

西门子828D 车削编程技术

XIMENZI 828D CHEXIAO BIANCHENG JISHU

孙德茂 编著



机械工业出版社
CHINA MACHINE PRESS

西门子 828D 车削编程技术

孙德茂 编著



机械工业出版社

西门子公司在中国市场继推出 802S、802D 之后,又推出了 828D 数控系统。该系统内容较丰富,接近 840D。本书以介绍其功能为主线,着重阐明对其功能的理解和应用,主要内容包括车削加工的基本编程指令、扩展的编程指令和特殊的编程指令,并对编程工艺功能(循环)也作了翔实的介绍。

本书可供使用西门子数控系统进行车削加工的程序员、操作者及相关人员使用,也可供大专院校数控及相关专业师生使用。

图书在版编目(CIP)数据

西门子 828D 车削编程技术/孙德茂编著. —北京:机械工业出版社, 2012. 11

ISBN 978 - 7 - 111 - 40051 - 6

I. ①西… II. ①孙… III. ①数控机床—车床—车削—程序设计
IV. ①TG519. 1

中国版本图书馆 CIP 数据核字(2012)第 243616 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:张秀恩 责任编辑:张秀恩 杨明远

版式设计:霍永明 责任印制:杨 曦

北京圣夫亚美印刷有限公司印刷

2013 年 1 月第 1 版第 1 次印刷

184mm × 260mm · 20.25 印张 · 498 千字

0001—3000 册

标准书号:ISBN 978 - 7 - 111 - 40051 - 6

定价:56.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心 : (010)88361066 教材网:<http://www.cmpedu.com>

销售一部 : (010)68326294 机工官网:<http://www.cmpbook.com>

销售二部 : (010)88379649 机工官博:<http://weibo.com/cmp1952>

读者购书热线:(010)88379203 封面无防伪标均为盗版

前 言

西门子公司在中国继推出 802S、802D 之后，又推出了 828D 数控系统。该系统内容较丰富，接近 840D。西门子公司将编程功能分为基本功能、扩展功能和特殊功能，并分在两本手册中加以介绍，一本是基础部分，另一本是工作准备部分，对一个具体的机床和所配置的系统具有的功能便不再区分，因此，本书将其分为 4 章，对车削编程功能加以介绍。

第 1 章编程基础，介绍了几何原理、编程规则、程序创建、文件和程序管理。第 2 章 NC 代码编程指令，介绍了换刀、刀具补偿、主轴运动、进给控制、几何设置、自动返回参考点、编程的工作区域极限和保护区、位移指令、特殊的位移指令、轨迹运行特性、刀具半径补偿、坐标变换（框架）、运动变换、轴耦合—联动、运动同步动作、其他功能、辅助功能输出和 PLC 变量的读写。第 3 章灵活的 NC 编程，介绍了变量、间接编程、运算功能、控制功能、中断程序（ASUP）、轴交换和转移、子程序和宏指令技术。第 4 章编程工艺功能（循环），以 ShopTurn 程序为例来介绍，包括车削循环、车削轮廓循环、钻孔循环、其他循环和功能以及其他 ShopTurn 循环和功能。

本书没有按惯例编写编程实例应用，本人认为零件程序与具体的机床结构和组成、配置的系统功能、零件的结构、毛坯的状态、工艺的要求、用户的条件、操作者和编程员的习惯都密切相关，因此没有通用性，所以只是在具体功能介绍时加强了举例，以加深对功能的理解，相信读者会写出实用的零件程序。

