

SIEMENS

编程原理

1

循环

2

典型车削程序

3

SINUMERIK

SINUMERIK 808D

车削 第二部分：编程 (西门子编程指令)

编程和操作手册

适用于：
SINUMERIK 808D 车削（软件版本：V4.4.2）

目标使用人群：
最终用户和服务工程师




12/2012

6FC5398-5DP10-0RA0

法律资讯

警告提示系统

为了您的人身安全以及避免财产损失，必须注意本手册中的提示。人身安全的提示用一个警告三角表示，仅与财产损失有关的提示不带警告三角。警告提示根据危险等级由高到低如下表示。

 危险
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。
 警告
表示如果不采取相应的小心措施， 可能 导致死亡或者严重的人身伤害。
 小心
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。
注意
表示如果不采取相应的小心措施，可能导致财产损失。


当出现多个危险等级的情况下，每次总是使用最高等级的警告提示。如果在某个警告提示中带有警告可能导致人身伤害的警告三角，则可能在该警告提示中另外还附带有可能导致财产损失的警告。

合格的专业人员

本文件所属的产品/系统只允许由符合各项工作要求的**合格人员**进行操作。其操作必须遵照各自自带的文件说明，特别是其中的安全及警告提示。由于具备相关培训及经验，合格人员可以察觉本产品/系统的风险，并避免可能的危险。

按规定使用 Siemens 产品

请注意下列说明：

 警告
Siemens 产品只允许用于目录和相关技术文件中规定的使用情况。如果要使用其他公司的产品和组件，必须得到 Siemens 推荐和允许。正确的运输、储存、组装、装配、安装、调试、操作和维护是产品安全、正常运行的前提。必须保证允许的环境条件。必须注意相关文件中的提示。

商标

所有带有标记符号®的都是西门子股份有限公司的注册商标。本印刷品中的其他符号可能是一些其他商标。若第三方出于自身目的使用这些商标，将侵害其所有者的权利。

责任免除

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。印刷品中的数据都按规定经过检测，必要的修正值包含在下一版本中。

目录

1	编程原理	7
1.1	编程基础知识	7
1.1.1	程序名称	7
1.1.2	程序结构	7
1.1.3	字结构和地址	8
1.1.4	字符集	9
1.1.5	程序段结构	10
1.1.6	指令表	12
1.2	位移说明	29
1.2.1	尺寸编程	29
1.2.2	绝对/增量尺寸: G90, G91, AC, IC	30
1.2.3	公制尺寸和英制尺寸: G71, G70, G710, G700	32
1.2.4	半径 - 直径尺寸: DIAMOF, DIAMON, DIAM90	33
1.2.5	可编程的零点偏移: TRANS, ATRANS	34
1.2.6	可编程的比例系数: SCALE, ASCALE	38
1.2.7	夹紧工件 - 可设定的零点偏移: G54 到 G59, G500, G53, G153	40
1.3	轴运行	42
1.3.1	快速移动直线插补: G0G0	42
1.3.2	进给率 F	43
1.3.3	带进给率的直线插补 G1	44
1.3.4	圆弧插补: G2,G3	45
1.3.5	通过中间点进行圆弧插补: CIP	49
1.3.6	切线过渡圆弧 CT	50
1.3.7	有恒定螺距的螺纹切削: G33	50
1.3.8	G33 的可编程导入和导出路程: DITS, DITE	54
1.3.9	变量螺距的螺纹切削: G34,G35	56
1.3.10	螺纹插补: G331,G332	57
1.3.11	返回固定点 G75	58
1.3.12	回参考点运行 G74	60
1.3.13	准停/连续路径运行 G9, G60, G64	60
1.3.14	加速度性能: BRISK, SOFT	64
1.3.15	第三轴	65
1.3.16	暂停时间: G4	66
1.4	主轴运动	67
1.4.1	主轴转速 S, 旋转方向	67
1.4.2	定位主轴	68
1.4.2.1	定位主轴 (SPOS, SPOSA, M19, M70, WAITS)	68
1.4.2.2	定位主轴 (SPOS, SPOSA, M19, M70, WAITS): 其它信息	75

1.4.3	齿轮级	76
1.5	特殊车削功能.....	77
1.5.1	恒定切削速度: G96,G97	77
1.5.2	倒圆、倒角	79
1.5.3	轮廓编程.....	83
1.6	刀具和刀具补偿	85
1.6.1	通用说明 (车削)	85
1.6.2	刀具 T (车削)	86
1.6.3	刀具补偿号 D (车削)	87
1.6.4	选择刀具半径补偿: G41,G42	90
1.6.5	拐角特性: G450,G451.....	92
1.6.6	取消刀具半径补偿: G40.....	93
1.6.7	刀具半径补偿的特殊情况.....	94
1.6.8	刀具半径补偿举例 (车床)	96
1.6.9	刀具补偿的特殊情况 (车床)	97
1.7	辅助功能 M.....	98
1.8	H 功能	99
1.9	计算参数 R, LUD 和 PLC 变量.....	99
1.9.1	计算参数 R.....	99
1.9.2	局部用户数据 (LUD)	102
1.9.3	PLC 变量的读和写.....	103
1.10	程序跳转.....	104
1.10.1	绝对程序跳转.....	104
1.10.2	有条件程序跳转	105
1.10.3	程序跳转举例.....	107
1.10.4	程序跳转的跳转目标.....	108
1.11	子程序	109
1.11.1	概述	109
1.11.2	调用加工循环 (车床)	112
1.11.3	执行外部子程序 (EXTCALL)	112
1.12	定时器和工件计数器.....	114
1.12.1	运行时间定时器.....	114
1.12.2	工件计数器	117
2	循环.....	119
2.1	循环概述.....	119
2.2	循环编程.....	120
2.3	程序编辑器中的图形循环支持.....	122
2.4	钻削循环.....	123
2.4.1	概述	123

2.4.2	前提条件.....	124
2.4.3	钻削, 定中心 - CYCLE81.....	127
2.4.4	钻削, 铤平面 - CYCLE82.....	130
2.4.5	深孔钻削 - CYCLE83.....	133
2.4.6	攻丝, 不带补偿衬套 - CYCLE84.....	139
2.4.7	攻丝, 带补偿衬套 - CYCLE840.....	146
2.4.8	铰孔 1 - CYCLE85.....	152
2.4.9	镗孔 - CYCLE86.....	155
2.4.10	带停止 1 的镗孔 - CYCLE87.....	158
2.4.11	带停止 2 的钻孔 - CYCLE88.....	160
2.4.12	铰孔 2 - CYCLE89.....	162
2.5	车削循环.....	164
2.5.1	前提条件.....	164
2.5.2	切断 - CYCLE92.....	166
2.5.3	切槽 - CYCLE93.....	169
2.5.4	退刀槽 (形状 E 和 F, 符合 DIN) - CYCLE94.....	179
2.5.5	毛坯切削, 带底切 - CYCLE95.....	184
2.5.6	螺纹退刀槽 - CYCLE96.....	202
2.5.7	螺纹链 - CYCLE98.....	206
2.5.8	螺纹切削 - CYCLE99.....	213
2.6	故障信息和故障处理.....	223
2.6.1	一般说明.....	223
2.6.2	循环中的故障处理.....	223
2.6.3	循环报警一览.....	223
2.6.4	循环中的显示消息.....	224
3	典型车削程序.....	225
	索引.....	235

编程原理

1.1 编程基础知识

1.1.1 程序名称

每个程序必须有程序名称。程序名称必须遵守以下规定：

- 程序名最多使用 **24** 个英文字母或 **12** 个中文字符（字符长度不包括文件扩展名）
- 仅使用小数点来隔开文件扩展名
- 如需创建子程序而当前默认程序类型为 **MPF**（主程序），必须输入文件扩展名**".SPF"**
- 如需创建主程序而当前默认程序类型为 **SPF**（子程序），必须输入文件扩展名**".MPF"**
- 如采用当前默认程序类型，则无需输入文件扩展名
- 应避免使用特殊字符作为程序名。

示例

WERKSTUECK527

1.1.2 程序结构

结构和内容

数控系统程序由一系列的**程序段**组成（参见下表）。每个程序块代表一个加工步骤。以**字**的形式将指令写入程序块。执行顺序中的最后一个程序段包含程序结束的一个特殊字，例如，**M2**。

下表显示数控系统程序结构的示例。

程序段	字	字	字	...	； 注释
程序段	N10	G0	X20	...	； 第一个程序段
程序段	N20	G2	Z37	...	； 第二个程序段
程序段	N30	G91	； ...
程序段	N40	
程序段	N50	M2			； 程序结束

1.1.3 字结构和地址

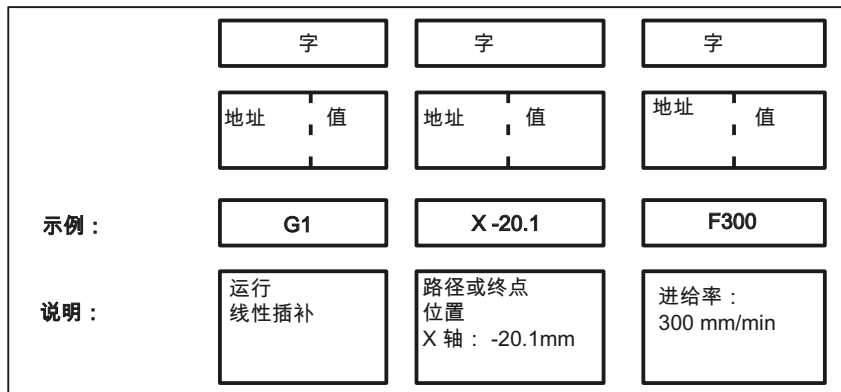
功能/结构

一个字是一个程序段元素，主要构成一个控制命令。字由以下两个部分组成：

- **地址符：** 通常为一个字母
- **数值：** 数字串，对于某些地址而言，该数字串前面可带有正负号，该数字串亦可带有小数点。

正号 (+) 可以不用写。

下图显示字结构示例。



多个地址符

一个字可以包含多个地址符。但是，在这种情况下，必须通过中间字符“=”分配数值。

例如：**CR=5.23**

另外，还可以使用符号名称调用 G 功能(更多的信息参见章节“指令表 (页 12)”)。

例如：**SCALE**；启用缩放系数

扩展地址

对于如下地址，地址可以通过 1 到 4 个数字进行地址扩展，从而可以产生较多的地址数量。在这种情况下，必须使用等号“=”分配值。

R	计算参数
H	H 功能
I, J, K	插补参数/中间点
M	特殊功能 M，借助其它选项而影响主轴。
S	主轴转速

示例：**R10=6.234 H5=12.1 I1=32.67 M2=5 S1=400**

1.1.4 字符集

下列字符用于编程。按照相关定义对它们进行说明。

字母，数字

A、B、C、D、E、F、G、H、I、J、K、L、M、N、O、P、Q、R、S、T、U、V、W、X、Y 和 Z

0、1、2、3、4、5、6、7、8 和 9

字母不区分大小写。

可打印的特殊字符

(左圆括号	"	引号
)	右圆括号	_	下划线（属于字母）
[左方括号	.	小数点
]	右方括号	,	逗号，分隔符
<	小于	;	注释引导
>	大于	%	保留；未占用
:	主程序段，标签结束	&	保留；未占用
=	分配；相等部分	'	保留；未占用
/	跳转	\$	系统变量标识符
*	乘法	?	保留；未占用
+	加法和加号	!	保留；未占用
-	减法，减号		

非可打印特殊字符

LF	程序段结束字符
空格	字之间的分隔符；空格
制表符	保留；未占用

1.1.5 程序段结构

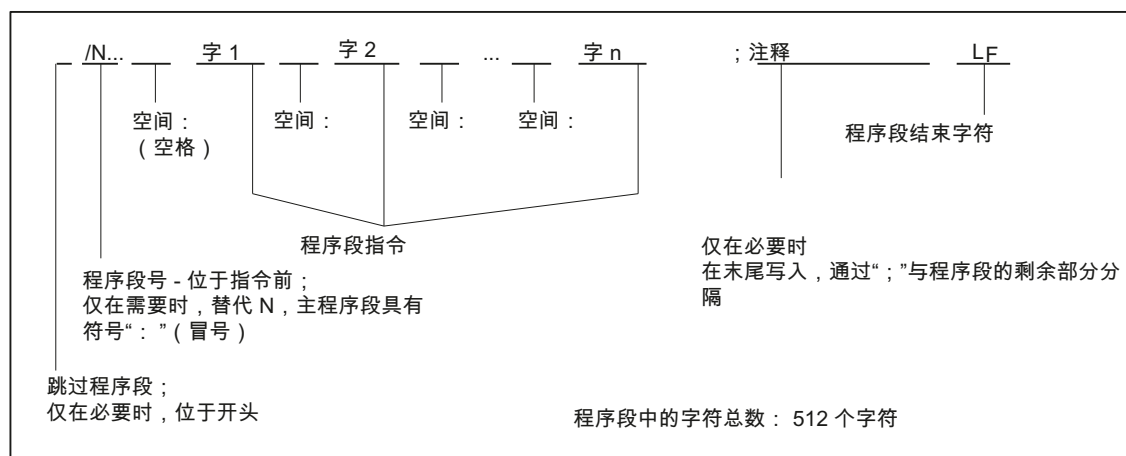
功能

程序段应包含执行加工步骤需要的所有数据。

通常，一个块由多个字组成，始终带有**程序段结束字符“LF”**（换行）。写入程序段时，要自动生成该字符，可按下外接键盘上的换行键或 PPU 上的键：



参见下面的程序段结构示意图：



字序

如果一个程序段中有多个指令，建议使用以下顺序：

N... G... X... Z... F... S... T... D... M... H...

程序段号方面的说明

首先在步骤 5 或 10 中选择程序段号。这样在以后插入程序段时仍能保持程序段号升序排列。

跳过程序段

可以通过程序段号前面的斜杠 /**标记**每个程序运行时不执行的程序段。

通过**操作**（程序控制：“SKP”）或提供给可编程控制器（信号）激活程序段跳过。如果连续多个程序段前都以“/”标记，则它们都将被跳过。

如果在程序执行过程中必须跳过程序段，不执行标记“/”的所有程序段。不考虑相关程序段中包含的所有指令。程序从下一个程序段（不带标记）开始继续执行。

注释, 备注

可以使用注释（备注）解释程序段中的指令。注释以符号“;”开始，以程序段末尾结束。在当前程序段显示中，注释与剩余程序段的内容一起显示。

消息

在单独程序段中编程消息。在特殊字段中显示消息，并且保持活动状态，直到执行带有新消息的程序段或直到达到程序结束为止。在消息文本中最多可以显示 **65** 个字符。不带消息文本的消息取消上一条消息。

MSG (“这是消息文本”)

编程示例

```
N10 ; G&S 公司, 订货号 12A71
N20 ; 泵零件 17, 图纸编号: 123 677 123 677
N30 ; 程序创建人: H. Adam, TV 4 部门
N40 MSG("DRAWING NO.: 123677")
:50 G54 F4.7 S220 D2 M3 ; 主程序段
N60 G0 G90 X100 Z200
N70 G1 Z185.6
N80 X112
/N90 X118 Z180 ; 此程序段可跳过
N100 X118 Z120
N110 G0 G90 X200
N120 M2 ; 程序结束
```

1.1.6 指令表

带 * 标记的功能在程序启动时生效（工艺“车削”控制系统的供货状态：如果没有另外编程并且机床制造商保持了标准设置）。

地址	含义	赋值	说明	编程
D	刀具补偿号	0 ... 9, 仅整数, 没有符号	包含特别刀具 T... 的补偿数据; D0 表示刀具无补偿, 一把刀具最多可激活 1~9 共九个数字, 即一把刀具最多可以同时携带 9 个不同的补偿数据。	D...
F	进给率	0.001 ... 99 999.999	刀具/工件的轨迹速度; 单位: 毫米/分钟或毫米/转 取决于 G94 或 G95	F...
F	停留时间 (带有 G4 的程序段)	0.001 ... 99 999.999	停留时间, 单位秒	G4 F...; 单独程序段
F	螺距变化 (包含 G34、G35 的程序段)	0.001 ... 99 999.999	单位: 毫米/转 ²	参见 G34、G35
M	G 功能 (准备功能)	仅整数, 指定值	G 功能分为 G 功能组。在一个程序段中可以最多编程一组一个 G 功能。 G 功能可以是模态 (直到通过同一组另一个功能取消) 或仅对其所在的编程程序段 (非模态) 有效。	G... 或符号名称, 例如: CIP

地址	含义	赋值	说明	编程
G 功能组:				
G0	快速移动直线插补		1: 运行指令（插补类型）， 模态有效	G0 X... Z...
G1 *	进给率的直线插补			G1 X...Z... F...
G2	顺时针圆弧插补			G2 X... Z... I... K... F...; 圆心和终点 G2 X... Z... CR=... F...; 半径和终点 G2 AR=... I... K... F...; 张角和圆心 G2 AR=... X... Z... F...; 张角和终点
				G3 ...; 其他, 同 G2
				CIP X... Z... I1=... K1=... F...; I1、K1 为 中间点
CT	圆弧插补; 切线过渡			N10 ... N20 CT Z... X... F...; 圆弧, 正切过渡到前 路段 N10
G33	恒螺距螺纹插补			; 恒定螺距 G33 Z... K... SF=...; 圆柱螺纹 G33 X... I... SF=...; 横向螺纹 G33 Z... X... K... SF=...; 圆锥螺纹, Z 轴上的轨迹大于 X 轴 上的轨迹 G33 Z... X... I... SF=...; 圆锥螺纹, X 轴上的轨迹大于 Z 轴 上的轨迹

地址	含义	赋值	说明	编程
G34	螺纹切削，螺距增加			G33 Z... K... SF=...; 圆柱螺纹，恒定螺距 G34 Z... K... F17.123; 螺距增加; 17.123 毫米/转 ²
G35	螺纹切削，螺距减少			G33 Z... K... SF=...; 圆柱螺纹 G35 Z... K... F7.321; 螺距减少; 7.321 毫米 /转 ²
G331	螺纹插补			N10 SPOS=...; 位置 控制中的主轴 N20 G331 Z... K... S...; 不带补偿夹具的 攻丝，例如，在 Z 轴 上; 通过螺距的符号 定义右侧或左侧螺纹 (例如, K+): + : 和 M3 一样 - : 和 M4 一样
G332	螺纹插补 - 退回			G332 Z... K...; 不带补 偿夹具的攻丝，例 如，在 Z 轴上，退回 运行 ; 螺距的符号和 G331 相同
G4	暂停时间			2: 特殊运动，停留时间，非 模态
G74	回参考点			G74 X1=0 Z1=0; 单 独程序段, (机床轴标识符!)

地址	含义	赋值	说明	编程
G75	接近固定点			G75 X1=0 Z1=0; 单独程序段, (机床轴标识符!)
TRANS	偏移, 可编程		3: 写入存储器, 非模态	TRANS X... Z...; 单独程序段
SCALE	可编程的缩放系数			SCALE X... Z...; 指定轴方向上的缩放系数, 单独程序段
ROT	旋转, 可编程			ROT RPL=...; 在当前平面 G17 至 G19 上旋转, 单独程序段
MIRROR	可编程镜像			MIRROR X0; 对坐标轴镜像, 单独程序段
ATRANS	补充偏移, 编程			ATRANS X... Z...; 单独程序段
ASCALE	补充可编程的缩放系数			ASCALE X... Z...; 指定轴方向上的缩放系数, 单独程序段
AROT	补充可编程旋转			AROT RPL=...; 在当前平面 G17 至 G19 上旋转, 单独程序段
AMIRROR	补充可编程镜像			AMIRROR X0; 对坐标轴镜像, 单独程序段
G17	X/Y 平面 (当需要中心钻削时)			6: 平面选择
G18 *	Z/X 平面 (标准车削)			
G19	Y/Z 平面			
G40 *	取消刀具半径补偿		7: 刀具半径补偿, 模态有效	
G41	刀具半径补偿; 沿刀具运动方向, 始终在轮廓左侧			
G42	刀具半径补偿; 沿刀具运动方向, 始终在轮廓右侧			
G500 *	取消可设定的零点偏移		8: 可设定的零点偏移, 模态	

1.1 编程基础知识

地址	含义	赋值	说明	编程	
G54	1. 可设定的零点偏移		有效		
G55	2. 可设定的零点偏移				
G56	3. 可设定的零点偏移				
G57	4. 可设定的零点偏移				
G58	5. 可设定的零点偏移				
G59	6. 可设定的零点偏移				
G53	取消可设定的零点偏移，非模态		9: 抑制可设定零点偏移，非模态		
G153	取消可设定的零点偏移，非模态，此外取消基本框架				
G60 *	准停		10: 逼近行为，情态有效		
G64	连续路径运行				
G62	激活刀具半径补偿时的内拐角减速 (G41, G42)		仅连同连续路径模式	G62 Z... G1	
G9	非模态准停		11: 非模态准停，非模态		
G601 *	精准停窗口，带有 G60、G9		12: 精准停窗口，情态有效		
G602	粗准停窗口，带有 G60、G9				
G621	所有角处拐角延迟		仅连同连续路径模式	G621 AIDS=...	
G70	英制尺寸输入		13: 英制/公制尺寸数据，模态有效		
G71 *	公制尺寸数据输入				
G700	英制尺寸数据输入；也用于进给率 F				
G710	公制尺寸数据输入；也用于进给率 F				
G90 *	绝对尺寸数据输入		14: 绝对/增量尺寸，情态有效		
G91	增量尺寸输入				
G94	进给率 F 以毫米/分钟为单位		15: 进给率/主轴，模态有效		
G95 *	主轴旋转进给率 F，单位：毫米/转				
G96	恒定切削速度“开” (F 以毫米/转为单位，S 以米/分钟为单位)				G96 S... LIMS=... F...
G97	恒定切削速度“关”				
G450 *	过渡圆弧		18: 与刀具半径补偿一起工		

地址	含义	赋值	说明	编程
G451	交点		作时在拐角处发挥作用，模态有效	
BRISK *	轨迹跳跃加速		21: 加速度属性，情态有效	
SOFT	轨迹平滑加速			
FFWOF *	前馈控制 OFF		24: 前馈控制，情态有效	
FFWON	前馈控制 ON			
DIAMOF	半径尺寸		29: 半径/直径尺寸，情态有效	
DIAMON *	直径尺寸			
G290 *	西门子模式		47: 外部 NC 语言，模态有效	
G291	外部模式			
H H0= to H9999=	H 功能	$\pm 0.0000001 \dots$ 9999 9999 (8 位小数) 或 指定为指数: $\pm (10-300 \dots$ 10+300)	传输到 PLC 的值; 含义由机床制造商定义	H0=... H9999=... 例如: H7=23.456
I	插补参数	$\pm 0.001 \dots 99$ 999.999 螺纹: 0.001 ... 2000.000	属于 X 轴; 含义取决于 G2,G3 ->圆心或 G33、 G34、G35 G331、G332 -> 螺距	参见 G2、G3 和 G33、G34、G35
K	插补参数	$\pm 0.001 \dots 99$ 999.999 螺纹: 0.001 ... 2000.000	属于 Z 轴; 否则, 同 I	参见 G2、G3 和 G33、G34、G35
I1=	圆弧插补的中间点	$\pm 0.001 \dots 99$ 999.999	用 CIP 进行圆弧插补的 X 轴 坐标	参见 CIP
K1=	圆弧插补的中间点	$\pm 0.001 \dots 99$ 999.999	属于 Z 轴; 指定 CIP 圆弧插 补	参见 CIP

地址	含义	赋值	说明	编程
L	子程序；名称和调用	7 个小数； 仅整数，没有符号	替代自由名称，还可以选择 L1 ...L99999999； 这还在单独程序段中调用子程序 (UP)， 请注意：L0001 不是始终等于 L1。 为换刀具子程序保留名称“LL6”。	L....；单独程序段
M	附加功能	0 ... 99 仅整数，没有符号	例如，用于启动开关操作，如“冷却液开”，每个程序段最多 5 个 M 功能。	M...
M0	编程停止		在包含 M0 的程序段结束时停止加工；按下 NC START 可继续。	
M1	可选停止		和 M0 一样，但是，只有在存在特殊信号（程序控制：“M01”）时才能执行停止。	
M2	返回到程序开始处时结束主程序		可以在处理顺序的最后一个程序段中找到	
M30	程序结束（与 M2 相同）		可以在处理顺序的最后一个程序段中找到	
M17	结束子程序		可以在处理顺序的最后一个程序段中找到	
M3	主轴顺时针旋转			
M4	主轴逆时针旋转			
M5	主轴停止			
Mn=3	主轴顺时针旋转		n=1	M1=3；主轴 1 顺时针旋转停止
Mn=4	主轴逆时针旋转（主轴 n）		n=1	M1=4；主轴 1 逆时针旋转停止
Mn=5	主轴停止（主轴 n）		n=1	M1=5；主轴 1 的主轴停止

地址	含义	赋值	说明	编程
M6	换刀		只有在通过机床控制面板使用 M6 激活时；否则，使用 T 命令直接更改	
M40	自动齿轮级切换			
Mn=40	自动齿轮级切换（主轴 n）	n=1		M1=40；自动齿轮级；主轴 1
M41 到 M45	齿轮级 1 到齿轮级 5			
Mn=41 到 Mn=45	齿轮级 1 到齿轮级 5（主轴 n）	n=1		M1=41；主轴 1 的第一齿轮级
M70, M19	-		保留；未占用	
M...	其它的 M 功能		功能不通过控制系统定义，因此可以被机床制造商自由使用	
N	程序段号 - 子程序段	0 ... 9999 9999 仅整数，没有符号	可以用于使用编号标识程序段；在程序段开头处写入。	N20
:	主程序段的程序段号	0 ... 9999 9999 仅整数，没有符号	特殊程序段标识，替代 N...；此类程序段应包含整个后续加工步骤的所有指令。	:20
P	零件程序运行次数	1 ... 9999 仅整数，没有符号	在子程序多次运行并且包含在与调用相同的程序段中时使用	L781 P...；单独程序段 N10 L871 P3；三个循环
R0 到 R299	计算参数	$\pm 0.0000001 \dots$ 9999 9999 (8 位小数) 或 指定为指数： $\pm (10-300 \dots$ 10+300)		R1=7.9431 R2=4 指定指数： R1=-1.9876EX9； R1=-1 987 600 000
	计算功能		除了使用操作数 + - * / 的 4 个基本计算功能之外，还有以下计算功能：	
SIN()	正弦	度		R1=SIN(17.35)

1.1 编程基础知识

地址	含义	赋值	说明	编程
COS()	余弦	度		R2=COS(R3)
TAN()	正切	度		R4=TAN(R5)
ASIN()	反正弦			R10=ASIN(0.35) ; R10: 20.487 度
ACOS()	反余弦			R20=ACOS(R2) ; R20: ... 度
ATAN2(,)	反正切 2		矢量和的角度由 2 个互相垂直的矢量得出。指定的第二个矢量始终用于角度参考。 结果范围: -180 至 +180 度	R40=ATAN2(30.5,80.1) ; R40: 20.8455 度
SQRT()	平方根			R6=SQRT(R7)
POT()	平方			R12=POT(R13)
ABS()	绝对值			R8=ABS(R9)
TRUNC()	整数取整			R10=TRUNC(R2)
LN()	自然对数			R12=LN(R9)
EXP()	指数函数			R13=EXP(R1)
RET	子例程结束		替代 M2 用于保持连续路径模式	RET; 单独程序段
S...	主轴转速	0.001 ... 99 999.999	主轴转速的测量单位	S...
S1=...	主轴 1 的转速	0.001 ... 99 999.999	主轴转速的测量单位	S1=725; 主轴 1 的转速 725 r.p.m.
S	G96 处于活动状态时的切削速度	0.001 ... 99 999.999	G96 切削速度计量单位为米/分; 仅用于主轴	G96 S...
S	停留时间 (带有 G4 的程序段)	0.001 ... 99 999.999	主轴暂停转数	G4 S...; 单独程序段
T	刀具号	1 ... 32 000 仅整数, 没有符号	可以直接使用 T 命令或只使用 M6 执行换刀。可以在机床数据中设置。	T...

地址	含义	赋值	说明	编程
X	轴	$\pm 0.001 \dots 99$ 999.999	位移说明	X...
Y	轴	$\pm 0.001 \dots 99$ 999.999	位移说明	Y...
Z	轴	$\pm 0.001 \dots 99$ 999.999	位移说明	Z...
AC	绝对坐标	-	可以为某些轴的终点和中心点指定尺寸，不考虑 G91。	N10 G91 X10 Z=AC(20); X - 增量尺寸, Z - 绝对尺寸
ACC[轴]	加速度倍率	1 ... 200, 整数	轴或主轴的加速度倍率; 以百分比的形式指定	N10 ACC[X]=80; 对于 X 轴, 80% N20 ACC[S]=50; 对于主轴: 50%
ACP	绝对坐标; 从正方向运行至某位置 (回转轴、主轴)	-	还可以使用 ACP(...) 指定回转轴的终点尺寸, 不考虑 G90/G91; 还应用于主轴定位	N10 A=ACP(45.3); 正向接近 A 轴的绝对位置 N20 SPOS=ACP(33.1); 定位主轴
ACN	绝对坐标; 从负方向运行至某位置 (回转轴、主轴)	-	还可以使用 ACN(...) 指定回转轴的终点尺寸, 不考虑 G90/G91; 还应用于主轴定位	N10 A=ACN(45.3); 负向接近 A 轴的绝对位置 N20 SPOS=ACN(33.1); 定位主轴
ANG	轮廓段中的直线角度	$\pm 0.00001 \dots$ 359.99999	以度为单位指定; 如果只知道平面的一个终点坐标 或 当平面中终点坐标已知或者多个程序段编程轮廓而最后的终点坐标未知时, 在 G0 或者 G1 下定义直线的一种方法	N10 G1 X... Z.... N11 X... ANG=... 或几个程序段上的轮廓: N10 G1 X... Z... N11 ANG=... N12 X... Z... ANG=...

地址	含义	赋值	说明	编程
AR	圆弧插补的张角	0.00001 ... 359.99999	以度为单位指定；可以在使用 G2/G3 时定义圆弧	参见 G2、G3
CALL	间接循环调用	-	循环调用的特殊形式；没有传输参数；循环的名称存储在变量中； 仅用于循环内部使用	N10 CALL VARNAME；变量名称
CHF	倒角； 通用	0.001 ... 99 999.999	在两个轮廓程序段之间插入指定倒角长度的倒角	N10 X... Z... CHF=... N11 X... Z...
CHR	倒角； 在轮廓定义中	0.001 ... 99 999.999	在两个轮廓间插入给定腰长的倒角	N10 X... Z... CHR=... N11 X... Z...
CR	圆弧插补半径	0.010 ... 99 999.999 负号 - 用于选择 圆弧：大于半 圆	可以在使用 G2/G3 时定义圆弧	参见 G2、G3
CYCLE...	加工循环	仅指定值	调用加工循环需要单独程序段；必须用值加载合适的传输参数。 也可以使用附加 MCALL 或 CALL 调用特殊循环。	
CYCLE81	钻削，定中心			N5 RTP=110 RFP=100；赋值 N10 CYCLE81(RTP, RFP, ...); 单独零件程序段
CYCLE82	钻削，镗平面			N5 RTP=110 RFP=100；赋值 N10 CYCLE82(RTP, RFP, ...); 单独程序段
CYCLE83	深孔钻削			N10 CYCLE83(110, 100, ...); 或直接传输 值，单独程序段
CYCLE84	刚性攻丝			N10 CYCLE84(...); 单独程序段

地址	含义	赋值	说明	编程
CYCLE840	带补偿攻丝			N10 CYCLE840(...); 单独程序段
CYCLE85	铰孔 1			N10 CYCLE85(...); 单独程序段
CYCLE86	镗孔			N10 CYCLE86(...); 单独程序段
CYCLE87	带停止 1 的钻孔			N10 CYCLE87(...); 单独零件程序段
CYCLE88	带停止 2 的钻孔			N10 CYCLE88(...); 单独程序段
CYCLE89	铰孔 2			N10 CYCLE89(...); 单独零件程序段
CYCLE92	切割			N10 CYCLE92(...); 单独程序段
CYCLE93	切槽			N10 CYCLE93(...); 单独程序段
CYCLE94	退刀槽 DIN76 (E 型和 F 型), 精加工			N10 CYCLE94(...); 单独程序段
CYCLE95	毛坯切削, 带底切			N10 CYCLE95(...); 单独程序段
CYCLE96	螺纹退刀槽			N10 CYCLE96(...); 单独零件程序段
CYCLE98	螺纹排列			N10 CYCLE98(...); 单独零件程序段
CYCLE99	螺纹切削			N10 CYCLE99(...); 单独程序段
DC	绝对坐标; 直接接近位置 (回转轴、主轴)	-	还可以使用 DC(...) 指定回转轴的终点尺寸, 不考虑 G90/G91; 还应用于主轴定位	N10 A=DC(45.3); 直接接近 A 轴的绝对位置 N20 SPOS=DC(33.1); 定位主轴

地址	含义	赋值	说明	编程
DEF	定义指令		在程序开头处直接定义 BOOL、CHAR、INT、 REAL 型的本地用户变量	DEF INT VARI1=24, VARI2; 2 个 INT 型变 量; 用户定义的名称
DITS	螺纹 G33 的导入行程	-1 ... < 0, 0, > 0	以设计的轴加速度启动; 以阶跃形加速度启动; 导入路径值, 可能带有倒圆	N10 G33 Z50 K5 DITS=4
DITE	螺纹 G33 的导出行程	-1 ... < 0, 0, > 0	使用配置的轴加速度制动; 使用突然加速度制动; 指定导出行程, 带有倒圆	N10 G33 Z50 K5 DITE=4
FRC	用于倒角/倒圆的非 模态进给率	0, >0	FRC=0 时, 进给率 F 将起 作用	对于单位, 参见 F 和 G94、G95; 对于倒角/倒圆, 参见 CHF、CHR、RND
FRCM	用于倒角/倒圆的模 态进给率	0, >0	FRCM=0 时, 进给率 F 将起 作用	对于单位, 参见 F 和 G94、G95; 对于倒角/倒圆, 参见 RND、RNDM
GOTOB	向后跳转指令	-	和跳转标记符一起使用, 向 程序开始方向跳转至标识的 程序段。	N10 LABEL1: N100 GOTOB LABEL1
GOTOF	GoForward 指令	-	对标签标记的程序段执行 GoTo 操作; 跳转目标在程 序结束方向上。	N10 GOTOF LABEL2 ... N130 LABEL2: ...
IC	使用增量尺寸指定 的坐标		可以为某些轴的终点和中心 点指定尺寸, 不考虑 G90。	N10 G90 X10 Z=IC(20); Z - 增量尺 寸, X - 绝对尺寸

地址	含义	赋值	说明	编程
IF	跳转条件	-	如果满足跳转条件，则对带有下列 标签的程序段 执行 GoTo 操作；否则至下一指令/程序段。一个程序段中，可以包含多个 IF 指令。 关系运算符： == 等于，<> 不等于 > 大于，< 小于 >= 大于等于 <= 小于等于	N10 IF R1>5 GOTOF LABEL3 ... N80 LABEL3: ...
LIMS	带有 G96、G97 的主轴转速上限	0.001 ... 99 999.999	在 G96 功能生效时 - 恒定切削速度以及 G97 时限制主轴转速	参见 G96
MEAS	测量，删除剩余行程	+1 -1	==+1: 测量输入 1，上升沿 =-1: 测量输入 1，下降沿	N10 MEAS=-1 G1 X... Z... F...
MEAW	测量，不删除剩余行程	+1 -1	==+1: 测量输入 1，上升沿 =-1: 测量输入 1，下降沿	N10 MEAW=1 G1 X... Z... F...
\$A_DBB[n] \$A_DBW[n] \$A_DBD[n] \$A_DBR[n]	数据字节 数据字 数据双字 实数数据		PLC 变量的读和写	N10 \$A_DBR[5]=16.3; 写入实数变量 ; 带有偏移位置 5 ; (在 NC 和 PLC 之间确定位置、类型和含义)
\$AA_MM[轴]	机床坐标系中轴的测量结果	-	轴: 测量时横移的轴 (X, Z) 的标识符	N10 R1=\$AA_MM[X]
\$AA_MW[轴]	工件坐标系中轴的测量结果	-	轴: 测量时横移的轴 (X, Z) 的标识符	N10 R2=\$AA_MW[X]
\$AC_MEA[1]	测量任务状态	-	默认条件: 0: 默认条件, 测量头不切换 1: 已接通测量头	N10 IF \$AC_MEAS[1]==1 GOTOF; 测量头已经切换时继续执行程序 ...

地址	含义	赋值	说明	编程
\$A..._..._ TIME	运行时计时器： \$AN_SETUP_TIME \$AN_POWERON_T IME \$AC_OPERATING_ TIME \$AC_CYCLE_TIME \$AC_CUTTING_TI ME	0.0 ... 10+300 分钟（只读 值） 分钟（只读 值） s s s	系统变量： 自控制系统最后一次启动起 的时间 自控制系统最后一次正常启 动起的时间 所有 NC 程序的总运行时 NC 程序的运行时（仅选择 的程序） 刀具动作时间	N10 IF \$AC_CYCLE_TIME== 50.5
\$AC_..._ PARTS	工件计数器： \$AC_TOTAL_PART S \$AC_REQUIRED _PARTS \$AC_ACTUAL_PA RTS \$AC_SPECIAL_PA RTS	0 ... 999 999 999, 整数	系统变量： 总实际计数 工件的设定数量 当前实际计数 工件计数 - 用户指定	N10 IF \$AC_ACTUAL_PART S==15
\$AC_ MSNUM	激活主轴数		只读	
\$P_ MSNUM	编程主轴数		只读	
\$P_NUM_ SPINDLES	配置主轴数		只读	
\$AA_S[n]	主轴 n 的实际速度		主轴编号 n =1 只读	
\$P_S[n]	最后编程的主轴 n 的速度		主轴编号 n =1 只读	
\$AC_ SDIR[n]	主轴 n 当前的旋转 方向		主轴编号 n =1 只读	
\$P_ SDIR[n]	最后编程的主轴 n 的旋转方向		主轴编号 n =1 只读	

地址	含义	赋值	说明	编程
\$P_ TOOLNO	有效刀具号 T	-	只读	N10 IF \$P_TOOLNO==12 GOTOF
\$P_TOOL	有效刀具的有效 D 号	-	只读	N10 IF \$P_TOOL==1 GOTOF
MSG ()	信号	最多 65 个字符	引号中的消息文本	MSG (“消息文本”) ; 单独程序段 ... N150 MSG(); 清除上 一条消息
OFFN	尺寸指定	-	对于设置的刀具半径补偿仅 G41、G42 有效	N10 OFFN=12.4
RND	倒圆	0.010 ... 99 999.999	在两个轮廓间插入规定半径 值的圆弧切线过渡	N10 X... Z... RND=... N11 X... Z...
RNDM	模态倒圆	0.010 ... 99 999.999 0	- 在以下的轮廓角间插入规 定半径值的圆弧切线过渡； 可能的进给率：FRCM=... - 取消模态倒圆	N10 X... Y.... RNDM=.7.3; 模态倒 圆 ON N11 X... Y... N100 RNDM=.0; 模 态倒圆 OFF
RPL	带有 ROT、AROT 时的旋转角度	±0.00001 ... 359.9999	单位为度；当前平面 G17 至 G19 中可编程旋转的角度	参见 ROT、AROT
SET(, , ,) REP()	设置变量字段值		SET:各种值，从指定的元素 到：按照值的数量 REP:相同的值，从指定的元 素到 字段末尾	DEF REAL VAR2[12]=REP(4.5) ; 所有元素值 4.5 N10 R10=SET(1.1,2.3,4.4) ; R10=1.1, R11=2.3, R4=4.4
SF	使用 G33 时的螺纹 起始点	0.001 ... 359.999	单位为度，在 G33 时螺纹起 始角偏移设定的角度值	参见 G33

地址	含义	赋值	说明	编程
SPI (n)	将主轴编号 n 转换为轴标识符		n =1, 轴标识符: 例如, “SP1”或 “C”	
SPOS SPOS(n)	主轴位置	0.0000 ... 359.9999	以度为单位指定; 主轴在指定位置停止 (要达到这一目的, 主轴必须提供合适的技术前提条件: 位置控制) 主轴编号 n: 1	N10 SPOS=.... N10 SPOS=ACP(...) N10 SPOS=ACN(...) N10 SPOS=IC(...) N10 SPOS=DC(...)
SPOSA	主轴位置	0.0000 ... 359.9999	SPOS 和 SPOSA 功能相同, 区别在于程序段切换特性: 使用 SPOS, 到达设定位置时, NC 程序段才启用。 使用 SPOSA 时, 即使尚未到达设定的位置, 也会切换至下一 NC 程序段。	SPOSA=<值> / SPOSA [<n>] = <值>/
STOPFIFO	停止快速加工步骤		特殊功能; 填充缓冲存储器, 直到检测到 STARTFIFO、“缓冲存储器已满”或“程序结束”为止。	STOPFIFO; 单独程序段, 开始填充 N10 X... N20 X...
STARTFIFO	启动快速加工步骤		特殊功能; 从预处理程序缓存载出。	N30 X... STARTFIFO; 单独程序段, 填充结束
STOPRE	预处理停止		特殊功能; 只有在完成 STOPRE 前的程序段时才能解码下一个程序段。	STOPRE; 单独程序段

1.2 位移说明

1.2.1 尺寸编程

在本章中您可以查找到各种指令，利用它们可以对从一个图纸中提取出的尺寸进行直接编程。其优点是，不必对 NC 程序设置进行大量的计算。

说明

在本章中描述的指令在大多数情况下位于 NC 程序的开始部分。这些功能的整理与专利申请无关。举例说工作平面的选择也完全可以在 NC 程序中的其它地方。本节及后面的章节主要给您作一个指南，目的在于介绍 NC 程序的“完整”结构。

典型尺寸一览

大多数 NC 程序的基础部分是一份带有具体尺寸的图纸。

在转换为 NC 程序时有提示帮助，将工件图纸的尺寸准确的接受到加工程序中。它们可以是：

- 绝对尺寸，G90 模态有效用于程序段中的所有轴，直至通过下一个程序段中的 G91 进行撤销。
- 绝对尺寸，X=AC（值）只有这个值适用于给定轴并且不受 G90/G91 的影响。也可以用于所有的轴、以及主轴定位 SPOS、SPOSA 和插补参数 I、J、K。
- 绝对尺寸，X=DC（值）直接按最短路径运行到位置上，只有这个值适用于给定的回转轴并且不受 G90/G91 的影响。也可以用于主轴定位 SPOS、SPOSA。
- 绝对尺寸，X=ACP（值）按正方向逼近位置，只有这个值适用于在机床数据中范围设置在 0...<360 度的回转轴。
- 绝对尺寸，X=ACN（值）按负方向逼近位置，只有这个值适用于在机床数据中范围设置在 0...<360 度的回转轴。
- 增量尺寸，G91 模态有效用于程序段中的所有轴，直至通过下一个程序段中的 G90 进行撤销。
- 增量尺寸，X=IC（值）只有这个值适用于给定轴并且不受 G90/G91 的影响。也可以用于所有的轴、以及主轴定位 SPOS、SPOSA 和插补参数 I、J、K。
- 英寸尺寸，G70 用于程序段中的所有线性轴，直至通过下一个程序段中的 G71 进行撤销。
- 米制尺寸，G71 用于程序段中的所有线性轴，直至通过下一个程序段中的 G70 进行撤销。

1.2 位移说明

- 英寸尺寸如 G70，也用于进给率和带有长度的设置参数。
- 米制尺寸如 G71，也用于进给率和带有长度的设置参数。
- 打开直径编程，DIAMON
- 关闭直径编程，DIAMOF

直径编程，DIAM90，用于带有 G90 的运行程序段。半径编程，用于带有 G91 的运行程序段。

1.2.2 绝对/增量尺寸： G90, G91, AC, IC

功能

使用指令 G90/G91 时，将写入的位置数据 X、Z、... 作为坐标点 (G90) 或轴横移位置 (G91) 评估。G90/91 应用于所有轴。

在某一位移数据不同于 G90/G91 的设定时，可以按程序段方式通过 AC/IC 以绝对尺寸/增量尺寸进行设定。

这些指令不会确定达到终点的路径；这通过 G 功能组 (G0、G1、G2 和 G3....) 提供。更多信息参见章节 "轴运行 (页 42)"。

编程

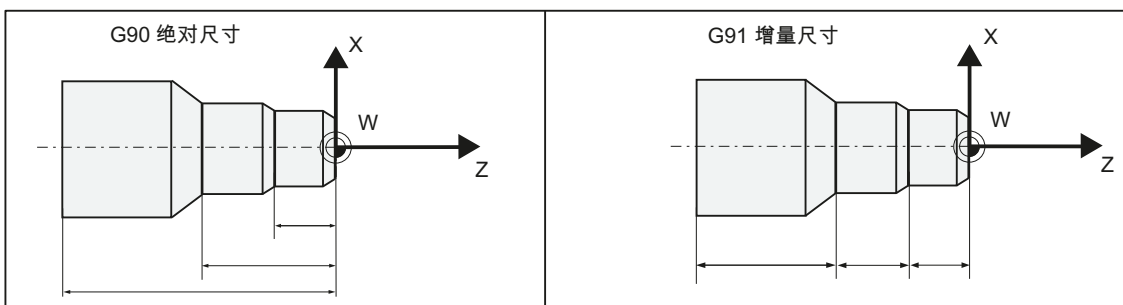
G90 ; 绝对尺寸数据

G91 ; 增量尺寸数据

Z=AC(...) ; 某些轴的绝对尺寸 (此处: Z 轴), 非模态

Z=IC(...) ; 某些轴的增量尺寸 (此处: Z 轴), 非模态

参见图纸中不同的尺寸类型:



绝对尺寸 G90

在绝对尺寸说明中尺寸取决于**当前有效坐标系的零点位置**。这取决于当前有效的偏移：可编程、可设定或无偏移。

程序启动时，G90 对于**所有轴**有效，并且通过 G91（增量尺寸数据）（模态有效）在后续程序段中取消选择前保持有效。

增量尺寸 G91

使用增量尺寸，路径信息的数值对应于**要横移的轴路径**。**移动方向**由符号决定。

G91 应用于所有轴，并且可以通过 G90（绝对尺寸数据）在后续程序段中取消选择。

以 =AC(...), =IC(...) 进行指定

在终点坐标后，写入等号。值必须置于圆括号中。

圆心坐标也可以以绝对尺寸用 =AC(...) 定义。否则，圆心的参考点为圆弧起点。

编程示例

N10 G90 X20 Z90	; 绝对尺寸
N20 X75 Z=IC(-32)	; X-尺寸保持绝对、增量 Z 尺寸
N180 G91 X40 Z2	; 切换到增量尺寸
N190 X-12 Z=AC(17)	; X 轴仍为增量尺寸，Z 轴为绝对尺寸

1.2.3 公制尺寸和英制尺寸： G71, G70, G710, G700

功能

工件标注尺寸可能不同于控制系统的基础系统设定（英寸或毫米），这些标注尺寸可以直接输入到程序中。控制系统会在基础系统中完成必要的转换工作。

编程

G70 ; 英制尺寸
 G71 ; 公制尺寸

G700 ; 英制尺寸，也用于进给 F
 G710 ; 公制尺寸，也用于进给 F

编程示例

```
N10 G70 X10 Z30           ; 英制尺寸
N20 X40 Z50             ; G70 继续有效
N80 G71 X19 Z17.3       ; 从此时开始使用公制尺寸
```

说明

根据**基本设置**控制系统可将所有几何值都用公制或英制尺寸表示。这里刀具补偿值和可设定的零点偏移值包括其显示也作为几何值；同样，进给率 F 的单位分别为毫米/分或英寸/分。

基本设置可以通过机床数据设定。

本说明中所给出的例子均为**公制的基本设置**。

G70 或 G71 用于设定所有与**工件**直接相关的几何数据，英制尺寸或公制尺寸，例如：

- 在 G0,G1,G2,G3,G33, CIP, CT 功能下的位移数据 X, Z, ...
- 插补参数 I, K (也包括螺距)
- 圆弧半径 CR
- **可编程**的零点偏移 (TRANS, ATRANS)

