

ABB 机器人基础应用培训

智能制造学院
工程训练中心

主要内容

1

ABB机器人操作安全事项

2

ABB机器人简介及硬件连接

3

机器人基本操作

4

Rapid程序指令介绍

5

应用案例

ABB机器人操作安全事项

- ABB机器人操作安全事项

安全事项:

保持安全距离

▶在对机器人进行操作时，应与机器人保持足够的安全距离

手动操作

▶在保护空间内有工作人员时，用手动操作机器人系统

随身携带示教器

▶进入保护空间时，牢记随身携带示教器，以免被人误操作

佩戴静电手环

▶对电气柜中的电气元件进行操作时，应佩戴静电手环

关闭总电源

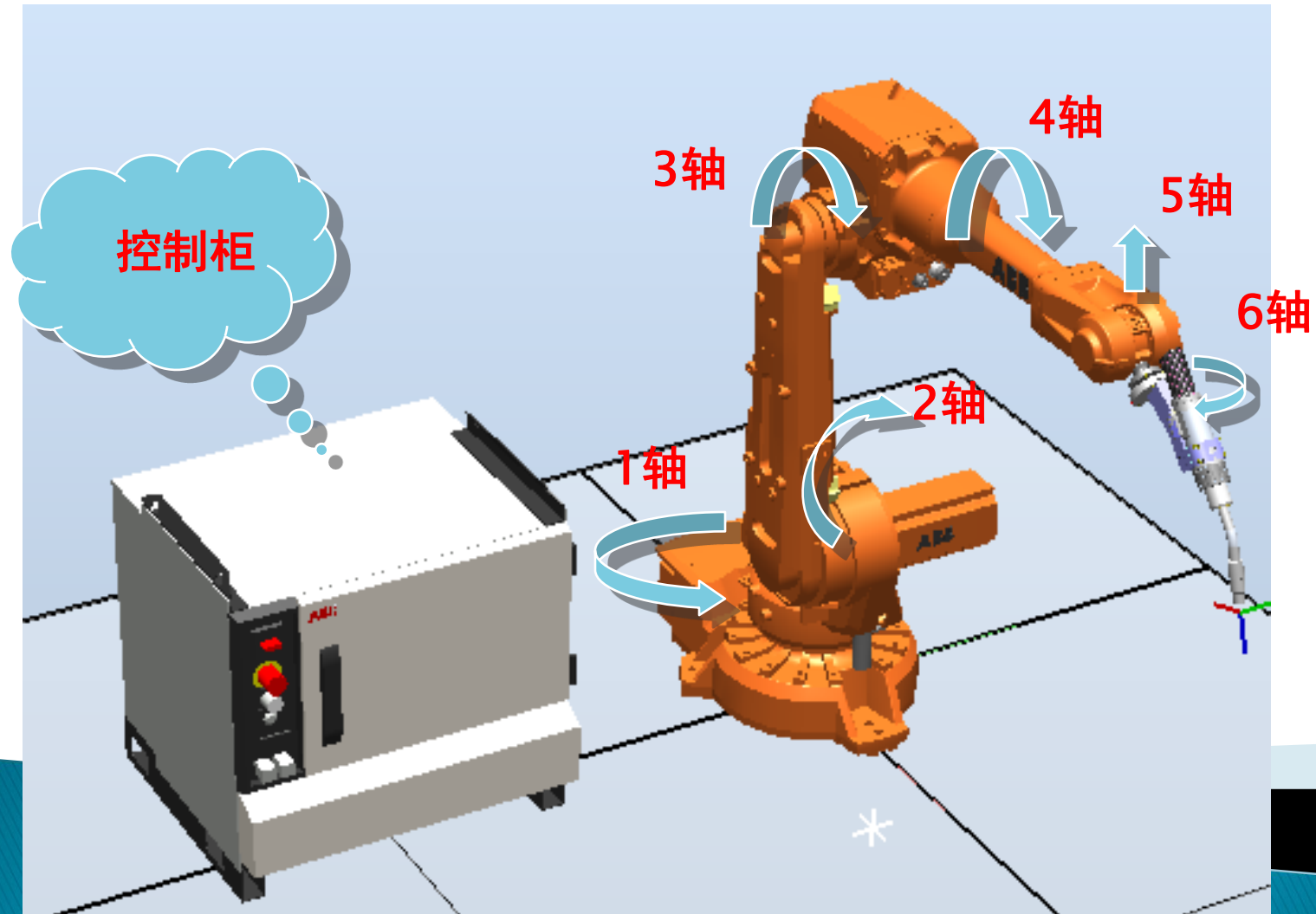
▶在对机器人进行安装，维修和保养时要将总电源关闭

灭火

▶用二氧化碳灭火器灭火

ABB机器人简介及硬件连接

- ABB机器人简介



- ABB机器人简介

1

IRB1410



负载/kg	5
RP/mm	0.025

2

IRB1600

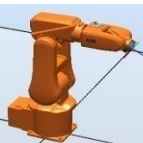


负载/kg	8
RP/mm	0.04

激光焊接常用机器人

5

IRB120



负载/kg	6
RP/mm	0.01

4

IRB4600



负载/kg	60
RP/mm	0.05

3

IRB2600



负载/kg	20
RP/mm	0.01

- ABB机器人硬件连接

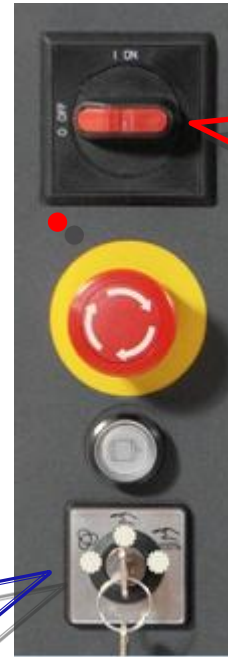
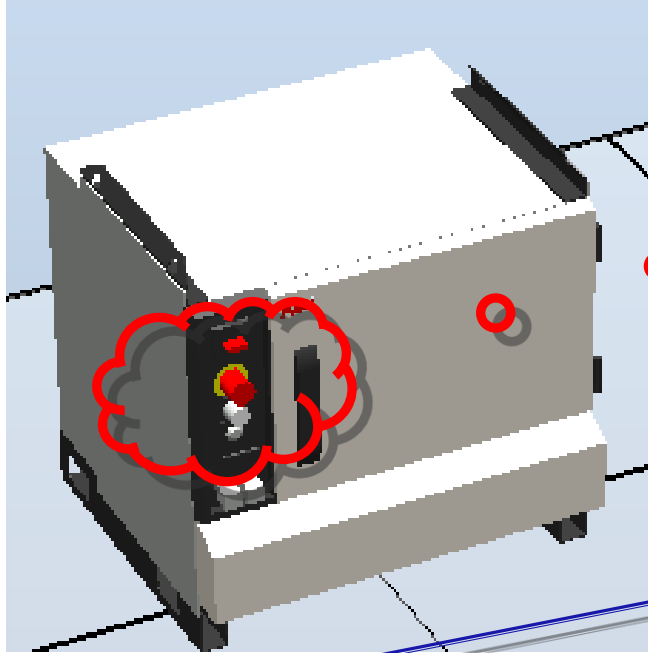
硬件连接:

机器人本体与控制柜之间的硬件连接主要是电动机动力电缆与转数计数器电缆、用户电缆的连接。

序号	名称	电缆标识 (机柜端)	机柜接口	机器人底座接口	电缆标识 (底座端)	上臂壳接口
1	电机动力电缆	XP1	XS1	R1. MP	R1. MP	-
2	转数计数器电缆	XP2	XS2	R1. SMB	R1. SMB	-
3	用户电缆	XP13	XS5/XS13	R1. CP/CS	R1. CP/CS	R2. CP

机器人基本操作

- ABB机器人运行模式



在确认输入电压正常后，
打开电源开关。



ABB机器人的电源电压：
IRB120为220V，其他
均为380V的三相电



自动运行模式



手动运行模式：最大速度为250mm/s



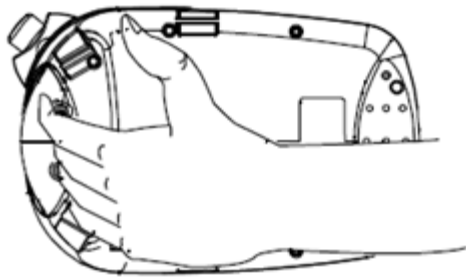
手动全速运行模式：100%

通过钥匙开关实现机器人运行模式的切换

- 示教器介绍



- A: 连接电缆
- B: 触摸屏
- C: 急停按钮
- D: 手动操纵摇杆
- E: USB口
- F: 使能器按键
- G: 触摸屏用笔
- H: 示教器复位按钮



使能器按钮分两档，手动状态下，按下第一档，机器人将处于电机开启状态，按下第二档，机器人又将处于防护装置停止状态。

- 示教器介绍

ABB菜单

操作员窗口

状态栏

关闭按钮

可编程按钮

快速选择机械单元

重复定位或线性运动模式切换

1-3轴或4-6轴关节运动模式切换

增量模式切换

快速设置菜单

任务栏

ABB



手动

test1 (A-A01-6040)

防护装置停止

已停止 (速度 100%)



HotEdit



输入输出



手动操纵



自动生产窗口



程序编辑器



程序数据



注销



备份与恢复



校准



控制面板



事件日志



FlexPendant



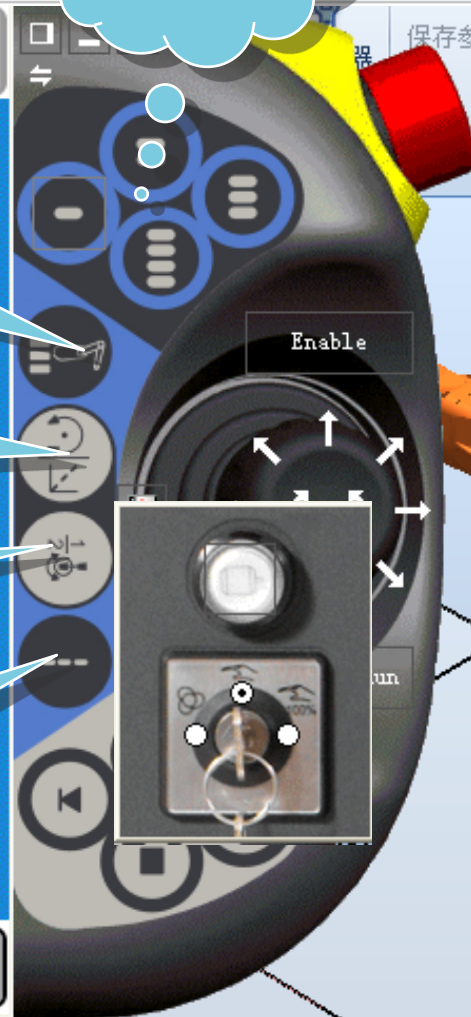
系统



重新启动

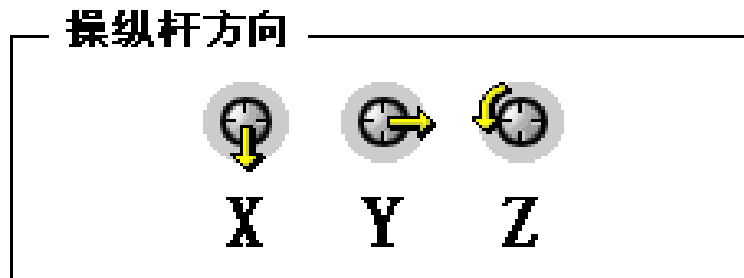


ROB_1

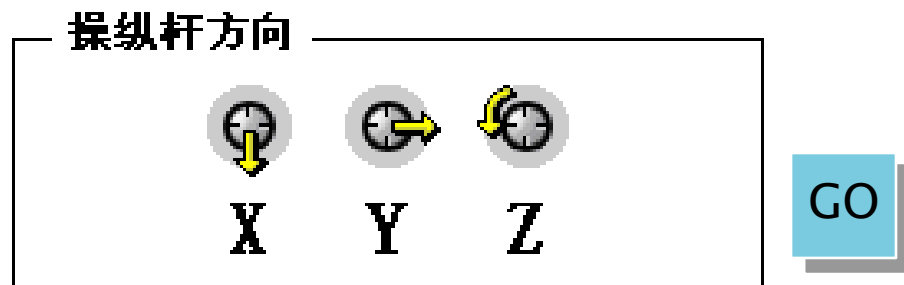


- 机器人手动操纵

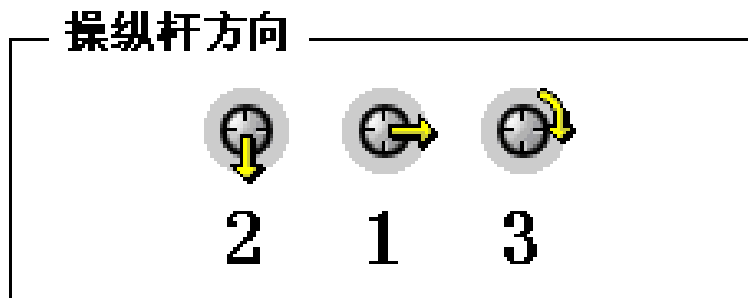
线性运动



重复定位运动



单轴运动



- 输入输出介绍

IO模块单元



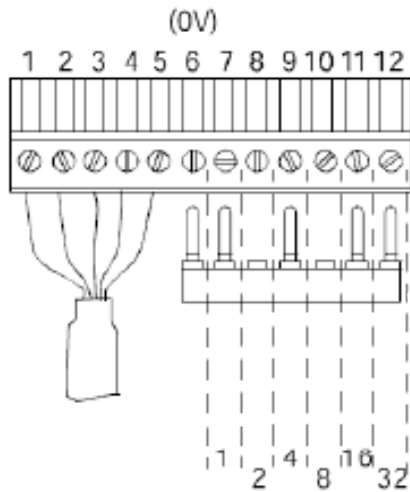
- IO模块介绍

ABB标准IO模块:

型号	说明
DSQC651	DI8/DO8 AO2
DSQC652	DI16/DO16
DSQC653	DI8/DO8带继电器
DSQC355A	AI4/AO4
DSQC377A	输送带跟踪单元

