



PAC 实时内核用户手册

大连理工计算机控制工程有限公司

2013 年 06 月 28 日

前言

感谢您使用 PAC 实时内核软件，本手册提供了软件必要的使用信息。为了确保能够正确使用产品，请认真阅读手册。

手册内容已经过严格确认，与所描述的软硬件信息相符合。由于遗漏或疏忽可能造成某些地方的错误，发现后请及时反馈给我们，我们会在后续版本中进行更正。

安全指南

手册包括应该遵守的注意事项，以保证人身安全，保护产品和所连接的设备免受损坏。注意事项分为四类：“危险”、“警告”、“注意”和“提示”。

危险：表示不正确的操作将导致危险情况发生，造成严重的人身伤害；

警告：表示不正确的操作将导致危险情况发生，造成中度或轻微的人身伤害；

注意：表示不正确的操作将导致产品无法正常工作，造成该部分功能无法实现或设备损坏；

提示：表示使用的操作方法能使 PAC 实时内核获得更好的性能或是对部分说明更详尽的补充。

手册用途

手册提供了 PAC 实时内核的安装方法、工作原理、操作接口、产品功能等帮助内容。根据手册说明操作，可减少因使用过程不当而造成的损失。若需要仔细了解 PAC 实时内核的编程方法，请参考与产品配套的编程手册《DCCE 网络化可编程控制器用户编程手册》和《DCCE 编程手册-运动控制指令篇》。

相关专有名词

- AIO: Analog Input Output, 模拟量信号输入输出
- DIO: Digital Input Output, 数字量信号输入输出
- DCCE: DUT Computer Control Engineering Co.LTD, 大连理工计算机控制工程有限公司
- EPA: Ethernet for Plant Automation, 是中国制定的第一个现场总线标准，被列为国际现场总线标准 IEC61158 第 14 类型
- FBD: Function Block Diagram, 功能块图表，基于 IEC61131-3 标准的一种编程语言
- LAD: Ladder Diagram, 梯形图表，基于 IEC61131-3 标准的一种编程语言
- IEC: International Electronic Committee, 国际电工委员会

- MB+: Modbus Plus, 指在原有 RS485 总线基础上通过特有技术实现的一种速率可达 1M bits/s 的总线
- PAC: Programmable Automation Controller, 可编程自动化控制器
- PEC: Programmable Ethernet Controller, 可编程以太网控制器
- PLC_Config: 可编程控制器编程软件, DCCE 公司编写

目录

前言	1
目录	3
1 概述	4
2 工作原理	5
3 软件安装	5
3.1 安装要求	6
3.2 安装步骤	12
4 实时内核功能介绍	14
4.1 启动实时内核	15
4.2 设备资源	17
4.2.1 内部资源	17
4.2.2 I/O 设备	19
4.2.3 定时器	21
4.2.4 中断源	22
4.2.5 可用参数	23
4.3 可编程控制	23
4.3.1 编程软件	23
4.3.2 变量	30
4.3.3 指令	33
4.3.4 程序	44
4.3.5 编程	46
4.4 网络通信服务	52
4.4.1 以太网自由通信	52
4.4.2 EPA 协议通信	53
4.4.3 ModbusTCP/ModbusUDP 通信	54
4.5 系统管理	54
4.5.1 PAC 控制系统	54
4.5.2 指令网络通信	55

1 概述

PAC 实时内核软件绑定在由我公司研发的一款高性能工业级平板电脑中,该款平板电脑基于英特尔 Luna Pier 平台,提供高达 1.8GHz 的双核芯处理速度、15 吋（可选）四线电阻式触摸屏、3 个 10/100/1000M 自适应以太网接口、1 路 232、2 路 485、6 个 USB2.0 接口,可应用于高速度、大容量、高稳定、低功耗的大型工业现场应用中。

iTC1015 带有我公司研发的实时内核软件程序,该程序具有逻辑控制、过程控制、运动控制等特点。支持 IEC61131-3 标准的梯形图、功能块编程,具有逻辑指令,运算指令,定时/计数器指令,控制指令,中断指令,网络通信指令,专用指令,以及符合 PLCOpen 标准的单轴指令,多轴指令,轴组指令等。支持 EPA、Modbus 等主流控制网络协议。以网络系统为整体,对系统中的设备进行统一管理和编程开发,可以实现大中小型可编程控制系统的各种功能,具有特点如下:

- 独立内核运行, iTC1015 独占一个 CPU 核芯, 高达 1.8GHz 的主频, 将用户指令的执行时间提高到百纳秒级, 保证了用户程序执行的实时性和高效性。

- Windows 操作系统同 PAC 内核并行运行, 互不影响, 即使 Windows 发生故障蓝屏等严重问题时, 也不会影响 PAC 实时内核的执行。

- Windows 与实时内核网络分离, iTC1015 提供 3 个以太网接口, 其中 1 路由 Windows 使用, 负责与外部网络连接; 另外 2 路由实时内核使用, 负责组建 IO 设备网络, 两个网络间互不影响, 即使 Windows 发生故障蓝屏等严重问题时, 也不会影响 PAC 实时内核的与 IO 设备间的正常通信。

- Windows 为用户提供人机界面接口, 提供与外部网络互联的以太网接口, 确保 PAC 与外部网络的数据交换, 内核独占。

- PAC 内核占用 2 个千兆以太网通过 EtherCAT 网络同 IO 设备进行数据交互; 同时占用 2 个 RS-485 接口, 当 EtherCAT 网络通信出错时, PAC 内核可以通过 485 接口与 IO 设备进行数据交互, 获得出错信息并告知用户, EtherCAT 网络同 MB+ 协议完美结合, 保证了独立内核运行的安全性。

- 提供了梯形图和功能块等多种编程语言, 带有自整定功能的高精度 PID 控制指令和工件检测、经济运行、数据校验等过程自动化和工厂自动化的控制指令, 控制性能好、精度高、使用方便。

- 提供了符合 PLCOpen 标准的单轴、多轴、轴组三类运动控制指令集, 支持空间直线、圆弧、NURBS 曲线插补等多种插补算法, 支持梯型曲线, S 曲线, E 指数曲线等多种加减速控制曲线, 支持直角模型、圆柱模型、四轴并联模型、三轴并联模型、三轴座椅模型、通用模型等多种动力学模型。

2 工作原理

PAC 工业平板电脑使用的 CPU 型号为 Intel 公司生产的 Atom 系列 D525CPU, 拥有两个独立的物理核芯, 实时内核软件在运行时抢占了一个完整的 CPU 核芯用来运行用户程序和 IO 设备管理工作, 保证了实时性, 并且不影响另一块 CPU 核芯运行 Windows 操作系统。

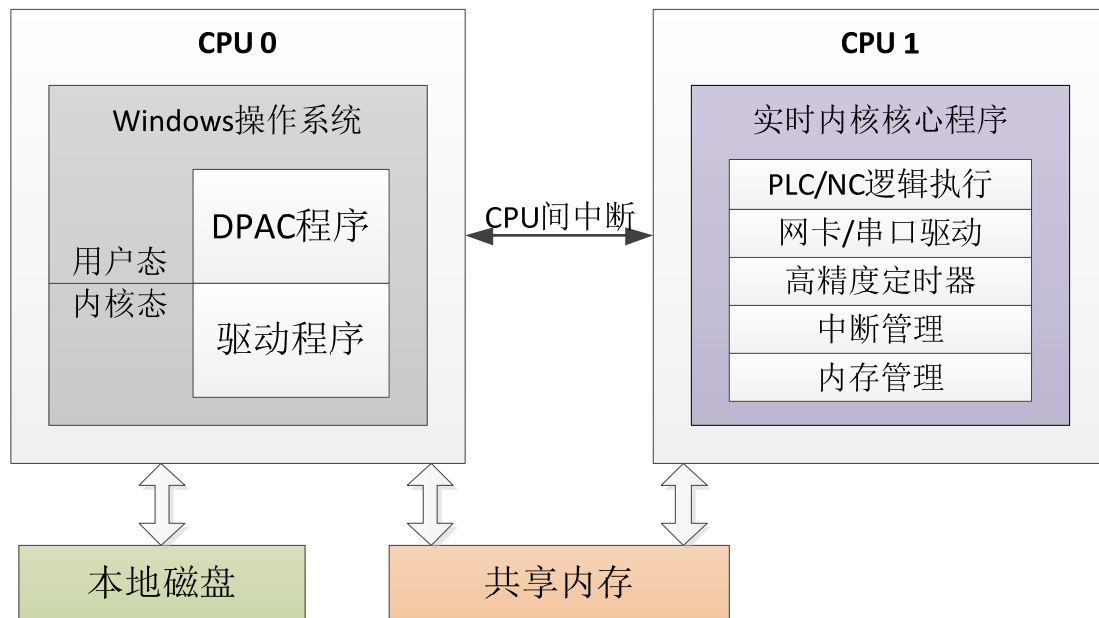


图 2-1 PAC 实时内核原理图

Windows 安装 PAC 实时内核程序后, 不但为实时内核提供一个独立的 CPU 核芯, 同时也会划分出独立的 256MB 内存, 为实时内核运行提供充足的空间, 整个过程随程序的安装自动完成, 无需用户操作。

DPAC 实时内核在 Windows 操作系统一侧, 为用户提供了操作接口软件 DPAC, 用户可通过该软件启动、停止实时内核并可监控实时内核的运行状态。DPAC 同时还为编程软件 PLC_Config 提供通信接口, 实现程序的上下载、变量访问服务、ModbusTCP 服务等。

内核态驱动程序负责唤醒独立 CPU 核芯, 并完成实时内核的装载和卸载工作, 实现流程随用户操作 DPAC 软件时自动完成。

3 软件安装

我公司提供 PAC 试试内核软件的安装包, 随 iTC1015 出库时已完成该软件的安装, 如果用户需要重新安装, 可登录我公司官方主页 www.dcce.cn, 在产品下载中找到 PAC 实时内核软件安装包, 下载到 PAC 中进行安装。

3.1 安装要求

(1) 硬件要求

PAC 实时内核必须安装在我公司推出的工业级平板电脑 PAC 中。

(2) 操作系统

Windows7-32 位版。

(3) 关闭超线程

Intel 公司生产的 D525 芯片为双核四线程 CPU，为保证实时内核占有的 CPU 核芯为物理核芯，启动 Windows 系统前需要在 BIOS 中关闭超线程。PAC 在出库前已完成此项配置，如果用户需要手动完成关闭超线程操作可参考如下步骤：

- 按下 PAC 设备右下角的电源键，为 PAC 上电。
- 在出现最原始黑色画面时立即点击键盘的“Delete”按键（PAC 需要外接键盘），可按多次，直到进入 BIOS 配置页面。



图 3-1-1 BIOS 主页面

- 进入 BIOS 界面后，按键盘的方向键“→”，进入“Advanced”配置页。

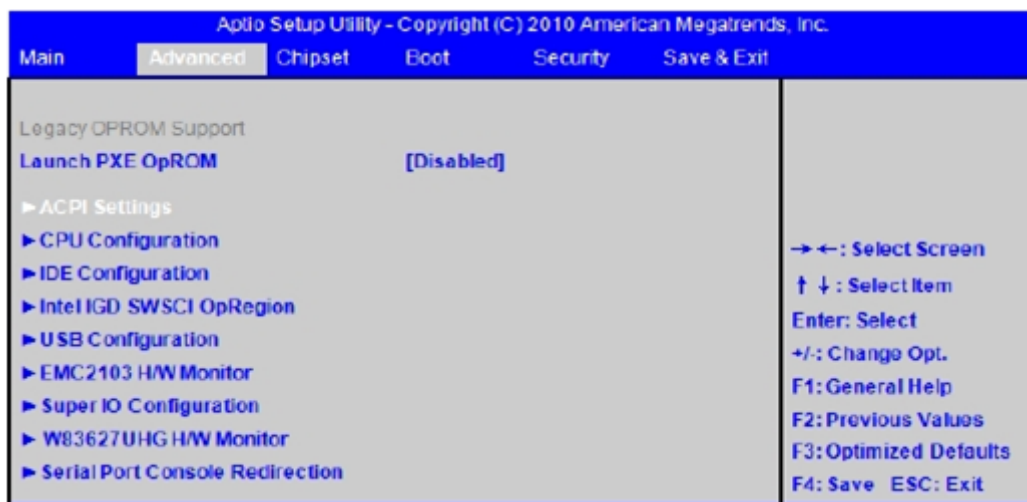


图 3-1-2 BIOS-Advanced 配置页面

- 通过键盘的方向键“↑↓”，选中“CPU Configuration”一项，按下 Enter 键进入到 CPU 配置页面。
- 通过键盘的方向键“↑↓”，选中“Hyper—Threading”一项，按下 Enter 键，在弹出界面中选择 Disabled，再按下 Enter 键完成配置，下面需要保存配置并退出。
- 点击“Esc”键退出 CPU 配置，通过键盘的方向键“→”选择“Save & Exit”一项，选择“Save Changes and Exit”一项，按下 Enter 键选择该项，弹出提示框后选择“Yes”再按下 Enter 键，计算机将重新启动。

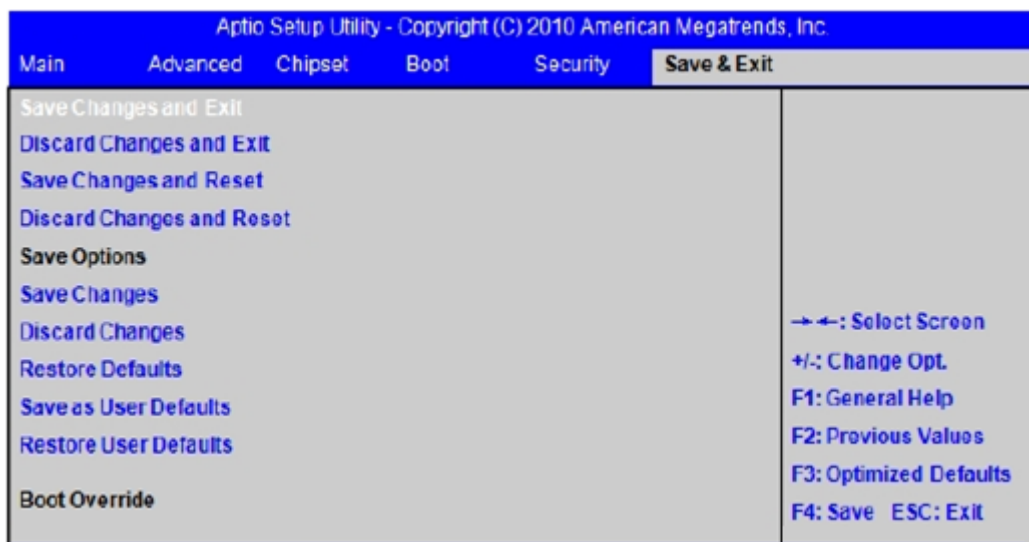


图 3-1-3 BIOS-Save & Exit 页面

(4) 更改 UAC 设置

用户需要在安装实时内核前更改 UAC 设置，从而能够正常运行实时内核程序（该项设置已在 PAC 出库前完成，如果用户需要手动完成此项配置可参考如下介绍）。点击 Window 开始菜单，在“搜索程序和文件”一栏中，输入“msconfig”。

