

新世纪电气自动化系列精品教材

XINSHIJIDIANQIZIDONGHUAXILIEJINGPINJIAOCAI

SHUKONGJIAGONGGONGYI
YUBIANCHENG

数控加工工艺 与编程

缪德建 顾雪艳 / 主



东南大学出版社
SOUTHEAST UNIVERSITY PRESS

新世纪电气自动化系列精品教材

数控加工工艺与编程

主编 缪德建 顾雪艳



东南大学出版社
SOUTHEAST UNIVERSITY PRESS

· 南京 ·

内容提要

本书编写结合企业实际案例,由数控技术应用专业领域资深一线教师依据相关课程教学与培训基本要求,结合多年的教学经验,坚持课程改革新理念,教材内容突出应用性和实践性。引用企业实际加工案例,确保先进性和权威性。力求将专业知识与专业技能相结合,体现先进性、实用性、易懂性。

本书共分8章,第1章、第2章介绍了数控编程的基础知识和数控加工工艺基础;第3章介绍了数控车削工艺与编程基础;第4章、第5章介绍了FANUC0i数控车床系统编程和机床操作方法;第6章介绍了数控铣削编程基础;第7章、第8章介绍了SIEMENS 802D系统的编程和数控铣床的操作方法。

本书可作为普通高等院校数控技术、机械制造、模具设计、机电一体化等专业的教学用书,也可作为职业技术学院的教材、企业数控加工人员的参考用书。

图书在版编目(CIP)数据

数控加工工艺与编程/缪德建,顾雪艳主编. —南京:东南大学出版社,2013.8

新世纪电气自动化系列精品教材

ISBN 978-7-5641-4410-4

I. ①数… II. ①缪… ②顾… III. ①数控机床—加工—高等学校—教材 ②数控机床—程序设计—高等学校—教材 IV. ①TG659

中国版本图书馆 CIP 数据核字(2013)第 165781 号

数控加工工艺与编程

出版发行 东南大学出版社

出版人 江建中

社 址 南京市四牌楼2号

邮 编 210096

经 销 全国各地新华书店

印 刷 南京工大印务有限公司

开 本 787 mm×1092 mm 1/16

印 张 15

字 数 384千字

书 号 ISBN 978-7-5641-4410-4

版 次 2013年8月第1版

印 次 2013年8月第1次印刷

印 数 1—3500册

定 价 32.00元

(本社图书若有印装质量问题,请直接与营销部联系,电话:025—83791830)

前 言

制造业是我国工业的支柱产业,而数控加工是制造业的基础,提高数控加工的水平 and 效率关系到我国制造业的竞争能力。

本书由数控技术应用专业领域资深一线教师依据相关课程教学与培训基本要求,结合多年的教学经验和企业实际案例,在实习讲义的基础上编写而成。本书编写时坚持课程改革新理念,具有以下特色:

(1) 教材内容突出应用性和实践性。

在编写时以 FANUC、SIMENS 数控系统为编程及操作对象,从工程实际出发,整合数控编程、数控机床、数控加工工艺、切削原理、刀具、机床操作等相关知识,有大量实用的编程实例,涵盖了数控加工所需的主要内容。

(2) 引用企业实际加工案例,确保先进性和权威性。

本书在编写过程中得到山特维克可乐满公司技术人员的支持,书中涉及的主要加工案例采用了山特维克刀具的实际切削用量及其加工新工艺方法。

(3) 体现课改理念,创新教材编写风格。

本书编写风格适用于具有职业教育特色的“做中教、做中学”的教学模式和行为导向教学原则下,操作步骤要点突出,编程案例经典详实,操作部分的插图以截图为主,直观清晰,增强可读性。

本书第1章介绍了数控编程的基础知识;第2章介绍了数控加工工艺基础;第3章介绍了数控车削工艺与编程基础;第4章介绍了 FANUC0i 数控车床系统的编程方法;第5章介绍了数控车床的操作方法;第6章介绍了数控铣削的编程基础;第7章介绍了 SIEMENS 802D 系统的编程方法;第8章介绍了数控铣床的操作方法。

