

封面记载内容

项目	记载内容
机种表述	EZMotion-NC E60/E68 系列
说明书名称	编程说明书（L 系）
说明书编号	IB-1500183 (CHN) -A

注意

本页为临时封面。
请不要包含在正文中。

EZMotion、MELDAS 是三菱电机株式会社的注册商标。
其他的产品名、公司名分别为各公司的商标或注册商标。

前言

本说明书是使用 EZMotion-NC E60/E68 系列时的说明书。

本说明书对编程加以说明，在使用之前，请务必仔细阅读本说明书。

此外，为了能够安全的使用本数控装置，请在熟读了下页的“安全注意事项”之后再使用本装置。

本说明书的记载内容

注意

 “限制事项”及“能够使用的功能”等记载事项方面，由机械制造商发行的说明书比本书具有更高的优先级。

 本说明书中未记载的事项，请解释为“不可以”。

 在编写本说明书时，是假定为附加了所有的选配功能。在使用时，请通过机械制造商所刊行的规格书加以确认。

 各工作机械的相关说明，请参阅由机械制造商所发行的说明书。

 能够使用的画面及功能，因各 NC 系统（或版本）而异。在使用前，请务必对规格加以确认。

一般性注意事项

- (1) 使用请参阅以下资料。

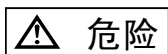
操作说明书

IB-1500182

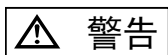
安全注意事项

在安装、运行、编程、维护・检修之前，请务必熟读机械制造商所刊行的规格书、本说明书、相关说明书、附属文件，然后正确使用。请在熟悉了本数控装置相关知识、安全信息及注意事项之后再使用。

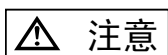
在本说明书中，将安全注意事项的等级分为“危险”、“警告”、“注意”。



如果错误使用，会直接带来导致使用者死亡或重伤的危险的场合。



如果错误使用，可能会导致使用者死亡或重伤的场合。



如果错误使用，会导致使用者负伤的场合，以及仅会导致物质损失的场合。

另外，即使是标记为“注意”的事项，根据情况，也可能导致重大后果。任何一项均是记录了重要内容，所以请务必遵守。

危 险

本说明书中无。

警 告

本说明书中无。

注 意

1. 产品、说明书中相关事项

 “限制事项”及“能够使用的功能”等记载事项方面，由机械制造商发行的说明书比本书具有更高的优先级。

 本说明书中未记载的事项，请解释为“不可以”。

 在编写本说明书时，是假定为附加了所有的选配功能。在使用时，请通过机械制造商所刊行的规格书加以确认。

 各工作机械的相关说明，请参阅由机械制造商所发行的说明书。

 能够使用的画面及功能，因各 NC 系统（或版本）而异。在使用前，请务必对规格加以确认。

2. 操作相关事项

 请在进行实际加工之前进行空运转，进行加工程序、刀具偏移量、工件偏移量等的确认。

 如果在单节停止时变更工件坐标系偏移量，则从下一单节起开始生效。

 请在镜像中心进行镜像的开/关。

（接下页）

⚠ 注意

(续)

⚠如果在自动运转中（含单节停止中）进行刀具偏移量的变更，则从下一单节或多单节之后的指令开始生效。

⚠在基准主轴与同期主轴同时夹紧同一工件的状态下，请不要将同期主轴端的旋转指令 OFF。由于同期主轴会停止，所以非常危险。

3. 编程相关事项

⚠“G 后无数值”指令，被看作为“G00”。

⚠“;”“EOB”及“%”“EOR”是用于说明的表述。在 ISO 中，与“;”“EOB”相对应的实际代码为“CR, LF”或“LF”。“%”“EOR”在 ISO 中为“%”。

在编辑画面中所创建的程序，以“CR, LF”的形式被储存在 NC 内存中，而通过外部设备创建的程序则可能会是“LF”的形式。

对于 EIA，则为“EOB (End Of Block)”和“EOR (End Of Record)”。

⚠在创建加工程序时，请选择适当的加工条件，不要超过机械、NC 的性能、容量、限制。例题已考虑到了加工条件。

⊘请不要在没有知会机械制造商的情况下变更固定循环程序。

目 录

1. 控制轴.....	1
1.1 坐标与控制轴.....	1
1.2 坐标系与坐标原点标记.....	2
2. 输入指令单位.....	3
2.1 输入指令单位.....	3
2.2 输入设定单位.....	3
3. 数据格式.....	5
3.1 纸带代码.....	5
3.2 程序格式.....	8
3.3 程序地址检查.....	10
3.4 纸带记忆格式.....	11
3.5 可选单节的跳跃.....	11
3.6 程序编号、顺序编号及节编号；O、N.....	12
3.7 检验 H/V.....	13
3.8 G 代码系列.....	14
3.9 加工前的注意事项.....	17
4. 缓存寄存器.....	18
4.1 输入缓存.....	18
4.2 预读缓存.....	19
5. 位置指令.....	20
5.1 增量值指令/绝对值指令.....	20
5.2 半径指令/直径指令.....	21
5.3 英制指令/公制指令切换；G20、G21.....	22
5.4 小数点输入.....	26
6. 插补功能.....	30
6.1 定位（快速进给）；G00.....	30
6.2 直线插补；G01.....	36
6.3 圆弧插补；G02、G03.....	39
6.4 R 指定圆弧插补；G02、G03.....	43
6.5 平面选择；G17、G18、G19.....	45
6.6 螺纹切削.....	47
6.6.1 固定导程螺纹切削；G33.....	47
6.6.2 英制螺纹切削；G33.....	51
6.6.3 连续螺纹切削.....	52
6.6.4 可变导程螺纹切削；G34.....	53
6.7 螺旋插补；G17、G18、G19 及 G02、G03.....	56
6.8 圆筒插补；G07.1（仅 G 代码系列 6、7）.....	59
6.9 极坐标插补；G12.1（仅 G 代码系列 6、7）.....	67
7. 进给功能.....	74
7.1 快速进给速度.....	74
7.2 切削进给速度.....	74
7.3 F1 数位进给.....	75

7.4 同期进给/非同期进给 ; G94、G95	76
7.5 进给速度的指定与对各控制轴的效果	78
7.6 螺纹切削模式.....	82
7.7 自动加减速	83
7.8 速度钳位	83
7.9 精确停止检查; G09.....	84
7.10 精确停止检查模式; G61.....	87
7.11 减速检查	88
7.11.1 G1→G0 减速检查	90
7.11.2 G1→G1 减速检查	91
7.12 自动转角超程 ; G62.....	92
7.13 攻牙模式; G63.....	96
7.14 切削模式; G64.....	96
8. 延时	97
8.1 每秒延时; G04.....	97
9. 辅助功能.....	99
9.1 辅助功能 (M8 位 BCD)	99
9.2 第 2 辅助功能 (A8 位、B8 位或 C8 位)	101
9.3 转台分度	102
10. 主轴功能.....	104
10.1 主轴功能 (S 功能)	104
10.1.1 主轴功能 (S6 位模拟)	104
10.1.2 主轴功能 (S8 位)	104
10.2 线速度恒定控制; G96、G97	105
10.3 主轴钳位速度设定; G92.....	106
10.4 主轴 • C 轴控制.....	107
10.5 主轴同期控制.....	109
10.5.1 主轴同期控制 I ; G114.1.....	110
10.5.2 主轴同期控制 II	116
10.5.3 使用主轴同期控制时的注意事项.....	121
10.6 多根主轴控制.....	123
10.6.1 多根主轴控制 I (多根主轴指令)	124
10.6.2 多根主轴控制 I (主轴选择指令)	125
10.6.3 多根主轴控制 II	127
11. 刀具功能.....	129
11.1 刀具功能 (T8 位 BCD)	129
12. 刀具偏移功能.....	130
12.1 刀具补偿	130
12.1.1 刀具补偿开始	131
12.1.2 刀具补偿开始方式扩展.....	132
12.2 刀具长度补偿.....	133
12.3 刀具刀刃磨损补偿.....	135
12.4 刀尖 R 补偿; G40、G41、G42、G46	136
12.4.1 刀刃点与补偿方向	138

12.4.2 刀尖 R 补偿的动作	141
12.4.3 刀尖 R 补偿中的其他动作	159
12.4.4 G41/G42 指令与 I、J、K 指定	167
12.4.5 刀尖 R 补偿中的插入	172
12.4.6 刀尖 R 补偿相关一般注意事项	174
12.4.7 干扰检查	175
12.5 程序补偿输入; G10、G11	180
12.6 刀具寿命管理 II	183
12.6.1 刀具寿命的计数方法	186
13. 程序辅助功能	188
13.1 车削用固定循环	188
13.1.1 横向切削循环; G77	189
13.1.2 螺纹切削循环; G78	191
13.1.3 端面切削循环; G79	194
13.2 车削用固定循环 (MELDAS 特殊格式)	197
13.3 复合型车削用固定循环	198
13.3.1 横向粗切削循环; G71	199
13.3.2 端面粗切削循环; G72	213
13.3.3 型材粗切削循环; G73	215
13.3.4 精整切削循环; G70	218
13.3.5 端面槽榫切削循环; G74	219
13.3.6 横向槽榫切削循环; G75	221
13.3.7 复合型螺纹切削循环; G76	223
13.3.8 复合型车削用固定循环 (G70~G76) 注意事项	227
13.4 复合型车削用固定循环 (MELDAS 特殊格式)	230
13.5 钻孔用固定循环; G80~G89	234
13.5.1 端面深孔钻孔循环 1; G83 (横向深孔钻孔循环 1; G87)	242
13.5.2 端面螺纹切削循环; G84 (横向攻牙循环; G88) / 端面反向攻牙循环; G84.1 (横向反向攻牙循环; G88.1)	243
13.5.3 端面镗孔循环; G85 (横向镗孔循环; G89)	247
13.5.4 深孔钻孔循环 2; G83.2	247
13.5.5 钻孔用固定循环取消; 80	250
13.5.6 钻孔用固定循环使用注意事项	250
13.6 钻孔用固定循环; G80~G89 (MELDAS 特殊格式)	252
13.6.1 起始点与 R 点基准返回; G98、G99	273
13.6.2 固定循环模式中的工件坐标设定	274
13.7 子程序控制; M98、M99	275
13.7.1 通过 M98、M99 指令调用子程序	275
13.8 变量指令	280
13.9 用户宏	282
13.9.1 用户宏; G65、G66、G66.1、G67	282
13.9.2 宏调用命令	283
13.9.3 ASCII 代码宏	291
13.9.4 变量	296

13.9.5 变量的种类	298
13.9.6 运算指令	331
13.9.7 控制指令	337
13.9.8 外部输出指令	340
13.9.9 注意事项	342
13.10 倒角/圆角 I	344
13.10.1 倒角 “,C_” (或 “I_”、“K_”、“C_”)	344
13.10.2 圆角 “,R_” (或 “R_”)	346
13.10.3 倒角/圆角扩展	348
13.10.4 倒角/圆角 R 中插入动作	350
13.11 倒角/圆角 II	352
13.11.1 倒角 “,C_” (或 “I_”、“K_”、“C_”)	352
13.11.2 圆角 “,R_” (或 “R_”)	355
13.11.3 倒角/圆角扩展	356
13.11.4 倒角/圆角中插入动作	356
13.12 直线角度指令	357
13.13 几何学	358
13.13.1 几何学 I	358
13.13.2 几何学 I B	361
13.14 程序参数输入; G10、G11	374
13.15 宏插入	375
13.16 刀具更换位置返回; G30.1~G30.5	384
14. 坐标系设定功能	387
14.1 坐标与控制轴	387
14.2 基本机械坐标系、工件坐标系与局部坐标系	388
14.3 机械原点与第 2 参考点 (原点)	389
14.4 自动坐标系设定	390
14.5 机械坐标系选择; G53	391
14.6 坐标系设定; G92	392
14.7 参考点 (原点) 返回; G28、G29	393
14.8 第 2、第 3、第 4 参考点 (原点) 返回; G30	397
14.9 简易原点返回	400
14.10 参考点校准; G27	405
14.11 工件坐标系设定及工件坐标系偏移; G54~G59 (G54.1)	406
14.12 局部坐标系设定; G52	412
14.13 工件坐标系预设; G92.1	413
14.14 旋转轴用坐标系	418
15. 保护功能	421
15.1 夹头屏障/尾架屏障检查; G22、G23	421
15.2 储存行程限制 (仅 G 代码系列 6、7)	426
16. 测量辅助功能	428
16.1 自动刀具长度测定; G37	428
16.2 跳跃功能; G31	432
16.3 多级跳跃功能 1; G31.n、G04	438

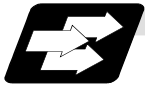
16.4 多级跳跃功能 2.....	440
16.5 可编程电流限制.....	443
附录 1. 程序参数输入 N 编号对应表	附-1
附录 2. 程序错误	附-4

1. 控制轴

1.1 坐标与控制轴

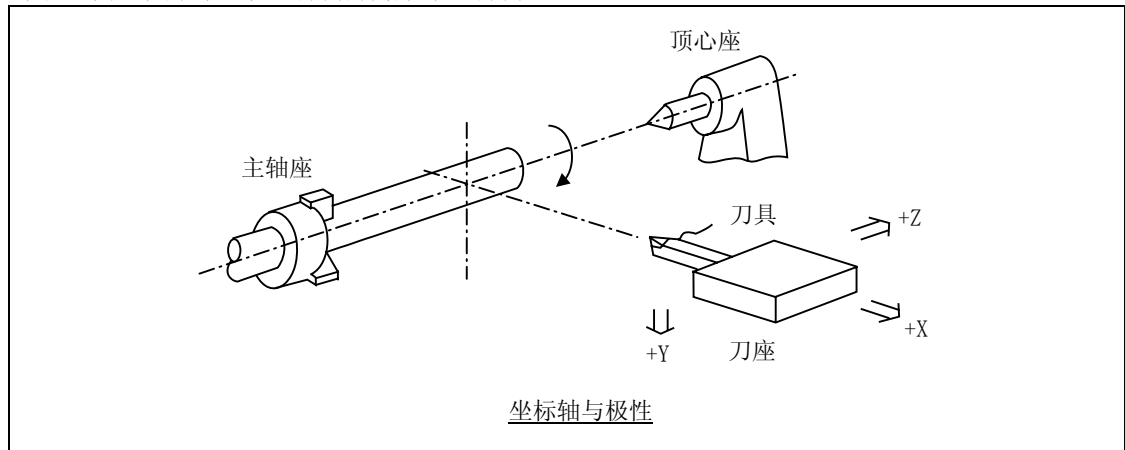
1. 控制轴

1.1 坐标与控制轴

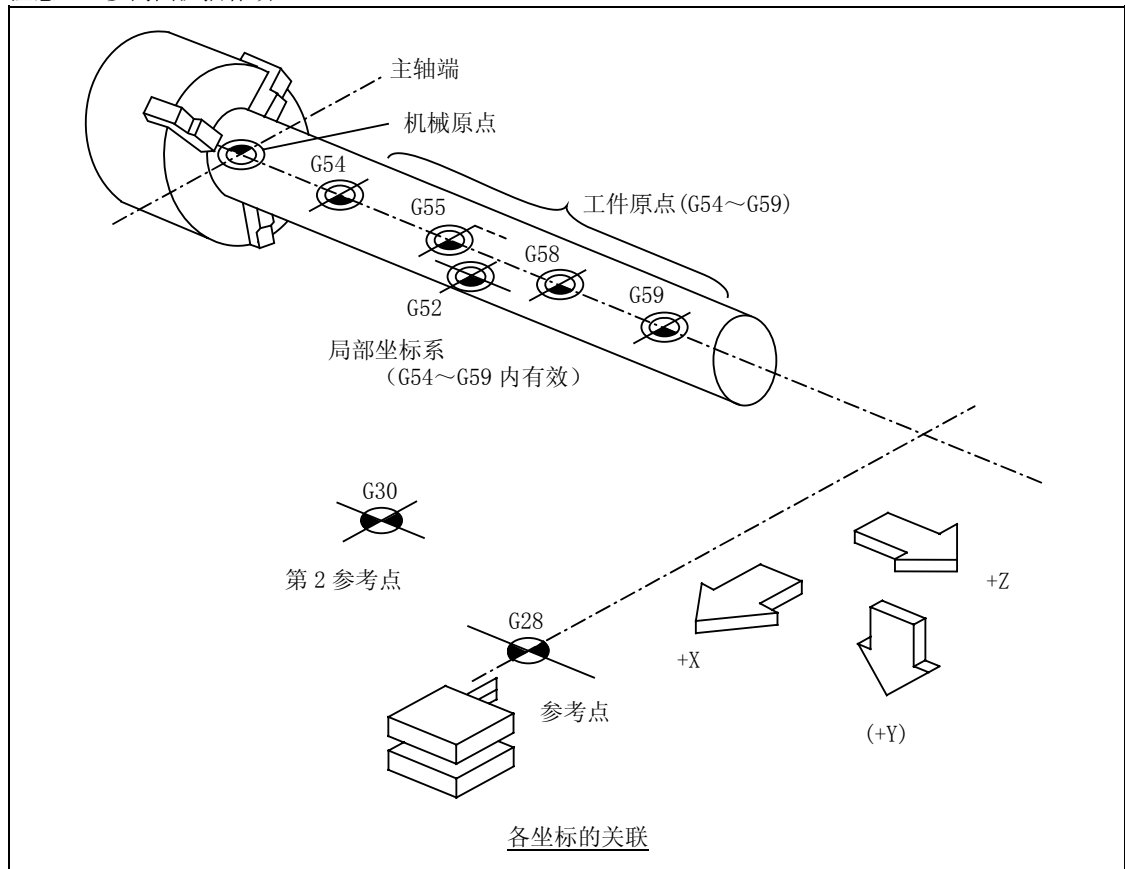


功能及目的

对于车床，轴的名称与方向如下图所示，与主轴平行的轴为Z轴，其正方向为刀座离开主轴座的方向，而与Z轴成直角的轴为X轴，其正方向为离开Z轴的方向。



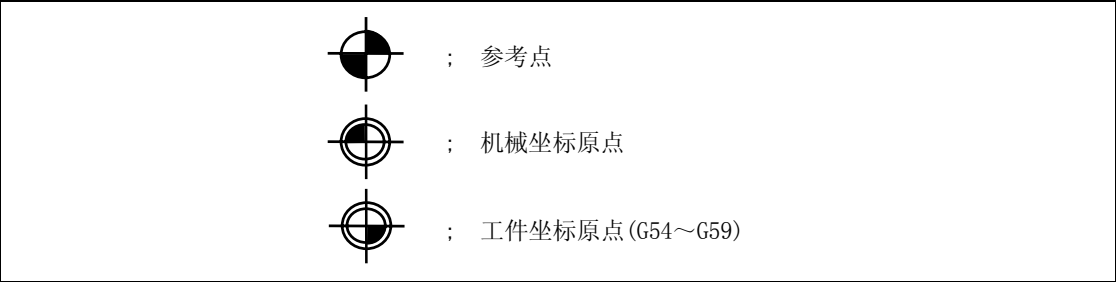
在车床上是使用右手系统的坐标，所以在如上图所示的状态下，与X、Z轴直角相交的Y轴以下图的向下方向为正方向。从Y轴正方向看，X、Z平面上的圆弧是以顺时针旋转、逆时针旋转的方式加以表现，所以需加以注意。（参阅圆弧插补项）



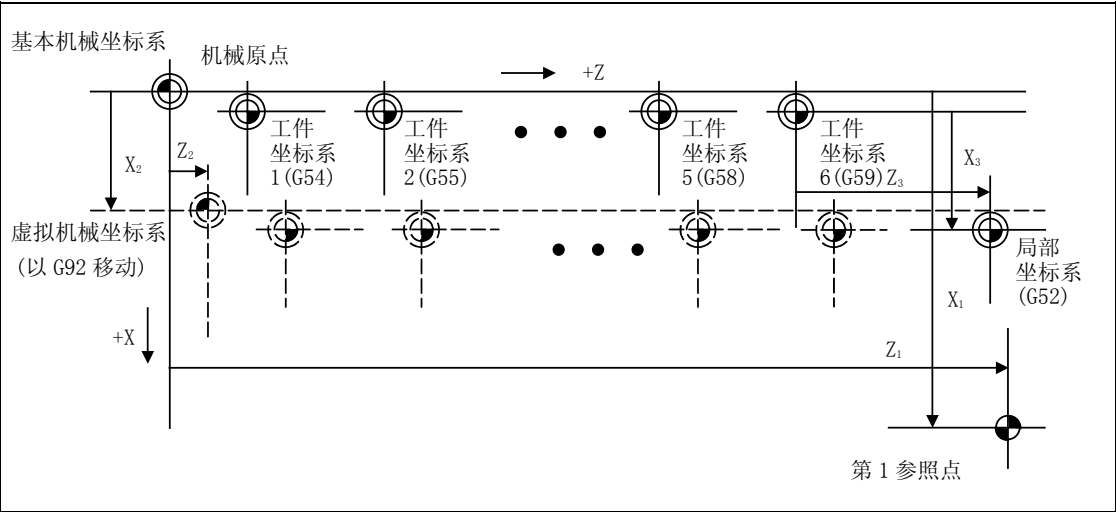
1.2 坐标系与坐标原点标记



功能及目的



在参考点重新启动完成时，参阅参数，自动设定基本机械坐标系及工件坐标系（G54~G59）。
此时，设定基本机械坐标系，以使第1参考点从基本机械坐标原点（机械原点）移动到参数中所指定的位置。



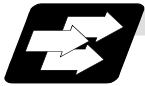
局部坐标系（G52）在工件坐标系1~6所指定的坐标系中有效。
另外，可通过G92指令，将基本机械坐标系移位，作为虚拟机械坐标系，此时，工件坐标系1~6也同时移位。

2. 输入指令单位

2.1 输入指令单位

2. 输入指令单位

2.1 输入指令单位



功能及目的

是通过MDI输入、指令纸带发出指令的程序中的移动量单位。以mm、inch、度（°）为单位表示。

2.2 输入设定单位



功能及目的

是诸如补偿量等各轴通用的设定数据的单位。

可通过参数从以下类型中为各轴分别选择输入指令单位，选择轴共用的输入设定单位。（关于设定详情，请参阅操作说明书。）

(1) E60

	输入单位参数	直线轴				旋转轴 (°)
		公制		英制		
		直径指令	半径指令	直径指令	半径指令	
输入指令单位	#1015 cunit = 10	0.001	0.001	0.0001	0.0001	0.001
最小移动单位	#1003 iunit = B	0.0005	0.001	0.00005	0.0001	0.001
输入设定单位	#1003 iunit = B	0.001	0.001	0.0001	0.0001	0.001

(2) E68

	输入单位参数	直线轴				旋转轴 (°)
		公制		英制		
		直径指令	半径指令	直径指令	半径指令	
输入指令单位	#1015 cunit = 10	0.001	0.001	0.0001	0.0001	0.001
	= 1	0.0001	0.0001	0.00001	0.00001	0.0001
最小移动单位	#1003 iunit = B	0.0005	0.001	0.00005	0.0001	0.001
	= C	0.00005	0.0001	0.000005	0.00001	0.0001
输入设定单位	#1003 iunit = B	0.001	0.001	0.0001	0.0001	0.001
	= C	0.0001	0.0001	0.00001	0.00001	0.0001

(注 1) 英制/公制的切换，可通过参数画面进行切换（#1041 I_inch；仅接通电源时有效），也可通过 G 指令（G20、G21）进行切换。

但是，通过 G 指令进行的切换仅能切换输入指令单位，无法切换输入设定单位。因此，刀具偏移量等补偿量及变量数据，请预先按照输入设定单位加以设定。

(注 2) 公制单位与英制单位不能并用。

(注 3) 当在输入指令单位不同的轴之间进行圆弧插补时，使用输入设定单位指定中心指令（I、J、K）及半径指令（R）。（为了防止混乱，请以带小数点的形式进行指定）

2. 输入指令单位

2.1 输入指令单位



详细说明

(1) 各种数据的单位

决定参数设定单位及程序指令单位、PLC轴・DDB・手轮脉冲等外部接口的单位的，是输入设定单位。
以下给出了随着输入设定单位的变化，各种数据的单位发生变化的规则。本表的对象轴为NC轴・PLC轴。

数据	公制/ 英制	设定值	输入设定单位		备注
			B (0.001)	C (0.0001)	
速度数据 例: rapid	公制	20000 (mm/min)	20000	200000	(注 1)
		设定范围	1~999999	1~999999	
	英制	2000 (inch/min)	2000	20000	
		设定范围	1~999999	1~999999	
位置数据 例: SoftLimit+	公制	123.123 (mm)	123.123	123.1230	
		设定范围	±99999.999	±9999.9999	
	英制	12.1234 (inch)	12.1234	12.1234	
		设定范围	±9999.9999	±999.99999	
插补单位数据	公制	10 (μm)	20	20	
		设定范围	±9999	±9999	
	英制	0.01 (inch)	20	20	
		设定范围	±9999	±9999	

(注 1) 当输入设定单位为 B (0.001) 时，以 mm/min、inch/min 为单位进行设定。

E68 上也可以使用 C (0.0001) 作为输入设定单位。当使用 C 时，以 10 倍于使用 B 时的值进行设定。

(2) 程序指令

程序指令单位如上表。

对于有小数点的数据，当输入设定单位缩小时，整数部分的位数减少，小数点部分的位数增加。

对于没有小数点的数据，当发出位置指令时，受到输入设定单位与输入指令单位的影响。

关于进给速度，当输入设定单位缩小时，整数部分的位数仍然保持不变，而小数点部分的位数增加。

3. 数据格式

3.1 纸带代码



功能及目的

本控制装置所使用的指令信息，由字母（A、B、C……Z）、数字（0、1、2……9）、符号（+、-、/……）构成，这些字母、数字、符号统称为字符。在纸带上，这些字符表现为8个孔的有无的组合。这样被表现出的，称为代码。
本控制装置使用ISO代码（R-840）。

- （注1）在运转中，如果指定了“纸带代码一览表”中所没有的代码，则发生程序错误（P32）。
- （注2）单节分隔中表示单节结束的（EOB/LF），虽然简化显示为“；”，但是在实际的编程中，如果使用“；”键，则无法编写出正确的程序。在编程中请使用下表中的键。

△ 注意

△ “；” “EOB” 及 “%” “EOR” 是用于说明的表述。在 ISO 中，与 “；” “EOB” 相对应的实际代码为 “CR, LF” 或 “LF”。“%” “EOR” 在 ISO 中为 “%”。

在编辑画面中所创建的程序，以 “CR, LF” 的形式被储存在 NC 内存中，而通过外部设备创建的程序则可能会是 “LF” 的形式。

对于 EIA，则为 “EOB（End Of Block）” 和 “EOR（End Of Record）”。



详细说明

- （1）在编程中请使用下表中的键。

EOB/EOR使用键与显示

使用键 \ 使用代码	ISO	画面显示
单节结束	LF 或 NL	；
录制结束	%	%

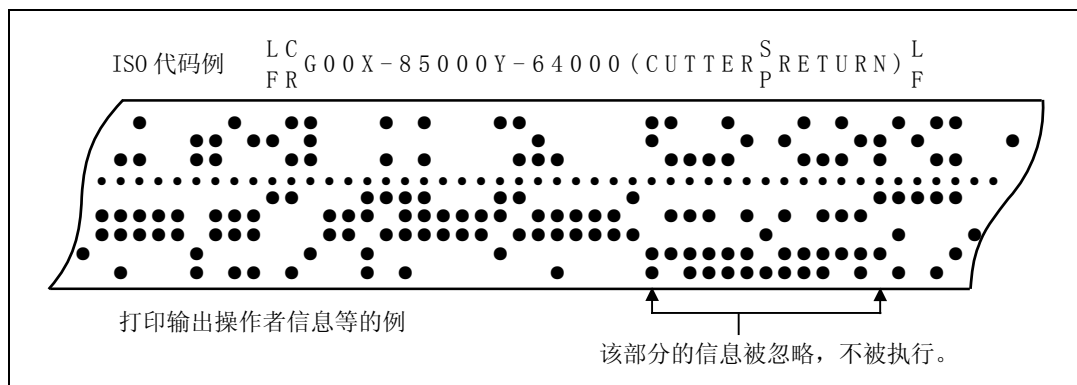
- （2）有含义信息分隔（标签跳跃功能）
利用纸带进行自动运转、保存到内存、呼叫动作时，在接通电源或重新启动时，跳跃纸带信息内的首个EOB（；）代码前的信息。即，纸带中的有含义信息，是指重新启动后的首个EOB（；）代码之后，从出现文字或符号代码，到发出重新启动指令为止的区间内的信息。

(3) 控制输出、控制输入

在ISO代码中，从控制输出“（”到控制输入“）”或“；”之间的所有信息被跳跃。但是，会显示在设定显示装置上。因此，在此范围内，可加入指令纸带的名称编号等与控制无关的信息。

另外，在记忆纸带时，这一范围内的信息（“纸带代码一览表”的B除外）也进行记忆。

接通电源时为控制输入状态。

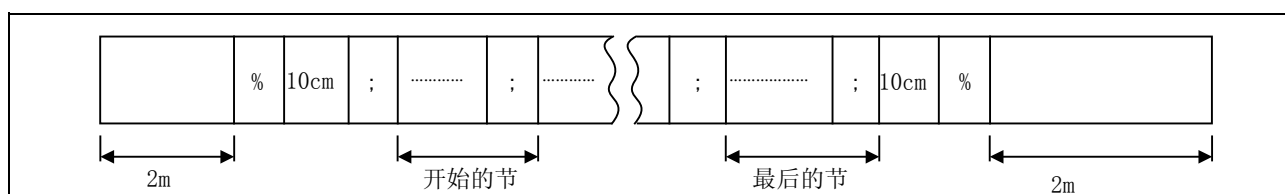


(4) EOR (%) 代码

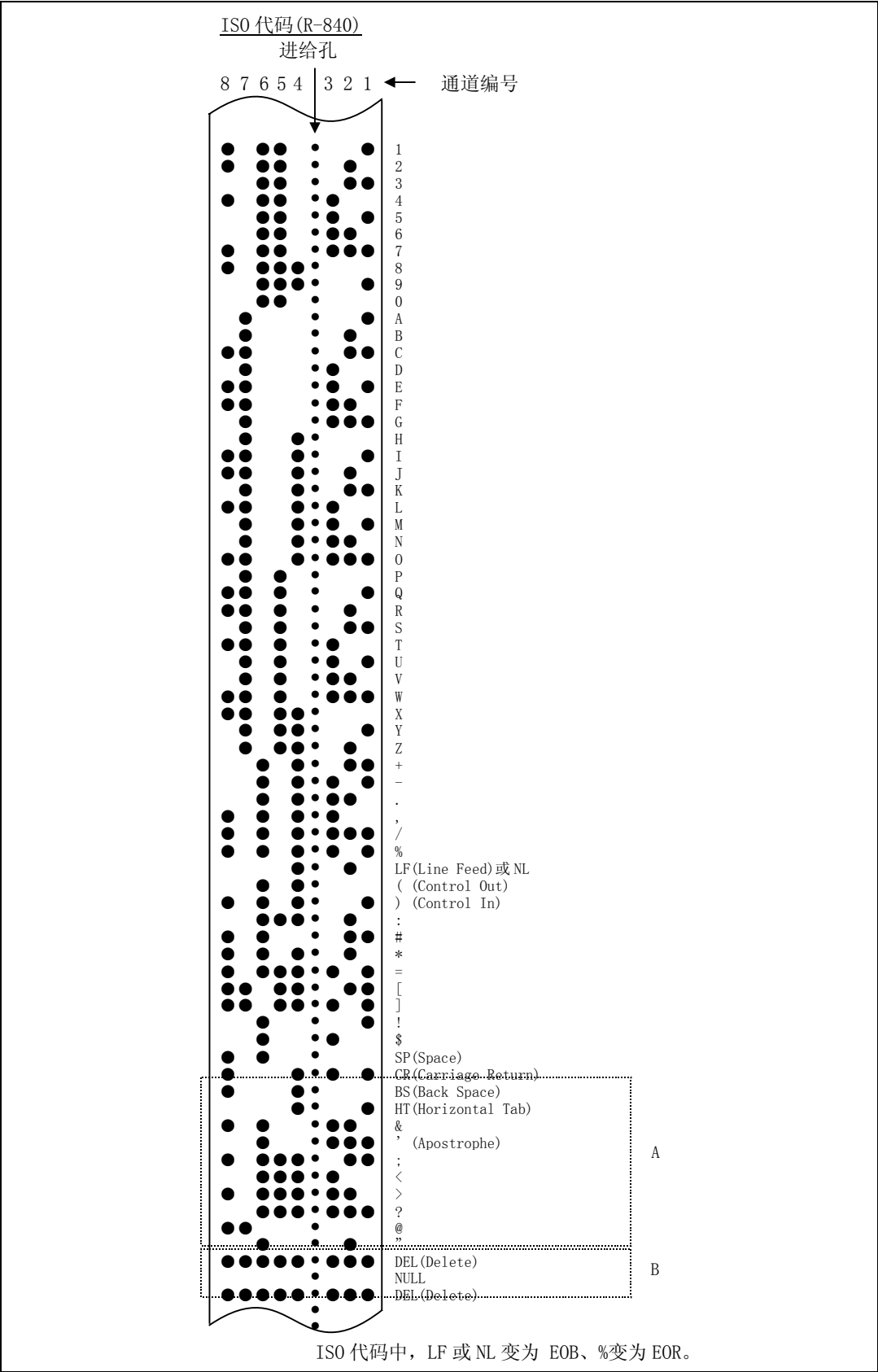
录制结束代码一般打在纸带的两端，具有如下的功能。

- (a) 回卷时回卷停止（带卷取装置时）
- (b) 纸带呼叫时的回卷起动（带卷取装置时）
- (c) 纸带记忆时记忆结束

(5) 用于纸带运转的纸带制作要领（使用卷取装置时）



不使用卷取装置时，无需保留两端的2m的空白以及开头的EOR (%)。



A纸带被记忆,但是在运转时(除非位于注释部位)发生错误。

B为无作业代码,总是被跳跃。(不作为校验V检查的对象。)

纸带代码一览表

3.2 程序格式



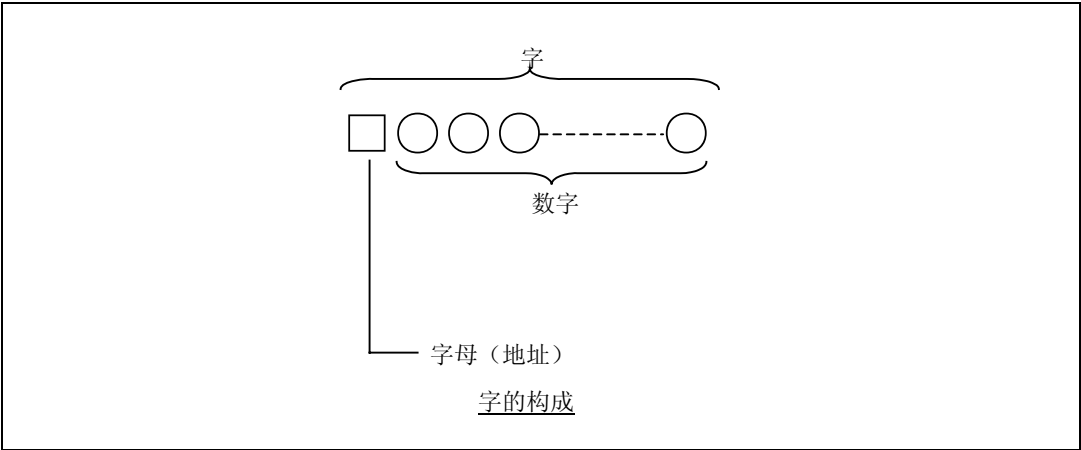
功能及目的

向控制装置提供控制信息时的规定样式，称为程序格式，本控制装置所使用的格式称为字地址格式。



详细说明

- (1) 字与地址
- 字（语句）为按照一定顺序排列的字符的集合，以字为单位，对信息进行处理，让机械执行某个特定的动作。
- 在本控制装置中，字由1个字母和接在其后的若干位数字构成。
- （也可以在数字的开头添加-符号。）



字的开头字母称为地址，它规定了后面的数值信息所具有的含义。

关于本控制装置中使用的字的种类及数字的有效位数，请参阅“格式详情”。

- (2) 单节
- 若干个字的集合称为单节，包含让机械执行某一特定动作所需的信息，以单节为单位，构成完整的指令。单节的结尾通过EOB（End of Block）代码表示分隔。
- (3) 程序
- 若干个单节的集合，形成一个程序。

3.3 程序地址检查



功能及目的

在运行加工程序时，可以以字为单位，对程序进行检查。



详细说明

(1) 地址检查

简便的以字为单位进行检查。当出现连续的字母时，发生程序错误(P32)。通过参数“#1227 aux11/bit4”选择是否进行地址检查。

但是，以下的场合不是错误。

- 预约字
- 注释文字

(2) 字范围检查

当在字数据部分使用计算公式时，检查1个字数据是否用“[]”括起来。当没有用括号括起来时，发生程序错误(P33)。

通过参数“#1274 ext10/bit7”选择是否进行字范围检查。



程序例

(1) 地址检查的程序例

(例1) 字母后面没有数字时

G28 X; → 发生错误请变更为“G28 X0; ”等。

(例2) 不正确的字符串时

TEST; → 发生错误。请变更为“(TEST); ”等。

(2) 字范围检查的程序例

发生错误(P33)。 → 请进行如下的变更。

X-100.-#24

X-[100.+#24]

Z+#500*TAN[#1]

Z+[#500*TAN[#1]]

Y-SIN[60.]

Y-[SIN[60.]]

括号的最大嵌套层数为5层。

如果嵌套了6层以上的括号，则发生程序错误(P280)。

3.4 纸带记忆格式



功能及目的

- (1) 记忆纸带与记忆区间（ISO、EIA自动判别）

记忆到内存中的纸带代码，与纸带运转时相同，可同时使用ISO代码、EIA代码，根据重新启动后的起始EOB代码自动判别ISO/EIA。

记忆到内存中的范围为重新启动之后，从开头的EOB到EOR代码之间的字符串。

在上述记忆区间中，实际被记忆到内存中的仅为“3.1 纸带代码”的“纸带代码一览表”中所示的有含义代码。其他的代码被跳跃，不被记忆到内存中。

另外，从控制输入“(”到控制输入“)”之间的数据也被记忆到内存中。

3.5 可选单节的跳跃



功能及目的

是选择性的跳跃加工程序中，以“/”（反斜杠）代码开头的特定单节的功能。



详细说明

- (1) 当可选单节的跳跃开关ON时，单节开头带有“/”（反斜杠）的单节被跳跃，开关OFF时，执行可选单节。

此时，不管可选单节开关的ON/OFF状态如何，校验检查均有效。

例如，当某些加工对象需要执行所有的单节，而其他加工对象不进行特定的单节的加工时，可通过制件在该特定单节的开头带有“/”代码的指令纸带，使用1根纸带加工出不同的部件。



跳跃可选节的使用注意事项

- (1) 可选单节的跳跃用的“/”代码请务必附加在单节的开头。如果插入到单节的中间，则作为用户宏的除法运算命令加以使用。

（例）N20 G1 X25. /Z25. ； 错误（用户宏的除法运算命令，此时为程序错误）

/N20 G1 X25. Z25. ； 正确

- (2) 校验检查（H及V）与可选单节的跳跃开关的状态无关，始终进行。

- (3) 在预读缓存之前，进行可选单节的跳跃处理。

因此，无法跳跃被读入到预读缓存的单节之前的单节。

- (4) 即使在顺序编号呼叫中，这些功能也有效。

- (5) 在纸带记忆、纸带输出中，不管可选单节的跳跃开关的状态如何，带“/”代码的单节全部进行输入输出。

3. 数据格式

3.6 程序编号、顺序编号及单节编号：O、N

3.6 程序编号、顺序编号及单节编号：O、N



功能及目的

这些编号是用于监视加工程序的执行状况，以及调用加工程序及加工程序中的特定工序。

- (1) 程序编号是与各工件相对应，或是以子程序为单位，对程序进行分类的编号，使用地址“O”（字母O）和接在后面的最多8位数值进行指定。
- (2) 顺序编号是附加在构成加工程序的各指令单节上的编号，使用地址“N”和接在后面的最多5位数值进行指定。
- (3) 单节编号是在内部自动创建的编号，每次读入程序编号或顺序编号时，重新启动为0，如果之后读入的单节中没有指定程序编号或顺序编号，则逐一累加。

因此，如下表所示，加工程序的所有单节可通过程序编号、顺序编号及单节编号的组合，决定其唯一性。

加工程序	MONITOR 显示		
	程序 编号	顺序 编号	单节 编号
O12345678 (DEMO, PROG) ;	12345678	0	0
N100 G00 G90 X120. Z100. ;	12345678	100	0
G94 S1000;	12345678	100	1
N102 G71 P210 Q220 I0.2 K0.2 D0.5 F600 ;	12345678	102	0
N200 G94 S1200 F300 ;	12345678	200	0
N210 G01 X0 Z95. ;	12345678	210	0
G01 X20. ;	12345678	210	1
G03 X50. Z80. K-15. ;	12345678	210	2
G01 Z55. ;	12345678	210	3
G02 X80. Z40. I15. ;	12345678	210	4
G01 X100. ;	12345678	210	5
G01 Z30. ;	12345678	210	6
G02 Z10. K-15. ;	12345678	210	7
N220 G01 Z0 ;	12345678	220	0
N230 G00 X120. Z150. ;	12345678	230	0
N240 M02 ;	12345678	240	0
%	12345678	240	0

3.7 校验H/V



功能及目的

作为用于检查纸带制作是否正确的手段之一，可使用校验检查。根据检查纸带上的打孔代码是否有错误，即，根据检查是否有打孔错误，可分为校验H与校验V两种。

(1) 校验H

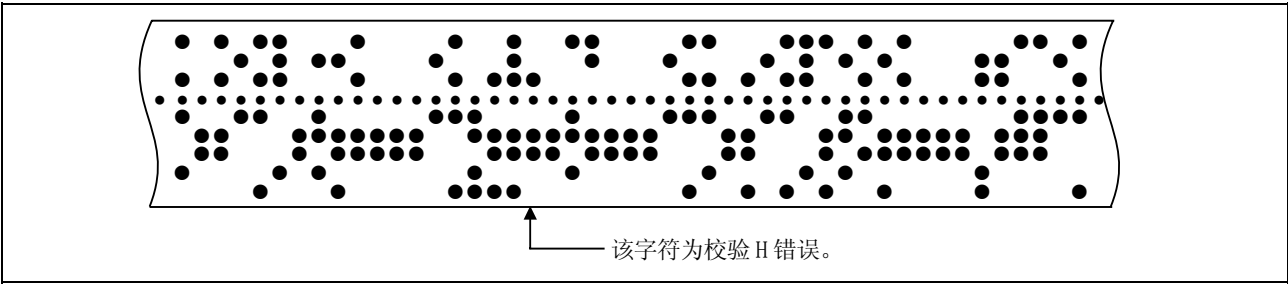
校验H检查是检查构成1个字符的孔的个数，所以在纸带运转、纸带输入、顺序编号呼叫等任何场合下均能进行。

如下条件下为校验H错误。

(a) ISO代码

当有含义信息区间内存在奇数个孔的代码时。

(例1) 校验H错误例（ISO代码时）



校验H错误时，纸带在报警代码后停止。

(2) 校验V

当I/O参数#9n15（n为设备编号，为1~5）校验V为“1”时，在纸带运转、纸带输入或顺序编号呼叫中实施校验V检查。但是，在内存运转时，不进行校验V检查。

在如下条件下，为校验V错误。

在有含义信息区间中，当从纸带垂直方向的首个有含义代码到EOB（；）之间的代码数量为奇数个时，也就是说，1单节内的字符数量为奇数个时。

校验V错误时，纸带停止在EOB（；）的后一个代码处。

(注1) 在进行校验V检查时，纸带代码中存在作为1个字符进行计数的代码和不作为1个字符进行计数的代码。详情请参阅“3.1 纸带代码”的“纸带代码一览表”。

(注2) 从首个EOB代码到出现地址代码或“/”代码为止，这一区间内的空格代码也是校验V的计数对象。

3.8 G代码系列



功能及目的

G代码包括G代码系列2、3、6、7共4个系列。通过参数“#1037 cmdtyp”的设定决定选择哪一个系列。

cmdtyp	G 代码系列
3	系列 2
4	系列 3
7	系列 6
8	系列 7

关于G功能的说明，以G代码系列3为目标，进行说明。

（注 1）指定了 G 代码一览表中所没有的 G 代码，则发生程序错误（P34）。

（注 2）指定无附加规格的 G 代码，则发生警报。

G代码系列一览表

G 代码系列				组	功能名	E60	E68
2	3	6	7				
△G00	△G00	△G00	△G00	01	定位	○	○
△G01	△G01	△G01	△G01	01	直线插补	○	○
G02	G02	G02	G02	01	圆弧插补 CW	○	○
G03	G03	G03	G03	01	圆弧插补 CCW	○	○
G04	G04	G04	G04	00	停止	○	○
		G07.1/ G107	G07.1/ G107	19	圆筒插补	-	○
G09	G09	G09	G09	00	精确停止检查	○	○
G10	G10	G10	G10	00	程序参数/补偿输入	○	○
G11	G11	G11	G11	00	程序参数输入取消	○	○
		G12.1/ G112	G12.1/ G112	19	极坐标插补 ON	-	○
		G13.1/ G113	G13.1/ G113	19	极坐标插补 取消	-	○
△G17	△G17	△G17	△G17	02	平面选择 X-Y	○	○
△G18	△G18	△G18	△G18	02	平面选择 Z-X	○	○
△G19	△G19	△G19	△G19	02	平面选择 Y-Z	○	○
△G20	△G20	△G20	△G20	06	英制指令	○	○
△G21	△G21	△G21	△G21	06	公制指令	○	○
G22	G22			04	禁区检查 ON	○	○
*G23	*G23			04	禁区检查 OFF	○	○
		G22	G22	00	软件限制 ON	○	○
		G23	G23	00	软件限制 OFF	○	○

3. 数据格式

3.8 G 代码系列

G 代码系列				组	功能名	E60	E68
2	3	6	7				
G27	G27	G27	G27	00	参考点比对	○	○
G28	G28	G28	G28	00	自动参考点返回	○	○
G29	G29	G29	G29	00	开始点返回	○	○
G30	G30	G30	G30	00	第 2、3、4 参考点返回	○	○
G30.1	G30.1	G30.1	G30.1	00	刀具更换位置返回 1	○	○
G30.2	G30.2			00	刀具更换位置返回 2	○	○
G30.3	G30.3			00	刀具更换位置返回 3	○	○
G30.4	G30.4			00	刀具更换位置返回 4	○	○
G30.5	G30.5			00	刀具更换位置返回 5	○	○
G31	G31	G31	G31	00	跳跃/多级跳跃 2	○	○
G31.1	G31.1	G31.1	G31.1	00	多极跳跃 1-1	○	○
G31.2	G31.2	G31.2	G31.2	00	多极跳跃 1-2	○	○
G31.3	G31.3	G31.3	G31.3	00	多极跳跃 1-3	○	○
G32	G33	G32	G33	01	螺纹切削	○	○
G34	G34	G34	G34	01	可变导程螺纹切削	○	○
G37	G37	G36/G37	G36/G37	00	自动刀具长度测定	○	○
*G40	*G40	*G40	*G40	07	刀尖 R 补偿 取消	○	○
G41	G41	G41	G41	07	刀尖 R 补偿 左	○	○
G42	G42	G42	G42	07	刀尖 R 补偿 右	○	○
G46	G46	G46	G46	07	刀尖 R 补偿 (方向自动决定) ON	○	○
G43.1	G43.1	G43.1	G43.1	20	第 1 主轴控制模式	-	○
G44.1	G44.1	G44.1	G44.1	20	第 2 主轴控制模式	-	○
G47.1	G47.1	G47.1	G47.1	20	全部主轴同期控制模式	-	○
G50	G92	G50	G92	00	坐标系设定/主轴钳位速度设定	○	○
G52	G52	G52	G52	00	局部坐标系设定	○	○
G53	G53	G53	G53	00	机械坐标系选择	○	○
*G54	*G54	*G54	*G54	12	工件坐标系选择 1	○	○
G55	G55	G55	G55	12	工件坐标系选择 2	○	○
G56	G56	G56	G56	12	工件坐标系选择 3	○	○
G57	G57	G57	G57	12	工件坐标系选择 4	○	○
G58	G58	G58	G58	12	工件坐标系选择 5	○	○
G59	G59	G59	G59	12	工件坐标系选择 6	○	○
G54.1	G54.1	G54.1	G54.1	12	工件坐标系选择扩展 48 组	○	○
G61	G61	G61	G61	13	精确定位校验	○	○
G62	G62	G62	G62	13	自动角超程	○	○
G63	G63	G63	G63	13	钻孔模式	○	○
*G64	*G64	*G64	*G64	13	切削模式	○	○

3. 数据格式

3.8 G 代码系列

G 代码系列				组	功能名	E60	E68
2	3	6	7				
G65	G65	G65	G65	00	用户宏 单纯调用	○	○
G66	G66	G66	G66	14	用户宏 模态调用 A	○	○
G66.1	G66.1	G66.1	G66.1	14	用户宏 模态调用 B	○	○
*G67	*G67	*G67	*G67	14	用户宏 模态调用 取消	○	○
G70	G70	G70	G70	09	精整循环	○	○
G71	G71	G71	G71	09	横向粗切削循环	○	○
G72	G72	G72	G72	09	端面粗切削循环	○	○
G73	G73	G73	G73	09	形材粗切削循环	○	○
G74	G74	G74	G74	09	端面平削循环	○	○
G75	G75	G75	G75	09	长度平削循环	○	○
G76	G76	G76	G76	09	复合型螺纹切削循环	○	○
G90	G77	G90	G77	09	横向切削固定循环	○	○
G92	G78	G92	G78	09	螺纹切削固定循环	○	○
G94	G79	G94	G79	09	端面切削固定循环	○	○
*G80	*G80	*G80	*G80	09	钻孔固定循环取消	○	○
G79	G83.2	G79	G83.2	09	深孔钻孔循环 2	○	○
G83	G83	G83	G83	09	深孔钻孔循环 (Z 轴)	○	○
G84	G84	G84	G84	09	攻牙循环 (Z 轴)	○	○
G85	G85	G85	G85	09	镗孔循环 (Z 轴)	○	○
G87	G87	G87	G87	09	深孔钻孔循环 (X 轴)	○	○
G88	G88	G88	G88	09	攻牙循环 (X 轴)	○	○
G89	G89	G89	G89	09	镗孔循环 (X 轴)	○	○
G84.1	G84.1			09	反向攻牙循环 (Z 轴)	○	○
G88.1	G88.1			09	反向攻牙循环 (X 轴)	○	○
G50.3	G92.1	G50.3	G92.1	00	工件坐标系预设	-	○
△G96	△G96	△G96	△G96	17	线速度恒定控制 ON	○	○
△G97	△G97	△G97	△G97	17	线速度恒定控制 OFF	○	○
△G98	△G94	△G98	△G94	05	非同步进给 (每分钟进给)	○	○
△G99	△G95	△G99	△G95	05	同步进给 (每转进给)	○	○
-	△G90	-	△G90	03	绝对值指令	○	○
-	△G91	-	△G91	03	增量值指令	○	○
-	*G98	-	*G98	10	固定循环 起始等级返回	○	○
-	G99	-	G99	10	固定循环 R 点等级返回	○	○
G114.1	G114.1			00	主轴同期控制	-	○

- (注 1) *标记表示在接通电源时, 或执行将模式初始化的重新启动时, 各组内选中的 G 代码。
- (注 2) △标记表示在接通电源时, 或执行将模式初始化的重新启动时, 可作为初始状态进行参数选择的 G 代码。但是, 英制/公制切换仅可在接通电源时选择。
- (注 3) 当指定了同一组内的 2 个以上的 G 代码时, 最后的 G 代码生效。
- (注 4) 本 G 指令一览表为原本的 G 指令的一览表。根据机械不同, 可能会通过调用 G 代码宏, 执行与原本的 G 指令不同的动作。请在参阅机械制造商所刊行的说明书的基础上, 加以确认。
- (注 5) 各重新启动输入中是否进行模式初始化, 各不相同。
- (1) “重新启动1”
当重新启动起始参数 (#1151 rstinit) 为 ON 时, 将模式初始化。
 - (2) “重新启动2”及“重新启动&回卷”
在输入信号时, 将模式初始化。
 - (3) 紧急停止解除时的重新启动
根据“重新启动1”。
 - (4) 开始参考点返回等个别功能时, 自动进行重新启动。
根据“重新启动&回卷”。

△ 注意

△ “G 后无数值”的指令, 被看作为“G00”。

3.9 加工前的注意事项



加工前注意事项

△ 注意

- △ 在创建加工程序时, 请选择适当的加工条件, 注意不要超过机械、NC 的性能、容量、限制。例题已考虑到了加工条件。
- △ 请在进行实际加工之前进行空运转, 进行加工程序、刀具偏移量、工件偏移量等的确认。

4. 缓存寄存器

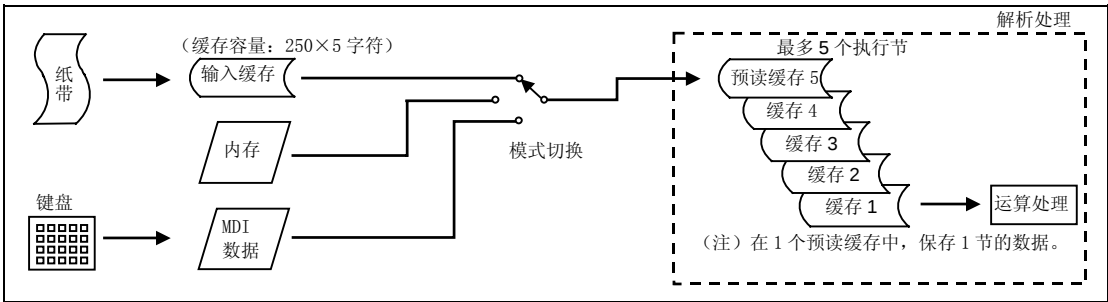
4.1 输入缓存



功能及目的

在纸带运转或RS-232C运转中，如果预读缓存为空，则立即从输入缓存的内容中将单个单节的数据转移到预读缓存中，如果输入缓存的记忆数据少于250×4字符，则继续从纸带读入后面的数据（最多250字符），保存在输入缓存中。

该缓存是用于消除因纸带读带机的读取时间而导致的动作延迟，平滑单节与单节之间的连接。
但是，当单节的执行时间比下一单节的纸带读取时间更短时，失去该预读效果。



输入缓存的内存容量为250×5字符（含EOB代码）。

- (1) 以250字符为单位更新输入缓存寄存器的内容。
- (2) 被读入到输入缓存中的代码仅为有含义信息区间的有含义代码。
- (3) 被夹在控制输入、控制输入中的代码（含“（”与“）”）及可选单节跳跃ON时，从/代码到EOB代码之间的代码也被读入输入缓存中。
- (4) 输入缓存的内容通过重新启动被清除。

（注 1）输入缓存的大小（250 字符）因机种而异。

4.2 预读缓存



功能及目的

通常，在自动运转时，为了圆滑的进行程序解析处理而进行单个单节的预读，但是在进行刀尖R补偿时，为了进行包括干涉检查在内的交点计算，进行最多5个执行单节的预读。

预读缓存的规格如下。

- (1) 记忆单个单节的数据。
- (2) 被记忆到预读缓存中的数据，仅为有含义信息区间内的有含义信息。
- (3) 夹在控制输出、控制输入之间的代码及可选单节跳跃ON时，从/（反斜杠）代码到EOB代码之间的内容不被读入到预读缓存中。
- (4) 预读缓存的内容通过重新启动被清除。
- (5) 在连续运转中，当单节ON时，预读缓存记忆到下一单节的数据为止。



其他注意事项

- (1) 连续执行程序与单节执行时，可选单节跳跃等外部控制信号的有效/无效时机各不相同。
- (2) 当通过M指令打开、关闭可选单节跳跃等外部控制信号时，对于通过缓存寄存器预读的程序，外部控制动作不会生效。
- (3) 对于进行外部控制的M指令，禁止预读，然后重新计算的方法如下。
判别通过PLC进行外部控制的M指令，打开PLC→NC接口平台的“重新计算请求”。（“重新计算请求”打开，则重新处理做了预读处理的程序。）

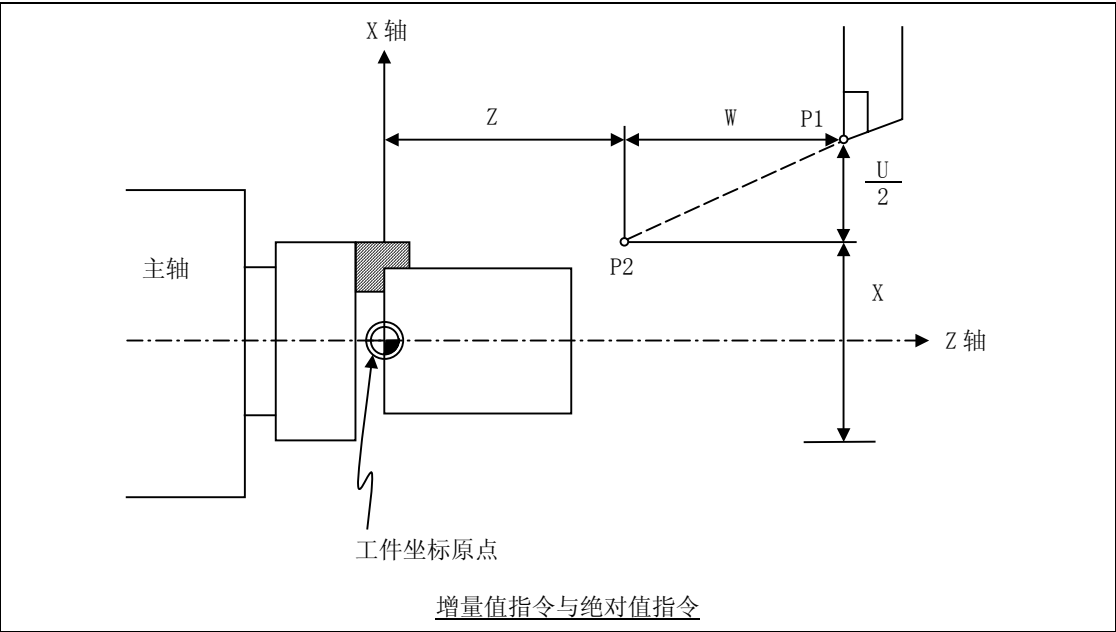
5. 位置指令

5.1 增量值指令/绝对值指令



功能及目的

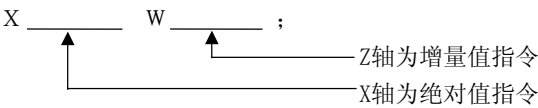
在指定刀具移动量时，有增量值方式与绝对值方式2种方法。
关于移动点的坐标，增量值方式是以距当前点的距离发出指令，而绝对值指令则是以距坐标原点的距离发出指令。下图表示刀具从点P1移动到点P2。



关于X轴与Z轴，当参数“#1076 AbsInc”为1时通过地址，为0时通过G代码（G90/G91）区分增量值指令与绝对值指令。
附加轴（C或Y轴）也同样通过地址或G代码加以区分。

		指令方法	备 注
绝对值	X 轴	地址 X	• 在“#1013 axname”及“#1014 incax”中设定地址与轴的对应。 • 在同一单节中可并用绝对值与增量值。
	Z 轴	地址 Z	
	C/Y 轴	地址 C/Y	
增量值	X 轴	地址 U	
	Z 轴	地址 W	
	C/Y 轴	地址 H/V	

（例）



（注 1）当参数“#1076 AbsInc”为 1，在增量指令地址中使用了 H 时，将 M98、G114.2 及 G10 L50 模态中的单节地址 H 作为各指令的参数使用，不进行轴移动。

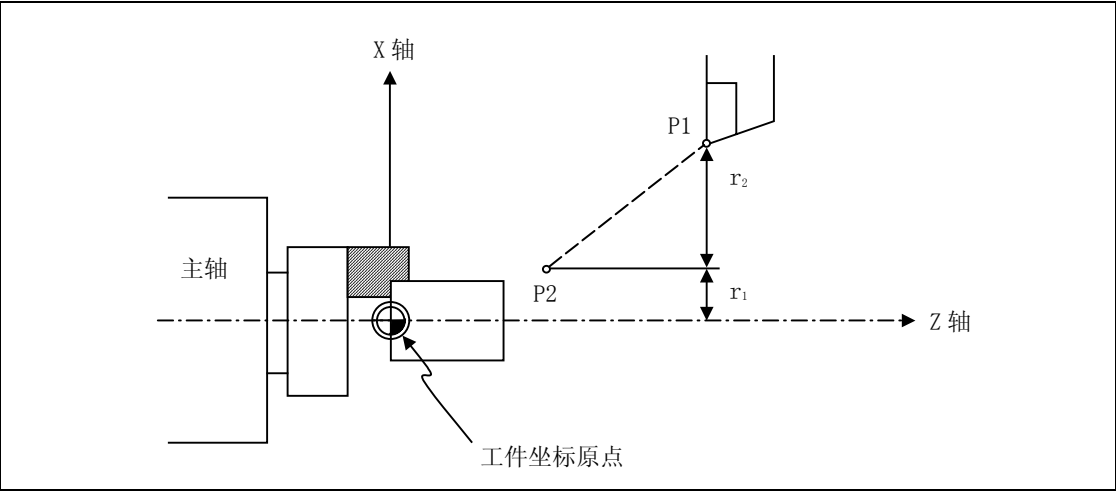
5.2 半径指令/直径指令



功能及目的

以车床加工的工件截面为圆形时，使用圆形的直径值或半径值作为X轴方向的移动指令。当使用半径指令时，刀具仅移动所指定的量，而当使用直径指令时，则在X轴方向仅移动所指定量的1/2，而在Z轴方向上仅移动指定的量。

对于本装置，可根据参数（#1019 dia）的设定，选择半径指令或直径指令。在下图中，给出了刀具从点P1移动到点P2时的指令要领。



X 指令		U 指令		备 注
半径	直径	半径	直径	
$X = r_1$	$X = 2r_1$	$U = r_2$	$U = 2r_2$	

半径指令与直径指令



注意事项・限制事项

- （1）在上述例中，刀具为从P1移动到P2，即向X轴的负方向移动，所以当采用增量值指令时，指令的数值上带有负号。
- （2）在本书中，为方便起见，假定X、U均采用直径指令，加以说明。

5.3 英制指令/公制指令切换：G20、G21



功能及目的

可利用G20/G21指令切换英制指令与公制指令。



指令格式

G20/G21;	
G20	: 英制指令
G21	: 公制指令



详细说明

G20/G21仅切换指令单位，无法切换输入单位。
另外，G20/G21的切换仅对直线轴有效。对旋转轴无效。

(例) 输入指令单位与 G20/G21 的关系 (小数点输入类型 1 时)

(1) E60

轴	输入指令单位 cunit	指令例	公制输出 (#1016 iout=0)		英制输出 (#1016 iout=1)	
			G21	G20	G21	G20
X	10	X100;	0.100 mm	0.254 mm	0.0039 inch	0.0100 inch
Z	10	Z100;	0.100 mm	0.254 mm	0.0039 inch	0.0100 inch

(2) E68

轴	输入指令单位 cunit	指令例	公制输出 (#1016 iout=0)		英制输出 (#1016 iout=1)	
			G21	G20	G21	G20
X	10	X100;	0.100 mm	0.254 mm	0.0039 inch	0.0100 inch
Z	10	Z100;	0.100 mm	0.254 mm	0.0039 inch	0.0100 inch
X	1	X100;	0.0100 mm	0.0254 mm	0.00039 inch	0.00100 inch
Z	1	Z100;	0.0100 mm	0.0254 mm	0.00039 inch	0.00100 inch

5. 位置指令

5.3 英制/公制指令切换



输出单位・指令单位・设定单位

可选择计数器及参数的设定・显示单位是使用G20/G21指令模式中所决定的指令单位，还是使用参数“#1041 I_inch”所决定的内部单位。在基本规格参数“#1226 aux10/bit6”中设定了公制起始（内部单位为公制）时，选择了以指令单位进行设定显示，则在G21指令模式时，以公制单位显示计数器及参数，在G20指令模式时，将内部单位为公制的数据转换为英制单位进行显示。另外，与内部单位无关，可通过基本规格参数“#1152 I_G20”选择接通电源及重新启动时的指令单位。

NC轴

项目	英制起始关闭（内部单位为公制） #1041 I_inch=0				英制起始打开（内部单位为英制） #1041 I_inch=1			
	#1226 aux10/bit6=0 根据内部单位		#1226 aux10/bit6=1 根据指令单位		#1226 aux10/bit6=0 根据内部单位		#1226 aux10/bit6=1 根据内部单位	
	G21	G20	G21	G20	G21	G20	G21	G20
移动・速度指令	公制	英制	公制	英制	公制	英制	在英制起始时“#1226 aux10/bit6”参数无效。 设定为1时，与设定为0时相同，以内部单位进行设定显示。	
计数器显示	公制	公制	公制	英制	英制	英制		
速度显示	公制	公制	公制	英制	英制	英制		
用户参数 设定・显示	公制	公制	公制	英制	英制	英制		
工件・刀具偏移设 定・显示	公制	公制	公制	英制	英制	英制		
手轮进给指令	公制	公制	公制	英制	英制	英制		
安装参数设定・显示	根据“#1040 M_inch”							

PLC轴

项目	#1042 pcinch=0 （公制）	#1042 pcinch=1 （英制）
移动・速度指令	公制	英制
计数器显示	公制	英制
用户参数 设定・显示	公制	英制
安装参数 设定・显示	根据“#1040 M_inch”	

NC轴、PLC轴均作为旋转轴使用时，即使在英制指令下，坐标数据显示等也显示至小数点后3位。

5. 位置指令

5.3 英制/公制指令切换



使用例

根据参数的设定，画面显示如下。

参数的设定 程序指令・画面操作	#1041 I_inch 0 公制起始 #1152 I_G20 0 重新启动时 G21 (公制) 模式 #1226 设定显示单位为内部单位 aux10/bit6 0	#1041 I_inch 0 公制起始 #1152 I_G20 0 重新启动时 G21 (公制) 模式 #1226 设定单位为指令单位 aux10/bit6 1
	[相对值] X 0.000 Y 0.000 Z 0.000 mm. [刀具补偿] # 1 0.100 11 0.000 2 0.200 12 0.000 3 0.300 13 0.000 mm.	[相对值] X 0.000 Y 0.000 Z 0.000 mm. [刀具补偿] # 1 0.100 11 0.000 2 0.200 12 0.000 3 0.300 13 0.000 mm.
英制指令 G20 G0 X1.Y2.Z3.;	[相对值] X 25.400 Y 50.800 Z 76.200 in. [刀具补偿] # 1 0.100 11 0.000 2 0.200 12 0.000 3 0.300 13 0.000 in. 内部数据 25.4 (mm) 50.8 (mm) 76.2 (mm)	根据指令单位，显示单位也采用英制 [相对值] X 1.0000 Y 2.0000 Z 3.0000 in. [刀具补偿] # 1 0.0039 11 0.0000 2 0.0078 12 0.0000 3 0.0118 13 0.0000 in. 内部数据 #1 0.1 (mm)
#(1) (1) 输入	[刀具补偿] # 1 1.000 11 0.000 ⋮ in. 内部数据 1.0 (mm)	以英制单位设定 [刀具补偿] # 1 1.0000 11 0.0000 ⋮ in. 内部数据 25.4 (mm)

5. 位置指令

5.3 英制/公制指令切换

程序 指令・画面操作	参数的 设定	#1041 I_inch 0 公制起始 #1152 I_G20 0 重新启动时 G21（公制）模式 #1226 设定显示单位为内部单位 aux10/bit6 0	#1041 I_inch 0 公制起始 #1152 I_G20 0 重新启动时 G21（公制）模式 #1226 设定单位为指令单位 aux10/bit6 1
	重新启动	[相对值] X 25.400 Y 50.800 Z 76.200 mm. [刀具补偿] # 1 1.000 11 0.000 2 0.200 12 0.000 3 0.300 13 0.000 mm.	根据指令单位，显示单位恢复公制 [相对值] X 25.400 Y 50.800 Z 76.200 mm. [刀具补偿] # 1 25.400 11 0.000 2 0.200 12 0.000 3 0.300 13 0.000 mm.



注意事项

- (1) 如果指令单位为inch，而内部单位为mm进行刀具及工件偏移数据的设定，可能会导致误差。
- (2) 内部单位是取决于#1041 I_inch的单位。
#1041 I_inch=0 时的内部单位： mm
#1041 I_inch=1 时的内部单位： inch
- (3) 参数的输入输出是以内部单位进行。
- (4) 当刀具数据输入输出单位选择（#1220 aux04/bit4: 1）有效时，以“#1153 I_G20”进行刀具数据的输入输出，除此以外的场合，则以内部单位进行刀具数据的输入输出。

5.4 小数点输入



功能及目的

在定义刀具轨迹、距离及速度的加工程序输入信息中，可以进行指定mm（毫米）或inch（英寸）的零点的小数点输入。

另外，可通过参数“#1078 Decpt2”选择是以无小数点数据的最低位作为最小输入指令单位（类型1），还是作为零位（类型2）。



指令格式

○○○○○. ○○○	公制体系
○○○○. ○○○○	英制体系



详细说明

- (1) 小数点指令对加工程序中的距离、角度、时间及速度指令有效。
- (2) 小数点指令的有效地址，请参阅表“使用地址与小数点指令的有效/无效”。
- (3) 小数点指令中的有效位数如下。（输入指令单位cunit=10时）

	移动指令（直线）		移动指令（旋转）		进给速度		延时（X）	
	整数部分	小数部分	整数部分	小数部分	整数部分	小数部分	整数部分	小数部分
MM （公制）	0～ 99999.	.000～ .999	0～ 99999.	.000～ .999	0～60000.	.000～.999	0.～ 99999.	.000～ .999
					0～999.	.0000～ .9999		
INCH （英制）	0～ 9999.	.0000 ～ .9999	99999. (359.)	.0～.9 99	0～2362.	.0000～ .9999	0～99.	.000～ .999
					0～99.	.000000～ .999999		

- （注）进给速度的上层为每分钟进给，下层为每转进给时。
- (4) 小数点指令对子程序等中使用的变量数据也有效。
 - (5) 对于小数点无效地址的小数点指令，仅将跳跃了小数点以下部分后的整数部分作为数据加以处理。小数点无效地址中，包括如下地址。（D、H、L、M、N、O、P、S、T）
- 但是，在变量指令中，全部作为带小数点数据加以使用。



注意事项

- (1) 当含有四则运算符时，作为带小数点数据加以使用。
- （例1）G00 X123+0;
是X轴123mm的指令。不会成为123 μ m。



程序例

(1) 小数点有效地址的程序例

按规格区分 程序例	小数点指令 1		小数点指令 2 1 = 1 μm 时
	1 = 1 μm 时	1 = 10 μm 时	
G0 X123.45 (小数点全部为 mm 点)	X123.450 mm	X123.450 mm	X123.450 mm
G0 X12345	X12.345 mm (最后位以 1 μm 为单位)	X123.450 mm	X12345.000 mm
#111 = 123 #112 = 5.55 X#111 Z#112	X123.000 mm Z5.550 mm	X123.000 mm Z5.550 mm	X123.000 mm Z5.550 mm
#113 = #111 + #112 (加运算)	#113 = 128.550	#113 = 128.550	#113 = 128.550
#114 = #111 - #112 (减运算)	#114 = 117.450	#114 = 117.450	#114 = 117.450
#115 = #111 * #112 (乘运算)	#115 = 682.650	#115 = 682.650	#115 = 682.650
#116 = #111 / #112 #117 = #112 / #111 (除运算)	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045



小数点输入 I、II 与小数点指令有效、无效

在下一项的一览表中，当在小数点指令有效的地址发出了未使用小数点的指令时，小数点输入 I、II 如下。另外，发出了使用小数点的指令时，小数点输入 I、II 相同。

(1) 小数点输入 I

指令数据的最低位与指令单位一致。

(例) 在 1 μm 系统中指定了“X1”时，与指定“X0.001”时相同。

(2) 小数点输入 II

指令数据的最低位与指令位置一致。

(例) 在 1 μm 系统中指定了“X1”时，与指定“X1.”时相同。

5. 位置指令

5.4 小数点输入

- 使用地址与小数点指令的有效/无效 -

地址	小数点指令	用途	备注
A	有效	坐标位置数据	
	无效	第2辅助功能代码	
	有效	角度数据	
	无效	MRC 程序编号	
	无效	程序参数输入 轴编号	
	有效	深孔钻孔循环 (2) 安全距离	
	有效	主轴同期 加减速时间常数	
B	有效	坐标位置数据	
	无效	第2辅助功能代码	
C	有效	坐标位置数据	
	无效	第2辅助功能代码	
	有效	转角倒角量	, C
	有效	程序刀具补偿输入 刀尖 R 补偿量 (增量)	
	有效	倒角宽度 (开槽循环)	
D	有效	自动刀具长度测定 减速区域 d	
	无效	程序参数输入 Byte 型数据	
	无效	主轴同期 同步主轴编号	
E	有效	英制螺纹圈数, 精密螺纹导程	
	有效	转角的切削进给速度	
F	有效	进给速度	
	有效	螺纹导程	
G	有效	准备功能代码	
H	有效	坐标位置数据	
	无效	子程序内的顺序编号	
	无效	程序参数输入 Bit 型数据	
	无效	直线-圆弧的交点选择 (几何加工)	
	无效	主轴同期 基准主轴编号	
I	有效	圆弧中心的坐标	
	有效	刀尖 R 补偿/ 刀具半径补偿的矢量成分	
	有效	深孔钻孔循环 (2) 第1次切入量	
	有效	G0/G1 就位宽度 钻孔循环 G0 就位宽度	, I

J	有效	圆弧中心的坐标	
	有效	刀尖 R 补偿/ 刀具半径补偿的矢量成分	
	无效	深孔钻孔循环 (2) 在返回点的延时	
	有效	钻孔循环 G1 就位宽度	, J
K	有效	圆弧中心的坐标	
	有效	刀尖 R 补偿/ 刀具直径补偿的矢量成分	
	无效	钻孔循环 重复次数	
	有效	深孔钻孔循环 (2) 第2循环以后的切入量	
	有效	螺纹导程增减量 (可变导程螺纹切削)	
L	无效	子程序重复次数	
	无效	程序刀具补偿输入 种类选择	L2 L10 L11
	无效	可编程电流限制值	L14
	无效	程序参数输入 选择	L50
	无效	程序参数输入 双字型数据	4Byte
M	无效	辅助功能代码	
N	无效	编码器编号	
	无效	程序参数输入 数据编号	
O	无效	程序编号	
P	无效	停止时间	
	无效	子程序调用 程序编号	
	无效	第2、3、4 参考点 编号	
	无效	线速度恒定控制 轴编号	
	无效	MRC 精整形状开始 编码器编号	
	有效	平削循环 移位量/切入量	
	无效	复合型螺纹切削循环 切入次数、端面倒角、 刀尖角度	

(注1) 对于用户宏自变量, 小数点全部有效。

5. 位置指令

5.4 小数点输入

地址	小数点指令	用途	备注
P	有效	复合型螺纹切削循环 螺纹高度	
	无效	程序刀具补偿输入 补偿编号	
	无效	程序参数输入 大区分编号	
	无效	来自子程序的 返回顺序编号	
	有效	坐标位置数据	
	无效	跳跃信号指令	
	有效	圆弧中心坐标（绝对值） （几何加工）	
	无效	扩展工件坐标系偏移	
	无效	主轴最低钳位转速	
	无效	MRC 精整形状结束 编码器编号	
Q	有效	平削循环 切入量/移位置	
	有效	复合型螺纹切削循环 最小切入量	
	有效	复合型螺纹切削循环 第 1 次切入量	
	有效	深孔钻孔循环 1 每次的切入量	
	无效	程序刀具补偿输入 虚拟刀尖点编号	
	无效	深孔钻孔循环（2） 在切入点上的延时	
	有效	圆弧中心坐标（绝对值） （几何加工）	
	有效	螺纹切削开始移位角度	
	有效	R 指定圆弧半径	
	有效	转角 R 圆弧半径	, R
R	有效	自动刀具长度测定 减速区域 r	
	有效	MRC 横向・端面 余量	
	无效	MRC 成型 分割次数	
	有效	平削循环 返回量	
	有效	平削循环 余量	
	有效	复合型螺纹切削循环 精整余量	
	有效	同期主轴 位相移位置	

R	有效	复合型螺纹切削循环・ 车削循环 锥轴差	
	有效	钻孔循环・深孔钻孔 循环（2）到达 R 点前的距离	
	有效	程序刀具补偿输入 刀尖 R 补偿量（绝对）	
	有效	坐标位置数据	
	有效	粗切削循环（横向） （端面）余量	
	无效	同期攻牙/非同期攻牙切换	, R
S	无效	主轴功能代码	
	无效	主轴最高钳位转速	
	无效	线速度恒定控制 线速度	
	无效	程序参数输入 字型数据	2Byte
T	无效	刀具功能代码	
U	有效	坐标位置数据	
	有效	程序刀具补偿输入	
	有效	粗切削循环（横向）切入量	
	有效	停止	
V	有效	坐标位置数据	
	有效	程序刀具补偿输入	
W	有效	坐标位置数据	
	有效	程序刀具补偿输入	
	有效	粗切削循环（端面）切入量	
X	有效	坐标位置数据	
	有效	停止	
	有效	程序刀具补偿输入	
Y	有效	坐标位置数据	
	有效	程序刀具补偿输入	
Z	有效	坐标位置数据	
	有效	程序刀具补偿输入	

6. 插补功能

6.1 定位（快速进给）；G00



功能及目的

本指令是通过坐标字，以当前点作为起点，通过直线或非直线轨迹定位到坐标字所指定的终点。



指令格式

G00 Xx/Uu Zz/Ww;

x、u、z、w : 表示坐标值。

指令地址对所有附加的轴有效。



详细说明

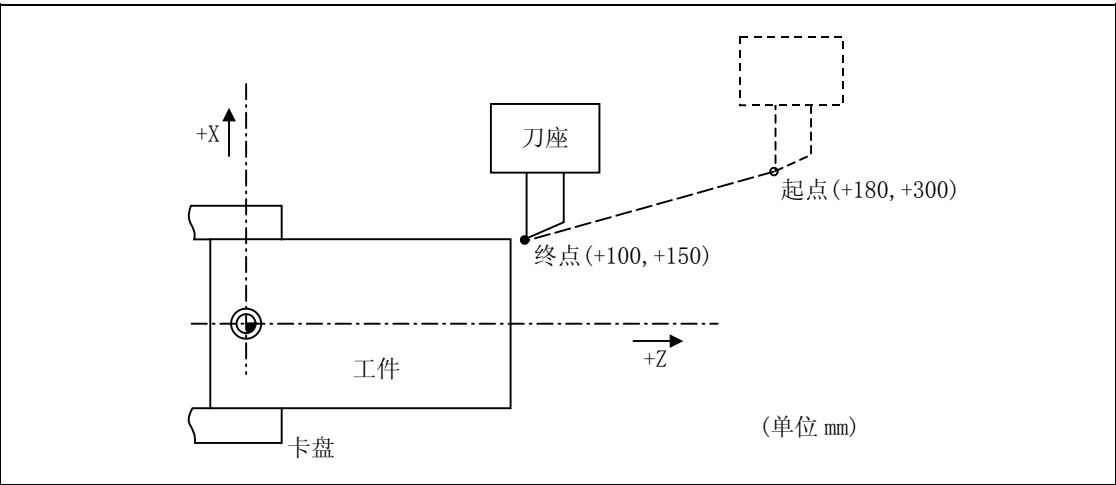
- (1) 执行一次本指令，则发出变更本G00模式的其他G功能，即发出01组的G01、G02、G03、G33、G34之前，保持G00模式。借此，如果下一指令仍然是G00，则只指定坐标字即可。
- (2) 在G00模式下，总是在单节的起点、终点进行加速、减速，当前单节的指令为0且确认加减速电路的追踪误差状态之后，进入下一单节。通过指令单节的地址（I）或参数设定就位宽度。
- (3) 根据G00指令，09组的G功能（G83~G89）为取消（G80）模式。
- (4) 可通过参数选择刀具的轨迹为直线或非直线，但是定位的时间不会发生改变。
 - (a) 直线轨迹……与直线插补（G01）相同，速度受到各轴的快速进给速度限制。
 - (b) 非直线轨迹……以各轴独立的快速进给速度进行定位。
- (5) 当G后无数值时，作为G00加以处理。

⚠ 注意

⚠ “G后无数值”的指令，被看作为“G00”。



程序例

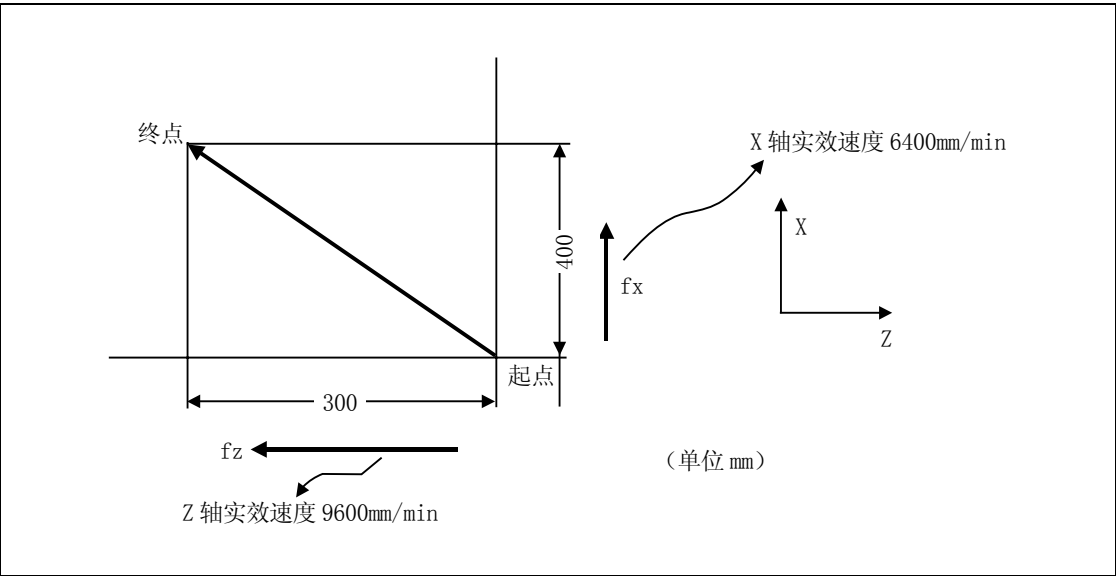


G00 X100000 Z150000 ;	绝对值指令
G00 U-80000 W-150000 ;	增量值指令（但是，输入设定单位 0.001mm 时）



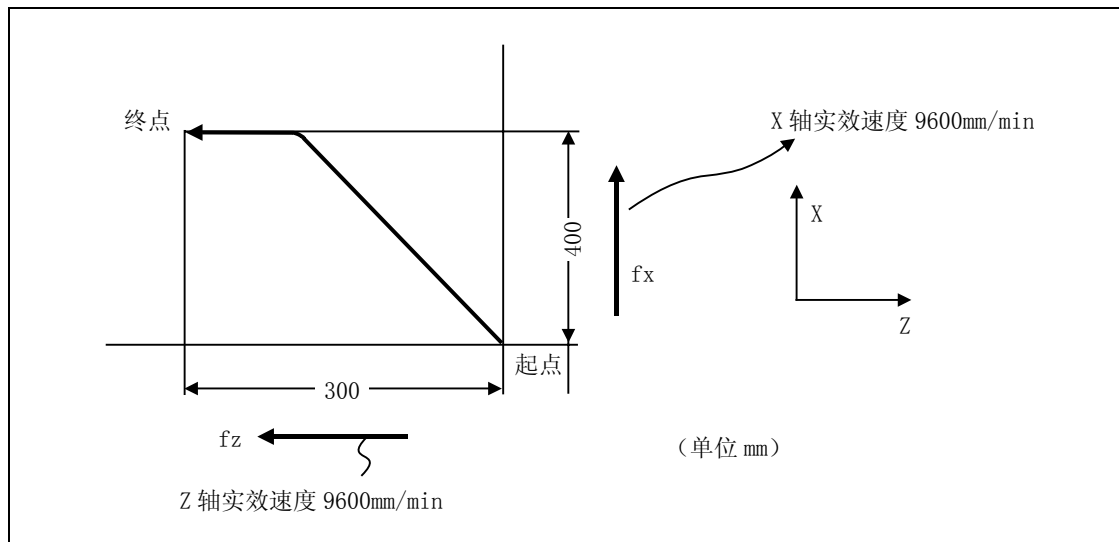
注意事项

（注 1）参数“#1086 G0Intp”为“0”时，定位时的刀具移动轨迹为连接起点与终点的最短路径。在指定的各轴速度不超过快速进给速度的前提下，自动计算定位速度，以确保分配时间最短。
例如，当 X 轴与 Z 轴的快速进给速度均为 9600 mm/min 时，在编程时设定
G00 Z-300000 X400000; （输入设定单位为 0.001mm 时）
则刀具按照如下图所示的路径进行移动。



（2）当参数“#1086 G0Intp”为“1”时，在定位过程中，刀具以快速进给速度从移动路径的起点移动到终点。

例如，当 X 轴与 Z 轴的快速进给速度均为 9600mm/min 时，在编程时设定
G00 Z-300000 X400000 ；（输入设定单位为 0.001mm 时）
则刀具按照如下图所示的轨迹移动。



（注 3）由 G00 所决定的各轴快速进给速度因各机械而异，所以，请参阅机械规格书。

（注 4）快速进给（G00）减速检查

快速进给时的减速检查方式中，包括指令减速方式与就位检查方式 2 种方式，可通过参数“#1193 inpos”加以选择。

■ “inpos”=1 时

快速进给（G00）处理完成后，确认各轴的剩余距离在一定值以下，然后执行下一单节。（参阅后述的“就位检查的动作”）

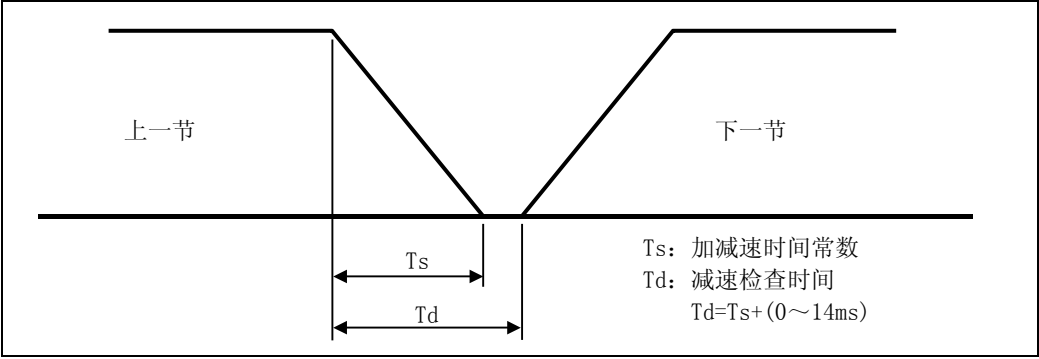
根据快速进给就位宽度 L_R 进行剩余距离的确认。 L_R 为伺服参数“#1224 sv024”或“#2077 G0inps”的设定值。

快速进给减速检查的目的，是缩短定位时间。如果增大就位宽度的设定值，则缩短时间增大，开始下一单节时，上一单节的剩余距离也会增大，在实际加工中，可能会导致发生故障。

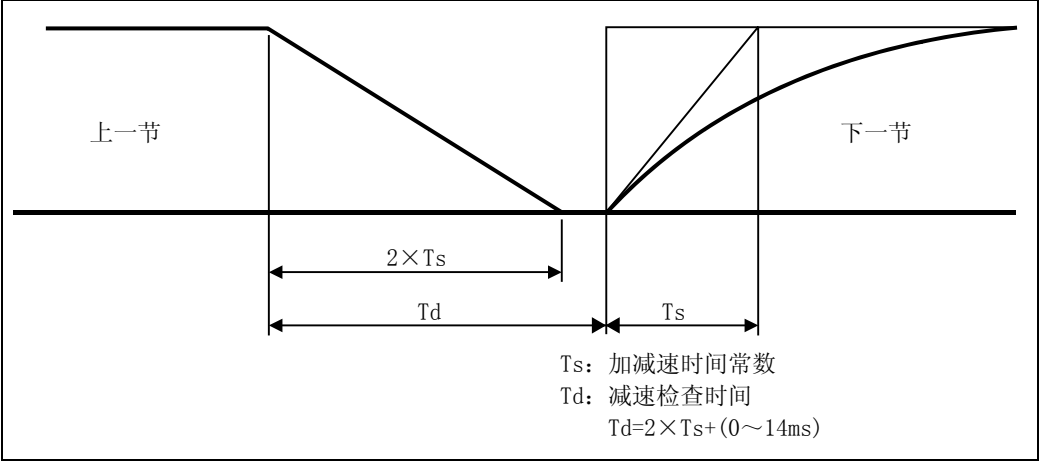
每隔一段时间进行一次剩余距离的检查。因此，可能会无法达到与就位宽度设定值所对应的定位时间缩短效果。

■ “inpos” =0 时
快速进给（G00）的处理完成后，经过减速检查时间（Td）之后，执行下一单节。
根据加减速类型，减速检查时间（Td）如下。

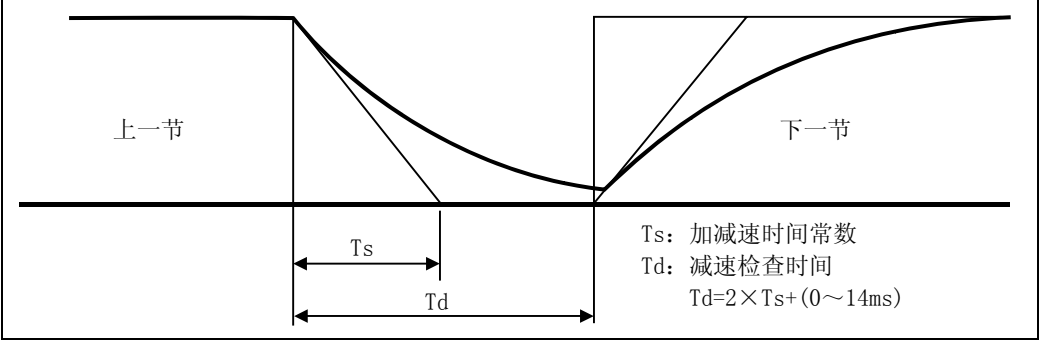
(1) 直线加速/直线减速…… $T_d = T_s + \alpha$



(2) 直线加速/直线减速…… $T_d = 2 \times T_s + \alpha$



(3) 直线加速/指数减速…… $T_d = 2 \times T_s + \alpha$



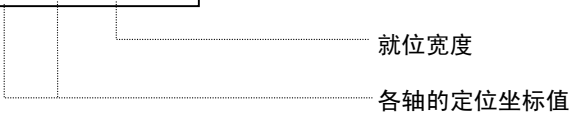
但是，Ts 为加速时间常数， $\alpha = 0 \sim 14\text{ms}$
快速进给时的减速检查所需时间，取决于同时发出指令轴的快速进给加减速模式及快速进给加减速时间常数所决定的各轴快速进给减速检查时间中，最长的一个。



定位指令时的就位宽度编程 [E68]

本指令是通过加工程序指定定位指令时的就位宽度。

```
G00 X_ Z_ ,I_ ;
```



,I_ : 就位宽度。发出小数点指令则发生程序错误。
仅指定的单节有效。因此，对于没有该地址的单节，则取决于参数“#1193 inpos”的设定。



就位检查的动作 [E68]

确认定位（快速进给：G00）指令单节及因直线插补（G01）指令而进行减速检查的单节的位置误差量低于本指令的就位宽度，然后开始执行下一单节。
由于本指令的就位宽度仅对指令单节有效，所以，没有就位宽度指令的单节采用基本规格参数“#1193 inpos”所决定的减速检查方式。
当有多根移动轴时，请在确认所有移动轴的位置误差量小于本指令的就位宽度之后，开始执行下一单节。
当通过参数让就位检查生效时（基本规格参数“#1193 inpos”设定为1，关于就位宽度，请参阅下页），与本指令的不同点如下图所示。

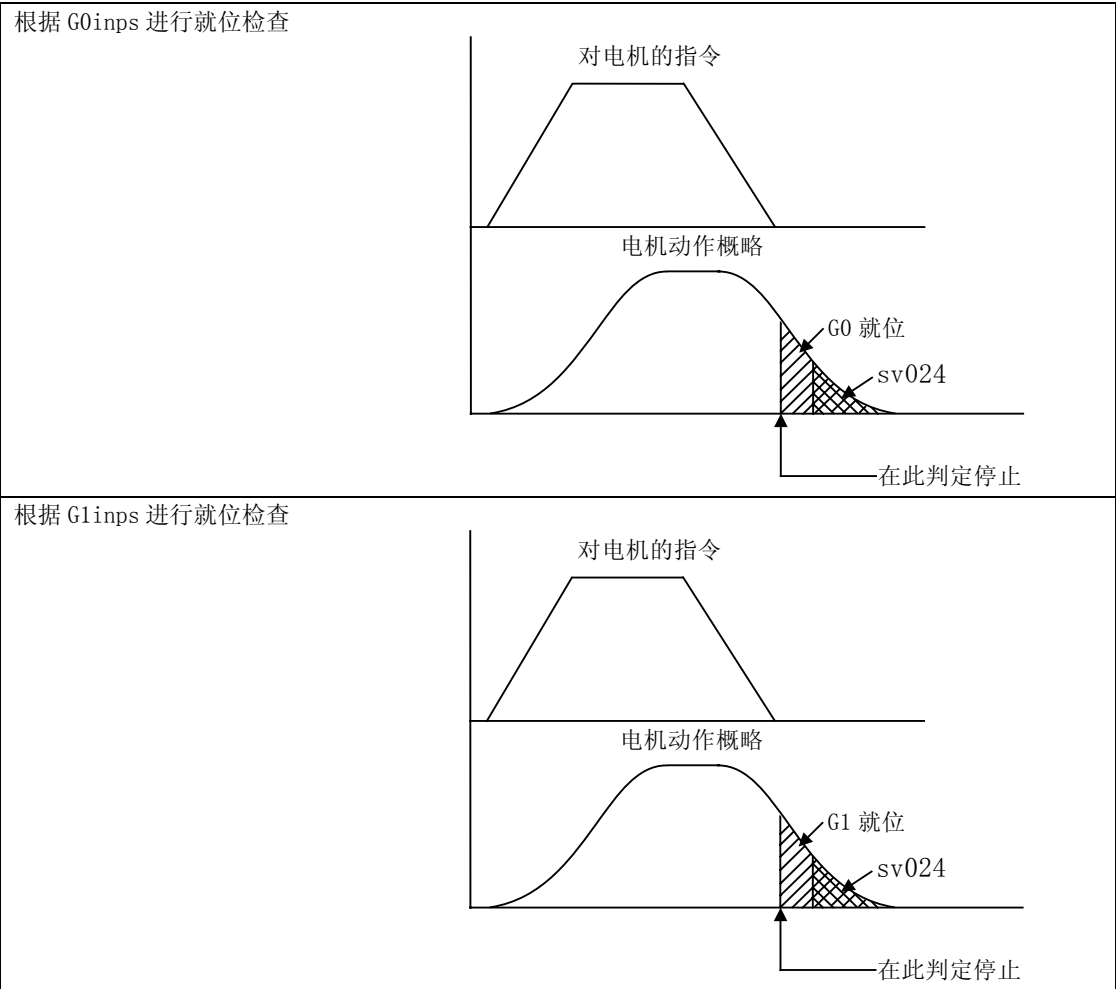
与通过参数进行就位检查的差异

通过“;I”地址指令进行就位检查	通过参数进行就位检查
指令系统开始减速之后，比较位置误差量与指定的就位宽度。	指令系统减速完成后，比较伺服系统的位置误差量与参数设定值（就位宽度）。
Ts: 加减速时间常数 Td: 减速检查时间 Td = Ts + (0~14ms)	



就位宽度设定

伺服参数“#2224 SV024”的设定值小于G0就位宽度“#2077 G0inps”或G1就位宽度“#2078 G1inps”的设定值时，根据G0就位宽度、G1就位宽度进行就位检查。



当sv024较大时，输入sv024，则就位检查完成。
就位检查方式取决于减速检查的参数的方式。

（注 1）对于 E68，当在加工程序中指定了就位宽度（可编程就位检查宽度）时，以通过参数设定的就位宽度（SV024、G0inps、G1inps）与程序中指定的就位宽度中较大的一个作为就位宽度，进行就位检查。

（注 2）SV024 的设定值比 G0 就位宽度/G1 就位宽度大时，根据 SV024 进行就位检查。

6.2 直线插补；G01



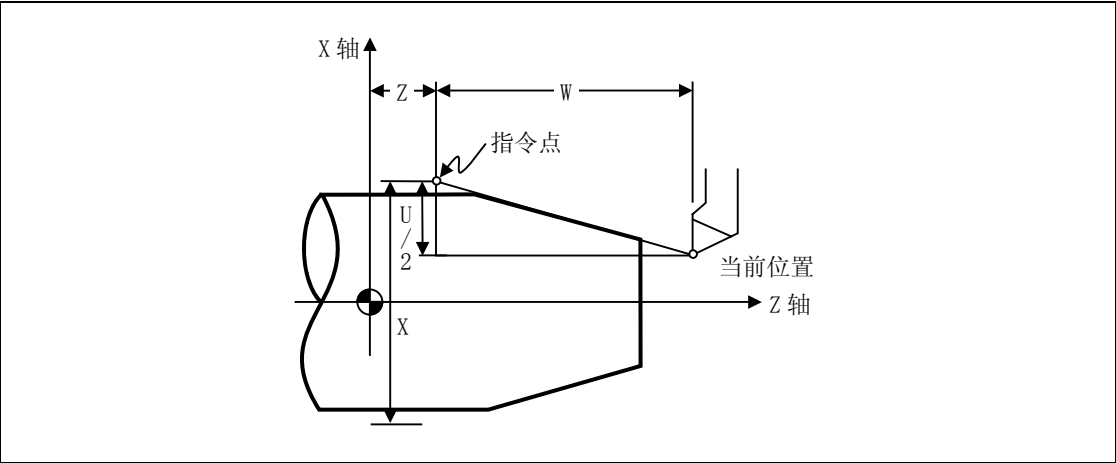
功能及目的

该指令是通过坐标语与进给速度指令的组合，以地址F中所指定的速度，将刀具从当前点直线移动（插补）到坐标语所指定的终点。但是此时，地址F所指定的进给速度，总是作为相对于刀具中心的进给方向的线速度而发挥作用。



指令格式

G00 Xx/Uu Zz/Ww α α Ff ; （α 为附加轴）	
x、u、z、w、α	: 表示坐标值。
f	: 进给速度。（mm/min 或 ° /min）



详细说明

执行过一次本指令，则在指定变更该G01模式的其他G功能，即01组的G00、G02、G03、G33、G34之前，保持该模式。借此，如果下一指令仍然是G01，且进给速度没有变化时，仅指定坐标语即可。当最初的G01指令中没有F指令时，发生程序错误（P62）。

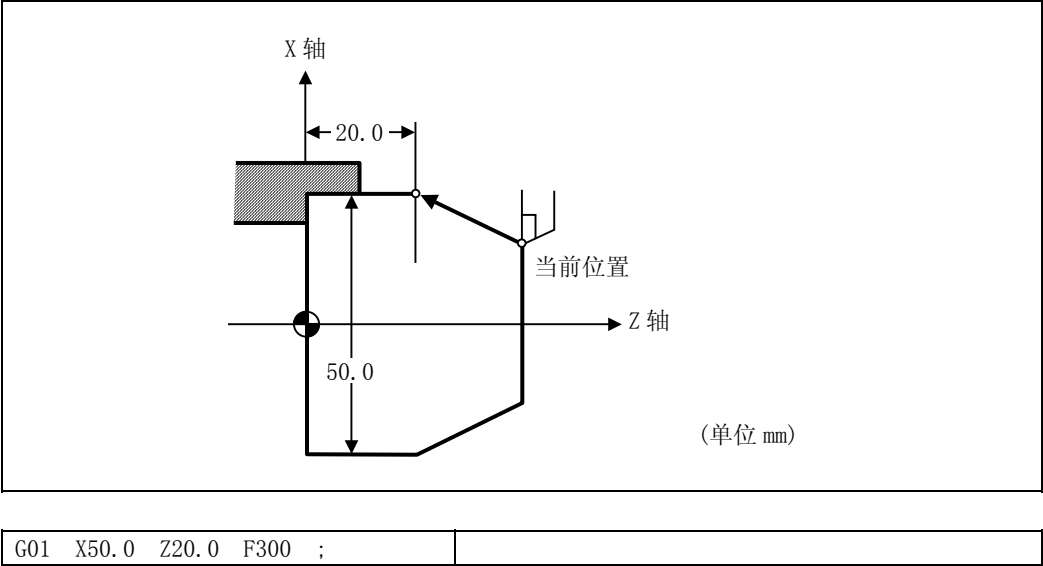
以° /min（小数点位置的单位）指定旋转轴的进给速度。（F300=300° /min）

根据G01指令，09组的G功能（G70~G89）被取消（G80）。

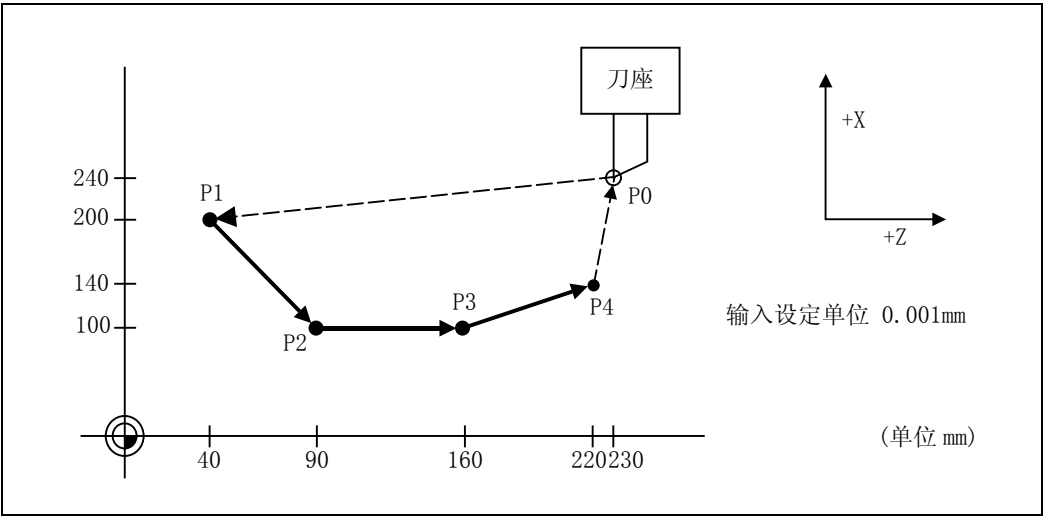


程序例

(例 1)



(例 2) 以进给速度 300mm/min切削 $P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4$
但是, $P_0 \rightarrow P_1$ 、 $P_4 \rightarrow P_0$ 为刀具定位



G00 X200000 Z40000 ;	$P_0 \rightarrow P_1$
G01 X100000 Z90000 F300;	$P_1 \rightarrow P_2$
Z160000;	$P_2 \rightarrow P_3$
X140000 Z220000;	$P_3 \rightarrow P_4$
G00 X240000 Z230000;	$P_4 \rightarrow P_0$

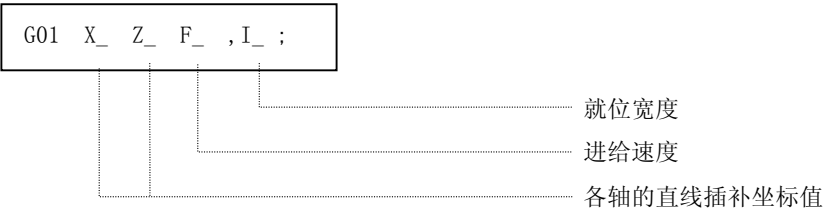


直线插补指令时的就位宽度编程指令

[E68]

本指令是通过加工程序指定直线插补指令时的就位宽度。
在直线插补指令中，仅当进行减速检查时，所指定的就位宽度有效。

- 当错误检测开关ON时。
- 同一单节中，发出了G09（准确定位检查）指令时。
- 选择了G61（准确定位检查模式）时。



,I_ : 就位宽度。发出小数点指令则发生程序错误。
仅指定的节有效。对于没有该地址的节，则取决于参数“#1193 inpos”的设定。

（注 1）就位检查的动作，请参阅“6.1 定位（快速进给）；G00”。

6.3 圆弧插补；G02、G03



功能及目的

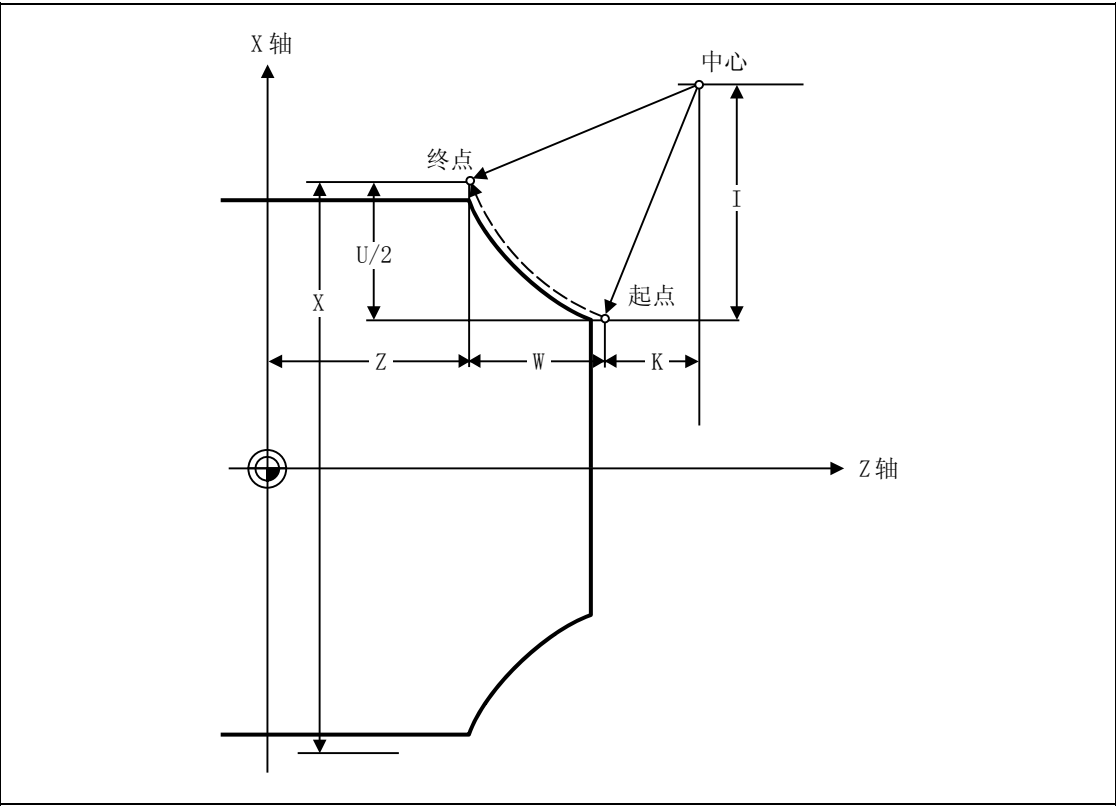
该指令是让刀具沿着圆弧移动。



指令格式

G02 (G03) Xx/Uu Zz/Ww Ii Kk Ff ;	
G02	: 顺时针旋转 (CW)
G03	: 逆时针旋转 (CCW)
Xx/Uu	: 圆弧终点坐标、X 轴 (X 为工件坐标系的绝对值, U 为距当前位置的增量值)
Zz/Ww	: 圆弧终点坐标、Z 轴 (Z 为工件坐标系的绝对值, W 为距当前位置的增量值)
Ii	: 圆弧中心、X 轴 (I 为从起点看的中心 X 坐标的半径指令增量值)
Kk	: 圆弧中心、Z 轴 (K 为从起点看的中心 Z 坐标的半径指令增量值)
Ff	: 进给速度

以输入设定单位指定圆弧中心坐标值。当对输入指令单位不同的轴发出圆弧指令时，必须加以注意。为了防止混乱，发出指令时请带上小数点。





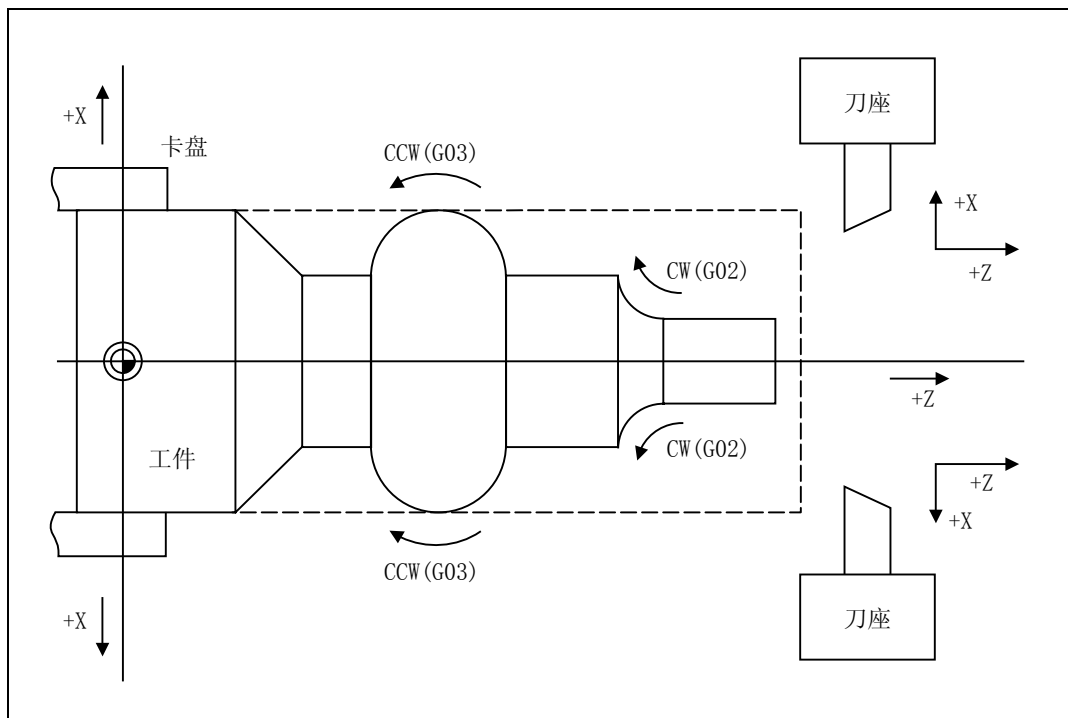
详细说明

(1) 在发出变更G02 (G03) 模式的其他G指令，即01组的G00或G01或G33之前，保持G02 (G03) 的状态。

以G02、G03区分圆弧的旋转方向。

G02 CW (顺时针旋转)

G03 CCW (逆时针旋转)



(2) 可在单个单节指令中执行跨越多个象限的圆弧。

(3) 为了进行圆弧插补，需要以下的信息。

(a) 旋转方向……………顺时针旋转 (G02) 还是逆时针旋转 (G03)。

(b) 圆弧终点坐标……………通过地址X、Z、U、W赋值。

(c) 圆弧中心坐标……………通过地址I、K赋值。(增量值指令)

(d) 进给速度……………通过地址F赋值。

(4) 如果I、K或R未指定，则发生程序错误。

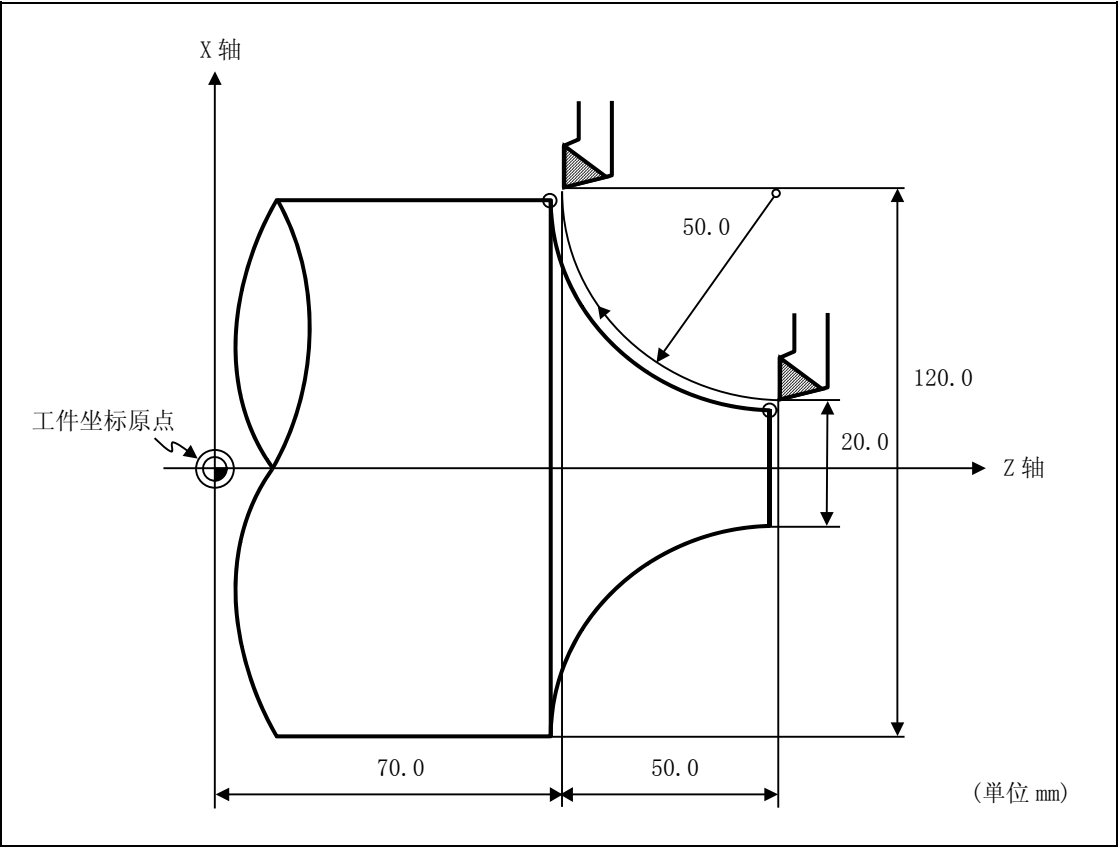
I、K是从起点看的距圆弧中心的X轴、Z轴方向上的距离，所以请考虑符号。

(5) 无法执行G2/G3模式中的T指令。

在G2/G3的模式下发出T指令，则发生程序错误 (P151)。



程序例

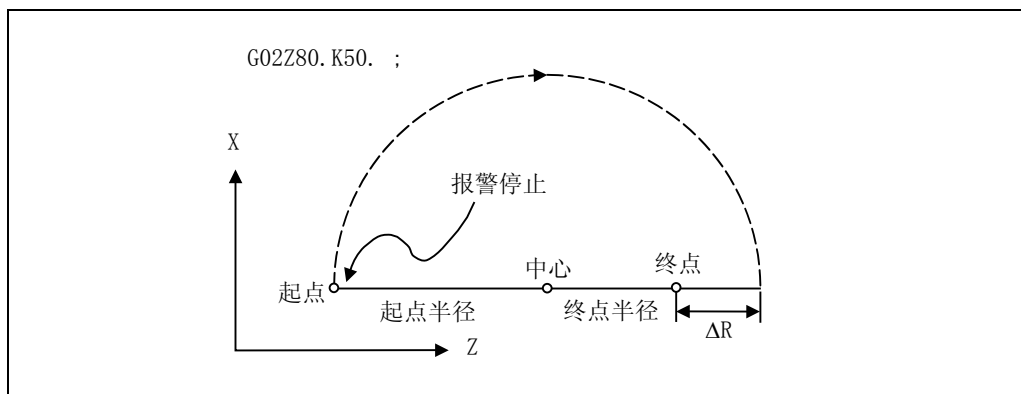


G2 X120.0 Z70.0 I50.0 F200;	绝对值指令
G2 U100.0 W-50.0 I50.0 F200;	增量值指令

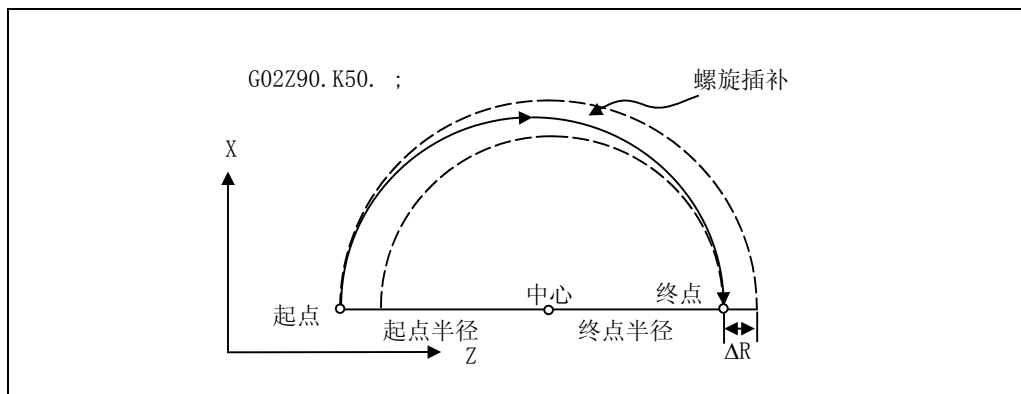


圆弧插补相关注意事项

- (1) 在圆弧动作中，顺时针旋转（G02）、逆时针旋转（G03）是指“在右手坐标系中，从与对象平面垂直相交的坐标轴的正方向向负方向看时”。
- (2) 如果将终点坐标全部省略，或是终点与起点位于同一位置，则使用I、K指定中心时，变为指定360°的圆弧（正圆）。
- (3) 在圆弧指令中，如果起点半径与终点半径不一致，则如下。
 - (a) 当误差A比参数“#1084 RadErr”大时，在圆弧起点发生程序错误（P70）。



- (b) 当误差 ΔR 小于参数值时，变为朝向指令终点的螺旋状插补。



6.4 R指定圆弧插补；G02、G03



功能及目的

除以往的通过指定圆弧中心坐标（I、K）进行圆弧插补指令外，还可以通过直接指定圆弧半径R发出圆弧插补指令。



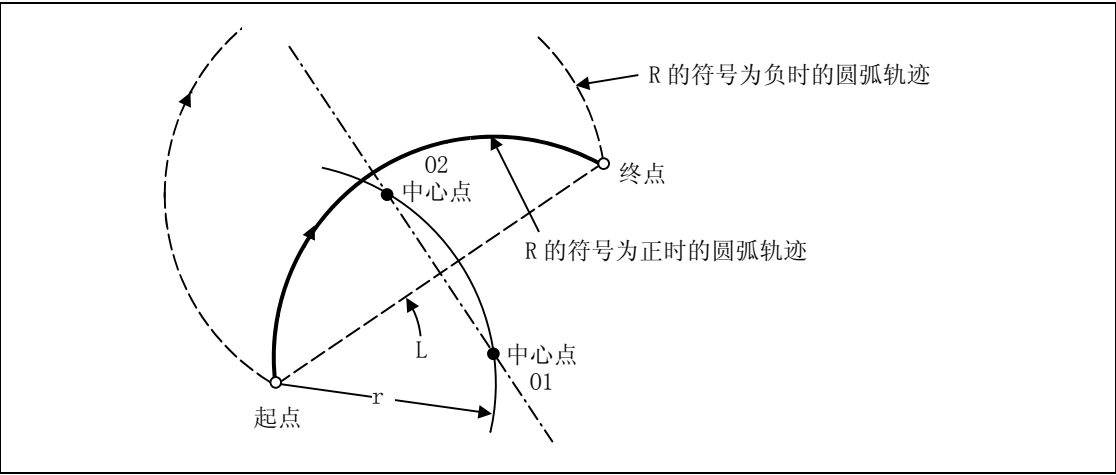
指令格式

G02（G03）	Xx/Uu Zz/Ww Rr Ff ;
x/u	: X 轴终点坐标
z/w	: Z 轴终点坐标
r	: 圆弧半径
f	: 进给速度



详细说明

圆弧中心位于与连接起点与终点的线段直角相交的2等分线上，与以开始点为中心的指定半径的圆的交点，就是所指定的圆弧指令的中心坐标。
当指令程序的R的符号为正时，表示半圆以下的圆弧指令，当指令程序的R的符号为负时，变为半圆以上的圆弧指令。



R指定圆弧插补指令需要满足如下的条件。

$$\frac{L}{2 \cdot r} \leq 1$$

$L/2 - r > \text{参数值} (\#1084 \text{ RadErr})$ 时，报警。

在这里，L为从起点到终点的直线线段。
在同一单节中，同时指定了R指令与I、K指令时，通过指定R发出的圆弧指令优先。
对于正圆指令（起点与终点一致），由于R指定圆弧指令会立即完成，不会进行任何动作，所以请使用I、K指定圆弧指令。



程序例

(例 1)

G03 Zz1 Xx1 Rr1 Ff1 ;	ZX 平面 R 指定圆弧
-----------------------	--------------

(例 2)

G02 Xx1 Zz1 Ii1 Kk1 Rr1 Ff1 ;	XZ 平面 R 指定圆弧 (当一单节中同时有通过 R 进行的指定和通过 I、K 进行的指定时，通过 R 进行的指定优先处理。)
-------------------------------	--

6.5 平面选择：G17、G18、G19



功能及目的

是选择控制平面或圆弧所在平面的指令。
可通过将3根基本轴及与之对应的平行轴作为参数加以注册，选择由任意2根非平行轴的轴所确定的平面。
如果将旋转轴注册为平行轴，则也可选择包含旋转轴在内的平面。
平面选择用于

- 进行圆弧插补的平面
- 进行刀尖R补偿的平面

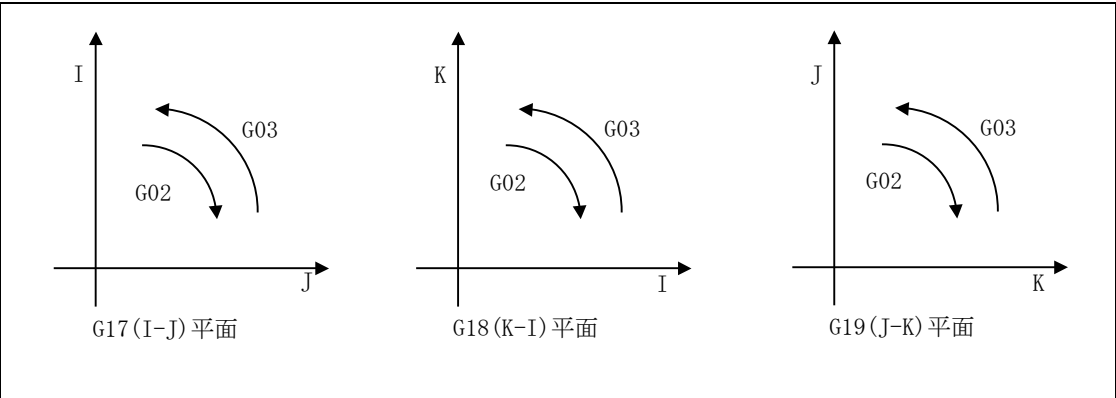
的选择。



指令格式

G17;	(IJ 平面的选择)
G18;	(KI 平面的选择)
G19;	(JK 平面的选择)

I、J、K表示各基本轴或其平行轴。
接通电源后及重新启动时，“#1025 I_plane”中所设定的平面被选中。





参数注册

	# 1026~1028 base_ I, J, K	#1029~1031 aux_ I, J, K
I	X	Y
J	Y	
K	Z	

在参数中，可进行基本轴与平行轴的注册。同一轴名称虽然可重复注册，但是如果重复发出指令，则根据平面选择方式（4）决定平面。
没有注册为控制轴的轴无法设定。

图 1 平面选择参数注册例

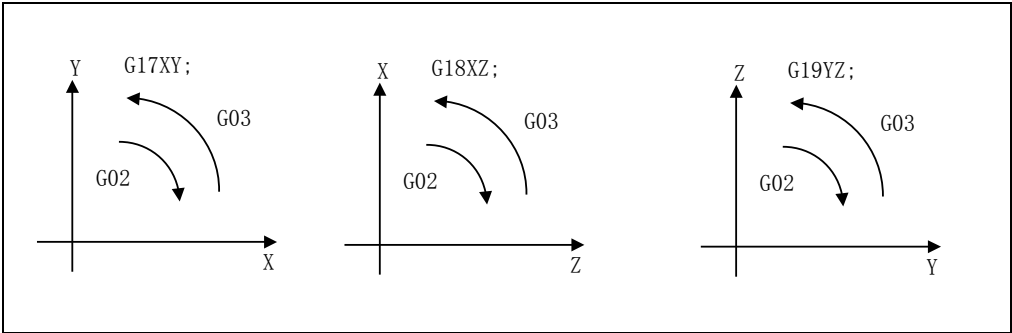


平面选择方式

对图1的参数注册制中的平面选择加以说明。

（1）通过平面选择（G17、G18、G19）与同一单节中指定的轴地址，决定是通过基本轴还是通过基本轴的平行轴选择平面。

（例）



（2）未指定平面选择G代码（G17、G18、G19）指定的单节中，无法进行平面切换。

G18 X_ Z_ ; ZX平面
Y_ Z_ ; ZX平面（不进行平面变化）

（3）当在指定了平面选择G代码（G17、G18、G19）的单节中省略了轴地址时，则以3根基本轴的轴地址作为轴地址指令。

G18 ; （ZX平面 = G18 XZ ; ）

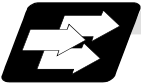
（4）如果与平面选择G代码（G17、G18、G19）在同一单节中重复指定了基本轴或其平行轴，则按照基本轴、平行轴的优先顺序决定平面。

G18 XYZ ; ZX平面被选中。因此，Y变为与选择平面无关。

（注 1）如果在参数“#1025 I_plane”中设定“2”，则接通电源时及重新启动时，G18 平面被选中。

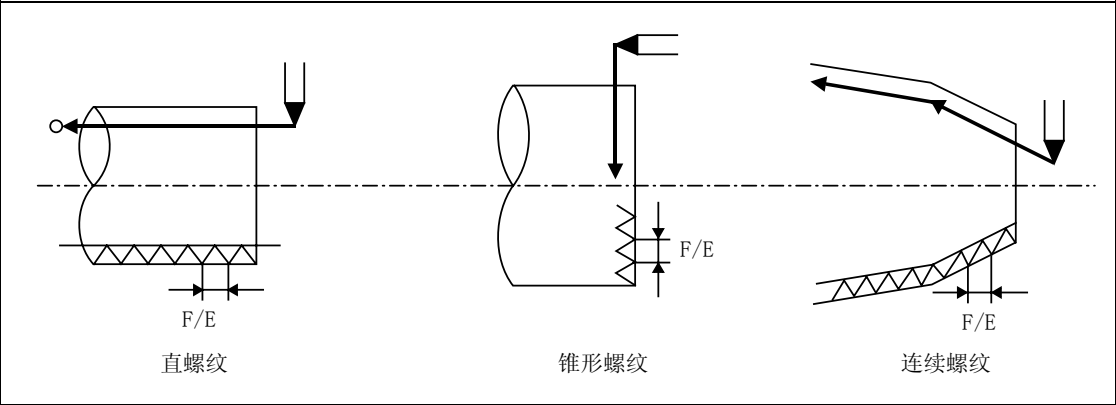
6.6 螺纹切削

6.6.1 固定导程螺纹切削：G33



功能及目的

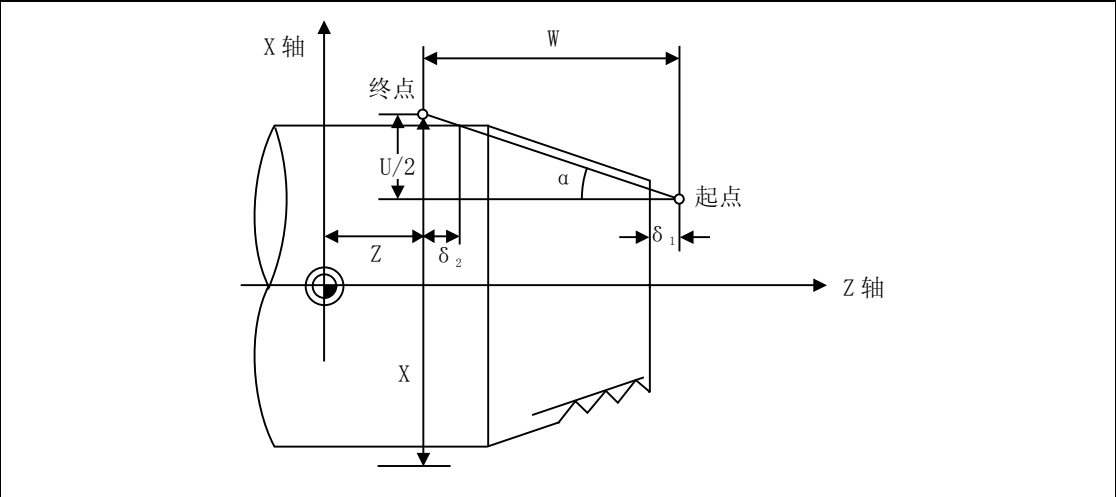
由于是通过G33指令进行与主轴旋转同步的刀具进给控制，所以可进行固定导程的直螺纹切削加工、锥形螺纹切削加工及连续螺纹切削加工。



指令格式

G33	Zz/Ww Xx/Uu Ff Qq ;	(普通导程螺纹切削指令)
Zz, Ww, Xx, Uu	:	螺纹的终点地址及坐标值
Ff	:	长轴（移动量最多的轴）方向导程
Qq	:	螺纹切削开始移位角度（0.001~360.000°）

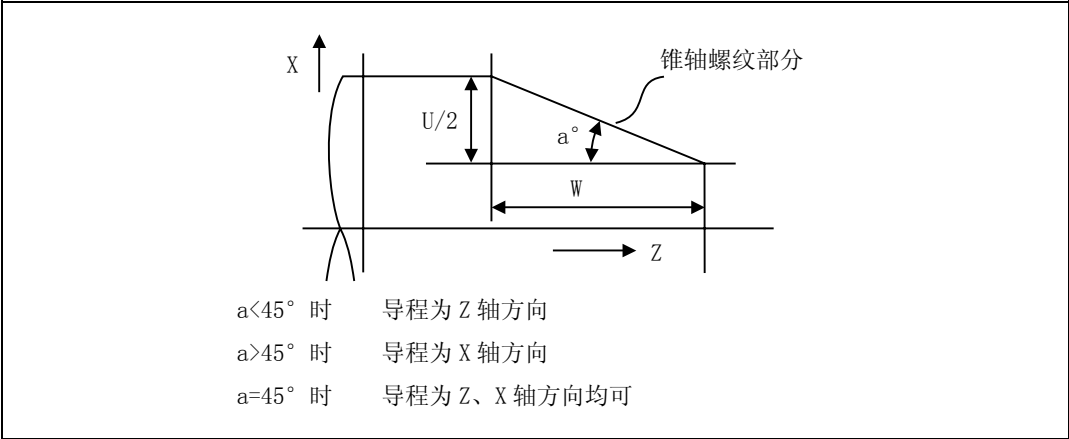
G33	Zz/Ww Xx/Uu Fe Qq ;	(精密导程螺纹切削指令)
Zz, Ww, Xx, Uu	:	螺纹的终点地址及坐标值
Ee	:	长轴（移动量最多的轴）方向导程
Qq	:	螺纹切削开始移位角度（0.001~360.000°）





详细说明

- (1) E指令也可用于英制螺纹切削的螺纹圈数，可通过参数设定选择是通过螺纹圈数进行指定，还是通过精密导程进行指定。（参数“#1229 set01/bit1”设定为“1”，则为通过精密导程进行指定。）
- (2) 指定长轴方向的导程作为锥形螺纹的导程



(a) E60

公制输入

英制输入

输入单位体系	0.001mm			0.0001inch		
指令地址	F(mm/rev)	E(mm/rev)	E(山/inch)	F(inch/rev)	E(inch/rev)	E(山/inch)
最小指令单位	1(=0.0001) (1.=1.0000)	1(=0.0001) (1.=1.00000)	1(=1) (1.=1.00)	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)	1(=1) (1.=1.0000)
指令范围	0.0001 ~999.9999	0.00001 ~999.99999	0.03 ~999.99	0.000001 ~99.999999	0.000010 ~9.9999999	0.0101 ~9999.9999

(b) E68

公制输入

输入单位体系	B(0.001mm)			C(0.0001mm)		
指令地址	F(mm/rev)	E(mm/rev)	E(圈/inch)	F(mm/rev)	E(mm/rev)	E(圈/inch)
最小指令单位	1(=0.0001) (1.=1.0000)	1(=0.0001) (1.=1.00000)	1(=1) (1.=1.00)	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1) (1.=1.000)
指令范围	0.0001 ~999.9999	0.00001 ~999.99999	0.03 ~999.99	0.00001 ~99.99999	0.000001 ~99.999999	0.255 ~9999.999

英寸输入

输入单位体系	B(0.0001inch)			C(0.00001inch)		
指令地址	F(inch/rev)	E(inch/rev)	E(圈/inch)	F(inch/rev)	E(inch/rev)	E(圈/inch)
最小指令单位	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)	1(=1) (1.=1.0000)	1(=0.0000001) (1.=1.0000000)	1(=0.00000001) (1.=1.00000000)	1(=1) (1.=1.00000)
指令范围	0.000001 ~99.999999	0.000010 ~9.9999999	0.0101 ~9999.9999	0.0000001 ~9.9999999	0.00000001 ~0.99999999	0.10001 ~999.99999

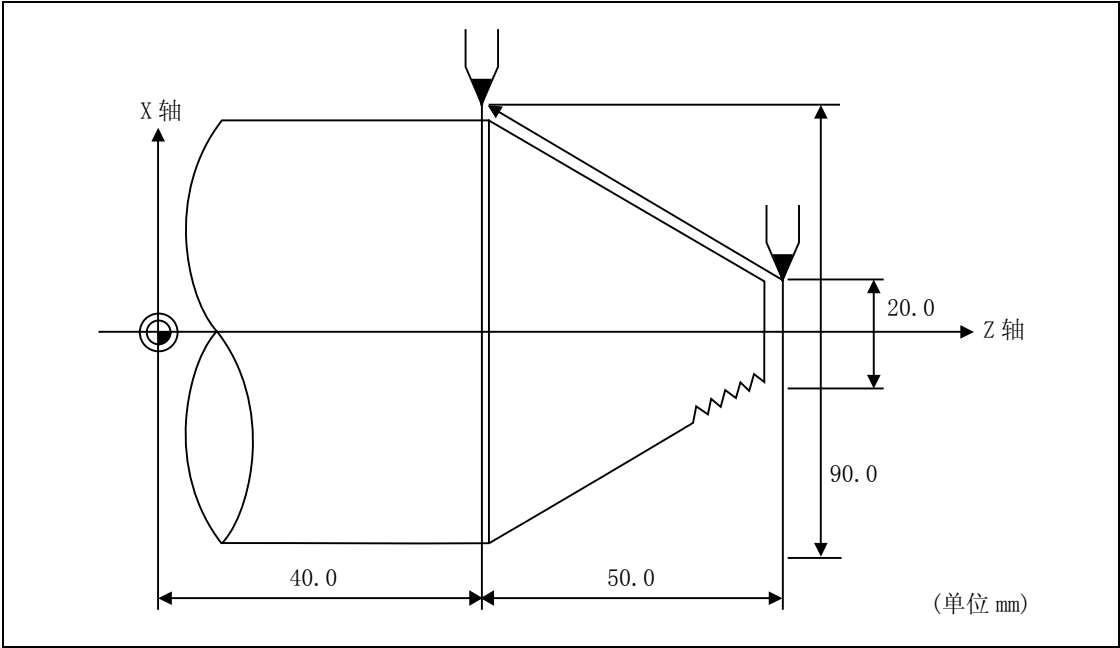
(注 1) 无法指定换算为每分钟进给后，进给速度高于最高切削进给速度的导程。

- (3) 请不要在锥形螺纹切削指令及直螺纹切削指令中使用线速度恒定控制。
- (4) 在粗切削到精整切削的过程中，主轴的转速请保持恒定。
- (5) 在螺纹切削中，如果因回馈等待而停止进给，可能会导致螺纹破损，所以在螺纹切削过程中，不能进行回馈等待。但是，从执行螺纹切削指令，到轴进行移动的过程中，回馈等待有效。
如果在螺纹切削中按了回馈等待开关，则螺纹切削结束后（退出G33模式后），在下一单节的终点停止。
- (6) 对于转换后的切削进给速度，在开始螺纹切削时，进行与切削进给钳位速度的比较，如果超过钳位速度，则发生操作错误。
- (7) 在螺纹切削中，为了保证导程，转换后的切削进给速度可能会超过切削进给钳位速度。
- (8) 在开始螺纹切削及螺纹切削结束时，通常会因为伺服系统的延迟而导致错误导程。
因此，在指定螺纹长度时，必须指定在所需螺纹长度上加上错误导程长度 δ_1 、 δ_2 后的长度。
- (9) 主轴的转速受到如下的限制。
- $$1 \leq R \leq \frac{\text{最高进给速度}}{\text{螺纹导程}}$$
- 但是， $R \leq$ 编码器的允许转速 (r/min)
 R: 主轴转速 (r/min)
 螺纹导程: mm或inch
 最高进给速度: mm/min或inch/min (受机械规格的限制)
- (10) 当相对于最高切削进给速度来说，螺纹导程过大，在(9)式中无法满足 $R < 1$ 时，可能会发生程序错误 (P97)。
- (11) 在螺纹切削中，空运转同样有效，但是空运转进给速度与主轴转速不同步。
在螺纹切削开始时，检查空运转信号，螺纹切削过程中进行的切换被跳跃。
- (12) 即使是在非同步进给 (G94) 指令时，螺纹切削指令也不会变为同步进给。
- (13) 在螺纹切削中，主轴超程及切削进给超程无效，变为100%固定。
- (14) 在刀尖R补偿中编入了螺纹切削指令，则临时取消刀尖R补偿后执行螺纹切削。
- (15) 如果在执行G33时将模式切换为其他自动模式，则在执行完下一单节非螺纹切削单节之后，自动运转停止。
- (16) 如果在执行G33时将模式切换为手动模式，则在执行完下一单节非螺纹切削单节之后，自动运转停止。
在单节时，在执行完下一单节非螺纹切削单节（退出G33模式）之后，自动运转停止。但是，在开始G33指令的轴移动之前，则立即自动运转停止。
- (17) 对于螺纹切削指令，等到收到编码器的1圈同步信号后，开始移动。
- (18) 在螺纹切削中，自动手轮插入有效。

- (19) 螺纹切削开始移位角度并非模态。当G33中没有Q指令时，看作为“Q0”。
- (20) 当在G33的Q中指令了超过360.000的值时，发生程序错误（P35）。
- (21) G33在1个循环中进行1条切削。当进行2条切削时，请变更Q的值，执行同一指令。



程序例



G33 X90.0 Z40.0 E12.34567;	绝对值指令
G33 U70.0 W-50.0 E12.34567;	增量值指令

6.6.2 英制螺纹切削；G33



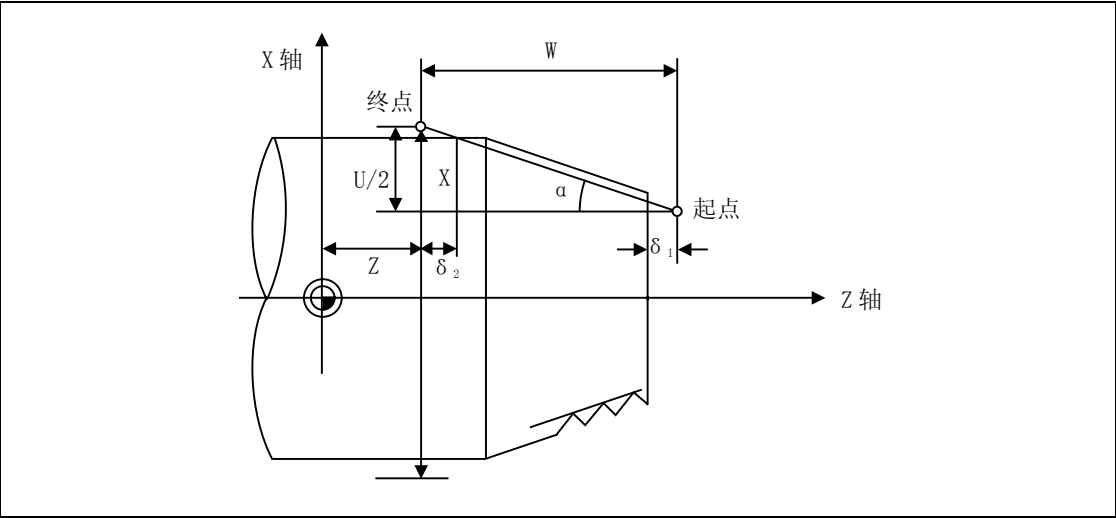
功能及目的

在G33指令中指定长轴方向的每英寸螺纹数，则进行与主轴旋转同步的刀具进给控制，所以可进行固定导程的直螺纹螺纹切削加工、锥形螺纹螺纹切削加工及连续螺纹螺纹切削加工。



指令格式

G33 Zz/Ww Xx/Uu Ee Qq ;	
Zz, Ww, Xx, Uu	: 螺纹的终点地址及坐标值
Ee	: 长轴（移动量最多的轴）方向的每英寸螺纹数（也可发出小数点指令）
Qq	: 螺纹切削开始移位角度（0.001~360.000°）

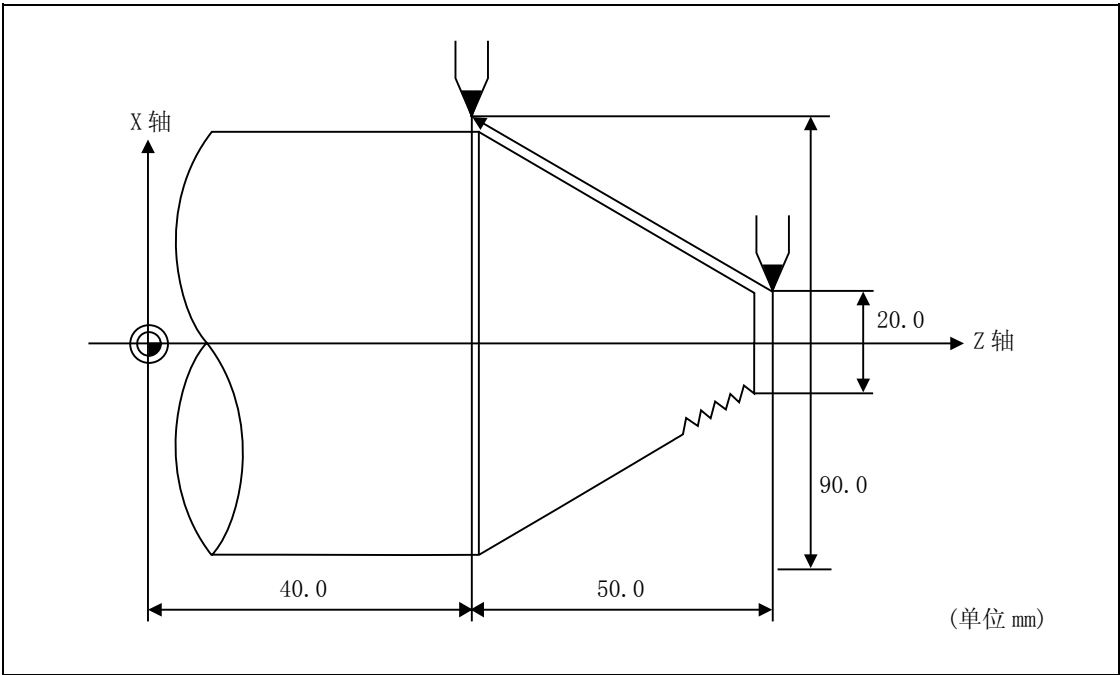


详细说明

- (1) 指定长轴方向的螺纹数作为每英寸的螺纹数。
- (2) E代码也可用于精密导程长度的指定，可通过参数选择是通过螺纹数量指定还是通过精密导程长度指定。（参数“#1229 set01/bit1”设定为“0”，则为通过螺纹数量进行指定。）
- (3) 关于E的指令值，请设定为进行导程换算之后，在导程值的范围之内的值。
- (4) 其他设定根据“6.6.1 固定导程螺纹切削”进行。。



程序例



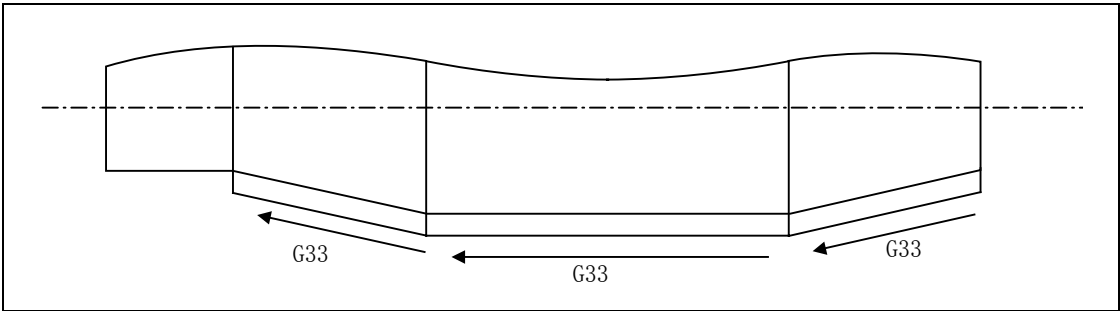
G33 X90.0 Z40.0 E12.0 ;	绝对值指令
G33 U70.0 W-50.0 E12.0 ;	增量值指令

