

目录

内容

1 声明	1
2 概述	2
3 调试	3
4 基础知识	10
5 建立模型示例	16
6 程序示例	32
7 参考文献	38
8 作者/联系人	38
9 版本信息	38

1 声明

本手册及样例包均为免费提供，仅供参考。所述版本、应用及案例可能与用户实际应用不符，请用户在使用前认真阅读相关使用说明，根据自身的应用环境及机床特点进行调整，并进行严格的测试，以规避可能存在的风险。对于在使用中发生的人员、财产等损失，由用户自行承担。

以上声明内容的最终解释权归西门子（中国）有限公司所有，后续内容更新恕不做另行通知。

MTS APC

2 概述

2.1 适用配置

系统软件	SW24	SW26	SW28（限单通道）
车	☐	☐	☐
铣	☐	☐	☐
磨	☐	☐	☐
选项	6FC5800-0AS03-0YB0		
软件版本	>=SW 4.8 SP3		
通道要求	双通道系统时，运动链只能定义单个通道的轴；		

O：选项 AS03

2.2 说明

防碰撞功能用于在 JOG/MDA/AUTO 下防止机械部件之间，机床部件和加工工件之间的碰撞，以免损坏部件和工件。是基于机床模型、运动链和碰撞组的定义实现的。

2.3 选项

碰撞监测 ECO	☒	☒
6FC5800-0AS03-0YB0		

选项激活后，“制造商”口令下，可以在【调试】--【NC】界面看到【机床模型】，这是防碰撞界面设置的入口。
未激活选项时，该按键（入口）不显示。

	PROFIBUS		机床模型	NC内存	刀具管理
--	----------	--	------	------	------

26280 轴 M21 碰撞危险 PA_TABLE PA_TOOL

树形图

有效数据

ROOT: OFFSET_Z

PA_BASE

BASE1

OFFSET_Y

AX_Y

OFFSET_X

AX_X

OFFSET_TAB

PA_TABLE

ELE_TABLE

OFFSET_Z

AX_Z

PA_Z

ELE_Z1

ELE_Z2

OFFSET_SP

AX_SP

OFFSET_TOOL

PA_TOOL

TOOL_88881

任意元素

碰撞组

PA_TABLE, PA_Z

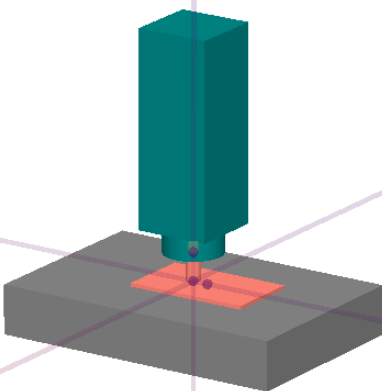
PA_TABLE, PA_TOOL

PA_BASE, PA_Z

PA_BASE, PA_TOOL

3D文件

图形



树形图

图形

新建单元

激活数据

复制

粘贴

剪切

删除元素

NC内存

PROFIBUS

机床模型

坐标转换

刀具管理

3 调试

3.1 基础参数设置

设置、检查如下基础参数，确保运动链元素数量等可用。

参数号	名称	描述
10619	\$MN_COLLISION_TOLERANCE	碰撞监测的公差
10622	\$MN_COLLISION_TOLERANCE	碰撞监测缺省的安全距离
16800	\$MN_ROOT_KIN_ELEM_NAME	运动链根元素的名称
18866	\$MN_MM_NUM_KIN_TRAFOS	可通过运动链定义的转换的最大数量
18880	\$MN_MM_MAXNUM_KIN_CHAIN_ELEM	运动链元素的最大数量
18882	\$MN_MM_MAXNUM_KIN_SWITCHES	运动链中开关的最大数量
18890	\$MN_MM_MAXNUM_3D_PROT_AREAS	3D 保护区的最大数量
18892	\$MN_MM_MAXNUM_3D_PROT_AREA_ELEM	保护区单元的最大数量
18893	\$MN_MM_MAXNUM_3D_T_PROT_ELEM	刀具保护区单元的最大数量
18894	\$MN_MM_MAXNUM_3D_FACETS_INTERN	用于内部创建的保护区的最大平面数量
18895	\$MN_MM_MAXNUM_3D_FACETS	用于所有保护区的最大平面数
18896	\$MN_MM_MAXNUM_3D_COLLISION	碰撞监测所需的临时最大存储容量
18897	\$MN_MM_MAXNUM_3D_INTERFACE_IN	决定了每个 NC 程序段所需的存储容量
18898	\$MN_MM_MAXNUM_3D_COLL_PAIRS	3D 碰撞组的最大数量
18899	\$MN_PROT_AREA_TOOL_MASK	设定在碰撞监测激活时以何种方式自动生成刀具保护区
51160	\$MNS_ACCESS_WRITE_CA_MACH_JOG	JOG 方式激活/关闭机床碰撞监测”的权限等级
51161	\$MNS_ACCESS_WRITE_CA_MACH_AUTO	AUTO 方式激活/关闭机床碰撞监测”的权限等级
51162	\$MNS_ACCESS_WRITE_CA_TOOL	激活/关闭刀具的机床碰撞监测”的权限等级

示例参数：

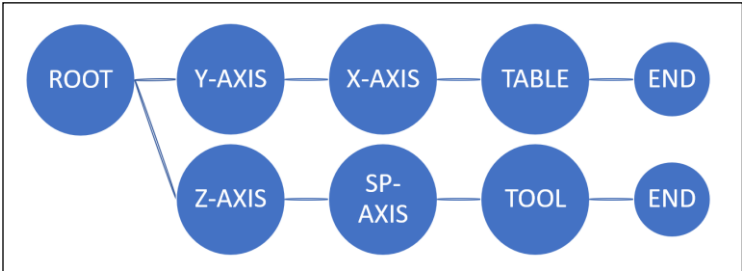
```
$MN_COLLISION_TOLERANCE = 1
$MN_COLLISION_PREP_CALC_TIME = 0.5
$MN_COLLISION_SAFETY_DIST = 5
$MN_MM_MAXNUM_3D_PROT_AREAS = 17
$MN_MM_MAXNUM_3D_PROT_AREA_ELEM = 68
$MN_MM_MAXNUM_3D_T_PROT_ELEM = 1
$MN_MM_MAXNUM_3D_FACETS_INTERN = 1000
$MN_MM_MAXNUM_3D_INTERFACE_IN = 16
$MN_MM_MAXNUM_3D_COLL_PAIR = 10
$MN_PROT_AREA_TOOL_MASK = 1
$MNS_ACCESS_WRITE_CA_MACH_JOG = 6
$MNS_ACCESS_WRITE_CA_MACH_AUTO = 3
```

```
$MNS_ACCESS_WRITE_CA_TOOL = 1
NEWCONF
M30
```

3.2 运动链

1.1.1 定义

把两个或两个以上的构件通过直接接触并能产生相对运动的活动联接而构成的相对可动的系统称为运动链。



运动链分为两条主线：

- 1. 基准点到工作台：
基准点—Y 轴—X 轴—工作台；
- 2. 基准点到刀具：
基准点—Z 轴—主轴—刀具；

3.2.1 变量说明

除了通过 Operate 界面建立防碰撞模型外，也可使用如下变量编写 NC 程序，使用执行程序的方式建立防碰撞模型。

变量名称	描述	示例
\$NK_NAME[n]	当前元素名	\$NK_NAME[n] = "ROOT"
\$NK_NEXT[n]	下个元素名	\$NK_NEXT[n] = "AXIS-SP1"
\$NK_PARALLEL[n]	平行分支的元素名	\$NK_PARALLEL[n] = "AXIS-Z"
\$NK_TYPE[n]	元素类型	\$NK_TYPE[n] = "AXIS_LIN"
\$NK_OFF_DIR[n,i]	偏移的方向（矢量）	\$NK_OFF_DIR[n,0] = 1
\$NK_AXIS[n]	机床轴名称（MD10000）	\$NK_AXIS[n] = "X1"
\$NK_A_OFF[n]	轴偏移	\$NK_A_OFF[n] = 100
\$NK_SWITCH[n]	开关	
\$NK_SWITCH_INDEX[n]	开关变量索引号	\$NK_SWITCH_INDEX[1] = 2
\$NK_SWITCH_POS[n]	开关关闭的设定值	\$NK_SWITCH_POS[2] = 2

3.2.2 元素名 \$NK_NAME[n]，\$NK_NEXT[n]，\$NK_PARALLEL[n]

- \$NK_NAME[n] 当前元素的元素名
- \$NK_NEXT[n] 下一个元素的元素名

- \$NK_PARALLEL[n] 并列元素的元素名

示例
如果使用 Operate 建立防碰撞模型，则模型元素下标是按照建立的先后顺序定义的，如元素的排列如下



元素名称的下标如下

变量	格式	值
\$NK_NAME[0]	A	ROOT
\$NK_NAME[1]	A	Y1
\$NK_NAME[2]	A	SWITCH_1_1
\$NK_NAME[3]	A	END_Y1
\$NK_NAME[4]	A	SWITCH_2_2
\$NK_NAME[5]	A	SWITCH_3_3

3.2.3 元素类型 \$NK_TYPE[n]

- \$NK_TYPE[n] = "xxxx", xxxx 有如下类型**
- OFFSET 偏移
 - ROT_CONST 转角度数
 - AXIS_LIN 机床直线轴
 - AXIS_ROT 机床旋转轴（包含主轴）
 - SWITCH 开关

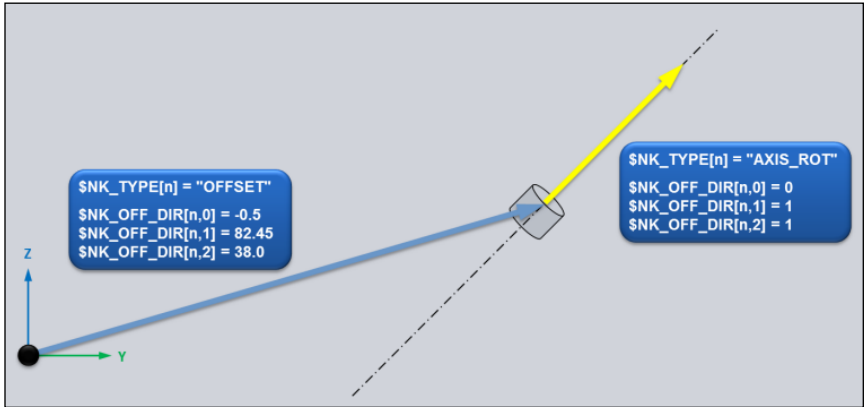
根元素	普通元素
新建动态转换单元 新建动态转换单元 旋转 线性轴 旋转轴 Offset 开关	新建动态转换单元 新建动态转换单元 旋转 线性轴 旋转轴 Offset 开关 平行旋转 平行线性轴 平行旋转轴 平行偏移 平行开关

3.2.4 矢量元素 \$NK_OFF_DIR[n,i]

3.2.4.1 \$NK_OFF_DIR[n,i]（方向）

i=0,1,2 分别表示几何轴 X,Y,Z 方向上。
如 X 轴正方向上的矢量表示为：
\$NK_OFF_DIR[n,0] = 1 , \$NK_OFF_DIR[n,1] = 0 , \$NK_OFF_DIR[n,2] = 0

矢量始终基于 MCS 坐标系，而非相对坐标系。

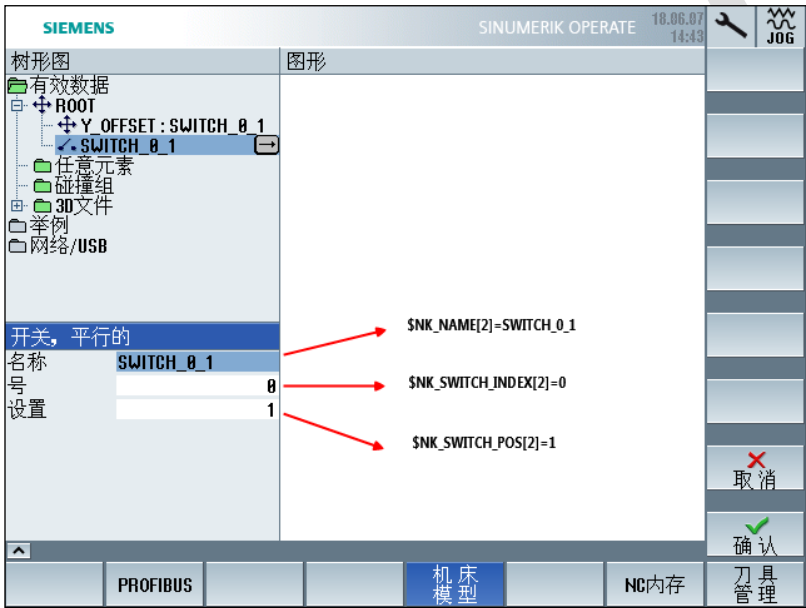


3.2.5 开关元素 \$NK_SWITCH[n], \$NK_SWITCH_INDEX[n], \$NK_SWITCH_POS[n]

定义开关元素前，应确保开关数量参数 MD18882 大于 0.

