



深圳市雷赛智能控制股份有限公司
SHENZHEN LEADSHINE TECHNOLOGY CO., LTD

雷泰运动控制卡

DMC5480

用户手册

Version 2.0

版权所有 不得翻印

©Copyright 2010 Leadshine Technology Co., Ltd.

All Rights Reserved.

版 权 说 明

本手册版权归深圳市雷赛科技有限公司所有，未经公司书面许可，任何人不得翻印、翻译和抄袭本手册中的任何内容。

本手册中的信息资料仅供参考。由于改进设计和功能等原因，雷赛公司保留对本资料的最终解释权，内容如有更改，恕不另行通知。



调试机器要注意安全！用户必须在机器中设计有效的安全保护装置，在软件中加入出错处理程序。否则所造成的损失，雷赛公司没有义务或责任对此负责。

目 录

0 关于本手册	1
1 引言	2
2 硬件概述	3
2.1 硬件结构尺寸	3
2.2 硬件功能描述	4
3 硬件配置与安装	22
3.1 硬件配置	22
3.2 硬件安装	24
4 软件系统概述	25
4.1 硬件驱动程序	25
4.2 运动控制函数库	25
4.3 演示程序	47
4.4 例子程序	48
4.5 多卡运行	53
5 驱动程序安装	54
6 演示软件及应用	57
6.1 参数设置演示	58
6.2 IO操作演示	61
6.3 基本运动演示	62
7 用户系统开发	64
7.1 基于WINDOWS平台的应用软件结构	64
7.2 VISUAL BASIC环境下编程	65
7.3 VISUAL C++环境下编程	67
8 附录	72
8.1 硬件信号接口表	72
8.2 运动控制函数库	76
8.3 常见问题库	99
8.4 ACC2410 接线盒端子定义	100

0 关于本手册

本手册旨在帮助你学习 DMC-5480 控制卡的使用，包括软件函数的调用、参数的设置、硬件接口的接线以及应用例程软件的编写等。

本手册总共分为 8 个部分：

1. 引言：关于本产品的大概描述和关于本产品的相关申明。
2. 硬件概述：关于本产品的硬件相关介绍，包括详细硬件结构尺寸、硬件功能描述以及运动控制平台位置传感器及控制信号布局示例。
3. 硬件配置与安装：包括硬件配置的跳线及开关配置以及硬件的安装说明。
4. 软件系统概述：关于本产品的软件相关介绍，包括本产号所支持的系统驱动程序、运动控制函数库详细说明、演示程序和例子程序的说明以及多卡运行部分。
5. 驱动程序安装：包括驱动程序详细的安装步骤。
6. 演示软件及应用：包括演示软件及应用详细的说明和使用。
7. 用户系统开发：提供 VB 和 VC++ 开发环境下的应用实例编程入门。
8. 附录：提供详细硬件信号接口表、运动控制函数列表、常见问题库。

1 引言

雷泰 DMC-5480 是一款基于 ASIC 技术的高性能，高可靠的 PCI 总线 4 轴运动控制卡，10MHz 脉冲频率，可以控制步进或者数字伺服电机，支持梯形，S 型加速度曲线，直线、圆弧插补，512 段指令缓存，具有运动前瞻功能，运动过程中可调速度和目标位置。该卡同时可接受 4 路编码器信号和位置锁存信号，最多可以控制四轴步进电机或数字式伺服电机， 特别适合应用于多轴高速连续轨迹运动，编码器位置检测等复杂的运动控制领域。

同时，雷赛科技有限公司引进美国 Motion Engineering 公司的先进技术，为 DMC5480 设计了一套易学易用、功能丰富的运动函数库，大大缩短了用户应用软件开发、调试时间。随卡免费提供的 Motion5480 软件，不但可以演示和测试 DMC5480 的绝大多数控制功能，而且还可方便客户测试控制卡及电机控制系统硬件。

2 硬件概述

雷泰(Leadtech) DMC-5480 运动控制卡兼容 PCI V2.3 标准的 32Bit PCI 标准半长卡规范。硬件方面分别提供了 4 轴的脉冲和方向控制信号同时提供多种运动控制功能，伺服驱动器信号，编码器信号，机械位置控制信号、手轮脉冲信号接口，以及同时启停信号和通用输入输出信号。具体硬件系统框图如图 2-1 所示：

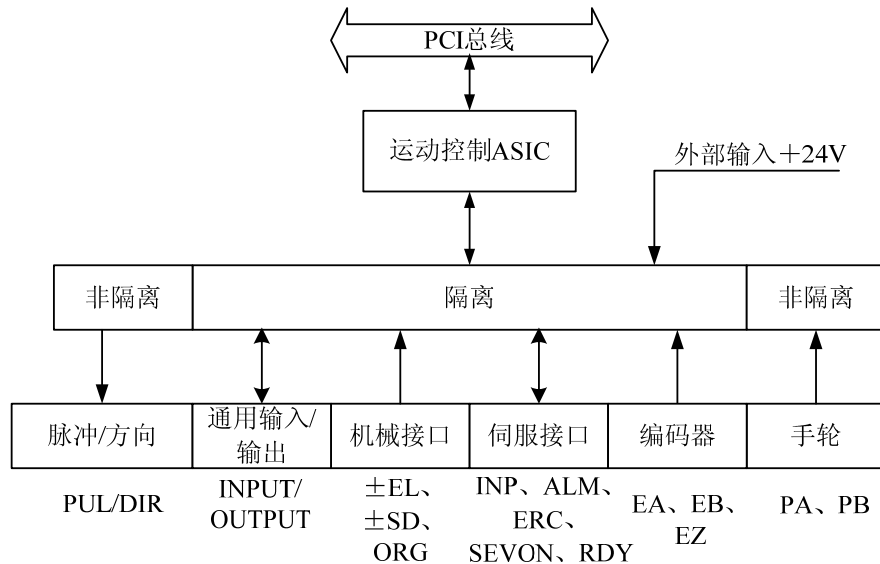


图2-1 DMC-5480运动控制卡系统框图

2.1 硬件结构尺寸

雷泰(Leadtech) DMC-5480 运动控制卡兼容 PCI V2.3 标准的 32Bit PCI 标准半长卡的尺寸规范，具体结构尺寸如图 2-2 所示：

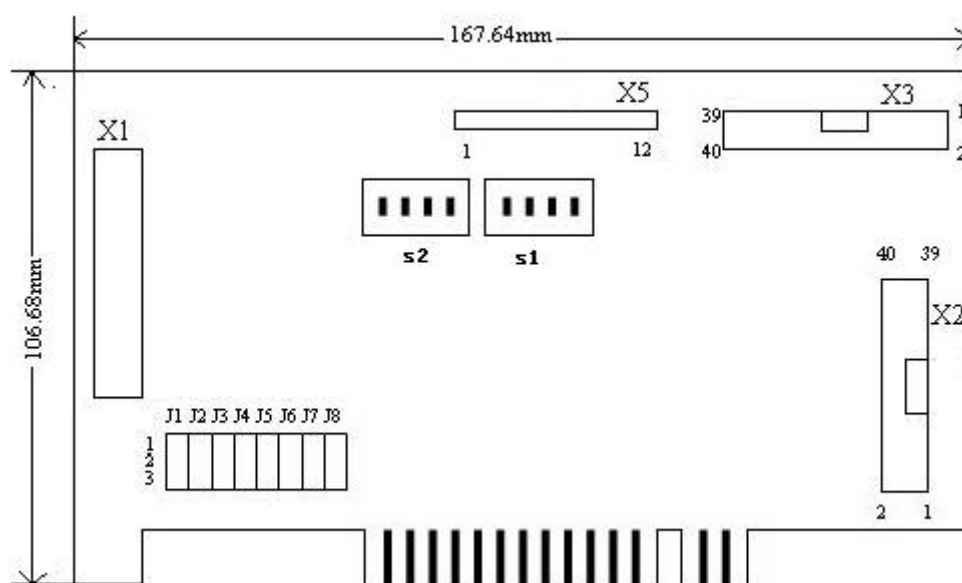


图 2-2 DMC5480 运动控制卡结构尺寸图

2.2 硬件功能描述

雷泰（Leadtech）DMC-5480 硬件方面提供了多种运动控制功能，同时提供了 4 轴的脉冲和方向控制信号，机械位置控制接口、伺服驱动器控制接口、编码器控制接口、手轮脉冲信号接口、以及同时启停控制接口和通用输入输出接口。

2.2.1 控制卡电源供给

在接外部输入输出前，需要先给 DMC-5480 运动控制卡接 24V 电源，提供各种输入输出接口电路的能量来源。具体电路原理图如图 2-3 所示：

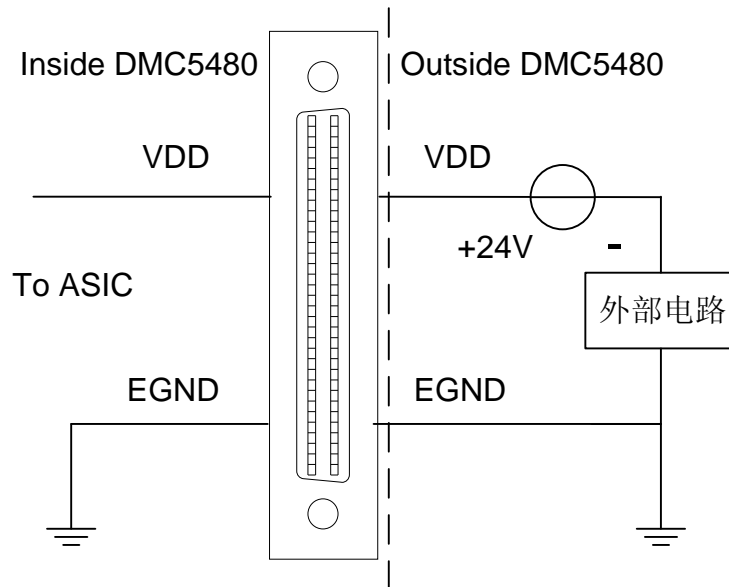


图2-3 板卡供电

2.2.2 运动控制功能

雷泰（Leadtech）DMC-5480 为用户提供了丰富的运动控制功能，其中包括：位置控制；梯形、S 型速度控制；直线、圆弧硬件插补、螺旋插补运动控制等功能。

2.2.2.1 位置控制

最基本的位置控制是指从当前位置运动到另一个位置，一般称为点位运动或定长运动。设置加/减速度以及起始速度和最大速度等参数后执行位移控制指令，上位机将要执行的指令脉冲数写入 DMC-5480 卡，DMC-5480 卡即按设定的速度输出脉冲；当输出脉冲数等于指令脉冲数时，DMC-5480 卡将停止输出。位移与时间的关系如图 2-4 所示：

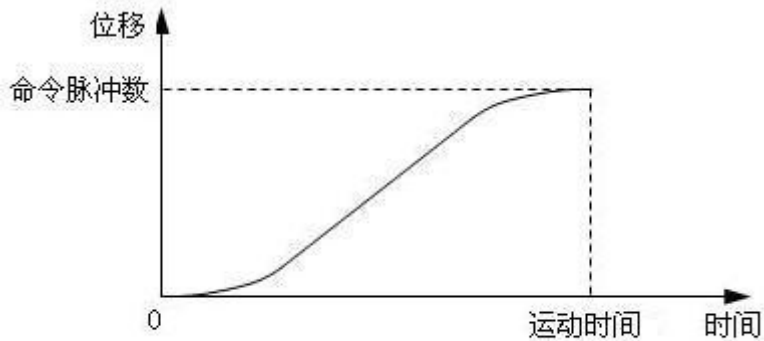


图2-4 位移曲线

2.2.2.2 速度控制

速度控制是指电机从起始速度开始运行，加速至指定速度连续运动。只有当接收到停止命令或外部停止信号后，才减速直至停止（也可设为立即停止）。该模式不控制运动距离。其速度与时间的关系如图 2-5 所示。

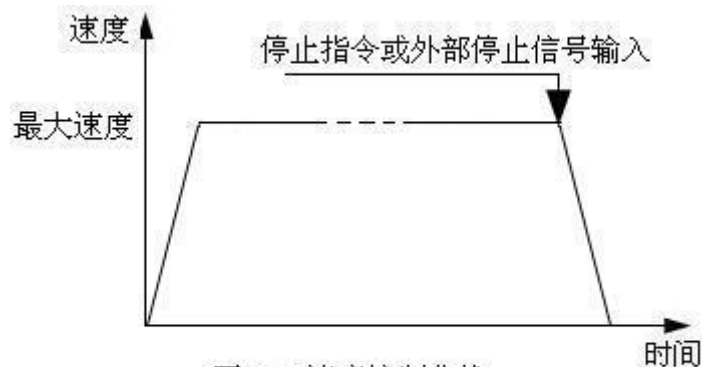


图2-5 速度控制曲线

雷泰（Leadtech）DMC-5480 为用户提供了梯形、S 型速度控制模式。

A) 梯形曲线速度控制

梯形速度曲线控制指令是使 DMC-5480 卡按梯形速度曲线输出指令脉冲。即：电机从起始速度开始运动，加速至最大速度后保持速度不变，临结束前减速至起始速度，并停止。运动的距离由定长运动命令设定。梯形速度曲线如图 2-6 所示。

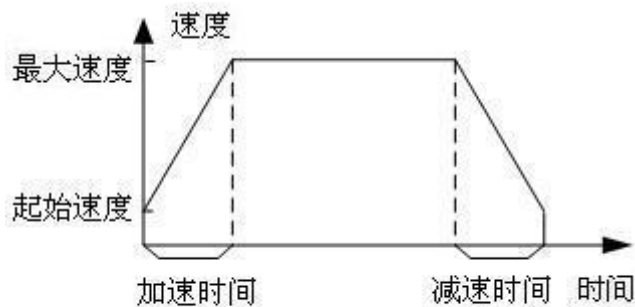


图2-6 梯形速度曲线

B) S 形曲线速度控制

梯形速度曲线虽然实现起来简单，但它的加速度有突变，速度曲线不平滑，因而运动中有冲击现象，容易引起机器噪声和传动机构的磨损。在梯形速度曲线上（参见图 2-7），运动的不平滑主要表现在四个瞬间的速度转折及相对应的加速度突变，这四个瞬间分别是：起始时、升至最高速度时、从最高速度下降时和最后停止时。

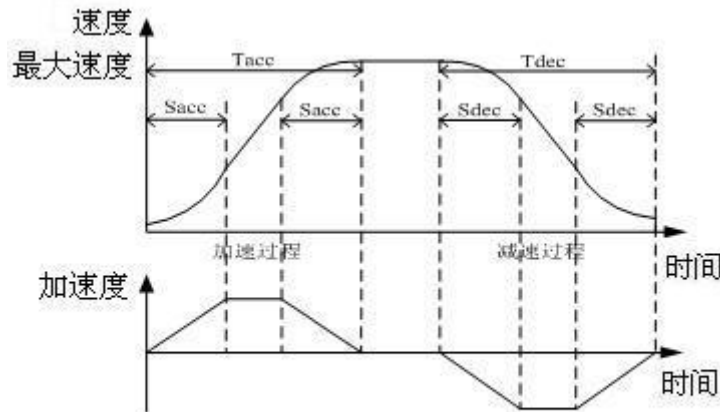


图2-7 S形速度曲线及其加速度曲线

若将加速度改为线性变化，则速度曲线相应将变得光滑，如图 2-8 所示。升速和减速阶段均变得像 S 的形状。采用此种速度曲线，运动更平稳，且有助于缩短加速过程、降低运动装置的振动和噪声，同时还可以延长机械传动部分的寿命。

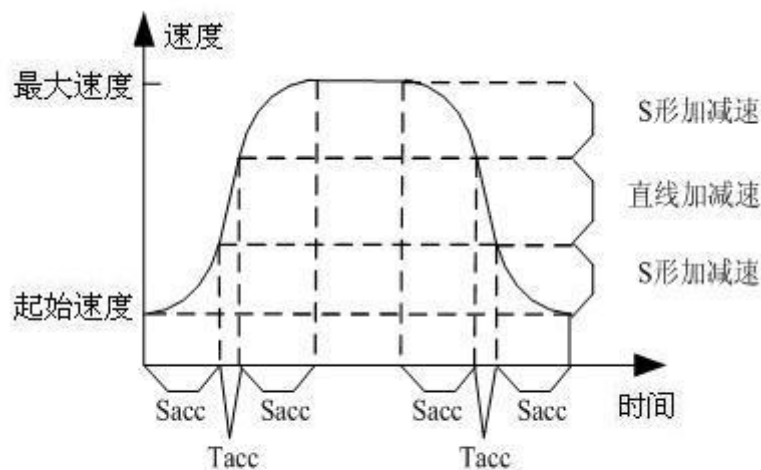


图2-8 S形速度曲线

2.2.2.3 插补运动

插补即根据给定的数学函数，在理想的轨迹式轮廓上的已知点之间，确定一些中间点的一种方法。插补方式有：直线插补、圆弧插补、抛物线插补、样条线插补等。DMC-5480 卡具有强大的插补功能（详细介绍请参考[4.2.5.2 节多轴运动控制](#)中相关内容）。

(1) 二轴至四轴直线插补

选择任意的二轴至四轴都可以进行直线插补。

(2) 圆弧插补

圆弧插补可以从四轴中任选二轴进行。圆弧插补从当前位置开始，根据所指定的圆心和终点位置以及插补的方向（按顺时针或逆时针）来进行。

(3) 恒定线速度

在插补运动中不管曲线如何变化，其线速度在运动过程中可保持恒定。

(4) 连续插补

连续插补是执行一系列的插补指令过程，例如连续执行直线插补、圆弧插补、直线插补、……等指令。在此过程中线段与线段之间，运动没有停顿、速度保持连续。

2.2.3 脉冲和方向控制接口

雷泰（Leadtech）DMC-5480 硬件方面提供 4 轴的脉冲和方向控制信号接口。脉冲信号PUL和方向信号DIR均为差分输出，典型接口电路如图 2-9 所示。同时可配置为单端输出，如图 2-10 所示。输出方式具体设置可参考[3.1.1 跳线配置](#)，对外接口SCSI68 具体引脚分配见[8.1.1 接口X1 引脚定义](#)。

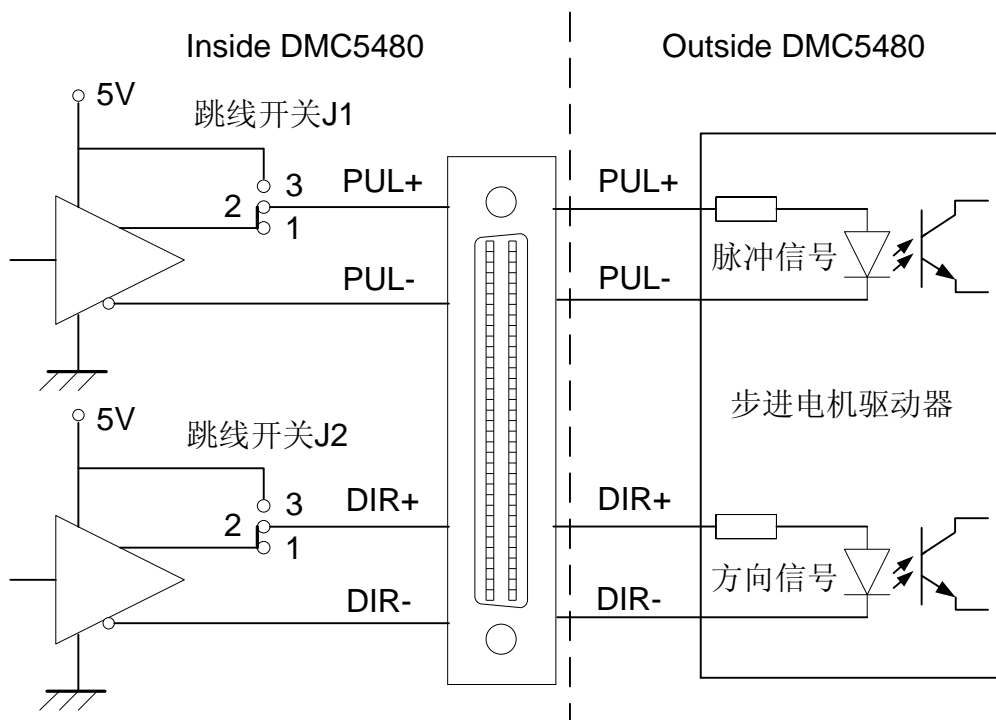


图 2-9 差分方式接线图

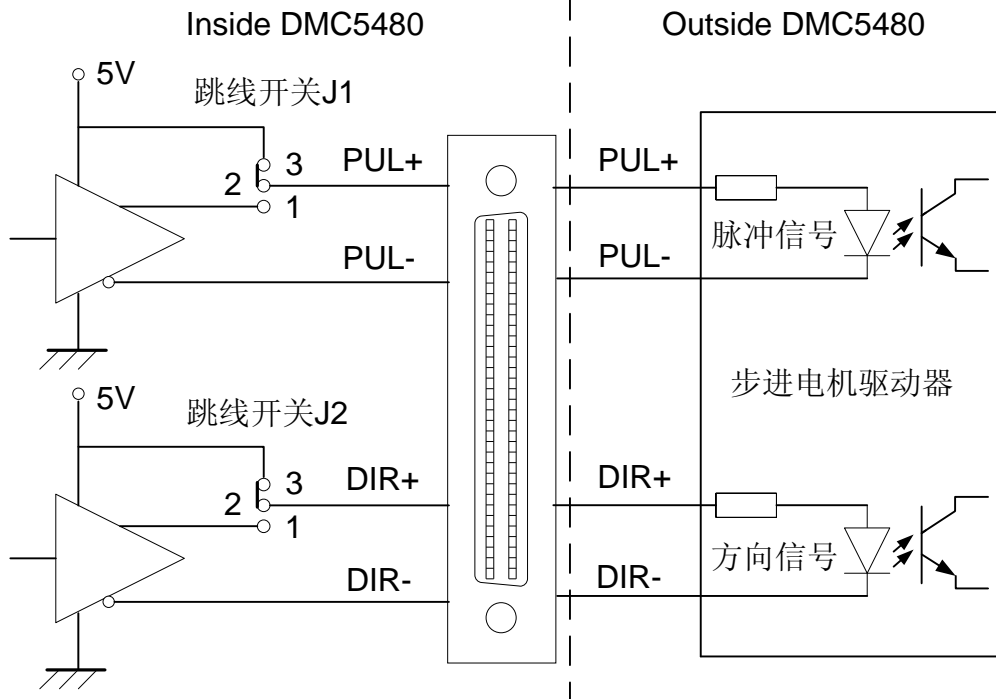


图2-10 单端输出方式接线图

雷泰（Leadtech）DMC-5480 卡可以输出两类指令脉冲方向信号：一种为脉冲+方向模式（单脉冲模式，如图 2-11）；一种为正脉冲+负脉冲模式（双脉冲模式，如图 2-12）。

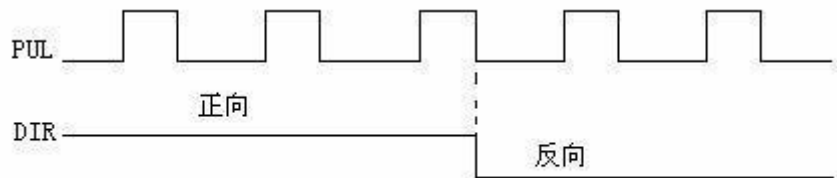


图2-11 单脉冲模式

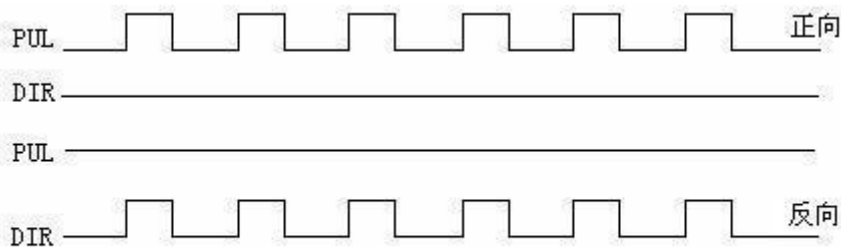


图2-12 双脉冲模式

2.2.4 机械位置控制接口

雷泰（Leadtech）DMC-5480 均为 4 轴提供了原点 ORG 位置信号输入接口、SD+/SD-减速位置信号输入接口以及 EL+/EL-正负限位信号输入接口。各信号功能实现均由控制卡硬件来完成。

2.2.4.1 ORG：原点位置信号输入接口

通常运动系统中都要用一个位置传感器设置一个位置参考点，即原点位置，以便于进行精确的位置控制。雷泰（Leadtech）DMC-5480 为每个轴都提供了一个原点位置传感器输入端口ORG，对外接口SCSI68 具体引脚分配见[8.1.1 接口X1 引脚定义](#)，典型接口电路如图 2-13 所示。

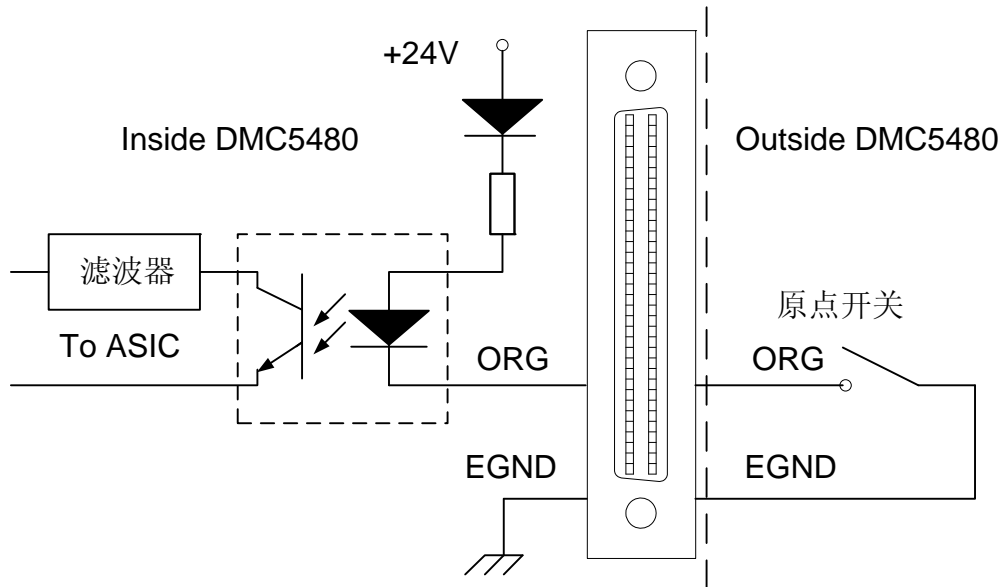


图2-13 原点信号接口原理图

雷泰（Leadtech）DMC-5480 控制卡的各轴的原点位置传感器信号经由 ORG 输入端口接入 DMC-5480 后，经过光电隔离后进入 ASIC 芯片。光电隔离可以有效地隔离外部电源对 DMC-5480 内部电路的干扰提高系统的可靠性。

通常在进行运动控制之前，都需要用回原点运动控制平台向原点方向运动，当运动控制卡检测到原点传感器ORG信号后，运动自动停止，并将停止位置设为该轴的原点。具体方式和操作请参考[4.2.3 回原点运动](#)。

2.2.4.2 SD：减速位置信号输入接口

DMC5480 减速信号保留，目前只作通用信号使用。对外接口SCSI68 具体引脚分配见[8.1.1 接口X1 引脚定义](#)，典型接口原理如下图 2-14 所示。

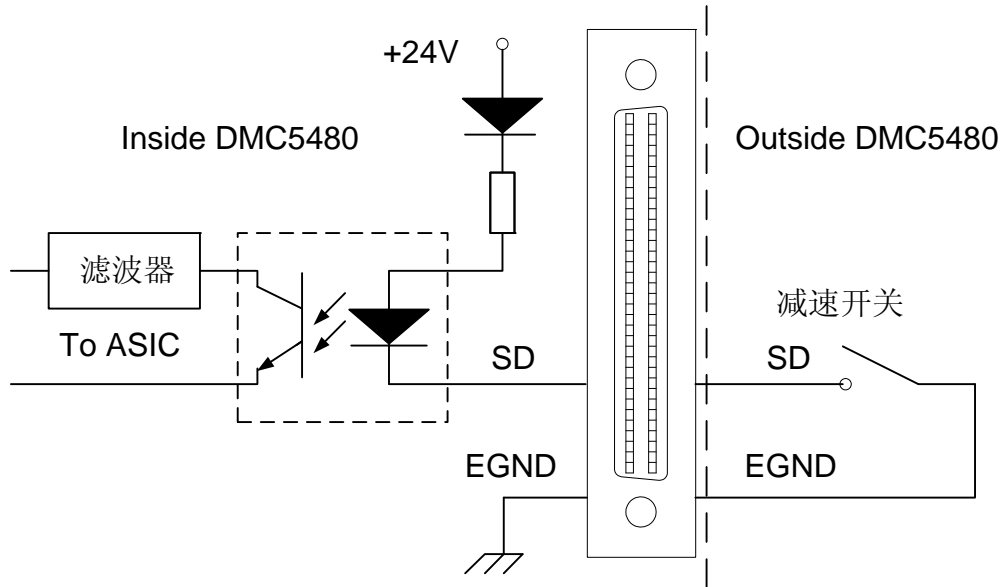


图2-14 减速信号接口原理图

雷泰（Leadtech）DMC-5480 各轴的减速位置传感器信号经由 SD 输入端口接入 DMC-5480 后，经过光电隔离后进入运动控制 ASIC 芯片。光电隔离可以有效地隔离外部电源对 DMC-5480 内部电路的干扰，从而提高 DMC-5480 以及整个控制系统的可靠性。

2.2.4.3 EL+/EL-: 正负限位信号输入接口

运动系统中通常会用一个位置传感器设置一个机械限位点，以确定运动的边界位置，保护机械设施。雷泰（Leadtech）DMC-5480 均为每一轴都提供两个机械限位信号EL+ 和 EL-，EL+为正向限位信号，EL-为反向限位信号。当运动部件接触到限位开关时，EL+/EL-即有效，DMC-5480 将禁止运动部件继续向限位方向运动。典型接口及接线原理如图 2-15 所示，对外接口SCSI68 具体信号引脚分配见[8.1.1 接口X1 引脚定义](#)。

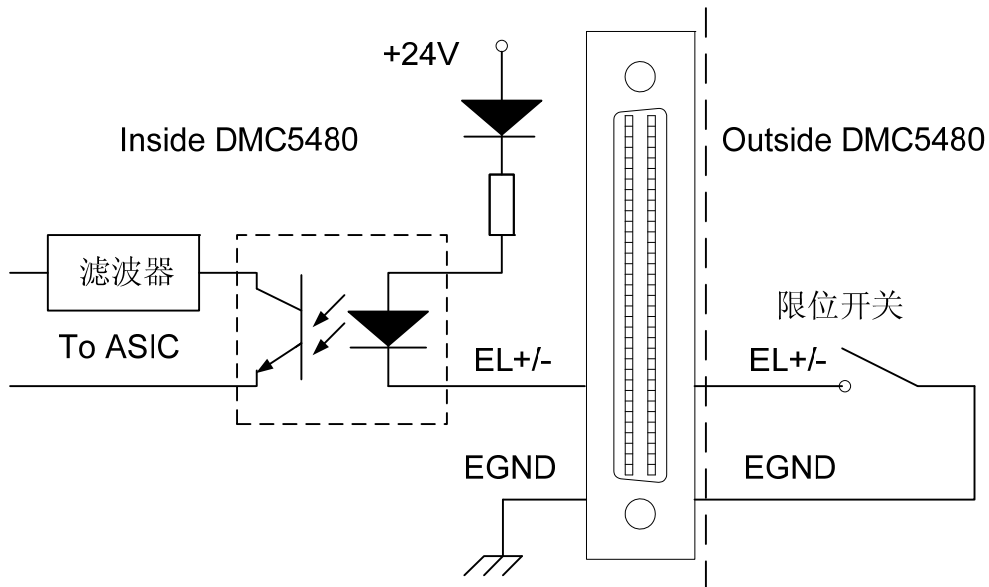


图2-15 限位信号接口原理图

使用雷泰（Leadtech）DMC-5480 运动控制卡的系统的各轴的机械限位传感器信号经由 EL+/EL-输入端口接入 DMC-5480 后，经过光电隔离后送入运动控制 ASIC 芯片。光电隔离可以有效的隔离外部电源对 DMC-5480 内部电路的干扰，从而提高 DMC-5480 以及整个控制系统的可靠性。