西门子系统中有不少英文（缩写），这对读者的英语提出了要求。相信读者只要努力，会逐步克服语言障碍，掌握系统的功能。

书中介绍的功能并不是 828D 的全部功能，没有介绍的功能请阅读系统的说明书。

书中的疏漏和差错之处敬请指正。

作 者

目 录

前言

第 1 章 编程基础	1
1.1 概述	1
1.2 几何原理	1
1.2.1 工件位置	1
1.2.2 工件平面	2
1.2.3 零点和特征点	2
1.2.4 坐标系	3
1.2.5 进给轴	4
1.3 编程规则	7
1.3.1 程序名	7
1.3.2 程序分量	7
1.3.3 程序段规则	8
1.3.4 赋值规则	9
1.3.5 注释	9
1.3.6 信息显示 (MSG)	9
1.3.7 程序段跳过	10
1.3.8 程序结束	10
1.4 程序创建	10
1.4.1 基本步骤	10
1.4.2 可用字符	11
1.4.3 程序头	11
1.4.4 程序例	12
1.5 文件和程序管理	14
1.5.1 程序存储器	14
1.5.2 工作存储器 (CHANDATA, COMPLETE, INITIAL)	16
1.5.3 步进编辑器中的结构化指令 (SEFORM)	18
第 2 章 NC 代码编程指令	19
2.1 换刀	19
2.1.1 无刀具管理情况下的换刀	19
2.1.2 使用刀具管理选件的换刀	19
2.2 刀具补偿	20
2.2.1 刀具补偿的常用信息	20
2.2.2 刀具长度补偿	21
2.2.3 刀具半径补偿	21
2.2.4 刀具补偿存储器	21
2.2.5 刀具类型	23
2.2.6 刀具补偿调用	25
2.2.7 修改刀具补偿数据	26
2.2.8 可编程的刀具补偿偏移 (TOFFL, TOFF, TOFFR)	26
2.3 主轴运动	29
2.3.1 主轴转速 (S), 主轴旋转方向 (M3, M4, M5)	29
2.3.2 切削速度 (SVC)	30
2.3.3 恒定切削速度 (G96/G961/G962, G97/G971/972, G973, LIMS, SCC)	31
2.3.4 可编程的主轴转速极限 (G25, G26)	34
2.4 进给控制	35
2.4.1 进给率 (G93, G94, G95, F, FGROUP, FL, FGROUP)	35
2.4.2 运行定位轴 (POS, POSA, FA, WAITP)	39
2.4.3 位置控制的主轴运动 (SPCON, SPCOF)	40
2.4.4 定位主轴 (SPOS, SPOSA, M19, M70, WAIT)	41
2.4.5 用于定位轴/主轴的进给率 (FA, FPR, FPRAON, FPRAOF)	43
2.4.6 可编程进给补偿 (OVR, OVRRAP, OVRA)	45
2.4.7 进给率: 带手轮倍率 (FD, FDA)	46
2.4.8 曲线轨迹部分的进给率优化 (CFTCP, CFC, CFIN)	48
2.4.9 非模态进给 (FB)	48
2.4.10 每齿进给量 (G95 FZ)	49
2.5 几何设置	51
2.5.1 可设定的零点偏移 (G54...G57, G505...G599, G53, G500,	

SUPA, G153)	51	2.10 轨迹运行特性	97
2.5.2 工作平面选择 (G17/G18/G19) ...	52	2.10.1 准停 (G60, G9, G601, G602, G603)	97
2.5.3 尺寸指令	53	2.10.2 可编程的运动结束条件 (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)	99
2.6 自动返回参考点	62	2.10.3 带有拐角减速的进给减速 (FENDNORM, G62, G621)	101
2.6.1 返回参考点 (G74)	62	2.10.4 连续路径运行 (G64)	101
2.6.2 返回固定点 (G75, G751)	62	2.10.5 带前馈控制运行 (FFWON, FFWOF)	102
2.6.3 运行到固定挡块 (FXS, FXST, FXSW)	64	2.10.6 轮廓精确度 (CPRECON, CPRECOF)	103
2.7 编程的工作区域极限和保护区	66	2.10.7 暂停时间 (G4)	103
2.7.1 BCS 中的工作区域限制 (G25/G26, WALIMON, WALIMOF)	66	2.10.8 内部预处理程序停止	104
2.7.2 在 WCS/ENS 中的工作区域限制 (WALCS0 ~ WALCS10)	68	2.10.9 加速性能的加速模式 (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)	104
2.7.3 保护区的确定 (CPROTDEF, NPROTDEF)	69	2.10.10 对运动控制的影响 (VELOLIM, JERKLIM)	105
2.7.4 激活/取消激活保护区 (CPROT, NPROT)	71	2.10.11 跟随轴时的加速影响 (VELOLIMA, ACCLIMA, JERKLIMA)	106
2.8 位移指令	73	2.10.12 可编程的加速度修调 (ACC) (选项)	107
2.8.1 快速运行 (G0, RTLION, RTLIOF)	74	2.10.13 激活工艺专用动态值 (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) ...	108
2.8.2 线性插补 (G1)	75	2.10.14 进给速度曲线 (FNORM, FLIN, FCUB)	109
2.8.3 圆弧插补	76	2.10.15 带有缓存的程序运行过程 (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)	109
2.8.4 轮廓编程	81	2.10.16 可以有条件中断的程序段 (DELAYFSTON, DELAYFSTOF)	111
2.8.5 恒定螺距的螺纹切削 (G33)	84	2.10.17 阻止 SERUPRO 的程序位置 (IPTRLOCK, IPTRUNLOCK)	113
2.8.6 递增螺距与递减螺距的螺纹切削 (G34, G35)	86	2.10.18 返回轮廓 (REPOSA, REPOS, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMI, RMB, RME, RMN)	115
2.8.7 不带补偿夹具的攻螺纹 (G331, G332)	87	2.11 刀具补偿	117
2.8.8 带补偿夹具的攻螺纹 (G63)	90	2.11.1 刀具半径补偿 (G40, G41, G42,	
2.8.9 螺纹切削时快速回程 (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN)	90		
2.8.10 倒角, 倒圆 (CHF, CHR, RND, RNDM, FRC, FRCM)	92		
2.9 特殊的位移指令	95		
2.9.1 逼近已经过编码处理的位置 (CAC, CIC, CDC, CACP, CACN)	95		
2.9.2 用接触式探头测量 (MEAS, MEAW)	96		
2.9.3 适用于 OEM 用户的专用函数 (OEMIP01, OEMIP02, G810 ~ G829)	97		

