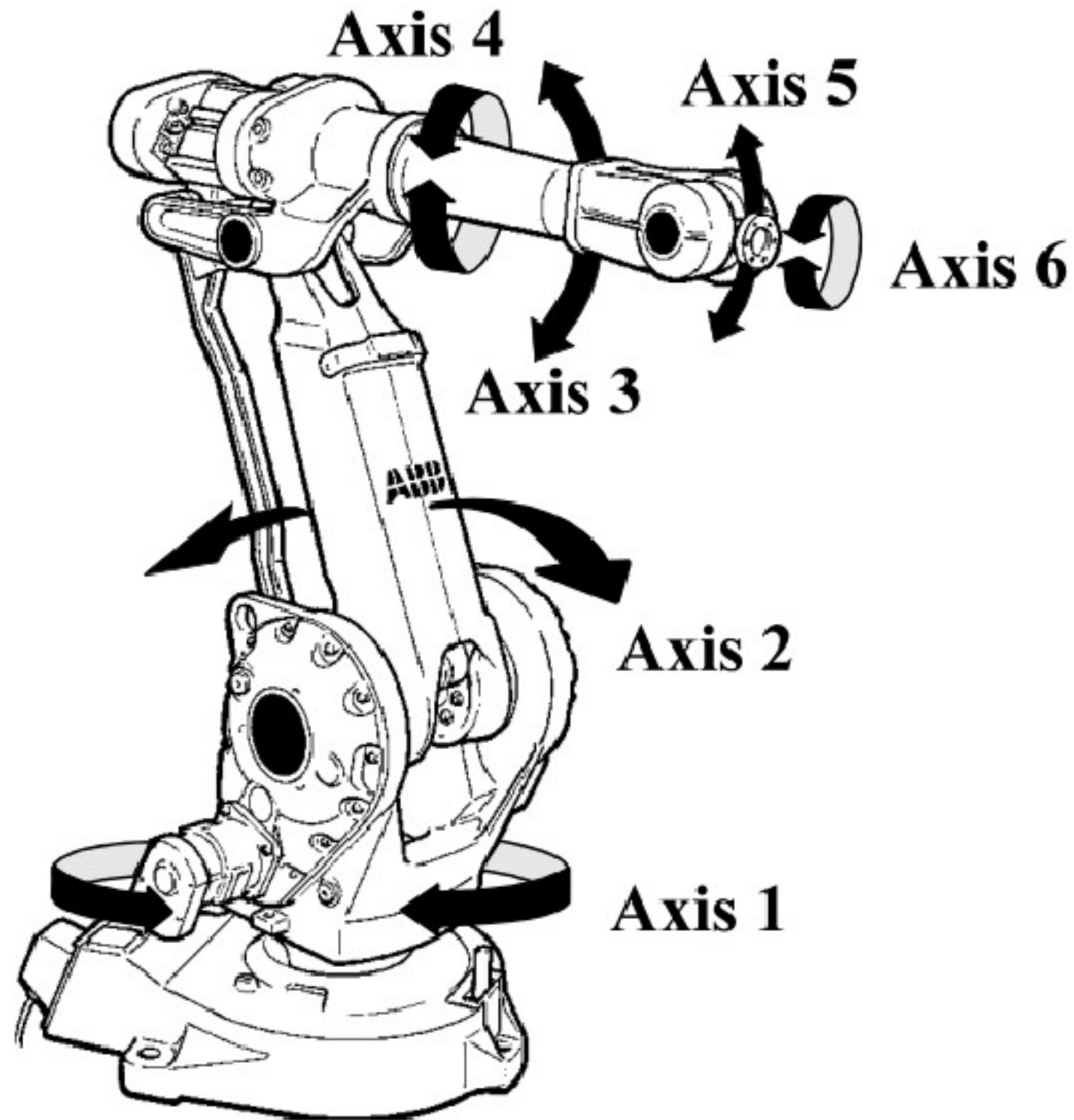
示教器摇杆的操作技巧



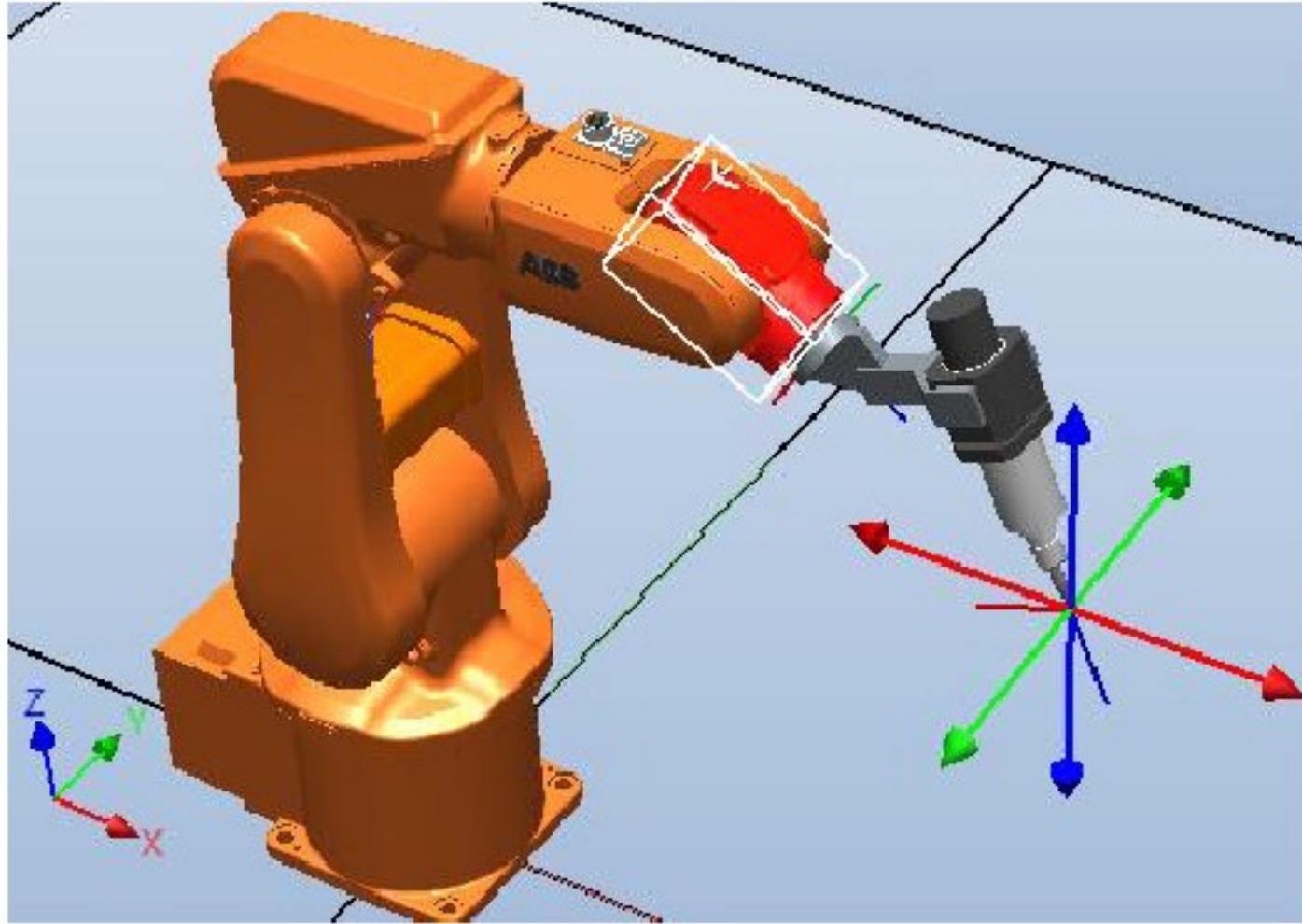
操纵杆的使用技巧：我们可以将机器人的操纵杆比作汽车的油门，操纵杆的操纵幅度是与机器人的运动速度相关的。

- 操纵幅度较小则机器人运动速度较慢。
- 操纵幅度较大则机器人运动速度较快。
- 所以大家在操作的时候，尽量以操纵小幅度使机器人慢慢运动，开始我们的手动操纵学习。

ABB机器人6个关节轴的示意图



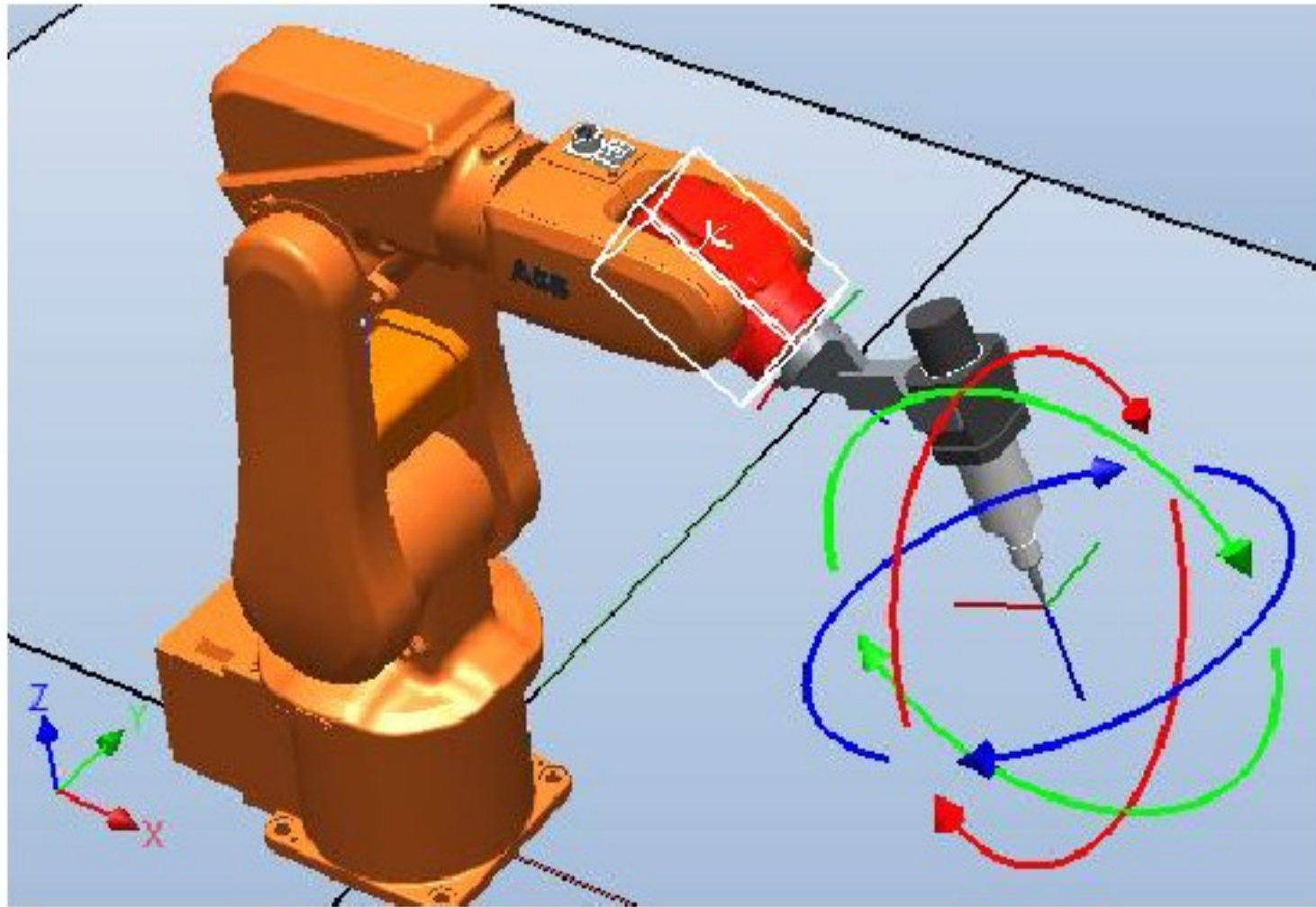
ABB机器人的手动操纵线性运动



机器人的线性运动是指安装在机器人第六轴法兰盘上的工具在空间中作线性运动。

- 如果对使用操纵杆通过位移幅度来控制机器人运动的速度不熟练的话。那么可以使用“增量”模式，来控制机器人运动。
- 在增量模式下，操纵杆每位移一次，机器人就移动一步。如果操纵杆持续一秒或数秒钟，机器人就会持续移动（速率为每秒10步）。

