

JIGONG ZHONGXIN HONG CHENGXU
YINGYONG ANLI

加工中心宏程序 应用案例

张喜江◎编著



```
( D42 XI )
```

```
S300 M03
```

```
G00 G90 G54 X-44 Y-61 A
```

```
Z50
```

```
G01 Z-35 F200
```

```
G01 X88 A720
```

```
#1=0
```

```
WHILE [#1 LE 1] DO1
```

```
#11=#1*360
```

```
#12=#1*#1*1.75+44*#1
```

```
G01 X[88+#12] A[720+#11]
```

```
#1=#1+0.05
```

```
END1
```

```
G01 X632.5 A4860
```

```
Y-55
```

```
G01 X133.75 A1080.
```

```
#1=1
```



金盾出版社

加工中心宏程序应用案例

张喜江 编著

金盾出版社

内 容 提 要

本书分为宏程序介绍和应用案例两章。前者内容有:有关宏程序的定义、变量、宏程序函数、宏程序的分支与循环、FANUC 0i 常用系统变量的介绍、用户宏程序的调用、编写出好的宏程序的要求和条件;后者内容有:相似零件的加工案例、曲线曲面插补的加工案例、设置机床加工参数、定制固定循环案例、订制 G 代码案例、对刀与测量、捷径应用、四轴机床加工。书后附有常用代码功能。

本书可作为职业院校和培训机构相关专业的辅助教材,以及数控机床操作人员和数控工艺人员的参考用书。

图书在版编目(CIP)数据

加工中心宏程序应用案例/张喜江编著. — 北京:金盾出版社,2013.4
ISBN 978-7-5082-8085-1

I. ①加… II. ①张… III. ①加工中心—程序设计 IV. ①TG659

中国版本图书馆 CIP 数据核字(2013)第 017005 号

金盾出版社出版、总发行

北京太平路 5 号(地铁万寿路站往南)

邮政编码:100036 电话:68214039 83219215

传真:68276683 网址:www.jdcbs.cn

封面印刷:北京凌奇印刷有限责任公司

正文印刷:北京军迪印刷有限责任公司

装订:兴浩装订厂

各地新华书店经销

开本:705×1000 1/16 印张:10.5 字数:252 千字

2013 年 4 月第 1 版第 1 次印刷

印数:1~3 500 册 定价:26.00 元

(凡购买金盾出版社的图书,如有缺页、
倒页、脱页者,本社发行部负责调换)

前 言

当前我国制造业正处于高速发展时期,其中数控加工在制造业中占有重要地位,企业急需高素质的数控技能人才。数控程序的编写已经成为数控工人必备的知识,宏编程更是提高编程技能与操作技能不可或缺的工具。

随着计算机技术的发展,CAD/CAM 编程已经成为当前主流的编程方式,但是它并不能替代宏编程。宏编程作为手工编程的扩展,可以提供更灵活的编程方式,它可以使我们的编程工作变得非常简单、高效。随着个人工作经验的增长,宏编程会发挥更大的作用,产生更高的生产效率。

不同的数控系统会提供不同的宏程序编写格式,甚至不同的系统型号也会有所差异,但是在编程思路与技巧上是一致的。本书采用市场拥有率较高的 FANUC 0i 系统作为我们学习宏编程的工具。尽管学习宏程序会花费一些时间,但这是非常值得的。

书中的案例大部分来自生产实际,是编者多年工作经验的积累。本书是一本能帮助现场生产的很好的企业培训教材,既可以作为宏编程初学者的入门学习书,也可以作为数控编程人员的参考书。尽管书中的案例都经过多次调试,但在编写过程中难免出现各种纰漏,读者在实际加工应用前,一定要多次调试。

本书由石家庄市职业技术教育中心的老师编写,其中第一章第二节由韩开生编写,第一章第三节由袁伟才编写,第一章第四节由赵丽梅编写,其余章节由张喜江编写并做全书统稿。在编写过程中,得到了张喜江的数控启蒙老师,加工中心高级技师赵生才的指导和帮助,在此表示衷心的感谢。

由于编者水平有限,对于书中欠妥之处,欢迎读者指正。张喜江的电子邮箱是: zhangxijiang8@126.com。我们使用宏程序的目的是把工作变得轻松和充满乐趣,预祝读者工作愉快!

