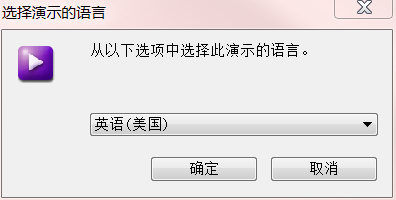
工业机器人编程与实践实验指导书

Robotstudio 的安装

1. 在目录里面找到Launch 图标，双击：

C:\Users\Administrator\AppData\Roaming\Tencent\Users\94026448\QQ\WinTemp\RichOle\%]7Q)H$8PFV`]}83[@0%@_D.png

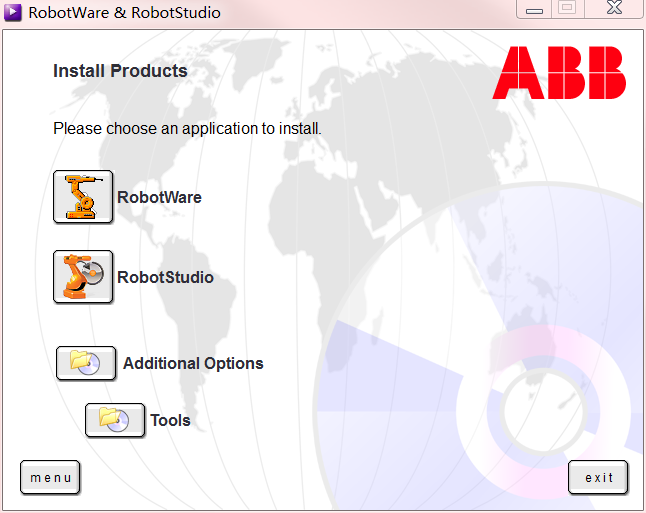
1. 选择安装语言



1. 选择install products：

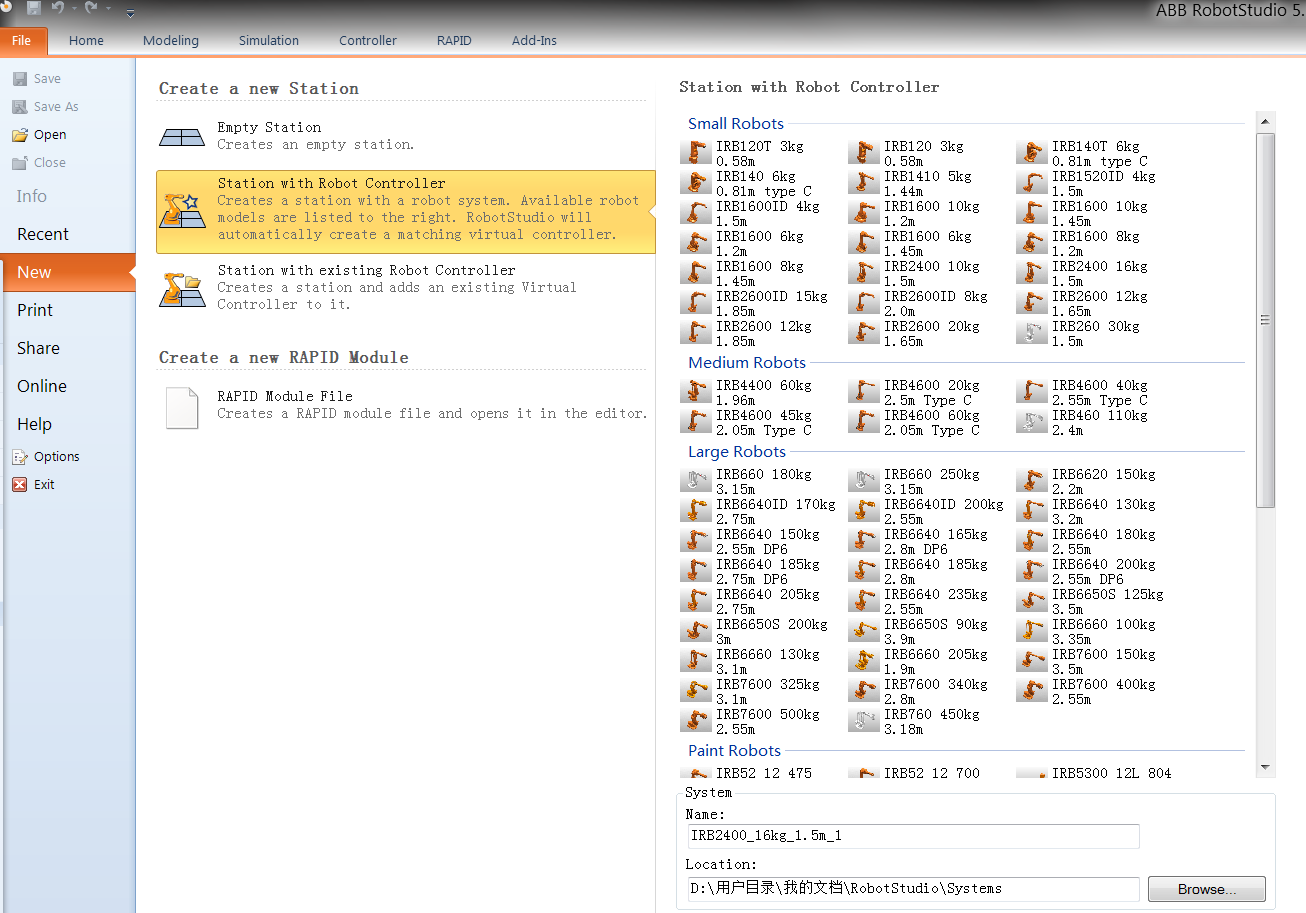


1. 首先安装robotware，安装完毕后再安装robotstuido

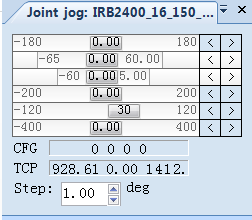
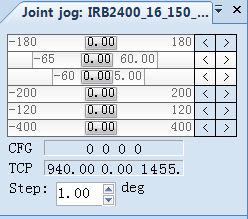


实验一：

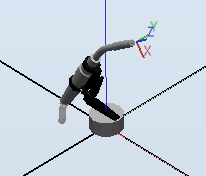
1. 创建station：打开robotstudio，在new选项下点击station with robot controller，选择机器人型号irb2400 16kg 1.5m，点击creat创建工程。



1. 在左侧工具栏的irb2400\_16\_150\_\_02上右键，选择mechanism joint jog,第五轴的30度旋转调至0，以方便安装焊枪。

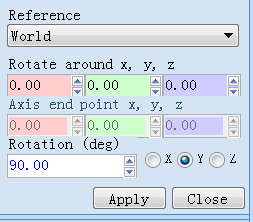
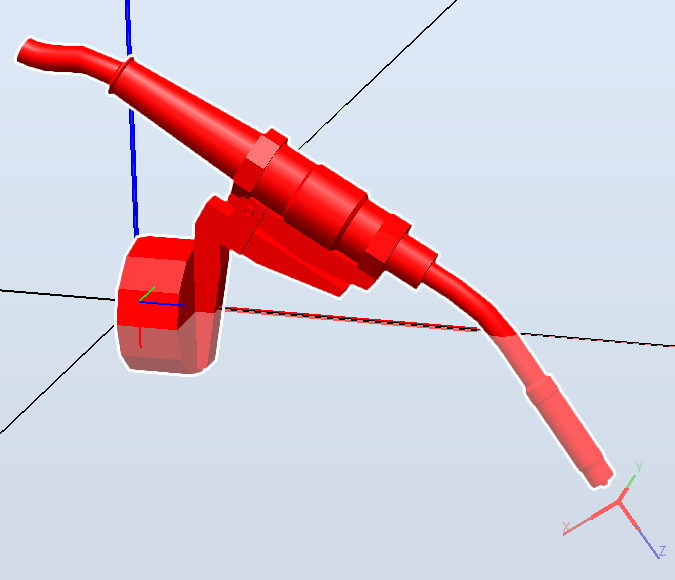
---🡪 

1. 在home菜单下的import library中的equipement中选择aw\_gun\_psf\_25，将该枪导入工作站中。在左侧工具栏的irb2400\_16\_150\_\_02上右键，将visible前面的对号取消，可以看到该枪出现在世界坐标系原点位置。

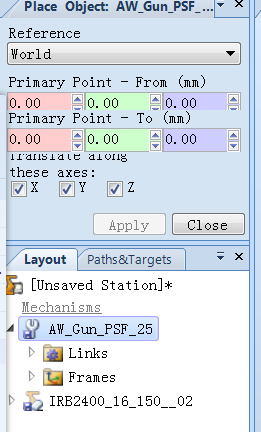


在工作站中点击左键，按住ctrl同时滑动鼠标滚轮可放大缩小查看。同时按住shift+control，按下鼠标左键不放并移动鼠标，可旋转查看。

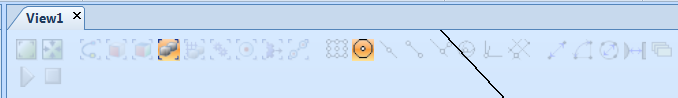
在aw\_gun\_psf\_25右键，点击rotate，在左边弹出的对话框中单击Y前面的圆圈，在文本框中输入90度，点击apply，得到枪沿y轴旋转90度后的姿态如图。使得喷枪底座与机器人六轴末端法兰盘平行，以方便装枪。