所有其他的几何数据，它们并不是直接的工件数据，例如：进给率、刀具补偿，**可设定的**零点偏移等，不受 **G70/G71** 影响。

与之相反，**G700/G710** 会影响进给率 F（英寸/分、英寸/转或者毫米/分、毫米/转）。

1.2.4 半径 - 直径尺寸: DIAMOF, DIAMON, DIAM90

功能

编程零件加工时，**X 轴**（横向轴）的位移数据为直径尺寸。如有需要，也可以将程序切换到半径尺寸。

DIAMOF 或者 **DIAMON** 分别用半径或者直径尺寸说明 X 轴的终点，实际值相应地显示在工件坐标系中。

DIAM90 则始终用直径尺寸来说明横向轴 X 的实际值，与运行方式（**G90/G91**）无关。这也适用于用读取指令 **MEAS**、**MEAW**、**\$P_EP[x]** 和 **\$AA_IW[x]** 在工件坐标系中的实际值。

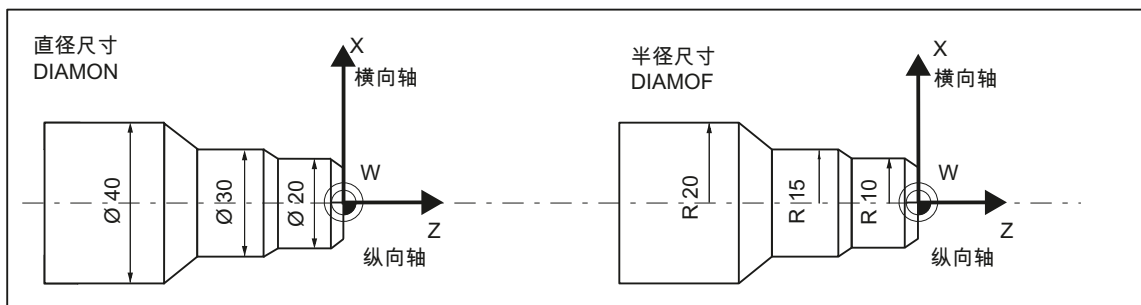
编程

DIAMOF ; 半径尺寸

DIAMON ; 直径尺寸

DIAM90 ; G90 时为直径尺寸，G91 时为半径尺寸

参见以下端面轴的直径和半径尺寸：



编程示例

```
N10 G0 X0 Z0 ; 回起始点
N20 DIAMOF ; 直径输入 OFF
N30 G1 X30 S2000 M03 F0.8 ; X 轴 = 横向轴, 半径编程有效
; 运行至半径位置 X30
N40 DIAMON ; 直径尺寸生效
N50 G1 X70 Z-20 ; 运行到直径位置 X70 和 Z-20
N60 Z-30
N70 DIAM90 ; 绝对尺寸的直径编程和
; 增量尺寸的半径编程
N80 G91 X10 Z-20 ; 增量尺寸
N90 G90 X10 绝对尺寸
N100 M30 ; 程序结束
```

说明

可编程的偏移 TRANS X... 或者 ATRANS X... 始终为半径尺寸。此功能的说明：参见下一章。

1.2.5 可编程的零点偏移: TRANS, ATRANS

功能

在下列情况下可以使用可编程的零点偏移：

- 工件在不同的位置有重复的形状/结构
- 选择了新的参考点说明尺寸
- 粗加工的余量

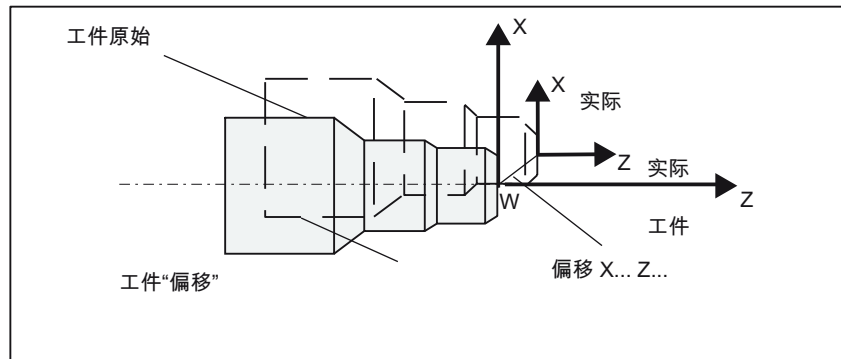
由此就产生一个**当前工件坐标系**。新输入的尺寸便以此坐标系为基准。

偏移适用于所有轴。

说明

由于使用直径编程 (DIAMON) 功能和恒定切削速度 (G96)，工件零点在 X 轴上位于旋转中心。所以在 X 轴上没有或者只有较少的偏移（例如：加工余量）。

参见可编程的零点偏移生效:



编程

TRANS Z... ; 可编程的偏移, 清除之前的偏移、旋转、比例缩放、镜像指令

ATRANS Z... ; 可编程的偏移, 补充当前指令

TRANS ; 不赋值: 清除之前的偏移、旋转、比例缩放、镜像指令

TRANS/ATRANS 必须在单独程序段中编程。

编程示例 1

```
N10 G54  
N20 TRANS Z5 ; 可编程的偏移, Z 轴 5 毫米  
N30 L10 ; 子程序调用, 包含待偏移的几何量  
N40 ATRANS X10 ; 可编程的偏移, x 轴上 10 毫米  
N50 TRANS ; 取消偏移  
N60 M30
```

子程序调用: 请参见章节“子程序 (页 109)”。

编程示例 2

```
G90 G18 G500
T3D1
M4S1500
G0X50 Z10
CYCLE95("CON1:CON1_E", 0.50000, 0.20000, 0.20000, ,0.20000, 0.20000, 0.15000,
9, , ,2.00000)

M4S1200
G0X100Z-10
R0=46
LAB1:
TRANS X=R0 Z-25
AROT RPL=-10
R1=-45
R2=14
R3=34
LAB:
TRANS X=R0 Z-25
AROT RPL=10
R5=R2*COS(R1)
R6=R3*SIN(R1)
G1 Z=R5 X=R6
R1=R1-0.5
IF R1>=-151 GOTOB LAB
R0=R0-0.5
IF R0>=40 GOTOB LAB1
G0X80
```

```
Z50  
AROT  
TRANS  
  
G500  
T5D1  
M4S1000  
G1F0.1  
CYCLE93( 58.00000, -36.00000, 22.00000, 0.90000, , , , , , , 0.10000, 0.10000,  
0.50000, 0.10000, 5, 2.00000)  
G0X80  
Z50  
  
T3D1  
M4S1500  
R0=29  
BB:  
TRANS Z-52 X=R0  
DIAMOF  
R4=720  
LL:  
R1=(3.14159*R4)/180  
R2=SIN(R4)  
G1 X=R2 Z=R1  
R4=R4-0.5  
IF R4>=0 GOTOB LL  
DIAMON  
R0=R0-0.5  
IF R0>=27 GOTOB BB
```

1.2 位移说明

```
G0X80
Z50
M30

;*****轮廓*****
CON1:
X42Z0
X54Z-13
X58
Z-60
X60
M02
CON1_E:;***** 轮廓终点 *****
```

1.2.6 可编程的比例系数：SCALE, ASCALE

功能

用 SCALE, ASCALE 可以为所有坐标轴编程一个比例缩放系数。按此比例放大或缩小各给定轴上的位移。

当前设定的坐标系用作比例缩放的参照标准。

编程

SCALE X... Z... ; 可编程的比例缩放系数，清除之前的偏移、旋转、比例缩放、镜像指令

ASCALE X... Z... ; 可编程的比例缩放系数，补充当前指令

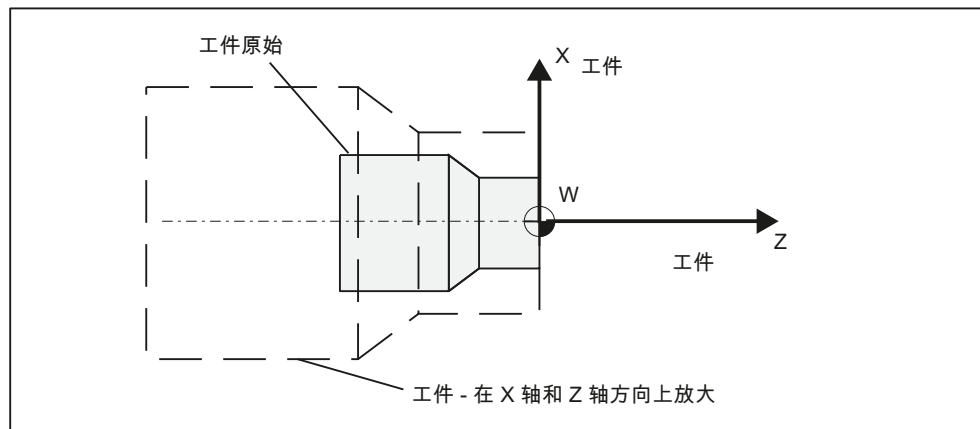
SCALE ; 不赋值：清除之前的偏移、旋转、比例缩放、镜像指令

SCALE、ASCALE 必须在单独程序段中编程。

说明

- 图形为圆时，两个轴的比例系数必须一致。
- 如果在 **SCALE/ASCALE** 有效时编程 **ATRANS**，则偏移量也同样被比例缩放。

参见以下可编程的比例系数示例：



编程示例

```
N10 L10           : 编程的原始轮廓  
N20 SCALE X2 Z2   : X 轴和 Z 轴方向的轮廓放大 2 倍  
N30 L10  
N40 ATRANS X2.5 Z1.8  
N50 L10  
N60 M30
```

子程序调用 - 参见章节“子程序 (页 109)”。

说明

除了可编程的偏移和比例系数外，还存在下列功能：

- 可编程的旋转, **ROT**, **AROT** 和
- 可编程的镜像: **MIRROR**, **AMIRROR**。

此功能通常用于铣削加工。

旋转和镜像的示例：请参见“指令表 (页 12)”部分。

1.2.7 夹紧工件 - 可设定的零点偏移： G54 到 G59, G500, G53, G153

功能

可调零点偏移指定机床上工件零点的位置（相对于机床零点的工件零点偏移）。将工件夹到机床中时确定该偏移，并且操作员必须将该偏移输入到对应的数据字段中。通过从六个可能的组中选择由程序激活值： G54 到 G59。

编程

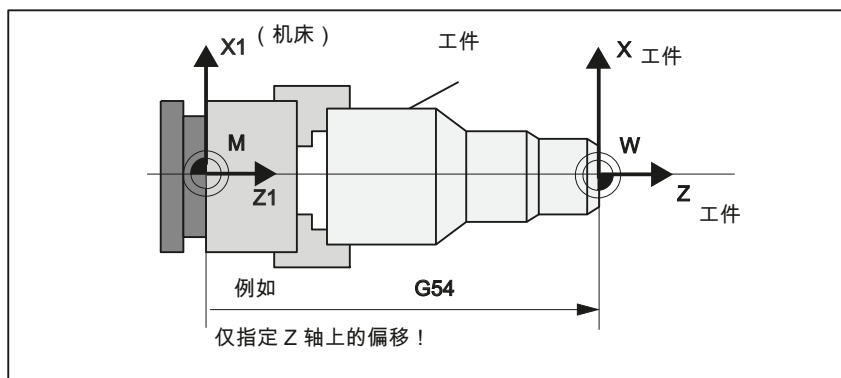
G54 到 G59 ; 1. 到第 6 个可设定的零点偏移

G500 ; 取消可设定的零点偏移 - 模态

G53 ; 取消可设定的零点偏移，非模态，还抑制可编程的偏移

G153 ; 和 G53 一样；另外抑制基本框架

有关可设定的零点偏移的描述，见下图：



编程示例

N10 G54 G0 X50 Z135

N20 X70 Z160

N30 T1 D1

N40 M3 S1000

N50 G0 X20 Z130

N60 G01 Z150 F0.12

N70 X50 F0.1

N80 G500 X100 Z170

N90 M30

1.3 轴运行

1.3.1 快速移动直线插补：G0G0

功能

快速移动 G0 用于快速定位刀具，但不用于直接的工件加工。

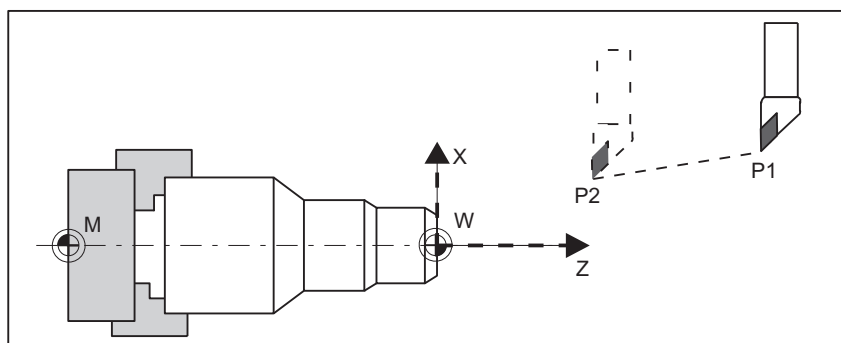
可以按直线轨迹同时运行所有的轴。

每个轴的最大速度(快速移动)均在机床数据中确定。如果只移动一个坐标轴，则该轴以快速移动速度进行移动。如果同时运行两个轴，则选择轨迹速度（生成速度）时需考虑两个轴最大的轨迹速度。

任何编程的进给率（F 字）与 G0 不相关。

G0 一直生效,直到被此 G 功能组中的其它指令 (G0, G1, G2, G3, ...) 取代为止。

有关从点 P1 快速移动到点 P2 的直线插补的描述，见下图：



编程示例

```
N10 G0 X100 Z65
```

说明

另外还可以使用角度 ANG= 进行线性编程。（更多的信息参见章节“轮廓编程 (页 83)”。）

说明

存在用于定位功能的其他 G 功能组（参见章节“准停/连续路径运行 G9, G60, G64 (页 60)”）。在 G60 准停时，可以用一个其它的 G 功能组选择带有不同精度的窗口。对于准停还有一个可选择的程序段方式有效的指令：G9。

在进行定位任务时请注意对几种方式的选择！

1.3.2 进给率 F

功能

进给率 F 是**轨迹速度**，它是所有参与轴速度分量的矢量和。坐标轴速度是轨迹位移在轴位移上的分量。

进给率 F 在 G1,G2,G3,CIP, CT 插补方式中生效，并且一直有效，直到写入一个新的 F 字。欲获得更多信息，参见章节“带进给率的直线插补 G1 (页 44)”和“圆弧插补：G2,G3 (页 45)”。

编程

F...

备注：在**整数值**方式下可以舍去小数点后的数据，例如：F300

进给率 F 的计量单位 G94、G95

进给率 F 字的单位由 G 功能确定：

- G94F 进给率，单位 **毫米/分钟**
- G95F 进给率，单位 **毫米/转**（仅与主轴转速有关！）

备注：

这些单位适用于公制尺寸。根据章节“公制和英制尺寸”，也可以采用英制尺寸的设置。

编程示例

```
N10 G94 F310           ; 进给率，单位毫米/分钟
N20 G01 X60 Z60
N30 M5
N40 S200 M3           ; 主轴旋转
N50 G95 F0.8         ; 进给率，单位毫米/转
N60 G01 X100 Z100
N70 M30
```

备注：切换 G94 - G95 时请写入新的 F 字！

说明

包含 G94 和 G95 的 G 功能组中还包含恒定切削速度：G96 和 G97 功能。这些功能对 S 指令也有影响。

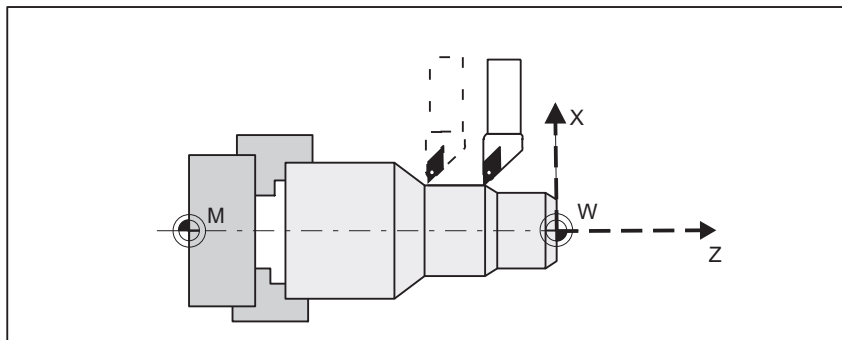
1.3.3 带进给率的直线插补 G1

功能

刀具在直线轨迹上从起始点运动到结束点。**轨迹速度**以已编程的 F 字为准。所有轴可以同时运行。

G1 一直有效，直到被该 G 能组中其它的指令 (G0, G2, G3, ...) 取代为止。

有关直线插补 G1 的描述，见下图：



编程示例

```

N05 G54 G0 G90 X40 Z200 S500 M3          ; 刀具快速移动，主轴转速 = 500 转/分，顺时针旋转
N10 G1 Z120 F0.15                        ; 以进给率 0.15 毫米/转进行直线插补
N15 X45 Z105
N20 Z80
N25 G0 X100                              ; 快速退回
N30 M2                                    ; 程序结束

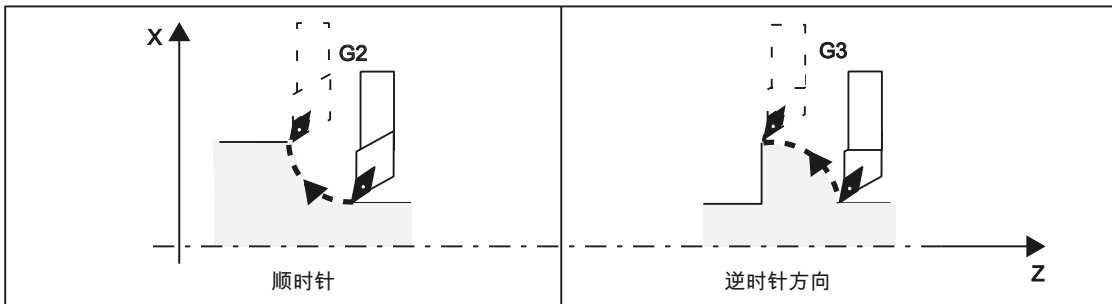
```

说明： 另外还可以使用角度 ANG= 进行线性编程。

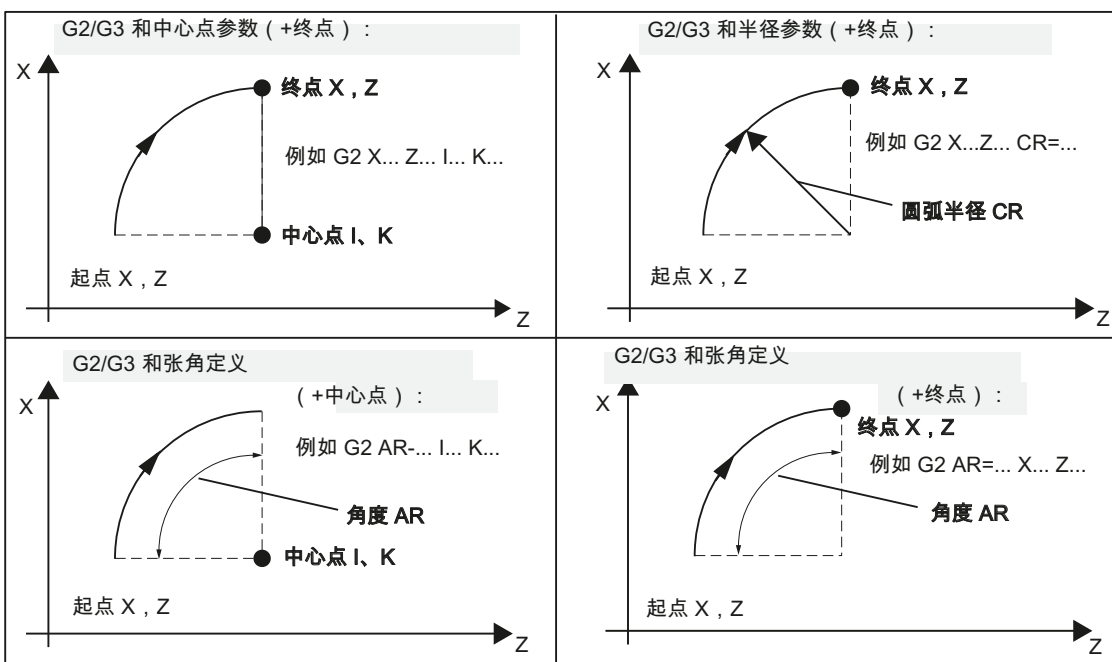
1.3.4 圆弧插补：G2,G3

功能

刀具在圆弧轨迹上从起始点运动到结束点。其方向由 G 功能确定：



所要求的圆弧可以用不同的方式进行描述：



G2/G3 一直有效,直到被 G 功能组中其它的指令(G0, G1, ...)取代为止。
轨迹速度以已编程的 F 字为准。

编程

G2/G3 X... Y... I... J...	; 圆心和终点
G2/G3 CR=... X... Y...	; 圆弧半径和终点
G2/G3 AR=... I... J...	; 圆心和张角
G2/G3 AR=... X... Y...	; 终点和张角
G2/G3 AP=... RP=...	; 极坐标, 以极点为圆心的圆弧

说明

其他进行圆弧编程的方法:

CT - 带有切线连接的圆弧和

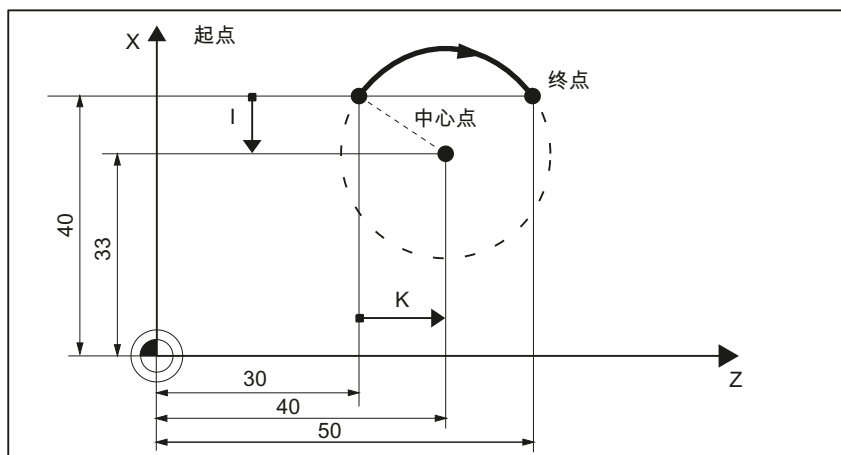
CIP - 通过中间点的圆弧 (见下章)。

圆弧的输入公差

系统仅能接受公差在一定范围内的圆弧。比较起始点和终点的圆弧半径。如果差值在公差以内, 则在内部精确地设定圆心。否则发出报警。

公差值可以通过机床数据设置。

编程示例: 圆心和终点定义



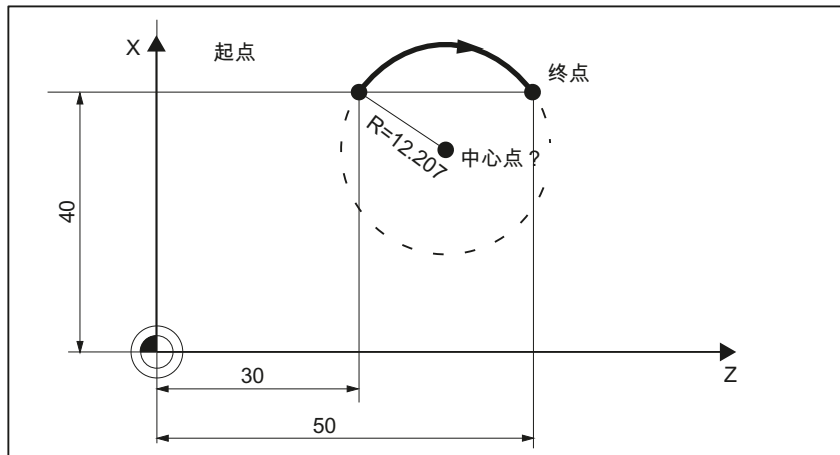
```

N5 G90 Z30 X40 ; N10 的圆弧起点
N10 G2 Z50 X40 K10 I-7 ; 终点和圆心
    
```

说明

圆心值以圆弧起点为基准!

编程示例：终点和半径定义

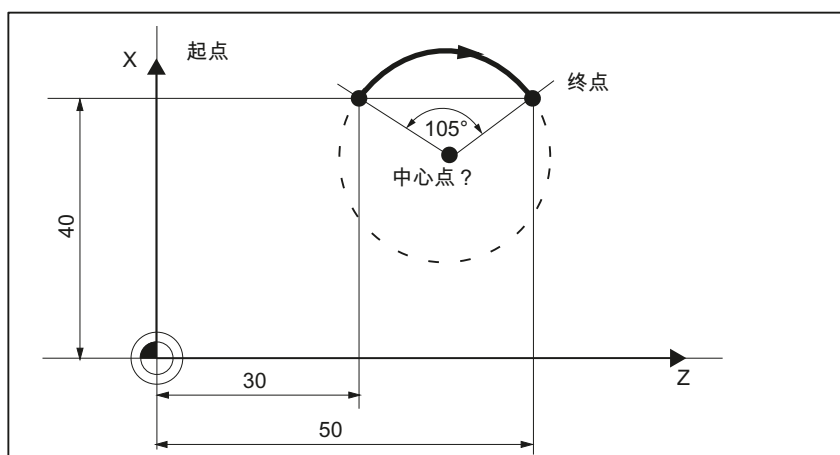


```
N5 G90 Z30 X40 ; N10 的圆弧起点
N10 G2 Z50 X40 CR=12.207 ; 终点和半径
```

说明

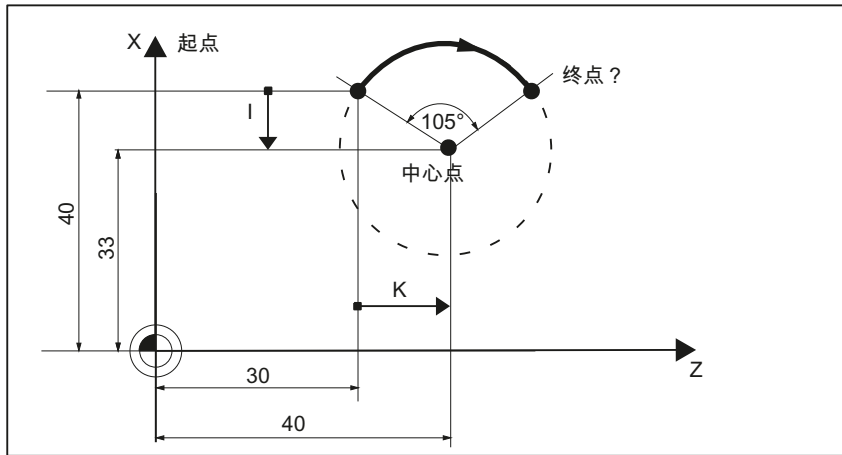
CR=-... 数值前的负号表示选择了一个大于半圆的圆弧段。

编程示例：终点和张角定义



```
N5 G90 Z30 X40 ; N10 的圆弧起点
N10 G2 Z50 X40 AR=105 ; 终点和张角
```

编程示例：圆心和张角定义



```
N5 G90 Z30 X40 ; N10 的圆弧起点  
N10 G2 K10 I-7 AR=105 ; 圆心和张角
```

说明

圆心值以圆弧起点为基准！

1.3.5 通过中间点进行圆弧插补： CIP

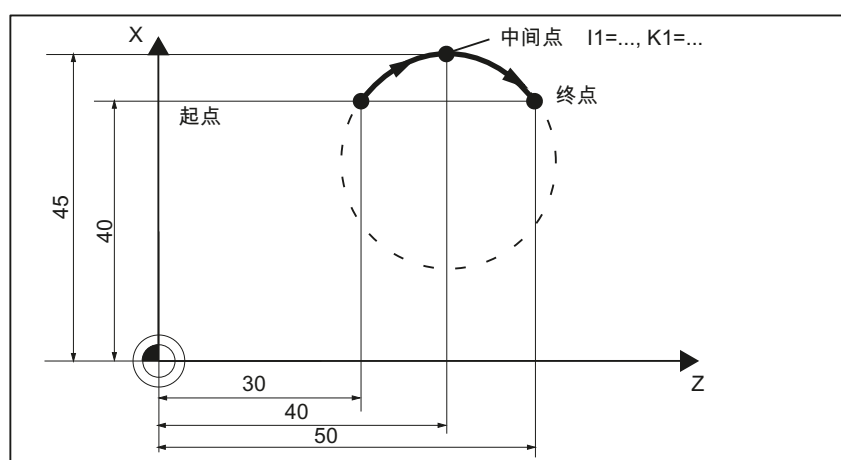
功能

此时，圆弧方向由中间点的位置确定（位于起始点和终点之间）。中间点数据： I1=... 表示 X 轴, K1=... 表示 Z 轴。

CIP 一直有效,直到被 G 功能组中其它的指令(G0, G1, ...)取代为止。

可设定的尺寸输入 G90 或 G91 指令对终点和中间点有效!

有关已知终点和中间点的圆弧的描述，见下图：



编程示例

```
N5 G90 Z30 X40 ; N10 的圆弧起点  
N10 CIP Z50 X40 K1=40 I1=45 ; 终点和中间点
```

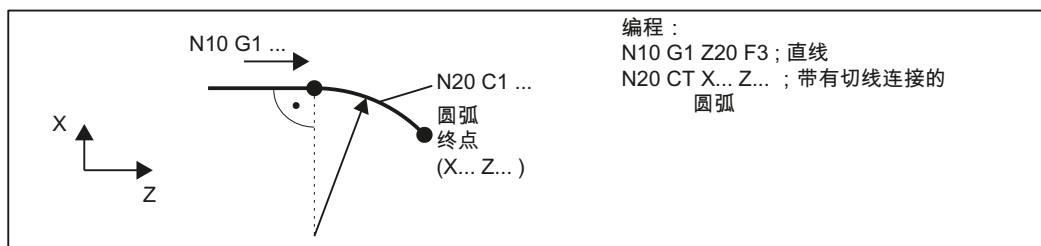
1.3.6 切线过渡圆弧 CT

功能

使用 CT 和当前平面 (G18: Z/X 平面) 中编程的终点, 产生正切连接到上一个轨迹 (圆弧或直线) 的圆弧。

圆弧的半径和圆心可以通过前一轨迹的几何特性和编程的圆弧终点确定。

有关与前一段轮廓为切线过渡的圆弧的描述, 见下图:



1.3.7 有恒定螺距的螺纹切削: G33

功能

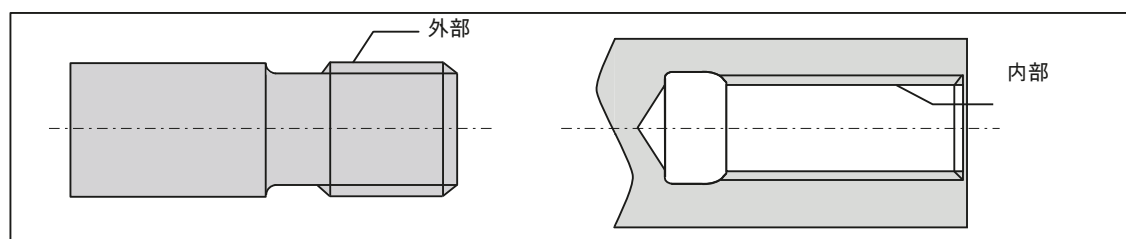
功能 G33 可以用于使用以下类型的恒定螺距加工螺纹:

- 圆柱结构上的螺纹
- 锥形结构上的螺纹
- 外螺纹
- 单头和多头螺纹
- 多行螺纹 (螺纹链)

这需要带位置测量系统的主轴。

G33 一直生效,直到被此 G 功能组中的其它指令 (G0, G1, G2, G3, ...) 取代为止。

有关圆柱体螺纹的内外螺纹的描述, 见下图:



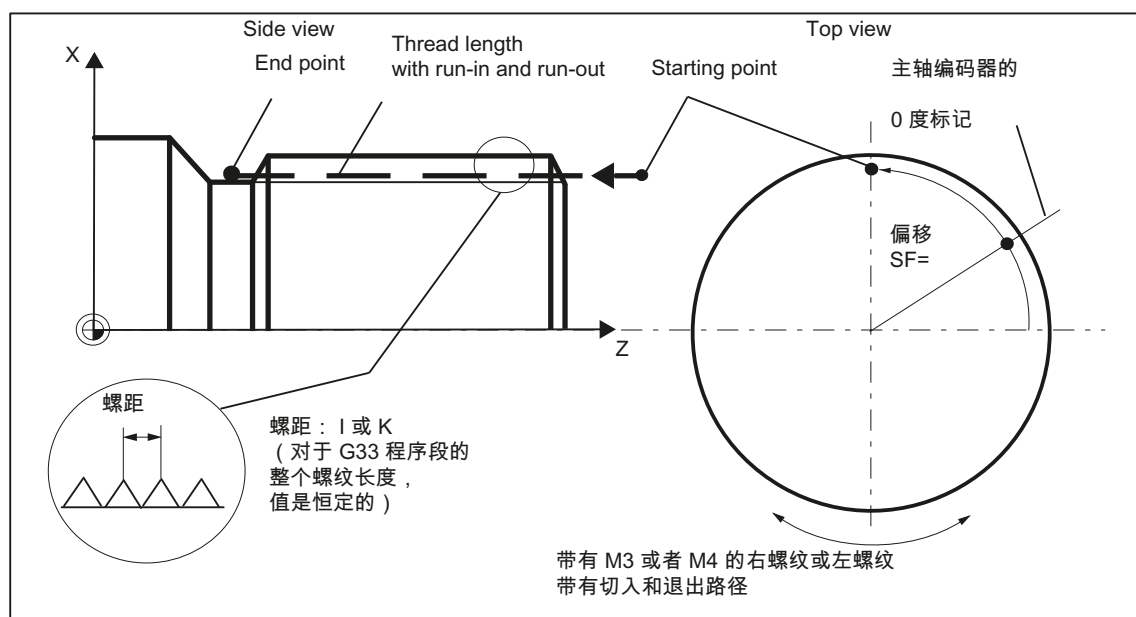
右螺纹或左螺纹

使用主轴旋转方向设置右螺纹或左螺纹（M3 右侧，M4 左侧）。为此，必须在地址 S 下编程旋转值或必须设置转速。

编程

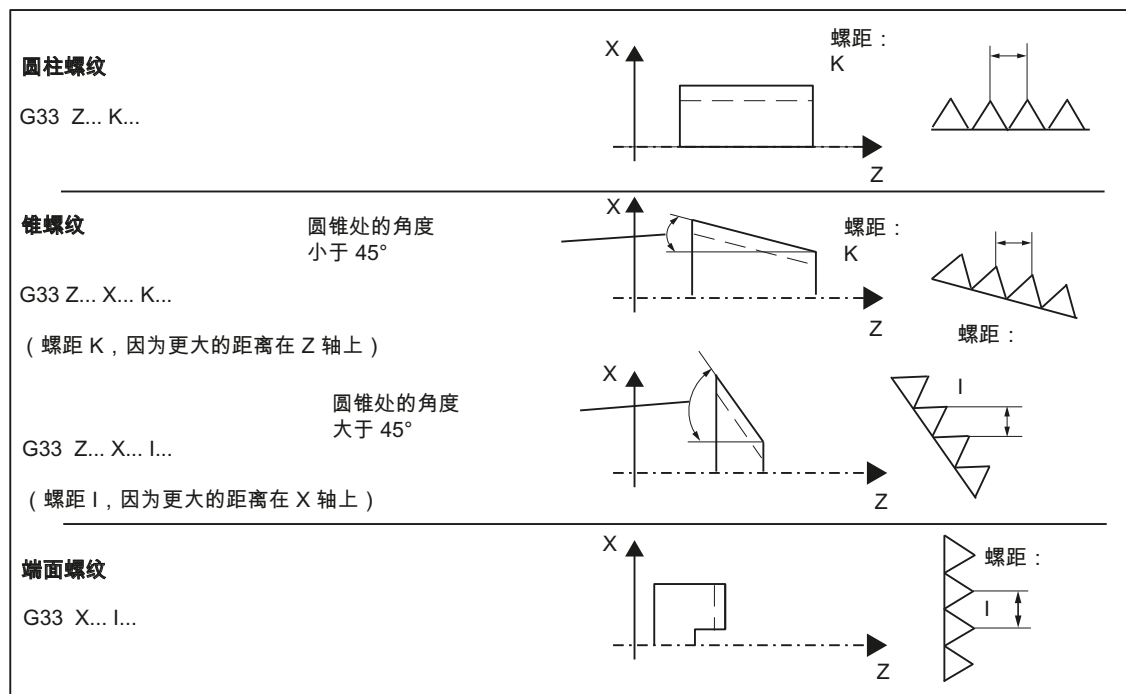
备注：必须针对螺纹长度考虑导入和导出位移。

有关用 G33 对螺纹尺寸编程的描述，见下图：



1.3 轴运行

有关圆柱螺纹、圆锥螺纹和平面螺纹的螺距分配的描述，见下图：



锥螺纹

"对于圆锥螺纹 (必须有 2 轴数据), 较长路程较大螺纹长度时必须使用轴的螺距地址 I 和 K。" 未定义第二个螺距。

起始点偏移 SF=

如果要在偏移部分中加工多头螺纹或单头螺纹, 主轴需要起始点偏移。在地址 SF (绝对位置) 下使用 G33 在螺纹程序段中编程起始点偏移。

如果没有写入起始点偏移 SF, 激活设置数据“螺纹的起始角”中的值 (SD 4200:THREAD_START_ANGLE) 激活。

请注意: 必须始终在设置数据中输入 SF 的编程值。

编程示例

圆柱螺纹，双螺纹，起始点偏移 180 度，螺纹长度（包括导入和导出）100 毫米，螺距 4 毫米/转。

```

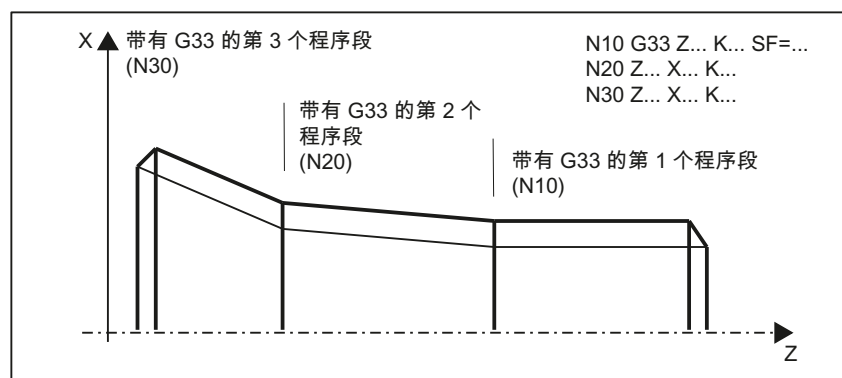
N10 G54 G0 G90 X50 Z0 S500 M3           ; 回起始点，主轴顺时针旋转
N20 G33 Z-100 K4 SF=0                   ; 螺距： 4 毫米/转
N30 G0 X54
N40 Z0
N50 X50
N60 G33 Z-100 K4 SF=180                 ; 第 2 个螺纹；偏移 180 度
N70 G0 X54
N80 Z0
N90 G0X50Z50
N100 M30
    
```

多行螺纹

如果连续编程多个螺纹程序段（多行螺纹），只对定义第 1 个螺纹程序段中的起始点偏移有意义。只在此处使用该值。

在 G64 连续路径模式中自动连接多行螺纹。

参见以下多行螺纹链的示例：



进给轴速度

使用 G33 螺纹，基于主轴转速和螺距确定螺纹长度的轴速度。此时进给率 F 不起作用。但是，其被存储。不能超过机床数据中定义的最大轴速度（快速移动）。这将产生报警。

说明

重要

- 在加工螺纹时主轴速度修调开关应保持位置不变。
- 进给倍率开关在该程序段中不起作用。

1.3.8 G33 的可编程导入和导出路程： DITS, DITE

功能

在螺纹 G33 有时需要另外运行导入与导出路程。在这个范围中可以进行轴的快速运行和制动（在锥形螺纹时为两个轴）。该路程取决于螺距，主轴转速以及轴的动力（设计）。

如果可供输入与输出使用的行程受到限制，则根据情况降低主轴转速，直到该行程够用。为了在这类情况下达到较好的切削值和较短的加工时间、或者为了能简单的解决疑难问题，可以在程序中额外给定输入与输出行程。未给定时设置数据 (SD) 的值生效。程序中的数据写在 SD42010:THREAD_RAMP_DISP[0] ... [1] 中。

如果该路程不足以使轴达到设计的运行加速度，则该轴的加速度超载。对于螺纹导入量会在其后发出报警 22280“编程的导入行程过短”。该报警仅用于提供信息对零件程序执行没有影响。

导出路程是指螺纹结束处的打光距离。这样可以完成轴运行中的无碰撞变更。

编程

DITS=... ; G33 的螺纹的导入行程

DITE=... ; G33 的螺纹的导出行程

DITS 和 DITE 的值或者 SD42010: THREAD_RAMP_DISP

-1 ... < 0: 用设计的加速度进行进给轴的启动/制动。

按照当前编程的 BRISK/SOFT 进行冲击。

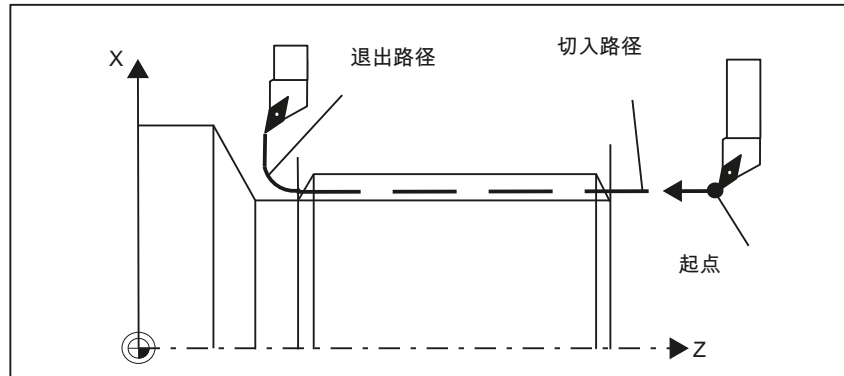
0: 阶梯曲线过后，启动/制动螺纹切削时的进给轴。

> 0: 规定了 G33 的螺纹导入/导出路程。

为了避免报警 22280，在导入或者导出行程很小时要注意轴的加速限制。

说明：复位后/程序开始后 SD42010 的值是 -1。

有关螺纹 G33 的倒圆导入/导出路程的描述，见下图：



编程示例

```
N10 G54  
N20 G90 G0 Z100 X10 M3 S500  
N30 G33 Z50 K5 SF=180 DITS=4 DITE=2 ; 导入 4 mm、导出 2 mm  
N40 G0 X30  
N50 G0 X100 Z100  
N60 M5  
N70 M30
```

1.3.9 变量螺距的螺纹切削：G34,G35

功能

用 G34 或者 G35 可以在程序段中用变量螺距加工螺纹：

- G34 ; 带有(线性)增加螺距的螺纹
- G35 ; 带有(线性)递减螺距的螺纹

两种功能都包括 G33 的特别功能并满足相同的前提条件。

G34 或者 G35 一直有效,直到被 G 功能组中其它的指令(G0, G1, G2, G3, G33...)取代为止。

螺距:

- I 或者 K ; 起始螺纹螺距 毫米/转, 属于轴 X 或者 Z

螺距改变:

在带有 G34 或者 G35 的程序段中, 地址 F 的意义是螺距改变:
螺距每转 (毫米每转) 改变。

- F ; 螺距变化单位 毫米/U²。

说明: 地址 F 除 G34 、 G35 外还包含进给率的含义, 或者在 G4 时的停留时间。那里编程的值被存储。

求出 F

如果已知一个螺纹的起始螺距和最终螺距, 那么就可以根据下面的等式计算出编程的螺距变化率:

$$F = \frac{|K_e^2 - K_a^2|}{2 \times L_G} \left[mm / U^2 \right]$$

这里表示:

K_e 轴目标点坐标的螺距 [毫米/转]

K_a 螺纹起始螺距 (在 I、K 下编程的)[毫米/转]

L_G 螺纹长度 [毫米]

编程

G34 Z... K... F...	;	带有递增螺距的圆柱螺纹
G35 X... I... F...	;	带有递减螺距的平面螺纹
G35 Z... X... K... F...	;	带有递减螺距的圆锥螺纹

编程示例

表格 1-1 圆柱螺纹，然后带有递减螺距

N10 M3 S40	;	接通主轴
N20 G0 G54 G90 G64 Z10 X60	;	回起始点
N30 G33 Z-100 K5 SF=15	;	螺纹，恒定螺距 5 毫米/转
	;	15 度时的使用点
N40 G35 Z-150 K5 F0.16	;	起始螺距 5 毫米/转
	;	螺距递减 0.16 毫米/转，
	;	螺纹长度 50 毫米，
	;	程序段结束时所希望的螺距 3 毫米/转
N50 G0 X80	;	在 x 方向退刀
N60 Z120		
N100 M2		

1.3.10 螺纹插补：G331,G332

功能

需要带位置测量系统的位置控制主轴。

如果主轴和轴的动态响应允许，使用 G331/G332，可以钻削不带补偿卡盘的螺纹。

如果仍然使用补偿卡盘，则减少补偿卡盘补偿的位置差异。这允许在更高的主轴转速下磨削螺纹。

G331 应用于磨削，G332 应用于反方向上的磨削。

通过轴指定磨削深度，例如 Z；通过对应插补参数（此处：K）设定的螺距。

对于 G332，相同螺距的编程与 G331 相同。主轴旋转方向自动反向。

使用 S 编程主轴转速；不带 M3/M4。

在使用 G331/G332 的螺纹磨削前，必须使用 SPOS=... 将主轴置于闭环位置控制模式中。

1.3 轴运行

右螺纹或左螺纹

螺距的符号确定主轴旋转方向：

正向：右侧（和 M3 一样）

负向：左侧（和 M4 一样）

进给轴速度

对于 G331/G332，通过主轴转速和螺距得出螺纹长度的轴的速度。此时进给率 F 不起作用。但是，其被存储。不能超过机床数据中定义的最大轴速度（快速移动）。这将产生报警。

编程示例

公制螺纹 5，

符合表的螺距：0.8 毫米/转，已经预加工的孔：

```
N10 G54 G0 G90 X10 Z5          ; 回起始点
N20 SPOS=0                      ; 位置控制中的主轴
N30 G331 Z-25 K0.8 S600         ; 螺纹磨削，K 正向 = 主轴顺时针旋转，终点 -25 毫米
N40 G332 Z5 K0.8                ; 退回
N50 G0 X10 Z5
N60 M30
```

1.3.11 返回固定点 G75

功能

使用 G75 可以逼近机床上的固定点，例如换刀点。该位置相对于所有轴固定地存储在机床数据中。每个轴最多可以定义 4 个固定点。

它不会产生偏移。每个轴的返回速度就是其快速移动速度。

G75 需要一独立程序段，并按程序段方式有效。机床坐标轴的名称必须要编程！

在 G75 之后的程序段中原先“插补方式”组中的 G 指令 (G0, G1, G2, ...) 将再次生效。

编程

G75 FP=<n> X1=0 Z1=0

说明

FPn 对应轴机床数据 MD30600 \$MA_FIX_POINT_POS[n-1]。如果未编程 FP，则第一个固定点生效。

表格 1-2 说明

指令	含义
G75	接近固定点
FP=<n>	需要逼近的固定点。给定固定点编号：<n> <n> 的值范围：1, 2, 3, 4 如果没有给定固定点编号，则自动逼近固定点 1。
X1=0 Z1=0	需要运行到固定点的机床轴。 将需要同步逼近固定点的轴设定为值“0”。 每根轴以最大轴速度运行。

编程示例

```

N05 G75 FP=1 X1=0           ; 在 x 轴上逼近固定点 1
N10 G75 FP=2 Z1=0         ; 在 z 轴上逼近固定点 2，例如用于换刀
N30 M30                   ; 程序结束