具体应用中用户需根据限位开关的类型来设置限位开关的有效工作电平。当使用常开型限位开关，应通过软件选择 EL+/EL-信号为低电平有效；当使用常闭型限位开关，应通过软件选择 EL+/EL-信号为高电平有效。

2.2.5 伺服驱动器控制接口

雷泰（Leadtech）DMC-5480 运动控制卡均为每一轴都提供了伺服电机驱动器专用信号接口（SEVON、INP、ALM 和 ERC），其中 INP 和 ALM 用于监控伺服电机的状态，SEVON 和 ERC 用于设置伺服电机的状态。

2.2.5.1 SEVON：使能伺服驱动器信号

雷泰（Leadtech）DMC-5480 运动控制卡均为每一轴提供了 1 个用于使能伺服驱动器的SEVON控制信号。当SEVON信号为无效状态时，伺服驱动器不使能，电机处于自由状态；当SEVON信号有效时，伺服驱动器使能，电机将会锁紧。典型接口及接线原理如图 2-16，对外接口SCSI68 具体引脚分配见[8.1.1 接口X1 引脚定义](#)。

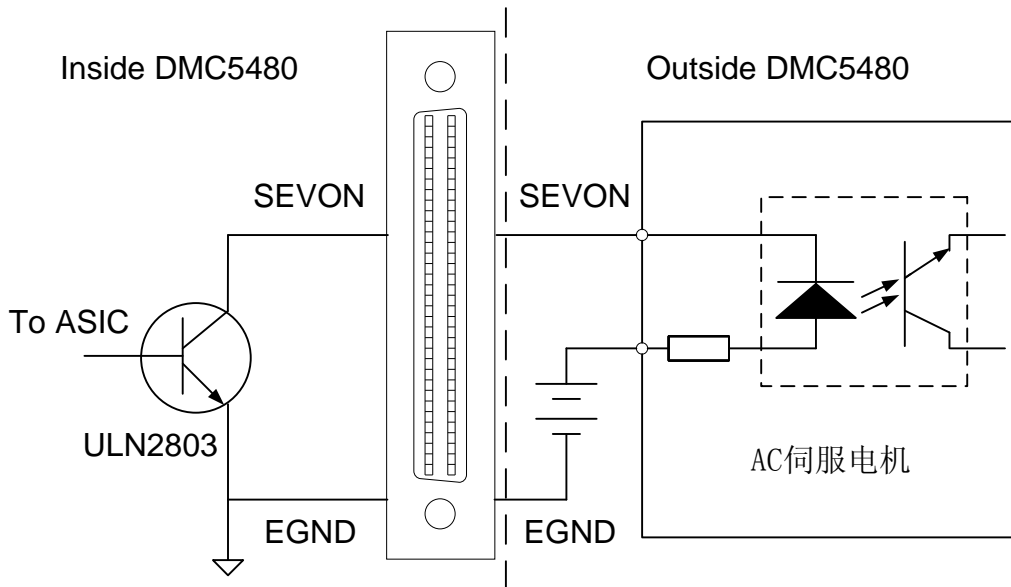


图2-16 SEVON信号接线原理图

用户应用系统中使用到伺服驱动器是可以将SEVON接到控制伺服驱动器的使能端口来控制其工作状态。SEVON信号由运动控制ASIC芯片发出后经过光电隔离后再经由达林顿驱动放大电路（2803）后输出到SCSI68端口。该端口不做SEVON信号输出端口使用时可用作通用数字输出口。

2.2.5.2 INP：伺服定位完成信号

雷泰（Leadtech）DMC-5480运动控制卡均为每一轴提供了用于监控伺服电机定位结果的INP信号接口。典型接口及接线原理如图2-17，对外接口SCSI68具体信号引脚分配见[8.1.1 接口X1引脚定义](#)。该端口不做INP信号输入端口使用时可用作通用数字输入口。

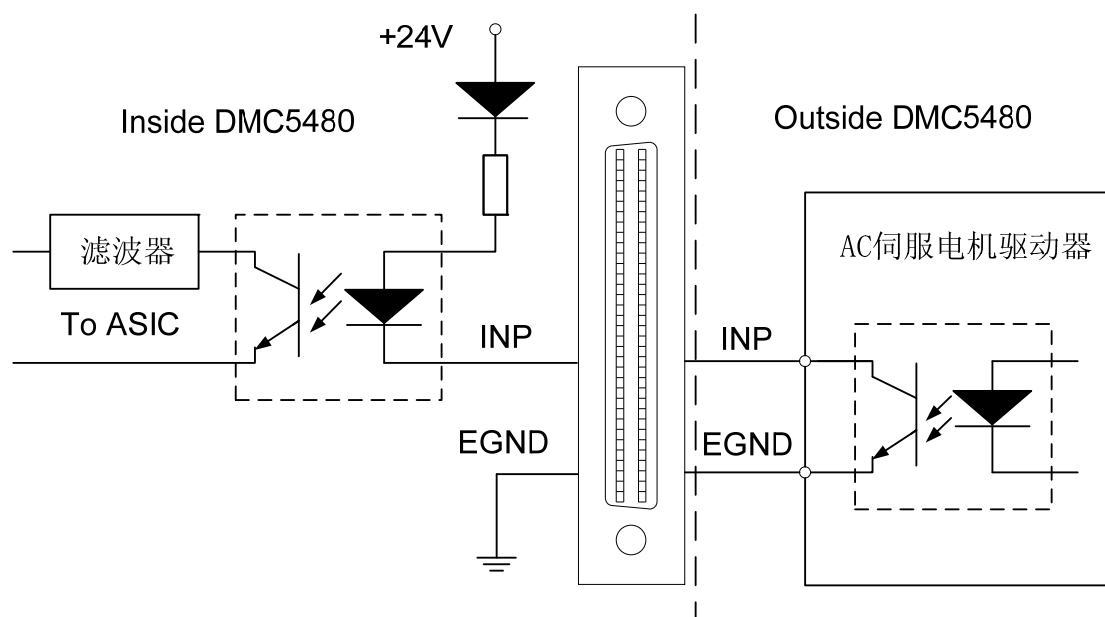


图2-17 INP信号接线原理图

伺服电机驱动器通常都有一个位置偏差计数器，记录指令脉冲和位置反馈脉冲之间的偏差。伺服电机驱动器将控制电机运动使位置偏差趋于 0，但是，电机实际位置总是滞后于指令脉冲的。所以，当运动控制卡的指令脉冲发送完毕时，伺服电机并没有立即停止，而是会继续运动（如图 2-18），直到位置偏差趋于 0；然后，驱动器将发出一个 INP 信号给运动控制器，告之运动控制卡伺服电机已经停止。

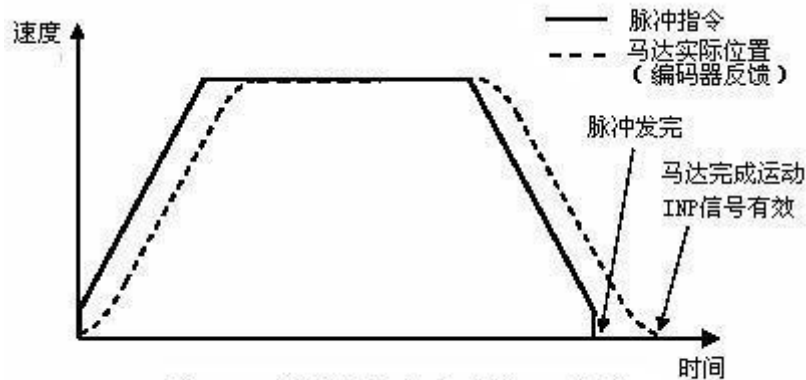


图2-18 伺服定位完成时的INP信号

2.2.5.3 ALM: 伺服驱动器报警信号

雷泰（Leadtech）DMC-5480 运动控制卡均为每一轴提供了用于监控伺服驱动及电机状态ALM信号输入接口。典型接口及接线原理如图 2-19，对外接口 SCSI68 具体信号引脚分配见[8.1.1 接口X1 引脚定义](#)。该端口不做ALM信号输入端口使用时可用作通用数字输入口。

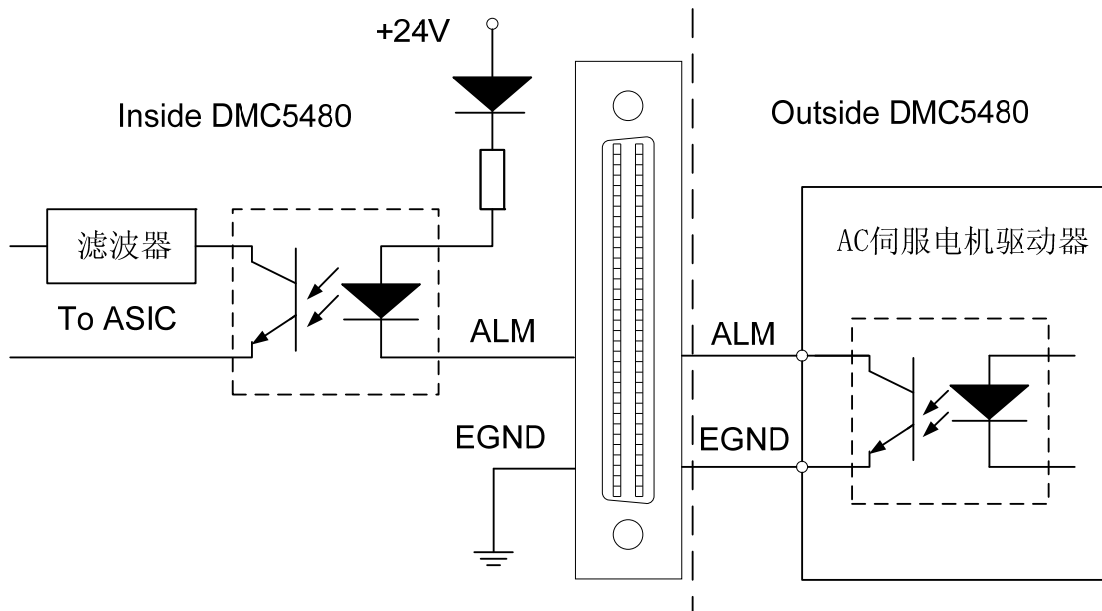


图2-19 ALM信号接线原理图

ALM 信号是 AC 伺服电机驱动器发出的报警信号。当 DMC-5480 接收到 ALM 信号后，将立即中止发送运动指令脉冲，或先减速再停止发送脉冲。

2.2.5.4 ERC：伺服驱动位置偏差计数器清零信号

雷泰（Leadtech）DMC-5480 运动控制卡均为每一轴提供了用于清零伺服驱动位置偏差计数器的ERC信号输出接口。伺服驱动器依赖电机目标位置（即要求电机达到的位置）和当前位置（即电机已经到达的位置）之间的误差来驱动电机运动，若这个误差为零电机将停止运动。ERC信号是控制卡输出给伺服驱动器的控制信号，当伺服驱动器接收到该信号时会立即清除误差并停止电机运转。典型接口及接线原理如图 2-20，对外接口SCSI68 插座具体引脚分配见[8.1.1 接口X1 引脚定义](#)。该端口不做ERC信号输出端口使用时可用作通用数字输出口。

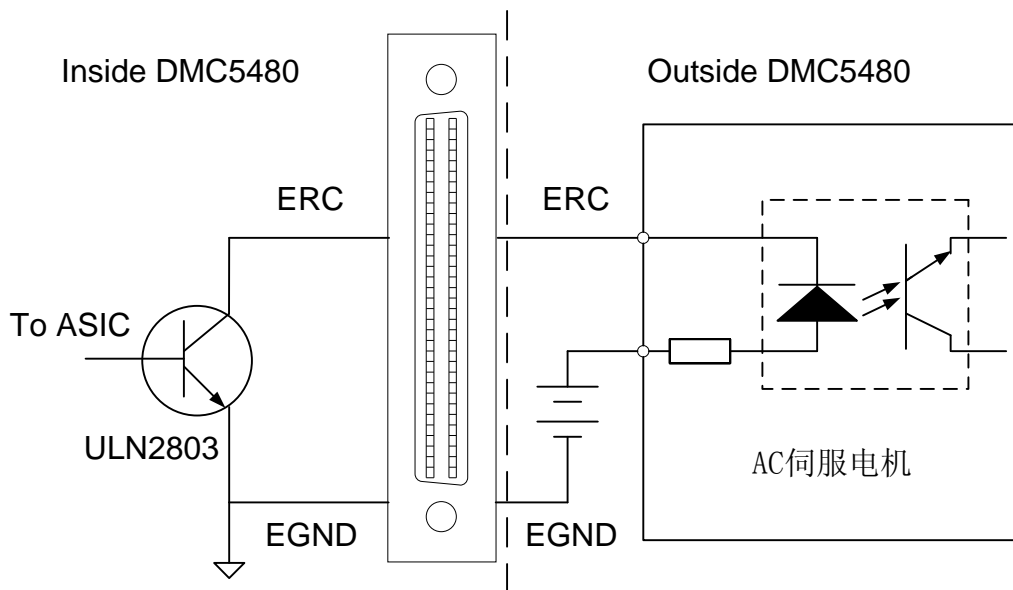


图2-20 ERC信号接线原理图

ERC（Deflection counter clear）信号将 AC 伺服电机驱动器中的位置偏差计数器清零，使电机立即停止。ERC 通常在以下 4 种情况下使用：(1) 回原点运动结束；(2) 运动到限位开关位置；(3) 接收到报警信号；(4) 接收到了一个急停信号。接线原理图如图 2-23 所示：

在以下几种情况下可能需要用到 ERC 信号：

- 回原点运动时触发原点信号时；
- 运动过程中，触发限位信号时；
- 伺服驱动器发出 ALM 信号时；
- 软件发出立即停止信号时。

这四种情况出现时，控制卡将停止输出脉冲，但是由于伺服电机本身的特性决定，命令脉冲停止到电机本身停止会出现一些延迟，电机将继续运行直到 INP 信号有效（即误差计数达到允许范围内），这时如果给伺服驱动器发出一个 ERC 信号就可将误差清零从而使电机停止。当指定轴在输出 ERC 信号后，开始启动定时器，定时器按设定使 ERC 保持有效，这个期间新的指令脉冲被忽略，直到定时结束（ERC 信号 OFF）后，才重新接收新的指令脉冲。

2.2.6 编码器信号输入接口

雷泰（Leadtech）DMC-5480 运动控制卡为每轴都提供了独立的编码器输入端口，用于检测电机运动的实际位置。

2.2.6.1 编码器信号输入接口

编码器输入信号包括 EA、EB 和 EZ，每个轴都有三对差分的 A 相、B 相和 Z 索引信号，EA 和 EB 用来进行位置计数，EZ 可用作原点信号。每对差分信号将被转化成 EA、EB、EZ 的 TTL 数字信号。雷泰（Leadtech）DMC-5480 运动控制卡均为每一轴提供了用于接编码器输入 EA+/-、EB+/-、EZ+/- 3 组差分信号输入的接口，并支持 2 种类型的编码器信号输入：正负脉冲输入或 A/B 相正交信号。

（1）非 AB 相脉冲输入模式

即为脉冲+方向方式。此模式下 EA 端子接收脉冲信号；EB 端子接收方向信号，高电平对应于计数器数值增加，低电平对应于数值减少。

（2）AB 相正交信号输入模式

在这种模式下，EA 脉冲信号“超前”或“滞后”EB 脉冲信号 90 度，而这种“超前”或“滞后”就表示电机的运转方向。如图 2-21 所示：当 EA 信号超前 EB 信号 90° 时，被视为正转；当 EB 信号超前 EA 信号 90° 时，被视为反转。

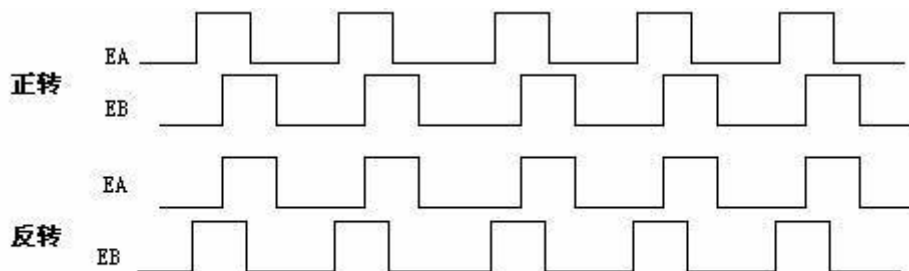


图2-21 正反转A/B相正交信号

当 EA 信号“超前”EB 信号时，编码器加计数；EA 信号“滞后”EB 信号时，编码器减计数。而且，用户可选用 4、2、1 倍计数模式对 EA，EB 信号进行计数设置，分别介绍如下：

4 倍计数：若为正向计数，编码器计数器的值为 EA 反馈脉冲数的 4 倍至 4 倍减 3 之间；若为负向计数，编码器计数器的值为 EB 反馈脉冲数的 4 倍至 4 倍减 3 之间。

2 倍计数：若为正向计数，编码器计数器的值为 EA 反馈脉冲数的 2 倍至 2 倍减 1 之间；若为负向计数，编码器计数器的值为 EB 反馈脉冲数的 2 倍至 2 倍减 1 之间。

1 倍计数：若为正向计数，编码器计数器的值为 EA 反馈脉冲数；若为负向计数，编码器计数器值为 EB 反馈脉冲数。

例如：如果使用的编码器为 2500 线，即电机转一周反馈的 EA、EB 脉冲数

都为 2500 个，让电机转一周，若编码器反馈输入模式为 4 倍计数，编码器计数器的值为 10000；若设置为 2 倍计数，编码器计数器的值为 5000；若设置为 1 倍计数，编码器计数器的值为 2500。这样可以提高编码器的分辨率。

编码器输入信号接线方法如下：

1. 差分模式：

如图 2-22 所示，在此模式下，输入信号的正端接 EA+/EB+/EZ+端，负端接 EA-/EB-/EZ-端。

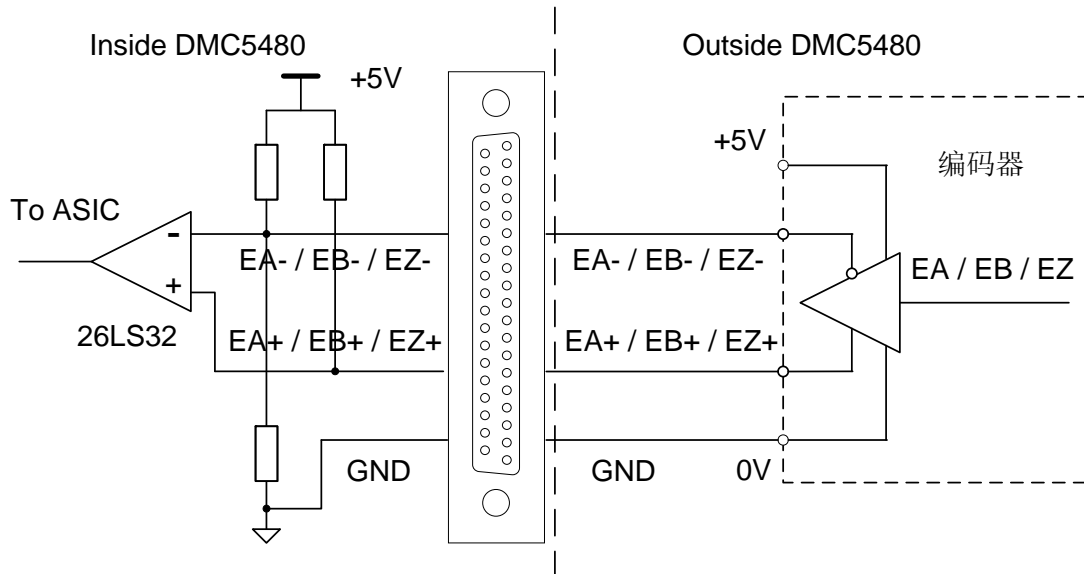


图2-22 差分输出编码器接线原理图

2. 集电极开路模式

如图 2-23 所示，如果使用集电极开路输出的编码器，则编码器输出信号接 EA+/EB+/EZ+端，而 EA-/EB-/EZ-端悬空。

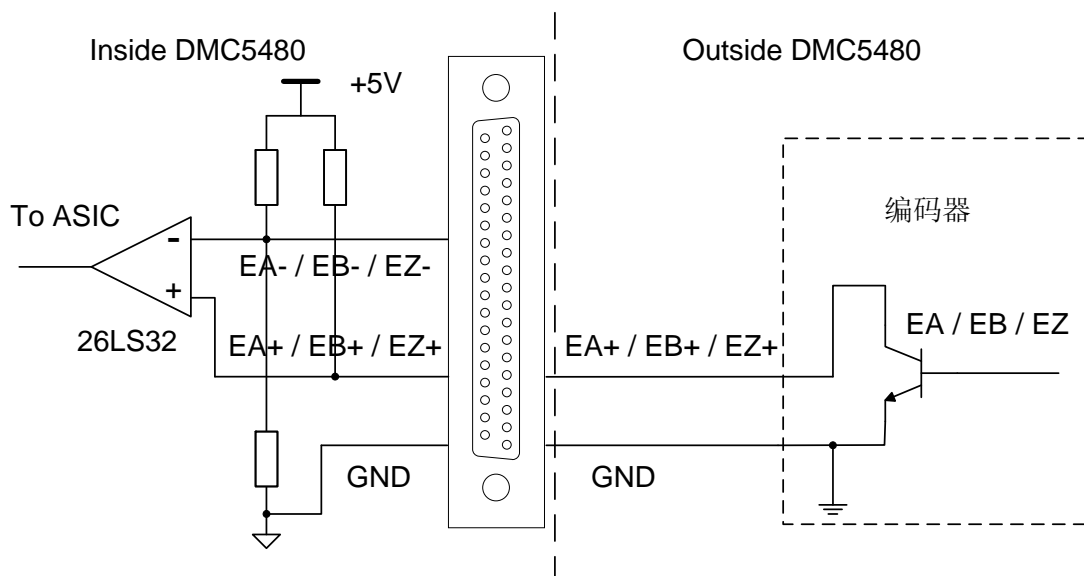


图2-23 集电极开路输出的编码器接线原理图

注意：

1. 编码器等脉冲输入信号的 EA+、EA-、EB+、EB- 和 EZ+、EZ- 的差分信号电压差必须高于 3.5V，小于 5V，且输出电流不应小于 6mA。
2. 需要将输入设备的地线和控制卡的 GND 连接。

编码器接口位于控制卡扩展接口 DIP40 插座 X2 上，其对外接口 DB37 具体引脚分配见 [8.1.2 接口 X2 引脚定义](#)。

2.2.6.2 位置锁存信号输入接口

雷泰（Leadtech）DMC-5480 运动控制卡每一轴都提供一个位置锁存输入信号 LTC，LTC 信号通过触发位置锁存器，捕获当前编码器位置或当前指令位置。LTC 信号接口原理图如图 2-24 所示：

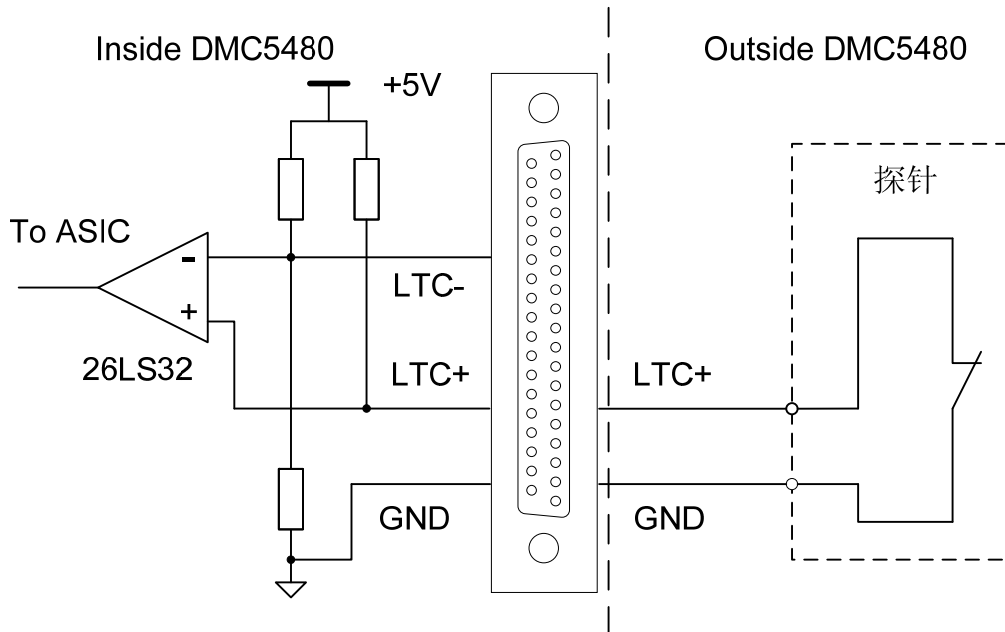


图2-24 位置锁存输入信号接口原理图

LTC1~LTC4 信号可以分别锁存 4 个轴的位置；也可以通过软件设置，由 LTC1 信号同时锁存 4 个轴的位置。该信号接口位于控制卡扩展接口 DIP40 插座 X2 及 X3 上，其对外接口 DB37 具体信号引脚分配见 [8.1.2 接口 X2 引脚定义](#) 以及 [8.1.3 接口 X3 引脚定义](#)。

2.2.6.3 位置比较输出信号接口

雷泰（Leadtech）DMC-5480 运动控制卡每个轴均配有 2 个位置比较输出接口。通过软件使能后，可分别设置触发条件，当某个轴的指令寄存器内的数值或编码器寄存器内数值满足触发条件时，硬件自动在 CMP 端口上输出一个开关信号。具体引脚分配请参照对外接口 SCSI68 插座 [8.1.1 接口 X1 引脚定义](#) 及 [8.1.3 接口 X3 引脚定义](#)。典型接口原理图如下图 2-25 所示：

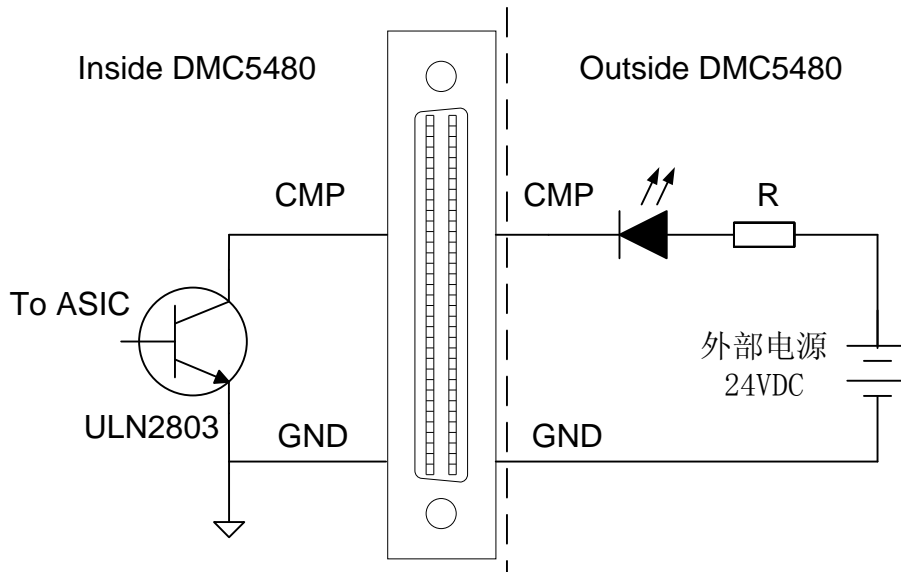


图2-25 位置比较输出信号原理图

2.2.7 手轮脉冲输入接口

雷泰（Leadtech）DMC-5480 运动控制卡为每个轴提供了手摇脉冲发生器输入接口PA（PB），典型接线原理如下图 2-26 所示。用户可以通过PA（PB）输入脉冲信号控制电机的运动，电机的运动距离和转速受输入的脉冲数和脉冲频率控制。具体引脚端口定义见[8.1.5 接口X5 引脚定义](#)，具体输入模式设置和运动控制功能使能见[4.2.7 手轮运动控制](#)。

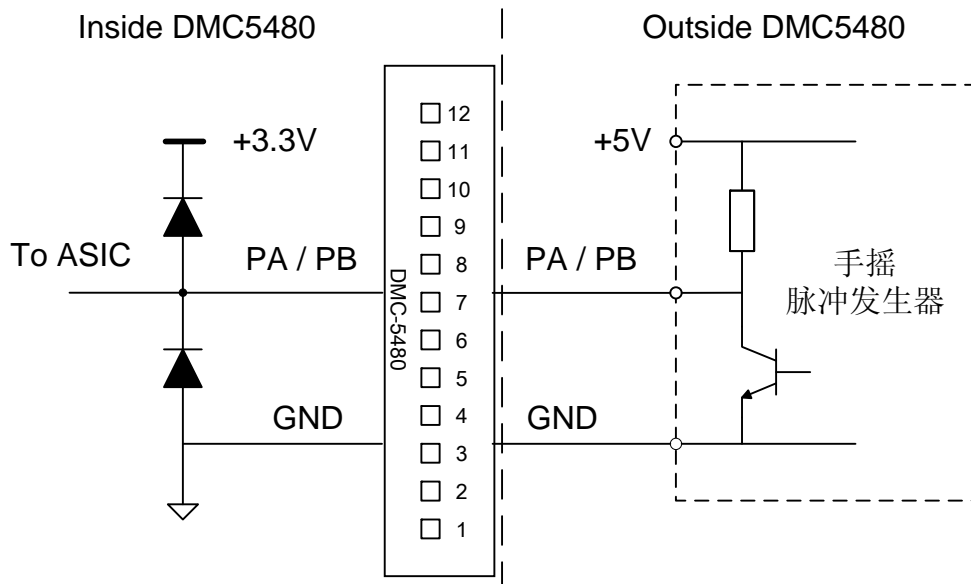


图2-26 外部脉冲信号输入原理图

2.2.8 通用数字输入/输出信号接口

雷泰（Leadtech）DMC-5480 运动控制卡除了提供专用的数字 I/O 接口外还提供了大量的通用数字 I/O 接口。包括最多 36 路通用数字输入信号和最多 32 路的通用输出信号，同时 SEVON、INP、ALM、ERC 专用 I/O 不做专用时也可作为通用数字 I/O 接口。

输出口 1~12 初始电平状态可选，可通过拨码开关来设置。改变初始电平后，输出口的电平逻辑会取反。具体设置方法请参考[3.1.2 开关配置](#)中说明。

2.2.8.1 INPUT通用数字输入信号接口

DMC-5480 卡为用户提供了最多 36 路通用数字输入信号(24 路通用 / 12 路专用信号接口如果不使用的话也可以当作通用输入信号使用)。用于开关信号、传感器信号或其它信号的输入。其接口电路加有光电隔离元件，可以有效隔离外部电路的干扰，以提高系统的可靠性。通用数字输入信号接口原理图如图 2-27 所示：

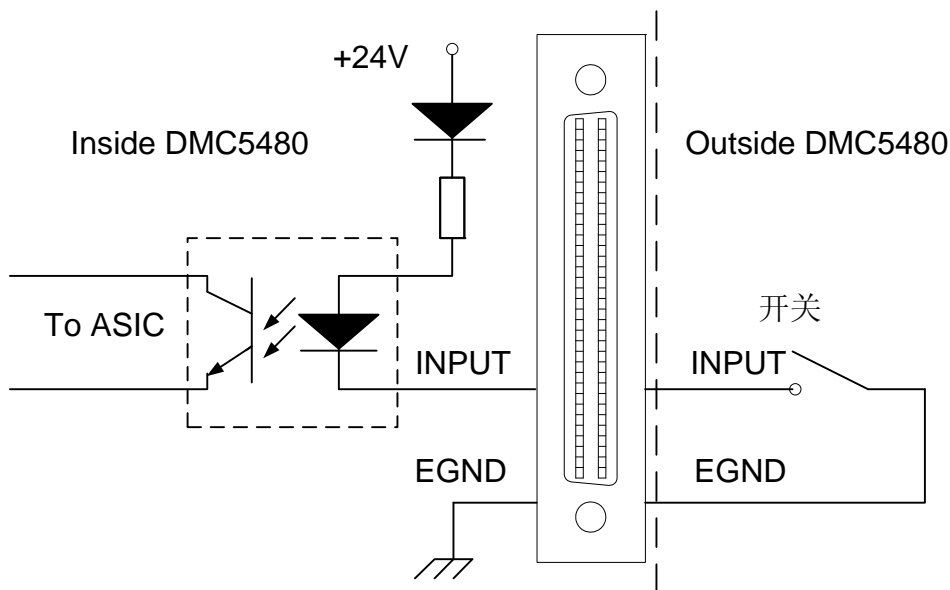


图2-27 通用输入口电路原理图

2.2.8.2 通用数字输出信号接口OUTPUT

DMC-5480 卡为用户提供了最多 32 路通用数字输出信号(20 路通用 / 12 路专用不使用的話也可以当作通用输出信号使用)，由 ULN2803 驱动，可用于对继电器、电磁阀、信号灯或其它设备的控制。

