

编程说明 版本 10.2004

# sinumerik

SINUMERIK 840D/840Di/810D  
循环

**SIEMENS**



# SIEMENS

## SINUMERIK 840D/840Di/810D

### 循环

#### 编程说明

#### 适用于

##### *控制系统*

SINUMERIK 840D powerline  
SINUMERIK 840DE powerline (出口版本)  
SINUMERIK 840Di  
SINUMERIK 840DiE (出口版本)  
SINUMERIK 810D powerline  
SINUMERIK 810DE powerline (出口版本)

##### *驱动*

SIMODRIVE 611 D

##### *软件*

循环

##### *版本*

6

版本 10.04

概论	1
钻削循环和钻削图	2
铣削循环	3
车削循环	4
故障信息和故障处理	5
附录	A

## SINUMERIK® —— 文献

### 版本说明

以下是当前版本及以前各版本的简要说明。

每个版本的状态由“附注”栏中的代码指明。

在“附注”栏中的状态码分别表示：

- A** …… 新文件
- B** …… 没有改动，但以新的订货号重印
- C** …… 有改动，并重新发行

若某页的内容在上一个版本后有实质性的更改，则在该页的顶部用新版本号来指明。

版本	订货号	附注
95年2月	6FC5298-2AB40-0AP0	A
95年4月	6FC5298-2AB40-0AP1	C
96年3月	6FC5298-3AB40-0AP0	C
97年8月	6FC5298-4AB40-0AP0	C
97年12月	6FC5298-4AB40-0AP1	C
98年12月	6FC5298-5AB40-0AP0	C
99年8月	6FC5298-5AB40-0AP1	C
2000年4月	6FC5298-5AB40-0AP2	C
2000年10月	6FC5298-6AB40-0AP0	C
2001年9月	6FC5298-6AB40-0AP1	C
2000年11月	6FC5298-6AB40-0AP2	C
2004年3月	<b>6FC5298-7AB40-0RP0</b>	C
2004年10月	<b>6FC5298-7AB40-0RP1</b>	C

### 注册商标

SIMATIC®、SIMATIC HMI®、SIMATIC NET®、SIROTEC®、SINUMERIK® 和 SIMODRIVE® 均为西门子公司的注册商标。印刷物中的其它符号可能是一些其它商标，任何第三方将其用于其它目的都会损坏所有者的权益。

其它信息可以上网查找：  
<http://www.siemens.com>

这些文献资料都是用 WinWord V 9.0  
和 Designer V 7.0 编写的。

可能是这些资料中未描述的控制装置中的其它一些可执行功能。在新提供的资料或者售后服务中还没有有关这些功能的描述。

我们已对印刷品中所述内容与硬件和软件的一致性作过检查。然而不排除存在偏差的可能性，因此我们不保证印刷品中所述内容与硬件和软件完全一致。文献中的有关信息会定期审核，而且一些必要的修改会包含在下一个版本中。您提出的每一条建议我们都将衷心感谢。

© 西门子股份公司 1995 – 2004。所有权所有。

保留技术变更权利。

## 前言

### 资料结构

SINUMERIK资料分为3种类型:

- 一般文献
- 用户文献
- 制造商/维修文献

### 读者对象

该资料面向机床用户。印刷物中详细描述了用户操作控制系统SINUMERIK 810D和840D 的须知。

### 标准功能范畴

在该编程说明中描述了标准的功能范畴（循环SW6.5）。机床制造商增添或者更改的功能，由机床制造商资料进行说明。

有关SINUMERIK810D和840D其它的印刷资料，以及适用于所有SINUMERIK控制系统的其它文献（比如通用接口，测量循环...），您可以从西门子办事处获取。

控制系统有可能执行本文献中未描述的某些功能。但是这并不意味着在提供系统时必须带有这些功能，或者为其提供有关的维修服务。

### 适用性

该编程说明适用于循环版本6.5

## SINUMERIK 840D powerline

自 2001 年 9 月起

- SINUMERIK 840D powerline 和
  - SINUMERIK 840DE powerline
- 将提供更好的性能。可提供的 powerline—模块列表您可在硬件描述 /PHD/ 章节 1.1 中找到

## SINUMERIK 810D powerline

自 2001 年 12 月起

- SINUMERIK 810D powerline 和
  - SINUMERIK 810DE powerline
- 将提供更好的性能。可提供的 powerline 模块列表您可在硬件描述 /PHC/ 章节 1.1 中找到

## 资料编排结构

所有的循环和编程方法—只要可能—均按照相同的内部结构进行描述。通过划分为不同的级别，您可以很方便地找到所需要的信息。

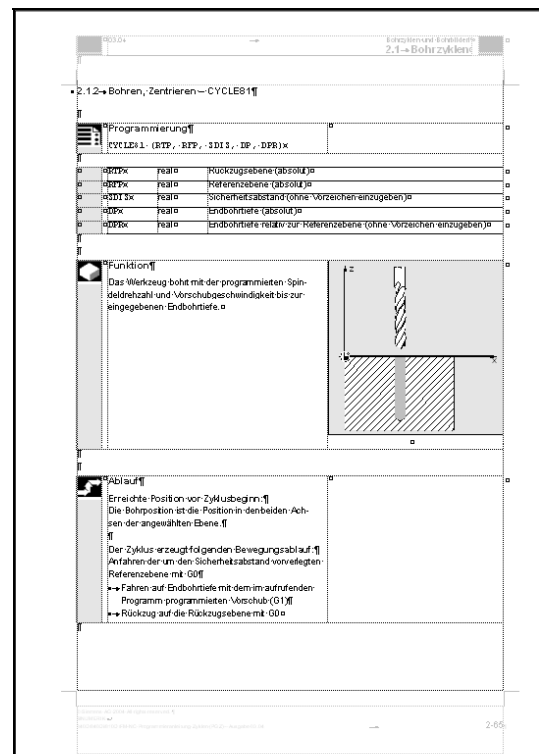
### 1. 快速一览

如果您要查找较少使用的循环或者想了解一个参数的含义，您可以看一下快速一览，了解其如何编程功能，并找到这些循环和参数的说明。

这些信息总是位于一页的顶部。

说明：

由于篇幅的限制，对各个循环和参数不可能用编程语言给出所有说明方式。这里所介绍的循环编程方法仅仅是在车间现场最经常使用的一种。



## 2. 详细说明

在理论部分，您可以找到下列问题的详细说明：



这些循环的作用如何？



循环有何效果？



如何执行？

这些参数有何作用？

还需要特别注意什么？

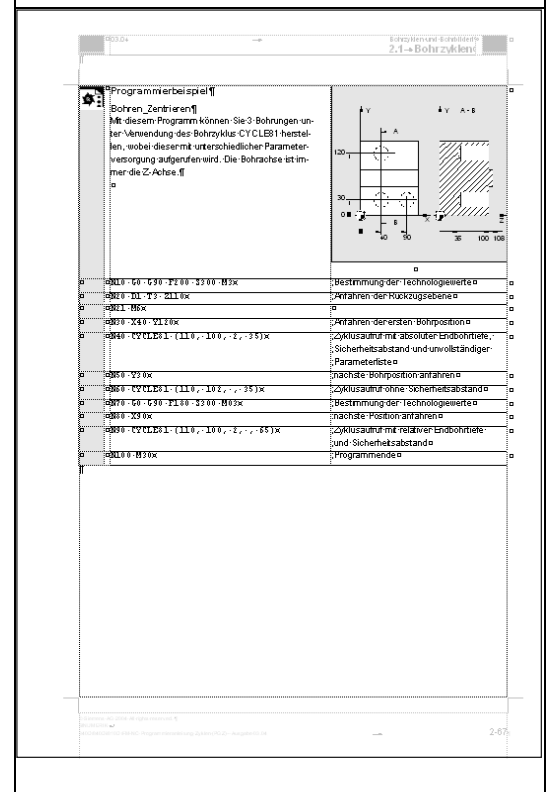
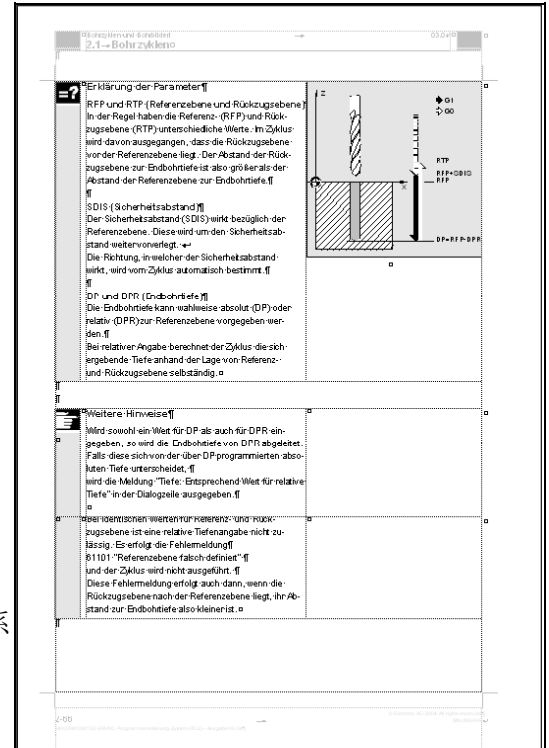
对于数控入门者来说，理论部分可以用作学习材料。至少要把此手册通读一遍，这样可以对SINUMERIK控制系统的功能范围和性能有一个初步的了解。

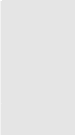






## 3. 从理论到实践

您如何把这些循环应用到具体的场合，请参见编程示例。

在理论部分之后所有循环均有一个应用示例。



 符号说明 工作流程 说明 功能 参数 编程举例 编程 其它说明 参见其它文献和章节 对危险或者故障产生的说明 附加的说明或者背景信息



## 警示符号

在本资料中使用下面不同的警示符号表示需要以不同的等级关注。



## 危险

该警示符号表示如果不采取相应的小心措施，**将会**导致死亡或者严重的人身伤害和巨大的财产损失。



## 警告

该警示符号表示如果不按照相应规定小心进行，**可能**将会导致死亡或者严重的人身伤害和巨大的财产损失。



## 小心

该警示符号（带有警告三角）表示如果不采取相应的小心措施，**可能**导致轻微的人身伤害。

## 小心

该警示符号（不带警告三角）表示如果不采取相应的小心措施，**可能**导致财产损失。

## 注意

该警示符号表明如果不注意相应的提示，则**可能**会引起不好的结果或状态。

## 准则

## 附加设备

西门子的控制系统可以根据不同的应用场合，利用西门子提供的附加仪器、附加设备及扩建等级进行扩展应用。

## 人员

仅允许由**合格的、受过专门培训的专业人员**操作。没有受过培训的人员不可以操作系统，哪怕很短的时间。

必须明确地规定安装调试、操作及维护人员的职责，并且对他们的**职责遵守情况进行监控和检查**。

## 职能

在对控制装置进行开机调试**前**，应确保相关人员已阅读并已了解使用说明。此外，操作人员**有责任专心观察**运行的控制装置的技术整体状态（从外表观察运行性能的不足、损坏和更改）。

## 维修

维修工作只能由**相关专业的、受过培训的合格人员**进行，他们必须根据维修和维护手册的说明进行这些工作。在此，必须注意遵守相关的安全规范。

## 热线电话

有问题时请打以下热线电话：

A&D 技术支持 电话：+49 (0) 180 5050 – 222

传真：+49 (0) 180 5050 – 223

<http://www.siemens.de/automation/support-request>

资料方面有疑问时（建议，更正）请发传真或电子邮件至：

传真：+49 (0) 0131 98 – 2176

电子邮件：[motioncontrol.docu@erlf.siemens.de](mailto:motioncontrol.docu@erlf.siemens.de)

传真表格：参见印刷物末尾的反馈信息单。

## 因特网地址

<http://www.siemens.com/motioncontrol>

**提示**

以下行为被认为是不正确操作，因此生产厂家不承担责任：

**每种**与上面所述正确用法相违背的应用。

如果在**非正常状态**使用控制系统，或者不遵循安全规范、没有遵照使用说明中所作的操作要求而进行操作。

没有在系统的开机调试**之前**排除可能对安全造成隐患的故障。

在控制系统中**改变、跳转**或者**取消**一些设备，它们有助于正常功能的使用和安全性能的发挥。



不正常的使用有可能造成**不可预见的危险**，它们会对：

- 人身安全造成危害，
- 也可能对系统、机床和企业与用户的其它财产造成损害。

用于记录

## 目录

<b>概论</b>	<b>1-17</b>
1.1 一般提示	1-18
1.2 循环概述	1-18
1.2.1 钻削循环，孔组合循环，铣削循环和车削循环	1-19
1.2.2 循环辅助子程序	1-20
1.3 循环编程	1-21
1.3.1 调用条件和返回条件	1-21
1.3.2 加工循环时的信息	1-22
1.3.3 循环调用和参数表	1-23
1.3.4 循环仿真	1-26
1.4 程序编辑器中的循环支持	1-27
1.4.1 菜单，循环选择	1-27
1.4.2 输入屏幕的新功能	1-28
1.5 用户循环的循环支持	1-35
1.5.1 所需文件的概述	1-35
1.5.2 登入循环支持	1-35
1.5.3 设计循环支持	1-36
1.5.4 位图尺寸和屏幕分辨率	1-37
1.5.5 在 HMI Advanced 配置时，存储位图在文件维护中	1-38
1.5.6 用于 HMI Embedded 配置的位图处理	1-38
1.6 循环调试	1-40
1.6.1 机床数据	1-40
1.6.2 定义文件，用于循环 GUD7.DEF 和 SMAC.DEF	1-41
1.6.3 在 HMI Advanced 配置中，循环的供货形式	1-43
1.6.4 循环升级，自 SW 6.4 起；在 HMI Advanced 中自 SW 6.3 起	1-43
1.7 循环辅助功能	1-44
<b>钻削循环和钻削图！</b>	<b>2-47</b>
2.1 钻削循环	2-48
2.1.1 前提条件	2-50
2.1.2 钻削，定中心 – CYCLE81	2-51
2.1.3 钻削，镗平面 – CYCL82	2-54
2.1.4 深孔钻削 – CYCLE83	2-56
2.1.5 攻丝，不带补偿衬套 – CYCLE84	2-63
2.1.6 攻丝，带补偿衬套 – CYCLE840	2-70
2.1.7 镗孔 1 – CYCLE85	2-78
2.1.8 镗孔 2 – CYCLE86	2-81
2.1.9 镗孔 3 – CYCLE87	2-85
2.1.10 镗孔 4 – CYCLE88	2-87
2.1.11 镗孔 5 – CYCLE89	2-89
2.2 模态调用钻削循环	2-91

2.3	钻削图循环 .....	2-94
2.3.1	前提条件 .....	2-94
2.3.2	成排孔- HOLES1.....	2-95
2.3.3	孔圆弧 - HOLES2.....	2-99
2.3.4	点栅格 - CYCLE801 .....	2-102
<b>铣削循环</b>		<b>3-105</b>
3.1	一般提示 .....	3-107
3.2	前提条件 .....	3-108
3.3	螺纹铣削CYCLE90.....	3-110
3.4	一个圆弧上的长方形孔 - LONGHOLE .....	3-117
3.5	一个圆弧上的键槽 - SLOT1 .....	3-122
3.6	环形槽 - SLOT2.....	3-130
3.7	铣削矩形槽 - POCKET1.....	3-136
3.8	铣削环形凹槽 - POCKET2 .....	3-140
3.9	铣削矩形凹槽 - POCKET3 .....	3-144
3.10	铣削环形凹槽 - POCKET4.....	3-153
3.11	平面铣削 - CYCLE71.....	3-158
3.12	轨迹铣削 - CYCLE72.....	3-164
3.13	铣削矩形轴颈- CYCLE76 .....	3-174
3.14	铣削环形轴颈 - CYCLE77 .....	3-179
3.15	带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75 .....	3-183
3.15.1	传送凹槽边缘轮廓- CYCLE74 .....	3-184
3.15.2	传送中心岛轮廓- CYCLE75.....	3-186
3.15.3	轮廓编程 .....	3-187
3.15.4	带中心岛的凹槽铣削 - CYCLE73.....	3-189
3.16	回转- CYCLE800.....	3-211
3.16.1	操作、参数和输入屏幕 .....	3-213
3.16.2	操作说明，编程说明 .....	3-217
3.16.3	参数 .....	3-218
3.16.4	设置刀具（自循环SW 6.5 起） .....	3-222
3.16.5	调试带有回转加工平面的工件（自循环 SW 6.5 起） .....	3-223
3.16.6	操作，参数“在 JOG 方式下回转”（自循环 SW 6.5 起） .....	3-227
3.16.7	回转循环CYCLE800 开机调试 .....	3-230
3.16.8	用户循环 TOOLCARR.SPF.....	3-251
3.16.9	故障信息 .....	3-255
3.17	高速设定 - CYCLE832（自循环 SW 6.3 起） .....	3-257
3.17.1	在 HMI 菜单树中调用 CYCLE832 .....	3-260
3.17.2	参数 .....	3-263
3.17.3	匹配工艺 .....	3-264

3.17.4	匹配附加的程序参数CYC_832T .....	3-266
3.17.5	接口 .....	3-268
3.17.6	故障信息 .....	3-269
3.18	模腔循环CYCLE60 (自循环 SW 6.4 起) .....	3-270
<b>车削循环</b>		<b>4-281</b>
4.1	一般提示 .....	4-282
4.2	前提条件 .....	4-283
4.3	切槽循环- CYCLE93 .....	4-286
4.4	退刀槽循环- CYCLE94 .....	4-295
4.5	切削循环- CYCLE95 .....	4-299
4.6	螺纹退刀槽 - CYCLE96 .....	4-312
4.7	螺纹切削 - CYCLE97 .....	4-316
4.8	螺纹链 - CYCLE98 .....	4-323
4.9	螺纹精整 .....	4-329
4.10	扩展的切削循环 - CYCLE950 .....	4-331
<b>故障信息和故障消除</b>		<b>5-353</b>
5.1	一般提示 .....	5-354
5.2	循环中的故障处理 .....	5-354
5.3	循环报警一览 .....	5-355
5.4	循环中的信息 .....	5-361
<b>附录</b>		<b>A-363</b>
A	缩略符 .....	A-364
B	术语 .....	A-373
C	参考文献 .....	A-393
D	名称 .....	A-405
E	索引 .....	I-409
F	指令/名称 .....	I-412

用于记录



## 概念

1.1	一般提示 .....	1-18
1.2	循环概述 .....	1-18
1.2.1	钻削循环，孔组合循环，铣削循环和车削循环 .....	1-19
1.2.2	循环辅助子程序 .....	1-20
1.3	循环编程 .....	1-21
1.3.1	调用条件和返回条件 .....	1-21
1.3.2	加工循环时的信息 .....	1-22
1.3.3	循环调用和参数表 .....	1-23
1.3.4	循环仿真 .....	1-26
1.4	程序编辑器中的循环支持 .....	1-27
1.4.1	菜单，循环选择 .....	1-27
1.4.2	输入屏幕的新功能 .....	1-28
1.5	用户循环的循环支持 .....	1-35
1.5.1	所需文件的概述 .....	1-35
1.5.2	登入循环支持 .....	1-35
1.5.3	设计循环支持 .....	1-36
1.5.4	位图尺寸和屏幕分辨率 .....	1-37
1.5.5	在 HMI Advanced 配置时，存储位图在文件维护中 .....	1-38
1.5.6	用于 HMI Embedded 配置的位图处理 .....	1-38
1.6	循环调试 .....	1-40
1.6.1	机床数据 .....	1-40
1.6.2	定义文件，用于循环 GUD7.DEF 和 SMAC.DEF .....	1-41
1.6.3	在 HMI Advanced 配置中，循环的供货形式 .....	1-43
1.6.4	循环升级，自 SW 6.4 起；在 HMI Advanced 中自 SW 6.3 起 .....	1-43
1.7	循环辅助功能 .....	1-44

## 1.1 一般提示

第一章对现有供使用的循环进行了概述。在下面的章节中说明了所有循环均普遍适用的条件，它们关于：

- 循环编程，和
- 循环调用的用户引导。

## 1.2 循环概述

循环是一种工艺子程序，借助这些循环可有效实现特定的加工过程，如钻削螺纹或铣削凹槽。通过所提供的参数将循环匹配到具体的问题位置上。

在系统中，根据具体的工艺要求：

- 钻削
- 铣削
- 车削

提供各不相同的标准循环。

### 1.2.1 钻削循环，孔组合循环，铣削循环和车削循环

控制系统 SINUMERIK 810D, 840D 和 840Di  
提供以下循环：

#### 钻削循环

CYCLE81	钻削，定中心
CYCLE82	钻孔，镗平面
CYCLE83	深孔钻削
CYCLE84	攻丝，不带补偿衬套
CYCLE840	攻丝，带补偿夹具
CYCLE85	镗孔1
CYCLE86	镗孔2
CYCLE87	镗孔3
CYCLE88	镗孔4
CYCLE89	镗孔5

#### 孔组合循环

HOLES1	成行孔加工
HOLES2	成圆弧状孔加工
CYCLE801	点栅格

#### 铣削循环

LONGHOLE	长方形孔铣削图，在一个圆弧上
SLOT1	键槽铣削图，在一个圆弧上
SLOT2	圆弧键槽铣削图
POCKET1	铣削矩形槽（用面铣刀）
POCKET2	铣削弧形槽（用面铣刀）
CYCLE90	铣螺纹
POCKET3	铣削矩形槽（用任意铣刀）
POCKET4	铣削弧形槽（用任意铣刀）
CYCLE71	平面铣削
CYCLE72	轮廓铣削
CYCLE73	带中心岛的凹槽铣削
CYCLE74	槽边轮廓传送
CYCLE75	小岛轮廓传送
CYCLE76	铣削矩形轴颈
CYCLE77	铣削圆弧轴颈

### 旋转循环

CYCLE93	切槽
CYCLE94	退刀槽（形状E和F，符合DIN）
CYCLE95	毛坯切削，带底切
CYCLE96	螺纹退刀槽切削（形状A, B, C和D符合DIN）
CYCLE97	螺纹切削
CYCLE98	螺纹级联
CYCLE950	扩展的毛坯切削

### 1.2.2 循环辅助子程序

属于循环软件包的辅助子程序有：

- STEIGUNG和
- MELDUNG。

它们必须始终装载到控制系统中。

## 1.3 循环编程

标准循环定义为带有名称和参数表的子程序。

“SINUMERIK 编程说明第一部分：基本原理”中所描述的操作适用于调用循环。



循环将存储在软盘上或针对 HMI Advanced 用各自的软件版本来提供。通过 V.24 接口将这些循环装载至控制系统的零件程序存储器中（参见操作说明）。

### 1.3.1 调用条件和返回条件

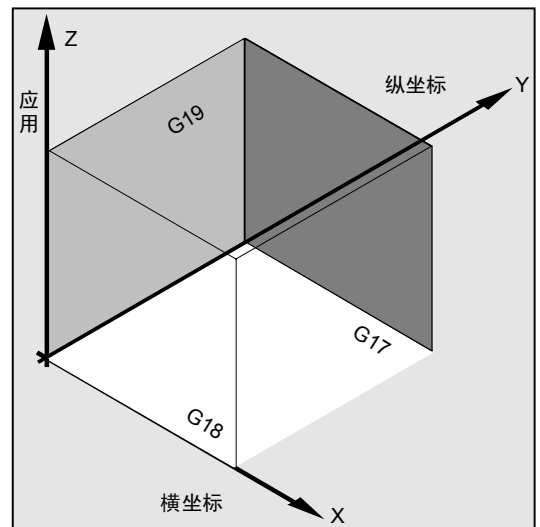
在循环调用之前有效的G功能和可编程的框架可以在循环之后仍然保持。

在循环调用之前定义加工平面

(G17, G18, G19)。循环工作于当前平面中,其具有

- 横坐标（第一个几何轴）
- 纵坐标（第二个几何轴）
- 应用（第三个几何轴，用于空间平面）

钻削循环时，在相对应于当前平面应用轴的轴向上进行钻削。铣削时，在该轴向进行深度进刀。



#### 平面和轴分配

指令	平面	垂直的进刀轴
G17	X/Y	Z
G18	Z/X	Y
G19	Y/Z	X

### 1.3.2 加工循环时的信息

在一些循环中，在加工过程中，会在屏幕上显示加工的信息。

这些信息不中断程序执行，并且一直保持直至下一个信息的出现。

信息文本和含义在相应循环中描述。



所有重要信息的综述可以在该编程说明的附录中获得。

#### 在执行一个循环时程序段显示

在整个循环运行过程中，循环调用始终位于当前的程序段显示中。

### 1.3.3 循环调用和参数表

标准循环按照用户定义的变量进行加工。用于循环的供给参数您可以通过循环调用时的参数表进行传送。



循环调用始终要求一个独立的程序段。

#### 标准循环中参数的基本说明

编程说明说明了每个循环的参数表，具有

- 顺序和
- 类型。

这些参数的顺序必须无条件遵守。

每个用于循环的供给参数都有一个特定的数据类型。循环调用时，必须注意这些当前所用参数的类型。在参数表中可

- 传送变量，或者
- 常量。

如果在参数表中传送变量，则必须事先在调用的程序定义这些变量，并进行赋值。这些循环可以

- 用一个不完整的参数表或者
- 忽略参数

进行调用。

如果您想要删除调用时写入的上一传输参数，可用符号“)”提前结束参数表。如果您想要在此之间省略参数，则写入逗号“..., ...”作为通配符。



带离散的或者限制范围的参数，不进行参数值的奇偶性校验，除非在一个循环中明确说明一个故障的反应。

如果在循环调用时，参数表中的登记比循环中定义的参数多，则显示NC报警12340“参数值太大”，并且不执行该循环。

### 循环调用

在下面通过举例循环CYCLE100，说明不同的方法用于写一个循环调用，它要求以下的输入参数。

#### 举例

FORM	定义待加工的形状 值：E 和F
MID	进刀深度（不输入符号）
FFR	进给率
VARI	加工方式 值：0, 1 或 2
FAL	精加工余量

循环通过指令CYCLE100 (FORM, MID, FFR, VARI, FAL) 调用。

#### 1. 参数表，带恒定值

在各个参数中，您可以直接输入执行该循环所用到的具体值。

#### 举例

CYCLE100 ("E", 5, 0.1, 1, 0) ; 循环调用

#### 2. 带变量的参数表作为传送参数

您可以作为计算变量传送这些参数，在调用该循环之前您必须定义这些参数并提供数值。



**举例**

DEF CHAR FORM="E"	; 一个参数的定义, 赋值
DEF REAL MID=5, FFR, FAL	; 参数的定义, 带和不带赋值
DEF INT VARI=1	
N10 FFR=0.1 FAL=0	; 赋值
N20 CYCLE100 (FORM, MID, FFR, -> -> VARI, FAL)	; 循环调用

**3. 使用预定义的变量, 作为传送参数**

您也可以使用变量, 如R参数,  
用于给循环提供参数。

**举例**

DEF CHAR FORM="E"	; 一个参数的定义, 赋值
N10 R1=5 R2=0.1 R3=1 R4=0	; 赋值
N20 CYCLE100 (FORM, R1, -> -> R2, R3, R4)	; 循环调用

因为R参数已经用实数型预定义, 所以必须注意所提供的目标参数和该实数型之间类型的一致性。



在编程说明中进一步说明数据类型和类型转换以及类型一致性。如果结果为类型不一致, 则系统报警12330“参数类型...错”。

**4. 不完整的参数表和忽略参数**

如果不需要用于循环调用的供给参数或该参数值应为零, 则可在参数表中省去该参数。在这种情况下, 仅写入逗号“..., ...”, 以确保正确分配随后的参数, 或提前用“)”关闭参数表。

**举例**

```
("F", 3, 0.3, CYCLE100, 1)
```

; 循环调用, 省略第4个参数  
(也就是说值零)

```
CYCLE100 ("F", 3, 0.3)
```

; 循环调用, 最后的两个参数分配值零  
(也就是说它们被省略)。

**5. 参数表中的表达式**

在参数表中也允许使用表达式,  
其结果被赋值到循环中相应的参数。

**举例**

```
MID DEF REAL =7, FFR=200
```

; 参数定义, 赋值

```
CYCLE100 ("E", MID*0.5, FFR+100,1)
```

; 循环调用

; 进刀深度3.5, 进给300

**1.3.4 循环仿真**

带有循环调用的程序可以首先通过仿真测试。

**功能**

在HMIEmbedded (内置) 这种配置中, 在仿真时程序如同正常加工一样执行, 在屏幕上并且同时给出运行图形。

用 HMI Advanced 进行配置时, 单独在HMI中结束`程序仿真。由此也可以进行不带刀具数据的循环或事先选择刀具补偿。

这样, 在刀具补偿参数必须算入位移运行之内的循环 (比如铣削凹槽和键槽, 车槽) 中运行到最后轮廓, 给出信息: “仿真有效, 不带刀具”。

这种功能可以应用于控制凹槽位置的场合。

## 1.4 程序编辑器中的循环支持

程序编辑器提供一个扩展的循环辅助，用于西门子循环和用户循环。



### 功能

循环辅助提供以下的功能：

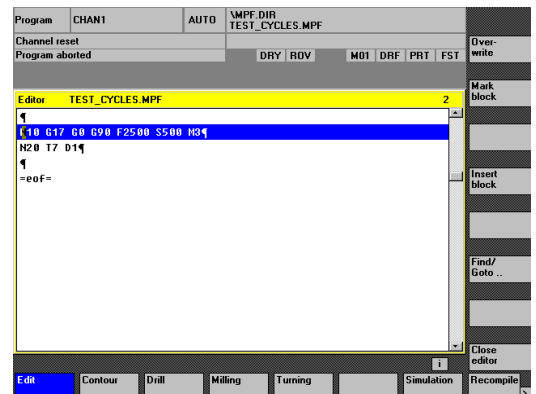
- 通过软键选择循环
- 输入屏幕，用于参数输入并带辅助图形
- 每个参数的在线帮助（仅限于HMI Advanced配置）
- 轮廓输入辅助

由各个表征码产生程序代码，它们可以返回翻译。

### 1.4.1 菜单，循环选择



循环的选择可以通过软键，按照不同的工艺要求进行：



轮廓

通过几何处理器或者轮廓表征码输入几何参数

钻削

输入屏幕，用于钻削循环和钻削图。

铣削

输入屏幕，用于铣削循环。

车削

输入屏幕，用于车削循环。

在按确认键结束输入屏幕中的输入后，工艺选择工具条保持显示。

相似的循环显示在同一个输入屏幕中。这样在该屏幕中可以通过软键进行循环之间的转换，比如在攻丝时或者

在退刀槽加工时。

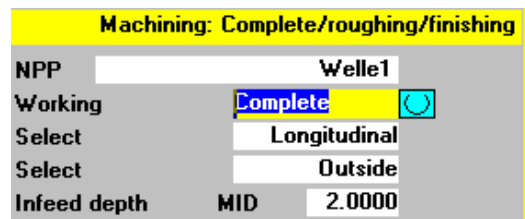
编辑器中的循环辅助也包含一些屏幕窗口，它们没有循环调用，而是插入几行未占用的DIN代码到程序中，比如轮廓输入屏幕以及输入任意钻削位置。

## 1.4.2 输入屏幕的新功能

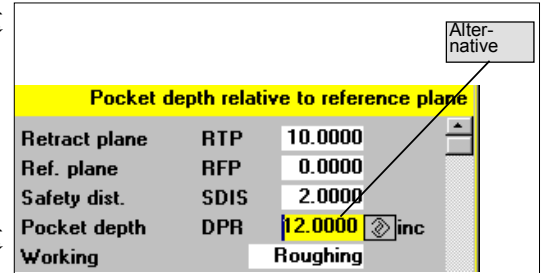


### 功能


- 在许多循环中，加工方式可由参数 VARI 决定。它通常包含多个设置，这些设置被编码为一个值。在新的循环支持屏幕中各个设置被分配在不同的输入栏中，这些输入栏可以通过切换键切换。
- 输入屏幕相应进行动态变化。总是只显示用于所选加工方式必需的输入栏，不需要的输入栏不会显示。例如，在精加工时，这情况适用于进给参数。
- 如果有必要，自动由一个输入来分配相互有关的参数。这情况在螺纹加工时出现，此时支持当前的公制螺纹表格。例如，对于螺纹切削循环 CYCLE97，通过输入12至螺纹尺寸（参数MPIT）输入栏中，自动分配螺距（参数PIT）为 1.75，螺纹深度（参数TDEP）为 1.137。如果没有选择公制螺纹表格，该功能不生效。
- 如果一个输入屏幕窗口显示第二次，则所有输入区均以最后输入的值预置。  
在一些循环中，如果它们在同一个程序中接连调用多次（比如用于粗加工和精加工的凹槽铣削），则仅需要修改几个参数。

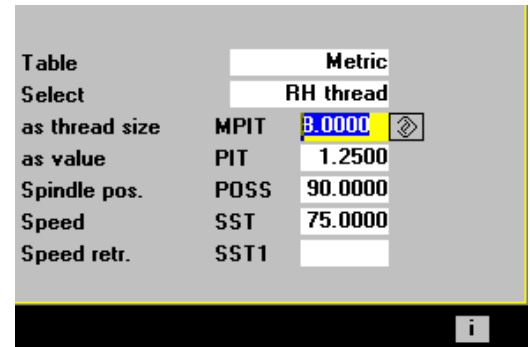


- 在钻削和铣削循环屏幕中可以存在一些特定参数，这些参数可作为绝对值或者相对值输入。对于这些参数，在输入栏后显示缩写 ABS（绝对值输入）或者缩写 INK（相对值输入）。这可通过软键“Alternativ（选择）”进行切换。在下次调用该屏幕时，同样保持这些设置。

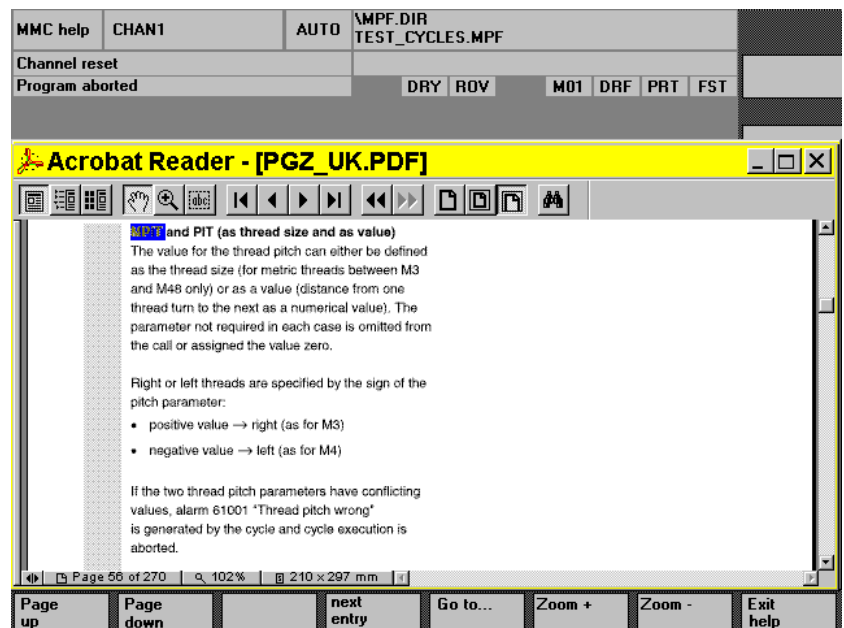


- 在HMI Advanced中可以通过在线帮助显示有关各个循环参数的附加信息。
- 将光标停留在某个参数上，将在右下处显示帮助图

标 ，可以激活帮助功能。



按一下信息键，打开并显示循环编程说明中有关该参数的说明。





## 帮助画面的操作

- |       |                 |
|-------|-----------------|
| 返回前页  | 在文件中返回到上一页。     |
| 向后翻页  | 在文件中翻到下一页。      |
| 下一个条目 | 跳转到帮助中预先规定的文本处。 |
| 转到    | 跳转到所选择的文本处。     |
| 放大    | 在帮助窗口中放大字体。     |
| 缩小    | 在帮助窗口中缩小字体。     |
| 退出帮助  | 返回到循环屏幕窗口。      |



## 轮廓输入支持

### 自由的轮廓编程

创建轮廓

启动自由轮廓编程，使用它可以输入相互关联的轮廓段（参见参考文献：/BA/，章节）。

### 轮廓编程

轮廓  
1-直线

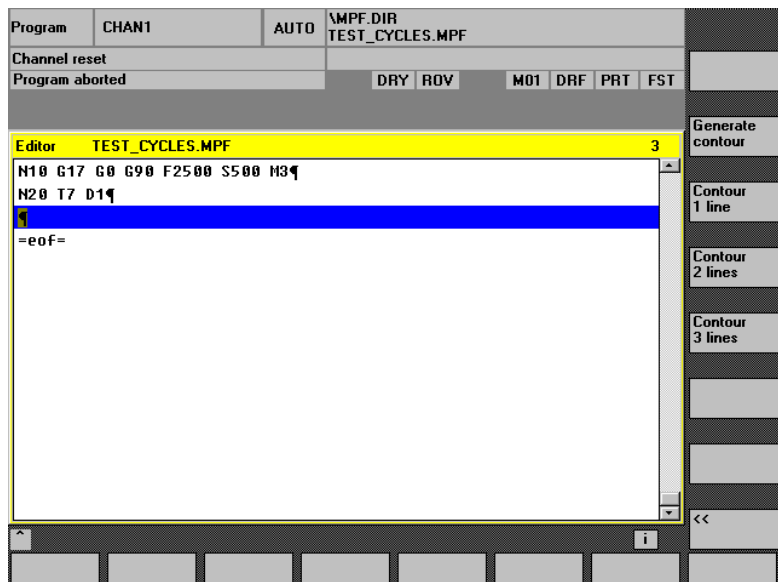
轮廓  
2-直线

轮廓  
3-直线

这些软键支持可能的轮廓段。

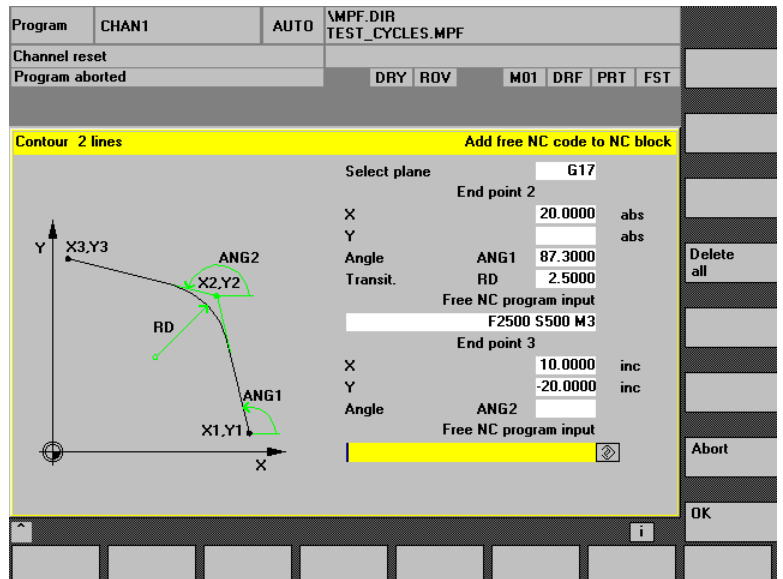
这些轮廓由一条或者多条直线以及中间的轮廓过渡元素（半径，倒角）组成。

可以通过终点或者点和角规定每个轮廓元素，并通过未占用的 DIN 代码进行补充。



### 举例

在下面用于2一直线的轮廓段的输入屏幕窗口中产生以下的DIN代码:



X=AC (20) ANG=87.3 RND=2.5 F2000 S500 M3  
X=IC (10) Y=IC (-20); 终点, 增量式



### 钻削支持

钻削支持包含钻削循环和钻削图的选择。

钻削定中心

深孔钻削

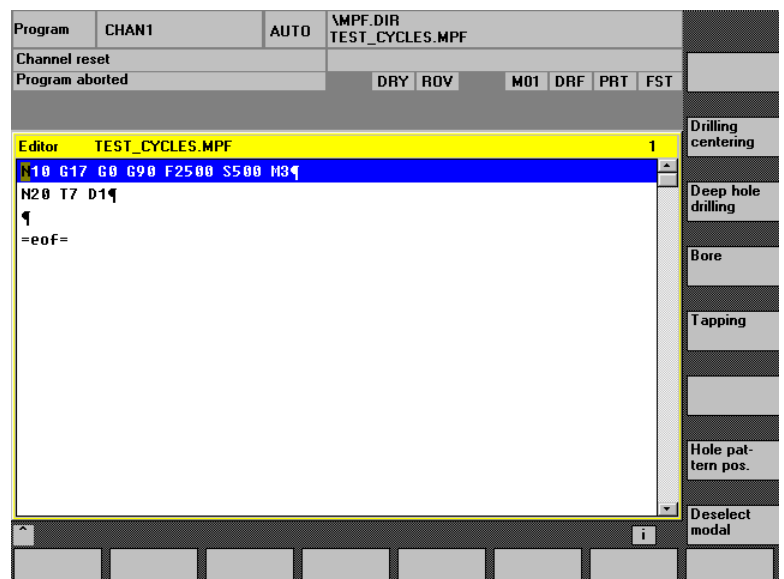
镗孔

攻丝

钻削图选择

钻削图位置

模式选择



循环 CYCLE81, CYCLE87 和 CYCLE89

不能用此支持编程参数。

CYCLE81 功能由

CYCLE82 (软键“钻削定中心”)一并覆盖,

CYCLE89 功能同样如此。CYCLE87 功能由

CYCLE88 功能覆盖

(软键“钻削定中心” → “钻削一并停止”)。

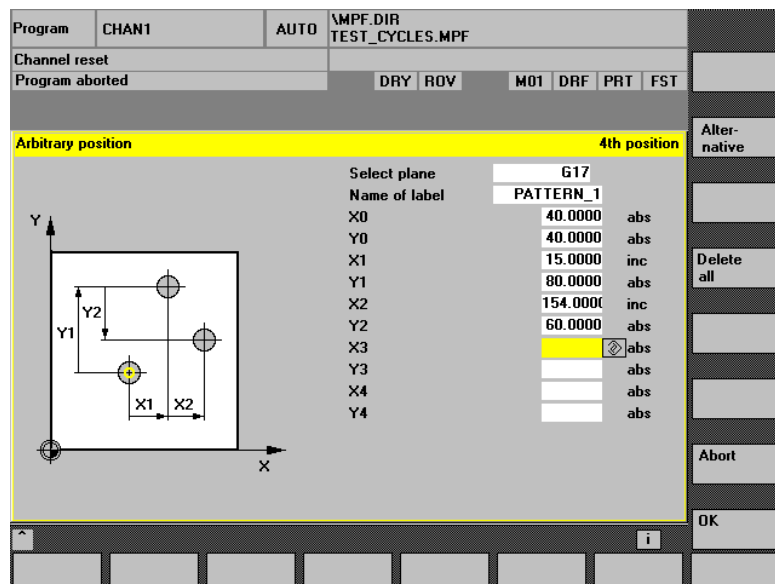
钻削图可以重复，例如要进行相邻钻削和攻丝时。为此在钻削图中规定一个钻削图名称，该名称随后输入在屏幕“重复定位”中。



### 循环支持的编程举例

N100 G17 G0 G90 Z20 F2000 S500 M3	; 主程序段
N110 T7 M6	; 换钻头
N120 G0 G90 X50 Y50	; 出发位置钻削
N130 MCALL CYCLE82(10,0,2,0,30,5)	; 模态钻削循环调用
N140 孔圆弧1:	; 标签—钻削图名称
N150 HOLES2(50,50,37,20,20,9)	; 调用钻削图循环
N160 ENDLABEL:	
N170 MCALL	; 撤消选择模态调用
N180 T8 M6	; 换丝锥
N190 S400 M3	
N200 MCALL	; 模态调用攻丝循环
CYCLE84(10,0,2,0,30,,3,5,0.8,180,300,500)	
N210 REPEAT 孔圆弧1	; 重复钻削图
N220 MCALL	; 撤消选择模态调用

此外，可以通过一个屏幕窗口把任意钻削位置作为一个可以重复的钻削模型输入。



在该平面中最多可以编程5个位置，所有值均可以选择绝对值或者相对值（用软键“Alternativ（选择）”（可转换）。通过软键“全部删除”产生一个空的屏幕窗口。



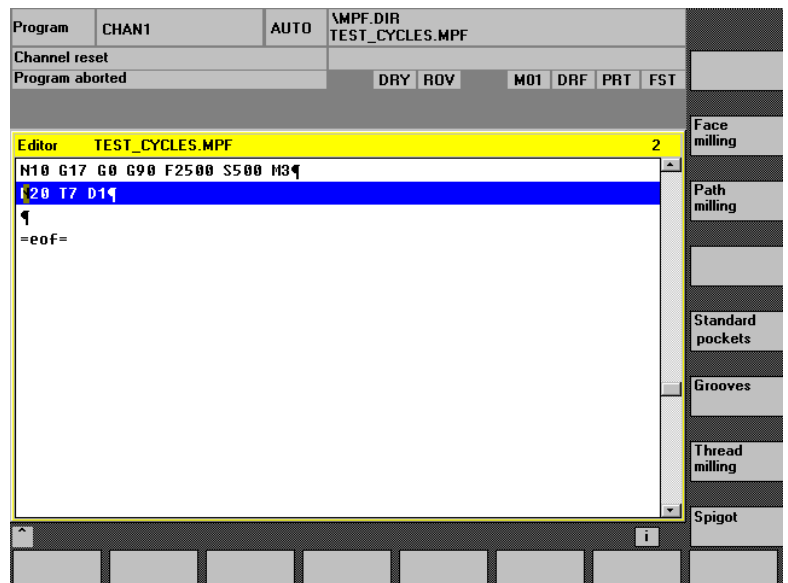


## 铣削支持

铣削支持具有以下选择方式：



如果在子菜单中选择几个凹槽铣削循环、键槽铣削循环或者轴颈铣削循环，则按软键“标准槽”、“键槽”和“轴颈”无效，拒绝接收。

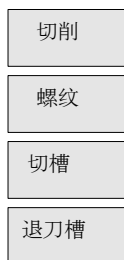


凹槽铣削循环POCKET1和POCKET2不可以通过该支持进行参数设定。



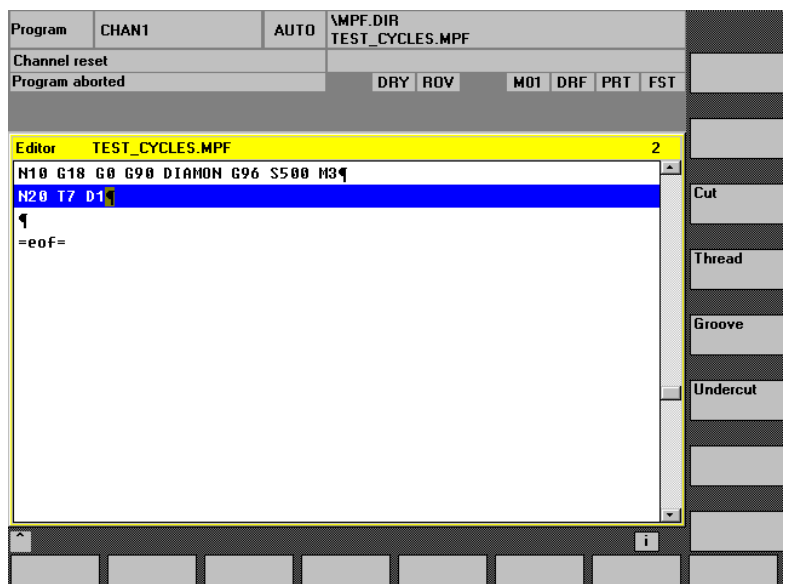
## 车削支持

车削支持具有以下选择方式：



退刀槽循环形状E和F

(CYCLE94) 以及螺纹退刀槽形状A到D (CYCLE96) 集合在软键“退刀槽”之下。



软键“螺纹”包含一个子菜单，用于选择简单的螺纹切削，或者是螺纹级联。



### 反编译

程序代码的反编译用于借助循环支持在一个已有的程序中进行更改。光标定位在要更改的行上并按下软键“反编译”。

这样就可以再次打开生成该程序的、相应的输入屏幕窗口，并且可以修改值。

如果直接在建立的 DIN 代码中进行更改，可能导致反编译无法再进行。因此，总是应该一直用循环支持进行处理，并借助于反编译进行更改。



### 设计用户循环支持



文献： /IAM/, 安装调试说明 HMI

BE1 “操作界面补充”

IM2 “HMI Embedded（内置）的开机调试”

IM4 “HMI Advanced（高级）的开机调试”

## 1.5 用户循环的循环支持

### 1.5.1 所需文件的概述

以下的文件为循环支持的基础：

分配	文件	应用	文件类型
循环选择	aeditor.com	标准循环和用户循环	文本文件
	common.com (仅在 HMI Embedded 配置中)	标准循环和用户循环	文本文件
用于输入参数的屏幕 窗口	*.com	标准循环或者用户循 环	文本文件
辅助图	*.bmp	标准循环或者用户循 环	位图
在线帮助 (仅在 HMI Advanced 配置中)	pgz_<语言>.pdf 和 pgz_<语言>.txt	仅在标准循环中	pdf-文件



循环支持 (\*.com) 的设计文件，其名称可以自由选择。

### 1.5.2 登入循环支持



#### 功能

在程序编辑器中设计水平软键  
HS6作为用户循环的登入软键。  
必须在文件aeditor.com 中设计该键的功能。  
对此，给该软键赋值一个文本，并且在Press  
Block中设计一个功能用于此软键操作。

**举例：**

```
//S(Start)
```

```
...
```

```
HS5=($80270,,se1)
```

```
PRESS(HS5)
```

```
LS("Drehen",,1)
```

```
END_PRESS
```

```
HS6=("Usercycle" se1) ; HS6软键设计文本“用户循环”
```

```
PRESS(HS6)
```

```
LS("SK_Cycles1" ,"cycproj1" ) ; 在软键操作时，由文件cycproj1.com装载一个软键工  
具条
```

```
END_PRESS
```

详细的设计说明参见：

文献： **/IAM/, 安装调试说明 HMI**  
BE1 “操作界面补充”

在HMI Embedded（内置）配置时，此外必须在文件common.com中进行以下登录，用于激活该软键：

```

%_N_COMMON_COM
; $PATH=/_N_CUS_DIR
...
[MMC_DOS]
...
SC315=AEDITOR.COM
SC316=AEDITOR.COM

```

### 1.5.3 设计循环支持

#### 功能

软键工具条和循环辅助输入屏幕窗口可以在任意的文件中设计，并且作为\*.com类型存储在此控制系统中的HMI中。

详细的设计说明参见：

文献： **/IAM/, 安装调试说明 HMI**  
BE1 “操作界面补充”

在HMI Advanced（高级）配置中，\*.com文件存储在目录：

- dh\cst.dir
- dh\cma.dir 或者
- dh\cus.dir

中，并且有通常的查找顺序：

cus.dir, cma.dir, cst.dir. 这些文件没有装载到NCU中。

对于 HMI Embedded，可将 \*.com 文件装载至 NCU 中（借助RS232 通过“服务”读入）。因为这些文件已占用 NC 存储器，最好将它们捆绑到 HMI 中。

为此必须将它们压缩并将它们捆绑到 HMI 版本的应用软件中。用于压缩的工具将在 \hmi\_emb\tools 下与标准循环软件一同提供。



### 编制步骤

- 在一台PC中，把文件arj.exe从目录\hmi\_emb\tools下复制到一个空目录下。
- 再把自身的设计文件\*.com复制到该目录下。
- 用命令 `arj a<目标文件名称>`  
<源文件名称>压缩每个单独的 com 文件。  
目标文件必须有扩展co\_。  
举例：设计文件cycproj1.com压缩到：  
`arj a cycproj1.co_ cycproj1.com.`
- 把文件\*.co\_ 复制到HMI应用软件相应的目录下，并且产生一个位置。



文献： /BEM/, 操作说明 HMI Embedded  
/IAM/, 安装调试说明 HMI  
IM2 “HMI mbedded（内置）的开机调试”

#### 1.5.4 位图尺寸和屏幕分辨率

软件版本SW6.2中，其HMI有三种不同的屏幕分辨率。对于每种分辨率规定一个最大的循环屏幕窗口的位图尺寸（参见下表），在构造自身的位图时必须要注意。

屏幕分辨率	位图尺寸
640 * 480	224* 224 像素
800 * 600	280* 280 像素
1024 * 768	352* 352 像素

这些位图作为16色的位图构造并保存。

### 1.5.5 在HMI Advanced配置时，存储位图在文件维护中

对于不同的屏幕分辨率，在文件维护中（自HMI6.2起）设置进入文件维护的新的路径，这样这些位图就可以以不同的尺寸并行地进行存储。

标准循环：

- dh\cst.dir\hlp.dir\640.dir
- dh\cst.dir\hlp.dir\800.dir
- dh\cst.dir\hlp.dir\1024.dir

制造商循环：

- dh\cma.dir\hlp.dir\640.dir
- dh\cma.dir\hlp.dir\800.dir
- dh\cma.dir\hlp.dir\1024.dir

用户循环：

- dh\cus.dir\hlp.dir\640.dir
- dh\cus.dir\hlp.dir\800.dir
- dh\cus.dir\hlp.dir\1024.dir

根据当前的分辨率，首先在合适的目录中查找（比如在640 \* 480时为dh\...\hlp.dir\640.dir），然后在dh\...\hlp.dir中。否则适用的查找顺序为cus.dir, cma.dir, cst.dir。

### 1.5.6 用于 HMI Embedded 配置的位图处理



#### 功能

在 HMI Embedded 配置时，位图捆绑在 HMI 软件中。这些位图被汇编到一个软件包 **cst.arj** 中。原则上，位图可以以 \*.bmp 格式捆绑在该软件包中。二进制格式 \*.bin 在显示时可节省空间且更为快捷。

为了创建这种格式，需要在标准循环软件配置下的目录 \hmi\_emb\tools 中提供的工具：

- arj.exe, bmp2bin.exe, 和
- sys\_conv.col
- arj\_idx.exe（自 SW 6.3）

和原稿文件：

- mcst\_640.bat,
- mcst\_800.bat 或者
- mcst1024.bat.

文件 **cst.arj** 包含所有的标准循环位图和用户循环位图。在对其进行编制时，标准循环位图还必须与自身的位图捆绑在一起。



### 编制步骤

- 在一台PC中，把所有文件从目录\hmi\_emb\tools下复制到一个空目录下。
- 在这里编制一个子目录\bmp\_文件。
- 把自身的位图\*.bmp复制到该子目录\bmp\_文件中。
- 根据分辨率的不同（为此构造一个**cst.arj**），启动mcst\_640.bat / mcst\_800.bat 或者 **cst1024.bat**。
- 这样所产生的**cst.arj**位于同一个目录下，如同生产工具。
- 自软件版本 SW 6.3 起，生成一个另外的文件**cst.idx**，该文件同时也存在于该目录中。该文件与 **cst.arj** 一起捆绑在 HMI 软件中。



在章节1.4.6中说明**cst.arj**文件如何捆绑到HIM软件中。



文献:            **/BEM/, 操作说明 HMI Embedded**  
                  **/IAM/, 安装调试说明 HMI**

IM2 “HMI mbedded（内置）的开机调试”

## 1.6 循环调试

### 1.6.1 机床数据

使用循环时需要注意下列机床数据。必须至少在表格中有这些给定的值。

#### 需要注意的机床数据

MD号	MD名称	最小值
18118	MM_NUM_GUD_MODULES	7
18130	MM_NUM_GUD_NAMES_CHAN	20
18150	MM_GUD_VALUES_MEM	2 * 通道数
18170	MM_NUM_MAX_FUNC_NAMES	40
18180	MM_NUM_MAX_FUNC_PARAM	500
28020	MM_NUM_LUD_NAMES_TOTAL	200
28040	MM_NUM_LUD_VALUES_MEM	25



这些数据仅适用于西门子标准循环。对于用户循环，必须添加这些相应的值。使用 ShopMill 或 ShopTurn 时，需要注意该产品相应的说明。

另外需要以下的机床数据设定：

MD号	MD名称	值
20240	CUTCOM_MAXNUM_CHECK_BLOCK	4

由机床制造商提供机床数据文件，并且已经进行了预置。需要注意的是：在修改了这些机床数据之后必需要进行一次上电。



另外，对于循环CYCLE840（攻丝，带补偿衬套）需要注意轴专用的机床数据MD30200:NUM\_ENCS。



## 1.6.2 定义文件，用于循环GUD7.DEF 和 SMAC.DEF

标准循环需要定义全局用户数据（GUD）和宏指令定义。这些定义存储在标准循环一同提供的定义文件 GUD7.DEF 和 SMAC.DEF 中。

为了使调试人员可以方便地把GUDs和宏指令整合到其模块中，而无需编辑西门子源文件，自SW6.3起，这些文件

- GUD7.DEF
- MAC.DEF

与“标准循环”一起全套提供。这两个文件中不包含定义，而仅仅是定义的，

与产品相关的定义文件的调用。目前位于循环中的调用机制可自动调用并汇编所有产品相关的 GUD 定义和宏指令定义。

每个包现在只带有它本身的定义。为此要引入新的循环文件 GUD7\_xxx.DEF 和 SMAC\_xxx.DEF，这些文件位于定义目录 DEF.DIR 中的文件维护中。

对于标准循环它们是新文件：

- GUD7\_SC.DEF 和
- SMAC\_SC.DEF

对于其它的循环文件包，目前，以下的文件标识由西门子占用：

（xxx 代表“GUD7” 或者“SMAC”）

- xxx\_JS JobShop 通用循环
- xxx\_MC 测量循环
- xxx\_MJ JOG方式下测量
- xxx\_MT ManualTurn
- xxx\_SM ShopMill
- xxx\_ST ShopTurn
- xxx\_ISO ISO 兼容
- xxx\_C950 扩展的毛坯切削
- xxx\_C73 带中心岛的凹槽

这里没有列出的其它标识可能会在以后由系统使用。

**提示**

依据定义用户数据（参见编程说明工作准备章节3.4“定义用户数据”），不为机床制造商/机床用户提供模块 GUD7 和 SMAC.DEF！应该将MGUD, UGUD, GUD4, 8, 9 或 MMAC, UMAC 优先用于用户应用程序。

但是为了让用户可以把已经存在的自身定义集成到该系统的这些模块中，保留以下的标识

- xxx\_CMA 机床制造商
- xxx\_CUS 用户

**安装调试，标准循环的升级：**

- 如果在控制系统中已经有一个GUD7.DEF有效，则通过“通讯”，“数据输出”，“NC有效数据”选择GUD7的用户数据，并且把当前的数值存档或者保护到磁盘上；
- 文件 GUD7\_SC.DEF 和 SMAC\_SC.DEF 由磁盘读入并装载到NCU中；
- 读入并激活GUD7.DEF 和 SMAC.DEF ；
- 执行NCU的Power-on（上电）；
- 存档中所保护的数值再次读入；

**装载一个其它的循环包：**

- 卸载GUD7.DEF 和 SMAC.DEF（如可能事先存储数值），
- 读入软件包中的循环 GUD7\_xxx.DEF 和 SMAC\_xxx.DEF 并装载到NCU中；
- 再次激活GUD7.DEF 和 SMAC.DEF 。

**提示**

再次装载或卸载单个定义文件时，必须卸载一个已装载的调用文件并装载一个新的调用文件。否则 NC 保留先前存在的 GUD / 宏指令配置。

**处理HMI Advanced（高级）配置中的仿真：**

在把循环装载到NCU之后，在启动仿真之后，首先要求使用NC复位仿真，从而对机床数据进行补偿，这样才可以激活所修改的定义文件。

### 1.6.3 在 HMI Advanced 配置中，循环的供货形式



自 HMI Advanced 6.3 起，HMI 配置中的标准循环供货形式将会发生改变。循环文件将不再直接存储在相应的文件维护目录中，而是作为存档文件存储。

→ 存档/循环存档。

由此，以前在文件维护中存在的循环状态在 HMI 升级时保持不变。

若要进行升级，必须通过“输入数据”读入这些存档文件。通过读入这些存档文件，完成升级过程后在 NCU 和硬盘上不再存在不同的循环版本。在 NCU 中覆盖装载的循环，而不是装载在硬盘上的循环。新的循环文件主要存储在硬盘上。



**文献：**有关当前信息，请参见：

- 供货软件（标准循环）的文件“siemensd.txt”，或者
- 在 HMI Advanced（高级）配置中 F:\dh\cst.dir\HLP.dir\siemensd.txt

### 1.6.4 循环升级，自 SW 6.4 起；在 HMI Advanced 中自 SW 6.3 起



在 HMI\_Advanced 中，自 SW 6.3 起在“存档”，“循环存档”

下将循环作为存档文件存储。要进行升级，必须读入供货软件中所包含的 HMI\_Adv 目录下的存档文件。

压缩循环存档文件，并覆盖之前循环状态中现有的存档文件。

来自 HMI 循环存档文件中的存档文件，现在必须通过“数据输入”按照工艺相关（车床，铣床，..）或语言相关读入。



**文献：**有关当前信息，请参见：

- 供货软件（标准循环）的文件“siemensd.txt”，或者
- 在 HMI Advanced（高级）配置中 F:\dh\cst.dir\HLP.dir\siemensd.txt

## 1.7 循环辅助功能



### 功能

可自软件版本 SW 6.3

起显示和使用版本画面，以概述和诊断循环状态及其定义文件。它们位于 HMI

中的“诊断”=>“维修信息”=>“版本”=>“循环”以及“定义”下。可生成一个 ASCII 格式的日志文件并可在“服务”=>“诊断”=>“日志文件”下读出。

该功能同样自SW6.3起仅可以用HMI软件版本执行。

循环版本显示可以产生不同的画面：

- 所有可用循环的概述
- 显示文件维护的各个目录，用于用户循环(CUS.DIR)，制造商循环 (CMA.DIR) 和西门子循环 (CST.DIR)。
- 显示系统中存在的循环软件包的概述
- 各个软件包和循环文件的详细情况。



文献： /BAD/, 操作说明 HMI Advanced  
/BEM/, 操作说明 HMI Embedded  
章节“维修信息”。

版本显示包含所有循环文件\*.SPF和所有循环辅助文件\*.COM。

对于目录或者所有循环的版本显示，不需要附加的文件。为了可以显示各个循环软件包，每个循环软件包必须携带一个所有相关文件的软件包清单。



### 软件包清单

引入一个新的文件类型用于软件包清单

**\*.cyp** (用于循环软件包),

以明文显示的循环软件包清单。

可由用户编制用于自身循环软件包的软件包清单。必须按如下方式进行：

一个软件包清单的构成：

1. 行： 版本登录（在关键字之后  
； **VERSION:** ）并且  
软件包名称（在关键字之后  
； **软件包:** ）

自第2行起： 属于循环软件包的文件清单  
带名称和类型

最后一行： M30

举例：

```
%_N_CYC_USER1_CYP
; $PATH=/_N_CUS_DIR
; VERSION: 01.02.03 31.10.2002 ; PACKAGE: $85200
ZYKL1.SPF
ZYKL2.SPF
ZYKL3.COM
M30
```

在文本文件uc.com 中登录：

85200 0 0 “循环软件包 1”

在软件包概述中显示：

Diagnosis	CHAN1	AUTO	MPFO
Channel RESET			Program aborted
			ROV SBL1
<b>Version data Cycles</b>			
Name	Version		
Measuring cycles	06.02.08 Mar 21, 2002		
Cycle package	01.02.03 31.10.2002		

NCU version
HMI version
Cycle version
Definition version

在文件概述中显示:

Diagnosis	CHAN1	AUTO	MPF0		
Channel RESET			Program aborted		
			ROV SBL1		
Version data ShopMill cycles					
Name	Type	Load	Length	Directory	Version
ZYKL1	SPF	1234	CUS.DIR	01.02.03	31.10.2002
ZYKL2	SPF	778	CUS.DIR	01.02.03	31.10.2002
ZYKL3	COM	521	CUS.DIR	01.02.03	31.10.2002

### 其它说明

在关键字PACKAGE之后的循环软件包名称也可以作为字符串在“”中写入，然后它还与语言相关。

### 循环中的版本登录

正如在软件包清单中一样，在“VERSION:”后面的登录作为版本标记。版本登录只允许在循环的第一个十行中，如果超过这十行，则无法查找到。

#### 举例:

```

%_N_ZYKL1_SPF
; $PATH=/_N_CUS_DIR
; VERSION: 01.02.03 31.10.2002
; 注释
PROC ZYKL1 (REAL PAR1)
...

```

■

## 钻削循环和钻削图！

2.1	钻削循环 .....	2-48
2.1.1	前提条件 .....	2-50
2.1.2	钻削，定中心 - CYCLE81 .....	2-51
2.1.3	钻削，镗平面 - CYCL82 .....	2-54
2.1.4	深孔钻削 - CYCLE83 .....	2-56
2.1.5	攻丝，不带补偿衬套 - CYCLE84 .....	2-63
2.1.6	攻丝，带补偿衬套 - CYCLE840 .....	2-70
2.1.7	镗孔 1 - CYCLE85 .....	2-78
2.1.8	镗孔 2 - CYCLE86 .....	2-81
2.1.9	镗孔 3 - CYCLE87 .....	2-85
2.1.10	镗孔 4 - CYCLE88 .....	2-87
2.1.11	镗孔 5 - CYCLE89 .....	2-89
2.2	模态调用钻削循环 .....	2-91
2.3	钻削图循环 .....	2-94
2.3.1	前提条件 .....	2-94
2.3.2	成排孔 - HOLES1 .....	2-95
2.3.3	孔圆弧 - HOLES2 .....	2-99
2.3.4	点栅格 - CYCLE801 .....	2-102

## 2.1 钻削循环

在下面的章节中，说明如何编程

- 钻削循环和
- 钻削图循环。

要求将该章节作为选择循环时的指南并使用所提供的各种参数。除了各个循环功能的详尽描述和为此所列出的参数外，您还可在各节结束处找到能简化循环过程的编程举例。

各节是按照以下的原则进行组合的：

- 编程
- 参数
- 功能
- 工作流程
- 参数说明
- 其它说明
- 程序举例

上述各点中，编程和参数两节为熟练的用户所准备，而作为初学者，有关循环编程的必要知识均可以在功能，工作流程，参数说明，其它说明和编程举例中找到。





钻削循环为根据DIN66025所确定的运动过程，用于钻削，镗孔，攻丝等等。

其调用通过一个确定的名称和参数表进行，作为子程序。

共提供有五个循环用于镗孔。它们在工艺流程中有所不同，由此在其参数上也不尽相同。

镗孔循环		参数设定的特点
镗孔1	CYLCE85	用于钻削和退回有不同的进给
镗孔2	CYLCE86	定向的主轴停，给定退回位移，快速退回，给定主轴方向
镗孔3	CYLCE87	在钻削深度的主轴停M5和程序停M0，NC启动后其它加工，快速退回，给定主轴方向
镗孔4	CYLCE88	如同CYCLE87，增加钻削深度的停留时间
镗孔5	CYLCE89	以相同的进给钻削和退回



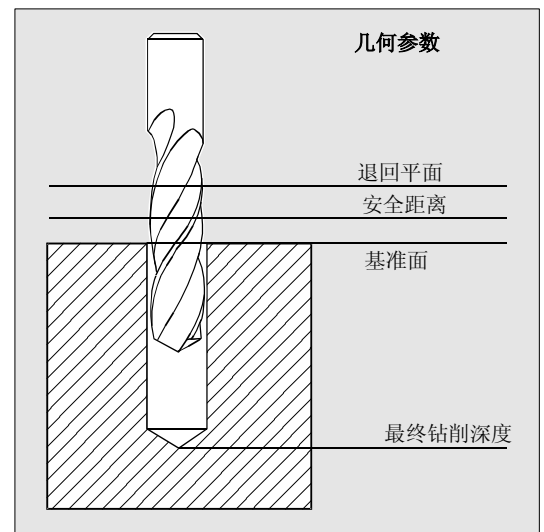
钻削循环可以模态有效，即在每个包含运动指令的程序段末端进行钻削循环。同样，可以模态调用其它由用户建立的循环（参见章节 2.2）。

有两种参数种类：

- 几何参数和
- 加工参数

几何参数对于所有钻削循环，钻削图循环和铣削循环都相同。该参数定义基准面和退回平面，安全距离以及绝对最终钻削深度和相对最终钻削深度。在第一个钻削循环 CYCLE81 时，几何参数仅描述一次。

对于各个循环，加工参数有不同的意义和作用。因此，对于各个循环分开描述。



### 2.1.1 前提条件

#### 调用条件和返回条件

钻削循环编程和具体的轴名称无关。在上一层程序中的循环调用之前，返回运行钻削位置。

如果在此所要求的进给，主轴转速和主轴方向的值在钻削循环中没有提供参数，则它们必须在零件程序中编程。

在循环调用之前有效的 G 功能和当前的框架可以在循环之后仍然保持。

#### 平面定义

在钻削循环中一般以以下为前提：当前加工的工件坐标系通过选择一个平面 G17, G18 或者 G19 和激活一个可编程的框架定义。钻孔轴始终为该坐标系中的应用轴方向。

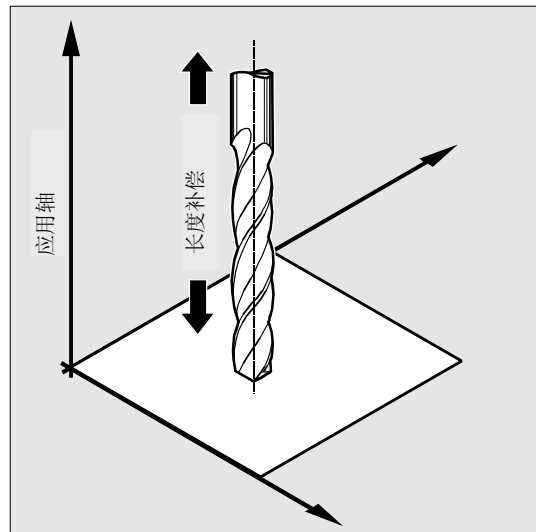
在调用前必须选择长度补偿。该长度补偿总是垂直作用于所选的平面并在循环结束后保持激活状态（另见编程说明）。

#### 主轴处理

在编制钻削循环时，始终使其中所包含的主轴指令以系统中有效的主主轴为基准。如果您想要在带有多个主轴的机床上使用钻削循环，则必须在此之前将要用于加工的主轴定义为主主轴（另见编程说明）。

#### 停留时间编程

在钻削循环中停留时间参数始终分配到 F 字，并相应地设定一个单位为秒的数值，其偏差必须明确说明。



## 2.1.2 钻削，定中心 – CYCLE81



### 编程

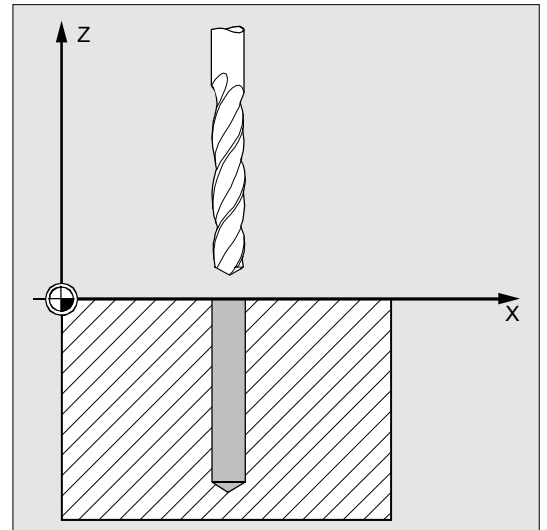
CYCLE81 (RTP, RFP, SDIS, DP, DPR)

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）



### 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。



### 工作流程

#### 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

#### 该循环产生以下的运动过程：

以G0返回到提前了相隔安全距离的基准面

- 以在所调用程序中编程的进给（G1）运行到钻削深度
- 以G0返回到退回平面



## 参数说明

### RFP 和 RTP（基准面和退回平面）

通常，基准面（RFP）和退回平面（RTP）包含有不同的值。在循环中这主要是指，退回平面位于基准面之前。退回平面与最终钻削深度的距离也大于基准面与最终钻削深度的距离。

### SDIS（安全距离）

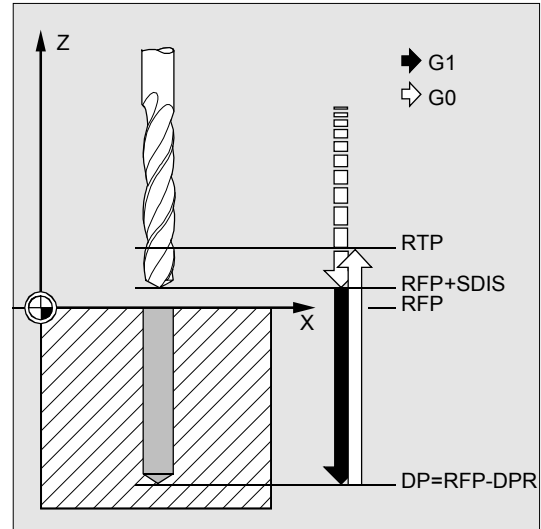
安全距离（SDIS）与基准面有关。基准面提前了安全距离。

安全距离生效的方向由循环自动确定。

### DP 和 DPR（钻削深度）

钻削深度可以以到基准面的绝对尺寸（DP）规定，也可以以到基准面的相对尺寸（DPR）规定。

在相对尺寸时，循环利用基准面和退回平面的位置自己计算所产生的深度。



## 其它说明

如果不仅输入了一个DP值，还输入了一个DPR值，则可导出DPR最终钻削深度。如果该值与通过DP编程的绝对深度不同，则在对话行中给出信息“深度：用于相对深度的相应值”。

如果基准面和退回平面的值一致，则不允许说明一个相对的深度尺寸。给出故障信息

### 61101 “基准面错误定义”

并且不执行该循环。

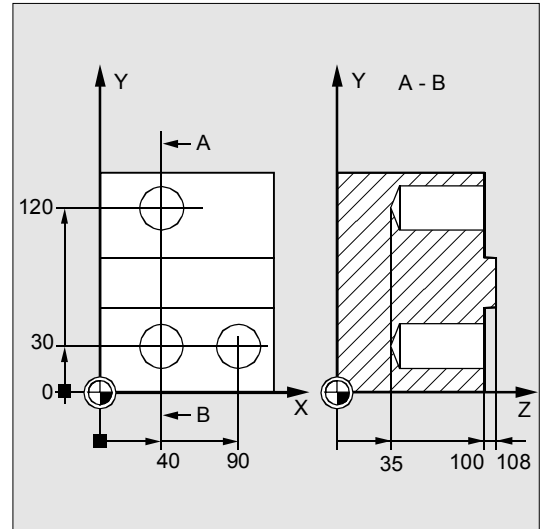
如果退回平面位于基准面之后，则也会生成一个故障信息，其到钻削深度的距离也较小。



### 编程举例

#### 钻削\_定中心

用此程序可以通过使用钻削循环 **CYCLE81** 加工 3 个孔，这里该程序使用不同的参数进行调用。钻削轴始终为 Z 轴。



N10 G0 G90 F200 S300 M3	; 确定工艺数值
N20 D1 T3 Z110	; 返回退回平面
N21 M6	
N30 X40 Y120	; 返回第一个钻削位置
N40 CYCLE81 (110, 100, 2, 35)	; 循环调用，带绝对钻削深度；安全距离和不完整的参数表
N50 Y30	; 返回到下一个钻削位置
N60 CYCLE81 (110, 102, , 35)	; 循环调用，没有安全距离
N70 G0 G90 F180 S300 M03	; 确定工艺数值
N80 X90	; 返回到下一个位置
N90 CYCLE81 (110, 100, 2, , 65)	; 循环调用，带相对钻削深度和安全距离
N100 M30	; 程序结束

## 2.1.3 钻削, 镗平面 – CYCL82



## 编程

CYCLE82 (RTP, RFP, SDIS, DP, DPR, DTB)



## 参数

RTP	实数	退回平面 (绝对)
RFP	实数	基准面 (绝对)
SDIS	实数	安全距离 (不输入符号)
DP	实数	孔底深度 (绝对)
DPR	实数	相对于基准面的钻削深度 (不输入符号)
DTB	实数	在钻削底部的停留时间 (断屑)



## 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。如果已达到最终钻削深度，则某个停留时间可生效。



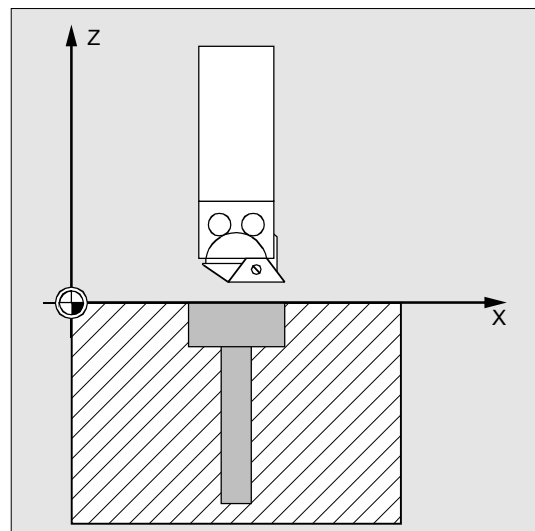
## 工作流程

## 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

## 该循环产生以下的运动过程：

- 以 G0 返回到提前了相隔安全距离的基准面
- 以在所调用程序中编程的进给 (G1) 运行到钻削深度
- 执行在钻削底部的停留时间
- 以 G0 返回到退回平面





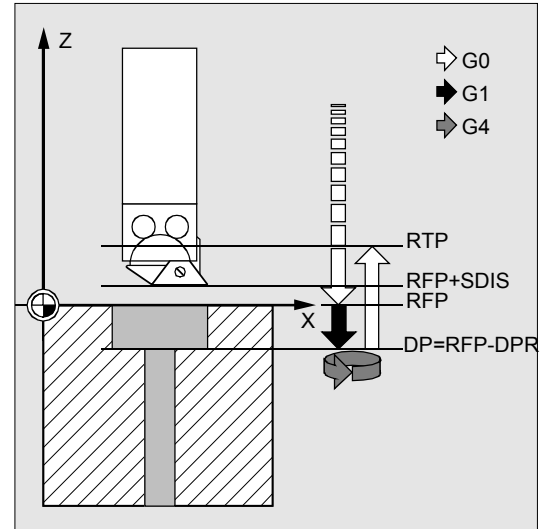
### 参数说明



参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

### DTB (停留时间)

在 DTB 下编程钻削底部 (断屑) 的停留时间,  
单位秒。

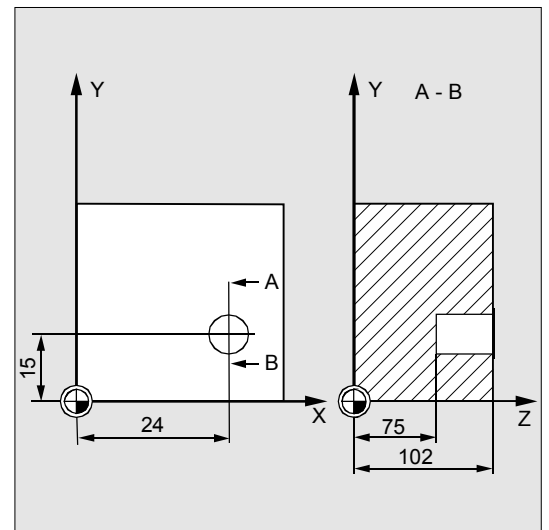


### 编程举例

#### 钻削\_镗平面

在 XY 平面中 X24 Y15 位置, 使用循环 CYCLE82  
执行一次钻孔, 深度为 27毫米。

停留时间为2秒, 在钻削轴Z上安全距离为4毫米。



N10 G0 G90 F200 S300 M3

; 确定工艺数值

N20 D1 T3 Z110

; 返回退回平面

N21 M6

N30 X24 Y15

; 返回钻削位置

N40 CYCLE82 (110, 102, 4, 75, , 2)

; 循环调用, 带绝对钻削深度和安全距离

N50 M30

; 程序结束

## 2.1.4 深孔钻削 – CYCLE83



## 编程

CYCLE83 (RTP, RFP, SDIS, DP, DPR, FDEP, FDPR, DAM, DTB, DTS, FRF, VARI, \_AXN, \_MDEP, \_VRT, \_DTD, \_DIS1)



## 参数

RTP	实数	退回平面 (绝对)
RFP	实数	基准面 (绝对)
SDIS	实数	安全距离 (不输入符号)
DP	实数	孔底深度 (绝对)
DPR	实数	相对于基准面的钻削深度 (不输入符号)
FDEP	实数	第一个钻削深度 (绝对)
FDPR	实数	相对于基准面的第一个钻削深度 (不输入符号)
DAM	实数	递减: (不输入符号) 值: > 0 递减量 < 0 递减系数 = 0 没有递减
DTB	实数	在钻削底部的停留时间 (断屑) 值: > 0 单位秒 < 0 单位转
DTS	实数	在起始处的停留时间, 用于退刀排屑值: > 0 单位秒 < 0 单位转
FRF	实数	进给系数, 用于第一个钻削深度 (不输入符号) 值范围: 0.001...1
VARI	整数	加工方式: 值: 0 断屑 1 退刀排屑
_AXN	整数	工具轴: 值: 1 = 1. 几何轴 2 = 2. 几何轴 其它为第三个几何轴
_MDEP	实数	最小孔深
_VRT	实数	在断屑时(VARI=0)可变的退回量 值: > 0 退回量 0 = 1 毫米
_DTD	实数	在钻削底部的停留时间值: > 0 单位秒 < 0 单位转 = 0 值如同 DTB
_DIS1	实数	在再次进入钻孔时可编程的移前距离 (在断屑VARI=1时) 值: > 0 可编程的值适用 = 0 自动计算





## 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。

对于深孔钻削也可以多次，分布地进行深度进刀，其最大进刀量可以规定，直至加工到钻削深度。

钻头可以在每次进刀深度（用于退刀排屑）之后退回到基准面+安全距离，或者也可以用于断屑退回编程的退回位移。



## 工作流程

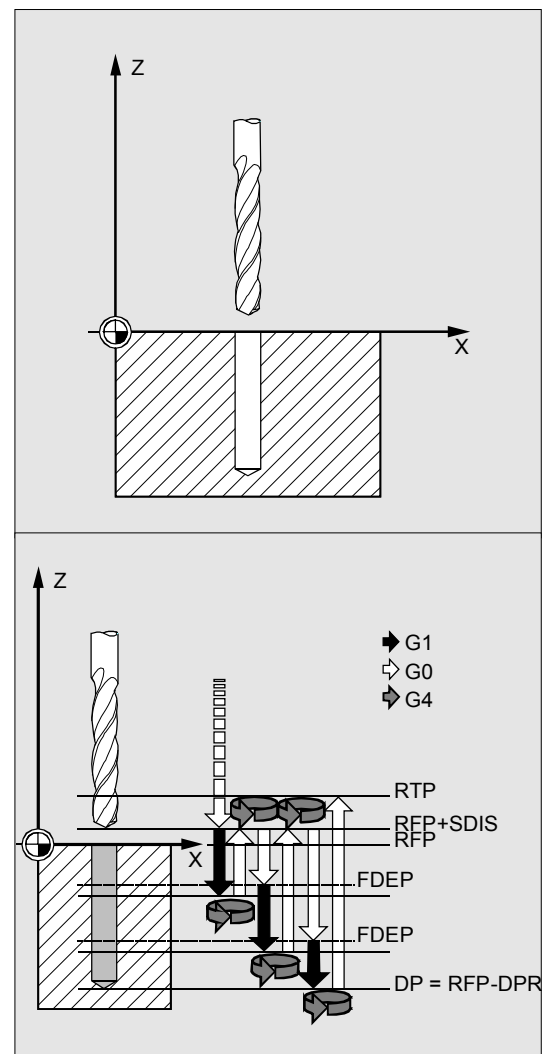
循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

该循环产生以下的运动过程：

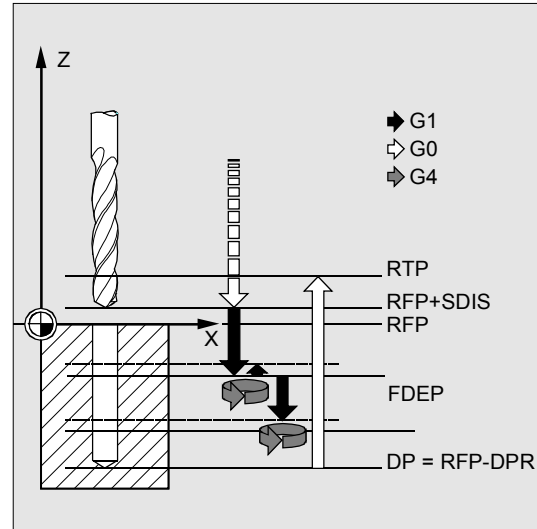
深孔钻削，带退刀排屑（VARI=1）：

- 以G0返回到提前了相隔安全距离的基准面
- 以G1运行到第一个钻削深度，进给由循环调用时编程的进给与参数FRF（进给系数）计算而产生
- 执行钻削深度（参数DTB）时的停留时间
- 以G0退回到提前了安全距离的基准面，用于退刀排屑
- 执行起始处（参数DTS）的停留时间
- 以G0返回到最后到达的钻削深度，减少循环内部计算的前移距离或者可编程的前移距离
- 以G1运行到下一个钻削深度（运动过程一直继续，直至到达钻削深度）
- 以G0返回到退回平面



**深孔钻削，带断屑 (VARI=0)：**

- 以G0返回到提前了相隔安全距离的基准面
- 以G1运行到第一个钻削深度，进给由循环调用时编程的进给与参数FRF（进给系数）计算而产生
- 执行钻削深度（参数DTB）时的停留时间
- 以G1和程序中编程的进给率从当前的钻削深度可変退回（参数\_VRT）（用于断屑）
- 以G1和编程的进给率运行到下一个钻削深度（运动过程一直继续，直至到达钻削深度）
- 以G0返回到退回平面

**参数说明**

参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

**FDEP 和 FDPR (第一个钻削深度, 绝对或者相对)**

可以有选择性地通过这两个参数中的一个编程第一个钻削深度。参数 FDPR 如同参数 DPR 一样在循环中生效。如果基准面和退回平面的值一致，则可以规定第一个钻削深度的相对值。

**DAM (递减)**

在分步钻削的深孔加工中，使用各个步骤递减的数值进行加工（递减）很有必要。因为这样可以排出铁屑，从而刀具不易折断。

对此，可以在参数中编程一个增量式递减数值，从而可以分步减少第一个钻削深度，或者也可以使用%值作为递减系数。

**DAM=0 没有递减**

**DAM>0 递减量**

在循环中可以按如下方式计算当前的深度：

- 如果第一步没有超出总钻削深度，则运行通过第一个钻削深度FDEP或者FDPR所设定的参数深度。
- 自第二步起，钻削行程等于最后一步钻削的深度减去递减量，如果钻削行程大于编程的递减量。  
如果已得出某个用于第二个钻削行程的值小于编程的递减值，则一步进刀至该值。
- 接下去的钻削行程等于递减量，只要剩余深度大于双倍的递减量。
- 最后两步钻削行程平均分布并运行，并且始终大于半个递减量。
- 如果第一次钻削的深度数值与总深度矛盾，则给出故障报警61107“第一次钻深错误定义”，循环不执行。

示例:

编程值 RTP=0, SDIS=0, DP=-40, FDEP=-12 和 DAM=3  
形成以下的钻削行程:

-12	等于第一次钻深
-21	第一次钻深12减少递减量3, 产生增量差9
-27	前面的钻深减去递减量3
-30, -33, -36	递减量
-38, -40	剩余深度分为两步

**DAM<0 (-0.001 至 -1) 递减系数**

在循环中可以按如下方式计算当前的深度:

- 如果第一步没有超出总钻削深度, 则运行通过第一个钻削深度FDED或者FDPR所设定的参数深度。
- 后面的钻削行程等于最后的钻削行程乘以递减系数, 只要该行程不小于最小钻深。
- 最后两步钻削行程平均分布并运行, 并且始终大于半个最小钻深。
- 如果第一次钻削的深度数值与总深度矛盾, 则给出故障报警61107 “第一次钻深错误定义”, 循环不执行。
- 如果编程一个值DAM<-1, 则循环中断并给出故障报警61019 “参数DAM定义错误”

示例:

编程值 RTP=0, SDIS=0, DP=-40, FDEP=-10, DAM=-0.8 和 MDEP=5  
形成以下的钻削行程:

- 10            等于第一次钻深
- 18            增量差 8 等于 0.8 \* 第一次钻深
- 24.4, -29.52    各自以前的钻深 \* 递减系数
- 34.52        最小钻深 MDEP 作用
- 37.26, -40    剩余深度分为两步

**DTB (停留时间)**

在DTB下编程钻削底部 (断屑) 的停留时间, 单位秒或者主轴转数。

- 0 单位秒
- < 0 单位转数

**DTS (停留时间)**

仅在VARI=1 (退刀排屑) 时, 起始处的停留时间才执行。

- 值 > 0 单位秒
- 值 < 0 单位转数

**FRF (进给系数)**

通过该参数您可以说明当前进给的换算系数, 这仅在循环运行到第一次钻深时加以考虑。

如果FRF编程过大，则不发出报警。循环内部系数限制为1。

### VARI（加工方式）

如果设定参数

VARI=0，则钻头在每次达到断屑的钻削深度后以编程的退回位移自由行进。当 VARI=1

（用于退刀排屑）时，钻头总是运行至提前了安全距离的基准面上。

### \_AXN（刀具轴）

在车床上使用深孔循环时，通过由\_AXN编程钻削轴，可以省去平面G18到G17的转换。

这表明：

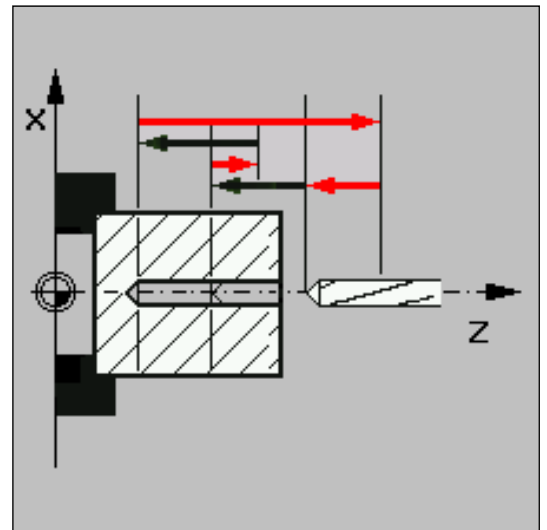
- \_AXN=1    1. 当前平面的第1轴
- \_AXN=2    2. 当前平面的第二轴
- \_AXN=3    3. 当前平面的第3轴

比如，为了进行G18平面中一次定中心钻削

（Z方向），您需要编程：

G18

\_AXN=1



### \_MDEP（最小钻削深度）

在计算钻削行程时，通过一个递减系数可以确定一个最小钻削深度。如果最小钻削深度小于计算后的钻削行程，则行程中剩余的钻削深度由最小钻削深度尺寸完成加工。

### \_VRT（在断屑VARI=0时，可变的退回量）

在断屑时可以编程退回位移。

值 > 0 退回量

值 = 0 退回量为1毫米

### \_DTD（在钻削底部的停留时间）

在钻削底部的停留时间可以为秒，也可以为转数。

值 > 0 单位秒

值 < 0 单位转数

值 = 0 停留时间如同DTB中编程的一样

**\_DIS1 (在 VARI=1 时可编程的前移距离)**

可以编程再次进入钻削后的前移距离。

值 > 0 定位到编程的值

值 = 0 自动计算

循环内部按如下方式计算前移距离：

- 至30毫米的钻削深度，该值设置为0.6毫米
- 在较大的钻削深度时，从 (RFP + SDIS - 当前深度) / 50中产生前移距离，如果这样计算的值 > 7，则限制到最大7毫米。

**编程举例****深孔钻削**

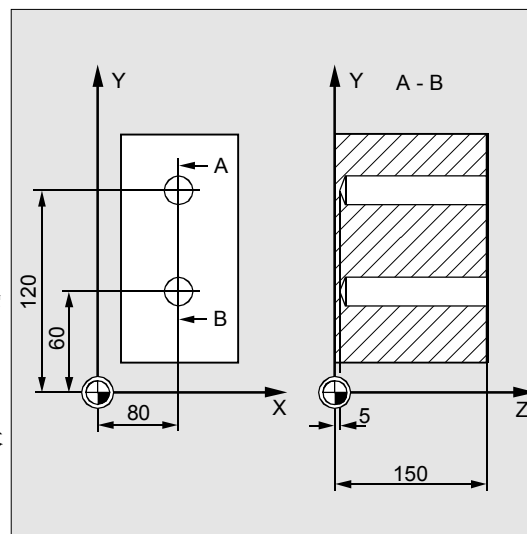
该程序在 XY 平面中的位置 X80 Y120 和 X80 Y60 上执行循环 CYCLE83。

第一个钻孔停留时间为零，并进行断屑加工方式。

最终钻削深度以及第一个钻削深度规定为绝对坐标值。

在第二次调用时，停留时间编程为1秒。已选择退刀排屑加工方式，规定相对于基准面的最终钻削深度。在这两种情况下，钻削轴为 Z 轴。

钻削行程通过一个递减系数计算，并且不应该小于8毫米的最小深度。



```

DEF REAL RTP=155, RFP=150, SDIS=1, DP=5, ; 参数定义
DPR=145, FDEP=100, FDPR=50, DAM=20,
DTB=1, FRF=1, VARI=0, _VRT=0.8, _MDEP=10,
_DIS1=0.4
N10 G0 G17 G90 F50 S500 M4 ; 确定工艺数值
N20 D1 T42 Z155 ; 返回退回平面
N30 X80 Y120 ; 返回第一个钻削位置
N40 CYCLE83 (RTP, RFP, SDIS, DP, , -> ; 调用循环，带绝对值的深度参数
-> FDEP, , DAM, , , FRF, VARI, , , _VRT)
N50 X80 Y60 ; 返回到下一个钻削位置
N55 DAM=-0.6 FRF=0.5 VARI=1 ; 赋值
N60 CYCLE83 (RTP, RFP, SDIS, , DPR, , ; 调用循环，带孔底深度和第一钻深的相
-> ; 对参数，安全距离为1毫米，进给系
-> FDPR, DAM, DTB, , FRF, VARI, , ; 数为0.5
_MDEP, -> , , _DIS1)