作 者

目 录

第一章 宏程序介绍	1
第一节 有关宏程序的定义	1
一、什么是宏程序	1
二、宏程序的种类	1
三、宏程序能解决什么问题	2
第二节 变量	3
一、变量的定义	3
二、变量的赋值	3
三、变量的种类	6
第三节 宏程序函数	11
一、算术函数	11
二、三角函数	13
三、四舍五入函数	14
四、辅助函数	15
五、比较函数	18
六、逻辑函数	18
第四节 宏程序的分支与循环	19
一、分支函数 IF	19
二、WHILE 循环	24
第五节 FANUC Oi 常用系统变量	38
一、用于数据设置的系统变量	38
二、用于模态数据的系统变量	46
三、用于 PLC 的系统变量	51
第六节 用户宏程序	52
一、子程序的调用	52
二、用户宏程序的调用	55
三、用户宏程序的模态调用	59
四、用户宏程序的保护与隐藏	63
第七节 编写出高质量宏程序的要求和条件	64
第二章 应用案例	65
第一节 相似零件的加工案例	65
一、模具底板	65

二、冲模型芯	68
三、钻模板	70
四、马达垫片	71
五、样板	73
六、孔的螺旋铣削	74
七、螺纹的铣削	76
第二节 曲线曲面插补的加工案例	78
一、插补椭圆	78
二、插补抛物线	79
三、正弦曲线插补	81
四、混合曲线插补	82
五、铣削公式曲线	83
六、端面螺纹的铣削	84
七、插补球面	85
八、插补正弦曲面	88
九、插补直纹面	90
第三节 设置机床加工参数	92
一、倒角	92
二、倒圆	96
三、综合练习	99
第四节 定制固定循环	103
一、钻孔循环	103
二、深孔排屑循环	105
三、深孔断屑循环	106
四、精镗孔循环	108
五、反镗孔循环	109
六、铣孔循环	111
七、螺旋铣孔循环	112
八、铣槽循环	115
第五节 定制 G 代码	118
一、定制圆周均布加工代码 G11	118
二、定制矩阵孔加工代码 G12	121
三、定制矩阵加工代码 G110	124
四、定制刀具切削寿命统计代码	128
第六节 对刀与测量	131
一、自动对刀、测量壁厚和宽度	131
二、探针对刀程序	134

三、探针测量程序 1	135
四、探针测量程序 2	136
第七节 捷径应用	137
一、程序 1——加工中心换刀	137
二、程序 2——交换工作台	138
第八节 四轴机床加工	141
一、零件十八——阀心	141
二、零件十九——槽轮	144
三、零件二十——偏心轴(孔加工)	150
四、零件二十一——箱体	152
附录	158
参考文献	160

第一章 宏程序介绍

宏编程作为手工编程的一部分,是手工编程的扩展和延伸,是对手工编程必要的补充。尽管 CAD/CAM 软件已经非常普及,但是它们并不能完全替代宏编程。宏编程使我们学会思考,能够更好地理解编程过程,并在编程中得到宝贵的训练,积累更多的编程经验。要想成为一名优秀的程序员,熟知技能是重要的前提条件,而基本技能则蕴涵在对手工编程特别是宏编程的理解中。

对于初学者,要学好宏程序,必须先熟知 G 代码和 M 代码、子程序、基本的加工经验和编程基础知识。

有关本章练习中的程序,可能仅仅是为了解释某个知识点,或为了某一项训练,并不代表是成熟的程序。

第一节 有关宏程序的定义

一、什么是宏程序

通常把含有宏语句的程序称之为宏程序,也有系统把参数化编程称之为编写宏程序。

宏编程就是一种手工编写零件加工程序的方法,它附加于标准 CNC 程序,使数控编程功能更强大、更灵活。从编程特点上说,具有计算机高级语言(例如 BASIC)编程的特征。