-🡪 

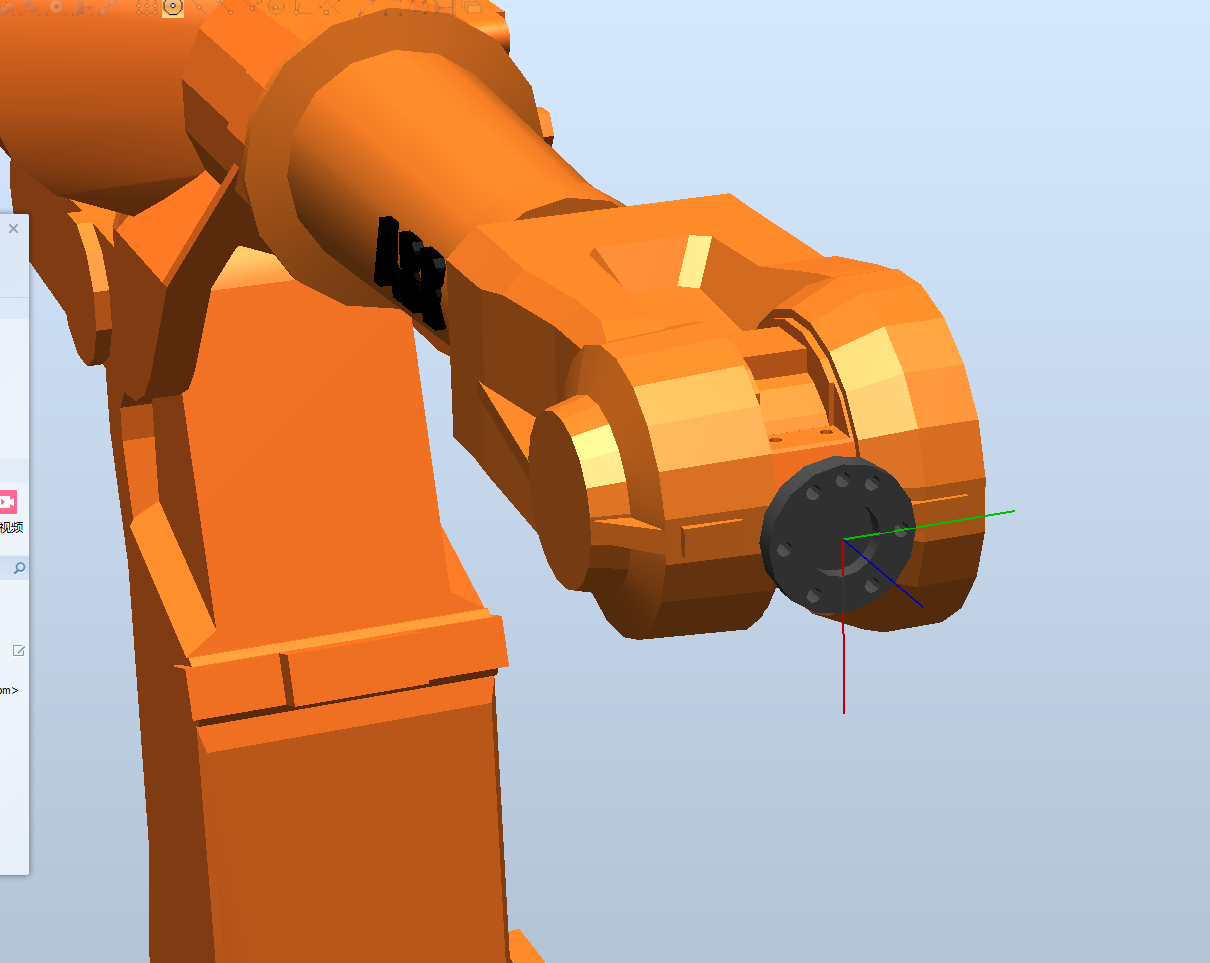
1. 将视角转换至枪底面，在aw\_gun\_psf\_25右键，选择place-one point,出现移动对话框。



点击primary point from下的空白文字框，然后点击图形环境中上方的snap center按钮，激活选择中心模式。

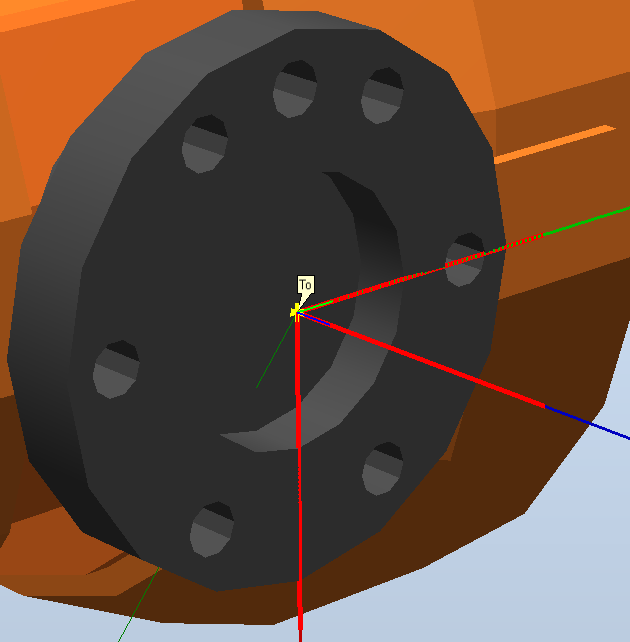


将鼠标移动到枪地面并点击，中心处出现白色圆球，同时左上方空白文字框处被赋值为0，0，0.

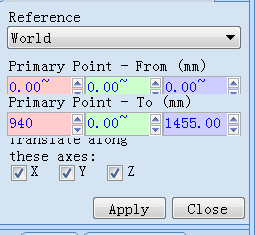
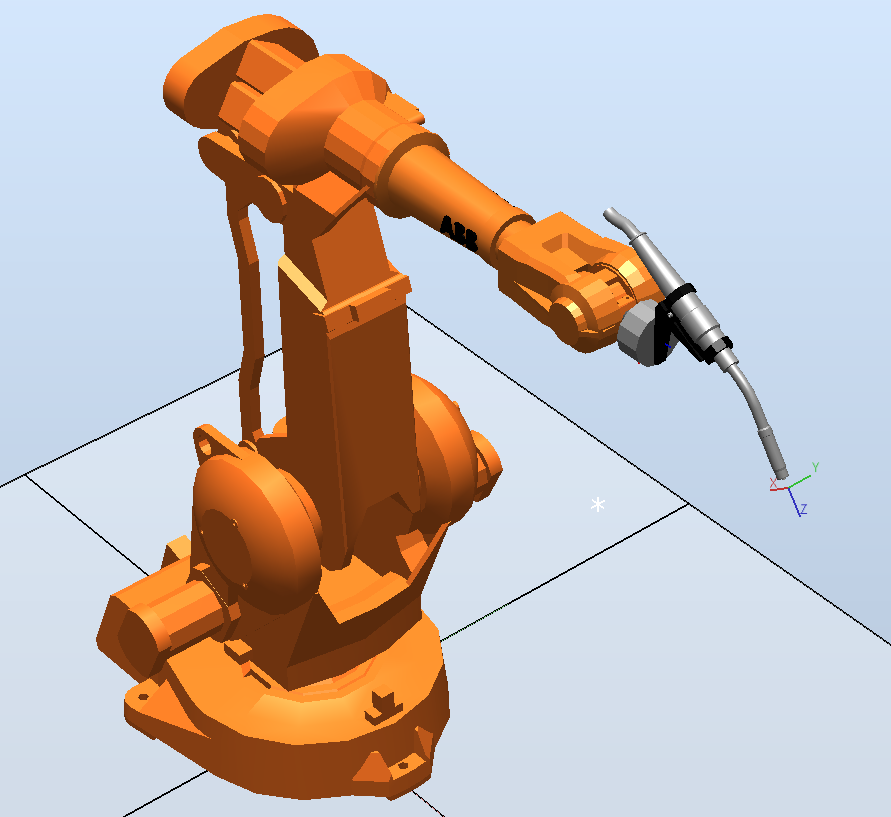
1. 将irb2400-16-150—02右键菜单的的visiable选中，调整机器人视角如下：
2. 