本书由缪德建、顾雪艳担任主编,全书由顾雪艳统稿。参加编写的人员有赵建峰、赵艺兵、李警仁、董彪、张帆。

本书在编写过程中参考了大量的文献资料,在此向文献资料的作者致以诚挚的谢意。由于编写时间及编写水平有限,书中难免有错误和不妥之处,恳请广大读者批评指正。

编 者
2013 年 6 月

目 录

1	数控编程基础	(1)
1.1	概述	(1)
1.1.1	数控编程的概念	(1)
1.1.2	数控编程的种类	(2)
1.1.3	数控编程的内容与步骤	(3)
1.2	数控机床的坐标系统	(4)
1.2.1	数控机床标准坐标系的规定	(4)
1.2.2	数控机床直角坐标轴、旋转运动坐标轴、附加坐标轴的确定	(4)
1.2.3	机床坐标轴的运动与联动	(6)
1.3	数控的程序结构与格式	(6)
1.3.1	程序结构	(6)
1.3.2	程序段格式	(7)
1.3.3	主程序与子程序	(7)
1.4	常用功能指令	(8)
1.4.1	准备功能 G 指令	(8)
1.4.2	辅助功能 M 指令	(10)
1.4.3	进给功能 F 指令	(12)
1.4.4	主轴转速功能 S 指令	(12)
1.4.5	刀具功能 T 指令	(12)
2	数控加工工艺基础	(13)
2.1	数控加工的切削基础	(13)
2.1.1	金属切削运动	(14)
2.1.2	工件表面的形成	(14)
2.1.3	切削用量三要素	(14)
2.1.4	切削用量选择	(15)
2.1.5	切屑与断屑	(16)
2.2	工件材料的分类	(18)
2.3	常见的切削刀具材料及性能	(20)
2.3.1	常见的切削刀具材料	(20)
2.3.2	刀具材料的成分组成和特性	(20)
2.4	数控加工工艺基础	(21)
2.4.1	数控加工工艺的主要内容	(21)

2.4.2	数控加工工艺的特点	(22)
2.4.3	数控加工的特点与适应性	(22)
2.4.4	数控机床的选用	(24)
2.4.5	数控加工的工艺文件	(25)
2.5	典型零件加工工艺制定	(31)
2.5.1	可乐瓶底加工工艺制定实例	(31)
2.5.2	叶片加工工艺制定实例	(32)
2.5.3	凸凹腔加工工艺制定实例	(35)
3	数控车削工艺与编程基础	(38)
3.1	数控车削工艺基础	(38)
3.1.1	切削原理	(38)
3.1.2	车刀、刀片(修光刃)的种类及其标记方法	(39)
3.1.3	车刀夹紧方式	(41)
3.1.4	刀具几何参数的选择原则	(41)
3.2	数控车削编程基础	(42)
3.2.1	概述	(42)
3.2.2	坐标系	(43)
3.2.3	数控车床常用指令及其特点	(44)
3.2.4	刀具功能	(48)
3.2.5	恒切削速度控制	(51)
3.2.6	螺纹切削的加工特点和切削用量选择	(51)
4	FANUC0i 数控车床系统编程	(56)
4.1	FANUC0i 系统常用指令功能	(57)
4.1.1	插补功能	(57)
4.1.2	倒角、倒圆编程功能	(58)
4.1.3	角度编程功能	(60)
4.1.4	坐标系功能	(61)
4.1.5	参考点功能	(61)
4.1.6	进给功能	(62)
4.1.7	主轴功能	(62)
4.1.8	刀具功能	(63)
4.1.9	辅助功能	(64)
4.2	螺纹切削功能	(64)
4.2.1	等螺距螺纹切削	(64)
4.2.2	螺纹切削循环	(66)
4.2.3	复合螺纹切削循环	(69)
4.3	固定循环功能	(70)
4.3.1	单一形状外径/内径切削循环功能	(70)
4.3.2	端面切削循环	(72)
4.3.3	复合型固定循环功能	(74)

4.4	综合实例	(78)
4.4.1	综合实例(一)	(78)
4.4.2	综合实例(二)	(80)
4.4.3	综合实例(三)	(82)
4.5	用户宏程序编程	(84)
4.5.1	宏程序格式	(85)
4.5.2	宏程序的调用	(88)
4.5.3	宏程序编程实例(一)	(90)
4.5.4	宏程序编程实例(二)	(92)
4.6	习题	(95)
5	FANUC0i 数控车床操作	(99)
5.1	机床面板介绍	(99)
5.1.1	设定与显示单元	(99)
5.1.2	功能键与软键	(100)
5.1.3	外部机床控制面板	(100)
5.2	机床返回参考点	(102)
5.3	手动方式操作(JOG)	(102)
5.4	手轮方式操作(HAND)	(104)
5.5	程序的输入与编辑	(104)
5.5.1	创建新程序	(104)
5.5.2	自动插入顺序号	(105)
5.5.3	程序清单的显示	(106)
5.5.4	字的插入、修改和删除	(106)
5.5.5	程序段的删除	(108)
5.5.6	程序号的检索	(109)
5.5.7	顺序号的检索	(110)
5.5.8	程序的删除	(110)
5.5.9	程序的复制	(111)
5.6	图形模拟	(112)
5.7	刀具参数与刀具补偿参数的设置	(114)
5.8	对刀操作	(115)
5.8.1	手动对刀	(115)
5.8.2	MDI 试切和磨耗设置	(117)
5.8.3	工件原点偏移设置	(118)
5.9	MDI 运行方式	(120)
5.10	空运行方式	(121)
5.11	自动加工方式(MEM)	(121)
5.12	与计算机进行数据传送方式(通信与 DNC)	(122)
5.12.1	通信接口参数的设置	(123)
5.12.2	DNC 运行	(123)

