

数控编程指令功能扩展的宏实现

詹华西 李艳华

(武汉职业技术学院, 武汉 430073)

[摘要]: 基于 HNC 系统的宏编程处理经验, 以一个固定循环指令实现阵列孔加工为例, 介绍了数控编程指令功能扩展的二次开发技术。

关键词: 宏编程; 阵列钻孔; 二次开发

the Macro Technology to Extend NC Programming Instruction Function

ZHAN Hua-xi, LI Yan-hua

(Wuhan Institute of Technology, Wuhan 430073, China)

[Abstract]: Based on the processing experience of macro programming in HNC system, this article introduced redevelopment technology of NC programming instruction function. It even gives example of a fixed instruction to machine arrayed drills.

Key Words: Macro programming; Machining arrayed drills; Secondary development

0. 引言

由于数控机床进行插补控制的主要是直线和圆弧, 系统能提供的直接用于加工的程序指令非常有限, 因此寻求合理的算法, 利用基本指令来扩展系统的编程指令功能, 使得用户仅用一个指令行即可实现相对复杂的加工功能, 这一直是系统开发人员的研究课题, 也是加工编程人员寻求的目标。车削循环、钻镗基本循环等都是数控系统开发人员对指令系统扩展的典型示例, 但不同的系统在这方面开发的程度是有差异的。比如 SIEMENS 系统已经具有直接用于阵列孔加工、规则形状的挖槽循环等扩展指令, 而 HNC、FANUC 系统目前还没有面向普通用户提供。对于非开放式的数控系统, 这种指令功能扩展只能依赖于系统生产厂家, 对于 HNC 这类基于 PC-NC 的开放式数控系统, 只要熟知宏编程处理技术, 普通用户即可自行开发定制。

1. 宏扩展编程的技术基础

HNC 系统在调用宏子程序的同时可进行参数传值^[1], 即将调用行所跟的主调参数 A~Z 各字段的内容拷贝到宏执行的子程序内为局部变量 #0~#25 预设的存贮空间中; 将指令的初始平面 Z 的模态值拷贝到 #26 中; 同时还拷贝当前通道九个轴 (XYZ/ABC/UVW) 的绝对位置坐标到宏子程序的局部变量 #30~#38 中; 并且还可以通过系统变量 #1120~#1145 来访问 A~Z 26 个地址字的模态数据, 通过系统变量 #1150~#1169 来访问 0~19 组 G 代码的模态值。

在 HNC 系统中, 对于每个局部变量, 都可用

系统宏 AR[] 来判别该变量是否被定义、是被定义为增量坐标还是绝对坐标方式。

调用格式为: AR [#变量号]

返回值:

0: 表示该变量没有被定义;

90: 表示该变量被定义为绝对方式 G90;

91: 表示该变量被定义为增量方式 G91。

例如: IF [AR[#23] EQ 0]

```
IF [AR[#1143] EQ 91]
```

```
#23=0
```

```
ELSE
```

```
#23=#1143
```

```
ENDIF
```

```
ENDIF
```

表达的意思是: 如果指令行的 X (对应 #23) 参数未指定, 且系统变量 #1143 (X 坐标以前的模态) 为增量方式, X 就取增量 0 值; 若 #1143 为绝对方式, X 就取以前的模态值。

HNC 作为一个开放式数控系统, 其用于钻镗基本循环 G73~G89 的宏扩展程序的源码已面向广大用户公开, 它就是利用宏子程序参数传值的处理方法, 将 G 指令定制的多个参数, 传值到宏子程序中, 由子程序对各参数数据进行整理后依据相应的加工工艺, 按一定的算法通过基本指令来定制动作实现加工。普通用户亦可参照这一思路进行编程指令的二次开发。