OFFN)	117	2. 15. 6 删除同步动作 (CANCEL)	184
2. 11. 2 轮廓返回和离开 (NORM, KONT)	119	2. 15. 7 特定运行状态下的控制属性	184
2. 11. 3 外角的补偿 (G450, G451, DISC)	120	2. 16 其他功能	186
2. 11. 4 平滑逼近和回退	121	2. 16. 1 轴功能 (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)	186
2. 11. 5 碰撞监控 (CDON, CDOF)	128	2. 16. 2 可转换的几何轴 (GEOAX)	187
2. 11. 6 刀具带相应的刀沿时的 补偿转换	129	2. 16. 3 交互式调用零件程序 (MMC) 窗口	189
2. 12 坐标转换 (框架)	129	2. 16. 4 程序执行时间/工件计数器	190
2. 12. 1 框架	129	2. 16. 5 报警 (SETAL)	193
2. 12. 2 框架指令	130	2. 17 辅助功能输出	193
2. 12. 3 可编程的零点偏移	131	2. 17. 1 M 功能	195
2. 12. 4 可编程旋转 (ROT, AROT, RPL)	132	2. 17. 2 H 功能	197
2. 12. 5 编程的框架旋转, 带立体角 (ROTS, AROTS, CROTS)	134	2. 18 PLC 变量的读和写	197
2. 12. 6 可编程的比例系数 (SCALE, ASCALE)	135	第 3 章 灵活的 NC 编程	198
2. 12. 7 可编程的镜像 (MIRROR, AMIRROR)	135	3. 1 变量	198
2. 12. 8 取消框架 (G53, G153, SUPA, G500)	137	3. 1. 1 变量的类型	198
2. 12. 9 取消叠加运行 (DRFOF, CORROF)	137	3. 1. 2 系统变量	198
2. 12. 10 通过框架变量转换坐标	139	3. 1. 3 预定义用户变量	199
2. 12. 11 给框架/框架变量赋值	140	3. 1. 4 用户变量	201
2. 12. 12 精偏移和粗偏移 (CFINE, CTRANS)	143	3. 1. 5 系统变量、用户变量和 NC 语言 指令的重新定义 (REDEF)	204
2. 12. 13 外部零点偏移	143	3. 2 间接编程	204
2. 12. 14 预设定位移 (PRESETON)	144	3. 2. 1 间接编程 G 代码	205
2. 12. 15 NCU 全局框架	144	3. 2. 2 间接编程位置属性 (GP)	206
2. 13 运动变换	147	3. 2. 3 间接编程零件程序行 (EXECSTRING)	207
2. 13. 1 端面铣削 (TRANSMIT)	148	3. 3 运算功能	207
2. 13. 2 柱面铣削 (TRACYL)	149	3. 3. 1 比较运算和逻辑运算	209
2. 14 轴耦合—联动 (TRAILON, TRAILOF)	152	3. 3. 2 比较错误的精度修正 (TRUNC)	210
2. 15 运动同步动作	153	3. 3. 3 运算的优先级	211
2. 15. 1 基础部分	153	3. 4 控制功能	211
2. 15. 2 条件和动作的运算符	157	3. 4. 1 程序跳转和分支	211
2. 15. 3 同步动作的主运行变量	157	3. 4. 2 程序部分重复 (REPEAT, REPEATB, ENDLABEL, P)	215
2. 15. 4 同步进行的动作	164	3. 4. 3 程序循环	219
2. 15. 5 工艺循环	179	3. 5 中断程序 (ASUP)	223
		3. 5. 1 建立中断程序	223
		3. 5. 2 中断程序赋值和启动 (SETINT, PRIO, BLSYNC)	224
		3. 5. 3 取消/再激活一个中断程序的赋值 (DISABLE, ENABLE)	225