```

N70 M30 ; 程序结束

-> 必须在一个程序段中编程

### 2.1.5 攻丝，不带补偿衬套 – CYCLE84



#### 编程

CYCLE84 (RTP, RFP, SDIS, DP, DPR, DTB, SDAC, MPIT, PIT, POSS, SST, SST1, \_AXN, \_PTAB, \_TECHNO, \_VARI, \_DAM, \_VRT)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DTB	实数	在螺纹深度的停留时间（断屑）
SDAC	整数	循环结束之后的旋转方向 值：3, 4或者5
MPIT	实数	螺距作为螺纹尺寸（带符号） 值范围：3（用于M3） ... 48（用于M48），符号确定螺纹中的旋转方向
PIT	实数	螺距作为值（带符号） 值范围：0.001 ... 2000.000 毫米}，该符号确定螺纹中的旋转方向： 如果 _PTAB=0 或者 1：以毫米为单位（迄今为止） 当 _PTAB=2时，每英寸螺纹导程
POSS	实数	主轴位置，用于循环中定向主轴停（单位度）
SST	实数	攻丝时转速
SST1	实数	退回时转速
_AXN	整数	刀具轴值： 1 = 1. 几何轴 2 = 2. 几何轴 其它为第三个几何轴
_PTAB	整数	求值螺距 PIT 值： 0...相当于编程的尺寸系统英制/公制 1...螺距单位毫米 2...螺距，每英寸的螺纹导程 3...螺距，单位英寸/转

_TECHNO	整数	工艺设定 个位：准停特性 值： 0...如同调用循环之前编程 1... (G601) 2... (G602) 3... (G603) 十位：预控制 值： 0...如同调用循环之前编程 1...带预控制 (FFWON) 2...不带预控制 (FFWOF) 百位：加速度 值： 0...如同调用循环之前编程 1...轴冲击限制加速度 (SOFT) 2...轴突变加速度 (BRISK) 3...轴降低加速度 (DRIVE) 千位： 值： 0...再次激活主轴驱动 (在 MCALL时) 1...处于位置控制运行 (在 MCALL时)
_VARI	整数	加工方式： 值： 0...一次进程中攻丝 1...深孔攻丝，带断屑 2...深孔攻丝，带退刀排屑
_DAM	实数	增量钻削深度 值范围：0 ≤ 最大值
_VRT	实数	可变退回量，用于断屑 值范围：0 ≤ 最大值



## 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的螺纹深度。

使用循环CYCLE84，您可以加工不带补偿衬套的攻丝。

该循环也可以选择分步执行攻丝（深孔钻削）。



如果用于钻削的主轴进入位置控制运行状态，则可以使用循环CYCLE84。



对于带补偿衬套的攻丝，则有一个自身的循环CYCLE840（参见章节2.1.6）。





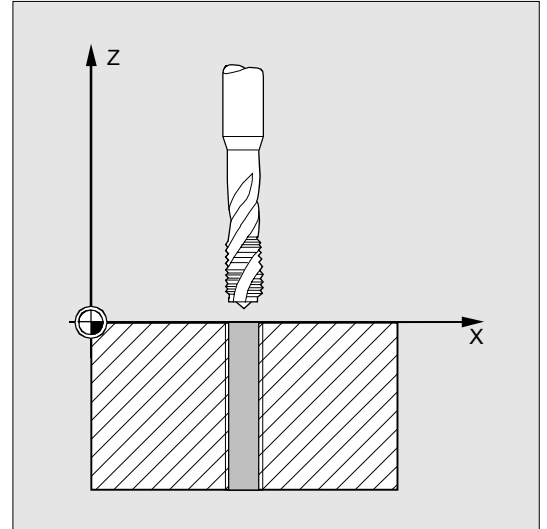
## 工作流程

### 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

### 该循环产生以下的运动过程：

- 以G0返回到提前了相隔安全距离的基准面
- 带 SPOS的定向主轴准停（参数 POSS中的值）并且主轴转入轴运行
- 用G331和转速SST攻丝，至孔底深度
- 执行螺纹深度（参数DTB）处的停留时间
- 以G332, 转速SST1和反向退回到提前了安全距离的基准面
- 用G0退回到退回平面，重写循环调用之前最后编程的主轴转速和在SDAC下编程的方向，使运行方式再次回到主轴运行。



## 参数说明



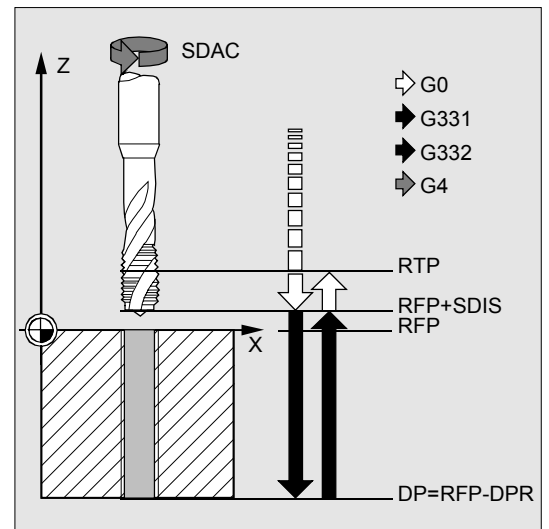
参数 RTP, RFP, SDIS, DP, DPR 参见章节 2.1.2. (钻削, 定中心 - CYCLE81)

### DTB (停留时间)

以秒钟为单位编程停留时间。在长袋孔钻削时建议可取消停留时间。

### SDAC (循环结束之后转向)

在结束循环之后，在SDAC之下编程转向。  
在攻丝时在循环内部自动进行方向转换。



**MPIT 和 PIT (作为螺纹尺寸和数值)**

螺距值可以规定为螺纹尺寸（仅用于M3和M48之间的公制螺纹），也可以规定为数值（从一个螺纹头至下一个螺纹头的间距作为数值）。取消每次调用中不需要的参数或者将值置零。

右螺纹或者左螺纹通过螺距参数的符号确定：

- 正值 → 右螺纹（如同M3）
- 负值 → 左螺纹（如同M4）

如果两个螺距参数有相互矛盾的值，则由循环产生报警61001“螺距错误”，并中断循环的加工。

**POSS (主轴位置)**

在循环中，在攻丝之前用指令 SPOS

定向停止主轴，并进入位置控制状态。

在POSS下编程主轴位置，用于主轴停止。

**SST (转速)**

参数SST包含主轴转速，用于带G331的攻丝程序段。

。

**SST1 (退回时转速)**

在SST1下编程从攻丝程序段（带G332）退回的转速。

如果该参数值为零，则以SST下编程的转速退回。

**\_AXN (工具轴)**

在车床上使用深孔攻丝时，通过由\_AXN编程钻削轴，可以省去平面G18到G17的转换。

这表明：

AXN=1      1. 当前平面的第1轴

AXN=2      2. 当前平面的第2轴

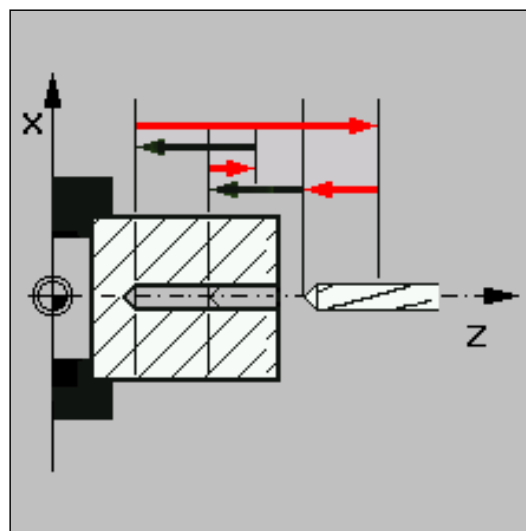
\_AXN=3      3. 当前平面的第3轴

比如，为了进行G18平面中一次定中心钻削

（Z方向），您需要编程：

G18

\_AXN=1



### **\_PTAB (螺距 PIT的求值)**

参数\_PTAB确定螺距的尺寸单位。

- 0=相当于编程的尺寸系统英制/公制
- 1=螺距, 单位毫米
- 2=螺距, 每英寸螺纹导程
- 3=螺距, 单位英寸/转

根据循环辅助中不同螺纹表格的选择方法不同, 需要该参数。

### **\_TECHNO (工艺设定)**

使用参数\_TECHNO可以设定攻丝时的工艺性能。

可能的值是:

#### **个位 (准停性能):**

- 0 = 如同调用循环之前编程
- 1 = (G601)
- 2 = (G602)
- 3 = (G603)

#### **十位 (预控制):**

- 0 = 如同调用循环之前编程
- 1 = 带预控制 (FFWON)
- 2 = 不带预控制 (FFWOF)

#### **百位 (加速度):**

- 0 = 如同调用循环之前编程
- 1 = 轴冲击限制加速度 (SOFT)
- 2 = 轴突变加速度 (BRISK)
- 3 = 轴降低加速度 (DRIVE)

#### **千位:**

- 0 = 再次激活主轴驱动 (在 MCALL时)
- 1 = 处于位置控制运行 (在 MCALL时)



在“工艺”输入区按“是”, 可以由机床制造商和操作人员/程序员进行攻丝时的工艺匹配。

### **机床制造商匹配 (自SW6.4)**

- 前提条件: 设置制造商口令, “工艺”输入区= “是”
- 在打开输入屏循环CYCLE84 情况下, 用GUD7-变量 \_SC\_MASK[0] 值预设参数。在更改参数时该值直接写入在GUD7变量中。
- 机床制造商由此可以匹配基本设定到其机床的具体情况。

### 操作人员/程序员匹配 (自SW6.4)

- 前提条件: 删除制造商口令, “工艺”输入区= “是”
- 更改的值用于生成CYCLE84。在重新调用一个攻丝循环时机床制造商设置重新生效。

输入区“工艺”和后面的输入区在钥匙键位置0或者1时不显示。

### 深孔攻丝 \_VARI, \_DAM, \_VRT

使用参数 \_VARI 可以区分简单攻丝

(\_VARI = 0)和深孔攻丝(\_VARI ≠ 0)。

在深孔攻丝时可以区分断屑 (从当前的钻深退回可变距离, 参数 \_VRT, \_VARI = 1)

和退刀排屑 (由基准面退回 \_VARI = 2)。

该功能与标准的深孔钻削循环CYCLE83类似。

通过参数 \_DAM 给定某步增量钻削深度。循环如下内部计算中间深度:

- 编程的增量钻深在每一步中执行, 直至剩余部分直达孔底深度  $< 2 * \_DAM$
- 剩余的钻削深度一分为二, 按照两步执行; 这样最小的钻深不小于  $\_DAM/2$ 。

### 其它说明

在该循环中攻丝时旋转方向始终自动反向。

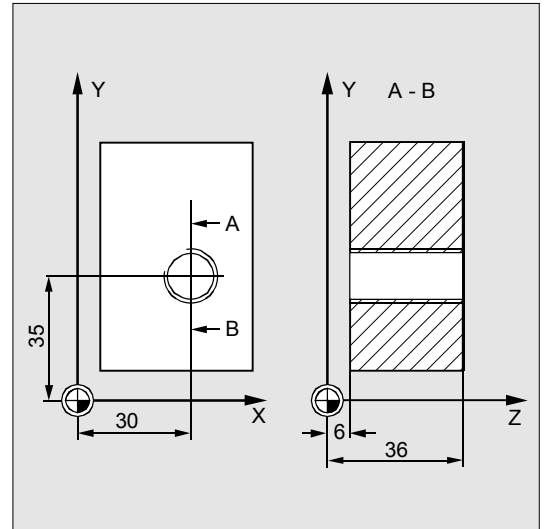


### 编程举例

#### 攻丝，不带补偿衬套

在XY层面中的位置X30

Y35上不带补偿衬套攻丝，钻削轴是Z轴。没有编程进行相对深度规定的停留时间。旋转方向和螺距参数必须赋予数值。钻削公制螺纹M5。



N10 G0 G90 T4 D1

; 确定工艺数值

N20 G17 X30 Y35 Z40

; 返回钻削位置

N30 CYCLE84 (40, 36, 2, , 30, , 3, 5, -> ; 循环调用，删除参数PIT，没有绝对深度参数，没有停留时间，主轴在90度时停止，攻丝转速为200，退回转速为500

N40 M30

程序结束

-> 必须在一个程序段中编程

## 2.1.6 攻丝，带补偿衬套 – CYCLE840



## 编程

CYCLE840 (RTP, RFP, SDIS, DP, DPR, DTB, SDR, SDAC, ENC, MPIT, PIT- AXN, \_PTAB, \_TECHNO)



## 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DTB	实数	螺纹深度停留时间：如果已编程>0，总是有效 值范围：0<=DTB
SDR	整数	退回时转动方向值： 0（旋转方向自动反向） 3 或者 4（用于 M3 或者 M4）
SDAC	整数	循环结束之后主轴方向 值：3，4或者5（用于 M3，M4 或者 M5）
ENC	整数	攻丝，带/不带编码器值： 0 = 带编码器，没有停留时间 1 = 没有编码器，在循环之前编程进给率 11 = 没有编码器，在循环中计算进给率 20 = 带编码器，有停留时间
MPIT	实数	螺距作为螺纹尺寸 值范围：3（用于 M3）..48（用于M48）
PIT	实数	螺距作为数值 值范围：0.001 ... 2000.000 毫米 如果_PTAB = 0 或者1：单位 毫米 如果_PTAB = 2，螺纹头，每英寸
_AXN	整数	工具轴： 值：1 = 1. 几何轴 2 = 2. 几何轴 其它为第三个几何轴
_PTAB	整数	求值螺距PIT值： 0...相当于编程的尺寸系统英制/公制 1...螺距单位毫米 2...螺距，每英寸的螺纹导程 3...螺距，单位英寸/转

_TECHNO	整数	工艺设定
		个位：准停特性
	值：	0...如同调用循环之前编程
		1... (G601)
		2... (G602)
		3... (G603)
		十位：预控制
	值：	0...如同调用循环之前编程
		1...带预控制 (FFWON)
		2...不带预控制 (FFWOF)
		百位：制动开始点
	值：	0...无计算
		1...带计算



### 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的螺纹深度。

使用该循环，可以带补偿衬套进行攻丝

- 不带编码器，
- 带编码器

加工。



### 工作流程

攻丝带补偿衬套，不带编码器 (ENC = 11)

循环开始之前到达的位置：

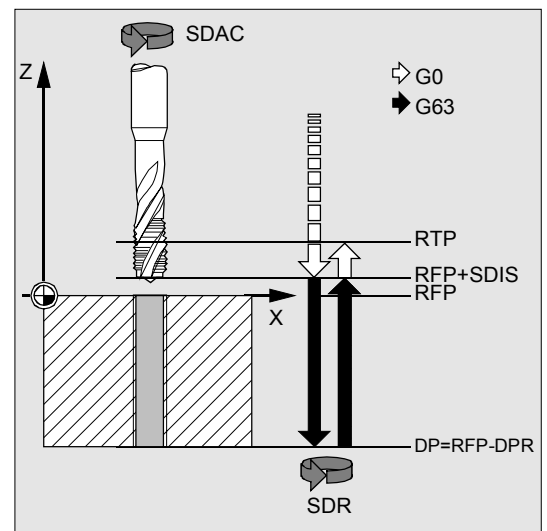
钻削位置是指所选择平面的两个轴位置。

该循环产生以下的运动过程：

- 以G0返回到提前了相隔安全距离的基准面
- 用G63进行攻丝，直至孔底深度
- 以G63退回到提前了安全距离的基准面
- 以G0返回到退回平面



主轴倍率必须位于**100%**。

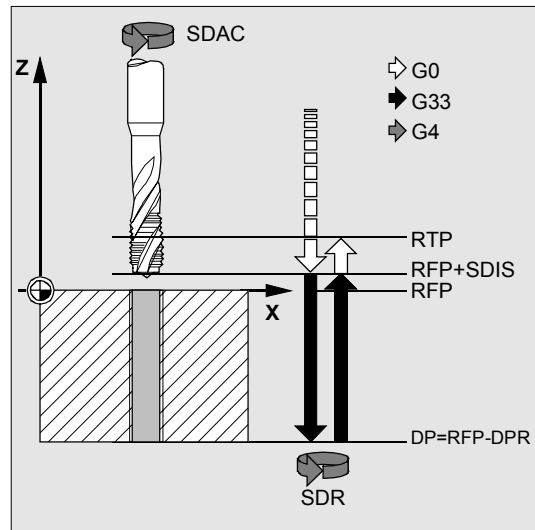


**攻丝带补偿衬套，带有编码器（ENC = 0/20）****循环开始之前到达的位置：**

钻削位置是指所选择平面的两个轴位置。

**该循环产生以下的运动过程：**

- 以G0返回到提前了相隔安全距离的基准面
- 用G33进行攻丝，直至孔底深度
- 执行螺纹深度上的停留时间（参数DTB）（仅当ENC = 20时）
- 以G33退回到提前了安全距离的基准面
- 以G0返回到退回平面

**参数说明**

参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

**DTB (停留时间)**

以秒钟为单位编程停留时间。停留时间根据选择的工艺方式在参数 ENC 中生效。

**SDR (退回时旋转方向)**

在参数 SDR

中编程用于攻丝时的退回旋转方向。对于带有编码器的攻丝，自动以主轴方向反向进行，然后将其置为 SDR = 0。

**SDAC (旋转方向)**

由于也可以模态调用循环（参见章节2.2），因此，在执行其它攻丝时需要一个旋转方向。在参数SDAC中编程该旋转方向，并与第一次调用前在上一层程序中写入的旋转方向相同。如果SDR = 0，则在循环中SDAC

下写入的值没有意义，可以在编程时删除该值。



### ENC (攻丝)

如果攻丝不带编码器进行，尽管已有一个编码器，也必须将参数 ENC 设为x1。

如果与此相反，没有编码器，并且参数为值 x0，则在循环中不考虑。

- **攻丝，不带编码器，输入螺距：**

在不带编码器（G63

螺纹）攻丝时，进给率和转速的关系通过编程的螺距在循环内部计算。该转速必须在循环调用之前编程。

在不带编码器攻丝时，螺距可以通过MPIP（公制螺距尺寸）规定，也可以选择通过PIT

（螺距作为数值）规定。

然后在循环内部由螺距和转速计算进给值。

在循环结束后上一次编程的进给值重新生效。

编程：

ENC=11，螺距在 MPIT 或者 PIT 中编程。

- **攻丝，带编码器，有停留时间：**

在带有编码器的攻丝（G33螺纹）时，可以在参数DTB中有选择性地编程一个停留时间。

该停留时间在攻丝后，退回至退回平面RTP

前生效，在带有无效的主轴运动的机床情况下需要该停留时间。

编程：

ENC=20，停留时间输入到参数 DTB 中

### MPIT 和 PIT（作为螺纹尺寸和数值）

螺距参数仅在与带有编码器的攻丝有关系时有意义。

由循环通过主轴转速和螺距计算进给值。

螺距值可以规定为螺纹尺寸（仅用于M3和M48之间的公制螺纹），也可以规定为数值（从一个螺纹线至下一个螺纹线间的距离作为数值）。不需要的参数在调用时删除或者保留值零。

如果两个螺距参数有相互矛盾的值，则由循环产生报警61001“螺距错误”，

并且中断循环的加工。

**\_AXN (刀具轴)**

在车床上使用深孔攻丝时，通过由 `_AXN` 编程钻削轴，可以省去平面 G18 到 G17 的转换。这表明：

`_AXN = 1` 1. 当前平面的第1轴

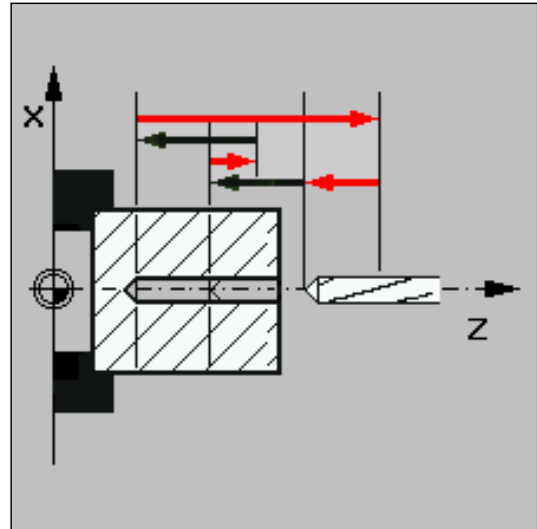
`_AXN = 2` 2. 当前平面的第二轴

`_AXN=3` 3.

当前平面的第3轴，比如，为了进行G18平面中一次定中心钻削（Z方向），您需要编程：

G18

`_AXN=1`

**\_PTAB (螺距 PIT的求值)**

参数 `_PTAB` 确定螺距的尺寸单位。

- 0 = 相对应于编程的尺寸系统英制/公制
- 1 = 螺距，以毫米为单位 (mm)
- 2 = 螺距，单位螺纹线/英寸
- 3 = 螺距，单位英寸/转

根据循环支持中不同螺纹表格的选择方法不同，需要该参数。

**\_TECHNO (工艺设定)**

使用参数 `_TECHNO` 可以设定攻丝时的工艺性能。

可能的值是：

**个位 (准停性能)：**

- 0 = 如同调用循环之前编程
- 1 = (G601)
- 2 = (G602)
- 3 = (G603)

**十位 (预控制)：**

- 0 = 如同调用循环之前编程
- 1 = 带预控制 (FFWON)
- 2 = 不带预控制 (FFWOF)

**百位 (制动开始点)：**

- 0 = 无计算
- 1 = 带计算



自软件版本 SW 6.4 起：

在“工艺”输入区按“是”，可以由机床制造商和操作人员/程序员进行攻丝时的工艺匹配。

#### 机床制造商匹配（自软件版本 SW6.4 起）

- 前提条件：设置制造商口令，“工艺”输入区=“是”
- 在打开循环 CYCLE840 输入屏幕情况下，用 GUD7 变量 \_SC\_MASK[1] 值预设参数。在更改参数时，该值直接写入在 GUD7 变量中。
- 机床制造商由此可以匹配基本设定到其机床的具体情况。

#### 操作人员/程序员匹配（自软件版本 SW6.4 起）

- 前提条件：删除制造商口令，“工艺”输入区=“是”
- 更改的值用于生成 CYCLE840。  
在重新调用一个攻丝循环时，机床制造商设置再次生效。

输入区“工艺”和后面的输入区在钥匙键位置 0 或者 1 时不显示。



#### 其它说明

循环根据机床数据 NUM\_ENCS 进行选择：  
是否带或不带编码器进行攻丝。

在循环调用之前，通过 M3 或者 M4  
编程主轴的旋转方向。

在带 G63  
的螺纹程序段中，进给率数值和主轴转速补偿开关固定在 100%。

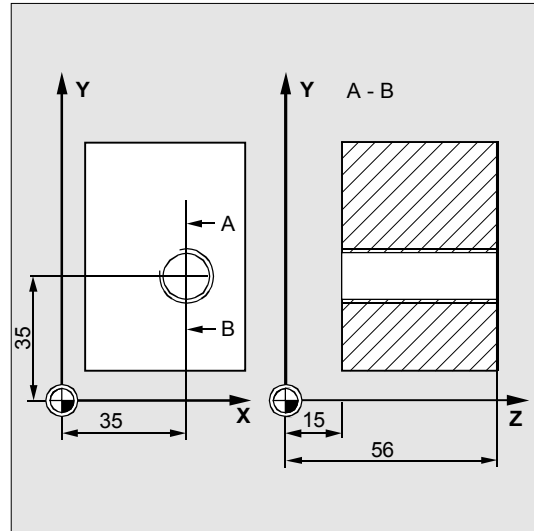
通常不带编码器的攻丝需要一个较长的补偿衬套。



### 程序举例

#### 螺纹，不带编码器

用该程序在XY平面中的位置 X35 Y35 上钻削不带编码器的螺纹，钻削轴为 Z 轴。必须规定旋转参数 SDR 和 SDAC，参数 ENC 预设为 1，深度规定为绝对值。可以删除螺距参数 PIT。使用一个补偿衬套以用于加工。



N10 G90 G0 D2 T2 S500 M3

; 确定工艺数值

N20 G17 X35 Y35 Z60

; 返回钻削位置

N30 G1 F200

; 确定轨迹进给率

N40 CYCLE840 (59, 56, , 15, , 1, 4, 3, 1); 循环调用，停留时间1s，SDR=4，  
SDAC=3，没有安全距离，参数MPIT，  
PIT不编程，也就是说螺距由自由编程的  
F和S值的关系产生。

N50 M30

; 程序结束



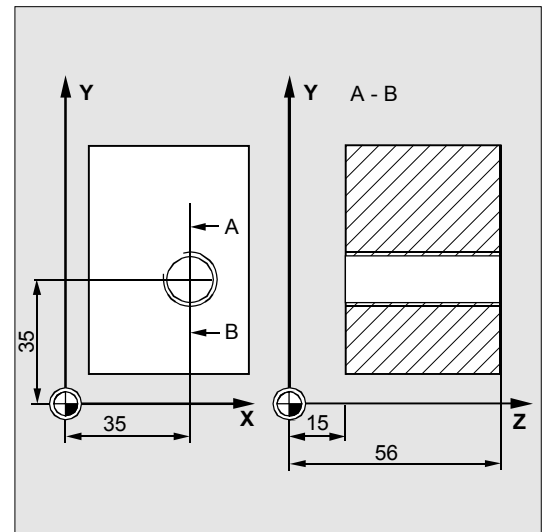
### 螺纹，带编码器

通过该程序，在 XY 平面中的位置 X35 Y35

上用编码器加工一个螺纹。钻削轴为 Z 轴。

必须规定螺距参数，编程一个自动的旋转方向转换。

使用一个补偿衬套以用于加工。



DEF INT SDR=0	; 参数定义，赋值
DEF REAL PIT=3.5	
N10 G90 G0 D2 T2 S500 M4	; 确定工艺数值
N20 G17 X35 Y35 Z60	; 返回钻削位置
N30 CYCLE840 (59, 56, , 15, , , , , ->	; 循环调用，没有安全距离，带绝对深度
->, PIT)	参数，删除SDAC, ENC, MPIT (也
	就是说值为零)。
N40 M30	; 程序结束

-> 必须在一个程序段中编程

## 2.1.7 镗孔 1 – CYCLE85



## 编程

CYCLE85 (RTP, RFP, SDIS, DP, DPR, DTB, FFR, RFF)



## 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DTB	实数	在钻削底部的停留时间（断屑）
FFR	实数	进给率
RFF	实数	退回进给



## 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。

分别以相应参数FFR和RFF中规定的进给率进行向内运动和向外运动。

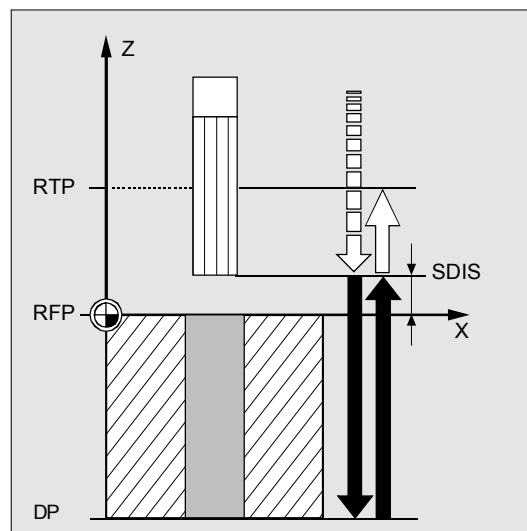
该循环可以用于铰孔（研磨）。



## 工作流程

循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。



该循环产生以下的运动过程：

- 以G0返回到提前了相隔安全距离的基准面
- 以G1和参数FFR下编程的进给率运行到孔底深度。
- 执行在钻削底部的停留时间
- 以G1和参数RFF下规定的退回进给率返回到提前了相隔安全距离的基准面
- 以G0返回到退回平面



### 参数说明



参数 RTP, RFP, SDIS, DP, DPR 参见章节 2.1.2. (钻削, 定中心 - CYCLE81)

#### DTB (停留时间)

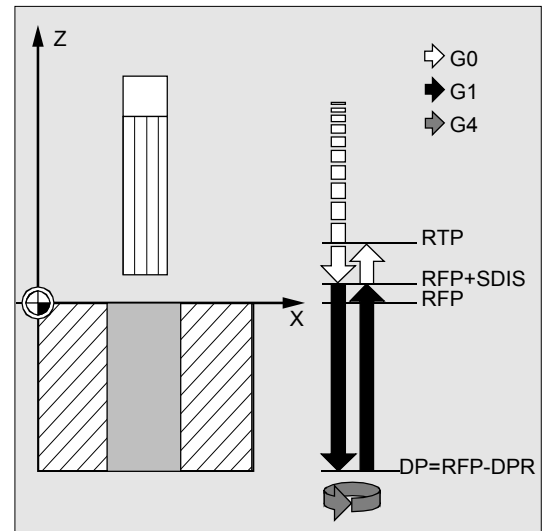
在DTB下编程钻削底部(断屑)的停留时间, 单位秒。

#### FFR (进给率)

在 FFR 下规定的进给值在钻削时生效。

#### RFF (退回进给)

在RFF下编程的进给值在从平面退回时生效。



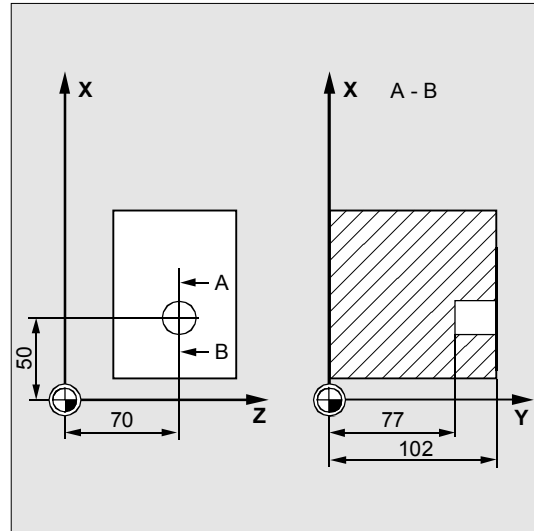
## 编程举例

## 第一次镗孔

在 ZX 平面中的 Z70 X50

上调用循环CYCLE85。钻削轴为 Y 轴。

在循环调用中相对给定最终钻削深度，不编程停留时间。工件上边缘位于 Y102 处。



```
DEF REAL FFR, RFF, RFP=102,
DPR=25,SDIS=2
```

; 参数定义, 赋值

```
N10 G0 FFR=300 RFF=1.5*FFR S500 M4
```

; 确定工艺数值

```
N20 G18 T1 D1 Z70 X50 Y105
```

; 返回钻削位置

```
N21 M6
```

```
N30 CYCLE85 (RFP+3, RFP, SDIS, , DPR, , -> ; 循环调用, 没有编程停留时间
-> FFR, RFF)
```

```
N40 M30
```

; 程序结束

-> 必须在一个程序段中编程



### 2.1.8 镗孔 2 – CYCLE86



#### 编程

CYCLE86 (RTP, RFP, SDIS, DP, DPR, DTB, SDIR, RPA, RPO, RPAP, POSS)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DTB	实数	在钻削底部的停留时间（断屑）
SDIR	整数	旋转方向值： 3（用于 M3） 4（用于 M4）
RPA	实数	在有效平面横坐标上的退回位移（增量，输入符号）
RPO	实数	在有效平面纵坐标上的退回位移（增量，输入符号）
RPAP	实数	在应用轴上的退回位移（增量，带符号输入）
POSS	实数	主轴位置，用于循环中定向主轴停（单位度）



#### 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。

在镗孔 2 时，在达到钻削深度后通过 SPOS 指令进行一个定向的主轴停止。接着，以快速移动运行至编程的退回位置，并自那里行进直至退回平面。



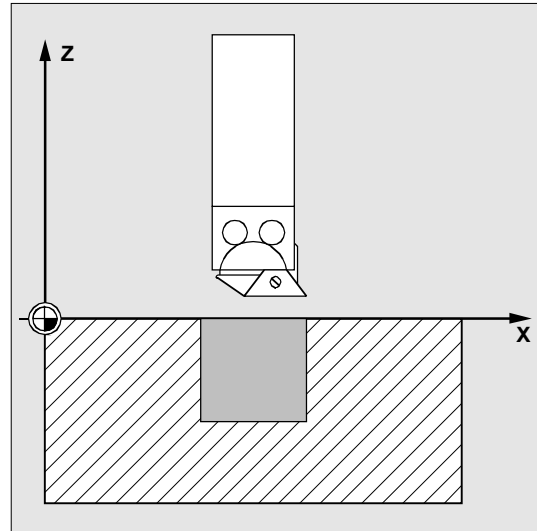
## 工作流程

### 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

### 该循环产生以下的运动过程：

- 以G0返回到提前了相隔安全距离的基准面
- 以G1和循环调用之前编程的进给率运行到孔底深度
- 执行在钻削底部的停留时间
- 在POSS下编程的主轴位置上定向主轴停止
- 以G0在最多3个轴上运行退回位移
- 以G0退回到提前了安全距离的基准面
- 以G0退回到退回平面（在平面中两个轴上的开始钻削位置）



## 参数说明

参数 RTP, RFP, SDIS, DP, DPR 参见章节 2.1.2. (钻削, 定中心 - CYCLE81)

### DTB (停留时间)

在DTB下编程钻削底部 (断屑) 的停留时间, 单位秒。

### SDIR (旋转方向)

用此参数您可以确定循环中钻孔的方向。如果不是值3或者4 (M3/M4), 则发出报警 61102

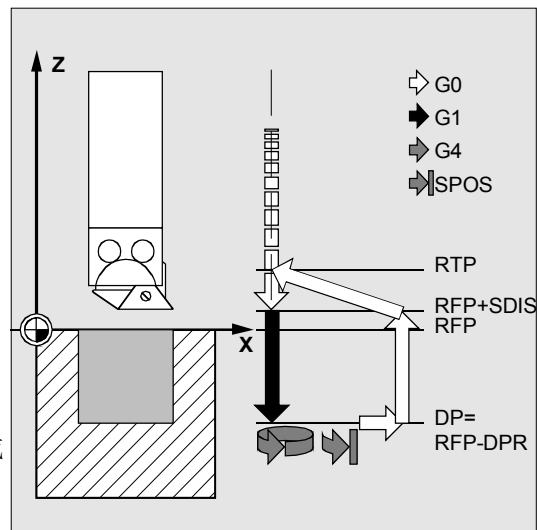
“没有编程主轴方向”并且不执行该循环。

### RPA (退回位移, 横坐标)

在此参数下定义横坐标上的退回运动, 在到达孔底深度和定向主轴停止之后执行该退回运动。

### RPO (退回位移, 纵坐标)

在此参数下定义纵坐标上的退回运动, 在到达孔底深度和定向主轴停止之后执行该退回运动。



**RPAP (退回位移, 应用轴)**

在此参数下定义钻削轴上的退回运动，在到达孔底深度和定向主轴停止之后执行该退回运动。

**POSS (主轴位置)**

在到达孔底深度之后，必须在POSS下编程定向主轴停止时的主轴位置，单位度。

**其它说明****通过 SPOS**

指令可保持有效的主轴持续定向。通过传输参数编程相应的角度值。



*如果用于钻削的主轴进入位置控制运行状态，则可以使用循环CYCLE86。*

**车床，没有Y轴**

循环 CYCLE86 现在可用于不带 Y 轴的车床上。那么，仅以两根轴运行退回至钻削深度。如果已编程一个退回位移用于第三根轴，则此项忽略不记。如果在调用循环时在G18平面中没有 Y 轴，则发出报警：61005 “没有第三根几何轴”，因为 Y 轴有可能是钻削轴。



### 编程举例

#### 第二次镗孔

在 XY 平面中在循环 CYCLE86 中的位置 X70

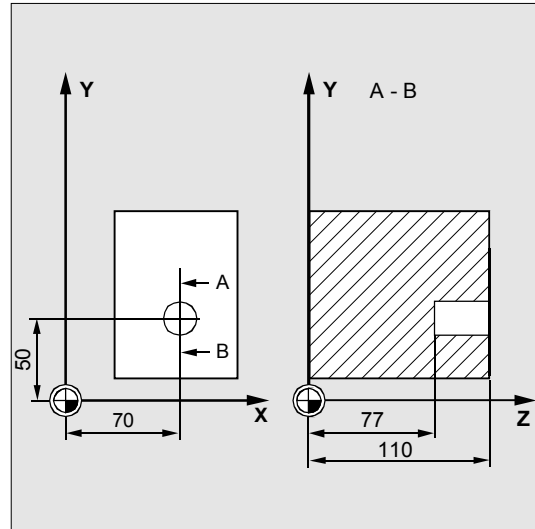
Y50进行调用。钻削轴为 Z 轴。

以绝对值编程最终钻削深度，未规定一个安全距离。

最终钻削深度上的停留时间为2 秒。

工件上边缘位于 Z110 处。在循环中，要求以 M3

车削主轴并在 45 度角处停止。



DEF REAL DP, DTB, POSS	; 参数定义
N10 DP=77 DTB=2 POSS=45	; 赋值
N20 G0 G17 G90 F200 S300	; 确定工艺数值
N30 D1 T3 Z112	; 返回退回平面
N40 X70 Y50	; 返回钻削位置
N50 CYCLE86 (112, 110, , DP, , DTB, 3, -> -> -1, -1, +1, POSS)	; 循环调用, 带绝对钻深
N60 M30	; 程序结束

-> 必须在一个程序段中编程

### 2.1.9 镗孔 3 – CYCLE87



#### 编程

CYCLE87 (RTP, RFP, SDIS, DP, DPR, SDIR)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
SDIR	整数	旋转方向 值： 3（用于 M3） 4（用于 M4）



#### 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。

镗孔 3 时，在到达最终钻削深度后，在没有定向 M5 的情况下主轴停止，接着产生编程的停止 M0。

通过 NC

启动键继续以快速移动向外运动，直至到达退回平面。



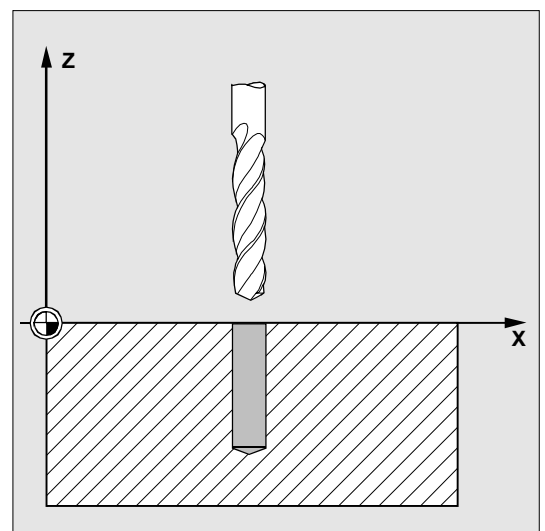
#### 工作流程

##### 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

##### 该循环产生以下的运动过程：

- 以 G0 返回到提前了相隔安全距离的基准面
- 以 G1 和循环调用之前编程的进给率运行到孔底深度
- 主轴停止，带 M5
- 按 NC 启动键
- 以 G0 返回到退回平面





### 参数说明

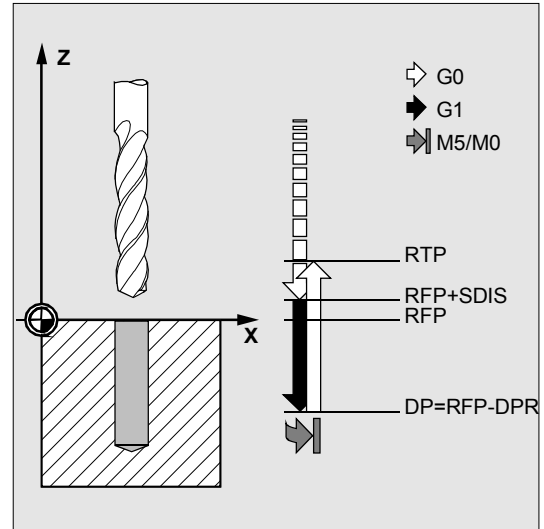


参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

#### SDIR (旋转方向)

用此参数您可以确定循环中钻孔的方向。

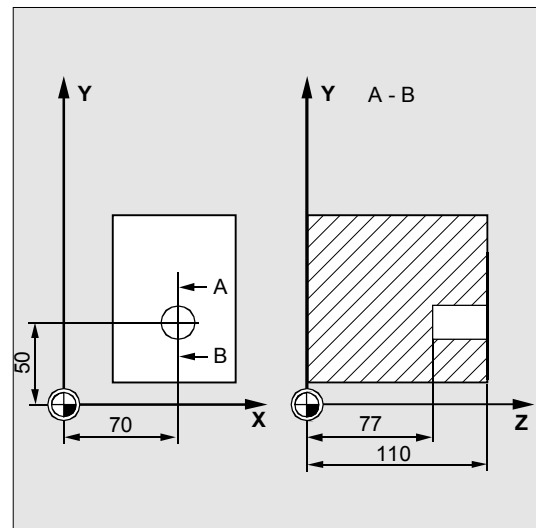
如果不是值3或者4 (M3/M4), 则发出报警61102  
“没有编程主轴转向”, 并且循环停止。



### 程序举例

#### 第三次镗孔

在 XY 平面中的 X70 Y50 上调用循环 CYCLE87。  
钻削轴为 Z 轴。以绝对值规定最终钻削深度。  
安全距离为 2 毫米。



DEF REAL DP, SDIS	; 参数定义
N10 DP=77 SDIS=2	; 赋值
N20 G0 G17 G90 F200 S300	; 确定工艺数值
N30 D1 T3 Z113	; 返回退回平面
N40 X70 Y50	; 返回钻削位置
N50 CYCLE87 (113, 110, SDIS, DP, , 3)	; 调用循环, 带编程的主轴转向M3
N60 M30	; 程序结束

### 2.1.10 镗孔 4 – CYCLE88



#### 编程

CYCLE88 (RTP, RFP, SDIS, DP, DPR, DTB, SDIR)



#### 参数

RTP	实数	退回平面（绝对）		
RFP	实数	基准面（绝对）		
SDIS	实数	安全距离（不输入符号）		
DP	实数	孔底深度（绝对）		
DPR	实数	相对于基准面的钻削深度（不输入符号）		
DTB	实数	在钻削底部的停留时间		
SDIR	整数	旋转方向	值： 3（用于 M3）	4（用于 M4）



#### 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。镗孔 4 时，在到达最终钻削深度后，在没有定向 M5 的情况下主轴停止，并产生编程的停止 M0。通过操作 NC 启动键继续以快速移动向外运动，直至到达退回平面。



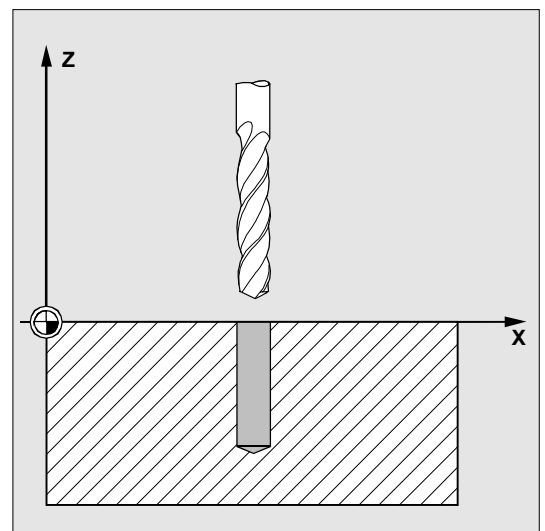
#### 工作流程

##### 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

##### 该循环产生以下的运动过程：

- 以 G0 返回到提前了相隔安全距离的基准面
- 以 G1 和循环调用之前编程的进给率运行到孔底深度
- 在钻削底部的停留时间
- 主轴停止，带 M5（ $\_ZSD[5]=1$ ），或者
- 带有 M5 M0 的主轴和程序停止（ $ZSD[5]=0$ ）。程序停止后，按 NC 启动键。
- 以 G0 返回到退回平面





### 参数说明



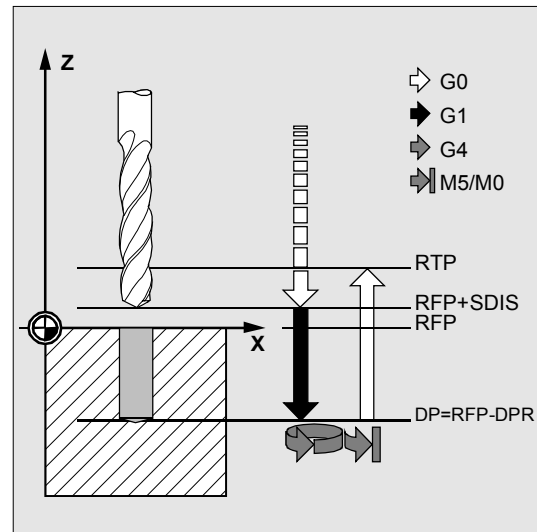
参数 RTP, RFP, SDIS, DP, DPR 参见章节 2.1.2. (钻削, 定中心 - CYCLE81) 循环设定参数\_ZSD[5]参见章节3.2.

#### DTB (停留时间)

在DTB下编程钻削底部(断屑)的停留时间, 单位秒。

#### SDIR (旋转方向)

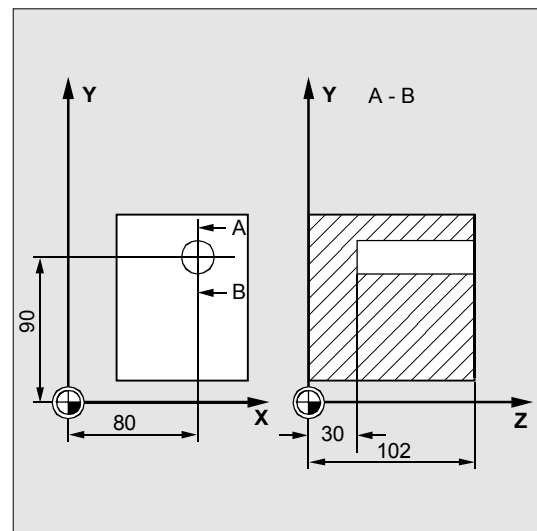
编程的旋转方向在到孔底深度的运行位移中有效。如果不是值3或者4 (M3/M4), 则发出报警61102 “没有编程主轴转向” 并且循环停止。



### 编程举例

#### 第四次镗孔

在XY平面中的X80 Y90上调用循环CYCLE88。钻削轴为Z轴。安全距离编程为3毫米, 规定钻削深度相对于基准面。在循环中M4起作用。



DEF REAL RFP, RTP, DPR, DTB, SDIS	; 参数定义
N10 RFP=102 RTP=105 DPR=72 DTB=3 SDIS=3	; 赋值
N20 G17 G90 T1 D1 F100 S450	; 确定工艺数值
N21 M6	
N30 G0 X80 Y90 Z105	; 返回运行到钻削位置
N40 CYCLE88 (RTP, RFP, SDIS, , DPR, -> -> DTB, 4)	; 用编程的 ; 主轴转速 M4调用循环
N50 M30	; 程序结束

-> 必须在一个程序段中编程



## 2.1.11 镗孔 5 – CYCLE89



## 编程

CYCLE89 (RTP, RFP, SDIS, DP, DPR, DTB)



## 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DTB	实数	在钻削底部的停留时间（断屑）



## 功能

刀具以编程的主轴转速和进给速度钻削，直至输入的钻削深度。如果已到达最终钻削深度，则可编程一个停留时间。



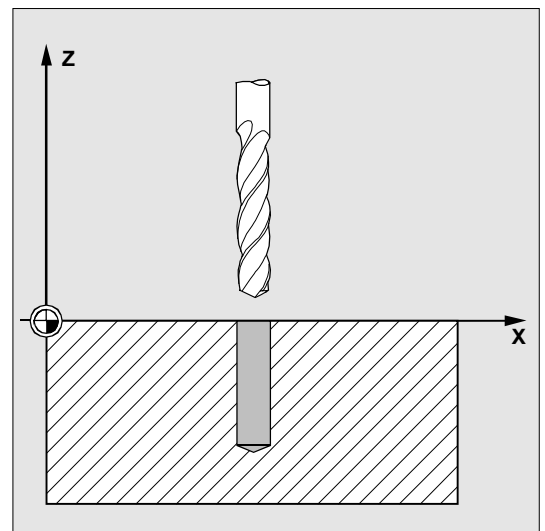
## 工作流程

## 循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

## 该循环产生以下的运动过程：

- 以G0返回到提前了相隔安全距离的基准面
- 以G1和循环调用之前编程的进给率运行到孔底深度
- 执行在钻削底部的停留时间
- 以G1和相同的进给值返回到提前了安全距离的基准面
- 以G0返回到退回平面





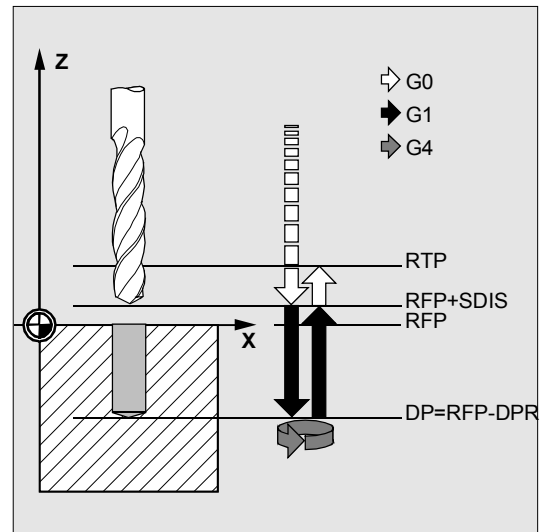
### 参数说明



参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

### DTB (停留时间)

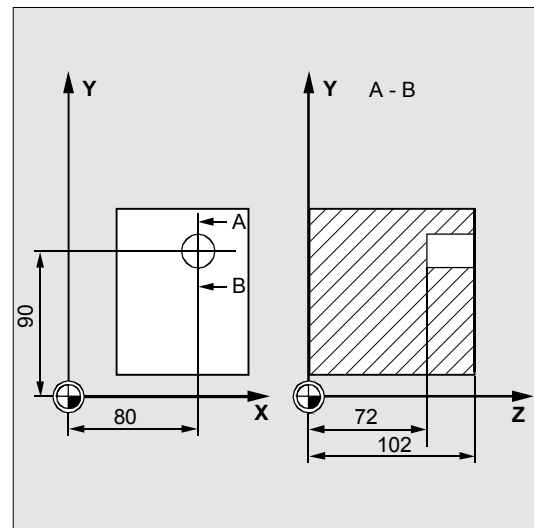
在DTB下编程钻削底部(断屑)的停留时间,  
单位秒。



### 编程举例

#### 第五次镗孔

在XY平面中位置 X80Y90 上调用钻削循环  
CYCLE89, 带一个 5 毫米的安全距离,  
并且以绝对值说明孔底深度。钻削轴为 Z 轴。



DEF REAL RFP, RTP, DP, DTB	; 参数定义
RFP=102 RTP=107 DP=72 DTB=3	; 赋值
N10 G90 G17 F100 S450 M4	; 确定工艺数值
N20 G0 T1 D1 X80 Y90 Z107	; 返回运行到钻削位置
N21 M6	
N30 CYCLE89 (RTP, RFP, 5, DP, , DTB)	; 循环调用
N40 M30	; 程序结束

## 2.2 模态调用钻削循环

在NC编程中可以以模态调用每个任意的子程序，也就是说自保持。

这在钻削循环中特别重要。



### 编程

模态调用一个子程序

MCALL

带钻削循环（举例）

MCALL CYCLE81 (RTP, RFP, SDIS, DP, DPR)



### 功能

在NC编程中可以以模态调用子程序和循环，也就是说自保持。

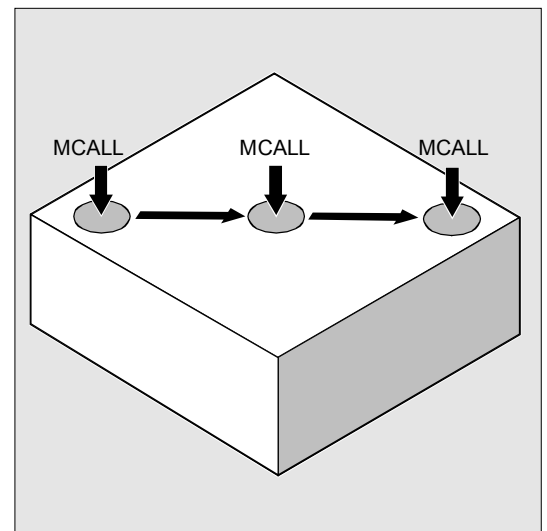
通过子程序名称前的关键字 **MCALL**（模态 UP 调用）模态调用一个子程序。通过该功能，以轨迹运动自动调用和处理每个程序段后面的子程序。

如果编程

**MCALL**，后面不带子程序名，则可以关闭该功能；或者通过重新模态调用一个其它的子程序，也可以改变该功能。

模态调用的嵌套是不可以的，这就是说模态调用的子程序中不可以在内部包含其它的模态子程序调用。

模态可调用的钻削循环的个数可以任意大，不受所预留 **G** 功能个数的限制。

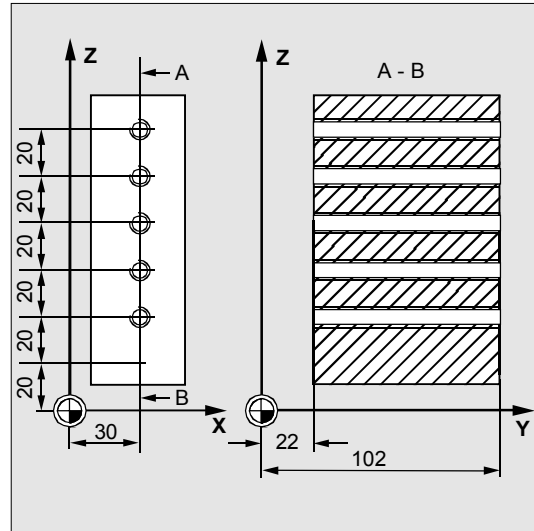




### 编程举例

#### 成排孔\_5

通过该程序可由 5 个平行于 ZX 平面中的 Z 轴的攻丝加工成排孔。钻孔相互之间的间距为 20 毫米。成排孔的出发点位于 Z20 X30 处，第一个钻孔和该点的距离为 20 毫米。这种情况下，在不使用循环的情况下描述成排孔的几何形状。首先用循环 CYCLE81 对其进行钻削，然后用 CYCLE84 钻削螺纹（不带补偿衬套）。钻孔深度为 80 毫米。这等于基准面和最终钻削深度之间的差值。



DEF REAL RFP=102, DP=22, RTP=105, ->	; 参数定义
-> PIT=4.2, SDIS	; 赋值
DEF INT ZAEHL=1	
N10 SDIS=3	; 安全距离值
N20 G90 F300 S500 M3 D1 T1	; 确定工艺数值
N30 G18 G0 Y105 Z20 X30	; 返回运行到出发位置
N40 MCALL CYCLE81 (RTP, RFP, SDIS, DP)	; 模态调用钻削循环
N50 MA1: G91 Z20	; 返回到下一个位置 (ZX平面), 执行循环
N60 ZAEHL=ZAEHL+1	; 成排孔钻削位置环
N70 IF ZAEHL<6 GOTOB MA1	
N80 MCALL	; 撤消选择模态调用
N90 G90 Y105 Z20	; 再次返回运行到出发位置
N100 ZAEHL=1	; 计数器设为零
N110 ...	; 换刀
N120 MCALL CYCLE84 (RTP, RFP, SDIS, ->	; 模态调用攻丝循环
-> DP , , , 3, , PIT, , 400)	
N130 MA2: G91 Z20	; 下一个钻削位置
N140 ZAEHL=ZAEHL+1	; 成排孔钻削位置环
N150 IF ZAEHL<6 GOTOB MA2	
N160 MCALL	; 撤消选择模态调用
N170 G90 X30 Y105 Z20	; 再次返回运行到出发位置
N180 M30	; 程序结束

-> 必须在一个程序段中编程



### 其它说明

#### 该示例的说明

程序段N80中撤消模态调用是必要的，因为其后必须返回到一个没有钻削的位置。

在这种形式的加工中，很有必要在一个子程序中量取钻削位置，该子程序在MA1或者MA2处调用。



在描述该钻削图循环时，程序再次以合适的，并且也因此简化的形式使用该循环。

在随后的章节2.3中所描述的钻削图循环以该调用原则为基础

MCALL BOHRZYKLUS (...)

BOHRBILD (...).

## 2.3 钻削图循环

钻削图循环仅描述平面中由钻孔排列的几何形状。  
通过在编程钻削图循环之前模态调用（参见  
章节 2.2）钻削循环来建立与该钻削循环的关系。

### 2.3.1 前提条件

#### 钻削图循环，不带钻削循环调用

该钻削图循环也可以用于其它的应用，不用事先模态调用一个钻削循环，因为钻削图循环设定参数时不要求用于钻削循环的参数。

但如果在调用钻削图循环之前没有模态调用子程序，则显示故障信息 62100 “没有钻削循环有效”。

可通过故障清除键确认该故障信息并继续通过 NC 启动键进行程序处理。随后，钻削循环依次自输入数据达到的位置返回运行，没有在该点上调用子程序。

#### 个数参数零时的性能

必须在某个钻削图中编程钻孔个数。

如果循环调用时的数量参数值为零（或者已在参数列表中删除该值），则发出报警 61103 “孔数为零”并停止循环的执行。

#### 在输入参数中参数值范围受限制时进行检查

一般来说在钻削图循环中，不对参数进行奇偶性检查，如果该参数没有明确说明如何进行相应的应答。

### 2.3.2 成排孔- HOLES1



#### 编程

HOLES1 (SPCA, SPCO, STA1, FDIS, DBH, NUM)



#### 参数

SPCA	实数	直线上参考点横坐标（绝对值）
SPCO	实数	参考点纵坐标（绝对值）
STA1	实数	与横坐标的夹角 值范围： $-180 < STA1 \leq 180$ 度
FDIS	实数	第一个钻孔与参考点的距离（不输入符号）
DBH	实数	两个钻孔之间的距离（不输入符号）
NUM	整数	钻孔个数



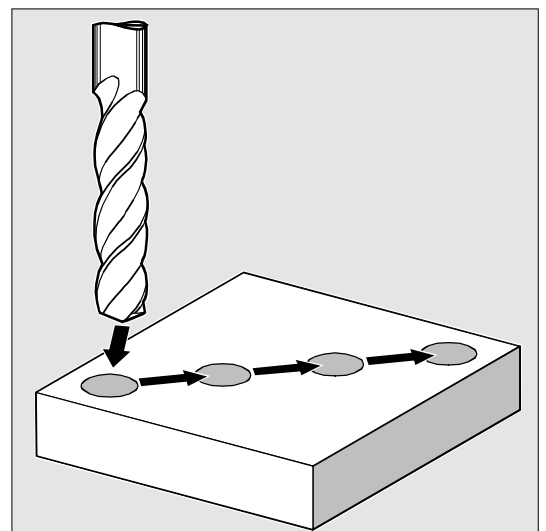
#### 功能

用该循环可加工一个成排孔，即一系列位于直线或者孔格网上的钻孔。钻孔的方式通过事先模态选择的钻削循环确定。



#### 工作流程

为了避免不必要的空行程，在循环内部可以利用平面中的轴的实际位置和成排孔的几何关系决定成排孔是否以第一个孔或者最后一个孔开始加工。然后再以快速移动依次返回到钻削位置。





### 参数说明

#### SPCA 和 SPCO (基准点横坐标和纵坐标)

将成排孔直线上的点规定为用于确定钻孔之间间距的基准点。自该点开始规定到第一个钻孔 FDIS 的距离。

#### STA1 (角度)

直线可接受平面中的任意一个位置。除了由 SPCA 和 SPCO 定义的点外，其还通过一个由直线与调用当前工件坐标系时闭合的横坐标构成的夹角确定。

在 STA1 下以度为单位输入角度。

#### FDIS 和 DBH (距离)

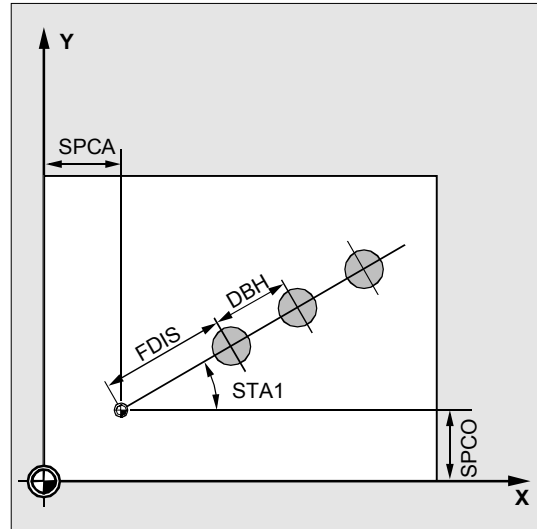
在 FDIS 下规定第一个钻孔到 SPCA 和 SPCO

下定义的基准点的距离。参数 DBH

包含每两个钻孔之间的间距。

#### NUM (个数)

通过参数 NUM 可以确定钻孔个数。







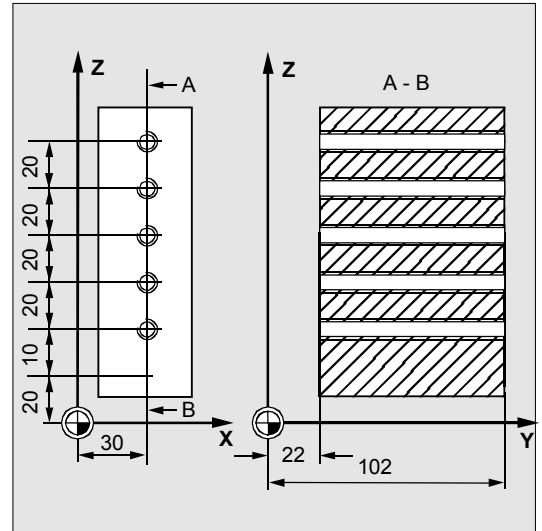
### 编程举例

#### 成排孔

通过该程序可由 5 个平行于 ZX 平面中的 Z 轴且相互之间的距离为 20 毫米的攻丝加工成排孔。成排孔的出发点位于 Z20 X30 处，第一个钻孔和该点的距离为 10 毫米。

通过循环 HOLES1 描述成排孔的几何形状。首先用循环 CYCLE81 对其进行钻削，然后用 CYCLE84 钻削螺纹（不带补偿衬套）。

钻孔深度为 80 毫米（基准面和最终钻削深度之间的差值）。



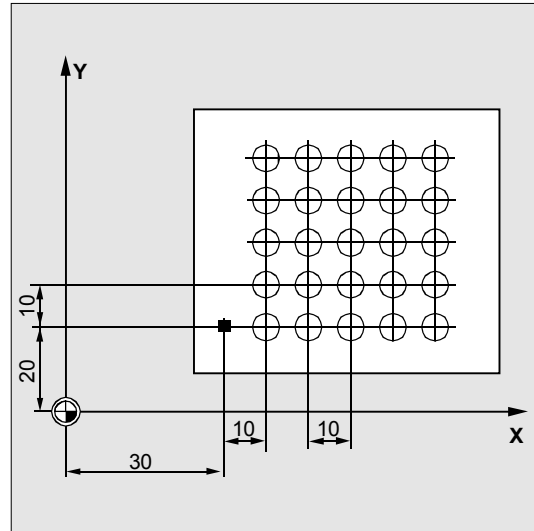
DEF REAL RFP=102, DP=22, RTP=105	; 参数定义, 赋值
DEF REAL SDIS, FDIS	
DEF REAL SPCA=30, SPCO=20, STA1=0, ->	
-> DBH=20	
DEF INT NUM=5	
N10 SDIS=3 FDIS=10	; 安全距离的值和第一个孔到参考点之间的距离值
N20 G90 F30 S500 M3 D1 T1	; 确定工艺数值, 用于加工截面
N30 G18 G0 Z20 Y105 X30	; 返回运行到出发位置
N40 MCALL CYCLE81 (RTP, RFP, SDIS, DP)	; 模态调用钻削循环
N50 HOLES1 (SPCA, SPCO, STA1, FDIS, ->	; 调用成排孔循环, 从第一个钻孔开始, 循环中仅返回钻削位置。
-> DBH, NUM)	
N60 MCALL	; 撤消选择模态调用
...	; 换刀
N70 G90 G0 Z30 Y75 X105	; 返回到第5个钻孔旁的位置
N80 MCALL CYCLE84 (RTP, RFP, SDIS, DP, ->	; 模态调用攻丝循环
-> , 3, , 4.2, , , 400)	
N90 HOLES1 (SPCA, SPCO, STA1, FDIS, ->	; 调用成排孔循环, 从第5个钻孔开始
-> DBH, NUM)	
N100 MCALL	; 撤消选择模态调用
N110 M30	; 程序结束
-> 必须在一个程序段中编程	



## 编程举例

## 孔格网

使用该程序您可以加工一个孔格网，它由5行5列钻孔组成，位于XY平面，相互间的间距为10毫米。孔格网的出发点位于X30Y20处。



```

DEF REAL RFP=102, DP=75, RTP=105, SDIS=3 ; 参数定义, 赋值
DEF REAL SPCA=30, SPCO=20, STA1=0, ->
-> DBH=10, FDIS=10
DEF INT NUM=5, ZEILNUM=5, ZAEL=0
DEF REAL ZEILABST
N10 ZEILABST=DBH ; 行间距=孔间距
N20 G90 F300 S500 M3 D1 T1 ; 确定工艺数值
N30 G17 G0 X=SPCA-10 Y=SPCO Z105 ; 返回运行到出发位置
N40 MCALL CYCLE81 (RTP, RFP, SDIS, DP) ; 模态调用钻削循环
N50 MARKE1: HOLES1 (SPCA, SPCO, STA1, -> ; 调用孔圆弧循环
-> FDIS, DBH, NUM)
N60 SPCO=SPCO+ZEILABST ; 参考点的纵坐标, 用于下一行
N70 ZAEL=ZAEL+1 ; 如果条件满足, 则跳转到MARKE1
N80 IF ZAEL<ZEILNUM GOTOB MARKE1
N90 MCALL ; 撤消选择模态调用
N100 G90 G0 X=SPCA-10 Y=SPCO Z105 ; 返回运行到出发位置
N110 M30 ; 程序结束

```

-> 必须在一个程序段中编程

### 2.3.3 孔圆弧 – HOLES2



#### 编程

HOLES2 (CPA, CPO, RAD, STA1, INDA, NUM)



#### 参数

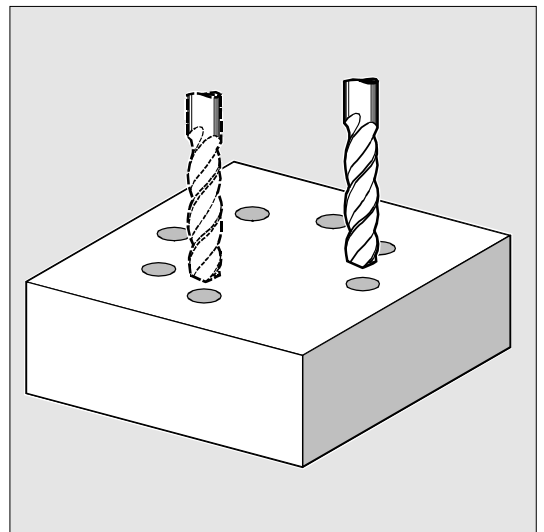
CPA	实数	圆弧孔的圆心，横坐标（绝对）
CPO	实数	孔圆弧圆心，纵坐标（绝对）
RAD	实数	圆弧孔半径（不输入符号）
STA1	实数	起始角 值范围：-180<STA1<=180度
INDA	实数	增量角度
NUM	整数	钻孔个数



#### 功能

借助该循环可加工一个孔圆弧。必须在调用循环之前确定加工平面。

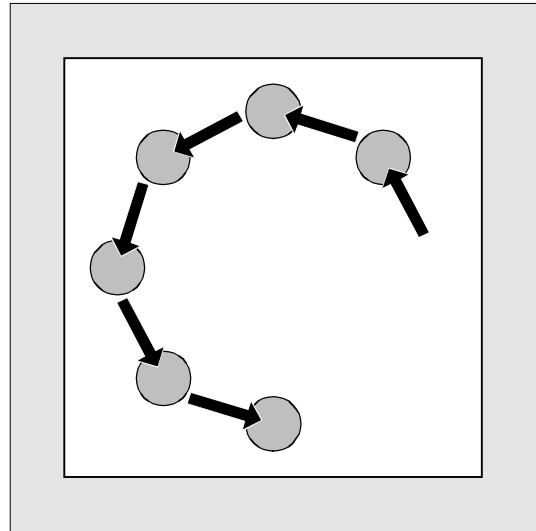
钻孔的方式通过事先模态选择的钻削循环确定。





### 工作流程

在循环中，以G0在平面中一个接一个返回运行到钻削位置（圆弧）。



### 参数说明

#### CPA, CPO 和

#### RAD（圆心和半径，横坐标，纵坐标）

通过圆心（参数 CPA 和 CPO）和半径（参数 RAD）定义加工平面中的孔圆弧位置。半径仅允许为正值。

#### STA1 和 INDA（起始角和增量角）

通过这两个参数可以确定孔圆弧上的钻孔排列。

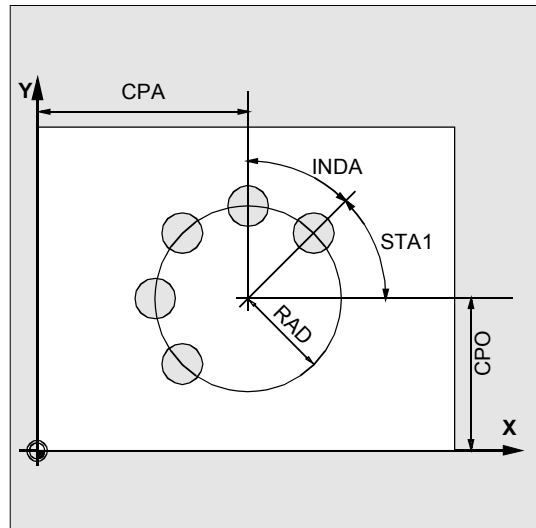
#### 参数

STA1说明循环调用之前实际工件坐标系横坐标的正方向与第一个钻孔之间的旋转角。参数INDA包含从一个钻孔到下一个钻孔之间的旋转角。

如果参数INDA的值为零，则在循环内部增量角由钻孔的个数计算，这些钻削个数均匀地分步在该圆弧上。

#### NUM（个数）

参数NUM确定钻孔个数。

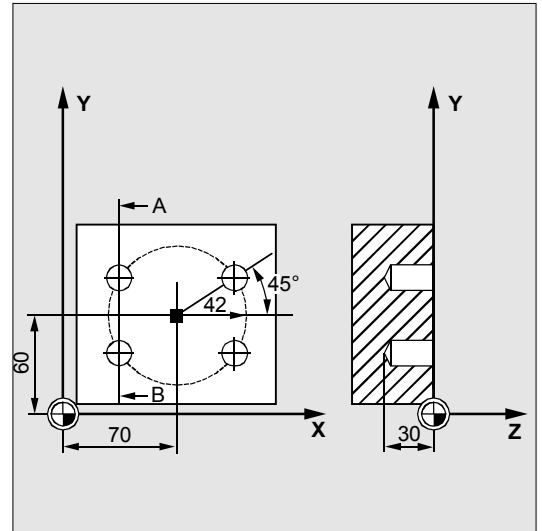




### 编程举例

#### 孔圆弧

通过该程序，使用循环 CYCLE82 4 加工深度为 30 毫米的钻孔。规定最终钻削深度为相对于基准面。  
通过圆心 X70 Y60 和半径 42 mm 在 XY 平面中确定圆弧。起始角为 45 度。  
在钻削轴 Z 中安全距离为 2 毫米。



DEF REAL CPA=70,CPO=60,RAD=42,STA1=45	; 参数定义, 赋值
DEF INT NUM=4	
N10 G90 F140 S710 M3 D1 T40	; 确定工艺数值
N20 G17 G0 X50 Y45 Z2	; 返回运行到出发位置
N30 MCALL CYCLE82 (2, 0,2, , 30)	; 模态调用钻削循环, 没有停留时间, DP没有编程
N40 HOLES2 (CPA, CPO, RAD, STA1, , NUM)	; 调用孔圆弧, 增量角在循环中计算, 因为 参数INDA已经删除
N50 MCALL	; 撤消选择模态调用
N60 M30	; 程序结束

### 2.3.4 点栅格 – CYCLE801



#### 编程

```
CYCLE801 (_SPCA, _SPCO, _STA, _DIS1,
          _DIS2, _NUM1, _NUM2)
```



#### 参数

_SPCA	实数	点栅格参考点，第一轴横坐标（绝对值）
_SPCO	实数	点栅格参考点，第二轴纵坐标（绝对值）
_STA	实数	与横坐标的夹角
_DIS1	实数	列间距（没有符号）
_DIS2	实数	行间距（没有符号）
_NUM1	整数	列数
_NUM2	整数	行数



#### 功能

用循环 CYCLE801 可加工钻削模型“孔网格”。  
钻孔的方式通过事先模态选择的钻削循环确定。



#### 工作流程

循环在内部确定钻孔顺序时，要使得在此之间产生的空行程尽可能小。利用平面中在调用之前到达的最后位置确定加工的起始位置。

起始位置为四个可能的角度位置中的一个。



### 参数说明

#### **\_SPCA** 和 **\_SPCO** (参考点横坐标和纵坐标)

这两个参数确定孔格网的第一个点。自该点出发规定行间距和列间距。

#### **\_STA** (角度)

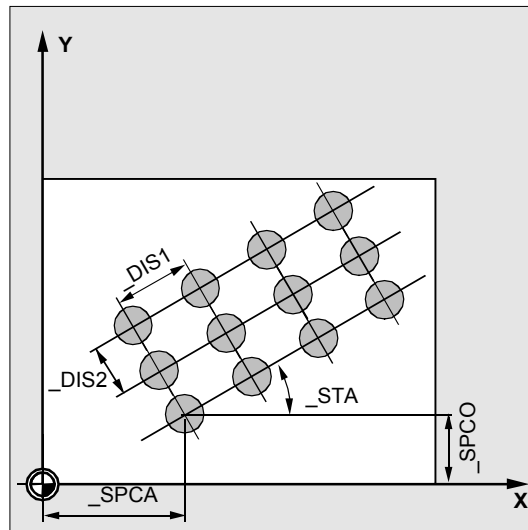
孔格网可位于平面中的任意一个角度下。在 **\_STA** 下以度为单位编程该角度，并以调用当前工件坐标系时的横坐标为基准。

#### **\_DIS1** 和 **\_DIS2** (列间距和行间距)

输入距离，不输入符号。要避免不必要的空行程，通过比较距离尺寸，以行方式或列方式处理点栅格。

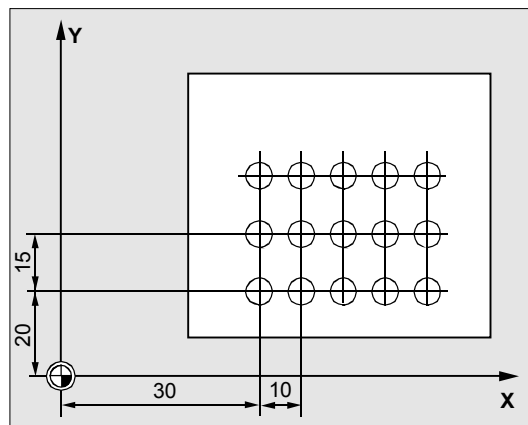
#### **\_NUM1** 和 **\_NUM2** (个数)

使用该参数确定列数或者行数。



### 编程举例

通过循环 **CYCLE801** 加工以3行和5列排列的15个钻孔组成的点栅格。为此首先模态调用所属的钻削程序。



N10 G90 G17 F900 S4000 M3 T2 D1	: 确定工艺数值
N15 MCALL CYCLE82 (10,0,1,-22,0,0)	: 模态调用钻削循环
N20 CYCLE801 (30,20,0,10,15,5,3)	: 调用点栅格
N25 M30	: 程序结束

用于记录



## 铣削循环

铣削循环	105
3.1 一般提示	107
3.2 前提条件	108
3.3 螺纹铣削CYCLE90	110
3.4 一个圆弧上的长方形孔 - LONGHOLE	117
3.5 一个圆弧上的键槽 - SLOT1	122
3.6 环形槽 - SLOT2	130
3.7 铣削矩形槽 - POCKET1	136
3.8 铣削环形凹槽 - POCKET2	140
3.9 铣削矩形凹槽 - POCKET3	144
3.10 铣削环形凹槽 - POCKET4	153
3.11 平面铣削 - CYCLE71	158
3.12 轨迹铣削 - CYCLE72	164
3.13 铣削矩形轴颈- CYCLE76	174
3.14 铣削环形轴颈 - CYCLE77	179
3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75	183
3.15.1 传送凹槽边缘轮廓- CYCLE74	184
3.15.2 传送中心岛轮廓- CYCLE75	186
3.15.3 轮廓编程	187
3.15.4 带中心岛的凹槽铣削 - CYCLE73	189
3.16 回转- CYCLE800	211
3.16.1 操作、参数和输入屏幕	213
3.16.2 操作说明，编程说明	217
3.16.3 参数	218
3.16.4 设置刀具（自循环SW 6.5 起）	222
3.16.5 调试带有回转加工平面的工件（自循环 SW 6.5 起）	223
3.16.6 操作，参数“在 JOG 方式下回转”（自循环 SW 6.5 起）	227
3.16.7 回转循环CYCLE800 开机调试	230
3.16.8 用户循环 TOOLCARR.SPF	251
3.16.9 故障信息	255
3.17 高速设定 - CYCLE832（自循环 SW 6.3 起）	257
3.17.1 在 HMI 菜单树中调用 CYCLE832	260
3.17.2 参数	263
3.17.3 匹配工艺	264
3.17.4 匹配附加的程序参数CYC_832T	266
3.17.5 接口	268
3.17.6 故障信息	269
3.18 模腔循环CYCLE60（自循环 SW 6.4 起）	270



### 3.1 一般提示

在下面的章节中，说明如何编程铣削循环。

要求将该章节作为选择循环时的指南并使用所提供的各种参数。除了各个循环功能的详尽描述和为此所列出的参数外，您还可在各节结束处找到能简化循环过程的编程举例。

各节是按照以下的原则进行组合的：

- 编程
- 参数
- 功能
- 工作流程
- 参数说明
- 其它说明
- 编程举例

上述各点中，编程和参数两节为熟练的用户所准备，而作为初学者，有关循环编程的必要知识均可以在功能、工作流程、参数说明、其它说明和编程举例中找到。

## 3.2 前提条件

### 控制系统中所需要的程序

铣削循环在内部调用以下程序

- MELDUNG.SPF 和
- STEIGUNG.SPF

作为子程序。此外您还需要数据块 GUD7.DEF和宏指令定义文件 SMAC.DEF。

在执行该铣削循环之前，把这些数据块装载到控制系统的零件程序存储器中。

### 调用条件和返回条件

编程铣削循环与具体的轴名称无关。调用铣削循环前必须激活刀具补偿。

如果在铣削循环中没有提供参数，则必须在零件程序中编程相应的数值用于进给、主轴转速和主轴方向。

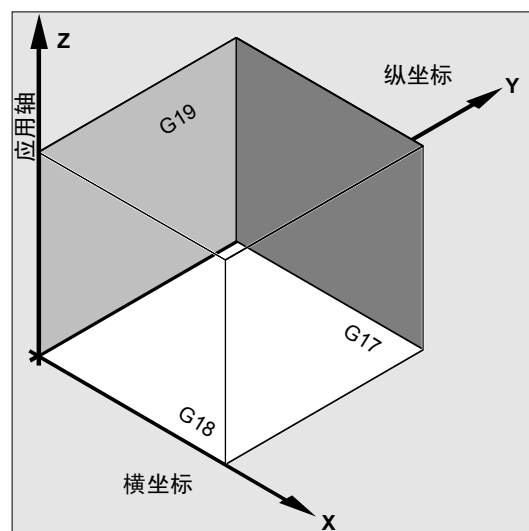
铣削图的圆心坐标或者待加工的凹槽在一个右手坐标系中编程。

在循环调用之前有效的G功能和当前可编程的框架可以在循环之后仍然保持。

### 平面定义

在铣削循环中前提条件就是：当前的工件坐标系通过选择平面G17、G18或者G19，并且激活一个可编程的框架（如果有要求）可以获得。进给轴总是该坐标系的第三根轴。

**文献：** /PG/, 编程说明—基础部分  
/PGA/, 编程说明  
工作准备部分



### 主轴处理

循环中的主轴指令始终与控制系统的当前主轴有关。

如果一个机床中带几个主轴，使用循环时必须事先通过指令把用于加工的主轴定义为主主轴。



**文献:** /PG/, 编程说明—基础部分  
/PGA/, 编程说明  
工作准备部分

### 加工状态的信息

在加工铣削循环时在控制系统屏幕上显示说明加工状态的信息。

可能会有下列信息:

- „长方形孔<序号>加工第一个形状”
- „键槽<序号>加工另一个形状”
- „圆弧形键槽<序号>加工最后一个形状”

在信息文本部分，每次均有一个正在加工形状的序号。

这些信息不中断程序执行，并且一直保持直至下一个信息的出现或者循环结束。

### 循环设定数据

铣削循环的某些参数和其性能可以通过循环设定改变。

循环设定数据在数据块GUD7.DEF中定义。

引入以下新的循环设定数据:

<u>_ZSD[x]</u>	值	含义	所涉及的循环
<u>_ZSD[1]</u>	0	在新的循环中深度在基准面+安全距离和深度（ <u>_RFP</u> + <u>_SDIS</u> - <u>_DP</u> ）之间进行计算	POCKET1 至 POCKET4、 LONGHOLE、 CYCLE71, SLOT1, CYCLE72, SLOT2
	1	进行深度计算时不包括安全距离	
<u>_ZSD[2]</u>	0	给定矩形槽或者矩形轴颈的尺寸，从圆心起	POCKET3 CYCLE76
	1	给定矩形槽或者矩形轴颈的尺寸，从一个角开始	
<u>_ZSD[5]</u>	0	执行到孔底深度M5M0	CYCLE88
	1	执行到孔底深度M5	

### 3.3 螺纹铣削CYCLE90



#### 编程

CYCLE90 (RTP, RFP, SDIS, DP, DPR, DIATH, KDIAM, PIT, FFR, CDIR, TYPTH, CPA, CPO)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	孔底深度（绝对）
DPR	实数	相对于基准面的钻削深度（不输入符号）
DIATH	实数	额定直径，螺纹的外径
KDIAM	实数	内部直径，螺纹的内径
PIT	实数	螺距；数值范围为：0.001 ... 2000.000 mm
FFR	实数	用于铣削螺纹的进给率（不输入符号）
CDIR	整数	铣削螺纹时方向 值： 2（用于带 G2的螺纹铣削） 3（用于带 G3的螺纹铣削）
TYPTH	整数	螺纹类型： 值： 0=内螺纹 1=外螺纹
CPA	实数	圆弧的圆心，横坐标（绝对）
CPO	实数	圆弧的圆心，纵坐标（绝对）



#### 功能

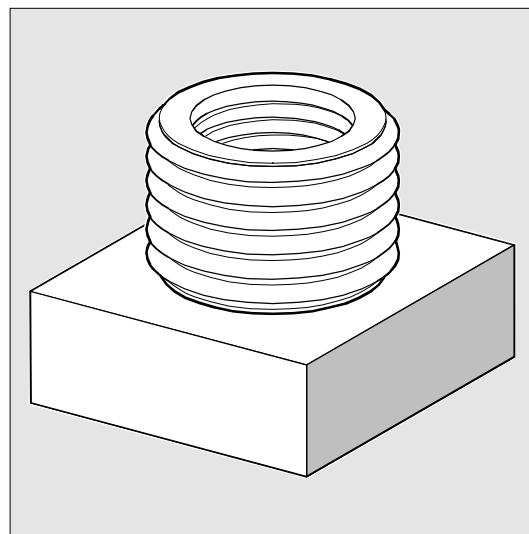
通过循环 CYCLE90

可加工内螺纹和外螺纹。铣削螺纹时，轨迹以螺旋线插补为基础。在循环调用前确定的当前平面的所有三个几何轴都参与到该运动中。

编程的进给率F有效，符合调用之前指令FGROUP中定义的轴组合



文献： /PG/, 编程说明—基础部分  
/PGA/, 编程说明  
工作准备部分





## 工作流程

### 外螺纹

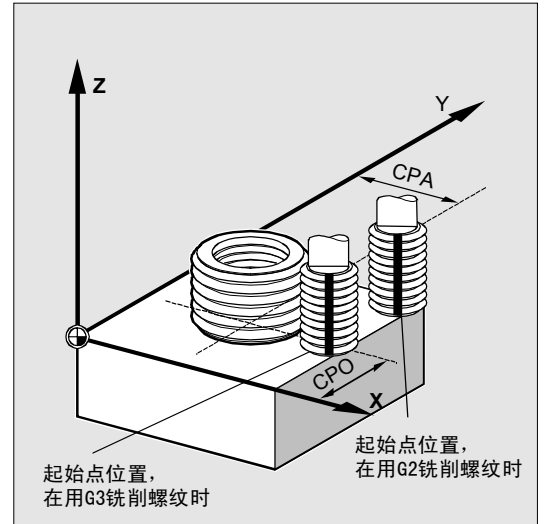
#### 循环开始之前到达的位置:

出发位置可以是任意一个位置，从这个出发位置使螺纹外径上的起始位置可以达到退回平面的高度，没有轮廓碰撞。

用G2 铣削螺纹时，该起始位置位于正向横坐标和当前平面中的正向纵坐标之间（也在坐标系的第1个象限中）。

用G3 切削螺纹时，起始位置位于正向横坐标和负向纵坐标之间（也在坐标系的第4个象限中）。

螺纹直径的距离取决于螺纹尺寸和所使用的刀具半径。



#### 该循环产生以下的运动过程:

- 在当前平面中应用轴上，以G0定位到起始点（在退回平面高度）
- 以G0横向进刀到提前了安全距离的基准面
- 按一个圆弧轨迹运行到螺纹直径，与CDIR下编程的方向G2/G3相反
- 以一个螺旋轨迹、G2/G3和进给率值FFR进行螺纹铣削
- 以一个圆弧轨迹、相反方向的转向G2/G3和减少的进给率FFR运行出来
- 在应用轴上以G0退回到退回平面

## 内螺纹

### 循环开始之前到达的位置:

出发位置可以是任意一个位置，从这个出发位置使螺纹的中心点位置可以达到（在退回平面的高度），没有轮廓碰撞。

### 该循环产生以下的运动过程:

- 在当前平面中应用轴上，以G0定位到螺纹的中心点（在退回平面高度）。
- 以G0横向进刀到提前了安全距离的基准面
- 以G1和减少的进给率FFR返回运行到循环内部计算的圆弧
- 按一个圆弧轨迹运行到螺纹直径，与CDIR下编程的方向G2/G3一致
- 以一个螺旋轨迹、G2/G3和进给率值FFR进行螺纹铣削
- 以一个圆弧轨迹、相同方向的转向G2/G3和减少的进给率FFR运行出来
- 以G0退回到螺纹的中心点
- 在应用轴上以G0退回到退回平面

### 螺纹，从下向上

出于工艺考虑，也可以从下向上加工螺纹。退回平面 RTP 位于螺纹深度 DP 之后。

加工时可以要求深度参数作为绝对值编程，并且在循环调用之前返回运行到退回平面或者一个位于退回平面之后的位置。





### 编程举例

（从下向上的螺纹）

应该以 3 mm 的螺距自 -20 开始铣削螺纹直至 0。  
退回平面位于 8 处。

```
N10 G17 X100 Y100 S300 M3 T1 D1 F1000
N20 Z8
N30 CYCLE90 (8, -20, 0, 0, 0, 46, 40, 3, 800, 3, 0, 50, 50)
N40 M2
```

钻孔必须至少有一个-21.5（半个螺距多）深度。

### 螺纹长度方向溢出行程

在铣削时在所有三个参加的轴上执行进入运行和退出运行，这就是说在螺纹退出时有一个附加的位移（在垂直轴上），它超出编程的螺纹深度。

计算溢出行程：

$$\Delta z = \frac{p}{4} * \frac{2*WR + RDIFF}{DIATH}$$

$\Delta z$  溢出行程，内部

$p$  螺距

$WR$  刀具半径

$DIATH$  螺纹外径

$RDIFF$  外伸圆弧半径公差

内螺纹时  $RDIFF = DIATH/2 - WR$ ,

外螺纹时  $RDIFF = DIATH/2 + WR$ 。

### 功能扩展，自 SW 6.3

将进给率换算到刀具半径上。对于内螺纹，现在在一个工艺有效的半圆轨迹上进行导入运动和导出运动，根据刀具在内部计算该循环的半圆轨迹半径。由此，现在按照下列公式得出溢出行程的循环内部计算结果：

- 在刀具直径 $<2/3$ 额定直径时

$$\Delta z = \frac{p}{2} * \frac{WR + DIATH/2}{DIATH}$$

- 在刀具直径 $\geq 2/3$  额定直径时

$$\Delta z = \frac{p/2 * KDIAM}{DIATH}$$

$\Delta z$  溢出行程，内部

$p$  螺距

$WR$  刀具半径

$DIATH$  螺纹外径

$KDIAM$  螺纹内核直径



### 参数说明



参数 RTP, RFP, SDIS, DP, DPR 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)

### DIATH, KDIAM 和 PIT (额定直径、内核直径和螺距)

通过这些参数确定额定直径、内核直径和螺距登螺纹数据。参数 DIATH 为螺纹外部直径，而参数 KDIAM 为螺纹内部直径。以这些参数为基础，在循环内部生成导入和导出运动。

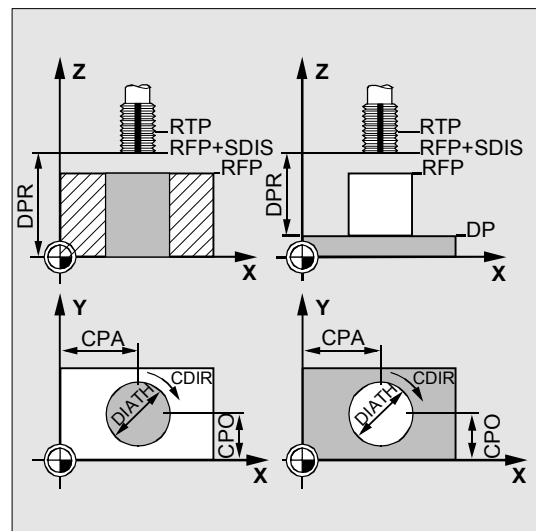
### FFR (进给)

在铣削螺纹时，将参数 FFR 的值规定为当前的进给值。该值在螺纹铣削期间沿螺旋线轨迹生效。

在循环中减少该值以用于导入运动和导出运动。以 G0 在螺旋线轨迹外部进行退回运动。

### CDIR (旋转方向)

在此参数下您可以规定螺纹加工方向的值。如果该参数有一个不允许的值，则显示信息“铣削方向错误，产生G3”。



在这种情况下继续执行该循环，并自动产生G3。

#### **TYPTH (螺纹类型)**

使用该参数TYPTH您可以确定是否应该进行外螺纹或者内螺纹的加工。

#### **CPA 和 CPO (中心点)**

在这些参数下，您可以确定进行螺纹加工时钻孔或者轴颈的中心点。



#### **其它说明**

在循环内部计算铣刀半径。因此，在循环调用前编程一个刀具补偿。否则会出现报警

**61000 “没有刀具补偿有效”**

并停止循环的执行。

在刀具半径=0时或者为负时，该循环同样以该报警中断。

在内螺纹时监控该刀具半径，并发出报警**61105 “铣刀半径太大”**

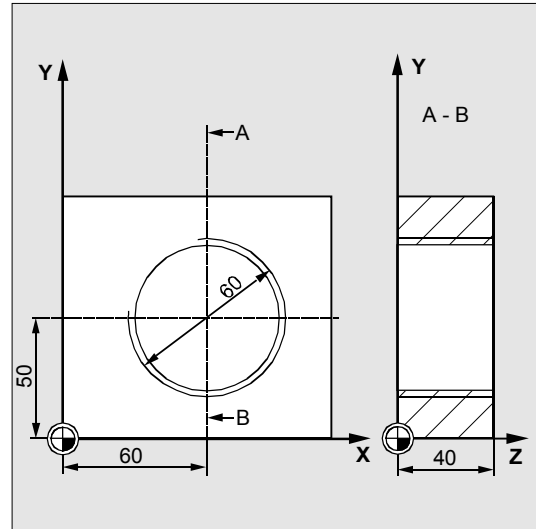
并且中断循环执行。



## 编程举例

## 内螺纹

使用该程序您可以在G17平面位置X60Y50处铣削一个内螺纹。



```

DEF REAL RTP=48, RFP=40, SDIS=5, -> ; 变量定义
-> DPR=40, DIATH=60, KDIAM=50 ; 赋值
DEF REAL PIT=2, FFR=500, CPA=60, CPO=50
DEF INT CDIR=2, TYPTH=0
N10 G90 G0 G17 X0 Y0 Z80 S200 M3 ; 返回运行到出发位置
N20 T5 D1 ; 确定工艺数值
N30 CYCLE90 (RTP, RFP, SDIS, DPR, -> ; 循环调用
-> DIATH, KDIAM, PIT, FFR, CDIR, ->
-> TYPTH, CPA, CPO)
N40 G0 G90 Z100 ; 循环之后返回运行到位置
N50 M02 ; 程序结束
-> 必须在一个程序段中编程

