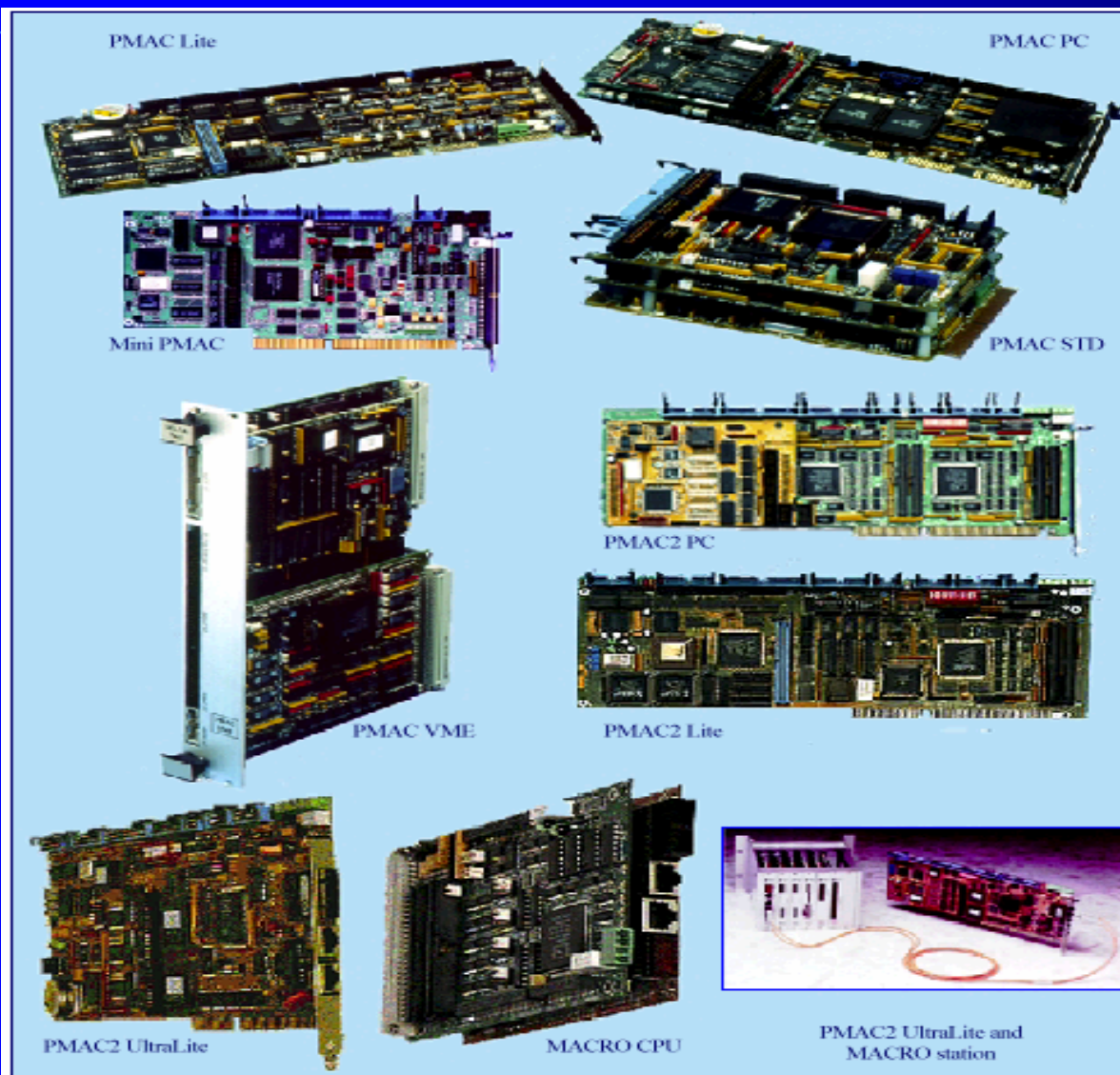


E-MOTION

PMAC多轴运动控制卡

(初级培训)



编写：章 文
贾慧鹏

E-MOTION

PMAC多轴运动控制卡

PMAC的含义:

PMAC是**program multiple axis controller** 可编程的多轴运动控制卡。

PMAC的特点:

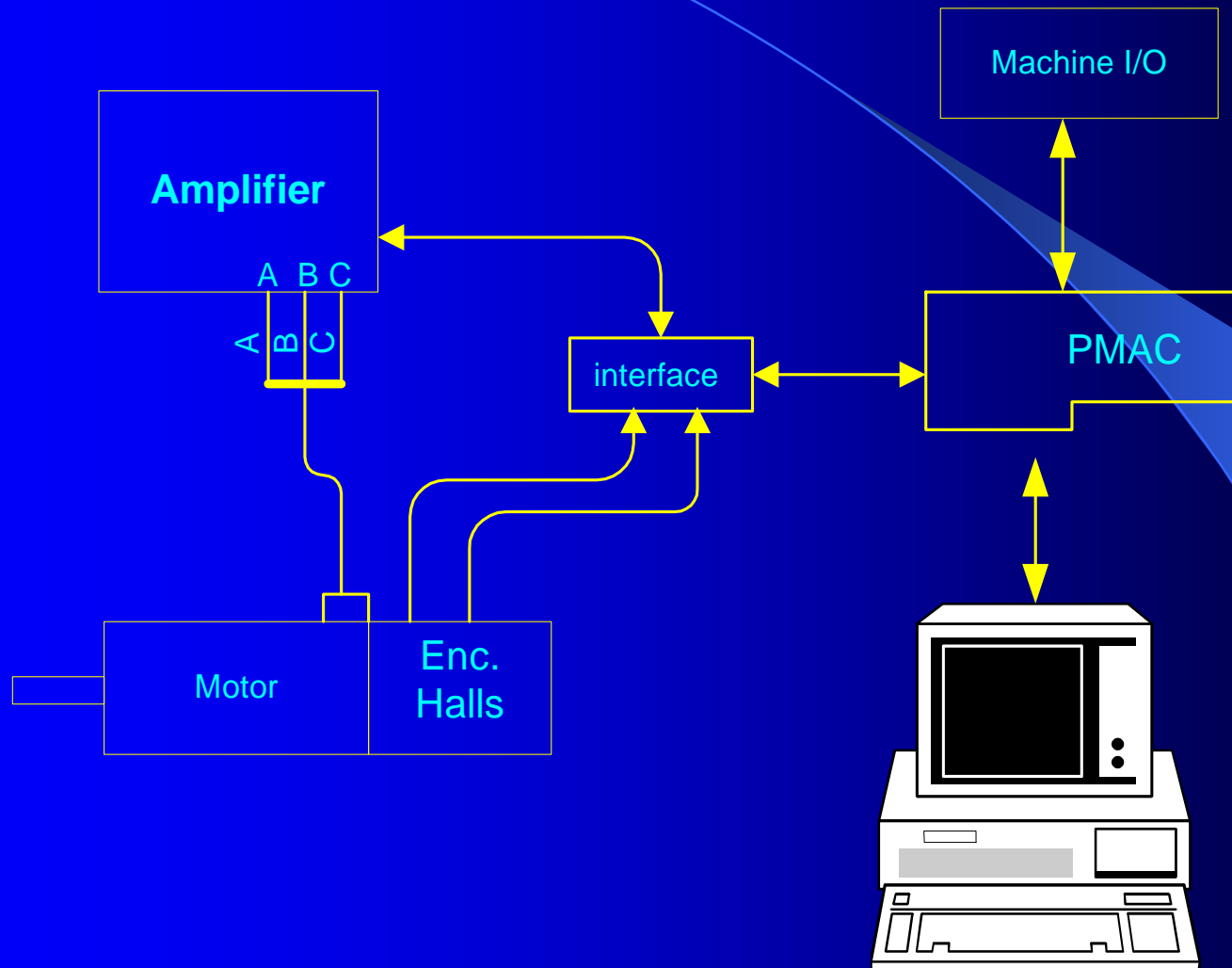
PMAC卡是美国**deltatau**公司的产品，是集运动轴控制，和**PLC**控制以及数据采集的多功能的运动控制产品。



NEW IDEAS IN MOTION

E-MOTION

PMAC多轴运动控制卡



Software Interface

- 指令8轴同时运动
- 使用功能强大的数字信号处理芯片 (DSP)
- PMAC的 CPU 使用 Motorola DSP56001 或 DSP56002 他可以 处理所有8轴的计算
- Turbo PMAC 使用 Motorola DSP5630x 处理所有32轴的计算

DSP 56001 → 20MHz 电池 RAM

DSP 56002 → 20-80 MHz 闪存RAM

DSP 5630x → 80-120 MHz 闪存RAM 对于TURBO PMAC

PMAC 关于特定的应用，可如下配置：

- 编写运动程序和PLC程序
- 选择硬件设定 (通过选项和附件)

每个**PMAC**固件有**8轴**的能力. 这**8轴**可以：

- 完全联动于一个坐标系下
- 可以独立的运行于各自的坐标系下
- 可以几个组合，实现某个功能
- 可以与其他 15块**PMAC**级联，实现128轴的完全同步运动



E-MOTION

PMAC多轴运动控制卡

PMAC's CPU 与轴的通讯 通过特殊设计的用户门阵列 ICs (DSPGATES). 每个门阵列:

- 4 路输出通道
- 4 路编码器输入通道
- 通过附件具有4路模拟量输入通道

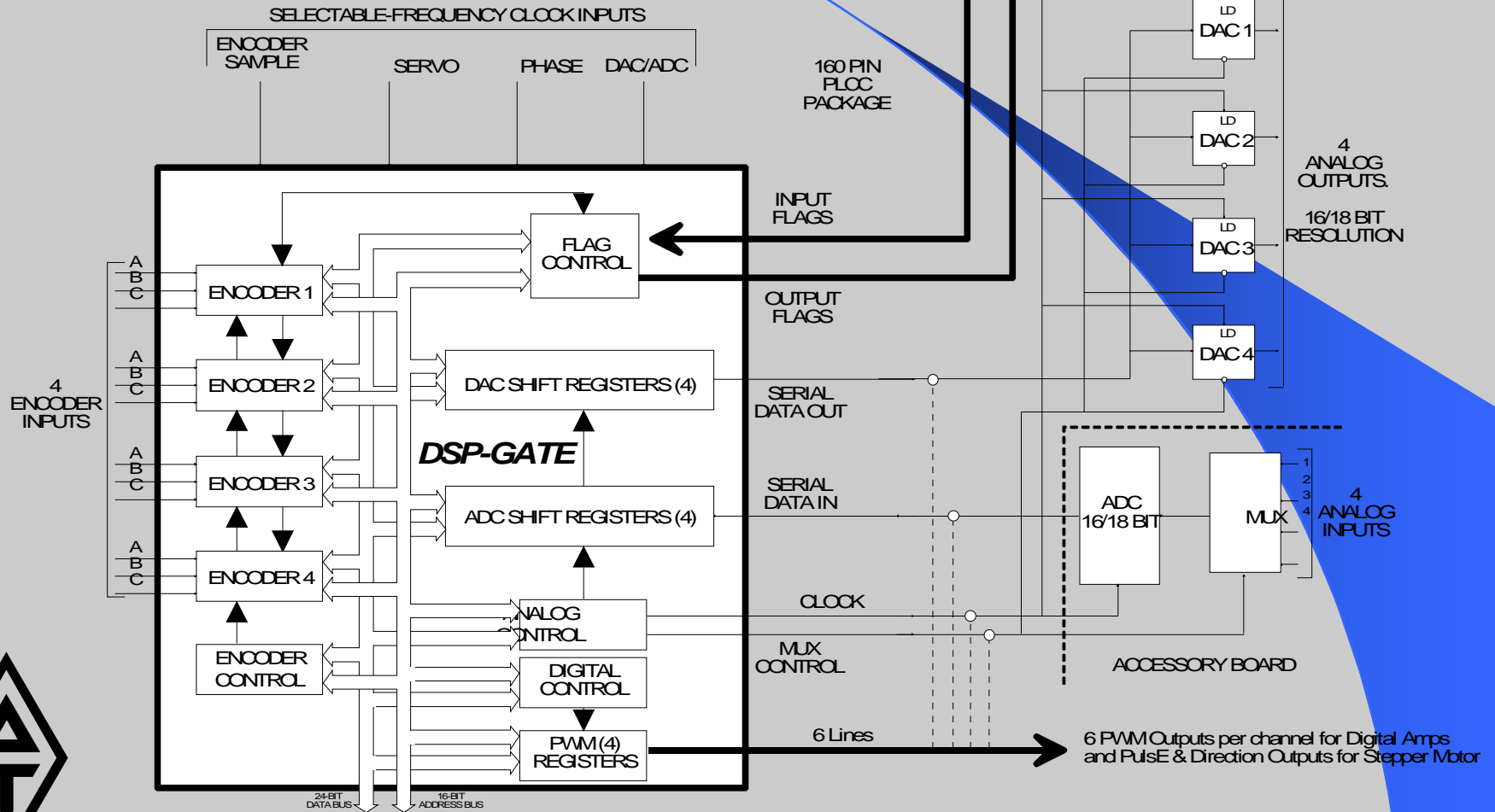
I 型 PMAC 可以使用 1 to 4 门阵列.



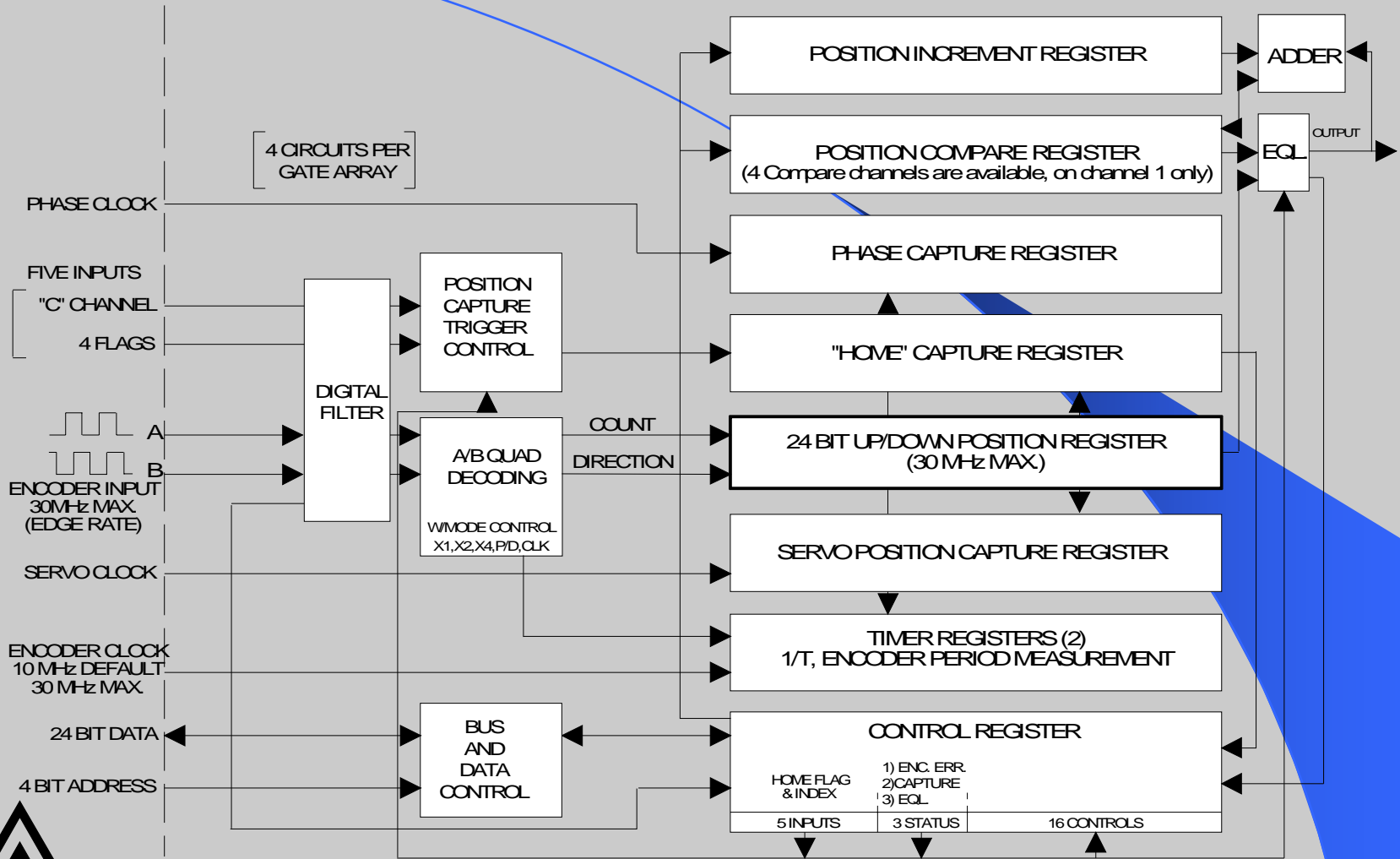
NEW IDEAS IN MOTION

PMAC MOTION CONTROLLER CUSTOM GATE ARRAY IC

HOME 1-4
+LIMIT 1-4
-LIMIT 1-4
FAULT 1-4



(GATE-ARY)



(BRADI ENCODER)

PMAC 1 CUSTOM GATE ARRAY (DSP GATE) ENCODER FUNCTIONS

PMAC 可以做过的工作

执行运动程序

- ◆ PMAC 在某一时间执行一个运动, 并执行有关运动的所有计算
- ◆ PMAC 总是提前混合即将执行的运动

执行PLC程序

- ◆ 以处理器允许的时间尽可能快的扫描PLC程序
- ◆ PLC适用于某些异步于运动程序的运动过程



PMAC 可以做过的工作(续)

伺服环更新

- ◆ 伺服环更新对于PMAC的用户是看不到的，是由PMAC卡自动执行的任务
- ◆ 伺服环根据运动的设计者编写程序公式，从当前的实际位置和指令位置增加指令的数值.

换向更新

- ◆ PMAC 以9KHZ的频率自动进行换向计算和控制
- ◆ PMAC 测量并估算 转子的磁场定向，然后处理电机的相之间的指令

PMAC 可以做过的工作(续)

常规管理

- ◆ 跟随误差限制
- ◆ 硬件超程限制
- ◆ 软件超程限制
- ◆ 放大器报警

在每个PLC扫描之间, PMAC 执行上述任务保证自身的正常更新, 如果这些功能不能在最小的频率内检测, 卡上的看门狗将报警.

同上位机通讯

- ◆ PMAC 可以与上位机实时通讯
- ◆ 如果指令是违法的, 他将报告给上位机

任务优先级

- ◆ 任务优先级保证卡的工作效率和安全
- ◆ 优先级是固定的, 但是他们的频率时可以由用户控制的

E-MOTION PMAC多轴运动控制卡

PMAC 反馈性能

- 不带附件：
 - 正交编码器带 1/T 插补细分
 - 脉冲加方向带 1/T 插补细分
- **ACC- 8D option 8 以及 ACC-51:**
 - 模拟量正弦输出信号，采用ACC-8D opt.8 可达到256倍频，采用ACC-51附件可达到4096倍频
- **ACC-14D/V:**
 - 接收并行绝对编码器反馈
 - 接收并行激光干涉仪反馈
 - 其他的并行字反馈
- **ACC-8D option 7:**
 - 接收4路旋变反馈

E-MOTION **PMAC多轴运动控制卡**

PMAC 反馈性能

- **ACC- 8D option 9**
 - 接收安川绝对编码器反馈
- **ACC-28A, ACC-28B , ACC-36, or PMAC2 Option12**
 - 接收线性电压信号, 电位机反馈信号
 - 接收其他类型的电压传感器信号
- **ACC- 29 (PMAC2内置):**
 - 磁制伸缩位移传感器
 - 相当的传感器
- **ACC-49:**
 - 三洋并行绝对编码器反馈
- **SSI Interface (ACC-xxE)**
 - 接收同步串行编码器, 适用于MACRO 或TURBO的装置

E-MOTION PMAC多轴运动控制卡

PMAC卡的种类:

。PMAC 1型卡包括:

MINI PMAC

控制轴数 : 2

UNIVERSAL PMAC

控制轴数: 4

PMAC-PC

控制轴数: 8

TURBO PMAC

控制轴数: 最多32轴

。PMAC 2型卡包括:

MINI PMAC2

控制轴数 : 2

UNIVERSAL PMAC2

控制轴数: 4

PMAC2-PC

控制轴数: 8

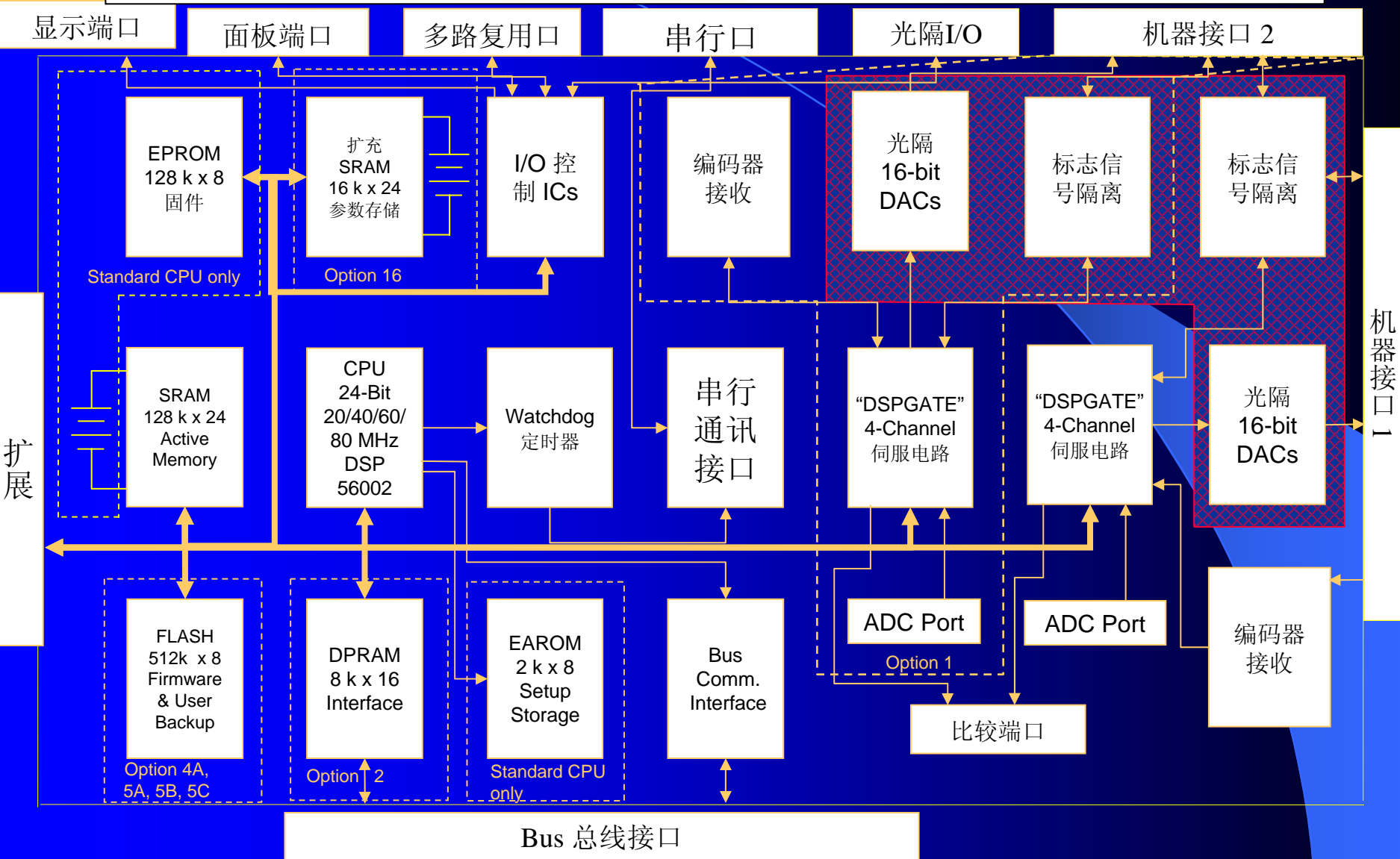
TURBO PMAC 2

控制轴数: 最多32轴

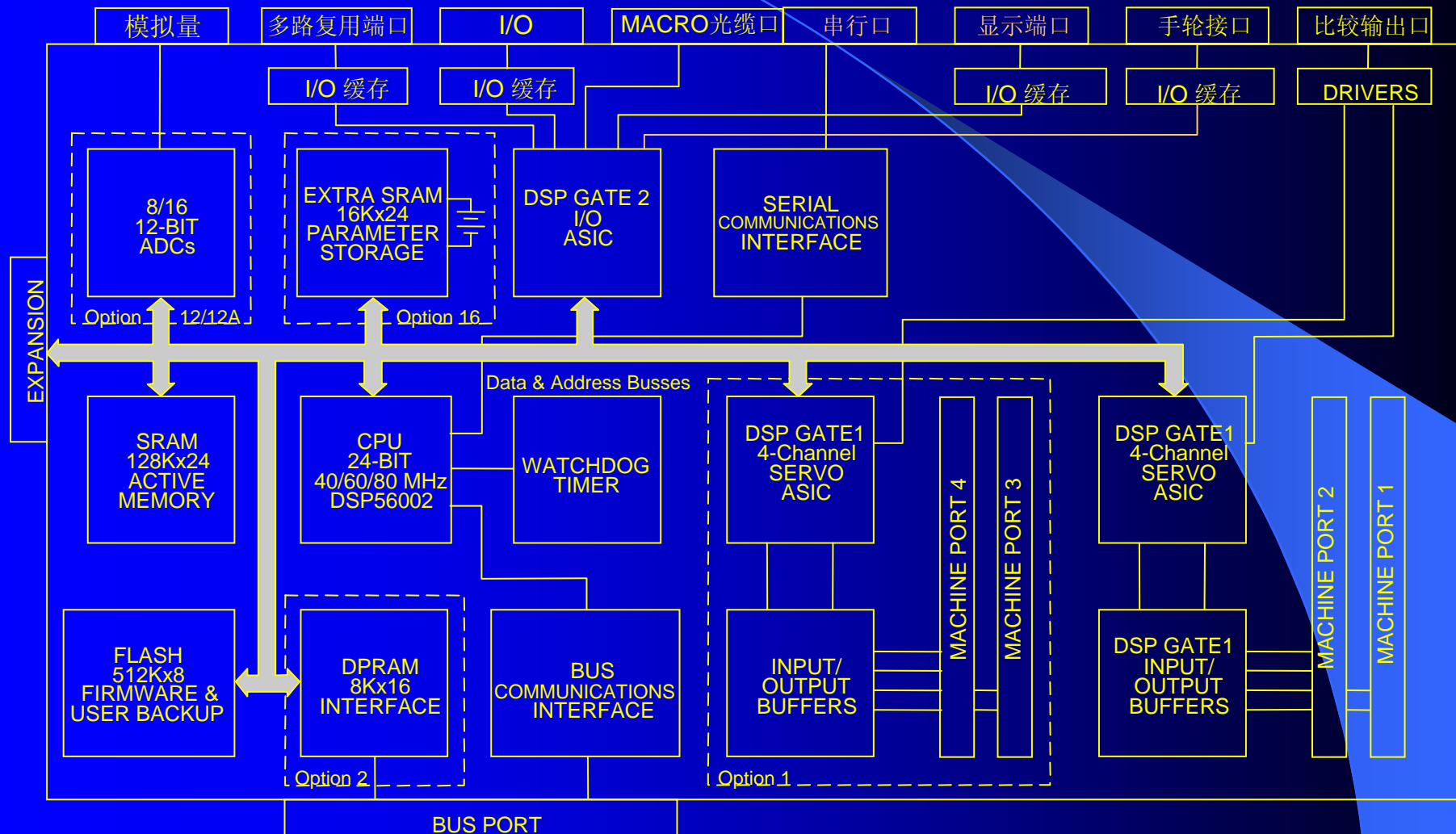
PMAC2- PC104

控制轴数: 4—8轴 (可选)

PMAC1 结构框图



PMAC2 结构框图



PMAC 1型卡与2型卡的主要区别:

	PMAC 1	PMAC2	PMAC2-PC104
CPU时钟（缺省）	20MHZ	40MHZ	40MHZ
缺省输出信号	DAC模拟量	PWM数字量	DAC/PPF混合
DPRAM 选项	只有8轴卡不在板	在板	在板
在板I/O点数	16IN 16OUT	32IN/OUT +8IN 8 OUT	32IN/OUT 8IN+8OUT
常用接线板	ACC8P ACC8D	ACC8F ACC8S ACC8E	ACC-1 ACC-2

PMAC2 相对 PMAC(1)