ABB机器人的手动操纵重定位运动

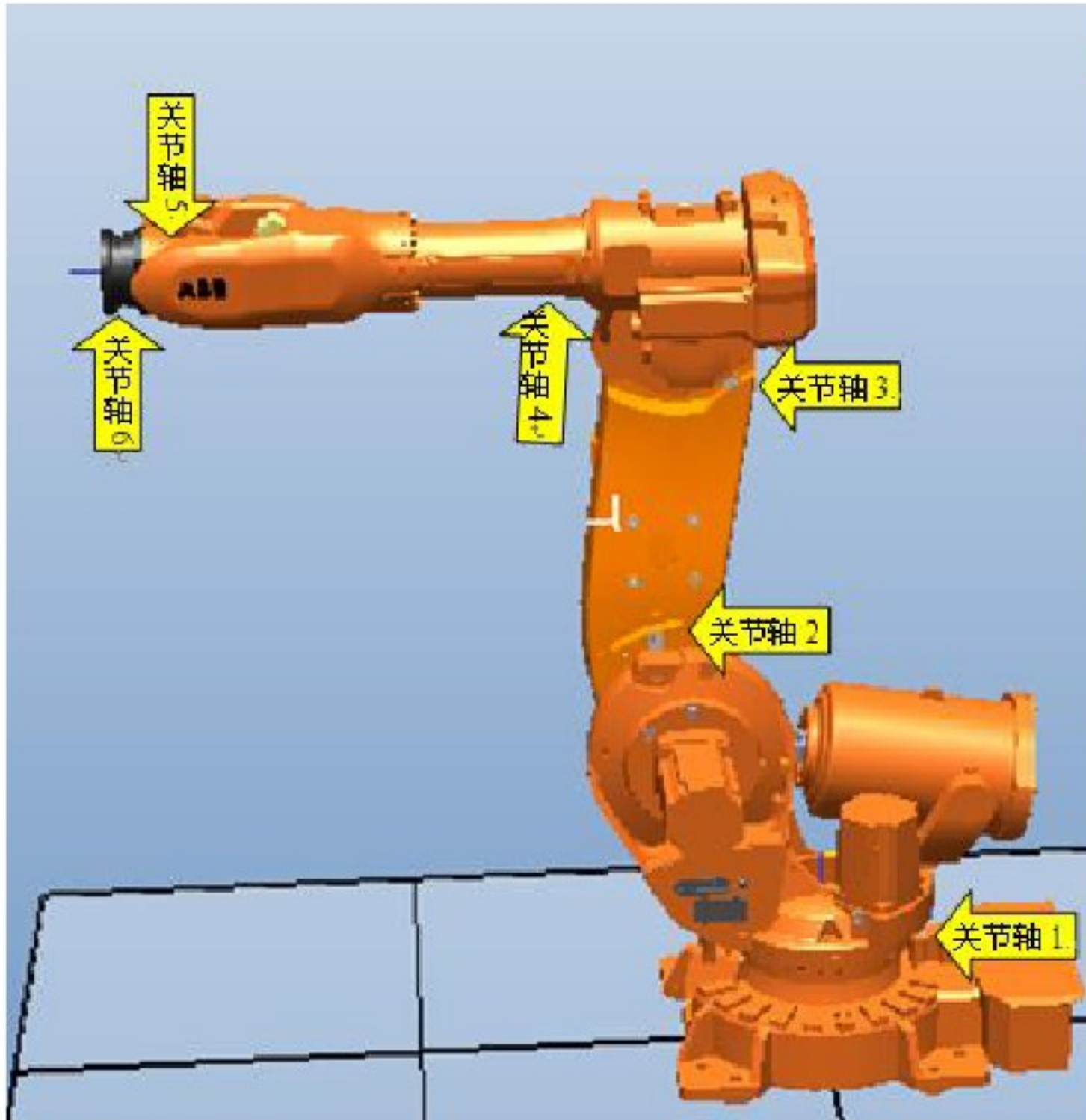


- 机器人的重定位运动是指机器人第六轴法兰盘上的工具TCP点在空间中绕着工具坐标系旋转的运动，也可理解为机器人绕着工具TCP点作姿态调整的运动。

- 如果对使用操纵杆通过位移幅度来控制机器人运动的速度不熟练的话。那么可以使用“增量”模式，来控制机器人运动。

- 在增量模式下，操纵杆每位移一次，机器人就移动一步。如果操纵杆持续一秒或数秒钟，机器人就会持续移动（速率为每秒10步）。

ABB机器人的转数计数器更新操作



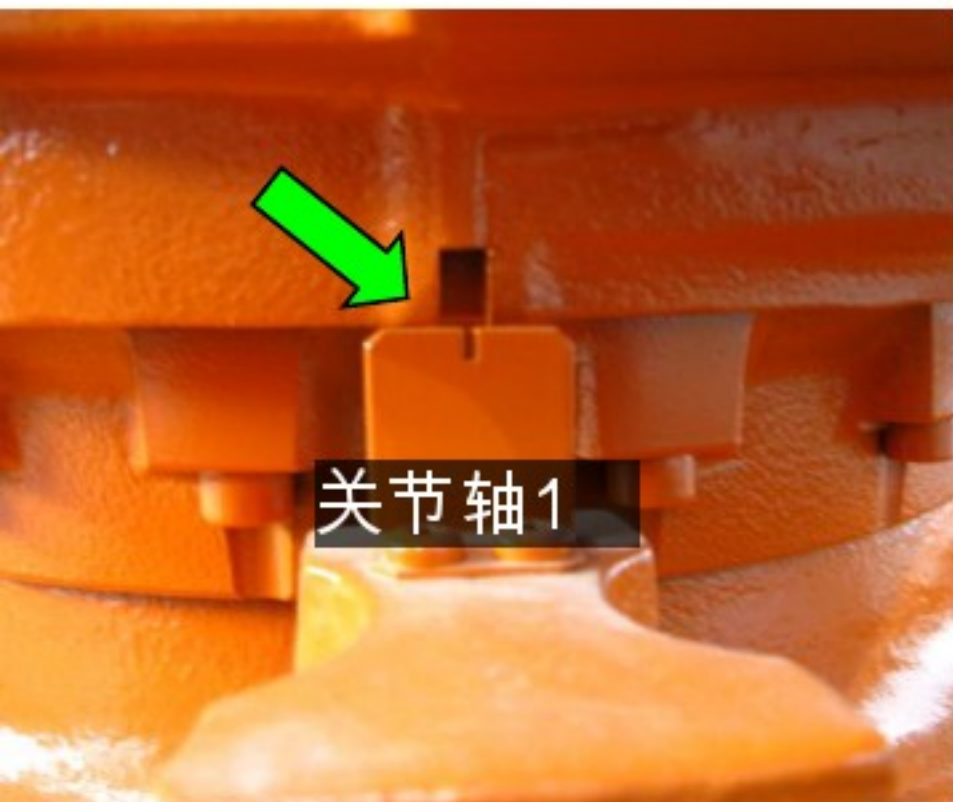
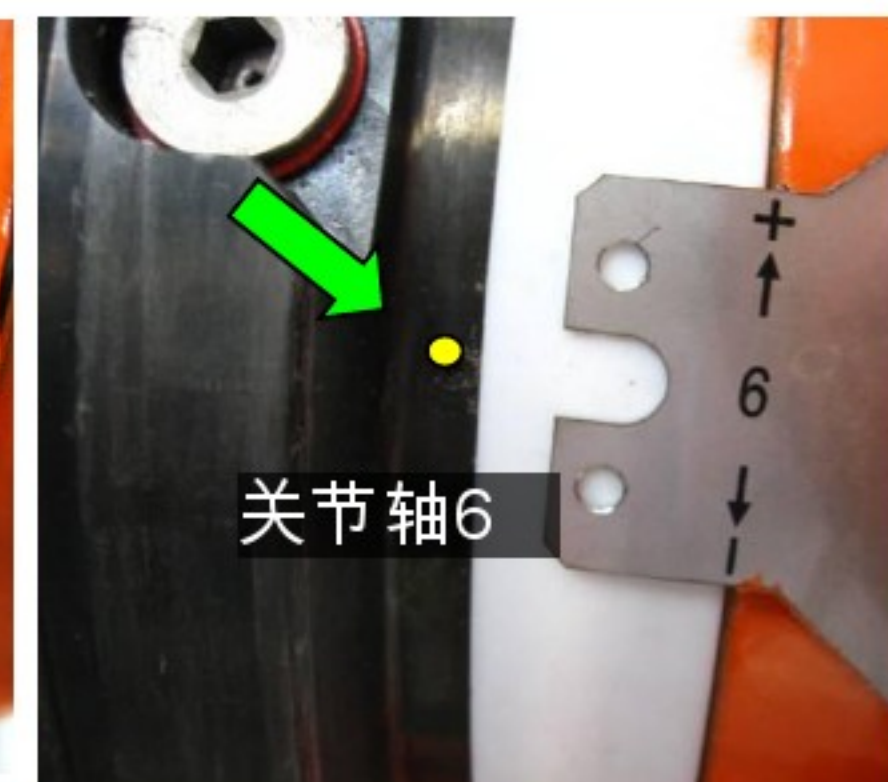
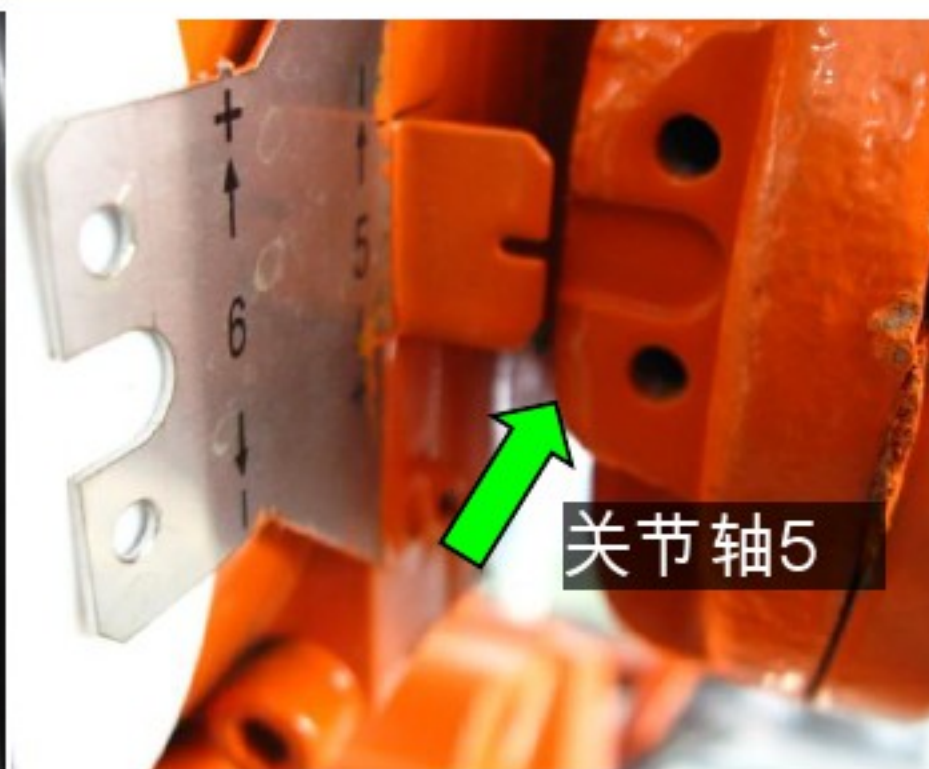
ABB机器人六个关节轴都有一个机械原点的位置。

在以下的情况，我们需要对机械原点的位置进行转数计数器更新操作：

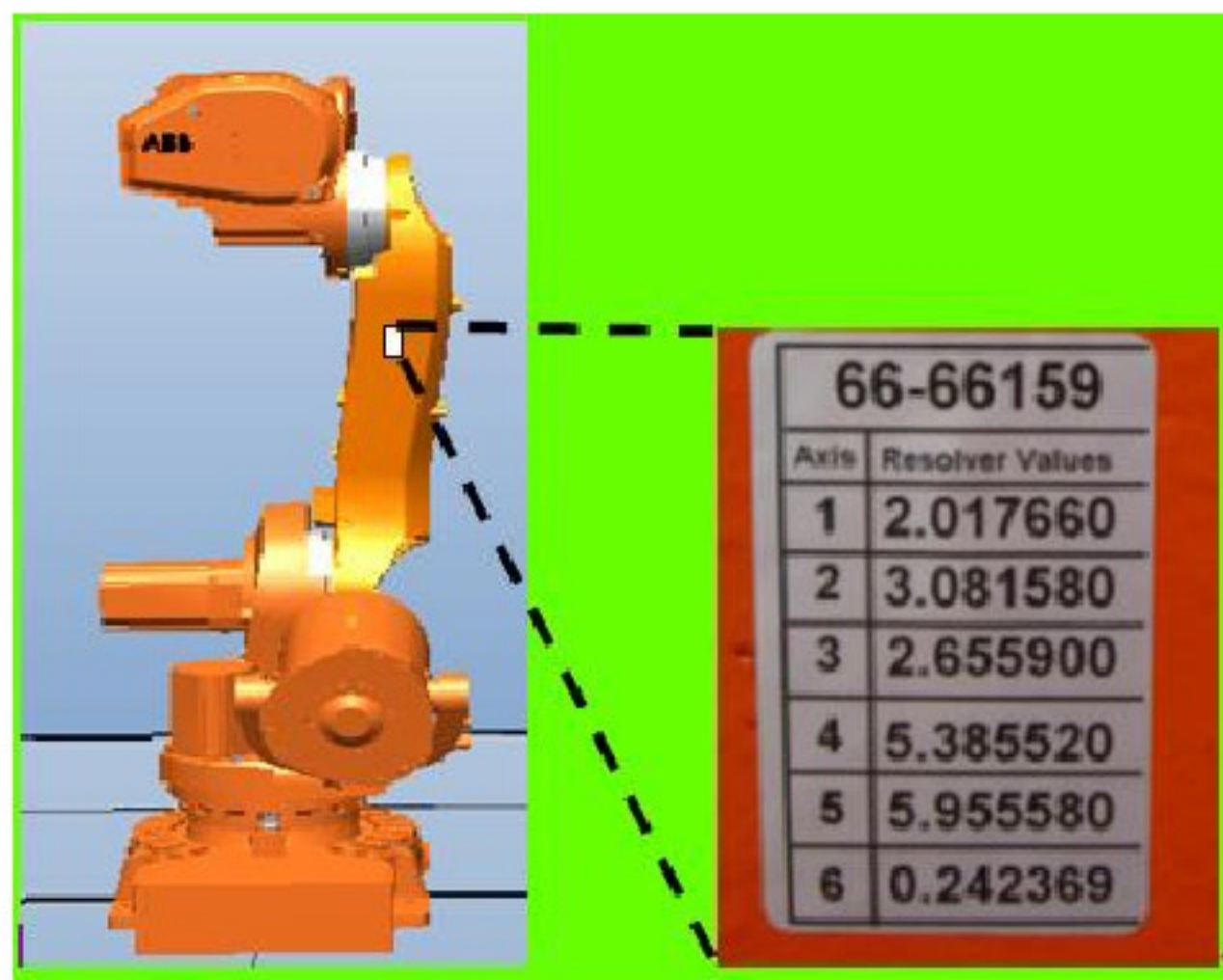
- ▶ 更换伺服电机转数计数器电池后。
- ▶ 当转数计数器发生故障，修复后。
- ▶ 转数计数器与测量板之间断开过以后。
- ▶ 断电后，机器人关节轴发生了移动。
- ▶ 当系统报警提示“10036 转数计数器未更新”时。

ABB机器人IRB6640机械原点刻度位置

- 各个型号的机器人机械原点刻度位置会有所不同，请参考ABB随机光盘说明书。



ABB机器人IRB6640转数计数器偏置数据



- 如果机器人由于安装位置的关系，无法六个轴同时到达机械原点刻度位置，则可以逐一对关节轴进行转数计数器更新。

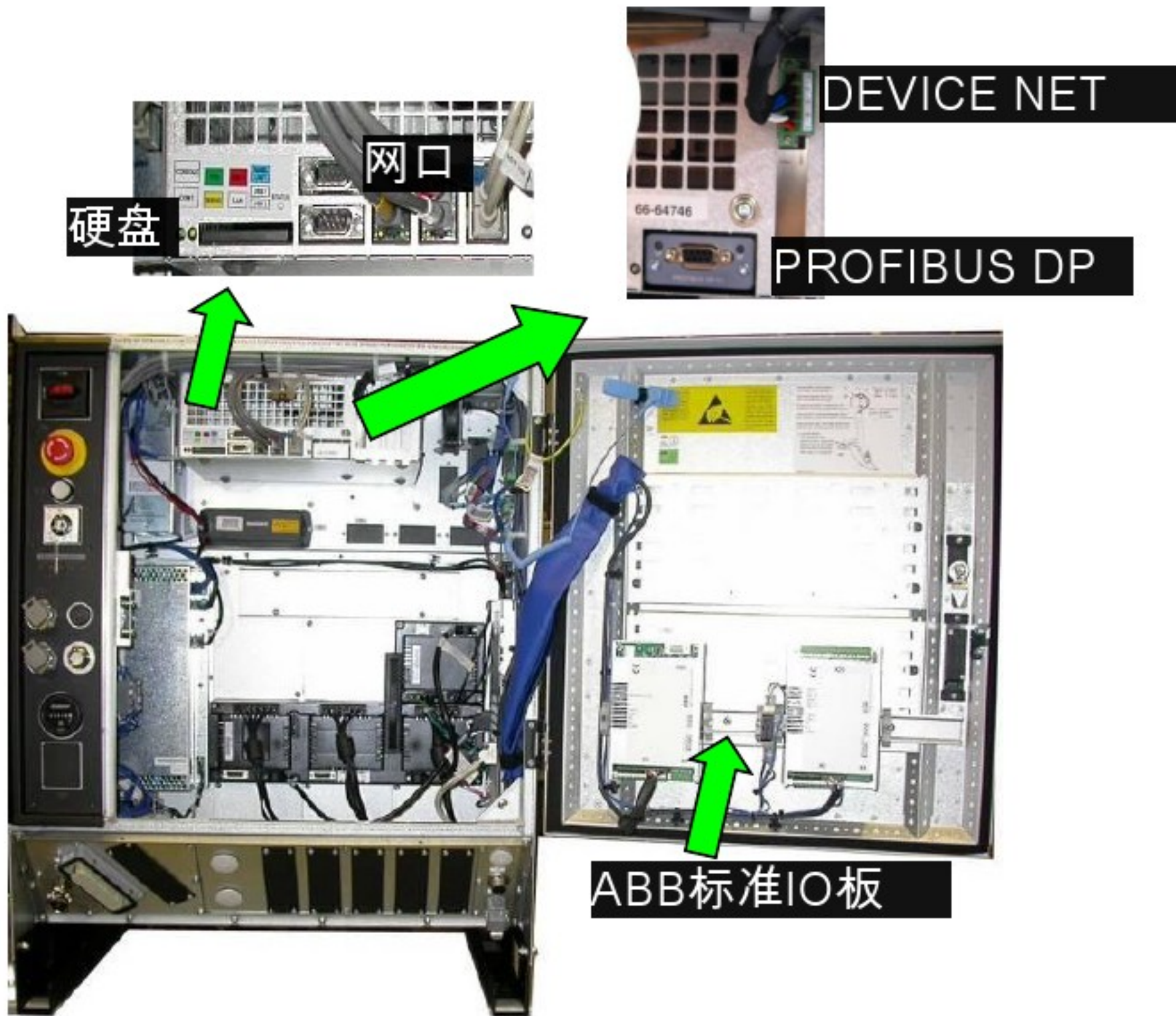
ABB机器人IO通讯的种类

ABB机器人		
PC	现场总线	ABB标准
RS232通讯 OPC server Socket Message	Device Net Profibus Profibus-DP Profinet EtherNet IP	标准IO板 PLC

关于ABB机器人IO通讯接口的说明：

- ABB的标准IO板提供的常用信号处理有数字输入DI、数字输出DO、模拟输入AI、模拟输出AO以及输送链跟踪，在本章中会对此作一介绍。
- ABB机器人可以选配标准ABB的PLC，省去了原来与外部PLC进行通讯设置的麻烦，并且在机器人的示教器上就能实现与PLC相关的操作。
- 我们就以最常用的ABB标准IO板DSQC651和Profibus-DP为例进行，进行详细的讲解如何进行相关的参数设定。