```

### 3.4 一个圆弧上的长方形孔 - LONGHOLE



#### 编程

LONGHOLE (RTP, RFP, SDIS, DP, DPR, NUM, LENG, CPA, CPO, RAD, STA1, INDA, FFD, FFP1, MID)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	长方形孔深度（绝对）
DPR	实数	相对于基准面的长方形孔深度（不输入符号）
NUM	整数	长方形孔个数
LENG	实数	长方形孔长度（不输入符号）
CPA	实数	圆弧的圆心，横坐标（绝对）
CPO	实数	圆弧的圆心，纵坐标（绝对）
RAD	实数	圆弧半径（不输入符号）
STA1	实数	起始角
INDA	实数	增量角度
FFD	实数	深度方向的进给
FFP1	实数	表面加工的进给
MID	实数	一个横向进给的最大进刀深度（不输入符号）



该循环要求一个铣刀带“过中心的端面齿切削”  
(DIN844)。



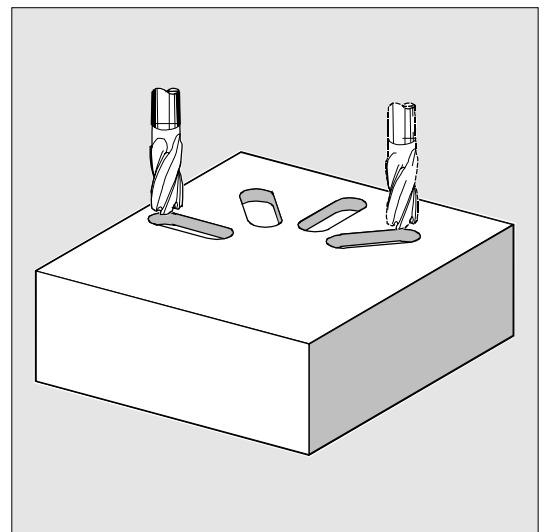
#### 功能

使用该循环您可以加工位于一个圆弧上的长方形孔。

长方形孔的纵向轴以半径对准。

与键槽相反，长方形孔的宽度由刀具直径确定。

在循环内部确定最佳的刀具运行行程，避免不必要的空行程。如果需要多个深度进给运行用于加工长方形孔，则在终点上进行进给运动切换。在平面中沿着长方形孔的纵向轴的待离开的轨迹根据每个进给运动变换方向。在过渡至下一个长方形孔时，循环自动查找最短的路径。





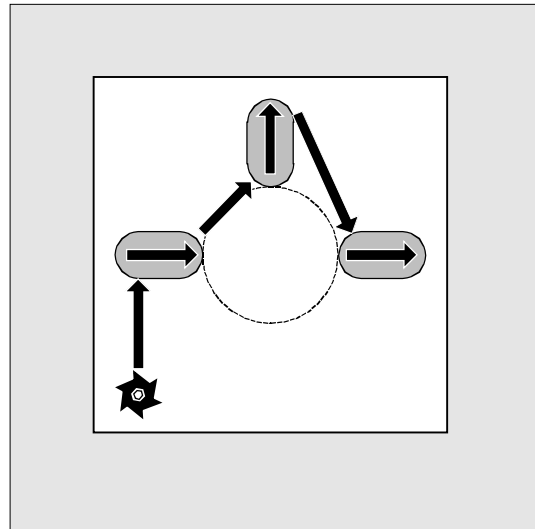
## 工作流程

### 循环开始之前到达的位置:

出发位置可以为一个任意位置，从这个出发位置可以返回运行到每个长方形孔，没有轮廓碰撞。

### 该循环产生以下的运动过程:

- 用G0返回运行到该循环的出发位置。在当前平面的两个轴中，返回运行到第一个待加工的长方形孔最近的终点（应用轴退回平面的高度），然后再在应用轴上下降到提前了安全距离的基准面。
- 以摆动运动铣削每个长方形孔。用 G1 和在 FFP1 下编程的进给值在平面中进行加工。在下一个循环内部计算的加工深度上用 G1 和进给率 FFD 在每个换向点进行进给运动，直至到达最终深度。
- 以G0退回到退回平面，以最短的行程返回运行到下一个长方形孔。
- 在结束最后一个长方形孔的加工之后，刀具在加工平面中最后到达的位置以G0运行直至退回平面，该循环结束。





### 参数说明

参数 RTP, RFP, SDIS 参见

章节 2.1.2 (钻削, 定中心 - CYCLE81)。

循环设定参数 `_ZSD[1]` 参见章节 3.2。

#### DP 和 DPR (长方形孔深度)

钻削深度可以以到基准面的绝对尺寸 (DP) 规定, 也可以以到基准面的相对尺寸 (DPR) 规定。

在相对尺寸时, 循环利用基准面和退回平面的位置自己计算所产生的深度。

#### NUM (个数)

使用参数 NUM 您可以确定长方形孔个数。

#### LENG (长方形孔长度)

在 LENG 下您可以编程长方形孔长度。

如果在循环中判别出该长度小于铣削直径, 则中止该循环, 并发出报警 61105 “铣削半径太大”。

#### MID (进刀深度)

通过这些参数, 您可以确定最大的进刀深度。

在循环中深度以均匀的进刀尺寸进行。

借助 MID 和总深度, 该循环自动计算

0.5 x 最大进刀深度和最大进刀深度之间的进刀量。

存放进刀尺寸最小可能的数量。MID=0

表示一步进刀到凹槽深度。

深度方向的进刀以提前了安全距离的基准面开始 (与 `_ZSD[1]` 相关)。

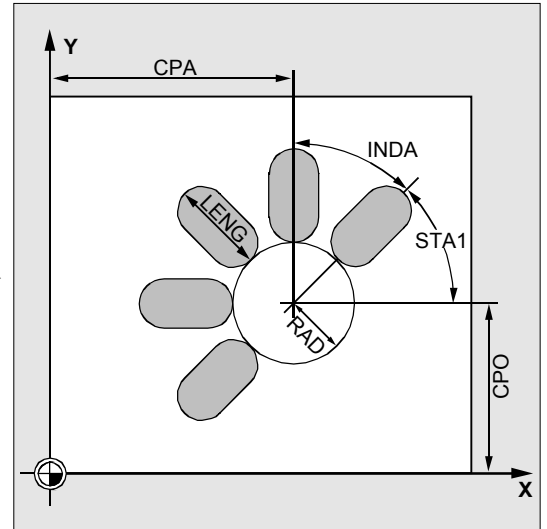
#### FFD 和 FFP1 (进给深度和面积)

进给 FFP1 在平面中以进给运行的所有运动中生效。

FFD 在垂直进刀至该平面时生效。

#### CPA, CPO 和 RAD (圆心和半径)

通过圆心 (CPA, CPO) 和半径 (RAD) 定义加工平面中的圆弧位置。半径仅允许为正值。



### STA1 和 INDA（起始角和增量角）

通过这些参数，您可以确定圆弧上长方形孔的排列。  
如果INDA=0，则增量角通过长方形孔的数量进行计算，它们均匀地分步在圆弧上。

### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会中断并有报警

61000 “没有刀具补偿有效”发出。

如果由于确定长方形孔的排列和大小的参数值错误，使长方形孔产生轮廓损伤，则不开始进行这个循环的加工。该循环在给出错误报警

61104 “键槽/长方形孔轮廓损伤”之后停止。

在循环内部平移和旋转工件坐标系。WCS中实际值始终如此显示，使正在加工的长方形孔的纵向轴位于当前加工平面的第1个轴上。

在结束该循环之后，工件坐标系位于循环调用之前相同的位置上。

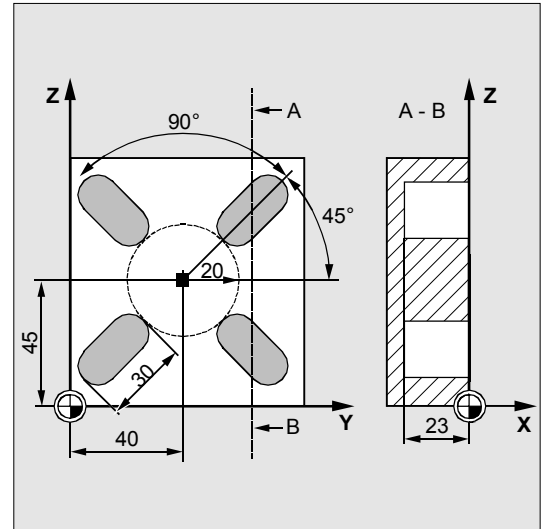




### 编程举例

#### 长方形孔加工

使用该程序，您可以加工4个长度为30毫米、相对深度为23毫米的长方形孔，（基准面和长方形孔底部的差值），它们位于一个圆弧上，圆心为Z45Y40、半径为20毫米，在YZ平面。起始角为45度，增量角为90度。最大进刀深度为6毫米，安全距离为1毫米。




---

N10 G19 G90 S600 M3

; 确定工艺数值

---

T10 D1

---

M6

---

N20 G0 Y50 Z25 X5

; 返回运行到出发位置

---

N30 LONGHOLE (5, 0, 1, , 23, 4, 30, ->  
-> 40, 45, 20, 45, 90, 100 ,320, 6)

; 循环调用

---

N40 M30

; 程序结束

-> 必须在一个程序段中编程

## 3.5 一个圆弧上的键槽 - SLOT1



## 编程

SLOT1 (RTP, RFP, SDIS, DP, DPR, NUM, LENG, WID, CPA, CPO, RAD, STA1, INDA, FFD, FFP1, MID, CDIR, FAL, VARI, MIDF, FFP2, SSF, \_FALD, \_STA2)



## 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	键槽深度（绝对）
DPR	实数	相对于基准面的键槽深度（不输入符号）
NUM	整数	键槽数量
LENG	实数	键槽长度（不输入符号）
WID	实数	键槽宽度（不输入符号）
CPA	实数	圆弧的圆心，横坐标（绝对）
CPO	实数	圆弧的圆心，纵坐标（绝对）
RAD	实数	圆弧半径（不输入符号）
STA1	实数	起始角
INDA	实数	增量角度
FFD	实数	深度方向的进给
FFP1	实数	表面加工的进给
MID	实数	一个横向进给的最大进刀深度（不输入符号）
CDIR	整数	键槽加工的铣削方向 值： 0...同向铣削（与主轴转向一致） 1...逆向铣削 2...用 G2（与主轴转向无关） 3...用 G3
FAL	实数	键槽边缘的精加工余量（不输入符号）
VARI	整数	加工方式（不输入符号）个位： 值： 0...全套加工 1...粗加工 2...精加工  十位： 值： 0...以 G0 垂直 1...以 G1 垂直 3...以 G1 摆动
MIDF	实数	精加工最大进刀深度
FFP2	实数	精加工进给
SSF	实数	精加工时速度

_FALD	实数	键槽底部精加工余量
_STA2	实数	摆动运动时最大插入角



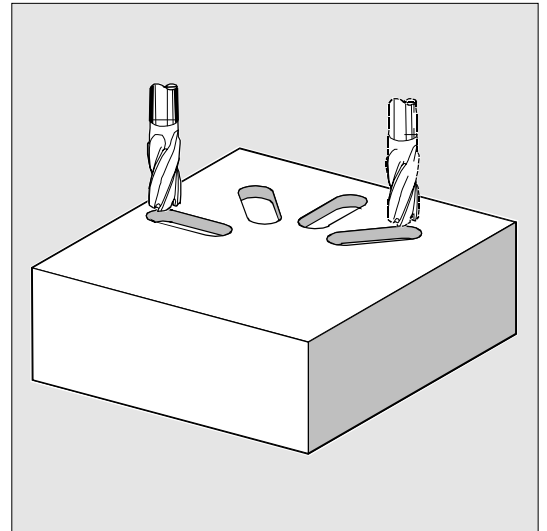
该循环要求一个铣刀带“过中心的端面齿切削”  
(DIN844)。



### 功能

循环SLOT1是一个组合的粗加工—精加工循环。

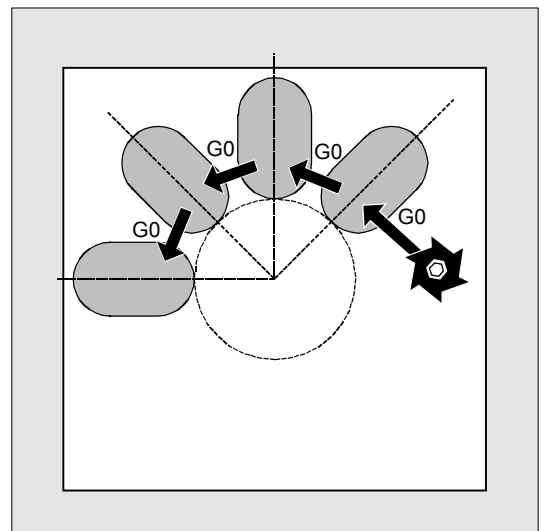
用该循环可加工位于一个圆弧上的键槽。键槽的纵向轴以半径对准。和长方形孔相反，为键槽宽度给定一个值。



### 工作流程

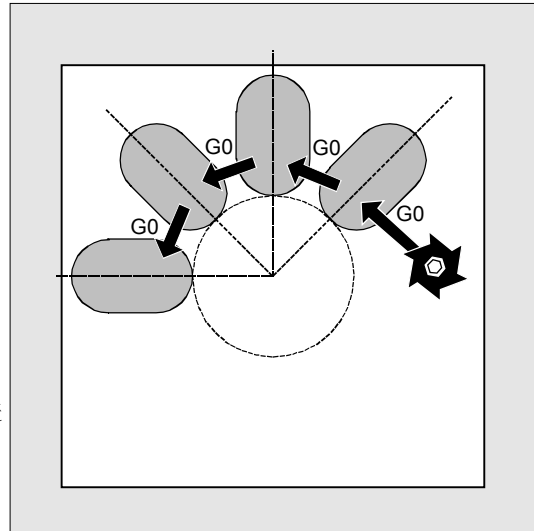
循环开始之前到达的位置：

出发位置可以为一个任意位置，从这个出发位置可以返回运行到每个键槽，没有轮廓碰撞。



该循环产生以下的运动过程：

- 在旁边图形中，以G0返回运行到循环开始
- 在全套加工过程中，一个键槽按如下步骤进行加工：
  - 以G0返回到提前了相隔安全距离的基准面
  - 进刀到下一个加工深度，如同VARI下编程和进给值FFD。
  - 用进给值FFP1铣出一个键槽，直至键槽底部和键槽边缘的精加工余量。紧接着使用进给值FFP2和主轴转速SSF沿着轮廓进行精加工，按照CDIR下编程的加工方向进行。
  - 以G0/G1进行垂直方向的深度进刀（在加工平面中同样的位置），直至到达键槽的底部。
  - 在摆动加工时选择起始点，使终点始终到达加工平面中相同的位置。
- 刀具退回到退回平面，并以G0过渡到下一个槽。
- 在结束最后键槽的加工之后，刀具以G0运行到退回平面，结束循环。

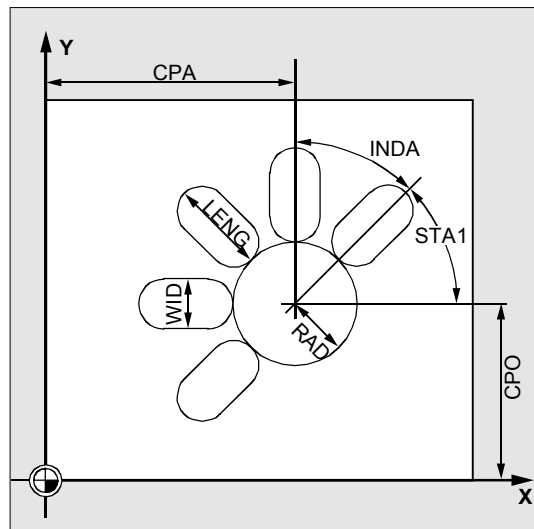


### 参数说明

参数 RTP、RFP、SDIS 参见  
 章节 2.1.2.（钻削，定中心 - CYCLE81）。  
 循环设定参数\_ZSD[1]参见章节3.2。

#### DP 和 DPR（键槽深度）

键槽深度可以以到基准面的绝对尺寸（DP）规定，也可以以到基准面的相对尺寸（DPR）规定。在相对尺寸时，循环利用基准面和退回平面的位置自己计算所产生的深度。



**NUM (个数)**

使用参数NUM您可以说明键槽个数。

**LENG 和 WID (键槽长度和键槽宽度)**

使用参数LENG和WID

确定平面中一个键槽的形状。铣刀直径必须小于键槽宽度。否则会出现报警

61105 “铣刀半径太大”且循环被中断。

铣刀直径不得小于键槽宽度的一半。不用进行检查。

**CPA, CPO 和 RAD (圆心和半径)**

通过圆心 (CPA, CPO) 和半径 (RAD) 定义加工平面中的孔圆弧位置。半径仅允许为正值。

**STA1 和 INDA (起始角和增量角)**

通过这些参数，您可以确定圆弧上键槽的排列。

STA1说明循环调用之前实际工件坐标系横坐标的正方向与第一个键槽之间的夹角。参数INDA包含从一个键槽到下一个键槽之间的夹角。

如果INDA=0，则增量角通过键槽的数量进行计算，它们均匀地分步在圆弧上。

**FFD 和 FFP1 (进给深度和面积)**

进给FFD在加工平面的垂直方向 (G1) 生效，并且在插入时带摆动运动。

进给FFP1在粗加工时生效，在平面中以进给运行的所有运动中。

**MID (进刀深度)**

通过这些参数，您可以确定最大的进刀深度。在循环中深度以均匀的进刀尺寸进行。

利用MID 和总深度，该循环自行计算

0.5 x最大进刀深度和最大进刀深度之间的进刀量。

存放进刀尺寸最小可能的数量。MID=0

表示一步进刀到键槽深度。

深度方向的进刀以提前了安全距离的基准面开始（与\_Z SD[1]相关）。

#### CDIR（铣削方向）

在此参数下规定键槽加工的方向。

通过参数CDIR可以

- 直接编程铣削方向“2用于G2”和“3用于G3”，或者
- 选择“同向运行”或者“逆向运行”。同向运行或者逆向运行由循环在内部通过循环调用之前激活的主轴方向确定。

同向运行	逆向运行
M3 → G3	M3 → G2
M4 → G2	M4 → G3

#### FAL（键槽边缘上精加工余量）

使用该参数可在键槽边缘编程一个精加工余量。

FAL 未对深度进刀作出反应。如果由

FAL给定的值大于其给定的宽度和所用的铣刀，则

FAL 自动将值减少到可能的最小值。

在这种情况下，粗加工时，在键槽的两个终点上以深度进刀进行摆动铣削。

#### VARI, MIDF, FFP2 和

#### SSF（加工方式、进刀深度、进给和转速）

使用参数VARI您可以确定加工方式。

可能的值是：

个位

- 0=在两个截面上进行综合加工
  - 扩孔加工键槽（SLOT1,SLOT2）或者凹槽(POCKET1, POCKET2)直至精加工余量，使用调用循环之前编程的主轴转速和进给FFP1。深度进刀通过MID进行。
  - 剩余精加工余量的扩孔通过由SSF给定的主轴转速和进给FFP2进行。使用参数MIDF可以编程一个不同于粗加工的其它深度进刀，但它仅在边缘精加工时生效。如果MIDF=0，则进刀立即到最终深度。

如果没有编程FFP2，则进给FFP1生效。

类似的如果没有SSF参数说明，则循环调用之前编程的转速生效。

- **1=粗加工**

使用循环调用之前编程的转速和进给FFP1对键槽

(SLOT1, SLOT2) 或者凹槽 (POCKET1, POCKET2) 进行扩孔，直至精加工余量。深度进刀通过MID进行编程。

- **2=精加工**

循环的前提条件是，键槽 (SLOT1, SLOT2) 或者凹槽 (POCKET1, POCKET2) 已扩孔至剩余的精加工余量，还需要扩孔该精加工余量。如果尚未编程 FFP2 和 SSF，则进给 FFP1 或在循环调用前编程的转速生效。可在参数 MIDF 编程一个深度进刀值以用于边缘精加工。在加工方式VARI=30时最后粗加工的深度之后是边缘精加工。

十位 (进刀)

- 0=以 G0 垂直
- 1=以 G1 垂直
- 3=以G1摆动

如果参数VARI编程一个其它值，则循环中断并发出报警 61102 “加工方式错误定义”。

### 铣刀直径=键槽宽 (WID)

- 在综合加工时仅在底部进行精加工。
- 在加工方式VARI=32时，在Z轴平行方向以G1进行定位，然后再进行精加工（可以通过MIDF进行进刀）。

### FALD (在键槽底部的精加工余量)

在粗加工时，在底部考虑一个分开的精加工余量。

### STA2 (再入角)

使用参数STA2定义最大的再入角，用于摆动运动。

- **垂直插入 (VARI=0X, VARI=1X)**

始终在加工平面中同样的位置进行垂直方向的深度

进刀，直至到达键槽的底部。

- **摆动插入到键槽中间轴 (VARI=3X)**

表示，铣刀中心点沿一条直线前后摆动斜向

插入，直至其到达下一个当前深度。在 `_STA2`

下编程最大插入角度，从 `LENG-WID`

中计算出摆动位移的长度。

摆动深度进刀在相同的位置结束，如同垂直进刀

时一样，在平面中相应计算起始点。如果已到达

当前深度，则在平面中开始进行粗加工。在 `FFD`

下编程进给率。

### 其它说明

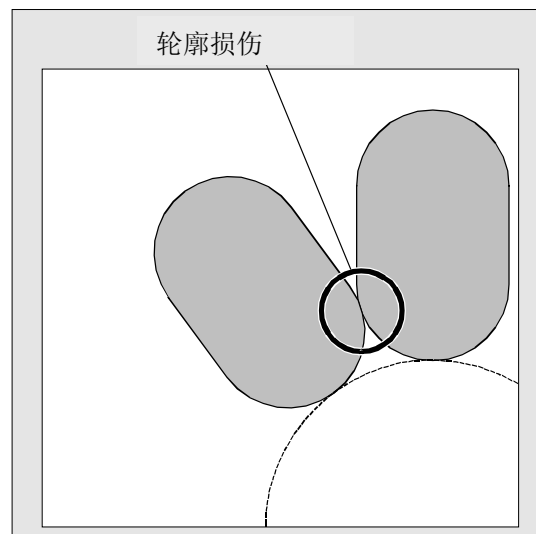
在循环调用之前激活一个刀具补偿。否则循环会被中断并给出报警“无刀具补偿有效”。

如果由于确定键槽的排列和大小的参数值错误，使键槽产生轮廓损伤，则不开始进行这个循环的加工。该循环在给出错误报警

**61104** “键槽/长方形孔轮廓损伤”后停止。

在循环内部平移和旋转工件坐标系。`WCS`中实际值始终如此显示，使正在加工的长方形孔的纵向轴位于当前加工平面的第1个轴上。

在结束该循环之后，工件坐标系位于循环调用之前相同的位置上。





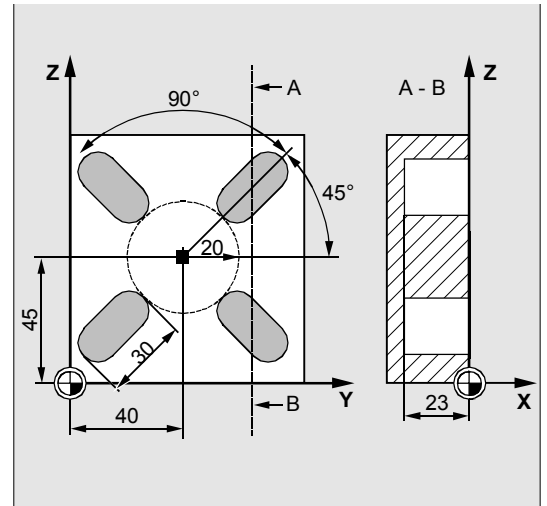


## 编程举例

## 键槽



该程序加工4个键槽，位于一个圆弧上，其分布如同长方形孔加工程序中一样（参见章节3.4）。  
键槽尺寸如下：长度为30 mm、宽度为15 mm且深度为23 mm。安全距离为1 mm，精加工余量为0.5 mm，铣削方向为G2，最大进刀深度为10 mm。键槽应该完全通过摆动插入进行加工。



N10 G19 G90 S600 M3

; 确定工艺数值

N15 T10 D1

N17 M6

N20 G0 Y20 Z50 X5

; 返回运行到出发位置

N30 SLOT1 (5, 0, 1, -23, , 4, 30, 15, -> ; 循环调用，删除参数VARI, MIDF, FFP2  
->40, 45, 20, 45, 90, 100, 320, 10, -> 和 SSF  
->2, 0.5, 30, 10, 400, 1200, 0.6, 5)

N40 M30

; 程序结束

-> 必须在一个程序段中编程

### 3.6 环形槽—SLOT2



#### 编程

SLOT2 (RTP, RFP, SDIS, DP, DPR, NUM, AFSL, WID, CPA, CPO, RAD, STA1, INDA, FFD, FFP1, MID, CDIR, FAL, VARI, MIDF, FFP2, SSF, \_FFCP)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	键槽深度（绝对）
DPR	实数	相对于基准面的键槽深度（不输入符号）
NUM	整数	键槽数量
AFSL	实数	用于键槽长度的角度（不输入符号）
WID	实数	环形槽宽度（不输入符号）
CPA	实数	圆弧的圆心，横坐标（绝对）
CPO	实数	圆弧的圆心，纵坐标（绝对）
RAD	实数	圆弧半径（不输入符号）
STA1	实数	起始角
INDA	实数	增量角度
FFD	实数	深度进刀的进给
FFP1	实数	表面加工的进给
MID	实数	一个横向进给的最大进刀深度（不输入符号）
CDIR	整数	环形槽加工的铣削方向 值： 2 (用于 G2) 3 (用于 G3)
FAL	实数	键槽边缘的精加工余量（不输入符号）
VARI	整数	加工方式 个位： 值： 0=综合加工 1=粗加工 2=精加工 十位 (自软件版本 SW 6.3起) 值： 0=以G0并在直线上由槽到槽的定位 1=以进给并在环形轨道上由槽到槽的定位
MIDF	实数	精加工最大进刀深度
FFP2	实数	精加工进给
SSF	实数	精加工时速度
_FFCP (自软件 版本)	实数	中间定位进给，环形轨道，单位毫米/分钟

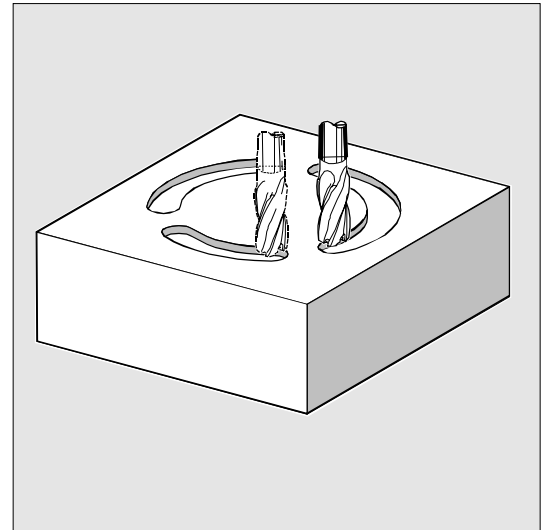


该循环要求一个铣刀带“过中心的端面齿切削”  
(DIN844)。



### 功能

循环SLOT2是一个组合的粗加工—精加工循环。  
使用该循环您可以加工环形槽，这些槽位于一个圆弧上。



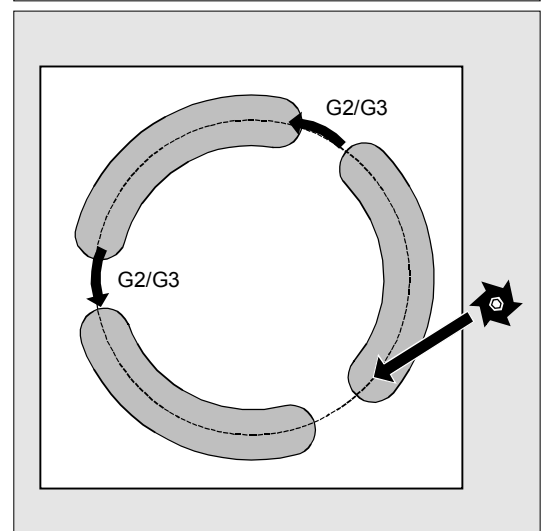
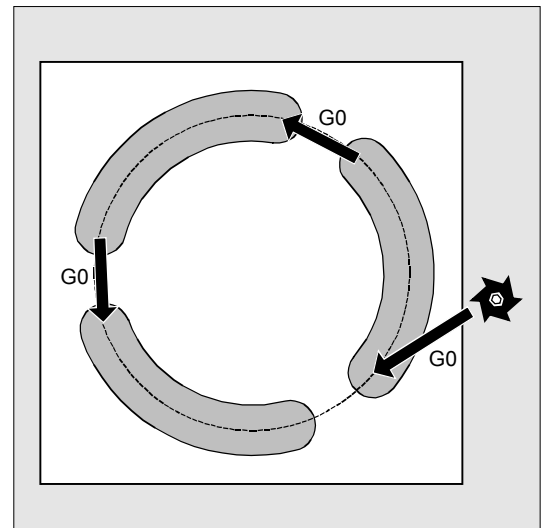
### 工作流程

#### 循环开始之前到达的位置:

出发位置可以为一个任意位置，从这个出发位置可以返回运行到每个键槽，没有轮廓碰撞。

#### 该循环产生以下的运动过程:

- 在右侧图中，以G0返回到循环开始的位置。
- 环形槽的加工以与长方形孔加工相同的步骤进行。
- 在环形槽的加工结束之后刀具退回到退回平面，并且以G0按照直线过渡到下一个键槽，或者以 \_FFCP中编程的进给按照环形轨道过渡到下一个键槽。
- 在结束最后一个键槽的加工之后，刀具在加工平面中最后到达的位置以G0运行直至退回平面，该循环结束。





## 参数说明



参数 RTP、RFP、SDIS 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)。



参数 DP、DPR、FFD、FFP1、MID、CDIR、FAL、  
VARI、MIDF、FFP2、SSF 参见章节  
3.5 (SLOT1)。

循环设定参数\_ZSD[1]参见章节3.2。

### NUM (个数)

使用参数NUM您可以说明键槽个数。

### AFSL 和 WID (角度和环形槽宽度)

使用参数AFSL和WID

确定平面中一个键槽的形状。在循环内部检查, 使用  
有效的刀具是否会损伤键槽宽度。否则会出现报警  
61105 “铣刀半径太大”

且循环被中断。

### CPA, CPO 和 RAD (圆心和半径)

通过圆心 (CPA, CPO) 和半径 (RAD) 定义加工  
平面中的圆弧位置。半径仅允许为正值。

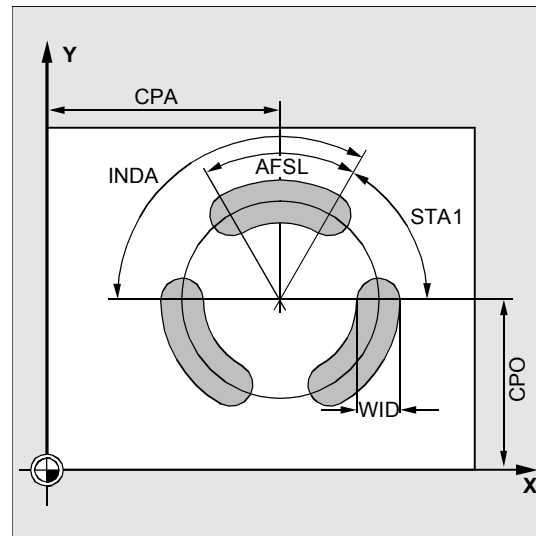
### STA1 和 INDA (起始角和增量角)

通过这些参数, 您可以确定圆弧上环形槽的排列。

STA1说明循环调用之前实际工件坐标系横坐标的正方向  
与第一个环形槽之间的夹角。

参数INDA包含从一个环形槽到下一个环形槽之间的  
夹角。

如果INDA=0, 则增量角通过环形槽的数量进行计  
算, 它们均匀地分步在圆弧上。



自软件版本**SW6.3**以后新的加工方式:

#### 在边缘上进行的精加工 (VARI=x3)

- 有一个新的选择“精加工边缘”。仅在这种加工方式时允许铣刀直径小于键槽宽度的一半。  
不用监控，尺寸是否足够用于处理精加工余量FAL。
- 可有多个深度进刀量。否则，这会如通过 MID 进行编程一样。键槽以每种深度绕行一次。
- 在循环内部返回和离开运行轮廓时产生一个柔和的环形轨道区段逼近。

#### 在环形轨道上进行中间定位 (VARI=1x)

- 特别是在车床的应用中，有时会在键槽圆弧的中间有一个轴颈，因为它而使得从一个键槽到下一个键槽的直接定位 (G0) 变得不可能。
- 将一个键槽所处的圆弧确定作为圆弧轨迹 (通过参数 CPA、CPO、RAD 确定)。如同用 G0 中间定位到直线上那样定位到相同的高度。圆弧轨迹的定位进给在参数下以 mm/min 为单位编程。

### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会中断并给出报警

61000 “无刀具补偿有效”。

如果由于确定键槽的排列和大小的参数值错误，使键槽产生轮廓损伤，则不开始进行这个循环的加工。

该循环在给出错误报警

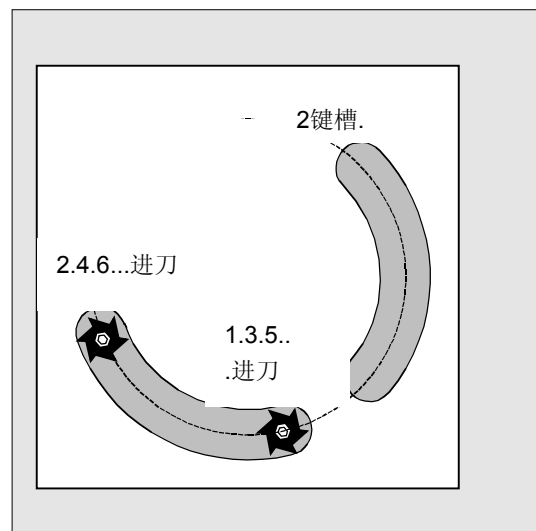
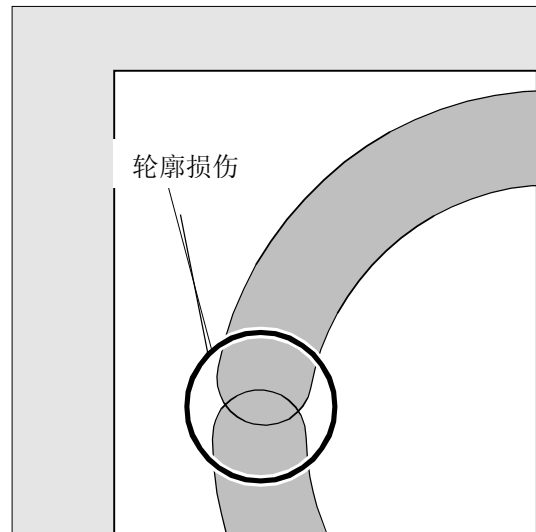
61104 “键槽/长方形孔轮廓损伤”之后停止。

在循环内部平移和旋转工件坐标系。在WCS中的实际值显示，始终使正在加工的环形槽从当前加工平面的第一个轴上开始，并且WCS的零点位于圆弧的圆心。

在结束该循环之后，工件坐标系位于循环调用之前相同的位置上。

#### 特殊情况：键槽宽度 = 铣刀直径

- 在粗加工和精加工时允许加工情况键槽宽度 = 铣刀直径。当键槽宽度  $WID - 2 * \text{精加工余量 } FAL = \text{铣刀直径}$  时，这种加工情况将会出现。
- 这样，运行方案与在循环LONGHOLE时一样，也就是说深度进刀交替地在换向点进行，参见图形。





### 编程举例

#### 键槽2

使用该程序，可加工三个位于圆心为 X60 Y60、XY 平面中的半径为 42 mm 的圆弧上的环形槽。

环形槽尺寸如下：宽度为 15 mm，

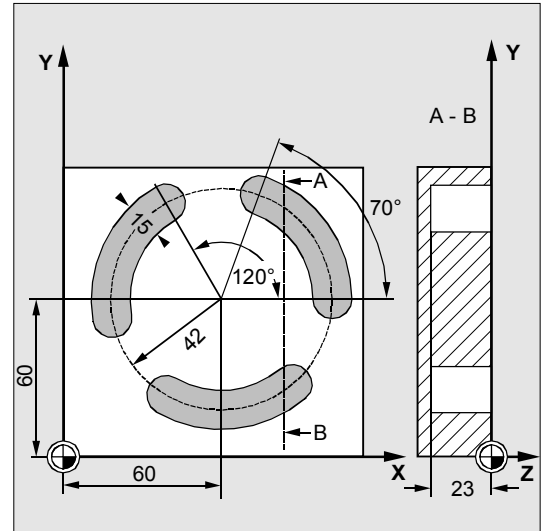
键槽长度的角度为 70 度，深度为 23 mm。

起始角为 0 度，增量角为 120 度。

在键槽轮廓上的精加工余量考虑为 0.5 mm，进给轴 Z 的安全距离为 2 mm，最大进刀深度为

6 mm。必须完全加工键槽。精加工时，

相同的转速和相同的进给应生效。精加工时，进刀量应等于键槽深度。



DEF REAL FFD=100	; 参数定义，赋值
N10 G17 G90 S600 M3	; 确定工艺数值
N15 T10 D1	
N17 M6	
N20 G0 X60 Y60 Z5	; 返回起始位置
N30 SLOT2 (2, 0, 2, -23, , 3, 70, -> -> 15, 60, 60, 42, , 120, FFD, -> -> FFD+200, 6, 2, 0.5)	; 循环调用 ; 基准面+SDIS=退回平面; 表示: 取消 在横向进给轴 (带G0) 方向到基准面 +SDIS的下降, 删除参数VAR、 MIDF、FFP2 和 SSF
N40 M30	; 程序结束

-> 必须在一个程序段中编程

## 3.7 铣削矩形槽 — POCKET1



## 编程

POCKET1 (RTP, RFP, SDIS, DP, DPR, LENG, WID, CRAD, CPA, CPD, STA1, FFD, FFP1, MID, CDIR, FAL, VARI, MIDF, FFP2, SSF)



## 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	凹槽深度（绝对）
DPR	实数	相对于基准面的槽深度（不输入符号）
LENG	实数	凹槽长度（不输入符号）
WID	实数	凹槽宽度（不输入符号）
CRAD	实数	拐角半径（不输入符号）
CPA	实数	凹槽中心点，横坐标（绝对）
CPO	实数	凹槽中心点，纵坐标（绝对）
STA1	实数	纵向轴和横坐标之间的夹角 值范围：0≤STA1<180度
FFD	实数	深度方向的进给
FFP1	实数	表面加工的进给
MID	实数	一个横向进给的最大进刀深度（不输入符号）
CDIR	整数	槽加工的铣削方向 值： 2（用于 G2） 3（用于 G3）
FAL	实数	槽边缘的精加工余量（不输入符号）
VARI	整数	加工方式 值： 0=综合加工 1=粗加工 2=精加工
MIDF	实数	精加工最大进刀深度
FFP2	实数	精加工进给
SSF	实数	精加工时速度



该循环要求一个铣刀带“过中心的端面齿切削”  
(DIN844)。



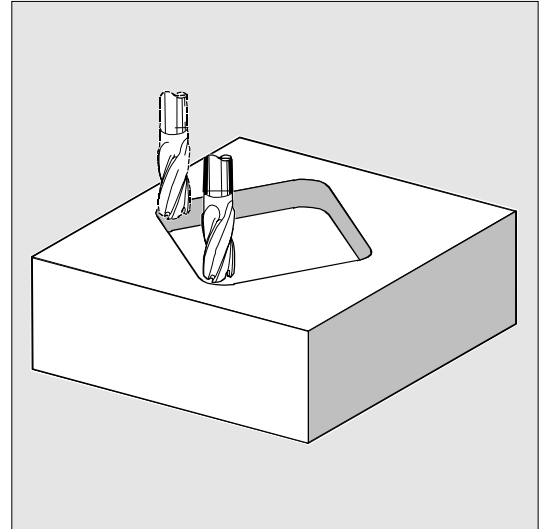
槽铣削循环POCKET3适用于使用任意刀具。





### 功能

该循环是一个组合的粗加工—精加工循环。  
利用该循环您可以加工平面中任意位置的矩形槽。



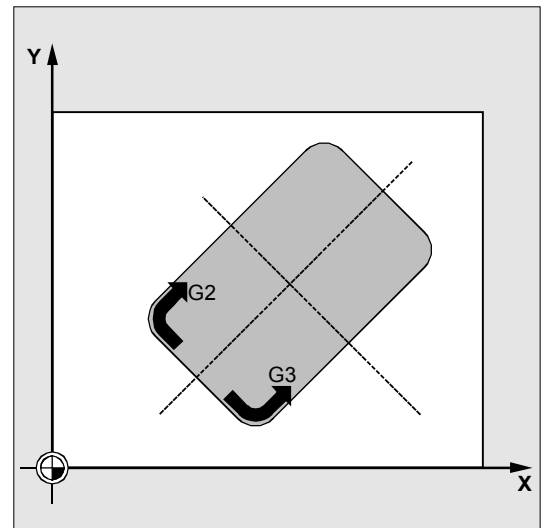
### 工作流程

#### 循环开始之前到达的位置:

出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到槽中心点，没有轮廓碰撞。

#### 该循环产生以下的运动过程:

- 以 G0 凹槽中心点返回到退回平面高度，接着同样以 G0 在该位置上运行到提前了安全距离的基准面。  
综合加工时遵循下列步骤加工凹槽：
  - 以 G1 和进给值 FFD 进刀到下一个加工深度
  - 以进给 FFP1 和循环调用之前有效的主轴转速铣削出凹槽，直至精加工余量。
- 粗加工结束之后：
  - 进刀到通过 MIDF 确定的加工深度
  - 以进给 FFP2 和转速 SSF 沿着轮廓进行精加工。
  - 加工方向按照 CDIR 下确定的方向。
- 在凹槽加工结束之后，刀具运行到槽中心点，直至退回平面，循环结束。





### 参数说明



参数 RTP、RFP、SDIS 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)。

参数 FFD、FFP1、MID、CDIR、FAL、VARI、  
MIDF、FFP2、SSF 参见章节 3.5 (SLOT1)。  
循环设定参数\_ZSD[1]参见章节3.2。

#### DP 和 DPR (槽深)

凹槽深度可以以到基准面的绝对尺寸 (DP) 规定, 也可以以到基准面的相对尺寸 (DPR) 规定。

在相对尺寸时, 循环利用基准面和退回平面的位置自己计算所产生的深度。

#### LENG、WID 和 CRAD (长度、宽度和半径)

使用参数LENG、WID 和

CRAD, 确定平面中一个凹槽的形状。

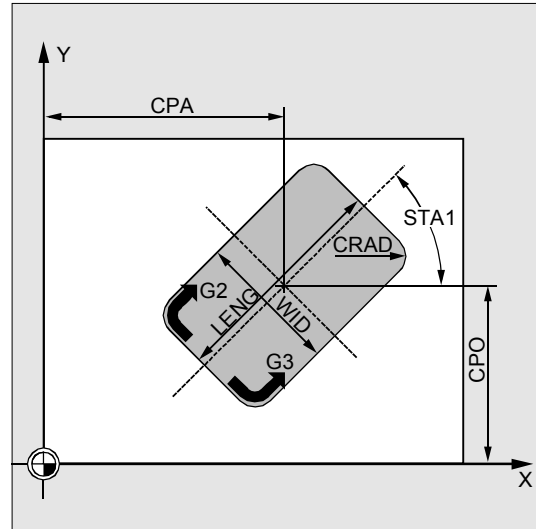
如果当前的刀具不能运行编程的拐角半径 (因为其半径太大), 则加工完毕的凹槽拐角半径等于刀具半径。如果刀具的铣刀半径大于凹槽一半的长度或者宽度, 则给出报警 61105 “铣刀半径太大”。

#### CPA, CPO (中心点)

使用参数CPA和CPO您可以定义凹槽在横坐标和纵坐标上的中心点。

#### STA1 (角度)

STA1给出正向横坐标和凹槽纵向轴之间的夹角。



### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会中断并给出报警

61000 “没有刀具补偿有效”。

在循环内部使用一个新的当前工件坐标系, 该坐标系影响实际值显示。该坐标系的零点就是凹槽中心点。

在循环结束之后, 原来的坐标系再次生效。



### 编程举例

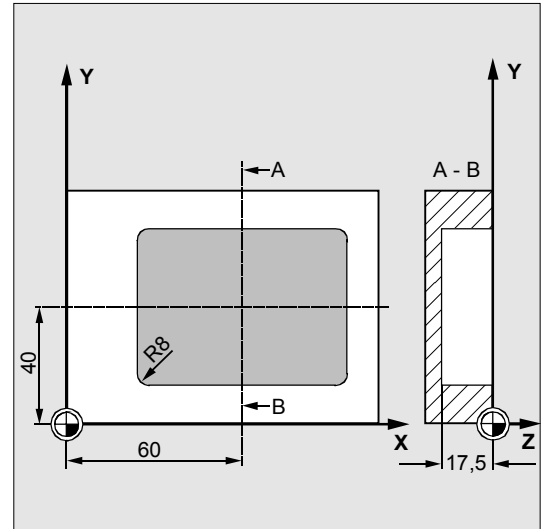
#### 凹槽

使用此程序您可以在 XY

平面中加工一个凹槽，其长度为60毫米，宽度40毫米，拐角半径8毫米，深度17.5毫米（基准面和槽底的差值）。凹槽与 X 轴成 0 度角。

凹槽边缘的精加工余量为 0.75 mm，Z 轴中增加到基准面上的安全距离为 0.5 mm。凹槽中心点为 X60，Y40，最大进刀深度为 4 mm。

仅进行一次粗加工。



```

DEF REAL LENG, WID, DPR, CRAD          ; 变量定义
DEF INT VARI
N10 LENG=60 WID=40 DPR=17.5 CRAD=8    ; 赋值
N20 VARI=1
N30 G90 S600 M4                          ; 确定工艺数值
N35 T20 D2
N37 M6
N40 G17 G0 X60 Y40 Z5                    ; 返回运行到出发位置
N50 POCKET1 (5, 0, 0.5, , DPR, ->      ; 循环调用
-> LENG, WID, CRAD, 60, 40, 0, ->
-> 120, 300, 4, 2, 0.75, VARI)         ; 删除参数MIDF、FFP2 和 SSF
N60 M30                                  ; 程序结束

```

-> 必须在一个程序段中编程

### 3.8 铣削环形凹槽 — POCKET2



#### 编程

POCKET2 (RTP, RFP, SDIS, DP, DPR, PRAD, CPA, CPO, FFD, FFP1, MID, CDIR, FAL, VARI, MIDF, FFP2, SSF)



#### 参数

RTP	实数	退回平面（绝对）
RFP	实数	基准面（绝对）
SDIS	实数	安全距离（不输入符号）
DP	实数	凹槽深度（绝对）
DPR	实数	相对于基准面的槽深度（不输入符号）
PRAD	实数	凹槽半径（不输入符号）
CPA	实数	凹槽中心点，横坐标（绝对）
CPO	实数	凹槽中心点，纵坐标（绝对）
FFD	实数	深度方向的进给
FFP1	实数	表面加工的进给
MID	实数	一个横向进给的最大进刀深度（不输入符号）
CDIR	整数	凹槽加工的铣削方向 值： 2（用于 G2） 3（用于 G3）
FAL	实数	凹槽边缘的精加工余量（不输入符号）
VARI	整数	加工方式 值： 0=综合加工 1=粗加工 2=精加工
MIDF	实数	精加工最大进刀深度
FFP2	实数	精加工进给
SSF	实数	精加工时的转速



该循环要求一个铣刀带“过中心的端面齿切削”  
(DIN844)。

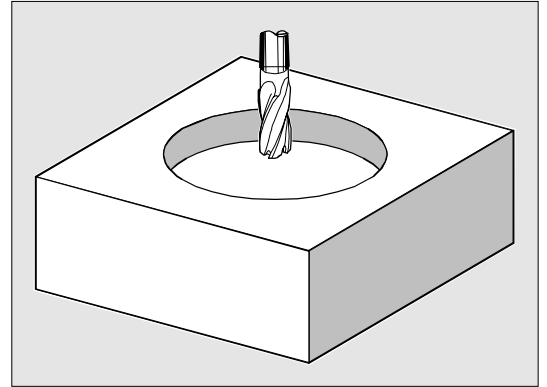


槽铣削循环POCKET4适用于使用任意刀具。



### 功能

该循环是一个组合的粗加工—精加工循环。  
使用该循环您可以在加工平面中加工环形凹槽。



### 工作流程

#### 循环开始之前到达的位置:

出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到槽中心点，没有轮廓碰撞。

#### 该循环产生以下的运动过程:

- 以 G0 凹槽中心点返回到退回平面高度，接着同样以 G0 在该位置上运行到提前了安全距离的基准面。  
综合加工时遵循下列步骤加工凹槽：
  - 以进给值 FFD 在凹槽中心点位置进刀到下一个加工深度。
  - 以进给 FFP1 和循环调用之前有效的主轴转速铣削出凹槽，直至精加工余量。
- 粗加工结束之后：
  - 进刀到下一个、通过 MIDF 确定的加工深度。
  - 以进给 FFP2 和转速 SSF 沿着轮廓进行精加工。
  - 加工方向按照 CDIR 下确定的方向。
- 在凹槽加工结束之后，刀具运行到槽中心点，直至退回平面，循环结束。



### 参数说明



参数 RTP、RFP、SDIS 参见  
章节 2.1.2. (钻削, 定中心 - CYCLE81)。  
参数 DP、DPR 参见章节 3.7。



参数 FFD、FFP1、MID、CDIR、FAL、VARI、  
MIDF、FFP2、SSF 参见章节 3.5 (SLOT1)。  
循环设定参数\_ZSD[1] 参见章节 3.2。

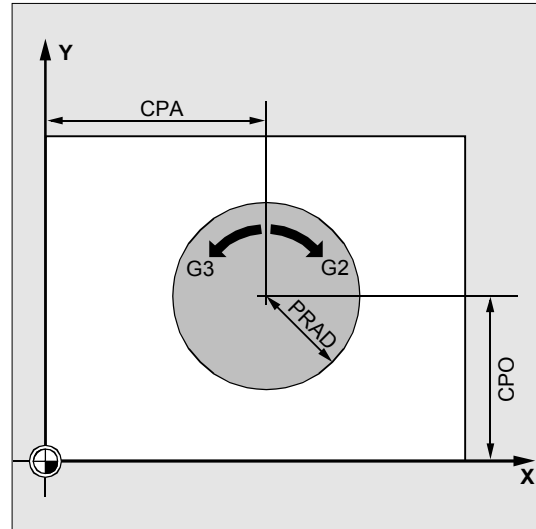
#### PRAD (凹槽半径)

环形凹槽的形状单独由其半径确定。

如果其半径小于当前刀具的半径, 则在发出报警  
61105 “铣刀半径太大” 之后循环中止。

#### CPA, CPO (凹槽中心点)

使用参数 CPA 和 CPO 您可以定义环形凹槽在横坐标和纵  
坐标上的中心点。



### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会中断  
并给出报警

61000 “没有刀具补偿有效”。

总是在凹槽中心点处进行深度进刀。也可以在那进行钻  
削。

在循环内部使用一个新的当前工件坐标系, 该坐标系  
影响实际值显示。该坐标系的零点就是凹槽中心点。

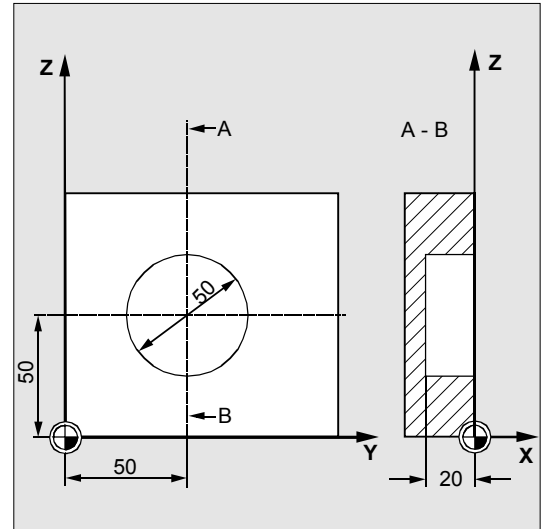
在循环结束之后, 原来的坐标系再次生效。



### 编程举例

#### 环形凹槽

使用该程序可以在 YZ 平面中加工一个环形凹槽。通过 Y50 Z50 确定中心点。深度进刀的进给轴为 X 轴，以绝对值给定凹槽深度。既不规定精加工余量也不规定安全距离。



```

DEF REAL RTP=3, RFP=0, DP=-20,-> ; 变量定义, 赋值
-> PRAD=25, FFD=100, FFP1, MID=6
N10 FFP1=FFD*2
N20 G19 G90 G0 S650 M3 ; 确定工艺数值
N25 T10 D1
N27 M6
N30 Y50 Z50 ; 返回运行到出发位置
N40 POCKET2 (RTP, RFP, , DP, , PRAD, -> ; 循环调用
-> 50, 50, FFD, FFP1, MID, 3, ) ; 删除参数FAL,VARI, MIDF, FFP2 和
SSF
N50 M30 ; 程序结束

```

-> 必须在一个程序段中编程

### 3.9 铣削矩形凹槽—POCKET3



循环POCKET3自软件版本4以后可以使用。



#### 编程

```
POCKET3 (_RTP, _RFP, _SDIS, _DP, _LENG, _WID, _CRAD, _PA, _PO, _STA,
_MID, _FAL, _FALD, _FFP1, _FFD, _CDIR, _VARI, _MIDA, _AP1, _AP2, _AD,
_RAD1, _DP1)
```



#### 参数

始终需要下面的输入参数：

_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到基准面，不输入符号）
_DP	实数	凹槽深度（绝对）
_LENG	实数	凹槽长度，在标注拐角尺寸时带符号
_WID	实数	凹槽宽度，在标注拐角尺寸时带符号
_CRAD	实数	凹槽拐角半径（不输入符号）
_PA	实数	凹槽基准点，横坐标（绝对）
_PO	实数	凹槽基准点，纵坐标（绝对）
_STA	实数	凹槽纵向轴和平面第一轴之间的夹角（横坐标，不输入符号）； 值范围： $0^\circ \leq \_STA < 180^\circ$
_MID	实数	最大进刀深度（不输入符号）
_FAL	实数	槽边缘的精加工余量（不输入符号）
_FALD	实数	底部精加工余量（不输入符号）
_FFP1	实数	表面加工的进给
_FFD	实数	深度方向的进给
_CDIR	整数	铣削方向：（不输入符号） 值： 0...同向铣削（与主轴转向一致） 1...逆向铣削 2...用 G2（与主轴转向无关） 3...用 G3
_VARI	整数	加工方式：（不输入符号） 个位： 值： 1...粗加工 2...精加工 十位： 值： 0...以G0垂直于凹槽中心 1...以G1垂直于凹槽中心 2...以螺旋轨迹 3...以凹槽纵向轴摆动



可选择性地规定其它参数。在扩孔时确定插入方案和叠加：（不输入符号）

_MIDA	实数	在平面中扩孔时最大的进刀宽度，作为数值
_AP1	实数	毛坯尺寸，凹槽长度
_AP2	实数	毛坯尺寸，槽宽度
_AD	实数	毛坯尺寸，槽到基准面的深度
_RAD1	实数	在插入时螺旋轨迹半径（与刀具中心点轨迹有关）或者摆动运动时最大的再入角
_DP1	实数	每360° 旋转时插入深度，在螺旋轨迹上插入



### 功能

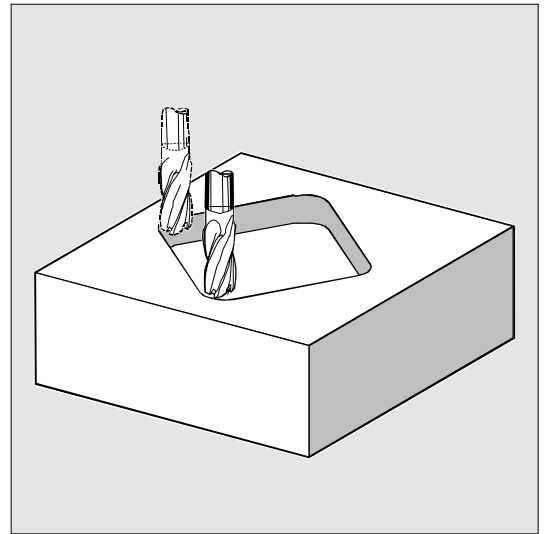
该循环可以用于粗加工和精加工。

在精加工时要求一个端面铣刀。

深度方向进刀始终从凹槽中心点开始，或者在那儿垂直进行；因此在此位置可以先预钻削孔。

#### 相对于POCKET1的新功能：

- 铣削方向可以由G指令（G2/G3）确定，也可以选择作为主轴转向的同向或者逆向进行铣削。
- 在扩孔时，可以编程平面中最大的进刀宽度
- 在凹槽底部也有精加工余量
- 三种不同的插入方案：
  - 垂直于凹槽中心
  - 在围绕凹槽中心的螺旋轨迹上
  - 摆动凹槽中间轴
- 在精加工时较短的行程返回平面
- 考虑平面中一个坯件轮廓和毛坯底部的尺寸（可以对预制的凹槽进行优化加工）。





### 工作流程

#### 循环开始之前到达的位置：

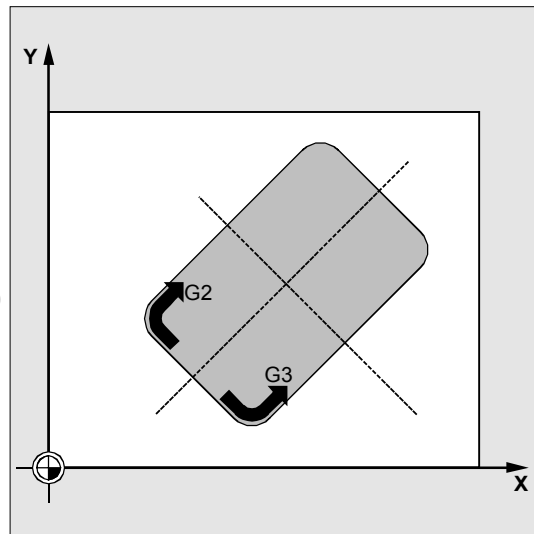
出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到槽中心点，没有轮廓碰撞。

#### 粗加工时的运动过程（VARI=X1）：

以 G0 凹槽中心点返回到退回平面高度，接着同样以 G0 在该位置上运行到提前了安全距离的基准面。随后，根据所选的插入方案并在考虑到编程的毛坯尺寸情况下进行凹槽加工。

#### 插入方案：

- 垂直于凹槽中心插入（VARI=0X, VARI=1X）表示在一个带G0或者G1的程序段中执行循环内部所计算的进刀深度（≤在\_MID下编程的最大的进刀深度）。
- 插入至螺旋线轨迹（VARI=2X）表示，铣刀中心点运行至通过半径 \_RAD1 和深度每转\_DP1 确定的螺旋线轨迹。同样在 \_FFD 下编程进给。螺旋线轨迹的转向与用以加工凹槽的转向相符。插入时在\_DP1下编程的深度作为最大深度计算，并且始终是整数个螺旋轨道的转数。在到达进刀实际深度时（可能为螺旋轨道上多次转数），还要执行一个整圆，从而消除插入时的倾斜轨迹。这之后在平面中扩孔凹槽，直至精加工余量。所说明的螺旋轨道的起始点位于凹槽纵向轴的“正方向”，并且用G1返回。



- 摆动插入到凹槽中间轴 (VARI=3X)

表示，铣刀中心点沿一条直线前后摆动斜向插入，直至其到达下一个当前深度。

在 `_RAD1` 下编程最大插入角度，在循环内部计算摆动位移长度。如果到达当前深度，则还要执行一次不带深度进刀的位移，从而消除插入时的倾斜轨迹。在 `_FFD` 下编程进给。

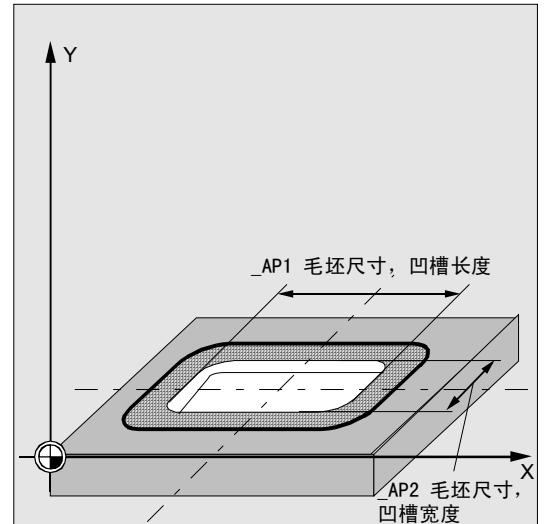
#### 考虑毛坯尺寸

在对凹槽扩孔时可以考虑毛坯尺寸（比如对事先浇注部件的加工）。

毛坯尺寸在长度和宽度上 (`_AP1` 和 `_AP2`) 不用符号编程并由循环以对称于凹槽中心点的方式计算设定。确定不在需要扩孔的凹槽部分。同样，毛坯尺寸在深度上 (`_AD`) 不用符号编程并由基准面以凹槽深度方向计算得出。

在考虑坯件尺寸时，根据编程的方式（螺旋轨迹，摆动，垂直）进行深度进刀。如果循环识别出所给定的坯件轮廓和当前刀具的半径在凹槽中心有足够的位置，则只要有可能，在凹槽中心点处垂直向下进刀，从而不用在空中运行费事的插入轨迹。

凹槽从上往下开始进行扩孔。



### 精加工时运行过程 (VARI=X2)

精加工的进行顺序为：先是边缘精加工，一直加工到底部精加工余量处，然后是底部精加工。如果精加工余量等于零，则去除该精加工部分。

- 边缘的精加工

在边缘部分进行精加工时每次仅绕行一次凹槽。

为了进行边缘精加工，沿一个通向拐角半径的四一圆弧轨迹返回。该轨迹的半径尺寸通常为 **2 mm** 或者拐角半径和铣刀半径之间的差值“位置较少”。如果边缘处精加工余量大于2毫米，则驶入半径要相应地增大。

深度进刀以G0在空切削回到凹槽中心，并且同样以G0到达驶入轨迹的起始点。

- 底部的精加工

为了进行底部的精加工，以 G0

返回到凹槽中心一直到凹槽深度 + 精加工余量 + 安全距离。自那开始总是以深度进刀的进给垂直运行到深度（因为用于底部的精加工占了一个刀具，这可通过端面进行铣削）。

凹槽的底部面积一次加工。



### 参数说明

参数 `_RTP`、`_RFP`、`_SDIS` 参见章节 2.1.2. (钻削，定中心 - CYCLE81)。

参数 `_DP` 参见章节 3.7。

循环设定参数 `_ZSD[1]`，`_ZSD[2]` 参见章节 3.2。

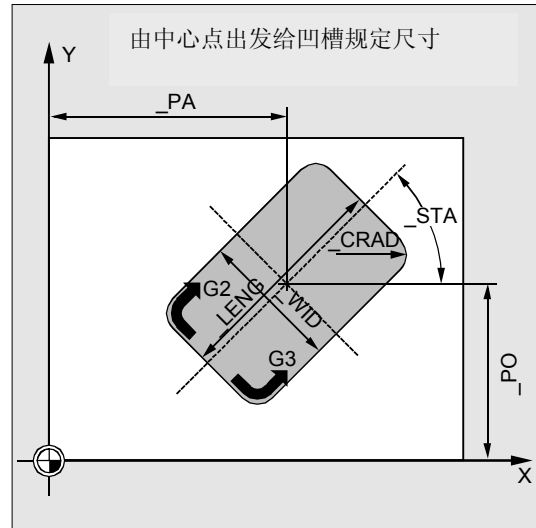
#### `_LENG`，`_WID` 和 `_CRAD`

(凹槽长度，凹槽宽度和拐角半径)

使用参数 `_LENG`，`_WID` 和

`_CRAD`，您可以确定平面中一个凹槽的形状。

此时，可由中心或由角点出发给凹槽规定尺寸。测量拐角时用符号输入 `_LENG` 和 `_WID`。



如果当前的刀具不能运行编程的拐角半径（因为其半径太大），则加工完毕的凹槽拐角半径等于刀具半径。

如果刀具的铣刀半径大于凹槽一半的长度或者宽度，则给出报警

61105 “铣刀半径太大”之后，循环中断。

#### **\_PA, \_PO (基准点)**

使用参数 **\_PA** 和 **\_PO** 您可以定义凹槽在横坐标和纵坐标上的基准点。

该基准点或者是凹槽中心点，或者一个角点。评估该参数取决于循环设定数据位 **\_ZSD[2]**：

- 0 表示凹槽中心点
- 1 表示拐角点

如果从一个拐角进行凹槽的尺寸标注，则长度和宽度（**\_LENG**, **\_WID**）的参数带符号输入，由此使凹槽的位置可以明确确定。

#### **\_STA (角度)**

**\_STA** 说明平面中第一轴（横坐标）和凹槽纵向轴之间的夹角。

#### **\_MID (进刀深度)**

通过这些参数，您可以确定粗加工时最大的进刀深度。

在循环中深度以均匀的进刀尺寸进行。

利用 **\_MID** 和总深度，该循环自动计算进刀量。存放进刀尺寸最小可能的数量。

**\_MID=0** 表示一步进刀到凹槽深度。

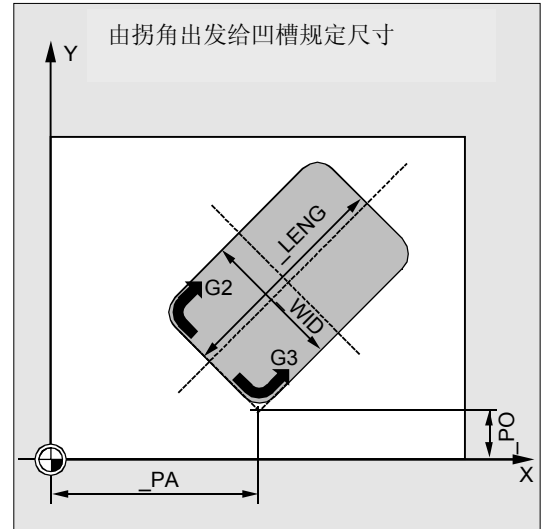
#### **\_FAL (边缘的精加工余量)**

该精加工余量仅在平面中加工凹槽边缘时生效。

如果精加工余量  $\geq$  刀具直径，则不保证凹槽进行完整的扩孔。显示提示信息：

“注意：精加工余量  $\geq$  刀具直径”

但是该循环继续。



**\_FALD（底部的精加工余量）**

在粗加工时考虑在底部进行一次单独的精加工余量（POCKET1在底部没有精加工余量）。

**\_FFD 和 \_FFP1（进给深度和面积）**

进给\_FFDD在插入到原料时生效。

进给\_FFPP1在平面中以进给运行的所有运动中生效。

**\_CDIR（铣削方向）**

在此参数下规定凹槽的加工方向。

通过参数\_CDIDR可以

- 直接编程铣削方向“2用于G2”和“3用于G3”，或者
- 选择“同向运行”或者“逆向运行”。同向运行或者逆向运行由循环在内部通过循环调用之前激活的主轴方向确定。

<b>同向运行</b>	<b>逆向运行</b>
M3 → G3	M3 → G2
M4 → G2	M4 → G3

**\_VARI（加工方式）**

使用参数\_VARI您可以确定加工方式。

可能的值是：

**个位：**

- 1=粗加工
- 2=精加工

**十位（进刀）：**

- 0=以G0垂直于凹槽中心
- 1=以G1垂直于凹槽中心
- 2=螺旋轨迹
- 3=摆动凹槽纵向轴

如果参数\_VARI编程一个其它值，则循环中断并发出报警61002“加工方式错误定义”。

**\_MIDA（最大进刀宽度）**

使用该参数您可以确定在平面中扩孔时的最大进刀宽度。类似于已知的进刀深度计算（用最大可能的值平均分配总深度）平均分配宽度，最大带有在\_MIDA

下编程的值。

如果该参数没有编程，或者为值0，则该循环在内部以80%的铣刀直径作为最大的进刀宽度。

#### 其它说明

在深度上达到完整的凹槽时，由边缘加工所计算的宽度进刀重新计算，在这种情况下适用。否则保持开始时所计算的宽度进刀（用于整个循环）。

#### **\_AP1, \_AP2, \_AD**（毛坯尺寸）

使用参数\_AP1、\_AP2 和 \_AD定义平面中凹槽的毛坯尺寸（增量方式）和深度。

#### **\_RAD1**（半径）

使用参数\_RAD1定义螺旋轨迹的半径（与刀具中心点轨迹相关）或者用于摆动运动的最大再入角。

#### **\_DP1**（插入深度）

使用参数\_DP1定义螺旋轨迹上插入时的进刀深度。

### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会中断并给出报警

61000 “无刀具补偿有效”。

在循环内部使用一个新的当前工件坐标系，该坐标系影响实际值显示。该坐标系的零点就是凹槽中心点。在循环结束之后，原来的坐标系再次生效。

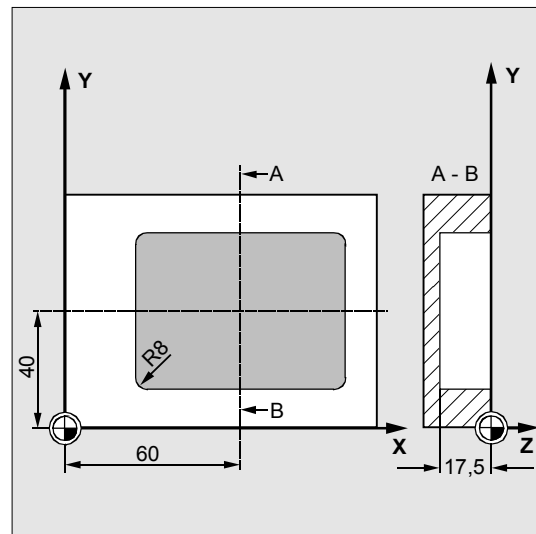
### 编程举例

#### 凹槽

使用此程序您可以在 XY 平面中加工一个凹槽，其长度为60毫米，宽度40毫米，拐角半径8毫米，深度17.5毫米。凹槽与 X 轴成 0 度角。凹槽边缘的精加工余量为 0.75 mm，在底部为 0.2 mm，Z 轴中增加到基准面上的安全距离为 0.5 mm。凹槽中心点是 X60 和 Y40，最大进刀深度为 4 mm。

由主轴转向和同向铣削形成加工方向。

仅进行一次粗加工。



```

N10 G90 S600 M4 ; 确定工艺数值
N15 T10 D1
N17 M6
N20 G17 G0 X60 Y40 Z5 ; 返回运行到出发位置
N25 _ZSD[2]=0 ; 通过中心点标注凹槽尺寸
N30 POCKET3 (5, 0, 0.5, -17.5, 60, -> ; 循环调用
-> 40, 8, 60, 40, 0, 4, 0.75, 0.2, ->
-> 1000, 750, 0, 11, 5)
N40 M30 ; 程序结束

```

-> 必须在一个程序段中编程



### 3.10 铣削环形凹槽 - POCKET4



循环POCKET4自软件版本4以后可以使用。



#### 编程

POCKET4 (\_RTP, \_RFP, \_SDIS, \_DP, \_PRAD, \_PA, \_PO, \_MID, \_FAL, \_FALD, \_FFP1, \_FFD, \_CDIR, \_VARI, \_MIDA, \_AP1, \_AD, \_RAD1, \_DP1)



#### 参数

下面的输入参数始终要求:

_RTP	实数	退回平面 (绝对)
_RFP	实数	基准面 (绝对)
_SDIS	实数	安全距离 (加到基准面, 不输入符号)
_DP	实数	凹槽深度 (绝对)
_PRAD	实数	凹槽半径
_PA	实数	凹槽中心点, 横坐标 (绝对)
_PO	实数	凹槽中心点, 纵坐标 (绝对)
_MID	实数	最大进刀深度 (不输入符号)
_FAL	实数	槽边缘的精加工余量 (不输入符号)
_FALD	实数	底部精加工余量 (不输入符号)
_FFP1	实数	表面加工的进给
_FFD	实数	深度方向的进给
_CDIR	整数	铣削方向: (不输入符号) 值: 0...同向铣削 (与主轴旋转方向一致) 1...逆向铣削 2...用 G2 (与主轴旋转方向无关) 3...用 G3
_VARI	整数	加工方式: (不输入符号) 个位: 值: 1...粗加工 2...精加工 十位: 值: 0...以G0垂直于凹槽中心 1...以G1垂直于凹槽中心 2...以螺旋轨迹
可选择性地规定其它参数。在扩孔时确定插入策略和叠加: (不输入符号)		
_MIDA	实数	在平面中扩孔时最大的进刀宽度, 作为数值
_AP1	实数	毛坯尺寸, 凹槽半径
_AD	实数	毛坯尺寸, 槽到基准面的深度
_RAD1	实数	插入时螺旋轨迹的半径 (与刀具中心点轨迹相关)
_DP1	实数	每360°-旋转时插入深度, 在螺旋轨迹上插入



## 功能

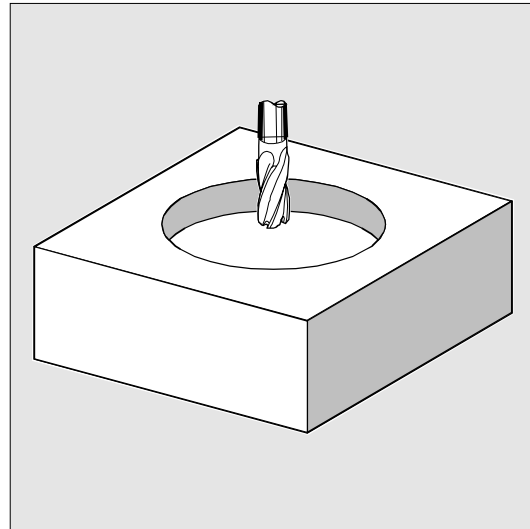
使用该循环您可以在加工平面中加工环形凹槽。

在精加工时要求一个端面铣刀。

深度方向进刀始终从凹槽中心点开始，或者在那儿垂直进行；因此在此位置可以先预钻削孔。

### 相对于POCKET2的新功能：

- 铣削方向可以由G指令（G2/G3）确定，也可以选择作为主轴旋转方向的同向或者逆向进行铣削。
- 在扩孔时，可以编程平面中最大的进刀宽度
- 在凹槽底部也有精加工余量
- 两种不同的插入方案：
  - 垂直于凹槽中心
  - 在围绕凹槽中心的螺旋轨迹上
- 在精加工时较短的行程返回平面
- 考虑平面中一个坯件轮廓和毛坯底部的尺寸（可以对预制的凹槽进行优化加工）。
- `_MIDA` 在边缘加工时重新计算。



## 工作流程

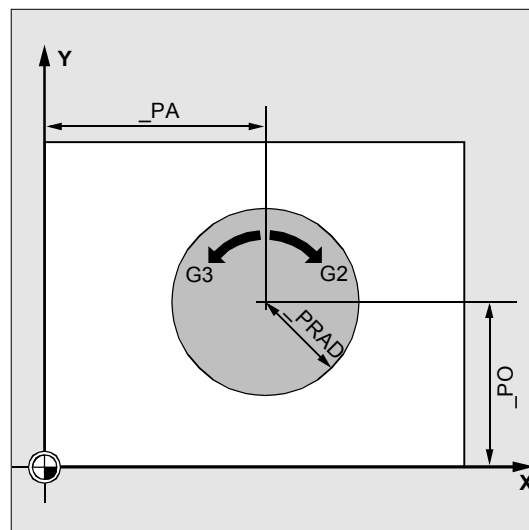
### 循环开始之前到达的位置：

出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到槽中心点，没有轮廓冲突。

### 粗加工时的运动过程（`VARI=X1`）：

以G0凹槽中心点返回到退回平面高度，接着同样以G0在该位置上运行到提前了安全距离的基准面。

随后，根据所选的插入方案并在考虑到编程的毛坯尺寸情况下进行凹槽加工。



**插入方案:**

参见章节 3.9 (POCKET3)

**考虑坯件尺寸**

在对凹槽扩孔时可以考虑坯件尺寸（比如对事先浇注部件的加工）。

在环形凹槽中，毛坯尺寸  $\_AP1$  同样是一个圆弧（使用较小的半径作为凹槽半径）。



其它说明参见章节 3.9 (POCKET3)

**精加工时运动过程 (VARI=X2) :**

精加工的进行顺序为：先是边缘精加工，一直加工到底部精加工余量处，然后是底部精加工。如果精加工余量等于零，则去除该精加工部分。

- 边缘的精加工

在边缘部分进行精加工时每次仅绕行一次凹槽。

为了进行边缘精加工，沿一个通向拐角半径的四一圆弧轨迹返回。轨迹半径尺寸最大为

2毫米，或者拐角半径和铣刀半径之间的差值“位置较少”。

深度进刀以 G0 在空运行中回到凹槽中心，

并且同样以 G0 到达驶入轨迹的起始点。

- 底部的精加工

为了进行底部的精加工，以 G0 返回到凹槽

中心一直到凹槽深度 + 精加工余量 +

安全距离。自那开始总是以深度进刀的进给垂直运行到深度（因为接受一个刀具用于底部的精加工，这可通过端面铣削）。

凹槽的底部面积一次加工。



### 参数说明

参数 `_RTP`, `_RFP`, `_SDIS` 参见

章节 2.1.2. (钻削, 定中心 - CYCLE81)。

参数 `_DP` 参见章节 3.7 (POCKET1)。

参数 `_MID`、

`_FAL`、`_FALD`、`_FFP1`、`_FFD`、`_CDIR`、`_MIDA`、`_AP1`、`_AD`、`_RAD1`、`_DP1` 参见章节 3.9 (POCKET3)。

循环设定参数 `_ZSD[1]` 参见章节 3.2。

#### `_PRAD` (凹槽半径)

环形凹槽的形状单独由其半径确定。

如果其半径小于当前刀具的半径, 则在发出报警61105  
“铣刀半径太大”之后循环中止。

#### `_PA`, `_PO` (凹槽中心点)

使用参数 `_PA` 和 `_PO`

定义凹槽中心点。环形凹槽始终通过中心进行尺寸标注。

#### `_VARI` (加工方式)

使用参数 `_VARI` 您可以确定加工方式。

可能的值是:

个位:

- 1=粗加工
- 2=精加工

十位 (进刀):

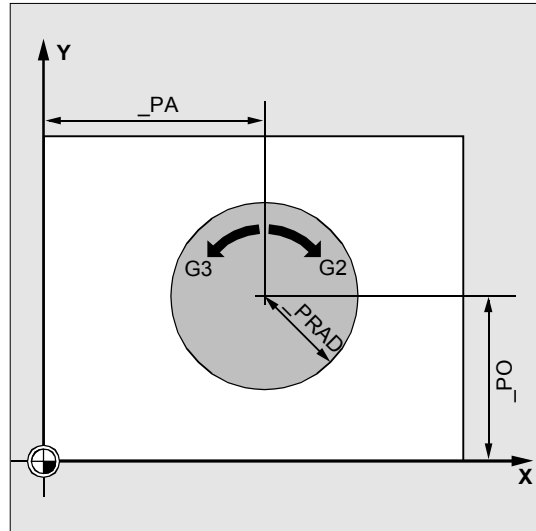
- 0=以 G0 垂直于凹槽中心
- 1=以 G1 垂直于凹槽中心
- 2=螺旋轨迹

如果参数 `_VARI` 编程一个其它值, 则循环中断并发出报警 61002 “加工方式错误定义”。

#### 边缘的粗加工, 用 **POCKET4**

**POCKET4** (环形凹槽) 现在仅可以进行边缘的粗加工。

为此必须这样定义深度的毛坯尺寸 (参数 `_AD`), 至少使它等于凹槽深度 (`DP`) 减去精加工余量深度 (`_FALD`)。



举例：

前提条件：深度计算，不算入安全距离

( $\_ZSD[1]=1$ )。

$\_RTP=0$             基准面  
 $\_SDIS=2$            安全距离  
 $\_DP=-21$            凹槽深度  
 $\_FALD=1.25$        余量深度  
 $\rightarrow \_AD \geq 19.75$    毛坯尺寸深度  
                          必须大于或者等于凹槽深度  
                          增量减去余量深度，  
                          也就是  $21-1.25=19.75$

### 其它说明

在循环调用之前激活一个刀具补偿。否则循环会被中断并发出报警 61000 “无刀具补偿有效”。

在循环内部使用一个新的当前工件坐标系，该坐标系影响实际值显示。该坐标系的零点就是凹槽中心点。

在循环结束之后，原来的坐标系再次生效。

### 编程举例

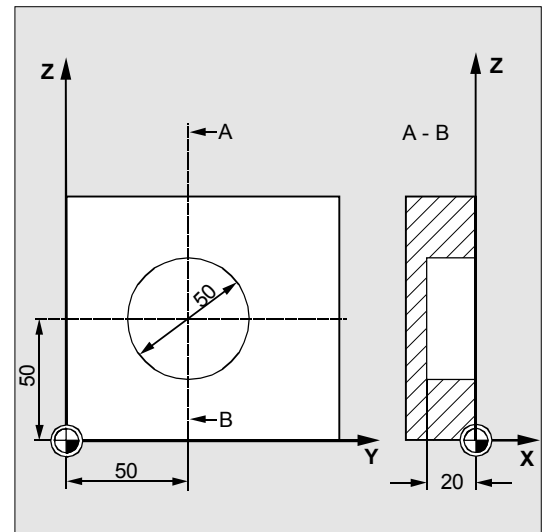
#### 环形凹槽

使用该程序可以在 YZ

平面中加工一个环形凹槽。通过 Y50 Z50

确定中心点。深度进刀的进给轴就是 X

轴。既不规定精加工余量也不规定安全距离。用逆向运行铣削加工凹槽。沿着一个螺旋线轨迹进刀。



```
N10 G19 G90 G0 S650 M3
```

；确定工艺数值

```
N15 T20 D1
```

```
N17 M6
```

```
N20 Y50 Z50
```

；返回运行到出发位置

```
N30 Pocket4(3, 0, 0, -20, 25, 50, -> ; 循环调用
50, 6, 0, 0, 200, 100, 1, 21, 0, ->
0, 0, 2, 3)
```

```
N40 M30
```

；程序结束

-> 必须在一个程序段中编程

### 3.11 平面铣削 - CYCLE71



循环CYCLE71自软件版本4以后可以使用。



#### 编程

CYCLE71 (\_RTP, \_RFP, \_SDIS, \_DP, \_PA, \_PO, \_LENG, \_WID, \_STA, \_MID, \_MIDA, \_FDP, \_FALD, \_FFP1, \_VARI, \_FDP1)



#### 参数

下面的输入参数始终要求：

_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到基准面，不输入符号）
_DP	实数	深度（绝对）
_PA	实数	起始点，横坐标（绝对）
_PO	实数	起始点，纵坐标（绝对）
_LENG	实数	矩形在第一轴上的长度，增量。 由符号给出此角（由此角标注尺寸）。
_WID	实数	矩形在第二轴上的长度，增量。 由符号给出此角（由此角标注尺寸）。
_STA	实数	矩形纵向轴和平面第一轴之间的夹角（横坐标，不输入符号）； 值范围： $0^\circ \leq \_STA < 180^\circ$
_MID	实数	最大进刀深度（不输入符号）
_MIDA	实数	在平面中进行剥离时最大的进刀宽度，作为数值（不输入符号）
_FDP	实数	切削方向空运行行程（增量，不输入符号）
_FALD	实数	深度方向的精加工余量（增量，不输入符号）在精加工加工方式中，_FALD表示表面上的剩余材料。
_FFP1	实数	表面加工的进给
_VARI	整数	加工方式：（不输入符号） 个位： 值： 1...粗加工 2...精加工 十位： 值： 1...平行于横坐标，在一个方向 2...平行于纵坐标，在一个方向 3...平行于横坐标，方向交替 4...平行于纵坐标，方向交替
_FDP1	实数	在平面横向进给方向溢出行程（增量，不输入符号）

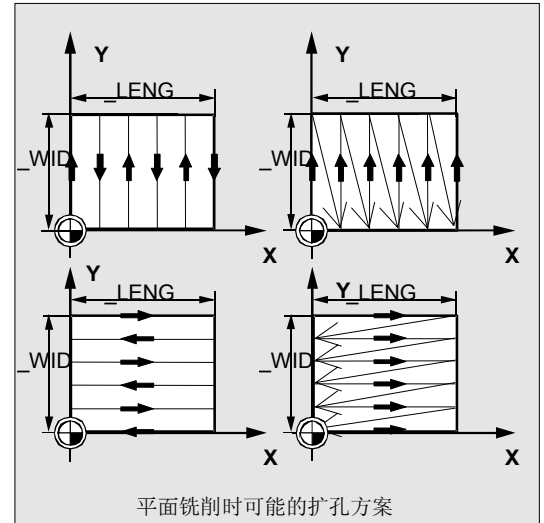


## 功能

使用循环 CYCLE71

可切削任意的矩形表面。循环在粗加工（以多步扩孔表面直到精加工余量）和精加工（对表面进行一次一次性铣削）之间加以区别。可以规定宽度和深度方向的最大进刀量。

循环在没有铣刀半径补偿的情况下工作。在空运行中进行深度进刀。



## 工作流程

**循环开始之前到达的位置：**

出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到进刀点，没有轮廓冲突。

**该循环产生以下的运动过程：**

- 以 G0 进刀点返回到当前位置高度，接着同样以 G0 在该位置上运行到提前了安全距离的基准面。随后，同样以 G0 进刀至加工平面。可以是 G0，因为进刀处在空运行状态中。可以预定几个材料铣削方案（轴向平行在一个方向或者来回方向）。
- 粗加工时的运动过程（VARI=X1）：可根据编程的值 `_DP`、`_MID` 和 `_FALD` 在多个平面上进行平面铣削。此时，从上向下加工，也就是说，分别扩孔一个平面，然后在空运行（参数 `_FDP`）中进行下一个深度进刀。在剥离平面时运行位移取决于参数 `_LENG`、`_WID`、`_MIDA`、`_FDP`、`_FDP1` 和当前刀具铣刀半径的数值。第一个待铣削的轨迹运行时，进刀宽度总是正好为 `_MIDA`，以便没有大于最大可能的宽度

进刀的状况出现。由此，刀具中心点不总是精确地在棱边上运行（仅在  $\_MIDA =$  铣刀半径时）。棱边外侧的刀具所运行的尺寸总是铣刀直径  $\_MIDA$ ，当在平面中仅进行一次切削时也是如此，即表面宽度 + 溢出行程小于  $\_MIDA$ 。在内部计算出宽度进刀的其它轨迹，以得出平均的轨迹宽度 ( $\leq \_MIDA$ )。

- 精加工时的运行过程 ( $VARI=X2$ ) :

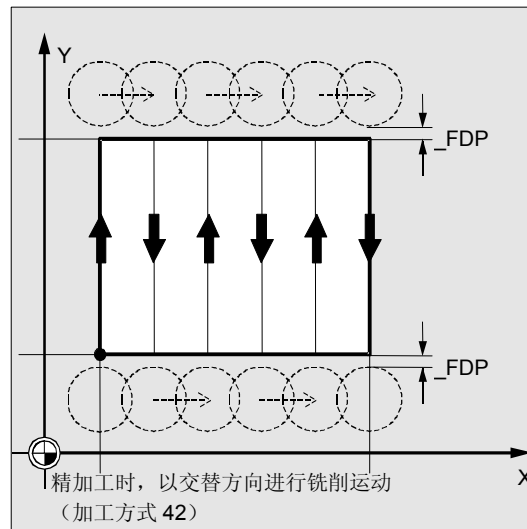
精加工时，在平面中铣削表面一次。粗加工时，也必须如此选择精加工余量，以能够用精加工刀具同时处理剩余的深度。

在每次进行完一次性铣削后，刀具在平面中真正进行空运行。在参数  $\_FDP$  下编程空运行位移。

在一个方向加工时退刀精加工余量 + 安全距离，并且以快速移动返回到下一个起始点。

在一个方向上粗加工时，以计算的进刀深度 + 安全距离退刀。同粗加工，在相同的点上进行深度进刀。

在结束精加工之后，刀具退回到最后到达的位置，直至退回平面  $\_RTP$ 。



### 参数说明

参数  $\_RTP$ 、 $\_RFP$ 、 $\_SDIS$  参见章节 2.1.2 (钻削，定中心 - CYCLE81)。

参数  $\_STA$ 、 $\_MID$ 、 $\_FFP1$  参见章节 3.9 (POCKET3)。

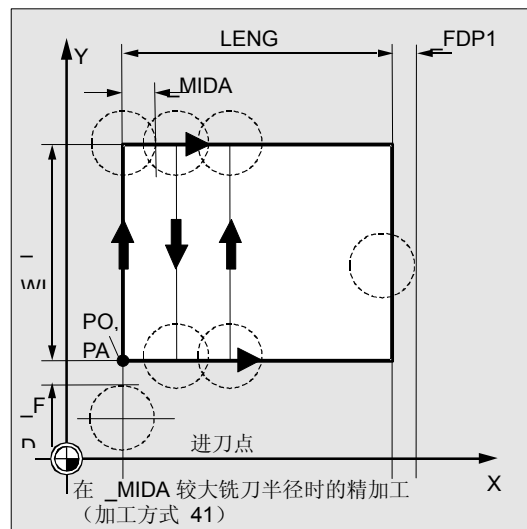
循环设定参数  $\_ZSD[1]$  参见章节 3.2。

#### $\_DP$ (深度)

深度可以以基准面进行绝对值规定 ( $\_DP$ )。

#### $\_PA$ , $\_PO$ (起始点)

使用参数  $\_PA$  和  $\_PO$  您可以在横坐标和纵坐标上定义平面的起始点。





**\_LENG, \_WID (长度)**

使用参数 **\_LENG** 和 **\_WID**

确定平面中的矩形长度和宽度。由符号得出矩形位置，与 **\_PA** 与 **\_PO** 相关。

**\_MIDA (最大进刀宽度)**

使用参数确定扩孔加工某个平面时的进刀宽度。类似于已知的进刀深度计算（用最大可能的值平均分配总深度）平均分配宽度，最大的带有在 **\_MIDA** 下编程的值。如果该参数没有编程，或者为值0，则该循环在内部以80% 的铣刀直径作为最大的进刀宽度。

**\_FDP (空运行行程)**

使用该参数您可以确定平面中空运行行程的尺寸。必须以大于0 的值编程该参数。

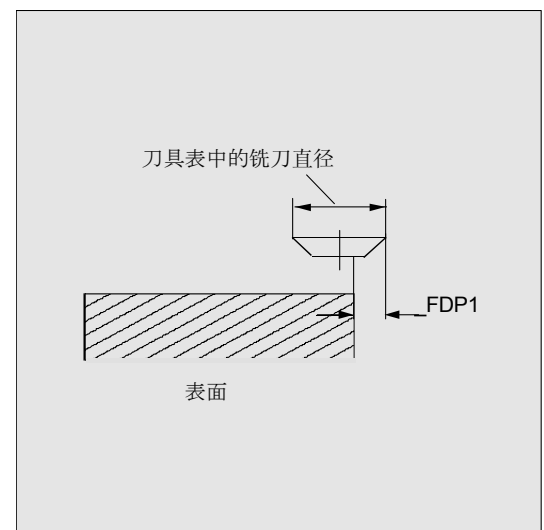
**\_FDP1 (溢出行程)**

使用参数可以规定平面进刀方向上的溢出行程（**\_MIDA**）。由此可补偿当前铣刀半径和刀尖（如刀沿半径和斜置的刀片）之间的差值。因此，最后的铣刀中心点轨迹得出的结果总是为 **\_LENG**（或 **\_WID**）+ **\_FDP1** - 刀具半径（补偿表）。

**\_FALD (精加工余量)**

在粗加工时考虑深度方向的精加工余量，该余量在该参数下编程。

在精加工时必须说明剩余材料，它们作为精加工附加量剩余，这样在退刀时和接着进刀到下一个切削的起始点时才不至于有轮廓冲突。



### **\_VARI (加工方式)**

使用参数 `_VARI` 您可以确定加工方式。

可能的值是：

#### **个位：**

- 1=粗加工，直至精加工余量
- 2=精加工

#### **十位：**

- 1=平行于横坐标，在一个方向
- 2=平行于纵坐标，在一个方向
- 3=平行于横坐标，方向交替
- 4=平行于纵坐标，方向交替

如果参数 `_VARI` 编程一个其它值，则循环中断并发出报警61002 “加工方式错误定义”。

### **其它说明**

在循环调用之前激活一个刀具补偿。否则循环会中断并发出报警

61000 “没有刀具补偿有效”。



### 编程举例

#### 表面的平面铣削

##### 循环调用参数:

- 退回平面: 10 mm
- 基准面: 0 mm
- 安全距离: 2 mm
- 铣削深度: -11 mm
- 最大进刀深度 6 mm
- 没有精加工余量 -
- 矩形的起始点 X = 100 mm  
Y = 100 mm
- 矩形尺寸 X = +60 mm  
Y = +40 mm
- 平面中旋转角 10度
- 最大进刀宽度 10 mm
- 在铣削轨迹结束处的空运行行程: 5 mm
- 表面加工的进给: 4,000 mm/min
- 加工方式: 粗加工平行于X轴,  
在交替的方向
- 最后切削的超程由切削几何量决定 2 mm

```

%_N_TSTCYC71_MPF ; 程序, 用CYCLE71进行平面铣削
; $PATH=/_N_MPF_DIR
;*
$TC_DP1[1,1]=120 ; 刀具类型
$TC_DP6[1,1]=10 ; 刀具半径
N100 T1
N102 M06
N110 G17 G0 G90 G54 G94 F2000 X0 Y0 Z20 ; 返回运行到出发位置
;
CYCLE71( 10, 0, 2,-11, 100, 100, -> ; 循环调用
-> 60, 40, 10, 6, 10, 5, 0, 4000, 31, 2)
N125 G0 G90 X0 Y0
N130 M30 ; 程序结束
-> 必须在一个程序段中编程