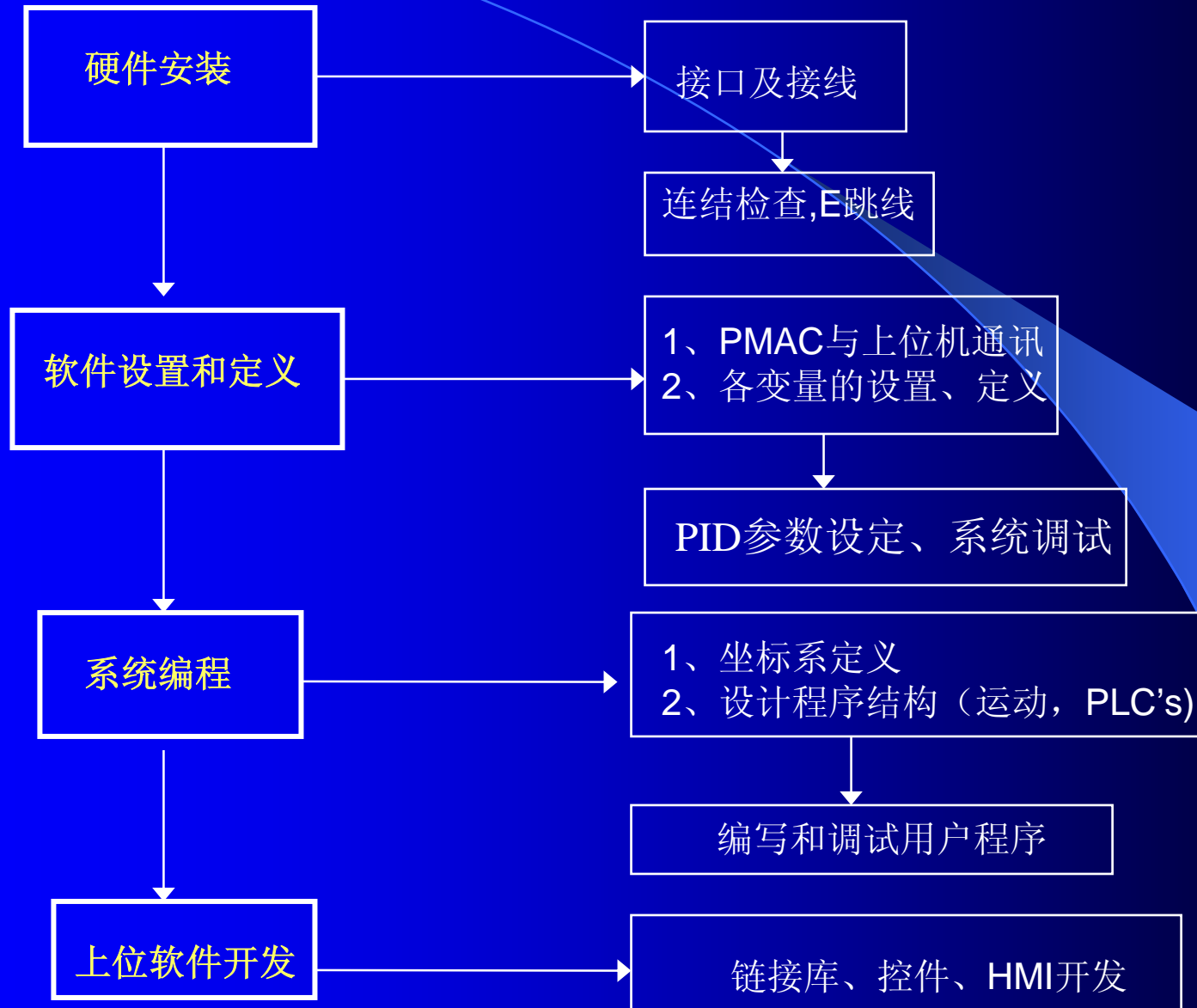
- 直接 PWM 输出
- 在板 MACRO 接口
- 更完善的正弦波信号输出控制
- 更完善的模拟量编码器信号分频能力
- 更强大的脉冲加方向信号输出控制
- 更出众的 磁致伸缩 MLDT 接口
- 增强的位置比较输出功能
- 更高的 DAC 信号输出分辨率(18-bit vs. 16-bit)
- 更多的在板功能选项:
 - DPRAM, ADCs, MLDT 接口, 追加编码器通道, 步进信号接口

PMAC(1) 相对于 PMAC2

- 原先的 PMAC 用户习惯应用
- 更加方便直接的传统模拟量信号控制接口：
 - 速度方式控制驱动
 - 转矩方式控制驱动
- 在板具有 24V I/O (JOPTO port)
- 直接控制面板接口(JPAN)



PMAC 系统应用



PMAC多轴运动控制卡

PMAC I型卡的接口:

. **MINI PMAC/ UNI PMAC/ PMAC-PC** 相同的接口

JEXP : PMAC卡的并行接口, 可连接的附件有
ACC14 并行I/O或反馈输入板
ACC39 三洋绝对编码器输入口
DPRAM 双端口RAM

J1(JDISP): PMAC 1型卡的显示接口。可连接的附件有
ACC12 系列的LCD 显示

J2(JPAN) :操作控制面板接口, 可连接的附件有**ACC18**,
I2=1 时, 该接口的输入及输出 可作为通用I/O点使用。

- **J3(JTHUMB):** 多路I/O点的扩展接口，该接口本身可提供8点输入及8点输出,还可与下属附件连接:

ACC34 I/O 扩展板，可提供32IN 32OUT 输入及输出

ACC16 BCD 拨码开关接口板

ACC27 专用8in/8out 接口板

ACC8D opt 7 旋转变压器反馈接口板

ACC8D opt 9 安川绝对编码器反馈接口板

ACC33 NC 操作面板接口板

- **J4(JRS422 or JRS232):** PMAC卡的串行接口
PMAC-PC 卡，在板为RS422口
MINI PMAC Universal PMAC 在板为RS232口

J5(JOPTO): 8IN/8OUT 通用I/O接口

JMACH: 60芯接口, 包括信号:

模拟量输出

编码其反馈

正负限位及回零信号

伺服使能和报警

卡的看门狗报警

卡的电源供给



E-MOTION

PMAC多轴运动控制卡

PMAC I型卡的接口：

. MINI PMAC 的特殊接口

- 1. J7 MINI PMAC 专门与附件 ACC23 ACC28A/B 连接接口**
- 2. J8 MINI PMAC 为JAUX口，提供辅助信号接口，其中包括：模拟量输入 位置比较 脉冲/方向输出 标志信号等**
- 3. MINI PMAC 在板具有4路编码器输入信号**
- 4. MINI PMAC在板具有2路V/F输出信号，可用于控制步进电机**
- 5. MINI PMAC 无J2(JPAN)接口**

E-MOTION

PMAC多轴运动控制卡

UNIVERSAL PMAC 的特殊接口

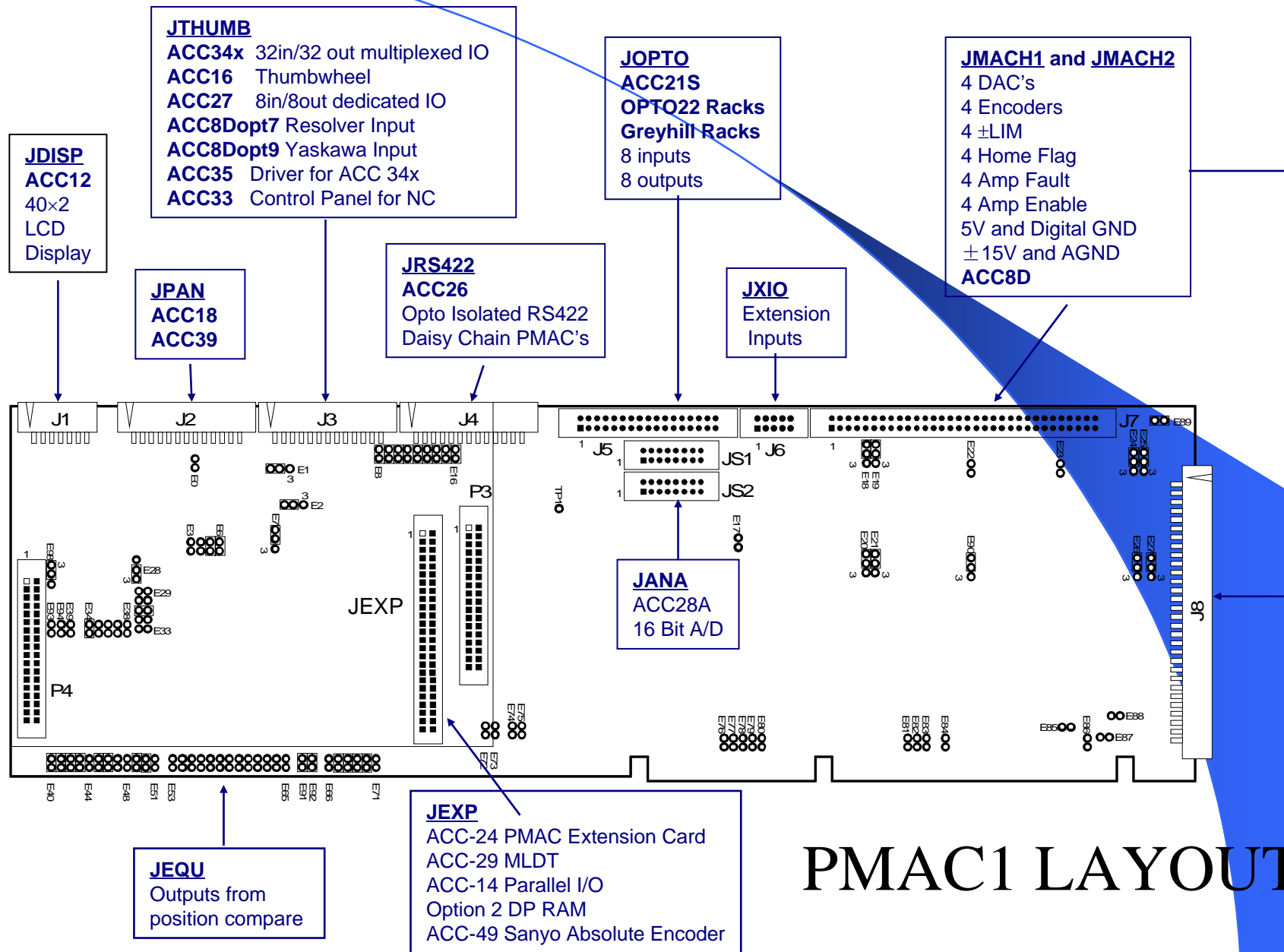
1. **J7 UNI- PMAC** 专门与附件 **ACC23 ACC28A/B** 连接接口
2. **J8 Universal PMAC** 为**JEQU**接口， 可提供**4**路伺服使能和**4**路位置比较相等信号

PMAC-PC 的特殊接口

1. **J6(JXIO)** 为多卡应用时的接口



NEW IDEAS IN MOTION



PMAC I 型卡的电源:

1. **+5V DC 电源** : 供给PMAC卡的数字电路工作
 - . **BUS 总线提供**
 - . **TB1 端子提供**
 - . **JMACH端子提供**
2. **+/-12~15V DC 电源**: 供给PMAC卡模拟信号工作
 - . **BUS 总线提供** 与相关跳线设置有关
 - . **TB1 端子提供**
 - . **JMACH端子提供**

注: **PMAC-PC卡无TB1端子**

PMAC I 型卡的跳线设置:

E3—E6: 伺服时钟频率 与I10变量有关, 缺省为E5 E6为ON

E29—E33: 相时钟频率, 缺省为9.04KHZ,E31设为ON.

E34—E38: 编码器采样时钟频率, 缺省为: 9.8304KHZ E34设为ON.

E17A-E17D: 1-4电机的伺服使能的极性控制

E24—E27: 1-4 编码器的输入形式控制, 是单端/差分输入。

E40—E43: PMAC卡号的设置, 缺省为E40—E43都设为ON,相当于是 @0号卡。

E44—E47: 设置串口通讯的波特率。缺省为9600波特率, E66设为ON.

E51: PMAC卡从新初始化的设置, 此跳线在PMAC卡正常工作时, 应为OFF, 卡由于故障或其他原因需要初始化时, 才需将此跳线设为ON..

PMAC I 型卡的跳线设置:

E85 E87 E88 : 设置PMAC卡模拟量电源, 缺省设置为都设为OFF状态, 表示PMAC卡的模拟量电源+-12~+-15V是通过外部供给, 而如果都设为ON,表示模拟量+-12~+-15电源是从PC机总线得到。

E89 E90 : 设置+-LIMn 和FLAG 的电源供给, 这两个跳线需要随着**E85 E88 E89**的设置变化而变化。

E91 E92 E66—E71 : PMAC卡占用计算机的总线地址设定。缺省地址为:
\$210(528),E66 E71设为OFF.

设“OFF”=1 “ON”=0

例:
$$\text{BUS ADD} = \$10 * \text{E66} + \$20 * \text{E67} + \$40 * \text{E68} + \$80 * \text{E69} + \$100 * \text{E70} + \$200 * \text{E71} \\ + \$400 * \text{E91} + \$800 * \text{E92}$$

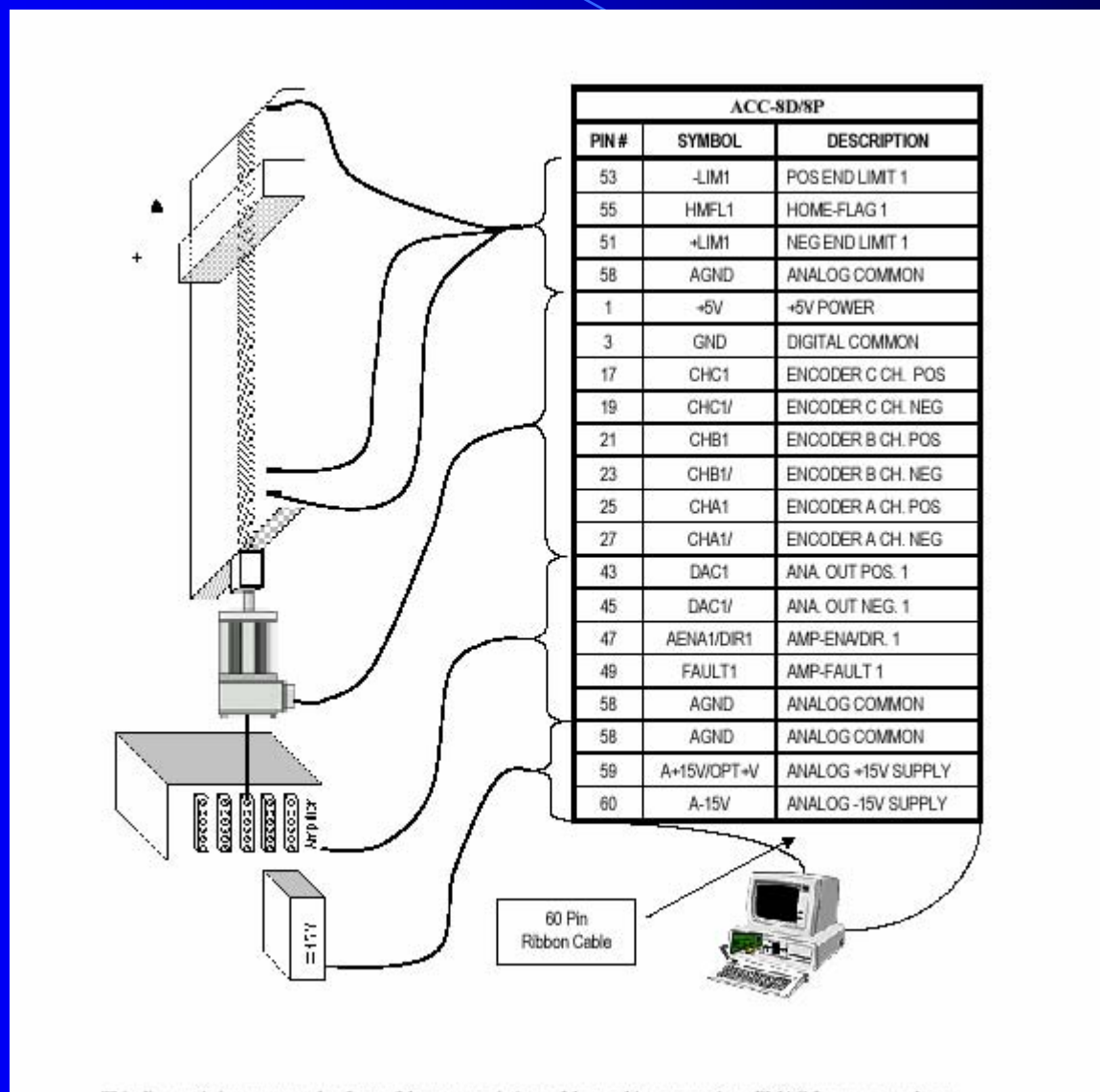
对应\$210地址, E66 E71设为OFF.

. MINI PMAC 卡V/F转换跳线设置:

E110~E115: V/F转换的频率设置

E116~E119: V/F转换信号PULSE 和DIR 信号设置





E-MOTION PMAC多轴运动控制卡

PMAC 1型卡接线

- PMAC1型卡可配接以下转接板：

1、控制信号输出及信号反馈接口 JMACH (64PIN)

- ACC-8P : 直接 $\pm 10V$ 的DAC输出 (4路)
- ACC-8D+OPT2 : V/F转换(脉冲+方向)输出
- ACC-8D+OPT4(4A) : PWM小功率驱动器 (20W/150W)驱动液压伺服 (比例阀控制) 或小功率DC直流伺服

PMAC 1型卡接线

- 2、I/O点连接

在板I/O

J5(JOPTO): 8IN/8OUT 通用I/O接口,24V输出电平, 带光隔。

J3(JTHUMB): 8IN/8OUT 5V TTL电平, 不带光隔。
(可选OPT-27光隔电路提供缓冲)

扩展I/O

ACC-34AA: (串行I/O) 32IN/32OUT, 24V电平带光隔。
可多块级联(最多32块), 与PMAC J3口相连。

ACC-14D:(并行I/O) 48 IN/OUT点(响应速度快),接PMAC J2口。

PMAC 1型卡接线

- 3、I型卡模拟量数据采集（A/D转换）

ACC-28A: 4路A/D INPUT 信号（15位分辨率）

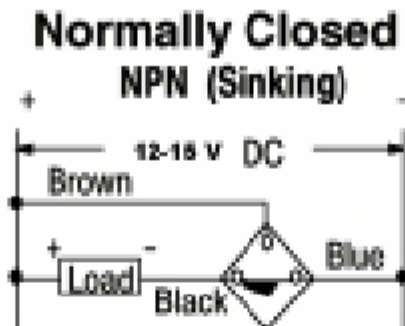
ACC-28B: 4路A/D INPUT 信号（16位分辨率）

ACC-36: 16路A/D INPUT 信号（12位分辨率）

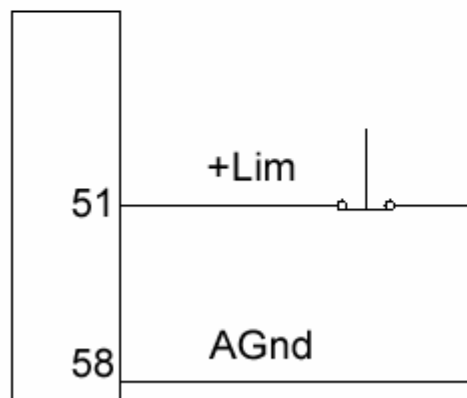
注：MINI-PMAC和PMAC-LITE 卡只能接JS1口总共4路A/D转换

PMAC-PC 可以接8路（JS1+JS2）

4、PMAC1 JMACH接口的LIM、HM 信号的连接 示意图

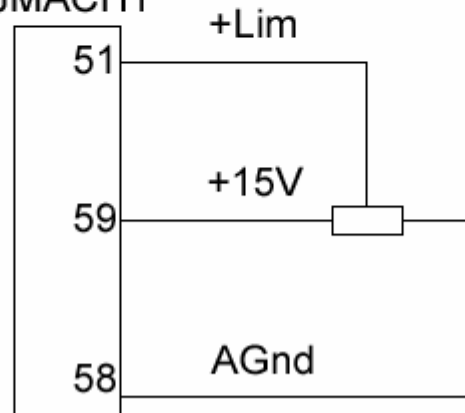


JMACH1



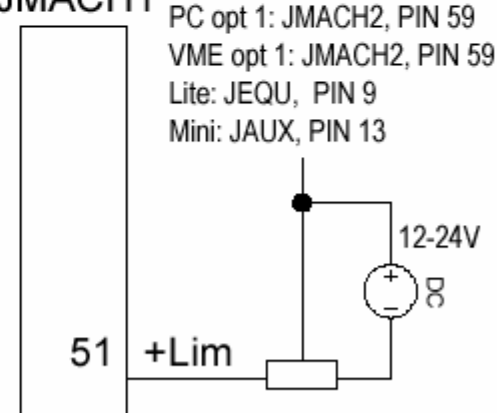
Dry Contact

JMACH1



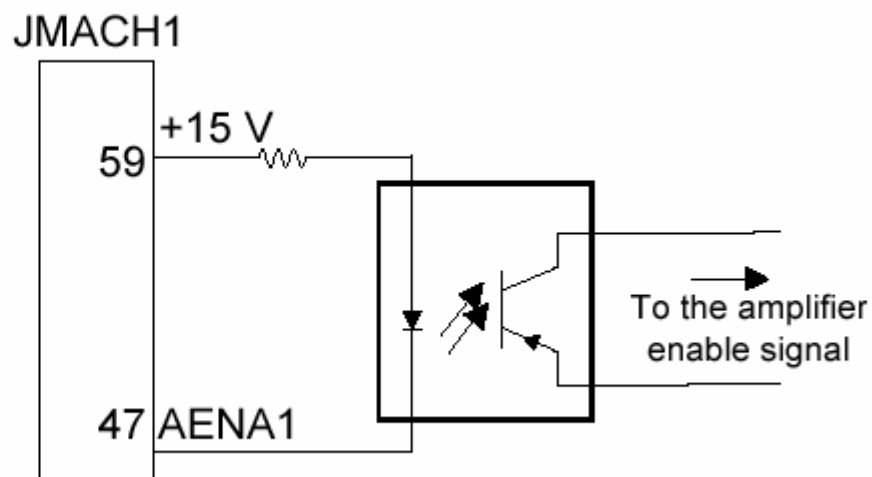
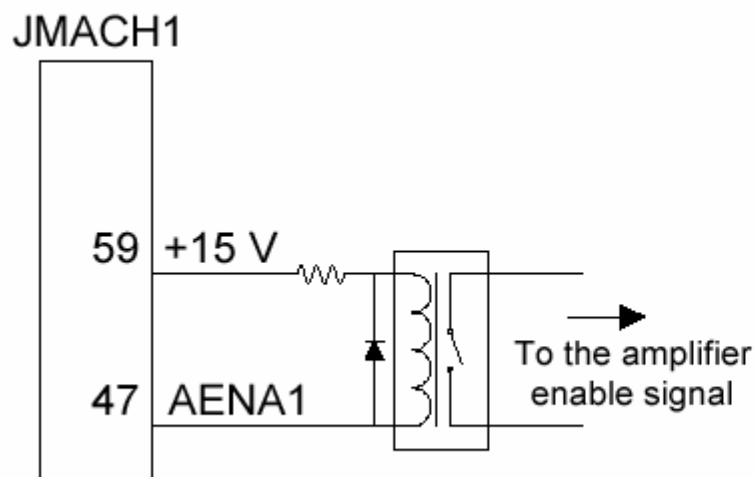
12-15 Volts proximity

JMACH1



15-24 Volts proximity

5、JMACH接口的AENA 使能信号的连接示意图



PMAC 2型卡的接口：

J1 (JANA)： 模拟量输入口，此接口可接受16路0-5V的模拟量信号，
注： MINI PMAC2没有此接口

J2 (JTHW)： 多端口I/O扩展口，此接口相当于1型卡的J3口，可连接ACC34A等附件，另外此接口可提供8In/8OUT点供用户使用。

J3 (JI/O)： 通用I/O接口，此接口可提供16in/16out输入及输出点，这32点I/O可由用户自行定义是输入还是输出，以及他们的极性。

J4(JMACRO)： MACRO 光缆接口，用于与具有光缆接口的驱动器和I/O板使用

PMAC 2型卡的接口：

J5 (JRS232 JRS422)： 串行数据接口。对于**MINI PMAC2**型卡只有**RS232**接口，而**PMAC2-PC**和**Universal PMAC2** **RS232**和**RS422**接口在板都具有。

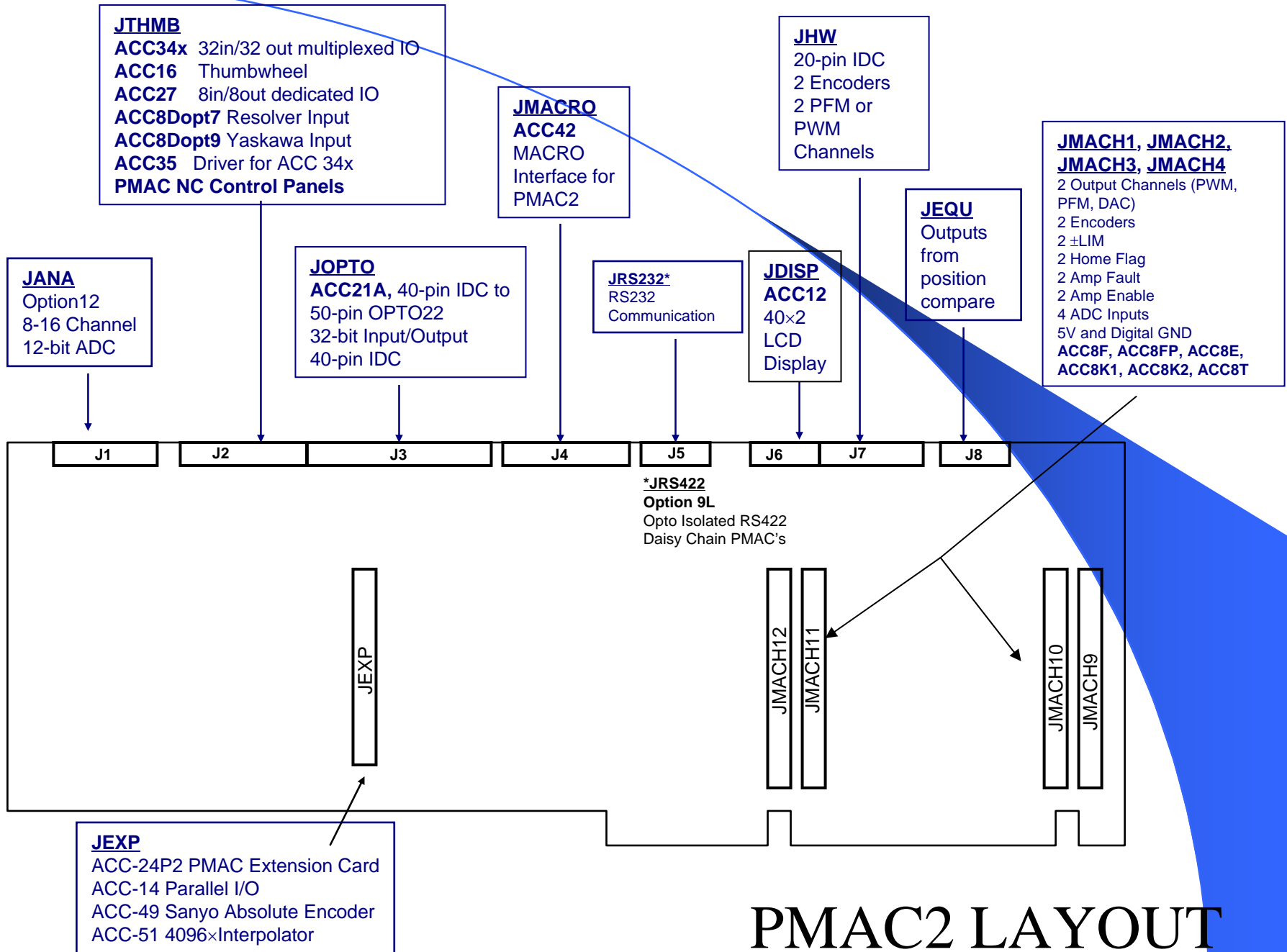
J6 (JDISP)： 显示器接口，相当于**1**型卡的**J1**口，可与**ACC12**系列的附件相连接

