# 表达式计算说明

Codeit变量定义以 Var开头。

## 基本变量

整型 int，示例：**Var** –-name=counter --type=int --value=4。

浮点型 double，示例：**Var** –-name=time –-type=double --value=1.5。

布尔型bool，示例：**Var** –-name=ifClose –-type=bool –-value=1。

字符串string，示例：**Var** –-name=na --type=string --value=”kaanh”；

数组array，示例：VAR int[] table[16]，table[6]=8；

VAR int[][] table[1][16]，table[1][6]=8；

VAR int[][] temp[2][3]={{1,2,3},{4,5,6}}。

（int 可换为double、bool、string）

## 其他结构体

1.1.1 load

—————————————————————————————

说明：该结构体包含一个刚体在某一个坐标系下的十个描述质量分布特性的double型参数：mass质量，cogx质心在x方向的偏移量，cogy质心在y方向的偏移量，cogz质心在z方向的偏移量，i表示惯性张量矩阵参数。

示例：

**Var** –-name=piece–-type=load --value={mass,cogx,cogy,cogz,ixx,ixy,ixz,iyy,iyz,izz}。

对于机器人来说，该参数可用于描述机器人末端负载参数特性，坐标系一般默认为末端法兰盘坐标系

1.1.2 jointtarget

—————————————————————————————

说明：该结构体共有十个double型参数（J0~J9），存储机器人目标关节角度（单位是度）和外部轴位置（单位是毫米或度）。Codeit会从该十个数依次匹配目标关节角度与外部轴位置。目标关节角度与外部轴总数不得超过十个。

示例：

**Var** –-name=home–-type=jointtarget --value={J0,J1,J2,J3,J4,J5,J6,J7,J8,J9}。

1.1.3 robtarget

说明：储笛卡尔空间的目标位置及姿态（double），包含x, y, z, q1, q2, q3, q4，eax\_a、eax\_b...eax\_f，适用于MoveJ, MoveL，MoveC指令的，位置单位为毫米。

示例：

**Var** –-name=p1 –-type=robtarget --value={x,y,z,q1,q2,q3,q4,eax\_a,eax\_b...eax\_f }。

***附加轴的位置。  
各单个轴（eax\_a、eax\_b...eax\_f）的位置 is defin如下：  
• 对于旋转轴，其位置定义为从校准位置起旋转的度数。  
• 对于线性轴，其位置定义为与校准位置的距离（以mm计）。  
附加轴eax\_a ... 为逻辑轴。在系统参数中定义逻辑轴编号与物理轴编号的相互关  
系。  
定义未连接轴的值9E9。如果位置数据中定义的轴不同于程序执行时实际连接的轴，  
则以下内容适用：  
• 如果未在位置数据（值9E9）中定义位置，则当连接且未激活该轴时，将忽略该  
值。但是如果激活该轴，则将会产生错误。  
• 如果在位置数据中定义该位置，则忽略该值，即便未连接该轴。  
如果未激活轴，则不会进行任何运动，且不会产生有关轴（含正确位置数据）的错误。  
如果一个附加轴以独立模式运行，且应由机械臂及其附加轴进行新的运动，则采用独  
立模式的附加轴的位置数据不得为9E9。数据必须为系统未使用的任意值。***

1.1.4 speed

**—————————————————————————————**

说明：包含5个double型成员变量，per关节速度百分比，适用于MoveAbsJ和MoveJ。tcp线速度，eri空间旋转速度，exj\_l外部轴线速度，exj\_r外部轴角速度。示例：speed v1= {40 300 100 200 1000}。

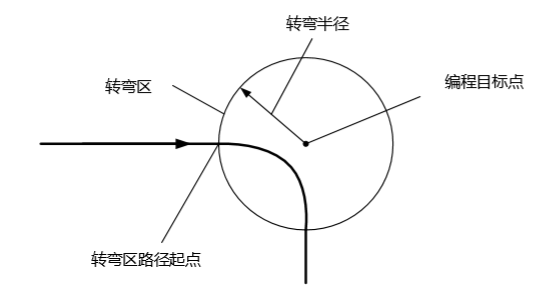
示例：**Var** –-name=v1 –-type=speed –-value={per,tcp,ori,exj\_r,exj\_l}。