```

说明

为 X1, Z1 编程的位置值（任意值，此处为 0）没有意义，但必须写入。

1.3.12 回参考点运行 G74

功能

用 G74 可以在 NC 程序中执行回参考点运行。各个轴的方向和转速信息储存在机床数据中。

G74 需要单独的程序段并根据程序段方式生效。必须编程机机床轴名称！

在 G74 之后的程序段中原先“插补方式”组中的 G 指令 (G0, G1,G2, ...)将再次生效。

编程示例

```
N10 G74 X1=0 Z1=0
```

备注：忽略程序段中须写入的 X1, Z1 位置值（此时为 0）。

1.3.13 准停/连续路径运行 G9, G60, G64

功能

为了设置程序段分界处的运行性能以及进行程序段转换，一组 G 功能可用于最大程度上满足不同的要求。例如需要坐标轴快速定位，或者通过多个程序段加工路径轮廓时。

编程

G60	； 准停 - 模态
G64	； 连续路径运行
G9	； 准停- 非模态
G601	； 精准停窗口
G602	； 粗准停窗口

准停 G60, G9

当准停(G60 或 G9)功能有效时，在到达准确的目标位置后，速度要在程序段结尾减小到零。

如果该程序段的运行结束并开始执行下一个程序段，则此时可以设定下一个模态 G 功能组。

- G601 精准停窗口

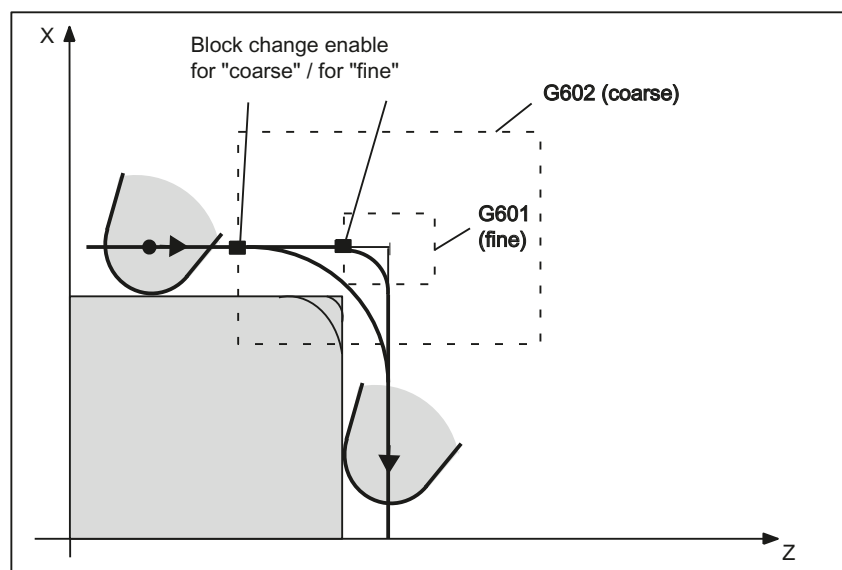
所有轴都达到“精准停窗口”（机床数据值）后，开始进行程序段转换。

- G602 粗准停窗口

所有轴都达到“粗准停窗口”（机床数据值）后，开始进行程序段转换。

在执行多个定位操作时，准停窗口的选择对加工的总时间影响很大。精确调整需要较多时间。

有关 G60 和 G64 速度特性比较的描述，见下图：



编程示例

```
N5 G60                ; 粗准停窗口
N10 G0 G60 Z10       ; 模态准停
N20 X20 Z0           ; G60 继续有效
N30 X30 Z-40
N40 M3 S1000
N50 G1 G601 X35 Z-50 F0.12 ; 精准停窗口
N60 G64 Z-65         ; 转换到连续路径运行方式
N70 X40 Z-70
N80 G0 G9 Z-80       ; 准停只在这个程序段中有效
N90 X45 Z-90         ; 再次进行连续路径运行
N100 M30
```

备注：指令 **G9** 只能使其所在的程序段产生准停；**G60** 一直有效，直到被 **G64** 取代为止。

连续路径运行 **G64**

连续路径运行的目的就是在程序段交界处避免停顿，并使其尽可能以**相同的轨迹速度**（切线过渡）转换到**下一程序段**。该功能以**预定速度控制**执行多个程序段（预读功能）。

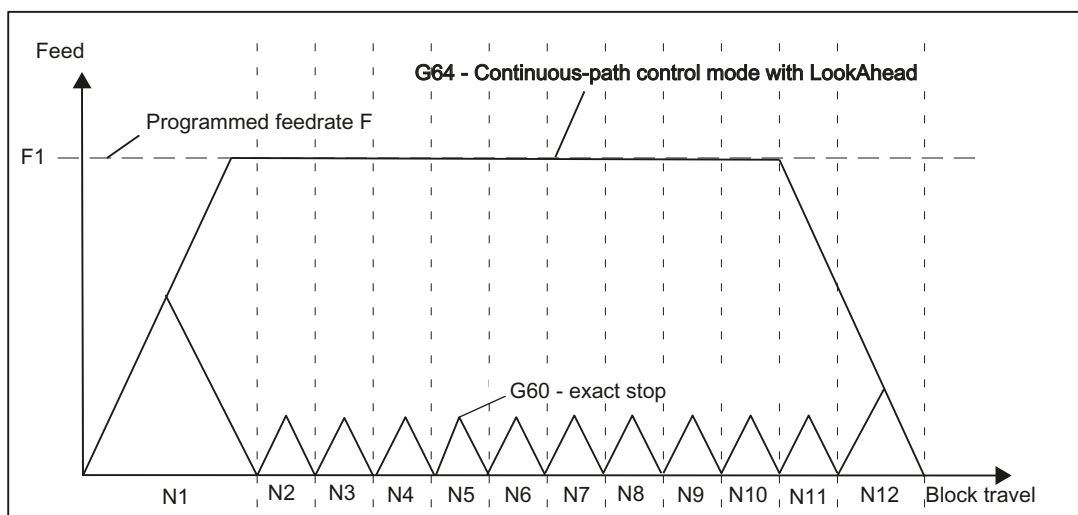
在非切线过渡（拐角）时，必要时必须快速降低速度，使得轴在短时间内速度发生相对较大的变化。这可能会导致急动（加速度变化）。激活 **SOFT** 功能可以降低急动强度。

编程示例

```
N10 G64 G1 Z5 F0.15 M3 S800 ; 连续路径运行
N20 X20 Z0                 ; 再次进行连续路径运行
N30 Z-40
N40 G60 X30 Z-50           ; 转换到准停
N50 X45 Z-70
N60 M30
```

预定速度控制

在使用 G64 的连续路径运行中，控制系统自动事先计算出多个 NC 程序段的速度控制。由此，在几个程序段的近似切线过渡中，可以加速或制动。若加工路径由 NC 程序段中几个较短的位移组成，则使用预读功能可以达到更高的速度。



1.3.14 加速度性能： BRISK, SOFT

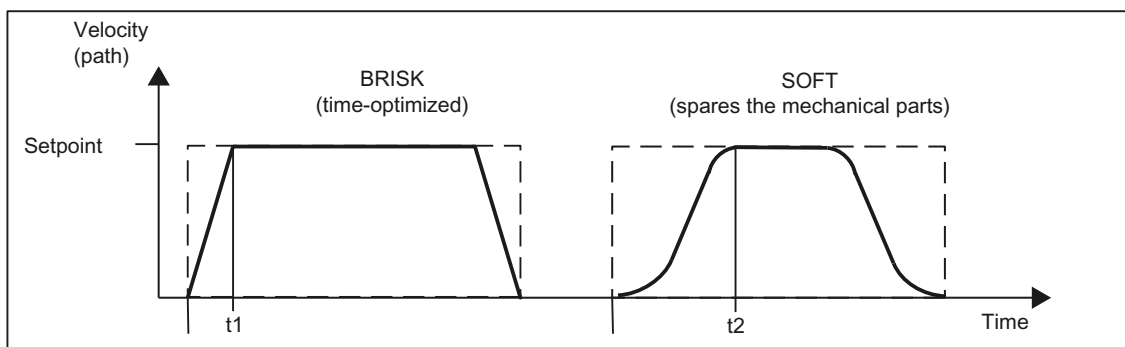
BRISK

机床坐标轴以允许的最大加速度更改其速度，直达到最终速度。BRISK 实现了最佳时间加工。在短时间内就可达到额定速度。在加速过程中会出现一些跳动。

SOFT

机床坐标轴按非线性的连续特征曲线加速，直至达到最终速度。SOFT 通过无冲击加速，减轻了机床负担。制动时也具有相同性能。

有关 BRISK 或 SOFT 轨迹速度的基本过程的描述，见下图：



编程

BRISK ; 跳跃路径加速
 SOFT ; 平滑路径加速

编程示例

```
N10 M3 S200
N20 SOFT G1 X30 Z84 F6.5 ; 平滑路径加速
N30 X46 Z92
N40 BRISK X87 Z104 ; 继续跳跃路径加速
N50 X95 Z110
N60 M30
```


1.3.15 第三轴

前提条件

控制系统必须用于三轴。

功能

依据机床结构可能需要第三轴。这些轴可以设计为直线轴或者回转轴。这些轴的名称由机床制造商确定（例如 B）。

在回转轴中，如果不存在取模轴，运行区域可以设定在 0 ...< 360 度（取模性能）之间，或者 -360 度...+360 度。

在相应的机床结构中，第 3 轴可以与其它轴同时直线运行。如果在一个程序段中用 G1 或 G2/G3 使轴与其它轴 (X,Z) 一起运行，则其不含有进给率 F 的分量。其速度取决于 X, Z 轴的轨迹时间。其运动与其它轨迹轴一起开始并结束。但是，速度不能超过定义的极限值。

如果只使用第 3 轴编程程序段，将在执行 G1 功能时使用有效进给率 F 移动轴。如果该轴是回转轴，F 的测量单位为度/分钟（使用 G94）或主轴的度/转（使用 G95）。

对于该轴可以设置可设定的偏移 (G54 ... G59) 和可编程的偏移 (TRANS, ATRANS)。

编程示例

第 3 轴为回转轴，轴名称为 B	
N5 G94	: F, 单位: 毫米/分钟或度/分钟
N10 G0 X10 Z30 B45	: 快速移动 X-Z 轨迹, 同时运行 B
N20 G1 X12 Z33 B60 F400	: 以 400 毫米/分钟的速度运行 X-Z 轨迹, 同时运行 B
N30 G1 B90 F3000	: 仅单独以 3000 度/分钟的速度运行轴 B 到 90 度位置

回转轴的特殊指令: DC, ACP, ACN

例如对于回转轴 A:	
A=DC (...)	: 绝对尺寸说明, 直接回位 (最短距离)
A=ACP (...)	: 绝对尺寸说明, 从正方向运行至某位置
A=ACN (...)	: 绝对尺寸说明, 从负方向运行至某位
示例:	
N10 A=ACP (55.7)	: 从正方向运行至绝对位置 55.7 度

1.3.16 暂停时间：G4

功能

通过插入一个 **G4 单独程序段**，可以在两个 **NC** 程序段之间使加工在定义的时间内中断；例如用于自由切削。

F...字或者 **S...**字只用于在该程序段中定义时间。在此之前编程的进给率 **F** 和主轴转速 **S** 仍然保持有效。

编程

G4 F... ; 暂停时间，单位秒

G4 S... ; 主轴暂停转数

编程示例

```
N5 G1 F3.8 Z-50 S300 M3 ; 进给率 F，主轴转速 S
N10 G4 F2.5 ; 暂停时间 2.5 秒
N20 Z70
N30 G4 S30 ; 主轴暂停 30 转，相当于在
; S = 300 转/分钟和转速倍率为 100 % 时暂停： t=0.1 分钟
N40 X20 ; 进给和主轴转速继续生效
N50 M30
```

备注

G4 S.. 只有在主轴受控的情况下才生效（当转速给定值同样通过 **S...** 编程时）。

1.4 主轴运动

1.4.1 主轴转速 S，旋转方向

功能

如果机床具有受控主轴，则可以在地址 S 下编程主轴的转速，单位转/分钟。而主轴旋转方向和运行起始点或终点可以通过 M 指令确定。

编程

M3 ; 主轴顺时针旋转

M4 ; 主轴逆时针旋转

M5 ; 主轴停止

备注： S 值取整时可以省略小数点后的位数，例如：S270

说明

如果将 M3 或者 M4 写入**包含轴运行指令的程序段**中，则 M 指令在轴运行指令**之前**生效。

缺省设置： 主轴开始旋转后(M3, M4)，坐标轴才开始运行。同样，M5 也在轴运行指令之前执行。但执行此指令时并不等待主轴停止。主轴停止前坐标轴即开始运行。欲停止主轴，可通过结束程序或按下以下键：



程序段开始时，零主轴转速(S0)生效。

注释： 其他设定可以通过机床数据进行。

1.4 主轴运动

编程示例

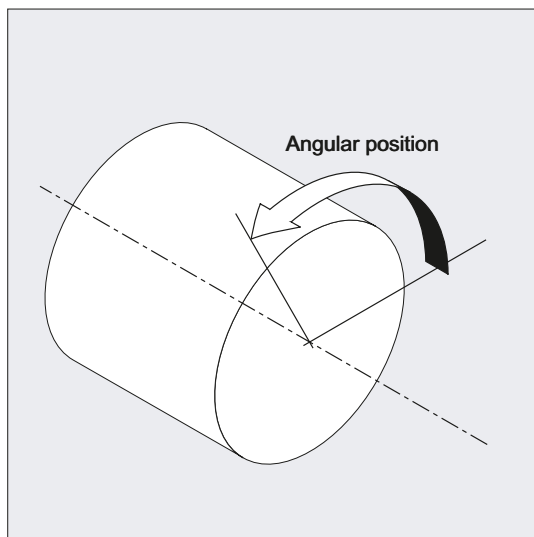
```
N10 G1 X70 Z20 F3 S270 M3 ; 在 X 轴、Z 轴运行前，主轴以 270 转/分钟顺时针方向启动
N20 X90 Z0
N30 Z-40
N40 M5
N50 M4 S290
N60 G1 X100 Z50
N70 S450 Z100 ; 改变转速
N80 X150 Z150
N90 G0 Z180 M5 ; Z 轴运行，主轴停止
N100 M30
```

1.4.2 定位主轴

1.4.2.1 定位主轴 (SPOS, SPOSA, M19, M70, WAITS)

功能

使用 SPOS, SPOSA 或 M19 可以将主轴定位在特定的角度，例如在换刀时。



编程 SPOS, SPOSA 和 M19 时会临时切换至位置控制运行，直到编程下一个 M3/M4/M5/M41...M45 指令。

在进给轴运行中定位

主轴也可以在机床数据中确定的地址下作为轨迹轴，同步轴或者定位轴来运行。指定轴名称后，主轴位于进给轴运行中。使用 M70 将主轴直接切换到进给轴运行。

定位结束

可通过 FINEA，CORSEA 或 IPOENDA 编程主轴定位时的运行结束标准。

如果已经达到所有在程序段中所要加工的主轴或轴的运行结束标准，并且也达到了轨迹插补的程序段转换标准，那么将继续执行下一个程序段。

同步

为了与主轴运行同步，可通过 WAITS 指令等待，直至到达主轴位置。

前提条件

待定位主轴必须能在位置控制方式下运行。

编程

定位主轴：

SPOS=<值>

SPOSA=<值>

M19 / M<n>=19

主轴切换到轴运行方式：

M70 / M<n>=70

确定运行结束标准：

FINEA / FINEA [S<n>]

COARSEA / COARSEA [S<n>]

IPOENDA / IPOENDA [S<n>]

IPOBRKA / IPOBRKA (<轴> [, <时间>]) ; 必须在单独 NC 程序段中编程!

主轴运行同步：

WAITS / WAITS (<n> , <m>) ; 必须在单独 NC 程序段中编程!

含义

SPOS /	将主轴定位至设定的角度
SPOSA:	<p>SPOS 和 SPOSA 功能相同，区别在于程序段切换特性：</p> <ul style="list-style-type: none"> • 使用 SPOS 时，只有到达设定的位置时，才会切换至下一 NC 程序段。 • 使用 SPOSA 时，即使尚未到达设定的位置，也会切换至下一 NC 程序段。 <p><值>： 主轴定位的角度。</p> <p>单位： 度</p> <p>类型： 实数</p> <p>编程位置逼近模式时有如下方案：</p> <p>=AC (<值>)： 绝对尺寸</p> <p>值范围： 0 ... 359,9999</p> <p>=IC (<值>)： 增量尺寸</p> <p>值范围： 0 ... ±99 999,999</p> <p>=DC (<值>)： 直接趋近绝对值</p> <p>=ACN (<值>)： 绝对尺寸，在负方向上运行</p> <p>=ACP (<值>)： 绝对尺寸，在正方向上运行</p> <p>=<值>： 如 DC (<值>)</p>
M<n>=19:	<p>将主轴 (M19 或 M0=19) 或编号为 <n> 的主轴 (M<n>=19) 定位到通过 SD43240 \$SA_M19_SPOS 设定的角度和 SD43250 \$SA_M19_SPOSMODE 中设定的位置逼近模式。</p> <p>到达设定位置时，NC 程序段才跳转。</p>
M<n>=70:	<p>将主轴 (M70 或 M0=70) 或编号为 <n> 的主轴 (M<n>=70) 切换到进给轴运行方式。</p> <p>不逼近定义的位置。主轴运行方式切换后，继续执行 NC 程序段。</p>
FINEA:	在到达“精准停”时运动结束
COARSEA:	在到达“粗准停”时运动结束
IPOENDA:	当到达插补器停止时结束运动

- IPOBRKA:** 可以在制动斜坡上进行程序段转换
<轴>: 通道轴识别符
<时间>: 程序段转换时间参考制动斜坡
单位: 百分比
值范围: 100 (制动斜坡启用时间) ... 0 (制动斜坡结束)
未设定参数<时间>时, 设定数据的当前值生效:
SD43600 \$SA_IPOBRAKE_BLOCK_EXCHANGE
提示:
时间为“0”时 IPOBRKA 与 IPOENDA 相同。
- WAITS:** 设定主轴的同步指令
执行以下程序段时系统将会等待, 设定的主轴和上一个 NC 程序段中使用 SPOSA 编程的主轴到达了终点位置 (精准停)。
M5 后 WAITS: 等待, 直至主轴停止。
M3/M4 后 WAITS: 等待, 直到主轴到达其设定点速度。
<n>, <m>: 同步指令适用的主轴编号
未设定主轴编号或主轴编号为“0”时, WAITS 生效于主轴。

说明

每个 NC 程序段可以有三个主轴定位说明。

说明

在增量尺寸 IC (<值>) 中, 可通过多次旋转进行主轴定位。

说明

如果在 SPOS 之前使用 SPCON 激活了位置控制, 则该运行方式一直生效, 直至编程了 SPCOF。

说明

控制系统会根据编程顺序自动识别到进给轴运行的过渡。因此不一定需要在零件程序中进行 M70 的显式编程。也可编程 M70, 以提高零件程序的可读性。

编程示例

示例 1： 负向旋转定位主轴

将主轴 1 负向旋转定位在 250°：

```
N10 SPOSA[1]=ACN(250)
```

必要时制动主轴，并反向加速进行定位。

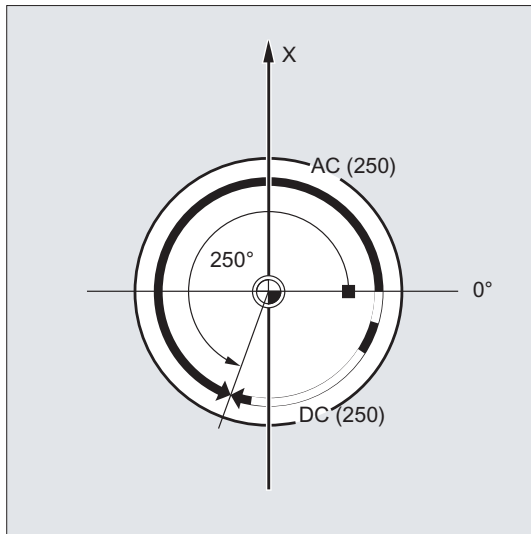
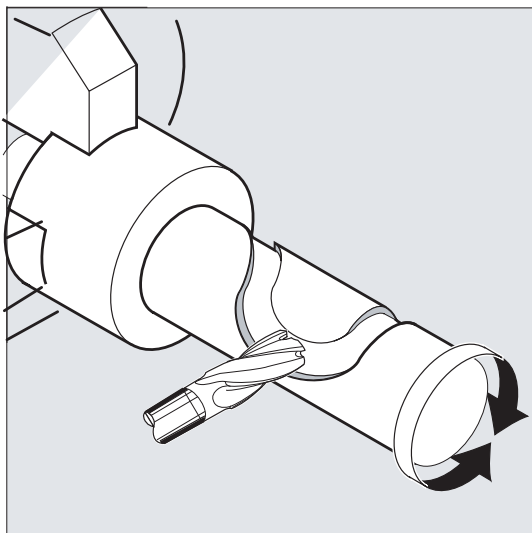


图 1-1 角度位置

示例 2： 在进给轴运行中定位主轴



编程方法 1:

```
N10 G0 X100 Z100  
N20 M3 S500  
N30 G0 X80 Z80  
N40 G01 X60 Z60 F0.25  
N50 SPOS=0  
N60 X50 C180  
N70 Z20 SPOS=90  
N80 M30
```

位置控制激活，主轴 1 定位在 0 处，在下一个程序段中以进给轴方式运行。

主轴（C 轴）在线性插补中和 x 同步运行。

主轴定位在 90 度。

编程方法 2:

```
N10 G0 X100 Z100  
N20 M3 S500  
N30 G0 X80 Z80  
N40 G01 X60 Z60 F0.25  
N50 M2=70  
N60 X50 C180  
N70 Z20 SPOS=90  
N80 M30
```

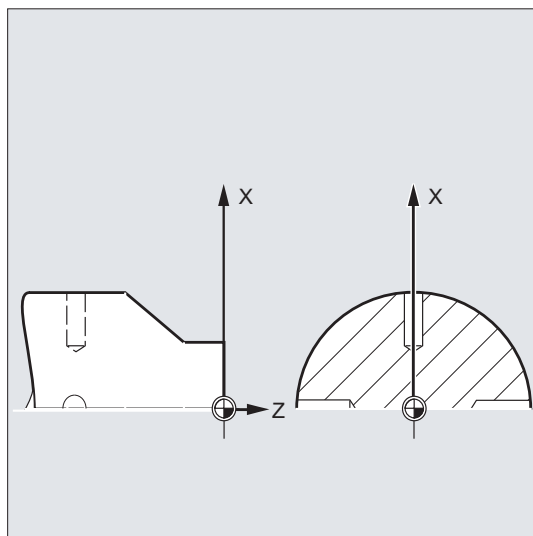
主轴切换至进给轴运行。

主轴（C 轴）在线性插补中和 x 同步运行。

主轴定位在 90 度。

示例 3：在车削件上钻十字孔

在此车削件上钻十字孔。运行的驱动主轴在零度停止，然后逐渐旋转 90 度，再停止，依次类推。



1.4 主轴运动

G0 X100 Z100	
N110 S2=1000 M2=3	激活十字钻。
N120 SPOSA=DC(0)	主主轴直接定位在 0°， 程序段立即执行跳转。
N125 G0 X34 Z-35	定位主轴时激活钻头。
N130 WAITS	等待，直到主主轴到达其位置。
N135 G1 G94 X10 F250	进给率以毫米/分钟为单位（G96 仅适用于多刃车刀和同步主轴，而不适用于横滑板上的动力刀架）。
N140 G0 X34	
N145 SPOS=IC(90)	定位时停止读取，并在正方向旋转 90°。
N150 G1 X10	
N155 G0 X34	
N160 SPOS=AC(180)	以主轴零点为基准定位至 180°位置。
N165 G1 X10	
N170 G0 X34	
N175 SPOS=IC(90)	主轴从绝对位置 180°正方向旋转 90°，停止在 270°的绝对位置。
N180 G1 X10	
N185 G0 X50	
M30	

1.4.2.2 定位主轴 (SPOS, SPOSA, M19, M70, WAITS): 其它信息

其它信息

使用 SPOSA 定位

程序段转换以及程序执行不受 SPOSA 影响。可以同时定位主轴和执行后续 NC 程序段。所有在程序段中编程的功能（除了主轴）达到它们的程序段结束标准后，会转换程序段。主轴定位可以占用多个程序段（参见 WAITS）。

说明

如果一个后续程序段中包含一个会生成隐式预处理停止的指令，那么直到所有的定位主轴都固定不动时才执行该程序段。

使用 SPOS / M19 定位

只有当所有程序段中编程的功能达到它们的程序段结束标准（例如，PLC 对所有辅助功能进行了响应，所有轴到达终点），并且主轴已到达编程位置时，才会转换程序段。

运行速度：

定位的速度和延时特性存储在机床数据中。设定的值可通过编程或同步进行修改。

主轴位置设定：

由于指令 G90/G91 在此不生效，必须使用尺寸数据如 AC, IC, DC, ACN, ACP。如果未进行设定，自动以 DC 运行。

带 WAITS 的主轴运动同步

使用 WAITS 可在 NC 程序中标注一个位置，在该位置等待，直到在前面的 NC 程序段中用 SPOSA 编程的主轴到达各自的位置。

示例：

```
N10 SPOSA[1]=180 SPOSA[1]=0
G01 X34
G00 X10
N40 WAITS(1) ; 在程序段中等待，直到主轴 1 到达程序段 N10 中指定的位置。
```

M5 之后，可以用 WAITS 等待主轴达到停止状态。M3/M4 之后，可以用 WAITS 等待，直至主轴达到设定的转速/旋转方向。

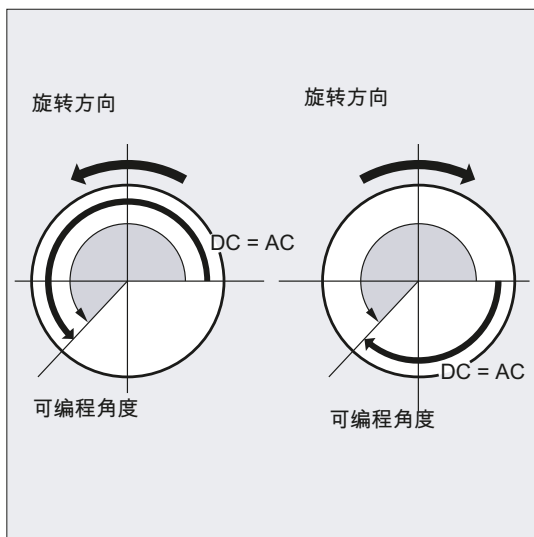
说明

如果主轴未按同步标记进行同步，那么正向旋转方向由机床数据定义（出厂时的状态）。

1.4 主轴运动

旋转中定位主轴 (M3/M4)

当 M3 或 M4 生效时，主轴到达编程的值后静止。



DC 和 AC 数据之间没有区别。在这两种情况下一直接 M3/M4 选定的方向旋转，直至到达绝对终点位置。使用 ACN 和 ACP 时，必要时进行制动并保持相应的逼近方向。使用 IC 时，主轴从当前位置旋转到设定的值。

从静止状态 (M5) 定位主轴

从静止状态 (M5) 开始按照设定精确运行所编程的路径。

1.4.3 齿轮级

功能

最多可以为一个主轴配置 5 个齿轮级来调节转速/扭矩。

编程

通过程序中的 M 指令来选择齿轮级：

M40 ; 自动齿轮级选择

M41 到 M45 ; 齿轮级 1 至 5

1.5 特殊车削功能

1.5.1 恒定切削速度：G96,G97

功能

前提条件： 必须有控制主轴。

G96 功能生效后，主轴速度可以自动适应当前工件直径（运行轴），这样的话，编程设好的切削速率 S 就可以在刀沿上保持恒定。

主轴转速 \times 直径 = 常数

自带 G96 的程序段起，S 字作为切削速率。在被功能组（G94, G95, G97）的其他 G 功能取消前，G96 一直有效。

编程

G96 S... LIMS=... F... ; 恒定切削速度“开”

G97 ; 恒定切削速度“关”

S ; 切削速率，度量单位为毫米/分钟。

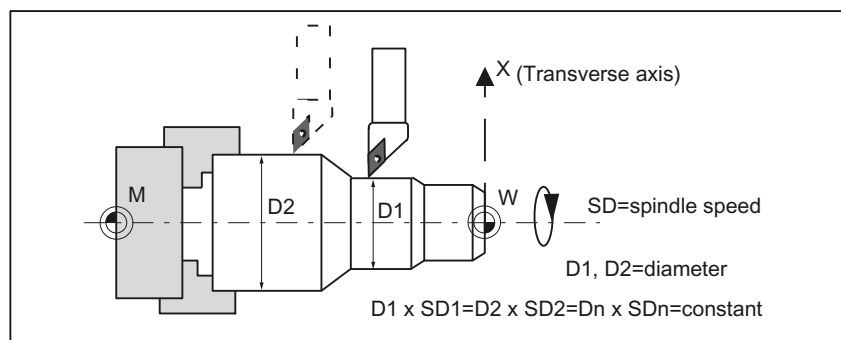
LIMS= ; G96、G97 生效的主轴转速上限

F ; 进给率，单位毫米/转 - 对于 G95

说明：

如果先前是 G94 而不是 G95 生效，必须写入合适的新 F 值。

有关恒定切削速度 G96 的描述，见下图：



快速移动

G0 快速移动时，转速不变。

例外： 如果轮廓接近快速移动且下个程序段包含 G1、G2、G3、CIP 或 CT（Contour block：轮廓程序段）时，轮廓程序段速度适用于带 G0 的逼近程序段。

转速上限 LIMS =

从小直径向大直径进行工件加工时，主轴转速急剧增大。在这种情况下，建议限定转速上限 LIMS = ...。LIMS 只对 G96 和 G97 有效。

通过制定 LIMS =...，覆盖输入至设定数据(SD 43230:

SPIND_MAX_VELO_LIMS) 的值。如果没有写入 LIMS，则此设定数据 (SD) 生效。

通过 G26 编程或机床数据设定的转速上限不可以用 LIMS= 进行覆盖。

取消恒定切削速度： G97

“恒定切削速度”功能可通过 G97 取消。如果 G97 生效，编程设定的 S 字在 RPM 中为**主轴速度**。

如果没有编程设定 S 字，那么主轴最终则为 G96 功能生效时的设定速度。

编程示例

N10 M3 S1000	； 主轴旋转方向
N20 G96 S120 LIMS=2500	； 激活恒定切削速度 120 毫米/分钟，最高转速 2,500 转/分钟
N30 G0 X150	； 由于 N31 程序段支持 G0，速度不变
N40 X50 Z20	； 由于 N32 程序段支持 G0，速度不变
N50 X40	； 沿轮廓运行，新速度自动设置为 N40 程序段开始时所需的值。
N60 G1 F0.2 X32 Z25	； 进给率 0.2 毫米/转
N70 X50 Z50	
N80 G97 X10 Z20	； 取消恒定切削速度：
N90 S600	； 新主轴转速，转/分钟。
N100 M30	

说明

G96 功能也可以通过 G94 或 G95（同属 G 功能组）取消。在这种情况下，如果没有编程设定新 S 字，最终编程主轴转速 S 会为剩下的加工工序生效。

可编程的偏移量 TRANS 或 ATRANS（参见章节“可编程的零点偏移： TRANS, ATRANS (页 34)”）不可用于 X 运行轴，也不可仅使用低值。工件零点应定位于旋转中心。只有这样才能保证 G96 的准功能。

1.5.2 倒圆、倒角

功能

在轮廓角中可以加入倒角（CHF 或 CHR）或倒圆（RND）。如果希望用同样的方法对若干轮廓拐角连续进行倒圆，使用“模态倒圆”（RNDM）命令。

可以用 FRC（非模态）或 FRCM（模态）命令给倒角/倒圆编程进给率。如果没有编程 FRC/FRCM，那么一般进给率 F 生效。

编程

CHF=...	； 插入倒角，值： 倒角底长
CHR=...	； 插入倒角，值： 倒角腰长
RND=...	； 插入倒圆，值： 倒圆半径
RNDM=...	； 模态倒圆： 值 > 0: 倒圆半径，模态倒圆功能 ON 自所有后面的轮廓角中插入倒圆。 值 = 0: 取消模态倒圆
FRC=...	； 用于倒角/倒圆的非模态进给率， 值>0，在 G94 时进给率以毫米/分为单位，在 G95 时以毫米/转为单位。
FRCM=...	； 用于倒角/倒圆的模态进给率， 值 > 0: 进给率以毫米/分（G94）或者毫米/转（G95）为单位， 倒角/倒圆的模态进给率 ON 值 = 0: 倒角/倒圆的模态进给率 OFF 在倒角/倒圆时进给率 F 起作用。

说明

在当前平面 G18 到 G19 执行倒角/倒圆功能。

在包含轴运行到轮廓角指令的程序段中写入指令 CHF= ... 或者 CHR=... 或者 RND=... 或者 RNDM=...。

如果其中一个程序段的轮廓长度不够，则在倒角或者倒圆时自动削减编程值。

不插入倒角/倒圆，如果：

- 三个以上的连续程序段不包含平面移动的命令，
- 转换平面

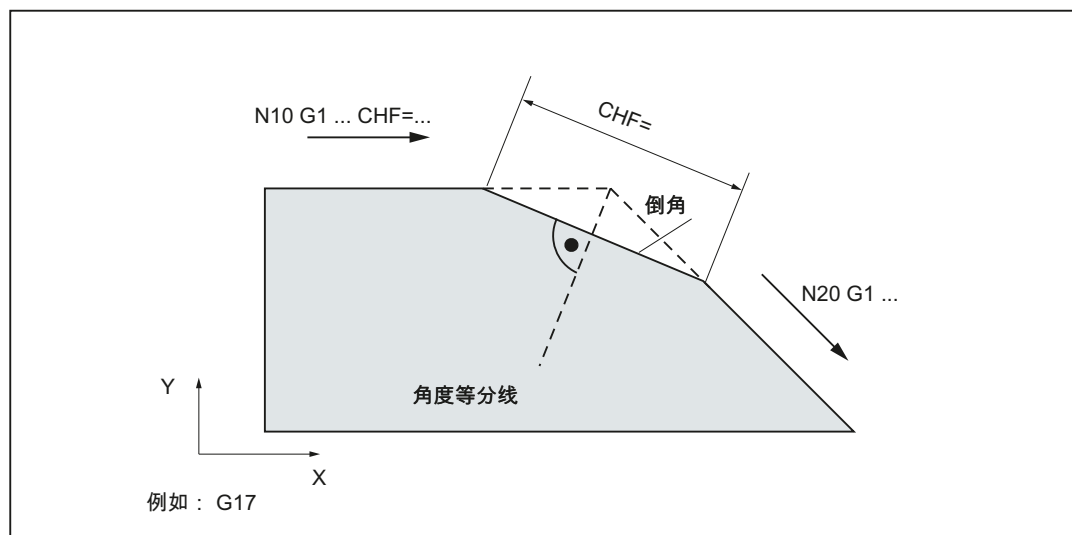
如果以 G0 进行倒角，则 F,FRC,FRCM 无效。

如果倒角/倒圆时进给率 F 生效，则在正常情况下进给率为离开轮廓角程序段中的值。其他设定可以通过机床数据进行。

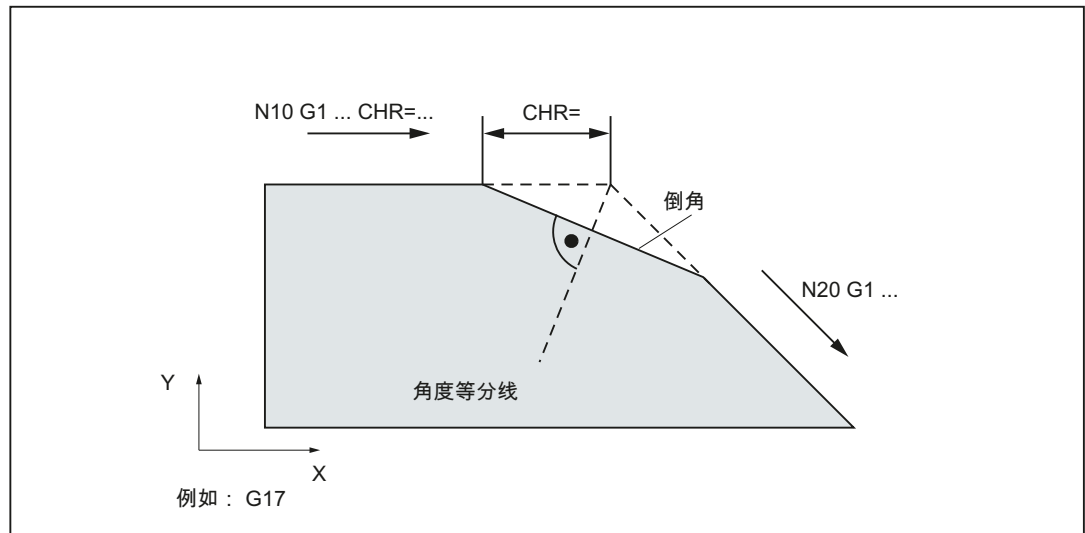
倒角 CHF 或者 CHR

在任意组合的直线和圆弧轮廓间插入一直线轮廓段。此直线倒去棱角。

用 CHF 添加倒角，举例：两条直线之间，见下图。



用 CHR 添加倒角，举例：两条直线之间，见下图。



倒角编程举例：

```

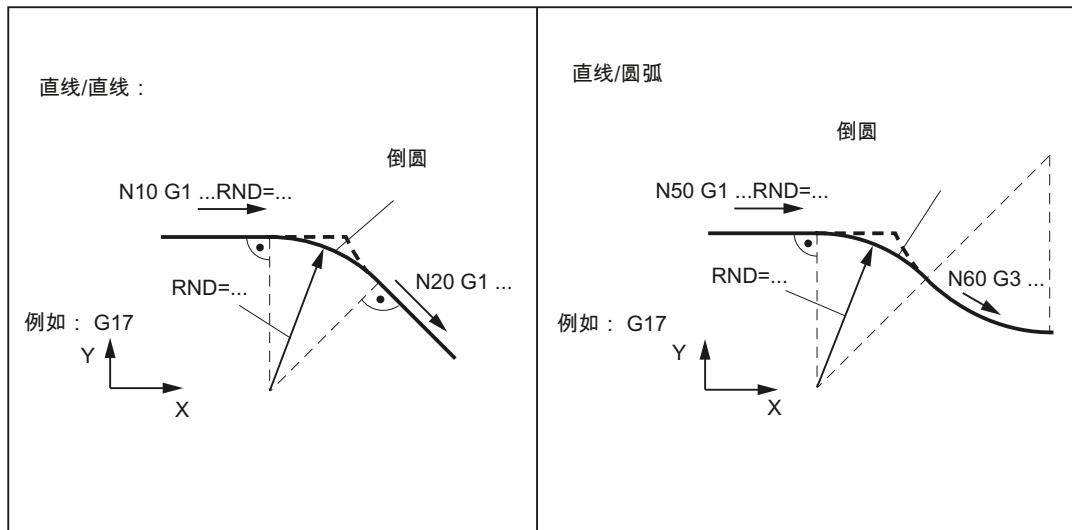
N10 G0 X100 Z100 G94 F100
N20 G1 X80 CHF=5           ; 插入倒角，倒角底长 5 毫米
N30 X50 Z60
N40 X40 Z50
N50 G1 X30 CHR=7          ; 插入倒角，倒角腰长 7 毫米
N60 X10 Z20
N70 X0 Z0
N80 G1 FRC=200 X100 CHR=4 ; 插入倒角，进给率 FRC
N90 X120 Z20
N100 M30

```

倒圆 RND 或者 RNDM

在任意组合的**直线和圆弧轮廓**间插入一圆弧，圆弧和轮廓相切。

有关举例：插入倒圆的描述，见下图。



倒圆编程举例

```

N10 G0 X100 Z100 G94 F100
N20 G1 X80 RND=8 ; 插入 1 个倒圆，半径 8 毫米，进给率 F
N30 X60 Z70
N40 X50 Z50
N50 G1 X40 FRCM= 200 RNDM=7.3 ; 模态倒圆，半径 7.3 毫米，专用进给率 FRCM (模态)
N60 G1 X20 Z10 ; 继续插入倒圆 - 直至 N70
N70 G1 X0 Z-45 RNDM=0 ; 取消模态倒圆
N80 M30
    
```

1.5.3 轮廓编程

功能

如果在加工图纸中不能看出直接的轮廓终点数据，则在确定直线时也可以使用角度数据。在轮廓角中可以加入倒角或倒圆。在构成角度的程序段中写入各个指令 **CHR= ...** 或者 **RND=...**

在程序段中用 **G0 或者 G1** 可以进行轮廓编程。

理论上可以链接任意多的直线路径，并在其中添加倒圆或者倒角。必须可以由点数据和/或角度数据确定每条直线。

编程

ANG=...	;用于确定一条直线的角度数据
RND=...	; 插入倒圆，值： 倒圆半径
CHR=...	; 插入倒角，值： 倒角腰长

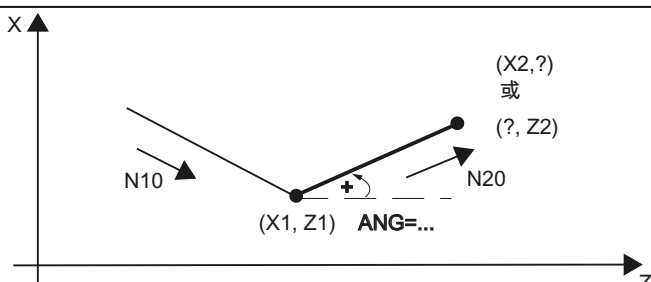
说明

在程序段中编程半径和倒角时，只用插入半径而与编程顺序无关。

角度 ANG=

如果对于直线仅平面终点坐标已知，或对于跨多个程序段的轮廓仅最后的终点已知，则可通过设定角度确定唯一的直线轨迹。角度与 Z 轴有关（正常情况下：**G18** 有效）。正角度朝向逆时针方向。

有关用于确定一条直线的角度数据的描述，见下图：

轮廓	编程
	<p>N20 的终点未知</p> <p>N10 G1 X1 Z1 N20 X2 ANG=...</p> <p>或：</p> <p>N10 G1 X1 Z1 N20 Z2 ANG=...</p> <p>以上数值仅为象征性数值</p>

1.5 特殊车削功能

有关多行轮廓举例的描述，见下图：

轮廓	编程
	<p>N20 的终点未知 N10 G1 X1 Z1 N20 ANG=30 N30 X5 Z3 ANG=60 N40 M30</p> <p>以上数值仅为示例。</p>
	<p>N20 的终点未知，插入倒圆： N10 G1 X1 Z1 N20 ANG=30 RND=0.1 N30 X5 Z3 ANG=60 模拟量 插入倒角： N10 G1 X1 Z1 N20 ANG=30 CHR=0.1 N30 X5 Z3 ANG=60</p>
	<p>N20 的终点已知 插入倒圆： N10 G1 X1 Z1 N20 X2 Z2 RND=0.5 N30 X5 Z3 模拟量 插入倒角： N10 G1 X1 Z1 N20 X2 Z2 CHR=0.2 N30 X5 Z3</p>
	<p>N20 的终点未知 插入倒圆： N10 G1 X1 Z1 N20 ANG=30 RND=0.3 N30 X5 Z3 ANG=60 RND=0.3 N40 X3 Z4 模拟量 插入倒角： N10 G1 X1 Z1 N20 ANG=30 CHR=0.3 N30 X5 Z3 ANG=60 CHR=0.3 N40 X3 Z4 N50 M30</p>

1.6 刀具和刀具补偿

1.6.1 通用说明（车削）

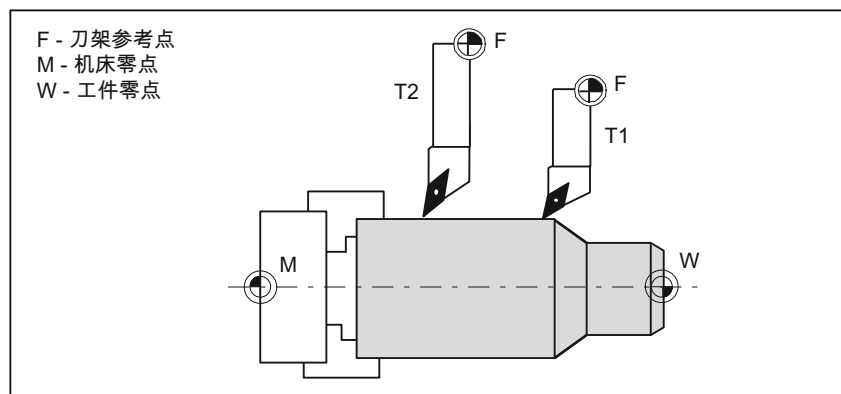
功能

在创建工件加工程序时无需考虑刀具长度或者刀尖半径. 可以直接编程工件尺寸, 例如: 根据图纸直接编程。

在专门的数据区单独输入刀具参数。

然后只需在程序中通过补偿数据调用所需的刀具。控制系统利用这些数据执行所要求的轨迹补偿, 从而加工出说明的工件。

有关使用不同刀具尺寸加工工件的描述, 见下图:



1.6.2 刀具 T (车削)

功能

通过编程 T 字可以进行换刀。此时无论是有关**换刀**或者只是有关**预选**，都在机床数据中定义：

- 使用 T 字直接换刀（刀具调用）（例如：通常用于车床上的刀具转塔）

或

- 通过辅助指令 **M6** 在用 T 字预选后换刀。

说明：

如果激活了某个刀具，则在程序结束和关闭/打开操作系统后，仍将其作为生效刀具保存。

如果手动换刀，则在控制系统中输入换刀数据，从而可以使控制系统识别出正确的刀具。比如可以在 **MDA** 运行方式下启动一个带有新 T 字的程序段。

编程示例

无 M6 换刀

N10 T1

N20 T3

N30 T2

N40 T6

N50 T7

N60 T5

N70 T588

N80 M30

在控制系统中最多可存储 64 个刀具。

1.6.3 刀具补偿号 D (车削)

功能

可以向某个特定刀具分配带不同刀具补偿程序段（用于多个刀沿）的 1 到 9 个数组。如果需要使用特殊刀沿，可通过 D 和相应的编号进行编程
如果没有写入 D 字，则 D1 自动生效。
如果编程 D0，则刀具补偿无效。

编程

D... ; 刀补号： 1 ... 9, D0: 没有补偿值生效!