OUT 1~OUT 12 端口可设置上电时的初始电平，详见 3.3 节拨码开关 S2 的设置。OUT13~OUT20 上电初始电平为高。

DMC-5480 卡的通用数字输出信号控制常用元器件的接法如下：

- 1、发光二极管

通用数字输出端口控制发光二极管时，需要接一限流电阻 R ，限制电流在 10mA 左右，电阻值大约在 2K 到 5K 左右，根据使用的电源来选择，电压越高，使用的电阻值越大些。

原理图如图 2-28 所示：

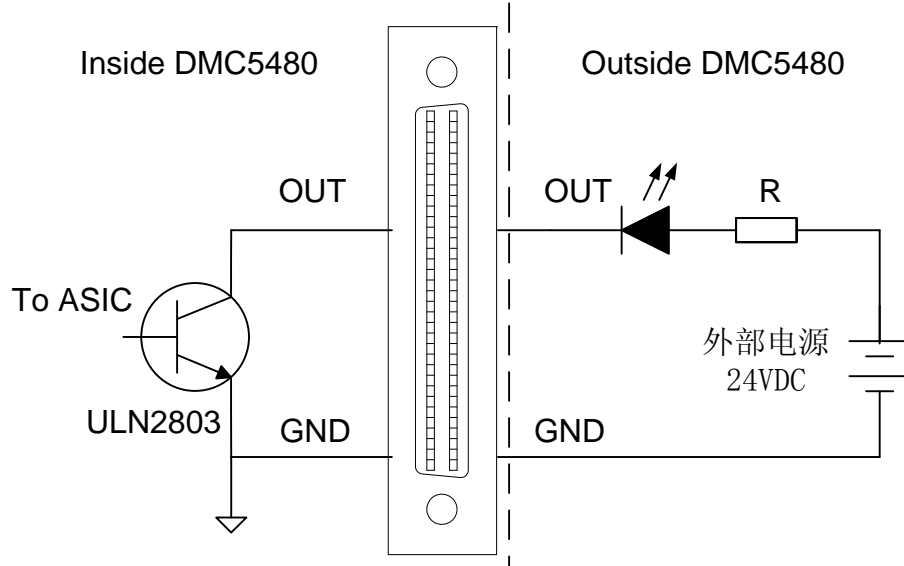


图2-28 输出口接发光二极管原理图

2、灯丝型指示灯

通用数字输出端口控制灯丝型指示灯时，为提高指示灯的寿命，需要接预热电阻 R ，电阻值的大小，以电阻接上后，输出口为 1 时，灯不亮为原则。

原理图如图 2-29 所示：

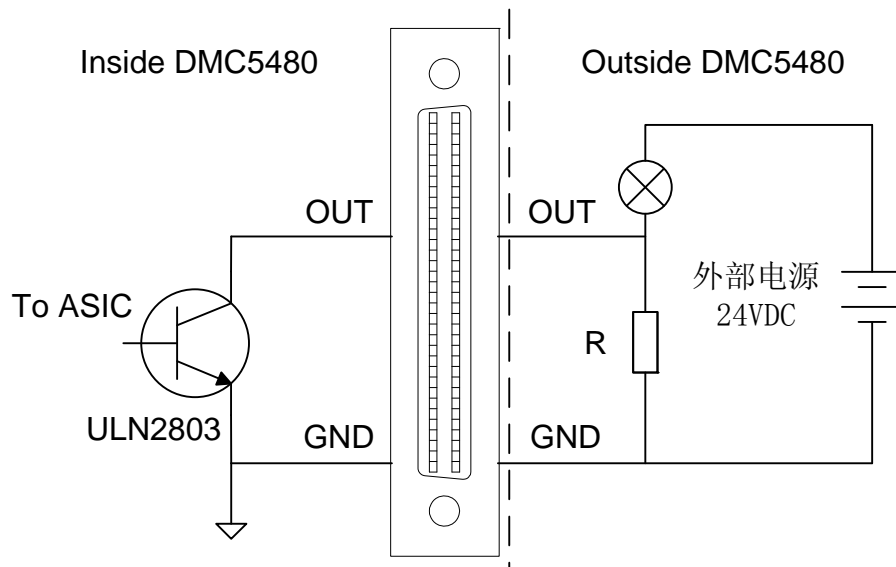


图2-29 输出口接指示灯原理图

3、小型继电器

继电器为感性负载，必须并联一个续流二极管，以保护 DMC-5480 卡的输出驱动元件 ULN2803。

继电器接线图如图 2-30 所示：

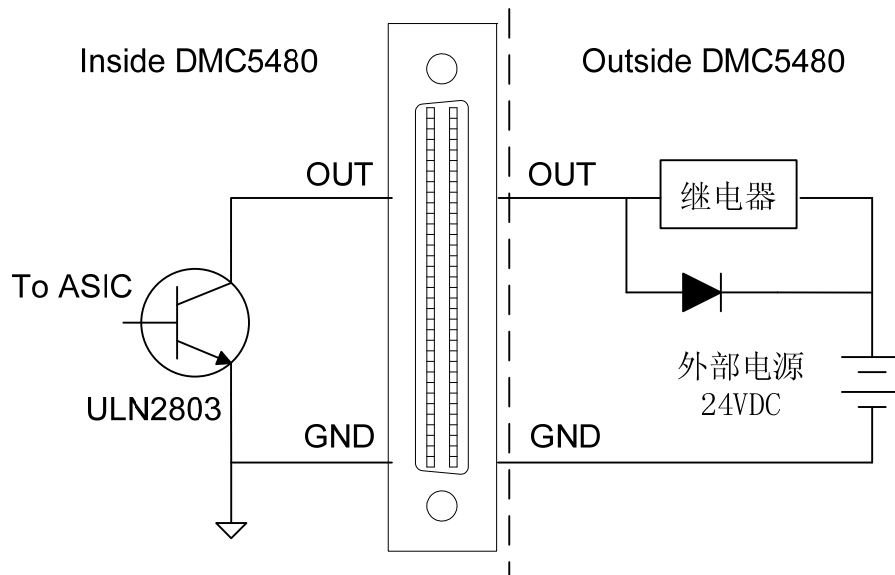


图2-30 接小型继电器原理图



注意： 1. 外部电源（如上图中的+24V）的地线必须与端子板上的 GND 端口相连。

2. 在使用通用数字输出端口时，切勿把外部电源直接接至通用数字输出端口上，否则，会损坏 ULN2803。

运动控制平台位置传感器及控制信号布局示例

雷泰（Leadtech）DMC-5480 运动控制卡每轴都配有 2 个限位信号、1 个原点信号。每路信号都进行了光电隔离，以减少外界对内部的干扰，保证动作的可靠。

图 2-31 为一般运动平台位置传感器及控制信号的布置图。

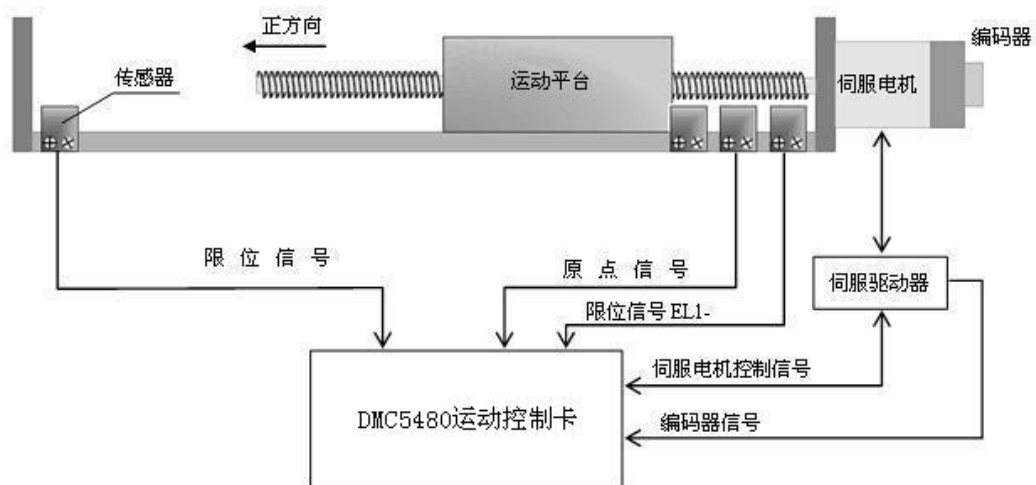


图 2-31 运动平台位置传感器及控制信号的布置图

3 硬件配置与安装

3.1 硬件配置

雷泰（Leadtech）DMC-5480 运动控制卡上有多组跳线开关 J1-J8 和拨码开关 S1-S2，分别用于设置 DMC-5480 卡的工作方式和参数。

DMC-5480 板卡跳线、开关布局如图 3-1 所示：

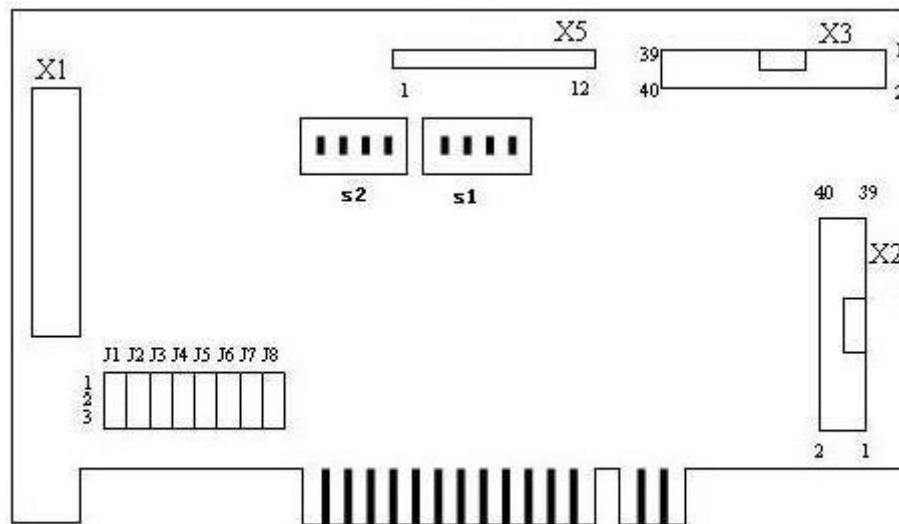


图3-1 DMC-5480板卡跳线、开关布局

3.1.1 跳线配置

跳线开关和拨码开关用于设置 DMC-5480 卡的工作方式：

J1~J8 跳线开关用于设置脉冲信号输出方式为差分输出或单端输出，其设置方式如图 3-2 和图 3-3 所示；各轴脉冲和方向信号和跳线开关的对应关系如表 3-1 所示。

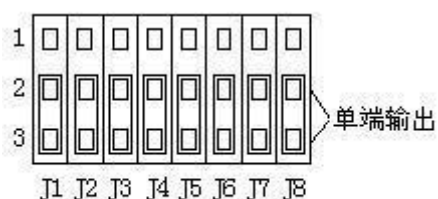


图3-2 单端输出的跳线设置

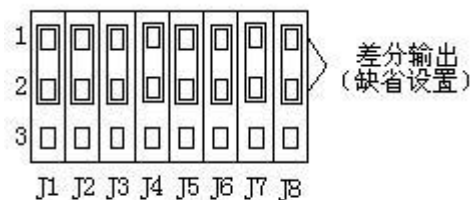


图3-3 差分输出方式的跳线设置

表 3-1 脉冲和方向信号与跳线开关的对应关系表

X1 引脚号	信号	跳线开关	X1 引脚号	信号	跳线开关
1	PUL1+	J1	35	PUL3+	J5
2	PUL1-		36	PUL3-	
3	DIR1+	J2	37	DIR3+	J6
4	DIR1-		38	DIR3-	
5	PUL2+	J3	39	PUL4+	J7
6	PUL2-		40	PUL4-	
7	DIR2+	J4	41	DIR4+	J8
8	DIR2-		42	DIR4-	

出厂时的缺省设置 J1~J8 全为 2-1 短路，即差分输出方式。

3.1.2 开关配置

DMC-5480 卡中配置了两个 4 位的拨码开关 S1 和 S2，具体说明如下。

(1) S1 设置控制卡标识值：

S1 的 1 位：卡标识位 1；

S1 的 2 位：卡标识位 2；

S1 的 3 位：卡标识位 3；

S1 的 4 位：保留。

当同时使用多张 DMC-5480 卡时，可设置卡标识来区分不同的卡，设置对应的标识值如下表 3-2：

表 3-2 设置控制卡标识值

卡标识位 3 位置	卡标识位 2 位置	卡标识位 1 位置	卡标识值
ON	ON	ON	7
ON	ON	OFF	6
ON	OFF	ON	5
ON	OFF	OFF	4
OFF	ON	ON	3
OFF	ON	OFF	2
OFF	OFF	ON	1
OFF	OFF	OFF	0

(2) S2 是 OUT1~OUT16 及 SEVON1~SEVON4 上电平初始值、输出端口电平逻辑的选择开关。

选择“ON”时，输出初始电平为高；在程序中对输出口写“0”，端口输出低电平。

选择“OFF”时，输出初始电平为低；在程序中对输出口写“0”，端口输出高电平。

S2 的 1 位：设置 OUT1~OUT4 端口。

S2 的 2 位：设置 OUT5~OUT8 端口。

S2 的 3 位：设置 OUT9~OUT12 端口。

S2 的 4 位：设置 OUT13~OUT16, SEVON1~SEVON4 端口。

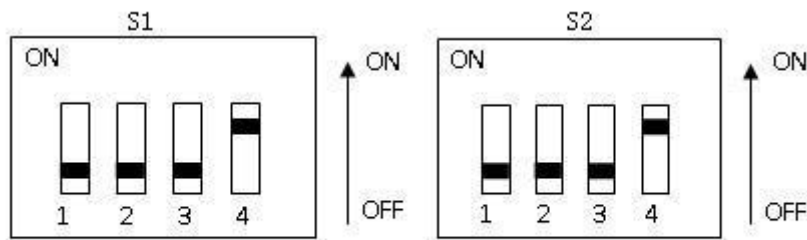


图3-4 DMC-5480 的拨码开关设置示意图

拨码开关出厂默认值设定全部为 ON，即 SEVON1~SEVON4 和 OUT1~OUT16 上电初始电平为高电平。

其它输出口初始电平固定为高电平，不能配置。

3.2 硬件安装

雷泰(Leadtech)DMC-5480 运动控制卡硬件结构遵从 32bit PCI 卡结构标准，其安装方法类同普通 32bit PCI 卡的安装，具体参考步骤如下：

- 1) 打开DMC-5480 的包装，参考[3.1 硬件配置](#)的说明，按照实际应用的需求，完成硬件配置；
- 2) 使用辅助接口的用户，请将辅助接口与 DMC-5480 对应的插座连接，并确保连接牢固，可靠；
- 3) 触摸地线，完全释放操作员身上的静电，带好防静电手套；
- 4) 确定 PC 机已经关闭，以及一切与 PC 相连的设备也已关闭；
- 5) 打开 PC 机的机箱；
- 6) 选择一个靠近处理器的 32bit PCI 插槽，将 DMC-5480 垂直插入插槽中；
- 7) 将 DMC-5480 用螺钉紧固在 PC 机机箱上，确保紧固、可靠。
- 8) 使用辅助接口的用户，请将辅助接口也用螺钉紧固在 PC 机机箱上，确保紧固、可靠。
- 9) 盖上 PC 机机箱，至此硬件安装完毕！

4 软件系统概述

雷泰(Leadtech) DMC-5480 运动控制卡软件系统包括：硬件驱动程序、运动控制函数库、演示程序、例子程序。

4.1 硬件驱动程序

雷泰(Leadtech) DMC-5480 配套提供Windows7/XP NT/2000 等操作系统环境下的驱动程序。客户可以根据自己的需要选择相应的系统平台来开发适合自己的应用软件。硬件驱动程序具体安装方法请参考：[5 驱动程序安装](#)。

4.2 运动控制函数库

雷泰（Leadtech）DMC-5480 为使客户能够开发适合自己的应用控制系统，提供了丰富的功能函数库，客户可以根据自己应用系统的需要灵活调用不同的功能函数。

注：各函数具体功能介绍参考附录：[8.2 运动控制函数库](#)。

4.2.1 初始化、关闭运动控制卡

在操作雷泰(Leadtech) DMC-5480 运动控制卡之前，必须调用控制卡初始化函数为运动控制卡分配资源。同样，当结束对运动控制卡的操作时，必须调用控制卡关闭函数释放运动控制卡所占用的系统资源，使得所占资源可被其它设备使用。具体相关函数和功能如表 4-1 所示：

表 4-1 初始关闭控制卡函数说明

	名称	功能	参考
1	d5480_board_init()	初始化 DMC-5480 并分配系统资源	8.2.2.1
2	d5480_board_close	关闭 DMC-5480 并释放系统资源	8.2.2.1
注意：程序结束时，必须调用 d5480_board_close()函数释放系统资源。			

例程：初始化和关闭控制卡（以标准 C 语言为例说明，下同）

.....

```
CardCount = d5480_board_init();  
if(CardCount == 0)  
{  
printf(“\n 没有发现运动控制卡”);  
getch();  
return();  
}
```

.....

```
d5480_board_close();
```

```
.....
```

4.2.2 设置脉冲输出模式

雷泰（Leadtech）DMC-5480 运动控制卡使用指令脉冲方式控制步进/伺服电机。市面上的众多电机驱动器厂家信号接口要求各有不同（常用的有六种类型），所以在使用控制卡控制具体的电机驱动器时，必须对脉冲输出方式进行正确的设定，电机才能正常工作。具体具体相关函数和功能如表 4-2 所示：

表 4-2 脉冲设置函数说明

	名称	功能	参考
1	d5480_set_pulse_outmode	设置指定轴的脉冲输出模式	8.2.2.2
2	d5480_get_pulse_outmode	读取指定轴的脉冲输出模式	8.2.2.2
注意：在调用运动控制函数之前应先调用该函数来设置指令脉冲模式。			

指令脉冲包括两项基本信息：电机运转距离即脉冲数和电机转动方向。有两种基本指令模式：两种基本模式如表 4-3 所示：

表 4-3 两种基本的指令脉冲输出方式

模式	PULn-脚输出	DIRn-脚输出
脉冲/方向（PULSE/DIR）	脉冲信号	方向信号（电平）
双脉冲（CW/CCW）	正向（CW）脉冲	反向（CCW）脉冲
注：具体怎样设置请参考d5480_set_pulse_outmode函数具体说明 8.2.2.2 。		

4.2.2.1 脉冲 / 方向模式

在此模式下，PULn-输出指令脉冲串，脉冲数对应电机运行的相应“距离”，而脉冲频率对应电机运行“速度”；DIRn-输出方向信号，该信号的不同电平对应电机不同的转动方向。此种模式在驱动器中最多。

脉冲信号可以设置为上升沿有效（即脉冲信号常态为低电平，变化为高时电机走一步）；也可设置为下降沿有效（即脉冲信号常态为高电平，变化为低时电机走一步）。方向信号可设置为高电平对应正向或低电平对应正向两种选择。所以实际上此种模式下有四种指令类型，如图 4-1 所示：

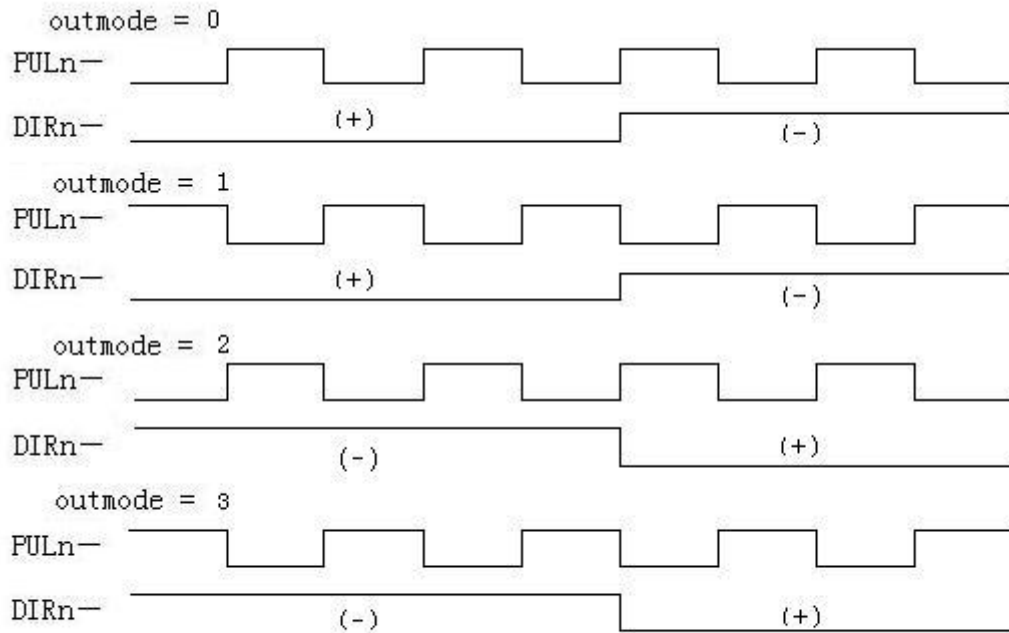


图4-1 单脉冲输出模式

4.2.2.2 双脉冲模式

在此模式下，PULn-和 DIRn-引脚分别表示正向（CW）和反向（CCW）脉冲输出。从PULn-引脚输出的脉冲使电机正转，而DIRn-引脚输出的脉冲使电机反转。脉冲信号有上升沿有效或下降沿有效的选择，所以此模式下共有两种指令类型，如图4-2所示：

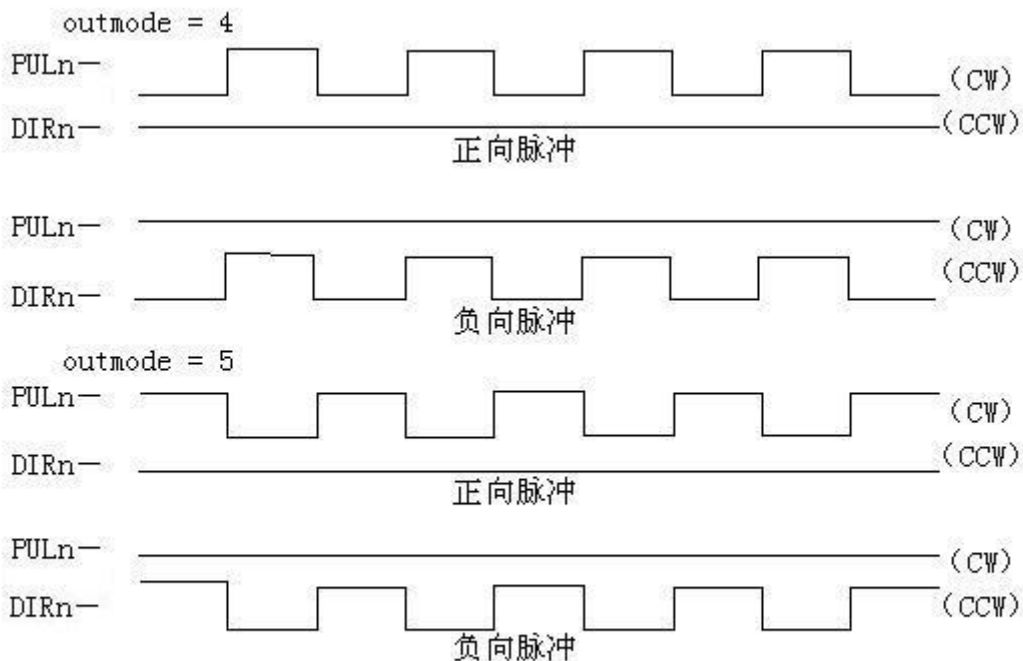


图4-2 双脉冲输出模式

例程：设置脉冲输出方式

.....

d5480_set_pulse_outmode (0,0); //设置第 0 轴脉冲输出模式为单脉冲模式，PULn-信号上升沿有效，DIRn-正向为低电平。

d5480_set_pulse_outmode (1,4); //设置第 1 轴脉冲输出模式为双脉冲模式，上升沿有效。

.....

4.2.3 回原点运动

在进行精确的运动控制之前，需要设定运动坐标系的原点。运动系统动作时通常会执行寻找该原点的动作讲系统坐标位置归零，即回原点运动。雷泰 DMC-5480 运动控制卡提供多种回原点运动的方式，下面介绍 2 种常用的回原点的方式：

方式 1：一次回零

该方式以低速回原点，适合于行程短、安全性要求高的场合。动作过程为：电机从初始位置以恒定低速度向原点方向运动，当到达原点开关位置，原点信号被触发，电机立即停止（过程 0）；将停止位置设为原点位置，如图 4-3 所示。



图4-3 回原点方式0

方式 2：两次回零

该方式为方式 1 和方式 3 的组合。先进行方式 3 的回零加反找，完成后再进行方式 1 的一次回零。可参见方式 1 和方式 3 的说明。

方式 3：一次回零加回找

该方式在回原点运动过程中，当找到原点信号后，还要等待所设定的该轴的 EZ 信号出现次数满足后，电机停止。若设定 EZ 信号出现次数为 1 次，则回原点过程如图 4-4 所示。

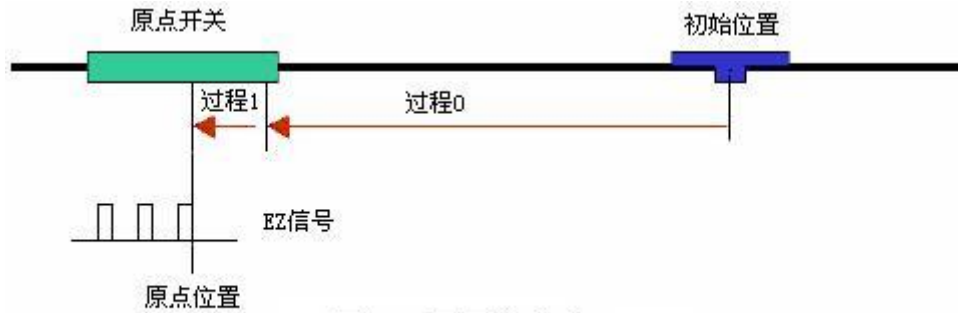


图4-4 回原点方式1

方式 10：以 EZ 作为原点进行一次回零

回零之前需要清除 EZ 状态，当 EZ 信号到来时，回零减速停止。

方式 11：以 EZ 作为原点进行一次回零，碰到限位后自动反找

回零方式类似方式 10，在其基础上增加了碰到限位后的一次反向回零的过程。

雷泰（Leadtech）DMC-5480 运动控制卡回原点的相关函数如下表 4-4 所示：

表 4-4 回原点相关函数说明

	名称	功能	参考
1	d5480_set_HOME_pin_logic	设置原点信号的电平和滤波器使能。	8.2.2.3
2	d5480_config_home_mode	选择回原点模式。	8.2.2.3
3	d5480_home_move	按指定的方向和速度方式开始回原点。	8.2.2.3
注意：执行完 d5480_home_move 函数后，指令脉冲计数器不会自动清零；如需清零可以在回零运动完成后，调用 d5480_set_position 函数软件清零。			

例程：一次回零

```

.....

d5480_set_HOME_pin_logic(0,0,0);    //设置0号轴的原点信号低电平有效
d5480_config_home_mode(0,2, 2000, 1,0); //设置0号轴模式为一次回零，负
方向回原点
d5480_home_move(0);
while (d5480_check_done(0) == 0)    //等待回原点动作完成
{
}
d5480_set_position(0,0);             //设置0号轴的指令脉冲计数器绝对
位置为 0
.....