5.12.3	程序的计算机输入	(124)
5.12.4	程序的计算机输出	(124)
5.13	机床的维护与保养	(124)
5.13.1	一般维护与保养注意事项	(124)
5.13.2	每日维护与保养工作	(125)
5.13.3	每周的维护与保养工作	(125)
5.13.4	每月维护与保养工作	(125)
5.13.5	每半年的维护与保养工作	(125)
5.13.6	每年的维护与保养工作	(125)
5.14	数控车床的安全操作注意事项	(126)
6	数控铣削工艺与编程基础	(127)
6.1	数控铣削工艺基础	(127)
6.1.1	铣刀的种类和用途	(127)
6.1.2	铣削方式	(128)
6.1.3	铣刀刀柄类型	(132)
6.2	数控铣削编程基础	(132)
6.2.1	坐标系	(132)
6.2.2	刀具补偿	(133)
6.2.3	数控铣床常用指令	(139)
6.2.4	编程举例	(145)
7	SIEMENS 802D 系统数控铣床编程	(148)
7.1	指令代码	(148)
7.2	SIEMENS 802D 编程指令	(151)
7.2.1	绝对和增量的混合编程 G90、G91、AC、IC	(151)
7.2.2	圆弧插补指令	(151)
7.2.3	倒角和倒圆角	(152)
7.2.4	螺旋插补 G3/G2、TRUN	(152)
7.2.5	螺纹切削 G33、G63、G331、G332	(153)
7.2.6	极坐标/极点定义 G110、G111、G112	(154)
7.2.7	返回固定点、回参考点	(155)
7.2.8	可编程功能指令	(155)
7.2.9	切进切出指令	(159)
7.2.10	编程举例	(162)
7.3	循环指令	(165)
7.3.1	钻孔循环指令	(167)
7.3.2	攻丝循环指令	(172)
7.3.3	镗孔循环指令	(174)
7.3.4	铣槽循环指令	(177)
7.3.5	铣端面 CYCLE71	(181)

7.4 参数化编程	(182)
7.4.1 参数赋值	(183)
7.4.2 函数表达式	(183)
7.4.3 程序跳转	(184)
7.5 编程实例	(185)
7.6 习题	(195)
8 SIEMENS 802D 数控铣床操作	(201)
8.1 机床面板介绍	(201)
8.2 开机回参考点	(204)
8.3 手动方式操作(JOG)	(205)
8.4 手轮方式操作(HAND)	(207)
8.5 程序的输入与编辑	(208)
8.6 图形模拟	(211)
8.7 刀具参数及刀具补偿参数的设置	(212)
8.7.1 输入刀具参数及刀具补偿参数	(212)
8.7.2 建立新刀具	(213)
8.7.3 输入/修改零点偏置值	(214)
8.7.4 R 参数设置	(218)
8.8 对刀操作	(218)
8.8.1 确定刀具补偿值	(218)
8.8.2 确定工件补偿值	(220)
8.8.3 工件零点 MDA 方式检验	(221)
8.9 MDA 运行方式	(221)
8.9.1 MDA 基本设置	(221)
8.9.2 端面铣削	(222)
8.10 自动加工方式(AUTO)	(223)
8.10.1 选择和启动零件程序加工	(225)
8.10.2 程序段搜索加工	(225)
8.10.3 停止、中断及重新返回加工	(227)
8.11 与计算机进行数据传送与 DNC 加工	(227)
8.11.1 通过 RS232 接口进行数据传送	(227)
8.11.2 DNC 自动加工	(228)
8.11.3 数控铣床安全操作注意事项	(229)
参考文献	(230)

1.1 概述

数控编程是针对数控机床编制的加工程序。数控机床是近几十年来发展起来的一种新型自动化机床。它集机械制造、计算机、微电子、现代控制及精密测量等多种技术为一体,使传统的机械加工工艺发生了质的变化,使其整个加工过程实现自动化。近年来,各种数控机床的精确性、可靠性、集成性、柔性和宜人性等各方面功能越来越完善,它在机械行业的自动化加工领域中占有越来越重要的地位。

1.1.1 数控编程的概念

数控编程即数控加工程序的编制。数控加工程序是数控机床在加工中不可缺少的一部分,数控机床之所以能加工出各种形状、不同尺寸和精度的零件,就是因为编程人员为它编制了不同的加工程序。如图 1.1 所示,数控机床与普通机床最大的区别在于:普通机床是通过人工手动操纵机床手柄,为进给机构提供所需进给动力;数控机床则是由数控加工程序控制机床进给运动,即把数控加工程序送入数控机床的“指挥中心”——数控系统,再通过一个使数控机床执行运动的操作中心——伺服系统,从而带动进给机构使机床按数控加工程序的顺序自动加工。

