

实训项目二：实训知识点

- 1、在 ROBOTSTUDIO 中进行机器人及周边设备的合理布局。
- 2、机器人 IO 信号的设定与链接
- 3、Smart 组件的基本使用
- 4、机器人轨迹的创建
- 5、仿真的调试。

机器人工作站的布局

所有的部件已包含在打包文件中。双击打开打包文件后，请按照以下的图 1 中所示进行布局。

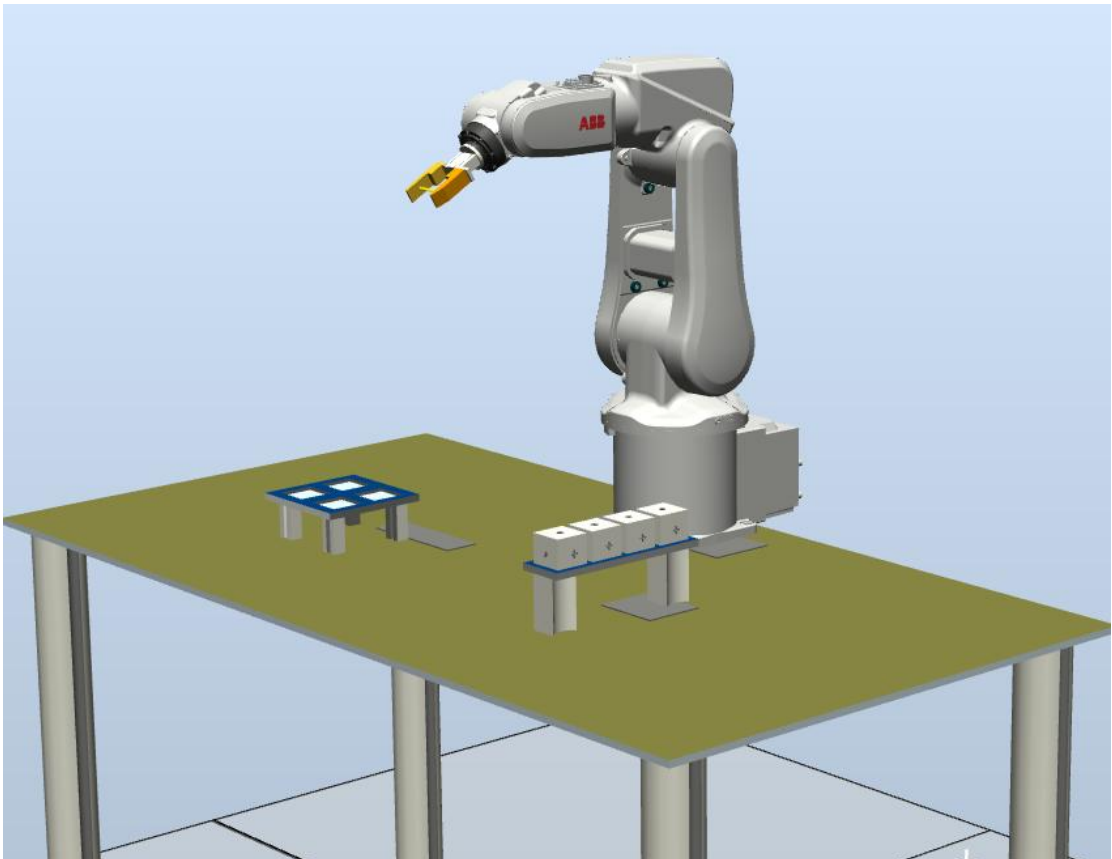


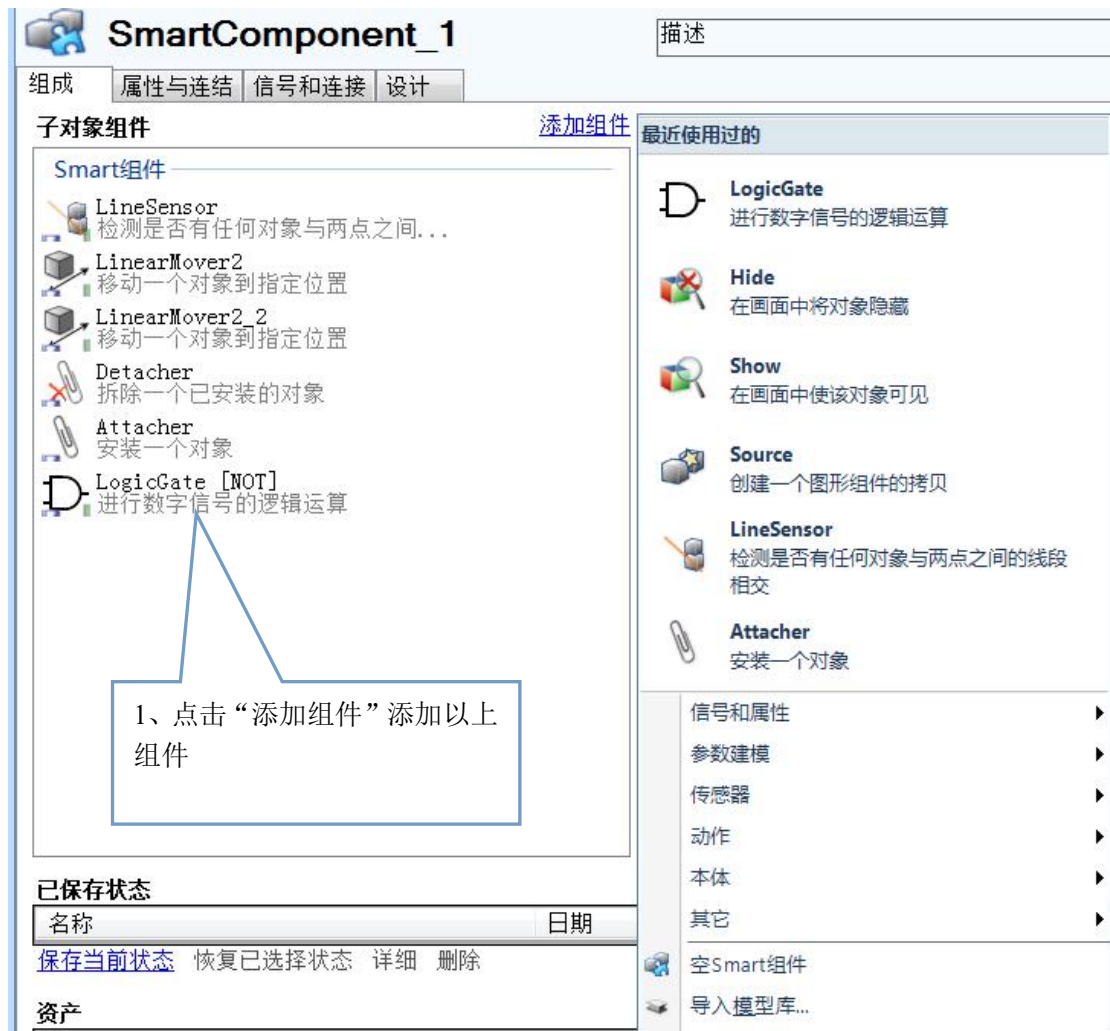
图 1 机器人工作站的布局示意

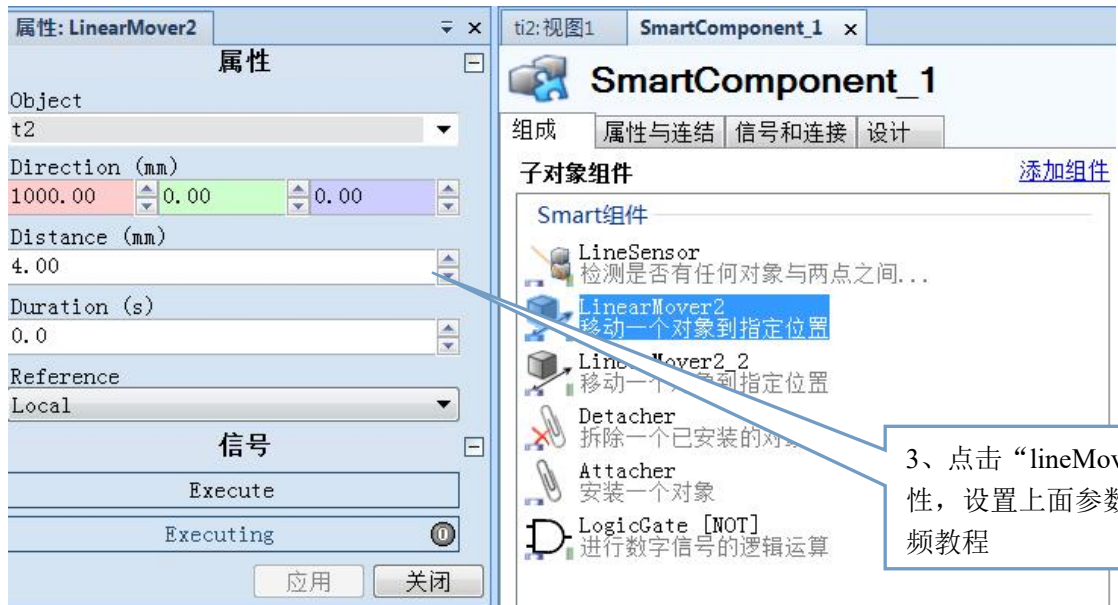
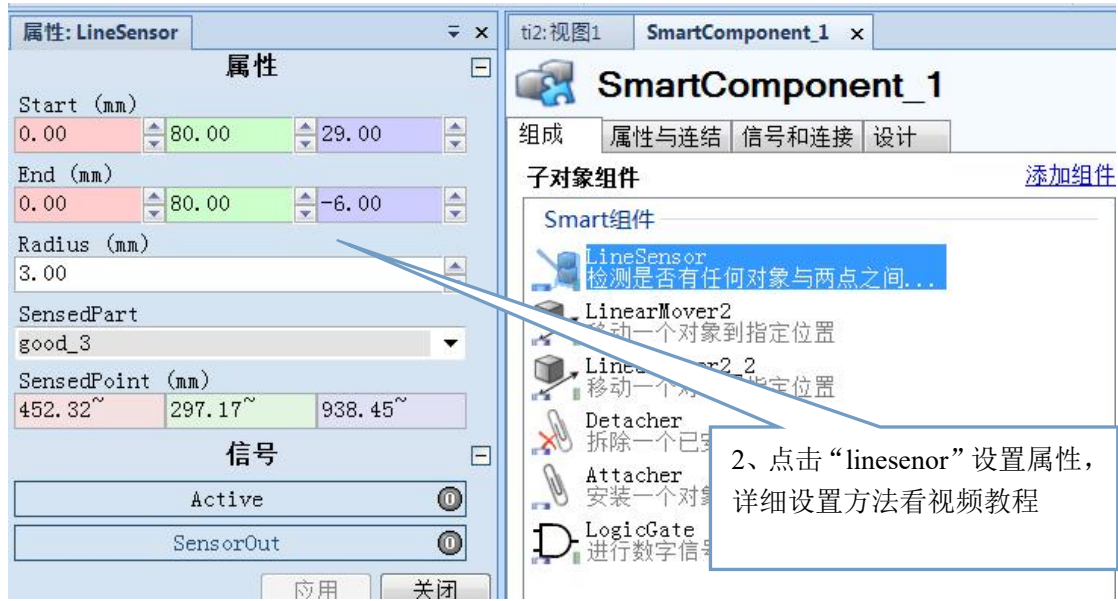
要注意的问题：

