

Koyo

Value & Technology

可编程序控制器 **NK0 系列**
用户手册

[第二版]

光洋电子(无锡)有限公司

前 言

此次承蒙采用本公司的 NK0 系列可程序控制器（PLC），表示衷心的感谢！在使用 NK0 系列 PLC 之前，请仔细阅读本手册。

本手册较为详细地介绍了 NK0 系列 PLC 的系统构成、性能、规格、外形尺寸、安装设置、运行准备、维护检修等方面的知识，为用户熟悉并应用该产品提供一个必需的资料。

NK0系列PLC是基于NK1系列PLC而开发的一款微型整体式PLC。其外形小巧（大小为NK1系列PLC扩展单元一样），适合窄空间、小面积安装环境下的安装使用。

NK0为整体式不可扩展系列PLC，根据其所带输入/输出点，分基本型和模拟量型2种机型。基本型根据所带输入/输出点数分16点机型、28点机型和32点机型，根据输出点类型的不同又分继电器输出型、晶体管输出型2种。模拟量型除带开关量输入/输出点外，还带一定数量的模拟量点，以拓宽NK0系列PLC的使用范围。模拟量型根据所带模拟量通道数和电流/电压类型的不同，又细分出多个机种。这样，NK0系列PLC提供丰富的机种系列，方便客户灵活选用。

NK0系列PLC向用户提供了采用传统的梯形图逻辑方法以及光洋特有的级式编程方法对一个控制系统进行开发的能力，其指令体系基本继承了NK1系列PLC的指令集（仅10多条和中断、日历时钟、运动控制、有关的指令不支持），所使用的编程工具软件也是和NK1同样的KPPsoft。这样，需要时，用户可以方便借用原来NK1的程序资源，以缩短程序编制时间，从而大幅缩短工程开发周期。

NK0系列PLC外形小巧，点数简少，但指令丰富，可广泛应用于轻工、纺织、塑料、食品、包装、金属制品、楼宇自动化、健康产业等各行业中的小型生产机械、楼宇设备、简易机床、康疗设备的控制系统中。

另外，NK0系列PLC非常适合作为智能化工厂的一个现场节点来参与智能工厂的建设。

NK0 系列 PLC 的主要特点如下：

- 微型整体式结构，适合窄空间安装使用；
- 提供基本型和模拟量型多种机型，方便客户选用；
- 全机型带 RS-485 串行通信口，方便实现简易网络化管理；
- NK0-CPU28DR-D 机型带有以太网通信口，方便实现高速 IOT 管理；
- 编程工具，指令体系兼容 NK1 系列，软件资料可沿用；
- 级式编程与梯形图编程方式共用，使程序流程更简洁、明了；
- 有较快的处理速度，单条逻辑指令执行时间最快 0.1 μ S；
- 有循环控制、子程序调用等程序控制指令和包括浮点运算、指数/对数运算、三角函数运算在内的丰富的数据处理指令；
- 8 位密码保护功能，更提供 PLC 内程序锁定功能，可有效保护程序资源不受侵害；
- 提供恢复出厂设置功能，方便客户维护。

如果你在阅读本手册或使用 NK0 系列 PLC 时有什么疑问，或你需要另外的信息时，请与本公司本部或驻各地办事处联系，以便尽快得到服务。

由于产品的改进等原因，本资料所刊内容会与实际的产品有些差别，请注意！

本公司保留对包括此资料在内的所有信息的专利权！

如果你有有关本手册的情况需要与我们联系，请首先确定手册的版本号！

资料名称：《NK0 系列 PLC 用户手册》

资料编号	编制日期	内容说明
KEW-M2611A	2020 年 10 月	原稿第一版
KEW-M2611B	2023 年 3 月	输入 ON 电压改成 DC17V 以上 增加 NK0-CPU28DR-D 机型

有关 NK0 的资料，除本手册资料外，还有以下资料：

《NK0/NK1 编程手册》：对 NK0/NK1 所有指令功能的说明；

《KPPSoft 工具软件入门手册》：程序编制、调试工具软件 KPPSoft 的使用说明；

《级式语言编程指导》：光洋级式编程方式的说明资料；

在阅读本手册时，如果需要，请一并参考以上技术资料。

注 意 事 项

使用安全上的注意事项

[使用环境·条件]

- 请不要在具有可燃性气体、爆炸性气体的地方安装、使用本 PLC，否则有可能引发人身事故或火灾。
- 在有关人身安全的用途中使用本产品请特别注意：设计时，请考虑即使万一本产品发生故障或误动作时，也不会危及人身的安全。为了安全，请在有可能发生机械损坏、事故等的部分，在外部设置联锁保护回路。
- 请在规格规定的使用环境（振动、冲击、温度、湿度等）范围内保存、使用本产品。
- 请在对本产品有一定了解的基础上使用本产品。

[安装·配线]

- 使用本产品时，请注意不要误配线。否则，有可能引发火灾或损坏本产品；
- NK0 系列产品仅提供 DC24V 直流电源工作机型，请在手册规定的电源电压范围内使用本产品。否则，容易引发火灾、电击或发生故障；
- 请按照手册的规定进行设置、配线。否则，有可能引发火灾或发生故障；
- 请在断开电源的状态下进行配线。否则，有可能发生电击或故障。

[其他注意事项]

- 请不要使本产品跌落或受直接冲击力；
- 不要让导线头、金属片等异物进入本产品；
- 端子螺丝的紧固请按规定的力矩进行；
- 报废后的 PLC 产品本体、扩展单元、电池等部品包含有一定数量的电子、塑料、金属等部件，这些部件可能包含有对水、土壤、大气等环境产生一定影响的物质。为了保护环境质量，请您按国家环境保护法律、法规规定以及所在地政府部门有关危险废弃物处理规定妥善处理报废部件。

关于产品的质保期和质保范围

[产品质保期]

本产品的质保期为用户购买后的一年间。

[质保范围]

在质保期内由于产品本身的质量问题或本公司的原因而引起产品故障的，本公司负责质保修理或质保调换本产品。

但是，由于以下原因而引起产品故障的，不属于本质保范围。

- 由于用户不正当的安装、使用而引起的问题；
- 故障是由于本产品以外的原因引起的；
- 用户自行拆开、改造、修理过的产品；
- 其他由于用户本人的责任引起问题的场合；
- 由于天灾、人祸及其他不可预测的原因而引起的问题。

另外，这儿所承诺的质保，是针对本公司所售出产品的。对于由此而引发的其他损害，本公司恕不承担任何责任。

目 录

第一章	NK0 系列 PLC 系统构成	1
1-1	概要.....	1
1-2	NK0 系列 PLC 型号构成.....	2
1-3	NK0 系列 PLC 系统构成.....	3
1-3-1	NK0 系列 PLC 基本系统构成.....	3
1-3-2	NK0 系列 PLC 网络构成.....	4
1-4	NK0 系列 PLC 外围设备.....	7
1-5	NK0 系列 PLC 外形尺寸图.....	10
第二章	NK0 系列 PLC 系统规格	11
2-1	NK0 系列 PLC 一般规格.....	11
2-2	NK0 系列 PLC 性能规格.....	12
2-3	NK0 系列 PLC 开关量 I/O 性能规格.....	14
2-3-1	汇点（Sink）和源（Soure）的概念.....	14
2-3-2	I/O 公共端的概念.....	15
2-3-3	DC 型 I/O 点与现场晶体管设备的连接.....	15
2-3-4	继电器输出点与现场设备的连接.....	16
2-3-5	DC 输入规格.....	19
2-3-6	继电器输出规格.....	21
2-3-7	NPN 晶体管输出规格.....	22
2-3-8	NK0 系列各型号 PLC I/O 接线示意图例.....	24
2-4	NK0 系列 PLC 模拟量 I/O 性能规格.....	27
2-4-1	NK0 系列 PLC 模拟量输入规格.....	27
2-4-2	NK0 系列 PLC 模拟量输出规格.....	28
2-4-3	NK0 系列 PLC 模拟量接线示意图例.....	29
第三章	NK0 系列 PLC 性能和操作	30
3-1	NK0 系列 PLC 硬件构成.....	30
3-2	NK0 系列 PLC 的 I/O 识别和定义号分配.....	35
3-2-1	NK0 系列 PLC I/O 的识别.....	35
3-2-2	NK0 系列 PLC I/O 定义号的分配.....	35
3-3	NK0 系列 PLC 功能存储器.....	36
3-3-1	NK0 系列 PLC 功能存储器一览表.....	37
3-3-2	NK0 系列 PLC 特殊线圈.....	39
3-3-3	NK0 系列 PLC 特殊寄存器.....	41
3-3-4	FlashROM 型特殊寄存器.....	42
3-3-5	模拟量型 NK0 系列 PLC 模拟量数据的存取.....	42
3-4	NK0 系列 PLC 用户存储器.....	43
3-4-1	程序存储区.....	43
3-4-2	系统参数区.....	44
3-4-3	系统参数区和用户程序传送.....	47
3-4-4	程序注释.....	47

3-5	NK0 系列 PLC 扫描及输入输出传送.....	48
3-5-1	NK0 系列 PLC CPU 的执行处理	48
3-5-2	循环扫描	49
3-5-3	级式指令和扫描.....	50
3-5-4	输入输出传送.....	51
3-5-5	成批传送方式.....	51
3-5-6	直接输入/输出方式	52
3-6	NK0 系列 PLC CPU 动作模式.....	53
3-7	NK0 系列 PLC 特殊功能.....	55
3-7-1	在线程序调试功能	55
3-7-2	离线模拟运行功能	57
3-7-3	工程文件包生成/下载功能.....	58
3-7-4	自诊断功能.....	60
3-8-5	错误履历情报保存/显示功能	61
第四章	NK0 系列 PLC 通信功能.....	62
4-1	NK0 系列 PLC RS-485 通信端口规格及通信线连接	62
4-2	NK0 系列 PLC RS-485 通信端口支持的通信协议及参数设置方法	63
4-2-1	NK0 系列 PLC RS-485 通信端口支持的通信协议.....	63
4-2-2	NK0 系列 PLC RS-485 通信端口通信参数设置寄存器	63
4-2-3	NK0 系列 PLC RS-485 通信端口通信参数设置方法.....	68
4-2-4	NK0 系列 PLC RS-485 通信端口通信程序编制	70
4-3	NK0 系列 PLC K-协议/CCM2 (DIRECTNET) 协议通信的实现	72
4-4	NK0 系列 PLC MODBUS RTU 协议通信的实现.....	74
4-4-1	MODBUS RTU 协议通信参数设置.....	74
4-4-2	MODBUS RTU 协议通信子局功能实现	76
4-4-3	MODBUS RTU 协议主局通信通信程序编制.....	81
4-4-4	MODBUS RTU 协议专用通信指令 MRX/MWX	83
4-5	NK0 系列 PLC 无协议串行通信的实现	88
4-5-1	NK0 系列 PLC 支持无协议通信功能.....	88
4-5-2	NK0 系列 PLC 无协议通信参数设置.....	90
4-5-3	NK0 系列 PLC 无协议通信功能的实现.....	93
4-5-4	NK0 系列 PLC 无协议串行通信指令 (ASCII IN/OUT 功能)	98
4-6	NK0 系列 PLC 以太网通信 (仅 NK0-CPU28DR-D).....	103
4-6-1	NK0 系列 PLC 以太网通信端口规格及通信线连接	103
4-6-2	NK0 系列 PLC 以太网通信端口支持的通信协议.....	105
4-6-3	NK0 系列 PLC 以太网通信端口通信参数设置寄存器	105
4-6-4	NK0 系列 PLC 以太网通信端口通信参数设置.....	109
4-6-5	NK0 系列 PLC 以太网通信端口 MODBUS TCP 通信功能的实现.....	110
4-6-6	NK0 系列 PLC 以太网通信端口电子邮件发送功能的实现.....	111
4-7	NK0 系列 PLC 通信相关特殊线圈的利用	113
4-7-1	通信出错后的后续处理.....	113
4-7-2	通信出错情况的计次处理.....	114
4-7-3	多个通信程序段的互锁.....	114

第五章	NK0 系列 PLC 运行准备	115
5-1	运行步骤	115
5-2	NK0 系列 PLC 安装接线及检查	116
5-2-1	NK0 系列 PLC 安装尺寸及安装方法.....	116
5-2-2	NK0 系列 PLC 输入/输出接线端子盖板的装卸.....	119
5-2-3	NK0 系列 PLC 输入/输出端子台的装卸	120
5-2-4	NK0 系列 PLC 机器连线.....	121
5-2-5	NK0 系列 PLC 安装上的注意事项.....	123
5-2-6	NK0 系列 PLC 安装检查.....	123
5-3	NK0 系列 PLC 通电	123
5-4	程序编制	124
5-5	程序文法检查.....	124
5-6	系统参数的设定	126
5-7	试运行	127
5-8	程序保存	127
5-9	运行	127
第六章	NK0 系列 PLC 系统维护	128
6-1	硬件系统的维护	128
6-2	NK0 系列 PLC 状态 LED 显示	129
6-3	通信问题	129
6-4	I/O 问题的解决.....	129
6-5	电气干扰问题.....	130
6-6	NK0 系列 PLC 错误代码一览表	131
附录一	NK0 系列 PLC 指令语一览表	133
1.	顺序指令（63 种）	133
2.	双字比较顺序指令（12 种）	136
3.	控制程序执行指令（21 种）	137
4.	数据处理指令（159 种）	138
5.	新增数据处理指令(8 种).....	144
6.	特殊对象指令(7 种).....	144
7.	IBOX 指令(28 种).....	145
附录二	有关软件专利权的说明	146

第一章 NK0 系列 PLC 系统构成

1-1 概要

NK0 系列 PLC 为本公司开发的新一代简易式微型可程序控制器系列产品，该系列产品为整体式不可扩展 PLC 产品。根据其输入/输出的不同，分基本型和模拟量型 2 个大类，基本型主要是开关量输入/输出，根据所带输入/输出点数，分 16 点机型、28 点机型和 32 点机型；模拟量型除带有开关量输入/输出点外，还带有一定数量的模拟量通道，根据模拟量类型的不同（电流或电压），又细分几种机型，NK0 系列 PLC 详细机型列表见 1-2 节型号一览表。



NK0 系列 PLC 以开关量和简易模拟量（电流：4~20mA, 或电压：0—10V）控制为主，不支持高速计数、PID 调节、脉冲输出等复杂功能。

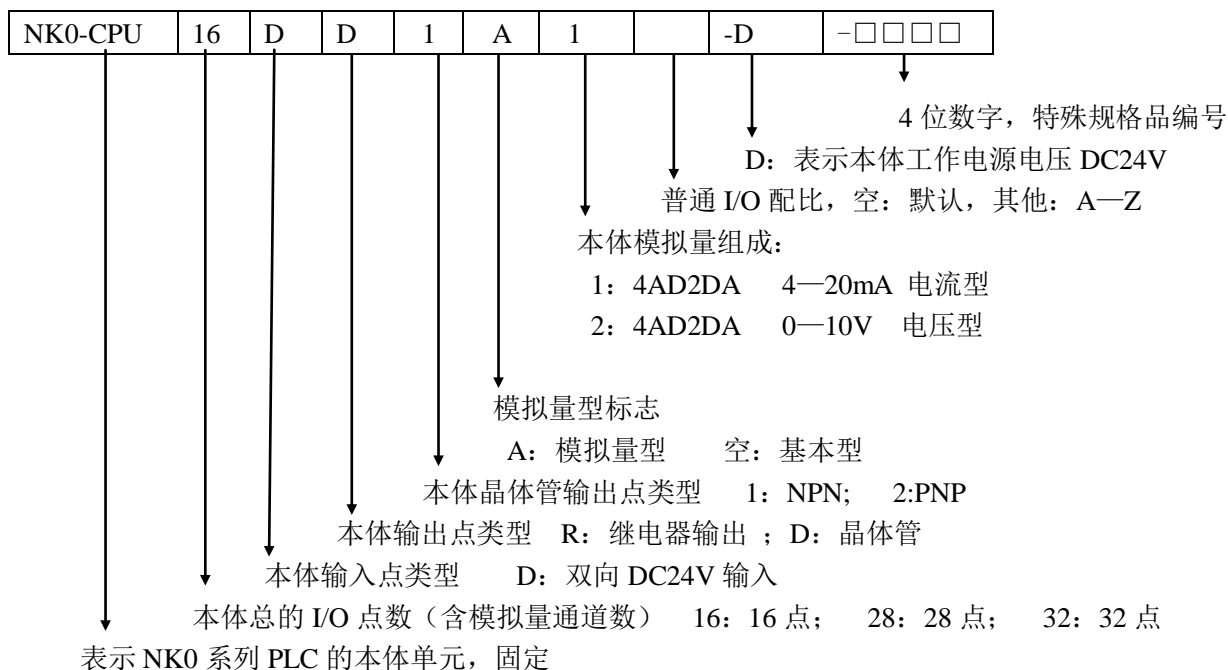
NK0 系列 PLC 本体带有一个“RUN”，“STOP” 2 位置。可以用户选择 PLC 的运行状态。

NK0 系列 PLC 带一个 3 端子台式 RS-485 串行通信口，支持 K 协议 (Slave)、CCM2 (Slave)、MODBUS RTU (Master/Slave) 协议、无协议 (Master/Slave)、K+协议 (KPPsoft 编程软件用) 通信。可以与支持这些通信协议的多种外围串行设备进行通信。另外，在需要时可组成一个 RS-485 串行通信网络。

NK0-CPU28DR-D 机型更带有一个以太网通信口，支持 MODBUS TCP 通信协议，并支持 SMTP、DHCP、HTTP、HAP+UDP 等通用以太网通信协议，方便 NK0 加入高速以太网网络。

1-2 NK0 系列 PLC 型号构成

NK0 系列 PLC 为微型整体式不可扩展 PLC。下面列出 NK0 系列 PLC 本体单元型号构成。



下表列出 NK0 系列 PLC 所有本体单元的型号一览表。注意，由于产品改进等原因，表中列出的产品可能会有变动。另外表中列出的产品，并不表示该型号产品可以现货提供，有关产品的具体购买事宜，请与本公司销售部门联系商洽。

种类	型号	规格	
基本型	晶体管输出	NK0-CPU16DD1-D	DC 输入 8 点/NPN 输出 8 点
		NK0-CPU32DD1-D	DC 输入 16 点/NPN 输出 16 点
	继电器输出	NK0-CPU16DR-D	DC 输入 8 点/继电器输出 8 点
		NK0-CPU28DR-D	DC 输入 16 点/继电器输出 12 点，带以太网口
		NK0-CPU32DR-D	DC 输入 16 点/继电器输出 16 点
模拟量型	晶体管输出	NK0-CPU16DD1A1-D	DC 输入 6 点/NPN 输出 4 点，4AD2DA 4-20mA 电流型
		NK0-CPU16DD1A2-D	DC 输入 6 点/NPN 输出 4 点，4AD2DA 0-10V 电压型
	继电器输出	NK0-CPU16DRA1-D	DC 输入 6 点/继电器输出 4 点，4AD2DA 4-20mA 电流型
		NK0-CPU16DRA2-D	DC 输入 6 点/继电器输出 4 点，4AD2DA 0-10V 电压型

注：所有 NK0 系列 PLC 都带有一个 RS-485 通信口。

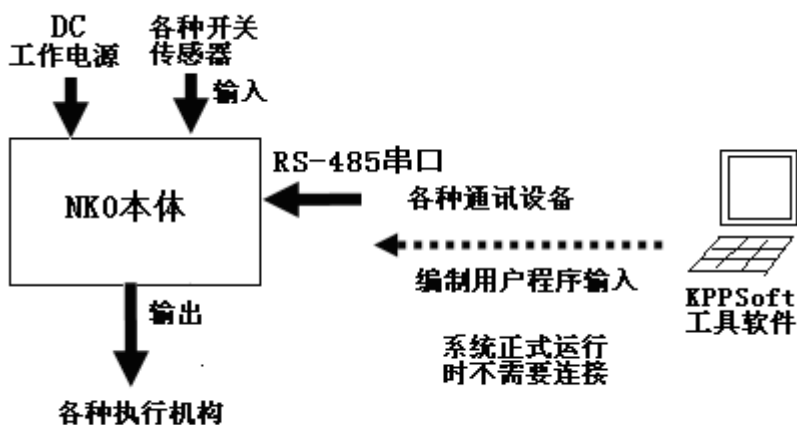
1-3 NK0 系列 PLC 系统构成

1-3-1 NK0 系列 PLC 基本系统构成

NK0 本体单元包含了构成一个 PLC 系统所需要的各种功能部件，包括 CPU，输入/输出接口，系统程序、用户程序（单独另外编制）、各功能单元存储器等等，单独的 NK0 本体单元即可构成独立的 PLC 控制系统。

NK0 系列 PLC 适用于窄空间、紧安装面积、少控制点数、主要为开关量控制的微型自动化控制系统（选择模拟量机型可以实现简单模拟量控制）。或者可以作为智慧工厂系统的一个现场控制节点控制器来使用。

根据应用需要，选择合适的 NK0 本体单元，输入用户程序后，就能实现需要的自动化控制功能。需要时，还可以通过 NK0 自带的 RS-485 通信口连接各种通信设备。NK0-CPU28DR-D 机型更带有以太网通信口，方便组成高速以太网网络。



下图为 NK0-CPU16DR-D，NK0-CPU32DR-D 的本体照片。



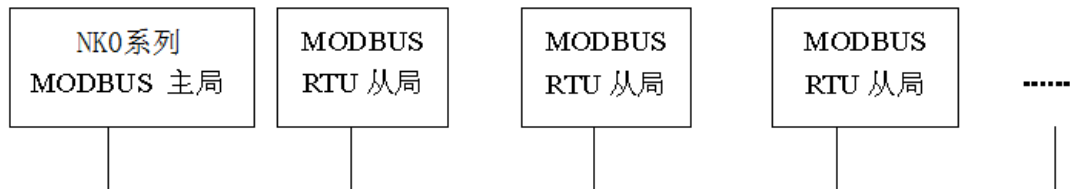
1-3-2 NK0 系列 PLC 网络构成

1、RS-485 通信口通信连接

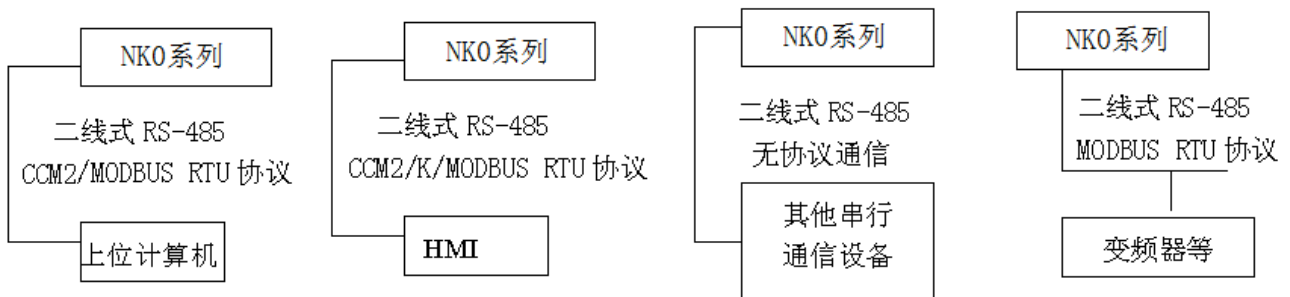
NK0 系列 PLC 本体带有一个 RS-485 通信口，该通信口为 3P 端子台二线式半双工通信口。通过它，可实现 CCM2 协议 (Slave)、K 协议 (Slave)、MODBUS RTU 协议 (Master/Slave)、无协议 (Master/Slave) 等通信协议方式的通信。



RS-485 MODBUS RTU 协议 最大 247 台（需要时加信号放大器）



RS-485 MODBUS RTU 协议 最大 32 台



RS-485 通信口 1 对 1 通信连接

注：NK0 系列 PLC 的 RS-485 通信口没有内置终端电阻，在使用 NK0 系列 PLC 组建 RS-485 通信网络时，需要时，用户可在 RS-485 网络 2 端的 A、B 接线端间各接入一个终端电阻，推荐终端电阻大小为 120 欧姆。

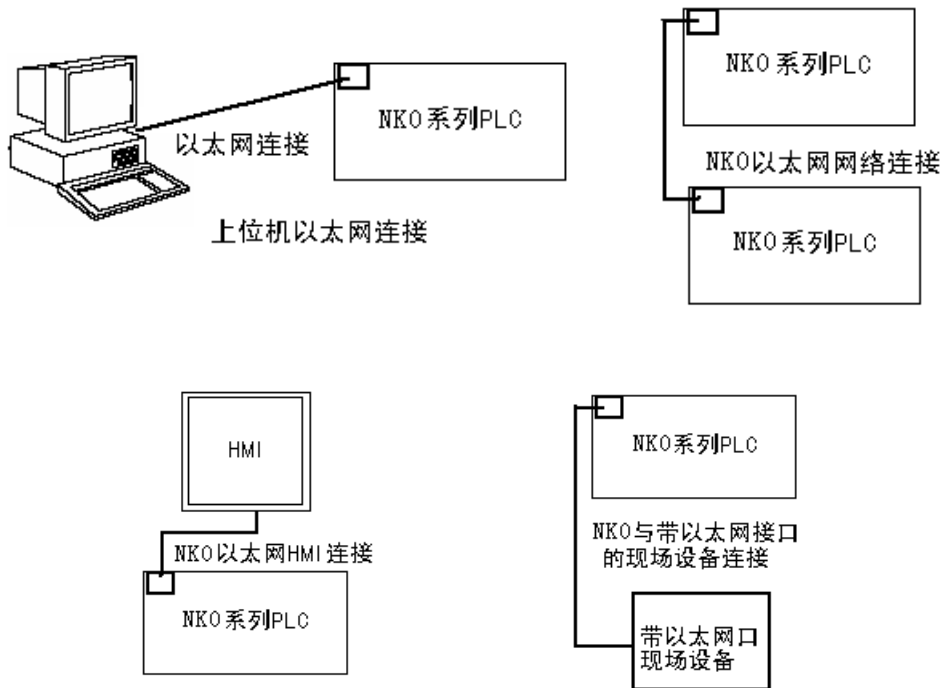
2、以太网通信口通信连接(仅 NK0-CPU28DR-D 机型)

NK0-CPU28DR-D 机型上带有一个 10M/100M 自适应标准以太网口，支持 MODBUS TCP 通信协议，还支持 SMTP、DHCP、HTTP、HAP+UDP 等通用以太网通信协议。通过 NK0 上的以太网口可以多台 NK0 构成网络或与带以太网口，并支持 MODBUS TCP 通信协议的 PLC 或现场设备进行快速以太网通信。

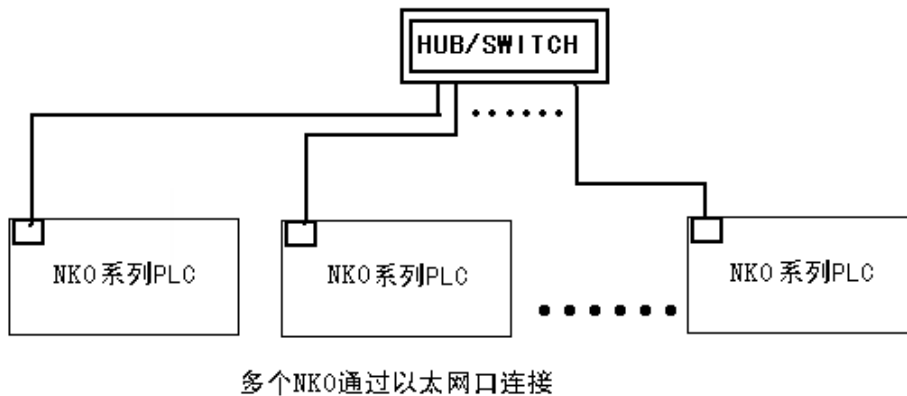
NK0-CPU28DR-D 的以太网通信口支持 MODBUS TCP 协议下的主局/从局通信，作为 MODBUS TCP 从局时最多支持同时建立 4 个通信连接（作为主局一般支持建立一个通信连接）。NK0 以太网口作为 MODBUS TCP 主局使用时的具体使用方法，请参见《KewNetEdit 工具软件资料》。

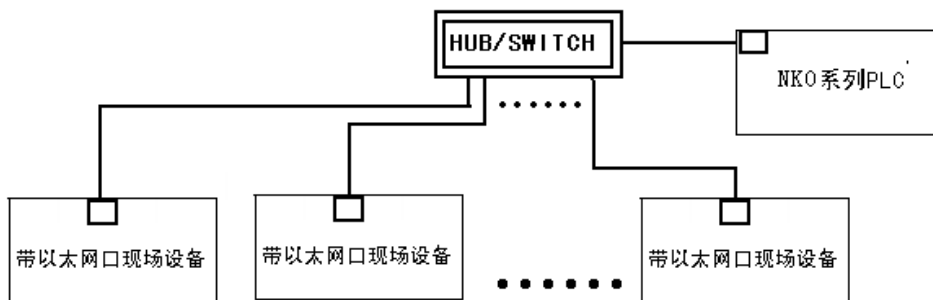
设备之间通过以太网联接的方法可以有很多种，下面介绍 NK0 系列 PLC 通过以太网口的几种常用网络联接方法。

1) NK0-CPU28DR-D 以太网 1 对 1 通信连接例如下：

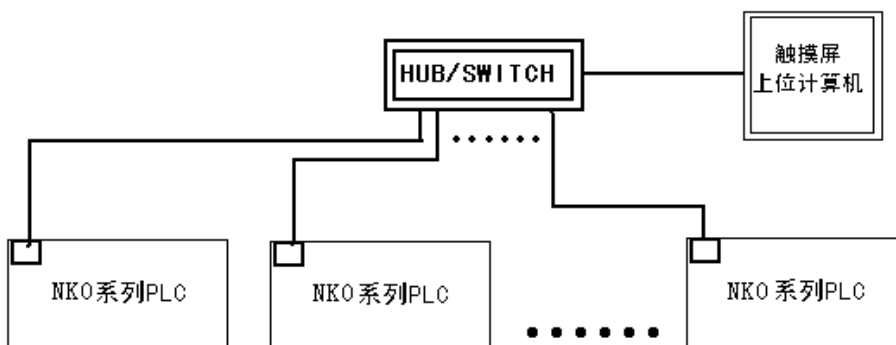


2) NK0-CPU28DR-D 通过以太网口实现多台 NK0 连接、1 对多、多对 1、多对多通信连接例如下：

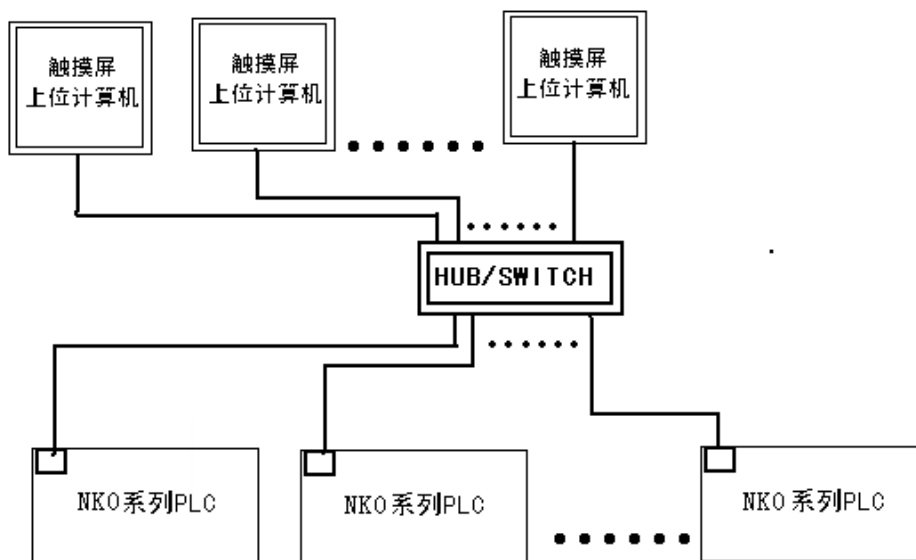




1台NK0带多台现场设备



1台上位计算机连接1台NK0



多台上位计算机与多台NK0相连接

1-4 NK0 系列 PLC 外围设备

NK0 本体带有的 RS-485 通信口，可以连接多种通信型外设，包括：

1、计算机通用编程软件 KPPSoft (Koyo PLC Programming Software)

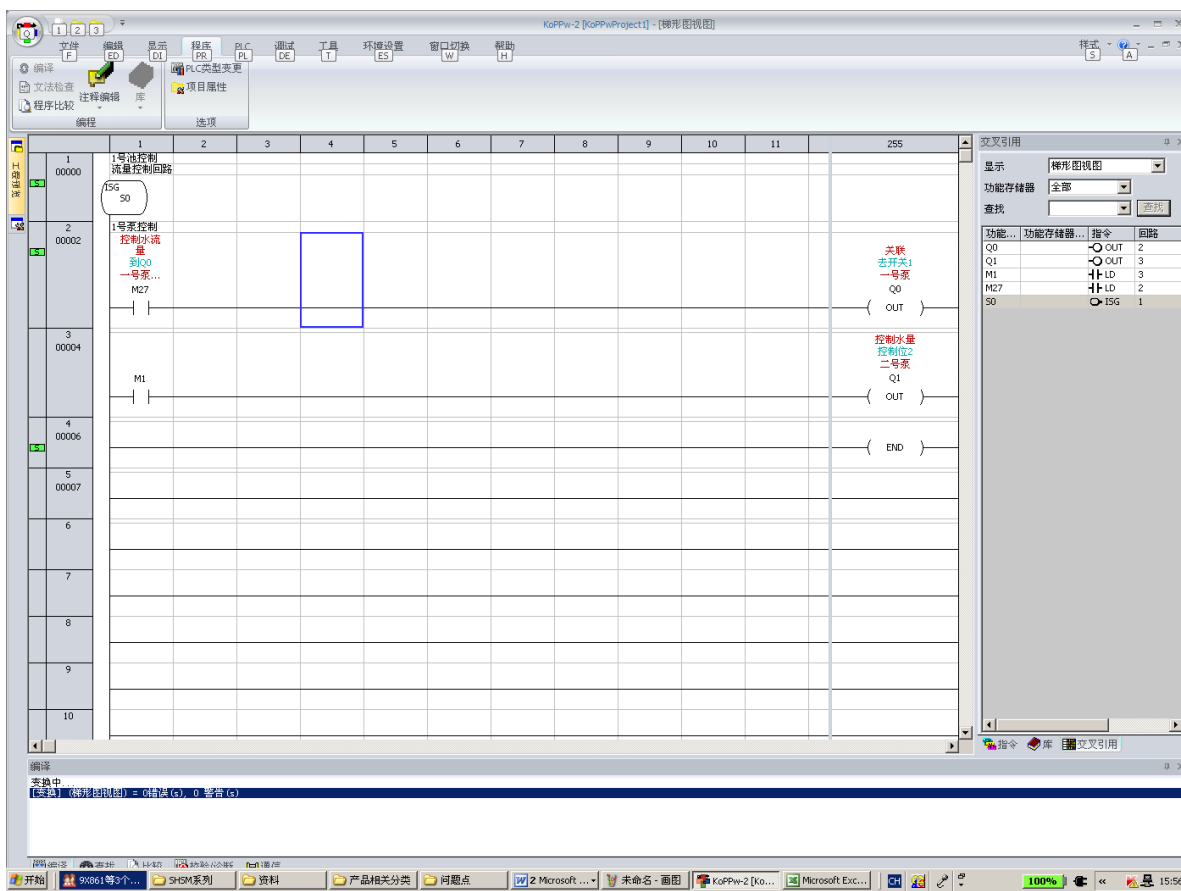
计算机编程软件安装于通用计算机上，通过 NK0 上的 RS-485 通信口与 NK0 系列 PLC 相连。通过 KPPSoft 计算机编程软件可以实现离线编程、离线模拟运行、在线编程、在线监控调试、使用状况报告、PLC 情况报告、PLC 参数设定、PLC 系统复制、编写注释说明文档、程序/文档资料打印、文件管理、环境设定等功能。

KPPSoft 是适用于 Windows 环境的编程软件，适用于 WIN XP, WIN Vista, WIN 7, WIN 8, WIN8.1, WIN10 等系统。

安装软件需要的内存空间：WIN XP：512MB 以上；WIN Vista：1GB 以上；其他系统：2GB 以上。

安装软件需要的硬盘容量：500MB 以上。

推荐的显示器分辨率：XGA 1028 × 768。



通用计算机编程工具软件 KPPSoft

NK0 系列 PLC 对应 KPPSoft 功能表

序号	功能	序号	功能
1	梯形图、指令语方式下的程序编辑（更改、追加、删除等）（离线方式下）	22	功能定义号说明文字的编辑、显示，包括：别名、接线信息、注释信息等 3 种
2	程序的梯形图显示	23	运行中程序的监视
3	程序的级式语言图形显示	24	密码口令编辑功能
4	程序的指令语显示	25	PLC 系统版本信息的显示
5	定义号交叉引用表显示	26	扫描时间的读出
6	程序的文法检查	27	停电保持区域的设置
7	指令、定义号的检索	28	WDOG 设置
8	定义号成批替换	29	寄存器全部清零
9	2 个程序内容的比较	30	程序全部删除
10	当前程序内容与磁盘保存内容的比较	31	定时器/计数器经过值清零
11	当前程序中指定文件的插入	32	PLC 系统配置信息的磁盘直接保存
12	程序的磁盘保存、读出	33	磁盘保存的系统配置信息写入 PLC
13	程序的下载、上传	34	PLC 运行中 I/O 状态的画面监视
14	PLC 程序的 LOCK 锁定下载	35	强制 ON/OFF
15	存储器内容的编辑、显示	36	强制寄存器数据写入
16	PLC 运行状态的监控、更改	37	运行中数据监视功能
17	程序、注释、说明内容的打印存档。	38	系统参数初始化
18	工程程序编制信息的显示、编辑	39	程序显示中光标指定地址跳转功能。
19	恢复出厂设置操作	40	本软件显示的颜色、风格的设定
20	回路、功能块注释的编辑、显示	41	工具软件的使用用户管理，功能限制
21	多语言注释编写		

2、工业触摸屏

NK0 系列 PLC 通过该 RS-485 通信口，可以与多家支持光洋 K 协议、CCM2 协议，或 MODBUS RTU 协议等串口通信协议的工业触摸屏相连接，构成现场监控、操作系统。例如：本公司的 CMORE/EA7E/GC/GC-A2 系列触摸屏。



GC-A2 系列触摸屏

另外，NK0-CPU28DR-D 可以通过以太网通信口和带以太网通信口的工业触摸屏进行以太网通信连接。当然所连接的工业触摸屏需要支持 NK0-CPU28DR-D 支持的以太网通信协议（MODBUS TCP 协议）。

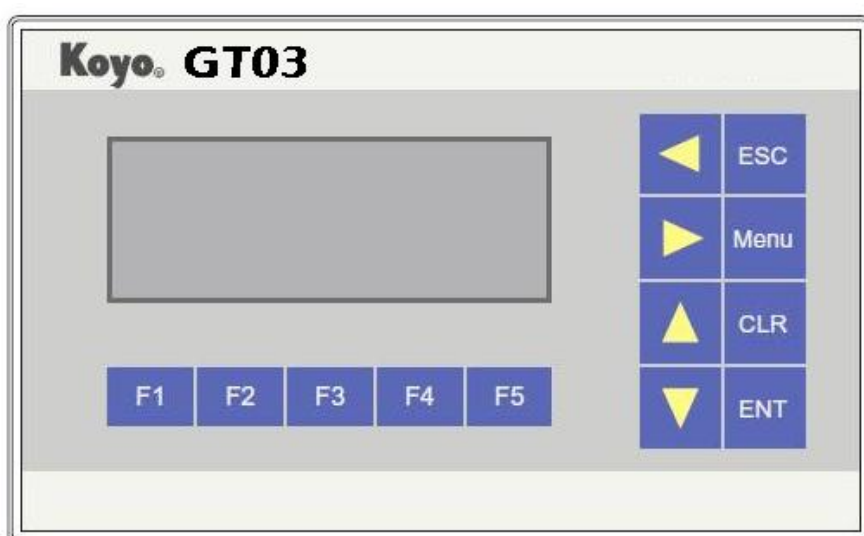
3、显示设定单元 GT03、DV-1000（S-10D）、CL-02/04DS

CL-02/04DS 为汉字式显示设定单元，2 行/4 行汉字显示可实现对 PLC 数据的显示设定，并具有报警功能，它利用计算机上的信息组态软件平台完成对 CL-02/04DS 显示信息的设置工作。（停产）

GT-03 为 CL-02/04DS 的升级机型，除能实现 CL-02/04DS 的功能外，还具有指示灯、趋势图、棒图等图形显示功能。

通过 PLC 编程设定、可在操作显示单元 DV-1000（S-10D）上对普通 I/O、寄存器、定时器、计数器等实时监控，并可随时修改寄存器、定时器、计数器的当前值。另外，DV-1000 又可作为操作按钮使用。在 DV-1000 上除了可显示数字数据外，它还可以显示文字，包括英文，日文以及汉字，并可显示系统错误履历和 FALT 履历。

NK0 系列 PLC 可以通过 RS-485 通信口和以上这些外设显示设备建立通信连接。



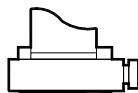
图形显示设定单元 GT-03

4、其它以串口方式与 PLC 交换信息的装置。

NK0 系列 PLC 除以上所列外设外，它通过该 RS-485 通信口还可与其它的支持 RS-485 通信的串行通信设备连接，例如：带 RS-485 通信口的第三方现场检测设备、串行打印机、条码读入机等。



磁卡读入机



打印机



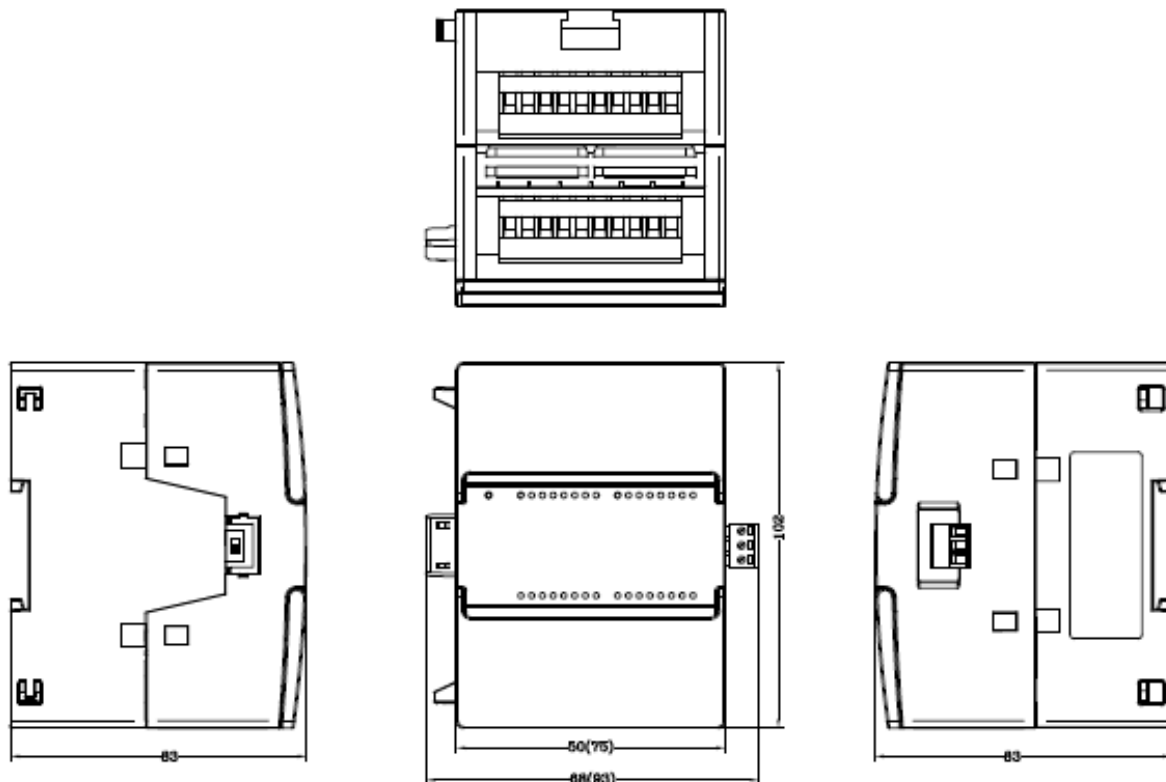
条形码读入器

其它串行通信设备

1-5 NK0 系列 PLC 外形尺寸图

根据 NK0 系列 CPU 型号的不同，其外形尺寸主要分 16 点机型和 32 点机型 2 种（28 点机型外形同 32 点机型）。2 种尺寸的外形，除长度不同外，其他外形尺寸都相同。