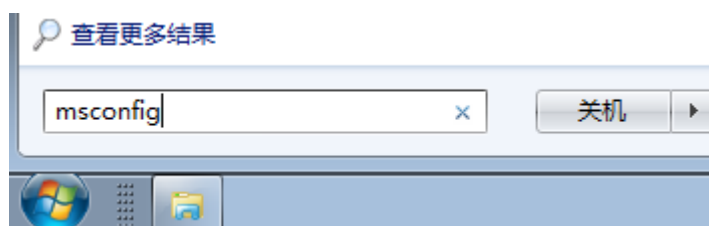


图 3-1-4 启动 Windows 系统配置命令

按下 Enter 键后，进入到 Windows 系统配置页。

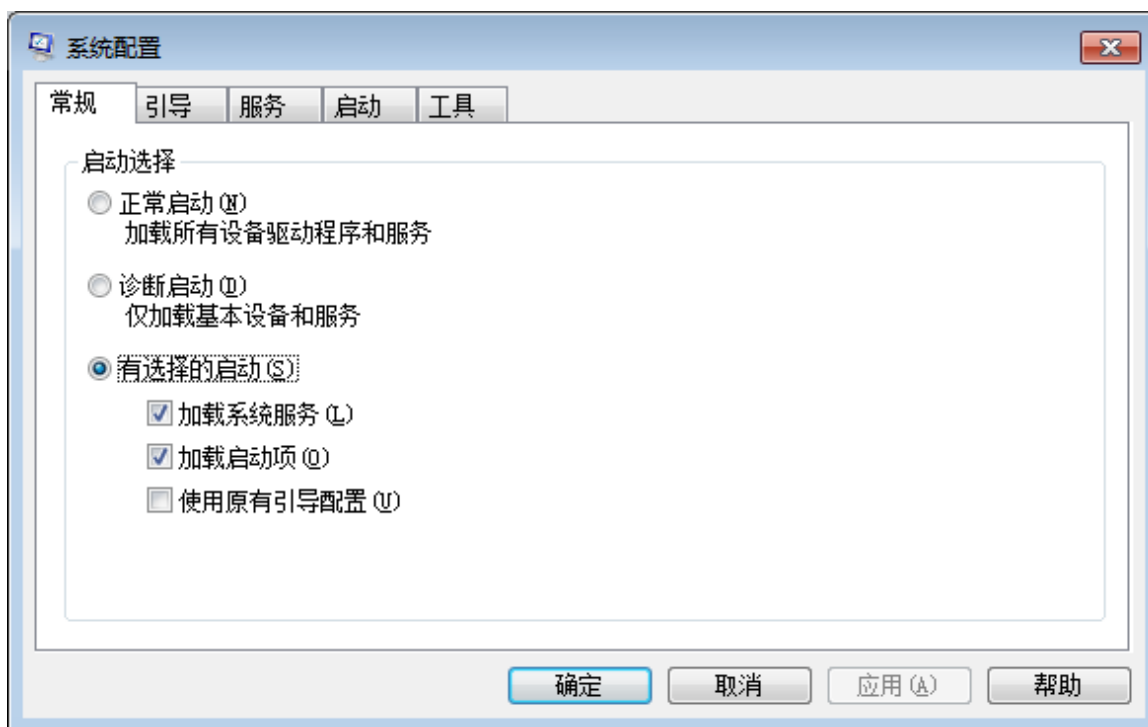


图 3-1-5 系统配置

点击工具一栏。

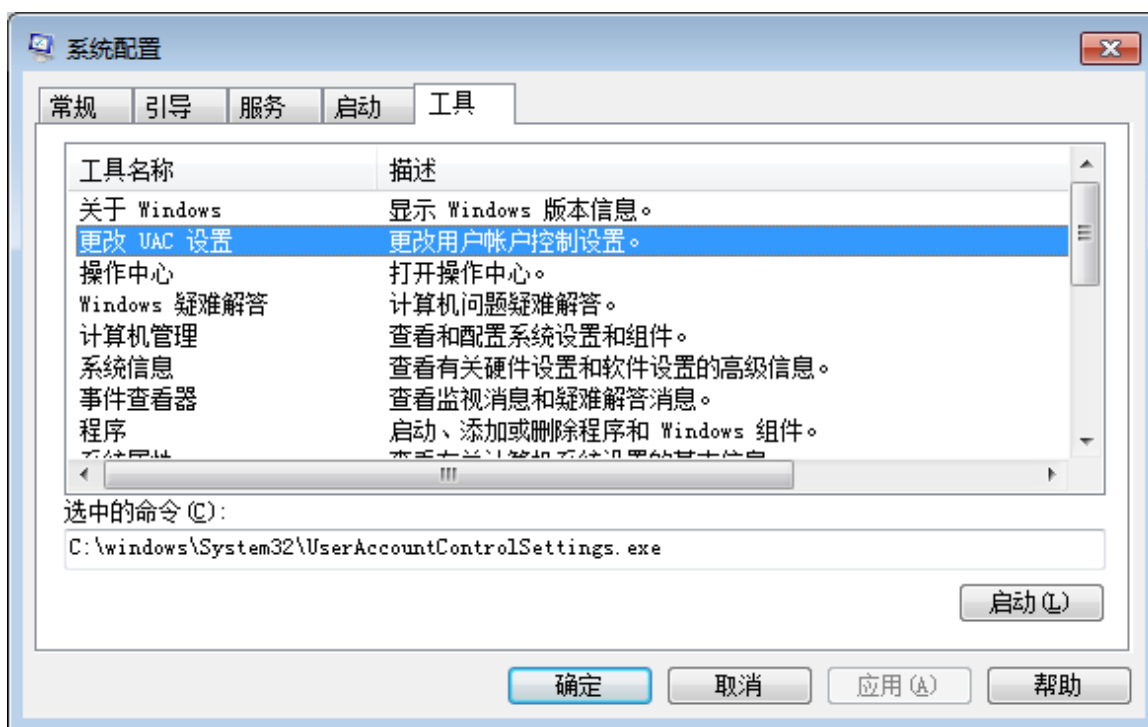


图 3-1-6 UAC 配置项

选择第二行的“更改 UAC 设置”，点击右下角的“启动”按钮。

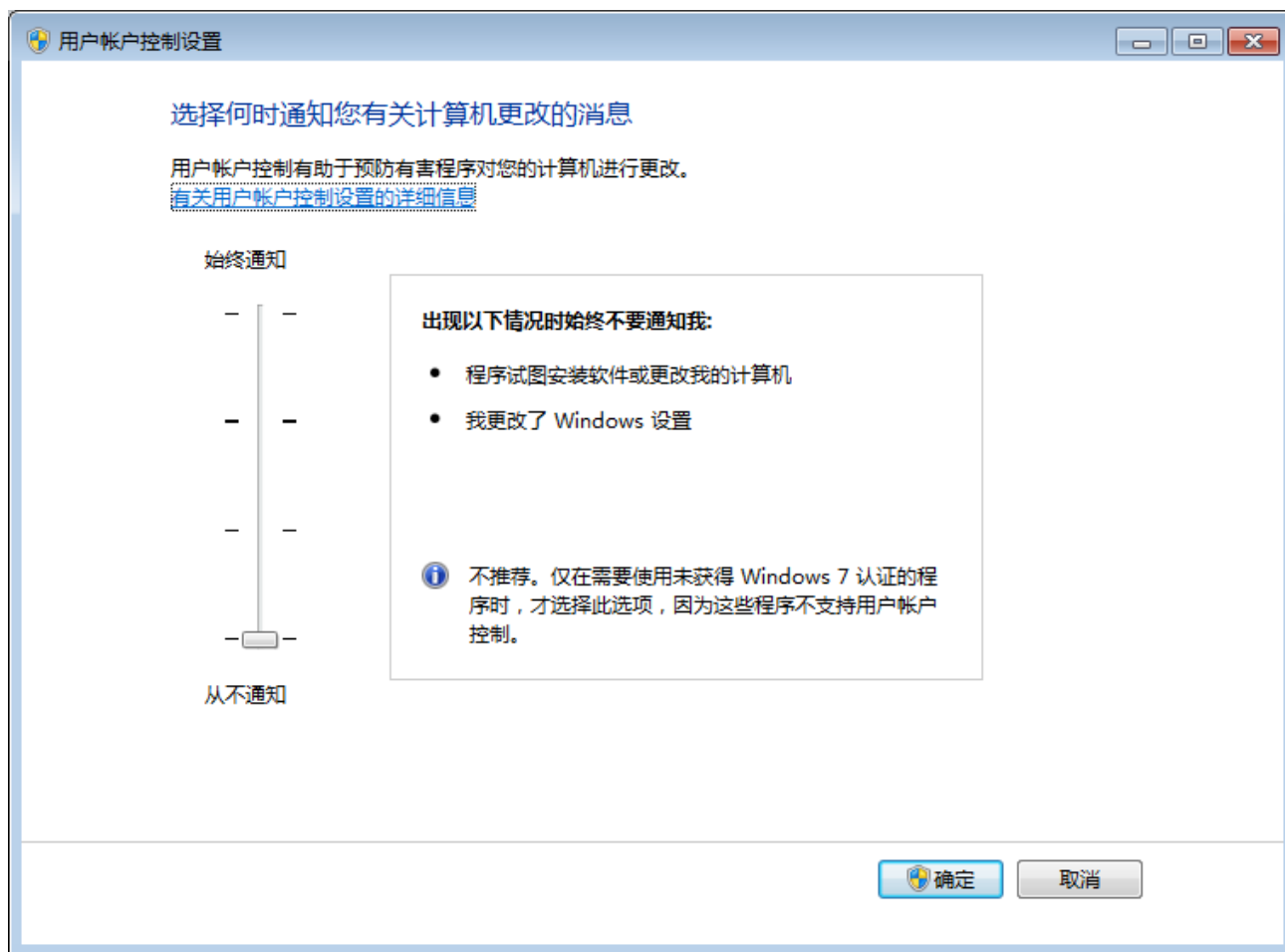


图 3-1-7 UAC 配置页面

将左侧中间的通知配置移动块拖动至最低端，到“从不通知”位置上，点击“确定”后退出。

更改 UAC 配置后，推荐用户重新启动 PAC。

（5）关闭防火墙

为保证通信正常，建议用户关闭 PAC 的防火设置，下面介绍如何关闭 Windows7 操作系统的防火墙配置（该项设置已在 PAC 出库前完成，如果用户需要手动完成此项配置可参考如下介绍）。

通过“控制面板” — “系统和安全” — “Windows 防火墙”进入到 Windows 防火墙的状态显示界面。

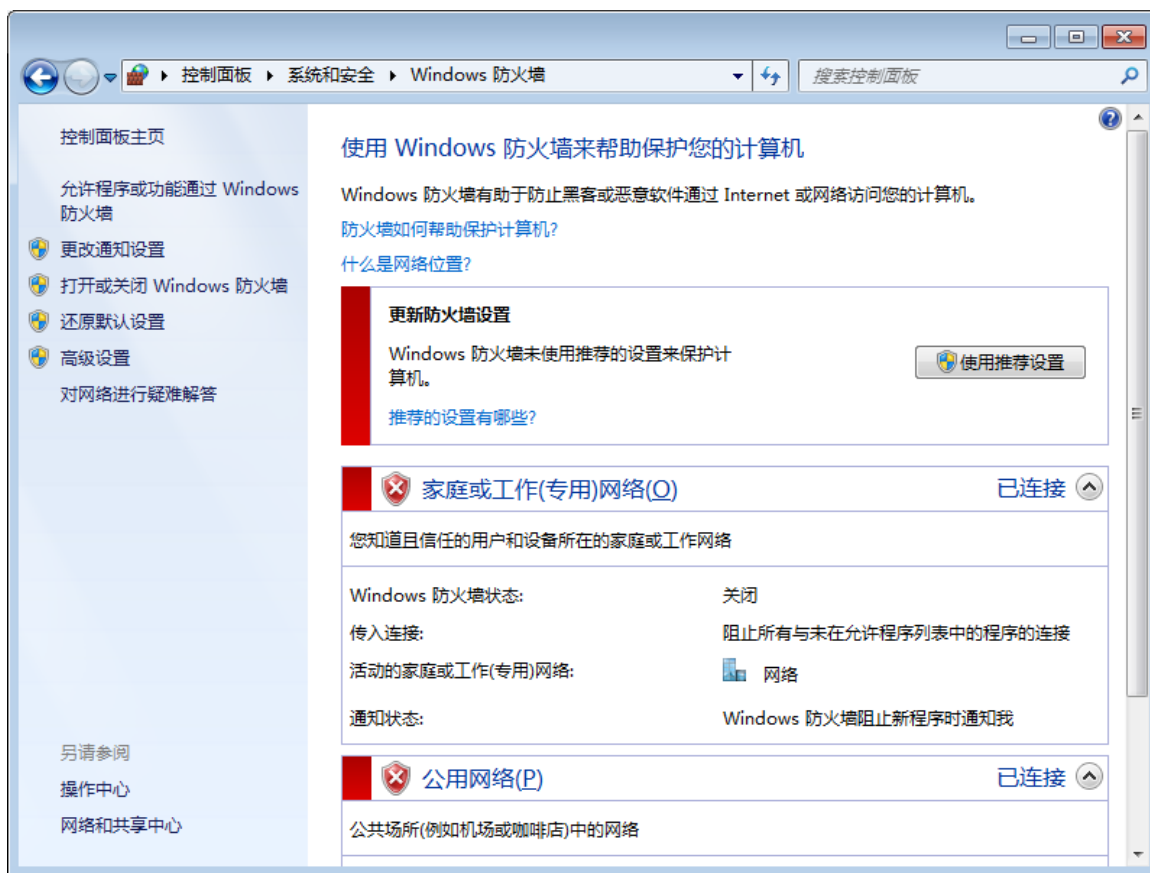


图 3-1-8 防火墙状态

在左侧列表中，选择“打开或关闭 Windows 防火墙”，进入到配置防火墙页面。

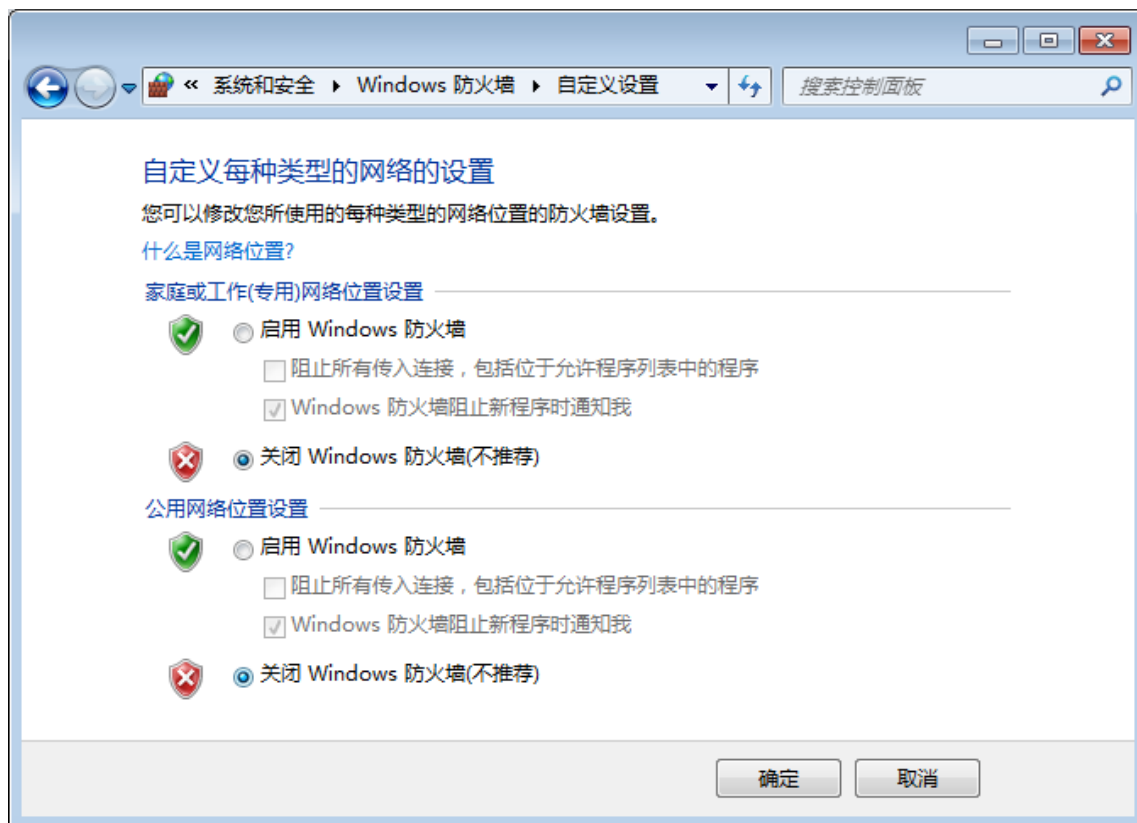


图 3-1-9 防火墙配置

（6）关闭蓝屏重启

PAC 实时内核具有一项安全特性，即在 Windows 发生蓝屏情况下，实时内核仍能够继续运行，不受 Windows 蓝屏影响，外部 IO 设备刷新的过程也不受 Windows 蓝屏影响，以保证系统的安全运行。为此，用户需要在 Windows 系统中手动关闭蓝屏重启功能，以保证不影响实时内核（该项设置已在 PAC 出库前完成，如果用户需要手动完成此项配置可参考如下介绍）。

右键“计算机”，选择“属性”菜单项，系统弹出 Windows 的系统配置主页面。



图 3-1-10 系统基本信息

点击左侧的“高级系统设置”一项，进入到系统属性配置，选择“高级”标签页。

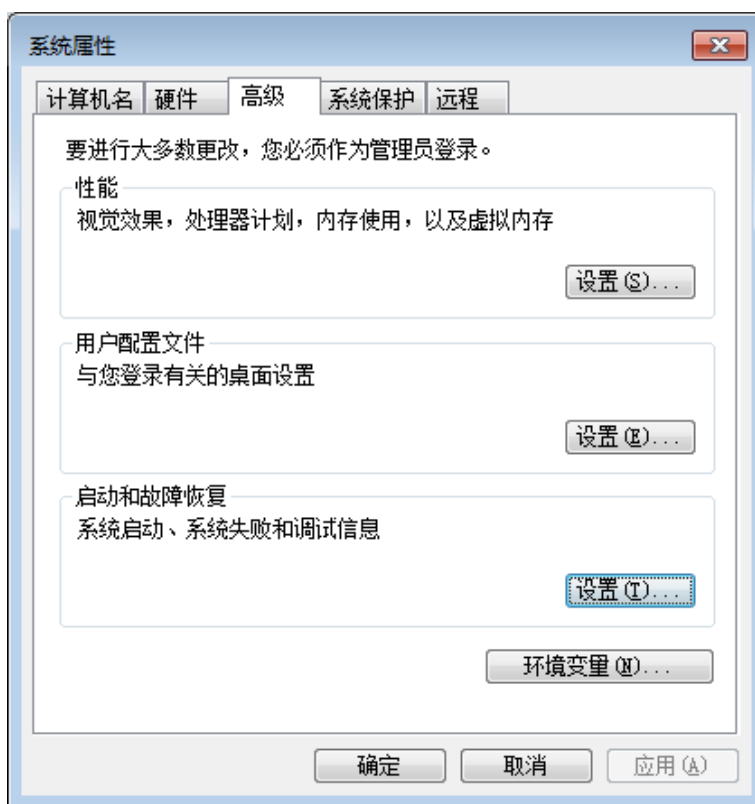


图 3-1-11 Windows 高级设置

点击下面区域“启动和故障恢复”中的“设置”按钮，进入到 Windows 启动和故障恢复

配置页面。

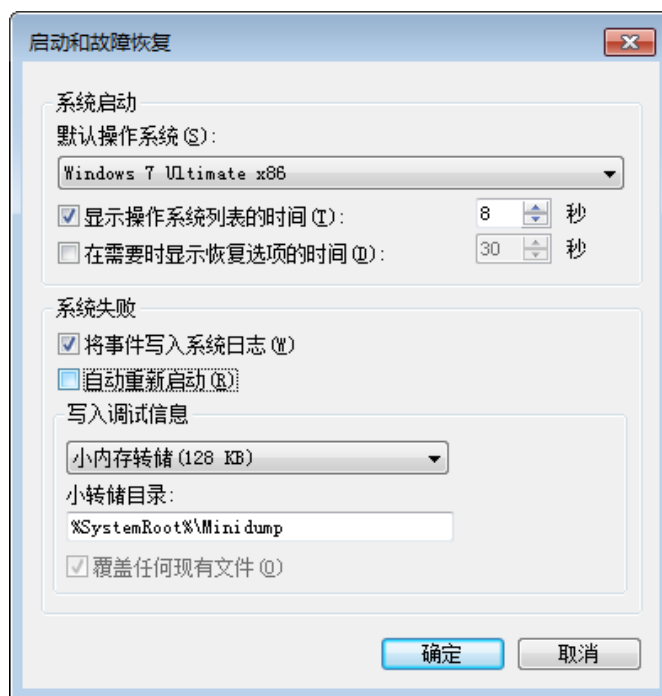


图 3-1-12 启动和故障恢复

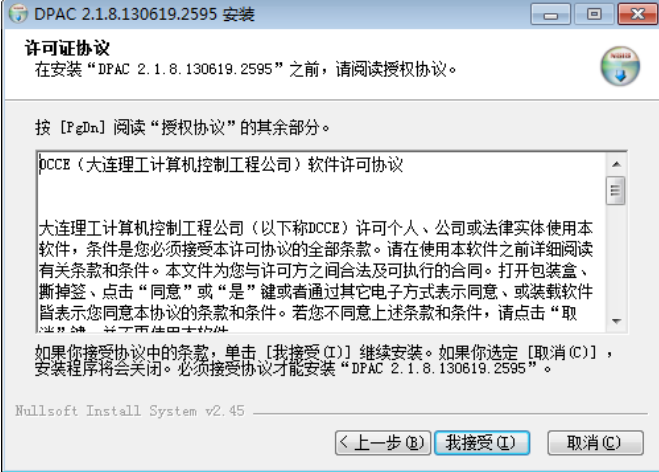
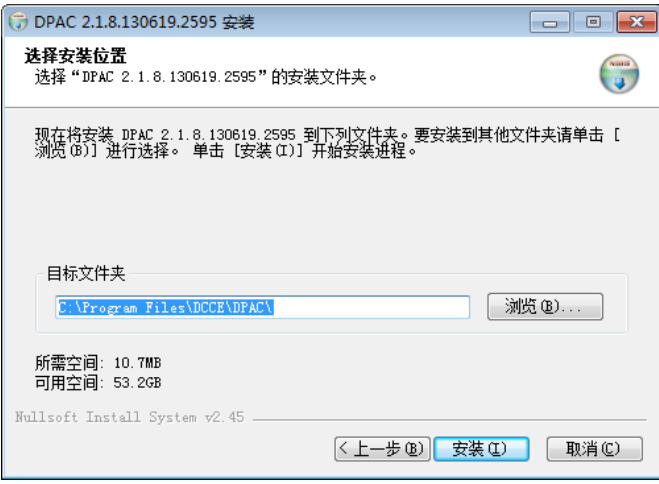
将下方“系统失败”区域中的“自动重新启动”一项设置为不选中状态，点击确定退出，完成关闭 Windows 蓝屏重启的过程。