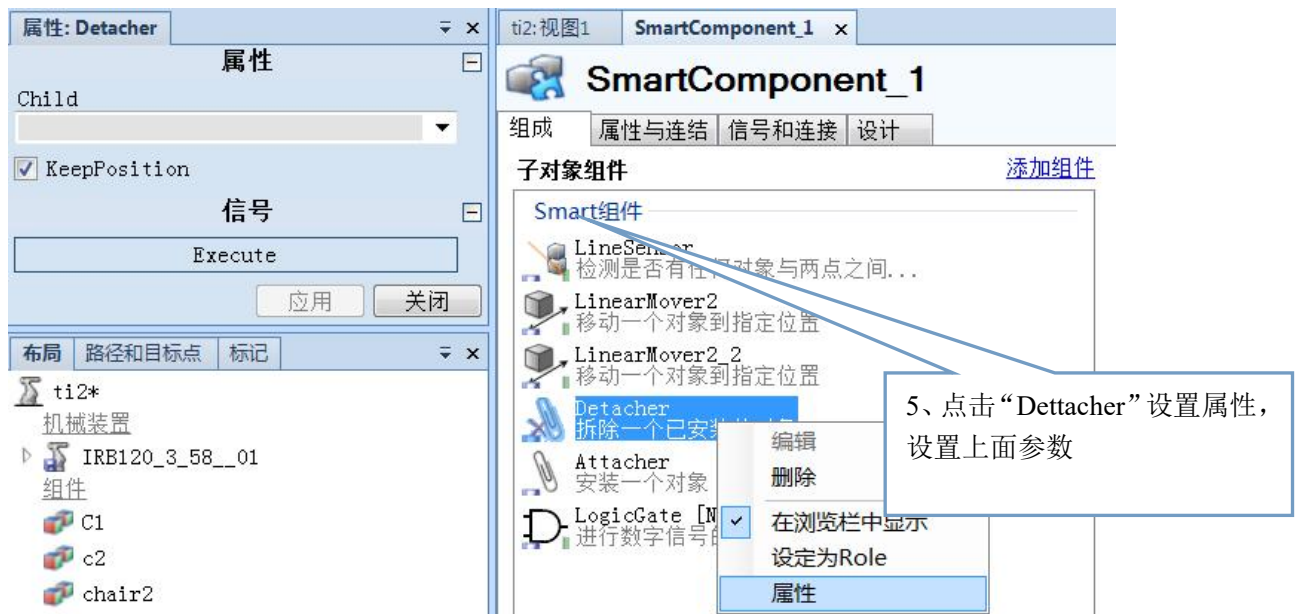
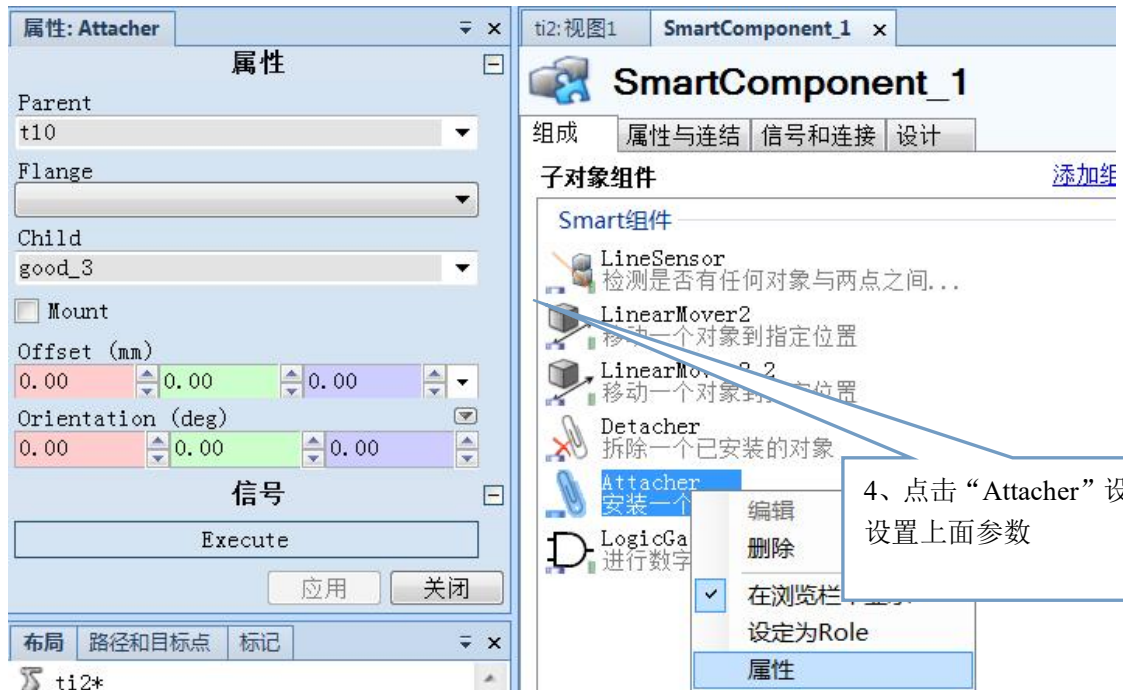
- 1、机器人与周边的部件的位置要合理，周边部件应在机器人的工作范围的中间位置为佳。
- 2、可以对机器人的操作，以确认机器人可以到达要取、放的最远端是可以顺利到达的，否则以后再调整就会很麻烦了。