ABB机器人IO通讯接口示例



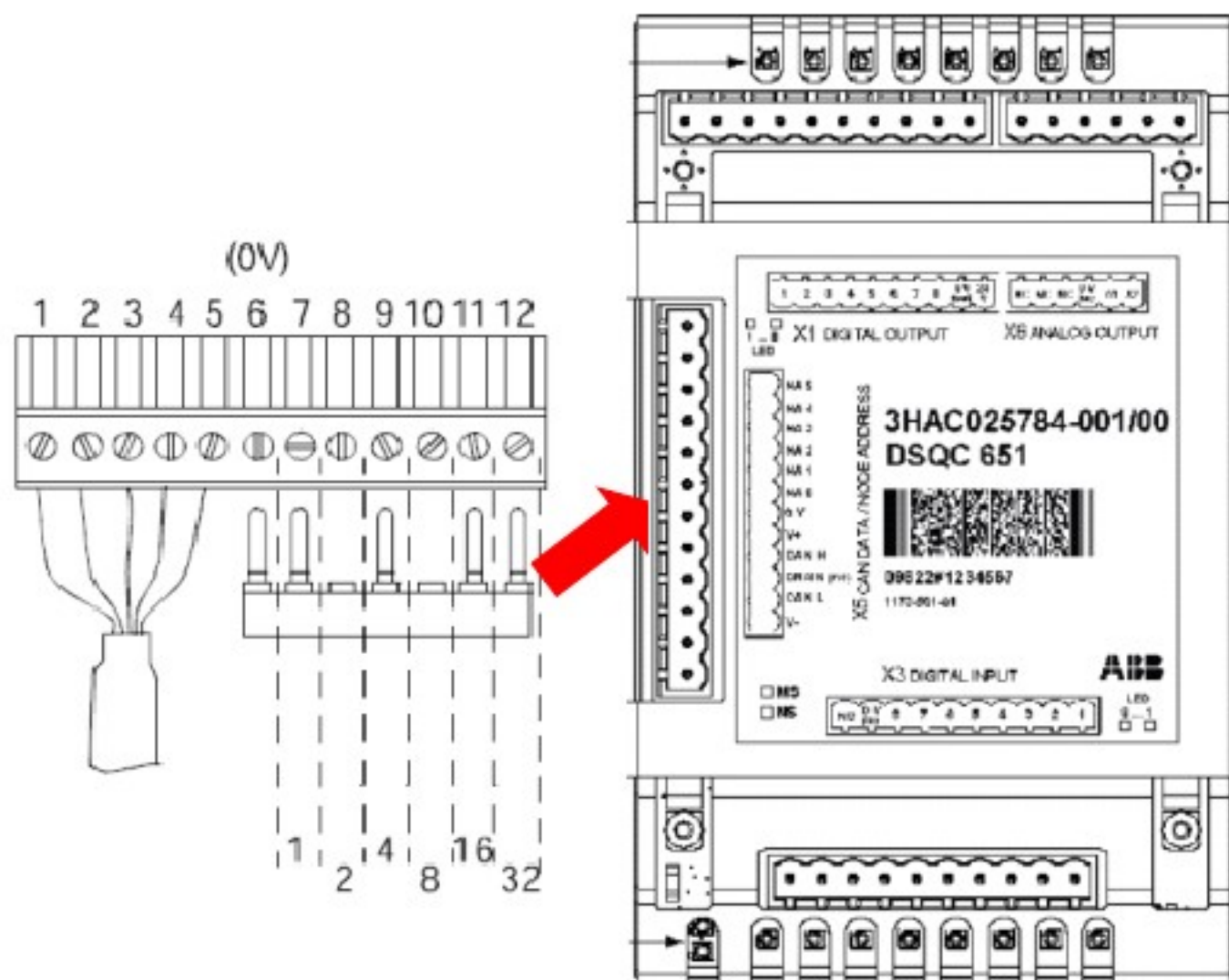
ABB机器人IO信号设定的顺序

- 设定IO模块单元 (UNIT)



- 设定IO信号

ABB机器人标准IO DSQC651

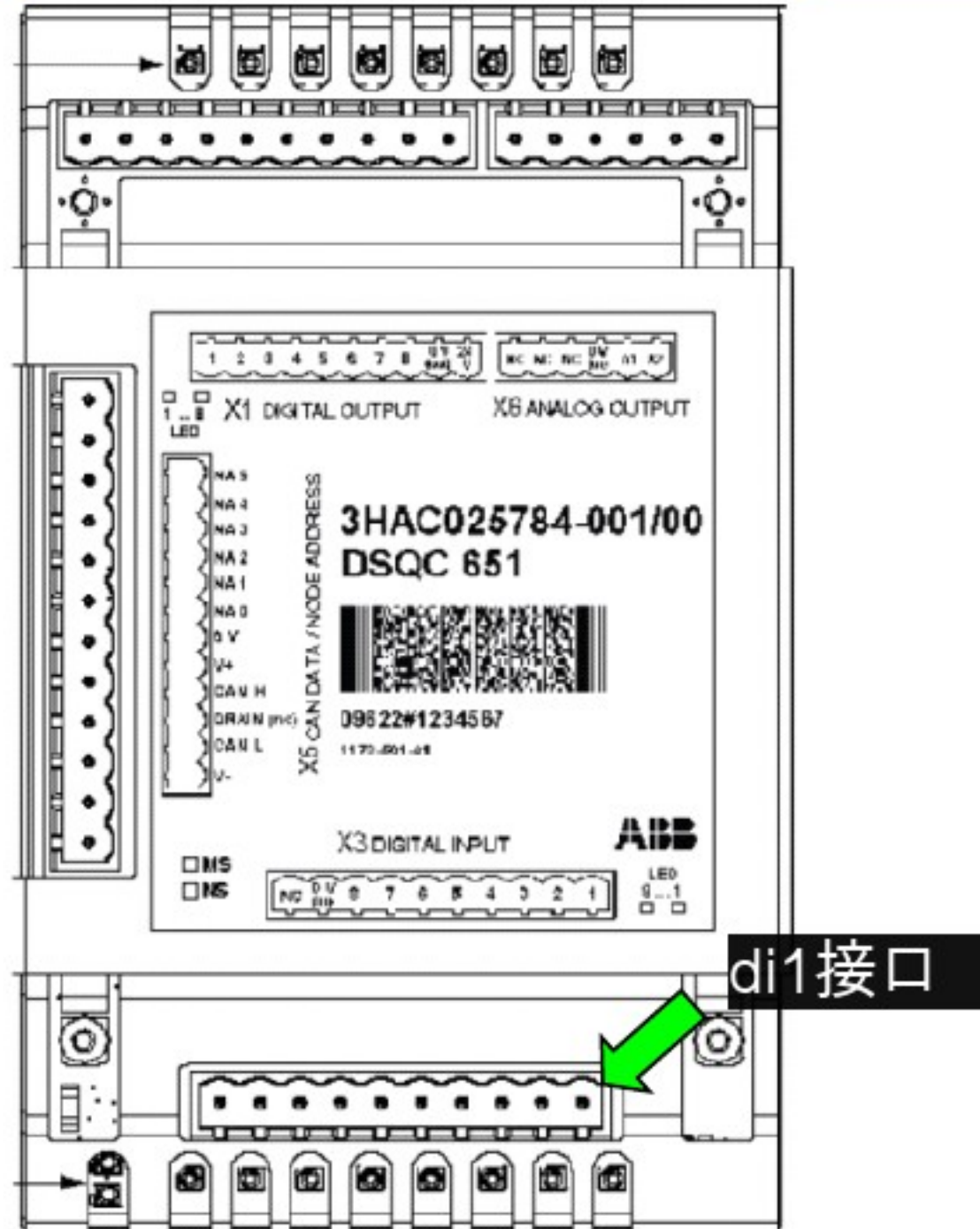


- ABB标准IO板是挂在DeviceNet网络上的，所以要设定模块在网络中的地址。端子X5的6-12的跳线就是用来决定模块的地址的，地址可用范围为10-63.
- 如上图所示，将第8脚和第10脚的跳线剪去， $2+8=10$ 就可以获得10的地址。

参数名称	设定值	说明
Name	board10	设定IO板在系统中的名字
Type of Unit	d651	设定IO板的类型
Connected to Bus	DeviceNet1	设定IO板连接的总线
DeviceNet Address	10	设定IO板在总线中的地址

ABB机器人标准IO di1 数字输入信号

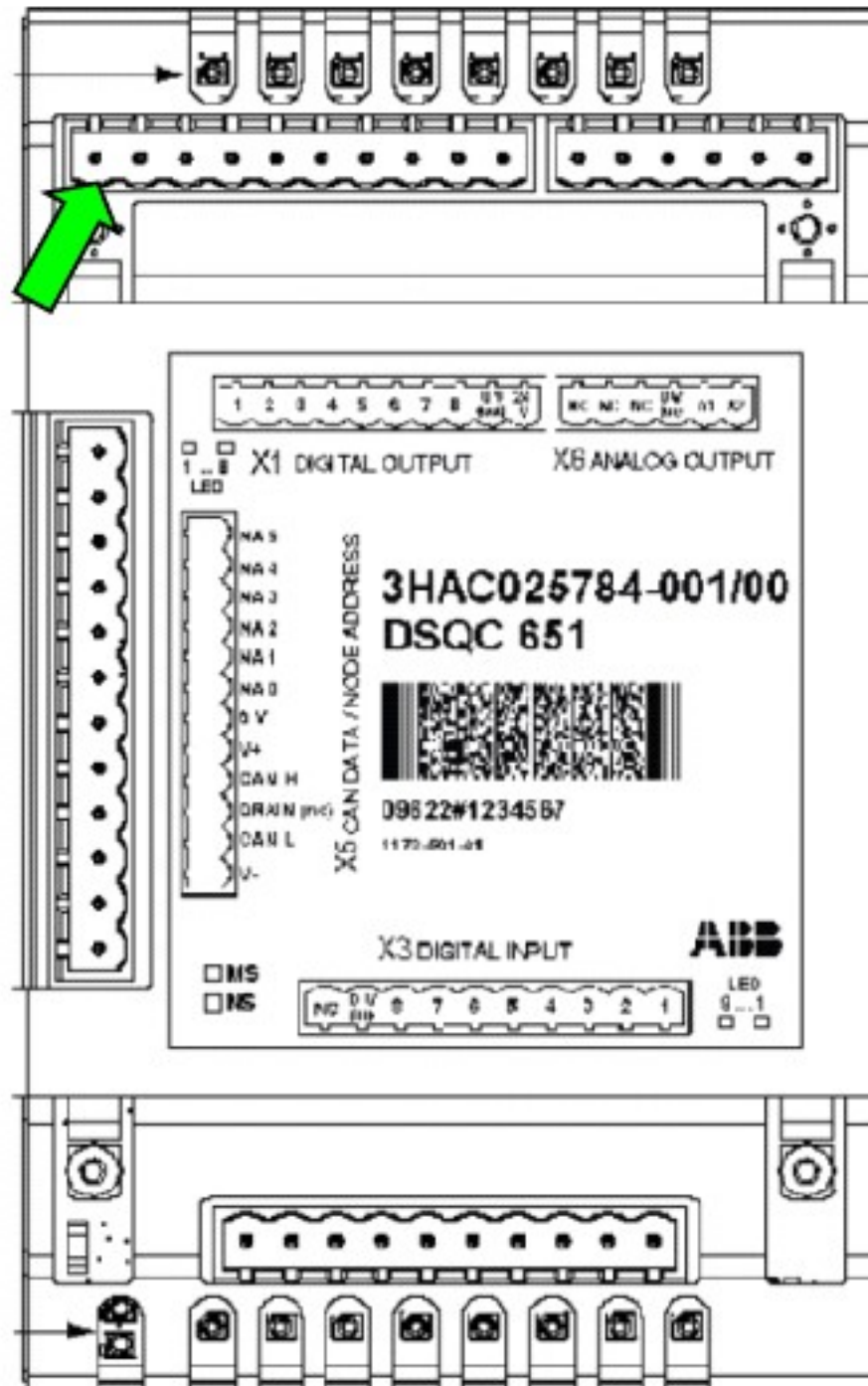
参数名称	设定值	说明
Name	di1	设定数字输入信号的名字
Type of Signal	Digital Input	设定信号的类型
Assigned to Unit	board10	设定信号所在的IO模块
Unit Mapping	0	设定信号所占用的地址



ABB机器人标准IO do1 数字输出信号

参数名称	设定值	说明
Name	do1	设定数字输出信号的名字
Type of Signal	Digital Output	设定信号的类型
Assigned to Unit	board10	设定信号所在的IO模块
Unit Mapping	32	设定信号所占用的地址

do1接口

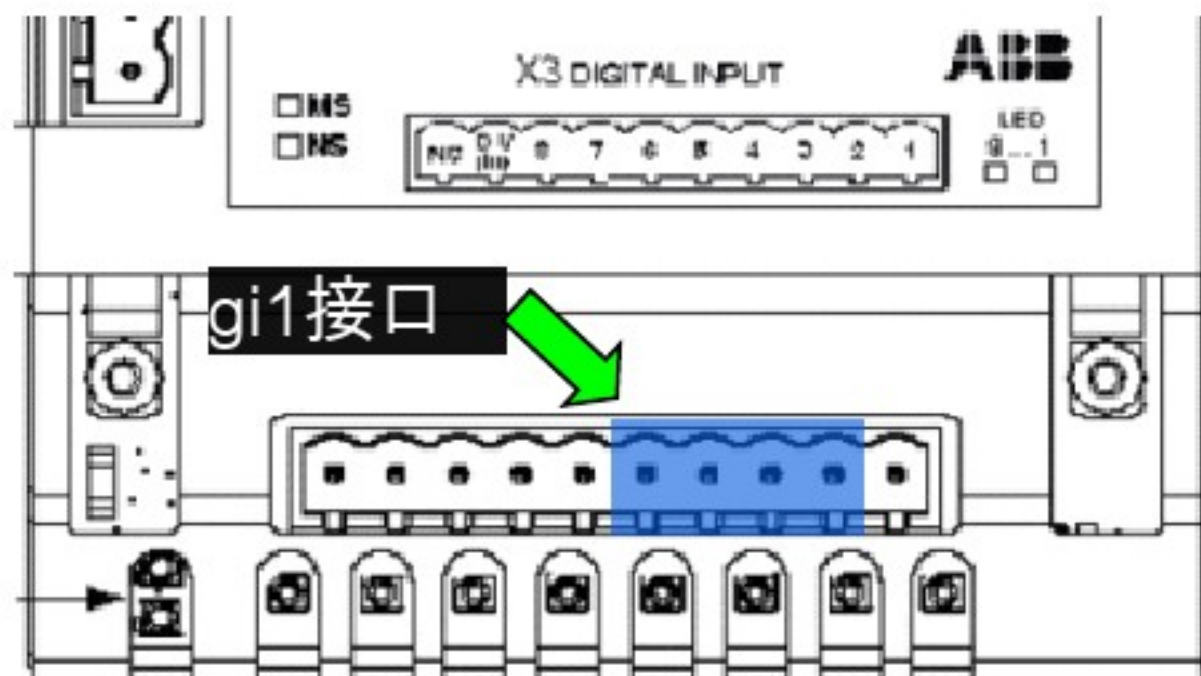


ABB机器人标准IO gi1 组输入信号

参数名称	设定值	说明
Name	gi1	设定组输入信号的名字
Type of Signal	Group Input	设定信号的类型
Assigned to Unit	board10	设定信号所在的IO模块
Unit Mapping	1-4	设定信号所占用的地址

	地址1	地址2	地址3	地址4	十进制数
	1	2	4	8	
状态1	0	1	0	1	2+8=10
状态2	1	0	1	1	1+4+8=13

- 组输入信号就是将几个数字输入信号组合起来使用，用于接受外围设备输入的BCD编码的十进制数。
- 此例中，gi1占用地址1-4共4位，可以代表十进制数0-15。如此类推，如果占用地址5位的话，可以代表十进制数0-31。

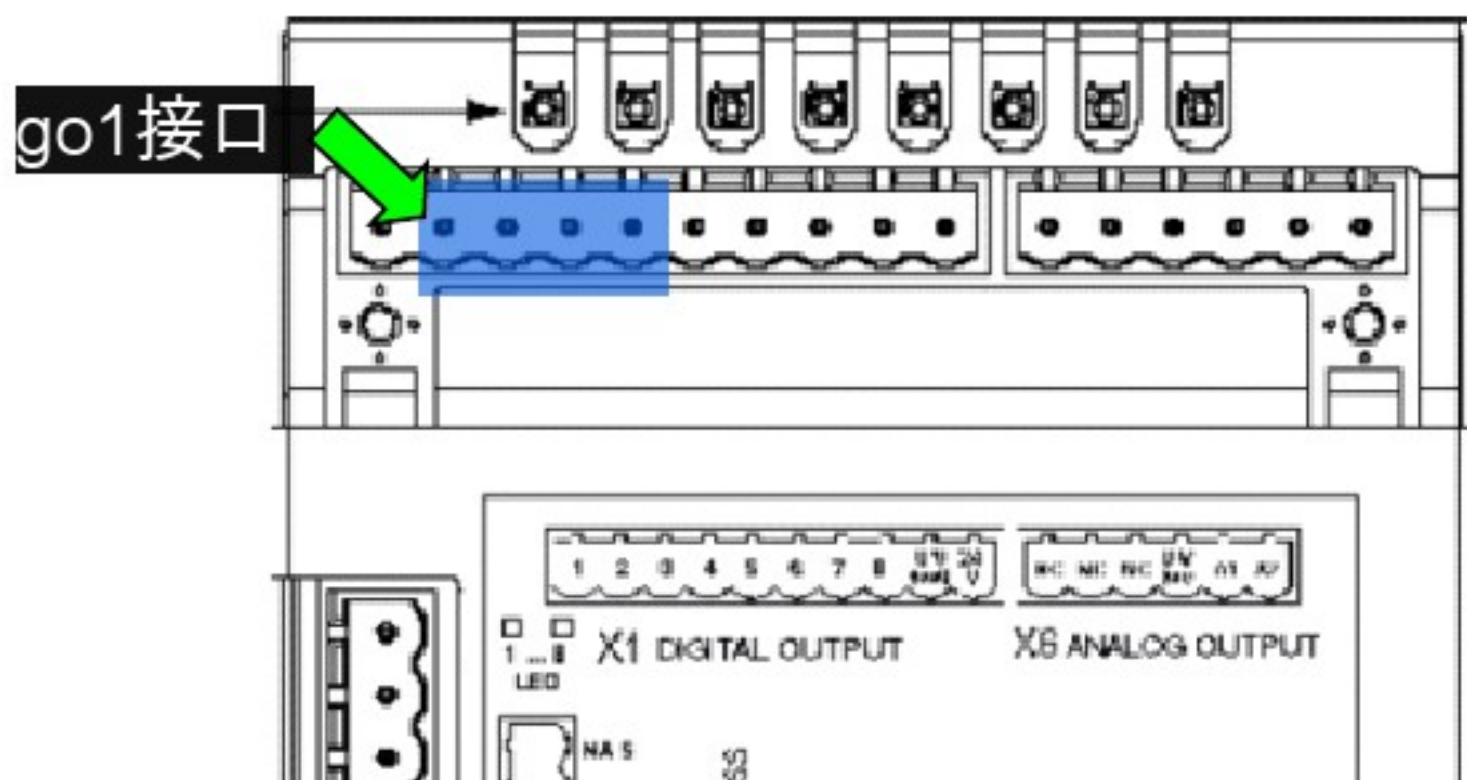


ABB机器人标准IO go1 组输出信号

参数名称	设定值	说明
Name	go1	设定组输出信号的名字
Type of Signal	Group Output	设定信号的类型
Assigned to Unit	board10	设定信号所在的IO模块
Unit Mapping	33-36	设定信号所占用的地址