开关元素的定义中，对应关系如下：

- 名称 \$NK_NAME[n]
- 号 \$NK_SWITCH_INDEX[n]
- 设置 \$NK_SWITCH_POS[n]



3.2.5.1 \$NK_SWITCH[n] (开关)

开关元素开启/关闭取决于\$NK_SWITCH_POS[n]的数值。

\$NK_SWITCH[n]=-1; 代表开关一直开启。

当其不等于-1 时，则与设置数值比较，不相等时为开启，相等时关闭。可在 NC 程序中对\$NK_SWITCH[n]赋值，从而实现防碰撞模型的切换。

示例：

\$NK_SWITCH[1] = 2; 代表开关元素 1 将在\$NK_SWITCH_POS[1] = 2 时关闭。
对\$NK_SWITCH[n]赋值后需要执行“NEWCONF”生效。

3.2.5.2 \$NK_SWITCH_POS[n] (位置)

定义了动态开关元素关闭的设定值。

示例：

\$NK_SWITCH_POS[2] = 2; 代表开关元素\$ NK_SWITCH[2]=2 时开关关闭。

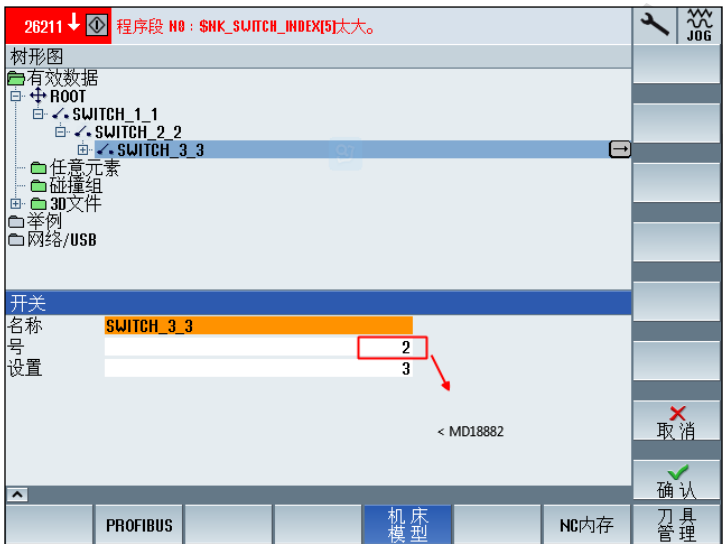
3.2.5.3 设定开关元素数量

18882	\$MN_MM_MAXNUM_KIN_SWITCHES	3	po
-------	-----------------------------	---	----

如：MD18882=3；则生效的开关元素为\$NK_SWITCH[0]、\$NK_SWITCH[1]、\$NK_SWITCH[2];

\$NK_SWITCH[3]	D	#
\$NK_SWITCH[2]	D	2
\$NK_SWITCH[1]	D	2
\$NK_SWITCH[0]	D	2

要求三个开关--【号】处填写的数值一次为 0，1，2，必须大于等于 0 且小于 MD18882 设定值

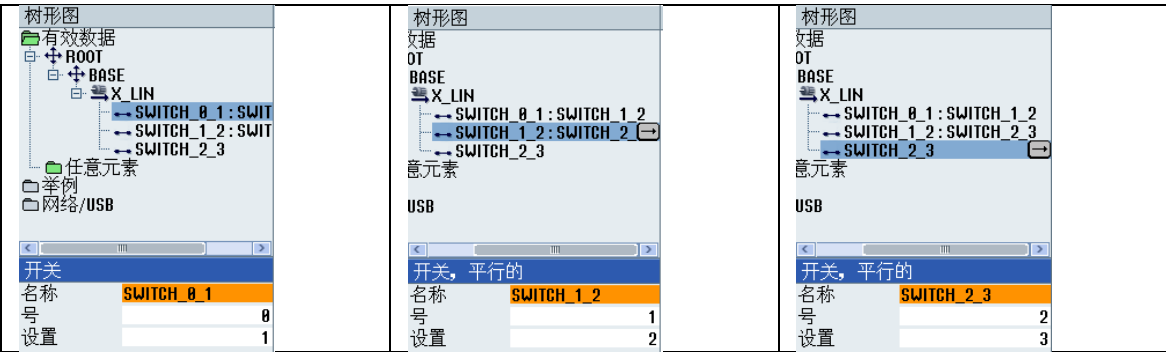


开关索引号有如下对应关系：

- \$NK_SWITCH_POS[n] 和 \$NK_SWITCH_INDEX[n] 和 \$NK_NAME[n] 的角标——对应；
- \$NK_SWITCH[i] 的角标与\$NK_SWITCH_INDEX[n]=i ——对应；

三个开关的设置如下：

SWITCH_0_1	SWITCH_1_2	SWITCH_2_3
------------	------------	------------



对应角标关系如下

变量	格式	值
\$NK_SWITCH[0]	D	1
\$NK_SWITCH[1]	D	2
\$NK_SWITCH[2]	D	3
\$NK_NAME[3]	A	SWITCH_0_1
\$NK_NAME[5]	A	SWITCH_1_2
\$NK_NAME[6]	A	SWITCH_2_3
\$NK_SWITCH_INDEX[3]	D	0
\$NK_SWITCH_INDEX[5]	D	1
\$NK_SWITCH_INDEX[6]	D	2
\$NK_SWITCH_POS[3]	D	1
\$NK_SWITCH_POS[5]	D	2
\$NK_SWITCH_POS[6]	D	3

3.2.5.4 示例：开关有条件关闭

显示开关打开的状态：即 \$NK_SWITCH[n]不等于 \$NK_SWITCH_POS[n]

设置\$NK_SWITCH[0/1/2]=0,0,0



变量	格式	值
\$NK_SWITCH[0]	D	0
\$NK_SWITCH[1]	D	0
\$NK_SWITCH[2]	D	0
\$NK_SWITCH_POS[3]	D	1
\$NK_SWITCH_POS[5]	D	2
\$NK_SWITCH_POS[6]	D	3
\$NK_SWITCH_INDEX[3]	D	0
\$NK_SWITCH_INDEX[5]	D	1
\$NK_SWITCH_INDEX[6]	D	2
\$NK_NAME[3]	A	SWITCH_0_1
\$NK_NAME[5]	A	SWITCH_1_2
\$NK_NAME[6]	A	SWITCH_2_3

显示开关关闭的状态：即 \$NK_SWITCH[n] == \$NK_SWITCH_POS[n]

设置\$NK_SWITCH[0/1/2]=1,2,3



变量	格式	值
\$NK_SWITCH[0]	D	1
\$NK_SWITCH[1]	D	2
\$NK_SWITCH[2]	D	3
\$NK_SWITCH_POS[3]	D	1
\$NK_SWITCH_POS[5]	D	2
\$NK_SWITCH_POS[6]	D	3
\$NK_SWITCH_INDEX[3]	D	0
\$NK_SWITCH_INDEX[5]	D	1
\$NK_SWITCH_INDEX[6]	D	2
\$NK_NAME[3]	A	SWITCH_0_1
\$NK_NAME[5]	A	SWITCH_1_2
\$NK_NAME[6]	A	SWITCH_2_3

4 基础知识

1.2 激活选项

Collision Avoidance ECO 6FC5800-0AS03-0YB0	☒	☒
---	-------------------------------------	-------------------------------------

1.3 设定参数

- ⌞ MD 10619 \$MN_COLLISION_TOLERANCE = 1
- ⌞ MD 10621 \$MN_COLLISION_PREP_CALC_TIME = 0.5
- ⌞ MD 10622 \$MN_COLLISION_SAFETY_DIST = 5
- ⌞ MD 18890 \$MN_MM_MAXNUM_3D_PROT_AREAS = 17
- ⌞ MD 18891 \$MN_MM_MAXIMUM_3D_WPFXPROT_ELEM = 0
- ⌞ MD 18892 \$MN_MM_MAXNUM_3D_PROT_AREA_ELEM = 68
- ⌞ MD 18893 \$MN_MM_MAXNUM_3D_T_PROT_ELEM = 1
- ⌞ MD 18894 \$MN_MM_MAXNUM_3D_FACETS_INTERN = 1000
- ⌞ MD 18895 \$MN_MM_MAXNUM_3D_FACETS = 0
- ⌞ MD 18896 \$MN_MM_MAXIMUM_3D_COLLISION = 0
- ⌞ MD 18897 \$MN_MM_MAXIMUM_3D_INTERFACE_IN = 16
- ⌞ MD 18898 \$MN_MM_MAXNUM_3D_COLL_PAIR = 10
- ⌞ MD 18899 \$MN_PROT_AREA_TOOL_MASK = 1
- ⌞ MD 51160 \$MNS_ACCESS_WRITE_CA_MACH_JOG = 6
- ⌞ MD 51161 \$MNS_ACCESS_WRITE_CA_MACH_AUTO = 3
- ⌞ MD 51162 \$MNS_ACCESS_WRITE_CA_TOOL = 1

