

•
难度水平：高级

创建一个安全原点区域例行程序 - 19477

编辑：王一粟

本例子对以下软件版本有效：

CB2 Software version: 1.8.16941

CB3 Software version: 3.2.18744

注意：比这个版本老的或新的软件有一些不同

适用机型：UR3、UR5 和 UR10

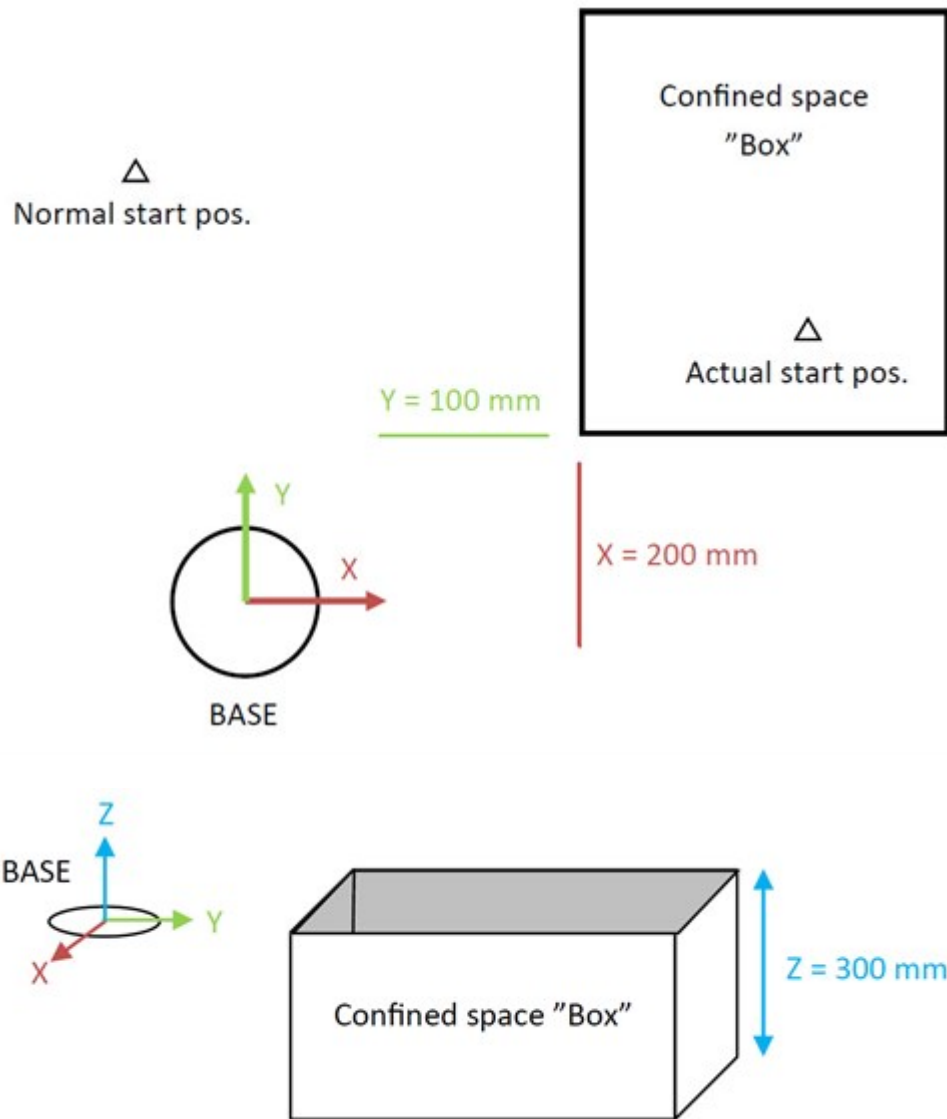
编辑时间：2016 年 2 月 5 日

这个文档描述如何在不用“自动”功能时，自动启动机器人。执行一个安全原点程序是为了当机器人在现在所在位置重新开始运行程序时，避免碰到其他物体，直到回到正常开始程序。

安全注意：

当程序初始路点是可变路点或相对路点，当按程序开始按钮（**Play**）机器人将开始运动，而不需要再按“自动”按钮，这样需要考虑机器人运动的风险。安全原点程序就像编程者设计的复杂程序，而不是机器人的安全设置。

当机器人运行程序时，考虑机器人进入它的工作空间内的一个盒子。如果机器人由于某些原因在盒子内停止（突然断电，急停或其他原因），然后机器人需要重新启动，可能机器人将运动不同的轨迹回到正常启动位置。



根据机器人所在位置，当开始程序时，我们执行一个安全原点区域程序确保机器人运行不同的轨迹到正常开始点，这样操作者将不需要用“自动”或“手动”功能移动机器人到开始点。

1. 当启动机器人程序时，我们从检查机器人当前位置着手，并且让机器人运动到一个可变路点，这个可变路点恰巧是当前位置。

```
start_pose = get_actual_tcp_pose()
MoveL
start_pose
```

我们用脚本代码 "`get_actual_tcp_pose()`" 保存 TCP 当前位置，并赋值给位姿变量 "`start_pose`".