2. 阵列钻孔扩展指令的开发定制

图 1 所示为一圆形阵列孔位关系分布图。和矩形阵列孔加工一样, 在 HNC 中目前还无法由一个指令行来编程实现, 但参照 G73~G89 钻镗基本

循环的宏子程序编制方法，可自行开发定制。如果以 G75 为圆形阵列钻孔的指令，其定制格式为：

$\left(\begin{matrix} G90 \\ G91 \end{matrix} \right) \left(\begin{matrix} G98 \\ G99 \end{matrix} \right) G75 \text{ X_Y_Z_R_A_B_C_D_E_} \dots \text{ F_}$

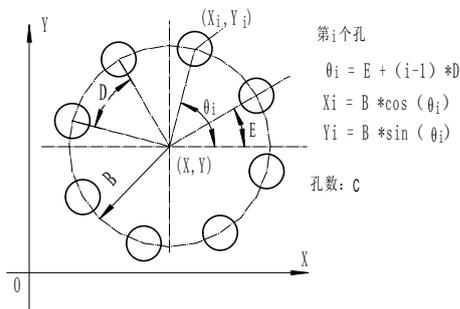


图 1 圆形阵列孔位关系

其中，X、Y：阵列中心的 X、Y 坐标

Z、R：孔底和 R 面的 Z 坐标

A：钻孔方式循环号 B：孔所在圆周半径

C：孔数 D：孔间角度

E：起始孔角度（与+X 的夹角，逆 + 顺 -）

A、B、C、E、G 为阵列钻孔增加的参数，如果循环号 A 调用 G73~G89 钻孔方式需要 I、J、K、Q、P 等参数，其含义按 G73~G89 中对应的定义添加。

则 G75 宏子程序的内容可参考编制如下：

```
%0075
IF [AR[#23] EQ 0] ; 如果没定义X
  IF [AR[#1143] EQ 91] ; 如果X以前的模式为G91
    #23=0 ; X=0
  ELSE ; 否则
    #23=#1143 ; X取以前的模式值
  ENDIF
ENDIF
IF [AR[#24] EQ 0] ; 如果没定义Y时的处理
  IF [AR[#1144] EQ 91]
    #24=0
  ELSE
    #24=#1144
  ENDIF
ENDIF
IF [AR[#17] EQ 0] ; 如果没有定义R
  #17=#1137 ; 取当前R的模式值
ENDIF
IF [AR[#25] EQ 0] ; 如果没有定义Z
  #25=#1145 ; 取当前Z的模式值
ENDIF
IF [AR[#0] EQ 0] ; 如果没有定义A
  #0=#1120 ; 取当前A的模式值
ENDIF
..... ; 同上处理B、C、D、E、P、Q、I、J、K 参数
IF [AR[#3] EQ 0] AND [#2 GT 1]
```

```
M-99 ; 如果没定义角度 D 且孔数>1 则返回出错信息
ENDIF
N10 G91 ; 切换到增量编程G91模式
IF AR[#23] EQ 90 ; 如果X值是绝对编程G90
  #23=#23-#30 ; 则按G91的算法换算X值
ENDIF ; #30为调用前X的绝对坐标
IF AR[#24] EQ 90 ; Y值的数据转换
  #24=#24-#31
ENDIF
IF AR[#17] EQ 90 ; R值的数据转换
  #17=#17-#32
ELSE
  IF AR[#26] NE 0 ; 如果有初始高度面
    #17=#17+#26-#32 ; 按初始高度换算
  ENDIF
ENDIF
IF AR[#25] EQ 90 ; 孔底Z值的换算
  #25=#25-#32-#17
ENDIF
IF [#25 GE 0] AND [#0 NE 87]
  M-99;如果G87之外的Z的增量为正，则返回出错信息
ENDIF
G00X[#23]Y[#24] ; 定位到阵列中心
#39=PI/180
#40=1 ; 孔数循环初值
#44=0 ; 起始孔 X 增量初值
#45=0 ; 起始孔 Y 增量初值
#46=#17 ; 另存R值到局部变量#46中
#47=#1145 ; 备份Z的模式
#48=#1137 ; 备份 R 的模式
WHILE #40 LE #2 ; 循环钻镗孔开始
#41=[#4+[#40-1]*#3]*#39 ; 第i个孔的角度（弧度）
#42=#1*COS[#41] ; 孔i相对阵列中心的X坐标
#43=#1*SIN[#41] ; 孔i相对阵列中心的Y坐标
G91G[#0]X[#42-#44]Y[#43-#45]Z[#25]
--R[#46]I[#8]J[#9]K[#10]P[#15]Q[#16]
; 调用循环号A所对应的G指令加工孔
#44=#42 ; 另存孔i中心相对阵列中心的X、Y坐标
#45=#43 ; 为下一孔i+1中心提供增量起点坐标数据
IF #1165 EQ 99 ; 如果系统变量第15组G代码模式值为G99
  #46=0 ; 将R后续增量清零
ENDIF
#40=#40+1 ; 孔数循环递增，加工下一个孔
ENDW
G00X[-#44]Y[-#45] ; 返回到阵列中心
#1137=#48 ; 恢复R的模式
#1145=#47 ; 恢复Z的模式
M99
```