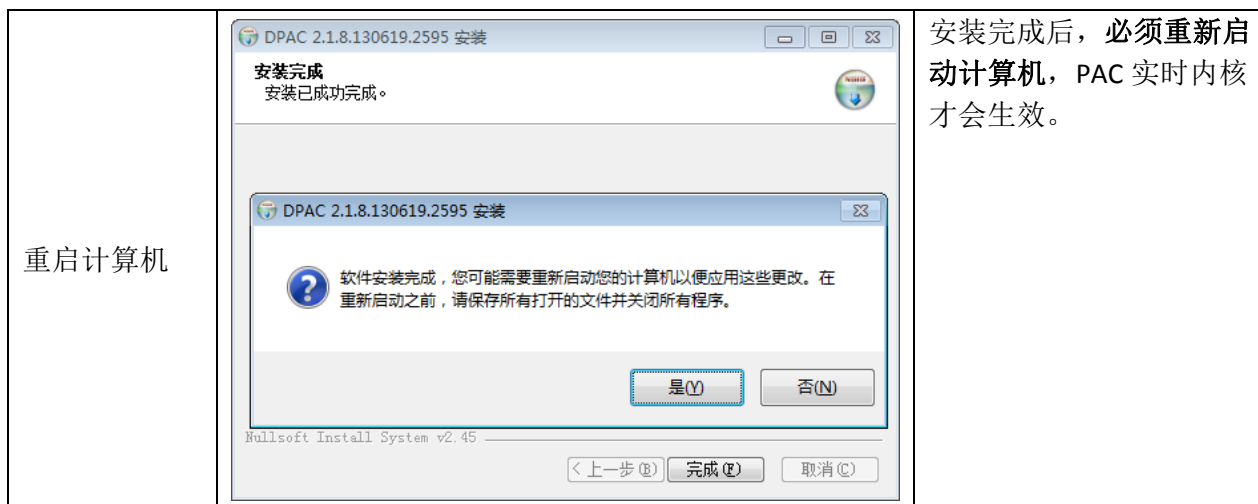
3.2 安装步骤

PAC 实时内核程序的安装文件可在我公司官方主页 www.dcce.cn 的产品列表中找到，或随 PAC 产品发货后，自动保存在 PAC 随机系统的硬盘中（该软件已在 PAC 出库前完成安装，如果用户需要手动安装此软件可参考如下介绍），安装步骤如表 3-2-1 所示。

表 3-2-1 软件安装流程

步骤	内容	说明
启动安装		启动界面，安装前，建议关闭其他所有应用程序，尤其是正在运行 DPAC 程序。 确定无误后点击“下一步”

许可协议		请仔细阅读可协议，如果同意则点击“我接受”
选择安装目录		默认安装在“C:\Program Files\DCCE\DPAC\”目录下，用户可使用“浏览”功能修改安装目录，确认无误后，点击“安装”。安装包将进入安装过程，用户可耐心等待。
安装完成		安装完成后，向导会提示用户，点击“完成”。



用户可在 Windows 系统的菜单中找到 DPAC 实时内核软件的启动文件，和卸载文件。

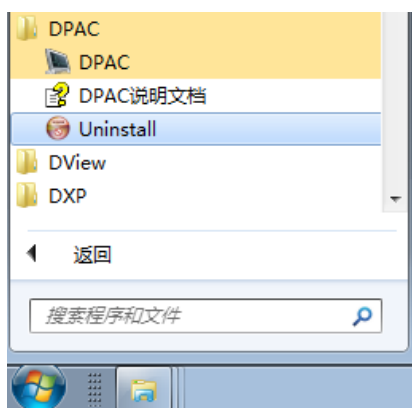


图 4-3-1 Windows 主菜单

卸载前请关闭正在运行 DPAC 程序，双击 Uninstall 文件即可进入到实时内核的卸载流程，按照提示要求可完成卸载过程。

4 实时内核功能介绍

iTC1015 基于英特尔 Luna Pier 平台，提供 1.8GHz 的 CUP 主频，支持 143 条逻辑控制指令，45 条运动控制指令，5M 的用户程序存储空间，支持梯形图、功能块、指令码多种编程语言，可实现过程控制、逻辑控制、运动控制等多项控制功能，为用户提供高效、稳定、安全的程序执行能力。iTC1015 拥有 1 路千兆以太网和 1 路 232 通信接口，支持 EPA、ModbusRTU、ModbusTCP、自由通信等多项协议，可实现外部数据的采集和控制功能；同时提供 2 路 IO 扩展千兆以太网和 2 个 485 通信接口，可通过 EtherCAT 协议连接多达 256 个 IO 设备，支持上万个 IO 点扩展，IO 同步周期可达 1 毫秒/次，满足对于大型工业现场组态的需求。

4.1 启动实时内核

如果实时内核程序已成功安装，用户可在 Windows 系统的桌面或主菜单中找到实时内核



的启动快捷方式，双击该快捷方式，可启动 DPAC。DPAC 提供“启动服务”按钮，用于启动实时内核程序，如图 4-1-1 所示。

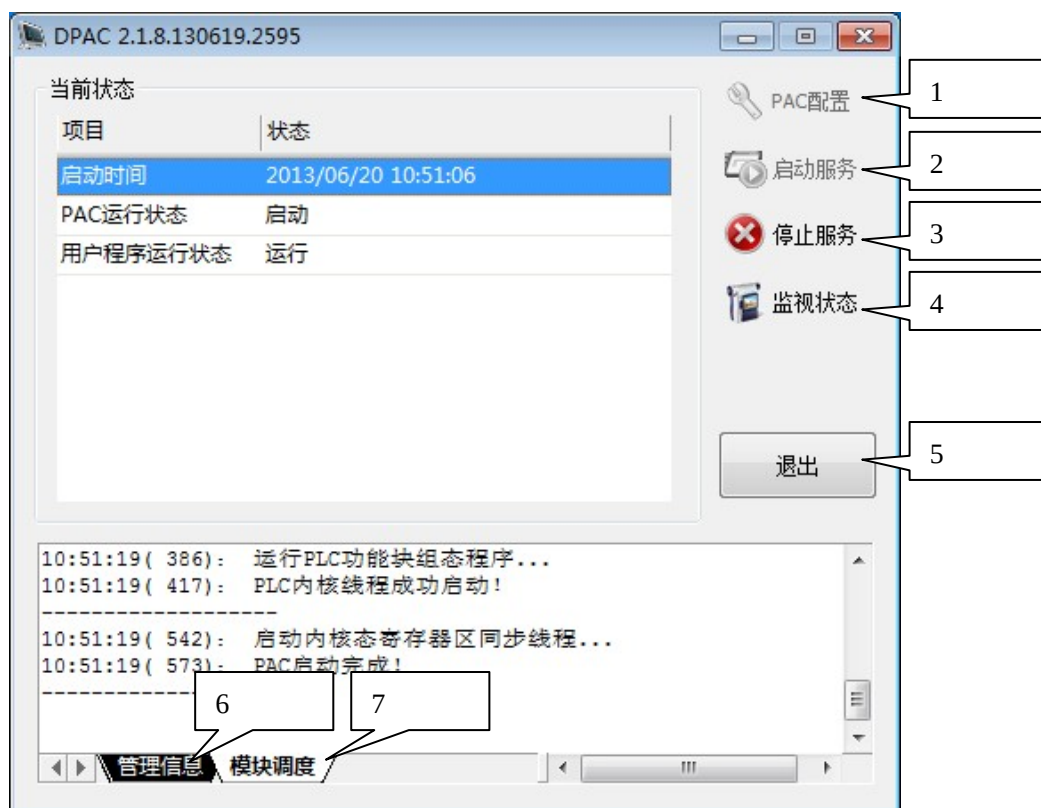


图 4-1-1 DPAC 主界面

DPAC 软件为用户提供如下功能：

(1) PAC 配置

配置 PAC 属性，内容如图 4-1-2 所示。提供以太网接口配置和 Windows 下的串口配置。

(2) 启动服务

点击后启动 PAC 实时内核程序。

(3) 停止服务

停止 PAC 实时内核程序。



图 4-1-2 PAC 参数配置界面

(4) 监视状态

运行状态下，监视当前实时内核的执行情况，如图 4-1-3 所示。



4-1-3 PAC 实时内核状态监控

其中各监视项说明如表 4-1-1 所示。

表 4-1-1 PAC 实时内核状态参数说明

监视项	说明
100ms 计数器	实时内核启动后每隔 100ms 自增 1
1s 计数器	实时内核启动后每隔 1s 自增 1，用户可通过此数据获知，实时内核启动至今的累积时间
主循环次数	实时内核的主循环，主循环每执行一次自增 1，如果存在用户程序，则每进一次主循环，则执行一个用户指令
用户程序执行轮次	执行用户程序的圈数
当前 pc 指针地址	表示当前执行到的指令位置
用户当前执行状态	表示用户程序当前的执行状态
EtherCAT 通信状态	EtherCAT 通信状态，可视 IO 设备的连接状态
网卡模式	分为单网卡模式和双网卡模式
当前通信网卡	单网卡模式下，此项显示网卡号，分为网卡 1 和网卡 2（PAC 为 EtherCAT 通信提供 2 个网卡）；双网卡模式下该项表示主网卡号

（5）退出

停止实时内核运行，并退出 DPAC 程序。

（6）管理信息

反映实时内核的错误状态，如果发生了异常，如配置 IO 设备资源映射失败，则会通过此输出栏显示给用户。

（7）模块调度

DPAC 在启动或停止时，此栏用于提示调度过程。

4.2 设备资源

iTC1015 提供三路千兆以太网接口，实时内核可通过其中两路连接扩展 IO 设备，最多支持 256 台 IO 设备，在运动控制方面，最多支持 16 台运动控制 IO 设备，可同时控制多达 64 个轴。

4.2.1 内部资源

iTC1015 为用户提供了丰富的内部资源，如表 4-2-1 所示。

（1）内核资源

表 4-2-1 PAC-iTC1015 内核资源

PAC工业平板电脑 iTC1015		
IO资源	可带IO设备	256个

	可映射IO点数	AI 4096个 AQ 4096个 DI 8192个 DQ 8192个
内存空间	用户程序存储器	5MB，可存储 10 万条指令左右
	临时数据存储器	51200 字（V 区）可自动保存
	静态数据存储器	512 字（SM 区）自动保存后 256 字
指令执行时间	位运算指令	< 1us
	整形数运算指令	< 1us
	浮点运算指令	< 2us
	运动控制类指令	< 20us
指令资源	定时器总数	1024 个
	计数器总数	128 个
	时间中断	4 个定时中断，4 个定时器中断
	外部中断	无
	中断程序数	16 个
	子程序数（非中断）	240 个
	实时时钟	支持
	NET 指令数	NETR_W/B/D 同时最多支持 64 个功能块 NETW_W/B/D 同时最多支持 64 个功能块
	NET_XMT	最多支持 64 个功能块
	NXMT_RCV	最多支持 32 个功能块
	SCR	SCR/W_B/W 最多支持 256 个功能块
	PID	最多支持 16 路，自整定最多支持 8 路
	CALL	最多支持 16 层子程序调用
	ATCH	最多支持 10 个功能块，即每一个功能块对应一个中断源
	基础指令	位逻辑、网络通信、比较、转换、计数器、浮点数运算、整形数运算、逻辑操作、变量转移、程序控制、中断、移位与循环、定时器
	运动控制	单轴控制指令、多轴控制指令、轴组指令

(2) 通信资源

iTC1015 共有三类通信接口，RS-485 接口、RS-232 接口和以太网接口，参数详见表 4-2-2。

表 4-2-2 PAC-iTC1015 通信参数列表

通信接口	RS-485	RS-232	以太网
通信速率	1200bps~1Mbps	1200bps~115200bps	10/100/1000Mbps自适应
通信协议	MB+协议	自由通信协议主从站	MODBUS TCP/UDP, EPA, 转发协议FEP, TCP/UDP自由通信协议
线缆长度	屏蔽双绞线最长1200米 (低速)	屏蔽双绞线最长10米 (低速)	屏蔽双绞线100米 (高速)

4.2.2 IO 设备

安装 PAC 实时内核软件后 iTC 具有了连接 IO 设备的能力，下面介绍 IO 设备的连接方法。
iTC1015 提供 3 个以太网接口，每个接口的作用各不相同。

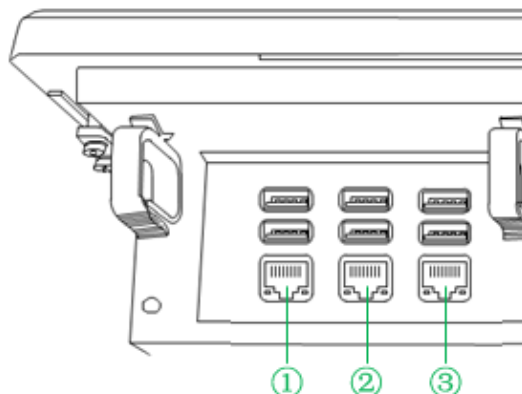


图 4-2-1 iTC1015 以太网接口分布图

表 4-2-3 以太网接口说明

iTC 的以太网接口	
编号	功能
①	由 Windows 管理，负责与外网通信，可以连接交换机、路由器或直接连接工程师站
②	2 号 IO 接口，需要在 Windows 中配置成禁用
③	1 号 IO 接口，需要在 Windows 中配置成禁用

连接 IO 设备分为两种模式：单网卡模式和双网卡模式。

（1）单网卡模式

单网卡模式主要用于连接仅有 1 个以太网接口的 IO 设备，iTC1015 在连接 IO 设备时仅需要使用 1 个 IO 接口皆可实现与 IO 设备的连接，连接方式为环网单工模式，上文描述的②和③接口均可以实现连接（可以在 PAC 实时内核软件中配置连接接口）。

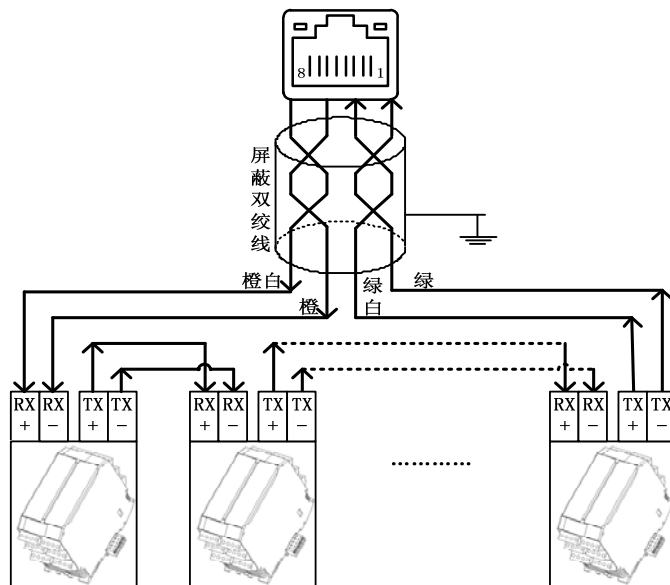


图 4-2-2 单网卡环网连接方式

(2) 双网卡模式

双网卡模式主要用于连接带有 2 个以太网接口的 IO 设备，iTC1015 在连接 IO 设备时使用两个以太网接口，连接方式为环网全双工模式，发送和接口的循序可以在实时内核中配置，默认状态下 1 号为主发送口，2 号为主接收口，下图以 2 号口为发送主口为例。

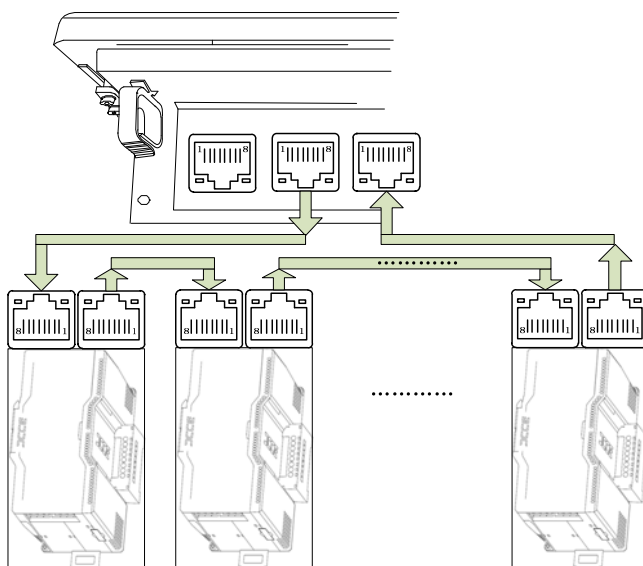


图 4-2-3 双网卡环网连接方式

iTC1015 目前支持的 IO 设备类型与其 IO 资源如表 4-2-4 所示。

表 4-2-4 IO 设备类型与资源列表

设备类型	功能	I	Q	AI	AQ
DEM6000	运动控制	32	32	96	96
IO8000	扩展IO	8	8	8	2

IO5000	扩展IO	16	16	8	8
IO3000	扩展IO	32	32	0	0
IO3400	扩展IO	32	32	0	0
DEM9000	扩展IO	16	16	16	16
DEM5000	扩展IO	0	0	8	8
DEM3000	扩展IO	32	32	48	48
ECAT1000	扩展IO	32	32	16	16
DEM1000	扩展IO	56	71	23	8
EK1100	扩展IO	0	0	0	0
EL2521	扩展IO	0	0	8	12
EL1008	扩展IO	0	0	1	0
EL2008	扩展IO	0	0	0	1
EL3314	扩展IO	0	0	8	4

关于 IO 设备的连接可参考《iTC1015 用户使用手册》，PAC 实时内核支持高精度定时器，可以准确的以 1 毫秒为周期与所有 IO 设备进行一次数据同步。

用户可以在 PLC_Config 软件的控制管理界面中，通过扫描 IO 设备功能，将 PAC 连接的 IO 设备扫描上线，如图 4-2-4 所示。

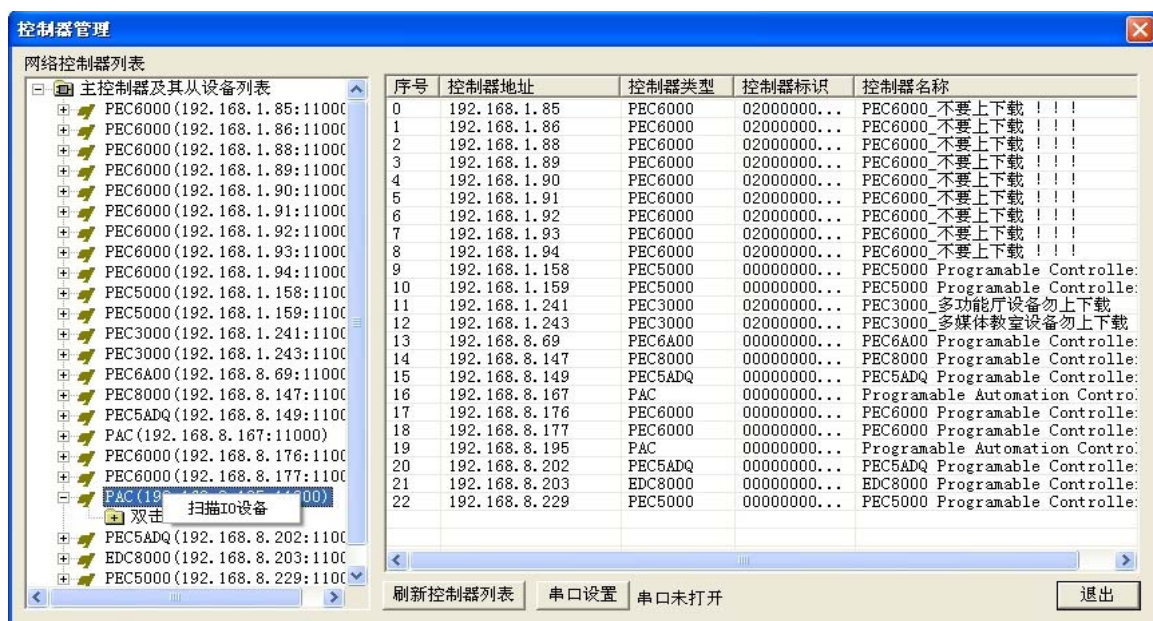


图 4-2-4 扫描 IO 设备

4.2.3 定时器

PAC 实时内核支持 1ms、10ms、100ms 三类定时器，共 1024 个，编号为 T0~T1023。每个定时器寄存器可由三种指令控制：TON、TOF 和 TONR。