控制系统中最多可以同时存储刀具补偿程序段的 64 个数据字段 (D 编号) :

T1	D1	D2	D3		D9
T2	D1				
T3	D1				
T6	D1	D2	D3		
T8	D1	D2			

每把刀具都有其补偿程序段，最多为 9 个

说明

一旦刀具有效，**刀具长度补偿立即生效**；如果没有编写任何 D 号，则 D1 自动生效。最先编程的相关长度补偿轴运行时，补偿开始。

此外**刀具半径补偿**必须另外通过 G41/G42 激活。

编程示例

换刀:

```

N10 T1      ; 激活刀具 1 和相应的 D1
N20 G0 X100 ; 覆盖长度补偿差值
N30 Z100
N40 T4 D2   ; 换入刀具 4, T4 的 D2 生效
N50 X50 Z50
N60 G0 Z62
N70 D1     ; 刀具 4 的 D1 生效, 只更换刀沿
N80 M30

```

1.6 刀具和刀具补偿

补偿存储器的内容

- 几何尺寸：长度，半径

尺寸包含多个分量（几何量，磨损量）。控制系统会对分量进行计算，再得出总尺寸（比如总长度 1，总半径）。各个总尺寸在激活补偿存储器时生效。

如何计算出坐标轴中的值，由刀具类型和当前平面 G17, G18, G19（参见下图）来决定。

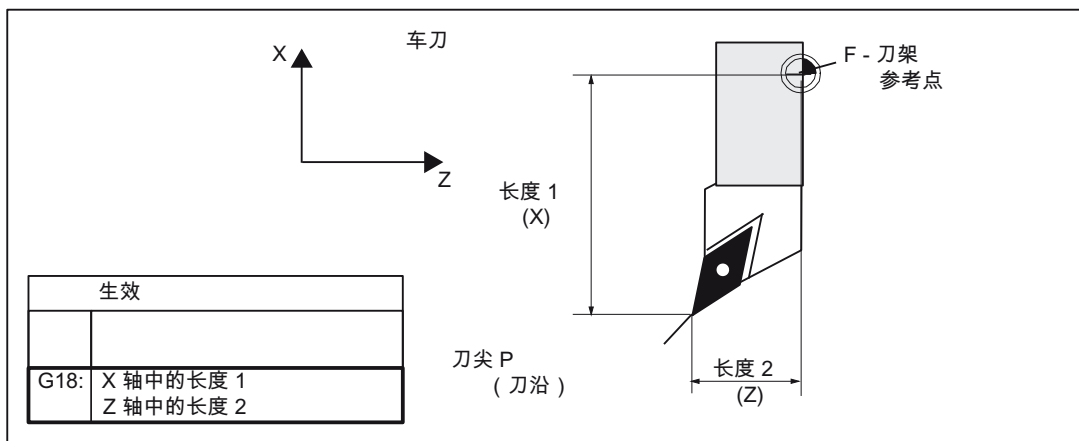
- 刀具类型

刀具类型（钻削或车削刀具）确定所需的几何数据以及如何计算这些数据。

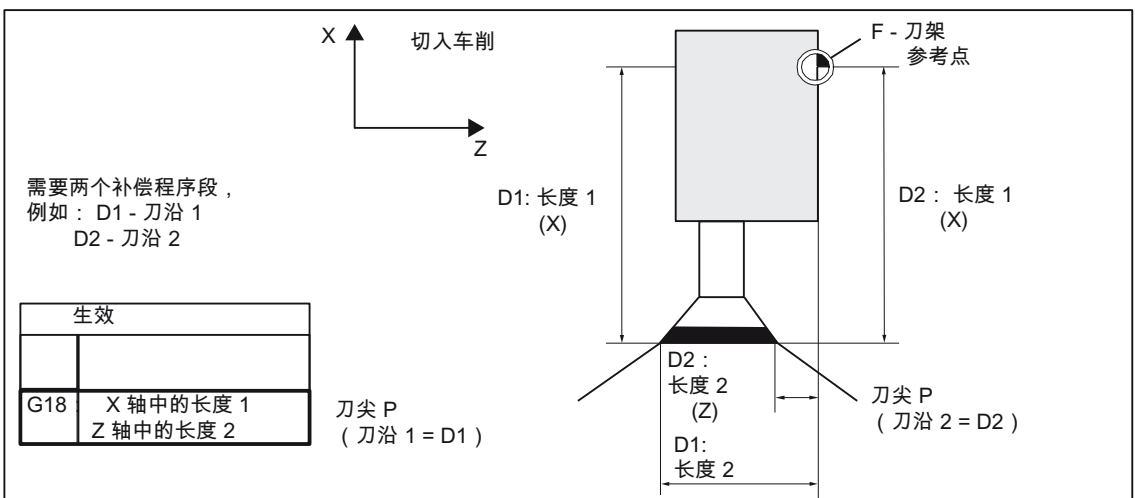
- 刀沿位置

对于“车刀”，还需另外说明刀沿位置。

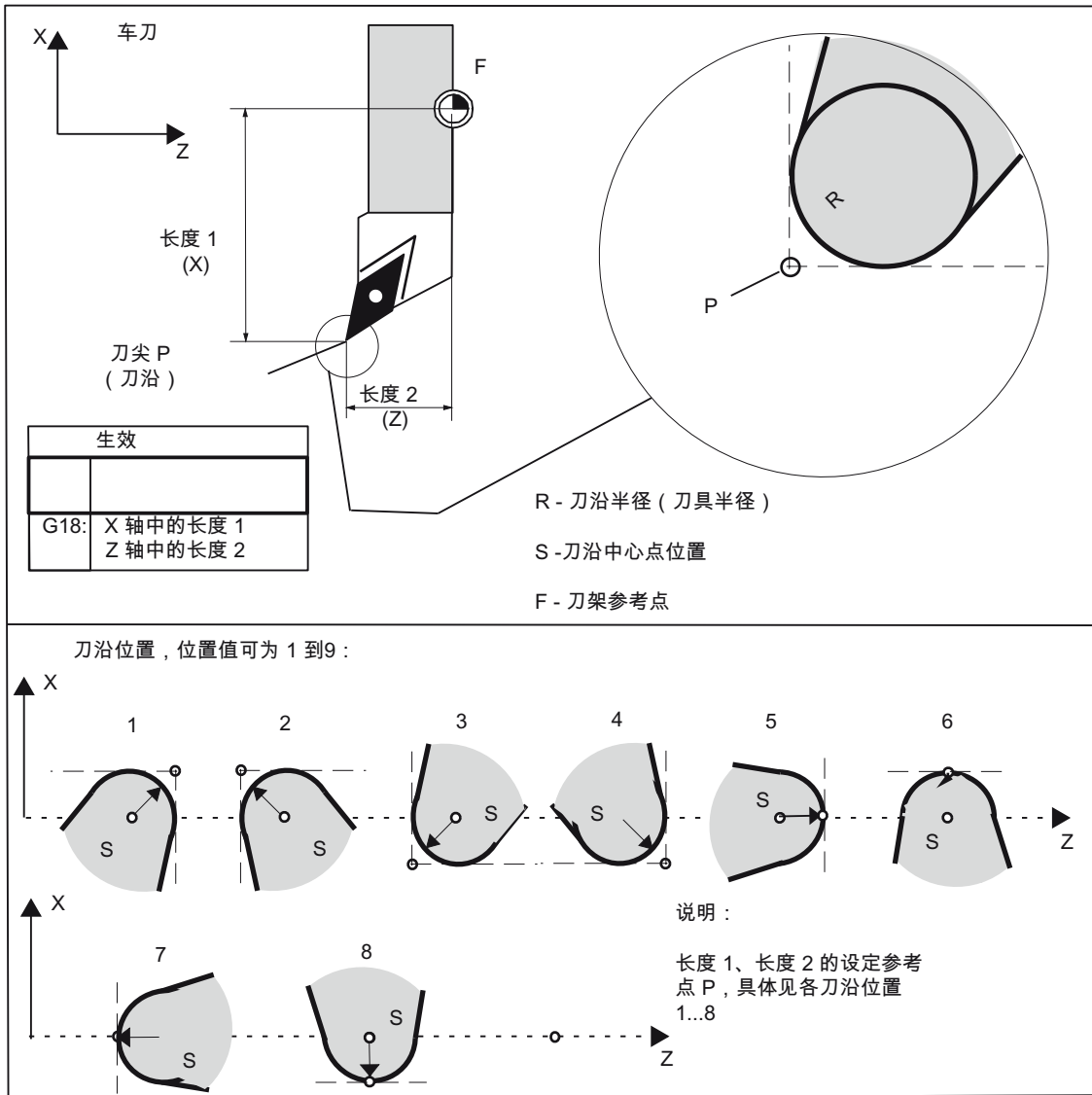
下图给出了各个刀具类型所需的刀具参数的信息。



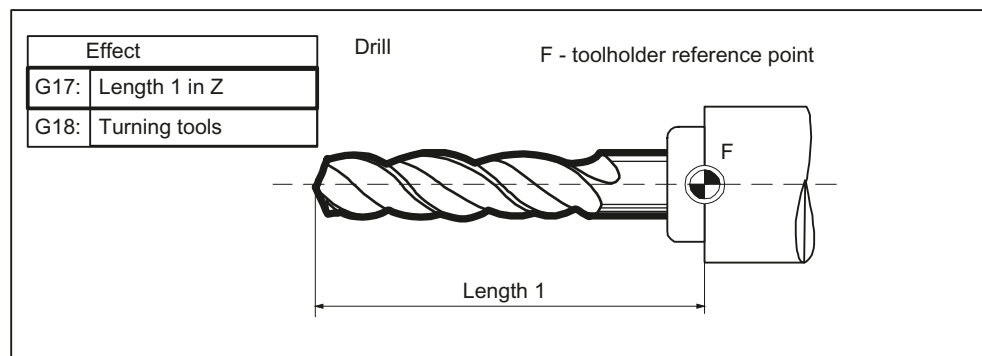
有关带有双刀沿 D1 和 D2 长度补偿的车刀的描述，见下图：



有关车刀长度补偿值的描述，见下图：



有关钻头的补偿作用的描述，见下图：



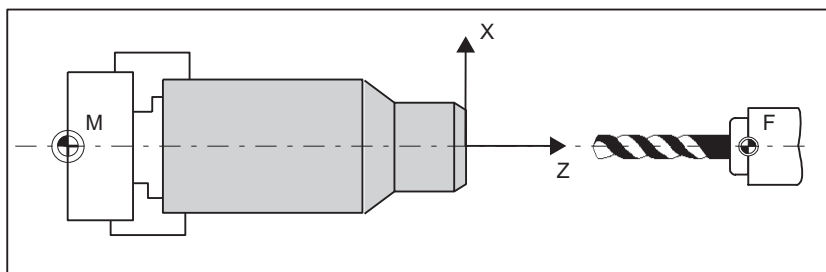
定中心钻

使用定中心钻时切换到 **G17**。则 **Z** 轴上钻头的长度补偿有效。钻孔后，用 **G18** 返回到车刀的普通补偿。

编程示例

```

N10 T3 D1                ;钻头
N20 G17 G1 F1 Z0 M3 S100 ; Z 轴上生效的长度补偿
N30 Z-15
N40 G18 M30              ; 钻削结束
    
```

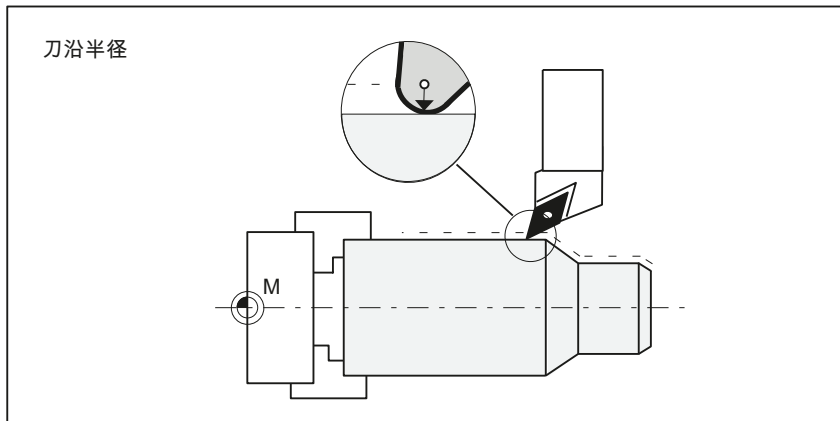


1.6.4 选择刀具半径补偿：G41,G42

功能

必须存在具有相应 **D** 号的生效刀具。通过 **G41/G42** 使刀具半径补偿（刀沿半径补偿）生效。然后，数控系统自动计算出当前刀具半径所需、与编程轮廓等距的刀具轨迹。**G18** 必须处于有效状态。

有关刀具半径补偿（刀沿半径补偿）的描述，见下图：



编程

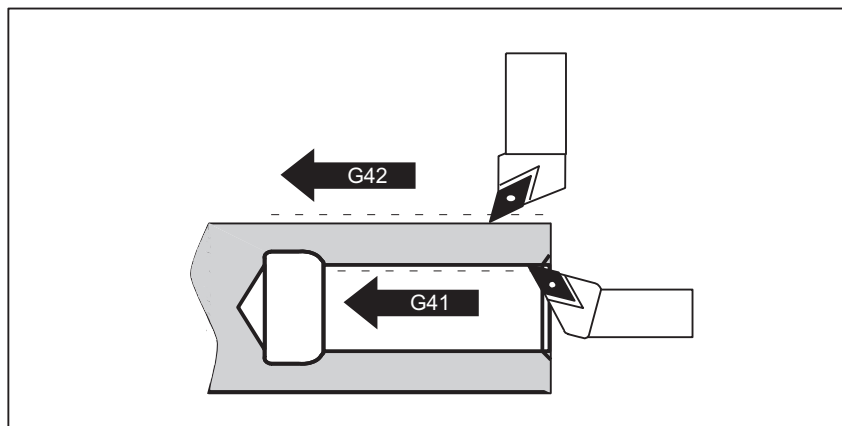
G41 X... Z... ; 刀具半径补偿, 轮廓左边

G42 X... Z... ; 刀具半径补偿, 轮廓右边

备注: 只有在直线插补 (G0, G1) 情况下才可以选择半径补偿。

编程两个坐标轴。如果你只给出一个坐标轴的尺寸, 则第二个坐标轴自动地以此前最后编程的尺寸赋值。

有关工件轮廓左边 - 右边补偿的描述, 见下图:

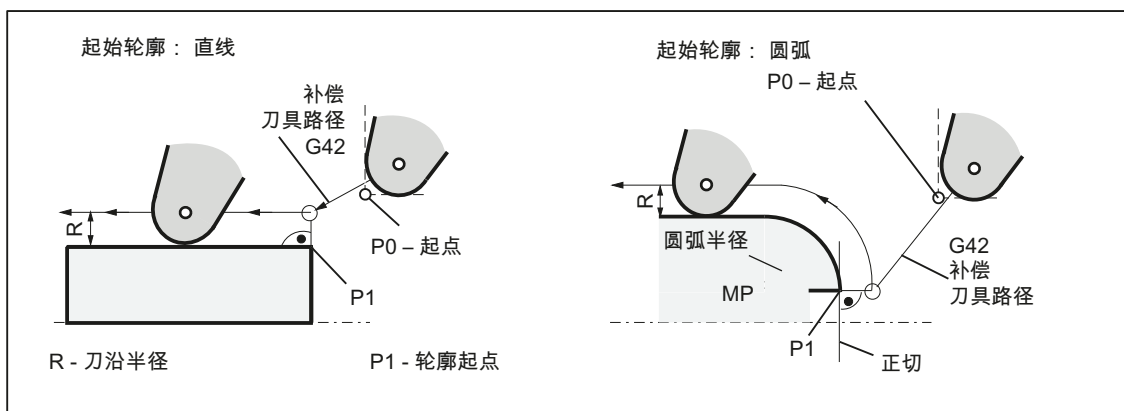


开始进行补偿

刀具以直线返回轮廓, 然后在轮廓起始点与轨迹切线垂直。

请选择合适的起点, 确保刀具运行过程中不发生碰撞!

参见以 **G42** 进行刀具半径补偿, 刀沿位置 = 3 的示例:



使用 **G41** 使得在刀具顺时针运行时刀尖环绕工件左侧; 使用 **G42** 使得在刀具逆时针运行时刀尖环绕工件右侧。

说明

通常在 G41/G42 程序段后接着执行加工工件轮廓的第一个程序段。然而，两程序段间、不含任何位移数据的某个程序段会中断轮廓描述，例如：程序段中只有 M 指令。

编程示例

```

N10 T4 D1 M3 S1000 F0.15
N20 G0 X0 Z0           : P0 - 起点
N30 G1 G42 X50 Z50     : 工件轮廓右边补偿, P1
N40 X0 Z0 G40 G1       : 起始轮廓, 圆弧或直线
N50 M30

```

1.6.5 拐角特性: G450, G451

功能

在 G41/G42 有效的情况下，一段轮廓到另一段轮廓以不连续的拐角过渡时可以通过 G450 和 G451 功能调节其特性（拐角特性）。

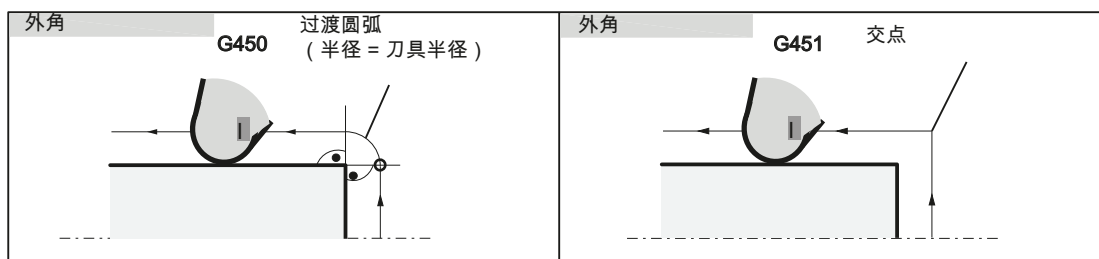
由控制系统自动识别内角和外角。如为内角，则必须要回到等距轨迹的交点。

编程

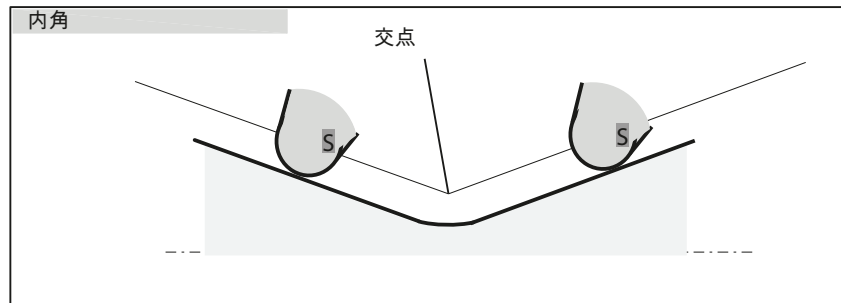
G450 ; 过渡圆弧

G451 ; 交点

有关外角拐角特性的描述，见下图：



有关内角拐角特性的描述，见下图：



过渡圆弧 G450

刀具中心点以圆弧形状绕行工件外拐角，刀具半径为离开距离。在数据计算中，圆弧过渡属于下一个带有运行指令的程序段；比如有关进给值。

交点 G451

在刀具中心轨迹（圆弧或直线）形成等距交点 G451 时返回该点（交点）。

1.6.6 取消刀具半径补偿：G40

功能

用 G40 取消补偿运行（G41/G42）。G40 也是编程开始时所处的状态。刀具在 G40 之前的程序段以正常方式结束（结束时补偿矢量垂直于轨迹终点处切线）；与起始角无关。

G40 生效时，参考点即为刀尖。这样在取消补偿时，刀尖返回编程点。

在选择 G40 程序段编程终点时要始终确保运行中不会发生碰撞！

编程

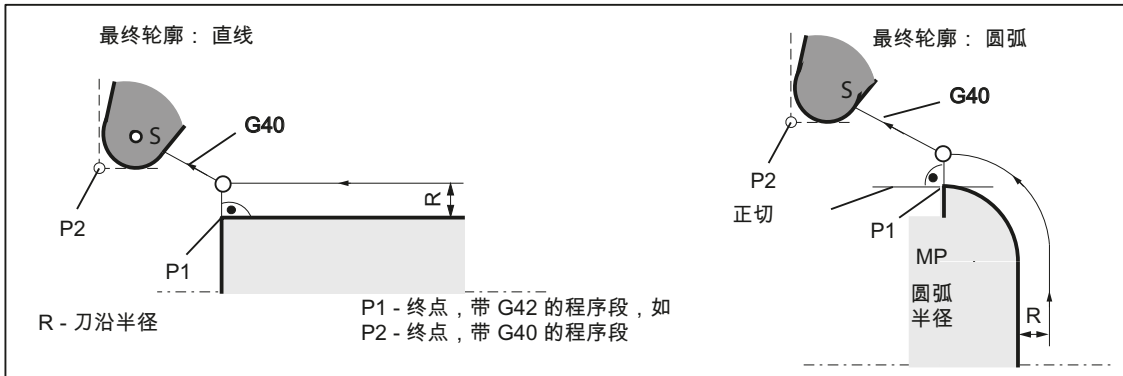
G40 X... Z... ; 取消刀具半径补偿

备注：只有在直线插补（G0，G1）情况下才可以取消补偿运行。

编程两个坐标轴。如果你只给出一个坐标轴的尺寸，则第二个坐标轴自动地以此前最后编程的尺寸赋值。

1.6 刀具和刀具补偿

有关用 G40 取消刀具半径补偿的描述，见下图：



编程示例

```

N10 T4 D1 M3 S1000 F0.1
N20 G0 X50 Z50
N30 G1 G42 X30 Z40
N40 G2 X20 Z20 R15
N50 G1 X10 Z10
N60 G40 G1 X0 Z0 ; 轮廓，圆弧或直线上的最后程序段，P1
N70 M30 ; 关闭刀具半径补偿，P2
    
```

1.6.7 刀具半径补偿的特殊情况

补偿方向的转换

补偿方向 G41 ⇌ G42 可以互相转换，无需在其中写入 G40 指令。
 原补偿方向的最后程序段在其轨迹终点处按补偿矢量的正常状态结束。然后按新的补偿方向开始进行补偿（在起点处以正常状态）。

重复 G41，G41 或者 G42，G42

重复执行相同的补偿方式时可以直接进行新的编程而无需在其中写入 G40 指令。
 新补偿调用之前的最后程序段在其轨迹终点处以补偿矢量的正常状态结束。然后开始进行新的补偿（特性与补偿方向的转换一样）。

补偿号 D 的更换

补偿号 D 可以在补偿运行时更换。刀具半径改变后，自新 D 号所在的程序段开始处生效。但整个变化需等到程序段结束才能完成。这些修改值由整个程序段连续执行；在圆弧插补时也一样。

通过 M2 结束补偿

如果通过 M2（程序结束），而不是用 G40 指令结束补偿运行，则最后的程序段以补偿矢量正常位置的坐标结束。这时**不会**出现补偿动作。程序在此刀具位置结束。

临界加工情况

在编程时特别要注意下列情况:内角过渡时轮廓位移小于刀具半径；在两个相连内角处轮廓位移小于刀具直径。

避免出现这种情况！

检查多个程序段，使轮廓中不要含有“瓶颈”。

如果进行测试/试运行，请选用可供选择的最大刀具半径。

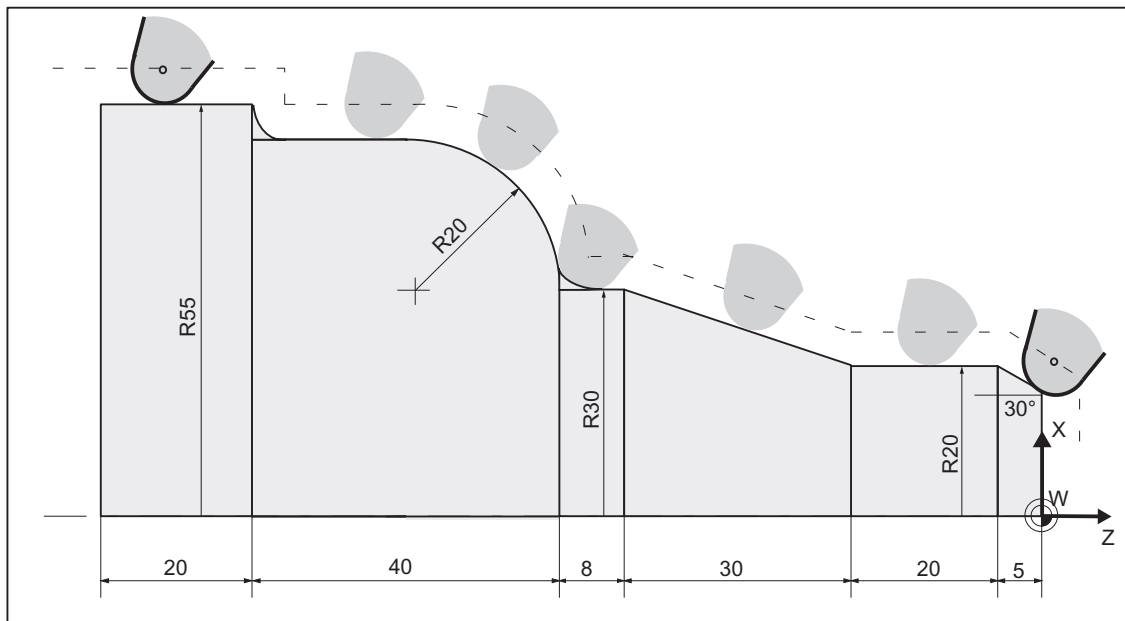
轮廓尖角

如果在 G451 交点有效时出现尖角，则会自动转换到过渡圆弧。这可以避免较长的空行程。

1.6 刀具和刀具补偿

1.6.8 刀具半径补偿举例（车床）

参见以下刀具半径补偿、刀沿半径放大显示的示例：



编程示例

```

N1                                     ; 轮廓切削
N2 T1                                  ; 刀具 1, 补偿号 D1
N10 DIAMOF F0.15 S1000 M3             ; 半径尺寸, 工艺值
N15 G54 G0 G90 X100 Z15
N20 X0 Z6
N30 G1 G42 G451 X0 Z0                 ; 开始补偿模式
N40 G91 X20 CHF=(5* 1.1223)          ; 插入倒角, 30 度
N50 Z-25
N60 X10 Z-30
N70 Z-8
N80 G3 X20 Z-20 CR=20
N90 G1 Z-20
N95 X5
N100 Z-25
N110 G40 G0 G90 X100                 ; 结束补偿模式
N120 M2
    
```


1.6.9 刀具补偿的特殊情况（车床）

设定数据的影响

操作者/编程者使用下列设定数据，会影响到如何使用刀具的**长度补偿值**：

- SD 42940: TOOL_LENGTH_CONST
(将刀具长度分量分配到几何轴)
- SD 42950: TOOL_LENGTH_TYPE
(刀具长度分量的分配与刀具类型无关)

说明

修改的设定数据将在选择新刀沿时生效。

举例

通过 SD 42950: TOOL_LENGTH_TYPE =2
在长度补偿中将所使用的铣刀当作车刀进行计算：

- G17: 长度 1 位于 Y 轴，长度 2 位于 X 轴
- G18: 长度 1 位于 X 轴，长度 2 位于 Z 轴
- G19: 长度 1 位于 Z 轴，长度 2 位于 Y 轴

通过 SD 42940: TOOL_LENGTH_CONST =18
G17 到 G19 中的长度分配和 G18 的相同：

- 长度 1 位于 X 轴，长度 2 位于 Z 轴

程序中的设定数据

除了操作时定义设定数据，也可以在程序中设定。

编程示例

```
N10 $MC_TOOL_LENGTH_TYPE=2  
N20 $MC_TOOL_LENGTH_CONST=18
```

1.7 辅助功能 M

功能

利用辅助功能 M 可以设定诸如开关操作、“冷却液 ON/OFF”等功能。

一小部分的 M 功能已经由控制系统制造商预置，作为固定功能占用。其它功能供机床生产厂商使用。

编程

M... ; 在一个程序段中最多可以有 5 个 M 功能

生效

在坐标轴运行程序段中的作用：

如果 M0, M1, M2 功能位于一个有坐标轴运行指令的程序段中，则这些 M 功能只有在坐标轴运行之后才会有效。

而 M3, M4, M5 功能则在坐标轴运行之前信号就输出到内部的匹配控制系统 (PLC) 上。只有当受控主轴按 M3 或 M4 启动之后，坐标轴才开始运行。在执行 M5 指令时并不等待主轴停止。坐标轴在主轴静止之前已经开始运动 (标准设置)。

其它的 M 功能信号与坐标轴运行信号一起输出到 PLC 上。

如果您想在坐标轴运行之前或之后对一个 M 功能进行编程，则你须插入一个独立的 M 功能程序段。

说明

M 功能会中断 G64 连续路径运行并产生准停：

编程示例

```
N10 S1000
N20 G1 X50 F0.1 M3 ; 程序段中 M 功能，有轴运动，在 X 轴运行之前主轴快速运行
N180 M78 M67 M10 M12 M37 ; 程序段中最多有 5 个 M 功能
M30
```

说明

除了 M 功能和 H 功能之外，T、D 和 S 功能也可以传送到 PLC (存储编程控制系统) 上。每个程序段中最多可以写入 10 个这样的功能指令。

1.8 H 功能

功能

使用 H 功能可从程序中向 PLC 传输浮点型数据（数据类型 REAL - 如使用计算参数时，参见章节“计算参数 R (页 99)”）。

某些 H 功能值的意义由机床制造商确定。

编程

H0=... 到 H9999=... ; 每个程序段最多 3 个 H 功能

编程示例

```
N10 H1=1.987 H2=978.123 H3=4 ; 程序段中有 3 个 H 功能  
N20 G0 X71.3 H99=-8978.234 ; 程序段中有轴运行指令  
N30 H5 ; 相当于 H0=5.0
```

说明

除了 M 功能和 H 功能之外，T、D 和 S 功能也可以传送到 PLC（存储编程控制系统）上。每个程序段中最多可以写入 10 个这样的功能指令。

1.9 计算参数 R, LUD 和 PLC 变量

1.9.1 计算参数 R

功能

如果一个 NC 程序不仅仅适用于一次性特定数值，或者必须要计算出数值，则可以使用计算参数。在程序运行时，可以通过控制系统计算或者设置所需要的数值。

另一个方法就是通过操作设定计算参数值。如果计算参数赋值，它们可以在程序中赋值其它数值可设定的 NC 地址。

编程

R0=... 到 R299=... ; 赋值计算参数

R[R0]=... ;间接编程 赋值计算参数 R, 例如将其编号赋在 R0 中

X=R0 ; 为 NC 地址赋值计算参数, 例如: X 轴

赋值

计算参数有以下的赋值范围:

$\pm(0.000\ 0001 \dots 9999\ 9999)$
(8 位小数, 带符号和小数点)。

在整数数值中小数点可以取消 正号可以不用写

示例:

R0=3.5678 R1=-37.3 R2=2 R3=-7 R4=-45678.123

使用**指数表示法**可以赋值更大的数值范围:

$\pm (10^{-300} \dots 10^{+300})$

指数数值写在 **EX** 符号之后; 最大的字符数: 10 (包括符号和小数点)

EX 的值范围: -300 至 +300

示例:

R0=-0.1EX-5	; 意义: R0 = -0.000 001
R1=1.874EX8	; 意义: R1 = 187 400 000

说明

一个程序段中可以有几个赋值指令; 也可以赋值计算表达式。

给其它地址赋值

一个 NC 程序的灵活性主要体现在: 可以把这些计算参数或者计算表达式用计算参数赋值给其它的 NC 地址。可以用数值、算术表达式或 R 参数对任意 NC 地址赋值; **例外: 地址 N、G 和 L。**

在赋值时, 在地址符之后写符号“=”。也可以带一个负号赋值。

如果给一个轴地址赋值 (运行指令), 则需要一个独立的程序段。

示例:

```
N10 G0 X=R2 ; 赋值 X 轴
```

计算操作/计算功能

在使用运算符/计算功能时，必须要遵守通常的数学运算规则。优先执行的过程通过圆括号设置。其它情况下，按照先乘除后加减运算。

在三角函数中单位使用度。

编程示例：使用 R 参数计算

```
N10 R1= R1+1 ; 新的 R1 等于旧的 R1 加 1
N20 R1=R2+R3 R4=R5-R6 R7=R8*R9 R10=R11/R12
N30 R13=SIN(25.3) ; R13 等同于正弦 25.3 度
N40 R14=R1*R2+R3 ; 先乘除后加减 R14=(R1*R2)+R3
N50 R14=R3+R2*R1 ; 结果，与程序段 N40 相同
N60 R15=SQRT(R1*R1+R2*R2) ; 意义:
N70 R1= -R1 ; 新的 R1 为原先 R1 的负值
```

编程示例：用 R 参数为坐标轴赋值

```
N10 G1 G91 G94 X=R1 Z=R2 F300 ; 单独程序段（运行程序段）
N20 Z=R3
N30 X=-R4
N40 Z= SIN(25.3)-R5 ; 带算术运算
M30
```

编程示例：间接编程

```
N10 R1=5 ; 直接赋值 5（整数）给 R1
R2=6
R1=R2-1
N100 R[R1]=27.123 ; 间接赋值 27.123 给 R5
M30
```

1.9.2 局部用户数据 (LUD)

功能

用户/编程人员（使用者）可以在程序中定义自己的不同数据类型的变量 (LUD= Local User Data 局部用户数据)。这些变量只在定义它们的程序中出现。可以在程序的开头直接定义这些变量并同时为它们赋值。否则初始值为零。

变量名可由编程器自行确定。命名时应遵守以下规则：

- 最大长度为 32 个字符
- 起始的两个字符必须是字母；其它的字符可以是字母，下划线或数字。
- 系统中已经使用的名字不能再使用(NC 地址，关键字，程序名，子程序名等)。

编程/数据类型

DEF BOOL 变量名 1	; 布尔类型，值： TRUE 真(=1), FALSE 假(=0)
DEF CHAR 变量名 2	; 字符型， ASCII 代码中的 1 个字符： "a", "b", ... ; 代码值： 0 ... 255
DEF INT 变量名 3	; 整型， 32 位范围内的整数值： ; -2 147 483 648 至 +2 147 483 647 （十进制）
DEF REAL 变量名 4	; 实型，自然数（比如计算参数 R）， ; 值范围： ±(0.000 0001 ... 9999 9999) ; （8 位数字，加符号和小数点）或 ; 指数书写方式： ±（10 的 -300 次方到 10 的 +300 次方）
DEF STRING[字符串长度] 变量名 41	; 字符串型， [字符串长度]： 最大字符数

每种数据类型要求单独的程序行。但可以在同一行中定义类型相同的多个变量。

示例：

```
DEF INT PVAR1, PVAR2, PVAR3=12, PVAR4 ; 整型的变量 4
```

赋值字符串类型举例：

```
DEF STRING[12] PVAR="Hello" ; 定义变量 PVAR 的最大字符长度为 12 个字符，并赋值字符串 "Hello"
```

字段

除了单个变量，还可以定义这些数据类型变量的一维或者二维域：

```
DEF INT PVAR5[n]           ; 整型的一维域，n: 整数
DEF INT PVAR6[n,m]        ; 整型的二维域，n, m: 整数
```

示例：

```
DEF INT PVAR7[3]           ; 域中包含 3 个整型元素
```

在程序中可以通过域索引读取各个域元素、并将其作为单独的变量来处理。索引顺序从 0 到较小的元素数量。

示例：

```
N10 PVAR7[2]=24           ; 第三个域元素（使用索引 2）的值为 24。
```

包含 SET 指令的域赋值：

```
N20 PVAR5[2]=SET(1,2,3)   ; 从第 3 个域元素起，分配不同的值。
```

包含 REP 指令的域赋值：

```
N20 PVAR7[4]=REP(2)       ; 从域元素 [4] 起，所有的元素具有相同的值，此处为 2。
```

1.9.3 PLC 变量的读和写

功能

为了在 NC 和 PLC 之间进行快速的数据交换，在 PLC 用户接口提供了一个长度为 512 字节的特殊数据区。在此区域中，PLC 数据具有相同的数据类型和位置偏移量。在 NC 程序中可以读写这些一致的变量。

为此，需提供专门的系统变量：

```
$A_DBB[n]           ; 数据字节（8 位值）
$A_DBW[n]           ; 数据字（16 位值）
$A_DBD[n]           ; 数据双字（32 位值）
$A_DBR[n]           ; 实型数据（32 位值）
```

n 表示位置偏移量（从数据区的起始处到变量的起始处），单位字节

1.10 程序跳转

编程示例

R1=\$A_DBR[5] ; 读取 REAL 值; 偏移量 5 (从区域的字节 5 处开始)

说明

读取变量会造成预处理程序停止 (内部 STOPRE)。

说明

PLC 变量的写入一般限制在三个变量 (元素) 范围内。

对于相继迅速写入的 PLC 变量, 每次写入过程需要一个元素。

如果需要执行多次写入而提供元素, 则必须确保程序段传送 (必要时触发预处理停止)。

示例:

```
$A_DBB[1]=1 $A_DBB[2]=2 $A_DBB[3]=3  
STOPRE  
$A_DBB[4]=4
```

1.10 程序跳转

1.10.1 绝对程序跳转

功能

NC 程序在运行时按写入时的顺序执行程序段。

程序在运行时可以通过插入程序跳转指令改变执行顺序。

跳转目标只能是有**标记符**或一个**程序段号**的程序段。该程序段必须在此程序之内。

绝对跳转指令必须占用一个独立的程序段。

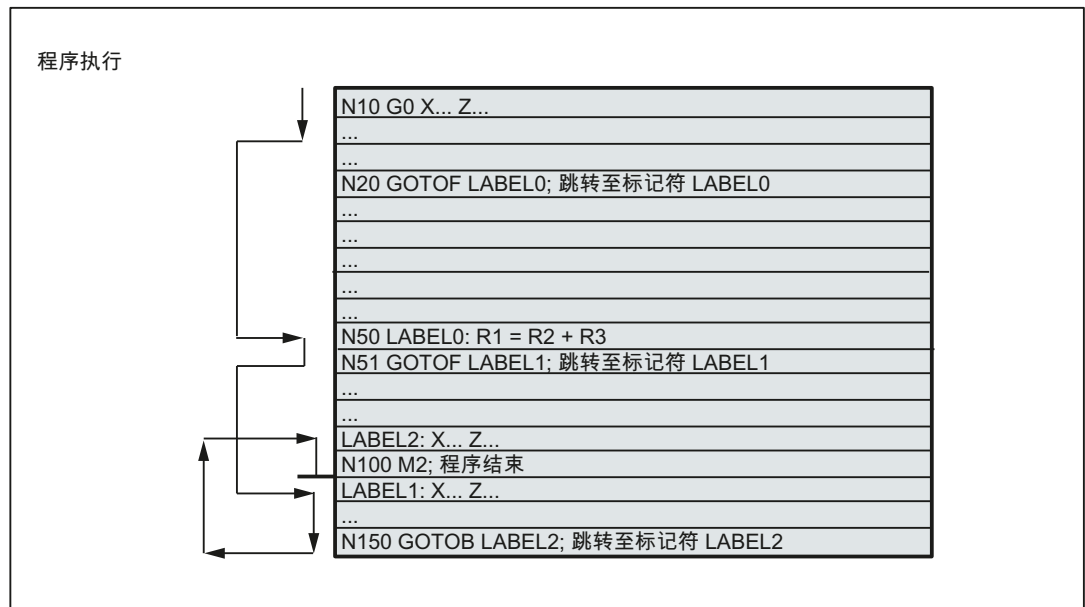
编程

GOTOF Label ; 向前跳转 (向程序结束的方向)

GOTOB Label ; 向后跳转 (向程序开始的方向)

标记符 ; 所选择标记符的字符顺序(跳转标记)或程序段号

参见以下绝对跳转的示例：



1.10.2 有条件程序跳转

功能

跳转条件在 **IF 指令**之后产生。如果满足跳转条件（**值不为零**），则会进行跳转。跳转目标只能是有**标记符**或一个**程序段号**的程序段。该程序段必须在此程序之内。条件跳转指令必须占用一个独立的程序段。数个条件跳转指令可位于同一程序段。如果必要，通过条件程序跳转还可以大幅缩减程序。

编程

IF 条件 GOTOF 标签	； 向前跳转
IF 条件 GOTOB 标签	； 向后跳转
GOTOF	； 向前跳转（向程序结束的方向）
GOTOB	； 向后跳转（向程序开始的方向）
标记符	； 所选择标记符的字符顺序(跳转标记)或程序段号
IF	； 跳转条件简介
Condition	； 计算参数、条件生成计算表达式

1.10 程序跳转

比较运算

运算符	含义
==	等于
<>	不等
>	大于
<	小于
>=	大于等于
<=	小于等于

比较运算可以生成跳转条件。计算表达式同样可以比较。

比较运算结果为“满足”或“不满足”。不满足”的值为零。

程序运算的编程示例

```

R1>1           ; R1 大于 1
1 < R1         ; 1 小于 R1
R1<R2+R3       ; R1 小于 R2 与 R3 之和
R6>=SIN( R7*R7) ; R6 大于等于 SIN (R7)的平方
    
```

编程示例

```

N10 IF R1 GOTOF LABEL1           ; 如果 R1 不为空, 则跳转至 LABEL1 的程序段
G0 X30 Z30
N90 LABEL1: G0 X50 Z50
N100 IF R1>1 GOTOF LABEL2       ; 如果 R1 大于 1, 则跳转至 LABEL2 的程序段
G0 X40 Z40
N150 LABEL2: G0 X60 Z60
G0 X70 Z70
N800 LABEL3: G0 X80 Z80
G0 X100 Z100
N1000 IF R45==R7+1 GOTOB LABEL3 ; 如果 R45 大于 R7 与 1 之和, 则跳转至 LABEL3 的
                                程序段

M30
程序段中的数个条件跳转:
N10 MA1: G0 X20 Z20
N20 G0 X0 Z0
N30 IF R1==1 GOTOB MA1 IF R1==2 GOTOF MA2 ...
N40 G0 X10 Z10
N50 MA2: G0 X50 Z50
N60 M30
    
```

说明

第一次满足条件时执行跳转。

1.10.3 程序跳转举例**任务**

圆弧上点的移动：

已知：

起始角度： 30° ，R1

圆弧半径：32 mm，R2

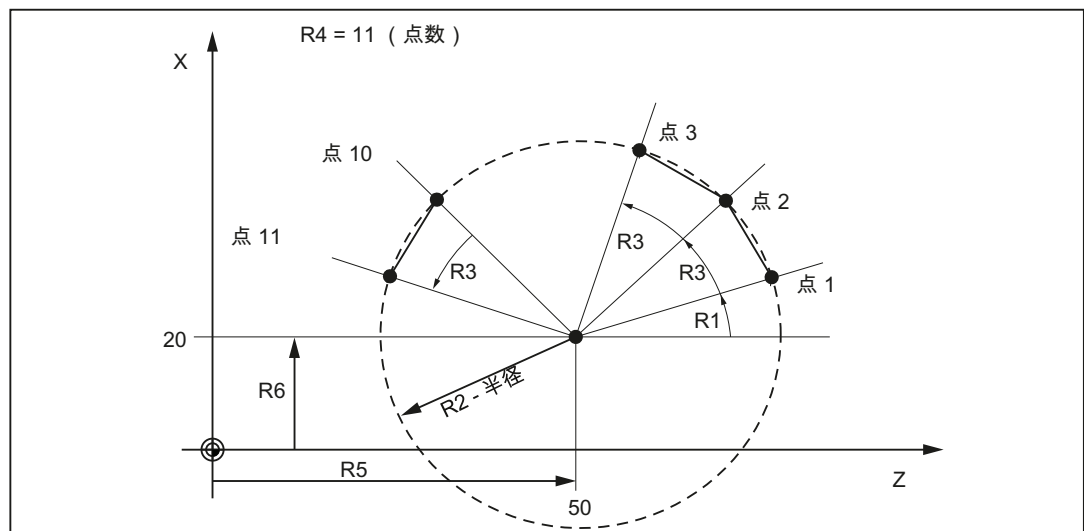
位置间距： 10° ，R3

点数：11，R4

Z 轴上的圆心位置：50 mm，R5

X 轴上的圆心位置：20 mm，R6

有关在圆弧上线性运行至各点的描述，见下图：



编程示例

```

N10 R1=30 R2=32 R3=10 R4=11 R5=50 R6=20      ; 初始值分配
N20 MA1: G0 Z=R2*COS (R1)+R5                ; 计算以及轴地址赋值
X=R2*SIN(R1)+R6
N30 R1=R1+R3 R4= R4-1
N40 IF R4 > 0 GOTOB MA1
N50 M2

```

说明

在程序段 N10 中为相应的计算参数赋值。在 N20 中计算坐标轴 X 和 Z 的数值计算并进行赋值处理。

在程序段 N30 中，R1 增加 R3 间距角，R4 较少数值 1。

如果 $R4 > 0$ ，重新执行 N20，否则运行 N50，程序结束。

1.10.4 程序跳转的跳转目标

功能

标记符或程序段号用于标记程序中所跳转的目标程序段。用跳转功能可以实现程序运行分支。

标记符可以自由选择，但必须由 2-8 个字母或数字组成，其中开始两个符号必须是字母或下划线。

跳转目标程序段中标记符后面必须以**冒号结束**。标记符始终位于程序段段首。如果程序段有段号，则标记符**紧跟着段号**。

在一个程序中，各标记符必须具有唯一的含义。

编程示例

```

N10 LABEL1: G1 X20                          ; LABEL1 为标记符，跳转目标
N20 G0 X10 Z10
TR789: G0 X10 Z20                           ; TR789 为标记符，跳转目标
G0 X30 Z30                                  - 无段号
N100 G0 X40 Z40                             ; 程序段号可以是跳转目标
M30

```

1.11 子程序

1.11.1 概述

使用

从原则上讲主程序和子程序之间并没有区别。

用子程序编写经常重复进行的加工，比如某一确定的轮廓形状。在主程序中，可以在需要的位置调用并运行子程序。

子程序的一种形式就是 **加工循环**。加工循环包含一般通用的加工工序。通过给规定的计算参数赋值就可以实现各种具体的加工。

结构

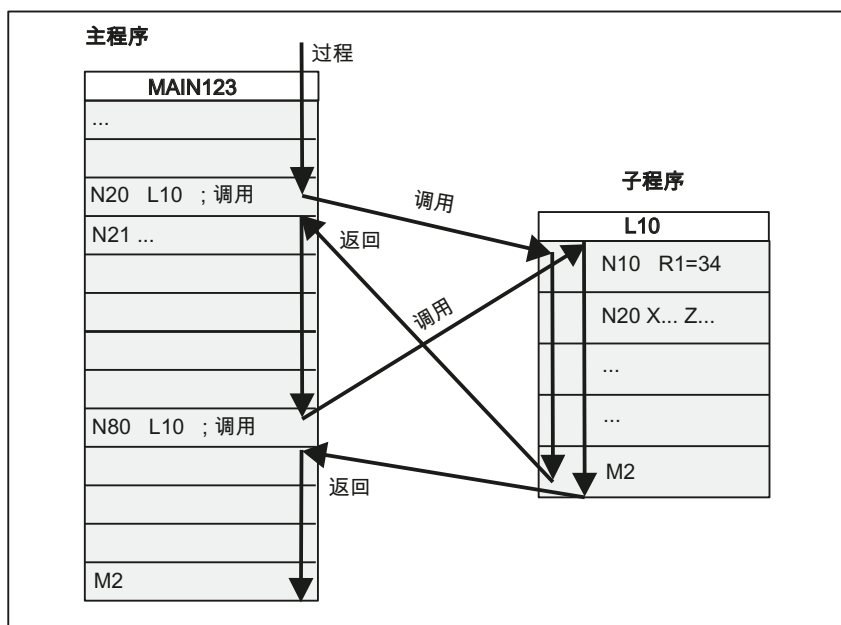
子程序的结构与主程序的结构一样（参见章节“程序结构 (页 7)”）。在子程序中与主程序一样，也是在最后一个程序段中使用 **M2（程序结束）** 结束运行。这就表示返回到所调用的程序界面。

程序结束

除了用 **M2** 指令外，还可以用 **RET** 指令结束子程序。

如果一个 **G64** 轨迹控制运行不要由于返回而中断，则需要使用 **RET** 指令。用 **M2** 指令则会中断 **G64** 运行方式并造成准停。

参见以下两次调用子程序的示例：



子程序名称

为了能够从众多的子程序中挑选出一个确定的子程序，则子程序必须要有自己的名称。在编制程序时可以自由选择名称，但是必须符合规定。

适用主程序命名的同样规则。

示例：**BUCHSE7**

另外，在子程序中还可以使用地址字 **L... L...**。其值可以是 7 位数（仅为整数）。

注意：地址 **L** 中，数字前的零有意义，用于区别。

示例：**L128** 不是 **L0128** 或 **L00128** ！

以上表示 3 个不同的子程序。

说明：子程序名称 **LL6** 预留给刀具更换！

子程序调用

在一个程序中（主程序或子程序）可以直接用程序名调用子程序。为此需要使用一个独立的程序段。

示例：

```

N10 L785           ; 调用子程序 L785
N20 WELLE7        ; 调用子程序 WELLE7
    
```

程序重复 P...