J7 (JHW)： 手轮编码器接口，可接受两路手轮或编码器信号，并且不占用电机的编码器通道。

J8 (JEQU)： 位置比较相等的输出信号接口，注：**MINI PMAC2**没有此接口

J9-J12 (JMACH1-4)： **1-8**电机的输入/输出接口

JS1： 此接口为**MINI PMAC2** 型卡模拟量的接受口，可与**ACC28A/B**连接



PMAC2型卡的跳线说明

E1 PMAC2型卡的卡号设置，缺省设置为“0”号卡，E1为OFF。

E3 PMAC2型卡从新初始化，此跳线用于2型卡在出现故障时，对卡进行初始化。

E7—E10 PMAC2型卡中断PC机的选择设置。

E14 PMAC2卡看门狗计时器控制，

PMAC 2 型卡的电源:

+5V DC 电源 : 供给PMAC卡的数字电路工作

- **BUS** 总线提供
- **TB1** 端子提供
- **JMACH**端子提供



E-MOTION

PMAC多轴运动控制卡

PMAC 2 型卡接线

- PMAC2 型卡可配接以下转接板：
 - 1、控制信号输出及信号反馈接口
JMACH (100 PIN) (2路)
 - ACC-8F : 直接PWM信号输出
 - ACC-8S : 脉冲+方向信号输出
 - ACC-8E : DAC 输出 ($\pm 10V$) 18位分辨率

E-MOTION PMAC多轴运动控制卡

PMAC 2 型卡接线

● 2、I/O点连接

在板I/O

J2 (JTHW): 8IN/8OUT I/O接口,5V输出电平不带光隔。

J3(I/O): 32 IN/OUT自定义 5V TTL电平，不带光隔。
(可选ACC-34AE光隔电路提供缓冲)

扩展I/O

ACC-34AA: (串行I/O) 32IN/32OUT, 24V电平带光隔。
可多块级联（最多32块），与PMAC2 J2口相连。

ACC-14D:(并行I/O) 48 IN/OUT点（响应速度快),接
PMAC J11口。

E-MOTION

PMAC多轴运动控制卡

PMAC 2 型卡接线

3、2型卡模拟量数据采集（A/D转换）

在板选项（**J1** 口）

OPTION 12 : 8路 A/D 输入（12位分辨率）

OPTION 12A: 16路A/D 输入（12位分辨率）

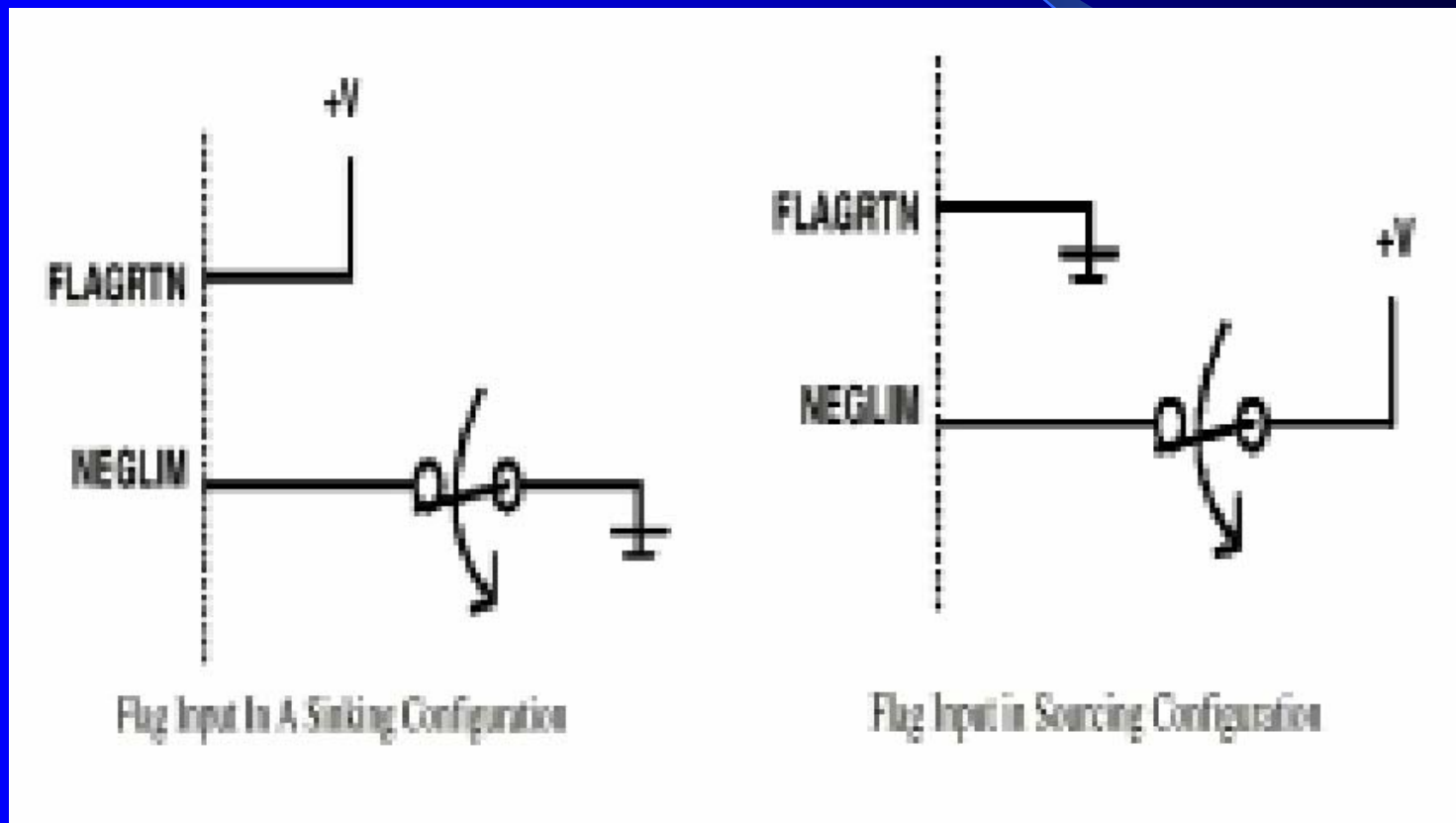
附件选项

只能接**ACC-28B**: 4路A/D INPUT 信号（16位分辨率）

注: **MINI-PMAC2** 卡直接接在**JS1**口

PMAC2-LITE 和 **PMAC2-PC** 必须通过**ACC-8T**转接

4、PMAC2 JMACH接口的+LIM -LIM HM 信号的连接 示意图



E-MOTION

PMAC多轴运动控制卡

PMAC 2-104型卡的接口及接线:



**PMAC2A-PC/104 Base Board shown
stack with the Option-1P and
Option-2P boards**

PMAC 2-104型卡的接口及接线:

PMAC2—PC104型卡可配接以下转接板:

1、控制信号输出及信号反馈接口 (4通道)

- JMACH1 (50PIN)

- **ACC-1** : **DAC** 输出和编码器反馈输入

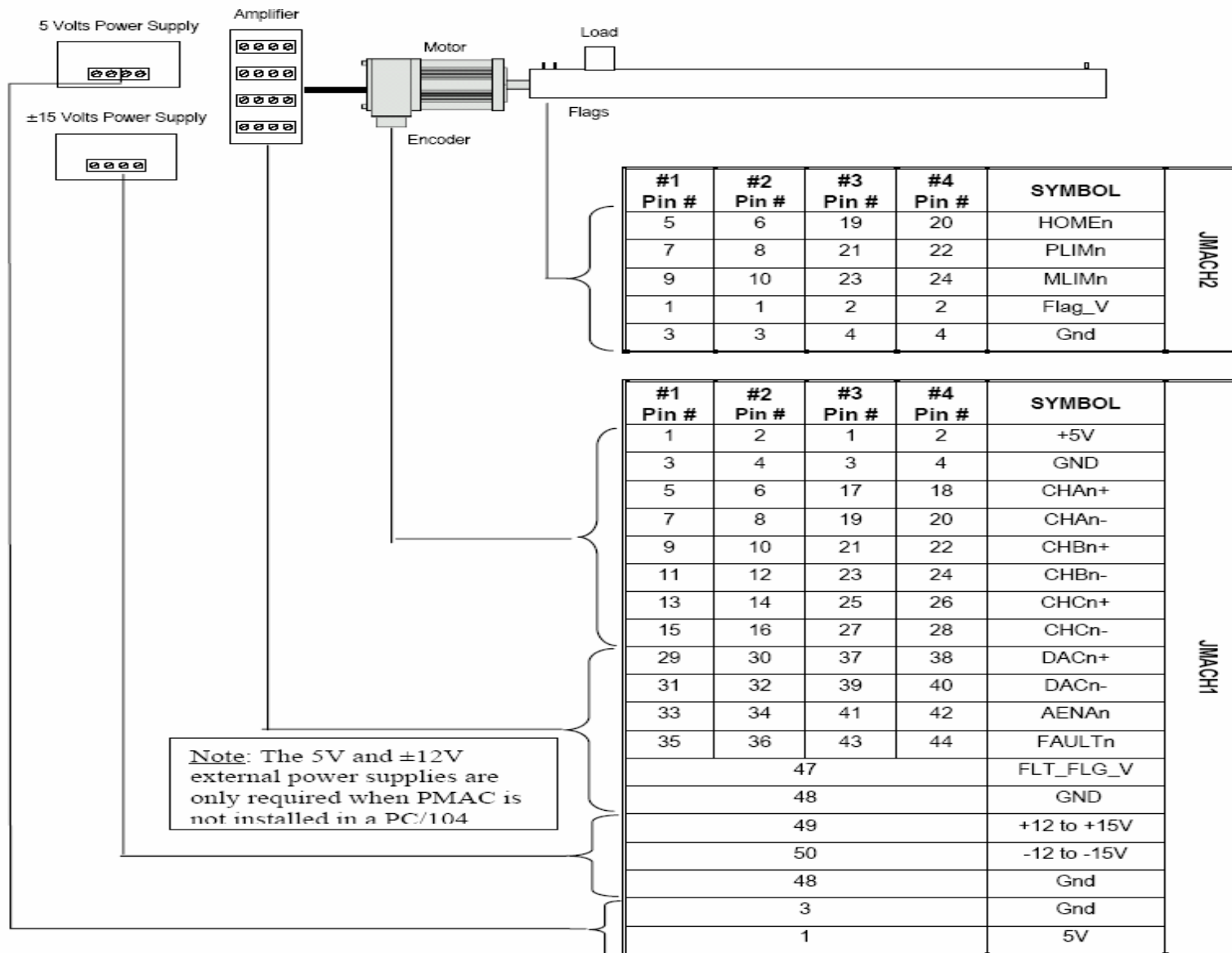
- JMACH2 (34PIN)

- ACC-2 : 脉冲加方向信号输出

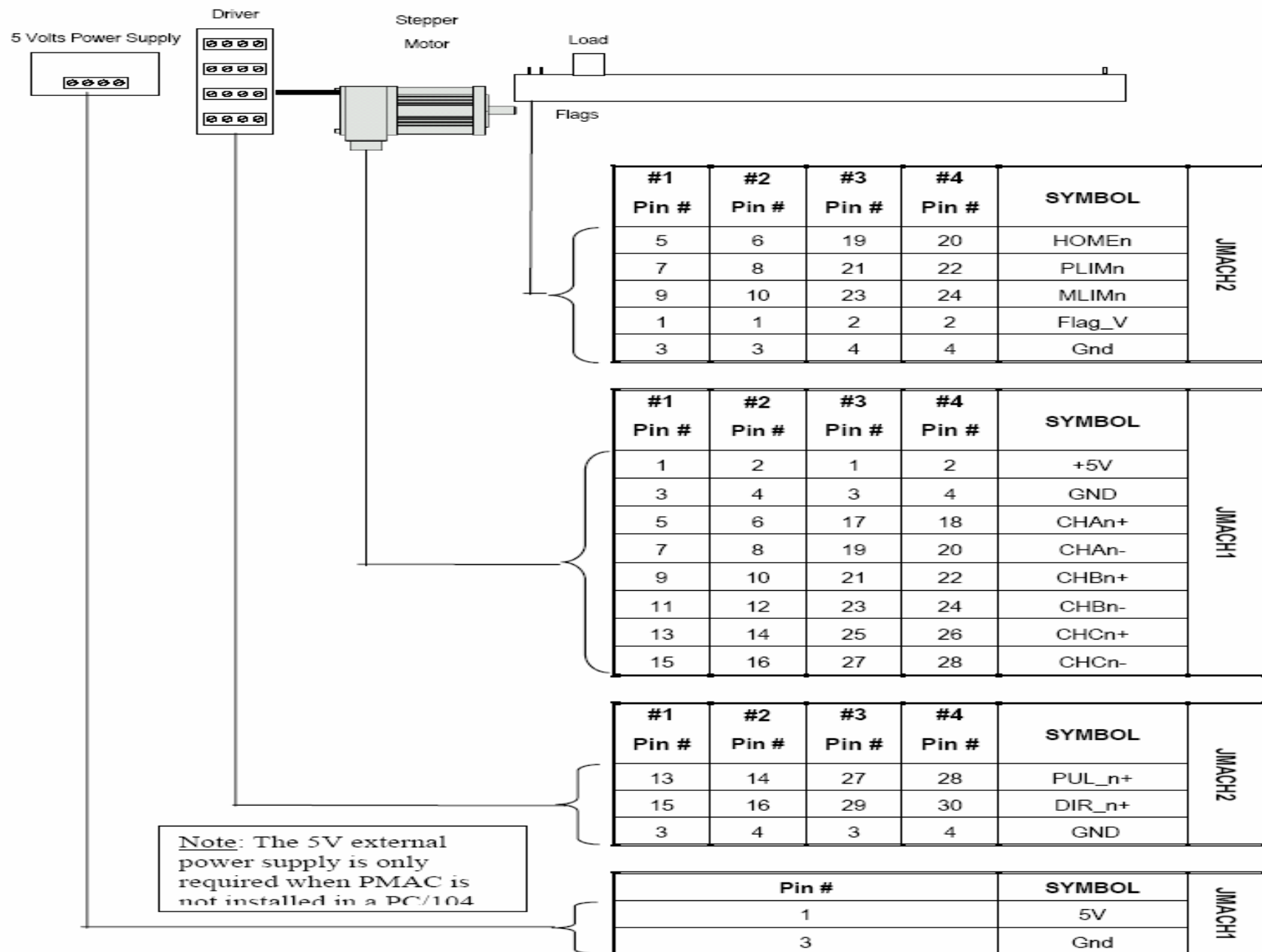
- ACC-8FS

- 4通道PWM信号输出

Machine Connections Example: Using Analog ± 10 Volts Amplifier



Machine Connections Example: using pulse and direction drivers



2、I/O点连接

在板I/O

- **ACC-1P OPT1 或 ACC-2P OPT3 的(J2口) :**
8IN/8OUT I/O接口,5V输出电平不带光隔。
- **ACC-2P OPT3 的(J7口) :**
16IN/OUT 自定义I/O接口, 5VTTL电平
- **ACC-1P OPT 1的 (J7口) :**
8 IN/8OUT 5-24V电平。

扩展I/O

ACC-34AA: (串行I/O) 32IN/32OUT, 24V电平带光隔.
可多块级联（最多32块），与ACC-1P或ACC-2P相连。

E-MOTION

PMAC多轴运动控制卡

PMAC 2-PC104 型卡接线

3、模拟量数据采集（A/D转换）

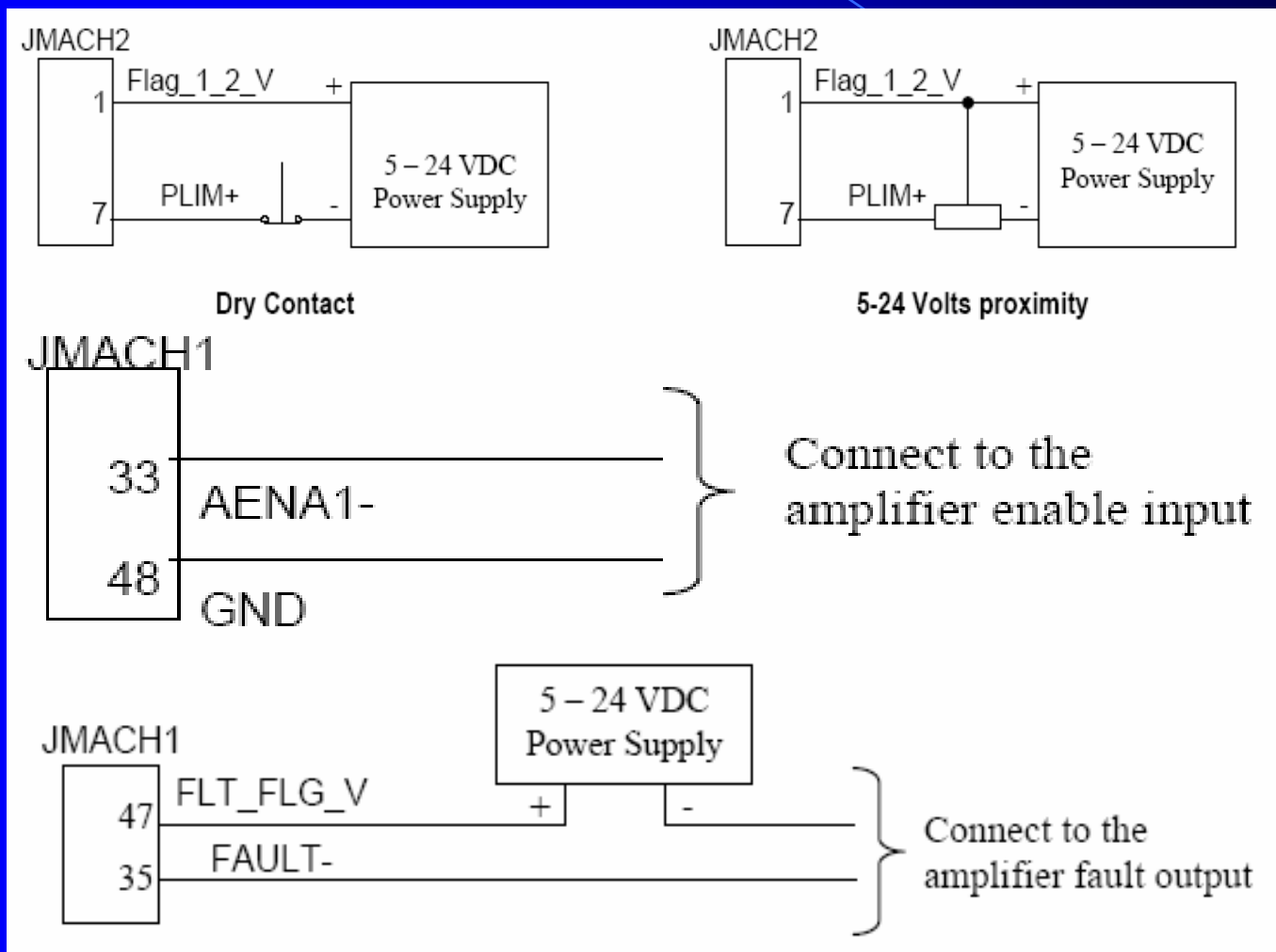
在板选项（**J1** 口）

OPTION 12：2路A/D 输入（12位分辨率）

附件选项

ACC-1P OPT2：2路A/D 输入（12位分辨率）

4、PMAc2-PC104 的LIM (HM)、使能、伺服报警信号的连接



PMAC 变量说明

PMAC I 变量

PMAC卡的I变量为卡，电机和编码器等参数变量，用于设置电机的速度，精度，回零等数值，以及坐标系的状态，还包括编码器的反馈形式。
PMAC卡的I变量分类如下：



PMAC 变量说明

PMAC I 变量：

I00—I99 :为PMAC卡的全局变量

I100—I186: 为#1电机的设置变量

I200—I286: 为#2电机的设置变量

I300—I386: 为#3电机的设置变量

。 。 。 。 。 。 。 。

I800—I886: 为#8电机的设置变量

I187—I199: 为&1坐标系的设置变量

I287—I299: 为&2坐标系的设置变量

。 。 。 。 。 。 。 。

I887—I899: 为&8坐标系的设置变量

PMAC 变量说明

PMAC I 变量：

I900—I904: 为编码器1的设置变量

I905—I909: 为编码器2的设置变量

。 。 。 。 。 。 。 。

I975—I979: 为编码器16的设置变量

注： I900以后的变量，PMAC2型卡与一型卡不同，具体的含义在后面介绍卡的操作时再分别说明。

P-变量

P-变量是用户全局变量可用于PMAC编程中的计算.

- ◆ 48-位浮点形式
- ◆ 1024 P-变量从 P0 to P1023

可用于:

1.计算

```
P100=P101*(sin(45))
```

2. 软件触发

```
IF( M1!= 1 AND P10 = 0 )
```


P-变量 继续

假设你编写运动轨迹为 $\text{SIN}(\theta) + \text{COS}(\theta)$ 的运动程序.
你可以如下编写:

使用提前计算出的每点坐标

或

使用变量公式

```
X1
X1.0173
X1.0343
.
.
X0.9824
X1
```

```
P1=0
WHILE (P1<361)
    P2=SIN(P1) +COS(P1)
    X(P2)
    P1=P1+1
ENDWHILE
```

编程计算

PMAC的 DSP具备在运动程序中执行大量的计算的能力

```
OPEN PROG 1 CLEAR
WHILE(1=1)
  IF(P1>0)
    P2=SIN(P1)+COS(P1)
    p3=2
    IF(P1>3)
      P2=SIN(P1)+COS(P1)
      P3=2
      ◆
      ◆
      ◆
      IF(P1>99)
        P2=SIN(P1)+COS(P1)
        p3=99
      ENDIF
      ◆
      ◆
      ◆
    ENDIF
  ENDIF
  X2000
  P1=P1+1
ENDWHILE
CLOSE
```

Q-变量

Q-变量是用户全局变量可用于PMAC编程中的计算.

Q-变量时坐标系下的 变量

多坐标系下使用需注意

Q变量同P变量，为48位浮点变量，共有1024个,Q0~Q1023,,每个Q变量与使用它的坐标系有关，同一个Q变量在不同的坐标系中占用不同的地址，而在PMAC的同一个地址，在不同的坐标系对应不同的Q变量。

Q变量的特殊用途：**read**命令可以将读入的值前面加上字母A到Z,分别放到Q101到Q126中。运动程序中的S表达式（主轴）将跟在后面的值放在Q127变量中。

需要学习:

- ◆ Q-变量在内存总是如何组成的
- ◆ 如何确定那个变量对您的系统有效

Q-变量内存图

内存访问通过键入 Q(序号) 改变, 根据当前的坐标系.

&1 Q0 *accesses location* **\$1400**

&2 Q0 *accesses location* **\$1600**

&7 Q0 *accesses location* **\$1580**

&8 Q0 *accesses location* **\$1780**

这种地址使多坐标系应用时, 简化内存管理。如果在8个坐标系下执行8个运动程序, 所有的程序可以使用相同的变量序号 Q0 to Q127减少繁琐, 并没有内存冲突



Q-变量 内存结构

在不同坐标系，访问同一内存地址的Q变量不相同

&1 Q0 *accesses location* \$1400

&2 Q512 *accesses location* \$1400

&7 Q640 *accesses location* \$1400

&8 Q128 *accesses location* \$1400

随着使用坐标系的增加，每个坐标系的重叠的Q变量
会减少



Q-Variable Map

PMAC Mem Loc.	Coord Sys. 1	Coord. Sys. 2	Coord. Sys. 3	Coord. Sys. 4	Coord. Sys. 5	Coord. Sys. 6	Coord. Sys. 7	Coord. Sys. 8
\$1400	Q0	Q512	Q768	Q256	Q896	Q384	Q640	Q128
...
\$147F	Q127	Q639	Q895	Q383	Q1023	Q511	Q767	Q255
\$1480	Q128	Q640	Q896	Q384	Q0	Q512	Q768	Q256
...
\$14FF	Q255	Q767	Q1023	Q511	Q127	Q639	Q895	Q383
\$1500	Q256	Q768	Q0	Q512	Q128	Q640	Q896	Q384
...
\$157F	Q383	Q895	Q127	Q639	Q255	Q767	Q1023	Q511
\$1580	Q384	Q896	Q128	Q640	Q256	Q768	Q0	Q512
...
\$15FF	Q511	Q1023	Q255	Q767	Q383	Q895	Q127	Q639
\$1600	Q512	Q0	Q256	Q768	Q384	Q896	Q128	Q640
...
\$167F	Q639	Q127	Q383	Q895	Q511	Q1023	Q255	Q767
\$1680	Q640	Q128	Q384	Q896	Q512	Q0	Q256	Q768
...
\$16FF	Q767	Q255	Q511	Q1023	Q639	Q127	Q383	Q895
\$1700	Q768	Q256	Q512	Q0	Q640	Q128	Q384	Q896
...
\$177F	Q895	Q383	Q639	Q127	Q767	Q255	Q511	Q1023
\$1780	Q896	Q384	Q640	Q128	Q768	Q256	Q512	Q0
...
\$17FF	Q1023	Q511	Q767	Q255	Q895	Q383	Q639	Q127



NEW IDEAS IN MOTION

M-变量

M-变量用于访问 PMAC 内存地址和I/O点地址

M-变量没有预先定义的含义.用户必须通过定义 M-Variables 访问PMAC 的地址.