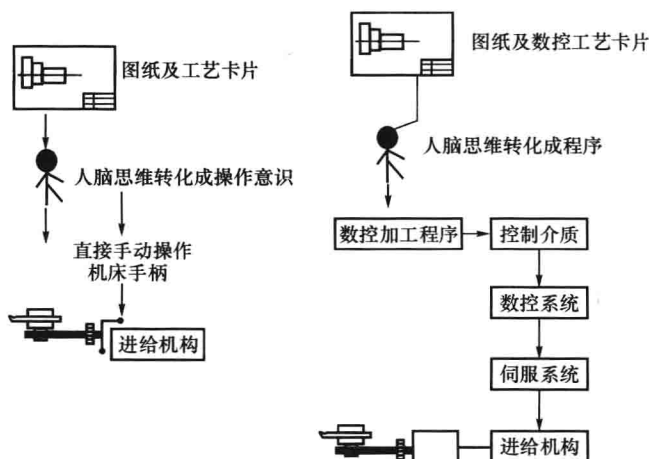


图 1.1 数控机床与普通机床的区别

数控加工程序就是把零件加工的工艺过程、工艺参数(进给速度和主轴转速等)、位移数据(几何形状和几何尺寸等)及开关命令(换刀、冷却液开/关和工件装卸等)等信息用数控系统规定的功能代码和格式按加工顺序编写成加工程序单,并记录在信息载体上,即制作成控制介质。

控制介质可以是:穿孔纸带、键盘、磁盘等各种可以记载二进制信息的媒体。通过数控机床

的输入装置,将信息载体上的数控加工程序输入机床数控系统,从而指挥数控机床按数控程序的内容加工出合格的零件。

已知信息对应功能代码如表 1.1 所示,图 1.2 所示是在数控铣床上加工台阶零件时编写加工程序的过程。

表 1.1 已知信息对应功能代码

信 息	功能代码
主轴正转、转速	M03,S1200
刀具	T,M06
快速运动	G00
快速运动终点坐标 P1	X,Y,Z
直线运动	G01
直线运动终点坐标 P2	X,Y,Z
进给速度	F
打开冷却液	M08

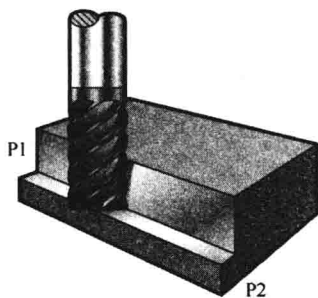


图 1.2 在数控铣床上加工台阶零件

假如,有加工任务:铣刀必须以 1 200 r/min 的速度顺时针转动,快速运动到起点 P1,铣刀垂直向下运行深至 5 mm,然后按一定的进给速度向右纵向铣削台阶面到 P2 点。

数控加工程序可写为:

```

N1 M03 S1200          (主轴正转,转速 1 200 r/min)
N2 T1 M06             (换 1 号刀具)
N3 G00 X50 Y50 Z5     (刀具中心快速移动到 P1 点)
N4 G01 Z-5 F60 M07     (刀具以 60 mm/min 的速度向下切深 5 mm,并打开冷却液)
N5 G01 X20 Y20 F80     (刀具以 80 mm/min 的速度铣削到 P2 点)
N6 G00 Z5             (快速抬刀到工件表面 5 mm 处)
N7 M30                (程序结束)
  
```

数控加工程序是有标准可循,有一定的格式要求的。常用的是 ISO 标准(国际标准化组织标准)和 EIA 标准(美国工业电子协会标准)。数控程序编写得如何,直接影响零件加工质量。

1.1.2 数控编程的种类

数控编程一般分为手工编程和自动编程两种。

手工编程是由人工完成刀具轨迹计算及加工程序的编制工作。当零件形状不十分复杂或加工程序不太长时,采用手工编程方便、经济。手工编程目前仍然是被广泛采用的编程方法。先进的自动编程方法中,许多重要的经验都来源于手工编程,手工编程是自动编程的基础,并不断推进自动编程的发展。

自动编程是计算机通过自动编程软件完成对刀具运动轨迹的自动计算,自动生成加工程序并在计算机屏幕上动态地显示出刀具的加工轨迹。当加工零件形状复杂,特别是涉及三维立体形状或刀具运动轨迹计算烦琐时,通常采用自动编程。

自动编程分为 APT 语言式自动编程和图形交互式自动编程两种方式。APT 语言式自动编程是用语言来编写源程序,由源程序通过计算机处理后自动生成加工程序。这种方法直观性

差,编程过程较复杂,不易掌握;随着计算机图形处理功能的增强,由图形到数控加工程序的计算机辅助编程技术应运而生。CAD/CAM 软件即可实现图形交互式自动编程。其速度快,精度高,直观性好,使用简便。目前常用的 CAD/CAM 软件有“UG”、“PRO-E”、“MASTER-CAM”、“CAXA”等。