系统变量 SM59.0 控制定时器的时基。SM59.0 的默认值为 0，在默认情况下，定时器 T0~T1007 以 100ms 为时基，T1008~T1023 以 1ms 为时基。当 SM59.0 的值被修改为 1 后，原

来以 100ms 为时基的定时器 T80~T119 和 T752~T1007 的时基被修改为以 10ms 为时基,以 1ms 为时基的定时器不变, PAC 实时内核对于时基修改后可立即生效, 时基配置关系见表 4-2-5。

表 4-2-5 定时器时基配置索引表

SM59.0	1ms	10ms	100ms
0	T1008~T1023	--	T0~T1007
1	T1008~T1023	T80~T119、T752~1007	T0~T79、T120~T751

定时器 T124、T125、T126、T127 除了支持定时器指令外, 还可作为中断源, 与中断连接指令配合使用。如果中断开放能够产生中断, 当计数值与预设值相等时, T124~T127 置位, 触发中断程序。定时器 T124~T127 对应的中断号分别是 12~15。

4.2.4 中断源

实时内核提供了 10 个中断源, 其中包括 8 个定时中断和 2 个自由通信中断。8 个定时中断分为 4 个定时器指令中断和 4 个时基匹配中断。如表 4-2-6 所示。

定时器中断有 8 个, 其中 4 个时基匹配中断分别对应中断号 8~11, 匹配设定值分别存储在 SMW190、SMW191、SMW192、SMW193 四个变量区中, 当计数值与匹配值相等时触发中断程序, 如表 4-2-5 所示。4 个定时器指令中断分别对应中断号 12~15, 使用定时器指令中断时需要调用定时器指令 (TON 或 TOF) 与中断连接指令配合使用。

表 4-2-6 定时匹配变量表

SM 区	描述
SMW190	时基匹配 0 定时时间, 单位为 10ms
SMW191	时基匹配 1 定时时间, 单位为 10ms
SMW192	时基匹配 2 定时时间, 单位为 10ms
SMW193	时基匹配 3 定时时间, 单位为 10ms

串行口接收发送中断分别是为串行自由通信指令的串行通信及时响应而设计的, 主要包括发送中断和接收中断。

表 4-2-7 中断事件一览表

中断号	MODE	中断事件	描 述
8	0	时基匹配 0	时基匹配 0 定时时间到产生中断
9	0	时基匹配 1	时基匹配 1 定时时间到产生中断
10	0	时基匹配 2	时基匹配 2 定时时间到产生中断
11	0	时基匹配 3	时基匹配 3 定时时间到产生中断
12	0	定时器 124	T124 定时时间到产生中断

13	0	定时器 125	T125 定时时间到产生中断
14	0	定时器 126	T126 定时时间到产生中断
15	0	定时器 127	T127 定时时间到产生中断
16	0	串口发送	串口 1 自由通信发送指令（XMT）发送结束产生中断
17	0	串口接收	串口 1 自由通信接收指令（RCV）接收到报文产生中断

提示：中断资源的具体使用方法参见《DCCE 网络化可编程控制器用户编程手册》中第 6 章中断指令内容。

4.2.5 可用参数

PAC 实时内核为用户提供了以下参数：始终和首次扫描为 1 标志、1ms、100ms 和 1s 定时器，秒和分钟脉冲和实时时钟等设备信息，如表 4-2-8 所示。可以使用这些数据控制程序运行。

表 4-2-8 设备参数

SM 区	描述	SM 区	描述
SM0.0	始终为 1	SM0.1	首次扫描为 1
SM0.8	指令加密位	SMW1	每 100ms 加 1
SMW6	秒定时器	SMD167	毫秒定时器
SM7.0	秒脉冲	SM7.1	分钟脉冲
SMW52	实时时钟的年	SMW53	实时时钟的月
SMW54	实时时钟的日	SMW55	实时时钟的时
SMW56	实时时钟的分	SMW57	实时时钟的秒
SMW58	实时时钟的星期		

4.3 可编程控制

PAC 实时内核支持 IEC61131-3 标准的梯形图、功能块和指令码编程，指令集丰富，具备过程控制、运动控制，逻辑控制等功能，提供了方便的设备运行信息访问接口，可有效地减少操作人员与维护工程师在故障诊断与排除方面所花费的时间。

PAC 实时内核通过 PLC_Config 编程开发软件进行逻辑组态和编程开发。

4.3.1 编程软件

PLC_Config 是大工计控自主开发的，用于对大工计控 PLC 产品和 PAC 产品进行逻辑组态的编程开发软件，包括工程管理、控制器管理、变量管理、程序编辑、监控等多项功能。支持兼容 IEC61131-3 标准的功能块、梯形图，以及符合 PLCOpen 标准的运动控制指令。在进行 iTC1015 程序开发时，建议使用 PLC_Config 2.8.10 及以上版本。

(1) 软件使用环境

PLC_Config 软件包是基于 Windows 操作系统的应用软件，它支持 32 位的 Windows2000、Windows2003、XP、Windows7 等操作系统。使用 PLC_Config 时，建议您的计算机系统要高于以下配置：

- ✓ CPU 主频 $\geq 1\text{GHz}$;
- ✓ 内存 $\geq 256\text{M}$;
- ✓ 显示器分辨率 $\geq 1024 \times 768$;
- ✓ 硬盘剩余空间 $\geq 512\text{M}$;

(2) 工程管理

PLC_Config 以工程为单位对控制器进行管理，支持新建工程、打开工程、保存工程、工程另存为、最近打开工程、关闭工程、导入工程、打印、软件配置等操作。图 4-3-1 为 PLC_Config 软件的主界面，包括菜单、工具栏、工程树、指令编辑区、输出栏和状态栏等信息。

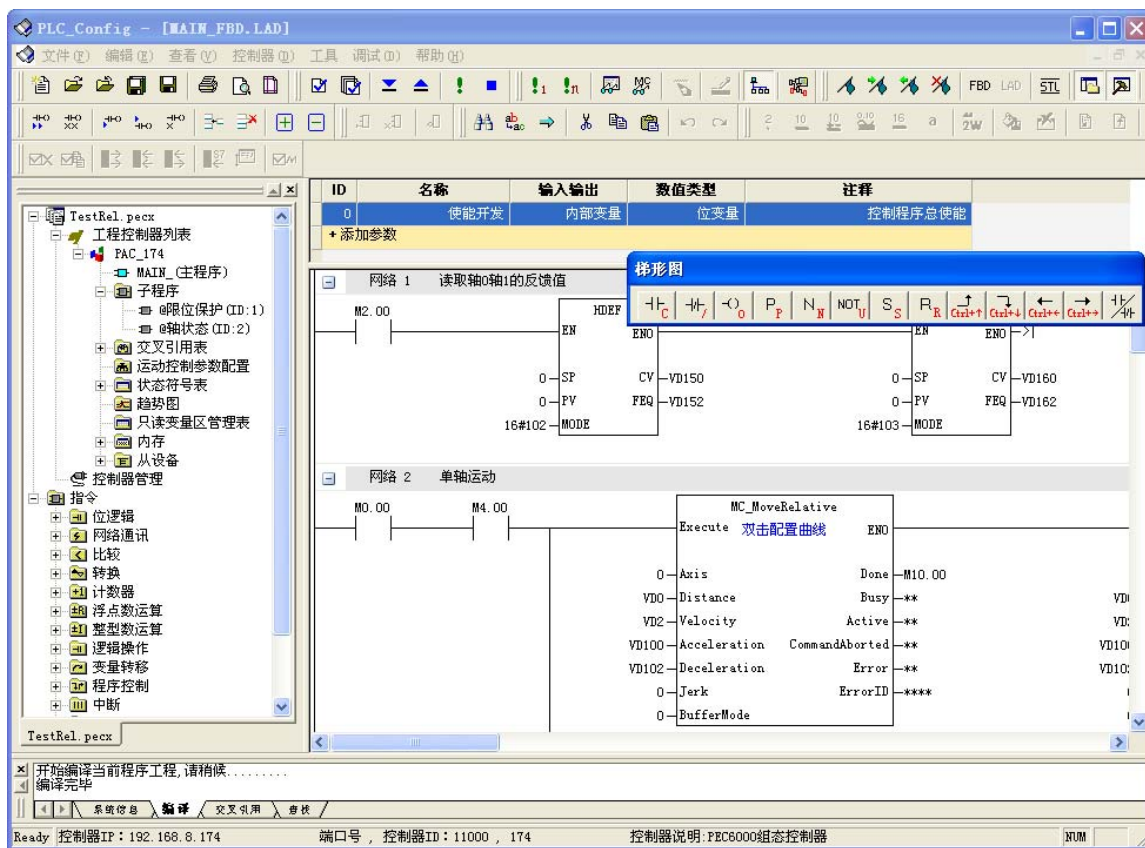


图 4-3-1 PLC_Config 主界面

PLC_Config 软件的主界面项及其对应的功能说明参见表 4-3-1。

表 4-3-1 PLC_Config 主界面说明表

主界面项	功能说明
菜单	列出了 PLC_Config 软件的所有功能
工具栏	列出了 PLC_Config 的常用功能
工程树	列出了工程的主要信息，包含工程名、控制器列表、主程序、子程序、

	交叉引用表、运动控制参数配置、状态符号表、趋势图、只读变量区管理表、内存表、从设备列表、控制器管理和指令树等内容
指令编辑区	支持功能块语言、梯形图语言的编程界面
输出栏	提供控制器上下线、程序编译、搜索结果、报警等信息显示的窗口
状态栏	显示用户编辑的控制器的基本信息

(3) 控制器管理

PLC_Config 软件可对网络内多个控制器的在线状态、控制器参数、从设备配置等信息进行统一管理。PLC_Config 软件将控制器分为网络控制器和工程控制器两类，图 4-3-2 为自动扫描发现的网络控制器列表。网络控制器是在控制器管理中显示出的当前网络内的所有控制器，它不会保存到工程文件中。用户可以选择自己使用的控制器，双击将其添加到 PLC_Config 的工程树中，添加在工程树中的网络控制器变为工程控制器，工程控制器的信息将被保存在工程文件中，下次打开工程时仍可看到该控制器信息。

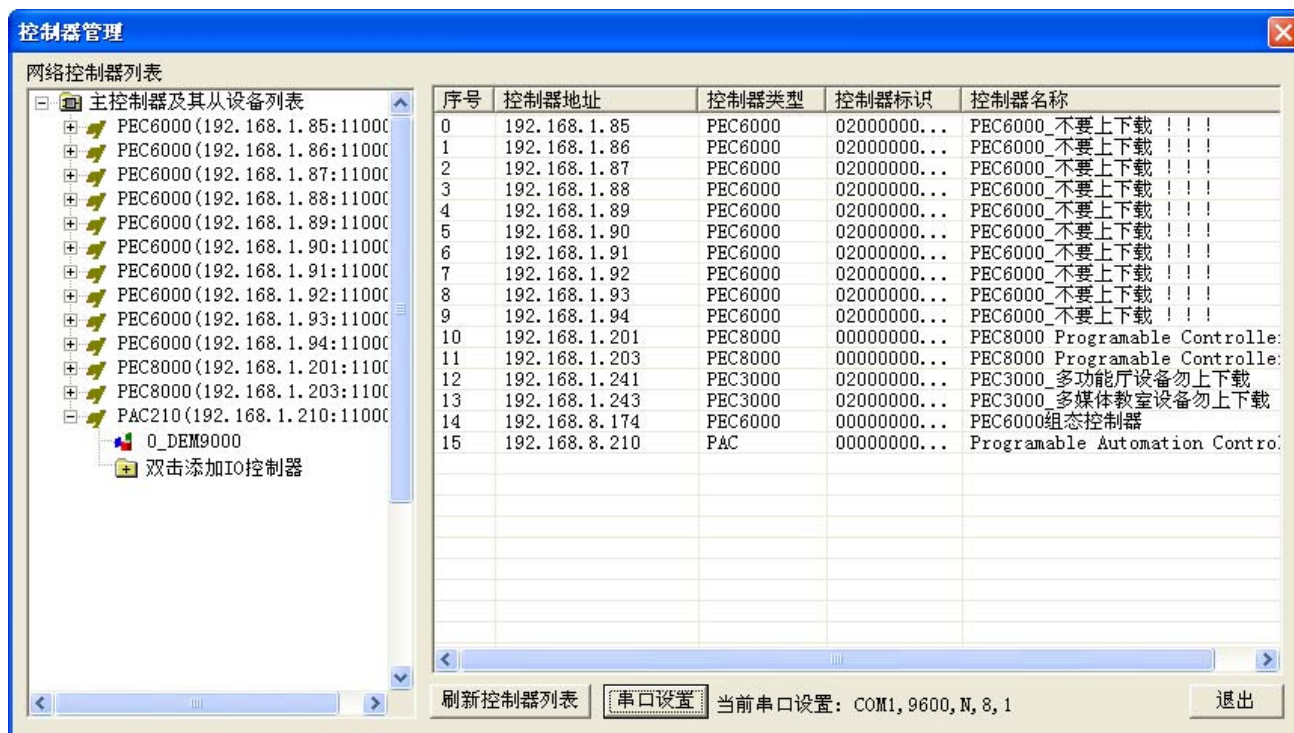


图 4-3-2 控制器管理界面

为满足大批量 IO 点控制需求，增强扩展性，PEC、DIO、AIO 等系列可编程控制器以及第三方设备可通过串口总线扩展为从设备。PLC_Config 软件可对主从设备数据映射方式、从设备参数等信息进行管理与配置，大工计控的产品作为从设备时，还可以直接对从设备进行编程。

(4) 变量管理

PLC_Config 提供多种变量管理功能，如状态符号表、交叉引用表、趋势图、只读变量区管理表、内存表等。用户可以使用这些功能，对程序中的变量进行管理。

状态符号表用于对程序中使用的变量进行集中显示、监控和管理。用户可以定义变量的符号名称、数据类型、变量说明，查看变量的 Modbus 地址，同时支持变量的在线监控、读写变量，和变量表的导入导出。PLC_Config 支持基于符号的编程方式，为用户提供多种符号的定义和使用接口，使用户能够更方便的理解组态程序。图 4-3-3 为基于符号编程的界面。用户在定义好变量与符号的对应关系后，可以直接使用符号进行编程开发。

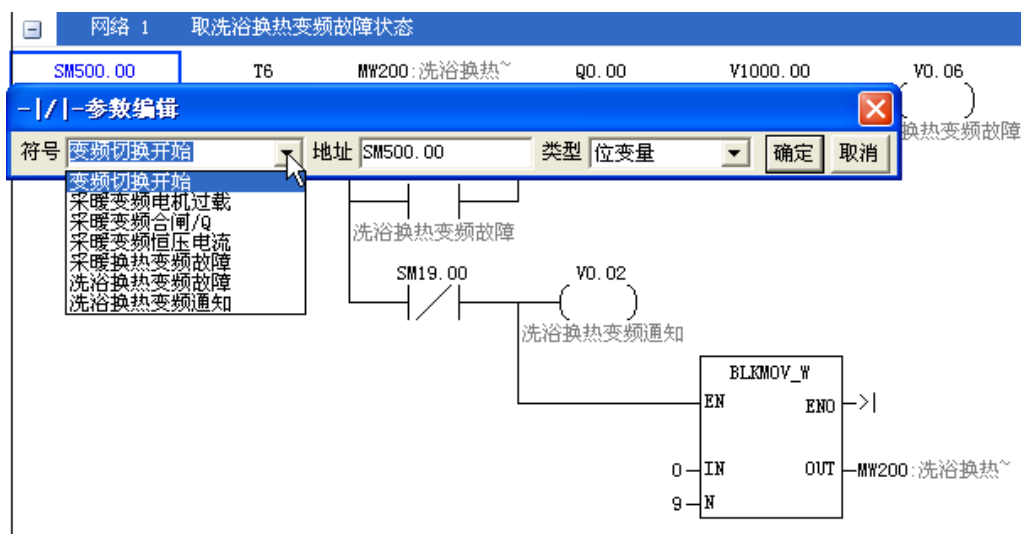


图 4-3-3 基于符号编程的界面

交叉引用表可以列出用户在组态程序中使用变量的情况。在程序编译完毕后，在左侧树中打开交叉引用表，可选择查看字应用、位应用和变量引用表，双击可以进行定位，定位信息显示在输出栏中。

PLC_Config 支持监控变量实时值的趋势图功能。每幅趋势图可以同时监控 8 条自定义颜色的变化曲线，PLC_Config 支持曲线图形和数据的导出，以及曲线图形的打印。只读变量表只能通过软件设置变量，主要对掉电保持区 P 区进行操作。

内存表可以对控制器的变量数据进行监控，将存储区内的数据通过表格以不同的形式显示出来，如二进制、十进制或字符串等，方便用户在不同的模式下进行观察。在内存表中可以根据需要修改变量值并写入到控制器中；也可以将变量值导出或导入，或者将内存表中的值清空。

(5) 程序编辑

PLC_Config 软件提供了主程序替换、添加或删除子程序、子程序的导入、子程序的调用与程序统计功能，方便了用户将已有的工程文件导入到新的工程当中直接应用。在编辑过程中，提供了指令的多重复制、交叉引用查找、拖拽移动指令、变量的替换、编辑网络指令或参数注释、符号的快速定义、子程序参数的快速定义等功能。

编译分为错误检查与生成下载信息两个过程，提供编译当前程序和编译全部程序两种方式。软件提供编译输出栏，提示编译信息，如果编译期间发现错误，则会在输出栏中提示出错误的原因和位置，用户可双击该信息定位到具体的程序位置。表 4-3-2 列出了 PLC_Config 软件一些常见的编译错误情况和解决办法。

表 4-3-2 PLC_Config 软件编译错误和解决办法列表

错误信息	原因和解决办法
无输入参数错误	指令输入没有命名变量地址，写入正确的输入参数
无输出参数错误	指令输出没有命名变量地址，写入正确的输出参数
无头部参数错误	指令头顶参数没有命名变量地址，写入正确的头顶参数
目的控制器 IP 地址与本地控制器 IP 地址相同	网络指令的目的 IP 填入的是本地控制器 IP 地址，写入正确的目的 IP
参数编址错误或者立即数越界	指令参数填入了错误的变量类型，查看指令说明，修改数据类型
PID 通道只支持 16 路	PID 回路号越界，超过了限制范围
网络太复杂不能编译	网络回路中有两个或更多的逻辑起点或一个指令有多个能流流入；梯形图网络出现回路。解决方法是：确保每个网络回路只有一个逻辑起点。如果想要多个参与逻辑控制，用变量传递的方法，在多网络中实现。确保一个功能块指令最多只有一个能流流入。
FOR-NEXT 不匹配	FOR 指令和 NEXT 指令不匹配，检查指令使之相匹配
此网络为转换后无法用梯形图表示的网络	指令参数填入了错误的变量类型，查看指令说明，修改数据类型 功能块转梯形图时，某些功能块网络结构无法用梯形图表示。解决方法：重新转换为功能块，并修改该网络的结构，不能转换的网络结构请查看《PLC_Config 软件编程手册》。