将左侧aw\_gun\_psf\_25名称上左键选中。

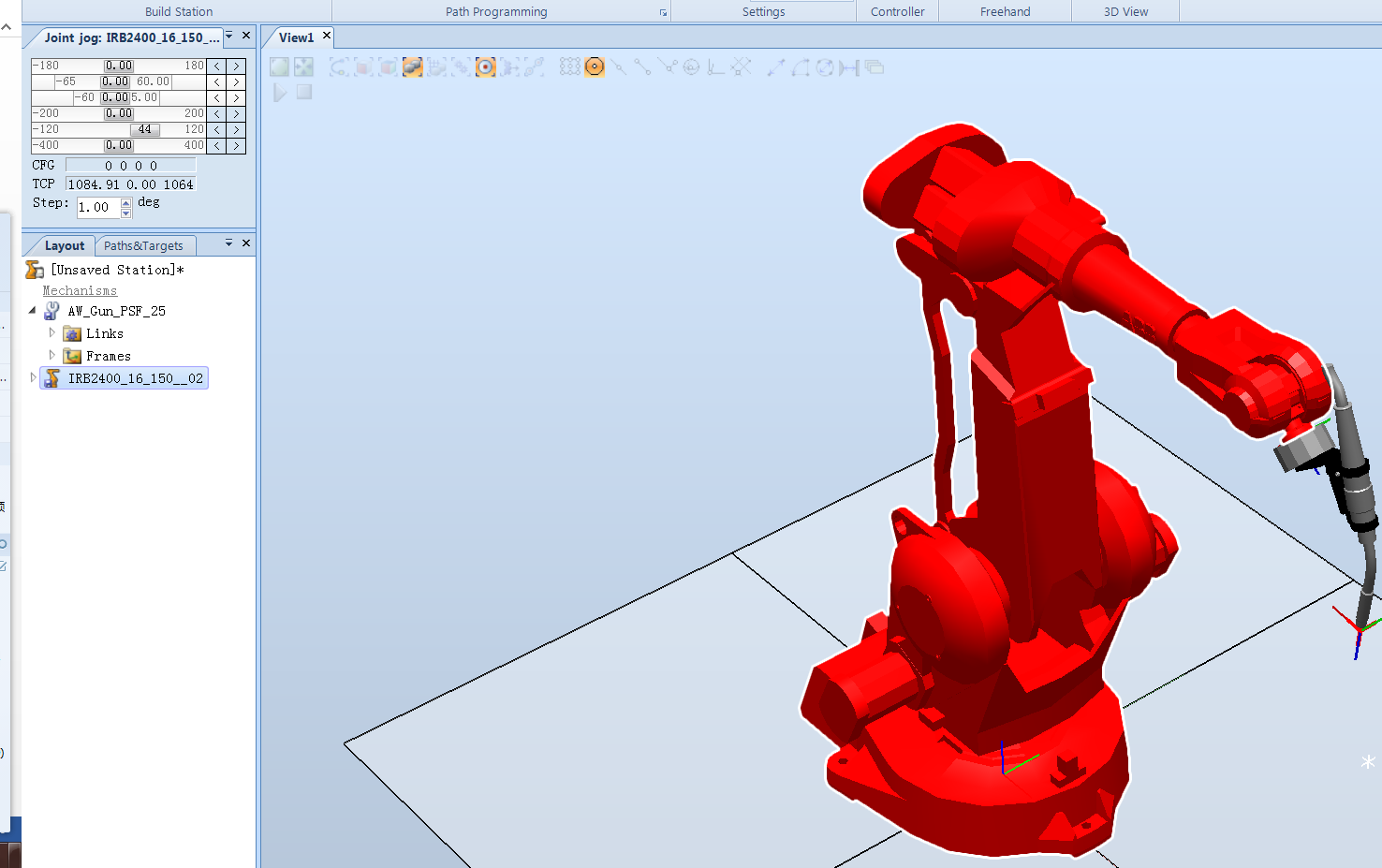
在左侧primary point-to 下方的空白文字框单击左键，同时点中图形环境上方的Target/frame selection 和 snap center，C:\Users\Administrator\AppData\Roaming\Tencent\Users\94026448\QQ\WinTemp\RichOle\_ZZ_OY@%DF_WW8O3L2RNJ91.png选择图中坐标原点作为移动终点。



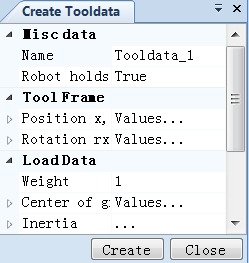
则左边工具栏的处被自动赋值。点击apply，枪被移动至目标位置。Close 左侧的place对话框。得到安装在机器人六轴末端的焊枪如下图所示。

--🡪 

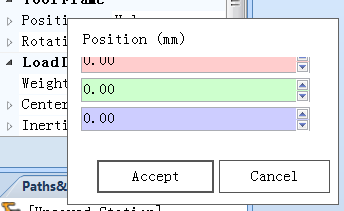
6． 在左侧aw\_gun\_psf\_25上右键，attach to- irb2400，在弹出的对话框中选择否。则枪被安装至机器人末端，可以随机器人一同动作，在左侧工具栏的irb2400\_16\_150\_\_02上右键，选择mechanism joint jog,第五轴的30度旋转调至40，可看到枪的姿态随之发生变化。将第五轴角度恢复至0度。



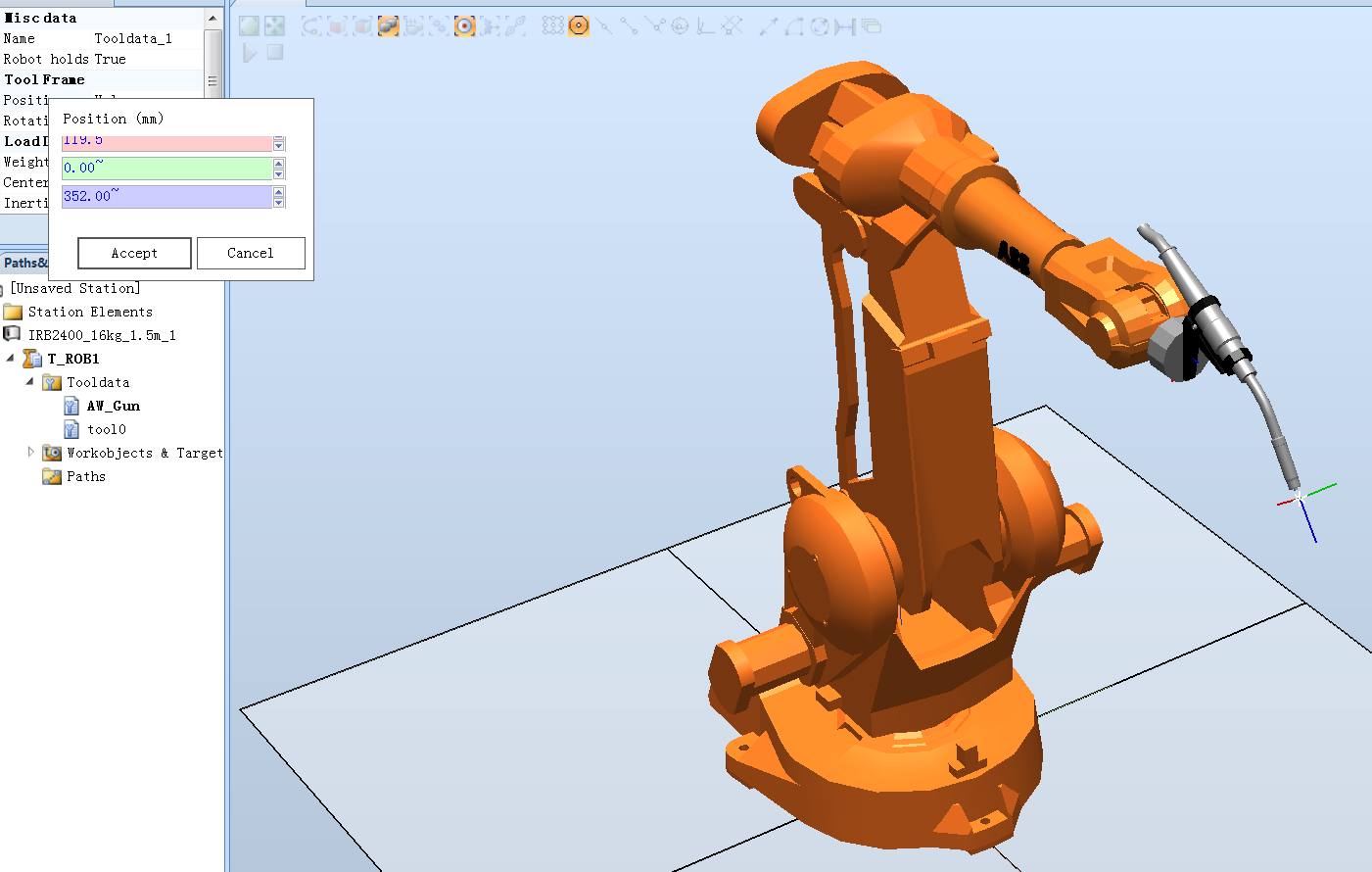
1. 在home菜单下的other中单击，选择creat tooldata，创建工具中心点。在左侧弹出对话框如下：



点击tool frame下的position x,values…，点击向下的箭头，出现tcp位置定义对话框。

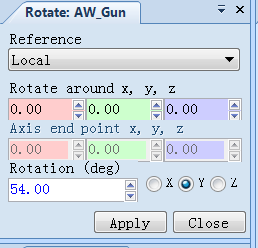


点击红色文本框中的0.00，并单击图形环境中的枪末端tcp坐标系，则TCP位置被自动赋值，如下。

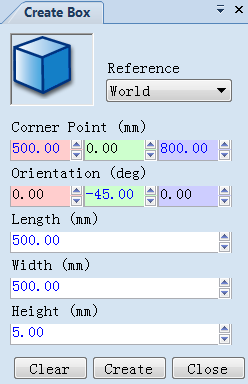


点击左侧的accept，点击creat tooldata下的creat。则左侧创建了名为Tooldata\_1的工具中心点。

1. 在Tooldata\_1上右键——rotate，选中Reference下的local选项，选中y前面的圆，同时在文本框出填入54，将TCP旋转至与aw\_gun重合。