该阵列钻孔加工子程序主要是计算各孔中心的坐标，具体钻孔还是通过再次调用系统定义的钻镗固定循环指令来实现的。

3. 扩展指令的加工应用

上述扩展指令宏子程序%0075 编制完成后，在 HNC 系统中，应将其内容添加存储到系统 BIN 文件夹下的 O0000 文件内，则以后用户即可象使用 G73~G89 固定循环指令那样直接使用 G75 指令功能来作圆形阵列孔的加工编程。

作者：詹华西 副教授 通信地址：武汉关山一路 463 号武汉职业技术学院机电工程学院
邮编：430073 E-Mail: zhanhuaxi@163.com
联系电话：13607121576

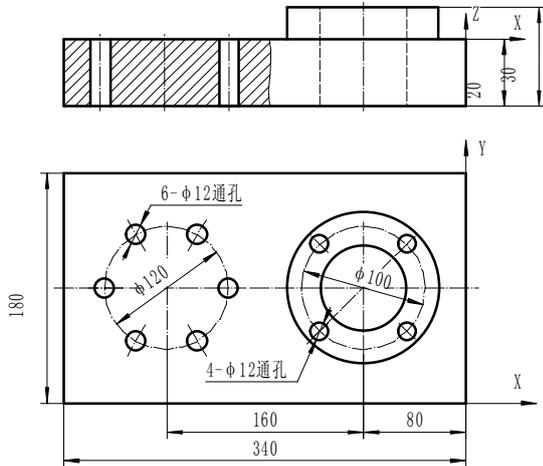


图 2 阵列孔加工示例

例如，对于图 2 所示阵列孔加工，若工件零点设在图示右下角，可编程如下：

```
O0001
G54G90G0X-240.0Y90.0S500M3
G43Z20.0H1M8
G99G75Z-25.0R5.0A81B60C6D60E0F60
X-80.0R15.0A83B50C4D90Q-5K3E45
G80M5
G91G28Z0M9
M30
```

4. 结束语

开放式 HNC 系统提供给用户一个以宏编程技术为基础，对编程指令系统进行扩展开发的平台。新指令功能的开发需要合理定制参数、熟知参数传值调用的规则、具有缜密可靠的技术算法、全面完善的错误处理对策等等。可利用这个平台按需要定制参数，自行开发一些如阵列孔加工、矩形轮廓铣削及挖槽、椭圆铣削及挖槽、凹凸球面加工等相对复杂而实用性较强的通用指令，让普通用户不需要使用自动编程软件即可简单快捷的编出加工程序来。

参考文献：

[1] 华中数控股份有限公司. 数控铣床编程说明书. 武汉：华中数控股份有限公司，2004