也可执行示例包中的 Collision avoidance ECO.MPF 修改参数。

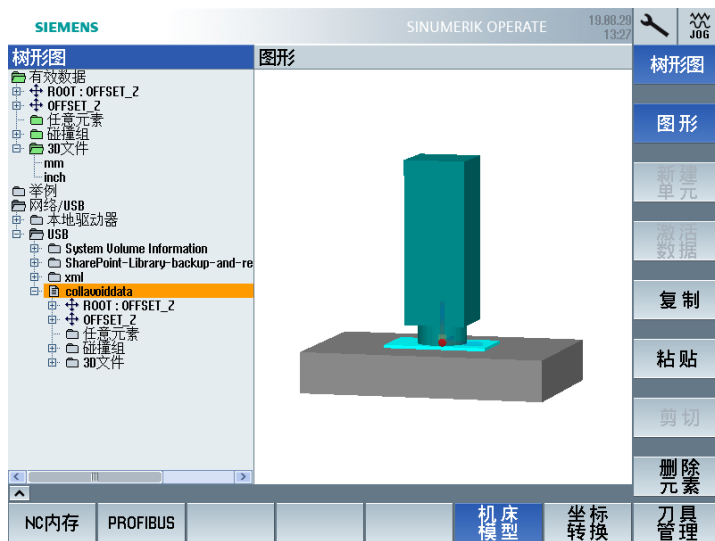
1.4 知识要点

1.4.1 模型建立

- 通过 Operate 界面建立，如第 3 章所示
- 执行 NC 程序建立，如第 4 章所示

1.4.2 复制模型

建立后的模型支持复制、粘贴。文件默认以 collavoiddata.ini 命名。
可在【机床模型】界面以树形结构展开。



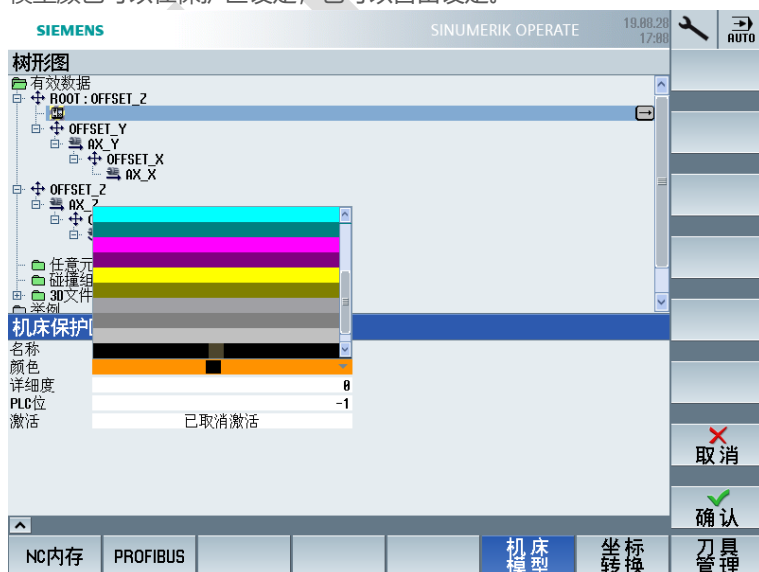
1.4.3 单元分类

模型中建立的单元分为运动单元和碰撞单元。

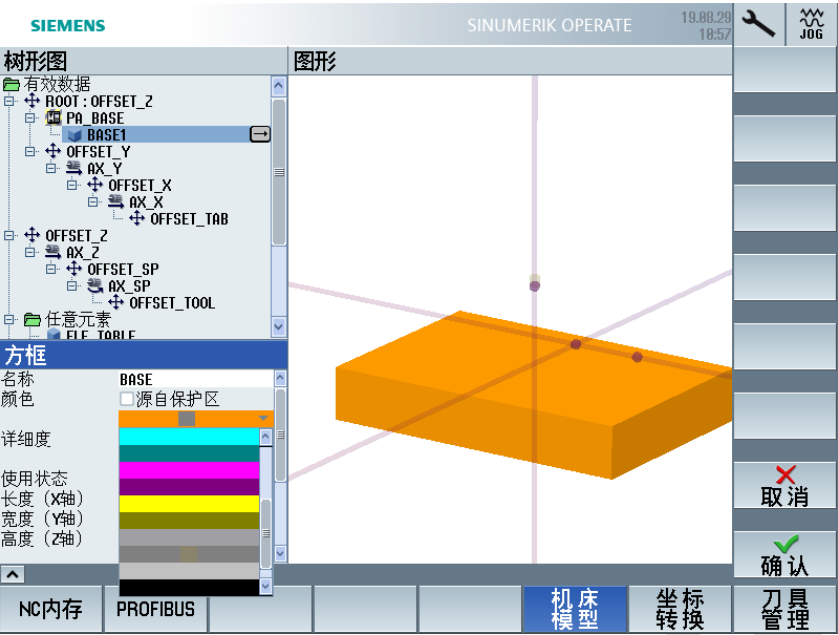
运动单元	碰撞单元
新建动态转换单元 - 旋转 - 线性轴 - 旋转轴 - Offset - 开关 - 平行旋转 - 平行线性轴 - 平行旋转轴 - 平行偏移 - 平行开关	新建碰撞单元 - 刀具保护区 - 机床保护区

1.4.4 模型颜色

模型颜色可以在保护区设定，也可以自由设定。



自由设定时将“源自保护区”的对勾去掉



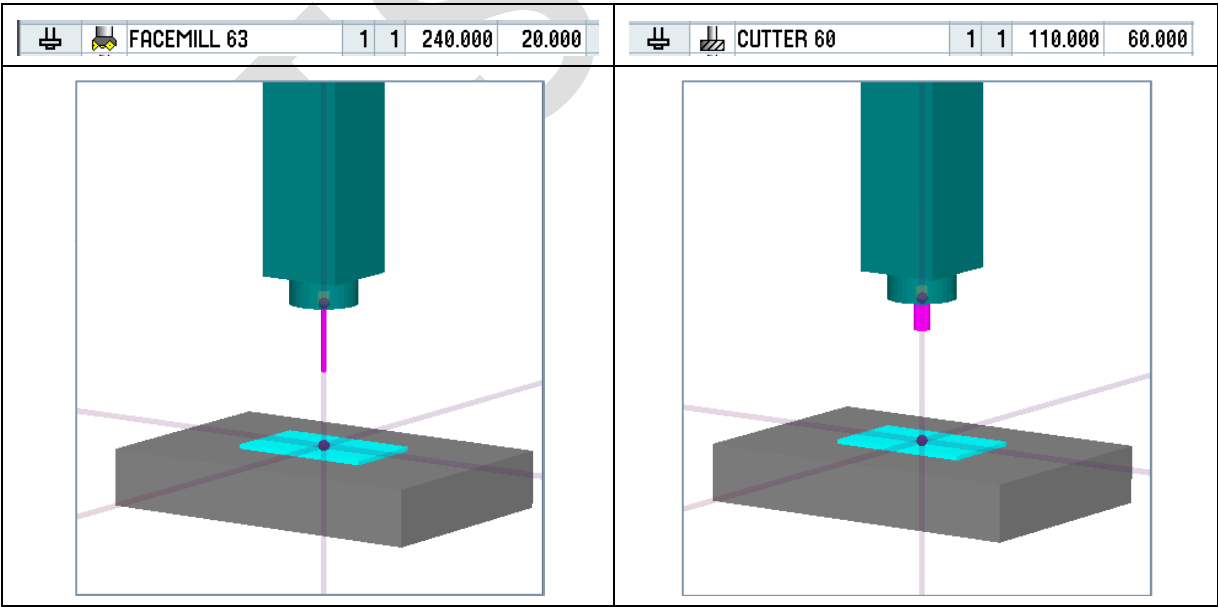
1.4.5 生效刀具

刀具保护区的数量，依据具体数量定义。

18893 \$MN_MM_MAXNUM_3D_T_PROT_ELEM 1

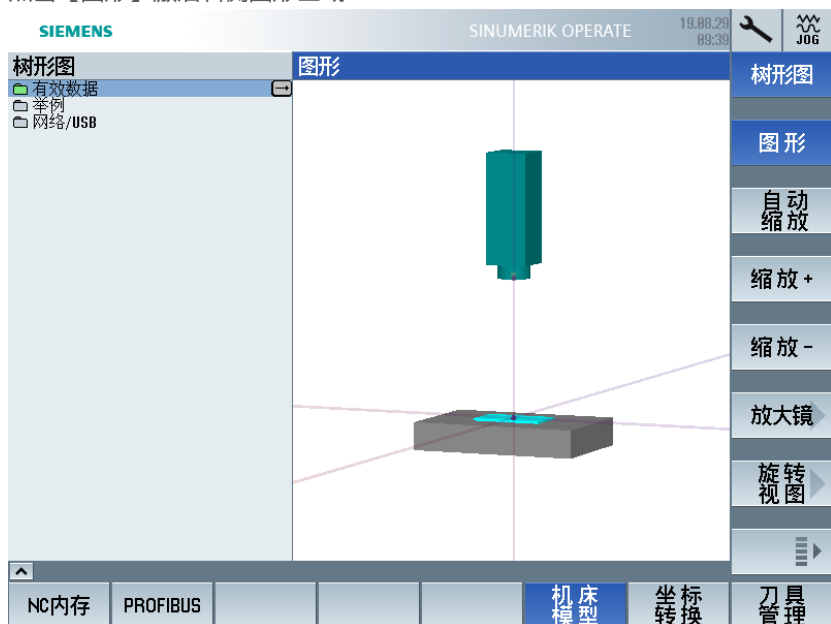
刀具保护区自动生效方式 设为 1，在碰撞监控中刀具尺寸自动生效。即换刀后自动更新刀具尺寸。

18899 \$MN_PROT_AREA_TOOL_MASK 1



1.4.6 查看图形

点击【图形】激活右侧图形区域

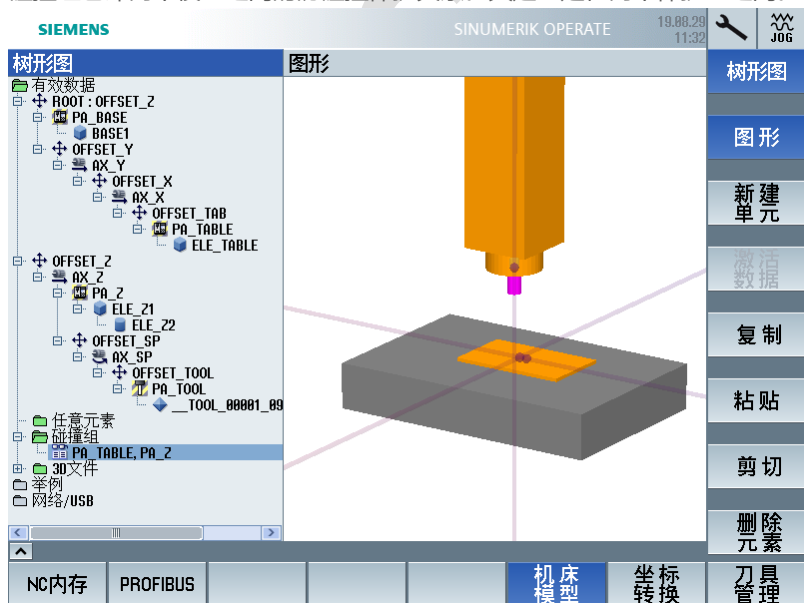


可使用右侧按钮操作图形。默认按住鼠标左键为平移，按住右键为旋转。

将光标定位到【有效数据】可显示“新建单元”等模型操作按键。

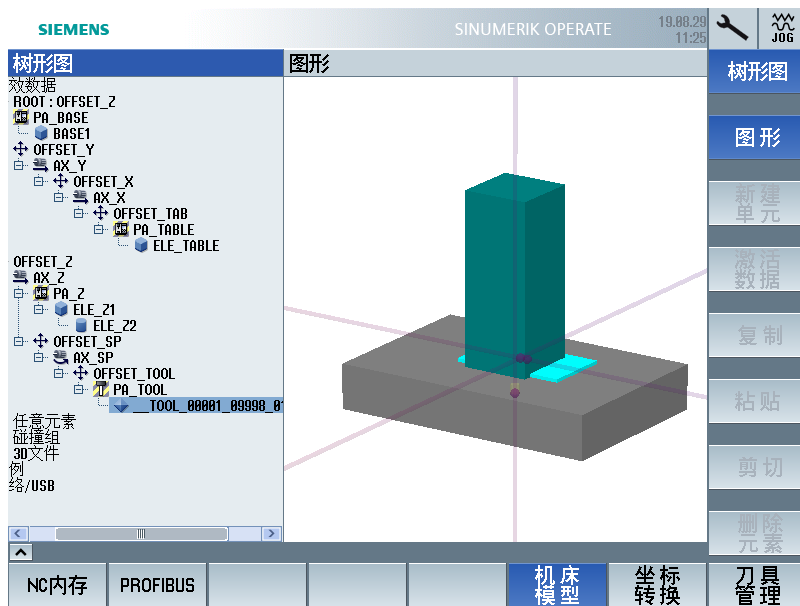
1.4.7 碰撞组

碰撞组也即两个模型之间的防碰撞保护关系。其建立是在两个保护区之间。



建立碰撞组之前，先将两个有碰撞关系的几何实体分开，不要接触。

未建立碰撞组时，模型之间不具有防碰撞保护功能。如下图

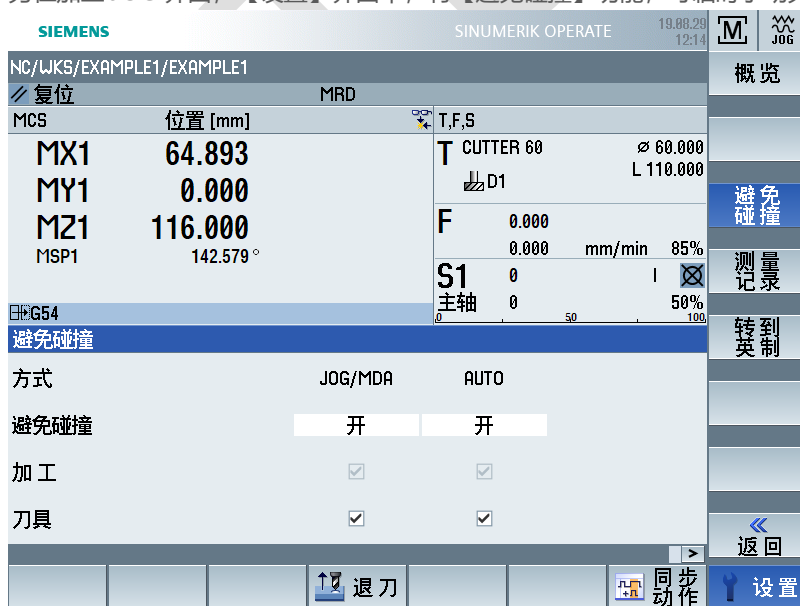


1.4.8 关闭碰撞监控

碰撞监控激活时，在加工界面有图标显示。

MCS	位置 [mm]
MX1	64.893
MY1	0.000
MZ1	116.000
MSP1	142.579 °

另在加工 JOG 界面，【设置】界面中，有【避免碰撞】功能，可临时手动关闭防碰撞监控。



如上界面信号需要在 PLC 程序中处理，否则无法在 HMI 界面上操作。
828D 的 PLC 处理机制如下。

Network 1

DB1900.DBX1008.0 DB2600.DBX2.0

Network 2

DB1900.DBX1008.1 DB2600.DBX2.1

Network 3

DB1900.DBX1008.4 DB2600.DBX2.4

Network 4

DB1900.DBX1008.5 DB2600.DBX2.5

Network 5 Kollisionsvermeidung Stopp X Achse

DB3300.DBX4005.0 DB1600.DBX4000.0

DB1900.DBB1008 [HMI→PLC]			
1008.0	E_CA_AutoMach	BOOL	Disable collision avoidance: AUTO Machine
1008.1	E_CA_AutoTools	BOOL	Disable collision avoidance: AUTO Tools
1008.2	E_CA_AutoWorkh	BOOL	Disable collision avoidance: AUTO Workholder
1008.3	E_CA_AutoWorkp	BOOL	Disable collision avoidance: AUTO Workpieces
1008.4	E_CA_JogMach	BOOL	Disable collision avoidance: JOG Machine
1008.5	E_CA_JogTools	BOOL	Disable collision avoidance: JOG Tools
1008.6	E_CA_JogWorkh	BOOL	Disable collision avoidance: JOG Workholder
1008.7	E_CA_JogWorkp	BOOL	Disable collision avoidance: JOG Workpieces