```

4.2.4 位置计数锁存比较

雷泰（Leadtech）DMC-5480 运动控制卡每个轴均有命令位置计数器和反馈位置计数器，命令位置计数器用于监测指令位置，反馈位置计数器用于监测机械

位置，同时提供位置锁存和位置比较输出功能。

4.2.4.1 命令位置计数器

指令位置计数器是一个 32 位正负计数器，对控制卡输出的指令脉冲进行计数。当输出一个正向脉冲后，计数器加 1；当输出一个负向脉冲后，计数器减 1。

相关函数如表 4-5 所示：

表 4-5 指令脉冲位置相关函数说明

	名称	功能	参考
1	d5480_get_position	读取指定轴的指令脉冲计数器。	8.2.2.4.1
2	d5480_set_position	设置指定轴的指令脉冲计数器。	8.2.2.4.1

例程：位置操作

```
d5480_set_postion(0,100);           //设置轴 0 的脉冲位置为 100
position = d5480_get_position(0);    //读轴 0 的当前位置值至变量 position
```

4.2.4.2 反馈位置计数器

位置反馈计数器是一个 28 位正负计数器，对通过控制卡编码器接口 EA，EB 输入的脉冲（如编码器、光栅尺反馈信号等）进行计数。

相关函数如表 4-6 所示：

表 4-6 编码器相关函数说明

	名称	功能	参考
1	d5480_counter_config()	设置编码器输入口的计数方式。	8.2.2.4.2
2	d5480_get_encoder()	读取编码器反馈的脉冲计数值。	8.2.2.4.2
3	d5480_set_encoder()	设置编码器的脉冲计数值。	8.2.2.4.2

例程：编码器反馈计数的操作

```
d5480_counter_config (0,3);          //设置轴 0 为 4 倍计数，默认的 EA、EB
计数方向
d5480_set_encoder(0,0);              //设置轴 0 的计数初始值为 0
X_Position = d5480_get_encoder(0);    //读轴 0 的计数器的数值至变量
X_Position
```

4.2.4.3 位置锁存

雷泰（Leadtech）DMC-5480 运动控制卡提供编码器计数值锁存功能，该功能广泛应用于各种测量行业。位置锁存方式可以选择对每个编码器信号独立锁存，也可以通过任一个锁存端口对全部编码器计数值同时锁存，触发指令 LTC 接口一般接测量探头的触发信号。该功能用于位置测量十分准确、方便。在复位触发标志位后，当锁存信号被触发，当前编码器计数值立即被捕获至位置锁存器中，并将触发标志位置位，保护当前锁存器内的数值，直到触发标志位被再次复

位。相关函数如下表 4-7 所示：

表 4-7 位置锁存相关函数说明

	名称	功能	参考
1	d5480_config_latch_mode	设置 LTC 端口的锁存方式。	8.2.2.4.3
2	d5480_get_latch_value	读取被触发锁存到锁存器内的编码器计数值。	8.2.2.4.3
3	d5480_get_latch_flag	读取指定控制卡的锁存器的标志位。	8.2.2.4.3
4	d5480_reset_latch_flag	复位指定控制卡的锁存器的标志位。	8.2.2.4.3

例程：位置锁存的操作(C 语言)

```
int key=0;                                //锁存状态标识
long LatchStatus;                         //触发状态标识
long xValue[100],yValue[100],zValue[100]; //存放锁存值的数组

int g_count=0;                            //计数器
d5480_config_latch_mode(0,1);             //设置为四轴同时锁存
while(g_count<100)                       //假设测 100 点
{
    LatchStatus = d5480_get_latch_flag(0); //读取锁存器状态
    if ( ( (LatchStatus & 0xF00) != 0) && key==0)
    {
        xValue[g_count] = d5480_get_latch_value(0); //获得锁存值
        yValue[g_count] = d5480_get_latch_value(1);
        zValue[g_count] = d5480_get_latch_value(2);
        g_count++;                                //计数器加 1
        key = 1;                                //标识置位
    }
    else if ( ( (LatchStatus & 0xF) == 0) && key==1)
    {
        d5480_reset_latch_flag(0);               //锁存器复位
        key = 0;                                //标识复位
    }
}
```

4.2.4.4 位置比较输出

雷泰（Leadtech）DMC-5480 运动控制卡提供了位置比较输出的功能，单卡可以一次设置 128 个比较点，同时可以配置比较条件和触发动作。

相关函数如表 4-8 所示：

表 4-8 位置比较相关函数说明

	名称	功能	参考
1	d5480_compare_config	配置比较器。	8.2.2.4.4
2	d5480_compare_get_config	读取配置比较器。	8.2.2.4.4
3	d5480_compare_clear_points	清除所有比较点。	8.2.2.4.4
4	d5480_compare_add_point	添加位置比较点。	8.2.2.4.4
5	d5480_compare_get_current_point	读取当前比较点。	8.2.2.4.4
6	d5480_compare_get_points_runned	查询已经比较过的点。	8.2.2.4.4
7	d5480_compare_get_points_remained	查询可以加入的比较点数量。	8.2.2.4.4

例程：位置比较输出操作

```
d5480_compare_clear_points(0);  
d5480_set_position(0, 0);  
d5480_compare_config (0,1, 0, 0 ); //设置 0 号 card 的位置比较功能  
d5480_compare_add_point(0, 10000, 1, 3, 1); //添加 0 号 card 的比较点，触发  
动作作为反向 OUT1 口  
d5480_compare_add_point(0, 20000, 1, 3, 1); //添加 0 号 card 的比较点，触发  
动作作为反向 OUT1 口  
d5480_compare_add_point(0, 30000, 1, 3, 1); //添加 0 号 card 的比较点，触发  
动作作为反向 OUT1 口  
d5480_set_profile(0, 0, 1000, 500, 500);  
d5480_pmove(0, 50000, 1); //启动轴 0 运动，比较点触发 OUT 端口状态变化
```

4.2.5 单轴多轴运动控制

4.2.5.1 单轴运动控制

雷泰（Leadtech）DMC-5480 运动控制卡在表述运动轨迹时可以用绝对坐标和相对坐标 2 种模式（图 4-5 所示）。2 种模式各有优点。如：在绝对坐标模式中用一系列坐标点定义一条曲线，如果要修改中间某点坐标时，不会影响后续点的坐标；在相对坐标模式中，用一系列坐标点定义一条曲线，用循环命令可以重复这条曲线轨迹多次。

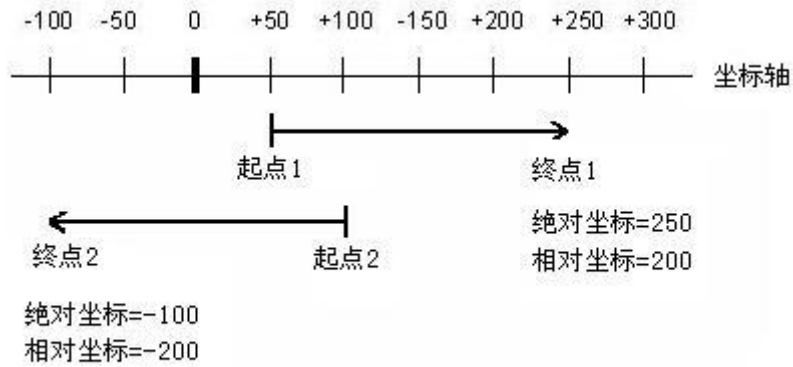


图4-5 绝对坐标与相对坐标中轨迹终点的不同表达方式

在 DMC-5480 函数库中距离或位置的单位为脉冲；速度单位为脉冲/秒；时间单位为秒。最基本的位置控制是指从当前位置运动到另一个位置，一般称为点位运动或定长运动。

DMC-5480 卡在执行单轴控制时，可使电机按照梯形速度曲线或 S 形速度曲线进行点位运动或连续运动。

4.2.5.1.1 梯形速度曲线运动模式

梯形速度曲线运动是位置控制中最基本的运动模式。在此模式下移动一段指定距离时，其运动速度按梯形曲线变化，如图 4-6 所示。

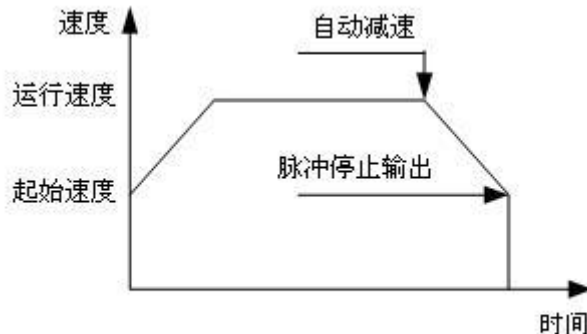


图4-6 简单的梯形速度曲线

运动速度之所以要按梯形曲线变化，是因为：电机转轴和被拖动的物体具有惯性，不可能在瞬间内达到指定速度，因此应该给予一定的加速时间；减速时亦是类似，否则电机因为瞬间力矩不足而出现丢步、过冲（步进系统）或振荡（伺服系统）现象。

实现以梯形速度曲线运动的点位控制函数如表 4-9 所示：

表 4-9 梯形点位控制相关函数说明

	名称	功能	参考
1	d5480_set_profile	设定梯形速度曲线的起始速度、运行速度、加速时间、减速时间。	8.2.2.5.1
2	d5480_pmove	让指定轴以对称梯形曲线速度作点位运动。	8.2.2.5.1

例程：执行以非对称梯形速度曲线作点位运动

.....

d5480_set_profile(0,500,6000,300000,600000); //设置 0 号轴起始速度为 500 脉冲/秒、运行速度为 6000 脉冲/秒、加速时间为 0.02 秒、减速时间为 0.01 秒。

d5480_pmove(0,50000,0); //设置 0 号轴、运动距离为 50000 个脉冲、相对坐标，并开始执行运动

.....

在单轴运行过程中，运动速度 Max_Vel 和目标位置 Dist 均可以实时改变如图 4-7 所示。

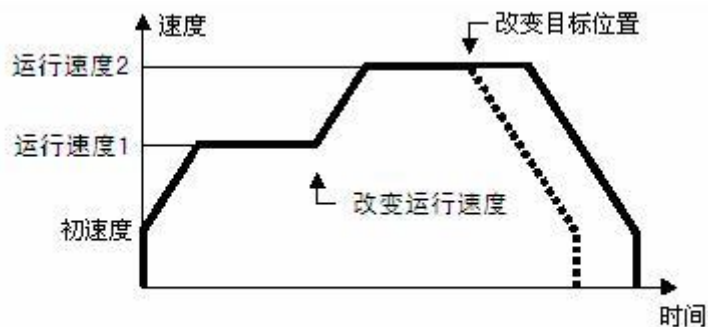


图4-7 改变速度及改变目标位置

若在减速时改变目标位置，电机的速度将如图 4-8 所示发生变化。

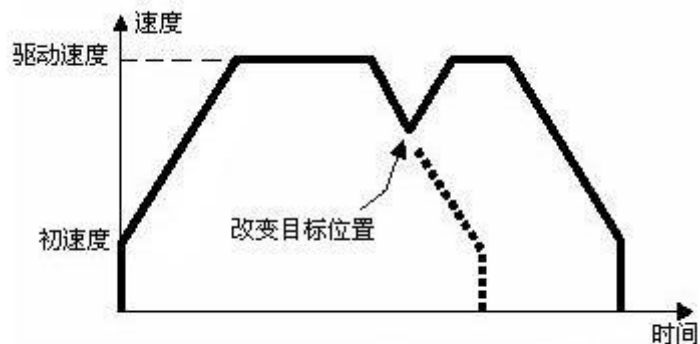


图4-8 改变速度及改变目标位置

实现这 2 个功能的函数如表 4-10 所示：

表 4-10 梯形点位控制相关函数说明

	名称	功能	参考
1	d5480_change_speed	单轴运行中改变当前运行速度的函数。	8.2.2.5.3
2	d5480_reset_target_position	改变目标位置函数。	8.2.2.5.1

例程：改变速度、改变终点位置

.....

/*设置梯形曲线速度、加、减速时间*/

d5480_set_profile(0,500,6000,60000,300000);

```
d5480_pmove(0,50000,0);           //设置轴号、运动距离 50000、相
对坐标模式

If(“改变速度条件”)                //如果改变速度条件满足，
则执行改变速度命令
{
    Curr_Vel= 9000;                 //设置新的速度
    d5480_change_speed(0,Curr_Vel); //执行改变速度指令
}

If(“改变终点位置条件”)            //如果改变终点位置条件满足，
则执行改变终点位置命令
{
    d5480_reset_target_position(0,55000); //改变终点位置为 55000
}
.....
```

如果将运动中的运行速度设置得小于起始速度，整个运动过程中将会以起始速度作恒速运动，如图 4-9 所示：

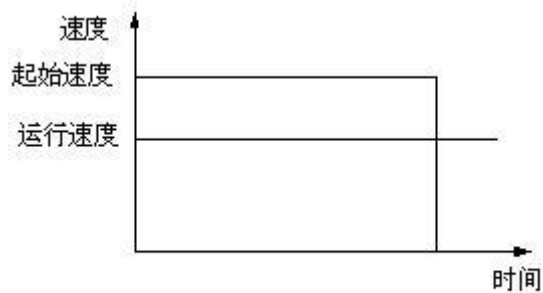


图4-9 运行速度小于起始速度导致恒速

如果运动距离很短，当距离小于或等于 $(Max_Vel+Min_Vel)*Tacc$ 时，理论上速度曲线将变为三角形；但 DMC-5480 运动控制卡有自动调整功能，将三角形的尖峰去，以避免速度变化太大发生冲击现象。请参见图 4-10 所示：

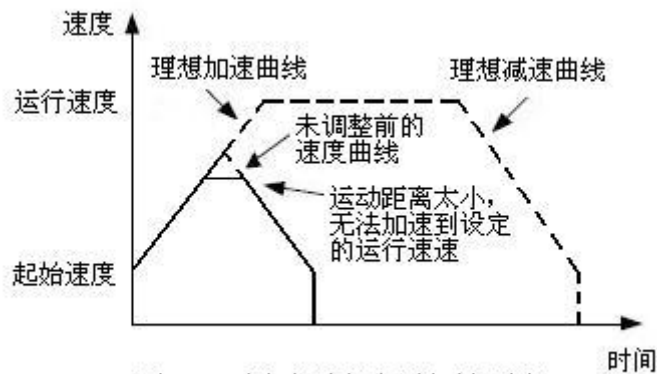


图4-10 运动距离小于加速距离

控制轴运动的指令发出后，可以用d5480_check_done函数检测电机当前运动状态。该函数具体定义可参考[8.2.2.5.2](#)小节。

4.2.5.1.2 S形速度曲线运动模式

梯形速度曲线虽然简单，但它的速度曲线不平滑，其加速度有突变，因而运动中有冲击现象，容易引起机器噪声和传动机构的磨损。若将加速度改为线性变化，则速度曲线相应将变得光滑，如[2.2.2.2 速度控制](#)中的B部分S形速度曲线说明所示。升速和减速阶段均变得象S形形状。采用此种速度曲线，运动更平稳，且有助于缩短加速过程、降低运动装置的振动和噪声，以及延长机械传动部分的寿命。

设置 S 形速度曲线及其点位运动的函数如表 4-11 所示：

表 4-11 S 形速度控制相关函数说明

	名称	功能	参考
1	d5480_set_s_profile	设置 S 形速度曲线的 S 段参数。	8.2.2.5.1
2	d5480_pmove	让指定轴以对称 S 形速度曲线作点位运动。	8.2.2.5.1

注意：

a.由于执行 S 形速度曲线运动时机器的振动较小，用户可以加大加速度，即提高速度曲线上线性升速区域的斜率，从而缩短加速或减速时间，从而缩短整个运动的时间。因此，S 形曲线在运动速度要求十分高的设备中被广泛使用；

b.使用 S 形曲线的目的是产生平滑运动，但是如果因为距离太短或加速太慢原因导致电机速度在加速段不能升至设定的最大值 Max_Vel 时，理论上加速段将突然切换至减速段，从而导致在速度曲线的中部出现尖三角，并因此引起该轴出现较大震动和相关问题。为了避免出现这种问题，DMC-5480 内置有自动调整功能，使得加减速段的过渡保持平滑，如图 4-11 所示。

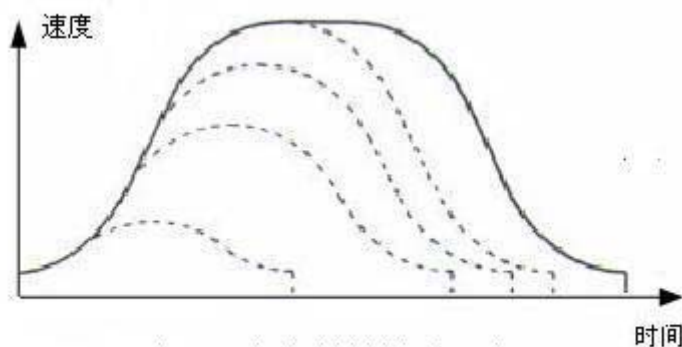


图4-11 自动降速避免尖三角

单轴运行情况下，S 形速度曲线运动过程中也可以调用 d5480_change_speed 和 d5480_reset_target_position 函数实时改变运行速度和目标位置。注意多轴插补运行情况下不能实时改变运行速度和目标位置。

4.2.5.1.3 连续运动模式

连续运动模式中，DMC-5480 控制卡可以控制电机以梯形或 S 形速度曲线在指定的加速时间内从起始速度加速至运行速度，然后以该速度连续运行，直至调用停止指令或者该轴遇到限位信号才会按启动时的速度曲线减速停止。连续运动的函数如表 4-12 所示：

表 4-12 连续运动相关函数说明

	名称	功能	参考
1	d5480_vmove	让指定轴加速到指定的运行速度后，连续运行。	8.2.2.5.3
2	d5480_decel_stop	指定轴减速停止。调用此函数后立即减速，到达起始速度后停止。	8.2.2.5.3

在单轴执行连续运动过程中，可以调用 d5480_change_speed 实时改变速度。注意：S 形加速连续运动中对运行速度的改变最好在加速已经完成的恒速段进行。图 4-12 和图 4-13 为梯形和 S 形加速下连续运动中变速和减速停止过程的速度变化曲线。

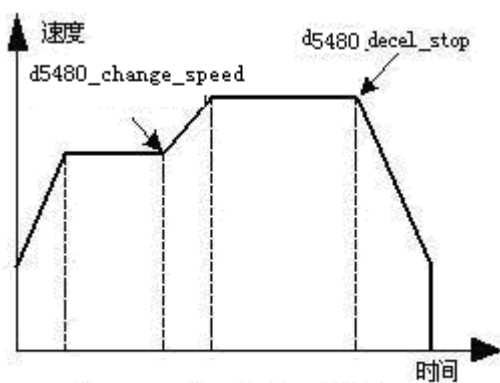


图4-12 梯形运动中变速

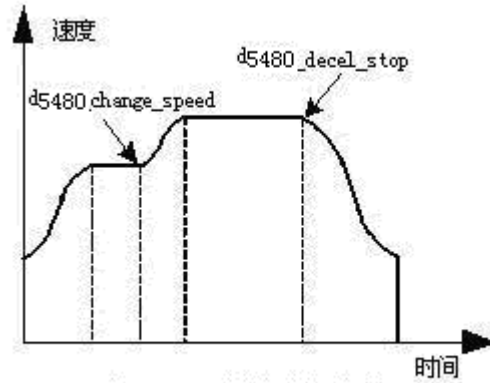


图4-13 S形运动中变速

例程：以 S 形速度曲线加速的连续运动及变速、停止控制

.....

d5480_set_profile(0,500,1000,10000,10000); //设置梯形曲线速度，加、减速时间

d5480_vmove(0,1);

//0 号轴连续运动，方向为正

if(“改变速度条件”)

//如果改变速度条件满足，则执行改变速度命令

{

Curr_Vel= 1200;

//设置新的速度

d5480_change_speed(0,Curr_Vel);

//执行改变速度指令

}

if(“停止条件”)

//如果运动停止条件满足，则执行减速停止命令


```
d5480_decel_stop(0,0.1);           //减速停止，减速时间为 0.1 秒
.....
```

4.2.5.1.4 加减速过程的距离（脉冲数）计算

对于梯形速度曲线运动，加减速段的运动距离（脉冲数）可以按以下公式计算：

$$D_{acc} = (1/2) \times (Max_Vel + Min_Vel) \times T_{acc}$$

$$D_{dec} = (1/2) \times (Max_Vel + Min_Vel) \times T_{dec}$$

其中：Dacc，Ddec 分别为加速段距离和减速段距离；

Min_Vel，Max_Vel 为起始速度和运行速度；

Tacc，Tdec 为加速时间和减速时间。

以上公式也完全适合于 S 曲线的情況。

4.2.5.2 多轴运动控制

雷泰（Leadtech）DMC-5480 运动控制卡单张卡可以同时控制 4 个轴以多种方式同时运动，平时常有的有：多轴联动，直线插补，圆弧插补，连续插补。

4.2.5.2.1 多轴联动

几个轴同时运动，一般称为多轴联动。

DMC-5480 控制卡可以控制多个电机同时执行 d5480_pmove 这类单轴运动函数。所谓同时执行，是在程序中顺序调用 d5480_pmove 等函数，因为程序执行速度很快，在瞬间几个电机都开始运动，给人的感觉就是同时开始运动。

多轴联动在各轴速度设置不当时，各轴停止时间不同、在起点与终点之间运动的轨迹也不是直线。如图 4-14 所示。

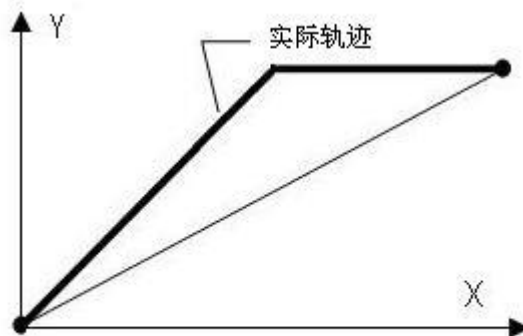


图4-14 二轴联动示意图

4.2.5.2.2 直线插补运动

插补运动与多轴联动则不同：插补运动不但能保证起点、终点位置准确外，X 轴和 Y 轴的脉冲是按照直线斜率成比例发出的，所以在插补运动过程中的每一个时刻，其运动轨迹与理论曲线的误差总是小于一个脉冲当量，如图 4-15 所示。

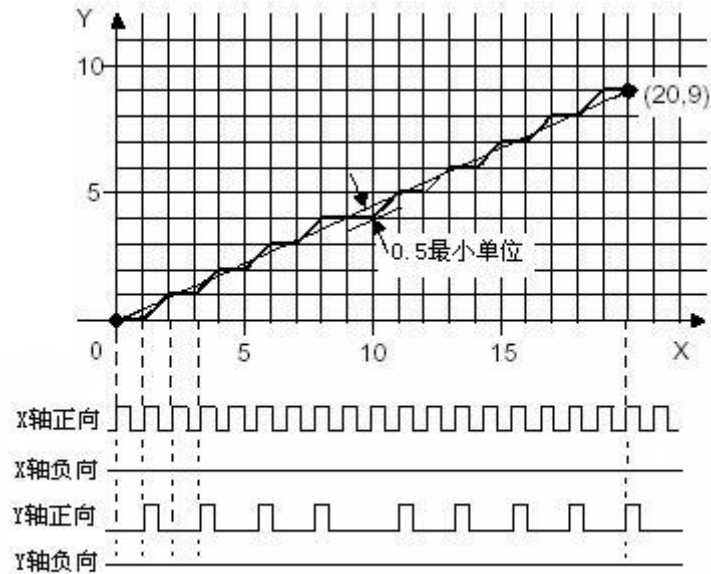


图4-15 直线插补示意图

DMC-5480 卡可以进行任意 2 轴、3 轴和 4 轴直线插补，插补工作由控制卡上硬件执行，用户只需将插补运动的速度、加速度、终点位置等参数写入相关函数，而无需介入插补过程中的计算工作。

二轴直线插补：

如图 4-16 所示，2 轴直线插补从 P0 点运动至 P1 点，X、Y 轴同时启动，并同时到达终点；X、Y 轴的运动速度之比为 $\Delta X : \Delta Y$ ，二轴合成的矢量速度为：

$$\frac{\Delta P}{\Delta t} = \sqrt{\left(\frac{\Delta X}{\Delta t}\right)^2 + \left(\frac{\Delta Y}{\Delta t}\right)^2}$$

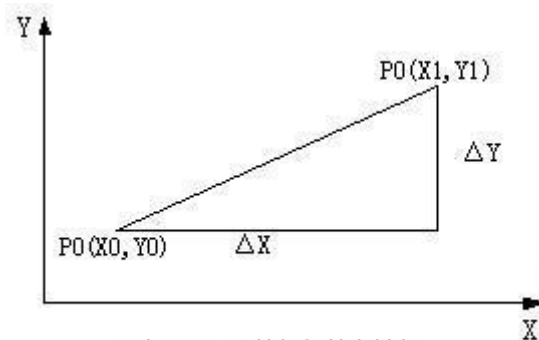


图4-16 两轴直线插补

三轴直线插补：

如图 4-17 所示，XYZ 3 轴直线插补从 P0 点运动至 P1 点。插补过程中 3 轴的速度比为 $\Delta X : \Delta Y : \Delta Z$ ，三轴合成的矢量速度为：

$$\frac{\Delta P}{\Delta t} = \sqrt{\left(\frac{\Delta X}{\Delta t}\right)^2 + \left(\frac{\Delta Y}{\Delta t}\right)^2 + \left(\frac{\Delta Z}{\Delta t}\right)^2}$$

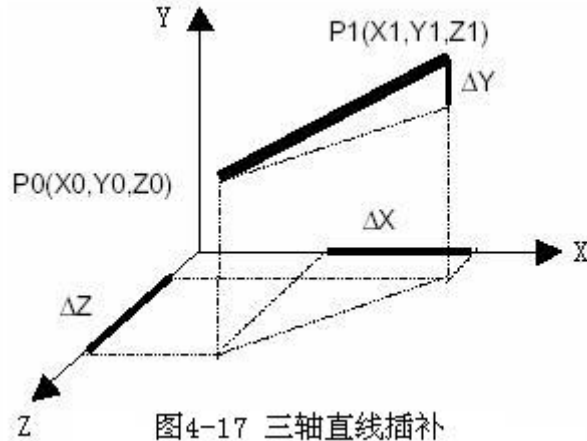


图4-17 三轴直线插补

四轴直线插补：

4 轴插补可以理解为在 4 维空间里的直线插补。一般情况是 3 个轴进行直线插补，另一个旋转轴也按照一定的比例关系和这条空间直线一起运动。其合成矢量速度为：

$$\frac{\Delta P}{\Delta t} = \sqrt{\left(\frac{\Delta X}{\Delta t}\right)^2 + \left(\frac{\Delta Y}{\Delta t}\right)^2 + \left(\frac{\Delta Z}{\Delta t}\right)^2 + \left(\frac{\Delta U}{\Delta t}\right)^2}$$

调用 2 轴直线插补函数时，调用者需提供矢量速度，包括其起始矢量速度 Min_Vel 和工作矢量速度 Max_Vel，梯形和 S 形速度曲线参数。

直线插补运动相关函数如表 4-13 所示：

表 4-13 直线插补运动相关函数说明

	名称	功能	参考
1	d5480_t_line2	让指定的两轴作对称的梯形加减速插补运动。	8.2.2.6.1
2	d5480_t_line3	让指定的三轴作对称的梯形加减速插补运动。	8.2.2.6.1
3	d5480_t_line4	指定四轴以对称的梯形速度曲线做插补运动。	8.2.2.6.1
4	d5480_set_vector_profile	设定插补矢量运动曲线的起始速度、运行速度、加速时间、减速时间	8.2.2.6.1

例程：XY 轴直线插补

```
short AxisArray[2];
AxisArray[0]=0;      //定义插补 0 轴为 X 轴
AxisArray[1]=1;      //定义插补 1 轴为 Y 轴
d5480_set_vector_profile(1000,5000,50000,25000);
d5480_t_line2(AxisArray[0],30000,AxisArray[1],40000,0);
```

该例程使 X，Y 轴进行相对模式直线插补运动，其相关参数为：

Δ X=30000 pulse

Δ Y=40000 pulse

起始矢量速度=1000pps （0 轴,1 轴分速度为 600， 800pps）

工作矢量速度=5000pps （0 轴,1 轴分速度为 3000， 4000pps）

梯形加速时间=0.1s

梯形减速时间=0.2s

4.2.5.2.3 圆弧插补运动

DMC-5480 卡的任意两轴之间可以进行圆弧插补，圆弧插补分为相对位置圆弧插补和绝对位置圆弧插补，运动的方向分为顺时针（CW）和逆时针（CCW），如图 4-18 所示。

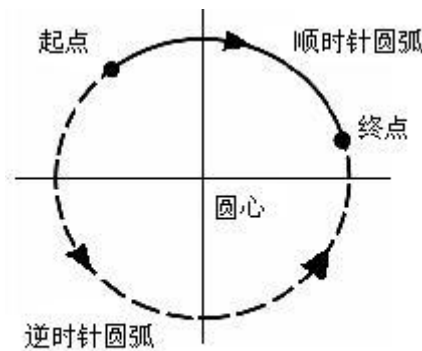


图4-18 两轴圆弧插补

相关函数如表 4-14 所示：

表 4-14 圆弧插补相关函数说明

	名称	功能	参考
1	d5480_arc_move	让指定的二轴作绝对位置圆弧插补运动。	8.2.2.6.2
2	d5480_rel_arc_move	让指定的二轴作相对位置圆弧插补运动。	8.2.2.6.2

例程：XY 轴圆弧插补

```
WORD AxisArray[2]={0, 1};
long target_pos[2]={5000, 0};
long cen_pos[2]={5000,-5000};
d5480_set_vector_profile(1000,3000,30000,15000);
d5480_arc_move (AxisArray, target_pos, cen_pos , 0);
// XY 轴进行顺时针方向绝对圆弧插补运动，终点（5000, 0），圆心（5000, -5000）
```

4.2.5.2.4 连续插补运动

DMC-5480 卡允许多轴电机进行连续多段插补运动，而且通过运动速度设置函数的简单设置就可以实现多线段恒定合成速度连续插补，线段之间没有加/减速过程，连续插补的效果可以用图 4-19 来表示：

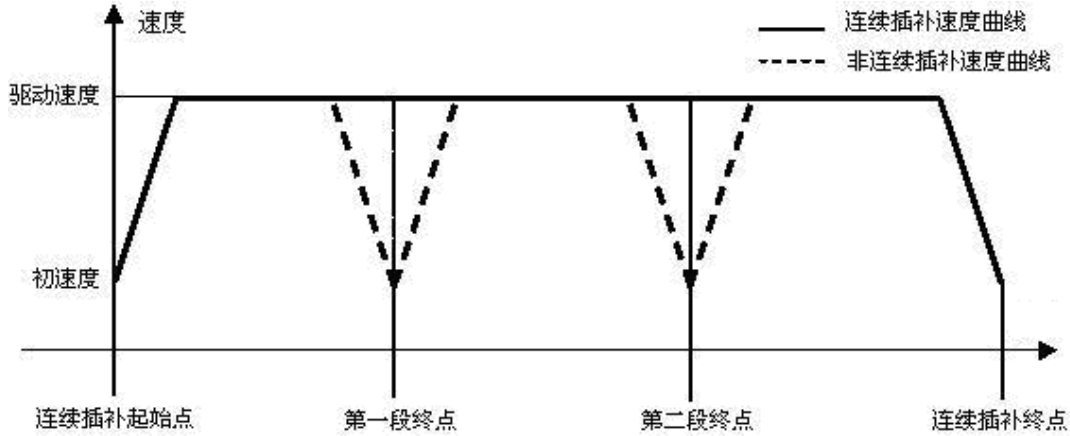


图4-19 连续插补运动

在 DMC-5480 控制卡上具有 512 级硬件缓冲功能，能预先存储 512 步运动的数据。控制卡当前运动的数据存储在工作寄存器 R (register) 中，下一个运动的数据存储在预置缓冲区 PR (pre-register) 中，如图 4-20 所示。当前运动完成，PR 中的数据自动移至 R 中开始执行，此时寄存器 PR 变为空。PC 机查询到 PR 为空后，即可补充下一运动的数据。如此不断循环，直至完成所有运动。

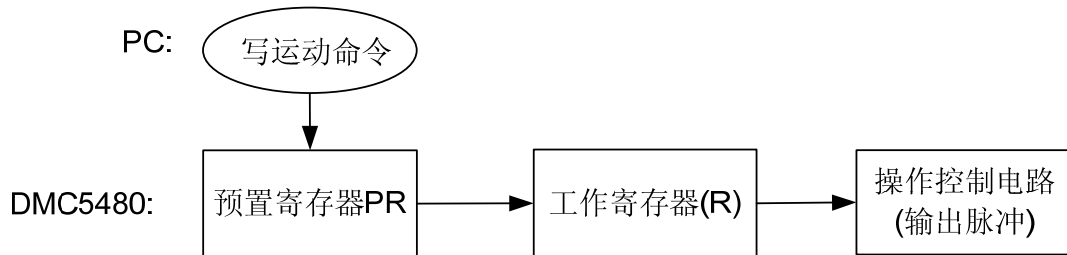


图4-20 DMC-5480 多级缓冲寄存器连续插补的工作原理

连续插补注意事项:

1. 预置缓冲区不为空时，用户不能写入新的运动命令，否则将导致错误。
2. 如果在插补运动过程中，出现触发限位而停止，那么后续写入的资料和命令都是无效的。

例程：连续插补

描述：假定在一工作平面上，刀具在系统 XY 轴物理原点的正向位置(X 轴对应 Axis[0], Y 轴对应 Axis[1])，现刀具先回到系统的物理原点，再执行插补运动 C1，然后执行圆弧插补运动 C2，接着执行插补运动 C3，最后执行插补运动 C4 回到 P1 点，如图 4-22 所示。

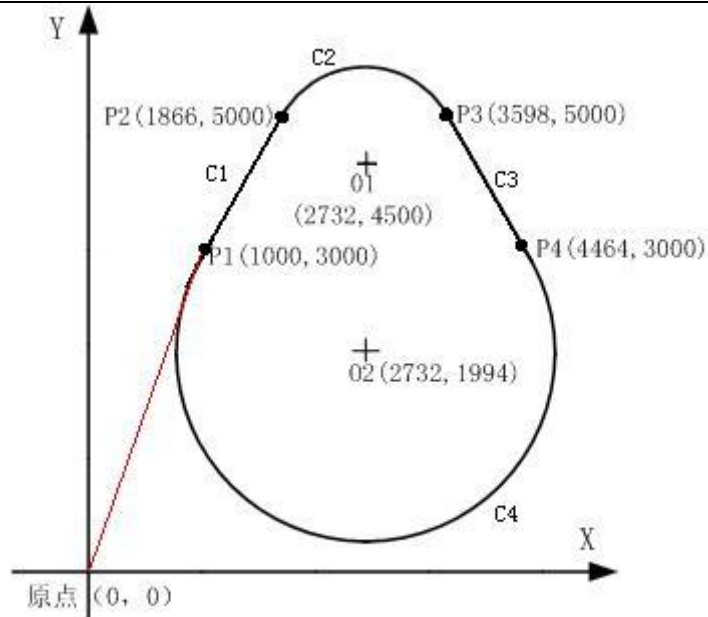


图4-22 DMC-2410B两轴连续插补示例

程序代码:

```
WORD Axis[2];                      //定义运动轴
Axis[0] = 0; Axis[1] = 1;
long Pos1[2] = { 1000, 3000};
long Pos2[2] = { 1866, 5000};
long Pos3[2] = { 3598, 5000};
long Pos4[2] = { 4464, 3000};
long Cen1[2] = { 2732, 4500};      //定义 C2 圆弧插补的圆心位置 O1
long Cen2[2] = { 2732, 1994};     //定义 C2 圆弧插补的圆心位置 O2
d5480_conti_set_mode(0, 0, 400, 20, 20000);      //设置连续插补模式
d5480_set_vector_profile (0, 0, 5000,20000, 20000); //设置插补速度、加速
度, 减速度
d5480_conti_open_list(0);          //打开连续缓存区
d5480_conti_start_list(0);         //开始连续插补
d5480_conti_lines (2, Axis, Pos1, 1);      //到达 P1 点
while (d5480_conti_check_remain_space(0) = 0)
{
    AfxGetApp()->PumpMessage();          //让出控制权
}
d5480_conti_lines (2, Axis, Pos2, 1);      //执行行程 C1,到达 P2 点
while (d5480_conti_check_remain_space(0) = 0)
{
}
```

```

d5480_conti_arc (Axis, Pos3, Cen1, 0, 1);           //执行行程 C2,到达 P3 点
while (d5480_conti_check_remain_space(0) = 0)
{
    AfxGetApp()->PumpMessage();                     //让出控制权
}
d5480_conti_lines (2, Axis, Pos4, 1);              //执行行程 C3,到达 P4 点
while (d5480_conti_check_remain_space(0) = 0)
{
    AfxGetApp()->PumpMessage();                     //让出控制权
}
d5480_conti_arc (Axis, Pos1, Cen2, 0, 1);          //执行行程 C4,到达 P1
点
d5480_conti_close_list(0);// 关闭连续缓存区

```



注意：缓冲寄存器不为空时，不得写入新的运动命令，否则将导致错误。

4.2.6 伺服驱动器专用

雷泰（Leadtech）DMC-5480 运动控制卡控制伺服驱动器时，软件系统分别为 4 轴提供了伺服驱动专用的 5 个信号（SVON，RDY，INP，ALM，ERC）的相应的功能函数，其具体功能和相应的函数如下：

4.2.6.1 SEVON：伺服驱动器使能

雷泰（Leadtech）DMC-5480 运动控制卡为每轴均提供了伺服驱动器使能（SEVON）信号输出接口及操作函数。当该信号为无效状态时，伺服驱动器不使能，电机处于自由状态；当该信号有效时，伺服驱动器使能，电机将锁紧。与之相关函数如下表 4-16 所示：

表 4-16 SEVON 信号相关函数说明

	名称	功能	参考
1	d5480_write_SEVON_PIN	输出对指定轴的伺服使能端子的控制。	8.2.2.7
2	d5480_read_SEVON_PIN	读取指定轴的伺服使能端子的电平状态。	8.2.2.7

例程：输出与读取指定轴伺服控制

.....

```

d5480_write_SEVON_PIN(0, 1);    //输出 0 轴伺服使能端子高电平
status=d5480_read_SEVON_PIN(0); //读取 0 轴伺服使能端子的状态

```

.....