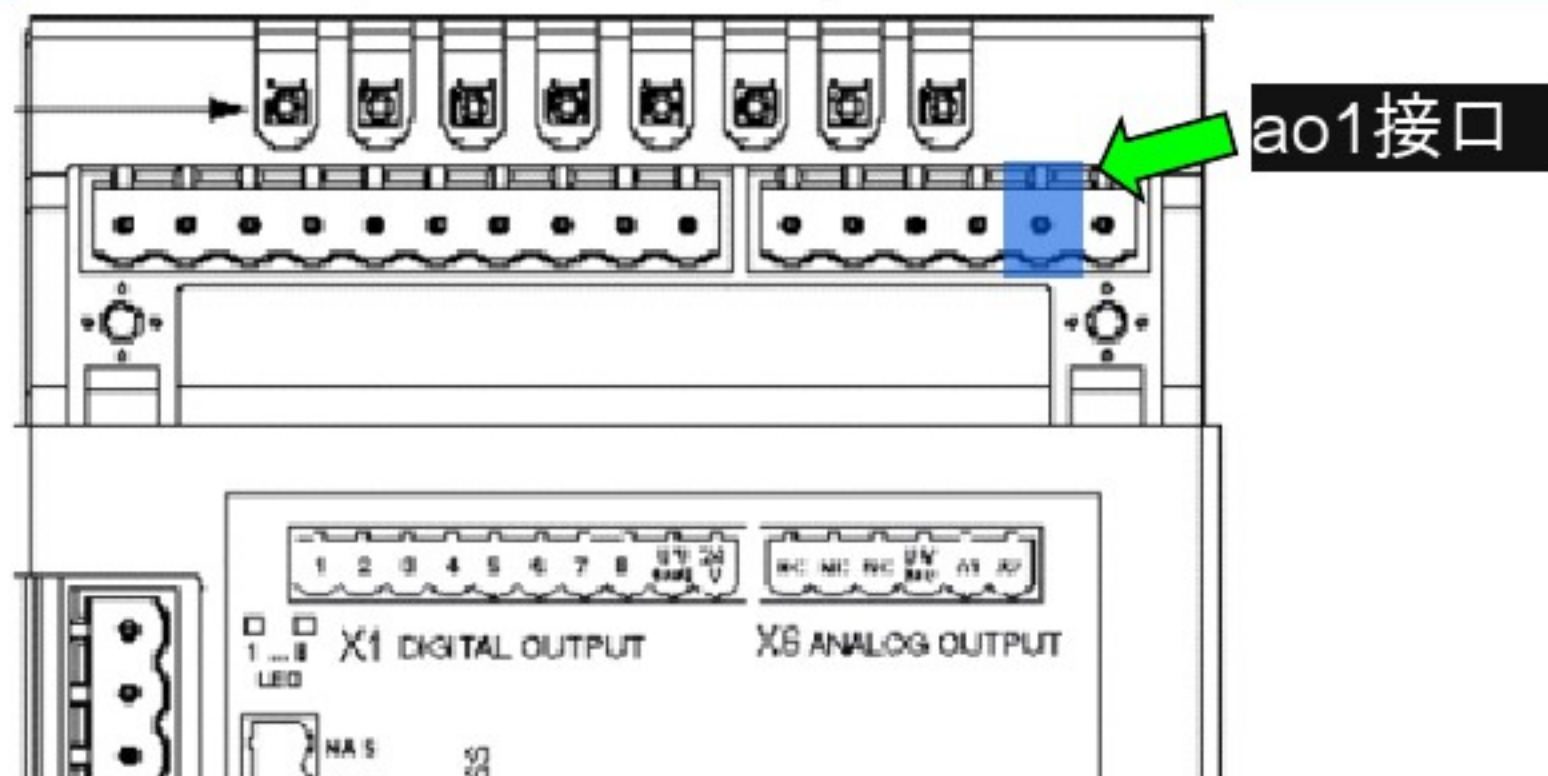
	地址33	地址34	地址35	地址36	十进制数
	1	2	4	8	
状态1	0	1	0	1	2+8=10
状态2	1	0	1	1	1+4+8=13

- 组输出信号就是将几个数字输出信号组合起来使用，用于输出BCD编码的十进制数。
- 此例中，go1占用地址33-36共4位，可以代表十进制数0-15。如此类推，如果占用地址5位的话，可以代表十进制数0-31。

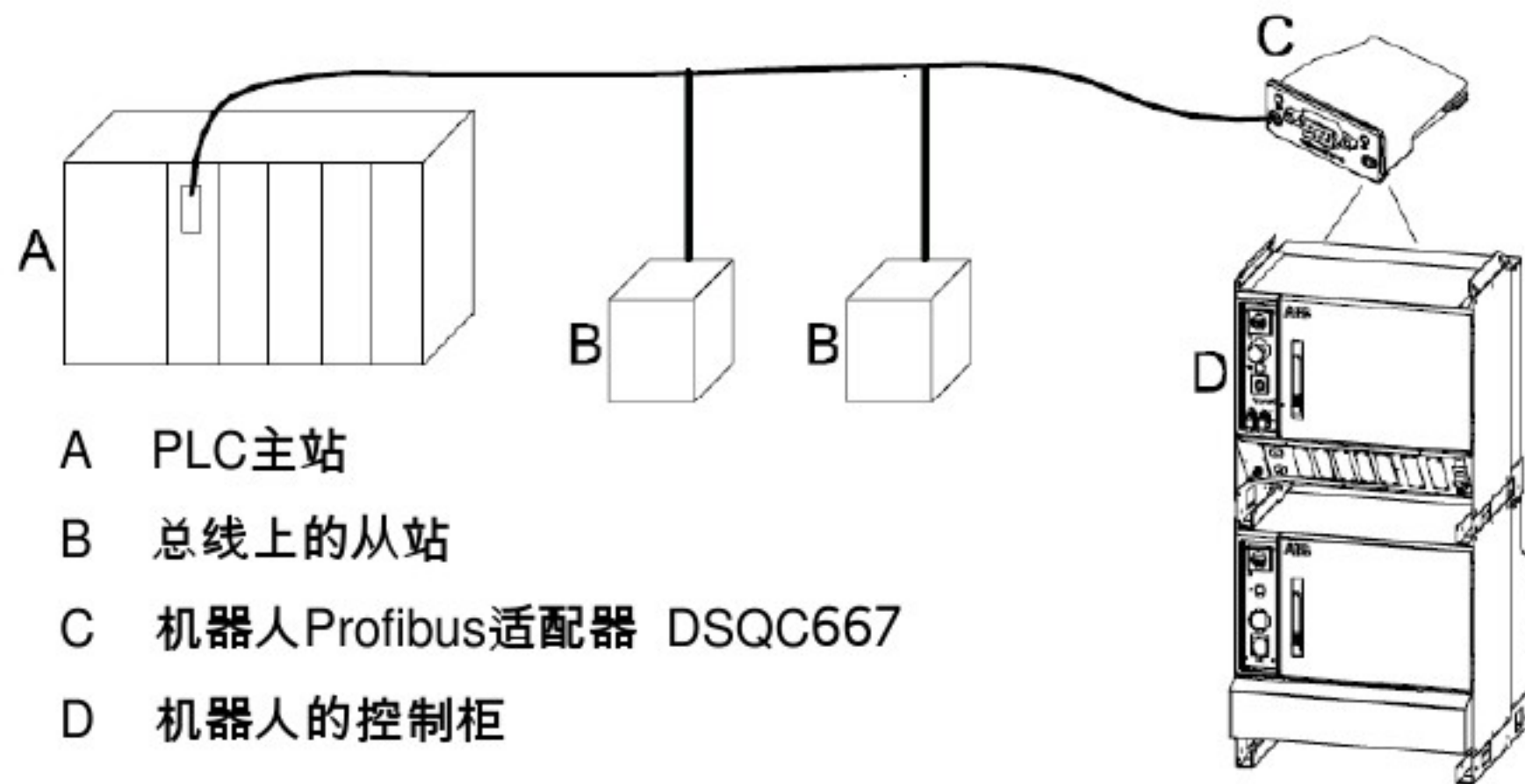


ABB机器人标准IO ao1 模拟输出信号

参数名称	设定值	说明
Name	ao1	设定模拟输出信号的名字
Type of Signal	Analog Output	设定信号的类型
Assigned to Unit	board10	设定信号所在的IO模块
Unit Mapping	0-15	设定信号所占用的地址
Analog Encoding Type	Unsigned	设定模拟信号属性
Maximum Logical Value	10	设定最大逻辑值
Maximum Physical Value	10	设定最大物理值 (V)
Maximum Bit Value	65535	设定最大位值



Profibus适配器的连接



A PLC主站

B 总线上的从站

C 机器人Profibus适配器 DSQC667

D 机器人的控制柜

- DSQC667模块是安装在电柜中的主机上，最多支持512个数字输入和512个数字输出。



- 除了通过ABB机器人提供的标准IO板进行与外围设备进行通讯，ABB机器人还可以使用DSQC667模块通过Profibus与PLC进行快捷和大数据量的通讯。

Profibus适配器的设定

参数名称	设定值	说明
Name	profibus8	设定IO板在系统中的名字
Type of Unit	DP_SLAVE	设定IO板的类型
Connected to Bus	Profibus1	设定IO板连接的总线
Profibus Address	8	设定IO板在总线中的地址

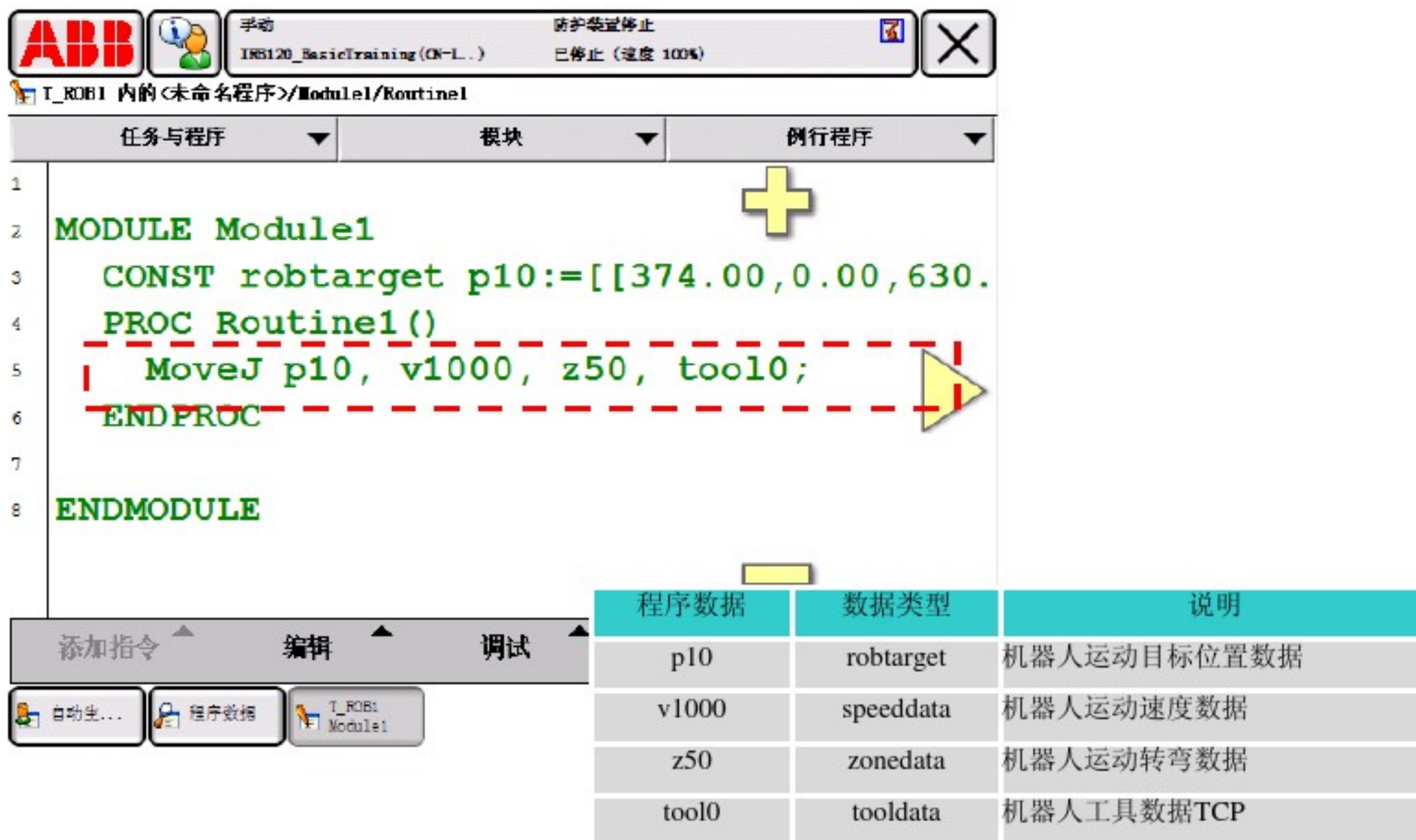
- 在完成了ABB机器人上的Profibus适配器模块设定以后，请在PLC端完成相关的操作：
- 将ABB机器人随机光盘的DSQC667配置文件（路径为\RobotWare 5.13\Utility\Fieldbus\Profibus\GSD\HMS_1811.GSD）在PLC的组态软件中打开。
- 在PLC的组态软件中找到“Anybus-CC PROFIBUS DP-V1”
- ABB机器人中设置的信号是与PLC端设置的信号是一一对应的。

系统输入/输出与IO信号的关联

- 我们可以将数字输入信号与系统的控制信号关联起来，就可以对系统进行控制（例如，电机开启、程序启动等等）。
- 系统的状态信号也可以与数字输出信号关联起来，将系统的状态输出给外围设备作控制之用。

什么是程序数据

- 程序数据是在程序模块或系统模块中设定值和定义一些环境数据。创建的程序数据由同一个模块或其它模块中的指令进行引用。如图4-1所示，虚线框中是一条常用的机器人关节运动的指令（MoveJ），并调用了4个程序数据。



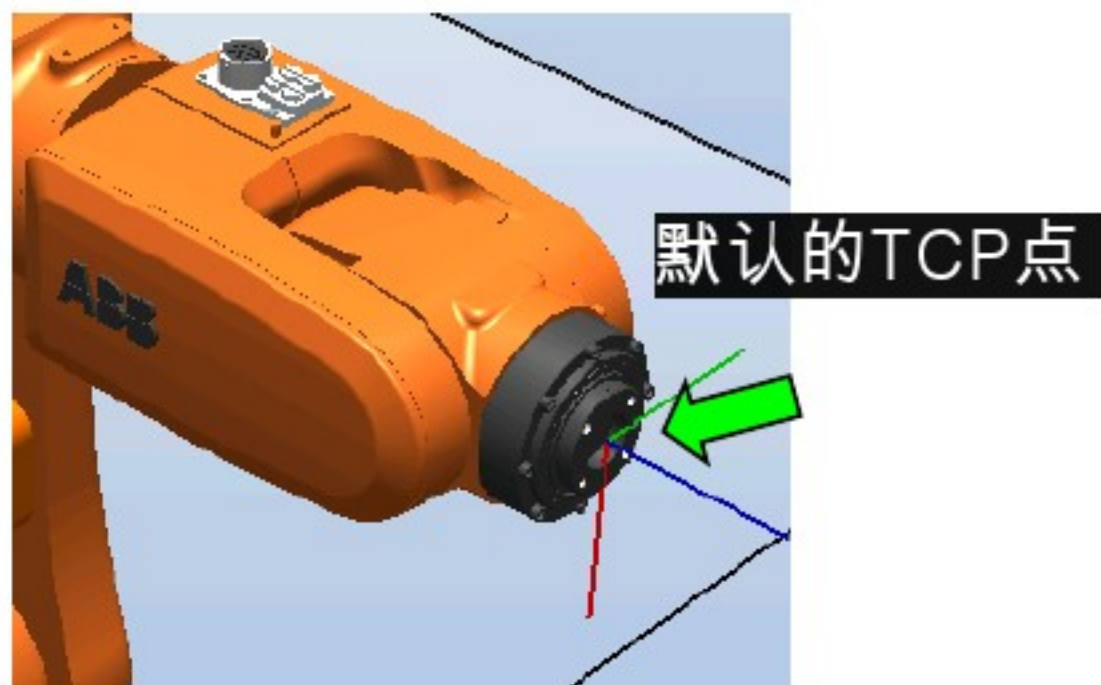
The screenshot shows the ABB robot software interface. The main window displays a routine named 'Routine1' within 'Module1'. The code is as follows:

```
1  
2 MODULE Module1  
3   CONST robtarget p10:=[[374.00,0.00,630.  
4   PROC Routine1()  
5   | MoveJ p10, v1000, z50, tool0;  
6   | ENDPROC  
7  
8 ENDMODULE
```

A yellow plus sign is positioned above the 'MoveJ' line, and a red dashed box highlights the 'MoveJ' line. A yellow arrow points from the 'MoveJ' line to the table below.