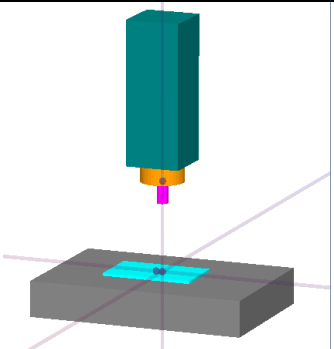
DB2600.DBB2 [PLC→NCK]			
2.0	A_CA_AutoMach	BOOL	Disable collision avoidance: AUTO Machine
2.1	A_CA_AutoTools	BOOL	Disable collision avoidance: AUTO Tools
2.2	A_CA_AutoWorkh	BOOL	Disable collision avoidance: AUTO Workholder
2.3	A_CA_AutoWorkp	BOOL	Disable collision avoidance: AUTO Workpieces
2.4	A_CA_JogMach	BOOL	Disable collision avoidance: JOG Machine
2.5	A_CA_JogTools	BOOL	Disable collision avoidance: JOG Tools
2.6	A_CA_JogWorkh	BOOL	Disable collision avoidance: JOG Workholder
2.7	A_CA_JogWorkp	BOOL	Disable collision avoidance: JOG Workpieces

DB3300.DBB4005 (反馈信号：已停止防碰撞监控)			
4005.0	E_CollCheckStop	BOOL	Stop of Collision Check

5 建立模型示例

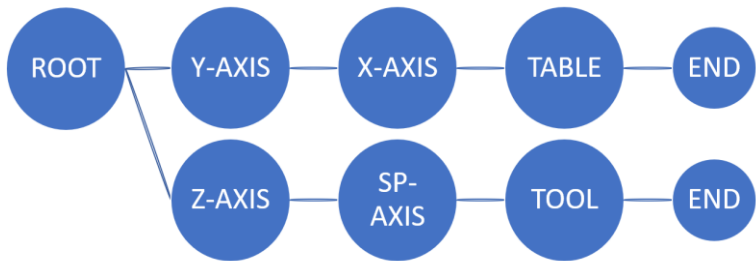
1.4.9 模型尺寸

表 1: 模型尺寸

示意图	机床模型元素	模型尺寸	坐标偏移 (MCS)
	床身	X=1200, Y=800, Z=200	X=0, Y=0, Z=-115
	工作台	X=500, Y=300, Z=30	X=0, Y=0, Z=-15
	主轴箱体 1	X=300, Y=300, Z=800	X=0, Y=0, Z=500
	主轴箱体 2	H=100, R=120	X=0, Y=0, Z=-450

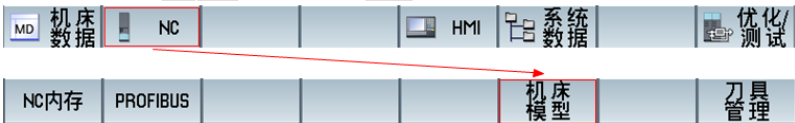
如下示例为通过 Operate 界面建立防碰撞模型

1.4.10 规划运动链



1.4.11 进入创建界面

进入【调试】--【NC】--【机床模型】

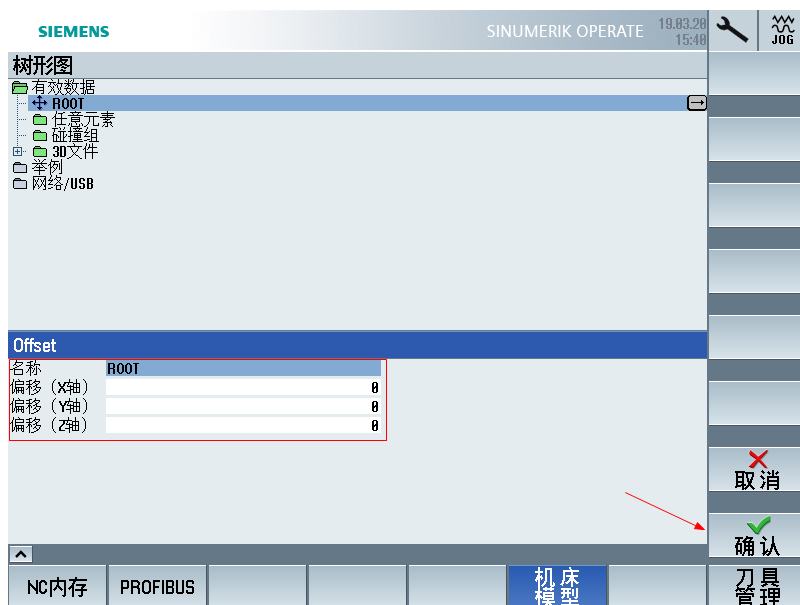


1.4.12 建立运动链

根据规划好的运动链建立。

1.4.12.1 建立 ROOT

点击“新建单元”，选择“运动单元”，选择“offset”



1.4.12.2 建立 Y-AXIS

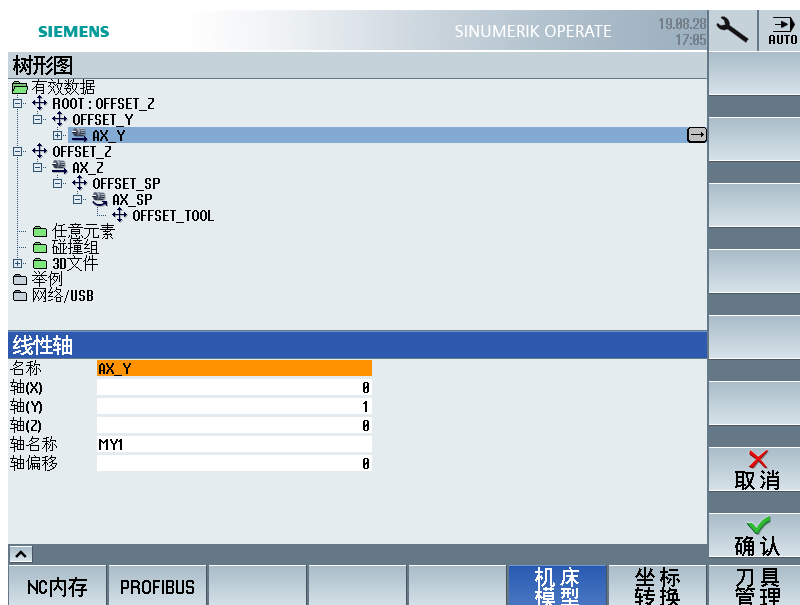
a. 建立 OFFSET

新建单元; “运动单元”——“offset”, 名称 OFFSET_Y



b. 建立线性轴

新建单元; “运动单元”——“线性轴”, 名称 AX_Y, 轴名称需与 MD10000 号机床轴名称一致, 轴(X,Y,Z)设 0,1,0 代表矢量方向为 Y。



1.4.12.3 建立 X-AXIS

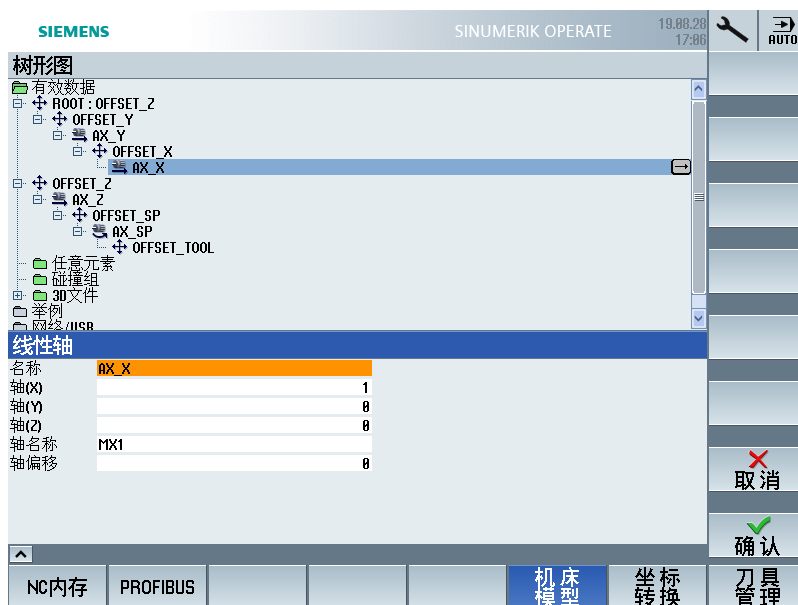
a. 建立 OFFSET

新建单元; “运动单元”——“offset”, 名称 OFFSET_X



b. 建立线性轴

新建单元; “运动单元”——“线性轴”, 名称 AX_X, 轴名称需与 MD10000 号机床轴名称一致, 轴(X,Y,Z)设 1,0,0 代表矢量方向为 X。



1.4.12.4 建立 Table

a. 建立 OFFSET

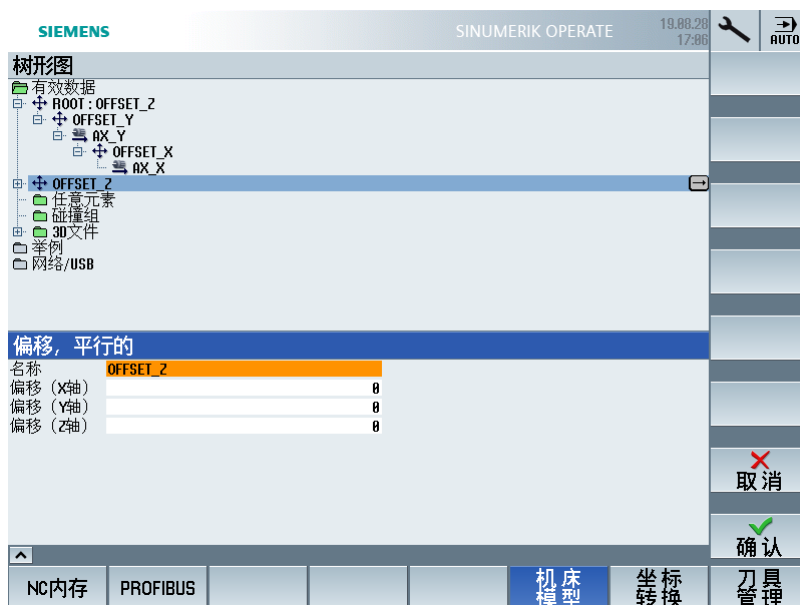
新建单元; “运动单元”——“offset”, 名称 OFFSET_TABLE, 偏移 0,0,0,



