

利用宏指令实现数控程序的防误功能

■ 西安航空发动机（集团）有限公司（陕西 710021） 冯永星

摘要：利用FANUC数控系统宏指令的各类功能，编制宏程序对手工输入的各类外部数据进行校验，使数控加工程序具有判断及纠错功能，最大限度地减少人为因素造成的废品损失。



数控机床作为高精度、高稳定性设备，在加工中人为出错是其造成废品的关键因素。尤其在多品种、小批量生产车间，频繁地更换零件导致了大量的辅助准备工作，人为出错造成废品的概率更是大大提高。因此如何防范由于人为操作失误或疏忽造成的损失就显得格外重要。

本文将介绍一种基于FANUC Oi TC控制系统车床程序的编制方法。采用该方法编制出的数控程序可将各类外部输入数据读取至宏程序各类变量中，并对其进行分析与判断，使数控程序具有防误及报警功能，从源头上最大限度地避免由于人为失误而造成的损失。

1. 数控加工中常见的人为失误

(1) 输入错误。在操作中常

见的输入内容包括：几何形状、磨损值、刀具半径及假想刀尖位置等。这些内容是数控系统最关键的外部数据，控制器所有的运算将基于这些数据，任何一项输入不正确便会导致错误运算，从而造成零件的超差甚至报废。

(2) 测量误差。在加工零件时，当更换新的刀片后必须重新对刀。当采用试切法对刀时，工人必须测量试切后零件的实际尺寸以输入机床寄存器，在此过程中若零件的实际尺寸测量不准确，将极易造成零件超差或报废。当加工要求比较严格的尺寸（通常指公差小于0.05mm的尺寸）时，程序中一般都有半精加工测量点，工人会根据测量的实际值来调整精加工的余量。若此时测量的数值有误，则将会直接导致零件报废。

(3) 疏忽大意。该类包括：刀具错误的安装、使用，实际使用的刀具或刀具信息与程序制定值不一致等。该类问题往往是由于操作人员不熟练或粗心大意造成的。因其特殊性，某些问题需要加强对工人的培训，而很大一

部分则可以在程序中对其校对和预判。

2. 程序的思路及流程

(1) 思路。①输入错误信息的防止。数控系统将所有的输入信息存储在寄存器中且可读取，而程序员在编写程序时会对其程序所使用的刀具等信息有一个预先的制定。因此，在编制程序时就可将这些制定值纳入其中，在加工时通过宏程序读取当前实际信息并与预先的定制数据相对比，当所有信息一致时方可运行程序，否则通过宏程序给予报警和相应的提示。②测量误差的防止。该类误差往往是在对刀的过程中产生的，因此可在首件零件调整加工合格后，将加工该零件的对刀参数换算成标准数值（即根据零件的实际尺寸换算，具体方法见下文）作为程序的预设信息存储在公共变量中，通过宏程序与更换刀片后重新对刀的信息相比较，当其误差值小于给定值时则继续运行程序，否则通过宏程序给予报警并提示。

程序预设值的换算方式：以图1所示零件为例，现要加工图中

$\phi 86.1_{-0.1}^{0}$ mm的直径与 $42_{-0.15}^{0}$ mm的端面。假设首件的实际直径为 $\phi 86.08$ mm,总长为41.98mm,此时该零件是合格的,但该零件的对刀参数不能作为预设值。因为该参数时零件的尺寸接近极限,易造成超差,故需经换算。假设加工该零件时寄存器中X、Z向刀补分别为-400.000与-600.000,在忽略磨损值与坐标系偏移时预设值应为:-400.030和-600.055。当然,在实际加工首件时工人一般都需要在磨损或工件移中进行尺寸调整,若将这些因素考虑在内则可按下列公式计算

$$X=X_1+T_x-W_x+\phi_1-\phi_2$$

$$Z=Z_1+T_z-W_z+L_1-L_2$$

公式中各参数的意义如下:

X为X轴预设值;Z为Z轴预设值;
 X_1 为X轴对刀值(也称作X轴几何形状补偿值,一般为负数);
 T_x 为X向磨损值; Z_1 为Z轴对刀值(也称作Z轴几何形状补偿值,一般为负数);
 T_z 为Z向磨损值;
 W_x 为X向工件偏移值; W_z 为Z向工件偏移值;
 ϕ_1 为理论直径中值尺寸; ϕ_2 为零件实际直径;
 L_1 为理论长度中值尺寸; L_2 为零件实际长度。

当程序使用的刀具数量较少时,可手工计算。否则可编制如下的宏程序由控制器进行计算,并直接将运算结果赋值给相应的

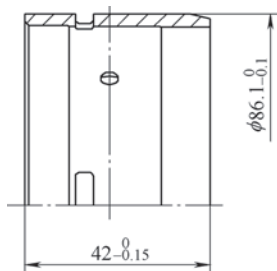


图1 转子空气封严

公共变量中,以供后文的防误子程序使用。以图1为例,在采用T01号刀、01号刀补时程序可为:

```
%
O0001
#1=86.08
#2=41.98
#3=86.05
#4=41.925
#10=1
#11=#[2700+#10]
#12=#[2800+#10]
#13=#[2000+#10]
#14=#[2100+#10]
#15=#2501
#16=#2601
#[500+#10]=#11+#13-#15+#3-#1
#[600+#10]=#12+#14-#16+#4-#2
M30
%
```

使用该程序时只需根据实际尺寸对#1~#4、#10五个变量赋值,其余由宏程序自动从控制器中读取并运算、赋值。该5个变量的意义依次为:X向测量值、Z向测量值、X向理论中值、Z向理论中值以及加工该尺寸所用刀具的刀补号。如前文所述的在不考虑磨损与工件移时运算结果为-400.030与-600.055并将该数据赋值给#501和#601。本例中所涉及各类变量均在下文有介绍。

(2) 流程。在加工出合格的首件前(试制或调试阶段)程序的主要工作流程为:使用宏程序读取系统参数,与预设值进行比较分析,仅当满足所有条件方运行,否则停止运行并给予提示性报警。

首件合格后(批产阶段)

程序的主要任务为误差分析与警报,并依次按照:对刀→误差分析→正常加工→对刀的顺序重复循环。详细的流程图如图2所示。

3. 宏程序相关变量的介绍

(1) 变量的分类及功能。变量根据变量号可分成空变量、局部变量、公共变量与系统变量4种类型,因这4类变量在本文中均有用到,特加以简单说明,具体如表1所示。

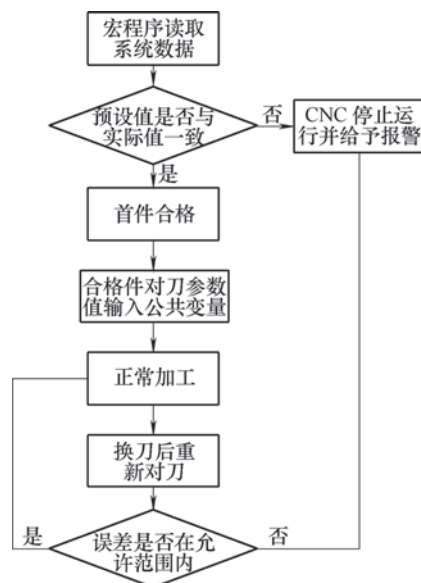


图2 工作流程图

表1 变量的分类

变量号	变量类型	功能
#0	空变量	该变量总是空,无法被赋值
#1~#33	局部变量	只能用在宏程序中存储数据,例如运算结果。当断电时局部变量被初始化为空。调用宏程序时,自变量对局部变量赋值
#100~#199 #500~#999	公共变量	在不同的宏程序中意义相同,当断电时,变量#100~#199初始化为空,变量#500~#999的数据保存,即使断电也不丢失
#1000~	系统变量	用于读写CNC运行时的各种数据,例如刀具的当前位置和补偿值

(2) 刀具补偿存储区的系统变量。该组变量用于存储对刀信息以及补偿信息，本文宏程序所调用的参数主要来自该系统变量。如表2中所示，控制系统X、Y、Z3个坐标轴的补偿值以及刀尖圆弧半径、假想刀尖位置（即下文所说的刀具类型）均在系统变量#2001~#2499中存储。以当前刀补为1号刀举例：X轴的对刀