1. 在modeling菜单下点击solid-box，创建立方体的尺寸如下：



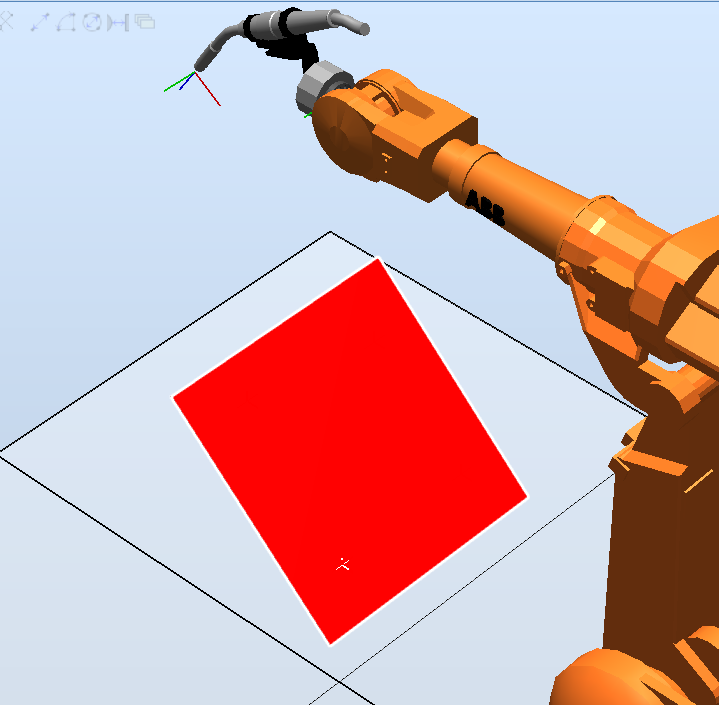
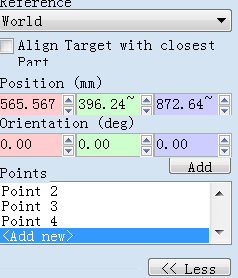
600

点击creat，创建立方体。

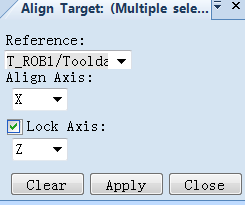
1. 点击图形环境上方的target/frameselection和snap center。



1. 在home菜单下的target下点击creat target，在position下的文本框单击后，在立方体上选取可构成四边形的四个点，则选中的点依次被加入到points列表中。

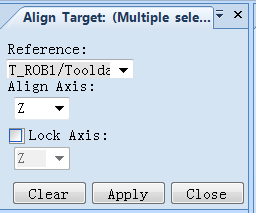
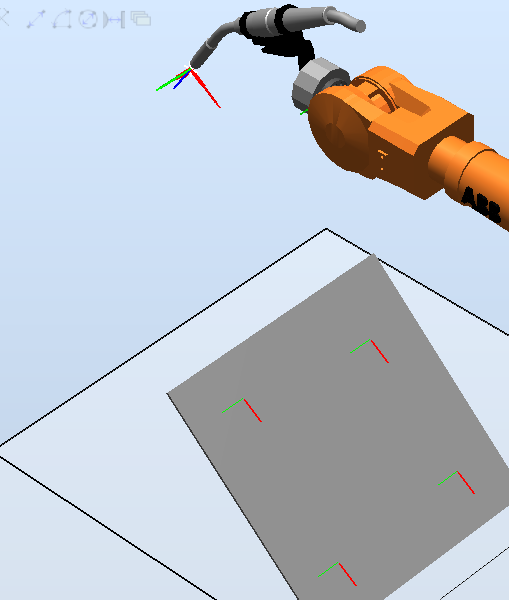
---🡪

点击creat，则生成四个路径点，分别为target\_10到target\_40, 全部选中后右键modify target，Align target orientation,在弹出的对话框中的reference下的文本框点击，将鼠标移动到图形环境中的tcp坐标系点击，则该坐标系被选中。

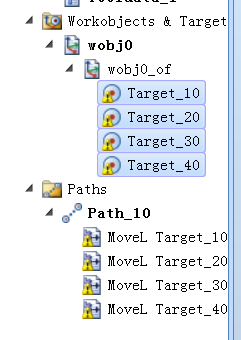
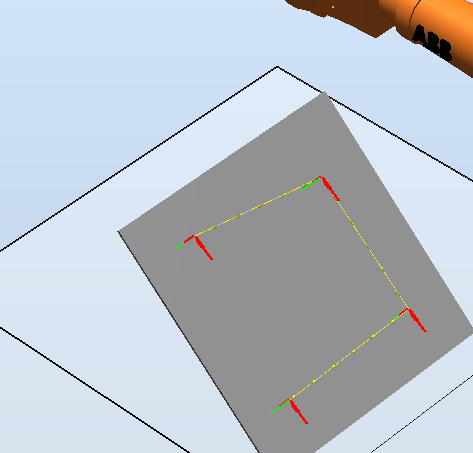
-🡪

点击TCP坐标系

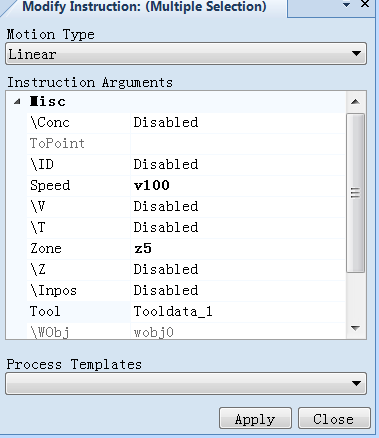
将lockaxis前的对号取消，点击apply，将align axis前的下拉菜单换为y，点击apply，将align axis的下拉菜单改为z，点击apply。则target10～40中的目标被旋转至与tcp方向一致。

-🡪

12. 点击home栏path下empty path，系统在左侧自动创建一个名为path10的路径。将wobj0下的target\_10～target\_40全选，按住左键拖动到path\_10名称上，则路径被创建出来。

---🡪

13 全选movel target10～movel target40，右键modify instruction，输入以下参数

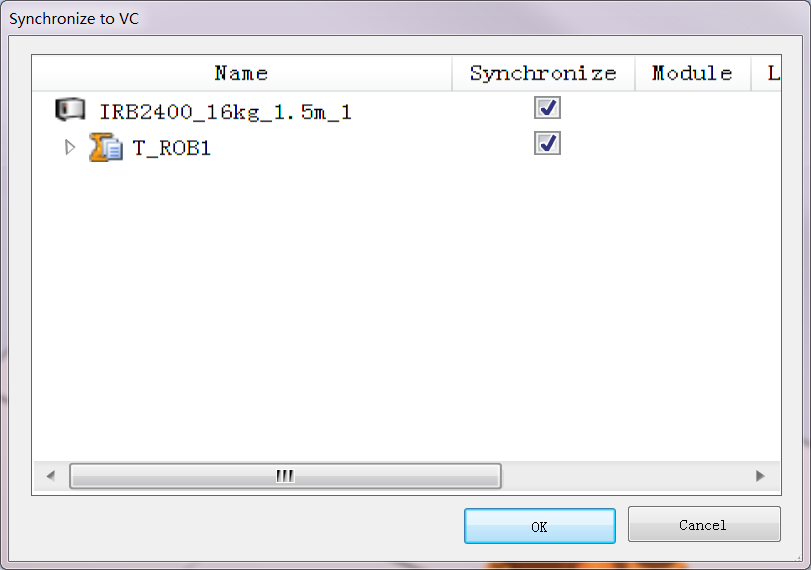


点击apply。

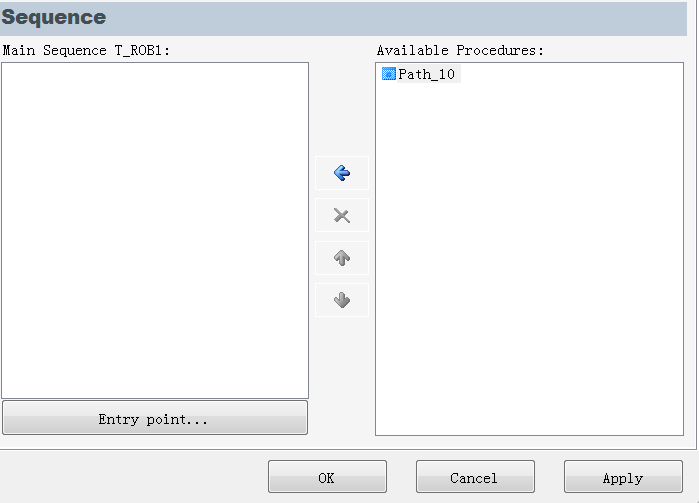
14. 在path\_10上右键，configurations->auto configuration，弹出的对话框中选择cfg1，apply，如下



15. 机器人运行无误后，在path\_10上右键，synchronize to vc，在弹出的对话框上选中两个复选框如下，点ok。



16.在simulation下的simulation setup，将avialble procedures下的path\_10选中，点击旁边向左的箭头将其加入到main sequence t\_rob1中。可多点几次。