1.4.12.5 建立 Z-AXIS

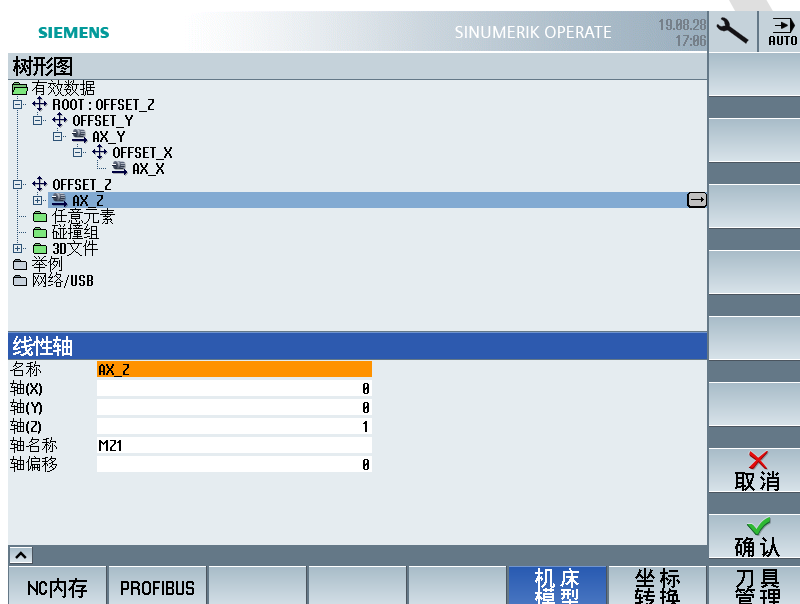
a. 建立 OFFSET

将光标定位在 ROOT-OFFSET 上, 新建单元; “运动单元”——“平行偏移”, 名称 OFFSET_Z



b. 建立线性轴

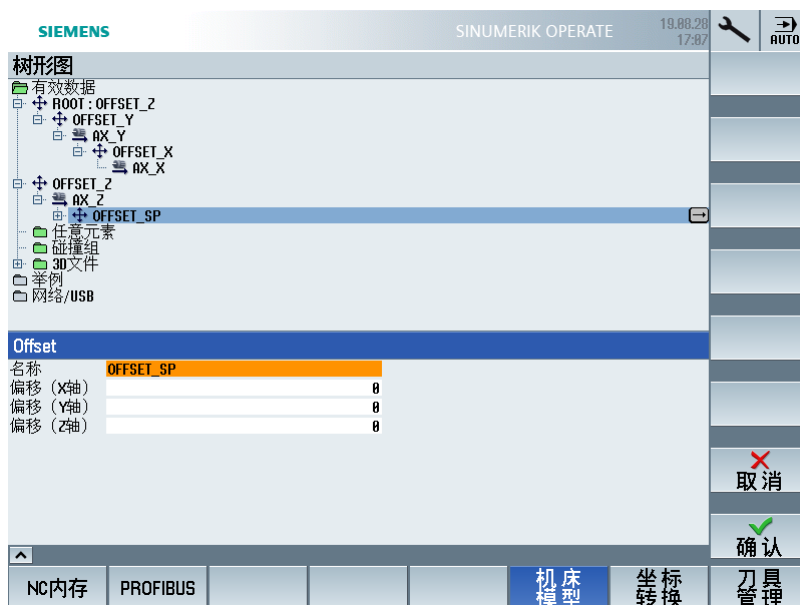
新建单元; “运动单元”——“线性轴”, 名称 AX_Z, 轴名称需与 MD10000 号机床轴名称一致, 轴(X,Y,Z)设 0,0,1 代表矢量方向为 Z。



1.4.12.6 建立 SP-AXIS

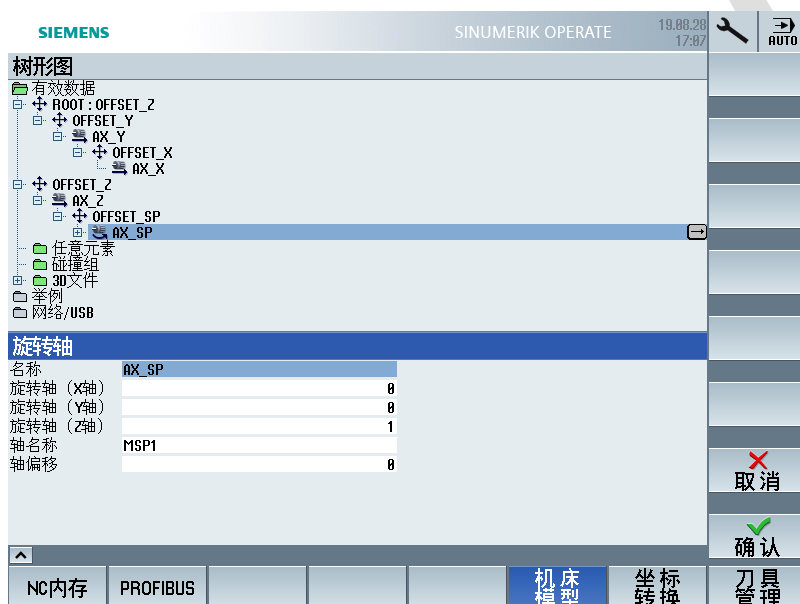
b. 建立 OFFSET

新建单元; “运动单元”——“offset”, 名称 OFFSET_Z



c. 建立旋转轴

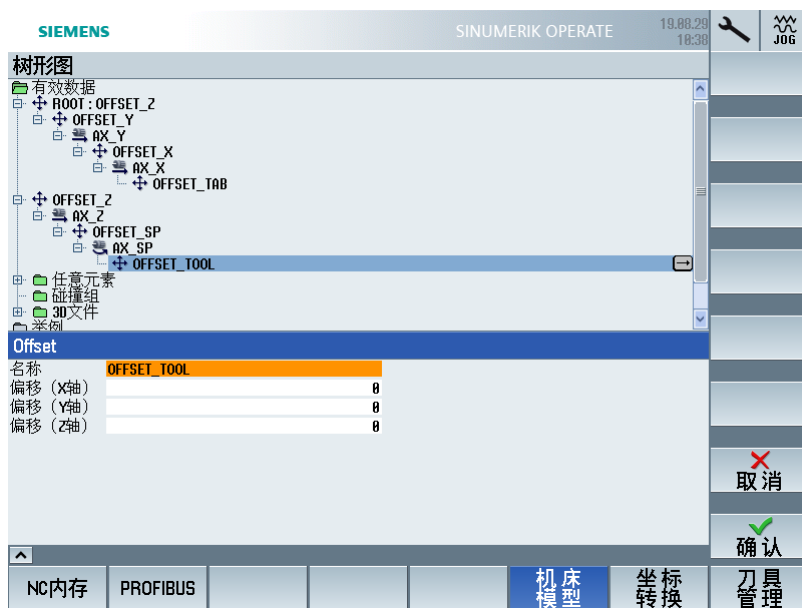
新建单元;“运动单元”——“旋转轴”, 名称 AX_SP, 轴名称需与 MD10000 号机床轴名称一致, 轴(X,Y,Z)设 0,0,1, 代表绕 Z 方向旋转。



1.4.12.7 建立 TOOL

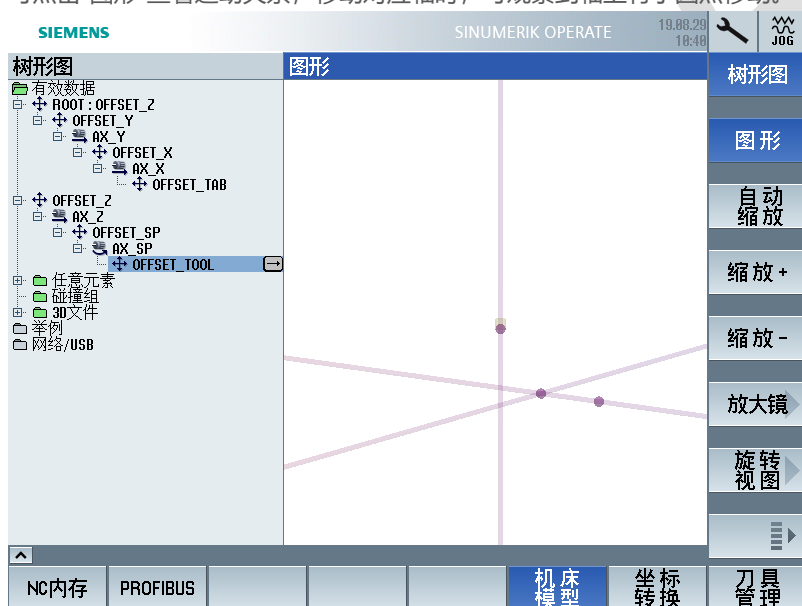
c. 建立 OFFSET

新建单元;“运动单元”——“offset”, 名称 OFFSET_TOOL



至此，运动关系已建立完毕。

可点击“图形”查看运动关系，移动对应轴时，可观察到轴上有小圆点移动。

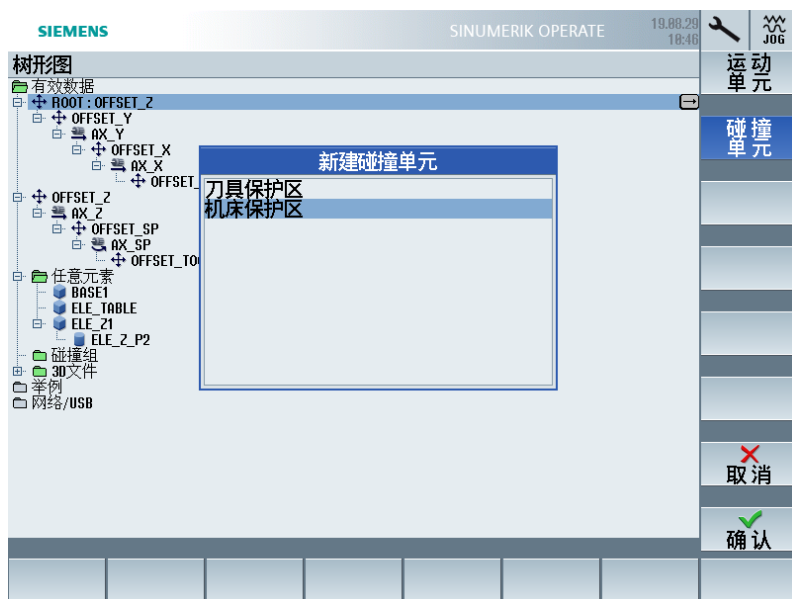


1.4.13 建立模型实体

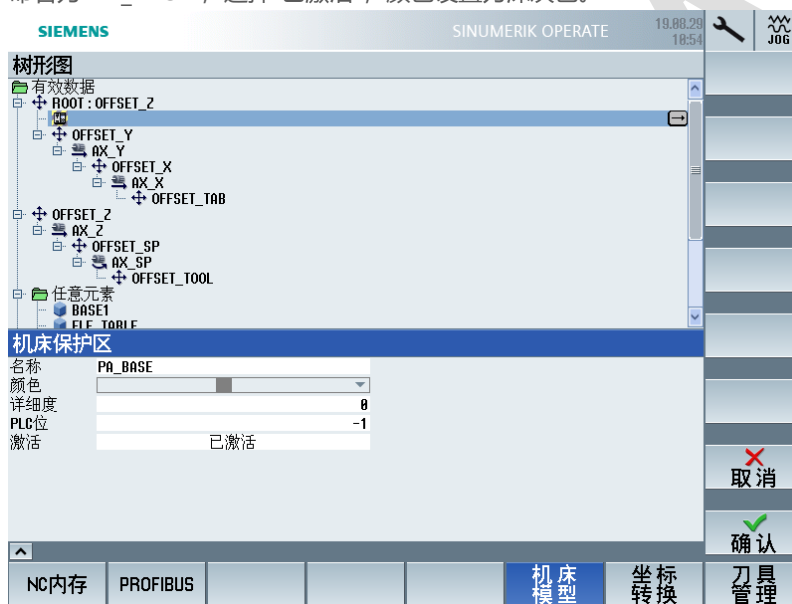
1.4.13.1 机床床身

a. 建立机床保护区

在 root 下建立机床保护区，新建“碰撞单元”——“机床保护区”

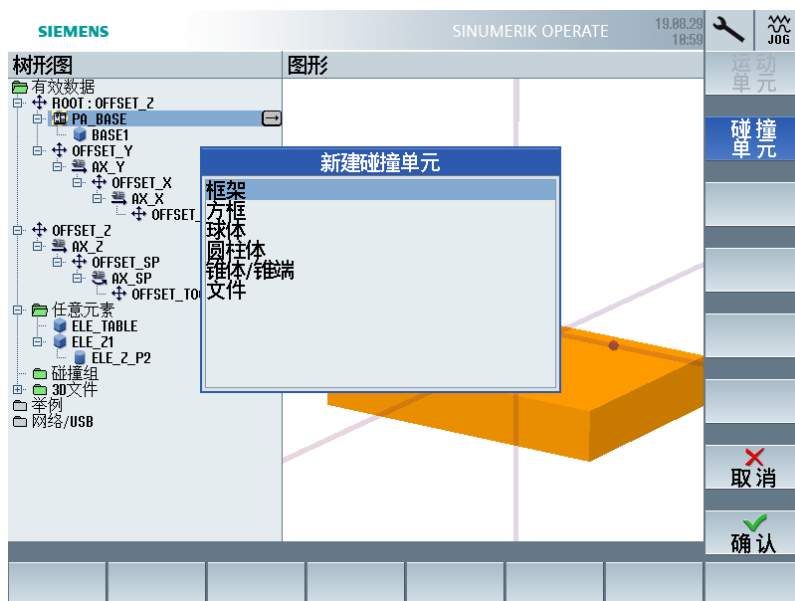


命名为“PA_BASE”，选择“已激活”，颜色设置为深灰色。



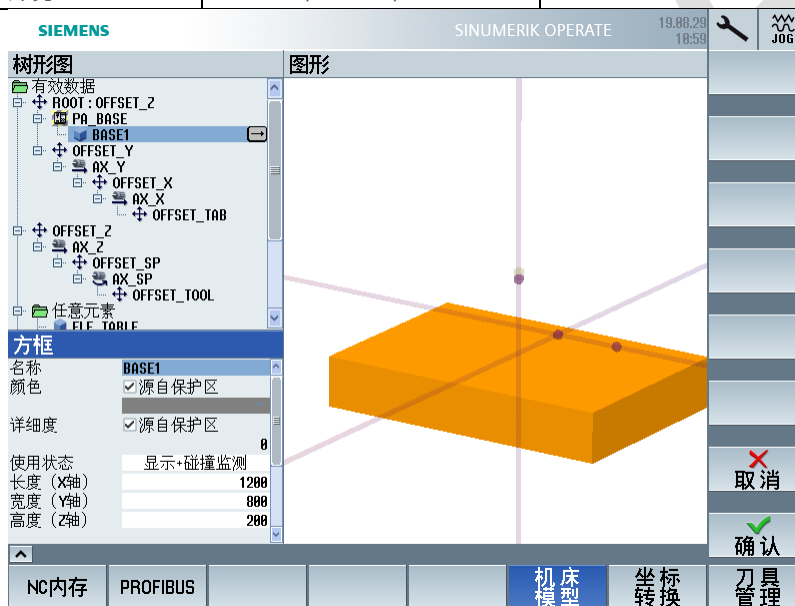
b. 建立几何体

新建单元；“碰撞单元”——“方框”