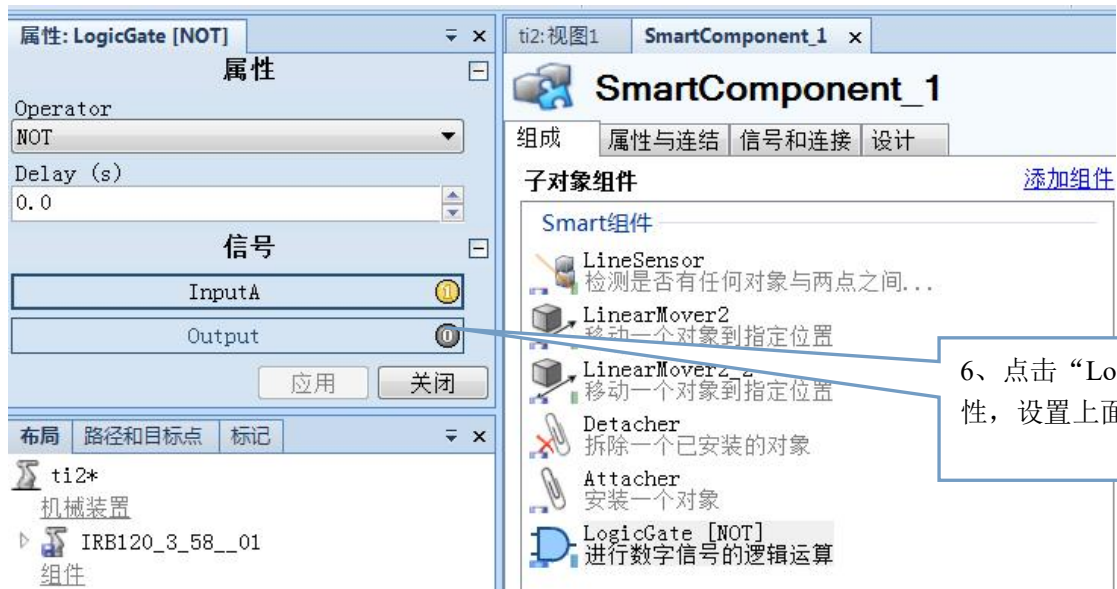
Smart 组件的基本使用

smart 组件创建操作：建模---smart 组件---添加组件，建立如下表组件，详情看视频。



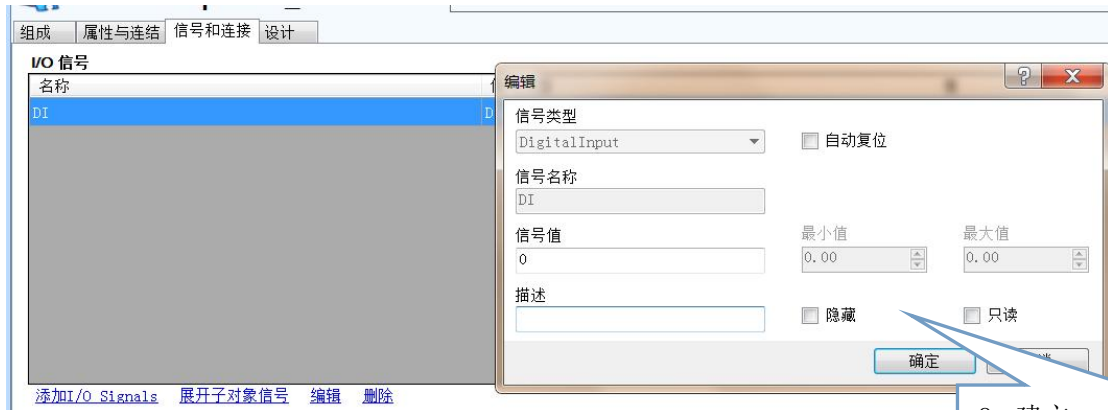






7、添加左边面两个属性与连接

源对象	源属性	目标对象	目标属性
LineSensor	SensedPart	Attacher	Child
Attacher	Child	Detacher	Child