6.6.3 连续螺纹切削



功能及目的

可通过连续发出螺纹切削指令，进行连续螺纹切削。
借此，可进行途中发生导程及形状变化的特殊螺纹的螺纹切削加工。



6.6.4 可变导程螺纹切削；G34



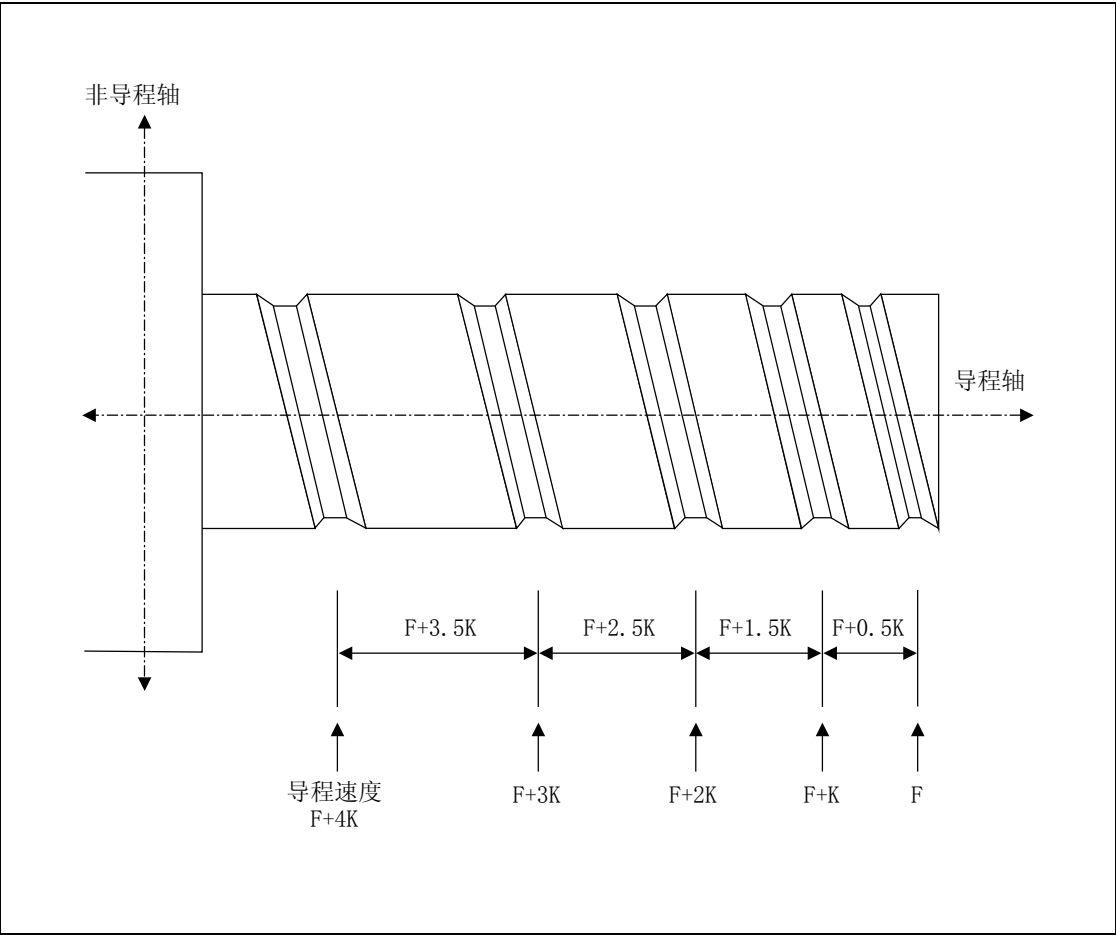
功能及目的

可通过指令每1圈螺纹的导程增减量，进行可变导程螺纹的螺纹切削加工。



指令格式

G34 Xx/Uu Zz/Ww Ff/Ee Kk ;	
Xx/Uu Zz/Ww	: 地址及坐标值
Ff/Ee	: 螺纹的基本导程
Kk	: 每 1 圈螺纹的导程增减量





详细说明

(1) 指令范围如下。

(a) E60

螺纹切削	公制输入		英制输入		K (n*mm/rev) n: 螺纹数 与 F 或 E 相同 (带符号)
输入单位体系	B (0.001mm)		B (0.0001inch)		
指令地址	F (mm/rev)	F (inch/rev)	F (inch/rev)	E (inch/rev)	
最小指令单位	1 (=0.0001) (1. =1.0000)	1 (=0.00001) (1.=1.00000)	1 (=0.000001) (1.=1.000000)	1 (=0.0000001) (1.=1.0000000)	
指令范围	0.0001~ 999.9999	0.00001~ 999.99999	0.000001~ 99.999999	0.0000001~ 9.9999999	

(b) E68

螺纹切削		公制输入		英制输入		K (n*mm/rev) n: 螺纹数 与 F 或 E 相同 (带符号)
输入单位体系	B (0.001mm)		C (0.0001mm)		B/C	
指令地址	F (mm/rev)	E (mm/rev)	F (mm/rev)	E (mm/rev)		
最小指令单位	1 (=0.0001) (1. =1.0000)	1 (=0.00001) (1.=1.00000)	1 (=0.00001) (1.=1.00000)	1 (=0.000001) (1.=1.000000)		
指令范围	0.0001~ 999.9999	0.00001~ 999.99999	0.00001~ 99.99999	0.000001~ 99.999999		

螺纹切削		英制输入			K (n*mm/rev) n: 螺纹数 与 F 或 E 相同 (带符号)
输入单位体系	B (0.0001inch)		C (0.00001inch)		
指令地址	F(inch/rev)	E(inch/rev)	F(inch/rev)	E(inch/rev)	
最小指令单位	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)	1(=0.0000001) (1.=1.0000000)	1(=0.00000001) (1.=1.00000000)	
指令范围	0.000001~ 99.999999	0.0000001~ 9.9999999	0.0000001~ 9.9999999	0.00000001~ 0.99999999	

(2) K为正时，变为增加螺距。

$$1 \text{ 单节的移动量 (n螺距)} = (F+K) + (F+2K) + (F+3K) + \dots + (F+nK)$$

(3) K为负时，变为减少螺距。

$$1 \text{ 单节的移动量 (n螺距)} = (F-K) + (F-2K) + (F-3K) + \dots + (F-nK)$$

(4) 当螺纹导程没有正确设定时，发生程序错误。

错误编号	内 容	处 理
P93	螺纹导程不正确 1) 发出螺纹切削指令时，F/E、K 的值不正确。 2) 最终导程超出 F/E 的指令范围。	确认 F/E、K 的值，正确进行设定。 参 1)

$$\text{参1) 最终导程} = \sqrt{F^2 + 2KZ}$$

$$\text{螺距数} = (-F + \text{最终导程}) / K$$

Z: 导程轴长度

(5) 其他内容与G33时相同。

请参阅“6.6.1 固定导程螺纹切削；G33”。



注意事项

可变导程螺纹切削功能的螺纹长度最大值取决于程序格式。

下表表示螺纹长度的最大值。超过该最大值，则发生程序错误（P60）。

指令单位（mm）	公制指令（mm）	英制指令（inch）
0.001	10737.377	10737.377
0.0001（E68）	1073.7377	1073.7377

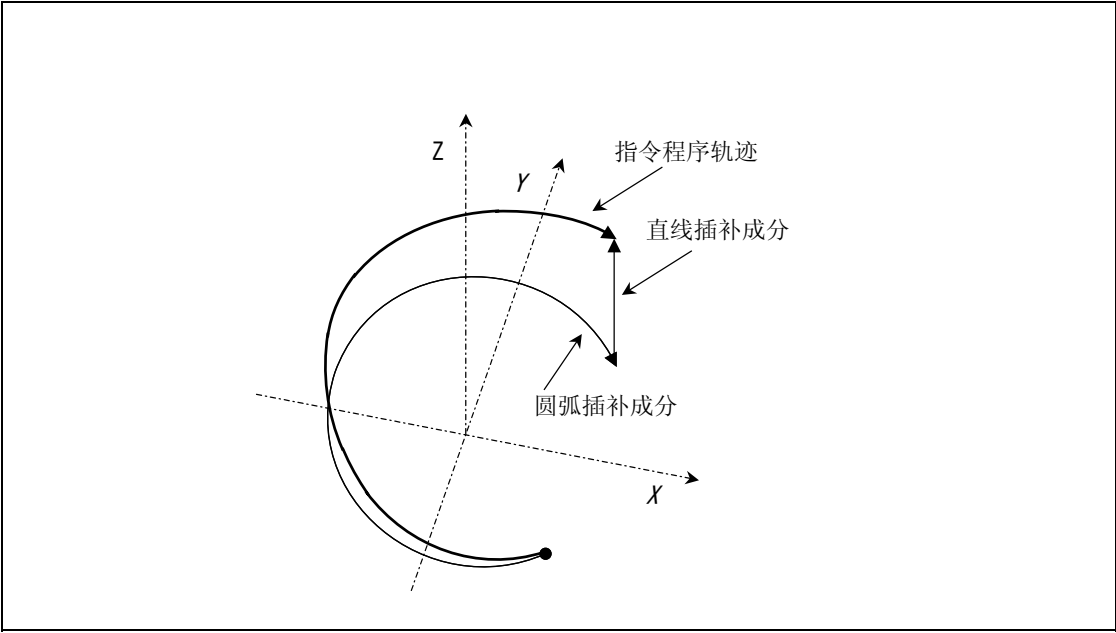
6.7 螺旋插补；G17、G18、G19 及 G02、G03

[E68]



功能及目的

对被包含在平面选择中的2轴进行圆弧插补，同时，同步进行其他轴的直线插补的功能。
对直角相交的3根轴执行本功能，可让刀具沿螺旋形轨迹进行移动。



指令格式

G17	G02 (G03)	Xx/Uu	Yy/Vv	Zz/Ww	Ii	Jj	Ff ;
G17	G02 (G03)	Xx/Uu	Yy/Vv	Zz/Ww	Rr	Ff ;	
G17		圆弧平面 (G17: XY 平面、G18: ZX 平面、G19: YZ 平面)					
G02 (G03)		圆弧旋转方向 (G02: 顺时针、G03: 逆时针)					
Xx/Uu, Yy/Vv		圆弧终点坐标					
Zz/Ww		直线轴终点坐标					
Ii, Jj		圆弧中心坐标					
Rr		圆弧半径					
Ff		进给速度					

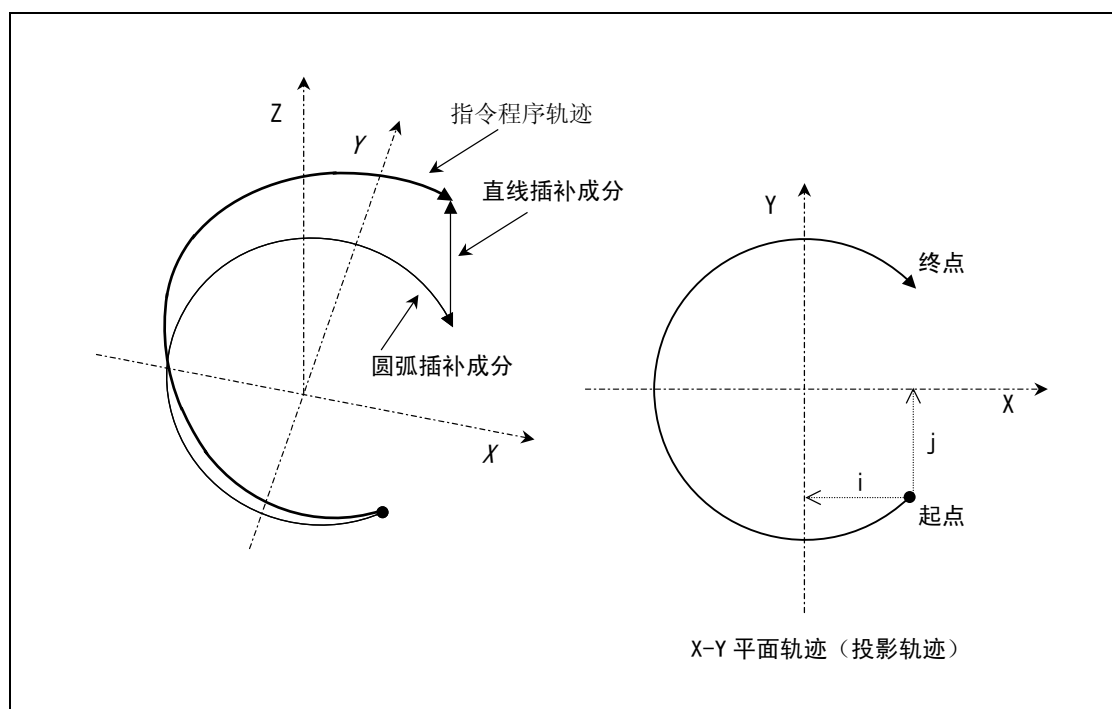
(注1) 本书以 I 轴: X、J 轴: Y、K 轴: Z 的设定为前提，加以记载。



详细说明

当执行如下所示的指令时，变为如下图所示的动作。

G17 G02 Xx Yy Zz Ii Jj Ff;



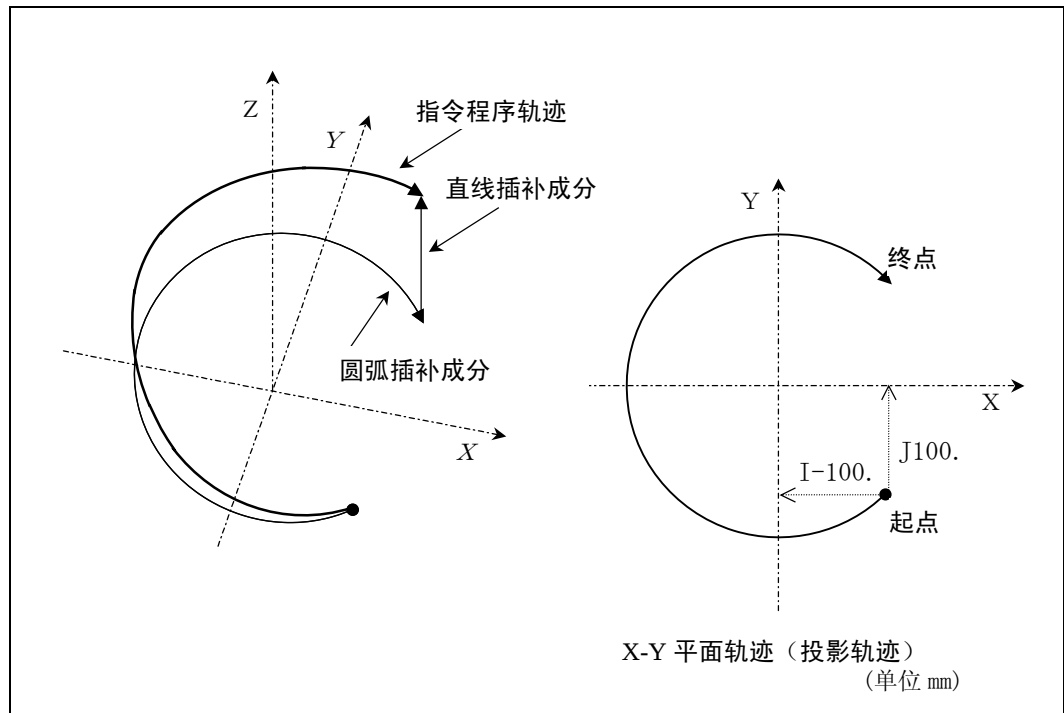
左图为处理的立体视图，右图为从正上方看圆弧平面的视图。



程序例

(例)

```
G17 G02 X100. Y100. Z100. I-100. J100. F120 ;
```



左图为处理的立体视图，右图为从正上方看圆弧平面的视图。

以从单节开始时的工件坐标（起点）向X轴方向移动-100mm，向Y轴方向移动100mm后的点为中心，一边旋转，一边以120mm/min的进给速度开始切削。



注意事项・限制事项

- (1) 进行螺旋插补时，请指定不包含圆弧插补指令与圆弧轴的其他直线轴（可指定多根轴）。
- (2) 可同时发出指令的轴为同时轮廓控制轴数。
- (3) 无法执行超过1圈的指令。（取决于圆弧插补指令的规格。）
- (4) 请指定各轴合成速度，作为进给速度。
- (5) 在螺旋插补中，构成平面的轴为圆弧插补轴，其他轴为直线插补轴。
- (6) 无法进行图像检查中的描画。（无法正确描画。）
- (7) 在转角倒角/转角R中，直线插补轴停止移动，仅圆弧插补轴进行移动。
- (8) 注意事项等请参阅圆弧插补（G02、G03）的说明。

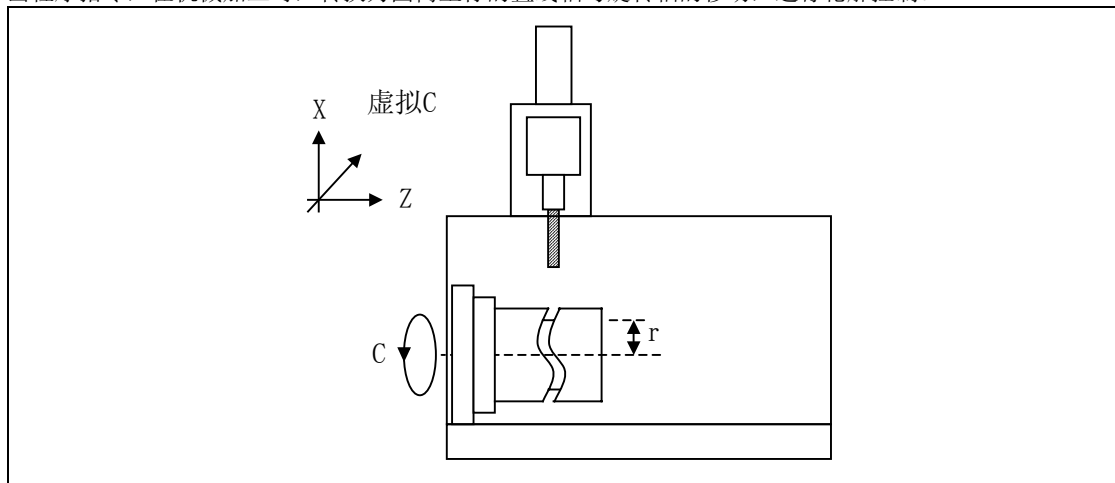
6.8 圆筒插补: G07.1 (仅G代码系列6、7)

[E68]

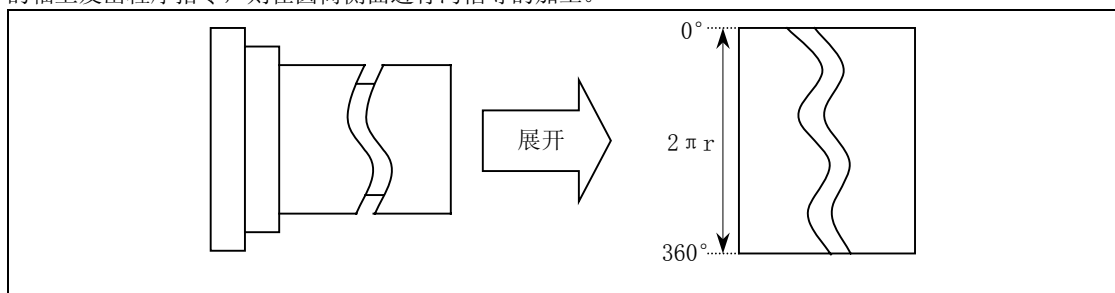


功能及目的

本功能是将位于圆筒侧面的形状（圆筒坐标系中的形状）展开为平面，将展开后的形状作为平面坐标，发出程序指令，在机械加工时，转换为圆筒坐标的直线轴与旋转轴的移动，进行轮廓控制。



由于可以将圆筒侧面展开后的形状进行编程，所以可进行圆筒凸轮等的加工。在旋转轴及与之直角相交的轴上发出程序指令，则在圆筒侧面进行沟槽等的加工。





指令格式

G07.1 Cc ;	(圆筒插补模式开始/取消)
Cc	圆筒半径值
	• 半径值 $\neq 0$; 圆筒插补模式开始
	• 半径值 $= 0$; 圆筒插补模式取消

(注) 上述格式是旋转轴名称为“C”时的设定。“C”的轴时，请使用所使用旋转轴的名称代替“C”。

- (1) 从圆筒插补模式开始到取消圆筒插补模式之间的坐标指令，为圆筒坐标系内的指令。
- G07.1 C 圆筒半径值 ;

圆筒插补模式开始

(开始圆筒插补)

:

:

(这一区间内的坐标指令为圆筒坐标系内的指令)

:

G07.1 C0 ;

圆筒插补模式取消

(结束圆筒插补)
- (2) 也可使用G107代替G07.1。
- (3) 以单节指定G07.1。如果在同一单节中指定其他G代码，则发生程序错误 (P33)。
- (4) 以角度对旋转轴进行编程。
- (5) 圆筒插补模式中，可发出直线插补或圆弧插补指令。但是在G07.1单节之前，请执行平面选择指令。
- (6) 坐标指令采用绝对指令、增量指令均可。
- (7) 对于程序指令，可附加刀具直径补偿。对刀具直径补偿后的路径进行圆筒插补。
- (8) 对于进给速度，请以F指定圆筒展开中的接线速度。F单位为mm/min或inch/min。



详细说明

- (1) 圆筒插补的精度
- 在圆筒插补模式中，将以角度进行指令的旋转轴移动量转换为圆周上的距离，进行与其他轴之间的直线、圆弧插补运算之后，再次转换为角度。
- 为此，当圆筒的半径较小时，实际的移动量可能会与指令值不同。不过，此时发生的误差不会累积。

(2) 相关参数

#	项 目		内 容	设定范围 (单位)
1516	mill_ax	铣削轴名称	设定极坐标插补、圆筒插补用旋转轴的轴名称。仅设定旋转轴中的一轴。	A~Z
8111		铣削半径值	选择进行极坐标插补、圆筒插补的直线轴的直径・半径。 0: 所有轴半径指令 1: 各轴设定 (根据#1019 dia 直径指定轴)	0 / 1
1267 (PR)	ext03 (bit0)	G代码切换	切换高速高精度的 G 代码类型。 0: G61.1型 1: G8型	0 / 1
1270 (PR)	ext06 (bit7)	圆筒插补中C轴坐标的操作	指定在圆筒插补中, 是否继续圆筒插补开始指令前的旋转轴坐标。 0: 不继续 1: 继续	0 / 1

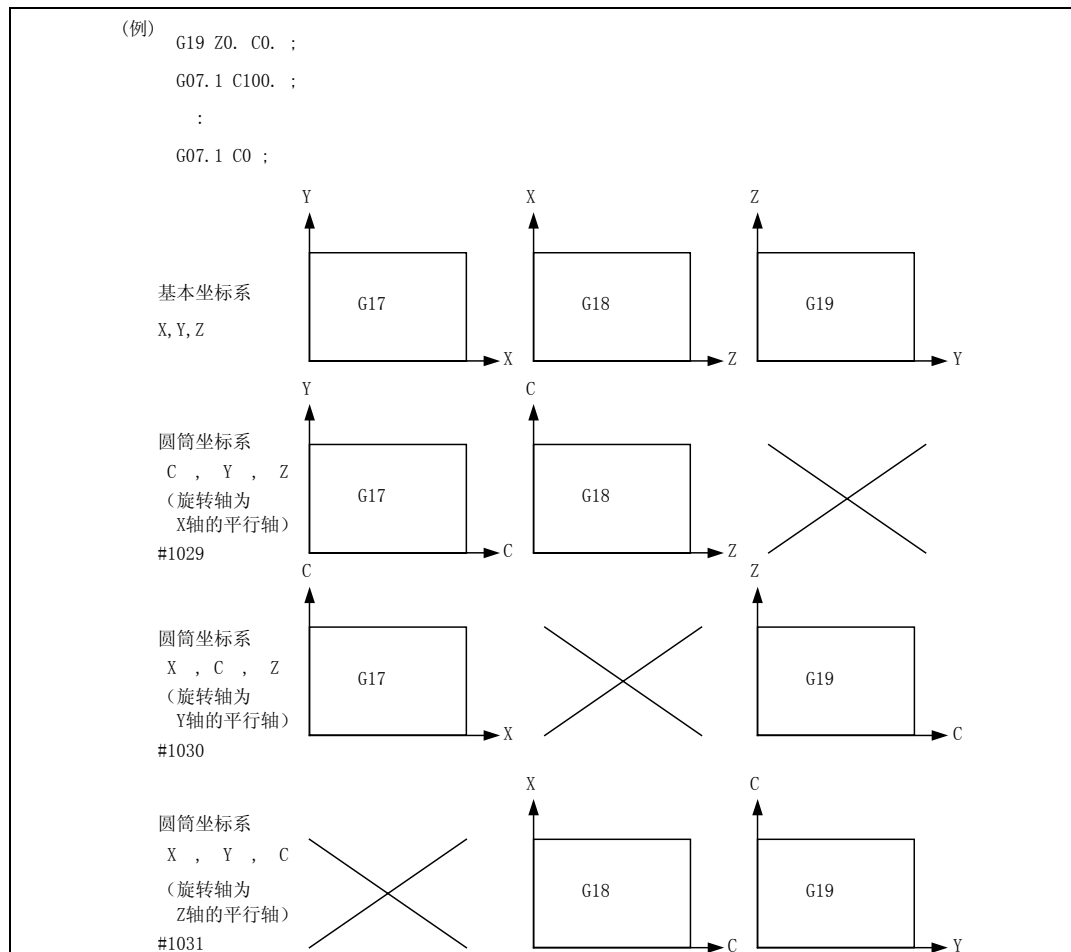
(3) 平面选择

必须以平面选择指令设定圆筒插补。(注)

通过参数(#1029、#1030、#1031)设定旋转轴与哪根轴的平行轴对应。

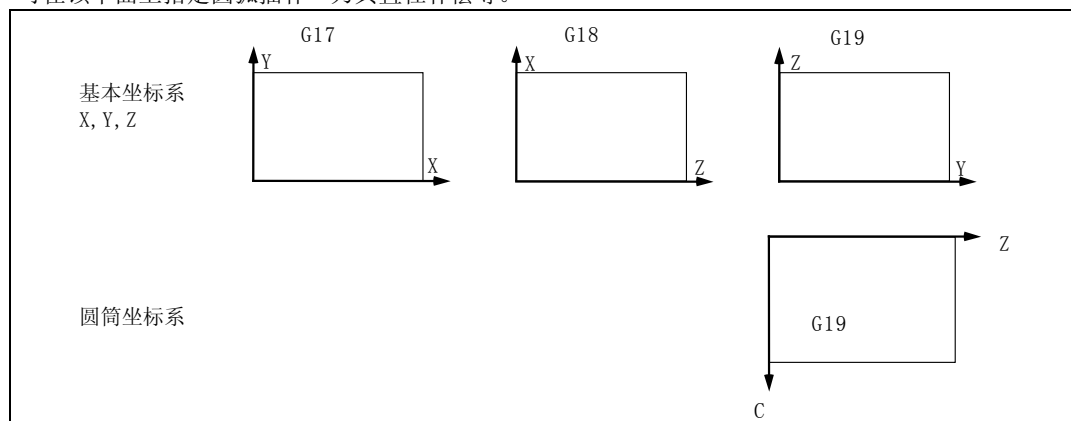
可在该平面上指定圆弧插补·刀具直径补偿等。

在执行G07.1指令之前或之后设定平面选择指令,如果在未设定时有移动指令,则发生程序错误(P485)。



(注) 根据机种或版本数不同,通过 G07.1 与 G19, Z-C 平面(Y-Z 圆筒平面)自动被选中。

可在该平面上指定圆弧插补·刀具直径补偿等。





与其他功能的关联

(1) 圆弧插补

- (a) 在圆筒插补模式中，可在旋转轴与直线轴之间进行圆弧插补。
- (b) 在圆弧插补中，可执行R指定指令。（无法进行I、J、K指定）

(2) 刀具直径补偿

在圆筒插补模式中，可进行刀具直径补偿。

- (a) 请与圆弧插补一样，执行平面选择指令。
进行刀具直径补偿时，请在圆筒插补模式中进行起动、取消。
- (b) 在刀具直径补偿中发出G07.1指令则发生程序错误（P485）。
- (c) 如果在刀具直径补偿取消后，没有执行移动指令就直接指定G07.1指令，则将G07.1指令单节的轴位置看作刀具直径补偿取消后的位置，进行以下的动作。

(3) 切削非同步进给

- (a) 随着圆筒插补模式开始，强制性的进入非同步模式。
- (b) 随着圆筒插补模式取消，同步模式恢复为圆筒插补模式开始前的状态。
- (c) 在线速度恒定控制模式中（G96）执行G07.1指令，则发生程序错误（P485）。

(4) 辅助功能

- (a) 在圆筒插补模式中也可指定辅助功能（M）及第2辅助功能。
- (b) 请不要使用主轴转速指定圆筒插补模式中的S指令，而是使用旋转刀具的转速进行指定。
- (c) 请在圆筒插补开始之前指定T指令。在圆筒插补模式中执行T指令，则发生程序错误（P485）。

```

:
:
T1212 ;           ...执行圆筒插补前的T指令 → 可
G0 X100. Z0. ;
G19 Z C ;
G07.1 C100. ;
:
T1200 ;           ...圆筒插补模式中的T指令 → 程序错误
:
G07.1 C0 ;

```

- (d) 在圆筒插补开始之前，请完成刀具补偿动作（刀具长度及磨损补偿量的移动）。

执行圆筒插补开始指令时，如果刀具补偿动作未完成，则如下。

- 即使指定G07.1指令，机械坐标也不发生变化。
- 执行G07.1指令，则工件坐标变为刀具长度补偿动作后的值。
(即使取消圆筒插补，该工件坐标也不会被解除。)

(5) 关于圆筒插补中的F指令

根据执行F指令前的每分钟进给指令（G94）、每转进给指令（G95）的模式，决定是否使用圆筒插补模式中的F指令。

(a) G07.1之前为G94时

当在圆筒插补中没有F指令时，直接使用之前的F指令的进给速度。
圆筒插补模式取消后的进给速度，保持圆筒插补模式开始时，或是圆筒插补中所设定的最终F指令的进给速度。

(b) G07.1之前为G95时

圆筒插补中，由于无法直接使用之前的F指令的进给速度，所以必须执行新的F指令。
圆筒插补模式取消后的进给速度，恢复到圆筒插补模式开始前的状态。

G07.1中没有F指令时

之前的模式	无 F 指令	G07.1 取消后
G94	使用之前的 F	←
G95	程序错误（P62）	使用 G07.1 之前的 F

G07.1中有F指令时

之前的模式	有 F 指令	G07.1 取消后
G94	使用指定的 F	←
G95	使用指定的 F *1	使用 G07.1 之前的 F

*1) 在G07.1中，按每分钟进给指令进行动作



限制与注意事项

(1) 圆筒插补模式中，能够使用的G代码指令如下。

G 代码	内容
G00	定位
G01	直线插补
G02	圆弧插补 (CW)
G03	圆弧插补 (CCW)
G04	延时
G09	精确停止检查
G22/23	卡盘屏障开/关
G40-42	刀具直径补偿
G61	精确停止模式
G64	切削模式
G65	用户宏 (纯调用)
G66	用户宏 (模态调用)
G66.1	用户宏 (各宏单节调用)
G67	用户宏 (模态调用取消)
G80-89	钻孔固定循环
G90/91	绝对/增量
G94	非同步进给
G98	固定循环/起始点回归
G99	固定循环/R 点回归

如果在圆筒插补中指定了上述以外的G代码，则可能会发生程序错误 (P481)。

- (2) 在接通电源时及重新启动时，变为圆筒插补模式取消状态。
- (3) 如果圆筒插补中的指令轴中，包括原点回归未完成的轴，则发生程序错误 (P484)。
- (4) 在取消圆筒插补模式时，必须取消刀具直径补偿。
- (5) 通过取消圆筒插补模式切换为车削模式，返回圆筒插补前所选中的平面。
- (6) 对于圆筒插补中的单节，无法重新启动程序 (程序继续)。
- (7) 当在镜像中执行了圆筒插补指令时，发生程序错误 (P486)。
- (8) 开始与取消圆筒插补模式时，进行减速检查。
- (9) 在圆筒插补模式中指定圆筒插补、极坐标插补，则发生程序错误 (P481)。
- (10) 在圆筒插补模式中，无法使用G84、G88的同步式螺纹切削循环。虽然在圆筒插补模式中可使用同步螺纹切削，但是请不要进行同步螺纹切削指令。



程序例

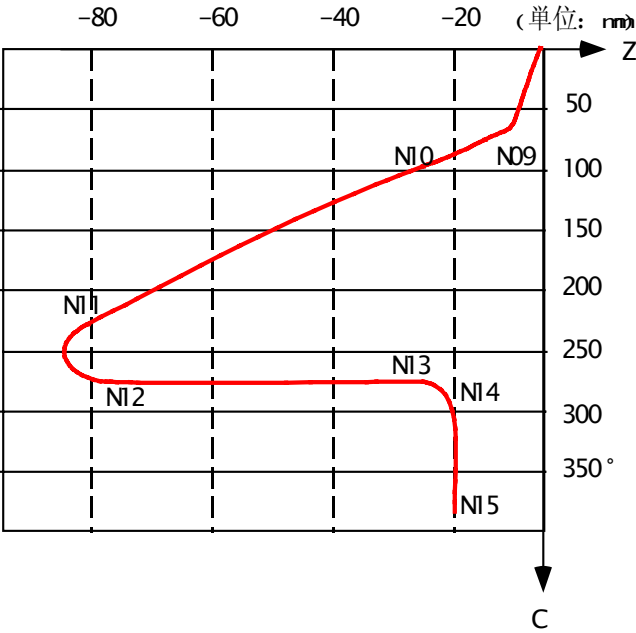
〈程序〉

```
N01 G28XZC;  
N02 T0202;  
N03 G97S100M23;  
N04 G00X50. Z0. ;  
N05 G94G01X40. F100. ;  
N06 G19C0Z0; ←  
N07 G07. 1C20. ; ←  
N08 G41;  
N09 G01Z-10. C80. F150;  
N10 Z-25. C90. ;  
N11 Z-80. C225;  
N12 G03Z-75. C270. R55. ;  
N13 G01Z-25;  
N14 G02Z-20. C280. R80. ;  
N15 G01C360.  
N16 G40;  
N17 G07. 1C0; ←  
N18 G01X50. ;  
N19 G0X100. Z100. ;  
N20 M25;  
N21 M30;
```

〈参数〉		
#1029	aux_I	
#1030	aux_J	C
#1031	aux_K	

进行圆筒插补的平面选择指令与进行插补的2轴指令
圆筒插补开始

圆筒插补取消



6.9 极坐标插补：G12.1（仅G代码系列6、7）

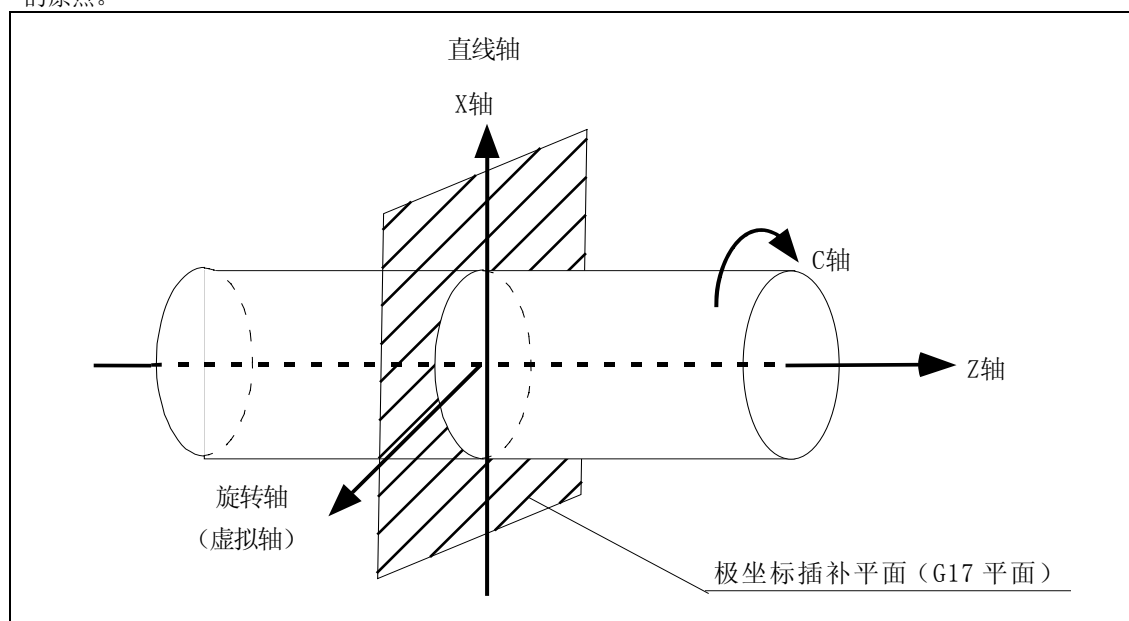
[E68]



功能及目的

本功能是将直角相交坐标系上编程的指令转换为直行轴的移动（刀具的移动）与旋转轴的移动（工件的旋转），进行轮廓控制。

以直线轴作为平面第1轴直角相交轴、以直角相交的虚拟轴作为平面第2轴的平面（以下称为“极坐标插补平面”）被选中。在该平面上进行极坐标插补。另外，在极坐标插补中，将工件坐标系的原点作为坐标系的原点。



在工件外径上切削位于直线上的切除部位时，以及研削凸轮轴等时，是很有效的功能。



指令格式

G12.1 ;	极坐标插补模式开始
G13.1 ;	极坐标插补模式取消

- (1) 从极坐标插补模式开始，到取消极坐标插补模式之间的坐标指令，为极坐标插补。
- G12.1 ;

:

:

:

G13.1 ;

极坐标插补模式开始

(开始极坐标插补)

(这一区间内的坐标指令为极坐标插补)

极坐标插补模式取消

(结束极坐标插补)
- (2) 也可使用G112、G113代替G12.1、G13.1。
- (3) 以单节指定G12.1、G13.1。如果在同一单节中指定其他G代码，则发生程序错误（P33）。
- (4) 极坐标插补模式中，可发出直线插补或圆弧插补指令。
- (5) 坐标指令采用绝对指令、增量指令均可。
- (6) 对于程序指令，可附加刀具直径补偿。对刀具直径补偿后的轨迹进行极坐标插补。
- (7) 对于进给速度，请以F指定极坐标插补平面（直角相交坐标系）中的接线速度。F单位为mm/min或inch/min。
- (8) 执行G12.1/G13.1指令时，进行减速检查。



详细说明

- (1) 平面选择
- 必须通过参数预先设定进行极坐标插补的直线轴与旋转轴。
- (a) 通过进行极坐标插补的直线轴的参数（#1533），决定进行极坐标插补的构成平面。
- | #1533的设定值 | 构成平面 |
|-----------|-----------|
| X | G17（XY平面） |
| Y | G19（YZ平面） |
| Z | G18（ZX平面） |
| 空白（无设定） | G17（XY平面） |
- (b) 在极坐标插补模式中执行平面选择指令（G16～G19），则发生程序错误（P485）。
- (注) 根据机种及版本不同，可能会有没有参数（#1533）的设备。此时，与参数（#1533）为空白（无设定）时的动作相同。

(2) 相关参数

#	项 目	内 容	设定范围（单位）
1516	mill_ax 铣削轴名称	设定极坐标插补、圆筒插补用旋转轴的轴名称。仅设定旋转轴中的一轴。	A～Z
1517	mill_c 铣削插补虚拟轴名称	在极坐标插补、圆筒插补中，选择虚拟轴的指令名称。 0: Y轴指令 1: 指定旋转轴名称	0 / 1
8111	铣削半径值	选择进行极坐标插补、圆筒插补的直线轴的直径・半径。 0: 所有轴半径指令 1: 各轴设定（根据#1019 dia 直径指定轴）	0 / 1



与其他功能的关联

(1) 极坐标插补中的程序指令

- (a) 极坐标插补模式中的程序指令，是在极坐标插补平面上，根据直线轴与旋转轴（虚拟轴）的直角相交坐标值进行指定。
在平面第2轴（虚拟轴）的指令轴地址中，指定旋转轴（C）的轴地址。
指令单位不是deg（度），而是与平面第1轴（直线轴）的轴地址指令单位相同（mm或inch）。
- (b) 指定G12.1时，虚拟轴坐标值变为“0”。即，将指定G12.1时的位置看作为角度=0，开始极坐标插补。

(2) 极坐标平面上的圆弧插补

根据直线轴参数（#1533），决定在极坐标插补模式中进行圆弧插补时的圆弧半径地址。

#1533的设定值	中心指定指令
X	I、J（将极坐标平面看作为XY平面）
Y	J、K（将极坐标平面看作为YZ平面）
Z	K、I（将极坐标平面看作为ZX平面）
空白（无设定）	I、J（将极坐标平面看作为XY平面）

另外，也可通过R指令指定圆弧半径。

（注）根据机种及版本不同，可能会有没有参数（#1533）的设备。此时，与参数（#1533）为空白（无设定）时的动作相同。

(3) 刀具直径补偿

在极坐标插补模式中，可进行刀具直径补偿。

- (a) 请与极坐标插补一样，执行平面选择指令。
进行刀具直径补偿时，请在极坐标插补模式中进行起动、取消。
- (b) 在刀具直径补偿中进行极坐标插补，则发生程序错误（P485）。
- (c) 如果在刀具直径补偿取消后，没有执行移动指令就直接指定G12.1/G13.1指令，则将G12.1/G13.1指令单节的轴位置看作刀具直径补偿取消后的位置，进行以下的动作。

(4) 切削非同步进给

- (a) 随着极坐标插补模式开始，强制性的进入非同步模式。
- (b) 随着极坐标插补模式取消，同步模式恢复为极坐标插补模式开始前的状态。
- (c) 在线速度恒定控制模式中（G96）执行G12.1指令，则发生程序错误（P485）。

(5) 辅助功能

- (a) 在极坐标插补模式中也可指定辅助功能（M）及第2辅助功能。
- (b) 请不要使用主轴转速指定极坐标插补模式中的S指令，而是使用旋转刀具的转速进行指定。
- (c) 请在极坐标插补开始之前指定T指令。在极坐标插补模式中执行T指令，则发生程序错误（P485）。

```
      :  
      :  
      T1212 ;           ...执行极坐标插补前的T指令 → 可  
      G0 X100. Z0. ;  
      G12.1 ;  
      :  
      T1200 ;           ...极坐标插补模式中的T指令 → 程序错误  
      :  
      G13.1 ;
```

- (d) 在极坐标插补开始之前，请完成刀具补偿动作（刀具长度及磨损补偿量的移动）。

执行极坐标插补开始指令时，如果刀具补偿动作未完成，则如下。

- 即使执行G12.1指令，机械坐标也不发生变化。
- 执行G12.1指令，则工件坐标变为刀具长度补偿动作后的值。
（即使取消极坐标插补，该工件坐标也不会被解除。）

（例）

工件坐标偏移（X轴）=20.

T0101的刀具补偿量（X轴）=100.

通过T指令后的移动指令设定补偿动作

〈加工程序〉	〈工件坐标〉		〈机械坐标〉		
:	[X 轴]	[C 轴]	[X 轴]	[C 轴]	
G00 X200. C0. ;	200.	0.	220.	0.	
T0101;	200.	0.	220.	0.	
G12.1 ;	100.	0.	220.	0.	←移动工件坐标系（没有轴的移动）
G01 X50. F1000;	50.	0.	170.	0.	
:					

(6) 关于极坐标插补中的F指令

根据执行F指令前的每分钟进给指令（G94）、每转进给指令（G95）的模式，决定是否使用极坐标插补模式中的F指令。

(a) G12.1之前为G94时

当在极坐标插补中没有F指令时，直接使用之前的F指令的进给速度。

极坐标插补模式取消后的进给速度，保持极坐标插补模式开始时，或是极坐标插补中所设定的最终F指令的进给速度。

(b) G12.1之前为G95时

极坐标插补中，无法使用之前的F指令的进给速度。必须指定新的F指令。

极坐标插补模式取消后的进给速度，恢复到极坐标插补模式开始前的状态。

G12.1中没有F指令时

之前的模式	无 F 指令	G13.1 后
G94	使用之前的 F	←
G95	程序错误 (P62)	使用 G12.1 之前的 F

G12.1中有F指令时

之前的模式	有 F 指令	G13.1 后
G94	使用指定的 F	←
G95	使用指定的 F *1	使用 G12.1 之前的 F

*1) 在G12.1中，以每分钟进给指令进行动作

(7) 极坐标插补的钻孔固定循环指令中的钻孔轴

在极坐标插补模式中，通过直线轴参数（#1533）决定钻孔固定循环指令中的钻孔轴。

#1533的设定值	钻孔轴
X	Z（将极坐标平面看作为XY平面）
Y	X（将极坐标平面看作为YZ平面）
Z	Y（将极坐标平面看作为ZX平面）
空白（无设定）	Z（将极坐标平面看作为XY平面）



限制与注意事项

(1) 极坐标插补模式中，能够使用的G代码指令如下。

G 代码	内容
G00	定位
G01	直线插补
G02	圆弧插补 (CW)
G03	圆弧插补 (CCW)
G04	延时
G09	精确停止检查
G22/23	卡盘屏障开/关
G40~42	刀具直径补偿
G61	精确停止模式
G64	切削模式
G65	用户宏 (纯调用)
G66	用户宏 (模态调用)
G66.1	用户宏 (各宏单节调用)
G67	用户宏 (模态调用取消)
G80~89	钻孔固定循环
G90/91	绝对/增量
G94	非同步进给
G98	固定循环/起始点回归
G99	固定循环/R 点回归

如果在极坐标插补中指定了上述以外的G代码，则可能会发生程序错误 (P481)。

(2) 对于极坐标插补中的单节，无法重新启动程序 (程序继续)。

(3) 在指定极坐标插补之前，请设定工件坐标系，确保旋转轴的中心成为坐标系原点。在极坐标插补模式中，请不要进行坐标系的变更。(G50、G52、G53、相对坐标的重新启动、G54~G59等)

(4) 极坐标插补中的进给速度，变为极坐标插补平面 (直角相交坐标系) 上的插补速度。

(根据极坐标转换，与刀具间的相对速度发生变化)

当在极坐标插补平面 (直角相交坐标系) 上通过旋转轴的中心附近时，极坐标插补后的旋转轴端进给速度变的很高。

(5) 极坐标插补中的平面外轴移动指令，与极坐标插补无关。

(6) 极坐标插补中的当前位置显示，全部是以实际的坐标值加以显示，仅“剩余移动量”显示极坐标输入平面上的移动量。

(7) 在接通电源时及重新启动时，变为极坐标插补模式取消状态。

(8) 如果极坐标插补中的指令轴中，包括原点回归未完成的轴，则发生程序错误 (P484)。

(9) 在取消极坐标插补模式时，必须取消刀具直径补偿。

(10) 通过取消极坐标插补模式切换为车削模式，返回极坐标插补前所选中的平面。

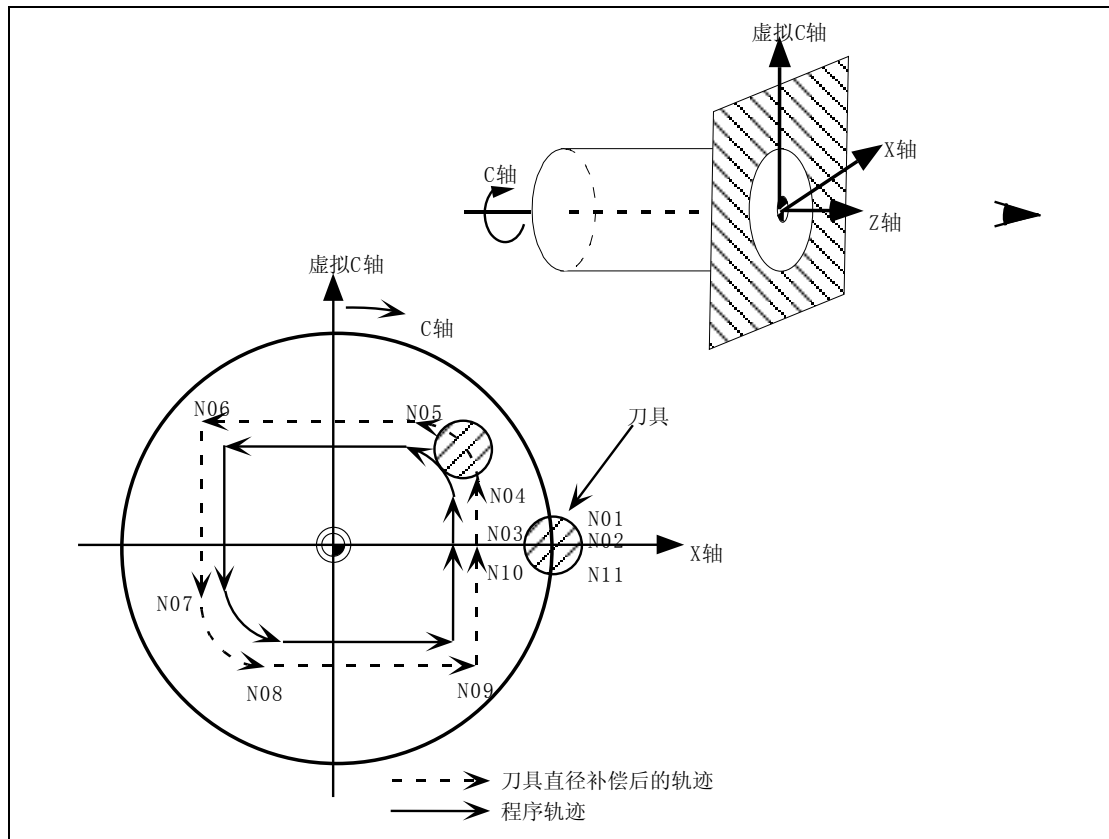
(11) 当在镜像中执行了极坐标插补指令时，发生程序错误 (P486)。

(12) 在极坐标插补模式中指定圆筒插补、极坐标插补，则发生程序错误 (P481)。

(13) 在极坐标插补模式中，无法使用G84、G88的同步式螺纹切削循环。虽然在极坐标插补模式中可使用同步螺纹切削，但是请不要进行同步螺纹切削指令。



程序例



<程序>

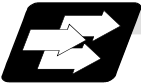
```

:
:
N00 T0101;
:
:
N01 G17 G90 G0 X40.0 C0 Z0;      开始位置的确定
N02 G12.1;                      极坐标插补模式: 开始
N03 G1 G42 X20.0 F2000;          实际加工开始
N04 C10.0;
N05 G3 X10.0 C20.0 R10.0;
N06 G1 X-20.0;                  形状程序
N07 C-10.0;
N08 G3 X-10.0 C-20.0 I10.0 J0;   (根据 X-C 虚拟轴平面上的直角相交坐标值。)
N09 G1 X20.0;
N10 C0;
N11 G40 X40.0;
N12 G13.1;                      极坐标插补模式: 取消
:
:
M30 ;

```

7. 进给功能

7.1 快速进给速度



功能及目的

可各轴独立设定快速进给速度。当输入设定单位为 $1\mu\text{m}$ 时，可设定的速度范围为 $1\text{mm/min}\sim 240,000\text{mm/min}$ （E60）或 $1\text{mm/min}\sim 1,000,000\text{mm/min}$ （E68）之间。但是根据机械规格不同，上限速度受到限制。

关于快速进给速度的设定值，请参阅机械的规格书。

定位时的轨迹，包括以直线在起点与终点之间进行插补的插补型，与以各轴的最高速度进行移动的非插补型，通过参数“#1086 G0Intp”进行选择。定位所需时间均相同。

（注）快速进给超程

对于手动与自动快速进给，可通过外部输入信号进行超程。这里有 2 种类型，取决于 PLC 规格。

类型 1：进行 1%、25%、50%、100%共 4 个阶段的超程。

类型 2：从 0%到 100%，以每 1%为一档进行超程。

7.2 切削进给速度



功能及目的

通过地址F与8位数字指定切削进给速度。

F8位的指定，包括整数部分5位与小数部分3位，带小数点。切削进给速度对G01、G02、G03、G33指令有效。

例（非同步进给）

进给速度				
G1	X100.	Z100.	F200 ;	200.0mm/min F200 等同于 F200.000。
F1	X100.	Z100.	F123.4 ;	123.4mm/min
G1	X100.	Z100.	F56.789 ;	56.789mm/min

可指令的速度范围

（1）E60

指令模式	F 指令范围	进给速度范围	备 注
mm/min	0.001 ~240000.000	0.001 ~240000.000 mm/min	
inch/min	0.0001~9448.8188	0.0001~9448.8188 inch/min	
° /min	0.001 ~240000.000	0.001 ~240000.000 ° /min	

（2）E68（输入设定单位为 $1\mu\text{m}$ 时）

指令模式	F 指令范围	进给速度范围	备 注
mm/min	0.001~ 1000000.000	0.001~ 1000000.000 mm/min	
inch/min	0.0001~ 39370.0787	0.0001~ 39370.0787 inch/min	
° /min	0.001~ 1000000.000	0.001~ 1000000.000 ° /min	

（注 1）接通电源后，当最初的切削指令（G01、G02、G03、G33、G34）中没有 F 指令时，发生程序错误（P62）。

7.3 F1数位进给



功能及目的

通过设置F1数位进给参数，以预先设定好的，与地址F后的1位数值所对应的进给速度，作为指令速度。
指定F0，则变为快速进给速度，变为与G00时相同的进给速度。（G模态不变。）
如果发出F1～F5指令，则分别与之对应设定的进给速度成为实际生效的指令速度。
F6位以上的指令，被看作为通常的切削进给速度。
F1数位指令仅在G01、G02、G03模态中有效。
F1数位也可用于固定循环。



详细说明

- (1) 当F1数位有效时，F1数位指令可与通常的切削进给速度指令并用。
(例1)
F0.....快速进给速度
F1～F5.....F1数位
F6以上.....通常的切削进给速度指令
- (2) G00模式时，F1～F5无效，成为快速进给速度。
- (3) 在G02、G03模式中使用F0，则发生程序错误（P121）。
- (4) F1. ～F5.（带小数点）时，不是F1数位指令，而是变为1 mm/min～5 mm/min的进给速度。
- (5) 当使用公制或度指令时，与F1～F5对应设定的进给速度，成为指令速度mm（°）/min。
- (6) 当在英制指令中使用时，与F1～F5对应设定的进给速度的1/10，成为指令速度inch/min。
- (7) F1数位指令中，将F1数位指令中信号与F1数位编号输出为PLC信号。
- (8) F1数位与G指令
 - (a) F1数位与同一单节的01组G指令

	执行进给速度	模态显示速度	G 模态
G0F0 F0G0	快速进给速度	0	G0
G0F1 F1G0	〃	1	G0
G1F0 F0G1	〃	0	G1
G1F1 F1G1	F1 的内容	1	G1

- (b) 也可在同一单节内指令F1数位与非模态指令。此时，执行非模态指令，同时，F1数位的模态也进行更新。

7. 进给功能

7.4 同步进给/非同步进给

7.4 同步进给/非同步进给 ; G94、G95



功能及目的

可通过G95指令，使用F代码指定每转的进给量。当使用本指令时，主轴上必须安装有旋转编码器。



指令格式

G94 ;
G95 ;

G94 : 每分钟进给 (mm/min) (非同步进给)
 G95 : 每转进给 (mm/rev) (同步进给)

由于G95指令是模态指令，所以在下一次指定G94（每分钟进给）指令前始终有效。

(1) F代码所决定的指令范围如下。

以F指令同步进给（每转进给）的主轴每转移动量，其指令范围如下表所示。

(a) E60

输入指令单位体系	公制输入		英制输入	
	B (0.001mm)		B (.0025mm)	
指令模式	每分钟进给	每转进给	每分钟进给	每转进给
指令地址	F (mm/min)	F (mm/rev)	F (inch/min)	F (inch/rev)
最小指令单位	1 (=1.000) (1. =1.000)	1 (=0.0001) (1. =1.0000)	1 (=0.01) (1. =1.0000)	1 (=0.000001) (1. =1.000000)
指令范围	0.001 ~1000000.000	0.0001 ~999.9999	0.0001 ~39370.0787	0.000001 ~99.999999

(b) E68

公制输入

输入指令单位体系	B (0.001mm)		C (0.0001mm)	
	每分钟进给	每转进给	每分钟进给	每转进给
指令地址	F (mm/min)	F (mm/rev)	F (mm/min)	F (mm/rev)
最小指令单位	1 (=1.000) (1. =1.000)	1 (=0.0001) (1. =1.0000)	1 (=1.0000) (1. =1.0000)	1 (=0.00001) (1. =1.00000)
指令范围	0.001 ~1000000.000	0.0001 ~999.9999	0.0001 ~100000.0000	0.00001 ~99.99999

英制输入

输入指令单位体系	B (0.0001 inch)		C (0.00001 inch)	
	每分钟进给	每转进给	每分钟进给	每转进给
指令地址	F (inch/min)	F (inch/rev)	F (inch/min)	F (inch/rev)
最小指令单位	1 (=0.01) (1. =1.0000)	1 (=0.000001) (1. =1.000000)	1 (=0.01) (1. =1.00000)	1 (=0.0000001) (1. =1.0000000)
指令范围	0.0001 ~39370.0787	0.000001 ~99.999999	0.00001 ~3937.00787	0.0000001 ~9.9999999

(2) 每转进给时的实际速度（实际的机械移动速度）如下式（式1）所示。

$$FC = F \times N \times OVR \cdots \cdots \cdots \text{（式1）}$$

FC : 实际速度 (mm/min、inch/min)

F : 指令速度 (mm/rev、inch/rev)

N : 主轴转速(r/min)

OVR: 切削进给超程

当同时针对多根轴指令通过式1计算出的实际速度时，则作用于该指令的螺旋方向。

(注1) 设定显示装置的“位置显示”画面中的FC，显示的是根据指令速度与主轴转速及切削进给超程换算为每分钟进给速度后的实际速度 (mm/min 或 inch/min)。

(注2) 当上述实际速度超过切削进给钳位速度时，以该钳位速度进行钳位。

(注3) 在执行同步进给时，当主轴转速为零时，发生操作错误“105”。

(注4) 在机床锁定高速处理时，与指令速度、主轴转速无关，E60，变为 240,000mm/min (或 9448inch/min、240,000°/min)，E68，变为 1,000,000mm/min (或 39,370inch/min、1,000,000°/min)

(注5) 在空运转时，变为非同步，以外部设定速度 (mm/min 或 inch/min) 进行运转。

(注6) 通过参数“#1074 I_Sync”的设定，决定在接通电源时或执行 M02、M30 时，是进行非同步进给 (G94)，还是进行同步进给 (G95)。

7. 进给功能

7.5 进给速度的指定与对各控制轴的效果

7.5 进给速度的指定与对各控制轴的效果



功能及目的

前面已经说过，机械中有各种控制轴，这些控制轴被分类为控制直线运动的直线轴，与控制旋转运动的旋转轴。进给速度是用于指定这些轴的变位速度，在控制直线轴与控制旋转轴时，对于刀具移动速度所产生的影响各不相同。

另外，各轴的位移量可针对各轴分别指定对应的量，但是进给速度并不是针对各轴分别指定，而是以同一个数值进行指定，所以，当同时对2根以上的轴进行控制时，必须先理解对各轴的作用方式。

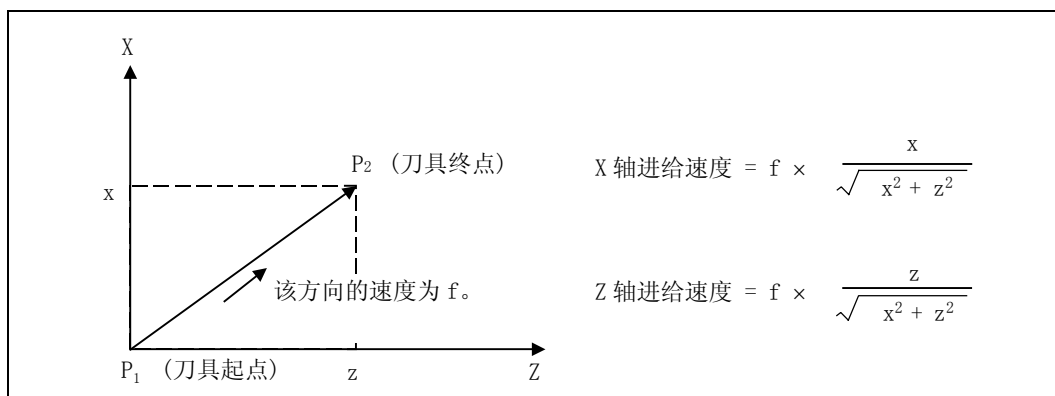
以下对与进给速度指定有关的事项加以说明。



控制直线轴时

不管机械是仅对1轴进行控制，还是同时控制2根以上的控制轴，通过F指定的进给速度均是作为刀具行进方向上的线速度起作用。

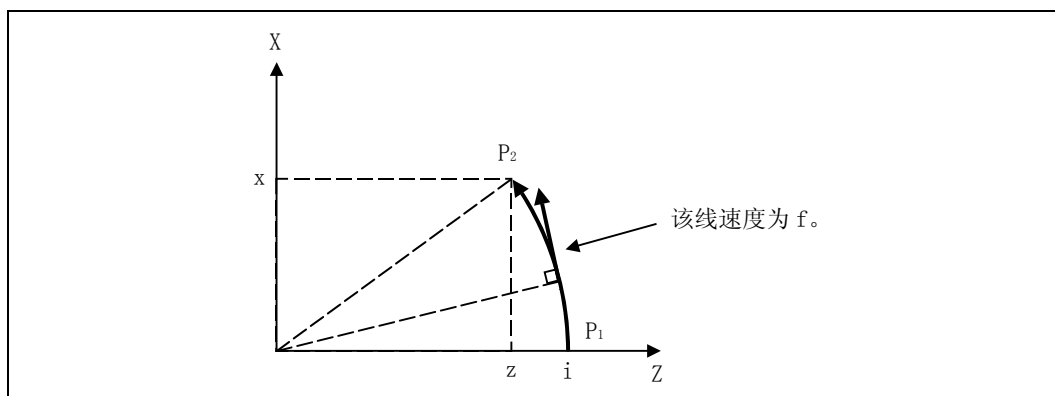
（例）假定进给速度的设定为 f ，对直线轴（X、Z 轴）进行控制时



当仅控制直线轴时，在程序中，只需指定切削速度即可。

将指定的进给速度分解为与移动量相对应的成分之后，即为各轴的进给速度。

（注）利用圆弧插补功能，通过直线控制轴让刀具沿圆周移动时，刀具的前进方向，亦即接线方向的速度，成为程序指定的进给速度。



（例）假定指定的进给速度为 f ，使用圆弧插补功能控制直线轴（X、Z 轴）时

此时，X、Z 各轴的进给速度均随着刀具的移动而发生变化。但是，合成后的速度依然保持为固定的值 f 。

7. 进给功能

7.5 进给速度的指定与对各控制轴的效果



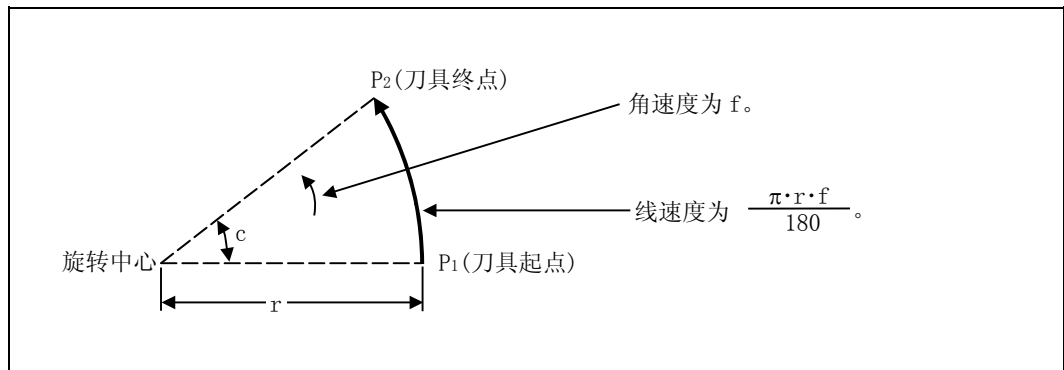
控制旋转轴时

当控制旋转轴时，指定的进给速度作为旋转轴的转速，即作为角速度而起作用。

因此，刀具进行方向的切削速度，即线速度，随着旋转中心与刀具之间的距离变化而变化。

程序中所指定的速度必须对这一距离加以考虑。

（例）假定指定的进给速度为 f ，对旋转轴（C 轴）进行控制时
（ f 的范围为 $^{\circ}/\text{min}$ ）



此时，由于刀具前进方向的切削速度（线速度）为 f_c

$$f_c = f \times \frac{\pi \cdot r}{180}$$

因此，程序中所指定的进给速度，必须如下。

$$f = f_c \times \frac{180}{\pi \cdot r}$$

7. 进给功能

7.5 进给速度的指定与对各控制轴的效果



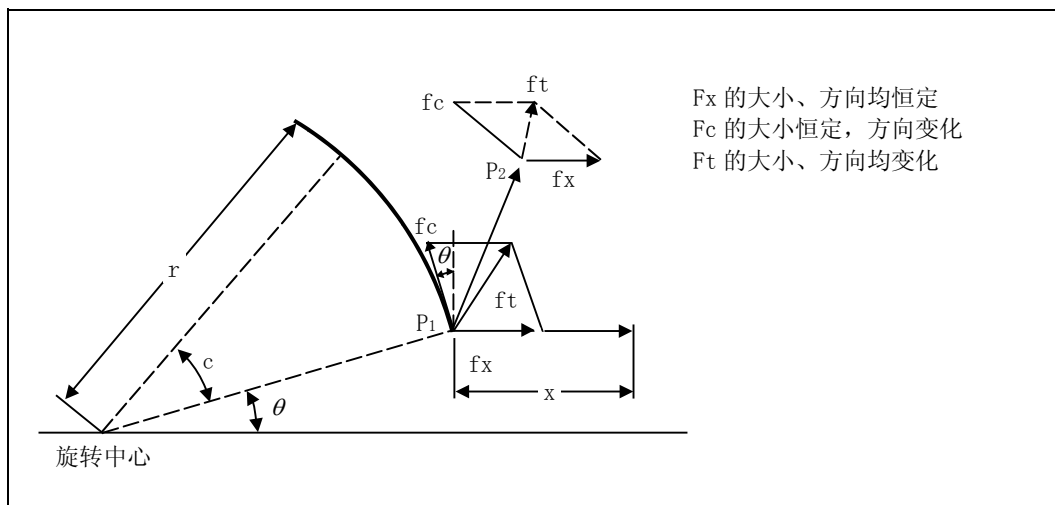
同时进行直线轴控制与旋转轴控制时

无论控制装置是控制直线轴，还是控制旋转轴，均起相同的作用。

当控制旋转轴时，通过坐标字（C、H）所设定的数值为角度，而通过进给速度（F）设定的数值全部作为线速度生效。也就是说，旋转轴的 1° 等价于直线轴的1mm。

因此，当同时对直线轴与旋转轴进行控制时，通过F设定的数值的角轴对应成分，与上述的“控制直线轴时”相同。但是此时，由于在直线轴控制中，速度成分的大小、方向均不会发生变化，而在旋转轴控制中，速度成分的方向会随着刀具的移动而发生变化（大小不变），所以，其结果就是，合成后的刀具前进方向上的进给速度，会随着刀具的移动而发生变化。

（例）假设指定的进给速度为F，同时对直线轴（X轴）与旋转轴（C轴）进行控制时
假定X轴增量指令值为 x ，C轴增量指令值为 c ，则



7. 进给功能

7.5 进给速度的指定与对各控制轴的效果

X轴的进给速度（线速度） f_x 及C轴的进给速度（角速度） ω 为

$$f_x = f \times \frac{x}{\sqrt{x^2 + c^2}} \quad \dots \dots \textcircled{1}$$

$$\omega = f \times \frac{c}{\sqrt{x^2 + c^2}} \quad \dots \dots \textcircled{2}$$

C轴控制中的线速度 f_c 为

$$f_c = \omega \times \frac{\pi \times r}{180} \quad \dots \dots \textcircled{3}$$

假设起点 P_1 上刀具前进方向的速度为 f_t ，其在X轴及Y轴方向上的分速度分别为 f_{tx} 、 f_{ty} ，则

$$f_{tx} = -r \sin \left(-\frac{\pi}{180} \theta \right) \times \frac{\pi}{180} \omega + f_x \quad \dots \dots \textcircled{4}$$

$$f_{ty} = -r \cos \left(-\frac{\pi}{180} \theta \right) \times \frac{\pi}{180} \omega \quad \dots \dots \textcircled{5}$$

在这里， r 为旋转中心与刀具间的距离（单位 mm）

θ 为在旋转中心， P_1 点与X轴间的角度（单位°）

根据①、②、③、④、⑤式，合成速度 f_t 为

$$\begin{aligned} f_t &= \sqrt{f_{tx}^2 + f_{ty}^2} \\ &= f \times \frac{\sqrt{x^2 - x \times c \times r \sin \left(-\frac{\pi}{180} \theta \right) \frac{\pi}{90} + \left(\frac{\pi \times r \times c}{180} \right)^2}}{x^2 + c^2} \quad \dots \dots \textcircled{6} \end{aligned}$$

因此，程序中指定的进给速度 f 必须为

$$f = f_t \times \frac{x^2 + c^2}{\sqrt{x^2 - x \times c \times r \sin \left(-\frac{\pi}{180} \theta \right) \frac{\pi}{90} + \left(\frac{\pi \times r \times c}{180} \right)^2}} \quad \dots \dots \textcircled{7}$$

但是，⑥式的 f_t 为在P点上的速度，由于C轴是旋转前进，所以 θ 的值在不断变化， f_t 的值也随之不断变化。因此，为了尽可能保持切削速度 f_t 恒定，1单节中指定的旋转角度必须尽可能小，尽可能的缩小 θ 值的变化幅度。

7.6 螺纹切削模式



功能及目的

对于螺纹切削模式（G33、G34、G76、G78指令），可通过F7位或E8位指定螺纹的导程。

螺纹导程指令范围为0.0001~999.9999mm/rev（F7位），或0.00001~999.99999mm/rev（E8位）。（输入单位为 μm 时）

(1) E60

攻丝 公制输入

英制输入

输入单位体系	B (0.001mm)			B (0.0001inch)		
指令地址	F (mm/rev)	E (mm/rev)	E (圈/inch)	F (inch/rev)	E (inch/rev)	E (圈/inch)
最小指令单位	1 (=0.0001) (1.=1.0000)	1 (=0.00001) (1.=1.00000)	1 (=1.00) (1.=1.00)	1 (=0.000001) (1.=1.000000)	1 (=0.0000001) (1.=1.0000000)	1 (=1.0000) (1.=1.0000)
指令范围	0.0001 ~999.9999	0.00001 ~999.99999	0.03 ~999.99	0.000001 ~99.999999	0.000010 ~9.9999999	0.0101 ~9999.9999

(2) E68

攻丝 公制输入

输入单位体系	B (0.001mm)			C (0.0001mm)		
指令地址	F (mm/rev)	E (mm/rev)	E (圈/inch)	F (mm/rev)	E (mm/rev)	E (圈/inch)
最小指令单位	1 (=0.0001) (1.=1.0000)	1 (=0.0001) (1.=1.00000)	1 (=1) (1.=1.00)	1 (=0.00001) (1.=1.00000)	1 (=0.000001) (1.=1.000000)	1 (=1) (1.=1.000)
指令范围	0.0001 ~999.9999	0.00001 ~999.99999	0.03 ~999.99	0.00001 ~99.99999	0.000001 ~99.999999	0.255 ~9999.999

攻丝 英制输入

输入单位体系	B (.0025mm)			C (0.00001inch)		
指令地址	F (inch/rev)	E (inch/rev)	E (圈/inch)	F (inch/rev)	E (inch/rev)	E (圈/inch)
最小指令单位	1 (=0.000001) (1.=1.000000)	1 (=0.0000001) (1.=1.0000000)	1 (=1.0000) (1.=1.0000)	1 (=0.0000001) (1.=1.0000000)	1 (=0.00000001) (1.=1.00000000)	1 (=1.00000) (1.=1.00000)
指令范围	0.000001 ~99.999999	0.000010 ~9.9999999	0.0101 ~9999.9999	0.0000001 ~9.9999999	0.0000010 ~0.99999999	0.10001 ~999.99999

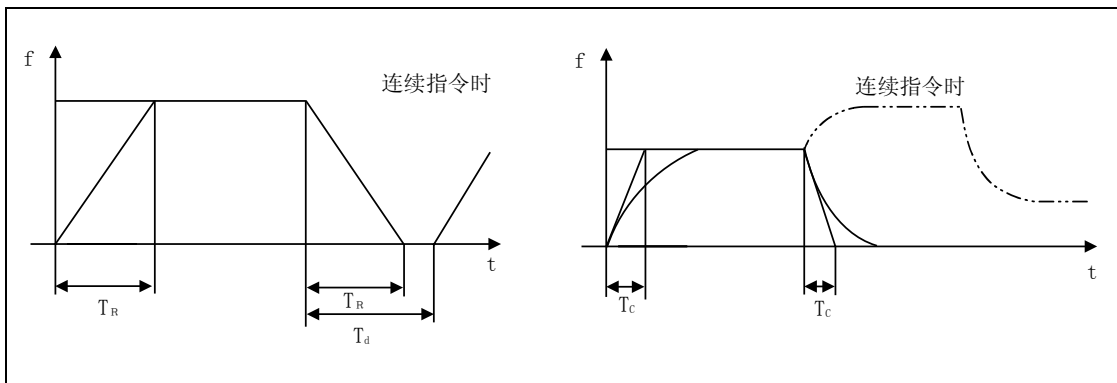
7.7 自动加减速



功能及目的

快速进给及手动进给的加减速曲线为直线加速、直线减速，时间常数 T_R 可通过参数，在1~500ms的范围内，每1ms为一档，对各轴进行独立设定。

快速进给及手动进给的加减速曲线为直线加速、直线减速，时间常数 T_R 可通过参数，在1~500ms的范围内，每1ms为一档，对各轴进行独立设定。（通常，所有轴设定相同的时间常数）



快速进给加减速曲线
(T_R =快速进给时间常数)
(T_d =减速检查时间)

切削进给加减速曲线
(T_c =切削进给时间常数)

在快速进给及手动进给中，当前单节的指令脉冲变为“0”，且加减速电路的追踪误差变为“0”之后，执行下一单节。而在切削进给中，当前单节指令脉冲变为“0”，则立即执行下一单节，当通过外部信号（错误检测）检测到加减速电路的追踪误差变为“0”时，也执行下一单节。当减速检查时的就位检查有效（通过参数“#1193 inpos”选择）时，确认加减速电路的追踪误差变为“0”之后，进一步确认位置偏差量低于参数设定值“#2224 sv024”，然后执行下一单节。是通过开关进行错误检测，还是通过M功能进行错误检测，因机械而异，因此，请参阅机械制造商刊行的说明书。

7.8 速度钳位



功能及目的

在速度控制中，应确保在切削进给速度指令的基础上做了补偿之后的实际切削进给速度，小于预先针对各轴独立设定的速度钳位值。

（注）同步进给、攻丝中，无需进行速度钳位。

7.9 精确停止检查：G09



功能及目的

当刀具的进给速度急剧变化时，为了缓和机械碰撞，及防止转角切削时的圆角等，可能会希望在机械减速停止之后，确认就位状态或经过减速检查时间之后，再开始下一单节的指令。精确停止检查就是为了应用于这一场合。

可通过参数（#1193 Inpos）选择是在减速检查时间中进行控制，还是在就位中进行控制。（参阅“7.11 减速检查”）

在伺服参数“#2224 sv024”或“#2077 G0inpos”“#2078 Glinps”中设定就位宽度。



指令格式

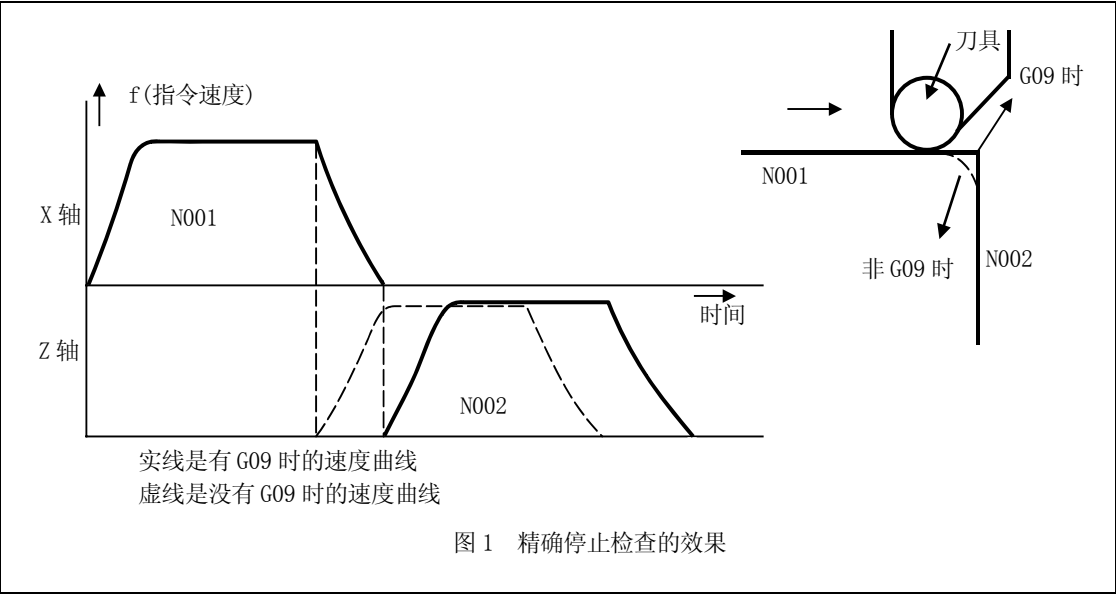
G09 G01 (G02,G03) ;

精确停止检查G09仅对该单节的切削指令（G01～G03）有效。



程序例

N001 G09 G01 X100.000 F150 ;	经过减速检查时间，或在减速停止后，确认就位状态之后，执行下一单节。
N002 Z100.000;	





详细动作

(1) 连续切削进给时

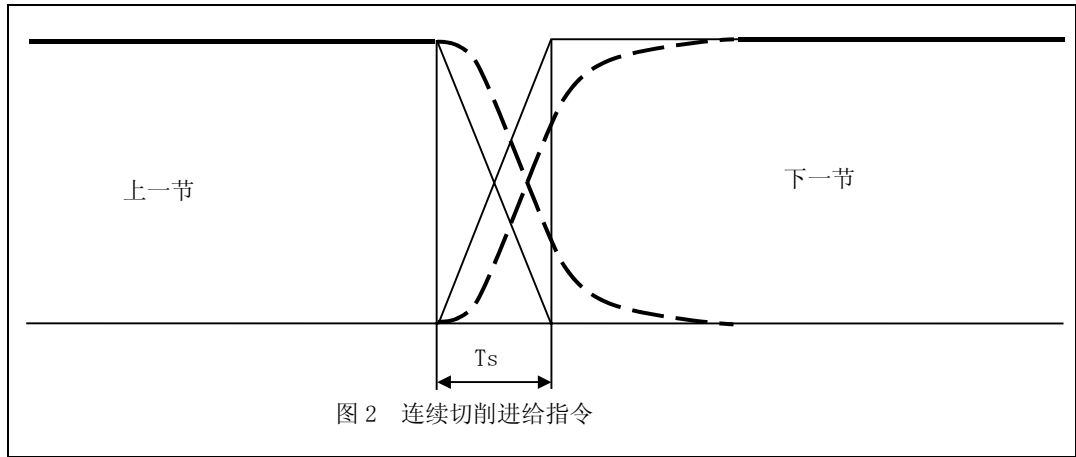


图 2 连续切削进给指令

(2) 切削进给就位检查时

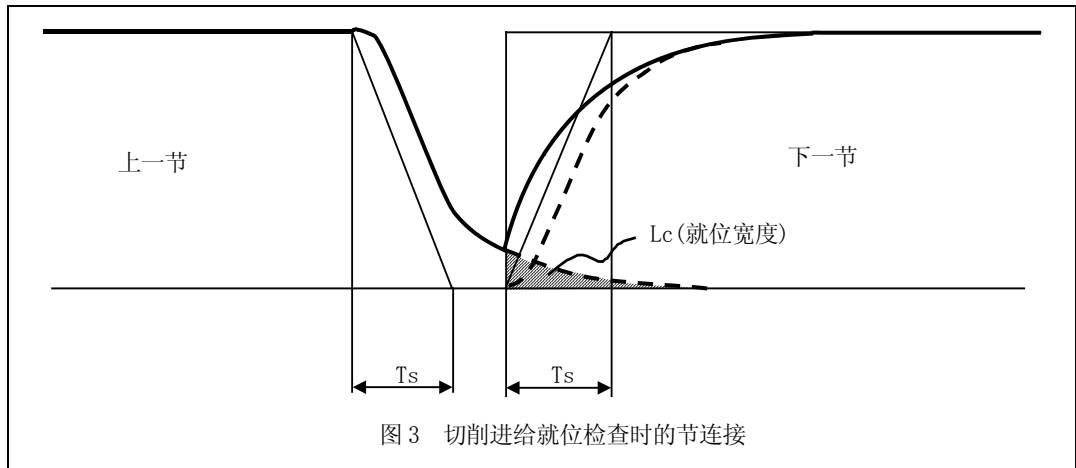


图 3 切削进给就位检查时的节连接

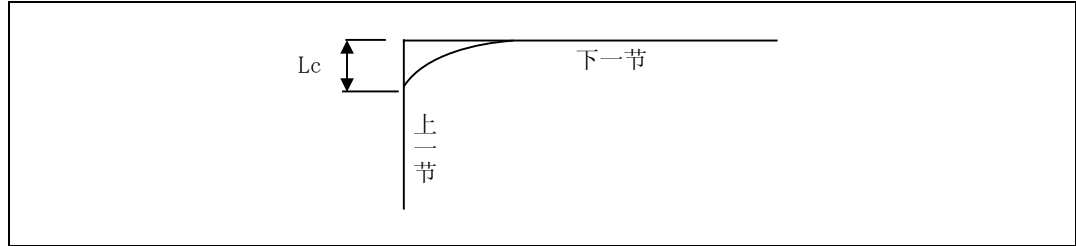
图2、3中

Ts: 切削进给加减速时间常数

Lc: 就位宽度

就位宽度Lc如图3所示，在伺服参数“#2224 sv024”中设定开始下一单节时，当前单节的剩余距离（图3的斜线部分的面积）。

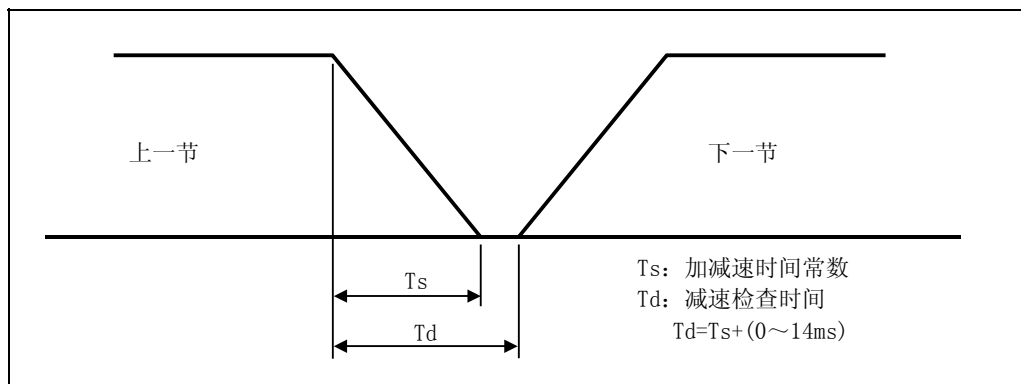
就位宽度是为了将加工对象转角上的圆角控制在一定值以下。



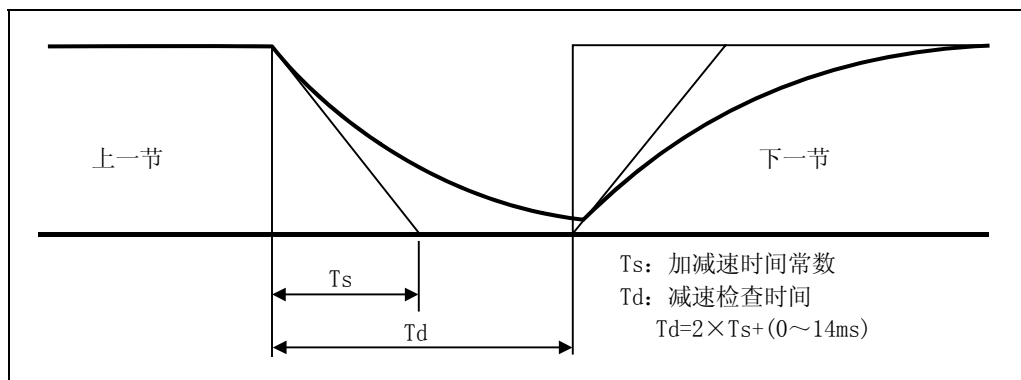
当想消除转角的圆角时，将伺服参数“#2224 sv024”设定为尽可能小的值，进行就位检查，或是在单节间指定延时（G04）指令。

(3) 进行减速检查时

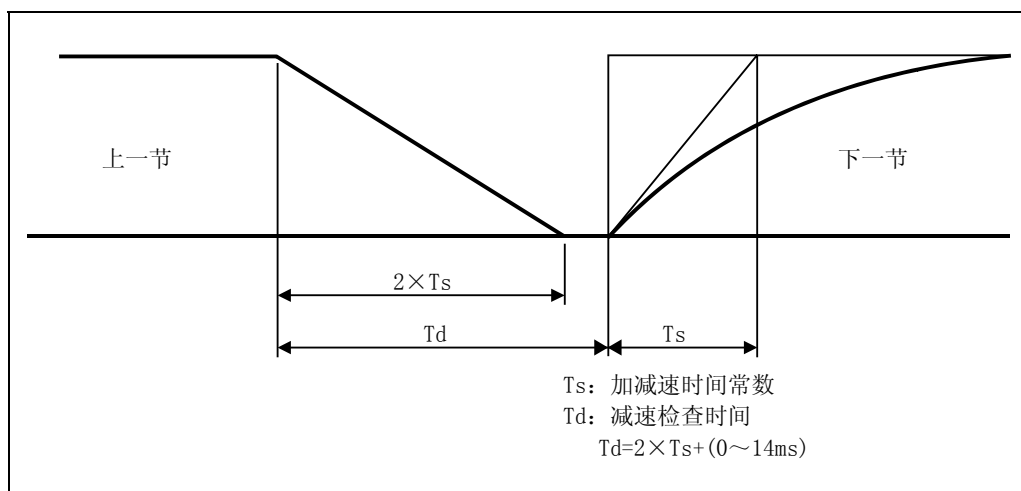
(a) 直线加减速时



(b) 指数加减速时



(c) 指数加速·直线减速时



切削进给时的减速检查所需时间，取决于同时发出指令的轴的切削进给加减速模式及切削进给加减速时间常数所决定的各轴切削进给减速检查时间中，最长的一个。

7.10 精确停止检查模式：G61



功能及目的

通过G09进行精确停止检查，则仅对该指令所在单节进行就位状态的确认，而G61则起模态的功能。因此，G61之后的切削指令（G01～G03）中，全部是在各单节的终点进行减速，进行就位状态的检查。G61可通过自动转角超程（G62）、螺纹切削模式（G63）或切削模式（G64）被解除。



指令格式

G61 ;

在G61的单节中进行就位检查，然后，在检查模式被解除之前，在切削指令单节的终点进行就位检查。

7.11 减速检查



功能及目的

减速检查是决定轴移动单节的移动完成检查方法的功能。
减速检查分为就位检查与指令速度检查两种。
可选择G0、G1的减速检查方式的组合。（请参阅“减速检查的组合”。）
可通过参数的变更，变更G1→G0及G1→G1的反方向时的减速检查。

(1) 减速检查的种类

指令速度检查 在指令速度检查中，通过对电机的指令完成，判定减速完成。	
就位检查 在就位检查中，当电机移动到参数中所指定的就位宽度时，判定减速完成。	

(2) 减速检查的指定

减速检查的参数指定中,包括“减速检查指定类型1”与“减速检查指定类型2”,通过参数“#1306 InpsTyp”进行切换。

(a) 减速检查指定类型1 (“#1306 InpsTyp”=0)

可根据基本规格参数的减速检查方式1 (inpos) 及减速检查方式2 (AUX07/BIT-1), 选择G0/G1等减速检查方式。

参数	快速进给指令	参数	快速进给以外的指令 (G1: G0 以外的指令)	
Inpos (#1193)	G0→XX (G0+G9→XX)	AUX07/BIT-1 (#1223/BIT-1)	G1+G9→XX	G1→XX
0	指令减速检查	0	指令减速检查	无减速检查
1	就位检查	1	就位检查	

(注) XX 表示所有的指令。

(b) 减速检查指定类型2 (“#1306 InpsTyp”=1)

通过 “inpos” 参数指定快速进给与切削进给的就位。

参数	指令单节		
#1193 inpos	G0	G1+G9	G1
0	指令减速检查	指令减速检查	无减速检查
1	就位检查	就位检查	无减速检查

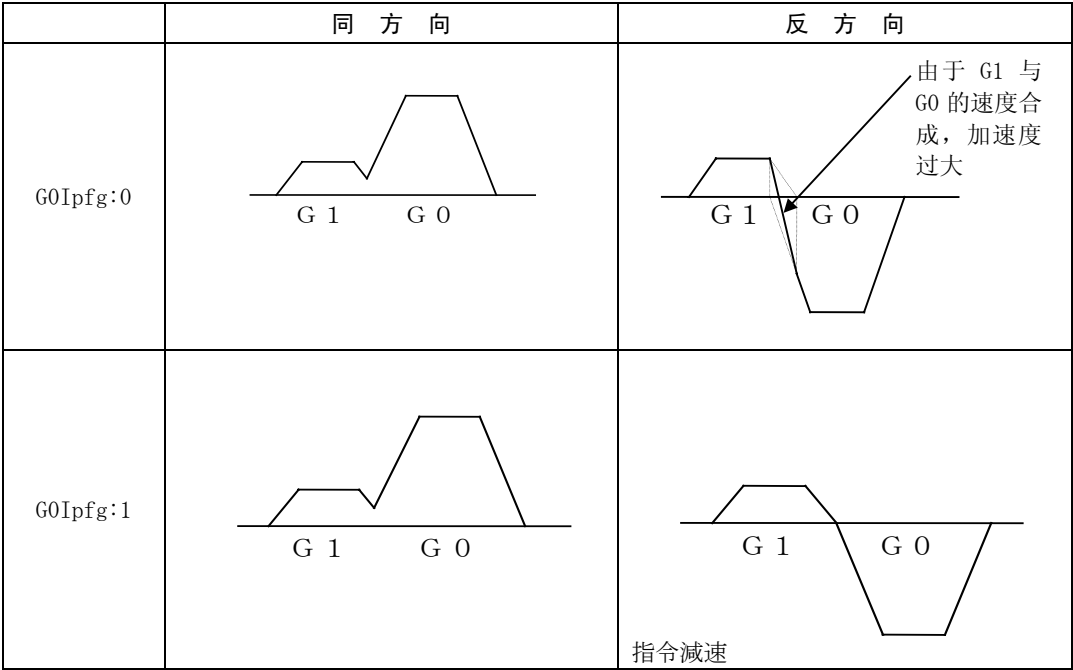
(注) “G0” 表示快速进给, “G1” 表示切削进给。

7.11.1 G1→G0减速检查



详细动作

(1) 在G1→G0连续单节中，可通过变更参数“#1502 G0Ipfg”，变更反方向时的减速检查。



程序例

在多根轴移动中，有减速检查时

(1)	G91 G1 X100. Z100. F4000 ; G0 X-100. Z120. ;	由于 X 轴向反方向移动，所以进行减速检查。
(2)	G91 G1 X100. Z-100. F4000 ; G0 X80. Z100. ;	由于 Z 轴向反方向移动，所以进行减速检查。
(3)	G90 G1 X100. Z100. F4000 ; G0 X80. Z120. ;	由于 X 轴向反方向移动，所以进行减速检查。 (程序开始位置为 X0、Z0 时)
(4)	G91 G1 X100. Z100. F4000 ; G0 X100. Z100. ;	由于 X 轴及 Z 轴均向同一方向移动，所以不进行减速检查。
(5)	G91 G1 X100. Z80. F4000 ; G0 X80. ;	X 轴向同一方向移动，且没有 Z 轴指令时，不进行减速检查。

7.11.2 G1→G1减速检查



详细动作

(1) 在G1→G1连续单节中，可通过变更参数“#1503 G1lpg”，变更反方向时的减速检查。

	同 方 向	反 方 向
G1lpg:0		
G1lpg:1		



程序例

在多根轴移动中，有减速检查时

(1)	G91 G1 X100. Z100. F4000 ; G1 X-100. Z120. ;	由于 X 轴向反方向移动，所以进行减速检查。
(2)	G91 G1 X100. Z-100. F4000 ; G1 X80. Z100. ;	由于 Z 轴向反方向移动，所以进行减速检查。
(3)	G90 G1 X100. Z100. F4000 ; G1 X80. Z120. ;	由于 X 轴向反方向移动，所以进行减速检查。 (程序开始位置为 X0、Z0 时)
(4)	G91 G1 X100. Z100. F4000 ; G1 X100. Z100. ;	由于 X 轴及 Z 轴均向同一方向移动，所以不进行减速检查。
(5)	G91 G1 X100. Z80. F4000 ; G1 X80. ;	X 轴向同一方向移动，且没有 Z 轴指令时，不进行减速检查。

7.12 自动转角超程 ; G62



功能及目的

刀具直径补偿中的切削操作中，为了减轻切削内侧转角时或是自动切削圆角内侧时的负载，自动对进给速度进行超程的指令。

在指定刀具直径补偿取消（G40）、精确停止检查模式（G61）、螺纹切削模式（G63）或切削模式（G64）之前，自动转角超程一直生效。



指令格式

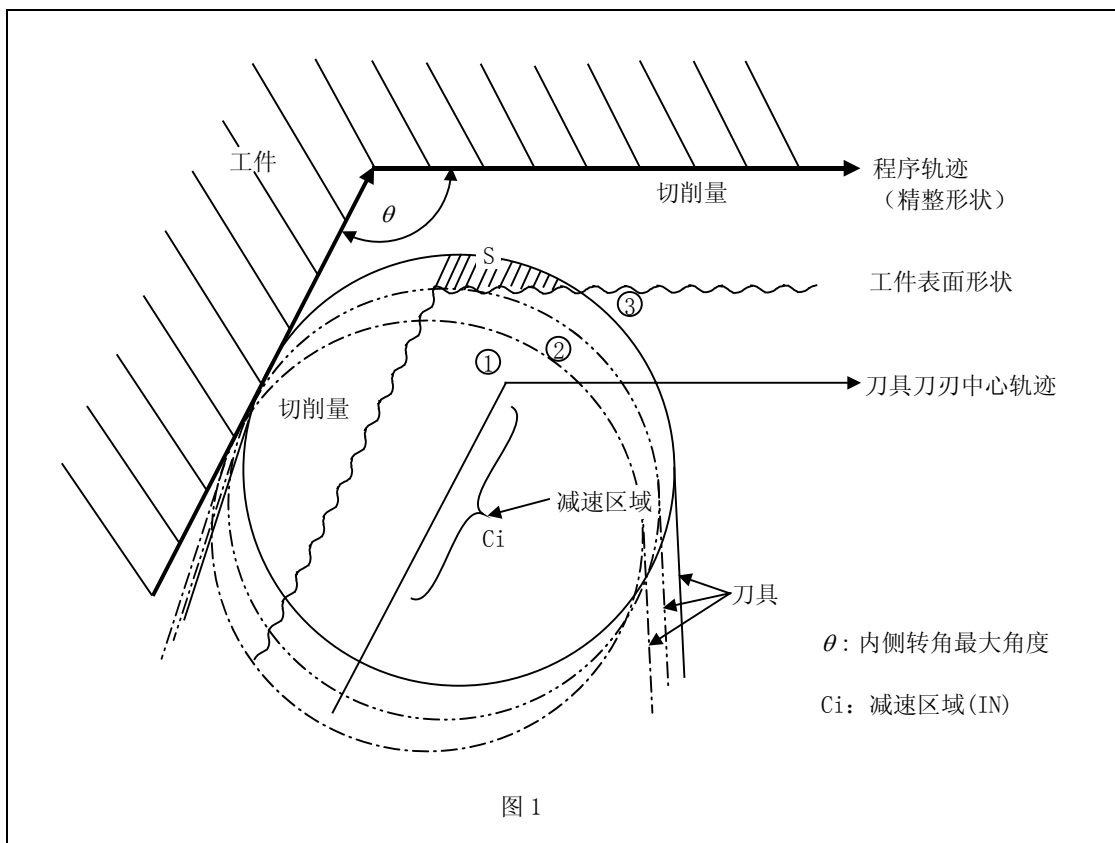
G62 ;



内侧转角时

当如图1所示切削内侧转角时，切削量增加，则刀具的负载增加。因此，在转角的设定范围内自动进行超程，降低进给速度，抑制负载的增加，进行良好的切削。

但是，仅当对精整形状进行了编程时有效。



(1) 动作

(a) 不进行自动转角超程时

图1中，当刀具按①→②→③的顺序移动时，由于③比②多了斜线S面积的切除量，所以刀具的负载增大。

(b) 进行自动转角超程时

图1中，内侧转角的角度 θ 小于参数中设定的角度时，在减速区域Ci内，自动进行参数中所设定的超程。

(2) 参数设定

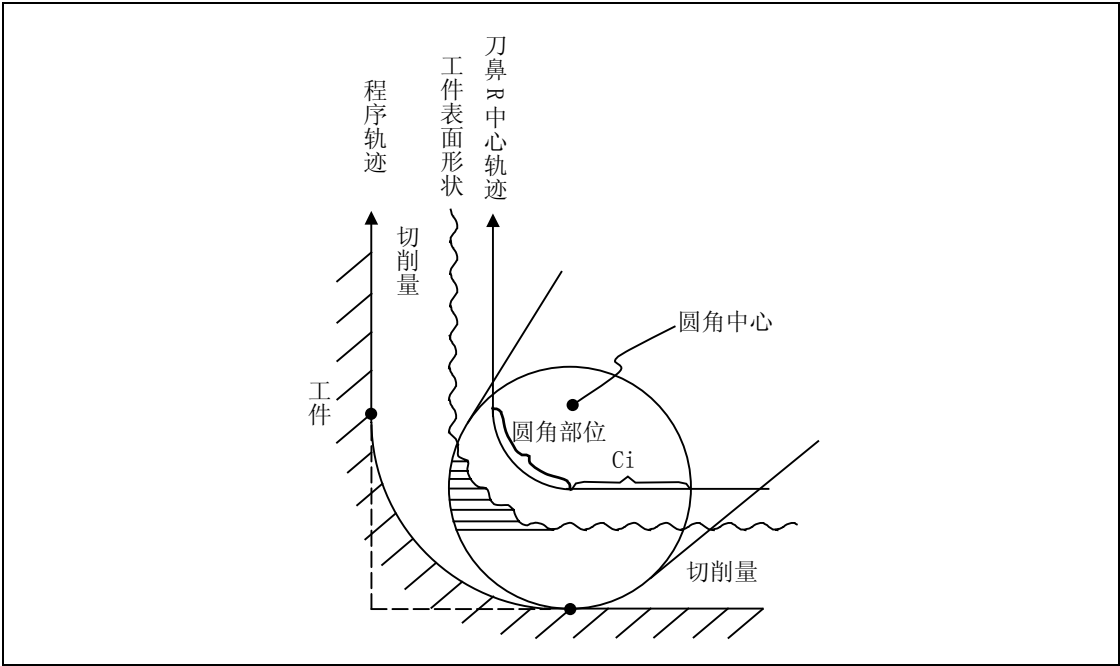
在加工参数中设定如下的参数。

#	参数	设定范围
#8007	超程	0~100 (%)
#8008	内侧转角的最大角度 θ	0~180 (°)
#8009	减速区域 Ci	0~99999.99 (mm) 或 0~3937.000 (inch)

设定方法请参阅操作说明书。



自动圆角时

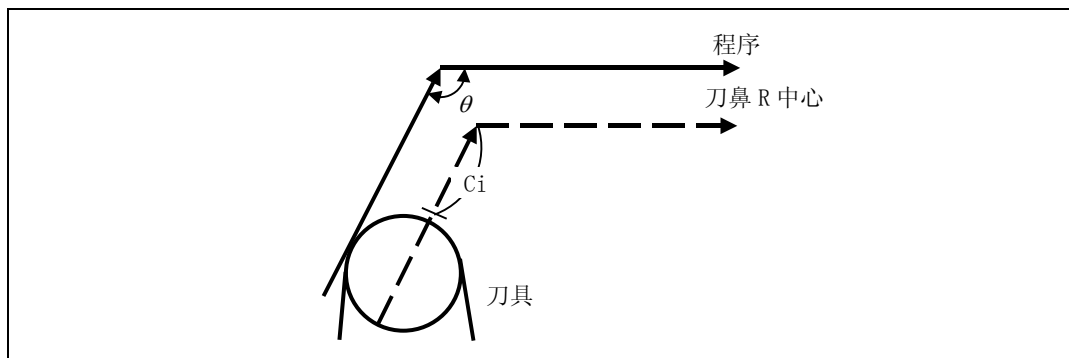


(1) 在自动转角R进行内侧修正时，在减速区域Ci与转角R部位自动进行参数中所设定的超程。（不进行角度的检查。）



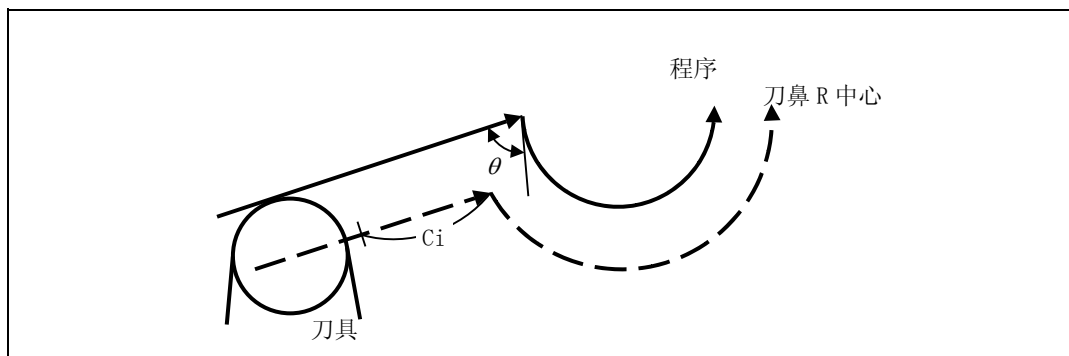
执行例

(1) 直线 - 直线转角



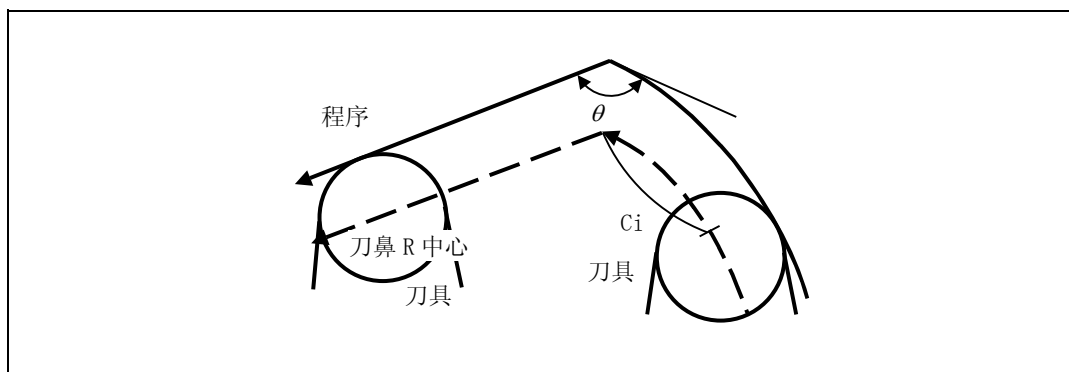
在Ci进行参数中所设定的超程。

(2) 直线 - 圆弧（外侧补偿）转角



在Ci进行参数中所设定的超程。

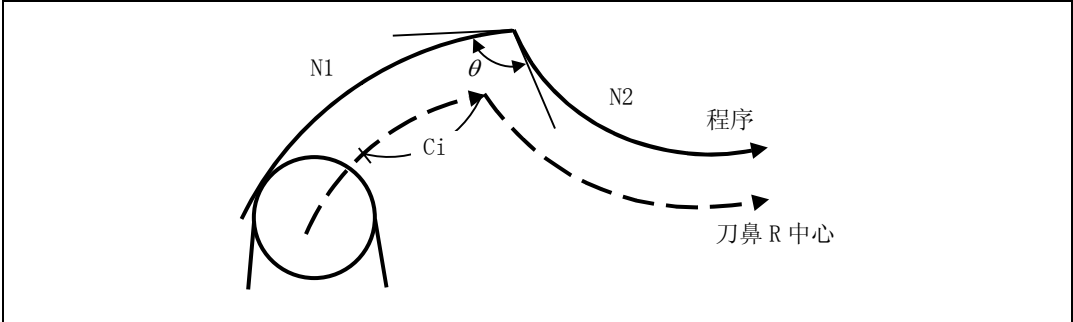
(3) 圆弧（内侧补偿） - 直线转角



在Ci进行参数中所设定的超程。

(注) 进行超程的减速区域Ci, 在执行圆弧指令时, 变为圆弧的长度。

(4) 圆弧（内侧补偿）－圆弧（外侧补偿）转角



在Ci进行参数中所设定的超程。



与其他功能的关联

功 能	转角上的超程
切削进给超程	在切削进给超程之后，进行自动转角超程。
超程取消	在超程取消中，自动转角超程被取消。
速度钳位	有效（自动转角超程后）
空运转	自动转角超程无效
同步进给	对同步进给的速度进行自动转角超程
攻丝	自动转角超程无效
G31 忽略	刀鼻 R 补偿中的 G31 为程序错误
机床锁定	有效
机床锁定高速	自动转角超程无效
G00	无效
G01	有效
G02, G03	有效



注意事项

- (1) 自动转角超程仅在G01、G02、G03模式下有效，在G00模式下无效。另外，当在转角从G00切换为G01（G02、G03）模式（相反时也相同）时，G00单节不会在该转角进行自动转角超程。
- (2) 即使进入自动转角超程模式，在刀鼻R补偿模式之前，不进行自动转角超程。
- (3) 在开始或取消刀鼻R补偿的转角，不进行自动转角超程。
- (4) 在有刀鼻R补偿的I、K螺旋指令的转角，不进行自动转角超程。
- (5) 当不进行交点运算时，不进行自动转角超程。
以下场合不进行交点运算。
 - 有4次以上不连续的移动指令单节时
- (6) 圆弧指令时的减速区域变为圆弧的长度。
- (7) 在参数设定中，内侧转角的角度为程序轨迹上的角度。
- (8) 将参数的最大角度设定为0或180，则不进行自动转角超程。
- (9) 将参数的超程设定为0或100，则不进行自动转角超程。

7.13 螺纹切削模式：G63



功能及目的

通过G63指令，变为如下的适合于攻丝加工的控制模式。

1. 切削超程100%固定
2. 单节间连接的减速指令无效
3. 回馈等待无效
4. 单节无效
5. 螺纹切削模式中输出信号

G63可通过精确停止检查模式（G61）、自动转角超程（G62）或切削模式（G64）被解除。



指令格式

G63 ;

7.14 切削模式：G64



功能及目的

通过G64指令，能够得到平滑的切削面的切削模式。在本模式中，与精确停止检查模式（G61）相反，在切削进给单节间不进行减速停止，而是连续的执行下一单节。

G64可通过精确停止检查模式（G61）、自动转角超程（G62）或螺纹切削模式（G63）被解除。
起始状态变为该切削模式。



指令格式

G64 ;

8. 延时

可通过G04指令，将下一单节的开始推迟。

8.1 每秒延时；G04



功能及目的

本功能是通过程序指令暂时停止机械的移动，进入等待状态的功能。借此，可将下一单节的开始推迟。可通过输入跳跃信号，取消等待状态。



指令格式

G04 X/U_ ; 或 G04 P_ ;

X、P、U…延时时间

对于E68，除了地址P、X外，可使用U（实际为在#1014 incax中所指定的，与X轴对应的地址）。但是，当#1076 AbsInc为0时无效。

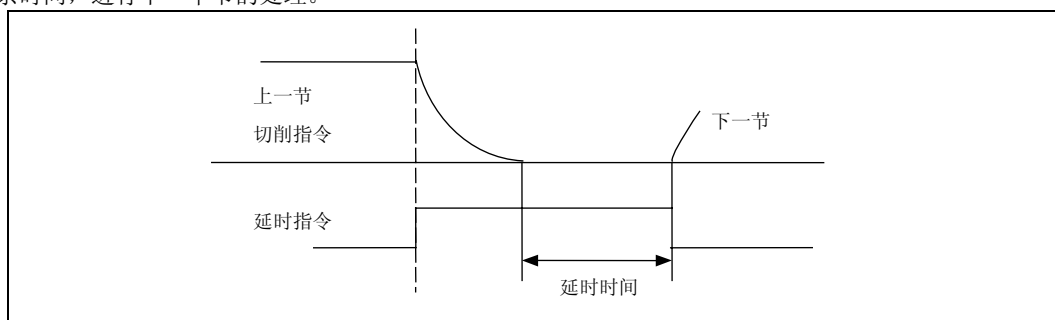


详细说明

- (1) 通过X、U进行延时时间指定时，小数点指令有效。
- (2) 通过P进行延时时间指定时，可通过参数（#8112）进行小数点指令有效・无效的切换。在参数设定中设定小数点指令无效时，通过P跳跃小数点之后的指令。
- (3) 小数点指令有效时・无效时，各延时时间指令范围如下。

小数点指令有效时的指令范围	小数点指令无效时的指令范围
0.001~99999.999 (s)	1~99999999 (ms)

- (4) 通过在参数“#1078 Decpt2”中设定1，可将无小数点时的延时时间设定单位设定为1秒。对X、U、P有效。但是，仅当小数点指令有效时（#8112=1），P可使用。详情请参阅“程序例”。
- (5) 如果在上一单节中有切削指令，则在减速停止完成后，开始延时时间的计算。另外，当在同一单节中指令了M、S、T、B指令时，同时起动。
- (6) 互锁中的延时有效。
- (7) 延时对于机床锁定也有效。
- (8) 可通过预先设定参数#1173 dwlskp，取消延时。当在延时时间中输入了设定的跳跃信号时，则舍弃剩余时间，进行下一单节的处理。



8. 延时

8.1 每秒延时



程序例

(1) 地址X、U

指令	延时时间 [秒]	
	#1078 Decpt2 = 0	#1078 Decpt2 = 1
G04 X500 ;	0.5	500
G04 X5000 ;	5	5000
G04 X5. ;	5	5
G04 X#100 ;	1000	1000
G04 U500 ;	0.5	500
G04 U5000 ;	5	5000
G04 U5. ;	5	5
G04 U#100 ;	1000	1000

(2) 地址P

指令	延时时间 [秒]			
	#1078 Decpt2 = 0		#1078 Decpt2 = 1	
	G04P 小数点无效	G04P 小数点有效	G04P 小数点无效	G04P 小数点有效
G04 P5000 ;	5		5	5000
G04 P12.345 ;	0.012	12.345	0.012	12.345
G04 P#100 ;	1	1000	1	1000

(注1) 上述的例子是以下条件下的结果。

• #100 = 1000 ;

(注2) 根据控制参数 (#8112) 的设定决定“G04P 小数点有效/无效”。

(注3) 输入设定单位为“C” (0.0001mm) 时, G04 前的 X 变为 10 倍。例如, “X5. G04;” 时, 延时时间变为 50 秒。



注意事项・限制事项

(1) 地址U可在E68系统中使用。

(2) 当使用本功能时, 为了明确延时的X、U, 请在G04之后指令X、U。

9. 辅助功能



功能及目的

9.1 辅助功能 (M8位BCD)

辅助功能也称为M功能，是用于指令主轴的正转、倒转、停止、冷却油的开、关等机械的辅助性功能的指令。在本控制装置中，通过在地址M后面加上8位数值(0~99999999)进行指定，在1单节中可指定最多4组指令。

(例) G00 Xx Mm₁ Mm₂ Mm₃ Mm₄ ;

当在1单节内指令了5组以上时，最后4组生效。输出信号为8位BCD代码与起动信号。

M00、M01、M02、M30、M96、M97、M98、M99是8种用于特定目的的辅助指令，无法作为一般性的辅助指令加以设定。因此，可进行92种指定。

具体的数值与功能的对应，请参阅机械制造商刊行的说明书。

另外，关于M00、M01、M02、M30，由于禁止预读处理，所以下一单节不会被读入到预读缓存中。

当在同一单节中同时指定M功能与其他指令时，指令的执行顺序，可能有以下2种状况。适用哪一种，取决于机械规格。

(1) 移动完成后，执行M功能。

(2) 在执行移动指令的同时，执行M功能。

另外，除M96、M97、M98、M99外，对于所有的M指令，需分别处理及完成顺序。

下面对于特殊目的的8种M指令加以说明。



程序停止：M00

读入本辅助功能，则纸带读带机停止读入下一单节。作为NC功能，仅停止纸带的读入，而主轴旋转、钳位等机械端功能是否停止，因机械而异。

通过按机械操作盘的自动起动按钮，进行重新启动。

是否通过M00进行重新启动，取决于机械规格。

9. 辅助功能

9.1 辅助功能（M8 位 BCD）



可选停止：M01

在机械操作盘的可选停止开关打开的状态下，读入M01指令，则纸带读带机停止，起到与上述的M00指令相同的功能。
如果可选停止开关关闭，则M01指令被跳跃。

(例)	<pre> { N10 G00 X1000 ; N11 M01 ; N12 G01 X2000 Z3000 F600 ; }</pre>	可选停止开关指令跳跃。
		打开时 通过 N11 停止 关闭时 不通过 N11 停止，而是执行下一指令（N12）



程序结束：M02 或 M30

该指令通常是用于完成加工的最终单节，主要是用作为纸带的回卷指令。是否进行回卷动作，取决于机械规格。
另外，根据机械规格，在回卷及同一单节中指令的其他指令完成之后，通过M02、M30重新启动。
（但是，通过该指令进行重新启动时，指令位置显示计数器的内容不会被清除，而模态指令、补偿量则被取消。）
回卷完成时（自动运转中指示灯熄灭），以下动作停止，所以，在重新启动时，必须进行按下自动起动按钮等操作。
（注1）M00、M01、M02、M30 也分别单独输出信号，但是通过按下重新启动键，M00、M01、M02、M30 的单独输出重新启动。
（注2）也可通过手动数据输入 MDI 指令 M02、M30。
此时，也与纸带模式相同，可同时执行其他指令。



宏插入：M96, M97

M96、M97是用于用户宏插入控制的M代码。
用户宏插入控制用M代码为内部处理，不会被输出。
当将M96、M97作为辅助功能使用时，请通过参数（#1109 subs_M及#1110 M96_M、#1111 M97_M）变更为其其他的M代码。



子程序调用、结束：M98, M99

作为对子程序的分歧及分歧目标子程序的回归命令使用。
由于M98、M99为内部处理，所以不输出M代码信号与开闭信号。



M00/M01/M02/M30 指令时的内部处理

读入M00、M01、M02、M30时，内部处理中止预读。除此以外的纸带回卷动作及重新启动处理所导致的模态初始状态因机械规格而异。

9.2 第2辅助功能（A8位、B8位或C8位）



功能及目的

是指定分度平台的位置等。在本控制装置中，可通过地址A、B、C后的8位数值，在0~99999999之间任意指定，哪一个代码对应于哪一个位置，则取决于机械规格。
当在同一单节中同时指定A、B、C功能与移动指令时，指令的执行顺序，可能有以下2种状况。适用哪一种，取决于机械规格。

- （1）移动指令完成后，执行A、B、C功能。
- （2）在执行移动指令的同时，执行A、B、C功能。

对于所有的第2辅助功能，需要处理及完成顺序。
地址的组合如下表所示。即，附加轴的轴名称与第2辅助功能不能使用同一地址。

附加轴名称 第2辅助功能	A	B	C
A	×	○	○
B	○	×	○
C	○	○	×

（注）当在第2辅助功能地址中指定了A时，以下的功能无法使用。

- （1）直线角度指令
- （2）几何指令
- （3）深孔钻孔循环2指令

9.3 转台分度

[E68]



功能及目的

可通过设定分度轴，进行转台分度。
分度指令只是在转台分度设定轴上指令分度角度。无需指令用于平台的钳位・解除钳位的特殊M代码，确保易于进行编程。



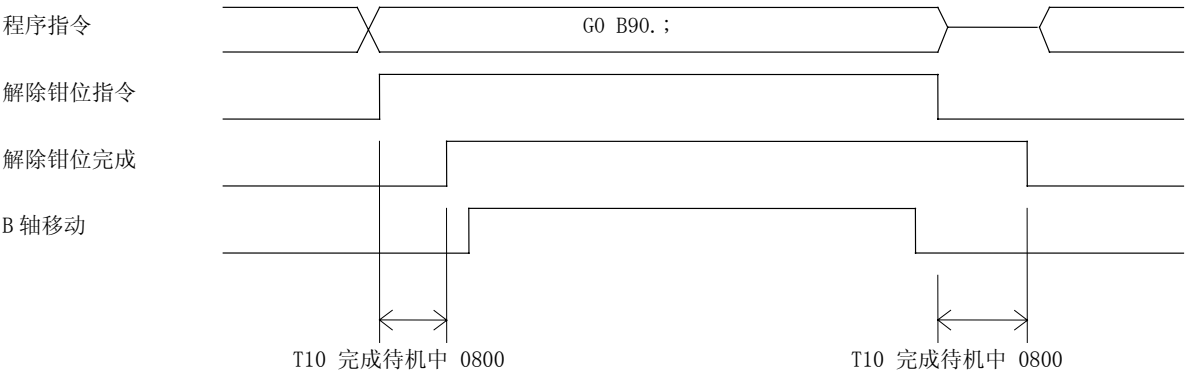
详细说明

转台分度功能进行如下的动作。
指令格式例

(例) G00 B90;
索引平台分度轴（参数#2076 index_x）中指定的轴

- (1) 将进行转台分度的轴的“分度轴选择”参数（#2076）设定为“1”。
- (2) 根据程序指令，执行所选轴的移动指令（绝对/增量均可）。
- (3) 在轴移动之前，进行解除钳位动作。
- (4) 解除钳位完成后，开始所指定轴的移动。
- (5) 移动完成后，进行钳位动作。
- (6) 解除钳位完成后，进行下一单节的处理。

＜动作时间表＞





注意事项

- (1) 可对多根轴进行转台分度轴的设定。
- (2) 转台分度轴的移动速度，取决于当时的模态（G0/G1）进给速度。
- (3) 当在同一单节中指令了转台分度轴与其他轴时，也进行转台分度轴的解除钳位指令。因此，在解除钳位动作完成前，也不执行同一单节中指定的其他轴的移动。
但是，执行非插补指令时，执行同一单节中指定的其他轴移动。
- (4) 转台分度轴是用于通常旋转轴，但是对于直线轴，本功能也会进行解除钳位动作。
- (5) 自动运转中，转台分度轴移动时，若发生导致解除钳位指令关闭的原因，则在解除钳位状态下，转台分度轴减速停止。
另外，同一单节中指定的其他轴，除非插补指令外，也同样减速停止。
- (6) 在转台分度轴的移动中，因为互锁等而导致轴移动中断时，保持解除钳位状态。
- (7) 当连续有转台分度轴的移动指令时，不进行钳位・解除钳位动作。
但是，单节运转时，即使有连续的移动指令，也执行钳位・解除钳位动作。
- (8) 请注意不要指定为无法进行钳位的位置。

10. 主轴功能

10.1 主轴功能 (S功能)



功能及目的

主轴功能也称为S功能，是用于指定主轴转速。当在同一节中同时指定S功能与移动指令时，指令的执行顺序，可能有以下2种状况。适用哪一种，取决于机械规格。

- (1) 移动完成后，执行S功能。
- (2) 在执行移动指令的同时，执行S功能。

10.1.1 主轴功能 (S6位模拟)



功能及目的

在S6位功能中，可在S代码后指定6位数值（0~999999）。

另外，此时请务必选择S指令二进制输出。

在本功能中，通过S代码后的6位数值指令，输出适当的齿轮信号、与指定的主轴转速相对应的电压及起动信号。

对于所有的S指令，需要处理及完成顺序。

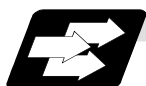
模拟信号的规格如下。

- (1) 输出电压……………0到10V
- (2) 分辨率……………1/4096 (2的-12次方)
- (3) 负载条件……………10K Ω
- (4) 输出阻抗……………220 Ω

预先设定最多4级各种参数设定，选择与S指令对应的齿轮级数，输出齿轮信号。根据输入齿轮信号进行模拟电压的计算。

- (1) 与各齿轮对应的参数……………极限转速、最高转速、移位转速、螺纹切削转速
- (2) 与所有齿轮对应的参数……………定位转速、最低转速

10.1.2 主轴功能 (S8位)



功能及目的

通过在地址S后面加上8位数值（0~99999999）进行指定，在1节中可指定最多1组指令。

输出信号是以带符号的32bit二进制数据作为起动信号。

对于所有的S指令，需要处理及完成顺序。

10.2 线速度恒定控制；G96、G97



功能及目的

对于直径方向的切削，随着坐标值的变化，自动对主轴转速进行控制，确保切削点的速度恒定，进行切削加工。



指令格式

G96 Ss Pp ; 线速度恒定控制 ON

Ss : 线速度

Pp : 线速度恒定控制轴

G97 ; 取消线速度恒定



详细说明

- (1) 通过参数（#1181 G96_ax）设定线速度恒定控制轴。
- 0: 第1轴固定（P指令无效）
 - 1: 第1轴
 - 2: 第2轴
 - 3: 第3轴
- (2) 当上述的参数不为0时，可通过地址P指定线速度恒定控制。
- （例）G96_ax为1时

程序	线速度恒定控制轴
G96 S100 ;	第 1 轴
G96 S100 P3 ;	第 3 轴

(3) 切换程序与动作例

G90 G96 G01 X50. Z100. S200 ;
{
}

G97 G01 X50. Z100. F300 S500 ;
{
}

M02;

控制主轴转速，确保线速度为 200m/min。

进行控制，确保主轴的转速为 500r/min。

模态返回初始值。

10.3 主轴钳位速度设定：G92



功能及目的

可通过G92后的地址S指定主轴的最高钳位转速，通过地址Q指定主轴的最低钳位转速。



指令格式

G92 Ss Qq ;	
Ss	: 最高钳位转速
Qq	: 最低钳位转速

与主轴及主轴电机间的齿轮切换相对应，可通过参数，以1r/min为单位，设定最多4级转速范围。通过该参数设定的转速范围，与通过“G92 Ss Qq ; ”设定的转速范围中，上限较低者与下限较高者有效。

通过参数（#1146 Sclamp、#1227 aux11/bit5）选择是仅在线速度恒定模式中进行转速钳位，还是在线速度恒定取消时也进行转速钳位。

（注）G92S 指令时及转速钳位动作

		Sclamp=0		Sclamp=1	
		aux11/bit5=0	aux11/bit5=1	aux11/bit5=0	aux11/bit5=1
指令	G96 中	转速钳位指令		转速钳位指令	转速钳位指令
	G97 中	主轴转速指令		转速钳位指令	转速钳位指令
动作	G96 中	执行转速钳位		执行转速钳位	执行转速钳位
	G97 中	无转速钳位		执行转速钳位	无转速钳位

10.4 主轴・C轴控制

[E68]



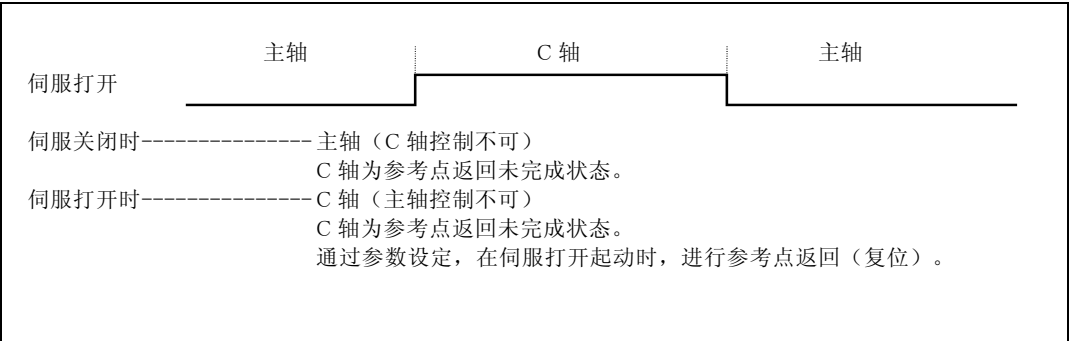
功能及目的

本功能是通过外部信号将1台主轴作为C轴（旋转轴）使用的功能。

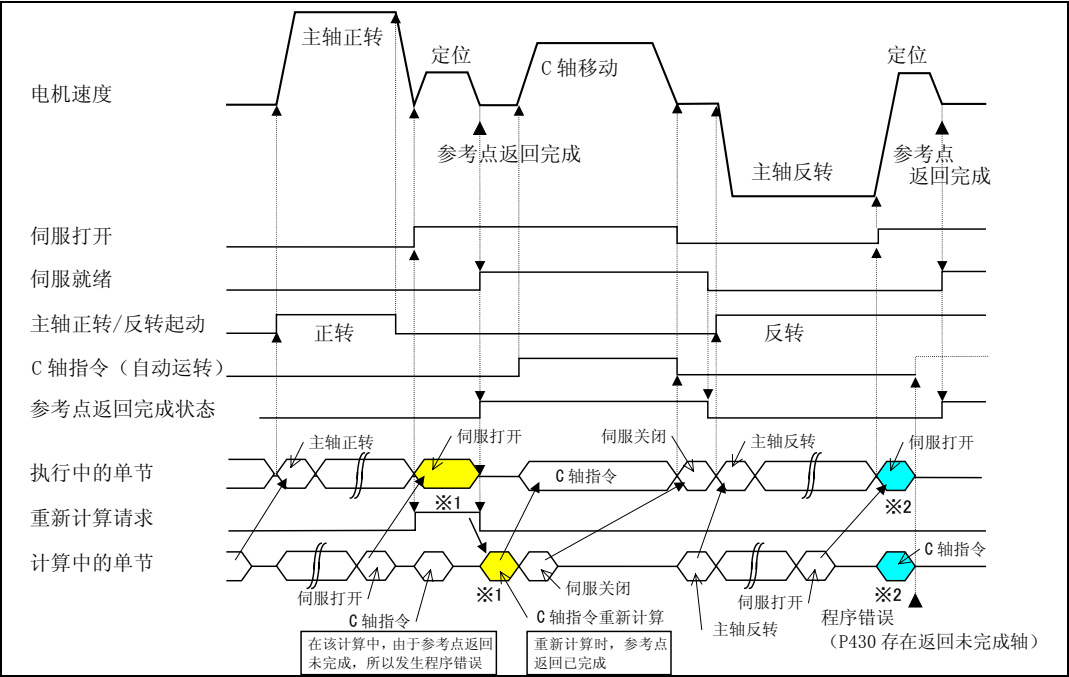


详细说明

(1) 主轴/C轴切换
通过C轴的伺服起动信号切换。



(2) 切换时间表例



(注1) 由于轴指令是在计算时进行参考点返回完成检查，所以当连续执行C轴伺服起动指令与C轴指令时，如※2所示，发生程序错误（P430）。

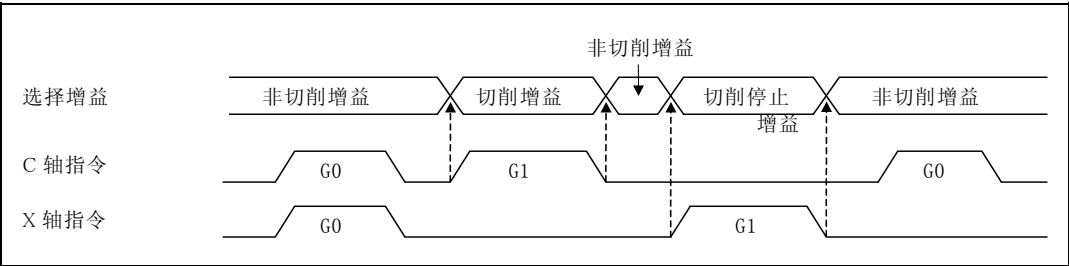
为了应对这样的场合，必须如※1所示，通过用户PLC进行以下2项处理。

- 通过伺服起动指令输入重新计算请求信号。
- 在C轴进入伺服就绪状态之前，等待伺服起动指令。

(3) C轴增益

根据C轴的切削状况，进行C轴的增益切换（选择最佳增益）。C轴切削进给时选择切削增益，其他轴切削进给（C轴面切削）时选择切削停止增益，其他情况下选择非切削增益。

关于增益的调整，请参阅所使用的C轴参数。



(注1) 切削增益包括第1到第3，通过梯形图进行选择。

(4) 包括主轴/C轴在内的移动减速检查

包含了在非切削时的位置回路增益（主轴参数#3203 PGC0）与切削时的位置回路增益（主轴参数#3330 PGC1～#3333 PGC4）中设定了不同值的轴移动指令时，主轴/C轴的减速检查，如下所示。这是因为，如果在轴移动中变更增益，则机械会发生振动等。

参数	快速进给指令	参数	快速进给以外的指令 (G1: G0 以外的指令)	
Inpos (#1193)	G0→XX (G0+G9→XX)	AUX07/BIT-1 (#1223/BIT-1)	G1→G0 (G1+G9→XX)	G1→G1
0	指令减速检查	0	就位检查	无减速检查
1	就位检查	1	(仅 SV024 适用)	

(注 1) G1 进给时，不管减速参数如何，都进行就位检查。

(注 2) XX 表示所有的指令。

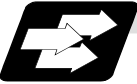


注意事项・限制事项

- (1) 编码器（PLG、ENC、其他）上没有Z相时，无法通过定位进行参考点返回。
当更换为带Z相时，或是直接使用时，请将位置控制切换类型设定为减速停止后（主轴参数“#3329 SP129/bitE”=1），作为无原点轴（原点返回参数“#2031 noref”=1）使用。
- (2) 在伺服关闭中或定位中执行C轴指令，则发生程序错误（P430）。
- (3) 请不要在执行C轴指令时进行伺服关闭。
C轴指令的剩余指令会在伺服起动时被清除。
（在C轴控制中关闭伺服，则进给停止，变为主轴控制。）
- (4) 在主轴旋转中伺服起动，则旋转停止，变为C轴控制。
- (5) 无法进行C轴的挡块式参考点返回。
请使用定位式（主轴参数“#3329 SP129/bitE”=0），或是作为无原点轴（原点返回参数“#2031 noref”=1）使用。

10.5 主轴同期控制

[E68]



功能及目的

对于具有2台主轴的机械，与主轴（基准主轴）的旋转同步，对另一主轴（同期主轴）的转速及相位进行控制。

当必须让2台主轴的转速一致时，例如，将由第1主轴夹持的工件转由第2主轴夹持时，或是由第1主轴、第2主轴2根主轴夹持同一个工件的状态下，为了变更主轴的转速而使用本功能。

主轴同期控制包括主轴同期控制 I 与主轴同期控制 II。

主轴同期控制 I	通过加工程序内的G指令，进行同期主轴的指定及同步的开始/结束。
主轴同期控制 II	同期主轴的选择及同步开始等全部通过PLC指定。详情请参阅机械制造商发行的说明书。



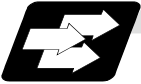
主轴同期控制 I、II 通用设定

进行主轴同期控制时，必须进行以下的设定。

- 夹头闭合
- 误差临时取消
- 相位监视
- 多级加减速

详情请参阅“10.5.3 使用主轴同期控制时的注意事项”。

10.5.1 主轴同期控制 I ； G114.1



功能及目的

主轴同期控制 I 是通过加工程序内的G指令，进行同期主轴的指定及同步的开始/结束。



指令格式

(1) 主轴同期控制起动 (G114.1)

该指令进行基准主轴与同期主轴的指定，使所指定的2台主轴处于同步状态。此外，可通过指令同期主轴相位移动量，进行基准主轴与同期主轴的相位匹配。

G114.1 H_ D_ R_ A_ ；	
H_	： 基准主轴选择
D_	： 同期主轴选择
R_	： 同期主轴相位移动量
A_	： 主轴同期加减速时间常数

(2) 主轴同期控制取消 (G113)

该指令解除因主轴同期指令而同步旋转的2台主轴的同步状态。

G113 ；	
---------------	--

地址	地址的含义	指令范围（单位）	备 注
H	基准主轴选择 指令进行同步的 2 台主轴中，作为基准的主轴编号。	1~2 1： 第 1 主轴 2： 第 2 主轴	• 如果指令了超出指令范围的数值或规格中不存在的主轴编号，则发生程序错误（P35）。 • 当无指令时，发生程序错误（P33）。 • 指令未进行串行连接的主轴，则发生程序错误（P700）。
D	同期主轴选择 指令进行同步的 2 台主轴中，与基准主轴进行同步的主轴编号。	1~2 或 -1~-2 1： 第 1 主轴 2： 第 2 主轴	• 如果指令了超出指令范围的数值，则发生程序错误（P35）。 • 当无指令时，发生程序错误（P33）。 • 当指定了基本主轴选择中所指定的主轴时，也发生程序错误（P33）。 • 通过 D 的符号，指令同期主轴相对于基准主轴的旋转方向。 • 指令未进行串行连接的主轴，则发生程序错误（P700）。

地址	地址的含义	指令范围（单位）	备 注
R	同期主轴相位移位量 指定距同期主轴的参考点（1 转信号）的移位量。	0~359.999 (°) 或、 0~359999 (° x10 ⁻³)	• 如果指令了超出指令范围的数值，则发生程序错误（P35）。 • 相对于基准主轴，指令移位量在顺时针方向上有效。 • 指令移位量的最小分辨率为 半闭环时 (仅仅齿轮比为 1: 1 时) 360/4096 (°) 全闭环时 (360/4096) * K (°) K: 主轴与编码器的齿轮比。 • 当没有 R 指令时，不进行相位匹配。
A	主轴同期加减速时间常数 设定主轴同期指令转速发生变化时的加减速时间常数。 (当试图按照参数中设定的时间常数，慢慢进行加减速时，执行本指令。)	0.001~9.999 (s) 或 1~9999 (ms)	• 如果指令了超出指令范围的数值，则发生程序错误（P35）。 • 当指令值小于参数中设定的加减速时间常数时，以参数中设定的值为准。



旋转轴与旋转方向

- (1) 在主轴同期控制中，基准主轴与同期主轴的转速及旋转方向，变为针对基准主轴指定的转速及旋转方向。但是，根据程序不同，同期主轴的旋转方向可能与基准主轴相反。
- (2) 基准主轴的转速及旋转方向在主轴同期控制中也能够进行变更。
- (3) 同期主轴的旋转指令，在主轴同期控制中也有效。
当指令了主轴同期控制时，在既没有向同期主轴输入正转指令，也没有输入反转指令时，不开始同期主轴的旋转，而是进入同步等待状态。在这一状态下，输入正转指令或反转指令，则开始同期主轴的旋转。但是，同期主轴的旋转方向变为程序中所指令的方向。
在主轴同期控制中，对同期主轴指令了主轴停止（正转指令、反转指令两者均被关闭）时，则同期主轴的旋转停止。
- (4) 在主轴同期控制中，对同期主轴进行的转速指令（S指令）及线速度恒定控制无效。但是仍然进行模式的更新，所以当主轴同期被取消时，生效。
- (5) 即使在主轴同期控制中，也可以通过对基准主轴发出指令，进行线速度恒定控制。

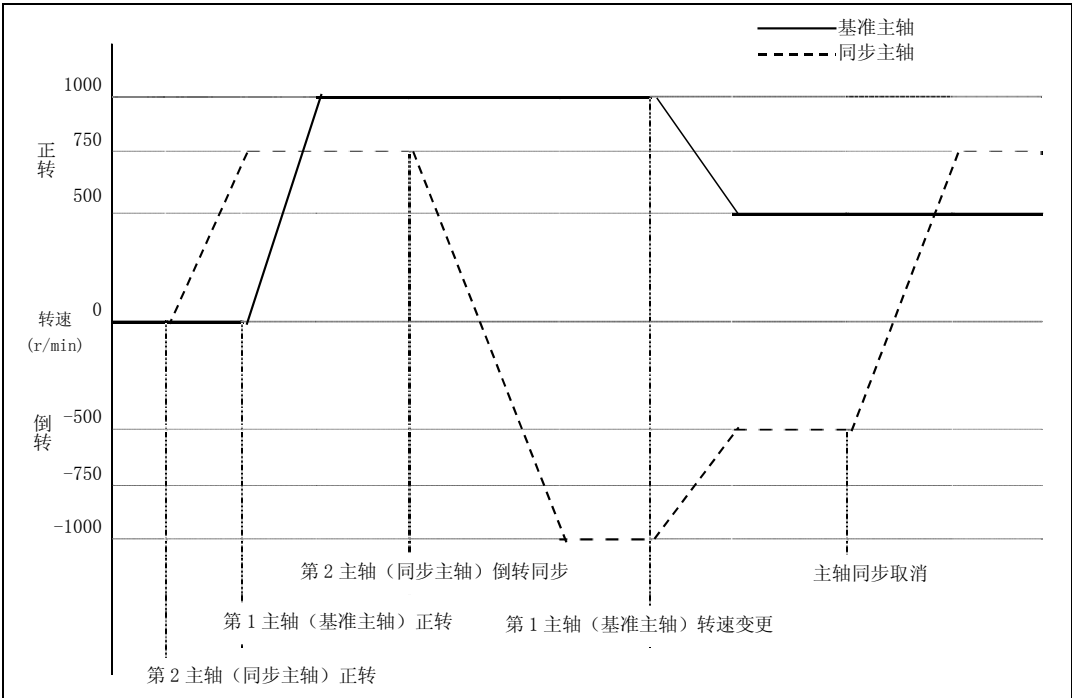


旋转同步

- (1) 通过G114.1指令发出旋转同步指令（无R地址指令），则以任意转速旋转的同期主轴，加减速到预先指定的基准主轴指令转速，进入旋转同步状态。
- (2) 在旋转同步状态下变更基准主轴的指令转速，则根据参数中设定的主轴加减速时间常数，始终保持同步状态进行加减速，达到指定的转速。
- (3) 在旋转同步状态下，即使是2台主轴同期夹持同一工件的状态下，也可对基准主轴进行线速度恒定控制。
- (4) 动作如下。

M23 S2=750 ;	……让第 2 主轴（同期主轴）以 750r/min 的转速正转（速度指令）
:	
M03 S1=1000 ;	……让第 1 主轴（基准主轴）以 1000r/min 的转速正转（速度指令）
:	
G114.1 H1 D-2 ;	……让第 2 主轴（同期主轴）反转，与第 1 主轴（基准主轴）同步
:	
S1=500 ;	……将第 1 主轴（基准主轴）的转速变更为 500r/min
:	
G113 ;	……取消主轴同期

<动作>



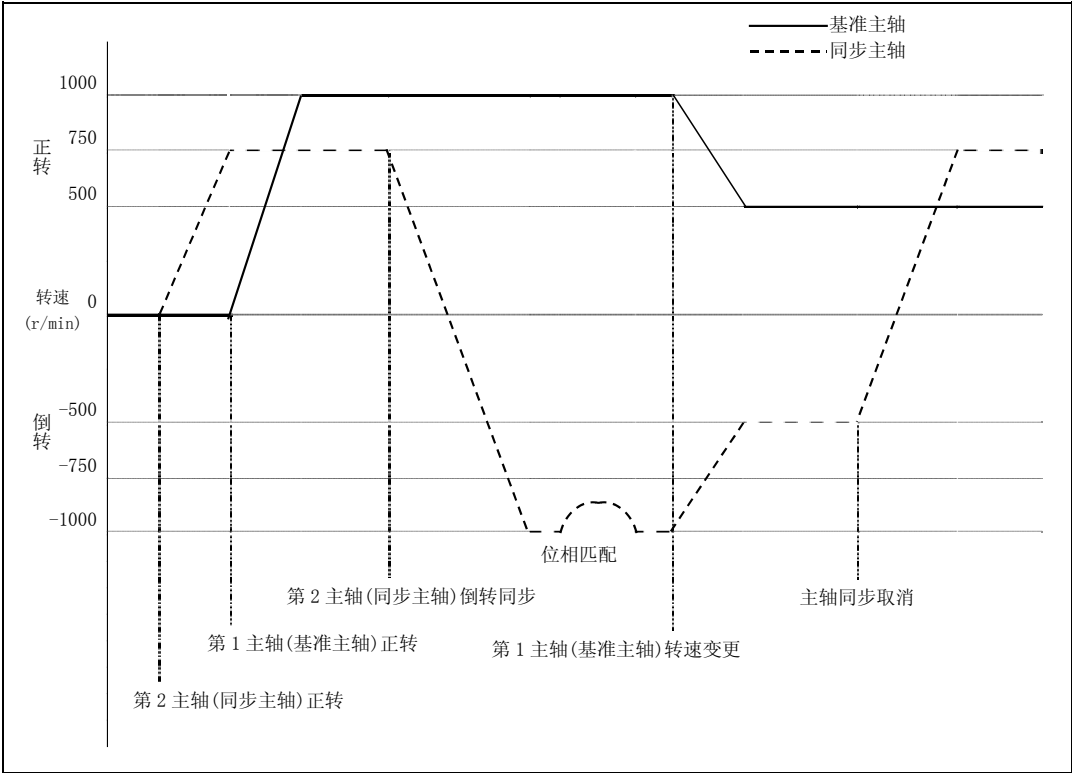


位相同步

- (1) 通过G114.1指令发出相位同步指令（无R地址指令），则以任意转速旋转的同期主轴，加减速到预先指定的基准主轴指令转速，进入旋转同步状态。之后，进行相位匹配，确保达到R地址所指定的旋转相位，进入相位同步状态。
- (2) 在相位同步状态下变更基准主轴的指令转速，则根据参数中设定的主轴加减速时间常数，始终保持同步状态进行加减速，达到指定的转速。
- (3) 在相位同步状态下，即使是2台主轴同期夹持同一工件的状态下，也可对基准主轴进行线速度恒定控制。
- (4) 动作如下。

M23 S2=750 ;	让第 2 主轴（同期主轴）以 750r/min 的转速正转（速度指令）
:	
M03 S1=1000 ;	让第 1 主轴（基准主轴）以 1000r/min 的转速正转（速度指令）
:	
G114.1 H1 D-2 Rxx ;	让第 2 主轴（同期主轴）反转，与第 1 主轴（基准主轴）同步
:	将同期主轴的相位移位 R 指令值。
:	
S1=500 ;	将第 1 主轴（基准主轴）的转速变更为 500r/min
:	
G113 ;	取消主轴同期

<动作>





编程注意事项

(1) 在基准主轴・同期主轴夹持同一工件的状态下进入旋转同步模式时，请在主轴同期控制模式ON之前，将基准主轴与同期主轴的旋转命令ON。

```

:
M6;           第1主轴夹头闭合
:
M5 S1=0;      第1主轴S=0停止
M3;           第1主轴旋转命令ON
:
M15;          第2主轴夹头闭合
M25 S2=0;     第2主轴S=0停止
M24;          第2主轴旋转命令ON
:
G114.1 H1 D-2; 旋转同步模式ON
:
S1=1500;      以S=1500同步旋转
:
S1=0;         两主轴停止
G113          同步模式取消
```

(2) 在相位同步模式中，基准主轴与同期主轴夹持同一工件时，请在进行相位匹配之后，再进行夹持。

```

:
M6;           第1主轴夹头闭合
:
M3 S1=1500;    第1主轴旋转命令ON
:
:
G114.1 H1 D-2 R0; 相位同步模式ON
:
M24;           第2主轴旋转命令ON
:
M15;           第2主轴夹头闭合（注1）
```

（注 1）请在确认主轴相位同步完成信号（X30A）为 ON（相位匹配完成）之后，进行夹头闭合。

△ 注意

△ 在主轴同期控制模式中，基准主轴与同期主轴同时夹紧同一工件的状态下，请不要将同期主轴端的旋转指令 OFF。由于同期主轴会停止，所以非常危险。



注意事项・限制事项

- (1) 利用紧急停止，在主轴同期控制中，旋转中的主轴停止。
- (2) 主轴同期控制中的转速钳位，取决于基准主轴・同期主轴中钳位值较小的一个。
- (3) 在主轴同期控制模式中，无法对基准主轴・同期主轴进行定位。在进行定位时，请先取消主轴同期控制模式，然后再进行。
- (4) 针对主轴同期中的同期主轴的转速指令（S指令）无效。但是仍然进行模态的更新，所以当主轴同期被取消时，生效。
- (5) 在主轴同期控制模式中，对同期主轴进行的线速度恒定控制无效。但是仍然进行模态的更新，所以当主轴同期控制被取消时，生效。
- (6) 当主轴同期控制被取消时，针对同期主轴的旋转指令（S指令）及线速度恒定生效，所以，在取消时，同期主轴可能会进行与之前不同的动作，请加以注意。
- (7) 在相位移位计算请求信号ON时，进行相位同步指令（有R地址指令），则发生操作错误（1106）。
- (8) 在旋转同步指令时，相位移位计算请求信号为ON的状态下，且基准主轴或同期主轴旋转时，发生操作错误（1106）。
- (9) 在相位偏移请求信号ON的状态下指令相位同步指令R0（〈例〉G114.1 H1 D-2 R0）时，将基准主轴和同期主轴与被记忆到NC内存中的基准主轴与同期主轴相位差进行相位匹配。
- (10) 在相位偏移请求信号ON的状态下指令相位同步指令R0以外的数值（〈例〉G114.1 H1 D-2 R100）时，将基准主轴和同期主轴，与在被记忆到NC内存中的基准主轴与同期主轴相位差的基础上，增加了R地址指令值之后的相位差进行相位匹配。
- (11) 当相位移位计算请求信号为ON时，相位偏移请求信号被忽略。
- (12) 仅当相位移位计算信号为ON，且旋转同步指令（无R指令）所指定的基准主轴选择（H₋）与同期主轴选择（D₋）的组合下，NC中所记忆的基准主轴与同期主轴相位差生效。例如，在“G114.1 H1 D-2；”中记忆了基准主轴与同期主轴的相位差时，仅当相位偏移请求信号为ON，且指令了“G114.1 H1 D-2 R***；”时，所记忆的相位差生效。另外，在本例中，指令“G114.1 H2 D-1 R***；”，则无法正确计算出相位移位量，所以必须加以注意。
- (13) 当主轴同期的指令方法为通过PLC I/F的方法时（#1300 ext36/bit7 OFF），通过G114.1/G113指令主轴同期控制，则发生程序错误（P610）。
- (14) 请务必实施夹头闭合。当不实施时，可能会给机械造成过大负载，或是发生报警。

10.5.2 主轴同期控制 II



功能及目的

在主轴同期控制 II 中，同期主轴的选择及同步开始等全部通过PLC指定。详情请参阅机械制造商发行的说明书。



基准主轴及同步主轴的选择

通过PLC选择进行同期控制的基准主轴及同期主轴。

设备编号	信号名称	简称	说明
R446	基准主轴选择	—	从串行连接的主轴中选择作为基准主轴加以控制的主轴。 (0: 第 1 主轴)、1: 第 1 主轴、2: 第 2 主轴 (注 1) 当选择了未串行连接的主轴时，不进行主轴同期控制。 (注 2) 当指定 “0” 时，将第 1 主轴作为基准主轴加以控制。
R447	同期主轴选择	—	从串行连接的主轴中选择作为同期主轴加以控制的主轴。 (0: 第 2 主轴)、1: 第 1 主轴、2: 第 2 主轴 (注 3) 当选择了未串行连接的主轴及已被选为基准主轴的主轴时，不进行主轴同期控制。 (注 4) 当指定 “0” 时，将第 2 主轴作为同期主轴加以控制。