- IO模块介绍

将此插头插在DeviceNet总线接口上



此短片用于确定IO板在总线上的地址,地址可用范围为10~63。此例中,第8脚和第10脚的跳线被剪掉,获得的地址为 $2+8=10$

DO指示灯

DeviceNet
总线接口

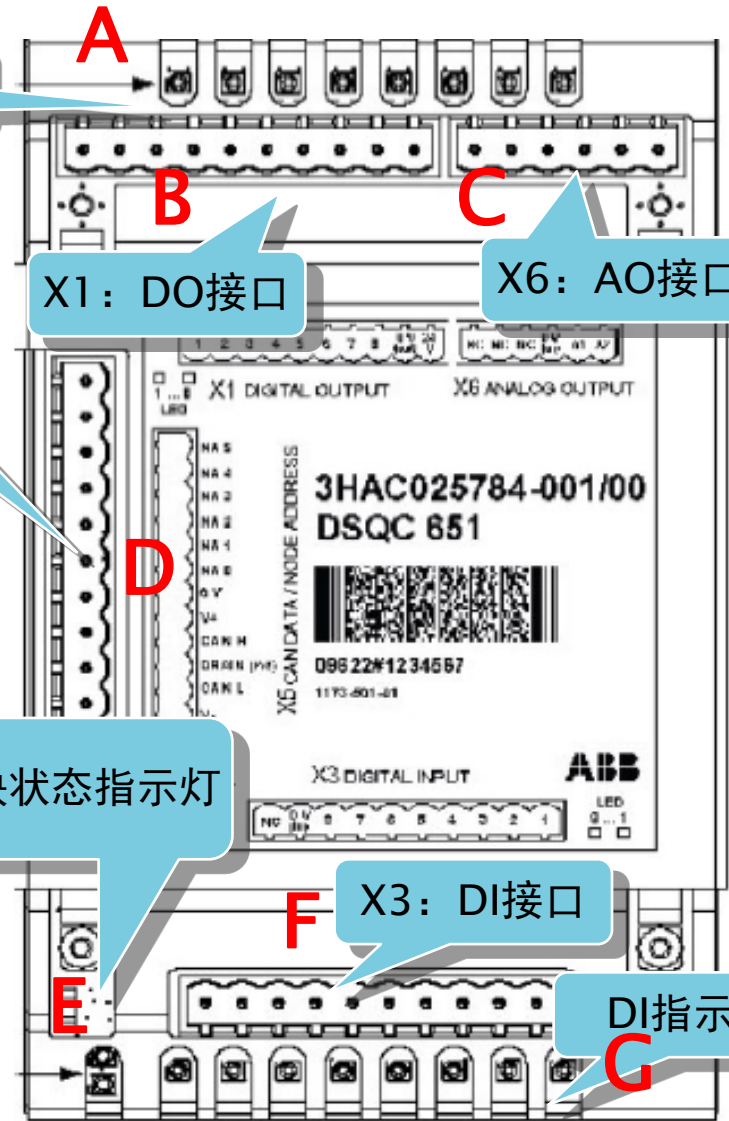
模块状态指示灯

X1: DO接口

X6: AO接口

X3: DI接口

DI指示灯



- IO模块介绍

IO模块接口说明:

X1端子编号	使用定义	地址分配	
		DSQC651	DSQC652
1	OUTPUT CH1	32	0
2	OUTPUT CH2	33	1
3	OUTPUT CH3	34	2
4	OUTPUT CH4	35	3
5	OUTPUT CH5	36	4
6	OUTPUT CH6	37	5
7	OUTPUT CH7	38	6
8	OUTPUT CH8	39	7
9	0V	-	-
10	24V	-	-

X2端子编号	使用定义	地址分配	
		DSQC651	DSQC652
1	OUTPUT CH9	无	8
2	OUTPUT CH10		9
3	OUTPUT CH11		10
4	OUTPUT CH12		11
5	OUTPUT CH13		12
6	OUTPUT CH14		13
7	OUTPUT CH15		14
8	OUTPUT CH16		15
9	0V		-
10	24V		-

X3端子编号	使用定义	地址分配	
		DSQC651	DSQC652
1	INPUT CH1	0	0
2	INPUT CH2	1	1
3	INPUT CH3	2	2
4	INPUT CH4	3	3
5	INPUT CH5	4	4
6	INPUT CH6	5	5
7	INPUT CH7	6	6
8	INPUT CH8	7	7
9	0V	-	-
10	未使用	-	-

X4端子编号	使用定义	地址分配	
		DSQC651	DSQC652
1	INPUT CH9	无	8
2	INPUT CH10		9
3	INPUT CH11		10
4	INPUT CH12		11
5	INPUT CH13		12
6	INPUT CH14		13
7	INPUT CH15		14
8	INPUT CH16		15
9	0V		-
10	未使用		-

- IO模块介绍

IO模块接口说明:

X5端子编号	使用定义
1	0V BLACK
2	CAN信号线low BLUE
3	屏蔽线
4	CAN信号线high WHITE
5	24V RED
6	GND地址选择公共端
7	模块ID bit0(LSB)
8	模块ID bit1(LSB)
9	模块ID bit2(LSB)
10	模块ID bit3(LSB)
11	模块ID bit4(LSB)
12	模块ID bit5(LSB)

X6端子编号	使用定义	地址分配
1	未使用	-
2	未使用	-
3	未使用	-
4	0V	-
5	A01	0~15
6	A02	16~31

- I/O信号设定

设定I/O模块单元

参数名称	设定值	说明
Name	Board10	设定I/O板在系统中的名称
Type of Unit	D651	设定I/O板类型
Connected to Bus	DeviceNet1	设定I/O板连接的总线
DeviceNet Address	10	设定I/O板在总线中的地址

设定数字I/O信号

参数名称	设定值				说明
	DI	DO	GI	GO	
Name	di1	do1	gi1	go1	设定信号名称
Type of Signal	Digital Input	Digital Output	Group Input	Group Output	设定信号类型
Assigned to Unit	board10	board10			
Unit Mapping	0	32	1-4	33-36	设定信号所占的地址

- IO信号设定

设定AO信号

参数名称	设定值	说明
Name	aol	设定信号名称
Type of Signal	Analog Output	设定信号类型
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
Maximum Bit Value	65535	设定最大...

GO

- IO信号监控

The image shows the ABB robot control interface. At the top left is the ABB logo. The top status bar shows "手动" (Manual) and "防护装置停止" (Safety device stop), with "chensi (A-A01-6021)" and "已停止 (速度 100%)" (Stopped (speed 100%)). The main menu is displayed in a grid:

- HotEdit
- 输入输出** (I/O) - highlighted in blue, with a green callout bubble containing the text "1、选择“输入输出”" (1. Select "I/O").
- 手动操纵 (Manual operation)
- 自动生产窗口 (Automatic production window)
- 程序编辑器 (Program editor)
- 程序数据 (Program data)
- 备份与恢复 (Backup and recovery)
- 校准 (Calibration)
- 控制面板 (Control panel)
- 事件日志 (Event log)
- FlexPendant 资源管理器 (FlexPendant resource manager)
- 系统信息 (System information)

At the bottom of the menu are "注销" (Log out) and "重新启动" (Restart). Below the main menu are two buttons: "自动生..." (Automatic production) and "I/O". In the bottom right corner, there is a "ROB_1" status indicator with a red stop sign.

