

EDUBOT

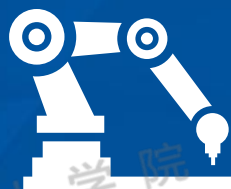
哈工海渡机器人学院

工业机器人入门实用教程 (ABB机器人)

主讲人：滕老师

www.irobot-edu.com

edubot_zhang@126.com



- 第01章 ABB机器人认知
- 第02章 机器人操作基础
- 第03章 机器人零点校准
- 第04章 机器人坐标系建立
- 第05章 机器人编程基础
- 第06章 机器人I/O通信
- 第07章 机器人基本指令
- 第08章 异常事件
- 第09章 编程实例
- 第10章 离线仿真

第1章 ABB机器人认知

教学内容：

- 1.1 ABB机器人应用领域
- 1.2 ABB机器人简介
- 1.3 ABB机器人系统组成
- 1.4 IRB120本体介绍
- 1.5 IRB120控制器介绍

学习目标：

- 1. 掌握ABB机器人不同领域的应用
- 2. 了解ABB机器人基本组成



1.1 ABB机器人应用领域

恶劣工作环境及危险工作



热锻

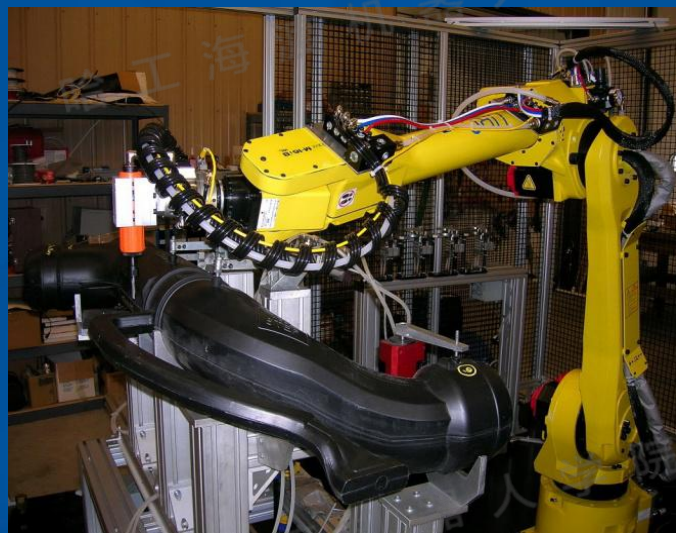


压铸



1.1 ABB机器人应用领域

自动化生产领域



装配



激光雕刻



1.2 ABB机器人简介

不同类型机器人



六轴机器人



码垛机器人



并联机器人



双臂机器人



1.3 ABB机器人系统组成

EDUBOT

哈工海渡机器人学院

工业机器人的人机交互接口，针对机器人的所有操作基本上都是通过示教器来完成的。

示教器