开始主轴同步

通过输入主轴同期控制信号（SPSYC），进入主轴同期控制模式。主轴同期控制模式中，与基准主轴的指令转速同步，控制同期主轴。

当基准主轴与同期主轴间的转速差达到主轴同期转速达到等级设定值（#3050 sprlv）时，输出主轴转速同步完成信号（FSPRV）。

通过主轴同期旋转方向指定，选择同期主轴的旋转方向是与基准主轴一致，还是反方向旋转。

设备编号	信号名称	简称	说明
Y398	主轴同期控制信号	SPSYC	通过打开本信号，进入主轴同期控制模式。
X308	主轴同期控制中	SPSYN1	通知处于主轴同期控制中。
X309	主轴旋转速度同步完成信号	FSPRV	在主轴同期控制模式中，当基准主轴与同期主轴的转速差达到主轴转速到达等级设定值时，变为 ON。 当主轴控制模式被解除时，或是在主轴同期控制模式中，误差超过主轴转速到达等级设定值时，本信号被关闭。
Y39A	主轴同期旋转方向指定	-	指定主轴同期控制时的基准主轴 • 同期主轴的旋转方向。 0：同期主轴与基准主轴向同一方向旋转。 1：同期主轴与基准主轴向反方向旋转。

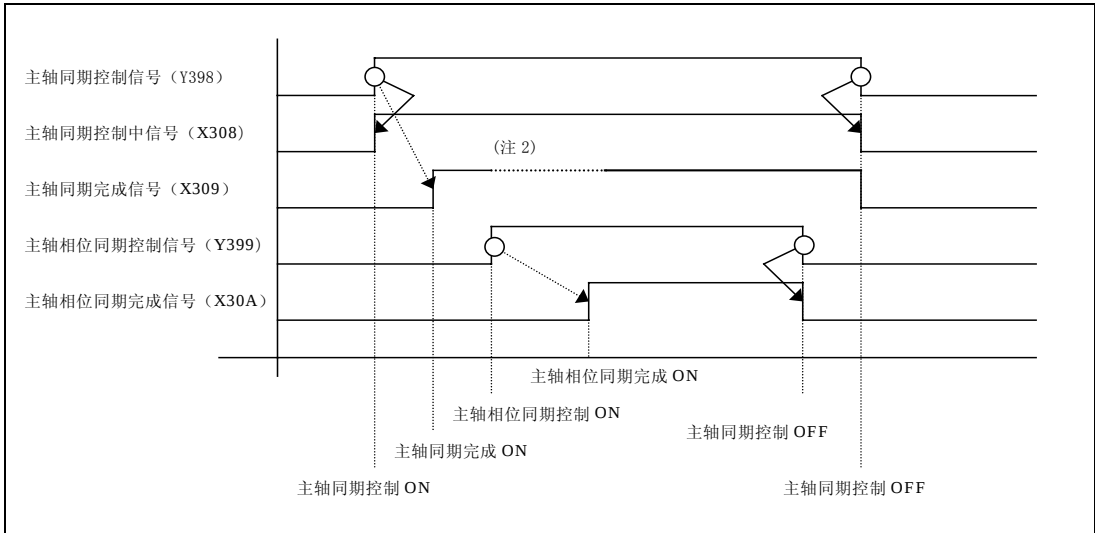


主轴位相匹配

在主轴同期控制模式中输入主轴相位同期控制信号（SPPHS），则开始主轴相位同步，当达到主轴同期相位到达等级设定值（#3051 spplv）时，输出主轴相位同步完成信号。

另外，也可通过PLC指定同期主轴的相位移位量。

设备编号	信号名称	简称	说明
Y399	主轴相位同期控制信号	SPPHS	在主轴同期控制模式中打开本信号，则开始主轴相位同步。 （注 1）除在主轴同期控制模式中外，即使本信号打开，也被忽略。
X30A	主轴相位同步完成信号	FSPPH	主轴相位同步开始后，当达到主轴同期相位到达等级时，输出本信号。
R448	相位移位量设定	—	指定同期主轴的相位移位量。 单位：360° /4096



（注2）在相位同步时，为了变化转速，暂时关闭。

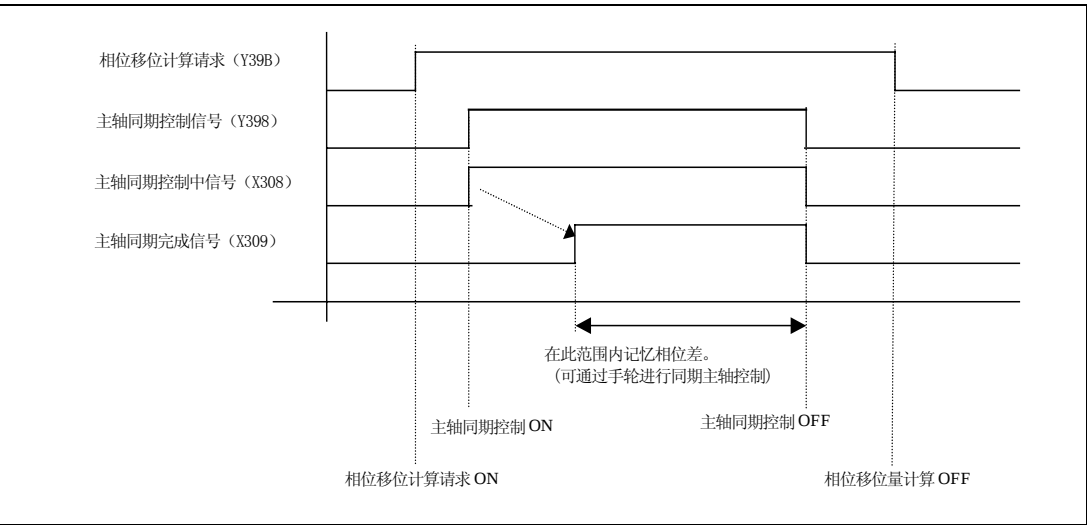


主轴同期相位位移量的计算与相位偏移请求

主轴相位位移量计算功能，是在执行主轴同期时，通过将PLC信号ON，计算基准主轴与同期主轴的相位差，并加以记忆。在主轴相位位移计算中，可通过手轮旋转同期主轴，所以可通过目视调整主轴间的相位关系。在相位偏移请求信号（SSPHF）ON的状态下，输入主轴相位同期控制信号，则以按照被记忆的相位位移量进行移位后的位置为基准，进行相位差匹配。

借此，易于在换向夹持两端形状不同的加工对象时，进行相位匹配。

设备编号	信号名称	简称	说明
Y39B	相位移位计算请求	SSPHM	在将本信号 ON 的状态下进行主轴同期，则计算、记忆基准主轴与同期主轴的相位差。
Y39C	相位偏移请求	SSPHF	在将本信号 ON 的状态下进行主轴相位同步，则以移动了所记忆的移位量之后的位置作为基准位置，进行相位匹配。
R474	相位差输出	—	输出相对于基准主轴的同期主轴延迟。 单位：360° / 4096 （注 1）当基准主轴/同期主轴中的任何一个未通过 Z 相，无法进行计算时，输出-1。 （注 2）仅在相位移位计算中或是主轴相位同步中，输出本数据。
R490	相位偏移数据	—	输出通过相位移位计算而记忆的相位差。 单位：360° / 4096 （注 3）仅在主轴同期控制中输出本数据。



（注4）在相位移位计算中，无法进行相位匹配。

（注5）当手动运转模式为手轮模式时，无法通过手轮旋转同期主轴。



注意事项・限制事项

- (1) 进行主轴同期时，基准主轴・同期主轴均需执行旋转指令。但是，同期主轴的旋转方向与正转・反转指令无关，而是取决于基准主轴的旋转方向与主轴同期旋转方向指定。
- (2) 在主轴转速指令为ON的状态下，将主轴同期控制信号打开，则虽然会进入主轴同期控制模式，但是不会进行实际的同期控制。对基准主轴进行转速指令之后，进行同期控制，输出主轴同期完成信号。
- (3) 通过紧急停止，主轴同期中的主轴停止。
- (4) 基准主轴及同期主轴指定不正确的状态下打开主轴同期控制信号，则发生操作错误。
- (5) 主轴同期控制中的转速钳位，取决于基准主轴・同期主轴钳位值中较小的一个。
- (6) 在主轴同期中，无法对基准主轴・同期主轴进行定位。进行定位时，必须将主轴同期控制信号关闭。
- (7) 针对主轴同期中的同期主轴的转速指令无效。取消主轴同期之后，指定的转速生效。
- (8) 主轴同期中，针对同期主轴的线速度恒定控制无效。
- (9) 在不进行相位移位计算的情况下，打开相位偏移请求信号，进行主轴相位同步，则由于未计算出移位量，所以必须加以注意。
- (10) 在进行相位偏移时，主轴Z相编码器位置参数无效。
- (11) 相位移位计算请求信号为ON的状态下进行主轴相位同步，则发生“M01 操作错误 1106”。
- (12) 请在基准主轴・同期主轴均停止时，打开相位移位计算请求信号。在任何一根主轴旋转的状态下打开相位移位计算请求信号，则发生“M01 操作错误1106”。
- (13) NC中保存的相位移位量，一直被保持到下一次计算相位移位之前为止。（即使切断电源，也被保持。）
- (14) 请务必实施夹头闭合。当不实施时，可能会给机械造成过大负载，或是发生报警。

△ 注意

- △ 在主轴同期控制模式中，基准主轴与同期主轴同时夹紧同一工件的状态下，请不要将同期主轴端的旋转指令 OFF。由于同期主轴会停止，所以非常危险。

10.5.3 使用主轴同期控制时的注意事项



功能及目的

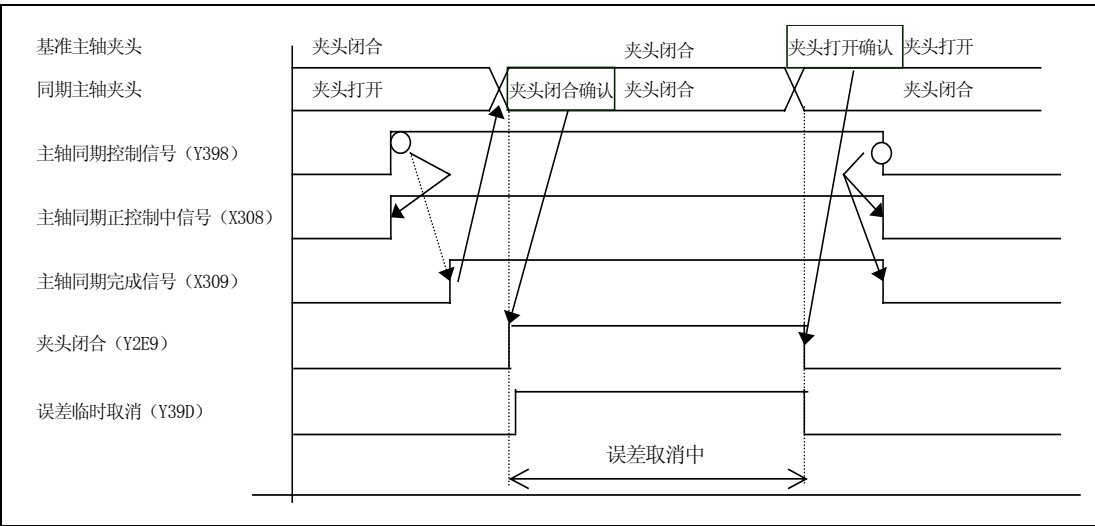
主轴同期控制 I、II 中，存在通用的 PLC 信号。如果不设定这些信号，则可能会发生负载过大及报警。详情请参阅机械制造商发行的说明书。在此，对各功能与信号加以说明。



卡盘闭合信号

当夹头打开时，在同期主轴端进行偏移量补偿，与基准主轴进行对准，如果关闭夹头，则偏移量补偿被附加，基准与同步间的差值扩大。通过位置补偿，保持使用夹头进行夹持的位置，以确保不会因为夹头关闭信号而进行偏移量补偿。

设备编号	信号名称	简称	说明
Y2E9	夹头闭合信号	-	当夹头闭合时，本信号 ON。通过打开本信号，基准主轴与同期主轴的补偿从偏移量补偿变为位置补偿。
X30E	夹头闭合确认信号	-	在主轴同期控制模式中，当接收到检查关闭信号时，本信号打开。



(注 1) 请仅在因夹头闭合信号而导致主轴与同步的误差时，使用误差临时取消。



误差临时取消功能

让基准主轴夹持工件进行旋转的状态下进行主轴同期，则由于同期主轴也夹持工件，所以夹头闭合时，会因为外在因素而导致速度变动，产生误差。如果不补偿该误差而继续进行主轴同期，则工件可能会产生扭曲。

可通过临时性的取消该误差，防止发生扭曲。

设备编号	信号名称	简称	说明
Y39D	误差临时取消信号	SPDRP0	本信号 ON 时，取消误差。



位相差监视器

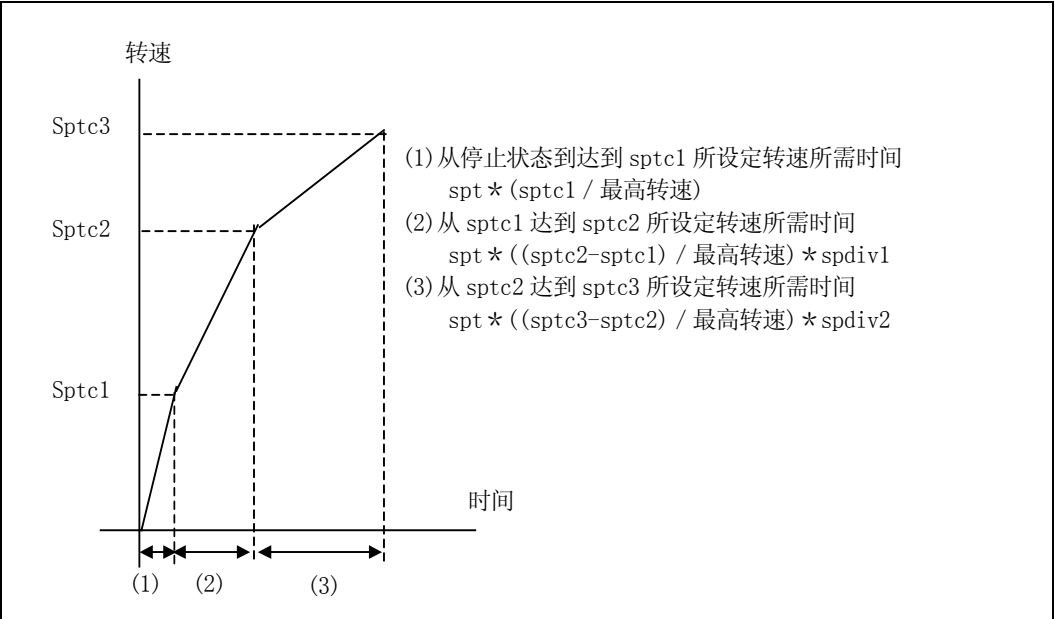
可监视主轴相位同步中的相位误差。

设备编号	信号名称	简称	说明
R477	相位差监视	-	以脉冲为单位，输出主轴相位同期控制中的相位误差。
R478	相位差监视 (下限值)	-	以脉冲为单位，输出主轴相位同期控制中的相位误差的下限值。
R479	相位差监视 (上限值)	-	以脉冲为单位，输出主轴相位同期控制中的相位误差的上限值。



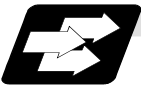
多级加减速

可根据主轴转速，为主轴同期时的加减速控制选择最多8级加减速时间常数。



10.6 多根主轴控制

[E68]



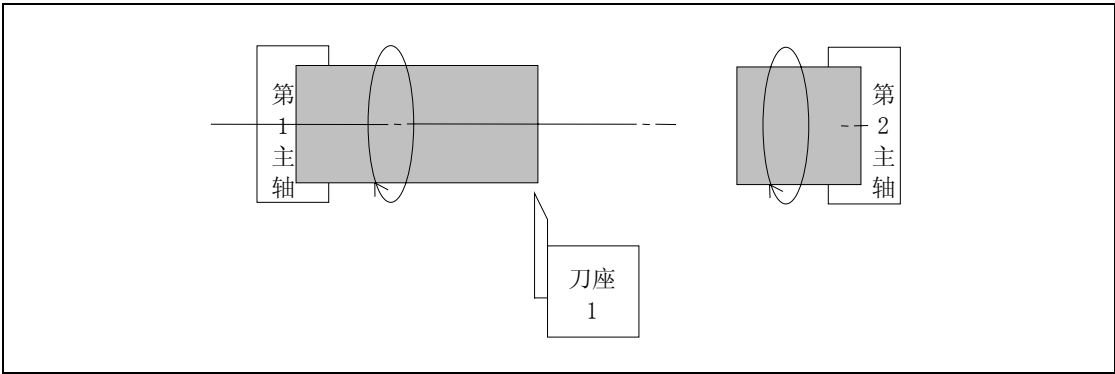
功能及目的

多根主轴控制，是针对除了主轴（第1主轴）外，还具有辅助轴（第2主轴）的机床，为了控制其辅助轴而具备的功能。

主轴的控制方法因多根主轴控制 I 与多根主轴控制 II 而异。通过参数（#1300 ext36/bit0）的设定，选择让多根主轴控制 I 与多根主轴控制 II 的任何一个生效。

多根主轴控制 I：通过主轴选择指令（G43.1等）与主轴控制指令（[S*****；]
(ext36/bit0 = 0) 或[SO=*****；]）等进行控制。

多根主轴控制 II：通过外部信号（主轴指令选择信号、主轴选择信号）与主
(ext36/bit0 = 1) 轴控制指令（仅[S*****；]）等进行控制。
不可使用主轴选择指令、[SO=*****；]。



10.6.1 多根主轴控制 I（多根主轴指令）



功能及目的

S指令除了S*****指令外，还可利用SO=*****指令区分第1及第2主轴的指令。



指令格式

SO=* * * * * ;	
○	: 使用 1 个数字（1：第 1 主轴/2：第 2 主轴）指定主轴编号。可指定变量。
* * * * *	: 转速或线速度指令值。可指定变量。

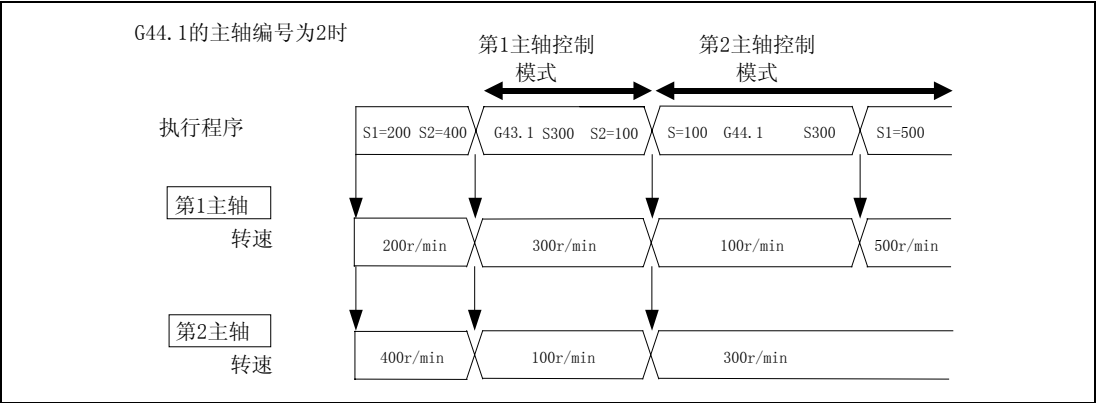
（注 1）○的值为 1 或 2 以外时，发生程序错误（P35）。

（注 2）在模态中，G47.1 为程序错误（P33）。



详细说明

- （1）根据○的内容，进行各主轴指令的区分。
（例）
S1=3500；第1主轴 3500（r/min）指令
S2=1500；第2主轴 1500（r/min）指令
- （2）可在1节内同时进行多根主轴的指令。
- （3）当在1节内对同一主轴进行了2个以上的指令时，则仅最后的指令有效。
（例）S1=3500 S1=3600 S1=3700 ； S1=3700为有效指令。
- （4）S*****指令与SO=*****指令可并用。
通过主轴选择指令，区分S*****指令中的指令对象主轴。



10.6.2 多根主轴控制 I（主轴选择指令）



功能及目的

根据主轴选择指令（G43.1等 [G组20]），可切换以下的S指令（S****）是针对第1主轴及第2主轴的哪一根轴进行。



指令格式

G43.1;	第 1 主轴控制模式打开
G44.1;	选择主轴控制模式打开（通过 SnG44.1 设定选择主轴编号）
G47.1;	所有主轴同期控制模式打开



详细说明

- （1）通过参数（#1534 SnG44.1）设定选择主轴编号。
- （2）主轴选择指令为模态G代码。
- （3）当多根主轴控制II有效时，主轴选择指令为程序错误（P33）。
- （4）可通过参数设定接通电源时，或是重新启动时的主轴控制模式。另外，接通电源时或重新启动时的状态，如下所示。

G组20模态状态：	在安装参数“#1199 Sselect”中设定 0:G43.1 1:G44.1 2:G47.1
G44.1的主轴编号：	在安装参数“#1534 SnG44.1”中设定 0: 第2主轴 1: 第1主轴 2: 第2主轴

- （5）当在同一节中同时指定了主轴选择指令与S指令时，则对通过主轴选择指令切换的主轴生效。

例)		
G43.1 S100 ;	→	以100转对第1主轴进行指令
S200 G44.1 S300 ;	→	以200转对第1主轴进行指令之后，以300转对第2主轴进行指令 (SnG44.1=2)

- （6）当设定了不存在的主轴时，为第2主轴。但是，当主轴数=1时，为第1主轴。



与其他功能的关联

主轴选择指令后进行切换的功能如下。

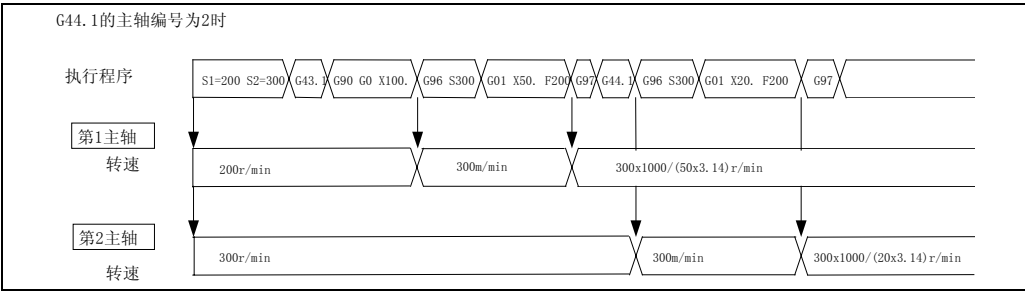
- (1) S指令 (S*****)
G97 (转速指令) /G98 (线速度恒定指令) 中的S指令，只对主轴选择指令所指定的主轴生效。
- (2) 主轴钳位速度指令
通过G92 S_Q_进行的主轴钳位速度指令也取决于主轴选择指令的模式。
- (3) 每转指令 (同步进给)
在G95模式中进行了F指令时，也变为主轴选择指令所指定主轴的每转进给速度。
- (4) 线速度恒定控制主轴的切换
线速度恒定控制也取决于主轴选择指令的模式。

通过S○=*****, 向不同于当前模式的其他主轴发出指令时，○所指定的主轴的转速指令优先。

(例)

G43系列模式中的S2=****第2主轴的转速指令
G44系列第2主轴选择模式中的S1=****第1主轴的转速指令

但是，线速度恒定控制 (G96) 的线速度恒定指令，与转速指令不同，属于个别的模态信息。



10.6.3 多根主轴控制 II



功能及目的

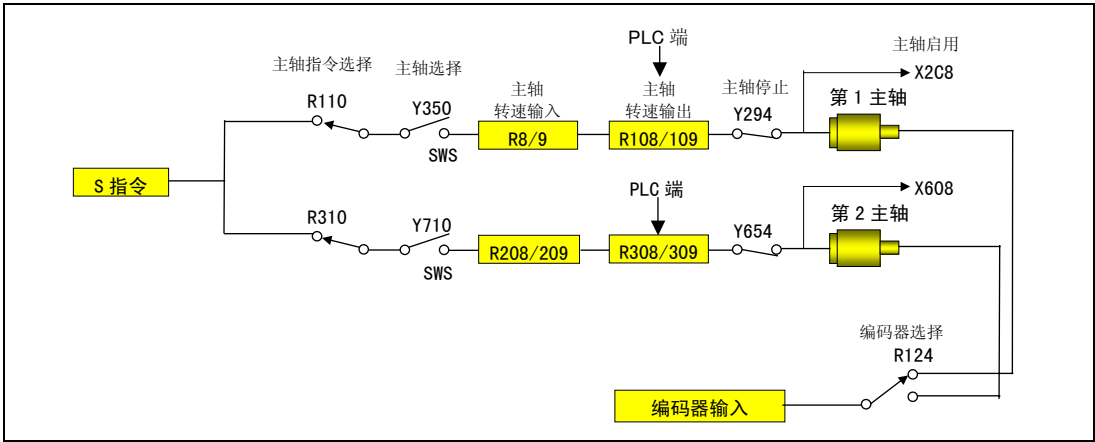
多根主轴控制 II 是通过来自 PLC 的信号，指定选择哪一根主轴的功能。通过 1 个 S 指令进行对主轴的指令。



详细说明

(1) 主轴指令选择、主轴选择

对于通过来自 PLC 的主轴选择信号（SWS）起动而被选中的主轴，对主轴的 S 指令是作为转速指令而被输出。被选中的主轴以输出的转速旋转。通过主轴选择信号（SWS）关闭而未选中的主轴，保持未选中前的转速，继续旋转。借此，可让各主轴同时以各自的转速进行旋转。



(注) 各信号的详情请参阅 PLC 接口说明书。



与其他功能的关联

(1) 主轴钳位速度设定 (G92)

仅对通过主轴选择信号 (SWS) 选中的主轴有效。

因主轴选择信号 (SWS) 而未选中的主轴，保持未选中前的转速，继续旋转。

(通过G92指令保持主轴钳位速度)

(2) 线速度恒定控制

可在所有的主轴上进行线速度恒定控制。

在线速度恒定中，主轴转速为自动控制，所以线速度恒定加工中，该主轴必须保持主轴选择信号 (SWS) 打开的状态。

因主轴选择信号 (SWS) 而未选中的主轴，保持未选中前的转速，继续旋转。

(3) 螺纹切削/同步进给

以通过主轴选择信号 (SWS) 选中的主轴进行螺纹切削。使用通过编码器选择信号选中的编码器进行编码器回馈。

(4) 同期攻牙

通过主轴选择信号 (SWS) 进行同期攻牙主轴的选择。

请在同步螺纹切削指令之前，进行同期攻牙主轴的选择。在同期攻牙模式中，请不要切换同期攻牙主轴的选择信号。

对同期攻牙主轴进行C轴模式指令时，发生“M01 操作错误 1026”。取消C轴指令，则解除错误，重新开始加工。

(5) 非同期攻牙

通过主轴选择信号 (SWS) 进行非同期攻牙主轴的选择。

请在攻牙指令之前，进行非同期攻牙主轴的选择。在切换非同期攻牙主轴选择时，请输入计算请求。

在非同期攻牙模式中，请不要切换非同期攻牙主轴的选择信号。

(6) 攻牙返回

通过主轴选择信号 (SWS) 进行攻牙返回主轴的选择。

请在打开攻牙返回信号之前，选择中断攻牙循环时的主轴。在选择了不同主轴的状态下，执行了攻牙返回时，发生“M01 操作错误1032”。请不要在攻牙返回中切换主轴选择信号。



限制事项

(1) 多根主轴控制 II 有效时，S 的手动数值指令无效。

(2) 多根主轴控制 II 有效时，安装参数“#1199 Sselect”无效。

(3) 多根主轴控制 II 有效时，主轴控制模式切换G代码无法使用。发生程序错误 (P34)。

(4) 多根主轴控制 II 有效时，“S1=***”、“S2=***”指令无效。发生程序错误 (P33)。

(5) 多根主轴控制 II 有效时，不输出主轴齿轮移位指令输出信号 (GR1/GR2)。

11. 刀具功能

11.1 刀具功能（T8位BCD）

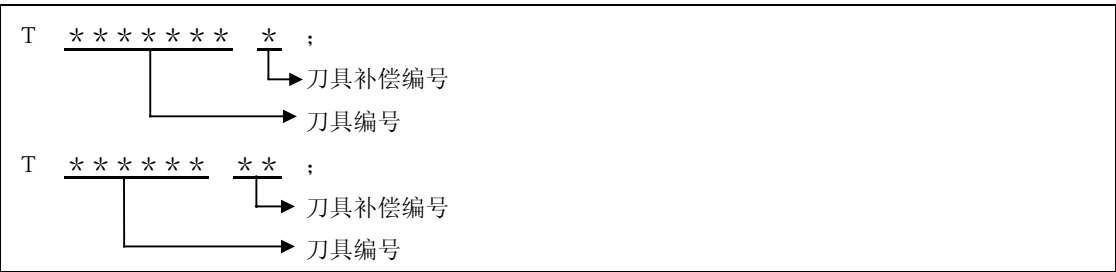


功能及目的

刀具功能也称为T功能，是用于指定刀具编号及刀具补偿编号的功能。在地址T后面接8位数值（0～99999999）发出指令，前面6位或7位为刀具编号，后面2位或1位为补偿编号。
使用哪一个取决于参数“#1098 Tlno.”的设定。另外，能够使用的T指令因各机械而异，所以，请参阅机械制造商发行的说明书。每1节可发出1组T指令。



指令格式



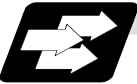
程序中指令的刀具编号与实际刀具的对应关系，请参阅机械制造商发行的说明书。
以BCD代码与起动信号输出。
当在同一节中同时指定T功能与移动指令时，指令的执行顺序，可能有以下2种状况。适用哪一种，取决于机械规格。

- （a）移动完成后，执行T功能。
- （b）在执行移动指令的同时，执行T功能。

对于所有的T指令，需要处理及完成顺序。

12. 刀具偏移功能

12.1 刀具补偿



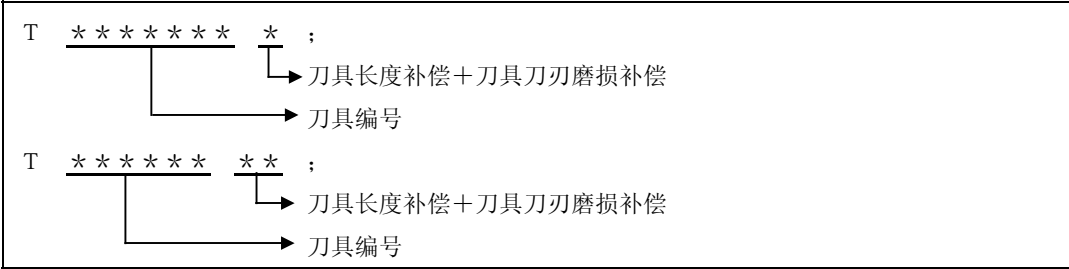
功能及目的

通过T指令进行刀具补偿，以接在地址T之后的3位、4位或8位数值发出指令。在刀具补偿中，包括刀具长度补偿与刀具刀刃磨损补偿，通过参数“#1098 TLno.”切换是以T指令的最后1位或2位指定刀具长度补偿与刀具刀刃磨损补偿，还是以T指令的最后1位或2位指定刀具刀刃磨损补偿、以刀具编号指定刀具长度补偿。另外，也可通过参数“#1097 Tldit”切换是以最后1位还是2位指定补偿。每1节可发出1组T指令。

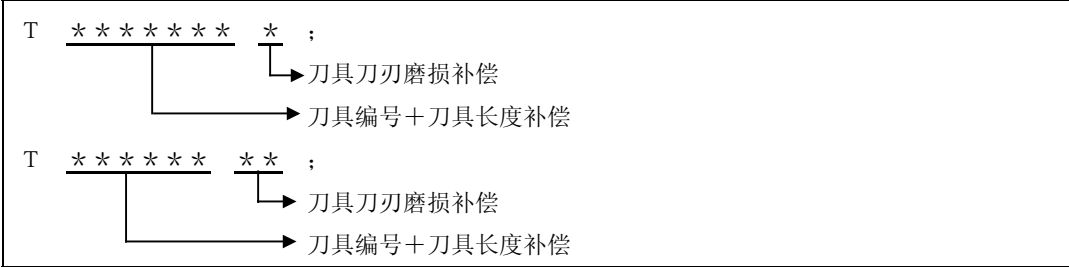


指令格式

(1) 当以T指令的最后1位或最后2位指定刀具长度与刀刃磨损的补偿编号时



(2) 区分刀具长度补偿编号与刀具刀刃磨损补偿编号时



刀具长度补偿编号为刀具编号的最后2位。

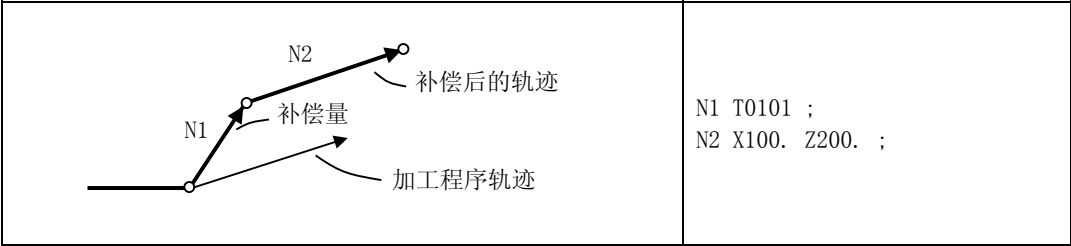
12.1.1 刀具补偿开始



详细说明

在执行刀具补偿时，可通过参数切换，是在执行了T指令之后进行补偿动作，还是在执行T指令之后不进行补偿动作，而是在执行有移动指令的节时执行补偿动作。

(1) 执行T指令时补偿



同时进行刀具长度补偿、刀刃磨损补偿。

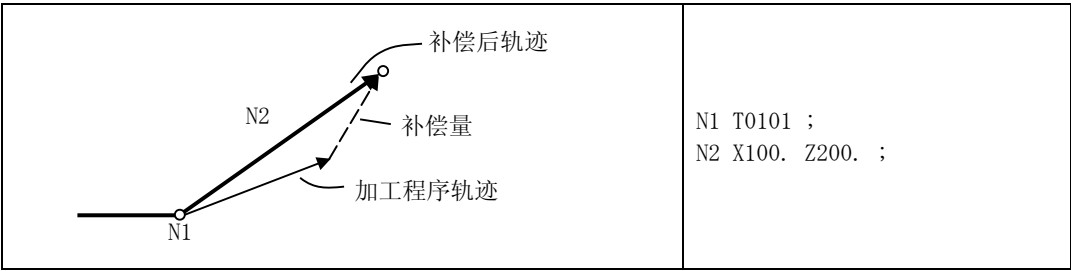
(注 1) 执行 T 指令中进行补偿时的移动，在 G00 模态中为快速进给。其他模态下为切削进给。

(注 2) 在执行 T 指令中进行补偿时，在圆弧模态下，作为直线移动进行补偿。

(注 3) 在执行 T 指令中进行补偿时，当如下所示在同一节中指令了 T 指令与 G 指令时，在接收到如下所示的 G 指令以外的 G 指令之前，不进行补偿。

- G04: 延时
- G10: 程序刀具补偿输入/程序参数输入
- G11: 取消程序参数输入模式
- G65: 用户宏 纯调用
- G92: 坐标系设定

(2) 移动指令时补偿



同时进行刀具长度补偿、刀刃磨损补偿。

(注 1) 在移动中进行补偿时，当首次通过圆弧指令进行补偿时，如果补偿量小于参数“#1084 RadErr”，则需要补偿。如果大于该参数，则发生程序错误 (P70)。(在执行 T 指令时进行补偿的场合下，当 T 指令与圆弧指令在同一节中时，也相同。)

12.1.2 刀具补偿开始方式扩展



功能及目的

关于T指令时的补偿动作，根据参数“#1100 Tmove”的设定，决定是在执行T指令时进行补偿动作，还是与移动指令重叠，进行补偿动作。

另外，根据参数的设定，也可以在执行T指令时仅对磨损量进行补偿，而刀具长度则与移动指令重叠，进行补偿。



详细说明

当选择在执行T指令时仅对磨损量进行补偿，而刀具长度与移动指令重叠，进行补偿的类型时，将“#1100 Tmove”设定为2。

设定为2时的动作，长度补偿是基于设定为1（与移动指令重叠）时、磨损补偿是基于设定为0（T指令时移动）时的动作。

“#1100 Tmove”的各设定的补偿动作如下所示。

长度补偿量：7.000mm 磨损补偿量：0.500mm

加工程序例	机械值		
	#1100 Tmove = 0	#1100 Tmove = 1	#1100 Tmove = 2
G28 X. ;	0.000	0.000	0.000
G0 T116 ;	7.500	0.000	0.500
G0 X100. ;	107.500	107.500	107.500
:			

（注1）当与T指令同一节中，有如下的G指令时，不在该节进行磨损补偿。

同一节补偿待机：G04、G10、G11、G65、G92

在进行了除此以外的G指令的节中，对磨损量进行补偿。（所有轴）

（注2）补偿量被临时取消后，针对移动指令轴，恢复补偿量。此时的移动，

长度补偿、磨损补偿均取决于参数“#1101 Tabsmv”。

进行临时取消的G指令：G28、G30、G53

（1）与“#1101 Tabsmv”（绝对指令时刀具补偿量重叠）的关联

“#1100 Tmove”为2时，补偿动作也取决于“#1101 Tabsmv”的设定。

（2）参数

#编号	项 目	内 容	设定范围
1100	Tmove	指定进行刀具长度补偿、磨损补偿的补偿动作的时机。 0: 执行T指令时进行补偿动作。 1: 与和T指令同一节内的移动指令重叠，进行补偿动作。 当在同一节内无移动时，则在之后的移动指令节中，与移动指令重叠进行补偿动作。 2: 执行T指令时进行磨损量的补偿动作。 刀具长度补偿量与同一节的移动指令重叠，进行补偿动作。 当在同一节中没有移动时，则在之后的移动指令节中，与移动指令重叠，进行补偿动作。	0~2
1101	Tabsmv	当Tmove为1或2时，指定重叠的移动指令。 0: 不管是绝对值指令还是增量值指令，都进行补偿动作。 1: 仅当绝对值指令下的移动指令时，进行补偿动作。	0 / 1

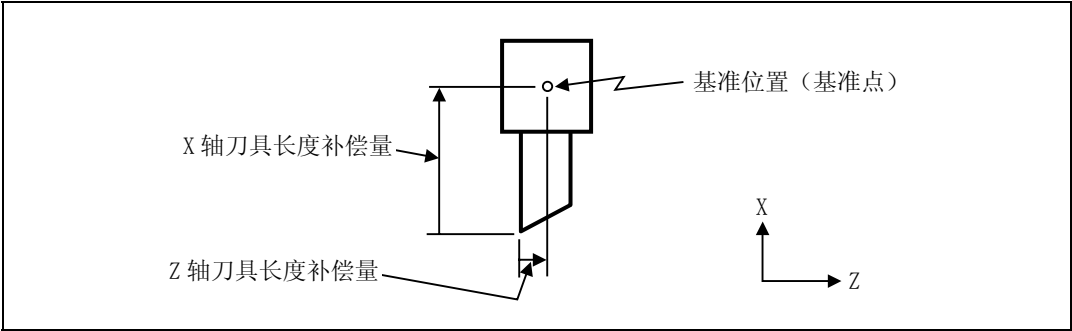
12.2 刀具长度补偿



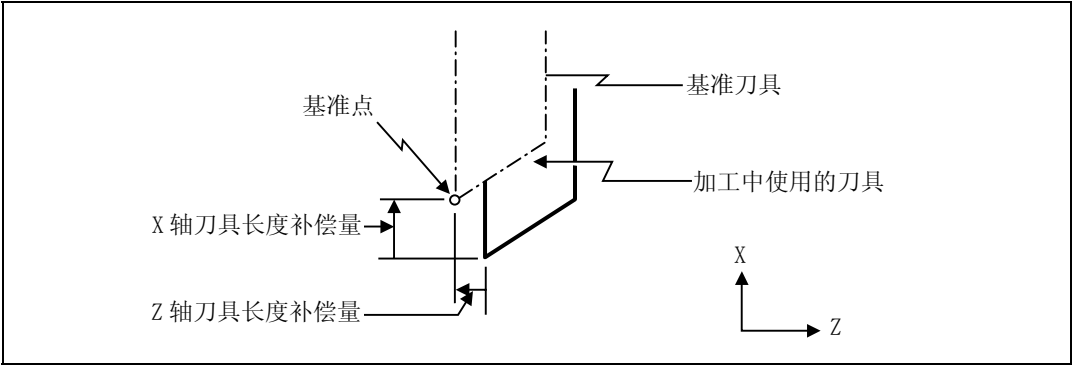
刀具长度补偿设定

补偿刀具相对于程序基准位置的长度。程序基准位置可能位于刀座的中心位置或是基准刀具的刀刃位置。

(1) 位于刀座中心位置时

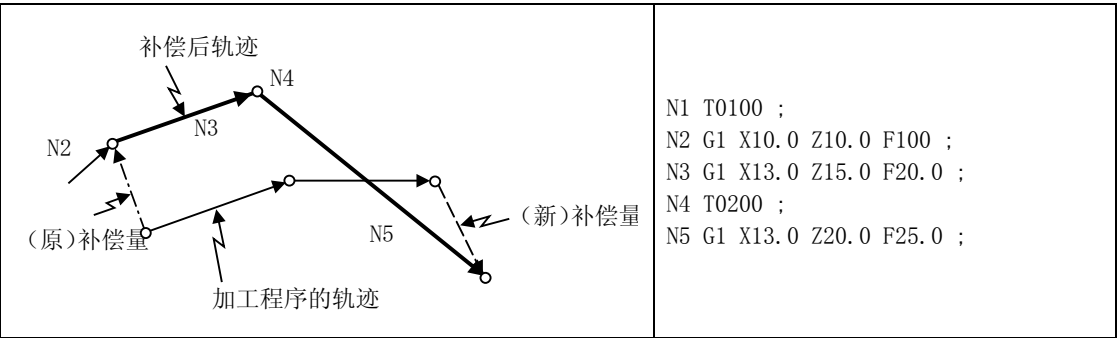


(2) 位于基准刀具的刀刃位置时



刀具长度补偿编号的变更

变更了刀具编号时，将与新刀具编号对应的刀具长度补偿量累加到加工程序中的移动量上。



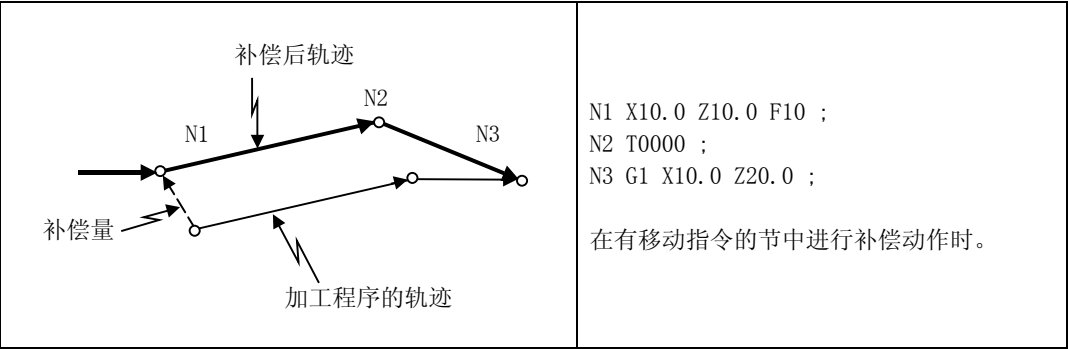
以刀具编号执行刀具长度补偿，在有移动指令的节中进行补偿动作的范例。



取消刀具长度补偿

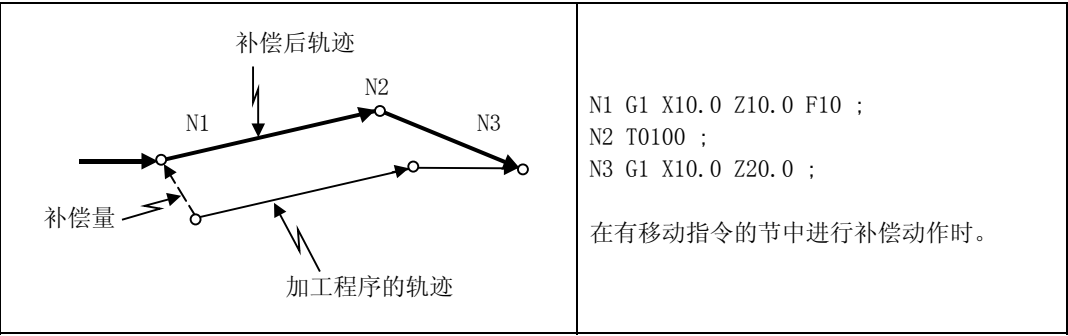
(1) 指令了补偿编号0时

在T指令中，将刀具长度补偿编号设定为0时，补偿被取消。



(2) 指令的补偿量为0时

在T指令中，将刀具长度补偿编号的补偿量设定为0时，补偿被取消。



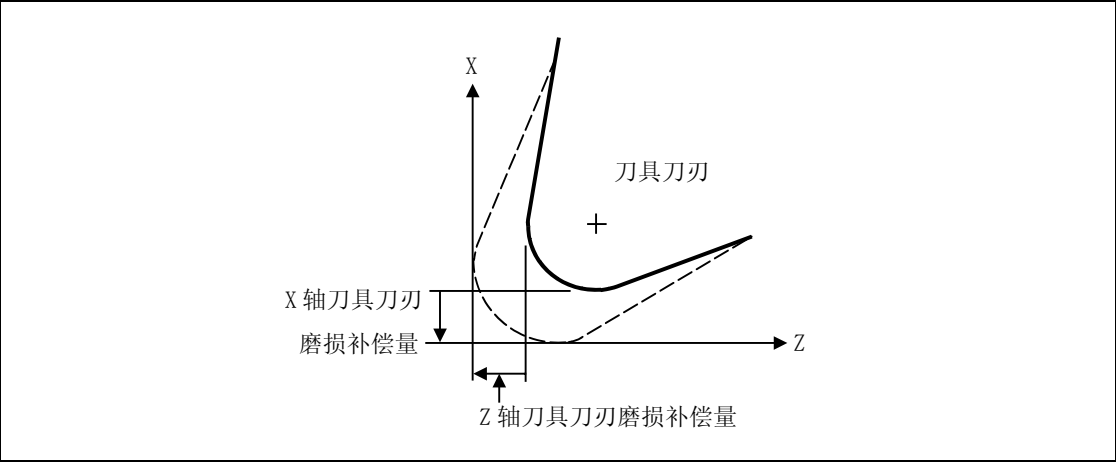
- (注 1) 指定 G28、G29、G30，则补偿被临时取消。为此，机床运动到补偿被取消的位置，但是补偿量被记忆下来，因此，在下次移动指令时，移动到补偿后的位置。
- (注 2) 当在同一节中指令了 G28、G29、G30 与补偿取消时，机床移动到补偿被取消的位置，补偿量被记忆住。因此，显示坐标可能会包含补偿量。为了不记忆补偿量，请在其他节中进行指令。
- (注 3) 在自动运转中，即使通过 MDI 等变更当前选中的补偿编号的补偿量，只要不再次执行同一编号的 T 指令，则变更后的补偿量不生效。
- (注 4) 刀具长度补偿、刀具刀刃磨损补偿量因重新启动、紧急停止而被清除。也可通过参数“#1099 Treset”予以保持。

12.3 刀具刀刃磨损补偿



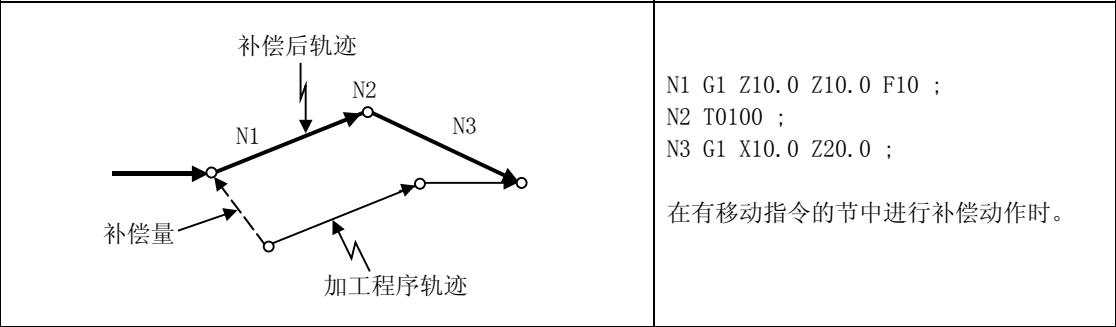
刀具刀刃磨损补偿量设定

所使用刀具的刀刃发生磨损时，可以对其加以补偿。



取消刀具刀刃磨损补偿

当刀刃磨损补偿编号为0时，补偿被取消。



- (注 1) 指定 G28、G29、G30，则补偿被临时取消。为此，机床运动到补偿量被取消的位置，但是补偿量被记忆下来，因此，在下次移动指令时，移动到补偿后的位置。
- (注 2) 当在同一节中指令了 G28、G29、G30 与补偿取消时，机床移动到补偿被取消的位置，补偿量被记忆住。为了不记忆补偿量，请在其他节中进行指令。
- (注 3) 在自动运转中，即使通过 MDI 等变更当前选中的补偿编号的补偿量，只要不再次执行同一编号的 T 指令，则变更后的补偿量不生效。
- (注 4) 刀具长度补偿、刀具刀刃磨损补偿量因重新启动、紧急停止而被清除。也可通过参数“#1099 Treset”予以保持。

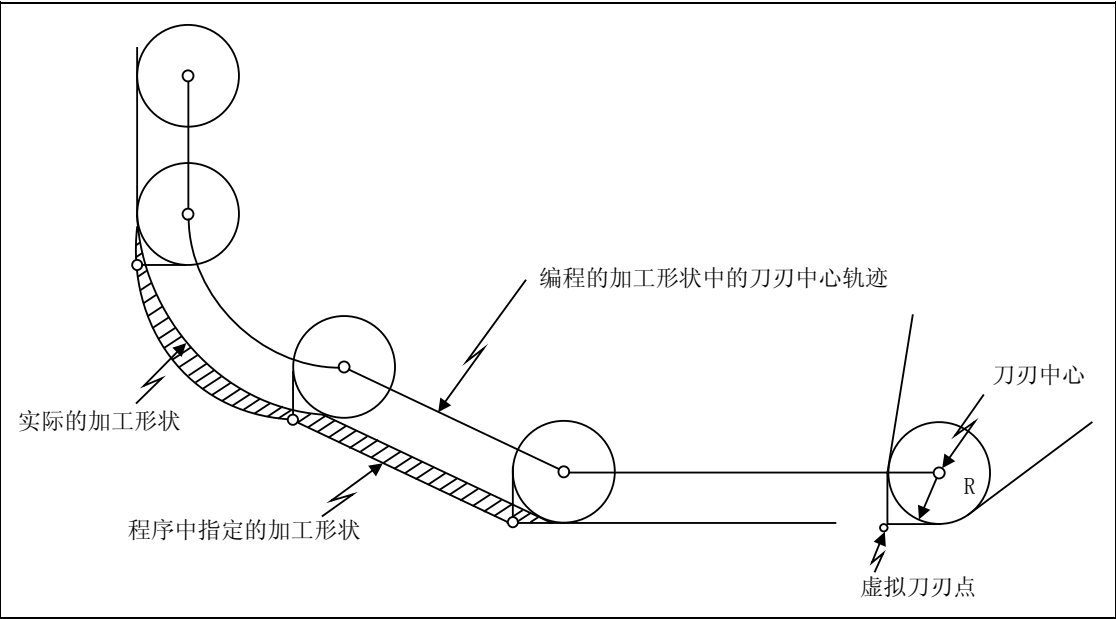
12.4 刀尖R补偿；G40、G41、G42、G46



功能及目的

由于刀具刀刃一般带有弧度，所以以虚拟刀刃点作为刀具的顶端，进行编程。因此，在进行锥轴切削及圆弧切削时，编程形状与切削形状之间，存在因刀具刀刃的弧度而导致的误差。刀尖R补偿是通过设定刀尖R值，自动计算误差，进行补偿的功能。

通过指定代码，可选择是固定补偿方向，还是自动判别补偿方向。

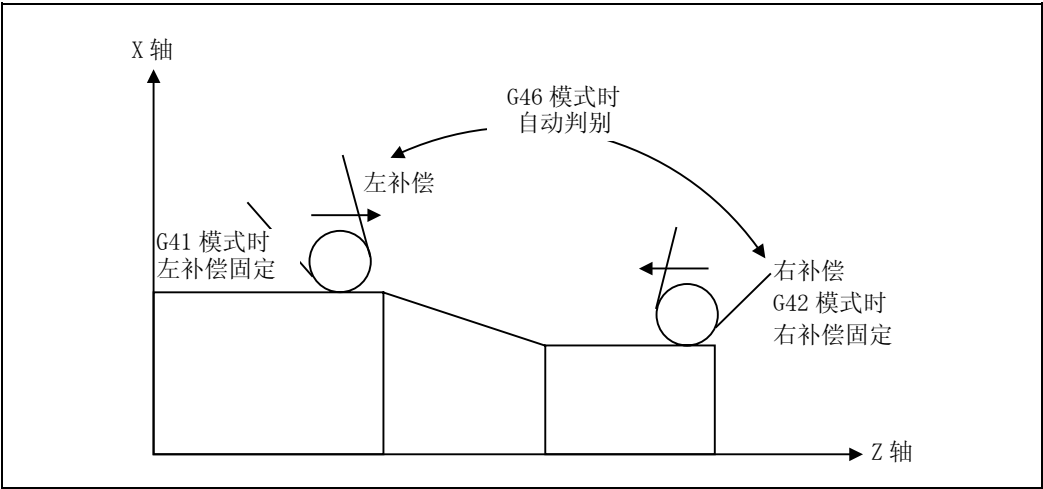


功能及指令格式

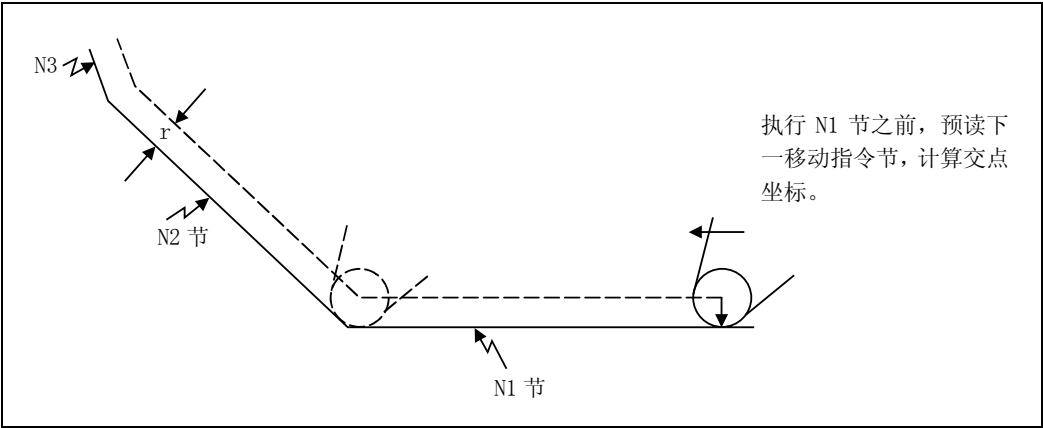
代码	功能	指令格式
G40	刀尖 R 补偿模式 取消	G40 (Xx/UuZz/WwIi Kk) ;
G41	刀尖 R 补偿 左模式打开	G41 (Xx/UuZz/Ww)
G42	刀尖 R 补偿 右模式打开	G42 (Xx/UuZz/Ww)
G46	刀尖 R 补偿自动方向判别模式打开	G46 (Xx/UuZz/Ww)

(注 1) 通过 G46 进行的刀尖 R 补偿，是根据预先设定的虚拟刀刃点与加工程序的移动指令，自动判别补偿方向，进行刀尖 R 补偿。

(注 2) G40 为取消刀尖 R 补偿模式。



(注 3) 刀尖 R 补偿，是预读以下 2 个移动指令的节数据（无移动指令时，最多 5 节），根据交点计算公式，将刀尖 R 刀刃中心的轨迹控制在距程序轨迹偏移了刀尖 R 半径后的轨迹上。



- (注 4) 在上图中， r 为刀尖 R 补偿量（刀尖 R 半径）。
- (注 5) 刀尖 R 补偿量是与刀具长度编号相对应，预先与刀刃点同时设定。
- (注 6) 当连续 5 节中，有 4 节以上没有移动量的节时，会发生过切或切入不足。
但是，可选节停止有效的节被跳跃。
- (注 7) 对于固定循环（G77~G79）、粗切削循环（G70、G71、G72、G73），刀尖 R 补偿也有效。
但是，粗切削循环是在取消进行刀尖 R 补偿的精整形状的状态下进行切削，结束之后，自动恢复到补偿模式。
- (注 8) 对于螺纹切削指令，提前 1 节临时取消。
- (注 9) 在刀尖 R 补偿（G46）中，可指定刀尖 R 补偿（G41/G42）。此时，无需通过 G40 取消补偿。
- (注 10) 补偿平面、移动轴、下一前进方向矢量取决于由 G17 到 G19 所指定的平面选择指令。

- G17.....XY平面 X、Y、I、J
- G18.....ZX平面 Z、X、K、I
- G19.....YZ平面 Y、Z、J、K

12.4.1 刀刃点与补偿方向

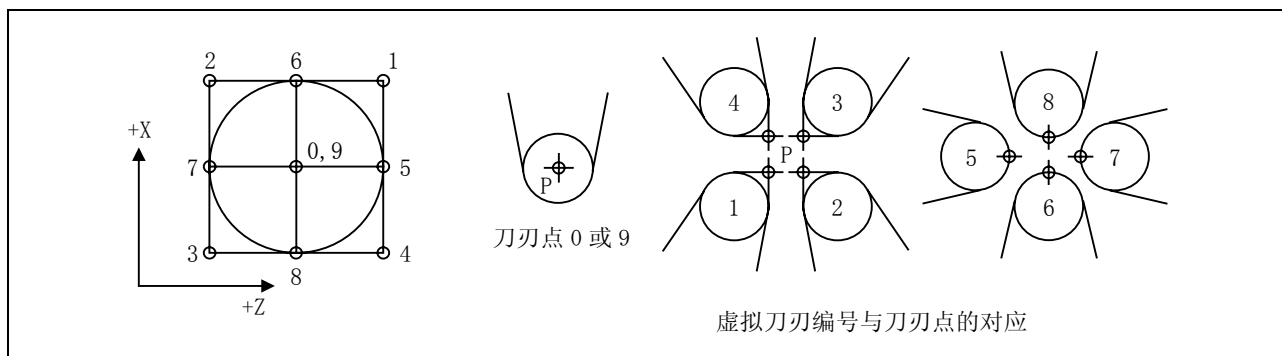


刀刃点

由于刀具的刀刃一般为弧形，所以，程序上的刀刃位置如下图的图例所示，对准P点。

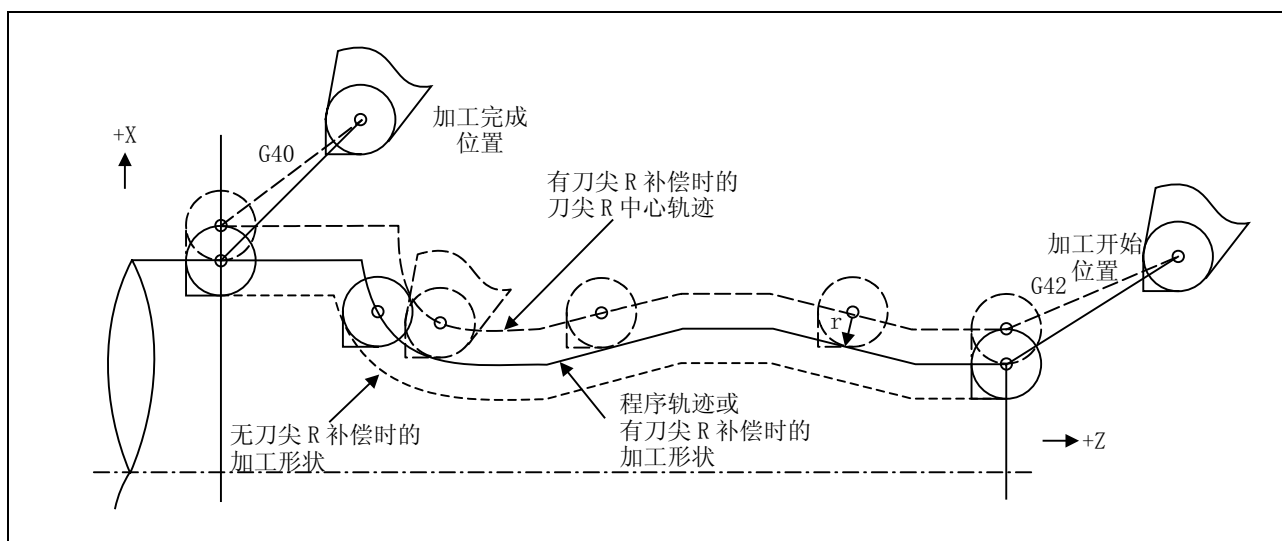
刀尖R补偿中，针对每一刀具长度编号，从下图中选择1点，预先设定位置关系。

（在G46模式中选择1~8，G41/G42模式中选择0~9。）

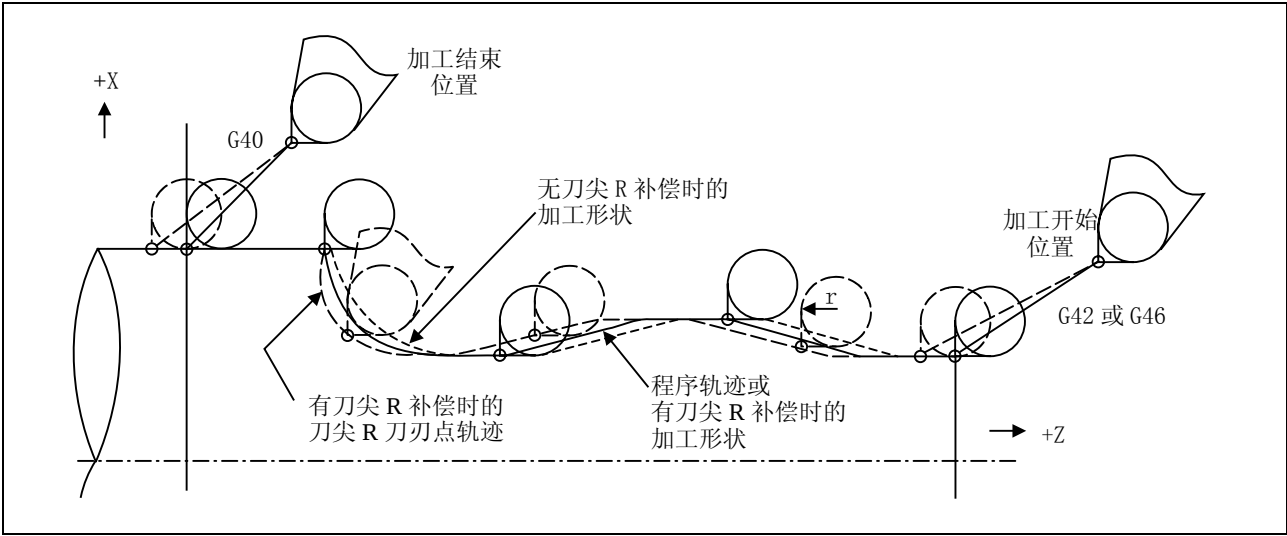


刀刃点与补偿动作

（1）将刀尖R中心对准加工开始位置进行加工时



(2) 将刀刃点对准加工开始位置进行加工时



补偿方向

- (1) G41/G42指令的补偿方向取决于G41/G42代码，而G46指令中的补偿方向，根据刀刃点与指定移动矢量的关系，按照下表自动决定。
- (2) 开始刀尖R补偿，最初的移动矢量（含G0）为表中的×标记时，由于无法确定补偿方向，所以通过下一移动矢量决定补偿方向。当即使预读了5节也无法决定补偿方向时，发生程序错误“P156”。
- (3) 刀尖R补偿中，当补偿方向翻转时，除非通过G00节翻转，否则发生程序错误“P157”。但是，即使G28、G30、G53节之前和之后的补偿方向不同，由于将补偿临时取消，所以不会发生错误。另外，也可通过参数（#8106 避免G46翻转轴错误），保持同一补偿方向，进行动作。

(4) 在刀尖R补偿中，补偿方向为下表中的×标记方向时，取决于之前的补偿方向。

G46中，根据刀刃点与移动矢量决定补偿方向的方法

刀刃的 前进方向 ↓ 刀刃的 补偿方向		刀 刃 点								刀刃的 补偿方向 ↓ 刀刃的 前进方向	
		①	②	③	④	⑤	⑥	⑦	⑧		
移动矢量 (刀刃点1~4) ↓		右	右	左	左	×	右	×	左	→	移动矢量 (刀刃点5~8) ↓
		×	右	×	左	左	右	右	左		
		左	右	右	左	左	×	右	×	↑	
		左	×	右	×	左	左	右	右		
		左	左	右	右	×	左	×	右		
		×	左	×	右	右	左	左	右		
		右	左	左	右	右	×	左	×	↓	
		右	×	左	×	右	右	左	左		

(注 1) 表中的×标记表示根据指令的移动矢量刀刃点决定补偿方向。

(注 2) 表中的标记表示 45° 方向的移动矢量。(其他移动矢量以此为准。)

(注 3) 表中的标记表示大于 45° 小于 135° 范围内的矢量。(其他移动矢量以此为准。)

(例) 刀刃点 3，移动矢量为 Z 轴 (-) 方向时 (移动矢量)

如左图所示，根据刀刃位置与刀刃移动方向，被切削对象位于 X 轴 (-) 端。因此，补偿方向为朝向刀刃前进方向，变为被切削对象的右侧补偿。

12.4.2 刀尖R补偿的动作



刀尖 R 补偿取消状态

在以下条件下，刀尖R补偿进入补偿取消模式。

- (1) 接通电源后
- (2) 按下设定显示装置的重新启动按钮后
- (3) 执行带重新启动功能的M02、M30后
- (4) 执行补偿取消指令（G40）之后
- (5) 选择刀具编号0（执行T00）之后

在补偿取消模式中，补偿矢量为0，刀刃点轨迹与程序轨迹一致。
请务必让包含刀尖R补偿的程序，在补偿取消状态下结束。



开始刀尖 R 补偿（起动）

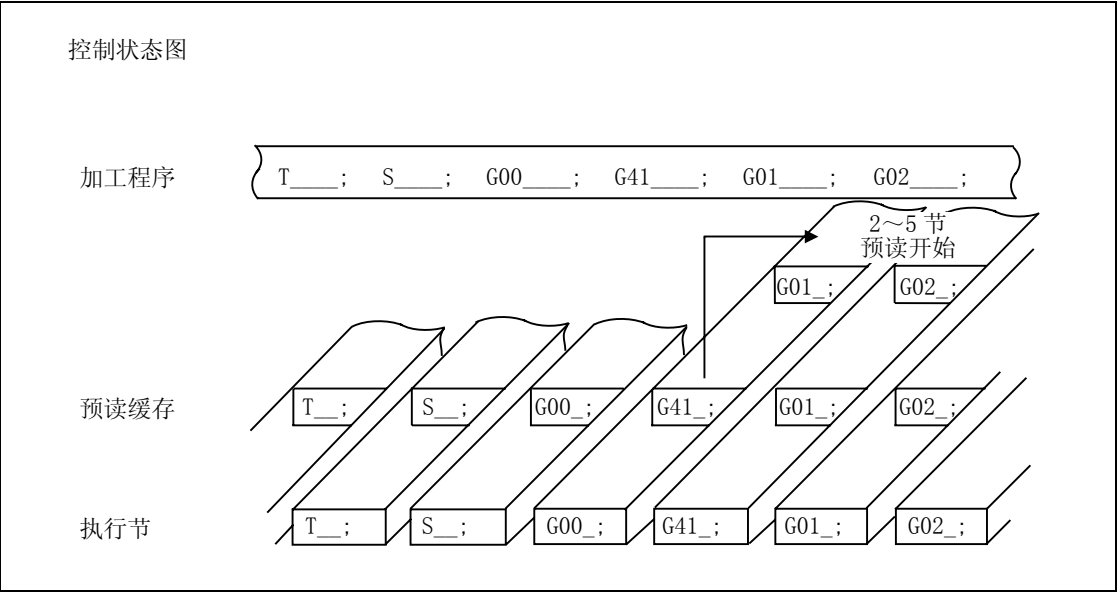
在补偿取消状态下，当满足以下的所有条件时，开始刀尖R补偿。

- （1）G41、G42或G46为指令后的移动指令。
- （2）除圆弧指令以外的移动指令。

在补偿开始时，不管是连续运转还是单节运转，由于交点运算的缘故，连续读入2~5节之后，再开始执行。

（在移动指令中，如果连续2节没有移动指令，则最多预读5节。）

另外，在补偿模式中，也同样预读最多5节，进行补偿运算。



补偿开始动作中，包括类型A与类型B两种。
通过参数“#1229 set01/bit2”设定选择哪一种类型。
另外，这一类型与补偿取消动作的类型相同。

类型	说明
类型 A	在刀尖 R 补偿及半径补偿中，执行起动、取消指令动作时，不将起动、取消指令节作为交点运算处理的对象，而是作为指令的直角方向上的偏移矢量。
类型 B	在刀尖 R 补偿半径补偿中，执行起动、取消指令动作时，进行指令节与下一指令节的交点运算处理。

在以下的说明图中，S表示单节停止点。



刀尖 R 补偿的开始动作

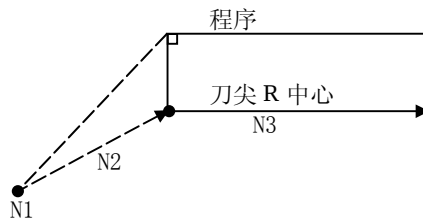
开始刀尖R补偿时，单独的G41/G42/G46指令，只移动刀尖R补偿量。G00指令无需进行刀尖R补偿。通过G01、G02、G03指令进行刀尖R补偿。但是，即使有轴指令，如果不进行移动，则不进行刀尖R补偿。

(1) 在转角内侧单独指定G41/G42/46指令时

```

    }
N1 G42;
N2 G00 X_Z_;
N3 G01 X_Z_F_;
    }

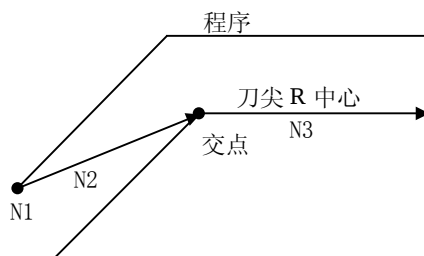
```



```

    }
N1 G42;
N2 G01 X_Z_F_;
N3 G01 X_Z_;
    }

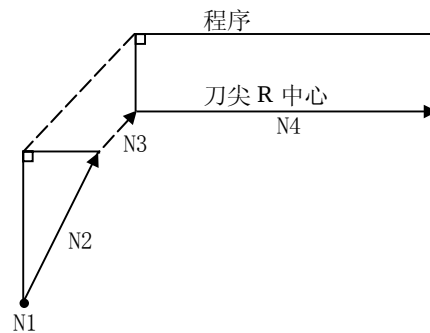
```



```

    }
N1 G42;
N2 G01 X_Z_F_;
N3 G00 X_Z_;
N4 G01 X_Z_;
    }

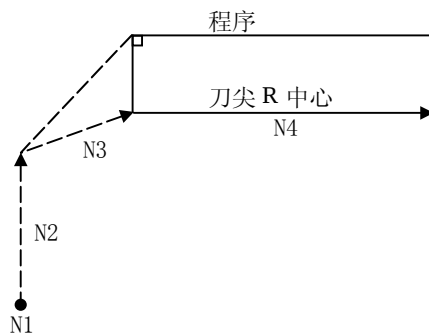
```



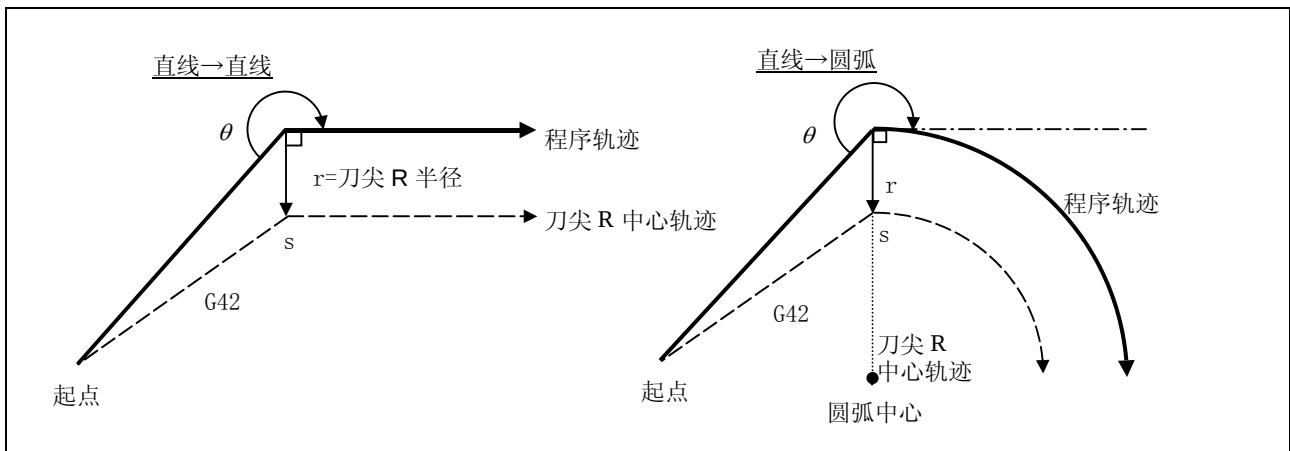
```

    }
N1 G42;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G01 X_Z_F_;
    }

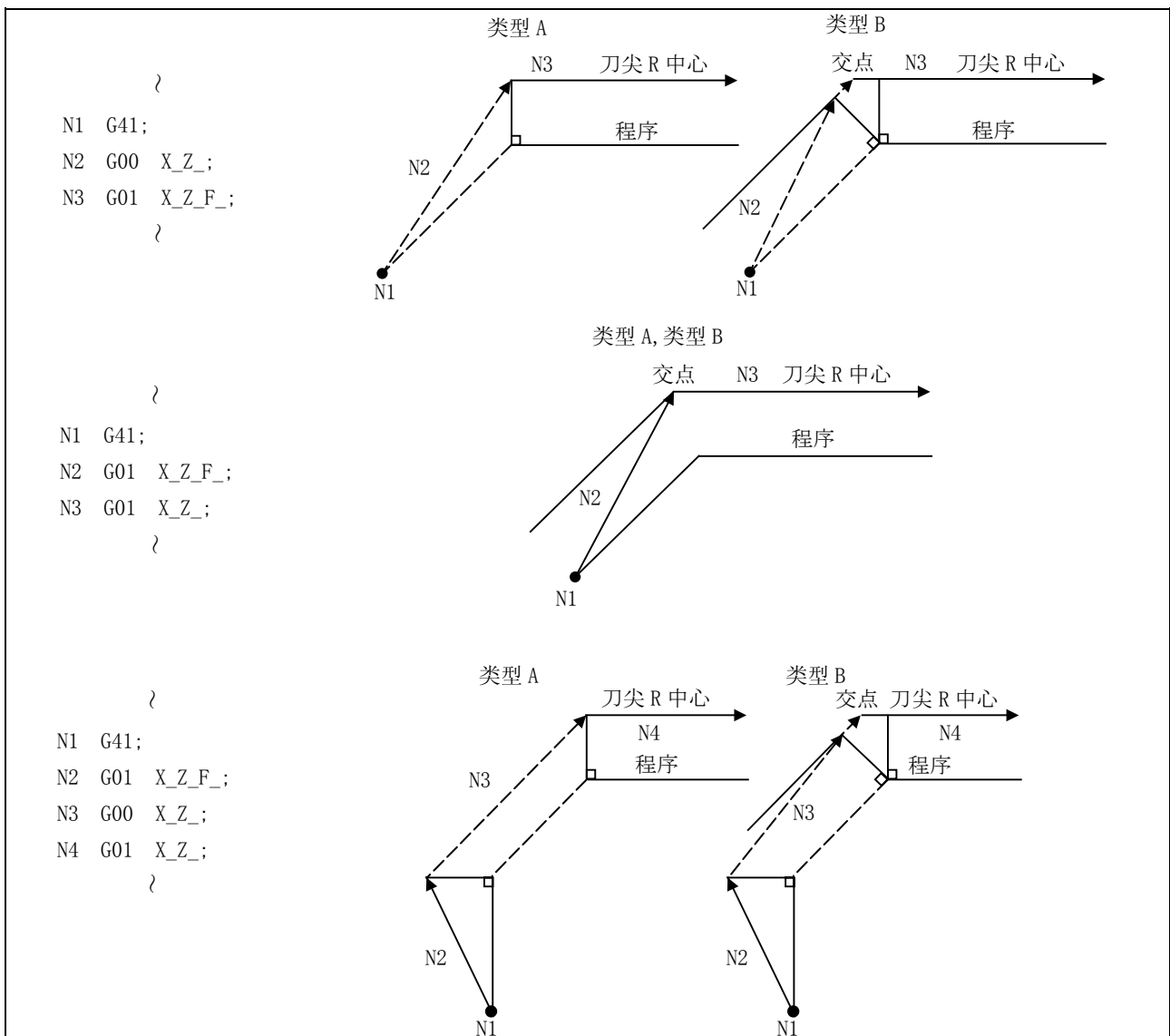
```



(2) 在同一节中，在转角内侧同时指令G41/G42/46与移动指令时



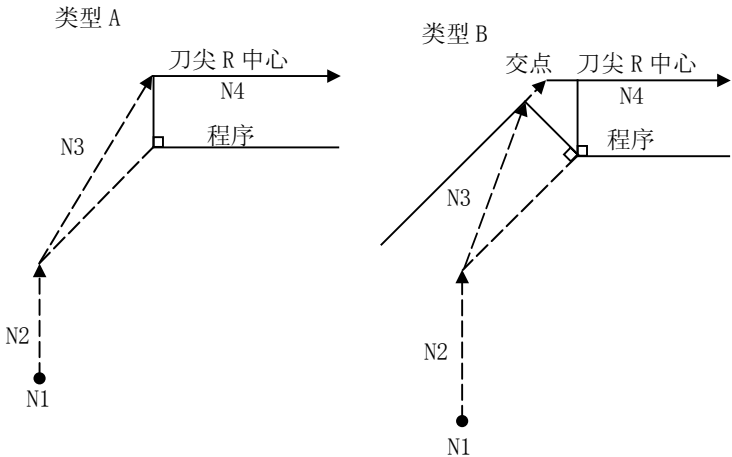
(3) 在转角外侧（钝角）单独指定G41/G42/46指令时



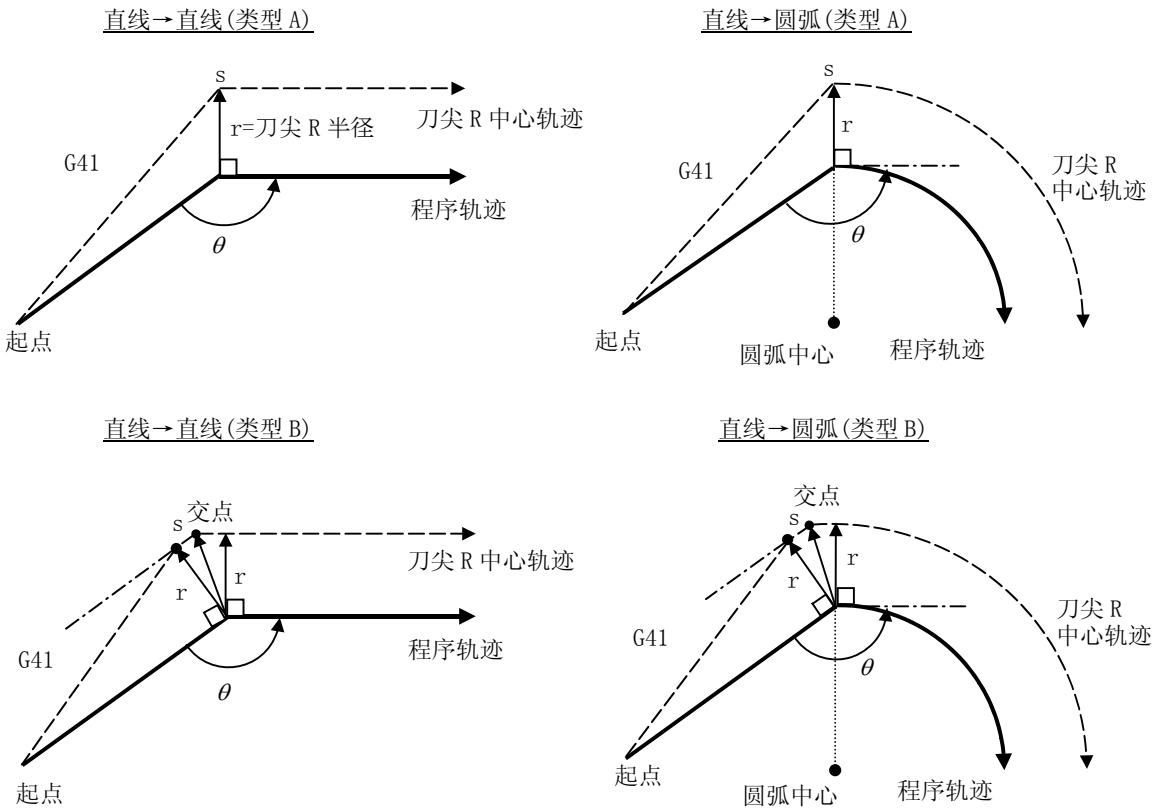
```

}
N1 G41;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G01 X_Z_F_;
}

```



(4) 在同一节中，在转角外侧（钝角）同时执行G41/G42/G46与移动指令时 [$90^{\circ} \leq \theta < 180^{\circ}$]

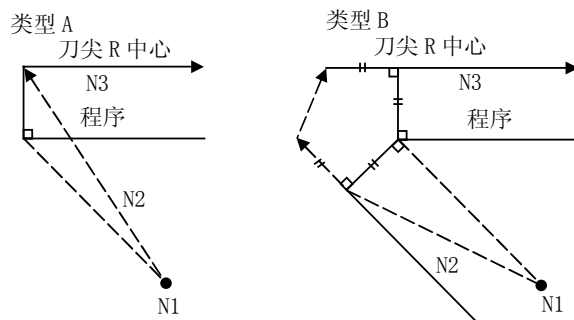


(5) 在转角外侧（锐角）单独指定G41/G42/46指令时

```

    }
N1 G41;
N2 G00 X_Z_;
N3 G01 X_Z_F_;
    }

```

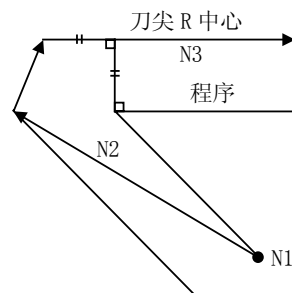


```

    }
N1 G41;
N2 G01 X_Z_F_;
N3 G01 X_Z_;
    }

```

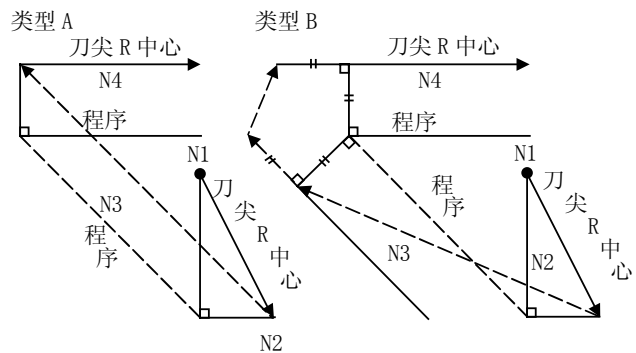
类型 A, 类型 B



```

    }
N1 G41;
N2 G01 X_Z_F_;
N3 G00 X_Z_;
N4 G01 X_Z_;
    }

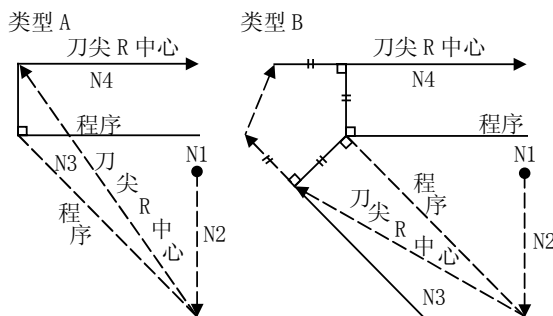
```



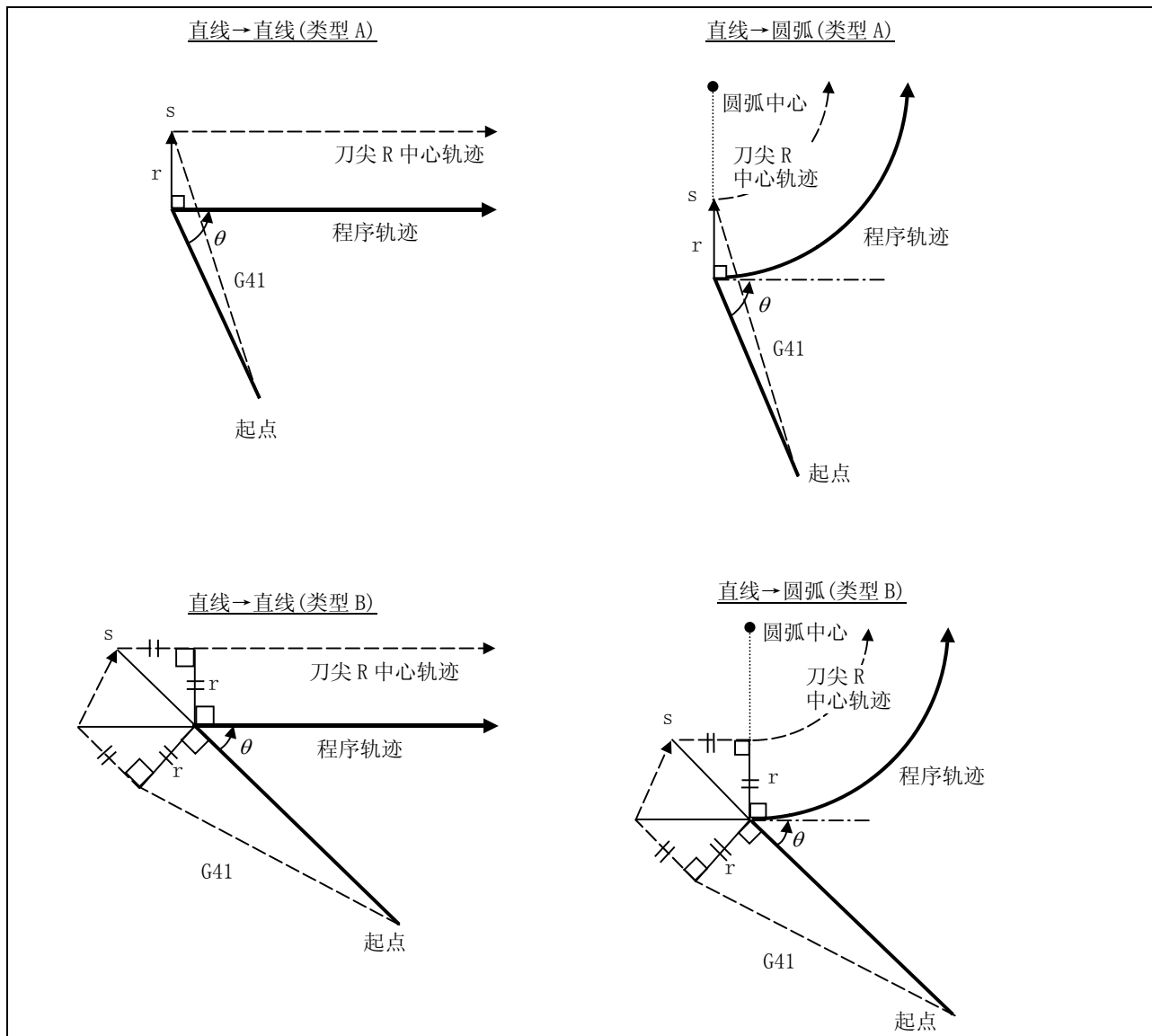
```

    }
N1 G41;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G01 X_Z_F_;
    }

```



(6) 在同一节中，在转角外侧（锐角）同时执行G41/G42/G46与移动指令时 [$\theta < 90^\circ$]



(注 1) 在同一节内没有轴移动指令时，可在下一节的垂直方向上进行补偿动作。



补偿模式中的动作

对于程序轨迹 (G00、G01、G02、G03)，根据直线/圆弧计算刀具轨迹，进行补偿。

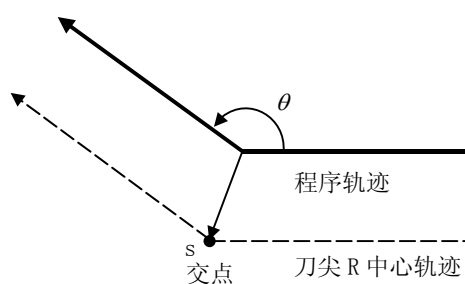
在刀尖R补偿 (G41、G42、G46) 模式中，即使指令了相同的刀尖R补偿指令 (G41、G42、G46)，也会被忽略。

如果在补偿模式中连续指令了4节以上无移动的节，则会发生过切或切入不足。

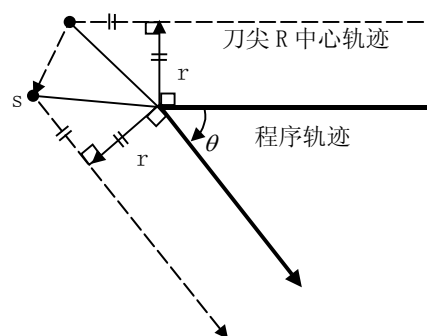
另外，在刀尖R补偿中，当指令了M00时，禁止预读。

(1) 旋转转角外侧时

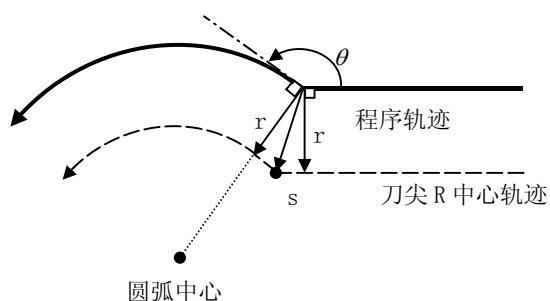
直线→直线 ($90^\circ \leq \theta < 180^\circ$)



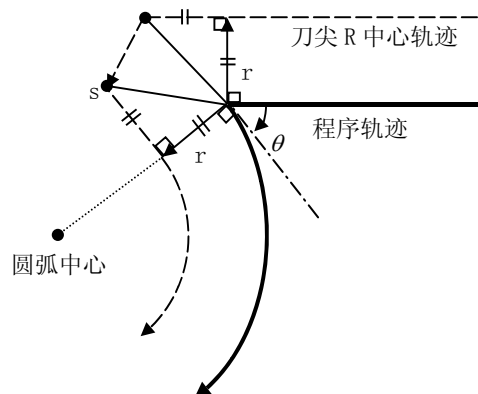
直线→直线 ($0^\circ < \theta < 90^\circ$)

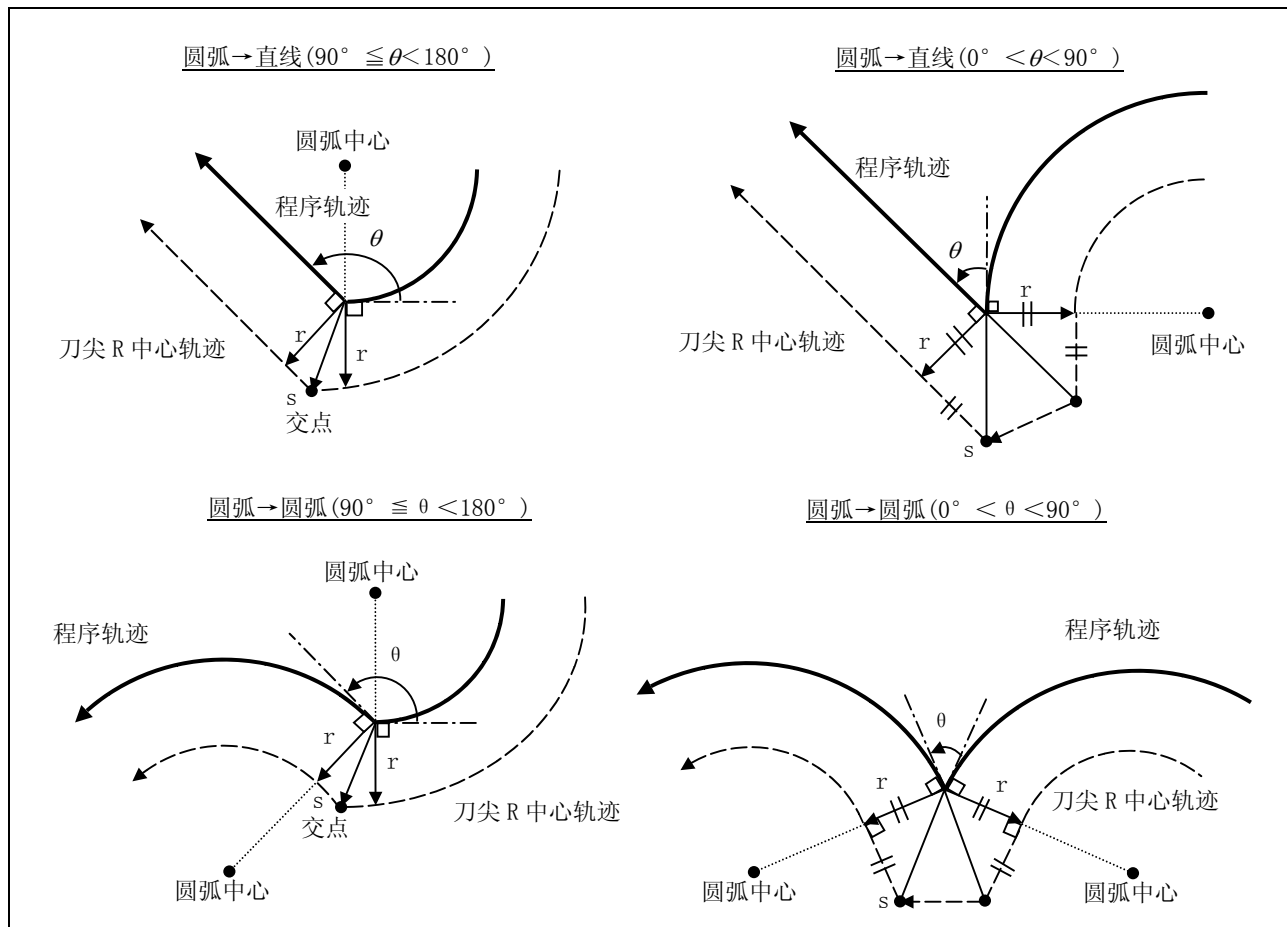


直线→圆弧 ($90^\circ \leq \theta < 180^\circ$)



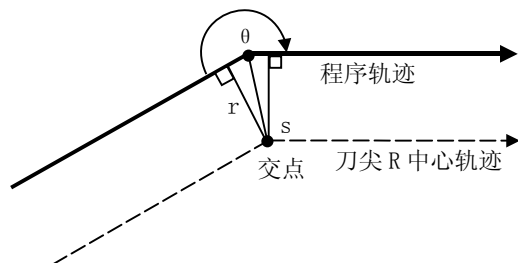
直线→圆弧 ($0^\circ < \theta < 90^\circ$)



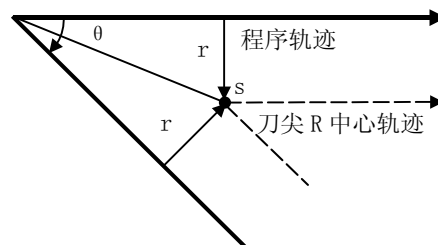


(2) 旋转转角内侧时

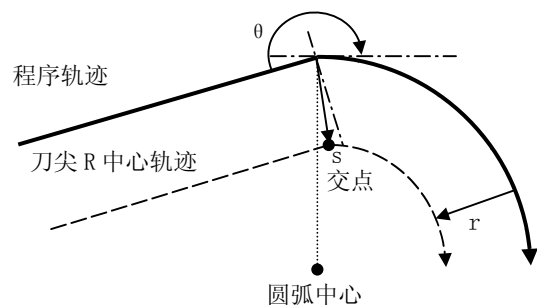
直线→直线(钝角)



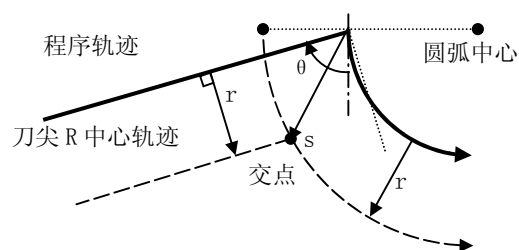
直线→直线(锐角)



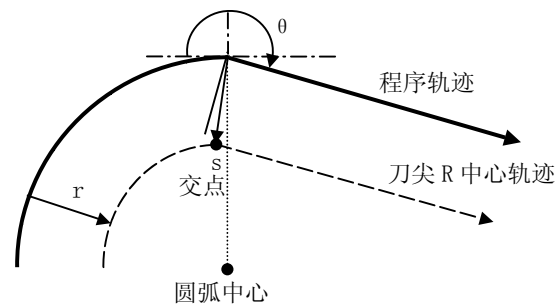
直线→圆弧(钝角)



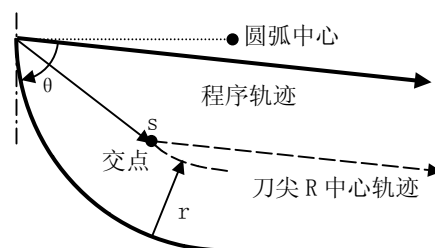
直线→圆弧(锐角)

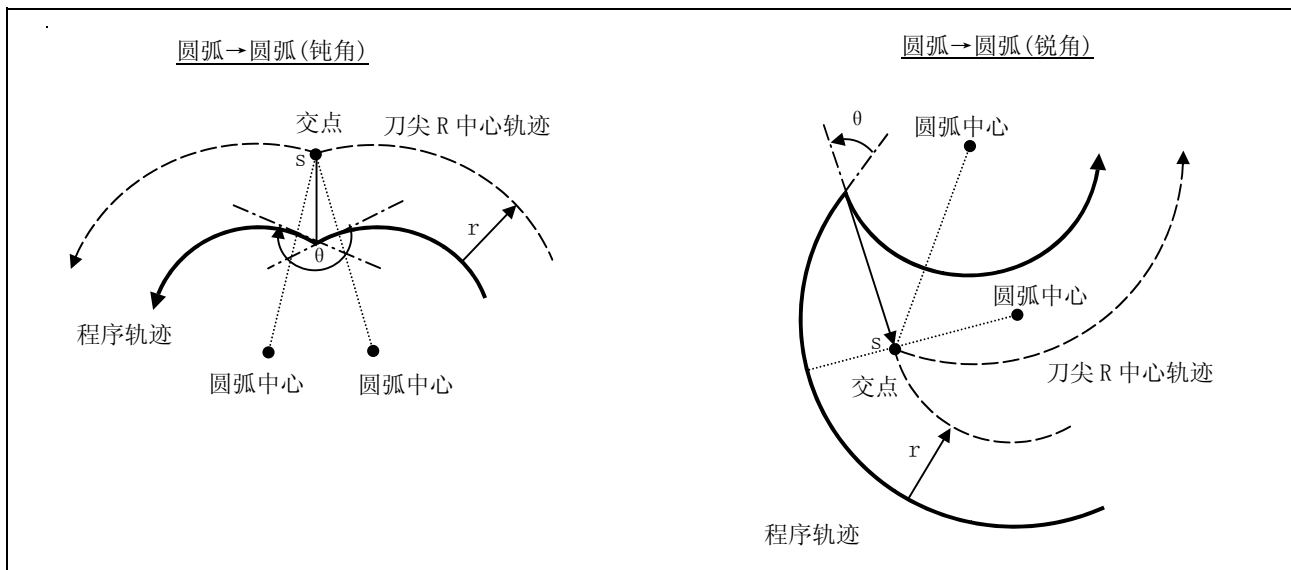


圆弧→直线(钝角)



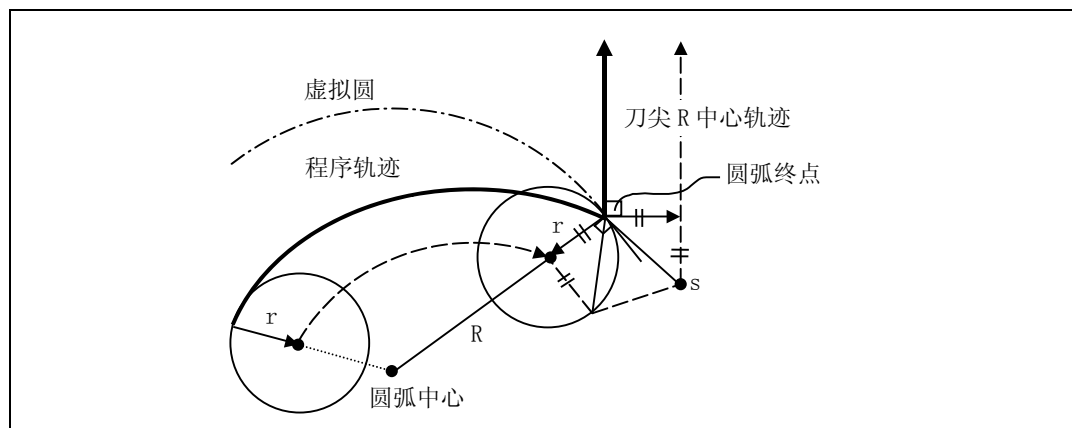
圆弧→直线(锐角)





(3) 当圆弧的终点不在圆弧上时

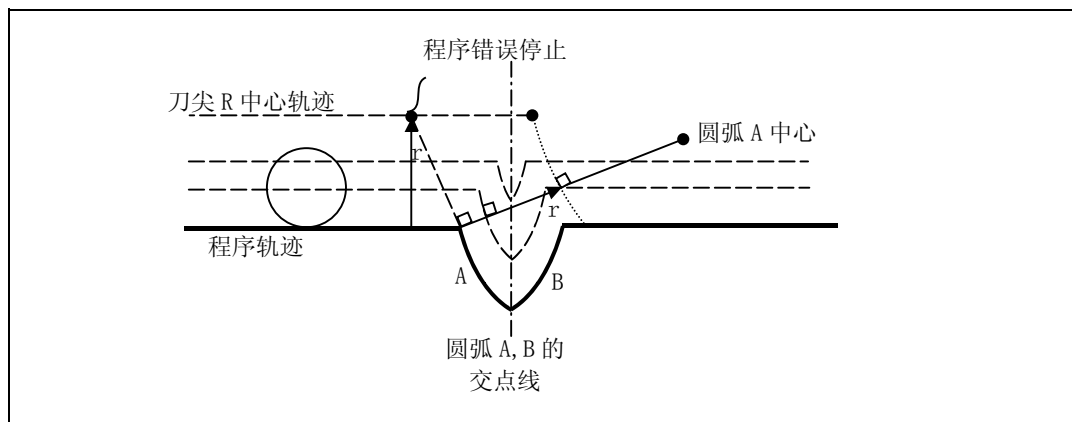
如果补偿后的误差在参数#1084 RadErr以内，则使用螺旋状圆弧在起点到终点之间进行插补。



(4) 当不存在内侧交点时

如下图所示的场合，根据补偿量不同，可能会不存在圆弧A、圆弧B的交点。

此时，则在上一节的终点显示程序错误“P152”，停止运行。



**取消刀尖 R 补偿**

在刀尖R补偿模式中，满足以下任何一个条件，则刀尖R补偿被取消。

但是此时，必须是圆弧指令以外的移动指令。

在圆弧指令中指令了取消补偿，则发生程序错误“P151”。

(1) 执行了G40指令。

(2) 执行了刀具编号T00。

在读入补偿取消指令后，进入取消模式，中止预读5节，变为预读1节。

**刀尖 R 补偿的取消动作**

当有刀尖R取消指令时，如下。

(1) 刀尖R补偿结束时，仅G40指令，当G40之前为G00时，保持暂时取消刀尖R补偿的状态，取消刀尖R补偿。

(2) 刀尖R补偿结束时，单独执行G40指令，当在G40指令之前为插补指令时，不取消刀尖R补偿，以确保刀尖R中心停止在垂直位置上。在G40以后的首个轴移动指令上，取消刀尖R补偿。即使有轴指令，如果没有移动指令，则不取消刀尖R补偿。在G40之后没有轴移动指令，通过M02等结束程序之后，保持进行刀鼻补偿的状态，进行重新启动，则虽然刀尖R补偿被取消，但是不进行取消动作。

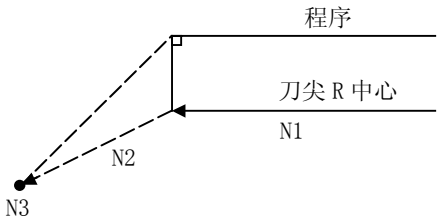
(3) 对于单独的T00指令，在该节进入刀尖R取消模式，移动到刀尖R取消位置。

(4) 转角的内侧/外侧与取消的关系

(a) -1 在转角内侧单独指令G40指令时

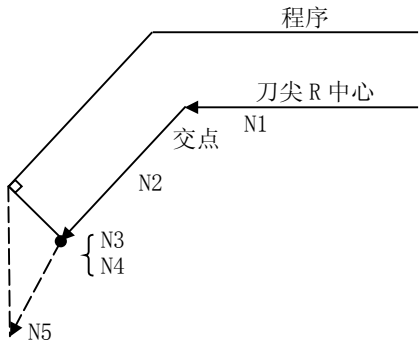
```

    }
N1  G01  X_Z_F_;
N2  G00  X_Z_;
N3  G40;
    }
```



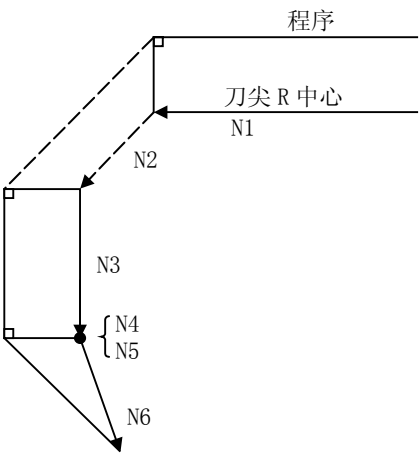
```

    }
N1  G01  X_Z_F_;
N2  G01  X_Z_;
N3  G40;
N4  M05;
N5  G00  X_Z_;
    }
```



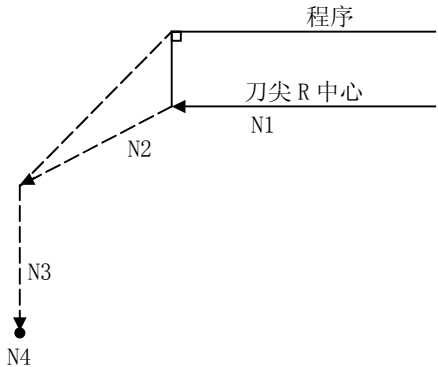
```

    }
N1  G01  X_Z_F_;
N2  G00  X_Z_;
N3  G01  X_Z_;
N4  G40;
N5  M05;
N6  G01  X_Z_;
    }
```

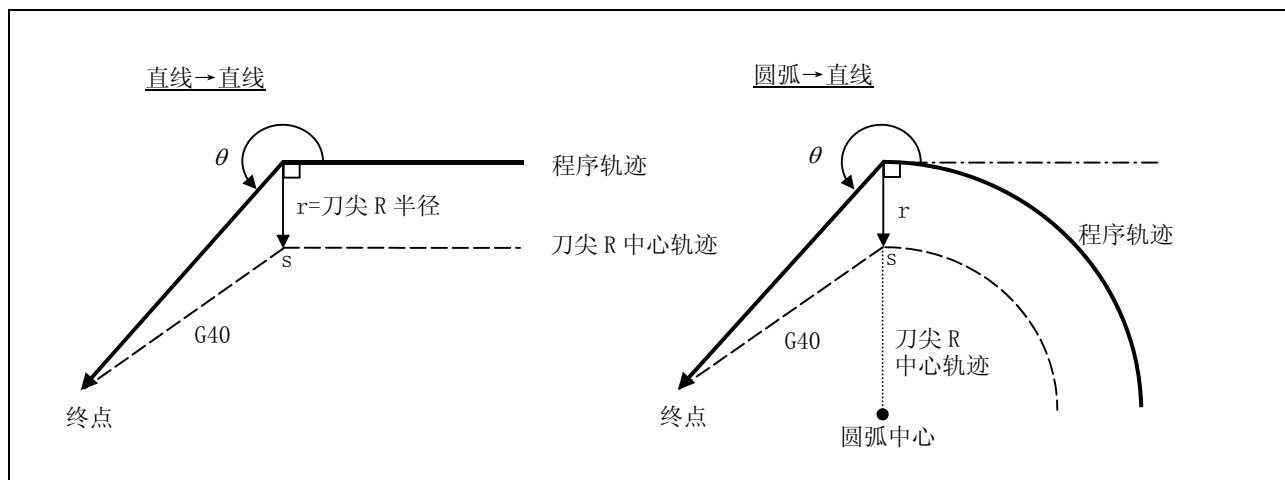


```

    }
N1  G01  X_Z_F_;
N2  G00  X_Z_;
N3  G00  X_Z_;
N4  G40;
    }
```



(a) -2 在转角内侧，在同一节内指令G40指令与移动指令时



(b) -1 在转角外侧（钝角）单独指令G40指令时

```

    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G40;
    }

```

```

    }
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G40;
N4 M05;
N5 G00 X_Z_;
    }

```

```

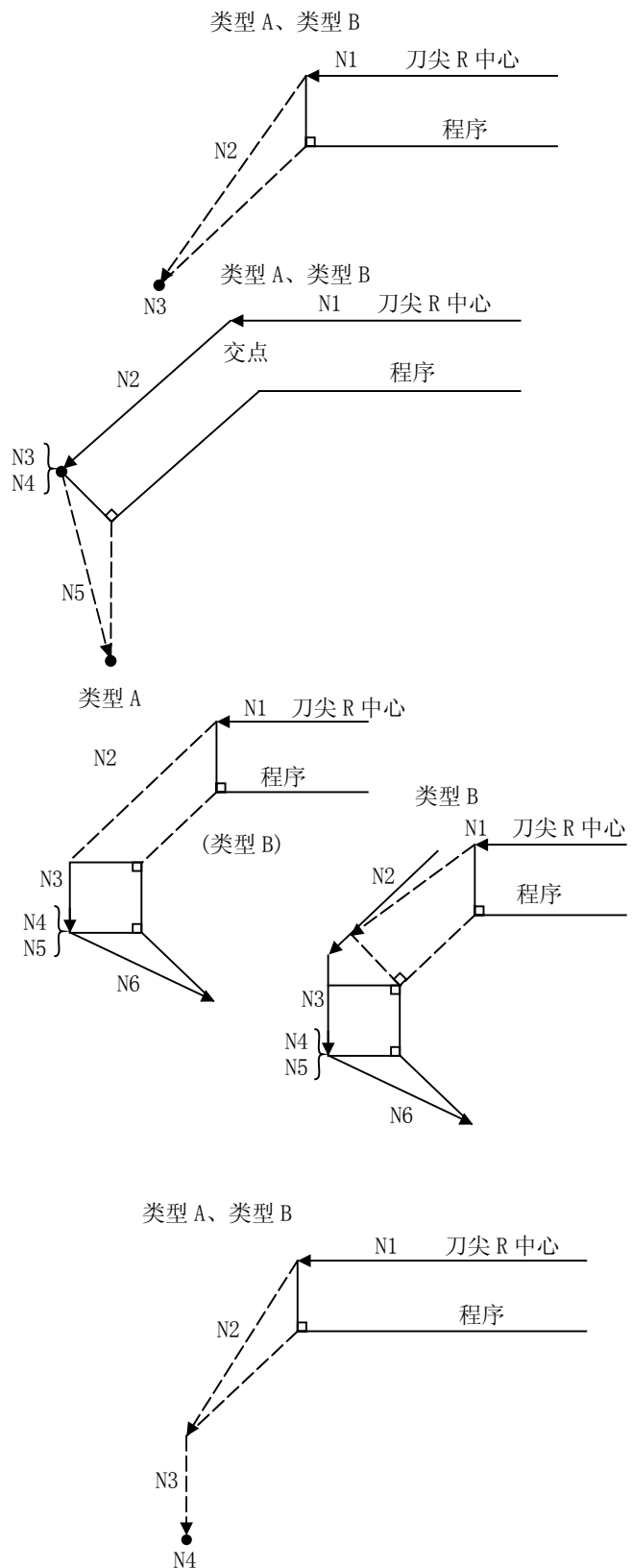
    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G01 X_Z_;
N4 G40;
N5 M05;
N6 G01 X_Z_;
    }

```

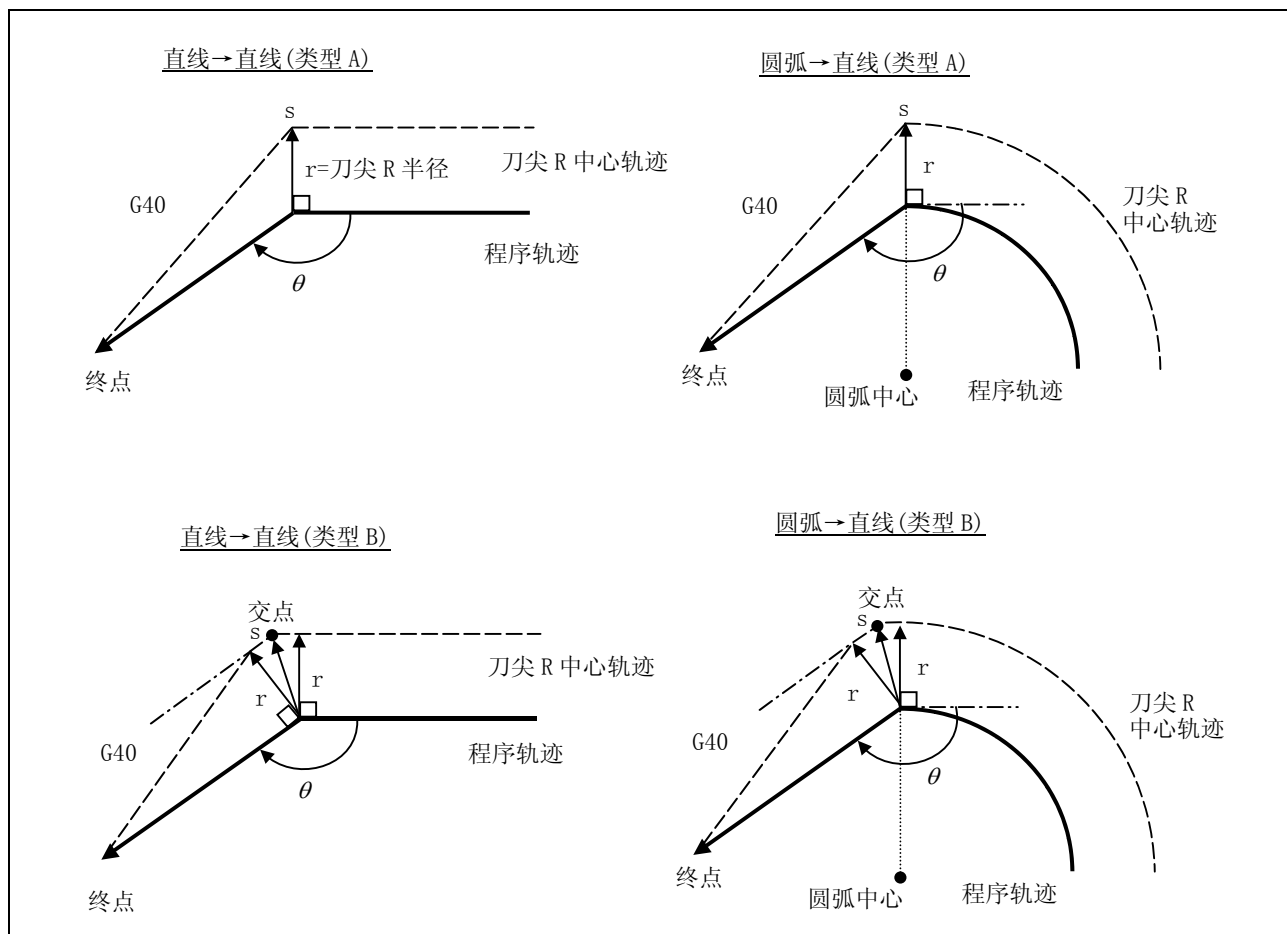
```

    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G40;
    }

```



(b) -2 在转角外侧（钝角），在同一节内指令G40指令与移动指令时



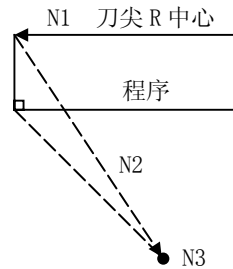
(c) -1 在转角外侧（锐角）单独指令G40指令时

```

    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G40;
    }

```

类型 A、类型 B

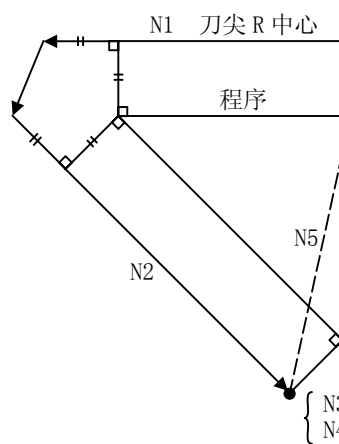


```

    }
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G40;
N4 M05;
N5 G00 X_Z_;
    }

```

类型 A、类型 B

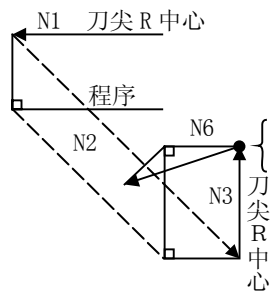


```

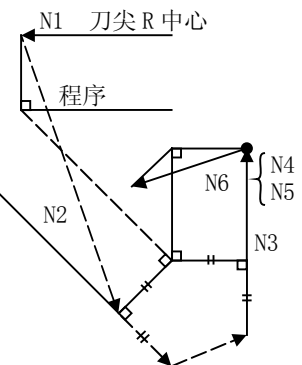
    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G01 X_Z_;
N4 G40;
N5 M05;
N6 G01 X_Z_;
    }

```

类型 A



类型 B

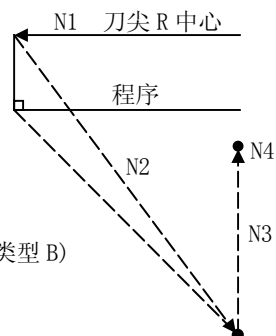


```

    }
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G40;
    }

```

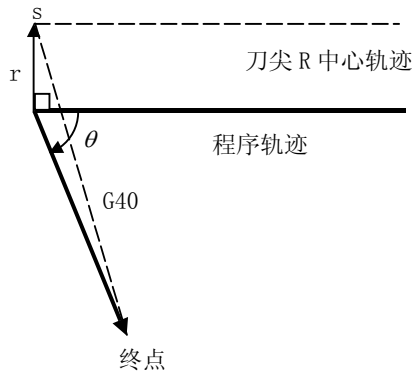
类型 A、类型 B



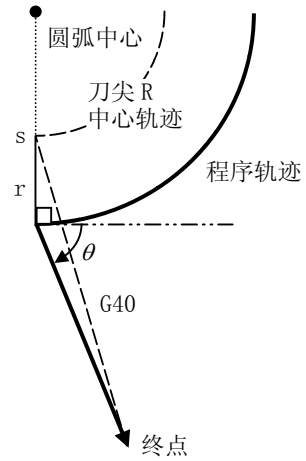
(类型 A、类型 B)

(c) -2 在转角外侧（锐角），在同一节内指令G40指令与移动指令时

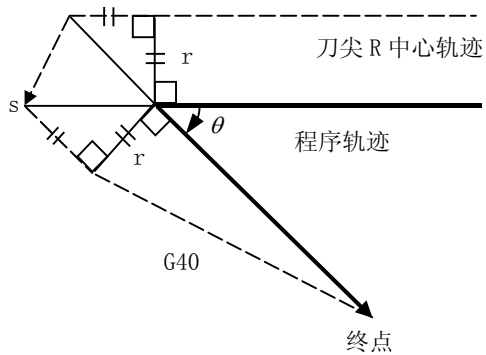
直线→直线(类型 A)



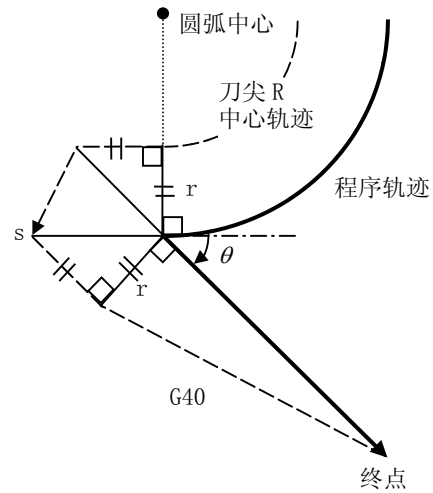
圆弧→直线(类型 A)



直线→直线(类型 B)



圆弧→直线(类型 B)



12.4.3 刀尖R补偿中的其他动作

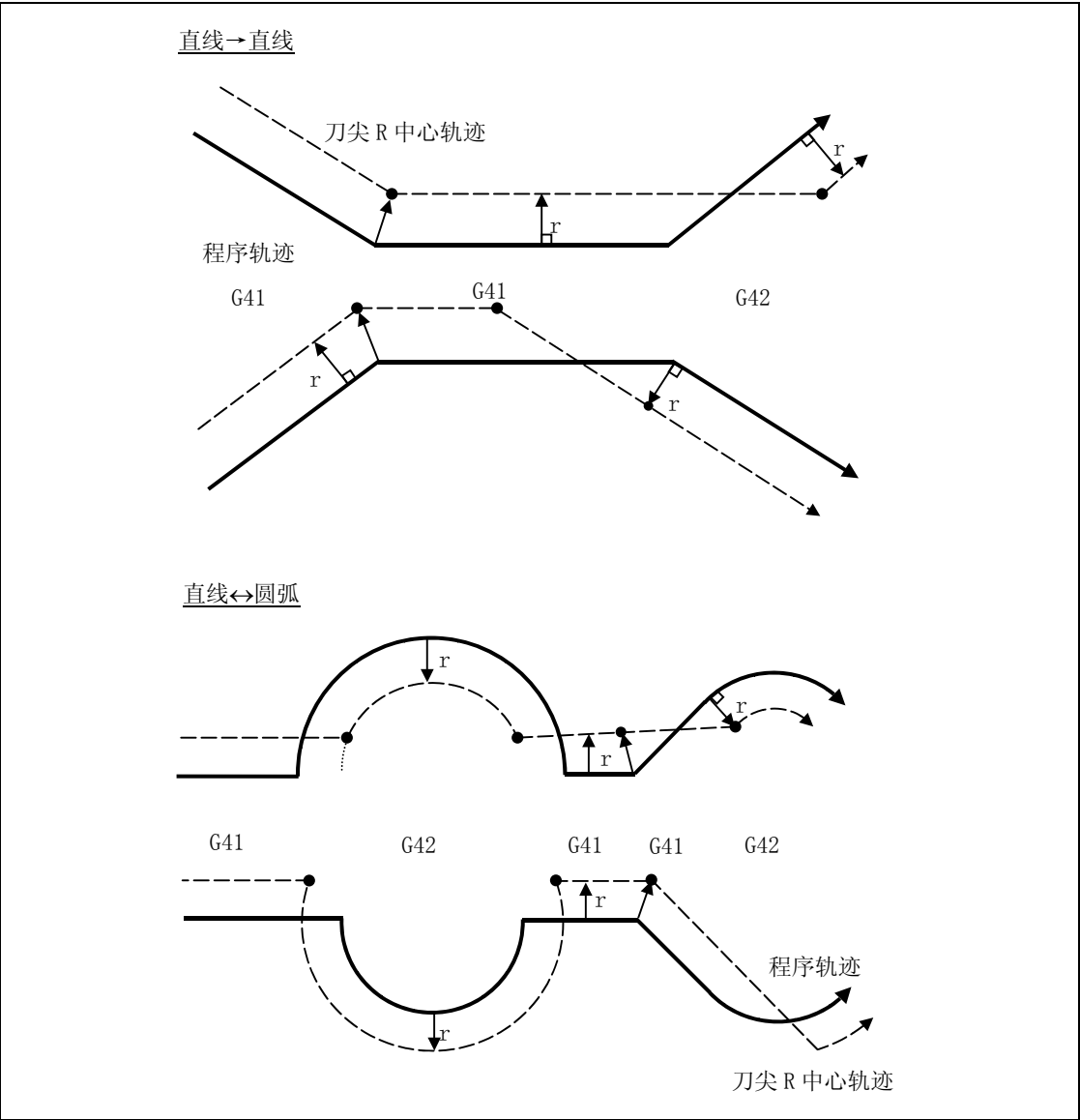


刀尖 R 补偿中的补偿方向变更

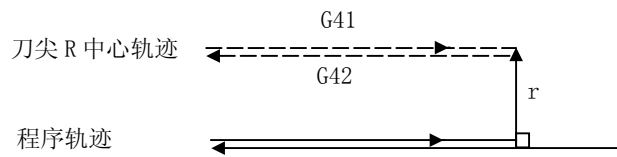
补偿方向取决于刀尖R补偿指令（G41、G42）。

G 代码	补偿方向
G41	左侧补偿
G42	右侧补偿

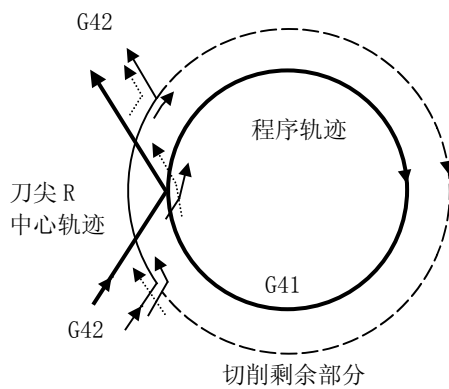
在补偿模式中，不指令补偿取消就变更补偿指令，则可变更补偿方向。
但是，在补偿开始节与下一节，无法进行变更。



沿直线往复时



根据补偿方法，圆弧为 360° 以上时



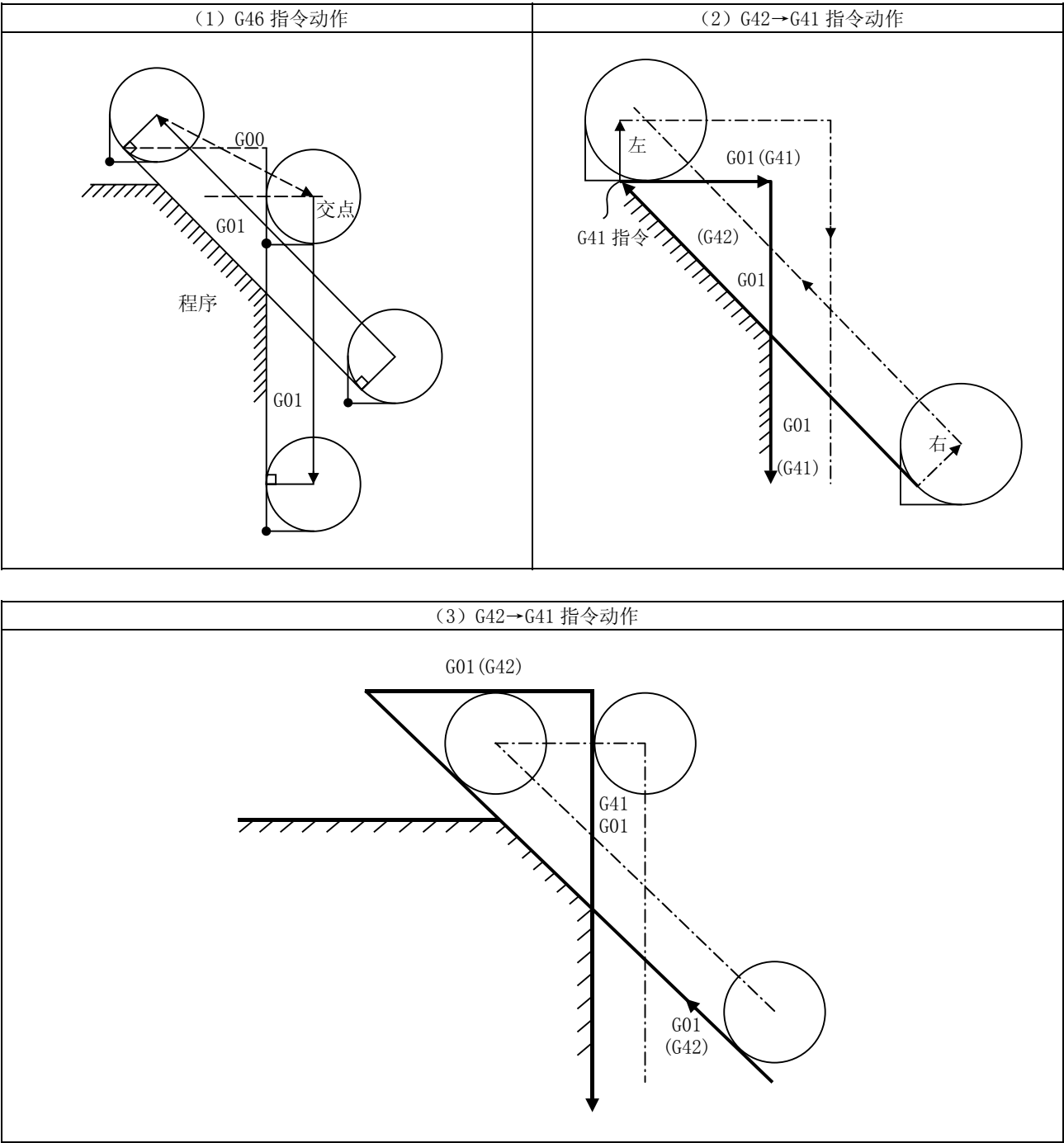
在下述场合下，圆弧可能会超过 360° 。

a. 随着 G41/G42 的切换，切换补偿方向。

此时，如左图所示进行补偿，产生切削剩余部分。



利用 G46/G41/G42 进行闭合轨迹的刀尖 R 补偿

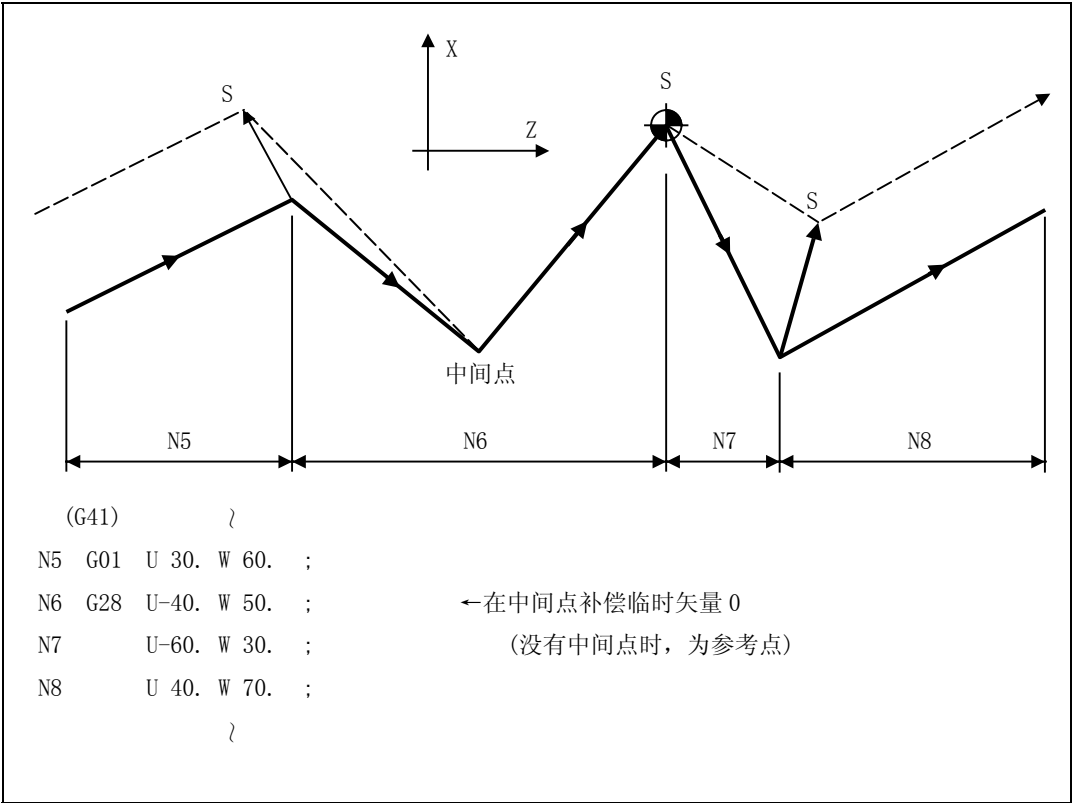




暂时取消补偿矢量指令

在补偿模式中，发出以下指令，则临时性的清除补偿矢量，之后，自动返回补偿模式。
此时，不进行补偿取消动作，在交点矢量之后，立即移动到无矢量点，也就是程序的指令点。返回补偿模式时，也立即移动到交点。

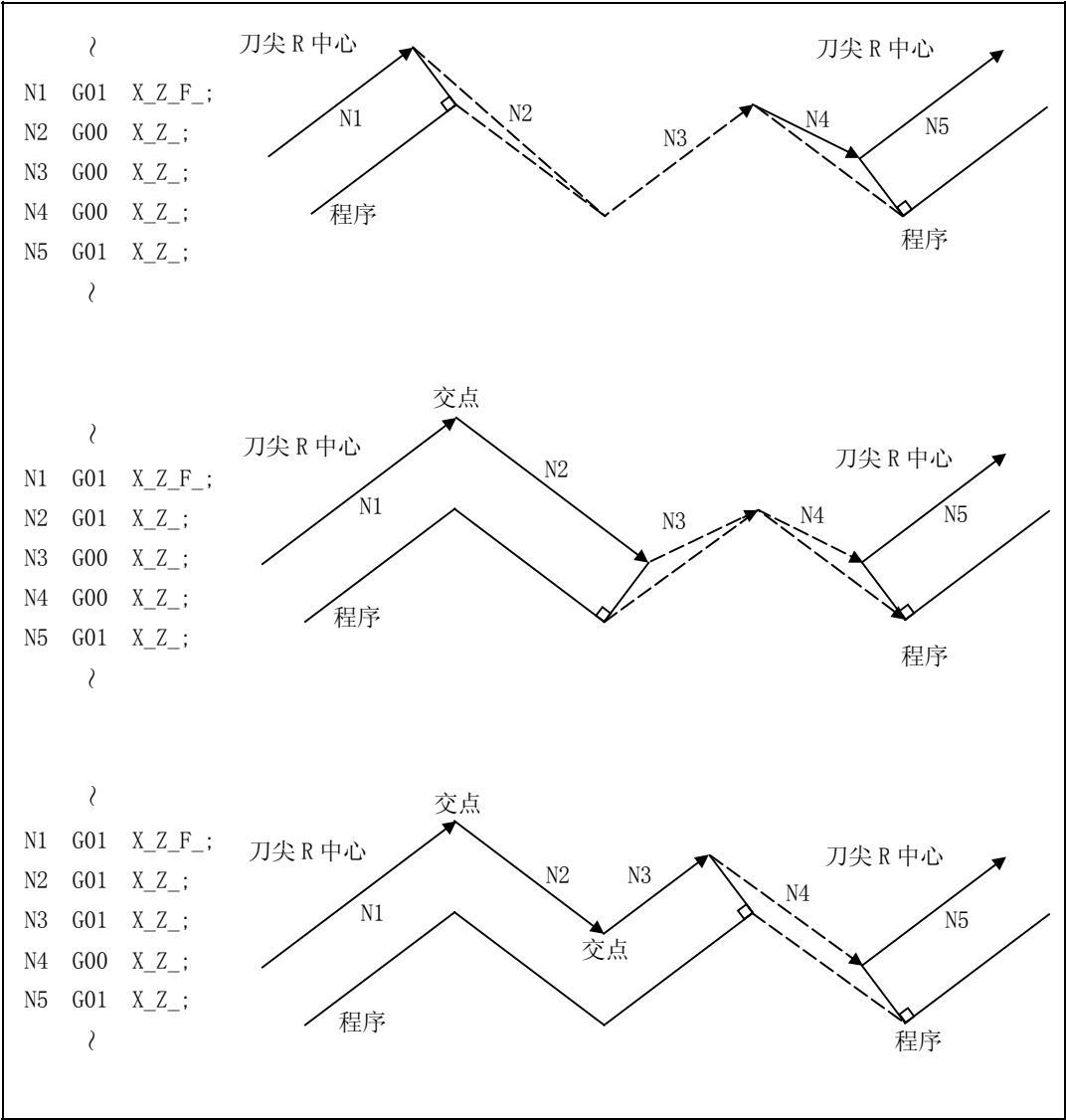
(1) 参考点返回指令



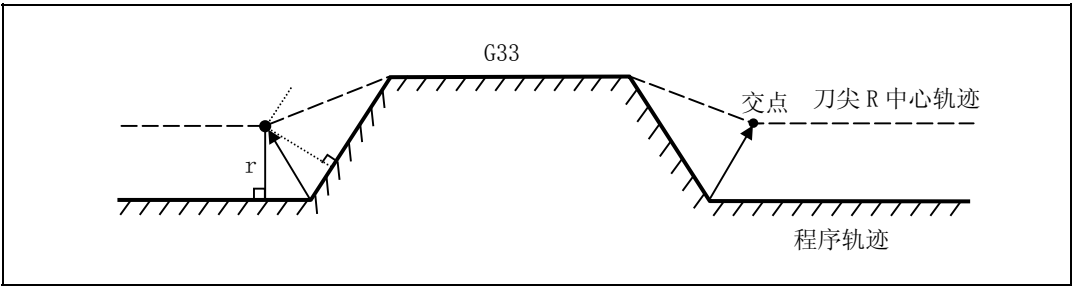
(2) 基本机械坐标系选择 (G53) 中，临时性的清除补偿矢量。

(注 1) 在坐标系设定 (G92) 指令中，补偿矢量不发生变化。

(3) 指令定位 (G00) 指令
G00指令临时取消刀尖R补偿。



(4) G33螺纹切削指令
在G33节中无需进行刀尖R补偿。



(5) 复合型车削用固定循环

如果指令了复合型车削用固定循环 I（G70、G71、G72、G73），则临时取消刀尖R补偿，在取消进行了刀尖R补偿的精整形状的状态下进行切削，切削结束后，则自动返回补偿模式。



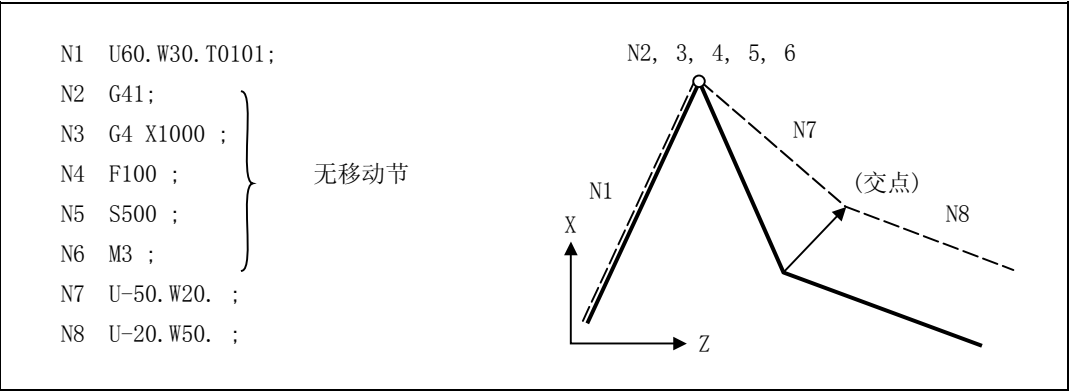
无移动节

如下的节称为无移动的节。

- | | | | |
|---------------------|-----------|---|-----|
| a. M03 ; | M 指令 | } | 无移动 |
| b. S12 ; | S 指令 | | |
| c. T0101 ; | T 指令 | | |
| d. G04X500 ; | 延时 | | |
| e. G10P01R50 ; | 补偿量设定 | | |
| f. G92X600. Z500. ; | 坐标系设定 | | |
| g. Y40. ; | 在补偿平面外的移动 | | |
| h. G00 ; | 仅 G 代码 | | |
| i. U0 ; | 移动量 0 | | |

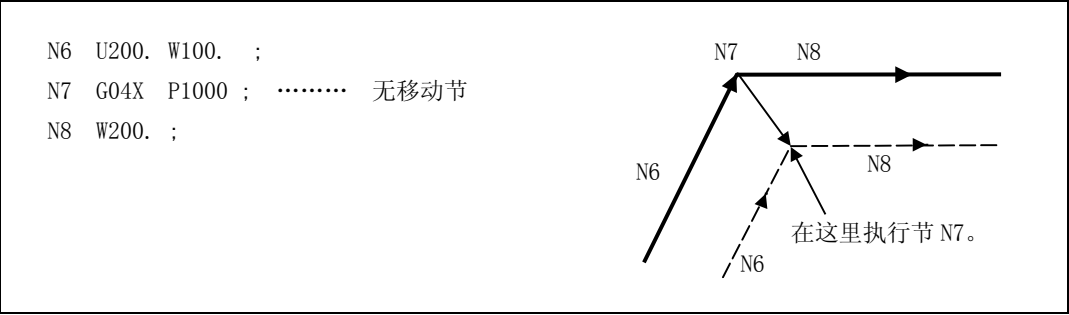
(1) 在补偿开始时进行指令时

当无移动的节连续4节以上时，不创建补偿矢量。

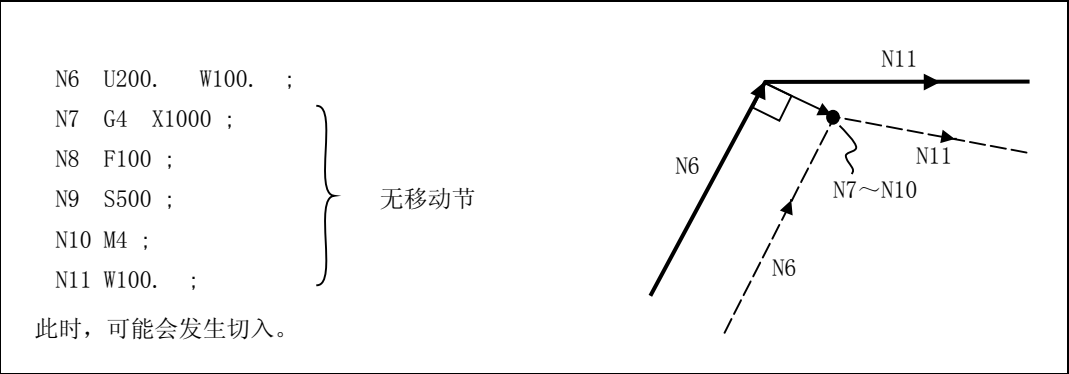


(2) 在补偿模式中进行指令时

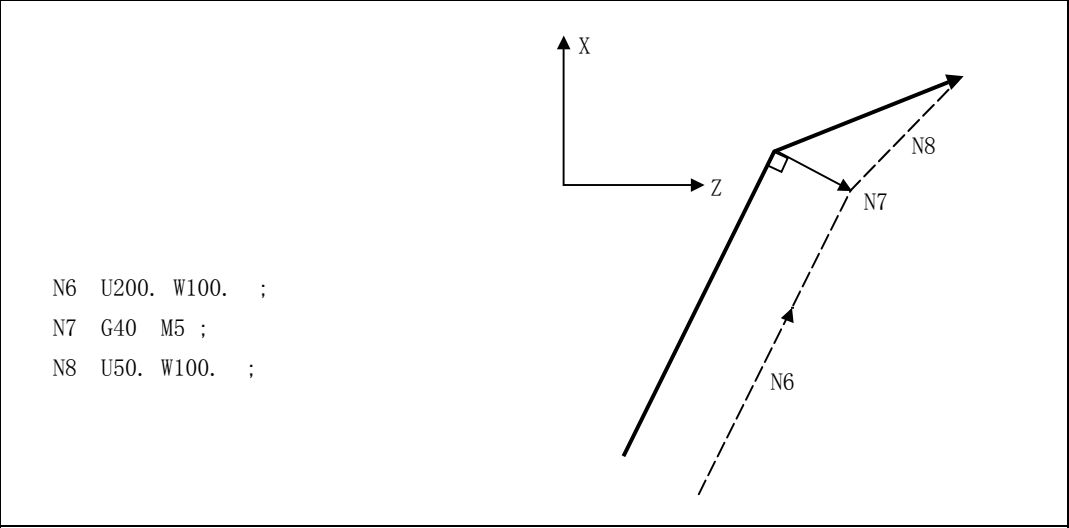
在补偿模式中，如果无移动节没有持续4节以上，则创建通常的交点矢量。



当无移动节持续4节以上时，在上一节的终点创建垂直的补偿矢量。

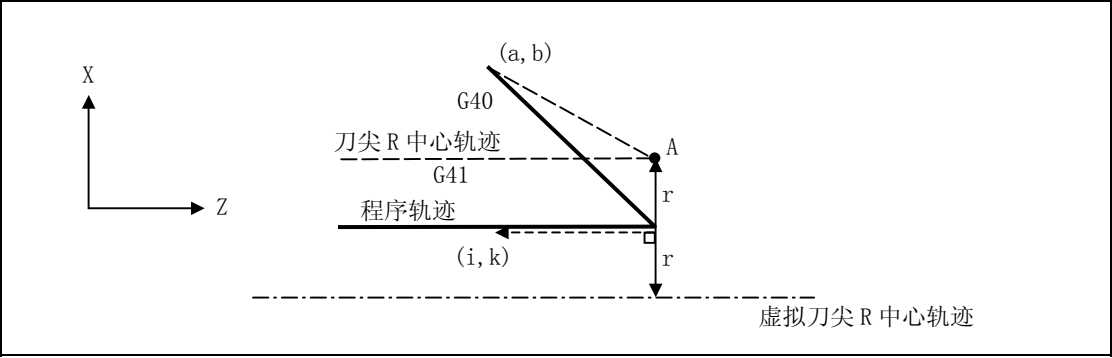


- (3) 在补偿取消的同时进行指令时
当与G40同时指令无移动节时，仅取消补偿矢量。



在 G40 中指令了 I、J、K 时

在G40的上一节创建垂直矢量。



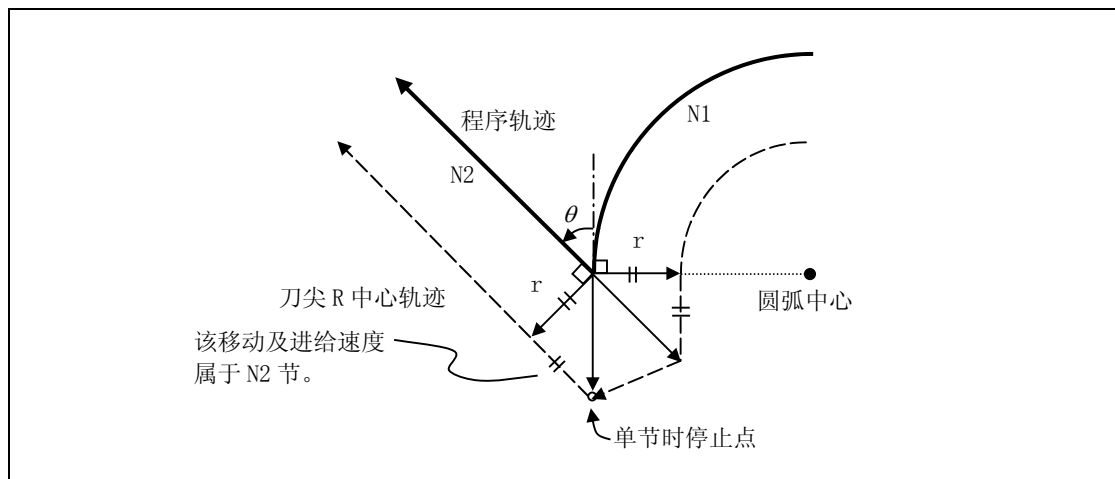


转角的移动

当移动指令节的连接处有多个补偿矢量时，则以直线移动该矢量时间。这一动作称为转角移动。

当矢量不一致时，进行用于旋转转角的移动。

在单节运转中，将上一节+转角移动作为1节加以执行，在下次起动时，将剩余的移动+下一节作为1节加以执行。



12.4.4 G41/G42指令与I、J、K指定



功能及目的

通过在同一节中指令G41/G42与I、J、K，可任意变更补偿方向。



指令格式

```
G18 (ZX 平面) G41/G42 X__ Z__ I__ K__ ;
```

此时，请通过直线指令（G00、G01）设定移动模式。



I、K 型矢量（G18 XZ 平面选择）

对通过本指令创建的新矢量I、K型矢量（G18平面）加以说明。（对于G17平面的IJ、G19平面的JK也请做同样的考虑。）

如下图所示，对于I、K型矢量不进行编程轨迹的交点运算，而是以垂直于I、K所指定的方向，且补偿量较大的矢量作为补偿矢量。不管是在补偿开始时（上一节为G40模式），还是在模式中（上一节为G41/G42模式），均可指令I、K矢量。