程序数据	数据类型	说明
p10	robtarget	机器人运动目标位置数据
v1000	speeddata	机器人运动速度数据
z50	zonedata	机器人运动转弯数据
tool0	tooldata	机器人工具数据TCP

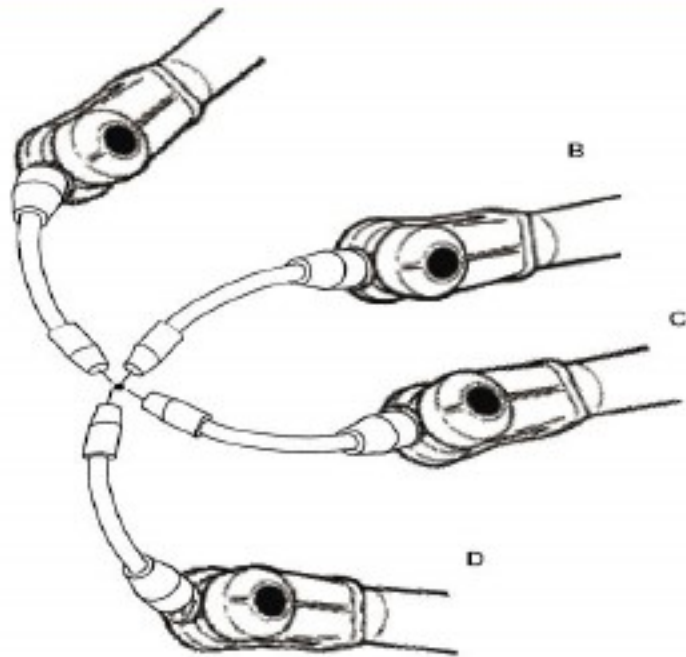
工具数据的原理与设定






- 工具数据TOOLDATA是用于描述安装在机器人第六轴上的工具的TCP，重量，重心等参数数据。
- 执行程序时，机器人就是将TCP移至编程位置，程序中所描述的速度与位置就是TCP点在对应工件坐标中的速度与位置。
- 所有机器人在手腕处都有一个预定义工具坐标系，该坐标系被称为tool0。这样就能将一个或多个新工具坐标系定义为tool0的偏移值。

TCP设定原理

- 首先在机器人工作范围内找一个非常精确的固定点作为参考点。
- 然后在工具上确定一个参考点（最好是工具的中心点TCP）。
- 通过之前学习到的手动操纵机器人的方法，去移动工具上的参考点以最少四种不同的机器人姿态尽可能与固定点刚好碰上。（为了获得更准确的TCP，我们在以下的例子中使用六点法进行操作，第四点是用工具的参考点垂直于固定点，第五点是工具参考点从固定点向将要设定为TCP的X方向移动，第六点是工具参考点从固定点向将要设定为TCP的Z方向移动。）
- 机器人就可以通过这四个位置点的位置数据计算求得TCP的数据，然后TCP的数据就保存在TOOLDATA这个程序数据中被程序进行调用。



- TCP取点数量的区别：
- 4点法，不改变tool0的坐标方向
- 5点法，改变tool0的Z方向
- 6点法，改变tool0的X和Z方向（在焊接应用最为常用）
- 前三个点的姿态相差尽量大些，这样有利于TCP精度的提高。



手动
防护装置停止


IRB120_BasicTraining(CN-L...)

已停止 (速度 100%)

程序数据 - tooldata - 定义 - 工具坐标定义

计算结果

工具坐标: tool1

点击“确定”确认结果，或点击“取消”重新定义源数据。

		1 到 6 共 7
方法	TCP	
最大误差	2.11913 毫米	
最小误差	1.107085 毫米	
平均误差	0.7712035 毫米	
X:	31.62189 毫米	
Y:	-0.1567581 毫米	

确定

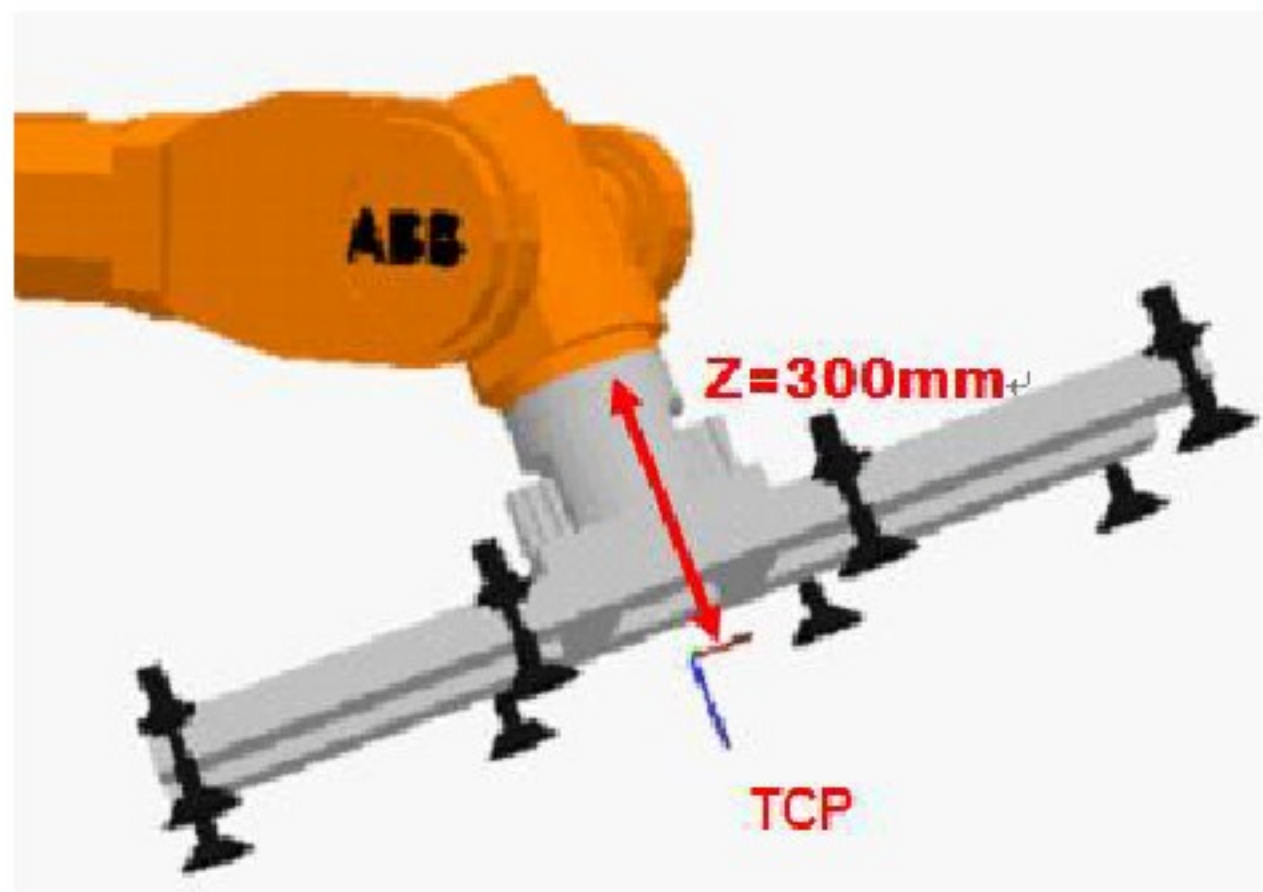
取消

 自动生...
  手动操纵

ROB_1



搬运运用的工具数据的设定

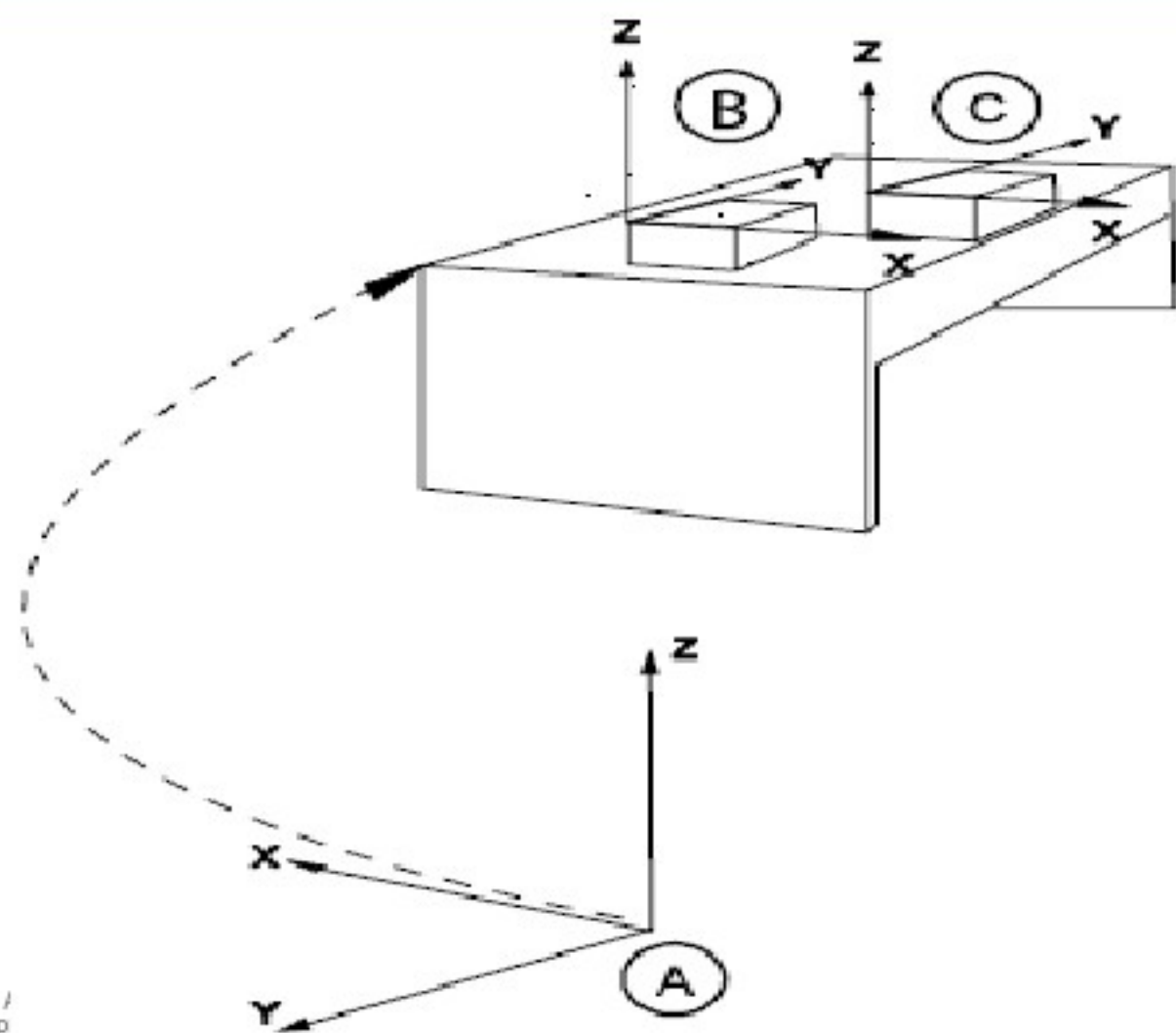


- 以图中的搬运薄板的真空吸盘夹具为例，重量是25kg，重心在默认tool0的Z正方向偏移250mm，TCP点设定在吸盘的接触面上，从默认tool0上的Z正方向偏移了300mm。

工件坐标WOBJDATA的设定 1

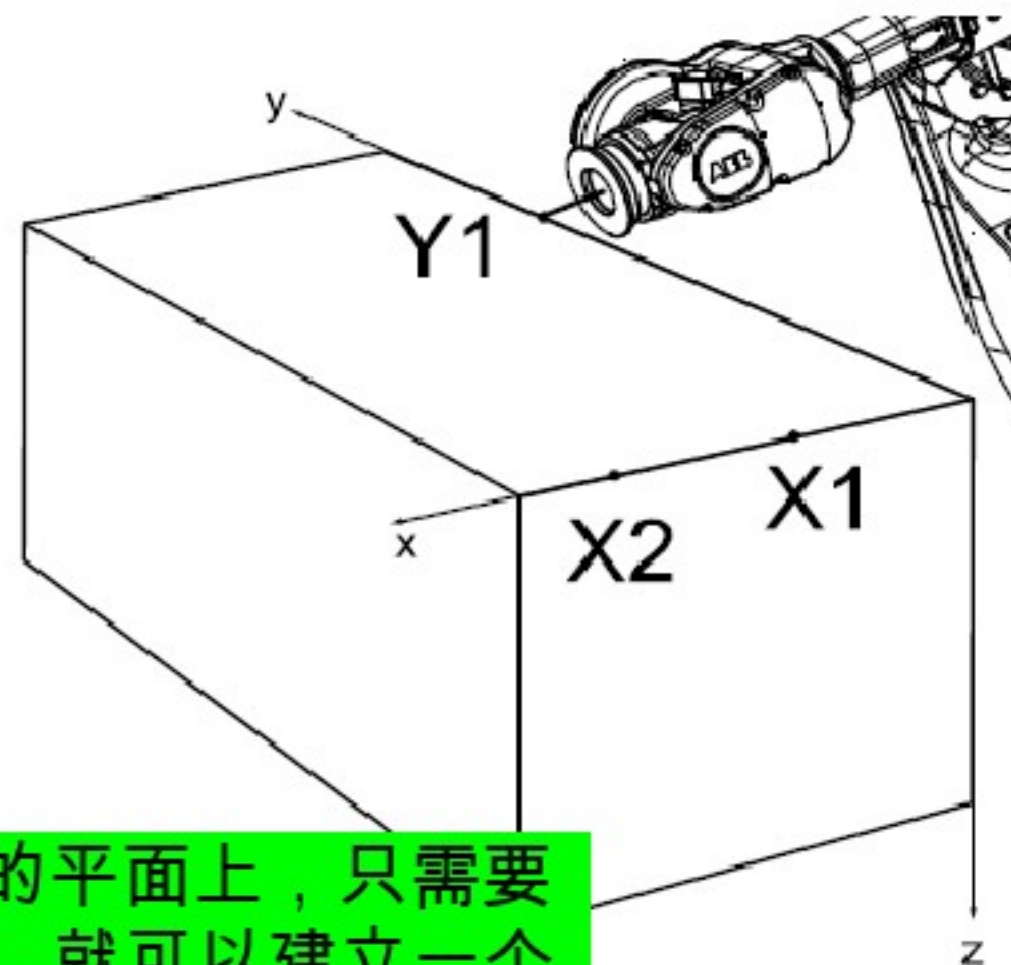
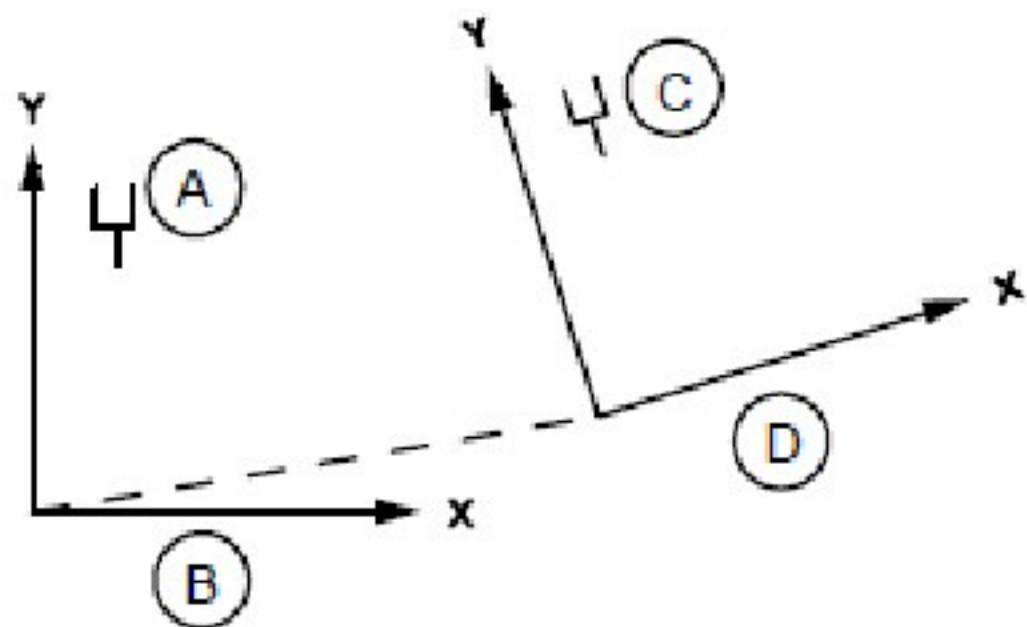
工件坐标系对应工件：它定义工件相对于大地坐标系（或其它坐标系）的位置。机器人可以拥有若干工件坐标系，或者表示不同工件，或者表示同一工件在不同位置的若干副本。