如果要求多次连续地执行某一子程序，则在编程时必须是在所调用子程序的程序后地址 P 写入调用次数。最多可以运行 9999 次 (P1 ... P9999)。

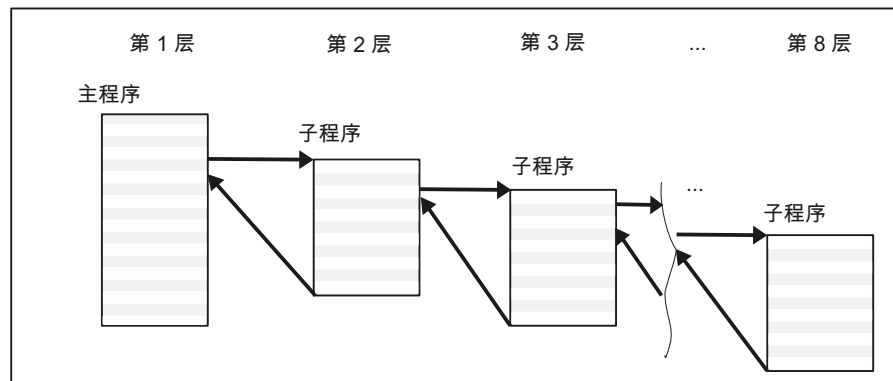
示例：

```
N10 L785 P3 ; 调用子程序 L785, 运行 3 次
```

嵌套深度

子程序不仅可以在一个主程序中调用，而且还可以在另一个子程序中调用。这样的嵌套调用总共有 8 个程序层可供使用；包括主程序层。

有关 8 个程序层的调用过程的描述，见下图：



说明

在子程序中可以改变模态有效的 G 功能，比如 G90 -> G91。在返回调用程序时请注意检查一下所有模态有效的功能指令，并按照要求进行调整。

对于 R 参数也同样需要注意，防止用上级程序界面中所使用的计算参数来修改下级程序界面的计算参数。

西门子循环进行工作时最多需要 7 个程序层。

1.11 子程序

1.11.2 调用加工循环（车床）

功能

循环是实现某个加工流程的工艺子程序。在解决具体问题时，通过改变参数/数值来直接调用各个循环。

编程示例

```
N10 DEF REAL RTP, RFP, SDIS, DP,
DTB
N20 G18 X100 Z100
N30 M3 S100 F0.1
N40 G17 X0
N50 CYCLE83(110, 90, 0, -80, 0, ; 调用循环 83: 直接传送数值,
-10, 0, 0, 0, 0, 1, 0)      单独程序段
N60 G0 X100 Z100
N70 RTP=100 RFP= 95.5 SDIS=2.4, ; 设定循环 82 的传送参数
DP=-20, DTB=3
N80 CYCLE82(RTP, RFP,SDIS, DP, , ; 调用循环 82, 单独程序段
DTB)
N90 M30
```

1.11.3 执行外部子程序（EXTCALL）

功能

使用 EXTCALL 命令，可以载入或执行外部 USB 记忆棒上存储的程序。

机床数据

在使用 EXTCALL 指令时需考虑以下机床数据：

- MD10132 \$MN_MMC_CMD_TIMEOUT
零件程序中用于命令的监控时间
- MD18362 \$MN_MM_EXT_PROG_NUM
外部同时待执行程序级的数量

编程

```
EXTCALL("<路径\程序名>")
```

参数

EXTCALL ; 用于子程序调用的关键字
<路径\程序名> ; 字符串型常量/变量

示例:

```
EXTCALL("D:\EXTERNE_UP\RECHTECKTASCHE")
```

说明

外部子程序不允许包含任何跳转指令，例如：GOTOF、GOTOB、CASE、FOR、LOOP、WHILE 或者 REPEAT。

可以有 IF-ELSE-ENDIF 结构。

可以进行子程序调用和嵌套的 EXTCALL 调用。

RESET, POWER ON

通过复位和 POWER ON（上电），可以中断外部的子程序调用，并且清除各自的后装载存储器。

示例

外部用户 USB 记忆棒的处理

主程序"Main.mpf" 位于 NC 存储器中，并已选择执行该程序:

```
N010 PROC MAIN  
N020 G0 X0 Z0  
N030 EXTCALL ("N:\EXTERNE_UP\BOHRUNG ")  
N040 G0 X100 Z100  
N050 M30
```

需要载入的子程序“BOHRUNG.SPF”在 USB 记忆棒中。

```
N010 PROC BOHRUNG  
N020 G1 F1000 G94  
N030 X=10 Z=10
```

1.12 定时器和工件计数器

```
N040 G0 X50 Z50  
...  
...  
N999999 M17
```

1.12 定时器和工件计数器

1.12.1 运行时间定时器

功能

准备计时器作为系统变量 (**\$A...**)，可用于监控程序中的工艺流程或只在显示屏中监控。这些计时器为只读。存在始终有效的计时器。其他计时器可以通过机床数据停用。

计时器 - 始终有效

- **\$AN_SETUP_TIME**

自最后一次以默认值控制上电起的时间（以分钟为单位）

每次系统以缺省值上电时，计时器自动归零。

- **\$AN_POWERON_TIME**

自最后一次控制上电起的时间（以分钟为单位）

控制系统每次上电后计时器自动归零。

可以停用的计时器

通过机床数据（默认设置）激活以下计时器。

启动为计时器特定启动。在停止的程序状态下或针对“进给倍率零”而自动中断每次的有效运行时间测量。

可以使用机床数据指定有效空运行进给率和程序测试的激活计时器行为。

- **\$AC_OPERATING_TIME**

"AUTO"模式中 NC 程序的总执行时间（以秒为单位）

在"AUTO"模式下，从程序启动到程序结束之间所有程序的运行时间累计值。系统每次上电后计时器自动设为零。

- **\$AC_CYCLE_TIME**

选择的 NC 程序运行时（以秒为单位）

计算所选 NC 程序在程序启动和程序结束之间的运行时间。当新的 NC 程序启动时，该定时器被删除。

- **\$AC_CUTTING_TIME**

刀具动作时间（以秒为单位）

快速进给无效而刀具有效时，程序启动和程序结束之间，在所有 NC 程序中测得的进给轴运行时间（默认设置）。

当暂停时间生效时，计算被中断。

控制系统每次上电后计时器自动设为零。

编程示例

```
N10 IF $AC_CUTTING_TIME >= R10 GOTOF WZZEIT           ; 刀具操作时间极限值?  
G0 X50 Z50  
N80 WZZEIT:G0 X60 Z60  
N90 MSG ("刀具动作时间: 达到极限值")  
N100 M0  
M30
```

显示

激活的系统变量内容通过以下键显示在窗口上：



窗口显示：

时间/计数器	
① 零件总数	0
② 需要的零件	0
③ 零件数	0
④ 总运行时间	0000 H 00 M 00 S
⑤ 程序运行时间	0000 H 00 M 00 S
⑥ 进给时间	0000 H 00 M 00 S
⑦ 冷启动后的时间	0019 H 22 M
⑧ 热启动后的时间	0000 H 48 M

- | | |
|-------------------------|-----------------------|
| ① = \$AC_TOTAL_PARTS | ⑤ = \$AC_CYCLE_TIME |
| ② = \$AC_REQUIRED_PARTS | ⑥ = \$AC_CUTTING_TIME |
| ③ = \$AC_ACTUAL_PARTS | ⑦ = \$AN_SETUP_TIME |
| \$AC_SPECIAL_PARTS 不显示。 | |
| ④ = \$AC_OPERATING_TIME | ⑧ = \$AN_POWERON_TIME |

也可通过以下操作区域查看有关运行时间定时器的信息：



参见

工件计数器 (页 117)

1.12.2 工件计数器

功能

“工件计数器”功能提供了可用于计算工件数量的计数器。

该计数器作为系统变量，可以通过程序或操作（注意写保护级！）进行读写存取。

通过机床数据可以对计数器激活、归零时刻和计数算法产生影响。

计数器

- **\$AC_REQUIRED_PARTS**

所需工件的个数（工件给定值）

在此计数器中可以定义工件的个数，在到达这个数值之后，实际工件的个数 **\$AC_ACTUAL_PARTS** 归零。

可以通过机床数据激活显示报警 21800“已达到工件额定值”。

- **\$AC_TOTAL_PARTS**

全部已生产工件的数量（总实际值）

计数器给出所有自开始时刻起所生产的工件数量。

当控制系统启动时，计数器自动复位至零。

- **\$AC_ACTUAL_PARTS**

当前工件的数量（当前实际值）

在这种计数器中记录自开始时刻起所生产的所有工件数量。当达到工作额定值时（**\$AC_REQUIRED_PARTS**，值大于零），计数器自动复位至零。

- **\$AC_SPECIAL_PARTS**

用户指定工件的数量

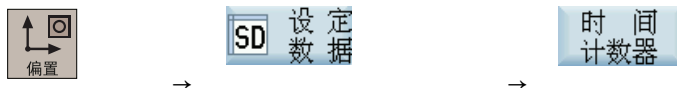
该计数器允许用户根据自定义来对工件计数。在与 **\$AC_REQUIRED_PARTS**（工件给定值）一致时可以定义一个报警输出。用户必须自行将该计数器归零。

编程示例

```
N10 IF $AC_TOTAL_PARTS==R15 GOTOF SIST           ; 达到工件数?
G0 X50 Z50
N80 SIST:G0 X60 Z60
N90 MSG ("达到额定工件数")
N100 M0
M30
```

显示

激活的系统变量内容通过以下键显示在窗口上：



窗口显示：

时间/计数器	
① 零件总数	0
② 需要的零件	0
③ 零件数	0
④ 总运行时间	0000 H 00 M 00 S
⑤ 程序运行时间	0000 H 00 M 00 S
⑥ 进给时间	0000 H 00 M 00 S
⑦ 冷启动后的时间	0019 H 22 M
⑧ 热启动后的时间	0000 H 48 M

- ① = \$AC_TOTAL_PARTS
- ② = \$AC_REQUIRED_PARTS
- ③ = \$AC_ACTUAL_PARTS
- ④ = \$AC_OPERATING_TIME
- ⑤ = \$AC_CYCLE_TIME
- ⑥ = \$AC_CUTTING_TIME
- ⑦ = \$AN_SETUP_TIME

\$AC_SPECIAL_PARTS 不显示。

- ⑧ = \$AN_POWERON_TIME

也可通过以下操作区域选择是否激活工件计数器功能：



循环

2.1 循环概述

循环是一种工艺子程序，借助这些循环可有效实行特定的加工过程，如螺纹攻丝。通过所提供的参数可以使循环和具体的加工要求相符。

钻削循环和车削循环

使用控制系统 SINUMERIK 808D 可以进行下列标准循环：

- 钻削循环

CYCLE81: 钻削，定中心

CYCLE82: 钻削，镗平面

CYCLE83: 深孔钻削

CYCLE84: 刚性攻丝

CYCLE840: 带补偿攻丝

CYCLE85: 铰孔 1

CYCLE86: 镗孔

CYCLE87: 带停止 1 的钻孔

CYCLE88: 带停止 2 的钻孔

CYCLE89: 铰孔 2

- 车削循环

CYCLE92: 切断

CYCLE93: 切槽

CYCLE94: 退刀槽（E 型和 F 型，符合 DIN）

CYCLE95: 毛坯切削，带底切

CYCLE96: 螺纹退刀槽

CYCLE98: 螺纹链

CYCLE99: 螺纹切削

2.2 循环编程

标准循环定义为带有名称和参数表的子程序。

调用和返回条件

在循环调用之前有效的 **G** 功能和可编程偏移在循环之后仍可以生效。

在循环调用之前定义加工平面 **G17**，用于钻削循环，或 **G18**，用于车削循环。

钻削循环时，在垂直于当前平面的轴向上进行钻削。

执行循环时的信息

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。

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。

循环调用与参数表

可以通过循环调用时的参数表传送该循环的供给参数。

说明

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

每个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。参数列表中的以下内容可以进行传输：

- R 参数（仅用于数值）
- 常量

如果在参数表中使用 R 参数，必须事先在程序中为其赋值。循环此时可按以下方式调用：

- 用一个不完整的参数表
或
- 忽略参数

如果要删除参数表结尾处的传输参数，必须提前使用“)”结束参数表。如果要在此时删除参数，则写入逗号“..., ...”作为通配符。

对于取值范围有限制的参数，不进行参数值的合理性检测，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。

说明

必须配置主轴的轴机床数据和通道机床数据。

循环调用

编写程序调用的各种方法可以参见单个循环的编程举例。

循环仿真

带有循环调用的程序可以首先通过仿真测试。

仿真时，循环的运行可在屏幕上模拟出来。

2.3 程序编辑器中的图形循环支持

控制系统中程序编辑器提供编程支持，可以在程序中插入循环调用并输入参数。

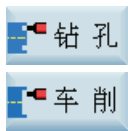






功能

循环支持由三部分组成：

1. 循环选择
2. 用于输入参数的屏幕窗口
3. 各个循环的辅助图

循环支持的条件

按照下列步骤向程序中添加循环调用：

- | | |
|---|--|
|  | <ol style="list-style-type: none"> 1. 在相关水平软键中选择循环类型以打开下一级垂直软键栏，直到带辅助图的相应输入屏幕跳出。 |
|  | <ol style="list-style-type: none"> 2. 值可以直接（数值）或间接（R 参数，例如：R27，或 R 参数表达式，例如：R27 + 10）输入。
在输入数值时，控制系统会自动检查该值是否在允许的值域内。 |
|  | <ol style="list-style-type: none"> 3. 有些只能使用少量数值的参数可使用该键进行选择。 |
|  | <ol style="list-style-type: none"> 4. 在钻削循环时，也可以使用该键来模态调用循环。欲取消模态调用，可按下以下软键： <div style="margin-left: 20px;">  </div> |
|  | <ol style="list-style-type: none"> 5. 按下该软键确认输入。欲取消输入，可按下以下软键： <div style="margin-left: 20px;">  </div> |

重新编译



程序代码的重新编译可以借助循环支持对现有的程序中进行更改。

将光标定位在需要更改的行上并按下该软键。这样就可以再次打开生成该程序的、相应的输入屏幕窗口，可以重新修改并保存数值。

2.4 钻削循环

2.4.1 概述

钻削循环是根据 DIN66025 所确定的运动过程，用于钻削、镗孔、攻丝等等。

作为子程序通过一个确定的名称和参数表调用。

它们在工艺流程中有所不同，由此在其参数上也不尽相同。

钻削循环可以模态有效，即，在包含运动命令的每个程序段结束处执行它们。

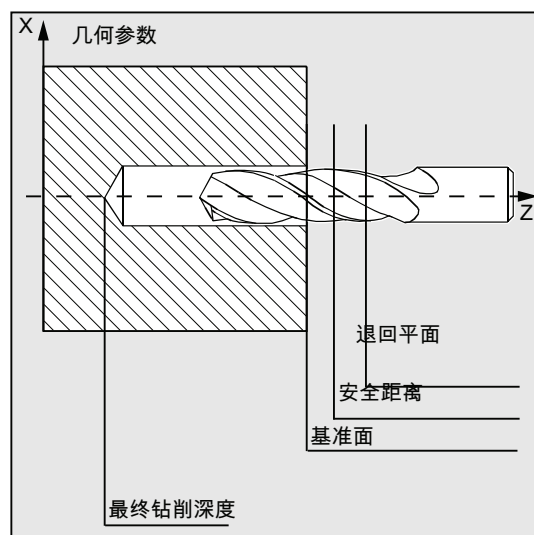
有两种参数种类：

- 几何参数
- 加工参数

几何参数对于所有钻削循环都相同。该参数定义基准面和退回平面、安全距离以及绝对和相对最终钻削深度。几何参数仅说明一次，即在描述第一个钻削循环 CYCLE82 时。

对于各个循环，加工参数有不同的含义和作用。因此将在各循环中单独说明。

有关几何参数的描述，见下图：



2.4.2 前提条件

调用和返回条件

钻削循环编程和具体的轴名称无关。在调用循环前，必须在上一级程序中逼近钻削位置。

如果在钻削循环中没有定义进给率、主轴转速和主轴旋转方向的参数，则须在零件程序中编程合适的值。

调用循环前生效的 G 功能和当前的数据组在循环中保留。

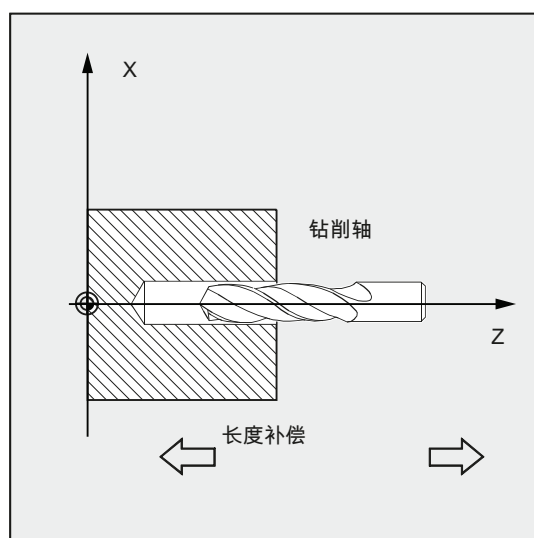
平面定义

钻削循环的前提条件是：通过选择 G17 平面并激活可编程的偏移定义了进行加工的工件坐标系。钻削轴始终是该坐标系中垂直于当前平面的轴。

在调用前必须选择长度补偿。该长度补偿始终垂直作用于所选的平面并在循环结束后仍有效。

因此在车床上，钻削轴为 Z 轴。在工件端面上钻削。

有关车削时的钻削轴的描述，见下图：



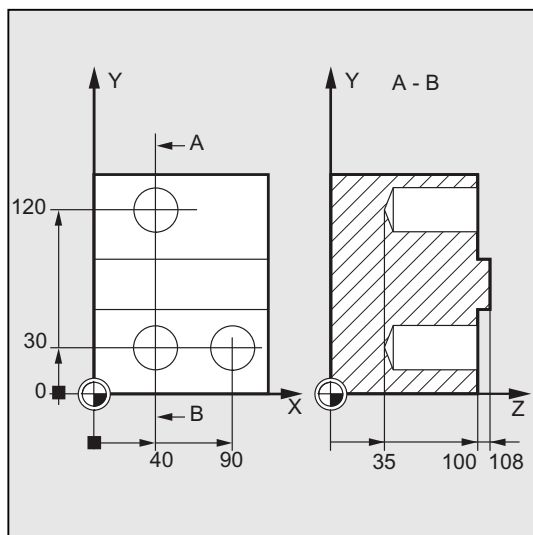
停留时间编程

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。

车床上钻削循环的应用特点

在简易型，不带驱动刀具的车床上，钻削循环只能在端面（使用 Z 轴）的车削中心上使用。必须在 G17 平面调用钻削循环。

有关不使用驱动刀具的车削中心钻削的描述，见下图：

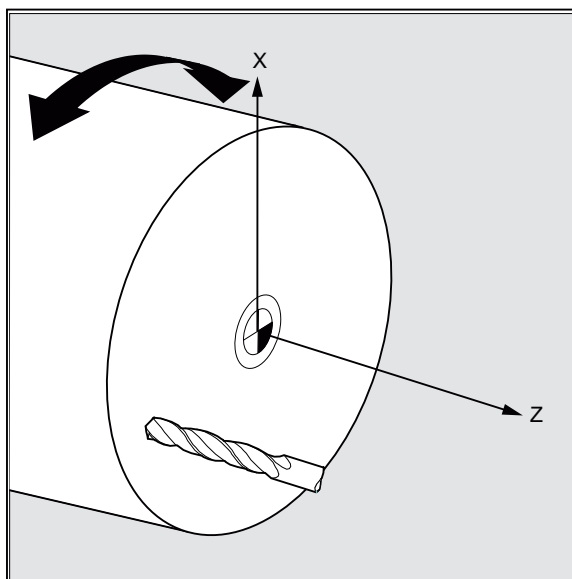


在带驱动刀具的车床上，当机床结构允许时，也可以在端面的偏心位置或侧面进行钻削。

在端面偏心钻削时，要注意以下内容：

- 工作平面为 G17 - 相应的 Z 轴为刀具轴。
- 还可关于 X 轴和 C 轴对钻削位置进行编程。

有关带驱动刀具的端面钻削的描述，见下图：

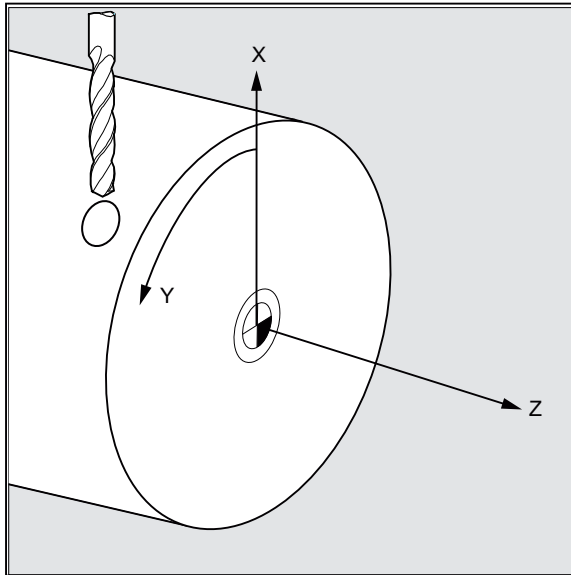


2.4 钻削循环

在侧面钻削时，要注意以下内容：

- 工作平面为 **G19** - 相应的 **X** 轴为刀具轴。
- 可以将钻削位置与 **Z** 轴和 **C** 轴一起编程。

有关带驱动刀具的侧面钻削的描述，见下图：



2.4.3 钻削，定中心 - CYCLE81

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）

功能

刀具以编程的主轴转速和进给速度钻削，直至输入的最终钻削深度。

过程

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：

使用 G0 逼近前移了安全距离的基准面

- 以在所调用程序中编程的进给率（G1）运行到最终钻削深度
- 使用 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）有不同的值。在循环中通常假设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）参考基准面而生效。基准面前移相应的安全距离。

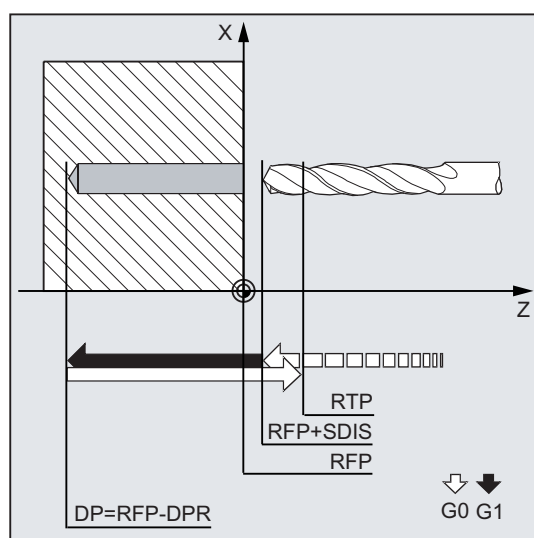
安全距离生效的方向由循环自动确定。

DP 和 DPR（最终钻削深度）

钻削深度可以通过到基准面的绝对尺寸（DP）设定，也可以通过相对尺寸（DPR）设定。

在通过相对尺寸设定时，循环通过基准面和退回平面的位置自行计算所产生的深度。

有关最终钻削深度的描述，见下图：

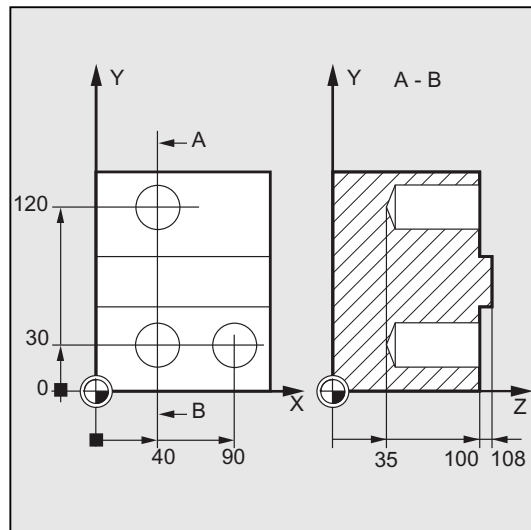
**说明**

如果既输入了 DP 值，又输入了 DPR 值，则会通过 DPR 推导出最终钻削深度。如果该值与通过 DP 编程的绝对深度不同，则在对话框中输出消息“深度：对应相对深度的值”。如果基准面和退回平面的值一致，则不允许设定相对深度。显示故障信息 61101：“基准面定义错误”，不执行循环。如果退回平面位于基准面之后，即其到最终钻削深度的距离更小时，也会输出故障信息。

编程示例：钻削_定心

该程序使用 **CYCLE81** 钻削循环产生三个钻孔。钻削轴始终为 Z 轴。

有关不使用驱动刀具的车削中心钻削的描述，见下图：



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	; 逼近第一个钻削位置
N40 CYCLE81 (110, 100, 2, 35,)	; 循环调用，使用绝对钻削深度，安全距离和不完全的参数表
N50 Y30	; 逼近下一个钻削位置
N60 CYCLE81 (110, 102, , 35,)	; 循环调用，不设定安全距离
N70 G0 G90 F180 S300 M03	; 确定工艺数值
N80 X90	; 逼近下一个位置
N90 CYCLE81 (110, 100, 2, , 65,)	; 循环调用，使用相对钻削深度和安全距离
N100 M30	; 程序结束

2.4.4 钻削，镗平面 - CYCLE82

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在最终钻削深度的停留时间（断屑）

功能

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

过程

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。

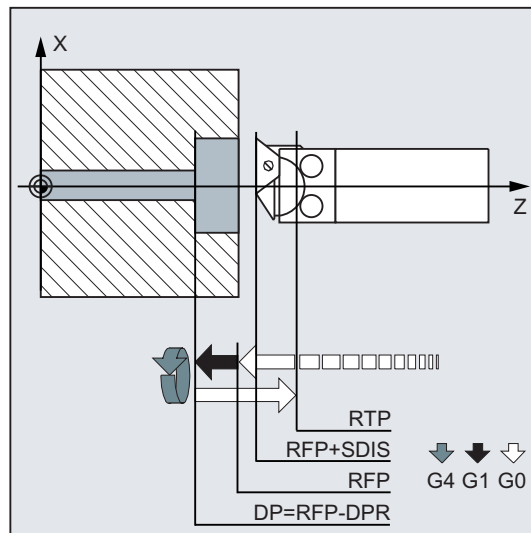
该循环产生以下的运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 使用循环调用前编程的进给率 (G1) 运行到最终钻削深度
- 在最终钻削深度的停留时间
- 使用 G0 返回到退回平面

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE82 的参数：



DTB（停留时间）

在 DTB 下编程在最终钻削深度（断屑）的停留时间，单位秒。

编程示例 1 钻削_镗平面

在 X0 位置，使用循环 CYCLE82 执行一次钻削，深度为 20 mm。

停留时间为 3 秒，在钻削轴 Z 上安全距离为 2.4 mm。

```

N10 G0 G90 G54 F2 S300 M3           ; 确定工艺数值
N20 D1 T6 Z50                       ; 返回退回平面
N30 G17 X0                           ; 逼近钻削位置
N40 CYCLE82 (3, 1.1, 2.4, -20, ,    ; 循环调用，以绝对值设定最终钻削深度和安全距离
3)
N50 M2                               ; 程序结束
    
```

编程示例 2

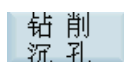
按照以下步骤进行：



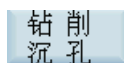
1. 选择所需操作区域。



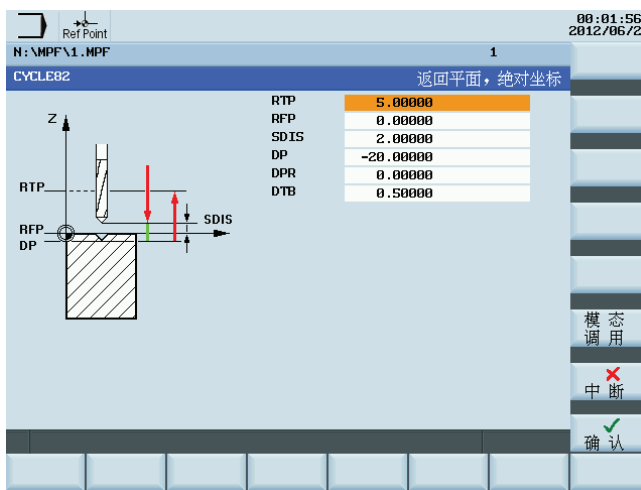
2. 打开可用钻削循环的垂直软键栏。



3. 在垂直软键栏中按下该软键。



4. 按下该软键打开 CYCLE82 的窗口。 根据需求设置循环参数。



5. 使用该软键确认所作的设置。 循环即作为单独程序段被自动传递至程序编辑器。

2.4.5 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

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（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	递减数量（不输入符号）
		值： >0: 以数值形式递减 <0: 递减系数 =0: 不递减
DTB	实数	在钻削深度的停留时间（断屑）
		值： >0: 以秒为单位 <0: 以转为单位
DTS	实数	在起始点和排屑的停留时间
		值： >0: 以秒为单位 <0: 以转为单位
FRF	实数	第一个钻削深度的进给率系数（不输入符号）值范围： 0.001 ... 1 0.001 ... 1
VARI	整数	加工方式：断屑=0，排屑=1

参数	数据类型	说明	
AXN	整数	工具轴	
		值:	1: 第一根几何轴 2: 第二根几何轴 3: 第 3 根几何轴
MDEP	实数	最小钻削深度（仅连同递减系数）	
VRT	实数	断屑的可变退回值 (VARI=0)	
		值:	>0: 在退回值情况下 =0: 设定退回值 1 mm
DTD	实数	在最终钻削深度的停留时间	
		值:	>0: 以秒为单位 <0: 以转为单位 =0: 值与 DTB 相同
DIS1	实数	重新插入钻孔的可编程限制距离（排屑 VARI=1）	
		值:	>0: 应用可编程值 =0: 自动计算

功能

刀具以编程的主轴转速和进给速度钻削，直至输入的最终钻削深度。

深孔钻削在预定的最大进刀量范围内多次、分步骤地进行，直至达到最终钻削深度。

在达到每个钻深后，钻头可以退回到“基准面+安全距离”的位置，进行排屑，或每次退回 1 mm。

过程

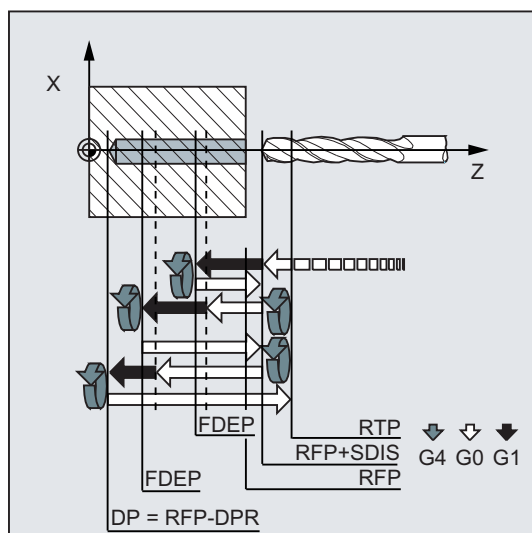
循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

该循环创建以下顺序：

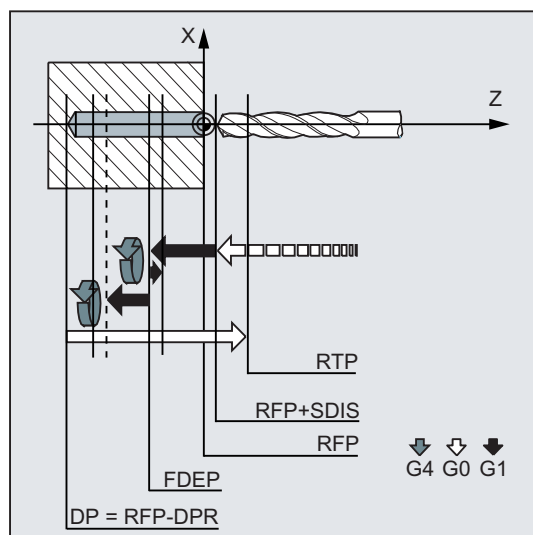
深孔钻削，带排屑 (VARI=1)：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 运行到第一个钻削深度，从使用程序调用通过参数 FRF（进给率系数）定义的进给率得出的进给率
- 在最终钻削深度的停留时间（参数 DTB）
- 使用 G0 退回到前移了安全距离的基准面，进行排屑
- 在起始点的停留时间（参数 DTS）
- 逼近最后达到的钻削深度，通过使用 G0 减少预期的距离
- 使用 G1 运行到下一个钻削深度（继续运动顺序直到达到最终钻削深度为止）
- 使用 G0 返回到退回平面



深孔钻削，带断屑 (VARI=0):

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 运行到第一个钻削深度，从使用程序调用通过参数 FRF（进给率系数）定义的进给率得出的进给率
- 在最终钻削深度的停留时间（参数 DTB）
- 使用 G1 和调用程序中编程的进给率从当前钻削深度退回 1 mm（进行断屑）
- 使用 G1 和编程的进给率运行到下一个钻削深度（继续运动顺序直到达到最终钻削深度为止）
- 使用 G0 返回到退回平面



参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参数 DP（或 DPR）、FDEP（或 FDPR）和 DAM 的关系

在循环中基于如下最终钻削深度、第一个钻削深度和递减数量计算中间钻削深度：

- 在第一个步骤中，移动使用第一个钻削深度参数化的深度，不允许超过总钻削深度。
- 第二次钻削距离等于“第一次钻削距离 - DAM”，只要递减量没有超过该距离。
- 只要剩余深度大于递减数量两倍，下一个钻削行程就相当于递减数量。
- 相等地划分并移动最后两个钻削行程，因此，始终大于递减数量的一半。

- 如果第一个钻削深度与总深度不匹配，则输出错误消息 61107“不正确定义的第一个钻削深度”，并且不执行循环。

FDPR 参数在循环中的作用与 DPR 参数相同。如果基准面和退回平面的值一致，则可以将第一个钻削深度定义为相对值。

如果编程的第一个钻削深度大于最终钻削深度，则从不超过最终钻削深度。循环将自动减少第一个钻削深度，直到在只钻削一次时达到最终钻削深度为止，因此将只钻削一次。

DTB（停留时间）

在 DTB 下编程在最终钻削深度（断屑）的停留时间，单位秒。

DTS（停留时间）

只有在 VARI=1（排屑）时才执行起始点的停留时间。

FRF（进给率系数）

使用该参数，可以输入有效进给率（只应用于接近到循环中的第一个钻削深度）的减少系数。

VARI（加工方式）

如果设定参数 VARI=0，达到断屑的每个钻削深度后，钻头将退回 1 mm。如果 VARI=1（排屑），在各种情况下，钻头都会移动到偏移安全距离的基准面。

说明

如下在循环内部计算预期距离：

- 如果钻削深度为 30 毫米，预期距离的值始终为 0.6 毫米。
 - 更大的钻深可使用公式“钻深/50”（最大值为 7 毫米）。
-

AXN (刀具轴)

通过 AXN 编程钻削轴，可以忽略在车床上使用深孔钻削循环时从平面 G18 到 G17 的切换。

这些标识符具有下列含义：

AXN=1	当前平面的第一根轴
AXN=2	当前平面的第二根轴
AXN=3	当前平面的第三根轴

例如，要在 G18 平面上加工中心孔（在 Z 中），编程：

G18

AXN=1

MDEP (最小钻削深度)

您可以基于递减系数定义钻削行程计算的最小钻削深度。如果计算的钻削行程比最小钻削深度短，则在等于最小钻削深度的行程上加工剩余深度。

VRT (VARI=0 时，断屑的可变退回值)

您可以对断屑的退回行程进行编程。

DTD (在最终钻削深度的停留时间)

可以以秒或转为单位输入在最终钻削深度的停留时间。

DIS1 (VARI=1 时的可编程限制距离)

可以编程重新插入孔后的限制距离。

如下在循环内计算限制距离：

- 钻削深度最大 30 毫米时，将值设为 0.6 毫米。
- 对于更大的钻削深度，限制距离是 $(RFP + SDIS - \text{当前深度}) / 50$ 的结果。如果该计算值 >7 ，则应用最大 7 毫米的极限。

编程示例：深孔钻削

该程序在位置 X0 执行循环 CYCLE83。第一次钻削深度上刀具的停留时间为零并带断屑。将最终钻削深度和第一个钻削深度作为绝对值输入。钻削轴为 Z 轴。

```

N10 G0 G54 G90 F5 S500 M4           ; 确定工艺数值
N20 D1 T6 Z50                       ; 返回退回平面
N30 G17 X0                           ; 逼近钻削位置
N40 CYCLE83(3.3, 0, 0, -80, 0, -10, 0, 0, 0, 0, 1, 0) ; 循环调用，带有绝对值的深度参数
N50 M2                               ; 程序结束

```

2.4.6 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, 0, 0, VARI, DAM, VRT)

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在螺纹深度的停留时间（断屑）
SDAC	整数	循环结束后的旋转方向 值： 3、4 或 5（表示 M3、M4 或 M5）
MPIT	实数	螺距作为螺纹尺寸（带符号）： 3（M3）到 48（M48）的数值范围；符号决定螺纹的旋转方向。

参数	数据类型	说明
PIT	实数	螺距作为值（带符号） 值范围： 0.001 ... 2000.000 mm）；符号决定螺纹的旋转方向
POSS	实数	主轴位置，用于循环中定向主轴停（单位度）
SST	实数	攻丝速度
SST1	实数	返回速度
AXN	整数	工具轴
		值 1): 1: 当前平面的第 1 根轴 2: 当前平面的第 2 根轴 3: 当前平面的第 3 根轴
PSYS	整数	内部参数；只允许默认值 0
PSYS	整数	内部参数；只允许默认值 0
VARI	整数	加工方式
		值: 0: 在一个周期中攻丝 1: 深孔攻丝，带断屑 2: 深孔攻丝，带排屑
DAM	实数	增量钻削深度 数值范围： 0 <= 最大值
VRT	实数	断屑的可变返回值 数值范围： 0 <= 最大值

1) 具体第 1、2、3 根轴的定义取决于当前所选平面。

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。

CYCLE84 可用于不带补偿夹具的攻丝孔。带补偿夹具的攻丝可以使用单独的循环 CYCLE840。

说明

如果用于镗削的主轴能够达到位置控制的运行状态，则始终可以使用循环 CYCLE84。

过程

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。

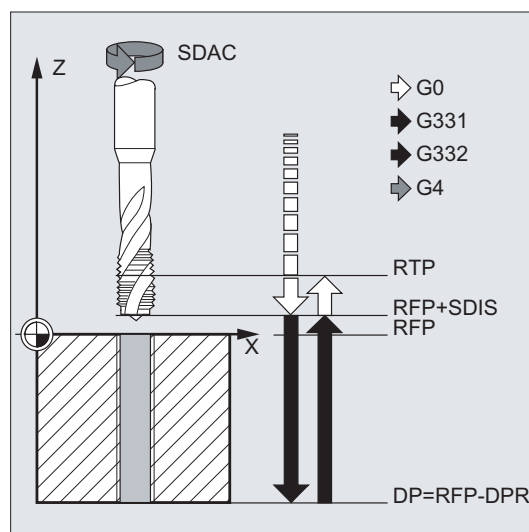
该循环产生以下的运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 引导主轴停止（参数 POSS 中的数值）并将主轴切换为进给轴模式
- 攻丝至最终的钻削深度和速度 SST
- 在螺纹深度的停留时间（参数 DTB）
- 退回到被安全距离、速度 SST1 以及方向反转前移了的基准面
- 退回到带 G0 的退回平面；通过在调用循环前重新设置主轴速度和在 SDAC 下设置的旋转方向来重新启动主轴模式。

参数说明

参数 RTP, RFP, SDIS, DP, DPR - 参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE84 的参数：



DTB（停留时间）

必须以秒为单位编程停留时间。在攻丝盲孔时，建议省去停留时间。

SDAC（循环结束后的旋转方向）

在 SDAC 下，在循环结束时设置旋转方向。

对于攻丝来说，这个方向会随着循环自动改变。

MPIT 和 PIT（螺距作为螺纹尺寸和值）

螺距的值可以定义为螺纹尺寸（仅 M3 和 M48 之间的公制螺纹）或定义为值（从一个螺纹线到下一个螺纹线的距离作为数值）。任何不需要的参数都会在调用中被省去或者被赋值为零。

RH 或者 LH 螺纹由螺距参数的符号定义：

- 正值 → 右（同 M3 一样）
- 负值 → 左（同 M4 一样）

如果两个螺距参数的值相互矛盾，循环会输出报警 61001“螺距错误”，并中断循环。

POSS（主轴位置）

在攻丝前，通过命令 SPOS 将主轴在循环中定位停止并转换为位置控制。

在 POSS 下设定主轴停止的位置。

SST（速度）

参数 SST 包含用于带 G331 的攻丝程序段的主轴速度。

SST1（返回速度）

从攻丝孔返回的速度是在 SST1 下设定的。

如果这个参数被设为零，则会以在 SST 下设定的速度返回。

AXN (刀具轴)

通过 AXN 编程钻削轴，可以忽略在车床上使用深孔钻削循环时从平面 G18 到 G17 的切换。

这些标识符具有下列含义：

平面	定义	进给方向
X/Y	G17	AXN=1: 当前平面的第 1 根轴为 X 轴 AXN=2: 当前平面的第 2 根轴为 Y 轴 ¹⁾ AXN=3: 当前平面的第 3 根轴为 Z 轴
Z/X	G18	AXN=1: 当前平面的第 1 根轴为 Z 轴 AXN=2: 当前平面的第 2 根轴为 X 轴 AXN=3: 当前平面的第 3 根轴为 Y 轴 ¹⁾
Y/Z	G19	AXN=1: 当前平面的第 1 根轴为 Y 轴 ¹⁾ AXN=2: 当前平面的第 2 根轴为 Z 轴 AXN=3: 当前平面的第 3 根轴为 X 轴

¹⁾ 如果 Y 轴存在。

例如，要在 G17 平面上加工中心孔（在 Z 中），编程：

G17

AXN=3

深孔攻丝： VARI, DAM, VRT

通过 VARI 参数，可以区分一般攻丝（VARI=0）和深孔攻丝（VARI≠0）。

在进行深孔攻丝时，可以选择断屑（以可变距离从当前的钻削深度返回，参数 VRT、VARI=1）或者排屑（从基准面返回，VARI=2）。普通深孔钻削循环 CYCLE83 也可以同样地完成这些功能。

可以通过参数 DAM 来指定单行程的增量钻削深度。循环内部会计算中间的距离：

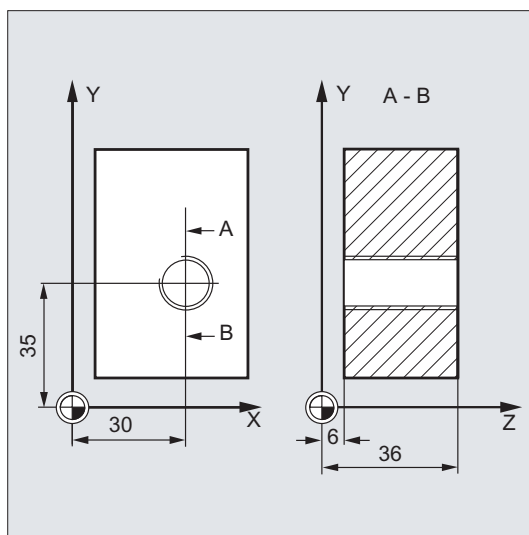
- 在每个步骤中执行设定的增量钻孔深度，直至距离最终钻削深度小于 (<) 2 x DAM。
- 剩下的钻削深度会被一分为二并以两个步骤执行。因此，最小的钻削深度不应小于 DAM/2。

说明

当在循环中进行攻丝时，旋转的方向会自动反转。

编程示例 1 刚性攻丝

在 XY 平面上的位置 X30 Y35 处进行螺纹攻丝（不带补偿夹具）；攻丝轴为 Z 轴。为设定停留时间；深度被设定为相对数值。必须对旋转方向参数和螺距参数进行赋值。公制螺纹 M5 被攻丝。



```

N10 G0 G90 T11 D1
N20 G17 X30 Y35 Z40
N30 CYCLE84(40, 36, 2, , 30, , 3, 5, , 90, 200,
500, 3, 0, 0,0, ,0.00000)
N40 M02
    
```

：确定工艺数值
 ；逼近钻削位置
 循环调用；参数 PIT 已经被省去；未输入绝对深度或者停留时间的数值；主轴停留在 90 度；攻丝速度为 200，退回速度为 500。
 ；程序结束

编程示例 2

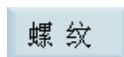
按照以下步骤进行：



1. 选择所需操作区域。



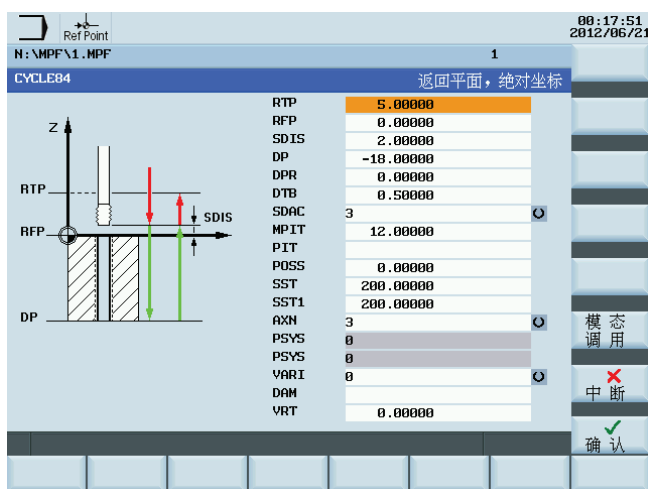
2. 打开可用钻削循环的垂直软键栏。



3. 按下该软键打开下一级软键栏。



4. 按下该软键打开 CYCLE84 的窗口。 根据需求设置循环参数。



5. 使用该软键确认所作的设置。循环即作为单独程序段被自动传递至程序编辑器。

2.4.7 攻丝，带补偿衬套 - CYCLE840

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在螺纹深度的停留时间（断屑）
SDR	整数	退回时的旋转方向 值： 0（自动反向）， 3 或 4（表示 M3 或 M4）
SDAC	整数	循环结束后的旋转方向 值： 3、4 或 5（表示 M3、M4 或 M5）
ENC	整数	带/不带编码器的攻丝 值： 0 = 带编码器， 1 = 不带编码器
MPIT	实数	螺距作为螺纹尺寸（带符号）： 值范围： 0（表示 M0）， 48（表示 M48）
PIT	实数	螺距作为值（带符号） 值范围： 0.001 ... 2000.000 毫米
AXN	整数	工具轴 值 1): 1: 当前平面的第 1 根轴 2: 当前平面的第 2 根轴 3: 当前平面的第 3 根轴

1) 具体第 1、2、3 根轴的定义取决于当前所选平面。

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。

使用该循环执行带补偿卡盘的攻丝：

- 不带编码器
- 带编码器

过程

带补偿卡盘、不带编码器的攻丝

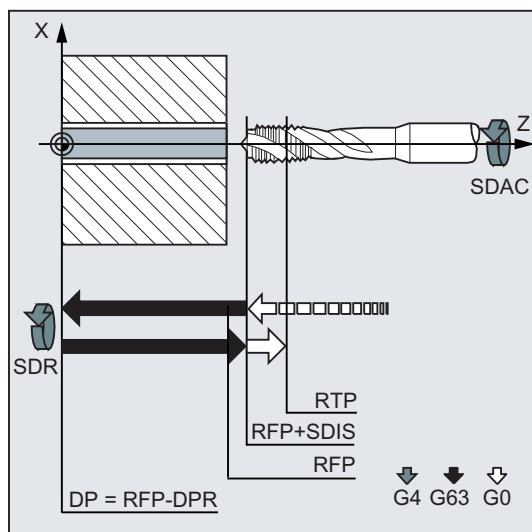
循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

该循环创建以下运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 攻丝直到最终钻深
- 在攻丝深度的停留时间（参数 DTB）
- 退回前移了安全距离的基准面
- 使用 G0 返回到退回平面

参见以下不带编码器的 CYCLE840 的参数：



过程

带补偿卡盘、带编码器的攻丝

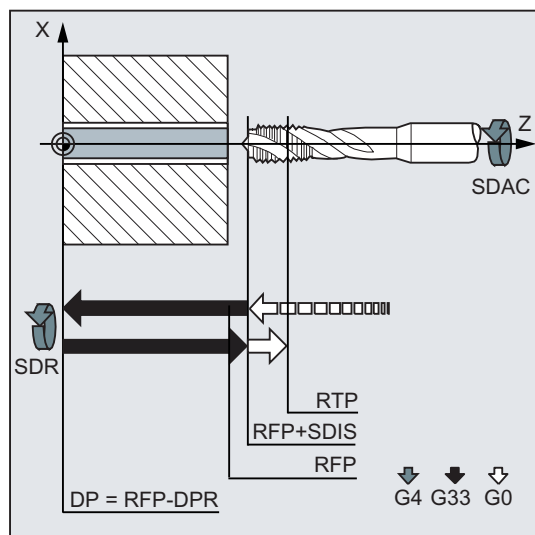
循环开始之前到达的位置：

钻削位置是指所选择平面的两个轴位置。

该循环创建以下运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 攻丝直到最终钻深
- 在螺纹深度的停留时间（参数 DTB）
- 退回前移了安全距离的基准面
- 使用 G0 返回到退回平面

