

§ 4 PMAC命令汇总:

注意:

PMAC的语法不区分大小写;
如非注释,空格在PMAC的语法中并不重要;
{ } ----在 { } 中的项目可被符合定义的任意项目替代 ;
[] ----在 [] 中的项目在语法上是可选可不选的;
[{项目} ...] ---表示前面的项目在语法上是可被重复的;
[. {项目}] ---这些包含在语法中的句点用来说明一个范围;
() ---- 圆括号也包含在语法中.

定义:

常量----以数值表示且其数值不可变;
变量----保持一可变值的量;
I-变量----设置卡的特性, 固定含义的变量;
P-变量----供编程用的全局变量;
Q-变量----供编程用的局部变量(在坐标系中);
M-变量----供用户使用的分配内存的指针变量;
预定义变量----在卡中拥有固定含义的记忆码;
函数----SIN,COS,TAN,ASIN,ACOS,ATAN,ATAN2,LN,EXP,SQRT,ABS,INI;
运算符----供数值进行算术运算和按位控制的逻辑运算:
+, -, *, %(mod), &(and), |(or), ^(xor);
表达式----常量,变量,函数和运算符的组合;
数据----常量或是表达式;
关系运算符----评估两值之间的关系: =, !=, >, !>, <, !<, ~, !~;
条件----基于关系运算之上的真或假的条件判别;
简单条件-----{表达式}{关系运算符}{表达式};
复合条件-----简单条件的逻辑组合;
电机----控制硬件设置的元件,通过数字来指定;
坐标系----同时工作的电机的集合;
轴----组成一个坐标系的元素,通过从字母X,Y,Z,A,B,C,U,V,W中选择字母来说明;
缓冲区-----用户存储器中为程序和列表所准备的空间,可容纳256个运动程序和32个可编程控制逻辑块.

在线命令:

(被PMAC接收以后可以立即执行)

在线通用命令:

地址模式命令:

@n----写地址卡片为n(n为从0到f的十六进制数),只用于串联主机;
@----将当前编址的卡报告给主机,只能串联主机;
#n----让电机n为当前编址的电机;
#----报告当前编址的电机号;
&n----定义坐标系n 为当前编址的坐标系;
&----报告当前编址的坐标系.

通信控制键:

<CTRL-H>---从主机上删去最后一个字符(相当于BACKSPACE);
<CTRL-I>---从主机上重复最后一个命令(相当于TAB);
<CTRL-M>---命令行的结束(相当于回车);
<CTRL-N>---报告当前命令行的校验和;
<CTRL-T>---选择串联通信方式为全双路通信或是半双路通信;
<CTRL-W>---从DPRAM缓冲区中执行ASCII命令;
<CTRL-X>---取消当前PMAC命令和响应字符;
<CTRL-Y>---将最后的命令报告给主机,准备重复备置卡片;
<CTRL-Z>---让串行口成为通信口.

一般全程命令:

\$\$\$---全部复位:包括所有的电机和坐标系;
\$\$\$***-----复位并重新初始化整个卡;
PASSWORD={字符}-----为PROG1000-32767,PLC0-15设置/确认口令
SAVE---将I-变量保存到EEPROM中;
UNDEFINE ALL-----取消所有的坐标系的定义.

全程作用命令:

<CTRL-A>-----取消所有的运动程序和运动;
<CTRL-D>-----使所有的PLC和PLCC程序无效;
<CTRL-K>-----取消所有电机的输出;
<CTRL-L>-----关闭循环程序缓冲区;
<CTRL-O>-----在所有坐标系中作进给保持;
<CTRL-Q>-----在计算运动结束后退出所有程序;
<CTRL-R>-----在所有坐标系中执行工作程序;
<CTRL-S>-----在所有坐标系中分步执行工作程序;
<CTRL-U>-----打开循环程序缓冲区.

全程状态命令:

<CTRL-B>-----将所有电机的状态字报告给主机;
<CTRL-C>-----将所有坐标系的状态字报告给主机;
<CTRL-E>-----根据数据的形式,以二进制的方式报告地址内容;
<CTRL-F>-----报告所有电机的随动误差(未转换);
<CTRL-G>-----以二进制的形式报告全程状态字;
<CTRL-P>-----报告所有电机的位置(未转换);
<CTRL-V>-----报告所有电机的速度(未转换);
???-----以十六进制ASCII的形式报告全程状态字;
DATE---报告所用版本程序包的日期;
LIST[{缓冲区}]-----报告所打开(或指定)缓冲区的内容;
SIZE-----以字形式(小组信息)报告打开的存储器的大小;
TYPE-----报告PMAC类型;
VERSION-----报告程序包的修订版本.