4.2.6.2 RDY：伺服准备好

雷泰（Leadtech）DMC-5480 运动控制卡为每轴均提供了伺服准备好（RDY）信号输入接口及操作函数。当伺服驱动器已经准备好时，伺服驱动器自动将该信号置为有效。此时，控制卡可以向伺服驱动器发出运动命令；若 RDY 信号无效则表示伺服驱动器还未准备好，这时控制卡发出脉冲信号，伺服驱动器也不会按该命令运动。相关函数如下表 4-17 所示。

表 4-17 RDY 信号相关函数说明

	名称	功能	参考
1	d5480_read_RDY_PIN	读取指定运动轴的“伺服准备好”端子的电平状态。	8.2.2.7
2	d5480_read_inbit	读取指定控制卡的某一位输入口的电平状态。	8.2.2.9

例程：读取指定轴伺服是否准备好

.....

```
status= d5480_read_RDY_PIN(0); //读取 0 轴 RDY 信号的状态
```

.....

4.2.6.3 INP：伺服定位完成

雷泰（Leadtech）DMC-5480 运动控制卡为每轴均提供了伺服定位完成（INP）信号输入接口及相应的操作函数。当伺服电机定位完成时，伺服驱动器自动将该信号置为有效。该信号可以通过软件对该信号的有效电平进行设置，同时还可以禁止和使能控制卡对该信号的响应。如果设成禁止（Disable），则控制卡将在定长运动脉冲发完后，不管伺服定位是否完成，都立刻翻转卡上的运动完成标识；若果设置成使能（Enable），那么控制卡在发完脉冲之后，还要等到 INP 信号变为有效之后，才翻转运动完成标识。相关函数如表 4-18 所示：

表 4-18 INP 信号相关函数说明

	名称	功能	参考
1	d5480_config_INP_PIN	设置允许/禁止 INP 信号及其有效的逻辑电平。	8.2.2.7
2	d5480_get_rsts	读取指定轴的外部信号状态。	8.2.2.7

例程：INP 信号设置

.....

```
d5480_config_INP_PIN(0, 1, 1); //设定 0 轴 INP 信号高电平有效
```

.....

4.2.6.4 ALM：伺服报警

雷泰（Leadtech）DMC-5480 运动控制卡为每轴均提供了伺服驱动器报警（ALM）信号输入接口及相应的操作函数。当伺服驱动器发生错误时将自动将该信号置为有效，用来报告伺服驱动器或电机出现了错误。该信号可以通过软件对其有效电平进行设置，同时还可以设置该信号有效时制动模式。若设置模式为

立即停止，则 ALM 信号有效时，DMC-5480 控制卡接将立即停止脉冲输出，该过程是一个硬件处理过程。与之相关函数如表 4-19 所示：

表 4-19 INP 信号相关函数说明

	名称	功能	参考
1	d5480_config_ALM_PIN	设置 ALM 的逻辑电平及其工作方式。	8.2.2.7
2	d5480_axis_io_status	读取指定轴有关运动信号的状态，包含指定轴的专用 I/O 状态。	8.2.2.7

例程：ALM 信号设置

.....

```
d5480_config_ALM_PIN(0, 1, 1);    //设定 0 轴 ALM 信号有效，且为减速停止制动方式
```

.....

4.2.6.5 ERC：误差清除信号

雷泰（Leadtech）DMC-5480 运动控制卡为每轴均提供了伺服驱动器位置误差计数器清零（ERC）信号输出接口及操作函数。当 ERC 信号有效时，伺服电机将停止运动。该信号可以通过软件来使能或禁止，同时可以对其有效电平以及输出方式进行设置。具体相关函数如表 4-20 所示：

表 4-20 INP 信号相关函数说明

	名称	功能	参考
1	d5480_config_ERC_PIN	设置允许/禁止 ERC 信号及其有效电平和输出方式。	8.2.2.7
2	d5480_write_ERC_PIN	控制指定轴“误差清除”端子信号的输出	8.2.2.7

例程：ERC 信号设置

.....

```
d5480_config_ERC_PIN(0, 0, 1, 0, 1);    //设定 0 轴 ERC 信号高电平有效，且为不自动输出模式，有效输出宽度为 12us，关断时间也为 12us
```

.....

4.2.7 手轮运动控制

雷泰（Leadtech）DMC-5480 运动控制卡为 4 轴分别提供了手轮输入接口和相应的功能函数。具体相关的函数如表 4-21 所示：

表 4-21 手轮控制相关函数说明

	名称	功能	参考
1	d5480_set_handwheel_inmode	设置输入手轮脉冲信号的计数方式。	8.2.2.8
2	d5480_handwheel_move	启动指定轴的手轮脉冲运动。	8.2.2.8

例程：手轮控制

```
.....  
d5480_set_handwheel_inmode(0, 0, 0); //设定 0 轴脉冲方式为 AB 相位信号,  
1 倍计数, 且为默认的 PA/PB 输入计数方向  
d5480_handwheel_move(0,vh); //启动 0 轴手轮脉冲运动  
.....
```

4.2.8 通用I/O控制

雷泰（Leadtech）DMC-5480 运动控制卡硬件上为用户提供了通用的数字输入输出接口，最多 32 路输入和最多 28 路输出。用户可以使用这些数字 I/O 口用于输入开关信号、传感器信号等信号，或者输出继电器、电磁阀等输出设备的控制信号。相关的函数如表 4-22 所示：

表 4-22 通用 IO 相关函数说明

	名称	功能	参考
1	d5480_read_inbit()	读取指定控制卡的某一位输入口的电平状态	8.2.2.9
2	d5480_write_outbit()	启动指定轴的手轮脉冲运动。	8.2.2.9

例程：通用输入输出

```
If(d5480_read_inbit(input1)==0) //读取 INPUT1 口的状态, 判断按键是否按下  
{  
d5480_write_outbit(out1, 0); //如果按键按下, OUT1 口输出为 0, LED 发光  
}  
else  
{  
d5480_write_outbit(out1, 1); //如果按键未按, OUT1 口输出为 1, LED 不亮  
} .....
```

4.3 演示程序

雷泰（Leadtech）DMC-5480 运动控制卡为了更方便用户利用 VC 或 VB 等编程工具对其进行开发，针对典型地运动，例如单轴运动、回零动作、插补运动、编码器读取、数据锁存等等，提供了示例源代码，用户可以直接将软件 CD 中相应目录中的代码直接拷贝到您的程序工程中使用。

演示软件的设计，大大简化了用户的调试过程。将 DMC-5480 的软件 CD 盘插入计算机光驱，在相应的目录下，例如“演示界面”，将其全部拷贝到计算机硬盘的任意指定位置后，运行 Motion5480.exe，即可对控制卡的各项主要功能进行检测、学习、还可以借此软件对您的整个自动化系统进行初步的运行，详细资料参考：[6.演示软件及应用](#)。

4.4 例子程序

雷泰（Leadtech）DMC-5480 运动控制卡为了方便用户利用 VC 或 VB 等进行应用开发，特针对典型的功能，如：单轴运动、回原点、插补运动、编码器读取、数据锁存等等，提供了示例源代码。

用户可以直接将配套的 CD 中相应目录中的代码直接拷贝到您的程序工程中使用。各个例程源代码所能实现的功能作如下：

4.4.1 单轴运动例程

此例程为四个轴的单轴定长运动，为每个轴增加设置了运动速度和加速度，设有启动，停止，减速停，急停，位置清零和退出功能按钮。这个例程用到了控制卡的初始化/关闭函数，读取各运动轴当前速度/位置函数，设置轴运动速度函数/定长运动函数，轴的运动状态函数，减速停函数/急停函数。通过这个例程的学习可以了解我司控制卡基本函数的调用，详细代码请参考光盘例程“例 1_单轴运动”。

单轴运动例程主界面如图 4-23 所示：

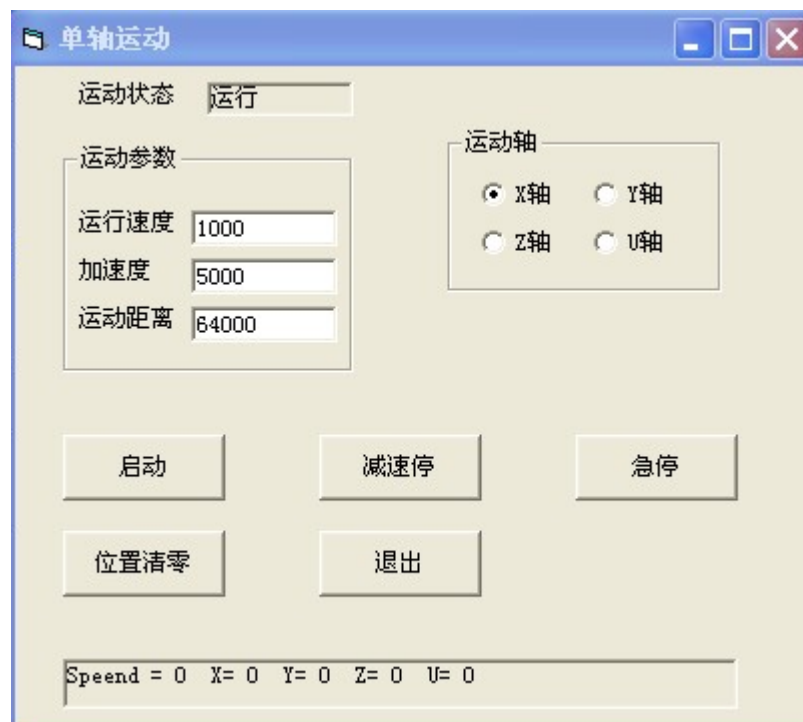


图4-23 单轴运动例程主界面

4.4.2 回原点例程

此例程为四个轴的回零点程序，为每个轴设置了回零运动速度，加速度和运动距离。设有启动，停止，减速停，急停，位置清零和退出功能按钮。设有回零点正负方向选择，回零点低/高速选择，回零点信号 HOME 和 HOME+EZ 两种方

式选择。在程序界面底层可以显示每个轴运动的实时速度，并可以实时检测轴的运动状态。详细代码请参考盘例程“例 2_回原点”。

回零点运动例程主界面如图 4-24 所示：



图4-24 回零点例程主界面

4.4.3 直线插补例程

此例程为任意两轴至四轴的直线插补运动例程。程序为每段插补线段设有运行速度，加/减速度，每个轴设有插补运动距离，提供两到四轴插补轴的任意方式选择。设有“执行”，“减速停止”，“急停”，“位置清零”和“退出”按钮等功能，程序底层可显示插补运动时每个轴的实时插补速度。并且可以在线检测轴的运动状态。详细代码请参考盘例程“例 3_直线插补”。

直线插补运动例程主界面如图 4-25 所示：



图4-25 直线插补运动例程主界面

4.4.4 两轴圆弧插补例程

此例程为第 0 轴和第 1 轴两轴的圆弧插补运动例程。其它任意两轴的圆弧插补与此例程的代码类似。圆弧插补的轴号，矢量运动速度和插补距离在代码里面已设置好。程序设有“启动”，“急停”，“位置清零”，“退出”按钮等功能。详细代码请参考光盘例程“例 4_两轴圆弧插补”。

两轴圆弧插补例程主界面如图 4-26 所示：



图4-26 两轴圆弧插补例程主界面

4.4.5 连续插补例程

此例程为第 0 轴和第 1 轴的连续插补运动例程。其它任意两轴圆弧插补和两到四轴的直线插补与此例程的代码类似。连续插补的轴号，运动速度，插补距离和运动轨迹在程序里已设置好，详细运动轨迹如下图 4-27。程序设有“插补运动”，“急停”，“位置清零”，“退出”等按钮功能，在线实时显示第 0 轴和第 1 轴的运动的位置和速度。程序还将连续插补剩余缓存数和单个轴的缓冲区的使用状态显示出来。详细代码相关轨迹数据请参考光盘例程“例 5_连续插补”。连续插补例程主界面如下图 4-28 所示：

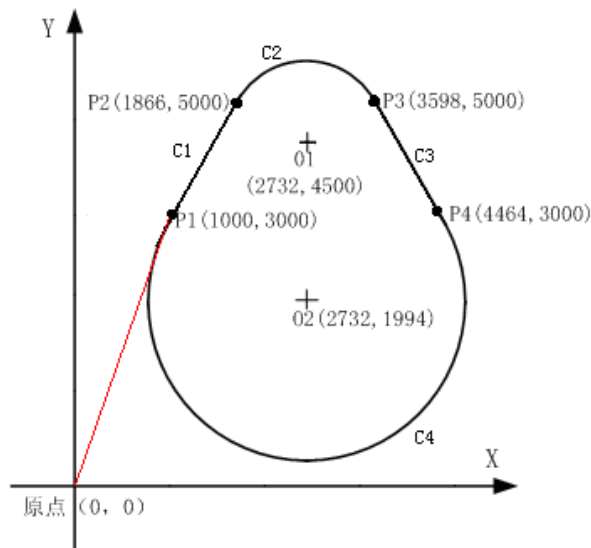


图4-27 两轴连续插补轨迹图

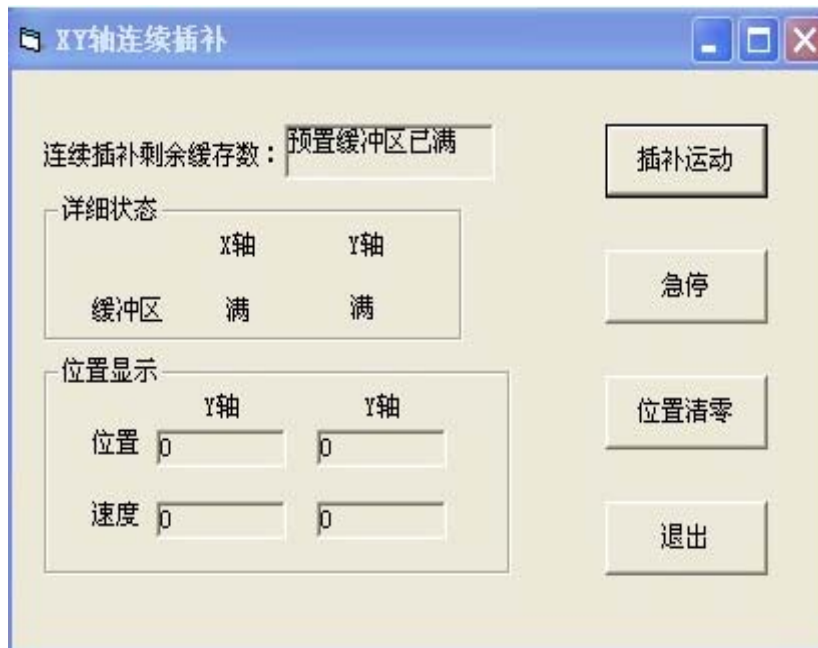


图4-28 两轴圆弧插补例程主界面

4.4.6 手轮运动例程

此例程为四个独立轴的手轮脉冲输入例程。有脉冲输入方式和轴的选择，手轮倍率数值文本框，设有“启动”“停止”“退出”按钮等功能，并且在界面底部可以实时检测当前手轮输入脉冲的位置数。详细代码请参考光盘例程“例 6_手轮运动”。

手轮运动主界面如下图 4-29 所示：



图4-29 手轮运动例程主界面

4.4.7 通用专用输入输出信号例程

此例程为 DMC-5480 控制卡通用数字 I/O 信号和每个轴的专用 I/O 信号的检测例程。程序里有所有通用输入/输出信号的检测图标，可以对每个输出口的输出状态进行操作，并可以检测每个轴的专用信号的输入状态。详细代码请参考光盘例程“例 7_通用专用输入输出”。其主界面如图 4-30 所示：

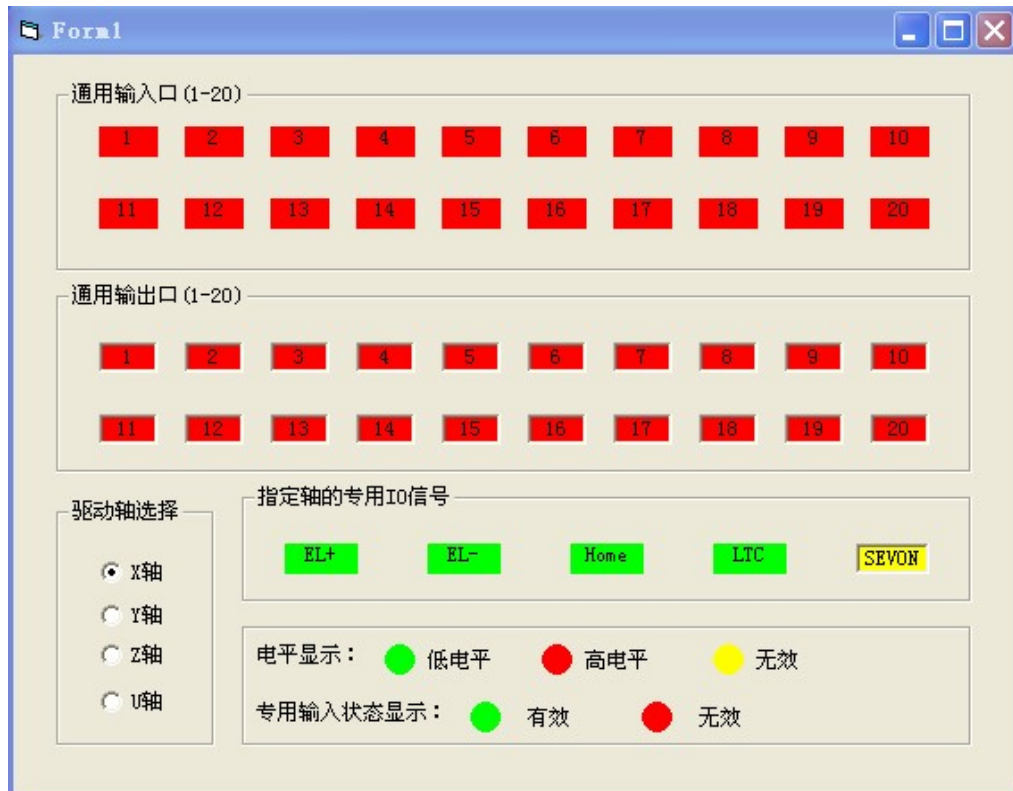


图4-30 通用专用输入输出例程主界面

4.5 多卡运行

雷泰（Leadtech）DMC-5480 运动控制卡支持单卡使用同时也支持多卡同时运行，最大支持 8 块同时工作。这样，一台 PC 机可能可以同时控制多达 $4 \times 8 = 32$ 轴步进/伺服同时工作。同时 DMC-5480 均支持即插即用，用户可不必去关心如何设置卡的基地址和 IRQ 中断值，当系统启动时这些全被 PC 操作系统自动设定。

用户使用 Motion5480 演示软件时卡号的和轴号的对应关系如下：卡号 1 —> 轴号 0—3；卡号 2 —> 轴号 4—7，...卡号 n —> 轴号 $4 \times (n-1)$ — $4n-1$ 。

5 驱动程序安装

雷泰（Leadtech）DMC-5480 运动控制卡的驱动程序遵从 32bit PCI 卡驱动标准，其安装方法类同普通 32bit PCI 卡驱动程序的安装方法。下面以 Windows XP 操作系统下的安装为例：

首先运行批处理文件 regist2k.bat

- 1) 确保已经通读本手册，并参照[3.2 硬件安装](#)将DMC-5480 硬件安装好；
- 2) 启动 PC 机，进入 Windows XP 操作系统；
- 3) 操作系统系统提示发现新硬件，并弹出图 5-1 所示的找到新硬件安装向导；

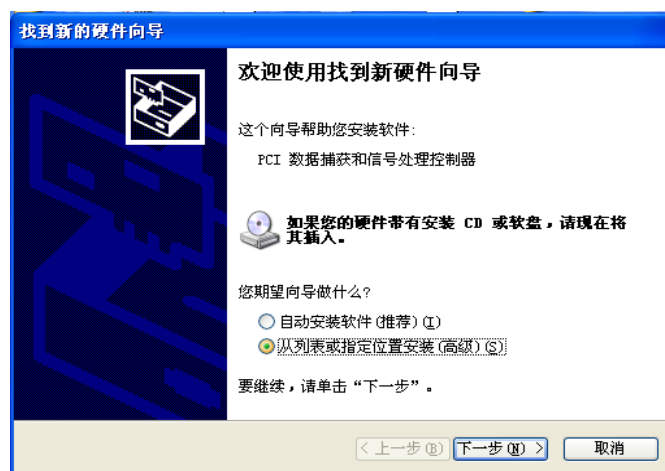


图 5-1 新的硬件安装向导

- 4) 将 DMC-5480 所配光盘放入光驱中，选择“从列表或指定位置安装（高级）（S）”，点击“下一步”；
- 5) 在图 5-2 所示的向导中，点选“在这些位置中搜索最佳驱动程序”并勾选上“在搜索中包括这个位置”，点击浏览；



图 5-2 浏览并选择安装文件夹 INF

- 6) 在弹出的“浏览文件夹”的窗口中，找到光盘中 INF 目录后，点击确定。
- 7) 在图 5-3 中，点击“下一步”继续安装；



图 5-3 开始安装驱动软件

8) 如图 5-4 所示，向导正在安装 Leadtech DMC-5480 运动控制卡的驱动程序；



图 5-4 安装驱动软件进行中

9) 等待安装完成，显示如图 5-5 所示界面，点击完成。



图 5-5 完成 DMC-5480 安装

10) 至此雷泰（Leadtech）DMC-5480 运动控制卡的驱动程序安装完成，可以正常使用了。

6 演示软件及应用

为了方便用户利用 VC 或 VB 等编程工具对 DMC-5480 进行开发, 针对典型地运动, 例如单轴运动、回零动作、插补运动、编码器读取、数据锁存等等, 雷泰公司提供了示例源代码, 用户可以直接将软件 CD 中相应目录中的代码直接拷贝到您的程序工程中使用。

当您将 DMC-5480 控制卡的软硬件正确地安装到 PC 机上后, 将软件 CD 上的演示软件拷贝到计算机上, 就可以直接运行该软件。启动 Motion5480 软件后, 首先进入演示软件主界面如图 6-1 所示。Motion5480 软件提供了参数设置、I/O 检测、计数功能、运动测试这四个主要的操作界面。根据界面的信息, 您可以进行一些基本的控制操作: 比如简单的点位运动、I/O 信号检测等。



图 6-1 演示软件主界面

有关 DMC-5480 卡的一些运动控制基本功能以及硬件接线的说明, 请仔细阅读 DMC-5480 硬件部分章节。

雷泰 (Leadtech) 公司为了用户尽快熟悉 DMC-5480 控制卡的各项功能, 调试、测试系统, 特引进美国技术, 为 DMC-5480 开发了 Motion5480 演示测试软件。

用户利用 Motion5480 软件, 既可以快速的熟悉 DMC-5480 运动控制卡的软硬件功能, 又可以方便快捷地测试电机驱动系统执行各种运动时的性能特点。

Motion5480 的使用过程参考如下:

1) 使用 Motion5480 前, 请确保已经通读本手册, 并参照 [3 硬件配置与安装](#), [5 驱动程序安装](#), 完成硬件配置、硬件安装和驱动程序安装。

- 2) 启动 PC 机，进入 Windows XP 操作系统。
- 3) 将 DMC-5480 所配光盘放入光驱中，找到光盘中的 Motion5480 目录，将其全部复制到硬盘中，弹出光盘。
- 4) 在复制到硬盘中的 Motion5480 目录中找到 Motion5480.exe，运行之，即显示如图 6-1 所示演示软件主界面。在此界面上可以通过点击相应按钮，进入下功能界面。
- 5) 选择所需的卡号，然后进行相应的操作。若只安装了一张 DMC-5480，则卡号为 0。
- 6) 至此，即可使用 Motion5480，对 DMC-5480 的各项功能进行熟悉、测试，以及对系统进行测试、调试。

DMC-5480 卡软件函数库支持最大 8 块控制卡同时工作。因此，一台 PC 机可能可以同时控制多达 32 轴步进/伺服电机同时工作。因为 DMC-5480 卡支持即插即用，用户可不必去关心如何设置卡的基地址和 IRQ 中断值，当系统启动时这些全被系统 BIOS 自动设定。

用户需要使用 Motion5480 测试软件确定控制卡卡号的接口对应关系，例如：往轴 1 发送定长运动，执行运动的电机即可被判定位于卡 1 (0 号卡)。卡号和轴号的对应关系：卡 1 (轴 0—3)；卡 2 (轴 4—7)，...卡 n (轴 $4 \times (n-1)$ ， $4n-1$)。

6.1 参数设置演示

Motion5480 对运动控制卡的参数设置由四个子界面组成，分别是脉冲设置、计数设置、回原点设置、限位设置。

当参数界面窗口打开时，程序会自动加载当前的控制卡参数。

6.1.1 脉冲参数设置

脉冲参数设置包括：脉冲输出类型设置、脉冲有效电平的设置和方向控制逻辑电平设置，如图 6-2 所示。

指令脉冲类型：也即脉冲输出类型，用户可以设置为正脉冲/负脉冲模式 (CW/CCW 模式) 或脉冲/方向模式 (pulse/dir 模式)。该参数根据所用电机驱动器接收脉冲的类型来决定。

脉冲输出有效电平：选择下降沿有效时，脉冲停止时脉冲信号为高电平；选择上升沿有效时，脉冲停止时脉冲信号为低电平。

方向控制逻辑电平：用户可以设置某一电平状态对应为电机的正转方向或反转方向。设置该参数可不改变硬件接线，就能改变电机的运动方向。

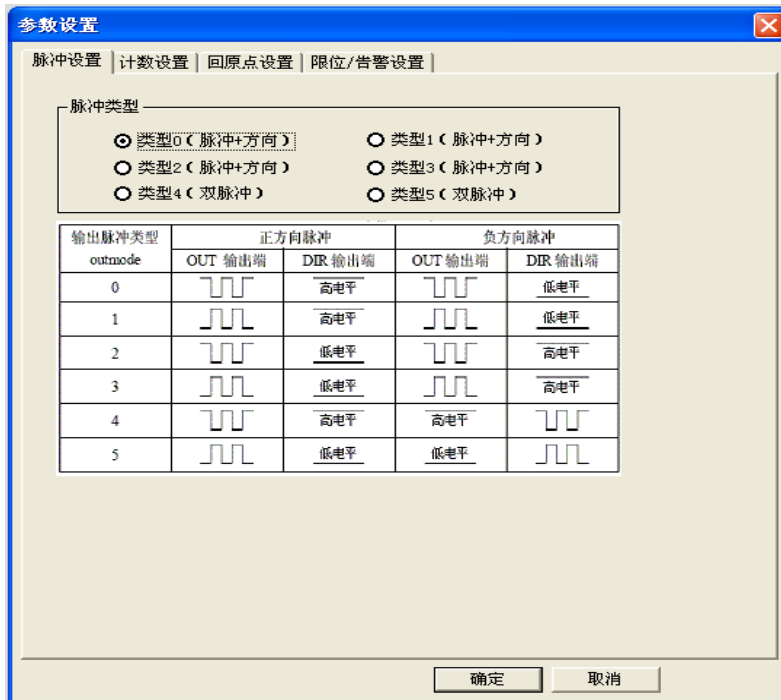


图 6-2 脉冲信号设置子界面

6.1.2 计数参数设置

计数参数设置包括：脉冲输入模式、EZ 信号设置、触发方式，如图 6-3 所示。



图 6-3 计数设置子界面

脉冲输入模式的设置：设置输入到编码器接口的信号类型，分为 AB 相或非 AB 相。

EZ 信号设置：分为 EZ 信号有效电平的设置、EZ 信号是否对编码器的计数

器自动清零的设置。

LTC 信号设置：设置触发信号 LTC 的有效电平为高还是低、单独锁存一个轴的数据还是全部轴的数据。

6.1.3 回原点参数设置

回原点参数设置包括：原点信号有效电平和回原点运行方式，如图 6-4 所示。

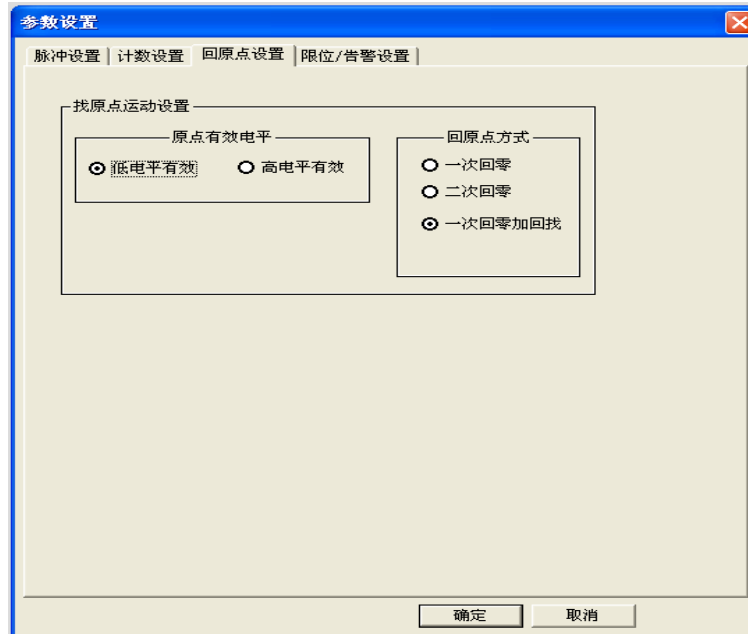


图 6-4 回原点设置子界面

原点信号有效电平：根据原点传感器的实际电路设置其有效电平。

回原点运行方式：参见 d5480_config_home_mode 函数介绍。

6.1.4 限位参数设置

限位参数设置包括硬件限位、软件限位、ALM 配置、反向间隙设置和 EMG 设置，如图 6-5 所示。具体的设置说明，请参见驱动器专用接口信号的设定函数。

硬件限位停止方式：可以设置外部限位开关触发后是立即停止、还是减速后停止。

限位开关有效电平：设置限位开关信号是高电平有效、还是低电平有效。

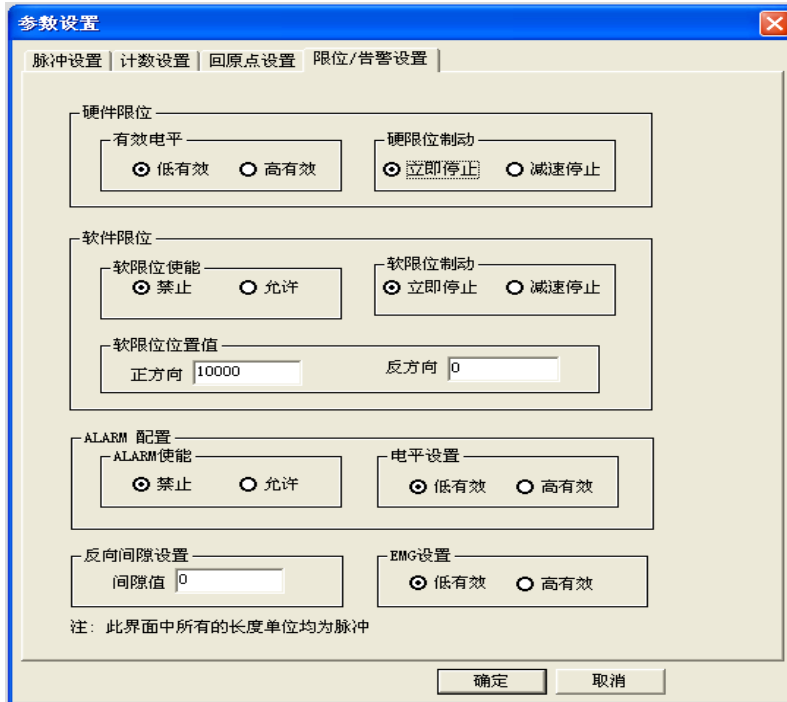


图 6-5 限位设置子界面

6.2 IO操作演示

I/O 信号测试操作界面用于检测各种 I/O 的实时状态，每一个指示灯对应一个 I/O 信号的状态，绿色表示 ON，红色表示 OFF；还可以通过按钮设置每一个通用输出口的电平。I/O 测试界面如图 6-6 所示：