3.5.4 删除中断程序的赋值 (CLRINT) ...	225	4.3.3 切削剩余 (CYCLE952)	271
3.5.5 快速离开工件轮廓 (SETINT, LIFTFAST, ALF)	226	4.3.4 切槽 (CYCLE952)	272
3.5.6 快速离开工件轮廓时的运动方向	227	4.3.5 切槽剩余 (CYCLE952)	274
3.5.7 中断程序下的运动过程	228	4.3.6 切槽加工 (CYCLE952)	275
3.6 轴交换和转移	228	4.3.7 切槽加工余料 (CYCLE952)	276
3.6.1 交换轴, 交换主轴 (RELEASE, GET, GETD)	228	4.4 钻孔循环	277
3.6.2 将轴移交到另一个通道中 (AXTOCHAN)	230	4.4.1 概述	277
3.7 子程序	231	4.4.2 定心 (CYCLE81)	279
3.7.1 概述	231	4.4.3 钻孔 (CYCLE82)	279
3.7.2 定义子程序	234	4.4.4 铰孔 (CYCLE85)	280
3.7.3 子程序调用	246	4.4.5 深孔钻削 (CYCLE83)	280
3.7.4 循环, 给用户循环设定参数	253	4.4.6 镗孔 (CYCLE86)	282
3.8 宏指令技术	255	4.4.7 攻螺纹 (CYCLE84, 840)	283
第4章 编程工艺功能 (循环)	257	4.4.8 钻孔螺纹铣削 (CYCLE78)	286
4.1 概述	257	4.4.9 定位和位置模式	288
4.2 车削循环	259	4.4.10 任意位置模式 (CYCLE802)	288
4.2.1 车削 (CYCLE951)	259	4.4.11 位置模式直线 (HOLES1), 栅格或框架 (CYCLE801)	289
4.2.2 切槽 (CYCLE930)	260	4.4.12 圆弧位置模式 (HOLES2)	290
4.2.3 E形和F形退刀槽 (CYCLE940)	262	4.4.13 位置重复	291
4.2.4 螺纹退刀槽 (CYCLE940)	262	4.5 其他循环和功能	292
4.2.5 螺纹车削 (CYCLE99)	263	4.5.1 高速设定 (CYCLE832)	292
4.2.6 链式螺纹 (CYCLE98)	266	4.5.2 子程序	292
4.2.7 切断 (CYCLE92)	268	4.6 其他 ShopTurn 循环和功能	293
4.3 车削轮廓循环	269	4.6.1 中心钻孔	293
4.3.1 概述	269	4.6.2 中心攻螺纹	295
4.3.2 切削 (CYCLE952)	269	4.6.3 转换	296
		4.6.4 直线或圆弧加工	297
		4.6.5 用副主轴加工	300
		附录 G 功能组	303

第 1 章 编程基础

1.1 概述

普通机床是由操作工控制的。操作工的技能决定了机床的加工能力、质量和效率。优秀操作工是企业的宝贵财富。

数控机床是由数控系统控制的。数控系统的档次决定了机床的档次。机床的加工靠运行零件加工程序，当加工程序一定时，机床的加工能力、质量和效率不受操作者的影响，甚至可以实现无人加工。加工程序由程序员设计，对于简单或不太复杂的零件，也可以由操作者完成。又好又快地编出加工程序，可以显著提高机床的生产效率。

数控系统的编程指令是由各系统生产厂决定的，因此，即使是同一个机床厂家生产的机床，若安装不同系统生产厂家的系统，其编程是不同的。因此，本书不以机床来介绍编程，而直接介绍数控系统的编程。一般数控系统均有直线和圆弧插补功能，若程序员将加工对象处理成系统能执行的直线和圆弧，则系统就能控制机床加工出所要求的零件形状。所以，决定系统能力的是程序员，而不是操作者的技能，当然，操作者也可以是程序员。因此，程序员应努力钻研数控系统的编程功能，提高编程效率。有人统计在数控技能竞赛中，不同的编程方法其效率是不同的，试题规定编程和加工用时为 6h，用参数编程，仅用 4h，而用 CAM 编程，用 6h 还未完成一半。

828D 系统是集车、铣、磨和冲于一体的系统。当车床的主轴变为伺服轴、刀塔上具有旋转刀具功能时，车床就变为车削中心，具有车削和铣削加工功能。本书仅介绍车削（含车削中心）编程。西门子系统的编程功能是向下兼容的，即 802D 的程序在 828D 上可以运行，828D 的程序可以在 840D 上运行。

828D 将编程功能分为两部分，一部分是基础部分，介绍符合 DIN66025 标准，即 ISO 标准的常用指令和语句；另一部分是工作准备部分，即工艺人员可以利用控制系统的专用编程语言对复杂的工件加工进行编程，例如，对自由成形曲面、通道坐标等进行编程，可提高编程效率。

828D 没有提供 G 代码编程工艺功能（循环）编程，而是以 ShopTurn 程序来介绍。本书将其放在编程部分介绍。

西门子公司将编程功能分为基本功能、扩展功能和特殊功能。对一个具体的机床和所配置的系统，它所具有的功能便不再区分，都是机床具有的功能，都应该掌握。

1.2 几何原理

1.2.1 工件位置

被加工点的位置在工件坐标系中定义，在 NC 程序中给定。为了使机床和系统能按照

NC 程序给定的位置加工，这些数据必须与标准坐标系一致，并给出机床轴的运动方向。在工件坐标系中定义被加工点，有以下几种方法：

1. 直角坐标系

在直角坐标系中给定轴的尺寸，定义坐标轴（ X 、 Y 和 Z ）的方向和数据。工件零点始终为坐标 $X0$ 、 $Y0$ 和 $Z0$ ，例如， $P1$ 相对于工件零点的位置为 $X100 Y45 Z-5$ 。

2. 极坐标

在定义工件位置时，还可以使用极坐标来代替直角坐标。如果一个工件或者工件中的一部分是用半径和角度标注的，则用这种方法定义就非常方便。标准尺寸的原点就是极点。极点与被加工位置之间的距离就是极坐标极径（ RP ）。极坐标极径与工作面水平轴之间的角度就是极坐标极角（ AP ）（以弧度 rad 计量），逆时针方向为正，顺时针方向为负。例如， $P1$ 可以表示为 $RP = 100 AP = 30$ 。