用来控制工业机器人按规定要求动作，是机器人的关键和核心部分。

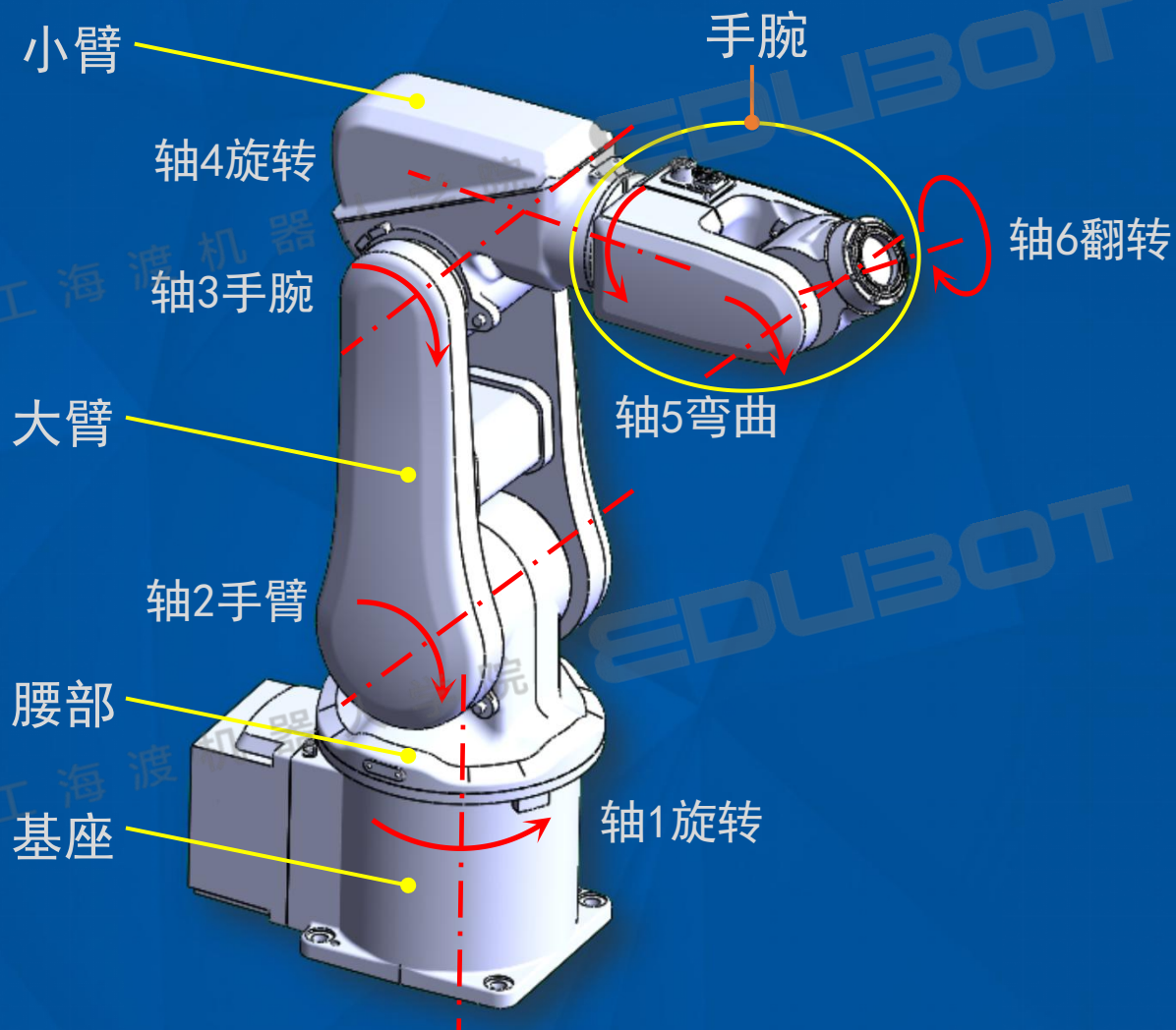
控制器

又称机器人本体，是工业机器人的机械主体，是用来完成规定任务的执行机构。

操作机



1.4 IRB120本体介绍



1.4 IRB120控制器介绍

IRC5紧凑型控制器



第2章 机器人操作基础

教学内容：

- 2.1 工作模式
- 2.2 动作模式
- 2.3 奇异点

学习目标：

- 1. 掌握机器人工作模式
- 2. 了解奇异点意义



2.1 IRB120基本概念

工作模式

手动模式



用于程序调试，运动速度限制在 250 mm/s 下。要激活电机上电，必须按下使动装置。

自动模式



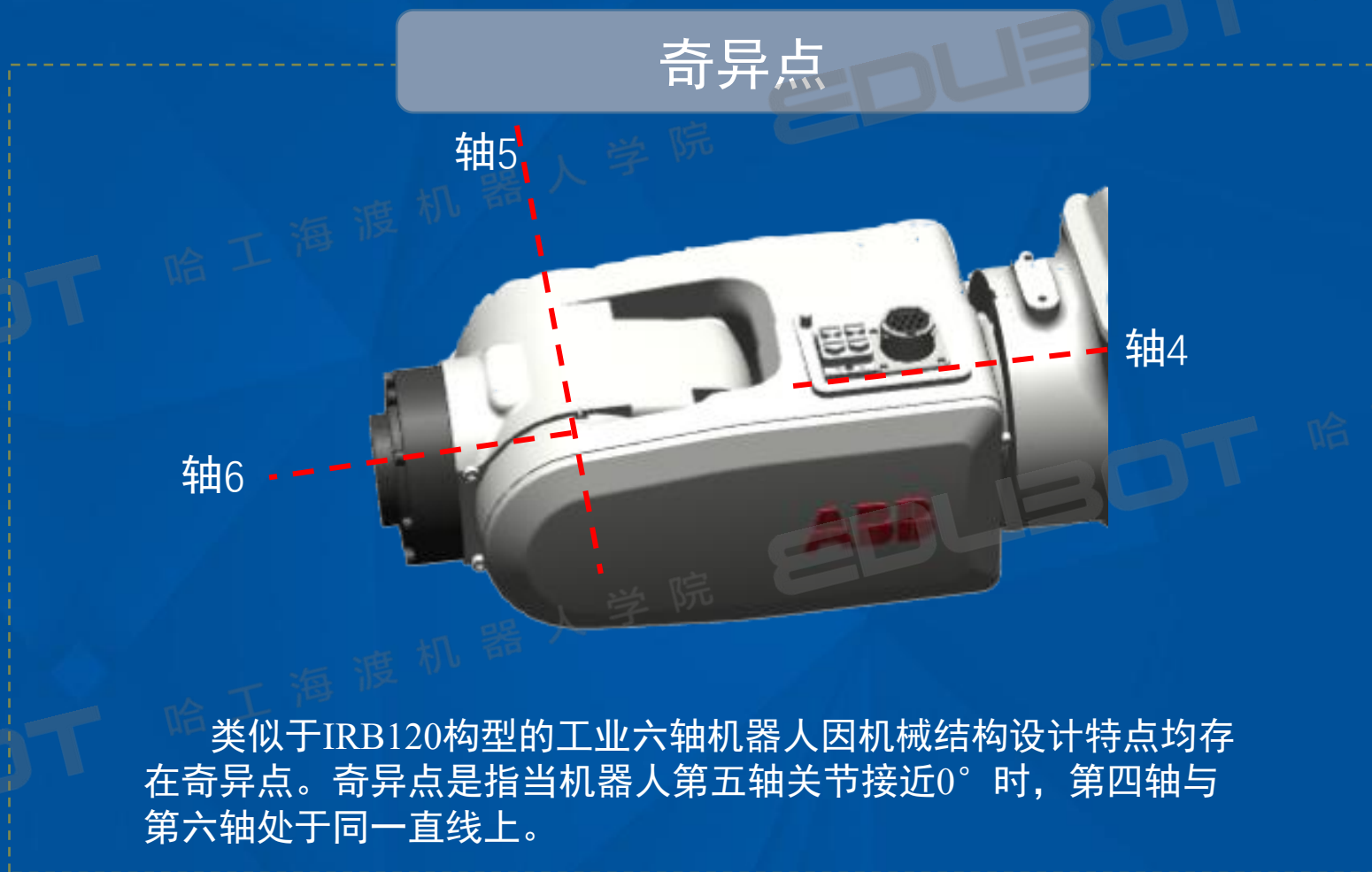
用于工业生产，机器人自动连续运行生产作业。



2.1 IRB120基本概念



2.1 IRB120基本概念



第3章 零点校准

教学内容：

- 3.1 零点校准情况
- 3.2 零点校准方法

学习目标：

- 1. 掌握机器人零点校准
- 2. 了解零点校准的情况



3.1 零点校准情况

零点校准情况

- 新购买机器人时，厂家未进行机器人零点校准。
- 电池电量不足，更换电池。
- 更换机器人本体或控制器。
- 转数计数器数据丢失。



3.2 零点校准方法

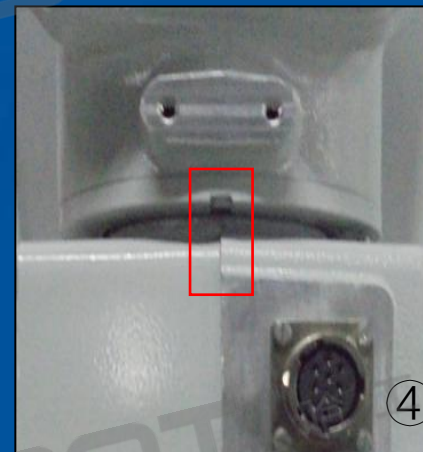
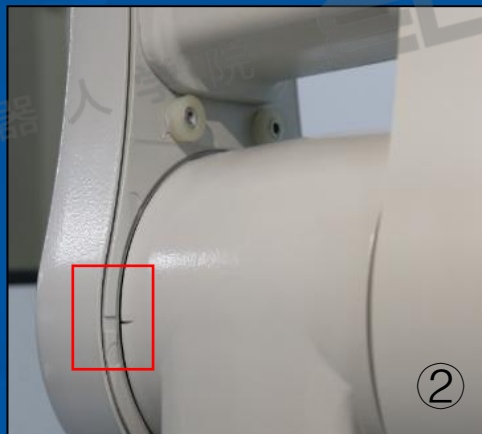
零点校准方法