NK0 系列 CPU 单元外形尺寸图如下（单位：mm）



注意：上图中 2 种尺寸的 CPU 单元合用一套图。2 者的宽度和高度都一样，仅外形长度不同，16 点机型长度为 68.00mm（不包含左侧凸出和右侧 RS-485 通信端子时为 50.00mm），32 点机型长度为 93.00mm（不包含左侧凸出和右侧 RS-485 通信端子时为 75.00mm）。

各 NK0 系列 PLC 的外形尺寸如下表。

种类	型号	CPU 外形尺寸 (L×W×H 单位：mm)	
基本型	晶体管输出	NK0-CPU16DD1-D	68×102×83
		NK0-CPU32DD1-D	93×102×83
	继电器输出	NK0-CPU16DR-D	68×102×83
		NK0-CPU28DR-D	93×102×83
		NK0-CPU32DR-D	93×102×83
模拟量型	晶体管输出	NK0-CPU16DD1A1-D	68×102×83
		NK0-CPU16DD1A2-D	68×102×83
	继电器输出	NK0-CPU16DRA1-D	68×102×83
		NK0-CPU16DRA2-D	68×102×83

注：上面尺寸中的长度尺寸包含了左侧凸出和右侧 RS-485 端子的尺寸。

第二章 NK0 系列 PLC 系统规格

2-1 NK0 系列 PLC 一般规格

项目	性能
工作温度	0℃~55℃
存放温度	-40℃~70℃
湿度(无凝露)	使用环境湿度 / 保存环境湿度: 30~95% (不结露)
使用环境气氛	无腐蚀性气体
额定电源电压	DC24V
供电电源范围	DC20.4V-28.8V (带接反保护)
允许瞬间掉电时间	5ms 以下, 间隔≥1S
耐压	详见“2-3 NK0 系列 PLC I/O 性能规格”
绝缘	DC 500V 20MΩ 以上
三基抗噪声	1000V 1μs INPULSE
抗冲击	在三个相互垂直轴的每一个轴上(6个方向), 峰值加速度为15G, 持续时间11ms, 各冲击2次。
抗振动	根据IEC61131-2标准、振动方向: X, Y, Z 3方向、扫频次数: 每5~9Hz扫10次, 振幅3.5mm恒定、9~150Hz: 恒定加速度 9.8m/s ² (1G)
自由跌落	1000mm, 5次, 带产品包装
最大消耗功率	10W
传感器用电源	无
安装方式	螺丝固定或 DIN 导轨固定
本体单元重量(克)	NK0-CPU16 机型 ≅ 225g, NK0-CPU32 机型 ≅ 325g
外形尺寸 (长×宽×高)(mm)	NK0-CPU16 机型 68L×102W×83H NK0-CPU32 机型 93L×102W×83H
符合规格	ROHS

2-2 NK0 系列 PLC 性能规格

项 目		性 能
控制方式		循环扫描、定时扫描方式
输入输出控制方式		成批传送和直接输入输出并用
编程方式		梯形图 / 级式并用
模式开关		STOP/RUN (TERM-STOP/TERM-RUN)
命令种类		298 种 (具体参见本资料“附录一 NK0 指令语一览表”)
命令执行速度	顺序命令	0.1 μ s ~
	数据处理命令	1.0 μ s ~
典型扫描时间		1msec / 1K 语
程序容量	内置 FROM	2K 语 (使用 KPPSoft 软件可以实现下传程序的锁定保护)
I/O (I/Q) 线圈		输入 I: 256 点 / 输出 Q: 256 点
通信 I/O (GI/GQ)		输入 GI: 256 点 / 输出 GQ: 256 点 (可作内部线圈使用)
内部线圈(M)		1024 点
定时器(T)	点数	128 点
计数器(C)	点数	128 点
特殊线圈(SP)		512 点
级(S)		256 点
数据寄存器 (R/P) (不含经过值, 特殊等)		2816 字 1 字 = 16 位
累加器 (ACC)		1 点 (32 位): 数据处理用
数据堆栈 (DS1~8)		1 点 (32 位): 数据处理辅助用
输入/输出分配		I/O 定义号自动分配, 带配置监视功能
直接输入/输出		有, 指令支持
数据运算功能		支持整数 BCD, BIN 运算, 浮点数运算, 三角函数运算, 指数对数运算
程序系统资源保护		密码保护: 8 位 BCD 数 (停止模式下, 输入 8 个 'C' 密码恢复出厂设置 密码输错 3 次后必须 NK0 重新上电才能再次输入密码 锁定保护: 下载程序时, 选择程序锁定后, NK0 中程序将不能被读出
数据停电保持区域		根据参数任意设定 (M, R, T, C, S) (无需电池), 最大 64 字。
自诊断功能		· WatchDog 时间 · 存储器异常检查 · 通信异常检查
监视功能		<ul style="list-style-type: none"> · 块监视 · 数据监视 · 执行状态监视 · 自诊断异常信息履历显示 · ON/OFF 监视 · I/O 动作显示 · 自诊断异常信息显示

(续)

项 目		性 能
调试功能		<ul style="list-style-type: none"> ▪ 强制 SET/RESET ▪ 强制数据写入 ▪ 扫描停止 ▪ 单步执行 ▪ N 次扫描执行 ▪ 离线模拟运行功能（编程工具软件 KPPSoft 实现）
外围设备支持		上位编程工具软件：KPPSoft 人机界面：GC-A2、EA7、EA7E、GC、GT03 等
通用串行通信口 (*注 1)	通信口	1 个 (port0)
	信号方式	RS-485 (非绝缘)
	连接方式	3 线式三端子台 (A/B/SG)
	传送距离	1000 米以内
	通信协议	K 协议 (从)、CCM2 (从)、MODBUS RTU (主/从) 无协议 (主/从), 编程软件 KPPSoft 专用通信协议
	通信速度	2400/4800/9600/14400/19200/28800//38400/57600/115200bps
	通信校验	无校验/奇校验/偶校验
	局号设定	根据参数设定
	数据形式	CCM2 协议时 HEX 方式/ASCII 方式可选
	通信参数	编程工具软件设置, 特殊寄存器设置
以太网通信口 (*注 2)	通信口	1 个 (port3)
	通信方式	10/100M 自适应
	通信距离	单段电缆长度不超过 100 米 (STP), 可加 HUB
	连接数	MODBUS TCP (主/从, 作为从局最多同时支持 4 个连接), 编程软件 1 个。
	通信协议	ECOM 协议 (包含编程软件 KPPSoft 专用协议、MODBUS TCP 协议。)

*注 1: 有关 NK0 的 RS-485 串行通信口, 补充以下说明:

在 NK0 处于 TERM-STOP 停止模式下 (与运行开关位置无关), 该 RS-485 通信口 (P0) 按以下固定通信参数动作 (目的: 连接 KPPSoft 工具软件):

局号 = 1 ; 通信速度 = 19200bps; 停止位 = 1 位; 奇偶校验 = 无校验。

其他情况下, 按其通信参数设置寄存器设置的参数动作。

*注 2: 仅 NK0-CPU28DR-D 带以太网口。

2-3 NK0 系列 PLC 开关量 I/O 性能规格

在介绍 NK0 系列 PLC 的普通开关量 I/O 性能规格前，我们先介绍一下晶体管输入/输出回路中的汇点（Sink）和源（Source）的概念，I/O 公共端的概念，以及各种类型 I/O 点如何与现场设备接线以构成应用回路的知识。

2-3-1 汇点（Sink）和源（Source）的概念

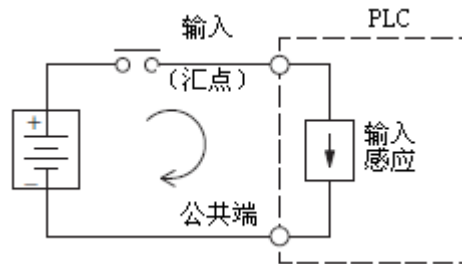
“汇点”和“源”等术语经常出现在输入输出电路的描绘中。为了让这些术语更易于理解，首先给出一个简短的定义，如下：

汇点：到地的路径（-）

源点：到源的路径（+）

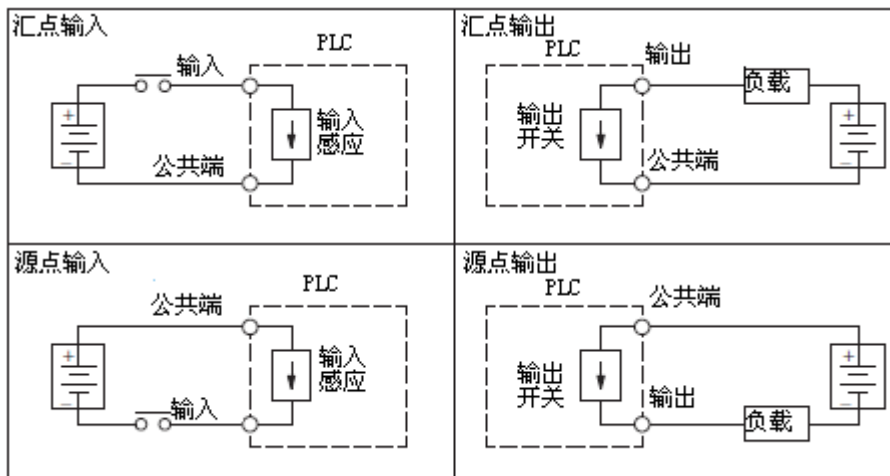
首先，你会注意到这两个术语只与直流电路有关，而与交流电路无关，因为提到了极性“+”和“-”，因此，汇点和源点只应用于直流输入输出电路。不管是汇点还是源点都引导电流流向一个方向，这就意味着把外部电源和现场设备接入 I/O 点时如果方向接错，电路将不动作。因此，理解了汇点和源点，我们就能正确的接入外部电源和现场设备。

右图是一个汇点输入的例子。把外部供电正确的接入，只需把它连接进电路，输入提供了到地（-）的路径。电流从 PLC 输入端开始，经过输入感应电路，在公共端结束，将电源的“-”端同公共端连接起来。通过在电源“+”端与 PLC 输入端之间增加一个开关，电路完成。开关闭合后，电流按图中箭头方向流动。



通过应用以上原则，输入/输出、汇点/源电路类型有四种方式，见下图。注意，图中所指的电流方向是指 PLC 内部回路中的电流方向。

在下面介绍 NK0 系列 PLC 的 I/O 性能规格时，你会发现 NK0 的所有输入点都是双向的，即这些输入点既可以接成汇点输入回路，也可以接成源输入回路。而 NPN 晶体管输出机型的 NK0 的晶体管输出点为汇点输出点。



在上面的介绍中，都提到一个 I/O 公共端的概念。那何谓 I/O 公共端呢？下面介绍 PLC 普通开关量输入/输出回路中的 I/O 公共端的概念。

2-3-2 I/O 公共端的概念

为了使 PLC 普通开关量 I/O 电路正常运行，电流必须从一端流进，从另一端流出，这就意味着每个 I/O 点至少连接两端。右图所示，输入或输出端是电路的主路径，另一端必须提供返回路径，返回到外供电源。

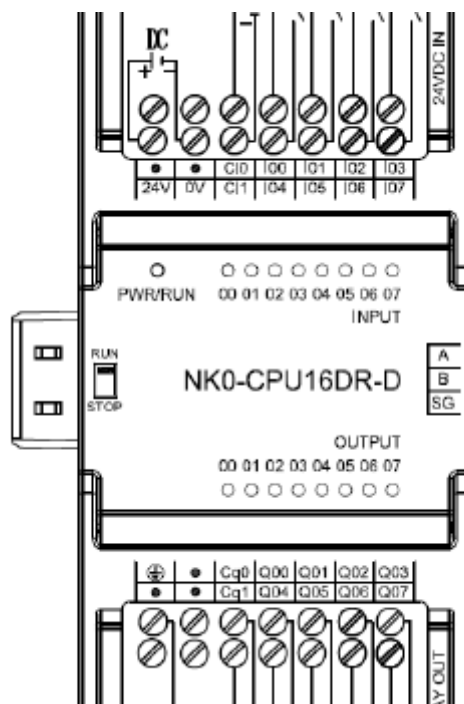
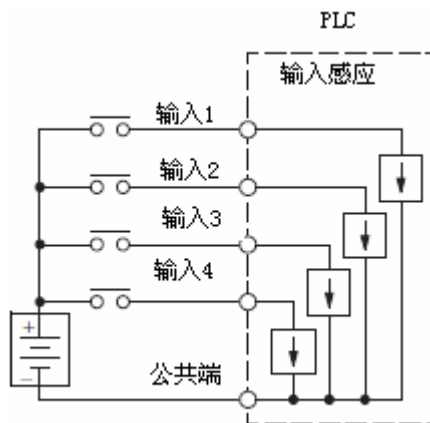
如果我们有无限的空间和预算提供给 I/O 端，那么每个 I/O 点都有两个专用端，但是在应用中这样做是很不切实际并且是没有必要的。实际应用中，大部分输入或输出点共用一个返回路径（公共端）。右图示范了 4 个输入点共用 1 个返回路径的例子，这样，4 个输入点只需要 5 根而不是 8 根线。

注意：上面电路中，如果所有输入同时动作，公共端上的电流将是每个回路的 4 倍。一般在公共端需选用大负载能力的电线，这在输出电路中尤为需要注意。

NK0 系列 PLC 的开关量输入和输出电路都采用上面介绍的回路构成方法，即多个输入点或输出点共用一个公共端（一般 4 点或 8 点）。并且 NK0 各公共端间是相互电气隔离的，在使用 NK0 系列 PLC 时，这点要充分注意。

右图给出 NK0-CPU16DR-D 机型的接线示意图。图中表明 NK0-CPU16DR-D 机型的 8 点输入分配有 Ci0/Ci1 共 2 个输入公共端，每 4 个输入点合用一个输入公共端；8 点输出也分配有 Cq0/Cq1 共 2 个输出公共端，每 4 个输出点合用一个输出公共端。

该机型上的 Ci0/Ci1/ Cq0/Cq1 这 4 个公共端相互间电气隔离，并且和电源 0V 端子间也是电气隔离的。



2-3-3 DC 型 I/O 点与现场晶体管设备的连接

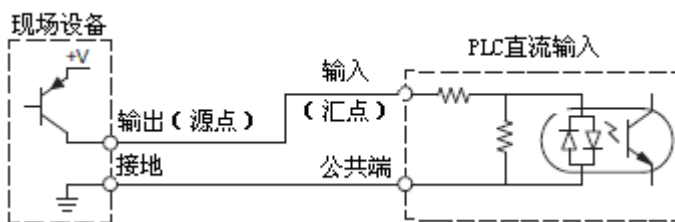
DC 型 I/O 回路只允许一种电流方向，对一些晶体管接口的现场设备来讲一般也是这样的，换句话说，现场设备可以是源点也可以是汇点。当两个设备连接成 DC 回路时，一个必须是源点，而另一个必须是汇点。

1、晶体管传感器输入连接

NK0 系列 PLC 的 DC 输入可以判别两种方向的电流，因此既可以接成源也可以接成汇。下面的电路中，现场设备的集电极开路 NPN 输出，它汇集从 PLC 输入点来的电流（即源电流）。需要+24VDC 电源。



下面的电路中，现场设备为射极开路 PNP 输出。它提供到 PLC 输入点的电流，然后汇集回到地，因为现场设备是源电流，不需要附加电源。

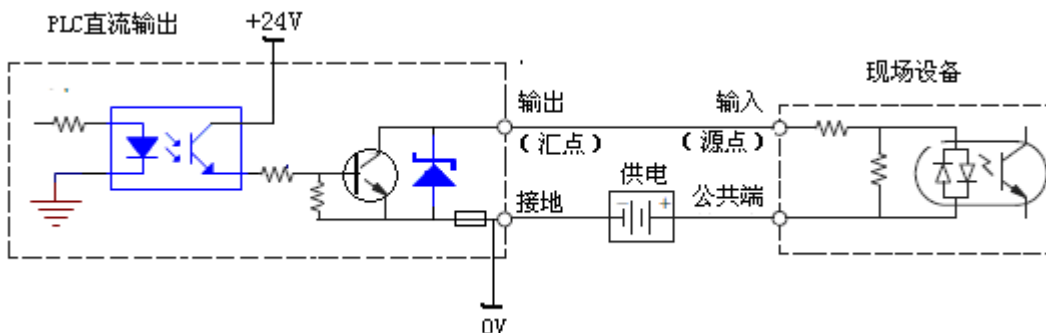


由于 NK0 系列 PLC 的 DC 输入可以是源或汇点方式，因此对于具有多组输入点的 NK0 机型，可以将一组输入接成源而另外一组接成汇点。但注意接在同一个公共端上的同组输入点必须同是源或同是汇点。

2、晶体管输出连接

有的应用中需要连接 PLC 晶体管输出点到设备的晶体管输入。这种连接通常传送低电平信号而不是功率驱动。例如以 NK0-CPU16DD1-D 机型为例，该机型的 DC 输出只有汇点输出（DD1 机型），这意味着每个输出点提供一条到地的通道；该机型产品晶体管输出点以 4 个输出点为一组合用 1 个电气公共端。

下图中，NK0-CPU16DD1-D 输出点动作时汇集电流到公共端，它与现场设备的源输入点连接。



2-3-4 继电器输出点与现场设备的连接

DR 机型的 NK0 都带有一定数量的继电器输出点。继电器输出比较适合以下场合应用：

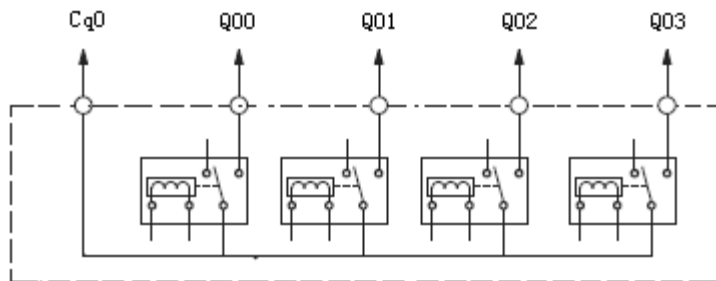
- 负载需要大的电流超过晶体管输出驱动能力的场合；
- 负载开关动作变化不快的场合；
- 某些输出需要和其它输出分离开来，例如一些负载需要 AC 而另一些需要 DC。

但继电器输出不适合以下场合：

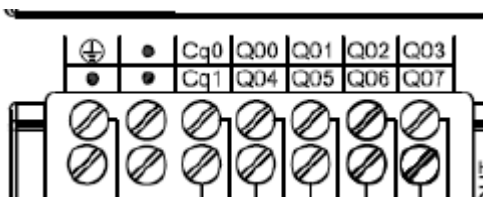
- 负载电流比较小的场合，一般当负载电流小于 10mA 时不适合使用；
- 负载需要快速开关的场合。

下面介绍继电器输出与负载的接线方法。

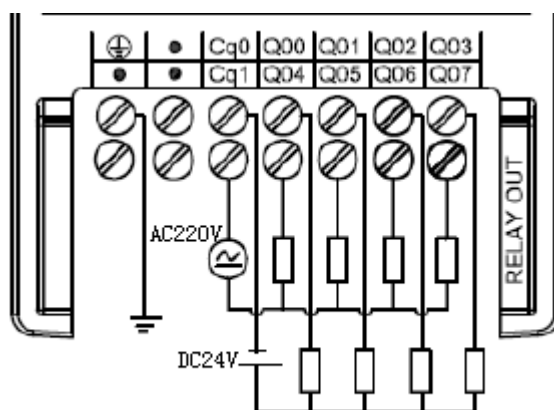
NK0 系列 PLC 的继电器输出点一般以 4 点或 8 点为一组，组之间相互电气独立。例如 NK0-CPU16DR-D 有 2 组共 8 点继电器输出点。下图给出 NK0-CPU16DR-D 第一组（Cq0 组）继电器输出点的内部电路示意图。



下图是 NK0-CPU16DR-D 上所有继电器输出点的接线端分布图。



在用继电器输出点连接现场设备时，可以让不同的继电器组合用一工作电源（公共端短接），也可以让不同的继电器组使用不同的工作电源，以驱动需要不同工作电源的负载组。例如对于 NK0-CPU16DR-D，你可以让第一组输出（Q00/Q01/Q02/Q03）驱动 DC24V 负载；让第二组输出（Q04/Q05/Q06/Q07）驱动 AC220V 负载，如右图所示。



在使用继电器输出点去驱动现场设备时，在使用中以下 2 点需要特别注意。

1、感性负载设备的浪涌抑制

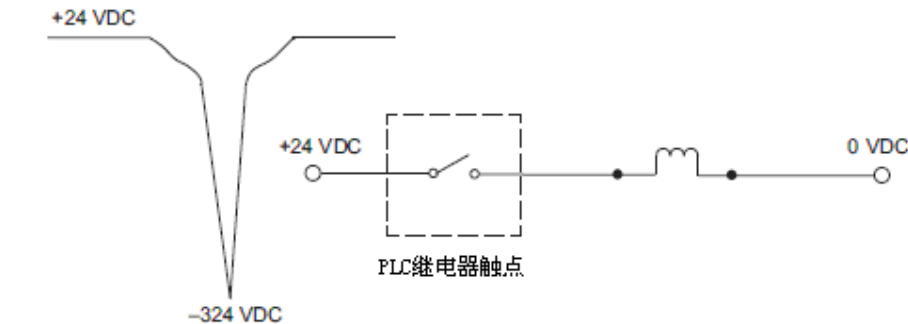
继电器驱动在现场设备分阻性负载设备和感性负载设备（带线圈的设备）。感性负载设备在继电器断电时会产生瞬间电压，当继电器闭合会“反弹”，这种“反弹”会不断的接通、切断线圈，直到“反弹”停止。这个瞬间电压相比外供电压（特别是直流电压）有很大的振幅。

开关一个直流供电感性负载，继电器断开时总会出现满额电压。开关一个交流供电感性负载，会有一个可能——在 60Hz 或 50Hz 时、正弦波在零点时继电器断开。如果继电器断开时电压不是零，则能量存贮在感应器中，在感应器突然断电时释放，这个能量的释放导致了瞬间电压的产生。

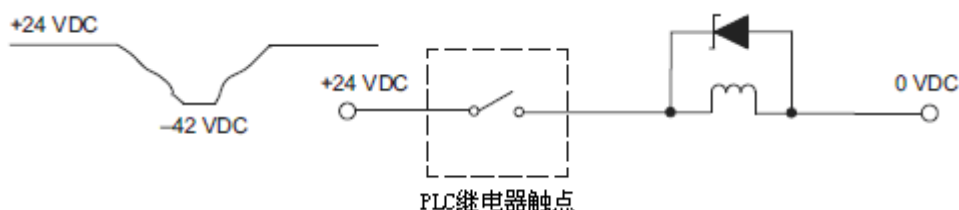
当用继电器触点控制感性负载设备（电动机、电动机启动器、插入式继电器、螺线管、阀门等）时，建议将浪涌抑制设备直接连接至线圈上。如果感性设备有插入式连接器，可将浪涌抑制设备安装在继电器输出的端子排上。

瞬态电压抑制器（TVS 或 transorb）为直流或交流线圈提供了最好的瞬态电压抑制，用最小的过调提供了最快的响应。

金属氧化物变阻器 (MOV) 为交流线圈提供了仅次于瞬态电压抑制器的瞬态电压抑制。下图中的波形演示了断开一个 24VDC 螺线管触点时的可能的能量释放，注意电压的峰值。



下图演示了在同样电路上的线圈上安装了瞬态电压抑制器后的能量释放，可以看到，电压的峰值已经明显减小。



2、如何延长继电器触点寿命

继电器触点的寿命与继电器开关的次数、频率、开关时产生的瞬间放电大小等因素有关。你可以采用多种方法来帮助延长继电器的寿命，例如：仅在需要时开关继电器；尽可能在小电流(在规格允许范围内)状态下开关负载；在驱动感性负载时，想办法抑制其产生的诱导电流等等。

在 DC 回路中驱动感性负载时，可在靠近感性负载的位置增加二极管来保护继电器（在 AC 回路中禁用此法!）。一般采用快速恢复型二极管（例如肖特基型二极管），其 PIV 值至少要 100PIV 以上，正向电流 3A 以上。注意不要用一般小信号的二极管，例如 IN914，IN941，并且方向不要搞错，否则当继电器动作时变成了短路。



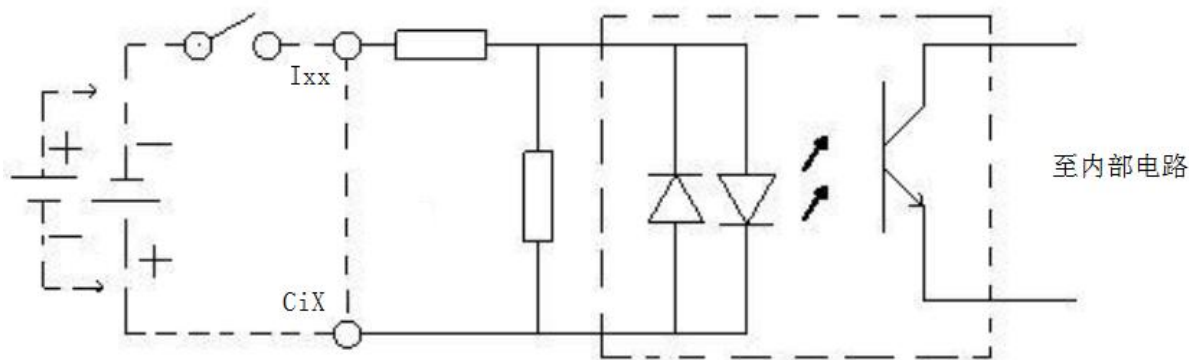
在介绍了晶体管输入/输出回路中的汇点 (Sink) 和源 (Source) 的概念，I/O 公共端的概念，以及各种类型 I/O 点如何与现场设备接线以构成应用回路的知识后，下面介绍 NK0 系列 PLC 的 I/O 性能规格。

2-3-5 DC 输入规格

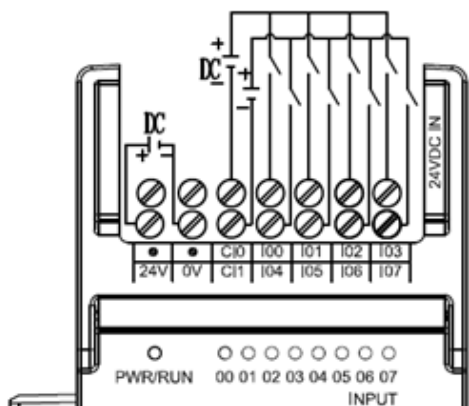
NK0 的所有开关量输入点为双向晶体管输入点，使用时需要外加 DC24V 工作电源。NK0 系列 PLC 根据型号的不同，其输入点按 2 点/4 点/8 点合用一个输入公共端的方式对输入点进行分组，且每组之间相互电气独立。同组之间的输入点需要按相同的电源流入方式进行接线，不同组输入点可以采用不同的接线方式。NK0 普通 DC 输入点规格如下表。

项 目	规 格
输入点数/公共点数	NK0-CPU16DRA1 -D (6 点, 2 公共点, 4 点/公共点, 2 点/公共点)
	NK0-CPU16DRA2 -D (6 点, 2 公共点, 4 点/公共点, 2 点/公共点)
	NK0-CPU16DD1A1 -D (6 点, 2 公共点, 4 点/公共点, 2 点/公共点)
	NK0-CPU16DD1A2 -D (6 点, 2 公共点, 4 点/公共点, 2 点/公共点)
输入点数/公共点数	NK0-CPU16DR -D (8 点, 2 公共点, 4 点/公共点)
	NK0-CPU16DD1-D (8 点, 2 公共点, 4 点/公共点)
输入点数/公共点数	NK0-CPU28DR-D (16 点, 2 公共点, 8 点/公共点)
	NK0-CPU32DR-D (16 点, 2 公共点, 8 点/公共点)
	NK0-CPU32DD1-D (16 点, 2 公共点, 8 点/公共点)
输入点类型	源点/汇点双向
允许电压范围	DC5—30V
额定输入电流	TYP 4mA/点(DC 24V)
输入阻抗	约 6KΩ
浪涌电压	DC35V 持续0.5s
输入 OFF 电压	DC 5V 以下
输入 ON 电压	DC 17V 以上
耐压	500V AC, 持续1min (输入端与电源端/FG之间)
输入 响应 时间	OFF→ON 5mS 以内
	ON→OFF 5mS 以内
绝缘方式	光耦隔离
公共点极性	无, 双向
动作表示	LED 表示
外部接线方法	可插拔式端子台
适合电线尺寸	14~28AWG

NK0 系列 PLC 普通输入点电气原理图如下：



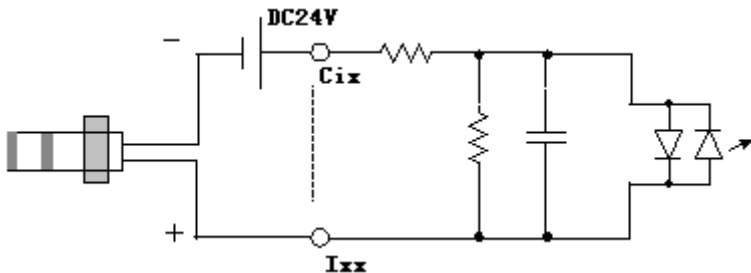
输入接线例：NK0-CPU16DR-D 的输入回路接线例如下：



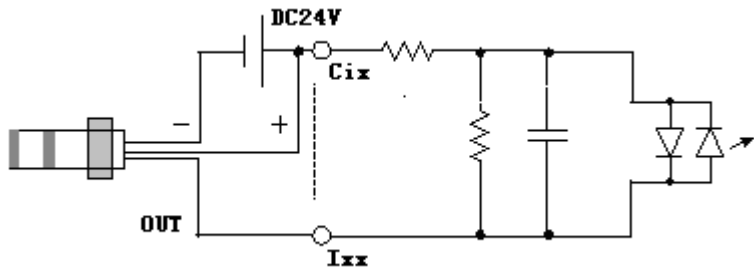
注意：NK0-CPU16DR-D 机型有 2 个输入公共端 Ci0/Ci0，这 2 个公共端内部互相独立。

下面给出NK0系列PLC的普通开关量输入点与接近开关、旋转编码器等传感器的连线图例(图例中给出的为NPN型输入传感器连接图例)：

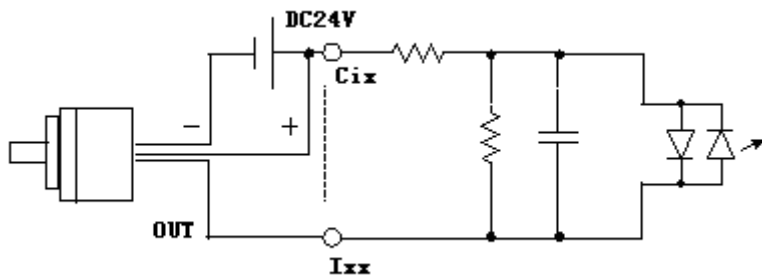
与 2 线式接近开关的连接例：



与 NPN 型 3 线式接近开关的连接例：



与旋转编码器的连接例：



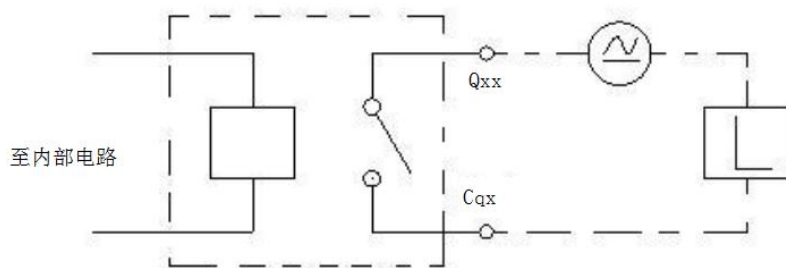
2-3-6 继电器输出规格

NK0 系列 PLC 的 DR 机型都是带有继电器输出点的机型。NK0 的继电器输出点以 4 点或 8 点为一组，合用一个输出公共点 Cqx，一般有几个输出公共点 Cqx，就有几组输出点，且各组之间相互电气隔离。同组输出点需要工作在同一个电源下；但不同组的输出点可以工作在不同的电源下。例如对于 NK0-CPU16DR -D，你可以让 Cq0 所在的输出点组工作在 AC220V 电源下；而让 Cq1 所在的输出点组工作在 DC24V 电源下。NK0 系列 PLC 继电器输出点的规格如下表。

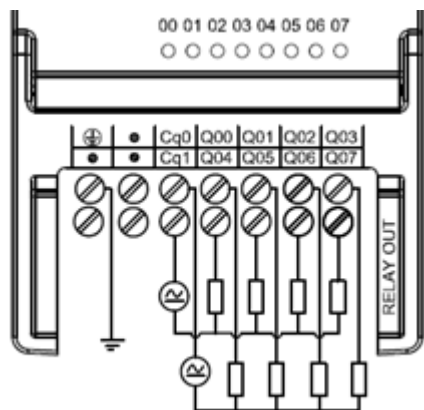
项 目	规 格
输出点数/公共点数	NK0-CPU16DRA1 -D (4 点, 1 公共点, 4 点/公共点)
	NK0-CPU16DRA2 -D (4 点, 1 公共点, 4 点/公共点)
	NK0-CPU16DR -D (8 点, 2 公共点, 4 点/公共点)
	NK0-CPU28DR-D (12 点, 2 公共点, 4 点/公共点、8 点/公共点)
	NK0-CPU32DR-D (16 点, 2 公共点, 8 点/公共点)
输出点类型	继电器, 干触点
工作电压范围	AC5V~250V (47~63Hz) DC5V—30V
浪涌电流	触点闭合时 7A
每点额定电流(最大)	2A (阻性负载) / 0.25A (感性负载)
通态触点电阻	新出厂 $\leq 0.03\Omega(6VDC, 1mA)$
过载保护	无 (需要时, 用户外加保护装置)
浪涌抑制回路	无 (需要时用户可以外加)
耐压	1500V AC, 持续1min (输出端与电源端/FG之间) 750VAC, 持续1min (断开的触点之间)
绝缘电阻	新出厂 $\geq 100M\Omega$
公共端方式	4 点每公共端/8 点每公共端 (各公共端间相互电气隔离)
每个公共点允许电流	16 点机型 $\leq 6A$; 32 点机型 $\leq 10A$
公共端极性	无
输出响应时间 (额定电压下)	OFF→ON: 10ms 以下 ON→OFF: 5ms 以下
机械寿命 (无负载)	2×10^7 次
触点寿命 (额定阻性负载)	1×10^5 次 (ON/OFF 次数, 每分钟 20 次)
最小负载 *1	5VDC, 1mA
动作表示	LED 表示
外部接线方法	可插拔式端子台
适合电线尺寸	14~28AWG

***1 备注:** 该最小负载是参考值, 适用于常温常湿常压的环境。该参考值会根据通断频率、环境条件和期望寿命的不同而改变, 因此请在使用前用实际负载进行确认试验

DR 机型 NK0 系列 PLC 继电器输出点电气原理图如下：



输出接线例：NK0-CPU16DR-D 的继电器输出回路接线例如下：



上述连线中，如果需要，2 组继电器输出点可以合用个工作电源，那时只要把 Cq0、Cq1 短接起来，连到电源的一端，电源另一端连接各负载然后连到各输出端即可。

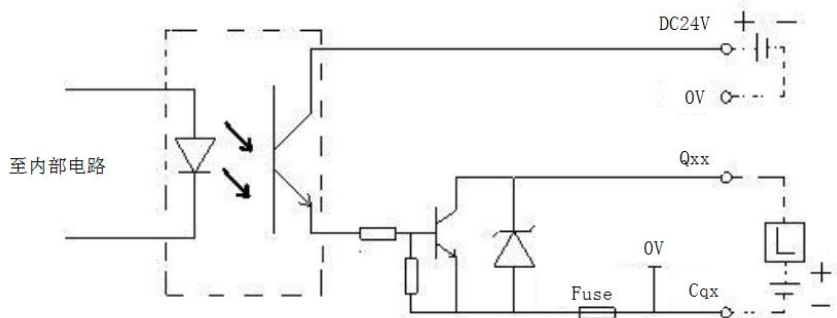
2-3-7 NPN 晶体管输出规格

NK0 系列 PLC 的 DD1 机型都是带有 NPN 晶体管输出点的机型。NK0 的 NPN 晶体管输出点以 4 点或 8 点为一组，合用一个输出公共点 Cqx，一般有几个输出公共点 Cqx，就有几组输出点，且各组之间相互电气隔离。同组输出点需要工作在同一个电源下(该电源为 DC24V,需要外加)；但不同组的输出点可以工作在不同的 DC24V 电源下。例如对于 NK0-CPU16DD1 -D，你可以让 Cq0 所在的输出点组工作在一个 DC24V 电源下；而让 Cq1 所在的输出点组工作在另一个 DC24V 电源下。

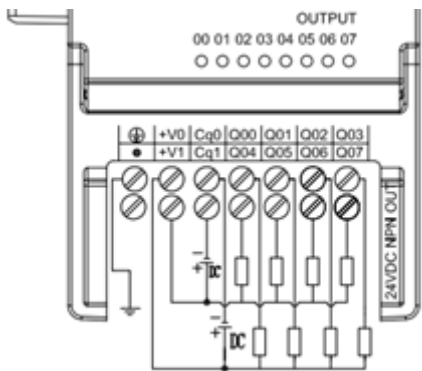
NK0 系列 PLC NPN 晶体管输出点的规格如下表。

项 目	规 格
输出点数/公共点数	NK0-CPU16DD1A1 -D (4 点, 1 公共点, 4 点/公共点)
	NK0-CPU16DD1A2 -D (4 点, 1 公共点, 4 点/公共点)
	NK0-CPU16DD1-D (8 点, 2 公共点, 4 点/公共点)
	NK0-CPU32DD1-D (16 点, 2 公共点, 8 点/公共点)
输出点类型	NPN 集电极开路输出
工作电压范围	DC20.4V—28.8V
每点额定电流 (最大)	约 0.3A
每点漏电流	≦100μ A (DC 40V 时)
最大残电压	2V (0.3A 时)
允许峰值电压	DC 40V (包含纹波的负载电源峰值电压)
耐压	无
输出响应时间	OFF→ON: 0.1mS 以内 ON→OFF: 0.1mS 以内
隔离方式	光耦隔离
过载保护	32点本体: 每8点一个公共点内部保险丝5A, 16点本体: 每4点一个公共点内部保险丝3A
公共端极性	共发射极
动作表示	LED 表示
外部接线方法	可插拔式端子台
适合电线尺寸	14~28AWG

DD1 机型 NK0 系列 PLC NPN 晶体管输出点电气原理图如下:



输出接线例: NK0-CPU16DD1-D 的 NPN 晶体管输出回路接线例如下:

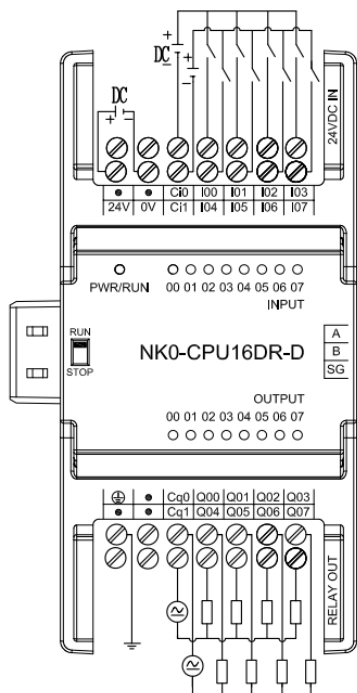


如左图所示连线中, 如果需要, 2 组 NPN 晶体管输出点可以合用一个工作电源, 那时只要把 Cq0、Cq1 短接起来, 连到电源的一端, 电源另一端连接各负载然后连到各输出端即可。

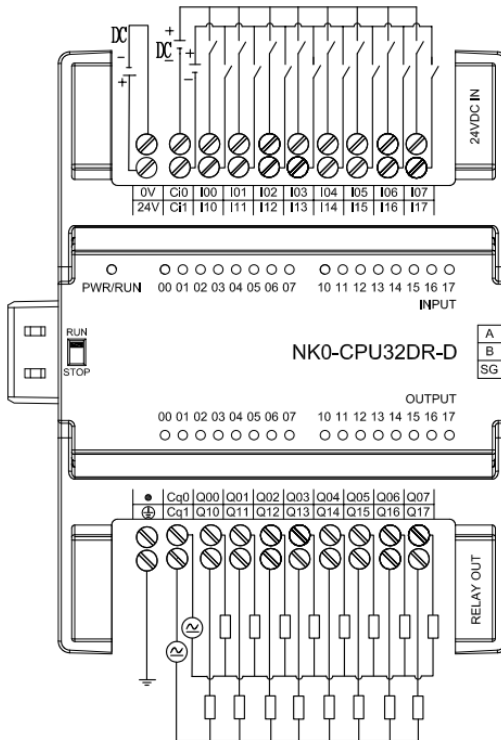
注意: 输出需要外加 DC24V 电源。

2-3-8 NK0 系列各型号 PLC I/O 接线示意图例

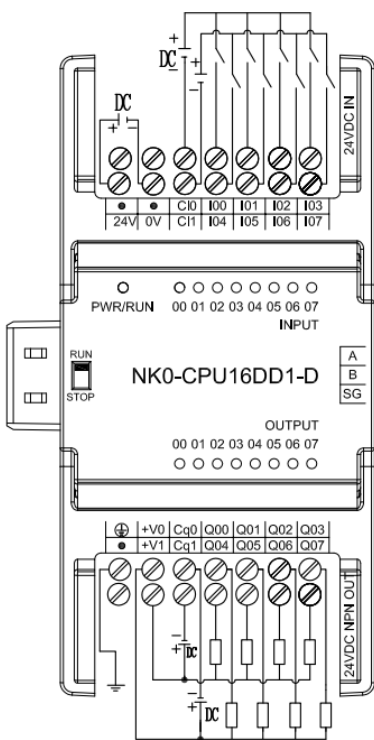
注意：所有 NK0 机型的 DC 输入是双向的，电源可以正反连接，图中仅画出了一种方向。