参见以下带编码器的 CYCLE840 的参数：



参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见主题“钻削，定中心 - CYCLE81 (页 127)”。

DTB (停留时间)

必须以秒为单位编程停留时间。 只在不带编码器的攻丝中有效。

SDR（返回时的旋转方向）

如果主轴自动反向，则必须设置 SDR=0。

如果定义机床数据，从而不设置编码器（在这种情况下，机床数据 MD30200 \$MA_NUM_ENCS 为 0），对于旋转反向，必须将参数赋值为 3 或 4；否则，将输出报警 61202“没有编程主轴方向”，且循环中断。

SDAC（旋转方向）

由于也可以模态调用循环（参见章节“程序编辑器中的图形循环支持 (页 122)”），需要更多螺纹孔攻丝的旋转方向。这在参数 SDAC 中编程并且相当于上位程序中第一次调用前编程的旋转方向。如果 SDR = 0，SDAC 下指定的值没有作用，在设置时被忽略。

ENC（攻丝）

虽然存在编码器，但是如果不使用编码器执行攻丝，必须将参数 ENC 赋值为 1。

但是，如果没有安装编码器，并且将参数赋值为 0，则在循环中忽略。

MPIT 和 PIT（螺距作为螺纹尺寸和值）

只有在使用编码器执行攻丝时螺距的参数才相关。循环根据主轴转速和螺距计算进给率。

螺距的值可以定义为螺纹尺寸（仅 M3 和 M48 之间的公制螺纹）或定义为值（从一个螺纹线到下一个螺纹线的距离作为数值）。在调用中忽略各种情况下不需要的参数或将参数赋值为零。

如果两个螺距参数的值相互矛盾，循环会输出报警 61001“螺距错误”，并中断循环。

说明

根据机床数据 MD30200 \$MA_NUM_ENCS 中的设置，循环选择是带编码器还是不带编码器执行攻丝。

必须使用 M3 或 M4 对主轴的旋转方向进行编程。

在带有 G63 的螺纹程序段中，进给倍率开关和主轴转速倍率开关的值固定为 100%。

不带编码器的攻丝通常需要更长的补偿卡盘。

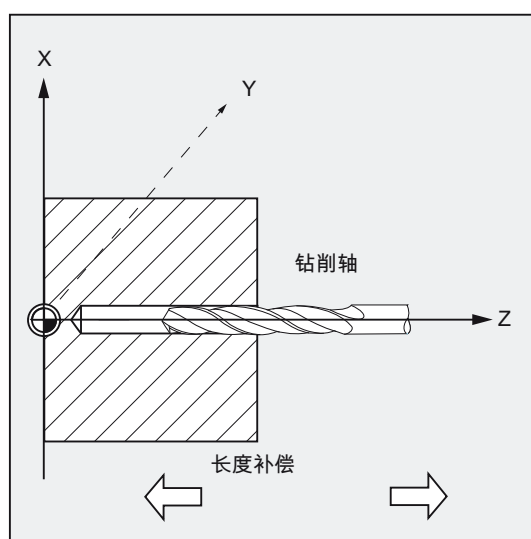
AXN (刀具轴)

下图显示要选择的钻削轴选项。

使用 G18:

- AXN=1; 相当于 Z
- AXN=2; 相当于 X
- AXN=3; 相当于 Y (如果 Y 轴存在的话)

有关 G18 平面的钻削轴的描述, 见下图:



使用 AXN (钻削轴的编号) 对钻削轴进行编程, 能够实现对钻削轴直接编程。

平面	定义	进给方向
X/Y	G17	AXN=1: 当前平面的第 1 根轴为 X 轴 AXN=2: 当前平面的第 2 根轴为 Y 轴 ¹⁾ AXN=3: 当前平面的第 3 根轴为 Z 轴
Z/X	G18	AXN=1: 当前平面的第 1 根轴为 Z 轴 AXN=2: 当前平面的第 2 根轴为 X 轴 AXN=3: 当前平面的第 3 根轴为 Y 轴 ¹⁾
Y/Z	G19	AXN=1: 当前平面的第 1 根轴为 Y 轴 ¹⁾ AXN=2: 当前平面的第 2 根轴为 Z 轴 AXN=3: 当前平面的第 3 根轴为 X 轴

¹⁾ 如果 Y 轴存在。

例如，要在 G17 平面上加工中心孔（在 Z 中），编程：

G17

AXN=3

编程示例：不带编码器的攻丝

在位置 X0 处不带编码器执行攻丝；钻削轴是 Z 轴。旋转方向的参数 SDR 和 SDAC 必须赋值；参数 ENC 赋值为 1，深度值为绝对值。可以忽略先导参数 PIT。在加工中使用补偿卡盘。

N10 G90 G0 G54 D1 T6 S500 M3	；确定工艺数值
N20 G17 X0 Z60	；逼近钻削位置
N30 G1 F200	；设置轨迹进给率
N40 CYCLE840(3, 0, , -15, 0, 1, 4, 3, 1, , , 3)	；循环调用：停留时间为 1 秒；退回 M4 的 旋转方向；循环 M3 后的旋转方向；没有安全距离
	已经忽略 MPIT 和 PIT 参数。
N50 M2	；程序结束

编程示例：带编码器的攻丝

该程序用于在位置 X0 处不带编码器的攻丝。钻削轴为 Z 轴。必须定义先导参数，编程旋转方向的自动反向。在加工中使用补偿卡盘。

N10 G90 G0 G54 D1 T6 S500 M3	；确定工艺数值
N20 G17 X0 Z60	；逼近钻削位置
N30 G1 F200	；设置轨迹进给率
N40 CYCLE840(3, 0, , -15, 0, 0, , , 0, 3.5, , 3)	；循环调用，不设定安全距离
N50 M2	；程序结束

2.4.8 铰孔 1 - CYCLE85

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在最终钻削深度的停留时间（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，直至输入的最终钻削深度。

分别以参数 FFR 和 RFF 中设定的进给率进行向内运动和向外运动。

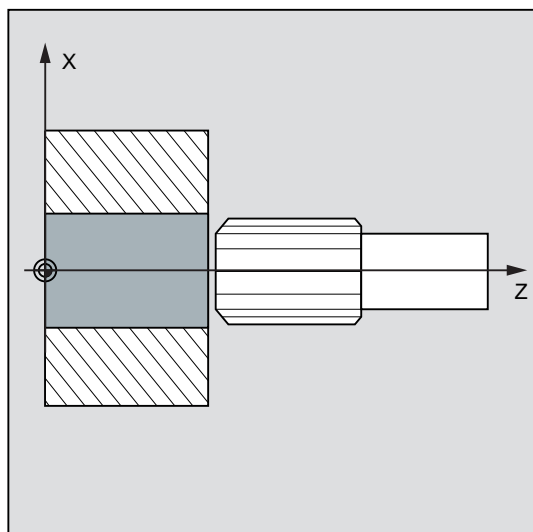
该循环可以用于铰孔。

过程

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。

有关操作顺序的描述，见下图：



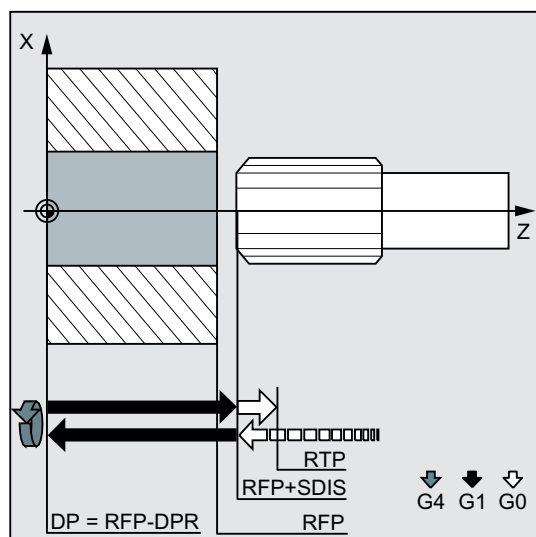
该循环产生以下的运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 和参数 FFR 下编程的进给率运行到最终钻削深度。
- 在最终钻削深度的停留时间
- 使用 G1 和参数 RFF 下设定的退回进给率返回到前移了安全距离的基准面
- 使用 G0 返回到退回平面

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE85 的参数：



DTB (停留时间)

在 DTB 下编程在最终钻削深度处的停留时间，单位秒。

FFR (进给率)

在 FFR 下设定的进给值在钻削时生效。

RFF (退回进给率)

在 RFF 下编程的进给值在从钻孔退回至“基准面 + 安全距离”时生效。

编程示例：第一次镗孔

在 Z70 X0 处调用循环 CYCLE85。钻削轴为 Z 轴。在循环调用中相对设定最终钻削深度，不编程停留时间。工件上边缘位于 Z0 处。

```

N10 G90 G0 S300 M3
N20 T3 G17 G54 Z70 X0 ; 逼近钻削位置
N30 CYCLE85(10, 2, 2, , 25, , 300, 450) ; 循环调用，不编程停留时间
N40 M2 ; 程序结束
    
```

2.4.9 镗孔 - CYCLE86

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在最终钻削深度的停留时间（断屑）
SDIR	整数	旋转方向 值： 3（表示 M3）， 4（表示 M4）
RPA	整数	平面内第 1 轴上的退回路径（增量，输入符号）
PSYS	整数	内部参数；只允许默认值 0
RPAP	实数	钻削轴上的退回路径（增量，输入符号）
POSS	实数	主轴位置，用于循环中定向主轴停（单位度）

功能

循环支持使用镗杆进行的镗孔。

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。

进行镗孔 2 时，到达钻削深度后主轴定向停止。接着，以快速移动运行至编程的退回位置，并从该位置返回退回平面。

过程

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。

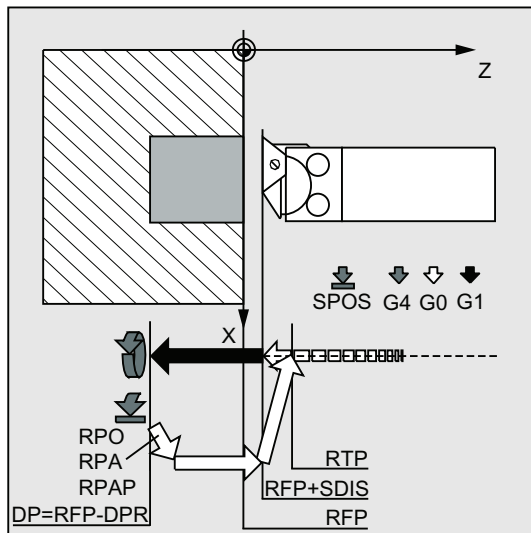
该循环产生以下的运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 和循环调用前编程的进给率运行到最终钻削深度
- 执行在最终钻削深度处的停留时间
- 在 POSS 下编程的主轴位置上执行定向主轴停止
- 使用 G0 在最多 3 个轴上运行退回路径
- 在钻削轴上使用 G0 退回前移了安全距离的基准面
- 使用 G0 返回到退回平面（在平面中两个轴上的起始钻削位置）

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE86 的参数：



DTB (停留时间)

在 DTB 下编程在最终钻削深度（断屑）的停留时间，单位秒。

SDIR (旋转方向)

使用此参数确定循环中执行钻削的旋转方向。如果旋转方向的赋值不是 3 或者 4 (M3/M4)，则输出报警 61102“未编程主轴旋转方向”，且不执行循环。

RPA (退回运行, 第 1 轴)

在此参数下定义到达最终钻削深度和定向主轴停止后，第 1 轴上执行的退回运行（横坐标）。

RPAP (退回运行, 钻削轴)

在此参数下定义到达最终钻削深度和定向主轴停止后，钻削轴上执行的退回运行。

POSS (主轴位置)

必须在 POSS 下编程到达最终钻削深度后，定向主轴停止的位置，单位度。

说明

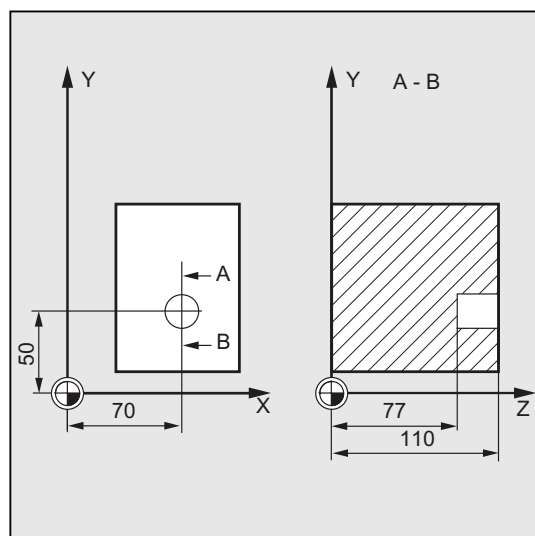
可以定向停止生效的主轴。通过传输参数编程相应的角度值。

如果钻削主轴可在位置控制运行中运行，才能使用循环 CYCLE86。

编程示例：第二次镗孔

在 XY 平面中的 X70 Y50 处调用循环 CYCLE86。钻削轴为 Z 轴。以绝对值编程最终钻削深度，不设置安全距离。在最终钻削深度处的停留时间为两秒。将工件顶部边沿定位在 Z110 处。在循环中要求主轴以 M3 方向旋转，并在 45 度角处停止。

参见以下第二次镗孔的示例：



2.4 钻削循环

```

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 ; 逼近钻削位置
N50 CYCLE86 (112, 110, , , , 3, -1, ; 循环调用, 使用绝对钻深
0, +1, )
N60 M30 ; 程序结束
    
```

2.4.10 带停止 1 的镗孔 - CYCLE87

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
SDIR	整数	旋转方向 值： 3（表示 M3）， 4（表示 M4）

功能

刀具以编程的主轴转速和进给速度钻削，直至输入的最终钻削深度。

进行镗孔 3 时，到达最终钻削深度后，执行不定向主轴停止 M5，之后执行编程的停止 M0。按下列键继续进行快速向外运动，直至到达退回平面：



过程

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：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 和循环调用前编程的进给率运行到最终钻削深度
- 使用 M5 停止主轴
- 按下列键：

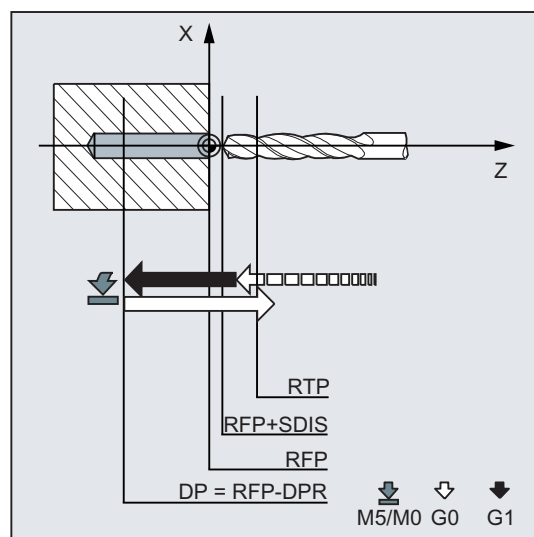


- 使用 G0 返回到退回平面

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE87 的参数：



SDIR (旋转方向)

使用此参数确定循环中执行钻削的旋转方向。

如果旋转方向的赋值不是 3 或者 4 (M3/M4)，则输出报警 61102“未编程主轴旋转方向”，且循环中断。

2.4 钻削循环

编程示例：第三次镗孔

在 XY 平面中的 X0 上调用循环 CYCLE87。钻削轴为 Z 轴。以绝对值设定最终钻削深度。安全距离为 2 毫米。

```

N20 G0 G17 G90 F200 S300 X0           ; 确定工艺值和钻削位置
N30 D3 T3 Z13                          ; 返回退回平面
N50 CYCLE87(13, 10, 2, -7, , 3)        ; 调用循环, 编程的主轴旋转方向为 M3
N60 M2                                   ; 程序结束
    
```

2.4.11 带停止 2 的钻孔 - CYCLE88

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在最终钻削深度的停留时间（断屑）
SDIR	整数	旋转方向 值： 3（表示 M3）， 4（表示 M4）

功能

刀具以编程的主轴转速和进给速度钻削，直至输入的最终钻削深度。进行镗孔 4 时，到达最终钻削深度后，产生停留时间、不定向主轴停止 M5 以及编程的停止 M0。按下列键继续进行快速向外运动，直至到达退回平面：



过程

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：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 和循环调用前编程的进给率运行到最终钻削深度
- 在最终钻削深度的停留时间
- 通过 M5 和 M0 停止主轴和程序。程序停止后，按下列键：

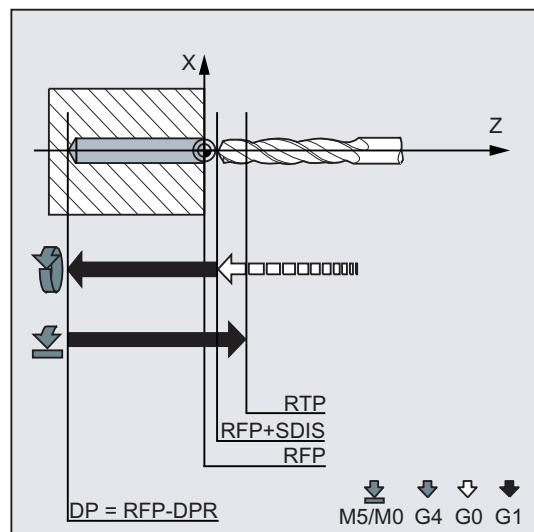


- 使用 G0 返回到退回平面

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE88 的参数：



DTB (停留时间)

在 DTB 下编程在最终钻削深度（断屑）的停留时间，单位秒。

SDIR (旋转方向)

编程的旋转方向在到最终钻削深度的运行路径中生效。

如果旋转方向的赋值不是 3 或者 4 (M3/M4)，则输出报警 61102“未编程主轴旋转方向”，且循环中断。

2.4 钻削循环

编程示例：第四次镗孔

在 X0 处调用循环 CYCLE88。钻削轴为 Z 轴。将安全距离编程为 3 毫米，相对于基准面设定钻削深度。M4 在循环中生效。

N10 G17 G54 G90 F1 S450 M3 T1	； 确定工艺数值
N20 G0 X0 Z10	； 逼近钻削位置
N30 CYCLE88 (5, 2, 3, , 72, 3, 4)	； 调用循环，编程的主轴旋转方向为 M4
N40 M2	； 程序结束

2.4.12 铰孔 2 - CYCLE89

编程

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

参数

参数	数据类型	说明
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	最终钻削深度（绝对）
DPR	实数	相对于基准面的最终钻削深度（不输入符号）
DTB	实数	在最终钻削深度的停留时间（断屑）

功能

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

过程

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。

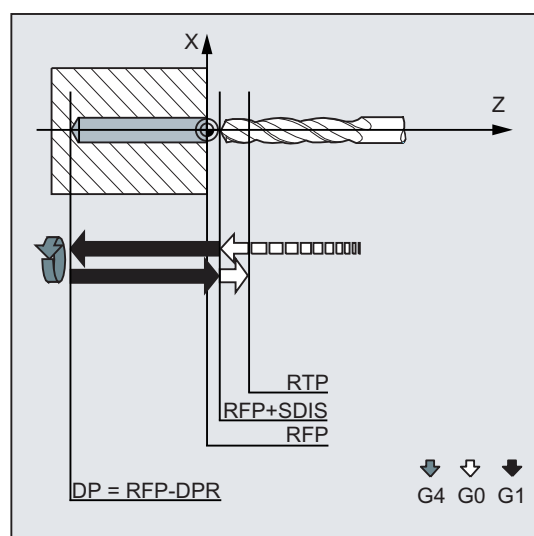
该循环产生以下的运动过程：

- 使用 G0 逼近前移了安全距离的基准面
- 使用 G1 和循环调用前编程的进给率运行到最终钻削深度
- 执行在最终钻削深度处的停留时间
- 使用 G1 和相同的进给值退回到前移了安全距离的基准面
- 使用 G0 返回到退回平面

参数说明

有关参数 RTP、RFP、SDIS、DP 和 DPR 的说明，参见章节“钻削，定中心 - CYCLE81 (页 127)”。

参见以下 CYCLE89 的参数：



DTB (停留时间)

在 DTB 下编程在最终钻削深度 (断屑) 的停留时间，单位秒。

2.5 车削循环

编程示例：第五次镗孔

在 X0 上调用钻削循环 CYCLE89，安全距离为 5 毫米，并且以绝对值设定最终钻削深度。钻削轴为 Z 轴。

N10 G90 G17 F100 S450 M4	； 确定工艺数值
N20 G0 X0 Z107	； 逼近钻削位置
N30 CYCLE89(107, 102, 5, 72, , 3)	； 循环调用
N40 M2	； 程序结束

2.5 车削循环

2.5.1 前提条件

调用和返回条件

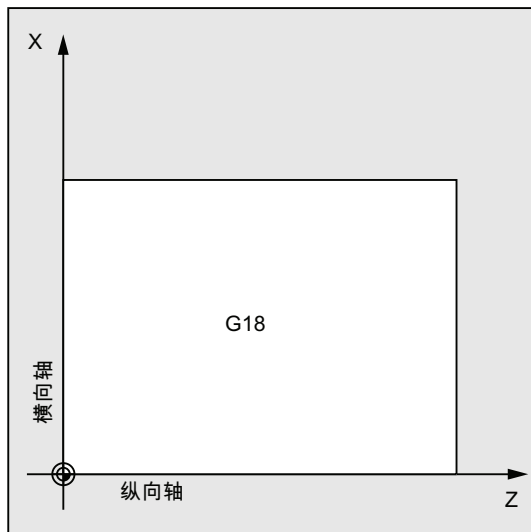
在循环调用之前有效的 G 功能在循环之后仍然保持有效。

平面定义

在循环调用之前定义加工平面。通常在车削时使用的是 G18 平面（ZX 平面）。下文中将车削时生效平面中的两根轴分别称为纵向轴（该平面的第一根轴）和横向轴（该平面的第二根轴）。

在车削循环中直径编程有效时始终将平面的第二轴作为横向轴（参见编程手册）。

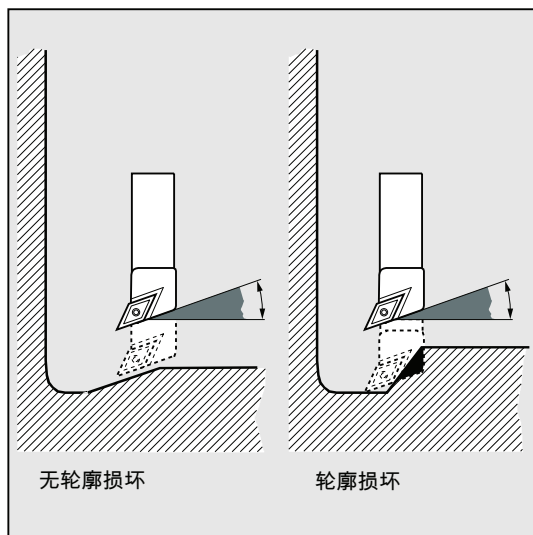
有关 G18 平面的描述，见下图：



轮廓监控与刀具的自由切削角度相关

使用特定的车削循环（在该循环中生成带底切的运行）监控生效刀具的自由切削角是否会造成轮廓损伤。该角度作为刀具补偿中的值输入（D 补偿中的参数 DP24 下）。输入角度值，1 到 90 度（0 = 无监控）之间，不带符号。

纵向轮廓监控：



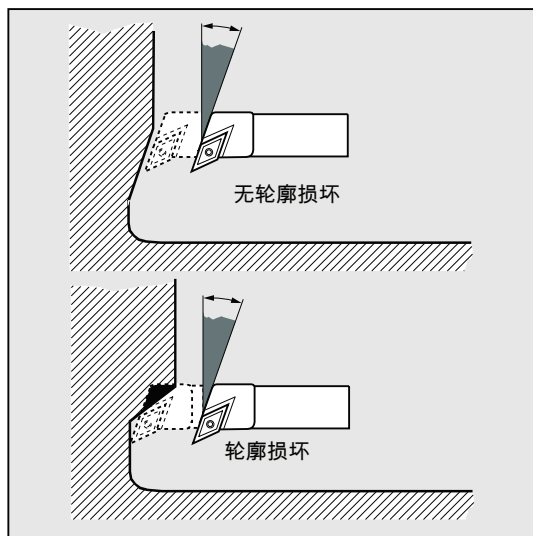
自由切削角的输入取决于加工方式为纵向还是横向。在纵向加工和横向加工时使用同一刀具时，必须根据不同的自由切削角使用两个刀具补偿。

在循环中检测用所选择的刀具是否可以加工编程的轮廓。

如果不能使用该刀具进行加工，循环将中断，并输出错误消息（在切削中）；或者，继续加工轮廓并输出消息（带退刀槽循环）。然后由刀沿几何尺寸确定轮廓。

如果自由切削角在刀具补偿中设定为零，则不执行监控。具体的响应在单独的循环中描述。

平面轮廓监控：



2.5.2 切断 - CYCLE92

编程

CYCLE92(SPD, SPL, DIAG1, DIAG2, RC, SDIS, SV1, SV2, SDAC, FF1, FF2, SS2, 0, VARI, 1, 0, AMODE)

参数

参数	数据类型	说明
SPD	实数	横向轴上的起始点（绝对，始终为直径）
SPL	实数	纵向轴上的起始点（绝对）
DIAG1	实数	减少速度的深度 \varnothing （绝对）
DIAG2	实数	最终深度 \varnothing （绝对）
RC	实数	倒角宽度或倒圆半径
SDIS	实数	安全距离（加到参考点，不输入符号）
SV1	实数	恒定切削速度 V
SV2	实数	恒定切削速度下的最大速度
SDAC	整数	主轴旋转方向 值： 3: M3 4: M4
FF1	实数	到达主轴减速深度前的进给率
FF2	实数	从减速深度到最终深度的进给率，单位 mm/rev
SS2	实数	减少主轴转速直到达到最终深度
PSYS	整数	内部参数；只允许默认值 0
VARI	整数	加工方式 值： 0: 退回 SPD 和 SDIS 前移了安全距离的基准面 1: 没有退回终点处
PSYS	整数	内部参数；只允许默认值 1
PSYS	整数	内部参数；只允许默认值 0
AMODE	整数	交替模式：半径或倒角 值： 10000: 半径 11000: 倒角

功能

CYCLE92 循环用于动态切割平衡件（例如，螺钉、螺栓或管道）。

可以在被加工零件的边沿上编写倒角。可以固定切削速率 **V** 或速度 **S** 加工到 **DIAG1** 深度，然后再以固定速度加工工件。也可以从深度 **DIAG2** 编写降低的进给率 **FF2** 或降低的速度 **SS2**，以便使速度适应较小的直径。

用参数 **DIAG2** 输入您希望通过切割所到达的最后深度。比如管道，不需要在中心进行切割，在切割时可以略超过管道的壁厚。

过程

1. 刀具首先快速移动到循环中内部计算的起始点。
2. 以加工进给率加工倒角或直径。
3. 以加工进给率切割到深度 **DIAG1**。
4. 以降低的进给率 **FF2** 和降低的速度 **SS2**，继续切割到深度 **DIAG2**。
5. 刀具以快进速率移回到安全距离。

编程示例 1

N10 G0 G90 Z30 X100 T5 D1 S1000 M3	; 循环开始前的起始点
N20 G95 F0.2	; 确定工艺数值
N30 CYCLE92(60, -30, 40, -2, 2, 1, 800, 200,3,1,1,300, 0, 0, 1, 0, 11000)	; 循环调用
N40 G0 G90 X100 Z30	; 下一个位置
N50 M02	; 程序结束

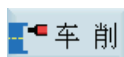
编程示例 2

分离部件的最简单方法是使用 CYCLE92。

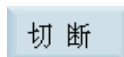
可从车削循环的主画面选择该循环并设置其参数。



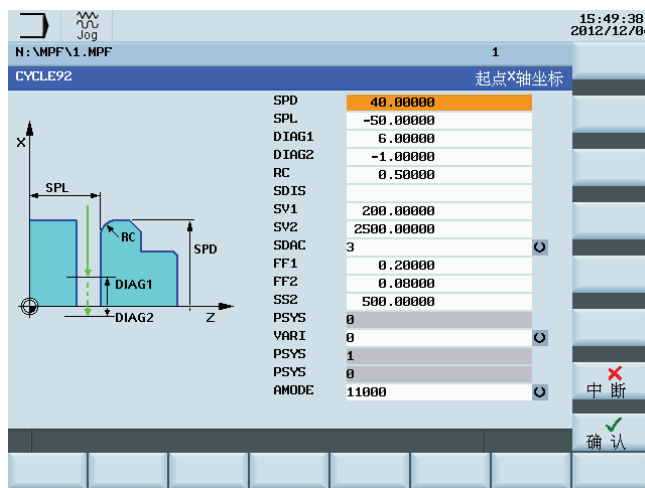
1. 选择所需操作区域。



2. 打开可用车削循环的垂直软键栏。



3. 按下该软键打开 CYCLE 92 的窗口。 根据需求设置循环参数。



4. 使用该软键确认所作的设置。 循环即作为单独程序段被自动传递至程序编辑器。

2.5.3 切槽 - CYCLE93

编程

CYCLE93 (SPD, SPL, WIDG, DIAG, STA1, ANG1, ANG2, RCO1, RCO2, RCI1, RCI2, FAL1, FAL2, IDEP, DTB, VARI, _VRT)

参数

参数	数据类型	说明
SPD	实数	横向轴上的起始点
SPL	实数	纵向轴上的起始点
WIDG	实数	槽宽度（不输入符号）
DIAG	实数	槽深度（不输入符号）
STA1	实数	纵向轴： $0 \leq STA \leq 180$ ，端面轴： $STA=90$
ANG1	实数	啮合角 1: 在通过起始点确定的槽一侧（不输入符号） 值范围： $0 \leq ANG1 < 89.999$ 度
ANG2	实数	啮合角 2: 在另一侧（不输入符号） 值范围： $0 \leq ANG2 < 89.999$
RCO1	实数	半径/倒角 1，外部： 在通过起始点确定的一侧
RCO2	实数	半径/倒角 2，外部
RCI1	实数	半径/倒角 1，内部： 在起始点侧
RCI2	实数	半径/倒角 2，内部
FAL1	实数	切槽底部的精加工余量
FAL2	实数	齿面处的精加工余量
IDEP	实数	进刀深度（不输入符号）
DTB	实数	切槽基础处的停留时间
VARI	整数	加工方式 值范围： 1...8 和 11...18
_VRT	实数	轮廓的可变退回距离，增量（不输入符号）

功能

切削循环可以在任意一段直线轮廓上加工出横向或纵向、对称或不对称的凹槽。可以产生外部和内部凹槽。

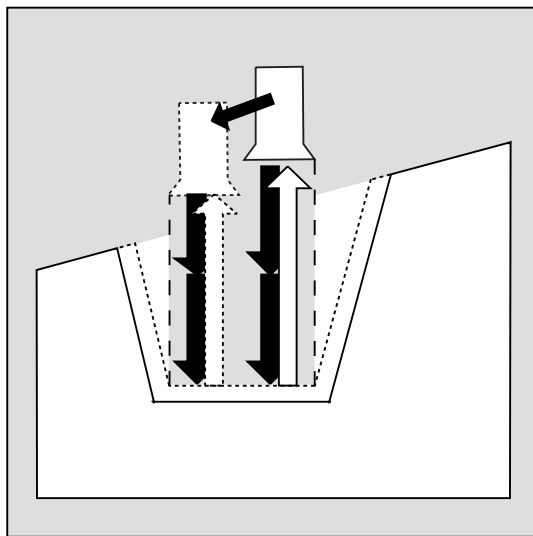
过程

循环内部会自动计算出刀具在深度（到切槽底部）和宽度（从切槽一侧到另一侧）方向的总进刀量，然后均匀划分该值并使每次的进刀量尽可能地大

在一个斜面上切槽时，刀具会以最短行程从一个槽逼近下一个槽，即平行于切槽加工所在的锥面。在这个过程中，在循环内部计算轮廓的安全距离。

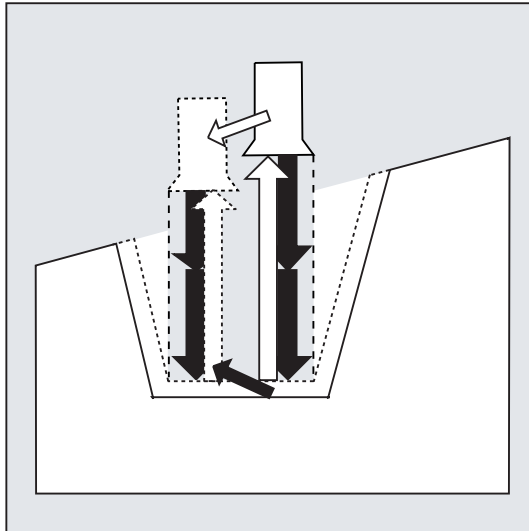
第 1 步

刀具在深度方向分为几步平行于轴进行粗加工，一直达到槽底
每次进刀后，退回刀具以断屑。



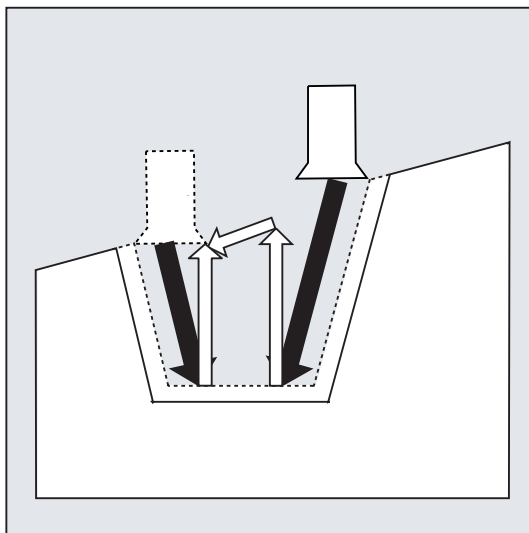
第 2 步

刀具在宽度方向上一步完成切削或分多步切削。从沿着槽宽向前的二次切割，刀具将在每次退回前退回 1 毫米。



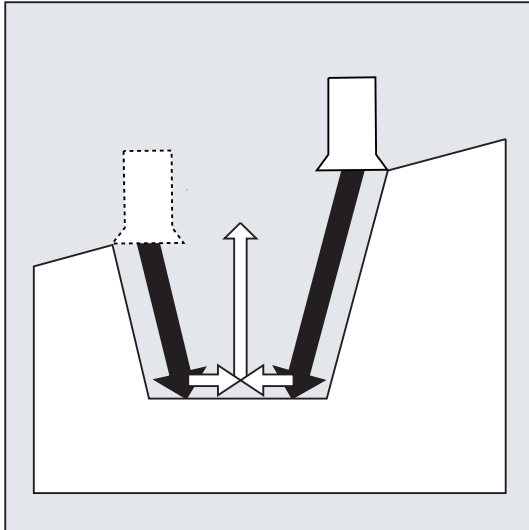
第 3 步

如果在 ANG1 或 ANG2 下编程角度，则在一个步骤中加工齿面。如果齿面宽度较大，则在多个步骤中执行沿着槽宽的进刀。



第 4 步

精加工余量的横向轴切削与从边缘到槽心的轮廓平行。在这个操作过程中，刀具半径补偿由循环自动选择和取消。



参数说明

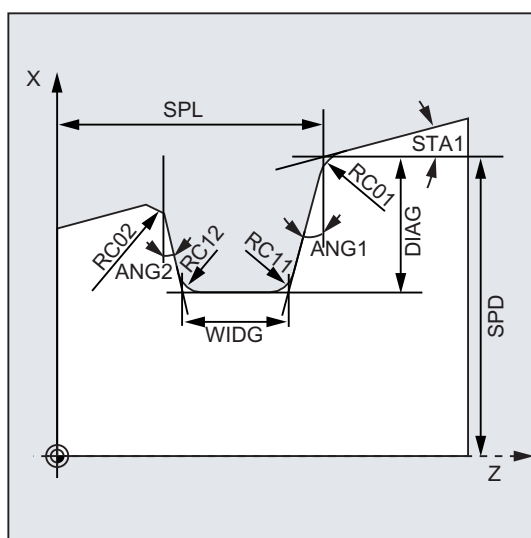
SPD 和 SPL（起始点）

"这两个坐标用于定义切槽起点，循环从该起点计算出切槽的形状。循环会自动计算出轴需要逼近的起点位置。对于外部切槽，运动在纵向轴方向上开始，对于内部切槽，运动在端面轴方向上开始。

弯曲轮廓上的切槽可以采用不同加工方式。取决于该弯曲轮廓的形状和弯曲半径，可以在最高点上方设置一条平行于轴的直线，或者在切槽边缘点上设置一条相切的斜线。

只有当边缘点位于循环中指定的直线上时，才需要在弯曲轮廓的槽边缘上设计倒角和倒圆。

有关 CYCLE93 参数的描述，见下图：



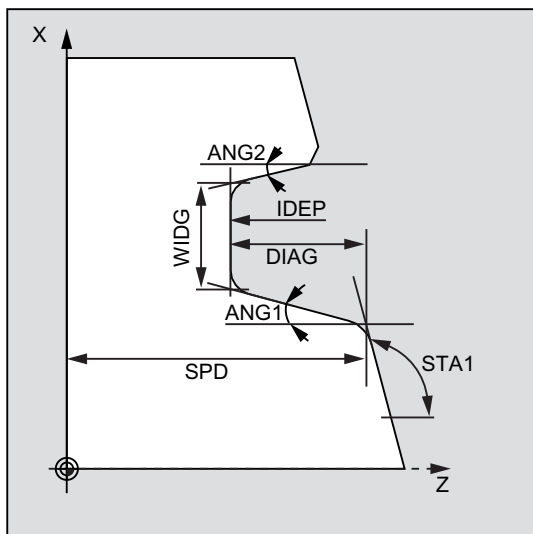
WIDG 和 DIAG（槽宽和槽深）

槽宽 (WIDG) 和槽深 (DIAG) 参数用于定义槽形。在计算中，循环始终假设在 SPD 和 SPL 下编程的点。

如果槽宽大于有效刀具宽度，则分几个步骤减少宽度。执行该操作时，通过循环均等地分配整个宽度。减去刀沿半径后，最大进刀为刀具宽度的 95%。以此获取切削重叠。

如果编程的槽宽小于真实的刀具宽度，将输出错误消息 61602“刀具宽度定义错误”并中断加工。如果在循环中检测到刀沿宽度等于零，也将出现报警。

2.5 车削循环



G90G95G18; Z/X 轴面的绝对尺寸，旋转进给率

T8; 刀具调用

M01; 可选停止

M3S1000; 主轴转速

M08; 冷却液 ON

G0X50Z10; 循环开始前的起始点

G1F0.1; 确定工艺数值

CYCLE93 (30.00000, -24.00000, 7.00000, 5.00000, , , , 1.00000, 1.00000, , , 0.20000, 0.20000, 1.50000, 0.20000, 5, 1.00000); 循环调用

G0X50

Z100; 退回安全位置

M9; 冷却液 OFF

STA1 (角度)

使用 STA1 参数编程斜线的角度，在该斜线上将加工切槽。角度取值的范围可以为 0 ~ 180 度，并且始终指的是纵向轴。

说明

对于运行切槽，角 STA1 通常为 90 度（为傍轴时）。

ANG1 和 ANG2 (啮合角)

单独指定侧面角可以加工出不对称的切槽。角度取值的范围可以为 0 ~ 89.999 度。

RCO1、RCO2 和 RCI1、RCI2 (半径/倒角)

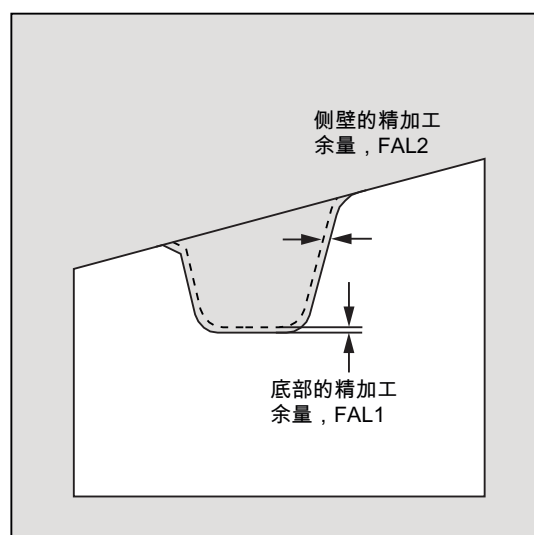
指定切槽边缘或底部上的倒角或倒圆可以修改切槽的形状。指定时注意倒圆要带正号、倒角要带负号。

VARI 的十位用于确定指定倒角的方式。

- 当 VARI<10 (十位 =0) 时, 使用 CHF=... 指定倒角
- VARI>10 时, 使用 CHR 编程倒角

FAL1 和 FAL2 (精加工余量)

可以为切槽基础和齿面编程单独的精加工余量。在粗加工过程中, 到达这些精加工余量后执行切削。然后, 同一刀具用于沿着最终轮廓加工轮廓平行切削。



IDEP (进刀深度)

通过编程进刀深度, 可以将傍轴切槽分为多个深度进刀。每次进刀后, 刀具退回 1 毫米以断屑。

在所有情况下, 必须编程 IDEP 参数。

DTB (停留时间)

应选择切槽基础处的停留时间, 从而至少执行一个主轴转数。以秒为单位编程。

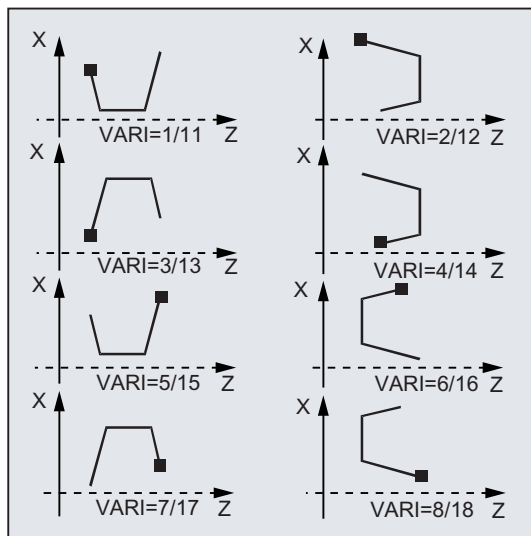
VARI (加工方式)

参数 VARI 的个位定义切槽的加工方式。可以采用图中指示的值。

参数 VARI 的十位数字确定如何考虑倒角。

VARI 1...8: 将倒角作为 CHF 计算

VARI 11...18: 将倒角作为 CHR 计算



如果该参数具有不同的值，则该循环中断，并输出报警 61002“加工方式定义错误”。

循环执行轮廓监控，从而得出合理的切槽轮廓结果。如果半径/倒角接触或在切槽基础相交，或者如果您尝试在与纵向轴平行的轮廓段处执行端面切槽操作，则不是这种情况。在此类情况下，循环中断，并输出报警 61603“槽形定义错误”。

_VRT (可变的退回位移)

可以基于切槽的外径和内径在 _VRT 参数中编程退回位移。

在 VRT=0 (参数未编程) 时，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。

相同的退回位移也用于切槽每次深度进刀后的断屑。

说明

调用切槽循环之前，必须已激活一把双刀沿刀具。"这两个刀沿的补偿值必须保存在该刀具的两个连续 D 号中，在调用循环前必须将其中第一个 D 号激活。"循环会自动确定哪个工序使用哪个刀具补偿，然后自动激活该补偿。"循环完成后，再次激活循环调用前编程的刀具补偿编号。如果调用循环时没有编程刀具补偿的 D 编号，则使用报警 61000“没有激活刀具补偿”中断循环的执行。

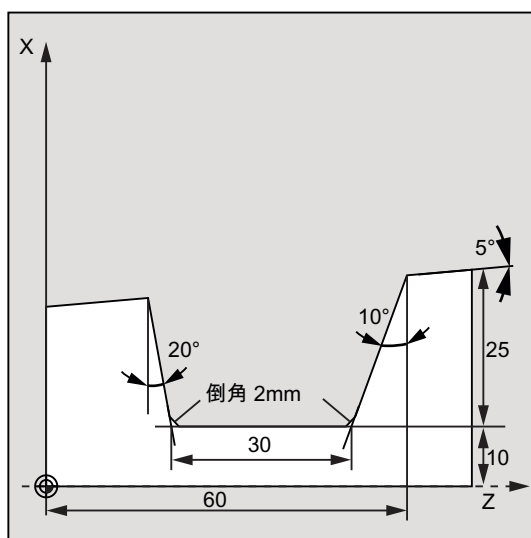
编程示例 1：切入车削

通过该程序在一个斜面上加工出一个纵向外部切槽。

起始点在 X35 Z60 处的右侧。

循环将使用刀具 T5 的刀具补偿 D1 和 D2。必须相应地定义切削刀具。

有关切入车削参数的描述，见下图：

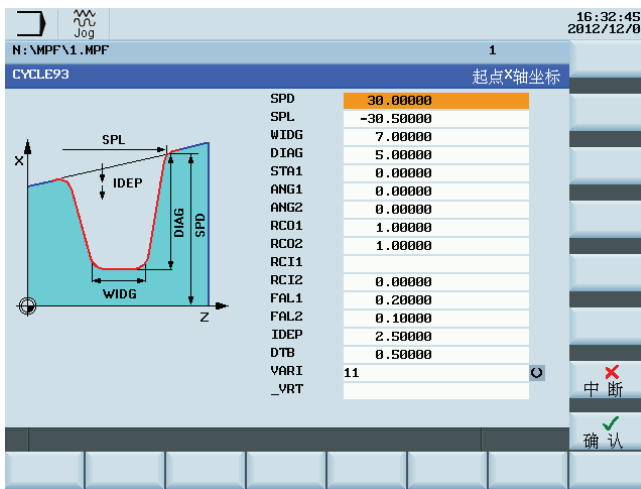


N10 G0 G90 Z65 X50 T5 D1 S400 M3	; 循环开始前的起始点
N20 G95 F0.2	; 确定工艺数值
N30 CYCLE93(35, 60, 30, 25, 5, 10, 20, 0, 0, -2, -2, 1, 1, 10, 1, 5, 0.2)	; 循环调用 编程的退回距离为 0.2 毫米
N40 G0 G90 X50 Z65	; 下一个位置
N50 M02	; 程序结束

编程示例 2



1. 选择所需操作区域。
2. 打开可用车削循环的垂直软键栏。
3. 按下该软键打开 CYCLE93 的窗口。 根据需求设置循环参数。



4. 使用该软键确认所作的设置。 循环即作为单独程序段被自动传递至程序编辑器。

2.5.4 退刀槽（形状 E 和 F，符合 DIN） - CYCLE94

编程

CYCLE94(SPD, SPL, FORM, VARI)

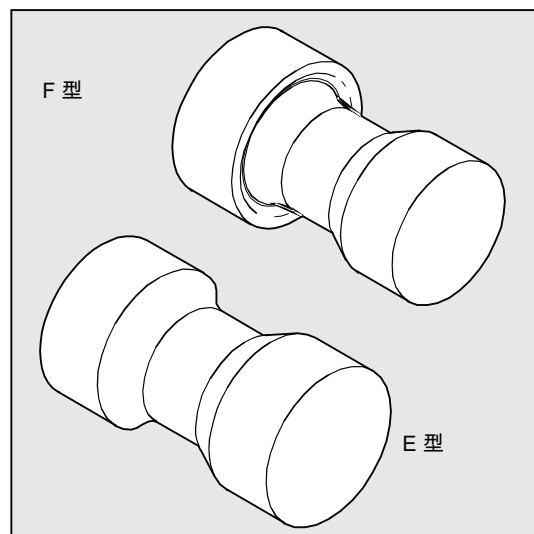
参数

参数	数据类型	说明
SPD	实数	横向轴上的起始点（不输入符号）
SPL	实数	纵向轴上刀具补偿的起始点（不输入符号）
FORM	CHAR	形状定义 值： E（E 型），F（F 型）
VARI	整数	确定退刀槽的位置 值： 0: 根据刀具的刀沿位置 1..4: 位置定义