8、建立一个 smart 组件的输入信号



9、对信号和组件进行关联

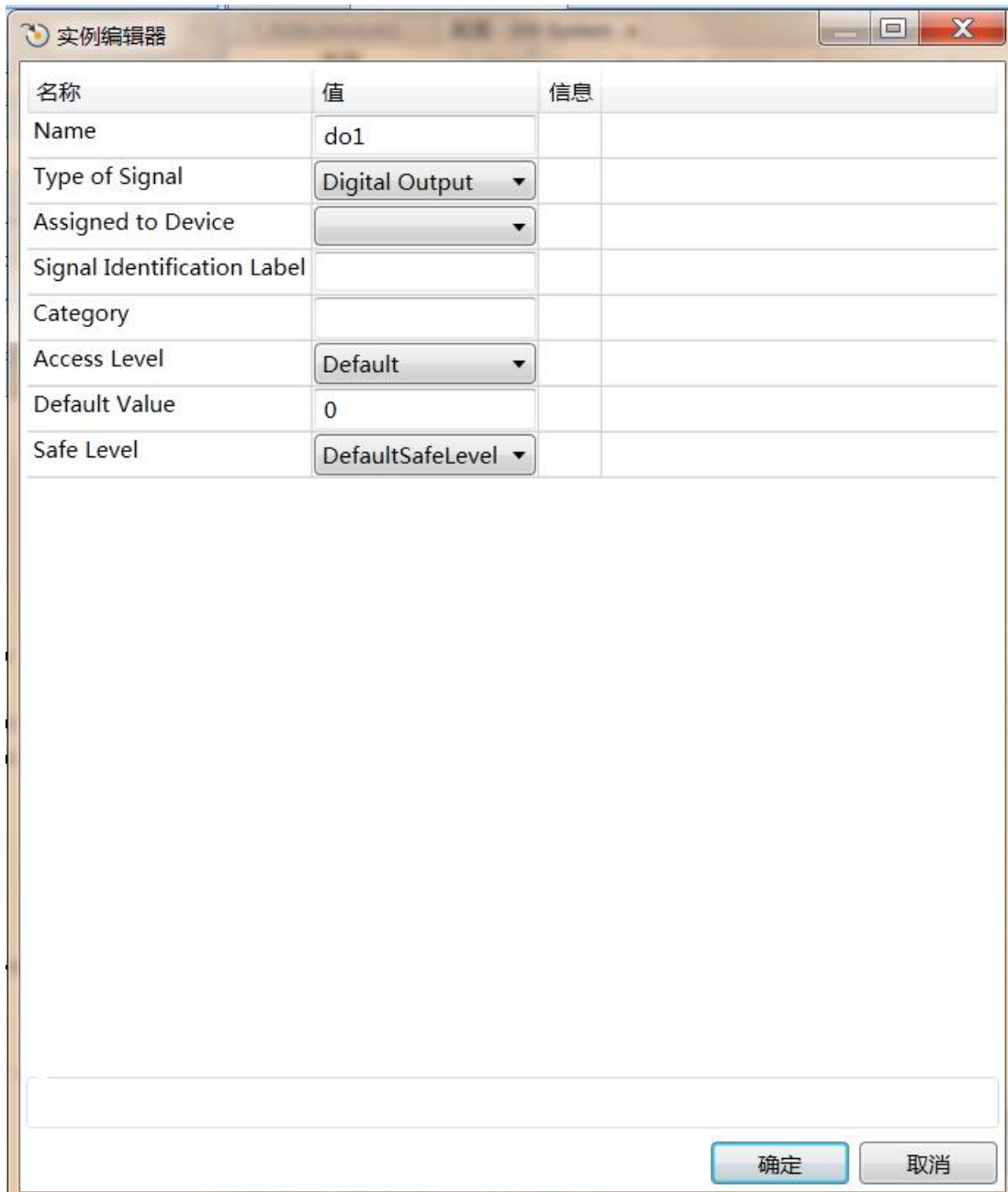
源对象	源信号	目标对象	目标对象
SmartComponent_1	DI	LineSensor	Active
LineSensor	SensorOut	Attacher	Execute
SmartComponent_1	DI	LinearMover2	Execute
SmartComponent_1	DI	LogicGate [NOT]	InputA
LogicGate [NOT]	Output	LinearMover2_2	Execute
LogicGate [NOT]	Output	Detacher	Execute

10、建立以下信号关联

机器人 IO 的设定

为了实现真夹具动作的夹/放的动作控制，为了至少需要设定一个虚拟的数字输出信号，这个信号只用于虚拟仿真的作用，并没有与实际的总线或 IO 板进行关联。

数字输出信号的设定菜单操作为：控制器---配置编辑器---IO SYSTEM---SIGNAL。然后将信号设定为以下的表 1 的参数：



名称	值	信息
Name	do1	
Type of Signal	Digital Output	
Assigned to Device		
Signal Identification Label		
Category		
Access Level	Default	
Default Value	0	
Safe Level	DefaultSafeLevel	

机器人轨迹的创建

机器人的动作是从左侧的码垛盘存放处搬运到右边的的方形的码垛盘处。

具体的操作方法如下：

1、设置正确的工件坐标与工具，如下图所示：



2、根据具体的情况，设定正确的机器人运动指令的参数，如下图所示：

3、根据动作的要求通过示教指令的方法，创建对应的轨迹程序，程序样板如下图所示：

MODULE Module1

CONST robtarget

Target_70:=[[-224.011205552,135.527108744,487.704338612],[0.353553313,-0.61237248,0.61237248,0.353553313],[-1,0,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

plz11:=[[37.869880471,44.814983349,115.908038328],[0.001652216,0.707104385,-0.707105313,0.00165409],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pla1:=[[37.8700052,44.815136086,39.490813564],[0.001651816,0.70710452,-0.707105178,0.001654058],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi1:=[[-489.951104876,140.965986471,162.762733411],[0.001652044,0.7071057,-0.707103996,0.001655228],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi1:=[[-489.951183238,140.966082149,59.407584776],[0.001651979,0.707105621,-0.707104075,0.00165472],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pla22:=[[37.869942427,116.557143713,155.39126366],[0.001651852,0.707104876,-0.707104822,0.001654266],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pla2:=[[37.869940738,116.557190745,39.490847993],[0.001651699,0.707104729,-0.707104969,0.001654142],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi22:=[[-489.951259604,80.653337944,156.122050005],[0.001652015,0.707105581,-0.707104116,0.001654666],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi2:=[[-489.951173225,80.653256714,59.407299329],[0.001651987,0.707105682,-0.707104014,0.001654959],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pla33:=[[115.738761397,44.409702788,133.350771359],[0.001651405,0.707105229,-0.707104447,0.001654215],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pla3:=[[115.738591832,44.409570256,39.490752672],[0.00165187,0.707105066,-0.707104632,0.001654128],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi33:=[[-489.951153989,20.789766822,176.305175859],[0.00165229,0.707105496,-0.707104197,0.001655658],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget

pi3:=[[-489.951229613,20.789932408,59.407225593],[0.00165209,0.707105525,-0.70710417,0.001655202],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];