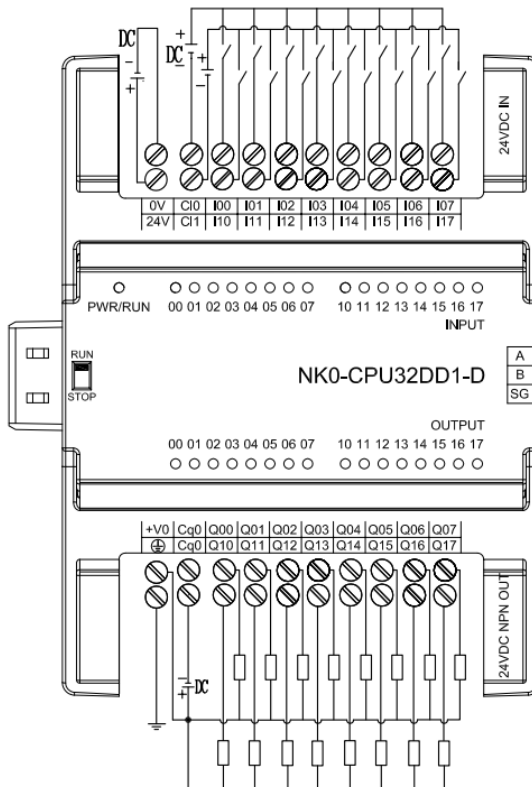
NK0-CPU16DR-D 输入/输出接线示意图例



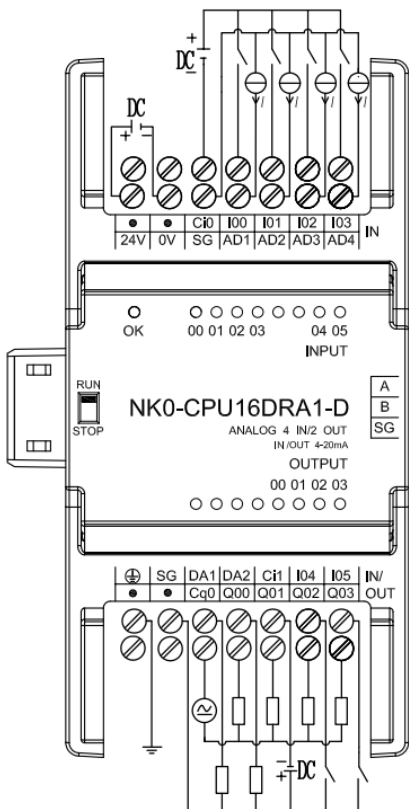
NK0-CPU32DR-D 输入/输出接线示意图例



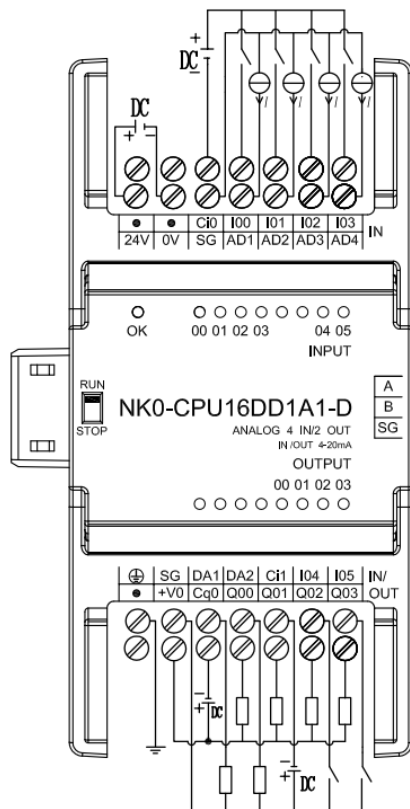
NK0-CPU16DD1-D 输入/输出接线示意图例



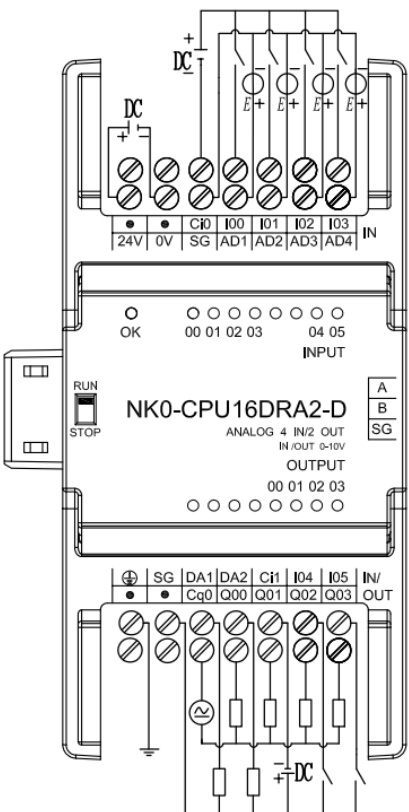
NK0-CPU32DD1-D 输入/输出接线示意图例



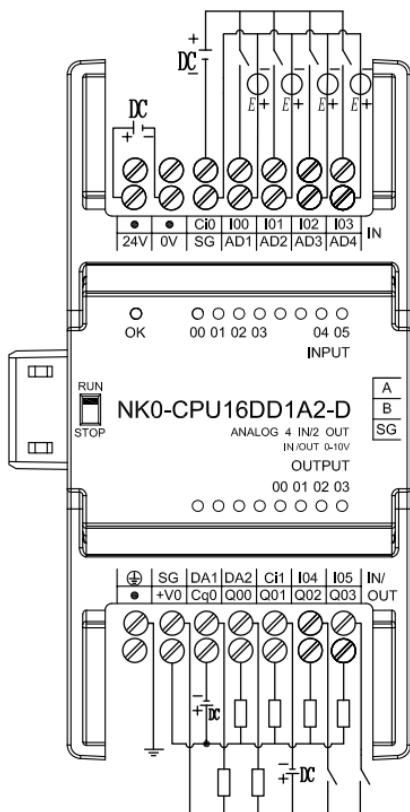
NK0-CPU16DRA1-D 输入/输出接线示意图例



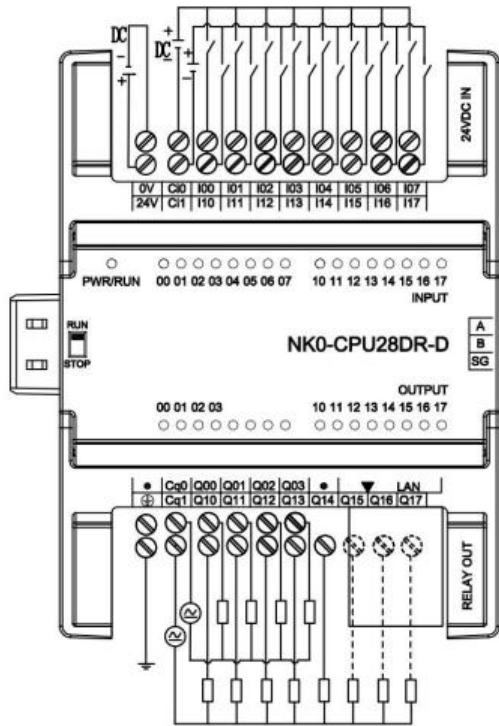
NK0-CPU16DD1A1-D 输入/输出接线示意图例



NK0-CPU16DRA2-D 输入/输出接线示意图例



NK0-CPU16DD1A2-D 输入/输出接线示意图例



NK0-CPU28DR-D 输入/输出接线示意图例

2-4 NK0 系列 PLC 模拟量 I/O 性能规格

每种模拟量机型 NK0 系列 PLC 都带 4 路 AD，2 路 DA，共 6 路模拟量通道。其中 A1 机型的模拟量为 4—20mA 的电流型模拟量信号；A2 机型的模拟量为 0—10V 的电压型模拟量信号。

下面 NK0 所带模拟量点的性能规格。

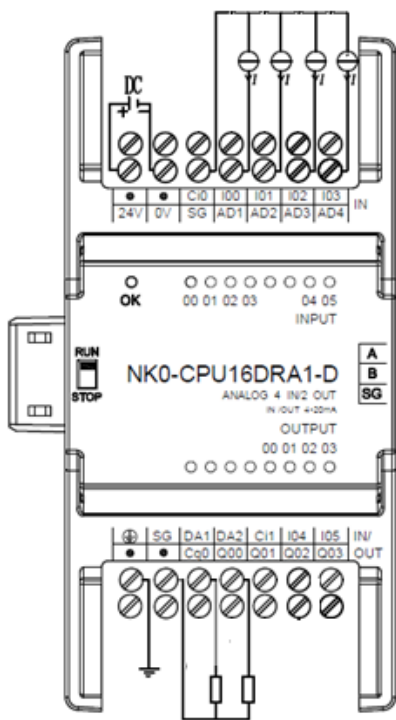
2-4-1 NK0 系列 PLC 模拟量输入规格

项 目	规 格
产品型号/模拟量输入点数/类型	NK0-CPU16DRA1 -D (4 路, 4-20mA)
	NK0-CPU16DRA2 -D (4 路, 0-10V)
	NK0-CPU16DD1A1 -D (4 路, 4-20mA)
	NK0-CPU16DD1A2 -D (4 路, 0-10V)
输入通道数	4 路
输入电流 / 输入电压	4-20mA / 0-10V
输入分辨率	12Bit (0-4095)
输入阻抗	1M Ω \pm 5%
PLC 输入更新率	一个扫描周期更新 4 通道
转换方式	过采样逐次逼近
隔离方式	无
输入误差	电压型：满量程的 \pm 1.0%
	电流型：满量程的 \pm 1.0%
输入公共点方式	4 路 (AD1-AD4) 共有 1 个输入公共点 SG；
输入公共点的电流 (最大)	100mA
噪声环境下最大瞬间偏差	满量程的 \pm 25%
动作表示	无
外部接线方法	可插拔式端子台
适合电线尺寸	14~28AWG

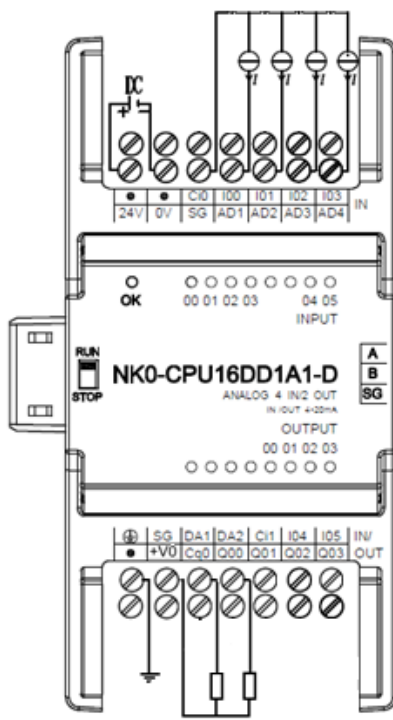
2-4-2 NK0 系列 PLC 模拟量输出规格

项 目	规 格
产品型号/模拟量输入点数/类型	NK0-CPU16DRA1 -D (2 路, 4-20mA)
	NK0-CPU16DRA2 -D (2 路, 0-10V)
	NK0-CPU16DD1A1 -D (2 路, 4-20mA)
	NK0-CPU16DD1A2 -D (2 路, 0-10V)
输出通道数	2 路
输出电流 / 输出电压	4-20mA / 0-10V
输出分辨率	12bit (0-4095)
输出负载阻抗	2000Ω 以上
PLC 更新所有输出通道时间	一个扫描周期更新 2 通道
隔离方式	无
输出误差	电压/电流 满量程的±1.0%
输出公共点方式	2 路(DA1 和 DA2) 共有 1 个输出公共点 SG;
输出公共点的电流 (最大)	50mA
噪声环境下最大瞬间偏差	满量程的±25%
动作表示	无
外部接线方法	可插拔式端子台
适合电线尺寸	14~28AWG

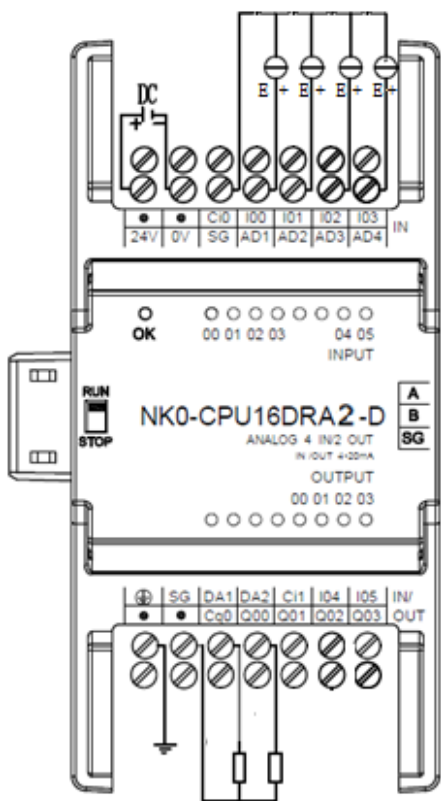
2-4-3 NK0 系列 PLC 模拟量接线示意图例



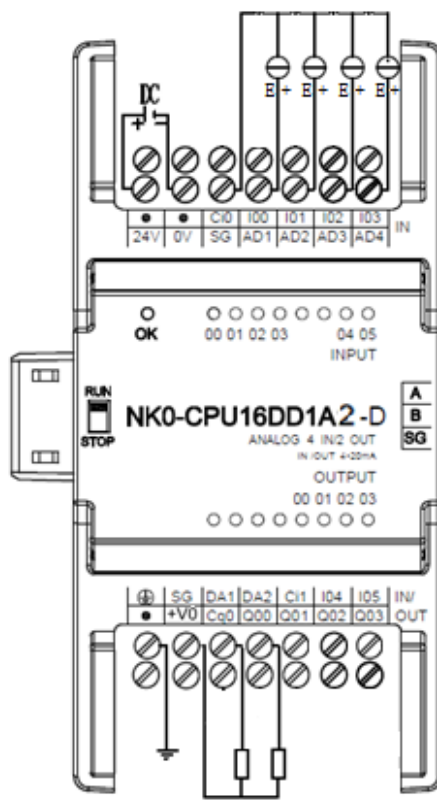
NK0-CPU16DRA1-D 模拟量接线示意图例



NK0-CPU16DD1A1-D 模拟量接线示意图例



NK0-CPU16DRA2-D 模拟量接线示意图例



NK0-CPU16DD1A2-D 模拟量接线示意图例

第三章 NK0 系列 PLC 性能和操作

3-1 NK0 系列 PLC 硬件构成

NK0 系列 PLC 为微型整体式不可扩展 PLC 系列。NK0 系列 PLC 根据其是否带有模拟量通道分成基本型和模拟量型 2 大类型，又根据其所带输入/输出类型和点数又细分成几个机种，这样 NK0 系列 PLC 目前共有 8 个机型供广大用户选择使用，以满足各种小型控制要求。

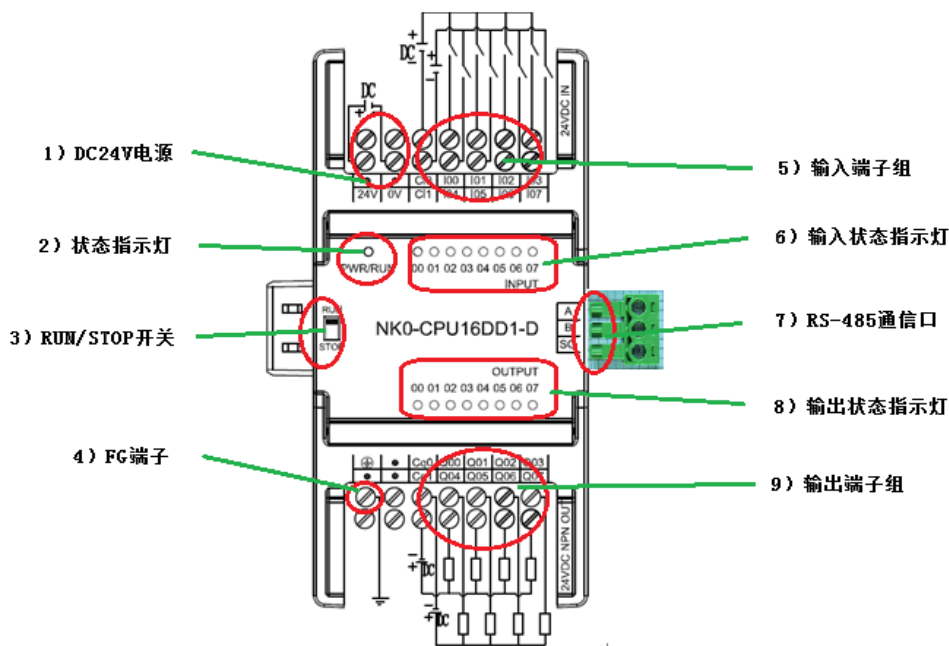
以下是最基本的 NK0-CPU16DR-D 和 NK0-CPU32DR-D 这 2 种机型的外形照片。



下面我们以 NK0-CPU16DR-D 为例，来介绍 NK0 系列 PLC 的硬件构成。你拿到的 NK0-CPU16DR-D 本体单元如下图所示（注意：新出厂的 NK0 产品上右侧的 3P 端子台没有实装，而是以附件的形式随附在产品包装里），表面带有上、下 2 块端子台盖板。



如下图所示为 NK0-CPU16DR-D 去掉上、下端盖后的正面示意图。



从图上可以看到，NK0 本体单元上下各有 2 排接线端子，上侧 2 排接线端子为工作电源输入端子和输入信号接线端子，下侧 2 排为输出信号接线端子和 FG 端子；NK0 本体上有 3 组 LED 状态指示灯，左上边的 PWR/RUN 指示灯为 PLC 工作状态指示灯，其右边的 INPUT 组为输入点状态指示灯；下面的 OUTPUT 组为输出点状态指示灯；本体左侧凸出下为 RNU/STOP 模式开关；本体右侧为 RS-485 通信口（3 端子台通信插座外接）。

下面详细介绍 NK0 的以上各功能部件。

1) DC24V 电源

用于接入 NK0 系列 PLC 工作需要的 DC24V 工作电源。

其允许的工作电源范围是 DC20.4V—28.8V，最大消耗 功率为 10W 。

注意：NK0 的工作电源内部设计有接反保护电路，即使输入电源接反，也能有效保护 NK0 内部电路不被烧坏。。



2) 状态指示灯



NK0 系列 PLC 用一个 PWR/RUN 指示灯来表示 NK0 的工作状态，该指示灯为红/绿双色封装 LED 指示灯，其中红色的为 PWR 指示灯，绿色的为 RUN 指示灯。

NK0 利用这双色指示灯的组合显示来表示其工作状态，详细组合含义如下。

表示 PLC 状态	PWR (红)	RUN (绿)	备注
未上电/或者电源电压低	灭	灭	
上电/停止运行状态 (STOP 模式)	亮	灭	
用户程序运行状态 (RUN 模式)	灭	亮	
RUN 模式，系统有警告性错误时	闪	亮	效果：橙闪
STOP 模式，系统有严重错误时	闪	灭	
固件升级中	灭	闪	

3) RUN/STOP 模式开关

NK0 根据其是否运行用户程序，有 2 种运行模式，分别是运行用户程序的 TERM-RUN 运行模式和不运行用户程序的 TERM-STOP 停止模式。你可以使用 NK0 的运行模式开关或通过 KPSSoft 工具软件来改变 NK0 的运行模式。在 NK0 本体左侧凸出下，有一个 2 位置开关，这是 NK0 的运行模式开关，有“RUN”，“STOP”2 个位置。



“RUN”位置：当把运行模式开关从“STOP”位置打到“RUN”位置时，如果 NK0 内已经存在了没有语法错误的用户程序，且系统没有发现任何软件或硬件的致命错误，则 NK0 进入 TERM-RUN 模式，开始运行用户程序。

“STOP”位置：当把运行模式开关从“RUN”位置打到“STOP”位置时，NK0 停止用户程序的运行，进入 TERM-STOP 停止模式。

注意：不管 NK0 模式开关处于哪个位置，都是处于允许外设操作的 TERM 状态，都可以通过 KPSSoft 工具软件来改变 NK0 的运行模式。

由于不管 NK0 模式开关处于哪个位置，都可以通过 KPSSoft 工具软件来改变 NK0 的运行模式，所以会出现模式开关在“RUN”位置而 NK0 处于不运行用户程序的 TERM-STOP 模式，和模式开关在“STOP”位置而 NK0 处于运行用户程序的 TERM-RUN 模式的情况。这时，你除了可以使用 KPSSoft 工具软件改变 NK0 的运行模式外，也可以通过拨动运行模式开关来改变 NK0 的运行模式。例如：当模式开关在“RUN”位置而 NK0 处于不运行用户程序的 TERM-STOP 模式时，你可以按以下顺序拨动模式开关来改变 NK0 的运行模式到 TERM-RUN 模式：先把模式开关打到“STOP”位置，接着把模式开关再打到“RUN”位置。如果这时 NK0 进不了 TERM-RUN 模式，则说明该 NK0 内没有用户程序或系统有致命的错误。

NK0 上电时，如果 NK0 内有没有语法错误的用户程序并且系统没有致命的错误，则系统按下表设置 NK0 的运行模式。

模式开关位置	NK0 运行模式	说明
“STOP”	TERM-STOP 停止模式	不运行用户程序，PWR 灯亮
“RUN”	TERM-RUN 运行模式	运行用户程序，RUN 灯亮


NK0 上电时，如果 NK0 内没有用户程序且系统没有其他致命的错误，则根据模式开关的位置，会出现如下表所示的情况。

模式开关位置	NK0 运行模式	说明
“STOP”	TERM-STOP 停止模式	无用户程序运行，PWR 灯亮
“RUN”	TERM-STOP 运行模式	PWR 灯闪烁，报 E401 错误（KPP 读出）

NK0 上电和运行用户程序过程中，如果 NK0 有致命的系统错误发生，NK0 会停止用户程序的执行，PWR 灯闪烁，并报告相应的错误代码（用 KPP 读出）。

注意：当 NK0 中没有任何用户程序时，如果把模式开关打到“RUN”位置，或通过 KPSSoft 编程软件把 NK0 运行方式切换为运行的话，则 NK0 的 PWR 指示灯会闪烁，表示发生了致命的错误，同时 NK0 保持为停止模式。如果这时连接有 KPSSoft 编程软件，你可看到 KPP 上显示有致命错误，查 KPP 错误履历，会发现系统报 E401（无用户程序）错误！这时，输入正确的用户程序就可以解除该错误报警（或把模式开关打到“STOP”位置重新给 PLC 上电）。

4) FG 端子

端为 NK0 系列 PLC 的安全地，用于连接系统的保护接地端，以对设备和人员提供更好的安全保护，请在需要时用 2mm² 以上的电线可靠连接至大地或机箱外壳上。

5) 输入端子组

用于连接需要输入 NK0 的各种外部开关、传感器信号。NK0 的所有输入端都为双向 DC24V 信号输入回路，可以根据需要接入汇点或源信号。NK0 提供 2 组输入组（每组一个公共点），各组之间相互电气独立。各组输入可以根据需要分别接入汇点或源信号，但同一组内各输入点必须使用相同的输入方式。

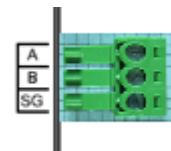
另外注意：NK0 的输入点没有提供内部 24V 连接，需要在外部加入 DC24V 电源。

6) 输入状态指示灯

一组红色指示灯，用于标识输入点的内部通断情况，某灯亮表示其对应的输入点内部为 ON；否则为 OFF。用户可以使用输入指示灯简单判别输入点回路的好坏，如果确认输入端子上有输入信号，而指示灯不亮，则有可能是该点对应的输入回路有问题了（当然也可能是该 LED 坏了）。

7) RS-485 通信口 (PORT3)

在 NK0 右侧面靠上有 1 个 3P 插针排（NK0 表面标有 A/B/SG 文字），这便是 NK0 的 RS-485 通信口，使用时需要安装上附着在 PLC 包装内的 3P 端子台，如右图。该通信口支持编程软件专用协议、K 协议 (S)、CCM2 协议 (S)、MODBUS RTU (M/S)、无协议 (M/S) 等多种通信协议，可连接计算机编程软件 KPPSoft、上位计算机、工业触摸屏、第三方串口通信设备、串行打印机、条码读入机等具串行通信功能的外部设备。



8) 输出状态指示灯

一组红色指示灯，用于标识输出点的内部通断情况，某灯亮表示其对应的输出点内部为 ON；否则为 OFF。可以使用输出指示灯简单判别输出点回路的好坏，如果用 KPPSoft 监视某个输出点对应的线圈导通，而该输出点对应的 LED 指示灯没有亮，则可能是 PLC 内部回路部分上有问题；如果某输出点对应的 LED 指示灯亮，而输出端子上没有信号，则有可能是该点对应的输出回路部分有问题了（包括 NK0 输出驱动部分器件和外部接线/部件问题）。

9) 输出端子组

用于连接驱动现场执行部件负载的一组端子。

根据机型不同，NK0 的输出分继电器输出和晶体管输出 2 种类型。

晶体管输出一般用于连接负载电流比较小而要求较快开关速度的场合。

继电器输出一般用于连接各种普通执行部件负载。继电器输出可以连接交流负载，也可以连接直流负载。连接直流负载时，可以双向输出。继电器输出点比较适合于以下场合：

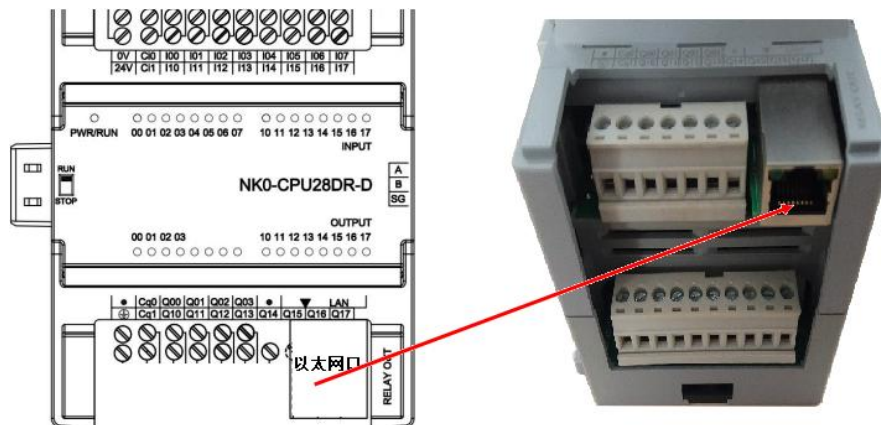
- 负载需要的电流超过晶体管输出的驱动能力；
- 开关动作不是太快的场合。

NK0 根据机型不同，提供 1 到 2 组输出点组（每组一个公共点），各组之间相互电气独立。继电器输出可以根据应用需要去驱动交流负载或直流负载。晶体管输出为 NPN 集电极开路型输出信号，为汇点型输出点，一般连接源型负载。

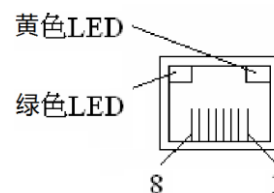
另外注意：NK0 的输出点不提供负载回路电源，需要在外部连接负载电源。

10) 以太网通信口 (PORT3)

仅 NK0-CPU28DR-D 机型带以太网通信口，该通信口在 NK0-CPU28DR-D 上位置示意图如下。



在 NK0-CPU28DR-D 本体下侧输出端子台盖板下右侧，配有一个 RJ45 标准的以太网口（如右），该以太网通信口为符合 IEEE802.3 规格的标准以太网口，通信速度根据所连接的网络，系统自己选择 10M 或 100M，支持 MODBUS TCP 通信协议，可用于连接 KPPSoft 编程软件或其他支持 MODBUS TCP 通信协议的以太网通信设备。



3-2 NK0 系列 PLC 的 I/O 识别和定义号分配

3-2-1 NK0 系列 PLC I/O 的识别

NK0 系列 PLC 为微型整体式不可扩展 PLC。为了有效识别 NK0 系列 PLC 的输入/输出点，我们对输入赋予识别记号 I，对输出赋予识别记号 Q。用“识别记号+数字”的方式来标识每一个输入/输出点，我们把“识别记号+数字”的标识号称为 PLC 的 I/O 定义号。

每一个输入/输出点，在 PLC 内部都有一个对应的存储器空间用于存放该点状态，PLC 内部这样的存储器，我们称为功能存储器。另外，一个 PLC 上，除输入/输出功能存储器外，还有其他的功能存储器，例如：中间线圈、定时器、计数器、特殊线圈、数据寄存器等。

为了识别各种功能存储器，也需要给所有的功能存储器编号，称为功能存储器的定义号。对于输入/输出来说，其功能存储器定义号与输入/输出识别定义号是一样的。NK0 系列 PLC 的所有定义号采用 8 进制地址表示，并且每种定义号从‘0’开始编号。这意味着在定义号地址码中不会出现数字‘8’和‘9’。

有关 NK0 系列 PLC 功能存储器的描述，请参见《3-3 NK0 系列 PLC 功能存储器》一节。

3-2-2 NK0 系列 PLC I/O 定义号的分配

NK0 的 I/O 定义号采用输入/输出自动分配的原则进行分配。其定义号分配原则如下：开关量 I/O 从 I0/Q0 开始自动分配。

各机型 NK0 的定义号分配如下表：

NK0 机型	输入定义号	输出定义号	备注
NK0-CPU16DD1-D	I00~I07	Q00~Q07	未分配给实装输入输出的定义号 (I、Q)，在程序上可作为内部线圈使用。 注 1: 实际占用输出点如下 (12 点): Q00~Q03 , Q10~Q17
NK0-CPU32DD1-D	I00~I17	Q00~Q17	
NK0-CPU16DR-D	I00~I07	Q00~Q07	
NK0-CPU28DR-D	I00~I17	Q00~Q17 (注 1)	
NK0-CPU32DR-D	I00~I17	Q00~Q17	
NK0-CPU16DD1A1-D	I00~I05	Q00~Q03	
NK0-CPU16DD1A2-D	I00~I05	Q00~Q03	
NK0-CPU16DRA1-D	I00~I05	Q00~Q03	
NK0-CPU16DRA2-D	I00~I05	Q00~Q03	

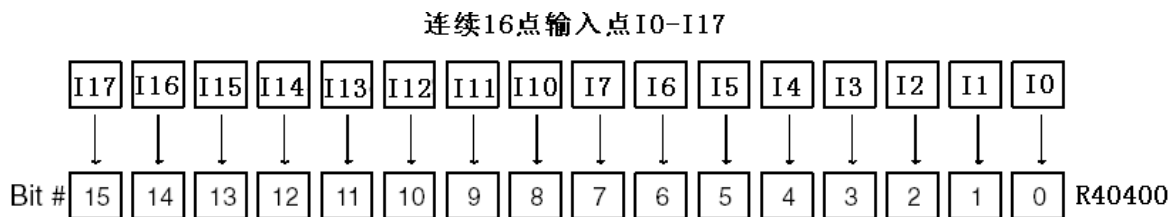
用 KPPSoft 连接 NK0 后，可以用“系统配置”来看到 NK0 的 I/O 定义号分配情况。



3-3 NK0 系列 PLC 功能存储器

在 PLC 系统中一般有很多信息需要处理，包括输入设备状态，输出设备状态，各定时器/计数器经过值及其接点状态，以及各种必要的中间接点、数值数据等，理解各种数据是如何在系统中存储和处理的非常重要。例如，你必须清楚地了解系统是如何区分输入点状态，输出点状态，数值数据的。在 NK0 系列 PLC 中，系统处理的各种信息都存放在内部存储器中，我们把存放各种状态信息的内部存储器称为功能存储器，同时用我们称之为识别记号的头文字来区分各种不同的功能存储器，每种功能存储器都用识别记号加功能号的办法来区分，并分别从 0 开始单独编号。NK0 系列 PLC 的功能存储器号采用 8 进制数编制。

NK0 系列 PLC 的功能存储器分位状态型功能存储器和字值型功能存储器 2 大类。位状态型功能存储器每个仅有一个 BIT 位，可以为 1 或 0。这类功能存储器有通信输入 GI、通信输出 GQ、输入 I、输出 Q、中间线圈 M、级 S、特殊线圈 SP、定时器接点 T、计数器接点 C；字值型功能存储器为具有 16Bits 的存储器，称为寄存器 R（用于间接寻址指令时用 P 表示。），主要用于存放、处理数值数据。有些信息被自动装入 R 寄存器中，例如定时器的经过值（也称当前值）等。另外，为了处理上的方便，NK0 系列 PLC 提供了一种用 R 寄存器来存取位状态型功能存储器信息的方法。每个 R 寄存器对应 16 个连续的位状态信息。例如，你可以用 R40400 来取得 I0~I17 这 16 个输入点的状态。



NK0 系列 PLC 各功能存储器性质说明如下：

- 1、输入 I
用于记忆连接到输入端子上的外部输入设备的开关状态。
- 2、输出 Q
用于记忆用来驱动与输出端子相连接的外部输出设备动作的输出状态，一般为程序执行的结果状态。
- 3、内部线圈 M
记忆用户程序的执行结果，用于内部处理的中间状态。
- 4、定时器 T
记忆定时器经过值（自动存放在 R 中）是否到达预先设定时间的状态标记（定时到标记）。
- 5、计数器 C
记忆计数器经过值（自动存放在 R 中）是否到达预先设定数值的状态标记（计数到标记）。
- 6、级 S
用于记忆级式编程方式中工程的动作状态。
- 7、特殊线圈 SP
系统预先定义好功能的一些状态标记，可作为用户程序中的逻辑条件加以直接利用。
- 8、寄存器 R
用于存放数据处理用的数据值。每个寄存器由 16Bits 组成，用于间接寻址指令时用 P 表示。
NK0 的数据寄存器分二种：一种是给用户程序使用的用于各种数值处理、存放的数据寄存器；另一种是 NK0 系统用于存放各种系统设置、工作参数的寄存器，称为特殊寄存器。
- 9、通信 I/O 输入（GI）/ 输出（GQ）
在光洋体系 PLC 中，用通信输入 GI，通信输出 GQ 来作为远程通信 I/O 点来使用，主要用于控制远程 I/O 系统中的输入、输出状态。当没有在远程 I/O 系统中使用时，可以作为中间线圈来使用。NK0 系列 PLC 目前不支持远程 I/O 系统，所以在 NK0 系列 PLC 中，GI/GQ 作为中间线圈来使用。

3-3-1 NK0 系列 PLC 功能存储器一览表

识别记号	名称	范围	点数/W	备注
I	输入线圈	I 0~I377	256 点	未分配给实装输入输出的定义号，可作为内部线圈使用。
Q	输出线圈	Q 0~Q377	256 点	
M	内部线圈	M 0~M1777	1024 点	可指定停电保持
S	级	S 0~S377	256 点	可指定停电保持
T	定时器接点	T 0~T177	128 点	可指定停电保持
C	计数器接点	C 0~C177	128 点	可指定停电保持
GI	通信线圈	GI 0~GI377	256 点	NK0 目前不支持远程 I/O，GI/GQ 作内部线圈用
GQ		GQ 0~GQ377	256 点	
SP	特殊线圈	SP 0~SP777	512 点	系统定义好的标志
R/P	定时器经过值	R 0~R177	128 字	定时器一点对应一个字 可指定停电保持
		P 0~P177		
	计数器经过值	R 1000~R1177	128 字	计数器一点对应一个字 可指定停电保持
		P 1000~P1177		
	特殊寄存器	R 7400~7777	256 字	系统定义好的特殊用途寄存器
		P 7400~7777		
	数据寄存器	R2000~R7377	2816 字	可指定停电保持
		P 2000~R7377		
R	通信输入线圈	R 40000~R40017	16 字	通信输入对应寄存器
	通信输出线圈	R 40200~R40217	16 字	通信输出对应寄存器
	输入线圈	R 40400~R40417	16 字	输入对应寄存器
	输出线圈	R 40500~R40517	16 字	输出对应寄存器
	内部线圈	R 40600~R40677	64 字	内部线圈对应寄存器
	级	R 41000~R41017	16 字	级对应寄存器
	定时器	R 41100~R41107	8 字	定时器接点对应寄存器
	计数器	R 41140~R41147	8 字	计数器接点对应寄存器
特殊线圈	R 41200~R41237	32 字	特殊线圈对应寄存器	
LBL	子程序	CLBL K 0~FFFF	256 段	
	GOTO 块	GLBL K 0~FFFF	256 块	
	数据定义块	DLBL K 0~FFFF	256 块	

<注意>

- ◆ 1 个寄存器对应 1 个字或者 2 个字节（16bits）。
- ◆ 停电保持区域无须另配电池，就可以实现停电保持，最大支持 64 个字。
- ◆ 特殊线圈和特殊寄存器区域已经在系统中被定义，用户程序中不可随意挪作它用。
- ◆ 累积定时器（ATMR，AHTMR）、加减计数器（UDCNT）等指令，在指令中用到 8 位 BCD 数据，其对应的经过值需要用到 2 个寄存器，使用时请注意其定义号不要重复使用。
- ◆ 没有使用的 I、Q、S 等可以作为内部线圈使用。
- ◆ 寄存器 R 中的数值，可以是 BCD 数、HEX 数、8 进制数或其它类型的数。我们可以用显示工具或指令来读出寄存器中的数据，并显示成 2 进制数、8 进制数、10 进制数、16 进制数，但它们在存放时都被转换成 2 进制数。我们并不能清楚地说出一个寄存器中数值的数据类型，但这并不重要。重要的是：某个寄存器 R 中数据的来源或写入方与后来用于读出该寄

寄存器数据的一方必须使用同一种数据类型，这就足够了。R 寄存器仅仅是个数据存放地，它本身并不改变或移动寄存器中的数据。

- ◆ 对于以位单位存取的领域[GI、GQ、I、Q、M、S、T、C、SP]，系统也配置了与之对应的以 16 个位单位地址为一组存取的 R 寄存器区域，以方便在数据运算等操作中存取各位状态信息。这是光洋 PLC 特有的功能。具体对应关系如下表所示：

寄存器范围	功能存储器内省类型	位状态范围	点数
R 4 0 0 0 0 ↓	通信输入线圈领域 (I) 1 6 字	GI 0 0 0 0 ↓	2 5 6 点
R 4 0 0 1 7		GI 0 3 7 7	
R 4 0 2 0 0 ↓	通信输出线圈领域 (Q) 1 6 字	GQ 0 0 0 0 ↓	2 5 6 点
R 4 0 2 1 7		GQ 0 3 7 7	
R 4 0 4 0 0 ↓	输入线圈领域 (I) 1 6 字	I 0 0 0 0 ↓	2 5 6 点
R 4 0 4 1 7		I 0 3 7 7	
R 4 0 5 0 0 ↓	输出线圈领域 (Q) 1 6 字	Q 0 0 0 0 ↓	2 5 6 点
R 4 0 5 1 7		Q 0 3 7 7	
R 4 0 6 0 0 ↓	内部线圈领域 (M) 6 4 字	M 0 0 0 0 ↓	1 0 2 4 点
R 4 0 6 7 7		M 1 7 7 7	
R 4 1 0 0 0 ↓	级定义号领域 (S) 1 6 字	S 0 0 0 0 ↓	2 5 6 点
R 4 1 0 1 7		S 0 3 7 7	
R 4 1 1 0 0 ↓	定时器接点领域 (T) 8 字	T 0 0 0 ↓	1 2 8 点
R 4 1 1 0 7		T 1 7 7	
R 4 1 1 4 0 ↓	计数器接点领域 (C) 8 字	C 0 0 0 ↓	1 2 8 点
R 4 1 1 4 7		C 1 7 7	
R 4 1 2 0 0 ↓	特殊线圈领域 (SP) 3 2 字	SP 0 0 0 0 ↓	5 1 2 点
R 4 1 2 3 7		SP 0 7 7 7	

注：上表中没有用到的 R 寄存器号，系统保留。

3-3-2 NK0 系列 PLC 特殊线圈

特殊线圈 SP 是在 NK0 系列 PLC 中已预先定义用途的内部线圈，在程序中，只能作为接点条件来使用。NK0 已经定义的特殊线圈列表如下：

SP 地址	功能	备注
开始和实时线圈		
SP000	初始复位	CPU 运行后的第一个扫描周期内接通
SP001	常时 ON	与 CPU 无关一直为 ON
SP002	常时 OFF	与 CPU 无关一直为 OFF
SP003	1 分钟时钟	30 秒 ON 30 秒 OFF（OFF 状态启动）
SP004	1 秒钟时钟	0.5 秒 ON 0.5 秒 OFF（OFF 状态启动）
SP005	100ms 时钟	50msON 50msOFF（OFF 状态启动）
SP006	50ms 时钟	25msON 25msOFF（OFF 状态启动）
SP007	扫描时钟	一次扫描 ON，一次扫描 OFF（ON 状态启动）
CPU 状态线圈		
SP010	未定义（常时 OFF）	未定义
SP011	未定义（常时 OFF）	未定义
SP012	TERM RUN 状态	当 CPU 处于 RUN 状态时 ON
SP013	TEST RUN 状态	当 CPU 处于 TEST RUN 时 ON
SP014	未定义（常时 OFF）	未定义
SP015	TEST STOP 状态	当 CPU 处于 TEST STOP 时 ON
SP016	TERM STOP 状态	当 CPU 处于 STOP 的时候 ON
SP017	未定义（常时 OFF）	未定义
SP020	STOP 继电器	当 STOP 指令执行的时候 ON
SP021	未定义（常时 OFF）	未定义
SP022	未定义（常时 OFF）	未定义
SP025	未定义（常时 OFF）	未定义
SP030	SP_DATA_SETTING_ERR	数值设定有错误时 ON
SP031	SP_NO_PROGRAM	无用户程序时 ON
SP032	SP_IP_READY	取得 IP 地址时 ON，以太网有关指令操作需要检查该标志（注 1）
SP033	SP_TCP_CONNECTION_FULL	MRX/MWX 指令使用的通信连接数溢出时 ON（注 1）
系统监视线圈		
SP040	致命、重度异常继电器	当出现关键错误的时候 ON
SP041	轻度异常继电器	当出现一个非关键错误的时候 ON
SP044	程序内存错误	当内存出现错误，例如校验错误的时候 ON
SP045	未定义（常时 OFF）	未定义
SP046	通信异常	当 RS-485 通信口出现通信异常时 ON
SP047	未定义（常时 OFF）	未定义
SP050	未定义（常时 OFF）	未定义
SP051	看门狗溢出标志	如果看门狗时间溢出则 ON
SP053	运算出错状态继电器	当无法进行运算的时候 ON
SP054	通信口参数错误	当通信口参数设定有错误时 ON

注 1：仅 NK0-CPU28DR-D 支持。

累加状态线圈		
SP060	小于标志	当 ACC 的值小于比较值的时候 ON
SP061	等于标志	当 ACC 的值等于比较值的时候 ON
SP062	大于标志	当 ACC 的值大于比较值的时候 ON
SP063	零标志	当运算结果为 0 的时候 ON
SP064	半借位标志	根据数据处理指令的运算结果而变化, 在执行减法指令向第 16 位借位时 ON
SP065	借位标志	根据数据处理指令的运算结果而变化, 在执行减法指令向第 32 位借位时 ON
SP066	半进位标志	执行加法指令, 往 16 位进位时 ON
SP067	进位标志	执行加法指令, 往 32 位进位时 ON
SP070	符号标志	ACC 的第 31 位的状态
SP071	间接指定出错	间接寄存器指定了不存在的范围的时候 ON
SP072	浮点数标志	任何时候 ACC 不是浮点数时 ON
SP073	溢出标志 1	带符号运算指令执行时, 结果溢出为 ON
SP074	溢出标志 2	当浮点数运算结果溢出的时候 ON
SP075	数据出错标志	BCD 运算的时候, 如果运算值不为 BCD 数的时候 ON
SP076	读零标志	执行读取数据命令的时候, 如果为 0 则 ON
通信监视线圈		
SP110	Port0 (RS-485 口) 通信标记	当 Port0 当作主局使用, 和其它设备通信中时 ON
SP111	Port0 (RS-485 口) 出错标记	当 Port0 当作主局使用, 通信出错的时候 ON
SP116	Port3 (以太网口) 通信标记	当 Port3 当作主局使用, 和其它设备通信中时 ON
SP117	Port3 (以太网口) 出错标记	当 Port3 当作主局使用, 通信出错的时候 ON

说明: 1) 表中没有列出的 SP, 为系统保留用, 通常为 OFF 状态, 请不要用作它用。
2) 使用 MRX/MWX 指令时, SP110、SP111, SP116、SP117 标记线圈无效。

3-3-3 NK0 系列 PLC 特殊寄存器

特殊寄存器，是系统已经规定了用途的一组寄存器，也称系统寄存器。在编制用户程序时，请按规定使用这些特殊寄存器。NK0 系列 PLC 系统寄存器范围为 R7400—R7777。其中有一些系统没有规定用途的预留寄存器，在将来也可能被定义为某个特殊用途，所以在应用中不要挪作它用。

特殊寄存器	功能	说明	FlashROM 对应
R7400- R7437	模拟量输入数据开始寄存器	4AD: (HEX 000-FFF) R7400: CH1 R7401: CH2 R7403: CH3 R7404: CH4	
R7440- R7477	模拟量输出数据开始寄存器	2DA: (HEX 000-FFF) R7440: CH1 R7441: CH2	
R7620	DV-1000 设定值变更寄存器设定	寄存器首地址的设定 (BIN)	
R7621	DV-1000 设定值变更注释寄存器设定	注释显示寄存器首地址的设定 (BIN)	
R7622	DV-1000 设定值变更寄存器数设定	寄存器数可设定设定 (1~32 BCD)	
R7623	DV-1000 数值显示寄存器首址设定	数值显示寄存器首址设定 (BIN)	
R7624	DV-1000 文字显示寄存器首址设定	文字显示寄存器首址设定 (BIN)	
R7625	DV-1000 开关量分配寄存器地址指定	I, Q, M 组寄存器地址号 BIN 数	
R7626	DV-1000 上电强制显示内容设定	BCD 功能码: 0, 1, 2, 3, 12	
R7627	DV-1000 设定值变更操作密码指定	1~999 的 BCD 码, 初始 0000	
R7633	BIN 运算符位设定	Bit15=0: 无符号 BIN 运算 (默认值) Bit15=1: 带符号 BIN 运算 初始值: 0000	有对应
R7650-R7661	PORT0 (RS-485) 通信口通信参数设置寄存器组	具体参见《第四章 NK0 系列 PLC 通信功能》	有对应
R7706-R7717	PORT3 (以太网) 通信口通信参数设置寄存器组	具体参见《第四章 NK0 系列 PLC 通信功能》	有对应
R7720	DV-1000 定时器值显示设定	定时器组首地址设定 (BIN)	
R7721	DV-1000 计数器值显示设定	计数器组首地址设定 (BIN)	
R7722	DV-1000 定时器/计数器显示设定	显示数量设定 (BCD)	
R7726	系统程序版本号 (高位)	(XXXX: 年份)	
R7727	系统程序版本号 (低位)	(YYZZ: YY 月份, ZZ 日期)	
R7747	10ms 定时计数器	每隔 10ms 加 1, 范围 00~99	
R7755	致命异常代码	发生致命错误时的错误代码	
R7756	重度异常代码	发生非致命错误时的错误代码	
R7757	轻度异常代码	发生警告错误时的错误代码	
R7765	扫描计数器	RUN 以后的程序扫描次数 (HEX)	
R7775	当前扫描时间	当前扫描时间(ms)	
R7776	RUN 开始后最短扫描时间	RUN 开始后最短扫描时间(ms)	
R7777	RUN 开始后最长扫描时间	RUN 开始后最短扫描时间(ms)	