寄存器存取命令:

R{地址},{内容}-----以十进制报告指定存储器字地址(或指定地址号)的内容;
RH{地址},{内容}-----以十六进制报告指定存储器字地址(或指定地址号)的内容;
W{地址},{常量},{常量}..-----将值写到指定存储器字地址中(或将一些值写到一定范围的地址中);

PLC控制命令:

ENABLE PLC{常量}[,{常量}...]-----使能未编译的PLC程序
DISABLE PLC{常量}[,{常量}...]-----禁止未编译的PLC程序
ENABLE PLCC{常量}[,{常量}...]-----使能编译的PLC程序
DISABLE PLCC{常量}[,{常量}...]-----禁止编译的PLC程序

全局变量命令:

{常量}-----如果所有表都已满则相当于P0={常量},否则数值进入到表中;
I{常量}={表达式}-----表达式的值被分配给I-变量;
I{常量}[..{常量}]=*-----设置I-变量[S]给缺省[S]值;
P{常量}[..{常量}]= {表达式}---表达式的值被分配给P-变量[S];
M{常量}[..{常量}]= {表达式}---表达式的值被分配给M-变量[S];
M{常量}-> {定义}-----定义M变量地址;
M{常量}->*-----M-变量被定义为非指针型的变量;
I{常量}[..{常量}]---将I-变量的值报告给主机;
P{常量}[..{常量}]---将P-变量的值报告给主机;
M{常量}[..{常量}]---将M-变量的值报告给主机;
M{常量}[..{常量}]->-----将M-变量的定义报告给主机;

缓冲区控制命令:

OPEN PROG{常量}---为进入/编辑打开指定运动程序缓冲区;
OPEN ROT-----打开所有定义好的循环程序缓冲区;
OPEN PLC{常量}-----打开指定的PLC程序缓冲区;
CLOSE-----关闭当前打开的缓冲区;
CLEAR-----删去打开的缓冲区的内容;
DEFINE GATHER[{常量}]-----用所有打开的存储器(或以指定的大小)建立一个数据采集缓冲区;
DELETE GATHER-----删去数据采集缓冲区;
GATHER[TRIGGER]---开始数据采集(通过外部触发器);
ENDGATHER-----停止数据采集;
DELETE PLCC{常量}-----删去指定的编译好的PLC程序;
DELETE TRACE---删去程序轨迹缓冲区(不产生动作; 保持反向兼容性);
DEFINE TBUF{常量}---建立指定序号的轴变换矩阵;
DELETE TBUF-----删去所有轴变换矩阵;
DEFINE UBUFFER{常量}-----定义一个指定序号的用户缓冲区.

在线坐标系命令:

(在指定的坐标系下被立即执行)

轴定义命令:

#n->[{常量}]{轴}[+{常量}]-----按电机号,轴比例系数,偏移量来定义轴;
例: #1->X
 #4->2000A+500
#n->[{常量}]{轴}[+{常量}]{轴}[+{常量}]{轴}][+{常量}]---
按电机号,轴比例系数,偏移量定义2或3轴.只在 XYZ或UVW组中才有效;
例: #1->8660X-5000Y
 #2->5000X+8660Y+5000
#n->-----报告在此代码段中电机n的轴的定义;

UNDEFINE-----删去在此代码段中的所有轴的定义.

一般坐标系命令:

??-----以十六进制ASCII的形式报告坐标系的状态;
%{常量}-----指定进给率修调值;
%-----将当前的进给率修调值报告给主机.

程序控制命令:

R----执行当前程序;
S----分步执行当前程序;
B[{常量}]-----设置程序序号到指定位置;
H---给坐标系进给保持;
A---取消现在即将开始的程序或运动;
Q---中止程序; 在最后的运算程序命令结束后停止运动;
/---在当前正在执行的运动结束后中止程序的执行;
****---当处在保持状态时使程序保持允许微动;
MFLUSH----删去未执行的同步M-变量存储堆栈的内容.

坐标系变量命令:

Q{常量}[..{常量}]= {表达式}----将表达式的值分配给Q-变量(S);
Q{常量}[..{常量}]----将Q-变量的值报告给主机.