3. 绝对尺寸

绝对尺寸是以当前有效的零点为基准的位置数据，绝对尺寸数据是该点的坐标位置。例如， $P1$ 的绝对尺寸为 $X20 Y35$ 。

4. 增量尺寸

尺寸不是以零点为基准，而是以另外一个点为基准。为了避免不必要的换算，可以使用相对尺寸（增量尺寸）数据，在该尺寸系统中，位置数据分别以前一个点为基准，因此，相对尺寸是运动的距离。例如， $P2$ 对 $P1$ 点的距离为 $X30 Y20$ 。

1.2.2 工件平面

NC 程序必须包含指定加工平面所在平面的信息，只有这样，系统才能在处理 NC 程序时正确计算刀具补偿值，进行圆弧插补和处理极坐标系中的数据。

每两个坐标轴就可以确定一个工作平面，而第三坐标轴垂直于该平面，并在二维加工中确定刀具的进给方向：

在 NC 程序中使用 G 指令 G17、G18 和 G19 定义工作平面。

G17 定义 XY 平面。

G18 定义 ZX 平面。

G19 定义 YZ 平面。

第 1 地址为横坐标，第 2 地址为纵坐标，第 3 地址为垂直坐标。车床为 G18 平面。

1.2.3 零点和特征点

1. 零点

零点用双圆（同心圆）表示，有如下几种：

（1）机床零点 使用机床零点可以确定机床坐标系（MCS），所有其他零点都以机床零点为基础。机床零点表示为 ，并用字母 M 标识。

（2）工件零点 以机床零点为基准的工件零点可以确定工件坐标系。表示为 ，并用字母 W 标识。工件零点一般为程序零点。

（3）定位点 定位点表示为 ，用字母 A 标识。在车床上可以与工件零点重合。

2. 特征点

特征点用单圆表示，有如下几种：

(1) 参考点 它和机床零点 M 之间的距离必须已知，用此处的轴位置准确地设定距离值，表示为 ，并用字母 R 标识。如果该距离为零，则参考点与机床零点重合。

(2) 起始点 起始点可以由程序确定，刀具从该点出发，表示为 ，并用字母 B 标识。

(3) 刀架基准点 该点位于刀具夹具安装位置上，通过输入刀具长度，控制系统可以计算出刀尖（刀位点）与刀架基准点的距离，表示为 ，并用字母 T 标识。

(4) 换刀点 换刀点表示在此处换刀，表示为 ，并用字母 N 标识。

1.2.4 坐标系

西门子系统的坐标系分为：

1) 机床坐标系（MCS），使用机床零点 M 。

2) 基准坐标系（BCS）。

3) 基准零点坐标系（BNS）。

4) 可设定的零点坐标系（ENS）。

5) 工件坐标系（WCS），使用工件零点 W 。

工件坐标系与标准坐标系平行。

1. 标准坐标系

国际标准化组织（ISO）对数控机床的坐标和方向制订了统一的标准 [ISO 841 (1974)]，为右手直角（笛卡儿）坐标系。西门子公司根据 DIN 66217 标准（与 ISO 标准等同）使用右旋直角（笛卡儿）坐标系，用拇指、食指、中指决定坐标系，称右手“三指定则”。规定基本的直线运动坐标轴用 X 、 Y 、 Z 表示，围绕 X 、 Y 、 Z 轴旋转的坐标轴分别用 A 、 B 、 C 表示。规定空间直角坐标系 X 、 Y 、 Z 三者的关系及方向由右手定则判定，拇指、食指、中指分别表示 X 、 Y 、 Z 轴及其方向， A 、 B 、 C 的正方向用右手螺旋法则判定，即拇指分别代表 X 、 Y 、 Z 的正向，其余 4 指握拳代表回转轴正向。工件固定，刀具移动时采用上面规定的法则；如果工件移动，刀具不动时，正方向反向，并加“'”表示。

这样规定之后，程序员在编程时不必考虑具体的机床是工件固定，还是工件移动的情况，永远假定工件不动、刀具移动来决定机床坐标的正方向。

2. 机床坐标系（MCS）

机床坐标系由机床实际的坐标轴构成。在机床坐标系中定义参考点、刀具点和托盘更换点（机床固定点）。

机床坐标系的零点由机床生产厂家确定。

机床坐标系一般不用于零件加工编程。

3. 基准坐标系（BCS）

不带运动转换的机床，例如，TRANSMIT/TRACYL 转换，若将 BCS 投影到 MCS 上时，

则二者是重合的。在该机床上，机床轴与几何轴可以使用相同的名称。

带运动转换的机床，若将 BCS 投影到 MCS，则二者将不重合。在该机床上，机床轴与几何轴必须使用不同的名称。

编程总是在工件坐标系（WCS）中进行。但在加工时，若使用回转轴或非垂直结构的直线轴，即使用倾斜轴的机床时，为了使 WCS 中编程的坐标投影到实际的机床轴运动中，需要用到运动转换。