（1）补偿开始时指令I、K时

```

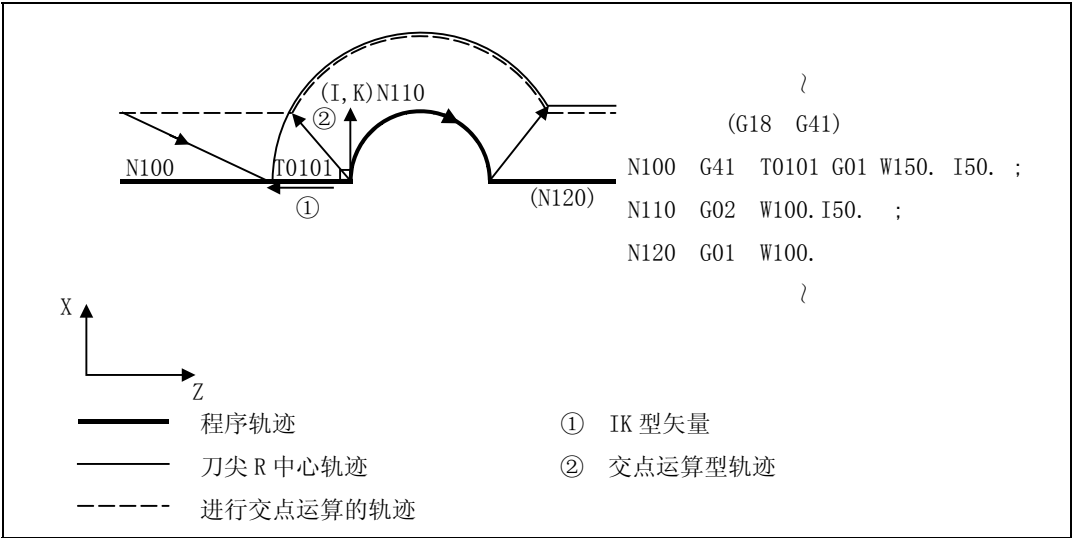
                                (G40)
                                }
N100 G41 U100. W100. K150. T0101 ;
N110 G04 X1000 ;
N120 G01 F1000 ;
N130 S500 ;
N140 M03 ;
N150 Z150. ;
                                }
```

（2）在补偿开始时没有移动指令时

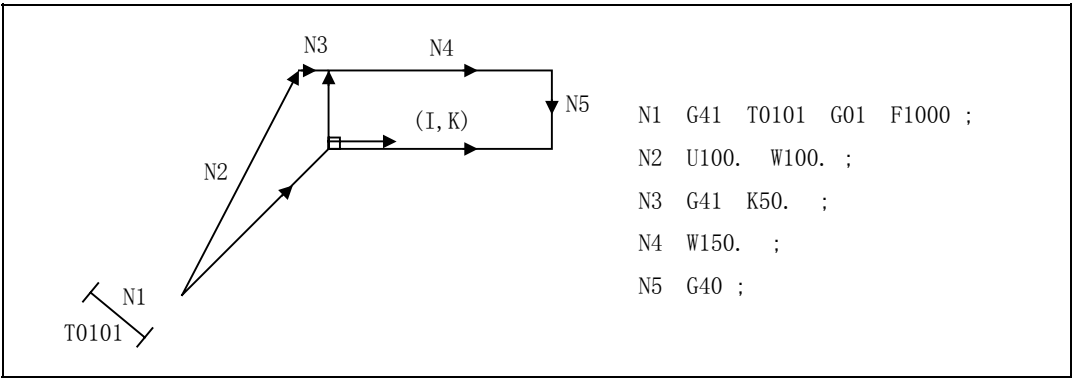
```

                                (G40)
                                }
N1 G41 K150. T0101 ;
N2 U100. W100. ;
N3 W150. ;
                                }
```

(3) 在模式中指令了I、K时（G18平面）



(4) 在无移动节中进行了指令时





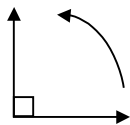
偏移矢量的方向

(1) G41模式时

从Y轴（第3轴）的正方向看I、K所指定方向的顶点，左旋90° 后的方向

(例 1) K100. 时

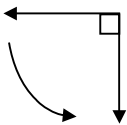
偏移矢量的方向



(0, 100) IK 方向

(例 2) K-100. 时

(0, -100) IK 方向



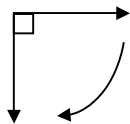
偏移矢量的方向

(2) G42模式时

从Y轴（第3轴）的正方向看I、K所指定方向的顶点，右旋90° 后的方向

(例 1) K100. 时

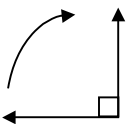
(0, 100) IK 方向



偏移矢量的方向

(例 2) K-100. 时

偏移矢量的方向

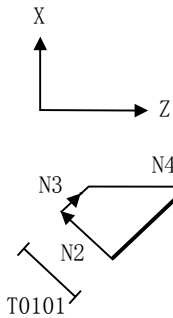


(0, -100) IK 方向



偏移的模式切换

可在中间切换G41/G42的模式。



```
N1 G28 X0Z0 ;
N2 G41 T0101 F1000 ;
N3 G01 U100. W100. ;
N4 G42 W100. I-100. K100.
    T0102 ;
N5 U-100. W100. ;
N6 G40 ;
N7 M02 ;
%
```



偏移矢量的补偿量

补偿量取决于有IK指定的节补偿编号（模态）。

<例 1>

```
(G41 T0101)
      }
N100  G41 W150. K50. ;
N110  U-100. W100. ;
      }
```

矢量A变为注册到N100节T补偿编号模态的1中的补偿量。

<例 2>

```
(G41 T0101)
      }
N200  G41 W150. K50. T0102 ;
N210  U-100. W100. ;
      }
```

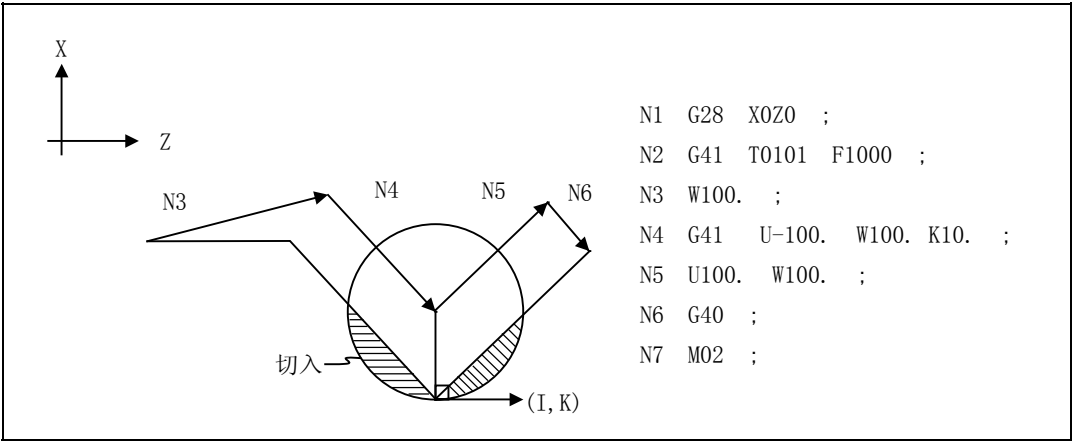
矢量B变为注册到N200节T补偿编号模态的2中的补偿量。



注意事项

(1) 指令I、K型矢量时，请通过直线模式（G0、G1）进行。在补偿开始时，如果是圆弧模式，则发生程序错误（P151）。
偏移模式中，圆弧模式时的IK指定变为圆弧的中心指定。

(2) 指定I、K型矢量时，即使存在干扰，也无法删除矢量（避免干扰）。因此，这种情况下，可能会发生过切。



(3) 通过G41/G42指令的有无与I、K、（J）指令的有无的组合，补偿方法如下表所示。

G41/G42	I、K（J）	补 偿 方 法
无	无	交点运算型矢量
无	有	交点运算型矢量
有	无	交点运算型矢量
有	有	I、K 型矢量 无插入节

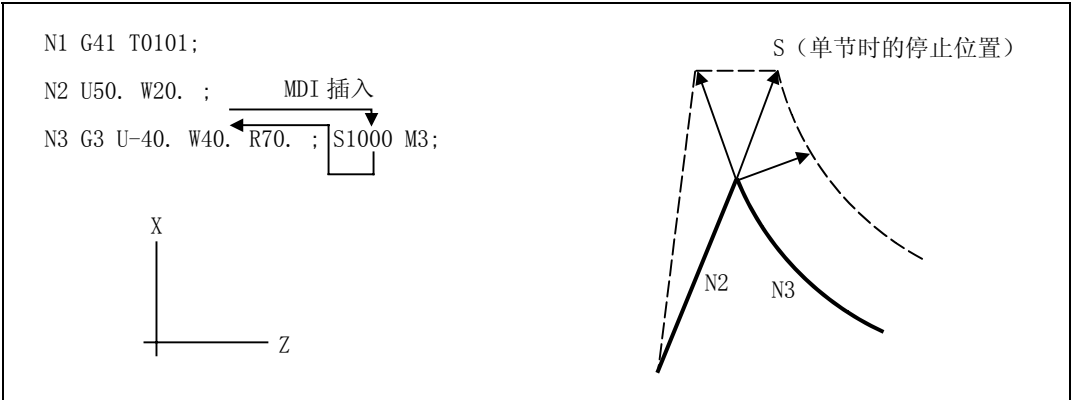
12.4.5 刀尖R补偿中的插入



MDI 插入

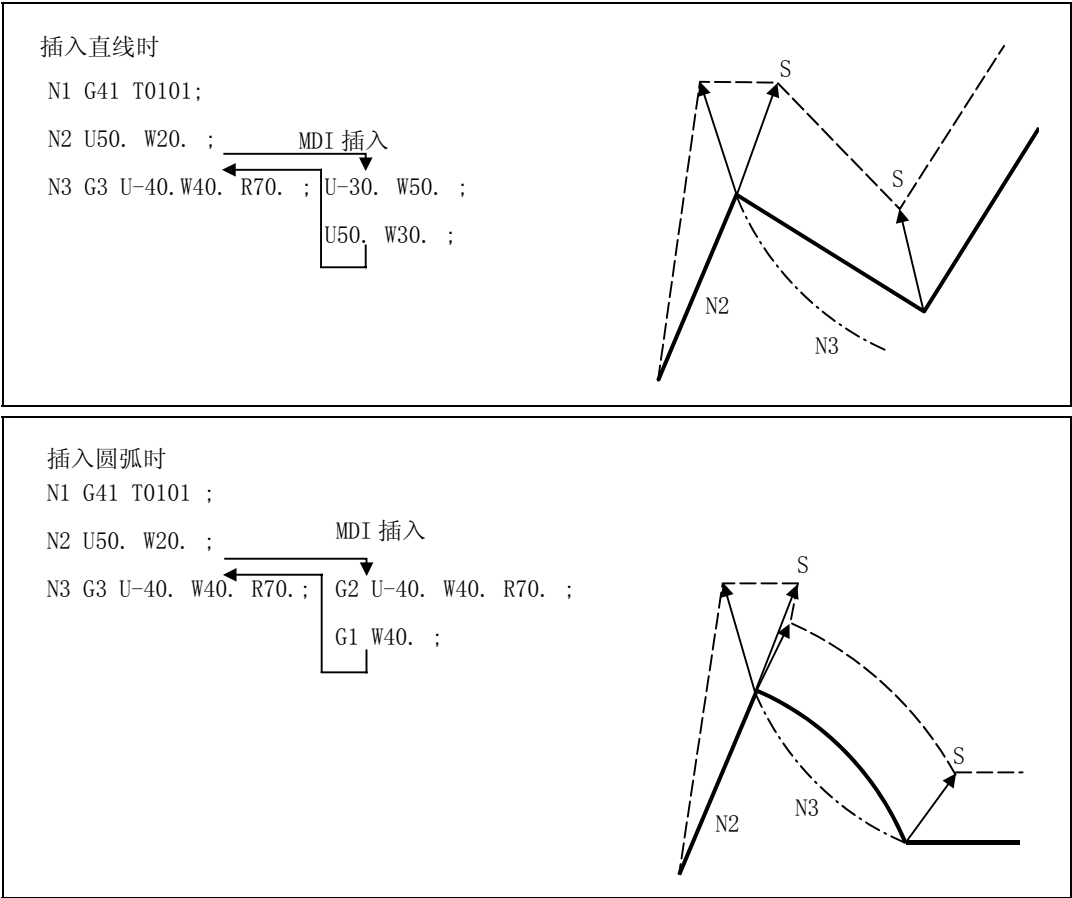
在纸带、内存、MDI运转等自动运转模式中，不管是何种运转模式，刀尖R补偿均有效。
在纸带、内存运转中，节停止之后通过MDI插入，则如下图所示。

(1) 无移动的插入（刀具轨迹不变化）



(2) 有移动的插入

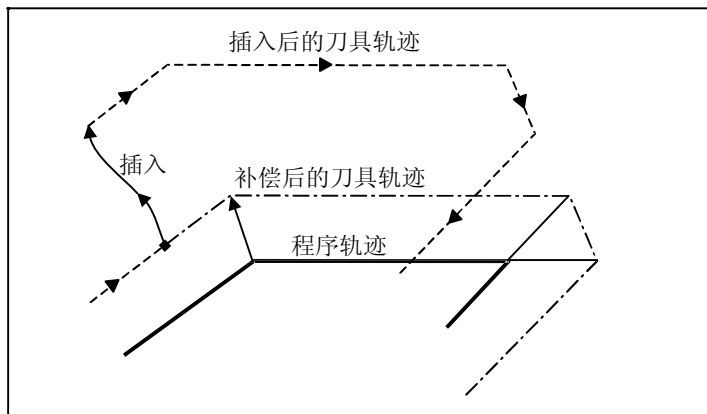
在插入后的移动节中，自动进行补偿矢量的重新运算。



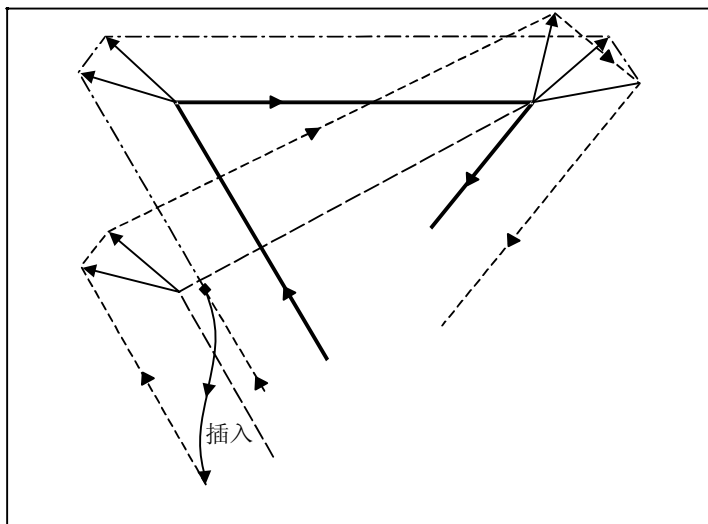


手动插入

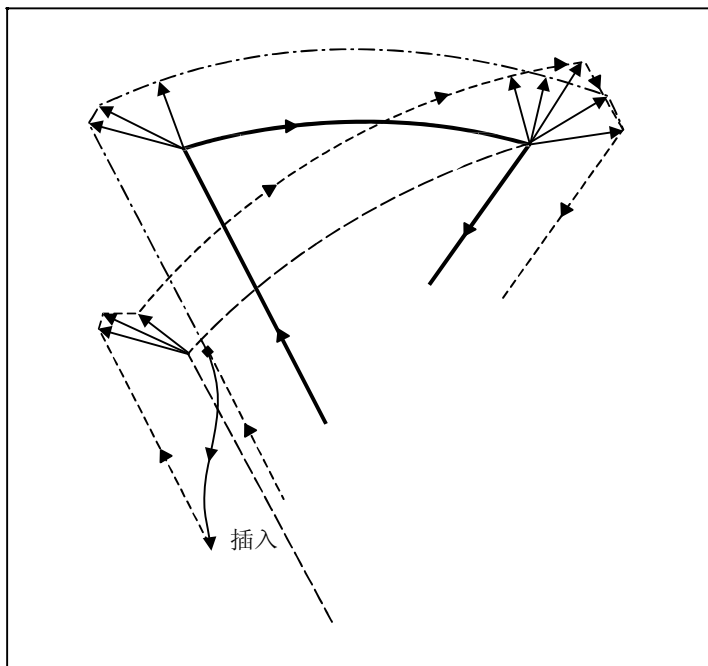
- (1) 手动绝对关闭时的插入 变为
仅偏移插入量后的轨迹。



- (2) 手动绝对打开时的插入 增量值
模式时，与手动绝对关闭 时的
动作相同。



绝对值模式时，如右图所示，在插入节的下一节终点返回本来的轨迹。



12.4.6 刀尖R补偿相关一般注意事项



补偿量

(1) 通常，通过T代码的最后1位或2位指定补偿量的编号，进行补偿量的指定。但是，根据机械规格，可能会有使用前几位进行指定的规格。指定一次T代码之后，在指定其他T代码之前，该T代码一直有效。T代码除用于指定刀尖R补偿的补偿量外，也用于指定刀具长度偏移的偏移量。

(2) 补偿量的变更，通常是在补偿取消模式中，通过选择其他刀具进行，在补偿模式中变更补偿量时，使用该节中指定的补偿量，计算节终点的矢量。



刀尖 R 补偿中的错误

- (1) 在刀尖R补偿中，对下一指令进行编程，则发生错误。
 - G17、G18、G19（当指令了与补偿中的平面不同的平面时……P112）
 - G31（P608）
 - G74, G75, G76（P155）
 - G81～G89（P155）
- (2) 在G46模式时，指定了刀刃点1～8以外的点。（P158）
- (3) 变为开始G46模式刀尖R补偿动作，即使预读5节，当最初的切削指令移动矢量中没有规定补偿方向时，发生错误。（P156）
- (4) 刀尖R补偿动作的开始节及结束节中的圆弧指令为错误。（P151）
- (5) 在G46模式中补偿方向翻转，则发生程序错误。（P157）
也可通过参数，保持同一补偿方向，进行动作。（控制参数#8106 G46 避免翻转错误）
- (6) 执行刀尖R补偿时，在干扰节处理中，如果没有计算出跳跃1节时的交点，则发生程序错误。（P152）
- (7) 在执行刀尖R补偿时，当预读的节中有错误时，发生程序错误。
- (8) 在执行刀尖R补偿中，在没有避免干扰时，如果发生干扰，则导致程序错误。（P153）
- (9) 不具备刀尖R补偿规格，但是指令了刀尖R补偿时，发生程序错误。（P150）

12.4.7 干扰检查



功能及目的

通过通常的2节预读，在刀尖R补偿中进行了补偿的刀刃，可能会切入毛坯中。
这一现象被称为干扰，用于防患干扰于未然的功能，称为干扰检查。
在干扰检查中，具有如下3种功能，可通过参数选择使用其中的哪一个。

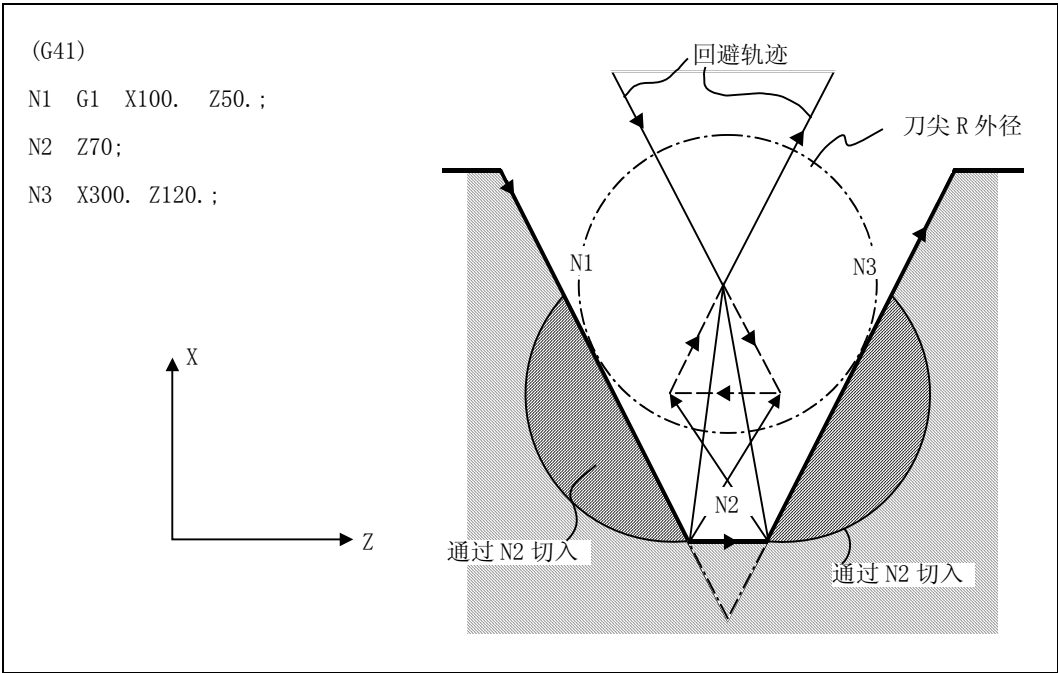
功 能	参 数	动 作
干扰检查报警功能	干扰回避关闭 干涉检查无效关闭	在执行发生切入的节之前，发出程序错误而停止。
干扰检查回避功能	干扰回避打开 干涉检查无效关闭	变更轨迹以避免发生过切。
干扰检查无效功能	干涉检查无效打开	即使发生过切也继续进行切削。 用于微小线段程序。

(注) #8102 干扰回避
 #8103 干扰检查无效



详细说明

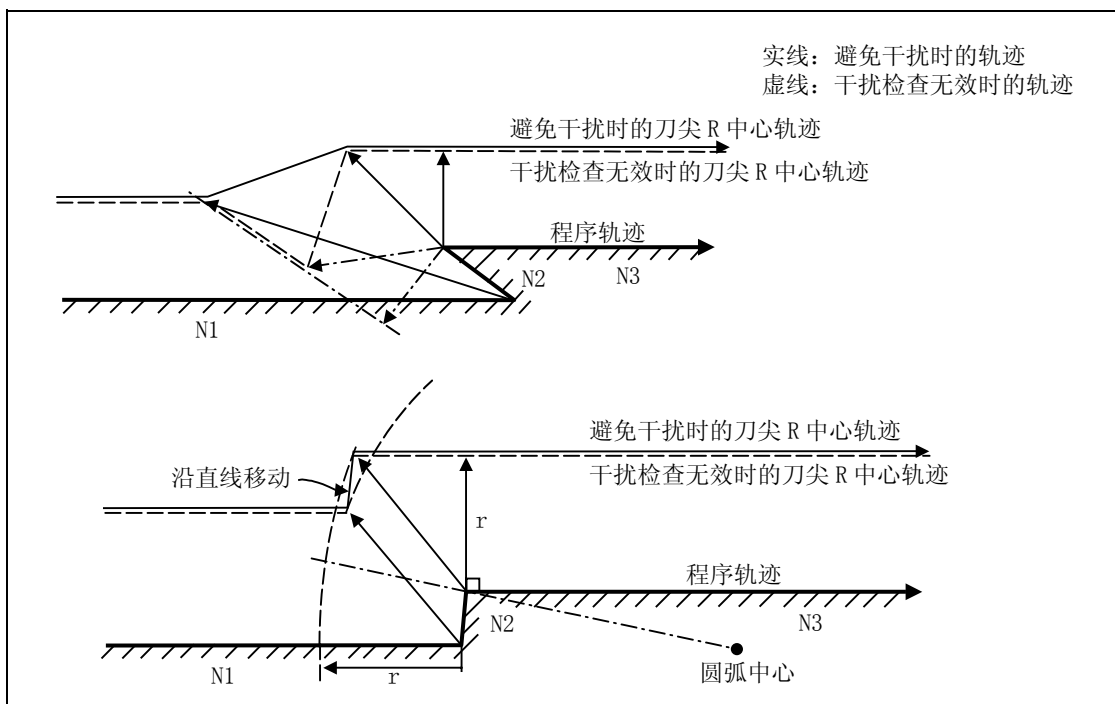
(例)



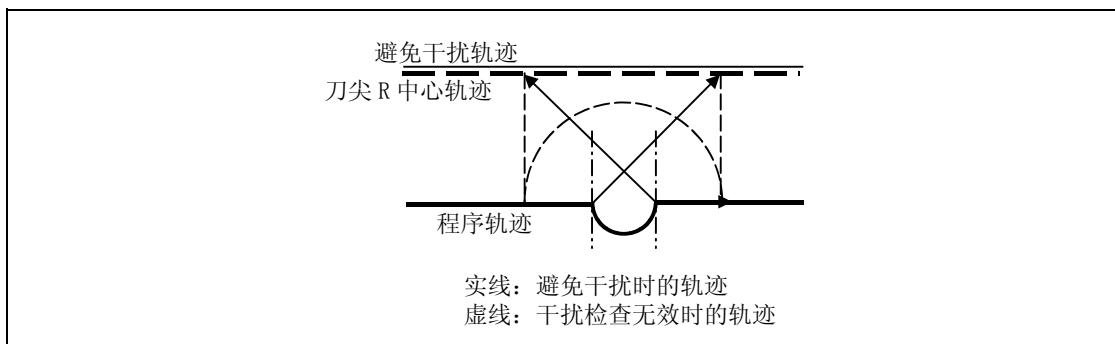
- (1) 报警功能时
 执行N1时发生报警，因此，使用编辑功能，变更为N1 G1 X-100. Z-20 ；等之后，可进行加工。
- (2) 回避功能时
 进行N1与N3的交点运算，创建干扰回避矢量。
- (3) 干扰检查无效功能时
 切入N1与N3的直线的同时，进行通过。



避免干扰时的动作



如下图时，沟槽为切削剩余。





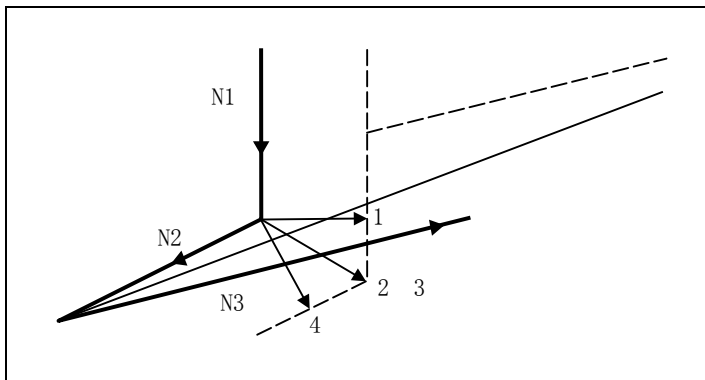
干扰检查报警

在以下条件下，发生干扰检查报警。

(1) 选择干扰检查报警时

(a) 将本节终点的矢量全部删除时

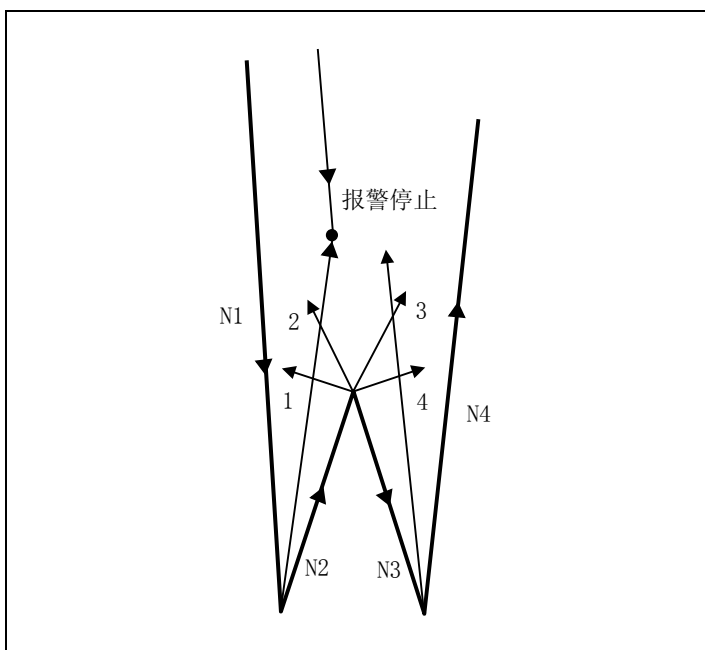
如右图所示，N1 节终点上的矢量 1~4 被全部删除时，在执行 N1 之前，发生程序错误“P153”。



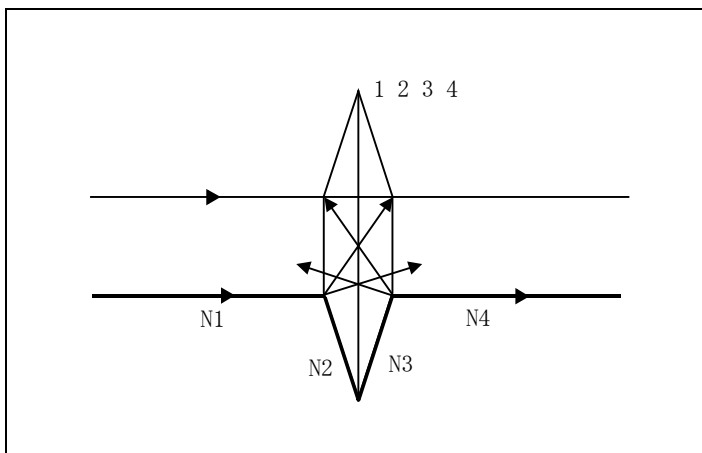
(2) 选择干扰检查回避功能时

(a) 即使本节的终点矢量被全部删除，下一节的终点矢量仍有效时

(i) 右图中，进行 N2 的干扰检查，则 N2 的终点矢量被全部删除，但是 N3 的终点矢量仍视为有效。此时，在 N1 的终点发生程序错误“P153”。

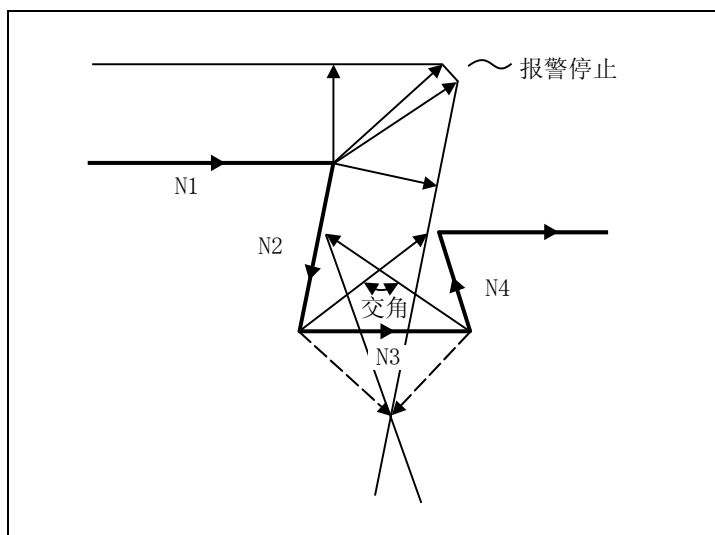
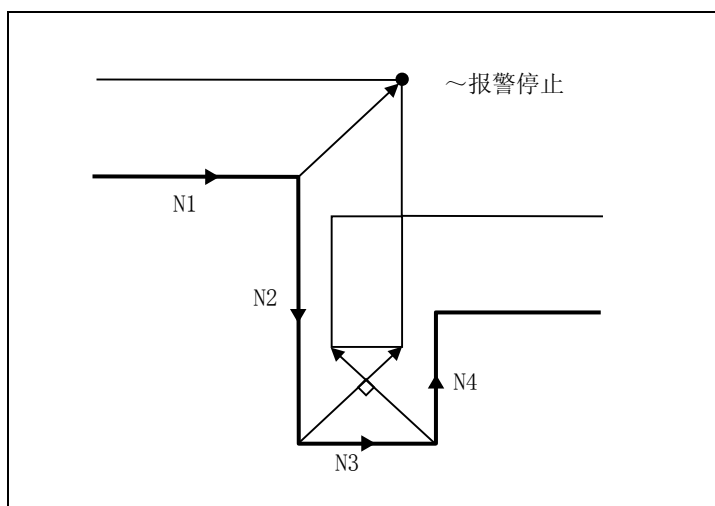


- (ii) 如右图，移动方向在 N2 反转。此时，执行 N1 之后，发生程序错误“P153”。



- (b) 当无法创建回避矢量时

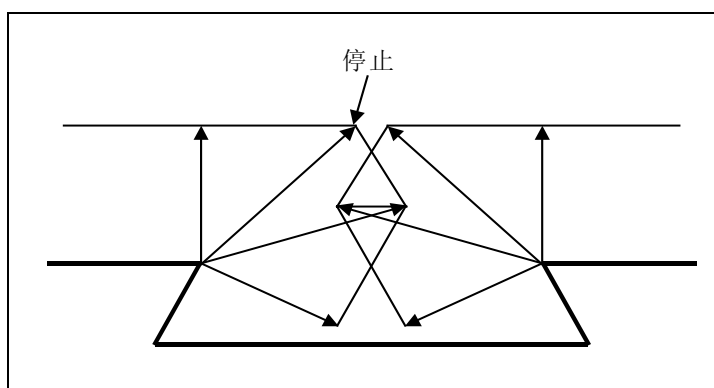
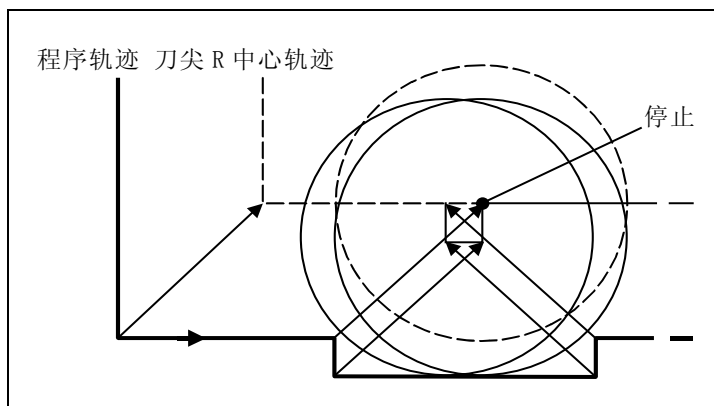
- (i) 如右图所示，即使满足回避矢量的创建条件，也可能会无法创建回避矢量，或是回避矢量在 N3 中发生干扰。所以，当矢量的交角为 90° 以上时，在 N1 的终点发生程序错误“P153”。



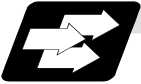
(c) 程序前进方向与补偿后的前进方向相反时

在如下情况下，即使实际上不会发生干扰，也可能被视为发生干扰。

编程加工比刀尖 R 指令小、平行或底部较宽的沟槽时。



12.5 程序补偿输入；G10、G11



功能及目的

可通过G10指令进行刀具偏移量及工件偏移量的设定/变更。当以绝对值（X、Z、R）进行指令时，补偿量为新的量，而当以增量值（U、W、C）进行指令时，以在当前设定的补偿量上增加了指定的补偿量之后的值，作为新的补偿量。



指令格式

(1) 工件偏移输入（L2）

G10 L2 P__ X__ (U__)Z__ (W__) ;	
P__	: 补偿编号
X__	: X 轴补偿量（绝对）
U__	: X 轴补偿量（增量）
Z__	: Z 轴补偿量（绝对）
W__	: Z 轴补偿量（增量）

(2) 刀具长度补偿输入（L10）

G10 L10 P__ X__ (U__)Z__ (W__)R__ (C__)Q__ ;	
P__	: 补偿编号
X__	: X 轴补偿量（绝对）
U__	: X 轴补偿量（增量）
Z__	: Z 轴补偿量（绝对）
W__	: Z 轴补偿量（增量）
R__	: 刀尖 R 补偿量（绝对）
C__	: 刀尖 R 补偿量（增量）
Q__	: 虚拟刀刃点

(3) 刀刃磨损补偿输入（L11）

G10 L11 P__ X__ (U__)Z__ (W__)R__ (C__)Q__ ;	
P__	: 补偿编号
X__	: X 轴补偿量（绝对）
U__	: X 轴补偿量（增量）
Z__	: Z 轴补偿量（绝对）
W__	: Z 轴补偿量（增量）
R__	: 刀尖 R 补偿量（绝对）
C__	: 刀尖 R 补偿量（增量）
Q__	: 虚拟刀刃点

(4) 没有通过刀具长度补偿输入（L10）、刀刃磨损补偿输入（L11）进行的L指令时

刀具长度补偿输入指令	: P = 10000 + 补偿编号
刀刃磨损补偿输入编号	: P = 补偿编号

(5) 补偿输入的取消

G11 ;



详细说明

(1) 补偿编号及虚拟刀刃点的设定范围如下。

地址	地址的含义	设 定 范 围		
		L2	L10	L11
P	补偿编号	0: 外部工件偏移 1: G54 工件偏移 2: G55 工件偏移 3: G56 工件偏移 4: G57 工件偏移 5: G58 工件偏移 6: G59 工件偏移	有 L 指令时 1~最大补偿组数 无 L 指令时 10001~ 10000+最大补偿组数	有 • 没有 L 指令时 均 1~最大补偿组数
Q	虚拟刀刃点		0~9	

(注 1) 在刀具补偿输入 (L10/L11) 中, P (补偿编号) 的最大补偿组数合计最多 80 组。
(由于组数因机种而异, 所以请确认规格。)

(2) 补偿量的设定单位如下。

在进行指令值单位转换之后, 与表中不符合的值变为程序错误 (P35)。
另外, 对于增量值指令, 补偿量设定范围为当前设定值与指令值之和。

(a) E60

输入设定 单位	刀具长度补偿量		磨损补偿量	
	公制体系	英制体系	公制体系	英制体系
#1003=B	±999.999 (mm)	±99.9999 (inch)	±99.999 (mm)	±9.9999 (inch)

(b) E68

输入设定 单位	刀具长度补偿量		磨损补偿量	
	公制体系	英制体系	公制体系	英制体系
#1003=B	±9999.999 (mm)	±999.9999 (inch)	±999.999 (mm)	±99.9999 (inch)
#1003=C	±999.9999 (mm)	±99.99999 (inch)	±99.9999 (mm)	±9.99999 (inch)



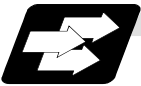
注意事项・限制事项

- (1) 补偿量设定范围的检查
在磨损补偿量的最大值与增量值指令中，每1次的补偿量分别是以磨损补偿输入检查的磨损数据最大值和最大增量值为优先，当将大于该值的值指令为磨损补偿量时，发生程序错误（P35）。
 - (2) G10为非模态，仅指令的节有效。
 - (3) 第3轴也同样可以进行补偿输入，即使是将C轴指令为第3轴时，在L10、L11中，地址C可作为刀尖R的增量值指令使用。
 - (4) 指令了错误的L编号、刀具补偿编号，则分别发生程序错误（P172、P170）。
 - (5) 在工件偏移输入中省略了P指令，则作为当前选中的工件偏移输入使用。
 - (6) 当补偿量超过设定范围时，发生程序错误（P35）。
 - (7) 当在1节内混合输入X、Z与U、W时，当指令了指令X、U或Z、W等同一补偿输入的地址时，后输入的地址生效。
 - (8) 只要指令了哪怕一个G10L（2/10/11） P_ 之后的地址，则被补偿输入。当1个也没有指令时，发生程序错误（P33）。
- （例） G10 L10 P3 ZZ50. ；
- ↓

[刀具长度数据]	
#	Z
3	50.000

被输入。
- (9) 补偿量为小数点有效。
 - (10) 在同一节中指令了G40~G42与G10时，G40~G42被忽略。
 - (11) 请不要在同一节中指令固定循环及子程序调用指令与G10。可能会导致误动作、程序错误。
 - (12) 通过参数“#1100 Tmove0”，在同一节内同时指令了G10与T指令时，在下一节中进行补偿。

12.6 刀具寿命管理 II



功能及目的

刀具寿命管理，是将所使用的刀具分类为若干组，在各组中，对刀具的寿命（使用时间、使用次数）进行管理，当达到寿命时，依次从该刀具所属的组中，选择同种的后备刀具进行使用的，带有后备刀具的刀具寿命管理功能，可以长时间的进行无人化运转。

- (1) 刀具寿命管理刀具根数 最多80根
- (2) 组数 最多80组
- (3) 组编号 1~9999
- (4) 组内刀具根数 最多16根
- (5) 寿命时间 0~999999分钟（约16667小时）
- (6) 寿命次数 0~999999次

在刀具寿命管理数据设定中，包括通过刀具寿命管理画面进行设定和通过NC程序进行管理2种方法。关于通过画面进行设定的方法，请参阅使用说明书。
通过NC程序进行设定时，通过与程序补偿输入相同的方法进行注册。



指令格式

G10 L3 ;	寿命管理用数据注册开始	
P_ L_ N_ ;	组编号、寿命、方式的注册	} 第一组
T_ ;	刀具编号的注册	
T_ ;		
P_ L_ N_ ;	下一组编号、寿命、方式的注册	} 下一组
T_ ;	刀具编号的注册	
T_ ;		
G11 ;	寿命管理用数据注册的结束	
P	: 组编号（1-9999）	
L	: 每一根刀具的寿命（0-999999 分钟 或 0-999999 次）	
N	: 方式选择（0：时间管理、1：次数管理）	
T	: 刀具编号按照此处注册的顺序，进行后备刀具的选择。	
	（刀具编号：1-999999、补偿编号：1-80）Tn 取决于规格。	



注意事项・限制事项

- (1) 利用内存、MDI模式，通过执行上述程序，进行注册。
- (2) 执行上述程序，则以前注册的数据（组编号、刀具编号、寿命）被全部删除。所注册的数据，即使关闭电源也会被保持。
- (3) 通过P指定的组编号不连续亦可，但是请尽可能按照升序排列。通过画面进行监视时，易于查看。另外，无法通过重复组编号进行指令。
- (4) 当省略了寿命数据L_时，该组的寿命数据变为“0”。另外，省略指定方式的N_时，该组的方式取决于基本规格参数“#1106 Tcount”。
- (5) 从G10 L3到G11 之间，无法带有顺序编号进行编程。
- (6) 当使用数据计数器有效信号（Y2CA）为打开时，无法指令G10 L3。
(P177 寿命计数中)



程序例

(1) 格式

：	
T□□□□99 ；	开始使用□□□□组的刀具
：	
：	
T□□□□88 ；	取消□□□□组的刀具偏移
：	（与 T△△00 等价：△△为使用中的刀具编号）
：	
M02(M30) ；	加工程序结束

(2) 具体例

：	
T0199 ；	开始使用 01 组的刀具。
：	
：	
T0188 ；	取消 01 组的刀具偏移。
：	假定使用中的刀具编号为 17，则与 T1700 等价。
：	
T0609 ；	选择刀具编号 06、偏移编号 09。
：	※对于刀具 06，不进行寿命管理。
：	
T0600 ；	取消刀具编号 06 的偏移。
：	
：	
T0299 ；	开始使用 02 组的刀具。
：	
：	
T0199 ；	开始使用 01 组的刀具。
：	如果选中的刀具是使用多个补偿编号的刀具，则第 2 个补偿编号被选中。



刀具选择动作例（在一个刀具上使用了多个补偿编号时）

- (1) 当一个刀具使用了多个补偿编号时，每次执行T□□□□99指令时，选择下一补偿编号。
- (2) 超过注册中的补偿编号个数，进行T□□□□99指令时，继续选择最后的补偿编号。（参阅下述内容）

注册到组 1	程序	刀具选择
T1701	T0199 ;	与 T1701 等价
:		
T1702	T0199 ;	与 T1702 等价
:		
T1703	T0199 ;	与 T1703 等价
:		
T1704	T0199 ;	与 T1704 等价
	:	:
(组 1)	:	: (以下与刀具 17 达到寿命之前相同)

- (3) 通过M02/M30重新启动，或通过外部重新启动输入重新启动后，执行上述程序时，再次从开头的补偿编号开始选择。

12.6.1 刀具寿命的计数方法



功能及目的

刀具寿命的计数方法，包括时间方式与次数方式2种。其中，关于次数方式，可通过参数的设定（#1277 ext13/bit0），将计数方法与时机切换为类型2。

计数出的结果，如果使用数据与寿命数据相等，或是超过寿命数据，则在下一次向该组发出选择指令时（T□□□□99），选择组内的后备刀具，对新选择的刀具进行计数。

当组内的刀具全部达到寿命，或后备刀具无法选择时，直接进行计数。

（1）采用时间方式时的时间计数

切削模式（G01、G02、G03、G31、G33等）中，以100ms为单位，对刀具的使用时间进行计数。

另外，在延时、机械锁定、辅助功能锁定、空运转状态下，不进行计数。是否在单节时进行计数，则通过参数设定进行切换。

- （注意）
- 寿命的最大值为 999999 分钟。
 - 刀具寿命管理画面中所显示的数据是以分为单位。

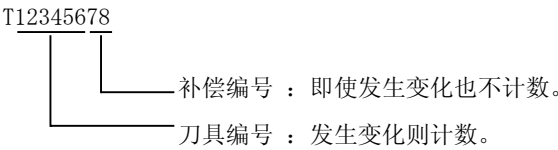
(2) 当采用次数方式时的次数计数

(a) 类型1 (#1277 ext13/bit0: 0)

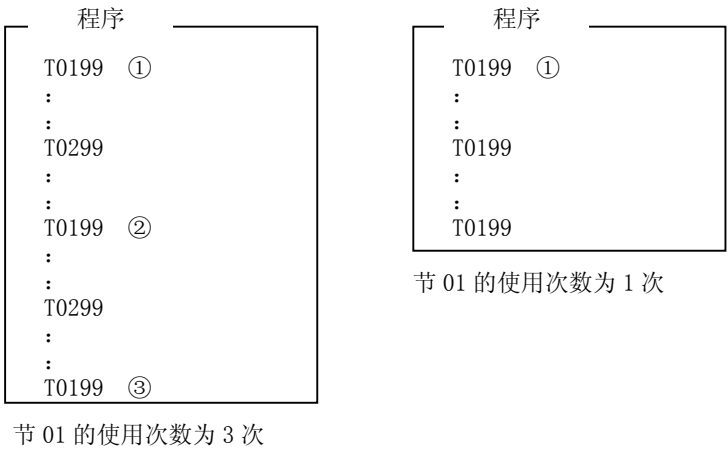
通过执行刀具选择指令(T□□□□99)，所使用的刀具编号发生变化，且在切削模式时（机床锁定、辅助功能锁定、空运转状态时除外）进行计数。
在编号变化后，没有进入过切削模式时，不进行计数。
是否在单节时进行计数，则通过参数设定进行切换。
(注意) • 寿命的最大值为999999次。

• 当仅使用中的刀具的补偿编号发生变化时，不进行计数。

例：使用中的刀具的T代码为 T12345678 时



《动作例》



★ 使用次数为执行一次程序时的统计数据。重新启动后，再次执行程序，则计数增加。

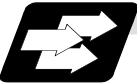
(b) 类型2 (#1277 ext13/bit0: 1)

- (1) 加工程序开始后，重新启动之前，切削中所使用的组只累加“1”。在重新启动时进行累计。
- (ii) 当发出重新计数M的指令时，之前所使用过的组计数器增加“1”。

- (注1) 机床锁定、辅助功能锁定、空运转状态下，不进行计数。
- (注2) 单节运转时，通过参数选择是否进行计数。
- (注3) 寿命的最大值为999999次。

13. 程序辅助功能

13.1 车削用固定循环



功能及目的

本功能是在车削加工中，当进行表面切削等时，可使用1个节指令通常需要多个节进行指令的形状，以有效简化加工程序的功能。车削用固定循环中，包括以下的种类。

G 代码	功 能
G77	横向车削循环
G78	螺纹切削循环
G79	端面车削循环



指令格式

G77 X/U_ Z/W_ R_ F_ ；

(G78、G79也相同。)



详细说明

- (1) 由于固定循环的指令是模态G代码，所以在发出同一模态组的指令或取消指令前，一直有效。
取消指令为以下G代码。
- G00, G01, G02, G03
G09,
G10, G11
G27, G28, G29, G30
G30,
G33, G34
G37,
G92,
G52, G53
G65,
- (2) 固定循环的调用，称为移动指令节。
调用移动指令节是指在固定循环模式中，仅当有轴移动指令时，调用固定循环用子程序，执行到固定循环被取消为止。
- (3) 在执行车削用固定循环（G77～G79）时，虽然可以进行手动插入，但是在插入结束后，请务必返回到进行手动插入的位置，然后重新起动车削用固定循环。
如果不返回就重新启动，则之后的动作会偏移手动插入量进行操作。

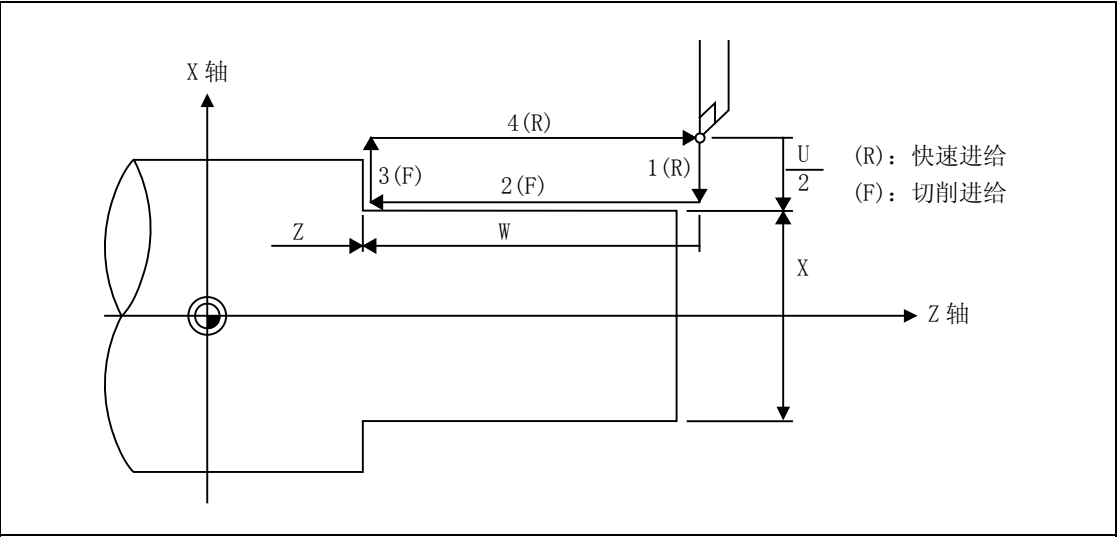
13.1.1 横向切削循环；G77



直线切削

通过以下指令，可进行直型工件横向方向上的连续切削。

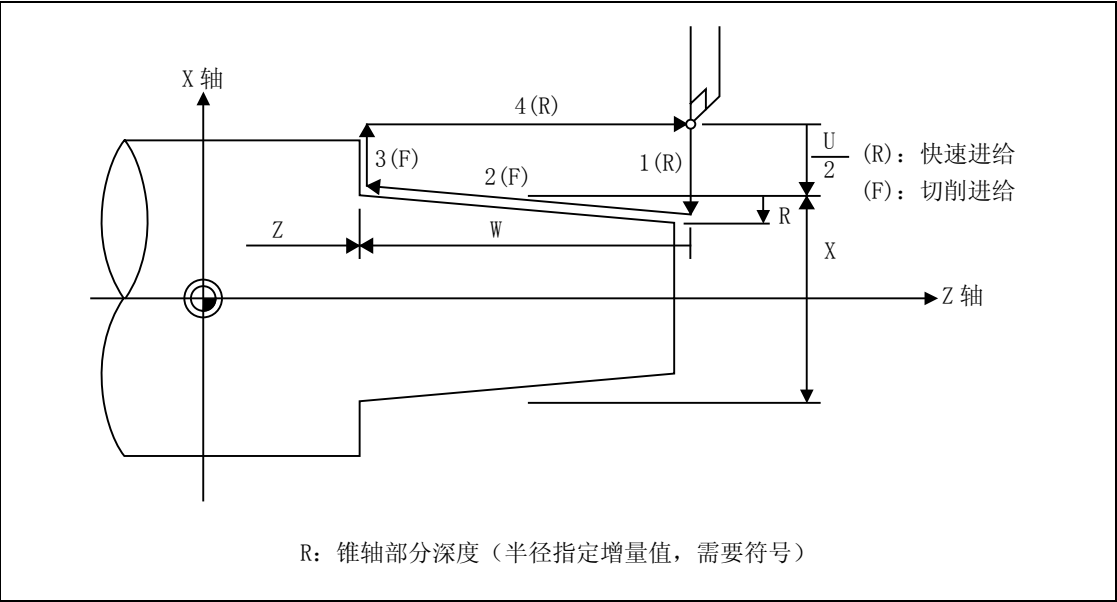
G77 X/U_ Z/W_ F_ ；



锥轴切削

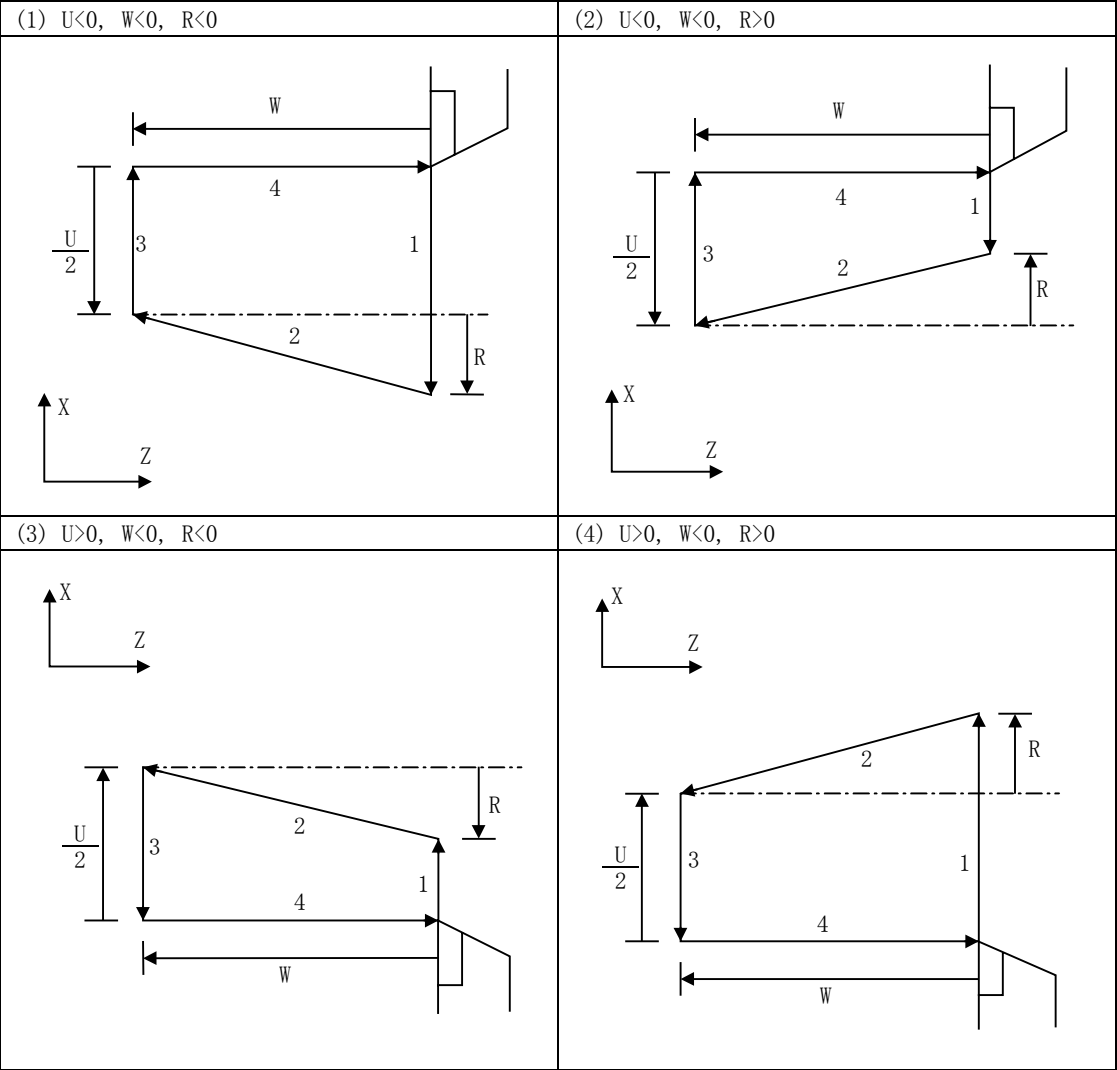
通过以下指令，可进行锥轴型工件横向方向上的连续切削。

G77 X/U_ Z/W_ R_ F_ ；



单节运转中，在1、2、3、4的各动作终点停止。

根据U、W及R的符号，变为如下的形状。



对于（2）、（3），如果不满足如下条件，则发生程序错误（P191）。

$$|U/2| \geq |r|$$

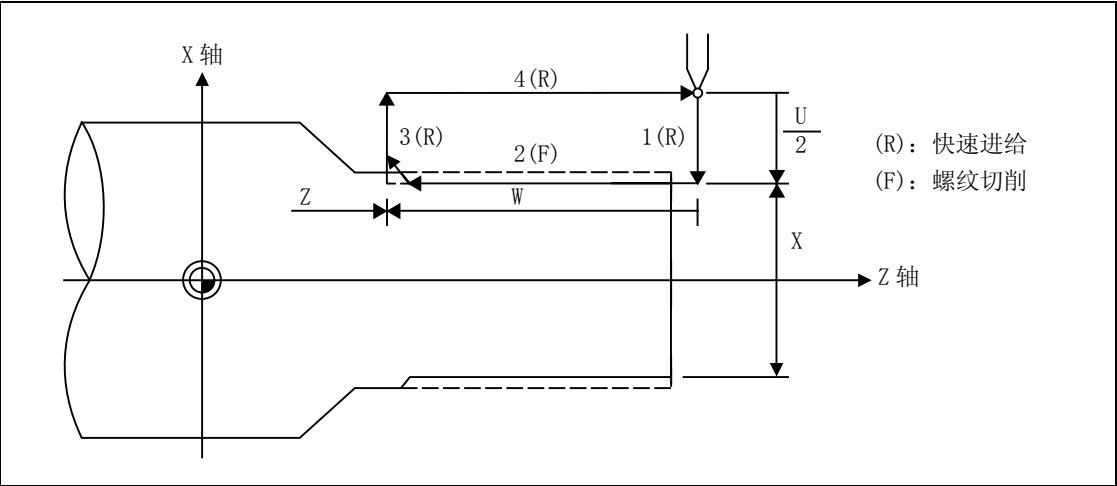
13.1.2 螺纹切削循环；G78



直线切削

通过如下的指令，可进行直型工件的螺纹切削。

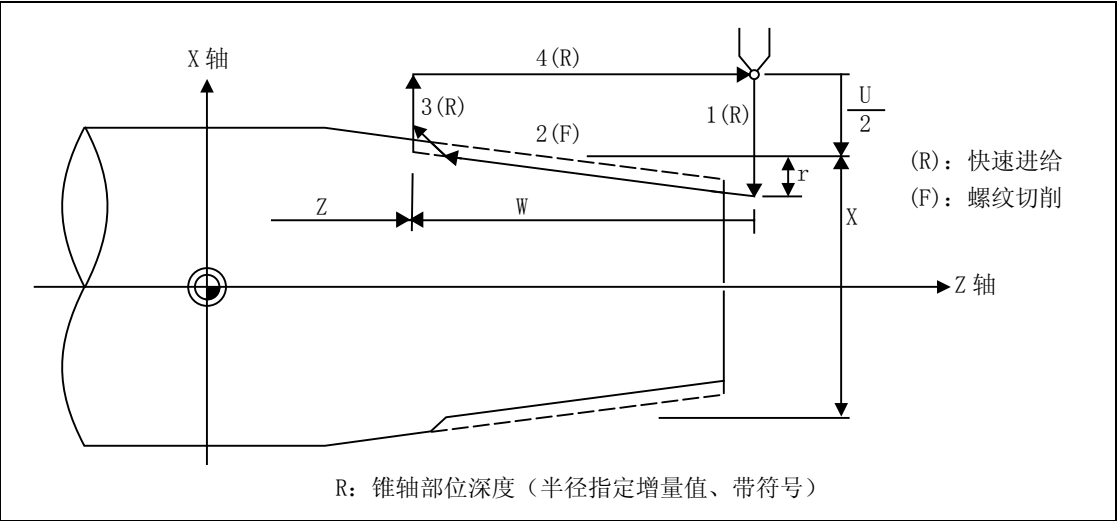
G78 X/U_ Z/W_ F/E_ Q_ ；



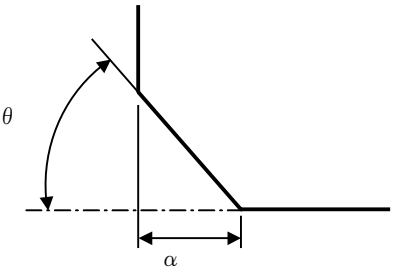
锥轴切削

通过如下的指令，可进行锥轴的螺纹切削。

G78 X/U_ Z/W_ R_ F/E_ Q_ ；



倒角

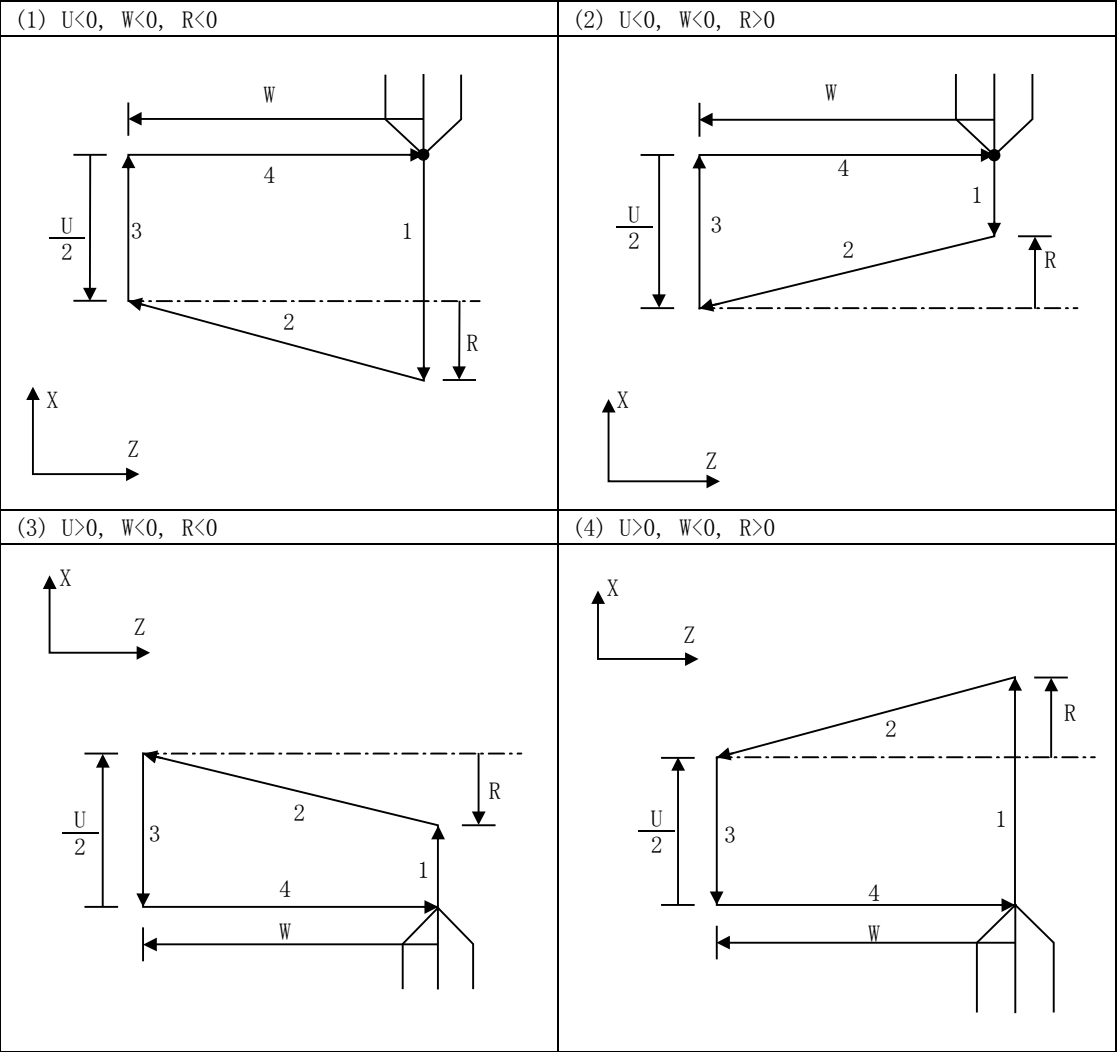
	α : 螺纹的切除量 如果以螺纹导程为 L, 则可在 0~12.7L 的范围内, 以每 0.1L 为一档设定参数。 θ : 螺纹的切除角度 可在 0~89° 的范围内, 以每 1° 为一档, 进行设定。
---	---

单节运转中, 在1、3、4节的终点停止。

在螺纹切削循环中回馈等待时, 从非螺纹切削模式及开始执行螺纹切削指令, 到轴开始移动之间, 进入自动运转停止状态, 螺纹切削时, 在螺纹切削的下一个移动完成（完成3）位置停止。

另外, 在螺纹切削中, 空运转的有效/无效不会发生变化。

根据U、W及R的符号，变为如下的形状。



对于（2）、（3），如果不满足如下条件，则发生程序错误（P191）。

$$|U/2| \geq |r|$$

- （注1）螺纹切削开始移位角度并非模态。当G78中没有Q指令时，看作为“Q0”。
- （注2）当在G78的Q中指令了超过360.000的值时，作为“Q360.000”加以使用。
- （注3）G78在1个循环中进行1条切削。当进行2条切削时，请变更Q的值，执行同一指令。

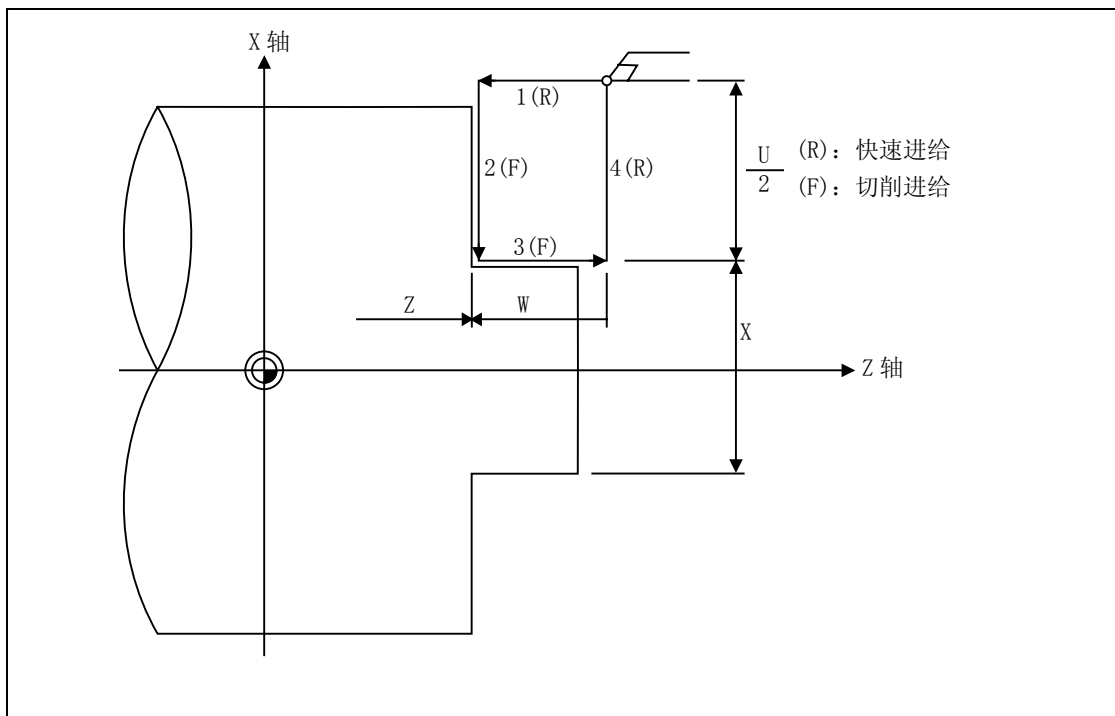
13.1.3 端面切削循环；G79



直线切削

通过以下指令，可进行直型工件端面方向上的连续切削。

G79 X/U_ Z/W_ F_ ；

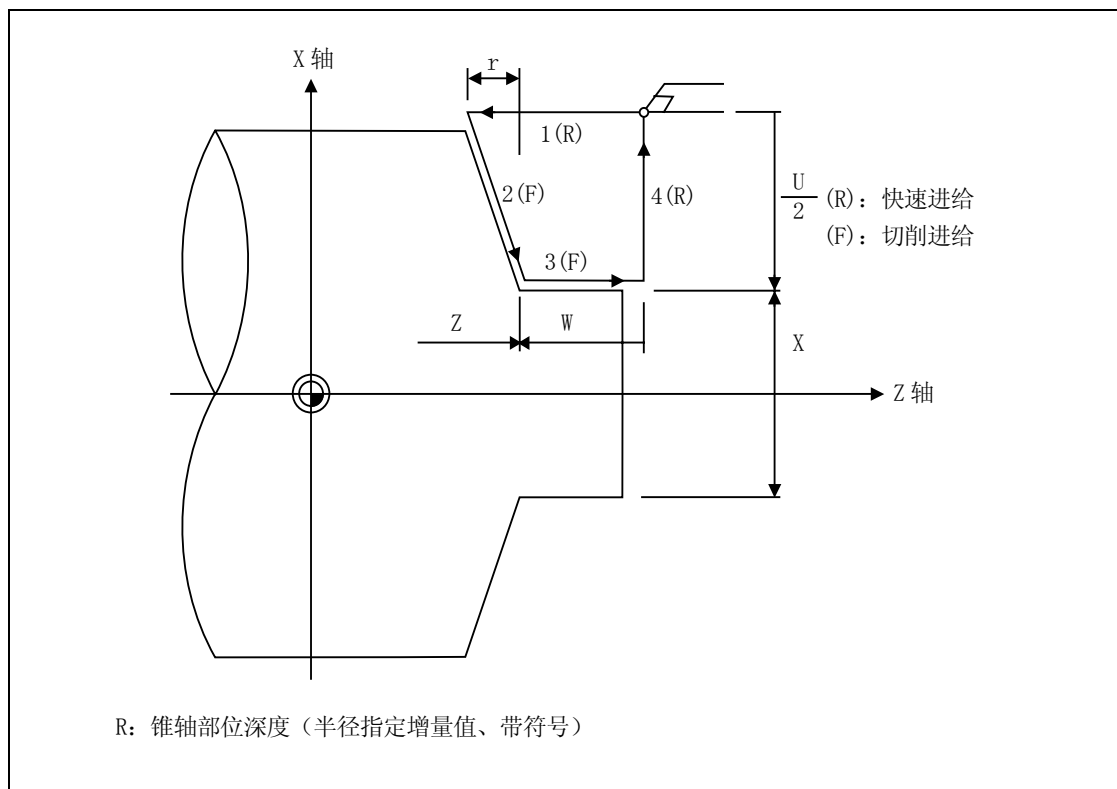




锥轴切削

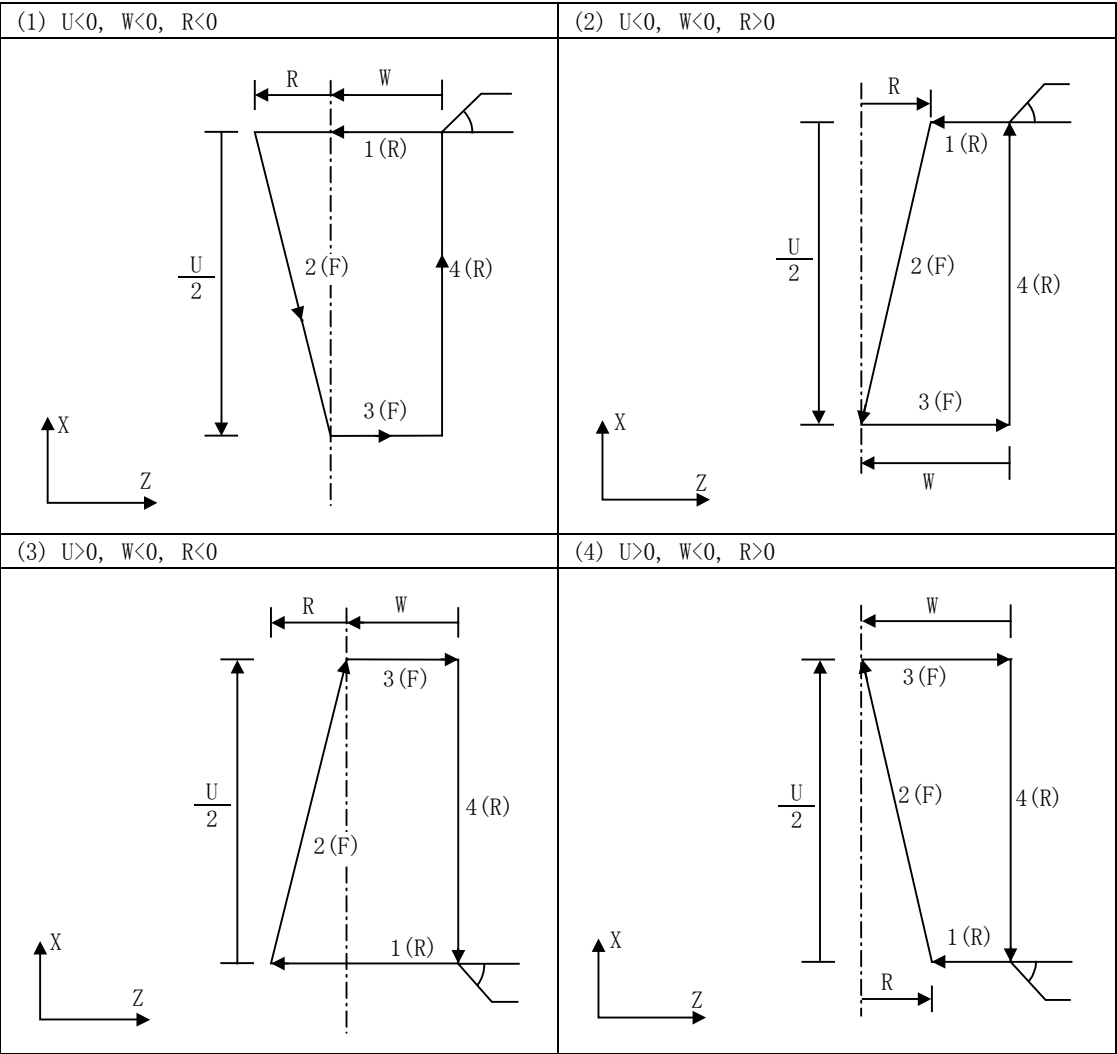
通过以下指令，可进行锥轴工件端面方向上的连续切削。

G79 X/U_ Z/W_ R_ F_ ;



单节运转中，在1、2、3、4的各动作终点停止。

根据U、W及R的符号，变为如下的形状。



对于（2）、（3），如果不满足如下条件，则发生程序错误（P191）。

$$|w| \geq |r|$$

13. 程序援助功能

13.2 车削用固定循环（MELDAS 特殊格式）

13.2 车削用固定循环（MELDAS特殊格式）

[E68]



功能及目的

本功能是在车削加工中，当进行粗切削等时，可使用1个节指令通常需要多个节进行指令的形状，以有效简化加工程序的功能。
MELDAS特殊格式时（#1265 ext01/bit1 打开），部分地址与通常的格式不同。在本节中，对与通常格式不同的部分加以说明。
车削用固定循环的详情，请参阅“13.1 车削用固定循环”。



指令格式

（G78 也相同。）

G79 X(U)_ Z(W)_ K_ F_ ;



详细说明

（1）MELDAS特殊格式与通常格式的比较
MELDAS特殊格式与通常格式有部分地址不同。

功能	MELDAS 特殊格式	通常格式	与通常格式的差异
横向车削循环	G77 X Z I F; 或 G77 U W I F;	G77 X Z R F; 或 G77 U W R F;	锥轴部位深度 通过 R→I 指定
螺纹切削循环	G78 X Z I F; 或 G78 U W I F;	G78 X Z R F; 或 G78 U W R F;	锥轴部位深度 通过 R→I 指定
端面车削循环	G79 X Z K F; 或 G79 U W K F;	G79 X Z R F; 或 G79 U W R F;	锥轴部位深度 通过 R→K 指定

13.3 复合型车削用固定循环



功能及目的

本功能可通过在1节中指令程序，执行事先准备好的固定循环。
固定循环包括如下种类。

G 代码	功能	
G70	精整循环	复合型车削用固定循环 I
G71	横向粗切削循环（精整切削形状切削）	
G72	端面粗切削循环（精整切削形状切削）	
G73	形材粗切削循环	
G74	端面平削循环	复合型车削用固定循环 II
G75	长度平削循环	
G76	复合型螺纹切削循环	

上述功能中，如果没有通过复合型车削用固定循环 I（G70～G73）将精整切削形状程序注册到内存中，则无法使用。
在G71、G72中包括进行袋形加工的类型（复合型车削用固定循环 类型 II）。这是E68专用规格。



指令格式

G70	A__ P__ Q__ ;
G71	U__ R__ ; G71 A__ P__ Q__ U__ W__ F__ S__ T__ ;
G71	W__ R__ ; G72 A__ P__ Q__ U__ W__ F__ S__ T__ ;
G73	U__ W__ R__ ; G73 A__ P__ Q__ U__ W__ F__ S__ T__ ;
G74	R__ ; G74 X__(U)__ Z(W)__ P__ Q__ R__ F__ ; (G75 也相同。)
G76	P__ R__ ; G76 X__(U)__ Z(W)__ P__ Q__ R__ F__ ;



详细说明

- (1) 复合型车削用固定循环 I 的A、P、Q指令如下。
- (a) 没有A的指令时，调用当前执行中的程序内的P、Q。
有A指令，没有P指令时，将通过A指令指定的程序的开头节看作为P指令。
 - (b) 没有Q指令时，继续执行到找到M99指令为止。
既没有Q指令也没有M99指令时，继续执行到精整切削形状程序的最终节为止。

13.3 复合型车削用固定循环



调用精整切削形状程序, 自动计算途中轨迹的同时, 在横向方向上进行粗切削。



G71 Ud Re Hh ; ← (使用通过参数设定的值时可省略) G71 Aa Pp Qq Uu Ww Ff Ss Tt ;	
<H0: 用于无袋形的加工形状>	<H1: 主要用于有袋形的加工形状>
Ud : 切入量 (模态) 可逆参数 (“G71 切入”) 单位 μm / 1/10000inch 半径值指令 Re : 退刀量 (模态) 可逆参数 (“G71 退刀”) 单位 μm / 1/10000inch 半径值指令 Hh : 袋形加工 (模态) 可逆参数 (“G71 袋形加工”) 0: 请只限于选择无凹陷部分 (袋形) 的加工形状。 ※结束部分, 是每 1 循环向 45° 方向退刀, 最后描绘加工形状。 1: 当加工形状中有/无凹陷部分 (袋形) 时均可选择。 ※结束部分, 是每 1 循环描绘加工形状的方式。 参数 退刀方式 [控制参数] “#8110 G71/G72 袋形加工” 0: 不进行袋形加工 1: 进行袋形加工 通过参数 “#8110 G71/G72 袋形加工” 或 Hh 指定, 进行袋形状加工的选择, 当以下参数打开时, 根据精整切削形状开始节的轴数, 自动决定袋形加工。 <ext08/bit0=0> 参数 “#8110 G71/G72 袋形加工” 或 Hh 指定。 <ext08/bit0=1> 精整切削形状开始节的轴指令为 X、Z 两轴时 : 袋形加工 (与 H1 等价) 精整切削形状开始节的轴指令为 X 或 Z 中的 1 轴时 : 与 H0 等价 (注) ext08/bit0=1 时, 结束时的退刀方向为 X 轴方向。 (请参阅 “结束方式・退刀量” 项。)	

Aa	: 精整切削形状程序编号（如省略则为执行中的程序） 省略 A 指令时，变为执行中的程序 P、Q 指令。 另外，省略 A 时的程序执行顺序，在执行完 G71 之后，变为 Qq（精整切削形状结束顺序编号）的下一程序。
Pp	: 加工形状开始顺序编号（如省略则为程序开头）
Qq	: 加工形状结束顺序编号（如省略则到程序结束为止） 但是，即使有 Q 指令，当 M99 在前时，则执行到 M99 为止。
Uu	: X 轴方向上加工量（如省略，则视为 X 轴方向加工量=0） 单位 μ 取决于 $m / 1/10000inch$ 直径/半径值指令（#1019 dia）。
Ww	: Z 轴方向上加工量（如省略，则视为 Z 轴方向加工量=0） 单位 $\mu m / 1/10000inch$ 半径值指令
Ff	: 切削速度（如省略，则为 G71 之前的切削速度（模态））
Ss, Tt	: 主轴指令、刀具指令



切削形状

(1) 精整切削形状开始节与精整切削形状结束节

G71 Aa Pp Qq

使用上述语句指定精整切削形状开始节与精整切削形状结束节。

精整切削形状节数，包括因转角倒角、转角R指令、刀尖R指令而插入的节在内，最多50节。当超过节数时，发生程序错误（P202）。

<袋形加工关闭时>	<袋形加工打开时>
<当在执行中程序以外的加工程序中创建精整切削形状程序时> G71 Aa Pp Qq a: 精整切削形状程序编号 p: 精整切削形状开始顺序编号 (如省略则为从精整切削形状程序开头开始) q: 精整切削形状结束顺序编号 (如省略, 则为到精整切削形状程序的最后或 M99 为止)	<当在执行中程序中创建精整切削形状程序时> G71 Pp Qq p: 精整切削形状开始顺序编号 q: 精整切削形状结束顺序编号
执行中的节 G0 X80.0 Z75.0 T0101; ← (a) G71 U10. R3.; ← (b) G71 <u>A100 P10 Q20</u> U3. W1.5 F500 S1500; ← (c) G70 A100 P10 Q20; ← (d) : 形状程序 <u>O100</u> G28 XZ; <u>N10</u> G0 X15.0 Z65.0; ———— G1 Z55. F450; G1 X30.0; G3 X40.0 Z50.0 R5.0; : G1 Z42.0; G1 X50.0 Z35.0; <u>N20</u> G1 X60.0; ———— N30 G0 X13.0 Z68.0; : 程序执行顺序 (a) (b) (c) (d) 精整形状程序	执行中的程序 G0 X80.0 Z75.0 T0101; ← (a) G71 U10. R3.; ← (b) G71 <u>P10 Q20</u> U3. W1.5 F500 S1500; ← (c) <u>N10</u> G0 X15.0 Z65.0; ———— G1 Z55. F450; G1 X30.0; G3 X40.0 Z50.0 R5.0; G1 Z42.0; G1 X50.0; G1 X55.0 Z35.0; <u>N20</u> G1 X60.0; ———— G70 A100 P10 Q20; ← (d) 程序执行顺序 (a) (b) (c) (d) 精整形状程序 ※G70 指令: 精整循环 按照精整程序进行精整切削, 最后返回 G70 指令位置。

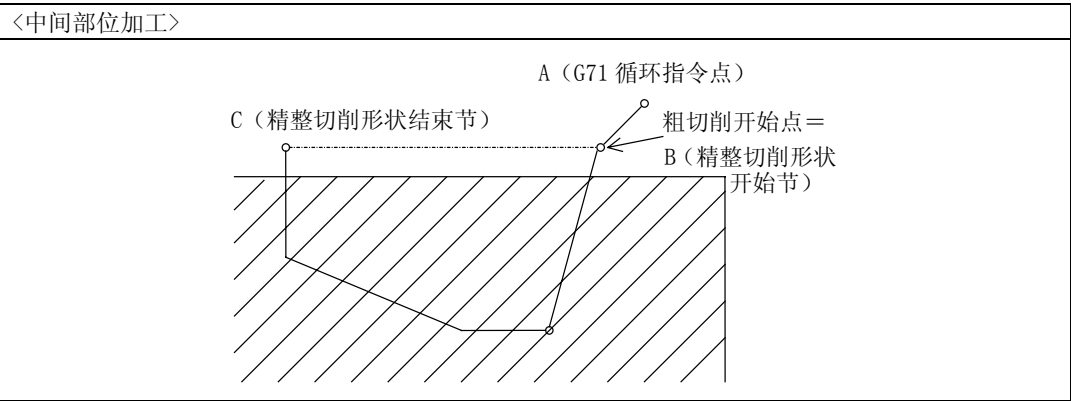
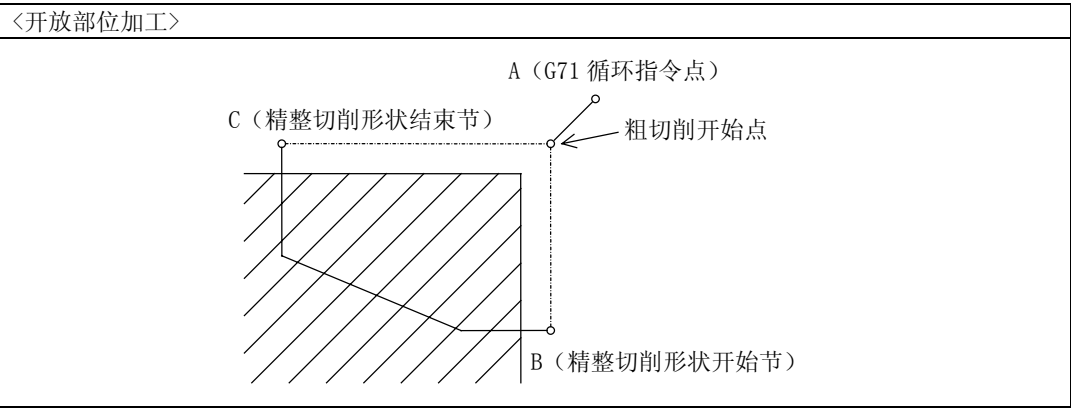
(2) 开放部位加工与中间部位加工

粗切削循环的加工中，包括工件一侧为开放部位时的加工，与从工件的中间开始进行的加工。

开放部位加工……袋形加工关闭时/打开时均可能出现

中间部位加工……袋形加工打开时可能出现。

(将整个粗切削循环看作为一个袋形加工)



(3) 粗切削方向

〈袋形加工关闭时的粗切削方向〉

选择以下任何一种。

- 根据精整切削形状自动决定 (#1273 ext09/bit2=0)

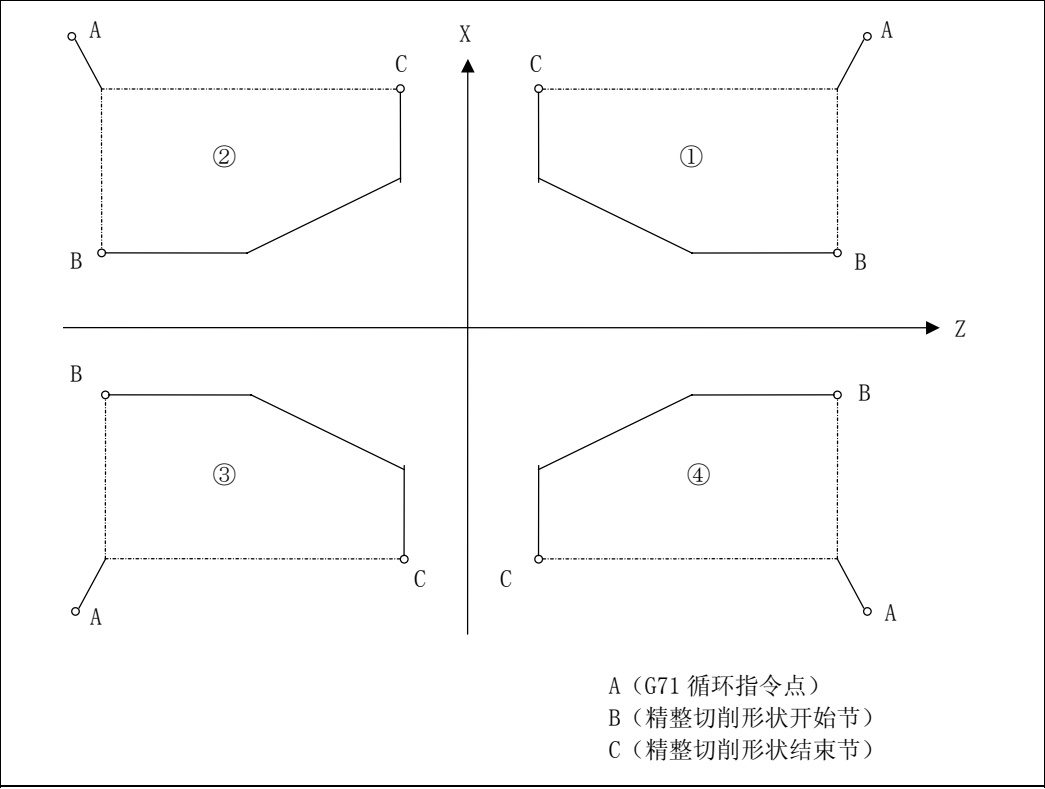
粗切削方向是根据精整切削形状，按照如下方式决定。精整切削形状程序为A→B→C。

C (精整切削形状结束节) 的 X 轴 > B (精整切削形状开始节) 的 Z 轴	①或②
C (精整切削形状结束节) 的 X 轴 < B (精整切削形状开始节) 的 Z 轴	②或④
C (精整切削形状结束节) 的 Z 轴 > B (精整切削形状开始节) 的 Z 轴	②或③
C (精整切削形状结束节) 的 Z 轴 < B (精整切削形状开始节) 的 Z 轴	②或④

※C (精整切削形状结束节) 的X轴 = B (精整切削形状开始节) 的X轴 或

C (精整切削形状结束节) 的Z轴 = B (精整切削形状开始节) 的Z轴

时，发生程序错误 (P203)。



- 在程序中指定 (#1273 ext09/bit2=1)

精整切削量的方向取决于程序中指定的精整切削量符号。

〈袋形加工打开时的粗切削方向〉

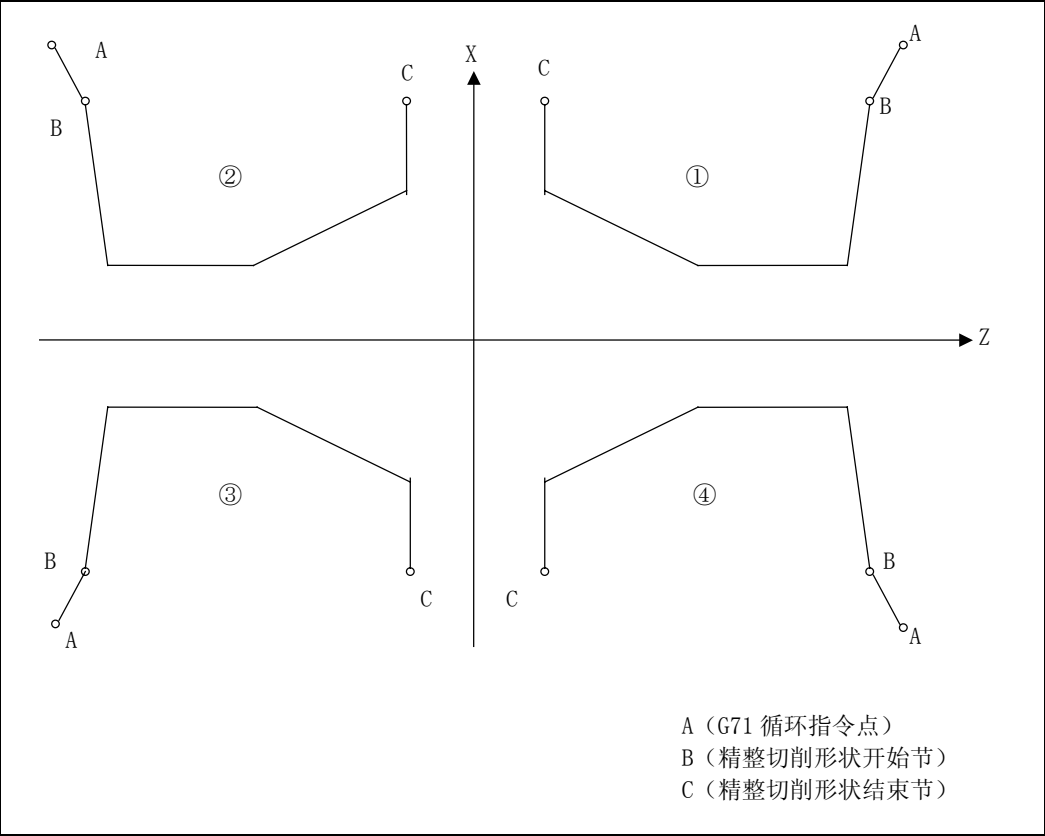
选择以下任何一种。

- 根据精整切削形状自动决定 (ext09/bit2=0)

粗切削方向是根据精整切削形状，按照如下方式决定。精整切削形状程序为A→B→C。

A (G71 循环指令点) 的 X 轴 > B (精整切削形状开始节) 的 X 轴 且 B (精整切削形状开始节) 的 X 轴 ≤ C (精整切削形状结束节) 的 X 轴	①或②
A (G71 循环指令点) 的 X 轴 < B (精整切削形状开始节) 的 X 轴 且 B (精整切削形状开始节) 的 X 轴 ≥ C (精整切削形状结束节) 的 X 轴	③或④
C (精整切削形状结束节) 的 Z 轴 > B (精整切削形状开始节) 的 Z 轴	②或③
C (精整切削形状结束节) 的 Z 轴 < B (精整切削形状开始节) 的 Z 轴	②或④

※C (精整切削形状结束节) 的 Z 轴 = B (精整切削形状开始节) 的 Z 轴
时，发生程序错误 (P203)。



※以B点的X轴为底部时，与“〈袋形加工关闭时的粗切削方向〉”相同。

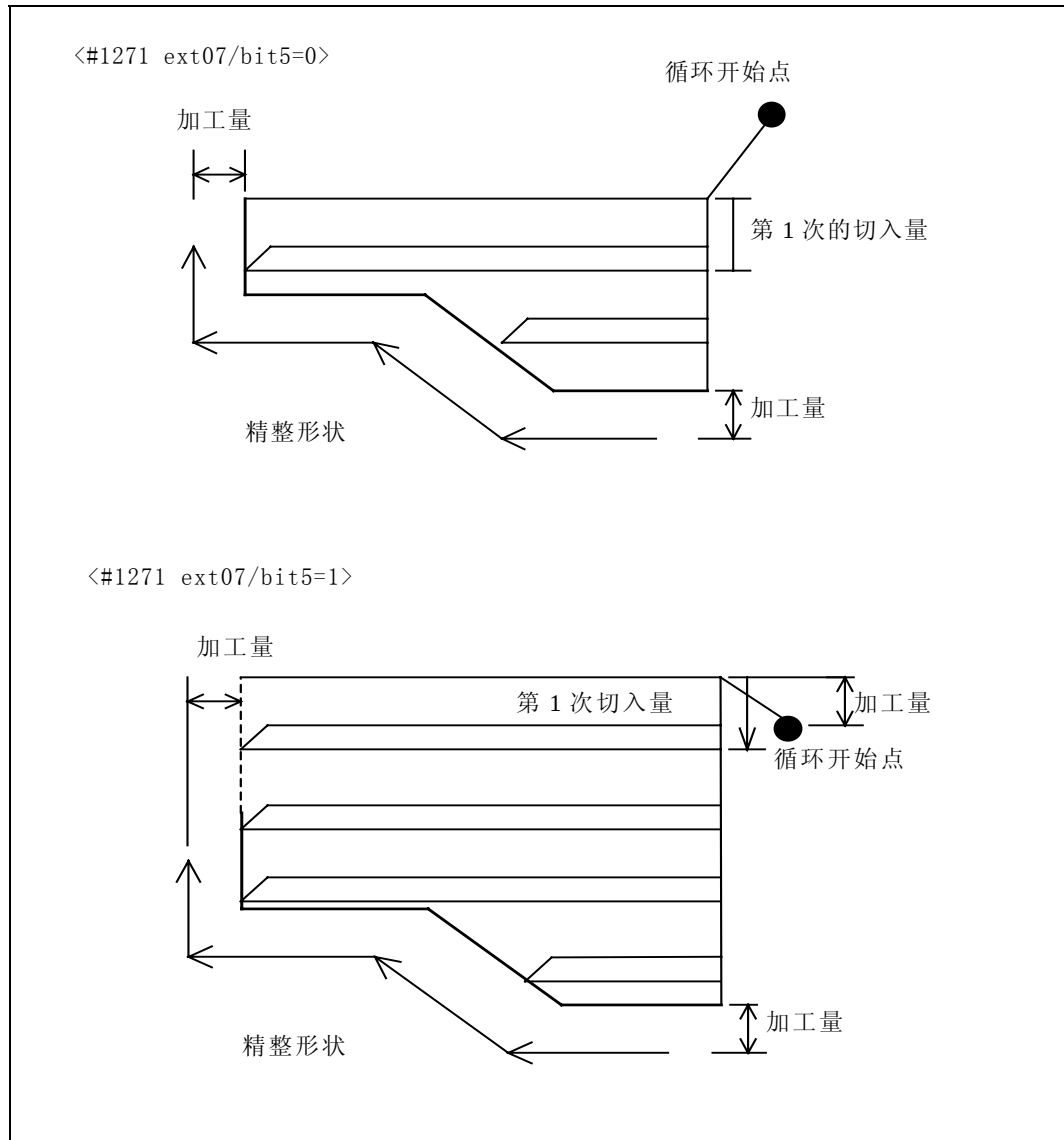
- 在程序中指定 (ext09/bit2=1)

精整切削量的方向取决于程序中指定的精整切削量符号。

(4) 精整切削形状

切入开始位置的选择 (#1271 ext07/bit5)

切入开始位置是根据精整切削形状程序的最终位置计算得出，可变为从循环开始点开始。



〈袋形加工关闭时的Z轴方向精整切削形状〉

Z轴方向的精整切削量，请务必单调变化（仅增加或仅减少）。

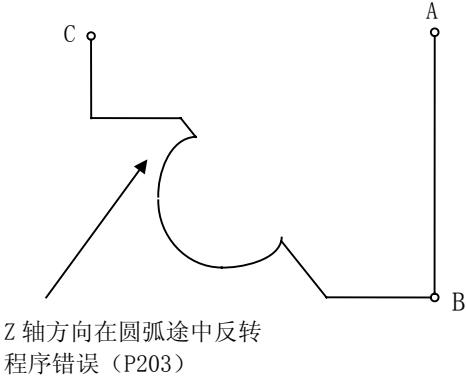
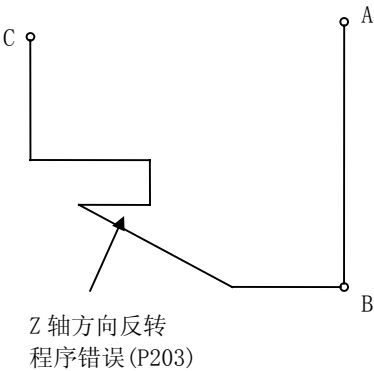
当形状不正确时，发生程序错误（P203）。

〈袋形加工关闭时的X轴方向精整切削形状〉

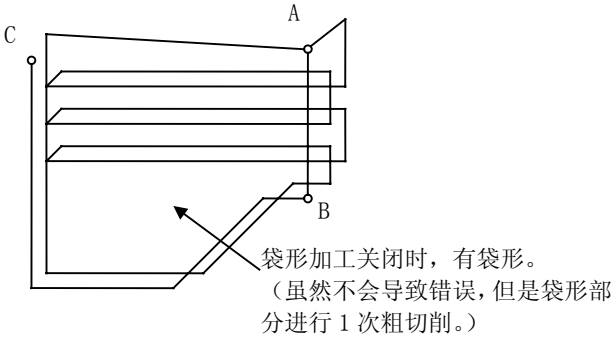
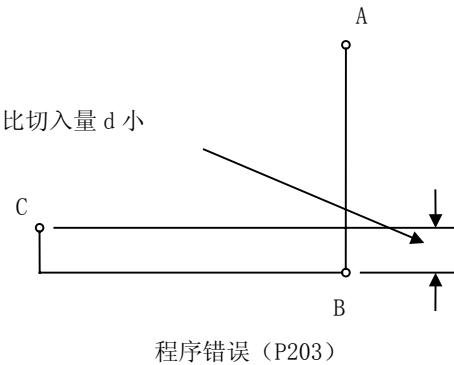
X轴方向的精整切削量，请确保变化的单调性（增加或减少）。

未进行单调变化的部分，不执行凹陷形状（袋形）部分的粗切削循环，但是精整切削量的描绘，是执行指定的形状，如果袋形的X轴深度大于切入量，则袋形部分的切削负载会增大，所以编程时请加以注意。

〈袋形加工关闭时：Z 轴方向的错误例〉



〈袋形加工关闭时：X 轴方向的错误例〉



〈袋形加工打开时的Z轴方向精整切削形状〉

Z轴方向精整切削量，请务必单调变化（仅增加或仅减少）。

非单调变化形状的部分，成为有遮盖的精整切削形状。

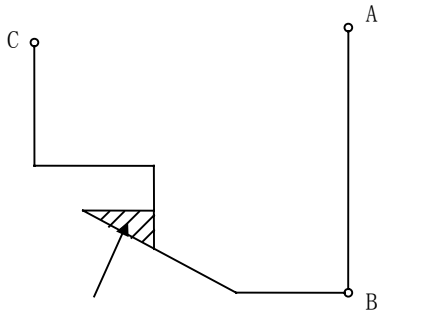
〈袋形加工打开时的X轴方向精整切削形状〉

请确保X轴方向精整切削形状为单调变化（增加或减少），仅凹陷形状（袋形）部分的方向相反。

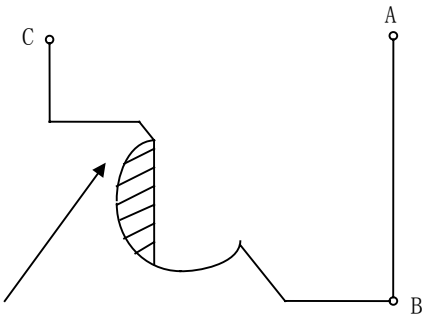
另外，在袋形加工中，请注意不要让精整切削形状中间节超过C（精整切削形状结束节）的X轴位置。

当中间节超过C（精整切削形状结束节）的X轴位置时，发生程序错误（P203）。

〈袋形加工打开时：Z轴方向的非单调变化部分例〉

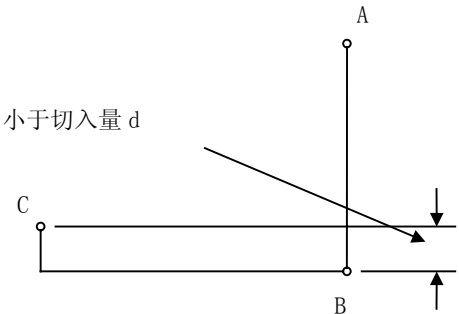


由于 Z 轴方向反转，所以创建精整切削形状时，遮盖住斜线部分。
（斜线部分不切削）

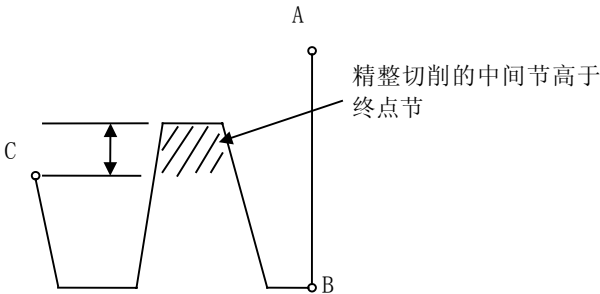


由于 Z 轴方向在圆弧的途中反转，所以创建精整切削形状时，遮盖住斜线部分。
（不切削斜线部分）

〈袋形加工打开时：X轴方向的错误例〉



程序错误（P203）



程序错误（P203）

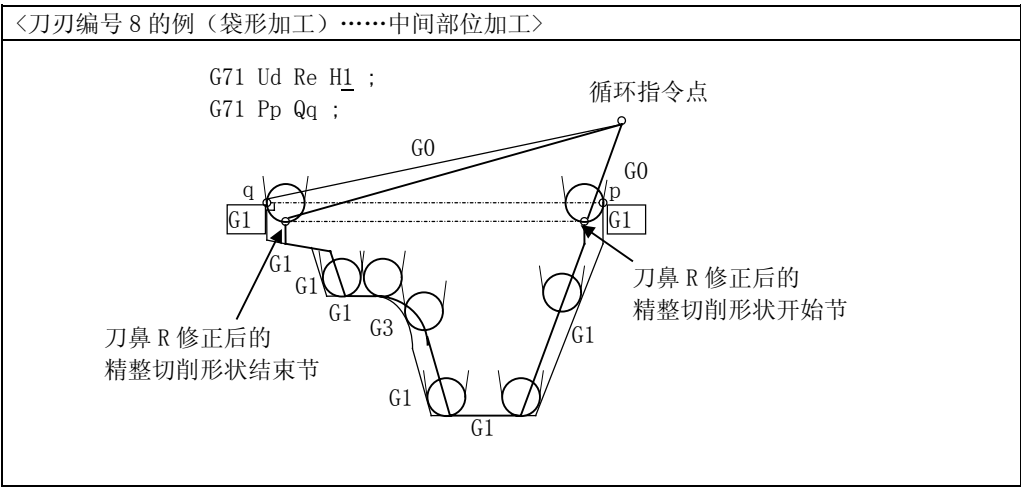
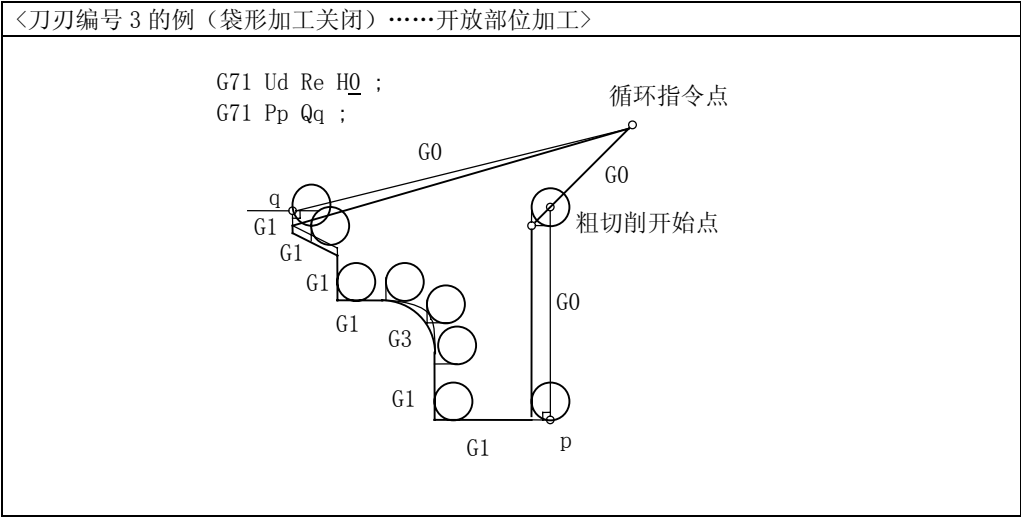
(5) 刀尖R补偿中的精整切削形状

刀尖R补偿的选择 (#1271 ext07/bit6)

<#1271 ext07/bit6=0>

刀尖R补偿中有G71指令，则G71循环指令点变为临时取消刀尖R补偿后的位置。在精整切削形状中，在进行刀尖R补偿的状态下创建精整切削形状。

(注) 当进行袋形加工打开时的中间部位加工时，请在进行刀尖 R 补偿后的状态下，确保精整切削形状开始节的 X 轴位置与精整切削形状结束节的 X 轴位置为相同值，进行编程。



在中间部位进行加工时的注意点

当刀尖R补偿后的精整切削形状开始节X轴位置，与精整切削形状结束节X轴位置为相同值时，进行中间部位加工。

作为编程方法，上图中以口包围起来的节，请仅进行X轴的移动（超过刀尖R直径的距离）。

<#1271 ext07/bit6=1>

将精整切削形状程序中，不进行刀尖R补偿的形状作为粗切削的形状。



切入量

(1) 虽然切入量是通过Ud加以指定，不过可通过使用参数设定切入变化量（ Δd ），每1次分别变化切入量。
切入量的指定（ext07/bit7）

<ext07/bit7=0>

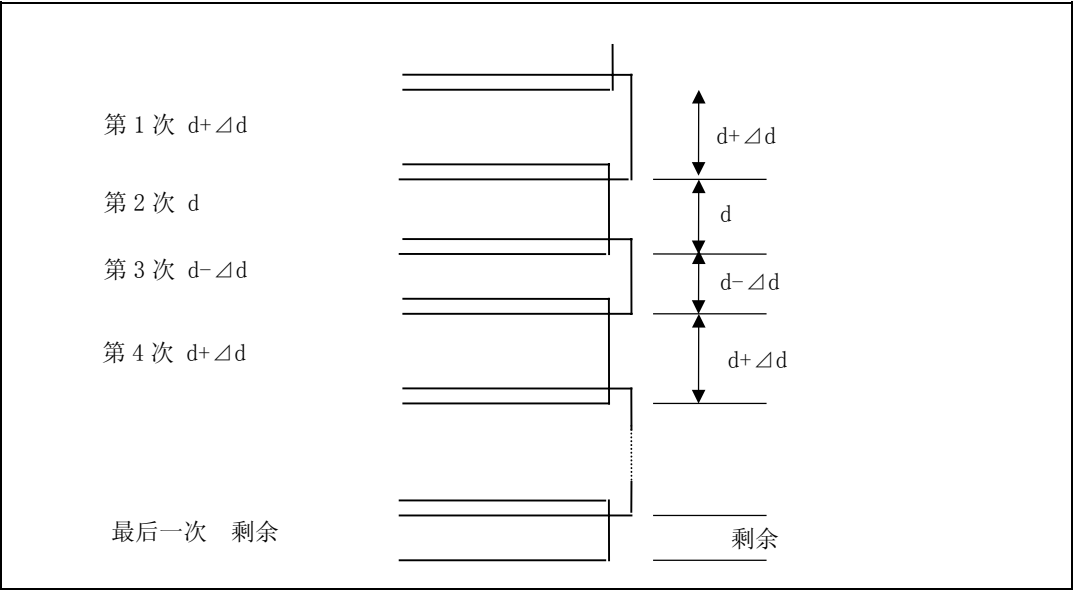
当程序中指定的一次切入量大于精整切削形状的切削深度时（ $d < \Delta d$ ），发生程序错误（P204）。

<ext07/bit7=1>

即使一次的切入量指定大于精整切削形状的切削深度，也不会发生程序错误，而是进行一次切削。
但是，一次的切入量指令，请控制在0~99.999mm的范围之内。超过该范围进行指令，则发生程序错误。

(2) 切入的最后，是以剩余的余量作为切入量，如果该剩余量小于参数中所设定的值，则不进行切入，而是执行精整粗切削。

参数 切入量（d）	[加工参数] “#8051 G71切入” 0~99.999mm
切入变化量（ Δd ）	[加工参数] “#8017 G71切入变化” 0~99.999mm
切入的最后最小切入量	[加工参数] “#8016 G71最小切入” 0~99.999mm





结束方式・退刀量

- (1) 结束方式因袋形加工开/关而异。
- 袋形加工关闭相对于工件，沿45° 方向退刀的方式
- 袋形加工开 描绘形状的方式
- (2) 以Re指定退刀量，可通过以参数设定间隙量的方式，指定退刀量。
- 袋形加工关闭 相对于工件，沿45° 方向退刀的方式
- 参数 退刀量 (e) [加工参数] “#8052 G71间隙” 0~99.999mm

<袋形加工关闭时>	
<袋形加工打开时>	
<ext08/bit0=0>	<ext08/bit0=1>

(注) ext08/bit0 同时也用于选择是否自动决定袋形加工，所以请加以注意。(请参阅“指令格式”的“Hh”的说明。)



加工量

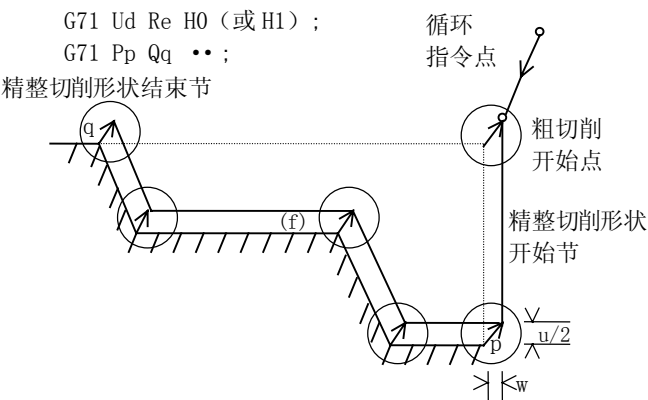
指定精整切削量，则对精整切削形状，保留 U_u/W_w 量的切削余量。

X轴精整切削量 在粗切削开始点方向上保留精整切削量。

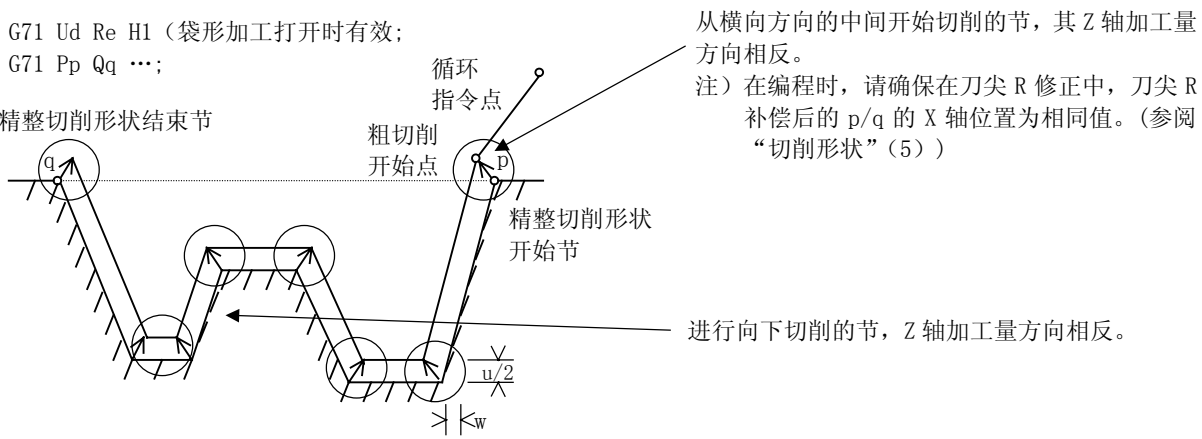
Z轴精整切削量 在粗切削开始点方向上保留精整切削量。

但是，袋形加工打开时的下降切削（X轴为指令了向孔底方向的节），与从横向方向的中间开始切削时，Z轴精整切削量方向反转。

<袋形加工关闭时的精整切削量 • 或袋形加工打开时的开放部位加工精整切削量
(精整切削形状开始节的 X 轴为底部) >



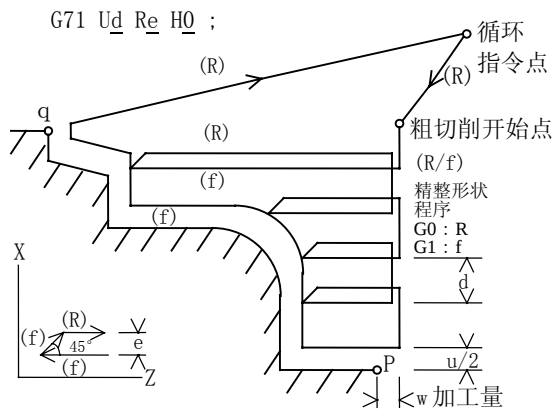
<袋形加工打开时的中间部位加工的精整切削量
(精整切削形状开始节的 X 轴 = 精整切削形状结束节的 X 轴) >





程序例

开放部位加工（袋形加工关闭的例）



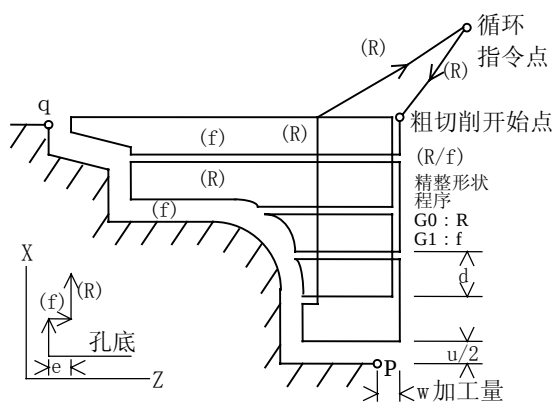
```

G0 X80.0 Z75.0 T0101;      ←①
G71 U10. R3. H0;          ←②
G71 P10 Q20 U3. W1.5 F500 S1500; ←③
N10 G0 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;               ←④
    
```

精整切削程序

程序执行顺序①②③④

开放部位加工（袋形加工打开的例）



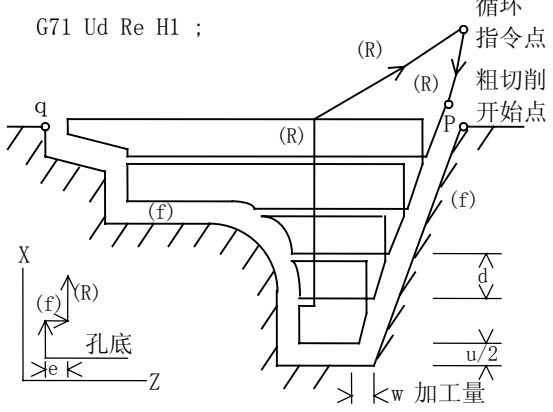
```

G0 X80.0 Z75.0 T0101;      ←①
G71 U10. R3. H1;          ←②
G71 P10 Q20 U3. W1.5 F500 S1500; ←③
N10 G0 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;               ←④
    
```

精整切削程序

程序执行顺序①②③④

中间部位加工（袋形加工打开的例）



```

G0 X80.0 Z75.0 T0101;      ←①
G71 U10. R3. H1;          ←②
G71 P10 Q20 U3. W1.5 F500 S1500; ←③
N10 G0 X60.0 Z73.0;
G1 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;               ←④
    
```

精整切削程序

程序执行顺序①②③④

Uu	: X轴方向上加工量（如省略，则视为X轴方向加工量=0） 单位 μ 取决于 m / 1/10000inch 直径/半径值指令（#1019 dia）。
Ww	: Z轴方向上加工量（如省略，则视为Z轴方向加工量=0） 单位 μ m / 1/10000inch 半径值指令
Ff	: 切削速度（如省略，则为 G72 之前的切削速度（模态））
Ss, Tt	: 主轴指令、刀具指令

※其他详细功能与“13.3.1 横向粗切削循环”同等。

13.3.3 型材粗切削循环：G73



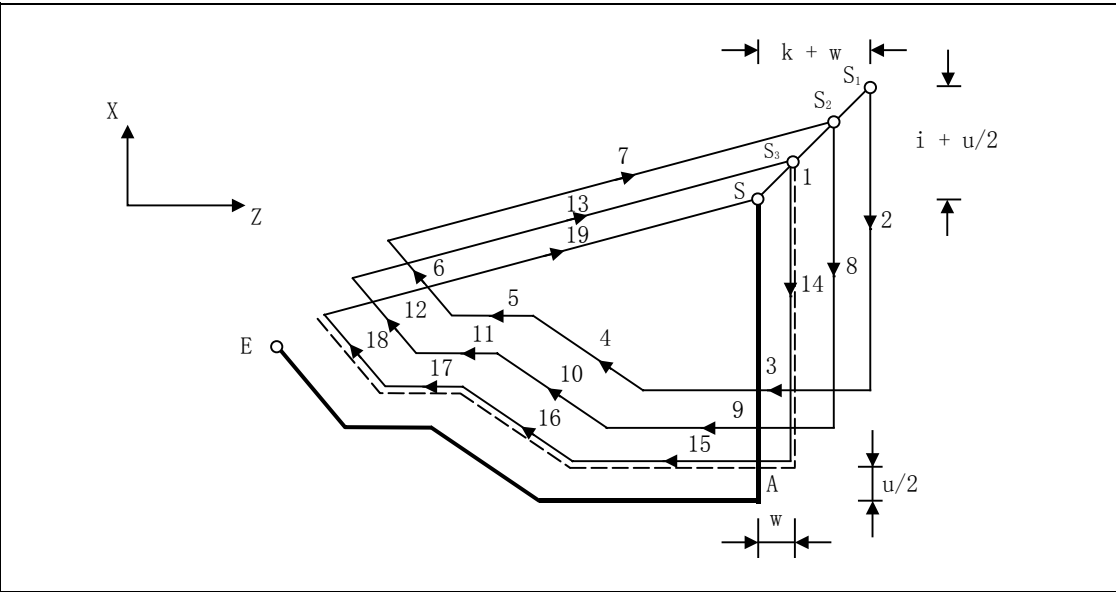
功能及目的

调用精整切削形状程序，自动计算中间轨迹的同时，按照精整切削形状进行粗切削。



指令格式

G73	Ui	Wk	Rd						;
G73	Aa	Pp	Qq	Uu	Ww	Ff	Ss	Tt	;
Ui	:	X 轴方向切削余量	i	}	• 无 P、Q 指令时，为切削余量。 • 为模态数据。 • 跳跃符号。 • 切削余量通过半径指定。				
Wk	:	Z 轴方向切削余量	k						
Rd	:	分割次数	d						
Aa	:	精整切削形状程序编号	(如省略则为本程序)						
Pp	:	精整切削形状开始顺序编号	(如省略则为首个程序)						
Qq	:	精整切削形状结束顺序编号	(如省略则为直到程序结束)						
但是，即使有 Qq，当有 M99 在前时，执行到 M99 为止									
Uu	:	X 轴方向精整切削余量	u	}	• 发出 P、Q 指令时为加工余量。 • 跳跃符号。 • 直径/半径指定可通过参数#1019 dia 改变。 • 移动的方向取决于形状。详情请参阅 G71 的“精整切削余量方向”。				
Ww	:	Z 轴方向精整切削余量	w						
Ff	:	切削速度 (F 功能)	精整切削形状程序中的 FST 指令被跳跃，粗切削指令中的值，或之前的值生效。						
Ss	:	主轴速度 (S 功能)							
Tt	:	刀具指令 (T 功能)							



(注) 单节时，在各节的终点停止。



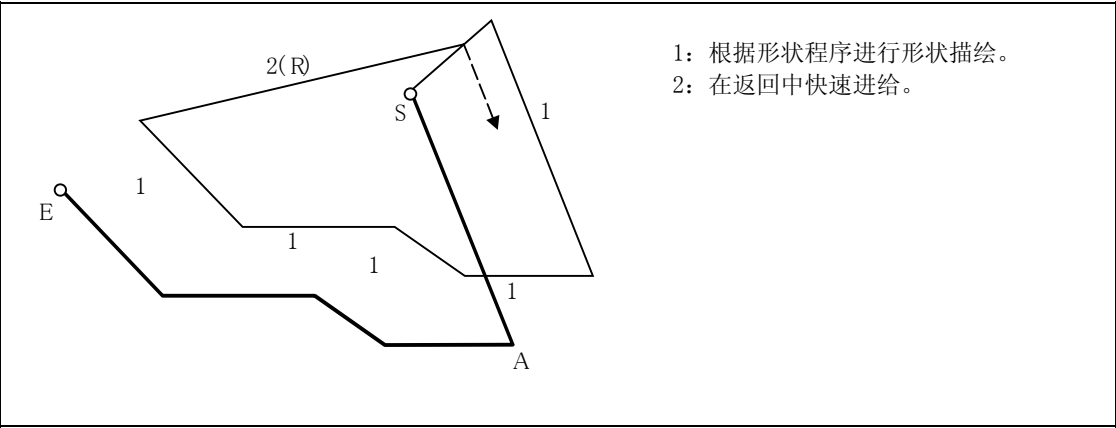
精整切削形状

在程序中，指令了上图的S→Z→E。
A→E之间，必须是X轴・Z轴方向上均是单调变化的形状。



1个循环构成

1个循环的构成如下。



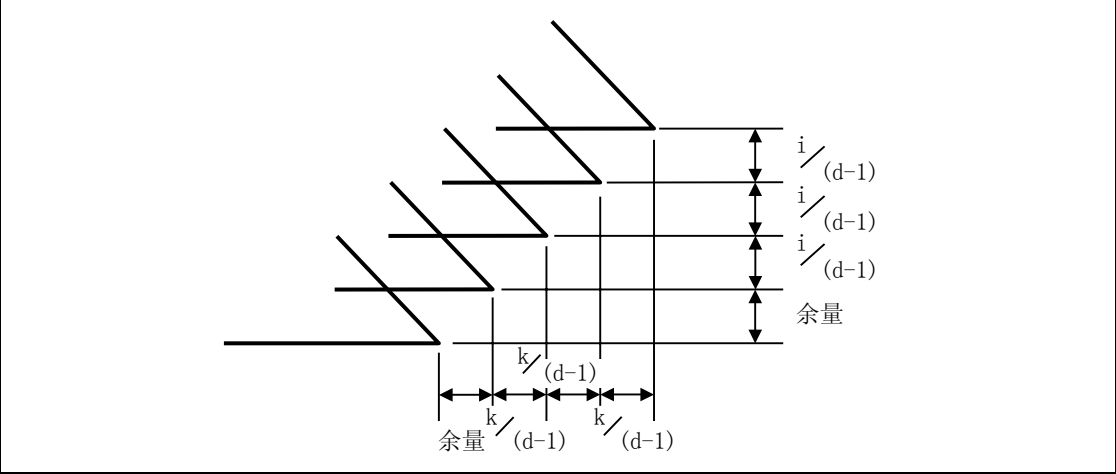
切入量

切入量为，用切削余量（i、k）除以分割次数（d-1）之后的值。

X轴方向 $i / (d-1)$

Z轴方向 $i / (d-1)$

但是，除不尽时，进行结束，在最后一次切削时进行调整。





刀尖 R 补偿

在刀尖R补偿模式下指令本循环，则对作为本循环对象的精整切削形状程序进行刀尖R补偿，针对该形状执行本循环。但是，在刀尖R补偿模式下指令本循环，则在本循环之前临时取消刀尖R补偿，在精整切削形状程序的开头节开始补偿。



其他

(1) 切入方向

- 在程序中指定（#1273 ext09/bit2=0）

用于切入的移位方向，根据精整切削程序的形状，如下表所示加以决定。

	1	2	3	4
图示				
起始 X 轴	-方向	-	+	+
整体的 Z 轴	-方向	+	+	-
X 轴 切削	+方向	+	-	-
Z 轴 切削	+方向	-	-	+

- 通过指令的精整切削余量/切削余量决定（ext09/bit2=1）

用于切入的移位方向，取决于程序中指定的切削余量等的符号。

13.3.4 精整切削循环：G70



功能及目的

通过G71~G73指令进行粗切削之后，可根据以下的指令进行精整切削。



指令格式

G70 A__ P__ Q__ ;

A

:

精整切削形状程序编号（如省略则为执行中的程序）

P

:

精整切削形状开始顺序编号（如省略则为程序开头）

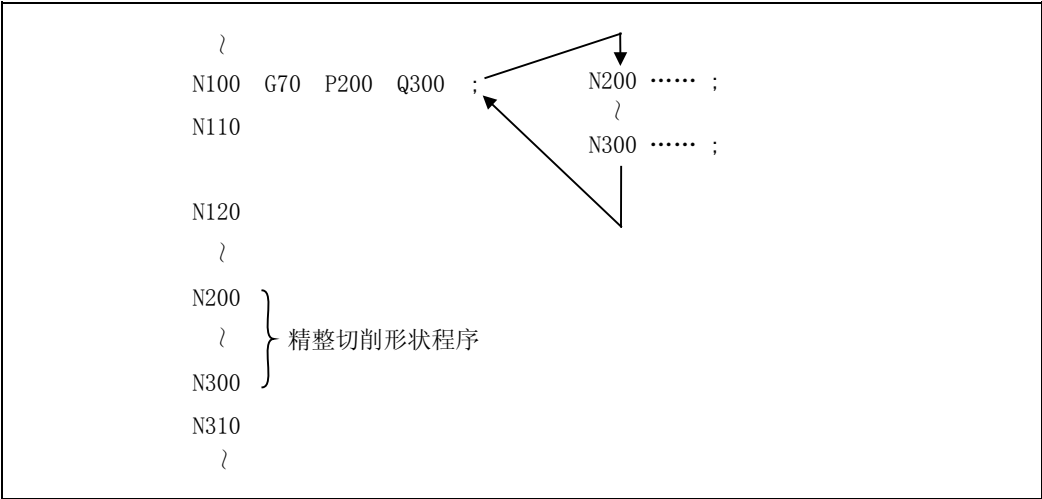
Q

:

加工形状结束顺序编号（如省略则到程序结束为止）

但是，即使有 Q 指令，当 M99 在前时，则执行到 M99 为止。

- （1）在精整切削循环中，精整切削形状程序中的F、S、T指令有效。
 - （2）G70的循环结束，则刀具通过快速进给返回起点，下一节被读入。
- （例 1）顺序编号指定时



13.3.5 端面槽棒切削循环；G74



功能及目的

G74是通过指定槽棒终点的坐标、切入量、位组的移位量、在槽底上的位组后移量，自动在棒材的端面方向执行槽棒加工的固定循环。加工程序的指令方法如下。



指令格式

G74 Re ;
G74 X/(U)x Z/(W)z Pi Qk Rd Ff ;

Re

: 返回量（无 X/U、P 指令时）（模态）

X/Ux

: B 点 X 坐标（绝对值/增量值）

Z/Wz

: B 点 Z 坐标（绝对值/增量值）

Pi

: 刀具的移位量（以半径指定，增量值，不需符号）

Qk

: 切入量（以半径指定，增量值，不需符号）

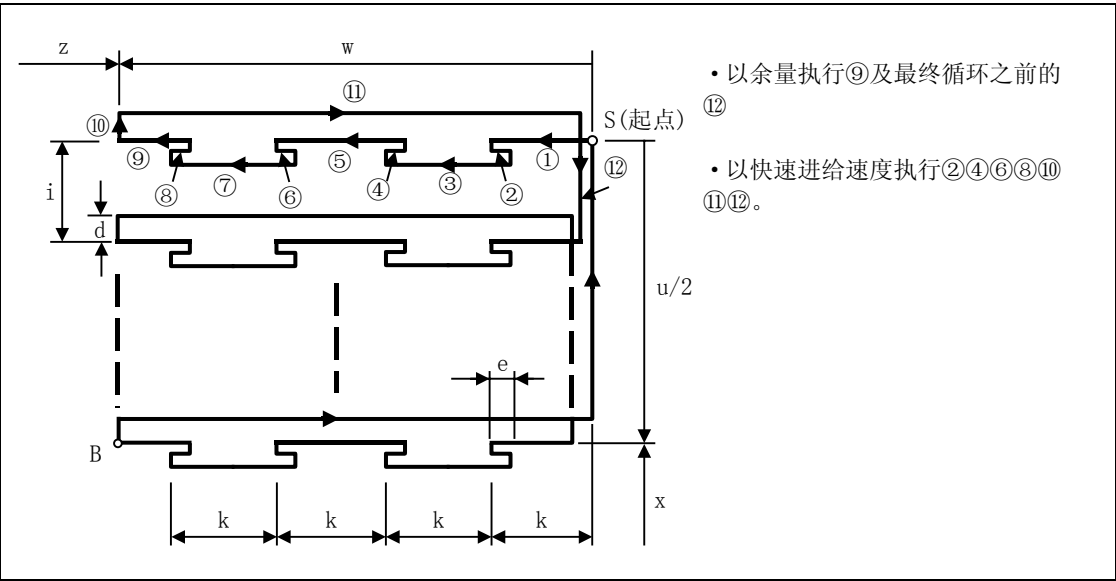
Rd

: 槽底位置的退刀量

当无符号时，在第 1 次加工出的沟槽底部也进行退刀，如果带一号，则第 1 次的沟槽底部不进行退刀，从第 2 次开始退刀。

Ff

: 进给速度





单节停止

在①～⑫的各节停止。



其他

- (1) 当省略X/U、P或x、i的值为0时，变为仅Z轴的动作。但是，有Rd指令但是无符号时，则在槽底执行退刀。
- (2) 当没有X/U、Z/W指令时，看作为参数设定指令（G74Re）。即使指定G74 Pi Qk Rd ； 时，也将Rd看作为Re，设定返回量。
- (3) Rd的指定中，无论是带-号，还是不带符号，退刀方向均不变。
- (4) 以下情况下，发生程序错误（P204）。
 - (a) 指令了X/U，但是I=0或未指令P时
 - (b) 刀具移位量I大于x的移动量时
 - (c) 退刀量d大于移位量i时
 - (d) 返回量e大于切入量k时
 - (e) 切入量k大于孔深度w时

13.3.6 横向槽榫切削循环；G75



功能及目的

G75是通过指定槽榫终点的坐标、切入量、位组的移位量、在槽底上的位组退刀量，自动在棒材的横向方向执行槽榫加工的固定循环。



指令格式

G75 **Re ;**

G75 **X/(U)x Z/(W)z Pi Qk Rd Ff ;**

Re : 返回量（无 X/U、P 指令时）（模态）

X/Ux : B 点 X 坐标（绝对值/增量值）

Z/Wz : B 点 Z 坐标（绝对值/增量值）

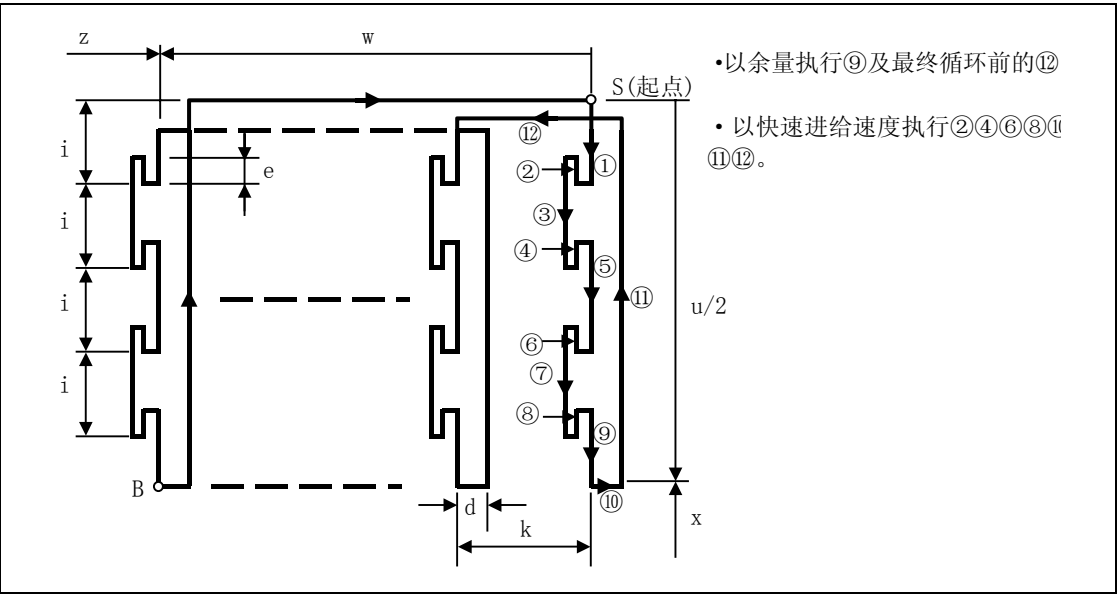
Pi : 切入量（以半径指定，增量值，不需符号）

Qk : 刀具的移位量（以半径指定，增量值，不需符号）

Rd : 槽底位置的退刀量

Ff : 进给速度

当无符号时，在第 1 次加工出的沟槽底部也进行退刀，如果带一号，则第 1 次的沟槽底部不进行退刀，从第 2 次开始退刀。





单节停止

在①～⑫的各节停止。



其他

- (1) 当省略Z/W、Q或z、k的值为0时，变为仅X轴的动作（槽榫加工）。但是，有Rd指令但是无符号时，则在槽底执行退刀。
- (2) 当没有X/U、Z/W指令时，看作为参数设定指令（G75Re）。即使指定G75 Pi Qk Rd；时，也将Rd看作为Re，设定返回量。
- (3) Rd的指定中，无论是带-号，还是不带符号，退刀方向均不变。
- (4) 以下情况下，发生程序错误（P204）。
 - (a) 指令了Z/W，但是k=0或未指令Q时
 - (b) 刀具移位量k大于z的移动量时
 - (c) 退刀量d大于移位量k时
 - (d) 返回量e大于切入量i时
 - (e) 切入量i大于孔深度u/2时

13.3.7 复合型螺纹切削循环：G76



功能及目的

G76是通过指定螺纹切削开始点、螺纹切削终点，能够在任意角度进行切入，并使每一次的切削截面积（切削扭矩）保持恒定的自动切入固定循环。
通过考虑螺纹终点坐标及锥轴高度成分的指令值，能够切削出各种横向的螺纹。



指令格式

G76	Pmra	Q	Δdmin	Rd	;		
G76	X/U	Z/W	Ri	Pk	QΔd	Fl	;
m	: 精整切削的切入次数 00~99（模态）						
r	: 端面倒角量 00~99（模态） 在 0.0I~9.9I 的范围内，以省略小数点的 2 位整数，指定基于螺纹导程 I 的切削量。						
a	: 刀刃角度（螺纹角度） 00~99（模态） 以 1 度为单位，指令 0° ~99° 的角度。 上述 m、r、a 为地址 P 中的指令。 （例）m =5、r = 1.5、a = 0° 时，P 值为 P051500，也就是变为 P051500，前后的 0 均不可省略。						
Δdmin	: 最小切入量 当计算出的切入量小于 Δdmin 时，以 Δdmin 进行钳位。（省略时的动作因#1222/bit4 的设定而异。）						
d	: 精整切削余量（模态）						
X/U	: 螺纹部位的 X 轴终点坐标 以绝对值或增量值指令螺纹部位内部终点的 X 坐标。						
Z/W	: 螺纹部位的 Z 轴终点坐标 以绝对值或增量值指令螺纹部位内部终点的 Z 坐标。						
i	: 螺纹部位的锥轴高度成分（半径值） i = 0 时，为直螺纹。						
k	: 螺纹高度 以正的半径值指令螺纹高度。						
Δd	: 切入量 以正指令第 1 次的切入量。						
l	: 螺纹导程						

(注 1) 上述 2 个 G76 指令不可汇总到 1 单节中。
通过 P、Q、R 指令的数据，根据轴地址 X/U、Z/W 的有无自动进行判别。

(注 2) 上述模态数据 r 也可使用参数（#8014 倒角量）的设定值，不过参数设定值会随着程序指令的改变而被覆盖。

(注 3) 倒角量的指定，在螺纹切削固定循环中也有效。

(注 4) 以下情况下，发生程序错误 (P204)。

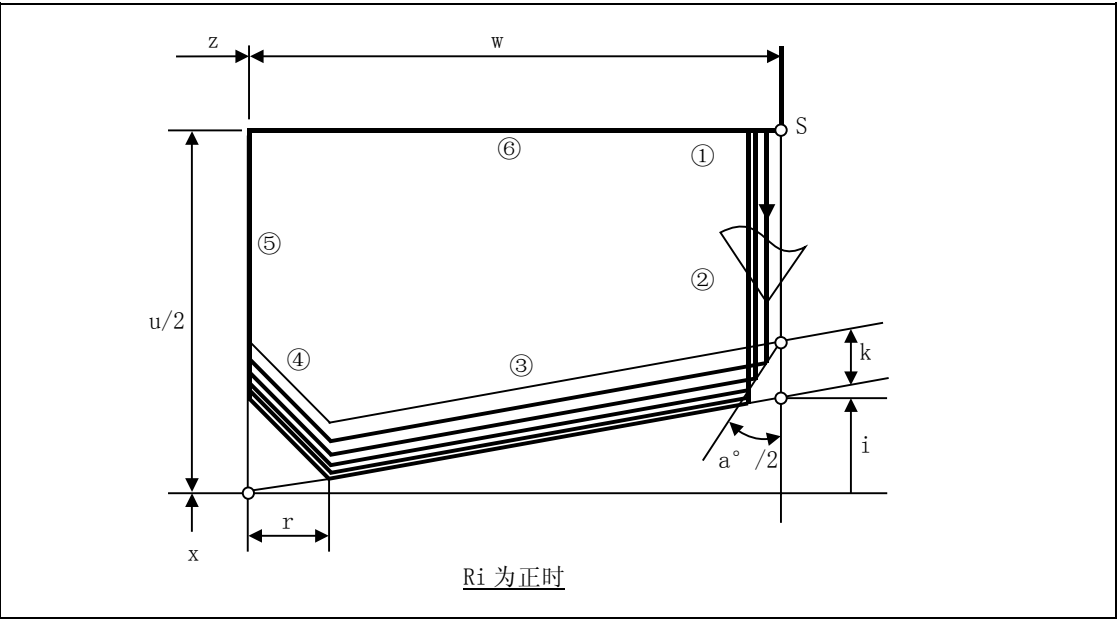
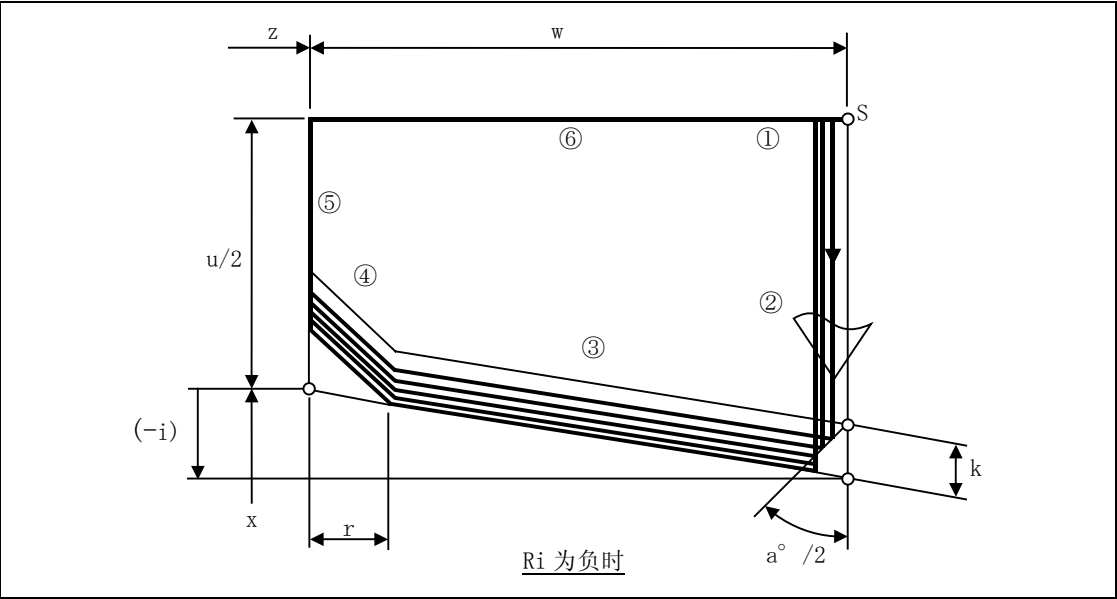
- (a) a 超过规定值时
- (b) 当没有指令 X 指令与 Z 指令中的任何一个时，或是 X 指令与 Z 指令中的任何一个的起点坐标与终点坐标相同时
- (c) 螺纹比到达螺纹底部之前的 X 轴移动更大时