二、宏程序的种类

1. 第一类:普通宏程序

这是学习宏程序的初级阶段,通常以主程序的形式出现。普通宏程序只考虑当前加工的需要,通常较简单,一般只为解决某一个问题,不具有通用性。但通过此阶段的学习可以为编制用户宏程序打下良好的基础。下面是一个普通宏程序:

```
O1
#1=10
G90 G00 G54 X0 Y0
Z100
G01 Z-5 F80
Y-#1
G02 J#1
G00 Z100
M30
```

2. 第二类:用户宏程序

这是宏程序应用的成熟阶段。它以子程序的方式出现,使用时通过主程序调用,并可

以通过指定参数向子程序传递加工数据。

用户宏程序通常是完成某一类型的加工任务而设计,需要事先编好,并在各种情况下进行可行性验证,而后作为子程序保存。使用时可用 G65(或 G66)调用,通用性较好。用户宏程序是用户知识、技巧、经验的积累和总结。

用户宏程序的特点是:短小,精练,高效。通俗地说,就是小程序解决大问题。

下面是一个用户宏程序的调用:

O1 (主程序)

G90 G00 G54 X0 Y0

G65 P1001 A5 J80

M30

O1001 (用户宏程序)

G01 Y-#1 F#5

G02 J#1

M99

三、宏程序能解决什么问题

宏程序之所以值得学习,是因为它可以帮助我们解决某类问题,可以简化我们的工作。下面是经常用到宏程序的一些地方(但并不是全部)。

1. 相似零件的加工

主要是完成零件某一部位的重复加工,或有规律地重复某一个动作。

2. 非标轨迹插补

也称曲线曲面的插补加工,是根据给定的数学公式、数学模型等已知条件,使用 G01 或 G02 来完成曲线、曲面的插补。

3. 设置机床加工参数(刀具参数、坐标参数)

把工件坐标系、刀具长度补偿、刀具半径补偿等一些参数通过特定的宏程序语句写在程序中。系统在执行这些宏语句后,根据提供的信息填写到对应的偏置寄存器中。

4. 定制固定循环

根据自己的特定加工领域,以用户宏程序的形式编写一些自己常用的加工循环,例如铣槽循环、钻孔循环、镗孔循环等。

5. 定制 G 代码

例如 G12, G13, G110 等,是把一些经常用到的计算方法、加工经验或经常调用的用户宏程序等设置到一个特定 G 代码中,以简化编程、提高效率。

6. 检测与测量

包括机床工作状态的检测、工件加工尺寸精度的测量、自动建立工件坐标系、机内对刀等。

7. 捷径应用

如加工中心的换刀程序、交换工作台等。

8. 多轴加工

提示:不同的数控系统有可能仅具有上述应用中的部分内容。

第二节 变 量

变量是宏程序最基本的特征,也是宏程序区别于普通程序的标志。

一、变量的定义

变量是一个数学等价物,是与常数相对应的。在计算机技术中,一个变量对应一个存储器。在宏程序中,变量只能存储数字。

可以用常见的小型科学计算器来解释变量的概念。即使是最便宜的计算器,也有一个临时存储单元,对应的按键是 M 键。我们计算的中间数据,可以存放到里面,供后面的计算使用,这个存储单元本身就是一个变量(计算器说明书上称之为存储器)。

变量名字本身意味着它里面的数据在计算过程中是随时变化的。

在 FANUC 系统中,用符号“#”和一个数字的组合表示一个变量,例如:

#3 表示 3 号变量, #13 表示 13 号变量, #123 表示 123 号变量。

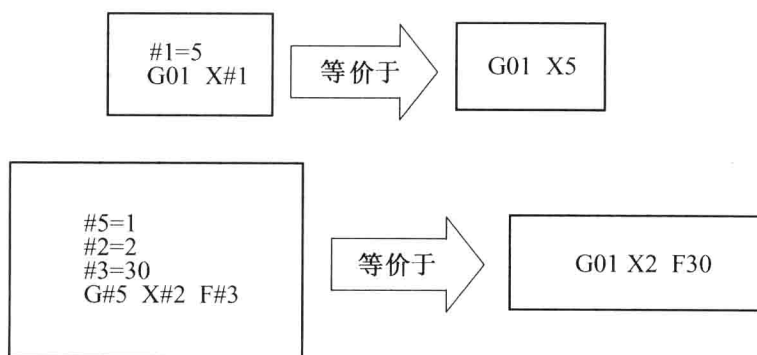
二、变量的赋值

在计算机高级语言编程中,变量的赋值也称之为变量的声明。变量在使用前,必须先往里面存入数据,存入数据的过程就是变量的赋值。例如:

#1=15 表示把数字 15 存入变量 #1, #12=1.05 表示把数字 1.05 存入变量 #12。

提示:在这里符号“=”不是等号,是赋值号。

变量赋值后,就可以使用了。例如:



1. 练习一(图 1-1)

(1)工艺条件 工件零点在工件上表面 R15mm 的圆心点,刀具采用 $\phi 16\text{mm}$ 高速钢铣刀。

(2)参考程序

O1

#1=7 (注释:R15-r8=7)

G90 G00 G54 X0 Y-22

M03 S450

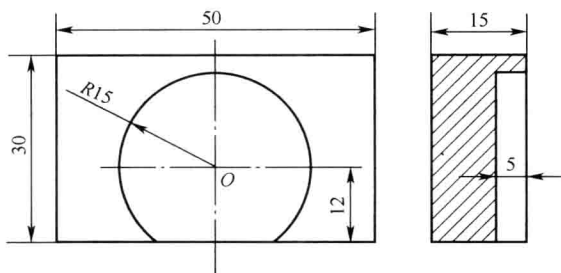


图 1-1 练习一加工图

```

Z100
Z5
G01 Z-5 F60
Y-#1
G02 J#1
G00 Z100
M30

```

提示:在程序中利用变量#1 来代表铣削半径。当刀具磨损后,直接修改变量值就可以实现精加工。使用变量后,利用 R15 圆弧的缺口进刀,程序就变得非常简单。

2. 练习二(图 1-2)

在单件生产中,可以利用修改变量的值实现分层加工。

(1)工艺条件 工件零点在工件上表面 $\phi 40\text{mm}$ 圆心点,刀具采用 $\phi 30\text{mm}$ 高速钢铣刀。

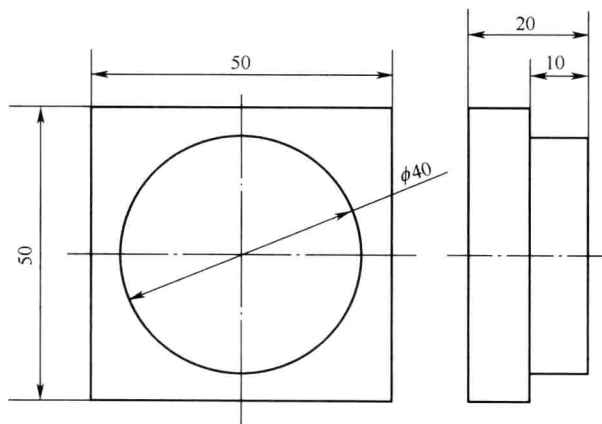


图 1-2 练习二加工图

(2)参考程序

```

O2
#1=-2
#2=35

```

```

G90 G00 G54 X0 Y50
M03 S240
Z100
Z5
Z#1
Y#2 F30
G02 J-#2
G00 Z100
M30

```

提示:程序每执行一遍,就修改程序中#1 的值为-4,-6,-8,-10,就可以完成零件的分层加工。

3. 练习三(图 1-3)