IRB120机器人本体的六个轴均有零点标记，手动将机器人各轴零点标记对准，记录当前转数计数器数据，控制器内部将自动计算出该轴的零点位置，并以此作为各轴的基准进行控制。



3.2 零点校准方法

零点校准的方法



第4章 机器人坐标系的建立

教学内容：

- 4.1 坐标系统的概念
- 4.2 工具坐标系的建立
- 4.3 工件坐标系的建立

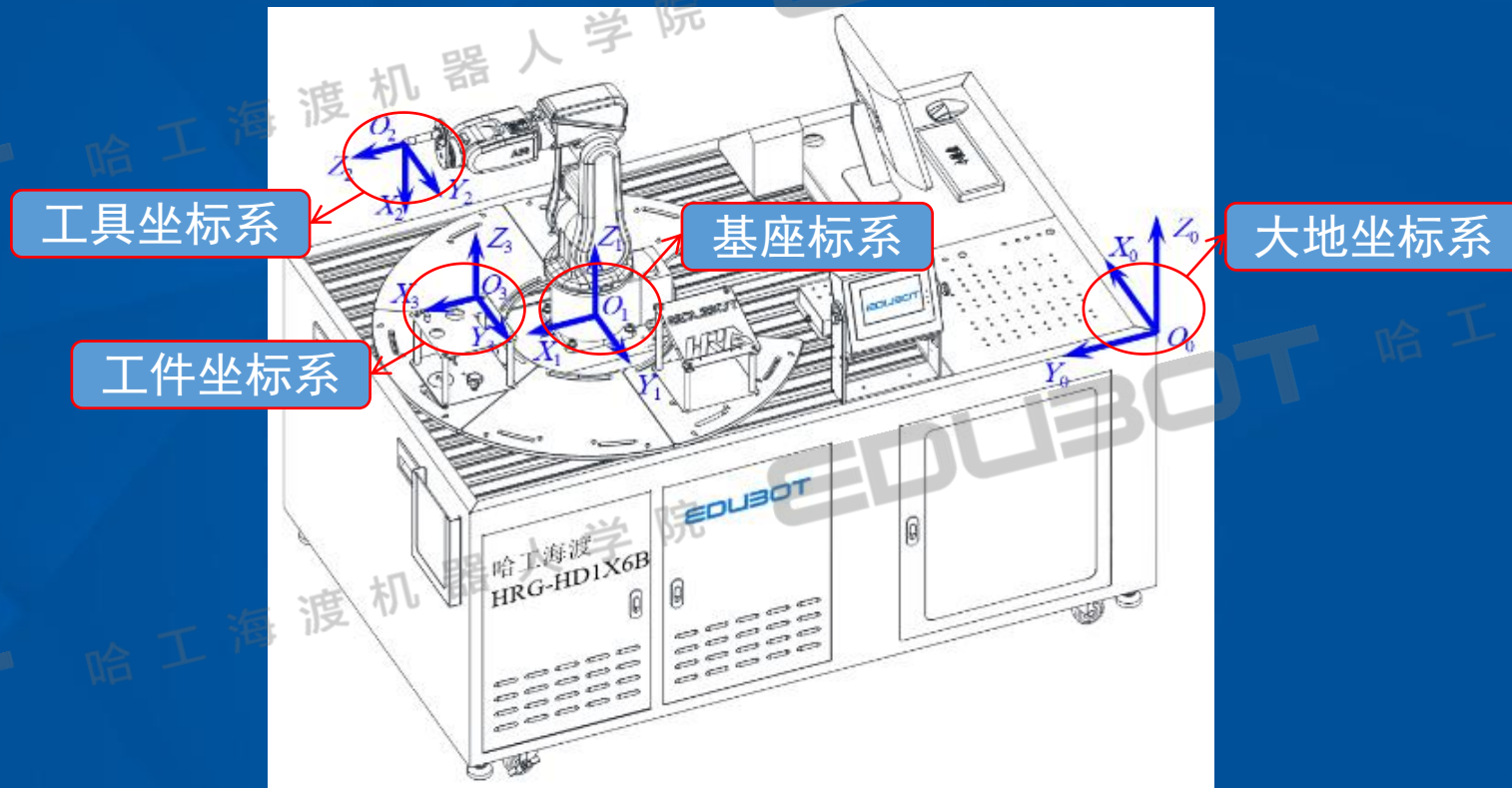
学习目标：

- 1. 掌握工具坐标系的建立
- 2. 掌握工件坐标系的建立



4.1 IRB120基本概念

参考坐标系



4.2 工具坐标系标定

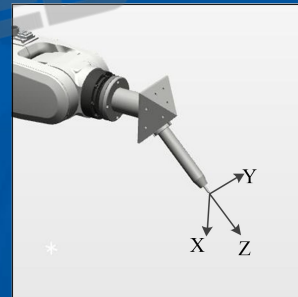
工具坐标系概念

机器人系统对其位置的描述和控制是以机器人的工具TCP（tool center point）为基准的，为机器人所装工具建立工具坐标系，可以将机器人的控制点转移到工具末端，方便手动操纵和编程调试。