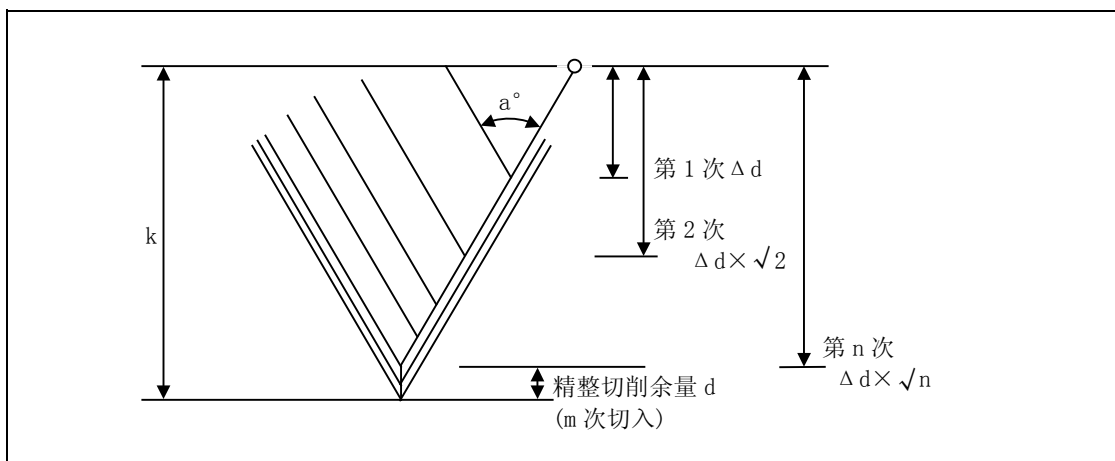
(注 5) 也取决于螺纹切削指令 (G33)、螺纹切削循环 (G78) 的注意事项。



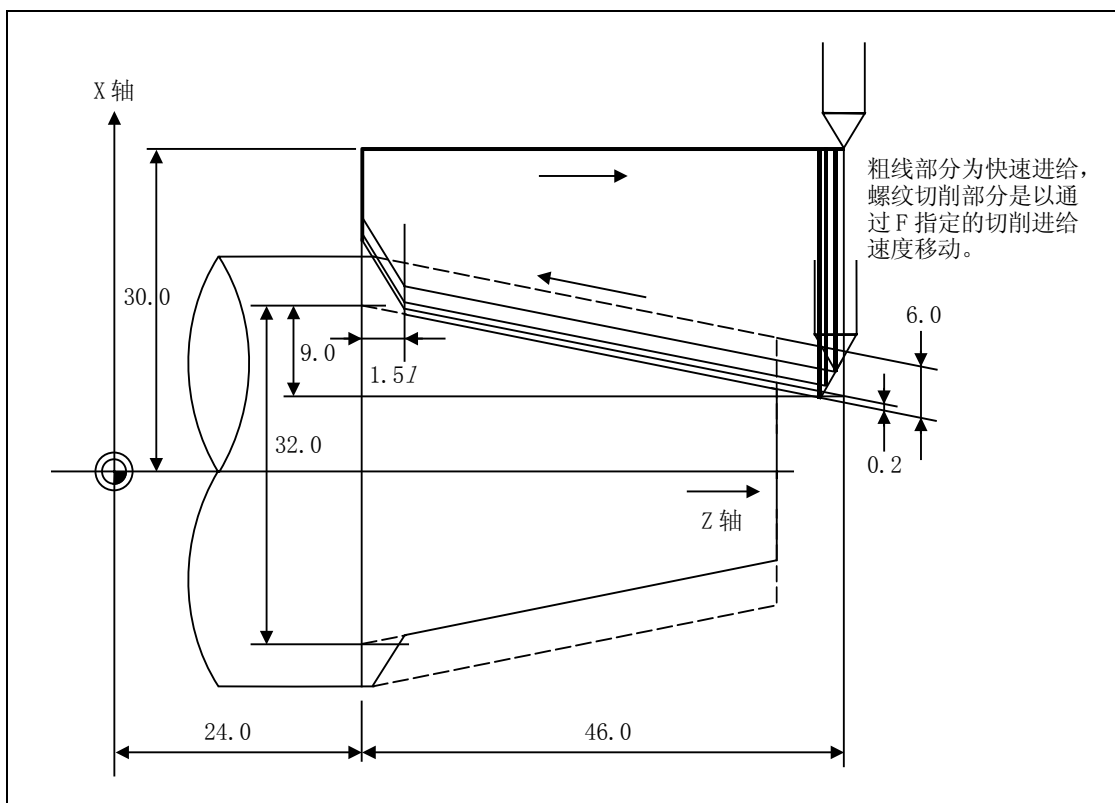
1 个循环的构成

1 个循环中的①②⑤⑥为快速进给，③④为根据 F 指定以切削进给进行移动。





程序例



```
G76 P011560 R0.2;
```

```
G76 U-28.0 W-46.0 R-9.0 P6.0 Q3.5 F4.0;
```



插入动作

- (1) 在执行G76时按回馈等待按钮，则在螺纹切削中时，执行完非螺纹切削单节之后，自动运转停止。（自动运转休止中指示灯会立即点亮，自动运转停止时熄灭。）
当不处于螺纹切削中时，以及从开始执行螺纹切削指令，到轴开始移动之间，自动运转休止中指示灯点亮，进入自动运转休止状态。
- (2) 在执行G76中切换为其他自动运转模式时、从自动运转切换为手动运转时、或是单节运转中，在完成①④⑤时停止。
- (3) 另外，在螺纹切削中，执行G76中的空运转的有效/无效不会发生变化。

13.3.8 复合型车削用固定循环（G70～G76）注意事项



注意事项

- (1) 请在复合型车削用固定循环的指令单节中，指令所有必须的参数。
- (2) 如果精整切削形状程序已注册到了内存中，则可在内存运转、MDI运转、纸带运转等各种模式下，执行复合型车削用固定循环 I。
- (3) 执行G70～G73时，在该程序内，请注意不要让通过P、Q指定的精整切削形状程序的顺序编号重复。
- (4) 在G71～G73的单节中，通过P、Q指定的精整切削形状程序，请在200单节之内，创建包括转角倒角、转角R指令及通过刀尖R补偿自动插入单节在内的所有指令。超过200单节时，发生程序错误（P202）。当将切入开始位置作为循环开始点时（#1271 ext07/bit5=1），精整切削形状程序的最大单节数为199单节。（当循环开始点位于精整切削形状程序最终位置的外侧时）
- (5) 请将G71～G73单节中指定的精整切削形状程序，编程为X轴、Z轴均是单调变化（仅增加或仅减少）的程序。
- (6) 精整切削形状程序中，无移动的单节被跳跃。
- (7) 精整切削形状程序中，N、F、S、M、T被跳跃。
- (8) 当精整切削形状程序内有如下指令时，发生程序错误（P201）。
 - (a) 参考点回归相关（G27、G28、G29、G30）
 - (b) 螺纹切削（G33）
 - (c) 固定循环
 - (d) 跳跃功能（G31、G37）
- (9) 如果在精整切削形状程序内有子程序调用、宏调用指令，则这些也执行。
- (10) 单节运转时，除螺纹切削循环外，在各单节的终点（起点）停止。

(11) G71、G72、G73指令结束时，在顺序编号指定的场合以及程序编号指定的场合，下单节有所不同，请加以注意。

(a) 顺序编号指定时	(b) 程序编号指定时
下一单节为通过 Q 指定的下一单节。 ? N100 G71P200 Q500 U_W_... ; N200 N300 N400 } 精整切削形状程序 N500 N600 循环结束后，前进到 N600 单节。	下一单节为循环指令的下一单节。 ? N100 G71A100 U_W_... ; N200 N300 N400 O100 N10X100. Z50. ; N20 : : : : 循环结束后，前进到 N200 单节。

(12) G70指令结束时的下一单节，为指令单节的下一单节。

<pre>{ N100 ; N200 ; N300 ; N400 ; N500 ; } N100G70P200 Q500 ; (或 G70A100 ;) N1100..... ; }</pre> G70 指令结束后，前进到 N1100 单节。

(13) 在执行复合型车削用固定循环（G70～G76）时，虽然可以进行手动插入，但是在插入结束后，请务必返回到进行手动插入的位置，然后重新启动复合型车削用固定循环。

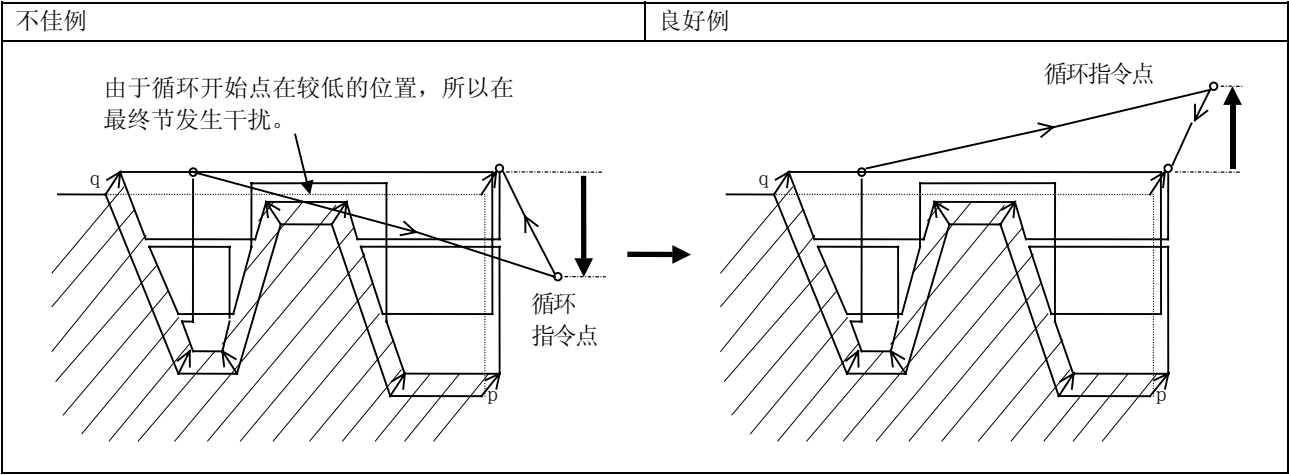
如果不返回就重新启动，则之后的动作会偏移手动插入量进行操作。

(14) 复合型车削用固定循环为非模态。因此，请根据需要进行指令。

(15) 在G71、G72中，当因为刀尖R补偿，第2单节中没有Z轴移动时，或是Z轴反向移动时，均发生程序错误（P203）。

(16) G70～G73中，当精整切削形状程序位于同一单节内时，当P及Q两者均未指定时，发生程序错误（P204）。但是，在G71～G73中，A、P、Q全部未指定时，看作为复合型车削用固定循环 I 的第1单节，不发生错误。

(17) 在决定循环开始点时，请确保复合型车削用固定循环 I 的最终单节为从工件切削面退刀的方向。为了防止在工件的凸出部位与刀具发生干扰，这一点是必须满足的。



(18) 复合型车削用固定循环的类型 II（袋形加工）为E68系统的规格。在E60系统上，即使指令了H1，也与H0进行相同的动作。

13. 程序援助功能

13.4 复合型车削用固定循环（MELDAS 特殊格式）

13.4 复合型车削用固定循环（MELDAS特殊格式）

[E68]



功能及目的

本功能可通过在1节中指令程序，执行事先准备好的固定循环。

MELDAS特殊格式时（#1265 ext01/bit0 打开），部分地址与通常的格式不同。另外，通常格式中，G71～G76是通过2节进行指令，而在MELDAS特殊格式中，则是通过1节进行指令。在本节中，对与通常格式不同的部分加以说明。

复合型车削用固定循环的详情，请参阅“13.3 复合型车削用固定循环”。



指令格式

G71 P__ Q__ U__ W__ D__ F__ S__ T__ ;
(G72 也相同。)

P	: 精整切削形状开始顺序编号
Q	: 精整切削形状结束顺序编号
U	: X 轴方向精整切削余量
W	: Z 轴方向精整切削余量
D	: 切入量
F	: 切削速度
S	: 主轴速度
T	: 刀具指令

G73 P__ Q__ U__ W__ I__ K__ D__ F__ S__ T__ ;

P	: 精整切削形状开始顺序编号
Q	: 精整切削形状结束顺序编号
U	: X 轴方向精整切削余量
W	: Z 轴方向精整切削余量
I	: X 轴方向切削余量
K	: Z 轴方向切削余量
D	: 分割次数
F	: 切削速度
S	: 主轴速度
T	: 刀具指令

G74 X(U)__ Z(W)__ I__ K__ F__ D__ ;
(G75 也相同。)

X (U)	: X 轴的槽榫终点坐标
Z (W)	: Z 轴的槽榫终点坐标
I	: 刀具移位量 (X/U 轴方向的移动量)
K	: 切入量 (Z/W 轴方向的切入量)
F	: 切削速度
D	: 在槽榫底部的刀具退刀量

G76 X (U) _ Z (W) _ I _ K _ D _ F _ A _ Q _ P _ ;	
X (U)	: 螺纹部位的 X 轴终点坐标
Z (W)	: 螺纹部位的 Z 轴终点坐标
I	: 螺纹部位的半径值
K	: 螺纹高度
D	: 第 1 次切入量
F	: 螺纹导程
A	: 螺纹角度
Q	: 攻丝开始移位角度 (0.001~360.000[°])
P	: P2 指令时, 为反向螺纹



详细说明

(1) 指令格式检查

检查是否有通常指令格式与MELDAS特殊格式并存。

[选择通常格式时 (#1265 ext01/bit0 关闭)]

(a) 指令格式的检查

当格式检查选择参数 (#1222 aux06/bit5) 为0时, 在以下场合下, 发生程序错误 (P33)。

- 无第1节指令。
- 进行了MELDAS特殊格式的指令。

(b) 指令地址的检查

在以下场合下, 发生程序错误 (P32)。

- 在G71、G72、G73、G74、G75节中指令了I、K、D中的任何一个。
- 在G76节中指令了I、K、D、A中的任何一个。

[选择MELDAS特殊格式时 (#1265 ext01/bit0 打开)]

(a) 指令格式的检查

进行通常格式的指令, 则发生程序错误 (P33)。

(b) 指令地址的检查

在以下场合下, 发生程序错误 (P32)。

- 在G71、G72、G73节中指令了R、A中的任何一个。
- 在G74、G75节中指令了P、Q、R中的任何一个。
- 在G76节中指令了地址R。

13. 程序援助功能

13.4 复合型车削用固定循环（MELDAS 特殊格式）

（2）MELDAS特殊格式与通常格式的比较

通常格式是通过2节进行指令，而MELDAS特殊格式则是通过1节进行指令。另外，部分地址不同。根据地址不同，也可能通过参数进行指定。

功能	MELDAS 特殊格式	通常格式	与通常格式的差异
横向粗切削循环 I	G71 P Q U W D F S T;	(1) G71 U R; (2) G71 A P Q U W F S T;	切入量 通过 U (1) → D 指定 退刀量 R (1) → 参数 (#8052 退刀) 无 A 指令 (精整切削形状程序编号)
端面粗切削循环 I	G72 P Q U W D F S T;	(1) G72 W R; (2) G72 A P Q U W F S T;	切入量 通过 W (1) → D 指定 退刀量 R (1) → 参数 (#8052 退刀) 无 A 指令 (精整切削形状程序编号)
形材粗切削循环	G73 P Q U W I K D F S T;	(1) G73 U W R; (2) G73 A P Q U W F S T;	X 轴切削余量 U (1) → I Z 轴切削余量 W (1) → K 分割次数 R (1) → D 无 A 指令 (精整切削形状程序编号)
端面槽榫切削循环	G74 X Z I K F D; 或 G74 U W I K F D;	(1) G74 R; (2) G74 X Z P Q R F; 或 (1) G74 R; (2) G74 U W P Q R F;	X 轴方向的移动量 P (2) → I Z 轴方向的切入量 Q (2) → K 在槽榫底部的退刀量 R (2) → D 返回量 R (1) → 参数 (#8056 G74 返回)
外径冲孔循环	G75 X Z I K F D; 或 G75 U W I K F D;	(1) G75 R; (2) G75 X Z P Q R F; 或 (1) G75 R; (2) G75 U W P Q R F;	X 轴方向的切入量 P (2) → I Z 轴方向的移动量 Q (2) → K 在槽榫底部的退刀量 R (2) → D 返回量 R (1) → 参数 (#8056 G74 返回)
复合型攻丝循环	G76 X_ Z_ I_ K_ D_ F_ A_ Q_ ;	(1) G76 Pmra R_ ; (2) G76 X_ Z_ R_ P_ Q_ F_ ;	螺纹部位的半径值 R (2) → I 螺纹高度 P (2) → K 第 1 次的切入量 Q (2) → D (MELDAS 特殊格式的 Q 为攻丝开始移位角度) 螺纹的角度 P (1) a → A 螺纹的切削 P (1 日) → 参数 (#8014 倒角量) 精整切削切入次数 P (1) m → 参数 (#8058 次数) 精整切削余量 R (1) → 参数 (#8057 精整切削余量)



限制与注意事项

- （1）请务必在G71～G73中指定P及Q。当有任何一个未指定时，发生程序错误（P204）。
- （2）攻丝开始移位角度并非模态。当G76中没有Q指令时，看作为“Q0”。
- （3）当在G76的Q中指令了超过360.000的值时，作为“Q360.000”加以使用。
- （4）G76在1个循环中进行1条切削。当进行2条切削时，请变更Q的值，执行同一指令。

13.5 钻孔用固定循环：G80～G89



功能及目的

钻孔用固定循环，是通过1节的指令，按照规定的作业顺序执行常规定位与钻孔、镗孔、攻牙等加工程序的功能。

固定循环包括如下种类。

（注）横向钻孔固定循环是将 X 轴指定为钻孔轴，不过可通过横向钻孔轴选择功能将 Y 轴指定为钻孔轴。

表中的“类型 II”是使用横向钻孔轴选择功能时的内容。

类型 II 是 E68 的规格。因此，E60 无法使用。

G 代码	钻孔轴	钻孔作业开始	在孔底的动作	返回动作	用 途	类型 II	
						横向钻孔轴选择信号（Y35C）的 ON/OFF 状态	钻孔轴
G80	...				取消	—	—
G83	Z	切削进给 间歇进给	就位检查 停止	快速进给	深孔钻孔循环 1	ON	Z
						OFF	
G84 (G84.1)	Z	切削进给	就位检查 停止 主轴反转	切削进给	攻牙循环 (反向攻牙循环)	ON	Z
						OFF	
G85	Z	切削进给	就位检查 停止	切削进给	镗孔循环	ON	Z
						OFF	
G87	X	切削进给 间歇进给	就位检查 停止	快速进给	深孔镗孔循环 1	ON	Y
						OFF	X
G88 (G88.1)	X	切削进给	就位检查 停止 主轴反转	切削进给	攻牙循环 (反向攻牙循环)	ON	Y
						OFF	X
G89	X	切削进给	就位检查 停止	切削进给	镗孔循环	ON	Y
						OFF	X
G83.2	Z/X	切削进给 间歇进给	就位检查 停止	快速进给	深孔镗孔循环 2	ON	Z/X
						OFF	

固定循环模式通过G80或01组的G指令被取消，同时，各数据也被清零。



指令格式

(1) 端面钻孔

G8△ X/U__ C/H__ Z/W__ R__ Q__ P__ F__ K__ D__ S__ ,S__ ,M__ ;	
G8△	: 孔加工模式 (G83、G84、G85)
X/U__ C/H__	: 孔位置数据
Z/W__ R__ Q__ P__ F__	: 孔加工数据
K__	: 重复次数
D__	: 刀具主轴编号
S__	: 主轴转速
,S__	: 返回时的主轴转速
M__	: 辅助指令

(2) 横向钻孔

G8□ Z/W__ C/H__ X/U__ R__ Q__ P__ F__ K__ D__ S__ ,S__ ,M__ ;	
G8□	: 孔加工模式 (G87、G88、G89)
Z/W__ C/H__	: 孔位置数据
X/U__ R__ Q__ P__ F__	: 孔加工数据
K__	: 重复次数
D__	: 刀具主轴编号
S__	: 主轴转速
,S__	: 返回时的主轴转速
M__	: 辅助指令

(3) 取消

G80 ;

(4) 数据概要与对应地址

- (a) 孔加工模式……是钻孔（G83、G87）、攻牙（G84、G88）、镗孔（G85、G89）的固定循环模式。
如果在模态中做了指定，则在发出其他孔加工指令、钻孔用固定循环的取消指令或01组的G指令之前，一直有效。
- (b) 孔位置数据……定位X（Z）、C轴时的数据。
在非模态中，当连续进行相同的孔加工模式时，对每1节发出指令。
- (c) 孔加工数据……是加工时的实际加工模式。
除Q以外为模态。关于G83、G87的Q，对于非模态中必不可少的节，对每一节发出指令。
- (d) 重复次数……为了进行等间隔的孔加工，在同一循环中重复操作时，指定重复次数。
设定范围为0～9999，小数点无效。
非模态下，仅对指定的节有效。
当没有指定重复次数时，视为K1。如果指定K0，则虽然会保存孔加工数据，但是不会进行孔加工。

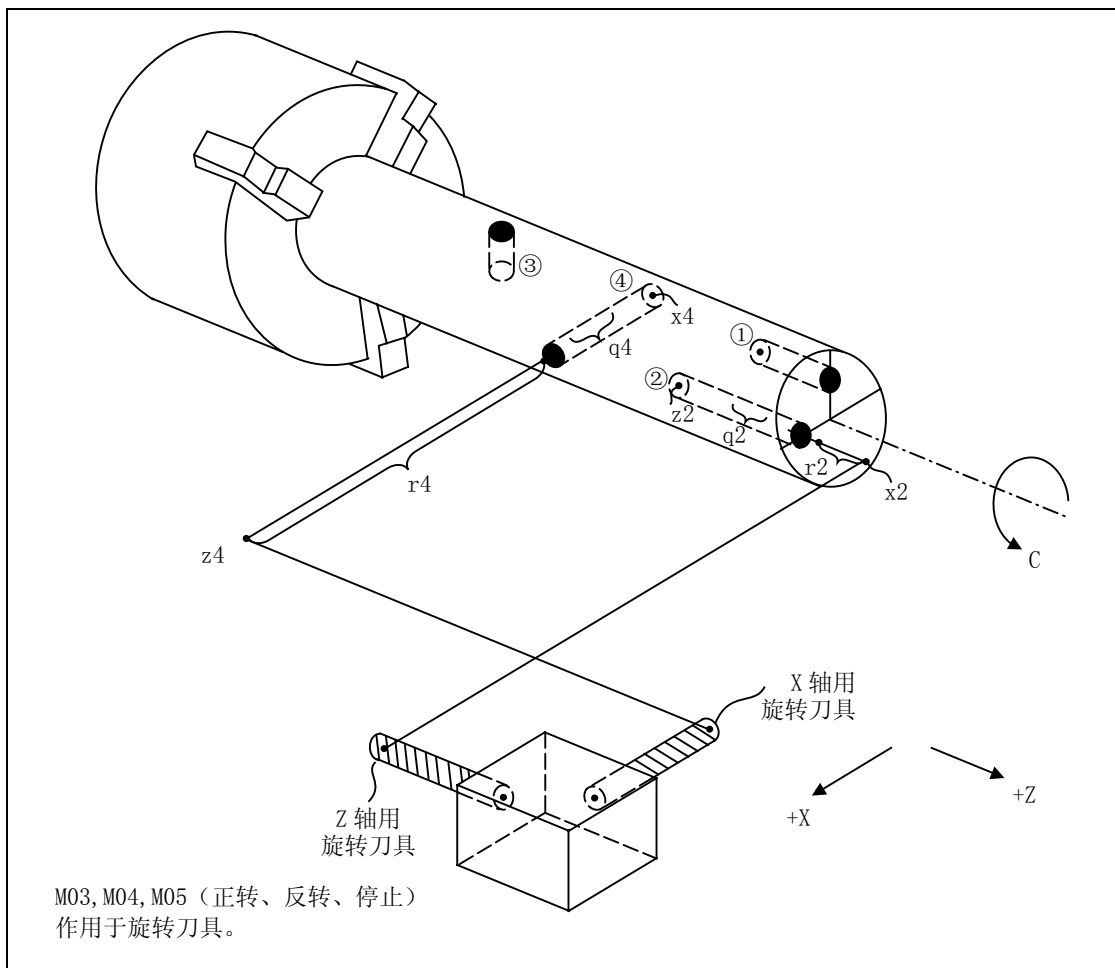
地址	地址的含义
G	孔加工循环顺序的选择（G90、G83、G84、G85、G87、G88、G89）
※ X/U（Z/W） C/H	孔位置起始点（绝对值/增量值）指定
※ A/W（X/U）	孔底位置（绝对值/距 R 点的增量值）指定
R	R 点位置（距起始点的增量值）的指定（符号跳跃）
Q	G83（G87）中，每次切入量的指定。总是以增量值指定半径值（符号跳跃）
P	孔底点的延时时间指定。时间与指定数值的关系与 G04 的指定相同
F	切削进给中的进给速度指定
K	重复次数的指定 0～9999（标准值 = 1）
D	攻牙循环中使用的主轴的指定（未使用）
S	主轴转速的指定
, S	返回时的主轴转速指定 （注）仅同期攻牙时有效。其他场合被跳跃。
, R	同期攻牙/非同期攻牙的指定 （注）主轴的旋转方向可根据同期攻牙指令极性反转信号而反转。
M	辅助指令的指定

※（ ）内为G87、G88、G89时



概略图

下图为钻孔用固定循环的钻孔轴、定位等的概略图。

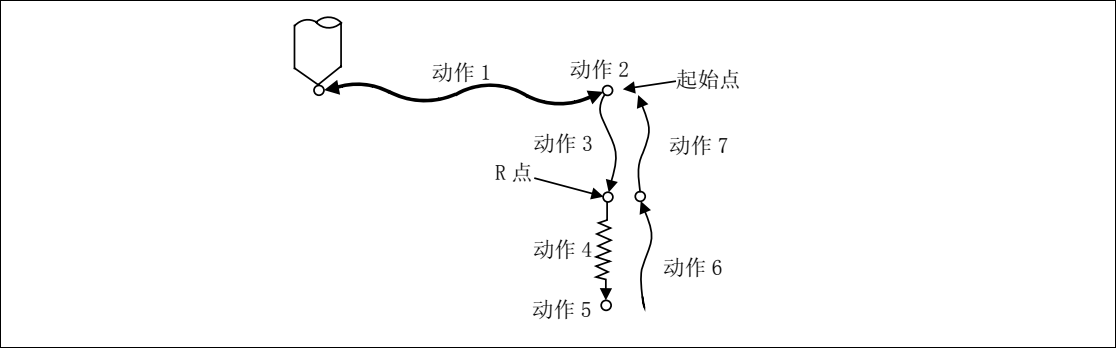


- ① G83 $Xx_1 Cc_1 Zz_1 Rr_1 Qq_1 Pp_1 Ff_1 Kk_1$;
- ② G83 $Xx_2 Cc_2 Zz_2 Rr_2 Qq_2 Pp_2 Ff_2 Kk_2$;
- ③ G87 $Zz_3 Cc_3 Xx_3 Rr_3 Qq_3 Pp_3 Ff_3 Kk_3$;
- ④ G87 $Zz_4 Cc_4 Xx_4 Rr_4 Qq_4 Pp_4 Ff_4 Kk_4$;



动作例

实际的动作分类为以下7种。



- 动作 1…X (Z)，定位（快速进给）到 C 轴的起始点。
 当有“定位轴就位宽度”的指定时，在节结束时，进行就位检查。
- 动作 2…如果发出了 C 轴钳位的 M 代码，则输出。
- 动作 3…定位（快速进给）到 R 点。
- 动作 4…通过切削进给进行孔加工。
 当有“钻孔轴就位范围”的指定时，在节结束时，进行就位检查。但是，对于深孔钻孔循环 1、2，在中途的孔加工中，不进行就位检查。在所指定的孔底位置（最后的孔加工）进行就位检查。
- 动作 5…在孔底位置进行动作，与固定循环模式不同，有旋转刀具反转（M04）、旋转刀具正转（M03）延时等。
- 动作 6…返回 R 点。
- 动作 7…通过快速进给返回起始点。
 （动作 6 与动作 7 可能会因固定循环模式而变为 1 个动作。）

（注）在同期攻牙指令中，根据参数#1223（同期攻牙改良），进行就位检查。（仅部分机种）

可通过以下的G指令，切换固定循环的完成是动作6还是动作7。

- G98……起始等级回归
- G99……R点等级回归

这些指令为模态，例如指定了一次G98之后，则下一次指定G99之前，一直是G98模式。运转准备完成时的初始状态下，为G98模式。



同期攻牙时的就位检查

其他钻孔（含非同期攻牙）循环的就位检查，请参阅“动作例”。

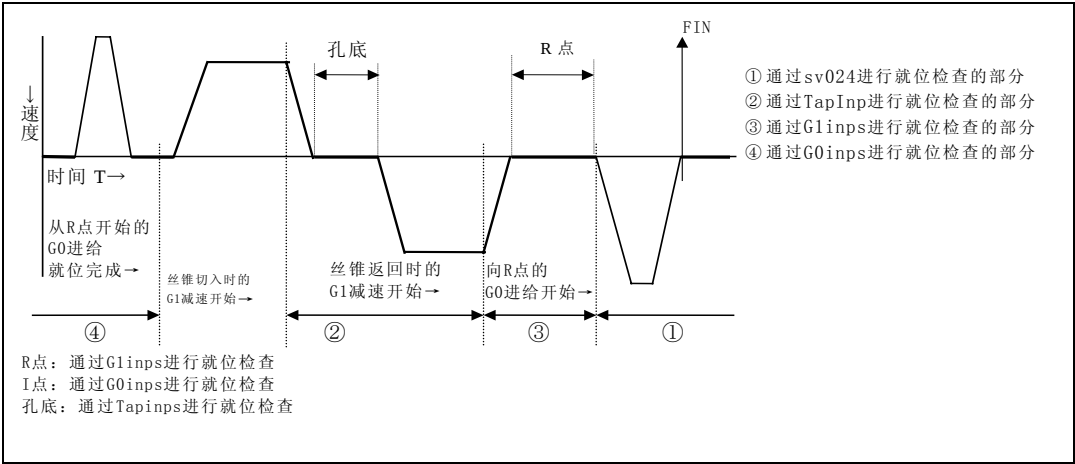
同期攻牙就位检查参数的设定值与攻牙轴的活动

(1) 同期攻牙就位检查的设定

#1223 aux07				G84/G74 指令的 P 指定	同期攻牙时的就位检查
bit3	bit4	bit5	bit2		
同期攻牙 就位检查	孔底	R 点	I 点→R 点		
0	—	—	—	—	在 I 点→R 点/R 点/孔底进行就位检查
1	—	—	—	无P 例: G84 F1. Z-5. S1000 R-5.	孔底: 不进行就位检查 R 点: 不进行就位检查 I 点→R 点: 不进行就位检查
1	1	1	1	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 根据攻牙用就位检查宽度进行就位检查 R 点: 进行就位检查 I 点→R 点: 进行就位检查
1	1	0	1	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 根据攻牙用就位检查宽度进行就位检查 R 点: 不进行就位检查 I 点→R 点: 进行就位检查
1	0	1	1	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 不进行就位检查 R 点: 进行就位检查 I 点→R 点: 进行就位检查
1	0	0	1	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 不进行就位检查 R 点: 不进行就位检查 I 点→R 点: 进行就位检查
1	1	1	0	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 根据攻牙用就位检查宽度进行就位检查 R 点: 进行就位检查 I 点→R 点: 不进行就位检查
1	1	0	0	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 根据攻牙用就位检查宽度进行就位检查 R 点: 不进行就位检查 I 点→R 点: 不进行就位检查
1	0	1	0	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 不进行就位检查 R 点: 进行就位检查 I 点→R 点: 不进行就位检查
1	0	0	0	有P 例: G84 F1. Z-5. S1000 PC R-5.	孔底: 不进行就位检查 R 点: 不进行就位检查 I 点→R 点: 不进行就位检查

(注 1) I 点指起始点。

(2) 同期攻牙检查的就位宽度与攻牙轴的移动



(3) 同期攻牙就位检查的参数设定值与攻牙轴移动的关系

#1223 aux07				孔底等待时间	在孔底的动作	在 R 点的动作	1 点→R 点中的动作
bit3	bit4	bit5	bit2				
同期攻牙就位检查	孔底	R 点	1 点→R 点				
0	—	—	—	通过 P 指定指定的时间 当没有 P 指定时，作为处理时间，需数 10ms	inpos(#1193)与 aux07(#1223): 根据 bit1 的设定	inpos(#1193)与 aux07(#1223): 根据 bit1 的设定	inpos(#1193)与 aux07(#1223): 根据 bit1 的设定
1	0	0	1	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	经过左栏的时间之前，等待		通过 G0inps 完成就位检查前，等待
1	0	1	1	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	经过左栏的时间之前，等待	通过 G1inps 完成就位检查前，等待	通过 G0inps 完成就位检查前，等待
1	1	0	1	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	就位检查完成后 经过左栏的延时时间之前，等待		通过 G0inps 完成就位检查前，等待
1	1	1	1	P 指定与 TapDw1 (#1313) 中较大者有效 如果两者均为 0，则作为处理时间，需数 10ms	就位检查完成后 经过左栏的延时时间之前，等待	通过 G1inps 完成就位检查前，等待	通过 G0inps 完成就位检查前，等待
1	0	0	0	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	经过左栏的时间之前，等待		
1	0	1	0	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	经过左栏的时间之前，等待	通过 G1inps 完成就位检查前，等待	
1	1	0	0	P 指定与 TapDw1 (#1313) 中较大者有效 如两者均为 0，则无延时	就位检查完成后 经过左栏的延时时间之前，等待		
1	1	1	0	P 指定与 TapDw1 (#1313) 中较大者有效 如果两者均为 0，则作为处理时间，需数 10ms	就位检查完成后 经过左栏的延时时间之前，等待	通过 G1inps 完成就位检查前，等待	

(注 1) I 点指起始点。

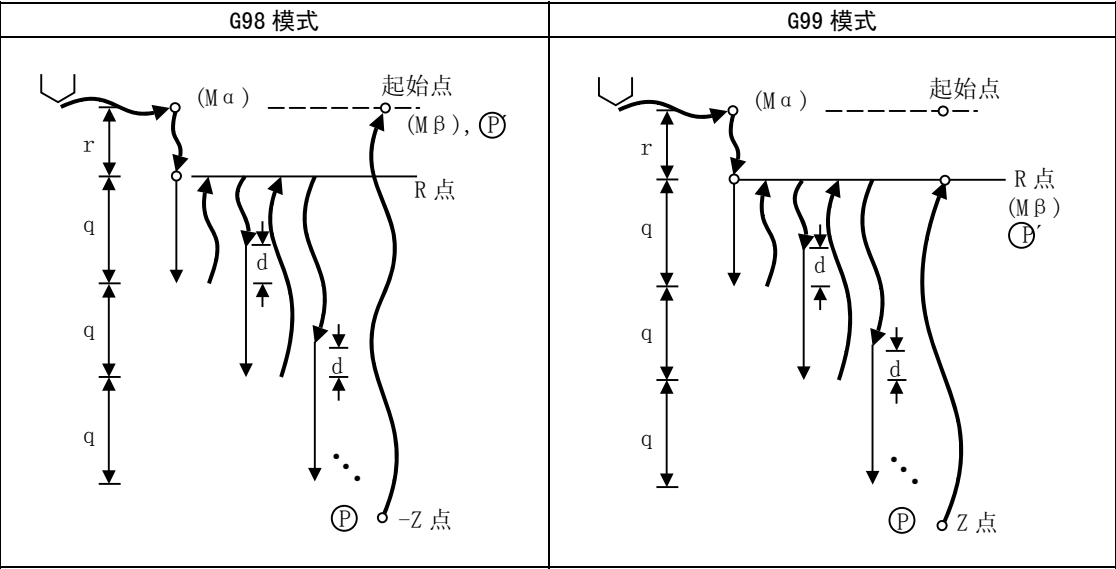
(注 2) 当 R 点上的就位检查无效时，可能会导致振动及精度降低。使用时请充分确认。

13.5.1 端面深孔钻孔循环1；G83（横向深孔钻孔循环1；G87）



有 Q 指令时（钻深孔）

G83 (G87) X(z)___ C___ Z(x)___ R_r_ Q_q_ P_p_ F_f_ K_k_ M_m_ ;

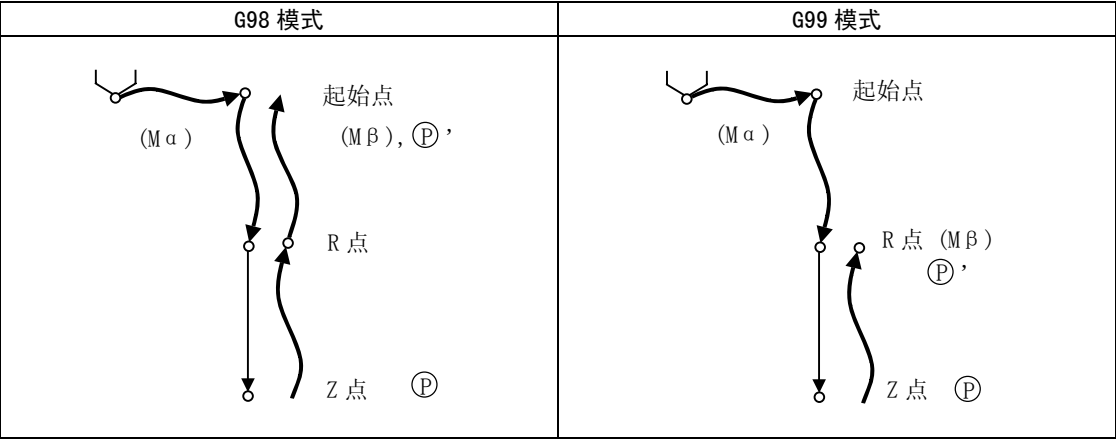


- (1) 通过参数（#8013 G83 返回）设定返回量。以快速进给返回。
- (2) (Mα)·· 当有 C 轴钳位 M 代码指令 (Mm) 时，输出该 M 代码 (Mm)。
- (3) (Mβ)·· 当有 C 轴钳位 M 代码指令 (Mm) 时，输出 C 轴解除钳位的 M 代码 (C 轴钳位的 M 代码+1 即 Mm+1)。
- (4) (P)···· 延时通过 P 指令的时间。
- (5) (P)'·· 输出 C 轴解除钳位 M 代码 (Mm+2) 之后，通过参数 (#1184 clmp_D) 进行所设定时间的延时。



没有 Q 指令时（钻孔）

G83 (G87) X(z)___ C___ Z(x)___ R_r_ P_p_ F_f_ K_k_ M_m_ ;



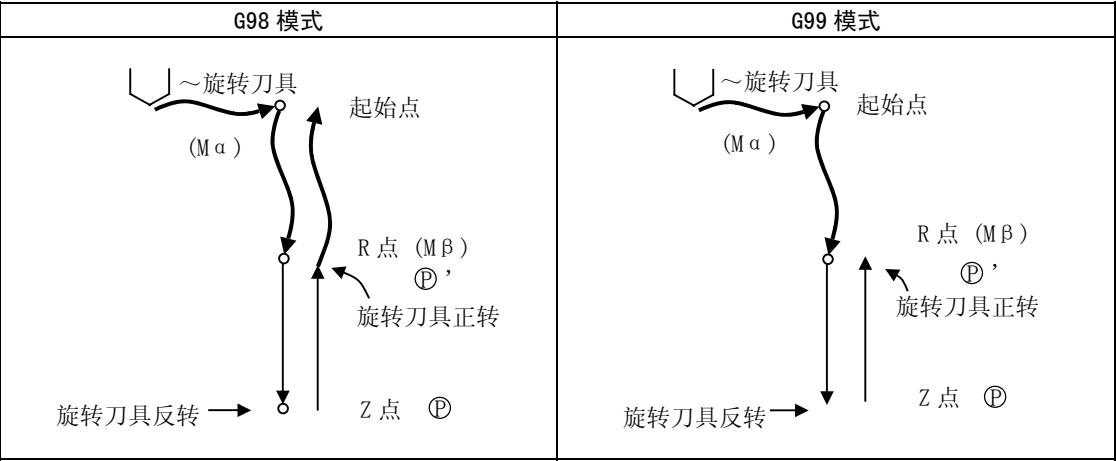
- (1) 关于，Mα、Mβ、(P)、(P)'，与“有Q指令时（钻深孔）”相同。

13.5.2 端面攻牙循环；G84（横向攻牙循环；G88）
/ 端面反向攻牙循环；G84.1（横向反向攻牙循环；G88.1）



详细说明

以下为G84（G88）的说明。反向攻牙循环G84.1（G88.1），是将丝锥的旋转方向反转。
G84（G88） Xx1 Cc1 Zz1 Rr1 Pp1 Ff1 Kk1 Dd1 Ss1 ,Ss2 ,Rr2 Mm1 ；



- (1) 关于，Mα、Mβ、⑰、⑰'，与13.5.1的“有Q指令时（钻深孔）”相同。
- (2) 在G84（G88）执行中，进入超程取消状态，超程自动变为100%。
- (3) 当控制参数“G00空运转”打开时，空运转变为对定位指令有效。另外，在G84（G88）执行中，按回馈等待按钮，则在回归动作结束之后，节停止。
- (4) 单节运转时，不在攻牙循环的折返位置停止。
- (5) G84（G88）模态中，输出“攻牙中”的NC输出信号。
- (6) G84（G88）同期攻牙模态中，不输出M3、M4、S代码等。
- (7) 在R点暂停，输出旋转刀具正转信号。
- (8) 旋转刀具在孔底反转，进行攻牙。
- (9) 输出旋转刀具反转（M04）、旋转刀具正转（M03）信号之前，如果需要旋转刀具停止（M05），则请编辑固定循环子程序。



同期・非同期攻牙选择

(1) 根据程序指令进行选择

攻牙循环的“R0/1”指令

G84 (G88) Xx1 Cc1 Zz1 Rr1 Pp1 Ff1 Kk1 Dd1 Ss1 , Ss2 , Rr2 Mm1 ;

r2=1时为同期攻牙模式，r2=0时为非同期攻牙模式。

(2) 根据参数进行选择

基本规格参数

#	项目		内容	设定范围
1229	set01	bit4	0: 攻牙循环为带浮动丝锥夹头的攻牙循环。 1: 攻牙循环为无浮动丝锥夹头的攻牙循环。	0/1

将本参数打开，则攻牙指令为同期攻牙循环。

(3) 根据M功能进行选择

基本规格参数

#	项目		内容	设定范围
1272 (PR)	ext08	bit1	M 功能同步 攻牙循环有效	0: 无效 1: 有效

本参数关闭时，不进行M功能下的同期攻牙选择。

基本规格参数

#	项目	内容	设定范围
1513	stapM	同步丝锥选择用 M 代码	0~ 99999999

通过本参数设定值的辅助功能代码，选择同期攻牙模式。

M功能可在攻牙指令之前，在同一节中进行指令。

(注) 请不要使用M00、01、02、30、96、97、98、99。

(例)

M29; (M指令同期攻牙)

G84 Z50. R20. F2.; 或 G84 Z50. R20. F2. M29 ;

通过上述3项进行的同步・非同期攻牙选择，其组合如下表所示。

	组合											
程序指令 (,R0/1)	0	0	0	0	1	1	1	1	无指令			
#1229 (bit4)	0	0	1	1	0	0	1	1	0	0	1	1
M 功能代码 (M**)	×	○	×	○	×	○	×	○	×	○	×	○
同期・非同期选择	非	非	非	非	同	同	同	同	非	同	同	同

×: 不指令 非: 非同期攻牙
○: 进行指令 同: 同期攻牙



同期攻牙选择的解除

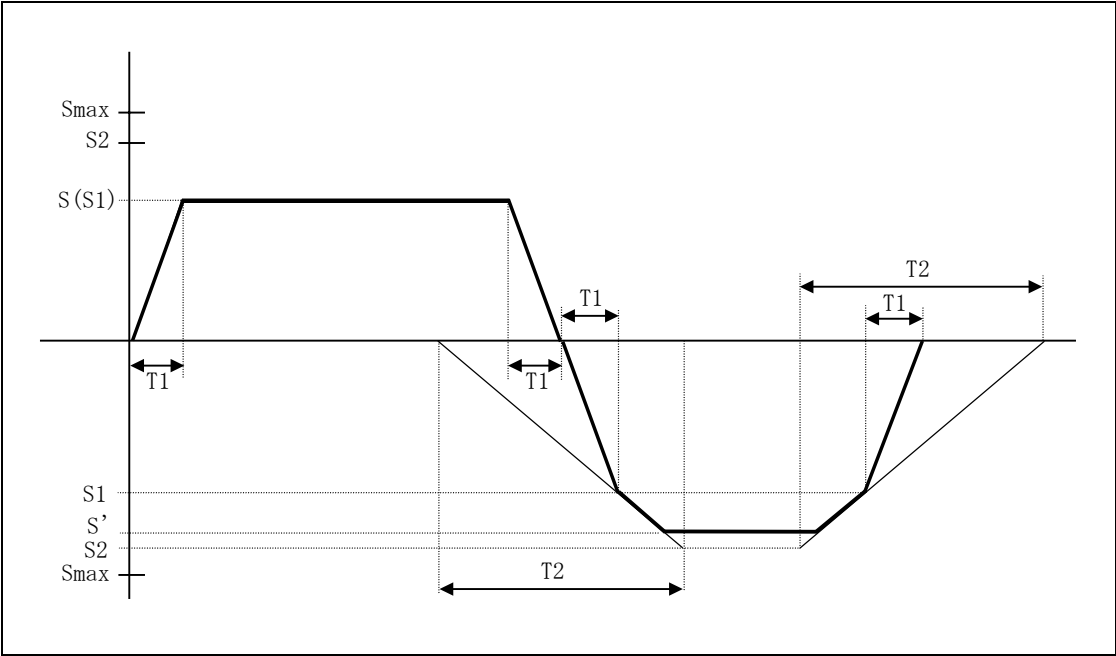
同期攻牙选择的解除，可通过指令重新启动、G80（钻孔固定循环取消）、01组的G代码、其他固定循环G代码，予以取消。



同期攻牙中，主轴的加减速曲线

本功能，是通过在同期攻牙中，将主轴及钻孔轴的加减速曲线进行最多3级的多级化，让主轴的加减速曲线接近速度回环时的加减速曲线。加减速曲线可针对各齿轮，分别最多设定3级。
另外，从孔底返回时，根据返回时的主轴转速，可能会快速返回。返回时的主轴转速，作为模式信息被予以保持。

(1) 攻牙转速<返回时的主轴转速≤同期攻牙切换主轴转速2 时



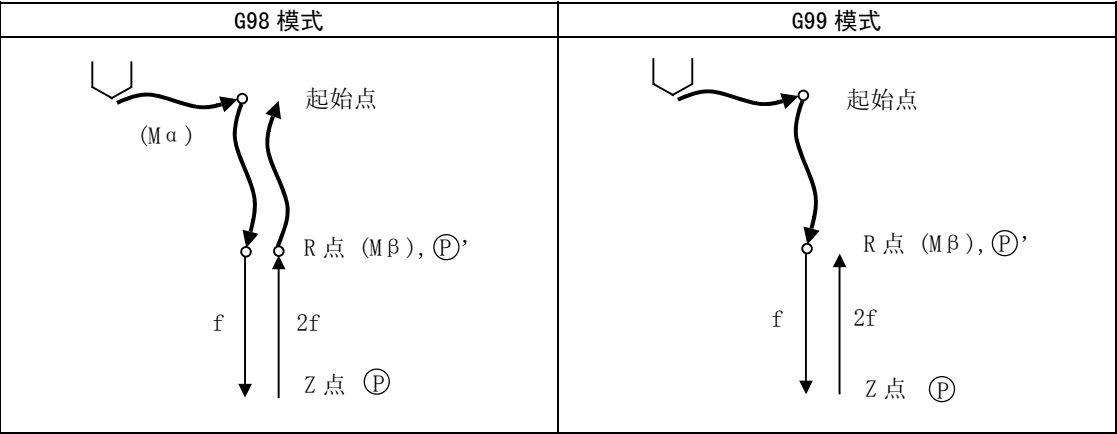
S	: 指令主轴转速
S'	: 返回时的主轴转速
S1	: 攻牙转速（主轴基本规格参数#3013～#3016）
S2	: 同期攻牙切换主轴转速2（主轴基本规格参数#3037～#3040）
Smax	: 最高转速（主轴基本规格参数#3005～#3008）
T1	: 攻牙时间常数（主轴基本规格参数#3017～#3020）
T2	: 同期攻牙切换时间常数2（主轴基本规格参数#3041～#3044）

13.5.3 端面镗孔循环；G85（横向镗孔循环；G89）



详细说明

G85 (G89) X (z)_ C_ Z (x)_ R_ P_ F_ K_ M_ ;



- (1) 关于， $M\alpha$ 、 $M\beta$ 、 $\textcircled{P}$ 、 $\textcircled{P}'$ ，与13.5.1的“有Q指令时（钻深孔）”相同。
- (2) R点前的返回，以2倍于指令的进给速度指令的切削进给速度进行。但是，不超过最大切削进给速度。

13.5.4 深孔钻孔循环2；G83.2



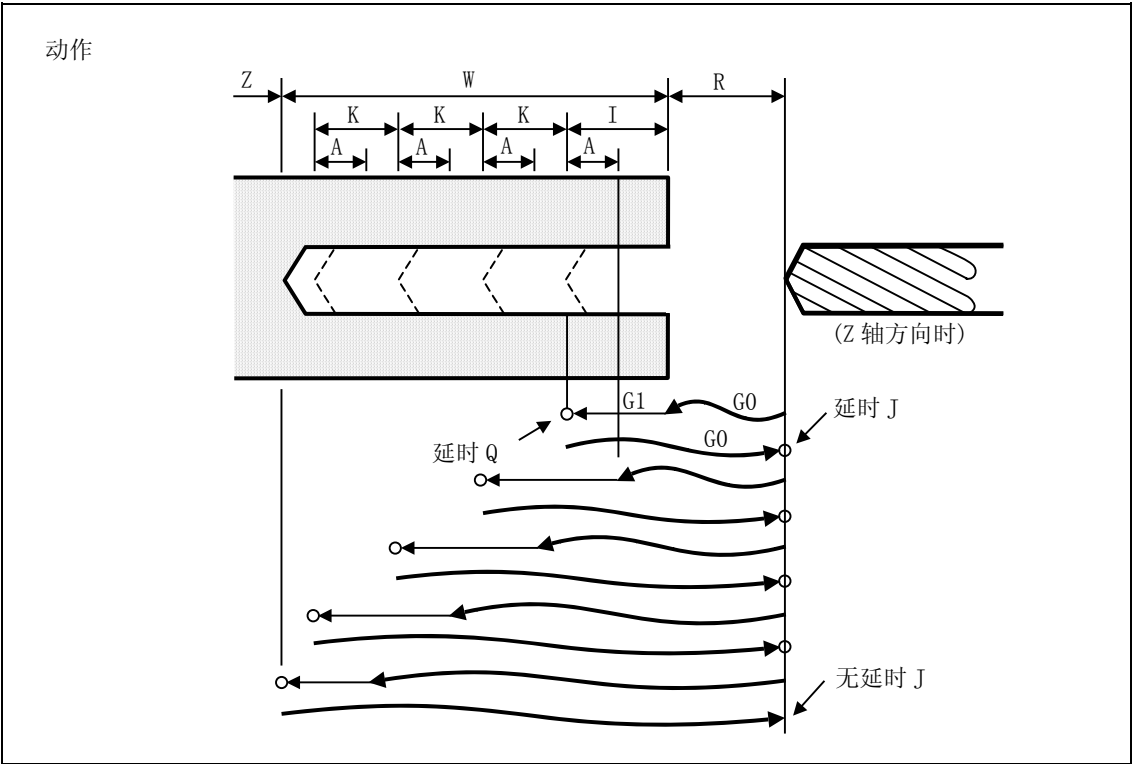
功能及目的

深孔钻孔循环2，是通过指令终点的X坐标或Z坐标与切削进给中的切入量，在X轴方向或Z轴方向上钻深孔。



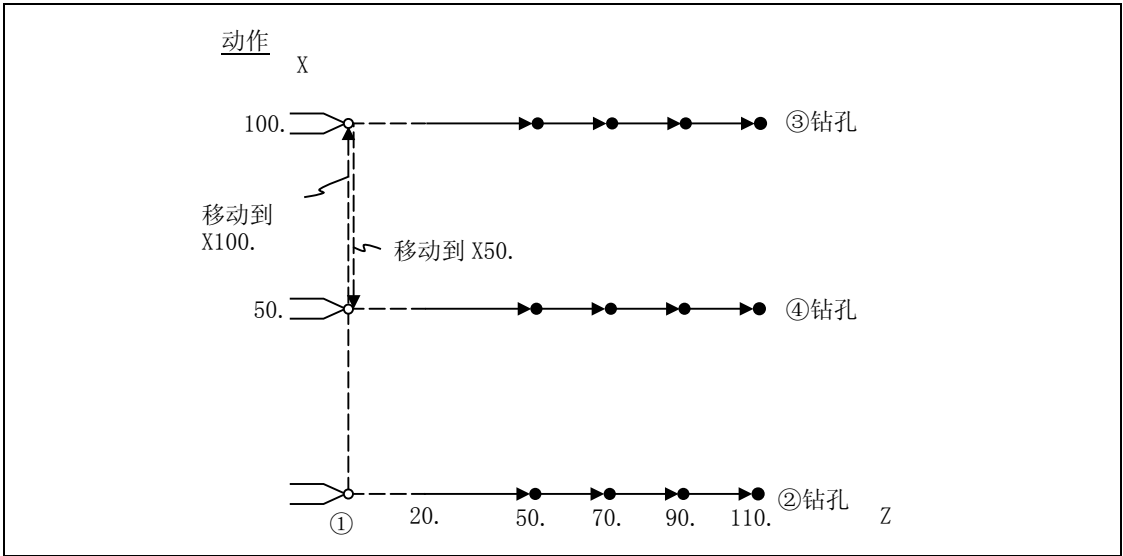
指令格式

G83.2 W/Z/U/X_ R_ I_ K_ A_ Q_ J_ F_ ;	
W/Z/U/X	: 距离钻孔切削开始点的增量值/孔底的坐标值（符号有效）
R	: 从当前位置到钻孔切削开始点的增量值（符号跳跃） （总是以增量值指定半径值）
I	: 1 次的切入量（符号跳跃）（总是以增量值指定半径值）
K	: 第 2 次以后的切入量（符号跳跃）（总是以增量值指定半径值）
A	: 第 2 次以后的钻头停止安全距离（符号跳跃）（总是以增量值指定半径值）
Q	: 在切入点上的延时时间（符号跳跃、小数点无效）
J	: 在返回点上的延时时间（符号跳跃、小数点无效）
F	: 切削进给速度



程序例（将深孔钻孔循环 2 作为模态指令使用时）

G28 XZ;	
G0 X0. Z0. ;	...①
G83.2 Z110. R20. I30. K20. A5. Q1000 J500 F300. ;	...②
X100. ;	...③
X50. ;	...④
M02;	





详细说明

- (1) 当没有钻头停止安全距离（地址“A”）的指令时，使用参数#8013“G83”返回的参数设定值。
- (2) 由于深孔钻孔循环2为模态指令，所以在有同一模态组的指令或取消指令（G80）之前，始终有效。
- (3) 当缺少第1次切入量（地址“I”）与第2次之后的切入量（地址“K”）中的某一个指令（含指令值=0）时，使用设置了指令值的项目值，按照I=K=指令值执行。
另外，当两者均未指令时，进行一次到孔底为止的钻孔。
- (4) 当在1节内指令了多次钻孔轴的轴地址时，则后指令的地址有效。
- (5) 除G80外，深孔钻孔循环2也可通过01组的G代码予以取消。如果在同一节内同时指令了固定循环，则固定循环被跳跃。

m=01组的代码 n=钻孔用固定循环的代码

(a) $\underbrace{G_m \ G_n X}_{\text{执行跳跃}} (z) \underbrace{C \ Z}_{\text{执行}} (x) \underbrace{R \ I \ K \ A \ Q \ J \ F}_{\text{跳跃 记忆}} ;$

(b) $\underbrace{G_n \ G_m X}_{\text{跳跃}} (z) \underbrace{C \ Z}_{\text{执行}} (x) \underbrace{R \ I \ K \ A \ Q \ J \ F}_{\text{跳跃 记忆}} ;$

(例)

G01 G83.2 Z50. R-10. I8. K10. A3. Q1000 J500 F100. ;
G83.2 G01 Z50. R-10. I8. K10. A3. Q1000 J500 F100. ;
两者均执行G01 Z50. F100. 。

- (6) 如果与深孔钻孔循环2在同一节内指令了辅助功能，则在最初定位的同时输出。
- (7) 在深孔钻孔循环2模式中，指令了刀具长度偏移指令（T功能），则按照刀具长度偏移功能执行。
- (8) 在深孔钻孔循环2模式中，如果在该节中有基本轴、附加轴或R的数据，则进行钻孔动作，如果没有数据，则不进行钻孔动作。
但是，即使是X轴数据，如果该节为延时（G04）时间指令，则不进行钻孔动作。
- (9) 请在进行钻孔动作的节（含基本轴、附加轴或R数据的节）中指令孔加工数据（A、I、K、Q、J）。
在不进行钻孔动作的节中，即使指令了这些数据，模态数据也不会被更新。
- (10) 以下指令导致程序错误（P204）。
 - (a) 指令了钻孔轴X轴（指令地址X或U）与钻孔轴Z（指令地址Z或W）时。
 - (b) 指令了X轴或Z轴以外的轴（指令地址X、U、Z、W以外）时。

- (11) 在执行深孔钻孔循环2时按了回馈等待按钮，则立即自动运转休止，再次起动自动运转，则执行剩余指令。
- (12) 在自动运转休止中，通过手动运转进行了插入动作时（手动ABS开关打开），自动运转重新启动后，深孔钻孔循环2模态中的动作被移动了插入移动量。
- (13) 在单节运转中，深孔钻孔循环2指令完成后，节停止。

13.5.5 钻孔用固定循环取消：80



详细说明

取消钻孔用固定循环（G83、G84、G85、G87、G88、G89）。孔加工模式、孔加工数据均被取消。

13.5.6 钻孔用固定循环使用注意事项



注意事项

- (1) 指令G84、G85的固定循环之前，必须预先通过辅助功能（M3、M4）将旋转刀具旋转到规定的方向上。
- (2) 在固定循环模式中，如果在该节中有基本轴、附加轴或R的数据，则进行钻孔动作，如果没有数据，则不进行钻孔动作。
但是，即使是X轴数据，如果该节为延时（G04）时间指令，则不进行钻孔动作。
- (3) 请在进行钻孔动作的节（含基本轴、附加轴或R数据的节）中指令孔加工数据（Q、P）。
在不进行钻孔动作的节中，即使指令了这些数据，模态数据也不会被更新。
- (4) 在执行G85（G89）时，如果进行重新启动，则F的模式可能会发生改变。

(5) 除G80外，钻孔固定循环也可通过01组的G代码予以取消。如果在同一节内同时指令了固定循环，则固定循环被跳跃。

m=01组的代码 n=钻孔用固定循环的代码

(a) $\underbrace{G_m \ G_n X}_{\text{执行跳跃}} (z) \underbrace{C \ Z}_{\text{执行}} (x) \underbrace{R \ Q \ P \ K \ F}_{\text{跳跃记忆}} ;$
 (b) $\underbrace{G_n \ G_m X}_{\text{跳跃}} (z) \underbrace{C \ Z}_{\text{执行}} (x) \underbrace{R \ Q \ P \ K \ F}_{\text{跳跃记忆}} ;$

(例)

G01 G83 X100. C30. Z50. R-10. Q10. P1 F100. ;

G83 G01 X100. C30. Z50. R-10. Q10. P1 F100. ;

两者均执行G01 X100. C30. Z50. F100.。

(6) 如果与固定循环2在同一节内指令了辅助功能，则在最初定位的同时输出。

但是，如果在同一节内指令了通过参数（#1183 clmp_M）设定的C轴钳位M代码，则在定位动作后输出M代码。（动作2）

钻孔动作后，返回回归点（G98模式时为起始点/G99模式时为R点）后，输出C轴解除钳位的M代码（钳位M+1），在参数（#1184 clmp_D）所设定的时间内进行延时。

当有次数指定时，除C轴钳位的M代码外，仅第一次时进行上述控制。关于C轴钳位・解除钳位的M指令，在模态因运转、固定循环取消指令而被取消之前，每1次都进行输出。

(7) 在钻孔固定循环模式中，指令了刀具长度偏移指令（T功能），则按照刀具长度偏移功能执行。

(8) 在刀尖R补偿中指令了钻孔用固定循环，则发生程序错误（P155）。

(9) G代码系列1时，起始点等级回归固定。无法根据G98/G99变更回归等级。如果指令G98/G99，则执行其他功能，所以请加以注意。

(10) 如下例所示，在某根轴的移动方向反转的节中，由于伺服系统的负载会变的非常大，所以请不要在加工程序中指令就位宽度。

G0 X100. , I10.0 ;

X-200. ;

13. 程序辅助功能

13.6 钻孔固定循环（MELDAS 特殊格式）

13.6 钻孔用固定循环：G80～G89（MELDAS特殊格式）

[E68]



功能及目的

本功能是通过1节的指令，按照预先决定的作业顺序，执行定位与钻孔、镗孔、攻牙等加工程序的功能，在加工顺序中，包括如下所示的内容。

当附加MELDAS特殊格式时（#1265 ext01/bit2打开），本功能生效。

G 代码	钻孔作业开始 (-Z 方向)	在孔底的动作		返回动作 (+Z 方向)	用 途
		停止	主轴		
G80	-	-	-	-	取消
G81	切削进给	-	-	快速进给	钻孔、 车铣万用钻循环
G82	切削进给	有	-	快速进给	钻孔、 计数器镗孔循环
G83	间歇进给	有	-	快速进给	深孔钻孔循环
G83.1	间歇进给	有	-	快速进给	步进循环
G84	切削进给	有	反转	切削进给	攻牙循环
G84.2	切削进给	有	反转	切削进给	同期攻牙循环
G85	切削进给	-	-	切削进给	镗孔循环
G89	切削进给	有	-	切削进给	镗孔循环

当接收到G80或01组的G指令（G00、G01、G02、G03）时，固定循环模式被取消。同时，各数据也被清零。



指令格式

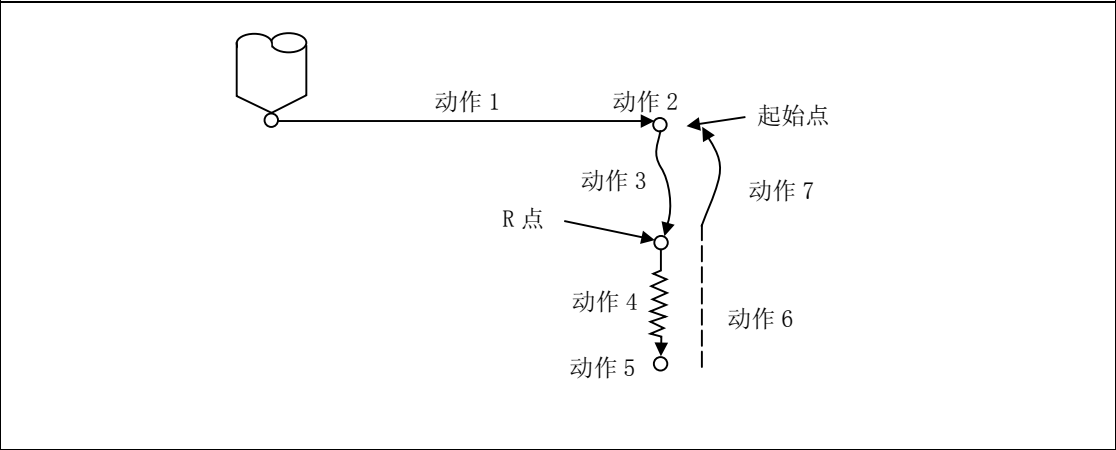
G8△ X__ Y__ Z__ R__ Q__ P__ F__ L__ S__ , S__ , I__ , J__					
G8△	:	孔加工模式			
X__ Y__ Z__	:	孔位置数据			
R__ Q__ P__ F__	:	孔加工数据			
L__	:	重复次数			
S__	:	主轴转速			
, S__	:	返回时的主轴转速			
, I__	:	定位轴的就位宽度			
, J__	:	钻孔轴的就位宽度			

如上所述，被分类为孔加工模式、孔位置数据、孔加工数据、重复次数、主轴转速、返回时的主轴转速、定位轴就位宽度、钻孔轴就位宽度。



详细说明

- (1) 孔加工模式是指钻孔、计数器镗孔、攻牙、镗孔等固定循环模式，孔位置数据是指定位X、Y轴时的数据，孔加工数据是指加工时的实际加工数据。孔位置数据及重复次数为非模态，而孔加工数据及返回时的主轴转速为模态。
- (2) 当与固定循环在同一节内，或是为固定循环模式时，如果指定M00、M01，则跳跃固定循环，定位后输出M00、M01。当指令了X、Y、Z、R中的任何一个时，执行固定循环。
- (3) 实际的动作可分类为以下7个。
 - 动作1 显示X、Y轴的定位，通过G00定位。
 - 动作2 定位完成后（起始点）动作中，指令了G87指令时，从控制装置向机械端输出M19指令。执行该M指令，控制装置接收到完成信号（FIN）后，开始下一动作。另外，当单节停止开关打开时，在定位完成后节停止。



- 动作3 以快速进给定位到R点。
- 动作4 通过切削进给进行孔加工。
- 动作5 是在孔位置进行的动作，因固定循环的模式而异，有主轴停止（M05）、主轴反转（M04）、主轴正转（M03）、延时、刀具移位等。
- 动作6 退刀到R点，根据固定循环的模式不同，可能会采用切削进给或快速进给。
- 动作7 以快速进给返回到起始点。

但是，可通过以下的G指令，切换固定循环的完成是动作6还是动作7。

G98起始基准返回

G99R点等级重新启动

这些指令为模态，例如指定了一次G98之后，则下一次指定G99之前，一直是G98模式。NC运转准备完成时的初始状态下，为G98模式。

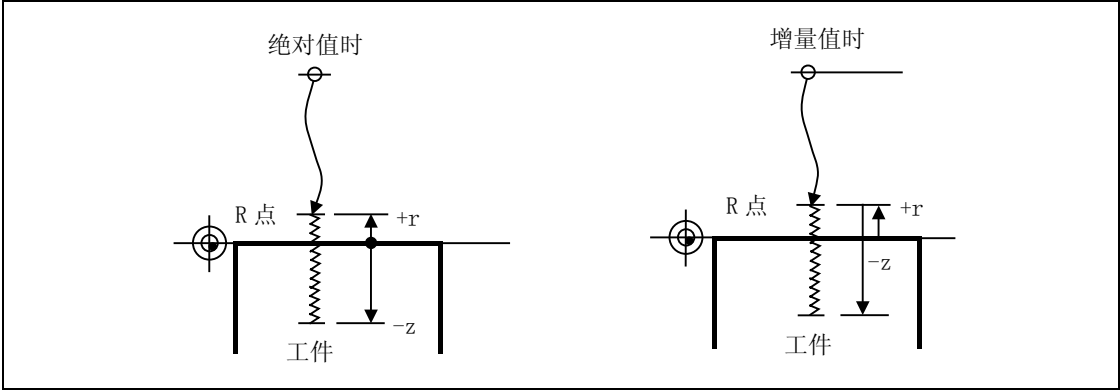
另外，当没有X、Y、Z、R指令中的任何一个时，跳跃孔加工数据。

本功能主要与特殊固定循环组合使用。

（4）固定循环的地址与含义

地址	地址的含义
G	固定循环顺序的选择（G80～G89）
X	钻孔点位置（绝对值或增量值）的指定
Y	钻孔点位置（绝对值或增量值）的指定
Z	孔底位置（绝对值或增量值）的指定
P	孔底位置延时时间的指定（跳跃小数点以下）
Q	在 G83 中指定每次的切入量，或在 G87 中指定移位量（增量值）
R	R 点位置（绝对值）的指定
F	在切削进给中指定进给速度
L	固定循环的重复次数指定 0～9999
S	主轴转速的指定
,S	返回时的主轴转速指定 (注) 仅同期攻牙时有效。其他场合被跳跃。
,I	就位宽度（位置误差量）
,J	固定循环中的钻孔轴用就位宽度（位置误差量）

（5）绝对值指令与增量值指令的区别



（6）攻牙循环/攻牙返回的进给速度

在攻牙循环、攻牙返回中，进给速度如下。

（a）非同步式攻牙循环的进给速度选择

G94/G95	控制参数 F1 位有效	F 指令值	速度指定
G94	关闭	F 指定	每分钟进给
	打开	F0～F8 以外	
		F0～F8 (无小数点)	F1 位进给
G95	-	F 指定	每转进给

- 为与设定无关

(b) 同期攻牙循环返回时的主轴转速

地址	地址的含义	指令范围（单位）	备注
,S	返回时的主轴转速	0~99999 (r/min)	作为模态信息被保持。 当设置为比主轴转速更小的值时，即使在返回时，主轴转速值也生效。 当返回时的主轴转速不为 0 时，攻牙返回超程值无效。



定位平面与钻孔轴

固定循环是定位平面与钻孔轴的基本控制要素，根据G17、G18、G19的平面选择指令决定定位平面，钻孔轴是垂直于上述平面的轴（X、Y、Z或其平行轴）。

平面选择	定位平面	钻孔轴
G17 (X-Y)	Xp-Yp	Zp
G18 (Z-X)	Zp-Xp	Yp
G19 (Y-Z)	Yp-Zp	Xp

Xp、Yp、Zp表示各基本轴X、Y、Z或基本轴的平行轴。

定位可指定钻孔轴以外的任意轴。

钻孔轴取决于与G81~G89指令在同一节内指令的钻孔轴轴地址。此时，如果没有指定轴地址，则基本轴成为钻孔轴。

（例 1）G17（X-Y 平面）被选中，以 Z 轴的平行轴为 W 轴时

```
G81      Z_ ； 以Z轴为钻孔轴。  
G81      W_ ； 以W轴为      "  
G81      ； （无Z、W）以Z轴为      "
```

- （注 1）通过参数#1080 Dril_Z，可将钻孔轴固定为 Z 轴。
- （注 2）请在固定循环取消状态下进行钻孔轴的切换。
- （注 3）当钻孔轴不存在时，发生程序错误。

在以下的说明中，是假定定位平面为XY平面、钻孔轴为Z轴，对固定循环的各模式加以说明。
但是，指令值全部是增量值，定位平面为XY平面，钻孔轴为Z轴。



固定循环中的就位宽度可编程指令

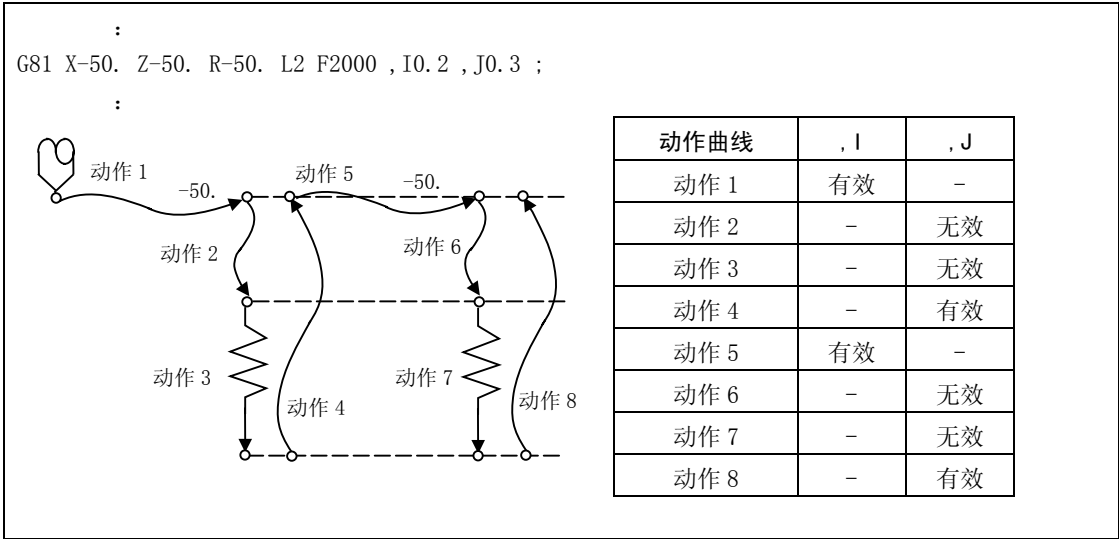
本指令是通过加工程序，指定固定循环指令时的就位范围。
指令的就位宽度，仅对G81（钻孔、车铣万用钻）、G82（钻孔、计数器镗孔）、G83（深孔钻孔循环）、G84（攻牙循环）、G85（镗孔）、G89（镗孔）、G83.1（步进循环）、G84.2（同期攻牙循环）8种固定循环有效，对于定位轴，通过“，I”地址，对于钻孔轴，通过“，J”地址进行指令。

地址	地址的含义	指令范围（单位）	备注
，I	就位宽度 （位置误差量）	1~999.999（mm） 以 1 μm 为单位	如果指令了超出指令范围的数值，则发生程序错误。 程序错误（P35）
，J	固定循环中的钻孔轴用就位宽度（位置误差量）		



固定循环程序的就位检查

固定循环中，当重复次数L的指定超过2次以上时，指令的就位宽度在重复节（动作5~动作8）中也直接生效。



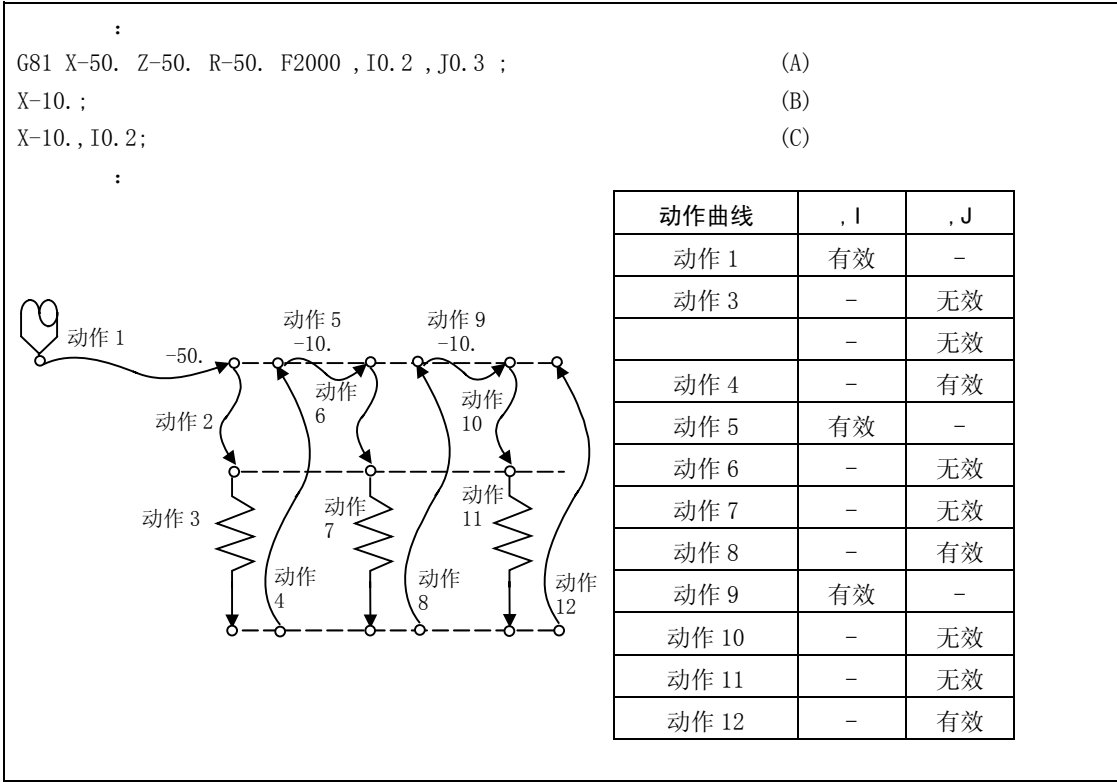
指定重复次数 L 时的动作

下述加工程序中，指令的就位宽度在图2的节中有效。
在（B）节中，上一节（A）中与定位有关的指令就位宽度（，I）无效（动作5）。但是，关于从孔底的返回，上一节（A）中指令的就位宽度（，J）有效（动作8）。

下述加工程序中，指令的就位宽度在下图的节中有效。

在（B）节中，上一节（A）中与定位有关的指令就位宽度（,I）无效（动作5）。但是，关于从孔底的返回，上一节（A）中指令的就位宽度（,J）有效（动作8）。

关于定位，当就位宽度有效时，请如（C）节所示再次进行指令（动作9）。



固定循环模式中的动作



同步攻丝就位检查参数的设定值与攻丝轴的移动

(1) 同期攻牙就位检查的设定

#1223 aux07				G84/G74 指令的 “P” 指定	同期攻牙时的 就位检查
bit3	bit4	bit5	bit2		
同期攻牙就 位检查	孔底	R 点	1 点→R 点		
0	-	-	-	-	在 I 点→R 点/R 点/孔底进行就位检查
1	-	-	-	无 P 例: G84 F1. Z-5. S1000 R-5.	孔底: 不进行就位检查 R 点: 不进行就位检查 1 点→R 点: 不进行就位检查
1	1	1	1	有 P 例: G84 F1. Z-5. S1000 P0 R-5.	孔底: 根据攻牙用就位检查宽度进行就位 检查 R 点: 进行就位检查 1 点→R 点: 进行就位检查

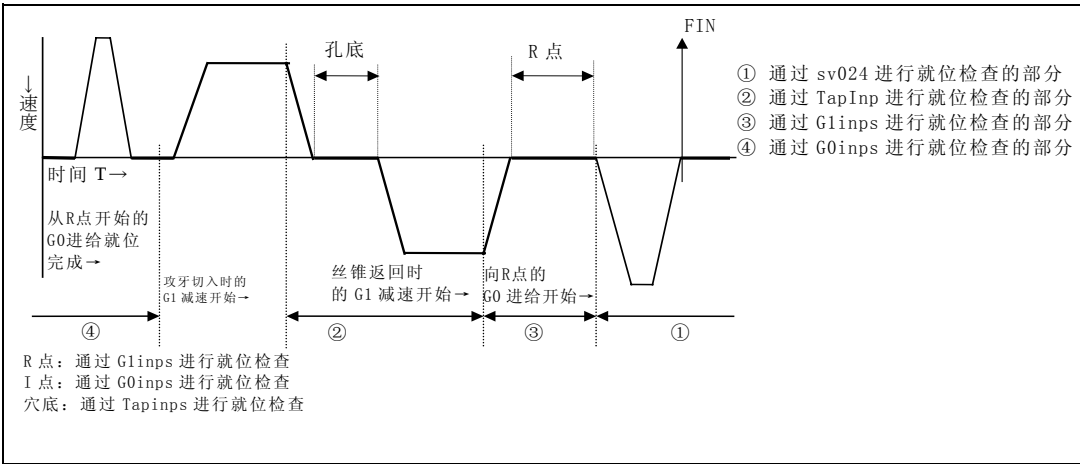
13. 程序辅助功能

13.6 钻孔固定循环（MELDAS 特殊格式）

#1223 aux07				G84/G74 指令的 “P” 指定	同期攻牙时的 就位检查
bit3	bit4	bit5	bit2		
同期攻牙就 位检查	孔底	R 点	1 点→R 点		
1	1	0	1	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 根据攻牙用就位检查宽度进行就位 检查 R 点： 不进行就位检查 1 点→R 点： 进行就位检查
1	0	1	1	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 不进行就位检查 R 点： 进行就位检查 1 点→R 点： 进行就位检查
1	0	0	1	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 不进行就位检查 R 点： 不进行就位检查 1 点→R 点： 进行就位检查
1	1	1	0	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 根据攻牙用就位检查宽度进行就位 检查 R 点： 进行就位检查 1 点→R 点： 不进行就位检查
1	1	0	0	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 根据攻牙用就位检查宽度进行就位 检查 R 点： 不进行就位检查 1 点→R 点： 不进行就位检查
1	0	1	0	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 不进行就位检查 R 点： 进行就位检查 1 点→R 点： 不进行就位检查
1	0	0	0	有 P 例：G84 F1. Z-5. S1000 P0 R-5.	孔底： 不进行就位检查 R 点： 不进行就位检查 1 点→R 点： 不进行就位检查

（注 1）I 点指起始点。

（2）同期攻牙检查的就位宽度与攻牙轴的移动



(3) 同期攻牙就位检查的参数设定值与攻牙轴移动的关系

#1223 aux07				孔底等待时间	在孔底的动作	在R点的动作	1点→R点之间的动作
bit3	bit4	bit5	bit2				
同期攻牙就位检查	孔底	R点	1点→R点				
0	-	-	-	通过P指定指定的时间 当没有P指定时， 作为处理时间， 需数10ms	根据inpos(#1193)与aux07(#1223): bit1的设定	根据inpos(#1193)与aux07(#1223): bit1的设定	根据inpos(#1193)与aux07(#1223): bit1的设定
1	0	0	1	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	经过左栏的时间之前，等待		通过G0inps完成就位检查前，等待
1	0	1	1	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	经过左栏的时间之前，等待	通过G1inps完成就位检查前，等待	通过G0inps完成就位检查前，等待
1	1	0	1	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	就位检查完成后经过左栏的延时时间之前，等待		通过G0inps完成就位检查前，等待
1	1	1	1	P指定与TapDw1(#1313)中较大者有效 如果两者均为0，则作为处理时间，需数10ms	就位检查完成后经过左栏的延时时间之前，等待	通过G1inps完成就位检查前，等待	通过G0inps完成就位检查前，等待
1	0	0	0	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	经过左栏的时间之前，等待		
1	0	1	0	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	经过左栏的时间之前，等待	通过G1inps完成就位检查前，等待	
1	1	0	0	P指定与TapDw1(#1313)中较大者有效 如两者均为0，则无延时	就位检查完成后经过左栏的延时时间之前，等待		
1	1	1	0	P指定与TapDw1(#1313)中较大者有效 如果两者均为0，则作为处理时间，需数10ms	就位检查完成后经过左栏的延时时间之前，等待	通过G1inps完成就位检查前，等待	

(注1) I点指起始点。

(注2) 当R点上的就位检查无效时，可能会导致振动及精度降低。使用时请充分确认。

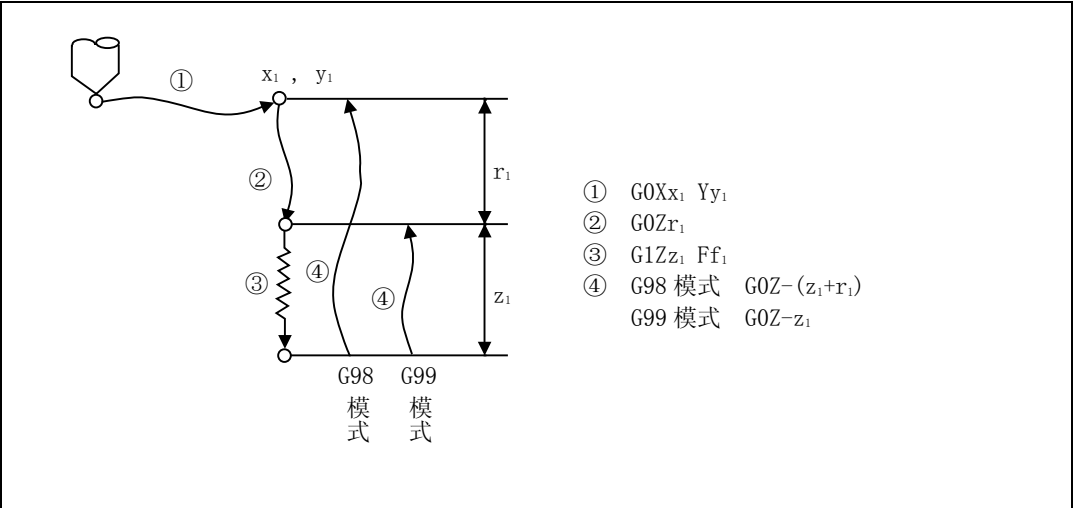


执行各固定循环时的动作

(a) G81（钻孔、车铣万能钻）

程序

G81 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ , Ii₁ , Jj₁ ;



单节运转时的停止位置，为①②④指令完成时。

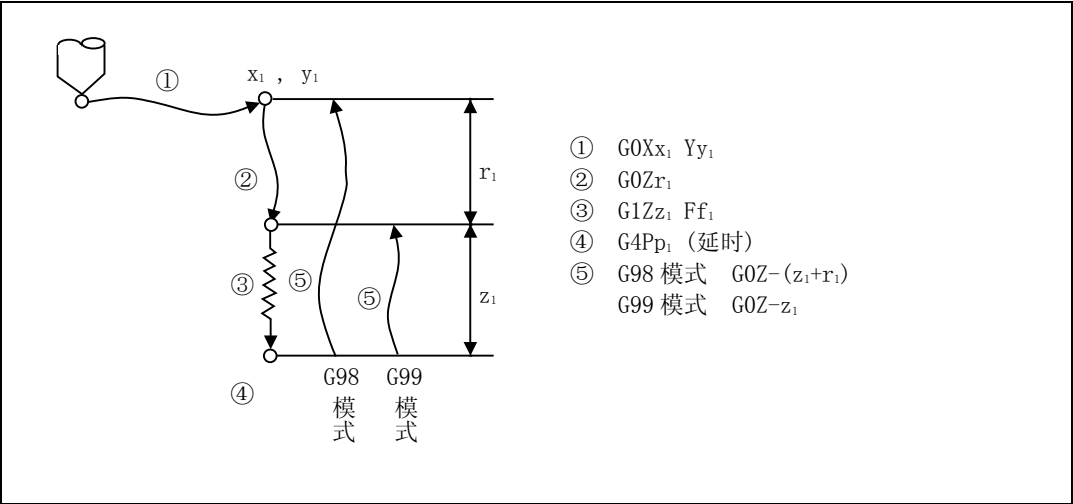
动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	有效

(b) G82（钻孔、计数器镗孔）

程序

G82 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ Pp₁ , Ii₁ , Jj₁ ;

P: 延时指定



动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	-
⑤	-	有效

单节运转时的停止位置，为①②⑤指令完成时。

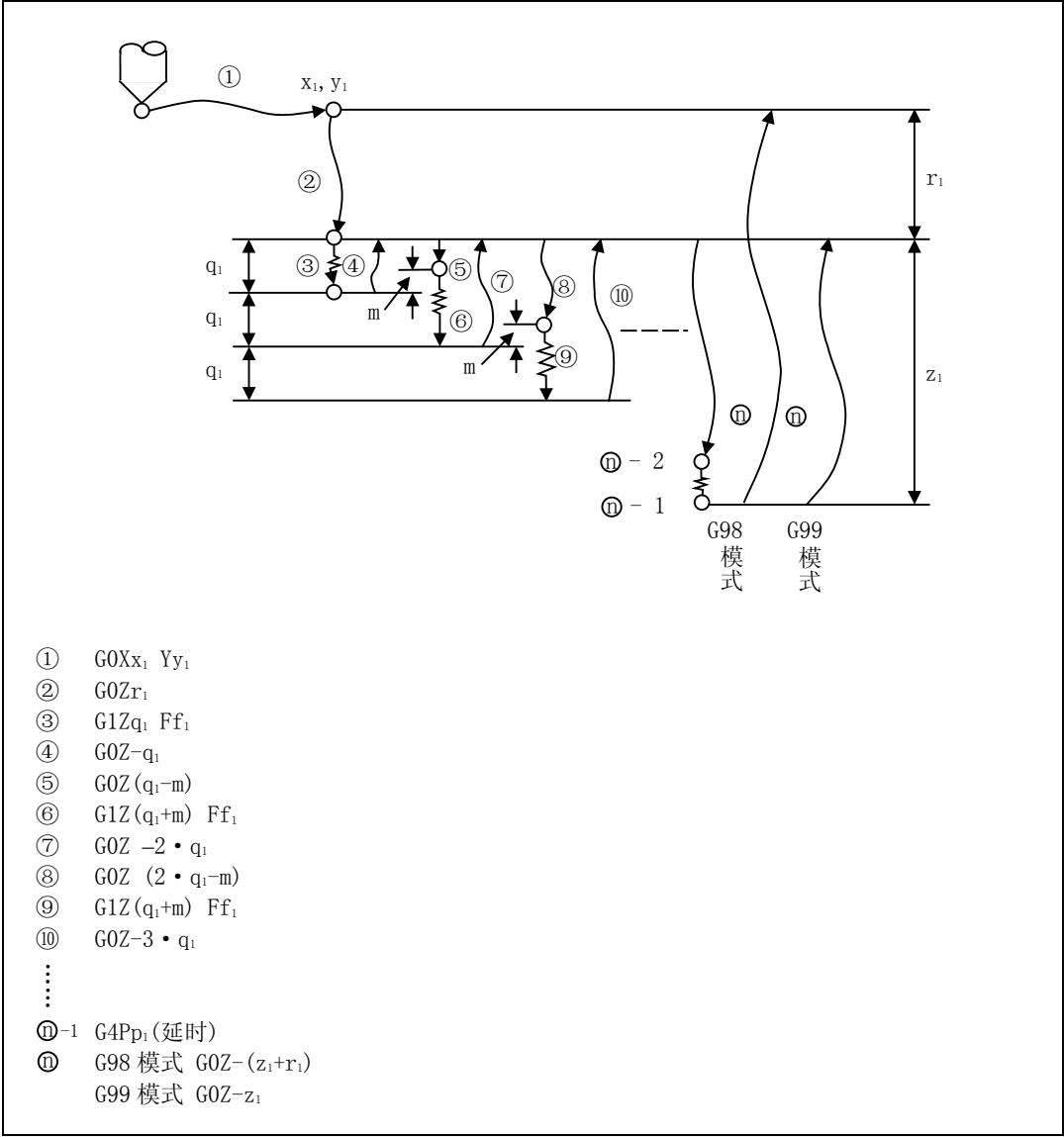
(c) G83（深孔钻孔循环）

程序

G83 Xx₁ Yy₁ Zz₁ Rr₁ Qq₁ Pp₁ Ff₁ , Ii₁ , Jj₁ ;

Q: 是用于指定每次切入量的地址，总是以增量值进行指定。

P: 在孔底的延时时间



动作曲线	, I	, J
①	有效	-
②	-	无效
③	-	无效
④	-	无效
⑤	-	无效
⑥	-	无效
⑦	-	无效
⑧	-	无效
⑨	-	无效
⑩	-	无效
:		
:		
n-2	-	无效
n-1	-	无效
N	-	有效

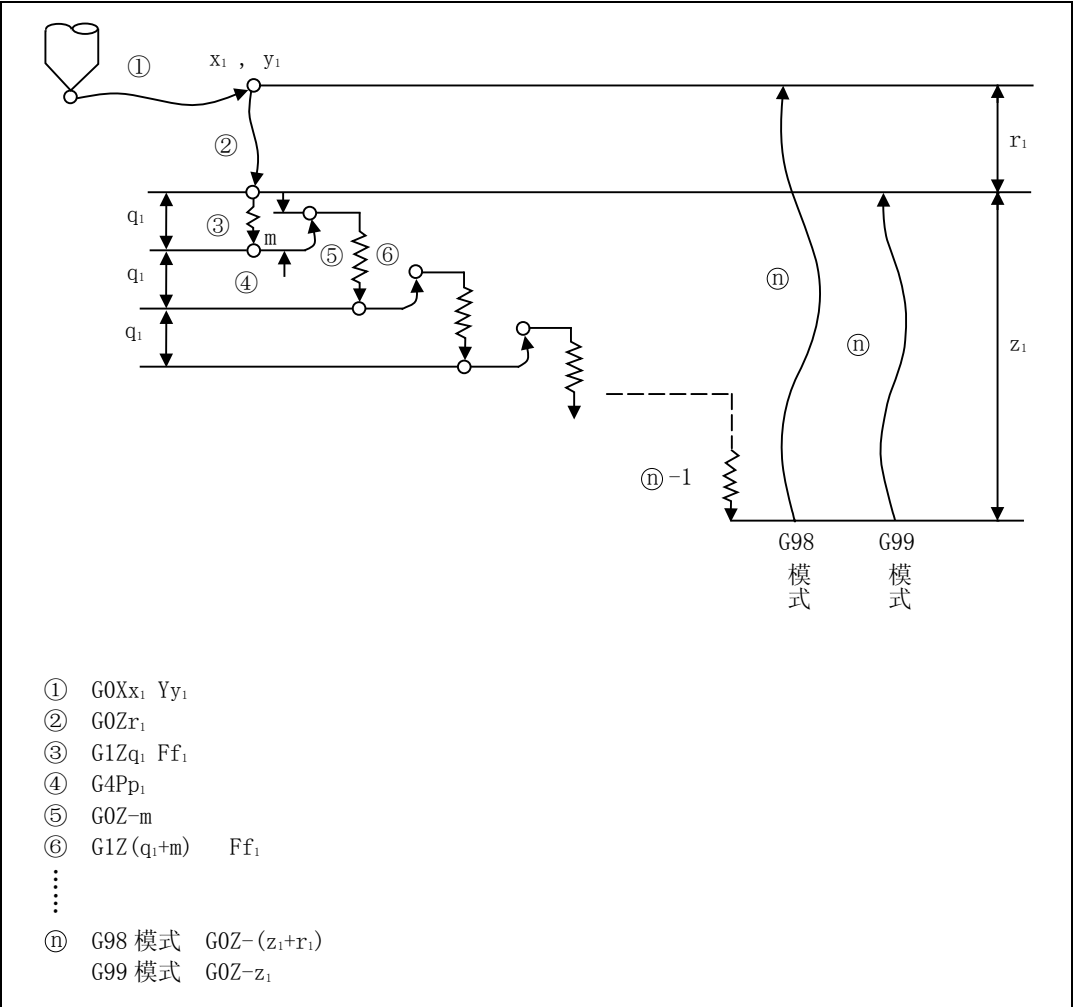
在G83中，进行第2次之后的切入时，在距加工位置还有m[mm]的位置上，从快速进给切换到切削进给。
到达孔底，则按照G98或G99模式返回。
m取决于参数#8013 G83返回。编程时，请确保 $q_1 > m$ 。
单节运转时的停止位置，为①②n指令完成时。

(d) G83.1（步进循环）

程序

G83.1 X_{x1} Y_{y1} Z_{z1} Q_{q1} R_{r1} F_{f1} P_{p1}, I_{i1}, J_{j1} ;

P: 延时指定



动作曲线	, I	, J
①	有效	-
②	-	无效
③	-	无效
④	-	-
⑤	-	无效
⑥	-	无效

:

:

n-1	-	无效
N	-	有效

在G83.1中，在进行第2次之后的切入时，以快速进给返回m[mm]，然后切换到切削进给。返回量m取决于参数#8013 G73返回。

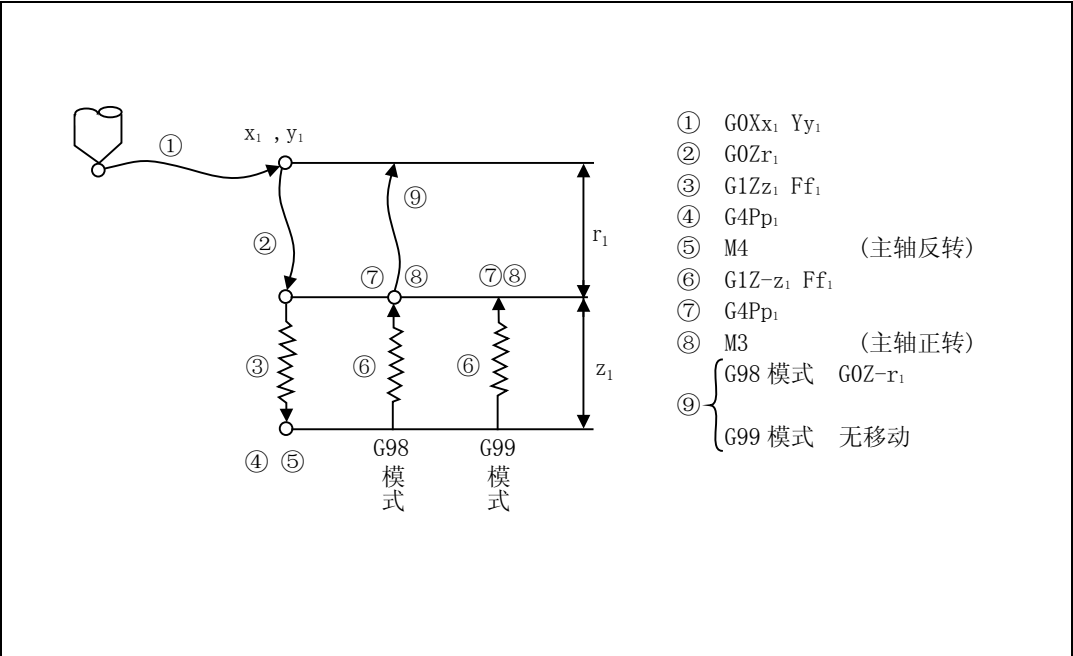
单节运转时的停止位置，为①②n指令完成时。

(e) G84（攻牙循环）

程序

G84 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ Pp₁, Rr₂, Ii₁, Jj₁ ;

P: 延时指定



动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	-
⑤	-	-
⑥	-	无效
⑦	-	-
⑧	-	-
⑨	-	有效

在执行G84时，进入超程取消状态，超程自动变为100%。当控制参数“G00空运转”打开时，空运转变为对定位指令有效。另外，如果在执行G84时按回馈等待按钮，则顺序③～⑥时，不是立即停止，而是在完成⑥之后再停止。进行顺序①、②、⑨的快速进给时，立即停止。

单节运转时的停止位置，为①②⑨指令完成时。

G84模态中，输出“攻牙中”的NC输出信号。

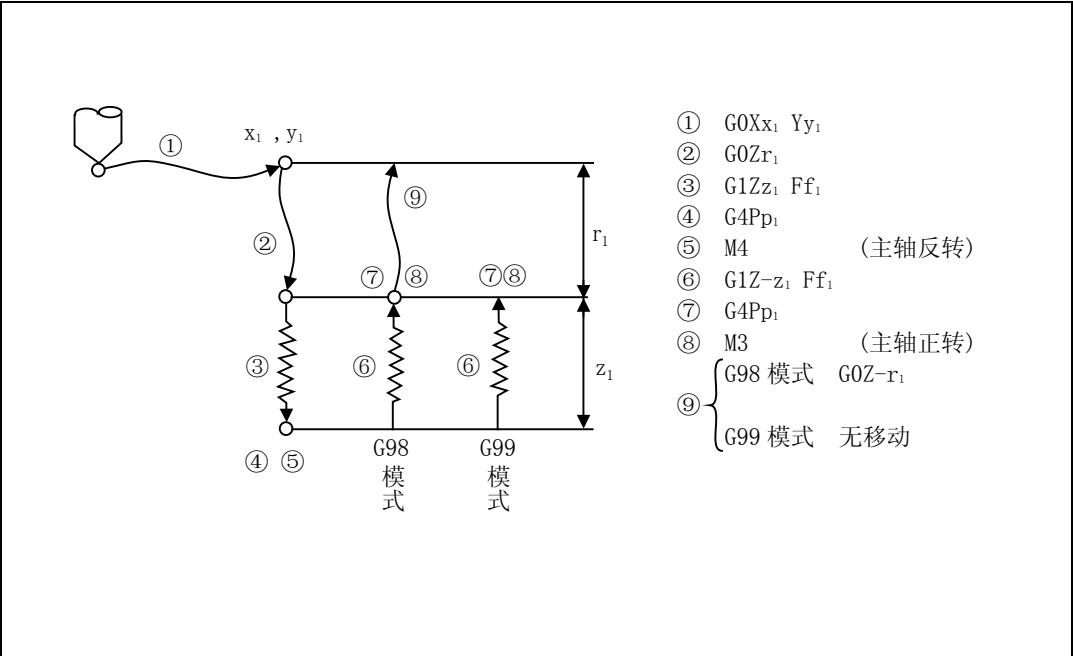
在G84同期攻牙模态中，不输出M3、M4、M5与S代码。

(f) G84.2（同期攻牙循环）

程序

G84.2 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ Pp₁, Ss₁, Ss₂, Ii₁, Jj₁ ;

P: 延时指定



动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	-
⑤	-	-
⑥	-	无效
⑦	-	-
⑧	-	-
⑨	-	有效

在执行G84.2时，进入超程取消状态，超程自动变为100%。当控制参数“G00空运转”打开时，空运转变为对定位指令有效。另外，如果在执行G84.2时按回馈等待按钮，则顺序③~⑥时，不是立即停止，而是在完成⑥之后再停止。进行顺序①、②、⑨的快速进给时，立即停止。

单节运转时的停止位置，为①②⑨指令完成时。

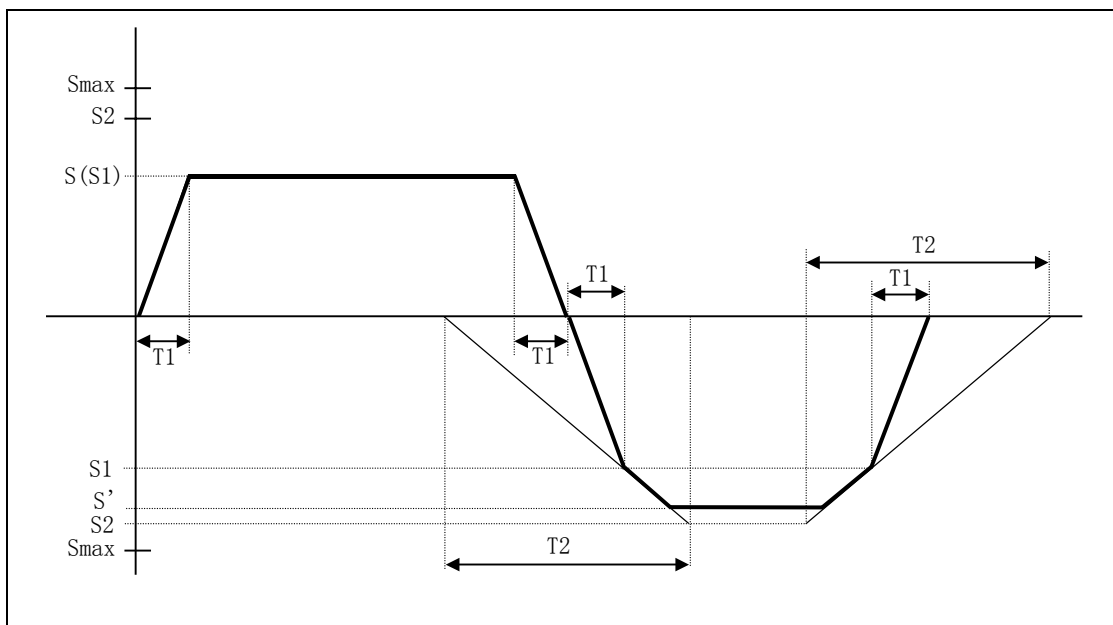
G84.2模态中，输出“攻牙中”的NC输出信号。

在G84.2同期攻牙模态中，不输出M3、M4、M5与S代码。

本功能，是通过在同期攻牙中，将主轴及钻孔轴的加减速曲线进行最多3级的多级化，让主轴的加减速曲线接近速度回环时的加减速曲线。加减速曲线可针对各齿轮，分别最多设定3级。

另外，从孔底返回时，根据返回时的主轴转速，可能会快速返回。返回时的主轴转速，作为模态信息被予以保持。

攻牙转速 < 返回时的主轴转速 ≤ 同期攻牙切换主轴转速2 时



S: 指令主轴转速

S': 返回时的主轴转速

S1: 攻牙转速（主轴基本规格参数#3013~#3016）

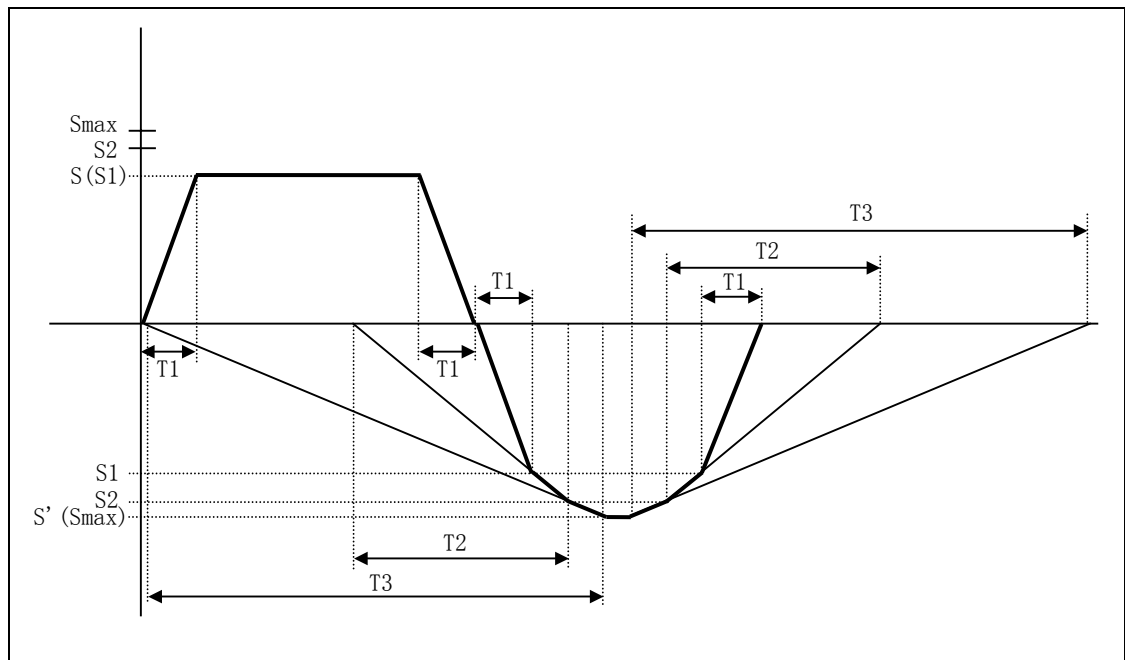
S2: 同期攻牙切换主轴转速2（主轴基本规格参数#3037~#3040）

Smax: 最高转速（主轴基本规格参数#3005~#3008）

T1: 攻牙时间常数（主轴基本规格参数#3017~#3020）

T2: 同期攻牙切换时间常数2（主轴基本规格参数#3041~#3044）

同期攻牙切换主轴转速2 < 返回主轴转速 时



S : 指令主轴转速

S' : 返回时的主轴转速

S1 : 攻牙转速（主轴基本规格参数#3013～#3016）

S2 : 同期攻牙切换主轴转速2（主轴基本规格参数#3037～#3040）

Smax: 最高转速（主轴基本规格参数#3005～#3008）

T1 : 攻牙时间常数（主轴基本规格参数#3017～#3020）

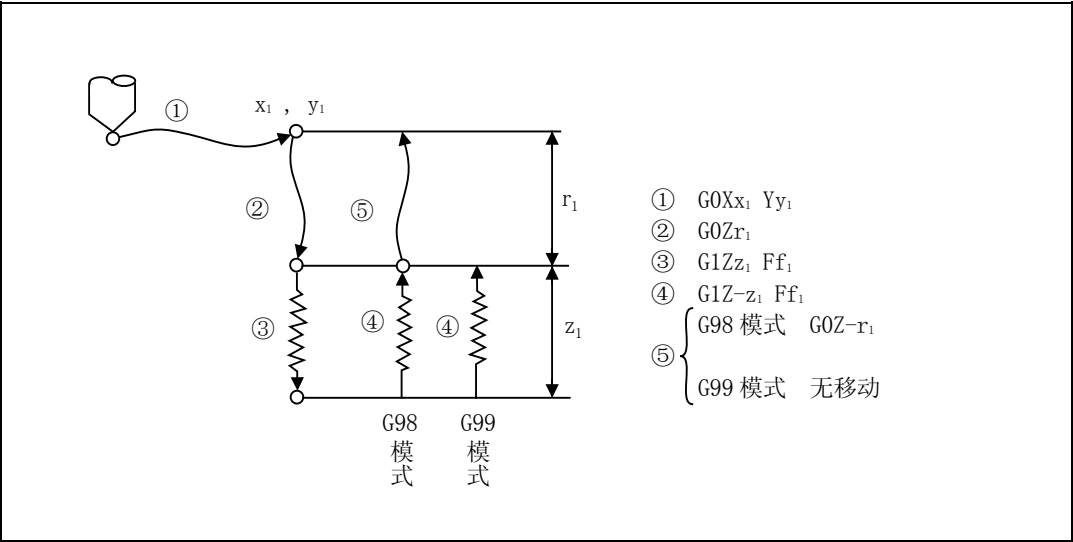
T2 : 同期攻牙切换时间常数2（主轴基本规格参数#3041～#3044）

T3 : 同期攻牙切换时间常数3（主轴基本规格参数#3045～#3048）

(g) G85（镗孔）

程序

G85 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ , Ii₁ , Jj₁ ;



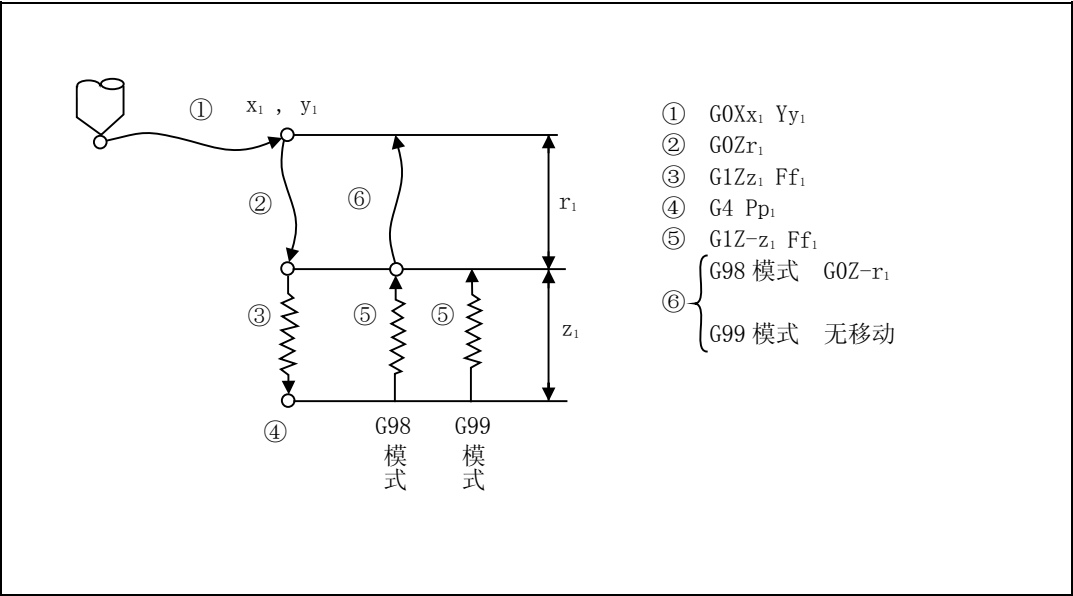
动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	无效
⑤	-	有效

单节运转时的停止位置，为①②④或⑤的指令完成时。

(h) G89（镗孔）

程序

G89 Xx₁ Yy₁ Zz₁ Rr₁ Ff₁ Pp₁ , Ii₁ , Jj₁;



动作曲线	i1	j1
①	有效	-
②	-	无效
③	-	无效
④	-	-
⑤	-	无效
⑥	-	有效

单节运转时的停止位置，为①②⑤或⑥的指令完成时。



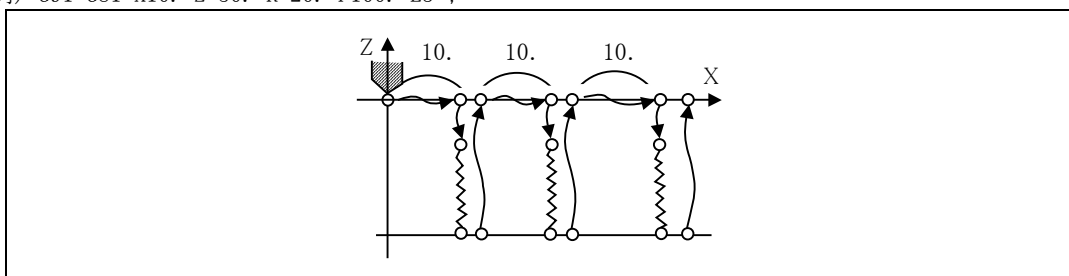
固定循环使用注意事项

- (1) 指令固定循环之前，必须预先通过辅助功能（M3 或 M4）将主轴旋转到规定的方向上。
- (2) 在固定循环模式中，如果在该节中有基本轴、附加轴或 R 的数据，则进行钻孔动作，如果没有数据，则不进行钻孔动作。
但是，即使有 X 轴数据，如果是延时（G04）时间指令，则不进行钻孔动作。
- (3) 请在进行钻孔动作的节（含基本轴、附加轴或 R 数据的节）中指令孔加工数据（Q、P、I、J、K）。
- (4) 除 G80 外，固定虚幻也会被 G00~G03、G33 指令取消。与固定循环在同一节内指定，则变为如下。
（以 m 代表 00~03、33，以 n 代表固定循环的代码，则）

Gm	Gn	X_	Y_	Z_	R_	Q_	P_	L_	F_	
执行	跳跃					执行		跳跃	记忆	
	Gn	Gm	X_	Y_	Z_	R_	Q_	P_	L_	F_
	跳跃					执行		跳跃	记忆	

 但是，当进行 G02、G03 指令时，R 作为圆弧半径使用。
- (5) 如果与固定循环指令在同一节内指令了辅助功能，则在最初进行定位动作时，输出 M 代码及 MF。或根据 FIN（完成信号）进入下一动作。
当有次数指定时，仅在首次进行上述控制。
- (6) 当与固定循环控制轴在同一节内指令了其他控制轴（例如旋转轴、附加轴），则首先移动其他控制轴，然后执行固定循环。
- (7) 当没有指定重复次数 L 时，看作为 L1。与固定循环 G 代码指令在同一节内指令了 L0，则对孔加工数据进行记忆，但是不进行孔加工。
（例）G83.1 X_ Y_ Z_ R_ Q_ P_ F_ L0_ ;

执行	仅记忆有地址的代码
----	-----------
- (8) 当执行了固定循环时，固定循环程序中指令的模式指令，仅在固定循环子程序内有效，对于调用固定循环的程序模式没有影响。
- (9) 无法在固定循环子程序中调用其他的子程序。
- (10) 固定循环子程序的移动指令跳跃小数点。
- (11) 在增量值模式中，当重复次数 L 大于 2 时，定位也变为每次增量。
（例）G91 G81 X10. Z-50. R-20. F100. L3 ;



- (12) 返回时的主轴转速值比主轴转速值小时，主轴转速值在返回时也生效。
- (13) 根据设定了参数的主轴的转速、时间常数，当第 2 级及第 3 级的加减速斜率比各自前一级的斜率更大时，前一级的斜率有效。
- (14) 当主轴基本规格参数的攻牙转速及同期攻牙切换主轴转速 2 的设定值超过最高转速时，主轴的转速被以最高转速进行钳位。

13. 程序辅助功能

13.6 钻孔固定循环（MELDAS 特殊格式）

- (15) 当返回时的主轴转速不为 0 时，攻牙返回超程值无效。
- (16) 如下例所示，在某根轴的移动方向反转的节中，由于伺服系统的负载会变的非常大，所以请不要在加工程序中指令就位宽度。
- ```
G0 X100. , I10.0 ;
X-200. ;
```
- (17) 如果增大就位宽度编程指令的就位宽度，则虽然能够缩短定位时间及直线插补时间，但是在下一节开始时，上一节的位置误差量也会增大，在实际加工中，可能会导致故障。
- (18) 由于每隔一定时间就进行就位宽度与位置误差量的比较，所以就位时的位置误差量小于就位宽度的设定值。
- (19) 当就位宽度编程指令的就位宽度较小时，通过指令减速检查或参数进行的就位检查可能会先生效。
- (20) 可选择同步・非同期攻牙。

基本规格参数

| #    | 项目    | 内容           | 设定范围       |
|------|-------|--------------|------------|
| 1513 | stapM | 同期攻牙选择用 M 代码 | 0~99999999 |

通过本参数设定值的辅助功能代码，选择同期攻牙模式。

M 功能可在攻牙指令之前，在同一节中进行指令。

同期・非同期攻牙的选择，取决于下表的组合。

| 程序指令（G84/G84.2） | 组合 |   |   |   |   |   |   |   |     |   |   |   |
|-----------------|----|---|---|---|---|---|---|---|-----|---|---|---|
|                 | 0  | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 无指令 |   |   |   |
| #1229（bit4）     | 0  | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0   | 0 | 1 | 1 |
| M 功能代码（M**）     | ×  | ○ | × | ○ | × | ○ | × | ○ | ×   | ○ | × | ○ |
| 同期・非同步选择        | 非  | 非 | 非 | 非 | 同 | 同 | 同 | 同 | 非   | 同 | 同 | 同 |

0: G84    ×: 不指令    非: 非同期攻牙

1: G84.2    ○: 指令    同: 同期攻牙

（注 1）请不要使用 M00、01、02、30、96、97、98、99。

（注 2）根据机种不同，无法根据 M 功能进行选择。

13. 程序辅助功能

13.6 钻孔固定循环（MELDAS 特殊格式）

13.6.1 起始点与R点基准返回；G98、G99



功能及目的

在固定循环中，可选择最终顺序中的返回基准是设置为R点，还是设置为起始基准。



指令格式

G98: 起始基准返回  
G99 : R 点基准重新启动



详细说明

G98/G99的模式与重复次数指定之间的关系，如下表所示。

| 钻孔次数        | 程序例                                      | G98<br><div>接通电源时，根据<br/>M02、M30 取消时，<br/>复位按钮</div> | G99                 |
|-------------|------------------------------------------|------------------------------------------------------|---------------------|
| 仅执行<br>1 次  | G81 X100. Y100.<br>Z-50. R25. F1000 ;    | <br>变为起始基准返回。                                        | <br>变为 R 点基准返回。     |
| 执行<br>2 次以上 | G81 X100. Y100.<br>Z-50. R25. L5 F1000 ; | <br>第 1 次 第 2 次 最终次<br>全部变为起始基准返回。                   | <br>第 1 次 第 2 次 最终次 |



程序例

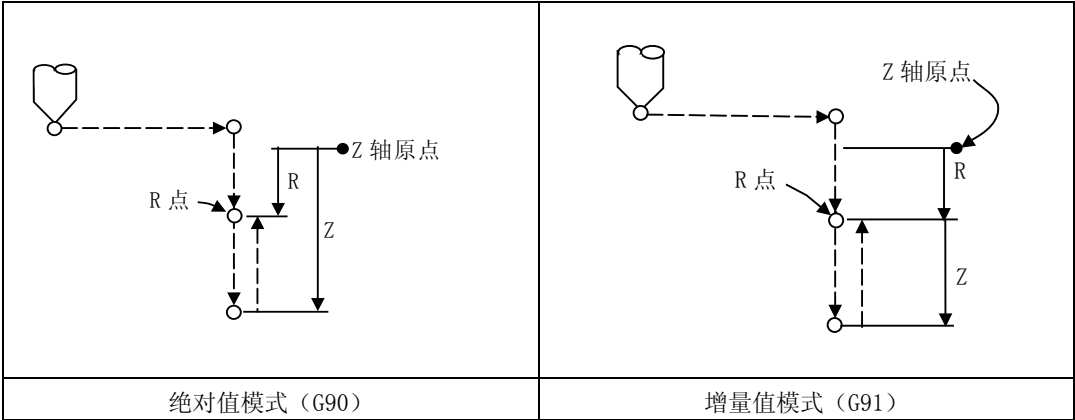
(例 1) G82 Zz<sub>1</sub> Rr<sub>1</sub> Pp<sub>1</sub> Ff<sub>1</sub> L0 ;仅记忆孔加工数据（不执行）  
Xx<sub>1</sub> Yy<sub>1</sub> ; 通过G82 模式执行钻孔动作

通过 L 进行固定循环的重复次数指定。但是，当没有 L1 或 L 指令时，变为固定循环 1 次。可指定的范围为 1~9999。

指令 L0 则仅进行孔加工数据的记忆。

G8△ Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Rr<sub>1</sub> Pp<sub>1</sub> Qq<sub>1</sub> Ff<sub>1</sub> Ll<sub>1</sub> ;

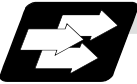
根据绝对值模式（G90）与增量值模式（G91），数据的考虑方法发生如下图所示的变化。



X、Y、Z 请指定带符号的指令值。R 表示距原点的坐标值，所以必须有符号。  
在固定循环中，孔加工数据如下例所示被保持。当接收到 G80 及 01 组的 G 指令（G00、G01、G02、G03、G2.1、G3.1、G33）时，固定循环模式被取消。

|                                                                                                  |                              |
|--------------------------------------------------------------------------------------------------|------------------------------|
| (例 2) N001 G81 XX <sub>1</sub> YY <sub>1</sub> ZZ <sub>1</sub> RR <sub>1</sub> FF <sub>1</sub> ; |                              |
| N002 G81 ;                                                                                       | 仅选择固定循环顺序                    |
| N003 XX2 YY2 ;                                                                                   | 变更定位点，执行固定循环                 |
| N004 M22 ;                                                                                       | 仅执行 M22                      |
| N005 G04 XX3 ;                                                                                   | 仅执行延时                        |
| N006 G92 XX4 YY4 ;                                                                               | 仅执行坐标系设定                     |
| N007 G28(G30) Z0 ;                                                                               | 仅执行参考点（原点）返回                 |
| M008 ;                                                                                           | 无作业                          |
| N009 G99 ZZ2 RR2 FF2 L0 ;                                                                        | 仅记忆孔加工数据                     |
| N010 XX5 YY5 LL5 ;                                                                               | 变更定位点，执行 15 次 R 点返回的固定循环     |
|                                                                                                  | 变更定位点，执行固定循环                 |
| N011 G98 XX6 YY6 ZZ6 RR6 ;                                                                       | 根据 N001 以前的 01 组模态，执行 W 轴之后， |
| N012 WW1 ;                                                                                       | 执行固定循环                       |

13.6.2 固定循环模式中的工件坐标设定



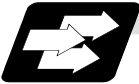
功能及目的

对于指定轴，在设定的工件坐标系中移动。  
定位完成后的R点定位或Z轴移动之后，Z轴有效。  
（注）关于地址Z及R，在工件坐标切换时，即使是相同的值，也请再次进行编程。

（例）G54 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub>;  
G81 Xx<sub>1</sub> Yy<sub>2</sub> Zz<sub>2</sub> Rr<sub>2</sub> ;  
G55 Xx<sub>3</sub> Yy<sub>3</sub> Zz<sub>2</sub> Rr<sub>2</sub> ;……即使Z、R与上一次相同，也必须再次指令  
Xx<sub>4</sub> Yy<sub>4</sub> ;  
Xx<sub>5</sub> Yy<sub>5</sub> ;

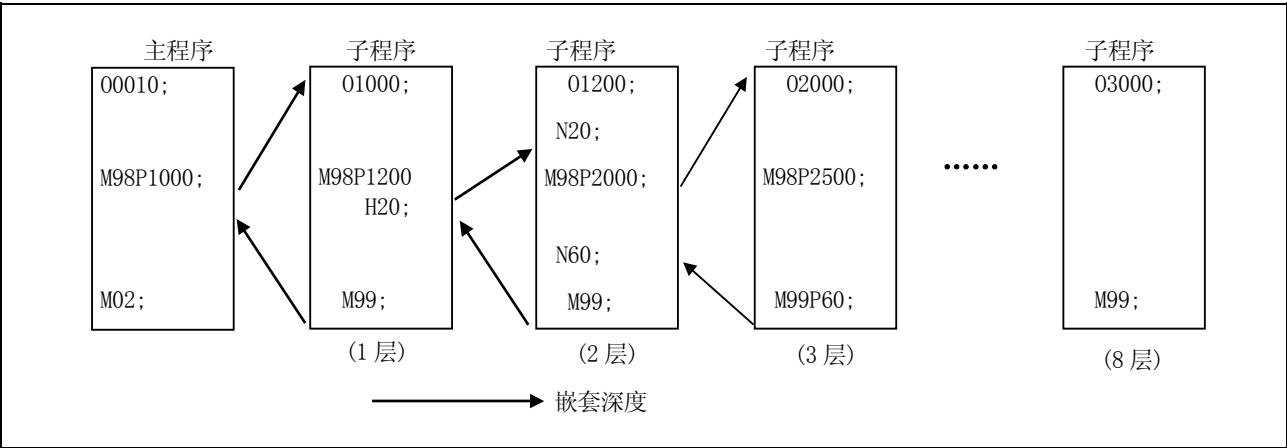
13.7 子程序控制；M98、M99

13.7.1 通过M98、M99指令调用子程序



功能及目的

对于某个固定的顺序及反复使用的参数，可预先作为子程序记忆在内存中，在需要时，从主程序中调用，加以使用。子程序的调用是通过M98指令进行，从子程序恢复则是通过M99指令进行。也可以在子程序中进行一步调用其他的子程序，其深度最多可达到8层。



根据纸带记忆编辑、子程序控制、固定循环的附加组合，可执行的功能如下表所示。

|                  | 案例 1 | 案例 2 | 案例 3 | 案例 4 |
|------------------|------|------|------|------|
| 1. 纸带记忆编辑        | 有    | 有    | 有    | 有    |
| 2. 子程序控制         | 无    | 有    | 有    | 无    |
| 3. 固定循环          | 无    | 无    | 有    | 有    |
| 功 能              |      |      |      |      |
| 1. 内存运转          | ○    | ○    | ○    | ○    |
| 2. 纸带编辑（主内存）     | ○    | ○    | ○    | ○    |
| 3. 子程序调用         | ×    | ○    | ○    | ×    |
| 4. 子程序的变量指定（注 2） | ×    | ○    | ○    | ×    |
| 5. 子程序的多层调用（注 3） | ×    | ○    | ○    | ×    |
| 6. 固定循环          | ×    | ×    | ○    | ○    |
| 7. 固定循环用子程序的编辑   | ×    | ×    | ○    | ○    |

（注 1）○标记的功能可使用，×标记的功能不可使用

（注 2）在 M98 中，虽然不进行变量迁移，但是如果有变量指令的选项，则可使用子程序内的变量指令。

（注 3）多层调用最多可进行 8 层调用。



指令格式

子程序的调用

**M98** **P**\_\_ **H**\_\_ **L**\_\_ ;

M98

:

子程序调用指令

M99

:

子程序返回指令

P

:

调用的子程序内的程序编号（如省略则为本程序）  
但是，仅当内存运转与 MDI 运转时，可省略 P。  
（最大 8 位数值）

H

:

调用的子程序内的顺序编号（如省略则为起始节）  
（最大 5 位数值）

L

:

子程序的重复次数  
（如果省略则看作为 L1，在 L0 时不执行。）  
（通过 4 位数值，1~9999 次）  
例如，  
M98 P1 L3 与  
M98 P1 ;  
M98 P1 ;  
M98 P1 ;  
等价。

从子程序中返回

**M99** **P**\_\_ **L**\_\_ ;

P

:

返回对象顺序编号（如省略，则返回到调用节的下一节）

L

:

变更重复次数后的次数



子程序的创建与注册

除非在最终节中作为单独节加入了子程序结束命令M99（P\_ L\_），则子程序的格式与通常的内存运转用加工程序的格式相同。

O △△△△△△△△ ;

..... ;

..... ;

..... ;

..... ;

M99;

%(EOR)

作为子程序编号的程序编号

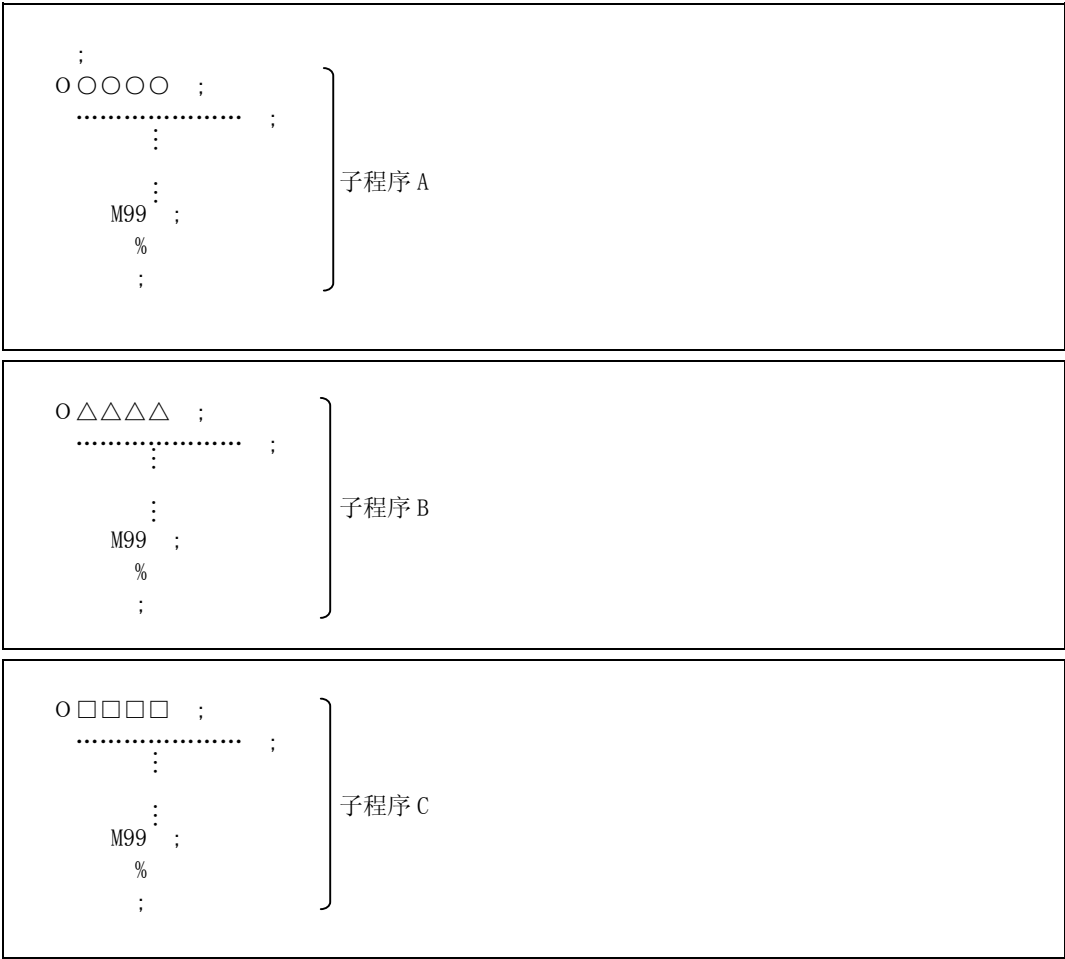
子程序主体

子程序返回指令

注册结束代码

(1) 通过设定显示装置“编辑”操作，注册如上所述的程序。详情请参阅“程序编辑”。

- (2) 子程序编号，仅可使用1~99999999中通过附加规格加以指定的种类。当纸带上没有程序编号时，以“程序输入”时的设定编号进行注册。
- (3) 从程序中调用子程序的深度，最多可达到8层，超过8层则发生程序错误（P230）。
- (4) 向内存中注册时，并不区分子程序、主程序的区别，而是按照读入的顺序注册，所以，请确保主程序与子程序的编号不一致。（如果一致，则注册时作为错误“E11”被处理）



- (5) 虽然主程序采用内存、纸带、MDI运转中的任何一个均可，但是子程序必须记忆在内存中。

(6) 作为子程序嵌套对象的，除了M98外，还有下述指令。

- G65     宏调用
- G66     模态调用
- G66.1   模态调用
- G 代码调用
- 辅助功能调用
- MDI 插入
- 自动刀具长度测定
- 宏插入
- 多级跳跃功能

(7) 以下指令不可作为子程序的嵌套对象，在8层以后可调用。

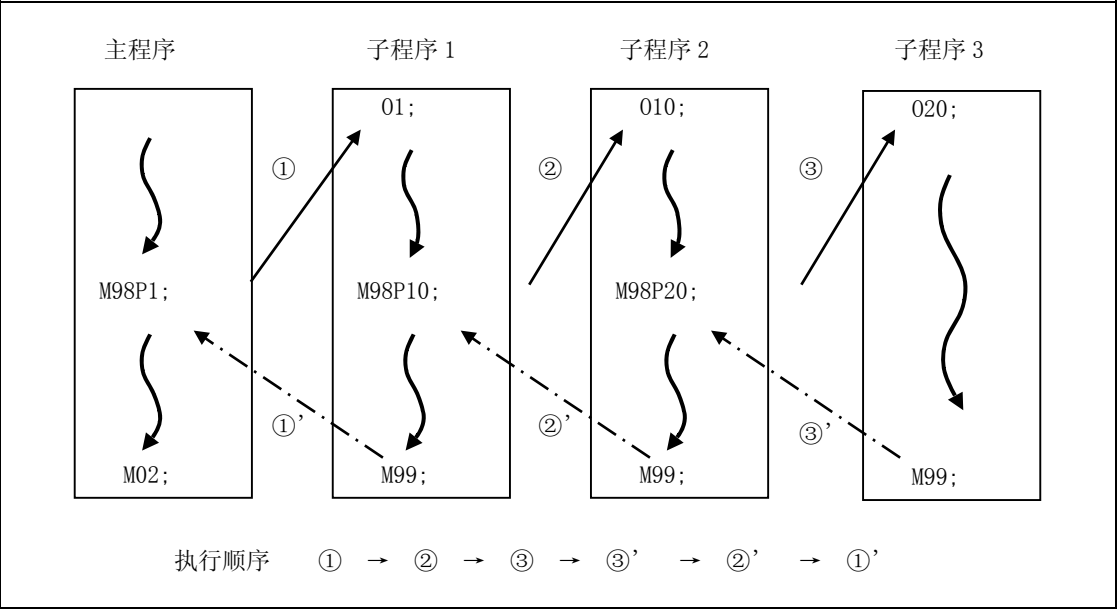
- 固定循环
- 曲线循环

(8) 希望重复使用子程序时，如果编程了M98 P<sub>i</sub> L<sub>i</sub>；，则重复执行<sub>i</sub>次。



程序例 1

子程序的调用为3次（称为嵌套3层）时



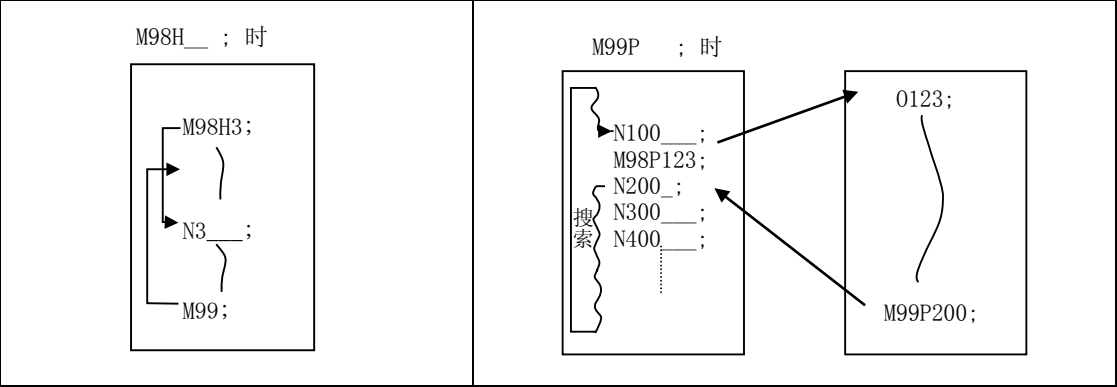
(1) 进行嵌套时，M98与M99请务必一一对应（对于①有①’、对于②有②’…………）。

(2) 模态信息不区分主程序、子程序，而是按照执行的顺序被覆盖，所以在编程时，调用子程序并执行之后，请充分注意模态数据的状态。



程序例 2

M98 H\_ ; M99 P\_ ; 是有调用指令的程序中的顺序编号。



注意事项

- (1) 当找不到指定的P（程序编号）时，发生程序错误“P232”。
- (2) M98 P\_ ; M99 ; 节不可进行单节停止。但是，当有除O、N、P、L、H以外的地址时，可进行单节停止。（如果X100. M98 P100 ; , 则在执行X100. 之后，分歧到0100.。）
- (3) 在主程序中指令M99，则返回开头。（MDI模式下也相同。）
- (4) 虽然可在纸带、BTR运转中通过M98 P\_ 分歧到子程序，但是无法通过M99 P\_ ; 指定返回的顺序编号。（P\_被跳跃。）
- (5) 通过M99 P\_ ; 指定顺序编号，需要较长的搜索时间，请加以注意。

13.8 变量指令



功能及目的

对于程序中的某个地址，指定变量代替直接指定数值，则可通过在执行程序时，随时根据情况给该变量赋值，让程序具有融通性、通用性。



指令格式

#△△△=□□□□□□□□或#△△△= [式]



详细说明

| (1) 变量的表示方法 |                 | 例          |
|-------------|-----------------|------------|
| (a) #m      | m 为由 0~9 构成的数值。 | #100       |
| (b) # [f]   | f 为运算式，指以下内容。   | # [-#120]  |
|             | 数值 m            | 123        |
|             | 变量              | #543       |
|             | 运算式 运算符 运算式     | #110+#119  |
|             | － (负) 运算式       | －#120      |
|             | [运算式]           | [#119]     |
|             | 函数 [运算式]        | SIN [#110] |

- (注 1) 标准运算符为+、-、\*、/这 4 种。
- (注 2) 如果没有用户宏规格，则无法使用函数。
- (注 3) 当变量编号为负时发生错误 (P241)。
- (注 4) 不正确的变量表示例如下。

| 错误      |   | 正确                           |
|---------|---|------------------------------|
| #6/2    | → | # [6/2] (#6/2 被解释为 [#6] /2。) |
| #--5    | → | # [- [-5] ]                  |
| #- [#1] | → | # [-#1]                      |

(2) 变量的种类

变量包括如下表所示的种类。

| 种类         | 编号      |         | 功 能                 |
|------------|---------|---------|---------------------|
| 通用变量 200 组 | 通用变量 1  | 通用变量 2  | • 主程序、子程序、各宏程序均可使用。 |
|            | 500~599 | 100~199 |                     |
| 局部变量       | 1 ~ 33  |         | 在宏程序内，可在局部使用。       |
| 系统变量       | 1000~   |         | 在系统中，用途固定。          |
| 固定循环变量     | 1 ~ 32  |         | 固定循环程序内的<br>局部变量    |

(注 1) 所有的通用变量，在切断电源时也被保持。

(注 2) 通过参数 (#1128 RstVC1、#1129 PwrVC1)，即使重新启动及切断电源，也可将变量设定为〈空〉。

(3) 变量的引用

除O、M及/（反斜杠）外，可对所有的地址使用。

(a) 变量值的直接使用

X#1                    将#1的值作为X值使用。

(b) 使用变量值的补数

X-#2        将#2的值改变符号之后，作为X值使用。

(c) 定义变量。

#3 = #5        变量#3使用等价变量#5的值。

#1 = 1000变量#1使用等价值1000（看作为1000.）。

(d) 定义变量运算式。

#1 = #3 + #2 - 100 使用#3 + #2 - 100. 的运算结果，作为#1的值。

X [#1 + #3 + 1000] 使用#1 + #3 + 1000的运算结果，作为X的值。

(注 1) 无法与地址在同一节中进行变量的定义。请分开进行定义。

|                 |   |                        |
|-----------------|---|------------------------|
| 错误              |   | 正确                     |
| X#1 = #3 + 100; | → | #1 = #3 + 100;<br>X#1; |

(注 2) [ ]最多可使用 5 层。

#543 = -[[[[[#120]/2+15.]\*3-#100]/#520+#125+#128]\*#130+#132];

(注 3) 在变量的定义中，对于变量的个数及字符数没有限制。

(注 4) 变量的值请控制在 0~±99999999 的范围内。

超过这一范围则可能会无法进行正确运算。

(注 5) 变量的定义，从定义时开始生效。

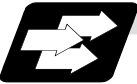
|                          |                    |
|--------------------------|--------------------|
| #1 = 100 ;               | #1 = 100           |
| #1 = 200 #2 = #1 + 200 ; | #1 = 200, #2 = 400 |
| #3 = #1 + 300 ;          | #3 = 500           |

(注 6) 变量的引用，总是看作为末尾有小数点的数值。

#100 = 10时  
X#100 ; 成为X10.。

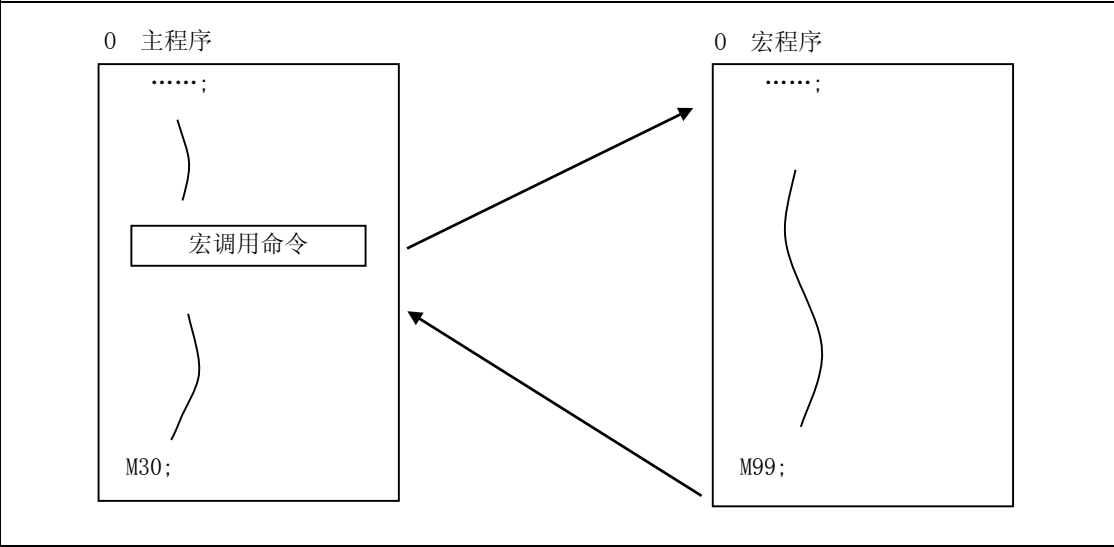
13.9 用户宏

13.9.1 用户宏：G65、G66、G66.1、G67



功能及目的

通过变量指令的组合，可进行宏程序的调用、各种运算、与PLC之间的数据输入输出、控制、判定、分歧等多种命令，可执行测量等。



宏程序是使用变量、运算命令、控制命令等，作为专用控制功能，加以子程序化的程序。在主程序中，可根据需要，使用宏调用命令调用、使用这些专用控制功能（宏程序）。

| G 代码  | 功 能                   |
|-------|-----------------------|
| G65   | 用户宏 纯调用               |
| G66   | 用户宏 模态调用 A（移动指令调用）    |
| G66.1 | 用户宏 模态调用 B（每节调用）      |
| G67   | 用户宏 模态调用（G66、G66.1）取消 |



详细说明

- （1）输入G66指令或G66.1指令，则在输入G67（取消）指令之前，执行完有移动指令的节之后，或是每节执行完之后，调用指定的用户宏程序。
- （2）在同一程序中，G66（G66.1）、G67指令必须成对出现。

13.9.2 宏调用命令

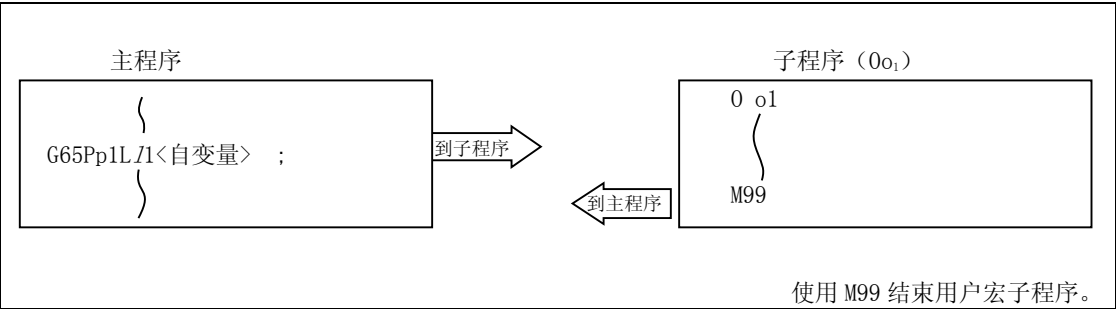


功能及目的

宏调用命令中，包括仅调用命令节的纯调用，与通过调用模式中的各节进行调用的模态调用（类型A、类型B）。



纯调用



格式

```
G65 P__ L__ <自变量>;
P__ : 程序编号
L__ : 重复次数
```

当在用户宏子程序中，需要将<自变量>作为局部变量进行转移时，请在地址后面指定实际的值。此时，自变量与地址无关，可使用符号、小数点。另外，自变量有如下2种类型。

- (1) 自变量指定 I  
格式: A\_ B\_ C\_ .....X\_ Y\_ Z\_

详细说明

- (a) 除G、L、N、O、P外，可使用所有的地址指定自变量。
- (b) 除I、J、K外，无需按字母顺序指定。
- (c) I、J、K必须按照字母顺序指定。  
I\_ J\_ K\_...可  
J\_ I\_ K\_...不可
- (d) 无需指定的地址可省略。
- (e) 自变量指定 I 中可使用地址与用户宏主体内变量编号的对应关系，如下表所示。

| 地址・变量编号对应   |       | 能够使用调用命令的地址 |       |
|-------------|-------|-------------|-------|
| 自变量指定 I 的地址 | 宏内的变量 | G65、G66     | G66.1 |
| A           | #1    | ○           | ○     |
| B           | #2    | ○           | ○     |
| C           | #3    | ○           | ○     |
| D           | #7    | ○           | ○     |
| E           | #8    | ○           | ○     |
| F           | #9    | ○           | ○     |
| G           | #10   | ×           | × *   |
| H           | #11   | ○           | ○     |
| I           | #4    | ○           | ○     |
| J           | #5    | ○           | ○     |
| K           | #6    | ○           | ○     |
| L           | #12   | ×           | × *   |
| M           | #13   | ○           | ○     |
| N           | #14   | ×           | × *   |
| O           | #15   | ×           | ×     |
| P           | #16   | ×           | × *   |
| Q           | #17   | ○           | ○     |
| R           | #18   | ○           | ○     |
| S           | #19   | ○           | ○     |
| T           | #20   | ○           | ○     |
| U           | #21   | ○           | ○     |
| V           | #22   | ○           | ○     |
| W           | #23   | ○           | ○     |
| X           | #24   | ○           | ○     |
| Y           | #25   | ○           | ○     |
| Z           | #26   | ○           | ○     |

○标记：可使用

×标记：不可使用

\* 标记：G66.1模态中可使用

(2) 自变量指定 II

格式：A\_B\_C\_I\_J\_K\_I\_J\_K\_...