轴特征命令:

{轴}={表达式}----改变定义轴的位置的值;
Z----令现在坐标系中所有轴的实际位置等于零;
INC[({轴},{轴}...)]----令所有轴(或指定的轴)以增量方式运动;
ABS[({轴},{轴}...)]----令所有轴(或指定的轴)以绝对值方式运动;
FRAX[({轴},{轴}...)]----令指定的轴用在矢量进给率计算中;
PMATCH----让坐标系中轴的位置重新匹配电机命令的位置(用在轴的定義或电机的位置在上一次轴的运动后有所改变的情况下); 但如仍有轴与电机交叉连接,此命令无效.

缓冲区控制命令:

PC---将下一个将执行的程序号和行(偏置)报告给主机;
LIST PC,[{常量}]----列出正在工作的程序将被计算的下一行(并指定其后的行);
PE---将当前正在执行的工作程序号和行(偏置)报告给主机;
LIST PE,[{常量}]----列出当前工作程序正在执行的行(并指定其后的行);
DEFINE ROT{常量}----为编址的坐标系确认指定字大小的循环运动程序缓冲区;
DELETE ROT----对编址的坐标系删去循环运动程序缓冲区;
PR----报告循环程序缓冲区中在正在执行点和上次读入行之间的行的数目;
LEARN----读入现在命令的位置,并加以轴命令来打开程序缓冲区;

在线电机命令:

(在当前已编址的电机上这些命令将被立即执行.所以当正在执行一个运动程序的坐标系时,除报告命令以外,这些命令将被拒绝.)

一般电机命令:

\$-----重置电机,反馈装置和调整相位;
HM----执行电机的回零程序;
HMZ---执行电机的回零程序不产生移动;
K-----关断给电机的输出;
O{常量}-----指定量级的开环输出.

电机微动命令:

J+ ----电机按正方向无限微动;
J- ----电机按反方向无限微动;
J/ ----电机微动停止,并恢复到位置控制;
J= ----电机微动到前一次微动或前一次手动的位置;
J={常量}-----电机微动到指定位置;
J=* -----可变的微动到某一位置;
J:{常量}-----电机微动到距当前位置指定距离的位置;
J:* -----从当前位置开始按可变的量进行微动;
J^{常量}----电机微动到距当前实际位置指定距离的位置;
J^* -----从当前实际位置开始可变的增量进行微动;
{jog command}^{常量}----触发时才微动,终值表示从触发器位置到停止的距离.

报告命令:

P---报告电机的位置;
V---报告电机的速度;
F---报告电机的随动误差;
?---以十六进制ASCII的形式报告给电机的状态字。

缓冲区控制命令:

DEFINE BLCOMP {入口},{计数长度}-----确认电机的间隙补偿表被指定的数值填充;
DELETE BLCOMP-----删去给电机的间隙补偿表;
DEFINE COMP {入口},{#{source},{#{target}},],[{计数长度}----确认电机的螺距补偿表被指定的数值填充;

DEFINE COMP {行}•{列},{#{source1},{#{source2},{#{target}},],[{ 计数长度1},{计数长度2}-----确认电机的两维螺距补偿表被指定的数值填充;
DELETE COMP-----删去给电机的螺距补偿表;
DEFINE TCOMP{入口},{计数长度}-----确认电机的扭矩补偿表被指定的数值填充;
DELETE TCOMP-----删去给电机的扭矩补偿表。

运动程序命令:

移动命令:

{轴}{数据}[{轴}{数据}...]----简单的运动语句; 可被用在混合的,过渡点的或样条方式下
例: X1000 Y(P1) Z(P2*P3)
{轴}{数据}:{数据}[{轴}{数据}:{数据}...]-----给定位置和速度的移动; 只能用在PVT模式下;
例: X5000:750 Y3500(P3) A(P5+P6):100
{轴}{数据}^{数据}[{轴}{数据}^{数据}...]-----触发时才移动,为RAPID模式的派生;
{轴}{数据}[{轴}{数据}...][{向量},{数据}...]----圆周运动; 只能用在圆周模式下; 向量指向圆心。