默认工具坐标系

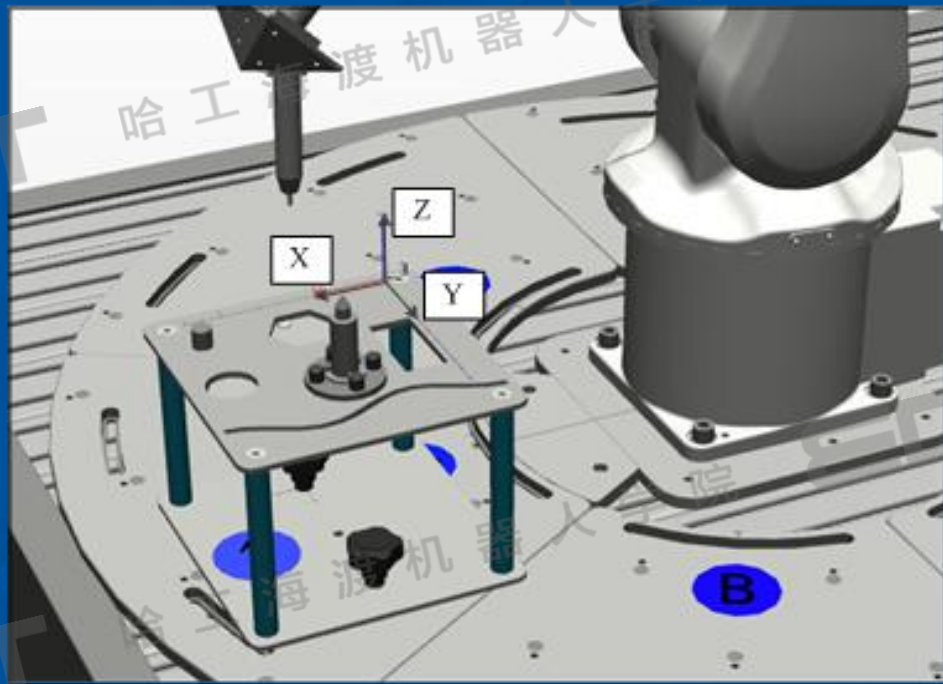


设定工具坐标系



4.3 工件坐标系标定

工件坐标系概念



工件坐标系用于定义工件相对于大地坐标系或者其他坐标系的位置，具有两个作用：一是方便用户以工件平面方向为参考手动操纵调试；

二是当工件位置更改后，通过重新定义该坐标系，机器人即可正常作业，不需要对机器人程序进行修改。



第5章 机器人通信

教学内容：

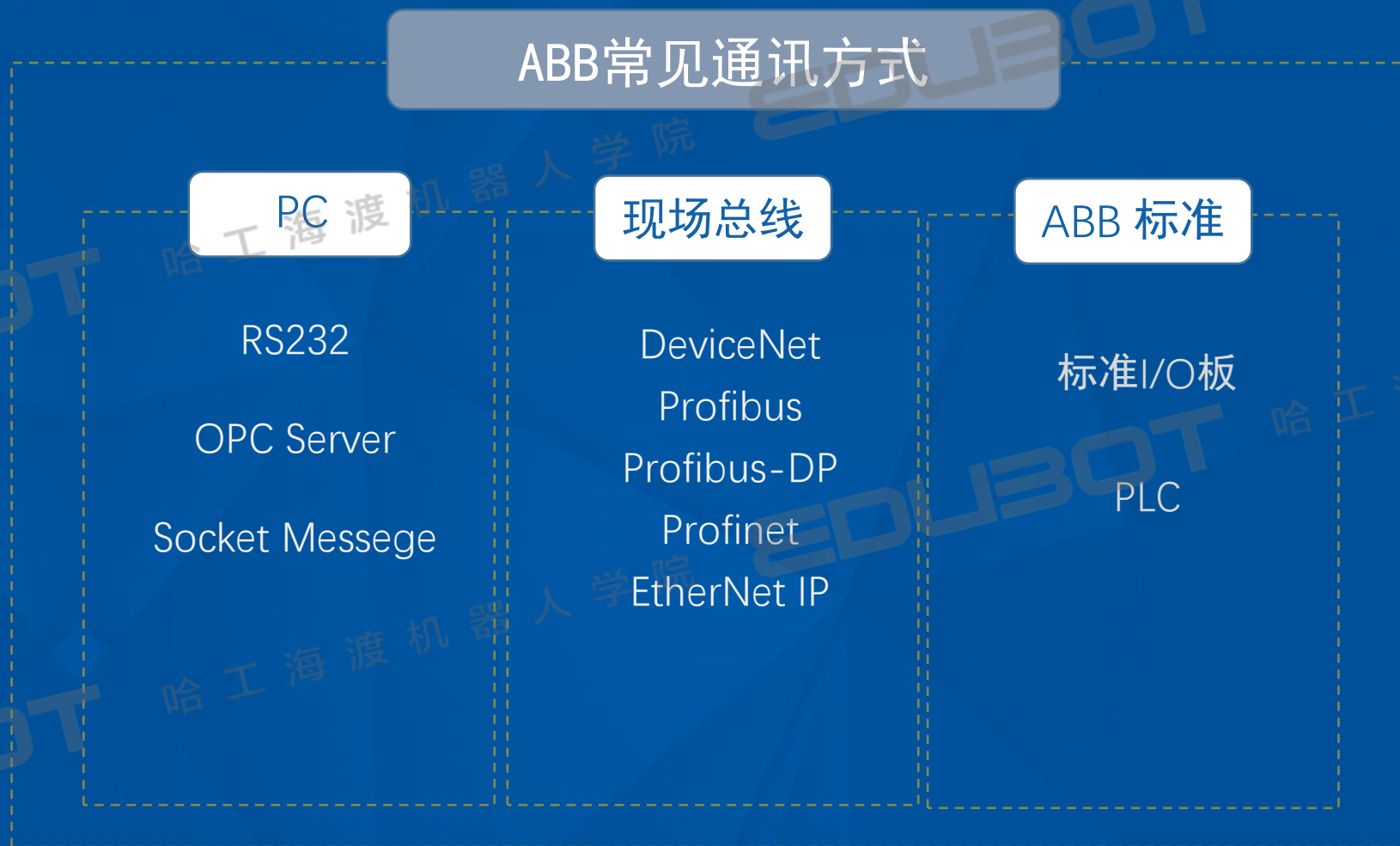
- 5.1 常见通信方式
- 5.2 ABB标准通信

学习目标：

- 1. 掌握机器人现场总线的应用
- 2. 掌握ABB标准IO通信



5.1 通信方式



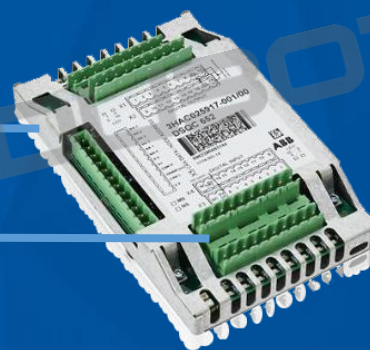
I/O信号标准配置

ABB标准I/O板安装完成后，需要对各信号进行一系列设置后才能在软件中使用，设置的过程称为I/O配置。



挂载I/O板

映射I/O信号



第6章 机器人编程基础

教学内容：

- 6.1 模块及例行程序的创建
- 6.2 数据类型
- 6.3 存储类型
- 6.4 动作指令解析

学习目标：

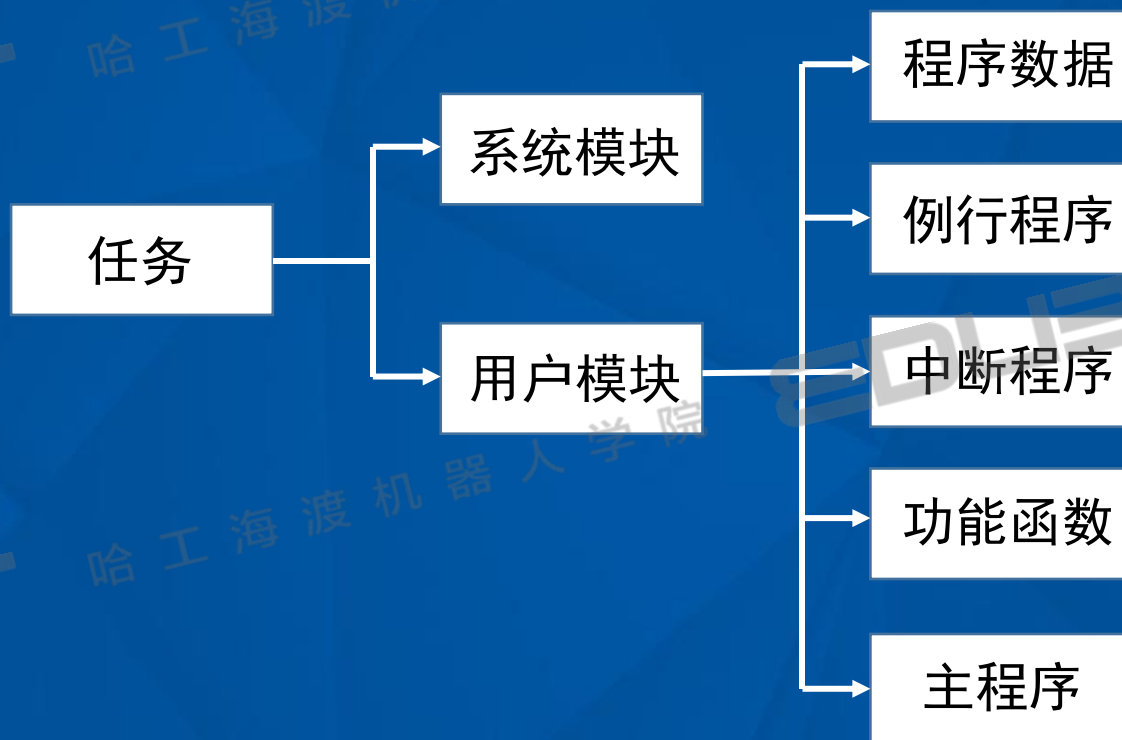
- 1. 掌握模块及例行程序的创建
- 2. 掌握动作指令的使用



6.1 模块及例行程序的创建

RAPID语言程序结构

ABB机器人程序包涵三个等级：任务、模块、例行程序，其结构如图所示。



6.2 数据类型

数据类型

ABB机器人的程序数据共有76个，并且可以根据实际情况进行程序数据的创建，为ABB机器人的程序设计带来了无限可能性。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。