🡪

点击ok。

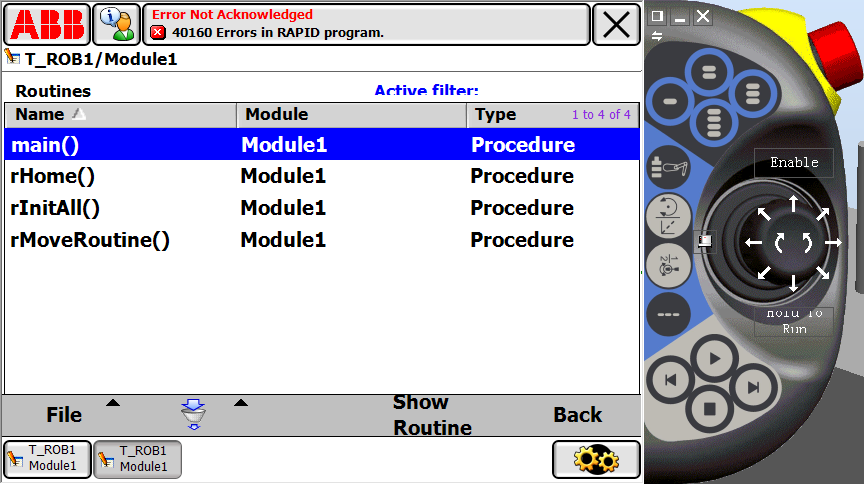
16.点击simulation setup旁边的play按钮查看仿真动画。



实验二：

建立一个可以运行的基本的rapid程序

\*\*\*\*\*\*\*\*\*\*

当此处存在红字时，存在错误，需要点击该红字，并在弹出对话框中点击右下角acknowledge。

\*\*\*\*\*\*\*\*\*\*

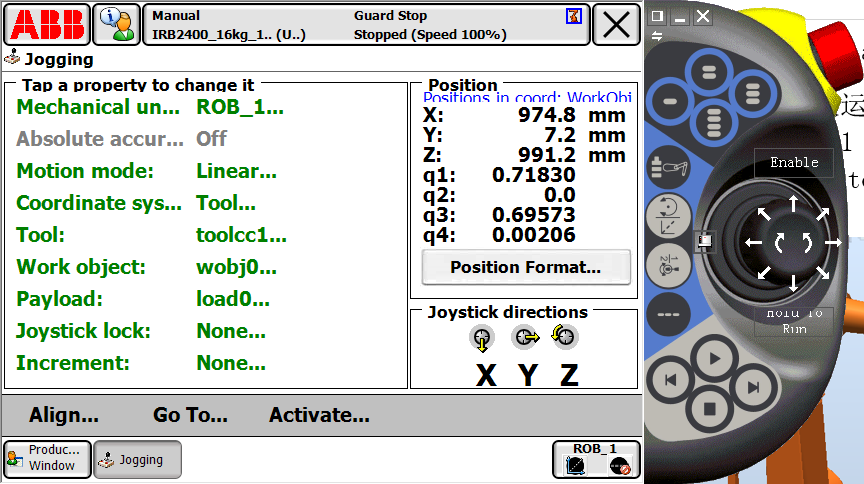
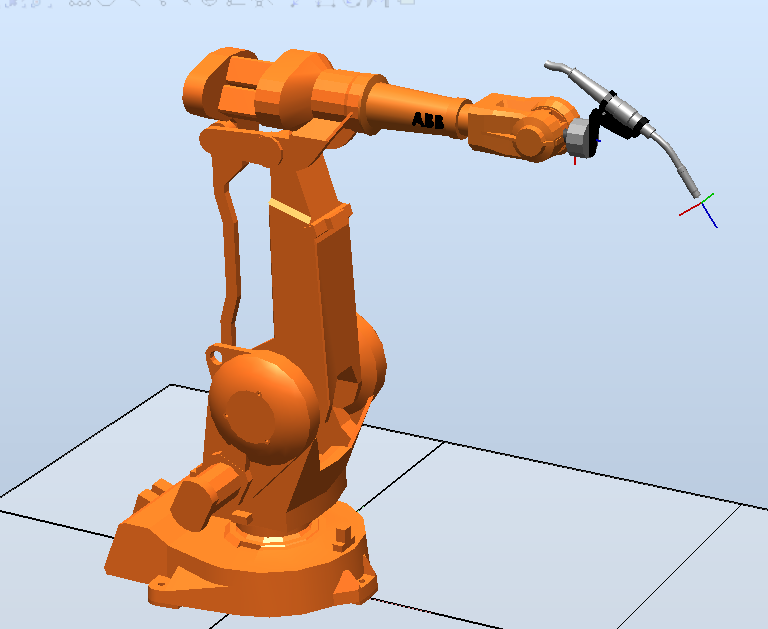
1. 机器人空闲时，在位置点pHome等待。

导入irb2400 16kg 1.5m机器人，等待初始化结束。

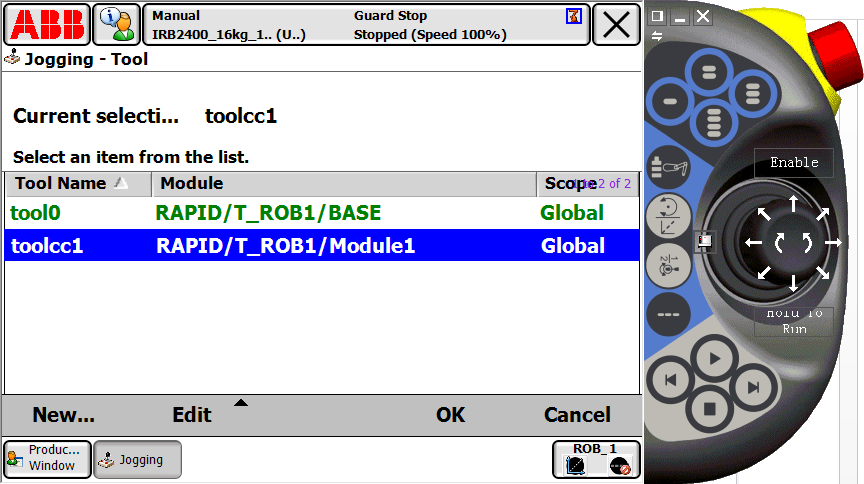
在home->import library中的equipment导入aw\_gun\_psf\_25焊枪。使用place->oneplace功能将该焊枪安装到机器人六轴末端，并与机器人运动关联(attach to irb2400)。

**重要：**使用三点法定义喷枪末端自带工具坐标为TCP：名称为toolcc1（后续图片中tool1大家全部看作toolcc1）如图中红框所示。

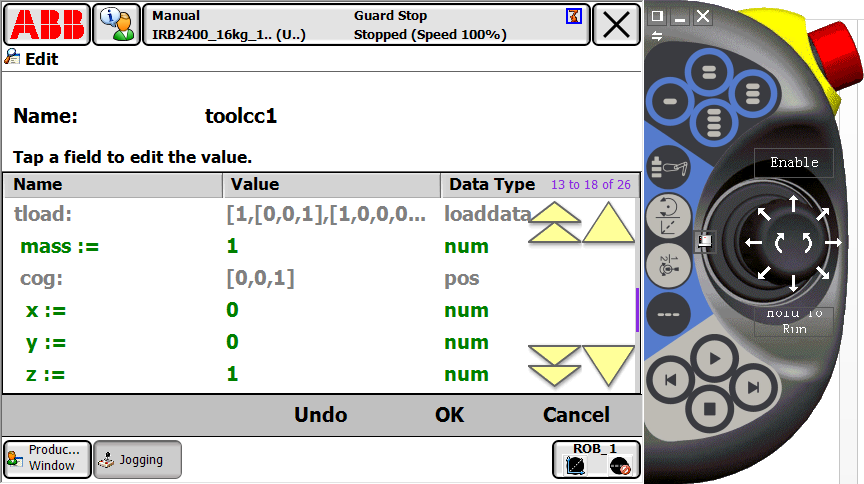
在左上角abb上单击左键，在jogging中，将tool后面tool0切换为toolcc1，如图中红框所示。



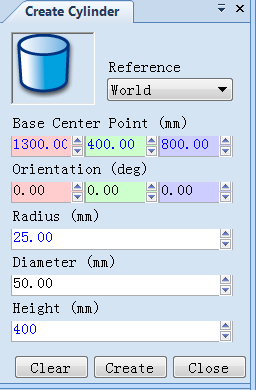
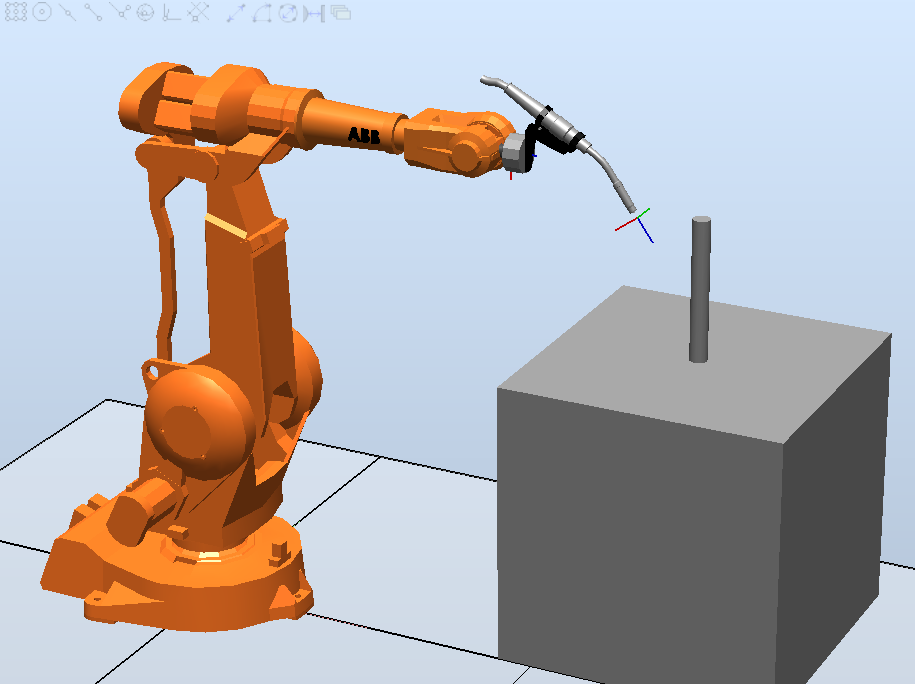
双击toolcc1，进入如下界面，选中toolcc1，点击edit-》change value，