M-变量一旦定义好后，可用于计算和判别触发

需要学习:

- ◆ M-变量如何指向内存
- ◆ 通过M-variables如何读和设定I/O点

PMAC 变量说明

PMAC M 变量

M变量称为**PMAC**卡的地址指针变量，变量范围：**M0**到**M1023**。

M变量定义格式为：**M****→ 地址，偏置地址，宽度，方向

M变量的功能为存取**PMAC**的内存和**I/O**点，**M**变量一旦定义，通过卡的后备电池或闪烁存储器可以被保存下来。**M**变量可以是**1**位的，也可以是一个字节的（**8**位）或者是一个**24**位的字，**48**位的浮点双字。



PMAC 变量说明

PMAC M 变量

M变量定义的地址前缀，可以是下列类型：

X: X内存中的**1-24**位的固定地址位

Y: Y内存中的**1-24**位的固定地址位

D: 同时占用X和Y内存的**48**位固定地址位

L: 同时占用X和Y内存的**48**位浮点地址位

DP: **32**位的固定地址位(双端口RAM使用)

F: **32**位的固定地址位(双端口RAM使用)

TWD: 多路数码开关BCD码

TWB: 多路数码开关串行I/O码

简单 M-变量定义

X: {地址}, {偏置位}, {宽度}, {格式}

Y: {地址}, {偏置位}, {宽度}, {格式}

偏置位 – 是起始的位号

宽度 – 缺省宽度为 1; 可以是 1, 4, 8, 12, 16, 20, or 24 位宽

格式- 缺省 U (无方向的); 可以设为有方向 S

定义 M-变量

M0->x: \$0000, 0, 24

指向伺服时钟

M1->Y: \$FFC2, 8, 1

指向机床输出点 1

M9->Y: \$FFC2, 8, 8

指向机床输出点 1-8

M102->Y: \$C003, 8, 16, s

指向DAC 1输出

M197->Y: \$0806, 0, 24, s

指向速率修调

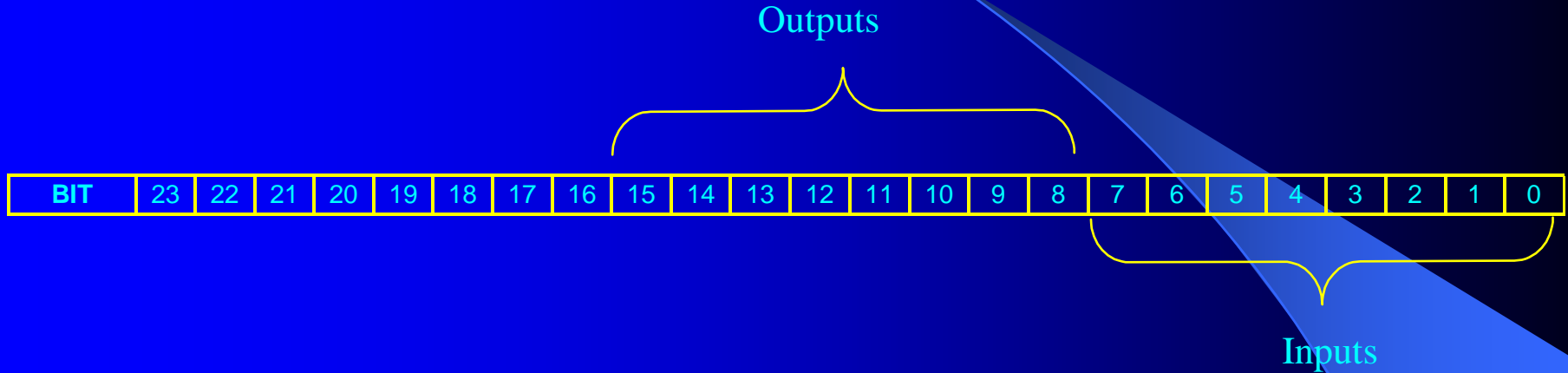
M120->Y: \$C000, 20, 1

指向轴1的回零标志信号

Delta Tau 建议的 M-变量定义在 PMAC 软件参考手册

M-变量 定义

PMAC1 memory location Y:\$FFC2 mapped to the JOPTO port: 8 inputs and 8 outputs



```
M9->Y:$FFC2,8,8      ;JOPTO port output word
M19->y:$FFC2,0,8      ;JOPTO port input word
```

使用 M-变量

M9=1 机床输出1有效

M9=45 机床输出1有效1,3,4,6 打开并且关闭
机床输出 2,5,7,8

45=00101101 二进制

PMAC 变量说明

PMAC M 变量

M变量应用实例：

M0→y:\$FFC2, 8, 1 指向卡的输出1

M9→y:\$FFC2, 8, 8 指向卡的输出1—8

当指令M0=1 打开卡的输出1

当指令m9=45 因为45=00101101，所以打开输出1， 3， 4， 6， 而关闭输出2， 5， 7， 8。

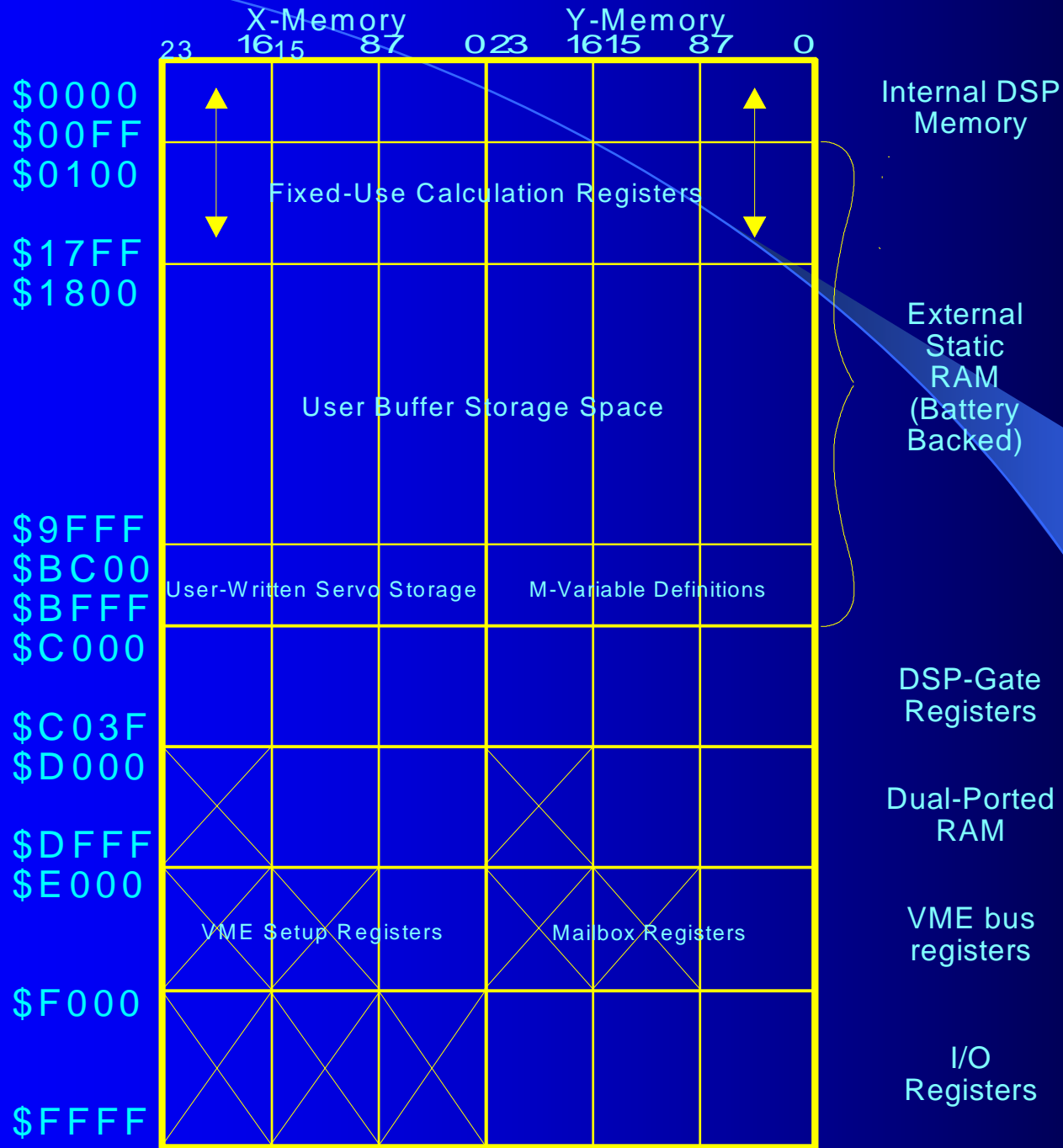
如M变量定义如下：

M102→Y:\$c003, 8, 16, s 指向卡的模拟量1（DAC1）输出，16位宽，有方向性

M162→D:\$002B 指向卡的#1电机的实际位置。

以上两个M变量M102 和M162, 用户可以编写程序，指令M102=**, 来指令卡的模拟量输出，或者键入M162，来查询卡的#1电机的实际位置。

PMAC Memory Mapping



PMAC卡的相关软件

.执行软件：

PE3 PMAC卡DOS环境的执行软件

PEWIN PMAC卡WINDOWS环境的执行软件,目前有16位的**PEWIN17**,和32位的**PEWIN32**

.开发软件：

PTALK：VB or VC下的控件，可实现pc机的上位软件与PMAC卡之间的通讯。

PCOMM32：windows下的动态链接库函数，可使用VB， VC ， C++ ， BLAND 开发。

PCOMM16：功能同PCOMM32,只是16位的DOS或windows3.1的开发环境。

PEWIN执行软件

执行程序是一个工具，用来帮助同PMAC卡进行通讯，和编译程序，以及诊断PMAC卡的故障。

执行程序可以使你存取所有PMAC的特性，例如，你可以：

- 给PMAC发送在线指令
- 监视位置 速度 跟随误差
- 监视PMAC卡的电机，坐标系和全局的状态
- 监视，修改和查询PMAC卡的变量
- 执行电机的伺服环调整和自动调整
- 备份和恢复PMAC卡的所有设置文件
- 使用帮助文件发现PMAC卡的问题

PMAC执行软件PEWIN功能

与PMAC卡建立通讯， 调试， 诊断错误。

给PMAC发在线（on-line）指令。

监视PMAC卡的电机， 坐标系及系统状态。

监视， 修改和查询PMAC卡的变量

备份及恢复PMAC的总体设置

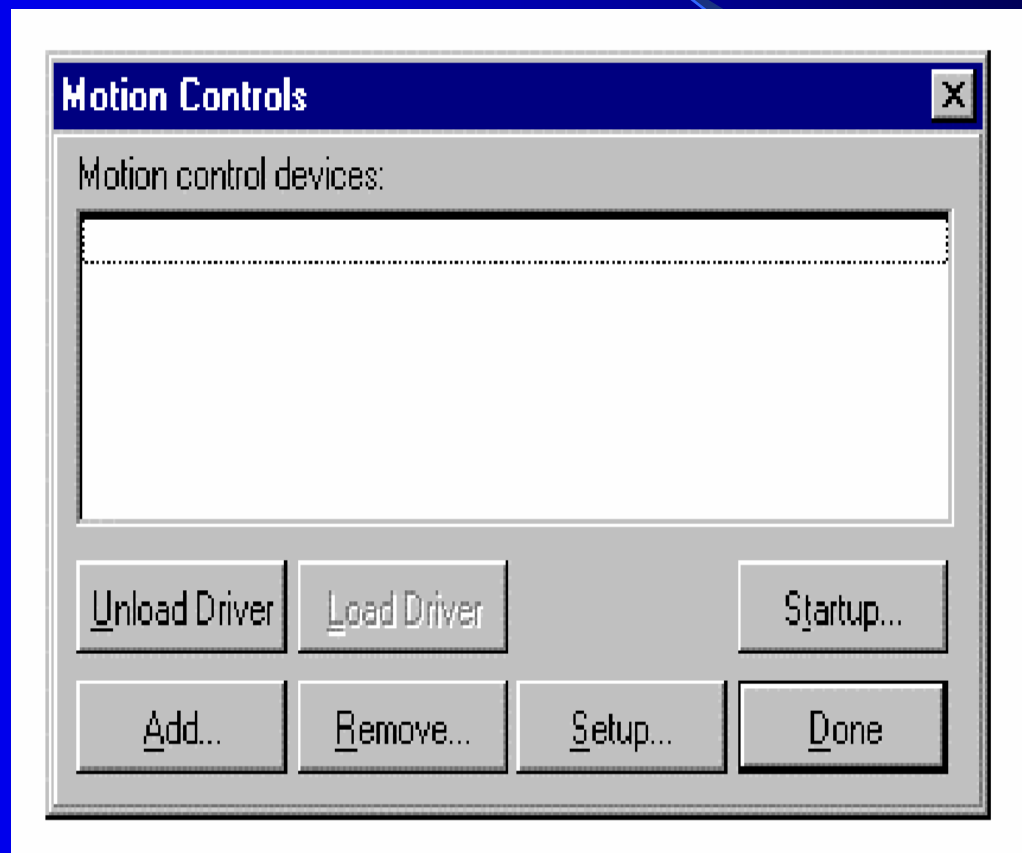
该软件需先安装， 安装完毕后， 可运行此软件， 该软件会提示您先与PMAC建立通讯， 通讯建立后， 进入其执行界面， 该软件为菜单式友好界面， 用户可根据菜单提示， 按步执行对卡的操作。



E-MOTION

PMAC多轴运动控制卡

PEWIN32 的MOTION.EXE建立与PMAC的通讯

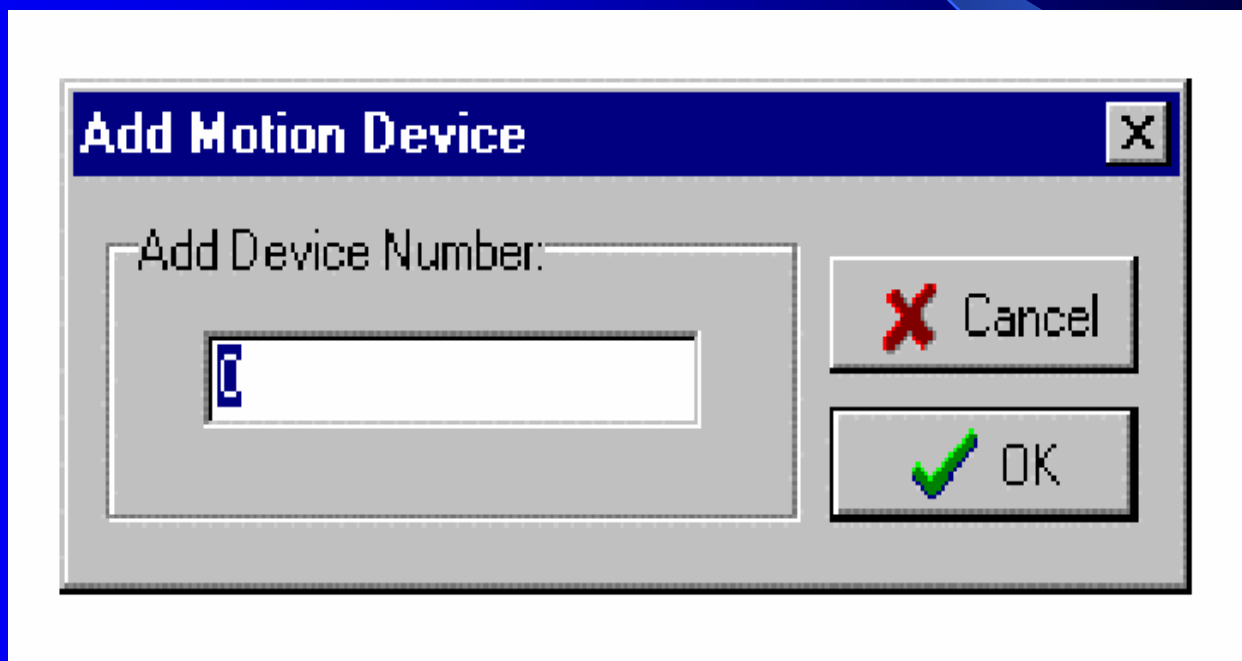


NEW IDEAS IN MOTION

E-MOTION

PMAC多轴运动控制卡

PEWIN32 的MOTION.EXE建立与PMAC的通讯



NEW IDEAS IN MOTION

E-MOTION

PMAC多轴运动控制卡

PEWIN32 的MOTION.EXE建立与PMAC的通讯

PMAC Device(0) Configuration

PMAC Win32 Communications Library
Version: 1.0.8.0
Copyright © 1994-98 Delta Tau Data Systems, Inc.

Device Location:

☒ In PC Bus ☐ VME Bus ☐ Serial Port

Port Address: 0x210

Interrupt Addr: None

DPRAM Addr: None

Host Computer: VMIC-IO Works C

Mail Box Base Addr: 7FA000

DPRAM Base Addr: 700000

Please use hexadecimal format

Address Modifier: 39, A24

IRQ Level: 7

IRQ Vector: A1

Serial Port: COM2

Baudrate: 38400

Parity: No Parity

OK Cancel Advanced

PMAC卡有关电机设置的I 变量

以下为PMAC卡的重要I变量，是用户经常会修改到的
以下*=1-8电机

I*00: 电机的禁能/使能控制，**I*00=1** 表示使能

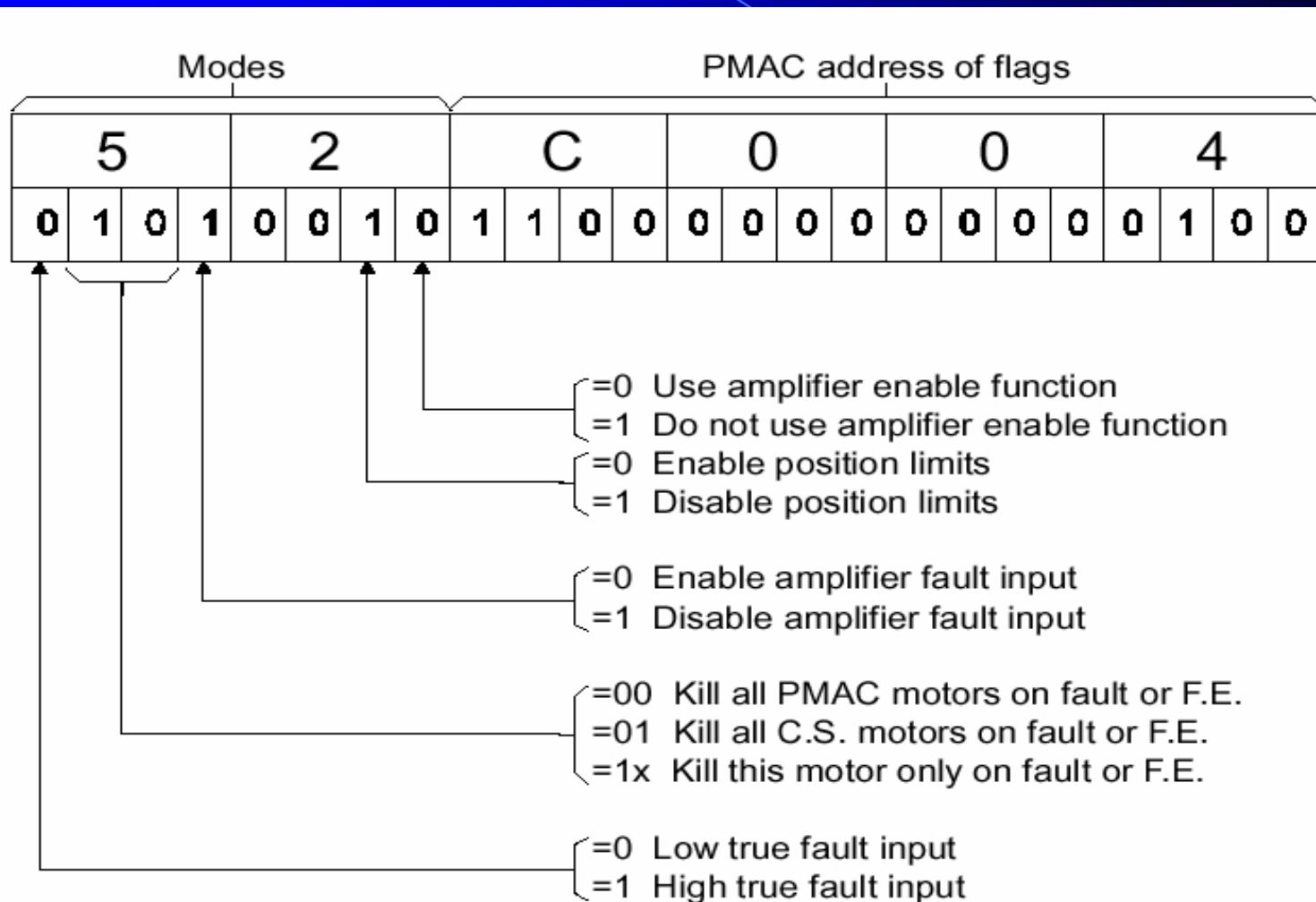
I*02: 电机的模拟量输出地址，该地址对于1型卡和2型卡应为特定的

I*03=位置反馈地址

I*04=速度反馈地址

对于同一个反馈元件，**I*03=I*04**

I*25 :电机的标志和方式变量



I*29:DAC模拟量输出的微调。此参数可以调整模拟量输出的偏差

I*30—I*35: PMAC卡的PID调整的相关变量

I*11 I*12:设置电机的跟随误差的报警和极限值，单位为cts

I*13 I*14:设置电机的正负软限位。单位为cts.

I*23: 电机的回零速度及方向，速度单位为cts/ms

I*26:电机的回零偏差，单位为cts1/16

I*69: 模拟量的输出范围，缺省为：20480—6.25V

I900: 对于1型卡，为反馈信号的极性和倍频，PMAC卡最大可以设置到4倍频

VALUE	MEANING
0	pulse and direction CW
1	x1 quadrature decode CW
2	x2 quadrature decode CW
3	x4 quadrature decode CW
4	pulse and direction CCW
5	x1 quadrature decode CCW
6	x2 quadrature decode CCW
7	x4 quadrature decode CCW

1902: PMAC的flag信号的捕捉方式，主要用于回零的使用

SETTING	MEANING
0	Software Control
1	Rising edge of CHCn (third channel)
2	Rising edge of Flag n (as set by Flag Select)
3	Rising edge of [CHCn AND Flag n] -- Low true index, high true Flag
4	Software Control
5	Falling edge of CHCn (third channel)
6	Rising edge of Flag n (as set by Flag Select)
7	Rising edge of [CHCn/ AND Flag n] -- Low true index, high true Flag
8	Software Control
9	Rising edge of CHCn (third channel)
10	Falling edge of Flag n (as set by Flag Select)
11	Rising edge of [CHCn AND Flag n/] -- High true index, low true Flag
12	Software Control
13	Falling edge of CHCn (third channel)
14	Falling edge of Flag n (as set by Flag Select)
15	Rising edge of [CHCn/ AND Flag n/] -- Low true index, low true Flag

I903:选择PMAC的flag信号的形式，可设为回零，也可设为+-限位信号

SETTING	MEANING
0	HMFLn (Home Flag n)
1	-LIMn (Positive Limit Signal n)
2	+LIMn (Negative Limit Signal n)
3	FAULTn (Amplifier Fault Signal n)

PMAC2型卡，电机的变量设置从**I100** 到 **I899**定义相同，只是**I900**之后含义不同,其中较重要的为：

I9n0: 功能同1型卡的**I900**号参数

I9n2: 功能同1型卡的**I902**号参数

I9n3: 功能同1型卡的**I903**号参数

I9n6: 2型卡输出信号的方式选择

0 = Outputs A & B are PWM; Output C is PWM

1 = Outputs A & B are DAC; Output C is PWM

2 = Outputs A & B are PWM; Output C is PFM

3 = Outputs A & B are DAC; Output C is PFM

PMAC 调整

PMAC的 伺服算法必须设置以适用于系统电机和放大器的工作.

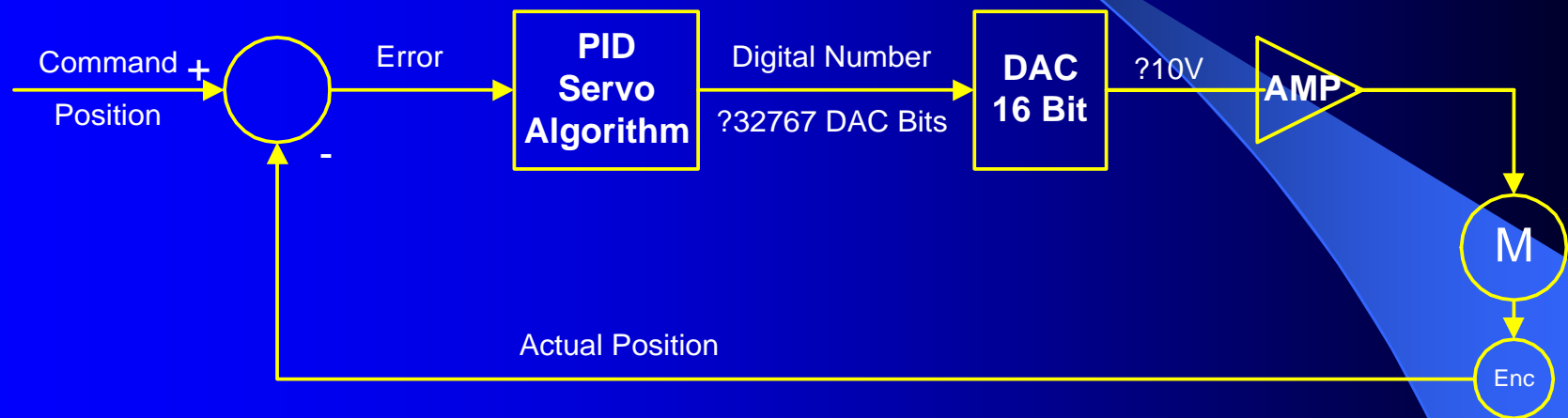
设置可通过I变量的调整, 获取适当的PID增益数值称作调整

需要学习:

- ◆ PID 增益的含义
- ◆ PID 伺服环的功能
- ◆ 如何设置系统 PID 增益 .

典型 P.I.D 伺服环

$$\begin{array}{ccccc} \text{Following} & = & \text{Commanded} & - & \text{Actual} \\ \text{Error} & & \text{Position} & & \text{Position} \\ \text{跟随误差} & & \text{指令位置} & & \text{实际位置} \end{array}$$



P (proportional gain比例增益)

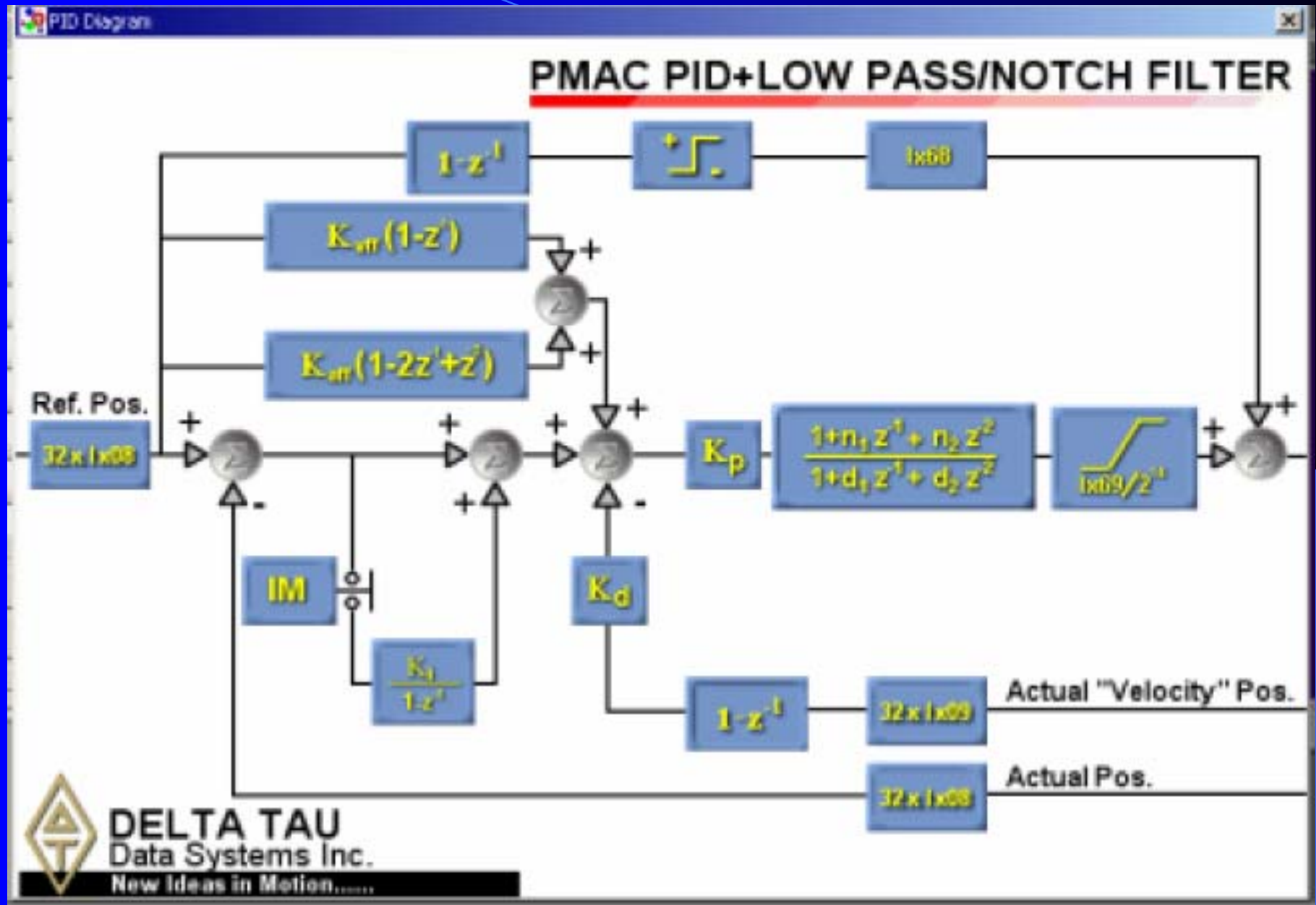
I (integral gain积分增益)

D (derivative gain微分增益)