- 您对机器人进行编程时就是在工件坐标系中创建目标和路径。这带来很多优点：
- 重新定位工作站中的工件时，您只需更改工件坐标系的位置，所有路径将即刻随之更新。
- 允许操作以外轴或传送导轨移动的工件，因为整个工件可连同其路径一起移动。



- A是机器人的大地坐标，为了方便编程为第一个工件建立了一个工件坐标B，并在这个工件坐标B进行轨迹编程。
- 如果台子上还有一个一样的工件需要走一样的轨迹，那你只需要建立一个工件坐标C，将工件坐标B中的轨迹复制一份，然后将工件坐标从B更新为C，则无需对一样的工件重复的轨迹编程了。

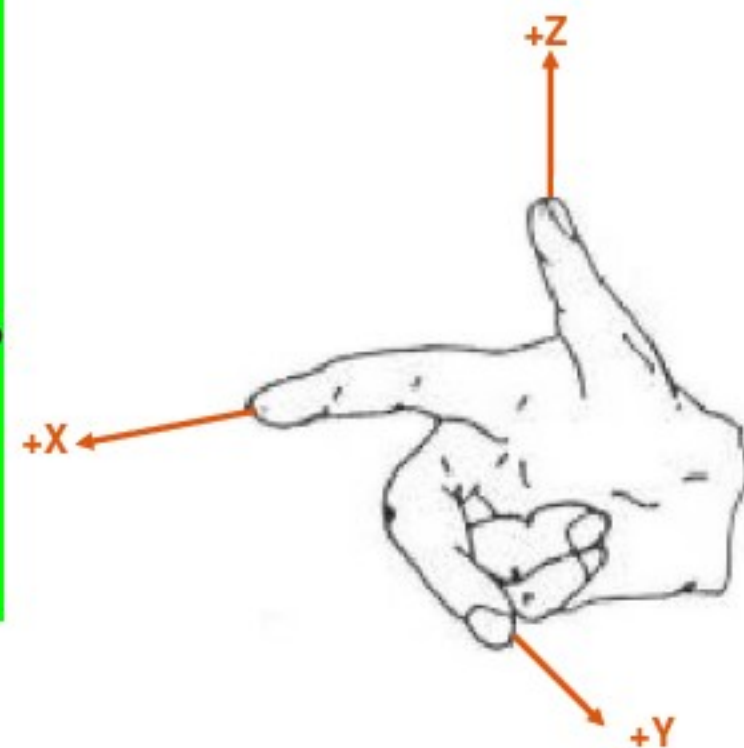
工件坐标WOBJDATA的设置2







在工件坐标B中对A对象进行了轨迹编程。如果工件坐标的位置变化成工件坐标D后，只需在机器人系统重新定义工件坐标D，则机器人的轨迹就自动更新到C了，不需要再次轨迹编程了。因A相对于B，C相对于D的关系是一样，并没有因为整体偏移而发生变化。

在对象的平面上，只需要定义三个点，就可以建立一个工件坐标。

- X1点确定工件坐标的原点
- X1---X2确定工件坐标X正方向。
- Y1确定工件坐标Y正方向。
- 工件坐标符合右手定则





手动 电机开启  

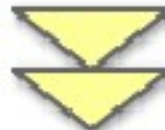

IRB120_BasicTr.. (ON-L-0313577) 已停止 (速度 100%)

程序数据 - wobjdata - 定义 - 工件坐标定义

计算结果

工件坐标: wobj1

点击“确定”确认结果，或点击“取消”重新定义源数据。

		1 到 6 共 9
用户方法:	WobjFrameCalib	
X:	280.4269 毫米	
Y:	-249.9521 毫米	
Z:	604.6683 毫米	
四个一组 1	0.9999960064888	 
四个一组 2	-2.44095986090542E-06	
确定 取消		

 自动生...
  手动操纵



有效载荷LOADDATA的设定

- 对于搬运应用的机器人，应该正确设定夹具的重量、重心
TOOLDATA以及搬运对象的重量和重心数据LOADDATA。

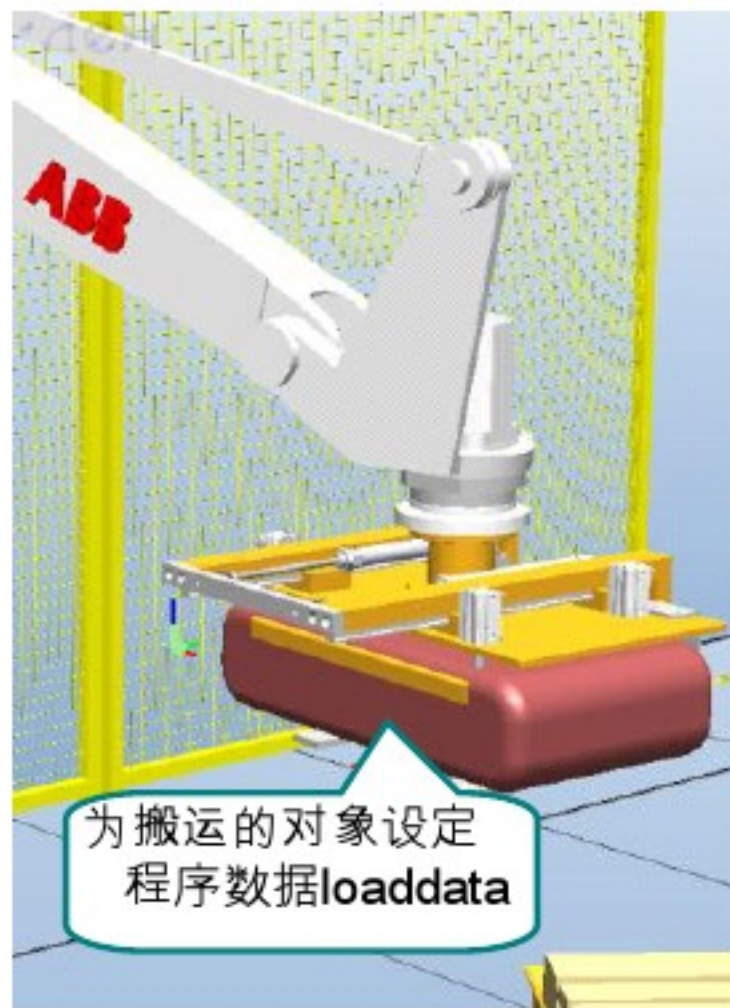


ABB 手动 防护装置停止
IMS2400 (CN-L-0813377) 已停止 (速度 100%)

NewProgramName - T_ROB1/MainModule/main

任务与程序 模块 例行程序

```
3 PROC main()  
4   Set do1;  
5   GripLoad load1;  
6   MoveJ *, v1000, z50, tool1;  
7   MoveL *, v1000, z50, tool1;  
8   MoveL *, v1000, z50, tool1;  
9   Reset do1;  
10  GripLoad load0;  
11 ENDPROC
```

夹具夹紧。
指定当前搬运对象的重量和重心load1。

夹具松开。
将搬运对象清除为load0

添加指令 编辑 调试 修改位置 显示声明

T_ROB1 : MainModule

什么是RAPID程序指令

- RAPID 程序中包含了一连串控制机器人的指令，执行这些指令可以实现需要的操作。
- 应用程序是使用称为 RAPID 编程语言的特定词汇和语法编写而成。RAPID 是一种英文编程语言，所包含的指令可以移动机器人、设置输出、读取输入，还能实现决策、重复其它指令、构造程序、与系统操作员交流等。

RAPID程序			
程序模块1	程序模块2	程序模块3	系统模块
程序数据	程序数据	程序数据
主程序main	例行程序	例行程序
例行程序	中断程序	中断程序
中断程序	功能	功能
功能		

- 关于RAPID程序的架构说明：
- RAPID程序是由程序模块与系统模块组成。一般地，我们只通过新建程序模块来构建机器人的程序，而系统模块多用于系统方面的控制之用。
- 可以根据不同的用途创建多个程序模块，如专门用于主控制的程序模块，用于位置计算的程序模块，用于存放数据的程序模块，这样的目的在于方便归类管理不同用途的例行程序与数据。
- 每一个程序模块包含了程序数据，例行程序，中断程序和功能四种对象，但不一定在一个模块都有这四种对象的存在，程序模块之间的数据，例行程序，中断程序和功能是可以互相调用的。
- 在RAPID程序中，只有一个主程序main，并且存在于任意一个程序模块中，并且是作为整个RAPID程序执行的起点。

常用RAPID指令

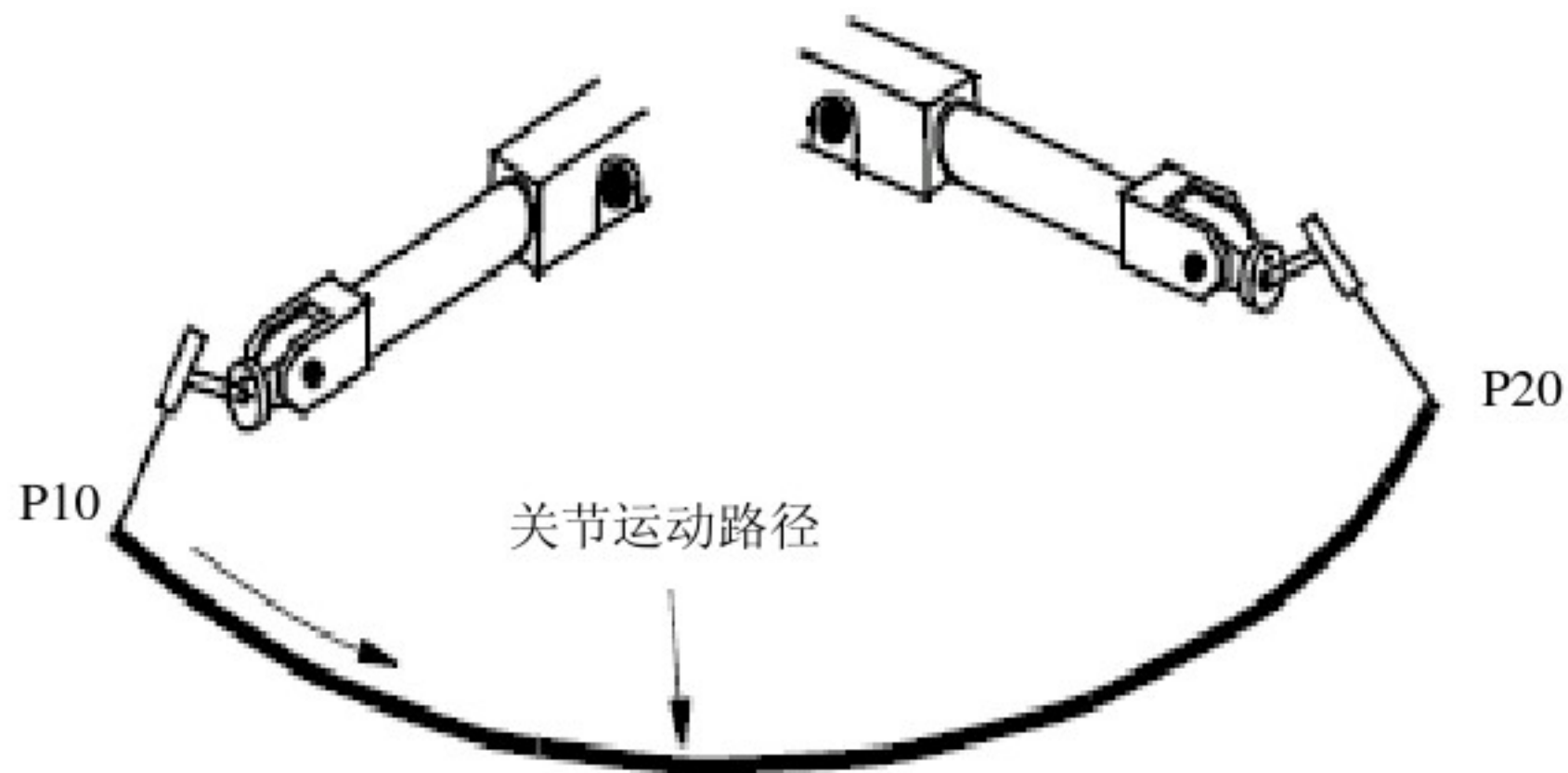
- “:=”赋值指令是用于对程序数据进行赋值，赋值可以是一个常量或数学表达式。
- 我们就以添加一个常量赋值与数学表达式赋值进行说明此指令的使用：
- 常量赋值：reg1 := 5;
- 数学表达式赋值：reg2 := reg1+4;

常用RAPID指令

- 机器人运动指令
- 机器人在空间中进行运动主要是四种方式，关节运动（MOVEJ），线性运动（MOVEL），圆弧运动（MOVEC）和绝对位置运动（MOVEABSJ）。

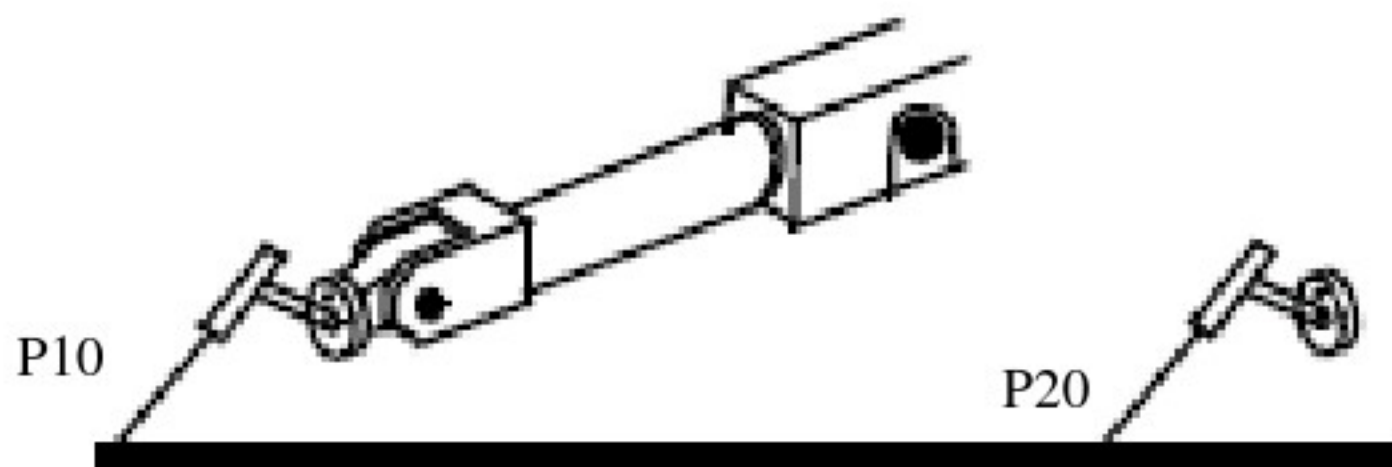
常用RAPID指令

- 关节运动指令是在对路径精度要求不高的情况，机器人的工具中心点TCP从一个位置移动到另一个位置，两个位置之间的路径不一定是直线。



常用RAPID指令

- 线性运动是机器人的TCP从起点到终点之间的路径始终保持为直线，一般如焊接，涂胶等应用对路径要求高的场合进行使用此指令。



- 线性运动路径

常用RAPID指令

中断程序TRAP

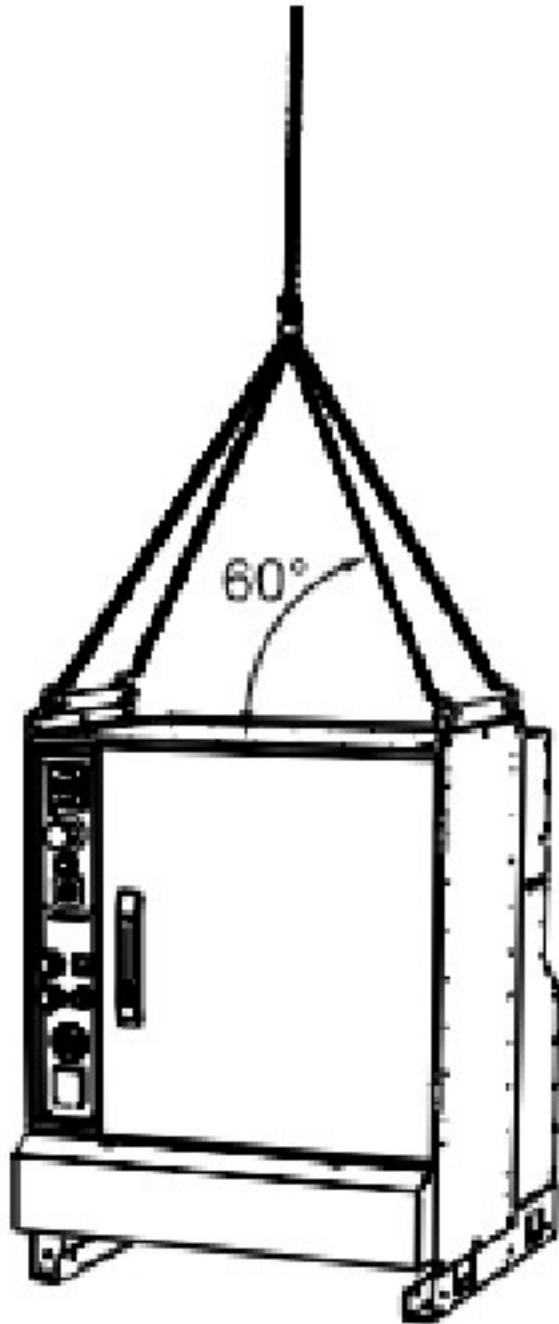
- RAPID程序的执行过程中，如果发生需要紧急处理的情况，这就要机器人中断当前的执行，程序指针PP马上跳转到专门的程序中对紧急的情况进行相应的处理，结束了以后程序指针PP返回到原来被中断的地方，继续往下执行程序。那么，专门用来处理紧急情况的专门程序，我们就叫做中断程序(TRAP)。
- 中断程序经常会用于出错处理，外部信号的响应这种实时响应要求高的场合。
- 比如我们需要对一个传感器的信号进行实时的监控为例编写一个中断程序：
- 在正常的情况下，di1的信号为0.
- 如果di1的信号从0变为1的话，我们就对reg1数据进行加1的操作。