1.1.3 数控编程的内容与步骤

数控编程中手工编程的步骤一般分为以下几个过程:



1) 分析零件图纸

拿到所要加工零件的图纸,首先应分析该零件的材料、形状、尺寸、精度以及毛坯形状和热处理要求等。其目的是选定在何种类型机床上进行加工。

2) 确定加工工艺过程

在分析零件图纸的基础上,确定加工顺序、加工路线、装卡方法,选择刀具、工装以及切削用量等工艺参数。同时充分利用数控机床的指令功能特点,简化程序,缩短加工路线,充分发挥机床效能。

3) 数值计算

根据已确定的加工路线和零件加工误差,计算出数控机床所需输入的数据。数值计算的复杂程度取决于零件的复杂程度和数控系统的功能。对于加工较简单的由圆弧与直线组成的平面零件,只需计算零件轮廓的相邻几何元素的交点或切点、起点、终点、圆弧的圆心坐标值或圆弧半径的坐标值。对于形状较复杂的零件并且组成该零件的几何元素与数控系统的插补功能不一致时,就需要较复杂的数值计算。例如对非圆曲线(如渐开线,阿基米德螺旋线等)需要用直线段或圆弧来逼近,在满足其精度要求的条件下计算出其节点坐标。对于这种情况,一般采用计算机辅助计算。

4) 编写程序单

加工路线、工艺参数及刀具运动轨迹的坐标值确定以后,编程人员可以根据数控系统具有的功能指令代码和程序段格式,编写加工程序单。必要时还应填写数控加工工序卡片和数控刀具卡片等有关工艺文件。

5) 制作控制介质

制作控制介质就是把编写好的程序单上的内容记录在控制介质即信息载体上,通过数控机床的输入装置,将信息载体上的数控加工程序输入机床数控装置。

6) 校验程序和首件试切

为了保证零件加工的正确性,数控程序必须经过校验和试切才能用于正式加工。一般通过图形显示和动态模拟功能或空走刀校验等方法以检查机床运动轨迹与动作的正确性。并通过对第一个零件的试切削,检验被加工零件的加工精度。若发现加工精度达不到要求时,应分析误差产生的原因,采取措施,加以纠正。

1.2 数控机床的坐标系统

在数控机床的编程中,为了便于描述机床运动,简化程序的编制,保证其通用性,数控机床的坐标和运动方向均已标准化,国际上采用 ISO 和 EIA 标准。我国采用 JB3051—1999《数控机床坐标和运动方向的命名》标准。

1.2.1 数控机床标准坐标系的规定

1) 标准坐标系的规定

为了精确控制机床移动部件的运动,需要在机床上建立一个坐标系,这个坐标系就叫标准坐标系,也叫机床坐标系。

数控机床的坐标系采用右手直角笛卡儿标准坐标系,如图 1.3 所示,坐标轴为 X 、 Y 、 Z 直角坐标,围绕 X 、 Y 、 Z 各轴的旋转运动轴为 A 、 B 、 C 。用右手直角笛卡儿坐标法则可判断 X 、 Y 、 Z 三轴的关系和正方向;用右手螺旋法则可判定三个直角坐标轴与 A 、 B 、 C 三个旋转轴的关系和 A 、 B 、 C 轴的正方向。当考虑刀具移动时,用 X 、 Y 、 Z 表示运动的正方向;当考虑工件移动时,则用 X' 、 Y' 、 Z' 表示运动的正方向。

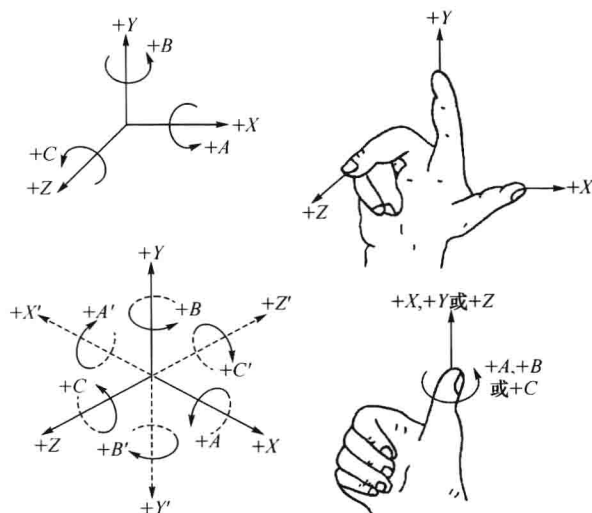


图 1.3 数控机床标准坐标系

2) 坐标和运动方向命名的原则

这一原则规定,不论机床的具体结构是工件静止、刀具运动,还是工件运动、刀具静止,在编写程序时一律看成是刀具相对于静止的工件而运动。机床某一部件运动的正方向,是增大工件与刀具之间距离的方向。