```

### 3.12 轨迹铣削 - CYCLE72



循环CYCLE72自软件版本4起可以使用  
(不适用于FM-NC)。



#### 编程

CYCLE72 (\_KNAME, \_RTP, \_RFP, \_SDIS, \_DP, \_MID, \_FAL, \_FALD, \_FFP1, \_FFD, \_VARI, \_RL, \_AS1, \_LP1, \_FF3, \_AS2, \_LP2)



#### 参数

下面的输入参数始终要求:

_KNAME	字符串	轮廓子程序名
_RTP	实数	退回平面 (绝对)
_RFP	实数	基准面 (绝对)
_SDIS	实数	安全距离 (加到基准面, 不输入符号)
_DP	实数	深度 (绝对)
_MID	实数	最大进刀深度 (增量, 不输入符号)
_FAL	实数	在边缘轮廓处的精加工余量 (不输入符号)
_FALD	实数	底部精加工余量 (增量, 不输入符号)
_FFP1	实数	表面加工的进给
_FFD	实数	用于深度进刀的进给 (不输入符号)
_VARI	整数	加工方式: (不输入符号) 个位: 值: 1...粗加工 2...精加工 十位: 值: 0...以 G0 进行的中间位移 1...以 G1 进行的中间位移 百位: 值: 0...在轮廓结束处退回, 直至 _RTP 1...在轮廓结束处退回, 直至 _RFP + _SDIS 2...在轮廓结束处退回 _SDIS 3...在轮廓结束处没有退回
_RL	整数	中间、右侧或者左侧绕行轮廓 (使用 G40, G41 或者 G42, 没有输入符号) 值: 40...G40 (返回运行和离开运行, 仅以直线) 41...G41 42...G42

_AS1	整数	返回运行方向和轨迹的规定（不输入符号）
		个位： 值： 1...直线切向 2...四分之一圆弧 3...半圆  十位： 值： 0...在平面中返回运行到轮廓 1...以一个空间轨迹返回运行到轮廓
_LP1	实数	返回运行位移的长度（直线）或者驶入圆弧的半径（圆弧时）（不输入符号）
其它的参数可以选择性规定（不输入符号）。		
_FF3	实数	退回运行时进给和在平面中中间定位时的进给（空运行）
_AS2	整数	返回运行方向和轨迹的规定（不输入符号）  个位： 值： 1...直线切向 2...四分之一圆弧 3...半圆  十位： 值： 0...在平面中从轮廓运行出来 1...以一个空间轨迹从轮廓运行出来
_LP2	实数	离开运行位移的长度（直线）或者离开运行圆弧的半径（圆弧时）（不输入符号）



## 功能

使用循环 CYCLE72

可沿着任意一个在子程序中定义的轮廓铣削。循环在有或没有铣刀半径补偿的情况下工作。

该轮廓不必强制封闭，内部加工或者外部加工通过铣刀半径补偿的位置（中心、轮廓左侧或者右侧）进行定义。

**必须以要铣削的方向编程轮廓并放在一个平面中。**此外，必须由至少 2

个轮廓程序段（起始点和终点）组成，因为轮廓程序在轮廓内部直接调用。

### 循环的功能：

- 选择粗加工（一次轮廓平行的绕行，考虑一个精加工余量，有时分为几个深度直至精加工余量）和精加工（绕行最终轮廓一次，有时分为几个深度）
- 柔性返回运行/离开运行轮廓，可选择切向或者径向（四分之一或者半圆）
- 深度进刀可编程
- 中间运动可选择快速移动或者以进给方式



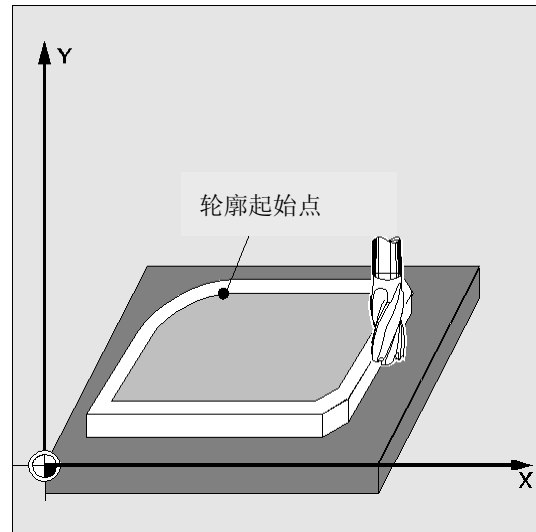
执行此循环的前提条件就是一个NC软件版本（4.3以后的版本），它包含功能“柔性返回运行和离开运行”。



## 工作流程

### 循环开始之前到达的位置：

出发位置是一个任意位置，从这个位置可以在退回平面的高度返回到轮廓起始点，没有轮廓冲突。



在粗加工时该循环产生以下的运动过程

(VARI=XX1) :

深度进刀以最大可能的数值根据所规定的参数平均分配。

- 用 G0/G1 (和 \_FF3)运行到起始点，用于第一次铣削。该起始点由系统内部计算，并且取决于
  - 轮廓起始点（子程序中第一个点），
  - 起始点轮廓的方向，
  - 返回运行模式及其参数和
  - 刀具半径。

在此程序段中开启铣刀半径补偿。

- 用 G0/G1 深度进刀第一个或下一个加工深度 + 编程的安全距离（参数 \_SDIS）。第一个加工深度由
  - 总深度，
  - 精加工余量和
  - 最大可能的深度进刀得出。
- 用深度进给率垂直返回运行至轮廓，然后在平面中用为表面加工而编程的进给率，或用带有根据柔性返回运行的编程而在 \_FAD 下编程的进给率的3D返回运行。
- 沿着轮廓以G40/G41/G42进行铣削。
- 以G1和用于表面加工的进给柔性离开运行轮廓，运行循环内部计算的退刀量。
- 以G0/G1退回（和用于中间位移\_FF3的进给），视编程而定。
- 以G0/G1（和 \_FF3）退回到深度进刀点。
- 在下一个加工平面重复该过程，直至在深度方向达到精加工余量。

在结束粗加工之后，刀具位于轮廓的离开运行点之上（控制系统内部计算），在退回平面高度。

在精加工时该循环产生以下的运行过程

(VARI=XX2) :

在精加工时以每次的进刀量沿着轮廓进行铣削，直至到达底部的成品尺寸。

轮廓根据现有的参数相应柔性返回运行或离开。为此在控制系统内部计算轨迹。

在循环结束之后，刀具位于轮廓的出发点，在退回平面高度。

### 轮廓编程

对轮廓进行编程时要注意以下几点：-{}-

- 在子程序中，在第一个编程的位置之前不允许选择可编程的框架（TRANS、ROT、SCALE、MIRROR）。
- 轮廓子程序的第一个程序段是一个带G90、G0的直线程序段，它定义轮廓的开始。
- 铣刀半径补偿由上一级的循环进行选择 and 撤消选择，因此在轮廓子程序中不编程G40、G41、G42。





### 参数说明



参数 `_RTP`、`_RFP`、`_SDIS` 参见  
 章节 2.1.2 (钻削, 定中心 - CYCLE81)。



参数 `_MID`、`_FAL`、`_FALD`、`_FFP1`、`_FFD`  
 参见章节 3.9。参数 `_DP` 参见章节 3.11。  
 循环设定参数 `_ZSD[1]` 参见章节 3.2。

### `_KNAME` (名称)

在子程序中完整编程要铣削的轮廓。用 `_KNAME`  
 确定轮廓程序名称。

铣削轮廓也可以是调用的其它程序或任意一个其它程  
 序的一个段。该部分通过起始标签和结束标签或者程  
 序段号码识别。此时, 以 “:” 标记程序名和标签/程  
 序段号码。

举例:

`_KNAME="KONTUR_1"`

铣削轮廓是完整的程序 `KONTUR_1`。

`_KNAME="ANFANG:ENDE"`

铣削轮廓作为所调用程序的一部分定义, 开  
 始程序段为标签 `ANFANG`, 结束程序段为标  
 签 `ENDE`。

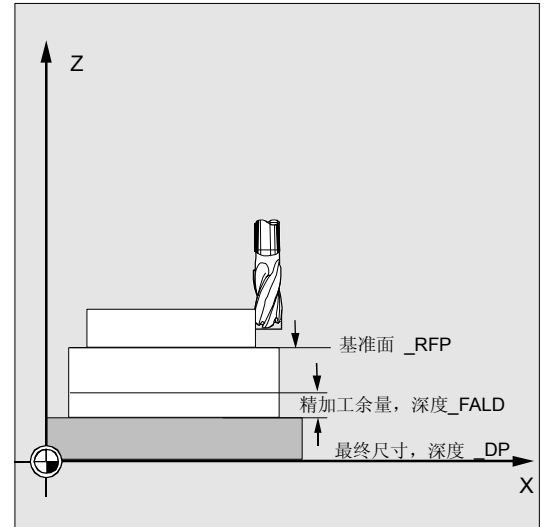
`_KNAME=  
 "/_N_SPF_DIR/_N_KONTUR_1_SPF:N130:N210"`

铣削轮廓在程序 `KONTUR_1` 的程序段 `N130`  
 至 `N210`

中定义。必须用完整的路径和扩展名写入程  
 序名称, 参见文献资料中的调用 (`CALL`) 说  
 明: `/PGA/`编程说明工作准备部分



如果这部分程序用程序段号定义, 则必须要注意: 在  
 程序修改之后, 并且后面带操作 “重新编号”, 则必  
 须使程序段号与 `_KNAME` 的这部分程序相匹配。



**\_VARI (加工方式)**

使用参数 **\_VARI** 您可以确定加工方式。可能的值，请参见“参数 **CYCLE72**”。

如果参数 **\_VARI** 编程一个其它值，则循环中断并发出报警 **61002** “加工方式错误定义”。

**\_RL (绕行轮廓)**

使用参数 **\_RL** 您可以编程轮廓的绕行，包括中心、右侧或左侧使用 **G40**、**G41** 或者 **G42**。可能的值参见“参数 **CYCLE72**”。

**\_AS1, \_AS2 (返回运行方向/轨迹, 离开运行方向/轨迹)**

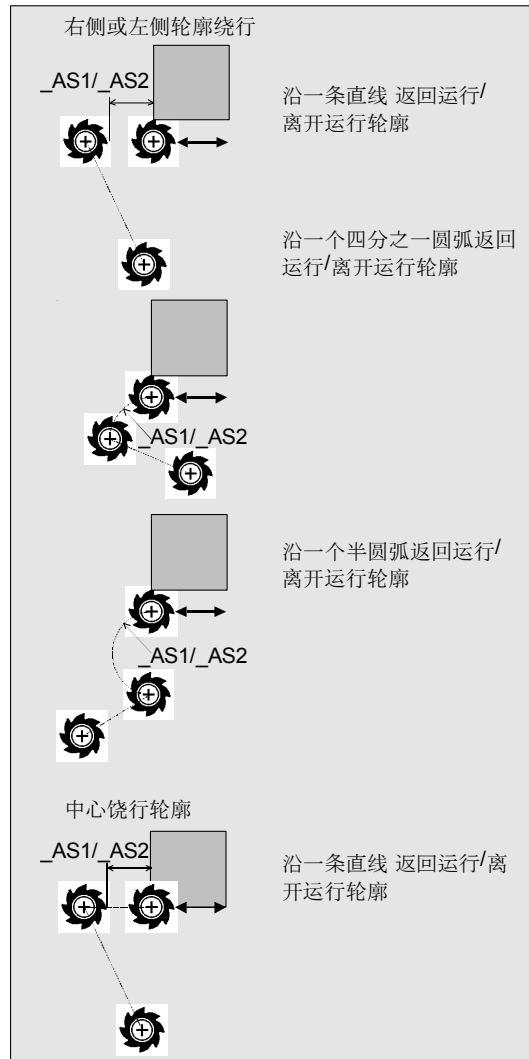
使用参数 **\_AS1** 编程返回运行位移，使用参数 **\_AS2** 编程离开运行位移。可能的值参见

“参数 **CYCLE72**”。如果未编程 **\_AS2**,

则离开运行位移的特性如同返回运行位移的特性。

只要当刀具还没有啮合，或者对此并不合适时，才可以编程空中（螺旋或者直线）柔性返回到轮廓。

如果是中心（**G40**）进行返回运行和离开运行，则仅可以编程直线。

**\_LP1, \_LP2 (长度, 半径)**

使用参数 **\_LP1** 您可以编程返回运行位移或者离开运行位移（刀具外边到轮廓起始点的距离），使用参数 **\_LP2** 编程离开运行位移或者离开运行半径（刀具外边到轮廓结束点的距离）。

值 **\_LP1**、**\_LP2** 必须大于 0。如果值为

0，则出现错误信息

**61116** “返回运行位移或者离开运行位移 =0”。

在 **G40** 时，返回运行位移或者离开运行位移等于刀具中心点到轮廓起始点或者终点的距离。



### **\_FF3 (退回进给)**

如果中间运行以进给 (G01) 进行, 则您可以用参数 **\_FF3** 定义一个退回进给, 用于平面中 (空运行) 中间定位。如果没有编程进给值, 则中间运行 (G01) 以表面进给进行。



### **其它说明**

在循环调用之前激活一个刀具补偿。否则循环会中断并给出报警

61000 “没有刀具补偿有效”。



### 编程举例1

#### 从外部铣削一个闭合的轮廓

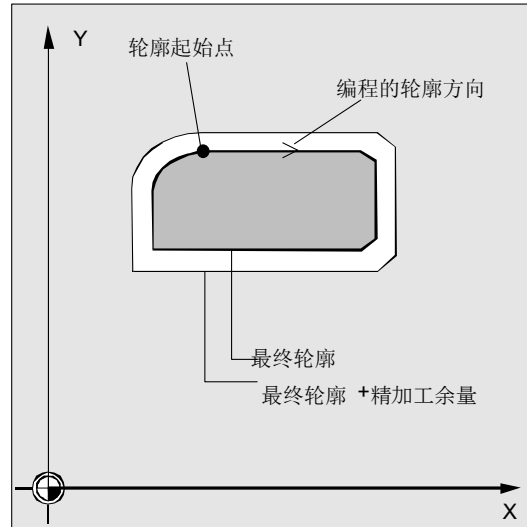
使用该程序应该铣削一个图中所描述的轮廓。

#### 循环调用参数:

- 退回平面 250 mm
- 基准面 200
- 安全距离 3 mm
- 深度 175 mm
- 最大的深度进刀 10 mm
- 深度方向的精加工余量 1.5 mm
- 深度进给 400 mm/min
- 平面中精加工余量 1 mm
- 平面中进给 800 mm/min
- 加工: 粗加工, 直至  
精加工余量, 中间位移  
用G1, 在中间位移中  
在Z方向退回至\_RFP + \_SDIS

#### 返回运行参数:

- G41- 在轮廓左侧, 也是外部加工
- 在平面中以四分之一圆弧进行返回运行和离开运行  
20 mm 半径
- 退回进给 1000 mm/min



%_N_RANDKONTUR1_MPF	; 程序, 以CYCLE72进行一个轮廓的铣削
;\$PATH=/_N_MPF_DIR	
N10 T20 D1	; T20: 铣刀带半径7
N15 M6	; 换入刀具T20
N20 S500 M3 F3000	; 进给, 编程转速
N25 G17 G0 G90 X100 Y200 Z250 G94	; 返回运行到出发位置
N30 CYCLE72 ("MYKONTUR", 250, 200, ->	; 循环调用
-> 3, 175, 10, 1, 1.5, 800, 400, 111, ->	
-> 41, 2, 20, 1000, 2, 20)	
N90 X100 Y200	
N95 M02	; 程序结束

-> 必须在一个程序段中编程

```

%_N_MYKONTUR_SPF ; 铣削轮廓子程序（举例）
; $PATH=/_N_SPF_DIR
N100 G1 G90 X150 Y160 ; 轮廓的起始点
N110 X230 CHF=10
N120 Y80 CHF=10
N130 X125
N140 Y135
N150 G2 X150 Y160 CR=25
N160 M17

```



### 编程举例2

从外部铣削一个封闭的轮廓，如同编程示例1中所述，在所调用程序中有轮廓编程

```

$TC_DP1[20,1]=120 $TC_DP6[20,1]=7
N10 T20 D1 ; T20: 铣刀带半径7
N15 M6 ; 换入刀具T20
N20 S500 M3 F3000 ; 进给，编程转速
N25 G17 G0 G90 G94 X100 Y200 Z250 -> ; 返回运行到出发位置，循环调用
CYCLE72 ("ANFANG:ENDE", 250, 200, ->
-> 3, 175, 10,1, 1.5, 800, 400, 11, ->
-> 41, 2, 20, 1000, 2, 20)
N30 G0 X100 Y200
N35 GOTOF ENDE
ANFANG:
N100 G1 G90 X150 Y160
N110 X230 CHF=10
N120 Y80 CHF=10
N130 X125
N140 Y135
N150 G2 X150 Y160 CR=25
ENDE:
N160 M02

```

### 3.13 铣削矩形轴颈- CYCLE76



#### 编程

```
CYCLE76 (_RTP, _RFP, _SDIS, _DP, _DPR, _LENG, _WID, _CRAD, _PA, _PO,
        _STA, _MID, _FAL, _FALD, _FFP1, _FFD, _CDIR, _VARI, _AP1, _AP2)
```



#### 参数

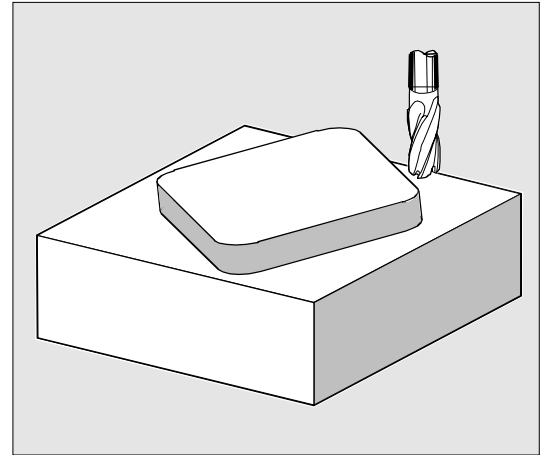
下面的输入参数始终要求：

_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到基准面，不输入符号）
_DP	实数	深度（绝对）
_DPR	实数	相对于基准面的深度（不输入符号）
_LENG	实数	轴颈长度，在标注拐角尺寸时带符号
_WID	实数	轴颈宽度，在标注拐角尺寸时带符号
_CRAD	实数	轴颈拐角半径（不输入符号）
_PA	实数	轴颈基准点，横坐标（绝对）
_PO	实数	轴颈基准点，纵坐标（绝对）
_STA	实数	纵向轴和平面中第一轴之间的角度
_MID	实数	最大进刀深度（增量，不输入符号）
_FAL	实数	边缘轮廓处精加工余量（增量）
_FALD	实数	底部精加工余量（增量，不输入符号）
_FFP1	实数	轮廓处进给
_FFD	实数	深度进刀方向的进给
_CDIR	整数	铣削方向：（不输入符号） 值： 0...同向铣削 1...逆向铣削 2 用 G2（与主轴旋转方向无关） 3...用 G3
_VARI	整数	加工方式： 值： 1...粗加工直至精加工余量 2...精加工（余量 X/Y/Z=0）
_AP1	实数	轴颈坯件长度
_AP2	实数	轴颈坯件宽度



### 功能

使用该循环您可以在加工平面中加工矩形轴颈。在精加工时要求一个端面铣刀。总是在半圆形运行前的位置上深度进刀至轮廓中。



### 工作流程

#### 循环开始之前到达的位置:

起始点在横坐标正向范围中，计入返回运行半圆并考虑所编程的横坐标一侧的毛坯尺寸。

#### 粗加工时运行过程 ( $\_VARI=1$ )

##### 轮廓返回运行和离开运行:

以快速移动返回运行到退回平面 ( $\_RTP$ )，以便随后能以该高度定位至加工平面中的起始点上。以横坐标为基准将起始点确定为 0 度。

以快速移动运行到安全距离 ( $\_SDIS$ )，接着以进给运行到加工深度。

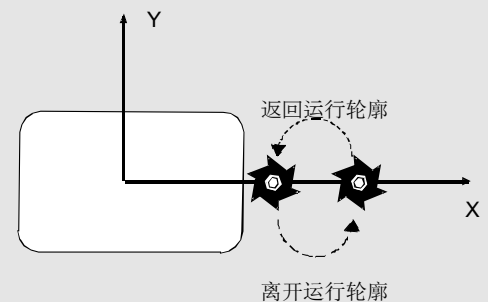
以一个半圆轨迹返回运行到轴颈轮廓。

铣削方向可以确定为同向铣削或者逆向铣削（主轴方向）。

如果轴颈绕行一次，则在平面中以半圆离开轮廓，并且进刀到下一个加工深度。

接着再次以半圆返回至轮廓，且轴颈绕行一次。该过程一直重复，直到到达编程的轴颈深度。接着以快速移动返回运行到退回平面 ( $\_RTP$ )。

主轴右旋和同向铣削时，  
以半圆返回运行/离开运行轮廓



**深度进刀:**

- 进刀到安全距离
- 插入到加工深度

第一个加工深度由以下计算:

- 总深度,
- 精加工余量和
- 最大可能的深度进刀。

**精加工时运行过程 (\_VARI=2)**

根据设置的参数 `_FAL` 和 `_FALD`,

在外表面轮廓上进行精加工, 或者在底部上进行精加工, 或者在外表面轮廓和底部上同时进行精加工。和粗加工时一样, 返回运行策略取决于平面中的运动。

**参数说明**

参数 `_RTP`、`_RFP`、`_SDIS`、`_DP`、`_DPR` 参见章节 2.1.2. (钻削, 定中心 - CYCLE81)。

参数 `_MID`、`_FAL`、`_FALD`、`_FFP1`、`_FFD` 参见章节 3.9。

循环设定参数 `_ZSD[2]` 参见章节 3.2。

**`_LENG`、`_WID` 和 `_CRAD` (轴颈长度, 轴颈宽度和拐角半径)**

使用参数 `_LENG`、`_WID` 和 `_CRAD`, 您可以确定平面中一个轴颈的形状。

此时, 可由中心或由角点出发给轴颈规定尺寸。测量拐角时用符号输入 `_LENG` 和 `_WID`。

长度的大小 (`_LENG`) 始终以横坐标为基准 (水平角度为零度)。

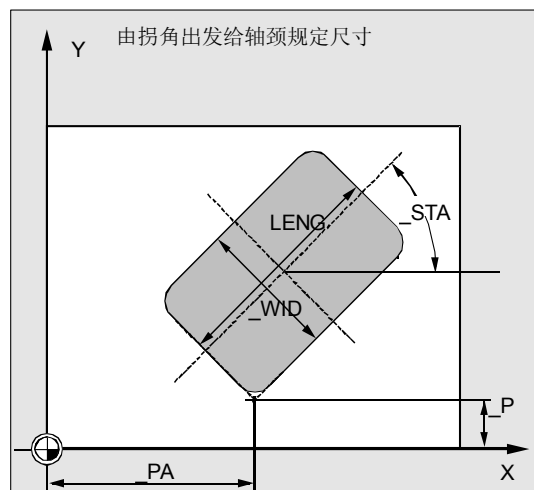
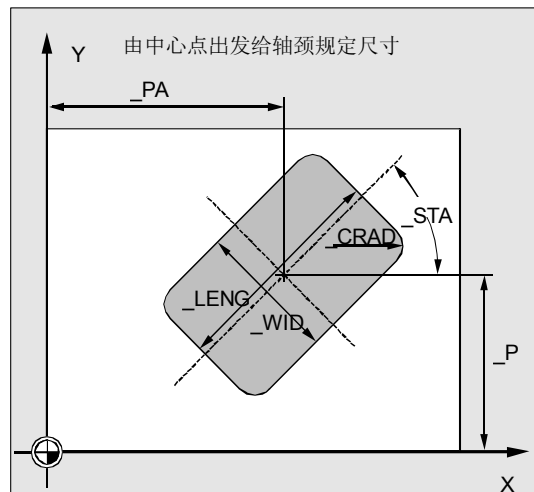
**`_PA`、`_PO` (基准点)**

使用参数 `_PA` 和 `_PO` 您可以定义轴颈在横坐标和纵坐标上的基准点。

该基准点或者是轴颈中心点, 或者一个角点。评估该参数取决于循环设定数据位 `_ZSD[2]`:

- 0 表示轴颈中心点
- 1 表示角点

如果从一个拐角进行轴颈的尺寸标注, 则长度和宽度 (`_LENG`, `_WID`) 的参数带符号输入, 由此使轴颈的位置可以明确确定。





**\_STA (角度)**

\_STA说明平面中第一轴（横坐标）和轴颈纵向轴之间的夹角。

**\_CDIR (铣削方向)**

在此参数下规定轴颈的加工方向。

通过参数\_CDIR可以

- 直接编程铣削方向“2用于G2”和“3用于G3”，或者
- 选择“同向运行”或者“逆向运行”。同向运行或者逆向运行由循环在内部通过循环调用之前激活的主轴方向计算。

同向运行	逆向运行
M3 → G3	M3 → G2
M4 → G2	M4 → G3

**\_VARI (加工方式)**

使用参数\_VARI您可以确定加工方式。

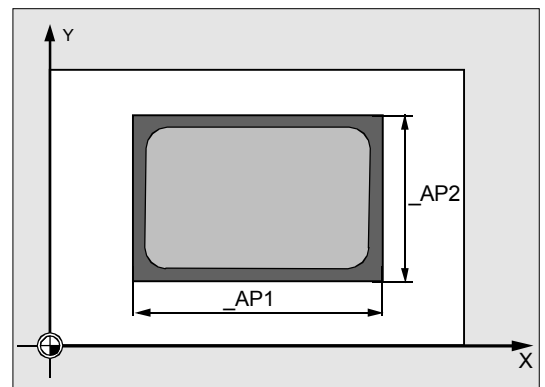
可能的值是：

- 1=粗加工
- 2=精加工

**\_AP1, \_AP2 (坯件尺寸)**

在对轴颈加工时可以考虑坯件尺寸（比如对事先浇注部件的加工）。

毛坯尺寸在长度和宽度上（\_AP1 和 \_AP2）不用符号编程并由循环以对称于轴颈中心点的方式计算设定。在内部计算的导入半圆半径取决于该尺寸。

**其它说明**

在循环调用之前激活一个刀具补偿。否则循环会被中断并发出报警 61009 “有效的刀具号码 =0”。

在循环内部使用一个新的当前工件坐标系，该坐标系影响实际值显示。该坐标系的零点就是凹槽中心点。

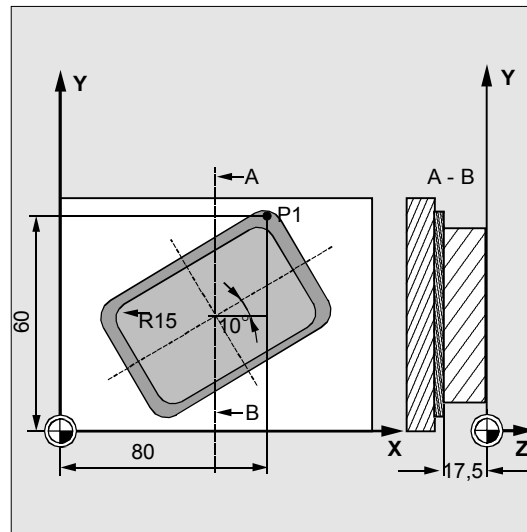
在循环结束之后，原来的坐标系再次生效。



### 编程举例

#### 轴颈

使用此程序您可以在 XY 平面中加工一个轴颈，其长度为60毫米，宽度为40毫米，拐角半径为15毫米。轴颈与X轴成10度的角，并由一个角点P1出发进行编程。规定拐角尺寸时，使用一个符号规定轴颈长度和宽度，以明确确定轴颈位置以长度80 mm、宽度50 mm 的余量预加工轴颈。



N10 G90 G0 G17 X100 Y100 T20 D1 S3000 ; 确定工艺数值

M3

N11 M6

N20 \_ZSD[2]=1 ; 通过拐角标注轴颈尺寸

N30 CYCLE76 (10, 0, 2, -17.5, , -60, -> ; 循环调用

-> -40, 15, 80, 60, 10, 11, , , 900, ->

-> 800, 0, 1, 80, 50)

N40 M30 ; 程序结束

-> 必须在一个程序段中编程

### 3.14 铣削环形轴颈 - CYCLE77



#### 编程

CYCLE77 (\_RTP, \_RFP, \_SDIS, \_DP, \_DPR, \_PRAD, \_PA, \_PO, \_MID, \_FAL, \_FALD, \_FFP1, \_FFD, \_CDIR, \_VARI, \_AP1)



#### 参数

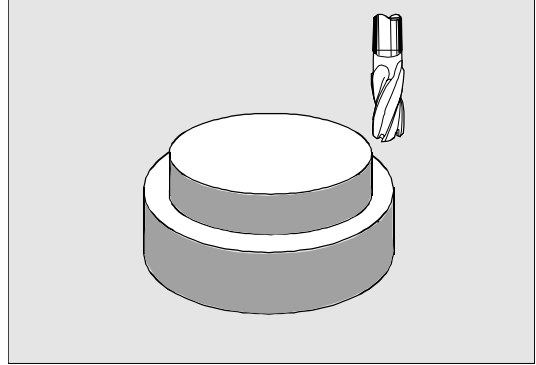
下面的输入参数始终要求：

_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到基准面，不输入符号）
_DP	实数	深度（绝对）
_DPR	实数	相对于基准面的深度（不输入符号）
_PRAD	实数	轴颈直径（不输入符号）
_PA	实数	轴颈圆心，横坐标（绝对）
_PO	实数	轴颈圆心，纵坐标（绝对）
_MID	实数	最大进刀深度（增量，不输入符号）
_FAL	实数	边缘轮廓处精加工余量（增量）
_FALD	实数	底部精加工余量（增量，不输入符号）
_FFP1	实数	轮廓处进给
_FFD	实数	深度进给（或者空间进刀）
_CDIR	整数	铣削方向：（不输入符号） 值： 0...同向铣削 1...逆向铣削 2 用 G2（与主轴旋转方向无关） 3...用 G3
_VARI	整数	加工方式 值： 1...粗加工直至精加工余量 2...精加工（余量 X/Y/Z=0）
_AP1	实数	轴颈坯件的直径



### 功能

使用该循环您可以在加工平面中加工环形轴颈。在精加工时要求一个端面铣刀。总是在半圆形运行前的位置上深度进刀至轮廓中。



### 工作流程

#### 循环开始之前到达的位置:

起始点在横坐标正向范围中，计入返回运行半圆并考虑所编程的毛坯尺寸。

#### 粗加工时运行过程 ( $\_VARI=1$ )

##### 轮廓返回运行和离开运行:

以快速移动返回到退回平面 ( $\_RTP$ )，以便随后能以该高度定位至加工平面中的起始点上。以横坐标轴为基准将起始点确定为 0 度。

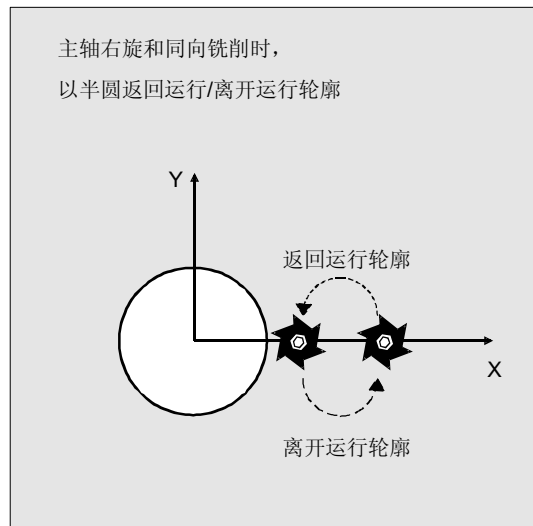
以快速移动运行到安全距离 ( $\_SDIS$ )，接着以进给运行到加工深度。在考虑编程的毛坯轴颈情况下，以一个半圆轨迹返回运行到轴颈轮廓。

铣削方向可以确定为同向铣削或者逆向铣削（主轴方向）。

如果轴颈绕行一次，则在平面中以半圆离开轮廓，并且进刀到下一个加工深度。

接着再次以半圆返回运行轮廓，且轴颈绕行一次。该过程一直重复，直到到达编程的轴颈深度。

接着以快速移动返回到退回平面 ( $\_RTP$ )。



#### 深度进刀:

- 进刀到安全距离
- 插入到加工深度

第一个加工深度由以下计算:

- 总深度,
- 精加工余量和
- 最大可能的深度进刀。

#### 精加工时运行过程 (\_VARI=2)

根据设置的参数 `_FAL` 和

`_FALD`, 在外表面轮廓上进行精加工, 或者在底部上进行精加工, 或者在外表面轮廓和底部上同时进行精加工。和粗加工时一样, 返回运行策略取决于平面中的运动。



#### 参数说明



参数 `_RTP`、`_RFP`、`_SDIS`、`_DP`、`_DPR` 参见章节 2.1.2 (钻削, 定中心 - CYCLE81)。

参数 `_MID`、`_FAL`、`_FALD`、`_FFP1`、`_FFD` 参见章节 3.9。

#### `_PRAD` (轴颈的直径)

直径不规定符号。

#### `_PA`, `_PO` (轴颈中心点)

使用参数 `_PA` 和 `_PO` 定义轴颈的基准点。

环形轴颈始终通过中心进行尺寸标注。

#### `_CDIR` (铣削方向)

在此参数下规定轴颈的加工方向。

通过参数 `_CDIR` 可以

- 直接编程铣削方向“2用于G2”和“3用于G3”, 或者
- 选择“同向运行”或者“逆向运行”。同向运行或者逆向运行由在循环内部通过循环调用之前激活的主轴旋转方向计算。

同向运行	逆向运行
M3 → G3	M3 → G2
M4 → G2	M4 → G3

#### `_VARI` (加工方式)

使用参数 `_VARI` 您可以确定加工方式。

可能的值是:

- 1=粗加工
- 2=精加工

**\_AP1 (轴颈坯件的直径)**

使用该参数定义轴颈毛坯尺寸（不带符号）。在内部计算的导入半圆半径取决于该尺寸。

**其它说明**

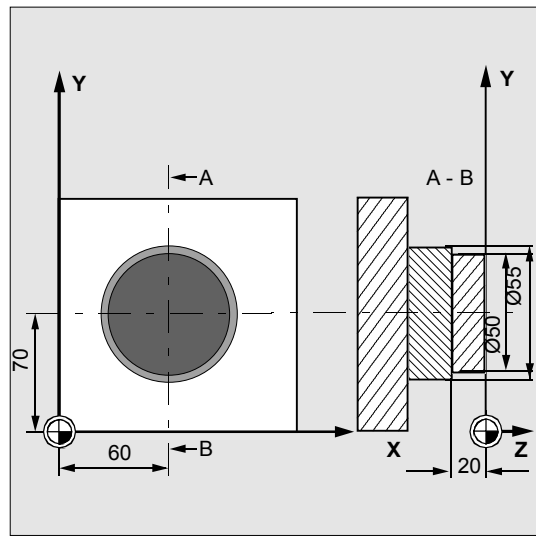
在循环调用之前激活一个刀具补偿。否则循环会被中断并发出报警 61009 “有效的刀具号码 =0”。

在循环内部使用一个新的当前工件坐标系，该坐标系影响实际值显示。该坐标系的零点就是凹槽中心点。

在循环结束之后，原来的坐标系再次生效。

**编程举例****环形轴颈**

以直径 55 mm 和每次切削的最大进刀量 10 mm 用毛坯进行轴颈加工。规定精加工余量，用于随后的轴颈外表面精加工。在逆向运行中进行整个加工。



N10 G90 G17 G0 S1800 M3 D1 T1 ; 确定工艺数值

N11 M6

N20 CYCLE77 (10, 0, 3, -20, , 50, 60, -> ; 循环调用粗加工  
-> 70, 10, 0.5, 0, 900, 800, 1, 1, 55)

N30 D1 T2 M6 ; 换刀

N40 S2400 M3 ; 确定工艺数值

N50 CYCLE77 (10, 0, 3, -20, , 50, 60, -> ; 循环调用精加工  
-> 70, 10, 0, 0, 800, 800, 1, 2, 55)

N40 M30 ; 程序结束

-> 必须在一个程序段中编程

### 3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75



带中心岛的凹槽铣削是一个选项，在NCK和HMI高级配置中要求软件版本SW6。

#### 前提条件

为了应用带中心岛的凹槽铣削循环，要求进行某些机床数据设定。



**文献：**有关当前信息，参见：

- 供货软件（标准循环）的文件“siemensd.txt”，或者
- 在 HMI Advanced（高级）配置中 F:\dh\cst.dir\HLP.dir\siemensd.txt



#### 功能

利用循环CYCLE73, CYCLE74 和 CYCLE75可以加工带中心岛的凹槽。

凹槽和中心岛的轮廓在同一个程序中以DIN代码定义，如同凹槽加工或者子程序。

通过循环CYCLE74 和

CYCLE75，凹槽边缘轮廓或者中心岛轮廓传给 CYCLE73原来的凹槽铣削循环。

CYCLE73借助几何处理器生成一个加工程序并对其进行处理。必须遵守循环调用的可编程的顺序，以保证能进行正确的编程处理。

- CYCLE74( ) ; 传送边缘轮廓
- CYCLE75( ) ; 传送中心岛轮廓1
- CYCLE75( ) ; 传送中心岛轮廓2
- ...
- CYCLE73( ) ; 加工凹槽

### 3.15.1 传送凹槽边缘轮廓- CYCLE74



带中心岛的凹槽铣削是一个选项，在NCK和HMI高级配置中要求软件版本SW6。



#### 编程

CYCLE74 (\_KNAME, \_LSANF, \_LSEND)



#### 参数

_KNAME	字符串	凹槽边缘轮廓的轮廓子程序名称
_LSANF	字符串	轮廓定义开始的程序段号/标签
_LSEND	字符串	轮廓定义结束的程序段号/标签



#### 功能

循环 CYCLE74 用于传送凹槽边缘轮廓至凹槽铣削循环 CYCLE73。为此，在标准循环目录中存放临时文件并在该目录中存储传送的参数值。

如果已经存在这样的文件，则必须清除并重新编制。

因此，始终要求通过调用CYCLE74开始一个程序，用于铣削带中心岛的凹槽。





### 参数说明

可在一个独立的程序中编程边缘轮廓，也可以在调用的主程序中编程边缘轮廓。通过参数\_KNAME、程序名称或\_LSANF、\_LSEND程序段标记从 ... 至通过程序段号码或者标签传输到循环上。

由此，共有三种轮廓编程方法：

- **轮廓位于自身的程序中，**  
这样，仅必须编程\_KNAME；  
比如 CYCLE74 ( “RAND” , “”, “” )
- **轮廓位于所调用的程序中，**  
这样，仅必须编程\_LSANF 和\_LSEND；  
比如 CYCLE74 ( “”, “N10”, “N160 “ )
- **边缘轮廓仅是一个程序的一部分，但不是调用该循环的程序，**  
这样必须编程所有3个参数，比如  
CYCLE74 ( “RAND “ ,  
“MARKE\_ANFANG” , “MARKE\_ENDE” )

该程序名可以通过路径名和程序类型编写。

举例：

```
_KNAME="/N_WKS_DIR/_N_BEISPIEL3_WPD/_N_RAND_MPF"
```

### 3.15.2 传送中心岛轮廓- CYCLE75



带中心岛的凹槽铣削是一个选项，在NCK和HMI高级配置中要求软件版本SW6。



#### 编程

CYCLE75 (\_KNAME, \_LSANF, \_LSEND)



#### 参数

_KNAME	字符串	中心岛轮廓的轮廓子程序名称
_LSANF	字符串	轮廓定义开始的程序段号/标签
_LSEND	字符串	轮廓定义结束的程序段号/标签



#### 功能

循环 CYCLE75 用于传送中心岛轮廓至凹槽铣削循环 CYCLE73。每个中心岛轮廓均由循环调用一次。如果没有中心岛，则不必调用。

所传送的参数值写入到由CYCLE74所建立的临时文件中。



#### 参数说明

参数的数量和含义符合CYCLE74。

(参见 CYCLE74)

### 3.15.3 轮廓编程



凹槽边缘轮廓和中心岛轮廓必须始终封闭，也就是说起始点和终点相同。

起始点，也就是说每个轮廓的第一个点始终以G0编程，所有其它的轮廓单元通过G1到G3进行编程。

在编程轮廓时在最后的轮廓单元中（带标签的程序段或者轮廓结束处的程序段号）不允许包含半径或者斜边。

刀具在调用CYCLE73之前不允许位于所编程轮廓单元的起始位置。

可总是将所要求的程序仅存放在一个目录中（工件程序或零件程序）。

允许将全局子程序存储器用于凹槽边缘轮廓或中心岛轮廓。

可选择性地以公制尺寸或英制尺寸编程与工件相关的几何尺寸数据。在各自的轮廓程序内部切换尺寸数据会导致生成有错误的加工程序。

在轮廓程序中使用 G90/G91

交替工作时必须要注意：在以待加工的轮廓程序为顺序的程序起始处，已经编程了正确的指令用于尺寸说明。

在计算凹槽的加工程序时仅考虑平面中的几何量。

如果在轮廓段编程了其它轴或功能（T..、D..、S..M..等等），则在循环内部进行轮廓预处理时跳过这些轴或功能。

在循环开始之前，必须编程所有编程所需的机床专用的指令（如刀具调用、转速、M 指令）。进给必须作为参数在 CYCLE73 中设置。

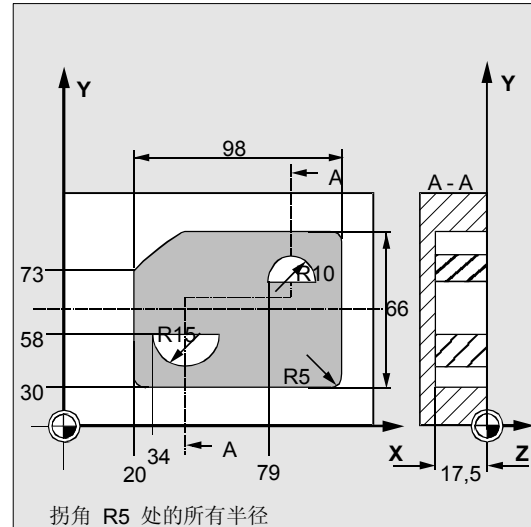
刀具半径必须大于零。

不可使用通过相应的控制指令（如零点偏移、框架，等等）通过偏移进行的中心岛轮廓重复。必须重新编程每个要重复的中心岛，此时，在坐标中计算偏移值。



## 编程举例

举例程序1.mpf（带中心岛的凹槽）



```
%_N_BEISPIEL1_MPF
```

```
;$PATH=/_N_MPF_DIR
```

```
;示例 1:带中心岛的凹槽
```

```
;
```

```
$TC_DP1[5,1]=120 $TC_DP6[5,1]=6 $TC_DP3[5,1]=111 ; 铣刀T5D1的刀具补偿
```

```
$TC_DP1[2,2]=120 $TC_DP6[2,2]=5 $TC_DP3[2,2]=130
```

```
N100 G17 G40 G90
```

```
; 输出端条件G代码
```

```
N110 T5 D1
```

```
; 更换铣刀
```

```
N120 M6
```

```
N130 S500 M3 F2000 M8
```

```
GOTOF _BEARBEITUNG
```

```
;
```

```
N510 _RAND:G0 G64 X25 Y30 F2000
```

```
; 定义边缘轮廓
```

```
N520 G1 X118 RND=5
```

```
N530 Y96 RND=5
```

```
N540 X40 RND=5
```

```
N545 X20 Y75 RND=5
```

```
N550 Y35
```

```
N560 _ENDRAND:G3 X25 Y30 CR=5
```

```
;
```

```
N570 _INSEL1:G0 X34 Y58
```

```
; 定义下面的中心岛
```

```
N580 G1 X64
```

```
N590 _ENDINSEL1:G2 X34 Y58 CR=15
```

```
;
```

```
N600 _INSEL2:G0 X79 Y73
```

```
; 定义上面的中心岛
```

```
N610 G1 X99
```

```
N620 _ENDINSEL2:G3 X79 Y73 CR=10
```

```
;
```

```
_BEARBEITUNG:
```

```
; 编程轮廓
```

```
BEISPIEL_CONT:
```

```
CYCLE74 ("BEISPIEL1", "_RAND", "_ENDRAND")
```

```
; 传送边缘轮廓
```

```
CYCLE75 ("BEISPIEL1", "_INSEL1", "_ENDINSEL1")
```

```
; 传送中心岛轮廓1
```

```
CYCLE75 ("BEISPIEL1", "_INSEL2", "_ENDINSEL2")
```

```
; 传送中心岛轮廓2
```

```
ENDLABEL:
```

```
M30
```

### 3.15.4 带中心岛的凹槽铣削 - CYCLE73



带中心岛的凹槽铣削是一个选项，在NCK和HMI高级配置中要求软件版本SW6。



#### 编程

CYCLE73 (\_VARI, \_BNAME, \_PNAME, \_TN, \_RTP, \_RFP, \_SDIS, \_DP, \_DPR, \_MID, \_MIDA, \_FAL, \_FALD, \_FFP1, \_FFD, \_CDIR, \_PA, \_PO, \_RAD, \_DP1)



#### 参数

_VARI	整数	加工方式：（不输入符号） 个位（加工选择）： 值： 1...由完整材料进行粗加工（扩孔） 2...由剩余材料进行粗加工 3...边缘进行精加工 4...底部进行精加工 5...预制孔 十位（选择插入方案）： 值： 1...以 G1 垂直 2...以螺旋轨迹 3...摆动 百位（选择退刀方式）： 值： 0...到退回平面（_RTP） 1...退回基准面（_RFP）之上的安全距离（_SDIS） 千位（选择起始点）： 值： 1...自动 2...手动
_BNAME	字符串	钻孔位置程序名称
_PNAME	字符串	凹槽铣削加工程序的名称
_TN	字符串	扩孔刀具名称
_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到基准面，不输入符号）
_DP	实数	凹槽深度（绝对）
_DPR	实数	凹槽深度（增量）
_MID	实数	一个横向进给的最大进刀深度（不输入符号）
_MIDA	实数	平面中最大进刀宽度（不输入符号）
_FAL	实数	平面中精加工余量（不输入符号）
_FALD	实数	底部的精加工余量（不输入符号）
_FFP1	实数	用于表面加工的进给
_FFD	实数	用于深度进刀的进给

<code>_CDIR</code>	整数	用于凹槽加工的铣削方向：（不输入符号） 值： 0...同向铣削（与主轴旋转方向一致） 1...逆向铣削 2...用 G2（与主轴旋转方向无关） 3...用 G3
<code>_PA</code>	实数	在第一轴上的起始点（仅在手动选择起始点时）
<code>_PO</code>	实数	在第二轴上的起始点（仅在手动选择起始点时）
<code>_RAD</code>	实数	在插入螺旋轨迹时圆心轨迹半径，或者摆动插入时的再入角。
<code>_DP1</code>	实数	每360°-旋转时插入深度，在螺旋轨迹上插入



## 功能

### 循环 CYCLE73

是一种加工循环，使用该循环可加工带或不带中心岛的凹槽。该循环支持此类凹槽的完整加工并提供下列加工步骤：

- 预制钻孔
- 扩孔凹槽
- 加工剩余材料
- 边缘精加工
- 底部精加工

以自由的DIN编程编制凹槽轮廓和中心岛轮廓，比如利用几何处理器。

循环根据用于每个加工步骤而编程的加工方式

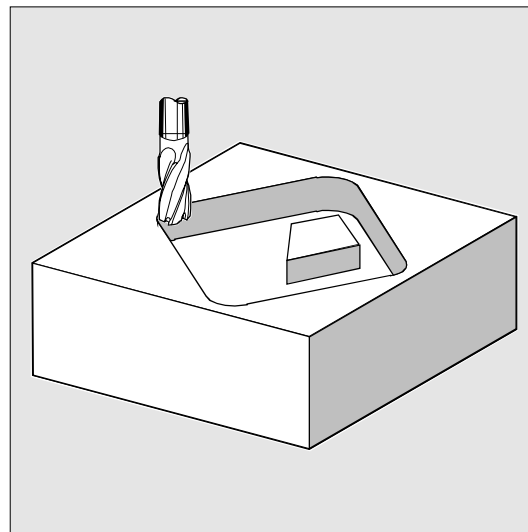
（`_VARI`）运行一次。这种情况下，需要进行粗加工和精加工，或要求一个附加的剩余材料粗加工，还必须重新调用CYCLE73。

### 扩孔凹槽

扩孔时，加工带有效刀具的凹槽，直至编程的精加工余量。可选择插入方案用于铣削。根据所规定的值以凹槽深度的方向进行切削分配（刀具轴）。

### 加工剩余材料

循环可用较小的铣刀扩孔材料。在生成的程序中输出由最后铣削过程中的剩余材料和当前刀具半径得出的



运行运动。可多次连续地以不断变小的刀具半径使用  
剩余材料工艺。

对加工之后凹槽中是否有剩余材料并不进行监控。

#### **边缘/底部精加工**

该循环一个接下去的功能就是对凹槽底部进行精  
加工，或者分为几个精加工步骤对凹槽和单个中心岛  
进行绕行。

#### **预制钻孔**

铣削时，根据所使用的刀具，可能要求在扩孔前进行  
钻削。循环根据随后的扩孔步骤自动计算预制钻孔位  
置。在每个这样的位置上执行事先模态调用的钻削循  
环。可用多个工艺加工步骤进行预制钻孔（如1定中  
心，2 钻削）。



### 预制钻孔步骤

在第一个预制钻孔的加工步骤中，在通过一个REPEAT指令模态调用钻削循环后，调用带CYCLE73

内容以及轮廓重复的加工步骤顺序。在下一个换刀前模态选择钻削循环。接着可进行其它的钻削工艺。

接着进行一个带CYCLE73的程序部分，其中包含所有要求的参数以及扩孔和钻削的程序。

除参数\_VARI之外，扩孔工艺的所有参数都是相关的，并且必须始终写入。

在这个时刻循环仅产生凹槽扩孔程序和钻削位置，并且调用钻削位置程序并接着进行加工。

多个不同的凹槽存在的条件是，在该段中再次调用所属的轮廓。在仅有一个凹槽时可取消该程序段。

整个加工部分是通过一个跳转指令转换到下一个凹槽部分进行扩孔。

### 举例

预制钻孔，带扩孔

ABNAHME4_CONT:	; 凹槽轮廓开始的标记，带名称
CYCLE74 ("RANDA01", ,)	; 定义凹槽边缘轮廓
CYCLE75 ("INS11A01", ,)	; 定义第一个中心岛轮廓
CYCLE75 ("INS1A01", ,)	
CYCLE75 ("INS2A01", ,)	
CYCLE75 ("INS3A01", ,)	
ENDLABEL:	; 凹槽轮廓结束标记
T4 M6	
D1 M3 F1000 S4000	
MCALL CYCLE81(10,0,1,-3)	; 模态调用钻削循环
REPEAT ABNAHME4_BEAR ABNAHME4_BEAR_END	; 钻削位置程序加工
MCALL	; 模态撤消钻削循环选择
GOTOF ABNAHME4_BEAR_END	; 跳转到凹槽扩孔
ABNAHME4_BEAR:	; 产生程序部分开始
;REPEAT ABNAHME4_CONT ENDLABEL	; 仅在超过一个凹槽轮廓时要求



```

CYCLE73(1015,"ABNAHME4_DRILL","ABNAHME4_MIL
L1","3",10,0,1,-
12,0,,2,0.5,,9000,400,0,,,,)
ABNAHME4_BEAR_END: ; 产生程序部分结束
T3 M6
D1 M3 S2000
;REPEAT ABNAHME4_CONT ENDLABEL ; 仅在超过一个凹槽轮廓时要求
CYCLE73(1011,"ABNAHME4_DRILL","ABNAHME4_MIL ; 扩孔凹槽
L1","3",10,0,1,-
12,0,,2,0.5,,9000,400,0,,,,)

```



### 粗加工步骤，扩孔（\_VARI=XXX1）

CYCLE73指令必须连同所有参数再次写入。

该程序执行以下的加工步骤：

- 返回运行手动计算得出或自动生成的位于退回平面高度的起始点。接着，用 G0 运行至提前了安全距离的基准面。
- 根据所选择的插入方案（\_VARI），以进给值 \_FFD 进刀到当前的加工深度。
- 以进给值 \_FFP1 扩孔带有中心岛的凹槽，直至精加工余量。- 加工方向按照 \_CDIR 下确定的方向。如果中心岛之间以及中心岛与边缘轮廓之间待加工的空间与铣刀直径相互抵触，则可以划分该凹槽。为了这个目的，该循环计算附加的起始点，用于铣刀插入。
- 根据所选择的退刀模式退刀，并返回到起始点用于下一次水平进刀。
- 在结束凹槽加工后，根据所选的退刀模式，刀具或者返回至退回平面，或者通过基准面以安全距离返

回。由生成的程序得出平面中的刀具位置，位于凹槽表面的上半部分。



#### 精加工步骤（\_VARI=XXX3）

- 在边缘部分进行精加工时，凹槽轮廓以及中心岛轮廓分别绕行一次。作为插入方案，必须用 G1 垂直编程（\_VARI）。沿着一个切向圆弧段返回运行和离开运行精加工起始点或者终点。
- 为了进行底部精加工，以 G0 运行直至凹槽深度 + 精加工余量 + 安全距离。自那里起，总是以垂直于深度的方向用深度进刀的进给运行。凹槽的底部面积一次加工。
- 退刀和退回如同扩孔时一样进行。
- 为了在平面中和底部能够同时进行精加工，必须占用参数 \_FAL 和 \_FALD 以及 \_VARI=XXX4。



## 参数说明

### **\_VARI (加工方式)**

使用参数\_VARI可以确定加工方式。可能的值是：

**个位：**

- 1=由完整材料进行粗加工（扩孔）
- 2=由剩余材料进行粗加工
- 3=边缘进行精加工
- 4=底部进行精加工
- 5=预制钻孔

使用“由完整材料进行粗加工”设定，加工程序则对凹槽进行扩孔直至精加工余量。

如果要求选择的铣刀直径不要加工待扩孔棱边的表面，则可以使用设定“2”和一个较小的铣刀，对该表面进行后清理。对此必须重新调用循环CYCLE73。

**十位：**

- 1=以 G1 垂直
- 2=螺旋轨迹
- 3=摆动

**选择插入方法：**

- **垂直插入（\_VARI=XX1X）**

表示在一个程序段中执行循环内部所计算的实际进刀深度。

- **插入至螺旋线轨迹（VARI= XX2X）**

表示，铣刀中心点运行至通过半径 \_RAD 和深度每转\_DP1 确定的螺旋线轨迹。同样在 \_FFD 下编程进给。螺旋线轨迹的转向与用以加工凹槽的转向相符。

插入时在\_DP1下编程的深度作为最大深度计算，并且始终是整数个螺旋轨道的转数。

在到达进刀实际深度时（可能为螺旋轨道上多次转数），还要执行一个整圆，从而消除插入时的倾斜轨迹。

之后在平面中扩孔凹槽，直至精加工余量。

- **摆动插入 (VARI= XX3X)**

表示，铣刀中心点沿一条直线前后摆动斜向插入，直至其到达下一个当前深度。在 `_RAD` 下编程最大插入角度，在循环内部计算摆动位移长度。如果到达当前深度，则还要执行一次不带深度进刀的位移，从而消除插入时的倾斜轨迹。在 `_FFD` 下编程进给。

**百位: (`_VARI=X1XX`)**

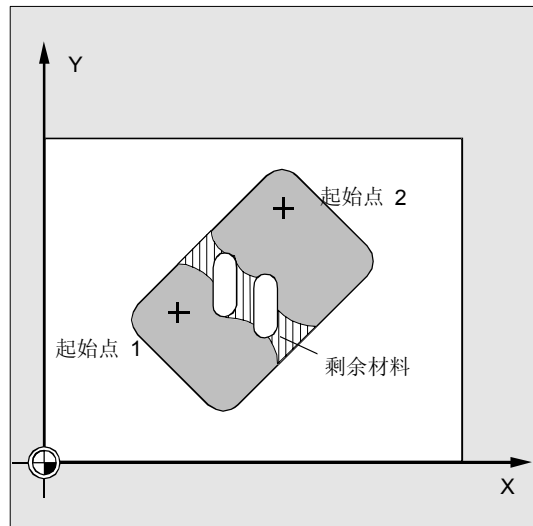
- 0=到退回平面 (`_RTP`)
- 1=基准面 (`_RFP`) 上方安全距离 (`_SDIS`)

**千位: (`_VARI=1XXX`)**

- 1=自动起始点
- 2=手动起始点

在自动选择起始点时，循环计算该起始点，用于其自身内部加工。

**注意：**在中心岛表面的区域中不允许存在有手动规定的起始位置。这种情况下，在内部不进行监控。如果通过该中心岛的位置和所使用铣刀直径的情况，对该凹槽进行了划分，则自动计算多个起始点。手动进行确定时，必须另外编程参数 `_PA` 和 `_PO`。因此，可仅编程起始点。在划分凹槽时自动计算所要求的起始点。



**\_BNAME** (钻削位置程序名)**\_PNAME** (凹槽加工程序名)

凹槽铣削循环生成带有预制钻孔或铣削所需的运行程序段的程序。如果由那里开始调用循环或存储在相应的工件目录中，则这些程序存储在该目录下的零件程序存储器中（该目录也存储有调用程序），也就是目录“零件程序”（MPF.DIR）中。这些程序总是为主程序（类型MPF）。

参数 **\_BNAME** 和 **\_PNAME** 定义该程序的名称。

仅在 **\_VARI=XXX5** 时要求钻削程序名。

示例： 没有钻削程序名：

**CYCLE73** (1011, “”、**ABNMAHME4\_MILL**, ...)

**\_TN** (扩孔刀具名)

用扩孔刀具占用该参数。这里，视带还是不带有有效的刀具管理而定，输入刀具名称或刀具号码。

示例：

- 带刀具管理

**CYCLE73** (1015, “**TEIL1\_DRILL**”、 “**TEIL1\_MILL**”、  
“**FRAESER3**”, ...)

- 不带刀具管理

**CYCLE73** (1015, “**TEIL1\_DRILL**”、 “**TEIL1\_MILL**”、  
“**3**”, ...)

参数 **\_TN** 作为必需参数确定，最多16个字符，因此在每次调用 **CYCLE73** 时用扩孔刀具占用。在多次使用剩余材料加工时，必须使用最后剩余材料加工的刀具。

刀具和补偿：

必须注意的是，仅由 **D1**

处理刀具补偿。不允许使用备用刀具策略。

### **\_RFP 和 \_RTP (基准面和退回平面)**

通常，基准面（RFP）和退回平面（RTP）有不同的值。在循环中，这主要是指，退回平面位于基准面之前。退回平面到钻孔底部的距离也大于基准面到钻孔底部的距离。

### **\_SDIS (安全距离)**

安全距离（SDIS）取决于基准面而生效。继续提前安全距离。

安全距离生效的方向由循环自动确定。

### **\_DP (绝对凹槽深度) 和**

### **\_DPR (增量凹槽深度)**

根据基准面，可将凹槽深度规定为绝对值（\_DP），也可以将凹槽深度规定为增量值（\_DPR）。在增量尺寸时，循环利用基准面和退回平面的位置自行计算所产生的深度。

### **\_MID (最大进刀深度)**

通过这些参数，您可以确定最大的进刀深度。在循环中深度以均匀的进刀尺寸进行。

利用\_MID和总深度，该循环自己计算进刀量。

存放进刀尺寸最小可能的数量。\_MID=0

表示一步进刀到凹槽深度。

### **\_MIDA (平面中最大的进刀宽度)**

使用该参数您可以确定平面中扩孔时最大的进刀宽度。决不超出该进刀深度。

如果该参数没有编程，或者为值0，则该循环在内部以 80% 的铣刀直径作为最大的进刀宽度。

如果编程的进刀宽度超出80%的铣刀直径，则发出报警61982“平面中进刀宽度太大”，循环停止。

**\_FAL (平面中精加工余量)**

该精加工余量仅在平面中加工凹槽边缘时生效。

如果精加工余量 $\geq$

刀具直径，则不保证凹槽进行完整的扩孔。

**\_FALD (底部的精加工余量)**

在粗加工时，在底部考虑一个分开的精加工余量。

**\_FFD 和 \_FFP1 (深度进刀和表面加工用进给)**

进给\_FFD在插入到原料时生效。

进给\_FFP1在平面中以进给运行的所有运动中生效。

**\_CDIR (铣削方向)**

在此参数下规定凹槽的加工方向。

通过参数\_CDIR可以

- 直接编程铣削方向“2用于G2”和“3用于G3”，或者
- 选择“同向运行”或者“逆向运行”。

同向运行或者逆向运行由循环在内部通过循环调用之前激活的主轴方向计算。

同向运行	逆向运行
M3 → G3	M3 → G2
M4 → G2	M4 → G3

**\_PA, \_PO (第一轴和第二轴的起始点)**

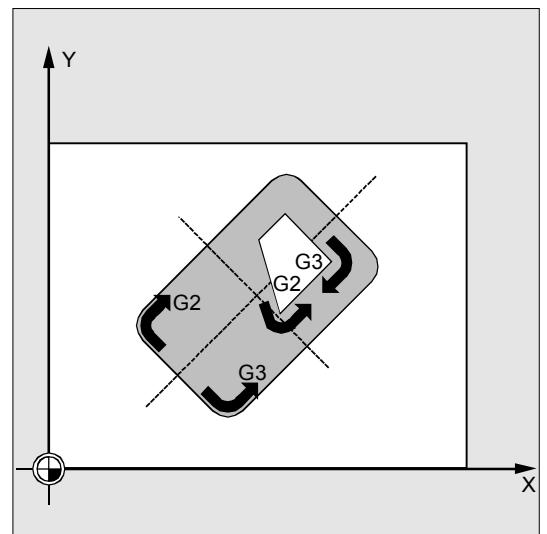
手动选择起始点时，在这些参数下编程起始点时，要使得能够无碰撞的返回运行。必须要注意的是，仅可编程起始点（参见参数说明 \_VARI）。

**\_RAD (圆心轨迹或者再入角)**

使用参数\_RAD定义螺旋轨迹的半径（与刀具中心点轨迹相关）或者用于摆动加工的最大再入角。

**\_DP1 (插入深度用于螺旋轨迹)**

使用参数\_DP1定义螺旋轨迹上插入时的进刀深度。



## 其它说明

### 凹槽加工的名称 (NAME)

通常，以多个工艺加工步骤进行凹槽加工。仅可一次定义描述凹槽几何形状的轮廓。

为了在程序中可以自动分配轮廓到各个加工步骤，给轮廓说明用标签标识，这些程序部分后面通过REPEAT指令进行重复。

因此，借助循环支持编程程序时，在各自的屏幕窗口中分别输入用于凹槽加工的名称。名称长度限定为最多 8 个字符。

在程序示例 2 中，比如为“ABNAHME4”。

### T

号码包含用于所有加工工艺的扩孔铣刀。多次加工剩余材料时，总是为此在使用时事先写入已知的刀具。

## 循环结构的说明

### 循环 CYCLE73

用于解决扩孔带中心岛的凹槽时所出现的非常复杂的问题，这对系统的计算能力提出很高的要求。为了使处理时间得到优化，这在 HMI 中进行计算。

就循环而言启动该计算，在其结果中生成程序，带钻削或铣削的运行程序段，程序被存储在系统的文件系统中。该程序由循环调用并执行。

该结构使得，仅在第一次用 CYCLE73

调用进行程序时才必须进行计算。从第二次程序运行起，已存在运行程序，并同样可以由循环调用。

在以下情况下重新计算：

- 所参与的轮廓发生改变；
- 该循环的传送参数发生改变；
- 在循环调用之前用其它的刀具补偿参数激活一个刀具；
- 在不同的工艺中，比如扩孔以及有不同的生成加工程序加工剩余材料时



### 文件系统中程序存储

如果在所调用的主程序之外编程CYCLE73轮廓，则以下说明适用于查找控制系统中文件系统的查找：

- 如果所调用的程序在一个工件目录中，则编程了边缘或中心岛轮廓的程序也必须在同一个工件目录下；
- 如果所调用的程序在目录“零件程序”（MPF.DIR）下，则同样在那儿查找程序。

由循环所产生的程序同样存放在调用该循环的程序目录下，也就是说在同样的工件目录或者MPF.DIR或SPF.DIR中。

### 模拟说明

在模拟凹槽铣削时生成的程序存储在NCU的文件系统中。由于刀具补偿数据引入至程序计算中，因此仅带有“NC激活数据”的设置有意义。

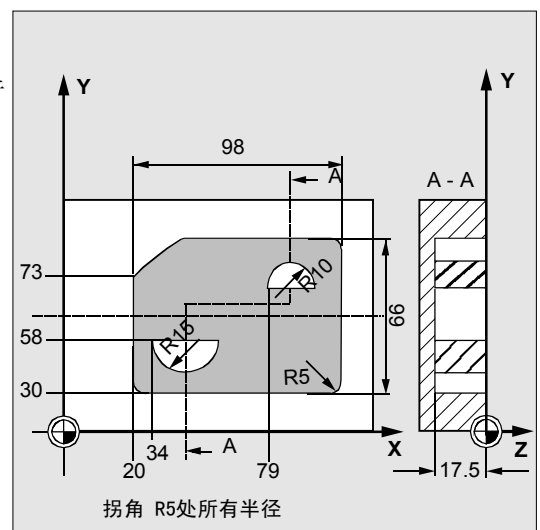


### 编程举例1

加工任务是：一个凹槽带2个中心岛，由完整的材料进行加工，接着在XY平面中进行精加工  
举例程序1.mpf（带中心岛的凹槽）

```
%_N_BEISPIEL1_MPF
;$PATH=/_N_WKS_DIR/_N_CC73BEI1_WPD
;示例_1:带中心岛的凹槽
; 扩孔和精加工
```

```
$TC_DP1[5,1]=120 $TC_DP3[5,1]=111 ; 铣刀T5D1的刀具补偿
$TC_DP6[5,1]=4
$TC_DP1[2,1]=120 $TC_DP3[2,1]=130
$TC_DP6[2,1]=5
N100 G17 G40 G90 ; 输出端条件G代码
N110 T5 D1 ; 更换铣刀
N120 M6
```



## 3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75

```

N130 M3 F2000 S500 M8
N140 GOTOF _加工
;
N510 _RAND:G0 G64 X25 Y30 ; 定义边缘轮廓
N520 G1 X118 RND=5
N530 Y96 RND=5
N540 X40 RND=5
N545 X20 Y75 RND=5
N550 Y35
N560 _ENDRAND:G3 X25 Y30 CR=5
;
N570 _INSEL1:G0 X34 Y58 ; 定义下面的中心岛
N580 G1 X64
N590 _ENDINSEL1:G2 X34 Y58 CR=15
;
N600 _INSEL2:G0 X79 Y73 ; 定义上面的中心岛
N610 G1 X99
N620 _ENDINSEL2:G3 X79 Y73 CR=10
;

; 编程轮廓
_BEARBEITUNG:
BEISPIEL1_CONT:
CYCLE74 ("", "_RAND", "_ENDRAND")
CYCLE75 ("", "_INSEL1", "_ENDINSEL1")
CYCLE75 ("", "_INSEL2", "_ENDINSEL2")
ENDLABEL:

; 凹槽铣削编程
CYCLE73 (1021, "", "BEISPIEL1_MILL1", "5", 10, 0, 1,
-17.5, 0, , 2, 0.5, , 9000, 3000, 0, , , 4, 3)

T2 D1 M6
S3000 M3

; 凹槽精加工编程
CYCLE73 (1113, "", "BEISPIEL1_MILL3", "5", 10, 0, 1,
-17.5, 0, , 2, , , 8000, 1000, 0, , , 4, 2)
M30

```



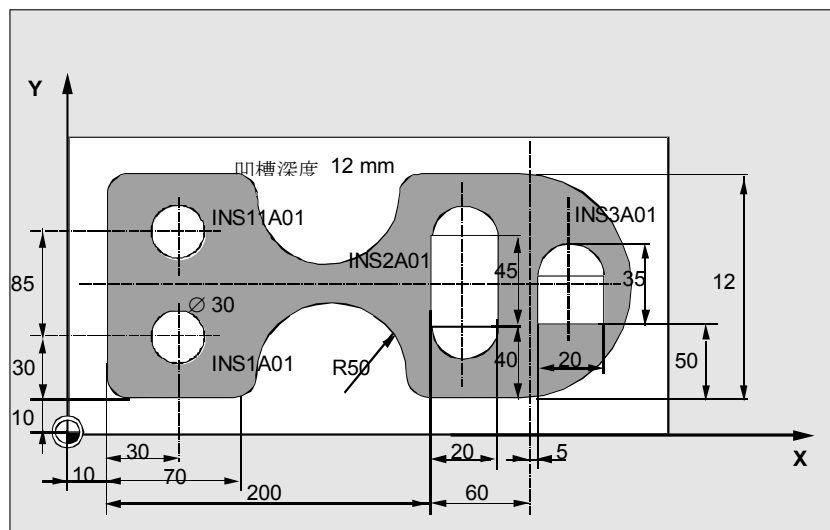
## 编程举例2

加工任务:

在凹槽铣削之前必须进行预制钻孔, 从而保证铣刀可以优化插入。

- 定中心, 用于预制钻孔
- 钻削
- 扩孔带中心岛的凹槽, 铣刀半径12毫米
- 扩孔剩余材料, 铣刀半径6毫米
- 凹槽精加工, 铣刀半径5毫米

加工示意图



加工程序:

```

%_N_BEISPIEL2_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD
; 示例 2: 带中心岛的凹槽
; 2* 预制钻孔, 扩孔, 剩余材料扩孔, 精加工
;
; 刀具补偿数据
$TC_DP1[2,1]=220 $TC_DP6[2,1]=10
$TC_DP1[3,1]=120 $TC_DP6[3,1]=12
$TC_DP1[4,1]=220 $TC_DP6[4,1]=3
$TC_DP1[5,1]=120 $TC_DP6[5,1]=5
$TC_DP1[6,1]=120 $TC_DP6[6,1]=6
TRANS X10 Y10

; 定义加工轮廓
ABNAHME4_CONT:
CYCLE74 ("RANDA01",,,)
CYCLE75 ("INS1A01",,,)
CYCLE75 ("INS1A01",,,)
CYCLE75 ("INS2A01",,,)
CYCLE75 ("INS3A01",,,)

ENDLABEL:

```

## 3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75

---

; 定中心编程

T4 M6

D1 M3 F1000 S4000

MCALL CYCLE81 (10,0,1,-3,)

REPEAT ABNAHME4\_BEAR ABNAHME4\_BEAR\_END

MCALL

---

; 钻孔编程

T2 M6

D1 M3 F2222 S3000

MCALL CYCLE81(10,0,1,-12,)

REPEAT ABNAHME4\_BEAR ABNAHME4\_BEAR\_END

MCALL

---

GOTOF ABNAHME4\_BEAR\_END

ABNAHME4\_BEAR:

REPEAT ABNAHME4\_CONT ENDLABEL

CYCLE73(1015,"ABNAHME4\_DRILL","ABNAHME4\_MILL1",  
"3",10,0,1,-12,0,,2,0.5,,2000,400,0,,))

ABNAHME4\_BEAR\_END:

---

; 扩孔编程

T3 M6

D1 M3 S4000

REPEAT ABNAHME4\_CONT ENDLABEL

CYCLE73(1011,"","ABNAHME4\_MILL1","3",10,0,1,  
-12,0,,2,0.5,,2000,400,0,,))

---

; 剩余材料扩孔编程

T6 M6

D1 M3 S4000

REPEAT ABNAHME4\_CONT ENDLABEL

CYCLE73(1012,"","ABNAHME4\_2\_MILL4","3",10,0,1  
,  
-12,0,,2,0.5,,1500,800,0,,))

---

; 精加工编程

T5 M6

D1 M3 S4500

REPEAT ABNAHME4\_CONT ENDLABEL

CYCLE73(1013,"","ABNAHME4\_MILL3","3",10,0,1,  
-12,0,,2,,3000,700,0,,))

M30

---

**边缘轮廓，编程示例2:**

```
%_N_RANDA01_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD
; Ste 17.05.99
```

边缘轮廓，编程示例2:

```
N5 G0 G90 X260 Y0
N7 G3 X260 Y120 CR=60
N8 G1 X170 RND=15
N9 G2 X70 Y120 CR=50
N10 G1 X0 RND=15
N11 Y0 RND=15
N35 X70 RND=15
N40 G2 X170 Y0 CR=50
N45 G1 X260 Y0
N50 M30
```

**中心岛轮廓，编程示例2**

```
%_N_INS1A01_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD
; Ste 18.06.99
```

; 中心岛轮廓，编程示例2

```
N5 G90 G0 X30 Y15
N10 G91 G3 X0 Y30 CR=15
N12 X0 Y-30 CR=15
N15 M30
```

```
%_N_INS11A01_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD
; Ste 18.06.99
```

; 中心岛轮廓，编程示例2

```
N5 G90 G0 X30 Y70
N10 G91 G3 X0 Y30 CR=15
N12 X0 Y-30 CR=15
N15 M30
```

```
%_N_INS2A01_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD
; Ste 18.06.99
```

; 中心岛轮廓，编程示例2

```
N5 G90 G0 X200 Y40
N10 G3 X220 Y40 CR=10
N15 G1 Y85
N20 G3 X200 Y85 CR=10
N25 G1 Y40
N30 M30
```

**3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75**

```
%_N_INS3A01_MPF  
;$PATH=/_N_WKS_DIR/_N_CC73BEI2_WPD  
;Ste 18.06.99  
; 中心岛轮廓, 编程示例2
```

```
N5 G0 G90 X265 Y50
```

```
N10 G1 G91 X20
```

```
N15 Y25
```

```
N20 G3 X-20 I-10
```

```
N25 G1 Y-25
```

```
N30 M30
```



### 编程举例3

加工任务:

表明一个加工任务的编程步骤，描述了两个不同的带中心岛的凹槽。加工过程与刀具紧密相连，也就是说在使用下一把刀进行加工之前，必须使用当前刀具加工完毕两个凹槽中所有可能的加工任务。

- 预制钻孔
- 带中心岛的凹槽扩孔
- 剩余材料扩孔

```

%_N_BEISPIEL3_MPF
; $PATH=/_N_WKS_DIR/_N_CC73BEI3_WPD
; 示例3
; 07.04.2000

; 刀具补偿数据
$TC_DP1[2,1]=220 $TC_DP3[2,1]=330 $TC_DP6[2,1]=10
$TC_DP1[3,1]=120 $TC_DP3[3,1]=210 $TC_DP6[3,1]=12
$TC_DP1[6,1]=120 $TC_DP3[6,1]=199 $TC_DP6[6,1]=6

; 工件零点
; G54
$P_UIFR[1,X,TR]=620
$P_UIFR[1,Y,TR]=50
$P_UIFR[1,Z,TR]=-320
; G55
$P_UIFR[2,X,TR]=550
$P_UIFR[2,Y,TR]=200
$P_UIFR[2,Z,TR]=-320
;
N10 G0 G17 G54 G40 G90
N20 T2
M6
D1 M3 F2000 S500 M8
N30 G0 Z20

; 加工轮廓凹槽1
GOTOF ENDLABEL
TASCHE1_CONT:
CYCLE74 ("RAND", " ", " ")
CYCLE75 ("INSEL1", " ", " ")
CYCLE75 ("INSEL2", " ", " ")
ENDLABEL:

```

## 3.15 带中心岛的凹槽铣削- CYCLE73, CYCLE74, CYCLE75

```

; 加工轮廓凹槽2
GOTOF ENDLABEL
BEISPIEL2_CONT:
CYCLE74("RANDA01",,,)
CYCLE75("INS11A01",,,)
CYCLE75("INS1A01",,,)
CYCLE75("INS2A01",,,)
CYCLE75("INS3A01",,,)
ENDLABEL:

; 钻削
T2 M6
D1 M3 F6000 S4000
MCALL CYCLE81(10,0,1,-8,)
REPEAT TASCHE1_BEAR TASCHE1_BEAR_END
MCALL

G55
MCALL CYCLE81(10,0,1,-8,)
REPEAT BEISPIEL2_BEAR BEISPIEL2_BEAR_END
MCALL

; 扩孔凹槽1
T3 M6
G54 D1 M3 S3300
GOTOF TASCHE1_BEAR_END
TASCHE1_BEAR:
REPEAT TASCHE1_CONT ENDLABEL
CYCLE73(1025,"TASCHE1_DRILL","TASCHE1_MILL1","3",10,0,1,-8,0,0,2,0,0,2000,400,0,0,0,3,4)
TASCHE1_BEAR_END:
REPEAT TASCHE1_CONT ENDLABEL
CYCLE73(1021,"TASCHE1_DRILL","TASCHE1_MILL1","3",10,0,1,-8,0,0,2,0,0,2000,400,0,0,0,3,4)

; 扩孔凹槽2
G55
GOTOF BEISPIEL2_BEAR_END
BEISPIEL2_BEAR:
REPEAT BEISPIEL2_CONT ENDLABEL
CYCLE73(1015,"BEISPIEL2_DRILL","BEISPIEL2_MILL1","3",10,0,1,-8,0,0,2,0,0,2000,400,0,0,0,3,4)
BEISPIEL2_BEAR_END:
REPEAT BEISPIEL2_CONT ENDLABEL
CYCLE73(1011,"BEISPIEL2_DRILL","BEISPIEL2_MILL1","3",10,0,1,-8,0,0,2,0,0,2000,400,0,0,0,3,4)

```



; 扩孔剩余材料凹槽1和凹槽2

T6 M6

D1 G54 M3 S222

REPEAT TASCHE1\_CONT ENDLABEL

CYCLE73(1012,"", "TASCHE1\_3\_MILL2", "3", 10, 0, 1, -8, 0, , 2, , , 2500, 800, 0, , , , )

G55

REPEAT BEISPIEL2\_CONT ENDLABEL

CYCLE73(1012,"", "BEISPIEL2\_3\_MILL2", "3", 10, 0, 1, -8, 0, , 2, , , 2500, 800, 0, , , , )

G0 Z100

M30

; 边缘和中心岛轮廓

; 凹槽2与程序示例2相同

凹槽1:

%\_N\_Rand\_MPF

;\$PATH=/\_N\_WKS\_DIR/\_N\_CC73BEI3\_WPD

;29.03.99

N1 G0 X0 Y0 G90

N3 G1 X200 Y0

N5 X200 Y100

N10 X0 Y100

N20 X0 Y0

M30

%\_N\_INSEL1\_MPF

;\$PATH=/\_N\_WKS\_DIR/\_N\_CC73BEI3\_WPD

;29.03.99

N100 G0 X130 Y30 Z50 G90

N110 G1 X150 Y30

N120 X150 Y60

N130 X130 Y60

N200 X130 Y30

M30

%\_N\_INSEL2\_MPF

;\$PATH=/\_N\_WKS\_DIR/\_N\_CC73BEI3\_WPD

;29.03.99

N12 G0 X60 Y20

N13 G1 X90 Y20

N14 X90 Y50

N30 X60 Y50

N40 X60 Y20

M30



## 说明

## CYCLE73...CYCLE75 报警

报警号码	报警文本	说明, 纠正
61703	“在文件中删除内部循环故障”	
61704	“在文件中写出内部循环故障”	
61705	“在文件中读出内部循环故障”	
61706	“在检查和构成中内部循环故障”	
61707	“在ACTIVATE到MMC时故障”	
61708	“在READYPROG到MMC时故障”	
61900	“没有轮廓出现”	
61901	“轮廓不闭合”	
61902	“没有存储器供使用”	
61903	“太多轮廓单元”	
61904	“太多交叉点”	
61905	“铣刀半径太小”	
61906	“太多轮廓”	
61907	“圆弧没有圆心说明”	
61908	“起始点说明出错”	
61909	“螺旋线半径太小”	
61910	“螺旋线破坏轮廓”	
61911	“要求更多的插入点”	
61912	“不要产生轨迹”	
61913	“没有产生剩余材料”	
61914	“编程的螺旋线破坏轮廓”	
61915	“返回运行/离开运行破坏轮廓”	
61916	“斜坡太短”	
61917	“如果小于 50% 搭接, 则可能会有残余角”	
61918	“剩余材料的铣刀半径太大”	
61980	“中心岛轮廓出错”	
61981	“边缘轮廓上出错”	
61982	“平面中进刀宽度太大”	
61983	“凹槽边缘轮廓出错”	
61984	“刀具参数_TN没有定义”	
61985	“钻孔位置程序名称出错”	
61986	“凹槽铣削程序出错”	
61987	“钻孔位置程序出错”	
61988	“凹槽铣削程序名称出错”	
61989	“没有编程D1作为有效的刀具边沿”	

### 3.16 回转- CYCLE800

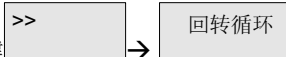


#### 回转调用 – CYCLE800

入口区程序/铣削

当调试回转数据程序段时 (MD

18088:MM\_NUM\_TOOL\_CARRIER

>0), 显示软键  (MM\_NUM\_TOOL\_CARRIER >0).