中断指令

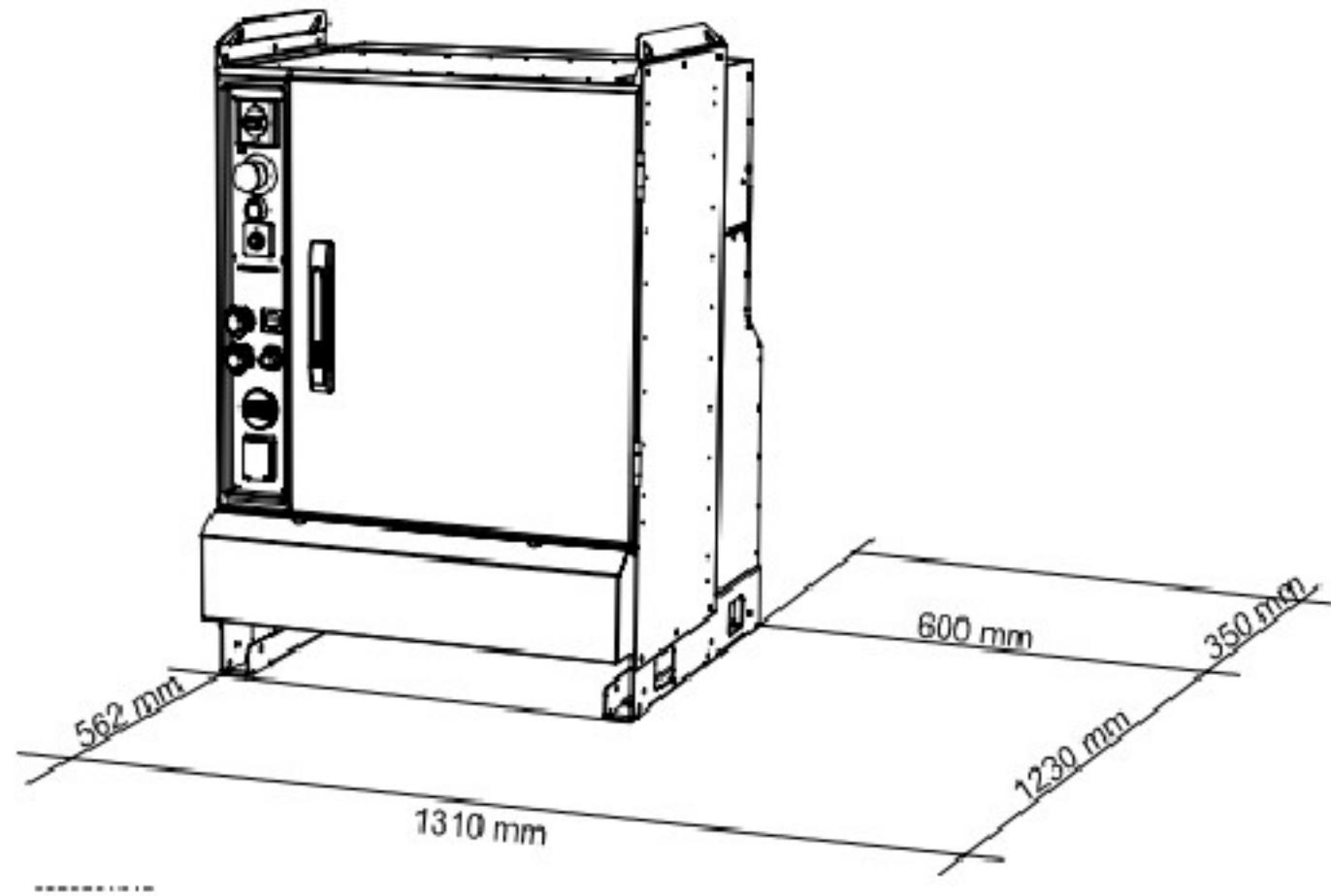
指令	说明
IDelete	取消指定的中断
CONNECT	连接一个中断符号到中断程序
ISignalDI	使用一个数字输入信号触发中断

ABB机器人的控制柜 1

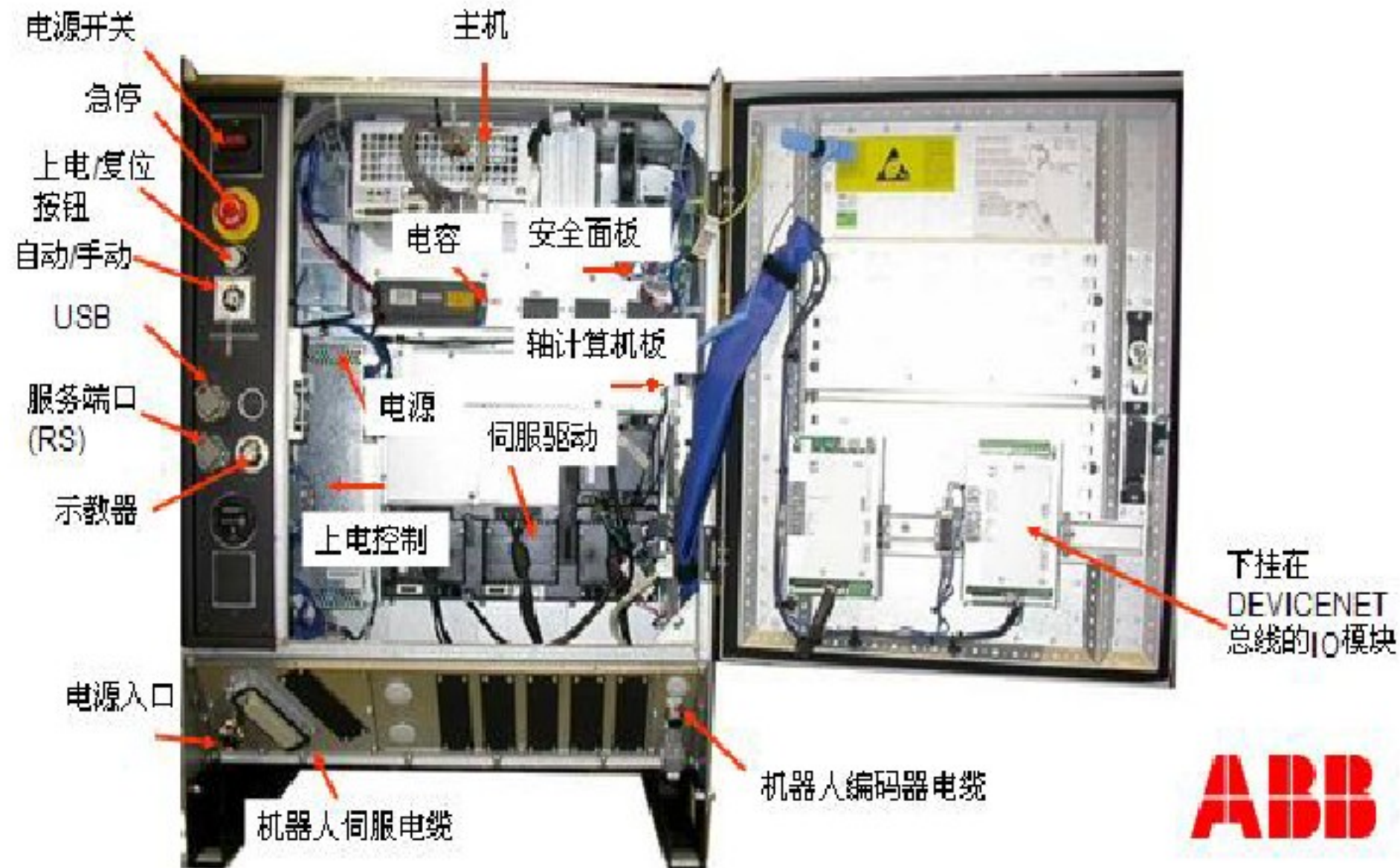
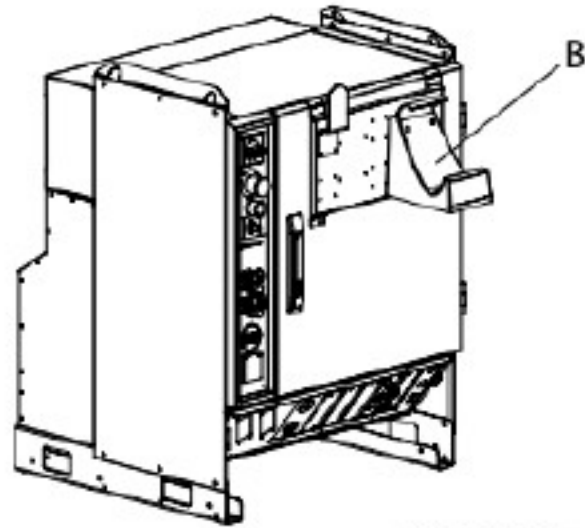
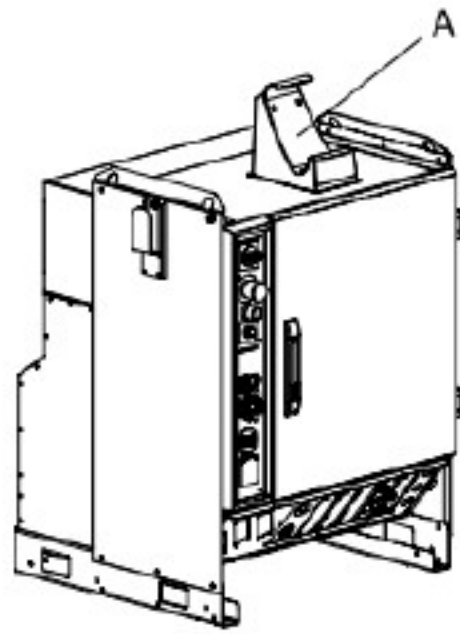
■ 吊装示意图



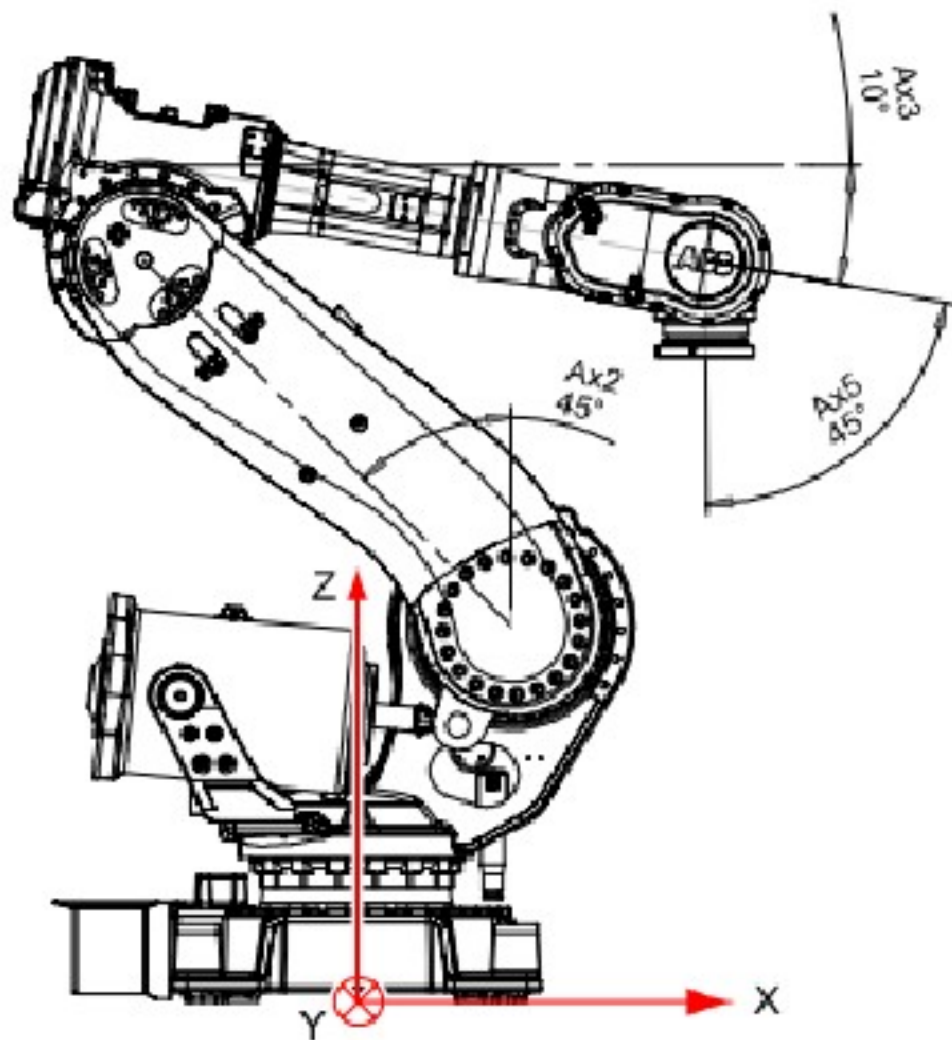
■ 布置示意图



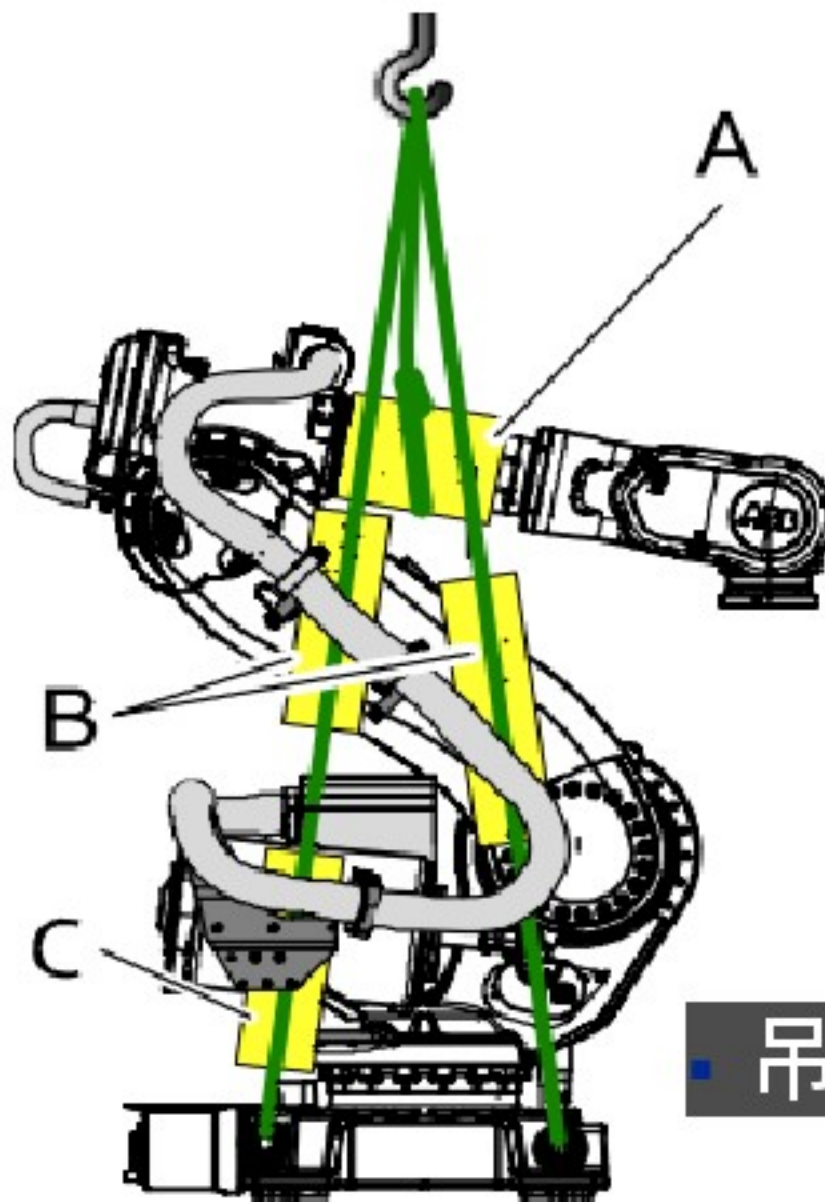
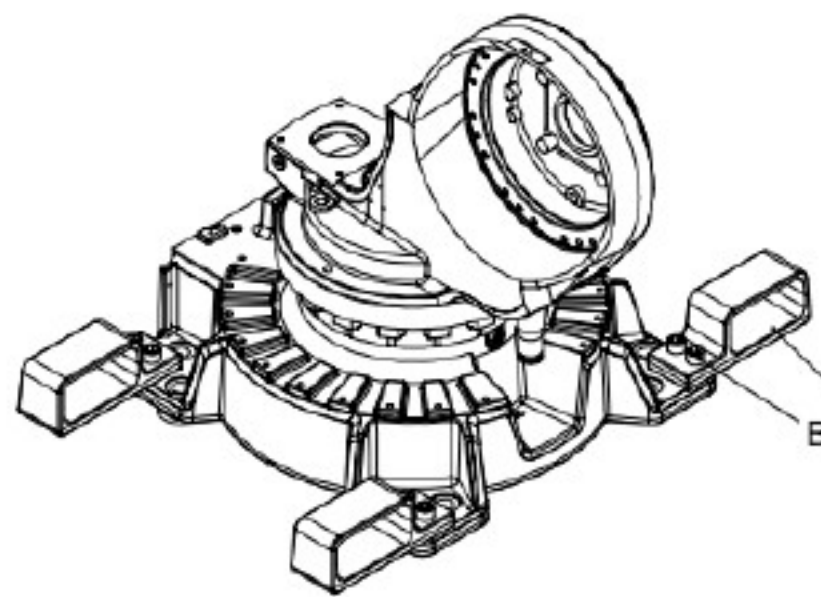
ABB机器人的控制柜 2