- I/O信号监控

ABB 手动 chensi (A-A01-6021) 防护装置停止 已停止 (速度 100%)

输入输出

常用 选择布局 默认

从列表选择一个 I/O 信号。

名称	值	类型
----	---	----

虚拟

自动生... I/O

ROB_1

2、选择“视图”

- 化名 I/O
- I/O 总线
- I/O 单元
- 全部信号
- 数字输入
- 数字输出
- 模拟输入
- 模拟输出
- 组输入
- 组输出
- ✓ 常用
- 安全信号
- 仿真信号
- 视图

- IO信号监控

ABB 手动 防护装置停止
chensi (A-A01-6021) 已停止 (速度 100%)

I/O 单元

I/O 单元上的信号: board10 选择布局
从列表选择一个 I/O 信号。 默认

名称 ▲	值	类型	仿真
aol	0.00	AO: 0-15	否
di1	0	DI: 0	是
dol	0	DO: 32	否
gil	0	GI: 1-4	否
gol	0	GO: 33-36	否

3、选择“di1”

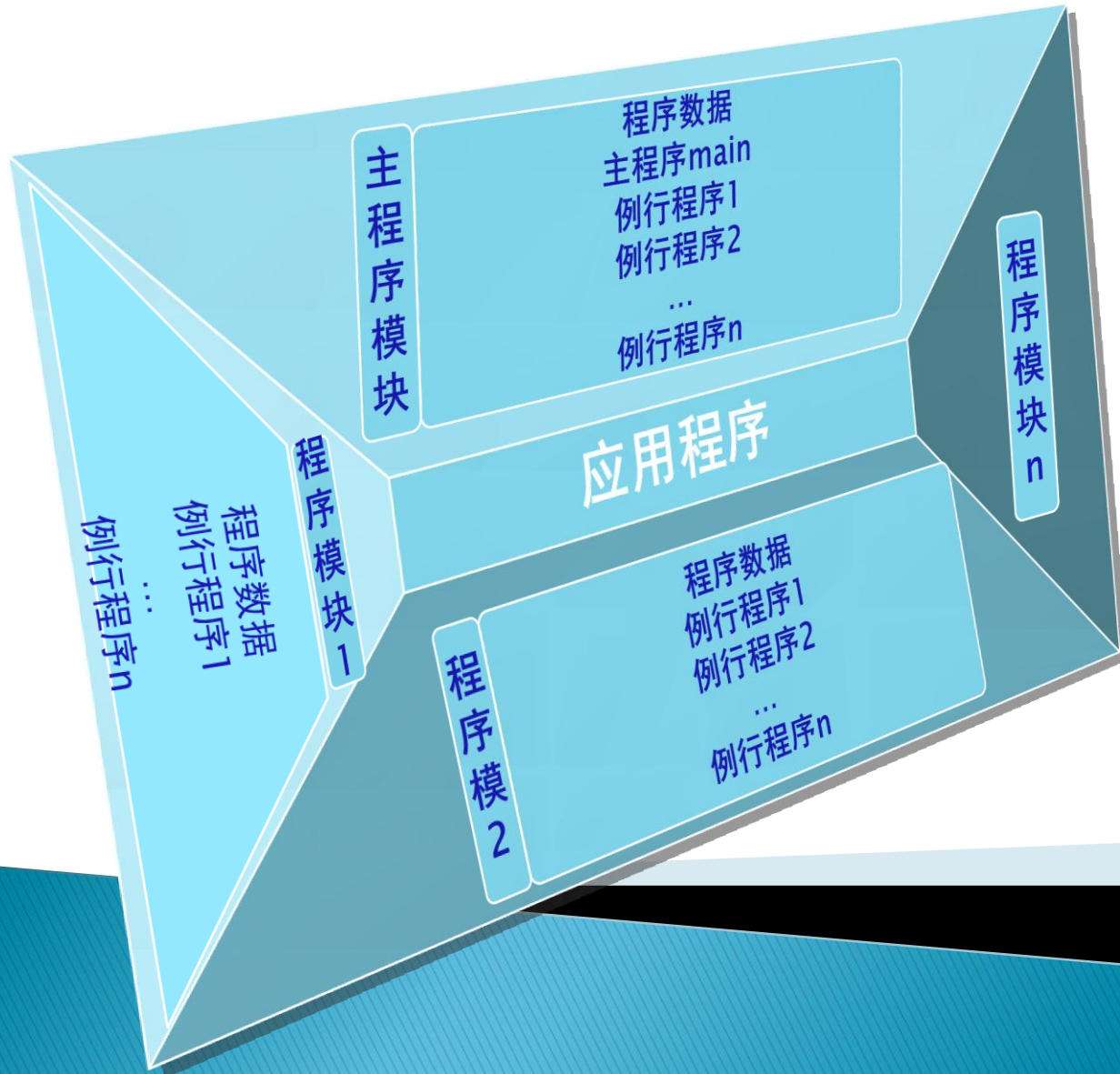
4、单击“仿真”，单击“1”，将di1的状态仿真为“1”。仿真结束后，单击“消除仿真”，di1恢复“0”值

0 1 消除仿真 虚拟 关闭

自动生... I/O ROB_1

Rapid程序指令介绍

- Rapid程序架构



- Rapid程序数据

➤程序数据：

程序数据是在程序模块或系统模块中设定的值和定义的一些环境数据。创建的程序数据由同一个模块或其他模块中的指令进行应用。

➤程序数据的类型分类：

ABB机器人常用的程序类型有76个，并可以根据实际需要进行程序数据的创建。在示教器的“程序数据”窗口，可以查看和创建所需要的程序数据。



- Rapid程序数据

程序数据的存储类型:

变量
VAR

在程序执行的过程中和停止时，会保持当前的值。但如果程序指针被移到主程序后，数值丢失。

eg: VAR num length:=0;

可变量
PERS

特点：无论程序的指针如何，都会保持最后赋予的值。

eg: PERS num br: =1;