3-3-4 FlashROM 型特殊寄存器

通常情况下，NK0 的特殊寄存器为 RAM 型存储器，如果没有设置成停电保持，其内容将在停电后丢失。但有些应用场合下，需要对某些特殊寄存器内容进行停电保持，即使该特殊寄存器没有被设置在停电保持范围内。为了能够在停电时保持特殊寄存器的内容，我们把某些特殊寄存器设计成对应 RAM 和 FlashROM 2 个数据存放位置。当通过 PLC 程序改写这些特殊寄存器内容时，其仅仅修改 RAM 中的内容；当使用 KPPSoft 工具软件、其他计算机上位软件、触摸屏、显示器等通过通信方式修改这些特殊寄存器内容时，会同时修改 RAM 和 FlashROM 中的内容。

每次给 NK0 系列 PLC 上电时，NK0 系列 PLC 会自动从 FlashROM 读出有关的内容，写入对应的 RAM 型特殊寄存器中。但注意 NK0 运行模式切换时，不会复制 FlashROM 内容到 RAM 寄存器中。（注意：NK0 系统在执行各种设置、控制、运算操作时采用 RAM 存储器中的内容。）

另外注意如果 PLC 用户程序中有对特殊寄存器的写入指令，那 PLC 用户程序运行后，NK0 会根据程序运行的结果，修改 RAM 型特殊寄存器内容，但 FlashROM 内容不会被 PLC 程序修改。

当通过 KPPSoft 软件读取 NK0 中内容生成磁盘文件时（包括.KPP 和 .KAD 文件。），其只读取 RAM 型寄存器中的内容，当把磁盘文件内容写入 NK0 时，系统会把有关寄存器内容同时写入 RAM 型寄存器和 FlashROM 型寄存器中。

下面列出 NK0 所有的支持 FlashROM 存放的特殊寄存器。

特殊寄存器	用途	备注
R7633	BIN 运算符位	
R7650 ~ R7661	RS-485 (PORT0) 通信口参数	
R7706-R7717	以太网 (PORT3) 通信口参数	

通过通信方式修改 FlashROM 寄存器的注意事项：

- 1、通过通信方式修改 FlashROM 寄存器时，会暂停系统执行，影响和时间/时序相关的功能执行。所以要尽量避免在 NK0 处于用户程序运行模式（RUN 模式）时，去修改 FlashROM 寄存器。
- 2、一般 FlashROM 存储器都有写入次数限制，为确保产品的使用寿命，请尽量减少 FlashROM 存储器改写次数。

3-3-5 模拟量型 NK0 系列 PLC 模拟量数据的存取

对于以下模拟量型 NK0 系列 PLC，其模拟量数据的存取采用特殊寄存器直接存取的方式进行。

型号	模拟量规格	备注
NK0-CPU16DRA1 -D	4AD2DA, 4-20mA	4 路输入/2 路输出，
NK0-CPU16DRA2 -D	4AD2DA, 0-10V	4 路输入/2 路输出，
NK0-CPU16DD1A1 -D	4AD2DA, 4-20mA	4 路输入/2 路输出，
NK0-CPU16DD1A2 -D	4AD2DA, 0-10V	4 路输入/2 路输出，

每一种模拟量机型的 NK0，都带有 4 路输入/2 路输出，共 6 路模拟量通道。NK0 系统为这些模拟量通道数据的存取分配有以下一组寄存器。

模拟量通道类型	模拟量通道号	寄存器	数据格式
模拟量输入	输入通道 1	R7400	HEX
	输入通道 2	R7401	HEX
	输入通道 3	R7402	HEX
	输入通道 4	R7403	HEX
模拟量输出	输出通道 1	R7440	HEX
	输出通道 2	R7441	HEX

对于模拟量输入，系统会自动读入有关输入通道的模拟量数据，存放到相应的寄存器中，你只要在用户程序中直接读相应寄存器中的数值，就能得到相应的模拟量输入数据。

对于模拟量输出，你只要把有关数值送到相应的寄存器中，系统会自动把该数值转换为电压/电流值，输出到相应的模拟量输出通道。如果输出寄存器值超过最大值（0xFF），则系统把该数值作为最大值 0xFF 处理。

3-4 NK0 系列 PLC 用户存储器

在用户存储器中存放有控制 PLC 动作的用户程序和对系统的基本构成进行定义的系统参数。在 NK0 系列 PLC 中，这些都是存放于 FlashROM 型的用户存储器中。

程序存储区	2K 语	存放用户程序，包括主程序、子程序
系统参数区	512 语	存放系统参数

3-4-1 程序存储区

程序存储区主要用于存放用户编写的用户程序，用户程序主要由主程序和子程序组成。

0 地址	主程序区域： 从程序 0 地址到 END 指令间的内容，CPU 对此间的程序进行循环扫描执行。
END	
...	子程序区域： 在某个特定条件成立的情况下才执行的程序段。其位置处于主程序 END 指令之后，必须用梯形图语言作成（不能使用级式指令），包括： CAL 子程序：由 CLBL CEND 指令组定义的，由 CAL 指令在主程序中调用的子程序，NK0 最多可定义 256 个 CAL 子程序。 数据块定义：由 DLBL 指令定义的数据块，由相应的数据块传送指令读入累加器中，或与寄存器组之间进行数据传送。NK0 最多可定义 256 块数据。
CLBL	
CEND	
...	
...	
DLBL	
ACON	
NCON	

3-4-2 系统参数区

系统参数区是用来存放有关 NK0 系统的最基本的信息的区域，主要信息包括：用户程序名、密码、停电保持参数、本体通信口设置参数、监控定时器时间设定等。

主要系统参数

程序名	… 用户程序名称为 8 位以内的英文字母、数字
密码	… 程序保密用密码，8 位 BCD 数字
停电保持参数	… 各功能存储器的停电记忆保持范围的设定
通信口设置参数	… RS-485 通信端口的通信参数设定
监控定时器	… 对运算用处理器的延迟监视用定时器扫描时间的设定

1、用户程序名称

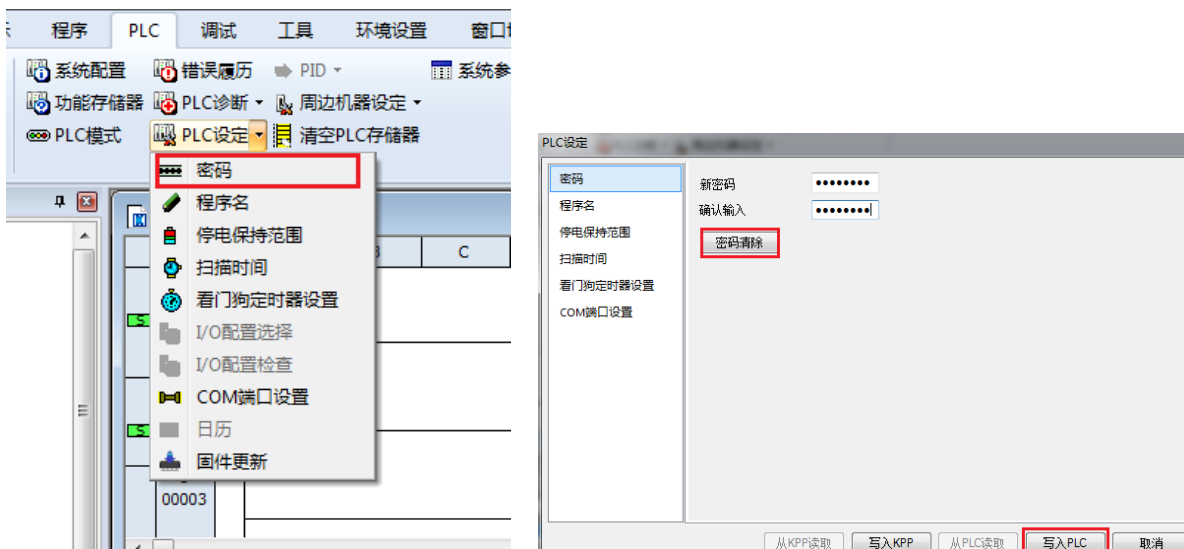
用 8 位以内的英文字母、数字登记，用于标记区分各用户程序。。

2、密码

密码用于限制通过 KPPSoft 软件或其他软件通过通信口对 NK1 的操作功能，用 8 个 BCD 码数字表示，出厂时全为零，表示没有设置密码。

1) NK0 密码的设置

你可以用 KPPSoft 来对 NK0 设置密码。设置密码时，首先要建立 KPPSoft 与 NK0 的通信连接，后选择[PLC]菜单下的[密码]子菜单，打开密码输入窗口（PLC 设定窗口），输入新的密码并重复输入新密码确认后，按[写入 PLC]键把密码写入到 PLC 中，NK0 密码设置完成。当断开 KPPSoft 与 NK0 的连接（或直接关闭 KPPSoft 软件）后，NK0 密码自动生效。



NK0 设置好密码后，你每次要用 KPPSoft 来连接该 NK0 的时候，KPPSoft 都会要求你输入正确的密码，如果密码输入不正确，KPPSoft 将不能与 NK0 建立连接，无法看到 NK0 中的用户程序，也无法用 KPPSoft 来监视修改 NK0 中的程序、参数等内容。但 NK0 设置密码后，仍然可以使用通信连接设备如触摸屏、显示器等通过通信口来监视、修改 NK0 内部功能存储器的状态。

建立连接后，你可以再次通过输入密码操作来修改原来的密码。

如果要删除 NK0 中的密码，可以在建立连接后，进入密码输入窗口，但不输入密码，直接点击[清除]按键，这样就可以删除 NK0 中原来的密码。

2) 恢复出厂设置超级密码

如果忘记了密码，不能读出 NK0 中的程序的内容，你可以使用这儿介绍的超级密码，来清除 NK0 中的所有内容，并恢复到产品出厂时的初始状态。

设置方法：把模式开关拨到“STOP”位置并确认 NK0 处于“STOP”状态。（如果原来模式开关就在 STOP 位置，但 NK0 为 RUN 状态，你可以先把模式开关打到“RUN”位置，再拨回到“STOP”位置，确保 NK1 处于“STOP”状态），然后连接 KPPSoft 软件，当提示输入密码时，在密码框中输入 8 个‘C’（大写），提示确认后 NK0 会清除以前设置的密码，同时程序及寄存器数据也随之清除，系统参数被恢复到出厂值。

注意：使用恢复出厂设置超级密码后，被清除的程序、数据将不能恢复！

3) 程序锁定功能

有些场合我们需要使用 KPPSoft 来进行一些数据监视及修改功能，但是不希望能够看到程序，这时使用密码功能显然不能满足这种应用要求；另外在很多时候，为了保护软件知识产权，我们不想下载到 NK0 系列 PLC 中的程序能够被随意读出来。

为了满足这样的需要，NK0 提供了一种程序锁定保护的功能，在向 NK0 系列 PLC 内传送程序的时候，会出现一个是否把程序写入 PLC 的确认窗口，勾选其中的[程序加锁]选项后（初始未勾选），那你传送到 NK0 系列 PLC 中的程序将被锁定，下次再有读出 NK0 中程序请求时，KPPSoft 会给出一个“程序包已被锁定”信息，终止程序的读出操作。



注意：由于锁定的程序将不能被读出，所以使用程序锁定下载以前，请务必先保存工程文件，否则，你的程序将不能再读出。

3、停电保持参数

NK0 的一些功能存储器，包括 M、R、T、C、S 等 5 种功能存储器的内容，可以设置成在系统掉电的情况下，还可以长久保持其内容不消失，这 5 种功能存储器的停电保持范围可以由用户自由设定。

	存储器类型	出厂初始设置	允许设定范围
停电保持区域	内部继电器	M300~M377	M0000~M1777
	数据寄存器	R2000~R2027	R2000~R7777
	定时器	无	T000~T177 (R0~R177)
	计数器	C000~C17	C000~C177 (R1000~R1177)
	级	无	S000~S377

NK0 无须另配电池，就可以实现停电保持区域的停电保持，但注意 NK0 最大只支持 64 字内容的停电保持。在使用 KPPSoft 软件设置 NK0 停电保持数据范围时，如果超过了 64 字的范围，那 KPPSoft 会报“停电保持设定超过最大范围”错误，并忽略本次停电保持数据范围参数修改操作。

对于前面 3-3-4 节介绍的 FlashROM 型特殊寄存器，注意即使其没有被设置在停电保持区域内，其内容在断电时也会被保存下来，这点在使用时要引起注意。

在计算停电保持数据范围时，按以下规则计算数据大小：点状态（内部线圈、级）以 16 点为 1 组计算成一个字，不满 16 点的尾数也占一个字；数据寄存器每一个寄存器为一个字；对于定时器/计数器，每一个定时器/计数器包括其接点状态信息（点状态）和经过值寄存器内容，所以每保存 16 个定时器/计数器内容需要使用 17 个字（16 个接点状态占一个字，16 个经过值寄存器需要 16 个字）。

5、通信口设置参数

NK0 带有 1 个 RS-485 通信口，1 个以太网通信口（仅 NK0-CPU28DR-D），这些通信口的通信参数可以通过 KPPSoft 软件的[COM 端口设置]菜单项来设置，也可以通过用户程序设置特殊寄存器的方式来设置。所有的设置参数全存放在这儿，包括：通信协议的选择，通信速度/数据位/停止位/校验位/延时时间/存放寄存器/无协议通信结束码/等等内容。具体设置方法请参考本资料第四章有关的内容。

6、监控定时器

监视用户程序执行时的运算专用处理器的延迟时间，用于检出由于程序错误而使处理进入无限循环导致失控的情况。该定时器的设定以 2ms 为单位进行设定，最大可设定为 9998ms。实际设定时，可在程序试运行后，确认下 PLC 扫描时间，然后设定一个适当的数值。出厂的初始设置值为 200 ms。

在 NK0 出厂时，系统参数都有一个出厂初始设置值，你可以通过 KPPSoft 编程软件来改变这些设置值以满足你的控制系统的应用需要，你也可以在需要时把这些参数恢复到出厂初始设置值。你可以使用 KPPSoft 软件的[系统参数初始化]操作来把这些参数恢复到出厂初始设置值。下表列出 NK0 各系统参数的可设定范围以及其出厂初始值。

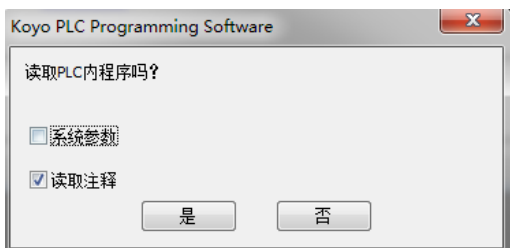
项 目		初始值（出厂值）	设 定 范 围
程序名		未登记	8 位以内英文字母、数字
密码		00000000（未登记）	8 位数字（BCD）
停电保持区域	内部继电器	M300~M377	M0000~M2777
	数据寄存器	R2000~R7377	R2000~R7777
	定时器	无	T000~T177（R0~R177）
	计数器	C000~C777	C000~C177（R1000~R1177）
	级	无	S0000~S377
通信口设置参数		P0: MODBUS 协议+编程专用协议 1 位停止位/无校验/19200bps/1 号局 无协议模式 70 P3: ECOM 协议（固定）节点号：1 IP 地址分配：手动 IP 地址：192.168.1.7 子网掩码：255.255.255.0 网关：192.168.1.1 协议：MODBUS 协议+编程专用协议 1 位停止位/无校验/19200bps/1 号局	通信协议的选择，通信速度/数据位/停止位/校验位/延时时间/存放寄存器/无协议通信结束码等
监控定时器		200ms	2~9998ms

3-4-3 系统参数区和用户程序传送

系统参数是 PLC 工程文件的重要组成部分，它用来设置 PLC 上各功能部件的工作参数。

KPPSoft 工具软件一般把系统参数区内容与用户程序看成一个整体，这样每次传送 PLC 程序时，其系统参数的内容同时被传送。另外在用 KPPSoft 工具软件连接 NK0 系列 PLC 时，如果两者间的用户程序或系统参数内容有不一致，KPPSoft 软件多会报程序不一致信息。

但有些应用场合我们在传送（上传/下载）用户程序的，不希望同时传送系统参数，要实现这个功能，只要在执行程序上传/下载操作时出现的设置窗口中，把“系统参数”项的勾选去掉，即可实现系统参数的不传送。



注意：我们不能只传送系统参数而不传送用户程序！

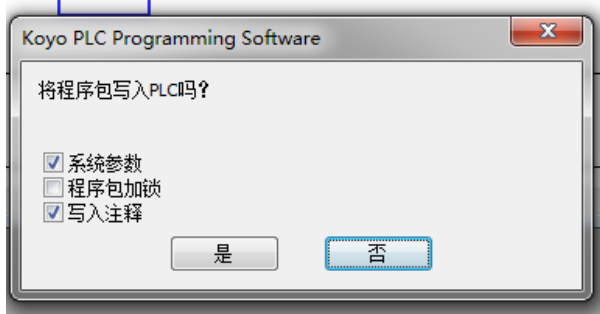
另外，KPPSoft 软件提供了系统参数的离线编辑功能，编辑好的系统参数和用户程序一起存为磁盘文件后，就可以直接一起传送到 PLC 中，这样更方便工程设计者的使用。

在使用 KPPSoft 软件的系统参数离线编辑功能时注意，在[PLC 设定]窗口中离线设置好参数后，一定要点击窗口下面的[写入 PC]键后，该参数才能先保存在 KPPSoft 软件内部，后在保存工程文件时和用户程序同时保存到磁盘上。

保存到磁盘工程文件中的系统参数，在工程文件被读入 KPPSoft 工具软件后，一定要在[PLC 设定]窗口中点击[从 PC 读取]键后，才能被 KPPSoft 工具软件读出，显示在[PLC 设定]窗口中。

3-4-4 程序注释

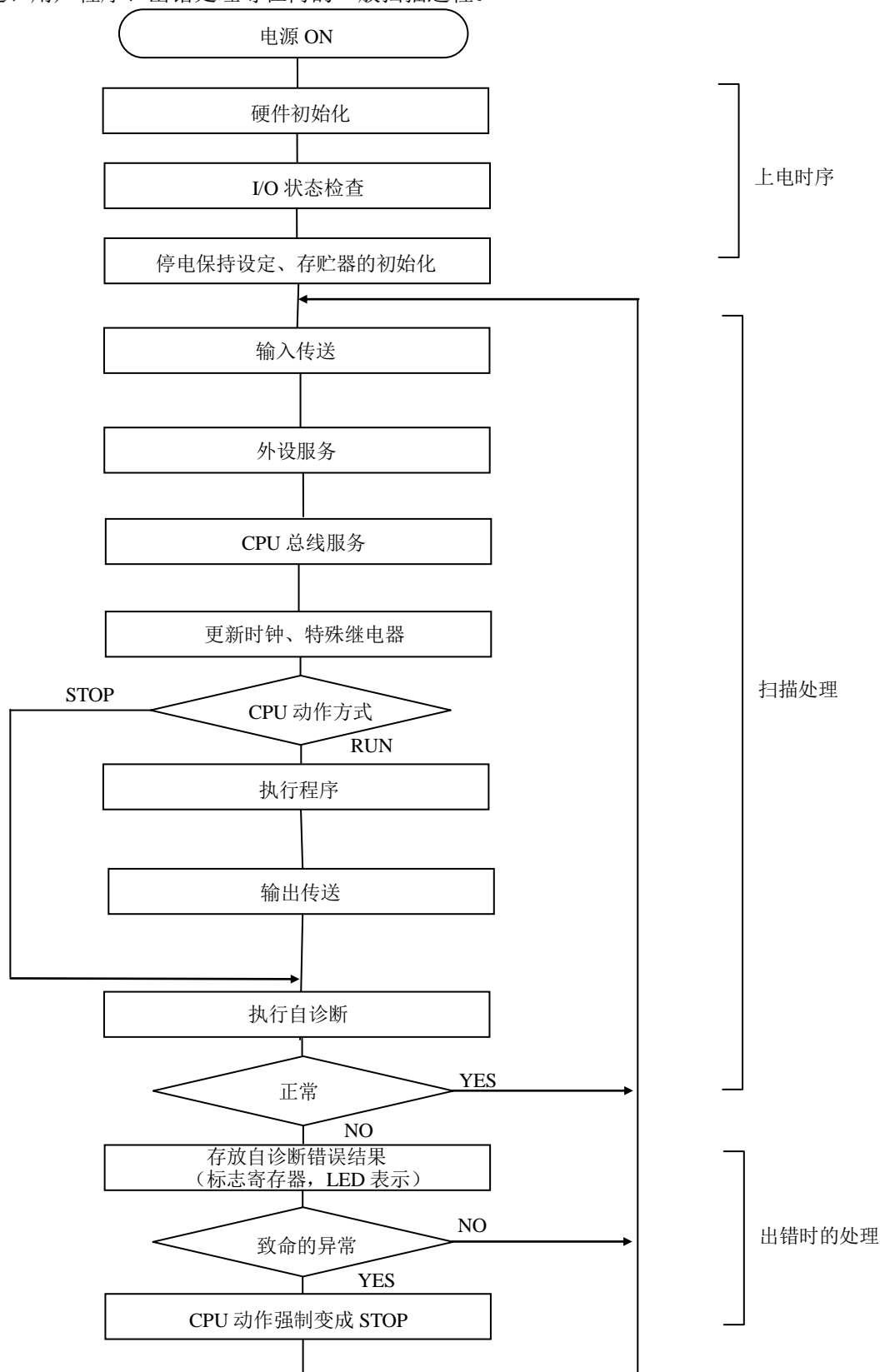
NK0 支持把编制用户工程时编写在工程中的程序/回路/级注释、以及功能存储器相关的变量名称/接线信息/注释等内容和用户程序、系统参数等一起传送到 NK0 系列 PLC 中，以方便将来的工程维护修改工作。当然你也可以选择不把这些注释内容传送到 NK0 系列 PLC 中。选择方法很简单，在把工程包数据写入 PLC 时，会出现如下的选择窗口，其中的“写入注释”选项就是用来选择是否把注释内容下载到 PLC 中去。缺省设置是一起进行下载。如果你不希望下传注释，则只要去掉这个勾选选项就可以了。



3-5 NK0 系列 PLC 扫描及输入输出传送

3-5-1 NK0 系列 PLC CPU 的执行处理

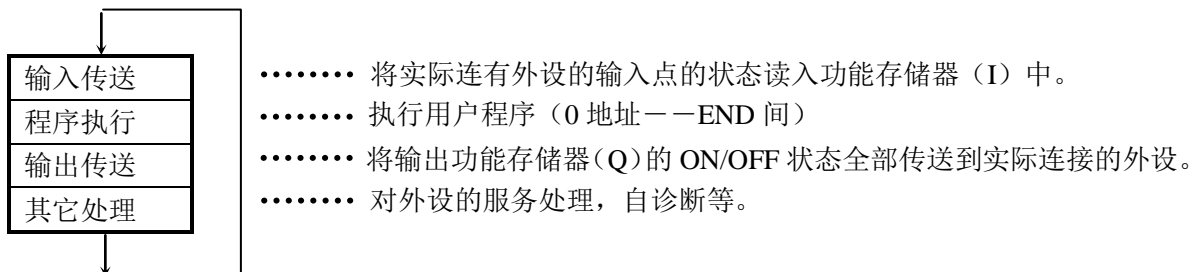
NK0 系列 PLC 上电后，其 CPU 按以下顺序执行：先执行上电时序段程序，后循环执行包括输入输出传送、用户程序、出错处理等在内的一般扫描过程。



3-5-2 循环扫描

NK0 系列 PLC 的动作方式进入运行 (RUN) 状态后, 对用户程序从起始地址开始, 按序执行到 END 指令处, NK0 系列 PLC 反复进行这样的处理 (循环)。另外, NK0 在运行状态下, 除执行用户程序外, 还要进行输入/输出的状态读/写处理, 输入与输出传送及其它服务等。

其主要过程如下:



当 NK0 系列 PLC 处于用户程序运行方式时, 上述过程是一直循环进行的, 因此称为循环扫描, 我们把一次这样的处理称为一次扫描, 执行一次这样的过程的时间称为扫描时间或扫描周期。扫描周期的长短主要取决于用户程序的长短和所使用指令的类型。NK0 系列 PLC 的典型扫描周期为 1ms /1K 语。扫描周期存放于下列寄存器中:

R7775: 当前扫描时间 (ms)

R7776: RUN 开始后最短扫描时间 (ms)

R7777: RUN 开始后最长扫描时间 (ms)

在 NK0 系列 PLC 内部, 有一个对扫描进行监视的监控定时器 (Watchdog Timer), 假如一次扫描在规定时间内 (如 200ms) 不能完成, 则认为是 PLC 系统故障, 停止运行。(监控定时器的时间设定在系统参数区进行。)

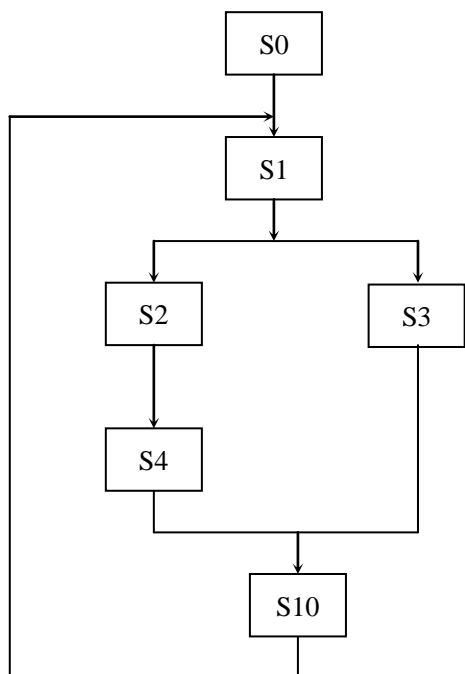
PLC 这样的用户程序处理方式我们称为循环扫描方式。循环扫描是 PLC 的常规处理方式。

3-5-3 级式指令和扫描

循环扫描在其程序内部都是一种顺序扫描的指令执行方式，编程人员不能对程序执行的流程进行控制。但在某些场合，若对执行次序进行控制，则能使动作顺序简单明了，实现高速处理。

级式语言是把动作按工序一步一步进行分解，并编制相应程序，然后按工序执行顺序和工序转移条件连接起来完成控制目的的编程语言。其每一工序对应于不同的级，由于工序的唯一性，表示工序的级号不能重复。这种以级为单位的程序（块），根据级的状态为 ON 或 OFF 而执行或不执行该级内的程序。这种级的执行和流向则可以由编程者进行控制，包括并行运行、跳转、根据不同的条件分流、合流等，因为只有那些为 ON 的级，其内部程序才会被执行。因此，级式语言使得编程根据工艺要求进行而变得简单且运行速度加快。

在同一时刻，可以有一个至多个级为 ON，这在级号允许范围内没有限制，而这时这些级的状态是并行运行的，但在程序的分布上有先后次序，PLC 则依照循环扫描的原理从前往后扫描，这是程序的执行过程，但从宏观上来看，各个级的状态由级式指令控制后，程序的执行有了跳越，不再是所有的程序指令在每次扫描中都被扫描、执行，这样实现了高速化。



由于原来相互连锁的动作可以编写在不同同时运行的级中，因此，各个动作的连锁不再那么复杂、繁琐，而由级的 ON/OFF 状态自动控制。当某个级的状态由 ON 变为 OFF 时，该级的 OUT、TMR 指令线圈全部自动复位，因此，级式语言也使得编程更简单。

级式语言编程是在原梯形图指令体系中增加了 ISG、SG、JMP、NJMP、CV、CVJMP、BREQ、BSTART、BEND 等指令而构成的，级内程序仍由梯形图程序组成。

级式语言是光洋系列 PLC 特有的编程语言方法，利用好级式语言编程，可以使得 PLC 的控制程序更富条理性，逻辑性更强。

有关级式语言的具体编程方法请参见以下资料：

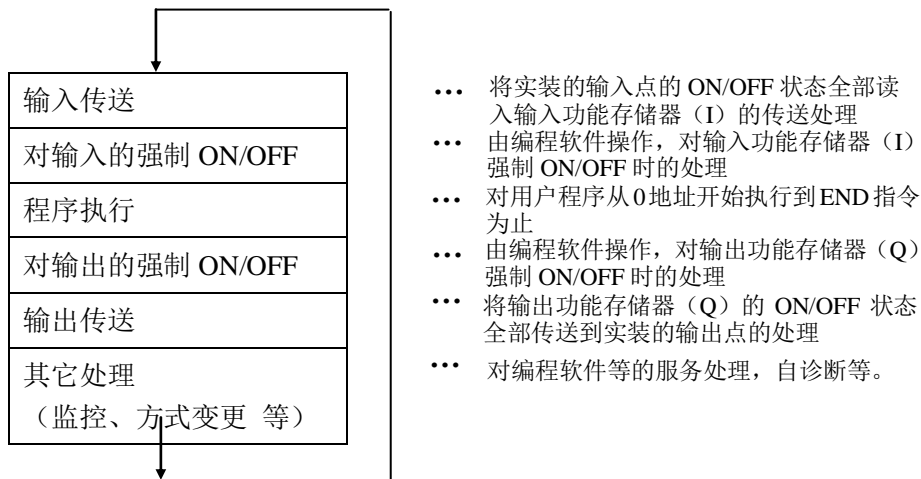
《级式语言编程指导》

3-5-4 输入输出传送

NK0 系列 PLC 对输入/输出状态的读/写方法，有成批传送方式（一次全部传送）和直接处理方式 2 种，使用不同的指令来区分不同的读/写方法。

在通常的 PLC 循环扫描过程中，NK0 在用户程序执行前，读入全部输入的状态（成批传送），在指令执行过程中，功能存储器（I）的 ON/OFF 状态是不变的，程序执行完后送出全部输出的状态。

CPU 的扫描，如下图进行。



- * 输入传送及输出传送，只对实装 I/O 点进行处理
- * 扫描时间，存入下列特殊寄存器中。（16 进制数）
 - R7775: 现在的扫描时间 (ms)
 - R7776: RUN 开始后最短的扫描时间 (ms)
 - R7777: RUN 开始后最长的扫描时间 (ms)

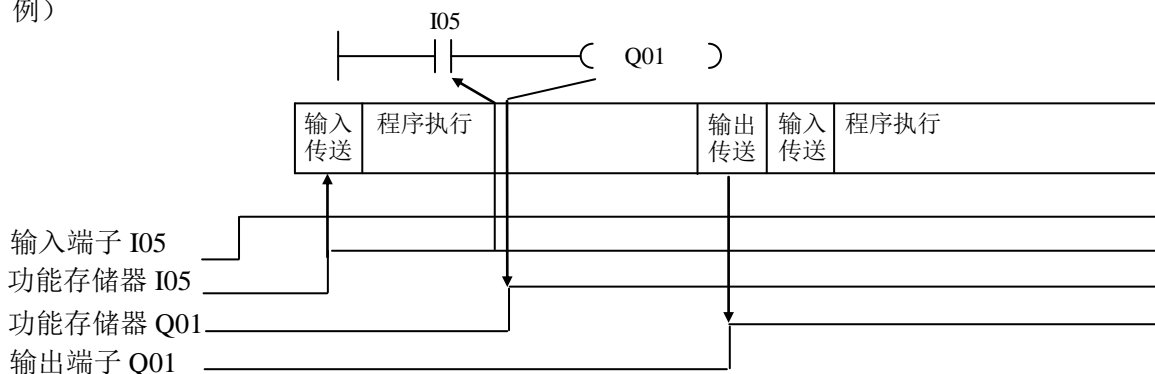
3-5-5 成批传送方式

成批输入传送是在一次扫描的最初进行的，程序执行中输入功能存储器（I）的 ON/OFF 状态（状态）是不变化的。

（即在同一次扫描中，输入（I）的接点条件是相同的。）

成批输出传送，是在一次扫描的最后将程序执行结果从输出功能存储器传送到输出点。在通常的程序中，使用成批传送的指令。

例)



3-6 NK0 系列 PLC CPU 动作模式

NK0 系列 PLC 根据需要，可以选择 TERM-RUN，TERM-STOP，TEST-RUN，TEST-HALT，TEST-STOP 等 5 种不同的动作模式，你可以通过 NK0 本体上的运行模式开关或通过 KPPSoft 编程软件来选择你需要的动作模式。

其中 TERM-RUN，TERM-STOP 为正式运行动作模式，可以通过 NK0 本体单元上的运行模式开关或 KPPSoft 编程软件来选择；TEST-RUN，TEST-HALT，TEST-STOP 是程序运行初期的调试运行模式，只可以通过 KPPSoft 编程软件来选择。

下表列出各工作模式下 NK0 系列 PLC 的动作内容。

CPU 动作模式	内 容
TERM-RUN	<ul style="list-style-type: none"> • 循环进行用户程序执行、输入输出传送的动作模式。 • 可通过编程软件、通信及本体模式开关改变运行模式。
TEST-RUN	<ul style="list-style-type: none"> • 在程序调试及试运行时的运行模式。 • 循环进行用户程序的执行、输入输出传送。 • 可设置断点执行程序，可进行各种调试运行操作。
TEST-HALT	<ul style="list-style-type: none"> • 在程序调试及试运行时的暂停运行模式。 • 程序执行过程中的临时停止状态。设置断点 TEST-RUN、单步执行、指定位置程序执行后进入该模式。 • 各功能存储器状态及输出点状态保持。
TEST-STOP	<ul style="list-style-type: none"> • 在程序调试及试运行时的停止模式。 • 可进行程序的写入、修改的停止状态。 • 不对功能存储器进行初始化处理，修改好程序后，可以继续程序的调试运行。 • 各功能存储器状态保持，输出点 OFF。
TERM-STOP	<ul style="list-style-type: none"> • 进行程序编辑，参数设定时用的模式。 • 功能存储器的状态，除停电保持区域外，全部被初始化。 • 输出全部 OFF。 • 可通过编程软件、通信及本体模式开关改变运行模式。

1、用模式运行开关切换选择 NK0 动作模式

NK0 系列 PLC 本体上有一个 2 位置开关，可以在正式运行时来选择 NK0 系列 PLC 的动作方式。NK0 通过该模式开关可以选择为 TERM-RUN 动作方式或 TERM-STOP 动作方式。



当把该模式开关打到 STOP 位置时，则 NK0 进入 TERM-STOP 动作模式；当把该模式开关打到 RUN 位置时，如果 NK0 系列 PLC 的软硬件系统正常，自检没有问题，则 NK0 进入 TERM-RUN 动作模式。

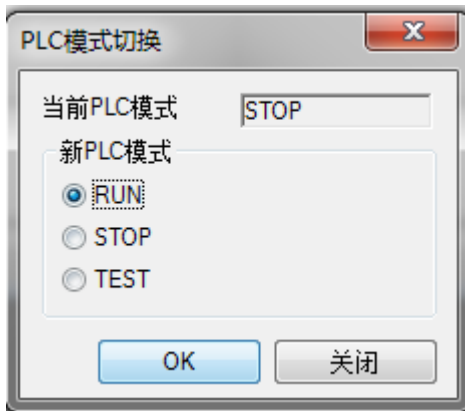
注意：

- 1)、不管该模式开关位置如何，都可以通过 KPPSoft 软件来改变 NK0 的动作模式；
- 2)、当 NK0 处于 TERM-RUN 动作中而该模式开关处于 STOP 位置时，可以通过把该模式开关先打到 RUN，然后再打到 STOP 位置的方法，来改变 NK0 的动作模式为 STOP；
- 3)、当 NK0 处于 TERM-STOP 动作中而该模式开关处于 RUN 位置时，可以通过把该模式开关先打到 STOP，然后再打到 RUN 位置的方法，来改变 NK0 的动作模式为 RUN。

2、用 KPPSoft 软件选择 NK0 动作方式

通过 KPPSoft 软件可以选择 NK0 系列 PLC 的 3 种运行模式：RUN、TEST、STOP，如右图。RUN 为 TERM-RUN 动作模式；STOP 为 TERM-STOP 动作模式；TEST 模式下包括 TEST-STOP、TEST-HALT、TEST-RUN 等 3 种动作模式，在 TEST 模式下，可选择各种调试操作（扫描执行、1 次/N 次扫描、单步执行、执行到指定位置等等）。

注意：不管 NK0 本体上的模式开关处于哪个位置，都可以通过 KPPSoft 来选择 NK0 的动作模式，即 NK0 的动作模式与其模式开关的位置不是一一对应关系。



3、上电时 NK0 动作模式

NK0 上电时，将根据其本体上的动作模式开关的位置来进入不同的动作模式。

NK0 本体动作模式开关位置	NK0 内部情况	NK0 上电动作模式
STOP	无关	TERM-STOP
RUN	无用户程序或系统自检有问题	TERM-STOP
	用户程序正确，自检没有问题	TERM-RUN

1、CPU 动作模式与功能存储器状态

下表列出 NK0 系列 PLC 在不同的动作模式下，各功能存储器的状态。

CPU 动作模式	输入	输出	其它的功能存储器及数据寄存器
TERM-RUN	输入传送	程序执行结果	程序执行结果
TEST-RUN	输入传送	程序执行结果	程序执行结果
TEST-HALT	输入传送	程序执行结果	程序执行结果
TEST-STOP	输入传送	全部 OFF	程序执行结果、停止前的值
TERM-STOP	输入传送	全部 OFF	OFF、0(停电保持区域保持停止前的值)

3-7 NK0 系列 PLC 特殊功能

NK0 系列 PLC 具有一些有用的特殊功能，利用好这些功能，可以帮助你在开始时更好地设置系统，编制、调试程序；在发生问题时，尽快找到并解决问题。这样的功能包括：在线程序调试功能、离线模拟运行功能、工程文件包生成/下载功能、自诊断功能、履历情报保存/显示功能等等，下面介绍这些功能的特点和其用法。

3-7-1 在线程序调试功能

调试功能，是为发现程序的错误或跟踪确认程序的执行结果而设置的功能。运行用户程序后，通过观察、修改各功能存储器的值来观察程序的执行结果并修改程序使得程序的执行结果满足设计要求的过程就是一个调试的过程。

调试功能一般需要调试工具的支持。NK0 系列 PLC 支持在线程序调试运行（TEST 模式）和离线模拟运行 2 种调试功能，2 种调试功能都需要使用 KPPSoft 编程软件。

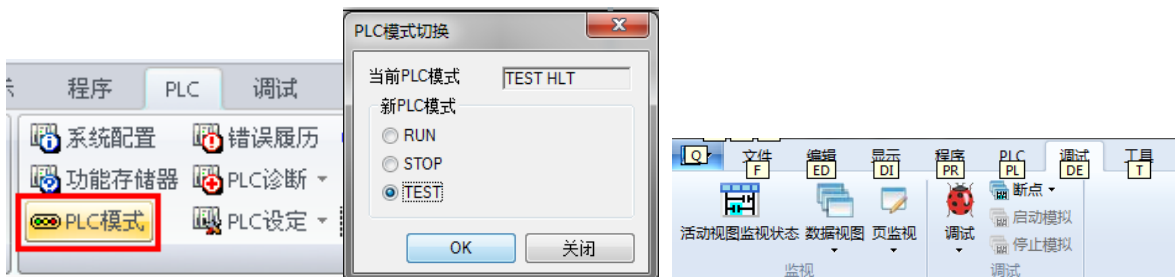
本节介绍在线程序调试功能，离线模拟运行功能下节介绍。

NK0 在线调试包含 3 种运行模式：TEST-RUN，TEST-HALT，TEST-STOP。

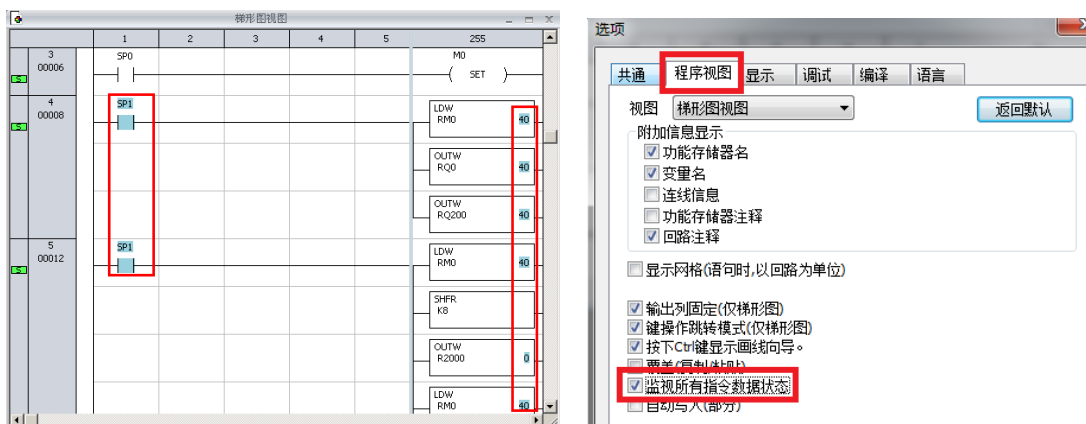
要使用 NK0 系列 PLC 的在线程序调试功能，需要使用 KPPSoft 工具软件连接 NK0 来实现。

用 KPPSoft 软件连接好 NK0 系列 PLC 后，通过 KPPSoft 软件[PLC->PLC 模式]菜单打开[PLC 模式切换]窗口，选择 NK0 的运行模式为 TEST 运行模式（这时 NK0 上看不出有显著变化）。

注意：选择 TEST 模式前，如果 NK0 处于 STOP 模式，则进入 TEST-STOP 模式；如果 NK0 处于 RUN 模式，则进入 TEST-RUN 模式。（TEST-HALT 模式不能直接选择进入。）



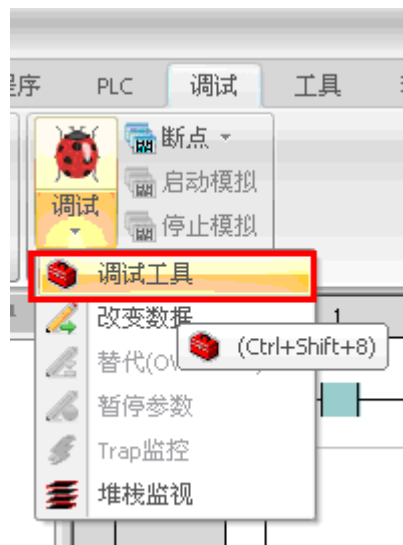
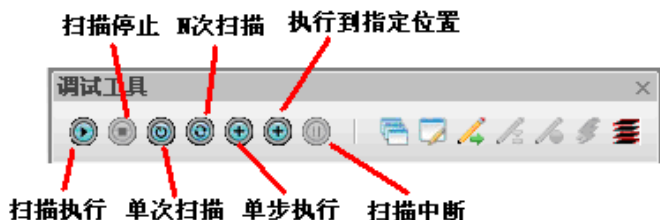
为了调试时方便观察 PLC 各数据的变化情况，你可以用[调试->活动视图监视状态]菜单项打开梯形图监视功能，来监视梯形图上各接点的通断情况以及各寄存器数据的变化情况。



如果你的监视页面上不能监视到寄存器值，则需要通过[环境设置->选项]菜单打开[选项]窗口，选择[程序视图]页，确认[监视所有指令数据状态]处于勾选状态，以保证在梯形图上监视到各寄存器内数据的变化（如上右图）。

在调试过程中，需要时，你还可以通过[数据视图]、[值修改]来观察、修改各功能存储器的内容；通过[页监视]来按页观察同一类功能存储器的内容。

接下来，你可以再通过 KPPSoft 软件[调试->调试工具]菜单项打开调试工具条，选择进行各项程序调试操作，包括：程序扫描执行（TEST-RUN，可设置多个断点）、单次扫描执行、N 次扫描执行、单步指令执行、执行程序到指定位置等。



A) TEST-RUN（扫描执行）

TEST-RUN 运行，是在用户系统投入运行前进行的调整程序的运行模式。

在 TEST-RUN 时，NK0 扫描执行所有用户程序，发现问题后，可执行[扫描停止]操作，使 NK0 转入 TEST-STOP 模式，此时功能存储器的状态全保存（但外部输出全为 OFF）。改正程序后，再点击[扫描执行]，进入 TEST-RUN 模式，可使程序顺着扫描停止时的参数继续动作。