Spring弹性, 弹力

Shock Absorber
减震, 吸收

PMAC典型P.I.D 伺服环



PMAC卡的电机PID调整

PID调整的前提

1. PEWIN执行软件与卡已建立通讯
2. 伺服系统已连接正常,电机必须为闭环状态
3. 使电机闭环的指令为: #nj/
4. 使用#n?,的在线指令来查询电机的状态字

812. 则表示闭环状态

85. 则表示开环状态

E12. 则表示电机在硬限位

A12. 则表示电机在软限位

012. 则表示电机未使能

与PID 调整有关的I 变量

I*30 : 伺服环的比例增益, 影响系统的刚性。

I*31 : 伺服环的微分增益, 提供给系统阻尼。

I*32 : 伺服环的速度前馈增益, 减小系统的跟随误差。

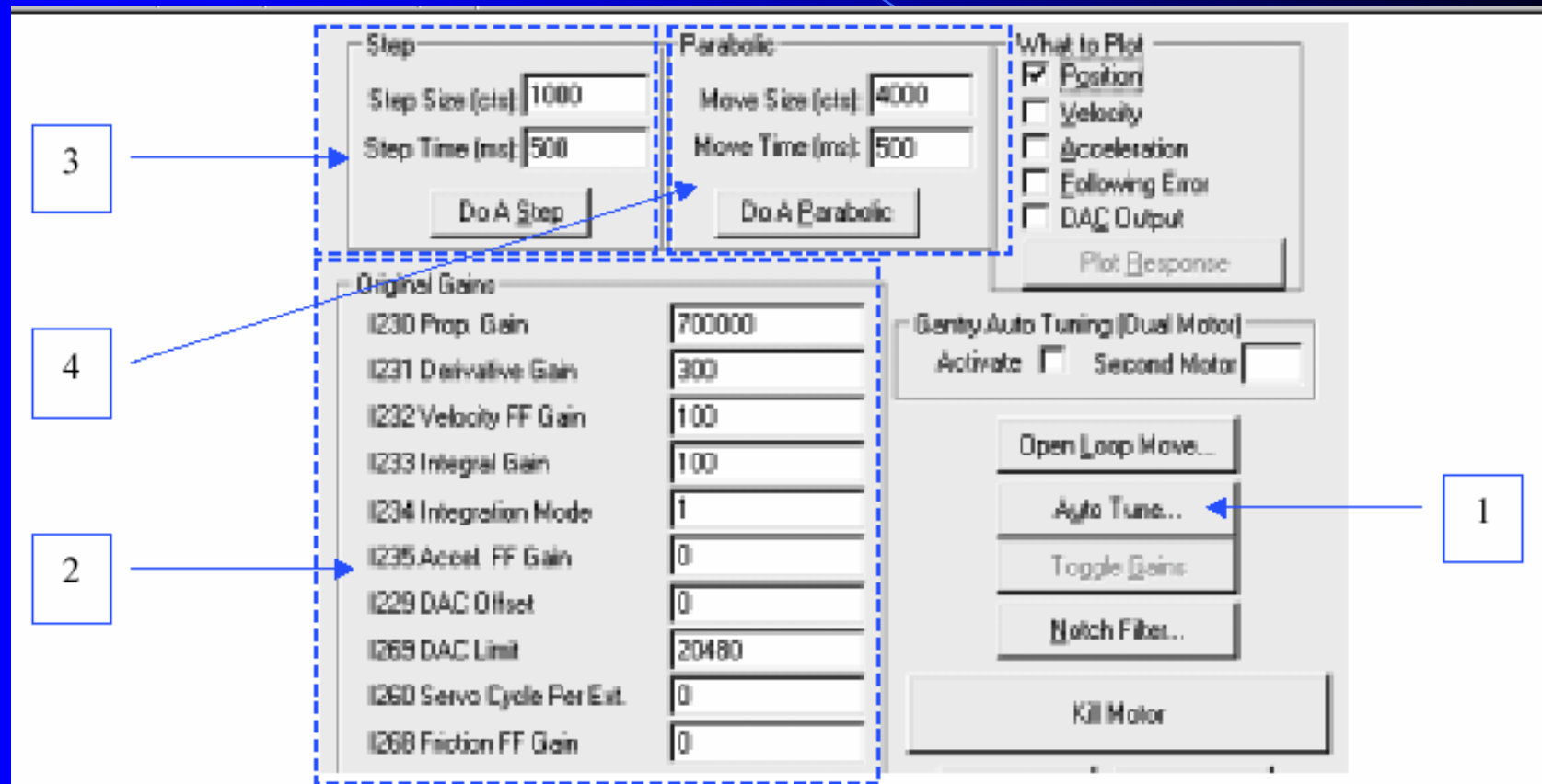
I*33 : 伺服环的积分增益, 减小系统的稳态误差。

I*35 : 伺服环的加速度前馈, 消除系统在加减速时的跟随误差。

I*68 : 摩擦增益, 减小由于摩擦产生的跟随误差。

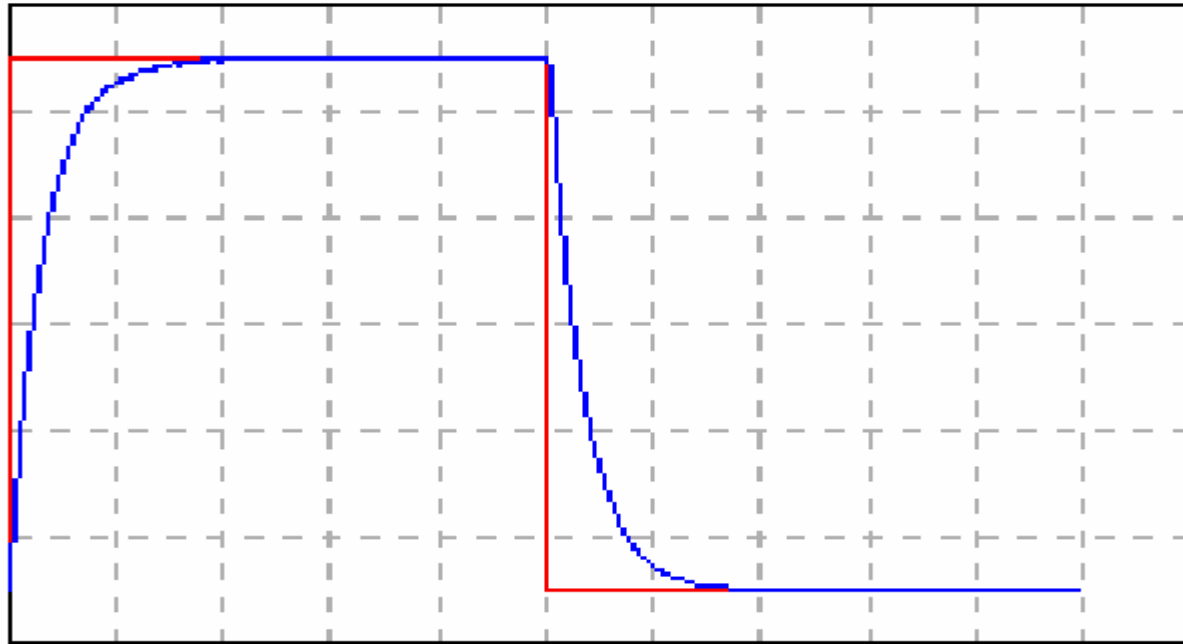


PMAC 卡PID调整的界面



) Select the Auto Tune feature. This is the first interaction to find a starting bandwidth parameter.

PMAC卡PID调整的STEP 曲线分析

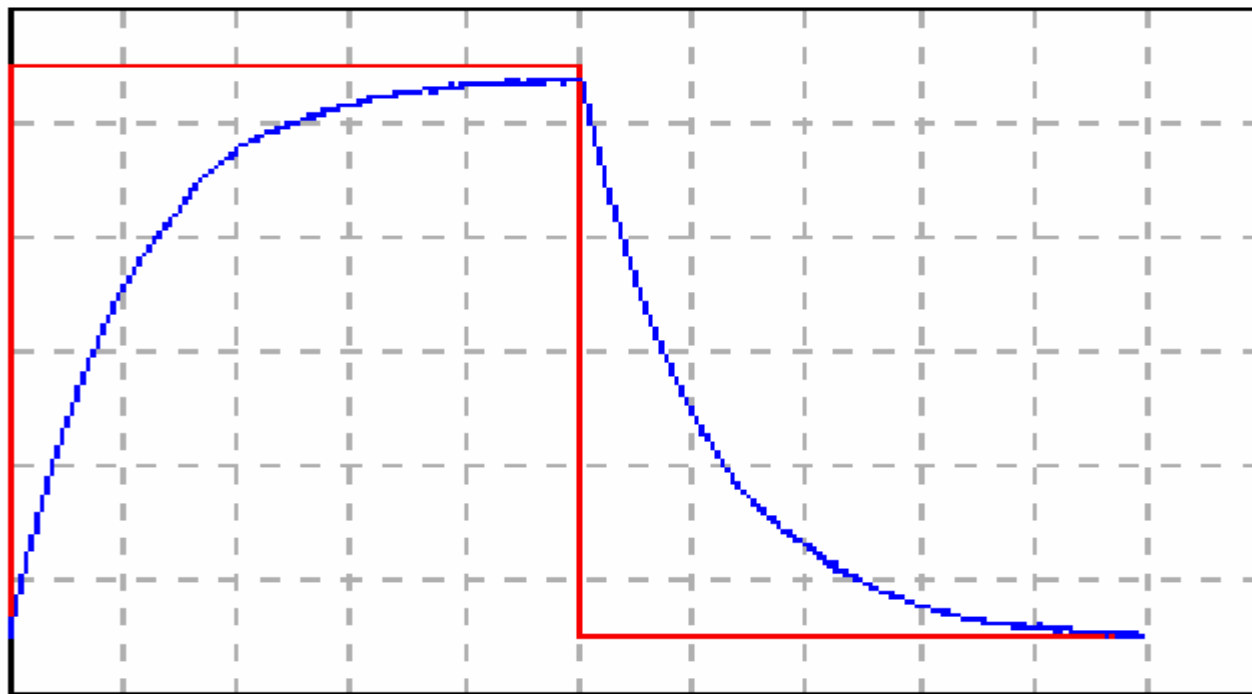


Position Offset

Cause: friction or constant force / system limitation

Fix: Increase K_I (Ix33) and maybe use more K_P (Ix30)

PMAC卡PID调整的STEP曲线分析

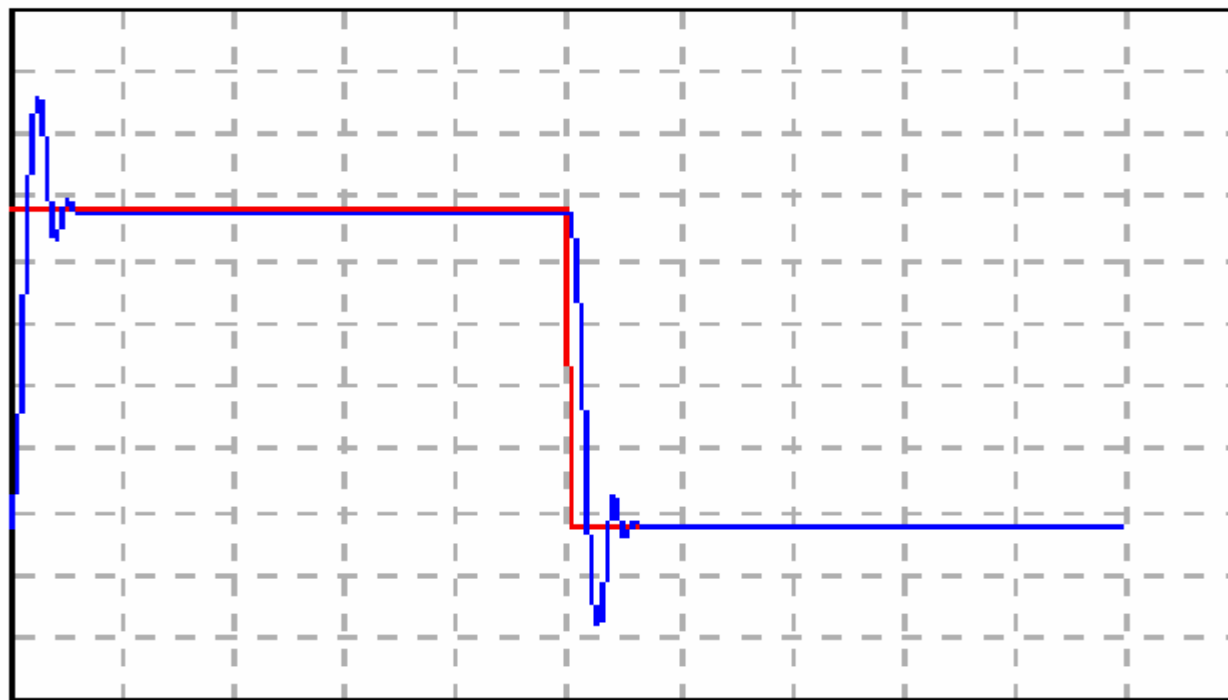


Sluggish Response

Cause: Too much damping or too little proportional gain

Fix: Increase K_P (Ix30) or decrease K_D (Ix31)

PMAC卡PID调整的STEP 曲线分析



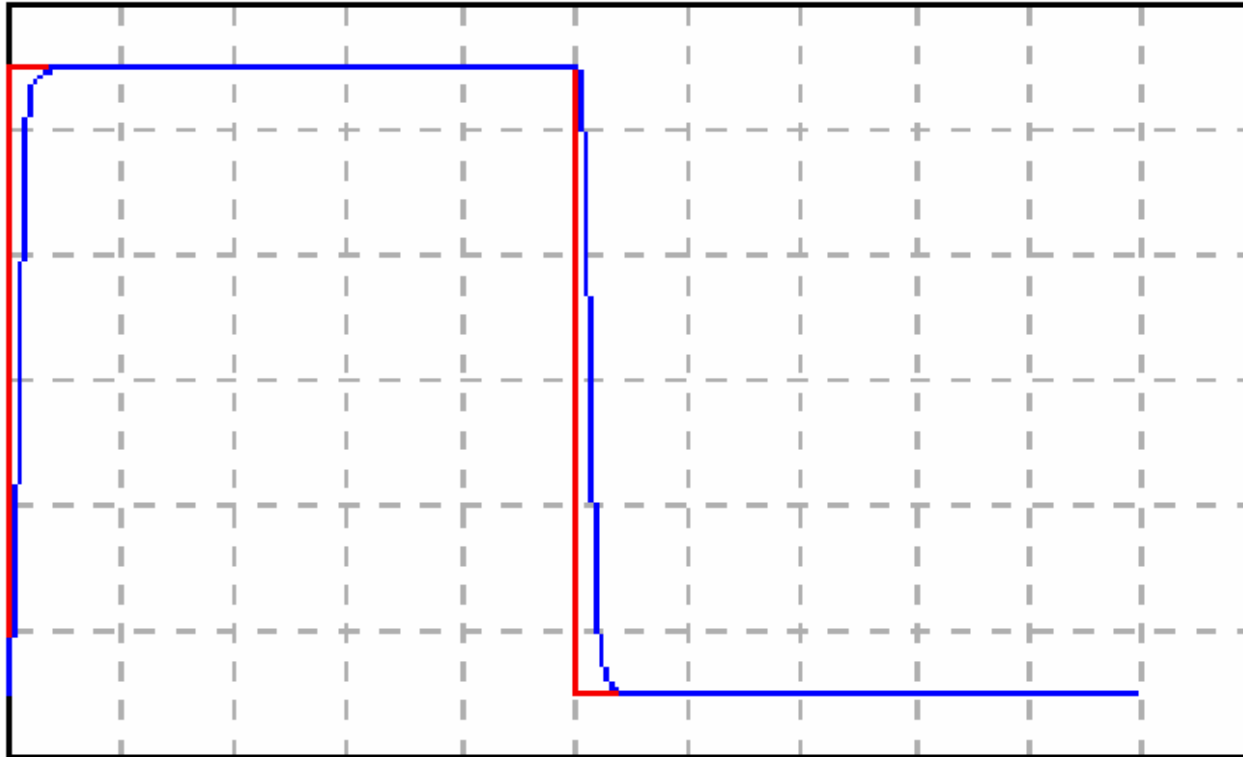
Overshoot and Oscillation

Cause: Too little damping or too much proportional gain
Fix: Decrease K_P (Ix30) or increase K_D (Ix31)

E-MOTION

PMAC多轴运动控制卡

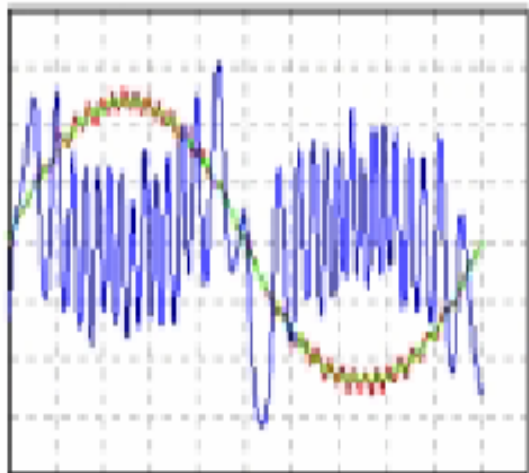
PMAC卡PID调整的STEP曲线分析



Ideal Case

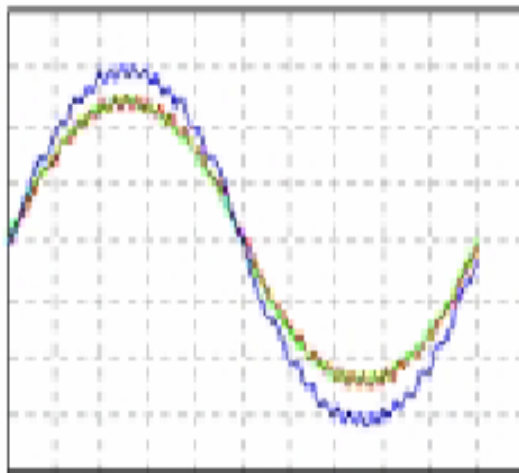
The motor closely follows the commanded position

PMAC 卡PID调整的Parabolic 抛物线曲线分析



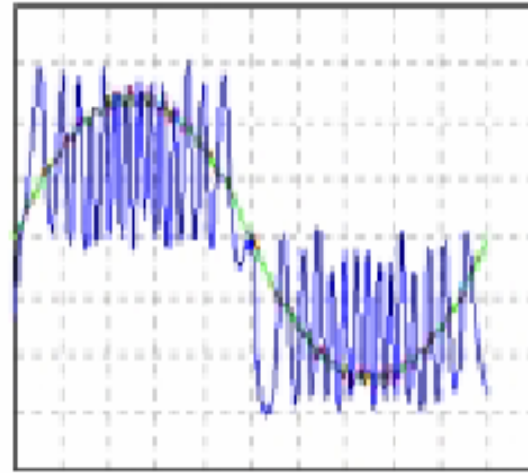
Ideal Case

The following error is reduced at minimum and is concentrated in the center, evenly along the move



High vel \ FE correlation

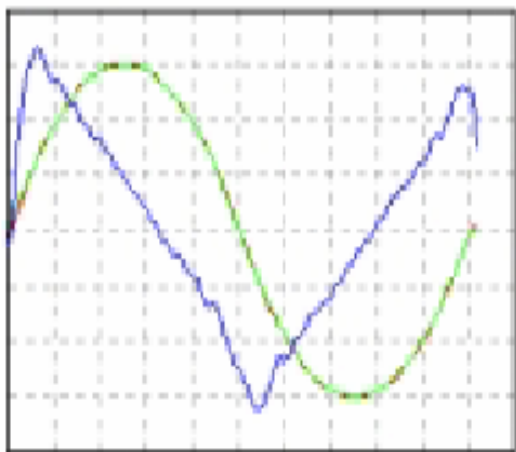
Cause: damping
Fix: Increase K_{vel} (lx32)



High vel \ FE correlation

Cause: friction
Fix: Increase Integral gain (lx33) or Friction Feedforward (lx68)

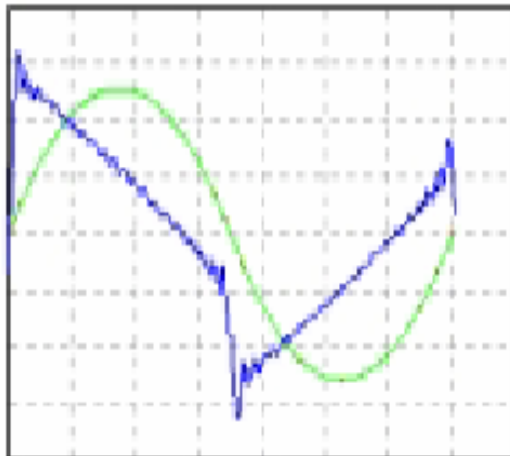
PMAC卡PID调整的Parabolic 抛物线曲线分析



High acc \ FE correlation

Cause: Integral lag

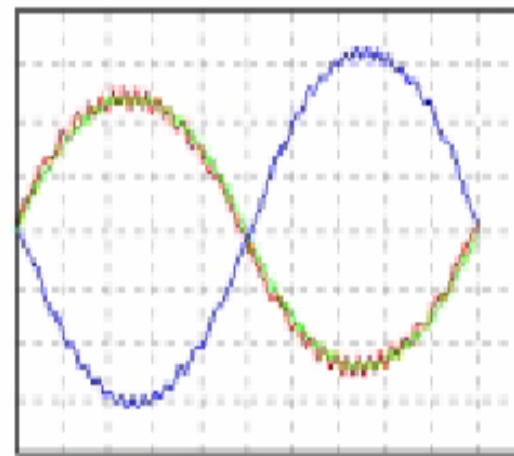
Fix: Increase K_{aff} (lx35)



High acc \ FE correlation

Cause: Physical system limitations

Fix: Use less sudden acceleration

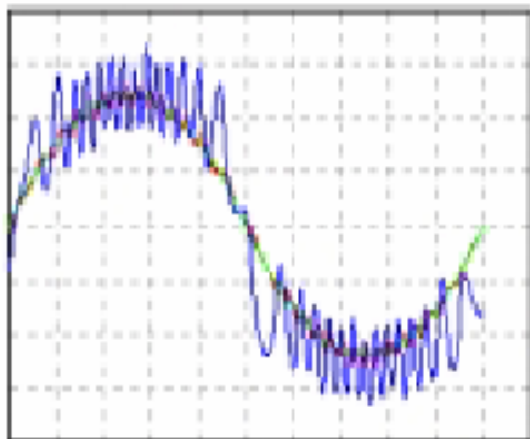


Negative vel \ FE correlation

Cause: Too much velocity FF

Fix: Decrease K_{vel} (lx32)

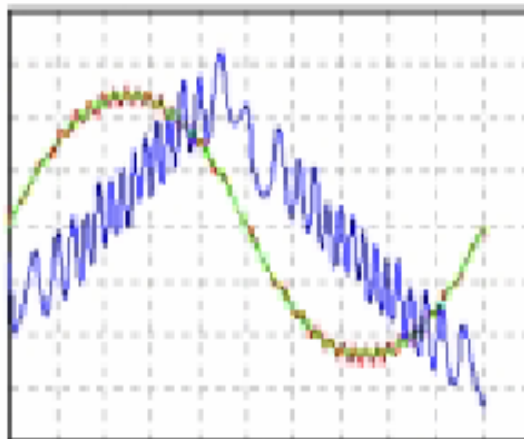
PMAC卡PID调整的Parabolic 抛物线曲线分析



High vel \ FE correlation

Cause: damping and friction

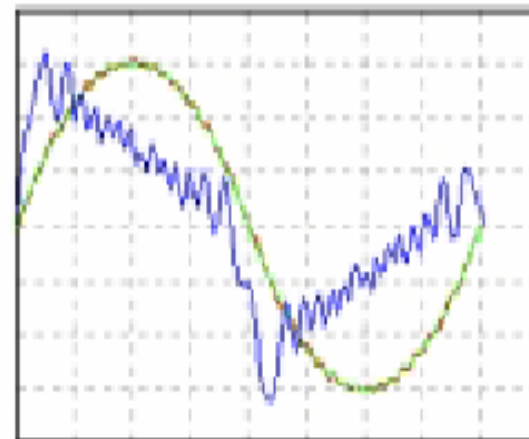
Fix: Increase K_{vel} (Ix32)



High acc \ FE correlation

Cause: Too much acc FF

Fix: Decrease K_{aff} (Ix35)



High vel\FE and acc\FE correlation

Cause: Integral lag and friction

Fix: Increase K_{aff} (Ix35)

PMAC PID自整定功能

PMAC Executive Software执行软件提供PID AUTO-TUNING自动调整功能，易于调整PID。

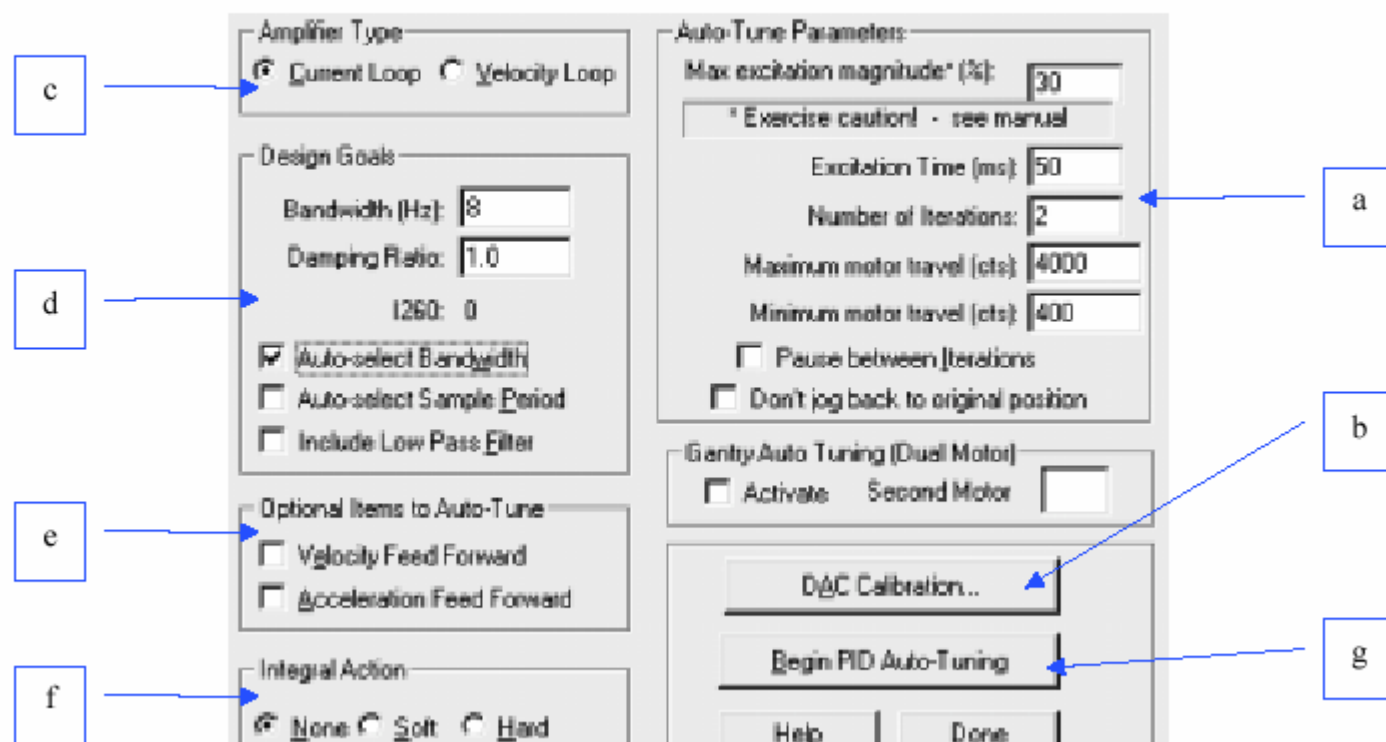
The Auto-Tuning 自动调整功能是电机运动,估算响应, 根据所需响应自动计算需要的增益值.

PMAC 的auto-tuning PID自动调整适用于从未使用过 PMAC Executive Auto-tuner的用户. 出于安全, 对新客户建议使电机空载, 来熟悉自动调整功能. 一旦掌握之后可以带负载调整.

自动调整根据电机的快速运动将计算电机的动态特性然后计算电机的PID增益. 如果在需要的负载方式进行自动增益, 电机不能发生剧烈运动, 说明不能进行自动增益调整, 如果是这种情况你可以手动调整PID增益.

PMAC卡的PID自动调整

1) Select the Auto Tune feature. This is the first interaction to find a starting bandwidth parameter.