回转不是选项, 供NCSW6.3(CCU SW 6.3)使用。

功能

- 3/2 轴斜置加工和
- 可定向刀架

供基本型使用。



文献: 840D/840Di/810D 功能描述

/W1/ “刀具补偿”

/R2/ “回转轴”

/K2/ “系统框架”

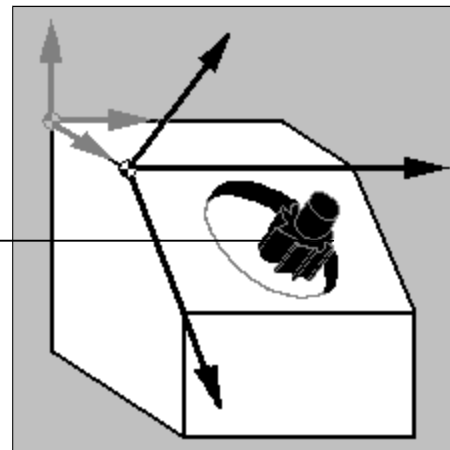


功能

循环用于回转到一个任意的表面, 以加工或者测量该表面。通过循环, 在考虑机床运动链的情况下, 通过调用相应的 NC 功能将有效的工件零点和刀具补偿换算到斜置表面上, 并定位回转轴 (选择性地)。

回转可以选择轴向进行, 作为投影角和立体角。

在定位回转轴之前, 线性轴可以选择空运行。



机床运动

1. 可回转的刀架 (回转头) → T型
2. 可回转的刀架 (回转台) → P型
3. 由第一点和第二点混合的运动 → M型



另见章节 | 3.16.7 中的“偏移回转轴矢量 (机床运动)”标题下的内容。

**重要**

第一次调用主程序的回转循环之前必须编程刀具（刀具刀沿  $D>0$ ）和零点偏移（NV），用该回转循环碰撞接触或测量工件。在回转循环中，该零点偏移换算至相应的加工平面上。保留 NV 值。转换和旋转部分存储在系统框架（回转框架）中（参见 HMI → 参数，有效的 NV）：

- 刀具基准（TOOLFRAME）
- 刀架（PARTFRAME）
- 工件基准（WPFRAME）

回转循环考虑当前的加工平面（G17、G18、G19）。

回转到一个加工表面或辅助表面时始终有3步：

- 在旋转之前平移基准点（根据 TRANS 或者 ATRANS）
- 旋转（根据 AROT 或者 AROTS）
- 在旋转之后平移（根据 ATRANS）



**偏移和旋转与机床相关并以工件坐标系 X, Y, Z 为基准。**

在回转循环中未使用可编程的框架。在附加回转时，考虑由用户编程的框架。

回转到一个新的回转平面时，删除可编程的框架（TRANS）。

可选择性地通过机床数据设定，程序复位后或断电时，最后的回转平面保持有效。可在回转平面上进行任意加工，如通过调用标准循环或测量循环。

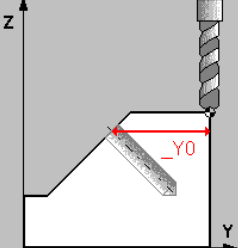
## 3.16.1 操作、参数和输入屏幕



## 参数说明

CYCLE800标准界面的输入屏幕

Swivel cycle/CYCLE800 Reference point for rotation around Y axis



Name:	
Retract:	Z
Schwenken:	Yes
Swivel plane:	new
Ref. point:	X0 0.000
	Y0 25.000
	Z0 0.000
Swivel mode:	axis by ax.
Rot. around	X (A) -15.000
Rot. around	Y (B) 0.000
Rot. around	Z (C) 0.000
Zero point:	X1 0.000
	Y1 0.000
	Z1 0.000
Direction:	Minus
Tracking TL	No

旋转之前的基准点

旋转

旋转之后的零点

## 回转数据组名称 \_TC

可以选择所调试的回转数据组（参见IBN CYCLE800）（选择键）。

每个回转数据组包含一个名称。如果仅有一个回转数据组，不必约定名称。

“0” → ； 撤消回转数据组选择

## 空运行 \_FR（在回转回转轴之前）

- 不空运行
- Z轴空运行<sup>1)</sup>
- Z轴、XY空运行<sup>1)2)</sup>
- 最大刀具方向的空运行（自循环 SW 6.5 起）<sup>2)</sup>
- 增量刀具方向的空运行（自循环 SW 6.5 起）<sup>2)</sup>。

在输入栏中输入用于刀具方向上的运行位移的增量值。

空运行位置可以登记到IBN菜单 CYCLE800 中。



1) 以绝对值运行空运行位置。如果想要一个其它的顺序或一个增量定位，可在开机调试时在用户循环 TOOLCARR中对其进行修改。

2) 仅当在 CYCLE800 IBN 菜单 中有效时。

**说明:**

在用标准循环进行编程时，以及编程一个较大的退回平面距离和较大的回转角时（在多面加工时回转90度），机床的运行空间可能不够（软件限位错误），因为在返回运行时始终是首先运行加工平面（在G17中为XY平面），然后运行横向进刀轴（Z）。

这种性能可以通过减小退回平面进行优化。

**回转，方向\_DIR**

- **回转 是**

回转轴被定位，或者手动回转轴可以由操作人员旋入。

- **回转 否（仅计算）**

如果要求在激活回转循环之后回转轴不运行，则适用于选择“回转 否”。

应用：辅助回转平面根据零件图纸。应用：辅助回转平面根据零件图纸。

- **正/负方向**

在选择运行方向用于回转循环时，以回转轴 1 或 2 为基准。通过机床运动回转轴的角度范围可由NCU 计算两个可能的解决方案。这里，通常工艺上要求有一个解决方案。

在 CYCLE800 IBN 菜单 中选择，这两个解决方案要以哪根回转轴（第 1 或第 2 根轴）为基准（参见章节 3.16.7）。

通过选择“正”或“负”方向来选择，要运行这两个可能的解决方案中的哪一个。

注意机床制造商的说明！

**回转平面 \_ST**

- **新**

到目前为止的回转框架和编程的框架被删除，在输入屏幕中定义的值构成新的回转框架。

每个主程序必须以回转平面**重新**开始一个回转循环，从而确保没有一个其它程序的回转框架生效。

- **附加**

该回转框架附加到最后回转循环的回转框架上。

如果在一个程序中编程了几个回转循环，并且在这些回转循环之间还有附加的可编程框架生效（比如AROT ATRANS），则这一点要在回转框架中加以考虑。

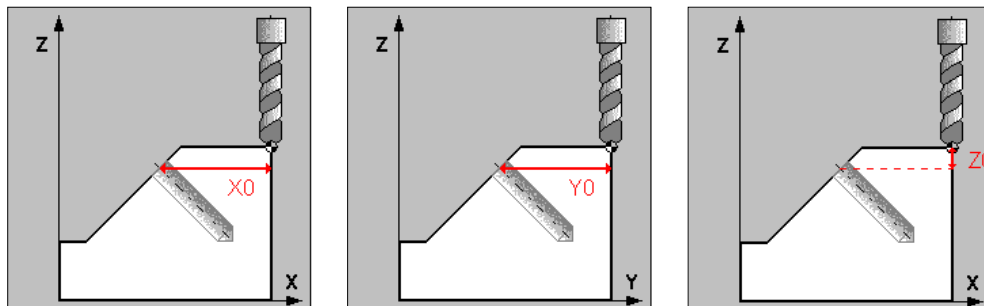


如果当前有效的零点偏移NV中包含旋转，比如由于事先测量工件，则这一点在回转循环中要加以考虑。

下面的辅助图形以加工平面G17为基准（工具轴Z）。

### 旋转前的基准点 X0, Y0, Z0

基准点



### 回转模式 \_MODE

使用该参数确定轴的回转模式。

- 轴方式
- 投影角<sup>1)2)</sup>
- 立体角<sup>1)</sup>

回转模式始终以工件坐标系为基准，它与机床无关。

在CYCLE800的IBN菜单中可以设定，哪一种回转模式可以使用。

- 进行轴方式回转时，围绕坐标系的各个轴旋转，此时，可将每个旋转置于前一个旋转上。可自由选择轴顺序。
- 用投影角回转时，回转平面的角度值被投影至坐标系前两根轴上。第三个旋转置于前一个旋转上。可自由选择轴顺序。
- 通过立体角回转时，首先围绕 Z 轴旋转，然后围绕 Y 轴旋转。第二个旋转置于第一个旋转上。

不同的回转变量其正方向旋转可以从辅助图形中了解。

1) 仅当机床制造商在 CYCLE800 IBN 菜单 中选择时可使用。

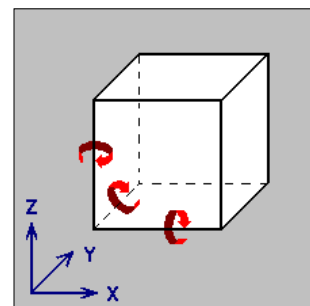
2) 编程 X 和 Y 时，新的 X 轴位于 Z-X 平面中。

编程 Z 和 X 时，新的 Z 轴位于 Y-Z 平面中。

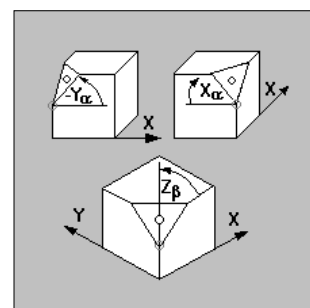
编程 Y 和 Z 轴时，新的 Y 轴位于 X-Y 平面中。

第三次旋转在所旋转的平面中进行。

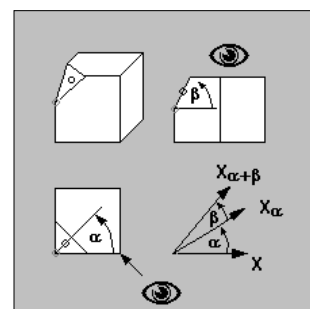
轴方式



投影角

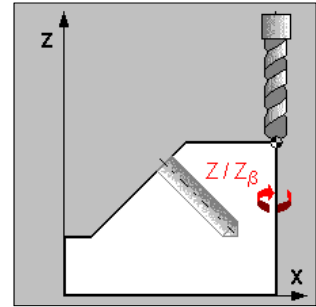
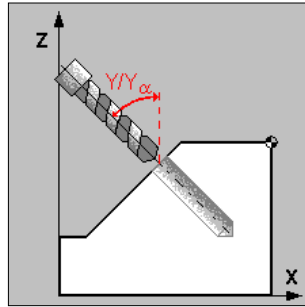
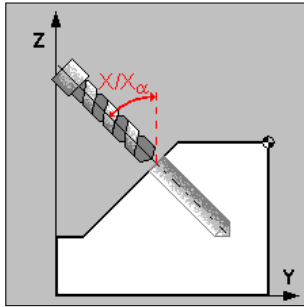


立体角

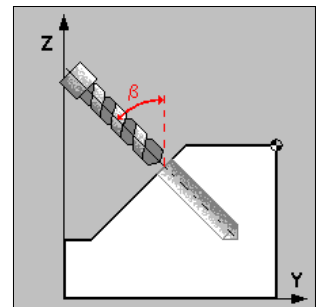
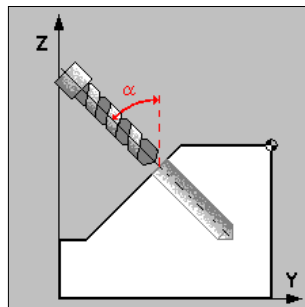


### 围绕 A, B, C 旋转

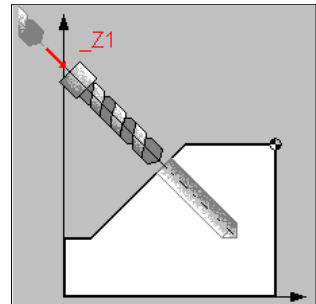
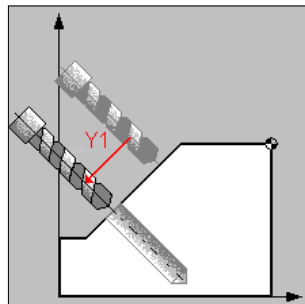
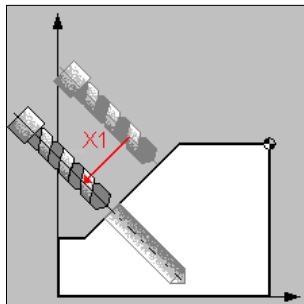
- 旋转（轴方式，投影角）



- 旋转（立体角）



### 旋转 X1, Y1, Z1 后的零点



### 跟踪运行 WZ\_TC\_N\_WZ

可以通过CYCLE800 IBN菜单隐藏跟踪运行显示区。

- 是：

在回转到一个加工平面时，为了防止轮廓碰撞，可以跟随线性轴运行。

前提条件：

1. 要求选项TRAORI。
2. 机床制造商相应地匹配到用户循环TOOLCARR.SPF。

- 否：

回转时，无线性轴跟踪运行。



### 3.16.2 操作说明, 编程说明



- 仅用CYCLE800 (TOOLCARR.SPF) 在带有回转加工平面 (CYCLE800) 的 NC 程序中定位回转轴切换 到5 轴转换除外。
- **调用5轴转换的说明 (TRAORI)**  
如果在回转的加工平面中执行一个程序, 该程序启动5轴转换 (TRAOR), 则在调用TRAORI之前必须关闭回转头/回转台的系统框架 (参见示例)。保留工件基准 (WPFRAME)。

举例 (带回转台的机床)

```

N1 G54
N2 T="MILL_10mm"
N3 M6
N4 CYCLE800(1,"",0,57,0,40,0,-45,0,0,0,0,-1) ; 回转循环
N5 CYCLE71(50,24,2,0,0,0,80,60,0,4,10,5,0,2000,31,5) ; 平面铣削
N6 TCARR=0 ; 撤消回转数据组选择
N7 PAROTOF
N8 TOROTOF ; (仅在机床运动类型 "T" 和 "M" 中)
N9 TRAORI
N10 G54 ; 零点偏移重新计算
N11 EXTCALL „WALZ“ ; 5 轴加工程序, 带方向矢量 (A3、B3、C3)
N12 M2

```

如果用循环“高速设定”CYCLE832启动 5 轴转换, 则可以取消程序段N6...N10。

- 如果机床运动中回转轴作为手动轴 (IBN菜单CYCLE800), 则待设定的回转角在删除报警 62180/62181中显示。在旋入回转角之后, 按NC启动继续NC程序。
- 当机床控制面板上的 WCS 键激活时, 可在有效的回转平面中以BA JOG 方式运行轴。因此, 运行的是机床轴, 而表示几何轴。
- 通过编程 **CYCLE800()** 可以撤消选择回转数据组, 删除回转框架 (WPFRAME、PARTFRAME、TOOLFRAME)。
- 在CYCLE800中作为输入值也可以传送参数 (比如测量循环结果变量\_OVR[19]。)
- 如果在回转轴的有效 NV 中输入值 (工件基本旋转), 则在回转时要考虑到这个值。回转到机床运动 (...0,0,0,...) [旋转=0]调试人员初始位置 (参见章节 3.16.5), 由 CYCLE800 相应校准 WCS。由此在回转框架 \$P\_WPFRAME (工件基准) 中得出旋转。

## 3.16.3 参数



## 编程

CYCLE800(\_FR, \_TC, \_ST, \_MODE, \_X0, \_Y0, \_Z0, \_A, \_B, \_C, \_X1, \_Y1, \_Z1, \_DIR, \_FR\_I)



## 参数

_FR	整数	空运行 值： 0...没有空运行 1...空运行轴 Z (标准) <sup>1)</sup> 2...空运行轴 Z、X、Y <sup>1)</sup> 4...刀具方向上的空运行，最大 <sup>2)</sup> 5...刀具方向上的空运行，增量 <sup>2)</sup>																																																																																																															
_TC	字符串[20]	回转数据组名称 “0” 撤消选择数据组																																																																																																															
_ST	整数	回转平面 个位： 值： 0... 新的 1... 附加的 十位 <sup>2)</sup> ： 前提条件：必须有选项 TRAORI。 值： 0x... 刀尖不跟踪运行 1x... 刀尖跟踪运行 百位： 值： 1xx... 设置刀具																																																																																																															
_MODE	整数	回转模式 角度运用： 参数模式为十进制，编码为二进制。 <div style="margin-left: 40px;"> <table style="border-collapse: collapse;"> <tr> <td style="text-align: center; border-bottom: 1px solid black;">7</td> <td style="text-align: center; border-bottom: 1px solid black;">6</td> <td style="text-align: center; border-bottom: 1px solid black;">5</td> <td style="text-align: center; border-bottom: 1px solid black;">4</td> <td style="text-align: center; border-bottom: 1px solid black;">3</td> <td style="text-align: center; border-bottom: 1px solid black;">2</td> <td style="text-align: center; border-bottom: 1px solid black;">1</td> <td style="text-align: center; border-bottom: 1px solid black;">0</td> <td></td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">01 : 围绕第 1 根轴旋转</td> <td rowspan="3" style="border-left: 1px solid black; vertical-align: middle;">} 旋转角度 1</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">10 : 围绕第 2 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">11 : 围绕第 3 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">01 : 围绕第 1 根轴旋转</td> <td rowspan="3" style="border-left: 1px solid black; vertical-align: middle;">} 旋转角度 2</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">10 : 围绕第 2 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">11 : 围绕第 3 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">01 : 围绕第 1 根轴旋转</td> <td rowspan="3" style="border-left: 1px solid black; vertical-align: middle;">} 旋转角度 3</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">10 : 围绕第 2 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td style="border-left: 1px solid black;">11 : 围绕第 3 根轴旋转</td> </tr> <tr> <td colspan="7"></td> <td>00 : 轴方式回转角度 (A、B、C)</td> <td></td> </tr> <tr> <td colspan="7"></td> <td>01 : 立体角 (A、B)</td> <td></td> </tr> <tr> <td colspan="7"></td> <td>10 : 投影角 (A、B、C)</td> <td></td> </tr> </table> </div> <div style="margin-left: 40px;"> <p>说明：在立体角时位0到5没有意义</p> <p>编码举例： →            轴方式旋转 zyx</p> <p>二进制：00011011        十进制： 27</p> </div>	7	6	5	4	3	2	1	0									01 : 围绕第 1 根轴旋转	} 旋转角度 1								10 : 围绕第 2 根轴旋转								11 : 围绕第 3 根轴旋转								01 : 围绕第 1 根轴旋转	} 旋转角度 2								10 : 围绕第 2 根轴旋转								11 : 围绕第 3 根轴旋转								01 : 围绕第 1 根轴旋转	} 旋转角度 3								10 : 围绕第 2 根轴旋转								11 : 围绕第 3 根轴旋转								00 : 轴方式回转角度 (A、B、C)									01 : 立体角 (A、B)									10 : 投影角 (A、B、C)	
7	6	5	4	3	2	1	0																																																																																																										
							01 : 围绕第 1 根轴旋转	} 旋转角度 1																																																																																																									
							10 : 围绕第 2 根轴旋转																																																																																																										
							11 : 围绕第 3 根轴旋转																																																																																																										
							01 : 围绕第 1 根轴旋转	} 旋转角度 2																																																																																																									
							10 : 围绕第 2 根轴旋转																																																																																																										
							11 : 围绕第 3 根轴旋转																																																																																																										
							01 : 围绕第 1 根轴旋转	} 旋转角度 3																																																																																																									
							10 : 围绕第 2 根轴旋转																																																																																																										
							11 : 围绕第 3 根轴旋转																																																																																																										
							00 : 轴方式回转角度 (A、B、C)																																																																																																										
							01 : 立体角 (A、B)																																																																																																										
							10 : 投影角 (A、B、C)																																																																																																										
_X0, _Y0, _Z0	实数	旋转之前的基准点																																																																																																															



- 1) - 当在 IBN 回转中约定时，
  - 可在用户循环 TOOLCARR 中匹配。
  - 注意机床制造商的说明！
- 2) 参见参数分配结束部分的说明

<b>_A</b>	实数	1. 轴交角（轴方式回转模式） 2. 在XY平面中围绕Z轴的旋转角（立体角回转模式） 3. 轴交角（通过投影角回转），轴顺序
<b>_B</b>	实数	1. 轴交角（轴方式回转模式） 2. 空间旋转角，围绕Y轴（立体角回转模式）
<b>_C</b>	实数	轴交角（轴方式回转模式，投影角）
<b>_X1, _Y1, _Z1</b>	实数	旋转之后的零点
<b>_DIR</b>	整数	方向 如果在调用回转循环时由 <b>NC</b> 计算出两个解决方案，则操作员可选择一个优先方向。由机床制造商确定，优先方向以哪根轴为基准。 值： -1 (负)...回转轴较小的值 (标准) +1 (正)...回转轴较大的值 0...回转轴没有运行 (只有计算)
<b>_FR_I</b>	实数	增量刀具方向的空运行值（自循环 <b>SW 6.5</b> 起）



2)

### 刀具方向的空运行（自循环 **SW 6.5** 起）

自循环 **SW 6.5** 起，如下扩展之前的空运行模式：

- “**最大刀具方向的空运行**”  
在回转前，空运行刀具轴直至软件限位。
- “**增量刀具方向的空运行**”  
刀具轴围绕增量输入值（**\_FR\_I**）空运行。

总是在正**刀具方向**（离开工件）进行最大刀具方向和增量刀具方向的空运行。



### 总是在回转回转轴之前进行刀具方向的空运行。

如果开始程序时，回转平面不确定，则首先要运行至安全位置或用空运行安全预定位 **Z**。

在用回转头进行加工前，在最大和增量刀具方向空运行有意义。如果涉及到带有回转头的机床且回转框架有效（工件基准中的旋转不等于 0），可在运行时，以刀具方向运行多根轴。

举例：

对于一个带有回转头的机床，**G17** 平面中的头（和工件坐标系）围绕 **X** 轴回转 **90** 度。

**Z** 轴运行时，**Y** 在程序中也运行。以刀具方向空运行时，机床轴运行直至正向 **SW** 结束位置。

空运行模式系列（可设定的**CYCLE800 IBN** 菜单）

回转前没有空运行	空运行：否
回转前进行空运行	空运行： <b>Z</b>
回转前空运行加工轴	空运行： <b>Z、XY</b>
最大刀具方向的空运行（自循环 <b>SW 6.5</b> 起）	空运行：最大刀具方向
增量刀具方向的空运行（自循环 <b>SW 6.5</b> 起）	空运行：增量刀具方向，例如 <b>100.00</b>



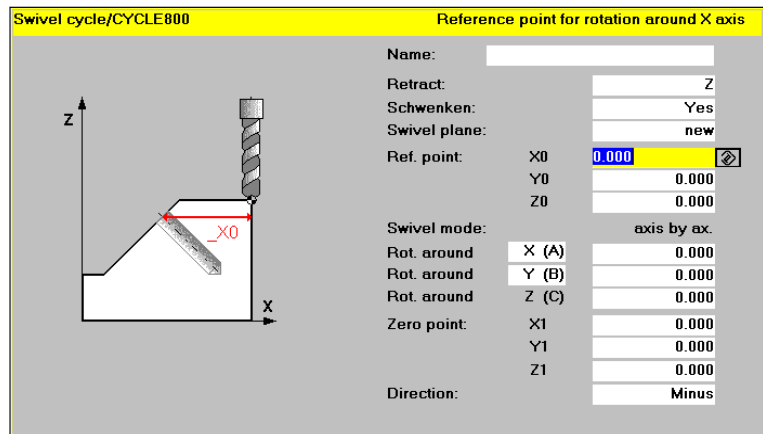
## 编程举例1

## 设定回转平面NULL

```

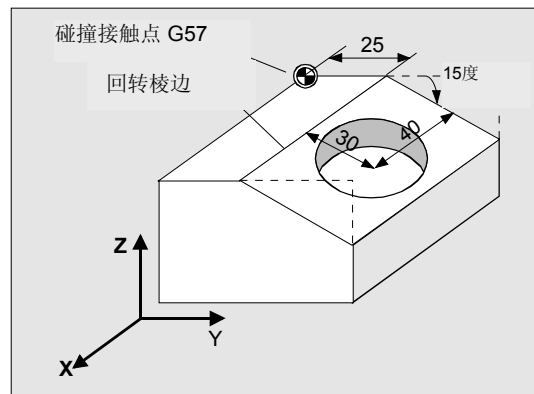
%_N_SCHWENK_0_SPF
; $PATH=/_N_WKS_DIR/_N_HAA_SCHWENK_WPD
G54
CYCLE800(1, "", 0, 57, 0, 0, 0, 0, 0, 0, 0, 0, -1)
M2

```



## 编程举例2

在一个15度回转的加工平面中进行一个环形凹槽的平面铣削和铣削



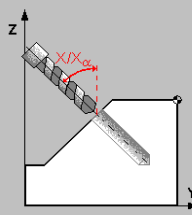
```

%_N_SCHWENK_KREISTASCHE_SPF
; $PATH=/_N_WKS_DIR/_N_HAA_SCHWENK_WPD
N12 T="MILL_26mm"
N14 M6
N16 G57
N18 CYCLE800(1, "", 0, 57, 0, 0, 0, 0, 0, 0, 0, 0, 1)
N20 M3 S5000
N22 CYCLE71(50, 2, 2, 0, 0, 0, 80, 60, 0, 4, 15, 5, 0, 2000, 31, 5)

```

; 平面铣削

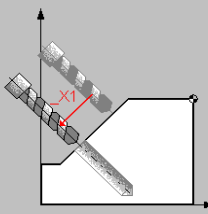
N24 CYCLE800(1, "", 0, 57, 0, 25, 0, -15, 0, 0, 0, 0, -1)

Swivel cycle/CYCLE800		Angle of 1st rotation
		
Name:		
Retract:	Z	
Schwenken:	Yes	
Swivel plane:	new	
Ref. point:	X0	0.000
	Y0	25.000
	Z0	0.000
Swivel mode:	axis by ax.	
Rot. around	X (A)	-15.000 <input checked="" type="checkbox"/>
Rot. around	Y (B)	0.000
Rot. around	Z (C)	0.000
Zero point:	X1	0.000
	Y1	0.000
	Z1	0.000
Direction:	Minus	

N26 CYCLE71(50, 12, 2, 0, 0, 0, 80, 60, 0, 4, 10, 5, 0, 2000, 31, 5)

; 平面铣削

N28 CYCLE800(1, "", 1, 57, 0, 0, 0, 0, 40, 30, 0, 1)

Swivel cycle/CYCLE800		Zero point of X axis swivel plane
		
Name:		
Retract:	Z	
Schwenken:	Yes	
Swivel plane:	new	
Ref. point:	X0	0.000
	Y0	0.000
	Z0	0.000
Swivel mode:	axis by ax.	
Rot. around	X (A)	0.000
Rot. around	Y (B)	0.000
Rot. around	Z (C)	0.000
Zero point:	X1	40.000 <input checked="" type="checkbox"/>
	Y1	30.000
	Z1	0.000
Direction:	Minus	
Tracking TL	No	

N30 T="MILL\_10mm"

N32 M6

N34 M3 S5000

N36 POCKET4(50, 0, 1, -15, 20, 0, 0, 4, 0.5, 0.5, 1000, 1000, 0, 11, , , , , ); 环形凹槽

N38 POCKET4(50, 0, 1, -15, 20, 0, 0, 4, 0, 0, 1000, 1000, 0, 12, , , , , )

N40 M2

### 3.16.4 设置刀具（自循环SW 6.5 起）

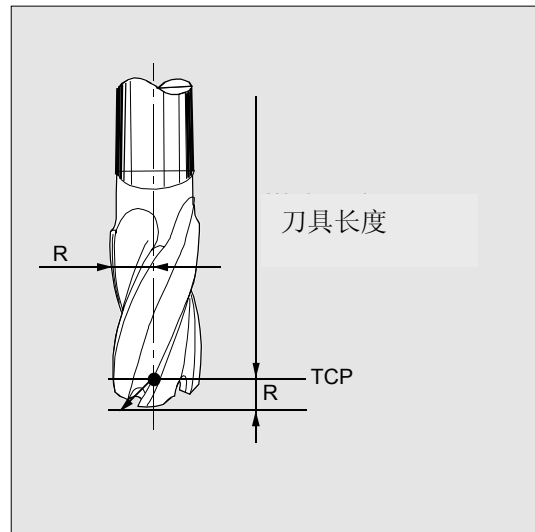
回转后，刀具方向总是垂直于加工平面。

用半径铣刀铣削时，可能在工艺上要求，在一个角度下设置用于表面标准矢量的刀具。

在回转循环时，通过某根轴旋转（最大  $\pm 90$  度）到有效的回转平面而生成后角。设置时，回转平面总是为“添加”。“设置刀具”时在回转循环输入屏幕中：仅显示旋转。可自由选择旋转顺序。



可输入TCP（刀具中心点Tool Center Point）作为半径铣刀的刀具长度。



如果要在带有多个回转循环的程序中使用设置后的刀具铣削，则总是以下列回转调用“重新”设定回转平面。

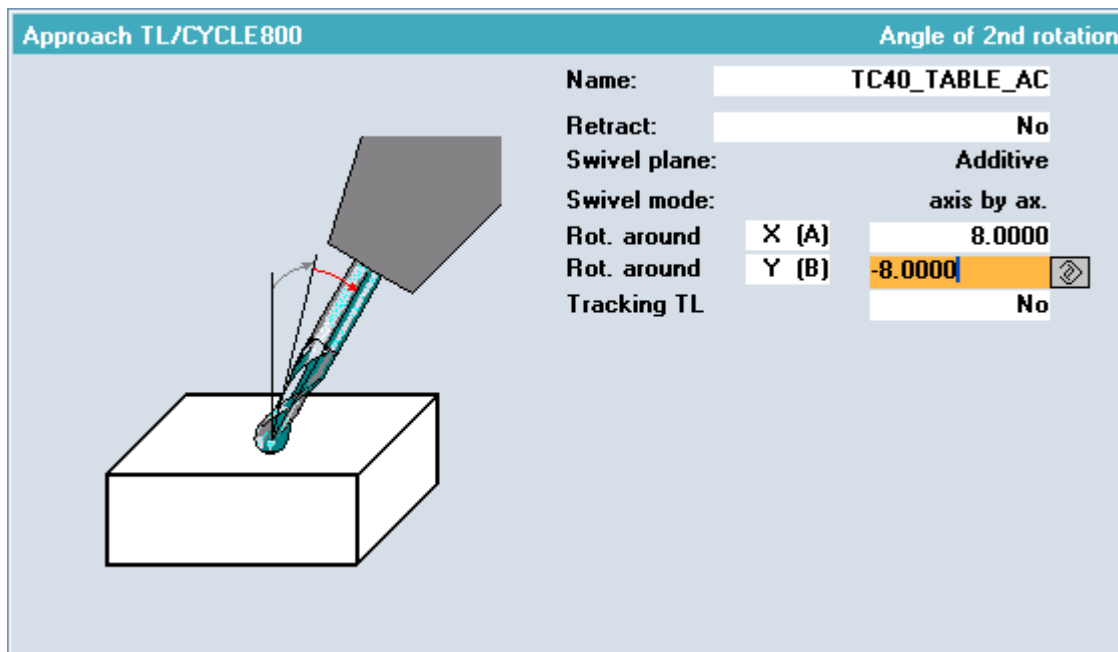
对于不带选项TRAORI的机床也适用于刀具设置（参见功能提前角 LEAD和侧向角TILD）。



调用

铣削 >> 设置刀具（4. 垂直软键）

输入屏幕



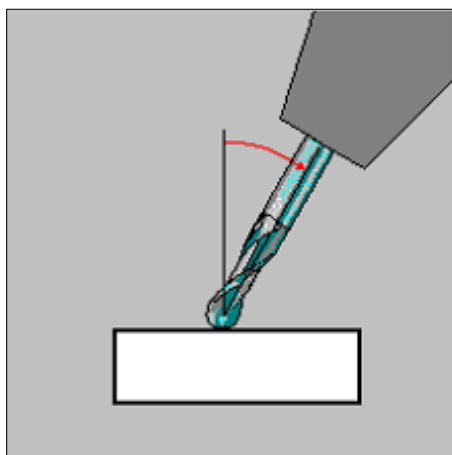
输入屏幕“设置刀具”与关闭的CYCLE800 输入屏幕相一致。

这表示，在输入屏幕中：

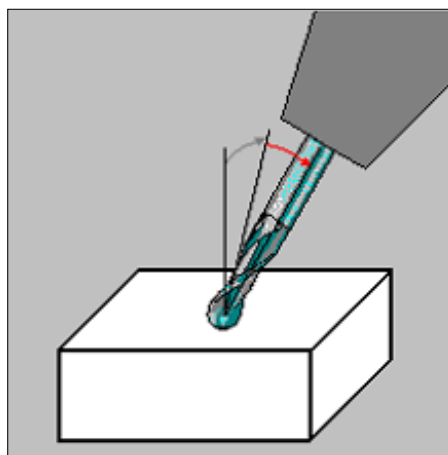
名称 _TC:	当前回转数据组
空运行 _FR:	同 CYCLE800
回转平面 _ST:	附加的（仅用于显示）
回转模式 _MODE:	轴方式（仅用于显示）
旋转:	在 WCS XY、YZ、XZ 中，最多围绕两根轴旋转
跟踪刀具:	同 CYCLE800（有关显示选项，参见 IBN CYCLE800）



设置刀具帮助画面



设置刀具：1旋转



设置刀具：2旋转

### 3.16.5 调试带有回转加工平面的工件 （自循环 SW 6.5 起）



在 JOG 运行方式下回转

“在JOG方式下回转”功能用于调试带回转加工平面的工件并说明其它调试过程的前提条件（棱边/拐角碰撞接触/测量）。

通过一个输入屏幕支持JOG运行方式下的回转。

用 NC 启动键定位回转轴（旋入手动回转轴时）并计算该平面的回转框架。

可通过“在JOG方式下回转”实现下列功能：

- 在机床上借助回转台补偿工件基本旋转。
- 将回转平面设为新的零点平面
- 将回转平面加到当前的零点平面并设为新的零点平面。

零点平面等于刀具平面（G17、G18、G19），包括有效的零点偏移（G500、G54，...）。



### 在有效的刀具平面中补偿基本旋转

术语“**基本旋转**”在下面用于工件旋转，以应用刀具轴，例如围绕刀具轴 Z（C 回转轴）旋转某个回转台。

操作人员/调试人员的目标是：通过旋转回转台来补偿工件基本旋转，以能够轴向平行加工工件。这可通过下述方式进行：

1. 通过用千分表离开运行工件棱边，此时回转轴一直旋转，直至千分表的偏差值为 0。  
可由操作人员将回转轴的位置值输入至回转轴的零点偏移（NV）中。
2. 通过在 JOG 方式下“通过测量点、钻孔或轴颈校准工件”。

使用所述的基本旋转补偿方法时，操作人员可在 JOG 运行方式下在输入屏幕“回转”中给定**初始位置**并用 NC 启动键相应定位回转轴。定位回转轴以用于基本旋转补偿时对 WCS 无影响。回转轴围绕刀具轴旋转时，WCS 保持不变。



### 将回转平面设置为新的零点平面

操作人员/调试人员必须具备下列基础知识：哪些机床位置、未回转的平面或者**机床运动的初始位置**（运动链）或者刀具方向在该初始位置中。

机床制造商在调试时“回转”功能或 5 轴转换功能时确定机床运动的初始位置。

用回转的（斜置）加工平面调试工件或者斜置夹紧工件时，术语**基本旋转**延伸为**两个回转轴**。

操作人员/调试人员必须基本确定，要将哪些平面（表面）定义为**新的零点平面**。这也可以是机床运动的初始位置。由此，旋转的表面被设为零点平面，将输入屏幕“在 JOG 方式下回转”中的旋转接受至当前的零点偏移。

由此得出一个新的**调试员初始位置**。既能够“重新”返回运行，也能够“附加”返回运行**调试员初始位置**。

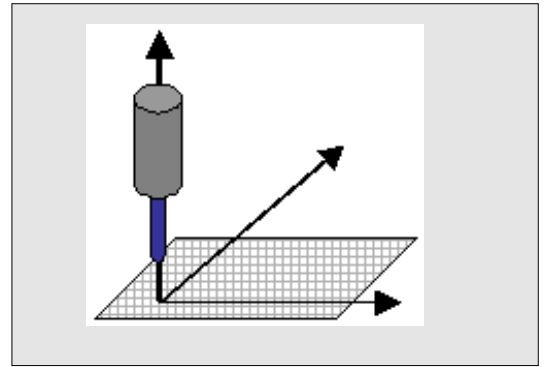
进行**调试员初始位置**返回运行后，通过菜单“碰撞接触”或者“在 JOG 方式下测量”用现有的测量头调试工件。

所有的后续处理，如碰撞接触、用测量头测量、用 NC 工件程序回转都以该零件平面或者**调试员初始位置**为基准。



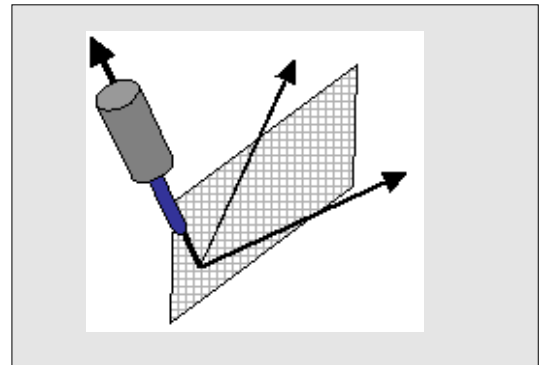
### 机床运动初始位置

- 刀具垂直于平面。
- 平面上的工件坐标系 (WCS) 的位置 “还” 未确定。



### 调试员初始位置

- 有效的 NV 包含旋转或一个/两个回转轴中的偏移。
- 刀具垂直于回转平面。
- 平面上的工件坐标系 (WCS) 的位置 “还” 未确定。



举例:

如果刀具加工运动的初始位置垂直于回转轴

C (垂直的刀具位置), 则在一个机床上, 用一个回转台围绕刀具轴 Z (回转轴) 旋转, 用一个回转头围绕机床轴 Y (回转轴 B) 旋转。

如果要用水平的刀具方向加工某个工件时, 必须首先通过  $Y = 90$  度的一个旋转 (可能与一个  $Z = 180$  度的旋转相联合) 通过 “在 JOG 方式下回转 ” 将刀具定位在水平位置上。

随后, 通过碰撞接触或在 JOG 方式下的测量来确定 NV, 同时刀具方向为水平方向。



### 在“在 JOG 方式下回转”时回转数据的数据传送

原则上，通过功能“在 JOG 方式下回转”仅将旋转写入至激活的框架链中

视旋转或者零点平面的组合方式而定，确定有下列将回转数据传送至当前框架链的方式：

- **回转平面：新 -> 新的零点平面**
  - 删除并级联几何轴 XYZ 中有效 NV<sup>1)</sup> 的旋转
  - 删除回转框架的工件基准部分 (WPFRAME) }
- **回转平面：添加 ->新的零点平面**
  - 描述几何轴 XYZ 中有效 NV<sup>1)</sup> 的旋转
  - 删除回转框架的工件基准部分 (WPFRAME)
- **回转平面：新 (到当前零点平面) <sup>2)</sup>**
  - 删除并级联工件基准 (WPFRAME) 的旋转
- **回转平面：添加 (到当前零点平面) <sup>2)</sup>**
  - 删除并级联工件基准 (WPFRAME) 的旋转



#### 1) NV – 当前零点偏移

如果基本 NV 有效 (G500) 且系统框架“设置零点” (\$P\_SETFRAME) 已设立，则将回转数据写入 SETFRAME 中。

如果基本 NV 有效 (G500) 且系统框架“设置零点” (\$P\_SETFRAME) 未设立，则将回转数据写入最后有效的基准框架中。

#### 2) 仅在 ShopMill/ShopTurn 下。

如果回转平面以当前零点平面为基准，则保留有效 NV 的几何轴 XYZ 的旋转。

当“在 JOG 方式下回转”时不考虑有效 NV 的几何轴 XYZ 的转换部分，且该部分保持不变。也就是说，回转的零点平面上的刀具尚未确定，要通过后面的操作步骤（如碰撞接触和测量）才能确定。

同样，有效 NV 回转轴的转换部分保持不变，且在 CYCLE800 中被考虑到，例如在用回转台加工时，C 轴的基本旋转。

### 3.16.6 操作，参数“在 JOG 方式下回转”（自循环 SW 6.5 起）



#### 输入屏幕参数的说明

- 用于“在 JOG 方式下回转”的进入软键在 JOG 运行方式为水平软键 8。
- “在 JOG 方式下测量”时的进入软键为水平软键 3。

Work	Position	Repos offset
X	-100.0000 mm	0.0000
Y	-100.0000 mm	0.0000
Z	-50.0000 mm	0.0000
C	0.0000 deg	0.0000
A	0.0000 deg	0.0000

Swivel plane	Rot. around	Direction
X2	0.0000 °	Minus
Y2	0.0000 °	
Z2	0.0000 °	

- 用于功能“在 JOG 方式下回转”的输入屏幕与回转 CYCLE800 功能的小型输入屏幕相一致。
- 回转模式（直接）和软键“初始位置”以及“示教”（仅在直接时）已得到扩展。
- 可在开机调试画面“回转”中通过显示选项设置输入栏“空运行”、“方向”和“跟踪刀具”。
- 显示有效的用于确定新的零点平面或者调试员初始位置的 NV。
- 显示有效 NV 的旋转和回转框架的旋转（\$P\_WPFFRAME）。

#### 名称

回转数据组的名称根据参数：

- `_TC`（CYCLE800）或
- `TRAORI`

在激活 5 轴转换和撤消选择回转数据组（TCARR=0）时，显示 TRAORI 或 TRAORI（2）。在这种情况下，借助 5 轴转换返回运行回转平面。

### 空运行

根据参数 `_FR` (空运行) CYCLE800。

### 回转平面

1. 新的回转平面 → 新的零点平面
2. 添加的回转平面 → 新的零点平面

### 回转模式

选择:

- **轴方式**

选择性地围绕 X、Y、Z 旋转，同 CYCLE800

- **直接**

由此可直接给定回转轴的位置。

定位回转轴后（或者旋入手动回转轴），为该位置计算回转框架。在回转模式“直接”中将显示带有有效回转数据组回转轴名称的输入栏，而不是输入栏“围绕……旋转”。在软键 VSK6 上提供“示教”功能，用于接受回转轴的实际值。

回转模式“直接”		
回转模式		直接
回转轴 1	C	-15.00
回转轴 2	B	0.000

### 方向

根据参数 `_DIR` (方向) CYCLE800。

- 负向，根据 `_DIR=-1`
- 正向，根据 `_DIR=1`

可通过回转 IBN 菜单的显示选项隐藏“方向”输入栏。

在直接回转模式下，“方向”输入栏不显示。

### 跟踪刀具

根据参数 `_ST=1x` (跟踪刀尖) CYCLE800。

可通过回转 IBN 菜单的显示选项隐藏“跟踪刀具”输入栏。

- **“跟踪刀具”否**

→ 回转时，没有跟踪刀尖。

- **“跟踪刀具”是**

→ 如果机床制造商已设计了此项，则在回转轴运行过程中，激活/取消激活循环 TOOLCARR.SPF 中的 TRAORI。



注意机床制造商的说明！

### 初始位置 (VSK3)

返回运行**调试人员初始位置**。根据机床运动的初始位置，在考虑回转轴（基本旋转）转换的偏移情况下，框架链中不包含旋转。

在 VSK 初始位置中预设输入屏幕：

- 名称：有效的回转数据组
- 空运行：Z
- 回转模式：轴方式
- 围绕 Z 0.0 旋转
- 围绕 Y 0.0 旋转
- 围绕X 0.0 旋转

### 取消 (VSK7)

→ 退出输入屏幕

没有对回转平面进行更改也没有进行机床运行运动。

### OK (VSK8)

→ 确认输入参数

用 **OK** 确认输入参数值接受后，通过文本信息：“启动旋转轴 -> 确认NC 启动”要求操作人员/调试人员用 **NC** 启动返回运行回转平面。



### 对激活 5 轴转换的说明 (TRAORI)

激活 5 轴转换时，显示输入屏幕“在 JOG 方式小回转”的下列数组：

- 名称：TRAORI（可通过撤消选择或者校准的回转数据组选择）
- 新的回转平面 / 添加 → 新的零点平面
- 回转模式：轴方式（不可选）
- 围绕 Z 轴旋转
- 围绕 Y 轴旋转
- 围绕X 轴旋转



### 对刀具轴空运行和回转轴运行的说明

可在回转轴回转之前空运行刀具轴（例如，在 G17=Z 时）。

通过用于 JOG 方式（MD 32010 \$MA\_JOG\_VELO\_RAPID[AX]）的常规快速移动进行定位。

如果要定位回转轴，同样用 G0 和用于 JOG 方式（MD 32010 \$MA\_JOG\_VELO\_RAPID[AX]）的常规快速移动运行。仅在循环 TOOLCARR.SPF

中进行运行运动，并可由机床制造商对该运动进行匹配。

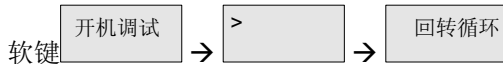
如果用激活的 TRAORI 运行回转平面，则同样在 Zyklus TOOLCARR.SPF

中进行运行运动，此时，通过相应的方向矢量将刀具方向定位为垂直于回转平面（例如，在 G17 带 C3=1 时）。

### 3.16.7 回转循环CYCLE800 开机调试



进入



仅在 CYCLE800 IBN 菜单 激活时才出现。



在对 CYCLE800 进行开机调试时，在刀具数据 \$TC\_CARR1... 中设置数据（回转数据组）。它们综合在回转的开机调试菜单中。  
→ 操作区“开机调试”；软键“回转循环”



**文献：**有关当前信息，请参见：

- 供货软件（标准循环）的文件“siemensd.txt”，或者
- 在 HMI Advanced（高级）配置中 F:\dh\cst.dir\HLP.dir\siemensd.txt
- 功能描述 840D/840Di/810D  
/W1/ „刀具补偿 (3/2 轴斜置加工)

必须已经装载以下的循环：

- CYCLE800.SPF、CYCPE\_SC.SPF（标准循环）
- TOOLCARR.SPF（标准循环）
- PROG\_EVENT.SPF（制造商循环）<sup>1)</sup>

GUD 变量 \_TC\_FR 到 \_TC\_NUM（GUD7）必须已经激活。

#### 激活 CYCLE800 IBN 菜单

在IBN操作区屏幕中必须按如下激活：

- HMI Embedded（内置）  
必须在文件 COMMON.COM 中激活软键“回转循环”。为此，必须打开文件 COMMON.COM 并删除 SC616 前的“;”。  
文件在目录“用户循环”中。随后，要求进行一次 NC 复位。
- HMI Advanced（高级）：  
在文件 STARTUP.COM（标准循环）中删除下面各行前的“;”。  

```

;HS14=($82084,ac7)
;PRESS(HS14)
;LM("SCHWENK1a","SCHWENK1.com")
;END_PRESS

```

进行这些变更后，必须重新启动 HMI Advanced。



#### 1) 对循环 PROG\_EVENT.SPF 的说明

##### 循环 PROG\_EVENT.SPF

作为制造商循环提供，用于在程序段查找之后预定位回转轴。机床制造商可以通过插入各自的制造商循环对PROG\_EVENT 功能进行扩展。应当将 CYCPE\_MA.SPF 用作制造商循环的名称。在 PROG\_EVENT.SPF 中为制造商循环 CYCPE\_MA.SPF 准备一个相应的跳出标记。



### 激活“在JOG方式下回转”功能（自循环 SW 6.5 起）

进入软键在机床/ JOG运行方式下为水平软键“在JOG方式下回转”。

在位于目录“标准循环”下的文件 MA\_JOE.COM 中激活软键 8。

在文件 MA\_JOE.COM 中删除下列行之前的“;”。

```
;HS8=($82119,,se1)
;PRESS(HS8)
;LM("SCHWENK3","jog_c800.com")
;END_PRESS
```

必须在复位时激活工件基准（WPFRAME），以在回转后能够在 JOG 方式下用有效的 NV 进行碰撞接触或者测量（串联测量）。

为此设定 MD24006 \$MC\_CHSFRAME\_RESET\_MASK BIT 4 = 1。



### 机床参数

为了应用回转，必须至少把以下的机床数据按照如下方式进行设定：

- 机床数据带精确赋值（G）->不允许修改的机床数据
- 机床数据带可变的赋值（V）->标准值可以设定到较高或者较低数值的机床数据

MD号	MD名称	值	注释	可改变
10602	\$MN_FRAME_GEOAX_CHANGE_MODE	1	<sup>1)</sup>	V
11450	\$MN_SEARCH_RUN_MODE	位 1=1	在程序段查找之激活 PROG_EVENT	G
18088	\$MN_MM_NUM_TOOL_CARRIER	n>0	n → 回转数据组 <sup>1)</sup> 的个数	G
20110	\$MC_RESET_MODE_MASK	'H4041'	位14=1	G
20112	\$MC_START_MODE_MASK	'H400'	-	G
21100	\$MC_ORIENTATION_IS_EULER	0	旋转角作为RPY说明	G
21110	\$MC_X_AXIS_IN_OLD_X_Z_PLANE	1		G
20126	\$MC_TOOL_CARRIER_RESET_VALUE	0...n	在CYCLE800中说明	V
20150	\$MC_GCODE_RESET_VALUES[41]	1	TCOABS <sup>1)</sup>	G
20150	\$MC_GCODE_RESET_VALUES[51]	2	PAROT <sup>1)</sup>	G
20150	\$MC_GCODE_RESET_VALUES[52]	1	TOROTOF <sup>1)</sup> (仅在运动类型 T 和 M 时)	V
20152	\$MC_GCODE_RESET_MODE[41]	0	(标准) <sup>1)</sup>	G
20152	\$MC_GCODE_RESET_MODE[51]	0	(标准) <sup>1)</sup>	V
20152	\$MC_GCODE_RESET_MODE[52]	0	(标准) <sup>1)</sup>	V
20180	\$MC_TOCARR_ROT_ANGLE_INCR[0]	0	(标准) <sup>1)</sup>	G
20180	\$MC_TOCARR_ROT_ANGLE_INCR[1]	0	(标准) <sup>1)</sup>	G
20182	\$MC_TOCARR_ROT_ANGLE_OFFSET[0]	0	(标准) <sup>1)</sup>	G
20182	\$MC_TOCARR_ROT_ANGLE_OFFSET[1]	0	(标准) <sup>1)</sup>	G
20184	\$MC_TOCARR_BASE_FRAME_NUMBER	-1	(标准) <sup>1)</sup>	G
22530	\$MC_TOCARR_CHANGE_M_CODE	0	<sup>1)</sup>	V
24006	\$MC_CHSFRAME_RESET_MASK	位 4=1	当系统框架 \$P_WPFRAME 要求在复位之后保持有效时	V
24007	\$MC_CHSFRAME_RESET_CLEAR_MASK	位2,3=0		G
24008	\$MC_CHSFRAME_POWERON_MASK	位4, 3, 2=1	当要在上电时删除系统框架 \$P_WPFRAME、\$P_TOOLFRAM E、\$P_PARTFRAME	V
28082	\$MC_MM_SYSTEM_FRAME_MASK	位4, 3, 2=1	调试系统框架 \$P_WPFRAME、\$P_TOOLFRAM E、\$P_PARTFRAME	G



1) 有关机床数据的说明参见下面几页。



MD号	MD名称	值	注释	可改变
28083	\$MC_MM_SYSTEM_DATAFRAME_MASK	位4, 3, 2=1	标准='H7f	G
30455	MISC_FUNCTION_MASK	位2, 0=1	用于回转轴, 它们作为取模轴 <sup>1)</sup>	V
32010	\$MA_JOG_VELO_RAPID[AX]		在 BA JOG 方式下快速移动, 用于在 JOG 方式下回转, 自循环 SW 6.5起 MD 用于空运行的轴和相关的回转轴	V



#### 设定数据

SD号	SD名称	值	注释	可改变
42980	\$SC_TOFRAME_MODE	2000	参见说明	V

所要求的机床数据的改变会影响缓冲存储器的重新组织（数据丢失！）  
要求在设置MD之后和在NCK复位之前进行串行调试。



文献： /IAM/, 安装调试说明 HMI



对机床和设定数据的说明, 请见下页

#### 对 MD 10602 的说明:

在TRAORI之后不要求重新编程NV（零点偏移），比如在跟踪刀具时。

#### 对 MD 18088 的说明:

如果在NCU中有几个通道, 则在考虑 MD 28085: MM\_LINK\_TOA\_UNIT 的情况下划分回转数据组的个数。

举例:

MD 18088 MM\_NUM\_TOOL\_CARRIER =4

通道个数=2。

每个通道可以使用2个回转数据组。

#### 对 MD 20180/MD 20182 的说明:

回转轴带切端面齿时, 在IBN菜单CYCLE800 中登记相应的值。

**对 MD 22530 的说明:**

如果每个通道有几个回转数据组，并且在更回转头或者回转台时要求释放机床功能，则可以在PLC程序中在回转数据组更换时触发一个M指令。

举例：通道1中回转数据组的个数=2

MD 22530: TOCARR\_CHANGE\_M\_CODE = -800

编程回转数据组1(TCARR=1) → M801

编程回转数据组2(TCARR=2) → M802

输出M指令PLC可以限制或者转换主轴转速。

**对 MD 30455 的说明**

回转轴在 G90 时以 DC（最短行程）运行；参见用户循环 TOOLCARR. SPF。

**对 MD 20150/MD 20152 和 SD 42980 的说明:**

对带切端面齿运动（回转头/混合运动）的说明:

与实际有效的工作平面（G17,G18,G19）相关，为了计算CYCLE800 切端面齿时补偿框架（在NCU中），编程指令 TOROT（或者TOROTX、TOROTY）（G 组53）。

如果切端面齿时编程的旋转与回转轴可能的位置相偏离，则在回转头运动中和混合运动中产生一个补偿框架 \$P\_TOOLFRAME（刀具框架）（参见 HMI 有效的零偏/刀具基准）。

计算与设定数据 SD 42980: \$SC\_TOFRAME\_MODE 和机床数据 MD 21110:

\$MC\_X\_AXIS\_IN\_OLD\_X\_Z\_PLANE=1（标准）相关。

在运动关系中（回转头/混合运动）用切端面齿设定数据SD 42980:

\$SC\_TOFRAME\_MODE=2000进行调节！

如果复位或程序结束后，补偿框架保持不变（TOOLFRAME），则在MD 20150:

\$MC\_GCODE\_RESET\_VALUES[52]中必须登记以下的数值:

- 在G17时(TOROT) → MD 20150=2
- 在G18时(TOROTY) → MD 20150=3
- 在G19时(TOROTX) → MD 20150=4

通过以下的菜单屏幕进行CYCLE800 开机调试:

### 运动链的开机调试

Kinematic channel1			
Kinematics	Inclinable head	Name:	HEAD_1 No.: 1
Retract:	X	Y	Z
Retract position			200.000
Offset vector I1	0.000	0.030	-63.000
Rotary axis vector V1	0.000	0.000	1.000
Offset vector I2	0.000	0.000	40.000
Rotary axis vector V2	1.000	0.000	0.000
Offset vector I3	0.000	-0.030	23.000
Display opt.			
Swivel mode:	axial +projection angle		
Direction:	Rot. axis 1		
Tracking TL	No		

必须给每个回转头、回转台或者每个回转头/回转台的组合编制一个回转数据组。

回转数据组可以在几个通道中协调一致。

回转数据组的个数通过以下的机床数据限制:

- MD 18088:MM\_NUM\_TOOL\_CARRIER 或者
- MD: NUM\_CHANNELS (通道个数选项)
- MD 28085: MM\_LINK\_TOA\_UNIT

回转数据组用参数 \$TC\_CARR1[n] 至 \$TC\_CARR40[n] 分配刀具数据, 或者\$TC\_CARR65[n] 到 NCU SW 7.1。

在“运动”屏幕窗口中, 参数具有以下的含义:

名称: 回转数据组 \$TC\_CARR34[n]

n → 回转数据组编号

如果每个 NC 通道约定有多个回转数据组, 则为每个回转数据组分配一个名称。

如果不能够更换可回转的刀架(每个通道的回转数据组), 则不必给定名称。

用软键(通道 +/- 回转数据组 +/-) 继续切换至下一个回转数据组和下一个通道。



回转数据组名称仅允许包含允许的 NC 编程字符(A...Z、0...9)。

### 运动类型 \$TC\_CARR23[n]

- 回转头(类型T)
- 回转台(类型P)
- 回转头+回转台(类型M)

### 空运行/空运行位置

`$TC_CARR38[n] X; $TC_CARR39[n] Y; $TC_CARR40[n] Z`

`n` → 回转数据组编号

开机调试确定：是否可以在回转循环的输入屏幕选择 Z 轴的空运行和 Z,X,Y 的空运行以及在刀具方向上的空运行（自循环 SW 6.5起）。

如果要修改空运行方式，则在用户循环 `TOOLCARR.SPF` 中进行修改（参见章节 3.16.8）。

如果用户循环 `TOOLCARR.SPF` 未修改，则进行空运行作为绝对机床位置。



在运行工具轴时请注意以下内容：

空运行工具轴，使回转时在刀具和工件之间不会发生冲突。

### 偏移—回转轴矢量（机床运动）

`$TC_CARR1[n] ... $TC_CARR20[n]`

这些矢量始终包含3个部分，它们表示机床轴（X、Y、Z）的基准。

运动链的位置由机床制造商测量，它们对回转头/回转台（回转数据组）非常重要。

偏移矢量 I1 到 I4 取决于**回转轴不回转的状态**（机床运动初始位置）。

不必完全实现所使用的机床运动。必须注意的是，可以限定回转平面中的运行区域。如果要求一个机床运动仅带一个回转轴，则这必须约定作为第一个回转轴。

可以手调的回转轴（手动模式）可以带/不带测量系统，在“简单机床”中可以使用。

存储数据组

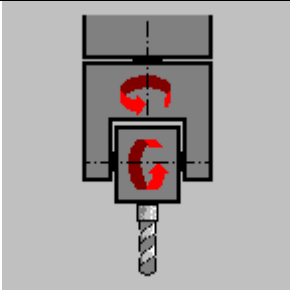
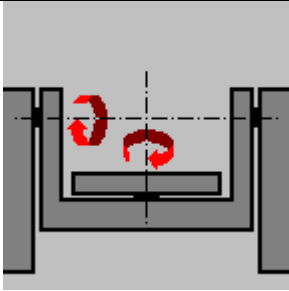
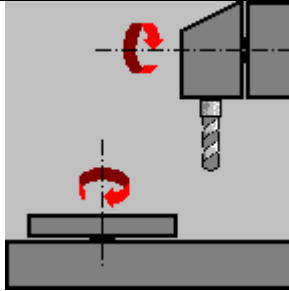
软键

将当前回转数据组做为零件程序存储。零件程序取决于回转数据组的名称。

删除数据组

软键

删除当前回转数据组。

回转头 (类型T)	回转台 (类型P)	回转头+回转台 (类型M)
		
偏移矢量I1	偏移矢量I2	偏移矢量I1
回转轴矢量V1	回转轴矢量V1	回转轴矢量V1
偏移矢量I2	偏移矢量I3	偏移矢量I2
回转轴矢量V2	回转轴矢量V2	偏移矢量I3
偏移矢量I3	偏移矢量I4	回转轴矢量V2
		偏移矢量I4

矢量总是取决于机床运动的初始位置。

#### 文献:

- 功能描述 840D/840Di/810D  
- /W1/ „刀具补偿 - (3/2 轴斜置加工)
- ISO 841-2001 或者 DIN 66217

#### 回转头

- I3 从刀具托架到第二个回转轴的旋转中心/交点的距离
- I2 从第二个回转轴的旋转中心/交点到第一个回转轴的旋转中心/交点的距离
- I1 当回转头不能更换时, 结束矢量链  $I1 = -(I2 + I3)$

#### 回转台

- I2 从机床基准点到第一个回转轴的旋转中心/交点的距离
- I3 从第一个回转轴的旋转中心/交点到第二个回转轴的旋转中心/交点的距离 (或者到刀具托架基准点)
- I4 当回转台不能更换时, 结束矢量链  $I4 = -(I2 + I3)$

#### 回转头/回转台 (混合运动)

- I2 从刀具托架到第二个回转轴的旋转中心/交点的距离
- I1 当回转头不能更换时, 结束矢量链  $I1 = -I2$
- I3 从机床基准点到第二个回转轴的旋转中心/交点的距离 (或者到刀具托架基准点)
- I4 当回转台不能更换时, 结束矢量链  $I4 = -I3$

不必强制这些偏移矢量指向回转轴的旋转中心。重要的是, 这些矢量指向旋转方向的一个点 (交点)。

偏移矢量和回转轴矢量的符号由轴方向的确定按照ISO841-2001或者DIN66217 (右手规则) 产生。



### 数据 TOOLCARRIER 关系?5 轴转换



自 NCU SW 7.1 起, 可用于转换类型 72 (\$MC\_TRAFO\_TYPE\_1) 的 5 轴转换。  
对于转换类型 72, 使用刀架 (TOOLCARRIER) 矢量。

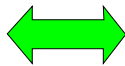
#### 回转头 (可回转刀具)

\$TC\_CARR23[1]="T"

I1 \$TC\_CARR1...3[n]

I2 \$TC\_CARR4...6[n]

I3 \$TC\_CARR15...17[n]



\$MC\_TRAFO\_TYPE\_1=24

\$MC\_TRAFO5\_PART\_OFFSET\_1[0...2]

\$MC\_TRAFO5\_JOINT\_OFFSET\_1[0...2]

\$MC\_TRAFO5\_BASE\_TOOL\_1 [0...2]

矢量链结束 I1=- (I2+I3) <sup>1)</sup>

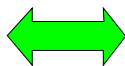
#### 回转台 (可回转工件)

\$TC\_CARR23[1]="P"

I2 \$TC\_CARR4...6[n]

I3 \$TC\_CARR15...17[n]

I4 \$TC\_CARR18...20[n]



\$MC\_TRAFO\_TYPE\_1=40

\$MC\_TRAFO5\_BASE\_TOOL\_1 [0..2]

\$MC\_TRAFO5\_JOINT\_OFFSET\_1 [0...2]

\$MC\_TRAFO5\_PART\_OFFSET\_1 [0...2]

矢量链结束 I4=- (I2+I3)

#### 回转头+回转台 (可回转刀具+工件)

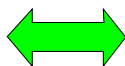
\$TC\_CARR23[1]="M"

I1 \$TC\_CARR1...3[n]

I2 \$TC\_CARR4...6[n]

I3 \$TC\_CARR15...17[n]

I4 \$TC\_CARR18...20[n]



\$MC\_TRAFO\_TYPE\_1=56

\$MC\_TRAFO5\_JOINT\_OFFSET\_1 [0...2]

\$MC\_TRAFO5\_BASE\_TOOL\_1 [0...2]

\$MC\_

TRAF05\_JOINT\_OFFSET\_PART\_1[0...2]

\$MC\_TRAFO5\_PART\_OFFSET\_1 [0...2]

结束矢量链 I1=-I2 I4=-I3



1) 对于固定安装的机床运动。

## 显示选项

- **\$TC\_CARR37[n]** (n → 回转数据组)

如果没有设置相应的显示选项，则在输入屏幕窗口中该值也不显示（参见章节3.16.1）。

8	7	6	5	4	3	2	1	0	(十进制位)
								0	轴方式
								1	轴方式+投影角
								2	轴方式+投影角+立体角
									回转轴1
								0	自动
								1	手动
									回转轴2
								0	自动
								1	手动
									选择轴的优选方向
								0	否
								1	以回转轴1为基准
								2	以回转轴2为基准
									跟踪刀尖
								0	否
								1	是
									预留
									空运行模式
								0	Z轴
								1	Z轴或ZX Y轴
									在刀具方向上空运行（自循环 SW6.5起） <sup>3)</sup>
									(左边) (右边)
								Z	最大刀具方向
								Z、X、Y	增量刀具方向
								Z+ Z、X、Y	最大+增量刀具方向
									回转数据组/刀具更换 <sup>1)</sup>
								0	否 2)
								1	手动 自动
								2	自动 自动
								3	否 2)
								4	手动 手动
								5	自动 手动

1) 仅对于 ShopMill/ShopTurn 相关。

2) 如果为约定回转数据组，则自动/手动刀具更换设定不相关。

3) 有关编码空运行模式的信息，参见表格

编码空运行模式 Z、ZXY，最大和增量刀具方向

Z	Z, X, Y	最大刀具方向	增量刀具方向	以 \$TC_CARR37 xXXxxxxxx 编码
1	0	0	0	00
1	1	0	0	01
0	1	0	0	02 <sup>1)</sup>
0	0	0	0	03
0	0	1	0	04
1	0	1	0	05
0	1	1	0	06
1	1	1	0	07
0	0	0	1	08
1	0	0	1	09
0	1	0	1	10
1	1	0	1	11
0	0	1	1	12
1	0	1	1	13
0	1	1	1	14
1	1	1	1	15

1) 自循环 SW 6.5 起编码 02...15 适用。

下面的显示选项对回转循环的输入屏幕窗口有影响：

- 回转模式
  - 轴方式
  - 轴方式和投影角
  - 轴方式和投影角和立体角

举例：

回转模式开机调试选择：轴方式，投影角

随后在输入菜单中仅以轴方式或投影角提供。

在这种情况下不要求编程立体角，也不可能编程立体角。

- 方向
  - 回转轴 1
  - 回转轴 2
  - 否

在用于回转轴循环的输入屏幕中选择运行方向时，以**回转轴 1** 或 **2** 为基准。

通过机床运动**回转轴**的角度范围，由 NCU 计算出两个可能的解决方案。

此时通常在工艺上要求一个解决方案。

- 在开机调试菜单中选择，这两个解决方案要以哪个**回转轴**为基准。
- 在回转循环的输入屏幕中选择，要运行这两个可能的解决方案中的哪一个。
- 如果为“否”，则在操作屏幕窗口中不显示方向参数。



- 跟踪 WZ (刀具)

→ 否

→ 是

在回转循环输入屏幕中显示“跟踪 WZ”。

功能跟踪 WZ 以选项 5 轴转换 (TRAORI) 为前提条件。



参见编程示例 TOOLCARR. SPF。

## 回转轴参数开机调试

Kinematic channel1			
Kinematics	Inclinable head	Name:	HEAD_2 No.: 2
Retract:	X	Y	Z
Retract position			200.000
Offset vector I1	0.000	-172.000	-265.800
Rotary axis vector V1	0.000	1.000	0.000
Offset vector I2	0.000	172.000	172.000
Rotary axis vector V2	0.000	-1.000	1.000
Offset vector I3	0.000	0.000	93.800
Display opt.			
Swivel mode:	axial + projection angle		
Direction:	Rot. axis 1		
Tracking TL	No		



回转轴1和2的数据输入，这对回转循环很重要。

名称/运动 → 参见开机调试菜单 CYCLE800 “运动”。

### 回转轴轴名称

\$TC\_CARR35[n] 回转轴 1

\$TC\_CARR36[n] 回转轴 2

优先选择以下的名称:

围绕机床轴X旋转的轴--> A

围绕机床轴Y旋转的轴--> B

围绕机床轴Z旋转的轴--> C

如果这些轴已经为 NCU 所知，则必须选择 NC 回转轴相同的轴名称 (参见自动方式)。

如果这些轴不为 NCU 所知，则可以使用任意的轴名称 (最大 6 个字母或者数字)。

### 模式

\$TC\_CARR37[n] 参见显示选项

- 自动方式

NC回转轴自动地运行到相应的回转角。

- 手动方式

操作人员以手动方式将**回转轴**调整到相应的位置上，例如，带有可手调**回转轴**的“简单机床”（测量系统：钢尺）。

通过信息 62180/62181 显示待调节的回转角度。

- 半自动

在“半自动”方式下，信息62180/62181（显示待调节的角度值）不显示。

应用：

由相应的机械师定位到计算好的角度的手动回转轴。机床制造商可为此相应修改用户循环

TOOLCARR，将回转角度（局部变量 `_A1`、`_A2`）传送到一个子程序中

（标记 `_M21...` 用于手动回转轴）。



允许有混合机床运动（例如1个自动回转轴，2个手动回转轴）和“不完全的”机床运动（例如1个围绕 X 轴旋转的回转轴）。

如果要求一个机床运动仅带一个回转轴，则这必须约定作为第一个回转轴。



回转角显示参见显示信息

CYCLE800 → 62180/62181。

### 角度范围

\$TC\_CARR30[n] .. \$TC\_CARR33[n]

必须为每个**回转轴**分配一个有效的角度范围。这个角度范围不必是与回转轴的软件限位范围相同。

在取模轴中，必须使运行区在 **0 到 360 度**之间。

### 运动偏移（自循环 SW 6.3起）

当运动的基本设定中回转轴的位置不为0时，则在输入屏幕窗口“偏移运动”中可以输入回转轴

1(\$TC\_CARR24[n]) 回转回转轴2 (\$TC\_CARR25[n])的偏移值。

切端面齿

**\$TC\_CARR26[n]...\$TC\_CARR29[n]**

否

下面的数组不显示。

是

→ 在开始切齿时有端面齿的**角偏移**。

→ 切端面齿的**角度刻度**

→ **自动补偿 是/否** (自软件版本 SW 6.3 起取消)

**回转数据组更换** (仅在ShopMill/ShopTurn中重要)

- 否
- 自动方式
- 手动方式

**刀具更换** (仅在ShopMill/ShopTurn中重要)

- 自动方式
- 手动方式

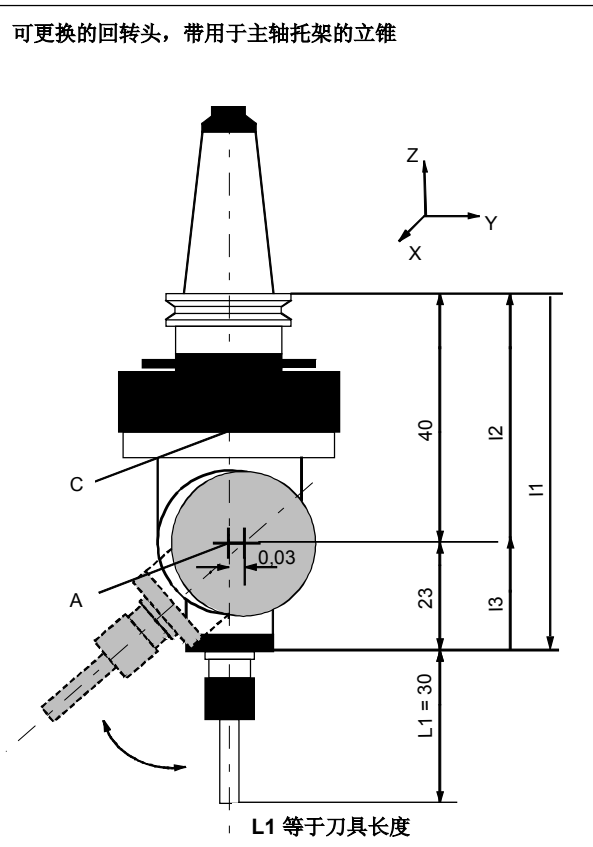
仅在运动类型 T 和 M 中显示“刀具更换”。



### 机床运动中开机调试示例

#### 举例1：回转头 1 “HEAD\_1”

- 回转轴 1 (C) (手动)，围绕 Z 轴旋转
  - 回转轴 2 (A) (手动)，围绕 X 轴旋转
  - 可手调并可更换的回转头 (手动)
- 矢量取决于运动的初始位置  
(图纸不按照比例)



#### SK 回转开机调试，运动

运动	回转头		HEAD_1
空运行	Z		
	X	Y	Z
			200.000
偏移矢量I1	0.000	0.030	-63.000
回转轴矢量 V1	0.000	0.000	1.000
偏移矢量I2	0.000	0.000	40.000
回转轴矢量V2:	1.000	0.000	0.000
偏移矢量I3	0.000	-0.030	23.000
显示选项			
回转模式	轴方式		
方向	回转轴 2		
跟踪刀具	否		

#### 回转轴

回转轴 1	C	模式	手动方式
角度范围	0.000		360.000
回转轴 2	A	模式	手动方式
角度范围	-15.000		100.000



### 举例2: 回转头 2 “HEAD\_2”

回转轴矢量V1: 回转轴B围绕Y

回转轴矢量V2: 回转轴C围绕Y和Z旋转

偏移矢量I1: 在固定安装回转头I1=-(I2+I3)时结束矢量,

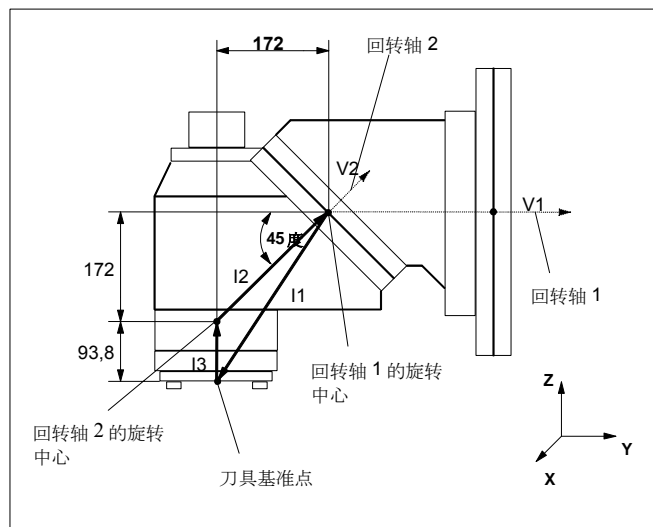
偏移矢量I2: 回转轴 1 的旋转中心和回转轴 2 的旋转中心之间的距离

偏移矢量 I3: 刀具基准点和回转轴 2 的旋转中心之间的距离

方向回转头 (可手调), 带切端面齿

#### 矢量取决于运动的初始位置

如果已固定安装回转头, 则结束矢量 (参见 I1)。



### SK 回转开机调试, 运动

运动	回转头		HEAD_2
空运行	Z	刀具方向	最大+增量
	X	Y	Z
			200.000
偏移矢量I1	0.000	-172.000	-265.800
回转轴矢量 V1	0.000	1.000	0.000
偏移矢量I2	0.000	172.000	172.000
回转轴矢量V2:	0.000	1.000 <sup>1)</sup>	1.000 <sup>1)</sup>
偏移矢量I3	0.000	0.000	93.800
显示选项			
回转模式	轴方式		
方向	回转轴 2		
跟踪刀具	否		

#### 回转轴

回转轴 1	B	模式	手动方式
角度范围	0.000		360.000
运动偏移值	0.000		
切端面齿	是	角度刻度	1.000

回转轴 2	C	模式	手动方式
角度范围	0.000		180.000
运动偏移值	0.000		
切端面齿	是	角度刻度	1.000



回转轴1和2的旋转中心基准点可以移动到旋转线，不必与机械旋转中心一致。

1) 计算回转轴矢量 V2: 45 度角

$$V2_y = \sin(45) = 0.7071$$

$$V2_z = \cos(45) = 0.7071$$

V2<sub>y</sub> 和 V2<sub>z</sub> 可标准化为 1。



### 举例3: 万向工作台“TABLE\_45”

矢量取决于运动的初始位置

回转轴矢量V1: 回转轴C围绕Y和Z旋转

回转轴矢量V2: 回转轴C围绕Z旋转

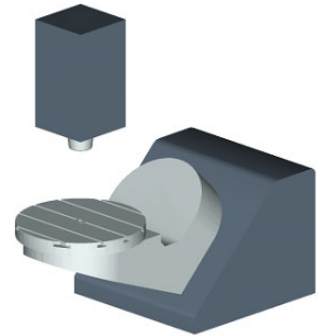
偏移矢量I2: 从机床基准点到回转轴 1

的旋转中心/交点的距离

偏移矢量 I3: 从回转轴 1 的旋转中心/交点到回转轴2

的旋转中心/交点的距离

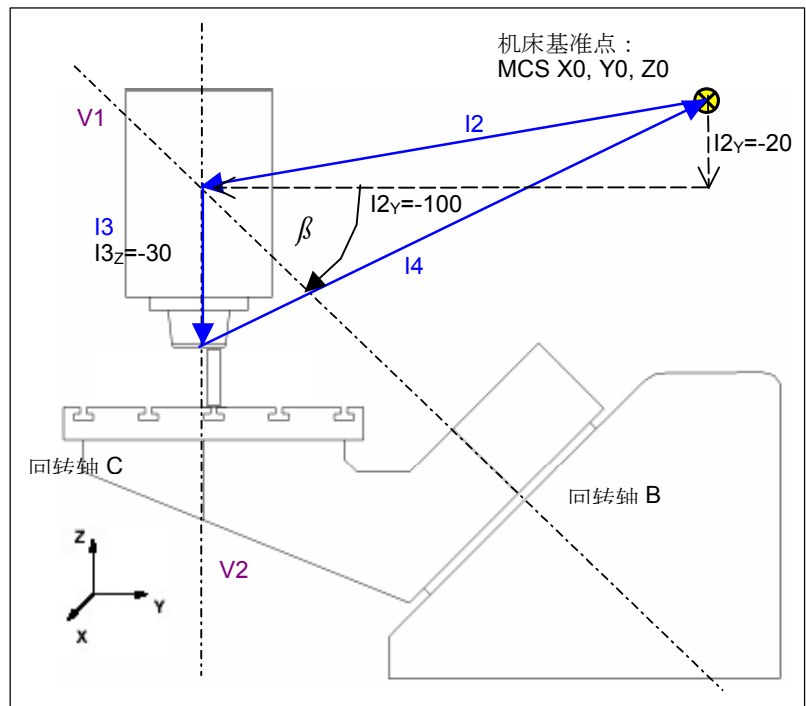
偏移矢量 I4: 结束矢量链 I4=-(I2+I3)



#### 机床侧视图:

通过工作台上边缘或者工作台中心将主轴（刀具托架）定位到定位块上。

借助主轴上的测量杆确定回转台的旋转中心。



## SK 回转开机调试, 运动

运动	回转台		TABLE_45
	X	Y	Z
偏移矢量I2	0.000	-100.000	-20.000
回转轴矢量 V1	0.000	-1.000 <sup>1)</sup>	1.000 <sup>1)</sup>
偏移矢量I3	0.000	0.000	-30.000
回转轴矢量V2:	0.000	0.000	-1.000
偏移矢量I4	0.000	100.000	50.000
显示选项			
回转模式	轴方式		
方向	回转轴 2		
跟踪刀具	否		

## 回转轴

回转轴 1	B	模式	自动方式
角度范围	0.000		180.000
回转轴 2	C	模式	自动方式
角度范围	0.000		360.000



1) 计算回转轴矢量 V1:  $\beta = -45$  度

$$V1_y = \sin(-45) = -0.7071$$

$$V1_z = \cos(-45) = 0.7071$$

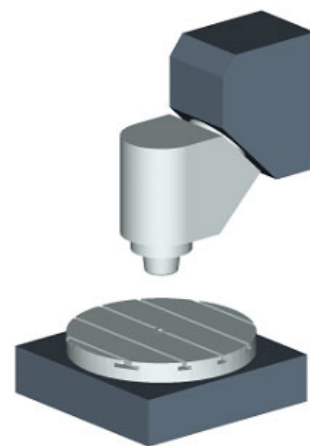
$V1_y$  和  $V1_z$  可以标准化为 -1 和 1。



## 举例4: 回转头/回转台 “MIXED\_45”

矢量取决于运动的初始位置

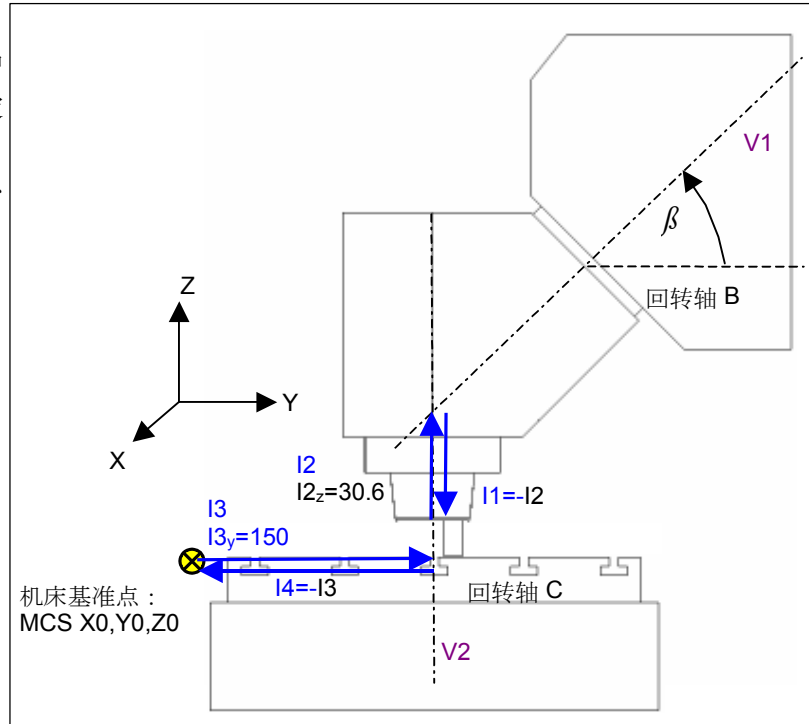
- 回转轴矢量V1: 回转轴 B 围绕 Y 和 Z 旋转  
 回转轴矢量V2: 回转轴 C 围绕 Z 旋转  
 偏移矢量I2: 从刀具托架基准点到回转轴 1 的旋转中心/交点的距离  
 偏移矢量I1: 结束矢量 I1=-I2  
 偏移矢量 I3: 从机床基准点到回转轴 2 的旋转中心/交点的距离  
 偏移矢量I4: 结束矢量 I4=-I2



**机床侧视图:**

通过工作台上边缘或者工作台中心将主轴（刀具托架）定位到定位块上。

借助主轴上的测量杆确定回转台的旋转中心。

**SK 回转开机调试，运动**

运动	回转台		MIXED_45
	X	Y	Z
偏移矢量I1	0.000	0.000	-30.600
回转轴矢量 V1	0.000	1.000 <sup>1)</sup>	1.000 <sup>1)</sup>
偏移矢量I2	0.000	0.000	30.600
偏移矢量I3	300.000	150.000	0.000
回转轴矢量V2:	0.000	0.000	-1.000
偏移矢量I4	-300.000	-150.000	0.000
显示选项			
回转模式	轴方式		
方向	回转轴 1		
跟踪刀具	是		

**回转轴**

回转轴 1	B	模式	自动方式
角度范围	0.000		180.000
回转轴2	C	模式	自动方式
角度范围	0.000		360.000

1) 计算回转轴矢量 V1:  $\beta = 45$  度

$$V_{V1y} = \sin(45) = 0.7071$$

$$V_{V1z} = \cos(45) = 0.7071$$

$V_{V1y}$  和  $V_{V1z}$  可标准化为 1。



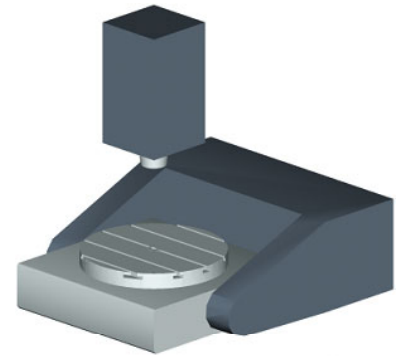




### 举例5: 回转台“TABLE\_5”

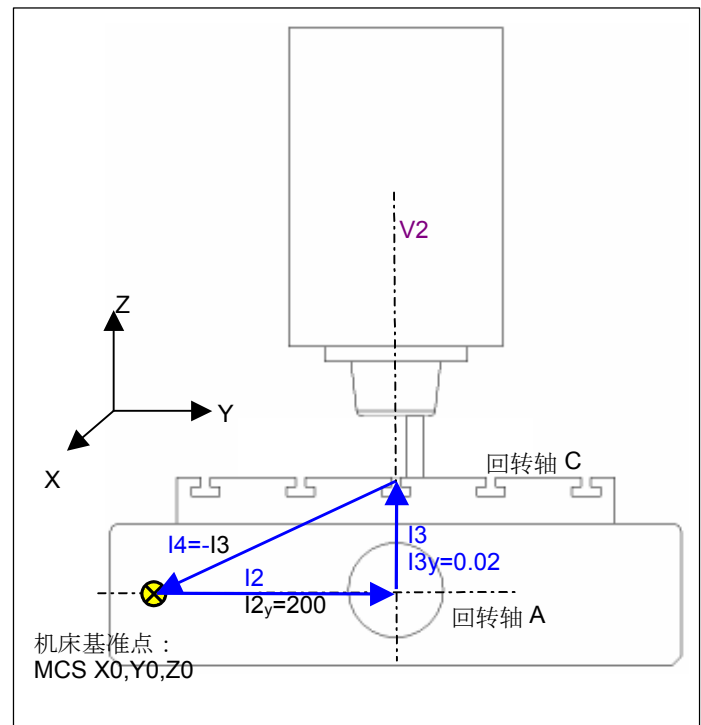
#### 矢量取决于运动初始位置

- 回转轴矢量V1: 回转轴 A 围绕 X 旋转  
 回转轴矢量V2: 回转轴 C 围绕 Z 旋转  
 偏移矢量I2: 从机床基准点到回转轴 1 的旋转中心/交点的距离  
 偏移矢量 I3: 从回转轴 1 的旋转中心/交点到回转轴2 的旋转中心/交点的距离  
 偏移矢量 I4: 结束矢量链  $I4=-(I2+I3)$

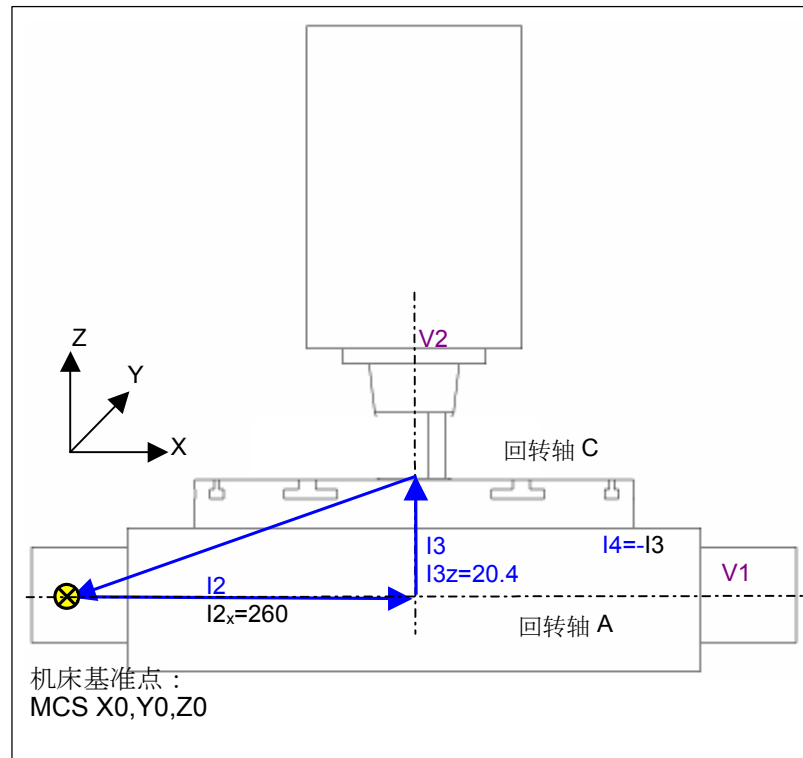


#### 自 X 方向的机床侧视图:

通过工作台上边缘（回转轴C）或者工作台中心将主轴（刀具托架）定位到定位块上。  
 借助主轴上的测量杆确定回转台的旋转中心。



自 Y 方向的机床前视图



## SK 回转开机调试，运动

运动	回转台		TABLE_5
	X	Y	Z
偏移矢量I2	260.000	200.000	0.000
回转轴矢量 V1	-1.000	0.000	0.000
偏移矢量I3	0.000	0.020	20.400
回转轴矢量V2:	0.000	0.000	-1.000
偏移矢量I4	-260.000	-200.020	-20.400
显示选项			
回转模式	轴方式		
方向	回转轴 1		
跟踪刀具	否		

## 回转轴

回转轴 1	A	模式	自动方式
角度范围	-90.000		90.000
回转轴 2	C	模式	auto
角度范围	0.000		360.000

## 3.16.8 用户循环 TOOLCARR.SPF



## 编程

TOOLCARR(\_MODE, \_TC1, \_A1, \_A2, \_TC2)



## 参数

_MODE	与后面所描述的结构标记相符
_TC1	回转头/回转台序号
_A1	第一旋转轴的角度
_A2	第二旋转轴的角度（如果有）
_TC2	在JOG方式下回转时进给率减少（自循环 SW 6.5 起）



## 机床制造商匹配

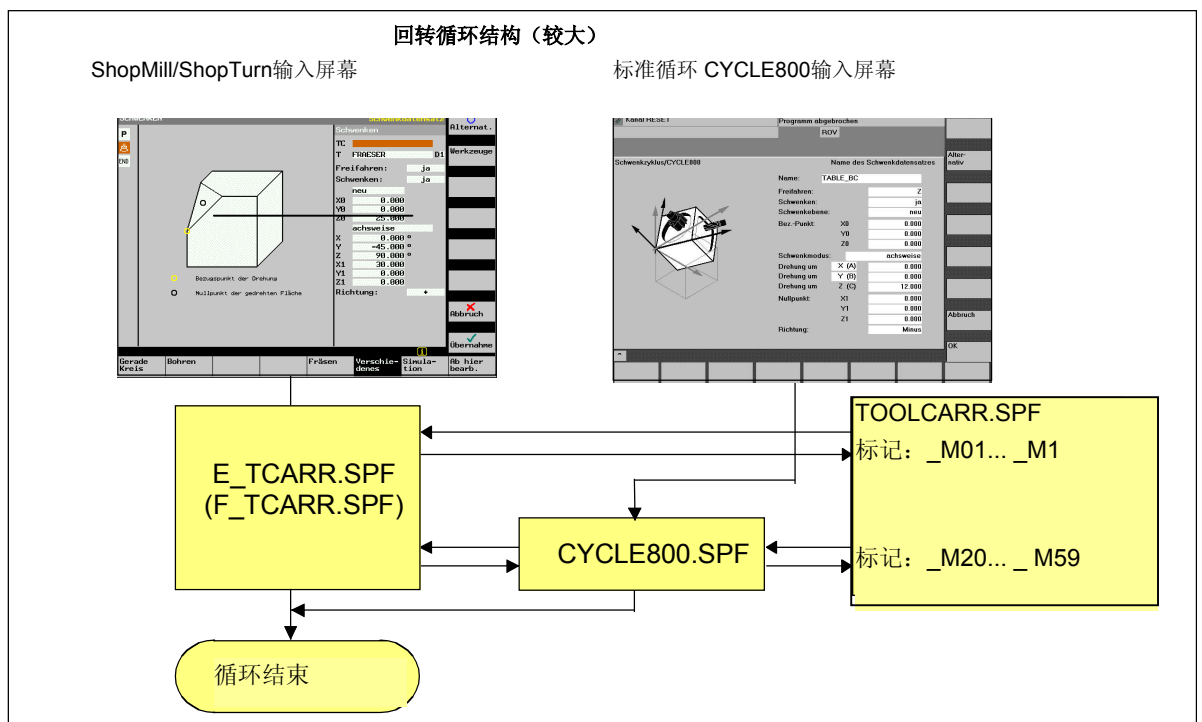
回转时，借助循环 TOOLCARR.SPF 运行所有轴位置。由回转循环 CYCLE800 以及 E\_TCARR（ShopMill）或 TCARR（ShopTurn）进行调用。

可由机床制造商在开机调试时修改循环，以熟悉机床专用的各个要素。

如果用户循环没有修改，则在回转之前空运行时首先运行轴Z（标记\_M41）或者轴Z然后轴X,Y（标记\_M42）。

位置符合CYCLE800 IBN菜单“运动”->空运行位置。

如果是机床 INCH 的基准系统，则相应修改循环 TOOLCARR。

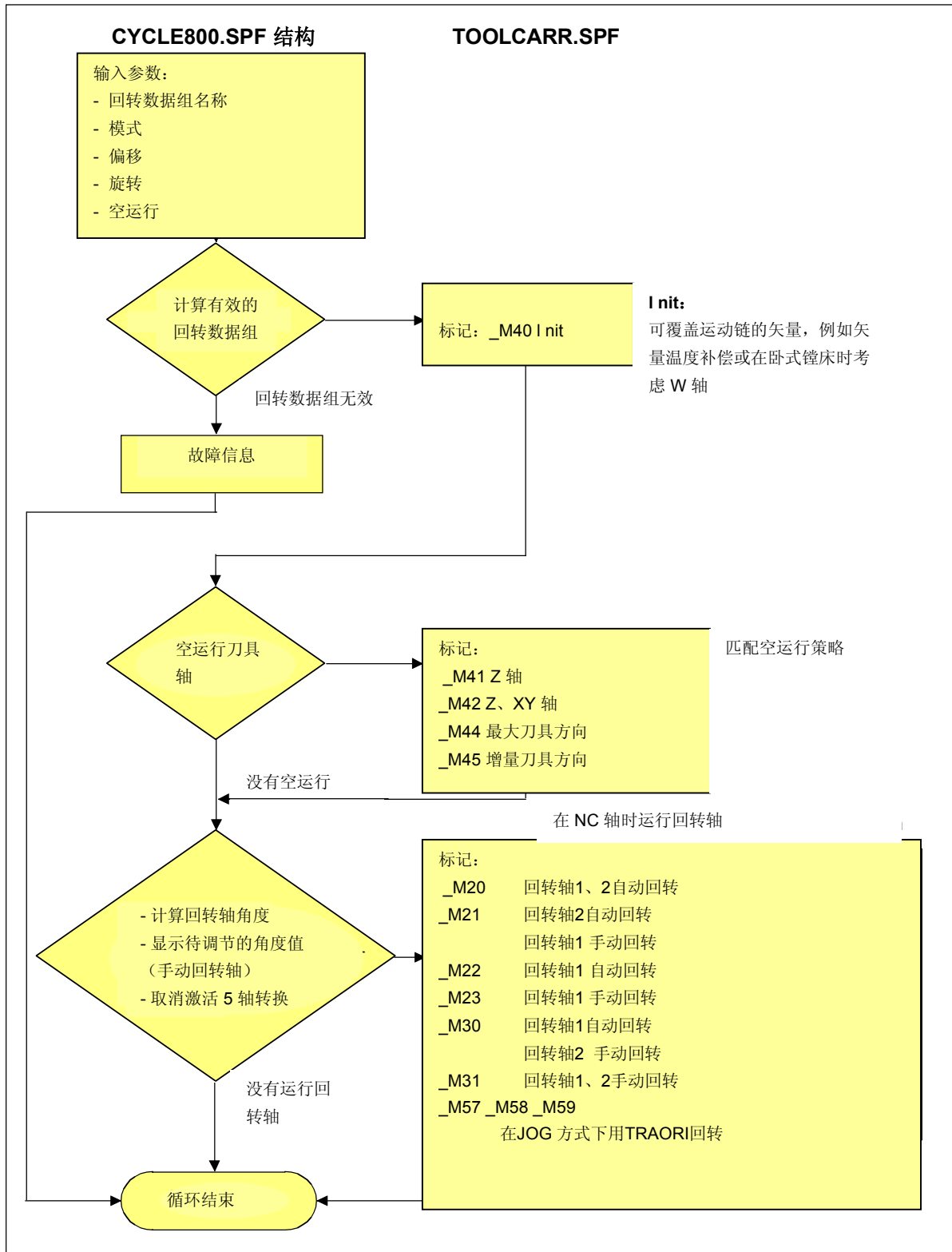


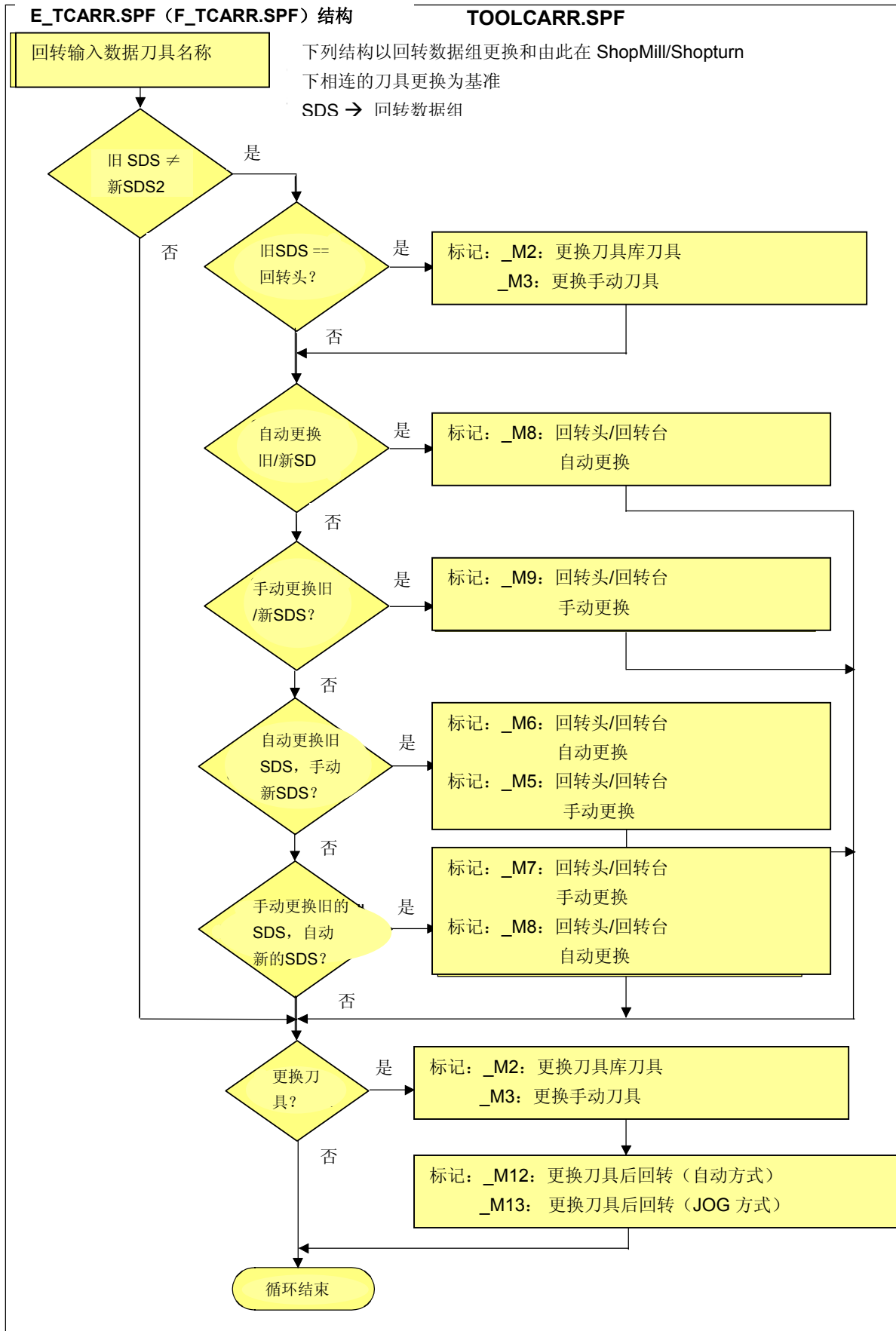
注意机床制造商的说明！



## 说明:

只有在 TOOLCARR 中有相应的跳转标记时, 这些参数才有效 → 参见后面的程序结构。  
如果是机床 INCH 的基准系统, 则相应修改循环 TOOLCARR。







### 对 \_M20 到 \_M37 的说明

标记 \_M20 到 \_M37 通过带两个或一个回转轴的运动相区别。

此外，在自动回转轴（已为 NCU 所知）和手动轴之间进行区别。

总是只有一个标记适用于有效的回转数据组。通过参数/GUD7 变量 \_TC\_ST 检查。



### 对 ShopMill/ShopTurn 的说明

在用户循环 TOOLCARR.SPF 中，在

ShopMill/ShopTurn（参见标记 \_M2 到 \_M9）下必须调用循环 E\_SWIV\_H 或者 F\_SWIV\_H。

参数 E\_SWIV\_H（参数 1、参数 2、参数 3）

- 参数 1: 回转数据组号 (\_TC1)
- 参数 2: 第一回转轴角度
- 参数 3: 第二回转轴角度

修改示例:

如果在回转文件交换/刀具更换时**不要求定位**回转轴（回转头/回转台），则可以在相应的标记处对循环 E\_SWIV\_H 的调用进行注释。

如果要求回转轴运行到指定的位置，则可以传送一个角度值到参数 2 和参数 3。



### 对“跟踪刀具”的说明

跟踪刀具的前提是，已对与相应的回转数据组相等的5轴转换进行调试。

将编程部分安装到制造商循环 TOOLCARR.SPF 的标记 \_M20 中。

通过加注释来激活相应的程序行。

## 3.16.9 故障信息



## 说明

## CYCLE800 报警原因

报警号	报警文本	说明, 纠正
61180	“尽管机床数据\$MN_MM_NUM_TOOL_CARRIER>1, 但是没有给回转数据组分配名称”	虽然有几个回转数据组(\$MN_MM_NUM_TOOL_CARRIER>0), 但是没有约定回转数据组(\$MN_MM_NUM_TOOL_CARRIER=0), 但是回转数据组没有分配名称。
61181	“NCK软件版本太低 (缺少功能TOOLCARRIER)”	功能 TOOLCARRIER 自 NCU 6.3xx 起
61182	“回转数据组名称未知”	参见回转循环CYCLE800开机调试 →运动名称 (回转数据组)
61183	“空运行方式 GUD7_TC_FR 在值范围 0.2 之外”	参见回转循环CYCLE800开机调试 → 空运行; 1. 传送参数 CYCLE800(x,...)有问题 >2
61184	“使用当前的输入角度值不可能解决”	或参数 _MODE 编码错误, 例如轴方式旋转 YXY
61185	“没有约定或者约定了错误 (最小>最大) 的回转轴角度范围”	检查回转循环CYCLE800开机调试
61186	“回转轴矢量无效→ 检查回转循环 CYCLE800 开机调试”	回转循环CYCLE800开机调试: 没有或错误的回转矢量 V1 或 V2 登记
61188	“没有约定第一回转轴的轴名称 →检查回转循环 CYCLE800 开机调试”	回转循环CYCLE800开机调试: 在回转轴 1 名称下没有登记
61189	“无法回转”	G500 有效 检查回转轴的基本 NV
61190	“无法在刀具方向上空运行”	见下页
61196	“在 JOG 方式下没有回转”	同时激活 5 轴转换和 TOOLCARRIER
61199	“不允许设置刀具和回转数据组更换 (TOOLCARRIER)”	
62180	“调整回转轴 x.x [度]”	在手动回转轴中待设定的角度
62181	“调整回转轴 x.x [度]”	在手动回转轴中待设定的角度
62185	“角度匹配到角度刻度上: %4”	切端面齿时为 %4 差角
62186	“在 JOG 方式下没有回转”	有效的 NV G %4 和基准框架包含有效 NV的%4 号旋转
62187	“在 JOG 方式下没有回转”	多个有效的基准框架 (G500) 包含旋转



### 对 61190 的说明

61190 无法在刀具方向上空运行 -->故障代码: %4

故障代码 %4 = abcd

a=0xxx -> 参数 CYCLE800\_FR 错误或者空运行变量在 IBN CYCLE800 中未调试

a=1xxx -> 无现有的应用轴 \$P\_AXN3

a=2xxx -> 最大空运行行程错误, 见 GUD\_TC\_P[8]

a=3xxx -> 增量空运行行程错误, 见 GUD\_TC\_P[8]

b=输入参数 \_FR\*100

cd=IBN 参数 \$P\_TCARR37[] (6.,7 十进制位置)

### 对 62186 和 62187 的说明

激活故障信息 62186 和 62187 的设置, 带 GUD7 参数 \_TC\_FR:

百位 0xx -> 没有故障分析 62186 62187

1xx -> 故障分析 62186 –有效的 NV G%4 和基准框架包含  
旋转

2xx -> 故障分析 62187 – 多个有效的基准框架 (G500) 包含  
旋转

3xx -> 故障分析 62186 和 62187



举例说明在 CYCLE800 中手动回转轴时待设定回转角的显示

---

62181 “设定回转轴 B: 32.5 [度]”

---



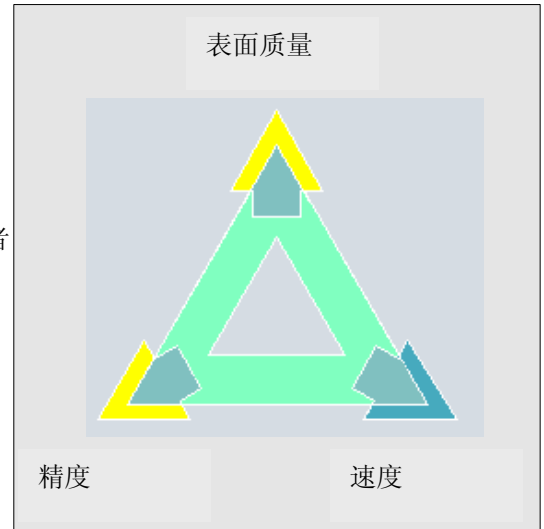
### 3.17 高速设定 — CYCLE832 (自循环 SW 6.3 起)



自软件版本6.3和NCU SW6.3(CCU SW4.3)起可以使用标准循环高速设定CYCLE832，用于HMI。

使用循环CYCLE832:

- 在3轴或者5轴高速加工范围（高速切削—HSC），在加工自由轮廓（表面）时作为工艺辅助
- 在HSC铣削范围优先使用（在车削或者磨削加工时使用方法）
- 综合了HSC加工时要求的主要G代码和机床数据或者设定数据
- 通过相应的NC程序结构分开工艺—几何



在 HSC 范围中执行CAM 程序时必须由控制系统在最短的 NC 程序段中处理高进给率。此时，在最大的加工进给率 >10 m/min 时，在  $\mu\text{m}$  范围中较高精度情况下，用户可获得所要求的较佳的表面质量。通过不同的加工策略，用户可借助 CYCLE832 精确确定程序。

在粗加工时，通过精磨削轮廓，关注的主要是速度。

在精加工时重点则放在精度上。

在两种情况下，通过给出公差保证加工轮廓，从而达到所要求的表面质量。确定公差值以精磨削轮廓时，操作人员必须严格掌握有关下面所述的 CAM 程序的相关知识。循环 CYCLE832