(1)工艺条件 工件零点在工件上表面中心点,刀具采用 $\phi 16\text{mm}$ 高速钢铣刀。编程轨迹如图 1-4 所示。

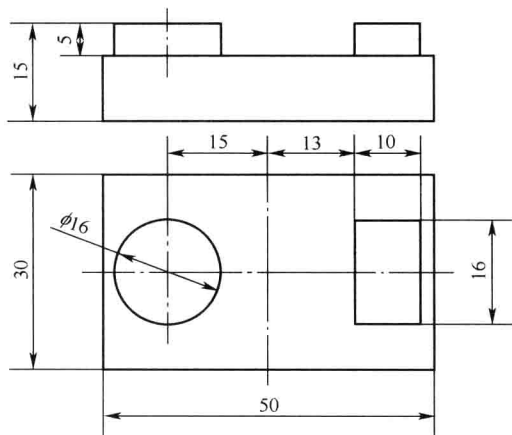


图 1-3 练习三加工图

(2)参考程序

```

O1
#1=-5 (依次设为-2.5,-5,实现分层加工)
G40 G17
G90 G00 G54 X-20 Y25
M03 S400
G43 H1 Z100
Z5
G01 Z#1 F100
G41 D1 Y8
X-15
G02 J-8

```



```

#2=20
#3=30
#1=#0 (清除变量#1中的数值,#1将变成空变量)
#2=#1 (清除变量#2中的数值,因为#1已经变成了一个空变量)
M00
N10 G90 G00 G54 X#1 Y#3
M03 S500
Z100
N20 G01 Z#2 F80
G00 Z100
M30

```

提示:在 N10 程序段,系统仅执行 G90 G00 G54 Y30,而忽略 X;在 N20 程序段,系统仅执行 G01 F80,而忽略 Z。

局部变量和全局变量经常使初学者产生混淆,下面我们将分别对局部变量和全局变量进行说明。

2. 局部变量

局部变量只在当前程序有效。变量在主程序中定义,那就只在主程序中有效;如果在子程序中定义,那就只在子程序中有效。在主程序中定义的局部变量不能被带到子程序中,同样在子程序中定义的局部变量也不能被带入到主程序中或其他的子程序中。

在 FANUC 系统中只定义 33 个局部变量,分别是 #1, #2, #3~#33。

当程序执行结束(M30, M02),或遇到复位操作时,局部变量将被清空。

练习五。阅读下面的程序,看看局部变量有什么变化。

```

O1
#1=15
#2=120
G90 G00 G54 X0 Y0
M03 S500
Z100
Z5
G01 Z-5 F80
N10 G01 X#1 Y#2
M98 P101
N30 G01 X#1 Y#2
M98 P102
N50 G01 X#1 Y#2
G00 Z100
M30

```

```

O101
#1=30
#2=200
N20 G01 X#1 Y#2
M99

```

```

O102
#1=50
#2=60
N40 G01 X#1 Y#2
M99

```

试试看:在执行程序 O1 的过程中,

程序段 N10 G01 X#1 Y#2;中#1 和#2 的值分别是____,____;

程序段 N20 G01 X#1 Y#2;中#1 和#2 的值分别是____,____;

程序段 N30 G01 X#1 Y#2;中#1 和#2 的值分别是____,____;

程序段 N40 G01 X#1 Y#2;中#1 和#2 的值分别是____,____;

程序段 N50 G01 X#1 Y#2;中#1 和#2 的值分别是____,____。

答案:在 N10,N30,N50 中#1 的值是 15, #2 的值是 120;在 N20 中#1 的值是 30, #2 的值是 200;在 N40 中#1 的值是 50, #2 的值是 60。

3. 全局变量

全局变量一旦定义,将以模态的形式存在,即使程序执行完毕,全局变量依然有效。当然复位操作后,全局变量也有效。

全局变量分为两个范围段: #100~#199, #500~#599。

当数控机床断电后,变量 #100~#199 中的数值就会丢失(清空),而变量 #500~#599 中存储的数值则不会丢失。当我们需要长期保存一些数据时,我们可以把这些数据存放到变量 #500~#599 中。

(1)练习六 阅读下面的程序,看全局变量有什么变化。

O11

#101=15

#102=120

G90 G00 G54 X0 Y0

M03 S500

Z100

Z5

G01 Z-5 F80

N10 G01 X#101 Y#102

M98 P101

N30 G01 X#101 Y#102

M98 P102

N50 G01 X#101 Y#102

G00 Z100

M30

O101

#101=30

#102=200

N20 G01 X#101 F#102

M99

O102

#101=50

#102=60

N40 G01 X#101 F#102

M99

试试看:在执行程序 O11 的过程中,

程序段 N10 G01 X#101 Y#102;中#101 和#102 的值分别是____,____;