统一下载功能为用户提供了一整套控制系统方案的一键下载功能。用户在完成组态程序和控制器参数配置后可利用 PLC_Config 软件生成统一下载镜像文件，在成批生产时使用 DownloadUtility 软件统一下载。现场的下载人员不需要了解整个控制方案的实现方法，只需按要求将各个模块物理连接起来即可。下载软件会根据镜像的指导将镜像文件的内容一一下载到各个模块中。统一下载功能需要两部分共同完成，先由 PLC_Config 组态软件生成镜像文件，再由 DownloadUtility 软件进行下载。

(6) 监控

PLC_Config 软件支持程序上下载、控制器的启动停止、强制写入、仿真运行、在线编辑、微分监视等功能，为用户编写调试程序提供了方便。

在程序编译完成后，可以通过下载键将编译完成的程序下载到控制器中运行。同时，也可以通过上载键将控制器中的程序上载到工程中。程序下载后控制器将自动进入运行状态，用户可以通过调试键控制程序运行的轮次，运行，停止等控制状态。

在线监控的状态下，可以强制改变变量的数值。为了方便工程师在监控状态下能够实时调试工程，在不停止控制器运行的情况下修改局部程序，PLC_Config 提供了运行时下载功能。

仿真运行是指对当前用户编辑的组态程序在仿真器中模拟运行，方便用户在不具备真实控制器的情况下调试组态程序。启动仿真后，PLC_Config 自动将当前组态设备的程序下载到仿真器中并启动程序监控和运行。

（7）运动控制

PLC_Config 软件为运动控制提供了运动控制参数配置、速度曲线动态调节、运动控制状态监视、指令引脚变量自动分配、运动轨迹仿真调试等多种功能，使用简洁，易于学习理解，方便用户实现更有特色的运动控制需求。

在编辑运动控制指令时，可以通过双击曲线配置进入 VT 曲线配置界面。PLC_Config 软件会根据指令中设置的位置、速度、加减速度和加加速度等信息，生成一条初始的 VT 曲线。用户可以根据应用需求，在此曲线上，手动输入修改运动参数，或者通过拖拽的方式修改运动参数，生成满足控制要求的速度曲线。图 4-3-4 为一条拖拽出来的速度控制曲线，从图上可以清晰的看出这段曲线的运动参数和各运动阶段的运行时间等信息。

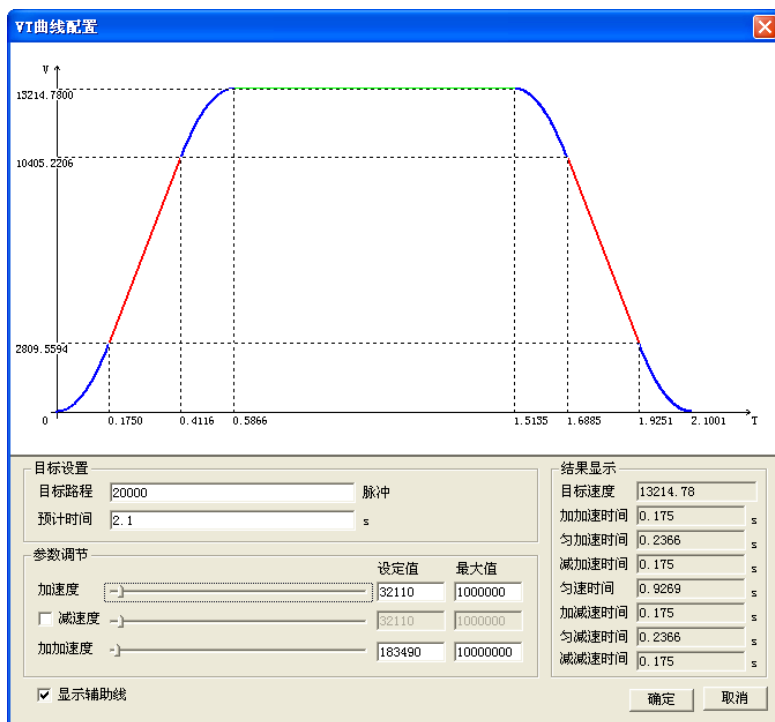


图 4-3-4 VT 曲线配置界面

PLC_Config 提供对轴和轴组的运行状态进行监控的功能。用户可以通过运动控制监控窗口，查看轴和轴组的当前运行状态、目标状态、异常信息等。表 4-3-3 列出了轴状态监控的参数，表 4-3-4 列出了轴组状态监控的参数。

表 4-3-3 轴状态监控参数一览表

参数名称	参数意义	参数类型
------	------	------

轴编号	表示轴的 ID	WORD
禁能状态	轴处于禁能状态	BOOL
停止状态	轴处于停止状态	BOOL
离散运动	轴处于离散运动(绝对、相对运动)状态	BOOL
连续运动	轴处于连续运动状态	BOOL
同步运动	轴处于同步运动状态	BOOL
回原点中	正在利用 MC_Home 或 MC_HomeEx 指令回原点	BOOL
减速停止中	正在利用 MC_Stop 指令进行减速停止	BOOL
异常停止	急停中和急停、异常停止	BOOL
准备就绪	轴准备就绪并处于停止状态	BOOL
被轴组使用	轴正在轴组中使用	BOOL
原点确定	原点确定后置位	BOOL
接近原点	正在接近原点	BOOL
达到速度上限	同步运动时轴速度达到设定的最大速度	BOOL
正方向	轴的运动方向为正	BOOL
负方向	轴的运动方向为负	BOOL
设定位置	轴运动的目标位置	REAL
设定速度	轴运动的目标速度	REAL
设定加速度	轴运动的目标加速度	REAL
设定减速度	轴运动的目标减速度	REAL
当前位置	运动中的实际位置	REAL
当前速度	运动中的实际速度	REAL
是否监视	是否对轴进行异常监视	BOOL
异常代码	异常发生时的异常代码	WORD

表 4-3-4 轴组状态监控参数一览表

参数名称	参数意义	数据类型
轴组编号	轴组编号用于表示轴组的 ID	WORD
禁能状态	轴组处于禁能状态	BOOL
停止状态	轴组处于停止状态	BOOL
运动中	正在向目标位置运动	BOOL
停止中	正在执行 MC_GroupStop 指令	BOOL
异常停止	急停中和急停、异常停止	BOOL
准备就绪	轴准备就绪并处于停止状态	BOOL
速度	轴组运动的目标速度	REAL
加速度	轴组运动的目标加速度	REAL
减速度	轴组运动的目标减速度	REAL
是否监视	是否对轴组进行异常监视	BOOL
异常代码	异常发生时的异常代码	WORD
动力学模型种类	0:直角坐标系,1:圆柱坐标系,2:四轴并联模型,3:三轴并联模型,4:三轴座椅模型,0xFFFF:通用模型	WORD

第一个轴编号	轴组中第一个轴的轴编号	WORD
第二个轴编号	轴组中第二个轴的轴编号	WORD
第三个轴编号	轴组中第三个轴的轴编号	WORD
第四个轴编号	轴组中第四个轴的轴编号	WORD

在使用运动控制指令时，PLC_Config 会为每个指令的输出引脚动态分配一个变量，用户可以直接使用，自动分配的变量为只读变量，不能更改，当需要再次使用这些引脚输出变量时，请用户自行更改为其他小区变量。

PLC_Config 软件为 NURBS 曲线提供了轨迹配置界面，用户可以通过图形化的拖拽进行运动轨迹规划。图 4-3-5 是 NURBS 曲线配置界面。从图中，可以清晰的看出空间轨迹的样式，空间轨迹在三个平面上的投影，以及型值点的坐标等信息。确定轨迹后，可以将轨迹的控制参数下载到指令中，控制轴按照设定的轨迹进行插补运动。

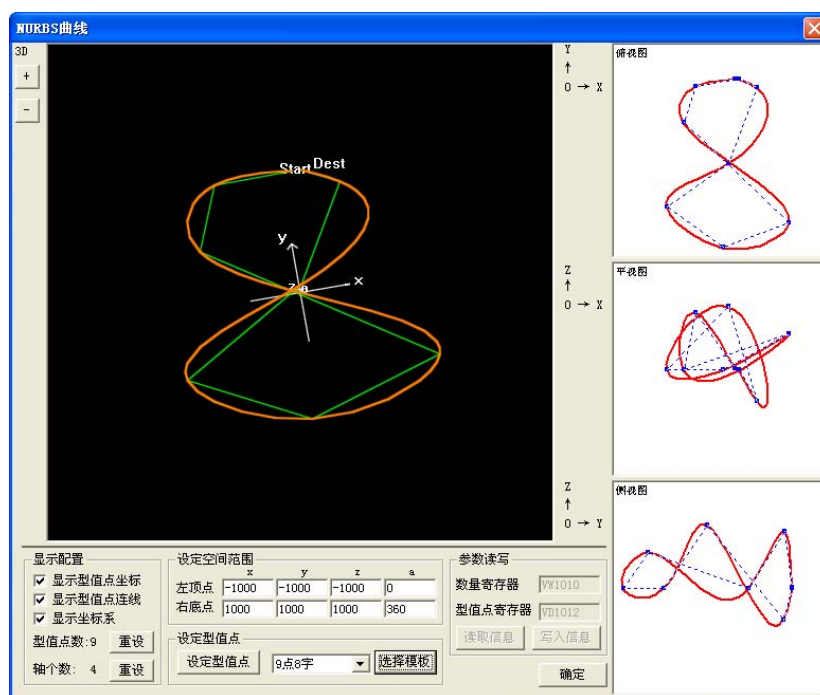


图 4-3-5 NURBS 曲线配置界面

提示：关于 PLC_Config 软件的详细使用信息，请查看《PLC_Config 软件编程手册》。

4.3.2 变量

变量区是为程序提供了一种内存资源，每个变量都有唯一的地址，可通过变量地址直接存取数据，编写程序前需要合理的划分所要使用的变量资源，从而提高内存的利用率和程序执行效率。

变量区的存取支持位、字和双字 3 种类型。位变量，仅有 0 和 1 两个值，主要用于能流控

制、位逻辑运算、使能、状态位存储和数字量输入输出的处理等。字（16 位二进制数）变量包括有符号字和无符号字两种类型，范围分别是-32768~32767 和 0~65535，字变量主要用于模拟量输入输出和数据处理，支持数据传送、算数运算、逻辑运算、定时计数、程序控制、循环移位等操作。双字（32 位二进制数）变量包括有符号双字、无符号双字和实数三种类型，每种变量表示范围分别是-2147483648~2147483647、0~4294967295 和 -3.402823E+38~3.402823E+38。有符号和无符号双字变量和字变量用法相同，主要用于 32 位的整型运算和数据处理。实数类型的双字变量主要用于数值范围较宽的运算和处理，如实数运算、PID 控制等。

变量区的寻址方式可分为直接寻址和间接寻址两种，直接寻址可对位、字和双字寻址。间接寻址只能寻址字和双字变量。PAC 实时内核的内存变量以字为单位，地址连续排列。不支持字节变量，如果使用字节变量，也要存储在字变量中。可以 1 个字节变量存储于 1 个字变量中，仅使用字变量的低 8 位；也可以 2 个字节变量存储于 1 个字变量中，分别存于字变量高低字节中。

各种变量类型定义如下：

(1) 字变量：[区标识符]+[W]+[字地址]

其中，区标识符分别是 V, S, L, M, I, Q 等。W 表示字变量。字地址表示变量存储位置，从 0 开始。例如：VW3，表示变量定义在 V 区中地址为 3 的区域中。

(2) 双字变量：[区标识符]+[D]+[字地址]

其中，区标识符与字变量定义相同。D 表示变量类型是双字变量。字地址表示该双字变量存储的位置。双字变量占用 2 个字变量，分别是字地址指定的字变量和下一个字地址指定的字变量，双字变量低位存于低地址，高位存于高地址。可以通过字变量存取双字变量的高位和低位。例如：VD4，表示双字变量存于字变量 4 和 5 中，可以用 VW4 和 VW5 分别存取该双字变量的低位和高位。

(3) 位变量：[区标识符]+[字地址]+[.]+[位号]

其中，区标识符与字变量定义相同。字地址与字变量定义相同。位号是字变量中的该位的位置，范围是 0~15。例如：V3.11 表示是 V 区字变量地址为 3，位偏移为 11 的位变量操作数，如图 4-3-6 所示。

字变量和双字变量里面存储的数据类型由处理程序定义，控制器中提供了对有符号整型数，无符号整型数，有符号长整型数，无符号长整型数，实数的处理指令。



图4-3-6 变量寻址示意图

除直接使用变量地址外，iTC1015 还支持使用指针的方式间接寻址，iTC1015 使用 V 区和 M 区定义指针变量，间接寻址其他变量。间接寻址中，指针变量为 32 位的整型数值，存储在双字变量中，用户可以通过修改指针实现对数据块变量区中的各个变量的访问。

注意：建议尽量少使用指针变量开发程序，或者使用指针变量时在程序中做越界检查，以免指针变量改变了其他变量而影响程序的正常运行。

提示：详细的指针建立与操作请参见《DCCE网络化可编程控制器用户编程手册》。

变量区分类及范围

iTC1015 定义了 17 个变量区，分别是 I,Q,XI,XQ,V,M,S,SM,L,T,C, AI,AQ,PAI,PAQ,XAI,XAQ 区，按照访问的数据类型可分为字访问（17 个）和位访问（11 个）两种，按照操作权限可分为只读（6 个）和读写（11 个）两种。变量区定义和功能如表 4-3-5 所示。

表 4-3-5 变量区大小

标示符	名称	描述	区大小
I	开关输入	数字量输入寄存器，程序运行周期开始刷新 I 区	4KB 字
Q	开关输出	数字量输出寄存器，程序运行周期结束刷新 Q 区	4KB 字
V	用户变量	用户变量区，掉电保持	51200 字
M	内存变量	变量存储区	512 字
S	顺序控制	以位变量应用为主，主要用于程序的顺序控制	16 字
L	局部变量	以位变量应用为主，作为头参数存放指令当前状态	16 字
SM	特殊功能	具有特殊功能，存放设备参数，系统参数和特殊指令参数	512 字
XI	扩展用户变量	扩展程序变量区	256 字
XQ	扩展用户变量	扩展程序变量区	256 字
AI	扩展用户变量	扩展程序变量区	4KB
AQ	扩展用户变量	扩展程序变量区	4KB
PAI	扩展用户变量	扩展程序变量区	256 字
XAI	扩展用户变量	扩展程序变量区	512 字
PAQ	扩展用户变量	扩展程序变量区	256 字

XAQ	扩展用户变量	扩展程序变量区	512 字
T	定时器	位变量表示时间到否，字变量表示定时器的过程值	1024 个
C	计数器	位变量表示计数到否，字变量表示当前计数值	128 个

变量使用：

变量主要由指令处理，以实现各种运算和控制功能，也可以通过网络与其他设备共享。因此，变量访问方式可分为指令读写，和以太网读写两种。

● **指令读写：**指令输入可以是常量也可以是变量，但输出必须是变量，用于存放指令执行结果。根据指令要求的变量类型，应该在变量区中使用相应变量。指令输入读取变量时将处理结果写入相应变量中。

● **以太网读写：**以太网是对等网络，控制器以读写指令形式实现设备之间变量映射。以太网读写指令支持 EPA、UDP、TCP 以及 ModbusTCP/UDP 等协议。编程软件使用 EPA 协议管理网络中的设备，完成控制程序开发。

存储方式

各种变量区具有不同性质，根据程序开发需要选择变量区，定义相应变量。

● 掉电保持区（V 区）：

V 区具有掉电保持功能，默认为全保持，当用户退出时，程序会将 V 区的信息以文件的形式保存在 PAC 软件的安装路径下。

● 特殊功能变量区（SM 区）：

控制器使用 SM 区开放了系统的一些功能，提供了和用户程序之间信息交互方法，可以通过 SM 区使用一些特殊功能，如：首次扫描为 1 标志和运行状态等。SM 区的前 256 个字变量是只读的，后 256 个字变量和 V 区相同，以文件的形式存放在 PAC 的安装路径下，具有永久保存能力。SM 区提供了控制器的厂家参数、设备参数、指令参数、通信参数和系统参数等。

● 其他变量区：

其他变量区，可保存过程中数据，掉电丢失，用于存储实时数据、控制逻辑、物理输入输出等。

4.3.3 指令

PAC 实时内核共支持 143 条逻辑控制指令，45 条运动控制指令，按具体功能共分为 15 类。

- **位逻辑指令**实现与、或、置位和复位等位运算功能。
- **网络通信指令**实现 Modbus、UDP、TCP、自由通信等协议的网络通信。
- **比较指令**实现字、整数、长整和实数的比较功能。
- **转换指令**实现实数与整数、整数与 BCD 码、长整与实数和长整与整数的相互转换功能。
- **计数指令**实现边沿计数、高速输入和高速输出的计数功能。

- **实数运算指令**实现实数运算功能，如 PID 运算、四则运算等。
- **整数运算指令**实现整数数学计算、整数递增和递减等运算功能。
- **逻辑操作指令**实现字、双字的取反、与、或和异或等逻辑功能。
- **变量转移指令**实现变量的字、双字、实数、字块的赋值和转移功能，以及字的立即传送功能。
- **程序控制指令**实现程序的循环、跳转和子程序调用功能。
- **中断指令**实现中断的允许、禁止、连接和分离功能。
- **移位与循环指令**实现字、双字的左移、右移、循环左移和循环右移功能，以及寄存器移位和交换功能。
- **定时指令**实现 1ms、10ms、100ms 时基的定时接通、定时断开、保持定时接通等定时功能和中断功能。
- **专用指令**实现冒泡排序、数据校验等特殊功能。
- **运动控制指令**通过动力学模型选择，插补运算和加减速曲线，实现运动控制功能。

表 4-3-6 功能块指令索引表

标识符	说明	标识符	说明
AND	位与	ABS	绝对值
OR	位或	I_PID	增量式 PID
P	上升沿	P_PID	位置式 PID
N	下降沿	ADD_I	整数加
=	输出	ADD_DI	长整加
=I	立即输出	SUB_I	整数减
S	置位	SUB_DI	长整减
SI	立即置位	MUL	整数乘
R	复位	MUL_I	整数短乘
RI	立即复位	MUL_DI	长整乘
RS	复位优先触发	DIV	整数除
SR	置位优先触发	DIV_I	整数整除
NETR_B	UDP 位读	DIV_DI	长整整除
NETR_W	UDP 字读	INC_I	整数加 1
NETR_D	UDP 双字读	INC_W	字加 1
NETW_B	UDP 位写	INC_DW	双字加 1
NETW_W	UDP 字写	DEC_I	整数减 1
NETW_D	UDP 双字写	DEC_W	字减 1
SCR_B	串行位读	DEC_DW	双字减 1
SCR_W	串行字读	INV_W	字取反
SCW_B	串行位写	INV_DW	双字取反
SCW_W	串行字写	WAND_W	字与
XMT	串行自由发送	WAND_DW	双字与
RCV	串行自由接收	WOR_W	字或
NET_XMT	以太网写	WOE_DW	双字或