支持机床类型，在这些机床类型中加工时最多分配 3 根线性轴和 2 根回转轴。

通过CYCLE832可以定义以下的功能，或者开关以下的功能：

- 4 种加工方式：精加工，初精整，粗加工，撤消选择（标准设定）
- 待加工轮廓的公差带
- 精磨削（G64、G641、G642）
- NC 程序段压缩器（COMPCAD、COMPCURV、COMPOF）<sup>1)</sup>
- 预控制（FFWON, FFWOF）
- 冲击限制（SOFT、BRISK）
- 5 轴转换（TRAORI、TRAFOF）<sup>1)</sup>
- B 样条<sup>1)</sup>

1) 仅在设置了相应的选项时。



## 功能

循环CYCLE832综合了HSC加工时要求的主要G代码和机床数据或者设定数据。

在CYCLE832中分为四种加工工艺：

- “精加工”
- “初精整”
- “粗加工”
- “撤消选择”（标准设定）

在 CAM 程序中 HSC 范围内，这四种加工方式与精度和轨迹速度直接相关（参见辅助图形）。操作人员/编程人员可通过公差值的设定对此加以重视。

这四种加工方式可以分配不同的公差和设定（配合工艺）。

在输入屏幕窗口已经预设置了相应的G代码（配合工艺），它们保证轨迹轮廓的精磨削或者CAM程序速度最优化加工。

该循环置于CAM程序的主程序之前（参见调用示例CYCLE832）。

要考虑不同的公差值说明，比如在G641时公差值作为ADIS=传送，在G642时更新轴专用的

MD 33100 COMPRESS\_POS\_TOL[AX]。

在激活输入区时可以接通或关闭“工艺匹配”（钥匙键位置?2）。

- 压缩（COMPCAD、COMPCURV、COMPOF、B-SPLINE），
- 轨迹控制运行（G64、G641、G642）或者
- 速度控制（FFWON、FFWOF、SOFT、BRISK）

在设置5轴转换（TRAORI）的情况下，可以在转换输入区接通或关闭这些设置。



注意机床制造商的说明！

## 调用举例CYCLE832

---

T1 D1

---

G54

---

M3 S12000

---

**CYCLE832 (0.2, 1003)** ; 粗加工

---

EXTCALL „CAM\_Form\_Schrupp“

---

**CYCLE832 (0.01, 102001)** ; 精加工

---

EXTCALL „CAM\_Form\_Schlicht“

---

**CYCLE832 (0.1, 0)** ; 撤消选择 (标准设定)

---

M02

---



对机床进行开机调试时，CYCLE832

可使得机床制造商免除不必要的优化工作。

这涉及到对参与加工的轴和 NCU

设置进行的优化工作（预控制、冲击限制，等等）。

#### 简化的程序调用

可以通过以下简化了的参数传送调用CYCLE832:

- CYCLE832() 与输入屏幕窗口  
“加工” “撤消选择” 的选择一致  
在CYCLE832中使用的G代码（参见章节3.17.5）  
设定到MD 20150:GCODE\_RESET\_VALUE中所  
设置的值。
- CYCLE832(0.01) 输入公差值。  
当前的G指令在  
循环中不改变。

### 3.17.1 在 HMI 菜单树中调用 CYCLE832



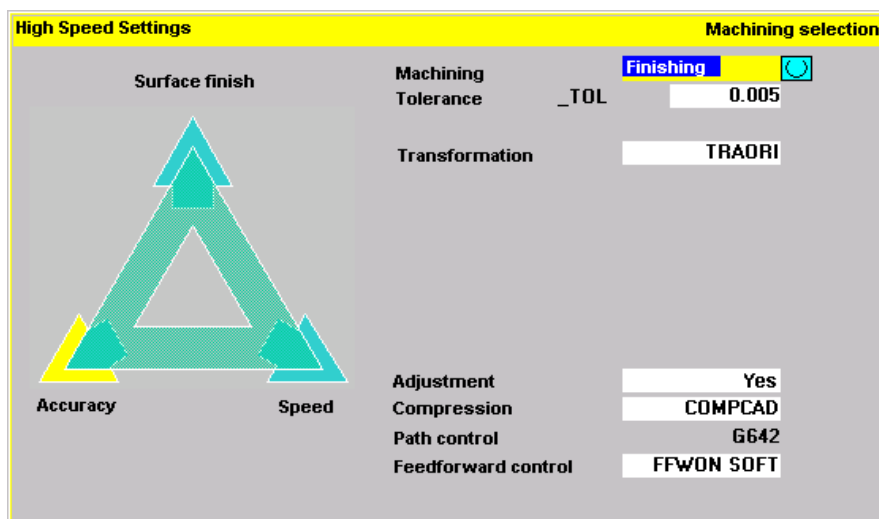
#### 参数说明

入口区程序/铣削

软键  →

将显示。

CYCLE832 标准界面的输入屏幕窗口



#### 加工 (\_TOLM)

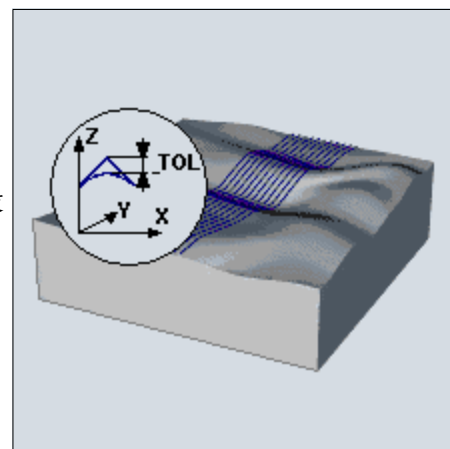
- 精加工 (缺省)
- 初精整
- 粗加工
- 撤销选择

变量 `_TOLM` 编码参见章节 3.17.2, 参数。

#### 公差 (\_TOL)

参与加工的轴的公差。该公差值根据 G 代码 (G642、COMPCAD、COMPCURV, ..) 写入到相应的机床数据或者设定数据 (参见章节 3.1.7.5)。

如果加工轴是一个回转轴, 则公差值带一个系数 (缺省系数=8) 写入到相应的回转轴的机床数据或者设定数据中。



在G641中公差值等于ADIS值。

在首次输入时按照以下数值预置公差：

- **精加工**：0.01（线性轴）0.08度（回转轴）
- **初精整**：0.05（线性轴）0.4度（回转轴）
- **初加工**：0.1（线性轴）0.8度（回转轴）
- **撤消选择**：0.01（线性轴）0.1度（回转轴）

考虑公制/英制尺寸系统。



如果要求公差值也对回转轴生效，则5轴转换必须通过机床制造商设置，但是并不是在每种情况下激活，比如CAM程序带解除的回转轴加工。

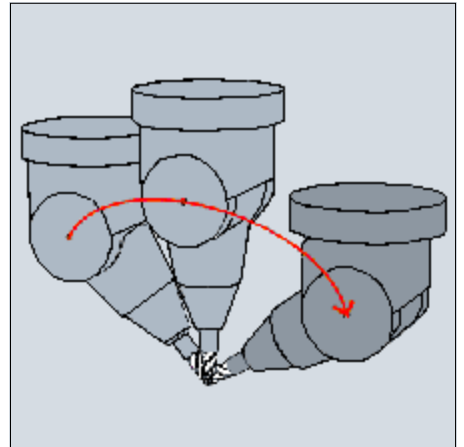
### 转换 (\_TOLM)

转换输入区仅在设定了NC选项时（5轴加工软件包已经设定）时显示。

- 否
- **TRAORI** → 1. 开启5轴转换
- **TRAORI(2)** → 2. 开启5轴转换

选择转换序号或者制造商循环，用于调用5轴转换：

- 可在 GUD7 变量 **\_TOLT2** 中存储制造商循环的名称，该名称用于调用转换的制造商循环。如果 **\_TOLT2** 为空（“缺省”），则在选择转换1,2... 时，用 **TRAORI (1)** 或者 **TRAORI (2)** 调用 5 轴转换。
- 如果在一个回转的平面中（参见CYCLE800）启动一个5轴转换程序，则清除刀架，在打开TRAOR后接收回转框架（工件基准）WPFRAME。



匹配, 匹配工艺 (参见章节 3.17.3)

- 是
- 否

只有当匹配设为“是”时, 才可以修改后面的输入参数。



“匹配”输入区和后面的输入区“压缩”、“轨迹控制”和“预控制”在钥匙开关位置0或者1时不可见。

压缩, NC-程序段压缩器 (\_TOLM)

- COMPOF (缺省)
- COMPCAD
- COMPCURV
- B-SPLINE



只有当压缩器功能选项已设置后, 输入区才可以显示。

当样条插补选项已设置后, 才可以选择B样条。

选项→ A, B 和 C 样条/压缩器功能

轨迹控制 (\_TOLM)

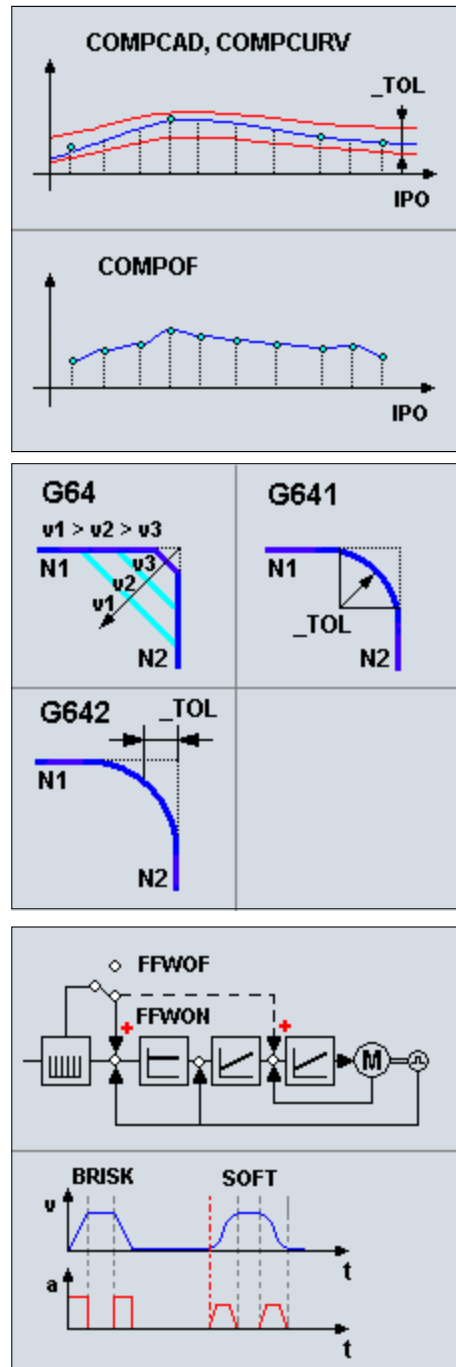
- G642 (缺省)
- G641
- G6450

NC程序段压缩器带COMPCAD、COMPCURV时始终选择G642。

预控制, 速度控制 (\_TOLM)

- FFWOF SOFT (缺省)
- FFWON SOFT
- FFWOF BRISK

选择预控制 (FFWON) 和冲击限制 (SOFT) 时, 以机床制造商对系统或者加工轴进行优化为前提条件。



## 3.17.2 参数



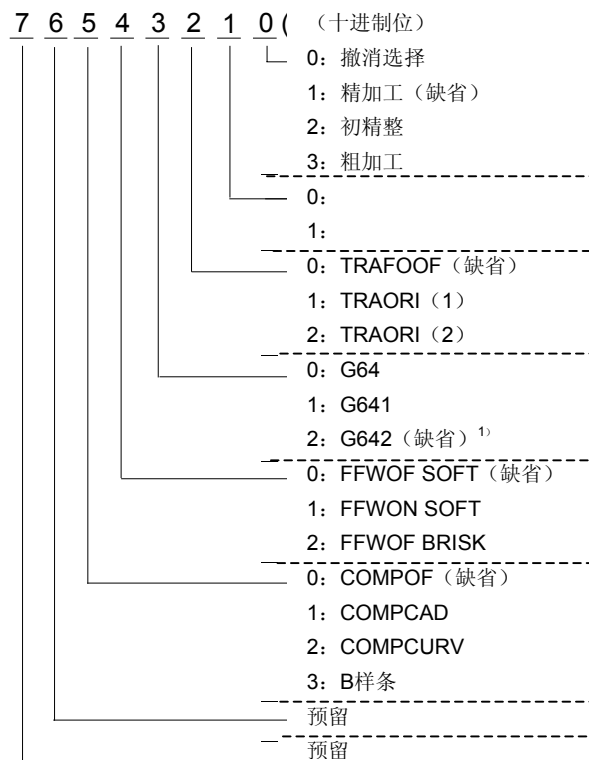
## 编程

CYCLE832(\_TOL, \_TOLM)



## 参数

_TOL	实数	加工轴公差 ->单位为毫米/英寸；度
_TOLM	整数	公差方式



1) 可由机床制造商更改设置，参见章节“工艺匹配”

### 3.17.3 匹配工艺



在“匹配工艺”输入区按“是”，可以由机床制造商和调试人员/程序员进行HSC加工时的工艺匹配。

这里必须始终注意后面CAM程序的工艺。

#### 机床制造商匹配

前提条件:



- 已经设置机床制造商口令字,
- 输入区“工艺匹配”->“是”

在打开循环CYCLE832输入屏幕窗口时, 参数以GUD7变量 `_TOLV[n]`, `_TOLT[n]`的值进行预置。n → 加工:

精加工, 初精整, 粗加工,

撤销选择

在修改参数时, 值直接写入到GUD7变量 `_TOLV[n]` 或者 `_TOLT[n]`中。

因此, 机床制造商可以把缺省设定匹配到其加工任务。

举例:

调用循环CYCLE832粗加工带3个轴, 加工轴公差0.1毫米, 带G642 (西门子缺省值)。

机床制造商可以修改粗加工工艺, 使用设定:

加工轴公差0.3毫米, TRAORI, G641。

在每次调用公差循环时显示该设定, 在加工时生效。



#### 机床制造商说明

1. 为了优化G64运行时的轨迹控制情况, 在CYCLE832中

按照以下表格重新计算过载系数速度跳转:

#### 计算所有加工轴过载系数速度跳转

IPO [ms]	过载系数
≥ 12	1.2
9	1.3
6	1.4
4	1.6
3	1.8



IPO:MD 10071: \$MN\_IPO\_CYCLE\_TIME

过载系数MD 32310:

\$MA\_MAX\_ACCEL\_OVL\_FACTOR[AX]



通过在循环CYC\_832T中设置局部变量\_OVL\_on=0，可以通过CYCLE832关闭过载系数的计算。

## 2. 激活的 NC

程序段压缩器 (COMPCAP) 或者精磨削 (G642) 时的公差在 CYCLE832 中写入到MD 33100:

\$MA\_COMPRESS\_POS\_TOL[AX] (线性加工轴) 上。

如果回转轴已参与到加工中 (TRAORI)，则用**系数 8**

将该公差写入到回转轴的 MD 33100:

\$MA\_COMPRESS\_POS\_TOL[AX] 上。

如果必须使用另一个系数，则可在循环

CYC\_832T 中用相应的值预置局部变量**系数**。

## 调试人员/编程人员匹配

前提条件:



- 删除制造商口令字
- 钥匙开关位置2或者3
- 保护级5, 4, 3, 2
- 输入变量“工艺匹配”->“是”

在对工艺进行匹配时，调试人员/编程人员必须对后面的CAM加工程序真正了解。

修改后的数据用于CYCLE832的生成，并适用于CYCLE832的实际调用。

机床制造商的缺省设定没有改变。

### 3.17.4 匹配附加的程序参数CYC\_832T



机床制造商还要求一个附加的 CYCLE832 特性，该特性通过工艺匹配而表现出，因此可在循环 CYC\_832T 中进行相应的修改。为此应将 CYC\_832T 复制到目录 CMA.dir 中（HMI 制造商）并装载至 NCU。CYC\_832T 说明一个框架程序。这些修改必须由机床制造商用文件记录下来。当循环 CYC\_832T 已经装载到 NCU 以后，它可以自动由 CYCLE832 调用。匹配 CYC\_832T 不要求对 CYCLE832 进行修改。



#### 参数

CYC\_832T(\_ASVS, \_FACTOR, \_OVL\_on)

_ASVS	在标记 _MD 到 _M4 中可以进行机床专用的匹配。 _M4 = Init CYCLE832 调用在 CYCLE832 自身程序运行结束之前进行。 _M0 = 撤消选择 CYCLE832 _M1 = 精加工 _M2 = 初精整加工 _M3 = 粗加工
_FACTOR <sup>1)</sup>	回转轴公差系数
_OVL_on <sup>1)</sup>	0=过载系数 MD \$MA_MAX_ACCEL_OVL_FACTOR 没有匹配

1) 参数 \_FACTOR 和 \_OVL\_on 仅在标记 \_M4 (INIT) 中生效。



#### 编程举例

机床制造商要进行以下的匹配：

1. 回转轴应该比线性轴的公差高系数 12
2. 在“精加工”“初精整”和“粗加工”加工方式中，轨迹冲击  
     (MD \$MC\_MAX\_PATH\_JERK)  
     确定为值 15，轴冲击  
     (MD \$MA\_MAX\_AX\_JERK[AX]) 确定为值 150。

3. 在撤消选择CYCLE832时，修改后的机床数据应还原为基本设定（1000,15）。

```

%_N_CYC_832T_SPF
; $PATH=/_N_CST_DIR
PROC CYC_832T(INT _ASVS,VAR INT
  _FACTOR,VAR INT _OVL_on) SAVE DISPLOF
...
N801 CASE _ASVS OF 0 GOTOF _M0 1 GOTOF
  _M1 2 GOTOF _M2 3 GOTOF _M3 4 GOTOF
  _M4 DEFAULT GOTOF _MEND
...
_M4:
  _FACTOR=12 ;Init
  GOTOF _MEND

_M0::撤销选择
$MC_MAX_PATH_JERK=1000
$MA_MAX_AX_JERK[X]=15
$MA_MAX_AX_JERK[Y]=15
$MA_MAX_AX_JERK[Z]=15
  GOTOF _MEND

_M1:: 精加工
_M2:: 初精整
_M3:: 粗加工
$MC_MAX_PATH_JERK=15
$MA_MAX_AX_JERK[X]=150
$MA_MAX_AX_JERK[Y]=150
$MA_MAX_AX_JERK[Z]=150

  GOTOF _MEND
_MEND:
RET

```

为了可以在CYC\_832T中，在程序运行过程中所有保护级的情况下均可以重写机床数据，必须用REDEF重新定义。

举例:

```

%_N_MGUD_DEF
; $PATH=/_N_DEF_DIR

REDEF $MC_MAX_PATH_JERK APR 7 APW 7
REDEF $MA_MAX_AX_JERK APR 7 APW 7

M30

```

### 3.17.5 接口



#### G代码

在CYCLE832中编程的G指令清单：

- G64, G641, G642
- G601
- FFWON, FFWOF
- SOFT, BRISK
- COMPCAD, COMPCURV, COMPOF, B-SPLINE
- TRAORI, TRAORI(2), TRAOF OF
- UPATH

说明：不应该在随后的 CAM 程序中生成G 指令。分开工艺—几何。



#### 机床参数

下面的机床数据在循环CYCLE832中使用，从而对公差值进行相应的描述：

MD号	MD名称	注释
10071	\$MN_IPO_CYCLE_TIME	IPO 节拍
20480	\$MC_SMOOTHING_MODE	
20482	\$MC_COMPRESSOR_MODE	
24100	机床数据5轴转换	
	到	
24462		

下面的机床数据在循环CYCLE832中改写：

MD号	MD名称	注释
20490	\$MC_IGNORE_OVL_FACTOR_FOR_ADIS	
33100	\$MA_COMPRESS_POS_TOL[AX]	几何轴1...3
33100	\$MA_COMPRESS_POS_TOL[AX]	回转轴1和2 <sup>1)</sup>
32310	\$MA_MAX_ACCEL_OVL_FACTOR[AX]	几何轴1...3
32310	\$MA_MAX_ACCEL_OVL_FACTOR[AX]	回转轴1和2 <sup>1)</sup>

1) 按照机床数据，5轴转换



#### 设定数据

设定数据清单，在CYCLE832中改写：

SD号	SD名称	注释
42450	\$SC_CONTPREC	在 CPRECON 和 G64 时
42465	\$SC_SMOOTH_CONTUR_TOL	符合线性轴公差
42466	\$SC_SMOOTH_ORI_TOL	符合回转轴公差
42475	\$SC_COMPRESS_CONTUR_TOL	仅在 COMPCURV 时
42476	\$SC_COMPRESS_ORI_TOL	仅在 COMPCURV 时

1) 设定数据 \$SC\_SMOOTH\_CONTUR\_TOL 和

\$SC\_SMOOTH\_ORI\_TOL 的生效性与 MD20480: \$MC\_SMOOTHING\_MODE 相关。

设定数据 \$SC\_COMPRESS\_CONTUR\_TOL 和

\$SC\_COMPRESS\_ORI\_TOL 的生效性与 MD20482: \$MC\_COMPRESSOR\_MODE 相关。



### 通道专用的变量GUD7

下面通道专用的变量必须激活用于CYCLE832功能（机床制造商）。

其定义是西门子标准循环软件包GUD7定义的组成部分。

参数	格式	占用	注释
<b>_TOLT2 [2]</b>	字符串[32]	„“（缺省）	子程序程序名称，用于调用5轴转换
<b>_TOLT [4]</b>	整数	<b>数组（4）：</b> 0: 撤销选择 1: 精加工 2: 初精整 3: 粗加工	数组，用于存储机床制造商的工艺数据设定 根据变量_TOLM编码（参见参数）
<b>_TOLV [4]</b>	实数	<b>数组（4）：</b> 0: 撤销选择 1: 精加工 2: 初精整 3: 粗加工	数组，用于存储加工轴的工艺数值，由机床制造商设定 （参见工艺匹配）。 缺省值： (GUD7.def)      0.01 撤销选择 0.01 精加工 0.05 初精整 0.1 粗加工

### 3.17.6 故障信息



#### 说明

#### CYCLE832 报警原因

报警号	报警文本	说明，纠正
61191	“5轴转换没有设置”	1. 选项5轴或者多轴插补加工软件包没有设置。 2. 检查 MD 24100: \$MC_TRAFO_TYPE_1 到\$MC_TRAFO_TYPE_8 到一个有效的 5轴转换类型
61192	“第二个5轴转换没有设置”	
61193	“选项压缩器没有设置”	设置选项样条插补
61194	“选项样条插补没有设置”	(A、B 和 C 样条/压缩器功能

### 3.18 模腔循环CYCLE60 (自循环 SW 6.4 起)



#### 编程

CYCLE60 (\_TEXT, \_RTP, \_RFP, \_SDIS, \_DP, \_DPR, \_PA, \_PO, \_STA, \_CP1, \_CP2, \_WID, \_DF, \_FFD, \_FFP1, \_VARI, \_CODEP)



#### 参数

_TEXT	字符串	待模腔加工的文本（最大91个字符）
_RTP	实数	退回平面（绝对）
_RFP	实数	基准面（绝对）
_SDIS	实数	安全距离（加到_RFP，不输入符号）
_DP	实数	深度（绝对）
_DPR	实数	相对于基准面的深度（不输入符号）
_PA	实数	文本排列基准点（绝对） <ul style="list-style-type: none"> <li>• 第一轴位置（在_VARI =直角时），或者</li> <li>• 圆弧半径（在_VARI =极线时）</li> </ul>
_PO	实数	文本排列基准点（绝对） <ul style="list-style-type: none"> <li>• 第二轴位置（在_VARI =直角时），或者</li> <li>• 与第一轴的角度（在_VARI =极线时）</li> </ul>
_STA	实数	与第一轴的角度（仅在_VARI =直线时）
_CP1	实数	圆弧圆心（绝对），（仅在找正圆弧时） <ul style="list-style-type: none"> <li>• 第一轴位置（在_VARI =直角时），或者</li> <li>• 圆弧半径（在_VARI =极线时），与圆心相关</li> </ul>
_CP2	实数	圆弧圆心（绝对），（仅在找正圆弧时） <ul style="list-style-type: none"> <li>• 第二轴位置（在_VARI =直角时），或者</li> <li>• 与第一轴的角度（在_VARI =极线时）</li> </ul>
_WID	实数	字体高度（不输入符号）
_DF	实数	字体宽度规格（根据_VARI十万位） <ul style="list-style-type: none"> <li>• 符号间距增量，毫米/英寸，或者</li> <li>• 文本总宽度增量，毫米/英寸，或者</li> <li>• 张角，度</li> </ul>
_FFD	实数	深度方向的进给
_FFP1	实数	表面加工的进给

_VARI	整数	加工方式：（不输入符号） 个位： 基准点 值： 0...直角（笛卡儿） 1...极坐标 十位： 文本对齐 值： 0...文本以直线对齐 1...文本对齐到一个圆弧，上面 2...文本对齐到一个圆弧，下面 百位： 保留 千位： 文本水平基准点 值： 0...左侧 1...中心 2...右侧 万位： 文本垂直基准点 值： 0...下面 1...中心 2...上面 十万位： 文本宽度 值： 0...符号间距 1...文本总宽度（仅在线性排列文本时） 2...张角（仅在圆弧排列文本时） 7. 右侧位置（百万）： 圆弧圆心 值： 0...直角（笛卡儿） 1...极坐标
_CODEP	整数	输入活字代码页序号 1252...代码页用于中欧语言 （值 0 在内部如同 1252 处理）

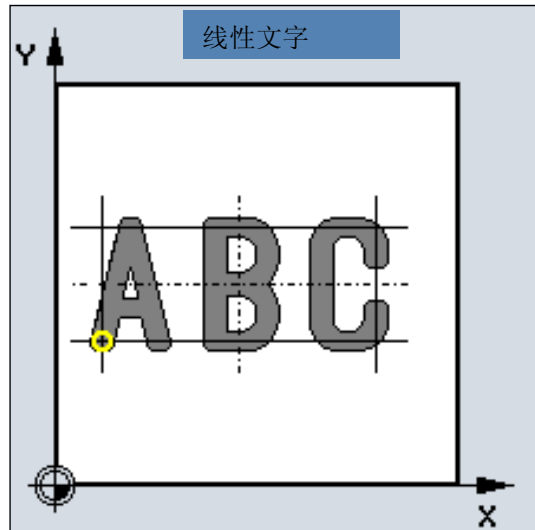


### 功能

使用模腔循环 CYCLE60 可铣削位于一条直线或一个圆弧的文字。沿着条圆弧可向上或向下对齐文字。

字体高度和文字总宽度，字符间距或者圆弧排列时的张角，以及字体的对齐均可以通过不同的参数进行改变。

无法调整字符形式。循环使用一个成比例的字体，也就是说，各个字符的大小不同。字符的行宽取决于刀具直径。



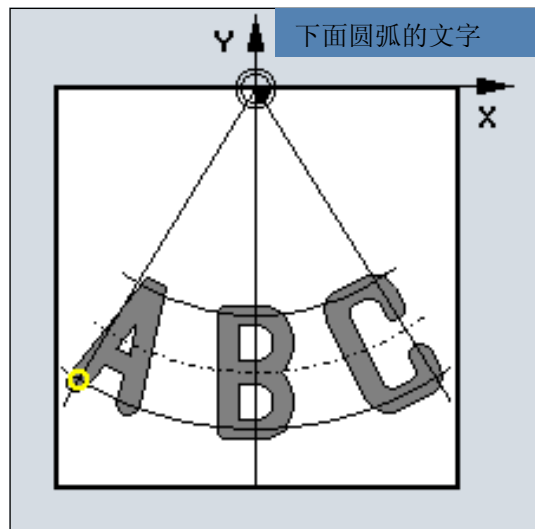
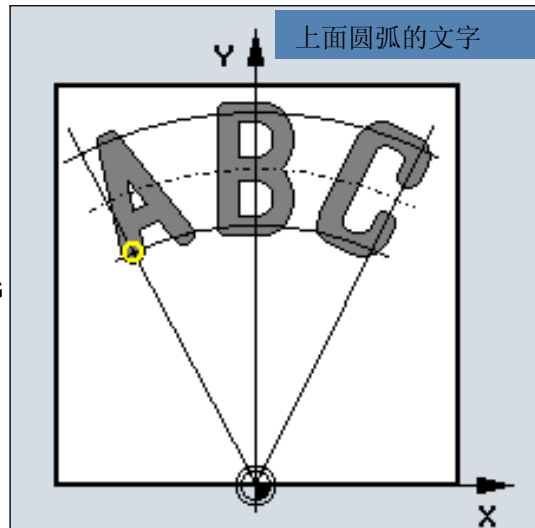
### 工作流程

循环开始之前到达的位置:

出发位置是一个任意位置，从这个位置可以返回到第一个字符的起始位置，没有轮廓冲突。

该循环产生以下的运动过程:

- 在加工平面以快速移动定位到起始位置，接着再以G0运行到提前了安全距离的基准面
- 以深度进给\_FFD的进给插入到编程的深度
- 以表面加工\_FFP1的进给加工各个字符
- 在每个字符加工完毕后，以G0退回到安全平面，并且以G0定位到下一个字符加工的起始点
- 在所有编程的字符加工后，刀具以G0定位到退回平面。







### 参数说明



参数 `_RTP`、`_RFP`、`_SDIS` 参见模拟量  
`RTP`、`RFP`、`SDIS`

章节 2.1.2 (钻削、定中心 – `CYCLE81`)。

#### **`_TEXT`** (待模腔加工的文字)

待模腔加工的文字最多可为 91 个字符。

允许的字符为所有大小写字母、数字以及代码页  
1252中的大部分特殊字符。

允许的特殊字符参见下一页说明“字符组”。

如果文字中有不允许的字符，则循环中断，并发出报  
警

61179 “字符不存在”。

#### **`_DP`, `_DPR`** (字体深度)

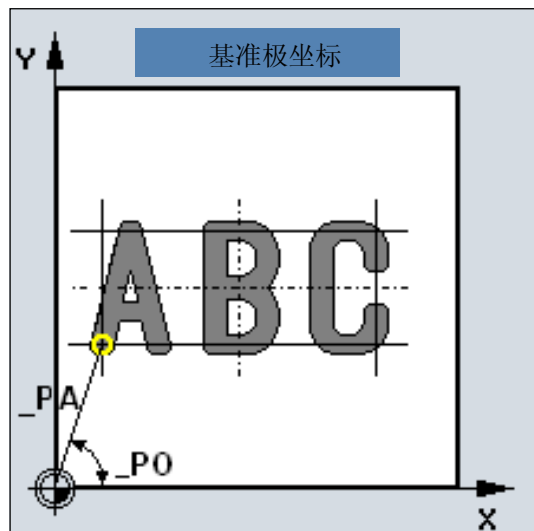
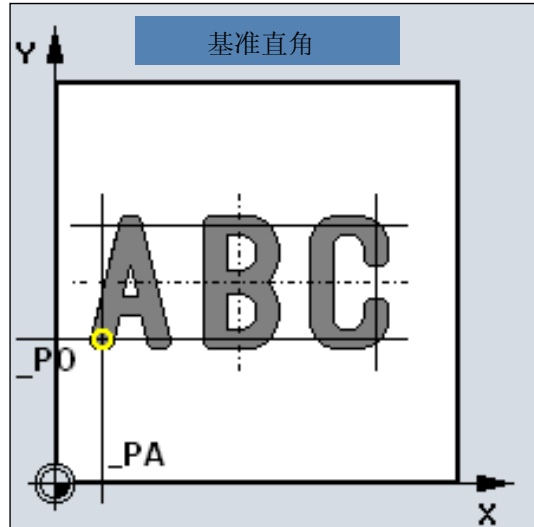
根据基准面，可规定字体深度为绝对值 (`_DP`)，也可  
以规定为相对值 (`_DPR`)。在相对尺寸时，循环利用基  
准面和退回平面的位置自行计算所产生的深度。

通过进刀达到字体深度，不会以单个进刀尺寸进行  
分配。通过 `G1` 垂直进行每个字符的深度进刀。

**\_PA, \_PO (文字排列基准点)**

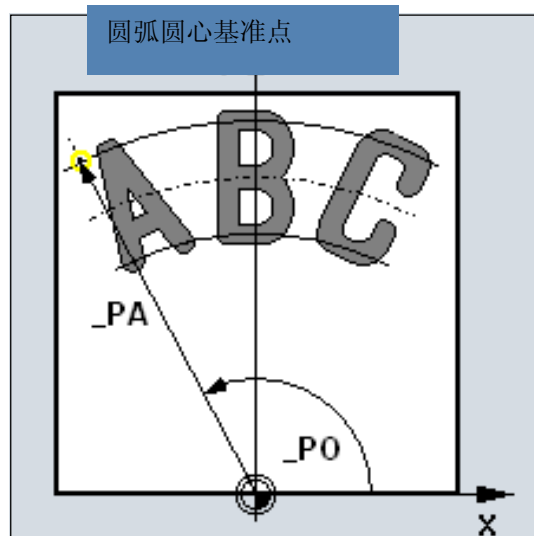
与文字是线性排列还是圆弧排列无关，基准点的编程可以选择直角（笛卡儿）坐标或者极坐标。

在文字线性排列时，基准点始终从当前的工件零点出发。



在以圆弧排列并且极坐标编程基准点时，这始终与圆弧圆心相联系。

是直角坐标还是极坐标说明基准点，这通过参数\_VARI确定。

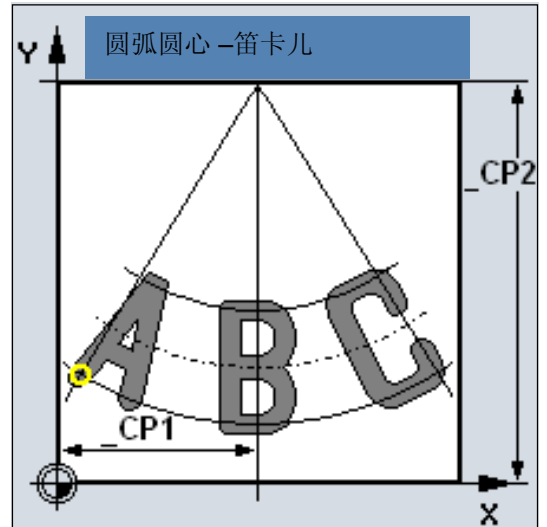


**\_CP1, \_CP2 (圆弧圆心)**

在以圆弧排列时，圆弧圆心同样可以选择直角（笛卡儿）或者极坐标编程。

是直角坐标还是极坐标说明圆心，这通过参数\_VARI确定。

这些参数仅在圆弧排列时生效。

**\_STA (文字定位角度)**

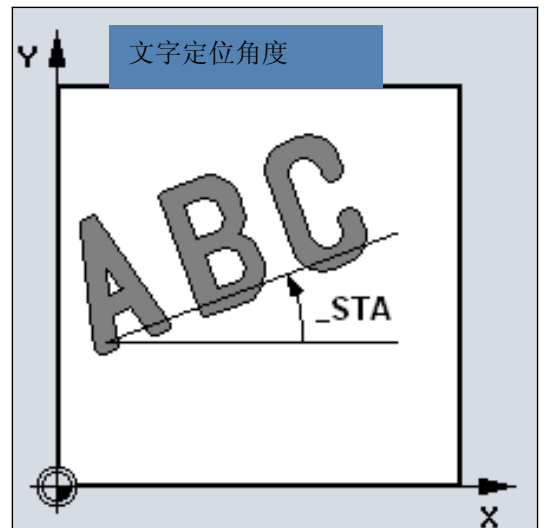
\_STA说明平面中第一轴（横坐标）和待写文字纵向方向（文字所在的直线）之间的夹角，该参数仅在文字以直线排列时起作用。

**\_WID (文字高度)**

编程的字体高度等于大写字母的高度或者数字的高度扣除2\*铣刀半径。

对于一些特殊字符，比如(),

上下各加尺寸0.15\*\_WID。

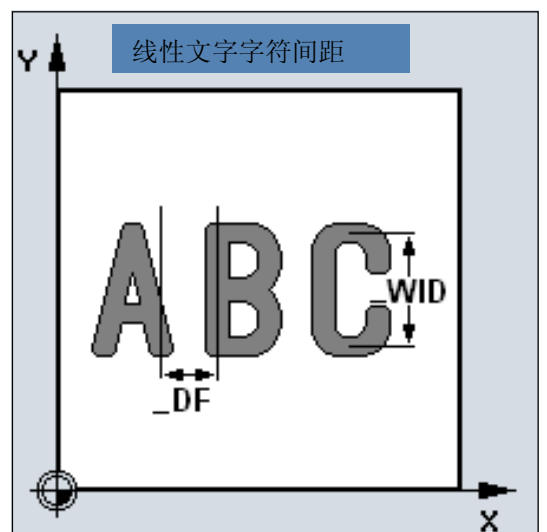
**\_DF (字符间距)**

对于线性文字，可以说明字符间距，也可以说明文字总宽度，这些值为增量值。

在循环中进行监控，即所编程的总宽度是否可以实现，也就是说字符宽度之和不要过大。


在有问题时出现报警：

61176 “文字长度\_DF编程太小”。



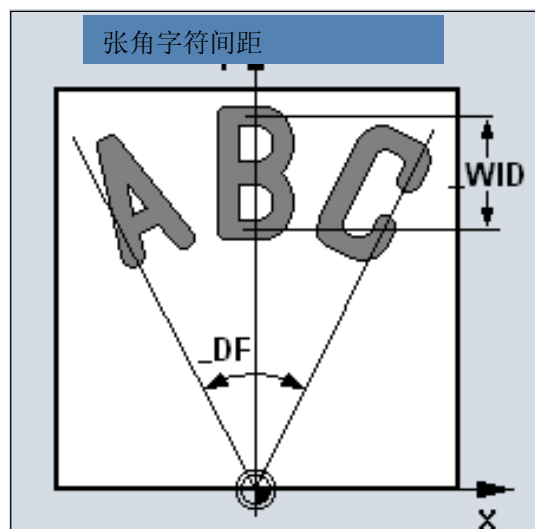
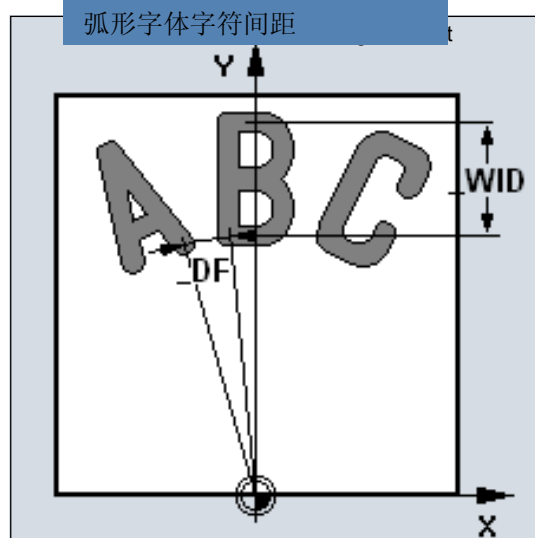
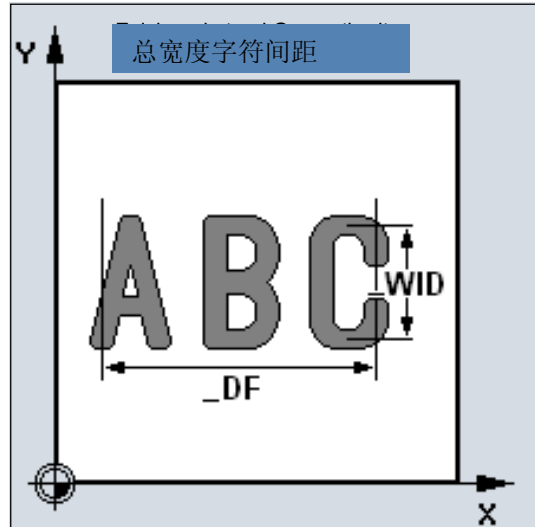
在圆弧上进行规定时，可给定字符间距或第一个字符和最后一个字符之间的张角。这里，字符间距并不是相邻字符之间的线性间距，而是用在圆弧上作为圆弧尺寸。总是将张角规定为正值。其与第一个字符的中心线与最后一个字符中心线之间的角度相关。

通过参数\_VARI说明字符间距、总宽度或者张角。

 如果必须在一个整圆上均匀分配字符，您只需简单编程\_DF=360。随后，循环自动将字符分配至整圆。这可省去计算第一个和最后一个字符之间的张角。

#### **\_FFD, \_FFP1 (进给)**

在平面的所有运动中进给\_FFP1均有效（铣削字符），该进给\_FFD在垂直插入时以G1生效。



### \_VARI (设定文字方向)

使用参数\_VARI可以确定文字方向。

值参见本章中参数\_VARI。

### \_CODEP (代码页序号)

现在，在 CYCLE60 中仅实现了代码页 1252 的字符。

因此，参数总是包含数值 1252。

如果传送的序号循环不识别，则发出报警：

61178 “通道 %1 程序段 %2: 字循环: 没有代码页”

并且循环停止。



### 符号组

除了字母和数字之外，也可以有以下的字符（程序段字符、括号、计算符号、货币符号和其它特殊符号）：

.	,	:	;	!	?	"	'	`	´	^	_
(	)	[	]	{	}		#	∅	ø	°	
+	-	*	/	\	÷	x	<	>	=	~	
€	\$	£	§	&	%	@	©	®			
à	á	â	ã	ä	å	À	Á	Â	Ã	Ä	Å
è	é	ê		ë		È	É	Ê		Ë	
ì	í	î		ï		Ì	Í	Î		Ï	
ò	ó	ô	õ	ö		Ò	Ó	Ô	Õ	Ö	
ù	ú	û		ü		Ù	Ú	Û		Ü	
	ý			ÿ		Ý				ÿ	
ñ	Ñ	š	Š	ž	Ž	ç	Ç	ø	¥		
ß	μ	ð	Ð	þ	Þ	æ	Æ	œ	Œ		

对于两个特殊字符单引号和双引号（引号）的编程，适用于一个特殊规则，因为这两个符号在NC语言的字符串处理中已经具有一定的功能。在文字字符串中它们必须在单引号中闭合写入。

举例：

产生以下的程序段：

这是一个带“和”的文字。

为了在\_TEXT中编程，要求：

这是一个带‘“’和‘”’的文字。

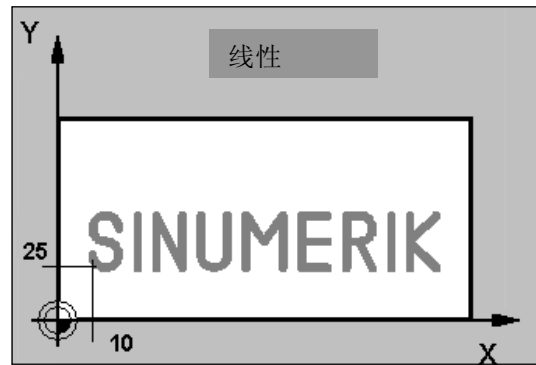
## 3.18 模腔循环CYCLE60 (自循环 SW 6.4 起)



## 编程举例

使用该程序雕刻字符“SINUMERIK”，在一条直线上。

基准点位于X10 Y25 左下方。字高14 mm，字符之间的间距规定为 5 mm。



```

N10 G17 DIAMOF F2000 S1500 M3
N20 T1 D1
N30 M6
N40 G0 G90 Z100
N50 CYCLE60 ("SINUMERIK",100,0,1,-
1.5,0,10,25,0,,,14,5,2500,2000,0,1252)
N60 M30

```

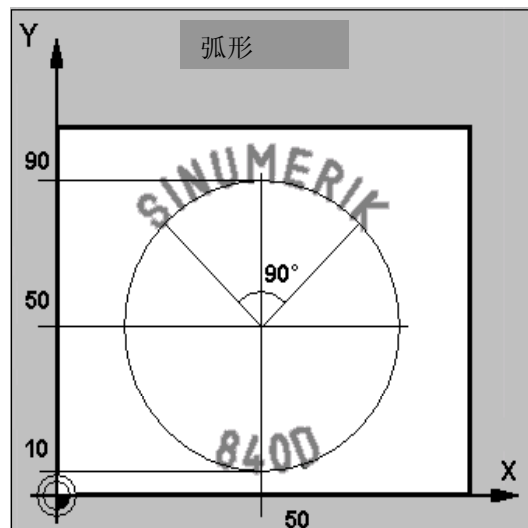


### 编程举例2

使用该程序雕刻2个字符串，成弧形排列，“SINUMERIK”在上面圆弧，“840”在下面圆弧。

基准点位于 X50, Y90 的中心处，X50 Y10 的下方。由基准点间的间距和圆弧中心点 X50, Y50 得出圆弧半径为 40 mm。

文字为9毫米高，字符间距由编程的张角90度或者30度产生。



```
N10 G17 DIAMOF F2000 S800 M3
```

```
N20 T1 D1
```

```
N30 M6
```

```
N40 G0 G90 Z100
```

```
N50 CYCLE60("SINUMERIK",100,0,1,-1.5,      ; 文字在上面的圆弧
0,50,90,0,50,50,9,90,2500,2000,
201010,1252)
```

```
N60 CYCLE60("840D",100,0,1,-1.5,      ; 文字在下面的圆弧
0,50,10,0,50,50,9,34,2500,2000,201020,
1252)
```

```
M30
```

用于记录



## 车削循环

4.1	一般提示.....	4-282
4.2	前提条件.....	4-283
4.3	切槽循环– CYCLE93.....	4-286
4.4	退刀槽循环– CYCLE94.....	4-295
4.5	切削循环– CYCLE95.....	4-299
4.6	螺纹退刀槽 – CYCLE96.....	4-312
4.7	螺纹切削 – CYCLE97.....	4-316
4.8	螺纹链 – CYCLE98.....	4-323
4.9	螺纹精整.....	4-329
4.10	扩展的切削循环 - CYCLE950.....	4-331

## 4.1 一般提示

在下面的章节中，说明如何编程车削循环。要求将该章节作为选择循环时的指南并使用所提供的各种参数。除了各个循环功能的描述和为此所列出的参数外，您还可在各节结束处找到能简化循环过程的编程举例，通过这些示例可以使您编程循环更加方便。

各节是按照以下的原则进行组合的：

- 编程
- 参数
- 功能
- 工作流程
- 参数说明
- 其它说明
- 编程举例

上述各点中，编程和参数两节为熟练的用户所准备，而作为初学者，有关循环编程的必要知识均可以在功能、工作流程、参数说明、其它说明和编程举例中找到。

## 4.2 前提条件

### 车削循环数据块

车削循环需要 GUD7.DEF

模块，它与循环一起位于磁盘中供使用。

### 调用条件和返回条件

在循环调用之前有效的G功能和可编程的框架可以在循环之后仍然保持。

### 平面定义

在循环调用之前定义加工平面。通常，在车削时涉及到的是G18（ZX平面）。对于车削，当前平面的两根轴在下面的章节中被命名为纵向轴（该平面的第一根轴）和平面轴（该平面的第二根轴）。

在车削循环中直径编程有效时始终把平面中的第二轴作为平面轴计算。**文献：** /PG/编程说明



### 主轴处理

在编制车削循环时，始终使其中所包含的主轴指令以系统中有效的主主轴为基准。

如果在一台机床中有几个主轴，则先前有效的主轴必须作为主主轴定义。

**文献：** /PG/编程说明

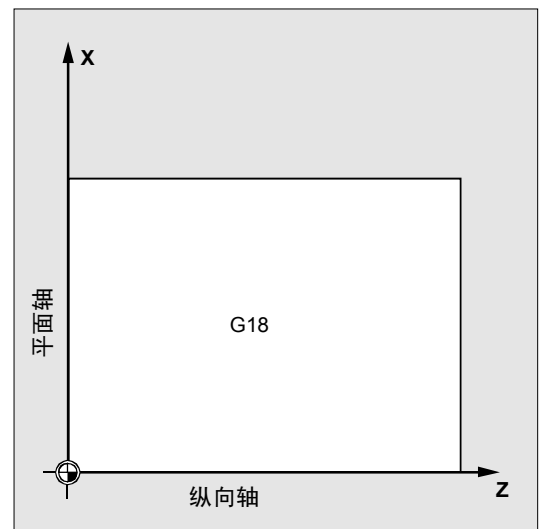


### 加工状态的信息

加工车削循环期间，在控制系统屏幕上显示说明加工状态的信息。可能会有下列信息：

- “螺纹导程 <序号> - 作为长螺纹加工”
- “螺纹导程 <序号> - 作为盘丝加工”

在信息文本部分，每次均有一个正在加工形状的序号。这些信息不中断程序执行，并且一直保持直至下一个信息的出现或者循环结束。



### 循环设定数据

毛坯切削循环CYCLE95中有一个设定数据，它位于模块GUD7.DEF中。

通过循环设定数据\_ZSD[0]可改变CYCLE95中深度进刀MID的计算情况。如果设置为零，则如前所述计算参数。

- \_ZSD[0]=1 MID 是一个半径值
- \_ZSD[0]=2 MID 是一个直径值

对于切槽循环CYCLE93，在GUD7.DEF中有一个设定数据。通过该循环设定数据\_ZSD[4]可以影响第一个切槽之后的退回。

- \_ZSD[4]=1 以G0退回
- \_ZSD[4]=0 以G1退回（如同往常一样）

对于切槽循环CYCLE93，在\_ZSD[6]中有一个设定方式，用于镜像时的性能。

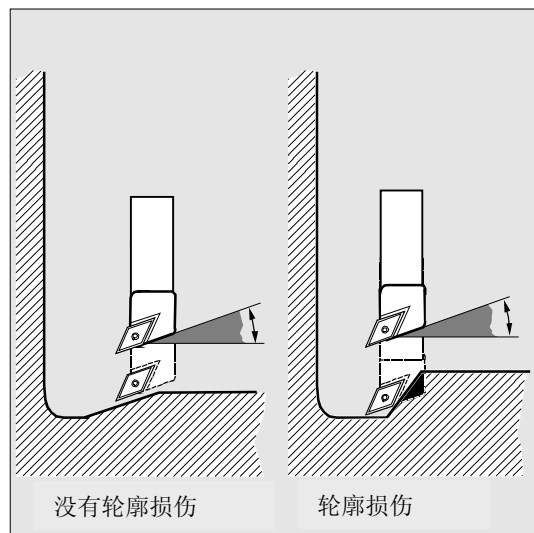
- \_ZSD[6]=0 在镜像有效时，  
在循环中交换刀具补偿
- （用于没有可定向的刀架）
- \_ZSD[6]=1 在镜像有效时，  
在循环中不交换  
刀具补偿
- （用于带可定向的刀架）

### 轮廓监控 与刀具的自由切削角度 有关

用特定的车削循环（在该循环中生成带底切的运行运动）监控有效刀具的自由切削角是否可能造成轮廓损伤。该角度将被作为刀具补偿中的值输入

（D 补偿中的参数 P24 下）。

输入角度值，0到90度之间，不带符号。



输入自由切削角时，必须要注意的是，该角度于是纵向加工方式还是平面加工方式。如果要求为纵向加工和平面加工使用刀具，则必须在不同的自由切削角情况下使用两个刀具补偿。

在循环中检测用所选择的刀具是否可以加工编程的轮廓。

如果使用这把刀不可以进行加工，则

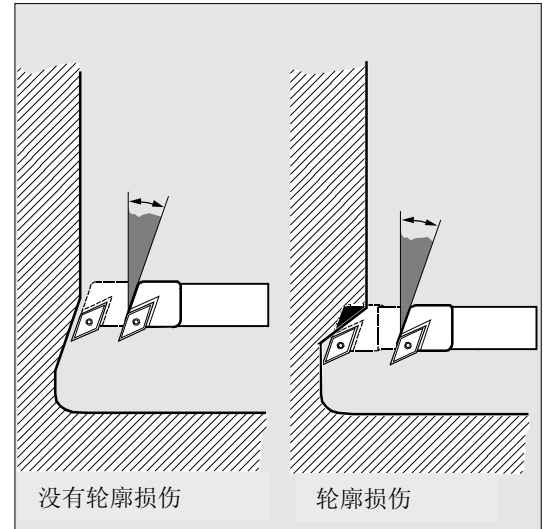
- 循环停止执行，并带报警（在切削时），或者
- 通过一个信息数据继续进行轮廓加工（在退刀槽循环时）。然后由刀沿几何尺寸确定轮廓。

必须要注意的是：通过有效的比例系数或者在当前平面中的旋转，改变角度的关系，而这在循环内部的轮廓监控中不可能被考虑。

如果自由切削角在刀具补偿中给出零，则该监控不进行。在各自的循环中描述出精确应答。

#### 车削循环带有效的适配器转换

自软件版本NCKSW6.2起，也可进行带有效适配器转换的车削循环。总是读出转换后的刀具补偿数据，用于刀沿位置和后角。



### 4.3 切槽循环 - CYCLE93



#### 编程

CYCLE93 (SPD, SPL, WIDG, DIAG, STA1, ANG1, ANG2, RCO1, RCO2, RCI1, RCI2, FAL1, FAL2, IDEP, DTB, VARI, \_VRT)



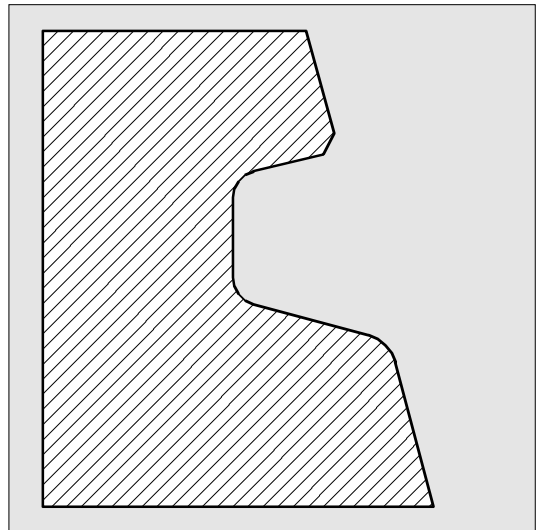
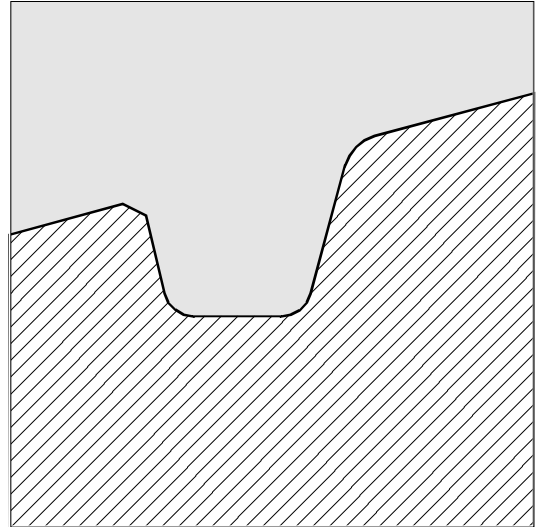
#### 参数

SPD	实数	平面轴中起始点（不输入符号）
SPL	实数	纵向轴起始点
WIDG	实数	切槽宽度（不输入符号）
DIAG	实数	切槽深度（不输入符号）
STA1	实数	轮廓和纵向轴之间的角度值范围：0<=STA1<=180 度
ANG1	实数	螺纹啮合角1：在起始点确定的切槽一侧（不输入符号） 值范围：0<=ANG1<89.999 度
ANG2	实数	螺纹啮合角2：在另一侧（不输入符号）值范围：0<=ANG2<89.999
RCO1	实数	半径/棱边1，外部：在由起始点确定的一侧
RCO2	实数	半径/棱边2，外部
RCI1	实数	半径/棱边1，内部：在起始点侧
RCI2	实数	半径/棱边2，内部
FAL1	实数	切槽底部的精加工余量
FAL2	实数	侧壁的精加工余量
IDEP	实数	进刀深度（不输入符号）
DTB	实数	切槽底部停留时间
VARI	整数	加工方式 值范围：1...8 和 11...18
_VRT	实数	可变的退回位移（自轮廓），增量（不输入符号）



### 功能

切槽循环可以加工对称的和非对称的切槽，用于纵向加工和平面加工，在任意的直线轮廓单元中。您可以加工外部切槽和内部切槽。



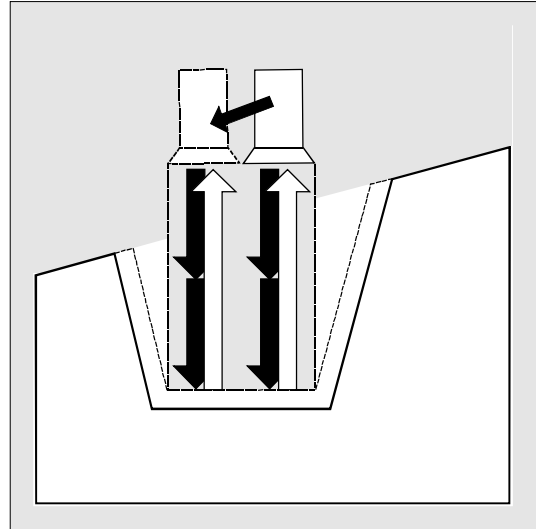
### 工作流程

在深度方向（到切槽底部）和宽度方向（从切槽到切槽）以最大可能的数值进行划分。

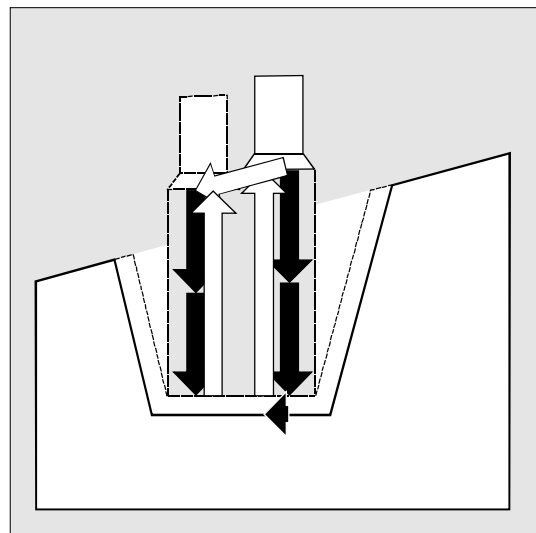
在倾斜方向切槽时，从一个切槽到下一个切槽以最短的位移、平行于锥面运行。这里在循环内部计算一个到轮廓的安全距离。

**1. 步骤**

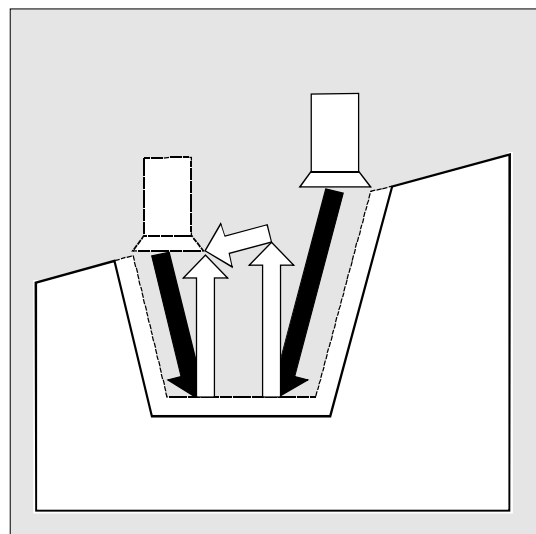
以单个的进刀尺寸轴向平行粗加工至底部。每次进刀后，空运行至断屑处。

**2. 步骤**

以垂直于进刀方向一次和多次切削加工切槽。此时，再次根据进刀深度分配每一次切削。自第二次切削起，沿着切槽宽度在退回前空运行1毫米。

**3. 步骤**

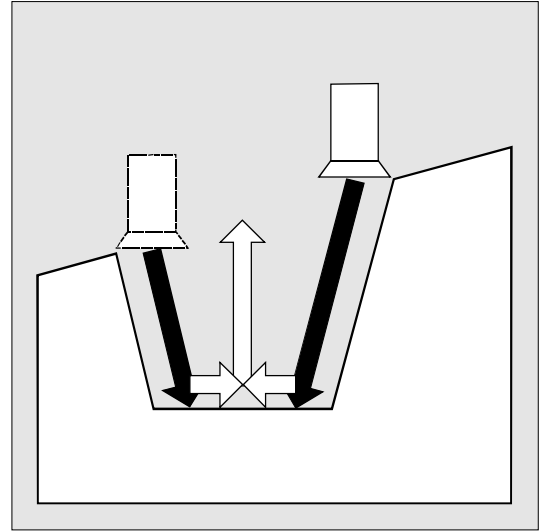
如果已在ANG1或ANG2下编程角度，则一步切削侧壁。如果侧壁宽度较大，则沿着切槽宽度以多步进刀。





#### 4. 步骤

与轮廓平行，自边缘切削精加工余量，直至切槽中心。此时，刀具半径补偿由循环自动选择并撤消选择。





## 参数说明

### SPD 和 SPL (起始点)

用该坐标定义切槽起始点，自该起始点出发在循环中计算出形状。由循环确定起始点，该起始点在开始时自行返回。如果是外部切槽，首先以纵向轴方向运行；如果是内部切槽，则首先以平面轴方向运行。在弯曲的轮廓单元中，切槽可以按照不同的方式进行。分别根据弯曲的形状和半径，可以设定一条轴向平行的直线（超过最大曲率），或者设定一条切槽边缘点的切向斜线。

只有当相应的边缘点位于直线上时，切槽边缘上半径和棱角在弯曲的轮廓中才有意义。

### WIDG 和DIAG (切槽宽度和切槽深度)

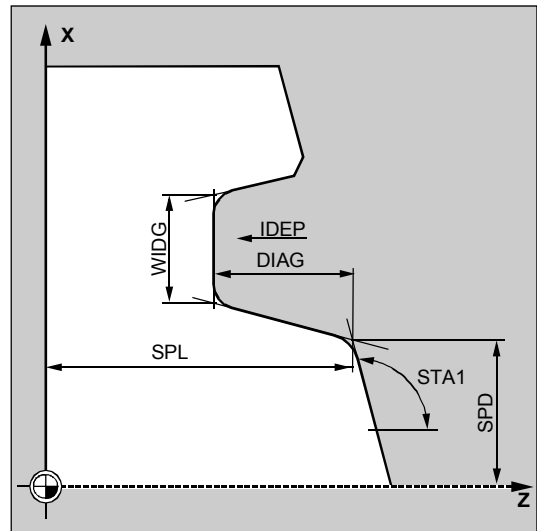
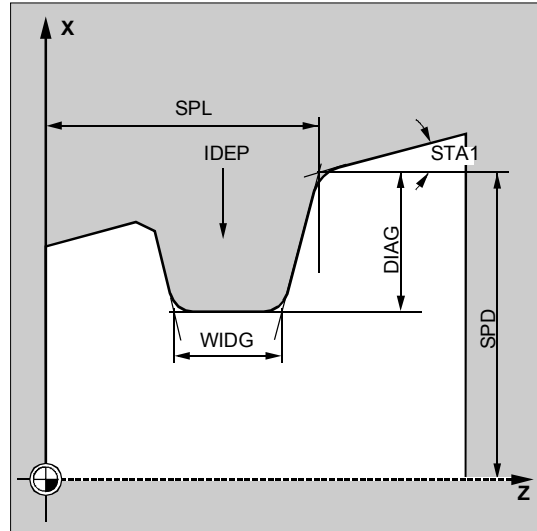
用参数切槽宽度 (WIDG) 和切槽深度 (DIAG) 确定切槽形状。循环在进行计算时，总是从 SPD 和 SPL 下编程的点开始。

如果切槽比有效的刀具宽，则以多步切削宽度。

此时，由循环平均分配总宽度。减去刀沿半径后，最大进刀为刀具宽度的95%。由此可允许进行切削重叠。

如果编程的切槽宽度小于实际的刀具宽度，则显示报警信息61602“刀具宽度错误定义”。

循环不启动循环加工，加工被中断。如果在循环内部识别到刀沿宽度为零，则也会显示报警。



**STA1 (角度)**

用参数 STA1

编程斜置平面的角度，要求在该平面上加工切槽。该角度的取值范围可为0~180度，且总是与纵轴有关。

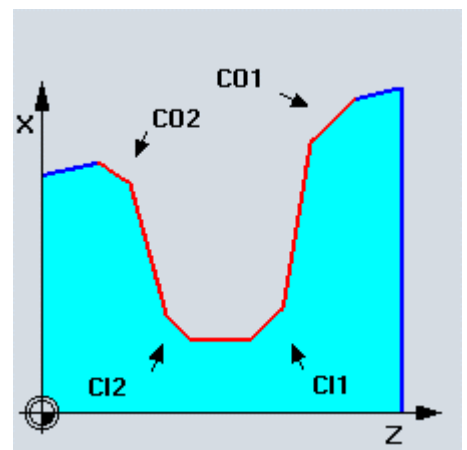
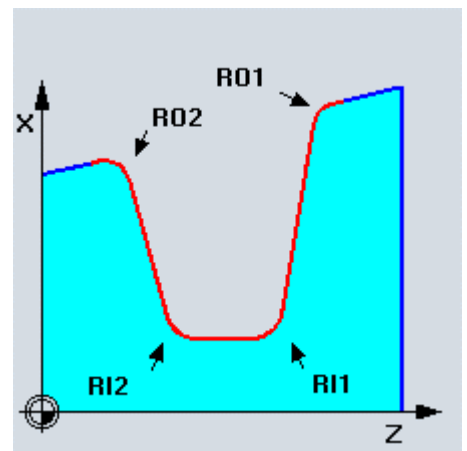
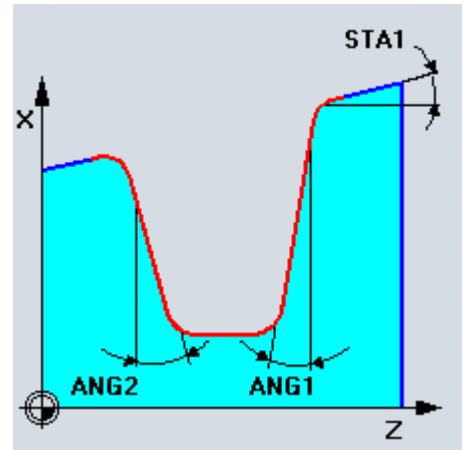
**ANG1 和 ANG2 (螺纹啮合角)**

通过分别规定的啮合角可描述非对称的切槽。该角度的取值范围可为 0 ~ 89.999 度

**RCO1, RCO2 和 RC11, RC12 (半径/棱角)**

通过在边缘或底部输入半径/棱角改变切槽形状。必须要注意的是，以正号输入半径，以负号输入棱角。根据参数VARI十位数确定计算编程棱角的方式。

- 当VARI<10时（十位=0）该参数量作为棱角长度（棱角用CHF编程）。
- 当VARI>10时，它作为减少的轨迹长度（棱角带CHR编程）。



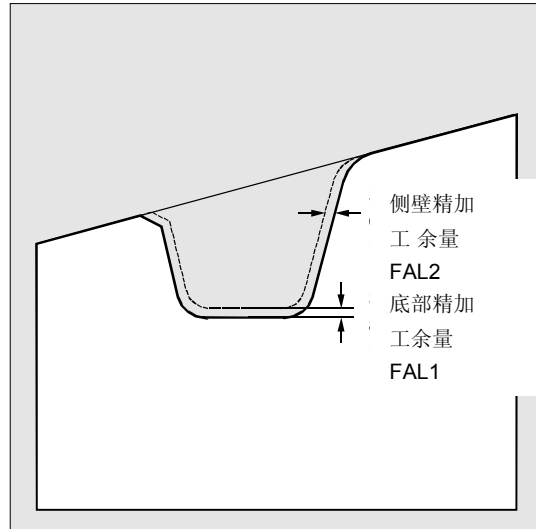
**FAL1 和 FAL2 (精加工余量)**

对于切槽底部和侧壁，可分别编程精加工余量。粗加工时，一直切削到该精加工余量。接着，沿着最终轮廓，用相同的刀具进行与轮廓平行的切削。

**IDEP (进刀深度)**

通过编程进刀深度，可以以多个深度进刀分配轴向平行的切槽。在每次进刀后，刀具以 1 毫米或在\_VRT 下编程的用于断屑的量回退。

在每种情况下，必须编程参数IDEP。

**DTB (停留时间)**

在切槽底部如此选择停留时间，以便至少可以进行一次主轴旋转。以秒为单位编程停留时间。

**VARI (加工方式)**

使用参数VARI的个位数，可以确定切槽的加工方式。其可接受插图中所示的值。

使用参数VARI的十位数，可以确定棱角计算的方式。

VARI 1...8: 棱角作为CHF计算

VARI 11...18: 棱角作为CHR计算

个位数的输入在循环辅助中分为三个选择区域:

1. 区域: 纵向/平面
2. 区域: 外部/内部
3. 区域: 起始点左侧/右侧 (在纵向) 或者上面/下面 (在平面)

VARI		在循环支持中选择
1/11		纵向, 外部, 左侧
5/15		纵向, 外部, 右侧
3/13		纵向, 内部, 左侧
7/17		纵向, 内部, 右侧
6/16		平面, 外部, 上面
8/18		平面, 外部, 下面
2/12		平面, 内部, 上面
4/14		平面, 内部, 下面

如果该参数有另一个值，则该循环中断，并发出报警 61002 “加工方式错误定义”。

由循环进行轮廓监控，以得出有意义的切槽轮廓。如果在切槽底部接触或切削半径/棱角，或者试图在一个平行于纵向轴运行的轮廓段平面插入，则情况就不是这样了。在这种情况下，循环中断，并发出报警 61603 “错误定义切槽形状”。

#### \_VRT（可变的退回位移）

在参数\_VRT下可以通过切槽的外部直径或者内部直径编程退回位移。

当\_VRT=0时（参数没有编程），退刀1毫米。退回位移始终与编程的尺寸系统英制/公制相关。

切槽时在每次深度进给之后，该退回位移在断屑时同时生效。



#### 其它说明

调用切槽循环之前，必须已激活两个刀沿的刀具。这两个刀沿的补偿值必须存储在两个相连的刀具D号码中，在循环调用前必须将它们先行激活。循环自行确定，它必须使用这两个刀具补偿值的哪一个来用于哪个加工步骤，并自行激活这些补偿值。结束循环后，再次激活循环调用前所编程的补偿号码。如果在循环调用时没有为刀具补偿编程D号码，则在发出报警 61000 “没有刀具补偿有效”之后，循环被中断。

使用循环设定数据 `_ZSD[4]` 可以影响第一个切槽之后的退回。

`_ZSD[4]=0` 表示用 `G1` 退回，与往常一样，

`_ZSD[4]=1` 表示用 `G0` 退回。

使用循环数据 `_ZSD[6]` 可以调整循环中刀具补偿的处理。

`_ZSD[6]=0` 在循环内部交换刀具补偿 `WZK`

(没有可定向的刀架)

`_ZSD[6]=1` 在循环内部不交换刀具补偿 `WZK`

(有可定向的刀架)



### 编程举例

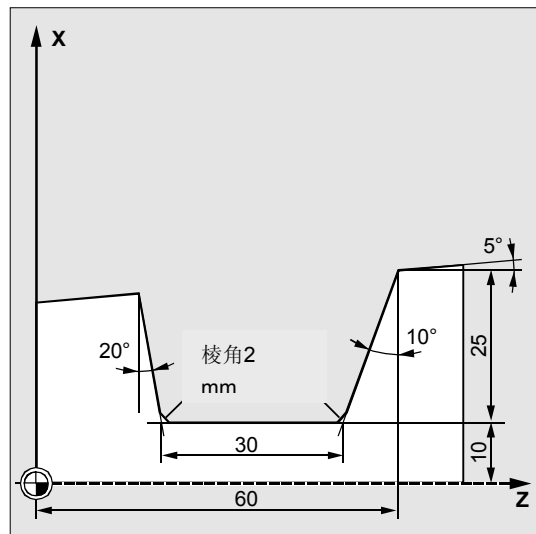
#### 切槽

通过该程序，可以在一个斜面上纵向外加工一个切槽。

起始点位于右侧 `X35Z60`。

循环使用刀具 `T1` 的刀具补偿值 `D1`

和 `D2`。应相应定义切槽刀具。



```
DEF REAL SPD=35, SPL=60, WIDG=30, -> ; 参数定义, 赋值
```

```
-> DIAG=25, STA1=5, ANG1=10, ANG2=20,
```

```
->
```

```
-> RCO1=0, RCI1=-2, RCI2=-2, RCO2=0, -
```

```
>
```

```
-> FAL1=1, FAL2=1, IDEP=10, DTB=1
```

```
DEF INT VARI=5
```

```
N10 G0 G18 G90 Z65 X50 T1 D1 S400 M3 ; 循环开始之前的起始点
```

```
N20 G95 F0.2 ; 确定工艺数值
```

```
N30 CYCLE93 (SPD, SPL, WIDG, DIAG, -> ; 循环调用
```

```
-> STA1, ANG1, ANG2, RCO1, RCO2, ->
```

```
-> RCI1, RCI2, FAL1, FAL2, IDEP, ->
```

```
-> DTB, VARI)
```

```
N40 G0 G90 X50 Z65 ; 下一个位置
```

```
N50 M02 ; 程序结束
```

-> 必须在一个程序段中编程

## 4.4 退刀槽循环 - CYCLE94



### 编程

CYCLE94 (SPD, SPL, FORM, \_VARI)



### 参数

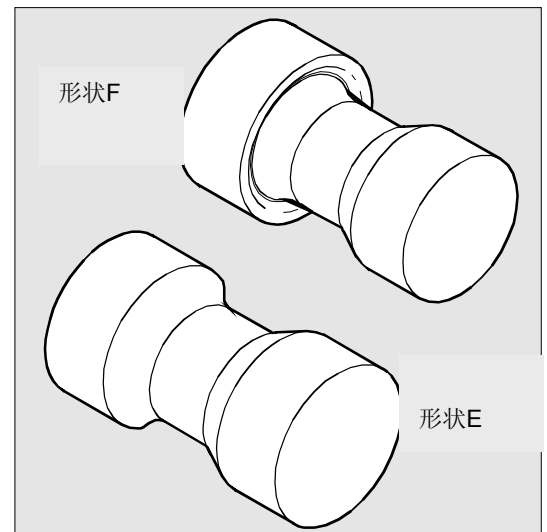
SPD	实数	平面轴中起始点（不输入符号）
SPL	实数	纵向轴中轮廓起始点（不输入符号）
FORM	字节	形状定义 值： E（用于形状 E） F（用于形状 F）
_VARI	整数	确定退刀槽的位置 值： 0 相应于刀沿位置 1...4 位置定义



### 功能

使用该循环，您可以根据DIN509形状E和F加工退刀槽，在成品直径 >3毫米时具有通常的应力要求。

为了加工螺纹退刀槽，可以使用另一个循环CYCLE96（参见章节4.6）。



### 工作流程

#### 循环开始之前到达的位置：

出发位置可以为一个任意位置，从这个出发位置可以返回运行到每个退刀槽，没有轮廓冲突。

#### 该循环产生以下的运动过程：

- 以G0返回到循环内部所求得起始点
- 根据当前的刀沿位置选择刀沿半径补偿，并且以循环调用之前编程的进给开始退刀槽轮廓运行
- 以G0退回到起始点，并按照G40撤消刀沿半径补偿的选择



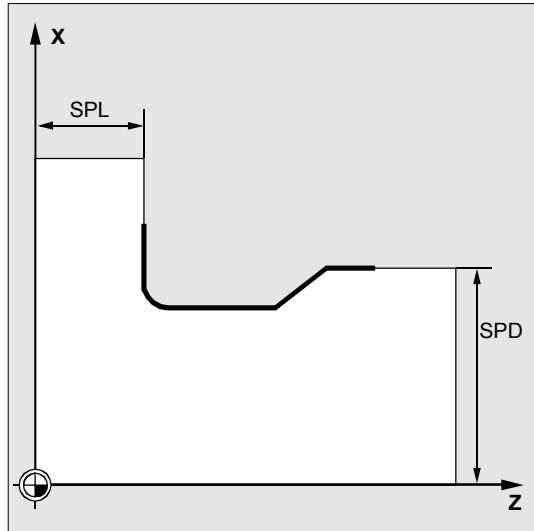
### 参数说明

#### SPD 和 SPL (起始点)

在参数SPD下规定退刀槽的成品直径。通过参数SPL确定纵向轴上的成品尺寸。

如果根据SPD编程的值产生一个最终直径<3毫米，则该循环中断，并发出报警

61601 “成品直径太小”。

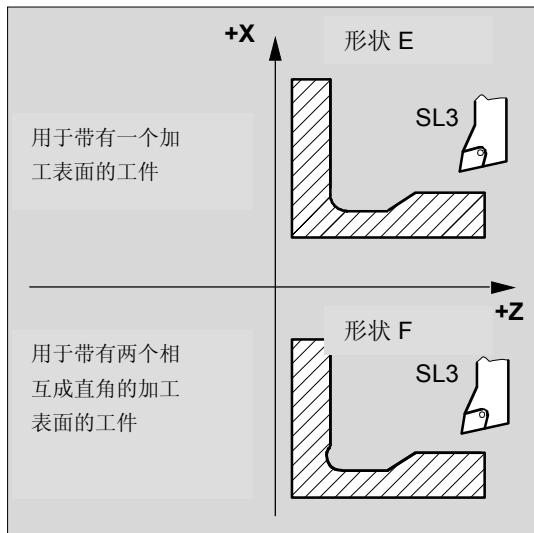


#### FORM (定义)

形状E和F在DIN509中定义，并可以通过这些参数确定。

如果该参数的值不同于E或者F，则该循环中断，并产生报警

61609 “形状错误定义”。



#### \_VARI (退刀槽位置)

使用参数\_VARI可以直接确定退刀槽的位置，或者由刀具刀沿位置产生退刀槽位置。

VARI=0: 相应于刀具的  
刀沿位置

循环通过激活的刀具补偿值自行确定刀具的刀沿位置 (SL)。接着循环只能用刀沿位置 1 ... 4 进行工作。

如果该循环识别的刀沿位置为 5...9，则显示报警

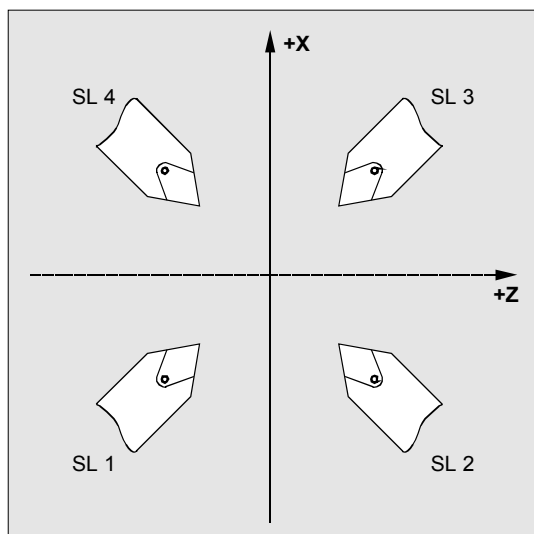
61608 “编程了错误的刀沿位置”

且循环被中断。

\_VARI=1...4: 定义退刀槽位置

循环自动确定其起始点。它位于纵向轴上距离最终直径2毫米和距离最终尺寸10毫米处。通过激活的刀具刀沿位置确定用于已编程的坐标值的起始点位置。

如果已在相应的刀具补偿值参数中规定了一个值，则可



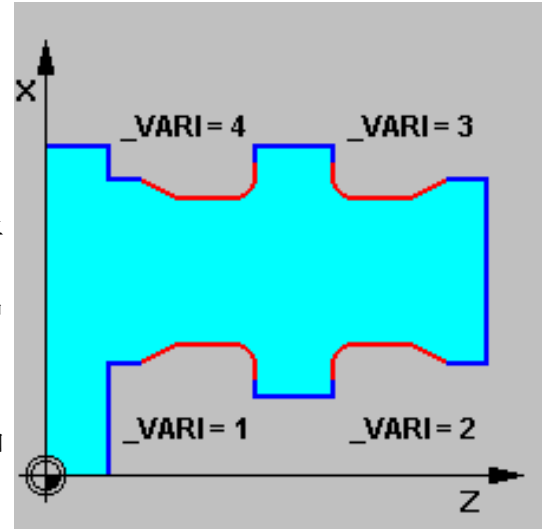


在循环中监控激活的刀具的自由切削角。如果确定，用所选的刀具无法加工出退刀槽形状，因为其自由切削角过小，在控制系统中显示信息

“退刀槽形状已改变”。尽管如此，加工进行继续。

在  $\_VARI \neq 0$  时适用于：

- 刀具的实际刀沿位置没有检测，也就是说只要工艺上有要求，所有的刀沿均可以使用
- 有些功能，比如适配器转换，可定向刀架，在循环中不需特别考虑—这主要是指用户已经了解机床的关系，位置已经正确的规定；
- 在循环中同样不需要特别考虑镜像，因为用户必须知道它是指什么。



#### 其它说明

在调用循环之前必须激活一个刀具补偿，否则在给出报警

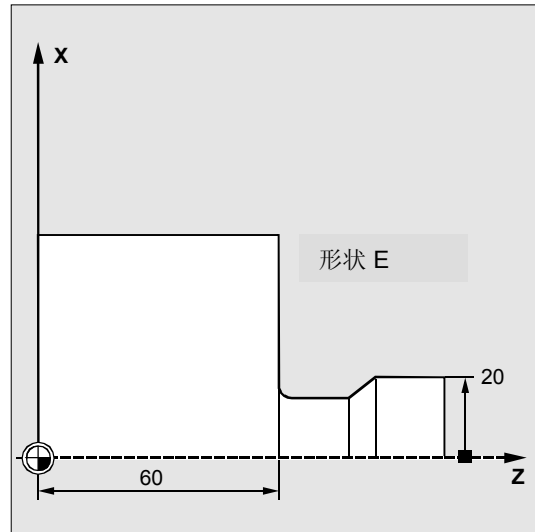
61000 “没有刀具补偿有效”之后中断循环。



## 编程举例

## 退刀槽形状E

使用该程序您可以加工E形状退刀槽。



```
N10 T25 D3 S300 M3 G18 G95 F0.3
```

; 确定工艺数值

```
N20 G0 G90 Z100 X50
```

; 选择起始位置

```
N30 CYCLE94 (20, 60, "E")
```

; 循环调用

```
N40 G90 G0 Z100 X50
```

; 返回到下一个位置

```
N50 M02
```

; 程序结束

## 4.5 切削循环 - CYCLE95



### 编程

CYCLE95 (NPP, MID, FALZ, FALX, FAL, FF1, FF2, FF3, VARI, DT, DAM, \_VRT)



### 参数

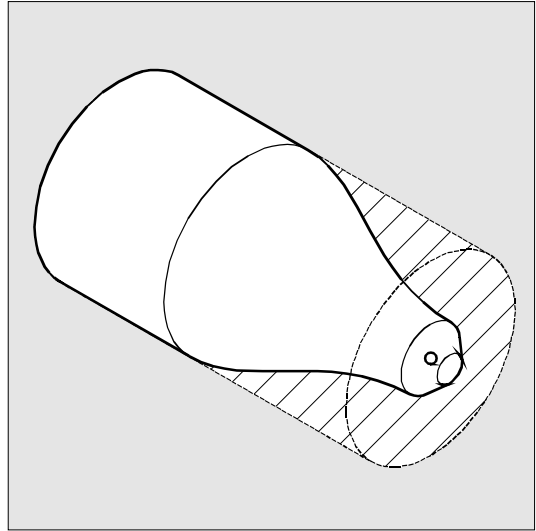
NPP	字符串	轮廓子程序名
MID	实数	进刀深度（不输入符号）
FALZ	实数	纵向轴中精加工余量（不输入符号）
FALX	实数	平面轴中精加工余量（不输入符号）
FAL	实数	与轮廓相符的精加工余量（不输入符号）
FF1	实数	粗加工进给，无底切
FF2	实数	在底切时插入进给
FF3	实数	精加工进给
VARI	整数	加工方式 值范围：1 ... 12 百位： 值： 0...带轮廓的拉削 没有余角，在轮廓上 已重叠拉削。这表示，通过 几个切削点拉削。 2...不带轮廓的拉削 拉削一直到先前的切削点 然后退刀。根据刀具半径和进刀深度（MID）的关系，此时可能 会有余角剩余。
DT	实数	粗加工时用于断屑的停留时间
DAM	实数	位移长度，每次粗加工切削断屑时均中断该长度
_VRT	实数	粗加工时从轮廓的退刀位移，增量（不输入符号）



## 功能

通过切削循环，您可以通过轴向平行的切削将一个坯件加工成为一个在子程序中编程的轮廓。在该轮廓中可以包含底切单元。通过该循环，可以在纵向加工和平面加工中内外加工轮廓。可自由选择工艺

（粗加工、精加工、综合加工）。粗加工轮廓时，由最大编程的切削深度产生轴向平行的切削，在达到一个切削点后，立即用轮廓产生的余角以与轮廓平行的方式同时切削。直至粗加工到编程的精加工余量。精加工方向与粗加工方向相同。刀具半径补偿由循环自动选择并撤消选择。



## 工作流程

### 循环开始之前到达的位置：

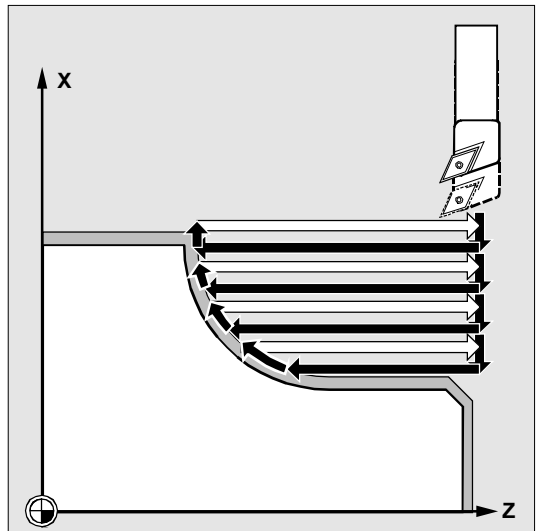
出发位置可以为一个任意位置，从这个出发位置可以返回运行到轮廓开始点，没有轮廓冲突。

### 该循环产生以下的运动过程：

- 循环起始点在内部计算，并以G0在两个轴中同时返回。

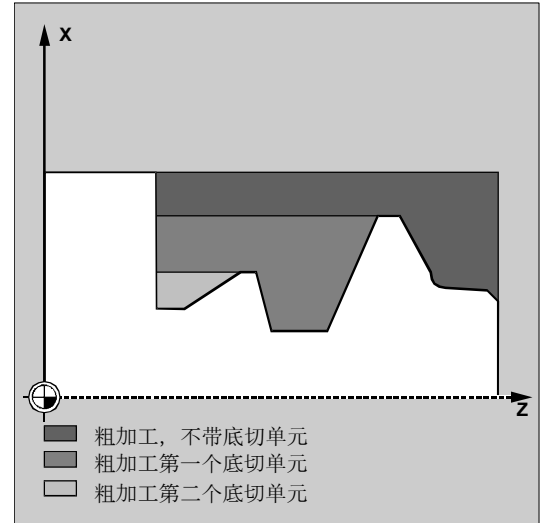
### 粗加工，不带底切单元：

- 轴向平行进刀到当前深度，这在内部进行计算，并以G0返回。
- 粗加工切削点与轴向平行，并以G1和进给FF1返回。
- 与轮廓平行，沿着轮廓+精加工余量以G1/G2/G3和FF1拉削，直至最后粗加工切削点。
- 在每个轴上退刀\_VRT下编程的量，并以G0返回。
- 重复该过程，直至到达加工截面的总深度。
- 在粗加工不带底切单元时，按轴方式退回到循环起始点。



**粗加工，带底切单元：**

- 用 G0  
以轴方式返回运行下一个底切的起始点。此时要注意一个附加的循环内部安全距离。
- 以 G1/G2/G3 和 FF2  
沿着轮廓+精加工余量并与轮廓平行进行进刀。
- 粗加工切削点与轴向平行，并以 G1 和进给 FF1 返回。
- 拉削直至最后的粗加工切削点。同第一个加工步骤，进行退刀并返回。奖
- 如果有其它的底切单元，则重复每个底切的运行。

**精加工：**

- 以 G0 在两个轴上同时返回所计算的循环起始点，并且选择刀沿半径补偿。
- 两个轴同时以 G0 运行一个位移：精加工余量+刀沿半径+1 毫米（起始点之前安全距离），然后以 G1 运行到轮廓起始点。
- 沿着轮廓以 G1/G2/G3 和 FF3 进行精加工。
- 两个轴以 G0 退回到起始点。



### 参数说明

#### NPP (名称)

在该参数下输入轮廓子程序名称。然而，轮廓子程序不允许是带有一个参数列表的子程序。

所有在编程说明中所描述的命名规范适用于轮廓子程序的名称。

文献: /PG/ 程序说明



切削轮廓也可以是一个调用的部分或者是任意其它程序的一个部分。该部分通过起始标签和结束标签或者程序段号码识别。在此程序名称和标签/程序段号码通过“:”识别。

举例:

NPP="KONTUR\_1 "

切削轮廓是完整的程序KONTUR\_1。

NPP="ANFANG:ENDE "

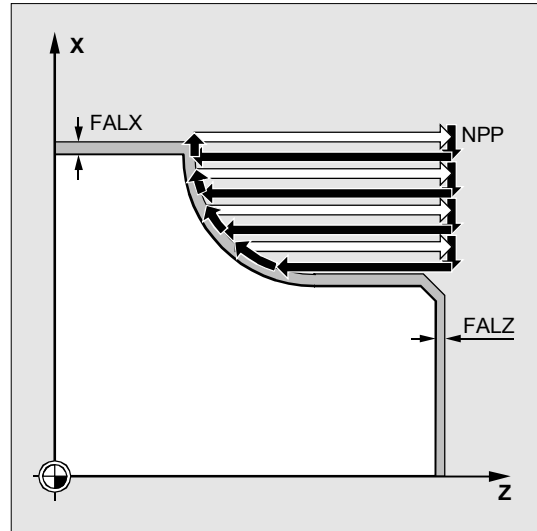
切削轮廓作为所调用程序的一部分定义，开始程序段为标签ANFANG，结束程序段为标签ENDE。

NPP="/\_N\_SPF\_DIR/\_N\_KONTUR\_1\_SPF:N130:N210 "

切削轮廓在程序 KONTUR\_1 的程序段 N130 至 N210 中定义。必须用完整的路径和扩展名写入程序名称，参见文献资料中的调用 (CALL) 说明: /PGA/编程说明 工作准备部分



如果这部分程序用程序段号定义，则必须要注意：在程序修改之后，并且后面带操作“重新编号”，则必须使程序段号与NPP的这部分程序相匹配。



### MID (进刀深度)

在此参数MID下定义粗加工过程最大可能的进刀深度。

自软件版本 4 起，该参数的评价与循环设定数据 `_ZSD[0]` 相关（参见章节4.2）。

该循环自行计算实际的进刀深度，用此进刀深度在粗加工时进行工作。

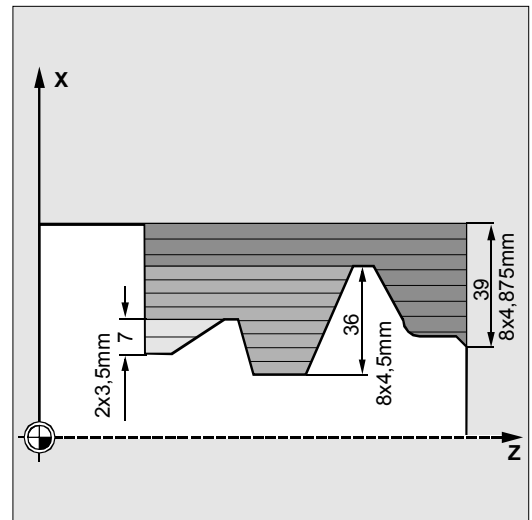
对于带有底切单元的轮廓，由循环将粗加工过程分成各个粗加工部分。对于每个粗加工部分，循环重新计算当前的进刀深度。该进刀深度总是位于编程的进刀深度和其二分之一值之间。利用粗加工部分的总深度和编程的最大进刀深度确定所需粗加工走刀的次数并平均分配待加工的总深度。以此获取最佳的切削条件。对于轮廓粗加工，得出上图中所描述的加工步骤。

举例说明当前进刀深度的计算：

加工步骤1有一个39毫米的总深度。在最大进刀深度为5毫米的情况下需要 8 个粗加工走刀。每次进刀为 4.875毫米。

在加工步骤 2 中同样有 8 个粗加工走刀，每次进刀 4.5毫米（总距离36毫米）。

在加工步骤3中，实际进刀3.5（总距离7毫米）毫米，粗加工两次。



### FAL、FALZ 和 FALX（精加工余量）

如果不同的轴规定不同的精加工余量，则可以通过参数 FALZ 和 FALX 规定精加工余量，用于粗加工；也可以通过参数 FAL 规定一个精加工余量，用于与轮廓相应的精加工。然后该值在两个轴中作为精加工余量计算在内。编程的值无法进行合理性检测。如果已为所有三个参数赋值，则所有这些精加工余量都由循环计算。然而，还是有必要确定一种或者其它类型和方式定义一个精加工余量。

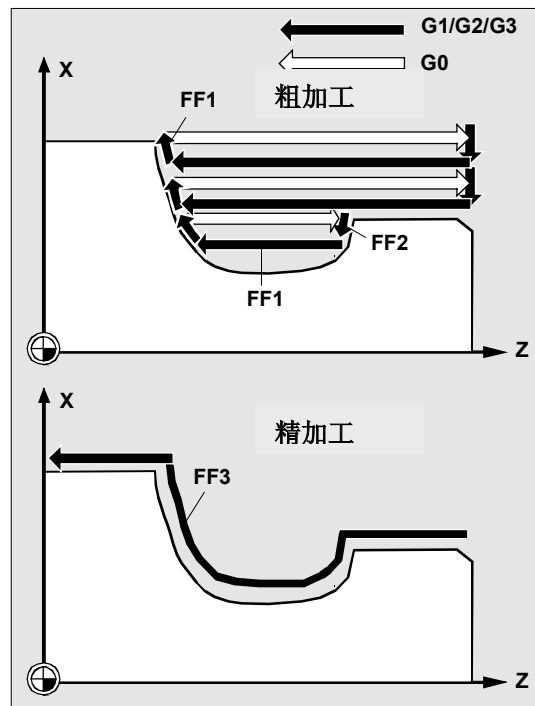
总是在到达该精加工余量后，才停止进行粗加工。

此时，在每次轴向平行的粗加工过程后立即以轮廓产生的余角以与轮廓平行的方式同时进行切削，以在粗加工结束后无需附加的余角走刀。如果尚未编程精加工余量，则在粗加工时一直切削到最终轮廓。