详细说明

- (a) 除地址A、B、C以外，以I、J、K为1组的自变量，最多可指定10组。
- (b) 相同地址重复时，请按照规定的顺序指定。
- (c) 无需指定的地址可省略。
- (d) 自变量指定 II 中可使用地址与用户宏主体内变量编号的对应关系，如下表所示。

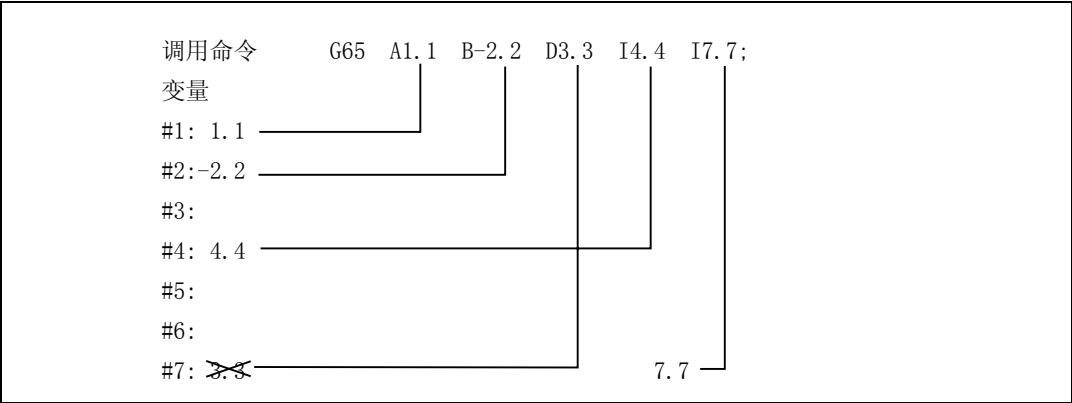
| 自变量指定 II 地址 | 宏内的变量 | 自变量指定 II 地址 | 宏内的变量 |
|-------------|-------|-------------|-------|
| A           | #1    | J5          | #17   |
| B           | #2    | K5          | #18   |
| C           | #3    | I6          | #19   |
| I1          | #4    | J6          | #20   |
| J1          | #5    | K6          | #21   |
| K1          | #6    | I7          | #22   |
| I2          | #7    | J7          | #23   |
| J2          | #8    | K7          | #24   |
| K2          | #9    | I8          | #25   |
| I3          | #10   | J8          | #26   |
| J3          | #11   | K8          | #27   |
| K3          | #12   | I9          | #28   |
| I4          | #13   | J9          | #29   |
| J4          | #14   | K9          | #30   |
| K4          | #15   | I10         | #31   |
| I5          | #16   | J10         | #32   |
|             |       | K10         | #33   |

(注1) I、J、K后面附加的1~10是表示指令组的顺序，实际命令中不需要。

(3) 自变量指定 I、II 的并存

在自变量指定中使用 I、II 两种类型时，如果命令了与同一变量相对应的地址，则后命令的生效。

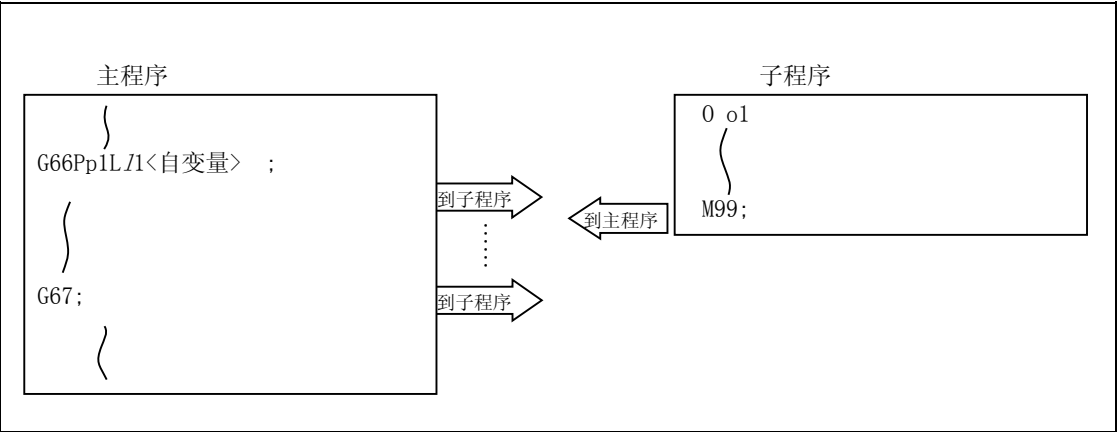
(例 1)



上例中，当对#7变量指令了D3.3 I7.7两个自变量时，后指令的I7.7有效。



模态调用 A（移动指令调用）



如果在G66到G67之间指令了有移动指令的节，则在执行完该移动指令后，执行指定的用户宏子程序。执行次数为每次调用时 I1 次。  
<自变量>与纯调用相同。

格式

**G66 P\_\_ L\_\_ <自变量>;**  
  
P\_\_ : 程序编号  
L\_\_ : 重复次数

详细说明

- (1) 输入G66指令，则在输入G67（取消）指令之前，执行完有移动指令的节之后，或是每节执行完之后，调用指定的用户宏程序。
- (2) 在同一程序中，G66、G67指令必须成对出现。  
如果在没有G66指令的情况下指令了G67，则发生程序错误。



模态调用 B（每节调用）

从G66.1到G67之间，在所指令的每一指令节中，无条件的调用指定的用户宏程序，执行指定的次数。  
<自变量>与纯调用相同。

格式

|                                   |        |
|-----------------------------------|--------|
| <b>G66.1 P__ L__ &lt;自变量&gt;;</b> |        |
| P__                               | : 程序编号 |
| L__                               | : 重复次数 |

详细说明

- (1) G66.1模式中，除被读出的各指令节的O、N及G代码外，全部不执行，而是作为自变量加以使用。但是，在最后指定的G代码，或是在O、N以外之后指定的N代码，作为自变量使用。
- (2) G66.1模式中的所有有含义节，与在节的开头指令G65 P\_时相同。  
(例1)  
G66.1 P1000; 模式中的  
N100 G01 G90 X100. Z100. F400 R1000; 与  
N100 G65 P1000 G01 G90 X100. Z200. F400 R1000; 相同。  
(注1) G66.1模式中的G66.1指令节中，也进行调用，自变量的地址及与变量编号的对应关系，与G65（纯调用）相同。
- (3) G66.1模式中，可作为新变量使用的G、N指令值范围，受到通常的NC指令值的限制。
- (4) 程序编号O、顺序编号N及模态G代码作为模态信息更新。



G 代码宏调用

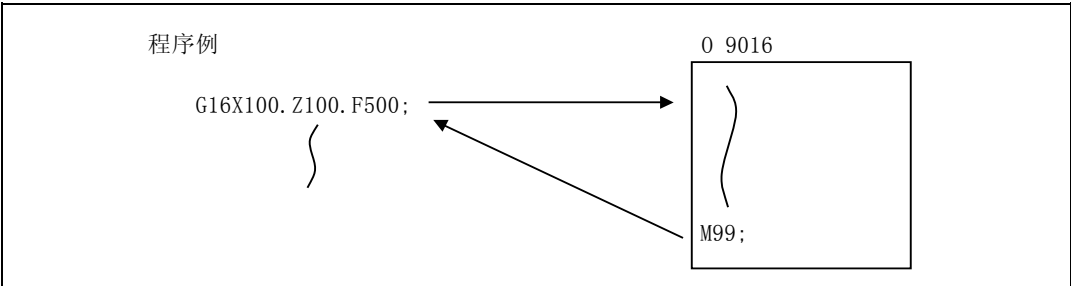
只需通过指令G代码，就可调用指定程序编号的用户宏程序。

格式

|            |                     |
|------------|---------------------|
| <b>G**</b> | <b>&lt;自变量&gt;;</b> |
| G**        | : 调用宏的 G 代码         |

详细说明

- (1) 上述命令与下述命令进行相同的动作，但是对应哪一条命令，则通过参数，对各G代码分别加以设定。
- a: M98 P△△△△;
  - b: G65 P△△△△△ <自变量>;
  - c: G66 P△△△△△ <自变量>;
  - d: G66.1 P△△△△△ <自变量>;
- 当设定了与上述c、d对应的参数时，为了取消模态调用，请在指令调用代码之后，或是在用户宏中指令取消指令（G67）。
- (2) 通过参数设定调用宏的\*\*与想调用的宏程序编号P△△△△间的对应关系。
- (3) 在本命令中，最多可使用G00～G255中的10个。（通过参数#1081 Gmac\_P，也可在G01～G99范围内）
- (4) 通过G代码调用的用户宏子程序中，无法指令。





辅助指令宏调用（M、S、T、B 代码宏调用）

只需通过指令M（或S、T、B）代码，就可调用指定程序编号的用户宏子程序。（M为注册过的代码，S、T、B是以所有的代码为对象。）

格式

|                                        |                         |
|----------------------------------------|-------------------------|
| <b>M** ; (或 S** ; , T** ; , B** ;)</b> |                         |
| M**                                    | : 进行宏调用的 M (或 S、T、B) 代码 |

详细说明

- (1) 上述命令与下述命令进行相同的动作，但是对应哪一条命令，则通过参数，对各G代码分别加以设定。  
(S、T、B代码也相同。)
- a: M98 P△△△△;  
b: G65 P△△△△M\*\*;  
c: G66 P△△△△M\*\*;  
d: G66.1 P△△△△M\*\*;
- } M98、M\*\*不输出。
- 当设定了与上述c、d对应的参数时，为了取消模态调用，请指令调用代码之后，或是在用户宏中指令取消指令（G67）。
- (2) 通过参数设定调用宏的\*\*与想调用的宏程序编号P△△△△间的对应关系。能够注册的M代码为M00～M95范围内，最多10个。  
但是，注册时所使用的代码，请使用除该机械的基本必须代码及M0、M1、M2、M30、M96～M99以外的代码。
- (3) 虽然与M98一样显示在设定显示装置上，但是不输出M代码、MF。
- (4) 在通过M代码调用的用户宏子程序中，指令了上述的已注册过的辅助指令时，不进行宏调用，而是作为通常的辅助指令加以使用。
- (5) S、T、B代码是以所有代码为对象，调用S、T、B功能所指定程序编号的子程序。
- (6) M代码最多可指定10个，当未达到10个时，请按照如下方式设定参数。

| [宏一览表] |                  |   |           |
|--------|------------------|---|-----------|
|        | <代码><类型><程序 NO.> |   |           |
| M[01]  | 20               | 0 | 8000      |
| M[02]  | 21               | 0 | 8001      |
| M[03]  | 9999             | 0 | 199999999 |
| M[04]  | 9999             | 0 | 199999999 |
| M[05]  | 9999             | 0 | 199999999 |
|        | :                | : | :         |
|        | :                | : | :         |
| M[10]  | 9999             | 0 | 199999999 |

-----M20 指令时，用于以类型 0（M98 型）调用 08000 的设定

-----M21 指令时，用于以类型 0（M98 型）调用 08001 的设定

-----不使用的参数请设定如左



M98 指令与 M65 指令的不同

- (1) G65中可指定自变量，而M98中无法指定。
- (2) M98中可指定顺序编号，而G65、G66、G66.1无法指定。
- (3) M98是在执行完M98节中除M、P、H、L以外的指令之后，执行子程序，而G65则是不做任何处理就进入子程序。
- (4) 当M98的节中包含O、N、P、H、L以外的地址时，进行单节停止，而G65则不进行单节停止。
- (5) M98局部变量的等级固定，而G65则随着嵌套深度而变化。  
(例如#1，在M98的前后具有相同的含义，而在G65的前后，则分别具有不同的含义。)
- (6) M98的调用嵌套深度，G65、G66、G66.1合计最多8层，而G65则是G66、G66.1合计最多4层。

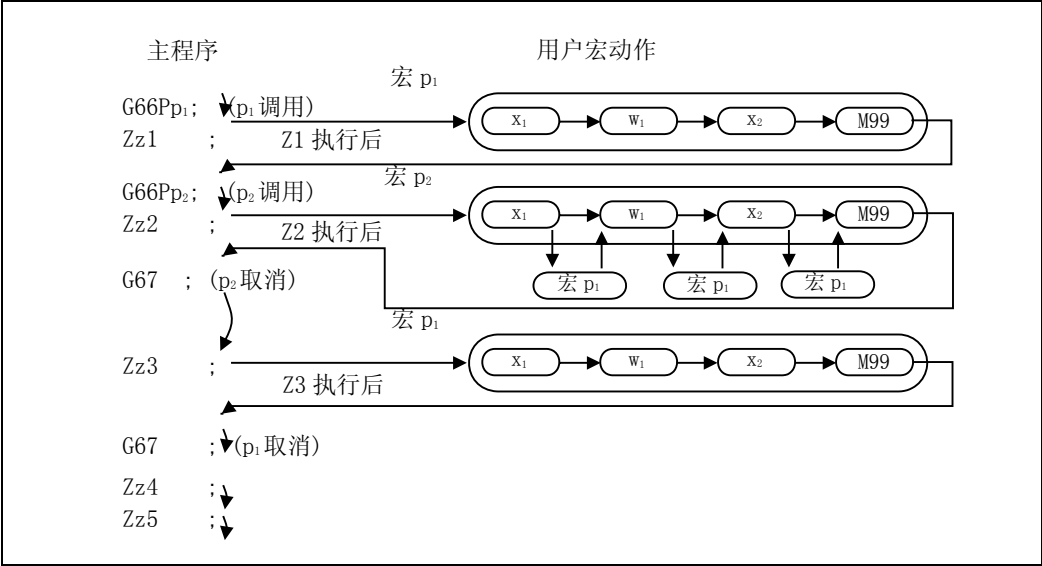


宏调用指令的嵌套

通过纯调用、模态调用进行宏子程序的调用时，最多可进行4层嵌套。  
宏调用命令时的自变量，仅在调用的宏等级内有效。宏调用的嵌套深度最多为4层，在程序中，可在各层宏调用中，分别将自变量作为局部变量使用。

- (注 1) 如果进行 G65、G66、G66.1、G 代码宏调用或辅助指令宏调用，则看作为 1 层嵌套，局部变量的等级也增大 1 级。
- (注 2) 模态调用 A 中，每次执行移动指令时，都将指定的用户宏作为子程序进行调用，而当嵌套了多层 G66 指令时，即使是对于宏之内的移动指令，也在每次执行轴移动时，调用下一个用户宏程序。  
从后面指令的用户宏子程序开始，依次调用。

(例 1)



13.9.3 ASCII代码宏



功能及目的

可通过参数预先设定已注册过的子程序（宏程序）与代码的对应关系，然后通过加工程序中指令ASCII代码，调用宏程序。  
本功能可在G、M、S、T、B的辅助指令宏调用功能基础上使用。

执行例1) M98型

主程序

子程序

00002;  
:  
D2000;  
:  
M30;

0200  
:  
:  
:  
M99;

向通用变量#146 中输出 2000 后，  
以 M98 子程序调用类型，调用程序编号为 200 的子程序。

参数

|                        |           |
|------------------------|-----------|
| # 7401 (ASCII 调用有效/无效) | 1 (有效)    |
| # 7402 (ASCII 代码)      | D         |
| # 7403 (调用类型)          | 0 (M98 型) |
| # 7404 (程序编号)          | 200       |
| # 7405 (通用变量编号)        | 146       |

执行例2) M65型

主程序

子程序

00003;  
:  
A500;  
:  
M30;

03000  
:  
:  
:  
M99;

向局部变量#1 输出 500 后，  
以 G65 宏调用类型，调用程序编号为 3000 的子程序。

参数

|                        |           |
|------------------------|-----------|
| # 7411 (ASCII 调用有效/无效) | 1 (有效)    |
| # 7412 (ASCII 代码)      | A         |
| # 7413 (调用类型)          | 1 (G65 型) |
| # 7414 (程序编号)          | 3000      |
| # 7415 (通用变量编号)        | 100 (不使用) |



指令格式

|                     |                                       |
|---------------------|---------------------------------------|
| <code>□****;</code> | 指定地址与代码。                              |
| <code>□</code>      | : 调用宏的 ASCII 代码 (1 个字符)               |
| <code>****</code>   | : 输出到变量的值或运算式<br>(设定范围: ±999999.9999) |



详细说明

- (1) 上述命令与下述命令进行相同的动作。通过参数, 设定各ASCII代码分别对应哪一个命令。
- 0: M98 P△△△△;
  - 1: G65 P△△△△△〈自变量〉;
  - 2: G66 P△△△△△〈自变量〉;
  - 3: G66.1 P△△△△△〈自变量〉;
- 当设定了与上述2、3对应的参数时, 为了取消模态调用, 请在指令调用代码之后, 或是在用户宏中指令取消指令 (G67)。
- (2) 通过参数, 设定调用宏的ASCII代码与想调用的宏程序编号P△△△△间的对应关系。  
最多可注册2个ASCII代码。
- (3) 代码的部分虽然输出到变量中, 但是输出目标因调用类型与地址而异。

- (a) M98型时  
输出到通用变量中, 通过参数设定变量编号。  
当与第1个地址 (参数#7401) 相对应时, 第1个变量编号 (参数#7404) 输出到指定的通用变量。
- (b) G65/G66/G66.1型时  
输出到局部变量。变量编号因地址而异, 对应于下表。

| 地址 | # | 地址 | #  | 地址 | #  | 地址 | #  | 地址 | #  | 地址 | #  |
|----|---|----|----|----|----|----|----|----|----|----|----|
| A  | 1 | F  | 9  | K  | 6  | P  | 16 | U  | 21 | Z  | 26 |
| B  | 2 | G  | 10 | L  | 12 | Q  | 17 | V  | 22 |    |    |
| C  | 3 | H  | 11 | M  | 13 | R  | 18 | W  | 23 |    |    |
| D  | 7 | I  | 4  | N  | 14 | S  | 19 | X  | 24 |    |    |
| E  | 8 | J  | 5  | O  | 15 | T  | 20 | Y  | 25 |    |    |

(注) 可使用的地址如下。

|                                       |
|---------------------------------------|
| A, B, D, F, H, I, J, K, M, Q, R, S, T |
|---------------------------------------|



程序例

在地址A进行顶心座的控制，所示的是加工细长工件的程序。

参数

|                |           |
|----------------|-----------|
| #7411 (有效/无效)  | 1 (有效)    |
| #7412 (地址)     | A         |
| #7413 (类型)     | 1 (G65 型) |
| #7414 (程序编号)   | 500       |
| #7415 (通用变量编号) | 100 (不使用) |

主程序

```
G28 XZ;
A1;
G00 X20. Z0.;
G01 X15. F100;
G01 Z100. F200;
X50 Z120. ;
A0;
M30 ;
```

子程序

```
O500;
IF[#1 EQ 0]GOTO 10;
G53 G00 A-205. ;
G53 G01 A-200. F100;
GOTO N20;
N10 M26;
G53 G0 A-400. ;
N20 M99;
```

顶心座前进

顶心座后退

地址 A 被转换为 G65P500

293



限制事项

- (1) 在通过ASCII代码调用的程序中，使用ASCII代码调用宏  
在通过ASCII代码调用的程序中，无法使用ASCII代码调用宏。  
其他方式请参阅下表。  
当结果为判定无法进行调用时，全部看作为通常的指令，进行操作。

|     |               | 被调用方  |         |               |     |
|-----|---------------|-------|---------|---------------|-----|
|     |               | ASCII | GMSTB 宏 | G65/G66/G66.1 | M98 |
| 调用方 | ASCII         | ×     | ×       | ○             | ○   |
|     | GMSTB 宏       | ×     | ×       | ○             | ○   |
|     | G65/G66/G66.1 | ○     | ○       | ○             | ○   |
|     | M98           | ○     | ○       | ○             | ○   |

- (2) 宏调用指令的嵌套深度  
通过纯调用（G65）、模态调用（G66/G66.1）调用宏子程序，最多可嵌套4层。  
另外，宏调用命令时的自变量，仅在调用的宏等级内有效。  
宏调用的嵌套深度最多为4层，在程序中，可在各层宏调用中，分别将自变量作为局部变量使用。
- (3) 子程序调用指令的嵌套深度  
从子程序中调用子程序（M98），如果以主程序为0，则最多可嵌套8层。  
子程序的嵌套对象为以下指令。
- (a) M98
  - (b) G65 G66 G66.1
  - (c) G代码调用 辅助功能调用（M/S/T/B）
  - (d) MDI插入
  - (e) 自动刀具长度测定
  - (f) 多级跳跃功能
- 另外，指令以下的指令时，与嵌套无关。
- (g) 固定循环
  - (h) 宏插入

(4) 指令的优先顺序

当在ASCII代码的地址中指定' M' 时，则与该机械的基本必须代码重复。此时，根据代码值，按照如下的优先顺序识别指令。

- (a) M98、M99（子程序调用指令）  
M00（程序停止指令）、M01（可选停止指令）  
M02、M30（终止指令）  
M96、M97（宏插入指令）
- (b) 有ASCII代码宏指令时
- (c) 通常的指令

在ASCII代码的地址中指定' S' 、' T' 、' B' 时，按照以下的优先顺序识别指令。

- (a) 有辅助指令宏调用指令（S、T、B）时
- (b) 有ASCII代码宏指令时
- (c) 通常的指令

其他地址中，当没有ASCII代码宏指令时，被识别为通常的指令。想使用的指令与ASCII代码宏指令重复时，需要使用ASCII代码指令调用的宏程序。  
但是，可能会如（5）所示，无条件的作为通常的指令进行处理。

(5) 将ASCII代码宏指令中设定的地址作为通常的指令加以处理的条件

- (a) 同一节内有数据设定指令（G10）时
- (b) 同一节内的G代码宏调用指令（M、S、T、B、ASCII时也相同）之后，进行了ASCII代码宏调用时  
例）在ASCII代码宏中设定地址'D'（G65型），且在宏调用（G65型）中设定了M50时

|           |                           |
|-----------|---------------------------|
| M50 D200; | 有自变量（将#7设置为200），且进行M代码宏调用 |
|-----------|---------------------------|

- (c) 输入参数中时
- (d) 地址前有逗号（，）时 例），D，R等
- (e) 固定循环内的指令
- (f) 通过G代码宏调用所调用的宏子程序内的指令  
（通过M、S、T、B、ASCII调用宏时也相同）

13.9.4 变量



功能及目的

用户宏中所使用的变量，需要变量规格与用户宏规格两方面的规格。  
本数控装置的变量中，#33以外的局部变量、通用变量及系统变量的修正量，即使电源关闭，也会被保持。  
(通用变量可通过参数#1129 PwrVC1予以清空。)



变量的嵌套

用户宏规格时的变量，可使用变量设置变量编号（嵌套），或是以<运算式>替换变量编号。  
当使用<运算式>时，仅可使用1个四则运算。

(例 1) 变量的嵌套

|                                      |                                                                     |
|--------------------------------------|---------------------------------------------------------------------|
| #1=10 #10=20 #20=30 ;<br>#5=# (#1) ; | } 通过#1=10 变为 (#1) =#10。<br>通过#10=20 变为#10=#20。<br>因此，#5=#20，即#5=30。 |
|--------------------------------------|---------------------------------------------------------------------|

|                                             |                                                                        |
|---------------------------------------------|------------------------------------------------------------------------|
| #1=10 #10=20 #20=30 #5=1000;<br># (#1) =#5; | } 通过#1=10 变为 (#1) =#10。<br>通过#10=20 变为#10=#20。<br>因此，#20=#5，即#20=1000。 |
|---------------------------------------------|------------------------------------------------------------------------|

(例 2) 变量的嵌套指定例

|                                |                                 |
|--------------------------------|---------------------------------|
| #10=5 ;<br>变为##10=100; #5=100。 | <运算式>##10=100;与#10=100;起到相同的作用。 |
|--------------------------------|---------------------------------|

(例 3) 以<运算式>替换变量编号。

|                                                                                              |                                                                     |
|----------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| #10=5;<br># (#10+1) = 1000;<br># (#10-1) = -1000;<br># (#10*3) = 100;<br># (#10 / 2) = -100; | 变为#6 = 1000。<br>变为#4 = -1000。<br>变为#15 = 100。<br>变为#3 = -100。(四舍五入) |
|----------------------------------------------------------------------------------------------|---------------------------------------------------------------------|



未定义变量

用户宏规格时的变量中，在接通电源后1次也没有使用过的变量，以及未在G65、G66、G66.1中作为自变量指定的局部变量，可作为<空>加以使用。另外，也可强制性的将变量设定为<空>。  
变量#0总是作为<空>变量使用，左边无法定义。

(1) 运算式

#1 = #0;            #1 = <空>  
#2 = #0 + 1;        #2 = 1  
#3 = 1 + #0;        #3 = 1  
#4 = #0 \* 10;       #4 = 0  
#5 = #0 + #0;       #5 = 0

运算式中，<空>起到与 0 相同的作用，所以请加以注意。  
<空> + <空> = 0  
<空> + <常数> = 常数  
<常数> + <空> = 常数

(2) 变量的引用

当仅引用了未定义的变量时，该地址被跳跃。  
#1=<空>时  
G0 X#1 Z1000; 与G0 Z1000; 等价。  
G0 X#1+10 Z1000; 与G0 X10 Z1000; 等价。

(3) 条件运算式

仅EQ、NE时，<空>与0不同。（#0为<空>）

| #101=<空>时             | #101=0 时             |
|-----------------------|----------------------|
| #101EQ#0<br><空>=<空>成立 | #101EQ#0<br>0=<空>不成立 |
| #101NE0<br><空>≠0 成立   | #101NE0<br>0≠0 不成立   |
| #101GE#0<br><空>≧<空>成立 | #101GE#0<br>0≧<空> 成立 |
| #101GT0<br><空>>0 不成立  | #101GT0<br>0>0 不成立   |
| #101LE#0<br><空>≦<空>成立 | #101LE#0<br>0≦<空> 成立 |
| #101LT0<br><空><0 不成立  | #101LT0<br>0<0 不成立   |

(注1) EQ及NE的比较，请仅对整数进行比较。当在有小数点以下数值的情况下进行比较时，请使用GE、GT、LE、LT。

13.9.5 变量的种类



通用变量

是可从任意位置共同使用的变量。通用变量的组数因规格而异。  
详情请参阅变量指令项。



局部变量（#1～#33）

除了可在调用1个宏子程序时，作为<自变量>加以定义外，还可以在主程序与子程序中，用于局部的变量，与宏之间无关，可重复。（最多4层）

|                                              |   |      |  |
|----------------------------------------------|---|------|--|
| G65 Pp <sub>1</sub> L l <sub>1</sub> <自变量> ； |   |      |  |
| p <sub>1</sub>                               | : | 程序编号 |  |
| l <sub>1</sub>                               | : | 重复次数 |  |

<自变量>为Aa1 Bb1 Cc1.....Zz1。  
通过<自变量>指定的地址，与用户宏主体内使用的局部变量编号之间的对应关系，如下表所示。

[自变量指定 I ]

| 调用命令       |       | 自变量地址 | 局部变量编号 |
|------------|-------|-------|--------|
| G65<br>G66 | G66.1 |       |        |
| ○          | ○     | A     | #1     |
| ○          | ○     | B     | #2     |
| ○          | ○     | C     | #3     |
| ○          | ○     | D     | #7     |
| ○          | ○     | E     | #8     |
| ○          | ○     | F     | #9     |
| ×          | × *   | G     | #10    |
| ○          | ○     | H     | #11    |
| ○          | ○     | I     | #4     |
| ○          | ○     | J     | #5     |
| ○          | ○     | K     | #6     |
| ×          | × *   | L     | #12    |
| ○          | ○     | M     | #13    |
| ×          | × *   | N     | #14    |
| ×          | ×     | O     | #15    |
| ×          | × *   | P     | #16    |

| 调用命令       |       | 自变量地址 | 局部变量编号 |
|------------|-------|-------|--------|
| G65<br>G66 | G66.1 |       |        |
| ○          | ○     | Q     | #17    |
| ○          | ○     | R     | #18    |
| ○          | ○     | S     | #19    |
| ○          | ○     | T     | #20    |
| ○          | ○     | U     | #21    |
| ○          | ○     | V     | #22    |
| ○          | ○     | W     | #23    |
| ○          | ○     | X     | #24    |
| ○          | ○     | Y     | #25    |
| ○          | ○     | Z     | #26    |
|            |       | —     | #27    |
|            |       | —     | #28    |
|            |       | —     | #29    |
|            |       | —     | #30    |
|            |       | —     | #31    |
|            |       | —     | #32    |
|            |       | —     | #33    |

上表中，×标记的自变量地址无法使用。但是，在G66.1模式中，可追加使用\*标记的自变量地址。  
另外，—标记没有对应的地址。

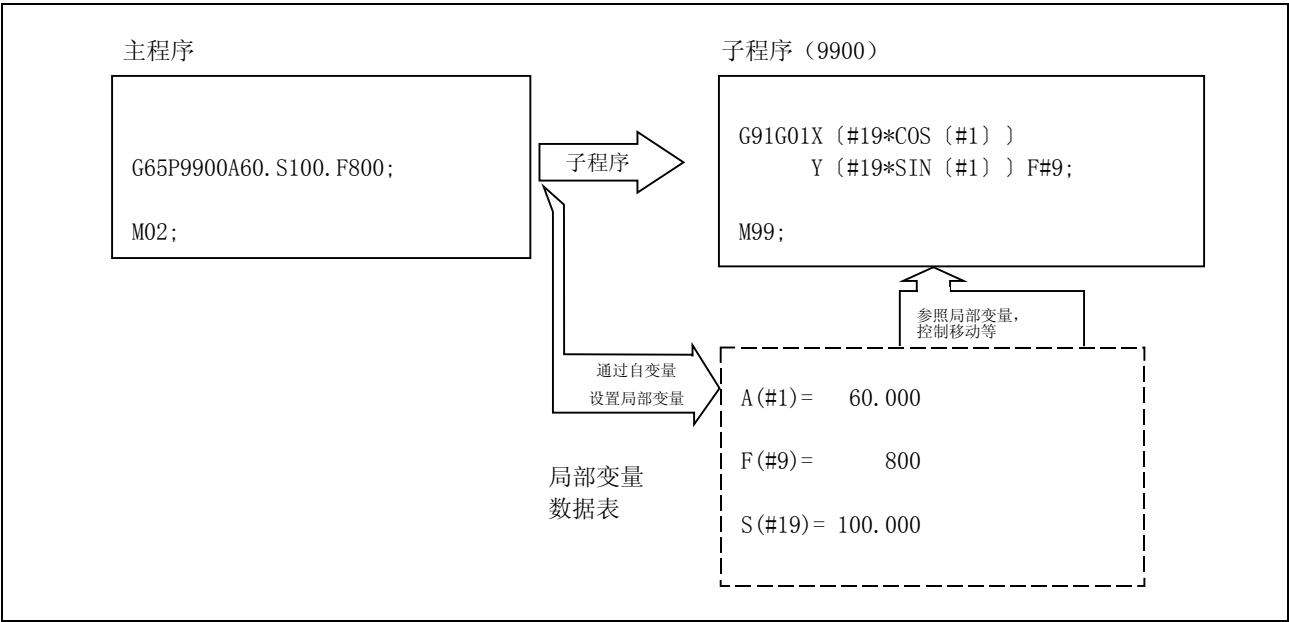
[自变量指定Ⅱ]

| 自变量指定Ⅱ地址 | 宏内的变量 |
|----------|-------|
| A        | #1    |
| B        | #2    |
| C        | #3    |
| I1       | #4    |
| J1       | #5    |
| K1       | #6    |
| I2       | #7    |
| J2       | #8    |
| K2       | #9    |
| I3       | #10   |
| J3       | #11   |
| K3       | #12   |
| I4       | #13   |
| J4       | #14   |
| K4       | #15   |
| I5       | #16   |

| 自变量指定Ⅱ地址 | 宏内的变量 |
|----------|-------|
| J5       | #17   |
| K5       | #18   |
| I5       | #19   |
| J6       | #20   |
| K6       | #21   |
| I7       | #22   |
| J7       | #23   |
| K7       | #24   |
| I8       | #25   |
| J8       | #26   |
| K8       | #27   |
| I9       | #28   |
| J9       | #29   |
| K9       | #30   |
| I10      | #31   |
| J10      | #32   |
| K10      | #33   |

(注1) I、J、K后面附加的1~10是表示指令组的顺序，实际命令中不需要。

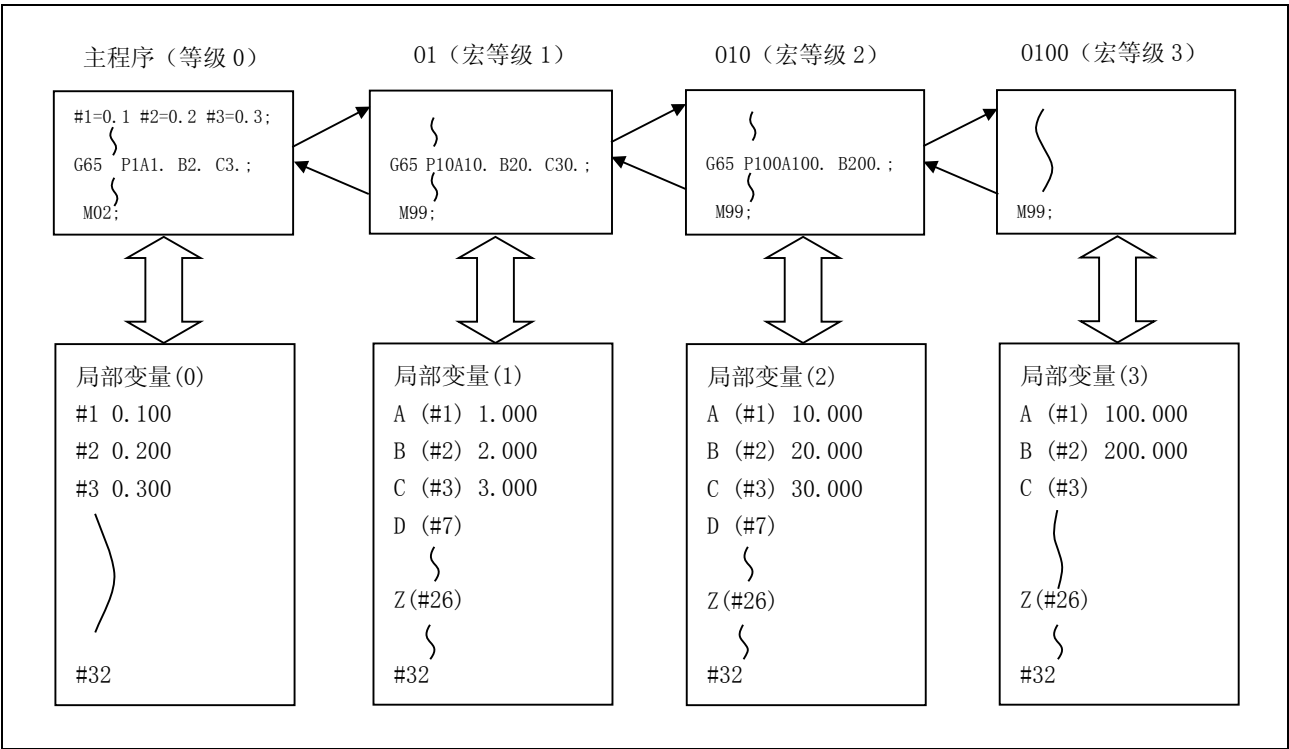
(1) 在调用宏时，可通过指定<自变量>，定义子程序的局部变量。（可在该子程序内自由使用局部变量。）



(2) 局部变量可针对调用宏（4层）的各等级，分别独立使用。

主程序（宏等级0）中，也可以独立设置局部变量。

但是，等级0的局部变量中无法使用自变量。



局部变量的使用状态被显示在设定显示装置上。

详情请参阅操作说明书。



## 宏、接口输入（#1000～#1035, #1200～#1295）：PLC→NC

通过读取变量编号#1000～#1035、#1200～#1295的值，可以了解接口输入信号的状态。

读取的变量值只有1或0（1=接点闭合、0=接点打开）2种。通过读取变量编号#1032的值，一次性读取#1000～#1031的所有输入信号。同样，通过读取变量编号#1033～#1035的值，读取#1200～#1231、#1232～#1263、#1264～#1295的输入信号。

#1000～#1035、#1200～#1295仅读取，无法放置到运算式的左边。

| 系统变量  | 点数 | 接口输入信号         | 系统变量  | 点数 | 接口输入信号         |
|-------|----|----------------|-------|----|----------------|
| #1000 | 1  | 寄存器 R72 的 bit0 | #1016 | 1  | 寄存器 R73 的 bit0 |
| #1001 | 1  | " 1            | #1017 | 1  | " 1            |
| #1002 | 1  | " 2            | #1018 | 1  | " 2            |
| #1003 | 1  | " 3            | #1019 | 1  | " 3            |
| #1004 | 1  | " 4            | #1020 | 1  | " 4            |
| #1005 | 1  | " 5            | #1021 | 1  | " 5            |
| #1006 | 1  | " 6            | #1022 | 1  | " 6            |
| #1007 | 1  | " 7            | #1023 | 1  | " 7            |
| #1008 | 1  | " 8            | #1024 | 1  | " 8            |
| #1009 | 1  | " 9            | #1025 | 1  | " 9            |
| #1010 | 1  | " 10           | #1026 | 1  | " 10           |
| #1011 | 1  | " 11           | #1027 | 1  | " 11           |
| #1012 | 1  | " 12           | #1028 | 1  | " 12           |
| #1013 | 1  | " 13           | #1029 | 1  | " 13           |
| #1014 | 1  | " 14           | #1030 | 1  | " 14           |
| #1015 | 1  | " 15           | #1031 | 1  | " 15           |

| 系统变量  | 点数 | 接口输入信号      |
|-------|----|-------------|
| #1032 | 32 | 寄存器 R72、R73 |
| #1033 | 32 | 寄存器 R74、R75 |
| #1034 | 32 | 寄存器 R76、R77 |
| #1035 | 32 | 寄存器 R78、R79 |

| 系统变量  | 点数 | 接口输入信号         | 系统变量  | 点数 | 接口输入信号         |
|-------|----|----------------|-------|----|----------------|
| #1200 | 1  | 寄存器 R74 的 bit0 | #1216 | 1  | 寄存器 R75 的 bit0 |
| #1201 | 1  | " 1            | #1217 | 1  | " 1            |
| #1202 | 1  | " 2            | #1218 | 1  | " 2            |
| #1203 | 1  | " 3            | #1219 | 1  | " 3            |
| #1204 | 1  | " 4            | #1220 | 1  | " 4            |
| #1205 | 1  | " 5            | #1221 | 1  | " 5            |
| #1206 | 1  | " 6            | #1222 | 1  | " 6            |
| #1207 | 1  | " 7            | #1223 | 1  | " 7            |
| #1208 | 1  | " 8            | #1224 | 1  | " 8            |
| #1209 | 1  | " 9            | #1225 | 1  | " 9            |
| #1210 | 1  | " 10           | #1226 | 1  | " 10           |
| #1211 | 1  | " 11           | #1227 | 1  | " 11           |
| #1212 | 1  | " 12           | #1228 | 1  | " 12           |
| #1213 | 1  | " 13           | #1229 | 1  | " 13           |
| #1214 | 1  | " 14           | #1230 | 1  | " 14           |
| #1215 | 1  | " 15           | #1231 | 1  | " 15           |

| 系统变量  | 点数 | 接口输入信号         | 系统变量  | 点数 | 接口输入信号         |
|-------|----|----------------|-------|----|----------------|
| #1232 | 1  | 寄存器 R76 的 bit0 | #1248 | 1  | 寄存器 R77 的 bit0 |
| #1233 | 1  | "      1       | #1249 | 1  | "      1       |
| #1234 | 1  | "      2       | #1250 | 1  | "      2       |
| #1235 | 1  | "      3       | #1251 | 1  | "      3       |
| #1236 | 1  | "      4       | #1252 | 1  | "      4       |
| #1237 | 1  | "      5       | #1253 | 1  | "      5       |
| #1238 | 1  | "      6       | #1254 | 1  | "      6       |
| #1239 | 1  | "      7       | #1255 | 1  | "      7       |
| #1240 | 1  | "      8       | #1256 | 1  | "      8       |
| #1241 | 1  | "      9       | #1257 | 1  | "      9       |
| #1242 | 1  | "     10       | #1258 | 1  | "     10       |
| #1243 | 1  | "     11       | #1259 | 1  | "     11       |
| #1244 | 1  | "     12       | #1260 | 1  | "     12       |
| #1245 | 1  | "     13       | #1261 | 1  | "     13       |
| #1246 | 1  | "     14       | #1262 | 1  | "     14       |
| #1247 | 1  | "     15       | #1263 | 1  | "     15       |

| 系统变量  | 点数 | 接口输入信号         | 系统变量  | 点数 | 接口输入信号         |
|-------|----|----------------|-------|----|----------------|
| #1264 | 1  | 寄存器 R78 的 bit0 | #1280 | 1  | 寄存器 R79 的 bit0 |
| #1265 | 1  | "      1       | #1281 | 1  | "      1       |
| #1266 | 1  | "      2       | #1282 | 1  | "      2       |
| #1267 | 1  | "      3       | #1283 | 1  | "      3       |
| #1268 | 1  | "      4       | #1284 | 1  | "      4       |
| #1269 | 1  | "      5       | #1285 | 1  | "      5       |
| #1270 | 1  | "      6       | #1286 | 1  | "      6       |
| #1271 | 1  | "      7       | #1287 | 1  | "      7       |
| #1272 | 1  | "      8       | #1288 | 1  | "      8       |
| #1273 | 1  | "      9       | #1289 | 1  | "      9       |
| #1274 | 1  | "     10       | #1290 | 1  | "     10       |
| #1275 | 1  | "     11       | #1291 | 1  | "     11       |
| #1276 | 1  | "     12       | #1292 | 1  | "     12       |
| #1277 | 1  | "     13       | #1293 | 1  | "     13       |
| #1278 | 1  | "     14       | #1294 | 1  | "     14       |
| #1279 | 1  | "     15       | #1295 | 1  | "     15       |



## 宏、接口输出（#1100～#1135, #1300～#1395）:NC→PLC

通过将值代入变量编号#1100～#1135、#1300～#1395中，可送出接口输出信号。输出信号只有0和1两种。

通过将值代入变化编号#1132中，可一次送出#1100～#1131的全部输出编号。同样，通过将值代入变量编号#1133～#1135中，可送出#1300～#1331、#1332～#1363、#1364～#1395的输出信号。

可为了偏移#1100～#1135、#1300～#1395输出信号而进行写入，及读取输出信号状态。

这里所说的输出，是指从NC端的输出。

| 系统变量  | 点数 | 接口输出信号          | 系统变量  | 点数 | 接口输出信号          |
|-------|----|-----------------|-------|----|-----------------|
| #1100 | 1  | 寄存器 R172 的 bit0 | #1116 | 1  | 寄存器 R173 的 bit0 |
| #1101 | 1  | " 1             | #1117 | 1  | " 1             |
| #1102 | 1  | " 2             | #1118 | 1  | " 2             |
| #1103 | 1  | " 3             | #1119 | 1  | " 3             |
| #1104 | 1  | " 4             | #1120 | 1  | " 4             |
| #1105 | 1  | " 5             | #1121 | 1  | " 5             |
| #1106 | 1  | " 6             | #1122 | 1  | " 6             |
| #1107 | 1  | " 7             | #1123 | 1  | " 7             |
| #1108 | 1  | " 8             | #1124 | 1  | " 8             |
| #1109 | 1  | " 9             | #1125 | 1  | " 9             |
| #1110 | 1  | " 10            | #1126 | 1  | " 10            |
| #1111 | 1  | " 11            | #1127 | 1  | " 11            |
| #1112 | 1  | " 12            | #1128 | 1  | " 12            |
| #1113 | 1  | " 13            | #1129 | 1  | " 13            |
| #1114 | 1  | " 14            | #1130 | 1  | " 14            |
| #1115 | 1  | " 15            | #1131 | 1  | " 15            |

| 系统变量  | 点数 | 接口输出信号        |
|-------|----|---------------|
| #1132 | 32 | 寄存器 R172、R173 |
| #1133 | 32 | 寄存器 R174、R175 |
| #1134 | 32 | 寄存器 R176、R177 |
| #1135 | 32 | 寄存器 R178、R179 |

| 系统变量  | 点数 | 接口输出信号          | 系统变量  | 点数 | 接口输出信号          |
|-------|----|-----------------|-------|----|-----------------|
| #1300 | 1  | 寄存器 R174 的 bit0 | #1316 | 1  | 寄存器 R175 的 bit0 |
| #1301 | 1  | " 1             | #1317 | 1  | " 1             |
| #1302 | 1  | " 2             | #1318 | 1  | " 2             |
| #1303 | 1  | " 3             | #1319 | 1  | " 3             |
| #1304 | 1  | " 4             | #1320 | 1  | " 4             |
| #1305 | 1  | " 5             | #1321 | 1  | " 5             |
| #1306 | 1  | " 6             | #1322 | 1  | " 6             |
| #1307 | 1  | " 7             | #1323 | 1  | " 7             |
| #1308 | 1  | " 8             | #1324 | 1  | " 8             |
| #1309 | 1  | " 9             | #1325 | 1  | " 9             |
| #1310 | 1  | " 10            | #1326 | 1  | " 10            |
| #1311 | 1  | " 11            | #1327 | 1  | " 11            |
| #1312 | 1  | " 12            | #1328 | 1  | " 12            |
| #1313 | 1  | " 13            | #1329 | 1  | " 13            |
| #1314 | 1  | " 14            | #1330 | 1  | " 14            |
| #1315 | 1  | " 15            | #1331 | 1  | " 15            |

| 系统变量  | 点数 | 接口输出信号          | 系统变量  | 点数 | 接口输出信号          |
|-------|----|-----------------|-------|----|-----------------|
| #1332 | 1  | 寄存器 R176 的 bit0 | #1348 | 1  | 寄存器 R177 的 bit0 |
| #1333 | 1  | " 1             | #1349 | 1  | " 1             |
| #1334 | 1  | " 2             | #1350 | 1  | " 2             |
| #1335 | 1  | " 3             | #1351 | 1  | " 3             |
| #1336 | 1  | " 4             | #1352 | 1  | " 4             |
| #1337 | 1  | " 5             | #1353 | 1  | " 5             |
| #1338 | 1  | " 6             | #1354 | 1  | " 6             |
| #1339 | 1  | " 7             | #1355 | 1  | " 7             |
| #1340 | 1  | " 8             | #1356 | 1  | " 8             |
| #1341 | 1  | " 9             | #1357 | 1  | " 9             |
| #1342 | 1  | " 10            | #1358 | 1  | " 10            |
| #1343 | 1  | " 11            | #1359 | 1  | " 11            |
| #1344 | 1  | " 12            | #1360 | 1  | " 12            |
| #1345 | 1  | " 13            | #1361 | 1  | " 13            |
| #1346 | 1  | " 14            | #1362 | 1  | " 14            |
| #1347 | 1  | " 15            | #1363 | 1  | " 15            |

| 系统变量  | 点数 | 接口输出信号          | 系统变量  | 点数 | 接口输出信号          |
|-------|----|-----------------|-------|----|-----------------|
| #1364 | 1  | 寄存器 R178 的 bit0 | #1380 | 1  | 寄存器 R179 的 bit0 |
| #1365 | 1  | " 1             | #1381 | 1  | " 1             |
| #1366 | 1  | " 2             | #1382 | 1  | " 2             |
| #1367 | 1  | " 3             | #1383 | 1  | " 3             |
| #1368 | 1  | " 4             | #1384 | 1  | " 4             |
| #1369 | 1  | " 5             | #1385 | 1  | " 5             |
| #1370 | 1  | " 6             | #1386 | 1  | " 6             |
| #1371 | 1  | " 7             | #1387 | 1  | " 7             |
| #1372 | 1  | " 8             | #1388 | 1  | " 8             |
| #1373 | 1  | " 9             | #1389 | 1  | " 9             |
| #1374 | 1  | " 10            | #1390 | 1  | " 10            |
| #1375 | 1  | " 11            | #1391 | 1  | " 11            |
| #1376 | 1  | " 12            | #1392 | 1  | " 12            |
| #1377 | 1  | " 13            | #1393 | 1  | " 13            |
| #1378 | 1  | " 14            | #1394 | 1  | " 14            |
| #1379 | 1  | " 15            | #1395 | 1  | " 15            |

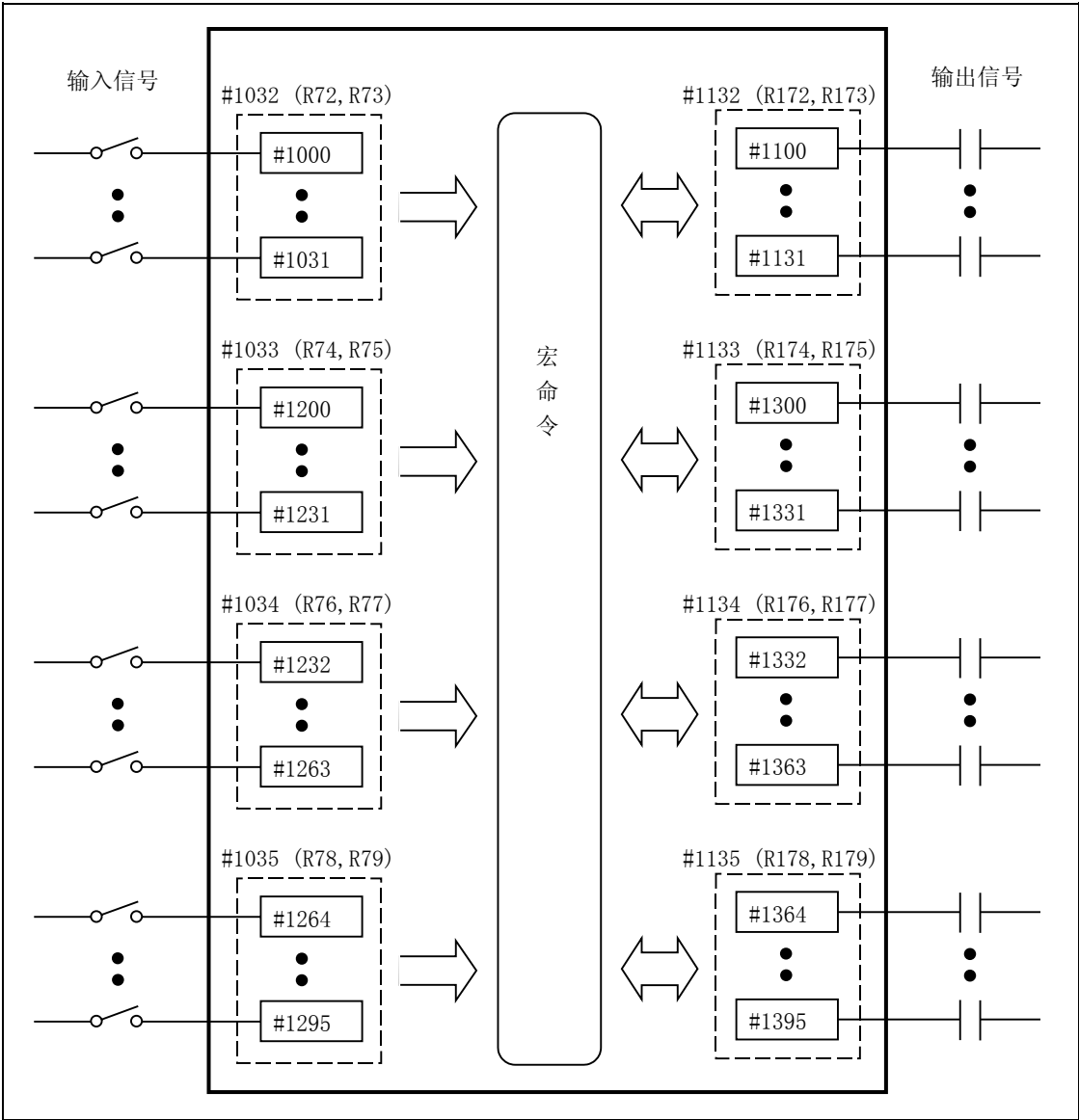
(注1) 系统变量#1100~#1135、#1300~#1395 的值，最后送出的值为或 0，被保存。  
(重新启动无法清除。)

(注2) 当在#1100~#1131、#1300~#1395 中代入 1 或 0 以外的值时，如下。

<空>看作为0。

<空>, 0以外看作为1。

但是，不足0.00000001为不确定。



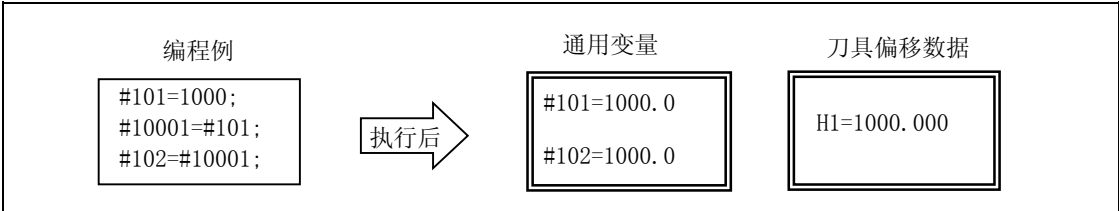


刀具偏移

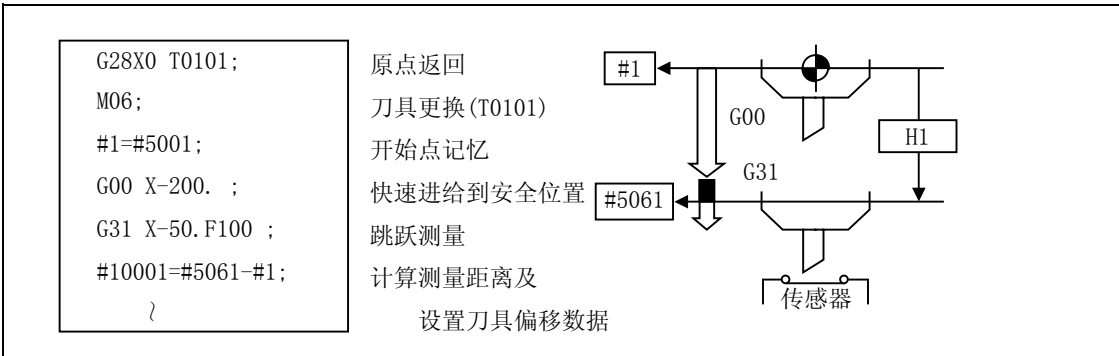
使用变量编号，可读取、设定刀具偏移数据。

| 变量编号范围          |                  |                  | 内 容          |
|-----------------|------------------|------------------|--------------|
|                 | #1120 TofVal = 0 | #1120 TofVal = 1 |              |
| #10001~#10000+n | #2001~#2000+n    | #2701~#2700+n    | X 形状修正量      |
| #11001~#11000+n | #2701~#2700+n    | #2001~#2000+n    | X 磨损修正量      |
| #12001~#12000+n |                  |                  | 附加轴<br>形状修正量 |
| #13001~#13000+n |                  |                  | 附加轴<br>磨损修正量 |
| #14001~#14000+n | #2101~#2100+n    | #2801~#2800+n    | Z 形状修正量      |
| #15001~#15000+n | #2801~#2800+n    | #2101~#2100+n    | Z 磨损修正量      |
| #16001~#16000+n | #2201~#2200+n    | #2901~#2900+n    | R 形状修正量      |
| #17001~#17000+n | #2901~#2900+n    | #2201~#2200+n    | R 磨损修正量      |
| #18001~#18000+n | #2301~#2300+n    |                  | 刀刃修正量        |

表中的n与刀具编号对应。n的最大值为刀具偏移组数。  
#10000编号组与#2000编号组的功能均相同。  
刀具偏移数据与其他变量相同，是带小数点的数据。  
如果编程为#10001=1000；，则刀具偏移数据中，设定为1000.000。  
对于E68，可使用第3轴/第4轴的任何一個，进行附加轴的刀具修正。  
通过参数“#1520 Tchg34”选择使用哪一个。  
可通过参数“#1122 TofVal”，切换与#2000编号组的形状/磨损修正量相对应的变量编号。



（例 1）刀具偏移数据测量例



（注 1）（例 1）中，未考虑跳跃用传感器的信号延迟。  
在这里，#5001 为 X 轴的开始点位置，#5061 为 X 轴的跳跃坐标，表示在执行 G31 时，输入跳跃信号的位置。



工件坐标系偏移

使用变量编号#5201~#532n，可读取工件坐标系偏移数据，或是将值代入。

(注) 可控制轴因规格而异。

变量编号的最后 1 位数字与控制轴编号对应。

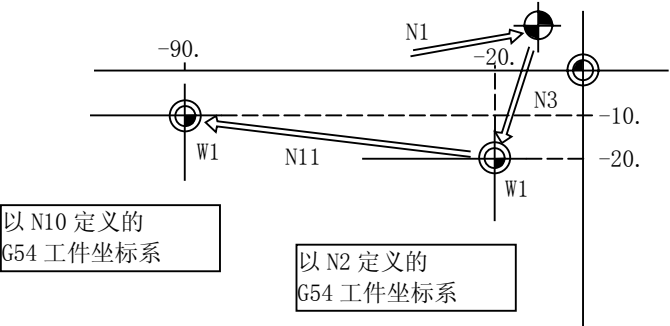
| 坐标名        | 轴编号 | 第 1 轴 | 第 2 轴 | 第 3 轴 | 备 注 |
|------------|-----|-------|-------|-------|-----|
| 外部<br>工件偏移 |     | #5201 | #5202 | #5203 |     |
| G54        |     | #5221 | #5222 | #5223 |     |
| G55        |     | #5241 | #5242 | #5243 |     |
| G56        |     | #5261 | #5262 | #5263 |     |
| G57        |     | #5281 | #5282 | #5283 |     |
| G58        |     | #5301 | #5302 | #5303 |     |
| G59        |     | #5321 | #5322 | #5323 |     |

(例 1)

```
N1 G28 X0 Z0 ;
N2 #5221=-20. #5222=-20. ;
N3 G00 G54 X0 Z0 ;

N10 #5221=-10. #5222=-90. ;
N11 G00 G54 X0Z0 ;

M02 ;
```

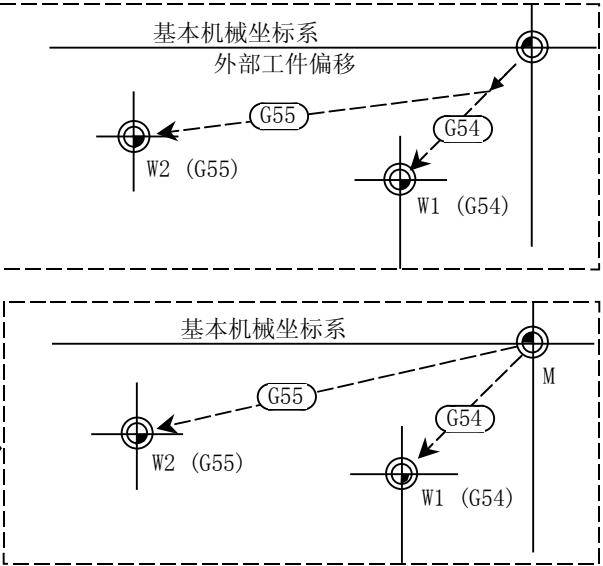


(例 2)

```
N100 #5221=#5221+#5201 ;
#5222=#5222+#5202 ;
#5241=#5241+#5201 ;
#5242=#5242+#5202 ;
#5201=0 #5202=0;
```

变更前的  
坐标系

变更后的  
坐标系



上图是工件坐标系位置不变，在各工件坐标（G54、G55）系偏移值的基础上，增加了外部工件偏移值时的例子。



警报（#3000）

通过使用变量编号#3000，可强制性的进入报警状态。

格式

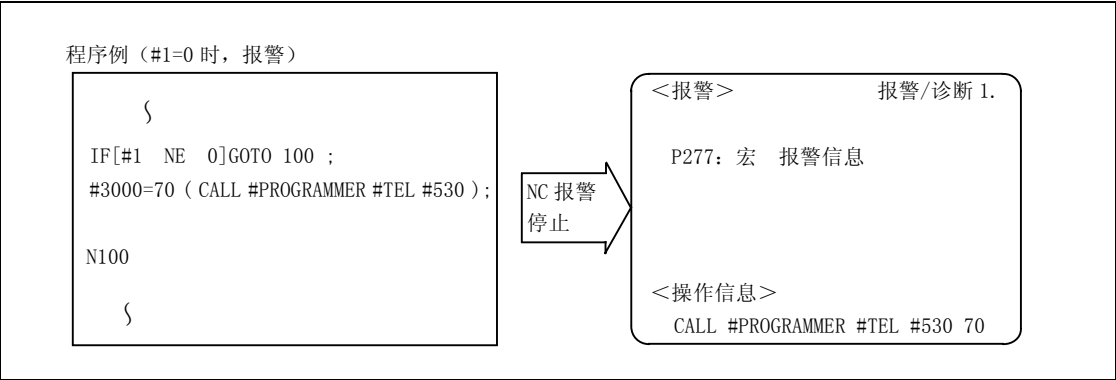
#3000= 70 （CALL #PROGRAMMER #TEL #530）；  
  
70 : 报警编号  
CALL #PROGRAMMER #TEL #530 : 报警信息

报警编号可从1到9999进行指令。

报警信息请控制在31个字符以内。

输出NC警报3（程序错误）信号。

在“报警诊断1.”画面的<报警>栏中，显示P277：宏 报警信息，并在<操作信息>中显示报警编号、报警信息70：（CALL#PROGRAMMER#TEL#530）。



（注 1）报警编号不显示 0。另外，无法显示超过 9999 的数字。

（注 2）右边将首个英文字母以后的内容看作为报警信息。因此，报警信息的首个字符无法指定为数字。推荐将报警信息括在（ ）括号中。

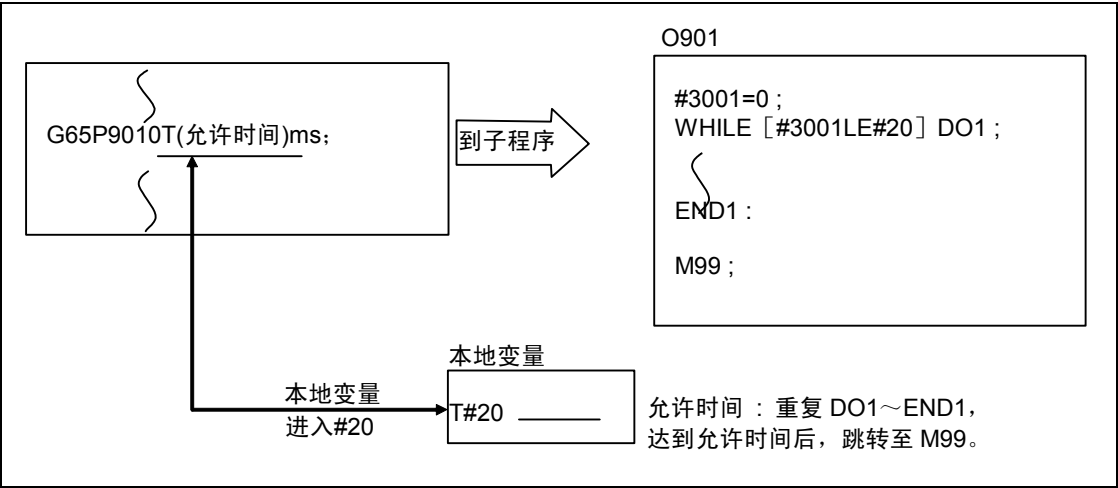


累计时间（#3001,#3002）

通过使用变量编号#3001和#3002，可以读取加电过程中和自动启动过程中的累计时间，并将数值代入变量。

| 种类   | 变量编号  | 单位 | 加电时的内容   | 内容初始化   | 计数条件    |
|------|-------|----|----------|---------|---------|
| 加电   | #3001 | ms | 与切断电源时相同 | 将数值代入变量 | 加电过程中随时 |
| 自动启动 | #3002 |    |          |         | 自动启动过程中 |

累计时间大约每 $2.44 \times 10^{11}$ ms(约7.7年)归零1次。



单个模块停止，辅助功能结束信号等待的抑制

将下列数值代入变量编号#3003，则其后的模块中，可以抑制单个模块的停止，无需等待辅助功能(M, S, T, B)的结束信号(FIN)，直接进入下一个模块。

| #3003 | 单个模块停止 | 辅助信号结束信号 |
|-------|--------|----------|
| 0     | 不抑制    | 等待       |
| 1     | 抑制     | 等待       |
| 2     | 不抑制    | 不等待      |
| 3     | 抑制     | 不等待      |

(注 1)#3003 通过重设变为 0。



进给保持、进给速度过载、G09 的启用和禁用

将下列数值代入变量编号#3004，则其后的模块中，可以启用或禁用进给保持、进给速度过载以及G09等功能。

| #3004<br>内容(值) | 比特 0 | 比特 1   | 比特 2   |
|----------------|------|--------|--------|
|                | 进给保持 | 进给速度过载 | G09 检查 |
| 0              | 启用   | 启用     | 启用     |
| 1              | 禁用   | 启用     | 启用     |
| 2              | 启用   | 禁用     | 启用     |
| 3              | 禁用   | 禁用     | 启用     |
| 4              | 启用   | 启用     | 禁用     |
| 5              | 禁用   | 启用     | 禁用     |
| 6              | 启用   | 禁用     | 禁用     |
| 7              | 禁用   | 禁用     | 禁用     |

- (注 1) #3004 通过重设变为 0。
- (注 2) 上述各位中，0 表示功能停止，1 表示禁用。
- (注 3) 在#3004 中将进给保持设定为无效的情况下，按下进给保持开关，将产生以下动作。
- 在螺纹切削过程中，模块将在螺纹切削已结束的下一个模块的终点处停止。
  - 攻牙循环的攻牙动作过程中，对 R 点的复位动作结束后，模块停止。
  - 除上述情况外，运行中的模块结束后，模块停止。



信息显示及停止

通过使用变量编号#3006，可以在运行后停止之前的模块，并在已指定信息显示数据的条件下，将该信息显示到操作员信息装置。

格式

|                           |
|---------------------------|
| #3006 = 1 ( TAKE FIVE ) ; |
| TAKE FIVE : 信息            |

信息字符数应在31字以内，并用( )括起。



镜像图像

通过读取变量编号#3007，可以获知各轴在此时间的镜像图像状态。

#3007的内容中，轴与各比特相对应，各比特的内容为：

|   |            |   |           |
|---|------------|---|-----------|
| { | 0 时的镜像图像无效 | } | 其涵义如左边所示。 |
|   | 1 时的镜像图像有效 |   |           |

#3007

| 比特    | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 第 n 轴 |    |    |    |    |    |    |   |   |   |   |   |   |   | 3 | 2 | 1 |



## G 模式

使用变量编号#4001~#4021可以读取指定到之前模块中的模式指令。

另外，在#4201~#4221中，可同样读取正在运行模块的模式。

| 变量编号       |          | 功 能                                                                       |
|------------|----------|---------------------------------------------------------------------------|
| 优先读取<br>模块 | 运行<br>模式 |                                                                           |
| #4001      | #4201    | 插补模式 G00: 0, G01: 1, G02: 2, G03: 3, G33: 33                              |
| #4002      | #4202    | 平面选择 G17: 17, G18: 18, G19: 19                                            |
| #4003      | #4203    | 绝对/增加 G90: 90, G91: 91                                                    |
| #4004      | #4204    | 障碍检查 G22: 22, G23: 23                                                     |
| #4005      | #4205    | 进给指定 G94: 94, G95: 95                                                     |
| #4006      | #4206    | 英制/公制 G20: 20, G21: 21                                                    |
| #4007      | #4207    | 刀尖 R 补偿 G40: 40, G41: 41, G42: 42, G46: 46                                |
| #4008      | #4208    | 无变量编号                                                                     |
| #4009      | #4209    | 固定循环 G80: 80, G70~G79: 70~79, G83~G85: 83~85, G83.2: 83.2, G87~G89: 87~89 |
| #4010      | #4210    | 复位水平 G98: 98, G99: 99                                                     |
| #4011      | #4211    |                                                                           |
| #4012      | #4212    | 工件坐标系 G54~G59: 54~59                                                      |
| #4013      | #4213    | 加减速 G61~G64: 61~64                                                        |
| #4014      | #4214    | 宏模式调用 G66: 66, G66.1: 66.1, G67: 67                                       |
| #4015      | #4215    |                                                                           |
| #4016      | #4216    | 无变量编号                                                                     |
| #4017      | #4217    | 圆周速度一定 G96: 96, G97: 97                                                   |
| #4018      | #4218    |                                                                           |
| #4019      | #4219    |                                                                           |
| #4020      | #4220    |                                                                           |
| #4021      | #4221    |                                                                           |

(例)

```
G28 X0 Z0;
G00 X150. Z200;
G65 P300 G02 W-30. K-15. F1000;
M02;
O300
#1 = #4001; = →组 01 G 模式（优先读取）#1 = 2.0
= #4201; = →组 01 G 模式（运行中）#2 = 0.0
G#1 W#24;
M99;
%
```



其他模式

使用变量编号#4101~4120可以读取指定到之前模块中的模式指令。  
另外，在#4301~#4320中，可同样读取正在运行模块的模式。

| 变量编号  |       | 模式信息   |
|-------|-------|--------|
| 优先读取  | 运行    |        |
| #4101 | #4301 |        |
| #4102 | #4302 |        |
| #4103 | #4303 |        |
| #4104 | #4304 |        |
| #4105 | #4305 |        |
| #4106 | #4306 |        |
| #4107 | #4307 |        |
| #4108 | #4308 |        |
| #4109 | #4309 | 进给速度 F |
| #4110 | #4310 |        |

| 变量编号  |       | 模式信息   |
|-------|-------|--------|
| 优先读取  | 运行    |        |
| #4111 | #4311 |        |
| #4112 | #4312 |        |
| #4113 | #4313 | 辅助功能 M |
| #4114 | #4314 | 序列编号 N |
| #4115 | #4315 | 程序编号 0 |
| #4116 | #4316 |        |
| #4117 | #4317 |        |
| #4118 | #4318 |        |
| #4119 | #4319 | 主轴功能 S |
| #4120 | #4320 | 刀具功能 T |

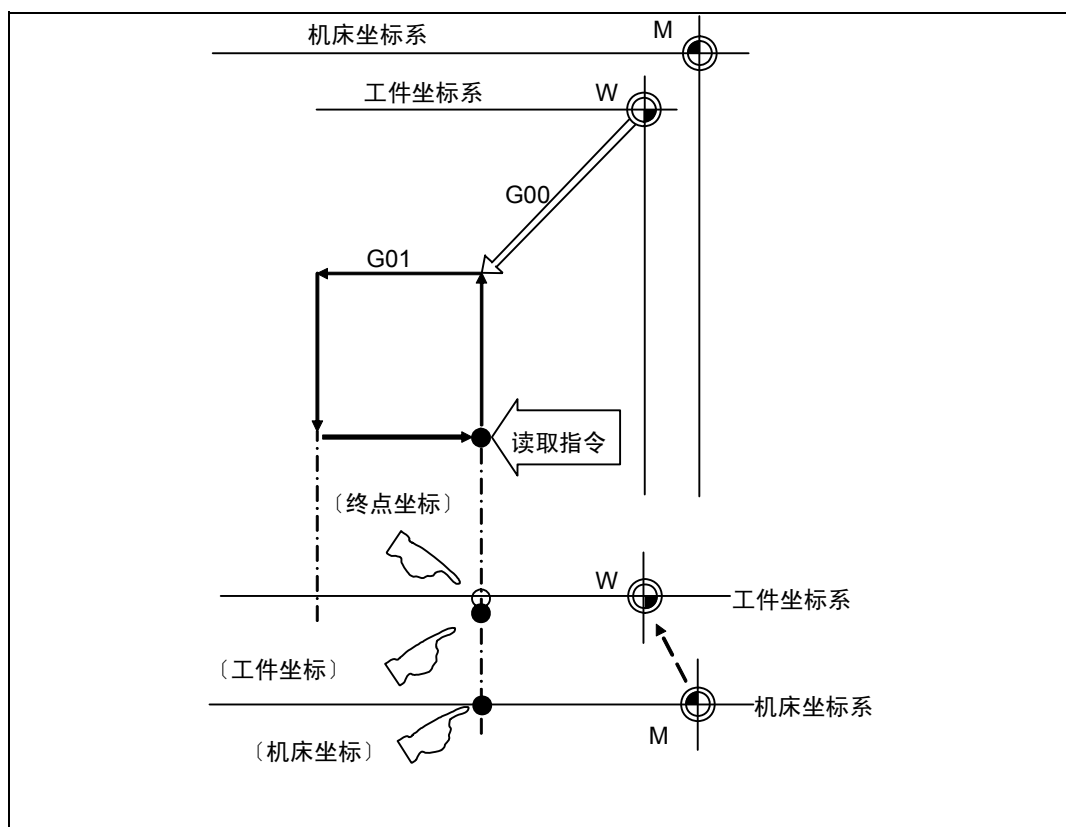


位置信息

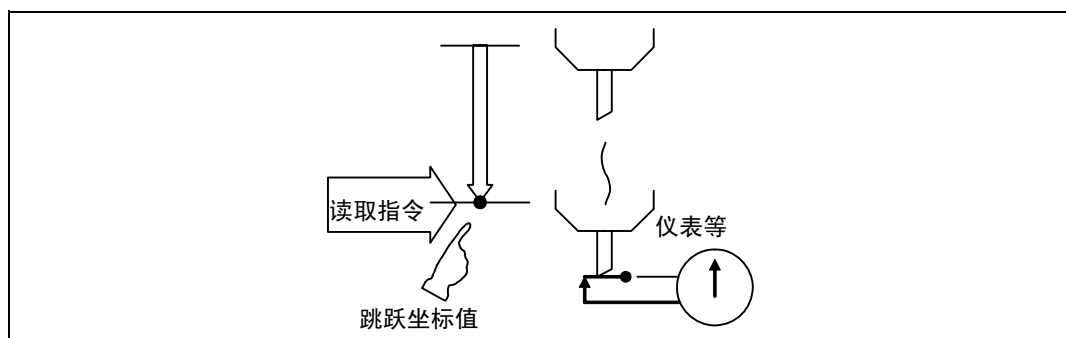
使用变量编号#5001~#5104，可以读取之前模块的终点坐标系、机床坐标值、工作坐标值、跳跃坐标值、伺服偏差量等。

| 位置信息<br>轴编号    | 之前模块的<br>终点坐标 | 机床坐标  | 工件坐标  | 跳跃坐标  | 伺服偏差量 |
|----------------|---------------|-------|-------|-------|-------|
| 1              | #5001         | #5021 | #5041 | #5061 | #5101 |
| 2              | #5002         | #5022 | #5042 | #5062 | #5102 |
| 3              | #5003         | #5023 | #5043 | #5063 | #5103 |
| 备注<br>(移动中的读取) | 可             | 不可    | 不可    | 可     | 可     |

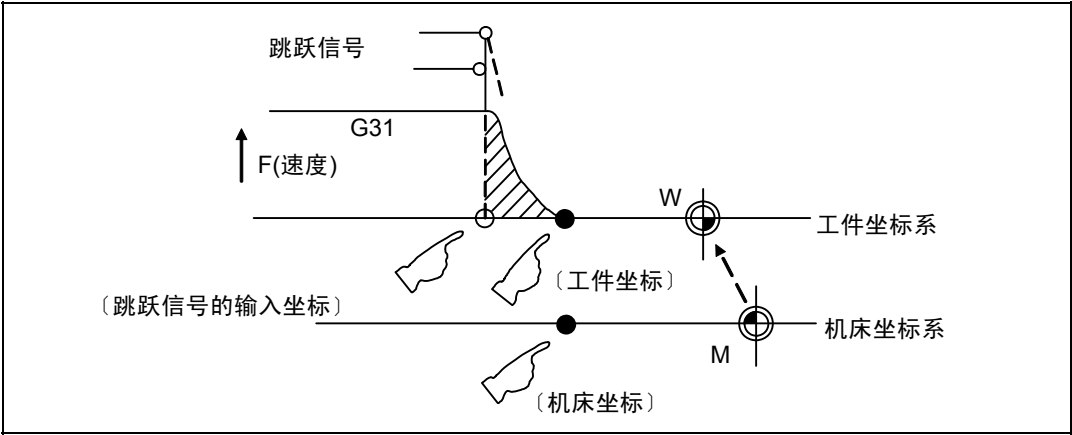
(注) 可控制的轴数根据规格的不同而不同。  
变量编号的下 1 位数字对应控制轴编号。



- (1) 终点坐标、跳跃坐标是在工件坐标系中的位置。
- (2) 终点坐标、跳跃坐标、伺服偏差量可以在移动中进行读取；机床坐标、工件坐标应确认停止移动后再进行读取。
- (3) 跳跃坐标表示G31模块中跳跃信号变为ON的位置。跳跃信号未变成ON时，表示其终点位置。  
(详情请参照刀具长度的测定。)



- (4) 终点位置表示不考虑刀具偏移量等因素的刀具顶端位置；机床坐标、工件坐标、跳跃坐标则分别表示考虑刀具偏移量的刀具标准点位置。



● 请在确认停止后读取。  
○ 在移动过程中也可进行读取。

跳跃信号的输入坐标值是在工件坐标系中的位置。#5061~#5064的坐标值存储机床移动过程中跳跃输入信号进入的瞬间，因此其后的任意时间均可读取。  
详情请参照“跳跃功能”。



变量名与设定的引用

常用变量的#500~#519可以附加任意的名称(变量名)。变量名应以字母开头，并限制在7个英数以内。变量名请勿使用“#”。如使用“#”，则运行时将发生错误。

格式

|                             |                 |
|-----------------------------|-----------------|
| SETVNn [ NAME1,NAME2,…… ] ; |                 |
| n                           | : 命名变量的开头编号     |
| NAME1                       | : #n 的名称(变量名)   |
| NAME2                       | : #n+1 的名称(变量名) |

各变量名之间用“,”分隔开。

详细说明

- (1) 设定1次的变量名切断电源后不会消失。
- (2) 通过变量名引用程序中的变量。但此时的变量应用[ ]括起来。  
(例1) G01X [#POINT1] ;
- (3) 设定值显示装置的画面中将显示变量编号、数据以及变量名称。

(例2)

程序.....SETVN500 [A234567,DIST,TOOL25] ;

|          |                               |         |
|----------|-------------------------------|---------|
| [ 常用变量 ] |                               |         |
| #500     | -12345.678                    | A234567 |
| #501     | 5670.000                      | DIST    |
| #502     | -156.500                      | TOOL25  |
| ~~~~~    |                               |         |
| #518     | 10.000                        | NUMBER  |
| 常用变量     | #(502)数据 (-156.5) 名称 (TOOL25) |         |

(注) 变量名的开头请勿使用运算命令中使用的 NC 专用字符(SIN, COS 等)。



外部工件坐标系偏移

使用变量编号#2501～#2601，可以读取外部工件坐标系的偏移。  
另外，通过将数值代入该变量编号，可以对外部工件坐标系的偏移进行变更。

| 轴编号 | 外部工件坐标系偏移 |
|-----|-----------|
| 1   | #2501     |
| 2   | #2601     |



工件加工数

使用变量编号#3901～#3902，可以读取工件加工数。  
另外，通过将数值代入该变量编号，可以对工件加工数进行变更。

| 种类    | 变量编号  | 数据设定范围   |
|-------|-------|----------|
| 工件加工数 | #3901 | 0～999999 |
| 工件最大值 | #3902 |          |

(注) 工件加工数必须代入正确的值。



刀具寿命管理

(1) 变量编号的定义

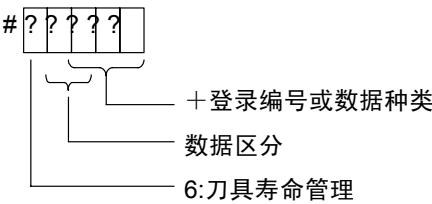
(a) 组编号的指定

#60000

通过将数值代入此变量编号，对#60001～#64700中读取的刀具寿命管理数据的组编号进行指定。  
未指定组编号时，将读取最先登录组的数据。重设前有效。

(b) 刀具寿命管理的系统变量编号(读取)

#60001～#64700



(c) 数据区分的细分

| 数据区分 | M 系列     | L 系列    | 备注     |
|------|----------|---------|--------|
| 00   | 控制用      | 控制用     | 参照数据种类 |
| 05   | 组编号      | 组编号     | 参照登录编号 |
| 10   | 刀具编号     | 刀具编号    | 参照登录编号 |
| 15   | 刀具数据图表   | 方式      | 参照登录编号 |
| 20   | 刀具状态     | 状态      | 参照登录编号 |
| 25   | 寿命数据     | 寿命时间、次数 | 参照登录编号 |
| 30   | 使用数据     | 使用时间、次数 | 参照登录编号 |
| 35   | 刀具长度补偿数据 | -       | 参照登录编号 |
| 40   | 刀具半径补偿数据 | -       | 参照登录编号 |
| 45   | 辅助数据     | -       | 参照登录编号 |

组编号、L系列的方式以及寿命等各数据均为组通用数据。

(d) 登录编号

|      |       |
|------|-------|
| M 系列 | 1～200 |
| L 系列 | 1～16  |

(e) 数据种类

| 种类 | M 系列         | L 系列         | 备注 |
|----|--------------|--------------|----|
| 1  | 登录刀具个数       | 登录刀具个数       |    |
| 2  | 寿命当前值        | 寿命当前值        |    |
| 3  | 刀具选择编号       | 刀具选择编号       |    |
| 4  | 登录刀具的剩余个数    | 登录刀具的剩余个数    |    |
| 5  | 运行中信号        | 运行中信号        |    |
| 6  | 切削时间累计值(min) | 切削时间累计值(min) |    |
| 7  | 寿命结束信号       | 寿命结束信号       |    |
| 8  | 寿命预告信号       | 寿命预告信号       |    |

| 变量编号  | 项目               | 种类                      | 内容                                                                                                | 数据范围                          |
|-------|------------------|-------------------------|---------------------------------------------------------------------------------------------------|-------------------------------|
| 60001 | 登录刀具个数           | 系统通用                    | 各组的登录刀具合计                                                                                         | 0~80                          |
| 60002 | 寿命当前值            | 每组<br>(指定组编号<br>#60000) | 使用中刀具的使用时间/次数<br><br>使用中刀具的使用数据(使用多个补偿编号的刀具, 其使用数据为各补偿编号的使用数据合计)                                  | 0~<br>999999min<br>0~999999 次 |
| 60003 | 刀具选择编号           |                         | 使用中刀具的登录编号<br><br>指定组中所选刀具(没有选择刀具时, 表示 ST:1 的初始刀具; 没有 ST:1 时, 表示 ST:0 的初始刀具; 所有寿命时, 表示最后的刀具)的登录编号 | 0~16                          |
| 60004 | 登录刀具的剩余个数        |                         | 该组的“可使用”刀具合计个数<br><br>指定组的登录刀具中, ST 为 0: 未使用的刀具个数                                                 | 0~16                          |
| 60005 | 运行中信号            |                         | 正在运行的程序中, 使用该组时为“1”。<br><br>指定组的刀具在选择过程中为“1”。                                                     | 0/1                           |
| 60006 | 切削时间累计值<br>(min) |                         | 表示正在运行的程序中使用该组的时间。                                                                                |                               |
| 60007 | 寿命结束信号           |                         | 该组的刀具寿命全部耗尽时为“1”<br><br>指定组中的登录刀具全部达到寿命时为“1”                                                      | 0/1                           |
| 60008 | 寿命预告信号           |                         | 该组通过下次指定选择新刀具时为“1”<br><br>指定组中的登录刀具,<br>ST 为 0: 存在未使用的刀具<br>ST 为 1: 没有使用中的刀具                      | 0/1                           |

| 变量编号          | 项目      | 种类                                                                                 | 内容                                                                           | 数据范围                          |
|---------------|---------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------|-------------------------------|
| 60500<br>+*** | 组编号     | 按组、<br>登录编号分类<br>(指定组编号<br>#60000、登录编号<br>***)<br><br>另外，<br>组编号、方式以及寿<br>命均为组通用数据 | 该组的编号                                                                        | 1~9999                        |
| 61000<br>+*** | 刀具编号    |                                                                                    | 刀具的刀具编号与补偿编号<br><br>刀具 No. + 补偿 No.<br>(刀具 No.=22, 补偿 No.=01 时<br>2201=899H) | 0~9999                        |
| 61500<br>+*** | 方式      |                                                                                    | 按时间或次数对该组进行寿命管理<br><br>0: 时间, 1: 次数                                          | 0/1                           |
| 62000<br>+*** | 状态      |                                                                                    | 刀具的使用状况<br><br>0: 未使用刀具<br>1: 使用中刀具<br>2: 正常寿命刀具<br>3: 刀具跳跃刀具                | 0~3                           |
| 62500<br>+*** | 寿命时间、次数 |                                                                                    | 该组的刀具寿命值                                                                     | 0~<br>999999min<br>0~999999 次 |
| 63000<br>+*** | 使用时间、次数 |                                                                                    |                                                                              | 0~<br>999999min<br>0~999999 次 |
| 63500<br>+*** | -       |                                                                                    |                                                                              |                               |
| 64000<br>+*** | -       |                                                                                    |                                                                              |                               |
| 64500<br>+*** | -       |                                                                                    |                                                                              |                               |



刀具寿命管理的程序例

(1) 普通指令

```
#101 = #60001; 读取登录刀具个数。
#102 = #60002; 读取寿命当前值。
#103 = #60003; 读取刀具选择编号。
#60000 = 10; 指定要读取的寿命数据的组编号。
#104 = #60004; 读取组 10 的登录刀具剩余个数。
#105 = #60005; 读取组 10 的运行中信号。
#111 = #61001; 读取组 10, #1 的刀具编号。
#112 = #62001; 读取组 10, #1 的状态。
#113 = #61002; 读取组 10, #2 的刀具编号。
%
```

程序编号的  
指定在重设  
前有效

(2) 未指定组编号时

```
#104 = #60004; 读取最先登录组的登录刀具剩余个数。
#111 = #61001; 读取最先登录组的#1的刀具编号。
%
```

(3) 指定未登录组的编号时(组9999不存在)

```
#60000 = 9999; 指定组编号。
#104 = #60004; #104 = -1。
```

(4) 指定未使用的登录编号时(组10的刀具假设为15个)

```
#60000 = 10; 指定组编号。
#111 = #61016; #111 = -1。
```

(5) 指定规格以外的登录编号时

```
#60000 = 10;
#111 = #61017; 程序错误(P241)
```

|                                  |       |                                |                    |
|----------------------------------|-------|--------------------------------|--------------------|
| (6) 指定组编号后，通过G10指定进行刀具寿命管理数据的登录时 |       |                                |                    |
| #60000 = 10;                     | ..... | 指定组编号。                         | } 登录组10的<br>寿命数据   |
| G10 L3;                          | ..... | 开始登录用于寿命管理的数据                  |                    |
| P10 LLn NNn;                     | ..... | 10表示组编号，Ln表示每个刀具的寿命，<br>Nn表示方式 |                    |
| TTn;                             | ..... | Tn表示刀具编号                       |                    |
| :                                |       |                                |                    |
| G11;                             | ..... | 通过G10指令登录组10的数据。               | } 登录组10以外<br>的寿命数据 |
| #111 = #61001;                   | ..... | 读取组10，#1的刀具编号。                 |                    |
| G10 L3;                          | ..... | 开始登录用于寿命管理的数据                  |                    |
| P10 LLn NNn;                     | ..... | 1表示组编号，Ln表示每个刀具的寿命，<br>Nn表示方式  |                    |
| TTn;                             | ..... | Tn表示刀具编号                       |                    |
| :                                |       |                                |                    |
| G11;                             | ..... | 通过G10指令登录寿命数据。                 |                    |
|                                  |       | (已登录的数据将被删除。)                  |                    |
| #111 = #61001;                   | ..... | 组10不存在。#111 = -1。              |                    |



刀具寿命管理的注意事项

- (1) 不指定组编号，指定刀具寿命管理的系统变量时，将对已登录的数据中，登录顺序在前的组数据进行读取。
- (2) 指定未登录的组编号，并指定刀具寿命管理的系统变量，数据读取为-1。
- (3) 指定未使用的登录编号的刀具寿命管理系统变量时，数据读取为-1。
- (4) 组编号从指定后至NC重设前有效。
- (5) 刀具寿命管理 I 规格的条件下，不存在刀具寿命管理的系统变量规格。如指定，将引起程序错误(P241)。



参数读取

通过系统变量可以读取参数数据。

(注) 仅限部分机型可用。

| 变量编号    | 用 途        |
|---------|------------|
| #100000 | 参数#编号指定    |
| #100001 | 系统备用       |
| #100002 | 轴编号/主轴编号指定 |
| #100010 | 参数值读取      |

参数值的读取使用这4个系统变量，按照如下4个模块的步骤执行。

#100000 = 1001; ..... 指定参数的#编号。

#100001 = 0;

#100002 = 1; ..... 指定轴编号/主轴编号。

#100 = #100010; ..... 读取参数值。

(1) 参数#编号指定(#100000)

将参数#编号代入此系统变量，指定要读取的参数。

不执行此项指定直接进行读取时，读取到的数值与指定参数#编号最小值(#1)时相同。指定1次以后，重新指定参数#编号或复位之前将保持设定值。

指定不存在的参数#编号时，将发生程序错误。

(2) 系统备用 (#100001)

应始终设定为“0”。

但PLC轴应设定为“10”。

(3) 轴/主轴编号指定(#100002)

(a)轴/主轴编号指定用系统变量

将Index值代入此系统变量，指定要读取的参数的轴编号/主轴编号。对既没有轴分类也没有主轴分类的参数进行读取时，本指定将被忽略。

指定不存在的轴/主轴编号时，将发生程序错误(P39)。

(b) Index值

| Indes 值 | 轴参数   | 主轴参数   |
|---------|-------|--------|
| 1       | 第 1 轴 | 第 1 主轴 |
| 2       | 第 2 轴 | 第 2 主轴 |
| 3       | 第 3 轴 | -      |

(4) 参数读取(#100010)

通过该系统变量，读取指定参数的数据。

根据参数类型读取的数据如下：

| 类型 | 读取数据                 |
|----|----------------------|
| 数值 | 输出参数画面中显示的数值。        |
| 文本 | 将 ASCII 码换算为 10 进制数。 |



参数读取的程序例

- (1) 读取参数[#1002 axisno轴数]时
- #100000 = 1002; ..... 指定[#1002]。  
#100001 = 0;  
#101 = #100010; ..... 读取轴数。  
#100001 = 10; ..... 指定[PLC 轴]。  
#110 = #100010; ..... 读取 PLC 轴的轴数。
- (2) 读取轴参数[#2037 G53ofs #1参考点]时  
[条件]
- |              | <第1轴>   | <第2轴>   |
|--------------|---------|---------|
| #2037 G53ofs | 100.000 | 200.000 |
- #100002 = 1; ..... 指定[第 1 轴]。  
#100000 = 2037; ..... 指定[#2037]。  
#101 = #100010; ..... 读取第 1 轴的[#1 参考点]。  
(#101=100.000。)  
#100002 = 2; ..... 指定[第 2 轴]。  
#102 = #100010; ..... 读取第 2 轴的[#1 参考点]。  
(#102=200.000。)
- (3) 读取每个轴/主轴的各参数时
- #100002 = 1; ..... 指定[第 1 主轴]。  
#100000 = 3001; ..... 指定[#3001]。  
#101 = #100010; ..... 读取第 1 主轴的[#3001 slimt1 界限转速齿轮 00]。  
#100000 = 3002; ..... 指定[#3002]。  
#102 = #100010; ..... 读取第 1 主轴的[#3002 slimt2 界限转速齿轮 01]。



参数读取的宏程序使用例

<宏规格>

G341 A\_. Q\_. ;  
A\_. .... 保存常用变量                      指定保存读取数据的常用变量编号。  
Q\_. .... 参数#编号指定                      轴/主轴参数中，通过保留小数点后1位的数值指定轴/主轴编号。

<宏内容>

|                                |                |
|--------------------------------|----------------|
| #100000 = FIX[#17] ;           | ..... 指定参数#编号  |
| #100002 = FIX[#17*10] MOD 10 ; | ..... 指定轴/主轴编号 |
| #[#1] = #100010 ;              | ..... 读取参数数据。  |
| M99 ;                          |                |



参数读取的注意事项

- (1) 主/主轴数表示规格的最大数。
- (2) 设定和显示的英制/公制切换功能对此项读取数据也有效。



PLC 数据读取

通过系统变量可以读取PLC数据。

(注) 仅限部分机型可用。

(注2) 可读取的设备存在限制。

| 变量编号    | 用 途       |
|---------|-----------|
| #100100 | 指定设备种类    |
| #100101 | 指定设备编号    |
| #100102 | 指定读取字节数   |
| #100103 | 指定读取比特    |
| #100110 | 读取 PLC 数据 |

PLC数据的读取使用这5个系统变量，按照如下5个模块的步骤执行。

- #100100 = 1; ..... 指定设备的种类。
- #100101 = 0; ..... 指定设备编号。
- #100102 = 1; ..... 指定字节数。
- #100103 = 2; ..... 指定比特。(仅限读取字符设备的比特时有效)
- #100=#100110; ..... 读取 PLC 数据。

(1)设备指定(#100100)

(a) 设备指定用系统变量

将设备指定值代入此系统变量，指定要读取的设备的种类。  
不执行此项指定直接进行读取时，读取到的数值与指定设备指定值最小值(0：设备M)时相同。指定1次以后，在重新指定设备或复位之前将保持设定值。  
指定不存在的参数#编号时，将发生程序错误(P39)。

(b) 设备指定值

| 设备<br>指定值 | 设备  | 单位 | 设备编号     | 设备<br>指定值 | 设备 | 单位 | 设备编号            |
|-----------|-----|----|----------|-----------|----|----|-----------------|
| 0         | M   | 比特 | M0~M8191 | 6         | R  | 字符 | R0~R8191<br>(注) |
| 1         | D   | 字符 | D0~D1023 | 7         | T  | 比特 | T0~T255         |
| 2         | C   | 比特 | C0~C127  | 9         | SM | 比特 | SM0 ~<br>SM127  |
| 4         | X ※ | 比特 | X0~X4BF  | 10        | F  | 比特 | F0~F127         |
| 5         | Y ※ | 比特 | Y0~Y53F  | 13        | L  | 比特 | L0~L255         |

1个设备编号对应的数据量单位，“字符”为16比特，“比特”为1比特。  
※表示以16进制数表示设备编号的设备。  
(注) 由于E60/E68使用系统1，因此不使用系统2对应的设备。

(2)设备编号指定(#100101)

将设备编号代入此系统变量，指定要读取的设备。  
用16进制数表示的设备应换算为10进制数进行指定。  
不执行此项指定直接进行读取时，读取到的数值与指定设备编号最小值(0)时相同。指定1次以后，在重新指定设备编号或复位之前将保持设定值。  
指定不存在的设备编号时，将发生程序错误(P39)。

## (3) 字节数指定(#100102)

## (a) 字节数指定用系统变量

将字节数指定值代入此系统变量，指定要读取大小。

不执行此项指定直接进行读取时，读取到的数值与指定字节数指定值的最小值(0：设备M)时相同。

指定1次以后，在重新指定字节数或复位之前将保持设定值。

指定规格中没有的字节数时，将发生程序错误(P39)。

## (b) 字节数指定值

| 字节数<br>指定值 | 读取数据 |    |                        | 动作                 |                    |
|------------|------|----|------------------------|--------------------|--------------------|
|            | 大小   | 符号 | 范围                     | 字符设备               | 比特设备               |
| 0          | 1 比特 | -  | 0~1                    | 读取比特指定值的比特。        | 读取指定设备编号的比特。       |
| 1          | 1 字节 | 无  | 0~255                  | 读取下位 1 字节。         | 从指定设备编号开始读取 8 比特。  |
| 101        |      | 有  | -128~127               |                    |                    |
| 2          | 2 字节 | 无  | 0~65535                | 读取 2 字节。           | 从指定设备编号开始读取 16 比特。 |
| 102        |      | 有  | -32768~32767           |                    |                    |
| 4          | 4 字节 | 无  | 0~4294967295           | 读取指定设备(L)和下一设备(H)。 | 从指定设备编号开始读取 32 比特。 |
| 104        |      | 有  | -2147483648~2147483647 |                    |                    |

0~4指定为无符号，101~104指定为有符号。

## (4) 比特指定(#100103)

## (a) 比特指定用系统变量

将比特指定值代入此系统变量，指定读取比特。

本指定仅在16比特设备的比特读取时有效，其他情况下无效。

不执行此项指定直接进行读取时，读取到的数值与指定比特指定值的最小值(0：设备M)时相同。

指定1次以后，在重新指定比特或复位之前将保持设定值。

指定规格中没有的比特时，将发生程序错误(P39)。

## (b) 比特指定值

| 比特指定值 | 读取比特  |
|-------|-------|
| 0     | 比特 0  |
| 1     | 比特 1  |
| :     | :     |
| 15    | 比特 15 |

## (5) PLC数据读取(#100110)

通过该系统变量，读取指定参数的数据。

读取数据范围参照字节数指定表。



## PLC 数据读取的程序例

## (1) 读取比特设备时

#100100 = 0; ..... 指定[设备 M]。  
 #100101 = 0; ..... 指定[设备编号 0]。  
 #100102 = 0; ..... 指定[比特]。  
 #100 = #100110; ..... 读取 M0(1 比特)。  
 #100102 = 1; ..... 指定[1 字节]。  
 #101 = #100110; ..... 读取 M0~M7(8 比特)。  
 (M7~M0 为 0001 0010 时, #102=18(0x12)。)  
 #100102 = 102; ..... 指定[有符号 2 字节]。  
 #102 = #100110; ..... 读取 M0~M15(16 比特)。  
 (M15~M0 为 1111 1110 1101 1100 时, #102=-292(0xFEDC)。)  
 #100102 = 4; ..... 指定[4 字节]。  
 #104 = #100110; ..... 读取 M0~M31(32 比特)。  
 (M31~M0 为 0001 0010 0011 0100 0101 0110 0111 1000 时,  
 #104=305419896(0x12345678)。)

## (2) 读取字符设备时

#100100 = 1; ..... 指定[设备 D]。  
 #100101 = 0; ..... 指定[设备编号 0]。  
 #100102 = 0; ..... 指定[比特]。  
 #100103 = 1; ..... 指定[比特 1]。  
 #100 = #100110; ..... 读取 D0 的比特 1。  
 (D0=0x0102 时, #101=1。)  
 #100102 = 1; ..... 指定[1 字节]。  
 #101 = #100110; ..... 读取 D0 的下位 1 字节。  
 (D0=0x0102 时, #101=2。)  
 #100102 = 2; ..... 指定[2 字节]。  
 #102 = #100110; ..... 读取 D0。(D0=0x0102 时, #102=258。)  
 #100102 = 104; ..... 指定[有符号 4 字节]。  
 #104 = #100110; ..... 读取 D0, D1。  
 (D0=0xFFFFE, D1=0xFFFF 时, #104=-2。)



PLC 数据读取的宏程序使用例

<宏规格>

G340 F\_\_ A\_\_ Q\_\_ H\_\_ ;

|     |       |        |    |       |                    |
|-----|-------|--------|----|-------|--------------------|
| F__ | ..... | 字节数指定  | F0 | ..... | 比特指定               |
|     |       |        | F1 | ..... | 指定1字节              |
|     |       |        | F2 | ..... | 指定2字节              |
| A__ | ..... | 设备指定   | A0 | ..... | 指定M                |
|     |       |        | A1 | ..... | 指定D                |
|     |       |        | A2 | ..... | 指定C                |
|     |       |        | A4 | ..... | 指定X                |
|     |       |        | A5 | ..... | 指定Y                |
|     |       |        | A6 | ..... | 指定R                |
|     |       |        | A7 | ..... | 指定T                |
| Q__ | ..... | 设备编号指定 |    |       | 比特条件下，在小数点后2位指定比特。 |
|     |       |        |    |       | 字节条件下，小数点后不设定。     |
| H__ | ..... | 保存常用变量 |    |       | 指定保存已读取数据的常用变量编号。  |

<宏内容>

|                                  |       |          |
|----------------------------------|-------|----------|
| #100100 = #1 ;                   | ..... | 指定设备     |
| #100101 = FIX[#17] ;             | ..... | 指定设备编号   |
| #100102 = #9 ;                   | ..... | 指定字节数    |
| #100103 = FIX[#17*100] MOD 100 ; | ..... | 指定比特     |
| #[#11] = #100110 ;               | ..... | 读取PLC数据。 |
| M99 ;                            |       |          |



PLC 数据读取的注意事项

- (1) PLC数据的读取与梯形图的运行不同步，因此，并不一定是程序运行时的数据。读取正在产生变化的设备时请注意。
- (2) 如果通过指定设备编号和字节数，读取不存在的设备时，则只有不存在的部分被读取为0。



### 时刻读取变量

通过扩展用户宏之中时刻的系统变量，可以完成以下操作。

- (1) 通过增加时刻信息的系统变量#3011、#3012，对当前日期(#3011)和当前时间(#3012)进行读取和写入。
- (2) 通过增加参数#1273/bit1，完成系统变量#3002自动启动过程中的累计时间的单位切换(毫秒单位/小时单位)

| 变量编号  | 内容                                                                                                                                                                          |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| #3001 | 可以读取加电过程中的累计时间以及代入数值。<br>单位为毫秒。                                                                                                                                             |
| #3002 | 可以读取自动启动过程中的累计时间以及代入数值。<br>可通过参数#1273/bit1 在毫秒和小时单位之间切换。                                                                                                                    |
| #3011 | 可读取或写入当前的日期。<br>YYYY 年 MM 月 DD 日读取为数值 YYYYMMDD。<br>写入数值 YYYYMMDD，即设定为 YY 年 MM 月 DD 日(年份用后 2 位表示)。<br>设定年月日时的指定范围 年 (YYYY) : 2000~2099<br>月 (MM) : 1~12<br>日 (DD) : 1~当月最大天数 |
| #3012 | 可读取或写入当前的时间。<br>HH 时 MM 分 SS 秒读取为数值 HHMMSS。<br>写入数值 HHMMSS，即设定为 HH 时 MM 分 SS 秒。<br>时间设定时的指定范围 时(HH) : 0~23(24 小时制)<br>分 (MM) : 0~59<br>秒 (SS) : 0~59                        |

- (3) 累计时间大约每 $2.44 \times 10^{11}$ ms(约7.7年)归零1次。
- (4) 在累计时间的设定中，如指定负值或超过244335917226毫秒(在#3002小时指定中，为67871.08811851 小时)，将发生程序错误(P35)。
- (5) 设定日期和时间时，如指定超出指令范围的数值，将发生程序错误(P35)。
- (6) 在日期和时间的设定中，月/日/时/分/秒必须以2位的数字进行设定。  
数字为1位时，设定时必须补0。(2001年2月14日 ⇒ #3011=20010214 ; 等)



## 时刻读取变量的使用例(#3011、#3012)

(例 1) 将当前日期(2001 年 2 月 14 日)读取到常用变量#100

#100 = #3001 ; (20010214 进入#100。)

(例 2) 将当前时间写入系统变量#3012

#3012 = 181306; (指令值的累计时间#2: 时间被设定为 18:13:06。)

(例 3) 通过下述程序示例, 可获知加工开始/结束时间(年/月/日/时/分/秒)。

```
#100=#3011; ⇒ 加工开始 年/月/日
#101=#3012; ⇒ 加工开始 时/分/秒
G28 X0 Y0 Z0 ;
G92 ;
G0 X50. ;
.
.
.
#102=#3011; ⇒ 加工结束 年/月/日
#103=#3012; ⇒ 加工结束 时/分/秒
M30 ;
```



## 时刻读取变量的限制事项与注意事项

(1) #3011以8位数读取日期, 因此读取到的两个时期差值不是天数。

(2) #3012以6位数读取时刻, 因此读取到的两个时刻之间差值不是时间差。

13.9.6 运算指令

可在变量间进行各种演算。



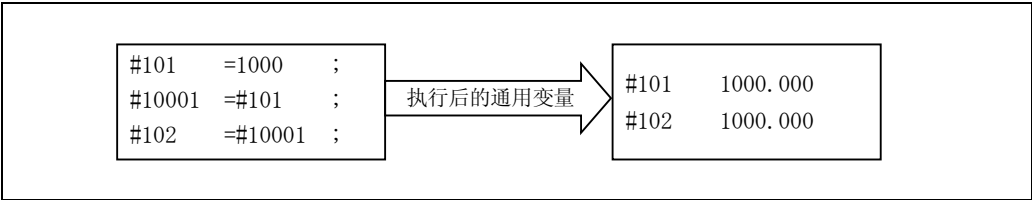
指令格式

#I = <运算式> ;

<运算式>是由常数、变量、函数或运算符构成。  
也可使用常数取代下述的#j、#k。

|              |                   |                            |
|--------------|-------------------|----------------------------|
| (1) 变量的定义、替换 | #i = #j           | 定义、替换                      |
| (2) 加法型运算    | #i = #j + #k      | 加运算                        |
|              | #i = #j - #k      | 减运算                        |
|              | #i = #j OR #k     | 逻辑和 (32bit 的各 bit)         |
|              | #i = #j XOR #k    | 排他性逻辑和 ( " )               |
| (3) 乘法型运算    | #i = #j * #k      | 乘运算                        |
|              | #i = #j / #k      | 除运算                        |
|              | #i = #j MOD #k    | 取余                         |
|              | #i = #j AND #k    | 逻辑积 (32bit 的各 bit)         |
| (4) 函数       | #i = SIN [ #k ]   | 正弦                         |
|              | #i = COS [ #k ]   | 余弦                         |
|              | #i = TAN [ #k ]   | 正接 tan θ 使用 sin θ /cos θ 。 |
|              | #i = ASIN [ #k ]  | 反正弦                        |
|              | #i = ATAN [ #k ]  | 反正接 (ATAN 或 ATN 均可)        |
|              | #i = ACOS [ #k ]  | 反余弦                        |
|              | #i = SQRT [ #k ]  | 平方值 (SQRT 或 SQR 均可)        |
|              | #i = ABS [ #k ]   | 绝对值                        |
|              | #i = BIN [ #k ]   | 从 BCD 变换为 BINARY           |
|              | #i = BCD [ #k ]   | 从 BINARY 变换为 BCD           |
|              | #i = ROUND [ #k ] | 四舍五入 (ROUND 或 RND 均可)      |
|              | #i = FIX [ #k ]   | 舍弃小数点以下                    |
|              | #i = FUP [ #k ]   | 小数点以下进位                    |
|              | #i = LN [ #k ]    | 自然对数                       |
|              | #i = EXP [ #k ]   | 以 e (=2.718……) 为底的对数       |

(注 1) 没有小数点的值，基本上都作为末尾有小数点 (1=1.000) 使用。  
(注 2) 从#10001 开始的偏移量、从#5201 开始的工件坐标系偏移值等，是带小数点的数据。因此，即使将没有小数点的数据定义到那些变量编号，也会成为带小数点的数据。  
(例)

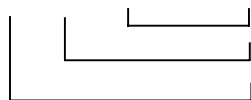




## 运算顺序

(1) 从①到③的运算顺序为，函数、乘法型运算、加法型运算。

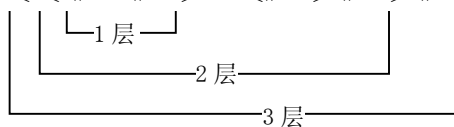
#101=#111+#112\*SIN (#113)



- ① 函数
- ② 乘法型运算
- ③ 加法型运算

(2) 可使用 [ ] 将想提高运算优先级的部分括起来。包括函数的 [ ] 在内，[ ] 最多可有5层。

#101=SQRT [ [ [ #111-#112] \*SIN (#113) +#114] \*#115] ;





运算指令例

|                    |                                                                                                                                                                                                                          |                                                                                                                                                                                                           |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) 主程序与自变量<br>指定  | G65 P100 A10 B20. ;<br>#101 = 100.000 #102 = 200.000;                                                                                                                                                                    | #1      10.000<br>#2      20.000<br>#101    100.000<br>#102    200.000                                                                                                                                    |
| (2) 定义、替换<br>=     | #1 = 1000<br>#2 = 1000.<br>#3 = #101<br>#4 = #102<br>#5 = #10001 (#10001=-10.)                                                                                                                                           | #1    1000.000<br>#2    1000.000<br>#3    100.000<br>#4    200.000<br>#5    -10.000    …根据刀具偏移<br>}    根据通用变量                                                                                             |
| (3) 加运算、减运算<br>+ - | #11 = #1+1000<br>#12 = #2-50.<br>#13 = #101 + #1<br>#14 = #10001-3. (#10001=-10.)<br>#15 = #10001 + #102                                                                                                                 | #11    2000.000<br>#12    950.000<br>#13    1100.000<br>#14    -13.000<br>#15    190.000                                                                                                                  |
| (4) 乘运算、除运算<br>* / | #21 = 100 * 100<br>#22 = 100. * 100<br>#23 = 100 * 100.<br>#24 = 100. * 100.<br>#25 = 100 / 100<br>#26 = 100. / 100<br>#27 = 100 / 100.<br>#28 = 100. / 100.<br>#29 = #10001 * #101 (#10001=-10.)<br>#30 = #10001 / #102 | #21    10000.000<br>#22    10000.000<br>#23    10000.000<br>#24    10000.000<br>#25          1.000<br>#26          1.000<br>#27          1.000<br>#28          1.000<br>#29    -1000.000<br>#30    -0.050 |
| (5) 剩余<br>MOD      | #19 = 48<br>#20 = 9<br>#31 = #19 MOD #20                                                                                                                                                                                 | $\frac{\#19}{\#20} = \frac{48}{9}$ 5 余 3                                                                                                                                                                  |
| (6) 逻辑和<br>OR      | #3 = 100<br>#4 = #3 OR 14                                                                                                                                                                                                | #3 = 01100100 (2 进制数字)<br>14 = 00001110 (2 进制数字)<br>#4 = 01101110 = 110                                                                                                                                   |
| (7) 排他性逻辑和<br>XOR  | #3 = 100<br>#4 = #3 XOR 14                                                                                                                                                                                               | #3 = 01100100 (2 进制数字)<br>14 = 00001110 (2 进制数字)<br>#4 = 01101010 = 106                                                                                                                                   |
| (8) 逻辑积<br>AND     | #9 = 100<br>#10 = #9 AND 15                                                                                                                                                                                              | #9 = 01100100 (2 进制数字)<br>15 = 00001111 (2 进制数字)<br>#10= 00000100 = 4                                                                                                                                     |

|                              |                                                                                                                                                                                   |                                                                                                                                        |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| (9) 正弦<br>SIN                | #501 = SIN [60]<br>#502 = SIN [60. ]<br>#503 = 1000 * SIN [60]<br>#504 = 1000 * SIN [60. ]<br>#505 = 1000. * SIN [60]<br>#506 = 1000. * SIN [60. ]<br>(注) SIN [60]与 SIN [60. ]等价。 | #501        0.866<br>#502        0.866<br>#503        866.025<br>#504        866.025<br>#505        866.025<br>#506        866.025     |
| (10) 余弦<br>COS               | #541 = COS [45]<br>#542 = COS [45. ]<br>#543 = 1000 * COS [45]<br>#544 = 1000 * COS [45. ]<br>#545 = 1000. * COS [45]<br>#546 = 1000. * COS [45. ]<br>(注) COS [45]与 COS [45. ]等价。 | #541        0.707<br>#542        0.707<br>#543        707.107<br>#544        707.107<br>#545        707.107<br>#546        707.107     |
| (11) 正接<br>TAN               | #551 = TAN [60]<br>#552 = TAN [60. ]<br>#553 = 1000 * TAN [60]<br>#554 = 1000 * TAN [60. ]<br>#555 = 1000. * TAN [60]<br>#556 = 1000. * TAN [60. ]<br>(注) TAN [60]与 TAN [60. ]等价。 | #551        1.732<br>#552        1.732<br>#553        1732.051<br>#554        1732.051<br>#555        1732.051<br>#556        1732.051 |
| (12) 反正弦<br>ASIN             | #531 = ASIN [100.500 / 201. ]<br>#532 = ASIN [100.500 / 201]<br>#533 = ASIN [0.500]<br>#534 = ASIN [-0.500]                                                                       | #531        30.000<br>#532        30.000<br>#533        30.000<br>#534        -30.000<br>(注) #1273/bit0 为 1 时,<br>成为#534 = 330°。       |
| (13) 反正接<br>ATN<br>或<br>ATAN | #561 = ATAN [173205 / 100000]<br>#562 = ATAN [173205 / 100000. ]<br>#563 = ATAN [173.205 / 100]<br>#564 = ATAN [173.205 / 100. ]<br>#565 = ATAN [1.73205]                         | #561        60.000<br>#562        60.000<br>#563        60.000<br>#564        60.000<br>#565        60.000                             |
| (14) 反余弦<br>ACOS             | #521 = ACOS [100 / 141.421]<br>#522 = ACOS [100. / 141.421]                                                                                                                       | #521        45.000<br>#522        45.000                                                                                               |
| (15) 平方根<br>SQR<br>或<br>SQRT | #571 = SQRT [1000]<br>#572 = SQRT [1000. ]<br>#573 = SQRT [10. * 10. + 20. * 20]<br>(注) 为了提高精度,请尽可能在[ ]中<br>进行运算。                                                                 | #571        31.623<br>#572        31.623<br>#573        22.360                                                                         |

|                                |                                                                                                                                                                                                              |                                                                                                                                                                                                      |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (16) 绝对值<br>ABS                | #576 = -1000<br>#577 = ABS [#576]<br>#3 = 70. #4 = -50.<br>#580 = ABS [#4 - #3]                                                                                                                              | #576        -1000.000<br>#577        1000.000<br><br>#580        120.000                                                                                                                             |
| (17) BIN、<br>BCD               | #1 = 100<br>#11 = BIN [#1]<br>#12 = BCD [#1]                                                                                                                                                                 | #11            64<br>#12            256                                                                                                                                                              |
| (18) 四舍五入<br>RND<br>或<br>ROUND | #21 = ROUND [14 / 3]<br>#22 = ROUND [14. / 3]<br>#23 = ROUND [14 / 3.]<br>#24 = ROUND [14. / 3.]<br><br>#25 = ROUND [-14 / 3]<br>#26 = ROUND [-14. / 3]<br>#27 = ROUND [-14 / 3.]<br>#28 = ROUND [-14. / 3.] | #21            5<br>#22            5<br>#23            5<br>#24            5<br><br>#25            -5<br>#26            -5<br>#27            -5<br>#28            -5                                 |
| (19) 小数点以下<br>舍弃<br>FIX        | #21 = FIX [14 / 3]<br>#22 = FIX [14. / 3]<br>#23 = FIX [14 / 3.]<br>#24 = FIX [14. / 3.]<br><br>#25 = FIX [-14 / 3]<br>#26 = FIX [-14. / 3]<br>#27 = FIX [-14 / 3.]<br>#28 = FIX [-14. / 3.]                 | #21            4.000<br>#22            4.000<br>#23            4.000<br>#24            4.000<br><br>#25            -4.000<br>#26            -4.000<br>#27            -4.000<br>#28            -4.000 |
| (20) 进位<br>FUP                 | #21 = FUP [14 / 3]<br>#22 = FUP [14. / 3]<br>#23 = FUP [14 / 3.]<br>#24 = FUP [14. / 3.]<br><br>#25 = FUP [-14 / 3]<br>#26 = FUP [-14. / 3]<br>#27 = FUP [-14 / 3.]<br>#28 = FUP [-14. / 3.]                 | #21            5.000<br>#22            5.000<br>#23            5.000<br>#24            5.000<br><br>#25            -5.000<br>#26            -5.000<br>#27            -5.000<br>#28            -5.000 |
| (21) 自然对数<br>LN                | #10 = LN [5]<br>#102 = LN [0.5]<br>#103 = LN [-5]                                                                                                                                                            | #101           1.609<br>#102           -0.693<br>错误           "P282"                                                                                                                                 |
| (22) 指数<br>EXP                 | #104 = EXP [2]<br>#105 = EXP [1]<br>#106 = EXP [-2]                                                                                                                                                          | #104           7.389<br>#105           2.718<br>#106           0.135                                                                                                                                 |



运算精度

通过进行1次运算，发生下表中所示的误差，并随着反复运算而被累积。

| 运算形式                     | 平均误差                   | 最大误差                   | 误差种类                                                                                                  |
|--------------------------|------------------------|------------------------|-------------------------------------------------------------------------------------------------------|
| a = b + c<br>a = b - c   | $2.33 \times 10^{-10}$ | $5.32 \times 10^{-10}$ | $\text{Min} \cdot \left  \frac{\varepsilon}{b} \right  \text{、} \left  \frac{\varepsilon}{c} \right $ |
| a = b * c                | $1.55 \times 10^{-10}$ | $4.66 \times 10^{-10}$ | 相对误差<br>$\left  \frac{\varepsilon}{a} \right $                                                        |
| a = b / c                | $4.66 \times 10^{-10}$ | $1.86 \times 10^{-9}$  |                                                                                                       |
| a = $\sqrt{b}$           | $1.24 \times 10^{-9}$  | $3.73 \times 10^{-9}$  |                                                                                                       |
| a = SIN[b]<br>a = COS[b] | $5.0 \times 10^{-9}$   | $1.0 \times 10^{-8}$   | 绝对误差<br>$ \varepsilon ^\circ$                                                                         |
| a = ATAN[b/c]            | $1.8 \times 10^{-6}$   | $3.6 \times 10^{-6}$   |                                                                                                       |

(注) 函数 TAN 进行 SIN / COS 的计算。



精度下降相关注意事项

(1) 加减运算

在加法运算或减法运算中，当绝对值起减法运算的功能时，无法将相对误差控制在10以内。

例如，根据#10与#20的运算结果取真值（该值无法直接代入），如下情况下

#10 = 234567898888.888  
#20 = 2345678901234.567

即使进行运算 #10 - #20，也无法得到#10 - #20 = 87654.321。这是因为，变量的有效位数为10进8进，所以上述#10与#20的值分别为

#10 = 2345679000000.000  
#20 = 2345678900000.000

左右的值（内部数据为2进制数值，所以严格来说，与上述的值有些不同）。因此，实际上

#10 - #20 = 100000.000

会发生较大的误差。

(2) 逻辑关系

关于EQ、NE、GT、LT、GE、LE，也进行与加减法运算相同的计算，所以请充分注意误差。例如在上例中，为了判断#10与#20是否相等而使用

IF [#10 EQ #20]

则由于前述的误差的缘故，无法进行正确判断。因此，如下式所述，如果#10与#20的差在规定的误差范围内，则视为两者相同。

IF [ABS [#10 - #20] LT 200000]

(3) 三角函数

在三角函数中，虽然能够保证绝对误差，但是因为相对误差大于10，所以在三角函数运算后进行乘除运算时，请加以注意。

13.9.7 控制指令

可通过IF～GOTO～及WHILE～DO～控制程序的流程。



分歧

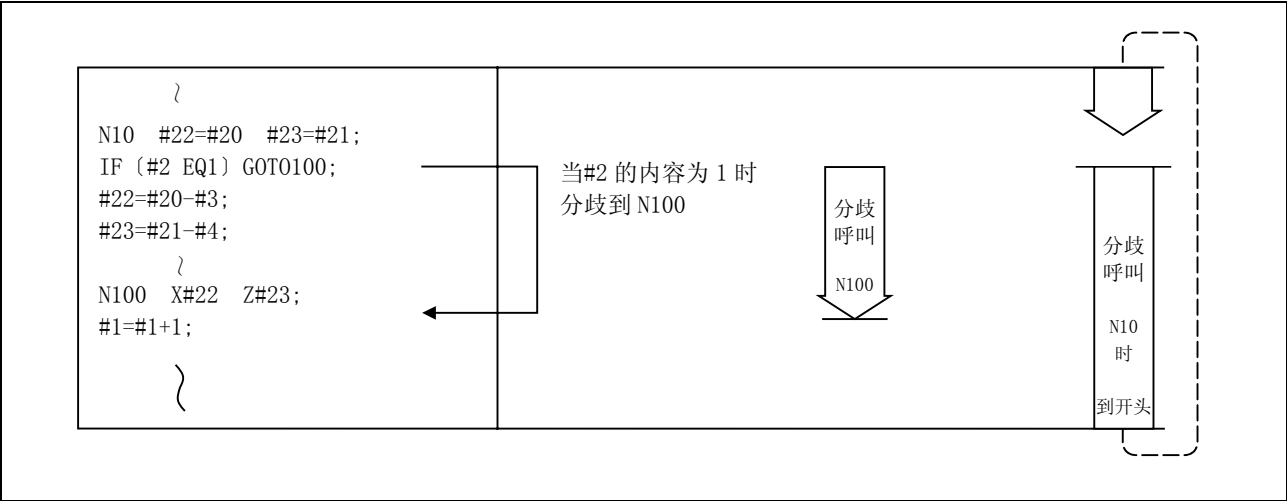
格式

**IF [条件式] GOTO n;** （n 为该程序内的顺序编号。）

条件成立时，分歧到n，不成立时，执行下一节。  
IF[条件式]也可省略，如省略，则无条件的分歧到n。  
[条件式]中，包括如下的种类。

|          |   |              |
|----------|---|--------------|
| #i EQ #j | = | #i 与 #j 相等时  |
| #i NE #j | ≠ | #i 与 #j 不相等时 |
| #i GT #j | > | #i 大于 #j 时   |
| #i LT #j | < | #i 小于 #j 时   |
| #i GE #j | ≧ | #i 大于等于 #j 时 |
| #i LE #j | ≦ | #i 小于等于 #j 时 |

GOTO n的n必须存在于同一程序内，如果不存在，则发生程序错误（P231）。可使用运算式或变量代替#i、#j、n。  
在GOTO n之后执行的顺序编号为n的节中，顺序编号Nn必须存在于节的开头，如果没有，则发生程序错误（P231）。  
但是，当节的开头为“/”，之后有Nn时，可以分歧到该顺序编号。



（注 1）呼叫分歧目标顺序编号时，从 IF……………；的下一节开始，呼叫到程序结束（%代码），如果没有，则从程序开头呼叫到 IF……………；的前一节。因此，向程序流程的反方向进行的分歧呼叫，与顺向分歧呼叫相比，执行时间更长。

（注 2）EQ 及 NE 的比较，请仅对整数进行比较。当在有小数点以下数值的情况下进行比较时，请使用 GE、GT、LE、LT。



重复

格式

```
WHILE [条件式] D0m ; (m = 1, 2, 3 127)
{
END m ;
```

当条件式成立时，从下一节开始，重复ENDm节，不成立时，转为执行ENDm的下一节。D0m即使在WHILE之前也没问题。

WHILE[条件式]D0m与ENDm应成对使用，如果省略WHILE[条件式]，则D0m~ENDm无限重复。重复识别编号为1~127。（D01、D02、D03、……D0127）但是，嵌套深度最多27层。

|                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>① 多次使用同一识别编号。</p> <div><p>可</p><div><div>WHILE~D01;</div><div>END1;</div></div><p>可</p><div><div>WHILE~D01;</div><div>END1;</div></div></div>                                                                                                                    | <p>② WHILE~D0m 的识别编号可自由设定。</p> <div><div><div>WHILE~D01;</div><div>END1;</div></div><div><div>WHILE~D03;</div><div>END3;</div></div><div><div>WHILE~D02;</div><div>END2;</div></div><div><div>WHILE~D01;</div><div>END1;</div></div></div> |
| <p>③ WHILE~D0m 的嵌套层数最多 27 层。<br/>m 为 1~127，对于嵌套的深度，可自由选择。</p> <div><p>可</p><div><div>WHILE~D01;</div><div>WHILE~D02;</div><div>WHILE~D027;</div><div>END27;</div><div>END2;</div><div>END1;</div></div><div><div>D01</div><div>D02</div><div>D027</div></div></div> | <p>④ WHILE~D0m 的嵌套层数不可超过 27 层。</p> <div><div><div>WHILE~D01;</div><div>WHILE~D02;</div><div>WHILE~D03;</div><div>WHILE~D028;</div><div>END28;</div><div>END3;</div><div>END2;</div><div>END1;</div></div><div><div>不可</div></div></div>    |

(注) 在嵌套时，不能使用已使用过的 m。

|                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                                                                                     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>⑤ 必须先指定 WHILE~D0m, 后指定 ENDm。</p> <div><div>不可</div><div><div>END 1; ;</div><div>WHILE~D01; ;</div></div></div>                                                                                                                                                                                                  | <p>⑥ 在同一节内, WHILE~D0m 与 ENDm 必须一一对应。</p> <div><div>不可</div><div><div>WHILE~D01; ;</div><div>WHILE~D01; ;</div><div>END1; ;</div></div></div>                                                                                                                                        |
| <p>⑦ 2 个 WHILE~D0m 不可交叉使用。</p> <div><div>不可</div><div><div>WHILE~D01; ;</div><div>WHILE~D02; ;</div><div>END1; ;</div><div>END2; ;</div></div></div>                                                                                                                                                               | <p>⑧ 不可分歧到 WHILE~D0m 的范围以外。</p> <div><div>可</div><div><div>WHILE~D01; ;</div><div>IF~GOTO n;</div><div>END1; ;</div><div>Nn; ;</div></div></div>                                                                                                                                    |
| <p>⑨ 不可分歧到 WHILE~D0m 中。</p> <div><div>不可</div><div><div>IF~GOTO n;</div><div>WHILE~D01; ;</div><div>Nn; ;</div><div>END1; ;</div></div><div><div>不可</div><div><div>WHILE~D01; ;</div><div>IF~GOTO n;</div><div>END1; ;</div><div>WHILE~D01; ;</div><div>Nn; ;</div><div>END1; ;</div></div></div></div>            | <p>⑩ 在 WHILE~D0m 之间, 可通过 M98、G65、G66 等调用子程序。</p> <div><div>可</div><div><div>主程序</div><div>WHILE~D01; ;</div><div>G65</div><div>P100; ;</div><div>END1; ;</div><div>M02; ;</div><div>到子程序</div><div>子程序</div><div>WHILE~D002; ;</div><div>END2; ;</div><div>M99; ;</div></div></div> |
| <p>⑪ 在 WHILE~D0m 之间通过 G65、G66 调用宏, 可从 1 开始指令。包括主程序及子程序在内, 最多可进行 27 层嵌套。</p> <div><div>可</div><div><div>主程序</div><div>WHILE~D01; ;</div><div>G65</div><div>P100; ;</div><div>END1; ;</div><div>M02; ;</div><div>到子程序</div><div>子程序(100)</div><div>WHILE~D01; ;</div><div>END1; ;</div><div>M99; ;</div></div></div> | <p>⑫ 子程序 (含宏程序) 内, 如果 WHILE 与 END 没有成对出现, 则因 M99 而发生程序错误。</p> <div><div>主程序</div><div>M98</div><div>P100; ;</div><div>M02; ;</div><div>到子程序</div><div>子程序</div><div>WHILE~D01; ;</div><div>M99; ;</div><div>使用 D0nENDn 发生错误。</div></div>                                            |

(注) 调用含 WHILE 的固定循环时, 也累计嵌套层数。

13.9.8 外部输出指令



功能及目的

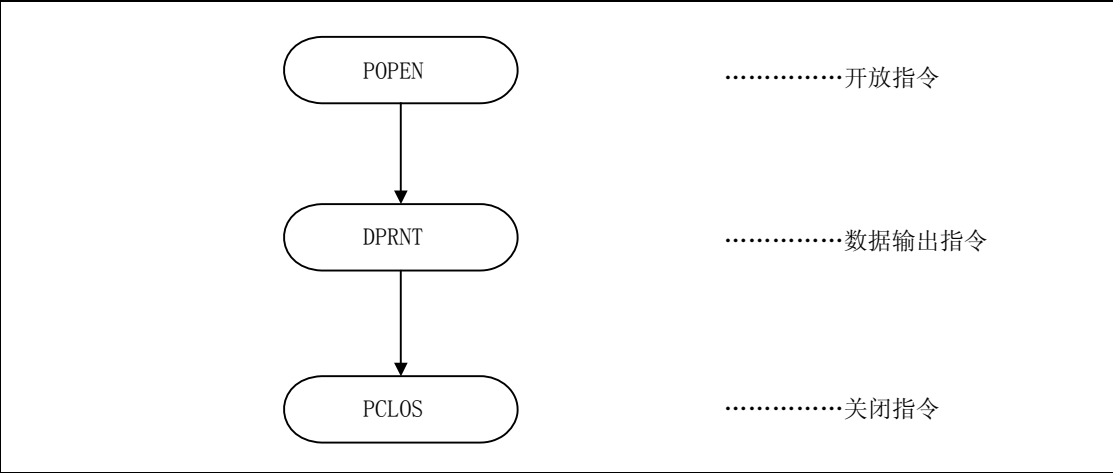
与标准用户宏不同，以下宏命令可作为外部输出指令使用。这些命令是可通过RS-232C接口输出的变量数值及字符。



指令格式

|              |                    |
|--------------|--------------------|
| <b>POPEN</b> | 进行数据输出的准备处理。       |
| <b>PCLOS</b> | 进行数据输出的结束处理。       |
| <b>DPRNT</b> | 进行字符输出及各位变量值的数值输出。 |

指令准备



开放指令 POPEN

- (1) 在一系列的数据输出指令前进行指令。
- (2) 从NC向外部输出设备输出DC2的控制代码与%代码。
- (3) 进行指令之后，则在指令PCLOS；之前始终有效。



关闭指令 PCLOS

- (1) 所有数据输出完成时进行指令。
- (2) 从NC向外部输出设备输出%代码与DC4的控制代码。
- (3) 本指令应与开放指令成对使用，未处于开放模式时，请不要进行关闭指令。
- (4) 在数据输出中，因重新启动等而中断时，请在程序的最后进行关闭指令。



数据输出指令 DPRNT

|                                                 |   |            |             |
|-------------------------------------------------|---|------------|-------------|
| DPRNT [ /1#v1 [d1 c1] /2#v2 [d2 c2] . . . . . ] |   |            |             |
| /1                                              | : | 字符串        |             |
| v1                                              | : | 变量编号       |             |
| d1                                              | : | 小数点以上的有效位数 | } C + d ≤ 8 |
| c1                                              | : | 小数点以下的有效位数 |             |

- (1) 以ISO代码进行字符的输出及变量值的10进制输出。
- (2) 字符串可直接以ISO代码输出指令的字符串。  
可使用英文字母数字（A～Z、0～9）与特殊字符（+、-、\*、/）。
- (3) 在[ ]内，分别指令变量值小数点以上与小数点以下的所需位数。借此，按照所指令的位数，从第一位开始，以ISO代码输出含小数点的10进制变量值。此时，不省略前导零。
- (4) 省略前导0。  
通过设定参数，输出空格代替省略的前导0。  
输出到打印机的数据，可按数据的最后一位对齐。

13.9.9 注意事项



注意事项

- (1) 如果使用用户宏指令，则可将MST指令等控制指令，与运算、判定、分歧等宏指令组合，创建加工程序。如果前者为执行语句，后者为宏语句，则由于宏语句的处理与直接机械控制无关，所以如果能够尽可能短的时间内进行处理，就可以有效的缩短加工时间。
- 为此，可设定参数#8101“宏单独”，在执行执行语句时，对宏语句进行并行处理。
- (通常加工时，可关闭参数，对宏语句进行整体处理，在检查程序时，打开参数，逐节执行每1节宏语句，所以，请根据目的进行设定。)

程序例

G91G28X0 Z0 ; ..... (1)

G92X0 Z0 ; ..... (2)

G00X-100.Z-100. ; ..... (3)

#101=100.\*COS [210.] ; ..... (4)

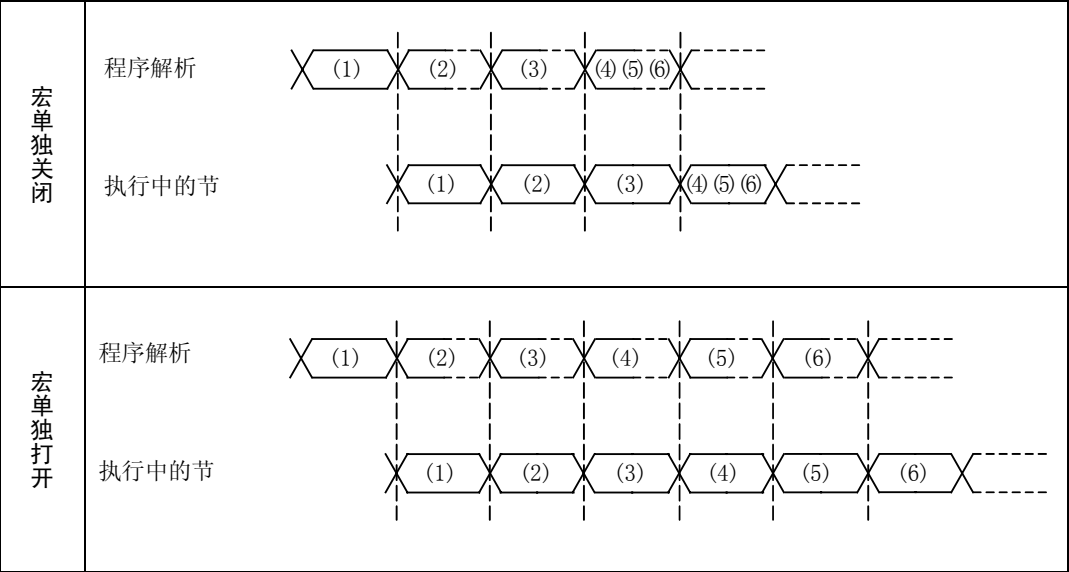
#103=100.\*SIN [210.] ; ..... (5)

G01 X#101 Z#103 F800 ; ..... (6)

宏语句

- 宏语句指以下语句。
- (a) 运算指令 (含=的节)
  - (b) 控制指令 (含GOTO.D0~END等的节)
  - (c) 通过宏调用指令 (G代码等) 调用宏及取消指令 (G65、G66、G66.1、G67) 也包括在内
- 另外，执行语句指除宏语句以外的语句。

按照上述程序例进行处理的流程



加工程序的显示

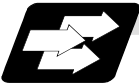
|       |                                                                                               |                                                                                                   |
|-------|-----------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| 宏单独关闭 | <div><div>(执行中) N3 G00 X-100. Z-100. ;</div><div>(下一指令) N6 G01 X#101 Z#103 F800 ;</div></div> | 进行 N3 执行语句的控制时，进行 N4、N5、N6 的并行处理，由于 N6 为执行语句，所以显示为下一指令。如果在 N3 控制中，来得及完成 N4、M5、M6 的解析，则机械控制进行连续运转。 |
| 宏单独打开 | <div><div>(执行中) N3 G00 X-100. Z-100. ;</div><div>(下一指令) N4 #101=100. *COS[210.] ;</div></div> | 进行 N3 执行语句的控制时，对 N4 进行并行处理，显示为下一指令。N3 结束后，对 N5、N6 进行解析，执行 N6，所以，机械控制会等待进行 N5、N6 解析所需的时间           |

13.10 倒角/圆角 I

在仅以直线形成转角的指令节内，通过在先指令的节最后附加“*C\_*”或“*R\_*”，自动进行任意角度的倒角或圆角处理。

通过参数设定，可使用“*I\_*”、“*K\_*”、“*C\_*”代替“*C\_*”进行倒角，使用“*R\_*”代替“*R\_*”进行圆角处理。

13.10.1 倒角“*C\_*”（或“*I\_*”、“*K\_*”、“*C\_*”）



功能及目的

在假定不进行倒角时的虚拟转角前后，分别减去各“*C\_*”（或“*I\_*”、“*K\_*”、“*C\_*”）指令长度，连接这2个位置，进行倒角处理。



指令格式

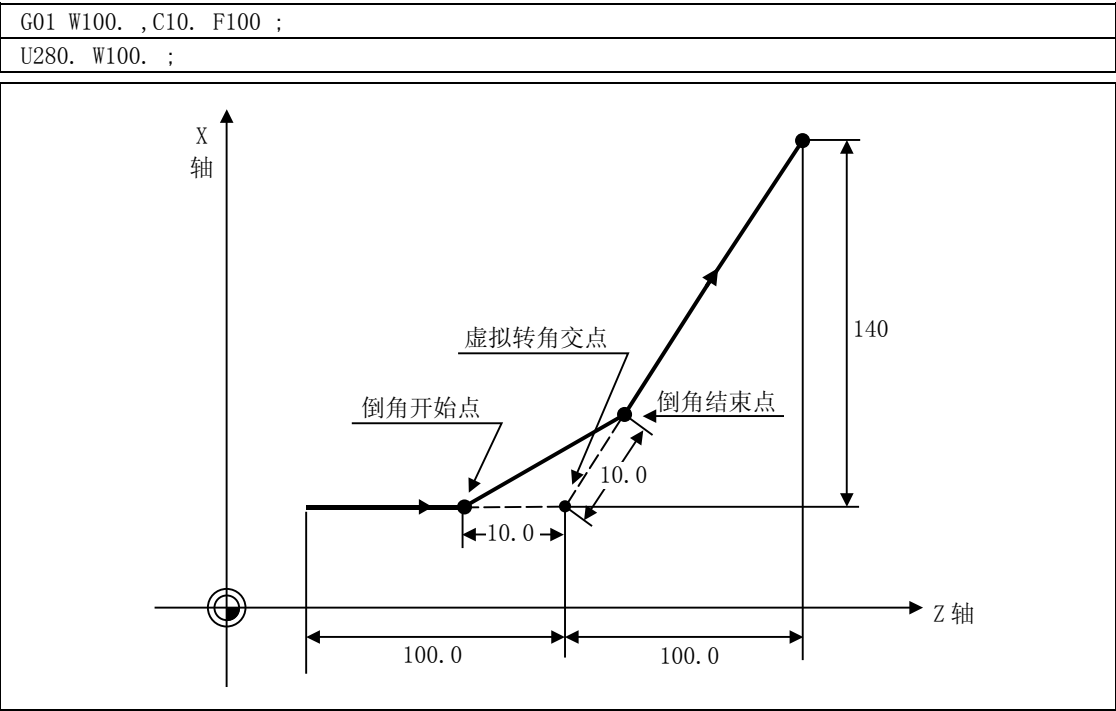
```
N100 G01 X_ Z_C_ (或 I_ / K_ / C_) ;
N200 G01 X_ Z_ ;

C_ / I_ / K_ / C_ : 从虚拟转角到倒角开始点、或倒角结束点的长度
```

在N100与N200的交点进行倒角处理。



程序例





## 详细说明

- (1) 以倒角的下一节起点，作为虚拟转角交点。
- (2) 当基本规格参数“#1272 ext08/bit6”为“0”时，如果没有“，C”的“，”，则看作为C指令。
- (3) 当在同一节中，指令了多个倒角指令，或是倒角指令重复时，最后的指令有效。
- (4) 在同一节中，存在倒角/圆角两种指令时，最后的指令有效。
- (5) 对进行了倒角之后的形状，计算刀具偏移。
- (6) 如倒角指令的下一节不是直线指令，则变为倒角/圆角 II。
- (7) 在指令了倒角的节中，当移动量小于倒角量时，发生程序错误（P383）。
- (8) 在指令了倒角节的下一节中，当移动量小于倒角量时，发生程序错误（P383）。
- (9) 当倒角 I 指令的下一节不是移动指令时，发生程序错误（P382）。
- (10) 作为轴名称及第2辅助功能使用了“C”时，无法以“C”发出倒角指令。
- (11) 无法在圆弧指令节中以“I”或“K”进行倒角指令。“I”、“K”为圆弧中心指令。



## 注意事项

- (1) 仅当倒角/圆角指令的第1节为直线时，可指令使用了“I”、“K”“R”的倒角/圆角。
- (2) 当倒角/圆角指令的第1节为直线，第2节为圆弧指令时，可根据“I”、“K”进行倒角及根据“R”进行圆角指令。第2节的“I”、“K”为圆弧中心指令。  
 N100 G01 Xx Zz Ii ;.....Ii 倒角长度  
 N200 G02 Xx Zz Ii Kk ;.....Ii,Kk 圆弧中心指令
- (3) 当在同一节中，有“，C\_”、“，R\_”或“I\_”、“C\_”、“R\_”时，“，C\_”、“，R\_”被优先处理。

13.10.2 圆角 “,R\_” (或 “R\_”)



功能及目的

在假定不进行圆角处理的虚拟转角前后，分别以通过 “,R\_” (或 “R\_” ) 指令半径的圆弧进行圆角处理。



指令格式

N100 G01 X\_ Z\_ ,R\_ (或 R\_ ) ;

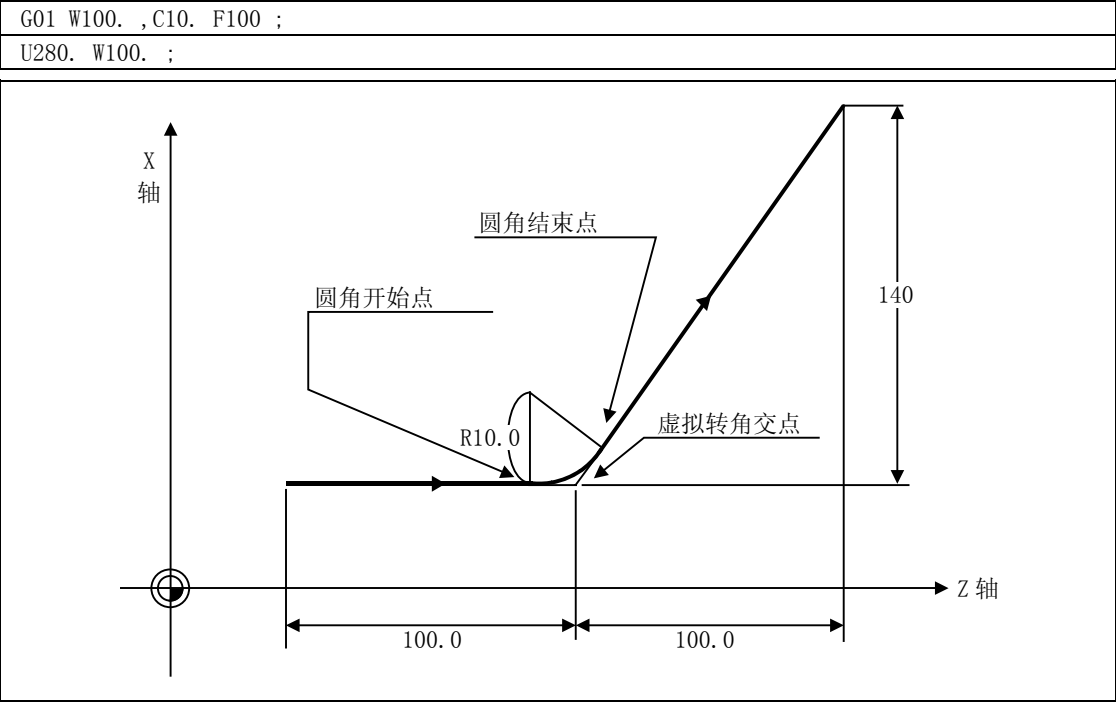
N200 G01 X\_ Z\_ ;

,R\_ / R\_ : 圆角 圆弧半径

在N100与N200的交点进行圆角处理。



程序例





## 详细说明

- (1) 以转角的下一节起点，作为虚拟转角交点。
- (2) 当基本规格参数“#1272 ext08/bit6”为“0”时，如果没有“R”的“，”，则看作为R指令。
- (3) 在同一节中，存在倒角/圆角两种指令时，最后的指令有效。
- (4) 对进行了圆角之后的形状，计算刀具偏移。
- (5) 如圆角指令的下一节不是直线指令，则变为倒角/圆角 II。
- (6) 在指令了圆角的节中，当移动量小于R值时，发生程序错误（P383）。
- (7) 在指令了圆角节的下一节中，当移动量小于R值时，发生程序错误（P384）。
- (8) 当圆角指令的下一节不是移动指令时，发生程序错误（P382）。
- (9) 在圆弧指令节中，无法以“R”指令圆角。“R”为圆弧半径指令。



## 注意事项

- (1) 仅当倒角/圆角指令的第1节为直线时，可指令使用了“I”、“K”“R”的倒角/圆角。
- (2) 当倒角/圆角指令的第1节为直线，第2节为圆弧指令时，可根据“I”、“K”进行倒角及根据“R”进行圆角指令。第2节的“I”、“K”为圆弧中心指令。
  - N100 G01 Xx Zz Ii ;.....Ii 倒角长度
  - N200 G02 Xx Zz Ii Kk ;.....Ii,Kk 圆弧中心指令
- (3) 当在同一节中，有“C\_”、“R\_”或“I\_”、“C\_”、“R\_”时，“C\_”、“R\_”被优先处理。