例: X2000 Y3000 Z1000 I500 J300 K500

DWELL{数据}----保持相同的命令的位置；基于定时的基础上；
DELAY{数据}----保持相同的命令位置；基于可变时间的基础上；
HOME{常量}[, {常量}...]----让指定电机返回零位；
HOMEZ{常量}[, {常量}...]----让指定电机作回零运动；

移动模式命令:

LINEAR-----混合线性插补运动模式；
RAPID-----所有轴以最大速度和加速度移动的模式；
CIRCLE1----顺时针圆弧插补移动模式；
CIRCLE2----反时针圆弧插补移动模式；
PVT{数据}—位置/速度/时间的移动模式；
SPLINE1----均匀三次样条曲线移动模式；
SPLINE2----非均匀三次样条曲线移动模式；
CC0 -----关闭刀具半径补偿；
CC1 -----打开左刀具半径补偿；
CC2 -----打开右刀具半径补偿；

轴特征命令:

ABS[({轴}[, {轴}, ...])]-----让所有(或指定)的轴处于绝对运动模式下；
INC[({轴}[, {轴}, ...])]-----让所有(或指定)的轴处于增量运动模式下；
FRAX[({轴}[, {轴}, ...])]----指定轴矢量进给率；
NORMAL{向量}{数据}[{向量}{数据}...]----给圆周运动和刀具补偿指定到平面的标准矢量；
PSET{轴}{数据}[{轴}{数据}...]----设置轴位置值；
R{数据}-----设置圆的半径值；
CCR{数据}---设置刀具半径补偿值(模态的)；
TSEL{数据}—选择指定的轴变换矩阵；
TINIT -----将被选择的轴变换矩阵初始化为单位矩阵；
ADIS{数据}—设置所选矩阵的位移向量给所指定的用于开始的Q-变量的值；
IDIS{数据}—增大所选矩阵的位移向量给所指定的用于开始的Q-变量的值；
AROT{数据}—设置所选矩阵的旋转角度/比例部分给所指定的用于开始的Q-变量的值；
IROT{数据}—通过指定的用于开始的Q-变量的值来放大所选矩阵从而增大它的旋转角度/比例部分的变化。

运动特征命令:

TM{数据}----指定运动时间(模态的)；
F{数据}-----指定运动速度(模态的)；
TA{数据}----指定运动加速度时间(模态的)；
TS{数据}----指定S-曲线加速度时间(模态的)；

变量赋值命令:

I{常量}={表达式}-----将表达式的值赋给指定的I-变量；
P{常量}={表达式}-----将表达式的值赋给指定的P-变量(S)；
Q{常量}={表达式}-----将表达式的值赋给指定的Q-变量(S)；
M{常量}={表达式}-----将表达式的值赋给指定的M-变量(S)；
M{常量}= {表达式}---在下次运动起始时同步地赋值；
M{常量}&={表达式}----M-变量的值与表达式的值作“与”运算后赋给M-变量；
M{常量}|={表达式}----M-变量的值与表达式的值作“或”运算后赋给M-变量；
M{常量}^={表达式}----M-变量的值与表达式的值作“异或”运算后赋给M-变量。

程序逻辑控制:

N{常量}-----行标;
O{常量}-----可变的行标;
GOTO{数据}-----跳到指定的N-标志的行; 不返回;
GOSUB{数据}[{字母}{轴}...]-----跳到指定的N-标志的行并返回(带自变量);
CALL{数据}[.{数据}][{字母}{轴}...]-----跳到指定的程序段(标志)(带自变量),并返回;
RETURN-----返回程序操作到最近的GOSUB或CALL;
READ({字母}[.{字母}...])-----允许子程序或子例程采用自变量;
G{数据}-----Gnn[.mmm]可解释为CALL
 1000.nnnmm; (PROG1000给想要的G代码动作提供了子程序)
M{数据}-----Mnn[.mmm]可解释为CALL 1001.nnnmm; (PROG1001给想要的M代码动作提供子程序)
T{数据}-----Tnn[.mmm]可解释为CALL
 1002.nnnmm; (PROG1002给想要的T代码动作提供了子程序)
D{数据}-----Dnn[.mmm]可解释为CALL
 1003.nnnmm; (PROG1003给想要的D代码动作提供了子程序)
S{数据}-----设置Q127给{数据}的值;
PRELUDE1{call command}-----在随后的运动之前给调用命令的模式执行;
PRELUDE0-----使模式PRELUDE调用不被激活;
IF({条件}){动作}-----符合条件则执行动作;
IF({条件})-----符合条件则执行以下的部分;
ELSE{动作}-----符合前面的“假”的条件时则执行动作;
ELSE-----符合前面的“假”的条件时则执行以下的部分;
ENDIF-----结束条件语句的声明;
WHILE({条件}){动作}---反复执行动作直到条件不为“真”;
WHILE({条件})-----循环执行以下的语句部分直到条件不为“真”;
ENDWHILE-----结束循环语句的声明;
BLOCKSTART-----从BLOCKSTART到BLOCKSTOP之间的所有命令将被单步执行;
BLOCKSTOP-----与BLOCKSTART对应的结束单步执行的声明;
STOP-----中止程序的执行; 准备重新开始;
WAIT-----与WHILE循环语句配合使用,当条件为“真”时中止程序的执行。