在加工方式为精加工时，不考虑精加工余量所编程的值，一直车削到最终尺寸。

### FF1, FF2 和 FF3（进给）

对于不同的加工步骤，您可以规定不同的进给，如同图中所示。





**VARI (加工方式)**

加工方式您可以按照以下所述调用:

加工   
(粗加工/精加工/综合加工)  
选择   
(纵向/平面)  
选择   
(外部/内部)

加工方式您可以从以下表中获得。

值	加工	选择	选择
1	粗加工	纵向	外部
2	粗加工	平面	外部
3	粗加工	纵向	内部
4	粗加工	平面	内部
5	精加工	纵向	外部
6	精加工	平面	外部
7	精加工	纵向	内部
8	精加工	平面	内部
9	综合加工	纵向	外部
10	综合加工	平面	外部
11	综合加工	纵向	内部
12	综合加工	平面	内部

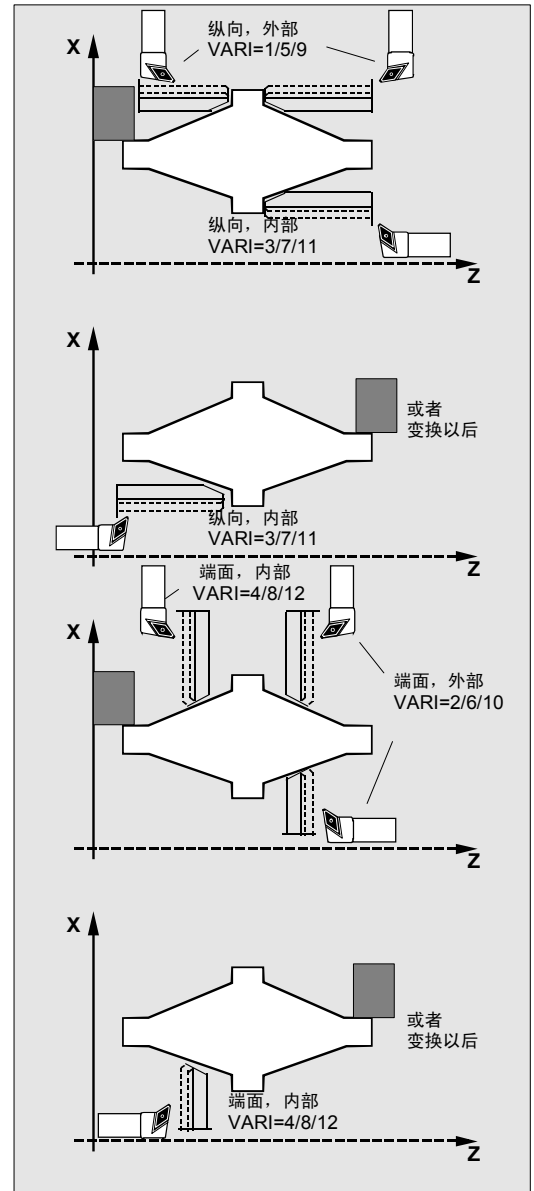


在纵向加工时始终在平面轴上进刀，在平面加工时则在纵向轴上进刀。

外加工表示，在负轴向方向上进刀。在内加工时，进刀以正轴向方向进行。

在通过CYCLE95 粗加工情况下，可以在轮廓上的“带有拉削”或者“不带拉削”之间选择。为此，在参数 VARI 中引入百位。

对于参数VARI可以进行合理性检测。当出现一个非法的值时，该循环中断，并发出报警61002“加工方式错误定义”。



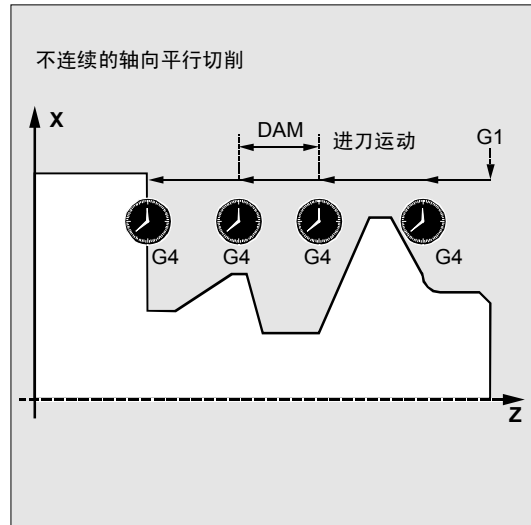
### DT 和 DAM（停留时间和位移长度）

借助于这两个参数，您可以在进行特定的行程后，实现中断每个粗加工走刀以达到断屑的目的。这两个参数仅对于粗加工有意义。在参数DAM中定义最大行程，在进行该行程后要求进行断屑。为此，在DT中可以编程一个在每个走刀中断点上执行的停留时间。如果没有规定用于走刀中断的行程（DAM=0），则在没有停留时间的情况下进行不间断的粗加工走刀。

### \_VRT（退刀位移）

在参数\_VRT下可以编程一个退刀量，在粗加工时在两个轴上退刀。

在\_VRT=0（参数未编程）情况下，退刀1毫米。退刀行程总是与所编程的英制或公制尺寸系统有关，即\_VRT=1 对于英制 → 编程，退刀 1 英寸。



### 其它说明

#### 轮廓定义

在一个子程序中编程轮廓，其名称作为参数规定。

在轮廓子程序中，在加工平面内，至少包含3个程序段，它们在两个轴上产生运动。

加工平面（G17、G18、G19）在循环调用之前在主程序中设定，或者根据机床中G

功能组的初始设置生效。无法在轮廓子程序中更改加工平面。

如果轮廓子程序很短，则给出报警10933“轮廓子程序包含太少的轮廓程序段”和报警61606“轮廓预处理时出错”，循环停止执行。

底切单元可以直接一起排列。

平面中不带运动的程序段可以不受限制写入。

在循环内部准备用于当前平面的前两根轴的所有运行程序段，因为仅这两根轴参与切削加工。其它轴的运动可以包含在轮廓子程序中，但是在循环运行期间，它们的运行行程无效。

作为轮廓中的几何尺寸，仅允许通过 **G0**、**G1**、**G2** 和 **G3** 进行直线编程和圆弧编程。此外，也可以编程用于倒圆和棱角的指令。如果在轮廓中对其它运动指令进行编程，则该循环中断，并发出报警

**10930** “切削轮廓中有不允许的插补方式”。

在实际加工平面中运行的第一个程序段内必须包含运行指令 **G0**、**G1**、**G2** 或者 **G3**，否则该循环中断，并发出报警

**15800** “错误的输出条件用于 **CONTPRON**”。

此外在有效的 **G41/G42** 时显示该报警。

轮廓的起始点是加工平面中轮廓子程序中第一个编程的位置。

在平面中，轮廓中最大可以带有运动的程序段数量取决于轮廓。底切数量原则上不受限制。

如果一个轮廓中包含的轮廓单元大于循环内部存储器所能接收的数量，则循环中断，并发出报警

**10934** “轮廓表溢出”。

然后，加工必须分为几个加工步骤，它们每次有自身的轮廓子程序，每个步骤可以单独调用该循环。

如果在一个轮廓子程序中，最大的直径不在编程的轮廓终点或者起始点，则由循环在加工结束处自动补充一个轴向平行的直线直至轮廓的最大值，轮廓的这个部分作为底切切削。

在轮廓子程序中的以下编程

- 半径补偿平面带 G17/G18/G19,
- 一个框架,
- 平面中进行切削的一个轴的运行作为定位轴, 以及
- 选择刀具半径补偿, 带G41/G42

导致循环停止, 并带报警

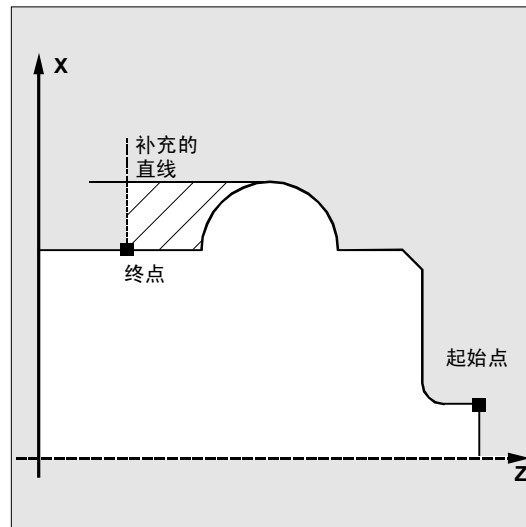
10931 “有缺陷的切削轮廓”。

### 轮廓方向

可自由选择切削轮廓编程的方向。在循环内部自动确定加工方向。在综合加工时, 用相同的方向精加工轮廓, 如同在粗加工时加工轮廓那样。

如果仅选择精加工, 则轮廓始终在编程的方向开始运行。

决定加工方向时, 要考虑到首先编程的轮廓点和最后编程的轮廓点。因此, 需要在第一个轮廓子程序的程序段中总是写入两个坐标。



### 轮廓监控

该轮廓提供一个轮廓监控，考虑以下几点：

- 有效刀具的自由切削角
- 圆弧的轮廓编程，张角 $>180$ 度

对于底切单元，在循环中检查是否可以用有效的刀具进行加工。如果循环识别到该加工可能导致轮廓损坏，则在发出报警

61604 “激活的刀具损坏编程的轮廓”后，循环中断。

如果自由切削角在刀具补偿中给出零，则该监控不进行。

如果在补偿中发现较大的圆弧，则显示报警 10931 “有缺陷的切削轮廓”。

伸出的轮廓不能通过 CYCLE95

进行加工。这类轮廓不受循环监控，因此不会发出报警信息。

### 起始点

循环自行确定加工起始点。起始点在进行深度进刀的轴上，距离轮廓尺寸为精加工余量+退刀位移

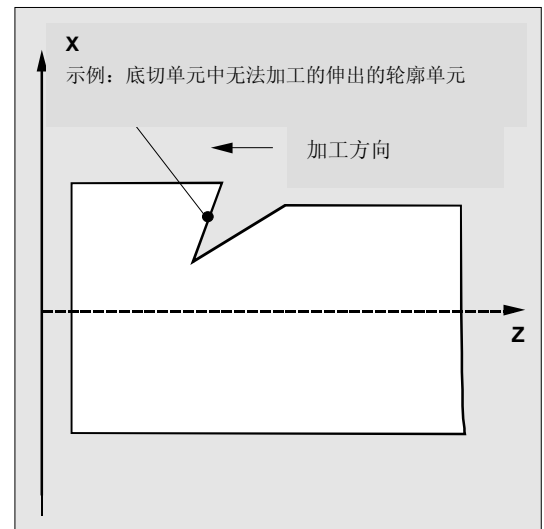
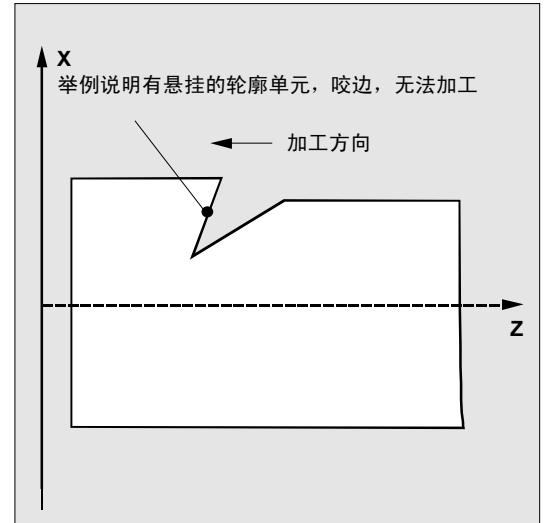
（参数\_VRT）。在其它轴中，起始点位于精加工余量 + 轮廓起始点前的 \_VRT。

在返回运行到起始点时，循环内部选择刀沿半径补偿。

因此，在调用循环之前必须选择好最后的点，保证不会有轮廓冲突，并且有足够的位置用于相应的补偿运动。

### 循环返回运行策略

由循环确定的起始点在粗加工时总是通过两根轴同时进行，在精加工时总是按轴方式返回运行。此时，在精加工时，进给轴首先运行。



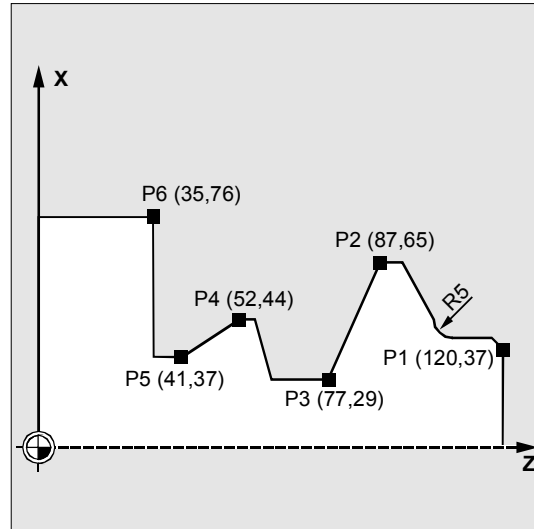


### 编程举例1

#### 切削循环

用于阐述供给参数的插图中指出的轮廓应该完全进行纵向外加工。规定了轴专用的精加工余量。在粗加工时不中断切削。最大进刀为 5毫米。

轮廓存放在一个分开的程序中。



DEF STRING[8] UPNAME	; 定义一个变量，用于轮廓名称
N10 T1 D1 G0 G18 G95 S500 M3 Z125 X81	; 调用之前的返回位置
UPNAME="KONTUR_1"	; 子程序名称赋值
N20 CYCLE95 (UPNAME, 5, 1.2, 0.6, , -> -> 0.2, 0.1, 0.2, 9, , , 0.5)	; 循环调用
N30 G0 G90 X81	; 重新返回起始位置
N40 Z125	; 轴方式运行
N50 M30	; 程序结束
PROC KONTUR_1	; 开始轮廓子程序
N100 G1 Z120 X37	; 轴方式运行
N110 Z117 X40	
N120 Z112	; 以半径5倒圆
N130 G1 Z95 X65 RND=5	; 轴方式运行
N140 Z87	
N150 Z77 X29	
N160 Z62	
N170 Z58 X44	
N180 Z52	
N190 Z41 X37	
N200 Z35	
N210 G1 X76	
N220 M17	; 子程序程序结束

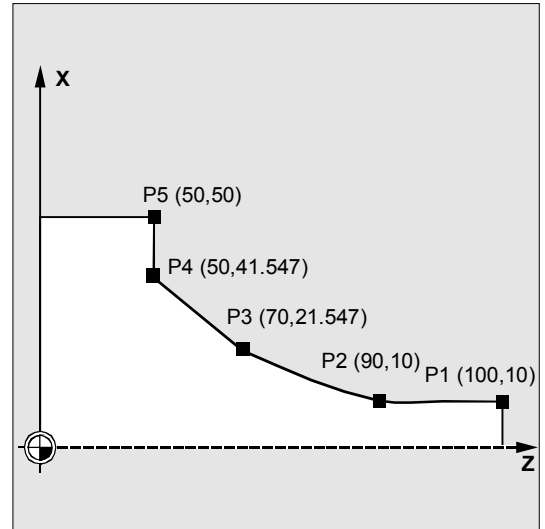
-> 必须在一个程序段中编程



## 编程举例2

### 切削循环

在调用的程序中定义切削轮廓。在切削循环后结束程序。



```

N110 G18 DIAMOF G90 G96 F0.8
N120 S500 M3
N130 T11 D1
N140 G0 X70
N150 Z60
N160 CYCLE95 ("ANFANG:ENDE",2.5,0.8, -> ; 循环调用
-> 0.8,0,0.8,0.75,0.6,1)
N170 M02
ANFANG:
N180 G1 X10 Z100 F0.6
N190 Z90
N200 Z=AC(70) ANG=150
N210 Z=AC(50) ANG=135
N220 Z=AC(50) X=AC(50)
ENDE:
N230 M02

```

## 4.6 螺纹退刀槽 - CYCLE96



## 编程

CYCLE96 (DIATH, SPL, FORM, \_VARI)



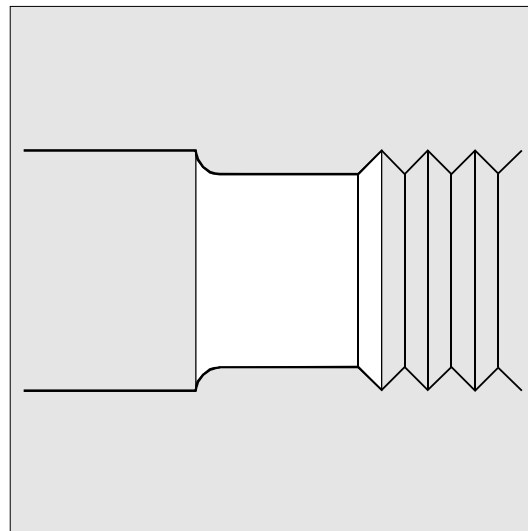
## 参数

DIATH	实数	螺纹额定直径
SPL	实数	纵向轴上轮廓起始点
FORM	字节	形状定义 值: A (用于形状 A) B (用于形状 B) C (用于形状 C) D (用于形状 D)
_VARI	整数	确定退刀槽的位置 值: 0 相应于刀沿位置 1..4 位置定义



## 功能

使用此循环您可以根据DIN76加工螺纹退刀槽，用于公制ISO螺纹零件。







### 工作流程

#### 循环开始之前到达的位置:

出发位置可以为一个任意位置，从这个出发位置可以返回运行到每个螺纹退刀槽，没有轮廓冲突。

#### 该循环产生以下的运动过程:

- 以G0返回到循环内部所求得起始点
- 选择相对于有效刀沿位置的刀具半径补偿。以循环调用前编程的进给离开运行退刀槽轮廓
- 以G0退回到起始点，并按照G40撤消刀沿半径补偿的选择



### 参数说明

#### DIATH (额定直径)

使用该循环您可以加工螺纹退刀槽，用于公制ISO螺纹M3到M68。

如果根据DIATH编程的值产生一个最终直径 <3毫米，则该循环中断，并发出报警

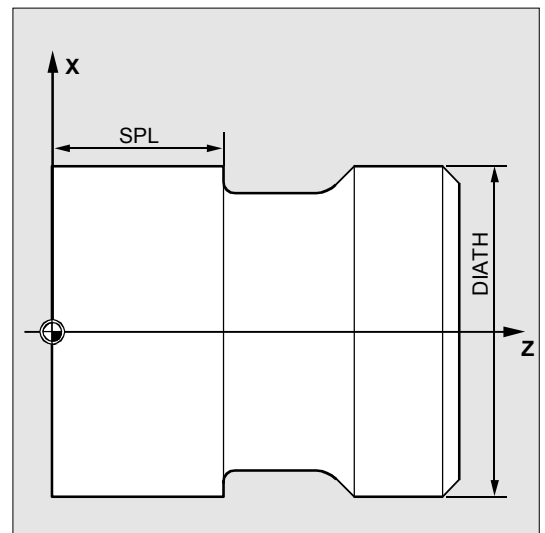
61601 “成品直径太小”。

如果该参数的值不同于DIN76部分1所规定的值，则该循环也在此中断，并产生报警

61001 “螺距错误定义”。

#### SPL (起始点)

使用参数SPL确定纵向轴的成品尺寸。



**FORM (定义)**

形状A和B的螺纹退刀槽定义为外螺纹，形状A为正常螺纹收尾，形状B为较短的螺纹收尾。

形状C和D的螺纹退刀槽用于内螺纹，形状C为正常螺纹收尾，形状D为较短的螺纹收尾。

如果该参数的值不同于A...D，则该循环中断，并产生报警

61609 “形状错误定义”。

循环内部自动选择刀具半径补偿。

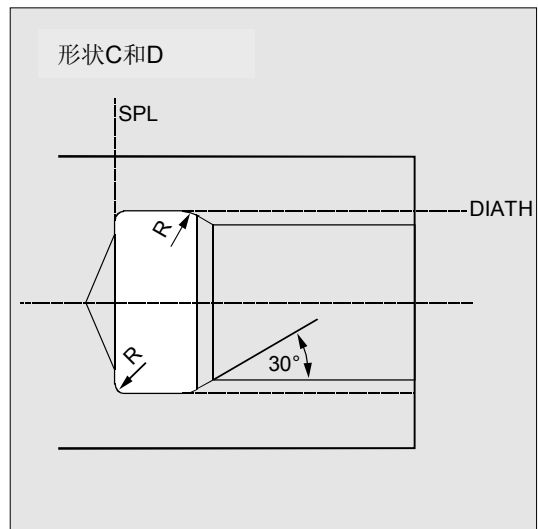
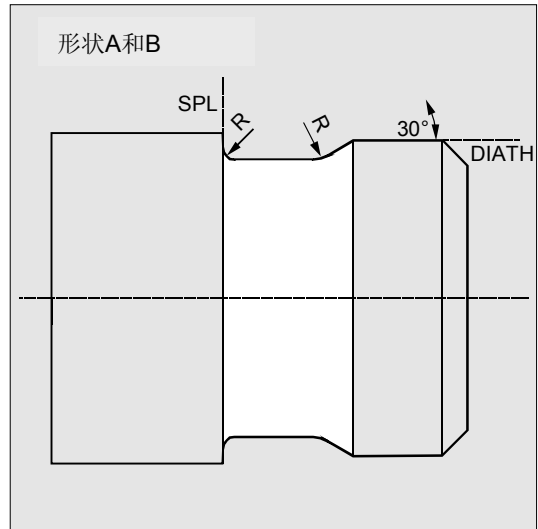
**\_VARI (退刀槽位置)**

使用参数\_VARI可以直接确定退刀槽的位置，或者由刀具刀沿位置产生退刀槽位置。参见\_VARI

(CYCLE94中)。

循环自动确定起始点，该起始点通过有效刀具的刀沿位置和螺纹直径确定。通过有效刀具的刀沿位置确定用于已编程的坐标值的起始点位置。

对于形状A和B，在循环中对激活刀具的自由切削角进行监控。如果已确定，则不能用所选的刀具加工退刀槽形状，显示信息“修改的退刀槽形状”，但是加工继续进行。

**其它说明**

在调用循环之前必须激活一个刀具补偿，否则在给出报警信息

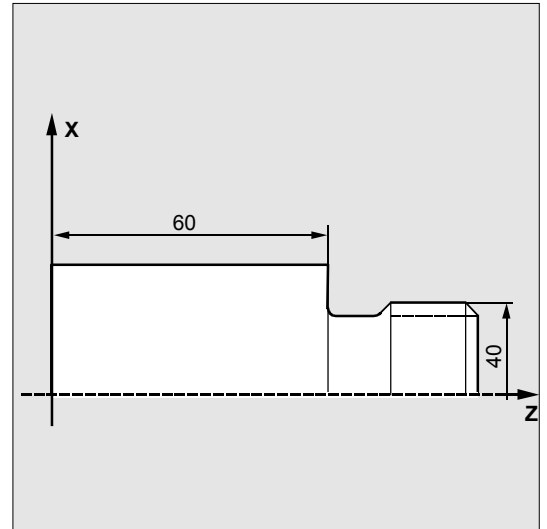
61000 “没有刀具补偿有效”之后循环停止。



### 编程举例

#### 螺纹退刀槽形状A

使用该程序您可以加工形状 A 螺纹退刀槽。



N10	D3 T1 S300 M3 G95 F0.3	; 确定工艺数值
N20	G0 G18 G90 Z100 X50	; 选择起始位置
N30	CYCLE96 (10, 60, "A")	; 循环调用
N40	G90 G0 X30 Z100	; 返回到下一个位置
N50	M30	; 程序结束

## 4.7 螺纹切削 - CYCLE97



## 编程

CYCLE97 (PIT, MPIT, SPL, FPL, DM1, DM2, APP, ROP, TDEP, FAL, IANG, NSP, NRC, NID, VARI, NUMT, \_VRT)



## 参数

PIT	实数	螺距值（不输入符号）
MPIT	实数	螺距，螺纹尺寸 值范围：3（用于 M3） .. 60（用于 M60）
SPL	实数	纵向轴上螺纹起始点
FPL	实数	纵向轴上螺纹终点
DM1	实数	起始点处螺纹的直径
DM2	实数	终点处螺纹的直径
APP	实数	导入位移（不输入符号）
ROP	实数	收尾位移（不输入符号）
TDEP	实数	螺纹深度（不输入符号）
FAL	实数	精加工余量（不输入符号）
IANG	实数	进给角度 值范围： "+"（齿面处齿面进刀） "- "（用于变换齿面进刀）
NSP	实数	第一个螺纹导程的起始点偏置（不输入符号）
NRC	整数	粗加工走刀次数（不输入符号）
NID	整数	空走刀次数（不输入符号）
VARI	整数	确定螺纹的加工方式，值范围： 1 ... 4
NUMT	整数	螺纹导程个数（不输入符号）
_VRT	实数	可变的退回位移（起始直径之上），增量（不输入符号）



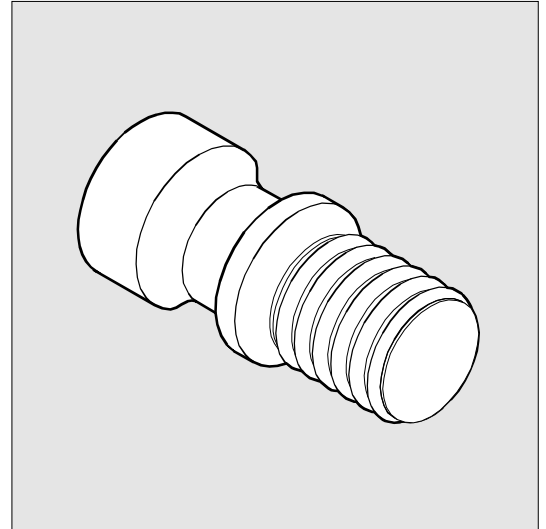
## 功能

通过螺纹切削循环，您可以在纵向和平面加工中以恒定螺距加工圆柱形和锥形外螺纹和内螺纹。螺纹可以是单头螺纹，也可以是多头螺纹。对于多头螺纹，依次对各个螺纹导程进行加工。

可以自动进刀，您可以选择每次走刀时不同的常数进刀量，也可以选择恒定的切削截面。

左旋螺纹或者右旋螺纹由主轴的旋向确定，这在循环调用之前必须编程。

在带有螺纹的运行程序段运行期间进给倍率无效。在建立螺纹期间，不允许更改主轴倍率。



**文献：** /PG/, 编程说明基础部分  
螺纹切削章节，带恒定  
螺距G33



使用该循环的前提条件就是速度可控制的主轴带位移测量系统。



## 工作流程

### 循环开始之前到达的位置：

出发位置可以为一个任意位置，从这个出发位置可以返回运行到编程的螺纹起始点+导入位移，没有轮廓冲突。

### 该循环产生以下的运动过程：

- 使用G0返回到循环内部计算的起始点，在第一个螺纹导程导入位移的开始处。
- 根据VARI下所确定的进刀方式进行进刀（粗加工）。
- 根据编程的粗加工走刀步数重复螺纹切削。
- 在后面的G33切削中，切削精加工余量。
- 根据空走刀步数重复切削。
- 对于每个其它的螺纹导程，重复整个运行过程。



## 参数说明

### PIT 和 MPIT (值和螺纹尺寸)

螺距是一个轴向平行的值，规定其不带符号。对于加工公制圆柱形螺纹，也可以通过参数MPIT将螺距规定作为螺纹尺寸（M3至M60）。应该有选择性地使用这两个参数。如果它们包含相互矛盾的值，则循环发出报警

61001 “螺距错误”

并停止。

### DM1 和 DM2 (直径)

使用该参数您可以确定螺纹起始点和终点处螺纹直径。在内螺纹时，这是中心孔直径。

### SPL、FPL、APP 和 ROP

#### 的关系 (起始点、终点、导入位移和收尾位移)

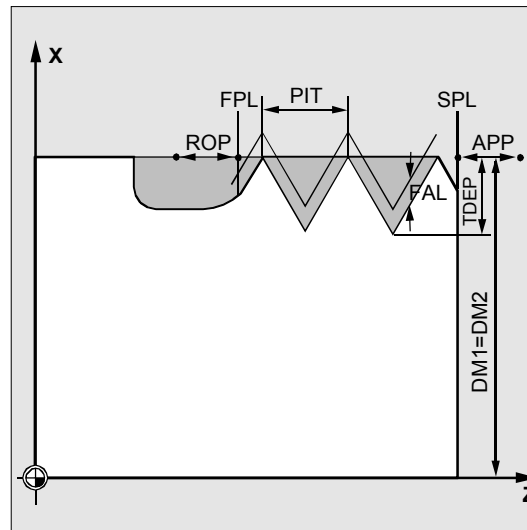
编程的起始点 (SPL) 以及结束点 (FPL) 描述了螺纹的原始出发点。循环中所使用的起始点在起始点导入位移 APP 之前，终点相应地在编程的终点收尾位移 ROP 之后。在平面轴中由循环确定的起始点总是在编程的螺纹直径上方 1 毫米处。退刀平面在控制系统内部自动形成。

### TDEP、FAL、NRC 和 NID

#### 关系 (螺纹深度、精加工余量、走刀步数)

编程的精加工余量轴向平行，由规定的螺纹深度 TDEP 减去，剩下的余量分割为粗加工走刀。循环自己计算各个实际进刀深度，取决于参数 VARI。

在分解进刀中待加工的螺纹深度时，通过恒定的切削截面使所有粗加工切削时的切削压力保持恒定。接着以不同的进刀深度值进行进刀。



第二种方式是，以恒定的进刀深度分配整个螺纹深度。此时，切削截面一次切削比一次切削大，然而对于较小的螺纹深度值，该工艺可以获得更好的切削条件。

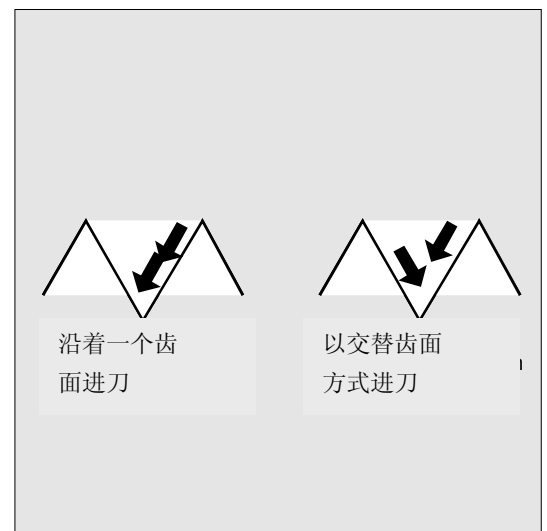
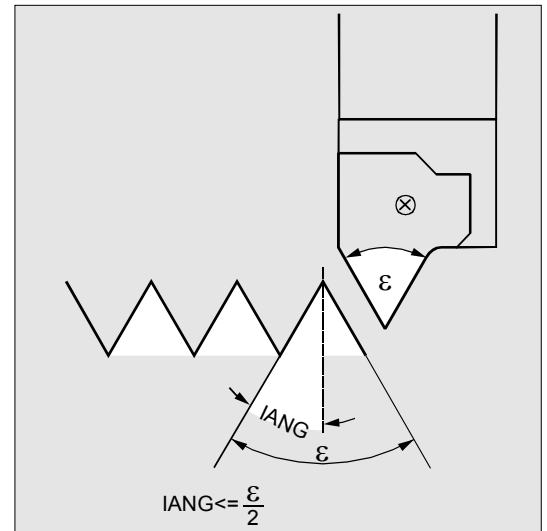
粗加工后，以一次切削切除精加工余量FAL。接着执行参数 NID 下编程的空走刀。

### IANG (进刀角度)

通过参数 IANG

确定螺纹中进刀的角度。如果在螺纹中以垂直于切削方向的角度进刀，则必须将该参数值置零。也就是说，可以在参数列表中删除该参数，因为在这种情况下，自动预设为零。如果沿着齿面进刀，则该参数绝对值最大允许为刀具啮合角的一半。

该参数符号确定进刀的执行情况。在正值情况下，总是在相同的齿面上进刀；在负值情况下，在两侧齿面上交替进刀。交替齿面的进刀方式仅可用于圆柱形螺纹。对于锥形螺纹，如果 IANG 值仍然为负，则由循环沿着一个齿面进行齿面进刀。



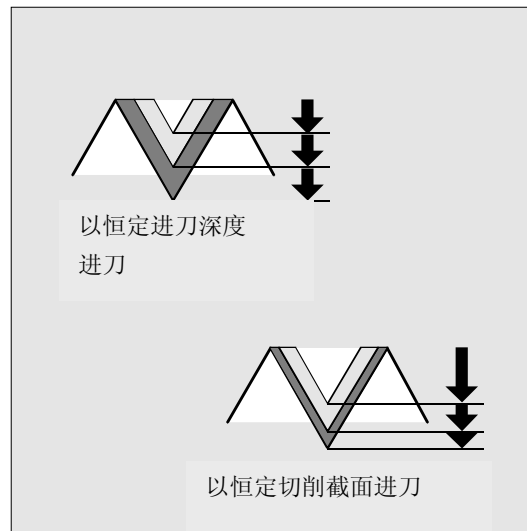
**NSP (起始点偏置)**

在该参数下可以对角度值编程，该角度值确定旋转部分圆周上的第一个螺纹导程的切削点。这里与起始点偏置位置有关。参数取值的范围可以为0.0001~+359.9999度。如果没有给定起始点偏置位置或者在参数列表中省略了该参数，则第一个螺纹导程自动在零度标记处开始。

**VARI (加工方式)**

通过参数 VARI

确定，是否要采用外加工或内加工，以及在粗加工涉及及到进刀时采用何种工艺进行加工。参数 VARI 可以采用 1 至 4 之间的数值，其含义分别如下：



值	外部/内部	恒定进刀/恒定切削截面
1	外部	恒定进刀
2	内部	恒定进刀
3	外部	恒定切削截面
4	内部	恒定切削截面

如果参数 VARI 编程一个其它值，  
则循环中断并发出报警 61102  
“加工方式错误定义”。



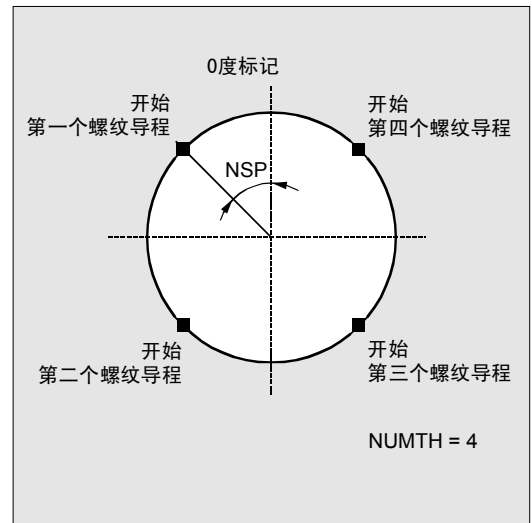
### NUMT (螺纹导程个数)

#### 通过参数 NUMT

确定多头螺纹的螺纹导程数量。对于单头螺纹，将该参数置零或者可以在参数列表中完全清除。

螺纹导程可以均匀地分布在车削件圆周上，第一个螺纹导程由参数NSP确定。

如果多头螺纹中其螺纹导程不均匀地分布在圆周上，则每个螺纹导程均在编程的相应的起始点偏置位置调用。



### \_VRT (可变的退回位移)

#### 在参数 \_VRT

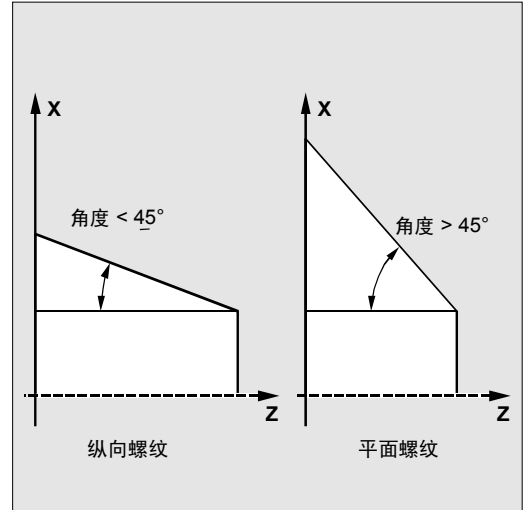
下，退回位移可以通过螺纹输出端直径编程。

当\_VRT=0时（参数没有编程），退刀1毫米。退回位移总是与编程的尺寸系统（英制或者公制）有关。

### 其它说明

#### 区分纵向螺纹- 和平面螺纹

由循环自行作出决定，是否要加工纵向螺纹或者平面螺纹。这取决于圆锥角度，在该圆锥上切削螺纹。如果圆锥上的角度为 $\leq 45^\circ$ 度，则加工纵轴中的螺纹，否则加工平面螺纹。

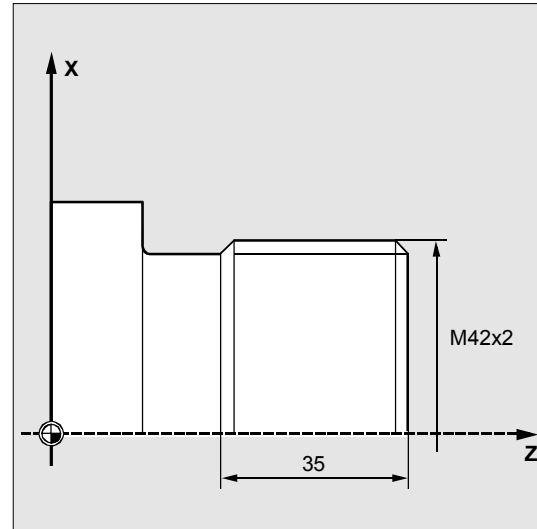




### 编程举例

#### 螺纹切削

借助该程序，可以通过齿面进刀加工一个公制外螺纹 M42x2。以恒定的切削截面进刀。在螺纹深度为1.23毫米时分5次粗切削，在没有精加工余量情况下，执行 5 次粗加工走刀。结束后，进行 2 次空走刀。



DEF REAL MPIT=42, SPL=0, FPL=-35,	; 参数定义, 赋值
DM1=42, DM2=42, APP=10, ROP=3,	
TDEP=1.23, FAL=0, IANG=30, NSP=0	
DEF INT NRC=5, NID=2, VARI=3, NUMT=1	
N10 G0 G18 G90 Z100 X60	; 选择起始位置
N20 G95 D1 T1 S1000 M4	; 确定工艺数值
N30 CYCLE97 ( , MPIT, SPL, FPL, DM1, ->	; 循环调用
-> DM2, APP, ROP, TDEP, FAL, IANG, ->	
-> NSP, NRC, NID, VARI, NUMT)	
N40 G90 G0 X100 Z100	; 返回到下一个位置
N50 M30	; 程序结束

-> 必须在一个程序段中编程

## 4.8 螺纹链 - CYCLE98



### 编程

CYCLE98 (PO1, DM1, PO2, DM2, PO3, DM3, PO4, DM4, APP, ROP, TDEP, FAL, IANG, NSP, NRC, NID, PP1, PP2, PP3, VARI, NUMT, \_VRT)



### 参数

PO1	实数	纵向轴上螺纹起始点
DM1	实数	起始点处螺纹的直径
PO2	实数	纵向轴中第一个中间点
DM2	实数	第一个中间点的直径
PO3	实数	第二个中间点
DM3	实数	第二个中间点的直径
PO4	实数	纵向轴上螺纹终点
DM4	实数	终点处直径
APP	实数	导入位移（不输入符号）
ROP	实数	收尾位移（不输入符号）
TDEP	实数	螺纹深度（不输入符号）
FAL	实数	精加工余量（不输入符号）
IANG	实数	进给角度 值范围： “+”（用于齿面处齿面进刀） “ - ”（用于变换齿面进刀）
NSP	实数	第一个螺纹导程的起始点偏置（不输入符号）
NRC	整数	粗加工走刀次数（不输入符号）
NID	整数	空走刀次数（不输入符号）
PP1	实数	螺距1值（不输入符号）
PP2	实数	螺距2值（不输入符号）
PP3	实数	螺距3值（不输入符号）
VARI	整数	确定螺纹的加工方式 值范围： 1 ... 4
NUMT	整数	螺纹导程个数（不输入符号）
_VRT	实数	可变的退回位移（起始直径之上），增量（不输入符号）



## 功能

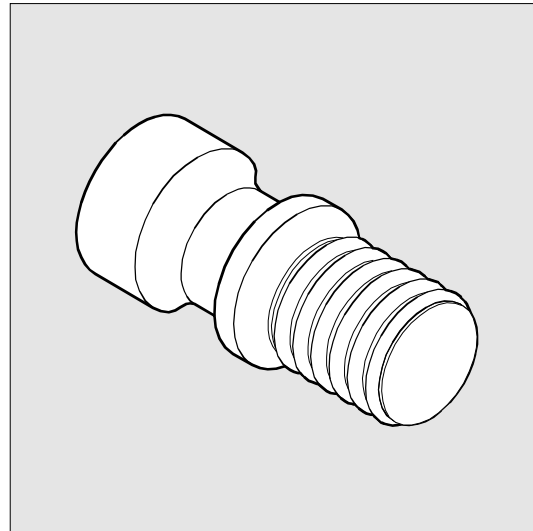
用此循环可以加工几个相连的圆柱螺纹或者圆锥螺纹，具有恒定的螺距，在纵向和平面加工，其螺距可以不同。

螺纹可以是单头螺纹，也可以是多头螺纹。对于多头螺纹，依次对各个螺纹导程进行加工。

可以自动进刀，您可以选择每次走刀时不同的常数进刀量，也可以选择恒定的切削截面。

左旋螺纹或者右旋螺纹由主轴的旋向确定，这在循环调用之前必须编程。

在带有螺纹的运行程序段运行期间进给倍率无效。在建立螺纹期间，不允许更改主轴倍率。



**文献：** /PG/, 编程说明基础部分

章节：螺纹切削，带恒定螺距，G33



## 工作流程

### 循环开始之前到达的位置：

出发位置可以为一个任意位置，从这个出发位置可以返回运行到编程的螺纹起始点+导入位移，没有轮廓冲突。

### 该循环产生以下的运动过程：

- 使用G0返回到循环内部计算的起始点，在第一个螺纹导程导入位移的开始处。
- 根据VARI下所确定的进刀方式进行进刀（粗加工）。
- 根据编程的粗加工走刀步数重复螺纹切削。
- 在后面的切削中，用G33切削精加工余量。
- 根据空走刀步数重复切削。
- 对于每个其它的螺纹导程，重复整个运行过程。



### 参数说明

#### PO1 和 DM1 (起始点和直径)

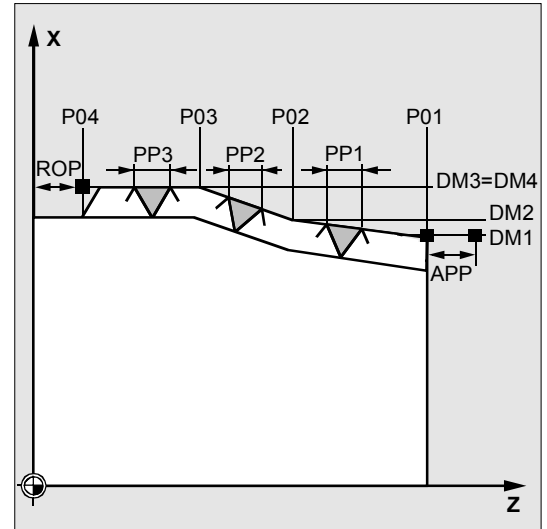
使用这些参数您可以确定螺纹链的原始起始点。由循环自己计算的起始点(以G0返回到开始)位于编程的起始点之前的导入位移处(前一页图中起始点A)。

#### PO2, DM2 和 PO3, DM3 (中间点和直径)

使用这些参数您可以在螺纹中确定两个中间点。

#### PO4 和 DM4 (中间点和直径)

在参数PO4和DM4下编程螺纹的原点。



在内螺纹时, 中心孔直径为DM1...DM4。

#### APP 和 ROP 的关系 (导入位移, 收尾位移)

循环中所使用的起始点在起始点导入位移APP之前, 终点相应地在编程的终点收尾位移ROP之后。

在平面轴中由循环确定的起始点总是在编程的螺纹直径上方1毫米处。退刀平面在控制系统内部自动形成。

#### TDEP、FAL、NRC 和 NID

#### 的关系 (螺纹深度、精加工余量、粗加工走刀次数和空走刀次数)

编程的精加工余量由规定的螺纹深度TDEP减去, 剩下的余量分割为粗加工走刀。循环自行计算各个实际进刀深度, 取决于参数VARI。在分割进刀中待加工的螺纹深度时, 通过恒定的切削截面使所有粗加工切削时的切削压力保持恒定。接着以不同的进刀深度值进行进刀。

第二种方式是, 以恒定的进刀深度分配整个螺纹深度。此时, 切削截面一次切削比一次切削大, 然而对于较小的螺纹深度值, 该工艺可以获得更好的切削条件。

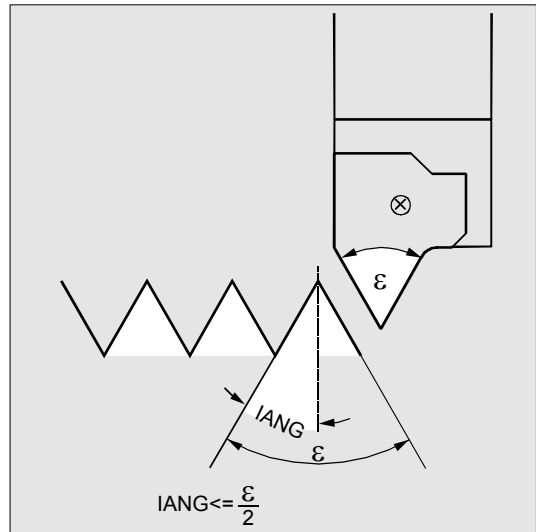
粗加工后，以一次切削切除精加工余量FAL。接着执行参数 NID 下编程的空走刀。

### IANG（进刀角度）

通过参数 IANG

确定螺纹中进刀的角度。如果在螺纹中以垂直于切削方向的角度进刀，则必须将该参数值置零。也就是说，可以在参数列表中删除该参数，因为在这种情况下，自动预设为零。如果沿着齿面进刀，则该参数绝对值最大允许为刀具啮合角的一半。

该参数符号确定进刀的执行情况。在正值情况下，总是在相同的齿面上进刀；在负值情况下，在两侧齿面上交替进刀。交替齿面的进刀方式仅可用于圆柱形螺纹。对于锥形螺纹，如果 IANG 值仍然为负，则由循环沿着一个齿面进行齿面进刀。



### NSP（起始点偏置）

在该参数下可以对角度值编程，该角度值确定旋转部分圆周上的第一个螺纹导程的切削点。这里与起始点偏置位置有关。参数取值的范围可以为

0.0001~+359.9999度。如果没有给定起始点偏置位置或者在参数列表中省略了该参数，则第一个螺纹导程自动在零度标记处开始。

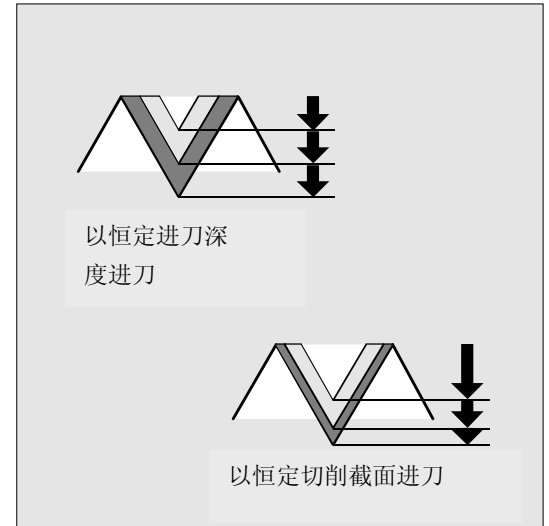
### PP1、PP2 和 PP3 (螺距)

通过这些参数确定螺纹链中三个截面中的螺距。  
此时，不带符号地输入螺距值作为轴向平行的值。

### VARI (加工方式)

通过参数 VARI

确定，是否要采用外加工或内加工，以及在粗加工涉及到进刀时采用何种工艺进行加工。参数 VARI 可以采用 1 至 4 之间的数值，其含义分别如下：



值	外部/内部	恒定进刀/恒定切削截面
1	外部	恒定进刀
2	内部	恒定进刀
3	外部	恒定切削截面
4	内部	恒定切削截面

如果参数 VARI

编程一个其它值，则循环中断并发出报警  
61002 “加工方式错误定义”。

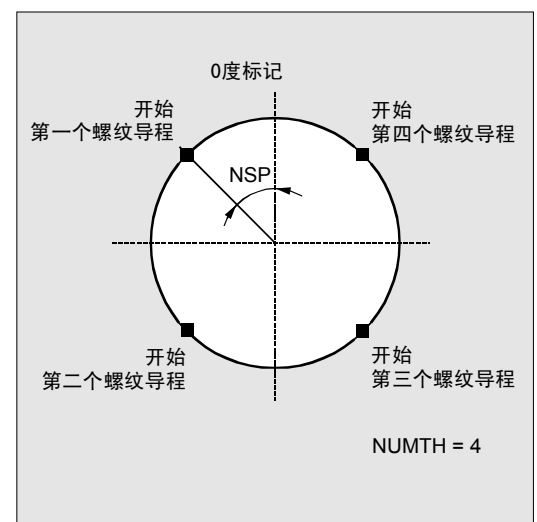
### NUMT (螺纹导程个数)

通过参数 NUMT

确定多头螺纹的螺纹导程数量。对于单头螺纹，将该参数置零或者可以在参数列表中完全清除。

螺纹导程可以均匀地分布在车削件圆周上，第一个螺纹导程由参数 NSP 确定。

如果多头螺纹中其螺纹导程不均匀地分布在圆周上，则每个螺纹导程均在编程的相应的起始点偏置位置调用。



**\_VRT (可变的退回位移)**

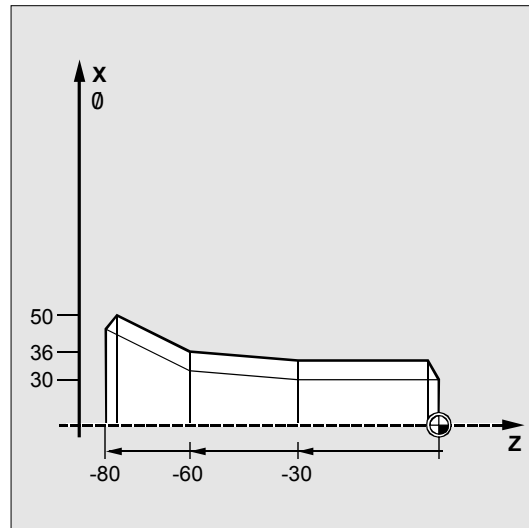
在参数\_VRT下，退回位移可以通过螺纹输出端直径编程。

当\_VRT=0时（参数没有编程），退回位移为1毫米。退回位移总是与编程的尺寸系统（英制或者公制）有关。

**编程举例****螺纹链**

通过这些参数，可借助一个圆柱形螺纹加工一个螺纹链。以垂直于螺纹的方式进行进刀，既未编程精加工余量，也未编程起始点偏置。进行5次粗加工走刀和一个空走刀。

加工方式规定为纵向，外部并带恒定切削截面。



N10 G18 G95 T5 D1 S1000 M4	; 确定工艺数值
N20 G0 X40 Z10	; 返回运行到出发位置
N30 CYCLE98 (0, 30, -30, 30, -60, -> -> 36, -80, 50, 10, 10, 0.92, , , , -> -> 5, 1, 1.5, 2, 2, 3, 1)	; 循环调用
N40 G0 X55	; 轴方式运行
N50 Z10	
N60 X40	
N70 M30	; 程序结束

-> 必须在一个程序段中编程



## 4.9 螺纹精整



螺纹切削循环CYCLE97和CYCLE98提供螺纹精加工。



### 功能

由于断刀或者测量工作会产生螺纹导程的角度偏置，这可以通过功能“螺纹精整”加以考虑并补偿。

该功能可以在JOG（手动）运行方式下，在操作区“加工”中执行。

每次循环由同步螺纹导程时存储的数据进行计算，得到一个附加的偏移角用于螺纹，它附加到编程的起始点偏置中。

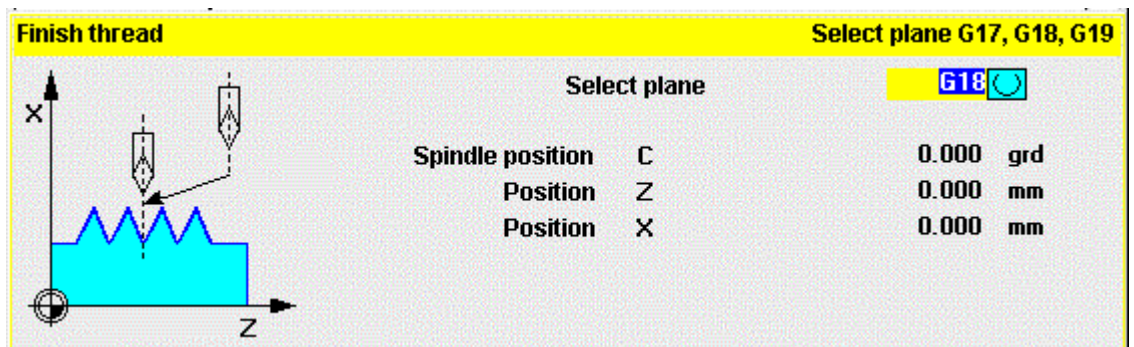
### 前提条件

该程序用于螺纹精整，应选择通道运行；参与的轴必须已经有基准。通道处于复位状态，主轴则停止。



### 工作流程

- JOG下选择操作区“加工”。
- 按软键螺纹精整\_
  - 打开该功能的屏幕窗口



- 使用该螺纹车刀，插入螺纹导程。
- 当螺纹车刀精确地位于螺纹导程中时，按软键“同步点”。
- 按软键“停止”返回到上一级软键菜单，不执行功能，并且不存储值到NC。

- 使用软键“确认”，所有的值接收到NC的GUD中。
- 接着刀具空运行，并进入开始位置。
- 选择”自动方式“，在调用螺纹循环之前定位程序指针，带程序段搜索。
- 用NC启动键启动程序。



### 辅助功能

使用软键“删除”可以删除以前输入的数值。

如果通道中有几个主轴，则在屏幕窗口中显示另一个选择区域，在此屏幕窗口可以选择加工螺纹的主轴。



### 开机调试

螺纹精整要求在 JOG

主窗口进行操作，对此必须激活文件 MA\_JOG.COM

中软键 HS8 “螺纹精整”。

- 用于HMI **Advanced**（高级）的螺纹精整开机调试：  
文件MA\_JOG.COM必须打开，并且在下面的行中去  
除分号：
  - ; HS8= (\$80720,,se1)
  - ; PRESS (HS8)
  - ; LM (“GENS”, “drehen2.com”)
  - ; END\_PRESS

该文件位于目录“标准循环”中，其后该软键生效。之后，必须重新启动 HMI 。

- 用于HMI **Embedded**（内置）的螺纹精整开机调试：  
为此，必须打开文件 **COMMON.COM** 并删除  
**SC108**前的“; ”。文件位于目录“用户循环”  
中。之后，必须重新启动 HMI 。

## 4.10 扩展的切削循环 - CYCLE950



扩展的切削循环是一个选项。

它要求HMI Advanced（高级）NCK中软件版本SW6。



### 编程

```
CYCLE950 (_NP1, _NP2, _NP3, _NP4, _VARI, _MID, _FALZ, _FALX, _FF1, _FF2,
_FF3, _FF4, _VRT, _ANGB, _SDIS, _NP5, _NP6, _NP7, _NP8, _APZ, _APZA, _APX,
_APXA, _TOL1)
```



### 参数

_NP1	字符串	成品轮廓的子程序名称
_NP2	字符串	标签/程序段号，成品轮廓起始，可选（由此可以定义轮廓部分）
_NP3	字符串	标签/程序段号，成品轮廓结束，可选（由此可以定义轮廓部分）
_NP4	字符串	待生成的切削程序名称
_VARI	整数	加工方式：（不输入符号） 个位： 值： 1... 纵向 2... 平面 3... 轮廓平行 十位： 值： 1... 编程的进刀方向 X- 2... 编程的进刀方向 X+ 3... 编程的进刀方向 Z- 4... 编程的进刀方向 Z+ 百位： 值： 1... 粗加工 2... 精加工 3... 综合加工 千位： 值： 1... 带拉削 2... 不带拉削（退刀） 万位 值： 1... 加工底切 2... 不加工底切 十万位 值： 1... 编程的加工方向 X- 2... 编程的加工方向 X+ 3... 编程的加工方向 Z- 4... 编程的加工方向 Z+
_MID	实数	进刀深度（不输入符号）
_FALZ	实数	纵向轴中精加工余量（不输入符号）

## 4.10 扩展的切削循环 - CYCLE950

_FALX	实数	平面轴中精加工余量（不输入符号）
_FF1	实数	纵向粗加工进给
_FF2	实数	平面粗加工进给
_FF3	实数	精加工进给
_FF4	实数	轮廓过渡单元进给（半径，棱角）
_VRT	实数	粗加工时退刀位移，增量（不输入符号）
_ANGB	实数	粗加工时退刀角度
_SDIS	实数	绕行障碍时的安全距离，增量
_NP5	字符串	毛坯轮廓的轮廓程序名称
_NP6	字符串	标签/程序段号，毛坯轮廓起始，可选（由此可以定义轮廓部分）
_NP7	字符串	标签/程序段号，毛坯轮廓结束，可选（由此可以定义轮廓部分）
_NP8	字符串	更新的毛坯轮廓的轮廓程序名称
_APZ	实数	轴向数值，用于纵向轴的毛坯定义
_APZA	整数	参数 _APZ 的求值，绝对或者增量，90=绝对，91=增量
_APX	实数	轴向数值，用于端面轴的毛坯定义
_APXA	整数	参数 _APX 的求值，绝对或者增量，90=绝对，91=增量
_TOL1	实数	毛坯公差



## 功能

借助扩展的循环CYCLE950，可通过轴向平行或与轮廓平行的切削加工轮廓。可定义一个在切削时会考虑的任意毛坯件。成品轮廓必须相互相关，并可包含任意多的底切单元。可将毛坯件规定为轮廓或通过轴方式的值。

通过该循环，可以以纵向加工或平面加工方式加工轮廓。可自由选择工艺（粗加工、精加工、综合加工、加工方向和进刀方向）。可进行毛坯件更新。

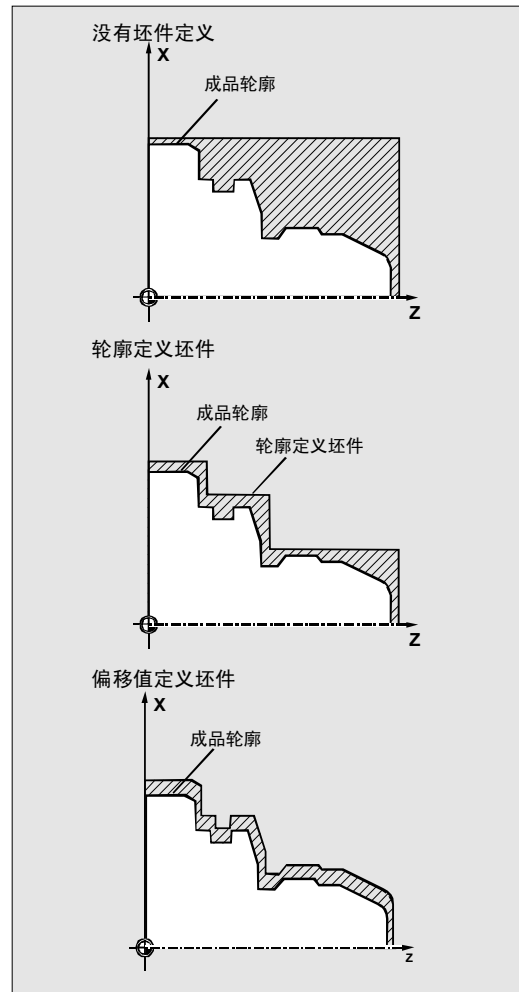
在粗加工时精确执行编程的进刀深度，最后的两个粗切削均匀分配。粗加工至编程的精加工余量。

精加工方向与粗加工方向相同。

刀具半径补偿由循环自动选择并撤消选择。

**新功能（与CYCLE95相对）：**

- 一个毛坯
  - 可以通过编程一个轮廓定义，通过说明成品轮廓的加工余量定义，也可以选择通过说明被切削的毛坯圆柱（或者空心圆柱，在内加工时）定义。
- 可识别出，无法用有效的刀具加工剩余的材料。循环可由此产生一个更新的毛坯件轮廓，该轮廓在零件程序存储器中作为程序存储。
- 毛坯切削时的轮廓可以选择
  - 一个单独的程序，
  - 在所调用的主程序中，或者
  - 规定为一个任意程序的部分。
- 在粗加工时可以在轴向平行加工和轮廓平行加工之间进行选择。
- 在粗加工时沿着轮廓拉削，这样不会剩余余角，或者选择在粗加工切削点立即退刀。
- 粗加工时可以从轮廓退刀的角度可以编程。
- 在粗加工时可以选择加工底切，或者省去底切。

**工作流程****循环开始之前到达的位置：**

出发位置可以为一个任意位置，从这个出发位置可以返回运行到毛坯轮廓，没有轮廓冲突。该循环计算到起始点的、无轮廓冲突的返回运行，用于加工，但是不考虑刀架数据。

**粗加工时轴向平行的运行过程：**

- 粗加工时循环内部计算出发点，并以G0返回运行。
- 以 G0 进刀至下一个根据 参数\_MID 中的给定值计算的深度，随后以 G1 进行轴向平行粗加工。粗加工时，根据轨迹在循环内

部由纵向进给和平面进给值将进给计算为所产生的进给率（\_FF1 和 \_FF2）。

- 在加工方式“轮廓上拉削”中与轮廓平行运行到前一个切削点。
- 在到达前一个切削点，或者在加工方式“轮廓不带拉削”时，以\_ANGLEB中编程的角度退刀，并且以G0退回到下一个进刀的出发点；在45度角时同样准确执行编程的退刀位移\_VRT，在其它的角度时不超出。
- 重复该过程，直至到达加工截面的总深度。

#### 粗加工时轮廓平行的运行过程:

- 粗加工时的出发点和各个进刀深度如同在轴向平行时粗加工一样进行计算，并且以G0或者G1返回运行。
- 粗加工在轮廓平行的轨迹上进行。
- 退刀和退回如同轴向平行时粗加工一样进行。



## 参数说明

### **\_NP1、\_NP2、\_NP3（成品的轮廓编程）**

可以在一个独立的程序中编程成品轮廓，也可以在一个调用的主程序中编程成品轮廓。通过参数 **\_NP1** - 程序名称或者 **\_NP2、\_NP3** - 程序段标记从 ... 至通过程序段号码或者标签传送到循环上。

由此，共有三种轮廓编程方法：

- 轮廓位于一个独立的程序中，必须仅编程 **\_NP1**；  
（参见编程举例1）
- 轮廓位于所调用的程序中，必须仅编程 **\_NP2**和**\_NP3**；  
（参见编程举例2）
- 毛坯切削轮廓是一个程序的一部分，但是不是调用循环的程序的一部分，必须编程所有三个参数。

在编程轮廓时在最后的轮廓单元中（带标签的程序段或者毛坯轮廓结束处的程序段号）不允许包含半径或者棱角。

**\_NP1**中的程序名可以通过路径名和程序类型编写。

举例：

```
_NP1="/_N_SPF_DIR/_N_TEIL1_SPF"
```

### **\_NP4（切削程序名称）**

切削循环生成一个用于坯件和成品之间切削所需的运行段程序。如果没有指定路径，该程序存储在调用的程序所在目录下的零件程序存储器中。否则根据路径数据存储在文件系统中。如果没有规定其它类型，该程序是一个主程序（类型MPF）。

参数 **\_NP4**定义该程序的名称。



**\_VARI (加工方式)**

使用参数\_VARI可以确定加工方式。

可能的值是：

个位：

- 1=纵向
- 2=平面
- 3=轮廓平行

十位：

- 1=编程的进刀方向 X-
- 2=编程的进刀方向 X+
- 3=编程的进刀方向 Z-
- 4=编程的进刀方向 Z+

百位：

- 1=粗加工
- 2=精加工
- 3=综合加工

千位：

- 1=带拉削
- 2=不带拉削 (退刀)

选择轮廓带或者不带拉削，可以确定是否立即退刀到切削点，或者是否沿着轮廓拉削至先前的切削点，从而不留余角。

万位：

- 1=加工底切
- 2=不加工底切

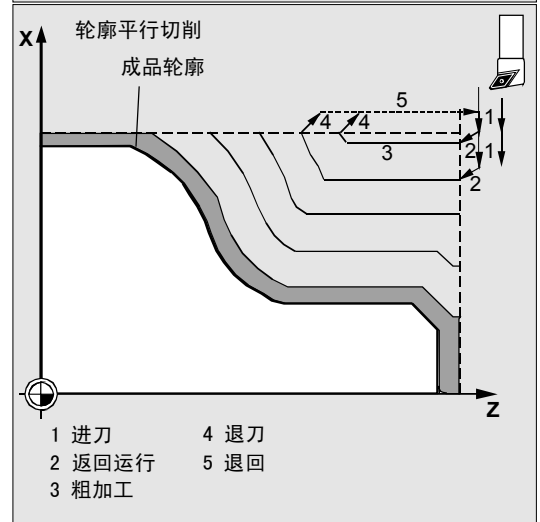
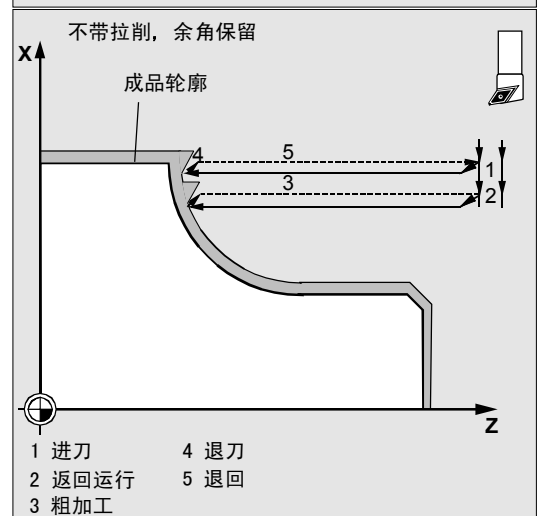
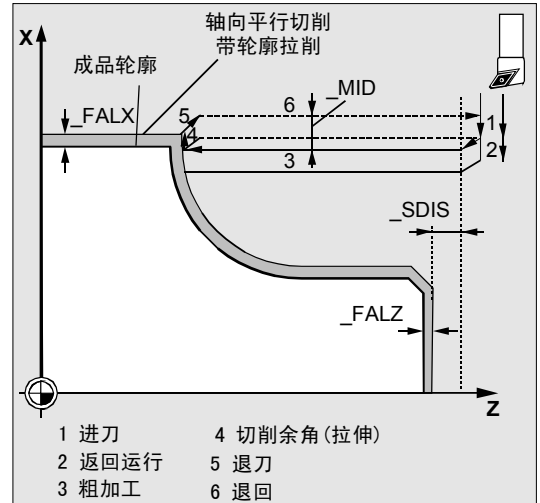
十万位：

- 1=编程的加工方向 X-
- 2=编程的加工方向 X+
- 3=编程的加工方向 Z-
- 4=编程的加工方向 Z+

举例：

\_VARI=312311 表示加工：

- 纵向，
- 进刀方向X- (也是外部)，
- 综合；
- 不沿着轮廓拉削，加工底切，
- 加工方向Z-。



**\_MID (粗加工时进刀深度)**

粗加工时，进刀深度在参数 **\_MID**

下编程。通过该进刀量建立粗加工走刀，直至剩余深度小于  $2 * \text{进刀深度}$ 。然后再分别用二分之一的剩余深度分二次进行走刀。

如果粗加工时平面轴参加进刀，则 **\_MID** 作为半径或者直径进行计算，取决于循环设定数据 **\_ZSD[0]**。

**\_ZSD[0]=0:** **\_MID** 根据 G 组 (半径编程/直径编程)

求值，

在 **DIAMOF** 时作为半径，否则作为直径。

**\_ZSD[0]=1:** **\_MID** 是一个半径值

**\_ZSD[0]=2:** **\_MID** 是一个直径值

在与轮廓平行的粗加工时，进刀深度与给定的进给轴无关，而是垂直于轮廓。由此，总是得到比用相同进刀深度进行轴向平行粗加工时多的走刀次数。

**\_FALZ、\_FALX (精加工余量)**

粗加工时所规定的精加工余量通过参数 **FALZ** (用于 Z 轴) 和 **FALX**

(用于 X 轴) 规定，粗加工一直到这个精加工余量。

如果没有编程精加工余量，则在粗加工时一直切削到最终轮廓。

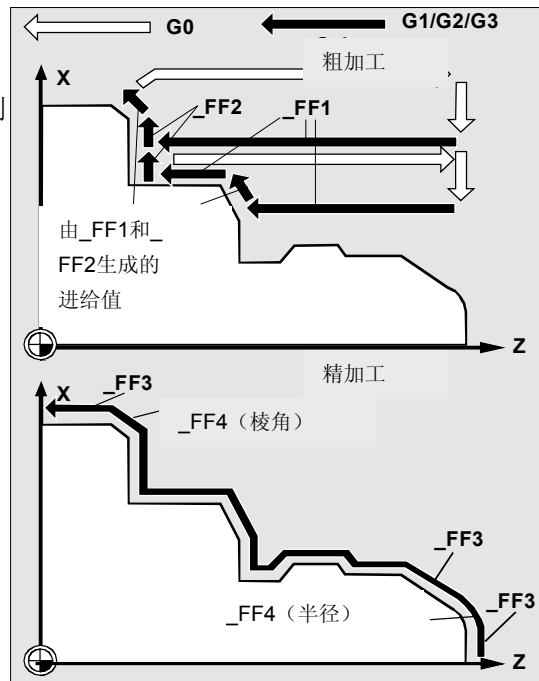
**\_FF1、\_FF2、\_FF3 和 FF4 (进给)**

粗加工和精加工可以分开规定进给率，如图中所示。

粗加工时，纵向进给 (**\_FF1**) 和平面进给 (**\_FF2**) 分别生效。如果在轮廓上拉削时运行斜面或者圆弧轨迹段，则在循环内部自动计算相应产生的进给值。

精加工时，轮廓上编程的进给值生效。如果没有编程，则 **\_FF3** 中的值作为精加工进给值生效；在半径和棱角上，用于该轮廓过渡元素的 **\_FF4** 中编程的进给值生效。

(图中该零件的编程参见编程举例1)。



### **\_VRT (退刀位移) 和 \_ANGB (退刀角)**

在参数\_VRT下可以编程一个退刀量，在粗加工时在两个轴上退刀。

当\_VRT=0时（参数没有编程），退刀1毫米。

另外，可以对参数 \_ANGB

下的角度进行编程，在该角度下从轮廓处退刀。如果没有编程，则在 45 度角下退刀。

### **\_SDIS (安全距离)**

参数\_SDIS确定以哪种距离绕开障碍物。该距离用于例如从底切离开并返回运行到下一个底切。

如果没有编程，则这段距离为1毫米。

### **\_NP5, \_NP6, \_NP7 (毛坯的轮廓编程)**

如果毛坯作为轮廓编程，则在参数\_NP5下编程序序名，或者在参数\_NP6和\_NP7下编程序序部分。

否则编程如同成品编程时一样

（参见\_NP1、\_NP2、\_NP3）。

### **\_NP8 (更新以后毛坯轮廓的轮廓程序名称)**

循环 CYCLE950

可以识别剩余材料，使用当前的刀具不可以切削该剩余材料。

为了可以继续用另一把刀具进行加工，可由此自动生成一个更新的毛坯轮廓。该毛坯轮廓作为程序存储在零件程序存储器中。也可以在参数 \_NP8

下有选择地通过路径数据对程序名称进行规定（参见编程举例 3）。

如果也生成一个运行程序，则会生成一个更新过的毛坯轮廓。

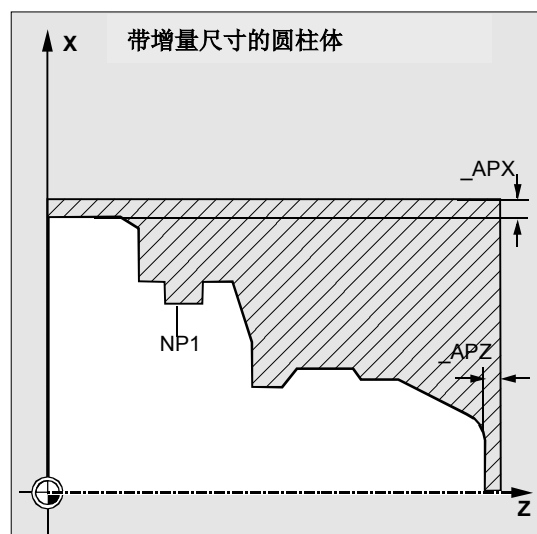
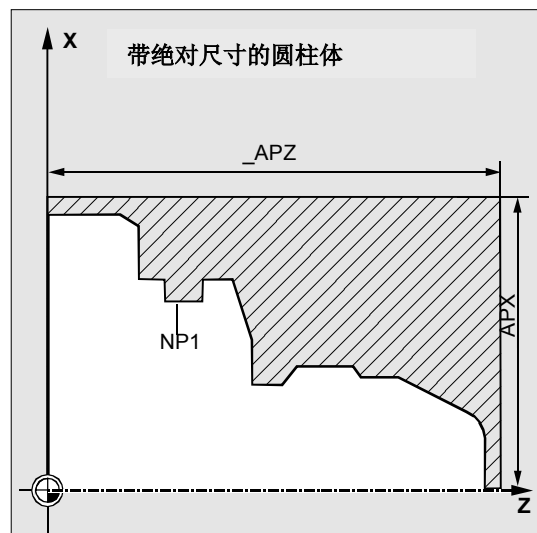
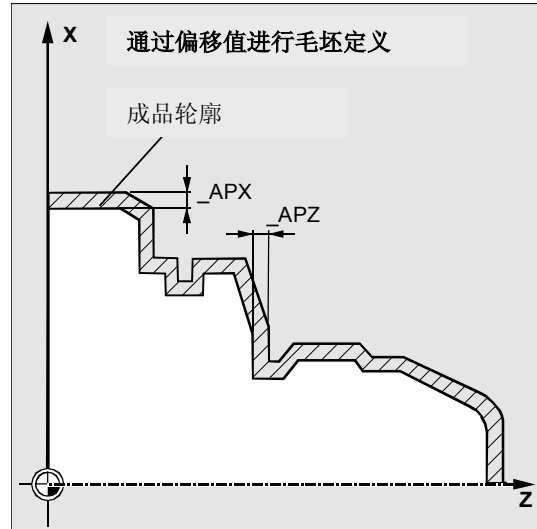
**\_APZ、\_APZA、\_APX、\_APXA (毛坯定义)**

一个毛坯件也可以通过给毛坯圆柱体

(或者空心圆柱体) 说明尺寸进行定义, 或者作为成品轮廓的加工余量在参数 **\_APZ** 和 **\_APX** 中定义。

圆柱体的尺寸可以作为绝对值输入, 也可以选择增量值输入, 成品轮廓的余量则始终作为增量值处理。

通过参数 **\_APZA** 和 **\_APXA** 可以区分绝对值或者增量值 (**\_APZA**, **\_APXA**: 90 – 绝对值, 91 – 增量值)。



### \_TOL1 (毛坯公差)

如果坯件是锻造或者浇注的，则可能会出现坯件与定义不是精确相符，这样最好在返回运行用于粗加工和进刀过程中不是以G0运行到坯件轮廓，而是事先就用G1进行运行，从而补偿可能出现的公差。参数 \_TOL1确定 G1在距坯件多大距离时生效。

从该毛坯部件前的增量值起用 G1

进行运行。如果参数未编程，将赋值为 1毫米。

### 其它说明

#### 轮廓定义

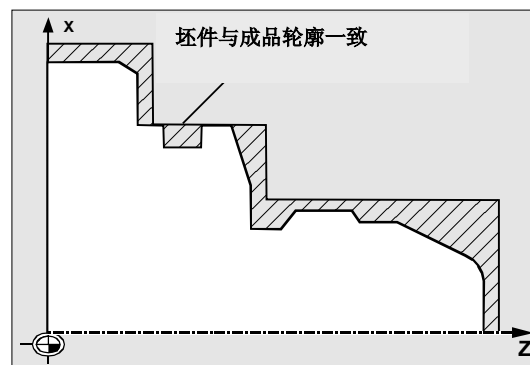
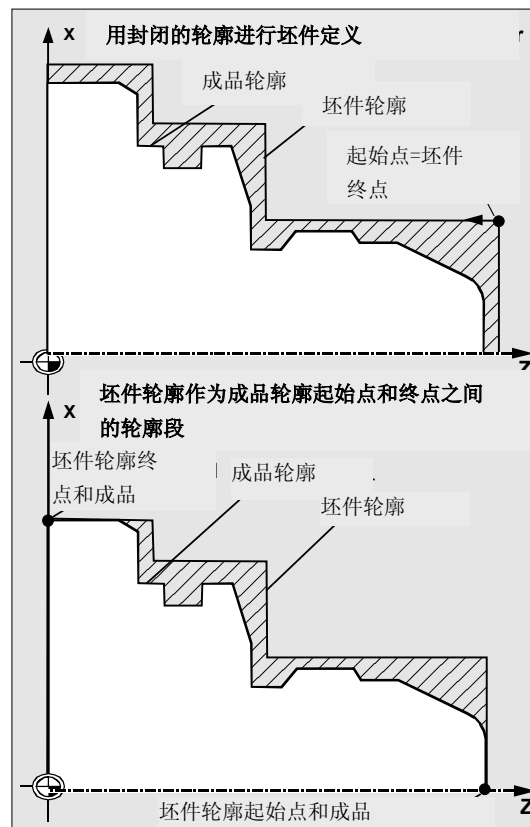
在与CYCLE95进行区分时，只需要在轮廓编程中编写一个包含当前平面中的一个位移的程序段即可以。

轮廓定义的有关说明参见CYCLE95。

#### 坯件轮廓定义

一个坯件轮廓必须是一个封闭的轮廓（起始点=终点），它全部或者部分地包围成品轮廓；或者坯件轮廓是一个成品轮廓在起始点和终点之间的一部分。编程方向在此没有关系。

在描述坯件轮廓时注意它们并不是逐件地与成品轮廓一致，也就是说待加工的材料不是相互关联的。



### 循环结构的说明

循环CYCLE950用于解决切削时所出现的非常复杂的问题，这对系统的计算能力提出很高的要求。为了使处理时间得到优化，这在HMI中进行计算。

循环启动该计算，结果中生成带运行程序段的程序，用于毛坯切削。程序存储在文件系统中，由循环立即调用并执行。

该结构使得，仅在第一次执行程序时必须通过CYCLE950调用进行计算。从第二次调用起，已存在运行程序，并同样可以由循环调用。

在以下情况下重新计算：

- 所参与的轮廓发生改变；
- 该循环的传送参数发生改变；
- 在循环调用之前激活一个带其它补偿参数的刀。

### 文件系统中程序存储

如果在所调用的主程序之外编程CYCLE950轮廓，则按照以下步骤查找控制系统中的文件系统：

- 如果所调用的程序在一个工件目录中，则编程了成品轮廓或者坯件轮廓的程序也必须在同一个工件目录下；
- 如果所调用的程序在目录“零件程序”（MPF.DIR）下，则在没有说明路径时同样在那儿查找程序。

该循环产生一个程序，在该程序中包含毛坯切削的运行程序段，还可以选择一个更新过的毛坯轮廓。

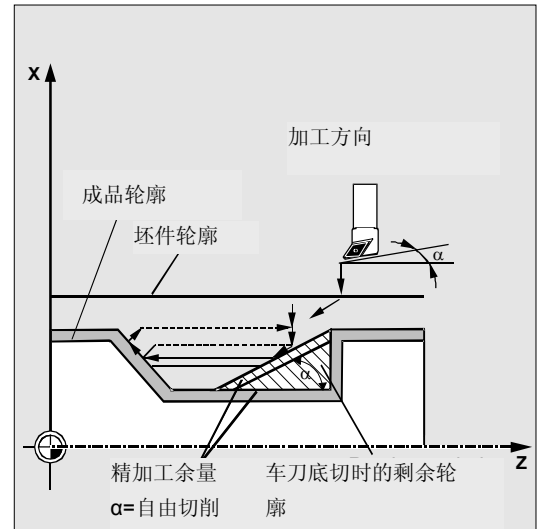
它们可以存放在调用循环的程序所在的目录下，也可以通过相应的路径说明。

### 模拟说明

在模拟扩展的切削循环 CYCLE950时生成的程序存储在NCU的文件系统中。由于刀具补偿数据引入至程序计算中，因此仅带有“NC激活数据”的设置有意义。

### 坯件更新

扩展的毛坯切削循环CYCLE950识别出粗加工时的剩余材料，可以在加工之外产生一个更新的坯件轮廓，它可以用于另一个加工步骤。



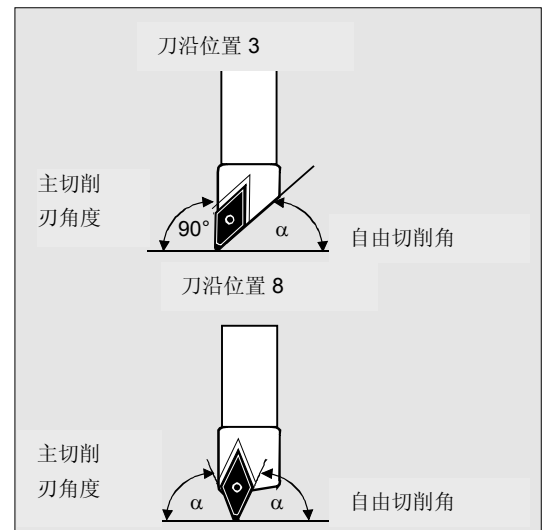
此外，循环在内部考虑刀沿处的角度。

对此，刀具的自由切削角度必须输入到刀具补偿参数（参数24）中。

主切削刃角度由循环根据刀沿位置自动确定。

对于刀沿位置1...4，以一个90度的主切削刃角度计算，用于坯件更新。对于刀沿位置5...9，采用与自由切削角相等的主切削刃角度。

如果在一个CYCLE950的程序中每次以坯件更新进行调用，则必须给所生成的坯件轮廓给定不同的名称；不允许多次使用同一个程序名称（参数\_NP8）。



扩展的坯件切削不可以执行m:n配置。



### 编程举例1

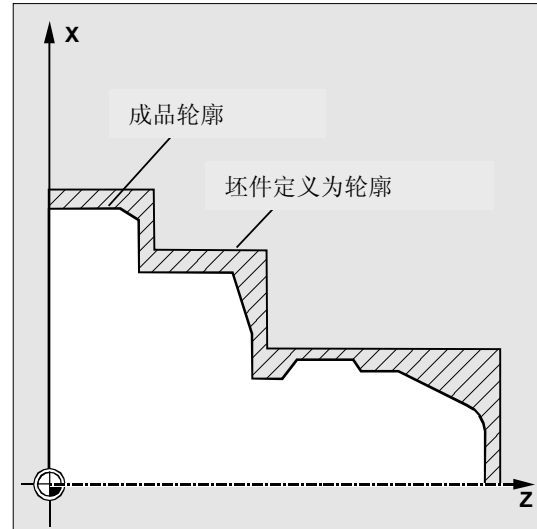
从一个预制的坯件中加工程序TEIL1.MPF中存放的轮廓。

坯件切削的加工方式为

- 仅粗加工，
- 纵向，
- 外部，
- 带拉削（从而没有余角剩余），
- 底切应进行加工。

坯件轮廓在程序ROHTEIL1.MPF中规定。

使用车刀，刀沿3，半径0.8毫米。



### 加工程序:

```

%_N_BEISPIEL_1_MPF
; $PATH=/_N_WKS_DIR/_N_ABSPANEN_NEU_WPD
; 举例 1: 用坯件切削
; Sca, 01.04.99
;
; 刀具补偿数据
N10 $TC_DP1[3,1]=500 $TC_DP2[3,1]=3
$TC_DP6[3,1]=0.8 $TC_DP24[3,1]=60
N15 G18 G0 G90 DIAMON
N20 T3 D1
N25 X300
N30 Z150
N35 G96 S500 M3 F2
N45 CYCLE950("零件1",,, "加工_零件1",
311111,1.25,1,1,0.8,0.7,0.6,0.3,0.5,45,2,
"坯件1",,,,,,,1)
N45 G0 X300
N50 Z150
N60 M2

```

### 成品轮廓:

```

%_N_TEIL1_MPF
; $PATH=/_N_WKS_DIR/_N_ABSPANEN_NEU_WPD
; 成品轮廓举例1
;

```



```

N100 G18 DIAMON F1000
N110 G1 X0 Z90
N120 X20 RND=4
N130 X30 Z80
N140 Z72
N150 X34
N160 Z58
N170 X28 Z55 F300
N180 Z50 F1000
N190 X40
N200 X60 Z46
N210 Z30
N220 X76 CHF=3
N230 Z0
N240 M17

```

#### 坯件轮廓:

```

%_N_ROHTEIL1_MPF
; $PATH=/_N_WKS_DIR/_N_ABSPANEN_NEU_WPD
; 坯件轮廓举例1
;
N100 G18 DIAMON F1000
N110 G0 X0 Z93
N120 G1 X37
N130 Z55
N140 X66
N150 Z35
N160 X80
N170 Z0
N180 X0
N190 Z93 ; 终点=起始点坯件轮廓必须封闭
N200 M17

```

在加工之后，在工件ABSPANEN\_NEU.WPD中有一个新的程序BEARBEITE\_TEIL1.MPF。该程序在第一次调用程序时产生，包含加工轮廓（根据坯件）时的运行。



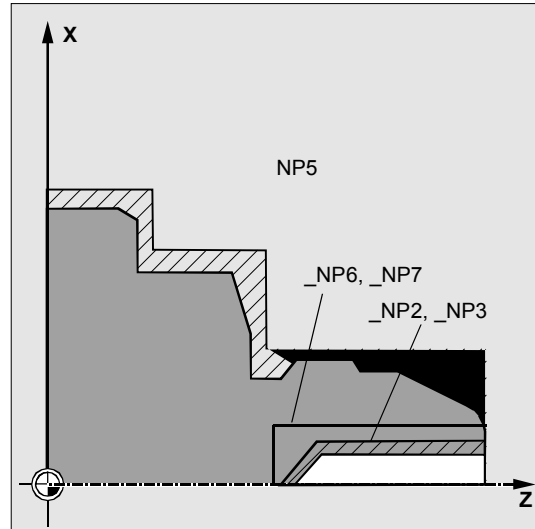
## 编程举例2

现在，对编程举例1中的零件进行简单的内部轮廓的加工。

为此，首先在中心预制孔，使用直径为 10 的钻头。然后与轮廓平行对内部轮廓进行粗加工，因此此钻削相当于到最终轮廓。

同样定义一个坯件轮廓用于内部加工。

毛坯轮廓在同一个程序中，如同循环调用在程序段N400到N420，坯件轮廓在程序段N430...N490。



### 加工程序:

```

%_N_BEISPIEL_2_MPF
; $PATH=/_N_WKS_DIR/_N_ABSPANEN_NEU_WPD
; 举例 2: 内部与轮廓平行切削

; Sca, 01.04.99
;
; 车刀补偿数据, 内部
N100 $TC_DP1[2,1]=500 $TC_DP2[2,1]=6
$TC_DP6[2,1]=0.5 $TC_DP24[2,1]=60
N105 $TC_DP1[1,1]=200 $TC_DP3[1,1]=100
$TC_DP6[1,1]=5
N110 G18 G0 G90 DIAMON

N120 X300
N130 Z150
N140 T1 D1 M6 ; 更换直径为10的钻头
N150 X0 ; 中心钻削分三步
N160 Z100
N170 F500 S400 M3
N175 G1 Z75
N180 Z76
N190 Z60
N200 Z61
N210 Z45
N220 G0 Z100
N230 X300 ; 返回刀具更换点
N240 Z150
N250 T2 D1 M6 ; 更换车刀, 用于内部加工
N260 G96 F0.5 S500 M3

```

```
N275 CYCLE950 ("", "N400", "N420",  
"加工_零件1_内部", 311123, 1.25, 0, 0,  
0.8, 0.5, 0.4, 0.3, 0.5, 45, 1, "", "N430", "N490"  
, , , , , , 1)  
N280 G0 X300  
N290 Z150  
N300 GOTOF _ENDE ; 轮廓定义跳转  
N400 G0 X14 Z90 ; N400到N420成品轮廓  
N410 G1 Z52  
N420 X0 Z45  
N430 G0 X10 Z90 ; N430到N490毛坯轮廓  
N440 X16  
N450 Z40  
N460 X0  
N470 Z47  
N480 X10 Z59  
N490 Z90  
N500 _ENDE:M2
```



### 编程举例3

与程序举例1中同一个零件，现在分为两步加工。

在第一步加工（N45）中，用一把带有刀沿位置 9 和较大半径的刀具以较大的进刀深度在没有规定坯件的情况下进行粗加工。最终要生成一个带有名称 ROHTEIL3.MPF 的更新的坯件。

该步骤的加工方式为：

- 仅粗加工，
- 纵向，
- 外部，
- 带拉削，
- 不加工底切。

在第二步加工中（N75），从该坯件出发，用另一把刀具加工其剩余材料，并接着进行精加工。

该步骤的加工方式为：

- 综合加工（粗加工和精加工）
- 纵向，
- 外部，
- 带拉削（从而没有余角  
剩余），
- 底切应进行加工。

#### 加工程序：

```

%_N_BEISPIEL_3_MPF
; $PATH=/_N_WKS_DIR/_N_ABSPANEN_NEU_WPD
; 举例 3：通过坯件更新，以两步进行切削
; Sca, 09.04.99
;
; 刀具补偿数据
; T3: 粗加工刀具用于粗加工，刀沿 9，半径 5
N05 $TC_DP1[3,1]=500 $TC_DP2[3,1]=9
$TC_DP6[3,1]=5 $TC_DP24[3,1]=80
; T4: 车刀用于剩余材料和精加工
; 刀沿 3，半径 0.4

```

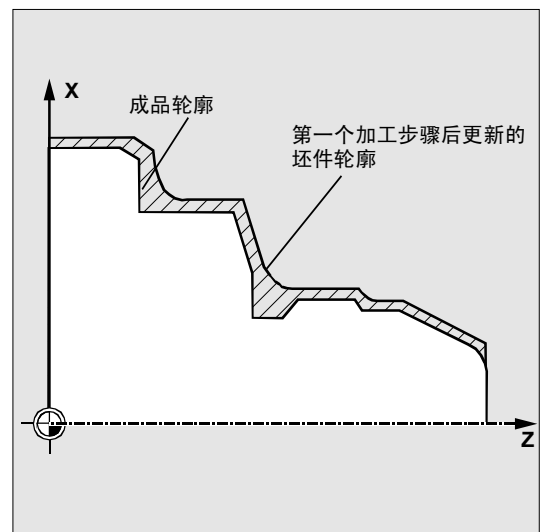
```

N10 $TC_DP1[4.1]=500 $TC_DP2[4.1]=3
$TC_DP6[4.1]=.4 $TC_DP24[4.1]=80
N15 G18 G0 G90 DIAMON
N20 T3 D1 ; 粗加工刀具
N25 X300
N30 Z150
N35 G96 S500 M3 F2
N45 CYCLE950 ("零件1",,, "加工_零件3",
321111,8,1,1,0.8,0.7,0.6,0.5,1,45,6,
"DEFAULT",,, "坯件3",0,91,0,91,1)
N50 G0 X300
N55 Z150
N60 T4 D1 ; 用于粗加工剩余材料和精加工的刀具
N65 G96 S500 M3 F2
N75
CYCLE950 ("零件1",,, "精加工_零件3",311311,0.
5,0.25,0.25,0.8,0.7,0.6,0.5,1,45,6,"坯件3
",,,,,,,1)
N160 M2

```

**成品轮廓:**

如同编程举例1





## 说明

## 报警原因CYCLE950

报警号	报警文本	说明, 纠正
61701	“成品轮廓描述错误”	没有提供参数_ <b>NP1</b> , <b>NP2</b> 和 <b>NP3</b> , 或者在成品轮廓编程中出错
61702	“坯件轮廓描述错误”	没有提供参数_ <b>NP5</b> , <b>NP6</b> 和 <b>NP7</b> , 或者在坯件轮廓编程中出错
61703	“在文件中删除内部循环故障”	
61704	“在文件中写出内部循环故障”	
61705	“在文件中读出内部循环故障”	
61706	“在检查和构成中内部循环故障”	
61707	“在 <b>ACTIVATE</b> 到 <b>MMC</b> 时内部循环故障”	
61708	“在 <b>READYPROG</b> 到 <b>MMC</b> 时内部循环故障”	
61709	“在轮廓计算时超出时间”	
61720	“非法输入”	
61721	“错误轮廓方向不可计算”	
61722	“系统故障”	
61723	“不可能加工”	使用有较大后角的刀具
61724	“没有材料”	
61725	“存储器位置有问题, 因此在产生轮廓时出错”	
61726	“内部错误: 存储器空间问题 _FILECTRL_EXTERNAL_ERROR”	
61727	“内部错误: 存储器空间问题 _FILECTRL_INTERNAL_ERROR”	
61728	“内部错误: 存储器空间问题 _ALLOC_P_INTERNAL_ERROR”	
61729	“内部错误: 存储器空间问题 _ALLOC_P_EXTERNAL_ERROR”	
61730	“内部错误: 无效的存储器”	
61731	“内部错误: 浮点溢出”	
61732	“内部错误: 无效的结构”	
61733	“内部错误: 浮点出错”	
61734	“刀沿与切削方向不相容”	