1.2.2 数控机床直角坐标轴、旋转运动坐标轴、附加坐标轴的确定

1) 数控机床直角坐标轴的确定

(1) Z 轴坐标 由传递切削力的主轴所决定,规定为平行于机床主轴的坐标轴。如果机床有一系列主轴,则选尽可能垂直于工件装夹面的主要轴作为 Z 轴。 Z 轴的正方向是增大工件与

刀具之间距离的方向,如图 1.4 所示。

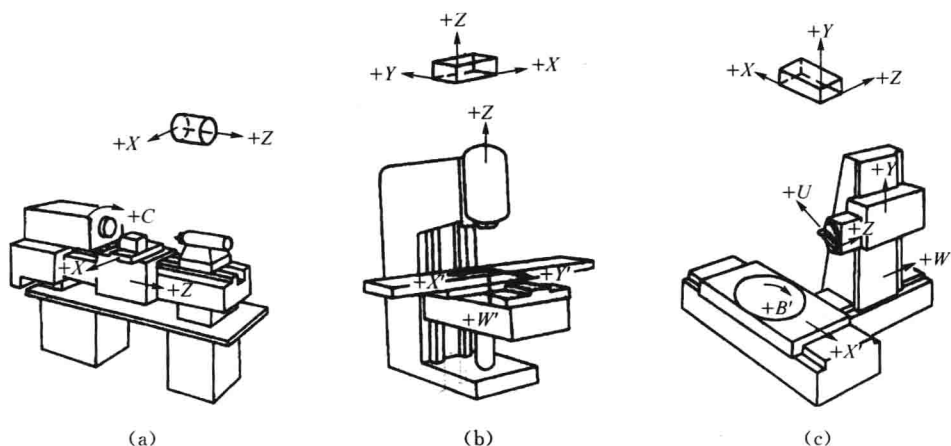


图 1.4 常见数控机床的坐标系

(2) X 轴坐标 作为水平的、平行于工件装夹平面的轴,它平行于主要的切削方向。对于工件旋转运动的机床(如车床),X 轴方向是指直径方向。正方向是增大工件与刀具之间距离的方向,如图 1.4 所示。

(3) Y 轴坐标 垂直于 X 及 Z 坐标,当 +X、+Z 确定后,按右手直角坐标系确定,如图 1.4 所示。

2) 旋转运动坐标轴的确定

旋转运动坐标轴 A、B 和 C 分别表示其轴线平行于 X、Y、Z 坐标的旋转运动。其正方向是按照右旋螺纹前进的方向。

3) 附加坐标轴的确定

对于直线运动:X、Y、Z 为主坐标系,或称第一坐标系。若有平行于 X、Y、Z 的第二组坐标和第三组坐标,则分别指定为 U、V、W 和 P、Q、R。靠近主轴的直线运动为第一坐标系,稍远的为第二组坐标,如图 1.5 所示。