功能

使用该循环，可以根据 DIN509 工件上加工成品直径 >3 毫米，满足标准应力需求的 E 型和 F 型退刀槽。

有关退刀槽形状 F 和形状 E 的描述，见下图：



过程

循环开始之前到达的位置:

起始位置可为能够无碰撞逼近退刀槽的任意位置。

该循环产生以下的运动过程:

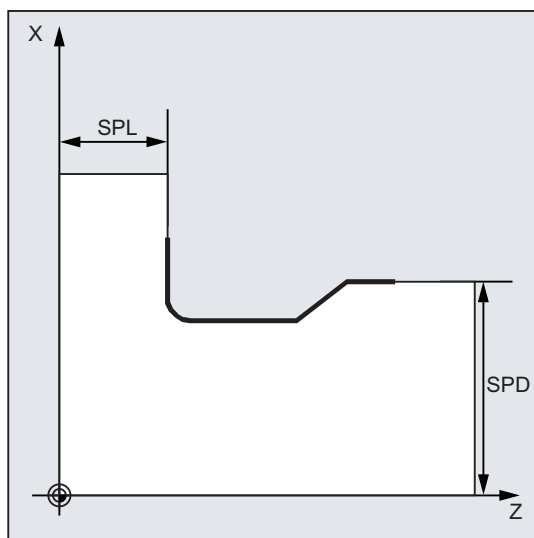
- 使用 G0 逼近循环内部确定的起始点
- 根据当前的刀沿位置选择刀沿半径补偿，并且以循环调用之前编程的进给率逼近退刀槽轮廓
- 使用 G0 退回到起始点，并使用 G40 取消刀沿半径补偿

参数说明

SPD 和 SPL (起始点)

在参数 SPD 下设定退刀槽的成品直径。通过参数 SPL 设定纵向轴上的成品尺寸。

如果根据 SPD 编程的值产生的成品直径 <3 毫米，则该循环中断，并输出报警 61601 “成品直径过小”。

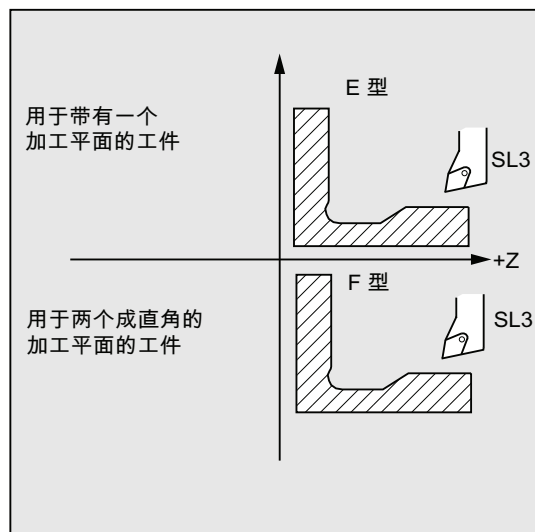


FORM (定义)

DIN509 中对 E 型和 F 型退刀槽进行了定义，使用此参数可以确定加工哪一种退刀槽。

如果参数赋值不为 E 或者 F，则循环中断，并输出报警 61609 “形状定义错误”。

有关形状 E 和形状 F 的描述，见下图：



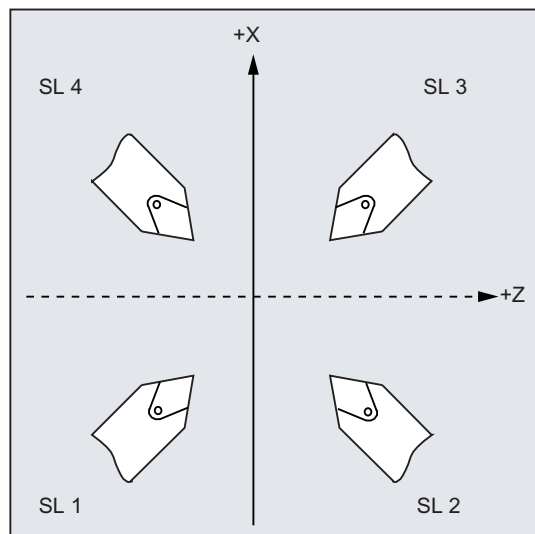
VARI (退刀槽位置)

使用参数 `_VARI` 可以直接确定退刀槽的位置，或者由刀具的刀沿位置产生退刀槽位置。

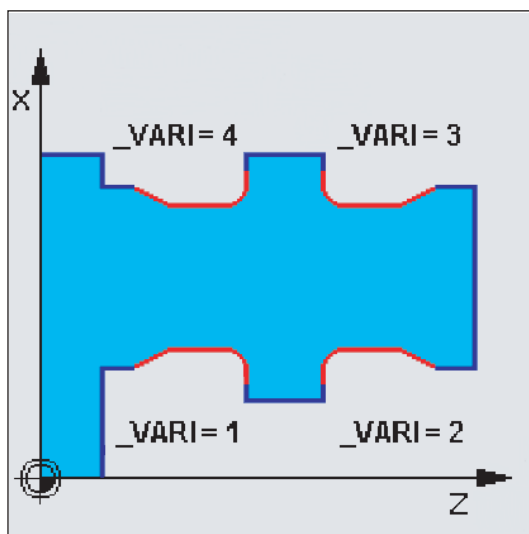
VARI=0: 根据刀具的刀沿位置

循环通过生效的刀具补偿值自动确定刀具的刀沿位置。循环可用刀沿位置 1 ... 4 运行。

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”，且循环中断。



VARI=1...4: 定义退刀槽位置



在 VARI<>0 时:

- 不检查实际刀具刀沿位置，即在工艺上可行的情况下可使用所有刀沿。

如果在相应的刀具补偿值参数中设定了数值，则在循环中对生效刀具的自由切削角进行监控。如果因为所选刀具的自由切削角过小而无法加工退刀槽形状，则在控制系统中显示信息“更改的退刀槽形状”。加工会继续进行。

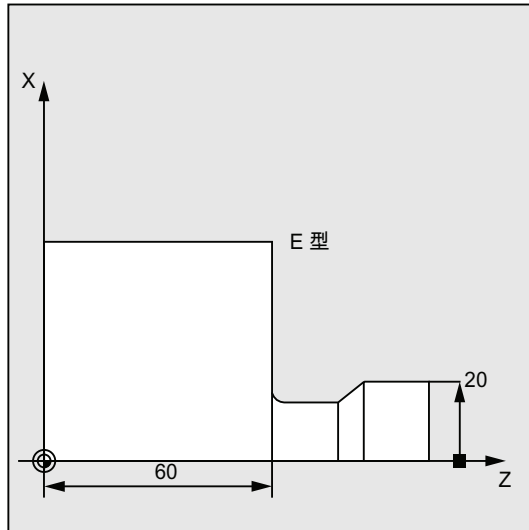
循环自动确定起始点。它位于纵向轴上距离最终直径 2 毫米和距离最终尺寸 10 毫米处。通过有效刀具的刀沿位置确定用于已编程的坐标值的起始点位置。

说明

在调用循环之前，必须激活刀具补偿；否则，输出报警 61000 “没有激活刀具补偿”后中断循环。

编程示例：E 型退刀槽

使用该程序加工 E 型退刀槽。



```
N10 T1 D1 S300 M3 G95 F0.3  
N20 G0 G90 Z100 X50  
N30 CYCLE94 (20, 60, "E", )  
N40 G90 G0 Z100 X50  
N50 M02
```

; 确定工艺数值
; 选择起始位置
; 循环调用
; 逼近下一个位置
; 程序结束

2.5.5 毛坯切削，带底切 - CYCLE95

编程

CYCLE95(NPP, MID, FALZ, FALX, FAL, FF1, FF2, FF3, VARI, DT, DAM, _VRT)

参数

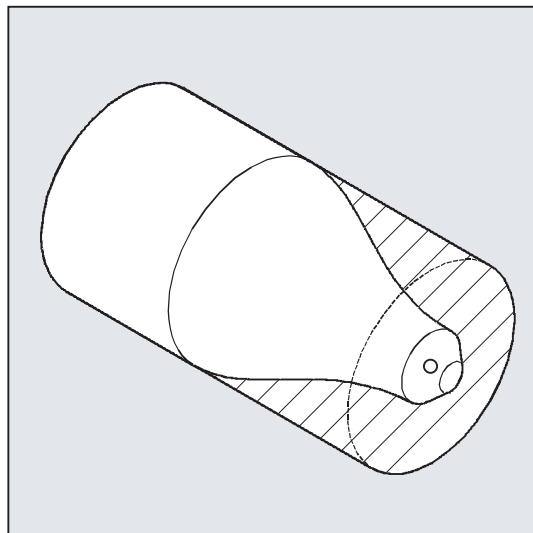
参数	数据类型	说明
NPP	STRING	轮廓子程序名
MID	实数	进刀深度（不输入符号）
FALZ	实数	纵向轴上精加工余量（不输入符号）
FALX	实数	横向轴上精加工余量（不输入符号）
FAL	实数	轮廓精加工余量（不输入符号）
FF1	实数	粗加工进给，无底切
FF2	实数	在底切时插入进给
FF3	实数	精加工进给
VARI	实数	加工方式 值范围： 1 ... 12
DT	实数	粗加工时用于断屑的停留时间
DAM	实数	位移长度，每次粗加工切削断屑时均中断该长度
_VRT	实数	粗加工时从轮廓的退刀位移，增量（不输入符号）

功能

通过切削循环，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。使用该循环，可在纵向加工和横向加工中内外加工轮廓。可自由选择工艺（粗加工、精加工、完全加工）。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，在达到切削点后，立即用轮廓产生的余角以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和取消。

有关 CYCLE95 的描述，见下图：



过程

循环开始之前到达的位置：

起始位置可为能无碰撞逼近轮廓起始点的任意位置，。

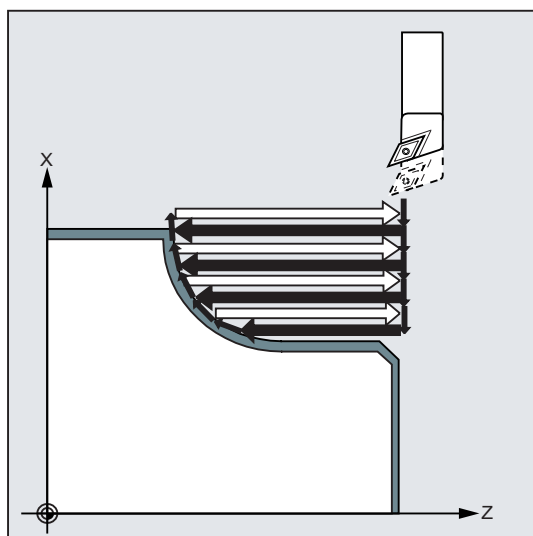
该循环产生以下的运动过程：

循环起始点在内部计算，并使用 G0 在两个轴中同时逼近。

粗加工，不带底切单元：

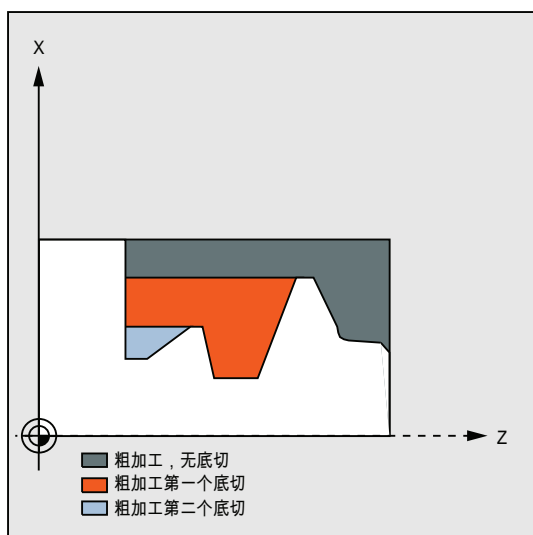
- 内部计算至当前深度的轴向平行进给，并使用 G0 逼近。
- 使用 G1 和进给率 FF1 轴向平行逼近粗加工切削点。
- 使用 G1/G2/G3 和 FF1 沿着轮廓+精加工余量并与轮廓平行进行进刀。
- 在每个轴上退回 _VRT 下编程的量，并以 G0 返回。
- 重复该过程，直至到达加工截面的总深度。
- 在粗加工不带底切单元时，按轴方式退回到循环起始点。

有关 CYCLE95 操作顺序的描述，见下图：



粗加工，带底切单元：

- 使用 G0 一根轴接一根轴地接近下一个底切的起始点。执行该操作时，遵守附加循环内部安全间距。
- 使用 G1/G2/G3 和 FF2 沿着轮廓+精加工余量并与轮廓平行进行进刀。
- 使用 G1 和进给率 FF1 轴向平行逼近粗加工切削点。
- 沿轮廓进行拉削，像第一步一样执行退刀和返回。
- 如果有其它的底切单元，则对每个底切重复执行此过程。



精加工：

- 使用 G0 以轴方式逼近循环起始点。
- 两个轴同时通过 G0 逼近轮廓起始点。
- 沿着轮廓使用 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）

在该参数中输入轮廓的名称。 可将轮廓定义为子程序或所调用程序的一部分。

- 将轮廓定义为子程序

NPP = 子程序的名称

- 如该子程序已存在，输入名称，继续。
- 该子程序不存在，输入名称，按下列软键：

新建

创建名称为输入名称的程序，并返回轮廓编辑器。

- 按下列软键确认输入并返回该循环的屏幕：

工艺
界面

- 将轮廓定义为所调用程序的一部分

NPP = 起始标签的名称 输入结束标签的名称

输入：

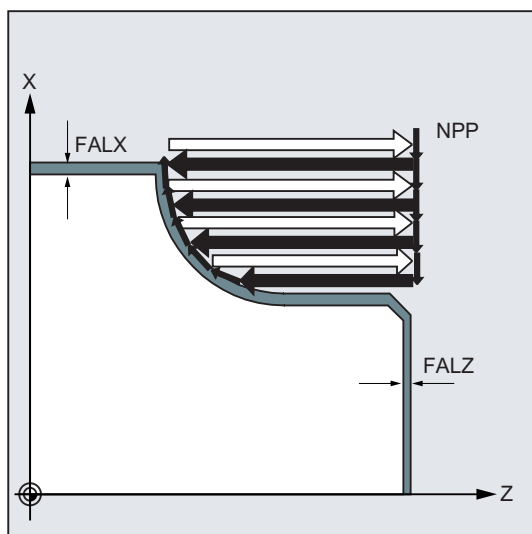
- 轮廓未说明，则输入起始标签的名称并按下列软键： 如轮廓已说明（起始标签的名称： 输入结束标签的名称），直接按下列软键：

轮廓
附加

控制系统从输入的名称自动创建起始标签和结束标签，程序跳转到轮廓编辑器。

- 按下列软键确认输入并返回该循环的屏幕：

工艺
界面



示例：

```
NPP=CONTOUR_1
NPP=START:END
```

；切削轮廓是完整的程序 CONTOUR_1。
；切削轮廓定义为所调用程序的一部分，开始程序段为标签 START，结束程序段为标签 END。

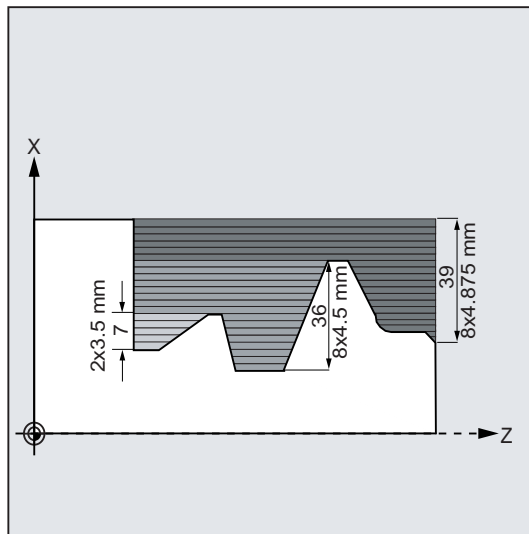
MID (进刀深度)

在参数 MID 下定义粗加工过程最大可能的进刀深度。

该循环自行计算进行粗加工时实际的进刀深度。

对于带有底切单元的轮廓，循环将粗加工过程分割成单独的粗加工部分。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。根据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部分的数量并平均分配待加工的总深度。以此获取最佳的切削条件。下图显示了轮廓粗加工的加工步骤。

参见以下进刀深度的示例：



加工部分 1 具有 39 毫米的总深度。如果最大进刀深度为 5 毫米，则需要八次粗加工切削。每次进刀为 4.875 毫米。

加工步骤 2 同样需要 8 次粗加工切削，每次进刀为 4.5 毫米（总距离 36 毫米）。

在加工步骤 3 中，执行两次粗加工切削，每次进刀 3.5 毫米（总距离 7 毫米）。

FAL、FALZ 和 FALX（精加工余量）

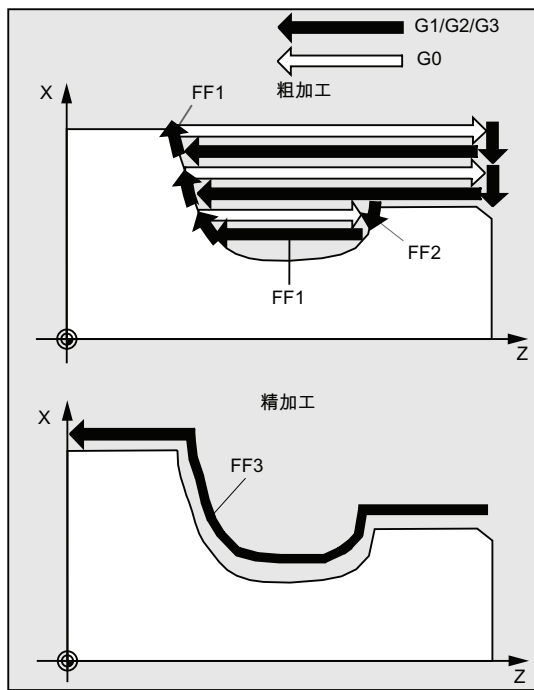
如需为不同的轴设定不同的精加工余量，可通过参数 **FALZ** 和 **FALX** 分别设定，或者可以通过参数 **FAL** 设定轮廓精加工余量。该值在两个轴中都被作为精加工余量。

不对编程的值进行合理性检测。换句话说：如果已为所有三个参数赋值，则循环会考虑所有精加工余量。因此应当决定一种方式来定义精加工余量。

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平于轮廓切削加工中产生的余角，从而在粗加工结束后无需附加的余角切削。如果未编程精加工余量，则在粗加工时一直切削到最终轮廓。

FF1, FF2 和 FF3 (进给率)

对于不同的加工步骤，可以设定不同的进给率，如图 NO TAG 中所示。



VARI (加工方式)

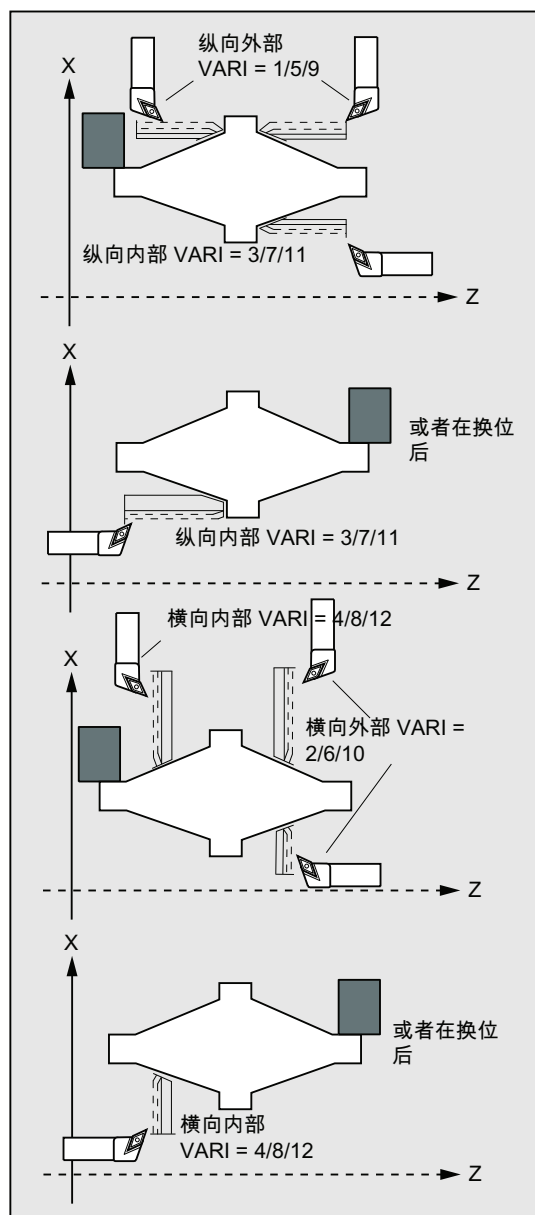
加工方式可在下表中找到。

值	纵向/横向	外部/内部	粗加工/精加工/完全加工
1	L	O	粗加工
2	P	O	粗加工
3	L	I	粗加工
4	P	I	粗加工
5	L	O	精加工
6	P	O	精加工
7	L	I	精加工
8	P	I	精加工
9	L	O	完整加工
10	P	O	完整加工
11	L	I	完整加工
12	P	I	完整加工

纵向加工时总是在横向轴上进刀，横向加工时总是在纵向轴上进刀。

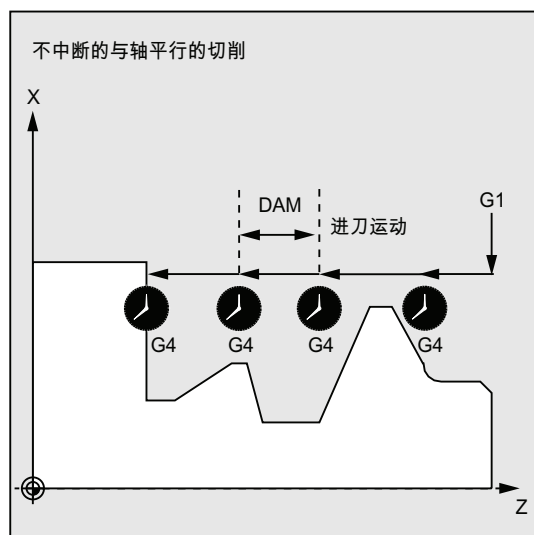
外加工表示，以轴的负方向进刀。在内加工时，进刀以轴的正方向进行。

对于参数 **VARI** 可以进行合理性检测。在循环调用时，如果值不在值域 1 ... 12 中，则该循环中断，并发出报警 61002 “加工方式定义错误”。



DT 和 DAM（停留时间和位移长度）

通过这两个参数，可以在特定行程后中断粗加工切削进行断屑。这两个参数仅用于粗加工。在参数 DAM 中定义最大行程，在此行程后执行断屑。为此在 DT 中可以编程在每个执行中断点执行的停留时间（单位秒）。如果未设定用于切削中断的行程（DAM=0），则执行无停留时间的不中断粗加工切削。



_VRT（退刀位移）

在参数 _VRT 下可以编程粗加工时两个轴上的退刀量。

在 _VRT=0（参数未编程）时，退刀 1 毫米。

轮廓定义

轮廓必须至少包含 3 个程序段，它们在加工平面的两个轴上产生运动。

如果轮廓太短，则输出报警 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”，且循环中断。

底切单元可以直接连续排列。不包含平面中的运动的程序段可以无限制写入。

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。

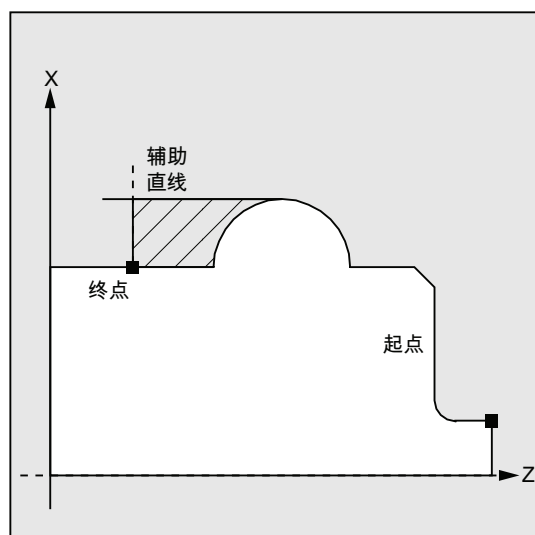
作为轮廓中的几何尺寸，仅允许通过 G0、G1、G2 和 G3 进行直线编程和圆弧编程。此外，也可以编程用于倒圆和倒角的指令。如果在轮廓中编程了其它的运动指令，则循环中断，并输出报警 10930“切削轮廓中有不允许的插补方式”。

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，并输出报警 15800“错误的 CONTPRON 输出条件”。如果 G41/42 有效，也会发出该报警。轮廓的起始点是加工平面中第一个编程位置。

必须准备可存储最大数量轮廓元素的循环内部存储器用于执行编程的轮廓；数量取决于轮廓。如果轮廓包含的轮廓元素过多，则输出报警 10934“轮廓表溢出”且循环中断。该轮廓必须划分为多个轮廓段，循环分别针对每个段进行调用。

如果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。

有关轮廓定义的描述，见下图：



使用 G41/G42 在轮廓子程序中编程刀具半径补偿会导致循环中断，并输出报警 10931“错误的切削轮廓”。

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。在完全加工时，轮廓的精加工方向与粗加工时相同。

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。因此在轮廓子程序的第一个程序段中总是需要给定两个坐标。

轮廓监控

循环提供轮廓监控，包含以下几点：

- 有效刀具的自由切削角
- 圆弧的轮廓编程，张角 > 180 度

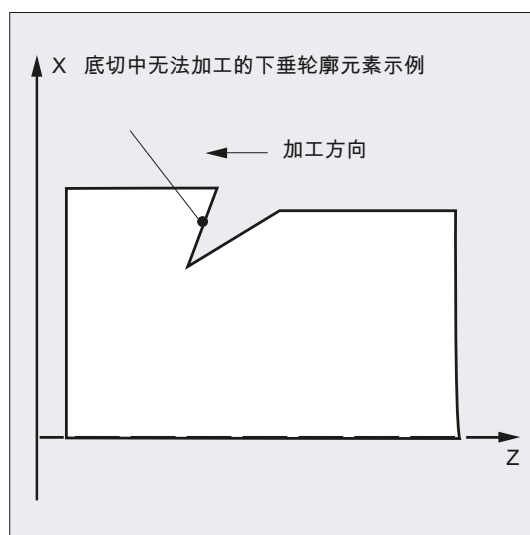
对于底切单元，在循环中检查是否可以用激活的刀具进行加工。如果循环识别到该加工可能导致轮廓损坏，则在输出报警 61604“当前刀具会损坏编程的轮廓”后，循环中断。

如果自由切削角在刀具补偿中设定为零，则不执行监控。

如果在补偿中发现过大的圆弧，则显示报警 10931“错误的切削轮廓”。

不能使用 CYCLE95 加工下垂的轮廓。此类轮廓不受循环监控，因此不会输出报警信息。

有关轮廓监控的描述，见下图：



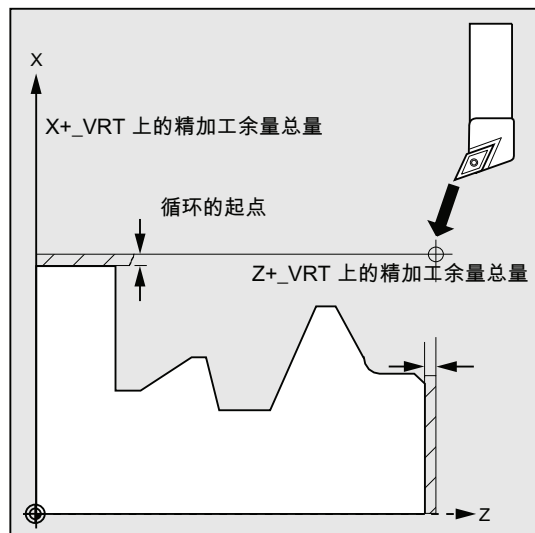
起始点

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，起始点位于轮廓起始点前的量为精加工余量 + `_VRT`。

在逼近起始点时，循环内部选择刀沿半径补偿。

因在选择调用循环之前最后的点时，必须确保不会发生碰撞并且有足够的空间用于相应的补偿运动。

有关起始点的描述，见下图：



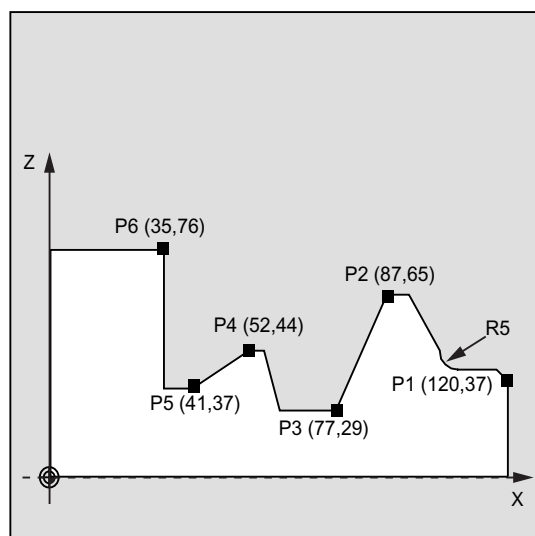
循环逼近方案

在粗加工时总是通过两根轴同时逼近通过循环确定的起始点，在精加工时总是按轴方式逼近。在精加工时首先运行进给轴。

编程示例 1：切削循环

图中显示的轮廓需要进行完全、纵向、外部加工。设定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。

轮廓保存在单独的程序中。



2.5 车削循环

以下为主程序:

```

N10 T1 D1 G0 G95 S500 M3 Z125 X81 ; 调用之前的逼近位置
N20 CYCLE95("CONTOUR_1", 5, 1.2, 0.6, , 0.2, ; 循环调用
0.1, 0.2, 9, , , 0.5)
N30 G0 G90 X81 ; 重新逼近起始位置
N40 Z125 ; 轴方式运行
N50 M2 ; 程序结束
    
```

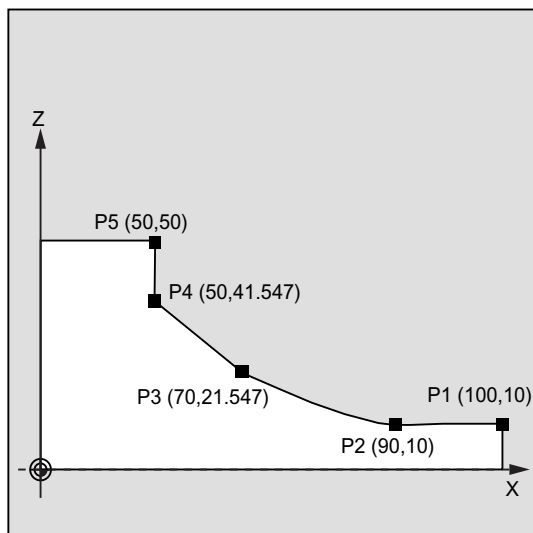
以下为子程序:

```

CONTOUR_1.SPF ; 轮廓车削用子程序(例子)
N100 Z120 X37 ; 轴方式运行
N110 Z117 X40
N120 Z112 RND=5 ; 以半径 5 倒圆
N130 Z95 X65 ; 轴方式运行
N140 Z87
N150 Z77 X29
N160 Z62
N170 Z58 X44
N180 Z52
N190 Z41 X37
N200 Z35
N210 X76
N220 M02 ; 子程序程序结束
    
```

编程示例 2: 切削循环

切削轮廓在调用的程序中定义，并在精加工循环调用后直接运行。



```

N110 G18 DIAMOF G90 G96 F0.8
N120 S500 M3
N130 T1 D1
N140 G0 X70
N150 Z160
N160 CYCLE95 ("START:END",2.5,0.8,          : 循环调用
0.8,0,0.8,0.75,0.6,1, , , )
N170 G0 X70 Z160
N175 M02
ANFANG:
N180 G1 X10 Z100 F0.6
N190 Z90
N200 Z70 ANG=150
N210 Z50 ANG=135
N220 Z50 X50
ENDE:
N230 M02
    
```

编程示例 3

按照以下步骤进行：



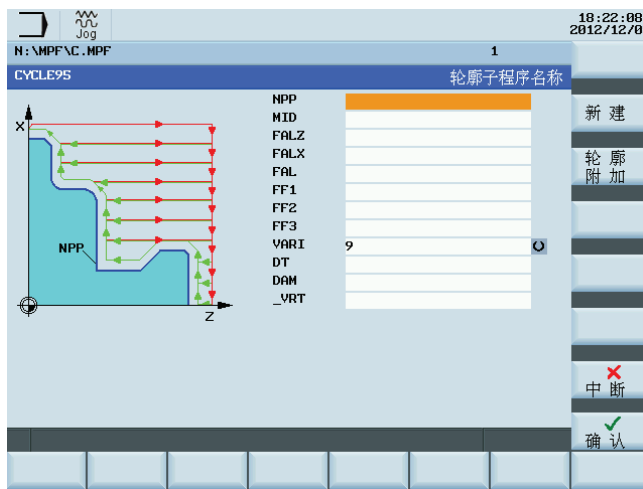
1. 选择所需操作区域。



2. 打开可用车削循环的垂直软键栏。



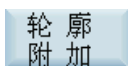
3. 按下该软键打开 CYCLE95 的窗口。在第一个输入区域输入名称。



4. 按下以下任一软键。程序自动跳转到程序编辑器的屏幕。



如需编辑并存储子程序轮廓，按下该软键。



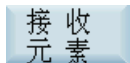
如需编辑并存储主程序轮廓，按下该软键。



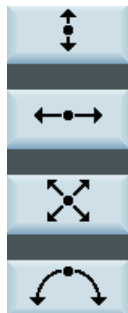
5. 按下该软键打开程序编辑器的窗口。逐步设置轮廓元素的参数。首先定义一个轮廓起始点并选择如何逼近该起始点。

说明：

步骤 5 到 10 描述轮廓元素编辑的基本步骤。更多关于轮廓编辑器的编程信息，参见 SINUMERIK 808D 编程和操作手册车削（第 1 部分）。

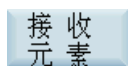
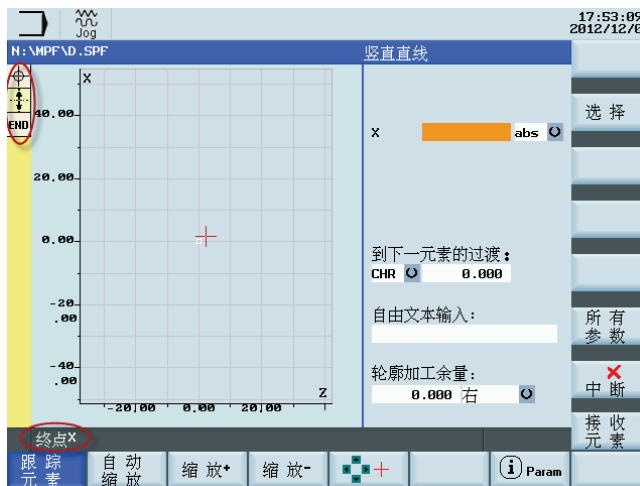


6. 按下该软键确认所作设置。



7. 使用相关软键选择所需加工方向及形状。根据加工图输入对应的坐标参数。

屏幕左上显示所选方向，屏幕下方的信息行显示对应的描述。



8. 按下该软键确认所作设置。

9. 选择不同元素定义轮廓，直到完成整个轮廓形状的编程。



10. 按下该软键保存轮廓信息。



11. 按下该软键，返回 CYCLE95 的屏幕。根据需求设置循环工艺参数。



12. 使用该软键确认所作的设置。循环被自动传递到程序编辑器。

说明：

所创建的循环程序必须作为主程序的一部分保存在 M30 指令后。



13. 如需再编译该循环，按下该软键。

腔轮廓

下例将说明如何处理腔轮廓和如何通过轮廓功能定义轮廓的子程序。

G500 G18 G95

G0X50

Z100

T5

M4S1500

G0X50Z0.5

2.5 车削循环

```
G01X-2F0.15
Z2
G0X50
Z100
T2
G0X50Z10
M4S1500
G1F0.2
CYCLE95("CON01", 0.50000, 0.20000, 0.20000, 0.20000, 0.20000, 0.20000, 0.10000,
1, , ,1.00000)
G0X55
Z100
M5
T1
M3S1500
G0X50Z10
CYCLE95("CON02:CON02_E", 0.50000, 0.20000, 0.20000, 0.20000, 0.20000, 0.20000,
0.10000, 5, , ,1.00000)

M30
,*****轮廓*****
CON02:

;#7__DlGK contour definition begin - Don't change!;*GP*;*RO*;*HD*
G18 G90 DIAMON;*GP*
G0 Z0 X0 ;*GP*
G1 X28 CHR=3 ;*GP*
Z-8.477 RND=2 ;*GP*
G2 Z-45.712 X40 K=AC(-25) I=AC(60) RND=2 ;*GP*
```



```
G1 Z-50 RND=3 ;*GP*
Z-55 X45 ;*GP*
;CON,V64,2,0.0000,0,0,MST:1,2,AX:Z,X,K,I;*GP*;*RO*;*HD*
;S,EX:0,EY:0,ASE:0;*GP*;*RO*;*HD*
;LU,EY:28;*GP*;*RO*;*HD*
;F,LFASE:3;*GP*;*RO*;*HD*
;LL;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;ACW,DIA:209/217,EY:40,CX:-25,CY:60,RAD:23;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;LL,EX:-50;*GP*;*RO*;*HD*
;R,RROUND:3;*GP*;*RO*;*HD*
;LA,EX:-55,EY:45;*GP*;*RO*;*HD*
;#End contour definition end - Don't change!;*GP*;*RO*;*HD*
M17

CON02_E.;***** 轮廓终点 *****
```

2.5.6 螺纹退刀槽 - CYCLE96

编程

CYCLE96 (DIATH, SPL, FORM, VARI)

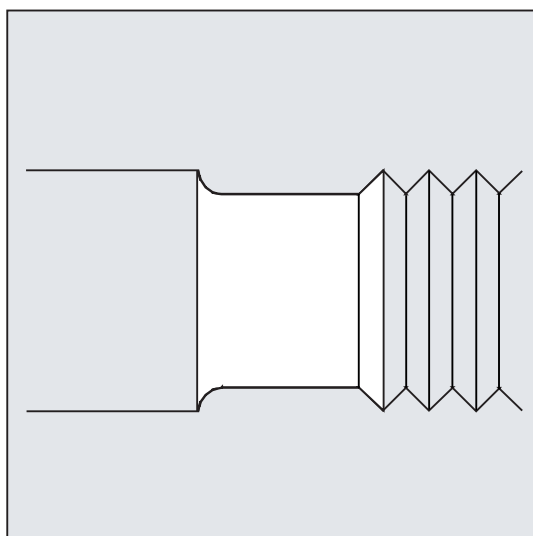
参数

参数	数据类型	说明
DIATH	实数	螺纹额定直径
SPL	实数	纵向轴上补偿起始点
FORM	CHAR	形状定义 值: A (A型), B (B型), C (C型), D (D型)
VARI	整数	确定退刀槽的位置 值: 0: 根据刀具的刀沿位置 1..4: 位置定义

功能

使用此循环，可以根据 DIN76 加工螺纹退刀槽，用于公制 ISO 螺纹零件。

有关 CYCLE96 的描述，见下图：



过程

循环开始之前到达的位置：

起始位置可为能够无碰撞逼近螺纹退刀槽的任意位置。

该循环产生以下的运动过程：

- 使用 G0 逼近循环内部确定的起始点
- 根据有效刀沿位置选择刀具半径补偿。使用循环调用前编程的进给沿退刀槽轮廓运行
- 使用 G0 退回到起始点，并用 G40 撤消刀具半径补偿

参数说明

DIATH（额定直径）

使用该循环，可以加工螺纹退刀槽，用于公制 ISO 螺纹 M3 到 M68。

如果根据 DIATH 编程的值产生的最终直径 <3 毫米，则该循环中断，并输出报警：

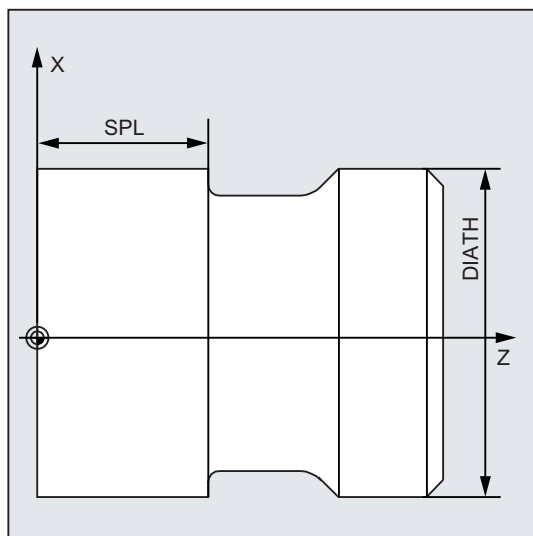
61601“成品直径过小”。

如果该参数的值不同于 DIN76 第 1 部分所规定的值，则循环中断，并输出报警：

61001“螺距定义错误”。

SPL（起始点）

使用参数 SPL 确定纵向轴的成品尺寸。

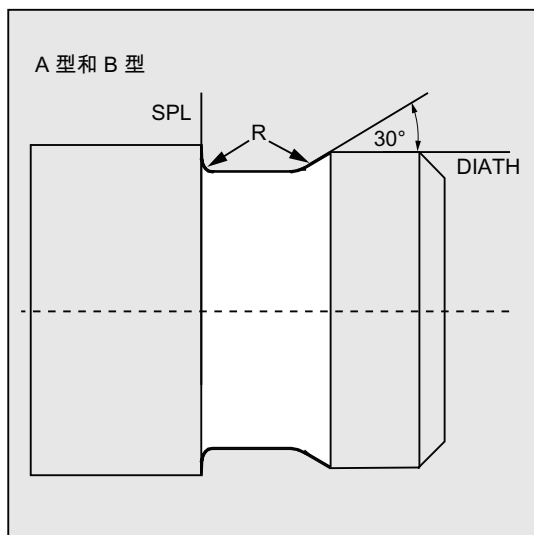


FORM (定义)

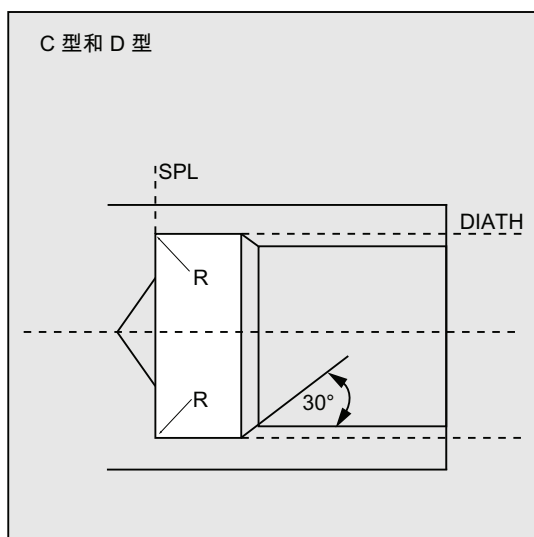
A 型和 B 型螺纹退刀槽用于外螺纹，A 型为正常螺纹收尾，B 型为较短的螺纹收尾。

C 型和 D 型螺纹退刀槽用于内螺纹，C 型为正常螺纹收尾，D 型为较短的螺纹收尾。

有关形状 A 和形状 B 的描述，见下图：



有关形状 C 和形状 D 的描述，见下图：



如果该参数的值不同于 A...D，则该循环中断，并输出报警 61609 “形状定义错误”。

循环内部自动择刀具半径补偿。

循环只能通过刀沿位置 1 ... 4 运行。如果该循环识别的刀沿位置为 5...9 或者无法使用所选择的刀沿位置加工退刀槽形状，则会显示报警 61608 “编程了错误的刀沿位置”，并且循环中断。

VARI (退刀槽位置)

使用参数 **VARI** 可以直接确定退刀槽的位置，或者由刀具的刀沿位置产生退刀槽位置。
另见章节 "退刀槽 (形状 E 和 F, 符合 DIN) - CYCLE94 (页 179)"。

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。

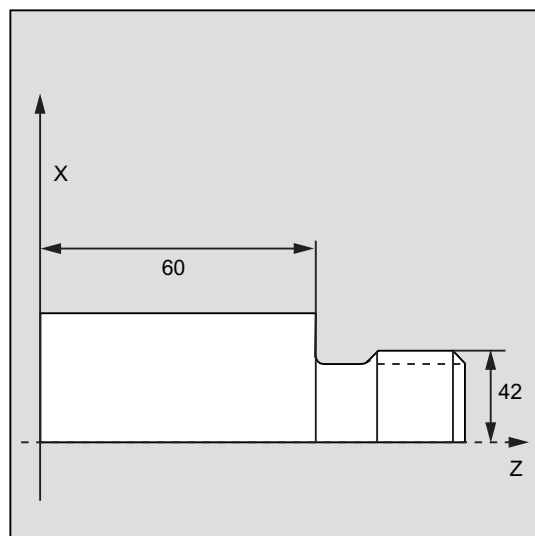
对于 **A** 型和 **B** 型，在循环中对激活刀具的自由切削角进行监控。如果确定不能用所选的刀具加工退刀槽形状，控制系统上显示信息“更改的退刀槽形状”，但是加工继续。

说明

在调用循环之前，必须激活刀具补偿；否则，中断循环并输出错误消息 61000“没有激活刀具补偿”。

编程示例：A 型螺纹退刀槽

使用该程序加工 A 型螺纹退刀槽。



N10 D3 T1 S300 M3 G95 F0.3	; 确定工艺数值
N20 G0 G90 Z100 X50	; 选择起始位置
N30 CYCLE96 (42, 60, "A",)	; 循环调用
N40 G90 G0 X100 Z100	; 逼近下一个位置
N50 M2	; 程序结束

2.5.7 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

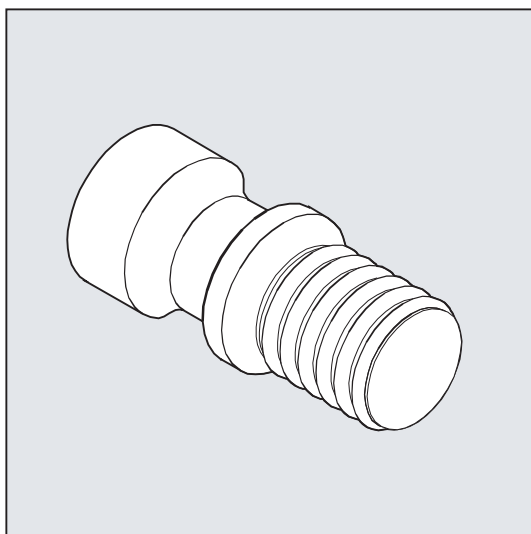
参数	数据类型	说明
PO1	实数	纵向轴上螺纹起始点
DM1	实数	起始点处螺纹的直径
PO2	实数	纵向轴中第一个中间点
DM2	实数	第一个中间点处的直径
PO3	实数	第二个中间点
DM3	实数	第二个中间点处的直径
PO4	实数	纵向轴上螺纹终点
DM4	实数	终点处直径
APP	实数	导入位移（不输入符号）
ROP	实数	收尾位移（不输入符号）
TDEP	实数	螺纹深度（不输入符号）
FAL	实数	精加工余量（不输入符号）
IANG	实数	进给角度 值范围： >0: 沿后侧齿面进刀 <0: 沿前侧齿面进刀 =0: 以垂直于切削方向的角度进刀
NSP	实数	第一个螺纹线的起始点偏移（不输入符号）
NRC	整数	粗加工切削次数（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）

参数	数据类型	说明
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
NUMTH	整数	螺纹线数量（不输入符号）
_VRT	实数	基于起始直径的可变退回位移，增量（不输入符号）