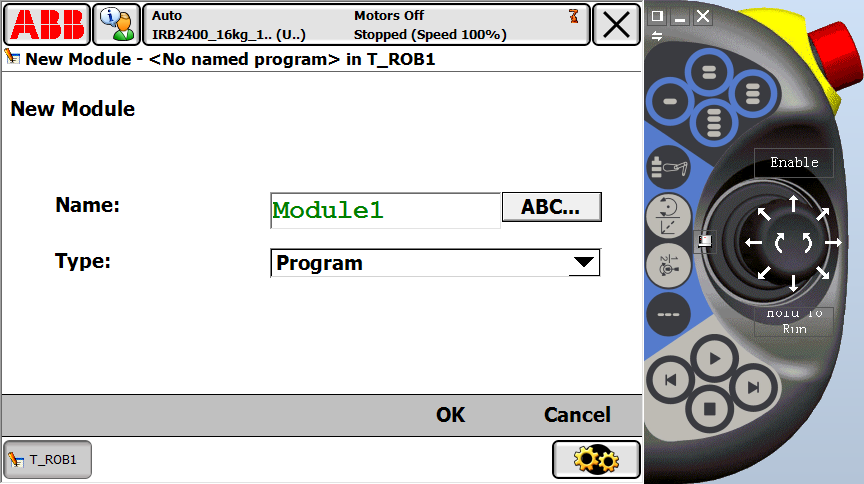
将mass和z中的值分别赋值为1，如下图所示。Ok，ok。



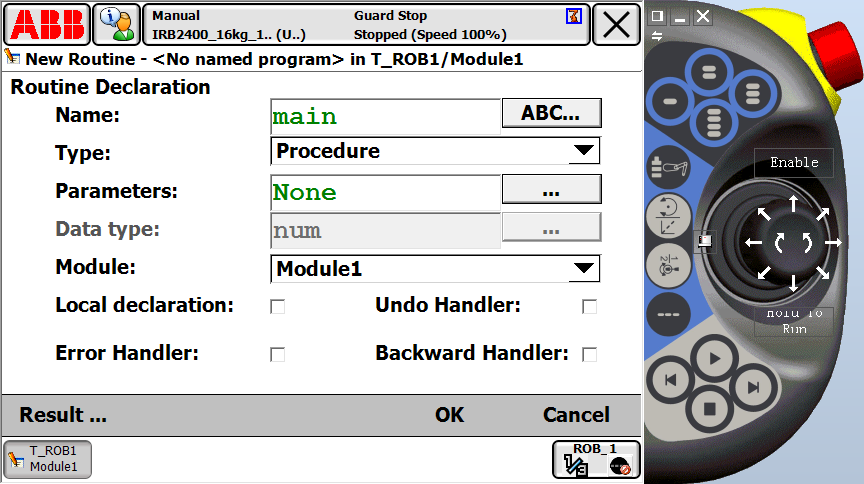
1. 新建立方体modeling->solid->box,设置参数如下：creat。新建圆柱体solid->cylinder，参数如如下，create。

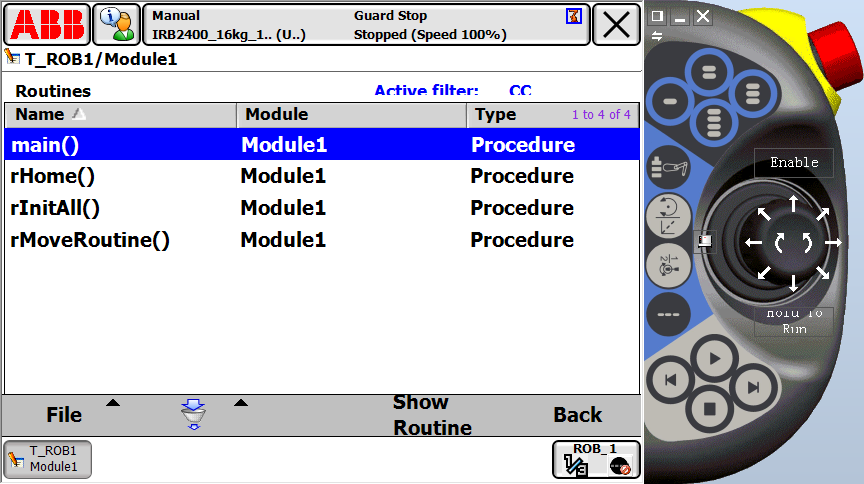
1. 在controller下的flexpendant中打开virtual flexpendant，选择program editor。在警示中点击取消。
2. File->new module->新模块名称定位为Module1，ok；



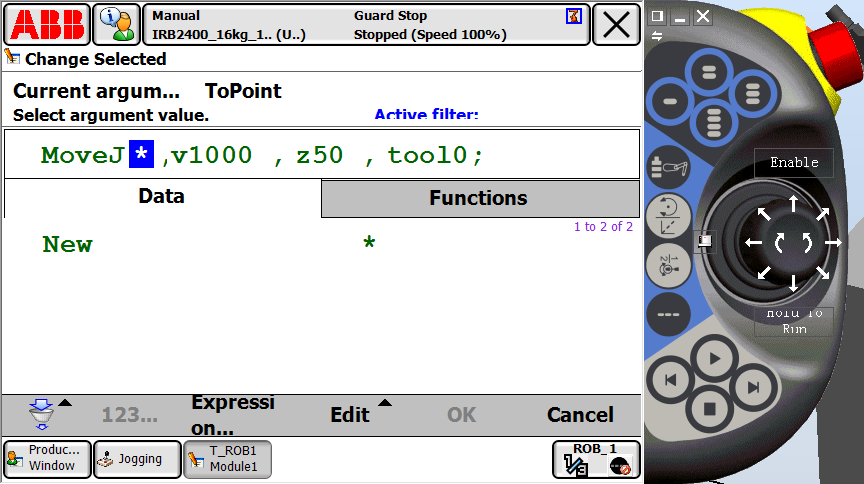
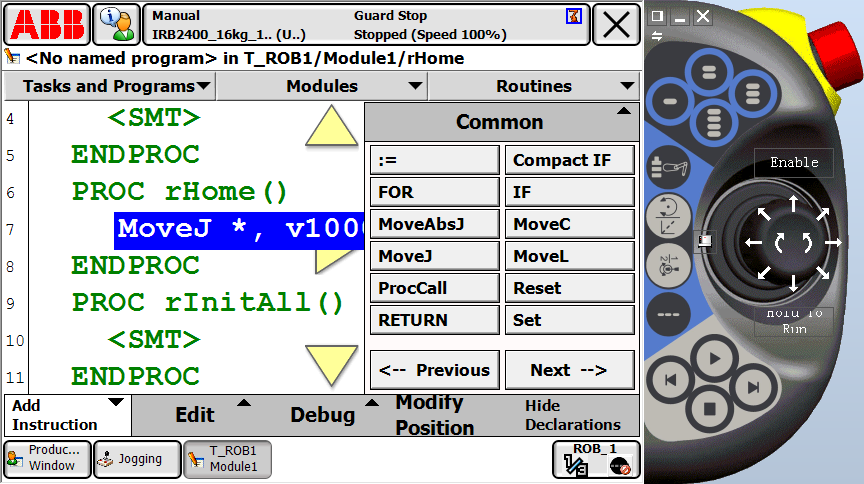
1. 选择Module1，单击show module->单击routines->File->new routine建立主程序main，ok



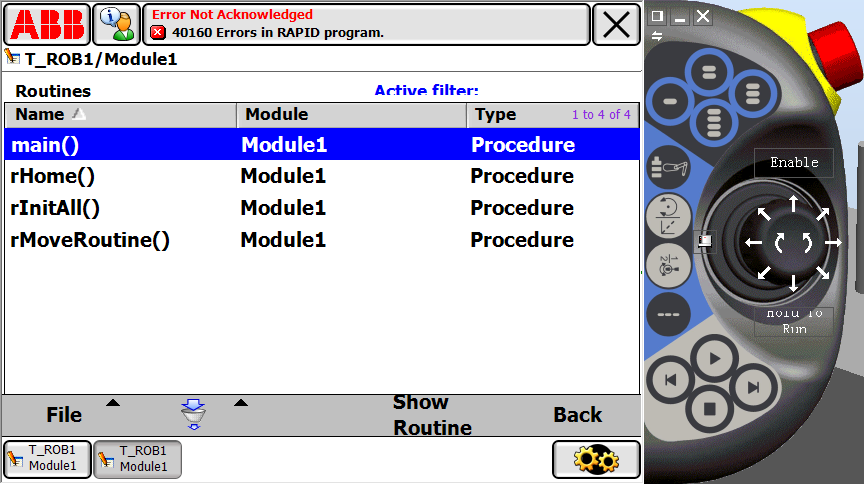
1. 根据步骤5建立另外三个routine(例行程序)，其中rHome()用于机器人回等待位，rInitAll()初始化，rMoveRoutine()存放直线运动路径。