对于旋转运动:除 A、B、C 之外,若还有第二组坐标,则分别指定为 D、E、F,如图 1.6 所示。

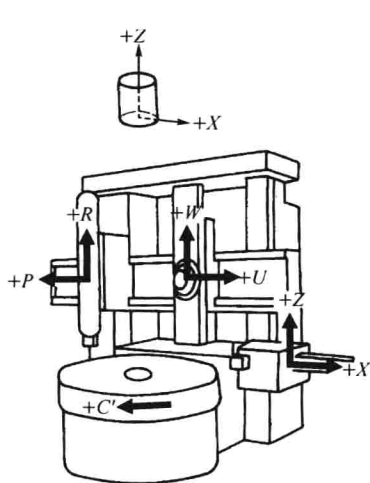


图 1.5 数控双柱立式车

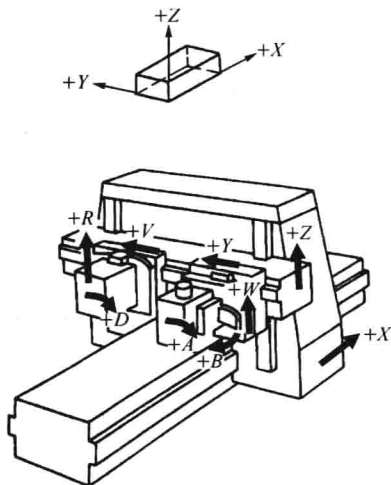


图 1.6 龙门移动式数控铣

1.2.3 机床坐标轴的运动与联动

数控机床的坐标运动由数控系统来控制。不同的数控系统控制的坐标轴数不同,数控系统可以控制坐标运动或坐标联动。坐标联动和坐标运动的概念不同,坐标联动是控制某几个坐标轴同时到达某个目标点,如数控车床是两轴联动,可控制 X 、 Z 坐标轴走直线或圆弧插补运动,使两轴同时到达某个目标点;坐标运动的概念是数控机床控制系统可以控制坐标运动,但不一定是坐标同时联动到达某个目标点。如一台数控加工中心控制 X 、 Y 、 Z 、 A 四轴运动,但只能 X 、 Y 、 Z 三轴联动同时到达某个目标,则称为四轴三联动。若 X 、 Y 、 Z 三轴坐标可以 X 、 Y 轴联动, Y 、 Z 轴联动和 X 、 Z 轴联动时,称为两轴半联动。数控系统控制的坐标轴数越多,同时联动的轴越多,系统功能就越强,价格也就越昂贵。如图 1.7 所示叶轮零件必须用五轴联动的数控机床才能加工出来。图 1.8 所示的车削中心可以控制八个坐标轴运动,但不一定是八个坐标轴联动。

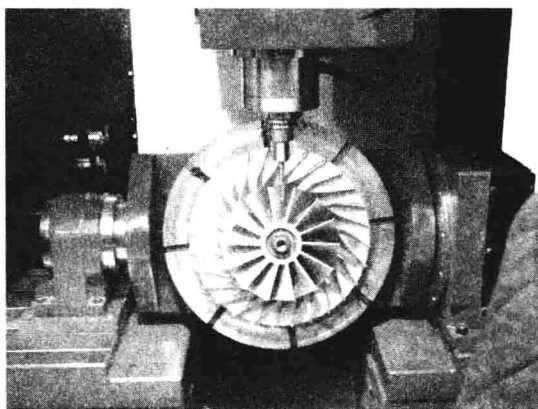


图 1.7 叶轮加工图

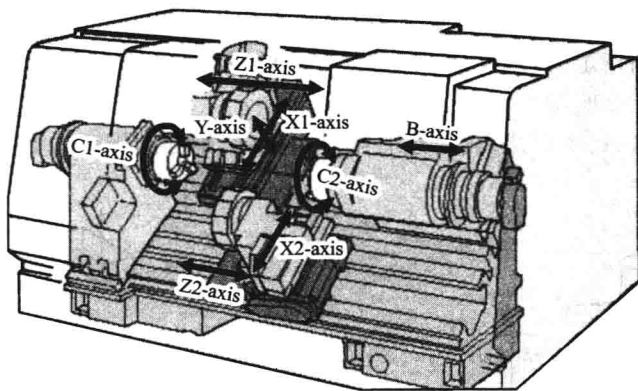


图 1.8 车削中心坐标轴示意图

1.3 数控的程序结构与格式

1.3.1 程序结构

数控加工程序是由若干程序段组成,程序段由一个或若干个指令字(如 $G01$)组成。指令字代表某一信息单元,由地址符(如 G)和数字(01)组成,它代表机床的一个位置或一个动作;地址

符由字母组成,每一个字母、数字和符号都称为字符。一个完整的加工程序包括开始符、程序名、程序主体和程序结束指令。加工程序的结构如图 1.9 所示。

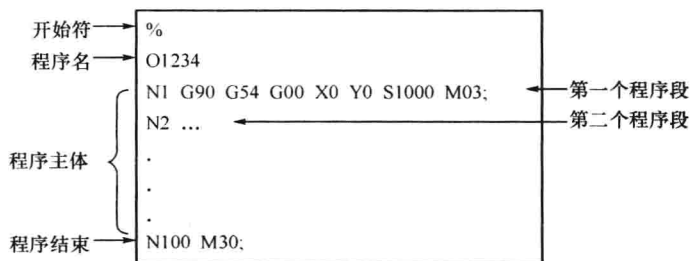


图 1.9 加工程序的结构

起始为程序段序号功能字,随后为工艺和几何方面的功能字,段末以换行符或回车符结束。其中准备功能字(G)指定数控系统应准备好某种运动和工作方式;辅助功能字(M)指定数控系统在加工过程中的辅助开关量控制功能。例如某数控加工程序段如下:

N010 M03 S800 T01 M06 G01 X60 Y40 F60;

表示本程序段为第 10 段,主轴以转速 800 r/min 顺时针方向旋转,换用 1 号刀具,机床运动部件以直线形式移动到 $X=60\text{ mm}$ 、 $Y=40\text{ mm}$ 的点,移动速度为 60 mm/min 。

1.3.2 程序段格式

程序段格式是指程序段中字、字符和数据的安排形式。分为固定程序段格式和可变程序段格式。常用的是字地址可变程序段格式。这种格式的程序段由若干个字组成,字首是一个英文字母作为地址符。在此格式程序段中,上一段程序中已写明,本程序段又不必变化的那些字仍然有效,例如对于模态(时序有效)G 指令(如 G00、G01),若在前面程序段中已有则可不再重写。这种格式的程序段,每个字长不固定,各个程序段的长度和程序字的个数都是可变的。