NXMT_RCV	以太网确认读	WXOR_W	字异或
==W、==I	相等	WXOR_DW	双字异或
<>W、<>I	不等于	MOV_W	字赋值
>=W、>=I	大于等于	MOV_DW	双字赋值
<=W、<=I	小于等于	MOV_R	实数赋值
>W、>I	大于	BLKMOV_W	字块赋值
<W、<I	小于	BLKMOV_DW	双字块赋值
==D、==R	相等	MOV_WIR	立即传送读
<>D、<>R	不等于	MOV_WIW	立即传送写
>=D、>=R	大于等于	FMOV_W	多点变量赋值
<=D、<=R	小于等于	FOR	循环头
>D、>R	大于	NEXT	循环尾
<D、<R	小于	JMP	跳转头
R_I	实数整数转换	LBL	跳转尾
I_R	整数实数转换	CALL	子程序调用
I_DI	整数长整转换	ENI	中断允许
DI_I	长整整数转换	DISI	中断禁止
DI_R	长整实数转换	ATCH	中断连接
R_DI	实数长整转换	DTCH	中断分离
BCD_I	BCD 整数转换	SHL_W	字左移
I_BCD	整数 BCD 转换	SHL_DW	双字左移
ROUND	四舍五入	SHR_W	字右移
TRUNC	取整	SHR_DW	双字右移
CTU	上升计数	ROL_W	字循环左移
CTD	下降计数	ROL_DW	双字循环左移
CTUD	上升下降计数	ROR_W	字循环右移
HDEF	高速计数	ROR_DW	双字循环右移
PLS	高速脉冲输出	SHRB	移位寄存器
ADD_R	实数加	SWAP	字节交换
SUB_R	实数减	EXCHG	字交换
NUL_R	实数乘	TON	定时接通
DIV_R	实数除	TONR	定时断开
SQRT	开方	TOF	保持定时接通
SIN	正弦	WPC	工件检测
COS	余弦	PS	经济运行
TAN	正切	STORE	存储
LN	自然对数	B_SortW	字冒泡排序
EXP	自然幂指数	B_SortD	双字冒泡排序
ARC_SIN	反正弦	B_SortF	实数冒泡排序
ARC_COS	反余弦	CKSUM_G	数据校验码生成
ARC_TAN	反正切	CKSUM_C	数据校验码检查

表 4-3-7 运动控制指令索引表

标识符	说明	标识符	说明
MC_MoveAbsolute	绝对定位	MC_Phasing	相位偏移
MC_MoveRelative	相对定位	CamTableCreate	创建凸轮表
MC_MoveAdditive	叠加定位	CamTableDelete	删除凸轮表
MC_MoveSuperImposed	重新定位	CamTableWriteData	写凸轮表数据
MC_MoveVelocity	恒速运行	CamTableReadData	读凸轮表数据
MC_Stop	停止轴	CamTableReadSlavePos	读从轴位置
MC_Jog	点动运行	MC_MoveLinearAbsolute	绝对直线运行
MC_Home	回原点	MC_MoveLinearRelative	相对直线运行
MC_HomeEx	扩展回原点	MC_MoveSplineAbsolute	绝对样条运行
MC_Reciprocator	往复运行	MC_GroupJog	轴组点动运行
MC_Reset	复位轴	MC_WriteActPosition	写轴组位置
MC_ReadStatus	读轴状态	MC_GroupMoveVelocity	轴组恒速运行
MC_ReadAxisError	读轴故障	MC_GroupStop	停止轴组
MC_ReadParameter	读轴参数	MC_MoveCircular2D	平面圆弧运行
MC_WriteParameter	写轴参数	MC_ContinuousCurve	连续曲线运行
MC_ReadBoolParameter	读布尔型参数	MC_GroupReadActualPosition	读轴组位置
MC_WriteBoolParameter	写布尔型参数	MC_GroupReadActualVelocity	读轴组速度
MC_ReadActualPosition	读轴位置	MC_GroupReadError	读轴组故障
MC_WriteActualPosition	写轴位置	MC_GroupReset	轴组复位
MC_CamIn	凸轮运行	MC_GroupReadStatus	读轴组状态
MC_CamOut	凸轮分离	MC_GroupEnable	轴组使能
MC_GearIn	齿轮运行	MC_GroupDisable	轴组禁能
MC_GearOut	齿轮分离	--	--

PAC 实时内核的指令主要应用于过程控制、逻辑控制和运动控制。

(1) PID 控制

PAC 实时内核具有带自整定功能的 PID 控制指令，可以根据被控对象自动生成控制参数；提供多种 PID 控制方式的选择，为用户提供更为专业的控制方案；实现精确的温度控制，控制精度可达 $\pm 0.1^{\circ}\text{C}$ 。

PID 控制参数，包括比例系数、积分系数和微分系数。在不同系统（温度系统、流量系统、压力系统等）中参数不同，其参数可根据以下三种方法获得：

A) 先设置为不同系统的经验值，然后根据调节效果修改。

B) 建立系统的数学模型，用仿真的方式计算出各参数的数值，再用于实际中。

C) PAC 实时内核提供了 PID 参数自整定功能，可以根据现场对象自整定出一套参数，供 PID 指令使用。

PID 自整定功能分为无过冲自整定和有过冲自整定。PID 自整定功能仅支持位置式 PID 算法，即 P_PID，并且只对 LOOP 号 0~7 的 PID 指令起作用。位置式 P_PID 指令如图 4-3-7 所示。

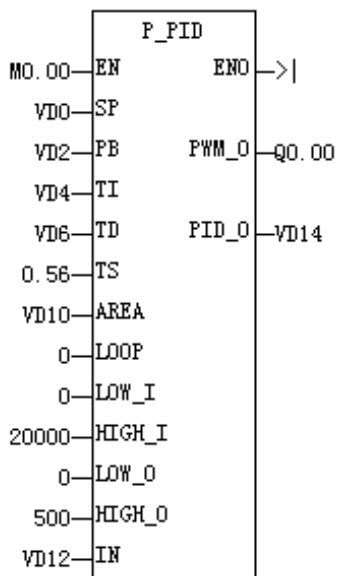


图 4-3-7 位置式 P_PID 指令

PID 自整定功能通过设置 SM 区的标志位来实现，进行自整定前需要根据选用的 PID 通道号配置自整定使能标志位 SM320.i (i=0-7)，整定方式选择字 SM109 和输出方式选择字 SM107 等，整定后的参数保存在 SM 区中，掉电不丢失。请根据被控对象的特性，适当的调节控制参数。

提示：PID 的使用请参见《DCCE 网络化可编程控制器用户编程手册》的 6.8.3 节。

● 有过冲自整定

有过冲自整定法基本原理是通过设定继电器滞环参数，使系统产生等幅振荡，然后在继电器反馈下观察过程的极限振荡环，由极限振荡环获得系统的临界状态，然后由临界状态计算出 PID 的参数。采用有过冲自整定法对某被控对象进行自整定时，其参数配置可根据控制环境和被控对象适当选择。如图 4-3-8 所示，是控温对象的控制参数整定和测量的整个过程曲线，曲线为温度的变化情况，该曲线图的设定温度为 80℃。

● 无过冲自整定

由于有过冲自整定的整定过程中被控对象需经过振荡过程，此过程中会出现测量值大于设定值的情况，这种情况在一些工业场合是不允许出现的，可能发生危险或使被控对象的性质发生变化（比如化学反应等）而失去控制意义。此外，对滞后特性较大的被控对象，采用有过冲自整定法所需周期较长（几十个小时以上），同时整定出来的参数控制稳定性也较差。基于上述因素，可采用飞升曲线法来确定 PID 控制参数，其工作原理是将被控对象从冷态上升至设定值的过程建立数学模型，获取被控对象的模型参数，从而计算出 PID 的控制参数，达到稳

定上升而不超调的目的。采用无过冲自整定法对某控制对象进行自整定控温时 also 需根据控温环境和被控对象适当选择参数，整个整定和控温过程曲线如图 4-3-9 所示，该曲线是温度随时间变化曲线，设定控温温度为 80℃。

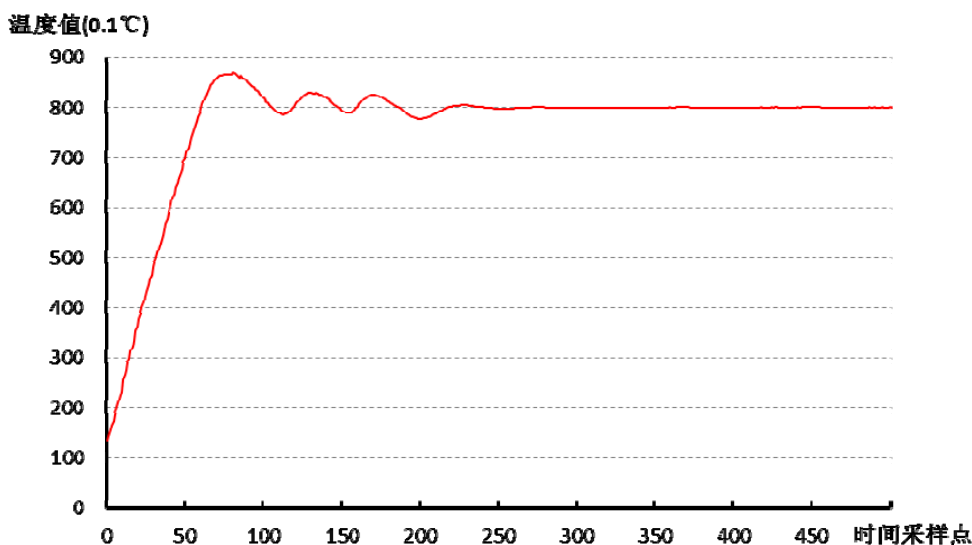


图 4-3-8 P_PID 有过冲自整定控温曲线图

注意：

- 如果 PID 指令使用 PWM 输出，PWM_O 引脚只能接开关输出变量 Q，对应到数字量输出引脚，不支持其他变量区。
- 连接到 PID_O 指令输出引脚，应完全由 PID 指令控制，不得作为程序中其他指令的输出使用，否则可能导致 PID 指令算法不执行或执行错误。

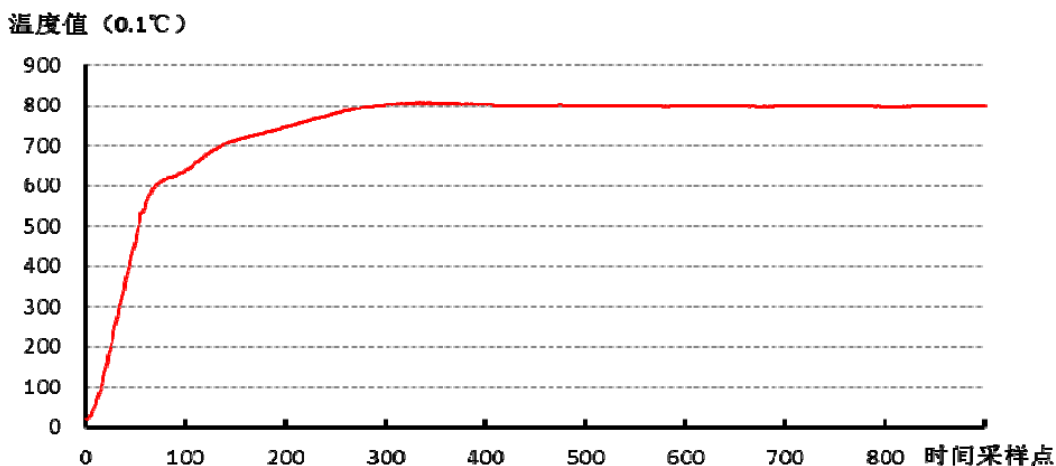


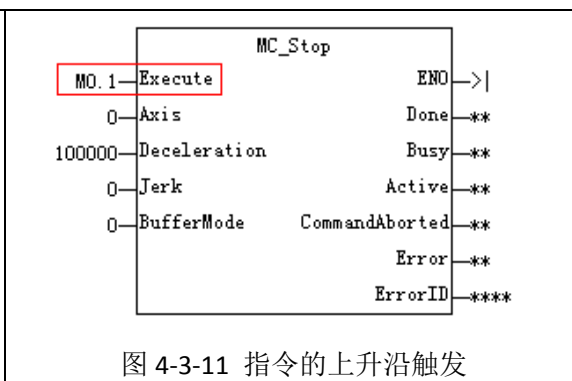
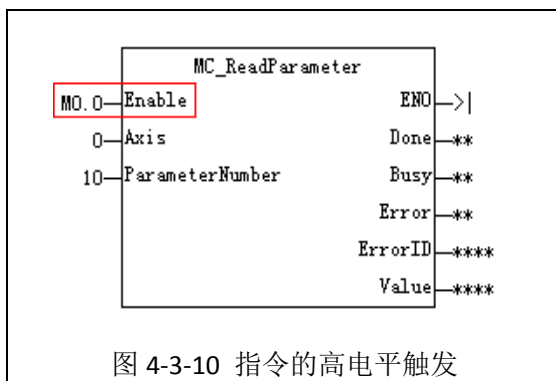
图 4-3-9 P_PID 无过冲自整定控温曲线图

(2) 运动控制

iTC1015 提供了符合 PLCopen 标准的单轴，多轴，轴组三大类运动控制指令，通过 PLC

指令的方式对机械运动部件的位置、速度等进行实时的控制管理，使其按照预期的运动轨迹和规定的运动参数进行运动。

在运动控制指令中，对于指令的触发方式，根据其使能引脚名称可分为高电平触发和上升沿触发两种。高电平触发方式使能引脚的名称为 Enable。当 Enable 引脚为 TRUE 时，指令使能，当 Enable 引脚为 FALSE 时，指令禁能。如图 4-3-10 所示。上升沿触发方式使能引脚的名称为 Execute。当 Execute 引脚的上升沿产生时，指令使能，读取输入变量，执行指令。其他情况均不执行指令体，如图 4-3-11 所示。



对于具有输出引脚 Done、Busy、CommandAborted 及 Error 的运动类指令，其输出引脚信号状态会受到能流输入引脚 Execute 的影响，以运动执行完成时的情况进行说明。运动执行完成后，指令输出引脚 Done 变为 TRUE。能流输入引脚 Execute 的信号状态影响输出引脚 Done 的持续时间，引脚的时序变化如图 4-3-12 所示。其他情况下引脚的时序类似，详细情况请参见《DCCE 编程手册 运动控制指令篇》。

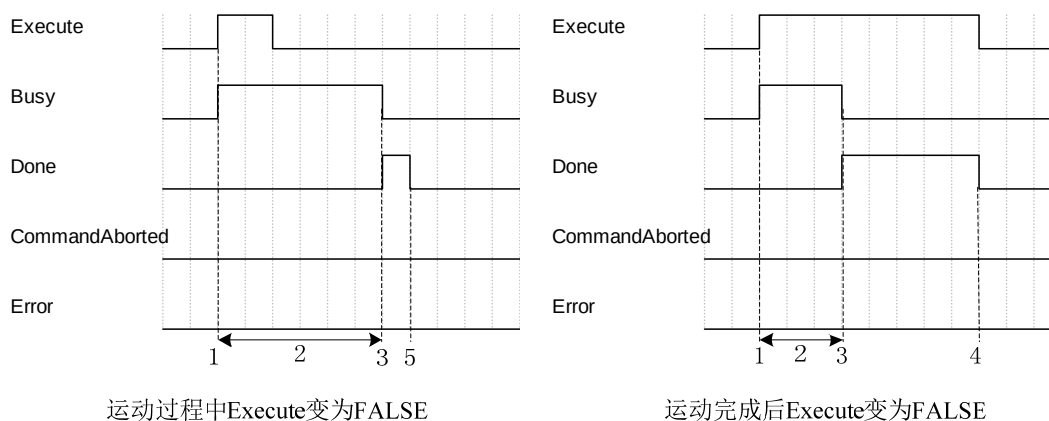


图 4-3-12 运动执行完成时的引脚时序

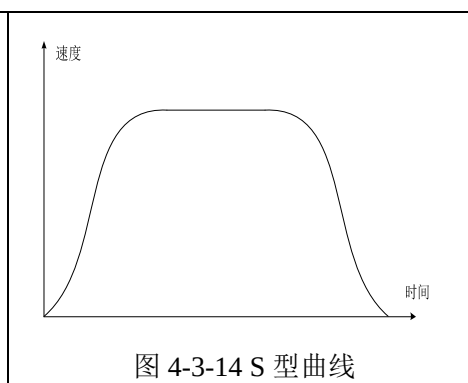
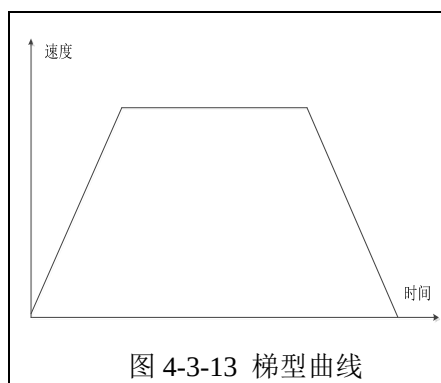
引脚时序变化各个阶段的说明参见表 4-3-8。

表 4-3-8 运动执行完成时的引脚信号变化

信号	变化
----	----

1	能流输入引脚 Execute 上升沿时启动运动控制。根据编程情况，Execute 引脚在运动过程中可能被置为 FALSE，或者保持为 TRUE。
2	运动过程中 Busy 引脚为 TRUE。
3	运动完成后，Busy 引脚变为 FALSE，Done 引脚变为 TRUE。
4	如果 Execute 引脚在运动完成之前保持 TRUE，则 Done 引脚保持为 TRUE 并随 Execute 一起变为 FALSE。
5	如果 Execute 引脚在运动完成之前置为 FALSE，则 Done 引脚仅在一个执行周期内为 TRUE。

运动控制指令的运动方式有两种，一种为梯型曲线，一种为 S 型曲线。梯型曲线运动时，需要设定的运动参数有加速度、减速度、目标速度，如果当前速度小于目标速度，则通过加速—恒速—减速三个阶段运行到目标位置；如果当前速度大于目标速度，则通过减速—恒速—加速三个阶段运行到目标位置；如果当前速度等于目标速度，则通过恒速—减速两个阶段运行到目标位置。运动的曲线如图 4-3-13 所示。



S 型曲线运动时，需要设定的运动参数有加加速度、加速度、减速度、目标速度。如果当前速度小于目标速度，则通过加加速—恒加速—减加速—恒速—减加速—恒减速—减减速七个阶段运行到目标位置；如果当前速度大于目标速度，则通过减加速—恒减速—减减速—恒速—减加速—恒减速—减减速七个阶段运行到目标位置；如果当前速度等于目标速度，则通过恒速—减加速—恒减速—减减速四个阶段运行到目标位置。运动的曲线如图 4-3-14 所示。

单轴指令主要是针对单个轴的位置，速度进行准确控制，一般应用在点位控制场合。下面以绝对定位指令 MC_MoveAbsolute 进行说明。图 4-3-15 显示了在第一个绝对定位指令执行完成后调用第二个绝对定位指令的情况。第一个指令控制轴运动到目标位置 100000 units（同时速度减为 0）后，其 Done 引脚输出高电平信号引起上升沿使能第二个指令，控制轴 0 运动到目标位置 200000 units。

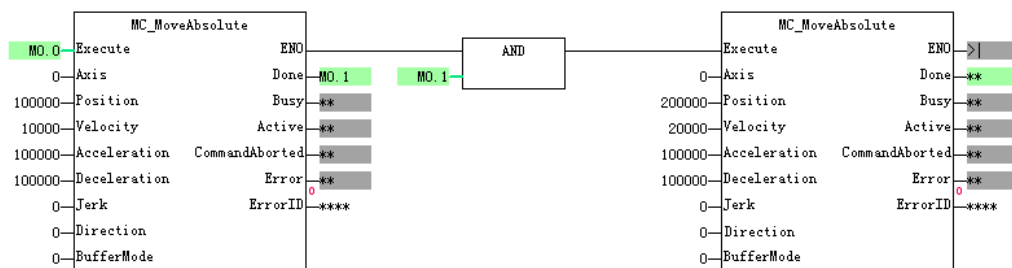


图 4-3-15 绝对定位运行

图 4-3-16 显示了绝对定位指令的引脚时序和速度、位移变化。Busy 和 Execute 同时变为 TRUE，Active 在下一周期变为 TRUE；当位移达到目标值而且定位完成时 Done 变为 TRUE；如果有其它的指令打断当前指令，CommandAborted 变为 TRUE，同时 Busy 和 Active 变为 FALSE；如果在指令执行期间有错误发生，则 Error 变为 TRUE，同时轴将停止运动，用户可以通过 ErrorID（错误码）的值找到导致错误的原因。