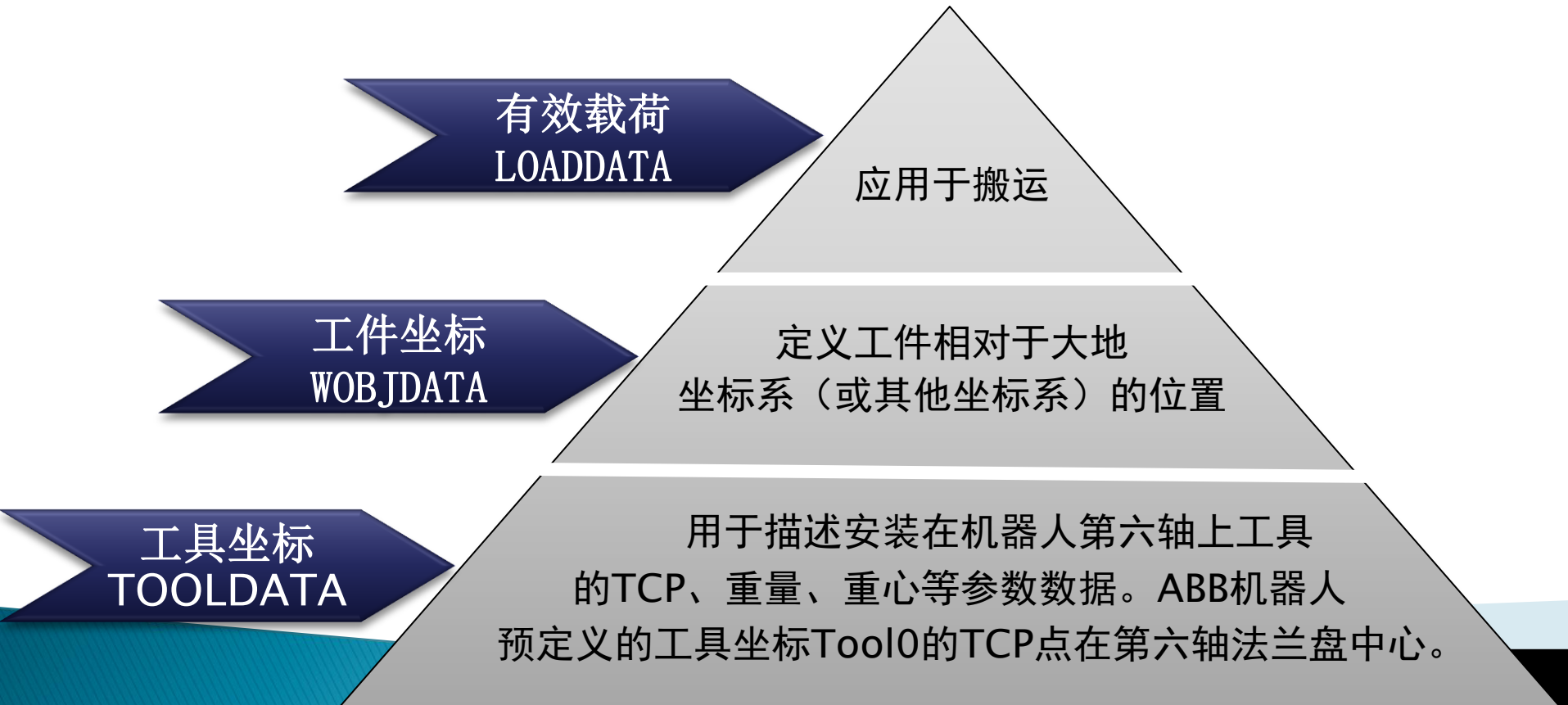
常量
CONST

特点：在定义时已赋予了数值，不能在程序中进行修改。

eg: CONST num
high:=9.81 ;

- Rapid程序数据

三个重要的Rapid程序数据:



- Rapid程序数据

工具坐标的设定:

➤工具坐标设定必备条件:

- 1) **固定参考点:** 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
- 2) **工具上的参考点:** 在工具上确定一个参考点, 推荐使用工具的中心点

➤坐标设定方法:

通过手动操作机器人的方法, 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, 将使用六点法进行操作, 第四点时用工具的参考点垂直于固定点, 第五点时工具参考点从固定点向将要设定为TCP的X方向移动, 第六点时工具参考点从固定点向将要设定为TCP的Z方向移动)

- 1) 4点法: 不改变tool0的坐标方向
- 2) 5点法: 改变tool0的Z方向
- 3) 6点法: 改变tool0的X和Z方向 (在焊接应用最为常用)

GO

- Rapid程序数据

工件坐标的设定:

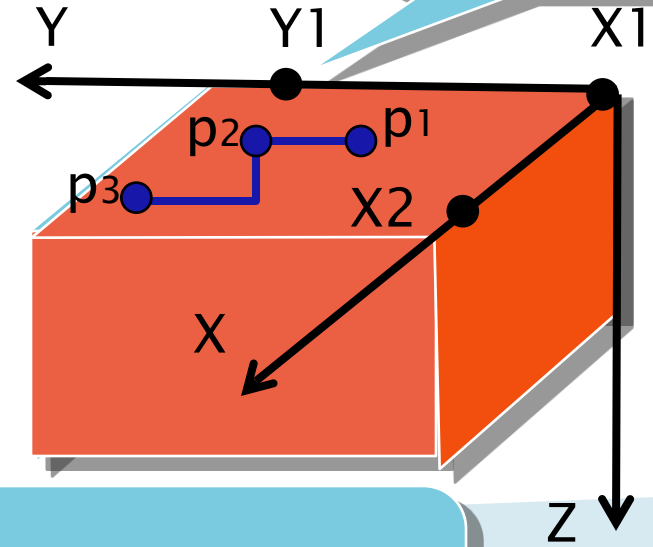
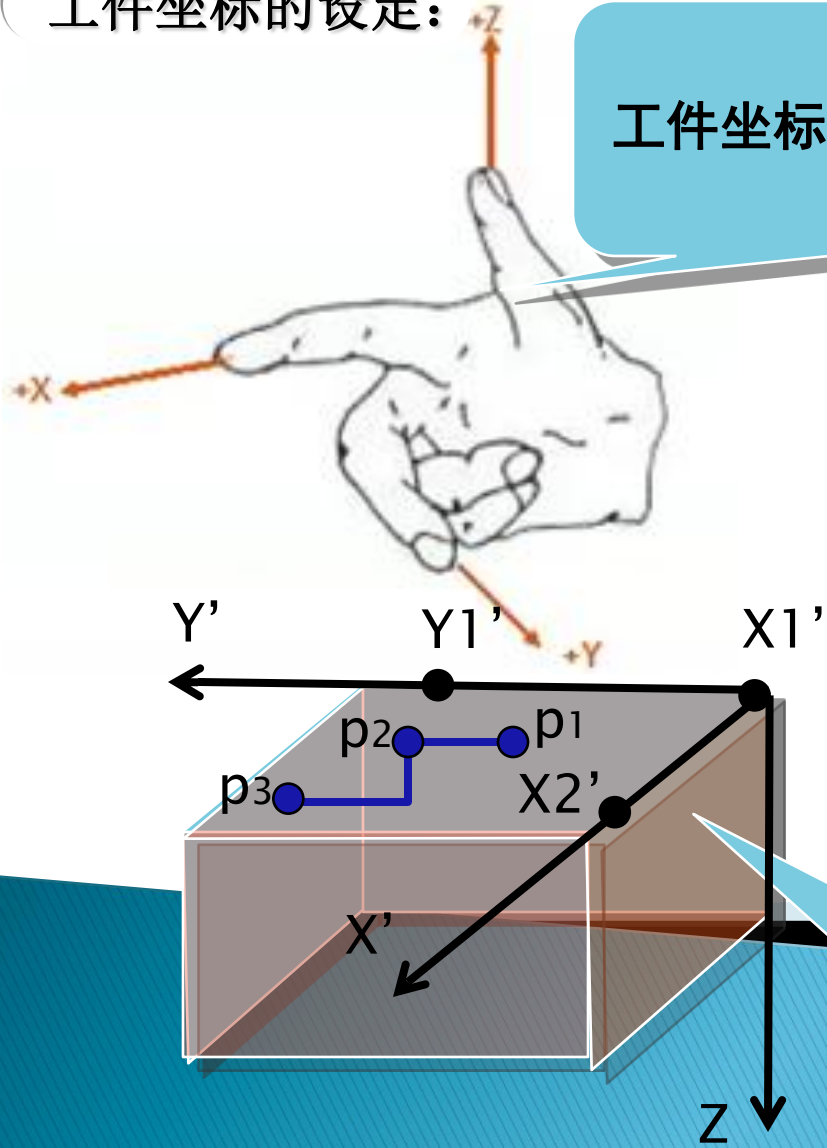
工件坐标符合右手定则

三点法:

X1: 原点

X2: X正方向

Y1: Y正方向



优点:

工件挪动后只需重新设定
工件坐标即可找回原先示
教的轨迹点

- 常用Rapid指令

➤ 赋值指令：

“:=” 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。

eg1: reg1:=5;

➤ 机器人运动指令：

eg2: reg2:=reg1+4;

机器人运动指令有：关节运动MOVEJ、线性运动MOVEL、圆弧运动MOVEC

eg: MoveL P10, v20, z5, tool0\WObj:=wobj0;