CONST robtarget


```
pla44:=[[115.738762369,116.557191188,146.072597057],[0.00165153,0.707104774,
-0.707104925,0.001654045],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
CONST robtarget
```

```
pla4:=[[115.738726819,116.557121339,39.490732226],[0.00165192,0.707104783,-0.
707104915,0.001654139],[-1,-1,-1,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
CONST robtarget
```

```
pi44:=[[-489.951097873,-38.971314643,204.450846504],[0.001651847,0.707105696,
-0.707103999,0.001655135],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
CONST robtarget
```

```
pi4:=[[-489.951092308,-38.971402841,59.407056847],[0.00165204,0.707105435,-0.
70710426,0.001655285],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
PROC Path_10()
```

```
ENDPROC
```

```
PROC Path_20()
```

```
MoveJ Target_70,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ plz11,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pla1,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Set do1;
```

```
WaitTime 0.3;
```

```
MoveJ plz11,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ pi11,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pi1,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Reset do1;
```

```
WaitTime 0.3;
```

```
MoveJ pi11,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ pla22,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pla2,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Set do1;
```

```
WaitTime 0.3;
```

```
MoveJ pla22,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ pi22,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pi2,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Reset do1;
```

```
WaitTime 0.3;
```

```
MoveJ pi22,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ pla33,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pla3,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Set do1;
```

```
WaitTime 0.3;
```

```
MoveJ pla33,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveJ pi33,v1000,z100,Tooldata_1\WObj:=Workobject_1;
```

```
MoveL pi3,v1000,fine,Tooldata_1\WObj:=Workobject_1;
```

```
Reset do1;
```

```

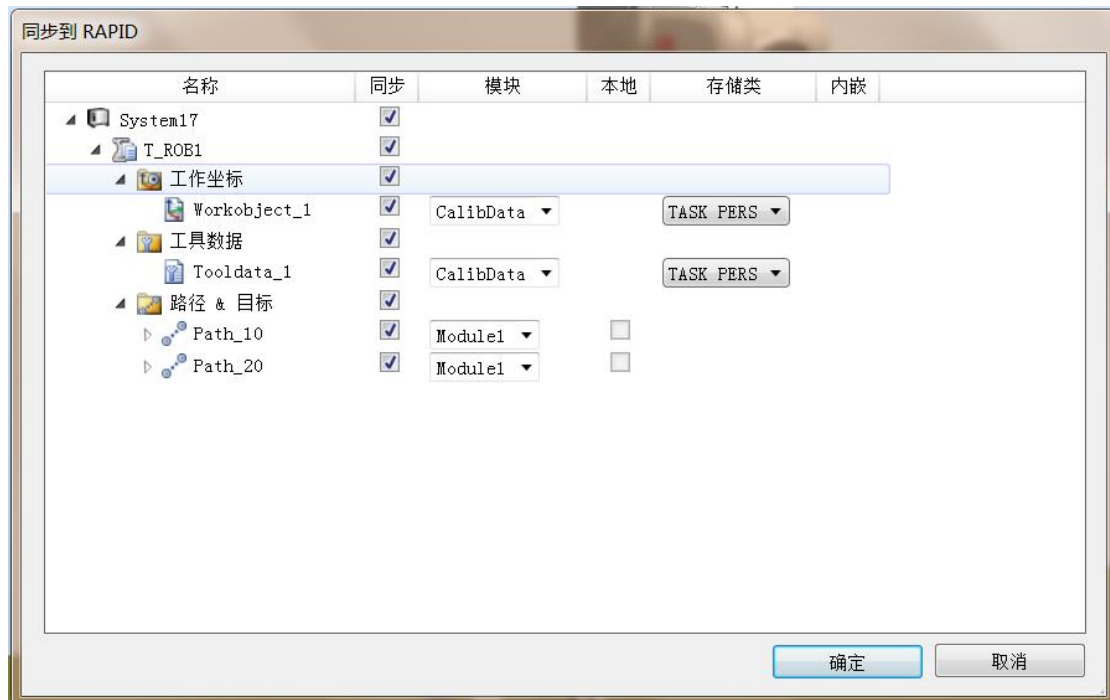
WaitTime 0.3;
MoveJ pi33,v1000,z100,Tooldata_1\WObj:=Workobject_1;
MoveJ pla44,v1000,z100,Tooldata_1\WObj:=Workobject_1;
MoveL pla4,v1000,fine,Tooldata_1\WObj:=Workobject_1;
Set do1;
WaitTime 0.3;
MoveJ pla44,v1000,z100,Tooldata_1\WObj:=Workobject_1;
MoveJ pi44,v1000,z100,Tooldata_1\WObj:=Workobject_1;
MoveL pi4,v1000,fine,Tooldata_1\WObj:=Workobject_1;
Reset do1;
WaitTime 0.3;
MoveJ pi44,v1000,z100,Tooldata_1\WObj:=Workobject_1;
MoveJ Target_70,v1000,z100,Tooldata_1\WObj:=Workobject_1;