NK0 支持断点设置功能，可以设置多个动作执行断点，设置好断点后，执行 TEST-RUN，则 NK0 在运行到第一个断点处后，自动停止 TEST-RUN，进入 TEST-HALT 模式，你可以通过 KPPSoft 软件观察此时的程序执行结果。再点击[扫描执行]，执行 TEST-RUN，则 NK0 自动执行程序到下一个断点处停止进入 TEST-HALT 模式。如此执行到最后断点后再点击[扫描执行]，执行 TEST-RUN，则 NK0 循环执行用户程序，停止在程序第一个断点处。

NK0 在 TEST-RUN 模式下，一旦设置好一个断点，则该断点马上有效，NK0 在下次扫描中，执行到该断点处，就停止程序运行进入 TEST-HALT 模式。

TEST-RUN、TEST-HALT 模式下，所有程序执行结果保持、输出点动作保持；

TEST-RUN 模式下，点击[扫描停止]，NK0 进入 TEST-STOP 模式。

TEST-STOP 模式下，所有程序执行结果保持、但输出点动作全部为 OFF。

注意：在 TEST-HALT 模式下时，不能直接执行[扫描停止]动作，进入 TEST-STOP 模式。而要先删除所有断点信息，点击[扫描执行]，进入 TEST-RUN 模式，后点击[扫描停止]，进入 TEST-STOP 模式。

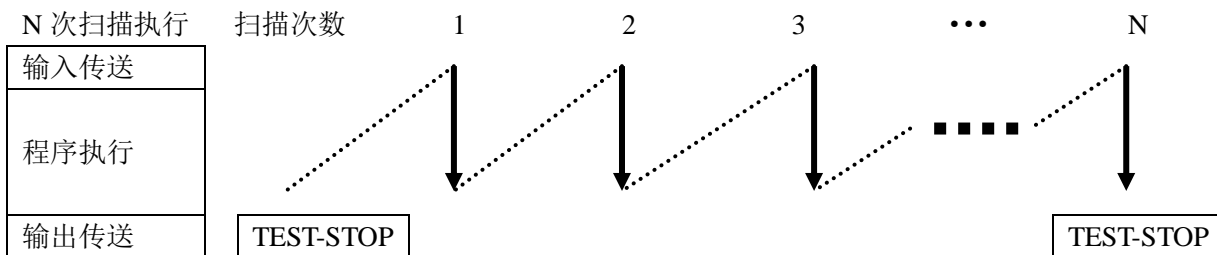
B) 单次扫描执行/N 次扫描执行

① 单次扫描是从零地址开始到 END 为止执行一遍后停止。

② N 次扫描执行是预先设定扫描次数，在执行了指定的扫描次数后停止。

点击调试工具栏的[单次扫描]按钮，执行单次扫描动作；

点击调试工具栏的[N 次扫描]按钮，输入需要执行的扫描次数，执行 N 次扫描动作。



NK0 完成单次/N 次扫描动作后，进入 TEST-STOP 模式。

C) 单步执行

执行单条程序指令的动作。你可以通过 KPPSoft 的单步执行功能来一条条执行程序指令，并通过 KPPSoft 来观察每条指令执行后，各功能存储器以及累加器的状态内容。

当 NK0 处于 TEST-STOP 或 TEST-HALT 模式下时，可以执行该单步执行动作，完成单步执行动作后，NK0 进入 TEST-HALT。

D) 指定程序地址执行

把梯形图显示页面中的光标移动到指定地址程序处，点击[执行到指定位置]，则 NK0 执行程序到光标指令处停下，此时，你可以通过 KPPSoft 来观察各功能存储器以及累加器的状态内容。

注意，如果光标位置处没有指令，点击[执行到指定位置]，将不执行任何操作。

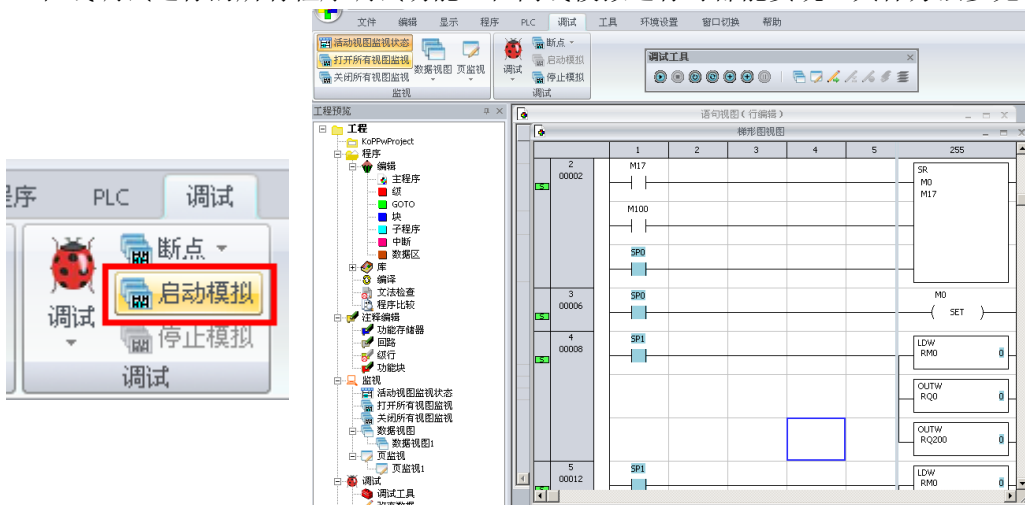
当 NK0 处于 TEST-STOP 或 TEST-HALT 模式下时，可以执行该动作，完成该动作后，NK0 进入 TEST-HALT。

3-7-2 离线模拟运行功能

使用离线模拟运行功能，可以在没有连接 NK0 系列 PLC 的情况下，用 KPPSoft 软件来模拟 NK0 用户程序的执行情况，以达到调试用户程序的目的。离线模拟运行功能只能在离线状态下使用。

要使用离线模拟运行功能，首先要在 KPPSoft 软件中启动该功能。点击 KPPSoft 软件[调试->启动模拟]菜单项，就启动了 KPPSOFT 对于 NK0 的离线模拟运行功能。

在线调试运行的所有程序调试功能，在离线模拟运行时都能实现，具体方法参见上节内容。



3-7-3 工程文件包生成/下载功能

通过 KPPSoft 软件，可以编制 NK0 系列 PLC 的用户工程文件，下载到 NK0 系列 PLC 中，并在需要时存成磁盘文件（.KPP 文件），以便留档备份。.KPP 文件可以再次用 KPPSoft 软件打开编辑修改，并下载到 NK0 系列 PLC 中。

有的时候，我们需要生成一个工程文件包，该工程文件包可以传给别人下载到 NK0 中去，但是别人不能够打开该工程文件包，看到程序代码。KPPSoft 软件的工程文件包生成/下载功能，就能实现这样的应用目的。

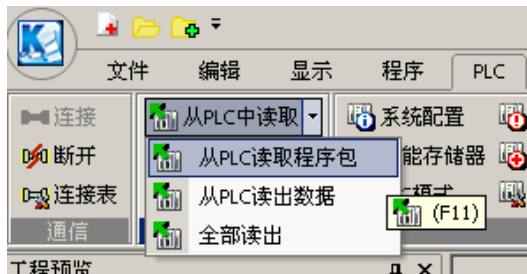
利用 KPPSoft 软件，可以生成一个只能用于文件下载，而不能打开编辑修改的工程文件包（.KAD 文件），该工程文件包可以通过 KPPSoft 软件下载到 NK0 中。

下面详细介绍利用 KPPSoft 软件，实现 NK0 系列 PLC 工程文件包的生成/下载功能。

1、NK0 系列 PLC 工程文件包（.KAD 文件）的生成

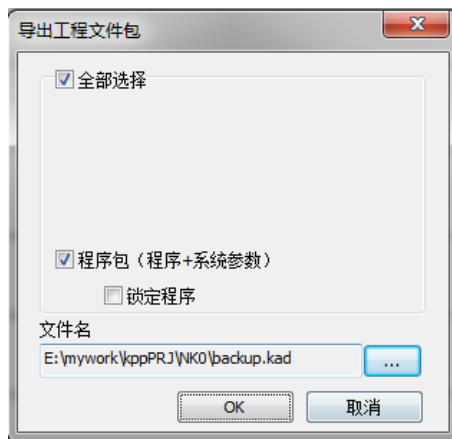
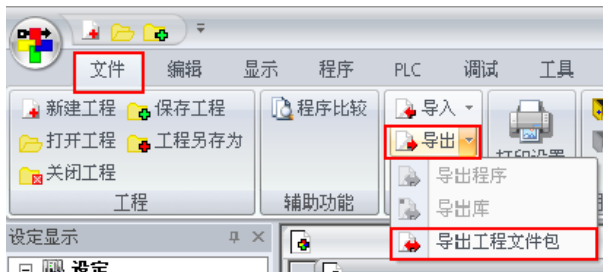
NK0 的工程文件包可以包含用户程序、从 NK0 系列 PLC 读出的 PLC 存储器信息等。

用 KPPSoft 软件编制好用户程序，需要时连接 NK0 系列 PLC，读出 NK0 内的程序包（包括用户程序和系统参数）和功能存储器内容。通过 KPPSoft 软件的“PLC→从 PLC 中读取”菜单下的各子菜单读取需要的内容，如右图。注意，如果该 NK0 中的程序被锁定的话，你将不能读出程序包，但仍然可以选择“从 PLC 读出数据”菜单打开[从 PLC 中读取]窗口，设置需要读取的存储器数据并读出相关数据到 KPPSoft 系统内存中，同时可直接把读取的数据生成到指定的.KAD 文件中。



然后，选择 KPPSoft 软件的“文件→导出→导出工程文件包”菜单，会弹出[导出工程文件包]设置窗口，选择需要装入工程文件包的内容，并选择生成工程文件包的名称和存放路径后，点击“OK”按键，KPPSoft 软件就会自动在磁盘指定位置生成一个指定文件名称的.KAD 工程文件包。

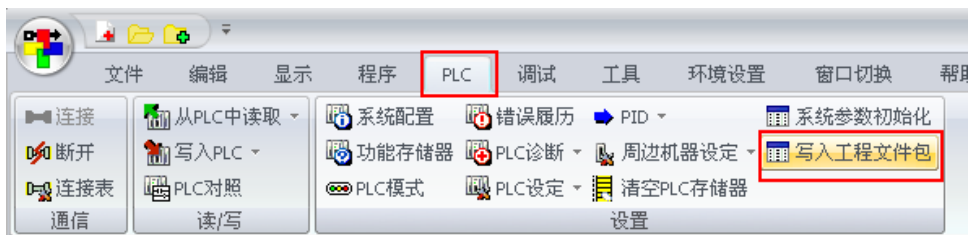
注意：如果勾选了“程序锁定”项，则该.KAD 文件在下载到 NK0 中后，该 NK0 中的用户程序以后将无法被用 KPPSoft 读取。



2、通过 KPPSoft 软件下载.KAD 工程文件包

生成的.KAD 工程文件包，可以直接通过 KPPSoft 软件，下载到所连接的 NK0 系列 PLC 中。

下载时，先连接 KPPSoft 软件和 NK0 系列 PLC，选择 KPPSoft 软件的“PLC→写入工程文件包”菜单，KPPSoft 软件操作界面如下：



弹出[文件打开]选择窗口，选择需要下载到 NK0 中去的.KAD 工程文件包名称，点击“打开”按钮，KPPSoft 软件会自动把所选择的.KAD 工程文件包内容下载到 NK0 系列 PLC 中去。下载过程中，软件下方的“通信”信息窗口中会显示文字信息，以说明下载的工作进程，同时画面上会出现如右下图所示的下载进程条显示窗口。



如果选择打开的文件内容不对，KPPSoft 软件会打开诸如“PLC 类型不匹配”之类的错误信息提示窗，点击提示窗中的“确认”键，关闭提示信息窗。

注意，写入完成后，如果下载的 KAD 文件中包含了密码，则 KPPSoft 会在下载完 KAD 文件后，弹出一个密码输入窗口，要求你输入密码，如果密码输入正确，则会建立连接，读出 NK0 中程序和系统参数，如果密码不正确，则会断开 KPPSoft 软件与 NK0 系列 PLC 的连接。

如果下载的 KAD 文件中没有密码，KPPSoft 软件并不打开显示所下载的工程文件包。要在 KPPSoft 软件上显示所下载的工程文件包内容，需要通过“PLC→从 PLC 中读取”菜单下的各子菜单读取需要的内容。如果下载工程文件包中的程序是被锁定的，则你将读不出程序来。

3-7-4 自诊断功能

NK0 系列 PLC 带有自诊断功能，NK0 在上电后，每隔一段时间，会对 NK0 自身的当前状态、用户程序等进行检测，当发现异常情况后，会自己报告错误代码，存放于特殊寄存器内，用户可以通过 KPPSoft 编程软件读出错误代码，以了解发生的错误内容。

另外异常时 CPU 面板上的状态 LED 或相关特殊线圈 SP 置为 ON。

发生致命错误的场合，扫描停止，转为 STOP 模式（TERM-STOP）。

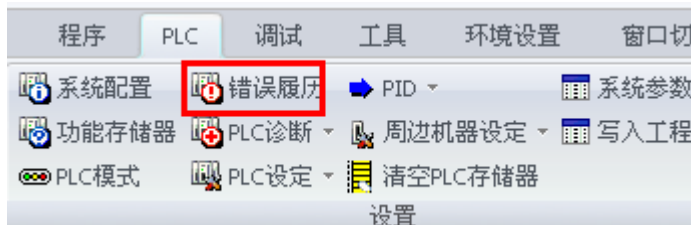
错误级别	检出内容	检出时间	CPU 的运行	异常继电器	错误代码 存储寄存器
致命	CPU 异常 内置看门狗时间溢出	常时	停止	SP51	R7755 (E003)
致命	MEN 异常 程序存储器奇偶出错	常时	停止	SP44	R7755 (E155)
	MEN 异常 没有用户程序	RUN 开始时	停止	SP31	R7755 (E401)
非致命	接收的数据出错 (通过以太网连接时)	常时	继续	SP46	R7756 (E312)
非致命	通信异常 接收地址出错	常时	继续	SP46	R7756 (E313)
非致命	通信异常 通信模式出错	常时	继续	SP46	R7756 (E316)

3-8-5 错误履历情报保存/显示功能

错误履历情报保存功能，是将 CPU 检出的自诊断系统错误信息自动保存的功能。

每次 NK0 发生自诊断出错时，发生的错误信息按：发生日期、时间及出错代码、出错信息的顺序记入 NK0 系统。

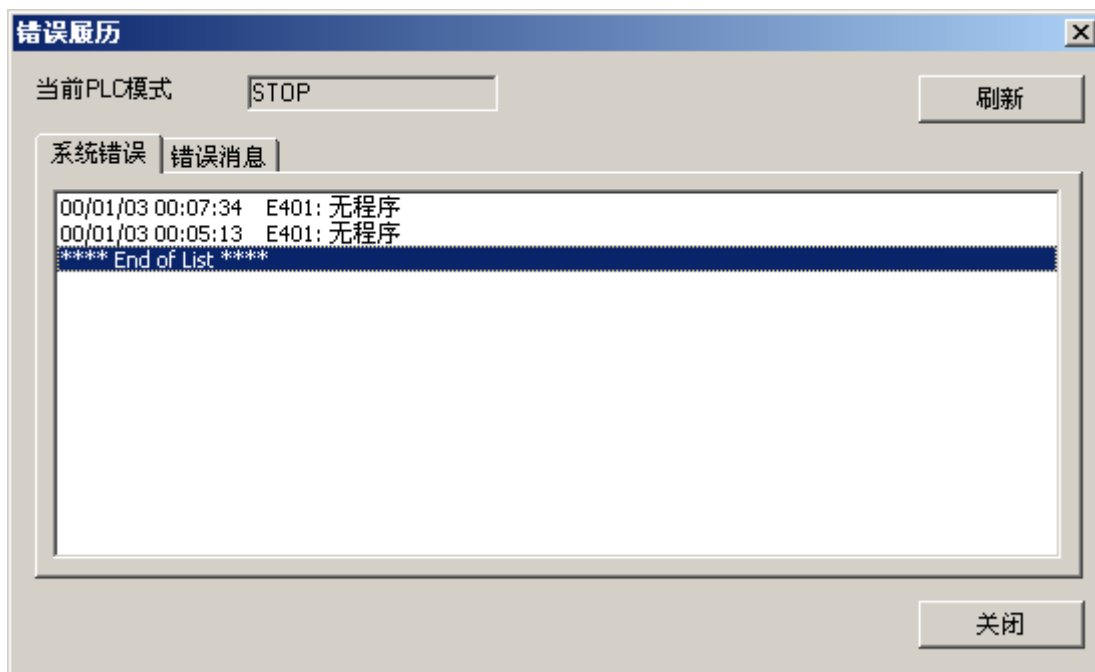
需要时，可以通过 KPPSoft 软件的[PLC->错误履历]菜单，打开 NK0 的错误履历窗口，选择[系统错误]页，查看系统错误履历信息，这对设备的维修、保养很有用。



NK0 最多可以记录 32 条系统出错信息，如已有 32 条记忆信息后再发生自诊断出错信息，就将最先的信息消去，新的信息依次记入。

出厂时，或进行系统参数初始化后，履历信息被清除。

下面列出 KPPSoft 中的 NK0 错误履历显示窗口的一个样例：



第四章 NK0 系列 PLC 通信功能

4-1 NK0 系列 PLC RS-485 通信端口规格及通信线连接

NK0 系列 PLC 带有 1 个 RS-485 串行通信口, 用于连接 KPPSoft 计算机编程软件, 编辑/下传 NK0 用户 PLC 程序, 配置 NK0 的工作参数, 改变 NK0 的运行模式, 监视 NK0 中用户程序的执行, 查看 NK0 系统状态等。

下面的资料中, 为了说明的方便, 我们有时用 Port0 口来指代 RS-485 通信端口。

NK0 系列 PLC 的 RS-485 通信口的规格如下:

- 3P 接线端子型端口。
- CCM2 (DirectNET) 协议 (S), K-协议 (S), MODBUS RTU (M/S), 无协议 (M/S), KPPSoft 编程软件专用通信协议。(通过通信参数设置, 或特殊寄存器设置。)
- RS-485 通信规格 (硬件性能符合 RS-485 标准)。
- 通信速度最高支持到 115200bps。
- 通信距离: 9600bps (包括) 以下, 最大 1Km; 14400bps (包括) 以上, 最大 500m。
- 节点数: 最多 32 个, 可组建 1 对多通信网络。
- 适用电线大小: 16-26 AWG。

PORT1 信号端子 说明		
1	A	差分信号+
2	B	差分信号-
3	SG	信号地

NK0 系列 PLC 的 RS-485 串行通信口既可以用于 2 个 RS-485 通信设备间的一对一连接, 也可以通过该 RS-485 口组建串行通信网络。

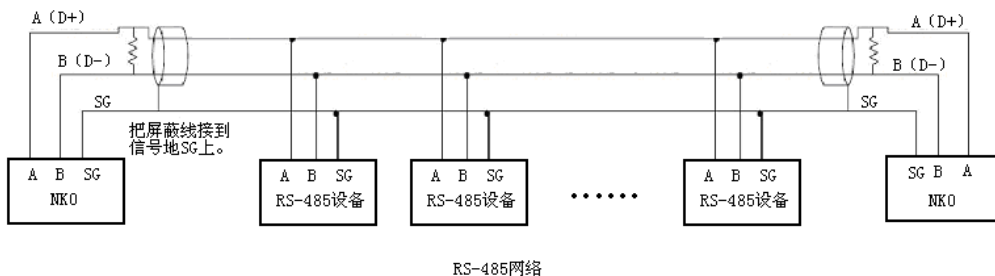
NK0 系列 PLC 通过 RS-485 口与其他 RS-485 设备一对一通信时电缆信号线连接分配如下:

NK0 RS-485 通信口		其他 RS-485 设备	备注
引脚	信号名	信号名	
1	A	A	有的 RS-485 口信号端用 D+ (+), D- (-) 标称, 一般, D+ 就是 A, D- 就是 B。
2	B	B	
3	SG	SG	

组建 RS-485 通信网络时, 只要把各 RS-485 设备的 A (D+), B (D-), SG 信号挂接到 RS-485 通信总线的各相应信号线上, 并在需要在 RS-485 网络 2 端的 A、B 之间接上必要的终端电阻即可(目的: 增加通信线路的抗干扰能力。)。注意终端电阻的阻值必须与传输线相匹配, 一般典型的终端电阻阻值为 120 欧姆。

使用 NK0 作为主局组建 RS-485 通信网络时, 最多支持挂接 32 个设备; NK0 也可以作为通信子局加入到 RS-485 通信网络中。

典型的 RS-485 通信网络连线示意图如下：



请使用带屏蔽的双绞电缆线制作 RS-485 通信电缆。布线时尽量与动力线等分开布，需要时可以单独穿铁管布线，并在需要时把电缆屏蔽线单端接地。好多设备的 RS-485 口不带 SG 信号端，在接线时，可以不接 NK0 的 SG 信号线。

4-2 NK0 系列 PLC RS-485 通信端口支持的通信协议及参数设置方法

4-2-1 NK0 系列 PLC RS-485 通信端口支持的通信协议

NK0 RS-485通信端口支持的通信协议如下表。

支持协议	RS-485 (Port0)	
	RUN	STOP
KPPSOFT 编程软件专用协议	△	○
K-协议	○	×
CCM2 (DirectNET) 协议	○	×
MODBUS RTU	○	×
无协议	○	×

○：支持连接 △：在下面说明 1 的固定通信参数下可以连接 ×：不支持

说明：1) 本通信口在 STOP 模式下，仅支持固定参数（1 号局/19200bps/1 位停止位/无校验）下的 KPPSoft 编程软件连接；在 RUN 模式下，按通信设置寄存器的设置参数进行动作，如果这时的通信参数设置成了（1 号局/19200bps/1 位停止位/无校验），则也可以通过编程软件专用协议来连接。

2) 通过编程软件专用协议可以读写用户程序、系统参数、功能存储器等所有 NK0 系列 PLC 的内容；通过其他协议只能读写功能存储器内容。

3) K-协议，CCM2 (DirectNET) 协议只支持子局功能。

4-2-2 NK0 系列 PLC RS-485 通信端口通信参数设置寄存器

NK0 给 RS-485 通信端口分配有一组共 10 个特殊的通信参数寄存器，用于存放 RS-485 通信口的通信动作参数。在使用 NK0 系列 PLC 上的 RS-485 通信口前，需要先对该通信端口的动作参数进行设置，以设置合适的通信参数，使得 NK0 可以按指定的参数通过该通信口与其他设备建立连接，交换信息。

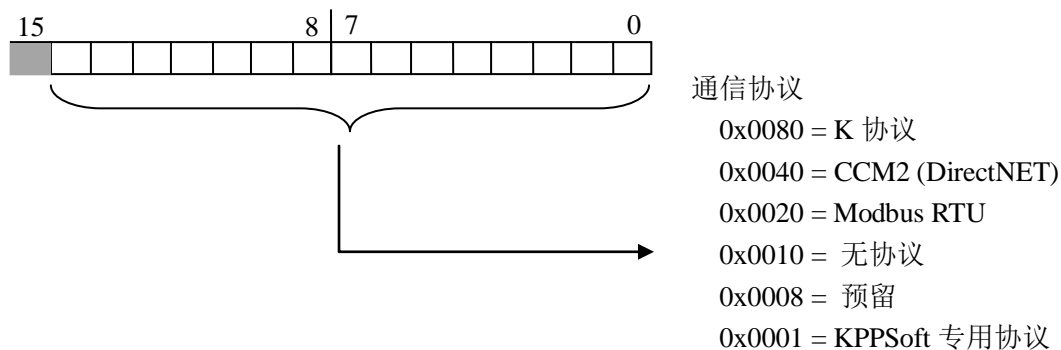
该组 RS-485 串口设置特殊寄存器及其内容如下表。

参数内容	特殊寄存器	寄存器出厂初始值
通信协议选择	R7650	0x21
有协议通信参数①/无协议参数①	R7651	00
有协议通信参数②/无协议参数②	R7652	0x0701
无协议参数③	R7653	0x70
无协议参数④	R7654	0
无协议参数⑤	R7655	0
无协议参数⑥	R7656	0
未使用, 预留	R7657	0
未使用, 预留	R7660	0
通信参数设置完成标志	R7661	0

下面详细说明 NK0 RS-485 通信参数寄存器内容。

1、通信协议选择寄存器（R7650）

选择通信口使用的通信协议。



NK0 系列 PLC RS-485 通信口不支持多协议自动判别功能，即你不可以在使用某个通信口时把该通信口设置成同时支持多个通信协议（但 MODBUS RTU 协议可以与 KPPSOFT 专用协议同时使用）。在实际使用时，当把某个通信口设置成 K 协议或 CCM2 协议或无协议通信方式时，在 PLC 处于 RUN 状态下时，该通信口将不能与 KPPSOFT 通信（虽然这时是默认选择了 KPPSOFT 专用协议）。但设置成使用 MODBUS RTU 协议与 KPPSOFT 专用协议时，在 RUN 方式下可以进行这 2 个协议的自动判别。当然，这种情况下各通信协议的局号/速度/停止位/校验位等参数都是相同的。NK0 在出厂时的缺省设置为：RS-485 通信口同时支持 MODBUS RTU 协议和 KPPSOFT 专用协议（通信协议寄存器的值为 0x21）。

注意：在 NK0 处于 TERM-STOP 停止模式下（与运行开关位置无关），RS-485 口按以下固定通信参数动作（目的：连接 KPPSoft 工具软件）：

局号 = 1；通信速度 = 19200bps；停止位 = 1 位；奇偶校验 = 无校验。

在 NK0 处于 TERM-RUN 运行模式下时，RS-485 口按其通信参数设置寄存器设置的参数动作。

当协议选择寄存器设置参数有错误时，系统自动把该协议选择寄存器内容恢复成出厂缺省值 0x21。

注意：在设置 NK0 RS-485 通信口的通信协议时，可能可以同时选中 K/CCM2/MODBUS RTU 通信协议中的几个协议，但请记住 NK0 RS-485 口只支持使用其中一个协议，几个协议的优先级如下：

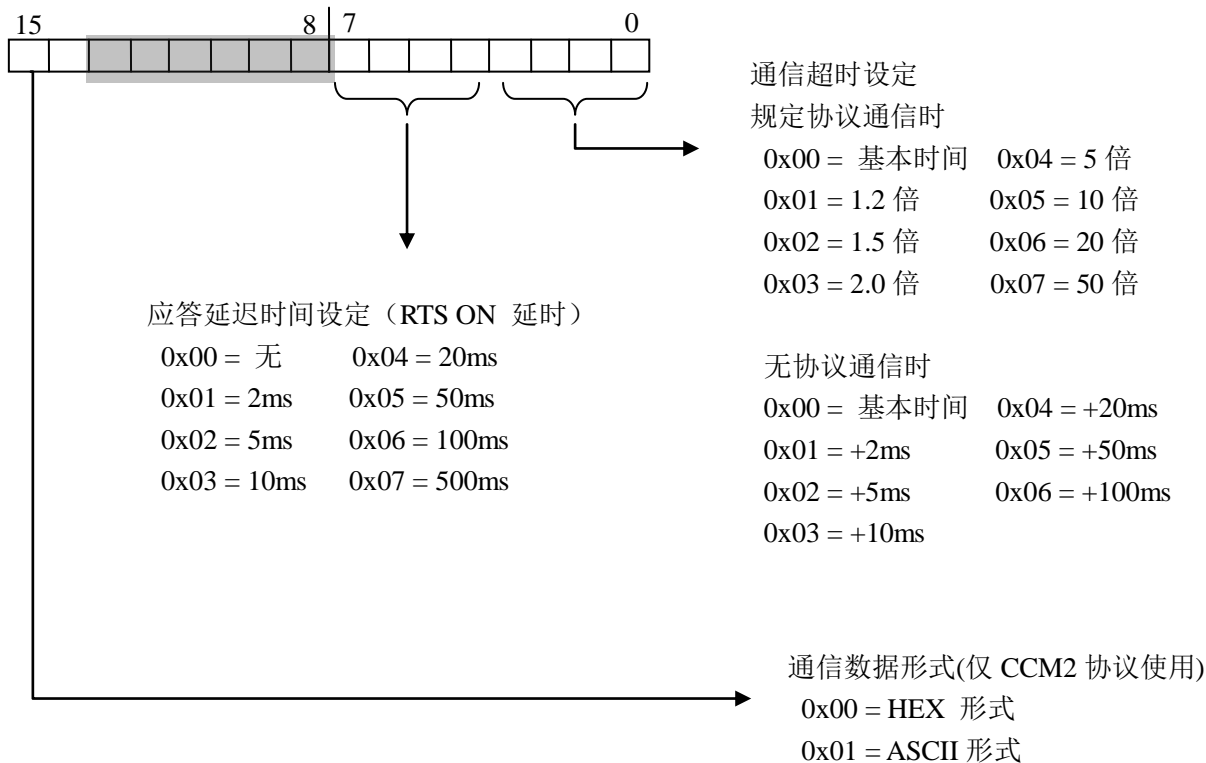
CCM2 协议：最高优先级协议，选择了这个协议后，其他协议都无效；

K 协议：第二高优先级协议，选择了这个协议后，在没有选择 CCM2 协议的情况下，使用该协议；

MODBUS RTU 协议：在没有选择 CCM2 协议和 K 协议的情况下，使用该协议。

2、有协议通信参数寄存器①/无协议参数寄存器①（R7651）

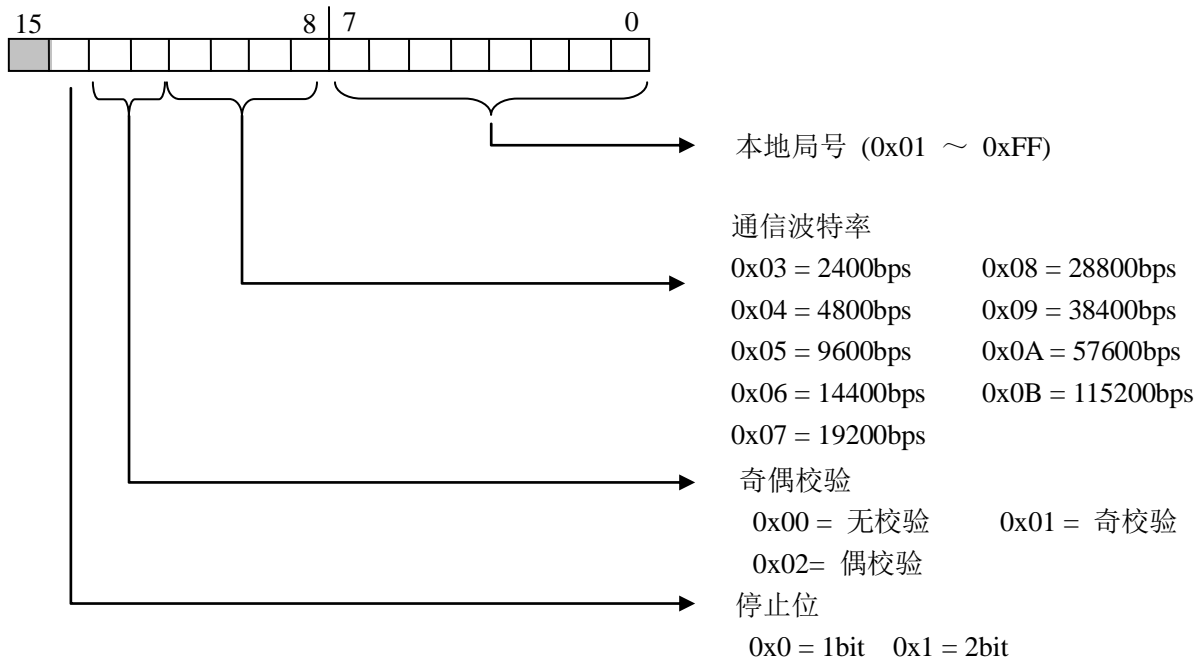
用于选择设置 RS-485 通信口的通信超时时间，RTS ON 应答延时时间，以及 CCM2 通信时的数据形式等通信参数。



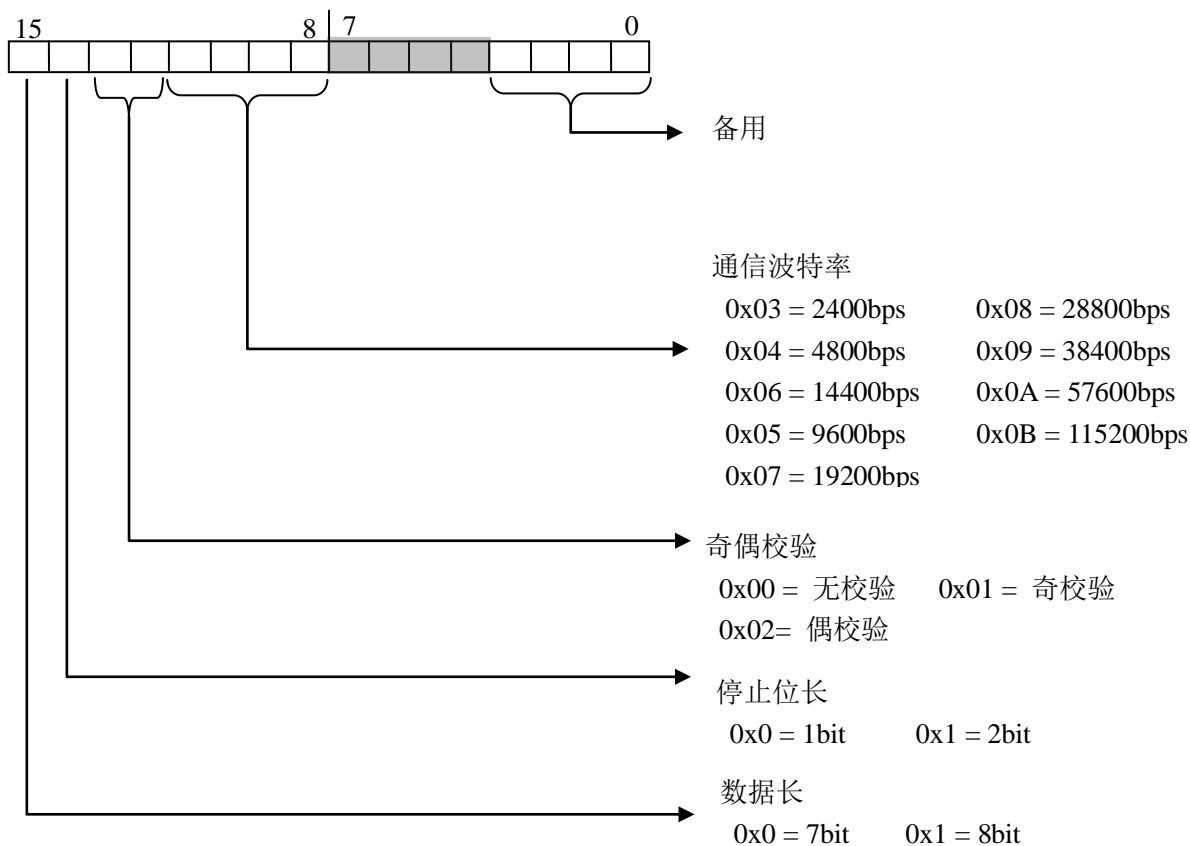
关于通信超时基本时间：
 在 K 协议/CCM2 协议通信时，为 800ms；
 在 MODBUS RTU 协议通信时，为 500ms；
 在无协议通信时，为 3 个字符通信时间。

3、有协议通信参数寄存器②/无协议参数寄存器②/ (R7652)

在 K 协议/CCM2 协议/MODBUS RTU 协议通信时，用于选择本地局号、通信速率、奇偶校验位、停止位等通信参数。注意：其中的局号部分，虽然其范围为 0x01 ~ 0xFF，但实际上对于 K 协议/CCM2 协议，其允许范围为 0x01 ~ 0x5A (01 ~ 90)；对于 MODBUS RTU 协议其允许范围为 0x01 ~ 0XF7 (01 ~ 247)。

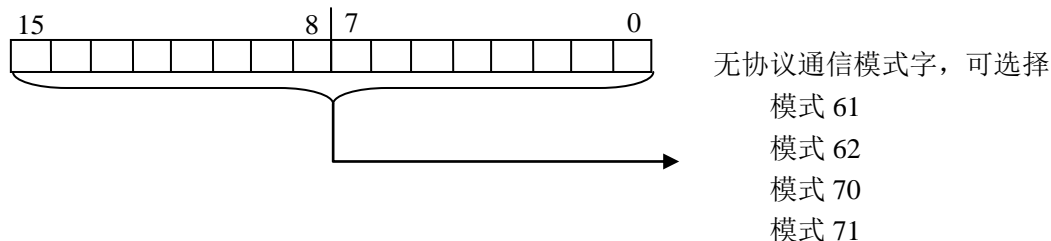


在无协议通信时，用于选择通信速率、奇偶校验位、停止位、数据位长等通信参数。



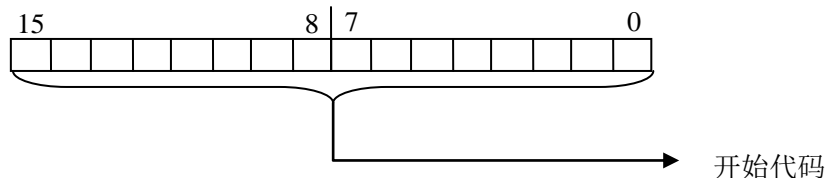
4、无协议参数寄存器③（R7653）

指定无协议通信的通信模式。NK0 系列 PLC 支持模式 61，模式 62，模式 70，模式 71 等 4 种无协议通信模式。把相应的数值送入该无协议通信模式寄存器中，以指定通信口使用的无协议通信模式。



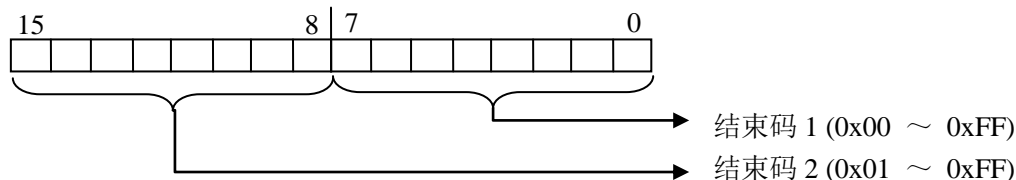
5、无协议参数寄存器④（R7654）

指定[模式 5E]无协议通信模式下数据接收开始代码。在该代码出现前的所有数据全部忽略。开始代码不能固定的场合，该数据设置成“00”。目前 NK0 不支持模式 5E 通信。该寄存器预留。



6、无协议参数寄存器⑤（R7655）

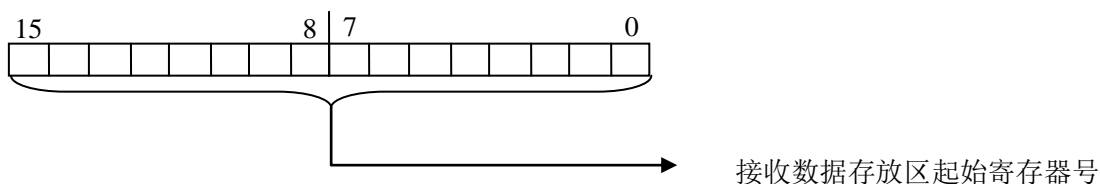
指定模式 70，模式 71 无协议通信时的数据接收通信结束码，在数据接收时接收到与结束码相同的数据，则表示 1 次串行接收通信完成。通信接收数据结束码为 2 字节结束码，当通信结束码寄存器高位字节为 0 时，表示使用 1 字节通信结束码。



7、无协议参数寄存器⑥（R7656）

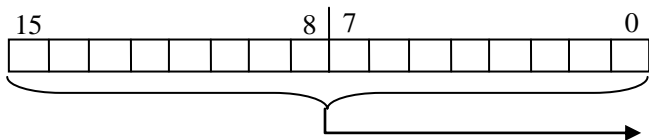
指定无协议通信时接收数据的存放地址，以二进制形式存入接收数据存放寄存器组（数据接收缓存区）的起始寄存器地址。原则上，可以将所有数据寄存器区域，作为首地址的设定范围。

例) 利用 P0 进行无协议通信，数据接收缓存区从 R2000 开始时，设置 R7656 = 0400 (H)
 8 进制数的 2000 即为 16 进制数的 400。 2000 (O) = 400 (H)



8、通信参数设置完成标志寄存器（R7661）

当把需要的通信参数寄存器设置完成后，在本通信参数设置完成标志寄存器中写入数值 0xA55A 后，NK0 即根据参数寄存器的内容来更新通信口设置，通信口更新完成后，相应通信口自动进入通信就绪状态。同时 NK0 系统自动在本通信参数设置完成标志寄存器中写入数值 0x5555。如果设置参数有错误，则系统把发现错误的参数寄存器号放入本寄存器中。



通信参数设置完成代码 0xA55A。
 参数设置正确，变成 0x5555。
 参数设置有错误，发现错误的寄存器号写入。

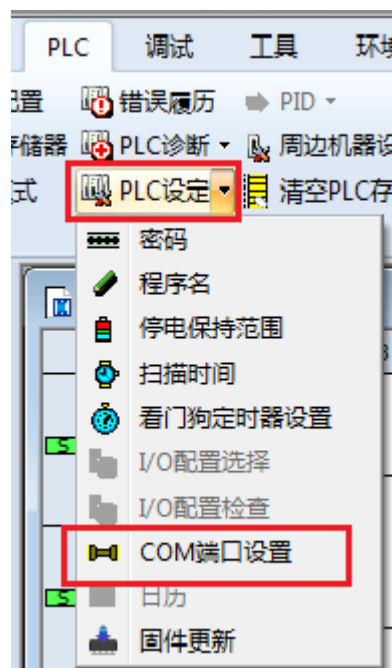
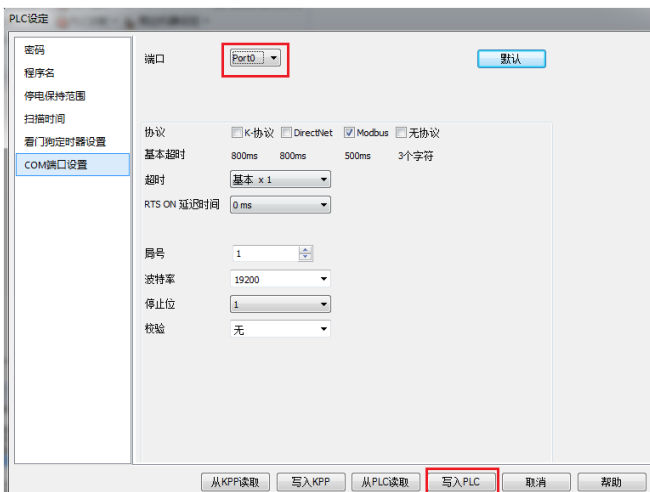
4-2-3 NK0 系列 PLC RS-485 通信端口通信参数设置方法

NK0 RS-485 通信端口通信参数的设置方法主要有以下 2 种：

方法一：

通过 KPPSoft 软件设置。KPPSoft 软件上有专门的 NK0 通信参数设置菜单项，可以用于设置 NK0 RS-485 通信口的动作参数，通过 KPPSoft 软件设置好通信口动作参数并写入 NK0 后，NK0 RS-485 通信口就能按设置好的通信参数进行动作。

通过 KPPSoft 软件设置 NK0 RS-485 通信端口通信参数的方法如下：打开 KPPSoft 软件，并与 NK0 建立好通信连接后，如右图所示通过[PLC->PLC 设定->COM 端口设置]菜单,打开 PLC 通信端口设置窗口，选择需要设置参数的通信端口，设置好相应的通信参数，按[写入 PLC]键，待数据写入 PLC 后，就完成了某个通信口通信参数的设置，示意图如下：



注意：在用 KPPSoft 软件设置 NK0 通信口通信参数时，当选择 MODBUS RTU 协议时，KPPSoft 软件专用协议总是隐含被选中的，即通信口被设置成同时支持 2 个通信协议（MODBUS RTU+KPPSoft 软件专用协议）。在实际使用时，NK0 会根据连接对象，自动选择合适的通信协议。但在选择其他通信协议后，该通信口将不支持 KPPSoft 软件专用协议，即再不能连接 KPPSoft 软件。（STOP 模式下支持 KPPSoft 软件专用协议。）