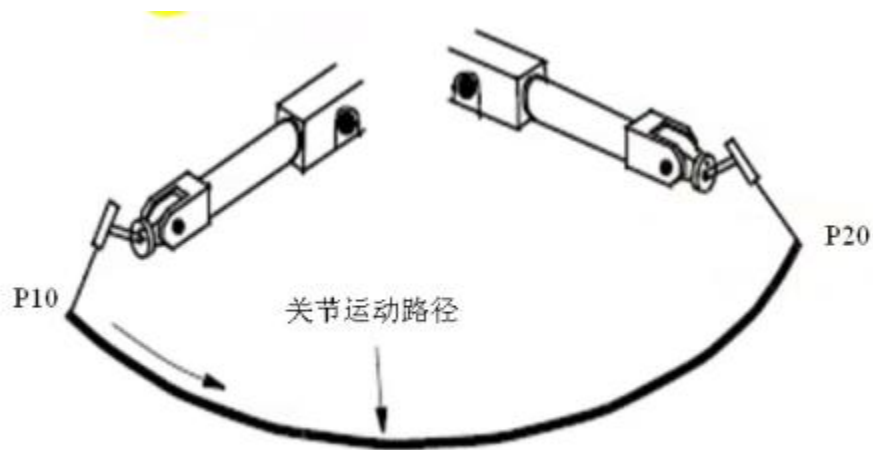
程序数据	数据类型	说明
P10	robotarget	机器人运动目标位置
v20	speeddata	机器人运动速度
z5	zonedata	机器人运动转弯数据
tool0	tooldata	机器人工具数据TCP
wobj0	wobjdata	机器人工件坐标数据

- 常用Rapid指令

关节运动MOVEJ:

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

```
MoveJ P20, v20, fine, tool0\WObj:=wobj0;
```

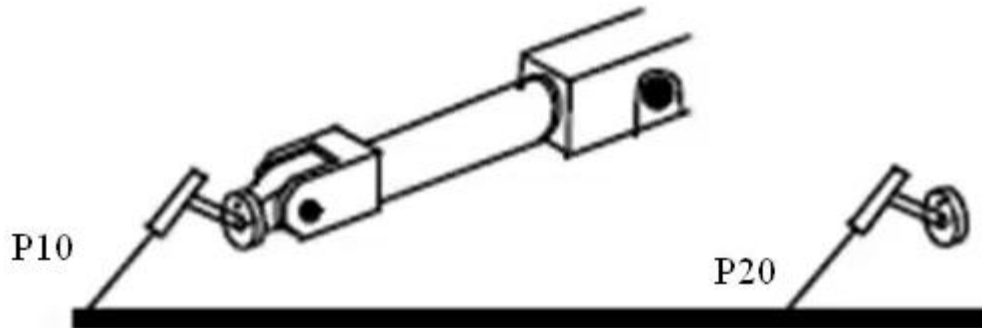


- 常用Rapid指令

线性运动MOVEL:

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，两个点之间不易太远，否则机器人容易走到死点，为保证精度，应使两点之间的距离小一点。

```
MoveL P20, v20, fine, tool0\WObj:=wobj0;
```



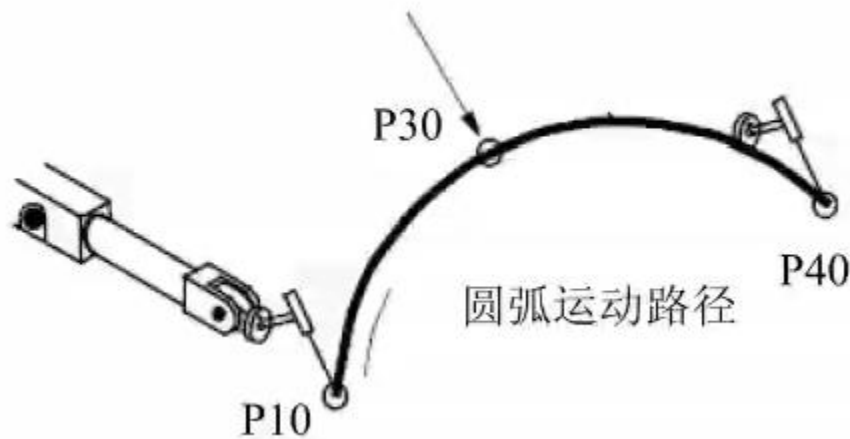
▪ 线性运动路径

- 常用Rapid指令

圆弧运动MOVEC:

圆弧运动是在机器人可到达的空间范围内定义三个位置点，第一个点是圆弧的起点，第二点是用于圆弧的曲率，第三个点是圆弧的终点。

```
MoveL p10, v1000, fine, tool1 \wobj:=wobj1;  
MoveC p30, p40, v1000, z1, tool1 \wobj:=wobj1;
```



- 常用Rapid指令

IO控制指令:

IO控制指令用于控制IO信号，以达到与机器人外围设备进行通讯的目的。

序号	程序指令	含义
1	Set do1;	将do1置位为1
2	Reset do1;	将do1置位为0
3	WaitDI di1, 1;	用于判断数字输入信号“di1”的值是否与目标值“1”的一致
4	WaitDO do1, 1;	用于判断数字输出信号“do1”的值是否与目标值“1”的一致
5	WaitUntil di1=1; WaitUntil do1=0; WaitUntil num1=4;	用于布尔量，数字量和IO信号值的判断，若条件到达指令中的设定值，程序继续往下执行，否则就一直等待，除非设定最大等待时间

- 常用Rapid指令

条件逻辑指令：

➤ Compact IF紧凑型条件判断指令：

CompactIF紧凑型条件判断指令，用于当一个条件满足时，就执行一句指令。

eg: IF flag1=TRUE Set do1;

➤ IF条件判断指令

IF条件判断指令是根据不同的条件去执行不同的指令。其条件判断的条件可以根据实际情况进行增加与减少。

```
eg: IF num1=1 THEN
      flag1:=TRUE;
      ELSEIF num1=2 THEN
      Flag1:= FALSE;
      ELSE
      Set do1;
      ENDIF
```

- 常用Rapid指令

循环指令：

➤FOR循环指令

FOR循环指令，是用于一个或多个指令需要重复执行数次的情况。

```
eg: FOR FROM 1 TO 10 DO  
      num: =num+1;  
      ENDFOR
```

➤WHILE循环指令

WHILE循环指令，是用于在给定的条件满足的情况下，一直重复执行对应的指令。

```
eg: WHILE num1 > num2 DO  
      num1:=num1-1;  
      ENDWHILE
```

- 常用Rapid指令

其他常用指令:

➤ WaitTime时间等待指令

WaitTime时间等待指令，用于程序在等待一个指定的时间以后，再继续往下执行。

eg: WaitTime 4
Reset do1;