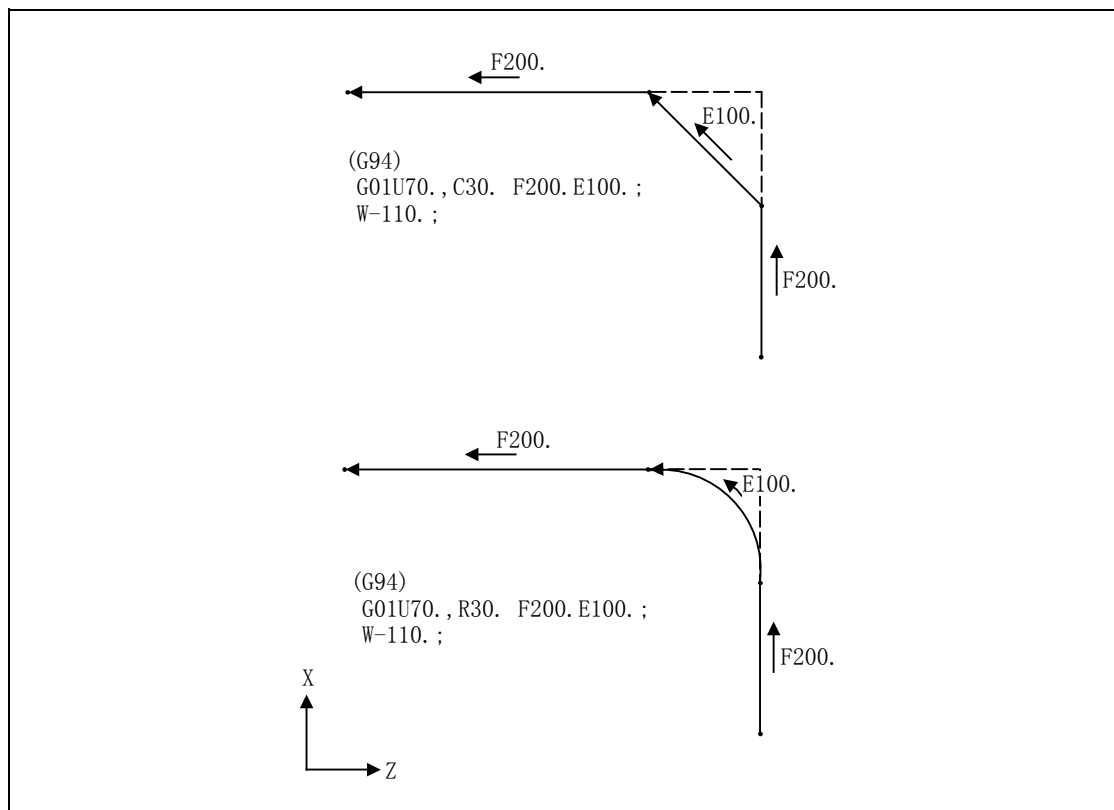
## 13.10.3 倒角/圆角扩展



## 功能及目的

通过E指令，可指定倒角、转角部分的进给速度。  
借此，可切削出转角部分的正确形状。

例

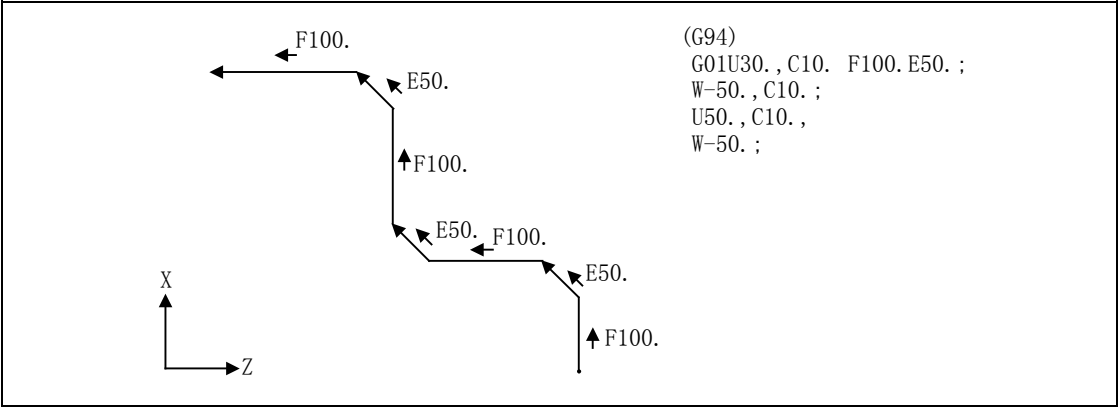




详细说明

(1) E指令为模态。对于下一倒角/圆角部分的进给也有效。

例

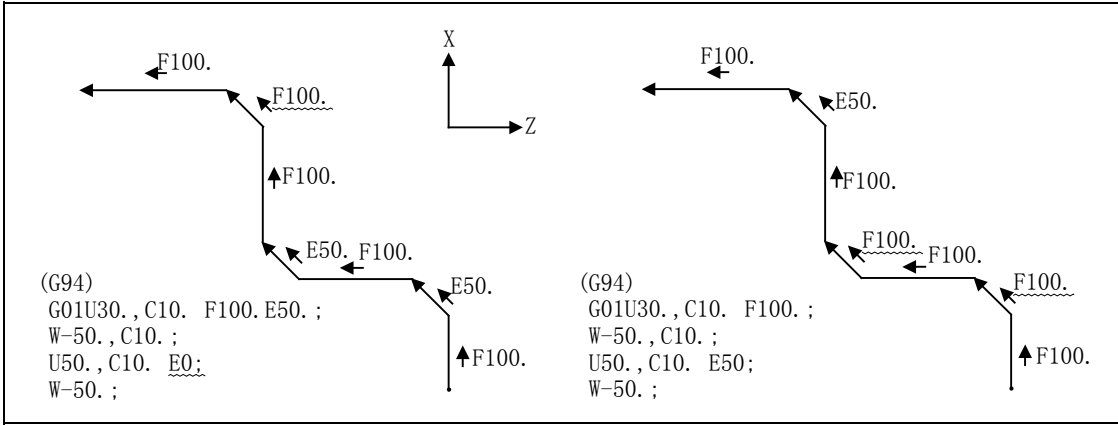


(2) E指令的模态，分别具有非同步进给速度模态和同步进给速度模态。

哪一个速度有效，取决于非同步/同步模式（G94/G95）。

(3) 当E指令为0或迄今为止没有指定过E指令时，倒角/圆角部分的进给速度，采用F指令的进给速度。

例



(4) E指令的模态，即使按下重新启动按钮，也不会被清除。

将电源关闭，则被清除。（与F指令相同。）

(5) 如下所示场合以外的E指令，成为倒角/圆角部分的进给速度。

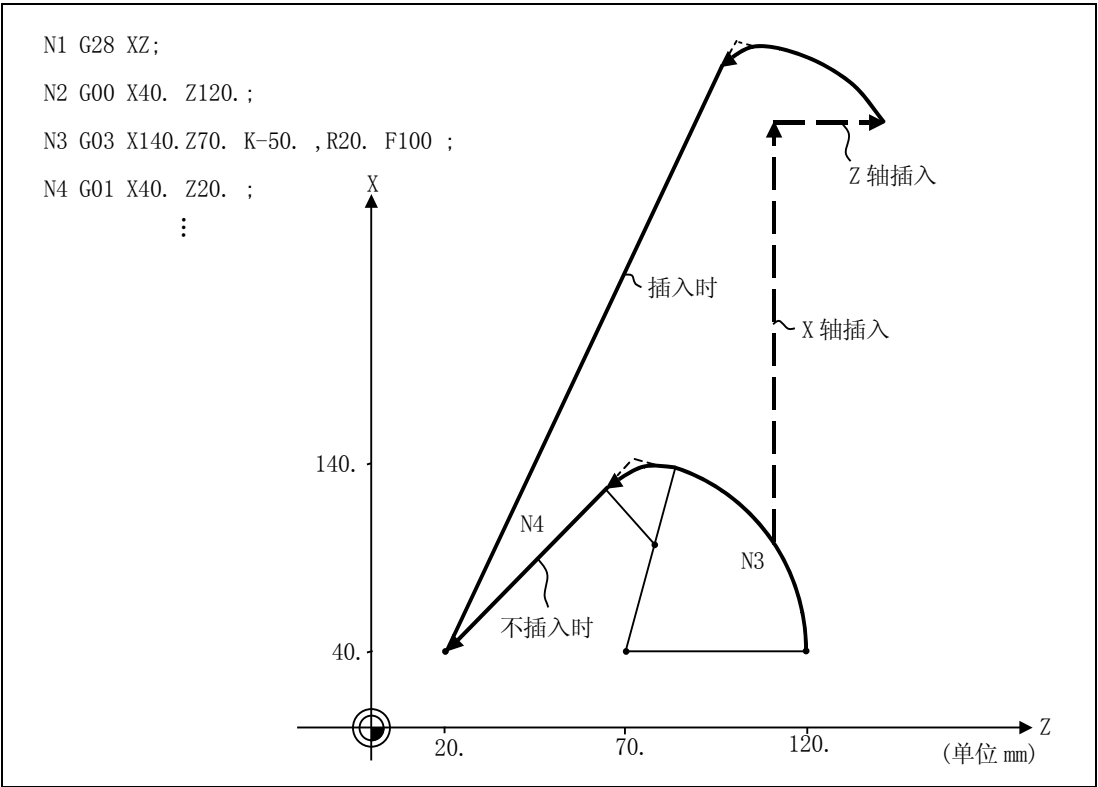
- 螺纹切削模态中的E指令
- 螺纹切削循环模态中的E指令

13.10.4 倒角/圆角R中插入动作

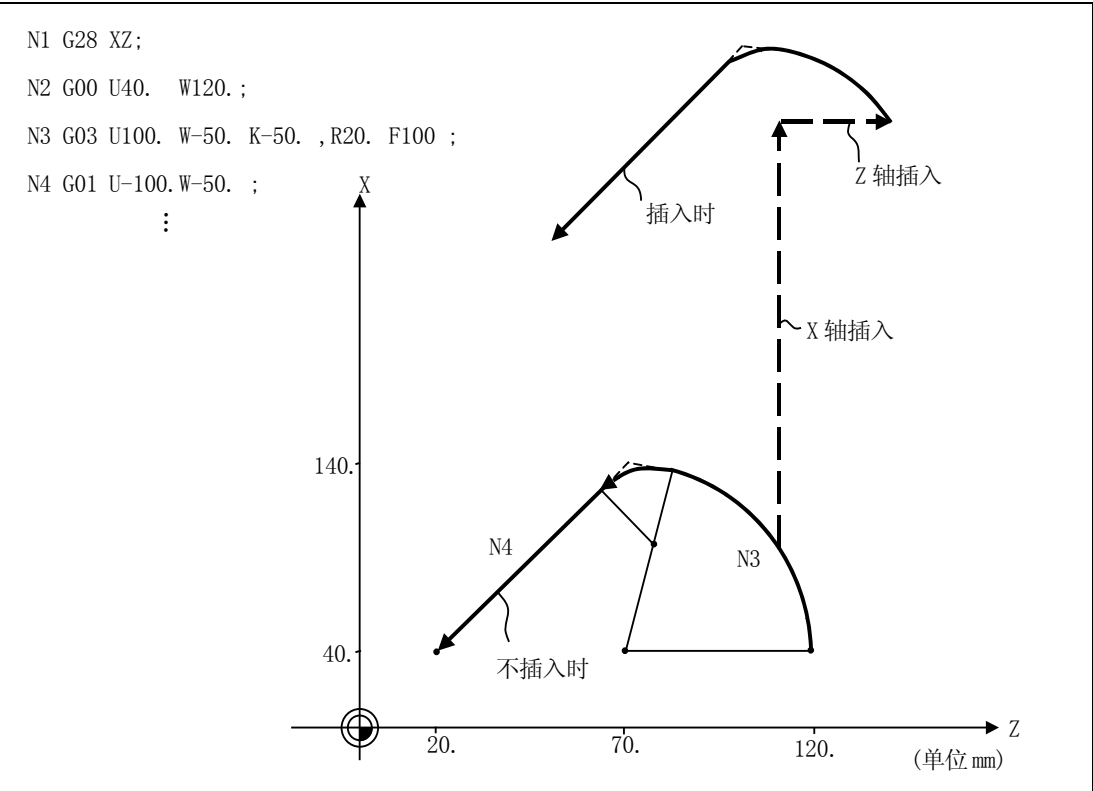


詳細説明

(1) 倒角、圆角R中的手动插入时动作，如下所示。  
绝对值指令且手动绝对开关打开时



增量值指令或手动绝对开关关闭时



(2) 在执行倒角、圆角之后，停止倒角、圆角中的单节。

13.11 倒角/圆角 II

在指令了由连续任意角度的直线或圆弧构成转角的节中，通过在先指令的节的最后附加“C”或“R”，进行倒角、圆角处理。

通过参数设定，可使用“L\_”、“K\_”、“C\_”代替“C\_”进行倒角，使用“R\_”代替“R\_”进行圆角处理。在倒角、圆角处理中，绝对值指令与增量值指令均可使用。

13.11.1 倒角“C\_”（或“L\_”、“K\_”、“C\_”）



功能及目的

对于含有连续圆弧的2节，通过在第一节中指令“C”（或“L\_”、“K\_”、“C\_”），进行倒角。圆弧时，为弦的长度。



指令格式

```
N100 G03 X_ Z_ L_ K_ ,C_ (或 C_) ;
N200 G01 X_ Z_ ;

,C_ / C_ : 从虚拟转角到倒角开始点、或倒角结束点的长度
```

在N100与N200的交点进行倒角处理。



程序例

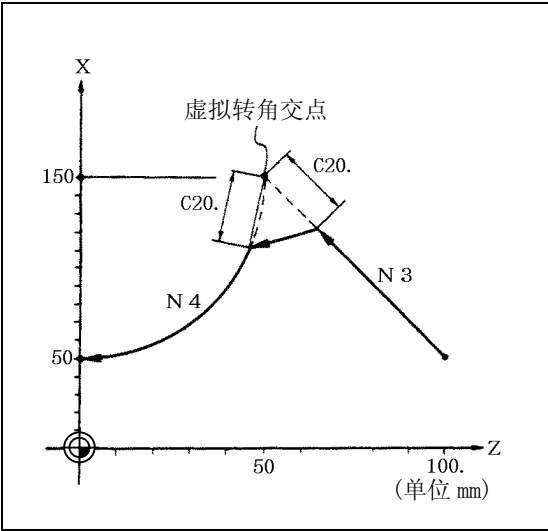
(1) 直线—圆弧

绝对值指令

|                                |
|--------------------------------|
| N1 G28 X Z ;                   |
| N2 G00 X50. Z100. ;            |
| N3 G01 X150. Z50. ,C20. F100 ; |
| N4 G02 X50. Z0 I0 K-50. ;      |
| :                              |

相对值指令

|                                |
|--------------------------------|
| N1 G28 X Z ;                   |
| N2 G00 U25. W100. ;            |
| N3 G01 U50. W-50. ,C20. F100 ; |
| N4 G02 U-50. W-50. I0 K-50. ;  |
| :                              |



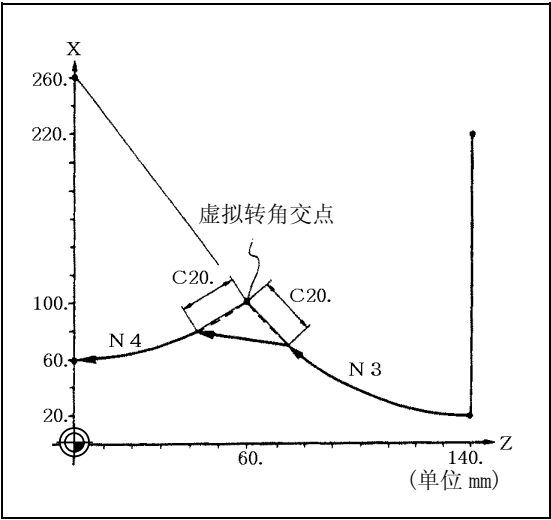
(2) 圆弧—圆弧

绝对值指令

|                                        |
|----------------------------------------|
| N1 G28 X Z;                            |
| N2 G00 X20. Z140. ;                    |
| N3 G02 X100. Z60 I100 K0. ,C20. F100 ; |
| N4 X60. Z0 I80. K-60. ;                |
| ;                                      |

相对值指令

|                                      |
|--------------------------------------|
| N1 G28 X Z ;                         |
| N2 G00 U10. W140. ;                  |
| N3 G02 U40. W-80. R100. ,C20. F100 ; |
| N4 U-20. W-60. I80. K-60. ;          |
| ;                                    |



详细说明

- (1) 以倒角的下一节起点，作为虚拟转角交点。
- (2) 当基本规格参数“#1272 ext08/bit6”为“0”时，如果没有“C”的“,”，则看作为C指令。
- (3) 当在同一节中，指令了多个倒角指令，或是倒角指令重复时，最后的指令有效。
- (4) 在同一节中，存在倒角/圆角两种指令时，最后的指令有效。
- (5) 对进行了倒角之后的形状，计算刀具偏移。
- (6) 当倒角指令节或下一节为定位指令或螺纹切削指令时，发生程序错误（P385）。
- (7) 倒角的下一节为组01以外的G指令或其他指令时，发生程序错误（P382）。
- (8) 在指令了倒角的节中，当移动量小于倒角量时，发生程序错误（P383）。
- (9) 在指令了倒角节的下一节中，当移动量小于倒角量时，发生程序错误（P383）。
- (10) 在倒角处理时，直径指令也变为半径指令值。
- (11) 当倒角II指令的下一节不是移动指令时，发生程序错误（P382）。
- (12) 作为轴名称及第2辅助功能使用了“C”时，无法以“C”发出倒角指令。
- (13) 无法在圆弧指令节中以“I”或“K”进行倒角指令。“I”、“K”为圆弧中心指令。



## 注意事项

- (1) 仅当倒角/圆角指令的第1节为直线时，可指令使用了“I”、“K”“R”的倒角/圆角。
- (2) 当倒角/圆角指令的第1节为直线，第2节为圆弧指令时，可根据“I”、“K”进行倒角及根据“R”进行圆角指令。第2节的“I”、“K”为圆弧中心指令。  
N100 G01 Xx Zz Ii ;.....Ii 倒角长度  
N200 G02 Xx Zz Ii Kk ;.....Ii,Kk 圆弧中心指令
- (3) 当在同一节中，有“C\_”、“R\_”或“I\_”、“C\_”、“R\_”时，“C\_”、“R\_”被优先处理。

13.11.2 圆角 “,R\_” (或 “R\_”)



功能及目的

对于含有连续圆弧的2节，通过在第一节中指令 “,R” (或 “R\_” )，进行圆角处理。



指令格式

N100 G03 X\_ Z\_ I\_ K\_ ,R\_ (或 R\_) ;  
N200 G01 X\_ Z\_ ;  
  
,R\_ / R\_ : 圆角 圆弧半径

在N100与N200的交点进行圆角处理。

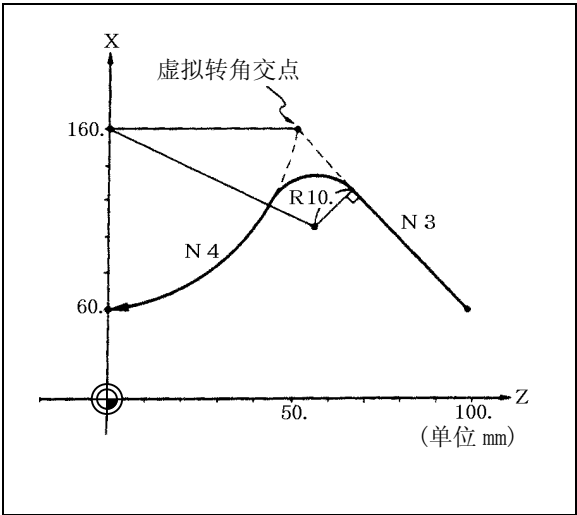


程序例

(1) 直线—圆弧

|                                |
|--------------------------------|
| 绝对值指令                          |
| N1 G28 X Z ;                   |
| N2 G00 X60. Z100. ;            |
| N3 G01 X160. Z50. ,R10. F100 ; |
| N4 G02 X60. Z0 I0 K-50. ;      |
| ;                              |

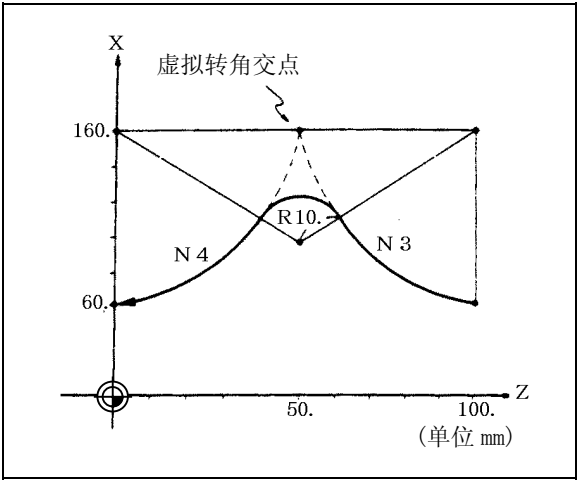
|                                |
|--------------------------------|
| 相对值指令                          |
| N1 G28 X Z ;                   |
| N2 G00 U30. W100. ;            |
| N3 G01 U50. W-50. ,R10. F100 ; |
| N4 G02 U-50. W-50. I0 K-50. ;  |
| ;                              |



(2) 圆弧—圆弧

|                                    |
|------------------------------------|
| 绝对值指令                              |
| N1 G28 X Z ;                       |
| N2 G00 X60. Z100. ;                |
| N3 G02 X160. Z50. R60 ,R10. F100 ; |
| N4 X60. Z0 R50. ;                  |
| ;                                  |

|                                        |
|----------------------------------------|
| 相对值指令                                  |
| N1 G28 X Z ;                           |
| N2 G00 U30. W100. ;                    |
| N3 G02 U50. W-50. I50. K0 ,R10. F100 ; |
| N4 U-50. W-50. I0. K-50. ;             |
| ;                                      |





## 详细说明

- (1) 以圆角的下一节起点，作为虚拟转角交点。
- (2) 当基本规格参数“#1272 ext08/bit6”为“0”时，如果没有“R”的“，”，则看作为R指令。
- (3) 在同一节中，存在倒角/圆角两种指令时，最后的指令有效。
- (4) 对进行了圆角之后的形状，计算刀具偏移。
- (5) 当圆角指令节或下一节为定位指令或螺纹切削指令时，发生程序错误（P385）。
- (6) 圆角的下一节为组01以外的G指令或其他指令时，发生程序错误（P382）。
- (7) 在指令了圆角的节中，当移动量小于R量时，发生程序错误（P383）。
- (8) 在指令了圆角节的下一节中，当移动量小于R量时，发生程序错误（P384）。
- (9) 在圆角处理时，直径指令也变为半径指令值。
- (10) 当圆角II指令的下一节不是移动指令时，发生程序错误（P382）。
- (11) 在圆弧指令节中，无法以“R”指令圆角。“R”为圆弧半径指令。



## 注意事项

- (1) 仅当倒角/圆角指令的第1节为直线时，可指令使用了“I”、“K”“R”的倒角/圆角。
- (2) 当倒角/圆角指令的第1节为直线，第2节为圆弧指令时，可根据“I”、“K”进行倒角及根据“R”进行圆角指令。第2节的“I”、“K”为圆弧中心指令。  
 N100 G01 Xx Zz Ii ;.....Ii 倒角长度  
 N200 G02 Xx Zz Ii Kk ;.....Ii,Kk 圆弧中心指令
- (3) 当在同一节中，有“C\_”、“R\_”或“I\_”、“C\_”、“R\_”时，“C\_”、“R\_”被优先处理。

## 13.11.3 倒角/圆角扩展

详情请参阅“13.10.3 倒角/圆角扩展”。

## 13.11.4 倒角/圆角中插入动作

详情请参阅“13.10.4 倒角/圆角中插入动作”。

13.12 直线角度指令



功能及目的

通过指令直线的角度及终点坐标的任意一轴，自动计算终点坐标。



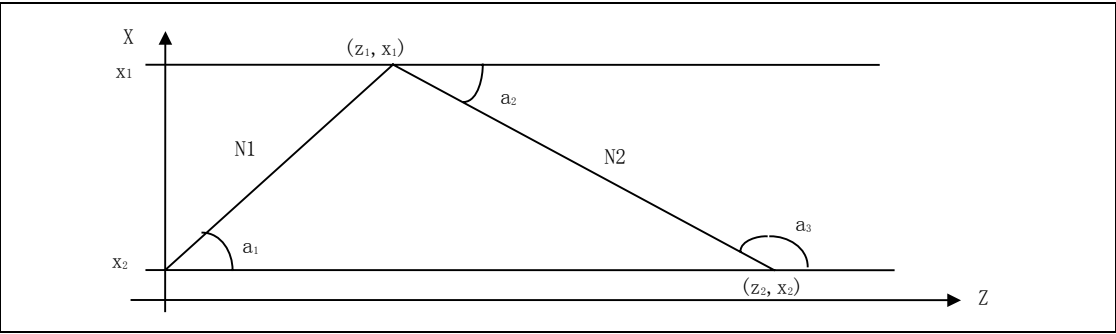
指令格式

```
N1 G01 Xx1(Zz1) Aa1;
N2 G01 Xx2(Zz2) A-a2; (将A-a2改为Aa3也相同。)

或

N1 G01 Xx1(Zz1), Aa1;
N2 G01 Xx2(Zz2), A-a2;
```

指定角度与X轴或Z轴的坐标。  
通过G17~G19选择指令平面。

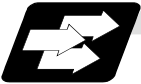


详细说明

- (1) 角度是直线与选择平面的横轴+方向间的角度，逆时针方向（CCW）为+，顺时针方向（CW）为-。
- (2) 终点是指令选择平面轴的任何一轴。
- (3) 当指令了角度与两轴的坐标时，角度被忽略。
- (4) 当仅指令了角度时，看作为几何学指令。
- (5) 关于角度，使用起点（a<sub>1</sub>）、终点（a<sub>2</sub>）任何一个的角度均可。
- (6) 当在轴名称或第2辅助功能中，使用了地址A时，请将“A”作为角度使用。
- (7) 本功能仅对G01指令有效，对于其他插补及定位无效。
- (8) 斜率a范围为  $-360.000 \leq a \leq 360.000$  。  
当进行了超过该范围的指令时，以除以360（°）后的余数进行指令。  
（例）当指令了400时，400/360的余数40° 成为指令角度。
- (9) 当在一节中同时存在“A”与“A”时，“A”被看作为角度。

13.13 几何学

13.13.1 几何学 I



功能及目的

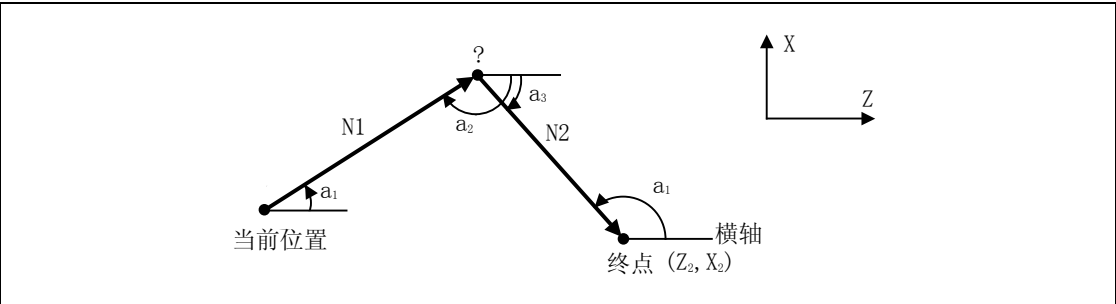
在连续直线插补指令中，当难以计算2直线的交点时，如果指令第1直线的斜率，与第2直线的终点绝对坐标值及斜率，则在NC内部自动计算第1直线的终点，并控制移动指令。

（注）当参数（#1082 Geomet）为 0 时，几何学 I 不生效。



指令格式

|                                                                                          |                    |
|------------------------------------------------------------------------------------------|--------------------|
| <b>N1 G01 Aa<sub>1</sub> (A—a<sub>2</sub>) Ff<sub>1</sub>;</b>                           | 指定角度与速度。           |
| <b>N2 Xx<sub>2</sub> Zz<sub>2</sub> A—a<sub>2</sub> (Aa<sub>3</sub>) Ff<sub>2</sub>;</b> | 指定下一节终点绝对坐标与角度及速度。 |
| Aa <sub>1</sub> , A—a <sub>2</sub> , Aa <sub>3</sub>                                     | : 角度               |
| Ff <sub>1</sub> , Ff <sub>2</sub>                                                        | : 速度               |
| Xx <sub>2</sub> , Zz <sub>2</sub>                                                        | : 下一节终点的绝对坐标       |



详细说明

- (1) 选择平面中没有的几何学指令，会导致程序错误（P396）。
- (2) 斜率表示与选择平面的横轴+方向所构成的角度，逆时针方向（CCW）为+，顺时针方向（CW）为-。
- (3) 斜率a范围为  $-360.000 \leq a \leq 360.000$  。  
当进行了超过该范围的指令时，以除以360（°）后的余数进行指令。  
（例）当指令了400时，400/360的余数40° 成为指令角度。
- (4) 可在起点、终点任意一端指令直线斜率。所指定的斜率是起点端还是终点端，在NC内部自动进行判别。
- (5) 请使用绝对坐标指令第2节的终点坐标。当采用增量值时，发生程序错误（P393）。
- (6) 可对各节分别指令速度。
- (7) 当2直线的交角在1° 以下时，发生工序错误（P392）。
- (8) 当在第1节与第2节中进行平面切换时，发生程序错误（P396）。

- (9) 当在轴名称或第2辅助功能中使用了地址A时，本功能被跳跃。
- (10) 可在第1节的终点进行单节停止。
- (11) 当第1节及第2节不是G01或G33时，发生程序错误（P394）。

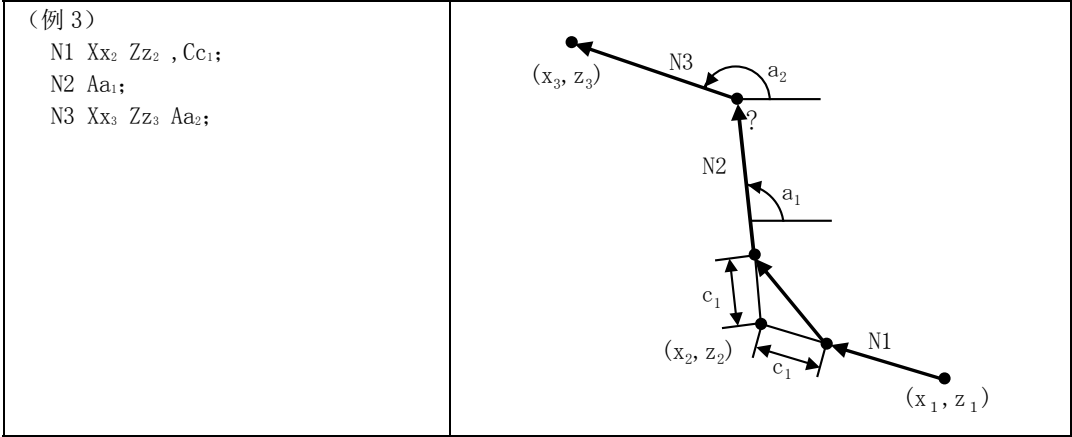


与其他功能的关联

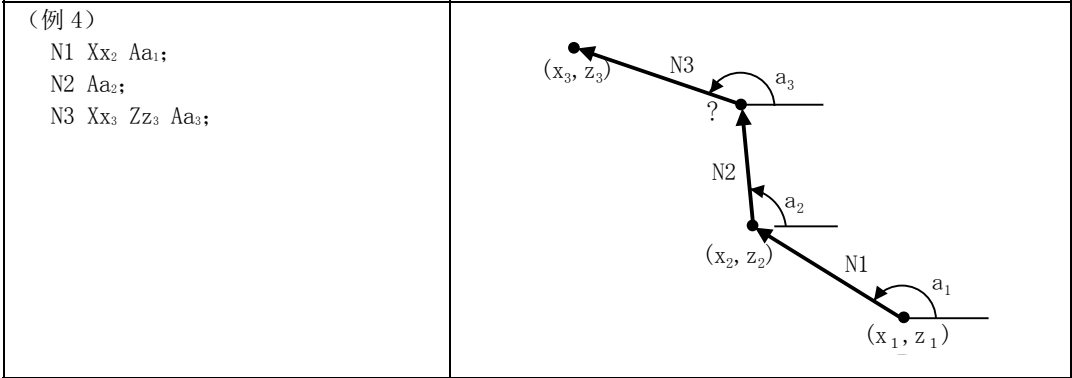
- (1) 在第1节中，可在角度指令后进行倒角、圆角指令。

|                                                                                                                            |  |
|----------------------------------------------------------------------------------------------------------------------------|--|
| <div>(例 1)</div> <div>N1 Aa<sub>1</sub> ,Cc<sub>1</sub>;</div> <div>N2 Xx<sub>2</sub> Zz<sub>2</sub> Aa<sub>2</sub>;</div> |  |
| <div>(例 2)</div> <div>N1 Aa<sub>1</sub> ,Rr<sub>1</sub>;</div> <div>N2 Xx<sub>2</sub> Zz<sub>2</sub> Aa<sub>2</sub>;</div> |  |

(2) 可在倒角、圆角指令之后，指令几何学指令 I。



(3) 可在直线角度指令之后，指令几何学指令 I。

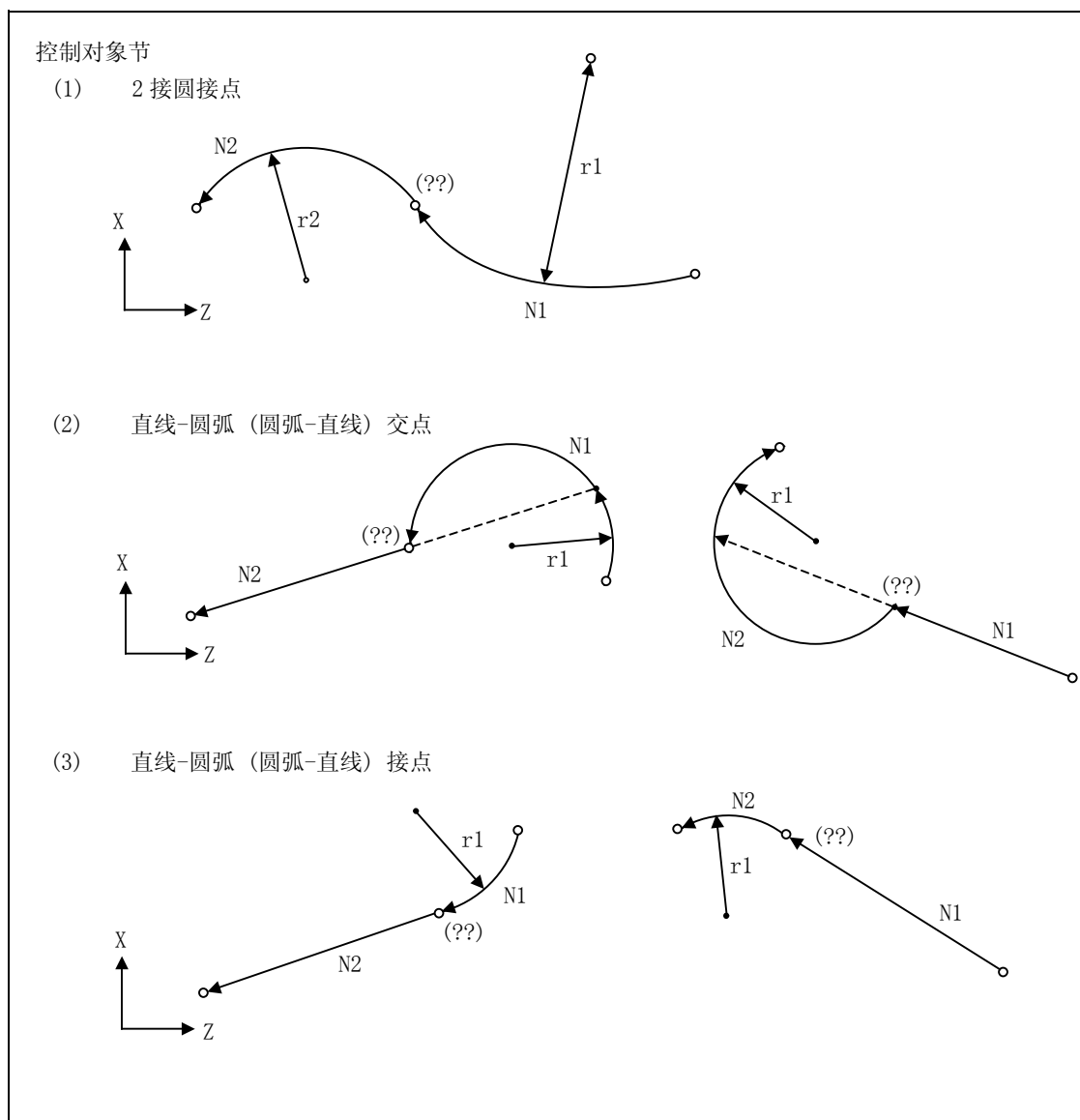


## 13.13.2 几何学 I B



## 功能及目的

几何学 I B，是在连续2节移动指令（仅限于含圆弧指令的节）中，通过指令圆弧中心点或直线角度代替最初节的终点，计算接点、交点。



（注）当参数（#1082 Geomet）为 2 时，几何学 I B 不生效。



功能及目的 1 (2 接点自动计算)

连续2个圆弧相接，但是其接点未在图纸上写出时，通过指定第1圆弧的中心坐标值或半径，与第2圆弧的终点绝对值与中心坐标值或是半径，自动计算出接点。



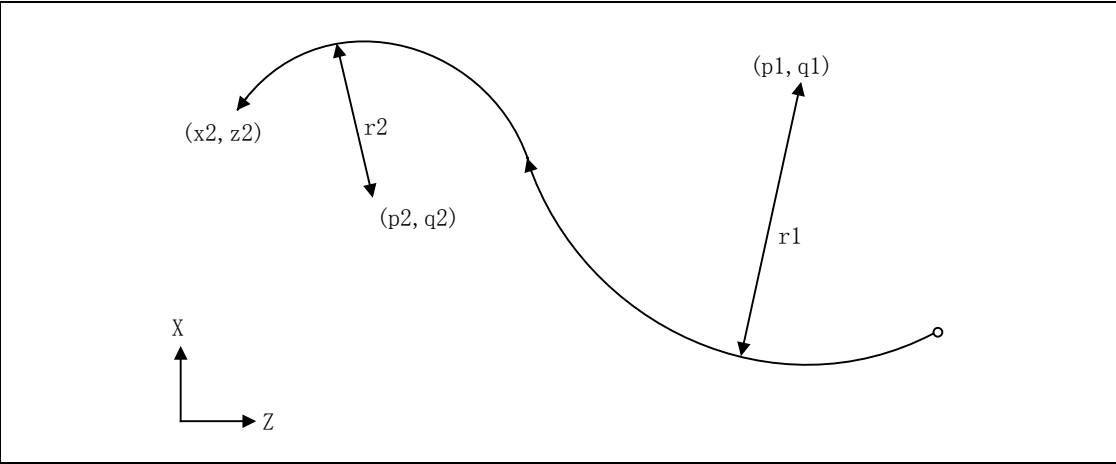
指令格式 1 (2 接点自动计算)

|              |                                                                                   |
|--------------|-----------------------------------------------------------------------------------|
| N1 G02 (G03) | Pp <sub>1</sub> Qq <sub>1</sub> Ff <sub>1</sub> ;                                 |
| N2 G03 (G02) | Xx <sub>2</sub> Zz <sub>2</sub> Pp <sub>2</sub> Qq <sub>2</sub> Ff <sub>2</sub> ; |
| N1 G02 (G03) | Pp <sub>1</sub> Qq <sub>1</sub> Ff <sub>1</sub> ;                                 |
| N2 G03 (G02) | Xx <sub>2</sub> Zz <sub>2</sub> Rr <sub>2</sub> Ff <sub>2</sub> ;                 |
| N1 G02 (G03) | Rr <sub>1</sub> Ff <sub>1</sub> ;                                                 |
| N2 G03 (G02) | Xx <sub>2</sub> Zz <sub>2</sub> Pp <sub>2</sub> Qq <sub>2</sub> Ff <sub>2</sub> ; |

P, Q : X、Z 轴圆弧中心坐标绝对值（直径/半径值指令）  
通过 A 指令第 3 轴的中心地址。  
R : 圆弧半径（带（-）符号，则判断为 180° 以上的圆弧）

※可指令I、K（X、Z轴圆弧中心坐标增量值）代替P、Q。

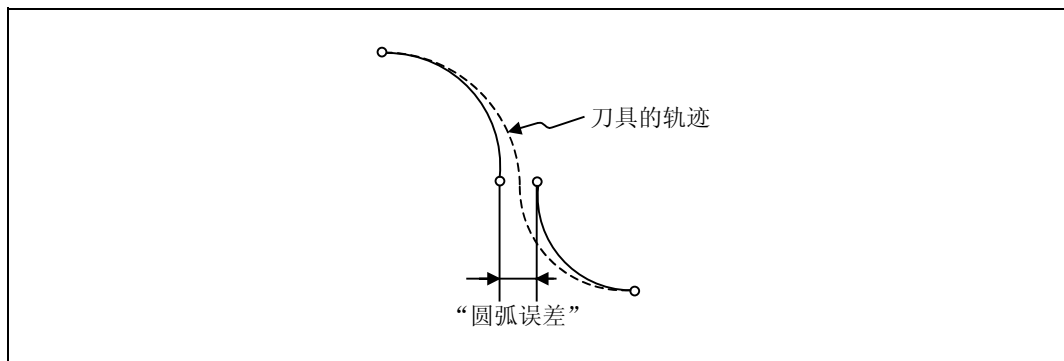
[ 第1节的圆弧时，为起点到中心的半径指令增量值  
第2节的圆弧时，为终点到中心的半径指令增量值 ]



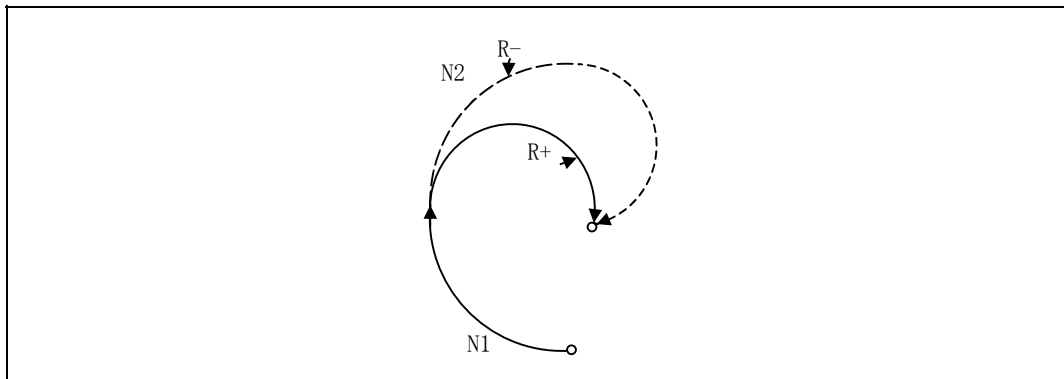


## 详细说明 1 (2 接点自动计算)

- (1) 如果第2节不是坐标绝对值指令，则在第1节前发生程序错误 (P393)。
- (2) 如果没有几何学 I B 规格，则在第1节之前发生程序错误 (P398)。
- (3) 如果没有在第2节中指定R (此时，第1节为P、Q < I、K) 或P、Q (I、K)，则在第1节之前发生程序错误 (P395)。
- (4) 如果在第2节中进行其他的平面选择指令 (G17~G19)，则在第1节之前发生程序错误 (P396)。
- (5) 如果进行2圆不相接的指令，则在第1节之前发生程序错误 (P397)。
- (6) 接点计算精度为 $\pm 1 \mu\text{m}$  (四舍五入)。
- (7) 如果进行单节运转，则在第1节停止。
- (8) 如果省略I或K，则看作为I0或K0。P、Q不可省略。
- (9) 计算接点的误差范围，取决于参数“#1084 RadErr”



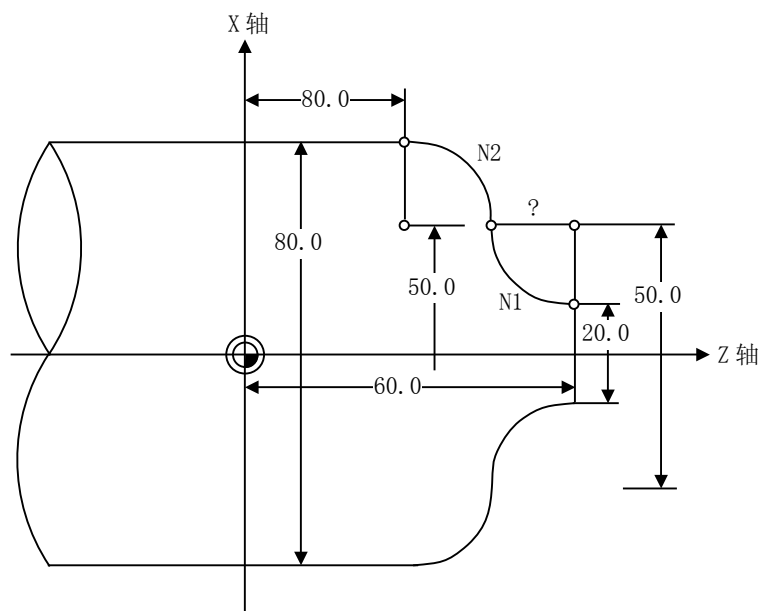
- (10) 圆弧节真圆指令 (圆弧节起点=圆弧节终点) 时，R指定圆弧指令会立即完成，不进行任何动作，所以，请使用PQ (IK) 指定圆弧指令。
- (11) 第1/第2节的G模式组1G代码可省略。
- (12) 作为轴名称使用的地址，不可作为圆弧中心坐标、圆弧半径的指令地址使用。
- (13) 当第2节圆弧内接于第1节圆弧时，当第2节为R指定圆弧时，若R的符号为正，则为内旋圆弧指令，如果R的符号为负，则为外旋圆弧指令。





程序例 1 (2 接点自动计算)

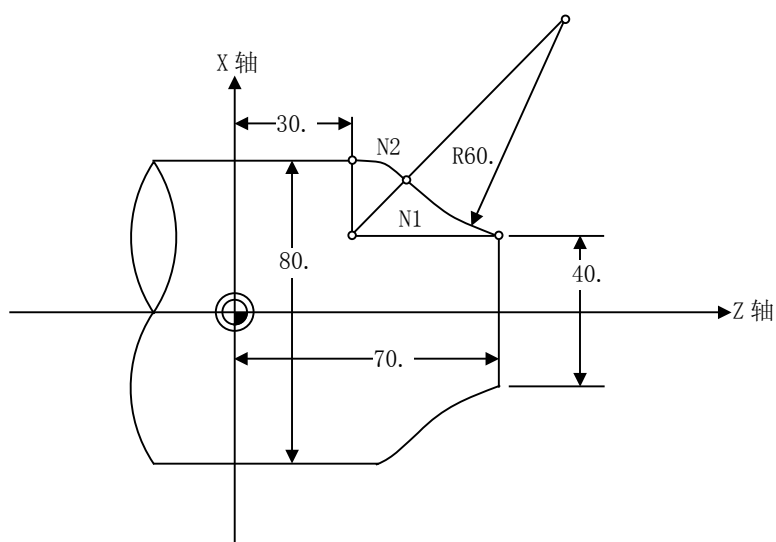
(1) PQ、PQ指令



```
G01 X20.0 Z60.0;
N1 G02 P50.0 Q60.0 F100;
N2 G03 X80.0 Z30.0 P50.0 Q30.0;
```

(单位 mm)

(2) PQ、R指令



```
G01 X40.0 Z70.0 F100;
N1 G02 R60.0;
N2 G03 X80.0 Z30.0 P40.0 Q30.0;
```

(单位 mm)



与其他功能的关联 1 (2 接点自动计算)

| 指 令                                                                            | 刀具的轨迹 |
|--------------------------------------------------------------------------------|-------|
| 几何学 I B+倒角 II<br><br>N1 G09 P_ Q_ ;<br>N2 G02 X_ Z_ R_ ,C_ ;<br>G02 X_ Z_ R_ ; |       |
| 几何学 I B+圆角 II<br><br>N1 G03 P_ Q_ ;<br>N2 G02 X_ Z_ R_ ,R_ ;<br>G02 X_ Z_ R_ ; |       |
| 几何学 I B+倒角 II<br><br>N1 G03 P_ Q_ ;<br>N2 G02 X_ Z_ R_ ,C_ ;<br>G01 X_ Z_ ;    |       |
| 几何学 I B+圆角 II<br><br>N1 G03 P_ Q_ ;<br>N2 G02 X_ Z_ R_ ,R_ ;<br>G01 X_ Z_ ;    |       |



功能及目的 2（直线-圆弧交点自动计算）

直线与圆弧相交的形状中，当图纸上未标明其交点时，通过进行如下的程序指令，自动计算交点。

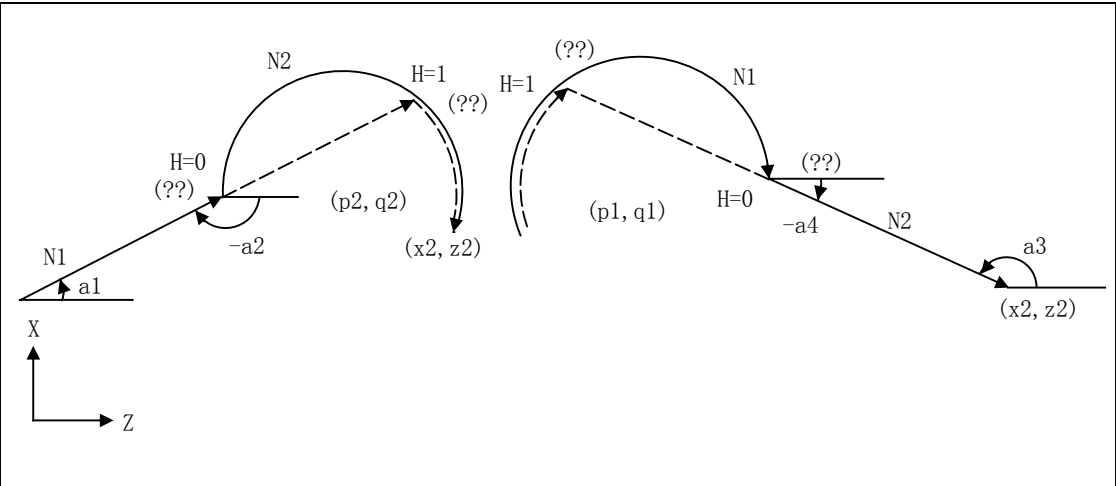


指令格式 2（直线-圆弧交点自动计算：G18 平面时）

|      |                                                    |                                                                                 |                   |
|------|----------------------------------------------------|---------------------------------------------------------------------------------|-------------------|
| N1   | G01                                                | Aa <sub>1</sub> (A-a <sub>2</sub> )                                             | Ff <sub>1</sub> ; |
| N2   | G02(G03)                                           | Xx <sub>2</sub> Zz <sub>2</sub> Pp <sub>2</sub> Qq <sub>2</sub> Hh <sub>2</sub> | Ff <sub>2</sub> ; |
| N1   | G02(G03)                                           | Pp <sub>1</sub> Qq <sub>1</sub> Hh <sub>1</sub>                                 | Ff <sub>1</sub> ; |
| N2   | G1                                                 | Xx <sub>2</sub> Zz <sub>2</sub> Aa <sub>3</sub> (A-a <sub>4</sub> )             | Ff <sub>2</sub> ; |
| A    | : 直线的角度 ( -360.000° ~ 360.000° )                   |                                                                                 |                   |
| P, Q | : X、Z 圆弧中心坐标绝对值 ( 直径/半径值指令 )<br>通过 A 指令第 3 轴的中心地址。 |                                                                                 |                   |
| H    | : 直线-圆弧的交点选择                                       |                                                                                 |                   |
|      |                                                    | [ 0: 直线较短时的交点<br>1: 直线较长时的交点 ]                                                  |                   |

※可指令 I、K (X、Z 轴圆弧中心坐标增量值) 代替 P、Q。

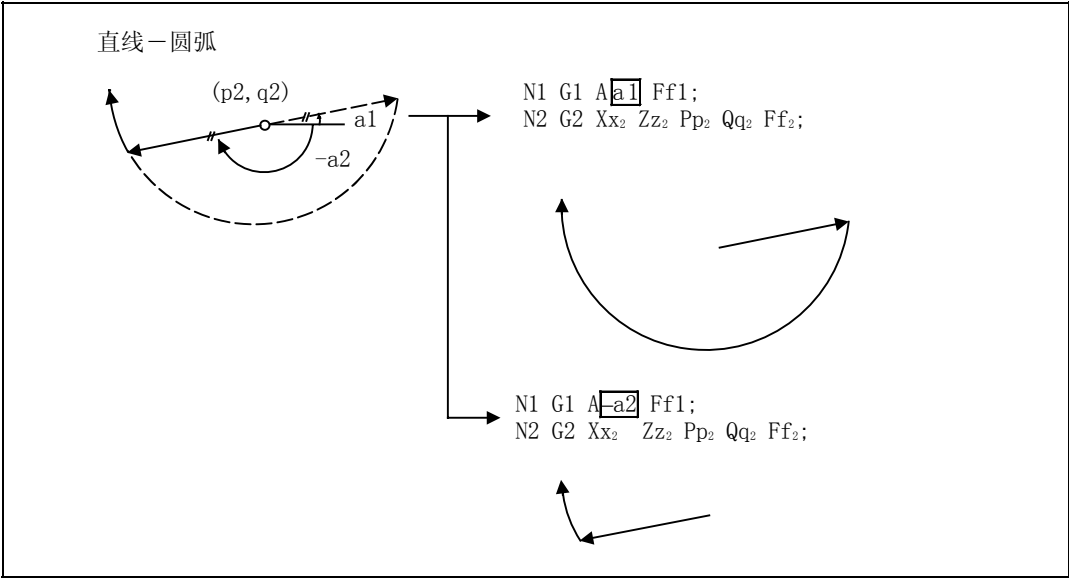
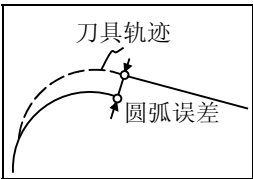
[ 第1节的圆弧时，为起点到中心的半径指令增量值  
第2节的圆弧时，为终点到中心的半径指令增量值 ]





详细说明 2（直线-圆弧交点自动计算）

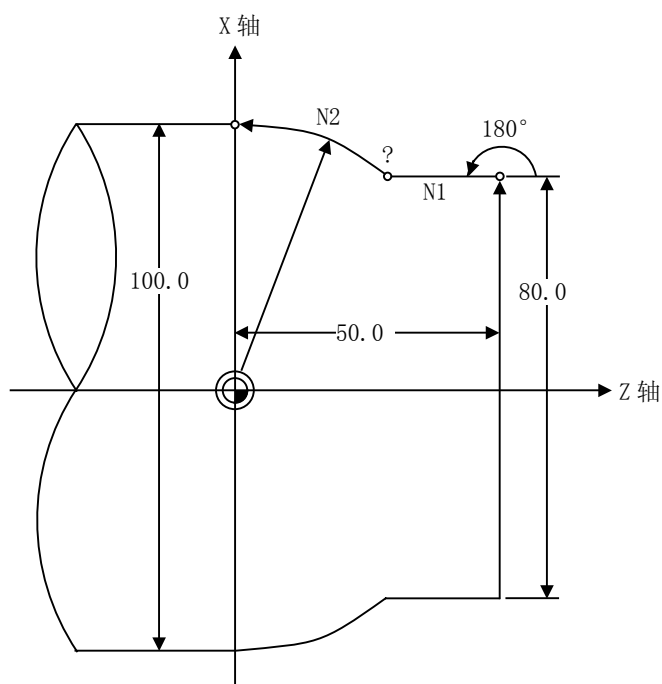
- (1) 当第2辅助功能的地址为A时，第2辅助功能有效，本功能无效。
- (2) 如果第2节不是坐标绝对值指令，则在第1节前发生程序错误（P393）。
- (3) 如果没有几何学 I B规格，则在第1节之前发生程序错误（P398）。
- (4) 第2节为圆弧指令时，如果没有指定P、Q（I、K），则在第1节之前发生程序错误（P395）。另外，当第2节为直线指令时，如果没有A指令，则发生程序错误（P395）。
- (5) 如果在第2节中进行其他的平面选择指令（G17~G19），则在第1节之前发生程序错误（P396）。
- (6) 当直线与圆弧未相接或相交时，在第1节之前发生程序错误（P397）。
- (7) 如果进行单节运转，则在第1节停止。
- (8) 如果省略I或K，则看作为I0或K0。P、Q不可省略。
- (9) 省略H，则看作为H0。
- (10) 使用R指定代替P、Q（I、K）指定，则进行直线-圆弧接点自动计算。
- (11) 计算交点的误差范围，取决于参数“#1084 RadErr”。
- (12) 直线的斜率表示与横轴所构成的角度，逆时针方向（CCW）为正（+），顺时针方向（CW）为负（-）。
- (13) 可从起点、终点任意一端开始，指令直线斜率。所指定的斜率是起点端还是终点端，自动进行判别。
- (14) 当到直线与圆弧交点的距离相同时（下图），无法通过H（距离长短选择）进行控制。此时，根据直线的角度进行判断。



- (15) 交点计算精度为 $\pm 1 \mu\text{m}$  (四舍五入)。
- (16) 在直线-圆弧交点, 圆弧指令仅为PQ (IK) 指令, 当圆弧节起点=圆弧节终点时, 圆弧为正圆。
- (17) 第1节的G模态组G代码可省略。
- (18) 作为轴名称使用的地址, 不可作为角度、圆弧中心坐标、交点选择的指令地址使用。
- (19) 指令了几何学 I B时, 预读2节。



### 程序例 2 (直线-圆弧交点自动计算)



G01 X80.0 Z50.0 F100;  
N1 G01 A180.0;  
N2 G03 X100.0 Z0 P0 Q0; (单位 mm)



与其他功能的关联 2（直线-圆弧交点自动计算）

| 指 令                                                                        | 刀具的轨迹 |
|----------------------------------------------------------------------------|-------|
| 几何学 I B+倒角 II<br>N1 G01 A_ ,C_ ;<br>N2 G03 X_ Z_ P_ Q_ H_ ;                |       |
| 几何学 I B+圆角 II<br>N1 G01 A_ ,R_ ;<br>N2 G03 X_ Z_ P_ Q_ H_ ;                |       |
| 几何学 I B+倒角 II<br>N1 G01 A_ ;<br>N2 G03 X_ Z_ P_ Q_ H_ ;<br>G01 X_ Z_ ;     |       |
| 几何学 I B+圆角 II<br>N1 G01 A_ ;<br>N2 G03 X_ Z_ P_ Q_ H_ ;<br>G01 X_ Z_ ;     |       |
| 几何学 I B+倒角<br>N1 G02 P_ Q_ H_ ;<br>N2 G01 X_ Z_ A_ ,C_ ;<br>G01 X_ Z_ ;    |       |
| 几何学 I B+圆角 II<br>N1 G02 P_ Q_ H_ ;<br>N2 G01 X_ Z_ A_ ,R_ ;<br>G01 X_ Z_ ; |       |



功能及目的 3（直线-圆弧接点自动计算）

直线与圆弧相接的形状中，当图纸上未标明其接点时，通过进行如下的程序指令，自动计算交点。

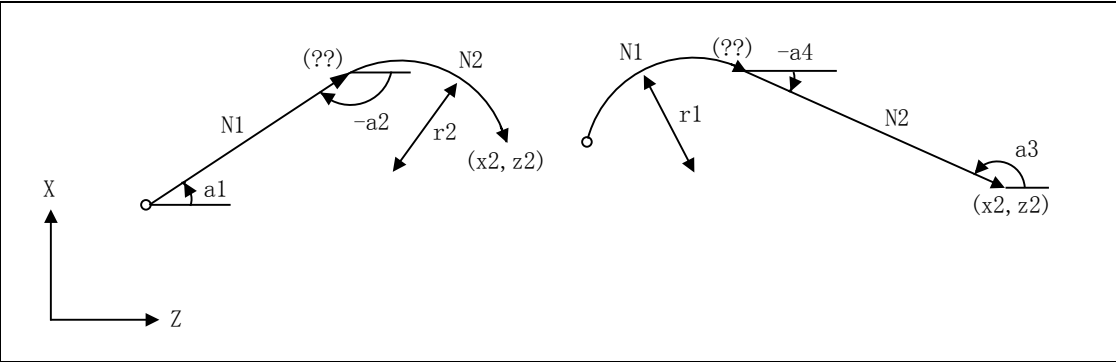


指令格式 3（直线-圆弧接点自动计算：G18 平面时）

```
N1 G01 Aa1(A-a2) Ff1;
N2 G03(G02) Xx2 Zz2 Rr2 Ff2;

N1 G03(G02) Rr1 Ff1;
N2 G01 Xx2 Zz2 Aa3(A-a4)Ff2;

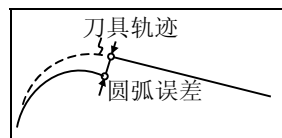
A : 直线的角度 (-360.000° ~ 360.000°)
R : 圆弧半径
```





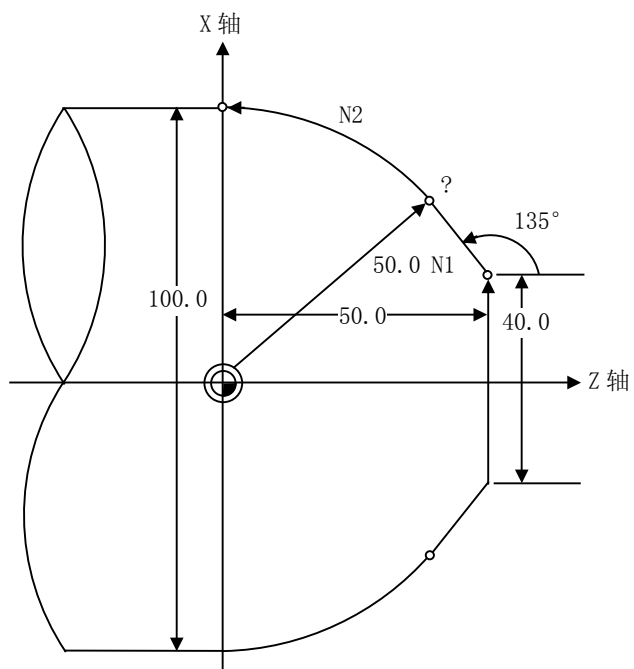
## 详细说明 3 (直线-圆弧接点自动计算)

- (1) 当第2辅助功能的地址为A时，第2辅助功能有效，本功能无效。
- (2) 如果第2节不是坐标绝对值指令，则在第1节前发生程序错误 (P393)。
- (3) 如果没有几何学 I B规格，则在第1节之前发生程序错误 (P398)。
- (4) 如果在第2节中进行其他的平面选择指令 (G17~G19)，则在第1节之前发生程序错误 (P396)。
- (5) 当直线与圆弧未相接时，在第1节之前发生程序错误 (P397)。
- (6) 第2节为圆弧指令时，如果没有指定R，则在第1节之前发生程序错误 (P395)。另外，直线指令时，如果没有A指令，则发生程序错误 (P395)。
- (7) 如果进行单节运转，则在第1节停止。
- (8) 使用P、Q (I、K) 指定代替R指定，则进行直线-圆弧交点自动计算。
- (9) 计算接点的误差范围，取决于参数 “#1084 RadErr”
- (10) 直线的斜率表示与横轴+方向所构成的角度，逆时针方向 (CCW) 为正 (+)，顺时针方向 (CW) 为负 (-)。
- (11) 可在起点、终点任意一端指令直线斜率。所指定的斜率是起点端还是终点端，自动进行判别。
- (12) 交点计算精度为 $\pm 1 \mu m$  (四舍五入)。
- (13) 在直线-圆弧接点，圆弧指令仅为R指令，当圆弧节起点=圆弧节终点时，圆弧指令立即完成，不进行动作。(不可进行正圆指令。)
- (14) 第1节的G模态组1G代码可省略。
- (15) 作为轴名称使用的地址，不可作为角度、圆弧半径的指令地址使用。
- (16) 指令了几何学 I B时，预读2节。





程序例 3 (直线-圆弧接点自动计算)



```
G01 X40.0 Z50.0 F100;
```

```
N1 G01 A135.0;
```

```
N2 G03 X100.0 Z0.0 R50.0;
```

(单位 mm)



与其他功能的关联 3（直线-圆弧接点自动计算）

| 指 令                                                                      | 刀具的轨迹 |
|--------------------------------------------------------------------------|-------|
| 几何学 I B+倒角<br><br>N1 G03 R_ ;<br>N2 G01 X_ Z_ A_ ,C_ ;<br>G01 X_ Z_ ;    |       |
| 几何学 I B+圆角<br><br>N1 G03 R_ ;<br>N2 G01 X_ Z_ A_ ,R_ ;<br>G01 X_ Z_ ;    |       |
| 几何学 I B+倒角 II<br><br>N1 G01 A_ ;<br>N2 G02 X_ Z_ R_ ,C_ ;<br>G01 X_ Z_ ; |       |
| 几何学 I B+圆角 II<br><br>N1 G01 A_ ;<br>N2 G02 X_ Z_ R_ ,R_ ;<br>G01 X_ Z_ ; |       |

13.14 程序参数输入；G10、G11



功能及目的

可通过设定显示装置，利用加工程序变更所设定的参数。  
数据设定的数据格式如下。



指令格式

**G10L50； 数据设定指令**

P大区分编号

N数据编号

H□二进制型参数

；

P大区分编号

A轴编号

N数据编号

D字节型参数

；

P大区分编号

A轴编号

N数据编号

S字符型参数

；

P大区分编号

A轴编号

N数据编号

L 2 字符型参数

；

**G11； 数据设定模式取消（数据设定完成）**

数据部分格式根据参数的种类（轴通用/轴独立）及数据型式，有以下8种类型。

|             |       |                        |
|-------------|-------|------------------------|
| 对于轴通用数据     |       |                        |
| 轴通用二进制型参数   | ..... | P____N____H□____；      |
| 轴通用字节型参数    | ..... | P____N____D____；       |
| 轴通用字符型参数    | ..... | P____N____S____；       |
| 轴通用 2 字符型参数 | ..... | P____N____L____；       |
| 对于轴独立数据     |       |                        |
| 轴独立二进制型参数   | ..... | P____A____N____H□____； |
| 轴独立字节型参数    | ..... | P____A____N____D____；  |
| 轴独立字符型参数    | ..... | P____A____N____S____；  |
| 轴独立 2 字符型参数 | ..... | P____A____N____L____；  |

- （注 1）1 节内，各地址的顺序必须与以上相同。
- （注 2）P、N 编号对应表请参阅附录 1。
- （注 3）二进制型参数时，数据类型为 H□（□为 0~7 的数字）。
- （注 4）轴编号是将第 1 轴设定为“1”、第 2 轴设定为“2”……以此类推。



程序例

（例）二进制选择 #6401的bit2打开时

G10 L50 ；  
P8 N1 H21 ；  
G11 ；

13.15 宏插入



功能及目的

所谓用户宏插入功能，是在程序执行中，通过从机械端输入用户宏插入信号（UIT），在当前执行的程序中，优先调用其他程序的功能。

通过使用本功能，可进行符合变化后的状况的程序动作。本功能的相关参数设定方法，请参阅警报/参数说明书。



指令格式

|                    |          |
|--------------------|----------|
| <b>M96 P_ H_ ;</b> | 用户宏插入有效  |
| <b>M97 ;</b>       | 用户宏插入无效  |
| P                  | : 插入程序编号 |
| H                  | : 插入顺序编号 |

用户宏插入功能通过在程序中运用M96、M97指令，将插入信号（UIT）在有效状态、无效状态间进行切换。亦即，在指令M96之后，则在指令M97之前，或是重新启动之前，用户宏插入有效，在此期间，只要从机械端输入插入信号（UIT），则用户宏插入被起动，通过P\_指令的程序插入到当前执行中的程序中，被执行。用户宏插入中，或是M97指令之后及重新启动之后的用户宏插入无效状态下输入的插入信号（UIT），在有M96指令之前，被跳跃。

M96、M97作为用户宏插入控制M代码，被予以内部处理。



有效条件

用户宏插入仅能在执行程序时加以执行。

因此，有效条件如下。

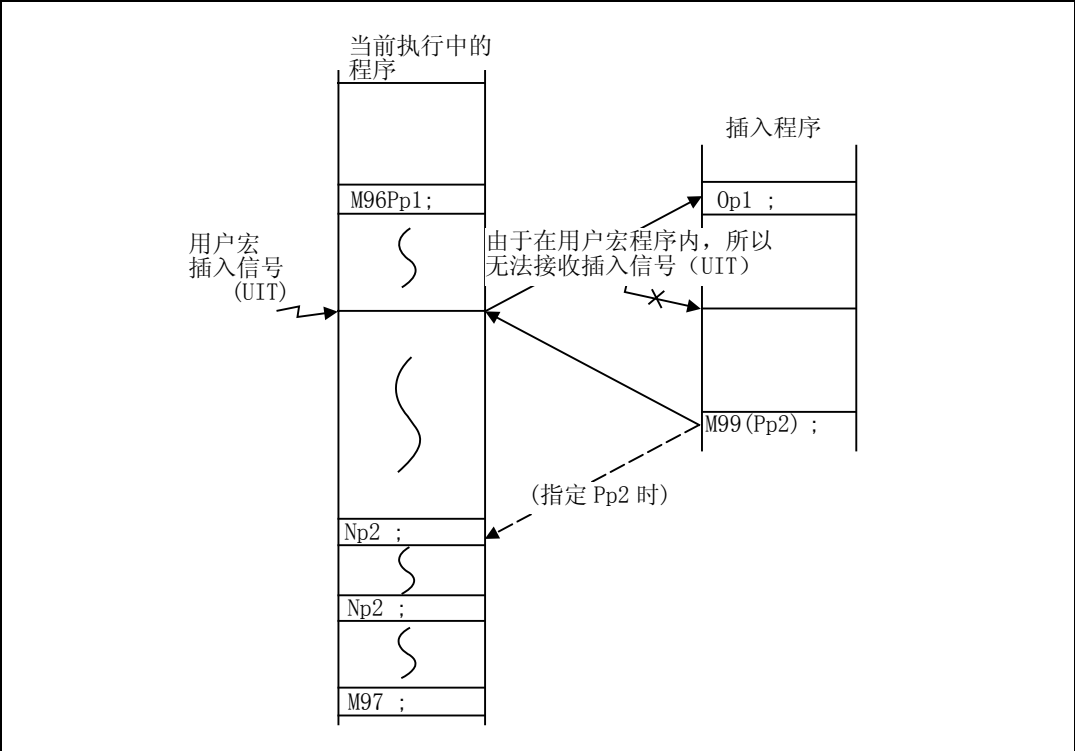
- （1）选择了内存、纸带中的任何一种自动运转模式，或MDI。
- （2）处于自动运转中
- （3）不处于用户宏插入处理中。

（注1）手动运转中（慢进、分步、手轮……等），宏插入无效。



动作概要

- (1) 在当前执行中的程序中指令了M96 Pp1；指令之后，输入用户宏插入信号（UIT），则执行插入程序0p1，通过插入程序内的M99；指令返回原来的程序。
- (2) 通过M99 Pp2；返回时，呼叫从被插入节的下一节到程序的最后节，如果没有呼叫到，则从程序的开头呼叫到被插入节的前一节，返回到首个出现的顺序编号为Np2；的节。





## 插入方式

插入方式中，包括类型1与类型2，通过参数“#1113 INT\_2”加以选择。

## [类型1]

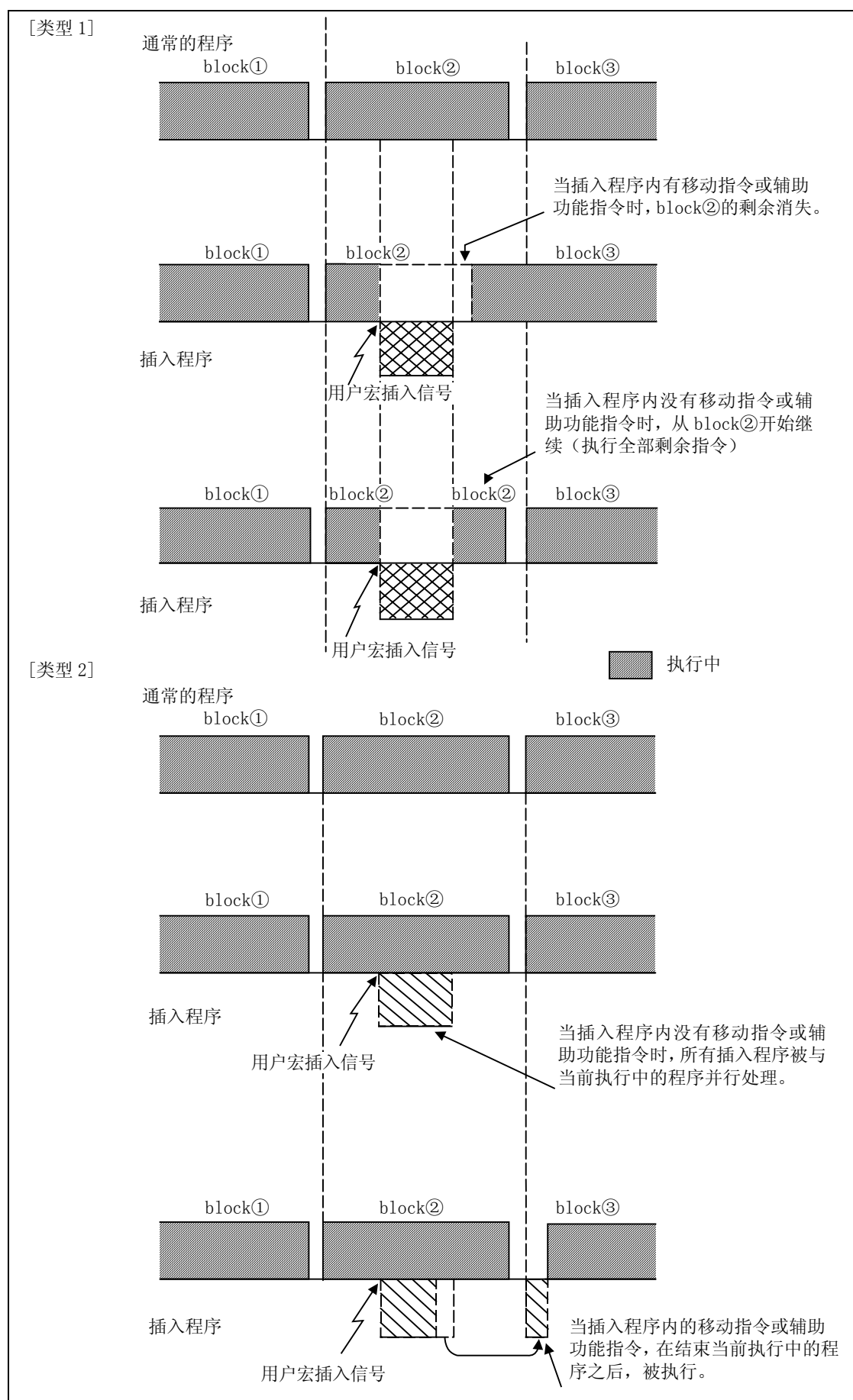
- 输入插入信号（UIT），则立即中断执行中的移动或延时，执行插入程序。
- 如果在插入程序内有移动指令或辅助功能指令（MSTB），则中断节的指令消失，执行插入程序。插入程序完成，则从中断节的下一节开始，继续执行。
- 如果插入程序内没有移动指令或辅助功能指令（MSTB），则从插入程序返回后，从中断节的中断处开始，继续执行。

但是，当在执行辅助功能指令（MSTB）时输入插入信号（UIT）时，由于NC进入完成信号（FIN）等待状态，所以是在输入FIN之后，执行插入程序内的移动指令或辅助功能指令（MSTB）。

## [类型2]

- 输入插入信号（UIT），则在结束执行中的节指令之后，执行插入程序。
- 如果插入程序内没有移动指令及辅助功能指令（MSTB），则不中断执行中的程序，执行插入程序。

但是，当即使原来的节结束了，插入程序也没有结束时，可能会暂时停止加工。





调用方式

用户宏插入中，根据插入程序的调用方法不同，有以下2种，通过参数“#1229 set01/bit0”加以选择。选择任何1种的情况下，均进行调用嵌套层数的累加。另外，插入程序内进行的子程序及用户宏调用，也分别被累加到嵌套层数中。

- a. 子程序型插入
- b. 宏型插入

|        |                                                                        |
|--------|------------------------------------------------------------------------|
| 子程序型插入 | 作为子程序，调用用户宏插入程序。（与 M98 调用相同）即，插入前后，局部变量的等级不发生变化。                       |
| 宏型插入   | 作为用户宏，调用用户宏插入程序。（与 G65 调用相同）即，插入前后，局部变量的等级发生变化。<br>另外，执行程序不向插入程序转移自变量。 |

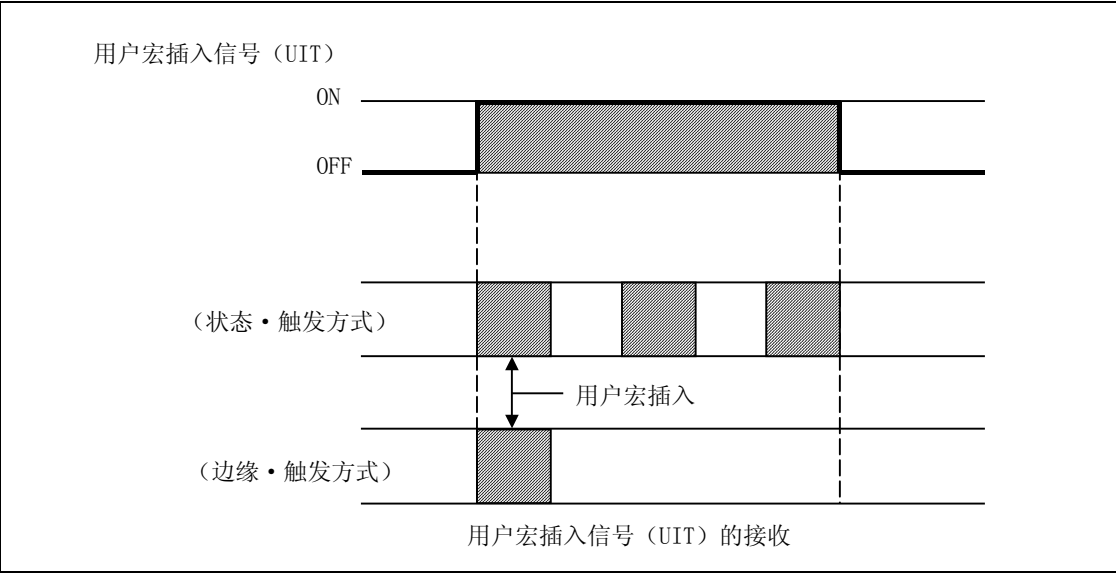


用户宏插入信号（UIT）的接收方式

用户宏插入信号（UIT）的接收方式有以下2种，通过参数“#1112 S\_TRG”加以选择。

- a. 状态・触发方式
- b. 边缘・触发方式

|         |                                                                                                                    |
|---------|--------------------------------------------------------------------------------------------------------------------|
| 状态・触发方式 | 用户宏插入信号（UIT）为 ON 的状态下，作为令信号生效的内容加以接收。<br>通过 M96 让用户宏插入有效时，如果插入信号（UIT）为 ON，则执行插入程序。<br>通过让插入信号（UIT）保持 ON，可反复执行插入程序。 |
| 边缘・触发方式 | 用户宏插入信号（UIT）从 OFF 到 ON，进行起动时，作为令信号生效的内容加以接收。<br>当只想执行 1 次插入程序时，可使用本方式。                                             |





#### 从用户宏插入信号的返回

M99 (P\_) ;

通过在插入程序内指令M99，从用户宏插入返回到原来的程序。

可通过地址P，指定返回程序内的顺序编号。

此时，呼叫从被插入节的下一节到程序的最后节，如果没有呼叫到，则从程序的开头呼叫到被插入节的前一节，返回到首个出现的指定顺序编号的节。

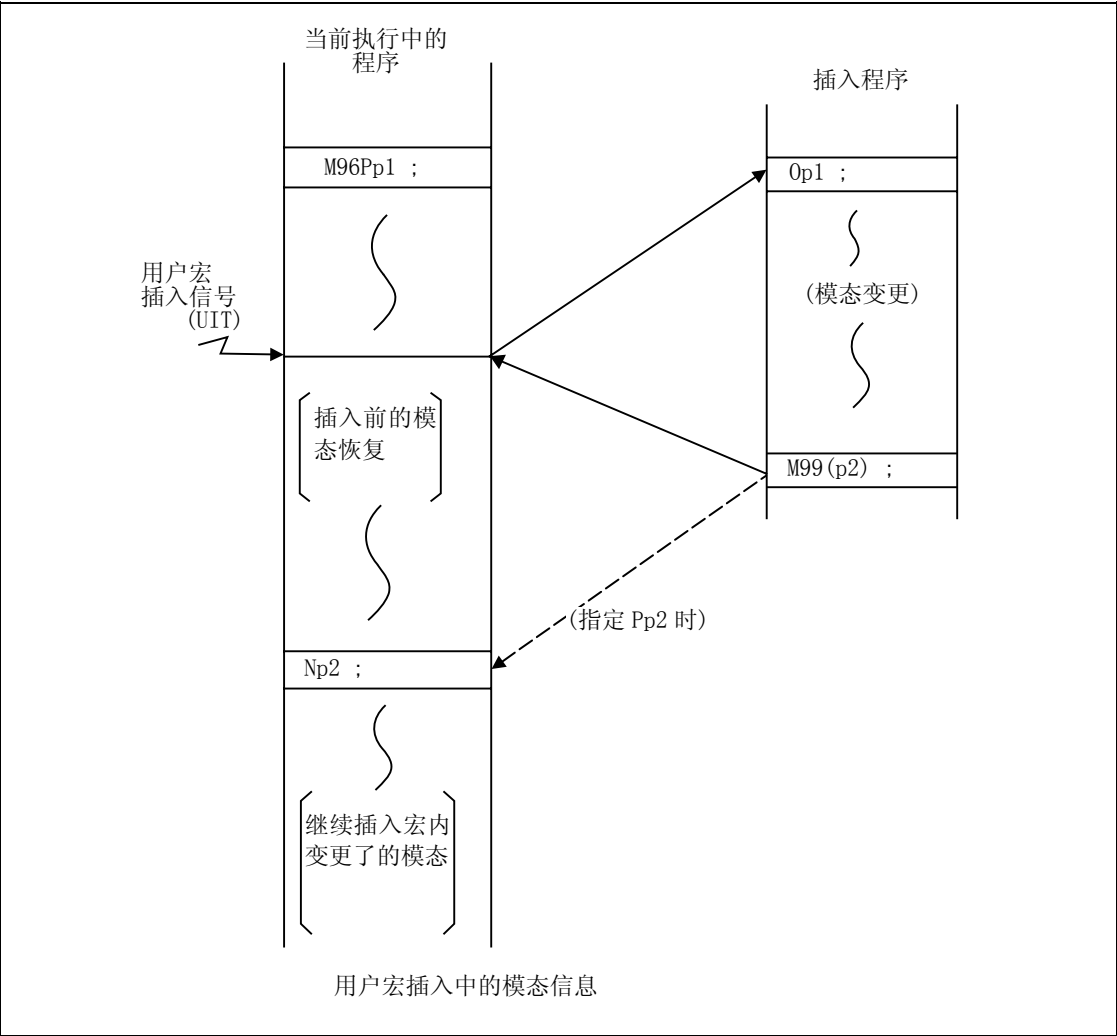
（与M98调用的M99 P\_相同）



用户宏插入中的模态信息

在插入程序内变更了模态信息时，从插入程序返回后的模态信息如下。

|              |                                                                                         |
|--------------|-----------------------------------------------------------------------------------------|
| 通过 M99；返回时   | 插入程序内变更的模态信息无效，返回到插入前的模态信息。<br>但是，当插入方式为类型 1 时，如果在插入程序内有移动指令或辅助功能指令（MSTB），则不返回插入前的模态信息。 |
| 通过 M99P_；返回时 | 当在插入程序内变更了模态信息时，即使从插入程序返回之后，插入程序内变更了的模态信息也被继续保持。当通过 M99P_；从通过 M98 等调用的程序中返回时，也相同。       |





模态信息变量（#4401～#4520）

可通过读取#4401～#4520的值，识别用户宏插入程序控制改变时的模态信息。  
单位为指令时的单位。

| 系统变量  | 模态信息       | } | 有未使用的组。 |
|-------|------------|---|---------|
| #4401 | G 代码（组 01） |   |         |
| {     | {          |   |         |
| #4421 | G 代码（组 21） | } | 有未使用的组。 |
| #4507 | D 代码       |   |         |
| #4509 | F 代码       |   |         |
| #4511 | H 代码       | } | 有未使用的组。 |
| #4513 | M 代码       |   |         |
| #4514 | 顺序编号       |   |         |
| #4515 | 程序编号       | } | 有未使用的组。 |
| #4519 | S 代码       |   |         |
| #4520 | T 代码       |   |         |

本变量仅能在用户宏插入程序内使用。  
当在用户宏插入程序以外使用时，发生程序错误（P241）。



用户宏插入控制用 M 代码

用户宏插入是通过M96、M97进行控制，当M96、M97已被用于其他用途时，可使用其他M代码代替。  
（程序失去兼容性。）  
可在参数“#1110 M96\_M”、“#1111 M97\_M”中设定代替M代码，并通过选择用于激活代替M代码的参数“#1109 subs\_M”，使用代替M代码进行用户宏插入控制。  
（但是，M代码的设定范围为03～97，除30以外。）  
当未选择用于激活代替M代码的参数“#1109 subs\_M”时，M96、M97为用户宏插入控制用M代码。  
在任何场合下，用户宏插入控制用M代码均为内部处理，不会被输出。



## 参数的种类

设定方法请参阅操作说明书。

- (1) 子程序型调用有效 “#1229 set01/bit0”
  - 1: 子程序型用户宏插入
  - 0: 宏程序型用户宏插入
- (2) 状态・触发方式有效 “#1112 S\_TRG”
  - 1: 状态・触发方式
  - 0: 边缘・触发方式
- (3) 插入方式类型2有效 “#1113 INT\_2”
  - 1: 等待节执行完成之后, 执行插入程序内的执行语句的方式 (类型2)
  - 0: 不等待节执行完成, 就执行插入程序内的执行语句的方式 (类型1)
- (4) 用户宏插入控制用代替M代码有效 “#1109 subs\_M”
  - 1: 有效
  - 0: 无效
- (5) 用户宏插入控制用代替M代码有效
  - 插入有效M代码 (相当于M96) “#1110 M96\_M”
  - 插入无效M代码 (相当于M97) “#1111 M97\_M”



## 限制事项

- (1) 为了在用户宏插入程序内读取坐标值, 而使用系统变量#5001~ (位置信息) 时, 成为在预读缓存内读取的坐标值。
- (2) 当在执行刀具半径补偿中插入时, 在从用户宏插入程序返回的指令中, 请务必进行顺序编号的指定 (M99P\_ ; )。如果没有顺序编号的指定, 则无法正确返回到原程序中。

13.16 刀具更换位置返回；G30.1～G30.5



功能及目的

可在参数（#8206 刀具更换）中设定刀具更换位置，通过在加工程序中执行刀具更换位置返回指令，在最佳位置进行刀具更换。  
此外，可通过指令指定进行刀具更换位置返回的轴，及开始返回的轴顺序。



指令格式

（1）刀具更换位置返回的指令格式如下。

**G30. n ;**

n = 1～5: 指令进行刀具更换位置返回的轴与返回顺序。

指令与返回顺序如下表所示。

| 指 令    | 返回顺序          |
|--------|---------------|
| G30. 1 | 仅 X 轴（→附加轴）   |
| G30. 2 | 仅 Z 轴（→附加轴）   |
| G30. 3 | X 轴→Z 轴（→附加轴） |
| G30. 4 | Z 轴→X 轴（→附加轴） |
| G30. 5 | X 轴・Z 轴（→附加轴） |

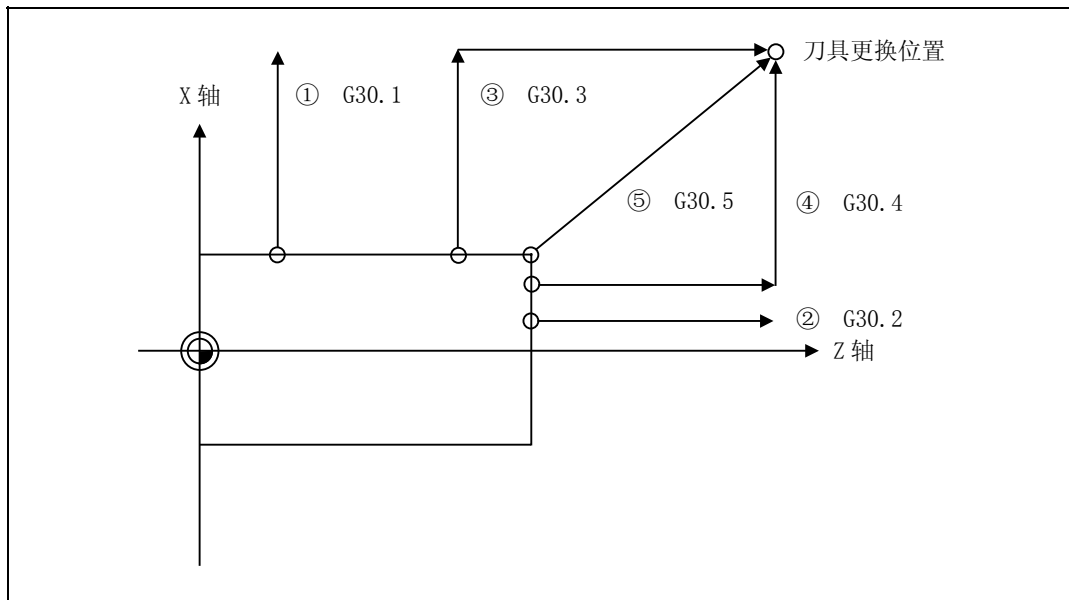
（注 1）箭头（→）表示开始返回的轴顺序，“・”表示同时开始移动。（例：Z 轴→X 轴是表示 Z 轴返回刀具更换位置后，X 轴进行刀具更换位置的返回）

- （2）关于附加轴，可通过参数“#1092 Tchg\_A”切换附加轴刀具更换位置返回的有效/无效。  
但是，刀具更换位置的返回顺序，是在标准轴完成刀具更换位置返回之后（参阅上表）。当有 2 根附加轴时，标准轴完成刀具更换位置返回之后，附加轴进行刀具更换位置返回。  
此外，无法仅对附加轴进行刀具更换位置返回。
- （3）如果与刀具更换位置返回指令在同一节中指令了轴地址，则发生程序错误（P33）。



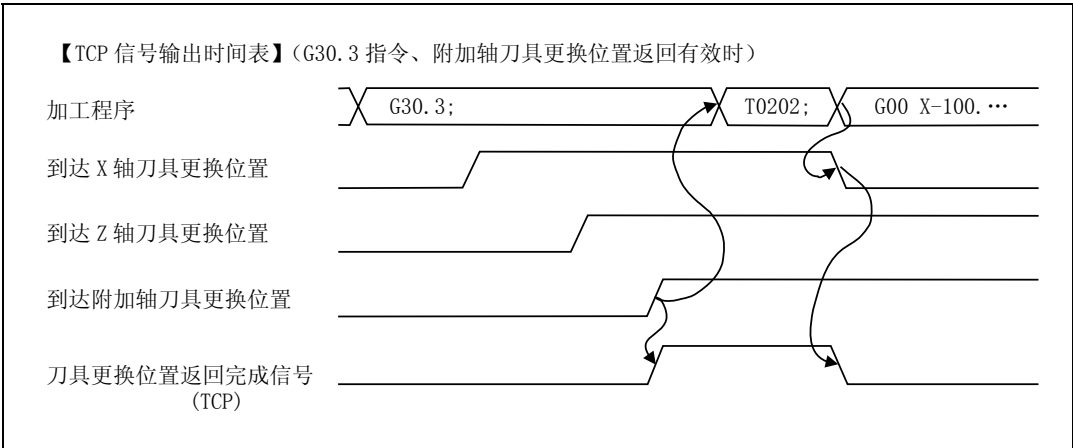
## 详细说明

(1) 下图为刀具更换位置返回指令时的动作例。



- ① G30.1指令：仅X轴返回刀具更换位置（如果附加轴也设定为刀具更换位置返回有效，则X轴到达刀具更换位置后，附加轴也进行刀具更换位置返回）
- ② G30.2指令：仅Z轴返回刀具更换位置（如果附加轴也设定为刀具更换位置返回有效，则Z轴到达刀具更换位置后，附加轴也进行刀具更换位置返回）
- ③ G30.3指令：X轴完成刀具更换位置返回后，Z轴进行刀具更换位置返回（如果附加轴也设定为刀具更换位置返回有效，则X、Z轴到达刀具更换位置后，附加轴也进行刀具更换位置返回）
- ④ G30.4指令：Z轴完成刀具更换位置返回后，X轴进行刀具更换位置返回（如果附加轴也设定为刀具更换位置返回有效，则Z轴、X轴到达刀具更换位置后，附加轴也进行刀具更换位置返回）
- ⑤ G30.5指令：X轴、Z轴同时进行刀具更换位置返回（如果附加轴也设定为刀具更换位置返回有效，则Z轴、X轴到达刀具更换位置后，附加轴也进行刀具更换位置返回）

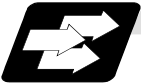
- (2) 通过G30. n指令，附加轴也完成刀具更换位置返回时，刀具更换位置返回完成信号TCP (X22B) 打开。
- 另外，根据G30. n指令向刀具更换位置移动的轴中，只要有任何1轴从刀具更换位置移出，则刀具更换位置返回完成信号立即关闭。
- 上述动作例中的G30. 3指令时，X轴到达刀具各个年换位置后，Z轴进行刀具更换位置返回动作，当Z轴到达刀具更换位置时，TCP信号打开。另外，当X轴或Z轴移动时，关闭。根据参数“#1092 Tchg\_A”，当附加轴也设定为刀具更换位置返回有效时，当附加轴也到达刀具更换位置时，TCP信号打开，X轴或Z轴又或附加轴移动时，TCP信号关闭。



- (3) 通过刀具更换位置返回指令，刀具长度补偿、刀刃磨损补偿等刀具补偿数据被暂时取消，虽然机械移动到参数中所设定的刀具更换位置，但是补偿量被记忆住，在下次移动指令中，移动到进行了刀具补偿的位置。
- (4) 本指令对每1轴按节加以分割执行。因此，在单节运转中有本指令时，由于每当1轴返回到刀具更换位置时，都会进入节停止，所以为了让下一根轴返回到刀具更换位置，需要进行循环起动。

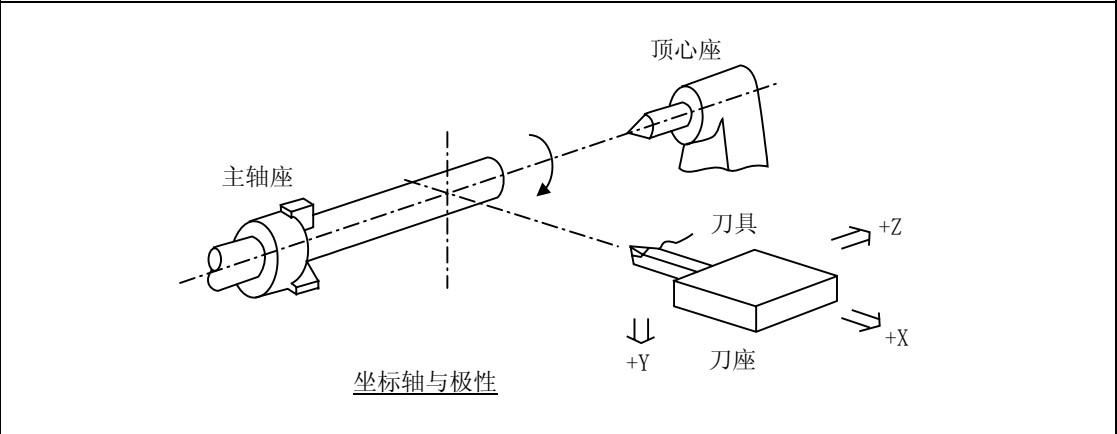
14. 坐标系设定功能

14.1 坐标与控制轴



功能及目的

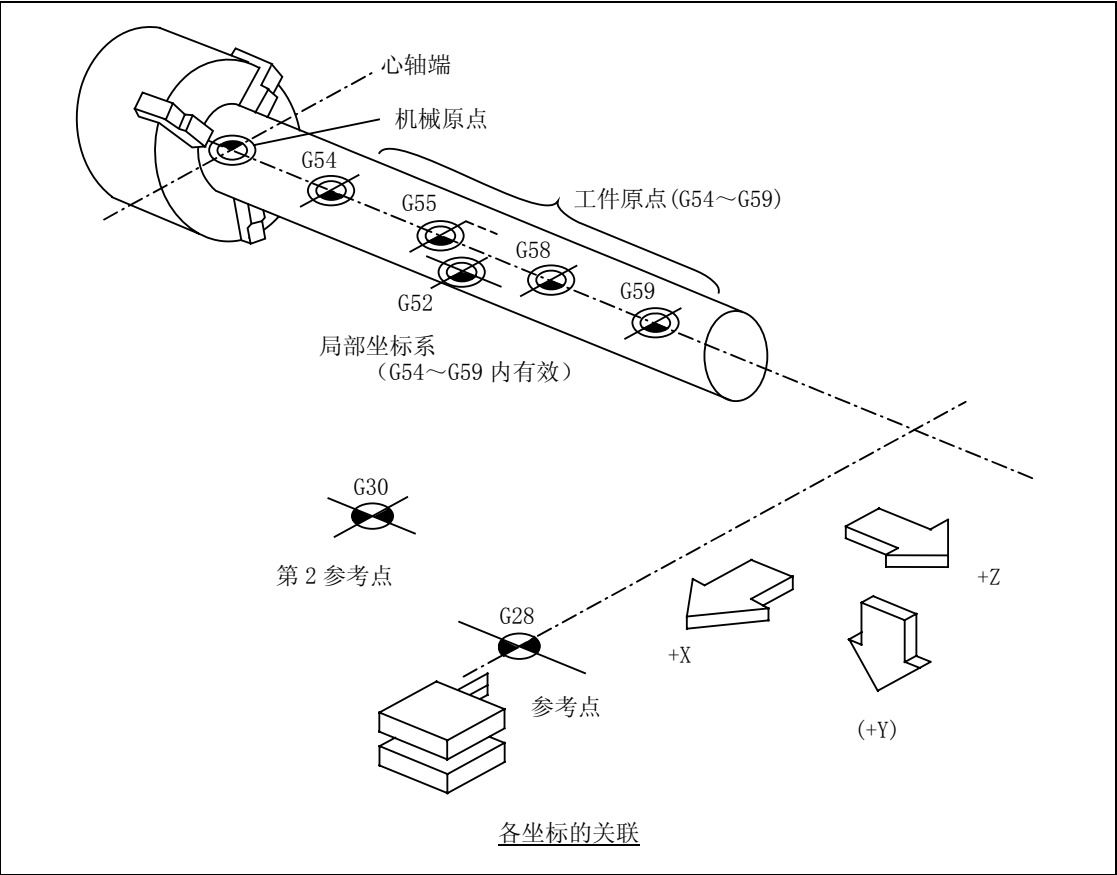
对于车床，轴的名称与方向如下图所示，与主轴平行的轴为Z轴，其正方向为刀座离开主轴座的方向，而与Z轴成直角的轴为X轴，其正方向为离开Z轴的方向。



在车床上是使用右手系统的坐标，所以在如上图所示的状态下，与X、Z轴直角相交的Y轴以下图的向下方向为正方向。

从Y轴正方向看，X、Z平面上的圆弧是以顺时针旋转、逆时针旋转的方式加以表现，所以需加以注意。

(参阅圆弧插补项)

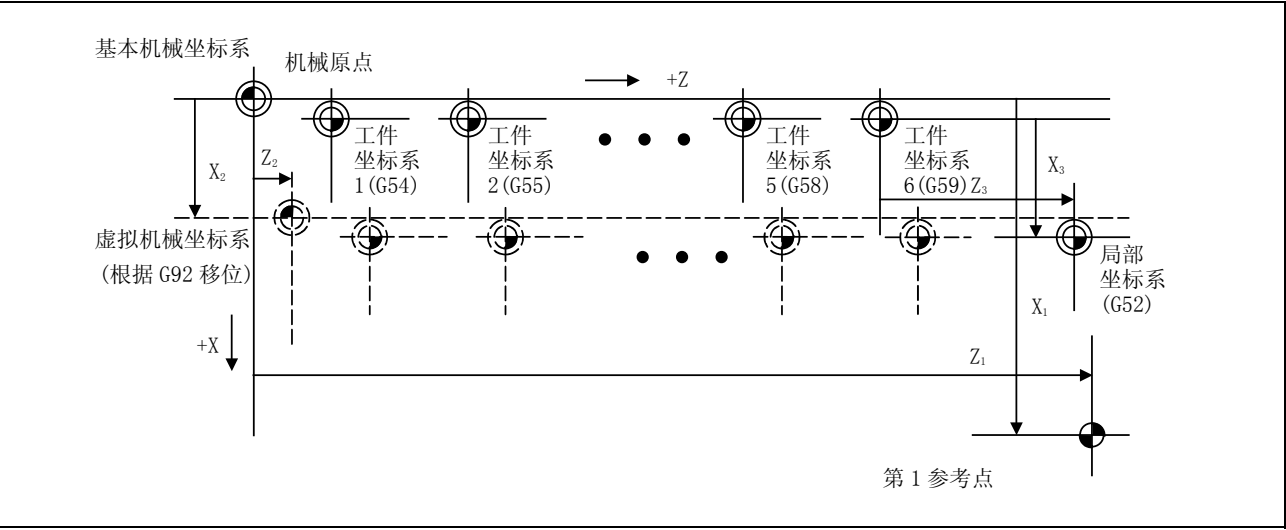


14.2 基本机械坐标系、工件坐标系与局部坐标系



功能及目的

基本机械坐标系为机械固定的坐标系，是表示机械固有位置的坐标系。  
工件坐标系是程序员编程时使用的坐标系，是以工件上的基准点为坐标原点设定的坐标系。  
局部坐标系是在工件坐标系上创建的坐标系，是为了易于创建部分加工的程序而设置的。  
在参考点返回完成时，参阅参数，自动设定基本机械坐标系及工件坐标系（G54~G59）。  
此时，设定基本机械坐标系，以使第1参考点从基本机械坐标原点（机械原点）移动到参数中所指定的位置。



局部坐标系（G52）在工件坐标系1~6所指定的坐标系中有效。  
另外，可通过G92指令，在基本机械坐标系上设定虚拟机械坐标系，此时，工件坐标系1~6也同时移位。

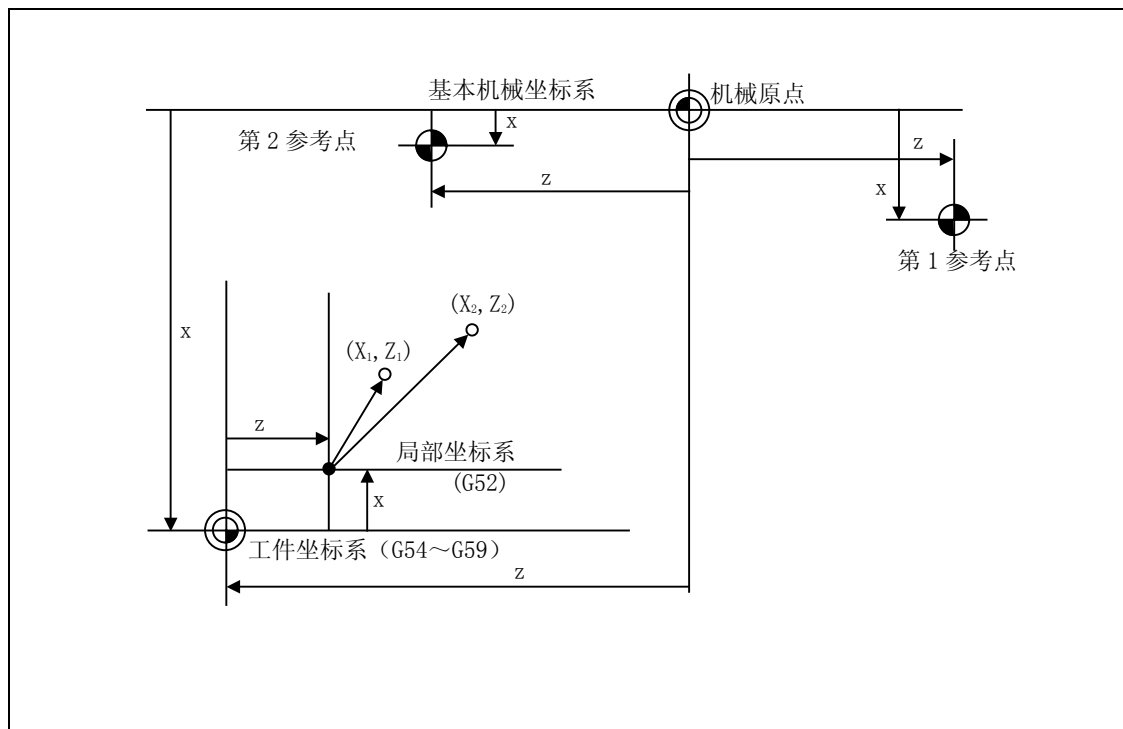
## 14.3 机械原点与第2参考点（原点）



## 功能及目的

机械原点是作为基本机械坐标系的基准的点，是参考点（原点）返回中，固定的机械固有点。

第2参考点（原点）是根据基本机械坐标系的原点，预先通过参数设定的坐标值位置的点。

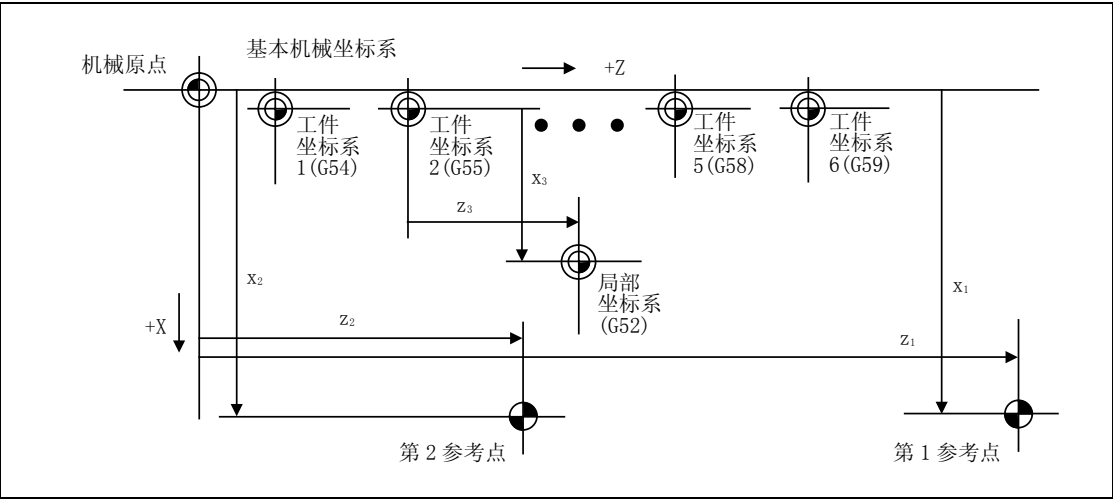


14.4 自动坐标系设定



功能及目的

本功能是接通NC电源之后，通过挡块式参考点返回到达参考点时，根据预先通过设定显示装置输入的参数值，创建各种坐标系。  
实际的加工程序，是在通过上述方式设定的坐标系上进行编程。



详细说明

- 通过本功能创建的坐标系如下。
  - 基本机械坐标系
  - 工件坐标系 (G54~G59)  
局部坐标系 (G52) 被取消。
- 与坐标系相关的参数，全部以距离基本坐标系原点的距离赋值。因此，决定将第1参考点放置在基本机械坐标系的哪个位置，之后，设定工件坐标系的原点位置。
- 执行自动坐标系设定功能，则通过G92进行的工件坐标系的移位、通过G52进行的局部坐标系设定、通过初始设置进行的工件坐标系移位、通过手动插入进行的工件坐标系移位全部取消。
- 挡块式参考点返回，是在接通电源后第1次手动参考点返回或自动参考点返回时执行，而当通过参数选择了挡块式时，是在第2次之后的手动参考点返回或自动参考点返回时执行。

 注意

 如果在自动运转中（含单节运转中）进行工件指标偏移量的变更，则从下一节或多节之后的指令开始生效。

14.5 机械坐标系选择；G53



功能及目的

通过G53指令与进给模式指令（G01或G00）及之后的坐标指令，将刀具移动到基本机械坐标系上的指令位置。



指令格式

G53 G00 Xx Zz α α ;

G53 G00 Uu Ww β β ;

α α

:

附加轴

β β

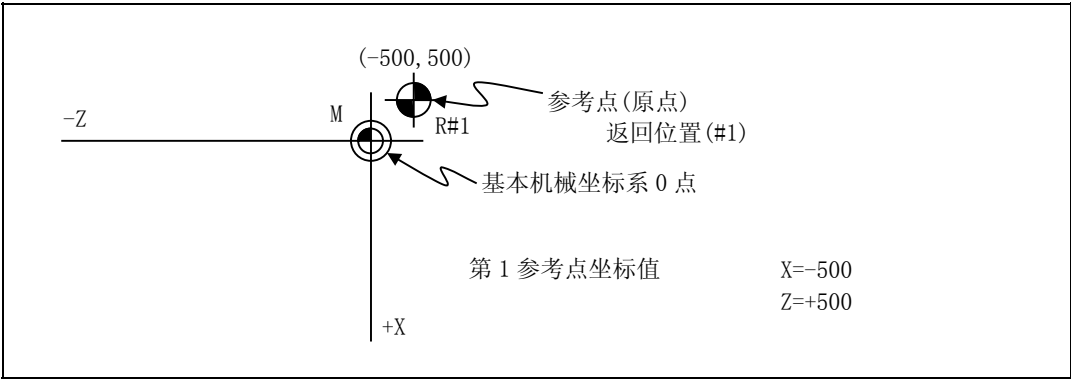
:

附加轴的增量值指令轴



详细说明

- （1）在接通电源时，通过自动或手动参考点（原点）返回，以规定的参考点（原点）返回位置为基准，自动设定基本机械坐标系。
- （2）基本机械坐标系不会因G92指令而变化。
- （3）G53指令，仅对指令的节有效。
- （4）当G53指令为增量值指令模式（U、W、β）时，以增量值在选中的坐标系中移动。
- （5）第1参考点坐标值表示参考点（原点）返回位置距基本机械坐标系0点的距离。

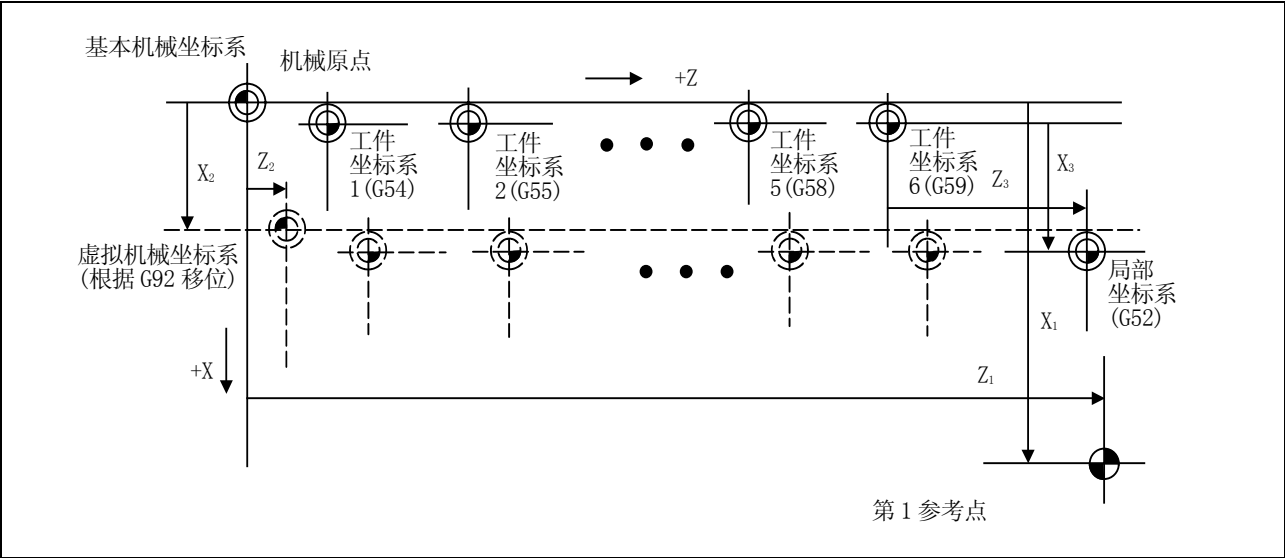


14.6 坐标系设定：G92



功能及目的

将刀具定位到任意位置，通过在该位置指令坐标系设定G92，设定坐标系。  
该坐标系可任意设定，但是通常设定为X轴、Y轴以工件中心为原点，Z轴以工件的端面为原点。



指令格式

```
G92 Xx2 Zz2 α α2;
α α : 附加轴
```



详细说明

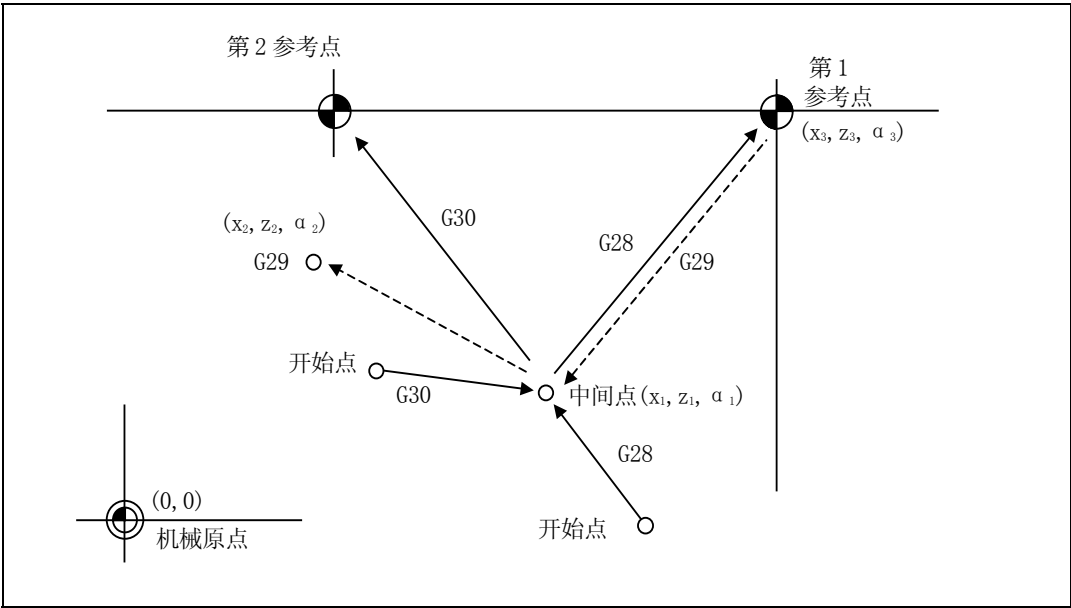
- (1) 通过G92指令，将基本机械坐标系移位，创建虚拟机械坐标系，此时，工件坐标系1~6也同时移位。
- (2) 如果与G92同时指令S或Q，则设定主轴钳位转速。（参阅主轴钳位速度设定项）

14.7 参考点（原点）返回；G28、G29



功能及目的

- (1) 通过指令G28，以G0进行指令轴的定位之后，分别以快速进给让各指令轴返回到第1参考点（原点）的功能。
- (2) 另外，通过指令G29，让各轴独立、高速定位到G28或G30的中间点之后，通过G0在指令位置进行定位。



指令格式

G28  $Xx_1$   $Zz_1$   $a_{a_1}$ ; 附加轴 自动参考点返回  
G29  $Xx_2$   $Zz_2$   $a_{a_2}$ ; 附加轴 开始位置返回  
 $a_{a_1}/a_{a_2}$  : 附加轴



## 详细说明

- (1) G28指令与下述指令等价。

G00 X<sub>x1</sub> Z<sub>z1</sub> α α<sub>1</sub> ;

G00 X<sub>x3</sub> Z<sub>z3</sub> α α<sub>3</sub> ;

在这里， $x_3, z_3, \alpha_3$ 为参考点的坐标值，通过参数“#2037 G53ofs”设定距离基本机械坐标系原点的距离。

- (2) 接通电源后，没有通过手动进行参考点（原点）返回的轴，与手动相同，进行挡块式返回。此时，将返回方向看作为指令符号方向，第2次之后，高速返回到第1次时记忆住的参考点（原点）。

- (3) 参考点（原点）返回完成，则输出到达原点输出信号，同时在设定显示装置画面的轴名称行中显示#1。

- (4) G29指令与下述指令等价。

G00 X<sub>x1</sub> Z<sub>z1</sub> α α<sub>1</sub> ;

G00 X<sub>x2</sub> Z<sub>z2</sub> α α<sub>2</sub> ;

} 为各轴独立的快速进给（非插补型）。

在这里， $x_1, z_1, \alpha$ 为G28的中间点，或是G30的中间点坐标值。

- (5) 接通电源后，不执行自动参考点（原点）返回（G28），而是指令G29，则发生程序错误（P430）。

- (6) 定位点的中间点坐标值（ $x_1, z_1, \alpha_1$ ）取决于绝对值/增量值指令。

- (7) G29对G28、G30均有效，在返回最新的中间点之后，进行指令轴的定位。

- (8) 参考点返回时，如果刀具偏移没有被取消，则在参考点返回中临时取消，中间点成为进行修正后的位置。

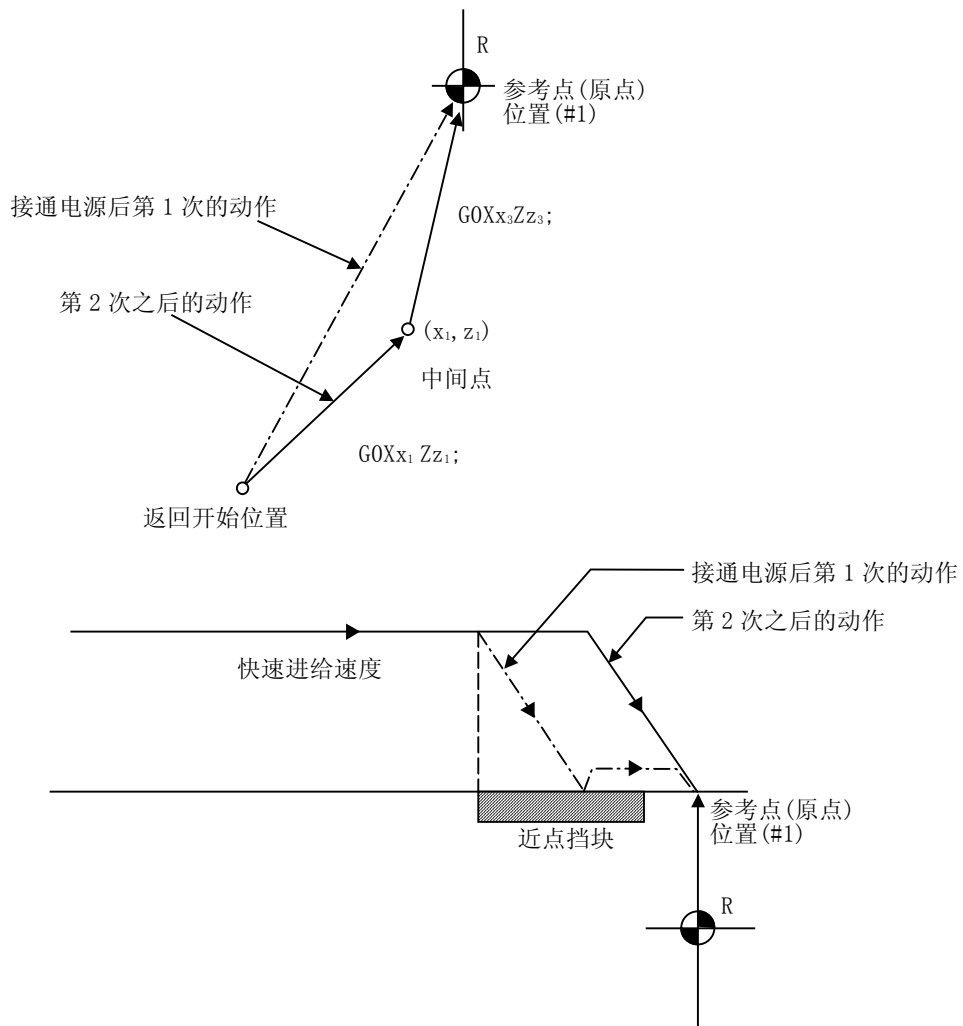
- (9) 根据参数“#1091 Mpoint”的设定，也可跳跃中间点。

- (10) 机床锁定状态下的参考点（原点）返回，跳跃从中间点到参考点（原点）之间的控制。如果指令轴到达中间点，则执行下一节。

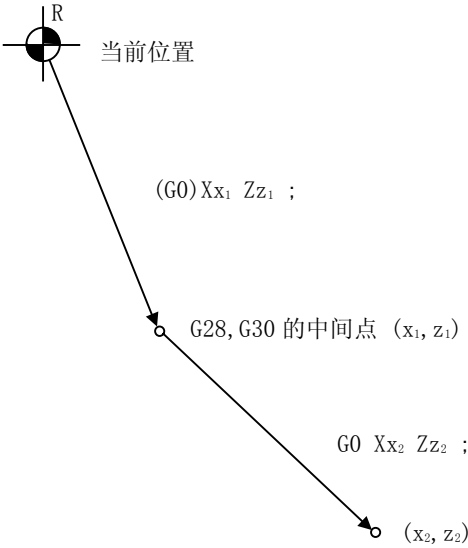
- (11) 镜像中的参考点（原点）返回，从起点到中间点，镜像有效，是向指令的反方向移动，而从中间点到参考点（原点）则是跳跃镜像，移动到参考点（原点）。



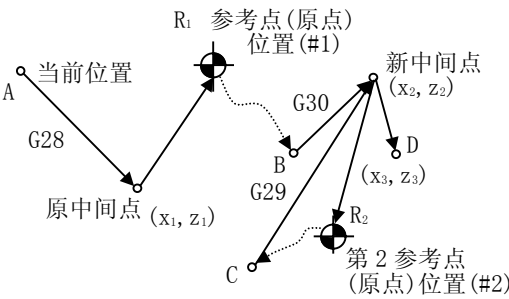
程序例

(例 1) G28 X<sub>x1</sub> Z<sub>z1</sub> ;

(例 2) G29 Xx<sub>2</sub> Zz<sub>2</sub> ;



(例 3) G28 Xx<sub>1</sub> Zz<sub>1</sub> ;  
          : (从 A 点到参考点  
          : (原点))  
G30 Xx<sub>2</sub> Zz<sub>2</sub> ;  
          : (从 B 点到第 2 参考点  
          : (原点))  
G29 Xx<sub>3</sub> Zz<sub>3</sub> ;  
          (从 C 点到 D 点)

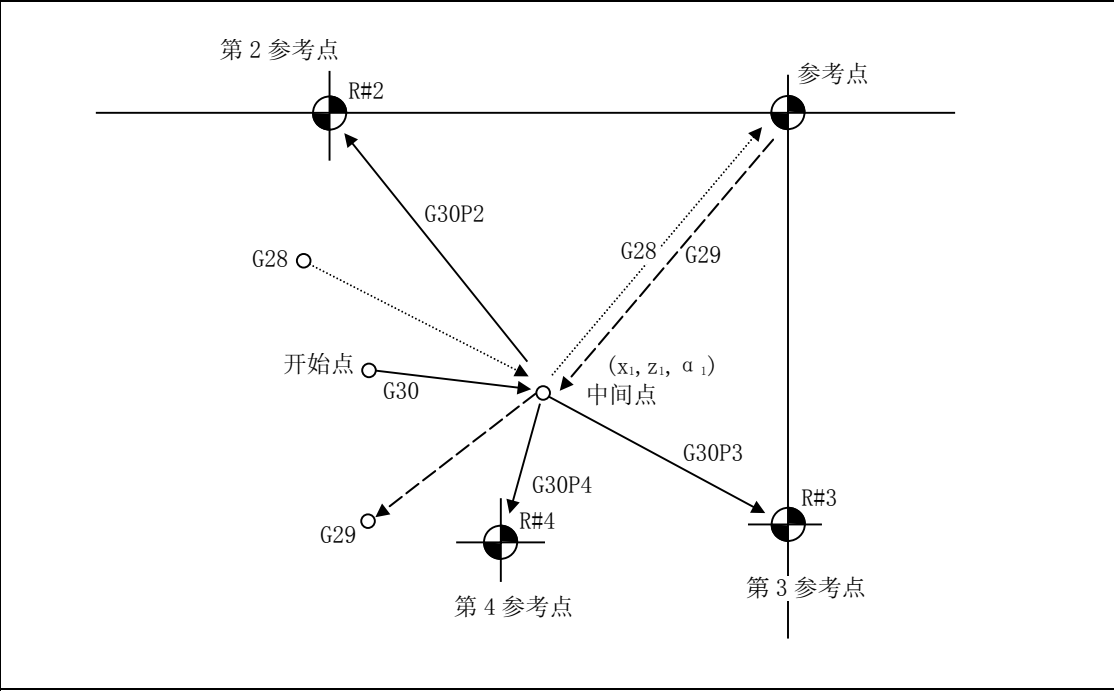


14.8 第2、第3、第4参考点（原点）返回；G30



功能及目的

通过指令G30 P2 (P3、P4) ，可返回到第2、第3或第4参考点（原点）位置。



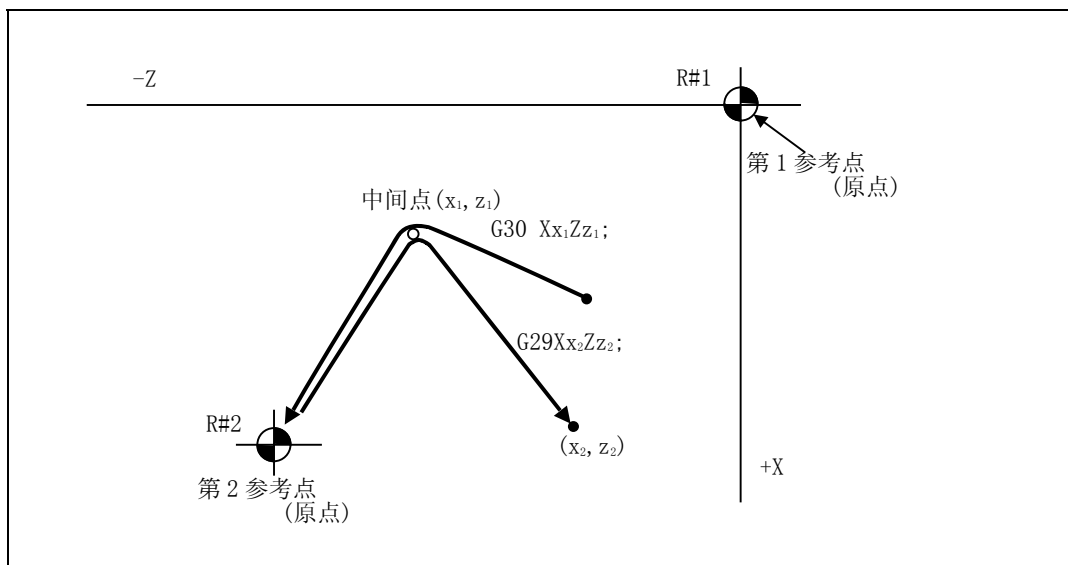
指令格式

**G30 P2 (P3, P4) Xx1 Zz1 a a1;**  
a a1 : 附加轴

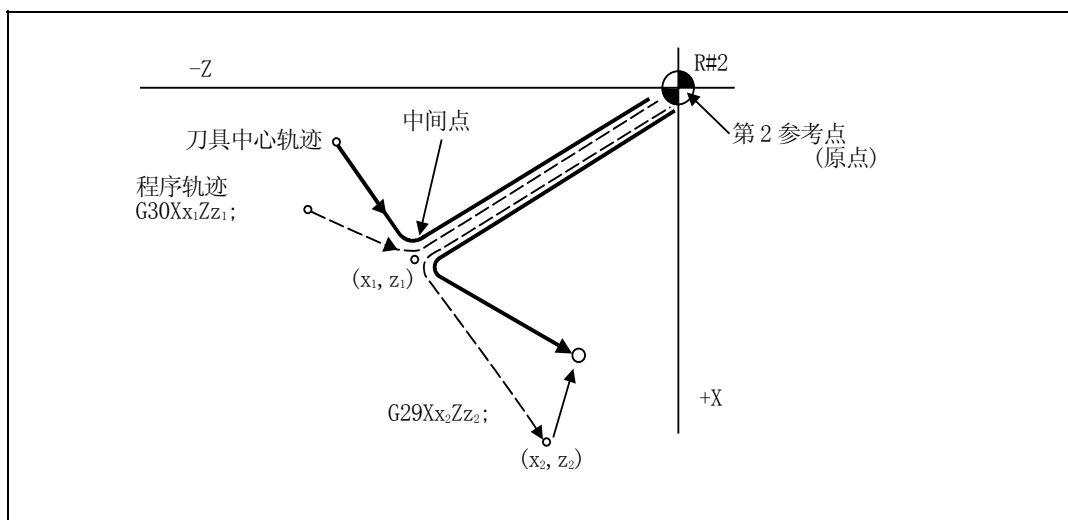


## 详细说明

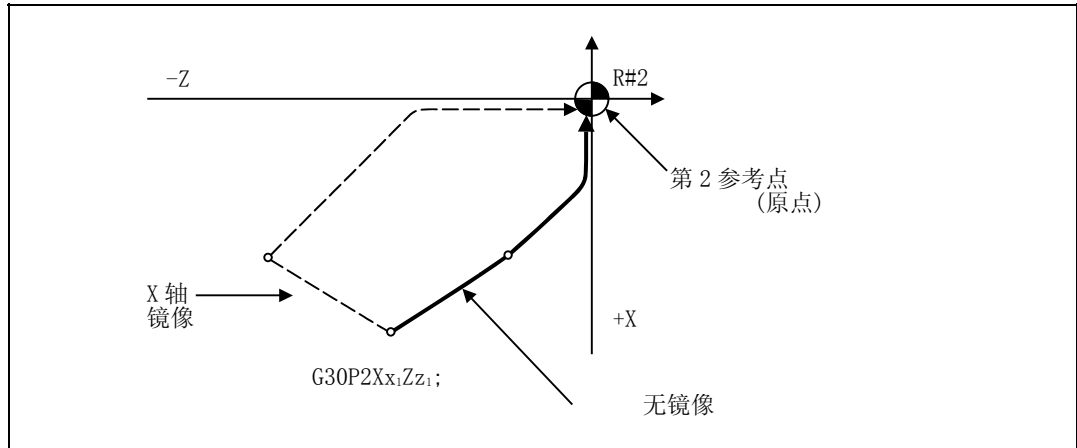
- (1) 通过P2、P3或P4进行第2、第3或第4参考点（原点）返回的指定，无P指令、P0、P1、P5以上被跳跃，成为第2参考点（原点）返回。
- (2) 第2、第3或第4参考点（原点）返回与第1参考点（原点）返回相同，是在经过由G30指定的中间点之后，返回到第2、第3或第4参考点（原点）位置。
- (3) 第2、第3、第4参考点（原点）位置坐标是机械固有的位置，可通过设定显示装置加以确认。
- (4) 在执行第2、第3、第4参考点（原点）返回之后，指令了G29指令时，G29返回时的中间点位置成为最后执行的参考点（原点）返回的中间点位置。



- (5) 补偿中的平面参考点（原点）返回，从中间点到参考点（原点）的轨迹为无刀尖R补偿（补偿为0）的轨迹。之后的G29指令中，从参考点（原点）到中间点，是以无刀尖R补偿进行移动，从中间点到G29是进行刀尖R补偿移动。



- (6) 第2、第3或第4参考点（原点）返回之后，该轴的刀具偏移量临时取消。
- (7) 机床锁定状态下的第2、第3或第4参考点（原点）返回，跳跃从中间点到参考点（原点）之间的控制。如果指令轴到达中间点，则执行下一节。
- (8) 镜像中的第2、第3或第4参考点（原点）返回，从起点到中间点，镜像有效，是向指令的反方向移动，而从中间点到参考点（原点）则是跳跃镜像，移动到参考点（原点）。



14.9 简易原点返回



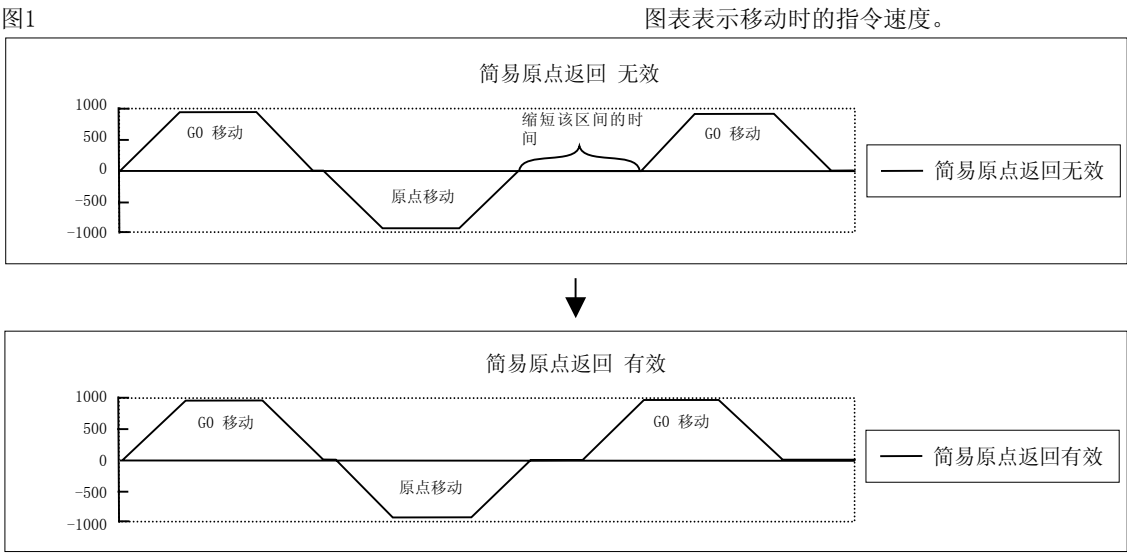
功能及目的

通过参数（#1222 auc06/bit7）设定，简化G28、G29、G30的定位动作，缩短时间。  
被缩短的时间在数十ms左右。  
当本功能有效时，定位精度可能会降低。  
当倾斜轴有效时，本功能无效。

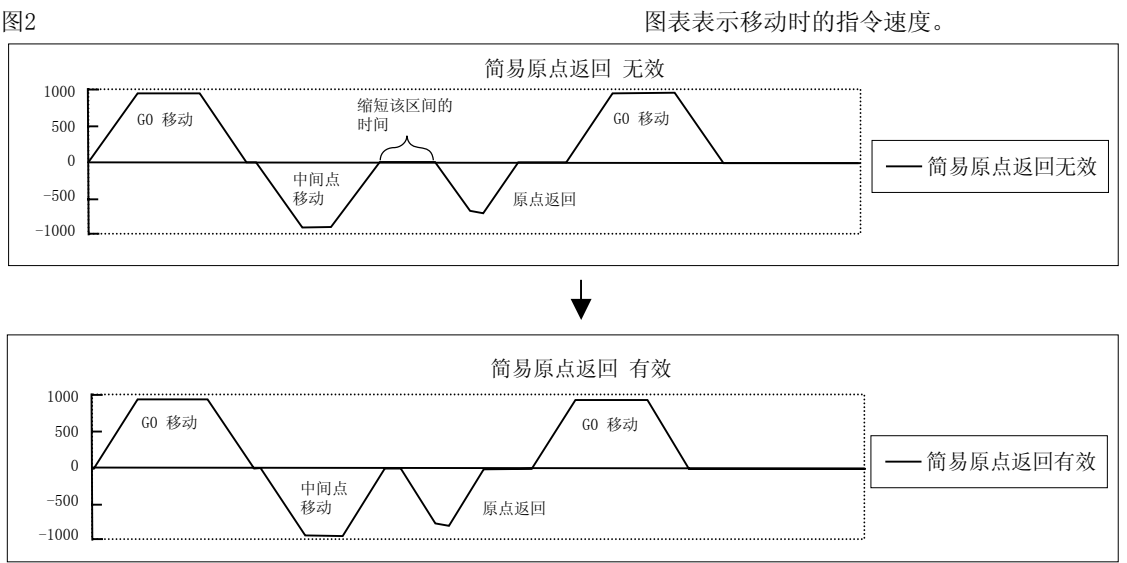


缩短时间方法

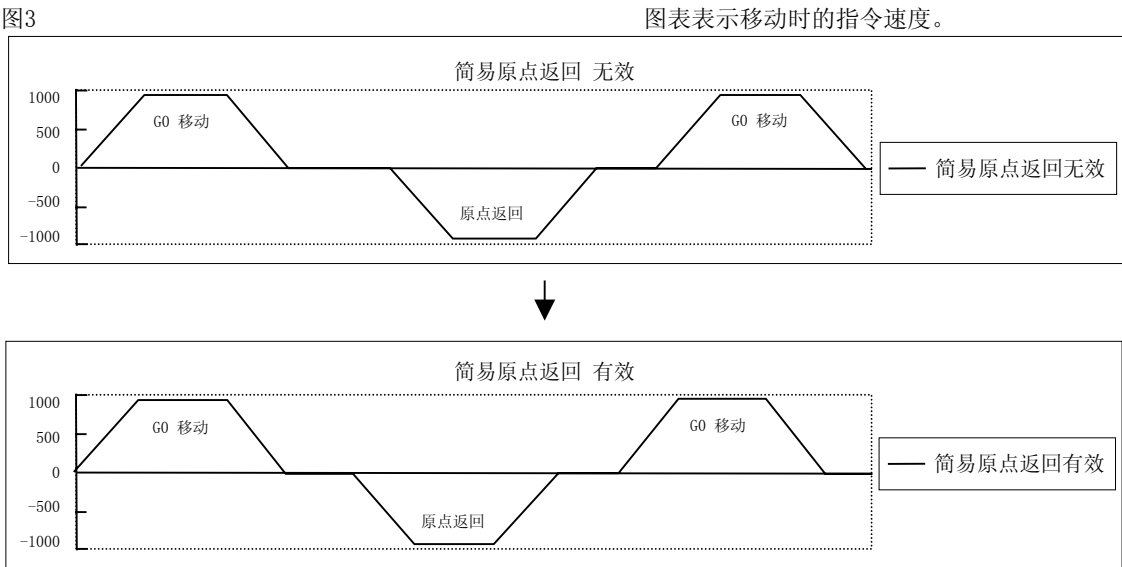
(1) 无中间点（绝对值指令）



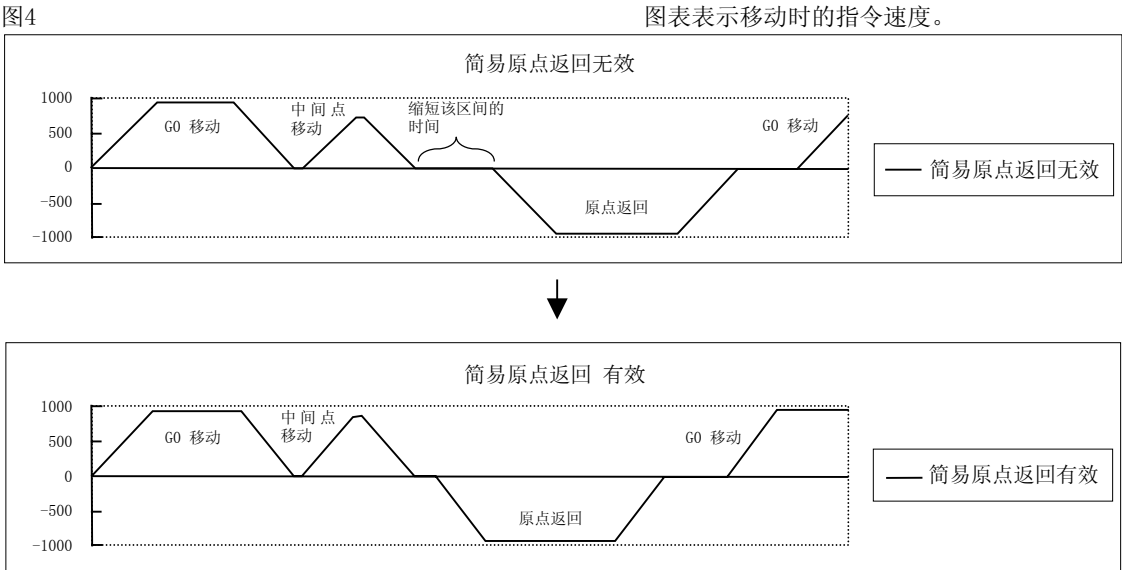
(2) 有中间点（绝对值指令）



(3) 无中间点（增量值指令）



(4) 有中间点（增量值指令）





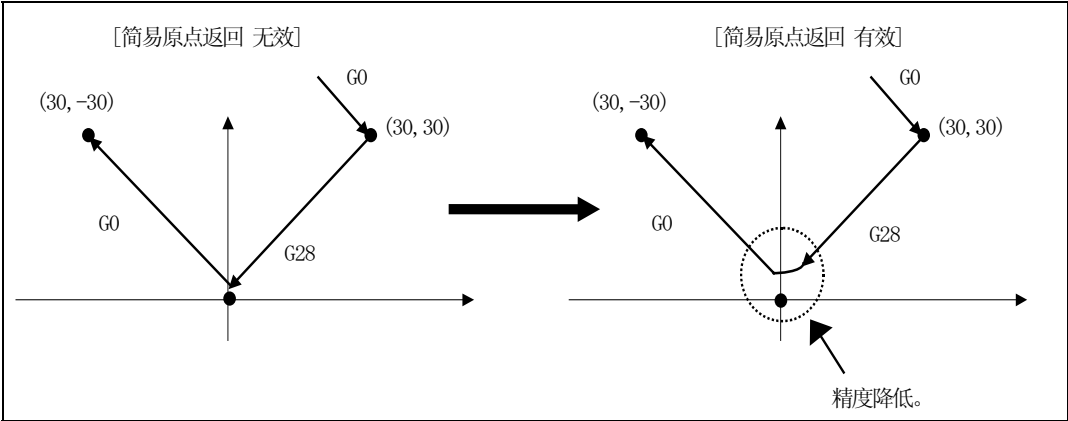
精度

(1) 无中间点（绝对值指令）

(程序例)

G00 X30.0 Z30.0;  
G28 XZ;  
G00 X30.0 Z-30.0;  
M30;

保持G28前的移动精度。  
G28的原点返回精度降低。（受定位速度・移动距离影响。）

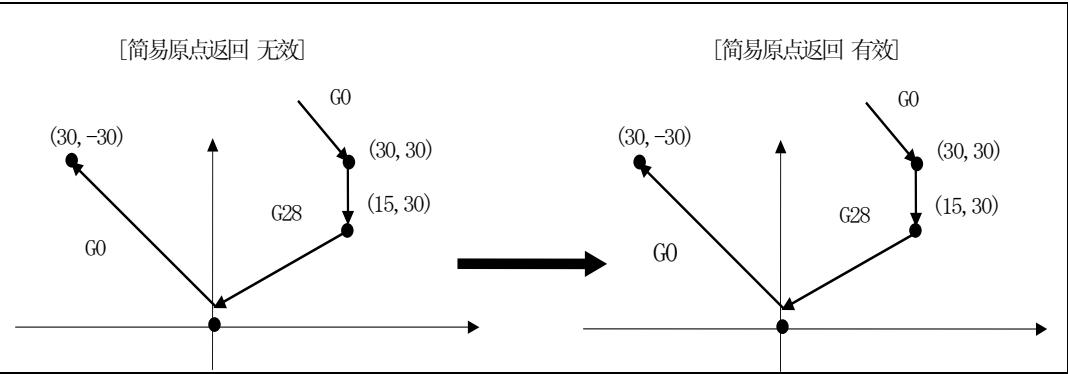


(2) 有中间点（绝对值指令）

(程序例)

G00 X30.0 Z30.0;  
G28 X15.0 Z30.0;  
G00 X30.0 Z-30.0;  
M30;

保持G28前的移动精度。  
保持移动到中间点的移动精度。

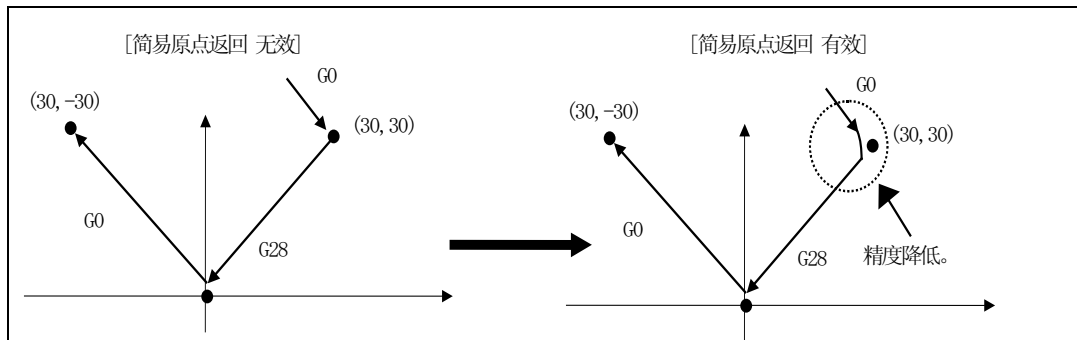


## (3) 无中间点（增量值指令）

(程序例)

```
G00 X30.0 Z30.0;
G28 UW;
G00 X30.0 Z-30.0;
M30;
```

G28前的移动精度降低。（受定位速度・移动距离影响。）



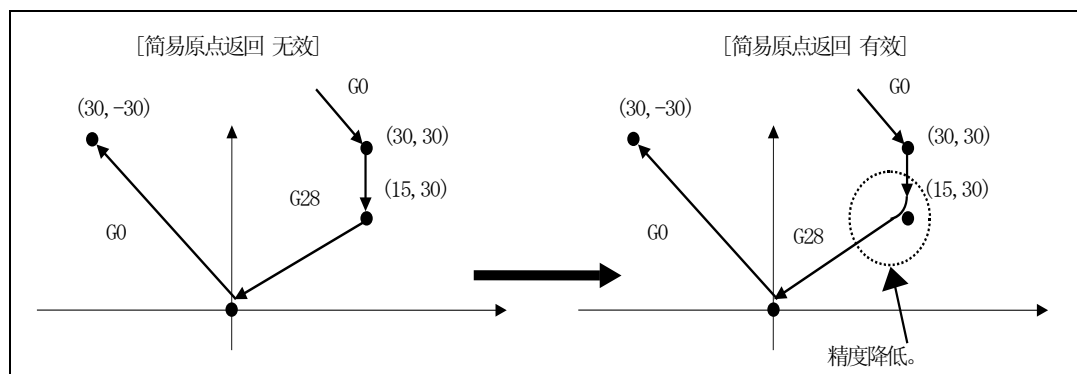
## (4) 有中间点（增量值指令）

(程序例)

```
G00 X30.0 Z30.0;
G28 U-15.0 W0.0;
G00 X30.0 Z-30.0;
M30;
```

保持G28前的移动精度。

移动到中间点的移动精度降低。（受定位速度・移动距离影响。）





#### 注意事项

- (1) 即使本功能的参数为ON，简易原点返回功能有效，当倾斜轴有效时，本功能的参数被强制关闭，本功能不生效。
- (2) 当倾斜轴有效时，如果将本功能参数打开，则导致“设定错误”，本功能不生效。
- (3) 当输入了“本功能有效”且“倾斜轴有效”的参数数据时，将本功能的参数强制性关闭，本功能不生效。
- (4) 当简易原点返回功能有效时，G28前的快速进给指令・切削进给指令及移动到中间点的移动指令精度可能会降低。  
当需要保证G28之前的快速进给指令・切削进给指令精度时，请使用以下的任何一个方法进行处理。
  - 请让本功能无效。
  - 请在G28/G29/G30前的节中，进行就位检查，确认精度。  
(G09: 精确停止检查等)
  - 请G28/G29/G30之前的节中，加入必要的延时指令。

14.10 参考点校准：G27



功能及目的

本指令是定位到通过程序给出的位置之后，如果该定位点为第1参考点，则与G28相同，向机械端输出到达参考点信号。因此，如果创建从第1参考点出发，再返回第1参考点的加工程序时，则可在执行该程序之后，检查是否返回到了参考点。



指令格式

|                              |                      |
|------------------------------|----------------------|
| <b>G27 Xx1 Zz1 α α1 Pp1;</b> |                      |
| G27                          | : 校准指令               |
| Xx1 Zz1 α α1                 | : 返回控制轴              |
| Pp1                          | : 校准编号               |
|                              | P1: 第 1 参考点校准        |
|                              | P2: 第 2            " |
|                              | P3: 第 3            " |
|                              | P4: 第 4            " |



详细说明

- (1) 当省略P指令时，变为第1参考点校准。
- (2) 能够同时进行参考点校准的轴数取决于联动轴数。
- (3) 指令完成后，如果没有到达参考点，则报警。

14.11 工件坐标系设定及工件坐标系偏移；G54～G59（G54.1）



功能及目的

- (1) 工件坐标系是以要加工的工件基准点为原点，以易于在工件上进行编程的坐标系。
- (2) 可通过本指令，移动到工件坐标系中的位置上。工件坐标系是程序员编程时使用的坐标系，除G54～G59这6组外，还有48组追加工件坐标系。
- (3) 通过本指令，在当前选中的工件坐标系中，重新设定工件坐标系，以使刀具当前位置成为指令的坐标值。（刀具的当前位置，包含刀尖R、刀具长度偏移量。）
- (4) 通过本指令，设定虚拟机械坐标系，让刀具的当前位置成为指令的坐标。  
（刀具的当前位置，包含刀尖R、刀具长度偏移量。）（G54、G92）