功能

使用该循环可加工多个连续的圆柱螺纹或锥螺纹。各个螺纹段可以有不同的螺距，但螺距在同一螺纹段中必须保持一致。

有关 CYCLE97 的描述，见下图：



过程

循环开始之前到达的位置：

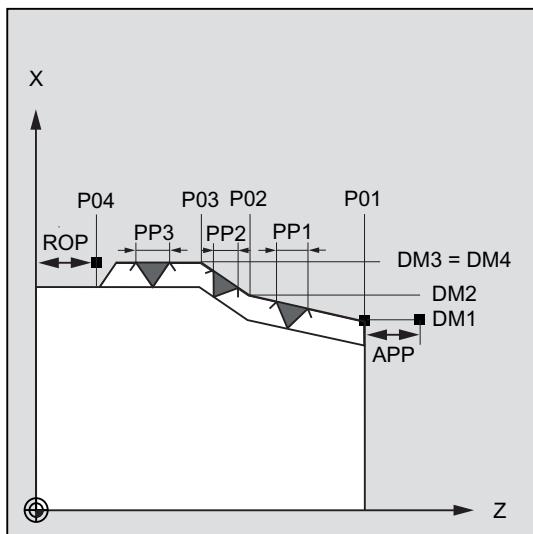
起始位置可为能无碰撞逼近螺纹起始点+导入位移的任意位置。

该循环产生以下的运动过程：

- 在第一个螺纹线导入位移的开始处使用 G0 逼近循环内部计算的起始点
- 根据 VARI 下所确定的进刀方式进行进刀（粗加工）
- 根据编程的粗加工切削次数重复螺纹切削。

- 接下来使用 **G33** 切削精加工余量。
- 根据空切削次数重复切削。
- 对之后的螺纹线重复此过程。

参数说明



P01 和 DM1 (起始点和直径)

使用这些参数定义螺纹链的原始起始点。循环自行确定的起始点 (开始使用 **G0** 逼近) 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**)。

P02, DM2 和 P03, DM3 (中间点和直径)

使用这些参数, 可以在螺纹中确定两个中间点。

P04 和 DM4 (终点和直径)

在参数 **P04** 和 **DM4** 下编程螺纹的原始终点。

对于内螺纹, **DM1...DM4** 为中心孔直径。

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。

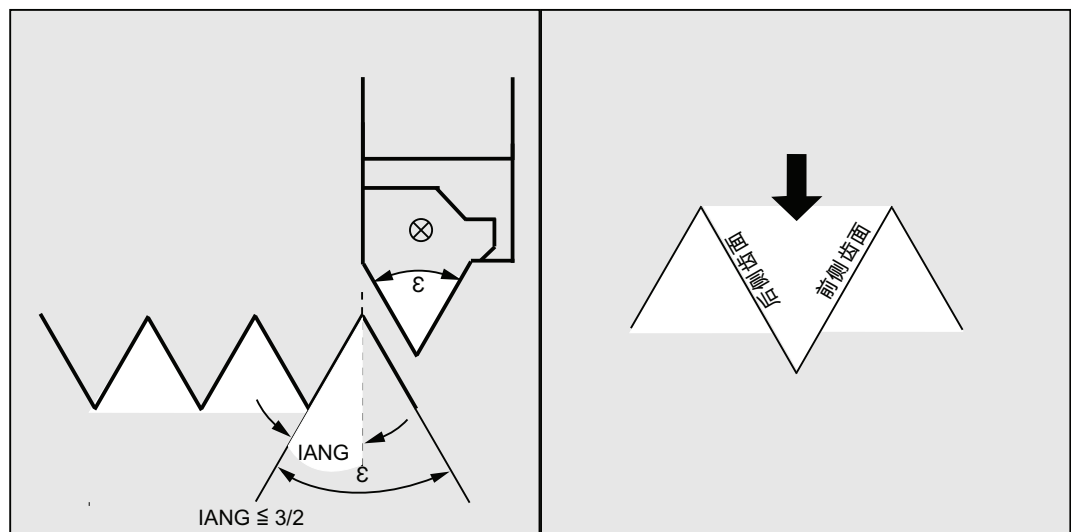
TDEP、FAL、NRC 和 NID 的关系（螺纹深度、精加工余量、粗加工切削次数和空切削次数）

将编程的精加工余量从设定的螺纹深度 TDEP 中减去，剩余量分割为多个粗加工切削。循环视参数 VARI 而定，自行计算各个实际进刀深度。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，将整个螺纹深度分割成恒定的进刀深度。此时切削截面会逐步增大，然而螺纹深度值较小时，采用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 下编程的空切削。

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。由于在这种情况下，值自动默认为零，所以，这意味着可以在参数列表中忽略该参数。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。

进刀的执行通过参数的符号定义。在正值情况下，始终沿后侧齿面进刀；在负值情况下，始终沿前侧齿面进刀。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（起始点偏移）

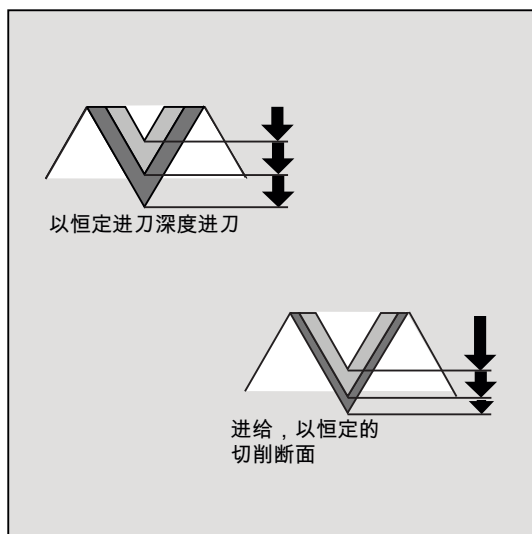
在该参数下可以编程角度值，来定义车削件圆周上第一个螺纹线的切削点。该值即起始点偏移。参数取值的范围可以为 0.0001 ~ +359.9999 度。如果没有设定起始点偏移位置或者在参数列表中省略了该参数，则第一个螺纹线自动在零度标记处开始。

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。必须输入不带符号的轴向平行值作为螺距值。

VARI（加工方式）

通过参数 VARI 确定，是否要采用外加工或内加工，以及在粗加工时采用何种工艺进刀。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

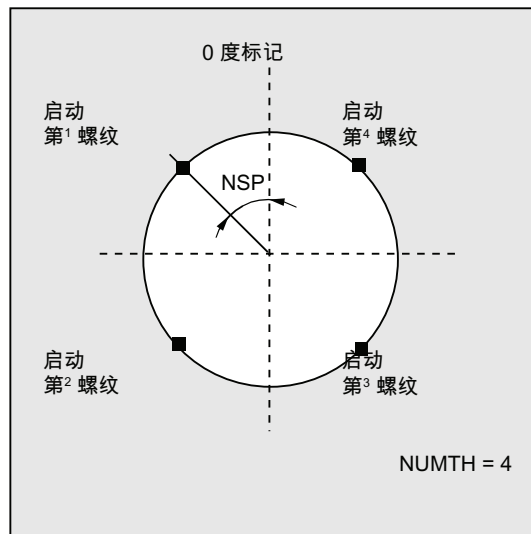
如果参数 VARI 被赋予了其它值，则循环中断并输出报警 61002“加工方式定义错误”。

NUMTH (螺纹线数量)

通过参数 NUMTH 确定多线螺纹的螺纹线数量。对于单线螺纹，将该参数置零或者可以在参数列表中完全清除。

螺纹线均匀地分布在车削件圆周上，第一个螺纹线由参数 NSP 确定。

如果需要将多线螺纹中的螺纹线不均匀地分布在圆周上，则应在编程相应的起始点偏移时为每个螺纹线调用循环。

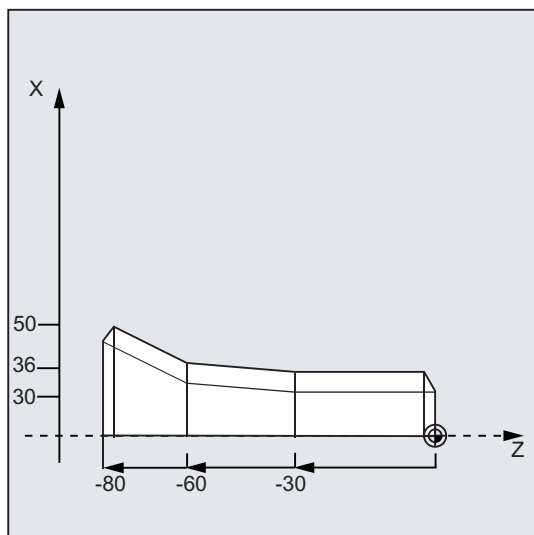


_VRT (可变的退回位移)

在参数 _VRT 下，退回位移可以通过螺纹导出端直径编程。在 _VRT = 0 (参数未编程) 时，退刀 1 毫米。退回位移总是与编程的测量系统 (英制或者公制) 有关。

编程示例： 螺纹链

使用此参数加工一个以圆柱螺纹开始的螺纹链。垂直于螺纹进刀，不编程精加工余量和起始点偏移。进行 5 次粗加工切削和一次空切削。设定加工方式为纵向、外部且切削截面恒定。



N10 G95 T5 D1 S1000 M4	: 确定工艺数值
N20 G0 X40 Z10	: 逼近起始位置
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 M2	: 程序结束

2.5.8 螺纹切削 - CYCLE99

编程

CYCLE99 (SPL, DM1, FPL, DM2, APP, ROP, TDEP, FAL, IANG, NSP, NRC, NID, PIT, VARI, NUMTH, _VRT, 0, 0, 0, 0, 0, 0, 0, 0, PITA, 0, 0, 0, PSYS)

参数

参数	数据类型	说明
SPL	实数	纵向轴上螺纹起始点
DM1	实数	起始点处螺纹的直径
FPL	实数	纵向轴上螺纹终点
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： 〉0：沿后侧齿面进刀 〈0：沿前侧齿面进刀 =0：以垂直于切削方向的角度进刀
NSP	实数	第一个螺纹线的起始点偏移（不输入符号）
NRC	整数	粗加工切削次数（不输入符号）
NID	整数	空走刀次数（不输入符号）
PIT	实数	螺距值（不输入符号） !!! 单位在参数 PITA 中定义。
VARI	整数	确定螺纹的加工方式
		参数值

2.5 车削循环

参数	数据类型	说明	
NUMTH	整数	螺纹线数量（不输入符号）	
_VRT	实数	基于初始直径的可变回退路径，增量（不输入符号）	
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PSYS	整数	内部参数，只允许默认值 0	
		值:	0
PITA	整数	参数 PIT 单位（螺距）	
		值:	1 螺距，单位为毫米/转 2 螺距，单位为螺纹数/英寸（TPI）
PSYS	STRING	内部参数，只允许默认值 0	
		值:	" "
PSYS	STRING	内部参数，只允许默认值 0	
		值:	" "
PSYS	STRING	内部参数，只允许默认值 0	
		值:	" "
PSYS	整数	内部参数，可能的值如下：	
		值:	0 纵向螺纹 10 横向螺纹 20 锥管螺纹

功能

螺纹切削循环包括以下三种循环：直螺纹、端面螺纹或锥形螺纹。

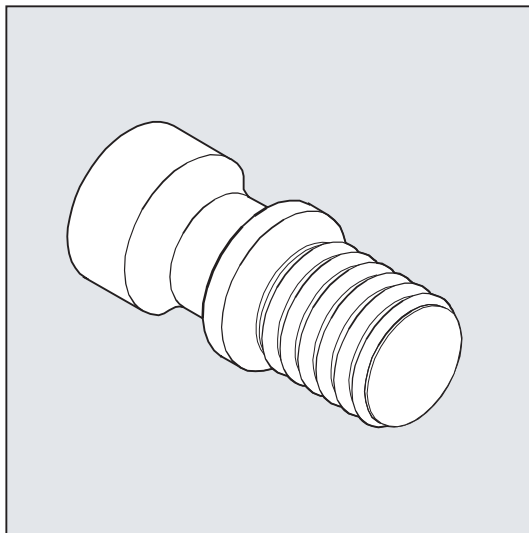
通过螺纹切削循环，可以在纵向和平面加工中以恒定螺距加工圆柱形和锥形外螺纹和内螺纹。螺纹可以是单线螺纹，也可以是多线螺纹。对于多线螺纹，依次对各个螺纹线进行加工。

进刀自动进行，可以选择每次切削时恒定的进给，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，这必须在循环调用之前编程。

进给倍率和主轴倍率在带螺纹的运行程序段中都无效。

有关 CYCLE99 的描述，见下图：



说明

使用该循环的前提条件是带位移测量系统的转速控制主轴。

过程

循环开始之前到达的位置：

起始位置可为能无碰撞逼近螺纹起始点+导入位移的任意位置。

该循环产生以下的运动过程：

- 在第一个螺纹线导入位移的开始处使用 G0 逼近循环内部计算的起始点
- 根据 VARI 下所确定的进刀方式进行进刀（粗加工）
- 根据编程的粗加工切削次数重复螺纹切削。

- 接下来使用 G33 切削精加工余量。
- 根据空切削次数重复切削。
- 对之后的螺纹线重复此过程。

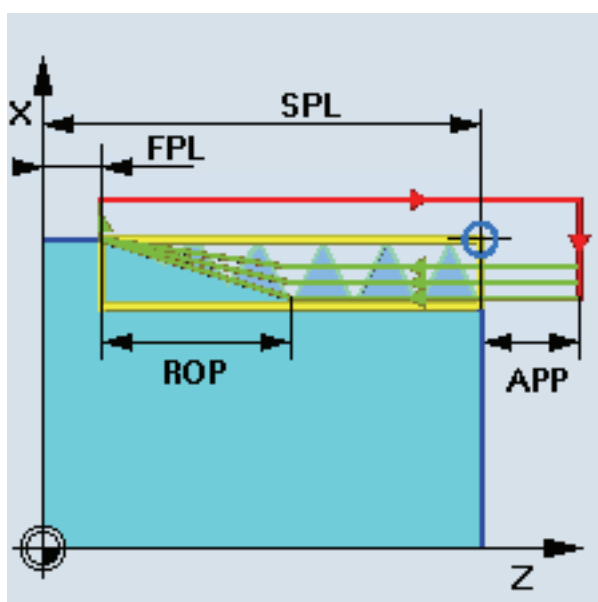
参数说明

DM1 和 DM2 (直径)

使用该参数，可以确定螺纹起始点和终点处螺纹直径。对于内螺纹为中心孔直径。

SPL、FPL、APP 和 ROP 的关系 (起始点、终点、导入位移和收尾位移)

有关 CYCLE99 参数的描述，见下图：



编程的起始点 (SPL) 以及终点 (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。

收尾位移 (切削) 在编程的终点 FPL 之前开始。它使螺纹终点提前，因而切削终点等于 FPL。

TDEP、FAL、NRC 和 NID 关系 (螺纹深度、精加工余量、切削次数)

将编程的精加工余量从设定的螺纹深度 TDEP 中减去，剩余量分割为多个粗加工切削。

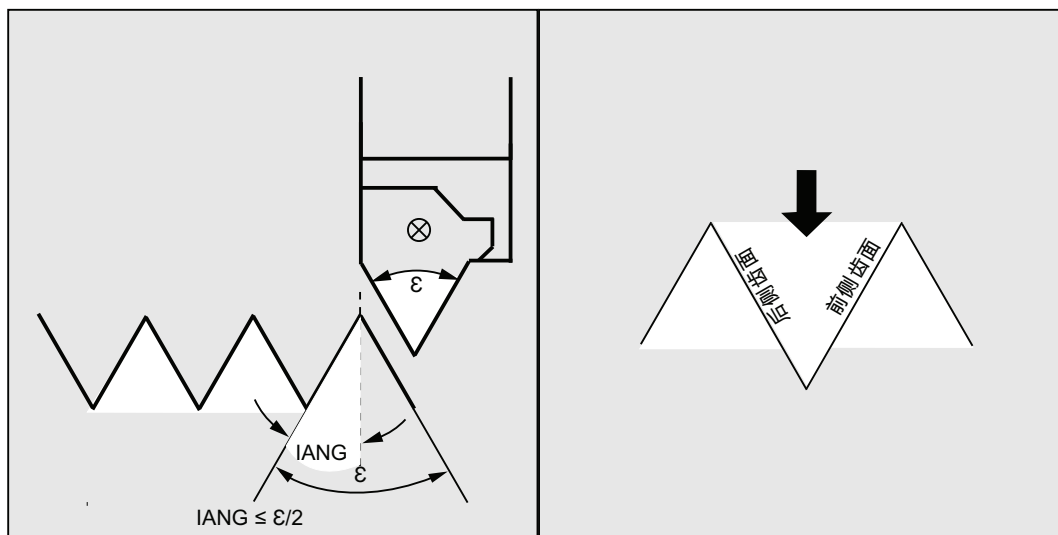
循环视参数 VARI 而定，自行计算各个实际进刀深度。

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，将整个螺纹深度分割成恒定的进刀深度。此时切削截面会逐步增大，然而螺纹深度值较小时，采用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 下编程的空切削。

IANG（进刀角度）

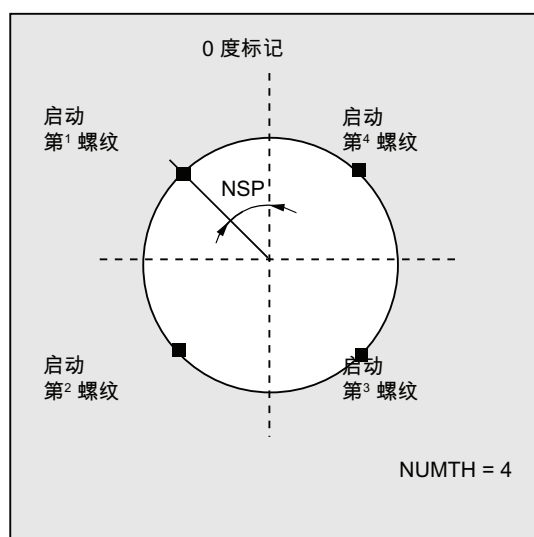


通过参数 IANG 确定螺纹中进刀的角度。如果在螺纹中以垂直于切削方向的角度进刀，则必须将该参数值置零。如果沿着齿面进刀，则该参数绝对值最大允许为刀具啮合角的一半。

进刀的执行通过参数的符号定义。在正值情况下，始终沿后侧齿面进刀；在负值情况下，始终沿前侧齿面进刀。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 (起始点偏移) 和 NUMTH (数量)

在该参数下可以编程角度值，来定义车削件圆周上第一个螺纹线的切削点。该值即起始点偏移。参数取值的范围可以为 $0 \sim +359.9999$ 度。如果没有设定起始点偏移位置或者在参数列表中省略了该参数，则第一个螺纹线自动在零度标记处开始。



通过参数 **NUMTH** 确定多线螺纹的螺纹线数量。对于单线螺纹，将该参数置零或者可以在参数列表中完全清除。

螺纹线均匀地分布在车削件圆周上，第一个螺纹线由参数 **NSP** 确定。

如果需要将多线螺纹中的螺纹线不均匀地分布在圆周上，则应在编程相应的起始点偏移时为每个螺纹线调用循环。

PIT (螺距) 和 PITA (螺距单位)

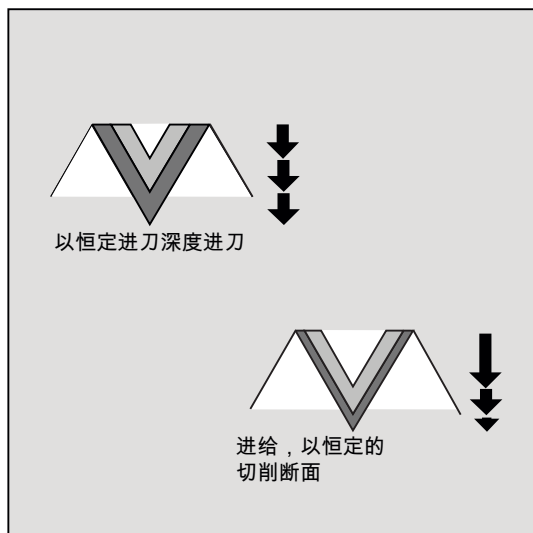
螺距是一个轴向平行的值，不设定符号。单位在参数 **PITA** 中定义。

PITA = 1 螺距，单位为毫米/转

= 2 螺距，单位为螺纹数/英寸 (TPI)

VARI（加工方式）

通过参数 VARI 确定，是否要采用外加工或内加工，以及在粗加工时采用何种工艺进刀。
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
300101	O	恒定进刀
300102	I	恒定进刀
300103	O	恒定切削截面
300104	I	恒定切削截面

如果参数 VARI 被赋予了其它值，则循环中断并输出报警 61002“加工方式定义错误”。

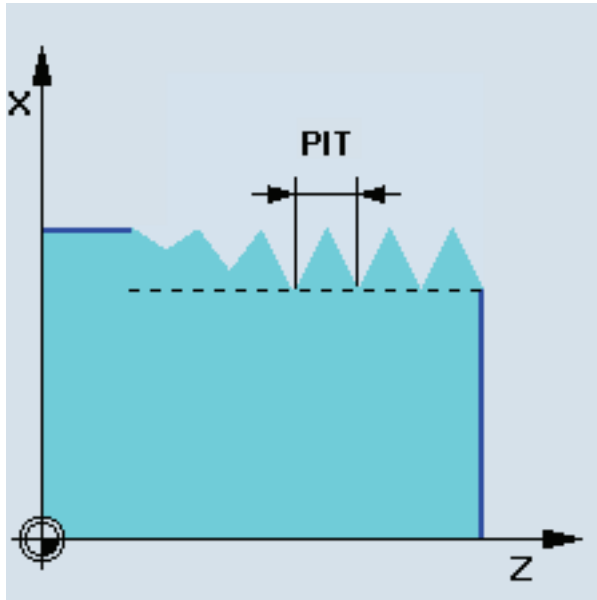
_VRT（可变的退回位移）

在参数 _VRT 下，退回位移可以通过螺纹导出端直径编程。在 _VRT = 0（参数未编程）时，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。

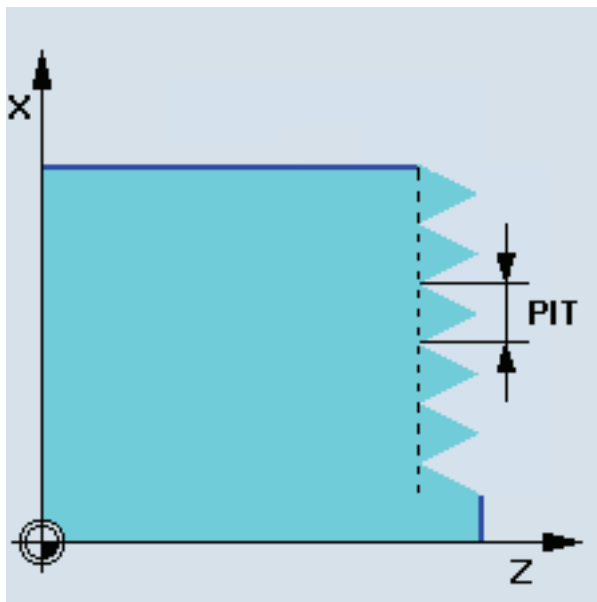
PSYS (画面上末次显示的值)

该参数说明了纵向螺纹、横向螺纹或椎管螺纹的选择。

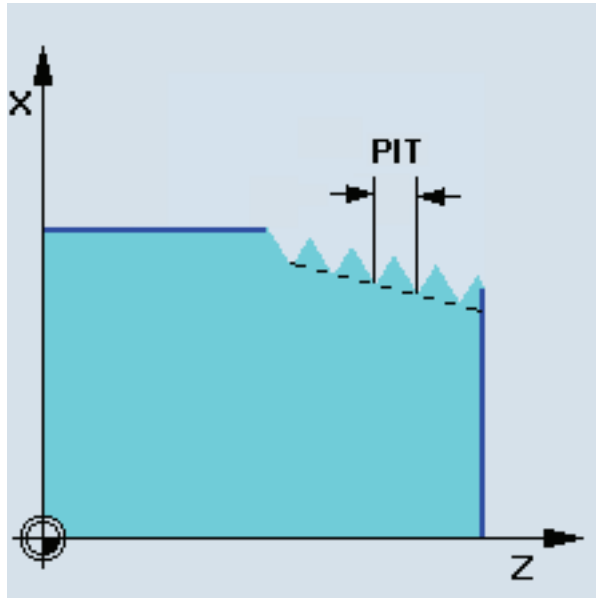
0 = 直螺纹:



10 = 端面螺纹:



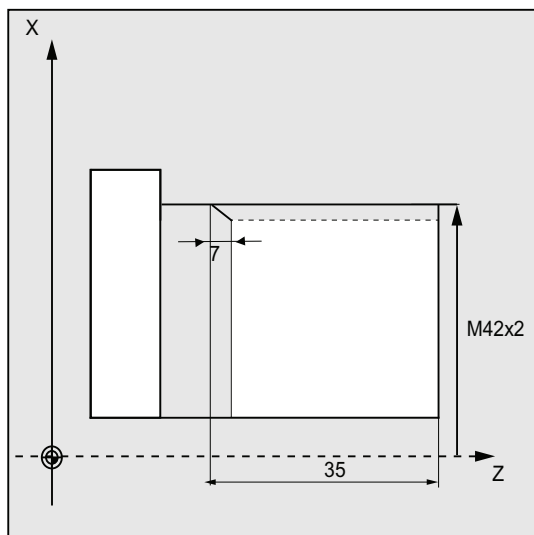
20 = 锥形螺纹:



2.5 车削循环

编程示例： 螺纹切削

使用该程序通过齿面进刀加工公制外螺纹 M42x2。以恒定的切削截面进刀。在螺纹终点处设定一个 7 毫米的切削。螺纹深度为 2.76 毫米时，在没有精加工余量情况下，执行 5 次粗加工切削。结束后，进行两次空切削。



```

N10 G0 G90 X60 Z100 G95 ; 选择起始位置
N20 T1 D1
N30 M6 ; 换刀
N40 S1000 M4 ; 确定工艺数值
N50 CYCLE99(0, 42, -35, 42, 5, 7, 2.76, 0, 0, 0, ; 循环调用
5, 2, 4.5, 300101, 1, 0, 0, 0, 0, 0, 0, 0, 0,
1, , , ,0)
N60 G0 G90 X100 Z100 ; 逼近下一个位置
N70 M30 ; 程序结束
    
```

2.6 故障信息和故障处理

2.6.1 一般说明

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信息行中输出信息。这个信息不会中断加工。

这些故障及其应答以及系统中信息行中的信息每次均在各个循环中进行描述。

2.6.2 循环中的故障处理

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。

与报警号码同时显示出的故障文本，可以给出关于故障原因的进一步阐述。

报警号	删除标准	报警反应
61000 ... 61999	NC 复位	中断 NC 中的程序段处理。
62000 ... 62999	清除按键	中断程序段处理，在删除报警后按下下列 MCP 上的键可继续循环。 

2.6.3 循环报警一览

报警号可以按如下分类：

6	_	X	_	_
---	---	---	---	---

- X=0 一般循环报警
- 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

2.6.4 循环中的显示消息

循环在控制器的信息行中输出信息。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直至循环结束均保持不变。消息示例如下：

"深度： 根据相对深度值"来自所有钻削循环。

典型车削程序

毛坯数据

毛坯材料：硬铝 V

毛坯直径：50 毫米

毛坯长度：60 毫米（加工长度：46 毫米；夹紧长度：10 毫米）

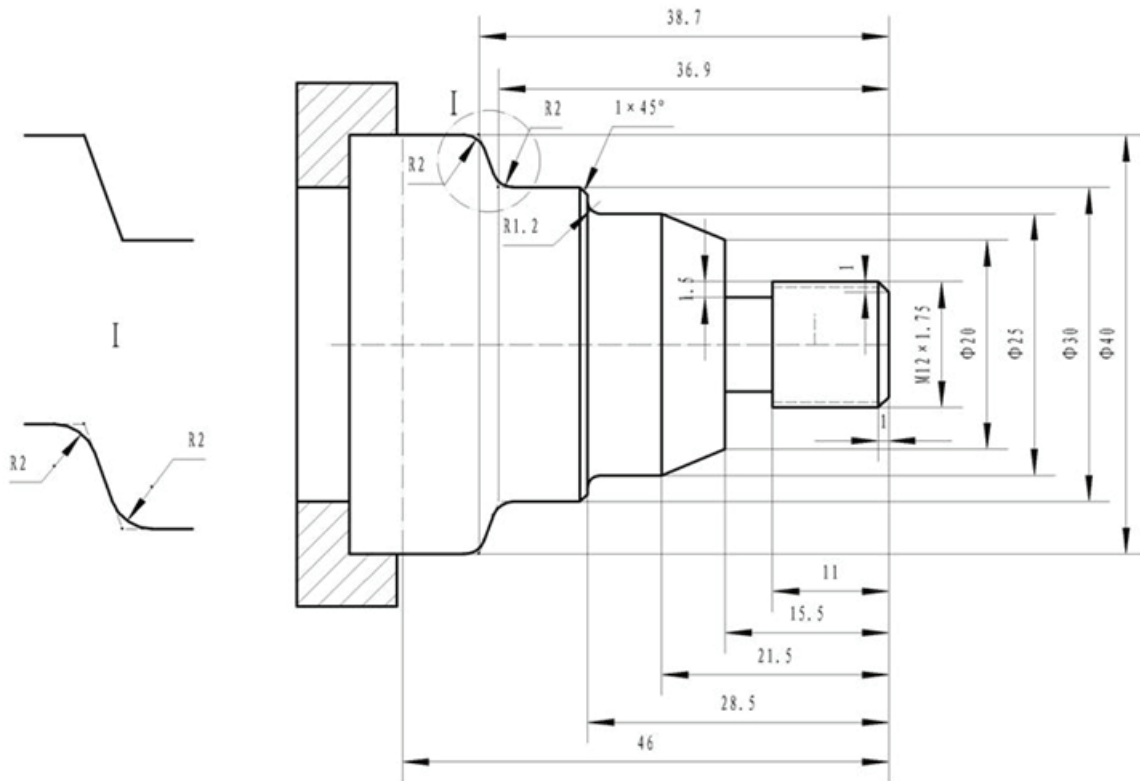
所需刀具

T1: 底切刀具

T2: 刻槽刀具

T3: 螺纹切削刀具

编程示例 1



```

T1
S2000 M03
F0.4
G0 X60 Z10
CYCLE95( "PART_CONTOUR:END_T", 1.00000, , ,0.20000, 0.30000, 0.20000, 0.10000,
9, , ,1.00000)
T2
S1000 M03
F0.2
CYCLE93( 20.00000, -11.00000, 4.50000, 1.50000, , , , , , , ,0.20000, 0.20000,
1.00000, ,5,)
T3
S1000 M03
CYCLE99( 1.75000, 0, 0.00000, -13.00000, 20.00000, 20.00000, 2.00000, 2.00000,
1.00000, 0.10000, ,0.00000, 8, 1, 1, 1)
M2
PART_CONTOUR:
G0 Z0 X0
    
```

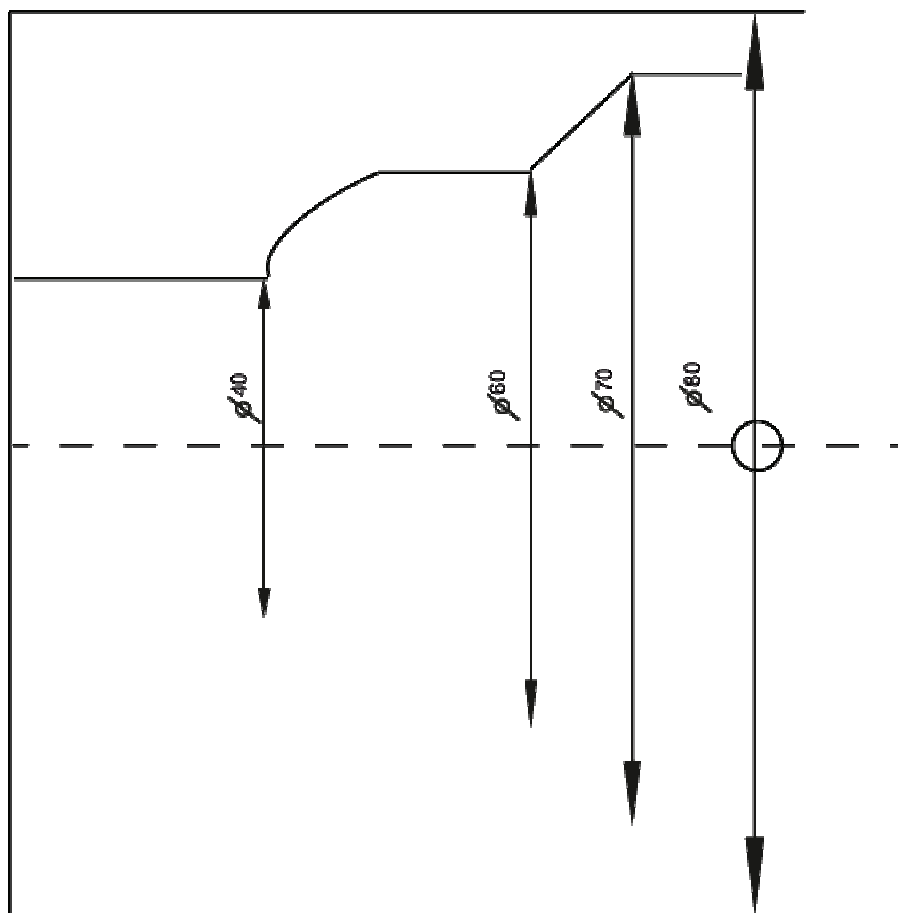
```

G1 X20 CHF=1
Z-15.5
X25 Z-21.5
Z-28.5 RND=1.2
X30 CHF=1
Z-36.9 RND=2
X40 Z-38.7 RND=2
Z-46
X50
END_T:

```

编程示例 2

下例将说明如何处理腔轮廓和如何通过轮廓功能定义轮廓的子程序。



```

G500 G18 G95
G0X50
Z100

```

```

T5
M4S1500
G0X50Z0.5
G01X-2F0.15
Z2
G0X50
Z100
T2
G0X50Z10
M4S1500
G1F0.2
CYCLE95( "CON01", 0.50000, 0.20000, 0.20000, 0.20000, 0.20000, 0.20000, 0.10000,
1, , ,1.00000)
G0X55
Z100
M5
T1
M3S1500
G0X50Z10
CYCLE95( "CON02:CON02_E", 0.50000, 0.20000, 0.20000, 0.20000, 0.20000, 0.20000,
0.10000, 5, , ,1.00000)

M30

;*****轮廓*****
CON02:

;#7__DlGK contour definition begin - Don't change!;*GP*;*RO*;*HD*
G18 G90 DIAMON;*GP*
G0 Z0 X0 ;*GP*
G1 X28 CHR=3 ;*GP*
Z-8.477 RND=2 ;*GP*
G2 Z-45.712 X40 K=AC(-25) I=AC(60) RND=2 ;*GP*
G1 Z-50 RND=3 ;*GP*
Z-55 X45 ;*GP*
;CON,V64,2,0.0000,0,0,MST:1,2,AX:Z,X,K,I;*GP*;*RO*;*HD*
;S,EX:0,EY:0,ASE:0;*GP*;*RO*;*HD*
;LU,EY:28;*GP*;*RO*;*HD*
;F,LFASE:3;*GP*;*RO*;*HD*
;LL;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;ACW,DIA:209/217,EY:40,CX:-25,CY:60,RAD:23;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;LL,EX:-50;*GP*;*RO*;*HD*
;R,RROUND:3;*GP*;*RO*;*HD*

```

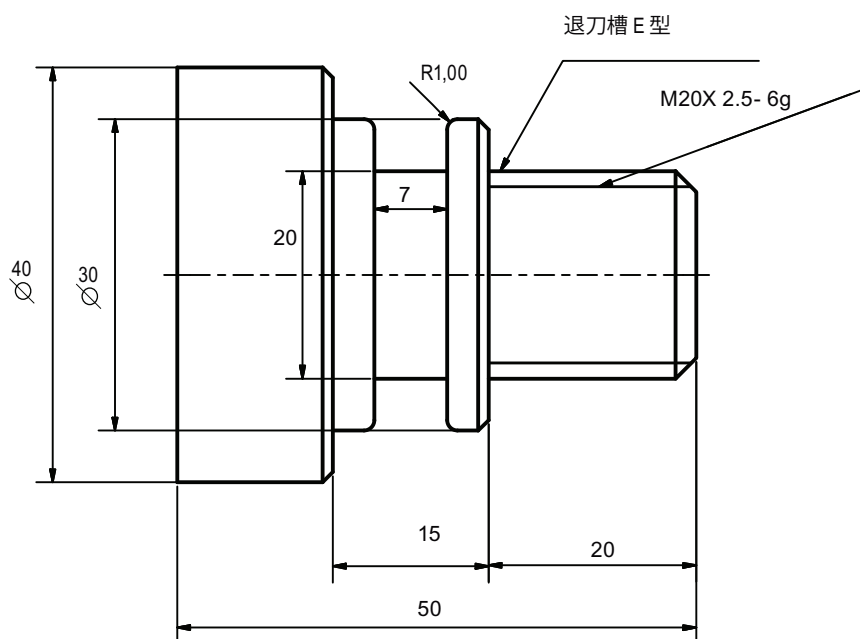
```

;LA,EX:-55,EY:45;*GP*;*RO*;*HD*
;#End contour definition end - Don't change!*GP*;*RO*;*HD*
M17

CON02_E:;***** 轮廓终点 *****

```

编程示例 3



```

N10 G00 G90 G95 G40 G71
N20 LIMS=4500

N30 T1 D1 ;ROUGH TURN
N40 G96 S250 M03 M08
N50 G00 X52.0 Z0.1
N60 G01 X-2.0 F0.35
N70 G00 Z2.0
N80 X52.0
CYCLE95( "DEMO:DEMO_E", 2.50000, 0.20000, 0.10000, 0.15000, 0.35000, 0.20000,
0.15000, 9, , ,)
N90 G00 G40 X500.0 Z500.0
N100 M01

N110 T2 D1 ;FINISH TURN

```

```

N120 G96 S350 M03 M08
N130 G00 X22.0 Z0.0
N140 G01 X-2.0 F0.15
N150 G00 Z2.0
N160 X52.0
N170 CYCLE95( "DEMO:DEMO_E", , , , , , ,0.15000, 5, , , )
N180 G00 G40 X500.0 Z500.0
N190 M01

N200 T3 D1 ;GROOVE
N210 G96 S200 M03 M08
N220 G00 X55.0 Z0.
N230 CYCLE93( 30.00000, -30.50000, 7.00000, 5.00000, 0.00000, 0.00000, 0.00000,
1.00000, 1.00000, ,0.00000, 0.20000, 0.10000, 2.50000, 0.50000, 11, )
N240 G00 G40 X500.0 Z500.0
N250 M01

N260 T4 D1 ;THREAD
N270 G95 S150 M03 M08
N280 G00 X50.0 Z10.0
N290 CYCLE99( ,20, 0.00000, -18.00000, 20.00000, 20.00000, 2.00000, 0.00000,
1.00000, 0.01000, 29.00000, 0.00000, 8, 2, 3, 1, )
N300 G00 G40 X500.0 Z500.0
N310 M01

N320 T5 D1 ;CUT-OFF
N330 G96 S200 M03 M08
N340 G00 X55.0 Z10.0
N350 CYCLE92( 40.00000, -50.00000, 6.00000, -1.00000, 0.50000, ,200.00000,
2500.00000, 3, 0.20000, 0.08000, 500.00000, 0, 0, 1, 0, 11000)
N360 G00 G40 X500.0 Z500.0

N370 M30

;*****轮廓*****
DEMO:

;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD*
G18 G90 DIAMON;*GP*
G0 Z0 X16 ;*GP*
G1 Z-2 X20 ;*GP*
Z-15 ;*GP*
Z-16.493 X19.2 RND=2.5 ;*GP*
Z-20 RND=2.5 ;*GP*
X30 CHR=1 ;*GP*

```

```

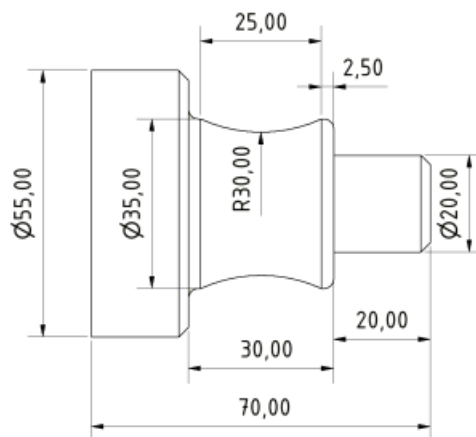
Z-35 ;*GP*
X40 CHR=1 ;*GP*
Z-55 ;*GP*
X50 ;*GP*
;CON,V64,2,0.0000,4,4,MST:1,2,AX:Z,X,K,I;*GP*;*RO*;*HD*
;S,EX:0,EY:16,ASE:0;*GP*;*RO*;*HD*
;LA,EX:-2,EY:20;*GP*;*RO*;*HD*
;LL,EX:-20;*GP*;*RO*;*HD*
;AB,IDX:8;*GP*;*RO*;*HD*
;LU,EY:30;*GP*;*RO*;*HD*
;F,LFASE:1;*GP*;*RO*;*HD*
;LL,DEX:-15;*GP*;*RO*;*HD*
;LU,EY:40;*GP*;*RO*;*HD*
;F,LFASE:1;*GP*;*RO*;*HD*
;LL,EX:-55;*GP*;*RO*;*HD*
;LU,EY:50;*GP*;*RO*;*HD*
;#End contour definition end - Don't change!;*GP*;*RO*;*HD*

DEMO_E:***** 轮廓终点 *****

```

车削零件程序示例 1

以下为车削零件程序示例：



```

G00 G90 G95 G40 G71
LIMS=4500
T1 D1
G96 S250 M03 M08
G00 X60 Z0

```

```
G01 X-2 F0.35
G00 Z2
G00 X60
CYCLE95( "CON1:CON1_E", 1.50000, 0.20000, 0.10000, ,0.50000, 0.30000, 0.20000,
9, , , )
T2 D1
G96 S250 M03 M08
CYCLE95( "CON2:CON2_E", 0.50000, , ,0.20000, 0.40000, 0.30000, 0.20000, 9, , , )
M30

CON1:
;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD*
G18 G90 DIAMON;*GP*
G0 Z0 X16 ;*GP*
G1 Z-2 X20 ;*GP*
Z-20 ;*GP*
X35 RND=2 ;*GP*
Z-50 RND=2 ;*GP*
X55 CHR=2 ;*GP*
Z-70 ;*GP*
;CON,V64,2,0.0000,6,6,MST:1,2,AX:Z,X,K,I;*GP*;*RO*;*HD*
;S,EX:0,EY:16,ASE:0;*GP*;*RO*;*HD*
;LA,EX:-2,EY:20;*GP*;*RO*;*HD*
;LL,EX:-20;*GP*;*RO*;*HD*
;LU,EY:35;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;LL,DEX:-30;*GP*;*RO*;*HD*
;R,RROUND:2;*GP*;*RO*;*HD*
;LU,EY:55;*GP*;*RO*;*HD*
;F,LFASE:2;*GP*;*RO*;*HD*
;LL,EX:-70;*GP*;*RO*;*HD*
;#End contour definition end - Don't change!;*GP*;*RO*;*HD*
CON1_E:

;*****轮廓*****
CON2:

;#7__DlgK contour definition begin - Don't change!;*GP*;*RO*;*HD*
G18 G90 DIAMON;*GP*
G0 Z-22.5 X35 ;*GP*
G2 Z-47.5 K=AC(-35) I=AC(89.544) ;*GP*
G1 Z-49.5 ;*GP*
;CON,V64,2,0.0000,1,1,MST:1,2,AX:Z,X,K,I;*GP*;*RO*;*HD*
;S,EX:-22.5,EY:35,ASE:0;*GP*;*RO*;*HD*
```



```

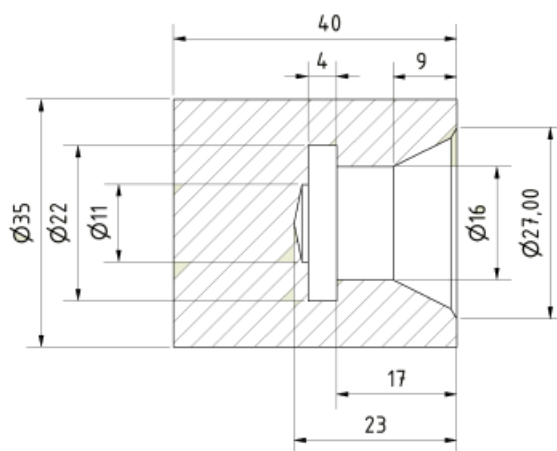
;ACW,DIA:0/235,DEX:-25,DEY:0,RAD:30;*GP*;*RO*;*HD*
;LL,DEX:-2;*GP*;*RO*;*HD*
;#End contour definition end - Don't change!*GP*;*RO*;*HD*

CON2_E:;***** 轮廓终点 *****

```

车削零件程序示例 2

以下为另一个车削零件程序示例:



```

N10 G54G00 G90 G95 G40 G71
N20 LIMS=4500
N30 T1 D1
N40 G96 S250 M03 M08
N50 G00 X35 Z0
N60 G01 X-2 F0.35
N70 G00 Z2
N80 G00 X35
N90 T13 D1
N100 G95 S1000 M4
N110 G00 Z1 X0
N120 CYCLE83( 10.00000, 0.00000, 2.00000, -23.00000, 0.00000, -
10.00000, ,5.00000, , ,1.00000, 0, 1, 5.00000, 0.00000, ,0.00000)
N130 G18
N140 T10 D1
CYCLE95( "CON1:CON1_E", 1.50000, 0.20000, 0.10000, ,0.50000, 0.30000, 0.20000,
11, , ,)
N30 T110 D1
N40 G96 S250 M03 M08

```


索引

C

CONTPRON, 193
CYCLE81, 127
CYCLE82, 130
CYCLE83, 133
CYCLE84, 139
CYCLE840, 146
CYCLE85, 152
CYCLE86, 155
CYCLE87, 158
CYCLE88, 160
CYCLE89, 162
CYCLE92, 166
CYCLE93, 169
CYCLE94, 179
CYCLE95, 184
CYCLE96, 202
CYCLE98, 206
CYCLE99, 213

E

EXTCALL, 113

G

G62, 16
G621, 16

M

M19, 68
M70, 68

S

SD43240, 70
SD43250, 70
SPOS, 68, 141, 142
SPOSA, 68

W

WAITS, 68

二划

几何参数, 123

四划

不带补偿夹具的攻丝, 139
切削循环 - CYCLE95, 184
切断 - CYCLE92, 166
切槽循环 - CYCLE93, 169
车削循环, 119

五划

主轴
 定位, 68
加工平面, 120

加工参数, 123
可打印的特殊字符, 9
平面定义, 120

六划

地址, 8
字结构, 8
字符集, 9
安全距离, 128
自由切削角度, 165

七划

返回条件, 120
进行配置的输入窗口, 122

八划

定心, 127
轮廓定义, 192
轮廓监控, 165, 194
非可打印特殊字符, 9

九划

带补偿卡盘、带编码器的攻丝, 148
带补偿攻丝, 146
相对钻削深度, 128
绝对钻削深度, 128
退刀槽循环 - CYCLE94, 179
退回平面, 128

十划

消息, 224
调用, 124

调用条件, 120
起始点, 194
钻削, 127
钻削, 镗平面, 130
钻削循环, 119

十一划

基准面, 128
深孔钻削, 133
深孔钻削, 带排屑, 135
深孔钻削, 带断屑, 136

十二划

循环支持的条件, 122
循环仿真, 121
循环报警, 223
循环报警概述, 223
循环调用, 120
程序段结构, 10
程序编辑器的循环支持, 122

十六划

镗孔, 123
镗孔 1, 152
镗孔 2, 155
镗孔 3, 158
镗孔 4, 160
镗孔 5, 162

十七划

螺纹切削 - CYCLE99, 213
螺纹退刀槽 - CYCLE96, 202
螺纹链 - CYCLE98, 206