➤ ProcCall调用例行程序指令

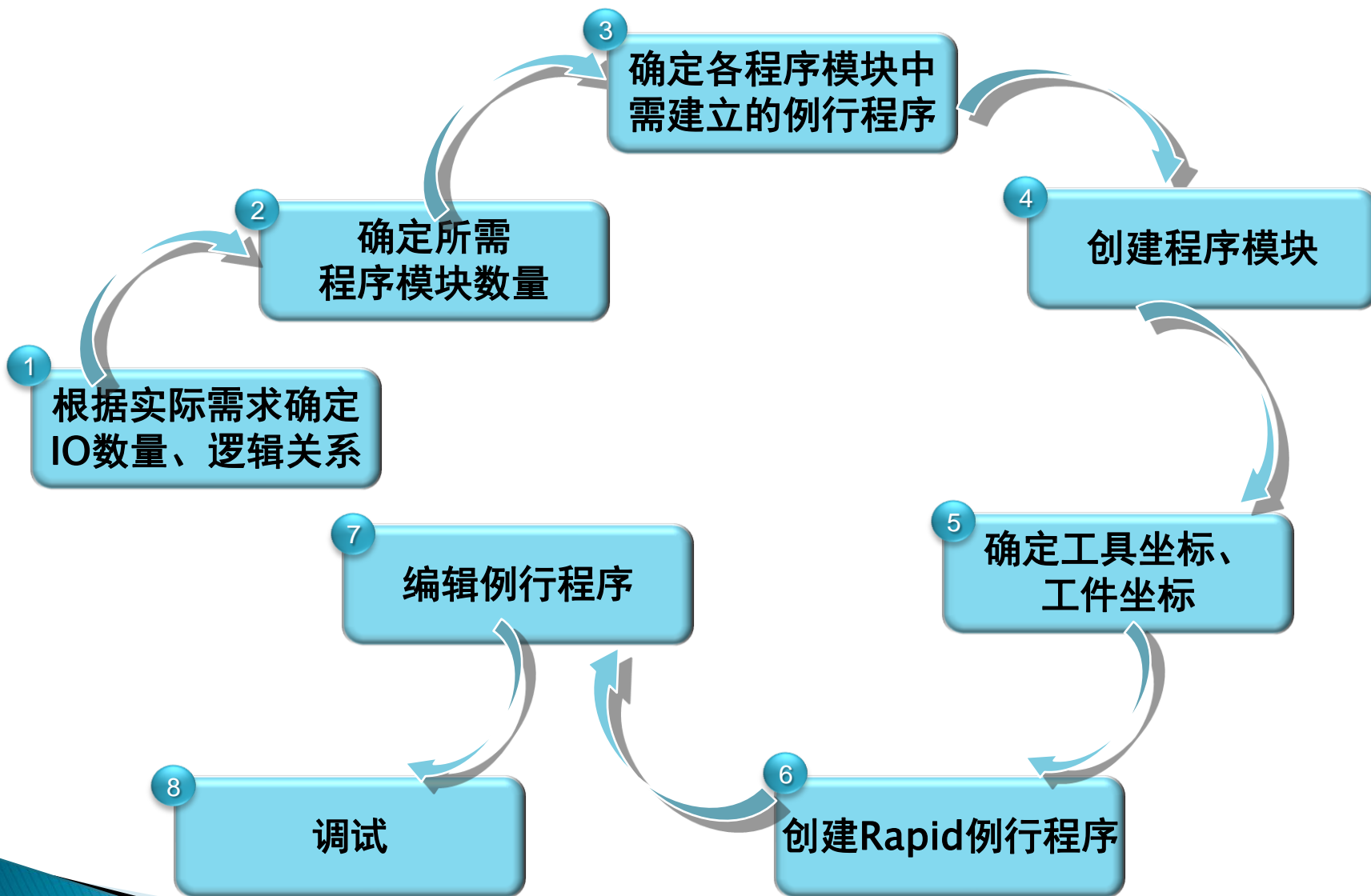
➤ RETURN返回例行程序指令

```
PROC Routine1()  
    MoveL p10, v1000, fine,  
    tool1 \wobj:=wobj1;  
    Routine2;  
    Set do1;  
ENDPROC  
PROC Routine2()  
    IF di1=1 THEN  
        RETURN;  
    ELSE  
        Stop;  
    ENDIF  
ENDPROC
```



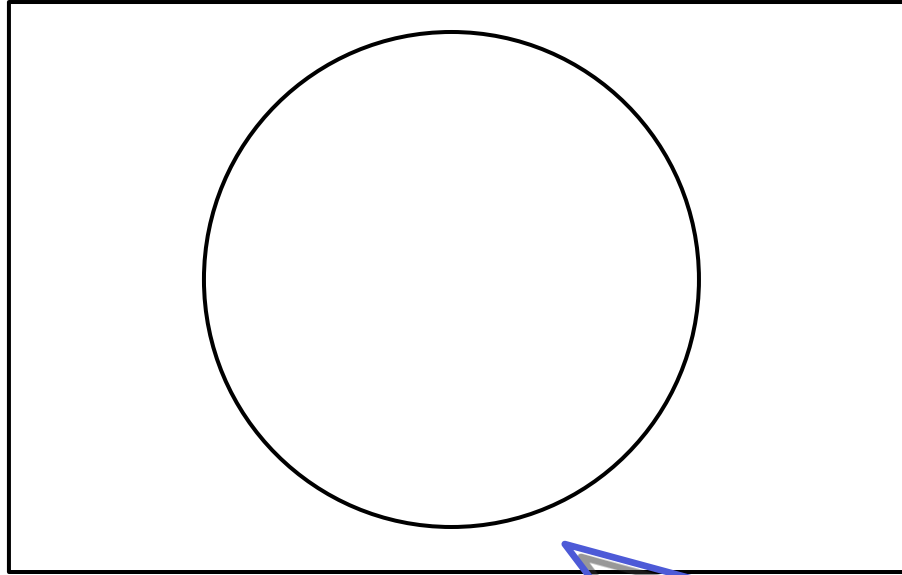
应用案例

- 程序编写流程



- 应用案例

要求：



- 1、要求机器人对上图所示的两个焊缝轨迹进行焊接
- 2、焊接轨迹1：长×宽为580mm × 400mm的矩形轨迹
焊接轨迹2：直径为350mm的圆形轨迹
- 3、当外部要求焊接圆形轨迹时，机器人进行圆形轨迹焊接
当外部要求焊接矩形轨迹时，机器人进行矩形轨迹焊接

- 应用案例

确定IO信号:

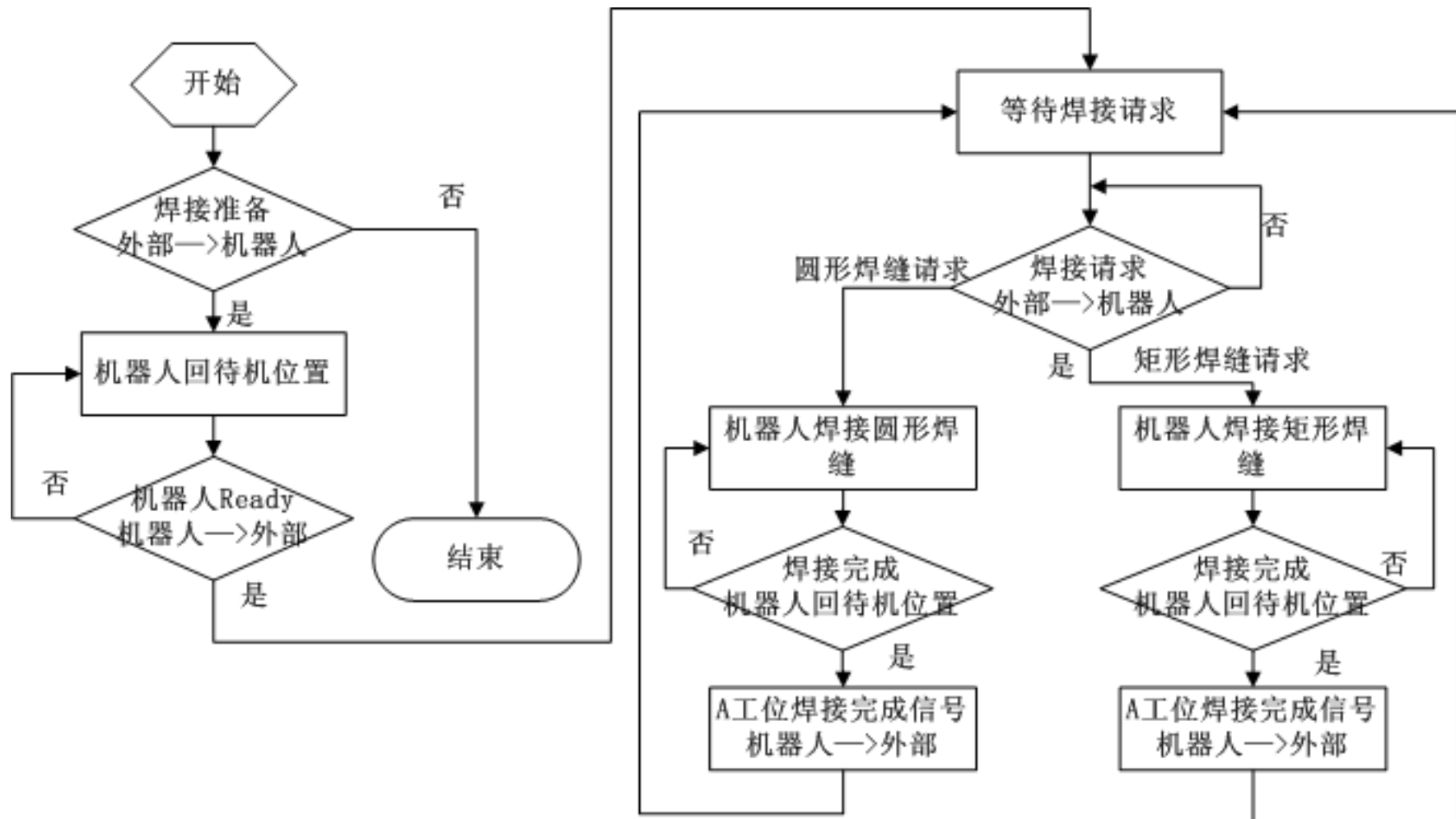
序号	类型	名称	IO地址	说明
1	DI	DI_PREPARE	1	焊接准备信号
2	DI	DI_REQUEST_A	2	焊接请求A: 圆形焊缝轨迹
3	DI	DI_REQUEST_B	3	焊接请求B: 矩形焊缝轨迹
4	DO	DO_RUNENABLE	1	机器人Ready
5	DO	DO_A_OVER	1	圆形轨迹焊接完毕 注意: 此时机器人必须在安全位置
6	DO	DO_B_OVER	2	矩形轨迹焊接完毕 注意: 此时机器人必须在安全位置
7	DO	DO_LASER_TRIG	3	触发出光



注意 本案例中机器人IO模块采用DSQC652

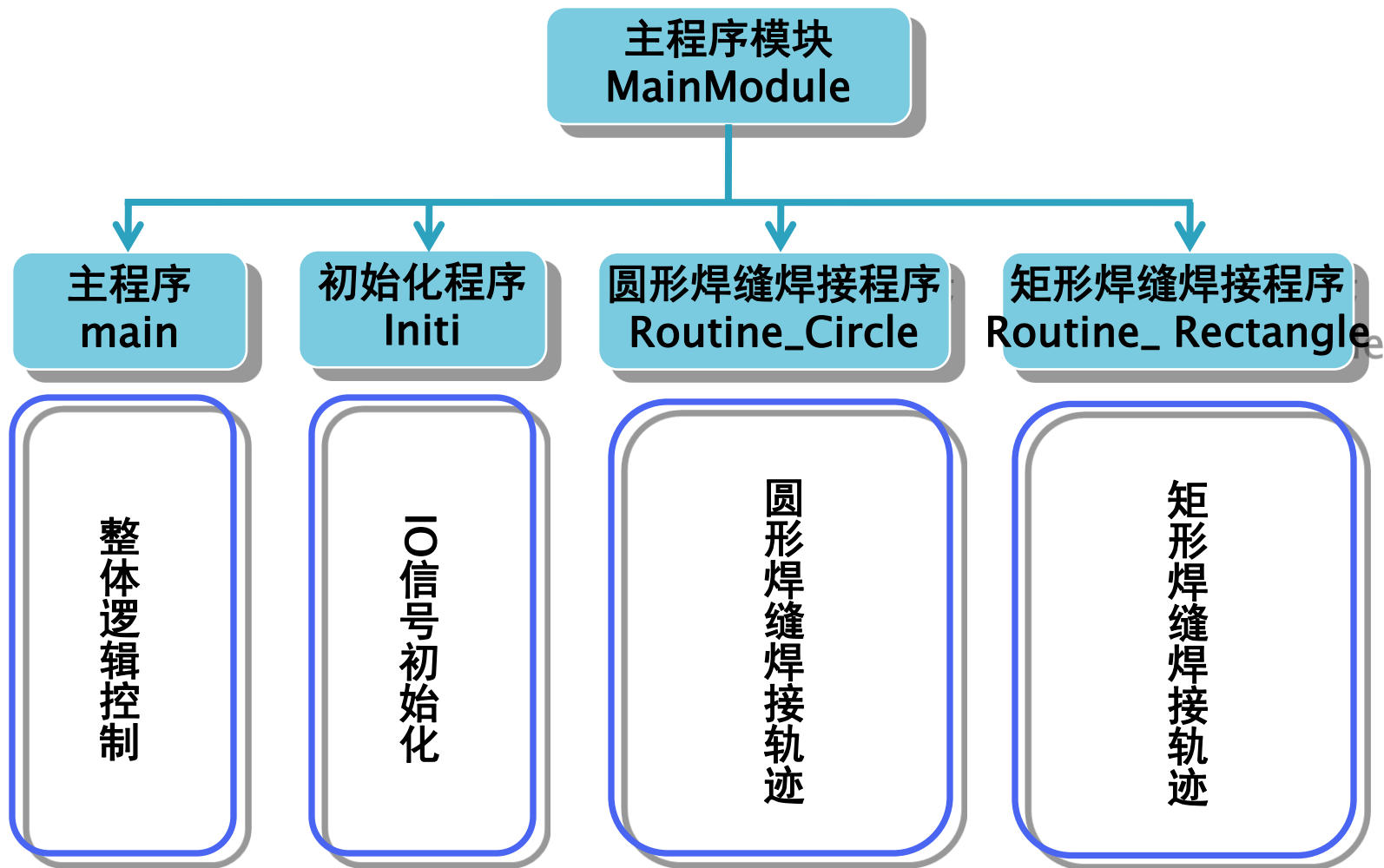
- 应用案例

确定逻辑关系:



- 应用案例

确定程序结构:



- 应用案例

创建程序:

666 (A-A01-6021) 防护装置停止 已停止 (速度 100%)

I_ROB1/MainModule

例行程序 活动过滤器:

名称 ▲	模块	类型	1 到 4 共 4
Initi()	MainModule	Procedure	
main()	MainModule	Procedure	
Routine_Circle()	MainModule	Procedure	
Routine_rectangle()	MainModule	Procedure	

注意
编辑Rapid程序前须先确定要使用的工具坐标和工件坐标。若需使用新的坐标,则须提前创建新的工具和工件坐标。

Rapid程序

文件 ▲ 显示例行程序 后退

I_ROB1 MainMo... ROB_1

- 应用案例

调试例行程序：

- 应用案例

确定程序结构：

- 应用案例

确定程序结构：

- 应用案例

确定程序结构：

谢谢！