2. 我们保存机器人当前位置之后，按坐标系分解不同的方向：X，Y 和 Z 方向。（另一篇文章解释如何读单独的位置：Read a single coordinate or axis rotation – 15379）

```

start_x = start_pose[0]
start_y = start_pose[1]
start_z = start_pose[2]

```

3. 在这个定义的空间内，按照上面的说明机器人能进入这个空间。这样如果实际的 TCP 位姿在这个盒子内，我们将首先移动机器人到盒子边界外；如果 TCP 的 X 方向超过 200mm（0.2m），Y 方向超过 100mm（0.1m），我们在盒子内并且移动到外边。

```
IF (start_x >= 0.2) and (start_y >= 0.1)
```

4. 如果两个值都是“真”，那么我们在盒子内，并且需要向上方运动；这样我们需要创建一个所有位置相同（除 Z-轴参数）的安全点 "Safe_Pose"，我们确保在盒子上方 350mm（0.35m）位置，并且能直接运动到正常起始位置 "Normal_start"。

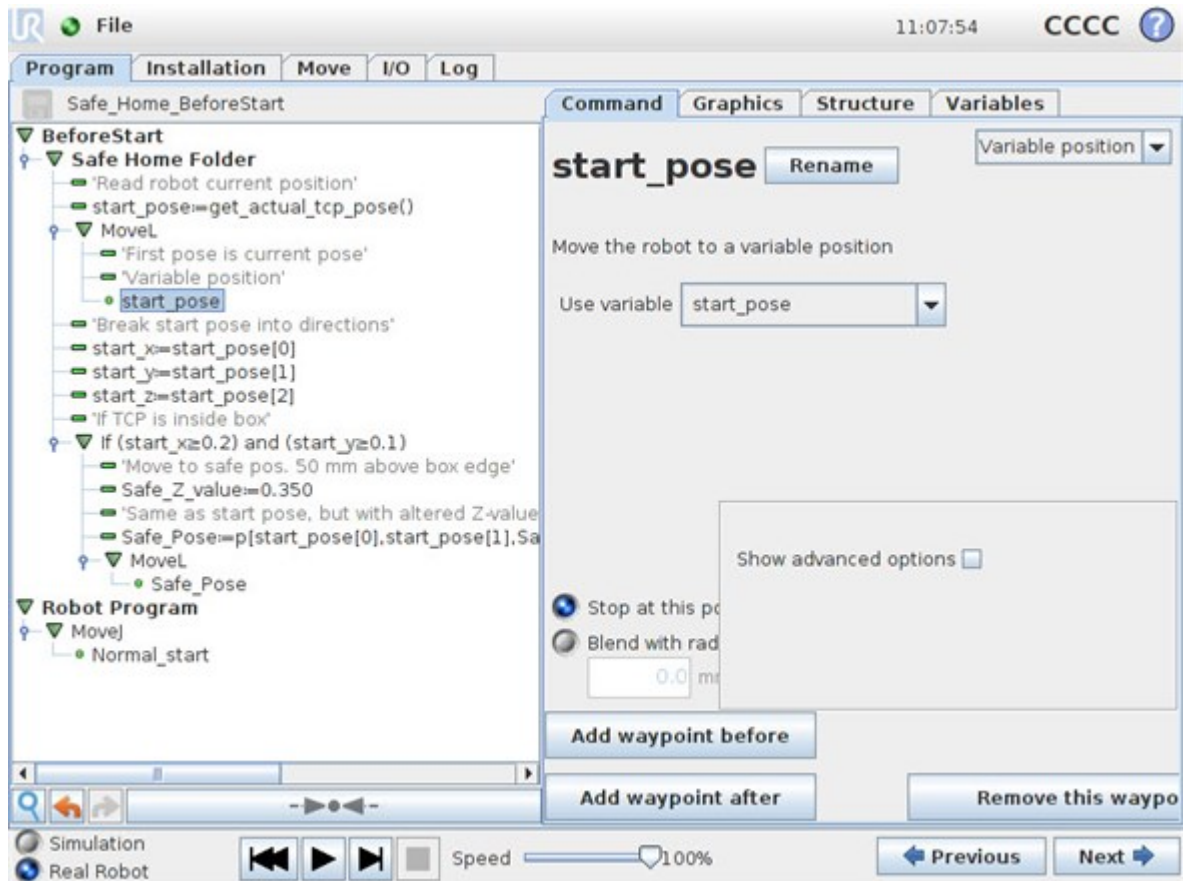
```

Safe_Z_value = 0.350
Safe_Pose =
p[start_pose[0],start_pose[1],Safe_Z_value,start_pose[3],start_pose[4],start_pose[5]]
MoveL
  Safe_Pose

```

这个位姿变量包含 6 个参数 $pose_variable = p[x,y,z,rx,ry,rz]$

5. 如果我们在盒子内，我们首先移动到正上方，然后直接移动到正常开始点位置 Normal_start；如果我们在盒子外，我们能直接移动到正常开始点位置 Normal_start position。这样，我们的安全原点区域例行程序完成，并且我们可以在机器人程序中执行它。



见附件“例行程序”

以上只是对安全原点例行程序做了一个简要的说明，如果用更复杂的轨迹和 if 语句，可以做更大的靠近机构位置的安全原点例行程序。

附加文件

- [Safe_Home_BeforeStart.urp](#)