报警号	报警文本	说明, 纠正
61735	“成品在坯件轮廓之外”	检查坯件轮廓定义
61736	“刀具插入长度<加工深度”	
61737	“加工_切削深度>最大_刀具_切削深度”	
61738	“加工_切削深度<最小_刀具_切削深度”	
61739	“刀具用于该加工的插入_位置错误”	
61740	“坯件必须是封闭的轮廓”	坯件轮廓必须封闭, 起始点=终点
61741	“由于存储器故障停止”	
61742	“返回运行时轮廓冲突, 不可能修正”	

■

用于记录



## 故障信息和故障消除

5.1	一般提示 .....	5-354
5.2	循环中的故障处理 .....	5-354
5.3	循环报警一览.....	5-355
5.4	循环中的信息 .....	5-361

## 5.1 一般提示

如果在循环中识别出错误的状态，则产生一个报警，并且中断该循环的执行。  
另外循环还会在控制器的对话框中输出信息。这个信息不会中断加工。



这些故障及其应答以及系统中对话框中的信息每次均在各个循环中进行描述。

## 5.2 循环中的故障处理

如果在循环中识别出错误的状态，则产生一个报警，并且中断该加工。

在循环中会产生编号为61000到62999之间的报警。根据报警反应和清除标准，对该号码区再次进行划分。

报警文本，同时显示报警号，对报警原因进行详细的说明。

报警序号	删除条件	报警应答
61000 ... 61999	NC复位	在NC中程序段处理中断
62000 ... 62999	删除键	程序处理不中断；仅显示

### 5.3 循环报警一览

除了少数例外情况外，  
报警序号可以如下分类：

6	-		-	-
---	---	--	---	---

- X=0 共同的循环报警
- X=1 钻削循环、钻削图循环和铣削循环报警
- X=6 车削循环报警

在下表中您可以找到循环中所出现的报警、它们的出现地点以及报警消除的说明。

报警序号	报警文本	地点	说明，纠正
61000	“没有刀具补偿有效”	长孔 槽1 槽2 腔体1 到 腔体4 循环1 循环72 循环90 循环93 到 循环96	D补偿必须在循环调用之前编程
61001	“螺距定义错误”	循环84 循环840 循环96 循环97	检查用于螺纹尺寸或者螺距说明的参数 (相互矛盾)
61002	“加工方式错误定义”	槽1 槽2 腔体1 到 腔体4 循环71 循环72 循环76 循环77 循环93 循环95 循环97 循环98	用于加工方式的参数VARI的值错误 规定，必须修改

报警序号	报警文本	地点	说明, 纠正
61003	”在循环中没有编程进给	“循环71 循环72	进给参数错误规定, 必须进行修改
61005	“3. 几何轴没有出现”	循环86	在不带Y轴的车床G18平面中使用
61009	“有效的刀具号=0”	循环71 循环72	在循环调用之前没有编程刀具 (T)。
61010	“精加工余量太大”	循环72	底部的精加工余量大于总深度, 必须缩小。
61011	“比例缩放不允许”	循环71 循环72	有该循环不允许的比例系数。
61012	“平面中比例缩放不同”	循环76 循环77	
61014	“超出回退平面”	循环72	检查参数 RTP。
61019	“参数错误编程”	循环83 循环60	检查参数值。
61101	“基准面错误定义”	循环71 循环72 循环81 到 循环90 循环840 槽1 槽2 腔体1 到 腔体4 长孔	在对深度进行相对说明时, 选择不同值的基准面和回退平面; 或者深度必须用绝对值规定。
61102	“没有编程主轴方向”	循环86 循环87 循环88 循环840 腔体3 腔体4	参数SDIR (或者SDR在循环840中) 必须编程。
61103	“孔数为零”	孔1 孔2	钻孔数没有编程数值。
61104	“键槽轮廓损伤”	槽1 槽2 长孔	在参数中错误地设定铣削图, 这些参数确定圆弧上键槽/长方形孔的位置及其形状。
61105	“铣刀半径太大”	槽1 槽2 腔体1 到 腔体4 长孔 循环90	所使用铣刀的直径对于待加工的轮廓来说太大; 或者是使用一个较小半径的刀具, 或者是修改轮廓。

报警序号	报警文本	地点	说明, 纠正
61106	“圆弧单元个数或者间距”	孔2 长孔 槽1 槽2	错误地定义NUM或者INDA的参数, 在一个整圆中不可能排列这些圆弧单元
61107	“第一个钻削深度错误定义”	循环83	第一个钻削深度与总钻深相矛盾
61108	“不允许的值用于参数_RAD1 和_DP1”	腔体3 腔体4	用于确定深度进刀轨迹的参数_RAD1 和_DP错误规定。
61109	“参数_CDIRE错误定义”	腔体3 腔体4	用于铣削方向的参数_CDIRE的值错误规定, 必须修改。
61110	“底部的精加工余量>深度进刀”	腔体3 腔体4	底部的精加工余量大于最大深度进刀; 或者是缩小精加工余量, 或者是放大深度进刀
61111	“进刀宽度>刀具直径”	循环71 腔体3 腔体4	编程的进刀宽度大于当前刀具的直径, 它必须缩小。
61112	“刀具半径为负”	循环72 循环76 循环77 循环90	当前刀具的半径为负, 这是不允许的。
61113	“拐角半径参数_CRAD太大”	腔体3	拐角半径_CRAD的参数规定太大, 它必须缩小。
61114	“加工方向G41/G42错误定义”	循环72	铣刀半径补偿G41/G42的加工方向错误选择。
61115	“返回运行或者开始运行模式(直线/圆弧/平面/空间)错误定义”	循环72	轮廓上返回运行或者开始运行模式错误定义; 检查参数_AS1或者_AS2。
61116	“返回运行位移或者开始运行位移=0”	循环72	返回运行位移或者开始运行位移规定为零, 必须加以放大; 检查参数_LP1或者_LP2。
61117	“有效的刀具半径<=0”	循环71 腔体3 腔体4	当前刀具的半径为负或者零, 这是不允许的。
61118	“长度或者宽度=0”	循环71	铣削平面的长度或者宽度不允许; 检查参数_长度和_宽度。

报警序号	报警文本	地点	说明, 纠正
61124	“进刀宽度没有编程”	循环71	在不带刀具进行有效模拟时, 必须始终编程一个值用于进刀宽度_MIDA。
61125	“参数_TECHNO中工艺选择错误定义”	循环84 循环840	检查参数_TECHNO。
61126	“螺纹长度太短”	循环840	编程较小的主轴转速/基准面设定较高
61127	“螺纹钻削轴的传动比错误定义(机床数据)”	循环84 循环840	检查钻削轴相应的齿轮级中机床数据31050和31060
61128	“在以摆动或者螺旋插入时再入角=0”	槽1	检查参数_STA2
61180	“尽管机床数据\$MN_MM_NUM_TOOL_CARRIER>1, 但是没有给回转文件程序段分配名称”	循环800	虽然有几个回转文件程序段(\$MN_MM_NUM_TOOL_CARRIER>0), 或者没有约定回转文件程序段(\$MN_MM_NUM_TOOL_CARRIER=0), 但是回转文件程序段没有分配名称。
61181	“NCK软件版本太低(缺少功能TOOLCARRIER)”	循环 800	功能 TOOLCARRIER 自 NCU 6.3xx 起
61182	“回转文件程序段名称未知”	循环800	参见回转循环循环800开机调试 →运动学名称(回转文件程序段)
61183	"空运行方式 GUD7_TC_FR 在值范围0..2之外"	循环800	参见回转循环800开机调试 → 空运行; 1. 传送参数循环800(x,...)有问题 >2
61184	“使用当前的输入角度值不可能解决”	循环800	
61185	“没有约定或者约定了错误(最小>最大)的回转轴角度范围”	循环 800	检查回转循环800开机调试
61186	“回转轴矢量无效→检查回转循环800开机调试”	循环 800	回转循环800开机调试: 没有或错误的回转矢量 V1 或 V2 登记
61187	“程序段查找不允许→选择带轮廓计算的程序段查找”	循环800	选择带轮廓计算的程序段查找

报警序号	报警文本	地点	说明, 纠正
61188	“无轴名称 1. 约定作为回转轴 →检查回转循环800开机 调试”	循环800	回转循环800开机调试: 在回转轴 1 名称下没有输入
61191	“5轴转换没有设置”	循环832	
61192	“第二个5轴转换没 有设置”	循环832	
61193	“选件压缩器没有设置”	循环832	
61194	“选件样条插补没有 设置”	循环832	
61200	“加工程序块太多单元”	循环76 循环77	修正加工块, 有时删除单元
61213	“圆弧半径太小”	循环77	输入较大的值用于圆弧半径
61215	“毛坯尺寸错误编程”	循环76 循环77	检查毛坯轴颈尺寸。毛坯轴颈尺寸必须大于 成品轴颈尺寸。
61601	“成品直径太小”	循环94 循环96	编程一个成品直径。
61602	“刀具宽度错误定义 ”	循环93	切槽刀具大于编程的切槽宽度。
61603	“切槽形状错误 定义”	循环93	<ul style="list-style-type: none"> <li>切槽底部半径/棱角不适合切槽宽度</li> <li>在平行于纵向轴走向的轮廓单元上平 面切槽不可能</li> </ul>
61604	“当前的刀具破坏编 程的轮廓”	循环95	所使用刀具的自由切削角度决定了底切单元 中轮廓损坏, 也就是说使用另一个刀具或者 检查轮廓子程序
61605	“轮廓错误编程”	循环76 循环77 循环95	识别不允许的底切单元
61606	“轮廓预处理出错”	循环95	在预处理轮廓时发现一个错误, 该报警始终 与NCK报警10930...10934, 15800或者 15810相关联
61607	“起始点错误编程”	循环95	在调用循环之前到达的起始点不在轮廓子程 序所描述的直角之外。
61608	“编程了错误的刀沿位置”	循环94 循环96	必须编程一个刀沿1...4, 与退刀槽形状相 应。

报警序号	报警文本	地点	说明, 纠正
61609	“形状错误定义”	循环94 循环96长孔 腔体3 槽1	检查退刀槽形状参数, 或者键槽、凹槽形状参数
61610	“没有编程进刀深度”	循环76 循环77 循环96	
61611	“没有找到切削点”	循环95	无法计算轮廓的切削点位置。检查轮廓编程或改变进刀深度
61612	“不可能进行螺纹精整”	循环97 循环98	
61613	“退刀槽位置错误定义”	循环94 循环96	检查参数_VARI的值
61803	“编程的轴不存在”	循环83 循环84 循环840	检查参数_AXN
61807	“编程错误的主轴方向(有效)”	循环840	检查参数SDR和SDAC
61905	“铣刀半径太小”	槽2	所使用的铣刀直径太小, 剩余材料仍留在键槽中; 使用较大半径的刀具
62100	“没有钻削循环有效”	孔1 孔2	在调用钻削图循环之前没有模态调用钻削循环
62101	“没有正确产生铣削方向-G3”	腔体3 腔体4 槽2	检查参数_CDIRE的值
62105	“列数或者行数为零”	循环800	
62180	“调整回转轴 x.x [度]”	循环800	在手动回转轴中待设定的角度
62181	“调整回转轴 x.x [度]”	循环800	在手动回转轴中待设定的角度



## 5.4 循环中的信息

循环在控制器的对话行中输出信息。这个信息不会中断加工。

信息提供循环和加工步骤某些特性的说明，通常在一个加工步骤之后或者一直至循环结束均保持不变。

可能会有下列信息：

信息文本	地点
“深度：相对深度的相应值”	循环81...循环89, 循环840
“长方形孔加工”	长孔
“键槽加工”	槽1
“圆弧槽加工”	槽2
“错误的铣削方向：生成 G3”	槽1, 槽2, 腔体1, 腔体2, 循环90
“修改的退刀槽形状”	循环94, 循环96
“1. 钻深：相对深度的相应值”	循环83
“注意：精加工余量 $\geq$ 刀具直径”	腔体1, 腔体2
“螺纹导程：— 作为纵向螺纹加工”	循环97, 循环98
“螺纹导程：— 作为平面螺纹加工”	循环97, 循环98
“模拟有效，没有编程刀具，运行最终轮廓”	腔体1...腔体4, 槽1, 槽2, 循环93, 循环72
“模拟有效，没有编程刀具”	循环71, 循环90, 循环94, 循环96
“等待主轴方向转换”	循环840

用于记录

## 附录

A	缩略符.....	A-364
B	术语.....	A-373
C	参考文献.....	A-393
D	名称.....	A-405
E	索引.....	I-409
F	指令/名称.....	I-412

## A 缩略符

<b>A</b>	输出端
<b>AS</b>	可编程控制系统
<b>ASCII</b>	American Standard Code for Information Interchange:美国信息互换标准码
<b>ASIC</b>	专用集成电路应用器件:用户开关回路
<b>ASUP</b>	异步子程序
<b>AV</b>	工作准备部分
<b>AWL</b>	指令表
<b>BA</b>	工作方式
<b>BAG</b>	工作方式组
<b>BB</b>	运行准备
<b>BuB, B&amp;B</b>	操作与观测
<b>BCD</b>	Binary Coded Decimals: 二进制编码十进制数
<b>BHG</b>	操作设备
<b>BIN</b>	二进制文件
<b>BIOS</b>	基本输入输出系统
<b>BKS</b>	基本坐标系统
<b>BOF</b>	操作界面
<b>BOT</b>	引导文件: SIMODRIVE 611D 的引导文件
<b>BT</b>	操作面板
<b>BTSS</b>	操作面板接口
<b>CAD</b>	计算机辅助设计

<b>CAM</b>	计算机辅助制造
<b>CNC</b>	Computerized Numerical Control: 计算机数字控制
<b>COM</b>	通讯
<b>CP</b>	通讯处理器
<b>CPU</b>	Central Processing Unit: 中央处理器
<b>CR</b>	回车
<b>CRT</b>	Cathode Ray Tube: 阴极射线管
<b>CSB</b>	中央维护主板: PLC 模块
<b>CTS</b>	Clear To Send: 串行接口发送就绪状态
<b>CUTOM</b>	Cutradiuscompensation: 刀具半径补偿
<b>DAU</b>	数字模拟转换器
<b>DB</b>	PLC中数据块
<b>DBB</b>	PLC中数据块字节
<b>DBW</b>	PLC中数据块字
<b>DBX</b>	PLC中数据块位
<b>DC</b>	直接控制:在一转内回转轴以最短距离移动到绝对位置
<b>DCD</b>	载波检测
<b>DDE</b>	动态数据交换
<b>DEE</b>	数据结束调试
<b>DIN</b>	德国工业标准
<b>DIO</b>	数据输入/输出: 数据传输显示
<b>DIR</b>	Directory: 目录

<b>DLL</b>	动态连接程序库
<b>DOE</b>	数据传输设备
<b>DOS</b>	磁盘操作系统
<b>DPM</b>	双端口存储器
<b>DPR</b>	双端口存储器
<b>DRAM</b>	动态随机存取存储器
<b>DRF</b>	DRF功能：直接测量功能（手轮）
<b>DRY</b>	Dry Run：空运行进给率
<b>DSB</b>	Decoding Single Block：解码的程序段
<b>DW</b>	数据字
<b>E</b>	输入端
<b>E/A</b>	输入/输出
<b>E/R</b>	SIMODRIVE 611（D）电源/回馈模块
<b>EIA 代码</b>	专门穿孔带编码，每个字符的穿孔数始终为奇数
<b>ENC</b>	编码器：实际值编码器
<b>EPROM</b>	可擦除可编程只读存储器
<b>ERROR</b>	打印机错误
<b>FB</b>	功能块
<b>FBS</b>	超薄显示屏
<b>FC</b>	功能调用：PLC中的功能块
<b>FDB</b>	生产数据库
<b>FDD</b>	软盘驱动器

<b>FEPROM</b>	闪存EPROM: 可读可写存储器
<b>FIFO</b>	先进先出: 存储器, 工作无需地址说明, 数据按存储的顺序读入
<b>FIPO</b>	精插补器
<b>FM</b>	功能模块
<b>FPU</b>	Floating Point Unit: 浮点单位
<b>FRA</b>	框架模块
<b>FRAME</b>	数据段
<b>FRK</b>	铣削半径补偿
<b>FST</b>	Feed Stop: 进给停止
<b>FUP</b>	功能图 (PLC编程方法)
<b>GP</b>	主程序
<b>GUD</b>	Global User Data: 全局用户数据
<b>HD</b>	Hard Disk: 硬盘
<b>HEX</b>	十六进制数代号
<b>HiFu</b>	辅助功能
<b>HMS</b>	高分辨率测量系统
<b>HSA</b>	主轴驱动
<b>HW</b>	硬件
<b>IBN</b>	开机调试
<b>IF</b>	驱动模块脉冲使能
<b>IK (GD)</b>	隐含通讯 (全局数据)
<b>IKA</b>	Interpolative Compensation: 可插补补偿

<b>IM</b>	Interface-Modul: 接口模块
<b>IMR</b>	Interface-Modul Receive:接收方接口模块
<b>IMS</b>	Interface-Modul Send:发送方接口模块
<b>INC</b>	Increment: 步进尺寸
<b>INI</b>	Initializing Data: 初始化数据
<b>IPO</b>	插补器
<b>ISA</b>	国际标准体系
<b>ISO</b>	国际标准组织
<b>ISO 代码</b>	专门穿孔带编码, 每个字符的穿孔数始终为偶数
<b>JOG</b>	Jogging: 手动工作方式
<b>K1..K4</b>	通道1到通道4
<b>K-Bus</b>	通讯总线
<b>KD</b>	坐标旋转
<b>KOP</b>	触点图 (PLC编程方法)
<b>K<sub>v</sub></b>	回路放大系数
<b>K<sub>ü</sub></b>	传动比
<b>LCD</b>	液晶显示器液晶显示
<b>LED</b>	Light Emitting Diode: 发光二极管
<b>LF</b>	线路馈电
<b>LMS</b>	位置测量系统
<b>LR</b>	位置调节器
<b>LUD</b>	局部用户数据
<b>MB</b>	兆字节



<b>MD</b>	机床数据
<b>MDA</b>	Maual Data Automatic: 手动数据输入
<b>MK</b>	测量回路
<b>MCS</b>	机床坐标系
<b>MLFB</b>	机器可识别产品符
<b>MMC</b>	Man Machine Communication: 用于操作, 编程和模拟的数字控制操作界面
<b>MPF</b>	Main Program File: NC零件程序 (主程序)
<b>MPI</b>	0多端口接口
<b>MS-</b>	微软 (软件制造商)
<b>MSTT</b>	机床控制面板
<b>NC</b>	数字控制:数字控制装置
<b>NCK</b>	Numerical Control Kernel: 带有程序段处理, 运行范围等等的数字内核
<b>NCU</b>	Numerical Control Unit: NCK硬件单元
<b>NRK</b>	NCK操作系统名称
<b>NST</b>	接口信号
<b>NURBS</b>	非一致性数理B样条
<b>NV</b>	零点偏移
<b>OB</b>	PLC中组织块
<b>OEM</b>	原设备制造商
<b>OP</b>	Operation Panel: 操作面板
<b>OPI</b>	Operation Panel Interface: 操作面板接口
<b>OPT</b>	Options:选件

<b>OSI</b>	开放式互联系统：计算机通讯标准
<b>P-Bus</b>	外设总线
<b>PC</b>	个人计算机
<b>PCIN</b>	与控制系统进行数据更换的软件名称
<b>PCMCIA</b>	个人计算机内存卡国际协会：存储器插卡标准
<b>PG</b>	编程装置
<b>PLC</b>	可编程逻辑控制器：匹配控制
<b>POS</b>	定位
<b>RAM</b>	随机存取存储器：可读可写程序存储器
<b>REF</b>	回参考点运行
<b>REPOS</b>	再定位功能
<b>RISC</b>	Reduced Instruction Set Computer：精简指令集计算机：带有小命令集和快速命令处理的处理器类型
<b>ROV</b>	Rapid Override：快速倍率
<b>RPA</b>	R 参数已激活：存储器范围 NCK中用于R参数号
<b>RPY</b>	旋转定位移动：一种坐标系旋转方式
<b>RTS</b>	Request To Send：开启发送方，控制信号自串行数据接口
<b>SBL</b>	Single Block：单程序段
<b>SD</b>	设定数据
<b>SDB</b>	系统数据块
<b>SEA</b>	Setting Data Active：设定数据符号（文件类型）
<b>SFB</b>	系统功能块

<b>SFC</b>	系统功能调用
<b>SK</b>	软键
<b>SKP</b>	Skip: 程序段跳过
<b>SM</b>	步进电机
<b>SPF</b>	Sub Program File: 子程序
<b>SPS</b>	存储器可编程控制
<b>SRAM</b>	静态存储器 (缓存)
<b>SRK</b>	刀尖补偿
<b>SSFK</b>	丝杠螺距误差补偿
<b>SSI</b>	Serial Synchron Interface: 串行同步接口
<b>SW</b>	软件
<b>SYF</b>	System Files: 系统文件
<b>TEA</b>	测试数据有效: 机床数据标志
<b>TO</b>	Tool Offset: 刀具补偿
<b>TOA</b>	Tool Offset Active: 刀具补偿符号 (文件类型)
<b>TRANSMIT</b>	Transform Milling into Turning: 在车床上用于铣削的坐标转换
<b>UFR</b>	用户框架: 零点偏移
<b>UP</b>	子程序
<b>VSA</b>	进给驱动
<b>V.24</b>	串行接口 (DEE和DUE之间数据交换定义)
<b>WCS</b>	工件坐标系
<b>WKZ</b>	刀具

<b>WLK</b>	刀具长度补偿
<b>WOP</b>	现场编程
<b>WDP</b>	Work Piece Directory: 工件目录
<b>WRK</b>	刀具半径补偿
<b>WZK</b>	刀具补偿
<b>WZW</b>	换刀
<b>ZOA</b>	Zero Offset Active: 零点偏移数据符号 (文件类型)
<b>μC</b>	微米级控制器

## B 术语

按照字母顺序给出术语说明。说明文字中出现的术语有单独的出处说明，在此用->表示。

## A

## 报警

所有的信息和报警均在操作面板上显示其文本，带日期和时间，并有相应的清除标准符号。报警和信息单独显示。

## 1. 零件程序中报警和信息

报警和信息可以直接从零件程序中以文字形式显示。

## 2. PLC的报警和信息

机床的报警和信息可以从PLC程序中以文字形式显示。在此无需另外的功能块软件包。

## 用户定义变量

用户可以定义用户变量，从而可以在零件程序或者数据块（全局用户数据）中任意使用。一个定义通常含有数据类型和变量名称。

参见->系统变量。

## B

## 模块

模块是指编程和程序执行时所需要的所有文件。

## 命名符

根据DIN 66025标准，字需要补充变量名

（计算变量，系统变量和用户变量）、子程序名、关键字名和带多个地址字母的字。这些补充的字在意义上与构成程序段的字一样。名称必须意义明确。同一个名称不可以用于不同的对象。

## 引导

上电后装载系统程序。

## C

## CNC

→ NC

<b>COM</b>	NC控制系统部件，用于执行和和协调通讯。
<b>CPU</b>	中央处理单元，->存储器可编程控制器
<b>D</b>	
<b>数据块</b>	<ol style="list-style-type: none"><li>1. 数据单元，PLC可以对HIGHSTEP程序进行存取。</li><li>2. 数据块-&gt;NC：数据块包含全局用户数据的数据定义。 数据可以在定义时直接初始化。</li></ol>
<b>数据传送程序PCIN</b>	PCIN是一种辅助程序，通过串行接口发送和接收CNC用户数据，比如零件程序、刀具补偿等等。PCIN程序可以在标准工业计算机中MSDOS下运行。
<b>数据字</b>	在数据块中两个字节大小的数据单位。
<b>诊断</b>	<ol style="list-style-type: none"><li>1. 系统操作区。</li><li>2. 控制系统不仅有自诊断程序，而且还可以进行维修时辅助测试。状态、报警和服务信息。</li></ol>
<b>操作</b>	系统操作区。
<b>数字量输入/输出模块</b>	数字量模块用于二进制过程信号的处理。
<b>DRF</b>	DRF功能：NC功能，在自动方式下利用电子齿轮产生增量式零点偏移。
<b>漂移补偿</b>	在CNC轴恒定量运行期间产生一个自动漂移补偿，用于模拟量转速调节。(SINUMERIK FM-NC).
<b>E</b>	
<b>编辑器</b>	利用编辑器可以进行程序/文本/程序段的编辑、修改、合并和插入。
<b>快速移动</b>	轴运行最快速度。比如，当刀具由静止状态运行到工件轮廓或者由工件轮廓返回时使用快速移动速度。

电子手轮	利用电子手轮可以在手动运行状态运行所选择的轴。手轮上刻度线值的大小由步距值确定。
外部零点偏移	由PLC给定的零点偏移。
<b>F</b>	
成品轮廓	成品工件的轮廓。参见->毛坯件。
返回固定点	机床中可以定义一些固定点，比如刀具更换点、装料点、托盘更换点等等，并可返回。这些点的坐标存储到控制系统中。控制系统控制相关轴运行，如果可能->以快速方式运行。
框架	框架定义一种运算规范，它把一种直角坐标系转换到另一种直角坐标系。框架中包含几个部分->零点偏移->旋转->标尺->镜像。
<b>G</b>	
准停	使用编程的准停指令，可以准确地、有时必须较慢地回到程序段中所设定的位置。为了减少准停时的逼近时间，对于快速移动和进给需定义准停界限。
准停界限	如果所有的轨迹轴均到达准停界限，则控制系统会认为已经精确到达目标。进行零件程序的程序段转换。
几何尺寸	工件在工件坐标系中的描述。
几何轴	几何轴用于描述工件坐标系中2维或者3维的尺寸。

<b>速度控制</b>	在轴移动时，为了可以使每个较小行程的程序段达到一个可以承受的运行速度，可以使用处理多个程序段的预见功能（->Look Ahead）。
<b>线性插补</b>	刀具以直线运行到目标点，同时进行工件的加工。
<b>攻丝，不带补偿衬套</b>	用此功能可以不带补偿衬套攻丝螺纹。通过插补运行，主轴作为回转轴和钻削轴进行螺纹加工，精确地至钻削深度，比如盲孔螺纹（前提条件：主轴作为进给轴运行）。
<b>全局主程序/子程序</b>	在一个目录下每个全局主程序/子程序只可以出现一次，不可以在不同目录下有不同内容的程序具有相同的程序名。
<b>极限速度</b>	最大/最小（主轴）速度：通过在机床数据、PLC数据或者设定数据中的规定，可以限制主轴的最大速度。
<b>H</b>	
<b>主程序</b>	用序号或者名称标志的零件程序，在主程序中可以调用其它的主程序、子程序或者循环。
<b>主程序段</b>	通过„:“引导的程序段，包含在零件程序中启动操作顺序所需要的所有数据。
<b>HIGHSTEP</b>	AS300/AS400系统中PLC所有编程方法的汇编。
<b>辅助功能</b>	在零件程序中，使用辅助功能可以把机床制造商定义的参数传送到PLC中，并释放其所定义功能。
<b>CNC 标准语言</b>	CNC标准语言提供：用户变量，预定义用户变量，系统变量，间接编程，运算功能和三角函数功能，比较运算和逻辑运算，程序跳转和分支，程序协调（SINUMERIK 840D），宏指令。



初始化模块	初始化模块是专用的程序模块。它包含在程序处理之前须执行的赋值。初始化模块主要用于初始化预定义的数据或者全局用户数据。
初始化文件	对应于每个工件可以编制一个初始化文件。在初始化文件中可以编制不同变量的赋值指令，它们仅适用于一个工件。
插补器	NCK的逻辑单元，根据零件程序中目标位置的参数确定进给轴待运行的中间值。
可插补补偿	利用插补补偿功能可以补偿生产过程所决定的丝杠螺距误差和测量系统误差（SSFK,MSFK）。
中断程序	中断程序是专门的子程序，它们可以通过加工过程中的外部事件（外部信号）启动。加工过程中零件程序的程序段被中断，进给轴的中断位置被自动存储。
<b>J</b>	
JOG	控制系统的一种运行方式（调试运行）：在Jog运行方式下，机床可以进行调试。各个进给轴和主轴可以通过方向键点动运行。在Jog手动运行方式中还有其它的一些功能，如回参考点运行，再定位以及预设定（设定实际值）。
<b>K</b>	
通道	一个通道是指可以单独处理一个零件程序，而与其它的通道无关。一个通道仅控制其所分配的进给轴和主轴。不同通道的零件程序其加工过程可以通过同步功能进行协调。
通道结构	利用通道结构可以同时/分开加工各个通道的程序。
补偿轴	设定值或者实际值可以通过补偿值进行修改的轴。
补偿表	支点表。补偿表给基准轴所选择的位置提供补偿轴的补偿值。

补偿值	测量传感器所测得的轴位置与所要求的、编程的轴位置之间的差值。
轮廓	工件的外部轮廓。
轮廓监控	作为轮廓监控的尺寸，滞后量误差控制在一个可定义的公差带之内。比如，当驱动负载过大时就可能产生一个不允许的、过高的滞后量误差。在这种情况下会产生一个报警，从而轴停止运行。
坐标系	参见机床坐标系、工件坐标系。
补偿存储器	控制系统中的一个数据区，刀具补偿数据存储在其中。
圆弧插补	在轮廓上两个固定点之间，刀具以给定的进给量按圆弧运行，从而加工出工件。
$K_{\dot{u}}$	传动比
$K_v$	回路放大系数，调节回路中可调节的物理量。
L	
装载存储器	在SPS的CPU314中，装载存储器就等同于工作存储器
线性轴	与回转轴相反，线性轴指按直线运行的轴。
预见功能	利用预见功能（Look Ahead），可以通过“预见”几个可参数化的程序段而获取加工速度的最优化。

**间隙补偿**

机床在机械方面的间隙补偿，比如滚珠丝杠的反向间隙。对于每个轴，可以分别输入间隙补偿。

**M****宏指令技术**

一个指令名称下汇编一串指令。在程序中，该指令名就代表这一串汇编的指令。

**机床**

系统操作区。

**加工轴**

在机床中表示实际存在的轴。

**机床固定点**

在机床中明确定义的点，比如参考点。

**回机床固定点**

返回到预定义的机床固定点。

**机床坐标系**

以机床轴为基准的坐标系。

**机床零点**

机床固定点，所有测量系统均可以以此点为出发点。

**机床控制面板**

机床中具有各个操作按键、旋钮开关以及各个显示单元如 LEDs 的控制面板，它们通过 PLC 对机床进行控制。

**尺寸系统：公制和英制**

在加工程序中，位置值和螺距值可以用英制编程。控制器设定一个基准系统，它与编程的尺寸系统（G70/G71）无关。

**接地**

接地是指在设备中连接到一起的所有无源器件，它们即使在出现故障时也不会有危险电压。

**MDA**

控制系统的一种运行方式：手动输入，自动运行在 MDA 方式下，可以输入单个程序段或者几个程序段，它们与主程序或者子程序无关，使用 NC 启动键可以立即执行。

**多端口接口**

多端点接口（MPI）是一个9芯的D-Sub接口。

通过多端口接口可以连接一系列设备，相互可以进行通讯：

- 编程器
- 操作与监控系统
- 其它可编程控制器

CPU的“多端口接口MPI”参数组包含有各个参数，用这些参数可以确定多端口接口的性能。

**报警**

零件程序中可编程的所有信息，以及系统可识别的报警均在操作面板上显示，带日期和时间，并有相应的清除标准符号。报警和信息单独显示。

**测量回路**

- SINUMERIK FM-

NC：在缺省情况下，进给轴和主轴所必需的测量回路已经按标准集成到控制模块中。总共可以有4个进给轴和主轴，其中最多为2个主轴。

- SINUMERIK 840D:测量传感器在SIMODRIVE 611D

驱动模块中处理。最大配置可以达到8个进给轴和主轴，其中最多允许5个主轴。

**公制测量系统**

单位均为公制：比如长度为毫米、米。

**N****NC**

数字控制:NC 控制装置包括所有机床控制装置的组件：

->NCK,->PLC,->MMC,->COM.

说明：对于SINUMERIK 840D或者FM-NC控制系统，

CNC控制系统应改为：计算机数控系统。

**NCK**

数字控制核心：NC控制系统部件，执行零件程序，并控制机床的运动过程。

**辅助程序段**

通过“N”引导的程序段，包含一个加工步骤的信息，比如一个位置说明。

**网络**

网络指通过连接电缆连接几个S7-300和其它终端设备，比如一台编程器。通过网络进行相连设备之间的数据交换。

**零点偏移**

在一个坐标系中，相对于目前的零点和框架规定一个新的基准点。

**1. 可设定：SINUMERIK FM-**

**NC:** 每个CNC进给轴可以选择4个相互独立的零点偏移。

**SINUMERIK**

**840D:**对于每个CNC轴，可以设定不同数量的零点偏移。通过G功能可选择的偏移可以选择性地使用。

**2. 外部:**

所有确定工件零点位置的偏移可以通过手轮（DRF偏移）或者通过PLC由一个外部零点偏移覆盖。通过手轮（DRF偏移），或者-通过PLC。

**3. 可编程:**

使用TRANS指令可以给所有的轨迹轴和定位轴编程零点偏移。

**NURBS**

系统内部的运动控制和轨迹插补根据NURBS（Non Uniform Rational B-Splines）进行。这样，在系统内部所有插补均有相同的方法（SINUMERIK 840D）。

**NRK**

数字机器人内核（NCK的运行系统）

**O****OEM**

**SINUMERIK 840D**

给机床制造商提供各种不同应用的使用空间（OEM应用），制造商可以自己设计操作界面或者在系统中开发专用的应用功能。

**定向主轴准停**

比如主轴在一给定角度位置停止，从而可以在某一固定位置进行其它的加工工作。

**定向刀具退回**

**RETOOL:** 当加工过程被停止时（比如刀具折断），刀具可以根据编程指令按照事先给定的方向后撤一段距离。

**倍率**

可以手动或者编程进行工作，允许操作人员覆盖编程的进给或者转速，使加工速度与具体的工件和材料相适应。

## P

## 参数

1. **S7-300:** 可以分为两种参数类型:
  - STEP 7 指令参数  
STEP 7 指令的参数就是待加工的操作数地址或者常数。  
待处理的运算域或者一个常数
  - 一个参数块的参数  
一个参数块的参数确定  
一个模块的性能
2. **840D:**
  - 系统操作区。  
计算参数，可以由零件程序的编程人员在程序中任意设定或者询问。  
在程序中任意设定或者询问。

## 外设模块

用外设模块建立CPU和过程之间的联系。外设模块是：

- 数字量输入/输出模块
- 模拟量输入/输出模块
- 仿真器模块

## PG

编程装置

## PLC

可编程逻辑控制器:存储器可编程控制NC控制系统部件：用于执行机床控制逻辑的转接控制。

## PLC-编程

PLC用软件STEP 7编程。编程软件STEP7基于WINDOWS标准软件、在STEP5编程功能的基础上发展的。

## PLC-编程存储器

- SINUMERIK FM-NC: 在CPU314的PLC用户存储器中，PLC用户程序和用户数据与PLC主程序一起存储。在S7-CPU314中，有一个24K的用户存储器供使用。
- SINUMERIK 840D:在PLC用户存储器中，PLC用户程序和用户数据与PLC主程序一起存储。PLC用户存储器可以通过存储器扩展至96K字节。

## 极坐标

极坐标系指在一个平面中确定一个点的位置，它由到零点的距离与半径矢量和一个轴之间的夹角确定。

多项式插补	用多项式插补功能可以产生不同的曲线，比如 <b>线性函数</b> 、 <b>抛物线函数</b> 和 <b>幂函数</b> （SINUMERIK 840D）。
定位轴	在机床中执行辅助运动的轴（例如刀库，托盘运输）。定位轴不与轨迹轴进行插补。
上电	关机后再次开机。
预设定	使用预设定功能可以在机床坐标系中重新定义系统的零点。在预设定中轴没有运动，它仅仅给当前轴的位置输入一个新的位置值。
型材导轨	型材导轨用于固定S7-300的模块。
程序	<ol style="list-style-type: none"><li>1. 系统操作区。</li><li>2. 到控制系统的连续指令。</li></ol>
程序块	程序块包含零件程序的主程序和子程序。
可编程的工作区域限制	刀具的运行区域限制到一个通过编程限制的区域范围。
可编程的框架	使用编程的框架可以在零件程序加工过程中，动态地定义新的坐标系原点。根据当前的原点，利用一个新框架和附加的确定值，与绝对的确定值加以区分。
编程码	编程码是一种字符和字符串，它们在零件程序的编程语言中具有确定的含义（参见编程说明）。
CNC编程语言	CNC编程语言的基础是DIN66025,带高级语言扩展。此外，CNC高级语言和编程语言允许使用宏指令定义（单个指令的汇编）。
备份电池	利用备份电池保证在电网掉电时，用户程序可以安全地存放在CPU中，并且确定的数据区以及剩余的标志位、定时器和计数器可以保持。

## Q

## 象限误差补偿

在象限过渡时，由于在导轨面上出现不同的摩擦而引起的轮廓误差，可以通过象限误差补偿予以消除。象限误差补偿的参数可以通过圆弧形状测试确定。

## R

## 参考点

机床中的一点，加工轴的测量系统以此为基准。

## 回参考点

如果所使用的位移测量系统没有绝对值编码器，则必须要回参考点运行，从而保证测量系统所提供的实际值与机床坐标值相一致。

## 剩磁

剩磁是指数据块中的数据区以及定时器、计数器和标志位在新启动时，或者在掉电时不会丢失。

## REPOS

1. 通过操作返回轮廓。使用Repos功能可以通过方向键再次回到中断点。
2. 通过程序返回轮廓。通过编程指令，可以选择几种不同的返回：返回到中断点，返回到程序段起始点，返回到程序段终点，返回到程序段起始点和中断点之间的一个轨迹点。

## 毛坯

毛坯指用于工件加工的原材料。

## 翻转

框架的一个部分，定义坐标系按照一定的角度进行旋转。

## R 参数

计算参数，可以由零件程序编程人员在程序中进行任意设定或者询问。

## 回转轴

回转轴指工件或者刀具旋转到一个给定的角度位置。

## 回转轴无限旋转

根据具体的应用场合，回转轴可以旋转小于360度，或者在两个方向无限旋转。无限旋转的回转轴，比如可以用于非圆加工、磨削加工和绕线加工。



倒圆轴	倒圆轴指工件或者刀具旋转到一个分度头给定的角度位置。到达分度头刻度后，倒圆轴“到达位置”。
<b>S</b>	
<b>S7-300 总线</b>	S7-300总线是一个串连数据总线，通过该数据总线模块可以相互进行通讯，同时总线自身提供电源。模块之间的联系通过总线连接器建立。
<b>S7配置</b>	S7配置是一个工具，用此工具可以给模块设定参数。使用S7配置可以在编程器上设定CPU和外设模块的各个参数块。 PG已设定。这些参数传送到CPU中。
程序段	零件程序的一个部分，换行后结束。分为主程序段和辅助程序段。
程序段查找过程	在进行零件程序测试时或者在中断一个加工后，可以通过程序段搜索功能找到程序中的任意位置，在此位置加工可以启动或者继续。
关键字	有确定写法的字，它们在编程语言中具有所定义的含义。
钥匙开关	<ol style="list-style-type: none"><li>1. <b>S7-300</b>: 钥匙开关是CPU的运行方式开关。 钥匙开关的操作通过一个可以插拔的钥匙进行。</li><li>2. <b>840D/FM-NC</b>: 机床控制面板上的钥匙开关有4个位置， 它们由控制器的操作系统分配相应的功能。钥匙开关有3个不同颜色的开关，它们可以在所给定的位置插拔。</li></ol>
刀尖半径补偿	在编程一个轮廓时，往往从刀具的尖端计算。但是，这在实际加工过程中并不可以实现，因为所使用的刀具会有一个弯曲半径，系统必须要考虑这个值。在此计算的加工点就位于其中心点，距离为半径的长度。
快速提刀	当中断加工时，可以通过CNC加工程序引入一个动作，使刀具从所加工的工件轮廓快速离开。此外还可以设定退刀的角度和位移的参数。在快速提刀以后可以另外执行一个中断程序。 (SINUMERIK FM-NC, 840D)。

**快速数字输入/输出**

通过数字输入端可以启动快速CNC程序（中断程序）。通过数字输出端可以释放快速的、程序控制的开关功能。（SINUMERIK 840D）

**斜面加工**

在工件表面进行钻削和铣削加工，它们不在机床坐标平面，但是可以通过“斜面加工”功能很方便地实现。

**螺旋线插补**

螺旋线插补特别适用于利用成形铣刀简单地加工内螺纹和外螺纹，以及铣削润滑槽。在这里螺旋线由两个运动组成：

1. 平面中的回转运动
2. 与此平面垂直的直线运动

**步进尺寸**

通过相对尺寸说明加工行程。相对尺寸可以作为设定数据存储，或者通过相应的增量键10、100、1000和10000进行选取。

**保护区**

在加工区之内的一个三维空间，刀尖不可以进入此区域。

**串行接口RS232**

在数据进行输入/输出时，

- 在MMC模块MMC100上有一个串行接口RS232，
- 在MMC模块MMC101和MMC102中有两个RS232接口。

可用。通过该接口可以装载和保护加工程序以及制造商和用户数据。

**设定数据**

设定数据确定机床的性能，按照系统软件定义的方法在系统中设定。

**安全功能**

系统中所具有的安全监控功能，通过安全监控功能数控系统中以及PLC和机床中的故障均可以尽早地予以识别，从而排除一切对工件、刀具或者机床可能造成的危害。在故障发生时，加工过程会中断，驱动停止，故障原因被存储并作为报警显示。同时通知PLC数控系统有一报警。

仿真器模块	仿真器模块是一种模块， <ul style="list-style-type: none"><li>• 通过操作部件可以模拟数字输入量，</li><li>• 显示数字输出量</li></ul>
比例尺	是构成框架的一个部分，可以改变某个轴的比例尺。
软键	软键在屏幕上显示，具有对应的区域，可以动态地与当前的操作情形相对应。这些功能键（软键）可以自由分配，它们由软件按照定义的功能进行分配。
软件限位开关	软件限位开关限制一个轴的移动范围，阻止滑枕冲撞硬件限位开关。每个轴可以给定两组数值，它们可以由PLC分别激活。
存储器可编程控制系统	存储器可编程的控制系统（SPS）是电子控制系统，它们的功能以程序的形式存储到控制器中。因此，控制器的结构和布线与控制系统的功能无关。存储器可编程的控制系统具有计算机的结构，它由带存储器的CPU（中央模块）、输入/输出模块和内部总线系统构成。外设和编程语言以控制技术为准。
镜像	使用镜像功能，使加工轮廓相关轴的坐标值符号相反。可以同时多个轴进行镜像。
主轴	主轴功能分为两种功率级别： <ol style="list-style-type: none"><li>1. 主轴：转速控制或者位置控制的主轴驱动，模拟量，数字量（SINUMERIK 840D）</li><li>2. 辅助主轴：转速控制的主轴驱动，功能包“辅助主轴”比如用于驱动的工具。</li></ol>
丝杠螺距误差补偿	滚珠丝杠在进给时产生机械误差，由控制系统通过存储的误差测量值进行补偿。
样条插补	通过样条插补，控制系统可以由理论轮廓上较少的、给定的支点生成一条光滑的曲线。

语言	操作界面的显示文本和系统信息、报警可以有五种语言（磁盘）： <b>德语，英语，法语，意大利语和西班牙语。</b> 在系统中可以选择并同时安装以上语言中的两种。
SPS	➔ 存储器可编程控制
标准循环	对于经常出现的加工情形，可以使用标准循环： <ul style="list-style-type: none"><li>• 适用于钻削/铣削</li><li>• 适用于车削（SINUMERIK FM-NC）</li></ul> 在“程序”操作区菜单“循环”下，列出所有供使用的循环清单。选择了所要求的加工循环后，屏幕上会显示参数赋值指令中必须设定的参数。
同步轴	同步轴运行时间与几何轴相同。
同步动作	<ol style="list-style-type: none"><li>1. 辅助功能输出 在工件加工期间，可以把工艺功能（辅助功能）从CNC程序中输出到PLC中。通过辅助功能可以控制机床的附加设备，比如顶尖套筒，夹持器，卡盘等等。</li><li>2. 快速辅助功能输出 对于时间较紧的开关功能，可以减少辅助功能的应答时间，避免加工过程不必要的停顿。</li></ol>
同步	零件程序中的指令，用于协调同一加工地点时不同通道中的加工过程。
系统存储器	系统存储器是CPU中的一个存储器，其内容为： <ul style="list-style-type: none"><li>• 操作系统所需要的数据</li><li>• 运算的定时器、计数器和标志位</li></ul>
系统变量	无需程序员的工作，已经存在的变量。它由数据类型和变量名定义，变量名之前有符号\$。参见用户定义的变量。

## T

## 示教

使用**示教功能**可以编制或者修改零件程序。各个程序段可以通过键盘输入，并可立即运行。通过方向键或者手轮运行的位置也可以存储。附加数据，如**G**功能、进给率或者**M**概念可以输入到同一个程序段中。

## 零件程序

**NC**控制系统中的连续指令，它们一起加工出确定的工件。也就是说在一个所提供的毛坯上进行一定的加工。

## 零件程序管理

零件程序可以按照工件管理。用户存储器的尺寸确定所管理的程序和数据数量。每个文件（程序和数据）可以命名最多**24**个字母数据字符的名称。

## 用户号码

如果几个用户通过网络进行通讯，则用户号码表示一个**CPU**或者编程器的“动作地址”，或者一个其它的智能外设模块的“动作地址”。使用**S7**工具“**S7-配置**”给**CPU**或者编程器赋值一个用户号码。

## 文本编辑器

编辑器

## 工具

一个工具是指用于输入和修改参数组参数的软件工具。工具如：

- **S7**配置
- **S7-TOP**
- **S7-信息**

## 转换

在一个直角坐标系中编程，在一个非直角坐标系中加工（比如加工轴作为回转轴）。

## U

## 子程序

一个子程序的连续指令，它们可以通过设定不同的参数反复调用。子程序从主程序中调用。没有授权的读取和显示会被子程序禁止。循环是子程序的一种。

**清零**

在清零时，CPU中以下的存储器将被清零：

- 工作存储器
- 装载存储器的读写区
- 系统存储器
- 备份存储器
- 

**V****变量定义**

定义变量时，包括确定数据类型和变量名。使用该变量名，也就是调用该变量值。

**连接电缆**

连接电缆指预制的或者由用户自己定制的两芯电缆，带两个插头。连接电缆通过多端口接口（MPI）把CPU与编程器或者其它CPU相连。

**运行范围**

线性轴中最大允许的运行范围可以达到 $\pm 9$ 位。绝对值取决于所选择的输入单位和位置控制单位，以及单位制（英制或者公制）。

**轮廓破坏预先识别**

控制系统识别和通报以下的轮廓冲突情形：

1. 轨迹行程短于刀具半径。
2. 内角的宽度小于刀具直径。

**预控制，动态**

滞后量误差所决定的轮廓误差，几乎可以通过动态的、由加速度决定的预控制消除。由此可以获得一个非常好的加工精度，即使是在轨迹速度很高的情况下。预控制可以通过零件程序根据相应的轴选择或者撤销选择。

**进给倍率**

通过机床控制面板或者PLC可以调节实际速度，并覆盖编程的速度（0—200%）。另外，进给速度也可以在加工程序中，通过一个编程的百分比（1—200%）进行修改。

**W****工件**

机床待加工的零件。

**工件轮廓**

待加工工件的给定轮廓。

工件坐标系	工件坐标系中有工件零点。在工件坐标系中编程时，尺寸和方向以工件坐标系为基准。
工件零点	工件零点构成了工件坐标系的原点。它由与机床零点的距离定义。
刀具	机床中进行加工的部件，诸如车刀、铣刀、钻头、激光...
刀具补偿	在程序段中编程一个 <b>T功能</b> （5位整数）可以选择刀具。每个T号可以最多有9个刀沿（D地址）。控制系统中所管理的刀具数量可以通过设计进行修改。
在刀具半径补偿时	为了可以直接编程一个所要求的工件轮廓，控制系统必须考虑所使用刀具的半径，与编程的轮廓等距离运行。（G41/G42）
<b>X</b>	
<b>Y</b>	
<b>Z</b>	
反比时间进给	在SINUMERIK FM-NC和840D中，可以编程一个程序段位移所需要的时间（G93），而不用编程轴的进给速度。
英制尺寸系统	长度以“英寸”及其导出单位为尺寸的尺寸系统。
存取权限	CNC程序块和数据通过一个7级存取权限进行保护。 <ul style="list-style-type: none"><li>• 三个口令字，分别用于系统生产厂家、机床制造商和用户，以及</li><li>• 4个钥匙开关位置，可以由PLC进行利用</li></ul>

**中间程序段**

带刀具补偿（G41/G42）的加工过程可以由一定数量的中间程序段（在补偿级的程序段，没有轴运动）中断，这样刀具补偿还可以进行正确地计算。先于控制系统读出所允许的中间程序段数量，可以通过系统参数设定。

**循环辅助**

在“程序”操作区菜单“循环”下，列出所有供使用的循环清单。选择了所要求的加工循环后，屏幕上会显示参数赋值指令中必须设定的参数。

**循环**

受保护的子程序，用于执行工件中反复出现的加工过程。



## C 参考文献

- /BU/** 一般文献  
SINUMERIK & SIMODRIVE, 加工机床的自动化系统  
目录 NC 60  
订货号: E86060-K4460-A101-A9  
订货号: E86060-K4460-A101-A9-7600 (英语)
- /IKPI/** 工业用通讯和现场设备  
目录 IK PI  
订货号: E86060-K6710-A101-B2  
订货号: E86060-K6710-A101-B2-7600 (英语)
- /ST7/** SIMATIC  
全集成自动装置和微型自动装置产品  
目录 ST 70  
订货号: E86060-K4670-A111-A8  
订货号: E86060-K4670-A111-A8-7600 (英语)
- /ZI/** MOTION-CONNECT  
SIMATIC、SINUMERIK、MASTERDRIVES 和 SIMOTION 连接技术  
& 系统组件  
目录 NC Z  
订货号: E86060-K4490-A001-B1  
订货号: E86060-K4490-A001-B1-7600 (英语)
- /CD1/** 电子文献  
SINUMERIK 系统 (版本 03.04)  
DOC ON CD  
(带所有 SINUMERIK 840D/840Di/810D/802 和  
SIMODRIVE 文件)  
订货号: 6FC5298-7CA00-0AG0

## 用户文献

/AUK/	SINUMERIK 840D/810D 操作 <b>AutoTurn</b> 简明手册 订货号: 6FC5298-4AA30-0AP2	(版本 09.99)
/AUP/	SINUMERIK 840D/810D 编程 / 建立 <b>AutoTurn</b> 图形编程系统操作说明 订货号: 6FC5298-4AA40-0AP3	(版本 02.02)
/BA/	SINUMERIK 840D/810D 操作说明 <b>MMC</b> 订货号: 6FC5298-6AA00-0AP0	(版本 10.00)
/BAD/	SINUMERIK 840D/840Di/810D 操作说明 <b>HMI Advanced</b> 订货号: 6FC5298-6AF00-0AP3	(版本 03.04)
/BAH/	SINUMERIK 840D/840Di/810D <b>HT 6</b> 操作说明 订货号: 6FC5298-0AD60-0AP3	(版本 03.04)
/BAK/	SINUMERIK 840D/840Di/810D 操作简要说明 订货号: 6FC5298-6AA10-0AP0	(版本 02.01)
/BAM/	SINUMERIK 810D/840D <b>ManualTurn</b> 操作/编程 订货号: 6FC5298-6AD00-0AP0	(版本 08.02)
/BAS/	SINUMERIK 840D/840Di/810D <b>ShopMill</b> 操作/编程 订货号: 6FC5298-6AD10-0AP3	(版本 10.04)
/BAT/	SINUMERIK 840D/810D <b>ShopTurn</b> 操作/编程 订货号: 6FC5298-6AD50-0AP2	(版本 06.03)
/BEM/	SINUMERIK 840D/810D 操作说明 订货号: 6FC5298-6AC00-0AP3	(版本 03.04)
/BNM/	SINUMERIK 840D840Di//810D 测量循环用户手册 订货号: 6FC5298-7AA70-0AP1	(版本 10.04)
/BTDI/	SINUMERIK 840D840Di//810D 运动控制信息系统 (MCIS) 刀具数据信息用户手册 订货号: 6FC5297-6AE01-0AP1	(版本 06.04)
/CAD/	SINUMERIK 840D/840Di/810D <b>CAD-Reader</b> 操作说明 订货号: (是在线帮助的组成部分)	(版本 03.02)

<b>/DA/</b>	SINUMERIK 840D/840Di/810D 诊断说明 订货号: 6FC5298-6AA20-0AP4	(版本 03.04)
<b>/KAM/</b>	SINUMERIK 840D/810D ManualTurn 简要说明 订货号: 6FC5298-5AD40-0AP0	(版本 04.01)
<b>/KAS/</b>	SINUMERIK 840D/810D ShopMill 简要说明 订货号: 6FC5298-5AD30-0AP0	(版本 04.01)
<b>/KAT/</b>	SINUMERIK 840D/810D ShopTurn 简要说明 订货号: 6FC5298-6AF20-0AP0	(版本 07.01)
<b>/PG/</b>	SINUMERIK 840D/840Di/810D 基本原理编程说明 订货号: 6FC5298-7AB00-0AP0	(版本 03.04)
<b>/PGA/</b>	SINUMERIK 840D/840Di/810D 工作准备编程说明 订货号: 6FC5298-7AB10-0AP0	(版本 03.04)
<b>/PGA1/</b>	SINUMERIK 840D/840Di/810D 系统变量列表手册 订货号: 6FC5298-7AE10-0AP0	(版本 03.04)
<b>/PGK/</b>	SINUMERIK 840D/840Di/810D 编程简要说明 订货号: 6FC5298-7AB30-0AP0	(版本 03.04)
<b>/PGM/</b>	SINUMERIK 840D/840Di/810D ISO Milling 编程指南 订货号: 6FC5298-6AC20-0BP2	(版本 11.02)
<b>/PGT/</b>	SINUMERIK 840D/840Di/810D ISO Turning 编程指南 订货号: 6FC5298-6AC10-0BP2	(版本 11.02)
<b>/PGZ/</b>	SINUMERIK 840D/840Di//810D 循环编程说明 订货号: 6FC5298-7AB40-0AP1	(版本 10.04)
<b>/PI/</b>	PCIN 4.4 用于 MMC 模块数据传输的软件 订货号: 6FX2060-4AA00-4XB0 (德语、英语和法语) 订货地点: WK Fürth	
<b>/SYI/</b>	SINUMERIK 840Di 系统一览 订货号: 6FC5298-6AE40-0AP0	(版本 02.01)

## 制造商/维修文献

## a) 列表

**/LIS/** SINUMERIK 840D/840Di/810D (版本 03.04)  
SIMODRIVE 611D  
列表  
订货号: 6FC5297-7AB70-0AP0

## b) 硬件

**/ASAL/** SIMODRIVE (版本 10.03)  
异步电机通用部件设计说明  
订货号: 6SN1197-0AC62-0AP0

**/APH2/** SIMODRIVE (版本 10.03)  
设计说明 异步电机 **1PH2**  
订货号: 6SN1197-0AC63-0AP0

**/APH4/** SIMODRIVE (版本 10.03)  
设计说明 异步电机 **1PH4**  
订货号: 6SN1197-0AC64-0AP0

**/APH7S/** SIMODRIVE (版本 04.04)  
设计说明 异步电机 **1PH7**  
订货号: 6SN1197-0AC65-0AP0

**/APL6/** MASTERDRIVES VC/MC (版本 03.04)  
设计说明 异步电机 **1PL6**  
订货号: 6SN1197-0AC67-0AP0

**/BH/** SINUMERIK 840D/840Di//810D (版本 04.04)  
操作部件手册  
订货号: 6FC5297-7AA50-0AP0

**/BHA/** SIMODRIVE 传感器 (版本 03.04)  
用户手册 (HW) 带有 **PROFIBUS-DP 绝对值传感器**  
订货号: 6SN1197-0AB10-0YP3

**/EMV/** SINUMERIK, SIROTEC, SIMODRIVE (版本 03.04)  
设计说明 **EMV-安装规范**  
订货号: 6FC5297-0AD30-0AP2

欲知最新的一致性说明, 请访问  
<http://www4.ad.siemens.de>

请于此处输入

ID号: 在“Suche (查找)”栏中输入15257461 (右上)  
并点击“go”。

**/GHA/** SINUMERIK/ SIMOTION (版本 02.03)  
**ADI4 – 模拟量驱动接口, 4 根轴**  
设备手册  
订货号: 6FC5297-0BA01-0AP1

<b>/PFK6/</b>	SIMODRIVE 611, SIMOVERT MASTERDRIVES (版本 05.03)  交流伺服电机 <b>1FK6</b> 订货号: 6SN1197-0AD05-0AP0	
<b>/PFK7/</b>	SIMODRIVE 611, SIMOVERT MASTERDRIVES (版本 01.03) 设计说明 交流伺服电机 <b>1FK7</b> 订货号: 6SN1197-0AD06-0AP0	
<b>PFS6/</b>	SIMOVERT MASTERDRIVES (版本 07.03) 设计说明 交流伺服电机 <b>1FS6</b> 订货号: 6SN1197-0AD08-0AP0	
<b>/PFT5/</b>	SIMODRIVE (版本 02.04) 设计说明 交流伺服电机 <b>1FT5</b> 订货号: 6SN1197-0AD01-0AP0	
<b>/PFT6/</b>	SIMODRIVE 611, SIMOVERT MASTERDRIVES (版本 01.04) 同步伺服电机 <b>1FT6</b> 设计说明 订货号: 6SN1197-0AD02-0AP0	
<b>/PFU/</b>	SINAMICS, SIMOVERT MASTERDRIVES, (版本 09.03) MICROMASTER <b>SIEMOSYN 电机 1FU8</b> 订货号: 6SN1197-0AC80-0AP0	
<b>/PHC/</b>	SINUMERIK 810D (版本 11.02) 设计 (HW) 手册 订货号: 6FC5297-6AD10-0AP1	
<b>/PHD/</b>	SINUMERIK 840D (版本 11.03) 设计 (HW) 手册 订货号: 6FC5297-6AC10-0AP3	
<b>/PJAL/</b>	SIMODRIVE 611, SIMOVERT MASTERDRIVES MC (版本 01.04) 设计说明 同步伺服电机 一般说明 <b>1FT- / 1FK-电机</b> 订货号: 6SN1197-0AD07-0AP1	
<b>/PJFE/</b>	SIMODRIVE (版本 02.03) 设计说明 同步内装式电机 <b>1FE1</b> 用于主轴驱动的三相电流电机 订货号: 6SN1197-0AC00-0AP4	
<b>/PJF1/</b>	SIMODRIVE (版本 12.02) 同步内装式电机 <b>1FE1 051.-1FE1 147</b> 。安装说明 用于主轴驱动的三相电流电机 订货号: 610.43000.02	
<b>/PJLM/</b>	SIMODRIVE (版本 06.02) 直线电机 <b>1FN1, 1FN3</b> 设计说明 所有 直线电机一般说明 <b>1FN1</b> 交流直线电机 <b>1FN1</b> <b>1FN3</b> 交流直线电机 <b>1FN3</b> <b>CON</b> 连接技术 订货号: 6SN1197-0AB70-0AP3	

/PJM/	SIMODRIVE 611, SIMOVERT MASTERDRIVES (版本 11.00) 设计说明 <b>伺服电机</b> 交流电机用于进给和主轴驱动 订货号: 6SN1197-0AA20-0AP4
/PJM2/	SIMODRIVE (版本 07.03) 设计说明 <b>伺服电机</b> 三相交流伺服电机用于进给和主轴驱动 订货号: 6SN1197-0AC20-0AP0
/PJTM/	SIMODRIVE (版本 11.03) 设计说明 <b>内装式扭矩电机 1FW6</b> 设计说明 订货号: 6SN1197-0AD00-0AP2
/PKTM/	SIMODRIVE, SIMOVERT MASTERDRIVES (版本 03.04) <b>整套扭矩电机 1FW3</b> 设计说明 订货号: 6SN1197-0AC70-0AP0
/PJU/	SIMODRIVE 611 (版本 10.04) 设计说明 <b>变频器</b> 订货号: 6SN1197-0AA00-0AP7
/PMH/	SIMODRIVE 传感器 (版本 07.02) <b>空心轴测量系统 SIMAG H</b> 设计/安装说明 订货号: 6SN1197-0AB30-0AP1
/PMH2/	SIMODRIVE 传感器 (版本 03.04) <b>空心轴测量系统 SIMAG H2</b> 设计/安装说明 订货号: 6SN1197-0AB31-0AP1
/PMHS/	SIMODRIVE (版本 12.00) <b>主轴驱动测量系统</b> 安装说明 <b>齿轮传感器 SIZAG2</b> 订货号: 6SN1197-0AB00-0YP3
/PMS/	SIMODRIVE (版本 03.04) 设计说明 <b>用于主轴驱动的 ECO-电机主轴</b> 订货号: 6SN1197-0AD04-0AP1
PPH/	SIMODRIVE (版本 12.01) <b>1PH2/1PH4/1PH7</b> 电机设计说明 用于主轴驱动的三相电流异步电机 订货号: 6SN1197-0AC60-0AP0
/PPM/	SIMODRIVE (版本 11.01) 设计说明 用于主轴驱动的空心轴电机 <b>1PM4</b> 和 <b>1PM6</b> 订货号: 6SN1197-0AD03-0AP0

c) 软件  
/FB1/

SINUMERIK 840D/840Di/810D/FM-NC (版本 03.04)

## 基本机床 (部分 1) 功能描述

(下面列出包含的书籍)

订货号: 6FC5297-7AC20-0AP0

A2	不同的接口信号
A3	轴监控, 保护区
B1	轨迹控制运行、准停和预见功能
B2	加速度
D1	诊断辅助方法
D2	对话编程
F1	运行到固定挡块
G2	速度、给定值系统/实际值系统、调节
H2	辅助功能输出到 PLC
K1	BAG、通道、程序运行
K2	轴、坐标系、框架、 工件实际值系统, 外部 零点偏移
K4	通讯
N2	紧急关闭
P1	端面轴
P3	PLC 基本程序
R1	返回参考点运行
S1	主轴
V1	进给
W1	刀具补偿

## /FB2/

SINUMERIK 840D/840Di/810D (版本 03.04)

## 功能描述 扩展功能 (部分 2)

包括 FM-NC: 转动, 步进电机

(下面列出包含的书籍)

订货号: 6FC5297-7AC30-0AP0

A4	数字和模拟NCK外设
B3	几个操作面板和NCU
B4	通过PG/PC操作
F3	远程诊断
H1	手动运行和手轮运行
K3	补偿
K5	BAG, 通道, 轴更换
L1	FM-NC 本地总线
M1	运动关系转换
M5	测量
N3	软件凸轮、行程开关信号
N4	冲裁和步冲
P2	定位轴
P5	摆动
R2	回转轴
S3	同步主轴
S5	同步措施 (至 SW 3 / 其后版本 /FBSY/)
S6	步进电机控制
S7	存储器配置
T1	分度轴
W3	刀具更换
W4	磨削

- /FB3/** SINUMERIK 840D/840Di/810D (版本 03.04)  
**特殊功能 (部分 3)** 功能描述  
 (下面列出包含的书籍)  
 订货号: 6FC5297-7AC80-0AP1  
 F2 3 轴 5 轴转换  
 G1 龙门架轴  
 G3 周期时间  
 K6 轮廓隧道监控  
 M3 轴耦合和 ESR  
 S8 恒定工件转数, 用于无心磨削  
 S9 给定值切换 (S9)  
 T3 正切控制  
 TE0 安装和激活编译循环  
 TE1 距离调节  
 TE2 模拟输出轴  
 TE3 主从转速/转矩耦合  
 TE4 坐标转换包  
 TE5 给定值切换  
 TE6 MCS 耦合  
 TE7 再次装备 —— 折返支持  
 TE8 与周期无关的轨迹同步开关信号输出  
 V2 预处理  
 W5 三维刀具半径补偿
- /FBA/** SIMODRIVE 611D/SINUMERIK 840D/810D (版本 03.04)  
**驱动功能** 功能描述  
 (下面列出包含的章节)  
 订货号: 6SN1197-0AA80-1AP1  
 DB1 运行提示信息/报警反应  
 DD1 诊断功能  
 DD2 转速控制回路  
 DE1 扩展的驱动功能  
 DF1 使能  
 DG1 传感器设定参数  
 DL1 直线电机的MD  
 DM1 计算电机参数/功率部件参数和调节器参数  
 DS1 电流调节回路  
 DÜ1 监控/限制
- /FBAN/** SINUMERIK 840D/SIMODRIVE 611 数字 (版本 02.00)  
 功能描述 **ANA-模块**  
 订货号: 6SN1197-0AB80-0AP0
- /FBD/** SINUMERIK 840D (版本 07.99)  
**数字化** 功能描述  
 订货号: 6FC5297-4AC50-0AP0  
 DI1 开机调试  
 DI2 用周期传感器扫描 (scancad 扫描)  
 DI3 用激光扫描 (scancad 激光)  
 DI4 铣削程序编制 (scancad 铣刀)
- /FBDM/** SINUMERIK 840D/840Di/810D (版本 09.03)  
 功能描述 NC 程序管理  
 DNC 机床  
 订货号: 6FC5297-1AE81-0AP0



<b>/FBDN/</b>	SINUMERIK 840D/840Di/810D 运动控制信息系统 (MCIS) <b>NC 程序管理 DNC</b> 功能描述 订货号: 6FC5297-1AE80-0AP0 DN1 DNC Plant / DNC Cell DN2 DNC IFC SINUMERIK, NC-数据监控通过网络	(版本 03.03)
<b>/FBFA/</b>	SINUMERIK 840D/840Di/810D 功能描述 <b>ISO-语言</b> , 用于 <b>SINUMERIK</b> 订货号: 6FC5297-6AE10-0AP3	(版本 11.02)
<b>/FBFE/</b>	SINUMERIK 840D/810D 功能描述 <b>远程诊断</b> 订货号: 6FC5297-0AF00-0AP2 FE1 远程诊断 ReachOut FE3 远程诊断 pcAnywhere	(版本 04.03)
<b>/FBH/</b>	SINUMERIK 840D/840Di/810D <b>HMI-编程软件包</b> 订货号: (是软件供货的组成部分)  部分 1 用户说明 部分 2 功能描述	(版本 02.04)
<b>/FBH1/</b>	SINUMERIK 840D/840Di/810D <b>HMI-编程软件包</b> <b>ProTool/Pro Option SINUMERIK</b> 订货号: (是软件供货的组成部分)	(版本 03.03)
<b>/FBHL/</b>	SINUMERIK 840D/SIMODRIVE 611 DIGITAL <b>HLA 模块</b> 功能描述 订货号: 6SN1197-0AB60-0AP3	(版本 10.03)
<b>/FBIC/</b>	SINUMERIK 840D/840Di/810D 运动控制信息系统 (MCIS) <b>TDI Ident Connection</b> 功能描述 订货号: 6FC5297-1AE60-0AP0	(版本 06.03)
<b>/FBMA/</b>	SINUMERIK 840D/810D <b>ManualTurn</b> 功能描述 订货号: 6FC5297-6AD50-0AP0	(版本 08.02)
<b>/FBO/</b>	SINUMERIK 840D/810D <b>操作界面 OP 030</b> 设计功能描述 (下面列出包含的章节) 订货号: 6FC5297-6AC40-0AP0 BA 操作说明 EU 开发环境 (设计包) PSE 操作面板设计入门 (IK 安装包软件升级和配置	(版本 09.01)
<b>/FBP/</b>	SINUMERIK 840D <b>C-PLC 编程</b> 功能描述 订货号: 6FC5297-3AB60-0AP0	(版本 03.96)

/FBR/	SINUMERIK 840D/840Di/810D 功能描述 <b>计算机耦合器 (SinCOM)</b> 订货号: 6FC5297-6AD60-0AP1 NFL 到生产主机的接口 NPL 到 PLC/NCK的接口	(版本 01.04)
/FBSI/	SINUMERIK 840D/SIMODRIVE SINUMERIK <b>安全集成功能描述</b> 订货号: 6FC5297-6AB80-0AP2	(版本 11.03)
/FBSP/	SINUMERIK 840D/840Di/810D <b>ShopMill</b> 功能描述 订货号: 6FC5297-6AD80-0AP2	(版本 08.03)
/FBST/	SIMATIC <b>FM STEPDRIVE/SIMOSTEP</b> 功能描述 订货号: 6SN1197-0AA70-0YP4	(版本 01.01)
/FBSY/	SINUMERIK 840D/810D <b>同步措施</b> 功能描述 订货号: 6FC5297-7AD40-0AP2	(版本 03.04)
/FBT/	SINUMERIK 840D/810D <b>ShopTurn</b> 功能描述 订货号: 6FC5297-6AD70-0AP2	(版本 03.04)
/FBTC/	SINUMERIK 840D/810D IT-Solutions <b>刀具数据通信 SinTDC</b> 功能描述 订货号: 6FC5297-5AF30-0AP0	(版本 01.02)
/FBTD/	SINUMERIK 840D/810D IT-Solutions <b>刀具需求确定 (SinTDI)</b> 功能描述 通过在线帮助 订货号: 6FC5297-6AE00-0AP0	(版本 02.01)
/FBTP/	SINUMERIK 840D/840Di/810D 运动控制信息系统 (MCIS) <b>预防维护 TPM</b> 功能描述 订货号: 文件是软件的一部分	(Ausgabe 01.03)
/FBU/	SIMODRIVE 611 universal/universal E <b>转数控制和定位的控制组件</b> 功能描述  订货号: 6SN1197-0AB20-1AP1	(版本10.04)
/FBU2/	SIMODRIVE 611 <b>universal</b> 安装说明 (在 SIMODRIVE 611 universal 中)	(版本 04.02)
/FBW/	SINUMERIK 840D/810D 功能描述 <b>刀具管理</b> 订货号: 6FC5297-6AC60-0AP1	(版本 03.04)
/HBA/	SINUMERIK 840D/840Di/810D 手册 <b>@Event</b> 订货号: 6AU1900-0CL20-0AA0	(版本 03.02)

<b>/HBI/</b>	SINUMERIK 840Di <b>SINUMERIK 840Di</b> 手册 订货号: 6FC5297-7AE60-0AP0	(版本 03.04)
<b>/INC/</b>	SINUMERIK 840D840Di//810D 系统描述 开机调试工具 <b>SINUMERIK SinuCOM NC</b> 订货号: (IBN 工具在线帮助的组成部分)	(版本 06.03)
<b>/PJE/</b>	SINUMERIK 840D/810D <b>HMI Embedded 设计包</b> 功能描述 软件更新, 配置, 安装 订货号: 6FC5297-6EA10-0AP0 (文字 PS 设计句法是软件供货的一部分 并作为 Pdf 文件提供)	(版本 08.01)
<b>/POS1/</b>	SIMODRIVE <b>POSMO A</b> <b>PROFIBUS DP 上分位置马达</b> 用户手册 订货号: 6SN2197-0AA00-0AP7	(版本 08.04)
<b>/POS2/</b>	SIMODRIVE <b>POSMO A</b> 安装说明 (位于每个 POSMO A)	(版本 05.03)
<b>/POS3/</b>	SIMODRIVE POSMO SI/CD/CA 分伺服驱动技术用户手册 订货号: 6SN2197-0AA20-0AP7	(版本 10.04)
<b>/POS4/</b>	SIMODRIVE <b>POSMO SI</b> 安装说明 (在每个 POSMO SI 旁)	(版本 08.03)
<b>/POS5/</b>	SIMODRIVE <b>POSMO CD/CA</b> 安装说明 (位于某个 POSMO CD/CA 上)	(版本 08.03)
<b>/S7H/</b>	SIMATIC S7-300 <b>技术功能</b> 安装手册 订货号: 6ES7398-8AA03-8AA0 - 参考手册: CPU 数据 (HW 描述) - 参考手册: 模块数据	(版本 2002)
<b>/S7HT/</b>	SIMATIC S7-300 手册 <b>STEP 7, 基本知识, V. 3.1</b> 订货号: 6ES7810-4CA02-8AA0	(版本 03.97)
<b>/S7HR/</b>	SIMATIC S7-300 手册 <b>STEP 7, 参考手册, V. 3.1</b> 订货号: 6ES7810-4CA02-8AR0	(版本 03.97)
<b>/S7S/</b>	SIMATIC S7-300 <b>FM 353 用于步进电机</b> 位置组件 连同设计包一起订购	(版本 04.02)
<b>/S7L/</b>	SIMATIC S7-300 <b>FM 354 用于伺服电机</b> 位置组件 连同设计包一起订购	(版本 04.02)
<b>/S7M/</b>	SIMATIC S7-300 <b>多重组件 FM 357.2</b> 用于伺服驱动或者步进驱动 与设计软件包一起订购	(版本 01.03)

/SP/	SIMODRIVE 611-A/611-D <b>SimoPro 3.1</b> 用于设计机床驱动机构的程序 订货号: 6SC6111-6PC00-0AA# 订货地: WK Fürth
d) 安装调试	
/BS/	SIMODRIVE 611 analog (版本 10.00) 用于主轴和异步电机模块 版本 <b>3.20</b> 开机调试软件描述 订货号: 6SN1197-0AA30-0AP1
/IAA/	SIMODRIVE 611 analog (版本 10.00) 开机调试说明 订货号: 6SN1197-0AA60-0AP6
/IAC/	SINUMERIK 810D (版本 11.02) 安装调试说明 (包括 SIMODRIVE 611D 开机调试软件描述) 订货号: 6FC5297-6AD20-0AP1
/IAD/	SINUMERIK 840D/SIMODRIVE 611D (版本 03.04) 开机调试说明 (包括 SIMODRIVE 611D 开机调试软件描述) 订货号: 6FC5297-7AB10-0AP0
/IAM/	SINUMERIK 840D/840Di/810D (版本 03.04) <b>HMI</b> 开机调试说明 订货号: 6FC5297-6AE20-0AP3 AE1 更新/补充 BE1 操作面板补充 HE1 在线帮助 IM2 开机调试HMI Embedded IM4 安装调试 HMI Advanced TX1 编制外语文本

## D 名称

循环的输入变量/输出变量清单

名称	英语来源	相应中文
<b>AD</b>	Allowance depth	毛坯尺寸，槽到基准面的深度
<b>AFSL</b>	Angle for slot length	键槽角度
<b>ANG1, ANG2</b>	Flank angle	螺纹啮合角
<b>ANGB</b>	Liftoff angle for roughing	粗加工时退刀角度
<b>AP1</b>	Unfinished dimension in plane	粗尺寸，槽长度/槽半径
<b>AP2</b>	Unfinished dimension in plane	毛坯尺寸，槽宽度
<b>APP</b>	Approach path	导入路径
<b>APX</b>	Axial value for defining blank for facing axis	轴向数值，用于端面轴的毛坯定义
<b>APXA</b>	Absolute or incremental evaluation of parameter APX	参数 _APX 的求值，绝对或者增量
<b>APZ</b>	Axial value for defining blank for longitudinal axis	轴向数值，用于纵向轴的毛坯定义
<b>APZA</b>	Absolute or incremental evaluation of parameter APZ	参数 _APZ 的求值，绝对或者增量
<b>AS1, AS2</b>	Direction of approach/approach travel	返回运行方向和路径的规定
<b>AXN</b>	Tool axis	工具轴
<b>BNAME</b>	Name for program of drill positions	钻孔位置程序名称
<b>CDIR</b>	Circle direction	旋转方向，铣削方向
<b>CPA</b>	Center point, abscissa	圆弧孔的圆心，横坐标（绝对）
<b>CPO</b>	Center point, ordinate	孔圆弧圆心，纵坐标（绝对）
<b>CRAD</b>	Corner radius	拐角半径
<b>DAM</b>	Degression value, Path for roughing interrupt	递减系数 / 位移长度
<b>DBH</b>	Distance between holes	孔间距
<b>DIAG</b>	Groove depth	排屑槽深度
<b>DIATH</b>	Diameter of thread	额定直径，螺纹的外径
<b>DIS1</b>	Distance	可编程的前置距离
<b>DIS1</b>	Distance between columns	列间距
<b>DIS2</b>	Number of lines, Distance between rows	行间距
<b>DM1...DM4</b>	Diameter	起始点处螺纹的直径
<b>DP</b>	Depth	深度（绝对）
<b>DP1</b>	First depth	插入深度
<b>DPR</b>	Depth, relative	相对于基准面的深度

<b>DT</b>	Dwell time	粗加工时用于断屑的停留时间
<b>DTB</b>	Dwell time at bottom	在钻孔底部/排屑槽底部的停留时间
<b>DTD</b>	Dwell time at depth	在钻孔底部的停留时间
<b>DTS</b>	Dwell time at starting point	在起始点处的停留时间
<b>ENC</b>	Tapping with/without encoder	攻丝，带/不带编码器
<b>FAL</b>	Finish allowance	与轮廓相吻合的精加工余量，在键槽边/凹槽边
<b>FAL1</b>	Finish allowance on groove base	切槽底部的精加工余量
<b>FAL2</b>	Finish allowance on flanks	侧壁的精加工余量
<b>FALD</b>	Finish allowance depth	底部的精加工余量
<b>FALZ</b>	Finish allowance, z axis	纵向轴的精加工余量
<b>FALX</b>	Finish allowance, x axis	端面轴的精加工余量
<b>FDEP</b>	First depth	第一个钻孔深度（绝对）
<b>FDIS</b>	First distance	第一个钻孔到基准点的距离
<b>FDP1</b>	Overrun path in direction to plane	平面横向进给方向溢出路径
<b>FDPR</b>	First depth, relative	相对于基准面的第一个钻孔深度
<b>FF1</b>	Feedrate for roughing	粗加工进给
<b>FF2</b>	Feedrate for insertion	插入时进给
<b>FF3</b>	Feedrate for finishing	精加工进给
<b>FF4</b>	Feedrate for contour transition elements	轮廓过渡单元上的进给
<b>FFCP</b> (ab SW 6.3)	Feedrate for circular positioning	用于圆弧轨迹中间定位的进给
<b>FFD</b>	Feedrate for depth	深度方向的进给
<b>FFP1</b>	Feedrate surface	表面加工的进给
<b>FFP2</b>	Feedrate for finishing	精加工进给
<b>FFR</b>	Feedrate	进给率
<b>FORM</b>	Definition of form	成型定义
<b>FPL</b>	Final point along longitudinal axis	纵向轴终点
<b>FRF</b>	Feedrate factor	进给系数
<b>IANG</b>	Infeed angle	进给角度
<b>INDA</b>	Incremental angle	增量角度
<b>IDEP</b>	Infeed depth	进给深度
<b>KNAME</b>	Name of the contour subroutine	轮廓子程序名
<b>LSANF</b>	Block number/label identifying start of contour definition	轮廓定义开始的程序段号/标签
<b>LSEND</b>	Block number/label identifying end of contour definition	轮廓定义结束的程序段号/标签
<b>LP1</b>	Length of approach travel, radius	返回路径长度，半径
<b>LP2</b>	Length of return travel, radius	离开路径长度，半径

<b>KDIAM</b>	Internal diameter of thread	内部直径, 螺纹的内径
<b>LENG</b>	Elongated hole length, pocket length	长孔长度, 凹槽长度
<b>MDEP</b>	Minimum depth	最小孔深
<b>MID</b>	Maximum infeed depth	最大进刀深度
<b>MIDA</b>	Maximum infeed width	最大进刀宽度
<b>MIDF</b>	Maximum infeed depth for finishing	精加工最大进刀深度
<b>MPIT</b>	Thread lead as thread size	螺距, 螺纹尺寸
<b>NID</b>	Number of noncuts	空切削次数
<b>NP1 ... NP8</b>	Name/Label ...	轮廓子程序名称, 加工轮廓/标签
<b>NPP</b>	Name of part program	轮廓子程序名
<b>NRC</b>	Number of roughing cuts	粗切削次数
<b>NSP</b>	Start point offset first thread	第一个螺纹的起始角偏置
<b>NUM</b>	Number of holes	钻孔个数
<b>NUM1</b>	Number of columns	列数
<b>NUM2</b>	Number of lines	行数
<b>NUMT</b>	Number of threads	螺纹个数
<b>PA</b>	Reference point, abscissa	基准点, 纵坐标
<b>PO</b>	Reference point, ordinate	基准点, 纵坐标
<b>PO1 ... PO4</b>	Point in longitudinal axis	纵向轴螺纹的起始点/中间点/终点
<b>PIT</b>	Thread lead	螺距
<b>PNAME</b>	Name for pocket milling machining program	凹槽铣削加工程序的名称
<b>POSS</b>	Position for oriented spindle stop	主轴位置
<b>PP1 ... PP3</b>	Thread pitch 1...3 as value	螺距1...3
<b>PRAD</b>	Pocket radius	凹槽半径
<b>RAD</b>	Radius	圆半径
<b>RAD1</b>	Radius	插入时延迟线半径
<b>RCO1, RCO2</b>	Radius/chamfer outside	半径/棱边, 外部
<b>RCI1, RCI2</b>	Radius/chamfer inside	半径/棱边, 内部
<b>RFF</b>	Retract feed	退回进给
<b>RFP</b>	Reference plane	基准面 (绝对)
<b>ROP</b>	Run out path	导出路径
<b>RPA</b>	Retract path, abscissa	退回路径, 横坐标
<b>RPAP</b>	Retract path, applicate	退回路径, 应用
<b>RPO</b>	Retract path, ordinate	退回路径, 纵坐标
<b>RL</b>	Bypass contour	轮廓中心点绕行
<b>RTP</b>	Retract plane	退回平面 (绝对)
<b>SDAC</b>	Spindle direction after cyle	循环结束之后主轴方向
<b>SDIR</b>	Spindle direction	主轴方向
<b>SDIS</b>	Safety distance	安全距离

<b>SDR</b>	Spindle direction for retraction	退回时主轴方向
<b>SPCA</b>	Reference point, abscissa	直线上参考点横坐标（绝对值）
<b>SPCO</b>	Reference point, ordinate	参考点纵坐标（绝对值）
<b>SPD</b>	Starting point in the facing axis	端面轴起始点
<b>SPL</b>	Starting point along longitudinal axis	纵向轴起始点
<b>SSF</b>	Speed for finishing	精加工时速度
<b>SST</b>	Speed for tapping	攻丝时转速
<b>SST1</b>	Speed for retraction	退回时转速
<b>STA, STA1</b>	Angle	角
<b>STA2</b>	Insertion angle	摆动运动时最大插入角
<b>TDEP</b>	Thread depth	螺纹高度
<b>TN</b>	Name of stock removal tool	扩孔刀具名称
<b>TOL1</b>	Blank tolerance	毛坯公差
<b>TYPTH</b>	Typ of thread	螺纹类型
<b>VARI</b>	Working	加工方式
<b>VRT</b>	Variable return path	退刀变量/退刀位移
<b>WID</b>	(Pocket) width	凹槽宽度
<b>WIDG</b>	Groove width	排屑槽宽度





**E 索引****C**

CONTPRON 4-307

CYCLE60 ( 自软件版本 SW 6.4起 ) 3-270

CYCLE73 3-183

CYCLE74 3-183

CYCLE75 3-183

CYCLE76 3-174

CYCLE800 3-211

CYCLE801 2-102

CYCLE81 2-51

CYCLE82 2-54

CYCLE832 ( 自软件版本 SW 6.3 起 ) 3-257

CYCLE84 2-63

CYCLE840 2-70

CYCLE86 2-81

CYCLE87 2-85

CYCLE88 2-87

CYCLE89 2-89

CYCLE90 3-110

CYCLE950 4-331

**F**

FGROUP 3-110

**H**

HOLES1 2-95

HOLES2 2-99

**L**

LONGHOLE 3-117

**M**

MCALL 2-91

**P**

POCKET1 3-136

POCKET2 3-140

POCKET3 3-144

POCKET450 3-153

**S**

SETMS 3-109

SLOT1 3-122

SLOT2 3-130

SPOS 2-65, 2-66

**—**

一个圆弧上的键槽 - SLOT1 3-122

一个圆弧上的长方形孔 - LONGHOLE 3-117

**不**

不带刀具的仿真 1-26

**与**

与轮廓平行的切削 4-333

**个**

个数参数零时的性能 2-94

**主**

主轴处理 4-283

**信**

信息 1-22

**内**

内螺纹 3-112

**几**

几何参数 2-49

**剩**

剩余材料 4-334

**加**

加工参数 2-49

加工平面 1-21

## 参

参数表 1-23

## 回

回转- CYCLE800 3-211

## 在

在 JOG 运行方式下回转

(自循环版本 SW 6.5起) 3-223

## 坯

坯件更新 4-343

## 基

基准面 2-52, 3-198

## 外

外螺纹 3-111

## 奇

奇偶性检查 2-94

## 孔

孔圆弧 2-99

孔组合循环 1-19

## 安

安全距离 2-52, 3-198

## 定

定中心 2-51

## 带

带中心岛的凹槽铣削 3-183

## 平

平面定义 1-21

平面螺纹 4-321

## 循

循环仿真 1-26

循环报警 5-355

循环报警概述 5-355

循环概述 1-18

循环设定数据, 车削 4-284

循环设定数据, 铣削 3-109

循环调用 1-23

循环辅助子程序 1-20

## 成

成排孔 2-95

## 扩

扩展的切削循环 - CYCLE950 4-331

## 攻

攻丝, 不带补偿衬套 2-63

攻丝, 带补偿夹具 2-70

攻丝带补偿衬套, 不带编码器 2-71

攻丝带补偿衬套, 带有编码器 2-72

## 旋

旋转循环 1-20

## 模

模态调用钻削循环 2-91

模腔循环 - CYCLE60

(自软件版本 SW 6.4起) 3-270

## 毛

毛坯 4-334

## 深

深孔钻削, 带断屑 2-58

深孔钻削, 带退刀排屑 2-57

## 点

点栅格 2-102

## 环

环形槽 - SLOT2 3-130

**用**

用于循环的定义文件 1-41

用户循环的循环支持) 1-35

**相**

相对钻削深度 2-52, 3-119, 3-124, 3-138, 3-160,  
3-198

**程**

程序编辑器中的循环支持 1-27

**纵**

纵向螺纹 4-321

**绝**

绝对钻削深度 2-52, 3-119, 3-124, 3-138, 3-160,  
3-198

**自**

自由切削角度 4-284

自由的轮廓编程 1-30

**螺**

螺纹精整 4-329

螺纹铣削 3-110

**调**

调用 1-21, 2-50

调用条件 1-21

**起**

起始点 4-309

**轮**

轮廓 1-直线 1-30

轮廓 2-直线 1-30

轮廓 3-直线 1-30

轮廓定义 4-306, 4-341

轮廓监控 4-284, 4-309

轮廓编程 4-336

轮廓编程 1-30

**轴**

轴分配 1-21

**返**

返回条件 1-21

**退**

退回平面 2-52, 3-198

**钻**

钻削 2-51

钻削, 镗平面 2-54

钻削图循环 2-94

钻削图循环, 不带钻削循环调用 2-94

钻削循环 1-19

**铣**

铣削循环 1-19

铣削环形凹槽 - POCKET2 3-140

铣削矩形凹槽- POCKET3 3-144

铣削矩形槽 - POCKET1 3-136

铣削矩形轴颈- CYCLE76 3-174

**镗**

镗孔 2-49

镗孔2 2-81

镗孔3 2-85

镗孔4 2-87

镗孔5 2-89

镗孔循环 2-49

**高**

高速设定- CYCLE832

(自软件版本 SW 6.3起) 3-257

**F 指令/名称****C**

CYCLE60 (ab SW 6.4) 3-270  
CYCLE71 3-158  
CYCLE72 3-164  
CYCLE73 3-189  
CYCLE74 3-184  
CYCLE75 3-186  
CYCLE76 3-174  
CYCLE77 3-179  
CYCLE800 3-211  
CYCLE801 2-102  
CYCLE81 2-51  
CYCLE82 2-54  
CYCLE83 2-56  
CYCLE832 (ab SW 6.3) 3-257  
CYCLE84 2-63  
CYCLE840 2-70  
CYCLE85 2-78  
CYCLE86 2-81  
CYCLE87 2-85  
CYCLE88 2-87  
CYCLE89 2-89  
CYCLE90 3-110  
CYCLE93 4-286  
CYCLE94 4-295  
CYCLE95 4-299  
CYCLE950 4-331  
CYCLE96 4-312  
CYCLE97 4-316  
CYCLE98 4-323

**H**

HOLES1 2-95  
HOLES2 2-99

**L**

LONGHOLE 3-117

**P**

POCKET1 3-136  
POCKET2 3-140  
POCKET3 3-144  
POCKET4 3-153

**S**

SLOT1 3-122  
SLOT2 3-130

寄：  
 SIEMENS AG  
 A&D MC BMS  
 Postfach 3180  
 D-91050 Erlangen  
 电话 0180 / 5050 – 222 [热线]  
 传真 09131 / 98 – 2176 [文献资料]  
 电子邮件  
 motioncontrol.docu@erlf.siemens.de

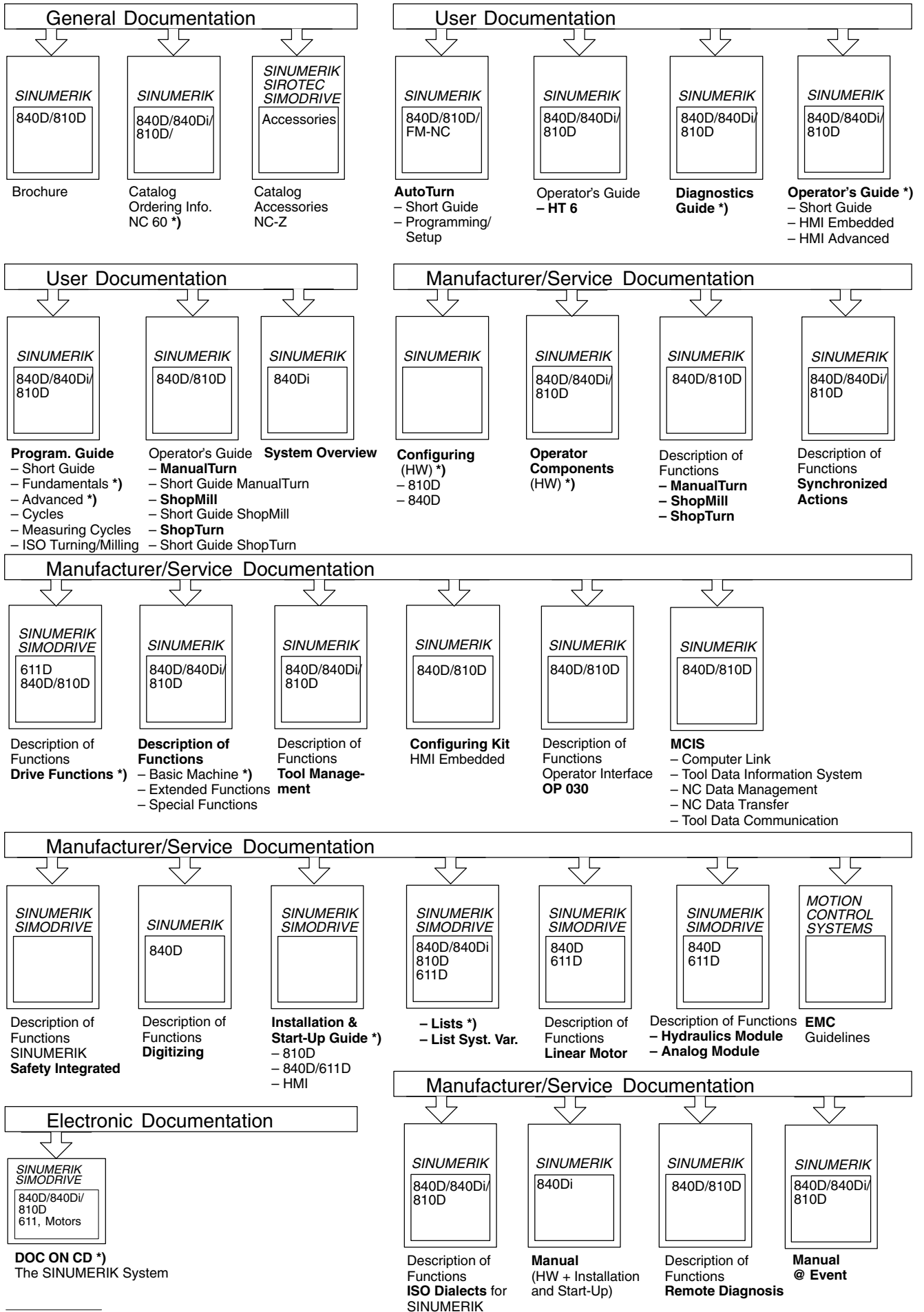
建议  
 更正  
 出版/手册：  
 SINUMERIK  
 840D/840Di/810D  
 循环  
 用户文献

<p>此信来自</p> <p>名称</p> <hr/> <p>公司地址</p> <p>街道：</p> <hr/> <p>邮编：                      地区：</p> <hr/> <p>电话：                      /</p> <hr/> <p>传真：                      /</p>	<p>编程说明</p> <p>订货号： 6FC5298-7AB40-0RP1</p> <p>版本 10.04</p> <hr/> <p>如果您在阅读文献资料时出现打印错误，请以表格形式告知我们。同样非常感谢您的鞭策和建议。</p>
--	---

建议和/或更正：



# Overview of SINUMERIK 840D/840Di/810D Documentation (03.2004)



\*) These documents are a minimum requirement

**Siemens AG**  
Automation & Drives  
Motion Control Systems  
Postfach 3180, D – 91050 Erlangen  
Germany

[www.siemens.com](http://www.siemens.com)

© Siemens AG 2004  
保留技术变更权利。  
订货号: 6FC5298-7AB40-0RP1

Printed in Germany