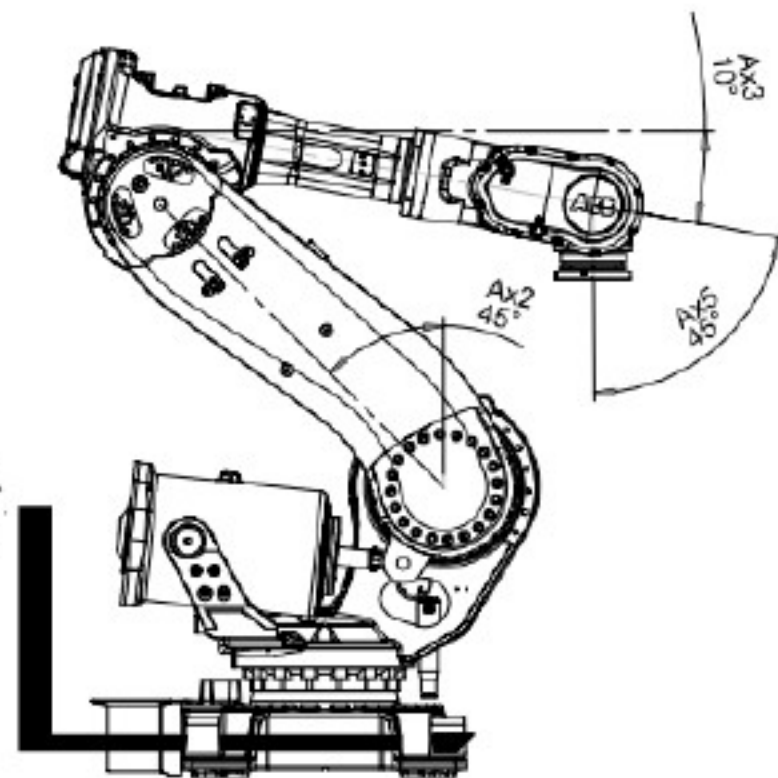
ABB机器人的本体



搬运姿态图

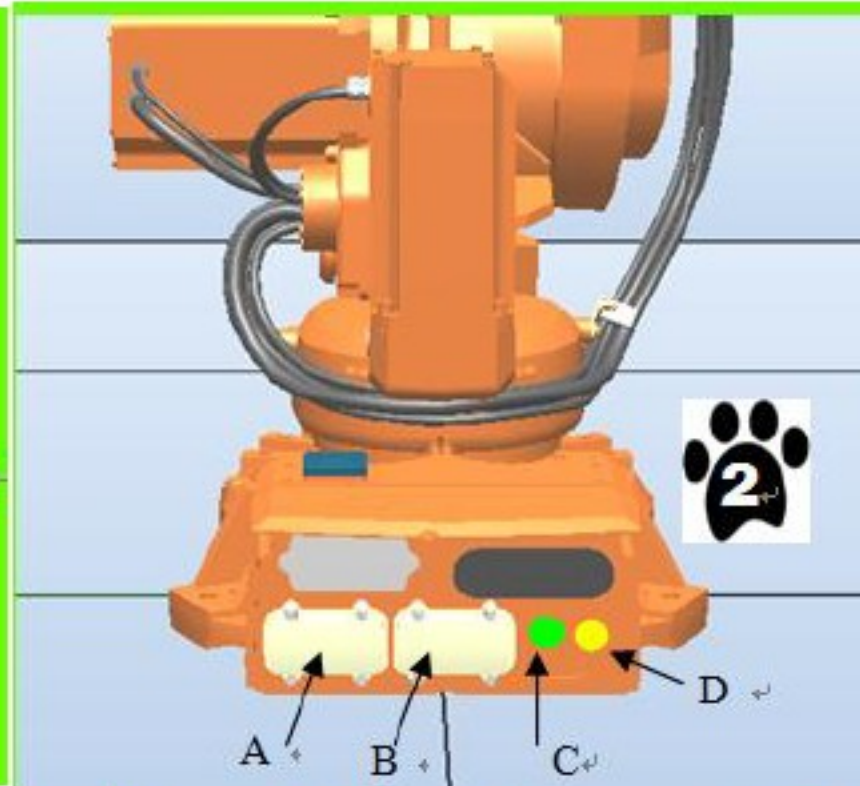


吊装搬运姿态图

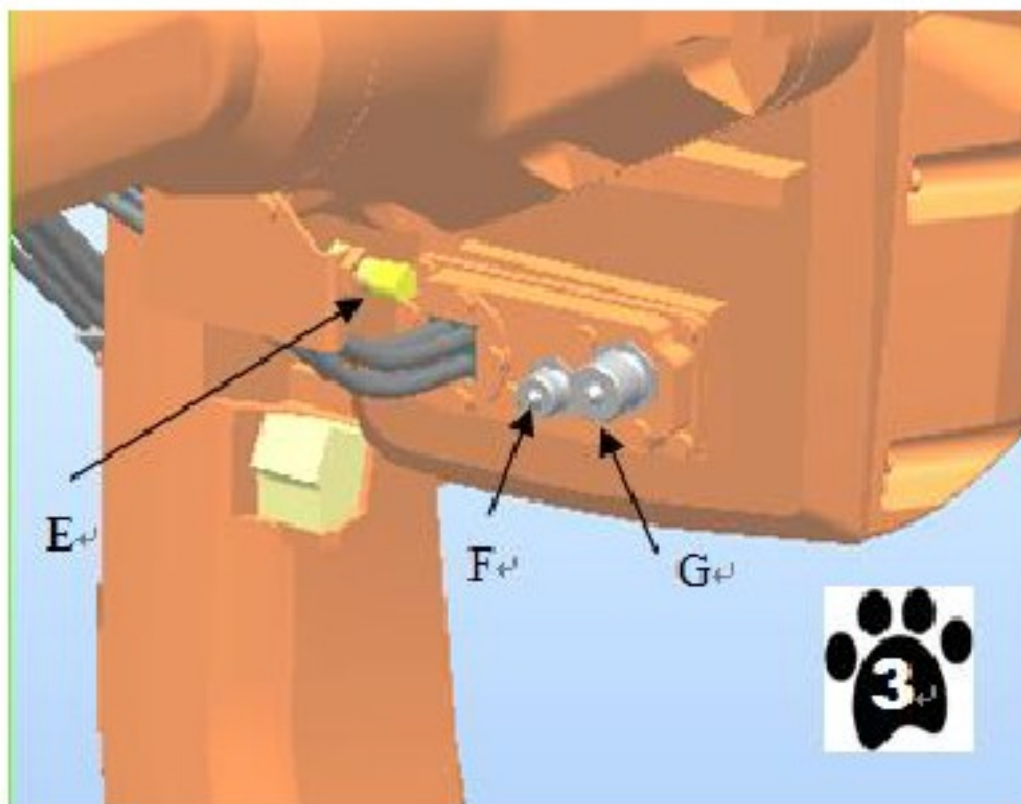


叉车搬运姿态图

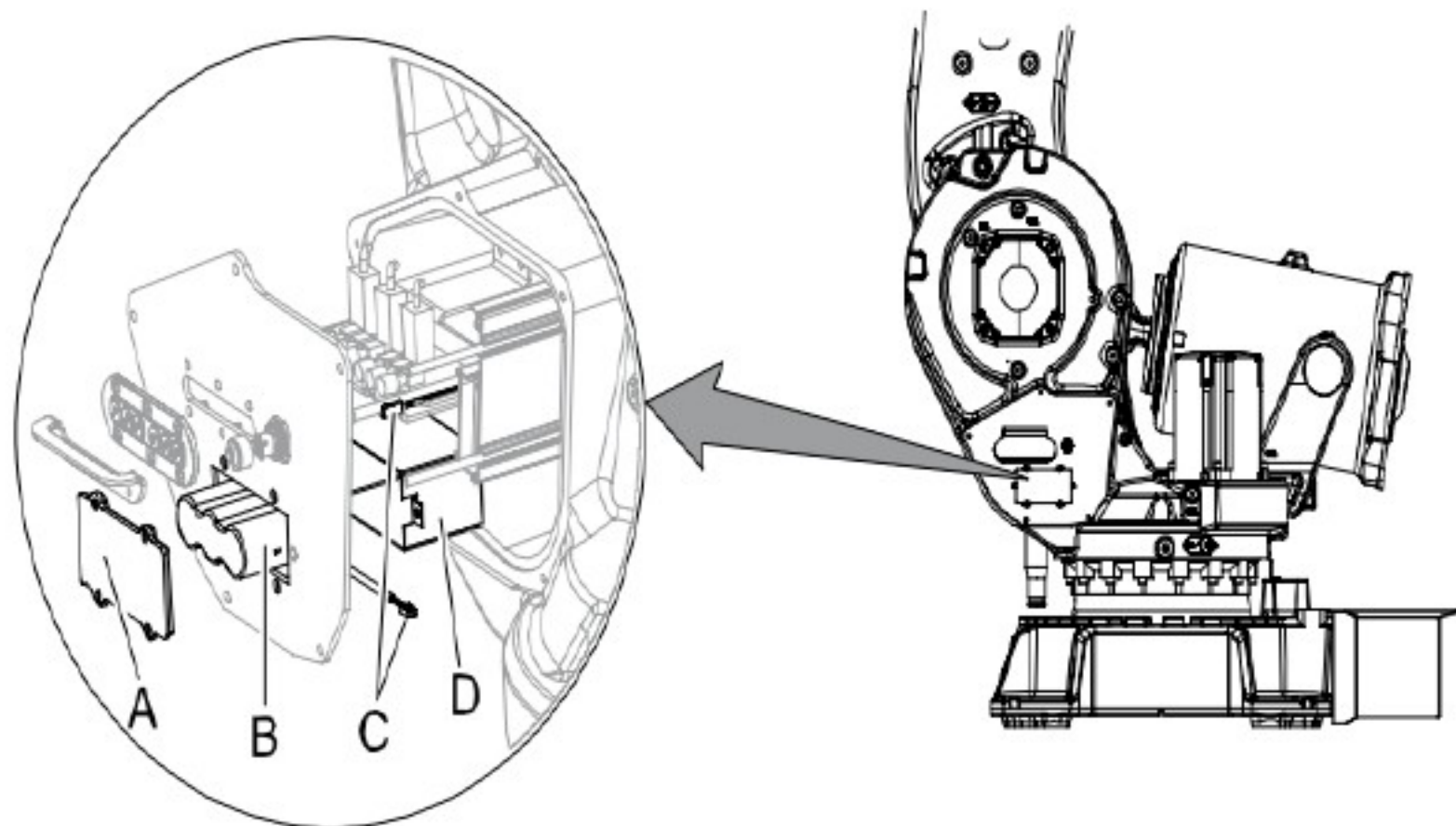
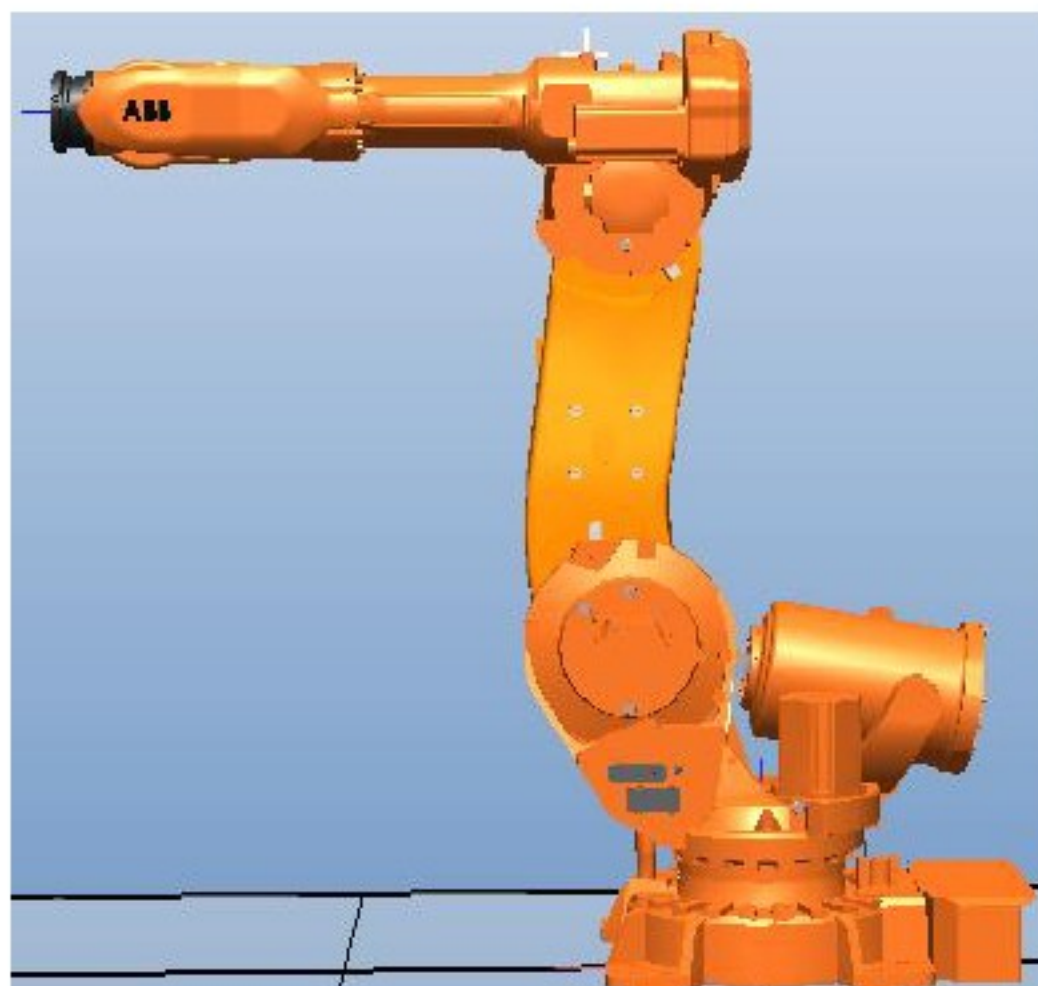
ABB机器人IRB4600本体的连接接口说明



- ABB机器人IRB4600本体的连接接口说明，不同的机器人的连接接口有所差异，具体请查看ABB机器人随机光盘手册。



更换SMB电池



ABB机器人在关掉控制柜主电源后，六个轴的位置数据是由电池提供电力进行保存的，所以在电池即将耗尽之前，需要对其进行更换，否则，每次主电源断电后再次上电，就要进行机器人转数计数器更新的操作。

机器人一般安装调试步骤

序号	安装调试内容	参考内容
1	将机器人本体与控制柜吊装到位	第42、43课
2	机器人本体与控制柜之间的电缆连接	第44课
3	示教器与控制柜连接	第44课
4	接入主电源	第44课
5	检查主电源正常后，上电	第44课
6	机器人六个轴机械原点的校准操作	第10课
7	I/O信号的设定	第11-16课
8	安装工具与周边设备	第20-22课
9	编程调试	第35-39课
10	投入自动运行	第39课

机器人的重新启动的操作

ABB 机器人系统可以长时间无人操作。无须定期重新启动运行的系统。

- 以下情况下需重新启动机器人系统：
- 安装了新的硬件。
- 更改了机器人系统配置参数。
- 出现系统故障(SYSFAIL)。
- RAPID程序出现程序故障。

重新启动类型	说明
热启动	使用当前的设置重新启动当前系统。
关机	关闭主机。
B-启动	重启并尝试回到上一次的无错状态。 一般地，当出现系统故障时使用。
P-启动	重启并将用户加载的RAPID程序全部删除。
I-启动	重启并将机器人系统恢复到出厂状态。

ABB机器人必备工具-----梅花L型套装扳手



ABB机器人必备工具-----梅花L型套装扳手