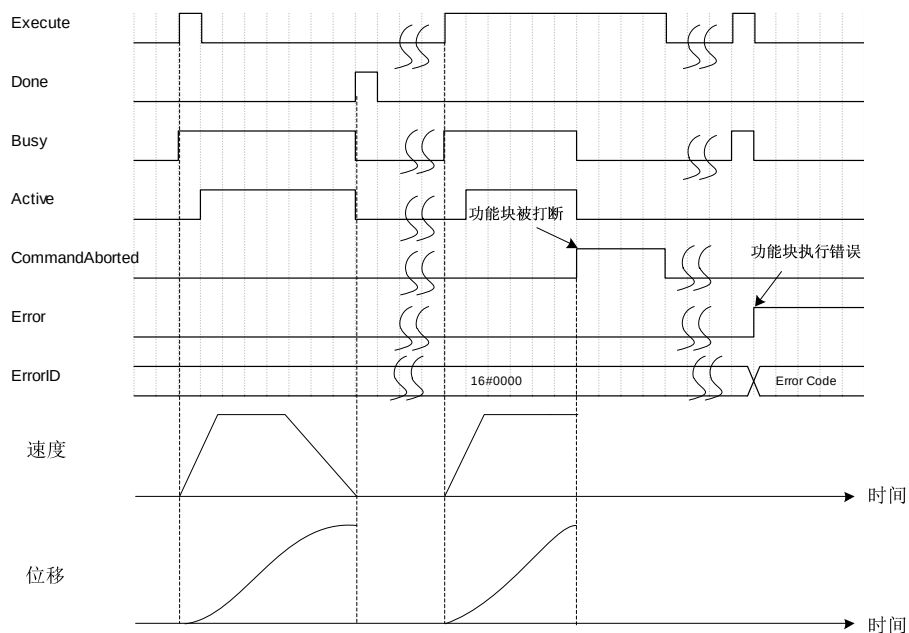


图 4-3-16 绝对定位指令引脚时序和速度、位移变化图

多轴指令主要用于轴和轴之间的运动协调控制，可以是多个轴在运动全程中进行同步，也可以是在运动过程中局部的速度同步，多轴指令主要由电子凸轮和电子齿轮组成。下面主要以电子凸轮指令进行说明。图 4-3-17 为凸轮运行指令。该指令使能后，主轴与从轴按照设定的凸轮表中主轴与从轴位置开始同步运动，主轴保持恒速运动，从轴以当前速度为起始速度，随着主轴自动变速运动。非周期执行时，凸轮表执行结束后，主轴与从轴的速度为零。当凸轮周期执行并且选择绝对位置时，凸轮表中的数据应大于两行。周期执行时凸轮表中最后一行主轴

与从轴的位置应与首行一致。

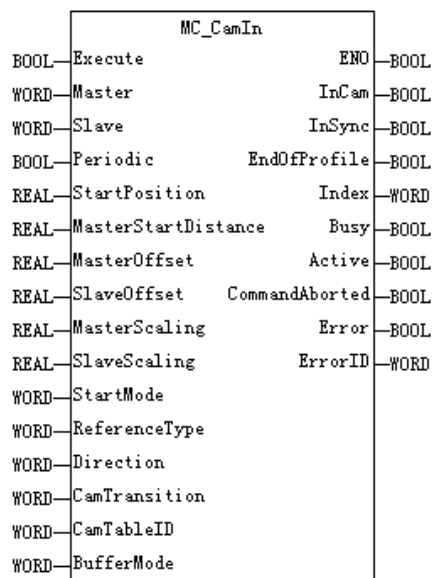


图 4-3-17 凸轮运行指令

图 4-3-18 显示了凸轮运行指令的引脚时序和位移变化。

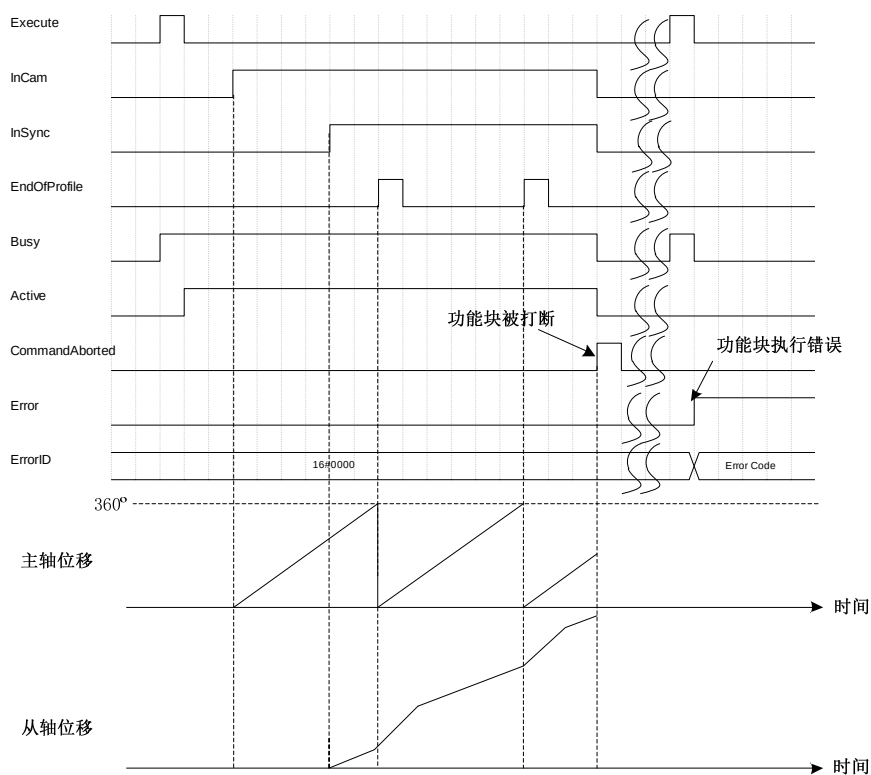


图 4-3-18 凸轮运行指令引脚时序和位移变化图

Busy 和 Execute 同时变为 TRUE，Active 在下一周期变为 TRUE；凸轮耦合开始时 InCam 变为 TRUE；耦合完成时 InSync 变为 TRUE；执行完凸轮表中的记录后 EndOfProfile 变为 TRUE；如果有其它的指令打断此指令，CommandAborted 变为 TRUE，同时 Busy 和 Active 变为 FALSE；如果在指令执行期间有错误发生，则 Error 变为 TRUE，同时轴将停止运动，用户可以通过 ErrorID（错误码）的值找到导致错误的原因。

轴组指令将多个轴绑定到一起，作为一个整体，进行统一的控制管理，为某种应用需求建立动力学模型，进行轨迹插补运行。PAC 支持直角模型、圆柱模型、四轴并联模型、三轴并联模型、三轴座椅模型、通用模型等多种动力学模型，支持空间直线、圆弧、NURBS 曲线插补等多种插补算法。下面以绝对样条运行指令为例进行说明。图 4-3-19 为绝对样条运行指令。设置的参数主要包括轴组号、型值点、权值、型值点个数、插补时的最大弓高误差、运动参数等信息。

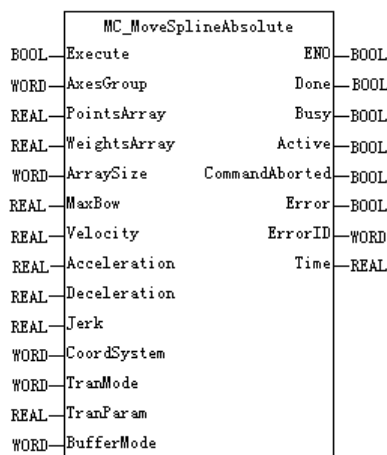


图 4-3-19 绝对样条运行指令

样条曲线根据用户输入的型值点、权值、型值点个数，拟合出一条 NURBS 曲线，然后根据运动参数和输入的最大弓高误差值，进行插补运行。图 4-3-20 为一条绝对样条运行指令生成的运动轨迹。空间轨迹拟合与插补计算后，再根据选择的动力学模型，分解输出到各个轴上。

(3) 逻辑控制

PAC 实时内核具有多种逻辑控制指令，可实现各种逻辑运算，包括：位、字、双字的逻辑与、或、置位、复位等操作，以及比较、转换、数学运算、变量转移、中断、定时、计数、移位与循环和程序控制等操作。同时，还将一些常用的特殊功能程序封装作为特殊指令提供给用户，极大地减少了用户的开发与调试时间。

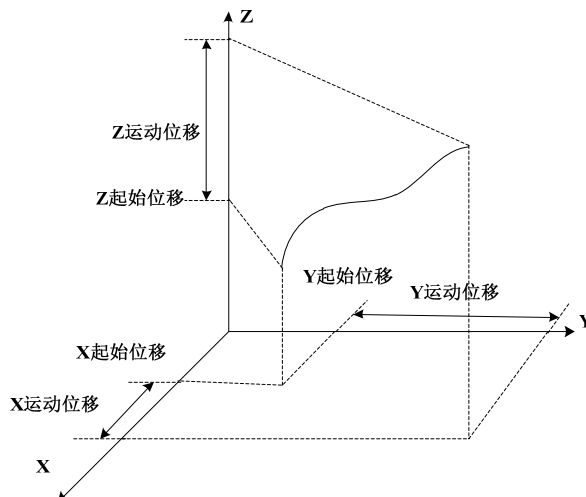


图 4-3-20 绝对样条运行轨迹

提示：关于运动控制指令的详细说明参见《DCCE 编程手册 运动控制指令篇》。

4.3.4 程序

程序是一组为完成某种功能而按一定顺序由算法确定编排的指令序列，PAC 实时内核将程序和输入输出物理量关联起来，通过程序执行控制物理量的输入输出变化，依据被控对象要求实现控制功能。编程就是为解决某个问题而使用梯形图或功能块编写程序代码，并最终得到希望结果的过程。下面从程序的基本组成，运行机制，工作模式，存储加密等四个方面介绍程序。

（1）程序组成

控制程序包含主程序、子程序和中断程序 3 个部分，其中子程序和中断程序是可选的，根据实际的控制过程需要由用户自由选择。

● 主程序

主程序是控制程序的基本程序，能够调用子程序，而不能被其他子程序所调用，是控制程序的中心部分。程序执行以周期扫描形式，从主程序的第一条指令开始顺序执行，循环往复。一个控制程序有且只有一个主程序。

● 子程序

子程序是用户程序的可选部分，只有当被调用的时候才能够执行。使用子程序可以将一段程序封装成功能独立的新的控制指令形式，增强了程序的复用性，有效地减少程序代码的冗余，并可以将很多功能独立程序应用于其他程序中，建立控制程序库。合理的使用子程序还可以减少控制程序的执行时间，提高运行速度。子程序由 CALL 指令调用，调用者可以是主程序或其他程序，编程软件 PLC_Config 将子程序直接显示成功能块格式，形式上相当于定义了新的功能块指令。子程序可嵌套调用，最多可嵌套 16 层。

● 中断程序

中断程序是用户程序的可选部分，只有当相应的中断事件发生时，中断程序才会被执行。中断程序由 ATCH 指令激活，与 PAC 实时内核内部的中断事件关联起来，由中断事件触发。中断事件到来时，会打断当前正在执行的程序，转而执行中断程序，执行完成后再继续执行被打断的程序。

注意：中断程序有可能随时发生，故用户在中断程序中需要对使用的变量和主程序或其他程序中的变量进行仔细区分，防止变量使用重叠造成程序运行异常。

● 其他功能

除上述的程序组成部分外，PAC 实时内核的硬件或其它配置信息也是程序正常运行必不可少的部分，使用时可根据实际情况进行选择配置。

(2) 运行机制

PAC 实时内核简化了扫描周期的工作，其主要工作内容如图 4-3-21 所示。

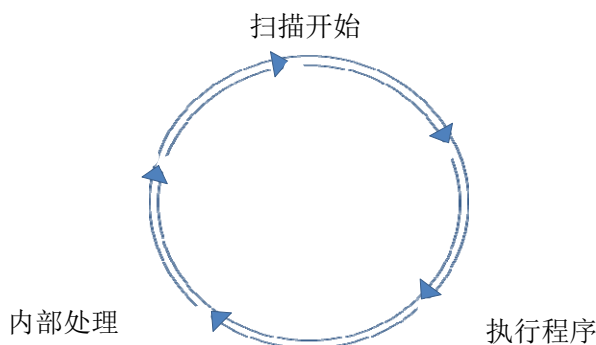


图4-3-21 扫描周期

● 执行程序：

执行程序就是控制器从用户程序区读取指令，译码执行。从第一条指令开始，直到最后一条指令结束。在程序执行过程中，如果遇到立即输入输出指令则直接对物理量进行刷新，否则需要等到周期结束后进行输出刷新。

程序中调用的中断程序或子程序不会在扫描周期中自动执行，只有当中断事件触发或子程序调用时才被执行。

● 内部处理：

控制器每执行一条指令都对自身硬件、指令执行和通信状态进行检测，当发现问题时做出相应处理，并给出故障信息。

控制器在 SM 区提供了程序执行时间信息，如表 4-3-9 所示。用户根据需要进行查询。

表 4-3-9 程序运行扫描时间

SM 区	数据类型	功能说明	单位
SMD8	无符号四字节	上一次用户程序扫描执行时间	ms
SMD10	无符号四字节	用户程序扫描执行的最短时间	ms
SMD12	无符号四字节	用户程序扫描执行的最长时间	ms

(3) 程序存储和加密

用户可以利用编程软件上载和下载用户程序，用户程序保存在 PAC 实时内核内部单独的程序存储区中，可永久保存。

为保证用户程序的安全，PAC 实时内核支持程序加密功能。在下载完程序后，用户只需要把 SM0.8 置位为 1，即可完成对用户程序的加密。加密后用户无法进行程序上载，但是可以下载新的程序，重新下载程序后加密功能自动解除，如果加密需要可再次置位 SM0.8。

4.3.5 编程

PAC 实时内核支持的用户程序程序主要分为主程序和子程序两部分，主程序在 PAC 实时内核执行期间不间断的循环执行，而子程序仅在被调用时才会执行。用户程序一般由一个主程序和若干个子程序组成。

PLC_Config 软件为用户提供了主程序和子程序操作、子程序参数编辑、网络操作、指令编辑、指令参数编辑、连线操作、功能块梯形图互转，以及查找、替换、快捷键等操作，使用简洁，方便用户进行程序开发。详细信息可查看《PLC_Config 软件编程手册》。

(1) 编程语言

● 梯形图语言 LAD

梯形图与电气连接图类似，允许程序仿真来自电源的电流通过一系列的逻辑输入条件，决定是否使能逻辑输出。一个梯形图程序左侧一定包含提供能流的能量线。其中闭合的触点，允许能量通过它们流到下一个元素，而打开的触点阻止能量的流动，如图 4-3-22 所示。该语言也是很多 PLC 程序员和维护人员首选的编程方法。

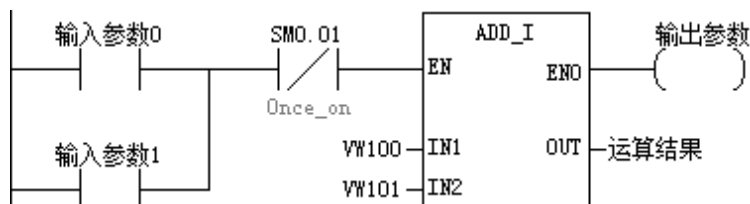


图 4-3-22 梯形图编程语言

其中符号意义如下：

—| |— 触点，代表逻辑输入条件，如开关、按钮、内部条件等。

—() 线圈，代表逻辑输出结果，如灯、马达启动器、内部输出条件等。

□ 块指令，具有一定算法能力的处理单元，例如定时器、计数器、数学运算指令等。

● 功能块语言 FBD

功能块用矩形块来表示，除特殊功能块外，每一个功能块的左侧应至少有一个输入端，在右侧应至少有一个输出端。功能块的类型名称通常写在块内的上部，输入输出参数的名称写在块内的输入输出点的相应地方。与梯形图编程语言的区别在于功能块是完全基于块指令的编程方式，功能块有自己的连线方式，不会像梯形图一样出现相交的情况。

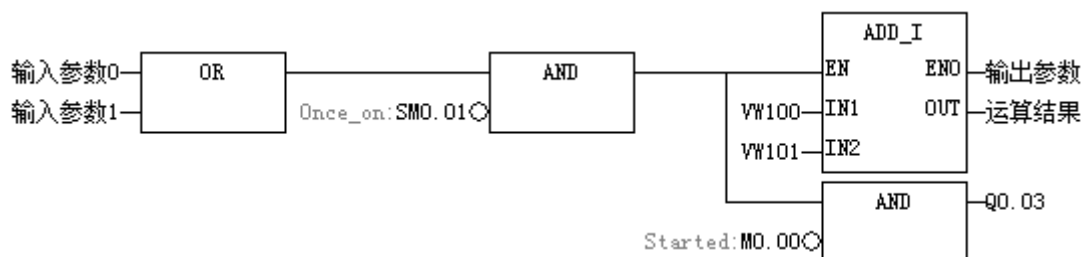


图 4-3-23 功能块编程语言

编程时用户可使用绝对地址或符号变量两种方式填写指令参数。在未填写任何参数时，指令的引脚会显示成默认状态

- << 能流输入，可输入位变量、位类型符号或连线，可以取反；
- >> 能流输出，可输入位变量、位类型符号或连线；
- >| 能流输出，与>>不同的是，该类型的参数可以不填写；
- ???.? 位变量，可输入位变量地址、位类型符号；
- ???? 字或双字变量，可输入字、双字类型的变量或符号。
- ** 系统自动分配的位变量。
- **** 系统自动分配的字、双字或浮点类型的变量。

(2) 编程步骤

PLC_Config 软件直接支持对 iTC1015 设备类型的管理和编程，用户可通过以下步骤实现对 iTC1015 的程序开发：

- 启动 PLC_Config，创建或打开已有工程
- 扫描并添加 PAC 设备；
- 扫描 IO 设备；
- 定义符号变量；

- 编写主程序，可创建并调用子程序；
- 完成编辑后，点击编译，修正编译错误；
- 下载程序，通过设备实际运行或程序仿真对程序进行调试；
- 完成编程工作。

(3) 监控

可以通过 PLC_Config 对程序、设备状态和变量进行监控，PLC_Config 提供监控功能如下：

- 程序监控，程序运行界面启动监控功能，可观察到各指令执行情况，还可以在监控状态，通过强制写入功能对程序进行调试；
- 状态符号表监控，在状态符号表界面点击监控，可观察到变量的当前值或改变当前值；
- 查看 SM 区当前状态，通过 PLC_Config 查看设备当前的各种运行状态；
- 内存表监控，以设备的整个变量区为单位进行监控，支持位、字、双字、浮点和字符串等多种显示格式。