4. 基准零点坐标系（BNS）

基准零点坐标系（BNS）由基准坐标系（BCS）偏移后得到。基准偏移表示 BCS 和 BNS 之间的坐标转换，由以下几部分组成：

- 1) 外部零点偏移。
- 2) DRF（手轮偏移）。
- 3) 链接的系统框架。
- 4) 链接的基准框架。

基准偏移可以确定例如托盘零点等的参数。

5. 可设定的零点坐标系（ENS）

通过可设定的零点偏移，可以由基准零点坐标系（BNS）得到可设定的零点坐标系（ENS）。在 NC 程序中使用 G 指令 G54 ~ G57 和 G505 ~ G599 来激活可设定的零点偏移。

当可编程的坐标转换（框架）未激活时，ENS 即为工件坐标系（WCS）。在一个 NC 程序中，有时需要将原先选定的工件坐标系或者可设定零点坐标系，通过位移、旋转、镜像或缩放定位到另一个位置，这可以通过可编程的坐标转换（框架）进行。可编程的坐标转换（框架）总是以可设定的零点坐标系为基准。

6. 工件坐标系（WCS）

在工件坐标系（WCS）中给出一个工件的几何尺寸，NC 程序中的数据以工件坐标系为基准。

工件坐标系始终是直角坐标系，并与具体的工件相联系。

7. 各种坐标系之间的相互关系

各种坐标系之间的相互关系如下：



其中：

- 1) 如果运动转换未激活，即机床坐标系与基准坐标系重合。
- 2) 通过基准偏移得到带有托盘零点的基准零点坐标系（BNS）。
- 3) 通过可设定的零点偏移 G54 或 G55 来确定工件 1 或工件 2 的可设定零点坐标系（ENS）。
- 4) 通过可编程的坐标转换确定工件坐标系（WCS）。

1.2.5 进给轴

在编程时可以有以下几种轴：机床轴、通道轴、几何轴、辅助轴、轨迹轴、同步轴、定

位轴、指令轴（同步运行）、PLC 轴、链接轴和引导链接轴。

其中，几何轴、同步轴和定位轴可以编程。

轨迹轴根据编程指令以进给率 F 运行。

同步轴与轨迹轴同步运行，运行时间与所有轨迹轴相同。

定位轴与所有其他的轴异步运行，其运行不受轨迹轴和同步轴运行的影响。

指令轴与所有其他的轴异步运行，其运行不受轨迹轴和同步轴的影响。

PLC 轴受 PLC 控制，可以与所有其他的轴异步运行，其运行不受轨迹轴和同步轴运行的影响。

1. 主要轴/几何轴

主要轴确定了一个右手直角坐标系（也称右旋直角坐标系），在该坐标系可以编程刀具运行。在 NC 工艺中，将主要进给轴称为几何轴。

可转换的几何轴，该轴对零件程序中通过机床数据配置的几何轴进行修改。

轴名称，几何轴 X 、 Z 和 Y 。

在对框架和工件几何尺寸（轮廓）进行程序设计时，最多可以使用 3 个几何轴，只要能进行映射，几何轴和通道轴的名称允许相同。在每个通道中，几何轴和通道轴名称可以相同，可以执行同样的程序。

2. 辅助轴

与几何轴相反，在辅助轴中没有定义这些轴之间的几何关系。典型的辅助轴有：刀具转塔轴和加料机轴。

3. 主轴、主主轴

哪个轴为主要主轴由机床运动特性确定，通常通过机床数据将该主轴定义为主主轴。该定义可以通过程序指令 SETMS（< 主轴编号 >）更改。编程 SETMS 时，如果未设定主轴编号，则切换回机床数据中确定的主主轴。某些功能，比如螺纹切削，只适用于主主轴。

4. 机床轴

机床轴指的是在机床上实际存在的轴，进给轴的运行也可通过转换分配到机床轴。如果为机床设置了转换，则必须在开机调试时（指机床制造商）确定不同的轴名称。仅在特殊的情况下才对机床轴名称进行编程，比如在返回参考点或固定挡块时需编程轴名称。

轴名称可以通过机床数据设定，默认设定中名称为 $X1$, $Y1$, $Z1$, $A1$, $B1$, $C1$, $U1$, $V1$ 。此外，还有始终可以使用的特定轴名称： $AX1$, $AX2$, ..., $AX < n >$ 。

5. 通道轴

通道轴指的是一个通道中运行的所有轴。轴名称为 X , Y , Z , A , B , C , U , V 。

6. 轨迹轴

轨迹轴描述了轨迹行程及空间内的刀具运行，编程的进给率在该轨迹方向一直有效。参加该轨迹的进给轴同时到达其位置。通常它们是几何轴，但哪些轴为轨迹轴并可以影响速度由默认设置定义。在 NC 程序中，可以使用 FGROUP 设定轨迹轴。

