

辅助功能M

辅助功能M

在西门子中，大家都知道有M功能，而M0-M99中的M功能是通过动态的M功能译码来实现的。在NC加工程序中M功能执行的时候，会发出一个信号，传送到PLC的接口，也就是DB21.DBB194-DB21.DBB206，而这些信号，在PLC中只存在一个PLC的扫描周期，所以，一般会使用S指令去置位到某一个地址，比如一个用户定义的数据块中的一位，然后在去进行相应功能的处理。

但是动态M的译码，在接口信号中只给出了0 - 99，那么100以上的M功能是如何在PLC中实现的呢？这就用到了扩展M功能，DB21.DBD72，所有执行的M功能的数值，就会发送到这个双字节中，在下一个M功能执行以前，他会保持不变。在PLC中我们就可以用这样的方法来实现。比如你要实现M200，Q1.0是M200所要执行的动作：

L 200

用S指令去置位到某一个地址，比如一个用户定义的数据块中的一位，然后在去进行相应功能的处理。

L DB21.DBD70

==I

S Q1.0

M功能的数值范围为：0 - 2147483647(max. INT value)

用上面的方法，我在840D上试验过M200000（几个0忘记了反正很多，呵呵）的功能，

还有可能出现一种M0和M1功能的替代M功能，这个功能有着M0和M1相似的功能，可以任意定义为M功能内的所有数值（不包含已经有定义的那些，比如M3,M4等），

The MD 22254: AUXFU_ASSOC_M0_VALUE must contain the value of an auxiliary function number, e.g. 134.

M134 then initiates the same function as performed by M0.

In the machine data 22256: AUXFU_ASSOC_M1_VALUE must contain the value of an auxiliary function number, e.g. 173.

M173 then initiates the same function as performed by M1.

It is possible to make particular distinctions in the programming, e.g.

M1 ; Conditional stop in part program and

M173 ; Conditional stop in user cycle.

Conclusive distinctions are taken care of in the user PLC program.

参考FB1中的辅助功能

有些具有特定意义的辅助功能是由机床参数来定义的。看FB1的570页。比如M0、M1一直到M6，M17,M19,M30,M40--M45,M70等。这些在系统的机床参数中作了预定义，定义是由一组机床参数组成。这里包括：22040 - - 22080，一共5个参数，其中22080最为重要，他定义了这些M功能的一些特性。

MD 22080 AUXFU_ PREDEF_SPEC	Bit7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Output specification	Output at block end	Output during motion	Output prior to motion	Spindle response following acknowledgment	No output to the PLC interface	No predefined function, output to the PLC interface	High-speed acknowledgment with OB40	Normal acknowledgment after OB1 cycle

BIT2位定义了这个M功能是作为一个用户的M功能，还是去触发一个系统的功能，比如M3，如果BIT2位为0的话，他是执行一个系统的主轴正转功能，而在bit2位位1的时候，他就不在执行主轴正转功能，而是作为一个用户定义的M功能，需要在PLC中使用动态M功能代码来进行相应的动作。

作为M3--M5，如果作为一个系统功能，那么其信号是不发往动态的M功能的PLC接口。

而他要是作位一个用户定义的功能，那么其信号就会发到PLC的动态M功能接口。

所以，M功能业分为2中，一种是在机床参数中定义的，可以触发一种系统的功能。

The configured M function for tool change:

MD22560 \$MC_TOOL_CHANGE_M_CODE (Tool change with M Function). 如果用此参数，那么换刀子程序名称TOOL.SP

或者如下：

Call of subroutine SUB_M102 through M102

MD10715 \$MC_M_NO_FCT_CYCLE[1] = 102

MD10716 \$MC_M_NO_FCT_CYCLE_NAME[1] = "SUB_M102"

一种是用户定义的，要在PLC中定义其相应的动作。

在调用M6换刀子程序的时候已经取消的零点偏移和基本偏移，换刀结束后恢复原来的偏移，参考一下换刀子程序看看。

M6换刀子程序：

```
%_N_TOOL_SPF          ; >>> 用户循环名称 <<<
; $PATH=/_N_CUS_DIR    ; >>> 用户循环路径 <<<
PROC TOOL_SAVE_DISPLOF ; >>> 用户循环定义 <<<
IF $P_ISTEST GOTOF _END ; 在“程序测试”状态下，换刀循环可不执行
IF $P_SEARCH<>0 GOTOF _END ; 在“程序搜索”时，换刀循环可不执行
IF $P_TOOLNO==$P_TOOLP GOTOF _NO ; 如果编程刀具（$P_TOOLP）等于
; 当前刀具（$P_TOOLNO），无换刀动作

G500 D0                ; 取消零点偏移，取消刀具补偿
G75 Z=0                ; Z轴进入固定点（换刀点）
SPOS=$MN_USER_DATA_FLOAT[0] ; 主轴准停位置存在机床数据 MD14514[0]
MSG(“换刀开始 *** 原刀具号: T” <<<$P_TOOLNO) ; 显示最多 127 个字符
; >>> 主轴定向后可利用自定义 M 代码启动 PLC 换刀逻辑
; >>> PLC 将“读入禁止” (V32000006.1)置位，使该固定循环停止
; >>> PLC 控制刀库和机械手动作
; >>> 换刀完成后，PLC 将“读入禁止”复位，使该固定循环继续执行
M206                  ; 激活刀具参数
STOPRE                ; 程序译码停止
G153 G01 Z0 F2000    ; G153 取消基本偏移（非模态）
MSG(“换刀开始 *** 新刀具号: T” <<<$P_TOOLP)
GOTOF _END
_NO:
MSG(“无换刀动作 *** 原因: 编程刀具号=主轴刀具号”)
_END:
M17                   ; >>> 固定循环结束 <<<
```