值存储于#2701参数，磨损值存储于#2001参数；Z轴的对刀值存储于#2801参数，磨损值存储于#2101参数；刀尖半径的对刀值存储于#2901参数，磨损值存储于#2201参数；刀具类型存储于#2301参数。其他刀位的信息可按表2类推，表中的Y轴补偿值因在大部分车床中未使用，故不做叙述。

表2 刀具补偿存储区的系统变量

补偿号	X轴补偿值		Z轴补偿值		刀尖半径补偿值		假想刀尖位置T	Y轴补偿值	
	磨损	几何形状	磨损	几何形状	磨损	几何形状		磨损	几何形状
1	#2001	#2701	#2101	#2801	#2201	#2901	#2301	#2401	#2451
2	#2002	#2702	#2102	#2802	#2202	#2902	#2302	#2402	#2452
3	#2003	#2703	#2103	#2803	#2203	#2903	#2303	#2403	#2453
...	...	...	...	...	...	...	...	...	...
49	#2049	#2749	#2149	#2849	#2249	#2949	#2349	#2449	#2499
...	...	...	...	...	...	...	...	...	...
64	#2064		#2164		#2264	#2964	#2364		

(3) 宏程序报警的系统变量（#3000）。当变量#3000被赋予的值为0~200时，CNC停止运行且报警，显示不超过26个字符的报警信息。CNC屏幕上显示报警号和报警信息，其中报警号为变量#3000的值加上3000。

例如：#3000=1（TOOL NOT FOUND）

报警屏幕上显示“3001 TOOL NOT FOUND”（刀具未找到）。

(4) 工件坐标系偏移。工件坐标系偏移量可以被读取，其数值也可以通过输入一个数据来改变，具体参数如表3所示。

4. 实例

本文以FANUC车床各类对刀数据（X向刀补、Z向刀补、圆弧

半径、刀具类型）为例，说明宏程序在车床上如何实现判断、纠错以达到防误的目的。以加工图1所示的零件为例，所使用的刀具信息如表4所列。

X、Z方向的最大误差取决于所加工尺寸的公差带以及机夹刀片的重复定位精度，也可予以适当的压缩以获得更良好的尺寸稳定性。当有多把刀具时，根据实际需要，加工不同位置的刀具可根据当前情况来分别确定最大允许误差值。

(1) 主程序。上述所列各类预设值需要在主程序中被正确赋值，然后方可调用防误子程序，进行数据的判断。主程序具体内容及解释如下：

%

表4 预设信息

刀尖半径(#13)	刀具类型(#14)	X向最大误差(#11)	Z向最大误差(#12)
R0.4mm	3	0.04mm	0.06mm

O0001
N10 T0101 （刀位号，此处为01号刀，01号刀补）
N15 #11=0.04 （预设的X向最大误差0.04mm）
N20 #12=0.06 （预设的Z向最大误差0.06mm）
N25 #13=0.4 （编制程序时使用的刀具圆弧半径R0.4mm）
N30 #14=3 （编制程序时使用的刀具类型，此处为3号类型）

N35 M98P6001 （呼叫O6001防误子程序）
N40 ... （程序主体起始）
N45 ... （程序主体结束）
N50 M30 （程序结束）
%

N15~N30行为赋值阶段，也就是将各类外部预设信息及刀具信息赋予局部变量。

N35行为防误阶段，在该阶段主程序将调用O6001号子程序对各类数据进行分析与判断，以达到防误目的。O6001号子程序将在下文详细介绍。

N40~N45行为正常加工阶段，各类信息完全符合要求时方能进行到该阶段。

(2) 子程序。本文中的各类防误判断与报警信息均是由如下的O6001号子程序来完成，该子程序可作为独立的内容在任何刀位的程序段中被调用。当然必须在主程序中对#11~#14参数正确赋值，下面将对该子程序进行详细的解释：