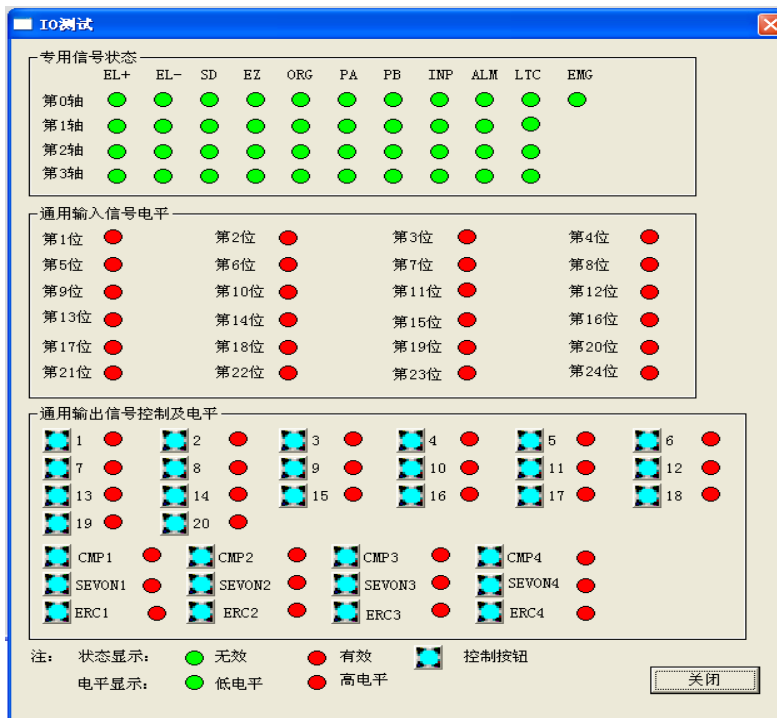
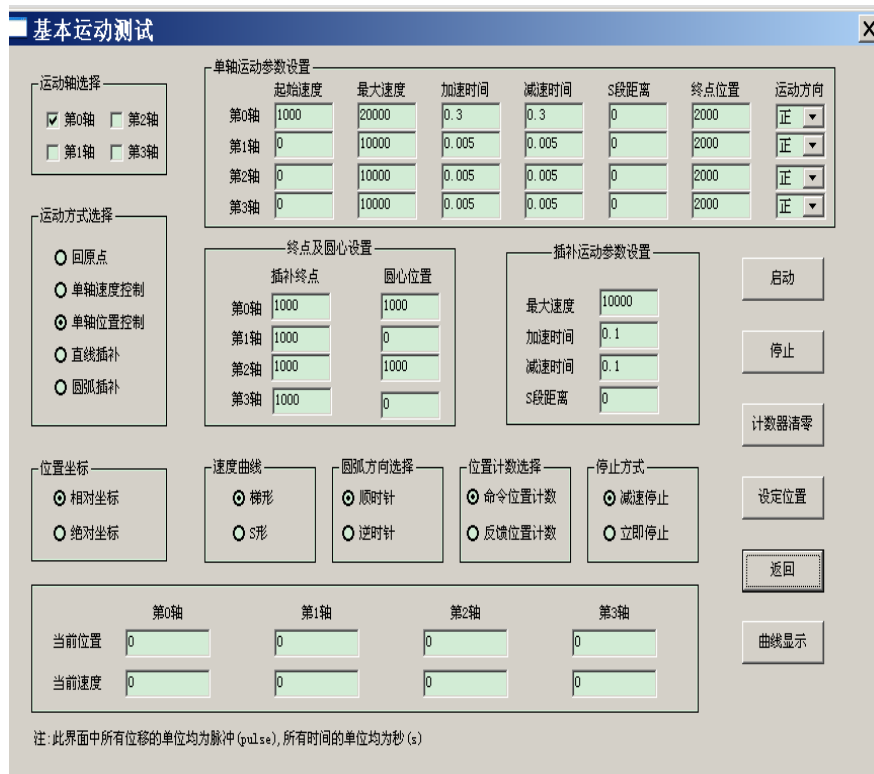


图 6-6 I/O 测试界面

6.3 基本运动演示

Motion5480 基本运动测试界面如图 6-7 所示。



基本运动测试

运动轴选择

☒ 第0轴 ☐ 第2轴
☐ 第1轴 ☐ 第3轴

运动方式选择

☐ 回原点
☐ 单轴速度控制
☒ 单轴位置控制
☐ 直线插补
☐ 圆弧插补

位置坐标

☒ 相对坐标
☐ 绝对坐标

速度曲线

☒ 梯形
☐ S形

圆弧方向选择

☒ 顺时针
☐ 逆时针

位置计数选择

☒ 命令位置计数
☐ 反馈位置计数

停止方式

☒ 减速停止
☐ 立即停止

单轴运动参数设置

	起始速度	最大速度	加速时间	减速时间	S段距离	终点位置	运动方向
第0轴	1000	20000	0.3	0.3	0	2000	正
第1轴	0	10000	0.005	0.005	0	2000	正
第2轴	0	10000	0.005	0.005	0	2000	正
第3轴	0	10000	0.005	0.005	0	2000	正

终点及圆心设置

	插补终点	圆心位置
第0轴	1000	1000
第1轴	1000	0
第2轴	1000	1000
第3轴	1000	0

插补运动参数设置

	最大速度	加速时间	减速时间	S段距离
	10000	0.1	0.1	0

当前位置/速度显示

	第0轴	第1轴	第2轴	第3轴
当前位置	0	0	0	0
当前速度	0	0	0	0

注: 此界面中所有位移的单位均为脉冲 (pulse), 所有时间的单位均为秒 (s)

启动 停止 计数器清零 设定位置 返回 曲线显示

图 6-7 运动操作测试界面

运动轴选择: 选择要测试的轴号, 根据运动类型: 单轴运动、2~4 轴直线插补或 2 轴圆弧插补, 选定相应的轴号。

运动方式选择: 运动模式有 5 种: 回原点运动、单轴速度控制、单轴位置控制、多轴直线插补运动、圆弧插补运动。

单轴速度控制是指: 只控制该轴的速度, 不控制其运动距离, 只有在接到停止信号后, 电机才会停止运动。

单轴位置控制即单轴的点位运动。

多轴直线插补可完成 2~4 轴的直线插补运动。

圆弧插补运动可在任意 2 轴间进行。

单轴运动参数设置: 设置 0~3 号轴在进行回原点、速度控制、位置控制时的运动参数, 包括: 最大速度、加速度、减速度、S 段加/减速时间、终点位置、运行方向。

插补运动参数设置: 插补运动的最大速度、加速度、减速度和 S 段加/减速时间。

终点及圆心设置: 设置 0~3 号轴的插补终点、圆心位置。

停止方式: 选择, 确定运动轴在遇到停止指令时, 是按照设置的运动曲线减速停止还是立即停止脉冲指令。

位置坐标：位移方式选择，确定位置坐标是相对还是绝对。

速度曲线：选择，确定加减速时的运动曲线是梯形还是 S 形。

圆弧方向选择：确定圆弧插补时的运动方向。

位置计数器选择：确定显示 0~3 号轴当前位置的信号是来自指令位置计数器还是反馈位置计数器。

按“曲线显示”键，弹出如图 6-8 所示的“曲线显示”窗口；

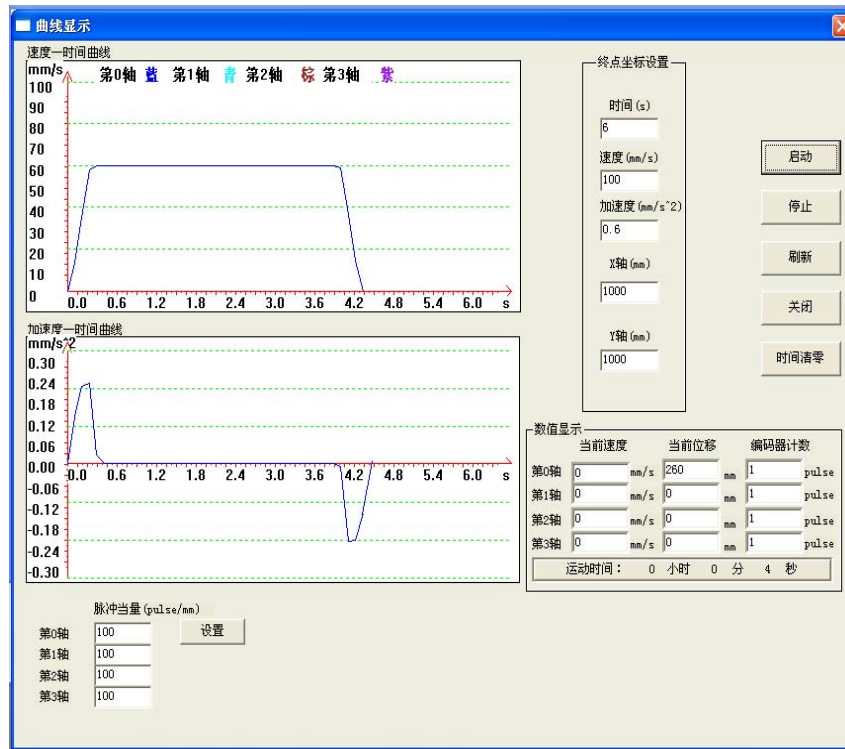


图 6-8 曲线显示界面

用于显示速度，加速度波形。使用者可以首先在各参数（如时间、速度、加速度等）的文本框输入合适的值，然后点击“启动”按钮，窗口上就会绘制出运动轴的“速度-时间”曲线和“加速度-时间”曲线，同时会将运动状态参数值实时显示在“数值显示”参数框中，便于使用者更加直观的分析。

按“启动”键，按照设置的参数和选择的运动方式开始运动。

按“停止”键，停止一切正在运动的轴。

按“计数器清零”键，将所有位置寄存器的数值清零。

点击“刷新”按钮，可以清除窗口中已经生成的曲线。

点击“时间清零”按钮，窗口右下方的“运动时间”显示框的数值将全部归零。

7 用户系统开发

如果您对 C、C++、Visual Basic 等程序语言一点都不了解的话，我们建议您先花几天时间去阅读至少一本该语言的培训教材，并且通过练习掌握该语言的基本技巧，例如如何编写简单的程序，如何创建窗体和调用函数。如果您曾用 C、C++、Visual Basic 等程序语言进行过运动控制软件的开发，并具有丰富的经验，那么可以浏览索引中的函数，并找到所需要的函数描述页码，跳转到“附录中运动函数库”查阅所需要的相关函数信息。

雷泰 (Leadtech) DMC-5480 运动控制卡用户控制软件可以在 Visual Basic 和 Visual C++ 开发环境下开发。

7.1 基于windows平台的应用软件结构

使用雷泰控制卡的机器控制系统构架如图 7-1 所示：

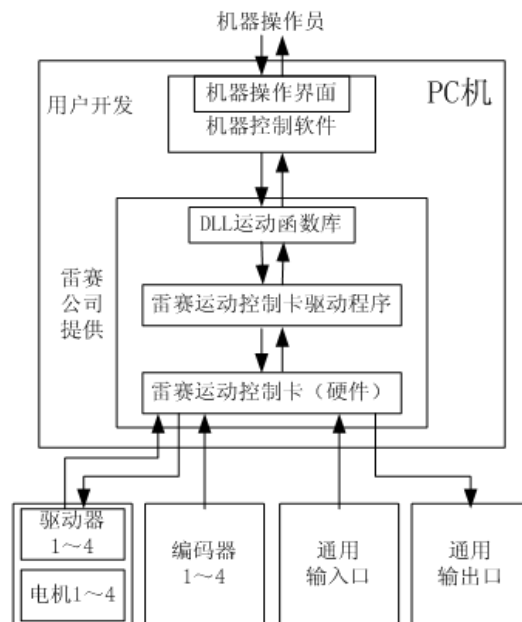


图 7-1 基于雷泰控制卡的机器控制系统构架

从上面的示意图可以看出，控制系统的工作原理可以简单描述为：

1. 操作员的操作信息通过操作接口（包括显示屏和键盘）传递给机器控制软件；
2. 机器控制软件将操作信息转化为运动参数并根据这些参数调用 DLL 库中运动函数；
3. 运动函数调用雷泰运动控制卡驱动程序发出控制指令给控制卡；
4. 雷泰运动控制卡再根据控制指令发出相应的驱动信号（如脉冲、方向信号）给驱动器及电机、读取编码器数据、读写通用输入输出。

用户在开发应用软件（即机器控制软件）的过程中所需要做的就是针对上面所说的第 1 步和第 2 步进行编程。雷泰公司已提供支持 DMC-5480 运动控制卡

的硬件驱动程序和 DLL 运动函数库，包括控制卡初始化函数、单轴及多轴控制函数、输入/输出脉冲模式设置函数等许多函数。这些函数提供了所有与运动控制相关的功能，使用极为方便。用户不需要更多了解硬件电路的细节以及运动和插补的计算细节，就能够使用 C、C++、Visual Basic 等程序语言调用这些函数来快速开发出自己的应用软件。

用户编写的机器控制软件的典型流程如图 7-2 所示：

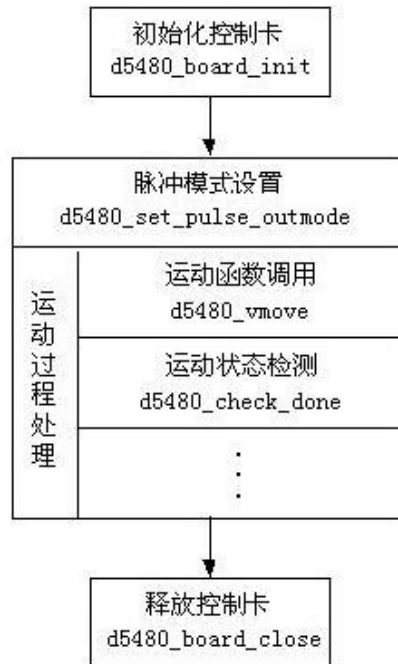


图 7-2 控制软件的典型流程

7.2 Visual Basic环境下编程

Visual Basic6.0 环境下编程，以定长运动为例：

- (1) 在磁盘上新建一个目录，如 E:\test1
- (2) 打开 Visual Basic 6.0，新建一个“标注 EXE”工程，在对话框上添加按钮“启动”（名称修改为 CB_Start）和“停止”（名称修改为 CB_Stop），如图 7-3 所示：

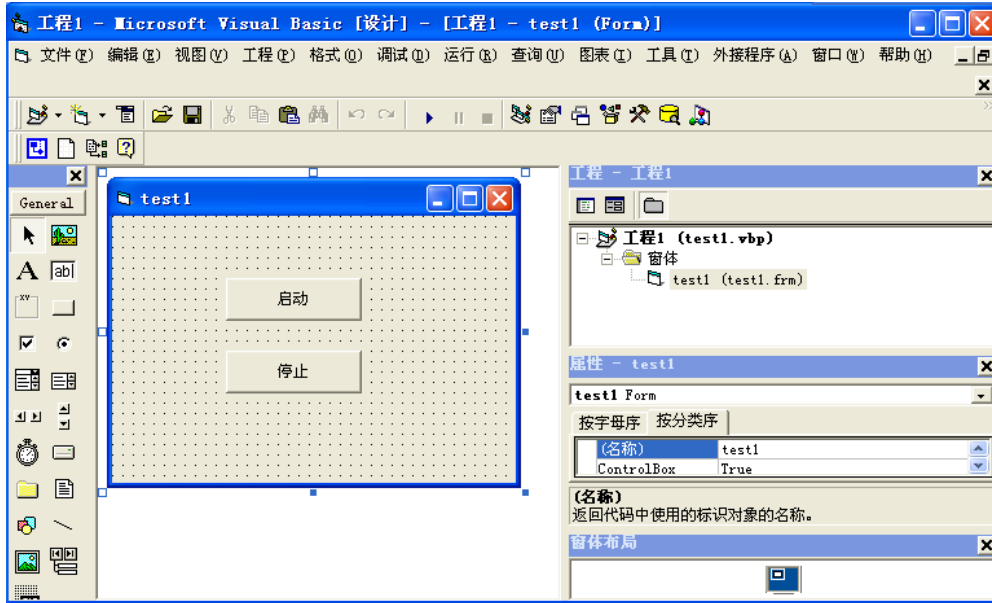


图 7-3 修改对话框（VB）

- (3) 工程保存在 E:\test1 目录下；
- (4) 在资料光盘相应目录下找到 DMC5480.bas 文件，拷贝到 test1 目录下；
- (5) 选择“工程”->“添加模块”->“现存”，找到 test1 目录下的 DMC5480.bas 文件，添加到工程中，如图 7-4 所示：

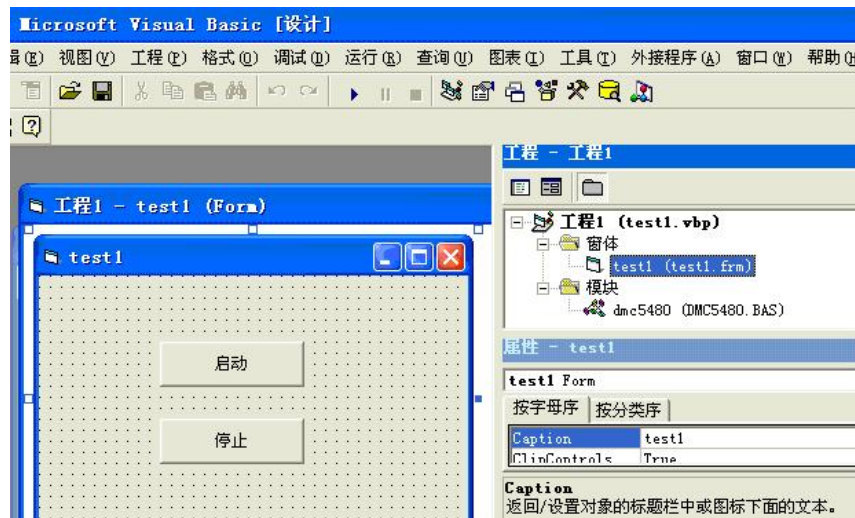


图 7-4 添加头文件

- (6) 如图 7-5 所示，双击窗口控件，在 Form_Load 事件中添加代码
`d5480_board_init` 选择 UnLoad 事件，在 Form_Unload 事件中添加代码
`d5480_board_close` 双击“启动”按钮，在 CB_Start_Click 事件中添加代码
`d5480_set_profile 0,500,5000, 500000,500000`
`d5480_pmove 0,200000,0`
 双击“停止”按钮，在 CB_Stop_Click()事件中添加代码
`d5480_decel_stop 0, 0.01`

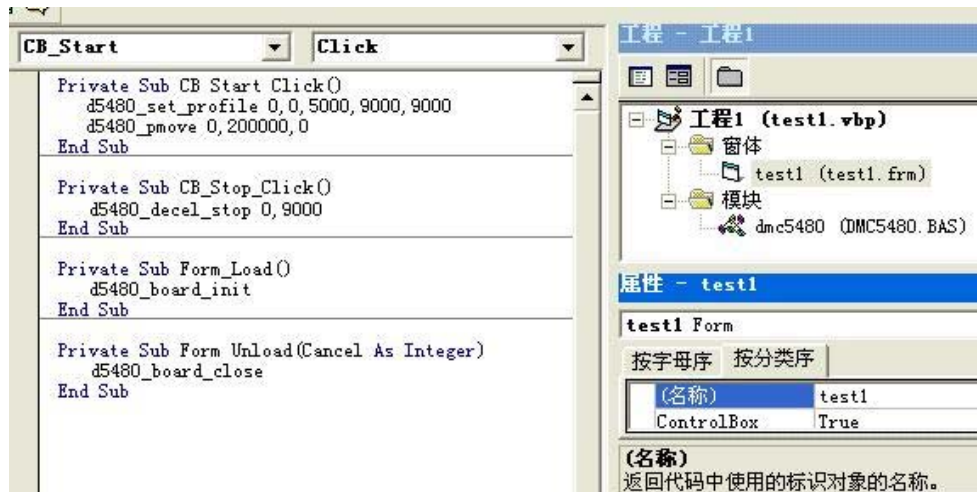


图 7-5 程序中调用运动控制卡库函数（VB）

（7）运行，按下“启动”按钮，第 0 轴就会输出长度为 20000 的脉冲，运动中可以按下“停止”按钮便会减速停止脉冲输出，如图 7-6 所示：



图 7-6 程序运行界面（VB）

注：我们在 DMC-5480 的用户光盘中还提供了以下几个示例供用户参考：

- 示例一，单轴运动。卡的初始化，关闭操作，脉冲模式设置和单轴运动命令。
- 示例二，回原点运动。包括回原点的速度和方向的设置。
- 示例三，直线插补运动。
- 示例四，两轴圆弧插补运动。
- 示例五，连续插补运动。
- 示例六，手轮运动。
- 示例七，通用专用输入输出。

7.3 Visual C++环境下编程

- （1）打开 Visual C++ 6.0；
- （2）新建一个工程；
- （3）选择 MFC APPWizard(exe)；

- (4) 选择工程保存路径，如：E:\；
(5) 输入工程名,如：test1。如图 7-7:

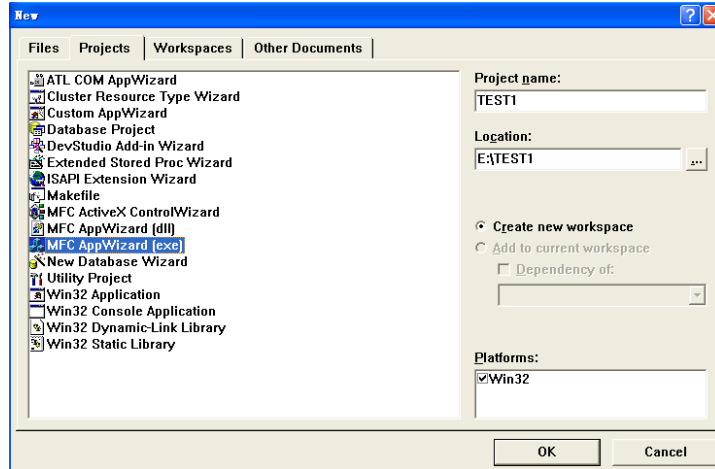


图 7-7 创建新工程

- (6) 在应用程序类型中选择“基于对话框”，按“完成”键，建立工程；
(7) 给对话框进行简单的修改，增加按钮“启动”（命名为 IDC_BUTTON_Start）和“停止”（命名为 IDC_BUTTON_Stop），如图 7-8 所示：

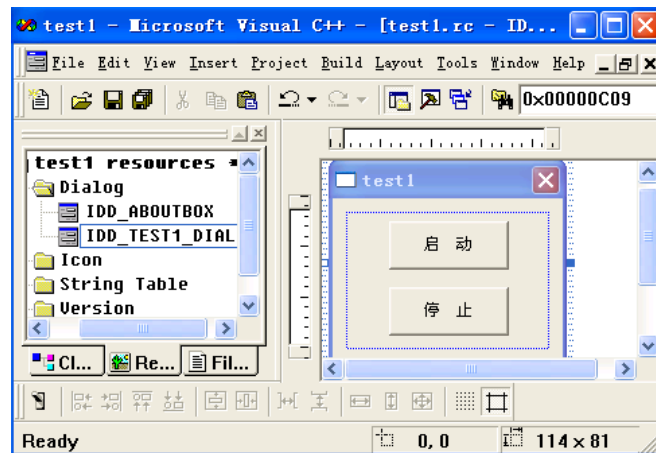


图 7-8 修改对话框

- (8) 在相应的目录下找到 DMC5480.h 和 DMC5480.lib 文件，拷贝到 E:\test1 目录；
(9) 选择“工程”->“添加工程”->“文件”，选中 DMC5480.lib 文件加入到工程中；
(10) 打开 test1.cpp 文件，在程序开始部分添加相应语句：#include “DMC5480.h” 或者 #include “DMC5480.h”，如图 7-9 所示：

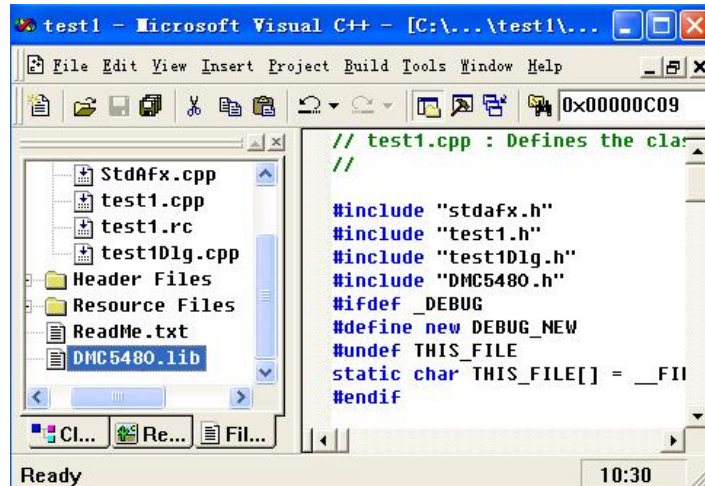


图 7-9 程序增加头文件

(11) 在 CTest1Dlg::OnInitDialog()函数中添加代码: d5480_board_init();如图 7-10:

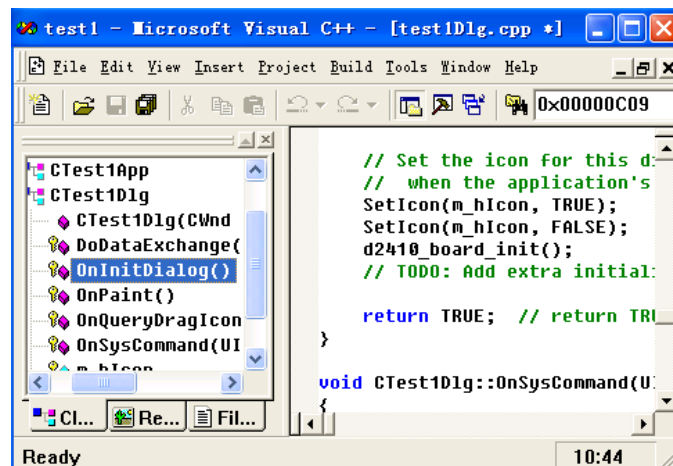


图 7-10 程序增加初始化函数

(12) 在 Ctest1Dlg 中添加一个成员函数 OnCancel,在 OnCancel 函数中添加代码:

```
D5480_board_close();  
CDialog::OnCancel();
```

如图 7-11:

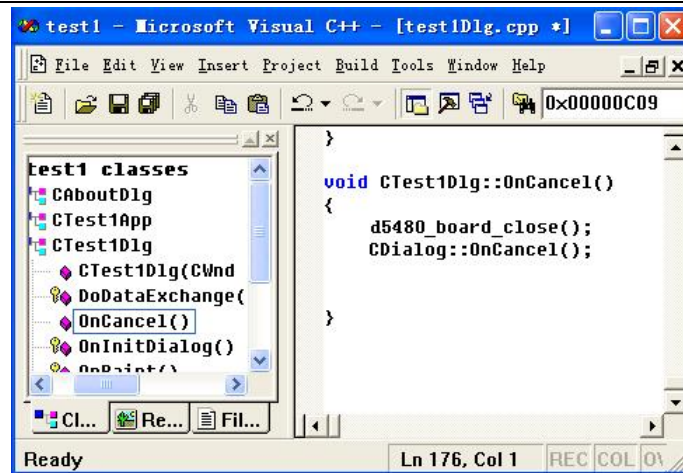


图 7-11 程序增加 OnCancel 函数

(13) 双击“启动”按钮在按钮点击事件中输入代码：

```
d5480_set_profile(0,500,5000, 500000,500000);
```

```
d5480_pmove(0,200000,0);
```

双击“停止”按钮在按钮点击事件中输入代码：

```
d5480_decel_stop(0,0.01);
```

如图 7-12 所示：

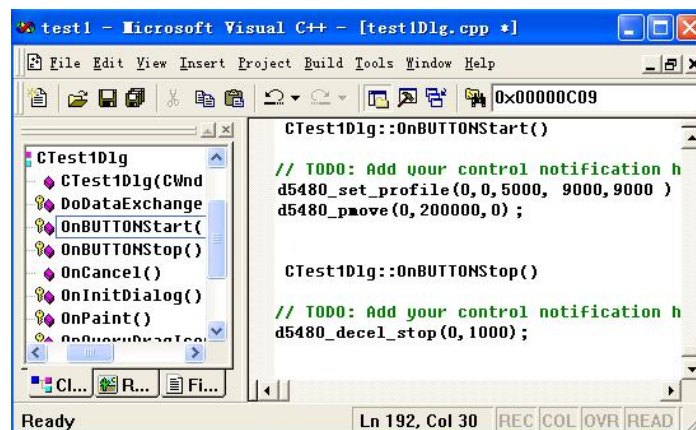


图 7-12 程序中调用运动控制卡库函数

(14) 编译，运行，按下“启动”按钮，第 0 轴就会输出长度为 200000 的脉冲，运动中可以按下“停止”按钮便会减速停止脉冲输出，如图 7-13 所示：

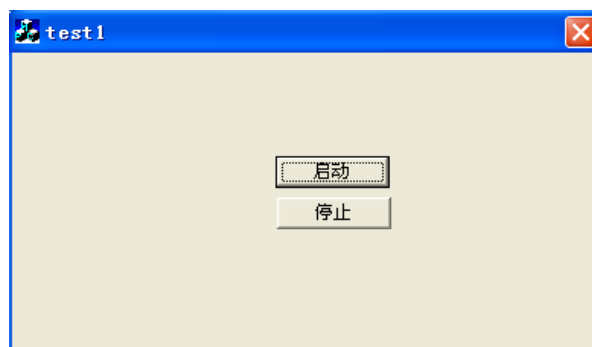


图 7-13 程序运行界面

注：我们在 DMC-5480 的用户光盘中还提供了以下几个示例供用户参考：

示例一，单轴运动。卡的初始化，关闭操作，脉冲模式设置和单轴运动命令。

示例二，回原点运动。包括回原点的速度和方向的设置。

示例三， 直线插补运动。

示例四， 两轴圆弧插补运动。

示例五，连续插补运动。

示例六， 手轮运动。

示例七，通用专用输入输出。

示例八，连续插补运动。

8 附录

8.1 硬件信号接口表

8.1.1 接口X1 引脚定义

如图 8-1 所示，接口 X1 是电机控制及 I/O 信号 SCSI-II 型 68 针插座。

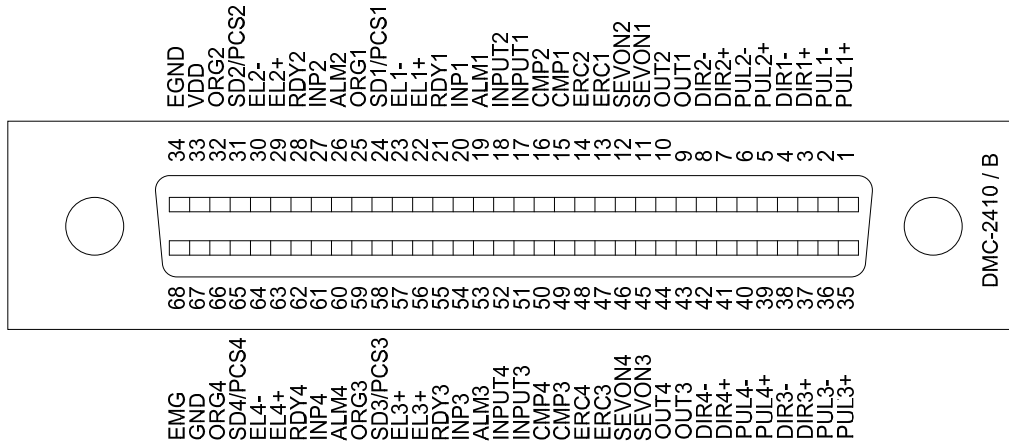


图8-1 接口X1示例

其引脚号和信号对应关系见表 8-1 所示：

表 8-1 接口 X1 引脚号和信号关系表

序	名称	I/O	说 明	序	名称	I/O	说 明
1	PUL1+	O	轴 1 脉冲信号(+)	35	PUL3+	O	轴 3 脉冲信号(+)
2	PUL1-	O	轴 1 脉冲信号(-)	36	PUL3-	O	轴 3 脉冲信号(-)
3	DIR1+	O	第一轴方向信号(+)	37	DIR3+	O	第三轴方向信号(+)
4	DIR1-	O	第一轴方向信号(-)	38	DIR3-	O	第三轴方向信号(-)
5	PUL2+	O	第二轴脉冲信号(+)	39	PUL4+	O	第四轴脉冲信号(+)
6	PUL2-	O	第二轴脉冲信号(-)	40	PUL4-	O	第四轴脉冲信号(-)
7	DIR2+	O	第二轴方向信号(+)	41	DIR4+	O	第四轴方向信号(+)
8	DIR2-	O	第二轴方向信号(-)	42	DIR4-	O	第四轴方向信号(-)
9	OUT1	O	通用输出信号 1	43	OUT3	O	通用输出信号 3
10	OUT2	O	通用输出信号 2	44	OUT4	O	通用输出信号 4
11	SEVON1/OUT25	O	第一轴伺服使能信号	45	SEVON3/OUT27	O	第三轴伺服使能信号
12	SEVON2/OUT26	O	第二轴伺服使能信号	46	SEVON4/OUT28	O	第四轴伺服使能信号
13	ERC1/OUT29	O	第一轴误差清除信号	47	ERC3/OUT31	O	第三轴误差清除信号
14	ERC2/OUT30	O	第二轴误差清除信号	48	ERC4/OUT32	O	第四轴误差清除信号
15	CMP1/OUT21	O	第一轴比较输出信号	49	CMP3/OUT23	O	第三轴比较输出信号
16	CMP2/OUT22	O	第二轴比较输出信号	50	CMP4/OUT24	O	第四轴比较输出信号
17	IN1	I	通用输入信号 1	51	IN3	I	通用输入信号 3