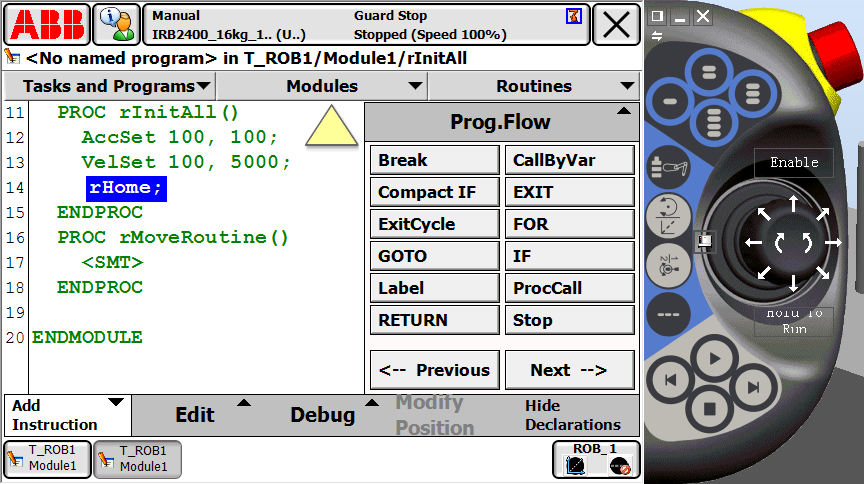
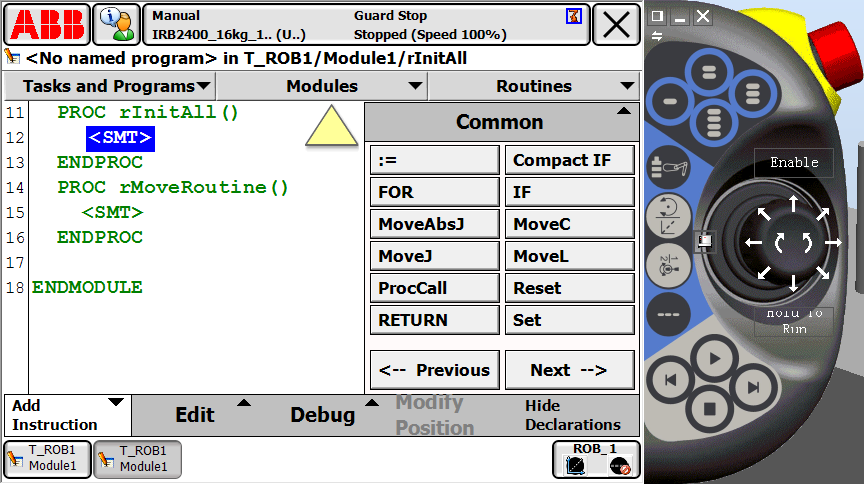
1. 回到程序编辑器，单击add instruction，打开指令列表。选中<smt>为插入指令的位置，在指令列表中选择MoveJ，双击\*,进入指令参数修改画面。



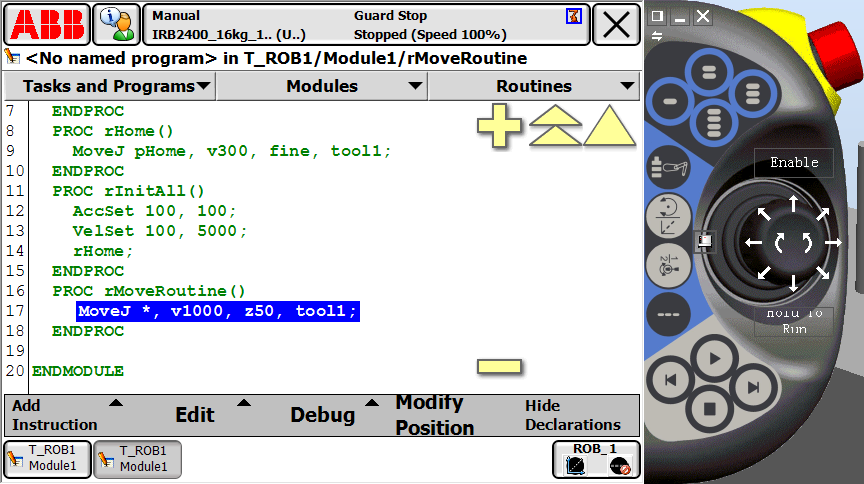
1. 双击new，新建名为pHome的目标点。
2. 选中pHome目标点，单击modify position，记录机器人当前位置数据。
3. 点击Routines，选择rInitAll()程序，选择show Routine。



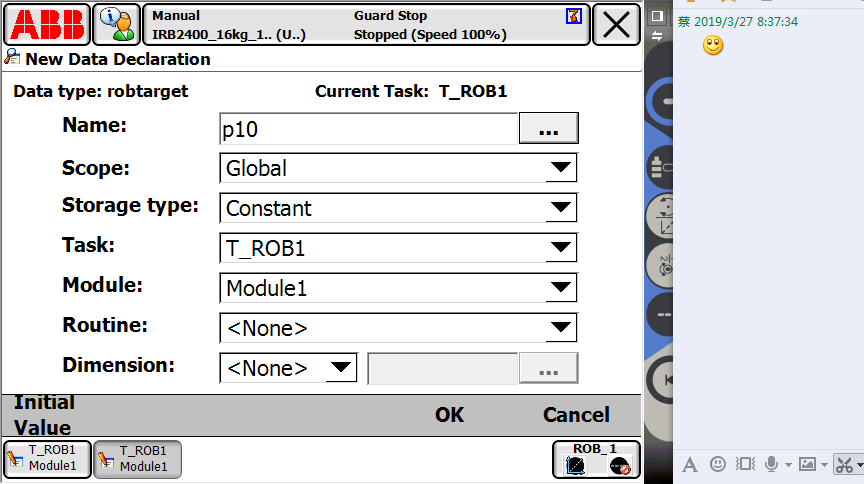
1. 在此例行程序中，加入程序运行前需要初始化的内容，如速度限定、夹具复位等。确保机器人在手动模式下，点击add instruction，通过next切换页面选择指令添加。使用prog.flow中的proccall添加一个回等待位的例行程序rHome.

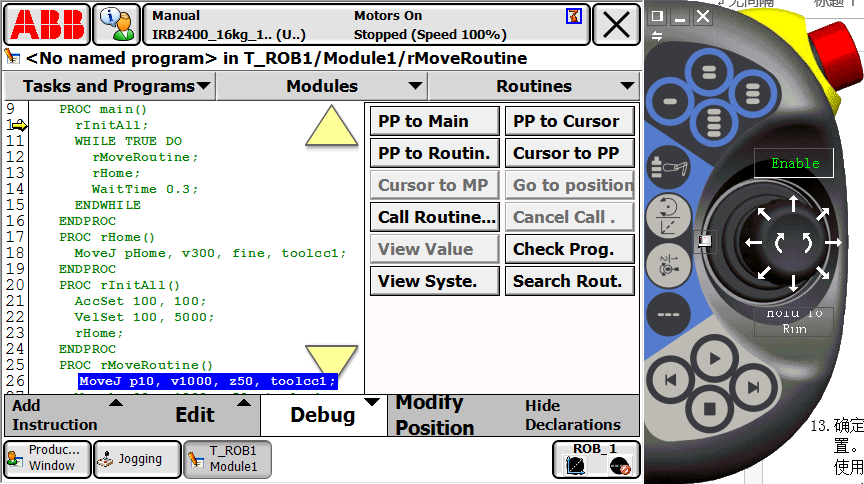


1. 单击routines标签，选择rMoveRoutine例行程序（routines），单击show routine。使用add instruction为该例行程序添加指令。

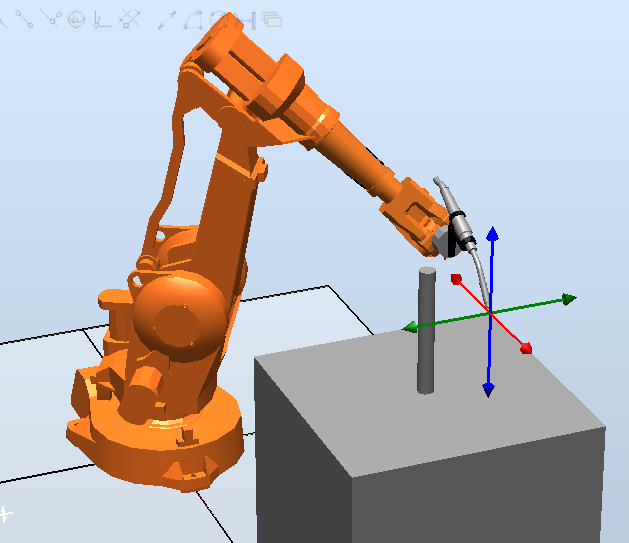


13.双击MoveJ后面的\*，双击new创建路径点p10，将示教器最小化，在图形环境中使用平移工具将机器人的tcp移动到立方体角点1位置。如第二图所示。（如无法成功移动，请点击摇杆上方的ENABLE，将其变成绿色，启动电机motors on）