名称自由设定，BASE1，长宽高及 offset 按表 1 数据。

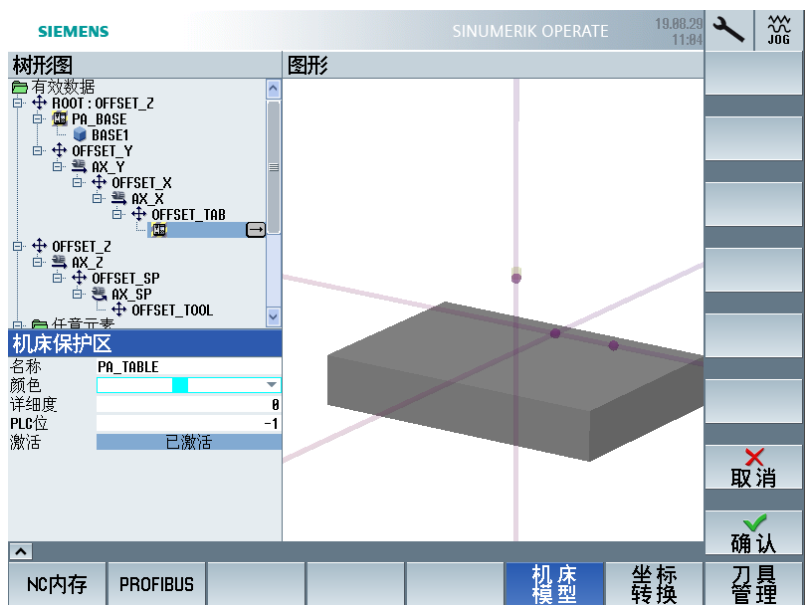
床身	X=1200, Y=800, Z=200	OFFSET	X=0, Y=0, Z=-115
----	----------------------	--------	------------------



1.4.13.2 工作台

a. 建立机床保护区

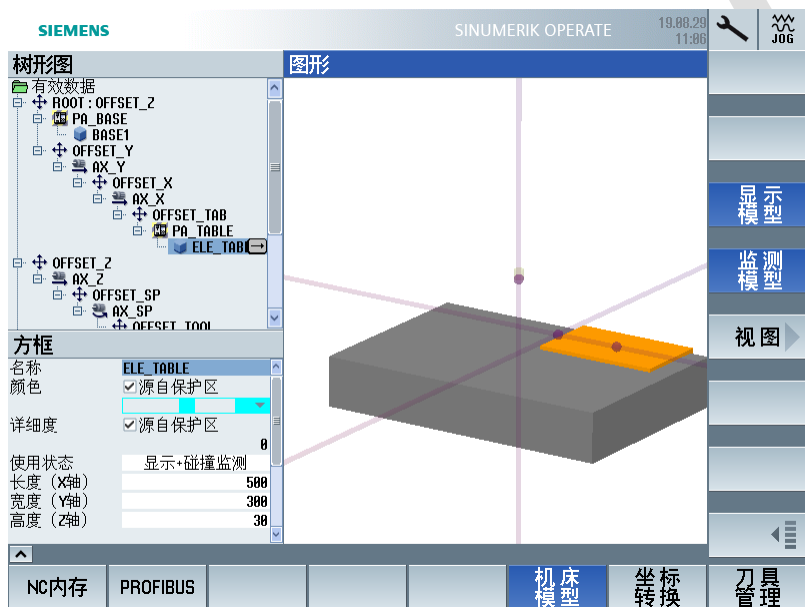
在 OFFSET_TABLE 下建立机床保护区，新建“碰撞单元”——“机床保护区”命名为“PA_TABLE”，选择“已激活”，颜色设置为青绿色。



b. 建立几何体

新建单元；“碰撞单元”——“方框”，名称自由设定，BASE1，长宽高及 offset 按表 1 数据。

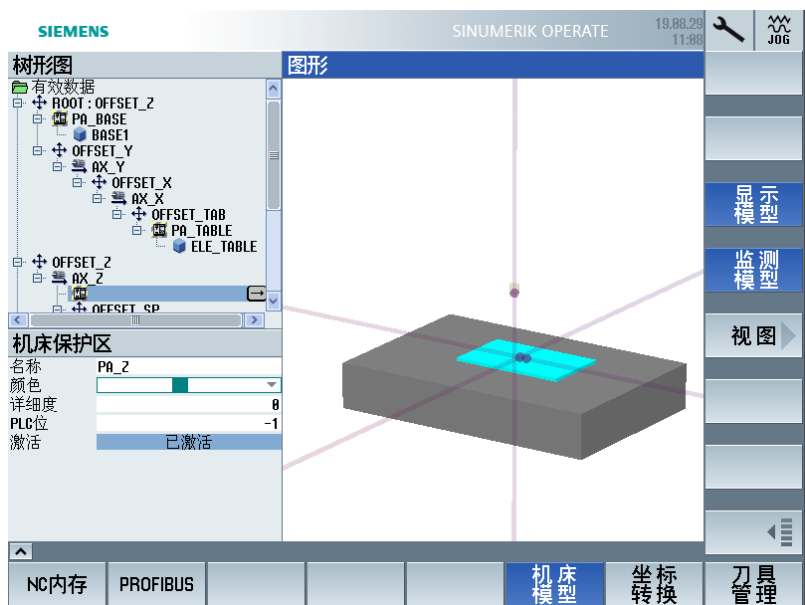
工作台	X=500, Y=300, Z=30	OFFSET	X=0, Y=0, Z=-15
-----	--------------------	--------	-----------------



1.4.13.3 主轴箱 (Z 轴)

a. 建立机床保护区

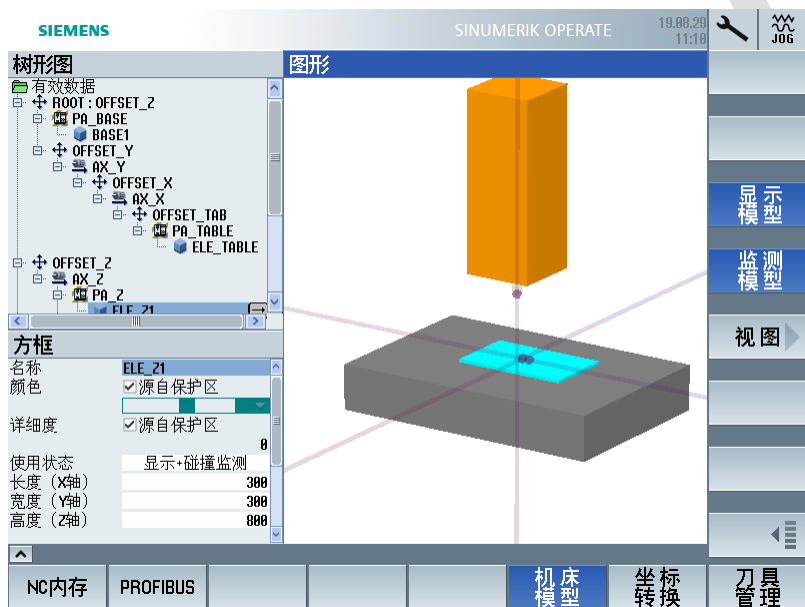
在线性轴 AX_Z 下建立机床保护区，新建“碰撞单元”——“机床保护区”命名为“PA_Z”，选择“已激活”，颜色设置为深绿色。



b. 建立几何体

新建单元;“碰撞单元”——“方框”, 名称自由设定, ELE_Z1, 长宽高及 offset 按表 1 数据。

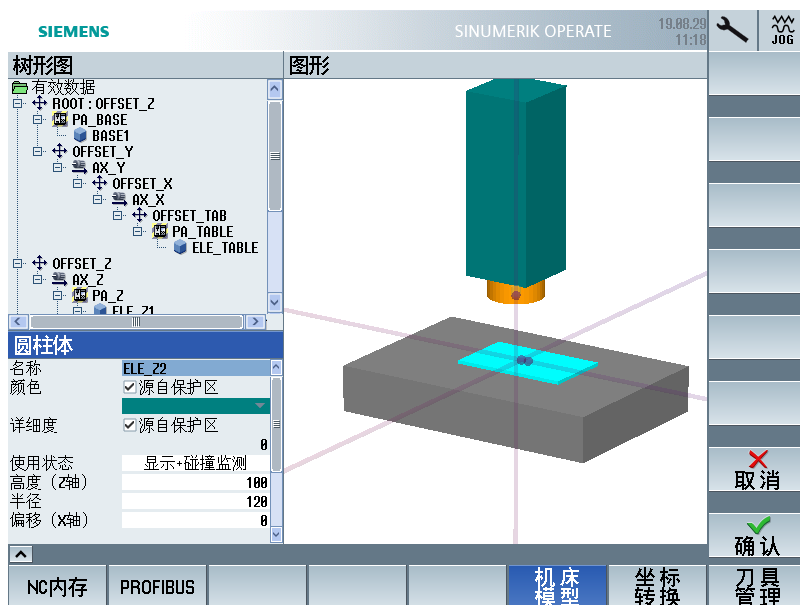
主轴箱体 1	X=300, Y=300, Z=800	OFFSET	X=0, Y=0, Z=500
--------	---------------------	--------	-----------------



c. 建立几何体

新建单元;“碰撞单元”——“圆柱体”, 名称自由设定, ELE_Z2, 长宽高及 offset 按表 1 数据。

主轴箱体 2	H=100, R=120	OFFSET	X=0, Y=0, Z=-450
--------	--------------	--------	------------------