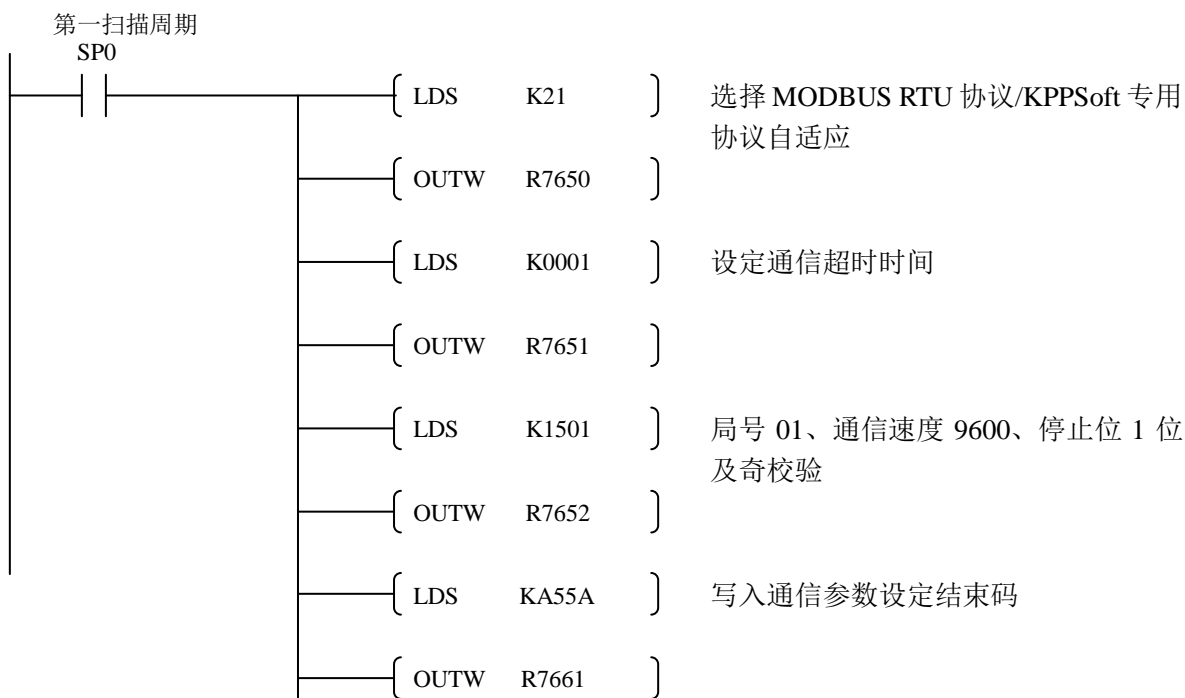
方法二：

可以通过设置特殊通信寄存器的方法来设置 RS-485 通信口的动作参数。

NK0 给 RS-485 通信端口分配有一组特殊的通信参数寄存器，你可以通过置值这些特殊寄存器的方法来设置该通信口的动作参数，NK0 RS-485 通信端口分配的特殊寄存器见上面 4-2-2 节的介绍内容。（实际上方法一也是设置了这些特殊寄存器）

通信参数的特殊寄存器设置方法一般是通过在用户程序中编写一段初始设置程序来实现的。下面给出一段 Port0 口的设置程序例，用于设置 Port0 口的通信参数。

设置通信口 : Port0 (RS-485)
 通信协议 : MODBUS RTU/KPPSoft 专用协议 自适应
 通信超时 : 规定时间 1.2 倍
 局号 : 01
 通信速度 : 9600bps
 停止位 : 1
 奇偶校验 : 奇校验



在用户程序中加上以上初始程序段，执行该用户程序后，就能按规定的通信参数初始化通信口 Port0。

注意：NK0 仅支持 MODBUS RTU 通信协议和 KPPSoft 软件专用协议的多协议自动判别功能，所以运行上述程序后，NK0 可以和 KPP 建立通信连接（其它通信参数：1 号局/19200bps/1 位停止位/无校验）。但 NK0 不支持 K/CCM2/无协议通信等其他协议和 KPPSoft 软件专用协议的自动判别功能，所以运行有关的通信设定程序后，将不能和 KPP 建立通信连接。即 K 协议/CCM2 协议/无协议通信不支持多协议自适应功能，这些协议通信只能单独设定。

但无论如何，RS-485 通信口在 STOP 模式下，总是可以与 KPPSoft 建立通信连接的！

4-2-4 NK0 系列 PLC RS-485 通信端口通信程序编制

NK0 系列 PLC 的 RS-485 端口一般有作为子局使用和作为主局使用 2 种用法。作为子局使用时，只要设置好相关的通信参数寄存器就可以进行数据接收/发送工作了，而并不需要编制特别的通信程序。而作为通信主局工作时，除了需要设置通信口通信参数寄存器外，还需要编制特别的通信程序段，用于说明 NK0 的 RS-485 通信口、在何种条件下、与网络中哪个通信子局、进行何种数据通信（接收或发送）、通信数据的存放位置、以及通信数据量等等。

用于 NK0 主局通信的指令一般为 RX/WX（MRX/MWX 指令为 MODBUS 通信专用指令，后述）。RX 为数据接收通信指令，即把指定子局上的某段数据接收到本机，并存放到的位置；WX 为数据发送通信指令，即把主局中的某段数据发送到指定子局的特定位置。在使用 NK0 串口的 MODBUS RTU 协议主局通信功能时，需要用到 RX/WX 通信指令；在使用 NK0 串口的无协议通信数据发送通信功能时，要用到 WX 指令。

通常的 NK0 通信主局通信程序段如下：



上面的例子程序为通过 RS-485 口进行数据发送的例子，只要把 WX 指令改成 RX，就成为数据接收程序。

程序段中有关的内容说明如下。

1) 通信口通信中标志 SP110

NK0 对于 RS-485 通信口，分配有一组特殊线圈 SP，用于表示通信口的状态。在编制通信程序时，你可以有效利用这些特殊线圈，来编制合理有效的通信程序。

特殊线圈	功能
SP54	RS-485 通信参数寄存器设置内容有错时为 ON 用)。
SP110	Port0 口通信指令执行中 ON
SP111	Port0 口执行通信指令时，出错时 ON

2) 使用通信端口以及对方子局号指定

LDS KF002

该条指令用于指定使用 NK0 的 RS-485 通信口和指定的通信子局进行通信。我们把这条指令改写成一般格式如下：

LDS KABCD

其中，AB 表示使用的通信端口标志代码，在 NK0 上规定如下：

AB = F0 表示使用 RS-485 通信口；

CD 表示通信对方子局号。

对于 NK0 的 RS-485 串行通信口，为 16 进制数表示的子局号，例如 K02 表示 2 号子局；K10 表示 16 号子局，等等。

注意：MODBUS RTU 协议下的子局号范围为 1~247，所以 CD 的范围为 0x01~0xF7。

K 协议/CCM2 协议下的子局号范围 1~90，所以 CD 的范围为 0x01~0x5A。

3) 一次通信传送字节数指定

LDS K20

指定通信字节数，为 BCD 数，最大为 128 字节。注意：NK0 的一个寄存器为 2 字节。

4) 通信数据存放位置指定

LDR O3000

指定通信数据的本地存放寄存器位置（起始寄存器号）。

WX R2000 （数据接收为 RX，数据发送为 WX）

指定对方局（子局）的通信数据存放寄存器位置（起始寄存器号）。在无协议通信时无特定意义。

上面介绍了 NK0 RS-485 通信口的通信参数寄存器及其参数设置方法，下面分协议介绍如何在 NK0 系列 PLC 上实现各种协议下 RS-485 通信的方法。

4-3 NK0 系列 PLC K-协议/CCM2（DirectNET）协议通信的实现

NK0 的 RS-485 通信口支持 K-协议和 CCM2（DirectNET）协议的子局通信功能。支持 K-协议和 CCM2（DirectNET）协议主局功能的设备利用这 2 个协议，可以通过其自身的串行通信口，来与 NK0 交换信息。这样的设备包括：上位计算机软件、工业触摸屏、文本显示器、PLC 等等，其支持的主要功能为：NK0 位状态的读写；NK0 数据寄存器内容的读写等等。

有关 CCM2（DirectNET）协议的详情请参考《CCM2 通信协议技术资料》。

要使用 NK0 RS-485 通信口的 K-协议和 CCM2（DirectNET）协议的子局通信功能，首先需要设置通信口的通信参数，NK0 RS-485 通信参数存放在一组特殊寄存器中，NK0 RS-485 口对应 K-协议和 CCM2（DirectNET）协议的子局通信功能的特殊参数寄存器如下：

参数内容	特殊参数寄存器
通信协议选择	R7650
通信超时、RTS ON 延时时间、CCM2 通信数据形式	R7651
本地局号、通信速度、奇偶校验位、停止位	R7652
通信参数设置完成标志	R7661

有关 K-协议/CCM2 协议通信参数特殊寄存器内容的详细说明请参见 4-2-2 节介绍。

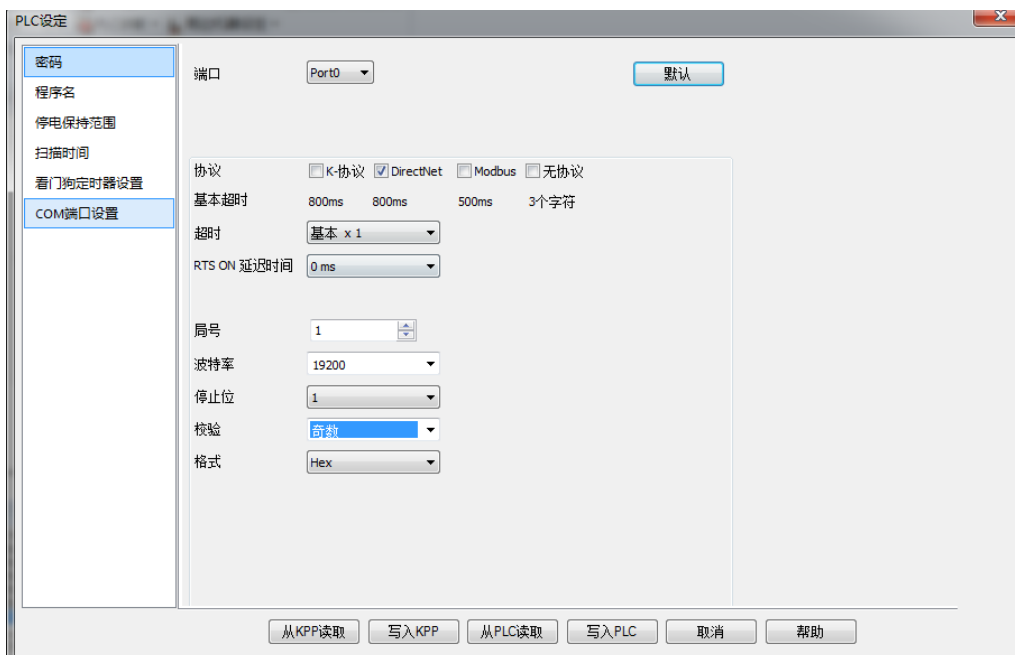
由于 NK0 的 RS-485 通信口只支持 K-协议和 CCM2（DirectNET）协议的子局通信功能，所以在使用该串行通信口的 K-协议和 CCM2（DirectNET）协议的子局通信功能时，只要设置好通信口的通信参数后就可以使用了，并不需要编制特别的通信程序。

NK0 的 RS-485 口通信参数设置方法有 KPPSoft 软件设置和用户 PLC 程序设置 2 种，下面给出 RS-485 口通信参数的设置方法例子，例子中 RS-485 口的通信参数设置如下。

通信协议 : CCM2（DirectNet）协议
 通信超时 : 规定时间
 RTS ON 延时 : 0ms
 局号 : 01
 数据形式 : CCM2 通信时采用 HEX 数据形式
 通信速度 : 19200bps
 停止位 : 1
 奇偶校验 : 奇校验

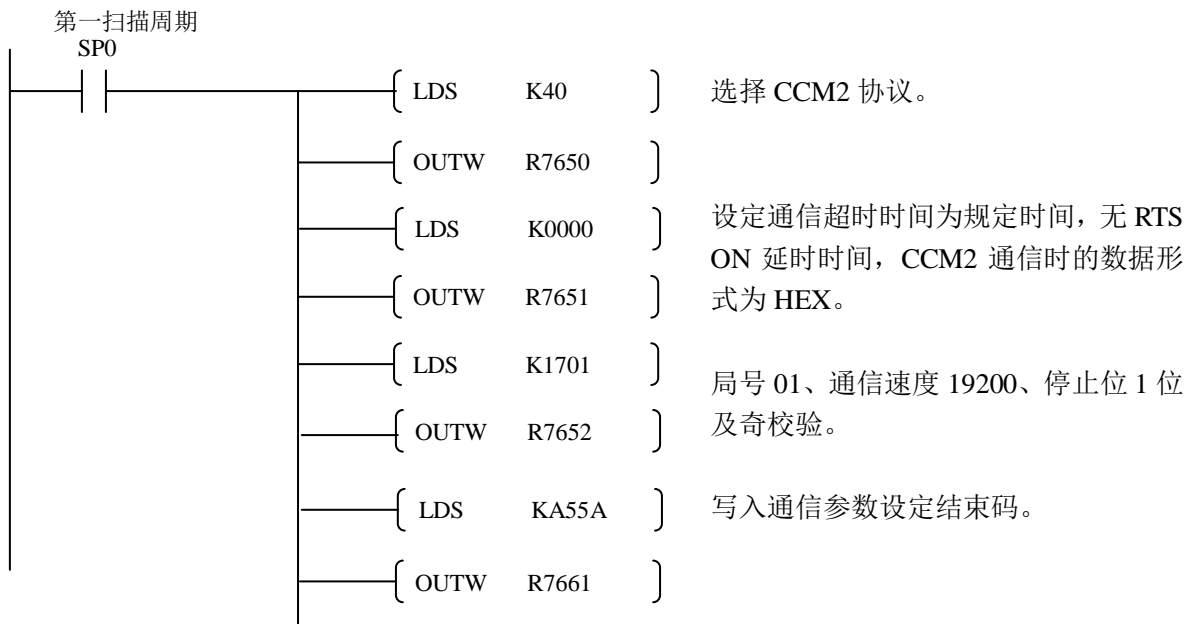
设置方法一：通过 KPPSoft 软件设置

通过[PLC->PLC 设定->COM 端口设置]菜单,打开 NK0 的 RS-485 通信端口设置窗口，设置其通信参数如下页图所示。



设置方法二：通过用户程序设置

其设置程序段如下：



在用户程序中加上以上初始程序段，执行该用户程序后，就能按规定的通信参数初始化 RS-485 通信口了。

4-4 NK0 系列 PLC MODBUS RTU 协议通信的实现

NK0 系列 PLC 的 RS-485 通信口支持 MODBUS RTU 协议的主局和子局通信功能。使用此协议 NK0 系列 PLC 可与公司支持 MODBUS RTU 通信协议的 PLC，工业触摸屏 GC-A2、EA7E、CMORE 系列，以及支持 MODBUS RTU 协议的第三方设备通信。

使用 NK0 系列 PLC 的 MODBUS RTU 通信功能，可实现以下数据的读、写。

- 定时器经过值，计数器计数值的读出
- 数据寄存器的读出，写入
- I/O 内部继电器，级等功能存储器（位）的读出，写入

4-4-1 MODBUS RTU 协议通信参数设置

要使用 NK0 系列 PLC RS-485 通信口的 MODBUS RTU 协议通信功能，首先需要设置该通信口的通信参数，NK0 RS-485 通信口通信参数存放在一组特殊参数寄存器中如下：

参数内容	特殊参数寄存器
通信协议选择	R7650
通信超时、RTS ON 延时时间	R7651
本地局号、通信速度、奇偶校验位、停止位	R7652
通信参数设置完成标志	R7661

NK0 RS-485 通信口的通信参数设置方法有 KPPSoft 软件设置和用户 PLC 程序设置 2 种，下面给出该通信口通信参数的设置方法例子，例子中 RS-485 口的通信参数设置如下。

通信协议 : MODBUS RTU 协议
 通信超时 : 基本时间
 RTS ON 延时 : 0ms
 局号 : 01
 通信速度 : 19200bps
 停止位 : 1
 奇偶校验 : 无校验

设置方法一：通过 KPPSoft 软件设置

通过[PLC->PLC 设定->COM 端口设置]菜单,打开 NK0 的 RS-485 通信端口设置窗口，设置其通信参数如下页图所示。

4-4-2 MODBUS RTU 协议通信子局功能实现

使用 NK0 RS-485 通信口的 MODBUS RTU 协议子局通信功能不需要编制通信程序，而只要设置好通信口的通信参数就可以了。在进行 MODBUS RTU 通信时，MODBUS 通信主局必须采用 MODBUS RTU 协议与 NK0 子局进行通信。此时，通信主局必须发出 MODBUS 通信功能码和对应于 NK0 系列 PLC 功能存储器定义号的 MODBUS 地址码。

下面介绍 NK0 系列 PLC 支持的 MODBUS 通信功能码和 NK0 系列 PLC 功能存储器定义号与 MODBUS 地址码的变换方法。

1. MODBUS 通信功能码

MODBUS 功能码用于说明一次通信是读操作还是写操作，以及读写的内容是单个数据点还是一组数据。

NK0 系列 PLC 支持的 MODBUS 功能码如下表所示：

MODBUS 功能码	通信功能	可使用的 NK0 数据类型
01	读一组线圈	Q, M, T, C
02	读一组输入信号	I, SP
05	置位/复位单个线圈（仅子局）	Q, M, T, C
15	置位/复位一组线圈	Q, M, T, C
03, 04	读单个或多个数据寄存器	R
06	改写单个数据寄存器（仅子局）	R
16	改写一组数据寄存器	R

2. MODBUS 地址的确定

在很多的 MODBUS 主局软件中约定有 2 种方法可用于确定 MODBUS 通信时 PLC 功能存储器定义号对应的 MODBUS 通信数据地址：

- ◆ 通过指定 MODBUS 数据类型和地址
- ◆ 通过仅指定 MODBUS 地址

下面我们分别说明如何利用该 2 种方法来指定对应于 NK0 系列 PLC 功能存储器地址的 MODBUS 通信数据地址。

(1)、通过指定 MODBUS 数据类型和地址

有很多的 MODBUS 主局软件允许你通过指定 MODBUS 数据类型和地址的方式来确定 NK0 系列 PLC 的功能地址，这是最容易的方法，但不是所有的软件都允许你用该种方式。

正确的计算地址的算法取决于你所选择的 PLC 数据的类型。在此种方式下，PLC 数据分成 2 大类：

- 离散型——GI, I, SP, GQ, Q, M, S, T, C
- 连续字——R, 定时器/计数器经过值

你首先要将 8 进制地址值转换为 10 进制数，然后加上适当的 MODBUS 地址偏移量（如果需要）。下表给出了对应于 NK0 系列 PLC 所有功能存储器的 MODBUS 地址码。

功能存储器	点数/字数	PLC 地址范围 (8 进数)	MODBUS 地址 范围(10 进数)	MODBUS 数据类型
对于离散型数据: 10 进数表示的 PLC 地址 + 开始地址 + 数据类型				
通信输入线圈(GI)	256 点	GI0~GI377	0~255	输入
输入线圈(I)	256 点	I0~I377	2048~2303	输入
特殊线圈(SP)	512 点	SP0~SP777	3072~3583	输入
通信输出线圈(Q)	256 点	GQ0~GQ377	0~255	线圈
输出线圈(Q)	256 点	Q0~Q377	2048~2303	线圈
内部线圈(M)	1024 点	M0~M1777	3072~4095	线圈
级(S)	256 点	S0~S377	5120~5375	线圈
定时器(T)	128 点	T0~T177	6144~6271	线圈
计数器(C)	128 点	C0~C177	6656~6783	线圈
对于连续型数据: 10 进数表示的 PLC 地址 + 数据类型				
定时器经过值(R)	128 字	R0~R177	0~127	输入寄存器
计数器经过值(R)	128 字	R1000~R1177	512~639	输入寄存器
数据寄存器(R)	2816 字	R2000~R7377	1024~3839	保持寄存器
特殊寄存器(R)	256 字	R7400~R7777	3840~4095	保持寄存器

注意：为了整个 S 系列 PLC 所对应 MODBUS 地址的一致性，NK0 支持的 MODBUS 地址不是连续的。

下面给出几个该种方式下，如何取得 PLC 定义号对应的 MODBUS 地址的例子。

1) R2100

- A) 在表中得到 R2100 对应的表项
- B) 把 R2100 (0) 转换为 10 进数 = 1088 (D)
- C) 加入表中对应的 MODBUS 数据类型

则，R2100 对应的 MODBUS 地址如下：

$$1088 + \text{Hold. Reg} = \text{Holding Reg. } 1088$$

2) Q10

- A) 在表中得到 Q10 对应的表项
- B) 把 Q10 (0) 转换为 10 进数 =8 (D)
- C) 加入表中对应的开始地址(2048)
- D) 加入表中对应的 MODBUS 数据类型

则，Q10 对应的 MODBUS 地址如下：

$$8 + 2048 + \text{Coil} = \text{Coil. } 2056$$

3) T10 经过值

- A) 在表中得到 T10 对应的表项
- B) 把 T10 (0) 转换为 10 进数 = 8 (D)
- C) 加入表中对应的 MODBUS 数据类型

则，T10 对应的 MODBUS 地址如下：

$$8 + \text{Input. Reg} = \text{Input Reg. } 8$$

4) M54

- A) 在表中得到 M54 对应的表项
- B) 把 M54 (0) 转换为 10 进数 =44 (D)
- C) 加入表中对应的开始地址 (3072)
- D) 加入表中对应的 MODBUS 数据类型

则, M54 对应的 MODBUS 地址如下:

$$44 + 3072 + \text{Coil} = \text{Coil}.3116$$

(2)、仅通过指定 MODBUS 数据地址

有些 MODBUS 主局仅通过 MODBUS 数据地址来确定对应的 PLC 地址, 这种方式下的地址指定方式有些不同, 但这仍然是很简单的。基本的, 该种方式下, 仍然按地址范围划分数据类型, 这意味着单凭地址就可精确描述数据类型和位置, 一般的方法是给地址增加一个偏移量来实现。有一点非常重要, 在这种方式下, 你的上位主局软件可有 2 种编址方式:

- 484 方式
- 584/984 方式

我们强力推荐使用 584/984 编址方式。因该种方式下, 能存取地址空间较大。当你的主局软件仅支持 484 方式时, 有些 PLC 地址有可能存取不到。正确的计算地址的算式取决于你所选择的 PLC 数据的类型。在此种方式下, PLC 数据分成 2 大类:

离散型——I, SP, Q, M, S, T, C

连续字——R, 定时器/计数器经过值

同样, 你首先要把 NK0 系列 PLC 的 8 进制地址值转换为 10 进制数, 然后加上适当的 MODBUS 地址偏移量 (如果需要)。下表给出了对应于 NK0 系列 PLC 的所有功能存储器的 MODBUS 地址码。

功能存储器	数量	NK1 定义号范围 (8 进制)	MODBUS 地址范围 (10 进制)	484 方 式地址	584/984 方式地址	MODBUS 数据类型
对于离散型数据: 10 进数表示的 PLC 地址 + 开始地址 + 方式地址						
通信输入 (GI)	256	GI0~GI377	0~255	1001	10001	输入
输入 (I)	256	I0~I377	2048~2303	1001	10001	输入
特殊线圈 (SP)	512	SP0~SP777	3072~3583	1001	10001	输入
通信输出 (GQ)	256	GQ0~GQ377	0~255	1	1	线圈
输出 (Q)	255	Q0~Q377	2048~2303	1	1	线圈
内部线圈 (M)	1024	M0~M1777	3072~4095	1	1	线圈
定时器 (T)	128	T0~T177	6144~6271	1	1	线圈
计数器 (C)	128	C0~C177	6656~6783	1	1	线圈
级 (S)	256	S0~S377	5120~5375	1	1	线圈
对于连续型数据: 10 进数表示的 PLC 地址 + 方式地址						
定时器经过值 (R)	128	R0~R177	0~127	3001 4001	30001 40001	输入寄存器 保持寄存器
计数器经过值 (R)	128	R1000~R177	512~639	3001 4001	30001 40001	输入寄存器 保持寄存器
数据寄存器 (R)	2816	R2000~R7377	1024~3839	4001	40001	保持寄存器
特殊寄存器 (R)	256	R7400~R7777	3840~4095	4001	40001	保持寄存器

注意: 为了整个 S 系列 PLC 所对应 MODBUS 地址的一致性, NK0 支持的 MODBUS 地址不是连续的。

下面给出加上了开始地址的 MODBUS 地址参考表，供参考。

离散型数据列表如下：

功能存储器类型	PLC 地址范围 (8 进数)	484 方式地址	584/984 方式地址	数据类型
通信输入 (GI)	GI0 ~ GI377	1001 ~ 1256	10001 ~ 10256	输入
输入 (I)	I0 ~ I377	---	12049 ~ 12304	输入
特殊线圈 (SP)	SP0 ~ SP777	---	13073 ~ 13584	输入
通信输出 (GQ)	GQ0 ~ GQ377	1 ~ 256	1 ~ 256	输出
输出 (Q)	Q0 ~ Q377	---	2049 ~ 2304	输出
中间继电器 (M)	M0 ~ M1777	---	3073 ~ 4096	输出
级 (S)	S0 ~ S377	---	5121 ~ 5376	输出
定时器 (T)	T0 ~ T177	---	6145 ~ 6272	输出
计数器 (C)	C0 ~ C177	---	6657 ~ 6784	输出

连续型数据列表如下：

寄存器 R	PLC 地址范围 (8 进制)	输入/保持 (484 方式地址)	输入/保持 (584/984 方式地址)
定时器经过值	R0~R177	3001/4001~3127/4127	30001/40001~30127/40127
计数器经过值	R1000~R1177	3513/4513~3640/4640	30513/40513~30640/40640
用户数据 寄存器	R2000~R7377	不支持	41025~43840
特殊参数 寄存器	R7400~R7777	不支持	43841~44096

注意：为了整个 S 系列 PLC 所对应 MODBUS 地址的一致性，NK0 支持的 MODBUS 地址不是连续的。

下面给出该种方式下，如何取得 PLC 地址对应的 MODBUS 地址的几个例子。

1) R2100 (584/984 方式)

- A) 表中得到 R2100 对应的表项
- B) 把 R2100 (0) 转换为 10 进数 = 1088 (D)
- C) 加入该方式下对应的 MODBUS 方式地址 (40001)

则，R2100 对应的 MODBUS 地址如下：

$$1088 + 40001 = 41089$$

2) Q10 (584/984 方式)

- A) 在表中得到 Q10 对应的表项
- B) 把 Q10 (0) 转换为 10 进数 = 8 (D)
- C) 加入该方式下对应的开始地址 (2048)
- D) 加入对应的 MODBUS 方式地址 (1)

则，Q10 对应的 MODBUS 地址如下：

$$8 + 2048 + 1 = 2057$$

3) T10 经过值 (484 方式)

- A) 在表中得到 T10 对应的表项
- B) 把 T10 (0) 转换为 10 进数 = 8 (D)
- C) 加入对应的 MODBUS 方式地址 (3001)

则，T10 对应的 MODBUS 地址如下：

$$8 + 3001 = 3009$$

4) M54 (584/984 方式)

- A) 在表中得到 M54 对应的表项
- B) 把 M54 (0) 转换为 10 进数 = 44 (D)
- C) 加入对应的开始地址 (3072)
- D) 加入对应的 MODBUS 方式地址 (1)

则，M54 对应的 MODBUS 地址如下：

$$44 + 3072 + 1 = 3117$$

4-4-3 MODBUS RTU 协议主局通信通信程序编制

要使用 MODBUS RTU 协议的主局通信功能，除了需要设置通信寄存器参数外，还需要另外编制主局通信程序。NK0 MODBUS RTU 主局通信程序主要通过 RX/WX、MRX/MWX 指令来实现 PLC 功能存储器的读写，RX/MRX 指令实现从子局读出数据的功能；WX/MWX 实现向子局写数据的功能。在执行 RX/WX 指令前，先要在累加器和数据堆栈中设置好通信所需的相关参数。

MODBUS RTU 协议下，RX/WX、MRX/MWX 指令主要实现寄存器数据的读写，RX 指令支持保持寄存器的读出（功能码 3），WX 指令支持成组寄存器值的改写（功能码 16）。

下面给出 NK0 作为主局用 RX/WX 指令与 SN 系列 PLC 通过 RS-485 口进行 MODBUS RTU 通信的例子程序。MRX/MWX 指令下的通信，参见下节介绍。

注意：光洋 PLC 间进行 MODBUS RTU 通信时，不需要进行地址的转换工作。

例子程序通信参数如下：

NK0（主局）

- 通信超时时间：基本时间
- RTS ON 延时：无
- 局号：01
- 传送速度：19200bps
- 停止位：1bit
- 奇偶校验：ODD（奇数）

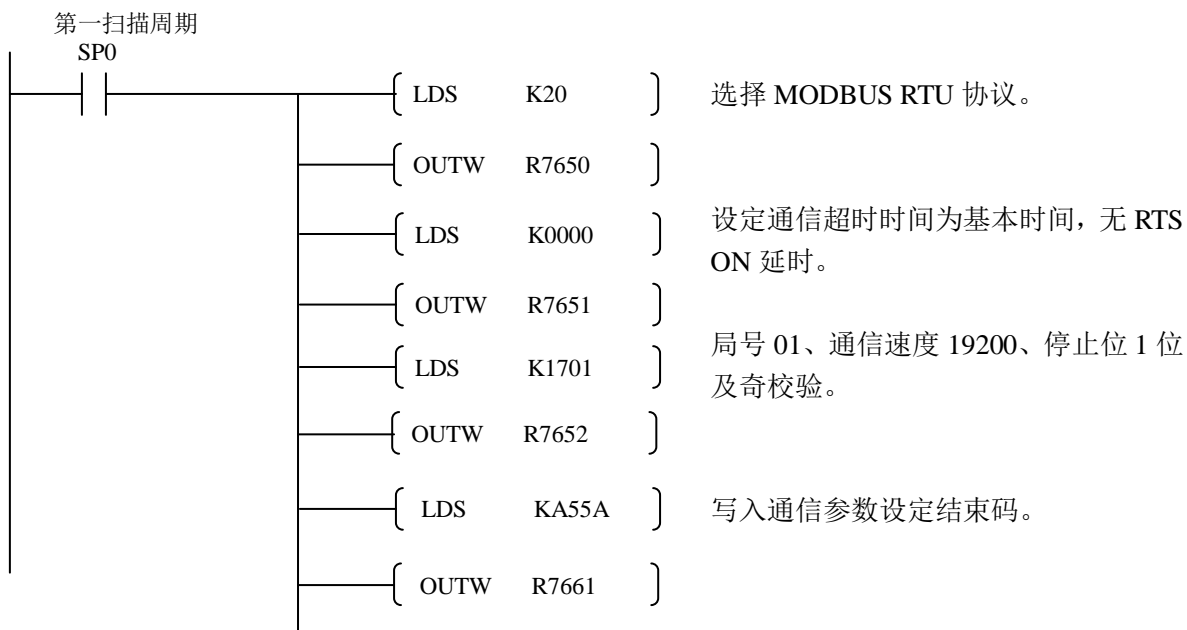
SN（子局）

- 子局号：02
- 通信超时时间：规定时间
- 应答延时时间：0ms
- 传送速度：19200bps
- 停止位：1bit
- 奇偶校验：ODD（奇数）

注意点：

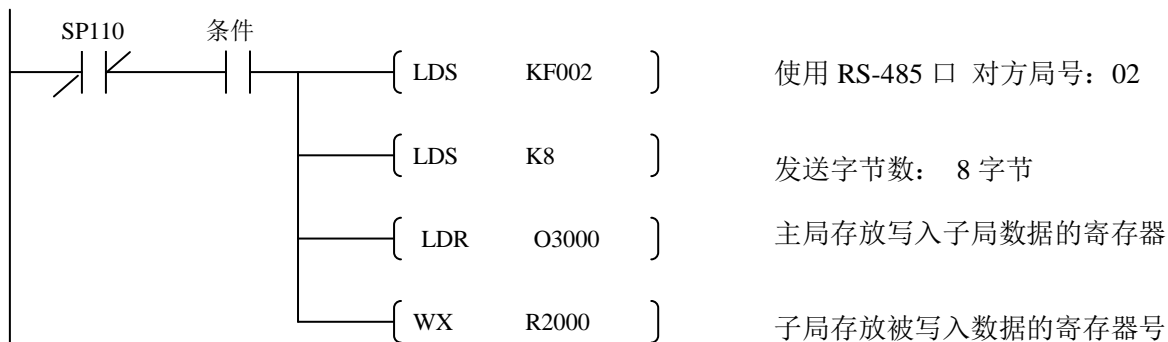
- 1) 主局、子局的通信速度、停止位、奇偶校验位等通信参数需要一致；
- 2) 通信子局号的设置要与通信主局 RX/WX 指令中的子局号一致；
- 3) 通信主局的局号可以是任意合法数，但不能与通信指令中的子局号一样。

A) NK0 主局通信参数设置程序段如下：

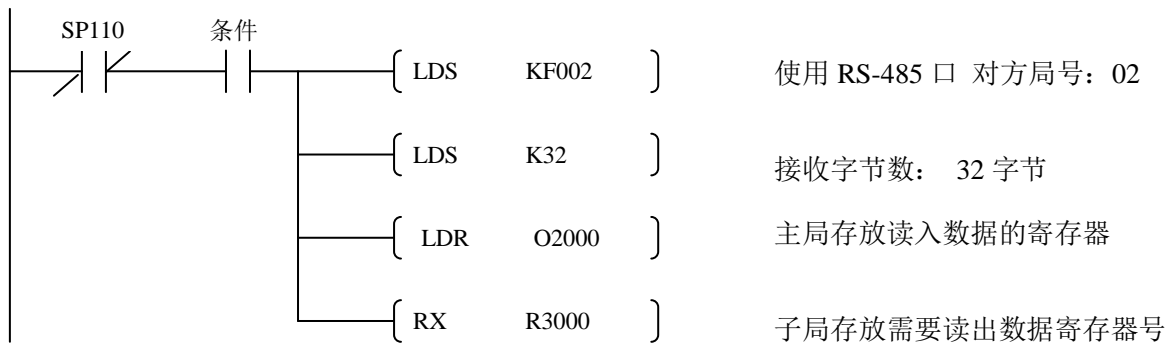


B) NK0 主局通信程序段例子

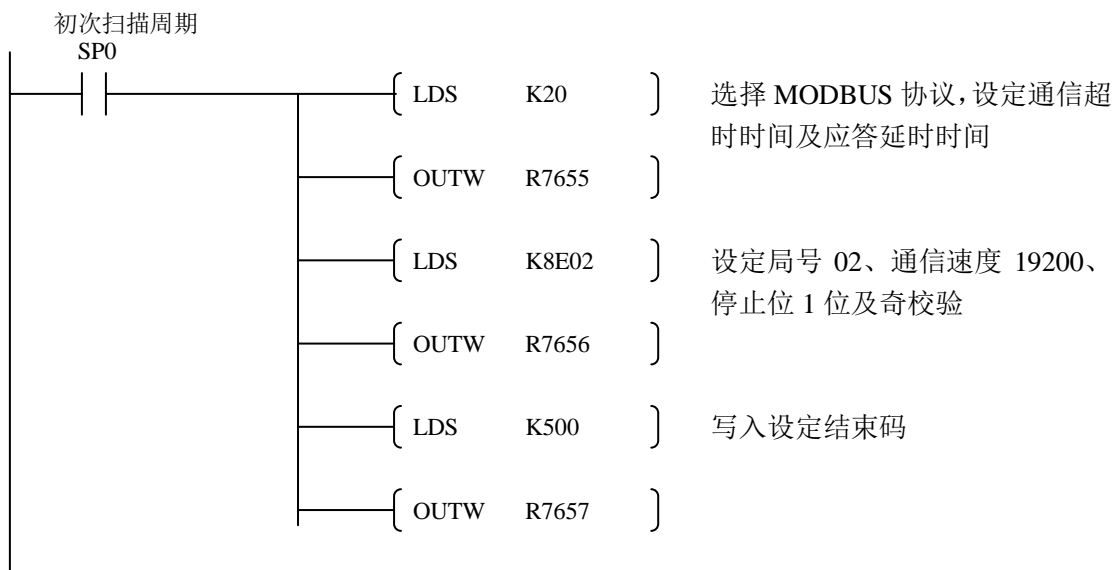
① 写出 把主局从 R3000 开始的 4 个字写出到 02 子局的 R2000 开始的 4 个寄存器中去。



② 读入 把 02 子局的 R3000 到 R3017 中的 16 字数据读入主局的 R2000 到 R2017 中



C) SN 子局通信参数设置程序如下:



注：在 SN 上仅需要编制通信口参数初始化程序，无需编制其他通信程序。

4-4-4 MODBUS RTU 协议专用通信指令 MRX/MWX

NK0 系列 PLC 除了提供标准的 RX/WX 通信指令来实现 MODBUS RTU 通信外，另外提供了一组 MODBUS RTU 协议专用通信指令 MRX/MWX，以便更灵活的实现 MODBUS RTU 通信。

当然在启动执行 MRX/MWX 指令前，先要设置好某个通信端口的通信参数。这可以通过 KPPSoft 软件来设置，或通过用户在用户程序中加入初始设置程序段来实现。

注意：使用 MRX/MWX 通信指令时，上面介绍的特殊线圈 SP110~SP111 将不起作用。

下面分别介绍 MRX 指令与 MWX 指令的用法。

1、MODBUS 读指令 MRX

MRX 指令用于 NK0 作 MODBUS 网络主局时，将网络子局的一个数据块读入主局中的功能存储器区内。该指令可让用户指定 MODBUS 的功能码、子局号、主局和子局的存储器开始地址，传送数据量、MODBUS 数据类型和异常响应缓冲寄存器。

- **CPU 端口号：**指定使用的 NK0 的通信口，固定 K0。
- **子局号地址：**指定对方子局号（0~247）。
- **功能码：**指定 MRX 指令使用的 MODBUS 功能码。MRX 指令支持下列 MODBUS 功能码：

- 01——读一组线圈
- 02——读一组输入信号
- 03——读保持寄存器
- 04——读输入寄存器
- 07——读异常状态

- **子局开始地址：**指定要读取子局数据的存储器开始地址，给出标准 MODBUS 地址（10 进制和八进制）和对应的 NK0 功能号地址。
- **主局开始地址：**指定存放读入数据的 NK0 主局存储器开始地址。根据 MODBUS 功能码的不同，支持的 NK0 功能存储器类型也不同。
- **数据大小：**指定要读入的线圈/输入点/保持寄存器/输入寄存器数量。
- **超时（ms）：**指定等待通信对方应答的超时时间。
- **MODBUS 数据类型：**指定使用何种 MODBUS 数据格式，可以选择 584/984 或 484 数据格式。
- **通信实行中标志：**启动 MRX 指令通信后置为 ON，通信结束后变为 OFF。
- **通信完成标志：**启动 MRX 指令通信时复位该标志，MRX 指令通信完成后置为 ON 并保持。
- **通信错误标志：**启动 MRX 指令通信时复位该标志，MRX 指令通信出错时置为 ON 并保持。
- **异常响应缓存：**指定主局存放通信异常响应码的存储器地址。

异常响应码内容及其含义如下：

异常响应码 = 1 表示指定的功能码不支持。

异常响应码 = 2 表示指定的地址、范围出错。

异常响应码 = 3 表示要求的数据内容异常。可能指定了当前 PLC 动作模式下是禁止读写的内容。

异常响应码 = 4 表示通信处理过程异常。



1) MRX 子局存储器地址范围

功能码	MODBUS 数据格式	子局地址范围
01—读线圈	484 方式	1 ~ 999
	584/984 方式	1 ~ 65536
02—读输入状态	484 方式	1001 ~ 1999
	584/984 方式	10001 ~ 19999 (5 位) 或 100001 ~ 165536 (6 位)
03—读保持寄存器	484 方式	4001 ~ 4999
	584/984 方式	40001 ~ 49999 (5 位) 或 400001 ~ 465536 (6 位)
04—读输入寄存器	484 方式	3001 ~ 3999
	584/984 方式	30001 ~ 39999 (5 位) 或 300001 ~ 365536 (6 位)
07—读异常状态	484 和 584/984 方式	N/A

2) MRX 主局存储器地址范围

操作数类型	NK0 范围
通信输入 GI	GI0 ~ GI377
输入 I	I0 ~ I377
通信输出 GQ	GQ0 ~ GQ377
输出 Q	Q0 ~ Q377
中间继电器 M	M0 ~ M1777
级 S	S0 ~ S377
定时器 T	T0 ~ T177
计数器 C	C0 ~ C177
特殊继电器 SP	SP0 ~ SP777
数据寄存器 R	R2000 ~ R7377

3) MRX 指令传送数据量指定

操作数类型	NK0 范围
数据寄存器 R	R2000 ~ R7377
常数 K	传送单个位时: 1~2000 传送寄存器时: 1~125

4) MRX 异常响应缓冲寄存器范围

操作数类型	NK0 范围
数据寄存器 R	R2000 ~ R7377

2、MODBUS 网络写指令 MWX

MWX 指令用于将 NK0（MODBUS 网络主局）中的一块数据写到网络中某个子局的指定存储区中。该指令可让用户指定 MODBUS 的功能码、子局号、主局和子局的存储器开始地址，传送数据量、MODBUS 数据类型和异常响应缓冲寄存器。

- **CPU 端口号：**指定使用的 NK0 的通信口，固定 K0。
- **子局号地址：**指定对方子局号（0~247）。
- **功能码：**指定 MWX 指令使用的 MODBUS 功能码。MWX 指令支持下列 MODBUS 功能码：
 - 05——强制置位/复位单个线圈
 - 06——改写单个寄存器
 - 15——强制置位/复位一组线圈
 - 16——改写一组寄存器
- **子局开始地址：**指定要写入数据的子局存储器开始地址，给出标准 MODBUS 地址（10 进制和八进制）和对应的 NK0 功能号地址。
- **主局开始地址：**指定要读出数据的 NK0 主局存储器开始地址，根据 MODBUS 功能码的不同，支持的 NK0 功能存储器也不同。
- **数据大小：**指定要写出的线圈/输入点/保持寄存器/输入寄存器数量。
- **超时（ms）：**指定等待通信对方应答的超时时间。
- **MODBUS 数据类型：**指定使用何种 MODBUS 数据格式，可以选择 584/984 或 484 数据格式。
- **通信实行中标志：**启动 MWX 指令通信后置为 ON，通信结束后变为 OFF。
- **通信完成标志：**启动 MWX 指令通信时复位该标志，MWX 指令通信完成后置为 ON 并保持。
- **通信错误标志：**启动 MWX 指令通信时复位该标志，MWX 指令通信出错时置为 ON 并保持。
- **异常响应缓存：**指定主局存放通信异常响应码的存储器地址。

异常响应码内容及其含义如下：

 - 异常响应码 = 1 表示指定的功能码不支持。
 - 异常响应码 = 2 表示指定的地址、范围出错。
 - 异常响应码 = 3 表示要求的数据内容异常。可能指定了当前 PLC 动作模式下是禁止写入的内容。
 - 异常响应码 = 4 表示通信处理过程异常。
 - 异常响应码 = 5 表示通信对方应答超时。



1) MWX 子局存储器地址范围

功能码	MODBUS 数据格式	子局地址范围
05—强制置位/复位单个线圈	484 方式	1 ~ 999
	584/984 方式	1 ~ 65536
06—改写单个寄存器	484 方式	4001 ~ 4999
	584/984 方式	40001 ~ 49999 (5 位) 或 400001 ~ 465536 (6 位)
15—强制置位/复位一组线圈	484 方式	1 ~ 999
	584/984 方式	1 ~ 65536
16—改写一组寄存器	484 方式	4001 ~ 4999
	584/984 方式	40001 ~ 49999 (5 位) 或 400001 ~ 465536 (6 位)

2) MWX 主局存储器地址范围

操作数类型	NK0 范围
通信输入 GI	GI0 ~ GI377
输入 I	I0 ~ I377
通信输出 GQ	GQ0 ~ GQ377
输出 Q	Q0 ~ Q377
中间继电器 M	M0 ~ M1777
级 S	S0 ~ S377
定时器 T	T0 ~ T177
计数器 C	C0 ~ C177
特殊继电器 SP	SP0 ~ SP777
数据寄存器 R	R2000 ~ R7777