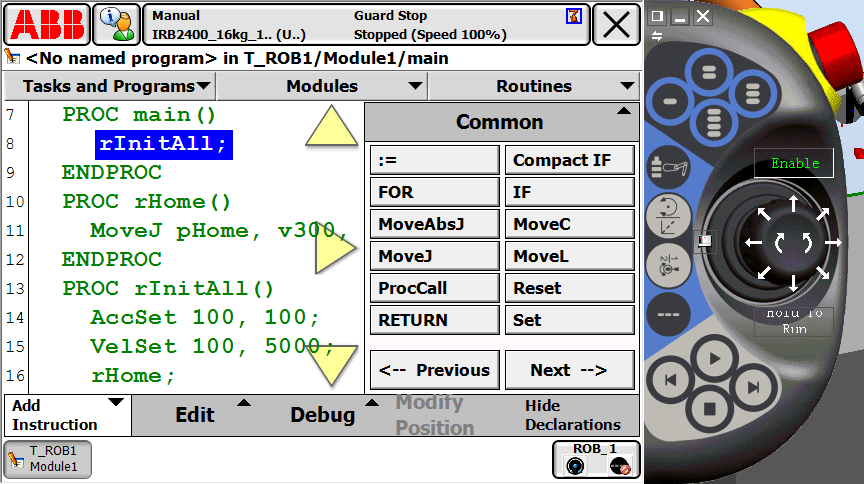


确定位置后重新切换到示教器的program editor界面，点击下方的modify position保存该位置。

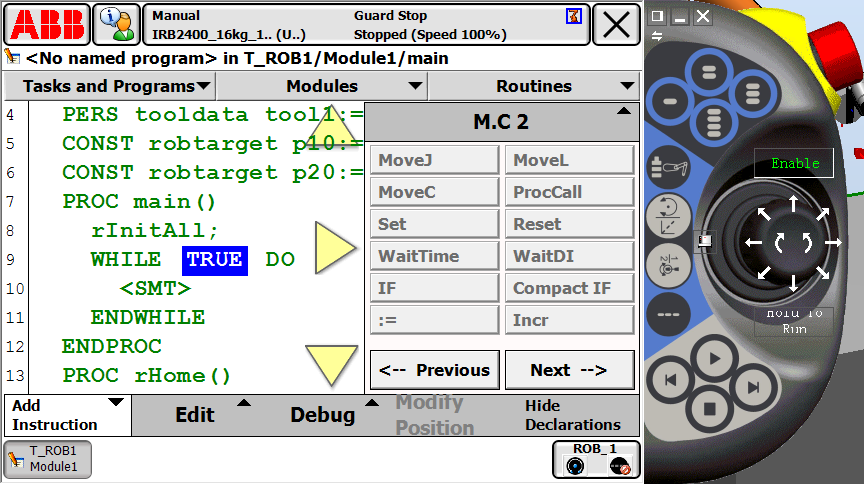
使用相同的方法添加MoveL指令，将tcp移动到立方体另一顶点存储p20位置（modify position）。



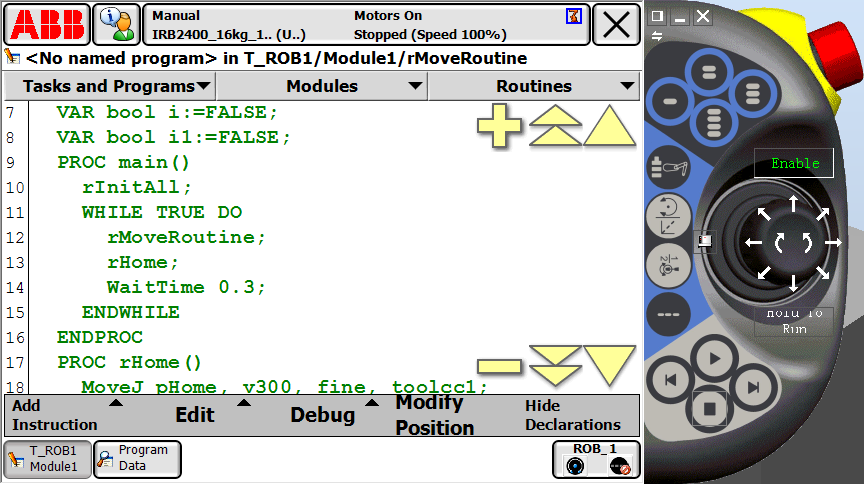
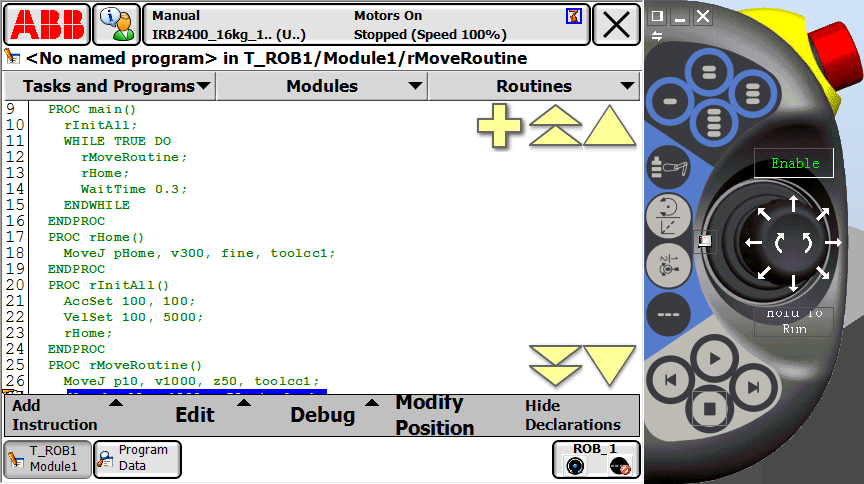
1. 选中main主程序，进行程序执行主体架构的设定。在开始位置调用初始化例行程序。调用rInitAll例行程序。

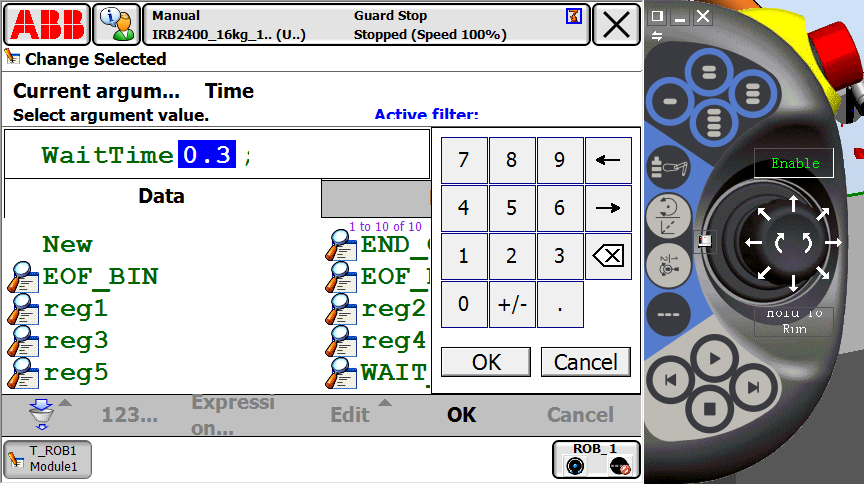


1. 添加while指令，并将条件设定为true

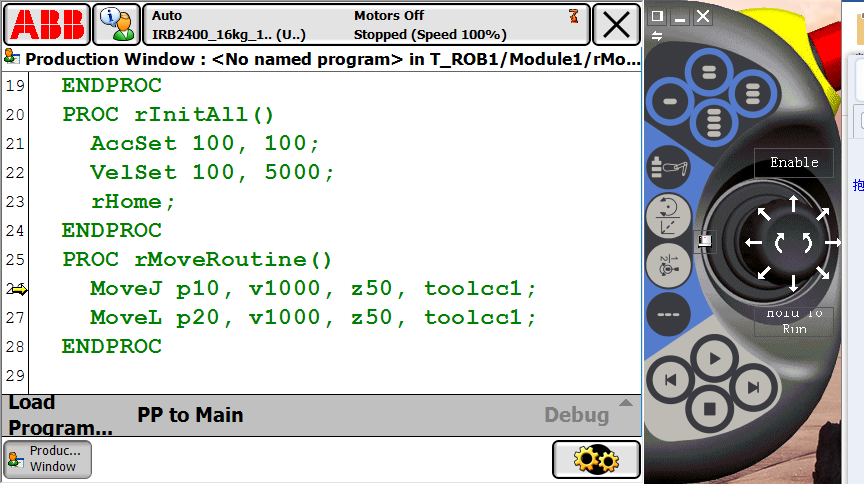


16.在while中添加其他代码如下：

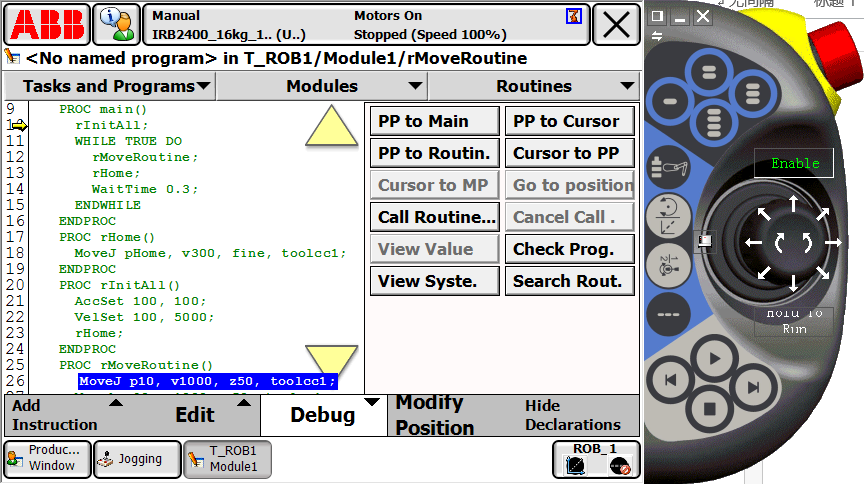




1. 打开调试菜单，单击检查程序，对程序的语法debug->check prog.进行检查。未出现任何错误后，单击确定完成，如有错系统会提示出错的具体位置与建议操作。
2. 确保在控制器在手动模式下。查看示教器为motors on模式，如果下图中的motorsoff或者guard stop模式，则需要将enable点为白色，再点为绿色，等待变为motors on。



点击下方debug，在弹出菜单中点击pp to main，让程序指针指向主程序main。最终编辑后的界面代码如下：



点击运行键查看仿真