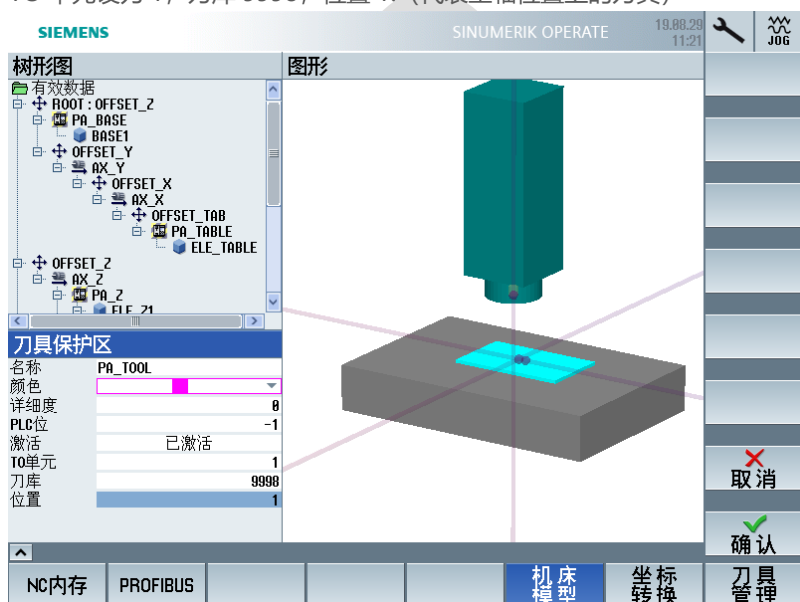


1.4.13.4 刀具

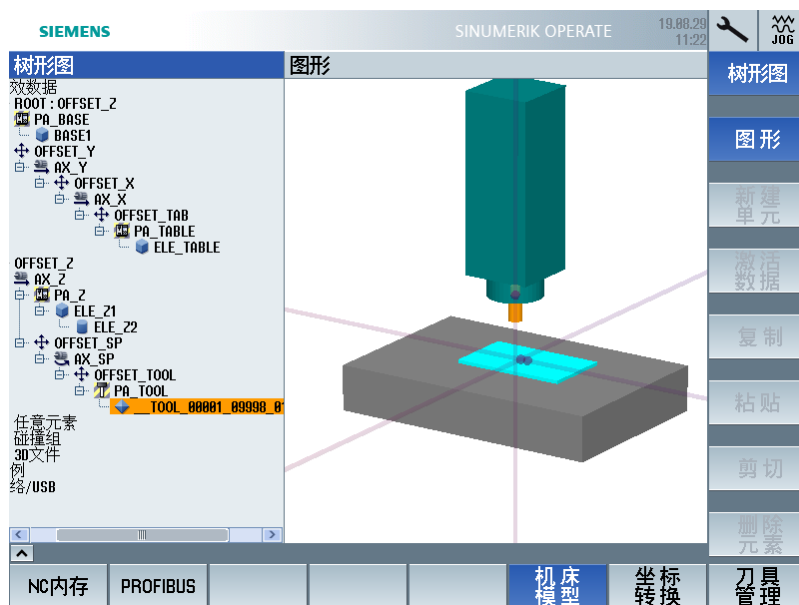
a. 建立刀具保护区

在 OFFSET_TOOL 下建立刀具保护区，新建“碰撞单元”——“刀具保护区”命名为“PA_TOOL”，选择“已激活”，颜色设置为紫色。

TO 单元设为 1，刀库 9998，位置 1。（代表主轴位置上的刀具）



建立完成，复位刷新即可。

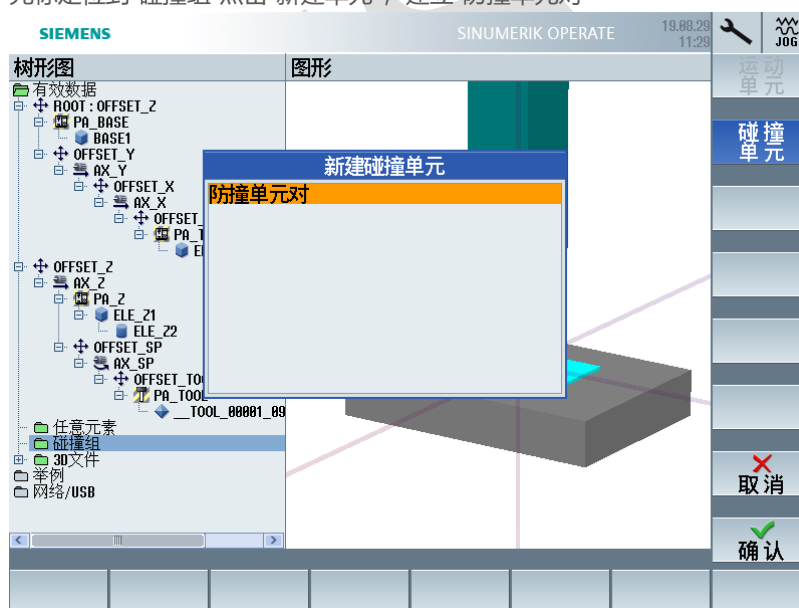


至此，模型建立完毕。

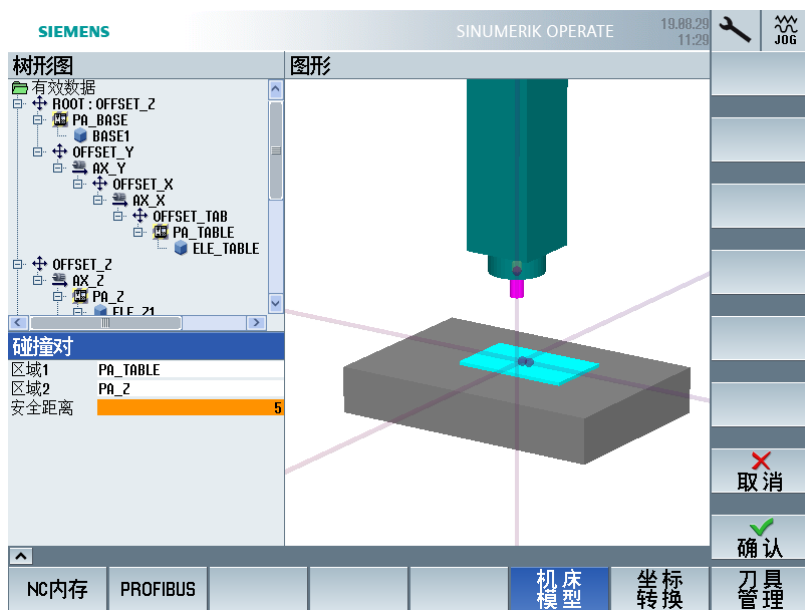
1.4.14 建立碰撞组

- PA_TABLE, PA_Z
- PA_TABLE, PA_TOOL
- PA_BASE, PA_Z
- PA_BASE, PA_TOOL

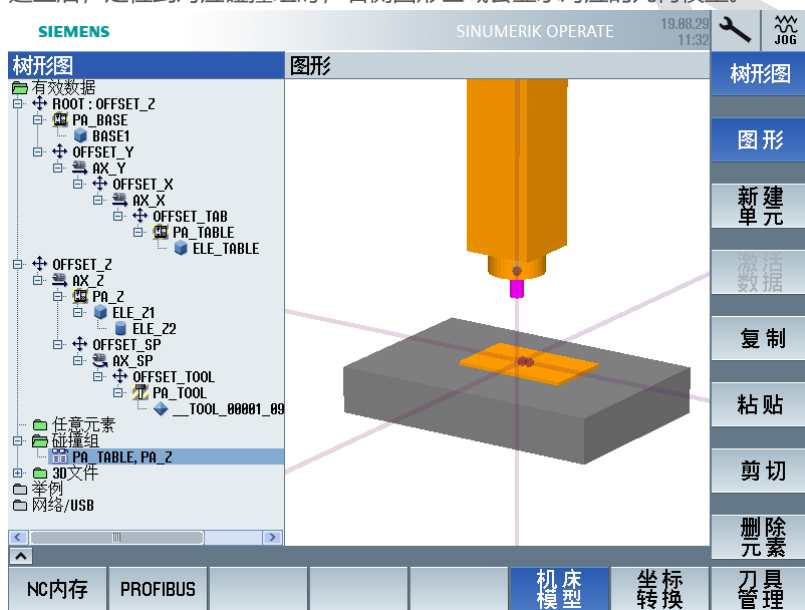
光标定位到“碰撞组”点击“新建单元”，建立“防撞单元对”



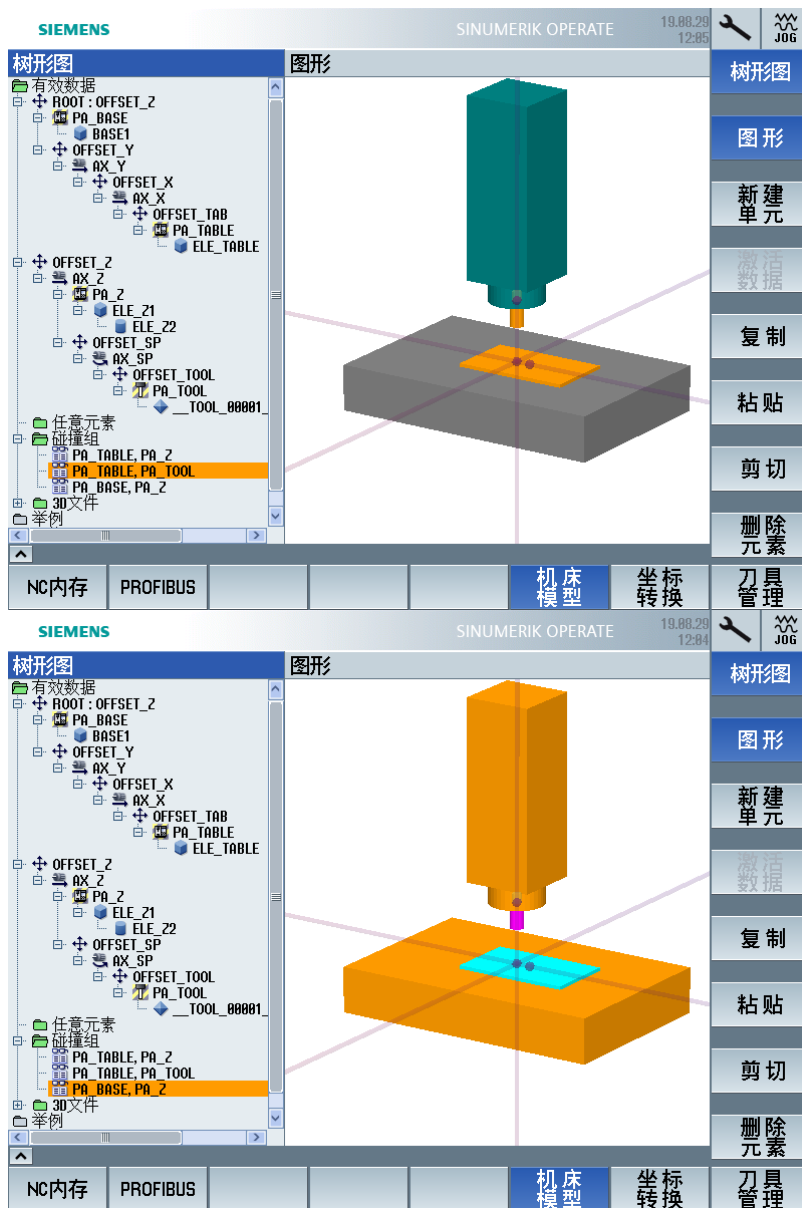
区域中填写保护区名称 PA_TABLE, PA_Z，安全距离用于位置监控。自定义，本例设为 5.