程序段 N20 G01 X#101 Y#102;中#101 和#102 的值分别是____,____;

程序段 N30 G01 X#101 Y#102;中#101 和#102 的值分别是____,____;

程序段 N40 G01 X#101 Y#102;中#101 和#102 的值分别是____,____;

程序段 N50 G01 X#101 Y#102;中#101 和#102 的值分别是____,____。

答案:在 N10 中#101 的值是 15, #102 的值是 120;在 N20,N30 中#101 的值是 30, #102 的值是 200;在 N40,N50 中#101 的值是 50, #102 的值是 60。

(2)练习七

O17

#111=15

#11=30

G90 G00 G54 X0 Y0

M03 S500

Z100

N10 G01 Z#111 F#11

G00 Z100

M30

O18

#11=50

G90 G00 G54 X0 Y0

N20 M03 S#123

Z100

N30 G01 Z#111 F#11

G00 Z100

M30

讨论:开机并返参后,首先调出程序 O17 并执行,而后我们调出程序 O18 继续执行。当执行到 N20 语句时,机床如何执行? 当执行到 N30 语句时,机床如何执行?

答案:在 N20 中,执行 M03(由于 #123 没有被赋值,即空变量,所以 S#123 被省略);在 N30 中,执行 G01 Z15 F50。

(3)练习八

O17

G90 G00 G54 X0 Y0

M03 S500

Z100

Z5

G81 Z-5 F80

G00 Z100

#501=#501+1

M30

提示:通过 #501 计算程序 O17 运行的次数;在批量加工零件时,可用来统计工件的加工数量。

4. 系统变量

系统变量不同于其他的变量,它们在宏程序中非常重要,而且自成体系。系统变量区别于其他变量的特征有两点:一是系统变量的编号从 #1000 开始,直到 5 位数(例如 #12000),数量和细分种类非常多;二是系统变量不能显示在屏幕上。

练习九。

O1

#1 = #1003

#110 = #5021

M30

提示:系统变量的值,可以被转存到局部变量或全局变量中后,通过查看局部变量或全局变量得知系统变量的值。

系统变量的编号已经被 FANUC 系统固定,并代表不同的含义,用户不可以改变。要想知道某个系统变量的含义只有查阅系统参考手册。

(1) 系统变量的用途

- ①和 PLC 系统双向传递参数;
- ②检测当前工件的坐标位置,包括机床坐标位置、工件坐标位置等;
- ③检测刀具补偿参数,包括刀具半径补偿和刀具长度补偿;
- ④检测每组 G 代码的当前状态;
- ⑤给工件坐标系赋值;
- ⑥给刀具补偿参数赋值;
- ⑦参数设定;

.....

总之,系统变量对于数控机床至关重要。对于每个控制系统来说,都有很多的系统变量。一个程序员不可能记住所有的系统变量;需要时,通过查阅手册很容易得到。

(2) 系统变量的分类

1) 可读写系统变量:用户可以通过一段程序或 MDI 来改变这类变量的数据。这类变量也可被系统读出,并由系统保存其变量值。

练习十。

O9

N10 #5221 = 20 (给工件坐标系 G54 的 X 坐标赋值)

N11 #5222 = 50 (给工件坐标系 G54 的 Y 坐标赋值)

N12 M00

N13 #1 = #5221 (读出坐标系 G54 的 X 坐标值到 #1)

N14 #2 = #5222 (读出坐标系 G54 的 Y 坐标值到 #2)

M30

提示:N10 语句为 #5221 写入数据 20,N13 语句把 #5221 中的数据读出。

2) 只读系统变量。例如, #1000 至之后的数十个变量通常对应数控机床上的某个行程开关,它们只能根据开关的闭合状态显示 1 或 0,不允许用户赋值。

①练习十一。

O9002 (加工中心交换工作台程序)

N10 #1 = #1001 (#1 工作台行程开关)

N11 #2 = #1002 (#2 工作台行程开关)

N12 IF [#1 EQ 1] GOTO20