3) MWX 指令传送数据量指定

操作数类型	NK0 范围
数据寄存器 R	R2000 ~ R7377
常数 K	传送单个位时: 1~2000 传送寄存器时: 1~125

4) MWX 异常响应缓冲寄存器范围

操作数类型	NK0 范围
数据寄存器 R	R2000 ~ R7377

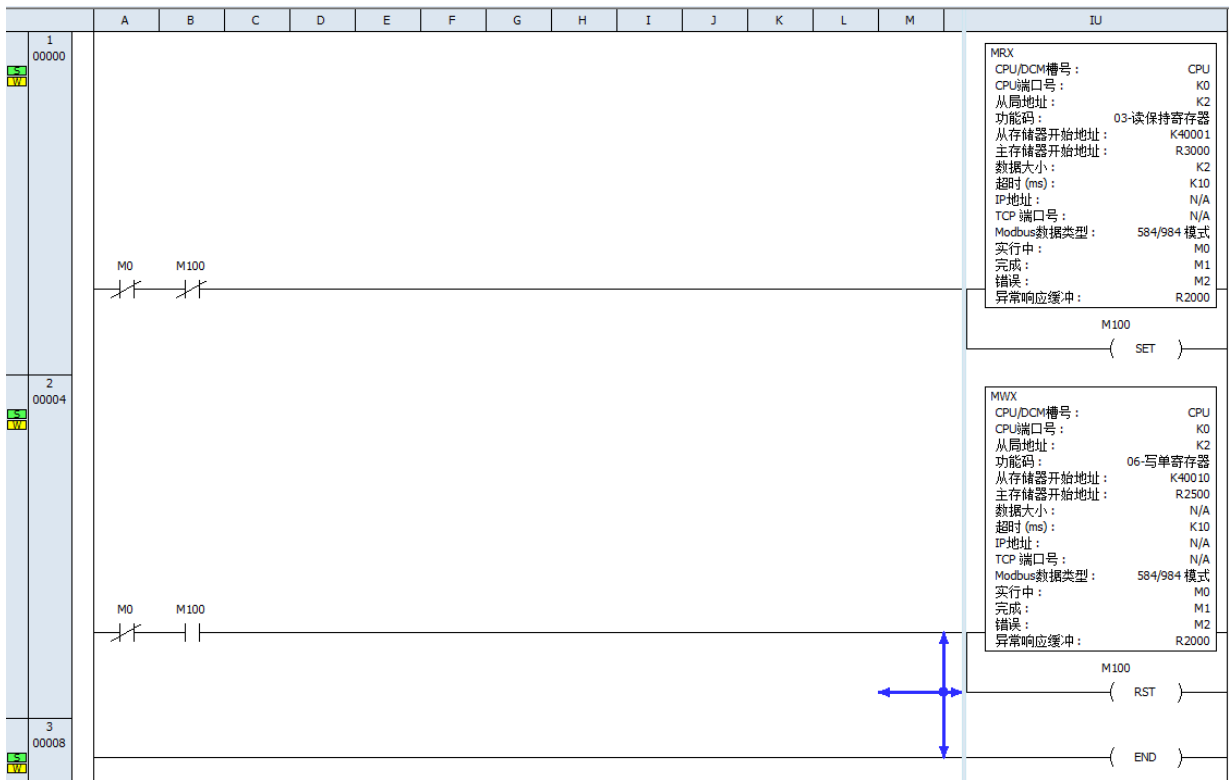
3、KPPSoft 软件中使用 MRX/MWX 的例子

下面的程序段使用 NK0 的 RS-485 通信口来实现 NK0 与支持 MODBUS RTU 通信协议的带 RS-485 串行通信口的设备之间的数据交换功能。

程序中用 MODBUS 协议读通信指令 MRX 的功能码 03，读出局号为 2 的子局的保持寄存器 40001~40002 的内容，保存到 NK0 的 R3000，R3001 中；用写通信指令 MWX 的功能码 06，把 NK0 主局寄存器 R2500 中的值写入局号为 2 的子局的保持寄存器 40010 中。

由于在一次扫描中只能有一个通信指令被执行，因此，为了保证读、写通信程序都能被执行到，这里使用了互锁线圈。

注意：本段程序利用了 MRX/MWX 指令的通信实行中标志（M0）来作为 MRX/MWX 指令的执行条件之一，以保证仅在 RS-485 通信口处于空闲状态时才启动一次通信。你还可以利用另外 2 个通信标志（通信完成，通信错误）来判断通信是否正常结束，通信数据是否已经正常发送/接收了。



4-5 NK0 系列 PLC 无协议串行通信的实现

NK0 系列 PLC 的 RS-485 通信端口可设置成无协议通信模式，与带有串行通信端口的智能终端连接，实现数据的交互通信。

4-5-1 NK0 系列 PLC 支持无协议通信功能

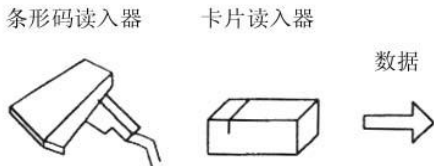
1、无协议串行通信功能分类

- ① 串行接收…接收串行数据，将数据写入 PLC 寄存器中。
- ② 串行发送…将存储在 PLC 寄存器内的数据从串行端口发送出去。
- ③ 串行接收/发送…可进行串行数据的发送和接收。

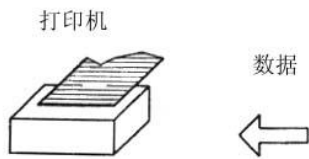


2、可连接的设备

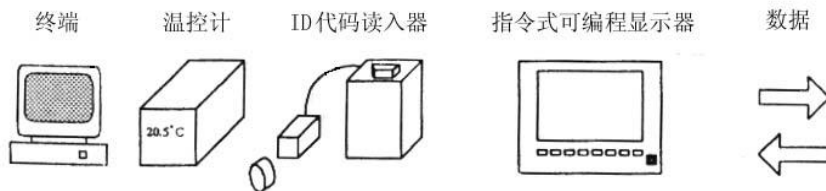
- ① 串行接收：条形码读入机，卡片读入机



- ② 串行发送：打印机



- ③ 串行接收/发送：智能终端、温控仪、ID 码读入器、指令式可编程显示器。



3、 通信流程控制

在进行无协议串行通信时，有 2 种通信流程控制功能：软件流程控制和硬件流程控制。

① 软件流程控制

用 X-ON (0x11H)、X-OFF (0x13H) 码进行通信流程控制。

- 数据接收终端在接收缓冲区放满时送出 X-OFF 代码。

PLC 在收到 X-OFF 码时，会中断通信发送。

- 数据接收终端的接收缓冲区空时送出 X-ON 代码。

PLC 在收到 X-ON 码时，再开始通信发送。

② 硬件流程控制

利用 RTS/CTS 硬件信号来控制数据的发送。NK0 系列 PLC 的 RS-485 通信口不支持硬件流程控制方式。

4、 NK0 系列 PLC 支持的无协议通信模式

NK0 系列 PLC 的无协议通信支持以下四种通信模式。

无协议通信模式	设定模式码	备注
A 型串行数据接收（模式 61）	0x0061	固定格式数据接收通信
B 型串行数据接收（模式 62）	0x0062	固定格式数据接收通信
A 型可变格式串行数据接收发送（模式 70）	0x0070	RTS 控制无效， X-ON/X-OFF 控制无效
B 型可变格式串行数据接收发送（模式 71）	0x0071	RTS 控制无效， X-ON/X-OFF 控制有效

下面简单说明这 4 种无协议通信

① A 型串行数据接收模式（模式 61）

针对特定的条码读入机等设备的数据接收通信模式，规定有简单通信协议如下：

STX	ASCII 数据（127 字节以内）	ETX	BCC
-----	--------------------	-----	-----

数据格式：

BCC：从 STX (02H) 到 ETX (03H) 所有数据的按位异或和数据

1 字节数据： 7 位长，偶校验

数据帧 127 字节以下 ASCII 码数据

NK0 系列 PLC 正常接收到数据时，发出 ACK；数据接收异常时，发出 NAK。

② B 型串行数据接收模式（模式 62）

也是针对特定的条码读入机等设备的数据接收通信模式，规定有简单通信协议如下：

STX	ASCII 数据（127 字节以内）	ETX	BCC
-----	--------------------	-----	-----

数据格式：

BCC：从第一个 ASCII 数据到 ETX (03H) 的按位异或和数据

1 字节数据： 7 位长，偶校验

数据帧 127 字节以下 ASCII 码数据

NK0 系列 PLC 正常接收到数据时，发出 ACK；数据接收异常时，发出 NAK。

③ 自由格式串行数据接收发送 X-ON/X-OFF 无效（模式 70）（缺省模式）

自由格式串行通信对于数据发送时的发送数据格式没有限制；对于通信接收数据格式以及表示通信接收结束的通信结束码都可通过用户通过参数任意设置。可通信的数据一般为 ASCII 码数据。

在该模式下，数据接收结束码并不一定必须。当没有设置通信结束码时，NK0 系列 PLC 如果在等待三个字符通信时间后，没有接收到新的字符，就自动结束本次数据接收通信。如果设置了结束码，则 NK0 系列 PLC 在接收到通信接收结束码，或自动在等待三个字符通信时间后（如果没有接收到新的字符），结束本次数据接收通信。

ASCII 码数据	通信结束码 1	通信结束码 2	模式 70 接收数据格式
-----------	---------	---------	--------------

数据格式：

数据位：7 位、8 位

奇偶校验位：无/奇校验/偶校验

停止位：1 位、2 位

数据帧：128 字节以下

通信结束码：1 字节、2 字节（任意非 0 码） 或无结束码

在模式 70 下，通信软件流程控制信号（X-ON/X-OFF）和硬件流程控制信号（RTS）都无效。

④ 自由格式串行数据接收发送 X-ON/X-OFF 有效（模式 71）

模式 71 自由格式串行通信与模式 70 基本相同，其唯一的不同是通信的软件流程控制信号 X-ON/X-OFF 信号在模式 71 下是有效的。

ASCII 码数据	通信结束码 1	通信结束码 2	模式 71 接收数据格式
-----------	---------	---------	--------------

模式 71 的数据格式与模式 70 相同。

在模式 71 下，NK0 数据发送过程会被接收到的 X-OFF（0x13）信号终止，NK0 一旦接收到 X-OFF 信号，就会停止数据的发送；直到接收到 X-ON（0x11H）信号才继续数据的发送。

4-5-2 NK0 系列 PLC 无协议通信参数设置

要使用 NK0 RS-485 口的无协议通信功能，首先需要设置该通信口的无协议通信参数，NK0 RS-485 通信口无协议通信参数存放在一组特殊参数寄存器中如下：

参数内容	特殊参数寄存器
通信协议选择	R7650
通信超时时间，通信时的 RTS ON 延时时间	R7651
数据位长、停止位、奇偶校验位、通信波特率、应答延时时间	R7652
无协议通信模式	R7653
开始代码（预留）	R7654
通信结束码	R7655
接收数据存放寄存器首地址	R7656
无协议通信参数设置完成标志	R7661

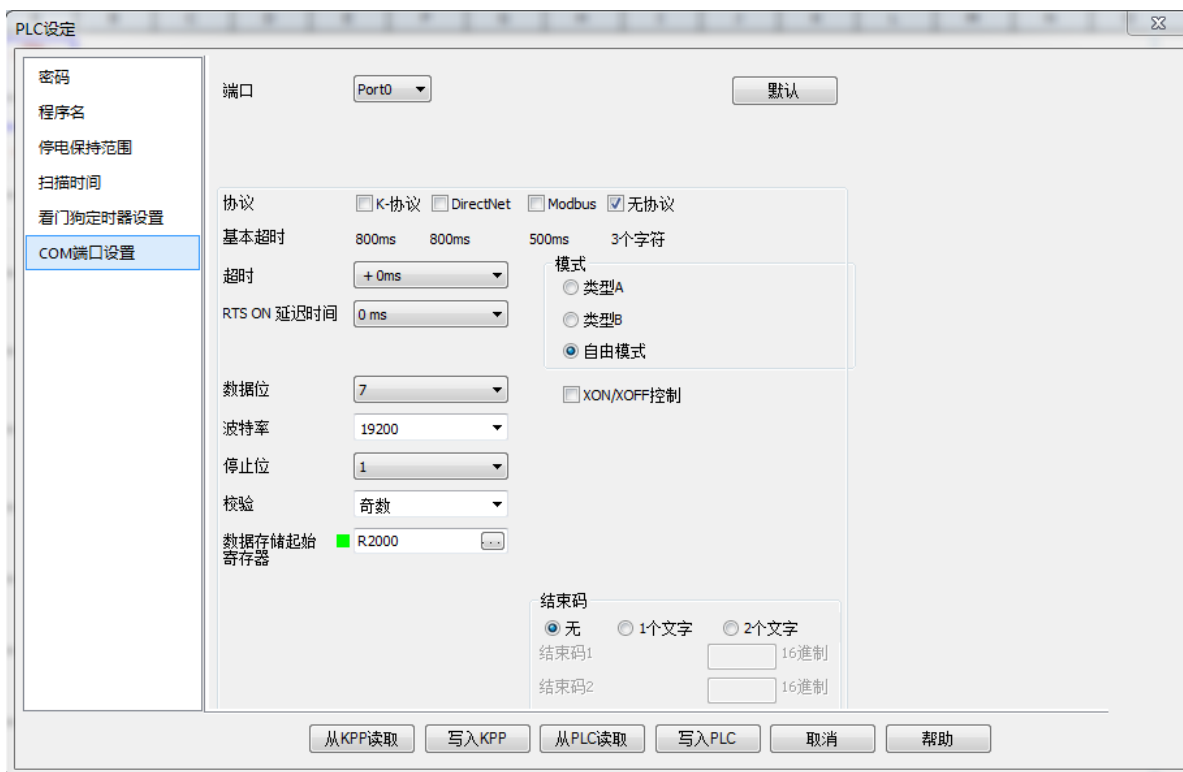
有关 NK0 系列 PLC 无协议通信参数特殊寄存器内容的详细说明请参见 4-2-2 节介绍。

NK0 RS-485 通信口通信参数的设置方法有 KPPSoft 软件设置和用户 PLC 程序设置 2 种，下面给出该通信口无协议通信参数的设置方法例子，假设需要设置的通信参数如下。

通信协议 : 无协议通信 模式 70 (X-ON/X-OFF 控制无效)
 通信数据格式 : 7 位数据位/1 位停止位/奇校验/通信速率 19200bps
 接收数据存放寄存器 : R2000
 通信应答延时 : 0ms
 通信结束码 : 无

设置方法一：通过 KPPSOFT 软件设置

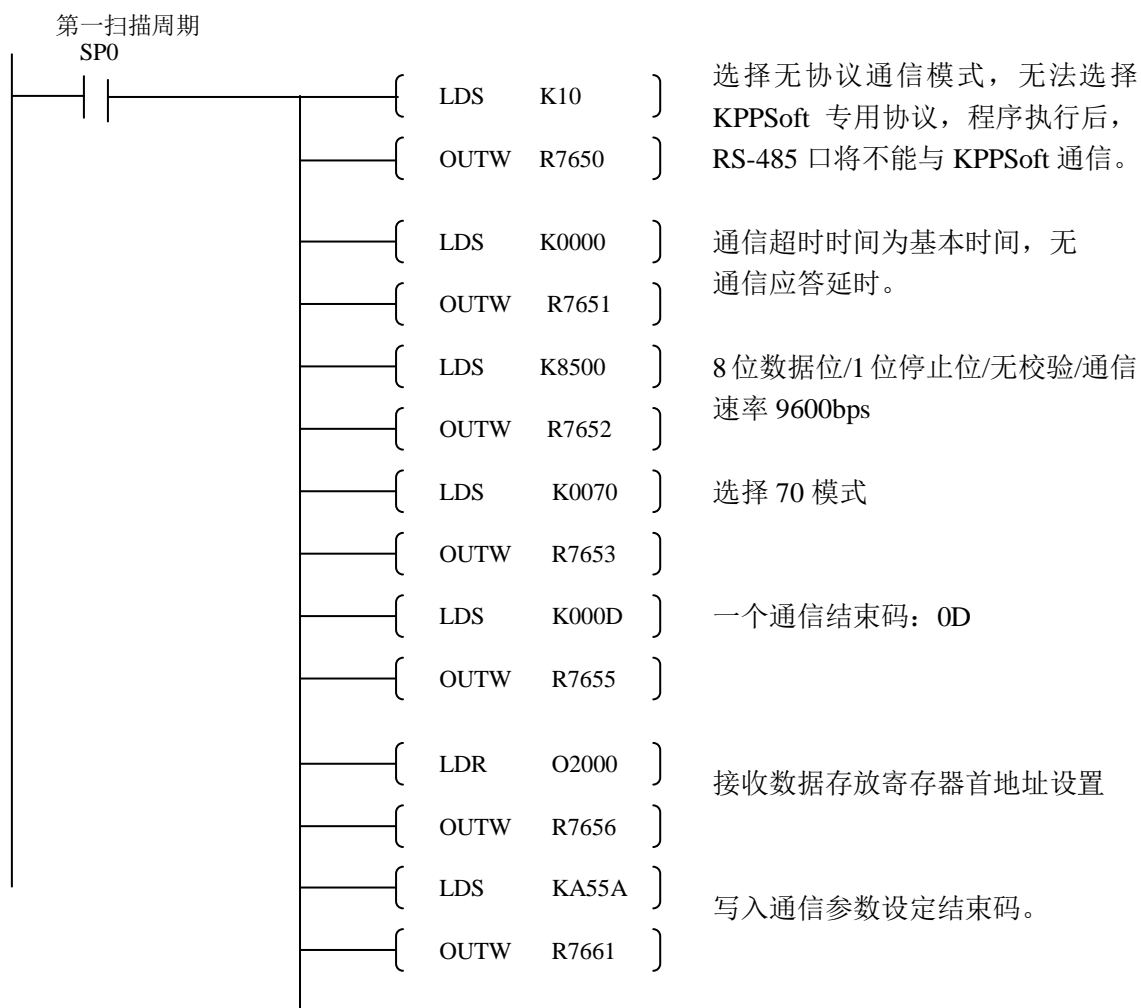
通过[PLC->PLC 设定->COM 端口设置]菜单,打开 NK0 通信端口设置窗口，设置其通信参数如下图所示。



注：通过 KPPSoft 软件把 NK0 的 RS-485 通信口设置成无协议通信后，则 NK0 与 KPPSoft 的通信连接将立即中断。如果需要再次建立 NK0 与 KPPSoft 的通信连接，需要先用运行模式开关把 NK0 运行状态切换到“STOP”模式，然后再建立 KPPSoft 和 NK0 的通信连接。

设置方法二：通过用户程序设置

其设置程序段如下：



在用户程序中加上以上初始程序段，执行该用户程序后，NK0 就能按规定的无协议通信参数初始化 RS-485 通信口。

注意：程序执行后，在 RUN 模式下，KPPSoft 软件将不能和 NK0 建立通信连接。

4-5-3 NK0 系列 PLC 无协议通信功能的实现

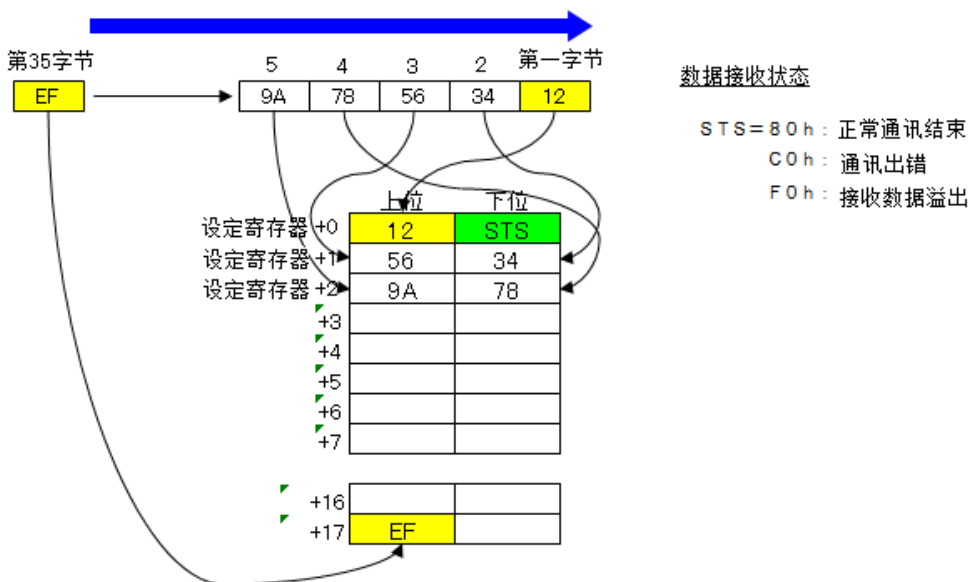
用上面的方法设置好 NK0 RS-485 通信口的通信参数后，该通信口就处于无协议通信准备就绪状态。

对于 NK0 系列 PLC，其无协议通信数据接收功能的实现并不需要编制通信程序；而其无协议通信数据发送将通过 WX 指令实现。当 NK0 RS-485 通信口的无协议通信参数初始化完成后，该通信口自动处于无协议通信数据接收状态，随时准备从通信口接收无协议通信数据。当 NK0 执行无协议通信数据发送指令 WX 指令后，则 NK0 根据指令参数要求，把相应的数据通过指定的通信口发送出去。数据发送完成后，该通信口又返回无协议通信数据接收状态，准备从通信口接收新的无协议通信数据。

1、NK0 无协议通信接收数据寄存器存放格式

NK0 无协议通信时接收数据存放在“接收数据存放寄存器首址”开始的一组寄存器中。模式 61/62 和模式 70/71 无协议通信模式下接收数据存放寄存器存放数据的格式稍微有所不同，下面分别说明。

模式 61/62 无协议通信模式下，其数据存放格式如下：



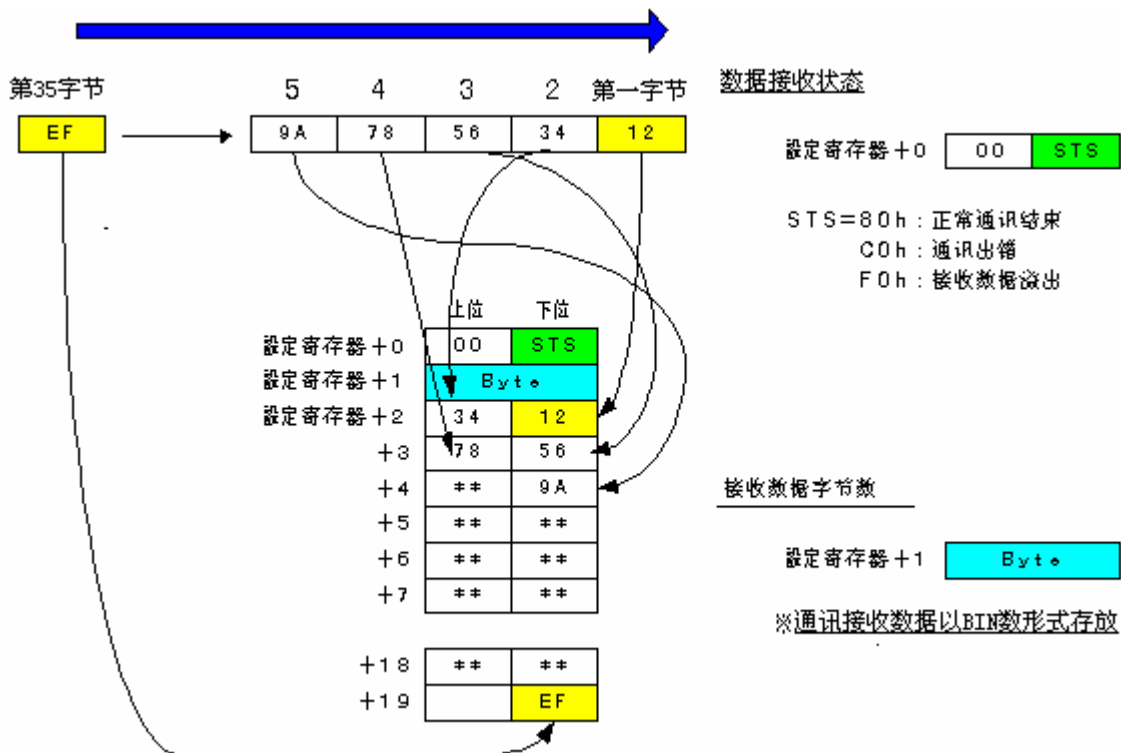
- 接收数据寄存器首地址的低字节存放通信状态字 (STS);
STS = 00H, 表示 NK0 已经作好数据接收准备，可以接收新的通信数据；
STS = 80H, 表示数据接收通信正常结束；
STS = COH, 表示数据接收通信发生了错误；
STS = FOH, 表示发生了接收数据溢出错误。

实际编制程序时，你可以根据 STS 的内容，来处理接收到的数据。另外，为了能够接收下一帧通信数据，在处理好通信接收数据后，请务必把 STS 置成 00H。如果 STS 为一非 0 数据，则 NK0 判断为没有对接收数据进行处理，而不接收下一帧通信数据。

- 接收数据寄存器首地址的高位字节，从该字节开始存放接收到的通信数据。
- 接收数据寄存器+1 该寄存器存放通信接收的第二、第三字节数据，低字节存放第二个通信接收数据；高字节存放第三个通信接收数据。

后面依次存放所有通信接收数据。

模式 70/71 无协议通信模式下，其数据存放格式如下：



- 接收数据寄存器首地址的低字节存放通信状态字（STS），高字节为00；
STS = 00H，表示 NK0 已经作好数据接收准备，可以接收新的通信数据；
STS = 80H，表示数据接收通信正常结束；
STS = COH，表示数据接收通信发生了错误；
STS = FOH，表示发生了接收数据溢出错误。

实际编制程序时，你可以根据 STS 的内容，来处理接收到的数据。另外，为了能够接收下一帧通信数据，在处理好通信接收数据后，请务必把 STS 置成 00H。如果 STS 为一非 0 数据，则 NK0 判断为没有对接收数据进行处理，而不接收下一帧通信数据。

- 接收数据寄存器+1 接收数据字节数（BIN 数表示），最大 128 字节（80h）；
- 接收数据寄存器+2 从该寄存器的低字节开始存放通信接收数据；

后面依次存放所有通信接收数据。

2、 模式 61/模式 62 下无协议通信数据接收的实现

NK0 RS-485 通信口无协议通信数据接收功能的实现并不需要编制特别的通信程序,而只要设置好该通信口对应的通信参数寄存器,进行简单的接收数据寄存器清零操作即可。下面说明 NK0 RS-485 通信口在模式 61/模式 62 下数据接收功能的实现方法过程。

- (1) 在 R7650 中写入 10h, 选择无协议通信方式;
- (2) 在 R7651 中设置好通信超时时间;
- (3) 在 R7652 中设定传送速度、应答延迟时间参数; (数据位长, 奇偶校验位固定)
- (4) 在 R7653 中写入 61 或 62, 选择合适的通信模式;
- (5) 在 R7656 中设定接收数据的存放起始寄存器号;
- (6) 在 R7661 中写入设定完成码 A55A (H), 完成设置, 进入接收准备完成状态;
- (7) 接收起始寄存器的低字节 (接收状态字节) 清零;
- (8) 接收数据, 自动按规定格式存入指定的寄存器内;
- (9) 在正常结束数据接收时, 送出 ACK 码; 在数据接收异常结束时, 送出 NAK 码。
- (10) 用户程序处理接收数据。对接收数据进行处理后, 请在用户程序中对数据接收状态字节进行清零处理。当该状态字节是 0 以外的数值时, 系统判断为没有对接收数据进行处理, 而不接收下一帧数据。

以上就完成了一次模式 61/模式 62 下数据接收的处理过程。

3、 可变格式 (模式 70/模式 71) 无协议通信数据发送接收的实现

模式 70/模式 71 下, NK0 的 RS-485 通信口既可以实现数据的接收功能, 又可以实现数据的发送功能。同样, 其数据接收功能的实现, 并不需要特别的通信程序, 而数据发送需要通过 WX 指令来实现。下面说明 NK0 RS-485 通信口在模式 70/模式 71 下数据接收/发送功能的实现方法过程。

- (1) 在 R7650 中写入 10h, 选择无协议通信方式;
- (2) 在 R7651 中设置好通信超时时间;
- (3) 在 R7652 中设定数据位、停止位、奇偶校验位、传送速度、应答延迟时间等参数;
- (4) 在 R7653 中写入 70 或 71, 选择合适的通信模式;
- (5) 在 R7655 中设定数据接收结束码;
- (6) 在 R7656 中设定接收数据的存放起始寄存器号;
- (7) 在 R7661 中写入设定完成码 A55A (H), 完成设置, 进入接收准备完成状态;

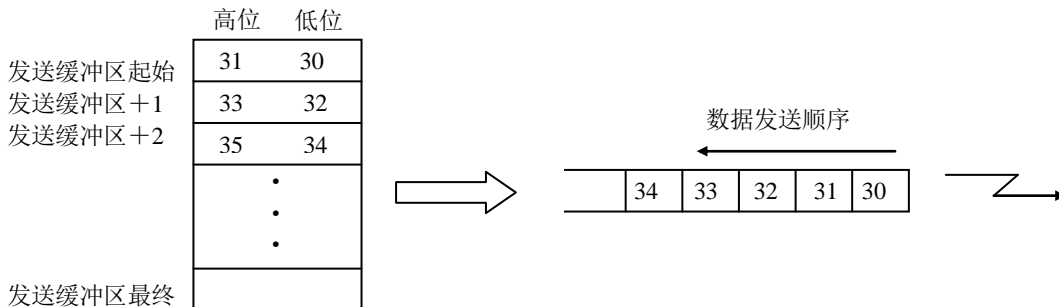
以上完成模式 70/模式 71 的参数寄存器设定。下面按数据接收/发送分别说明实现过程。

(一)、 数据接收时

- (8) 接收起始寄存器的低字节 (接收状态字节) 清零;
- (9) 接收数据, 自动按规定格式存入指定的寄存器内;
- (10) 用户程序处理接收数据。对接收数据进行处理后, 请在用户程序中对数据接收状态字节进行清零处理。当该状态字节是 0 以外的数值时, 系统判断为没有对接收数据进行处理, 而不接收下一帧数据。

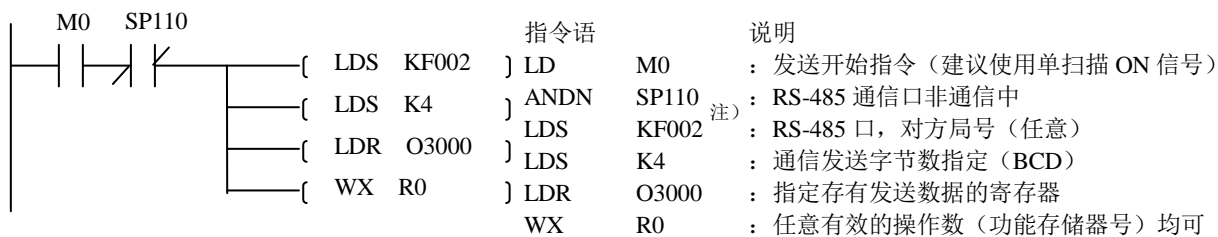
(二)、数据发送时

(8) 把需要发送的数据按发送的顺序，依次存入发送寄存器的低位字节、高位字节，如下图；



- (9) 通过在用户程序中执行 **WX** 指令进行数据发送。为了保证数据发送指令的正确执行，将通信口通信中特殊线圈 **SP110** 为 **OFF** 作为发送指令条件之一加入；
- (10) 发送指令执行条件成立，NK0 执行 **WX** 指令（**SP110** 变为 **ON**），将存放在指定的发送寄存器中的数据按指定的字节数送出。
- (11) 数据发送通信完成后，NK0 自动把 **SP110** 置为 **OFF**。

通过 RS-485 通信口进行数据发送程序例子如下：



发送字节数 ≤ 128。

注) 对方局号可以任意。

NK0 在模式 70/模式 71 下，可以实现数据的接收和发送。通信参数设置完成后，通信口将自动进入通信数据接收准备状态。只有在执行通信发送指令时，才会进入通信数据发送状态，数据发送完成后，通信口又自动转回数据接收准备状态。

由于 RS-485 通信口在同一时间不能既做数据接收工作，又做数据发送工作，所以在利用 NK0 的无协议功能实现数据接收/发送时，一定要注意对通信口数据传送的时序控制。

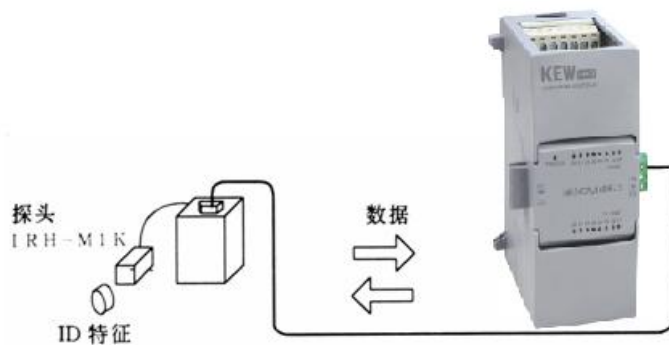
4、 无协议通信数据发送/接收实现的例子程序

(1) 实现功能

利用 NK0 的 RS-485 通信口与 ID 特征码读入机相连，NK0 通过 RS-485 通信口将指令送给 ID 特征码读入机，将读取的 ID 特征码存入寄存器中。

(2) 系统构成

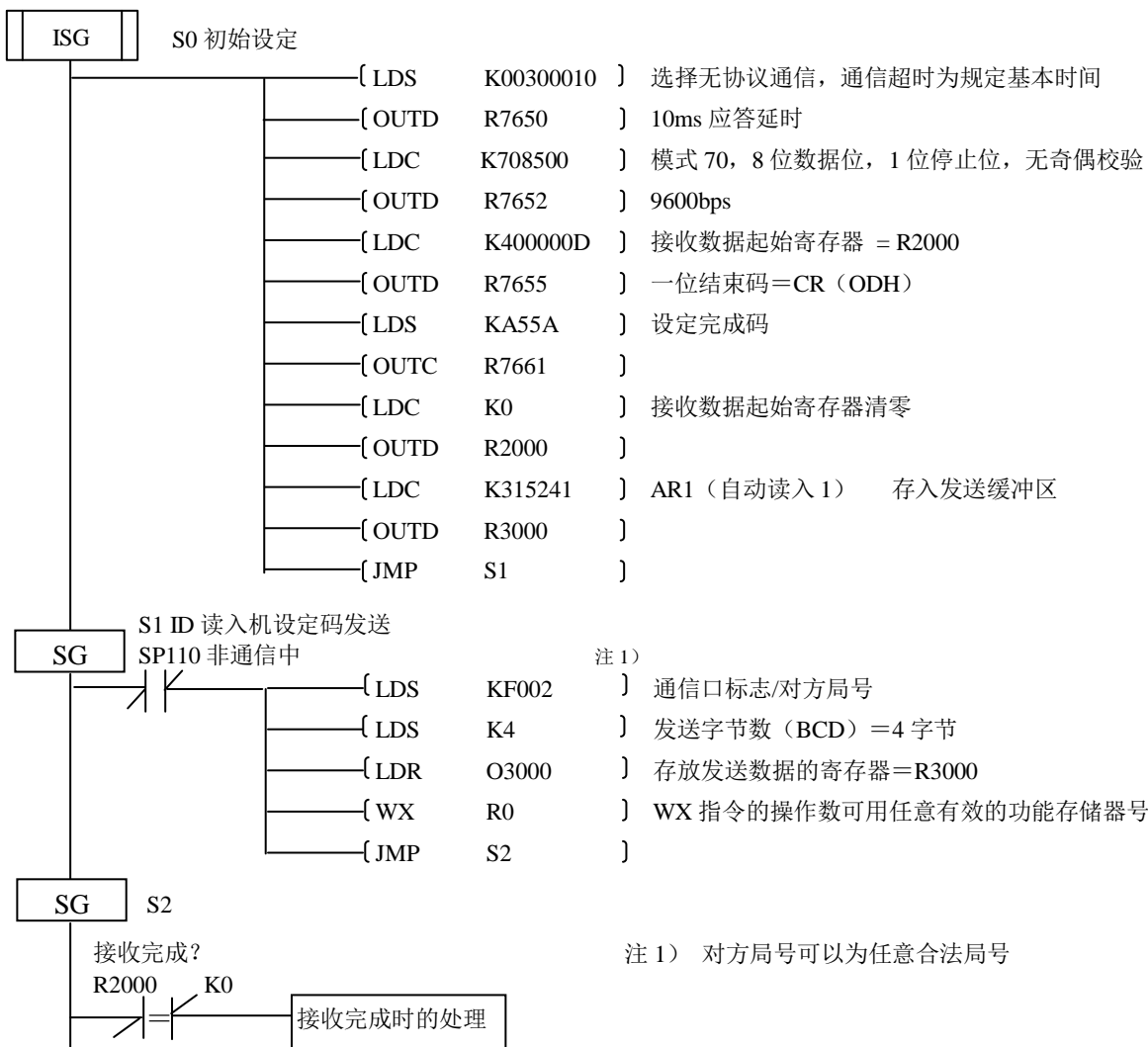
ID 码读入机（IRI-KHA6-4H）



(3) 条件

- ① R7650 = 10 : 选择无协议通信
- ② R7651 = 0030: 通信超时基本时间, 应答延迟时间 10ms
- ③ R7652 = 8500 : 8 位数据位、1 位停止位、无奇偶校验、9600bps、
- ④ R7653 = 70 : 模式 70 自由格式通信 (接收/发送)
- ⑤ R7655 = 0D : 结束码为 CR (0DH), 1 字节
- ⑥ R7656 = 400 : 接收数据存入 R2000 开始的寄存器中

(4) 程序



4-5-4 NK0 系列 PLC 无协议串行通信指令（ASCII IN/OUT 功能）

上一节介绍了标准的 NK0 RS-485 无协议通信方式，除了这些方式外，NK0 还支持用无协议串行通信指令来实现数据传送的功能。NK0 系列 PLC 支持的无协议通信 ASCII 输入/输出指令有：

ASCII 码输入指令： AIN

ASCII 码输出指令： PRINTV

要使用 NK0 的无协议通信 ASCII 输入/输出指令，首先需要把 RS-485 通信口设置成无协议通信模式的自由格式（模式 70 或模式 71）通信模式。

在上一节介绍的标准无协议通信方式下，其数据接收并不需要使用数据接收通信指令 RX。但你也可以使用 AIN 指令来接收数据。如果使用了 AIN 指令，则 AIN 指令的处理将优先执行。

为了避免引起不必要的问题，建议在使用 NK0 RS-485 通信口的无协议通信功能时，不要混合使用标准方式和无协议串行通信 ASCII 输入/输出指令方式。

如果通信口参数设置错误或没有设置成合适的通信模式，则在使用 NK0 的无协议串行通信指令时，会把 SP54 置为 ON。

下面介绍 AIN，PRINTV 指令的用法。

1、 AIN 指令



AIN 指令允许 NK0 系列 PLC 从通信串行通信端口读入一组 ASCII 字符并把它们存入指定的寄存器组中。你可以设置接收固定长度的 ASCII 字符串，或带特殊结束码的可变长度字符串。其它可设置的特性还包括：接收数据存放位置，接收数据字节交换存放，字符超时，以及用户自定义的各种标志（忙、完成、超时出错标志等）。

在 KPPSoft 软件中输入 AIN 指令并按回车，会弹出以上的 AIN 指令参数设置对话框，此时可对 AIN 指令进行设置。

- **字符串类型：**可选择“固定长度”类型和“可变长度”类型。
- **端口号：**选择使用 AIN 指令的串行通信口，NK0 RS-485 通信口设置为 K0。
- **目标数据：**指定存放通过 AIN 指令接收的字符串的数据寄存器开始地址。
（数据目标地址）处存放通信接收字节数。
（数据目标地址+1）处开始存放实际通信接收到数据。
- **固定长度（“固定长度”类型时）：**指定每次通信接收的字符数，最大 128 字节。
- **最大可变长度（“可变长度”类型时）：**指定每次通信接收的最大字符数，最大 128 字节。
- **字符间超时：**如果接收字符间的时间间隔大于该设定时间，则对应的“字符间超时错误”标志会置位，此时不保存接收数据到寄存器中。当设置为“NONE”时，将忽略字符间超时检查特性。

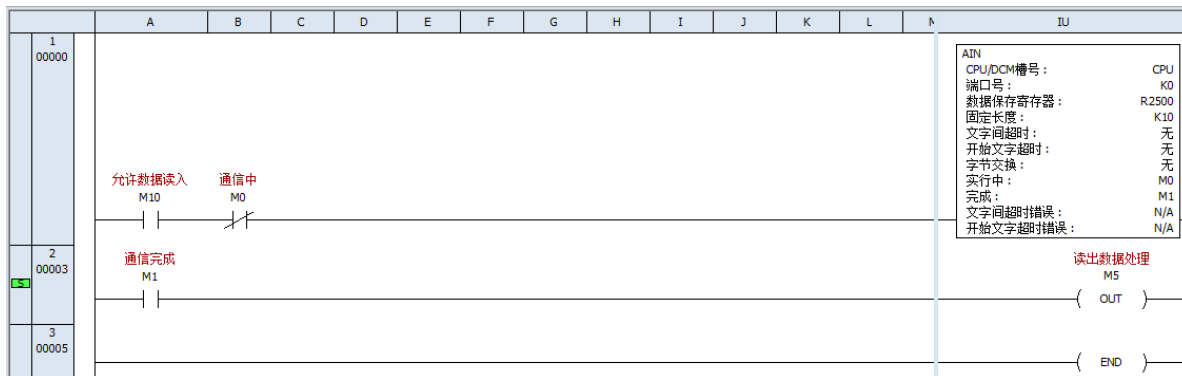
- **开始字符超时:** 如果从 AIN 指令被允许执行到接收到第一个字符间的时间间隔大于该设定时间，则对应的“开始字符超时错”标志会置位。当设置为“NONE”时，将忽略开始字符超时检查特性。
- **字节交换:** 把接收到数据的寄存器高位字节和低位字节数据互换。
- **执行中:** 当 AIN 接收数据指令执行中时，该标志位为 ON，AIN 指令执行完成，该标志为 OFF。
- **完成标志:** “固定长度”类型时，当 AIN 指令成功接收到指定长度的字符串后被置位；“可变长度”类型时，当 AIN 指令成功接收到指定长度的字符串后被置位。当 AIN 指令允许位无效时将被复位。
- **字符间超时错误:** 当接收字符间时间间隔大于字符间超时设定时间时，该标志位被置位。当 AIN 指令允许条件无效时将复位字符间超时错误标志位。
- **开始字符超时错:** 当接收首字符接收等待时间大于首字符超时设定时间时，该标志位被置位。当 AIN 指令允许条件无效时将复位开始字符超时错误标志位。

另外，当选择“可变长度”类型时，还有以下 2 个参数需要设置。

- **结束码:** 设置 AIN 指令数据接收的结束码个数以及结束码内容，可以设置使用一个或二个结束码。
- **溢出错误:** 当接收的字符数超出规定的最大字符数时，该标志位被置位。当 AIN 指令允许条件无效时溢出错误标志将被复位。

“固定长度”类型 AIN 指令程序例

下面的程序段中，NK0 根据 M10 信号的导通情况，不断读取 RS-485 口的数据，存放到 R2500 开始的寄存器组中，并进行后续处理。每次通信读取 10 个字节数据。例子中不考虑各种超时时间。

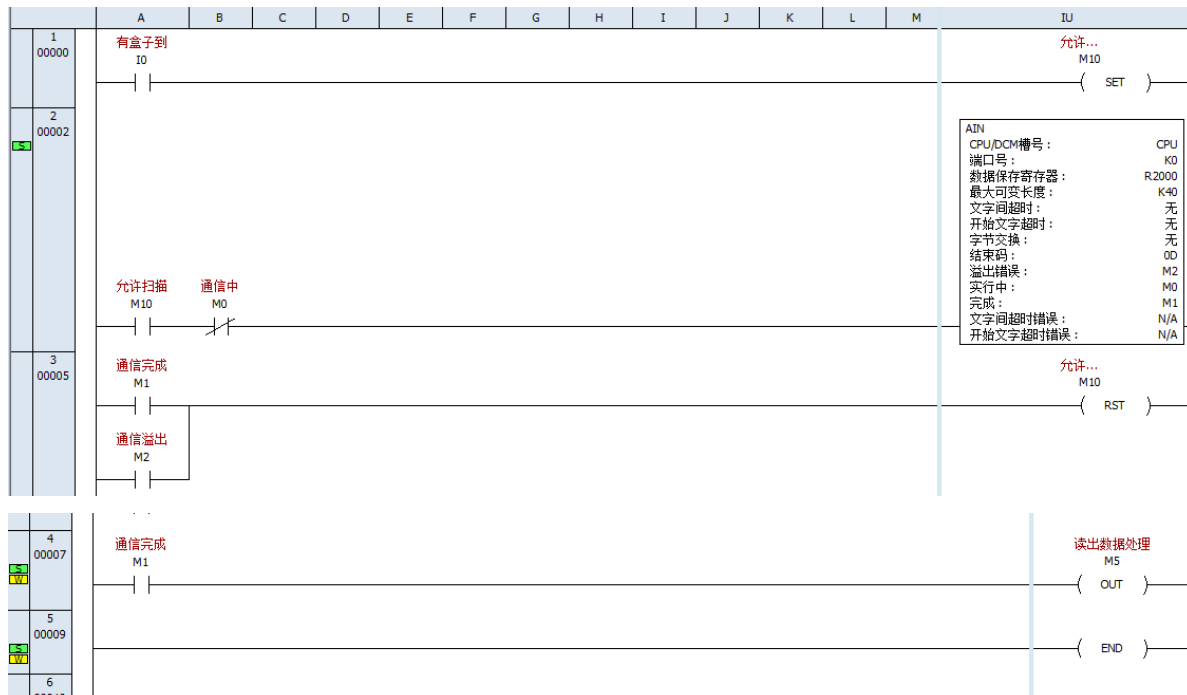


“可变长度”类型 AIN 指令程序例

利用可变长度 AIN 指令读入盒子上条形码的例子程序段。

假设条形码最大长度为 40 字符，以 0Dh 作为条形码接收结束码，字符接收超过 40 个字符时，M2 作为溢出标志被置位。接收到的条形码数据存放在 R2000 开始的寄存器组中。