PMAC卡的PID自动调整步骤

1. 执行DAC校验，保证DAC输出位0%时，电机无零漂，执行**Calibrate DAC** 在**PID auto turning** 选项中。
2. 将电机与负载脱离。
3. 选择**PID auto turn** ,会出现一对话框。
4. 选择自动定义带宽，确认不选择速度前馈，加速度前馈，且不作积分，按下**PID auto turning** 按钮，PMAC 会得到一个较低的带宽值。
5. 将较低的带宽值增加2或3倍，并选择0.7倍阻尼比，取消自动选择带宽，再执行**Begin PID auto turning**.
6. 选择速度前馈和加速度前馈和积分方式，选择**Begin PID auto turning** 按钮。
7. 增加带宽，选择**Begin PID auto turning** 按钮，不断增加带宽直到电机开始震动，再减小带宽，再做**PID auto turning** ,直到电机不再振动为止。

注：PMAC的PID自动调整最好在电机空载时进行。

PMAC卡的手动开环指令

1.开环指令

指令格式: **#n O**** 回车, **代表最大模拟量输出的百分比。

最大模拟量的数值由变量**I*69**来决定, 最大为**+10V**。

例如: **#1O10**回车, 表示第一个电机以最大模拟量的**10%**输出模拟量

2. 开环指令的作用

.验证模拟量输出通道

.伺服电机转动后, 可检验编码器的反馈方向, 保证**O+**指令时, 编码器计数为正, **O-**时, 编码器计数递减, 否则可修改**I900 (I9n0)**等编码器极性参数。



电机手动JOG运动方式

PMAC卡的手动闭环指令

JOG方式是PMAC卡的最简单闭环运动形式

JOG的控制指令为PMAC卡的在线指令

Jogging 可用于简单定位或以常速运动，象在主轴的应用

需要学习的:

- ◆ 怎样JOG指令电机
- ◆ 怎样改变JOG的速度
- ◆ 怎样改变JOG的加速度

在线JOG指令

J+ - Jog 正向连续运动

J- - Jog 反向连续运动

J/ - Jog 运动停止 (或者闭环)

J= - Jog 返回前一个JOG的 (或上一个编程) 位置

J={常数} - Jog 到指定的位置 (单位是脉冲counts)

J^{常数} - Jog 从当前实际位置运动指定的距离

J:{常数} - Jog 从当前指令位置运动指定的距离

J=* - Jog 运动一个变量 (PMAC 的内存地址)

J={常数或 *}^{常数} - Jog 直到触发

变量 Ix19-Ix22 用于JOG控制. 变量可以在运动中改变, 但只有到下一个JOG指令时才有效



在线JOG指令例子

#1J+ - Jog #1电机正向运动

#1J=1000 #2j+ - Jog #1电机正向运动到1000cts 并且 jog #2电机正向运动

I122=30 - Set 电机#1 jog 速度为 30 counts/msec

I220=50 - Set 电机#2 jog 加速时间为 50 msec

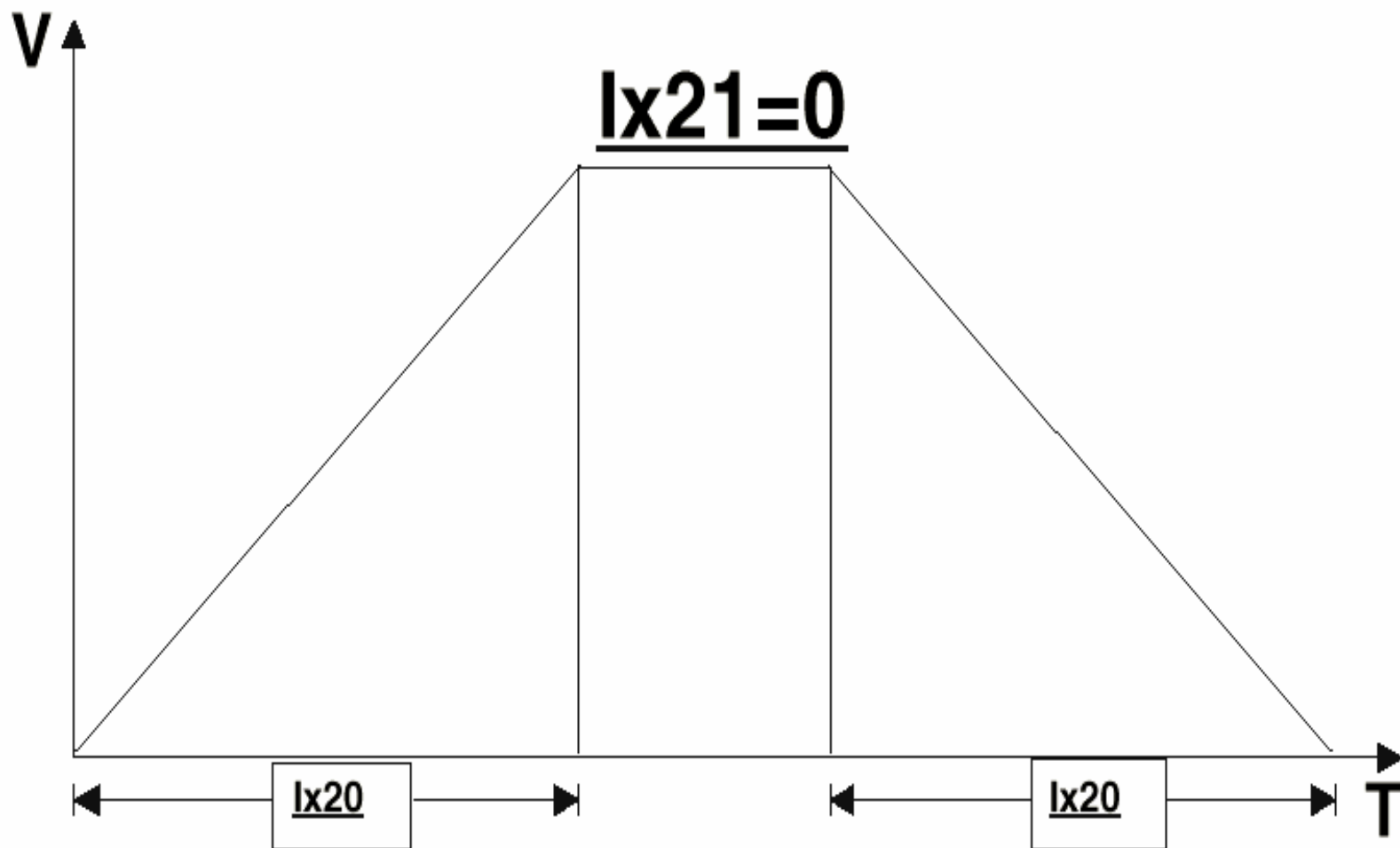
注：电机如果定义在坐标系下，且正在执行程序时，是不能对此电机进行JOG运动



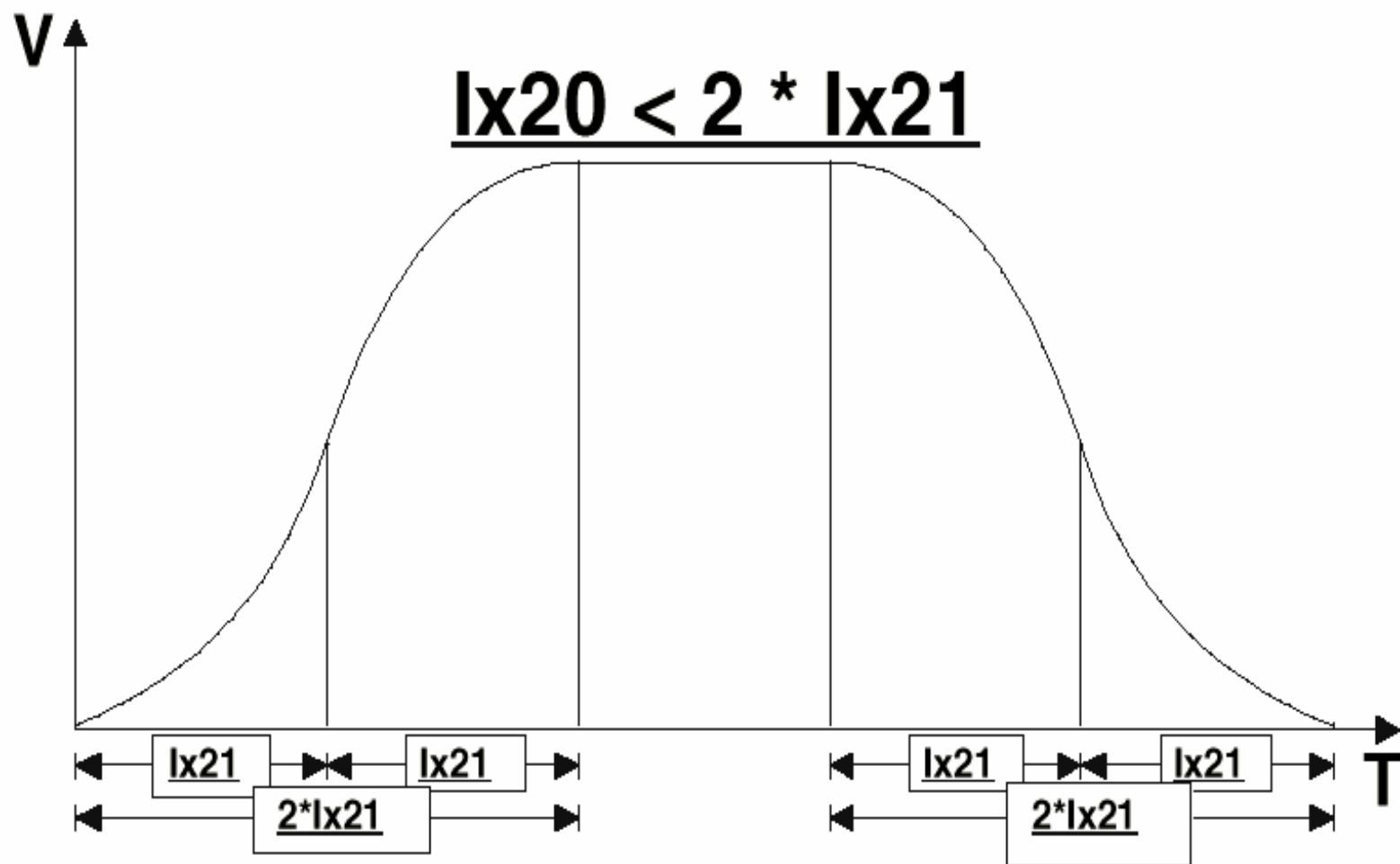
电机 x Jog 变量

- Ix19** 最大允许的 **Jog** 加速度
(单位: cts/msec²; 浮点形式)
可以修调
如果 Ix20=0 and Ix21=0, 使用**Ix19**的设定
- Ix20** Jog 加速时间: (单位: msec; 整形)
如小于2倍的**Ix21**, 则无效
- Ix21** Jog S-曲线加速时间 (单位: msec; 整形)
可以修调TA(**Ix20**)
- Ix22** Jog 速度 (单位: cts/msec; 浮点)
绝对值

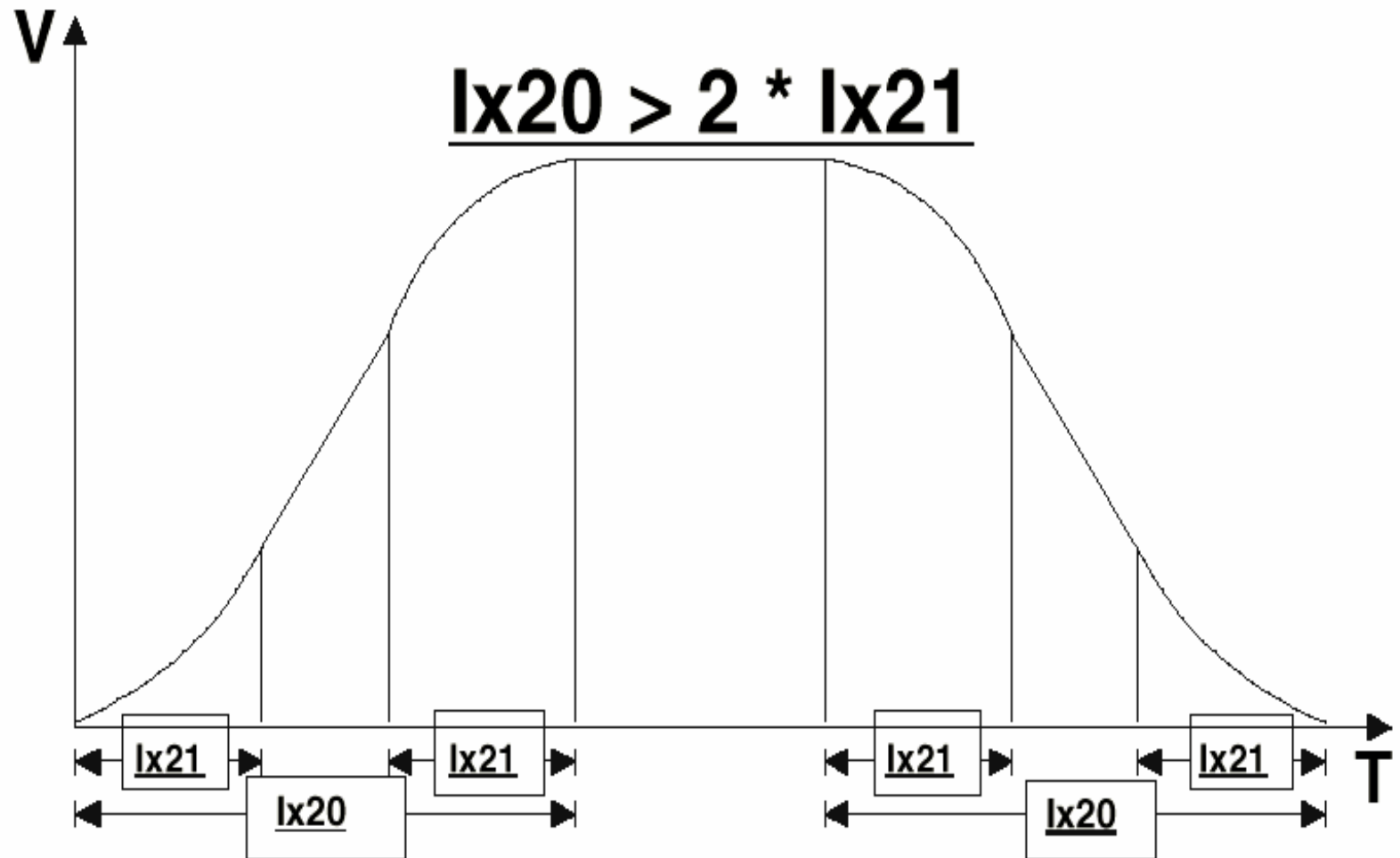
PMAC JOG 直线加速轨迹曲线



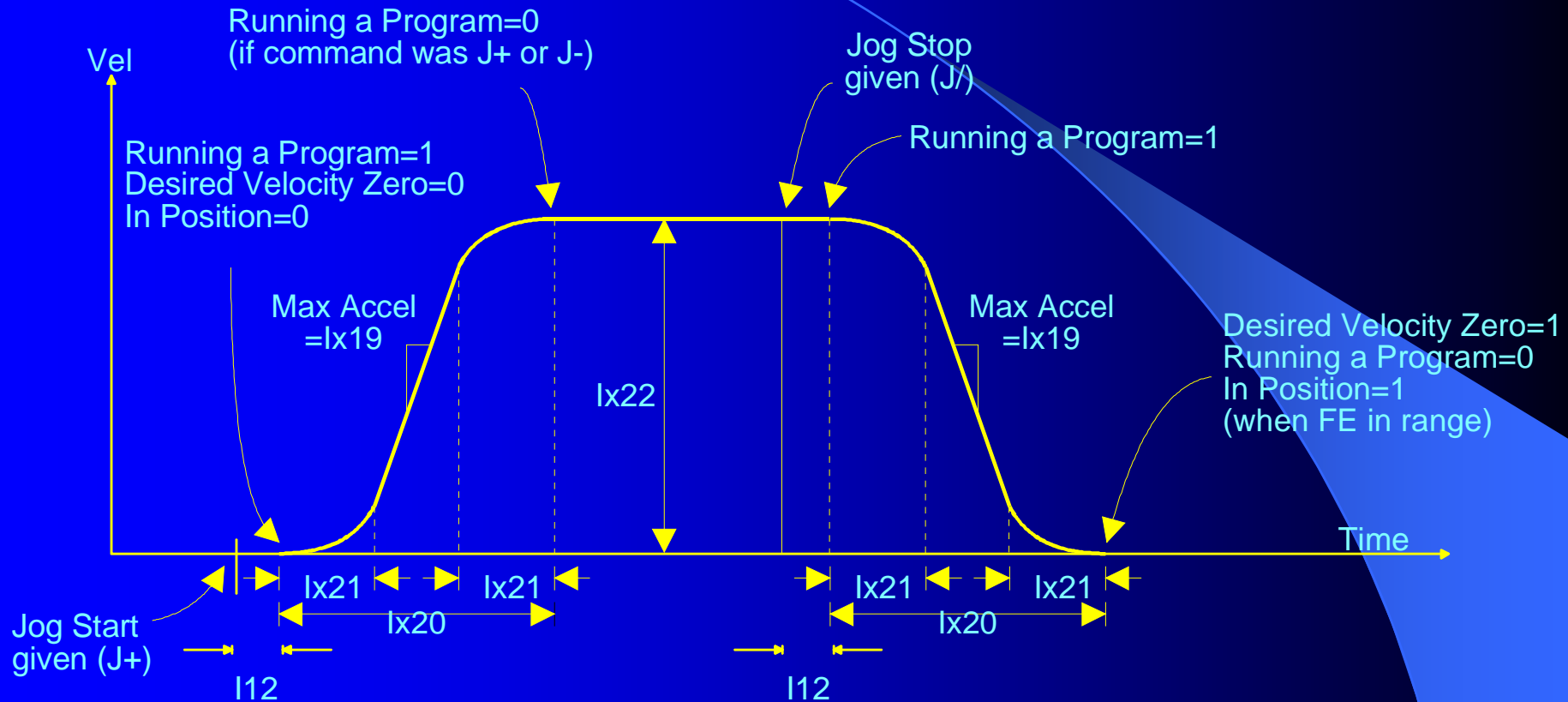
PMAC JOG S 曲线加速轨迹曲线



PMAC JOG S 曲线与直线加速轨迹曲线



Jog Move Trajectory



Note: If the jog command is not of indefinite length (not J+ or J-) then the "Running a Program" status bit will be high for the entire jog move

Note: Rate of acceleration limited by Ix19 - can override Ix20 and Ix21

电机回零运动

回零运动可通过一个运动程序或在线指令实现.

回零触发由用户选择

在上电和复位时 大多数机床需要回零以确定编程的零位

需要学习的:

- 怎样使用在线指令回零
- 怎样改变回零的速度和方向
- 怎样设定回零触发.



在线回零指令

HM - 作回零运动

HMZ - 将当前位置作为零位

在线指令举例:

#1HM - 电机#1执行回零运动

#1HM #2HM #3HM - 同时开始回零运动

注: 当一个电机在一个坐标系下正在执行程序时,
在线回零指令无效

PMAC 回零变量

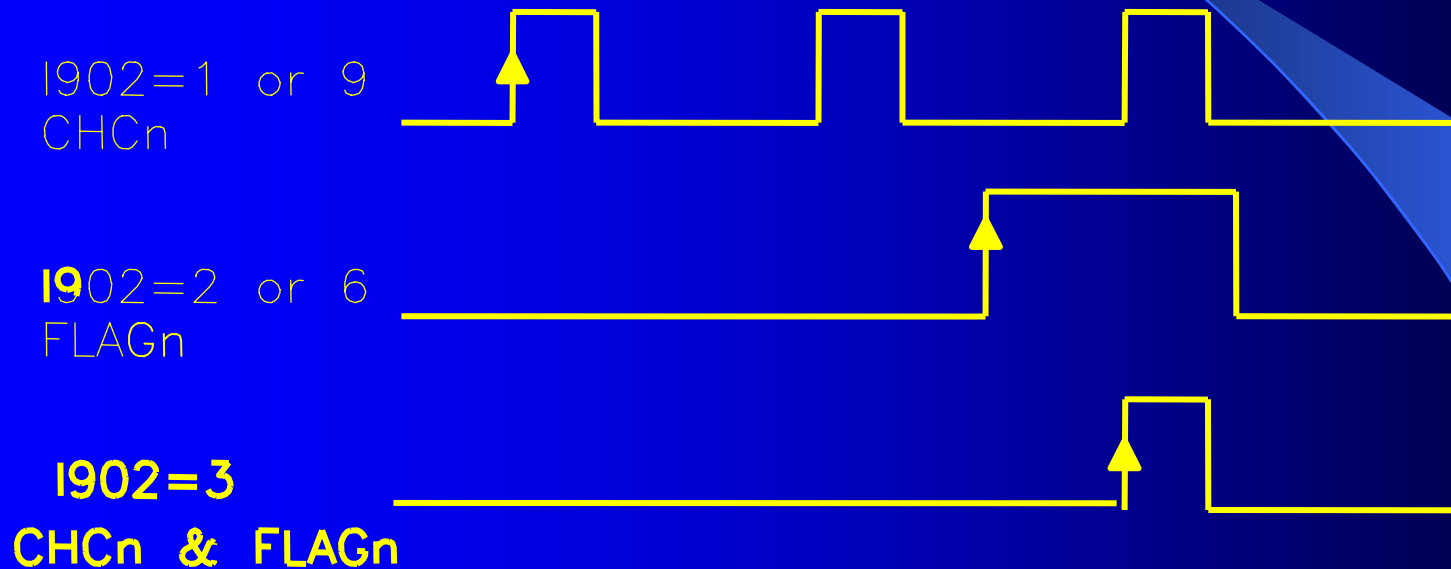
- Ix19** 最大允许的 **Jog** 加速度
(单位: cts/msec^2 ; 浮点形式)
可以修调
如果 $\text{Ix20}=0$ and $\text{Ix21}=0$, 使用**Ix19**的设定
- Ix20** 回零加速时间: (单位: msec; 整形)
如果小于2倍的**Ix21**, 此参数无效
- Ix21** 回零 S-曲线加速时间 (单位: msec; 整形)
可以修调TA(**Ix20**)
- Ix23** 回零速度 (单位: cts/msec ; 浮点)
- Ix26** 回零偏置值 (单位: $1/16 \text{ cts}$)
距离为从零点开始



MOTOR x 回零触发

I902, I907, ..I977: 编码器位置捕捉控制 (*I9n2* 对 *PMAC2*)

变量设定决定那个信号或几个信号的组合作为回零和寄存的触发



I903, I908, ... , I978: 标志信号的选择控制 (*I9n3* 对 *PMAC2*)
决定那个 FLAGn 用于位置捕捉

0: HMFLn

2: +LIMn*

1: -LIMn*

3: FAULTn*

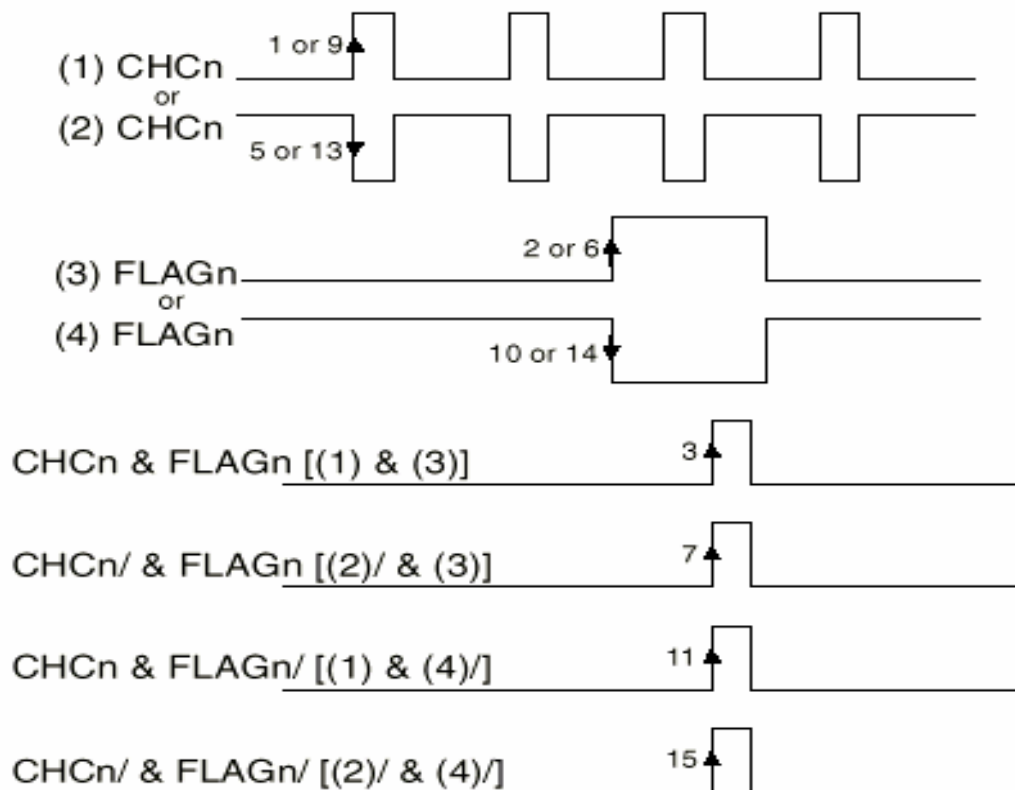
* must disable normal function with Ix25

PMAC 电机回零指令及相关 I 变量

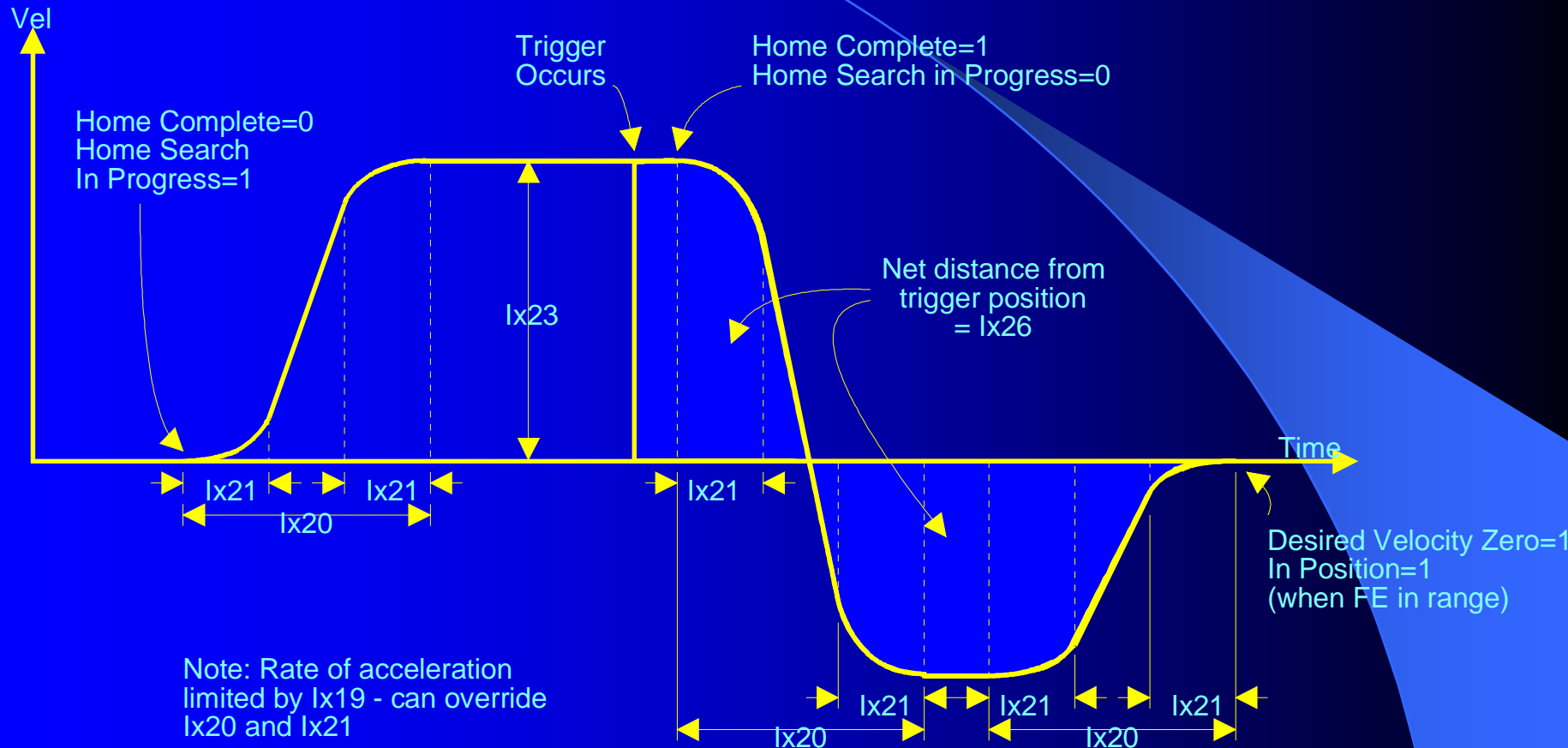
回零相关的 I 变量:

I902 (#1为例) (I9n2 PMAC 2 型卡):

回零信号的捕捉方式, 可设置如下:



Homing Search Move Trajectory



Important Setup I-Variables

SETUP PROGRAM FUNCTION	I-VARIABLES CONFIGURED
<i>Communications Variables</i>	I3, I4, I9
<i>Motor Activated</i>	I100
<i>Flags & Modes Variable</i>	I125
<i>DAC Configuration</i>	I102, I169
<i>Position Encoder Configuration</i>	I103, I900
<i>DAC Trimming</i>	I129
<i>"Velocity" Encoder Configuration</i>	I104
<i>Encoder Relative Scaling</i>	I109
<i>Preliminary Tuning Variables</i>	I111, I112, I169 I130, I131, I132
<i>Jogging Setup Variables</i>	I119, I120 I121, I122
<i>Position Capture Configuration</i>	I902, I903
<i>Homing Setup Variables</i>	I123, I126

PMAC 卡如何编写运动程序

1、建立坐标系

一个运动程序需要运行在一个坐标系下.