如下两段程序,字数和字符的个数就相差很大:

N20 G01 X50 Y60 Z10 F60 M03 S600 T0101 M06;

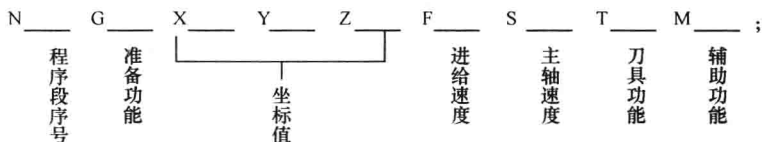
N21 X100;

第一个程序(N20 段)也可写成:

N20 T0101 M06 M03 S600 G01 X50 Y60 Z10 F60;

或 N20 M03 S600 G01 X50 Y60 Z10 F60 T0101 M06;

即同一程序段中各指令字的位置可以任意排列。但为了书写、输入、检查和校验方便,习惯按一定的顺序排列,如 N、G、X、Y、Z、F、S、T、M 顺序。



1.3.3 主程序与子程序

在程序中,若某一固定的加工操作重复出现时,可把这部分操作编写成子程序,然后根据需要调用,这样可使程序变得非常简单。调用第一层子程序的指令所在的加工程序叫做主程序。一个子程序调用语句,可以多次重复调用子程序。主程序与子程序的执行关系如下:

主程序	子程序
O0001	O1010
N0010……;	N1020……;
N0020 M98 P1010 L2;	N1030……;
N0030 ……;	N1040……;
N0040 M98 P1010;	N1050……;
N0050 ……;	N1060 M99;

主程序执行到 N0020 时转去执行子程序, M98 为子程序调用指令, P1010 指令为被调用的子程序名, L2 表示指令调用两次。子程序执行到 N1060 时跳出子程序, 继续向后执行主程序, M99 为子程序返回指令。

子程序可以由主程序调用, 已被调用的子程序还可以调用其他子程序, 这种方式称为子程序嵌套, 子程序嵌套可达四次, 其结构如图 1.10 所示。

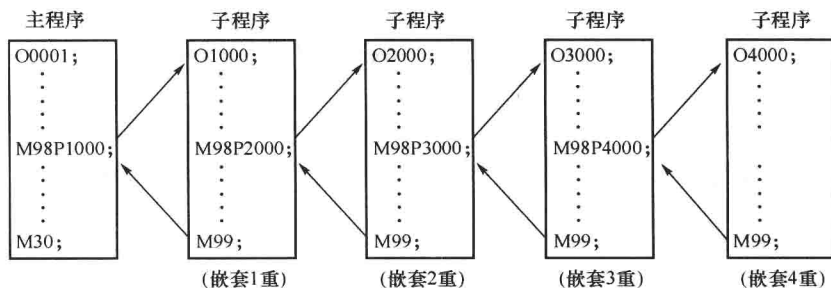


图 1.10 子程序嵌套

1.4 常用功能指令

常用的功能指令有准备功能(G 功能)和辅助功能(M 功能), 另外还有进给功能(F 功能), 主轴转速功能(S 功能)和刀具功能(T 功能)等。准备功能和辅助功能描述了程序段的各种操作和运动特征, 是程序段的主要组成部分。国际上已广泛使用 ISO 制定的 G 代码和 M 代码标准。我国也已制定了与 ISO 等效的 JB/T3208—1999 标准。

1.4.1 准备功能 G 指令

准备功能 G 指令又称 G 功能或 G 代码。该功能主要命令数控机床进行何种运动, 为控制系统的插补运算作准备。

G 代码由地址符 G 后跟两位数字组成, 从 G00 至 G99 共 100 种。随着数控机床功能的增加, G00 至 G99 已不够使用, 有些数控系统的地址符 G 后已经使用三位数字。从目前国内生产的数控系统和使用的国外数控系统所用功能来看, 只有一些常用的 G 指令按标准规定, 具有一定的灵活性。因此用户在编程时必须依据机床系统说明书, 不可张冠李戴。G 指令的具体用法将在以下章节中具体讲述。

表 1.2 为我国制定的 JB/T3208—1999 标准规定的 G 代码定义, 表序号(2)中的 a、c、…、k、i 各字母所对应的 G 代码称为模态代码(即时序有效代码)。它表示一旦被应用(如 a 组中的 G01)就一直有效, 且可省略不写, 直到出现同组(a 组)其他任一 G 代码(如 G00)时才失效。其他 c、d、f 等各组同理。序号(3)中有“*”号的 G 代码为非模态代码, 即只在本句有效, 下一程序段需要时必须重写。序号(4)中的“不指定”代码用作将来修订标准时, 有可能指定新的功能定义; “永不指定”代码, 表示即使将来修订标准时, 也不指定新的定义。