序	名称	I/O	说 明	序	名称	I/O	说 明
18	IN2	I	通用输入信号 2	52	IN4	I	通用输入信号 4
19	ALM1/IN33	I	第一轴报警信号	53	ALM3/IN35	I	第三轴报警信号
20	INP1/IN29	I	第一轴位置到信号	54	INP3/IN31	I	第三轴位置到信号
21	RDY1/IN21	I	第一轴准备好输入信号	55	RDY3/IN23	I	第三轴准备好输入信号
22	EL1+	I	第一轴限位信号(+)	56	EL3+	I	第三轴限位信号(+)
23	EL1-	I	第一轴限位信号(-)	57	EL3-	I	第三轴限位信号(-)
24	IN25	I	第一轴减速点信号	58	IN27	I	第三轴减速点信号
25	ORG1	I	第一轴原点信号	59	ORG3	I	第三轴原点信号
26	ALM2/IN34	I	第二轴报警信号	60	ALM4/IN36	I	第四轴报警信号
27	INP2/IN30	I	第二轴位置到信号	61	INP4/IN32	I	第四轴位置到信号
28	RDY2/IN22	I	第二轴准备好输入信号	62	RDY4/IN24	I	第四轴准备好输入信号
29	EL2+	I	第二轴限位信号(+)	63	EL4+	I	第四轴限位信号(+)
30	EL2-	I	第二轴限位信号(-)	64	EL4-	I	第四轴限位信号(-)
31	IN26	I	第二轴减速点信号	65	IN28	I	第四轴减速点信号
32	ORG2	I	第二轴原点信号	66	ORG4	I	第四轴原点信号
33	VDD	I	外部电源,+12V~+24V	67	GND		PC 电源地
34	EGND		外部电源电源地	68	EMG	I	紧急停止(对全部轴)

注意：以下特殊 I/O 口不用时，可作通用输入输出口。

CMP1~CMP4 做通用输出时为 OUT21~OUT24；

SEVON1~SEVON4 做通用输出时为 OUT25~OUT28；

ERC1~ERC4 做通用输出时为 OUT29~OUT32；

RDY1~RDY4 做通用输入时为 IN21~IN24；

INP1~INP4 做通用输入时为 IN29~IN32；

ALM1~ALM4 做通用输入时为 IN33~IN36。

8.1.2 接口X2 引脚定义

接口 X2 是编码器信号和其辅助信号的接口，为 40 针 IDC 插座。其对外接口为 DB37 插座，如图 8-2 所示：

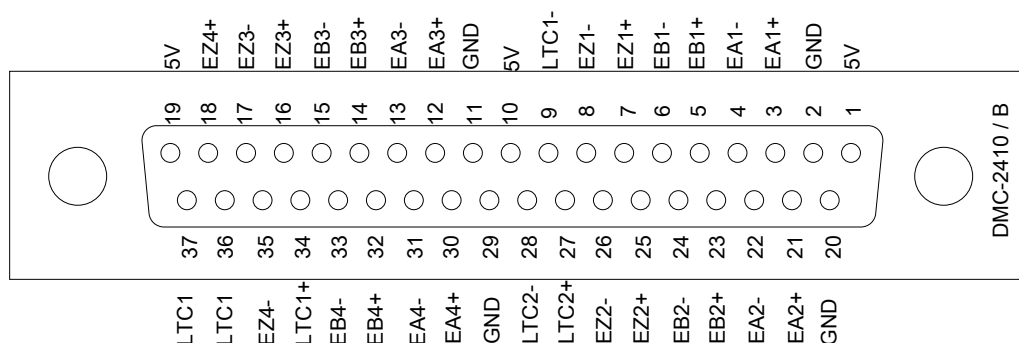


图8-2 接口X2示例

其引脚号和信号对应关系见表 8-2 所示。

表 8-2 接口 X2 引脚号和信号关系表

序号	名称	I/O	说 明	序号	名称	I/O	说 明
1	5V		PC5V 电源	20	GND		PC 电源地
2	GND	I	PC 电源地	21	EA2+	I	第二轴编码器 A 相(+)
3	EA1+	I	第一轴编码器 A 相(+)	22	EA2-	I	第二轴编码器 A 相(-)
4	EA1-	I	第一轴编码器 A 相(-)	23	EB2+	I	第二轴编码器 B 相(+)
5	EB1+	I	第一轴编码器 B 相(+)	24	EB2-	I	第二轴编码器 B 相(-)
6	EB1-	I	第一轴编码器 B 相(-)	25	EZ2+	I	第二轴编码器 Z 信号(+)
7	EZ1+	I	第一轴编码器 Z 信号(+)	26	EZ2-	I	第二轴编码器 Z 信号(-)
8	EZ1-	I	第一轴编码器 Z 信号(-)	27	LTC2+	I	第二轴锁存信号(+)
9	LTC1-	I	第一轴锁存信号(-)	28	LTC2-	I	第二轴锁存信号(-)
10	5V		PC5V 电源	29	GND		PC 电源地
11	GND		PC 电源地	30	EA4+	I	第四轴编码器 A 相(+)
12	EA3+	I	第三轴编码器 A 相(+)	31	EA4-	I	第四轴编码器 A 相(-)
13	EA3-	I	第三轴编码器 A 相(-)	32	EB4+	I	第四轴编码器 B 相(+)
14	EB3+	I	第三轴编码器 B 相(+)	33	EB4-	I	第四轴编码器 B 相(-)
15	EB3-	I	第三轴编码器 B 相(-)	34	LTC1+	I	第一轴锁存信号(+)
16	EZ3+	I	第三轴编码器 Z 信号(+)	35	EZ4-	I	第四轴编码器 Z 信号(-)
17	EZ3-	I	第三轴编码器 Z 信号(-)	36	LTC1	O	LTC1 反相输出
18	EZ4+	I	第四轴编码器 Z 信号(+)	37	LTC1	O	LTC1 反相输出
19	5V		PC5V 电源				

8.1.3 接口X3 引脚定义

接口 X3 是通用 I/O 信号的接口，为 40 针 IDC 插座。其对外接口为 DB37 插座，如图 8-3 所示：

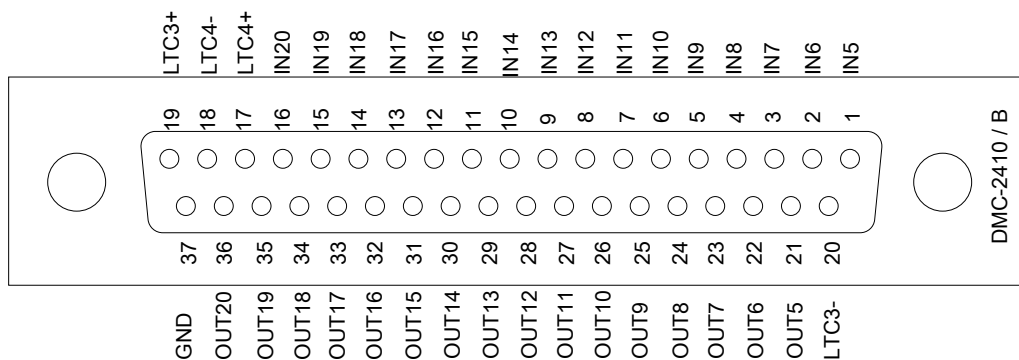


图8-3 接口X3示例

其引脚号和信号对应关系见表 8-3。

表 8-3 X3 引脚号和信号关系表

脚号	名称	I/O	功 能	脚号	名称	I/O	功 能
1	IN5	I	通用输入 5	20	LTC3-	I	第三轴锁存输入信号(-)
2	IN6	I	通用输入 6	21	OUT5	O	通用输出 5
3	IN7	I	通用输入 7	22	OUT6	O	通用输出 6
4	IN8	I	通用输入 8	23	OUT7	O	通用输出 7
5	IN9	I	通用输入 9	24	OUT8	O	通用输出 8
6	IN10	I	通用输入 10	25	OUT9	O	通用输出 9
7	IN11	I	通用输入 11	26	OUT10	O	通用输出 10
8	IN12	I	通用输入 12	27	OUT11	O	通用输出 11
9	IN13	I	通用输入 13	28	OUT12	O	通用输出 12
10	IN14	I	通用输入 14	29	OUT13	O	通用输出 13
11	IN15	I	通用输入 15	30	OUT14	O	通用输出 14
12	IN16	I	通用输入 16	31	OUT15	O	通用输出 15
13	IN17	I	通用输入 17	32	OUT16	O	通用输出 16
14	IN18	I	通用输入 18	33	OUT17	O	通用输出 17/CMP5'
15	IN19	I	通用输入 19	34	OUT18	O	通用输出 18/CMP6'
16	IN20	I	通用输入 20	35	OUT19	O	通用输出 19/CMP7'
17	LTC4+	I	第四轴锁存输入信号(+)	36	OUT20	O	通用输出 20/CMP8'
18	LTC4-	I	第四轴锁存输入信号(-)	37	GND		PC 电源地
19	LTC3+	I	第三轴锁存输入信号(+)				

注：OUT17~OUT20 可做比较输出口 CMP5~CMP8。

8.1.4 接口X5 引脚定义

接口 X5 用于手轮输入口，如图 8-4 所示：

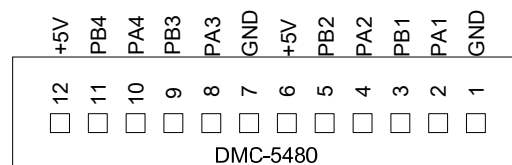


图 8-4 接口 X5 示例

引脚号和信号对应关系见表 8-4。

表 8-4 接口 X5 引脚号和信号关系表

序号	名称	说 明
1	GND	PC 电源地
2	PA1	第一轴手轮 A 相脉冲输入
3	PB1	第一轴手轮 B 相脉冲输入
4	PA2	第二轴手轮 A 相脉冲输入
5	PB2	第二轴手轮 B 相脉冲输入
6	+5V	PC 电源

7	GND	PC 电源地
8	PA3	第三轴手轮 A 相脉冲输入
9	PB3	第三轴手轮 B 相脉冲输入
10	PA4	第四轴手轮 A 相脉冲输入
11	PB4	第四轴手轮 B 相脉冲输入
12	+5V	PC 电源

8.2 运动控制函数库

雷泰 DMC-5480 运动控制卡的运动控制函数库共有 16 类 109 个函数，分类列表如下：

8.2.1 函数列表

	函数名	描 述
初始化函数	d5480_board_init	初始化控制卡
	d5480_board_close	关闭控制卡
	d5480_get_card_ID	读取控制卡卡号
	d5480_get_lib_version	读取函数库版本
	d5480_get_card_version	读取控制卡硬件版本
	d5480_get_card_soft_version	读取控制卡硬件的固件版本
	d5480_get_total_axes	读取指定卡轴数
脉冲模式设置函数	d5480_set_pulse_outmode	设定脉冲输出模式
	d5480_get_pulse_outmode	读取脉冲输出模式
回原点函数	d5480_config_home_mode	设置回原点模式
	d5480_get_config_home_mode	读取回原点模式
	d5480_home_move	启动回原点运动
位置计数器控制函数	d5480_get_position	读取指定轴的当前位置
	d5480_set_position	设定指定轴的当前位置
	d5480_get_counter_flag	读取计数器标志
	d5480_reset_counter_flag	复位计数器的计数标志
	d5480_reset_clear_flag	复位计数器的清零标志
编码器相关函数	d5480_counter_config	设定编码器的计数方式
	d5480_get_counter_config	读取编码器的计数方式
	d5480_set_encoder	设置编码器反馈脉冲计数值
	d5480_get_encoder	读取编码器反馈脉冲计数值
	d5480_set_speaker_logic	设置编码器 Speaker 和 LED 的输出逻辑
	d5480_get_speaker_logic	读取设置编码器 Speaker 和 LED 输出逻辑
位置锁存相关	d5480_config_latch_mode	设置锁存方式

函数	d5480_get_config_latch_mode	读取设置锁存方式
	d5480_get_latch_value	读取编码器锁存器的值
	d5480_get_latch_flag	读取锁存器标志
	d5480_reset_latch_flag	复位锁存器标志
	d5480_trigger_chunnel	选择全部锁存时的外触发信号通道
位置比较功能函数	d5480_compare_config	配置比较器
	d5480_compare_get_config	读取配置比较器
	d5480_compare_clear_points	清除所有比较点
	d5480_compare_add_point	添加比较点
	d5480_compare_get_current_point	读取当前比较点
	d5480_compare_get_points_runned	查询已经比较过的点
	d5480_compare_get_points_remained	查询可以加入的比较点数量
单轴运动控制函数	d5480_set_profile	设置速度曲线
	d5480_get_profile	读取速度曲线设置
	d5480_set_s_profile	设置 S 形速度曲线
	d5480_get_s_profile	读取 S 形速度曲线设置
	d5480_pmove	单轴位置控制（点位运动）
连续运动及运动速度控制函数	d5480_vmove	单轴速度控制（连续运动）
	d5480_check_done	读取指定轴的运动状态
	d5480_reset_target_position	在运动中改变目标位置
	d5480_config_softlimit	配置软件限位
	d5480_get_config_softlimit	读取软限位配置
	d5480_change_speed	在线改变指定轴的运动速度
	d5480_read_current_speed	读取指定轴的当前速度
	d5480_decel_stop	单轴减速停止
	d5480_imd_stop	单轴立即停止
	d5480_simultaneous_stop	同步停止所有轴
	d5480_emg_stop	紧急停止所有轴
直线插补函数	d5480_set_vector_profile	设置插补运动的矢量速度
	d5480_get_vector_profile	读取插补运动的矢量速度设置
	d5480_read_vector_speed	读取当前卡的插补速度
	d5480_line2	任意 2 轴直线插补
	d5480_line3	任意 3 轴直线插补
	d5480_line4	指定 4 轴直线插补
两轴圆弧插补函数	d5480_arc_move	任意两轴绝对位置圆弧插补
	d5480_rel_arc_move	任意两轴相对位置圆弧插补

螺旋插补函数	d5480_rel_helix_move	相对螺旋插补运动函数
	d5480_helix_move	绝对螺旋插补运动函数
连续插补运动函数	d5480_conti_set_mode	连续插补模式设置
	d5480_conti_get_mode	读取连续插补模式设置
	d5480_conti_open_list	打开连续缓存区
	d5480_conti_close_list	关闭连续缓存区
	d5480_conti_decel_stop_list	连续插补中减速停止
	d5480_conti_sudden_stop_list	连续插补中急停
	d5480_conti_pause_list	连续插补中暂停
	d5480_conti_start_list	开始连续插补
	d5480_conti_check_remain_space	查连续插补剩余缓存数
	d5480_conti_restrain_speed	多轴连续插补拐点限速
	d5480_conti_change_speed_ratio	设置插补中动态变速
	d5480_conti_lines	连续直线插补函数
	d5480_conti_arc	连续圆弧插补函数
	d5480_conti_extern_lines	带扩展标号的连续直线插补函数
	d5480_conti_extern_arc	带扩展标号的连续圆弧插补函数
	d5480_conti_read_current_mark	读取当前连续插补段的标号
伺服电机驱动器专用接口信号的设定函数	d5480_config_INP_PIN	设置 INP 信号
	d5480_get_config_INP_PIN	读取设置 INP 信号
	d5480_config_ERC_PIN	设置 ERC 信号
	d5480_get_config_ERC_PIN	读取设置 ERC 信号
	d5480_config_ALM_PIN	设置 ALM 信号
	d5480_get_config_ALM_PIN	读取设置 ALM 信号
	d5480_config_EZ_PIN	设置 EZ 信号
	d5480_get_config_EZ_PIN	读取设置 EZ 信号
	d5480_config_LTC_PIN	设置 LTC 信号
	d5480_get_config_LTC_PIN	读取设置 LTC 信号
	d5480_config_EL_MODE	设置 EL 信号
	d5480_get_config_EL_PIN	读取设置 EL 信号
	d5480_config_HOME_PIN_logic	设置 HOME 信号
	d5480_get_config_HOME_PIN_logic	读取设置 HOME 信号
	d5480_write_SEVON_PIN	输出 SEVON 信号
	d5480_set_backlash	设置间隙补偿值
	d5480_get_backlash	读取间隙补偿值
	d5480_read_RDY_PIN	读取 RDY 状态

	d5480_write_ERC_PIN	控制 ERC 信号输出
	d5480_config_EMG_PIN	设置 EMG 信号
	d5480_get_config_EMG_PIN	读取设置 EMG 信号
	d5480_axis_io_status	读取指定轴有关运动信号的状态
手轮运动控制 函数	d5480_set_handwheel_inmode	设置手轮脉冲输入方式
	d5480_get_handwheel_inmode	读取手轮脉冲输入方式
	d5480_handwheel_move	启动手轮运动
通用 I/O 接口 函数	d5480_read_inbit	读取输入口的状态
	d5480_write_outbit	设置输出口的状态
	d5480_read_outbit	读取输出口的状态
	d5480_read_inport	读取输入端口的值
	d5480_read_outport	读取输出端口的值
	d5480_write_outport	设置输出端口的值

8.2.2 函数说明

下面我们对这些函数分类进行详细说明。

8.2.2.1 初始化函数

WORD d5480_board_init(void)

功 能：为控制卡分配系统资源，并初始化控制卡

参 数：无

返回值：卡数(0 - 8)，其中 0 表示没有卡

void d5480_board_close(void)

功 能：关闭控制卡，释放系统资源

参 数：无

返回值：无

DWORD d5480_get_card_ID(WORD card)

功 能：读取控制卡的 ID 拨码开关。

参 数：卡号

返回值：控制卡的 ID 拨码开关标志值。

DWORD d5480_get_lib_version(void)

功 能：读取软件版本号。

参 数: 卡号

返回值: 控制卡函数库的版本号

DWORD d5480_get_card_version(WORD card)

功 能: 读取卡的硬件版本号

参 数: 无

返回值: 硬件版本号

DWORD d5480_get_card_soft_version (WORD card, WORD* firm_id, DWORD* sub_firm_id)

功 能: 读取卡硬件的固件版本号

参 数: card: 卡号。

Firm_id: 固件版本号 id

sub_firm_id: 固件版本号子 id

返回值: 错误代码。（详见 7.15 章节，以下同）

DWORD d5480_get_total_axes (WORD card)

功 能: 读取卡的总轴数

参 数: 无

返回值: 轴数

8.2.2.2 脉冲模式设置函数









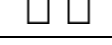
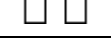
DWORD d5480_set_pulse_outmode(WORD axis, WORD outmode)



DWORD d5480_get_pulse_outmode(WORD axis, WORD* outmode)

功 能: 设置/读取指定轴的脉冲输出模式

参 数: axis 指定轴号

outmode 脉冲输出方式选择，其值如下表所示：

输出脉冲类型 outmode	正方向脉冲		负方向脉冲	
	OUT 输出端	DIR 输出端	OUT 输出端	DIR 输出端
0		高电平		低电平
1		高电平		低电平
2		低电平		高电平
3		低电平		高电平
4		高电平	高电平	

5		低电平	低电平	
---	---	-----	-----	---

返回值：错误代码

注意：在调用运动函数（如：d5480_vmove 等）输出脉冲之前，一定要根据驱动器接收脉冲的模式调用 d5480_set_pulse_outmode 设置控制卡脉冲输出模式。

8.2.2.3 回原点运动函数

DWORD d5480_config_home_mode(WORD axis,WORD home_dir, double vel, WORD mode, WORD EZ_count)

DWORD d5480_get_config_home_mode(WORD axis,WORD* home_dir, double* vel,WORD* home_mode,WORD* EZ_count)

功 能：设定/读取指定轴的回原点模式

参 数：axis 指定轴号

 home_dir 回零方向， 1 正向, 2:负向

 vel 回零速度

 mode 回原点的信号模式

 0 - 一次回零

 1 - 二次回零

 2 - 一次回零加回找

 10 - 以 EZ 作为原点进行一次回零

 11 - 以 EZ 作为原点进行一次回零，碰到限位后自动

反找。

 EZ_count 保留

返回值：错误代码

DWORD d5480_home_move(WORD axis)

功 能：单轴回原点运动

参 数：axis 指定轴号

返回值：错误代码

8.2.2.4 位置计数锁存比较

8.2.2.4.1 位置计数器控制函数

long d5480_get_position(WORD axis)

功 能：读取指定轴的指令脉冲位置

参 数：axis 指定轴号

返回值：指定运动轴的命令脉冲数，单位：脉冲

DWORD d5480_set_position(WORD axis,long current_position)

功 能：设置指定轴的指令脉冲位置

参 数: axis 指定轴号
 current_position 绝对位置值

返回值: 无

long d5480_get_counter_flag(WORD cardno)

功 能: 读取指定控制卡的计数器的标识位

参 数: cardno 指定控制卡号

返回值: 见表

返回值位号	描述
0	指定卡的 X 轴 借位触发标志: 计数器每下溢出时触发一次
1	进位触发标志: 计数器每上溢出时触发一次
2	符号标志: 计数器上溢出为 0, 下溢出为 1
3	上下计数标志: 上计数时为 1, 下计数时为 0
4~7	保留
8	指定卡的 Y 轴 借位触发标志: 计数器每下溢出时触发一次
9	进位触发标志: 计数器每上溢出时触发一次
10	符号标志: 计数器上溢出为 0, 下溢出为 1
11	上下计数标志: 上计数时为 1, 下计数时为 0
12~15	保留
16	指定卡的 Z 轴 借位触发标志: 计数器每下溢出时触发一次
17	进位触发标志: 计数器每上溢出时触发一次
18	符号标志: 计数器上溢出为 0, 下溢出为 1
19	上下计数标志: 上计数时为 1, 下计数时为 0
20~23	保留
24	指定卡的 U 轴 借位触发标志: 计数器每下溢出时触发一次
25	进位触发标志: 计数器每上溢出时触发一次
26	符号标志: 计数器上溢出为 0, 下溢出为 1
27	上下计数标志: 上计数时为 1, 下计数时为 0
28~31	保留

DWORD d5480_reset_counter_flag(WORD cardno)

功 能: 复位计数器的计数标志位, 范围 (0 — N - 1, N 为卡数)

参 数: cardno 指定控制卡号

返回值: 无

DWORD d5480_reset_clear_flag(WORD cardno)

功 能: 复位计数器的清零标志位, 范围 (0 — N - 1, N 为卡数)

参 数: cardno 指定控制卡号

返回值: 无

8.2.2.4.2 编码器相关函数

DWORD d5480_counter_config(WORD axis, WORD mode)

DWORD d5480_get_counter_config(WORD axis, WORD* mode)

功 能：设置/读取编码器的计数方式

参 数：axis 指定轴号

mode 编码器的计数方式：

0 -- 非 A/B 相脉冲信号（脉冲+方向信号）

1 -- 1 倍 A/B 相脉冲信号

2 -- 2 倍 A/B 相脉冲信号

3 -- 4 倍 A/B 相脉冲信号

返回值：错误代码

long d5480_get_encoder(WORD axis)

功 能：读取指定轴编码器反馈位置脉冲计数值，范围：28 位有符号数

参 数：axis 指定轴号

返回值：位置反馈脉冲值，单位：脉冲数

DWORD d5480_set_encoder(WORD axis, long encoder_value)

功 能：设置指定轴编码器反馈脉冲计数值，范围：28 位有符号数

参 数：axis 指定轴号

encoder_value 编码器的设定值。

返回值：无

DWORD d5480_set_speaker_logic(WORD cardno, WORD logic)

DWORD d5480_get_speaker_logic(WORD cardno, WORD* logic)

功 能：选择编码器 Speaker 和 LED 的输出逻辑，默认为低有效

输 入：cardno 卡号

logic 0：低有效，1：高有效

返回值：无

8.2.2.4.3 位置锁存相关函数

DWORD d5480_config_latch_mode(WORD cardno, WORD all_enable)

DWORD d5480_get_config_latch_mode(WORD cardno, WORD* all_enable)

功 能：设置锁存方式为单轴锁存或是四轴同时锁存

参 数：cardno 指定控制卡号

all_enable 锁存方式：0—单独锁存，1—四轴同时锁存

返回值：无

long d5480_get_latch_value(WORD axis)

功 能：读取编码器锁存器的值

参 数：axis 指定轴号

返回值：锁存器内的编码器脉冲数，单位：脉冲

long d5480_get_latch_flag(WORD cardno)

功 能：读取指定控制卡的锁存器的标志位

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）

返回值：见表

返回值位号	描述
0	1: 指定卡的 0 轴有触发信号.
1	1: 1 轴有触发信号.
2	1: 2 轴有触发信号.
3	1: 3 轴有触发信号.
4	1: 指定卡的 0 轴有清零信号
5	1: 1 轴有清零信号
6	1: 2 轴有清零信号
7	1: 3 轴有清零信号
8	1: 0 轴位置已锁存
9	1: 1 轴位置已锁存
10	1: 2 轴位置已锁存
11	1: 3 轴位置已锁存
12	1: 指定卡的 0 轴位置已清零
13	1: 1 轴位置已清零
14	1: 2 轴位置已清零
15	1: 3 轴位置已清零
16~31	保留

DWORD d5480_reset_latch_flag(WORD cardno)

功 能：复位指定控制卡的锁存器的标志位

参 数：cardno 指定控制卡号

返回值：无

DWORD d5480_triger_chunnel(WORD cardno, WORD num)

功 能：选择全部锁存时的外触发信号通道;可以由二个信号通道输入，即 LTC1, LTC2；默认为 LTC1。

输 入：cardno 卡号

 Num 信号通道选择号

 0: LTC1 锁存四个轴，

 1: LTC2 锁存四个轴

返回值：无

8.2.2.4.4 位置比较功能函数

DWORD d5480_compare_config (WORD card, WORD enable, WORD axis, WORD cmp_source)

DWORD d5480_compare_get_config (WORD card, WORD* enable, WORD* axis, WORD* cmp_source)

功 能：设置/读取比较器配置

参 数：card 卡号
 enable 1：使能比较功能， 0：禁止比较功能
 axis 轴号
 cmp_source 比较源， 0：比较指令位置， 1：比较编码器位置

返回值：错误代码

DWORD d5480_compare_clear_points (WORD card)

功 能：清除所有比较点

参 数：card 卡号

返回值：错误代码

DWORD d5480_compare_add_point (WORD card, long pos, WORD dir, WORD action, long actpara)

功 能：添加比较点

参 数：card 卡号
 Pos: 位置坐标
 dir: 比较方向 0: 小于等于 1: 大于等于
 action: 比较点触发功能
 actpara: 比较点触发功能参数

返回值：错误代码

D5480_compare_add_point 函数 action, actpara 参数值含义：

action	actpara	功能
1	IO 号	打开 IO
2	IO 号	关闭 IO
3	IO 号	取反 IO
5	IO 号	输出 100us 脉冲
6	IO 号	输出 1ms 脉冲
7	IO 号	输出 10ms 脉冲
8	IO 号	输出 100ms 脉冲
11	速度值	当前轴变速
13	轴号	停止指定轴

long d5480_compare_get_current_point (WORD card)

功 能：读取当前比较点

参 数：card 卡号

返回值：错误代码

long d5480_compare_get_points_runned (WORD card)

功 能：查询已经比较过的点

参 数：card 卡号

返回值：错误代码

long d5480_compare_get_points_remained (WORD card)

功 能：查询可以加入的比较点数量

参 数：card 卡号

返回值：错误代码

8.2.2.5 单轴位置运动控制函数

8.2.2.5.1 点位运动函数

DWORD d5480_set_profile (WORD axis, double Min_Vel, double Max_Vel, double acc, double dec)

DWORD d5480_get_profile(WORD axis, double* Min_Vel, double* Max_Vel, double* acc, double* dec)

功 能：设定/读取速度曲线的最大速度、加速度、减速度

参 数：axis 要设置的轴号

Min_Vel 起初速度，单位 p/s

Max_Vel 最大速度，单位 p/s

acc 加速度，单位 p/s²

dec 减速度，单位 p/s²

返回值：错误代码

DWORD d5480_set_s_profile(WORD axis,double s_para)

DWORD d5480_get_s_profile(WORD axis,double* s_para)

功 能：设定/读取S形曲线运动的参数。具体参数含意请参阅[4.2.7.2 节S形速度曲线运动模式](#)

参 数：axis 要设置的轴号

s_para S 加速段占总加速段的比例，0~1.0 之间。当设置为 0 时，没有 S 段。

返回值：错误代码

DWORD d5480_pmove(WORD axis,long Dist,WORD posi_mode)

功 能：使指定轴做定长位移运动。

参 数: axis 指定轴号
 Dist (绝对/相对) 位移值, 单位: 脉冲数
 posi_mode 位移模式设定: 0 表示相对位移, 1 表示绝对位移
返回值: 无

8.2.2.5.2 连续运动及速度控制函数

DWORD d5480_vmove(WORD axis,WORD dir,double vel)

功 能: 使指定轴做连续运动

参 数: axis 指定轴号
 dir 指定运动的方向, 其中 0 表示负方向, 1 表示正方向
 vel vmove 的速度

返回值: 错误代码

WORD d5480_check_done(WORD axis)

功 能: 检测指定轴的运动状态, 停止或是在运行中。

参 数: axis 指定轴号

返回值: 0 表示指定轴正在运行, 1 表示指定轴已停止。

DWORD d5480_reset_target_position(WORD axis,long dist)

功 能: 在单轴相对运动中改变目标位置。注意: 该函数仅能在绝对位置模式下使用。

参 数: axis 指定轴号
 dist 相对位置值

返回值: 无

DWORD d5480_config_softlimit(WORD axis,WORD ON_OFF, WORD source_sel,WORD SL_action, long N_limit,long P_limit)