7. 定位轴

定位轴单独插补，也就是说，每个定位轴有自己的轴插补器和进给率。定位轴不与轨迹轴一同插补。定位轴由 NC 程序或者 PLC 运行。如果一个轴同时由 NC 程序和 PLC 运行，则输出报警。典型的定位轴有：工件上下料的装料机，刀库/刀塔。

其区别在于定位轴是同步到达程序段终点，还是通过多个程序段到达终点。指令如下：

POS 轴：当所有在该程序段中编程的轨迹轴和定位轴到达它们编程的终点后，在程序段结束处执行程序段切换。

POSA 轴：定位轴的运动持续多个程序段。

POSM 轴：刀库定位。

POSP 轴：定位轴分段运行至终点位置（摆动）。

注意，没有特殊标识 POS/POSA 的定位轴变为同步轴运行。只有当定位轴（POS）在轨迹轴之前到达其终点位置时，轨迹轴才可以使用连续路径运行（G64）。POS/POSA 编程的轨迹轴从轨迹轴组中撤出，用于此程序段。

8. 同步轴

同步轴的运行和轨迹行程同步，即从起点开始到编程的终点位置。在 F 中编程的进给率适用于所有在程序段中编程的轨迹轴，但是不适用于同步轴。同步轴运行时间与轨迹轴相同。比如同步轴可以是一个回转轴，它与轨迹插补同时运行。

9. 指令轴

在同步工作中，指令轴通过一个事件（指令）启动，它们可能会与零件程序完全异步地定位、启动和停止。一个轴不能同时在零件程序和同步动作中运行。指令轴单独插补，也就是说，每个指令轴有自己的轴插补器和进给率。

10. PLC 轴

PLC 轴由 PLC 通过主程序中特殊的功能块运行，可以与所有其他的轴异步运行。其运行不受轨迹和同步运行的影响。

11. 链接轴

链接轴与另一个 NCU 以物理形式相连，并处于位置闭环中。链接轴可以动态地分配至另一个 NCU 通道，对于特定的 NCU，链接轴不是本地轴。

轴容器方案用于动态变更对 NCU 的分配。在零件程序中通过 GET 和 RELEASE 进行的轴交换不适用于链接轴。其前提条件是：①相关的 NCU1 和 NCU2 必须通过链接模块进行快速通信。②轴必须通过机床数据进行相应的配置。③必须选择了“链接轴”选项。

位置闭环位于轴与驱动物理连接的 NCU 上，相应的轴 VDI 接口也位于该 NCU 上。链接轴的位置设定值在另一个 NCU 上生成，并通过 NCU 链接进行通信。

通过链接通信实现插补器与位置控制器以及 PLC 接口之间的协同运行。必须将通过插补器计算的设定值传输到源 NCU 上的位置环中，或必须将实际值重新传回去。

轴容器是指一种环形缓冲数据结构，其中进行本地轴/链接轴和通道轴的分配，环形缓冲器中的记录为循环浮动。

12. 引导链接轴

引导链接轴是指一个轴通过一个 NCU 插件，将一个或多个其他的 NCU 作为引导轴使用，用于引导跟随轴。与引导链接轴相联系的 NCU 可使用以下引导链接轴的耦合：①引导值（设定值、实际值、模拟引导值）。②耦合运动。③同步主轴。