1.1.5 zone

说明：变量用于定义某一个运动如何结束或者说定义两条运动轨迹之间转弯区的大小。该结构体包含两个double型变量，distance为笛卡尔空间的转弯区大小，适用于MoveLL指令，单位是毫米；percent为转弯区所占运动时间的百分比，适用于MoveJ、MoveL、MoveC等运动指令直接的融合

转弯区的距离大小不能超过前后两条待融合路径长度的一半，如果超过，系统会自动将转弯区长度缩小到前后两条路径中长度较小的一半；转弯区的时间不能超过前后两条待融合运动时间的一半，如果超过，系统会自动将转弯区时间缩小到前后两条路径中时间较小的一半。使用转弯区可以避免机器人频繁启停，显著减少节拍时间。

示例：**Var** –-name=z100 –-type=zone –-value={dis,per}。



1.1.6 pose

————————————————————————————

说明：该结构体存储笛卡尔空间的位置及姿态（double），包含x,y,z,q1,q2,q3,q4，其中位置单位为毫米

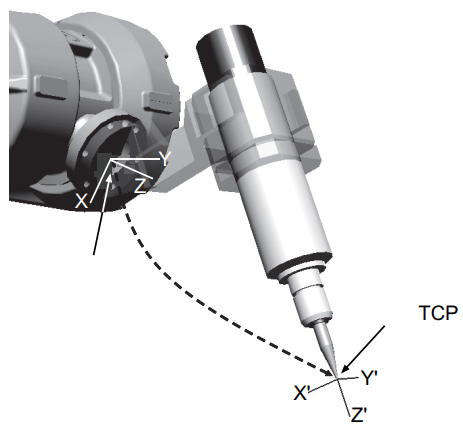
示例：**Var** –-name=pose1 –-type=pose --value= {x,y,z,q1,q2,q3,q4}。

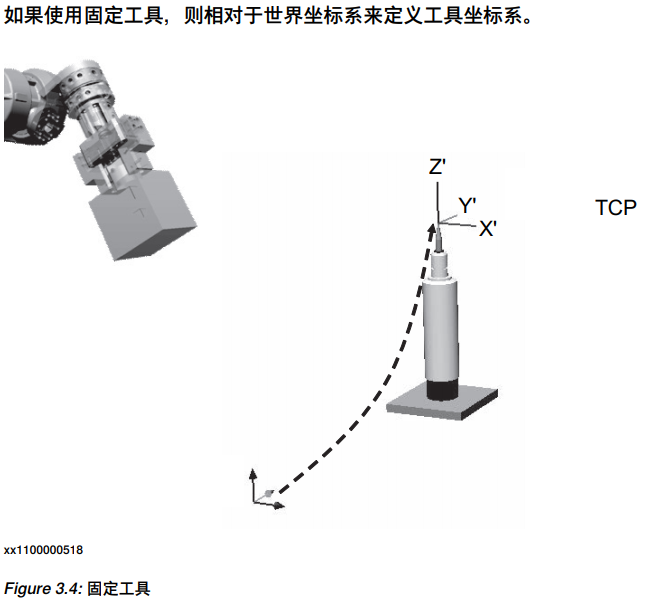
1.3.7 tool

说明：记录工具参数，包含机器人所用工具的TCP、姿态，具体包含以下三个子结构体： pose类型，包含TCP相对于机器人末端法兰盘坐标系的x,y,z三个方向的偏移量，单位为毫米，以及工具坐标系相对于法兰坐标系的姿态偏移量，用四元数表示。pos结构体如下：pose pose1={x,y,z,q1, q2, q3, q4}。

示例：**Var** –-name=tool –-type=tool

--value={true,x,y,z,q1,q2,q3,q4}。





1.3.8 wobj

**—————————————————————————————**

说明：所有运动指令中使用的位置都是在工件坐标系下定义的（如果没有指定工件坐标系，则默认 在世界坐标系下定义，世界坐标系可以被看做是 wobj0）。pose类型。

示例：

**Var** –-name=wobj2 –-type=wobj --value ={x, y, z, q1, q2, q3, q4}.