DWORD d5480_get_config_softlimit (WORD axis,WORD* ON_OFF, WORD* source_sel,WORD* SL_action,long* N_limit,long* P_limit)

功 能: 设置/读取软件限位的使能, 限位数值, 响应动作

参 数: axis 指定轴号
 ON_OFF 软限位使能, 0—禁止; 1—使能
 source_sel 比较源选择, 保留, 设置为指令脉冲。
 SL_action 限位动作, 0—急停, 1—减速停
 N_limit 负限位值
 P_limit 正限位值

返回值: 错误代码

DWORD d5480_change_speed(WORD axis,double Curr_Vel)

功 能: 在线改变指定轴的当前运动速度。该函数只适用于单轴运动中的变速。

参 数: axis 要设置的轴号

Curr_Vel 新的运行速度，单位 pps

返回值：错误代码

double d5480_read_current_speed(WORD axis)

功 能：读取当前速度值，单位 pps

参 数：axis 指定轴号

返回值：指定轴的速度脉冲数

DWORD d5480_decel_stop(WORD axis,double dec)

功 能：指定轴减速停止，调用此函数时立即减速，直到停止脉冲输出

参 数：axis 指定轴号

 dec 减速度，保留，暂不起作用。

返回值：错误代码

DWORD d5480_imd_stop(WORD axis)

功 能：使指定轴立即停止，没有任何减速的过程

参 数：axis 指定轴号

返回值：错误代码

DWORD d5480_simultaneous_stop(WORD axis)

功 能：使多卡上的所有轴同时停止运行。

该函数在指定轴的 CSTOP 端子上输出一个单脉冲的停止信号，如果有多个轴的

CSTOP 信号相连接时则所有 CSTOP 相连运动轴将同时停止运行。

参 数：axis 指定轴号

返回值：错误代码

DWORD d5480_emg_stop(void)

功 能：使所有的运动轴紧急停止。

参 数：无

返回值：错误代码

8.2.2.6 多轴运动函数

8.2.2.6.1 直线插补运动函数

DWORD d5480_set_vector_profile (WORD cardno,double s_para, double Max_Vel, double acc,double dec)

DWORD d5480_get_vector_profile(WORD cardno,double* s_para, double* Max_Vel, double* acc,double* dec)

功 能：设定/读取插补矢量运动曲线的 S 段参数、运行速度、加速度、减速度

参 数：s_para S 加速段占总加速段的比例，0–1.0 之间。当设置为 0 时，没有 S 段。

 Max_Vel 运行速度，单位 pps

 acc 加速度

 dec 减速度

返回值：错误代码

double d5480_read_vector_speed (WORD card)

功 能: 读取当前卡的插补速度, 单位 pps

参 数: card 卡号

返回值: 插补速度

DWORD d5480_line2(WORD axis1,long Dist1,WORD axis2,long Dist2,WORD posi_mode)

功 能: 指定任意两轴做插补运动

参 数: axis1 指定两轴插补的第一轴
axis2 指定两轴插补的第二轴
Dist1 指定 axis1 的位移值, 单位: 脉冲数
Dist2 指定 axis2 的位移值, 单位: 脉冲数
posi_mode 位移模式设定: 0 表示相对位移, 1 表示绝对位移

返回值: 错误代码



注意: 此函数可以对多条具有前后相连的线段实现连续插补的功能; 当使用连续插补时, posi_mode 的值必须为 0, 即设为相对位移模式。以下多轴插补函数均需同此设置, 故不再说明。连续插补方式的详细说明, 请参阅 5.5.4 章节。

DWORD d5480_line3(WORD *axis,long Dist1,long Dist2,long Dist3,WORD posi_mode)

功 能: 指定任意三轴做插补运动

参 数: axis 轴号列表的指针
Dist1 指定 axis[0]轴的位移值, 单位: 脉冲数
Dist2 指定 axis[1]轴的位移值, 单位: 脉冲数
Dist3 指定 axis[2]轴的位移值, 单位: 脉冲数
posi_mode 位移模式设定: 0 表示相对位移, 1 表示绝对位移

返回值: 错误代码

DWORD d5480_line4(WORD cardno, long Dist1, long Dist2, long Dist3, long Dist4,WORD posi_mode)

功 能: 指定四轴做插补运动

参 数: cardno 指定插补运动的板卡号, 范围 (0 — N - 1, N 为卡数)
Dist1 指定第一轴的位移值, 单位: 脉冲数
Dist2 指定第二轴的位移值, 单位: 脉冲数
Dist3 指定第三轴的位移值, 单位: 脉冲数
Dist4 指定第四轴的位移值, 单位: 脉冲数
posi_mode 位移模式设定: 0 表示相对位移, 1 表示绝对位移

返回值: 错误代码

8.2.2.6.2 两轴圆弧插补函数

DWORD d5480_arc_move(WORD *axis,long *target_pos,long *cen_pos, WORD arc_dir)

功 能: 指定任意的两轴以当前位置为起点, 按指定的圆心、目标绝对位置和方向作圆弧插补运动,

参 数: axis 轴号列表指针
 target_pos 目标绝对位置列表指针, 单位: 脉冲数
 cen_pos 圆心绝对位置列表指针, 单位: 脉冲数
 arc_dir 圆弧方向: 0 表示顺时针, 1 表示逆时针

返回值: 错误代码

DWORD d5480_rel_arc_move(WORD *axis,long *rel_pos,long *rel_cen, WORD arc_dir)

功 能: 指定任意的两轴以当前位置为起点, 按指定的圆心、目标相对位置和方向作圆弧插补运动,

参 数: axis 轴号列表指针
 rel_pos 目标相对位置列表指针, 单位: 脉冲数
 cen_pos 圆心相对位置列表指针, 单位: 脉冲数
 arc_dir 圆弧方向: 0 表示顺时针, 1 表示逆时针

返回值: 错误代码

8.2.2.6.3 螺旋插补运动函数

DWORD d5480_rel_helix_move(WORD card, WORD* axisList, long *rel_pos,long *rel_cen,long height, WORD arc_dir)

功 能: 相对坐标螺旋插补函数

参 数: card 卡号
 axisList 轴号列表
 target_pos 终点坐标指针
 cen_pos: 圆心坐标指针
 height 第三轴的运行距离
 arc_dir 圆弧方向, 0-顺时针, 1-逆时针

返 回 值: 错误码

DWORD d5480_helix_move(WORD card, WORD* axisList, long *target_pos,long *cen_pos,long height, WORD arc_dir)

功 能: 绝对坐标螺旋插补函数

参 数: card 卡号
 axisList 轴号列表
 target_pos 终点坐标指针
 cen_pos: 圆心坐标指针,
 height 第三轴的运行距离
 arc_dir 圆弧方向, 0-顺时针, 1-逆时针

返 回 值: 错误码

8.2.2.6.4 连续插补运动函数

DWORD d5480_conti_set_mode(WORD card, WORD conti_mode, double conti_vl, double conti_para, double filter)

DWORD d5480_conti_get_mode(WORD card, WORD* conti_mode, double* conti_vl, double* conti_para, double* filter)

功 能：设置/读取连续插补模式

参 数：card 卡号

conti_mode 0- 自动拐点减速, 1-用户自定义拐点减速

conti_vl 连续插补模式低速设置

conti_para 连续插补限速范围参数， 设置为 10 - 30 之间，值越小，拐角减速角度范围越大。

filter 连续插补限速加速度参数，曲线加速度限速，设置越小，小圆限速越大。单位： pulse/s/s

返回值：错误代码

DWORD d5480_conti_open_list(WORD axisNum, WORD *piaxisList)

功 能：打开连续缓存区

参 数：axisNum 连续插补轴数

piaxisList 连续插补轴号列表

返回值：错误代码

DWORD d5480_conti_close_list (WORD card)

功 能：关闭连续缓存区

参 数：card 卡号

返回值：错误代码

DWORD d5480_conti_decel_stop_list (WORD card)

功 能：连续插补中减速停止

参 数：card 卡号

返回值：错误代码

DWORD d5480_conti_sudden_stop_list (WORD card)

功 能：连续插补中急停

参 数：card 卡号

返回值：错误代码

DWORD d5480_conti_pause_list (WORD card)

功 能: 连续插补中暂停

参 数: card 卡号

返回值: 错误代码

DWORD d5480_conti_start_list (WORD card)

功 能: 开始连续插补

参 数: card 卡号

返回值: 错误代码

DWORD d5480_conti_check_remain_space (WORD card)

功 能: 查连续插补剩余缓存数

参 数: card 卡号

返回值: 剩余空间数

DWORD d5480_conti_restrain_speed (WORD card, double v)

功 能: 多轴连续插补拐点限速

参 数: card 卡号

v: 限速速度值

返回值: 错误代码

DWORD d5480_conti_change_speed_ratio (WORD card , double percent)

功 能: 设置插补中动态变速

参 数: card 卡号

percent: 速度比例

返回值: 错误代码

DWORD d5480_conti_lines (WORD axisNum, WORD *piaxisListw, long

***pPosList, WORD posi_mode)**

功 能: 连续直线插补函数

参 数: axisNum 轴数

piaxisList 轴号列表,

pPosList 位置列表

posi_mode 0 - 相对, 1-绝对位置模式

返回值: 错误代码

DWORD d5480_conti_arc (WORD *axis, long *rel_pos, long *rel_cen, WORD

arc_dir, WORD posi_mode)

功 能: 连续圆弧插补函数

参 数: axis 轴号列表,
rel_pos 圆弧终点位置列表
rel_cen 圆弧圆心位置列表
arc_dir 0 -顺时针, 1-逆时针
posi_mode 0 - 相对, 1-绝对位置模式
返回值: 错误代码

DWORD d5480_conti_extern_lines (WORD axisNum, WORD *piaxisListw, long *pPosList, WORD posi_mode, long imark)

功 能: 连续直线插补函数

参 数: axisNum 轴数
piaxisList 轴号列表,
pPosList 位置列表
posi_mode 0 - 相对, 1-绝对位置模式
imark 当前段的标号设置

返回值: 错误代码

DWORD d5480_conti_extern_arc (WORD *axis, long *rel_pos, long *rel_cen, WORD arc_dir, WORD posi_mode, long imark)

功 能: 连续圆弧插补函数

参 数: axis 轴号列表,
rel_pos 圆弧终点位置列表
rel_cen 圆弧圆心位置列表
arc_dir 0 -顺时针, 1-逆时针
posi_mode 0 - 相对, 1-绝对位置模式
imark 当前段的标号设置

返回值: 错误代码

DWORD d5480_conti_read_current_mark (WORD card)

功 能: 读取当前运行的段标号

参 数: card 卡号

返回值: 当前段的标号

8.2.2.7 驱动器专用接口信号的设定函数

DWORD d5480_config_INP_PIN(WORD axis, WORD enable, WORD inp_logic)

DWORD d5480_get_config_INP_PIN(WORD axis, WORD* enable, WORD*

inp_logic)

功 能: 设置/读取 INP 信号使能及其有效的逻辑电平

参 数: axis 指定轴号
enable 允许/禁止信号功能: 0—无效, 1—有效
inp_logic 设置 INP 信号的有效电平: 0—低电平有效, 1—高电平有效
返回值: 错误代码

**DWORD d5480_config_ERC_PIN(WORD axis, WORD enable, WORD
erc_logic, WORD erc_width, WORD erc_off_time)**

**DWORD d5480_get_config_ERC_PIN(WORD axis, WORD* enable, WORD*
erc_logic, WORD* erc_width, WORD* erc_off_time)**

功 能: 设置/读取 ERC 信号使能及其有效电平和输出方式

参 数: axis 指定轴号
enable 范围: 0 - 3;
0—不自动输出 ERC 信号
1—输出 ERC 信号
erc_logic 设置 INP 信号的有效电平: 0—低电平有效, 1—高电平有效
erc_width 误差清除信号 ERC 有效输出宽度时间。
单位: us
erc_off_time ERC 信号的关断时间:
单位: us

返回值: 错误代码

**DWORD d5480_config_ALM_PIN(WORD axis, WORD alm_logic, WORD
alm_action)**

**DWORD d5480_get_config_ALM_PIN(WORD axis, WORD* enable, WORD*
alm_logic, WORD* alm_action)**

功 能: 设置 ALM 的逻辑电平及其工作方式

参 数: axis 指定轴号
alm_logic ALM 信号的输入电平: 0—低电平有效, 1—高电平有效
alm_action ALM 信号的制动方式: 0—立即停止, 1—减速停止

返回值: 错误代码

DWORD d5480_config_EZ_PIN(WORD axis, WORD ez_logic, WORD ez_mode)

**DWORD d5480_get_config_EZ_PIN(WORD axis, WORD* ez_logic, WORD*
ez_mode)**

功 能: 设置指定轴的 EZ 信号的有效电平及其作用

参 数: axis 指定轴号
ez_logic EZ 信号逻辑电平: 0—低有效, 1—高有效
ez_mode EZ 信号的工作方式:
0—EZ 信号无效
1—EZ 是计数器复位信号
2—EZ 是原点信号, 且不复位计数器
3—EZ 是原点信号, 且复位计数器

返回值: 错误代码

DWORD d5480_config_LTC_PIN(WORD axis,WORD ltc_logic, WORD ltc_mode)

DWORD d5480_get_config_LTC_PIN(WORD axis,WORD* ltc_logic, WORD* ltc_mode)

功 能: 设置指定轴“锁存”信号的有效电平及其和工作方式。

参 数: axis 指定轴号
 ltc_logic LTC 信号逻辑电平: 0—低有效, 1—高有效
 ltc_mode 保留, 可设为任意值

返回值: 错误代码

DWORD d5480_config_EL_MODE(WORD axis, WORD el_logic,WORD el_mode)

DWORD d5480_get_config_EL_PIN(WORD axis, WORD* el_logic, WORD* el_mode)

功 能: 设置 EL 信号的有效电平及制动方式

参 数: axis 指定轴号
 el_logic EL 有效电平
 el_mode EL 有效电平和制动方式:
 0—立即停
 1—减速停

返回值: 错误代码

DWORD d5480_set_HOME_PIN_logic(WORD axis,WORD org_logic,WORD filter)

DWORD d5480_get_config_HOME_PIN_logic(WORD axis,WORD* org_logic,WORD* filter)

功 能: 设置 ORG 信号的有效电平, 以及允许/禁止滤波功能

参 数: axis 指定轴号
 org_logic ORG 信号的有效电平: 0—低电平有效, 1—高电平有效
 filter 保留

返回值: 错误代码

DWORD d5480_write_SEVON_PIN(WORD axis, WORD on_off)

功 能: 输出对指定轴的伺服使能端子的控制

参 数: axis 指定轴号
 on_off 设定管脚电平状态: 0—低, 1—高。注意: 其中 SEVON 输出口初始状态可选, 当拨码开关对应的位设置为 OFF 后, 则 SEVON 的电平逻辑会取反。

返回值: 错误代码

DWORD d5480_set_backlash(WORD axis, long backlash)

DWORD d5480_get_backlash(WORD axis, long* backlash)

功 能: 设置/读取间隙补偿值

参 数: axis 指定轴号
 backlash 间隙补偿值, 单位: 脉冲

返回值: 错误代码

int d5480_read_RDY_PIN(WORD axis)

功 能: 读取指定运动轴的“伺服准备好”端子的电平状态

参 数: axis 指定轴号

返回值: 0—低电平, 1—高电平

DWORD d5480_write_ERC_PIN(WORD axis, WORD sel)

功 能: 控制指定轴“误差清除”端子信号的输出

参 数: axis 指定轴号

sel 0—复位 ERC 信号, 1—软件强制输出 ERC 信号

返回值: 错误代码

DWORD d5480_config_EMG_PIN(WORD cardno, WORD enable,WORD emg_logic)
DWORD d5480_get_config_EMG_PIN(WORD cardno, WORD* enable, WORD* emg_logic);

功 能: EMG 信号设置, 急停信号有效后会立即停止所有轴。

参 数: cardno 卡号

enable 保留

emg_logic: 0: :低有效; 1:高有效

返回值: 错误代码

WORD d5480_axis_io_status(WORD axis)

功 能: 读取指定轴有关运动信号的状态, 包含指定轴的专用 I/O 状态。

参 数: axis 指定轴号

返回值: 见表

位号	信号名称	描述
0	ALM	ALM 信号
1	EL+	EL+信号
2	EL-	EL-信号
3	EMG	EMG 信号
4	HOME	HOME 信号
5	SD	减速信号 (注: 只能作通用输入口使用)
6	SL+	软限位信号, 最大值
7	SL-	软限位信号, 最小值
8	INP	INP 信号
9	HANDA	手轮信号, A 相。
10	HANDB	手轮信号, B 相
11	EA	编码器信号, A 相
12	EB	编码器信号, B 相

其它位	保留	
-----	----	--

8.2.2.8 手轮运动控制函数

DWORD d5480_set_handwheel_inmode (WORD axis,WORD inmode, double multi)

DWORD d5480_get_handwheel_inmode(WORD axis, WORD * inmode, double* multi)

功 能：设置输入手轮脉冲信号的计数方式

参 数：axis 指定轴号

inmode 表示输入方式：0—A、B 相位正交计数，1—双脉冲信号

multi 计数器的计数方向及倍率设置：设置手轮的倍率，>=0 表示默认方向，<0 表示与默认方向相反。

返回值：无

DWORD d5480_handwheel_move(WORD axis)

功 能：启动指定轴的手轮脉冲运动

参 数：axis 指定轴号

返回值：无

8.2.2.9 通用IO控制函数

int d5480_read_inbit(WORD cardno, WORD bitno)

功 能：读取指定控制卡的某一位输入口的电平状态

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）

bitno 指定输入口位号（取值范围：1—36）

通用输入： 1 ~ 20

Rdy: 21 ~ 24

Sd: 25 ~ 28

Inp: 29 ~ 32

Alarm: 33 ~ 36

返回值：0 表示低电平；1 表示高电平

DWORD d5480_write_outbit (WORD cardno, WORD bitno,WORD on_off)

功 能：对指定控制卡的某一位输出口置位

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）

bitno 指定输出口位号（取值范围：1—32）

通用输出口： 1 ~ 20

CMP: 21 ~ 24

SERVON: 25 ~ 28

ERC: 29 ~ 32

on_off 输出电平：0—表示输出低电平，1—表示输出高电平。
 注意：其中输出口初始状态可选，当拨码开关 S2 对应的位设置为 OFF 后，则输出口的电平逻辑会取反。

返回值：错误代码

int d5480_read_outbit(WORD cardno, WORD bitno)

功 能：读取指定控制卡的某一位输出口的电平状态

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）
 bitno 指定输入口位号（取值范围：1—32）

返回值：0 表示低电平；1 表示高电平。注意：其中输出口初始状态可选，当拨码开关 S2 对应的位设置为 OFF 后，则读取的输出口电平逻辑会取反。

long d5480_read_inport(WORD cardno)

功 能：读取指定控制卡的全部通用输入口的电平状态

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）

返回值：bit0 – bit31 位值分别代表第 1 – 32 号输入端口值。

long d5480_read_outport(WORD cardno)

功 能：读取指定控制卡的全部通用输出口的电平状态

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）

返回值：bit0 – bit31 位值分别代表第 1 – 32 号输出端口值。注意：其中输出口初始状态可选，当拨码开关 S2 对应的位设置为 OFF 后，则输出口相应位电平逻辑会取反。

DWORD d5480_write_outport(WORD cardno, DWORD port_value)

功 能：指定控制卡的全部通用输出口的电平状态

参 数：cardno 指定控制卡号，范围（0 — N - 1 ,N 为卡数）
 port_value bit0 – bit31 位值分别代表第 1 – 32 号输出端口值。注意：其中输出口初始状态可选，当拨码开关 S2 对应的位设置为 OFF 后，则输出口相应位电平逻辑会取反。

返回值：错误代码

8.2.3 运动函数错误码说明

错误码	名称	含义
0	ERR_NOERR	成功
1	ERR_UNKNOWN	未知错误
2	ERR_PARAERR	参数错误
3	ERR_TIMEOUT	操作超时
4	ERR_CONTROLLERBUSY	控制卡状态忙
6	ERR_CONTILINE	连续插补出错
10	ERR_SENDErr	数据传输错误，请检查 PCI 槽位是否松动

错误码	名称	含义
20	ERR_FIRMWARE_INVALID_PARA	控制器返回参数错误
22	ERR_FIRMWARE_STATE_ERR	控制器返回当前状态不允许操作
24	ERR_FIRMWARE_CARD_NOT_SUPPORT	控制器不支持的功能

8.3 常见问题库

出现问题	解决建议
板卡插上后，PC 机系统还不能识别控制卡	检查板卡驱动是否正确安装，在WINDOWS 的设备管理器（可参看WINDOWS 帮助文件）中查看驱动程序安装是否正常。如果有相关的黄色感叹号标志，说明安装不正确，需要按照软件部分安装指引，重新安装； 计算机主板兼容性差，请咨询主板供应商； PCI插槽是否完好； PCI 金手指是否有异物，可用酒精清洗。
PC 机不能和控制卡通讯	PCI金手指是否有异物，可用酒精清洗； 参考软件手册检查应用软件是否编写正确。
板卡和驱动器电机连接后，发出脉冲时，电机不转动	板卡上的设置脉冲发送方式和驱动器的输入脉冲方式是否匹配，跳线J1—J8是否正确； 可以用 Motion5480演示软件进行测试，观察脉冲计数等是否正常； 是否已经接上供给脉冲和方向的外部电源。
控制卡已经正常工作，正常发出脉冲，但电机不转动	检查驱动器和电机之间的连接是否正确。可以使用 Motion5480 演示软件进行测试。 确保驱动器工作正常，没有出现报警。
电机可以转动，但工作不正常	检查控制卡和驱动器是否正确接地，抗干扰措施是否做好； 脉冲和方向信号输出端光电隔离电路中使用的限流电阻过大，工作电流偏小。
能够控制电机，但电机出现振荡或是过冲	可能是驱动器参数设置不当，检查驱动器参数设置； 应用软件中加减速时间和运动速度设置不合理。
能够控制电机，但工作时，回原点定位不准	检查屏蔽线是否接地； 原点信号开关是否工作正常； 所有编码信号和原点信号是否受到干扰。
限位信号不起作用	限位传感器工作不正常； 限位传感器信号受干扰； 应用程序紊乱。
不能读入编码器信号	请检查编码器信号类型是否是脉冲TTL方波； 参看所选编码器说明书，检查接线是否正确； 编码器供电是否正常； 检查函数调用是否正确。
对编码器的读数不准	检查全部编码器及触发源的接线；

出现问题	解决建议
确	做好信号线的接地屏蔽。
不能锁存编码器读数	检查触发源的接线； 检查函数的调用是否正确。
锁存数据的重复精度差	检查函数调用； 程序中是否进行了去抖动处理； 触发信号的设定。
数字输入信号不能读取	接线是否正常； 检查函数调用。
数字输出信号不正常	接线是否正常； 检查函数调用。

8.4 ACC2410 接线盒端子定义

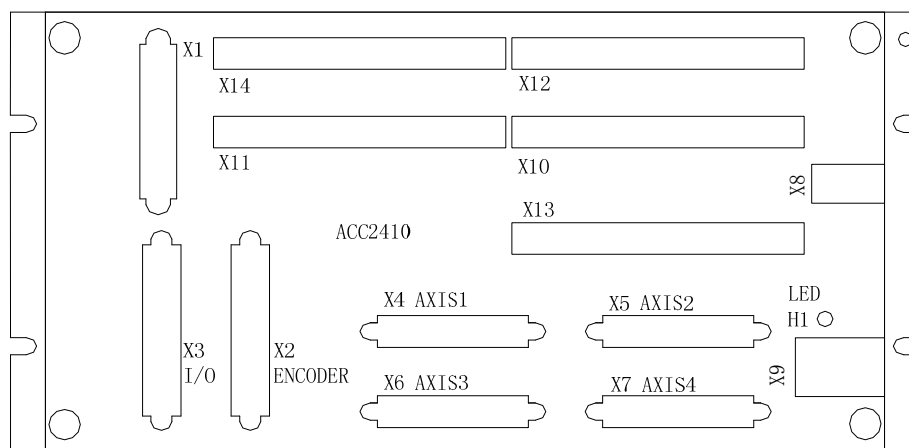


图 8-6 ACC2410 接线盒示意图

X9 电源接口

1	VDD	I	外部电源, +12V~+24V
2	EGND		外部电源地

X8 急停接口

1	EMG	I	急停
2	EGND		外部电源地

轴1接口定义表：X4

序号	名称	I/O	说明	序号	名称	I/O	说明
1	EGND		外部电源地	14	VDD		12V/24V外部电源
2	ALM1	I	第一轴报警信号	15	ERC1	0	误差清零
3	SEVON1	0	第一轴伺服使能信号	16	INP1	I	第一轴位置到信号
4	EA1-	I	第一轴编码器A相(-)	17	EA1+	I	第一轴编码器A相(+)

序号	名称	I/O	说明	序号	名称	I/O	说明
5	EB1-	I	第一轴编码器B相(-)	18	EB1+	I	第一轴编码器B相(+)
6	EZ1-	I	第一轴编码器Z相(-)	19	EZ1+	I	第一轴编码器Z相(+)
7	5V		PC5V电源	20	GND		PC电源地
8	RDY1	I	第一轴准备好输入信号	21	GND		PC电源地
9	DIR1+	0	第一轴方向信号(+)	22	DIR1-	0	第一轴方向信号(-)
10	GND		PC电源地	23	PUL1+	0	第一轴脉冲信号(+)
11	PUL1-	0	第一轴脉冲信号(-)	24	GND		PC电源地
12	LTC1+	I	第一轴锁存输入信号(+)	25	LTC1-	I	第一轴锁存输入信号(-)
13	GND		PC电源地				

Y轴接口, Z轴接口, U轴接口 定义同上

轴2接口定义表: X5

序号	名称	I/O	说明	序号	名称	I/O	说明
1	EGND	0	外部电源地	14	VDD		12V/24V外部电源
2	ALM2	I	第二轴报警信号	15	ERC2	0	误差清零
3	SEVON2	0	第二轴伺服使能信号	16	INP2	I	第二轴位置到信号
4	EA2-	I	第二轴编码器A相(-)	17	EA2+	I	第二轴编码器A相(+)
5	EB2-	I	第二轴编码器B相(-)	18	EB2+	I	第二轴编码器B相(+)
6	EZ2-	I	第二轴编码器Z相(-)	19	EZ2+	I	第二轴编码器Z相(+)
7	5V		PC5V电源	20	GND		PC电源地
8	RDY2	I*	第二轴准备好输入信号	21	GND		PC电源地
9	DIR2+	0	第二轴方向信号(+)	22	DIR2-	0	第二轴方向信号(-)
10	GND		PC电源地	23	PUL2+	0	第二轴脉冲信号(+)
11	PUL2-	0	第二轴脉冲信号(-)	24	GND		PC电源地
12	LTC2+	I	第二轴锁存输入信号(+)	25	LTC2-	I	第二轴锁存输入信号(-)
13	GND		PC电源地				

轴3接口定义表: X6

序号	名称	I/O	说明	序号	名称	I/O	说明
1	EGND	0	外部电源地	14	VDD		12V/24V外部电源
2	ALM3	I	第三轴报警信号	15	ERC3	0	误差清零
3	SEVON3	0	第三轴伺服使能信号	16	INP3	I	第三轴位置到信号
4	EA3-	I	第三轴编码器A相(-)	17	EA3+	I	第三轴编码器A相(+)
5	EB3-	I	第三轴编码器B相(-)	18	EB3+	I	第三轴编码器B相(+)
6	EZ3-	I	第三轴编码器Z相(-)	19	EZ3+	I	第三轴编码器Z相(+)

序号	名称	I/O	说明	序号	名称	I/O	说明
7	5V		PC5V电源	20	GND		PC电源地
8	RDY3	I	第三轴准备好输入信号	21	GND		PC电源地
9	DIR3+	0	第三轴方向信号(+)	22	DIR3-	0	第三轴方向信号(-)
10	GND		PC电源地	23	PUL3+	0	第三轴脉冲信号(+)
11	PUL3-	0	第三轴脉冲信号(-)	24	GND		PC电源地
12	LTC3+	I	第三轴锁存输入信号(+)	25	LTC3-	I	第三轴锁存输入信号(-)
13	GND		PC电源地				

轴4接口定义表: X7

序号	名称	I/O	说明	序号	名称	I/O	说明
1	EGND	0	外部电源地	14	VDD		12V/24V外部电源
2	ALM4	I	第四轴报警信号	15	ERC4	0	误差清零
3	SEVON4	0	第四轴伺服使能信号	16	INP4	I	第四轴位置到信号
4	EA4-	I	第四轴编码器A相(-)	17	EA4+	I	第四轴编码器A相(+)
5	EB4-	I	第四轴编码器B相(-)	18	EB4+	I	第四轴编码器B相(+)
6	EZ4-	I	第四轴编码器Z相(-)	19	EZ4+	I	第四轴编码器Z相(+)
7	5V		PC5V电源	20	GND		PC电源地
8	RDY4	I*	第四轴准备好输入信号	21	GND		PC电源地
9	DIR4+	0	第四轴方向信号(+)	22	DIR4-	0	第四轴方向信号(-)
10	GND		PC电源地	23	PUL4+	0	第四轴脉冲信号(+)
11	PUL4-	0	第四轴脉冲信号(-)	24	GND		PC电源地
12	LTC4+	I	第四轴锁存输入信号(+)	25	LTC4-	I	第四轴锁存输入信号(-)
13	GND		PC电源地				

专用信号输入口 X14

1	ORG1	I	1轴原点信号(+)
2	ORG2	I	2轴原点信号(+)
3	ORG3	I	3轴原点信号(+)
4	ORG4	I	4轴原点信号(+)
5	EL1+	I	1轴端点限位信号(+)
6	EL1-	I	1轴端点限位信号(-)
7	EL2+	I	2轴端点限位信号(+)
8	EL2-	I	2轴端点限位信号(-)
9	EL3+	I	3轴端点限位信号(+)
10	EL3-	I	3轴端点限位信号(-)
11	EL4+	I	4轴端点限位信号(+)

12	EL4-	I	4轴端点限位信号(-)
13	EGND		外部电源地
14	EGND		外部电源地
15	EGND		外部电源地
16	VDD		12V/24V外部电源

通用输入口 X10

1	IN1	I	通用输入1
2	IN2	I	通用输入2
3	IN3	I	通用输入3
4	IN4	I	通用输入4
5	IN5	I	通用输入5
6	IN6	I	通用输入6
7	IN7	I	通用输入7
8	IN8	I	通用输入8
9	IN9	I	通用输入9
10	IN10	I	通用输入10
11	IN25		通用输入25
12	IN26		通用输入26
13	EGND		外部电源地
14	EGND		外部电源地
15	EGND		外部电源地
16	VDD		12V/24V外部电源

通用输入口 X11

1	IN11	I	通用输入11
2	IN12	I	通用输入12
3	IN13	I	通用输入13
4	IN14	I	通用输入14
5	IN15	I	通用输入15
6	IN16	I	通用输入16
7	IN17	I	通用输入17
8	IN18	I	通用输入18
9	IN19	I	通用输入19
10	IN20	I	通用输入20
11	IN27		通用输入27
12	IN28		通用输入28
13	EGND		外部电源地
14	EGND		外部电源地
15	EGND		外部电源地
16	VDD		12V/24V外部电源

通用输出口 X12

1	OUT1	0	通用输出1
2	OUT2	0	通用输出2
3	OUT3	0	通用输出3
4	OUT4	0	通用输出4
5	OUT5	0	通用输出5
6	OUT6	0	通用输出6
7	OUT7	0	通用输出7
8	OUT8	0	通用输出8
9	OUT9	0	通用输出9
10	OUT10	0	通用输出10
11	CMP1	0	1轴比较输出信号
12	CMP2	0	2轴比较输出信号
13	LED	0	探针发光二极管
14	EGND		PC电源地
15	EGND		PC电源地
16	5V		PC5V电源

通用输出口 X13

1	OUT11	0	通用输出11
2	OUT12	0	通用输出12
3	OUT13	0	通用输出13
4	OUT14	0	通用输出14
5	OUT15	0	通用输出15
6	OUT16	0	通用输出16
7	OUT17	0	通用输出17
8	OUT18	0	通用输出18
9	OUT19	0	通用输出19
10	OUT20	0	通用输出20
11	CMP3	0	3轴比较输出信号
12	CMP4	0	4轴比较输出信号
13	BUF	0	蜂鸣器
14	EGND		PC电源地
15	EGND		PC电源地
16	5V		PC5V电源