范围: RAPID/T_ROB1		更改范围
accdata	aiotrigger	bool
btnres	busstate	buttondata
byte	cameradev	cameraextdata
camerasortdata	cameratarget	cfgdomain
clock	cnvcmd	confdata
confsupdata	corrdescr	cssframe
datapos	dionum	dir
dnum	egmident	errdomain

1 到 24 共 102

显示数据 视图

- bool - 逻辑值
- 字节 (byte) - 整数值0 - 255
- 时钟- 时间测量 (clock)
- intnum - 中断识别号
- loaddata - 加载数据
- num - 数值
- pos - 位置 (仅X、Y和Z)
- pose - 坐标变换
- robtargt - 位置数据
- signalxx - 数字和模拟信号
- speeddata - 速度数据
- string - 字符串
- tooldata - 工具数据
- wobjdata - 工件数据
- zonedata - 区域数据



程序数据存储类型

变量VAR：

变量型数据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，数据就会丢失。

可变量PRES：

可变量PRES最大的特点是，无论程序的指针如何，都会保持最后赋予的值。在机器人执行的RAPID程序中也可以对可变量存储类型数据进行赋值操作，在程序执行以后，赋值的结果会一直保持，直到对其进行重新赋值。

常量CONST：

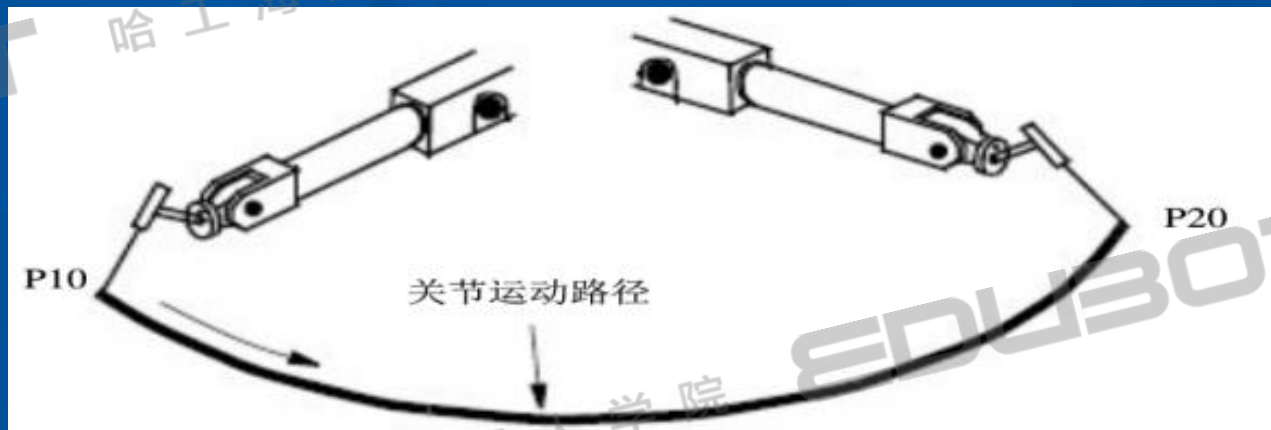
常量CONST的特点是在定义时已赋予了数值，并不能再程序中进行修改，除了手动修改。