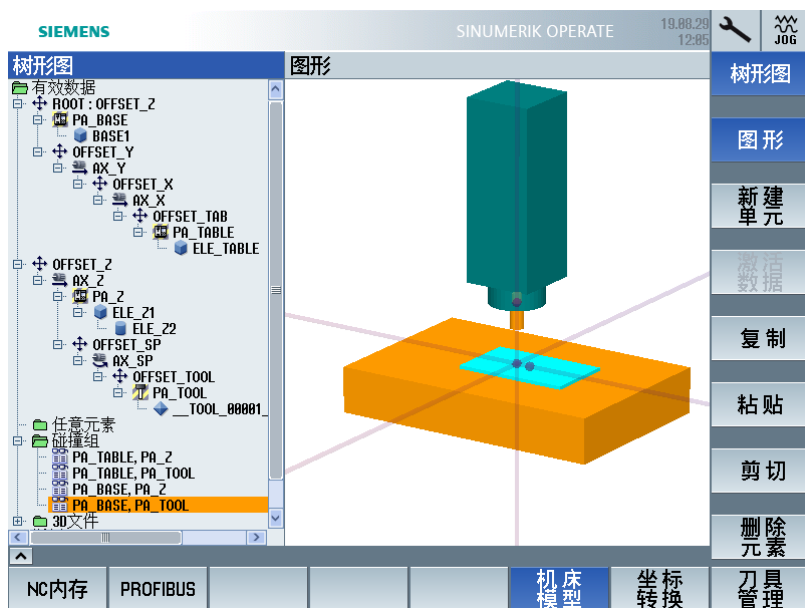


建立后，定位到对应碰撞组时，右侧图形区域会显示对应的几何模型。



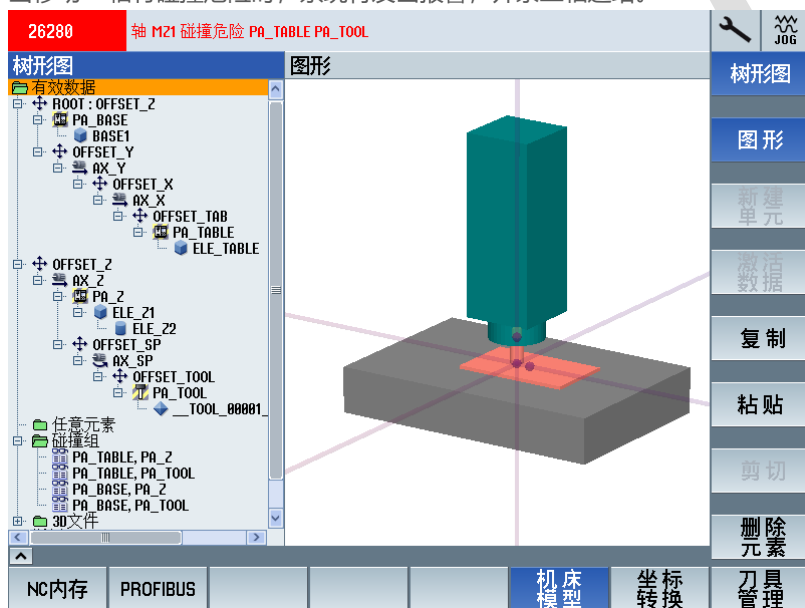
依次建立多个碰撞组。





1.4.15 验证

当移动 Z 轴有碰撞危险时，系统将发出报警，并禁止轴进给。



6 程序示例

如下是上述示例的 NC 程序实现方式。即不需逐个建立模型，执行程序后自动生成防碰撞模型。

```
;COLLISION AVOIDANCE DATA
;LOC_ROOT_NAME[0]="ROOT"
;LOC_CAD_FILE_DIR[0]="collavoiddata.stl_dir"
$NK_NAME[0]="ROOT"
$NK_NEXT[0]="OFFSET_Y"
$NK_PARALLEL[0]="OFFSET_Z"
$NK_TYPE[0]="OFFSET"
$NK_OFF_DIR[0,0]=0
$NK_OFF_DIR[0,1]=0
$NK_OFF_DIR[0,2]=0
$NK_AXIS[0]=" "
$NK_A_OFF[0]=0
$NK_SWITCH_INDEX[0]=-1
$NK_SWITCH_POS[0]=0
$NK_NAME[1]="OFFSET_Y"
$NK_NEXT[1]="AX_Y"
$NK_PARALLEL[1]=" "
$NK_TYPE[1]="OFFSET"
$NK_OFF_DIR[1,0]=0
$NK_OFF_DIR[1,1]=0
$NK_OFF_DIR[1,2]=0
$NK_AXIS[1]=" "
$NK_A_OFF[1]=0
$NK_SWITCH_INDEX[1]=-1
$NK_SWITCH_POS[1]=0
$NK_NAME[2]="AX_Y"
$NK_NEXT[2]="OFFSET_X"
$NK_PARALLEL[2]=" "
$NK_TYPE[2]="AXIS_LIN"
$NK_OFF_DIR[2,0]=0
$NK_OFF_DIR[2,1]=1
$NK_OFF_DIR[2,2]=0
$NK_AXIS[2]="MY1"
$NK_A_OFF[2]=0
$NK_SWITCH_INDEX[2]=-1
$NK_SWITCH_POS[2]=0
$NK_NAME[3]="OFFSET_X"
```

```
$NK_NEXT[3]="AX_X"  
$NK_PARALLEL[3]=""  
$NK_TYPE[3]="OFFSET"  
$NK_OFF_DIR[3,0]=0  
$NK_OFF_DIR[3,1]=0  
$NK_OFF_DIR[3,2]=0  
$NK_AXIS[3]=""  
$NK_A_OFF[3]=0  
$NK_SWITCH_INDEX[3]=-1  
$NK_SWITCH_POS[3]=0  
$NK_NAME[4]="AX_X"  
$NK_NEXT[4]="OFFSET_TAB"  
$NK_PARALLEL[4]=""  
$NK_TYPE[4]="AXIS_LIN"  
$NK_OFF_DIR[4,0]=1  
$NK_OFF_DIR[4,1]=0  
$NK_OFF_DIR[4,2]=0  
$NK_AXIS[4]="MX1"  
$NK_A_OFF[4]=0  
$NK_SWITCH_INDEX[4]=-1  
$NK_SWITCH_POS[4]=0  
$NK_NAME[5]="OFFSET_TAB"  
$NK_NEXT[5]=""  
$NK_PARALLEL[5]=""  
$NK_TYPE[5]="OFFSET"  
$NK_OFF_DIR[5,0]=0  
$NK_OFF_DIR[5,1]=0  
$NK_OFF_DIR[5,2]=0  
$NK_AXIS[5]=""  
$NK_A_OFF[5]=0  
$NK_SWITCH_INDEX[5]=-1  
$NK_SWITCH_POS[5]=0  
$NP_PROT_NAME[0]="PA_BASE"  
$NP_CHAIN_ELEM[0]="ROOT"  
$NP_PROT_TYPE[0]="MACHINE"  
$NP_1ST_PROT[0]="BASE1"  
$NP_PROT_COLOR[0]='Hff808080'  
$NP_PROT_D_LEVEL[0]=0  
$NP_BIT_NO[0]=-1  
$NP_INIT_STAT[0]="A"  
$NP_INDEX[0,0]=0
```

```
$NP_INDEX[0,1]=0
$NP_INDEX[0,2]=0
$NP_NAME[0]="BASE1"
$NP_NEXT[0]=" "
$NP_NEXTP[0]=" "
$NP_COLOR[0]='H0'
$NP_D_LEVEL[0]=0
$NP_USAGE[0]="A"
$NP_TYPE[0]="BOX"
$NP_FILENAME[0]=" "
$NP_PARA[0,0]=1200
$NP_PARA[0,1]=800
$NP_PARA[0,2]=200
$NP_OFF[0,0]=0
$NP_OFF[0,1]=0
$NP_OFF[0,2]=-115
$NP_DIR[0,0]=0
$NP_DIR[0,1]=0
$NP_DIR[0,2]=0
$NP_ANG[0]=0
$NP_PROT_NAME[1]="PA_TABLE"
$NP_CHAIN_ELEM[1]="OFFSET_TAB"
$NP_PROT_TYPE[1]="MACHINE"
$NP_1ST_PROT[1]="ELE_TABLE"
$NP_PROT_COLOR[1]='Hff00ffff'
$NP_PROT_D_LEVEL[1]=0
$NP_BIT_NO[1]=-1
$NP_INIT_STAT[1]="A"
$NP_INDEX[1,0]=0
$NP_INDEX[1,1]=0
$NP_INDEX[1,2]=0
$NP_NAME[1]="ELE_TABLE"
$NP_NEXT[1]=" "
$NP_NEXTP[1]=" "
$NP_COLOR[1]='H0'
$NP_D_LEVEL[1]=0
$NP_USAGE[1]="A"
$NP_TYPE[1]="BOX"
$NP_FILENAME[1]=" "
$NP_PARA[1,0]=500
$NP_PARA[1,1]=300
```

```
$NP_PARA[1,2]=30
$NP_OFF[1,0]=0
$NP_OFF[1,1]=0
$NP_OFF[1,2]=-15
$NP_DIR[1,0]=0
$NP_DIR[1,1]=0
$NP_DIR[1,2]=0
$NP_ANG[1]=0
$NK_NAME[6]="OFFSET_Z"
$NK_NEXT[6]="AX_Z"
$NK_PARALLEL[6]=" "
$NK_TYPE[6]="OFFSET"
$NK_OFF_DIR[6,0]=0
$NK_OFF_DIR[6,1]=0
$NK_OFF_DIR[6,2]=0
$NK_AXIS[6]=" "
$NK_A_OFF[6]=0
$NK_SWITCH_INDEX[6]=-1
$NK_SWITCH_POS[6]=0
$NK_NAME[7]="AX_Z"
$NK_NEXT[7]="OFFSET_SP"
$NK_PARALLEL[7]=" "
$NK_TYPE[7]="AXIS_LIN"
$NK_OFF_DIR[7,0]=0
$NK_OFF_DIR[7,1]=0
$NK_OFF_DIR[7,2]=1
$NK_AXIS[7]="MZ1"
$NK_A_OFF[7]=0
$NK_SWITCH_INDEX[7]=-1
$NK_SWITCH_POS[7]=0
$NK_NAME[8]="OFFSET_SP"
$NK_NEXT[8]="AX_SP"
$NK_PARALLEL[8]=" "
$NK_TYPE[8]="OFFSET"
$NK_OFF_DIR[8,0]=0
$NK_OFF_DIR[8,1]=0
$NK_OFF_DIR[8,2]=0
$NK_AXIS[8]=" "
$NK_A_OFF[8]=0
$NK_SWITCH_INDEX[8]=-1
$NK_SWITCH_POS[8]=0
```

```
$NK_NAME[9]="AX_SP"  
$NK_NEXT[9]="OFFSET_TOOL"  
$NK_PARALLEL[9]=""  
$NK_TYPE[9]="AXIS_ROT"  
$NK_OFF_DIR[9,0]=0  
$NK_OFF_DIR[9,1]=0  
$NK_OFF_DIR[9,2]=1  
$NK_AXIS[9]="MSP1"  
$NK_A_OFF[9]=0  
$NK_SWITCH_INDEX[9]=-1  
$NK_SWITCH_POS[9]=0  
$NK_NAME[10]="OFFSET_TOOL"  
$NK_NEXT[10]=""  
$NK_PARALLEL[10]=""  
$NK_TYPE[10]="OFFSET"  
$NK_OFF_DIR[10,0]=0  
$NK_OFF_DIR[10,1]=0  
$NK_OFF_DIR[10,2]=0  
$NK_AXIS[10]=""  
$NK_A_OFF[10]=0  
$NK_SWITCH_INDEX[10]=-1  
$NK_SWITCH_POS[10]=0  
$NP_PROT_NAME[2]="PA_Z"  
$NP_CHAIN_ELEM[2]="AX_Z"  
$NP_PROT_TYPE[2]="MACHINE"  
$NP_1ST_PROT[2]="ELE_Z1"  
$NP_PROT_COLOR[2]='Hff008080'  
$NP_PROT_D_LEVEL[2]=0  
$NP_BIT_NO[2]=-1  
$NP_INIT_STAT[2]="A"  
$NP_INDEX[2,0]=0  
$NP_INDEX[2,1]=0  
$NP_INDEX[2,2]=0  
$NP_NAME[2]="ELE_Z1"  
$NP_NEXT[2]="ELE_Z2"  
$NP_NEXTP[2]=""  
$NP_COLOR[2]='H0'  
$NP_D_LEVEL[2]=0  
$NP_USAGE[2]="A"  
$NP_TYPE[2]="BOX"  
$NP_FILENAME[2]=""
```

```
$NP_PARA[2,0]=300
$NP_PARA[2,1]=300
$NP_PARA[2,2]=800
$NP_OFF[2,0]=0
$NP_OFF[2,1]=0
$NP_OFF[2,2]=500
$NP_DIR[2,0]=0
$NP_DIR[2,1]=0
$NP_DIR[2,2]=0
$NP_ANG[2]=0
$NP_NAME[3]="ELE_Z2"
$NP_NEXT[3]=" "
$NP_NEXTP[3]=" "
$NP_COLOR[3]='H0'
$NP_D_LEVEL[3]=0
$NP_USAGE[3]="A"
$NP_TYPE[3]="CYLINDER"
$NP_FILENAME[3]=" "
$NP_PARA[3,0]=100
$NP_PARA[3,1]=120
$NP_PARA[3,2]=0
$NP_OFF[3,0]=0
$NP_OFF[3,1]=0
$NP_OFF[3,2]=-450
$NP_DIR[3,0]=0
$NP_DIR[3,1]=0
$NP_DIR[3,2]=0
$NP_ANG[3]=0
$NP_PROT_NAME[3]="PA_TOOL"
$NP_CHAIN_ELEM[3]="OFFSET_TOOL"
$NP_PROT_TYPE[3]="TOOL"
$NP_1ST_PROT[3]=" "
$NP_PROT_COLOR[3]='Hffff00ff'
$NP_PROT_D_LEVEL[3]=0
$NP_BIT_NO[3]=-1
$NP_INIT_STAT[3]="A"
$NP_INDEX[3,0]=1
$NP_INDEX[3,1]=9998
$NP_INDEX[3,2]=1
$NP_COLL_PAIR[0,0]="PA_BASE"
$NP_COLL_PAIR[0,1]="PA_TOOL"
```

```
$NP_SAFETY_DIST[0]=5
$NP_COLL_PAIR[1,0]="PA_BASE"
$NP_COLL_PAIR[1,1]="PA_Z"
$NP_SAFETY_DIST[1]=5
$NP_COLL_PAIR[2,0]="PA_TABLE"
$NP_COLL_PAIR[2,1]="PA_TOOL"
$NP_SAFETY_DIST[2]=5
$NP_COLL_PAIR[3,0]="PA_TABLE"
$NP_COLL_PAIR[3,1]="PA_Z"
$NP_SAFETY_DIST[3]=5
```

M30

7 参考文献

1. 《SINUMERIK 840D sl / 828D Special functions》2018.08

8 作者/联系人

Cheng Fei
2019.07.10

9 版本信息

版本	日期	修改内容
V1.0	2019.07.10	第一版
V1.1	2019.08.29	模板更新