根据运动程序中需要运动的电机，将需要运动的电机组合在一个坐标系下

需要学习：

- PMAC轴的含义
- 坐标系的定义
- 如何定义坐标系

PMAC 轴的特性

允许定义的轴: X,Y,Z,U,V,W,A,B,C

X,Y,Z: 传统上的主要直线轴

- ◆ 矩阵轴的定义
- ◆ 矩阵轴的转换
- ◆ 圆弧插补
- ◆ 切削半径补偿

PMAC 轴的特性 (续)

U,V,W: 传统上的第二直线轴

◆ 矩阵轴的定义

A,B,C: 传统的旋转轴

(A 围绕 X, B 围绕 Y, C 围绕 Z)

◆ 位置翻转 (Ix27)



坐标系定义

坐标系定义如下，坐标系下的轴分配到该坐标系下

&1

#1->X

#2->Y

#3->Z

&2

#4->20X

#6->25.4Y



坐标系定义 (续)

目的是组合电机在一个坐标系下

- 组合的电机将使许多电机运行一个运动程序. 一些应用, 如等高线, 需要电机精确同步运动. 一个运动程序可以计算许多电机组合的运动轨迹, 然后高精度的执行所需轨迹的运动.

应用举例:

加工中心, 机器人, 龙门结构, X-Y 平台



定义坐标系 (续)

将每个轴定义在不同的坐标系下

- 如果PMAC控制多个机床, 每个独立的运动需要一个运动程序. 对于此情况, 你可以设置多个坐标系使一个机床独立于其他的机床.

例如:

包装线, 机器人搬运, 主轴



多轴定义

(&->坐标系; #->电机; X->轴)

&1

#1->X

&2

#2->X



该定义是允许的.

两个电机在不同的坐标系下定义为X轴

&1

#1->X

#2->X



该定义是允许的.

电机将同样作X-轴轨迹,
象龙门机床



NEW IDEAS IN MOTION

多轴定义 (续)

&1
#1->X
#1->Y



该定义不允许.
一个电机不能在一个程序中执行不同的运动轨迹.
第一个轴的定义将被第二个轴的定义代替

&1
#1->X
&2
#1->X



该定义不允许.
当运行两个程序时, 一个电机将接收不一致的指令.
第二个坐标系的定义将被拒绝.



坐标系定义

- ◆ PMAC 支持8轴 (控制环)
- ◆ 每个电机可以有一个输出 (非换相) 或者两个输出 (换相)
- ◆ 8个电机可以定义如下:

构成8个独立的系统
构成一个8轴的系统
系统可以任意组合



坐标系定义 (续)

- ◆ 每个坐标系可以定义速率轴“FEEDRATE”，其他轴可以是坐标系下的时间轴。
- ◆ 分配给坐标系下的电机以“轴定义表达式”，使轴与电机匹配，并且带有放大和平移的功能：

例： $\#1 \rightarrow 10000X + 5000$



坐标系定义 (续)

- ◆ 笛卡尔坐标轴可以是直线轴电机的组合
- ◆ **PMAC**卡可如下定义轴矩阵：

$$\text{\#1} \rightarrow A_{11}X + A_{12}Y + A_{13}Z + B_1$$

$$\text{\#2} \rightarrow A_{21}X + A_{22}Y + A_{23}Z + B_2$$

$$\text{\#3} \rightarrow A_{31}X + A_{32}Y + A_{33}Z + B_3$$

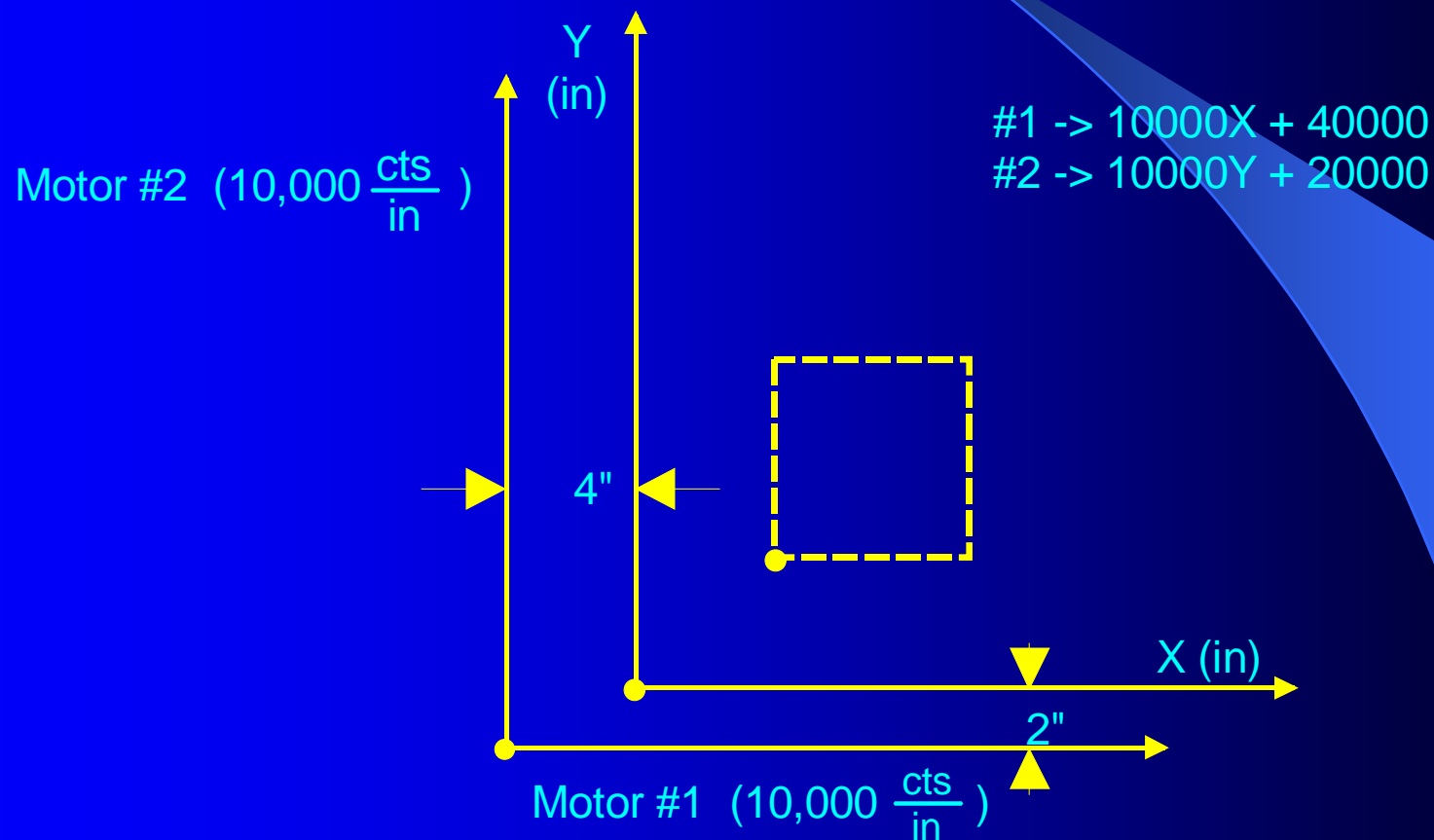
这里 A_{11} 到 A_{33} 是比例因子 而 B_1 到 B_3 是偏移量

此定义形式可以实现对坐标系的：

平移、放大、旋转、校正

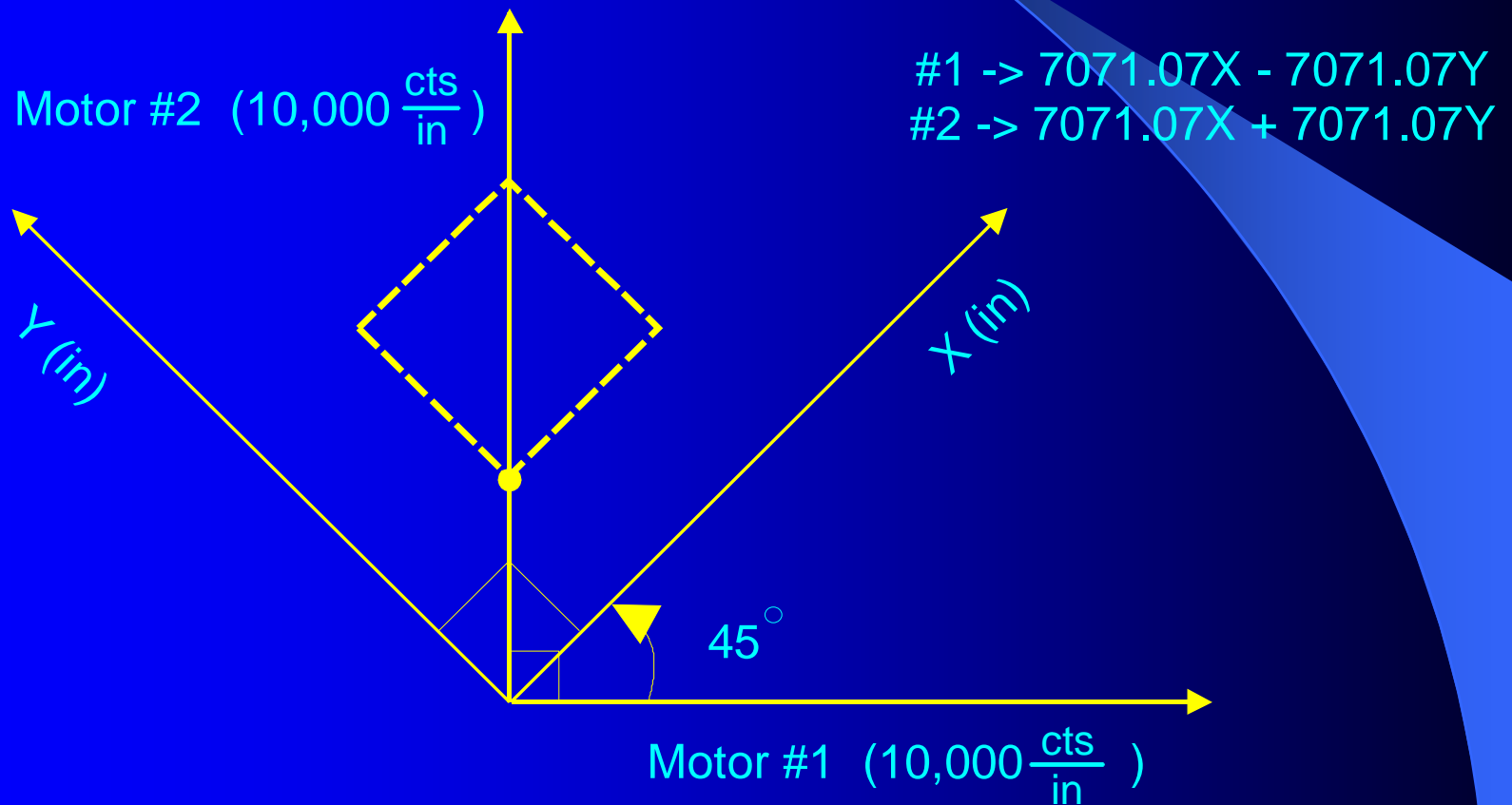
坐标系定义

◆ 比例和平移



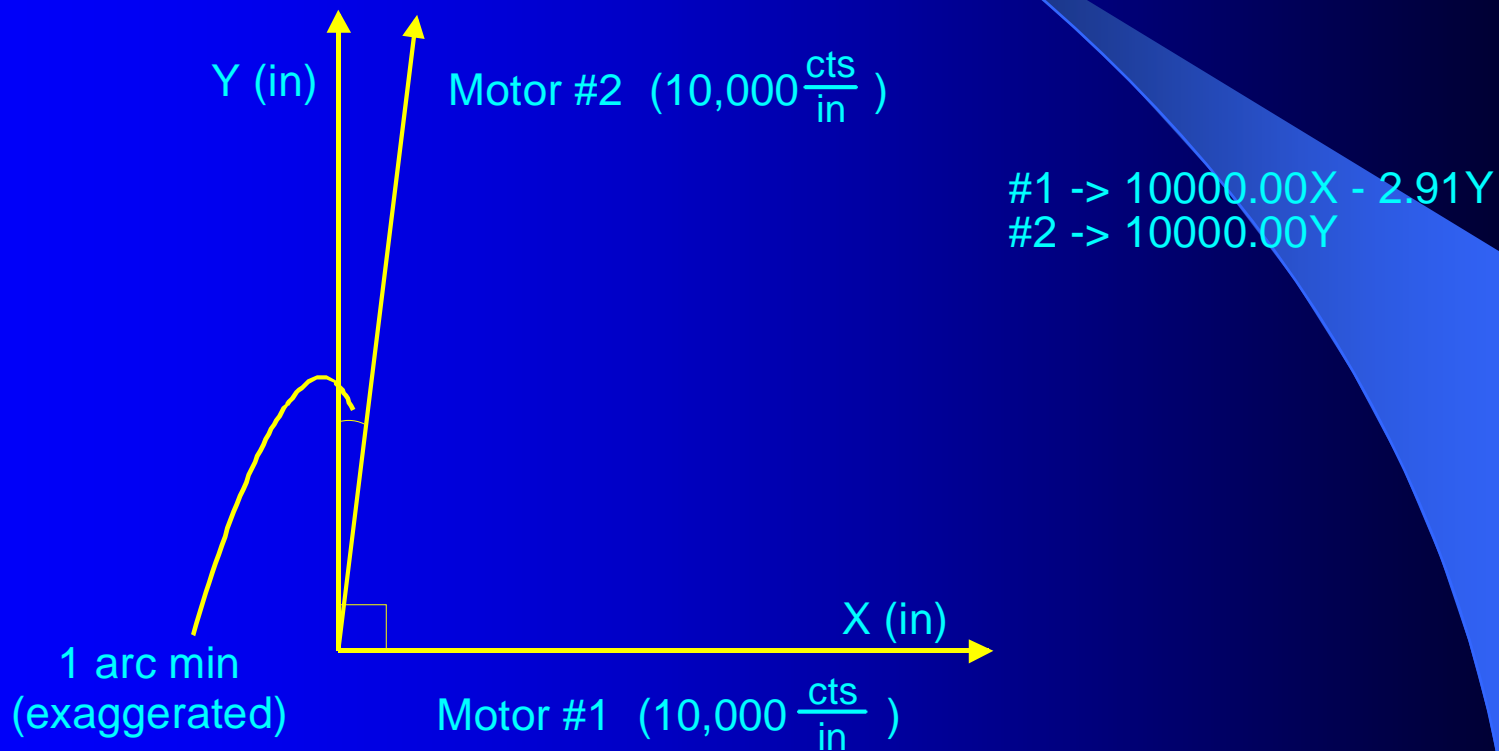
坐标系定义(续)

◆ 比例和旋转



坐标系定义(续)

◆ 正交校正



2、编写运动程序

运动程序是指令坐标系的轴作定位和轮廓运动等.

运动程序是多数PMAC卡应用的核心.

需要学习:

- 什么是运动程序
- 如何写一个运动程序
- 如何运行一个运动程序



运动程序

PMAC 编程

- ◆ 在某一时刻执行一个运动，执行运动所需的所有计算
- ◆ 在坐标系下运行
- ◆ 一个程序可以在多个坐标系下同时运行
- ◆ 一个程序可以在不同的坐标系下运行
- ◆ 一个坐标系一段时间只能运行一个运动程序



运动程序(续)

开始一个程序

- ◆ 指向一个坐标系，用在线指令: **&n**
- ◆ 用在线指令指向程序: **Bn**
- ◆ 用在线指令运行: **R** or **<CTRL-R>**

停止程序

- ◆ 用在线指令指向坐标系: **&n**
- ◆ 用在线指令: **Q, S, A, or <CTRL-Q>, <CTRL-S>, <CTRL-A>, <CTRL-K>**



PMAC 运动程序表达式

- 运动指令

X1000 Y2000 Z3000

U(P1*3.14159) V(20*SIN(Q6))

DWELL, DELAY

- 模态指令

ABS, INC, FRAX, NORMAL

LINEAR, RAPID, CIRCLe_n, SPLINEn, PVT

TA, TS, TM, F

- 变量赋值

{variable} = {expression}

PMAC运动程序表达式(续)

- 逻辑控制表达式

N, O, GOTO, GOSUB, CALL, RETURN
G, M, T, D (special CALL statements)
IF, ELSE, ENDIF, WHILE, ENDWHILE

- 辅助表达式

COMMAND, SEND, DISPLAY
ENABLE PLC, DISABLE PLC



PMAC 逻辑操作

used in Motion Programs and PLCs

■ Logic Operators 逻辑操作符

&	(bit by bit AND)
	(bit by bit OR)
^	(bit by bit Exclusive OR)

Comparators 比较符

=	(equal to)
!=	(not equal to)
>	(greater than)
!>	(not greater than; less than or equal to)
<	(less than)

■ Functions 函数

SIN, COS, TAN, ASIN, ACOS, ATAN, ATAN2,
SQRT, LN, EXP, ABS, INT

PMAC 程序结构

建立一个程序:

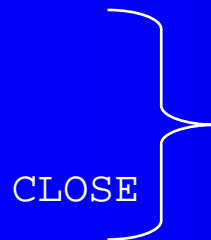
```
CLOSE
```

```
DEL GAT
```

```
&n
```

```
#{motor}->{axis scaling}{axis}
```

```
OPEN PROG n CLEAR
```

 Motion program statements

从内存中删除一个程序

```
OPEN PROGn CLEAR
```

```
CLOSE
```

例：A Simple Move一个简单的运动

This example shows how to program a simple move on the program specifies how to do the move, then commands the move.

***** Set-up and Definitions *****

```
DEL GAT ; Erase any defined gather buffer
&1      ; Coordinate System 1
CLOSE   ; Make sure all buffers are closed
#1->X   ; Assign motor 1 to the X-axis - 1 program unit
        ; of X is 1 encoder count of motor #1
```

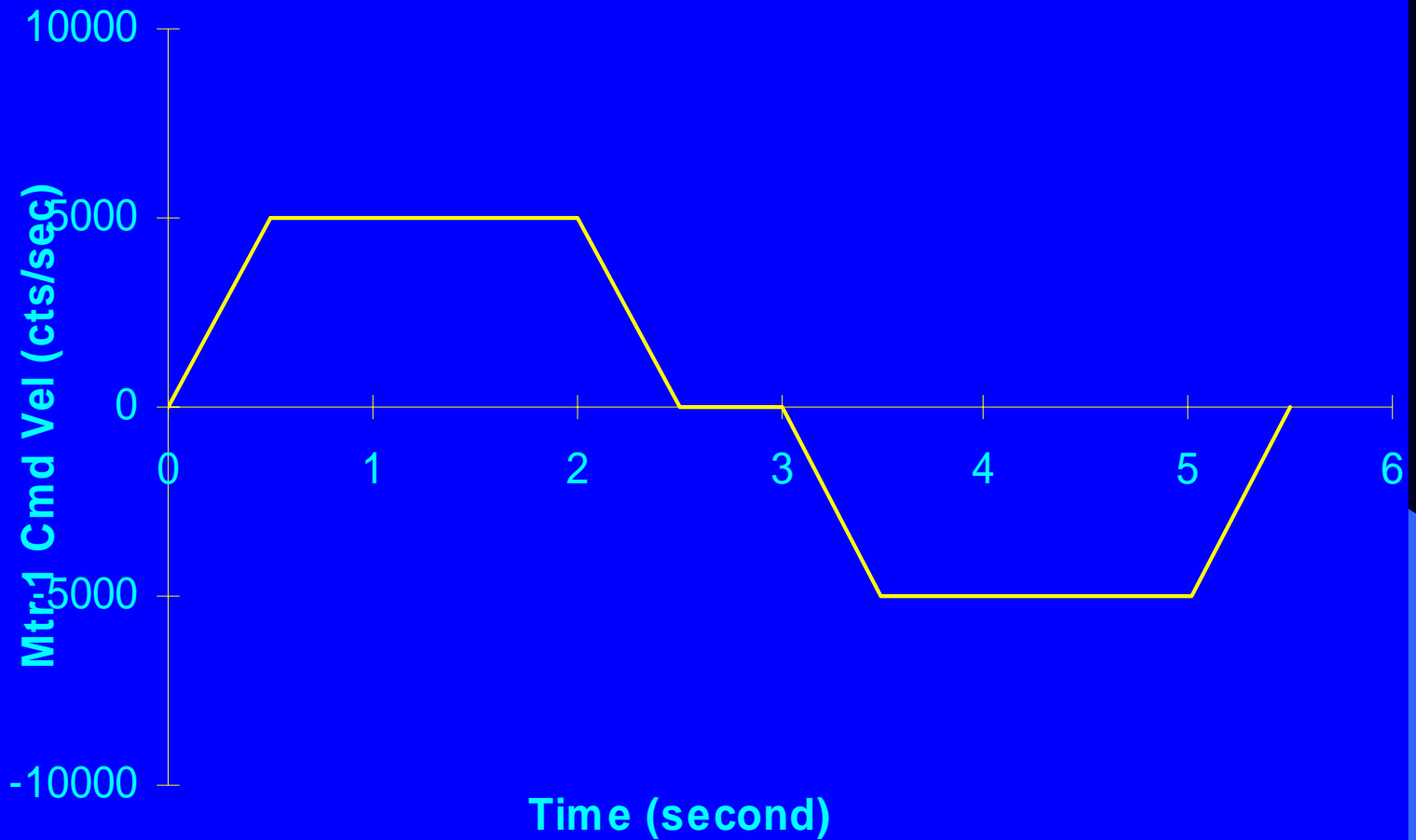
***** Motion Program Text *****

```
OPEN PROG 1 ; Open buffer for program entry, Program #1
CLEAR       ; Erase existing contents of buffer
LINEAR      ; Blended linear interpolation move mode
ABS         ; Absolute mode - moves specified by position
TA500       ; Set 1/2 sec (500 msec) acceleration time
TS0         ; Set no S-curve acceleration time
F5000       ; Set feedrate (speed) of 5000 units(cts)/sec
X10000      ; Move X-axis to position 10000
DWELL500    ; Stay in position for 1/2 sec (500 msec)
X0          ; Move X-axis to position 0
CLOSE       ; Close buffer - end of program
```

To run this program:

```
&1 B1 R ; Coord. System 1, point to Beginning of Program 1, Run
```


Example : A Simple Move



E-MOTION

PMAC多轴运动控制卡 PLC编程

PMAC卡除了运行运动程序，还可以运行 PLC程序.

PMAC PLC 程序使能 PMAC卡执行一个硬件的PLC任务，
并且独立于运动程序

学习什么：

如何写一个PMAC卡的PLC程序

如何运行PMAC 的PLC 程序

如何使用 PMAC PLC 程序



NEW IDEAS IN MOTION

PMAC PLC 编程

- * 象许多硬件 PLC一样执行许多任务
- * 通过计算重复循环并且快速 and rapidly regardless of status of motion programs

PLCs 用于:

监测输入点
设定输出点
改变增益
监测卡的状态
指令动作
发送信息



PMAC PLC 类型

前台PLC (PLC0 or PLCC0)

执行与伺服中断

扫描速率由 I8控制

对于时间临界的任务 – 尽可能的短!!

后台 PLC (PLC1-31 or PLCC1-31)

在伺服周期之间运行

重复速率是由:

- 伺服频率

- 电机的轴数

- 运动程序的计算

- PLC 程序的长度和复杂程度

E-MOTION

PMAC多轴运动控制卡

PLC & PLCC 区别

PLCC's 编译的 PLC's

Faster Execution from:

消除解释时间

具有整形计算的能力

浮点操作在编译的 PLC程序要快 2到 3 倍

整形操作 (包括布尔型) 要快 20到 30 倍




NEW IDEAS IN MOTION

后台 PLC/PLCC

执行示例

Uncompiled PLC1, 2, 3
Compiled PLCC1, 2, 3, 4

Execution order:



PLC1
PLCC1, 2, 3, 4
PLC2
PLCC1, 2, 3, 4
PLC3
PLCC1, 2, 3, 4

PLC0 and PLCC0 are at a higher priority level and can interrupt any of the background PLC's

PLC 编程控制

- I5 = 0** 无PLC程序被使能
- = 1** 前台PLC's使能
后台PLC's不使能
- = 2** 前台PLC's 不使能
后台PLC's 使能
- = 3** 所有 PLC's 和 PLCC's 使能

所有存在的PLC's在上电和复位时由 I5变量决定



PLC 编程控制 (续.)

在线指令, 运动程序和PLC程序表达式

ENABLE PLC n

DISABLE PLC n

控制程序可以是独立的或成组的

<CONTROL-D> 禁能 PLC 程序

OPEN PLC n 禁能 PLC n

CLOSE 不使能 PLC n



PMAC PLC程序表达式

1. 条件表达式 (可嵌套)

IF({condition})

WHILE({condition})

AND({condition})

OR({condition})

where {condition}={expression}{comparator}{expression}

[AND/OR{expression}{comparator}{expression}]

2. 逻辑控制表达式

ELSE

ENDIF

ENDWHILE

3. 动作表达式

{variable} = {expression}

COMMAND "{on-line command}"

SEND "{message}"

DISPLAY "{message}"

PLC 延时计时器

既然 DWELL and DELAY 指令可用于运动程序, PMAC 定时器寄存器可用于PLC编程的延时使用

4个 PMAC 内存地址可用:

X:\$0700,0,24,s

Y:\$0700,0,24,s

X:\$0701,0,24,s

Y:\$0701,0,24,s

例: 如果希望在 **PLC**程序中实现**1s**延时

M70->X:\$0700,0,24,s

open plc 1 clear

.

m70=(1000)*8388608/(I10)

while (m70>0)

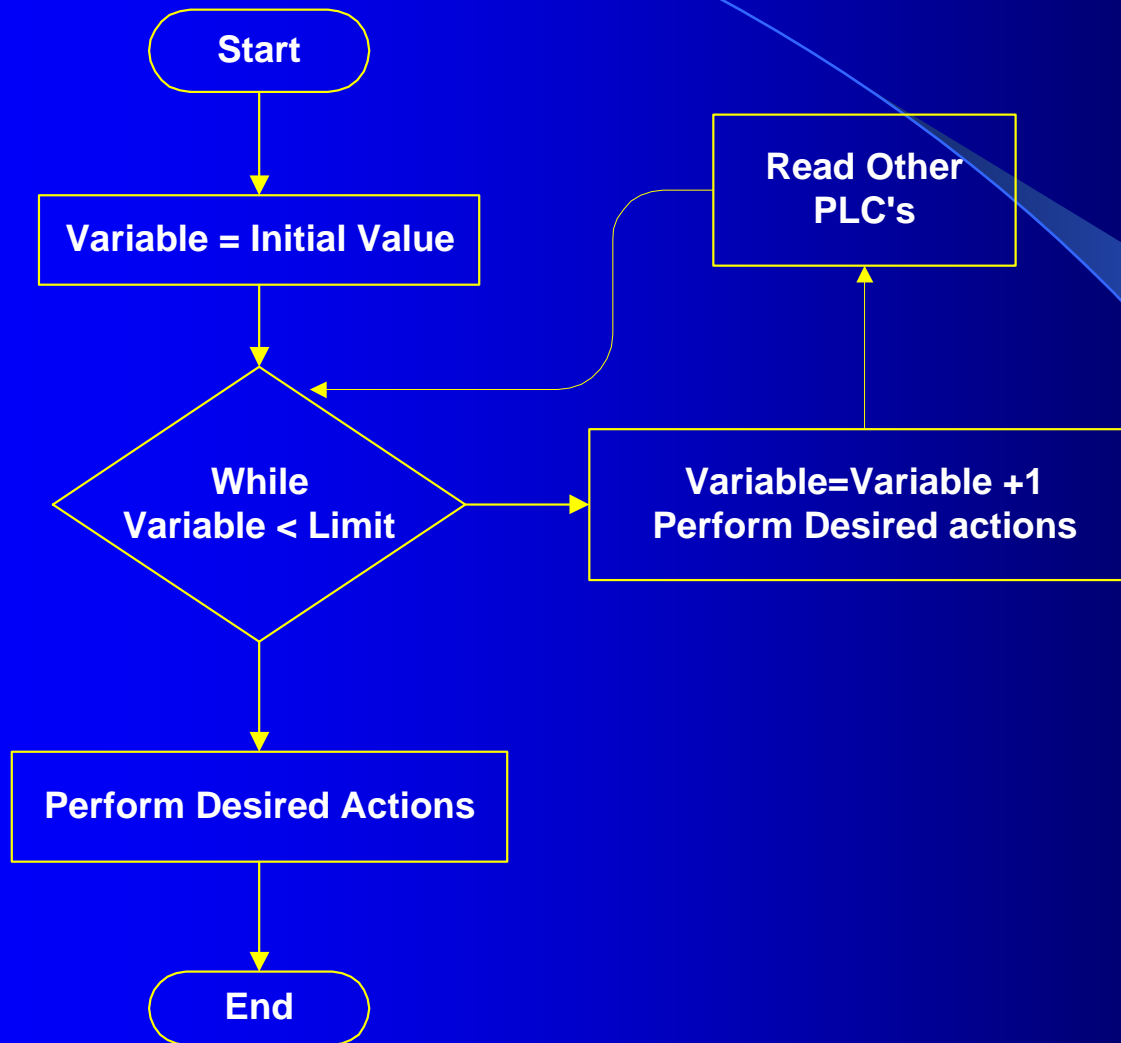
endwhile

.