6.4 动作指令解析

运动指令-MoveJ

机器人以最快捷的方式运动至目标点，机器人运动状态不完全可控，但运动路径保持唯一，运动中不会有死点，常用于机器人在空间大范围移动。



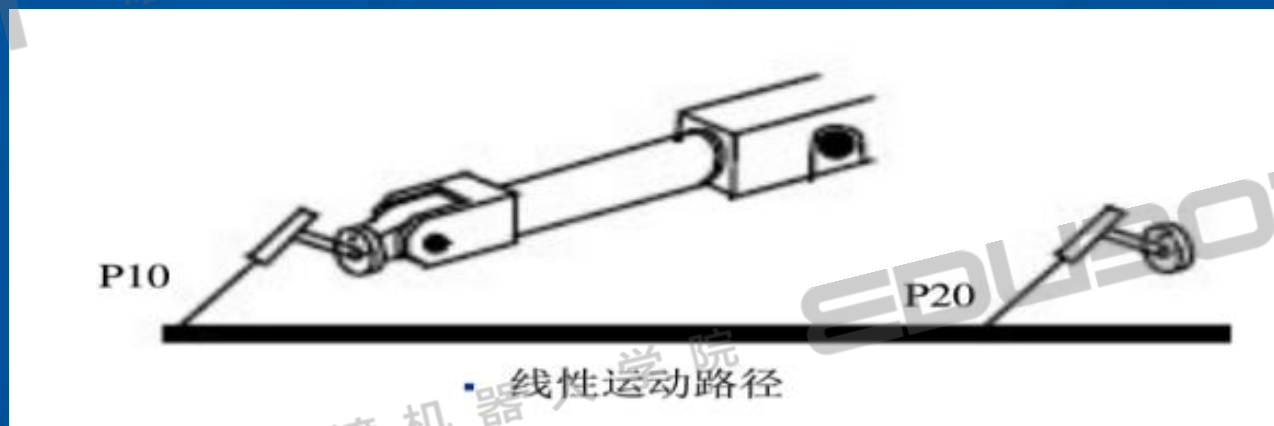
```
MoveJ p10, v1000, z50, tool0\WObj:=wobj0;
```



6.4 动作指令解析

运动指令-MoveL

机器人以线性移动方式运动至目标点，当前点与目标点两点决定一条直线，机器人运动状态可控制，运动路径唯一，可能出现死点，常用于机器人在工作状态移动。



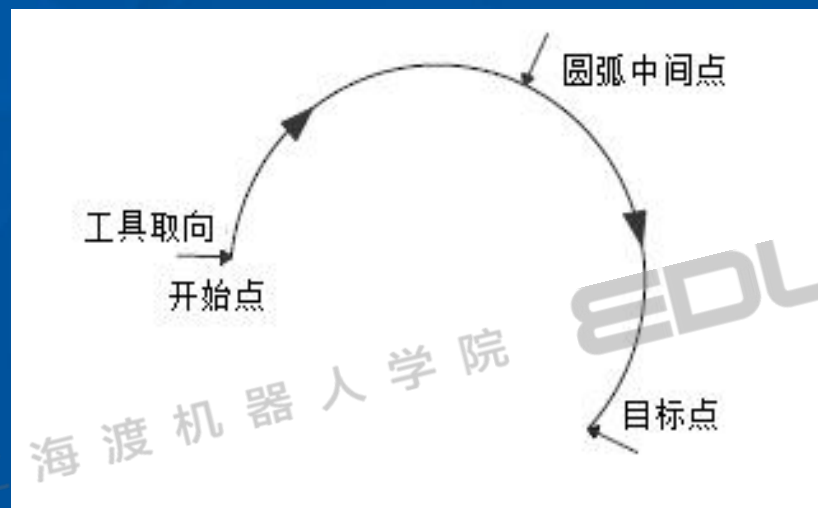
```
MoveL p20, v1000, z50, tool10\WObj:=wobj0;
```



6.4 动作指令解析

运动指令-MoveC

机器人通过中间点以圆弧移动方式运动至目标点，当前点、中间点与目标点三点决定一段圆弧，机器人运动状态可控制，运动路径保持唯一，常用于机器人在工作状态移动。



```
MoveC p30, p40, v1000, z10, tool0\Wobj:=wobj0;
```



第7章 机器人基本指令

教学内容：

- 7.1 I/O控制指令解析
- 7.2 程序控制指令
- 7.3 流程指令
- 7.4 功能指令
- 7.5 中断指令

学习目标：

- 1. 掌握I/O控制指令应用
- 2. 掌握程序控制指令应用
- 3. 掌握功能及中断应用



I/O控制指令

Set指令：设置数字输出信号

Set用于将数字输出信号的值设置为1，即打开数字输出信号。

```
Set do08_Laser;
```

Reset指令：重置数字输出信号

Reset用于将数字输出信号的值重置为0，即关闭数字输出信号。

```
Reset do08_Laser;
```

WaitDI：等待直至已设置数字输入信号

WaitDI用于一直等待数字输入信号，当等待信号条件成立时执行下面程序，否则一直等待。

```
WaitDI di02_start, 1;
```



7.2 程序控制指令

程序控制指令-ProcCall

通过ProcCall指令将程序指针移至对应的例行程序并开始执行，
执行完例行程序指针返回到调用位置，执行后续指令。



通过ProcCall指令调用例行程序

Init，即在程序的开始执行init（初始化）程序。然后依次执行Routine1、Routine2。



条件指令-IF

IF指令：IF条件判断。根据是否满足条件，执行不同的指令时，使用IF。


```
2  PROC main()
3      init;
4      IF di_jiance = 1 THEN
5          ssdby;
6      ELSE
7          jichu;
8      ENDIF
9  ENDPROC
```