```

ENDPROC

ENDMODULE

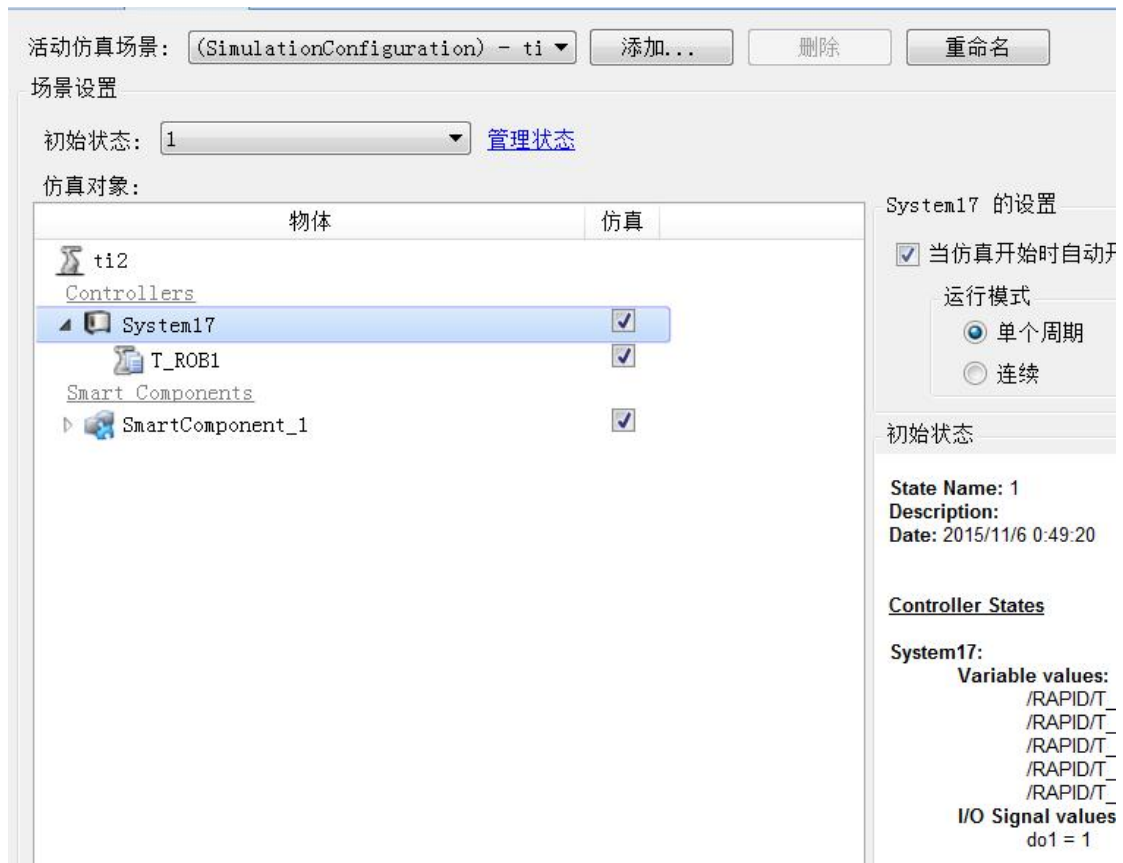
将写好的程序同步到RAPID，菜单操作：基本---同步---同步到RAPID，如下图所示：



仿真的调试

在完成了设置与编程以后，先保存初始状态，接着下来就是要验证一下仿真动画的结果了，具体的操作如下：

1、设定要运行的 RAPID 子程序，在本项目中是 PATH20，菜单操作如下：仿真---仿真设定---指定 PATH20，如下图所示：



2、点击仿真菜单中的“播放”就可以看到动画效果了。动画结束后，点击“重置”，恢复到原来的状态。