混合命令:

COMMAND “{command}” -----替代主机发出命令;
COMMAND^{字母}-----发出字符控制命令;
SEND “{消息}”-----通过主机界面传送消息;
SENDS “{消息}”-----通过串行接口传送消息;
SENDP “{消息}”-----通过并行接口传送消息;
DISPLAY[{常量}] “{消息}”---将消息自指定的地址送给LCD显示;
DISPLAY{常量},{常量},{常量},{变量}-----将变量值用指定的地址和格式送给LCD;
ENABLE PLC{常量}[.{常量}...]---使指定的PLC程序有效;
DISABLE PLC{常量}[.{常量}...]--使指定的PLC程序无效;
ENABLE PLCC{常量}[.{常量}...]--使编译好的PLC程序有效;
DISABLE PLCC{常量}[.{常量}...]-使编译好的PLC程序无效;

PLC程序命令:

条件:

IF({条件})-----通过对条件的判别以决定进入哪一个分支;
WHILE({条件})---条件循环的开始; 如果条件为真,则始终执行WHILE循环中的PLC操作;
AND({条件})-----IF或WHILE中的混合条件形式;
OR({条件})-----IF或WHILE中的混合条件形式;
ELSE-----IF语句中,符合“假”条件时则转入相应分支;
ENDIF---IF语句的结束,只能用在ELSE语句之后,不能在其前;
ENDWHILE---WHILE循环语句的结束。

动作:

{变量}={表达式}----给变量赋值;
COMMAND “{command}”-----代替主机发命令;
COMMAND^ {字母}-----发出字符控制命令;
SEND “{消息}”-----通过主机界面传送消息;
SENDS “{消息}”-----通过串行接口传送消息;
SENDP “{消息}”-----通过并行接口传送消息;
DISPLAY[{常量}] “{消息}”---将消息自指定的地址送给LCD显示;
DISPLAY{常量},{常量}. {常量},{变量}----将变量值用指定的地址和格式送给LCD;
ENABLE PLC{常量}[,{常量}...]---使指定的PLC程序有效;
DISABLE PLC{常量}[,{常量}...]--使指定的PLC程序无效;
ENABLE PLCC{常量}[,{常量}...]--使编译好的PLC程序有效;
DISABLE PLCC{常量}[,{常量}...]-使编译好的PLC程序无效;

PMAC I-变量汇总

I1~I99 -----卡的总的设置;
I100~I186-----电机#1的设置;
I187~I199-----坐标系1的设置;
I200~I286-----电机#2的设置;
I287~I299-----坐标系2的设置;
...
I800~I886-----电机#8的设置;
I887~I899-----坐标系8的设置;
I900~I979-----编码器1~16的设置(5个一组);
I980~I1023----保留给将来使用。

全程I-变量:

I1-----串行接口通信方式;
I2-----面板失效控制;
I3-----信号交换输入输出控制;
I4-----通信校验和有效;
I5-----PLC程序开/关;
I6-----错误报告模式;
I7-----循环到达位置号;
I9-----实时中断周期;
I10-----随动中断时间;
I11-----可编程移动计算时间;
I12-----微动到位计算时间;
I13-----可编程移动分段时间;
I14-----执行自动位置匹配;
I15-----用户程序所用的度数/弧度控制;
I16-----循环缓冲区要求开启;
I17-----循环缓冲区要求关断;
I18-----固定缓冲区已满的报井点;
I19-----数据采集周期(在随动循环系统中);
I20-----数据采集选择标志;
I21~I44—数据采集资源地址1~24;
I45-----数据采集缓冲区的位置和模式;
I47-----<CTRL-W>命令的地址指针;
I48-----DPRAM伺服数据有效;
I49-----DPRAM后台数据有效;
I50-----快速移动速度模式有效;
I51-----螺距补偿有效;