程序指针移动到IF时开始做条件判断，如果di_jiance=1时执行例行程序ssdby（输送带搬运模块），如果di_jiance=0时执行例行程序jichu（基础模块）



TEST指令

TEST指令：当前指令通过判断相应数据变量与其对应的值，控制需要执行的相应指令



```
4 TEST count
5 CASE 1:
6   reg1 := reg1 + 1;
7 CASE 2:
8   reg1 := reg1 + 2;
9 DEFAULT:
10  reg1 := reg1 + 3;
11 ENDTEST
12 ENDPROC
```

根据count值执行相应case，没有对应值则执行default。



流程指令-**WHILE/**FOR

➤ **WHILE**:判断指令

如果条件满足，则重复执行对应程序，例如：

```
WHILE Reg1 <reg2 Do
```

```
Reg1 := Reg1+ 1;
```

```
ENDWHILE
```

如果变量Reg1<reg2条件成立，则一直重复执行，Reg1加 1，直到条件不满足为止。

➤ **FOR**:循环指令

根据指定的次数，重复执行对应的

程序，例如：

```
FOR i FROM 1 TO 10 DO
```

```
Routine1;
```

```
ENDFOR
```

重复执行10次routine1里的程序。



功能函数-CRobT

读取当前位置功能函数CRobT ()

读取当前机器人TCP位置数据进行赋值。例如：

```
VAR      robtarget          p10;
```

```
P10     := CRobT (\Tool := tool1\Wobj := wobj0);
```

读取当前机器人TCP位置数据，指定工具数据为tool1，工件坐标数据为wobj0（若不指定，则默认工具数据为tool0，默认工件坐标系数据为wobj0），之后将读取的目标点数据赋值给p10。



功能函数-Offs

■ 工件坐标偏移函数Offs ()

采用函数offs, 反馈一个参变量, Offs (p, x, y, z) 代表一个离p1点X轴偏差量为x, Y轴偏差量为y, Z轴偏差量为z。

```
MoveI offs (P1, 100, 50, 0), v100, z100, tool0;
```

P1开始位置;

100表示以开始位置向X轴方向偏移100mm;

50表示以开始位置向Y轴方向偏移50mm;

0表示以开始位置向Z轴方向没有偏移。



中断功能意义

什么叫中断？有哪几种不同类型的中断？

由于某个事件的发生，CPU暂停当前正在执行的程序，转而执行处理该事件的一个程序。该程序执行完成后，CPU接着执行被暂停的程序。这个过程称为中断。根据中断源的位置，有两种类型的中断。有的中断源在CPU的内部，称为内部中断。大多数的中断源在CPU的外部，称为外部中断。



中断功能解析

主要介绍计数指令，IDelete、CONNECT、ISignalDI、Iwatch、ISleep等四种指令。

VAR intnum intno1; (定义中断数据intno1)

IDelete intno1; (取消当前中断符intno1的连接，预防误触发。)

CONNECT intno1 WITH Trap1; (将中断符与中断程序Trap1连接。)

ISignalDI di1,1, intno1; (当输入信号di1为1时，触发该中断程序。)

IWatch: 激活中断监控。系统启动后默认为激活状态，只要中断条件满足，即会触发中断。

IWatch intno1;

ISleep: 使中断监控失效，在失效期间，该中断程序不会被触发

ISleep intno1;



第8章 系统事件

教学内容：

- 8.1 系统事件意义
- 8.2 系统事件类别

学习目标：

- 1. 了解系统事件的意义
- 2. 掌握系统事件的应用



8.1 系统事件意义

机器人系统事件

机器人系统事件：当系统发生一个动作时触发相应的例行程序，事件界面如图所示：



点击【主菜单】，
选择【控制面板】，
选择【配置】，
进入【I/O System】界面。
点击【主题】，
点击【Controller】，
选择【Event Routine】。



8.2 系统事件解析

系统事件类别

序号	事件	说明
1	POWER_ON	系统上电或重启触发事件，执行对应的例行程序。
2	START	按下启动按钮或外部启动信号触发事件，执行对应的例行程序。
3	STOP	按下停止按钮或外部停止信号触发事件，执行对应的例行程序。
4	QSTOP	机器人迅速停止（即紧急停止）触发事件，执行对应的例行程序。



8.2 系统事件解析

系统事件类别

序号	事件	说明
5	RESTART	从停止位置开始执行时触发事件，执行对应的例行程序。
6	RESET	先关闭，然后用示教器载入一则新程序触发事件，执行对应的例行程序。
7	STEP	步进或步退触发事件，执行对应的例行程序。



第9章 编程实例

教学内容：

- 9.1 基础模块编程
- 9.2 输送带模块编程
- 9.3 码垛搬运模块编程

学习目标：

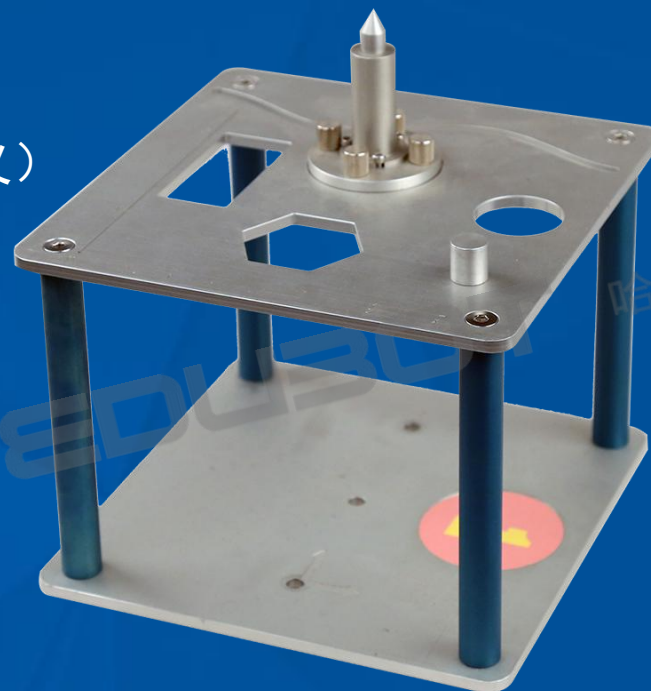
- 1. 掌握基本模块编程方法
- 2. 掌握输送带模块编程思路
- 3. 掌握码垛搬运模块功能