引导 NCU：只有物理分配了引导值的轴的 NCU 才可以为该轴编程运行指令，编程不必考虑特殊情况。

跟随轴的 NCU：在跟随轴的 NCU 中编程时，不可为引导链接轴（引导值轴）编程运行指令，否则报警。

引导链接轴通过通道轴名称按通常的方式应用，引导链接轴的状态可以通过所选择的系统变量进行存取。

下面的系统变量可以与引导链接轴的通道轴名称一起使用：

\$AA_LEAD_SP：模拟的引导值_位置。

\$AA_LEAD_SV：模拟的引导值_速度。

1.3 编程规则

各系统生产厂家规定的编程规则是不同的，因此零件加工程序互不相同。西门子系统的 NC 程序要求符合 DIN66025 标准。

NC 程序可以分为主程序和子程序。一般程序按主程序运行，当遇到子程序时，按子程序运行，运行结束后返回主程序继续运行，直至程序结束。

程序由程序名、若干程序段和程序结束符组成。

1.3.1 程序名

1. 程序命名规则

每个 NC 程序要有一个名称（标识符）。在创建程序时要按照下列规则选择：

(1) 名称的长度不得超过 24 个字符 在 NC 屏幕上只能显示 24 个字符。

(2) 允许使用的字符

1) 字母，大小写均可，A ~ Z，a ~ z。

2) 数字 0 ~ 9。

3) 下划线_。

(3) 程序名开始的两个字符

1) 两个字母。

2) 一条下划线和一个字母。

例：_MPF100, WELLE, WELL_2。

当满足该条件时，才能通过程序名称将 NC 程序作为子程序从其他程序中进行调用。若程序名称用数字开头，子程序调用只能通过 CALL 指令进行。

2. 穿孔带格式文件

通过 V24 接口读入到 NC 中的外部创建的程序文件，必须以穿孔带格式保存。对于穿孔带格式文件的名称，其附加规则如下：

1) 程序名称必须以字符“%”开始，即% <名称>。

2) 程序名称必须有一个 3 位长度的标识，即% <名称> _ x x x。例如：
% Flansch_MPF。

3) 存储在 NC 程序存储器内部的文件，其名称以“_N_”开始。例如：%_N_轴
123_MPF。

1.3.2 程序分量

每个程序段都包含有工件加工时执行加工步骤的数据。

1. 符合 DIN66025 的指令

符合 DIN66025 的指令由一个地址符和一个数字或者一串数字组成，它们表示一个算术值。

地址符（通常为一个字母）用来定义指令的含义。例如：

G，即 G 功能（准备功能）。

X，用于 X 轴的行程信息。

S，指定主轴转速。

数字串表示赋给该地址的值。数字串可以包含一个符号和小数点。符号位于地址字母和数字串之间，正号（+）和后续的零（0）可以省去。例，G1 X-50 S2000。

2. NC 高级语言元素

由于 DIN66025 所规定的指令程序段已经无法应对先进机床上的综合加工过程编程，因此将 NC 高级语言元素扩展到程序段中。其中包括：

（1）NC 高级语言指令 与 DIN66025 指令不同，NC 高级语言指令由多个地址符构成。例如：OVR，用于转速补偿（倍率）。SPOS，用于主轴定位。

（2）标识符（定义的名称） 用于系统变量，用户定义变量，子程序，关键字，跳转标记，宏。

注意：标识符必须唯一，不可以用于不同的对象。

（3）比较运算符、逻辑运算、运算功能和控制结构 详见第 3 章 灵活的 NC 编程。

3. 指令的有效性

指令分模态指令和程序段方式指令：

（1）模态指令 模态有效的指令可以一直保持编程值的有效性，在所有后续程序段中一直有效，直到编程新值或编程一个使它失效的指令。

（2）程序段方式指令 程序段方式有效的指令仅在编程的程序段中有效。

1.3.3 程序段规则

1. 程序段开始

程序段可以在其开始处使用程序段号进行标识。程序段号由一个字符“N”和一个正整数构成。例：N40。

程序段号的顺序可以任意，推荐使用升序的程序段号。

在一个程序中程序段号必须唯一，这样在查找时会有一个明确的结果。

2. 程序段结束

程序段以字符“LF”（换行符）结束。

字符“LF”可以省略，可以通过换行自动生成。

3. 程序段长度

一个程序段可以包含最多 512 个字符，包含注释和程序段结束符“LF”。

在通常情况下，在屏幕上一次显示 3 个程序段，每个程序段最多 66 个字符。注释也同

样显示。信息则在独立的信息窗口显示。

4. 指令的顺序

为了使程序段结构清晰明了，程序段中的指令应按如下顺序排列：

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

有些地址可以在一个程序段中多次使用，比如 G、M、H。

1.3.4 赋值规则

有些地址采用下列规则赋值：

1) 当地址是由几个字符构成，或值由几个常数构成时，地址与值之间必须写入符号“=”（等号）。

如果地址是单个字母，并且值仅由一个常量构成，则可以不写符号“=”。

2) 允许使用正负号。

3) 可以在地址字符之后使用分隔符。

例：X10 ; 给单个地址 X 赋值，不要求写“=”符号

X1 = 10 ; 地址(X)带扩展数字(1)，赋值(10)要求写“=”符号

X = 10 * (5 + SIN(37.5)) ; 通过表达式进行赋值，要求写“=”符号

在数字扩展后必须紧跟“=”，“（”，“[”，“）”，“]”，“，”等符号中的一个，或者一个运算符，从而可以把带数字扩展的地址与带数值的地址字母区分开。

1.3.5 注释

为了使 NC 程序更容易理解，可以在 NC 程序段加上注释。注释放在程序段的结束处，并且用分号“;”将其与程序部分隔开。

例：N10 ; 公司 G&S，任务号 12A71

N20 ; 程序由 Mueller 先生编制，部门 TV4，时间 94.11.21

N50 ; 零件号 12，潜水泵壳体，型号 TP23A

N70 G1 F100 X10 Y20 ; 解释 NC 程序段的注释

注释语句被存储，并在程序运行时显示在程序段之后。

1.3.6 信息显示 (MSG)

通过编程提示信息，操作人员可以在程序运行时了解当前的加工情况。编程：

MSG (“<信息文本>”)

MSG ()

其中：

MSG：信息文件编程的关键字；

<信息文本>：作为信息显示的字符串。

一个信息文本的长度最多可以为 124 个字符，分为两行显示（每行 62 个字符）。

在一个提示信息文本之内也可以显示变量的内容。

通过编程无信息文本的 MSG () 可以清除一条信息。

例 1：激活/清除信息