I52-----“V”程序保持转换速度；
I53-----程序单步模式控制；
I55-----DPRAM后台缓冲区控制；
I56-----DPRAM ASCII通信中断有效；
I57-----DPRAM二进制循环缓冲区有效；
I58-----DPRAM ASCII通信有效；
I59-----DPRAM缓冲区最大电机/C.S.号；
I60-----自动转换的ADC寄存器地址；
I61-----自动转换的ADC寄存器的号；
I62-----复原控制的内部信息；
I8x-----电机x的齿轮转速比的第三旋变装置；
I89-----刀具补偿外角断点；
I9x-----电机x的齿轮转速比的第二旋变装置；
I99-----间隙滞后作用。

电机I-变量 x=电机号(#x,x=1到8)

电机定义I-变量:

Ix00-----电机x被激活；
Ix01-----电机x的PMAC换相有效；
Ix02-----电机x命令输出(DAC)地址；
Ix03-----电机x位置环反馈地址；
Ix04-----电机x速度环反馈地址；
Ix05-----电机x的主(手轮)位置地址；
Ix06-----电机x的主(手轮)随动有效；
Ix07-----电机x的主(手轮)比例因子；
Ix08-----电机x位置环比例因子；
Ix09-----电机x速度环比例因子；
Ix10-----电机x上电伺服位置地址。

电机安全I-变量:

Ix11-----电机x的致命(断电)随动误差限制；
Ix12-----电机x的报警跟随误差限制；
Ix13-----电机x的正向软件限位；
Ix14-----电机x的反向软件限位；
Ix15-----电机x在极限位置或行止位置的减速速率；
Ix16-----电机x的最大编程速度；
Ix17-----电机x的最大编程加速度；
Ix19-----电机x的最大允许手动(JOG)加速度。

电机运动I-变量:

Ix20-----电机x的(手动和回零)加速时间；
Ix21-----电机x的(手动和回零)S-曲线加速度时间；
Ix22-----电机x的手动速度；
Ix23-----电机x的回零速度和方向；
Ix25-----电机x的 限位/回零标志/放大器标志 地址；
Ix26-----电机x的回零偏置；
Ix27-----电机x的位置滚动范围；
Ix28-----电机x的到位宽度；
Ix29-----电机x的输出-或第一位相-DAC偏差。

电机的基本随动控制I-变量:

Ix30-----电机x的PID比例增益；
Ix31-----电机x的PID微分增益；

lx32-----电机x的PID速度前馈增益;
lx33-----电机x的PID积分增益;
lx34-----电机x的PID积分模式;
lx35-----电机x的PID加速度前馈增益;
lx36-----电机x的陷波滤波器系数N1;
lx37-----电机x的陷波滤波器系数N2;
lx38-----电机x的陷波滤波器系数D1;
lx39-----电机x的陷波滤波器系数D2;
lx40~lx56—电机x扩展的伺服算法I-变量。

电机单环伺服机构改进:

lx57-----电机x的连续电流限制;
lx58-----电机x的积分电流误差等级;
lx59-----使电机x的用户写伺服算法有效;
lx60-----电机x的伺服环扩展周期;
lx63-----电机x的积分限制;
lx64-----电机x的“死区增益因子”;
lx65-----电机x的死区;
lx67-----电机x的线性位置误差(大步)限制;
lx68-----电机x的摩擦前馈;
lx69-----电机x的输出命令(DAC)限制。

电机换相I-变量:

lx70-----电机x的对于定义换相大小的的换向周期(N)号;
lx71-----电机x的每N个换向周期的编码器计算数;
lx72-----电机x的换相相角;
lx73-----电机x的相位查找输出值(DAC);
lx74-----电机x的相位查找时间;
lx75-----电机x的相位偏置;
lx76-----电机x的速度相位超前增益;
lx77-----电机x的感应电机磁场电流;
lx78-----电机x的感应电机滑动增益;
lx79-----电机x的第二相位输出(DAC)偏置;
lx80-----电机x的上电模式;
lx81-----电机x的上电相位位置地址;
lx83-----电机x的运行相位位置地址。