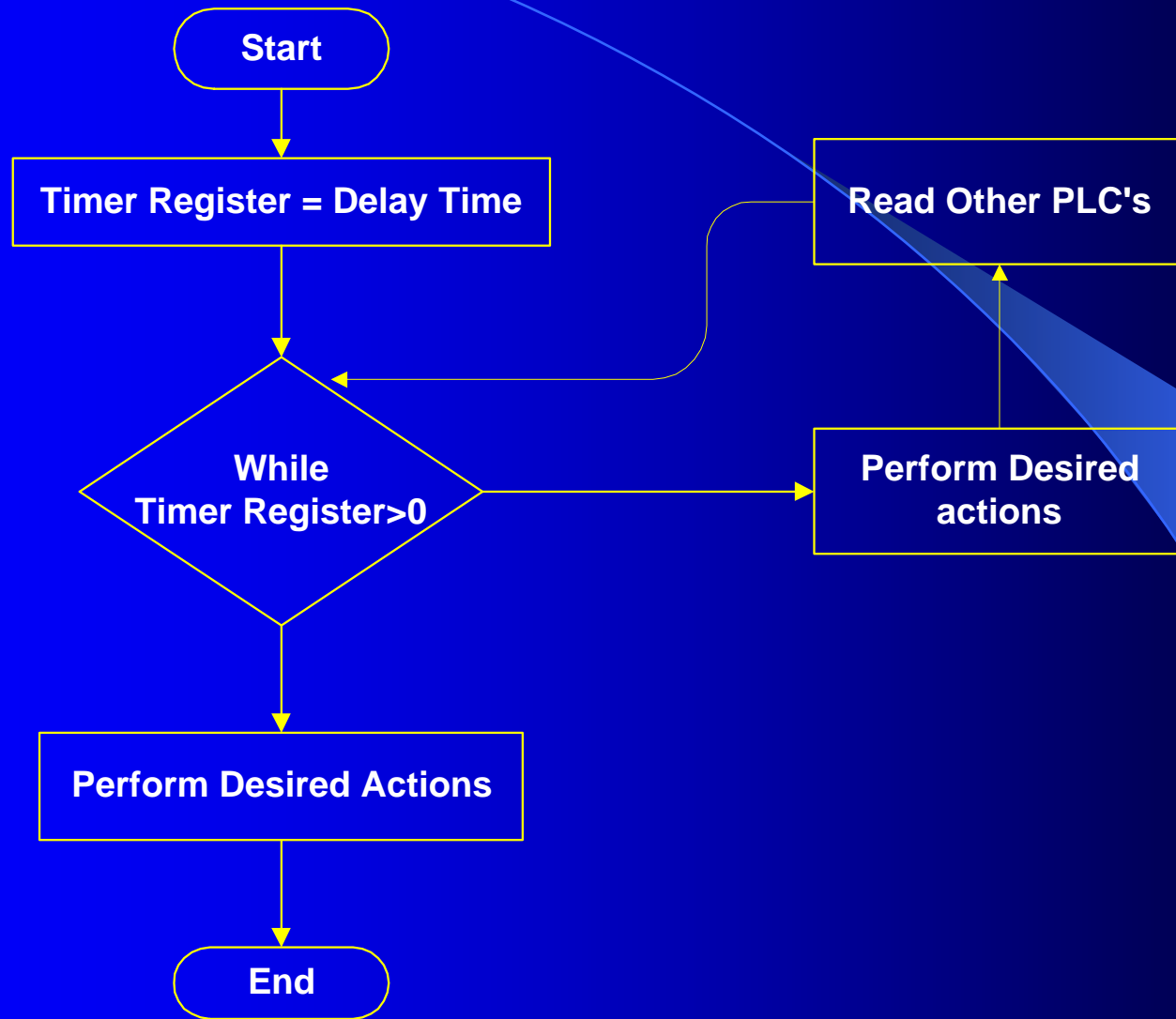
close

计时器寄存单位是伺服周期. 每个伺服周期寄存器连续递减.

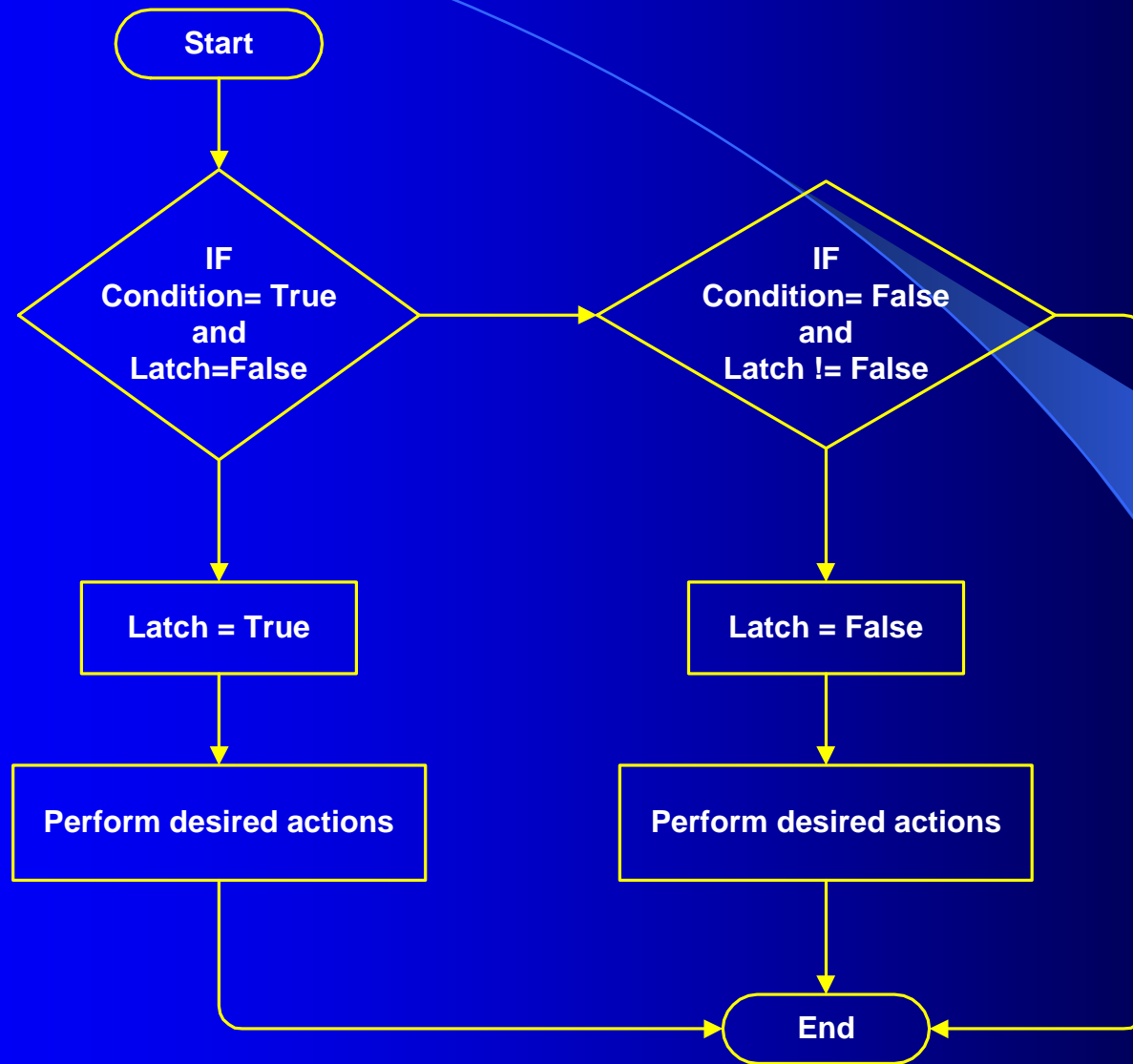
PLC 计数器延时



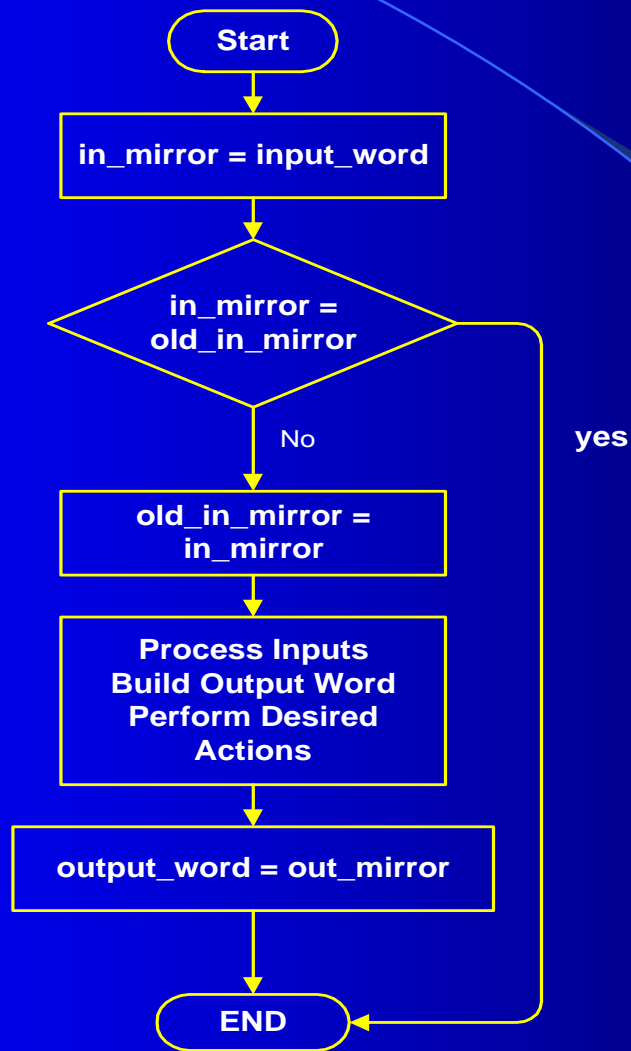
PLC 计时器延时



PLC I/O 结构



ONE-TIME 读/写 PLC I/O 状态



数据采集

PMAC 已经内置数据获取能力称为数据采集功能.

需要学习:

- 什么数据可以被采集
- 如何进行数据采集
- 何时进行数据采集



PMAC 数据采集

- 实时采集任何PMAC卡的地址信息
- 最多达 24 个地址,每个地址 24 或 48 位宽, (由 I21-I44定义; I20作为标志)
- 采样周期从 1 到 8千万个伺服周期 (I19)
- 可以由外部触发采集
- 可上载到计算机用于处理和分析
- PMAC 执行程序可产生采集数据的图形和平台
- 用于:
 - 系统判别
 - 伺服环调整
 - 程序的编译
 - 设备的编译和维护



数据采集过程 & 采样周期

PMAC 可用在线指令采集数据. 指令可通过在线窗口或运动程序中的在线指令或 PLC 程序发送.

DEFFINE GATHER (DEF GAT)
GATHER (GAT)
END GATHER (ENDG)

PMAC准备采集数据
PMAC开始采集过程
PMAC的采集结束

在运动程序或 PLC 程序, 采集指令可如下使用:

COMMAND "DEFFINE GATHER"
COMMAND "GATHER"
COMMAND "END GATHER"

CMD"DEF GAT"
CMD"GAT"
CMD"ENDG"

数据采集周期由 I19 设定. (I19 的单位是伺服周期.)

如 I19 设定为 1, 采样周期为 2250 Hz 在缺省! 如果希望采样周期为 100 Hz (22.5 次低于), I19 应设为大约 22 或 23 (I19 应设为整数). 这将产生一个实际的采样周期 110 或 95 Hz.

一个简单的数据采集的例程

该例子表示如何编写一个运动，并加入采集.

```
***** Set-up and Definitions *****

DEL GAT      ; Erase any defined gather buffer
&l           ; Coordinate System 1
CLOSE        ; Make sure all buffers are closed
#1->X         ; Assign motor 1 to the X-axis - 1 program unit
              ; of X is 1 encoder count of motor #1

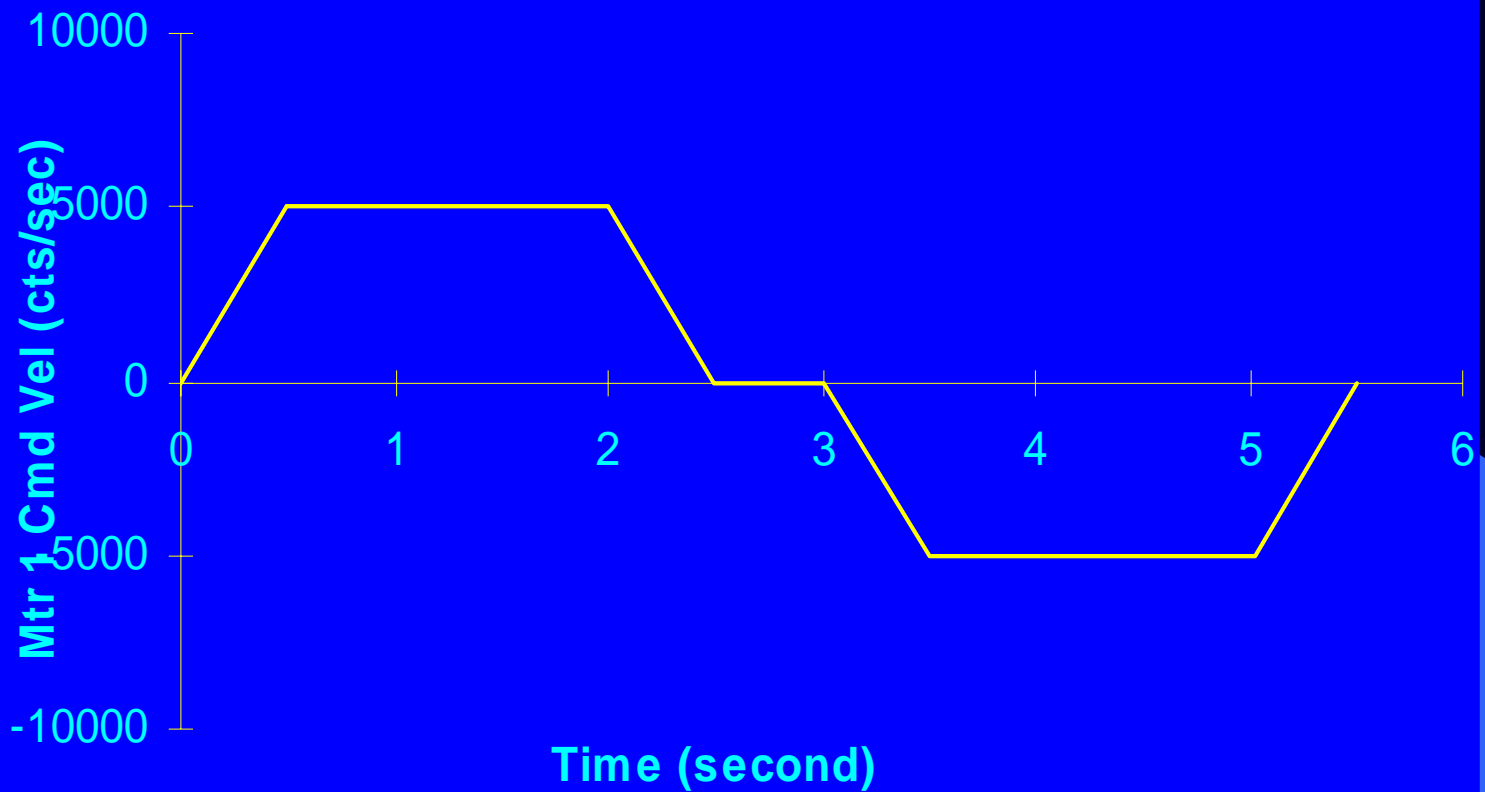
***** Motion Program Text *****

OPEN PROG 1      ; Open buffer for program entry, Program #1
CLEAR          ; Erase existing contents of buffer
LINEAR         ; Blended linear interpolation move mode
ABS           ; Absolute mode - moves specified by position
TA500          ; Set 1/2 sec (500 msec) acceleration time
TS0            ; Set no S-curve acceleration time
F5000          ; Set feedrate (speed) of 5000 units(cts)/sec
X10000         ; Move X-axis to position 10000
DWELL500       ; Stay in position for 1/2 sec (500 msec)
X0             ; Move X-axis to position 0
DWELL0        ; Stop program lookahead
CMD"ENDG"    ; Send On-line command to stop data gathering
CLOSE         ; Close buffer - end of program
```

运行程序和采集:

```
DEF GAT<CR>
GAT &l B1 R      ; 采集, 坐标系1,指向程序1的开始,运行
```

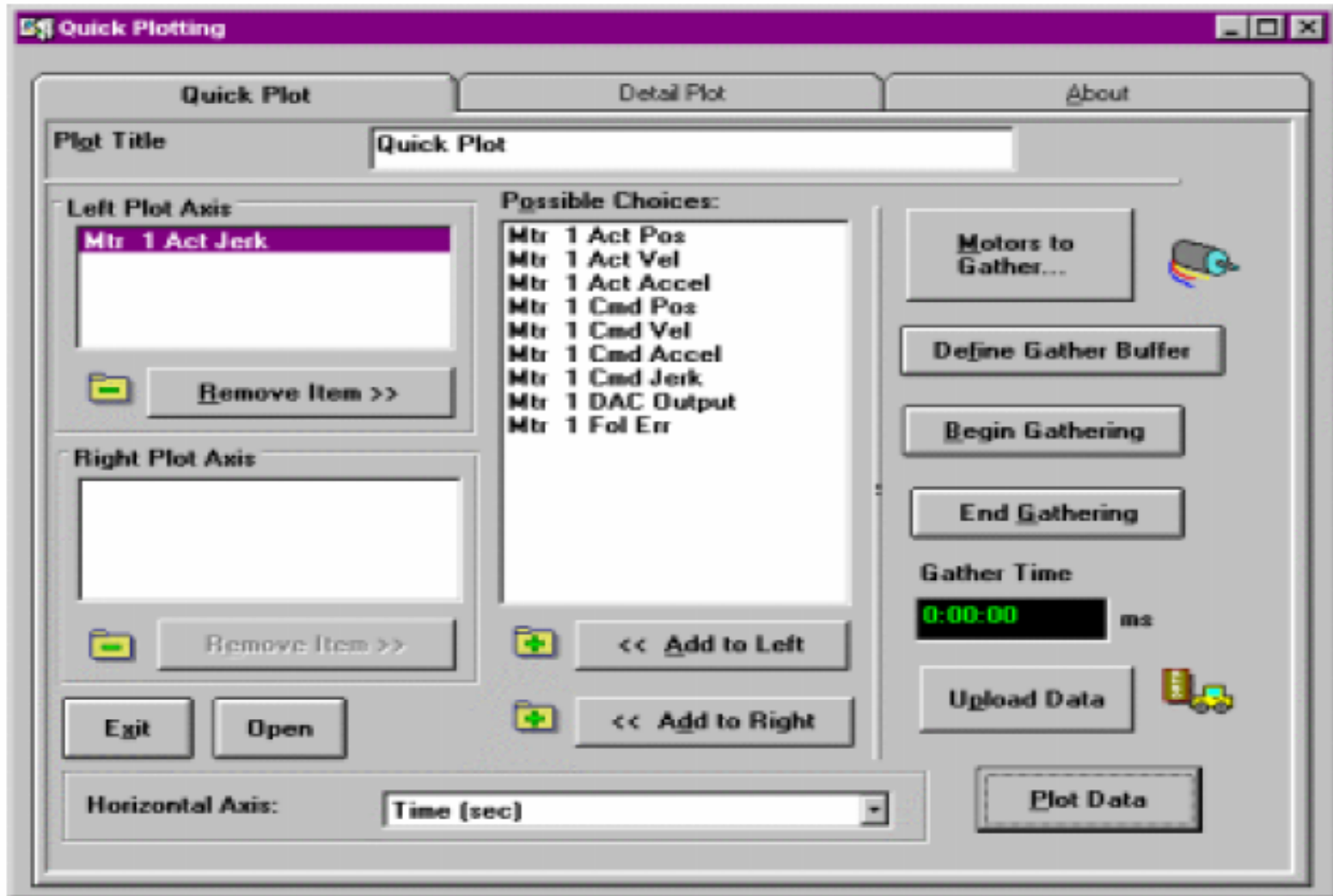
一个简单的数据采集的例程



PMAC PLOT数据采集软件

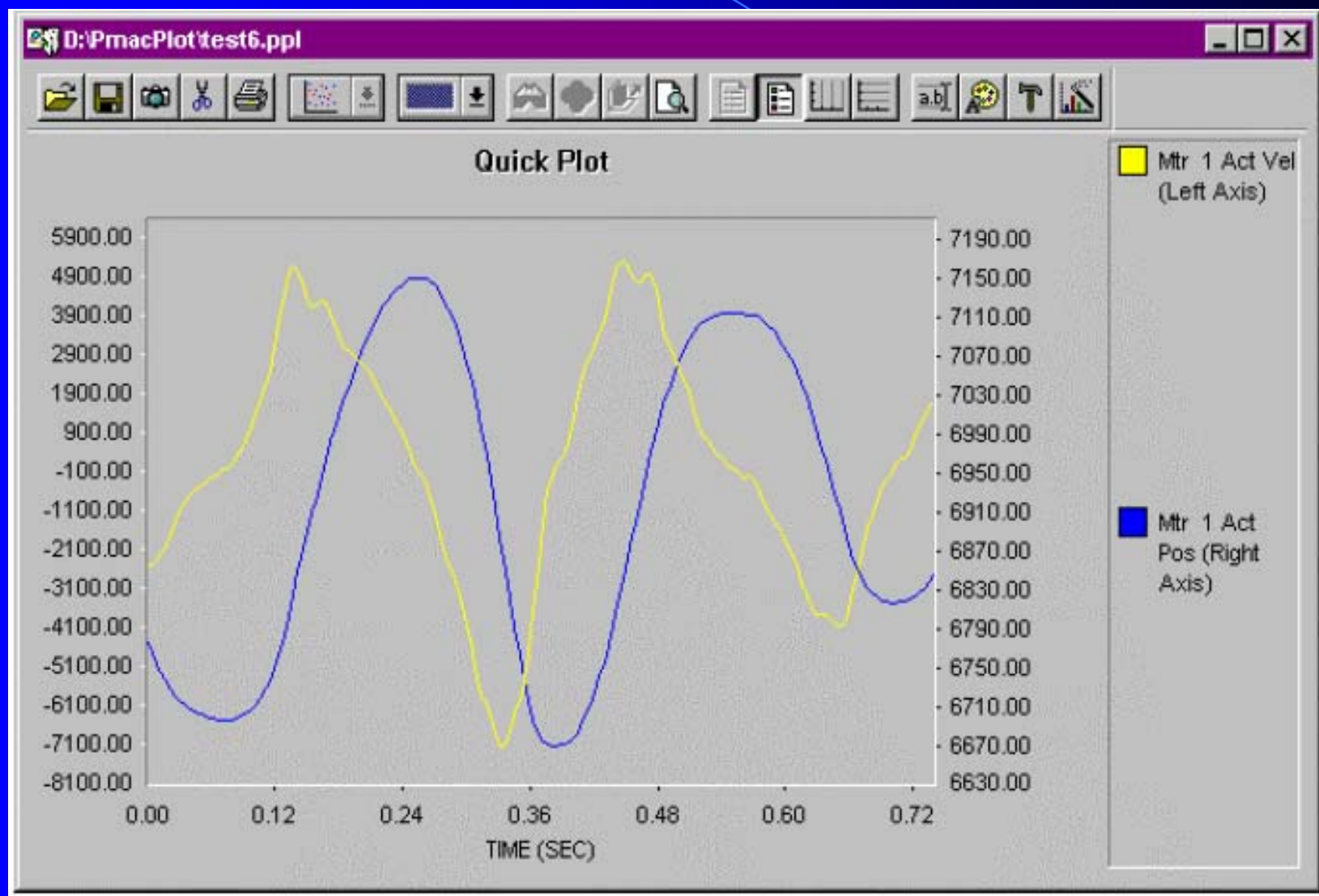
主界面窗口

Main Quick Plot Screen



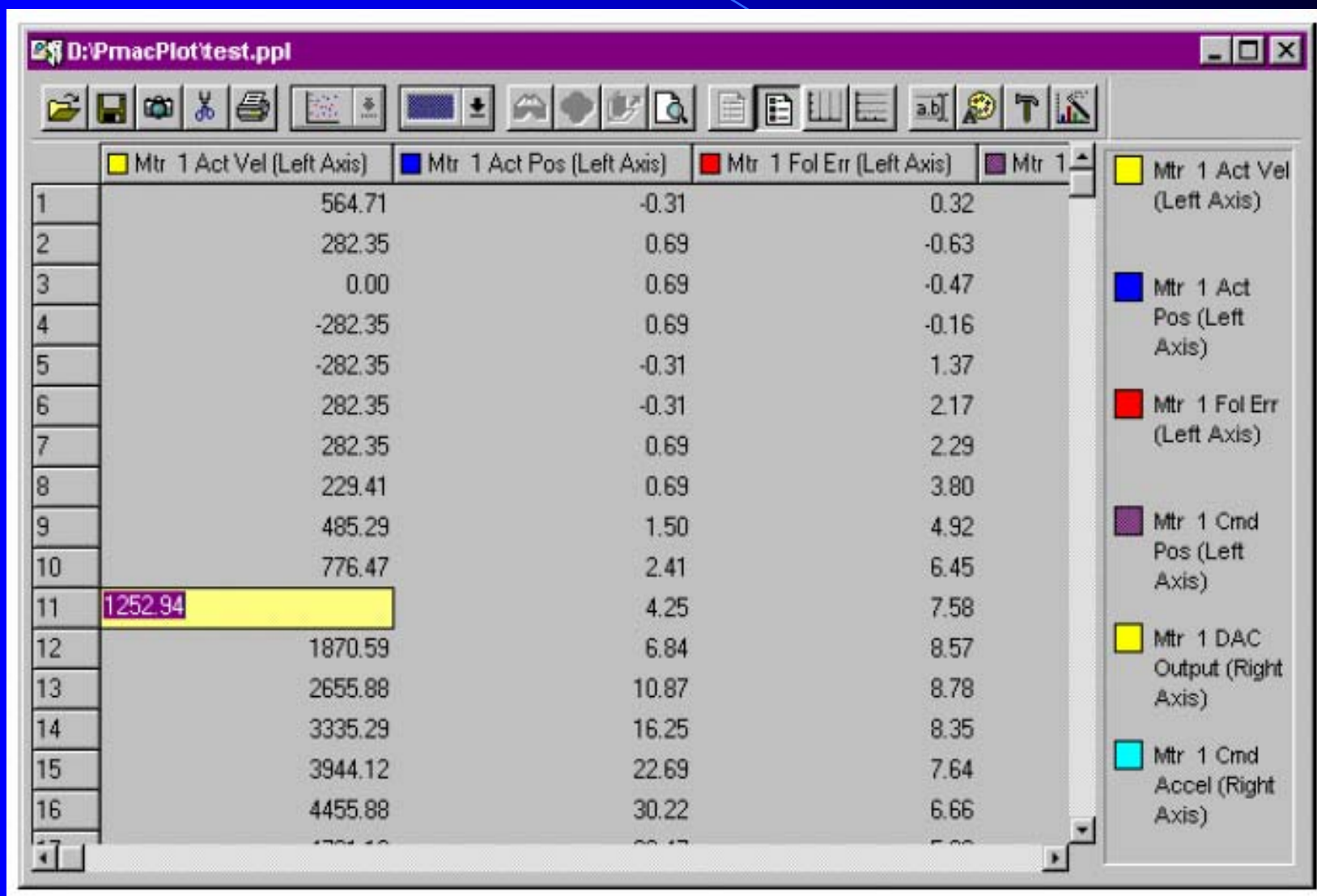
PMAC PLOT数据采集软件

图形采集窗口



PMAC PLOT数据采集软件

数据分析窗口



PMAC 用户手册

PMAC用户手册包括8章. 下面是这些章节的题目及功能

1. 介绍
2. PMAC开始
3. PMAC 特性
4. 与PMAC通讯
5. 故障诊断
6. 输入/输出: PMAC与机床连结
7. 配置电机
8. 配置PMAC通讯
9. 闭合伺服环
10. 使应用安全
11. 基本电机运行
12. 定义坐标系
13. 计算特性
14. PMAC的程序编写
15. 索引

PMAC 软件手册

The PMAC 软件手册包含7章.
下面是每章的标题

1. PMAC 信息概要
2. PMAC I-变量定义
3. PMAC 在线指令讲解
4. PMAC 编程指令讲解
5. PMAC I/O 和内存图
6. 编程举例
7. PMAC PROM 软件升级列表