(4) 仿真

仿真是 PLC_Config 提供的一项离线程序开发功能，通过仿真功能，用户可以在没有实际设备的情况下开发程序，进行模拟运行和调试。

启动仿真后，系统会弹出一个类似设备灯板的界面，显示出仿真设备当前的 I、Q、AI 和 AQ 状态，用户可实时的修改仿真设备的输入输出值，从而模拟真实的外部信号输入，达到仿真运行的效果，并完成对程序进行调试。

仿真运行期间，用户可以进行组态程序监控、状态符号表监控、实时曲线监控、变量区监控、运动控制运行监控等操作，还可以控制程序的运行、停止、首次扫描、多次扫描等操作。这些操作不影响真实 PAC 设备，无论 PAC 设备是否在线，用户都可以通过仿真运行功能调试组态程序。

(5) 轴参数设置

iTC1015 作为可编程运动控制设备，在使用运动控制类指令前需要对轴或轴组参数进行配置，使控制程序中的控制信息与实际被控对象进行关联。配置完参数后 PAC 实时内核会重新运行用户程序，轴参数也会立即生效。

用户应根据工程的需要，进行轴和轴组的数量配置。程序会根据配置的轴和轴组的数量信息，生成对应数量的详细配置界面。目前支持的轴个数为 0~24 个，轴组个数为 0~2 个，每个轴组中最多允许有 4 个轴。轴的参数配置包括轴的单位、电子齿轮比、最大速度、回原点信息等。详见表 4-3-10 轴参数一览表。

表 4-3-10 轴参数一览表

参数名称	功能	设定范围	初始值
轴编号	轴编号用于表示轴的 ID	0~3	根据选择轴的数量编号依次递增

是否使用	是否使用选项能够使能和禁能该轴	TRUE, FALSE	TRUE
单位	位移参数的单位, 以 units 表示	脉冲, 毫米, 微米, 纳米, 度, 英尺	脉冲
伺服单周脉冲	伺服电机旋转 1 周的脉冲个数	$1-1.8 \times 10^{19}$	10000 脉冲
伺服单周位移	伺服电机旋转 1 周的机构位移量	$0.000000001-1.8 \times 10^{19}$	10000units
最大速度	设置轴的最高速度, 需为正实数	正实数	400000000 units/s
点动最大速度	单轴点动的最大速度, 需为正实数	正实数	1000000 units/s
最大加速度	最大加速度。置 0 时没有加速度限制, 需为 ≥ 0 的实数	≥ 0 的实数	$1000000 \text{ units/s}^2$
最大减速度	最大减速度。置 0 时没有减速度限制, 需为 ≥ 0 的实数	≥ 0 的实数	$1000000 \text{ units/s}^2$
正向位置软件限制	软件限制的轴运行的正向最大位置	实数	4000000000 units
负向位置软件限制	负向位置软件限制, 需为实数	实数	-4000000000 units
回原点动作模式	回原点动作模式设置, 回原点指令 MC_Home 执行前, 需在此处设置回原点动作模式	接近信号 OFF (反转), 接近信号 ON (反转), 接近信号 OFF, 接近信号 ON, 限位信号 OFF, 原点信号屏蔽, 只用限位信号, 只用原点信号, 无接近信号	接近信号 OFF (反转)
原点输入信号类型	选择原点输入信号的类型, Z 相: 编码器的循环脉冲输出, 外部输入: 触发器触发信号	Z 相 外部输入	Z 相
回原点开始方向	回原点运动的起始运动方向	正向 反向	正向
原点搜索方向	到达原点位置前一时刻的轴的运动方向	正向 反向	正向
回原点速度	回原点速度, 回原点过程开始运动的速度	正实数	10000 units
接近原点速度	接近原点时的速度, 小于回原点速度	正实数	1000 units
回原点加速度	回原点加速度	正实数	0
回原点减速度	回原点减速度	正实数	0
回原点加加速度	回原点加加速度	正实数	0
原点检测屏蔽距离	原点检测屏蔽距离, 回原点指令的原点信号屏蔽模式中的原点检测屏蔽距离, 详细用法见 MC_Home 指令说明中的原点信号屏蔽模式	正实数	10000 units
原点附近输入信号	原点附近输入信号为当轴运动到接	I00-I15	I00

	近原点位置时的输入信号，指定原点附近输入信号端子号码（DI）		
原点附近输入信号种类	常态下原点附近输入信号种类	常开/常闭	常开
原点输入信号	原点输入信号为当轴运动到达原点位置时的输入信号，指定原点输入信号端子号码（DI）	I00-I15	I01
原点输入信号种类	常态下原点输入信号种类	常开/常闭	常开
正方向限位输入信号	正方向限位输入信号为当轴运动到正方向最大位置时的输入信号，指定正方向限位输入信号端子号码（DI）	I00-I15	I02
正方向限位输入信号种类	常态下正方向限位输入信号种类	常开/常闭	常开
负方向限位输入信号	负方向限位输入信号为当轴运动到负方向最大位置时的输入信号，指定负方向限位输入信号端子号码（DI）	I00-I15	I03
负方向限位输入信号种类	常态下负方向限位输入信号种类	常开/常闭	常开

轴组的参数配置包括轴组编号、组成轴组的轴个数及其编号、轴组的运动参数限制、轴组选用的动力学模型等信息。详见表 4-3-11 轴组参数一览表。

表 4-3-11 轴组参数一览表

参数名称	功能	设定范围	初始值
轴组编号	轴组编号用于表示轴组的 ID	0-1	根据选择轴组的数量依次递增
是否使用	使能和禁能该轴组	TRUE, FALSE	FALSE
组成轴组的轴的个数	轴组中轴的个数（不可设置），根据组成轴组的轴自动生成。	0-4	组成轴组的轴的总数
组成轴组的轴	添加到轴组中的轴的编号	0-3	0,1,2,3
最大插补速度	轴组运动的最大速度	正实数	800000000 units/s
最大插补加速度	轴组运动的最大加速度	正实数	1000000 units/s ²
最大插补减速度	轴组运动的最大减速度	正实数	1000000 units/s ²
运动模型	提供轴组运动模型种类	直角坐标系，圆柱坐标系，四轴并联机器人，三轴并联机器人，三轴座椅，通用模型	直角坐标系

每一个运动控制指令的右侧输出引脚均有参数 Error 和 ErrorID，这两个参数用来提示当前

指令的错误状态和错误信息。当发现 Error 置为 TRUE 时，可以根据 ErrorID 查询表 4-3-12 找出错误的原因。

表 4-3-12 运动控制指令错误码一览表

错误类型	错误码	说明
轴参数错误	0x1300	轴 ID 不存在或不允许访问
	0x1301	轴类型错误
	0x1302	设定速度的最小值不可达
	0x1303	设定速度的最大值不可达
	0x1304	设定的最大加速度不可达
	0x1305	设定的最大减速度不可达
	0x1306	设定的加加（减减）速度不可达
	0x1307	设定的速度-位置延迟不正确
	0x1308	重载速度不正确
	0x1309	设定主从轴间速度因子不正确
	0x130a	设定的运动方向错误
	0x130b	轴处于工作状态
	0x130c	轴未初始化
	0x130e	同步运行时，两轴的方向不一致
	0x130f	同步运行时，轴的状态错误
	0x1310	轴状态为禁止
PLC 参数错误	0x1500	轴 ID 不存在或不允许访问
	0x1501	位置设置错误
	0x1502	加速度设置错误
	0x1503	减速度设置错误
	0x1504	加加速度设置错误
	0x1505	速度设置错误
	0x1508	时间-位置对设置错误
	0x1509	时间-速度对设置错误
	0x150a	时间-加速度对设置错误
	0x150b	时间-位置，时间-速度，时间-加速度数组维数错误：超过最大允许维数或小于等于 0
参数错误	0x1701	相位偏移时参数设置错误
	0x1703	设置方向错误
	0x1704	设置样条参数失败
	0x1705	设置直线参数失败
轴组参数错误	0x1900	轴组 ID 不存在或不允许访问
	0x1901	轴组内该索引处无轴 ID
	0x1902	轴组已空
模型设置错误	0x2000	模型不存在或未设定
	0x2001	模型参数错误
	0x2002	模型类型与所需轴个数不匹配错误

提示：关于运动控制编程的详细说明参见《DCCE 编程手册 运动控制指令篇》。

4.4 网络通信服务

PAC 实时内核提供 3 路以太网接口，其中 1 号以太网接口支持标准的 UDP/TCP 自由通信协议、EPA 通信协议和 ModbusTCP/ModbusUDP 通信协议；其他两个以太网接口用于与 IO 设备通信。

EPA 通信端口号默认为 11000，ModbusTCP 端口号为 502~506，ModbusUDP 端口号为 507~510 和 11000，以太网接口支持支持 10/100/1000M 自适应全双工通信模式。

4.4.1 以太网自由通信

以太网自由通信报文带有 UDP 或 TCP 报文头，数据段用户自定义。PAC 实时内核使用 UDP 和 TCP 自由通信指令实现以太网自由通信，调用自由通信指令时，通过指令引脚配置目的 IP 和目的端口号等参数。

● UDP 自由通信

PAC 实时内核提供的 UDP 自由通信指令包括：NXMT_RCV，NET_XMT。通过上述指令可实现同网段或跨网段的 UDP 通信。

➤ 同网段 UDP 通信

当使用 UDP 自由通信指令进行同一网段内的 UDP 通信时，必须保证进行通信的设备具有相同的网络号。例如：设备 A 的 IP 地址网络为 192.168.8.123，则与其通信设备的 IP 地址应设置为 192.168.8.xxx（IP 尾号范围为 1~254）。

➤ 跨网段 UDP 通信

当使用 UDP 自由通信指令进行跨网段通信时，需要对网关进行配置，以图 4-4-1 中的网络结构为例，A 设备通过路由器可以跨网关向 B 设备和 PC 机发送数据报文，实现跨网段通信。

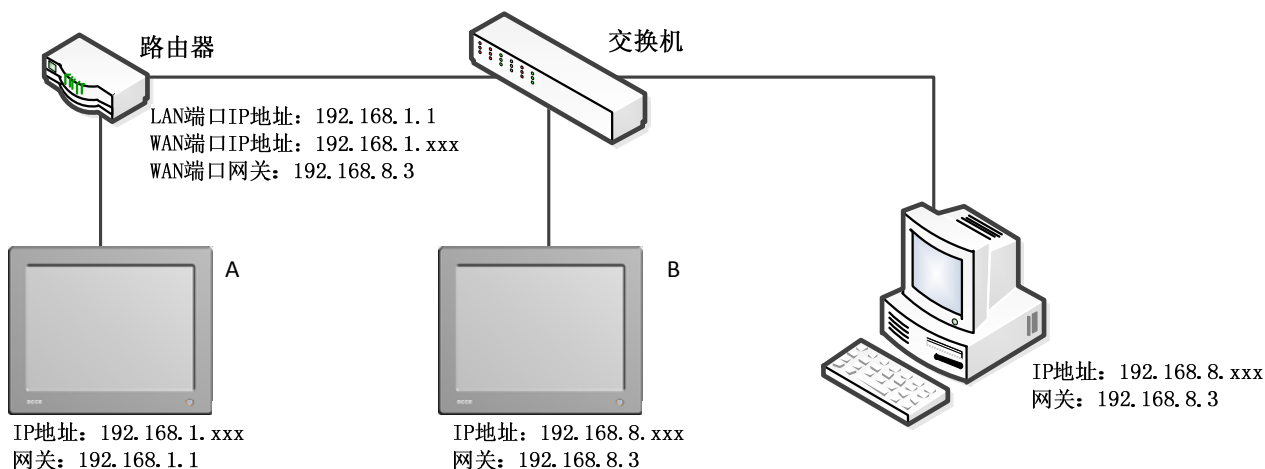


图 4-4-1 跨网段通信

4.4.2 EPA 协议通信

PAC 实时内核利用标准的 EPA 协议实现与 PLC_Config 编程软件的通信，完成内部参数配置、程序上下载和变量监控等功能。

● EAP 报文

EPA 报文头由 7 部分组成，表 4-4-1 描述了各部分含义。PAC 实时内核提供了读写变量区的报文格式，可利用 EPA 变量读写报文访问各变量区，实现对 PAC 实时内核的监控。

表 4-4-1 EPA 报文头

参数名	数据类型	编码位置偏离 (单位：字节)	字节长度	描述
ServiceID	Unsigned8	0	1	报文类别码
Reserved	Ocetstring	1	3	保留
Length	Unsigned16	4	2	报文长度
MessageID	Unsigned16	6	2	消息号
SourceAppID	Unsigned16	8	2	源功能块标识
DestinationAppID	Unsigned16	10	2	变量区标识
DestinationObjectID	Unsigned16	12	2	目的对象标识
SubIndex	Unsigned16	14	2	对象的子索引

● 变量区读写

通过 EPA 协议可访问 18 个变量区，每个变量区都支持单个变量、多个变量和区变量（512 个字节）的读写操作，对应的变量区标识如表 4-4-2 所示。

表4-4-2 变量区标识

变量区标识	区域	变量区标识	区域
101	I 区	110	AI 区
102	Q 区	111	AQ 区
103	V 区	112	XI 区
104	L 区	113	XQ 区
105	M 区	114	XAI 区
106	S 区	115	XAQ 区
107	SM 区	116	PAI 区
108	T 区	117	PAQ 区
109	C 区	119	P 区

详细协议组织格式可参见《DCCE 网络化可编程控制器用户编程手册》。

4.4.3 ModbusTCP/ModbusUDP 通信

PAC 实时内核支持 ModbusTCP/ModbusUDP 通信协议。进行 TCP 方式通信时，通信设备间需要建立 TCP 连接，PAC 实时内核可接收连接请求，对接收的报文进行相应的处理；使用 UDP 方式通信时，不需要建立连接，可接收报文或通过指令主动发送报文，实现与设备之间的通信，端口划分可参见表 4-4-3。

表4-4-3 Modbus网络协议端口说明

协议	端口号
ModbusTCP	502
	503
	504
	505
	506
ModbusUDP	507
	508
	509
	510
	11000

4.5 系统管理

4.5.1 PAC 控制系统

PAC 实时内核可通过 EtherCAT 协议连接扩展 IO 设备，实现现场数据的分布采集与集中管理，并通过一款高达 1.8GHz 的高速处理器确保现场控制的实时性，其分控制系统组成如图 4-5-1 所示。

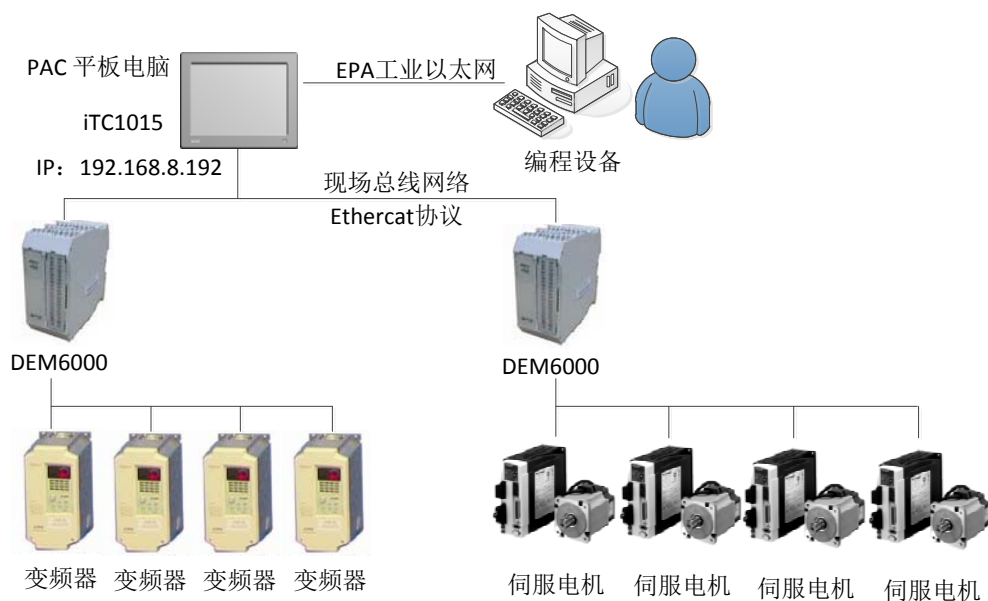


图 4-5-1 分布式控制系统

PAC 控制系统由编程设备、PAC 和扩展 IO 设备 3 部分组成，用户通过编程设备对系统中的 PAC 实时内核进行操作和管理，PAC 利用 1.8GHz 的中央处理器对用户程序进行高速的运算和处理，并通过 EtherCAT 协议与现场 IO 设备进行数据同步，同步周期可达 1ms，同步的过程分为两步：将 PAC 的 Q 信息同步到 IO 设备中；将 IO 设备的 I 信息同步到 PAC 中。两个步骤在一次通信过程中完成，确保通信效率。

作为控制系统的核心，PAC 实时内核可最多扩展 256 个 IO 设备，实现上万点的 IO 映射，基于对主流控制网络的良好兼容性和集中控制特点，可满足大中小型控制系统的通用控制需求。

4.5.2 指令网络通信

指令通信网络系统分为以太网和串行网络，分别通过以太网读写指令和串行读写指令实现变量的映射，通信校验和同步运行等功能。

（1）变量映射

● 以太网变量读写

PAC 实时内核支持 12 种以太网读写指令，用于多个以太网对等控制器之间的变量映射和系统的协同控制，例如，PAC 实时内核可通过以太网字读指令获取指定 IP 地址的设备变量信息。

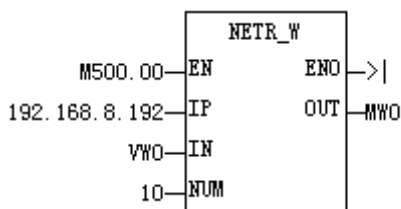


图 4-5-2 以太网字读指令

如图 4-5-2 所示，PAC 实时内核运行以太网字读指令读取 IP 地址为 192.168.8.192 的控制器的 VW0~VW9。读取后将其放入到本地的 MW0~MW9 中，供 PAC 实时内核使用。

● 串行变量读写

PAC 实时内核支持 4 种串行读写指令，用于通过串行通信实现设备间的变量映射。例如，PAC 实时内核可通过串行字读指令获取指定 Modbus 站地址的设备变量信息。

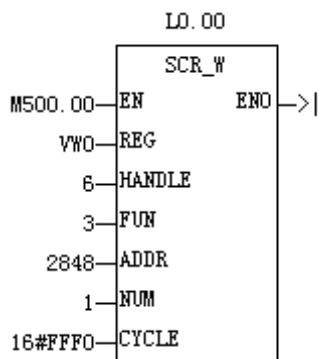


图 4-5-3 串行字读指令

如图 5-4-3 所示，PAC 实时内核运行串行字读指令读取 Modbus 站地址为 6 的串行设备的一个 Modbus 地址为 2848 的字变量，并将读取到的数据放入本地的 VW0 中，供 PAC 实时内核使用。

(2) 通信校验

用户可根据实际情况自行设计通信校验方法，以防止错误报文的处理，影响系统正常工作。例如，发送设备可修改发送报文，在报文的起始和结束处添加检测字符，接收设备根据接收报文的起始字符与结束字符判断是否为所需报文，若为所需报文，对其进行相应的处理。此外，PAC 实时内核还提供了数据校验码生成和检查指令，发送设备可通过数据校验码生成指令对发送数据进行校验，生成校验码重新打包后再发送，接收设备可通过数据校验码检查指令，判断接收报文是否正确，若正确对其进行相应的处理。