更深入的电机I-变量:

lx85-----电机x的间隙吸收率;
lx86-----电机x的间隙大小。

坐标系I-变量:

x=坐标系号(&x,x=1 d到8)
lx87-----坐标系x的缺省程序加速时间;
lx88-----坐标系x的缺省程序S-曲线加速时间;
lx89-----坐标系x的缺省程序反馈率;
lx90-----坐标系x的速率时间单元;
lx91-----坐标系x的缺省的工作程序号;
lx92-----使坐标系x的运动混合无效;
lx93-----坐标系x的时基控制寄存器地址;
lx94-----坐标系x的时基转换速率(和限制);
lx95-----坐标系x的进给保持减速速率;
lx96-----坐标系x的最大圆弧误差限制。

编码器/标志 设置I-变量:

给编码器 n(n=1->16)

I900~I904-----编码器1;

I905~I909-----编码器2;

I910~I914-----编码器3;

I915~I919-----编码器4;

...

I970~I974-----编码器15;

I975~I979-----编码器16;

I900,I905,I910,I915,I920,I925,I930,I935,

I940,I945,I950,I955,I960,I965,I970,I975 (编码器I-变量0)编码器n译码控制;

I901,I906,I911,I916,I921,I926,I931,I936,

I941,I946,I951,I956,I961,I966,I971,I976 (编码器I-变量1)使编码器n的滤波器无效;

I902,I907,I912,I917,I922,I927,I932,I937,

I942,I947,I952,I957,I962,I967,I972,I977 (编码器I-变量2)编码器 n的位置捕获控制;

I903,I908,I913,I918,I923,I928,I933,I938,

I943,I948,I953,I958,I963,I968,I973,I978 (编码器I-变量3)编码器n的标志选择控制。

宏代码支持I-变量:

I1000-----使宏结点辅助寄存器有效;

I1001-----光缆通讯校验时间周期。

PMAC错误代码汇总:

PMAC能够在响应命令是报告以下的错误信息:

Error	问题	解决方法
Error001	命令在程序执行时不响应	发出命令前应中止程序执行
Error002	口令错误	键入正确的口令
Error003	数据错误或不能识别的命令	纠正命令中的语法错误
Error004	非法字符:错误的值(>127 ASCII)或串行奇偶校验/成帧错误	纠正字符或检查串行线上的声音
Error005	因缓冲区关闭,命令被拒绝	应先打开一个缓冲区
Error006	缓冲区中已无空间给命令	删除或清除一些缓冲区以提供多一些的空间
Error007	缓冲区已在使用状态中	应先关闭当前打开的缓冲区
Error009	程序结构错误(如有ENDIF却没有IF)	纠正程序结构中的错误
Error010	在代码段中两个超行程限制设置给了一个电机	纠正或使限制无效
Error011	前一个运动还未完成	中止该运动或让该运动完成
Error013	坐标系中的一个电机未被激活	设置Ix00为1或从代码段中去该电机
Error014	坐标系中没有电机	在代码段中至少应定义一个电机

Error015	没有指向有效程序缓冲区的指针	应先用命令B,或清除不规则的缓冲区
Error016	执行了结构不正确的程序	纠正程序结构中的错误
Error017	在电机处于停止位置外时想要重始	用J=to motor[s]回的停止位置 return

注释:变量I6控制是否输出和怎样输出这些错误信息。

PMAC语法注释:

1. PMAC的语法不区分大小写。也就是说,在任何一个命令或语句中用大写字母或小写字母都是可以的。PMAC命令写在文件中时,大写字母有助于将它们同解释说明的文字更好地区分开来。
例如: X1000和x1000对于PMAC来说是相同的表达式。
2. 在语法定义中,括在花括号中的项目,如{数据},意味着用户可以在其中放入他所希望放入的内容,只要该内容是在此项目的定义限制以内。
例如: 如果语法为X{数据},则用户可以在其中放入X1000,X(P1),
或X(P2*P3+50),因为1000,(P1), 和(P2*P3+50)都是符合
{数据}的定义限制的。
3. 在语法定义中,括在[]中的项目是可选择的。如在[]之中又有圆括号(),则表示其中的项目是可以重复的。例如: 如果语法定为{轴}{数据} [{轴}{数据}...],则用户可以写
X1000,X1000Y1000,或X1000Y1000Z1000。
4. 如非特别说明,空格在PMAC的语法中是不重要的。