%
O6001
N100 #10=[[#4120/100]-FIX[#4120/100]]*100
N105 #1=#[2700+#10]

```

N110 #2=#[2800+#10]
N115 #3=#[500+#10]
N120 #4=#[600+#10]
N125 #101=ABS[#1-#3]
N130 #102=ABS[#2-#4]
N135 #701=#[2900+#10]
N140 #801=#[2300+#10]
N145 IF[#701NE#13]GOTO3001
N150 IF[#801NE#14]GOTO3002
N155 IF[#500+#10]EQ#0]
GOTO3003
N160 IF[#600+#10]EQ#0]
GOTO3004
N165 IF[#101GT#11]GOTO3005
N170 IF[#102GT#12]GOTO3006
N175 M99
N3001#3000=1(CHECK TOOL
RADIUS)
N195 M99
N3002#3000=2(CHECK TOOL
TYPE)
N205 M99
N3003#3000=3(CHECK X-AXIS
PRESET)
N220 M99
N3004#3000=4(CHECK Z-AXIS
PRESET)
N230 M99
N3005#3000=5(CHECK X-AXIS
OFFSET)
N240 M99
N3006#3000=6(CHECK Z-AXIS
OFFSET)
N250 M99
%
```

N100行为读取当前刀位刀具所调用的实际刀补号。N105~N140行为整个子程序的赋值与运算阶段，在该阶段宏程序从系统参数中读取各类实际信息。以表4所列举的各类信息为例，当当前

刀具所调用的刀补号为1时，#1~#4所读取的信息分别为：X轴实际对刀值、Z轴实际对刀值、X轴预设值以及Z轴预设值；#101与#102分别为X与Z轴对刀误差；#701与#801则为当前实际刀尖半径和刀具类型。

N145~N170行为子程序的判断与跳转阶段，在该阶段宏程序将对各类信息与预设值进行判断，以决定继续运行程序还是停止并报警提示。若该阶段各类信息经判断无误则返回主程序的加工阶段，否则根据相应的问题跳转至相应的报警。

N3001~N3006行为报警信息输出阶段，宏程序根据之前的分析与跳转结果给予提示性的警报，以方便操作者或技术人员查找出错的原因。譬如当实际圆弧半径与预设的值不一致时（即：#701不等于#13）则输出如图3所示的报警，同时机床停止运行。当工人遇到该报警时就应该仔细检查自己所使用的刀片是否与程序要求的一致，或者检查刀具圆

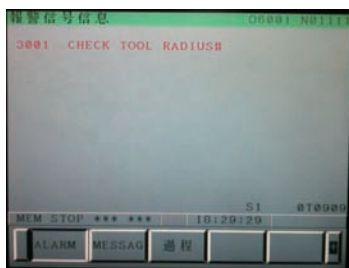


图3 刀具报警信息



图4 对刀报警信息

弧半径是否正确输入到寄存器中。

又如当程序运行后出现图4所示的报警信息，工人就应立即检查该刀具X轴对刀值输入是否正确、刀片的安装是否正确、刀片的定位是否可靠，然后重新对X轴进行对刀，直至该报警消失即达到预设的精度。至此，子程序的防误阶段便告结束。

5. 结语

以上是笔者提出的结合宏指令编制具有防误功能数控程序的方法。采用这种方法基本上可以杜绝由于人为因素导致的废品，提高产品的合格率。而且工人或技术人员根据报警的提示可方便快捷地找出问题所在，快速地具有针对性地将问题解决。

在实际使用中，我们还可加入如磨损值等更多数据的防误。在具有对刀臂的机床上与自动对刀相结合更是可以实现无需人工输入的全自动操作，将大大提高加工过程的自动化。

本文例子中的程序均在台湾FTC-50车床上运行通过，已达到期望的功能。

参考文献：

- [1] [美]彼得·斯密德.数控编程手册[M].北京：化学工业出版社，2006.
- [2] 北京FANUC机电有限公司.FANUC Series 0i-TC操作说明书[E].北京：北京FANUC机电有限公司，2004.
- [3] 刘旭林.数控车床的坐标转换方法研究[J].机械设计与制造，2008，(1)：171-172.

MW (收稿日期：20140913)