指令格式

- (1) 工件坐标系（G54～G59）

G54 Xx<sub>1</sub> Zz<sub>1</sub> α α<sub>1</sub>;  
α α<sub>1</sub> : 附加轴

- (2) 工件坐标系设定（G54～G59）

(G54) G92 Xx<sub>1</sub> Zz<sub>1</sub> α α<sub>1</sub>;  
α α<sub>1</sub> : 附加轴

- (3) 工件坐标系选择（P1～P48）

G54.1 Pn;

- (4) 工件坐标系设定（P1～P48）

G54.1 Pn;  
G92 Xx Zz;

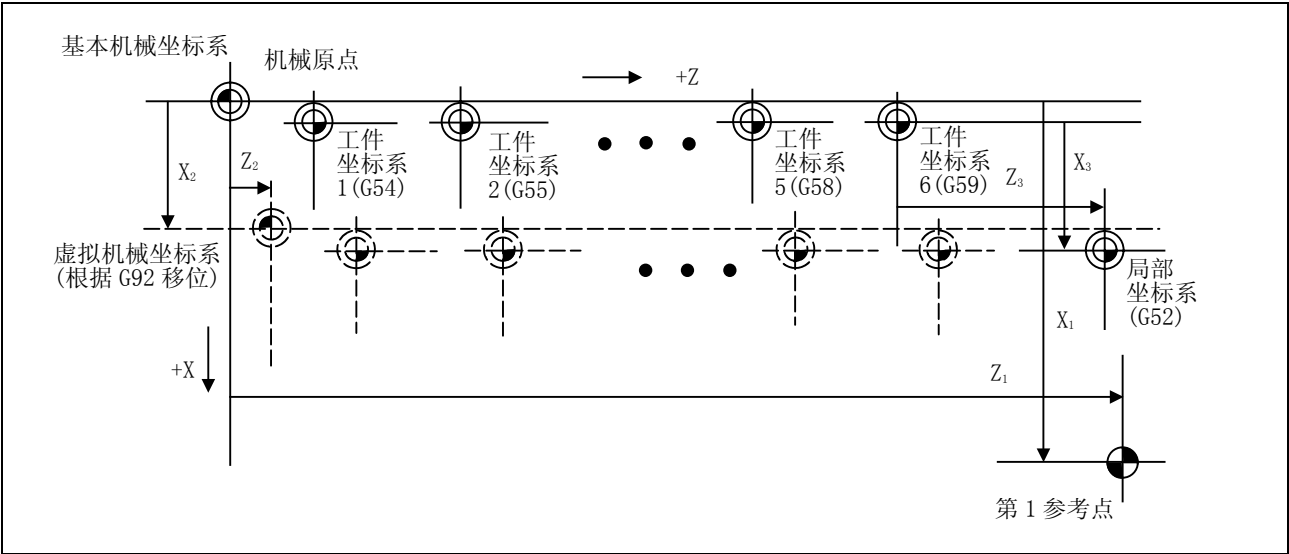
- (5) 工件坐标系偏移量的设定（P1～P48）

G10 L20 Pn Xx Zz;



详细说明

- (1) G54~G59指令中，即使指令了工件坐标系的切换，指令轴的刀尖R补偿量也会被取消。
- (2) 接通电源时，G54的坐标系被选中。
- (3) G54~G59为模态指令（12组）。
- (4) 工件坐标系中的G92，移动坐标。
- (5) 工件坐标系的偏移设定量，表示距离基本机械坐标系0点的距离。



- (6) 工件坐标系的偏移设定量，可进行多次变更。（也可通过G10 L2 Pp1 Xx1 Zz1变更。）

L 或 P 被省略时的使用

G10 L2 Pn Xx Zz ;

    ;n=0 : 在外部工件坐标系中设置偏移量。  
    n=1~6 : 在指定的工件坐标系中设置偏移量。  
    其 他 : 发生程序错误 (P35)。

G10 L2 Pn Xx Zz ;

    在当前选中的工件坐标系中设置偏移量。  
    另外，当处于 G54.1 模态中时，发生程序错误 (P33)。

G10 L20 Pn Xx Zz ;

    n=1~48: 在指定的工件坐标系中设定偏移量。  
    其 他 : 发生程序错误 (P35)。

G10 L20 Xx Zz ;

    在当前选中的工件坐标系中设置偏移量。  
    另外，当处于 G54~G59 模态中时，发生程序错误 (P33)。

G10 Pn Xx Zz ;  
G10 Xx Zz ;

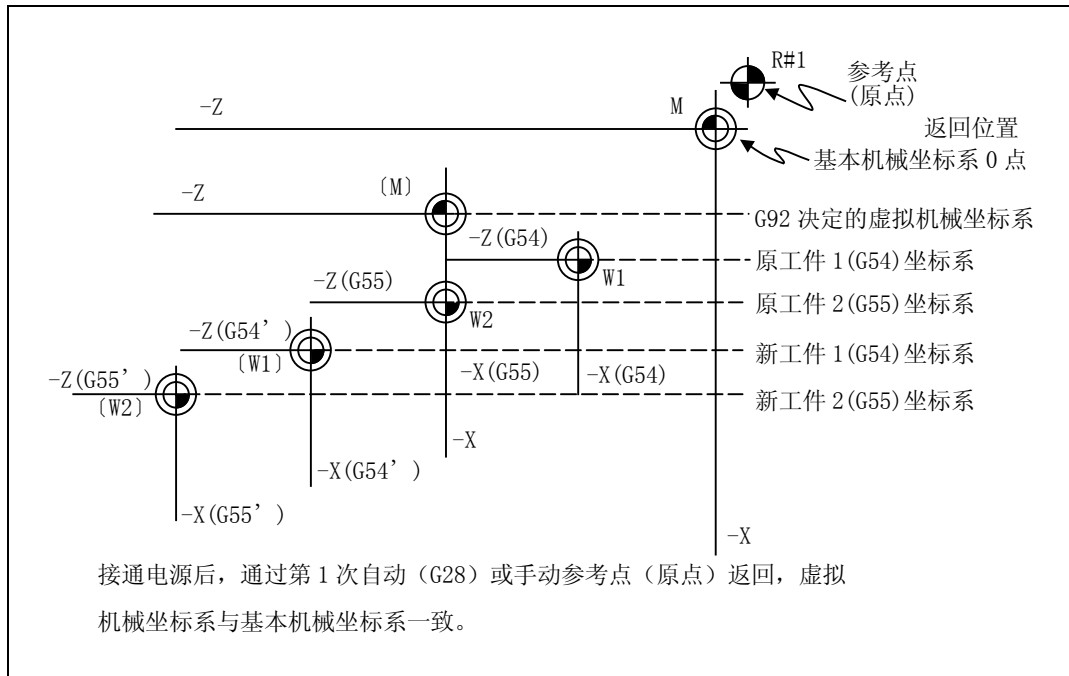
    } 没有 L 时，看作为 L10 (刀具偏移)。

## 14. 坐标系设定功能

### 14.11 工件坐标系设定及工件坐标系偏移

(7) 通过在G54（工件坐标系1）模式下指令G92，设定新的工件坐标系，同时，其他工件坐标系2~6（G55~G59）也平行移动工件坐标系，设定为新工件坐标系2~6。

(8) 根据新参考点（原点），在工件坐标系偏移后的位置上，构成虚拟机械坐标系。



(9) 通过设定虚拟机械坐标系，距虚拟机械坐标系原点移位了工件坐标系偏移量的位置，被设定为新的工件坐标系。

(10) 接通电源后的首次（G28）或手动参考点（原点）返回完成后，取决于参数设定的基本机械坐标系、工件坐标系被自动设定。

(11) 在接通电源后的参考点返回（包括自动、手动）之后，如果指令G54X-；，则发生程序错误（P62）。（由于是以G01速度进行控制，所以必须有速度指令。）

(12) 请不要与G54.1在同一节内指令使用P代码的G代码。如果进行了指令，则P代码被优先的G指令所使用。

(13) 当未附加工件偏移组数追加规格时，如果执行G54.1指令，则发生程序错误（P39）。

(14) 当未附加工件偏移组数追加规格时，如果执行G10 L20指令，则发生程序错误（P172）。

(15) G54.1模态中，无法使用局部坐标系。在G54.1模态中执行了G52指令，则发生程序错误（P438）。

(16) 通过在G54.1 P1模式下指令G92，设定新的工件坐标系P1，同时，其他工件坐标系G54~G59、G54.1、P2~P48也平行移动工件坐标系，设定为新工件坐标系2~6。

(17) 扩展工件坐标系偏移量，被分配为表1所示的变量编号。

表1 扩展工件坐标偏移系统变量编号表

|     | 1 轴～n 轴     |     | 1 轴～n 轴     |
|-----|-------------|-----|-------------|
| P1  | #7001～#700n | P25 | #7481～#748n |
| P2  | #7021～#702n | P26 | #7501～#750n |
| P3  | #7041～#704n | P27 | #7521～#752n |
| P4  | #7061～#706n | P28 | #7541～#754n |
| P5  | #7081～#708n | P29 | #7561～#756n |
| P6  | #7101～#710n | P30 | #7581～#758n |
| P7  | #7121～#712n | P31 | #7601～#760n |
| P8  | #7141～#714n | P32 | #7621～#762n |
| P9  | #7161～#716n | P33 | #7641～#764n |
| P10 | #7181～#718n | P34 | #7661～#766n |
| P11 | #7201～#720n | P35 | #7681～#768n |
| P12 | #7221～#722n | P36 | #7701～#770n |
| P13 | #7241～#724n | P37 | #7721～#772n |
| P14 | #7261～#726n | P38 | #7741～#774n |
| P15 | #7281～#728n | P39 | #7761～#776n |
| P16 | #7301～#730n | P40 | #7781～#778n |
| P17 | #7321～#732n | P41 | #7801～#780n |
| P18 | #7341～#734n | P42 | #7821～#782n |
| P19 | #7361～#736n | P43 | #7841～#784n |
| P20 | #7381～#738n | P44 | #7861～#786n |
| P21 | #7401～#740n | P45 | #7881～#788n |
| P22 | #7421～#742n | P46 | #7901～#790n |
| P23 | #7441～#744n | P47 | #7921～#792n |
| P24 | #7461～#746n | P48 | #7941～#794n |

△注 意

△ 如果在单节停止时变更工件坐标系偏移量，则从下一节起开始生效。

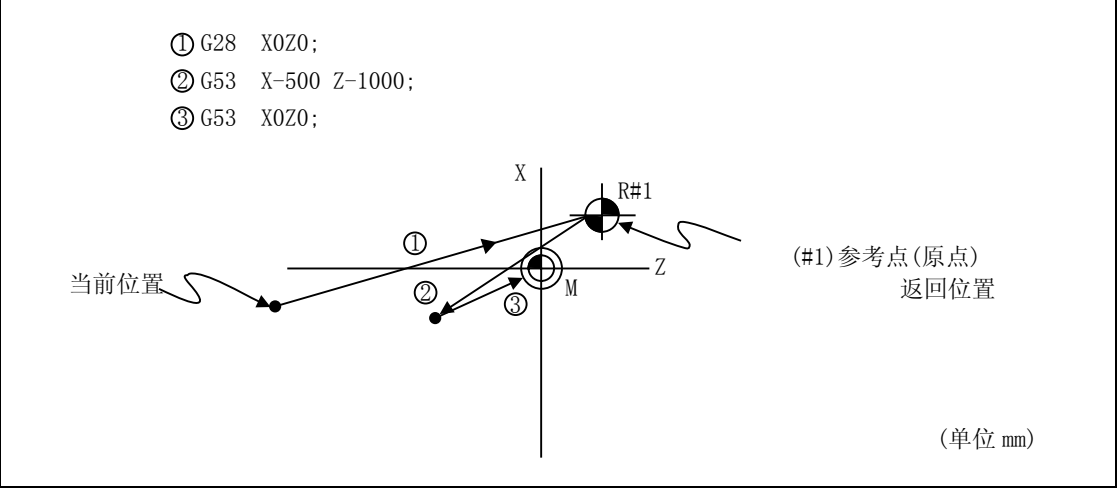
# 14. 坐标系设定功能

## 14.11 工件坐标系设定及工件坐标系偏移



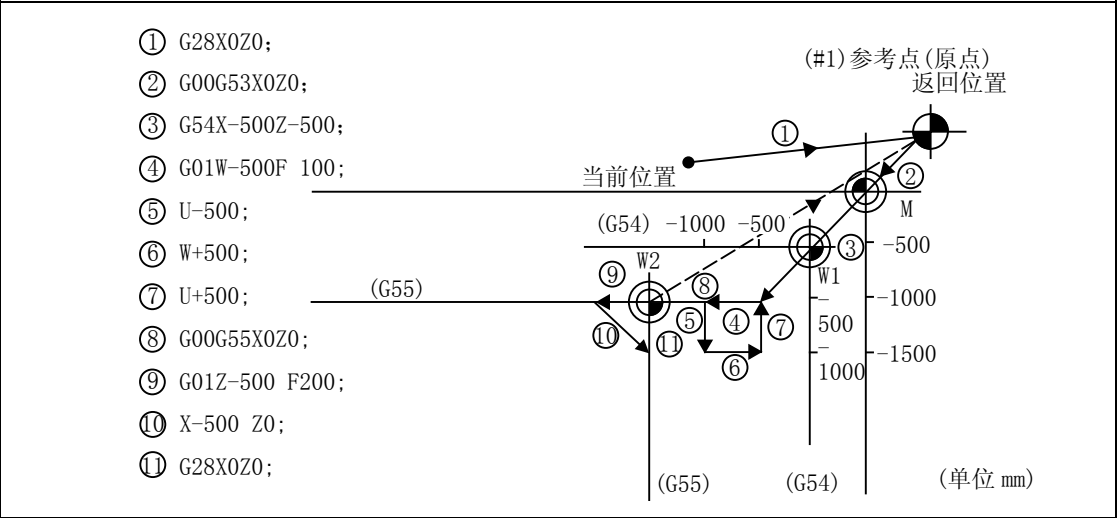
程序例

(例 1)



当第1参考点坐标值为0（零）时，基本机械坐标系0点与参考点（原点）返回位置（#1）一致。

(例 2)

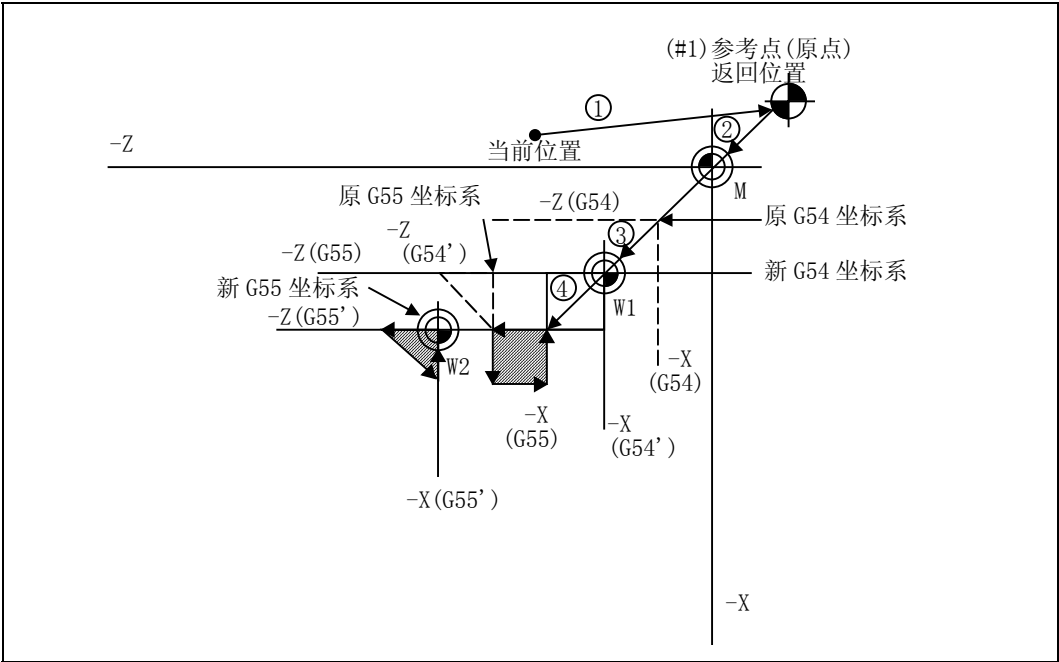


14. 坐标系设定功能

14.11 工件坐标系设定及工件坐标系偏移

(例 3) 例 2 中，工件坐标系 G54 偏离了 (-500、-500)。(将例 2 的③~⑩注册到子程序 1111 中。)

|                   |                    |
|-------------------|--------------------|
| ①G28 X0 Z0        |                    |
| ②G00 G53 X0 Z0;   | (当没有基本坐标系的偏移时，不需要) |
| ③G54 X-500 Z-500; | 工件坐标系偏移量           |
| ④G92 X0 Z0;       | 新工件坐标系正设定          |
| ⑤M98 P1111;       |                    |



(注) 如果反复使用③~⑤，则工件坐标系发生偏移，所以在程序结束时，请指令参考点返回 (G28)。

 注意

如果在单节停止时变更工件坐标系偏移量，则从下一节起开始生效。

14.12 局部坐标系设定；G52



功能及目的

可通过G52指令，在G54～G59的各工件坐标系上独立设定局部坐标系，以确保指令的位置为程序原点。  
也可使用G52指令代替G92指令，变更加工程序原点与加工工件原点之间的偏移。



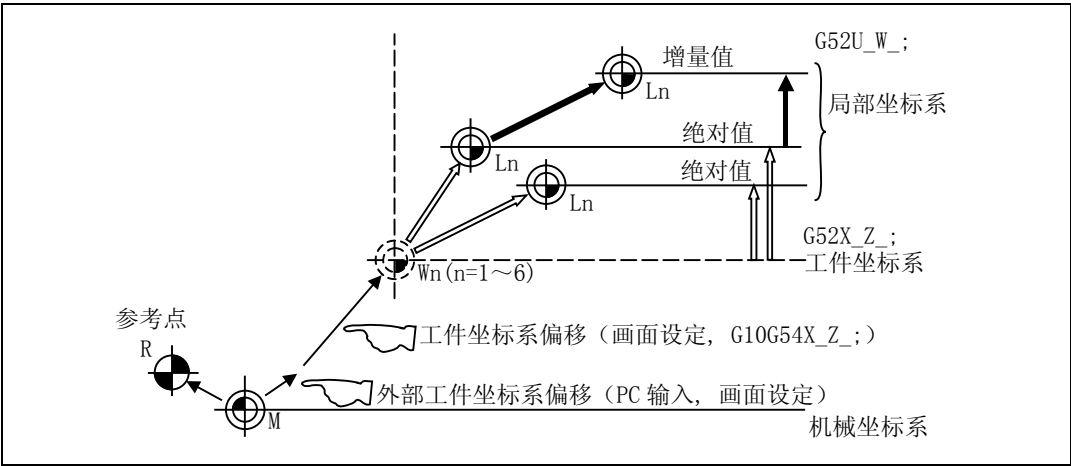
指令格式

```
G54 (G54～G59) G52 Xx1 Zz1;
```



详细说明

- (1) 在进行新的G52指令之前，G52指令一直有效，不进行移动。G52指令不变更工件坐标系（G54～G59）的原点位置，便于使用另一个坐标系。
- (2) 在接通电源后的参考点（原点）返回及挡块式手动参考点（原点）返回中，局部坐标系偏移被清除。
- (3) 通过（G54～G59）G52 X0 Z0 ；进行局部坐标系的取消。
- (4) 绝对值模式中的坐标指令，变为移动到局部坐标系中的位置。



（注）如果反复执行程序，则工件坐标系发生偏移，所以在程序结束时，请指令参考点返回动作。

## 14. 坐标系设定功能

### 14.13 工件坐标系预设

#### 14.13 工件坐标系预设: G92.1

[E68]



## 功能及目的

- (1) 是通过程序指令 (G92.1)，将通过手动运转或程序指令进行移位后的工件坐标系，预设为从机械原点偏移了工件坐标偏移量之后的工件坐标系的功能。
- (2) 当进行了如下的操作或程序指令时，设定的工件坐标系是由机械坐标系移动而成。
- 手动绝对关闭状态下手动介入时
  - 机床锁定状态下执行了移动指令时
  - 通过手轮切入而移动时
  - 通过镜像进行运转时
  - 通过G52设定局部坐标系
  - 通过G92移动工件坐标系

本功能与进行手动参考点返回时相同，将移位后过的工件坐标系预设为从机械原点偏移工件坐标偏移量之后的工件坐标系。此外，可通过参数选择是否也对相对坐标进行预设。



## 指令格式

---

**G92.1**      X0 Y0 Z0 α 0;

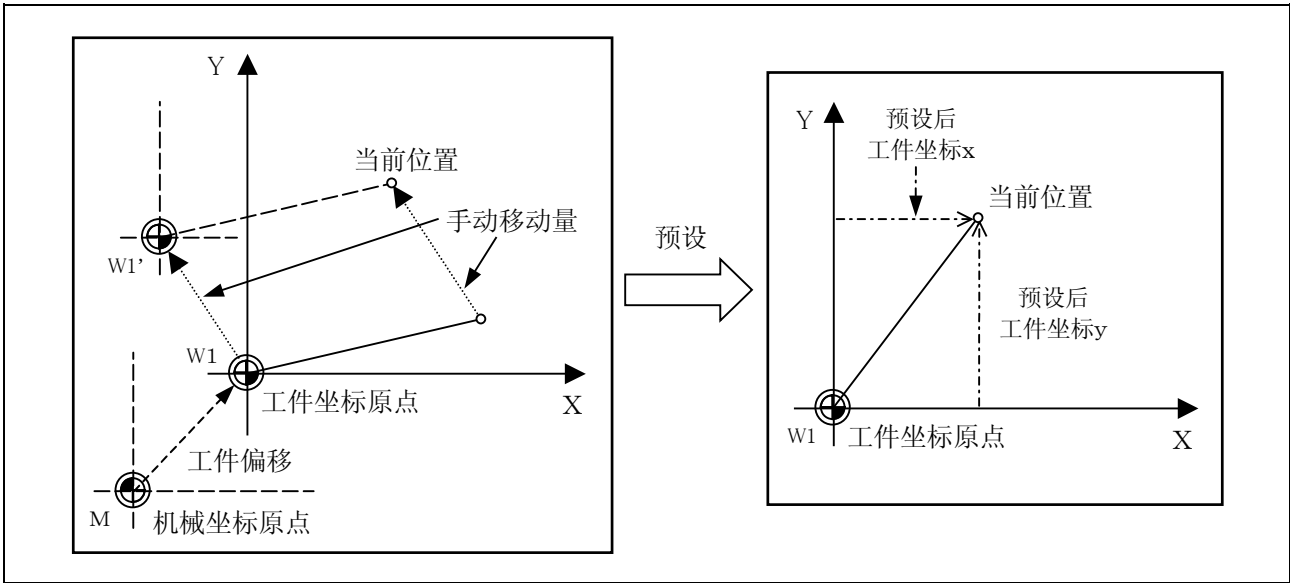
**(G50. 3)**

 $\alpha 0$  : 附加轴



详细说明

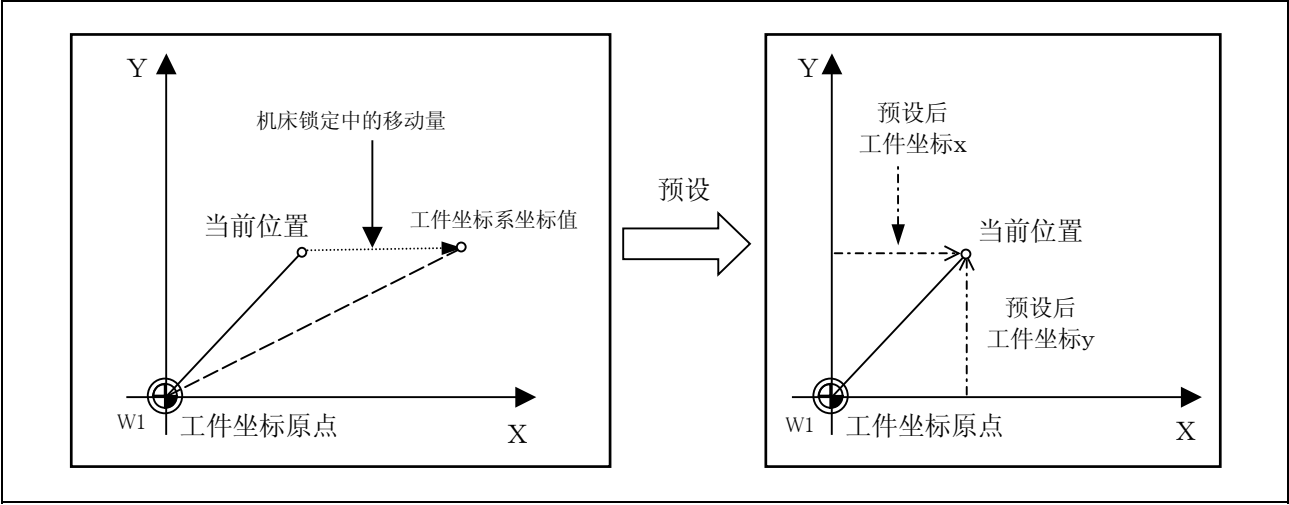
- (1) 指定希望进行预设的轴地址。未指定的轴不会被预设。
- (2) 在指令值中指定了0以外的值时，发生程序错误（P35）。
- (3) 能够指令的G代码系列如下。
  - G代码系列2、3、6、7
- (4) 根据G代码系列，G代码变为“G50.3”。
- (5) 手动绝对关闭状态下，通过手动运转及手轮插入移动时



如果在手动绝对关闭状态下，通过手动运转及手轮插入进行移动，则工件坐标系按照手动移动量进行移位。

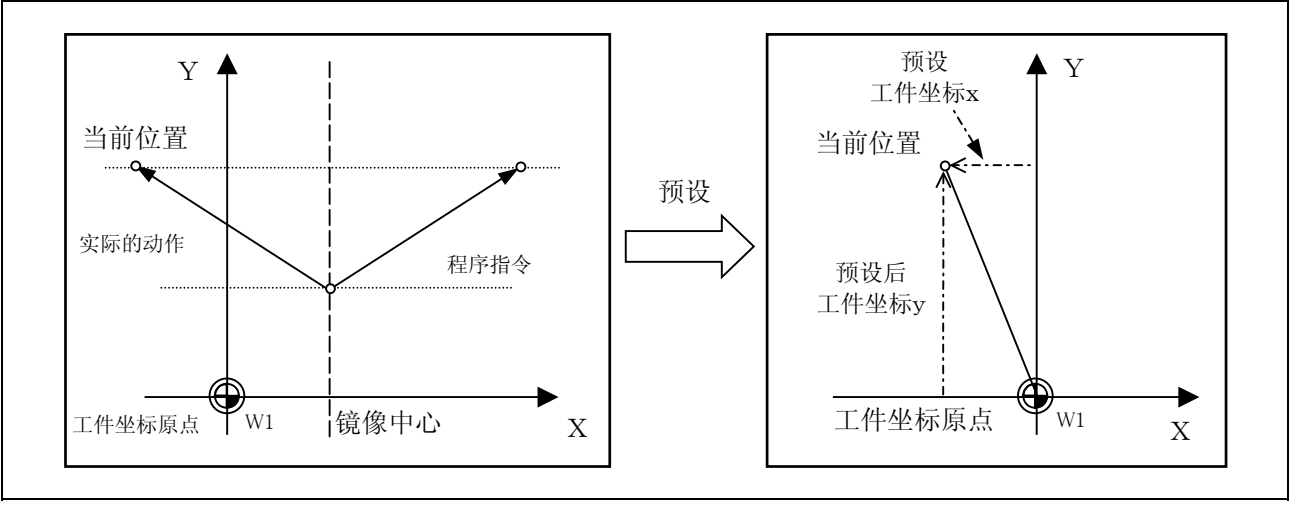
本功能是将移位后的工件坐标原点W1'返回原来的工件坐标原点W1，以从W1到当前位置的距离，作为工件坐标系的当前位置。

(6) 机床锁定状态下执行了移动指令时



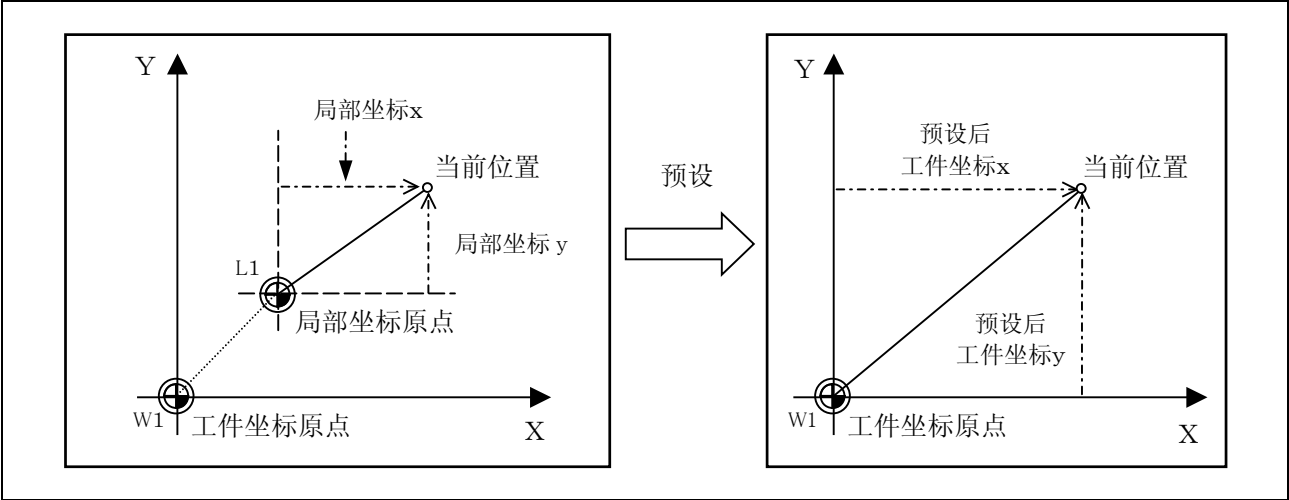
如果在机床锁定中进行移动指令，则不移动当前位置，而是只移动工件坐标。  
本功能让移动后的工件坐标返回原来的当前位置，以从W1到当前位置的距离作为工件坐标系的当前位置。

(7) 通过镜像进行运转时



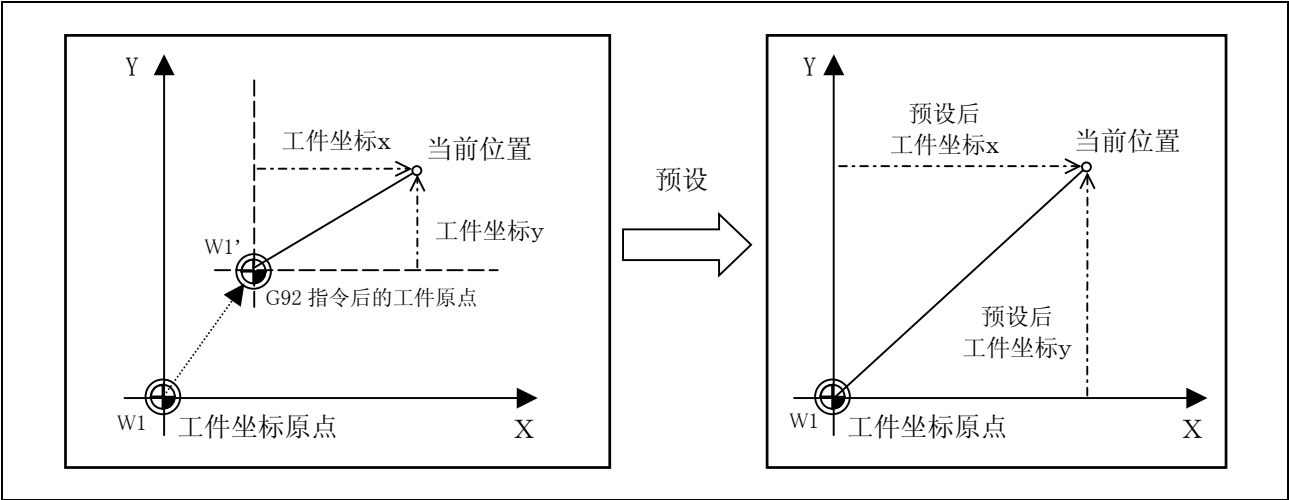
如果通过镜像进行运转，则只有NC的内部坐标变为程序指令作保，其他坐标变为当前位置坐标。  
本功能是将NC的内部坐标也作为当前位置坐标。

(8) 通过G52设定局部坐标系



通过G52指令设定局部坐标系，则在局部坐标系中进行程序指令等。  
在本功能中，所设定的进步坐标系被取消，程序指令等变为以W1为原点的工件坐标系。被取消的局部坐标系仅为选中的工件坐标系。

(9) 通过G92移动工件坐标系

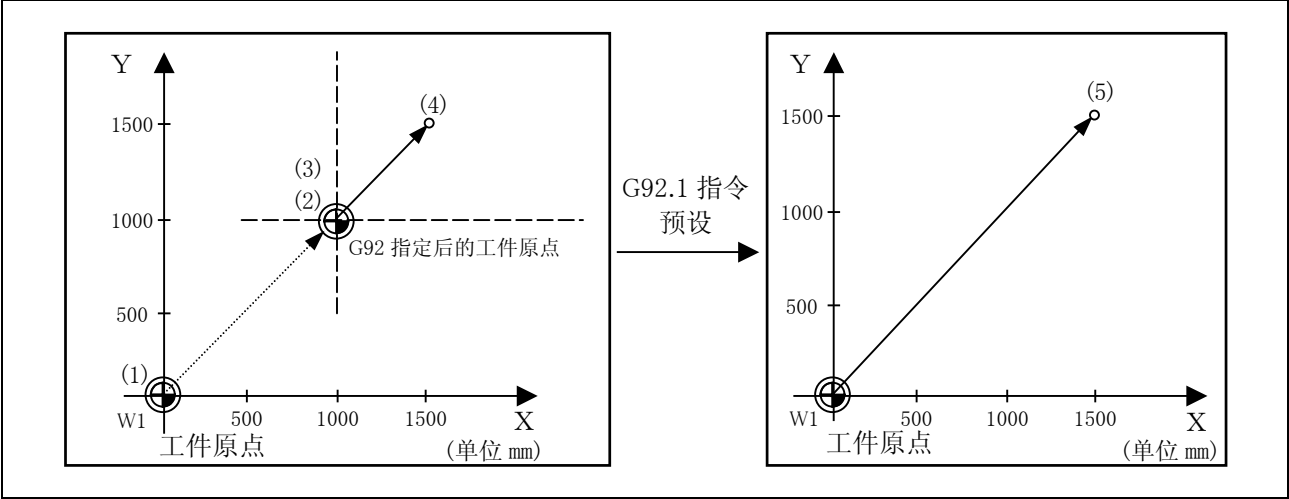


通过G92指令移动工件坐标系，W1'与当前位置的距离变为工件坐标系的当前位置。  
本功能是将移位后的工件坐标原点返回W1，以从W1到当前位置的距离，作为工件坐标系的当前位置。  
对所有的工件坐标系有效。



程序例

通过G92.1，对通过G92移位的工件坐标系进行预设。



(例)

|                   |           |
|-------------------|-----------|
| G28 X0 Y0;        | • • • (1) |
| G00 G90 X1. Y1. ; | • • • (2) |
| G92 X0 Y0;        | • • • (3) |
| G00 X500 Y500;    | • • • (4) |
| G92.1 X0 Y0;      | • • (5)   |
|                   | •         |



注意事项

- (1) 进行本功能时，请取消刀具半径补偿、刀具长度补偿、刀具位置偏移。在不取消的情况下执行本功能，则由于工件坐标是从机械值减去工件坐标偏移量之后的值，所以进入补偿矢量暂时取消的状态。
- (2) 在程序重新启动时，本功能无法执行。
- (3) 在换算、坐标旋转、程序镜像各模式中，请不要指令本功能。  
当进行指令时，发生程序错误 (P34)。

## 14.14 旋转轴用坐标系



## 功能及目的

通过参数指定为旋转轴的轴，以旋转轴的坐标系进行控制。

旋转轴的种类中，包括旋转型（附近有效/无效）与直线型。

显示范围，对于旋转型为 $0 \sim 359.999^\circ$ ，直线型为 $0 \sim \pm 99999.999^\circ$ 。

各坐标系的范围，旋转型为 $0 \sim \pm 359.999^\circ$ ，直线型为 $0 \sim \pm 99999.999^\circ$ 。

与英制・公制的指定无关，旋转轴是以度（°）为单位进行指定。

旋转轴的种类为轴通用，通过参数“#1089 Cut\_RT 旋转轴接近”与“#1090 Lin\_RT 直线型旋转轴”进行设定。

|                 | 旋转轴                                    |                                      |                                          | 直线轴 |
|-----------------|----------------------------------------|--------------------------------------|------------------------------------------|-----|
|                 | 旋转型旋转轴                                 |                                      | 直线型旋转轴                                   |     |
|                 | 附近无效                                   | 附近有效                                 |                                          |     |
| 「#1089」の<br>设定值 | 0                                      | 1                                    | -                                        | -   |
| 「#1090」の<br>设定值 | 0                                      |                                      | 1                                        | -   |
| 工件坐标位置          | 以 0～359.999° 的范围加以显示。                  |                                      | 以 0～±99999.999° 的范围加以显示。                 |     |
| 机械坐标位置<br>/相对位置 | 以 0～359.999° 的范围加以显示。                  |                                      | 以 0～±99999.999° 的范围加以显示。                 |     |
| ABS 指令          | 用 360 度减去从终点到当前位置的增量后剩余的值，根据其符号移动相应的量。 | 用 360 度减去从终点到当前位置的增量后剩余的值，在附近移动相应的量。 | 与通常的直线轴相同，根据符号，移动从终点到当前位置的移动量（不足 360 度）。 |     |
| INC 指令          | 将当前位置向指定符号的方向移动指定的增量。                  |                                      |                                          |     |
| 参考点<br>返回       | 到中间点的移动，是基于绝对指令或增量指令。                  |                                      |                                          |     |
|                 | 从中间点到参考点，通过 360 度以内的移动进行返回。            |                                      | 向参考点方向移动相当于从中间点到参考点之间的量，完成返回。            |     |



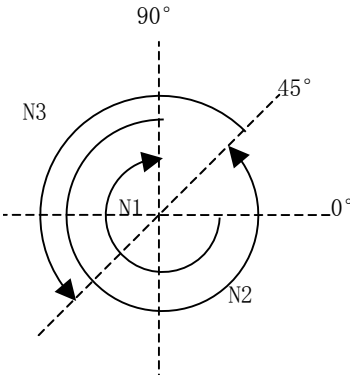
动作例

表示根据旋转坐标的种类而进行的动作的不同与计数器的显示例。

(工件偏移为0° 的设定。)

(1) 旋转型 (#1090=0) + 接近无效 (#1089=0)

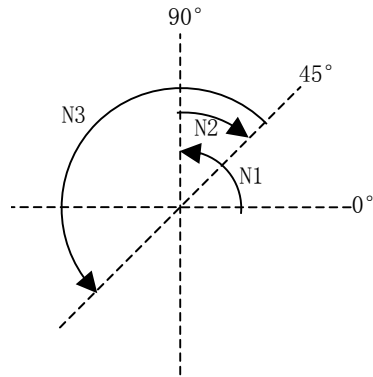
- (a) 工件坐标位置、机械坐标位置、相对位置中的任何一个均以0~359.999° 的范围加以显示。
- (b) 进行绝对指令时，根据符号进行移动。



| 程序            | 工件      | 机械      |
|---------------|---------|---------|
| G28 C0.       |         |         |
| N1 G90 C-270. | 90.000  | 90.000  |
| N2 C405.      | 45.000  | 45.000  |
| N3 G91 C180   | 225.000 | 225.000 |

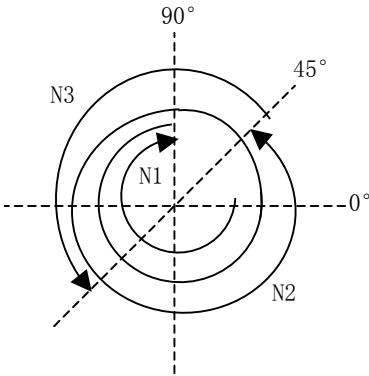
(2) 旋转型 (#1090=0) + 接近有效 (#1089=1)

- (a) 机械坐标位置、工件坐标位置、相对位置中的任何一个均以0~359.999° 的范围加以显示。
- (b) 绝对指令时，向到终点的移动量较小的方向进行移动。



| 程序            | 工件      | 机械      |
|---------------|---------|---------|
| G28 C0.       |         |         |
| N1 G90 C-270. | 90.000  | 90.000  |
| N2 C405.      | 45.000  | 45.000  |
| N3 G91 C180   | 225.000 | 225.000 |

- (3) 旋转型 (#1090=1时)
- (a) 机械坐标位置、工件坐标位置、相对位置中的任何一个均以 $0\sim\pm 99999.999^{\circ}$  的范围加以显示。
  - (b) 绝对值指令中，超过 $360^{\circ}$  的指令与直线轴进行相同的移动。  
(例) 指令 $420^{\circ}$  ，则在通过 $360^{\circ}$  的位置之后，移动到 $60^{\circ}$  的位置。
  - (c) 参考点返回中，到中间点的移动与直线轴进行相同的动作。  
从中间点到参考点，仅旋转到参考点为止的差值，进行返回。
  - (d) 绝对位置检测时，如果重新接通电源，则在关闭电源时的位置进行起动。



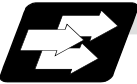
| 程序            | 工件       | 机械       | 相对位置     |
|---------------|----------|----------|----------|
| G28 C0.       |          |          |          |
| N1 G90 C-270. | -270.000 | -270.000 | -270.000 |
| N2 C405.      | 405.000  | 405.000  | 405.000  |
| N3 G91 C180   | 585.000  | 585.000  | 585.000  |

重新接通电源 ↓

| 工件      | 机械      |
|---------|---------|
| 585.000 | 585.000 |

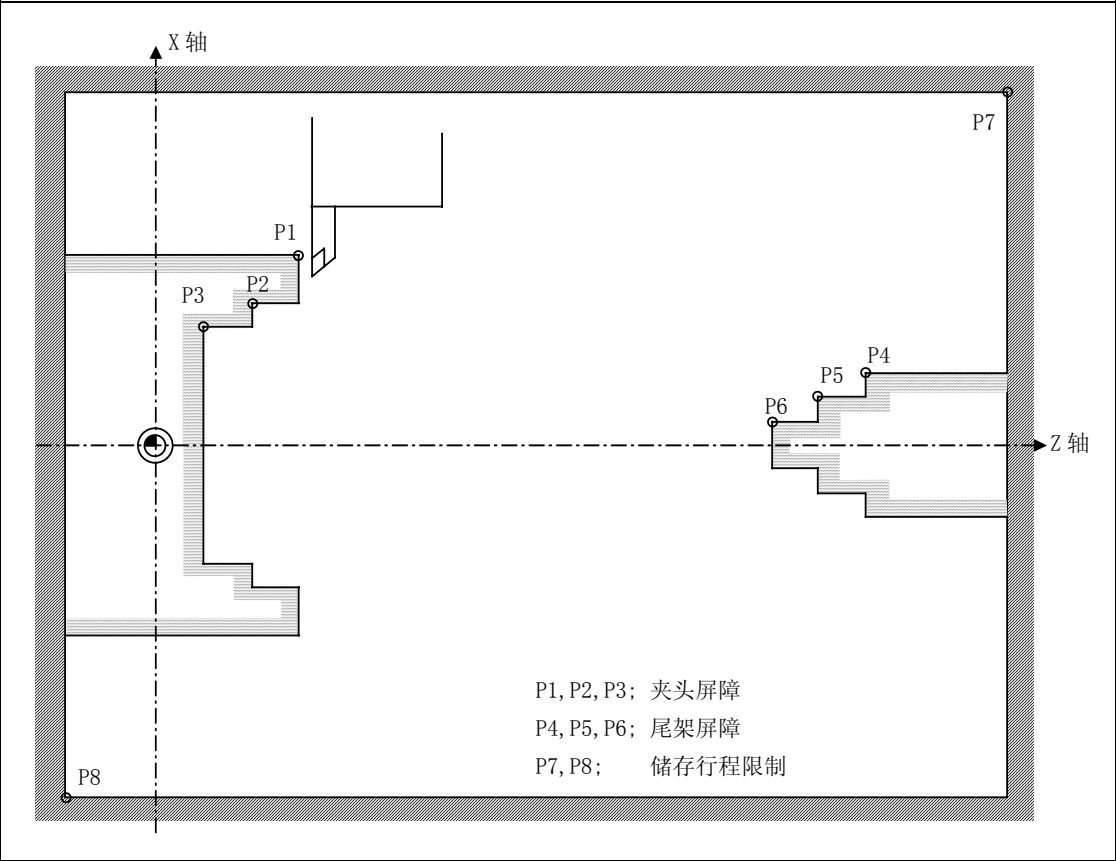
15. 保护功能

15.1 夹头屏障/尾架屏障检查；G22、G23



功能及目的

夹头屏障、尾架屏障，是通过限制刀刀点的移动范围，防止因为程序错误而导致与夹头屏障、尾架屏障发生冲突。对于超过了参数中所设定区域的移动指令，自动在屏障区域停止。



指令格式

|       |   |      |
|-------|---|------|
| G22 ; | : | 屏障有效 |
| G23 ; | : | 屏障无效 |

请在单独节中指令G22、G23指令。



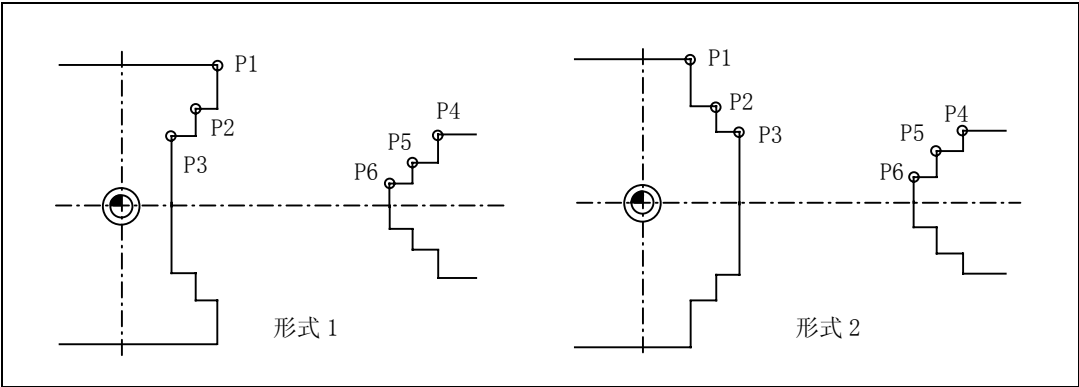
详细说明

- (1) 当机械即将超过这一区域时，机械停止，同时显示警报。  
通过重新启动解除该警报。
- (2) 本功能在机床锁定时也有效。
- (3) 当设定了夹头屏障、尾架屏障的轴全部完成参考点回归时，本功能有效。
- (4) 有储存行程检查功能，且设定了储存行程限制区域时，夹头屏障/尾架屏障功能与储存行程检查功能同时有效。
- (5) 可通过PLC外部信号，分别设定左右的详细屏障区域。



夹头屏障/尾架屏障的设定

- (1) 使用G22、G23时



- (a) 夹头屏障、尾架屏障均可作为参数进行3点输入，以机械坐标进行设定。  
点P1、P2、P3为夹头屏障，点P4、P5、P6为尾架屏障。
- (b) 屏障区域是关于Z轴对称的形状，当屏障点P<sub>i</sub>的X轴坐标为负值时，将符号反转到正端进行换算、检查。  
此外，各屏障点的X轴坐标绝对值，必须按如下的方式设定。  
$$P1 \geq P2 \geq P3, \quad P4 \geq P5 \geq P6$$
  
(但是，Z轴坐标无需满足此条件。)

(2) 使用PLC外部信号输入时

设定从“#8300 P0”到“#8314 P10”的各点坐标值。

P0为夹头・尾架屏障的基准X坐标。以半径值设定基本机械坐标系中的工件中心坐标。

P1~P10的X轴，是以半径值设定距工件中心（P0）的坐标值。另外，基本机械坐标系的坐标设定Z轴。

屏障区域为关于P0点对称的形状。

各点的X轴坐标必须设定如下。

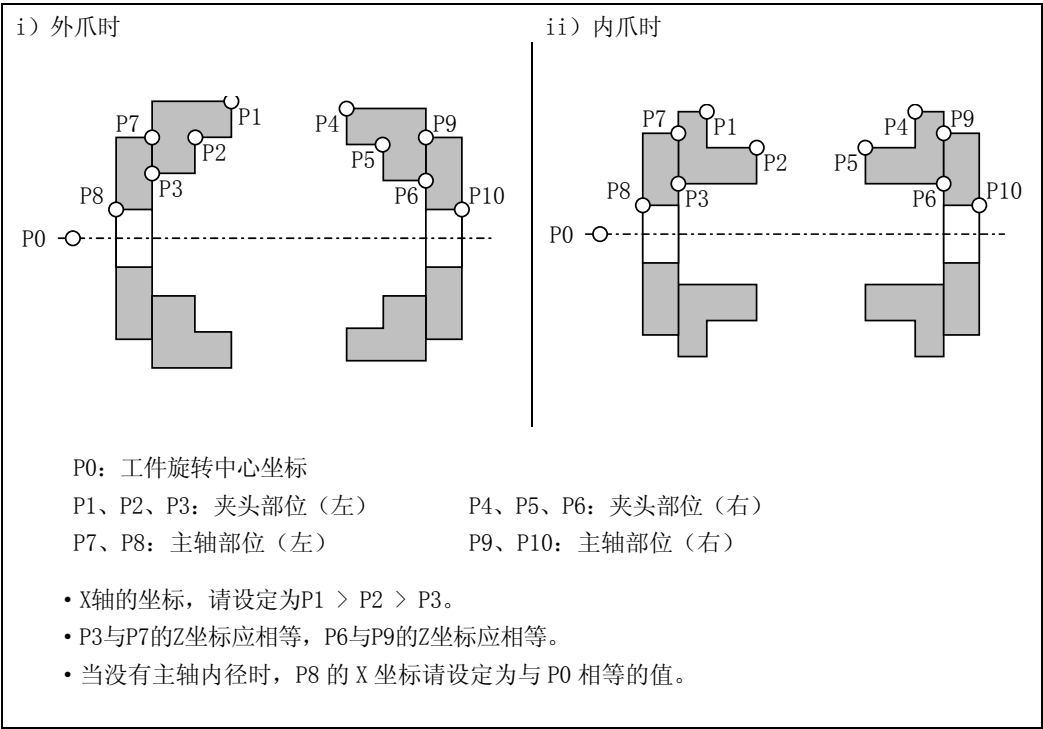
$$\begin{aligned} P1 &\geq P2 \geq P3 \\ P7 &\geq P8 \end{aligned}$$

$$\begin{aligned} P4 &\geq P5 \geq P6 \\ P9 &\geq P10 \end{aligned}$$

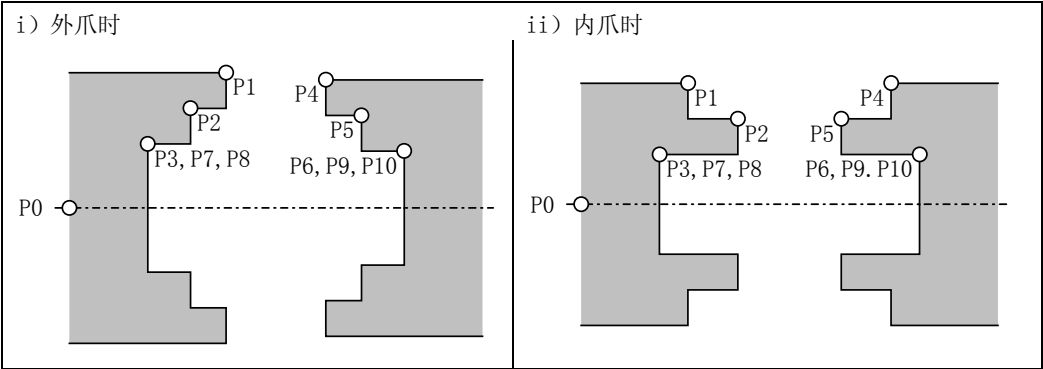
另外，P8的Z轴坐标应小于P1~P3，P10的Z轴坐标必须大于P4~P6。

夹头/尾架上的P0~P10各点位置如下。

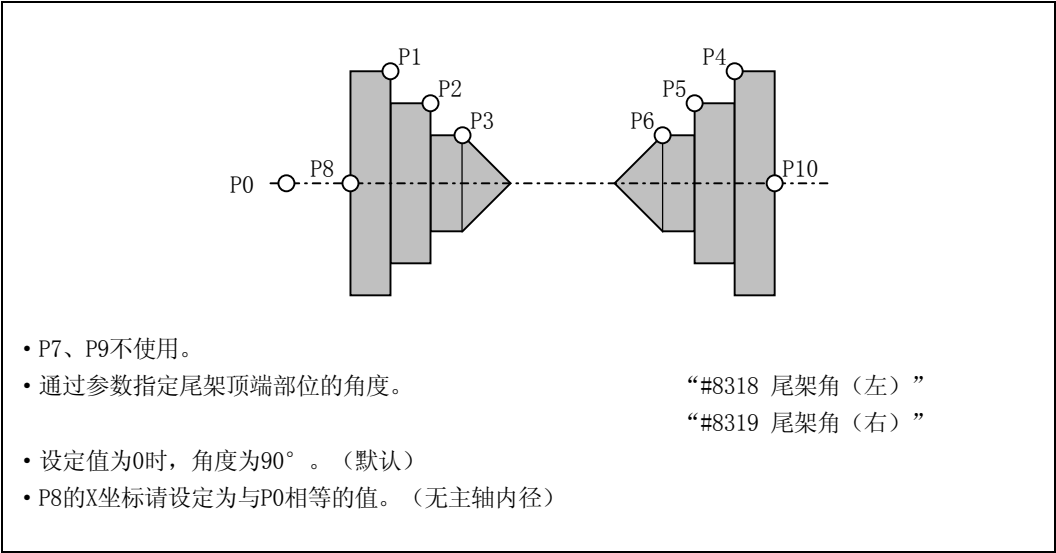
(a) 夹头的设定



当没有象以往的屏障功能那样设定主轴部位时，请在P3、P7、P8中设定相等的值，在P6、P9、P10中设定相等的值。此时的屏障区域如下。



(b) 尾架的设定

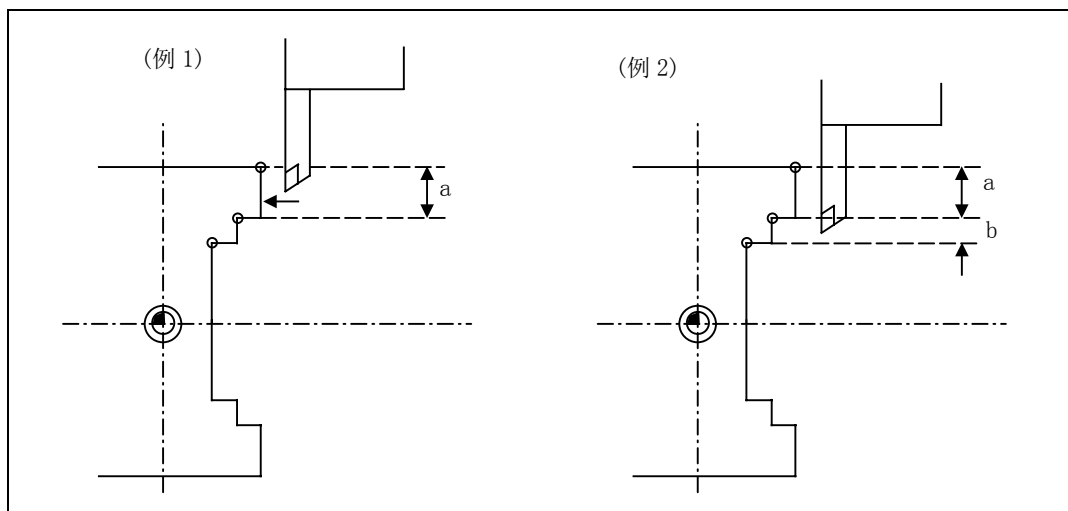




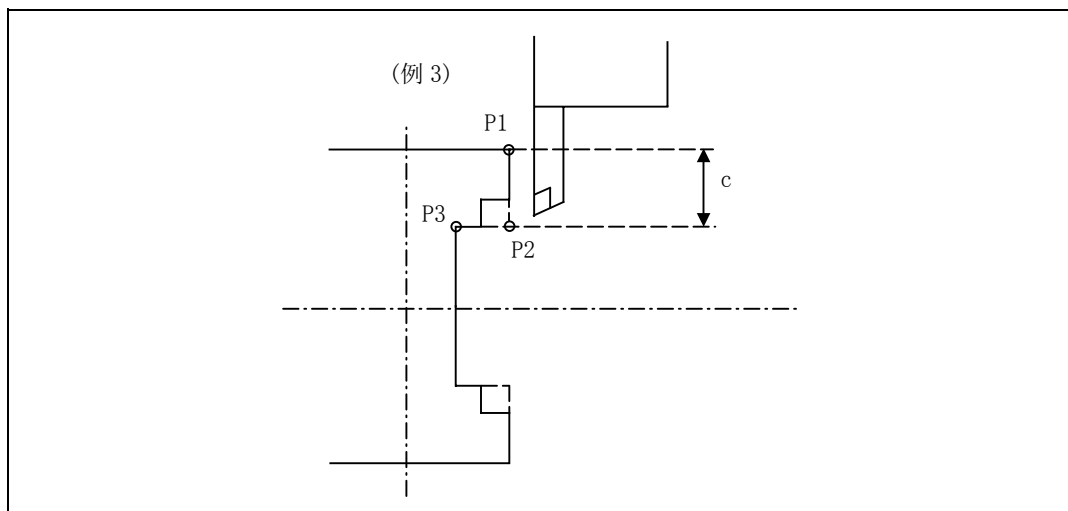
## 限制事项

(1) 相对于夹头屏障/尾架屏障，刀具检查点只有1点，所以，必须注意如下要点。

在下例中，将屏障点设定为通过虚拟刀刃点进行检查，向图中箭头方向移动时，在例1中，由于检查点在a的范围内，所以在屏障边界自动停止，而在例2中，由于检查点在b的范围内，所以在a的范围内，刀具与夹头可能会发生冲突。



为了避免出现这种情况，例如，如例3所示，设定屏障点P1、P2、P3，让检查点处于c的范围内，则可以让刀具停止在屏障边界上。



(2) 进入了屏障区域，发生警报时，在重新起动解除警报之后，可向移入方向的反方向移动。

(3) 没有参考点回归功能的轴，没有屏障区域。因此，该轴不会导致屏障警报。

(4) 进入了被取消的屏障区域之后，当屏障变为有效时，如果试图移动，则立即报警。

此时，通过重新起动解除警报之后，请在让屏障无效（G23）之后进行回避，或是变更各屏障点的设定值。

(5) 即使屏障无效（G23），软件限制也有效。

15.2 储存行程限制（仅G代码系列6、7）



功能及目的

可通过储存行程限制功能，设定刀具的禁止进入区域。通过参数设定这些区域。  
使用G22、G23，可在程序中通过指令变更储存行程限制II功能的禁止区域（参数），或是进行开/关控制。



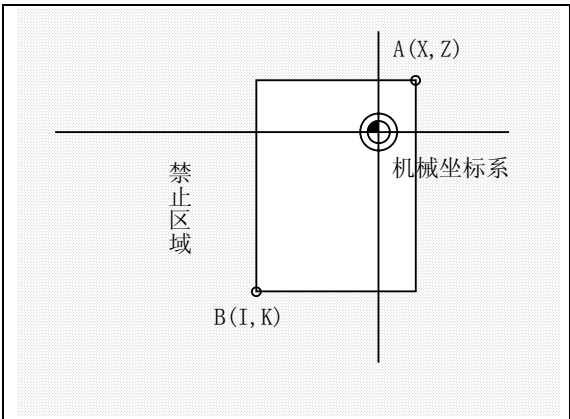
指令而是

|                               |                                         |
|-------------------------------|-----------------------------------------|
| <b>G22 X_ Z_ C_ I_ J_ K_;</b> |                                         |
| G22                           | : 储存行程限制II功能的参数变更<br>储存行程限制II功能 打开（所有轴） |
| G23                           | : 储存行程限制II功能 关闭（所有轴）                    |
| X、Z、C（第1～第3轴的轴名称）             | : 储存行程限制 +端指定                           |
| I、J、K（平面选择的轴名称）               | : 储存行程限制 -端指定                           |

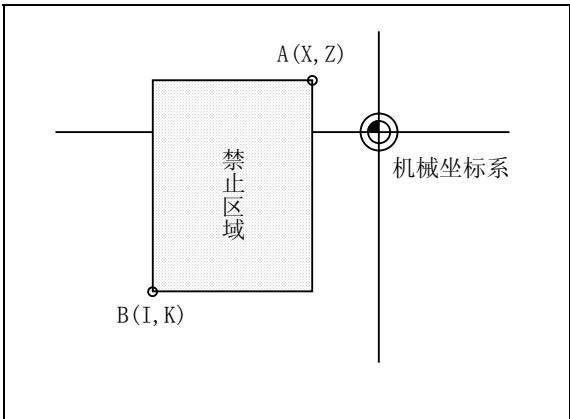


详细说明

请在单独节中指令G22、G23指令。



禁止区域在外部时



禁止区域在内部时（仅 E68）

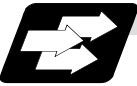


#### 注意事项

- (1) 将储存行程限制的禁止区域最大值与最小值设定为同一数据，则会出现如下状况，请加以注意。
  - (a) 当在最大值、最小值中设定为0时，如果外侧为禁止区域，则全部变为禁止区域，如果内侧为禁止区域，则全部变为可动区域。
  - (b) 将最大值、最小值设定为0以外的数据时，全部变为可动区域。
- (2) 根据储存行程限制 II 的程序指令进行的设定 (G22 • G23)，在 G 代码系列 6 • 7 (cmdtyp=7、8) 中变为有效。
- (3) G22 • G23 为非模态 (组 0)，仅对进行该指令的节有效。

16. 测量辅助功能

16.1 自动刀具长度测定：G37



功能及目的

发出从测量开始位置到测定位置的指令值，让刀具向测定位置方向移动，刀具到达传感器则将机械停止，自动计算此时的坐标值与指令中的测定位置坐标值之间的差值，作为该刀具的补偿量。

另外，当已经进行了刀具补偿时，在已进行补偿的状态下向测定位置方向移动，当测定、计算出的结果是需进一步进行补偿时，则对当前的磨损补偿量进行进一步补偿。



指令格式

**G37** α \_ R\_ D\_ F\_ ;

α       : 是测量轴地址及测定位置的坐标值。.....X, Z

R       : 指定以测量速度开始移动的点与到测定位置的距离。（半径值固定・增量值）

D       : 指定刀具应停止的范围。（半径值固定・增量值）

F       : 指定测量速度。

          当省略 R\_、D\_、F\_ 时，使用参数中设定的值。

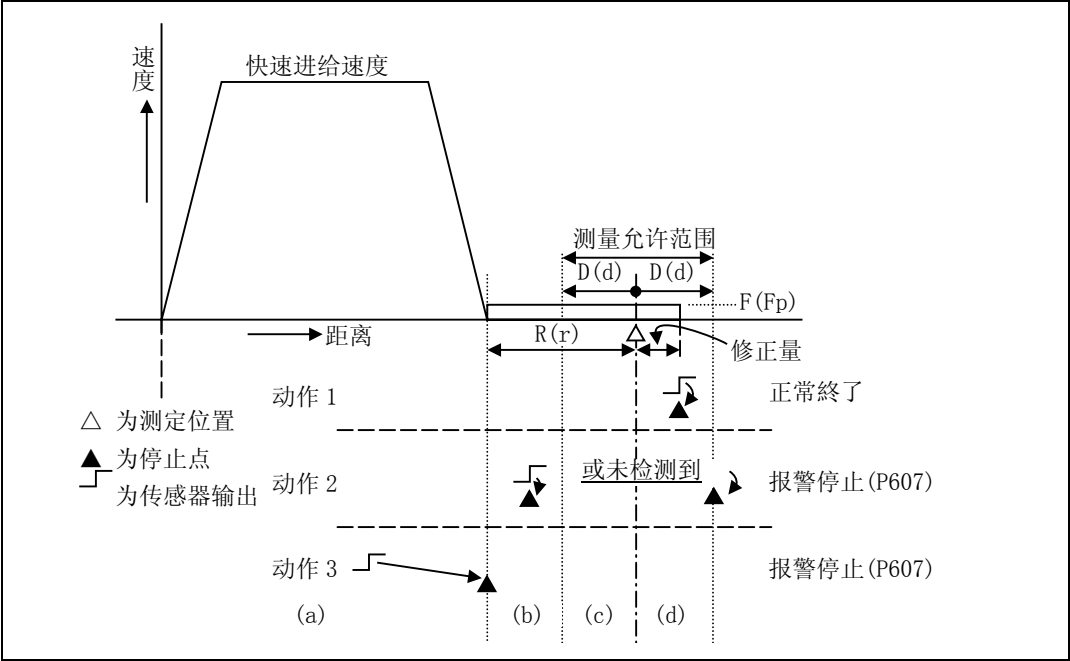
          <参数>（加工参数画面的“自动刀具长度测定”）

- #8004  测量速度       0~60000（mm/min）
- #8005  减速区域 r    0~99999.999（mm）
- #8006  测量区域 d    0~99999.999（mm）



详细说明

(1) G37指令时的动作



(2) 传感器信号（到达测定位置信号）与跳跃信号共用。

(3) 当F指令及参数的测量速度为0时，进给速度为1mm/min。

(4) 在同步进给模式中，以同步进给 [mm/rev] 移动。

(5) 更新后的补偿例，从G37指令的下一T指令开始生效。

(6) 在PLC端去除传感器信号处理中的延迟与误差，为0~0.2ms。

因此，产生如下所述的测量误差。

$$\text{最大测量误差}[\text{mm}] = \text{测量误差}[\text{mm/min}] \cdot \frac{1}{60} \cdot \frac{0.2[\text{ms}]}{1000}$$

(7) 根据传感器信号的检测，读取当时的机械位置坐标，机械在进行相当于伺服偏移量的过走后停止。

$$\text{最大过走量}[\text{mm}] = \text{测量速度}[\text{mm/min}] \cdot \frac{1}{60} \cdot \frac{1}{\text{位置回路增益}[\text{1/s}]}$$

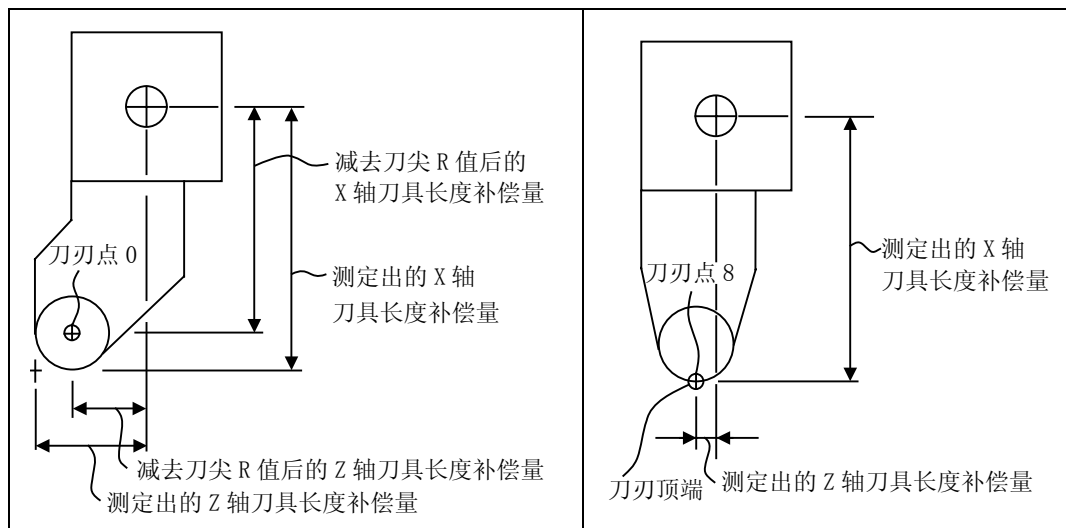
标准位置回路增益为33[1/s]。



## 注意事项

- (1) 如果在未附加自动刀具长度测定功能的机种上指令了G37，则发生程序错误（P600）。
- (2) 如果在G37的节中未指令轴，或是指令了2根以上的轴，则发生程序错误（P604）。
- (3) 如果在G37的节中指令了T代码，则发生程序错误（P605）。但是，T的后1位或后2位为0时，发生（4）的错误。
- (4) 如果G37的节之前未指令T代码，则发生程序错误（P606）。另外，即使指令了T代码，当T的后1位或后2位为0时，发生程序错误（P606）。
- (5) 如果输入了超过测量允许范围的传感器信号，或是即使到达了终点，也没有检测到传感器信号，则发生程序错误（P607）。但是，在详细说明（1）的动作3中，在传感器信号打开的状态下，没有（b）区域时，看作为正常测定。
- (6) 以测量速度移动过程中进入手动插入时，请务必在返回插入前的位置之后，再重新启动。
- (7) 请确保G37中指令的数据或参数设定数据满足以下条件。  
| 测定点—开始点 | > R指令或参数r > D指令或参数d
- (8) 在上述（7）中，当D指令及参数d为0时，仅当指令的测定点与传感器信号检测点一致时，正常结束。其他情况下，发生程序错误（P607）。
- (9) 在上述（7）中，当R指令、D指令、参数r、参数d全部为0时，定位到指令的测定点之后，不管是否有传感器信号，都发生程序错误（P607）。
- (10) 测定信号距离 < 测量允许范围时，变为所有测量允许范围。

- (11) 测定指令距离 < 测量速度移动距离时，全部以测量速度进行移动。
- (12) 测量允许范围 > 测量速度移动距离时，以测量速度在测量允许范围内移动。
- (13) 在指令G37之前，请务必取消刀尖R补偿。
- (14) 不考虑刀尖R的值及刀刃点编号，进行刀具长度补偿的计算。  
 当希望刀刃点编号为0时，请从测量出的刀具长度补偿量中，减去刀尖R的值。  
 当刀具的刀刃点编号（刀刃形状）为5、6、7、8时，请在刀具的顶端进行刀具长度的测量。



16.2 跳跃功能：G31



功能及目的

通过G31指令进行直线插补过程中，如果从外部输入跳跃信号，则立即停止机械的进给，放弃剩余距离，开始执行下一节的指令。



指令格式

|                             |                       |
|-----------------------------|-----------------------|
| <b>G31 Xx/Uu Zz/Ww Ff ;</b> |                       |
| x, z, u, w                  | : 各轴坐标值。以绝对值或增量值进行指令。 |
| f                           | : 进给速度 (mm/min)       |

可通过本指令进行直线插补。在执行本指令时，如果从外部输入跳跃信号，则机械停止，取消剩余指令，从下一节开始执行。



详细说明

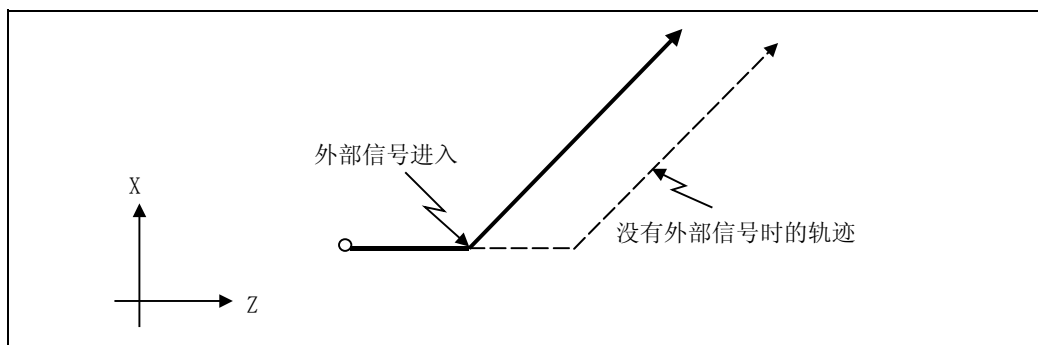
- (1) 如果指令了Ff，则进给速度为指令速度f，如果没有指令Ff，则以参数“#1174 skip\_F”所设定的值作为进给速度。但是，任何场合下，F模态不会被更新。
- (2) G31节中不进行自动加减速。
- (3) G31指令时，超程无效，固定为100%。空运转也无效。但是停止条件（回馈等待、互锁、超程零、行程结束）有效。外部减速也有效。
- (4) G31指令为模态指令。因此，此时必须进行指令。
- (5) G31指令开始时，如果输入跳跃信号，则G31指令立即完成。  
另外，G31节结束之前，当没有输入跳跃信号时，在移动指令完成后，G31指令也完成。
- (6) 如果在刀尖R补偿中进行G31指令，则发生程序错误（P608）。
- (7) 当G31指令中没有G指令，参数速度也为0时，发生程序错误（P603）。
- (8) 机床锁定时，或在Z轴取消开关打开状态下仅指令Z轴，则跳跃信号被跳跃，执行到节的最后为止。



## G31 执行例

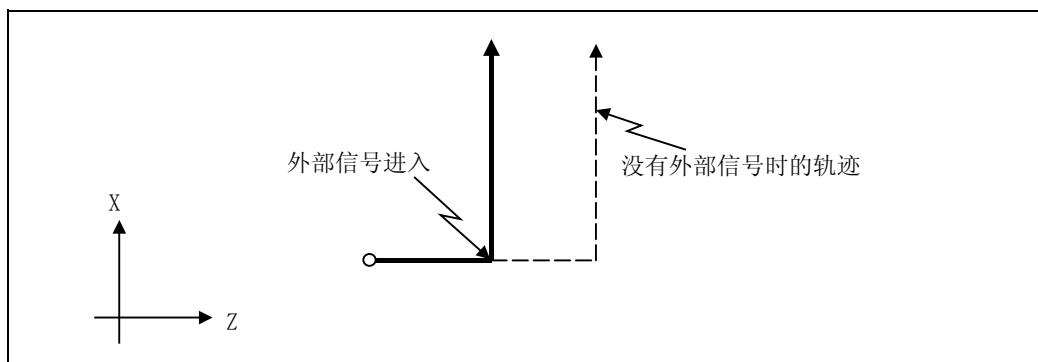
(例 1) 以下节为增量值指令

```
G31 Z1000 F100;
G01 U2000 W1000;
```



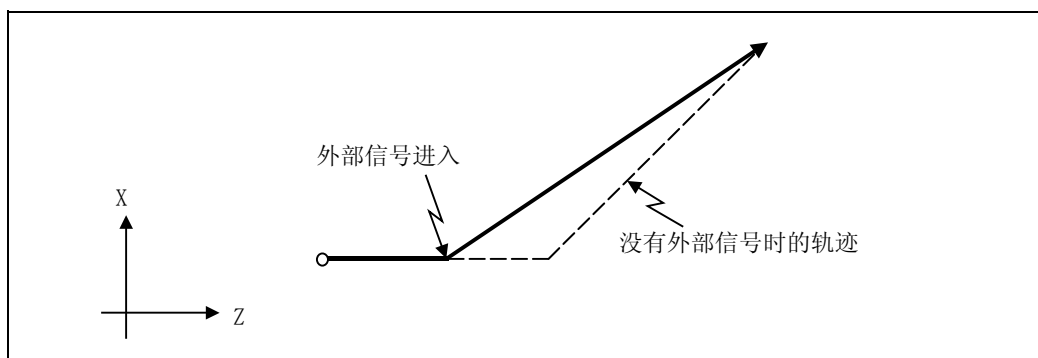
(例 2) 以下节为通过绝对值进行 1 轴的移动指令

```
G31 Z1000 F100;
G01 X1000;
```



(例 3) 以下节为通过绝对值进行 2 轴的移动指令

```
G31 Z1000 F100;
G01 X1000 Z2000;
```





详细说明（读取跳跃坐标）

由于输入跳跃信号的坐标位置被储存在系统变量#5061（第1轴）～#5063（第3轴）中，所以可在用户宏中使用。

```
}
G00 X-100. ;
G31 X-200. F60 ;
#101=#5061
}
```

忽略指令

将忽略信号输入坐标值（工件坐标系）读取到变量#101 中。



详细说明（G31 惯性移动量）

G31指令中，从输入跳跃信号到停止之间的惯性移动量，因参数（#1174 skip\_F）及G31中的F指令而异。由于从开始应答跳跃信号，到减速停止之间的时间很短，所以能够进行惯性移动量小、精度高的停止。惯性移动量按照如下的计算公式计算得出。

$$\delta_o = \frac{F}{60} \times T_p + \frac{F}{60} \times (t_1 \pm t_2)$$
$$= \underbrace{\frac{F}{60} \times (T_p + t_1)}_{\delta_1} \pm \underbrace{\frac{F}{60} \times t_2}_{\delta_2}$$

$\delta_o$  : 惯性移动量 (mm)

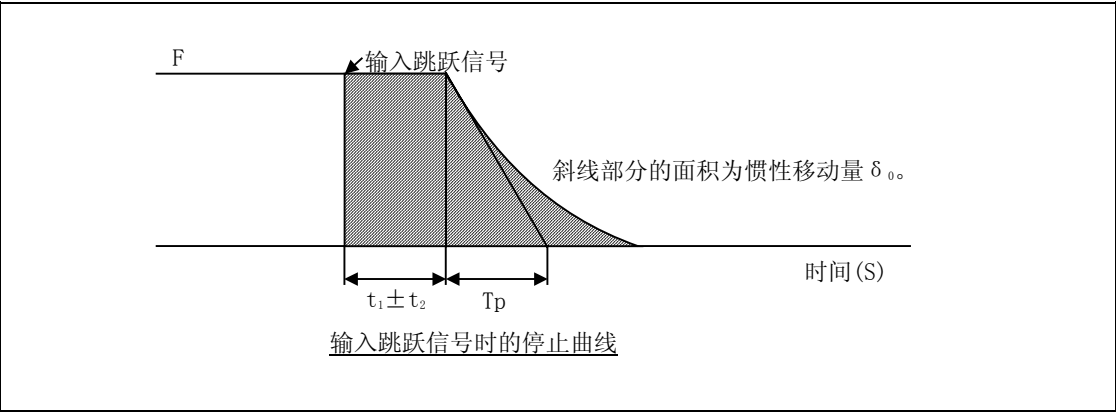
F : G31 跳跃速度 (mm/min)

$T_p$  : 位置回路时间常数 (s) = (位置回路增益)

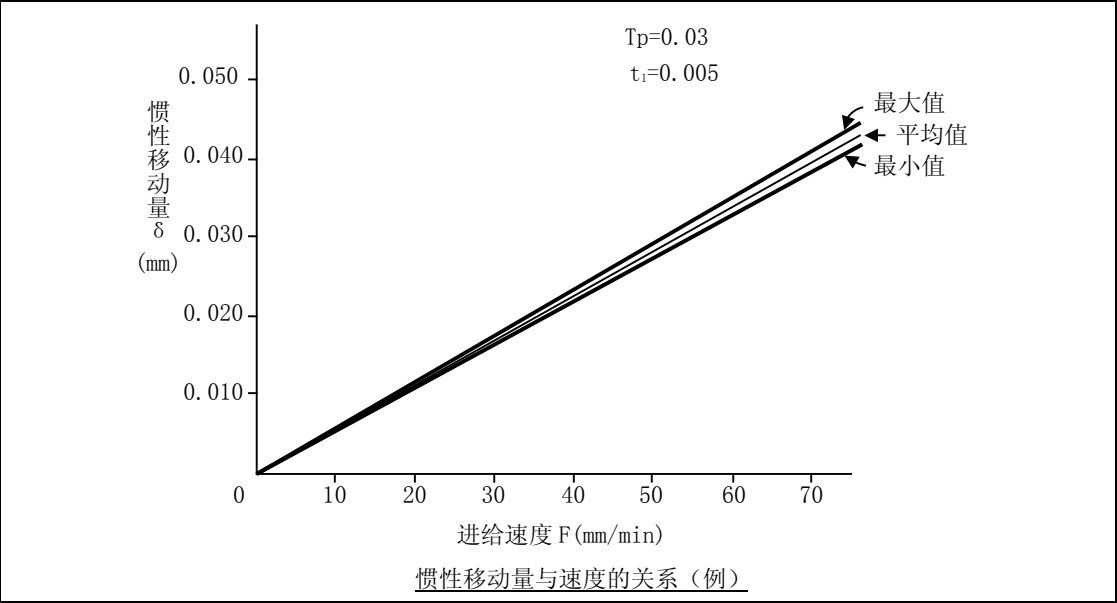
$t_1$  : 应答延迟时间 (s) = (从检测到跳跃信号，到通过 PC 到达控制装置的时间)

$t_2$  : 应答误差时间 0.001 (s)

当在测量等中使用G31 指令时，可进行上式的  $\delta_1$ 测量值的补偿，  $\delta_2$ 成为测量误差。



当 $T_p=30\text{ms}$ 、 $t_i=5\text{ms}$ 时，速度与惯性移动量的关系如下图所示。

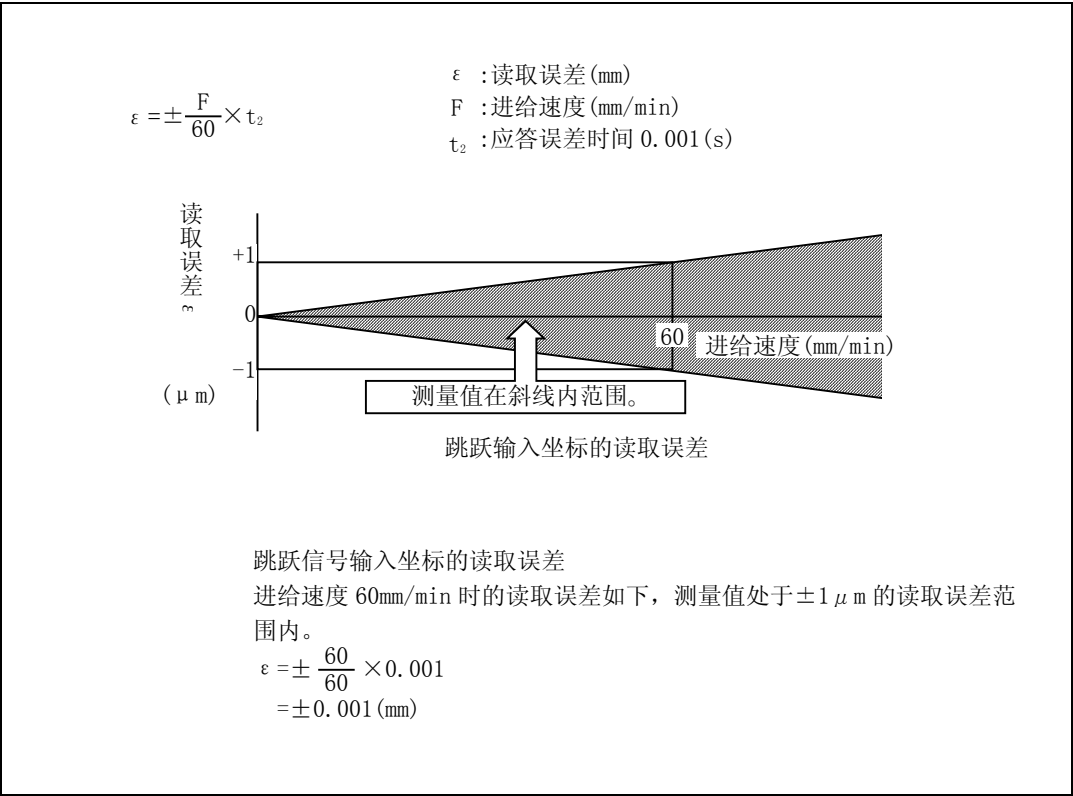




详细说明（跳跃坐标的读取误差 mm）

（1）读取跳跃信号输入坐标

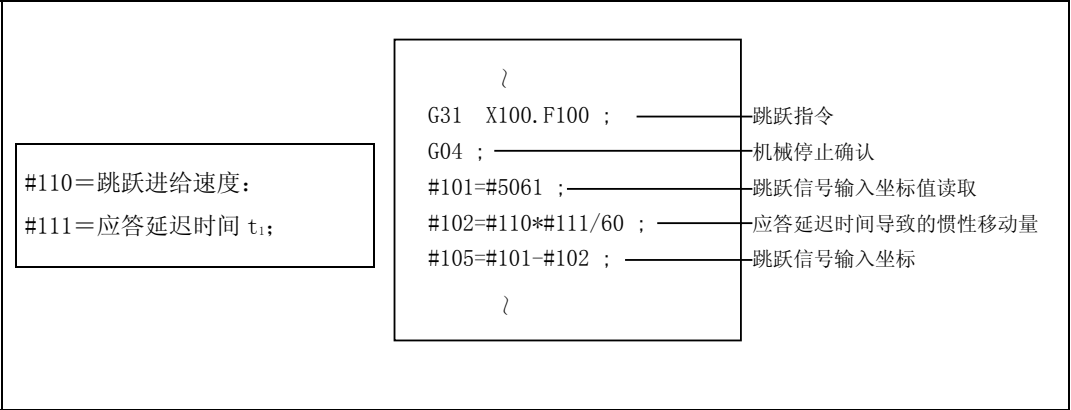
跳跃信号输入坐标值中，不包含因位置回路时间常数Tp及切削进给时间常数Ts而导致的惯性移动量。因此，可在以下公式的误差范围内，读取输入跳跃信号时的工件坐标值，作为跳跃信号输入坐标值。但是，由于因应答延迟时间t1而导致的惯性移动量会成为测量误差，所以请进行补偿。



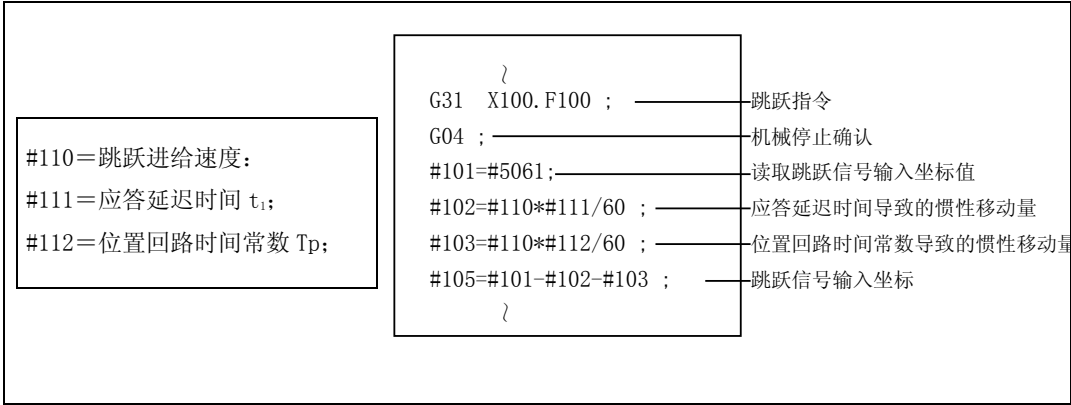


惯性移动量的补偿例

(1) 跳跃信号输入坐标值的补偿



(2) 工件坐标值的补偿



16.3 多级跳跃功能1；G31.n、G04



功能及目的

通过预先设定输入的跳跃信号组合，可在各种条件下进行跳跃。跳跃动作与G31相同。  
可进行跳跃指定的G指令包括G31.1、G31.2、G31.3、G04，可通过参数设定各G指令与跳跃信号的对应关系。



指令格式

|                             |   |                 |  |
|-----------------------------|---|-----------------|--|
| <b>G31.1 Xx Zz α α Ff ;</b> |   |                 |  |
| Xx Zz α α                   | : | 指令格式轴坐标字母及目标坐标值 |  |
| Ff                          | : | 进给速度 (mm/min)   |  |

G31.2、G31.3也相同，G04中无需要Ff。  
通过该指令，与G31指令相同，执行直线插补，当满足预先设定的跳跃信号条件时，机械停止，取消剩余指令，执行下一单节。



详细说明

- (1) 取决于参数的进给速度，G31.1与“#1176 skip1f”、G31.2与“#1178 skip2f”、G31.3与“#1180 skip3f”相对应。
- (2) 当各指令满足跳跃信号条件时，被跳跃。
- (3) 除上述 (1) (2) 项以外，与G31指令相同。
- (4) 可通过参数，设定与G31.1、G31.2、G31.3各指令相对应的速度。
- (5) 预先通过参数设定与G31.1、G31.2、G31.3、G04各指令相对应的跳跃条件（为设定的跳跃信号的逻辑和）。

| 参数设定值 | 有效跳跃信号 |   |   |
|-------|--------|---|---|
|       | 1      | 2 | 3 |
| 1     | ○      |   |   |
| 2     |        | ○ |   |
| 3     | ○      | ○ |   |
| 4     |        |   | ○ |
| 5     | ○      |   | ○ |
| 6     |        | ○ | ○ |
| 7     | ○      | ○ | ○ |

（当接收到○信号时中断）



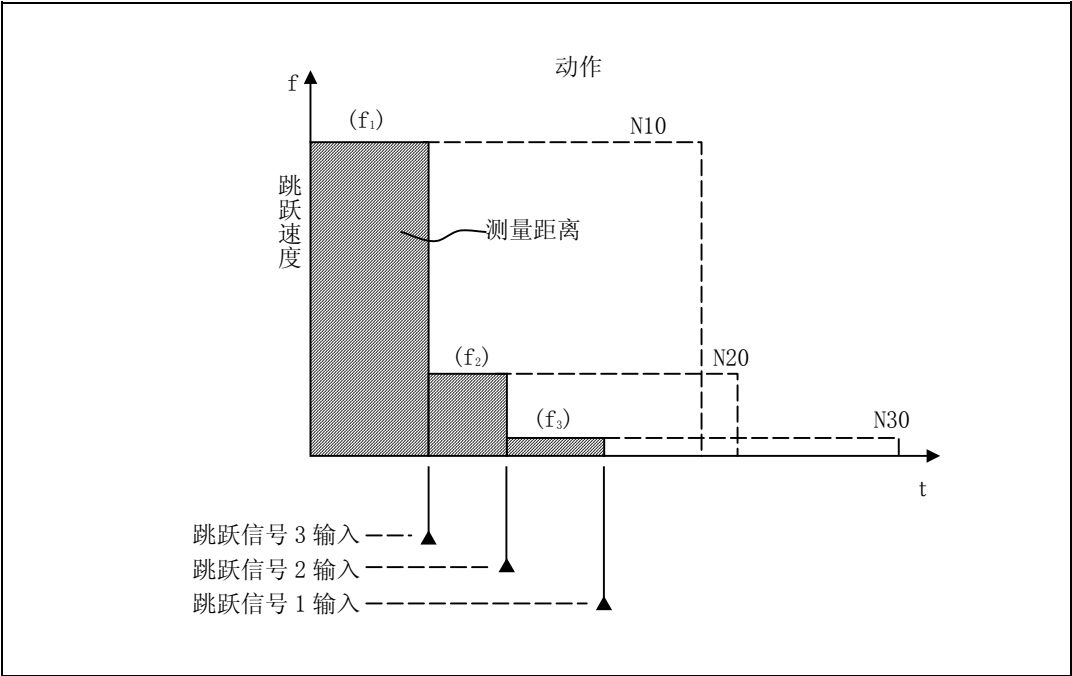
动作

(1) 通过使用多级跳跃，可进行如下的控制，可在提高测量精度的同时，缩短测量时间。

|          |                 |  |
|----------|-----------------|--|
| 参数设定如下时  |                 |  |
| 跳跃条件     | 跳跃速度            |  |
| G31.1 :7 | 20.0mm/min (f1) |  |
| G31.2 :3 | 5.0mm/min (f2)  |  |
| G31.3 :1 | 1.0mm/min (f3)  |  |

程序例

```
N10 G31.1 X200.0;
N20 G31.2 X40.0;
N30 G31.3 X1.0;
```



(注 1) 在上述动作中，当在跳跃信号 3 之前输入了跳跃信号 1 时，N20 被跳跃，N30 也被跳跃。

(2) 如果在G04（延时）中输入了所设定条件的跳跃信号，则取消延时的剩余时间，执行下一单节。

16.4 多级跳跃功能2



功能及目的

通过跳跃指令（G31）进行直线插补中（区别来自外部的跳跃信号1~4），可在跳跃信号指令Pp的条件下进行跳跃。

另外，在延时指令（G04）中，通过参数“#1173 dwlskp”所设定（区别来自外部的跳跃信号1~4）的跳跃条件下，取消停止剩余时间，执行下一单节。



指令格式

|                              |                   |
|------------------------------|-------------------|
| <b>G31 Xx Zz α α Pp Ff ;</b> |                   |
| Xx Zz α α                    | : 指令格式轴坐标字母及目标坐标值 |
| Pp                           | : 跳跃信号指令（参阅下表）    |
| Ff                           | : 进给速度（mm/min）    |



详细说明

- (1) 以指令速度f指令跳跃速度。但是，F模态不会被更新。
- (2) 跳跃信号的指令，通过跳跃信号p进行指定。p在1~15的范围内进行指定。当超出指令范围时，发生程序错误（P35）。

| 跳跃信号指令<br>P | 有效跳跃信号 |   |   |   |
|-------------|--------|---|---|---|
|             | 4      | 3 | 2 | 1 |
| 1           |        |   |   | ○ |
| 2           |        |   | ○ |   |
| 3           |        |   | ○ | ○ |
| 4           |        | ○ |   |   |
| 5           |        | ○ |   | ○ |
| 6           |        | ○ | ○ |   |
| 7           |        | ○ | ○ | ○ |
| 8           | ○      |   |   |   |
| ~           | ~      | ~ | ~ | ~ |
| ~           | ~      | ~ | ~ | ~ |
| 13          | ○      | ○ |   | ○ |
| 14          | ○      | ○ | ○ |   |
| 15          | ○      | ○ | ○ | ○ |

（当接收到○信号时中断）

- (3) 指令的跳跃信号指令，为跳跃信号的逻辑和。

(例) G31 X100. P5 F100;  
          ↑  
          ——输入忽略信号 1 或 3 时，忽略？

(4) 当没有跳跃信号指令Pp时，成为通常的跳跃信号（G31）。另外，当没有速度指令Ff时，则按照G31跳跃速度参数所设定的跳跃速度。

表2. 跳跃与多级跳跃的关系

| 跳跃规格                    | ×              |     | ○    |     |
|-------------------------|----------------|-----|------|-----|
|                         | 条 件            | 速 度 | 条 件  | 速 度 |
| G31 X100;<br>(无 P、F)    | 程序错误<br>(P601) |     | 跳跃 1 | 参数  |
| G31 X100 P5;<br>(无 F)   | 程序错误<br>(P602) |     | 指令值  | 参数  |
| G31 X100 F100;<br>(无 P) | 程序错误<br>(P601) |     | 跳跃 1 | 指令值 |
| G31 X100 P5 F100;       | 程序错误<br>(P602) |     | 指令值  | 指令值 |

(注 1) 表中的“参数”是 G31 跳跃速度的参数。

(5) 当跳跃规格有效，且轴地址中使用了P时，以跳跃信号指令P为优先。轴地址P被跳跃。

(例) G31 P500. F100;  
          ↑  
          |  
          └─ 看作为忽略信号指令。  
              (发生程序错误 (P35))

(6) 除上述 (1) ～ (5) 项以外，与跳跃功能（无G31的P）相同。

16.5 可编程电流限制

[E68]



功能及目的

可在程序中将伺服轴的电流限制值变更为任意值的功能，可用于工件的推顶等。  
以限制电流相对于额定电流的比例的方式，指定电流限制值。



指令格式

|                     |                        |
|---------------------|------------------------|
| <b>G10 L14 Xn ;</b> |                        |
| L14                 | : 限制值的设定（+端/-端共用）      |
| X                   | : 轴地址                  |
| n                   | : 电流限制值（%） 设定范围： 1~300 |



注意事项・限制事项

- （1）在电流限制有效中，当达到电流限制值时，输出达到电流限制信号。
- （2）达到电流限制后的动作，根据外部信号不同，有以下2种模式。根据外部信号，决定进入哪一种状态。
  - 通常模式  
直接执行移动指令。  
在自动运转中，执行完移动指令，在保持偏移量的状态下进入下一单节。
  - 互锁模式  
在发生偏移时，进入内部互锁状态，不进行下一单节的移动。  
在自动运转中，停止该节，进入下一单节。  
在手动运转中，跳跃之后的同一方向的指令。
- （3）通过解除外部信号的电流限制切换信号，可解除因电流限制而导致的位置偏移。（但是，轴不应处于移动中）
- （4）电流限制值的移动范围为1%~300%。超过该范围，则发生程序错误（P35）。
- （5）利用G10指令指定小数点时，仅整数部分有效。  
例）G10 L14 X10.123； 电流限制值被设定为10%
- （6）轴名称为“C”的轴，无法进行由程序（G10指令）设定电流限制值。  
当通过程序进行设定时，请以增量轴名称设定轴地址，或是将轴名称设定为“C”以外的名称。

# 附录

## 附录 1. 程序参数输入 N 编号对应表

### 附录1. 程序参数输入N编号对应表

(注 1) 表中的单位，表示参数数据的最小设定单位。

(注 2) 表中的设定范围为画面上的设定范围。与长度相关的参数，请以输入设定单位的 2 倍加以指定。

(例1) 在公制系统中，当输入设定单位为“B”（0.001mm）时，在参数中设定30mm。

L60000

(例2) 在英制系统中，当输入设定单位为“B”（0.001mm）时，在参数中设定5inch。

L100000

(注 3) 二进制型的参数请转换为字节型数据，以地址 D 后面的 10 进制数据进行指令。

(例1) 二进制数据

$01010101_B = 55_{10} = 85_D \cdots \cdots$  指令85

(例2) ASCII代码

“M” =  $01001101_B = 4D_{10} = 77_D \cdots \cdots$  指令77

(B表示Binary (2进制)、H表示Hexa (16进制)、D表示Decimal (10进制)。)

[加工参数]

参数1.

| #    | 项目       | P  | A | N    | 数据型 | 设定范围        | (单位)     | 备注 |
|------|----------|----|---|------|-----|-------------|----------|----|
| 8007 | 超程       | 11 | - | 1010 | D   | 0~100       | (%)      |    |
| 8008 | 最大角度     | 11 | - | 756  | L   | 0~180       | (°)      |    |
| 8009 | 角前长度     | 11 | - | 760  | L   | 0~99999.999 | (mm)     |    |
| 8010 | 最大值      | 11 | - | 776  | L   | 0~99.999    | (mm)     |    |
| 8011 | 最大累加值    | 11 | - | 780  | L   | 0~99.999    | (mm)     |    |
| 8013 | G83 返回   | 11 | - | 832  | L   | 0~99999.999 | (mm)     |    |
| 8014 | 斜切量      | 11 | - | 1012 | D   | 0~127       | 0.1 lead |    |
| 8015 | 角度       | 11 | - | 1011 | D   | 0~89        | (°)      |    |
| 8016 | G71 最小切入 | 11 | - | 788  | L   | 0~99.999    | (mm)     |    |
| 8017 | 切入变化     | 11 | - | 792  | L   | 0~99.999    | (mm)     |    |
| 8051 | G71 切入   | 11 | - | 784  | L   |             | (mm)     |    |
| 8052 | 余量       | 11 | - | 796  | L   |             | (mm)     |    |
| 8053 | G73 切削 X | 11 | - | 800  | L   |             | (mm)     |    |
| 8054 | 切削 Z     | 11 | - | 804  | L   |             | (mm)     |    |
| 8055 | 次数       | 11 | - | 808  | L   |             | (次)      |    |
| 8056 | G74 返回   | 11 | - | 820  | L   |             | (mm)     |    |
| 8057 | G76 精整余量 | 11 | - | 824  |     |             | (mm)     |    |
| 8058 | 次数       | 11 | - | 997  | D   | 0~99        | (次)      |    |
| 8059 | 螺纹       | 11 | - | 998  | D   | 0~99        | (°)      |    |

## 附录 1. 程序参数输入 N 编号对应表

[轴参数]

参数1.

| #    | 项目     | P | A   | N   | 数据类型 | 设定范围       | (单位) | 备注 |
|------|--------|---|-----|-----|------|------------|------|----|
| 8202 | 软件限制无效 | 2 | No. | 897 | H2   | 0~1        |      |    |
| 8204 | 软件限制-  | 2 | No. | 916 | L    | ±99999.999 | (mm) |    |
| 8205 | 软件限制+  | 2 | No. | 912 | L    | ±99999.999 | (mm) |    |

[屏障数据]

参数1.

| #    | 项目     | P  | A   | N    | 数据类型 | 设定范围       | (单位) | 备注 |
|------|--------|----|-----|------|------|------------|------|----|
| 8300 | (P0) X | 11 | —   | 1128 | L    | ±99999.999 | (mm) |    |
| 8301 | (P1) X | 11 | No. | 1136 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1160 | L    | ±99999.999 | (mm) |    |
| 8302 | (P2) X | 11 | No. | 1140 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1164 | L    | ±99999.999 | (mm) |    |
| 8303 | (P3) X | 11 | No. | 1144 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1168 | L    | ±99999.999 | (mm) |    |
| 8304 | (P4) X | 11 | No. | 1148 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1172 | L    | ±99999.999 | (mm) |    |
| 8305 | (P5) X | 11 | No. | 1152 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1176 | L    | ±99999.999 | (mm) |    |
| 8306 | (P6) X | 11 | No. | 1156 | L    | ±99999.999 | (mm) |    |
|      | Z      | 11 | No. | 1180 | L    | ±99999.999 | (mm) |    |

[PLC数据]

安装参数6.

| #     | 项目     | P | A | N  | 数据类型 | 设定范围      | (单位) | 备注 |
|-------|--------|---|---|----|------|-----------|------|----|
| #6301 | PLC 常数 | 5 | — | 1  | L    | ±99999999 |      |    |
| ~     |        |   |   | ~  |      |           |      |    |
| #6348 |        |   |   | 48 |      |           |      |    |

[PLC计时器]

安装参数6.

| #     | 项目                      | P | A | N   | 数据类型 | 设定范围    | (单位)  | 备注 |
|-------|-------------------------|---|---|-----|------|---------|-------|----|
| #6000 | 10ms TIMER (T0~T15)     | 6 | — | 0   | S    | 0~32767 | 0.01s |    |
| ~     |                         |   |   | ~   |      |         |       |    |
| #6015 |                         |   |   | 15  |      |         |       |    |
| #6016 | 100ms TIMER (T56~T135)  | 6 | — | 16  | S    | 0~32767 | 0.1s  |    |
| ~     |                         |   |   | ~   |      |         |       |    |
| #6095 |                         |   |   | 95  |      |         |       |    |
| #6096 | 100ms TIMER (T232~T239) | 6 | — | 96  | S    | 0~32767 | 0.1s  |    |
| ~     | (INC.)                  |   |   | ~   |      |         |       |    |
| #6103 |                         |   |   | 103 |      |         |       |    |

# 附录 1. 程序参数输入 N 编号对应表

[PLC计数器]

安装参数6.

| #              | 项目      | P | A | N            | 数据类型 | 设定范围    | (单位) | 备注 |
|----------------|---------|---|---|--------------|------|---------|------|----|
| C0<br>~<br>C23 | PLC 计数器 | 7 | - | 0<br>~<br>23 | L    | 0~32767 |      |    |

[位选择]

安装参数6.

| #                 | 项目  | P | A | N            | 数据类型             | 设定范围                          | (单位)           | 备注 |
|-------------------|-----|---|---|--------------|------------------|-------------------------------|----------------|----|
| 6401<br>~<br>6496 | 位选择 | 8 | - | 1<br>~<br>96 | D<br>or<br>H0~H7 | 0x00~0xFF (注)<br>0~256<br>0~1 | (hex)<br>(dec) |    |

(注) 请将0x00~0xFF数据转换为10进制数字加以使用。

## 附录 2. 程序错误

### 附录2. 程序错误

（粗体的文字是画面上的信息显示。）

是自动运转中发生的警报，当加工程序创建错误以及未创建符合规格的程序时，发生程序错误。

| 错误编号 | 内 容                                                                                                                                                     | 处 理                                                                                                    |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| P 10 | <b>超过联动轴数</b><br>同一单节中所指令的轴地址数量超过规格数量。                                                                                                                  | <ul style="list-style-type: none"> <li>• 将报警单节的指令分割为 2 单节。</li> <li>• 进行规格确认。</li> </ul>               |
| P 11 | <b>轴名称设定不正确</b><br>程序指令的轴地址名与参数设定的轴地址名不同。                                                                                                               | <ul style="list-style-type: none"> <li>• 修正程序的轴名称。</li> </ul>                                          |
| P 20 | <b>计算错误</b><br>指令了指令单位无法理解的轴指令。                                                                                                                         | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                              |
| P 29 | <b>无法指令状态</b> <ul style="list-style-type: none"> <li>• 在无法进行法线控制的模态中，指令了法线控制指令（G40.1、G41.1、G42.1）。</li> <li>• 在无法进行 2 系统同时螺纹切削指令的模态中，指令了该指令。</li> </ul> | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                              |
| P 30 | <b>奇偶校验 H</b><br>纸带上的 1 字母的孔数，采用 EIA 代码时为偶数，采用 ISO 代码时为奇数。                                                                                              | <ul style="list-style-type: none"> <li>• 进行纸带的确认。</li> <li>• 进行纸带打孔机及读带机的确认。</li> </ul>                |
| P 31 | <b>V 校验</b><br>纸带上的 1 单节的字符数为奇数个。                                                                                                                       | <ul style="list-style-type: none"> <li>• 纸带上的 1 单节字符数为偶数个。</li> <li>• 关闭参数的奇偶校验 V 选择。</li> </ul>       |
| P 32 | <b>不正确地址</b><br>使用了规格中所没有的地址。                                                                                                                           | <ul style="list-style-type: none"> <li>• 确认、修正程序地址。</li> <li>• 确认、修正参数值。</li> <li>• 进行规格确认。</li> </ul> |
| P 33 | <b>格式错误</b><br>程序上的指令格式不正确。                                                                                                                             | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                              |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                      | 处 理                                                                                                                                                        |
|------|--------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| P 34 | <b>错误 G 代码</b><br>指令了规格中没有的 G 代码。<br>指令了坐标旋转指令(G68)中无法执行的 G 代码。                                                          | <ul style="list-style-type: none"> <li>• 确认、修正程序的 G 代码地址。</li> </ul>                                                                                       |
| P 35 | <b>超出指令值</b><br>超过了各地址的设定范围。                                                                                             | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                                                                                  |
| P 36 | <b>程序结束错误</b><br>纸带及内存运转中读入了“EOR”。                                                                                       | <ul style="list-style-type: none"> <li>• 在程序的最后加入 M02 及 M30。</li> <li>• 在子程序的最后加入 M99。</li> </ul>                                                          |
| P 37 | <b>标签、N 编号为 0</b><br>在程序编号及序列编号中指定了 0 号。                                                                                 | <ul style="list-style-type: none"> <li>• 在 1~99999999 号的范围内，指定程序编号。</li> <li>• 在 1~99999 号的范围内，指定顺序编号。</li> </ul>                                          |
| P 39 | <b>无规格</b> <ul style="list-style-type: none"> <li>• 指令了规格中没有的 G 代码。</li> <li>• 无高速程序伺服运转的规格。</li> </ul>                  | <ul style="list-style-type: none"> <li>• 确认规格。</li> </ul>                                                                                                  |
| P 40 | <b>预读单节中错误</b><br>执行刀具半径补偿时，由于预读的单节中存在错误，所以无法进行干扰检查。                                                                     | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                                                                                  |
| P 60 | <b>超过插补长度</b><br>指令移动距离过大。（超过 $2^{31}$ ）                                                                                 | <ul style="list-style-type: none"> <li>• 检查轴地址的指令范围。</li> </ul>                                                                                            |
| P 62 | <b>无 F 指令</b> <ul style="list-style-type: none"> <li>• 未输入进给速度指令。</li> <li>• 指令 G95 模式之后的圆筒插补/极坐标插补中，没有 F 指令。</li> </ul> | <ul style="list-style-type: none"> <li>• 由于接通电源时，移动模态指令变为 G01，所以即使程序中没有指定 G01，只要有移动指令，就会按照 G01 进行移动，发出警报。以 F 指令指定进给速度。</li> <li>• 在丝杠导程指令中指定 F。</li> </ul> |
| P 65 | <b>无高速模式 3</b>                                                                                                           | <ul style="list-style-type: none"> <li>• 进行高速模式Ⅲ规格确认。</li> </ul>                                                                                           |

+

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                      | 处 理                                                                                                                                                                  |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| P 70 | <b>圆弧终点偏差大</b> <ul style="list-style-type: none"> <li>圆弧的起点、终点及圆弧中心错误。</li> <li>通过起点的渐开曲线与终点间的差距过大。</li> <li>发出圆弧指令时，构成圆弧平面的 2 轴中，1 轴成为定标有效轴。</li> </ul> | <ul style="list-style-type: none"> <li>确认程序的起点、终点、圆弧中心及半径指定地址的数值。</li> <li>确认地址数值的正、负方向。</li> <li>请检查换算有效轴。</li> </ul>                                               |
| P 71 | <b>无法计算圆中心</b> <ul style="list-style-type: none"> <li>R 指定圆弧插补时，无法计算出圆弧的中心。</li> <li>无法计算出渐开曲线的曲率中心。</li> </ul>                                          | <ul style="list-style-type: none"> <li>确认程序的各地址数值。</li> <li>确认起点或终点是否在渐开线插补基圆的内侧。当进行刀具半径补偿时，确认修正后的起点、终点是否在渐开线插补基圆的内侧。</li> <li>确认起点与终点距离渐开线插补基圆中心的距离是否等距。</li> </ul> |
| P 72 | <b>无螺旋规格</b> <p>没有螺旋规格但是指令了螺旋指令。</p>                                                                                                                     | <ul style="list-style-type: none"> <li>进行螺旋规格确认。</li> <li>在圆弧插补指令中发出 3 轴指令。如果不是螺旋规格，则将直线指令轴移动到下一单节。</li> </ul>                                                       |
| P 90 | <b>无螺纹切削规格</b> <p>没有螺纹切削指令规格，但是指令了螺纹切削指令。</p>                                                                                                            | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                                                                            |
| P 93 | <b>螺纹导程不正确</b> <p>发出螺纹切削指令时，螺纹导程（螺纹间距）不正确。</p>                                                                                                           | <ul style="list-style-type: none"> <li>在螺纹切削指令中正确设定螺纹导程指令。</li> </ul>                                                                                                |
| P100 | <b>无圆筒插补</b> <p>没有圆筒插补规格，但是指令了圆筒插补指令。</p>                                                                                                                | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                                                                            |
| P111 | <b>坐标旋转平面选择</b> <p>在坐标旋转指令（G68）中指令了平面选择指令（G17、G18、G19）。</p>                                                                                              | <ul style="list-style-type: none"> <li>发出 G68 指令后，务必在实施 G69（坐标旋转取消）指令之后，再发出平面选择指令。</li> </ul>                                                                        |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                                                      | 处 理                                                                                                     |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| P112 | <b>R 修正&amp;平面选择</b> <ul style="list-style-type: none"> <li>• 执行刀具半径补偿(G41、G42)及刀尖 R 补偿(G41、G42、G46)指令时, 指令了平面选择指令(G17、G18、G19)。</li> <li>• 刀尖 R 补偿结束时, G40 指令以后没有轴移动指令, 当修正未被取消时, 发出了平面选择指令。</li> </ul> | <ul style="list-style-type: none"> <li>• 完成刀具半径补偿及刀尖 R 补偿指令(G40 的取消指令之后发出轴移动指令)之后, 发出平面选择指令。</li> </ul> |
| P113 | <b>平面选择不正确</b> <p>圆弧指令轴与选择平面不同。</p>                                                                                                                                                                      | <ul style="list-style-type: none"> <li>• 通过正确的平面选择进行圆弧指令。</li> </ul>                                    |
| P122 | <b>无自动角</b> <p>没有自动转角超程(G62)规格, 但是指令了自动转角超程指令。</p>                                                                                                                                                       | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> <li>• 从程序中删除 G62 指令。</li> </ul>                   |
| P126 | <b>高精度中无法执行</b> <p>指令了在高精度控制模式中无法执行的指令。</p> <ul style="list-style-type: none"> <li>• 在高精度控制模式中, 指令了 G 代码组 13 的指令。</li> <li>• 在高精度控制模式中, 指令了铣削/圆筒插补/极坐标插补指令。</li> </ul>                                   | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                               |
| P130 | <b>第 2M 名称不正确</b> <p>程序中所指定的第 2 辅助功能地址与参数中设定的地址不同。</p>                                                                                                                                                   | <ul style="list-style-type: none"> <li>• 确认、修正程序上的第 2 辅助功能的地址。</li> </ul>                               |
| P131 | <b>无速率恒定</b> <p>规格中没有, 但是指令了速率恒定指令(G98)。</p>                                                                                                                                                             | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> <li>• 从速率恒定指令(G96)变更为转速指令(G97)。</li> </ul>        |
| P132 | <b>主轴速度 S=0</b> <p>未输入主轴速度指令。</p>                                                                                                                                                                        | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                               |
| P133 | <b>控制轴编号不正确</b> <p>速率恒定控制轴的指定不正确。</p>                                                                                                                                                                    | <ul style="list-style-type: none"> <li>• 检查速率恒定控制轴的参数程序指定。</li> </ul>                                   |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                       | 处 理                                                                                                       |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| P140 | <b>无位置补偿</b><br>无位置补偿指令（G45~G48）的规格。                                                                                                                      | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                 |
| P141 | <b>旋转中位置补偿</b><br>在图像旋转或坐标旋转指令中指令了位置补偿指令。                                                                                                                 | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                   |
| P142 | <b>无法进行位置补偿圆弧</b><br>指令了无法进行位置补偿的圆弧指令。                                                                                                                    | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                   |
| P150 | <b>无 R 修正规格</b> <ul style="list-style-type: none"> <li>没有刀具半径补偿规格，但是指令了刀具半径补偿*(G41、G42) 指令。</li> <li>没有刀尖 R 补偿规格，但是指令了刀尖 R 补偿(G41、G42、G46) 指令。</li> </ul> | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                 |
| P151 | <b>圆弧模式中修正</b><br>圆弧模式（G02、G03）中，指令了补偿指令（G40、G41、G42、G43、G44、G46）。                                                                                        | <ul style="list-style-type: none"> <li>在补偿指令单节或取消单节中，指令了直线指令（G01）或快速进给指令（G00）。<br/>（将模态转为直线插补）</li> </ul> |
| P152 | <b>无交点</b><br>在执行刀具半径补偿（G41、G42）及刀尖 R 补偿（G41、G42、G46）时，在干扰单节处理中，没有计算出忽略 1 单节时的交点。                                                                         | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                   |
| P153 | <b>修正干扰</b><br>执行刀具半径补偿(G41、G42)及刀尖 R 补偿(G41、G42、G46) 指令时，发生干扰错误。                                                                                         | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                   |
| P155 | <b>修正中固定循环</b><br>直径修正模式中，指令了固定循环指令。                                                                                                                      | <ul style="list-style-type: none"> <li>执行固定循环指令时，由于为直径修正模式，所以执行直径修正取消（G40）指令。</li> </ul>                  |
| P156 | <b>补偿方向未定</b><br>G46 刀尖 R 补偿开始时，补偿方向为未确定的移动矢量。                                                                                                            | <ul style="list-style-type: none"> <li>变更为补偿方向的移动矢量。</li> <li>更换刀刃顶点编号不同的刀具。</li> </ul>                   |

## 附录 2. 程序错误

| 错误编号  | 内 容                                                                              | 处 理                                                                                                                                                                                     |    |    |     |    |       |    |
|-------|----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|-----|----|-------|----|
| P157  | <b>补偿方向翻转</b><br><br>G46 刀尖 R 补偿中，补偿方向翻转。                                        | <ul style="list-style-type: none"><li>• 变更为补偿方向翻转亦可的 G 指令（G00、G28、G30、G33、G53）。</li><li>• 更换为刀刃点编号不同的刀具。</li><li>• 打开#8106 G46 翻转错误回避参数。</li></ul>                                      |    |    |     |    |       |    |
| P158  | <b>不正确的刀具顶点</b><br><br>G46 刀尖 R 补偿中，刀具顶点不正确(1～8 以外)。                             | <ul style="list-style-type: none"><li>• 变更为正确的刀具顶点编号。</li></ul>                                                                                                                         |    |    |     |    |       |    |
| P170  | <b>无补偿编号</b><br><br>发出修正（G41、G42、G43、G46）指令时，没有指定补偿编号（D○○、T○○、H○○）。或补偿编号比规格的组数大。 | <ul style="list-style-type: none"><li>• 在补偿指令单节中，附加补偿编号指令。</li><li>• 确认补偿编号组数，修正为补偿组数以内的补偿编号指令。</li></ul>                                                                               |    |    |     |    |       |    |
| P172  | <b>G10 L 编号错误</b><br><br>G10 指令时，地址指令不正确。                                        | <ul style="list-style-type: none"><li>• 确认 G10 指令的地址 L 的编号，发出正确编号的指令。</li></ul>                                                                                                         |    |    |     |    |       |    |
| P173  | <b>G10 补偿编号错误</b><br><br>G10 指令时，在补偿编号指令中，指定了超过规格组数的补偿编号。                        | <ul style="list-style-type: none"><li>• 在确认补偿组数的基础上，将地址 P 的指令修正为组数以内的指令。</li></ul>                                                                                                      |    |    |     |    |       |    |
| P177  | <b>计算寿命中</b><br><br>在使用数据计数有效信号开启时，执行利用 G10 进行的刀具寿命管理数据的注册。                      | <ul style="list-style-type: none"><li>• 使用数据计数中，无法进行刀具寿命管理数据的注册。使用数据计数有效信号为关闭。</li></ul>                                                                                                |    |    |     |    |       |    |
| P178  | <b>超过寿命注册数量</b><br><br>指令的注册组数、总注册刀具根数或 1 组内的注册根数超过规格范围。                         | <ul style="list-style-type: none"><li>• 检查注册根数。<br/>各最大注册数如下。<table><tr><td>组数</td><td>80</td></tr><tr><td>总根数</td><td>80</td></tr><tr><td>每 1 组</td><td>16</td></tr></table></li></ul> | 组数 | 80 | 总根数 | 80 | 每 1 组 | 16 |
| 组数    | 80                                                                               |                                                                                                                                                                                         |    |    |     |    |       |    |
| 总根数   | 80                                                                               |                                                                                                                                                                                         |    |    |     |    |       |    |
| 每 1 组 | 16                                                                               |                                                                                                                                                                                         |    |    |     |    |       |    |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                                                                      | 处 理                                                                                                    |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| P179 | <b>组编号不正确</b> <ul style="list-style-type: none"> <li>在通过 G10 进行刀具寿命管理数据的注册时，指令了组编号重复的指令。</li> <li>进行 T□□□□99 指令时，指定了未注册的组编号。</li> <li>必须单独进行指令的 M 代码指令与其他 M 代码指令存在于同一单节中。</li> <li>被设定在同一组中的 M 代码指令，存在于同一单节中。</li> </ul> | <ul style="list-style-type: none"> <li>无法重复组编号进行指令。请对各组分别汇总，进行寿命数据的注册。</li> <li>请修正为正确的组编号。</li> </ul> |
| P180 | <b>无钻孔循环</b> <p>没有固定循环（G72~G89）规格，但是指令了固定循环指令。</p>                                                                                                                                                                       | <ul style="list-style-type: none"> <li>进行规格确认。</li> <li>修正程序。</li> </ul>                               |
| P181 | <b>无螺纹切削 S 指令</b> <p>执行钻孔固定循环指令时，未进行主轴转速指令。</p>                                                                                                                                                                          | <ul style="list-style-type: none"> <li>钻孔固定循环 G84、G74（G84、G88）指令时，指令了主轴旋转指令（S）。</li> </ul>             |
| P182 | <b>同步式螺纹切削不正确</b> <p>未能与主轴单元结合。</p>                                                                                                                                                                                      | <ul style="list-style-type: none"> <li>确认与主轴单元的结合。</li> <li>确认有无主轴编码器。</li> </ul>                      |
| P183 | <b>没有间距/螺纹数</b> <p>在钻孔固定循环指令的螺纹切削循环中，没有间距或螺纹圈数指令。</p>                                                                                                                                                                    | <ul style="list-style-type: none"> <li>利用 F 或 E 指令指定间距、螺纹圈数。</li> </ul>                                |
| P184 | <b>间距/螺纹圈数不正确</b> <ul style="list-style-type: none"> <li>在钻孔固定循环指令的螺纹切削循环中，间距或螺纹圈数的指令不正确。</li> <li>相对于主轴的转速，间距过小。</li> <li>相对于主轴的转速，间距过大。</li> </ul>                                                                     | <ul style="list-style-type: none"> <li>确认间距或螺纹圈数。</li> </ul>                                           |
| P190 | <b>无车削循环</b> <p>没有车削循环的规格，但是指令了车削循环指令。</p>                                                                                                                                                                               | <ul style="list-style-type: none"> <li>确认规格。</li> <li>删除车削循环的指令。</li> </ul>                            |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                                                                           | 处 理                                                                                                                                                                                                              |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| P191 | <b>锥形部分长度不正确</b><br><br>发出车削循环指令时，锥形部分长度指令不正确。                                                                                                                                                                                | <ul style="list-style-type: none"> <li>车削循环指令中的 R 指令值比轴的移动量小。</li> </ul>                                                                                                                                         |
| P192 | <b>端面倒角不正确</b><br><br>螺纹切削循环中的端面倒角不正确。                                                                                                                                                                                        | <ul style="list-style-type: none"> <li>设定循环中无法产生的端面倒角量。</li> </ul>                                                                                                                                               |
| P200 | <b>无 MRC 循环规格</b><br><br>没有复合型车削用固定循环 I 的规格，但是指令了复合型车削用固定循环 I 指令（G70～G73）。                                                                                                                                                    | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                                                                                                                        |
| P201 | <b>MRC 程序错误</b> <ul style="list-style-type: none"> <li>通过复合型车削用固定循环 I 调用的子程序中，有如下指令。<br/><br/>               基准点重新启动指令（G27、G28、G29、G30）、螺纹切削（G33、G34）、固定循环、忽略功能（G31）</li> <li>复合型车削用固定循环 I 的加工形状程序的第一个移动单节中，有圆弧指令。</li> </ul> | <ul style="list-style-type: none"> <li>从通过复合型车削用固定循环 I（G70～G73）调用的子程序中，删除以下的 G 代码。<br/><br/>               G27、G28、G29、G30、G31、G33、G34 固定循环的 G 代码</li> <li>从复合型车削用固定循环 I 的加工形状程序的第一个移动单节中，删除 G02/G03。</li> </ul> |
| F202 | <b>超出 MRC 单节</b><br><br>复合型车削用固定循环 I 的形状程序的单节数超过 50 或 200 单节（因机种而异）。                                                                                                                                                          | <ul style="list-style-type: none"> <li>将通过复合型车削用固定循环 I（G70～G73）调用的形状程序的单节数控制在 50 或 200 单节（因机种而异）以下。</li> </ul>                                                                                                   |
| P203 | <b>MRC 形状错误</b><br><br>不是复合型车削用固定循环 I（G70～G73）的形状程序能够正确切削的形状。                                                                                                                                                                 | <ul style="list-style-type: none"> <li>检查复合型车削用固定循环 I（G70～G73）的形状程序。</li> </ul>                                                                                                                                  |
| P204 | <b>MRC 循环指令错误</b><br><br>复合型车削用固定循环（G70～G78）的指令值不正确。                                                                                                                                                                          | <ul style="list-style-type: none"> <li>检查复合型车削用固定循环（G70～G78）的指令值。</li> </ul>                                                                                                                                     |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                            | 处 理                                                                                                              |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| P210 | <b>无纹样循环</b> <ul style="list-style-type: none"> <li>没有复合型车削用固定循环 II（G74~G76）的规格，但是指令了该指令。</li> <li>在 1 系统系统中指令了 2 系统同时螺纹切削指令。</li> </ul>                       | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                        |
| P220 | <b>无固定循环</b> <p>没有特别固定循环。</p>                                                                                                                                  | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                        |
| P221 | <b>特别固定孔数为 0</b> <p>在特别固定循环中，孔个数的指定为 0。</p>                                                                                                                    | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                          |
| P222 | <b>G36 角度间隔错误</b> <p>G36 中，角度间隔为 0。</p>                                                                                                                        | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                          |
| P223 | <b>圆切削半径错误</b> <p>G12、G13 中的半径值小于修正量。</p>                                                                                                                      | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                          |
| P224 | <b>无圆切削</b> <p>没有圆切削规格。</p>                                                                                                                                    | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                        |
| P230 | <b>子程序嵌套超出</b> <ul style="list-style-type: none"> <li>在子程序中调用子程序的次数超过 8 次。</li> <li>IC 卡内的程序中，有 M198 指令。</li> <li>嵌套调用 IC 卡内的程序。（IC 卡程序在嵌套中只能调用一次。）</li> </ul> | <ul style="list-style-type: none"> <li>确认子程序的调用次数，修正程序，确保不超过 8 次。</li> <li>确认 IC 卡内的程序与想调用的 IC 卡程序编号。</li> </ul> |
| P231 | <b>无顺序编号</b> <p>调用子程序时，未设定通过子程序重新启动时或通过 GOTO 指定的顺序编号。</p>                                                                                                      | <ul style="list-style-type: none"> <li>在适当的单节中指定顺序编号。</li> <li>当使用 IC 卡时，确认 IC 卡内的程序及其编号。</li> </ul>             |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                     | 处 理                                                                              |
|------|-------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| P232 | <b>无程序编号</b> <ul style="list-style-type: none"> <li>所指定的子程序未注册到内存及 IC 卡中。</li> <li>IC 卡中所注册的程序的文件名与 0 编号不一致。</li> </ul> | <ul style="list-style-type: none"> <li>注册子程序。</li> <li>确认 IC 卡内的程序编号。</li> </ul> |
| P241 | <b>无变量编号</b> <p>所指定的变量编号大于规格的变量编号。</p>                                                                                  | <ul style="list-style-type: none"> <li>进行规格确认。</li> <li>进行程序变量编号的确认。</li> </ul>  |
| P242 | <b>变量定义=无</b> <p>定义变量时，未使用“=”指令。</p>                                                                                    | <ul style="list-style-type: none"> <li>在程序的变量定义中设定“=”。</li> </ul>                |
| P243 | <b>变量使用不正确</b> <p>有公式的左边或右边无法使用的变量。</p>                                                                                 | <ul style="list-style-type: none"> <li>修正程序。</li> </ul>                          |
| P252 | <b>图像&amp;坐标旋转</b> <p>在图像旋转中，指令了坐标旋转相关指令（G68、G89）。</p>                                                                  | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                          |
| P260 | <b>无坐标旋转</b> <p>没有坐标旋转规格，但是指令了坐标旋转指令。</p>                                                                               | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                        |
| P270 | <b>无宏规格</b> <p>没有宏规格但是指令了宏指令。</p>                                                                                       | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                        |
| P271 | <b>没有宏插入</b> <p>没有宏插入规格，但是指令了宏插入指令。</p>                                                                                 | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                        |
| P272 | <b>宏语句同时</b> <p>同一单节中，执行语句与宏语句并存。</p>                                                                                   | <ul style="list-style-type: none"> <li>检查程序，将执行语句与宏语句分为不同的单节。</li> </ul>         |

## 附录 2. 程序错误

| 错误编号 | 内 容                                               | 处 理                                                                                                     |
|------|---------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| P273 | <b>宏嵌套过多</b><br>宏调用的嵌套超过规格的次数。                    | <ul style="list-style-type: none"> <li>检查程序，修正为宏调用不超过规格的次数。</li> </ul>                                  |
| P275 | <b>宏自变量组超出</b><br>宏调用自变量类型 II 中，自变量的组数过多。         | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                 |
| P276 | <b>调用取消单独</b><br>不处于 G66 指令模态中，但是指令了 G67 指令。      | <ul style="list-style-type: none"> <li>检查程序。</li> <li>G67 指令是调用取消指令，所以在执行 G67 指令前应执行 G66 指令。</li> </ul> |
| P277 | <b>宏警报信息</b><br>通过#3000 执行了警报指令。                  | <ul style="list-style-type: none"> <li>请参阅诊断画面的操作信息。</li> <li>请参阅机械制造商发行的使用说明书。</li> </ul>              |
| P280 | <b>[, ] 嵌套超出</b><br>1 单节中的 “[” 或 “]” 的数量超过 5 层以上。 | <ul style="list-style-type: none"> <li>检查程序，修正程序以确保 “[” 或 “]” 的数量在 5 层以内。</li> </ul>                    |
| P281 | <b>[, ] 数不同</b><br>1 单节中所指定的 “[” 与 “]” 数量不同。      | <ul style="list-style-type: none"> <li>检查程序，修正程序以确保 “[” 或 “]” 成对出现。</li> </ul>                          |
| P282 | <b>无法运算</b><br>公式不正确。                             | <ul style="list-style-type: none"> <li>检查程序，纠正公式。</li> </ul>                                            |
| P283 | <b>分母为 0</b><br>除法运算的分母为 0。                       | <ul style="list-style-type: none"> <li>检查程序，进行修正，确保公式的除法运算的分母不为 0。</li> </ul>                           |
| P290 | <b>IF 语句错误</b><br>IF[<条件式>]GOTO□语句错误。             | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                 |
| P291 | <b>WHILE 语句错误</b><br>WHILE[<条件式>]DO□～END□语句错误。    | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                 |
| P292 | <b>SETVN 语句错误</b><br>变量名设定、SETVN□语句错误。            | <ul style="list-style-type: none"> <li>检查程序。</li> <li>将 SETVN 语句的变量名的字符数控制在 7 个字符以下。</li> </ul>         |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                            | 处 理                                                                                         |
|------|--------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| P293 | <b>DO-END 嵌套过多</b><br><br>WHILE[<条件式>]DO□～END□语句的 DO 与 END □的□数量（嵌套次数）超过 27 次。 | <ul style="list-style-type: none"> <li>检查并修正程序，确保 DO～END 语句的嵌套次数不超过 27 次。</li> </ul>        |
| P294 | <b>DO-END 不成对</b><br><br>DO 与 END 不成对。                                         | <ul style="list-style-type: none"> <li>检查程序，并修正为 DO～END 成对出现。</li> </ul>                    |
| P295 | <b>纸带 WHILE/GOTO</b><br><br>在纸带运转中，纸带上有 WHILE 或 GOTO 语句。                       | <ul style="list-style-type: none"> <li>纸带运转中，无法执行含有 WHILE 或 GOTO 语句的程序，所以转为内存运行。</li> </ul> |
| P296 | <b>宏地址不足</b><br><br>用户宏中需要地址，但是未指定。                                            | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                     |
| P297 | <b>A 不是变量</b><br><br>用户宏中，地址 A 未指定为变量。                                         | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                     |
| P298 | <b>G200-G202 纸带</b><br><br>纸带运转、MDI 运转中，指定了用户宏的 G200～G202。                     | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                     |
| P300 | <b>变量名不正确</b><br><br>未正确指定变量名。                                                 | <ul style="list-style-type: none"> <li>检查程序的变量名，修正为正确的变量名。</li> </ul>                       |
| P301 | <b>变量名重复</b><br><br>变量名重复。                                                     | <ul style="list-style-type: none"> <li>修正程序确保变量名不重复。</li> </ul>                             |
| P360 | <b>无程序镜像</b><br><br>没有程序镜像规格，但是执行了镜像（G50.1、G51.1）指令。                           | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                   |
| P370 | <b>无对向镜像</b><br><br>无对向刀具台镜像规格。                                                | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                   |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                            | 处 理                                                                                           |
|------|--------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| P371 | <b>对向镜像不正确</b><br><br>向外部镜像、参数镜像中的轴执行了对向刀具台镜像指令。<br><br>对旋转轴执行了镜像生效的对向刀具台镜像指令。 | <ul style="list-style-type: none"> <li>• 检查程序。</li> <li>• 检查参数。</li> </ul>                    |
| P380 | <b>无圆角/C 规格</b><br><br>没有端面倒角/圆角的规格,但是执行了该指令。                                  | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> <li>• 从程序中删除端面倒角/圆角。</li> </ul>         |
| P381 | <b>无圆弧规格</b><br><br>没有端面倒角/圆角 II 的规格,但是在圆弧插补单节中执行了端面倒角/圆角指令。                   | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                                   |
| P382 | <b>角后无移动</b><br><br>端面倒角/圆角的下一单节不是移动指令。                                        | <ul style="list-style-type: none"> <li>• 将端面倒角/圆角指令的下一单节改为 G01 指令。</li> </ul>                 |
| P383 | <b>角移动过短</b><br><br>在端面倒角/圆角指令中,移动距离比端面倒角/圆角的指令更短。                             | <ul style="list-style-type: none"> <li>• 由于移动距离比端面倒角/圆角更短,所以将端面倒角/圆角缩小到小于移动距离。</li> </ul>     |
| P384 | <b>角后移动过短</b><br><br>在端面倒角/圆角指令中,下一单节的移动距离小于端面倒角/圆角。                           | <ul style="list-style-type: none"> <li>• 由于下一单节的移动距离小于端面倒角/圆角,所以将端面倒角/圆角缩小到小于移动距离。</li> </ul> |
| P385 | <b>G0 G33 中角</b><br><br>端面倒角/圆角的单节处于 G00 或 G33 的模式中。                           | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                     |
| P390 | <b>无几何学</b><br><br>没有几何学规格,但是执行了几何学指令。                                         | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                                   |
| P391 | <b>无几何圆弧规格</b><br><br>没有几何 IB 的规格。                                             | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                                   |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                 | 处 理                   |
|------|-----------------------------------------------------------------------------------------------------|-----------------------|
| P392 | <b>无几何直线角差</b><br>几何直线一直线的角度差在 1 度以内。                                                               | • 修正几何角度。             |
| P393 | <b>几何后增量值</b><br>以增量值指令几何学的第 2 单节。                                                                  | • 以绝对值指令几何的第 2 单节。    |
| P394 | <b>无几何后直线</b><br>几何后第 2 单节中没有直线指令。                                                                  | • 执行 G01 指令。          |
| P395 | <b>几何地址不足</b><br>几何格式错误。                                                                            | • 检查程序。               |
| P396 | <b>几何中平面切换</b><br>执行几何指令过程中，指令了平面切换指令。                                                              | • 在几何之前进行平面切换。        |
| P397 | <b>几何圆弧终点偏差</b><br>在几何 IB 中，圆弧终点没有与下一单节的起点连接。或相交。                                                   | • 检查包括几何的圆弧指令在内的前后指令。 |
| P398 | <b>无几何加工 IB</b><br>没有几何加工 IB 规格，但是执行了几何加工指令。                                                        | • 进行规格确认。             |
| P421 | <b>参数输入错误</b><br>• 所指令的参数编号、设定数据不正确。<br>• 在参数输入模式中，指定了不正确的 G 指令地址。<br>• 固定循环模态中或顶点 R 修正中，执行了参数输入指令。 | • 检查程序。               |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                     | 处 理                                                                                                                                               |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| P430 | <b>存在重新启动未完成轴</b> <ul style="list-style-type: none"> <li>对未执行基准点重新启动的轴执行了基准点重新启动以外的移动指令。</li> <li>向轴拆卸轴发出指令。</li> </ul>                                 | <ul style="list-style-type: none"> <li>手动执行基准点重新启动。</li> <li>由于向轴拆卸有效的轴指令了指令，所以轴拆卸失效。</li> </ul>                                                  |
| P431 | <b>无 2、3、4 重新启动</b> <p>没有第 2、第 3、第 4 基准点重新启动规格，但是指令了该指令。</p>                                                                                            | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                                                         |
| P434 | <b>存在校准错误轴</b> <p>执行原点校准指令（G27）时，存在未返回原点位置的轴。</p>                                                                                                       | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                                                           |
| P435 | <b>G27-M 组合错误</b> <p>在 G27 的指令单节中，同时指令了 M 单独指令。</p>                                                                                                     | <ul style="list-style-type: none"> <li>在 G27 指令单节中，无法进行 M 单独指令，所以把 G27 指令与 M 单独指令分到不同的单节中。</li> </ul>                                             |
| P436 | <b>G29-M 组合错误</b> <p>在 G29 的指令单节中，同时指令了 M 单独指令。</p>                                                                                                     | <ul style="list-style-type: none"> <li>在 G29 指令单节中，无法进行 M 单独指令，所以把 G29 指令与 M 单独指令分到不同的单节中。</li> </ul>                                             |
| P438 | <b>在 G54.1 中无法执行 G52</b> <p>在 G54.1 指令中，执行了局部坐标系指令。</p>                                                                                                 | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                                                           |
| P450 | <b>无夹头隔壁</b> <p>没有夹头隔壁，但是执行了夹头隔壁有效指令（G22）。</p>                                                                                                          | <ul style="list-style-type: none"> <li>进行规格确认。</li> </ul>                                                                                         |
| P460 | <b>纸带输入输出错误</b> <ul style="list-style-type: none"> <li>读带机发生错误。或打印宏时，打印机发生错误。</li> <li>以纸带模式（纸带运转、IC 卡运转）进行继续搜索之后，进行重新启动，然后不进行搜索操作就试图进行继续搜索。</li> </ul> | <ul style="list-style-type: none"> <li>进行连接装置、电缆的确认。</li> <li>确认输入输出装置的参数。</li> <li>请插入 IC 卡。</li> <li>重新启动后，请务必进行搜索，进行程序排序之后，再进行继续搜索。</li> </ul> |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                | 处 理                                                                                                                         |
|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| P461 | <b>文件输入输出错误</b><br>无法读入加工程序的文件。                                                                                                                                    | <ul style="list-style-type: none"> <li>• 内存运转时，可能是内存中储存的程序损坏。完全输出程序及刀具数据等之后，进行格式化。</li> <li>• 确认储存文件的外部设备是否正确安装。</li> </ul> |
| P462 | <b>电脑连接错误</b><br>BTR 运转中，发生通信错误。                                                                                                                                   | <ul style="list-style-type: none"> <li>• 由于同时显示 L01 电脑连接错误，所以根据错误编号进行处理。</li> </ul>                                         |
| P480 | <b>无铣削规格</b> <ul style="list-style-type: none"> <li>• 没有铣削功能规格，但是指令了铣削指令。</li> <li>• 没有极坐标插补规格，但是指令了极坐标插补指令。</li> </ul>                                            | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                                                                 |
| P481 | <b>不正确 G 代码（铣削）</b> <ul style="list-style-type: none"> <li>• 在铣削模式中使用了不正确的 G 代码。</li> <li>• 在圆筒插补/极坐标插补模式中使用了错误的 G 代码。</li> <li>• 在刀具半径补偿中执行了 G07.1 指令。</li> </ul> | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                                                                   |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                                                                                                                                                                               | 处 理                                                                      |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| P482 | <b>不正确轴指令（铣削）</b> <ul style="list-style-type: none"> <li>在铣削模式中，执行了旋转轴指令。</li> <li>在铣削轴编号上设定了不正确值的情况下，执行了铣削。</li> <li>在镜像中执行了圆筒插补/极坐标插补指令。</li> <li>在 T 指令后的刀具半径补偿动作未完成的状态下，执行了圆筒插补/极坐标插补指令。</li> <li>在无法进行圆筒插补的状态（不包含旋转轴/外部镜像开）下，执行了 G07.1 指令。</li> <li>在无法进行极坐标插补的状态下，执行了 G12.1 指令。</li> <li>在圆筒插补中，执行了圆筒坐标系轴以外的轴指令。</li> </ul> | <ul style="list-style-type: none"> <li>请确认加工程序、参数、PLC I/F 信号。</li> </ul> |
| P484 | <b>重新启动未完成轴指令（铣削）</b> <ul style="list-style-type: none"> <li>在铣削模式中，对基准点重新启动未完成的轴执行了轴移动指令。</li> <li>在圆筒插补/极坐标插补中，对基准点重新启动未完成的轴执行了轴移动指令。</li> </ul>                                                                                                                                                                                | <ul style="list-style-type: none"> <li>请手动进行基准点重新启动。</li> </ul>          |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                                                                                                                                                                                                                                                                                                                                                            | 处 理                                                                                                                                                                                                                                                       |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| P485 | <p><b>不正确模态（铣削）</b></p> <ul style="list-style-type: none"> <li>在顶点 R 修正中或周速恒定控制中，铣削模式启动。</li> <li>在铣削模式中执行了 T 指令。</li> <li>在刀具半径补偿中，从铣削模式切换到切削模式。</li> <li>在周速恒定控制模式中（G96），指令了圆筒插补/极坐标插补指令。</li> <li>在圆筒插补模式下，执行了未许可的指令。</li> <li>在圆筒插补/极坐标插补模式中，执行了 T 指令。</li> <li>在执行 G07.1 指令之前或之后，在未设定平面选择指令的状态下，执行了移动指令。</li> <li>在极坐标插补模式中，执行了平面选择指令。</li> <li>在刀具半径补偿中执行了圆筒插补/极坐标插补指令。</li> <li>指定了圆筒半径为 0 的 G16 平面。</li> <li>在程序坐标旋转中（G68），执行了圆筒插补/极坐标插补指令。</li> </ul> | <ul style="list-style-type: none"> <li>检查程序。</li> <li>在执行 G12.1 指令前，请执行 G40（顶圆角修正模式取消）或 G97（周速恒定取消）指令。</li> <li>在执行 G12.1 指令前，请执行 T 指令。</li> <li>在执行 G13.1 指令前，请执行 G40（刀具半径补偿取消）指令。</li> <li>请指定 0 以外的圆筒半径值。或请在 G12.1/G16 指令前，将 X 轴指定为 0 以外的值。</li> </ul> |
| P486 | <p><b>不可铣削状态</b></p> <ul style="list-style-type: none"> <li>在镜像中（参数/外部输入 ON 时），执行了铣削指令。</li> <li>在对向刀具台镜像中，执行了极坐标插补、圆筒插补、铣削插补指令。</li> <li>在法线控制中，执行了极坐标插补、圆筒插补的开始指令。</li> </ul>                                                                                                                                                                                                                                                                                | <ul style="list-style-type: none"> <li>检查程序。</li> </ul>                                                                                                                                                                                                   |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                                                          | 处 理                                                                        |
|------|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| P511 | <b>等待代码错误</b> <ul style="list-style-type: none"> <li>• 在同一单节中指定了 2 个以上的等待代码。</li> <li>• 在同一单节中指定了等待 M 代码与 “!” 代码。</li> </ul> | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                  |
| P600 | <b>无自动 TLM</b> <p>没有自动刀具长度测定规格，但是指令了自动刀具长度测定指令（G37）。</p>                                                                     | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                |
| P601 | <b>无忽略规格</b> <p>没有忽略规格，但是指令了忽略指令（G31）。</p>                                                                                   | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                |
| P602 | <b>无多级忽略</b> <p>没有多级忽略指令规格，但是执行了多级忽略指令（G31.1、G31.2、G31.3）。</p>                                                               | <ul style="list-style-type: none"> <li>• 进行规格确认。</li> </ul>                |
| P603 | <b>忽略速度 0</b> <p>忽略速度为 0。</p>                                                                                                | <ul style="list-style-type: none"> <li>• 指定略过速度。</li> </ul>                |
| P604 | <b>TLM 不正确轴指令</b> <p>在自动刀具长度测定单节中，未指定轴。或指定了 2 轴以上。</p>                                                                       | <ul style="list-style-type: none"> <li>• 发出仅包含轴的指令。</li> </ul>             |
| P605 | <b>TLM T 相同单节</b> <p>T 代码与自动刀具长度测定指令位于同一单节中。</p>                                                                             | <ul style="list-style-type: none"> <li>• 在自动刀具长度测定指令单节前执行 T 指令。</li> </ul> |
| P606 | <b>TLM 以前未指定 T</b> <p>在自动刀具长度测量指令中，未指定 T 代码。</p>                                                                             | <ul style="list-style-type: none"> <li>• 在自动刀具长度测定指令单节前执行 T 指令。</li> </ul> |
| P607 | <b>TLM 信号不正确 ON</b> <p>在到达 D 指令或参数的减速区域 d 所指定的区域前，测定位置到达信号 ON。或信号一直没有 ON。</p>                                                | <ul style="list-style-type: none"> <li>• 检查程序。</li> </ul>                  |

## 附录 2. 程序错误

| 错误编号 | 内 容                                                                                               | 处 理                                |
|------|---------------------------------------------------------------------------------------------------|------------------------------------|
| P608 | <b>直径修正中略过</b><br>在直径修正中，执行了略过指令。                                                                 | • 执行直径修正取消（G40）指令，或删除略过指令。         |
| P610 | <b>参数不正确</b><br>• 参数设定不正确。                                                                        | • 检查程序。<br>• 检查参数。                 |
| P612 | <b>无法进行指数函数插补</b><br>在对向刀具台镜像中，指令了指数函数插补中的轴移动指令。                                                  | • 检查程序。                            |
| P700 | <b>指令值不正确</b><br>对未进行串行连接的主轴指令了主轴同步指令。                                                            | • 检查程序。<br>• 检查参数。                 |
| P900 | <b>无法线控制</b><br>没有法线控制规格，但是执行了法线控制指令（G40.1、G41.1、G42.1）。                                          | • 确认规格。                            |
| P901 | <b>法线轴 G92</b><br>法线控制中，向法线控制轴指令了坐标系预设指令（G92）。                                                    | • 检查程序。                            |
| P902 | <b>法线轴不正确</b><br>• 将法线控制轴设定为了直线轴。<br>• 将法线控制轴设定为了直线型旋转轴 II 轴。<br>• 法线控制轴未设定。<br>• 法线控制轴与平面选择的轴重合。 | • 修正法线控制轴。                         |
| P903 | <b>法线中平面选择</b><br>在法线控制中，指令了平面选择指令（G17、G18、G19）。                                                  | • 从法线控制中的程序中删除平面选择指令（G17、G18、G19）。 |
| P990 | <b>无法进行预计算</b><br>根据需要预读的命令（顶圆角修正、端面倒角/圆角、几何加工 I、几何加工 IB、复合型车削用固定循环）的组合，预读单节数达到 8 个以上。            | • 减少或删除需要预读的命令组合。                  |

## 修 订 记 录

| 修订日     | 说明书编号               | 修 订 内 容 |
|---------|---------------------|---------|
| 2006年4月 | IB-1500183 (CHN) -A | 初 版 完 成 |
|         |                     |         |

#### 提示

虽然本说明书的记述内容努力追踪软件、硬件的修订，但是仍然可能会无法与修订保持同步。  
在使用时，如果有不清楚的地方，请向我公司营业所等咨询。

三菱电机株式会社名古屋制作所     NC 系统部  
〒461-8670 名古屋市东区矢田南五丁目 1 番 14 号   TEL (052)7212111 (代表)

#### 禁止擅自转载

未经过本公司许可，禁止以任何形式转载或复制本说明书的部分或全部内容。

©2002-2005 MITSUBISHI ELECTRIC CORPORATION  
ALL RIGHTS RESERVED