基础模块编程

控制要求：

- 1、创建所需的模块及例行程序（名称自定义）
- 2、建立工具坐标系 tool10
- 3、建立安全点Phome
- 4、利用激光器雕刻模块上相应的图形，机器人末端激光到达每个图形第一点时激光ON，结束时OFF。完成每个相应的图形雕刻。



9.2 输送带模块

输送带模块

控制要求：

- 1、创建所需的模块及例行程序（名称自定义）
- 2、建立工件坐标系 wobj10
- 3、建立安全点Phome
- 4、如果输送带光电检测有信号机器人将圆饼从输送带一端搬运至另一端，如果没有检测信号机器人执行基础模块。



9.3 码垛搬运模块

码垛搬运模块

控制要求：

- 1、创建所需的模块及例行程序（名称自定义）
- 2、建立工件坐标系 wobj10
- 3、建立安全点Phone
- 3、利用工件坐标偏移函数offs（）将圆饼从1号位搬至4号回到phone点后再从4号位将圆饼搬至6号位，回到phone点结束。

注：整个过程只容许建立一个位置数据“P_10”



第10章 离线仿真

教学内容：

10.1 自动路径规划

10.2 smart组件

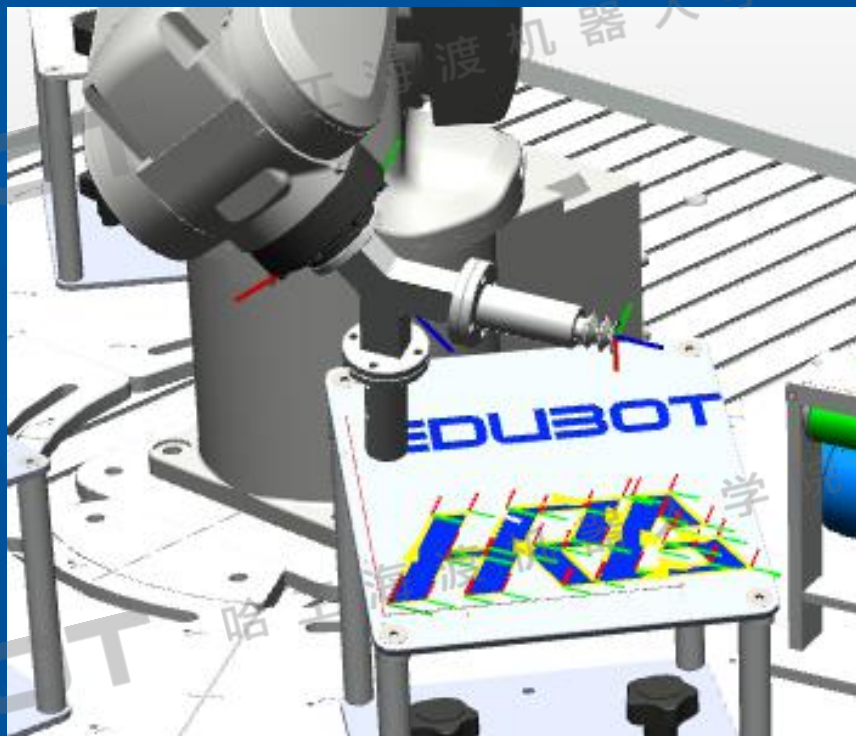
学习目标：

1. 掌握自动路径规划

2. 掌握smart组件的应用



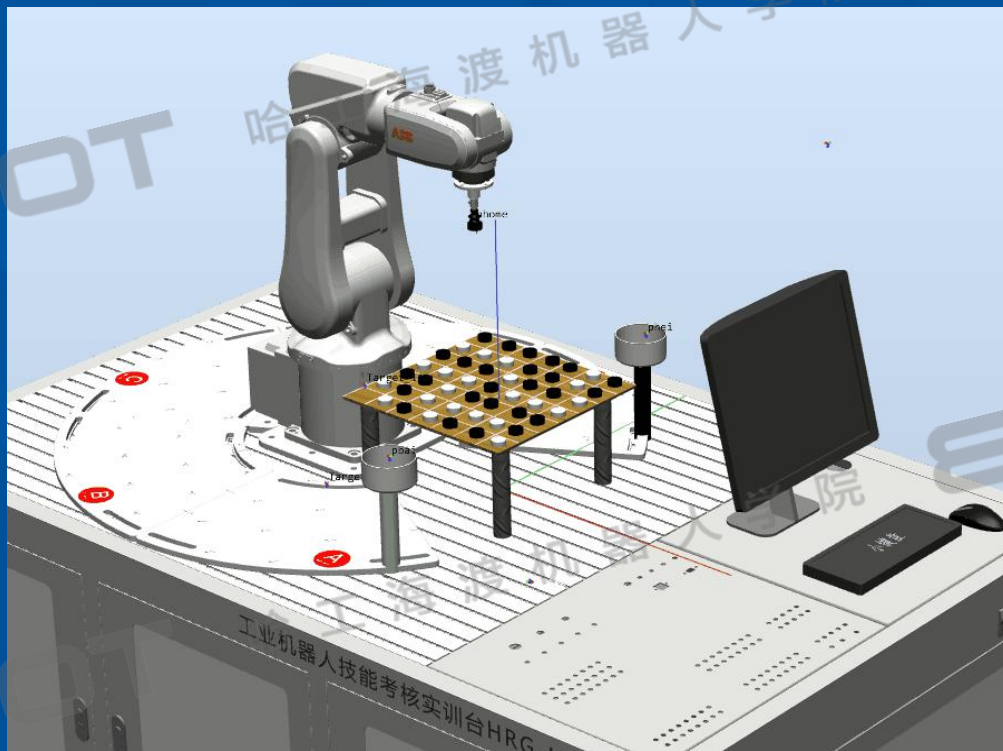
自动路径规划



完成复杂的轨迹我们一般使用自动规划行走路径，人工示教比较复杂。



Smart组件应用



Smart组件功能能够让机器人末端吸盘吸取和释放托盘上的棋子。





扫一扫关注“哈工海渡”
回复“惊喜”获得更多免费课程

网址：www.irobot-edu.com
邮箱：edubot_zhang@126.com

《工业机器人入门实用教程（ABB机器人）》

THANK YOU

EDUBOT
哈工海波机器人学院

www.irobot-edu.com
edubot_zhang@126.com