具体参见如下程序段。



在使用 AIN 指令，设定其参数时可使用的功能存储器范围如下：

参数	NK0 功能存储器范围
目标数据	R2000~R7377(R,P)
固定长度（字节数）	K1~128
最大可变长度（字节数）	K1~128
标志位：实行中、完成、超时、溢出	M0~M1777

使用 AIN 指令时，影响到的特殊线圈如下：

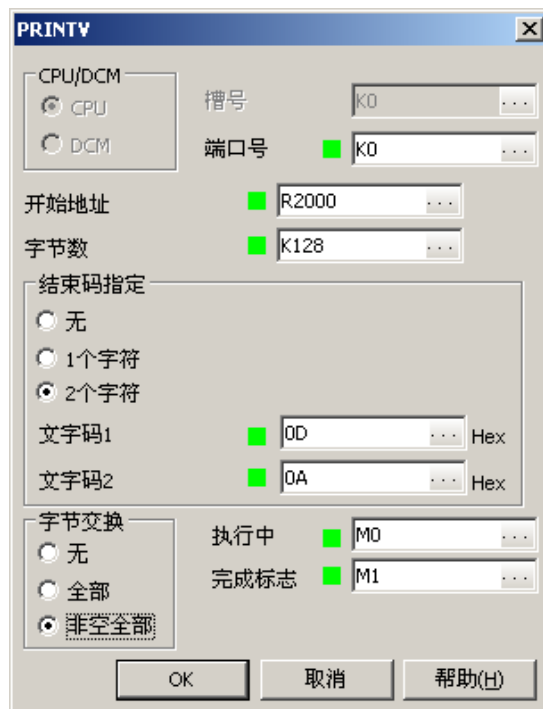
特殊线圈	说明
SP54	当通信口没有被设置成无协议通信时为 ON 或者当指令不能被正确执行时 ON
SP71	数据寄存器间接指定出错时 ON
SP110	当 RS-485 口通信中时 ON
SP111	当 RS-485 口发生通信错误时 ON

2、 PRINTV 指令

该指令把用户预先存放在一组寄存器中的 ASCII 字符通过 NK0 的 RS-485 串行通信端口发送出去。该指令可选择特性包括：通信端口的选择，发送数据存放寄存器首地址的指定，发送字节数的指定，结束码的添加，字节交换与否的设定，以及自定义的指令执行中和完成标志的设置。

PRINTV 指令参数设置画面如右：

- **端口号：**选择使用 PRINTV 指令的串行通信口，NK0 固定 K0（对应 RS-485 通信口）。
- **开始地址：**指定存放通过 PRINTV 指令发送字符串的数据寄存器开始地址。
- **字节数：**指定每次通信送出的字符个数，可以用 K 常数直接指定，也可以用 R 寄存器间接指定。一次通信发送最大 128 字节。
- **结束码指定：**可以指定一个或二个字符（以 16 进制数表示）添加在发送字符串后，以对应有特殊结束字符需要的设备。
- **字节交换：**指定在数据发送时是否交换发送寄存器的高位字节和低位字节数据。
- **执行中标记：**在使用 PRINTV 指令发送数据时为 ON。
- **完成标记：**一次发送完成后，该标记被置位一次，当 PRINTV 指令允许条件无效时该标记将被复位。



在使用 PRINTV 指令，设定其参数时可使用的功能存储器范围如下：

参数	NK0 功能存储器范围
开始地址	所有 R 寄存器 (R,P)
字节数	K1~128
标志位：执行中、完成标志	M0~M1777

使用 PRINTV 指令时，影响到的特殊线圈如下：

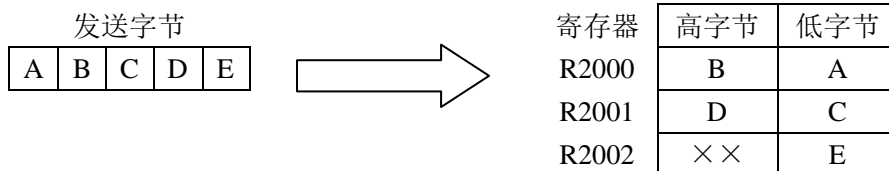
特殊线圈	说明
SP54	当指令不能被正确执行时 ON 或者当通信口没有被设置成无协议通信时为 ON
SP110	当 RS-485 口通信中时 ON
SP111	当 RS-485 口发生通信错误时 ON

注意：请在执行 PRINTV 数据发送指令前，先把有关数据存放到发送字符串寄存器组中。

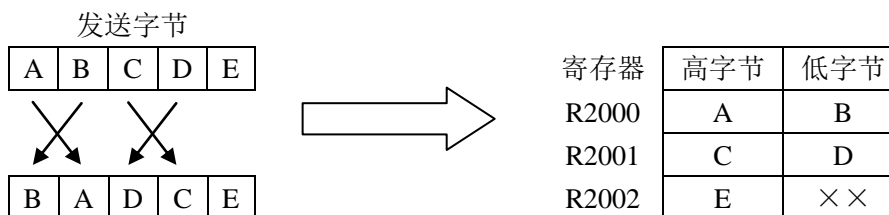
3、 AIN、 PRINTV 指令中字节交换的说明

在使用 AIN、PRINTV 这 2 个指令传送数据时，可以指定传送数据每个寄存器字内的高、低位字节数据的发送顺序，你可以在不交换、全部交换、非空数据全部交换这 3 种方式中选择一种方式传送数据。下面以传送 A、B、C、D、E（先传送 A，后 B，后 C，后 D，最后传送 E）这 5 个字节数据为例说明这 3 种方式下数据在 NK0 寄存器（假设从 R2000 开始）的存放情况。

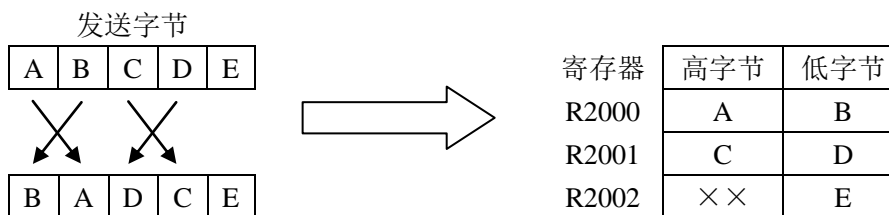
1)、无字节交换



2)、全部交换



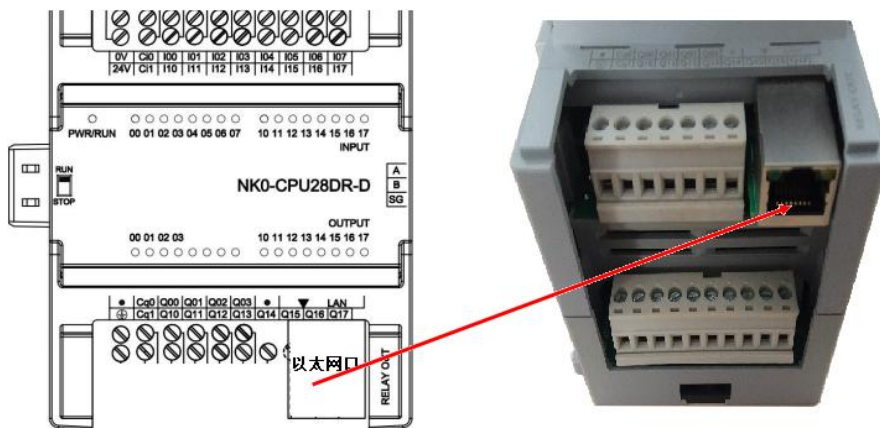
3)、非空数据全部交换



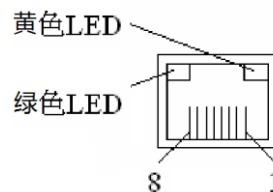
4-6 NK0 系列 PLC 以太网通信（仅 NK0-CPU28DR-D）

4-6-1 NK0 系列 PLC 以太网通信端口规格及通信线连接

NK0 系列 PLC 中，目前仅 NK0-CPU28DR-D 带有一个 10M/100M 自适应以太网通信口，该通信口在 NK0-CPU28DR-D 上位置示意图如下。



在 NK0-CPU28DR-D 本体下侧输出端子台盖板下右侧，配有一个 RJ45 标准的以太网口（如右），该以太网通信口为符合 IEEE802.3 规格的标准以太网口，通信速度根据所连接的网路，系统自己选择 10M 或 100M，支持 MODBUS TCP 通信协议，可用于连接 KPPSoft 编程软件或其他支持 MODBUS TCP 通信协议的以太网通信设备。



NK0-CPU28DR-D 以太网通信口（简称 port3 口）的规格如下：

- 标准 RJ45 插座和插头、符合 IEEE802.3 规格。
- 支持 ECOM 协议（内含 KPPSoft 编程软件专用通信协议、MODBUS TCP 协议）
- 通信速度 10/100M 自适用。
- 端间传输距离不超过 100 米（STP）。
- 连接数：MODBUS TCP 连接数 4 个，ECOM UDP 连接多个，编程软件 1 个。

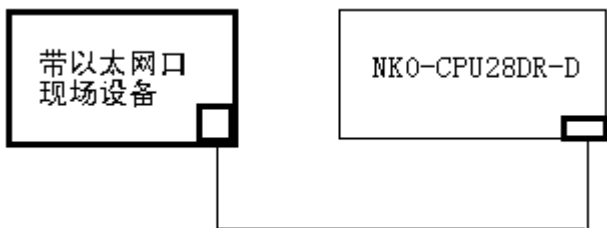
	引脚号	信号名	引脚号	信号名
	1	TD+	5	空
	2	TD-	6	RD-
	3	RD+	7	空
	4	空	8	空

黄色LED：收发信动作中亮灯；绿色LED：正常连接时亮灯。

NK0-CPU28DR-D 以太网口通信电缆使用标准的以太网通信电缆，你可以直接购买商用的标准超 5 类以太网通信电缆，也可以自己制作以太网通信电缆。注意在制作通信电缆时，请使用超 5 类双绞电缆线。同时需要 TD+、TD-使用一对双绞线；RD+、RD-使用一对双绞线。

根据NK0-CPU28DR-D与以太网设备的连接形式的不同，需要使用不同的以太网电缆，下面分别介绍。

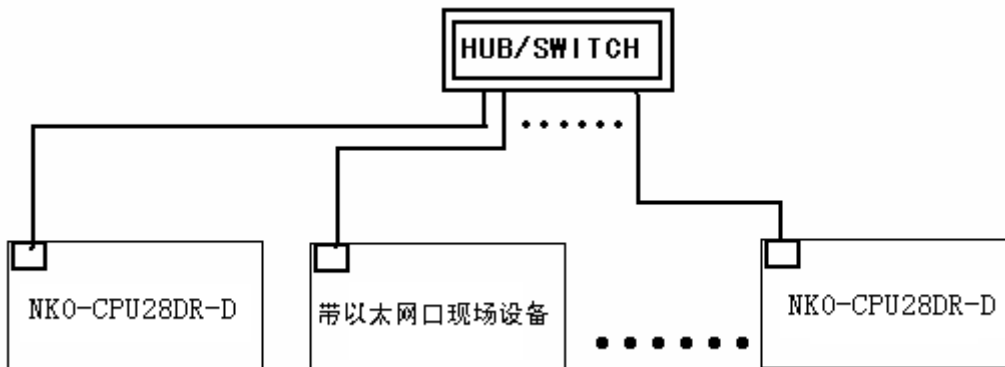
- ① NK0-CPU28DR-D与以太网设备直接一对一连接时，连接例子如下。



这时要制作使用直接连接电缆，连接双方的2对信号线要分别交叉连接，如下所示。

NK0-CPU28DR-D以太网口			其他以太网通信设备		备注
引脚号	信号名		引脚号	信号名	
1	TD+		1	TD+	
2	TD-		2	TD-	
3	RD+		3	RD+	
6	RD-		6	RD-	

- ② NK0-CPU28DR-D通过HUB等设备加入以太网网络中时连接例子如下。



这时要制作使用以太网HUB连接电缆，连接双方的2对信号线要直接连接，如下所示。

NK0-CPU28DR-D以太网口			其他以太网通信设备		备注
引脚号	信号名		引脚号	信号名	
1	TD+		1	TD+	
2	TD-		2	TD-	
3	RD+		3	RD+	
6	RD-		6	RD-	

注：现在一般的HUB/SWITCH设备，具备自动跳线功能，这时信号线交叉的直接连接电缆也可以使用在HUB连接方式中，这点在使用时请注意。

4-6-2 NK0 系列 PLC 以太网通信端口支持的通信协议

NK0-CPU28DR-D 以太网通信口支持的通信协议如下：

协议	以太网通信口 (○：表示支持连接)	
	RUN	STOP
MODBUS TCP	○	○
SMTP	○	○
DHCP	○	○
HAP+UDP	○	○
KPPSoft 专用协议	○	○

说明：1)利用NK0-CPU28DR-D的以太网通信口可以实现KPPSoft 工具软件连接(注意使用KPPSoft 软件的计算机以及所连接的NK0上的以太网口的IP地址需要设置在同一个网段中!)，MODBUS TCP 通信（主/从局），电子邮件发送等功能。

2) NK0-CPU28DR-D 作为 MODBUS TCP 子局使用时，其 TCP 端口号固定为 502。

4) 虽然可以通过以太网口把 NK0-CPU28DR-D 加入到公共以太网网络中，但在使用中需要配置必要的网络安全设备，以保证 NK0-CPU28DR-D 系统的安全性。如果不是必须，建议不要把 NK0-CPU28DR-D 加入公共以太网网络中。

4-6-3 NK0 系列 PLC 以太网通信端口通信参数设置寄存器

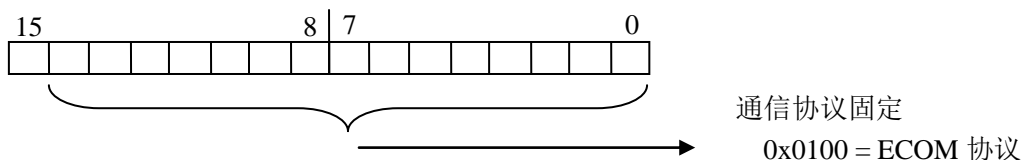
为了使用 NK0 系列 PLC 的以太网通信口，首先要对该通信口进行包括通信协议、节点号、IP 地址、子网掩码、网关等通信参数的设置，与串行通信口一样，这些通信参数也存放在一组特殊寄存器中。下面给出 NK0 系列 PLC 以太网通信口设置特殊寄存器一览表如下。

P3 以太网口用特殊寄存器	参数内容	寄存器出厂初始值
R7706	协议寄存器，固定 0X0100，ECOM 协议	0X0100
R7707	节点号	0X0001
R7710	IP 地址分配方式指定	1
R7711	IP 地址 下位	0X0107
R7712	IP 地址 上位	0XC0A8
R7713	子网掩码 (MASK) 地址 下位	0XFF00
R7714	子网掩码 (MASK) 地址 上位	0XFFFF
R7715	网关 (GateWay) 地址 下位	0X0101
R7716	网关 (GateWay) 地址 上位	0XC0A8
R7717	通信参数设置完成标志	0

各设置寄存器的功能含义说明如下。

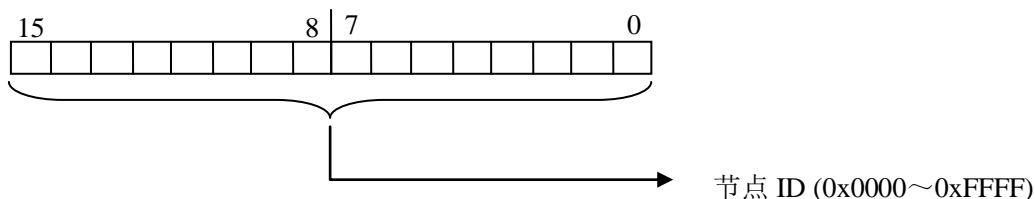
1、协议设定寄存器 R7706

选择设定以太网通信口使用的通信协议，NK0 以太网通信口只支持 ECOM 协议，本协议寄存器的内容固定为 0x0100。注意 NK0 系列 PLC 的 ECOM 协议包括对 MODBUS TCP、DHCP、SMTP、KPPSoft 专用通信协议等的支持。



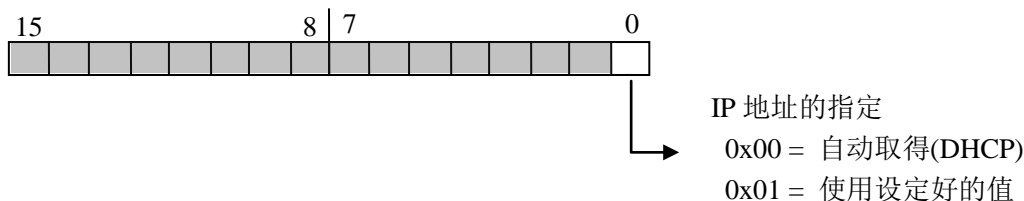
2、节点 ID 号寄存器 R7707

设定以太网通信口的节点 ID 号，范围为 0x0000~0xFFFF (0~65535)



3、IP 地址分配方式指定寄存器 R7710

指定以太网通信口的 IP 地址获得方法，可以选择 DHCP 自动获取或手动分配 2 种方式，手动分配时注意 IP 地址、子网掩码地址、网关地址的正确性。

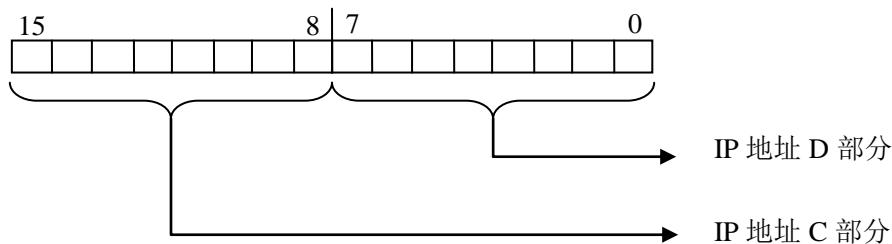


当采用 DHCP 自动 IP 地址分配时，下面的 R7711, R7712, R7713, R7714, R7715, R7716 的内容就不需要手动设置了，这些地址内数值会在 NK0 加入以太网、自动获取有关地址后被更新。

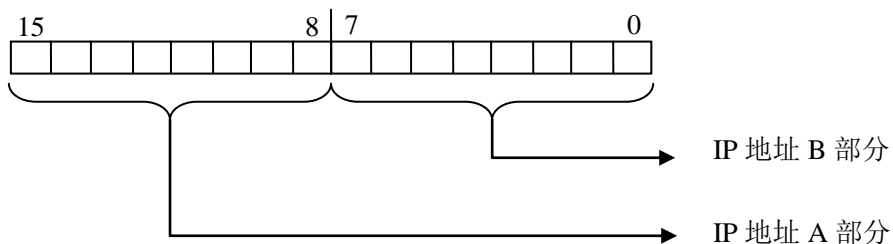
4、IP 地址指定寄存器 R7711, R7712

手工设定以太网的 IP 地址。IP 地址的 4 段数值转换成 HEX 数，存放在 R7711, R7712 中。假设 IP 地址为 A.B.C.D,则设置如下 R7711 = CD R7712 = AB

R7711 寄存器存放内容



R7712 寄存器存放内容



例如 IP 地址为 192.168.11.1 时，其数值变换如下：

A = 192 (0xC0) B = 168(0xA8) C = 11 (0x0B) D = 1(0x01)

则数值存放如下：

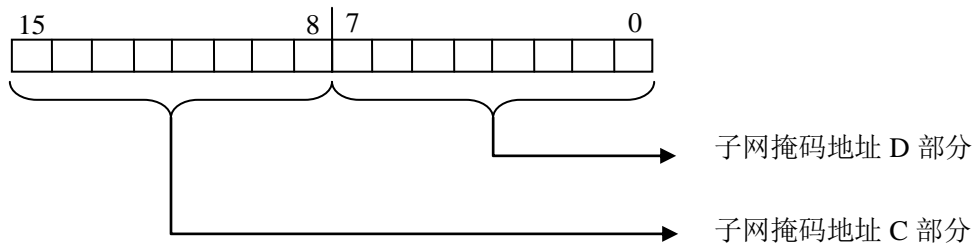
R7711 = 0x0B01 R7712 = 0xC0A8

5、子网掩码地址指定寄存器 R7713, R7714

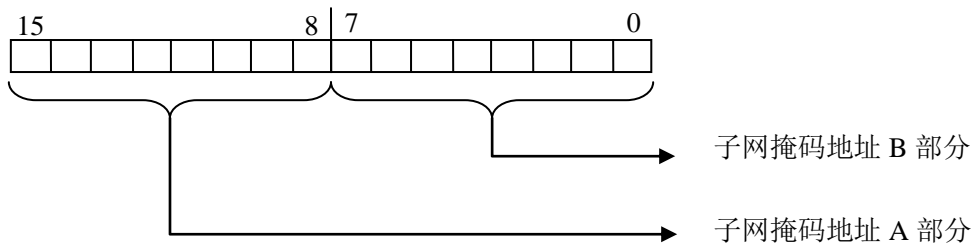
手工设定以太网的子网掩码 (MASK) 地址。MASK 地址的 4 段数值转换成 HEX 数，存放在 R7713, R7714 中。

假设子网掩码地址为 A.B.C.D,则设置如下 R7713 = CD R7714 = AB

R7713 寄存器存放内容



R7714 寄存器存放内容



例如 子网掩码地址为 255.254.253.0 时，其数值变换如下：

A = 255 (0xFF) B = 254(0xFE) C = 253 (0xFD) D = 0(0x00)

则数值存放如下：

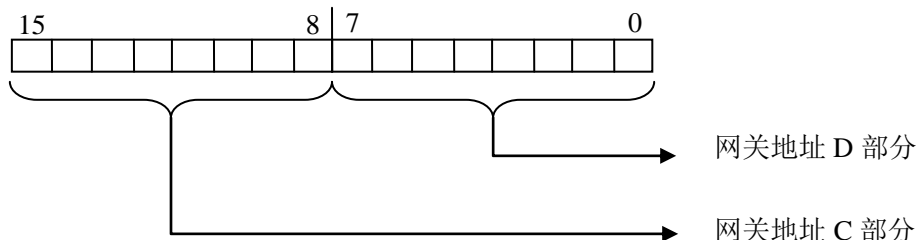
R7713 = 0xFD00 R7714 = 0xFFFE

6、GataWay 地址指定寄存器 R7715, R7716

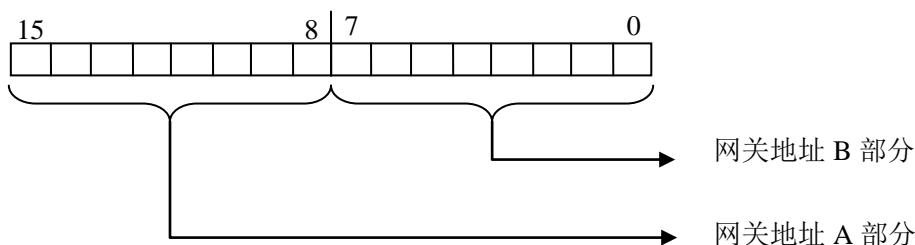
手工设定以太网的网关 (GataWay) 地址。网关地址的 4 段数值转换成 HEX 数, 存放在 R7715, R7716 中。

假设网关地址为 A.B.C.D, 则设置如下 R7715 = CD R7716 = AB

R7715 寄存器存放内容



R7716 寄存器存放内容



例如 网关地址为 192.168.11.9 时, 其数值变换关系如下:

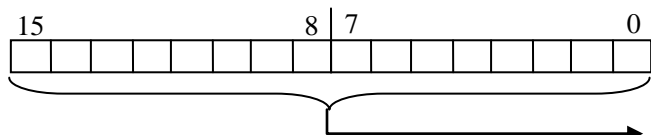
A = 192 (0xC0) B = 168(0xA8) C = 11 (0x0B) D = 9(0x09)

则数值存放如下:

R7715 = 0x0B09 R7716= 0xC0A8

7、以太网通信参数设置完成标志寄存器 R7717

当把所有的以太网通信参数寄存器设置完成后, 在本通信参数设置完成标志寄存器中写入数值 0xA55A 后, NK1 即根据参数寄存器的内容来自动更新以太网通信口设置, 以太网通信口参数更新完成后, 则自动进入通信就绪状态。同时 NK0 系统自动在本通信参数设置完成标志寄存器中写入数值 0x5555。如果设置参数有错误, 则系统把发现错误的参数寄存器号放入本寄存器中。



通信参数设置完成代码 0xA55A。
 参数设置正确, 变成 0x5555。
 参数设置有错误, 发现错误的寄存器号写入。

4-6-4 NK0 系列 PLC 以太网通信端口通信参数设置

与串口一样，你可通过 KPPSoft 软件或用户程序初始设定段来设定 NK0 系列 PLC 的以太网通信端口的通信参数，分别介绍如下：

设置方法一：通过 KPPSoft 软件设置

通过[PLC->PLC 设定->COM 端口设置]菜单,打开 NK0 系列 PLC 的 PORT3 通信端口设置窗口，设置其通信参数如下图所示。



节点 ID: 设置 NK0 系列 PLC 以太网口的节点 ID 号。有些软件，例如 KPPSoft，就可以通过该节点号寻找到 NK0 系列 PLC，并建立通信连接，如右图。

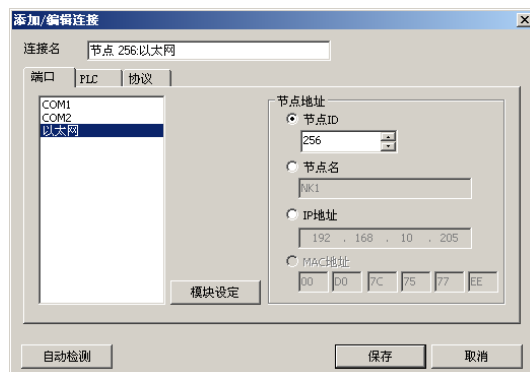
自动获取 IP 地址: 在具有 DHCP 服务器的环境中，可以让 NK0 系列 PLC 自动从该 DHCP 服务器上分配得到一个 IP 地址。

使用此 IP 地址: 由用户手工分配 IP 地址，子网掩码，网关地址。手工设置时，注意 IP 地址不要与其他设备有冲突，另外，子网掩码，网关地址也必须设置正确。

邮箱设置: 用于设置 NK0 系列 PLC 邮件发送功能的一些参数，只在需要用到 NK0 系列 PLC 的邮件发送功能时才需要设置。（具体邮件发送功能通过 EMAIL 指令实现，详情见 4-6-6 节内容。）

以上正确设置好以太网口的各参数后，点[写入 PLC]，则 KPPSoft 把所设置的参数写入 NK0 中，相应的设置参数寄存器内容被修改。同时 KPPSoft 软件会弹出一个[PLC 写入完成]的提示信息窗口，按[确定]键关闭提示信息窗。

你也可以在离线状态下对各通信参数进行设置，设置好后点[写入 PC]，把有关的设置参数存放到磁盘工程文件中，下次与 NK0 建立连接时，可以把有关的参数一次性传送到 NK0 系列 PLC 中去。



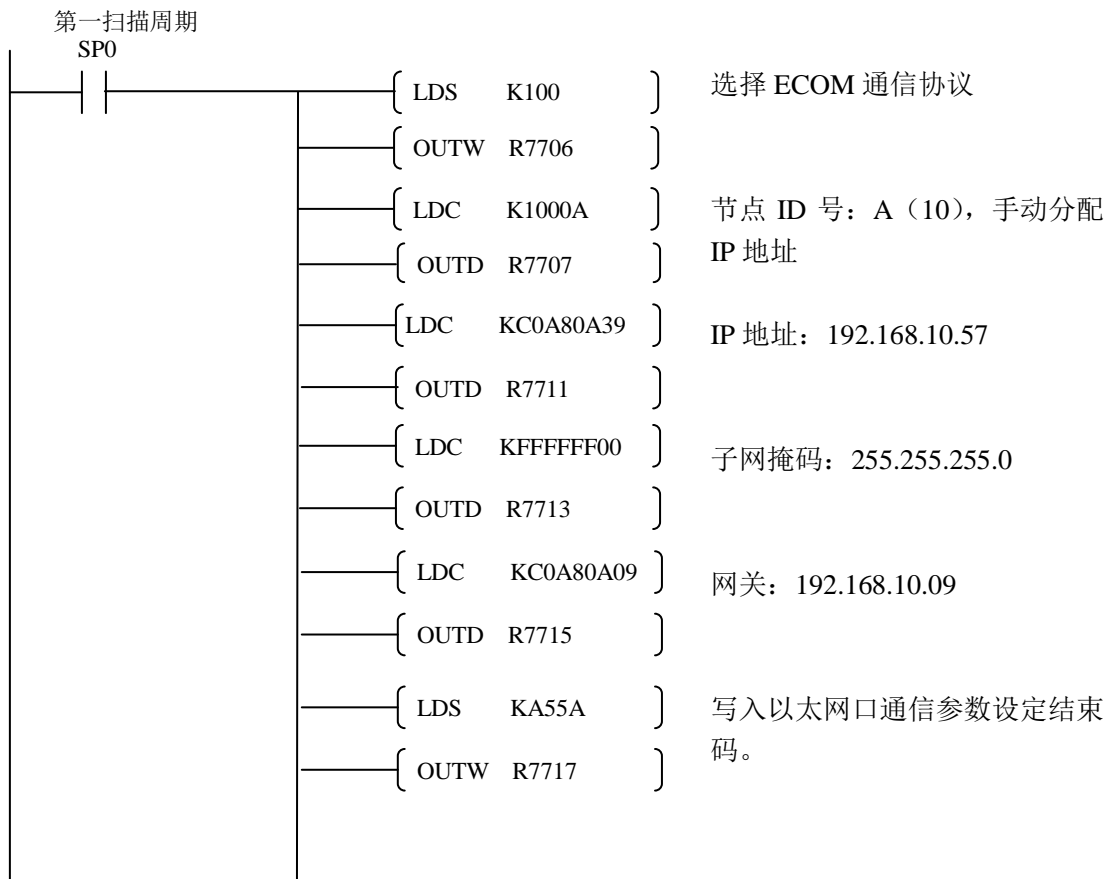
设置方法二：通过用户程序设置

假设 NK1 的以太网通信口做如下设置： 节点 ID 号：10；

手动分配 IP 地址：

IP 地址：192.168.10.57 子网掩码：255.255.255.0 网关：192.168.10.9

则其设置程序段如下：

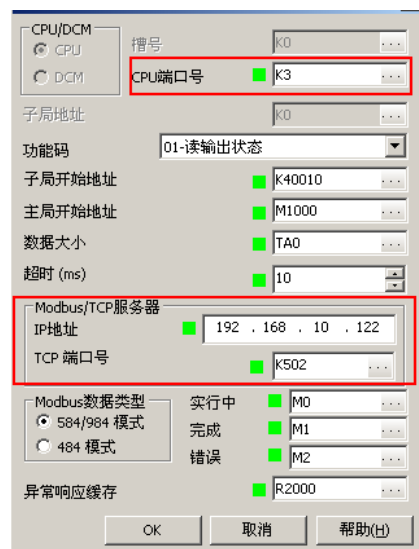


在用户程序中加上以上初始程序段，执行该用户程序后，NK0 就能按指定的 IP 地址参数初始化以太网通信口 P3。

除以上 2 种设置方法外，你也可以使用 KewNetEdit 工具软件来设置 NK0 以太网通信口的基本通信参数。（参见 KewNetEdit 工具软件“基本设置”，其中“模块 ID”，就是上面的“节点 ID”。）

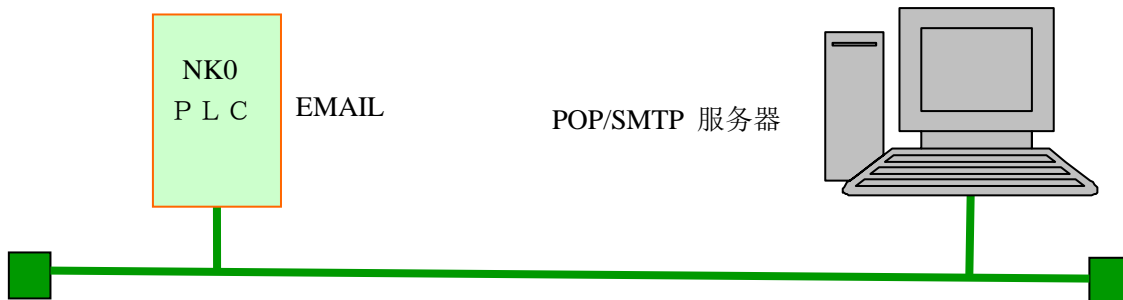
4-6-5 NK0 系列 PLC 以太网通信端口 MODBUS TCP 通信功能的实现

NK0 系列 PLC 上的以太网通信口支持 MODBUS TCP 通信的主、从功能。作为从局通信口，MODBUS TCP 主局可以通过该以太网口直接读写 NK0 的功能存储器内容；作为主局通信口，该以太网口支持通过 MRX/MWX 通信指令来读写与之相连接的 MODBUS TCP 从局的数据。MRX/MWX 指令使用时的参数设置方法与在串口上使用时基本相同，但以下 3 点需要注意：一是通信端口号要选择 P3 口（K3）；二是代替串口时的子局地址，要设置通信对方子局的 IP 地址和 TCP 端口号（一般固定为 502）；三是加入同一 MODBUS TCP 通信网络的各以太网单元的 IP 地址必须保证在同一网段中。



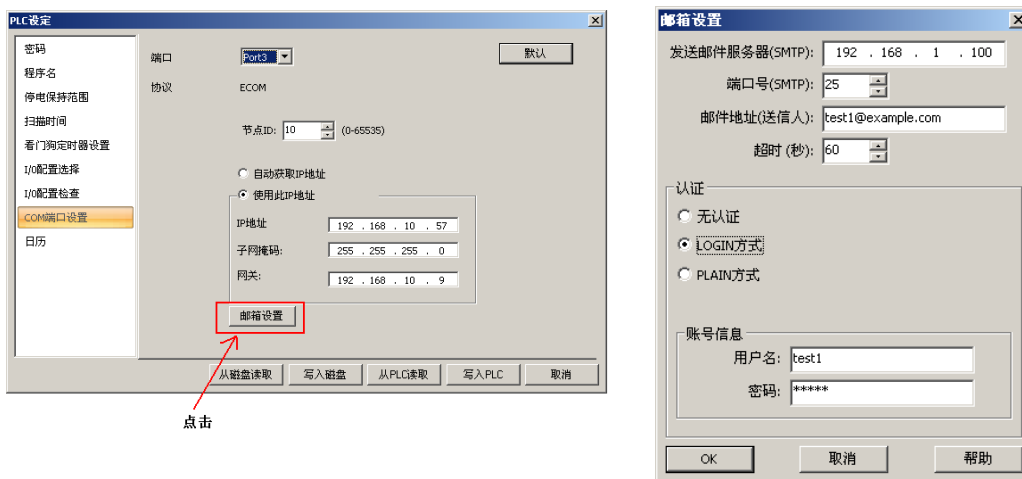
4-6-6 NK0 系列 PLC 以太网通信端口电子邮件发送功能的实现

NK0 系列 PLC 上的以太网通信口，支持 SMTP 邮件发送功能，你可以通过该以太网口，向外边发送电子邮件，电子邮件的发送通过 EMAIL 指令实现。



为了使 NK0 的以太网口可以通过 EMAIL 指令发送电子邮件，首先需要对 NK0 的以太网端口进行一些必要的设置，这些设置除了有关以太网口的节点 ID 号、IP 地址、子网掩码、网关等参数外，还需要设置邮件服务器的一些参数。注意该设置只能通过 KPPSoft 软件来设置，而不能用用户程序段设置，也不能通过 KewNetEdit 工具软件来设置。设置方法如下：

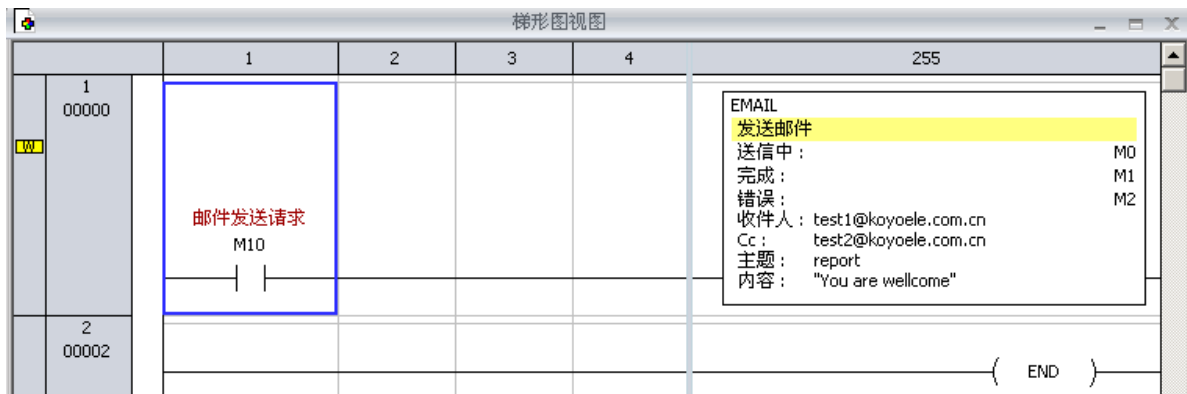
打开 NK0 以太网通信口参数设置窗口，如下页左图，点击其上的[邮箱设置]，会打开邮箱设置窗口，如下右图，该窗口上各设置项内容如下表。



设定内容	含义	初始值
发送邮件服务器 (SMTP)	邮件服务器的 IP 地址	192.168.1.1
端口号 (SMTP)	设置 SMTP 服务器的端口号	25
邮件地址 (送信人)	设置 NK0 的邮箱地址	
超时 (秒)	设置连接邮件服务器的超时时间	
认证	邮件发送认证方式的指定 无认证/LOGIN 认证/PLAIN 认证	无认证
用户名*1	登录 SMTP 服务器时使用的帐户名称	
密码*1	登录 SMTP 服务器时使用的密码	

*1: 用户名、密码在无认证登录方式下无效。

设置好 SMTP 邮件服务器相关内容后，就可以编写邮件发送程序了。



EMAIL 指令中各参数的意义如下：

- **送信中标志：**启动 EMAIL 指令后置为 ON，邮件发送指令执行完成后变为 OFF。
- **完成标志：**启动 EMAIL 指令时复位该标志，EMAIL 指令正常完成后置为 ON 并保持。
- **错误标志：**启动 EMAIL 指令时复位该标志，EMAIL 指令执行出错时置为 ON 并保持。
- **收件人：**指定邮件接收方邮箱地址。
- **CC：**指定邮件抄送方邮箱地址。
- **主题：**指定邮件标题。
- **内容：**要发送邮件的详细内容，注意发送内容要用双引号括起，包括双引号在内的邮件内容不得超过 128 字节。

执行上面的程序段，每当 M10 接通后，就启动一次邮件发送动作。

4-7 NK0 系列 PLC 通信相关特殊线圈的利用

NK0 的 RS-485 通信端口有 2 个特殊线圈与主局通信有关：一个为端口忙标志线圈，一个为端口通信错误标志线圈。每次启动 RS-485 通信端口的主局通信后（执行 RX/WX 通信指令），其对应的端口忙标志线圈会接通；通信结束后，该端口忙标志线圈会被复位。必须等到该端口忙标志线圈复位后才允许启动下一次通信，注意每次通信的时间可能会大于一个扫描周期。如果出现通信错误，其对应的端口通信错误标志线圈会接通。该端口通信错误标志线圈一直会保持接通状态，直到通信指令再次被执行，或程序停止运行后又启动运行，该线圈才被复位。如果你需要在程序中使用该端口通信错误标志线圈来作为某个处理的条件，你必须把该处理指令放在所有通信程序的前面，因为每次主局通信的启动会复位该端口通信错误标志线圈。

一般 RS-485 通信口对应的这 2 个通信特殊线圈不会同时为 ON。

下表列出 NK0 和 RS-485 通信相关特殊线圈。

特殊线圈	功能
SP54	通信参数寄存器设置内容有错时为 ON。
SP110	RS-485 口通信指令执行中 ON
SP111	RS-485 口执行通信指令出错时 ON

利用这几个特殊线圈，可以更灵活合理地配置你的用户程序。

注意：MRX/MWX 通信指令不影响通信口忙和通信口通信出错标志！其指令动作也不受这 2 个标志线圈影响。

4-7-1 通信出错后的后续处理

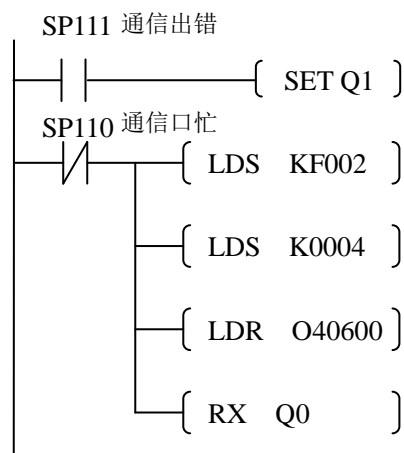
右边的例子程序段说明了在通信程序中这两个特殊线圈的使用方法。程序中 SP110 为通信口忙标志；SP111 为通信口通信出错标志。在通信进行中，SP110 一直保持为 ON，完成一次通信后，SP110 变为 OFF。当 SP110 变为 OFF 后，可以进行下一次通信。

当 NK0 发现通信出错时，会将 SP111 置为 ON，你可以在程序中使用该通信出错线圈进行一些特别的处理（程序中为置位 Q1）。注意当需要在程序中使用该特殊线圈时，必须把对该特殊线圈的处理指令放在所有的通信指令的前面，因为每当执行 RS-485 口的 WX/RX 通信指令时，NK0 会首先复位 SP111。

利用通信出错标志，你可以确认某次通信过程是否正常完成。

一般情况下，一次通信的时间会大于一个扫描周期。

RS-485 通信口要等一次通信结束后才能进行下一次通信。

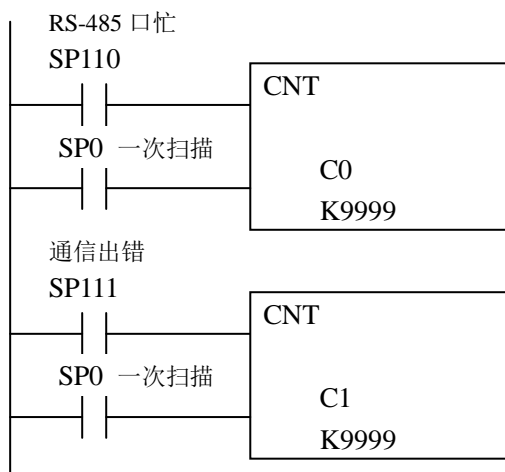


4-7-2 通信出错情况的计次处理

利用右边的程序段可以对 RS-485 通信端口发出通信的次数以及发现通信错误的次数进行统计。每次对 RS-485 口执行 RX 或 WX 指令时，SP110 会接通，计数器 C0 会增加 1。

当 RS-485 口发生通信错误时，SP111 会接通，计数器 C1 会计数加 1。

正常情况下，SP110 会不断计数而 SP111 不会。不过在现场实际环境下，由于干扰等问题，会有偶尔的通信错误发生而使 SP111 计数，你可以通过比较 C0、C1 的值来得到通信错误的比率值。



可能影响通信，发生通信错误的情况如下：

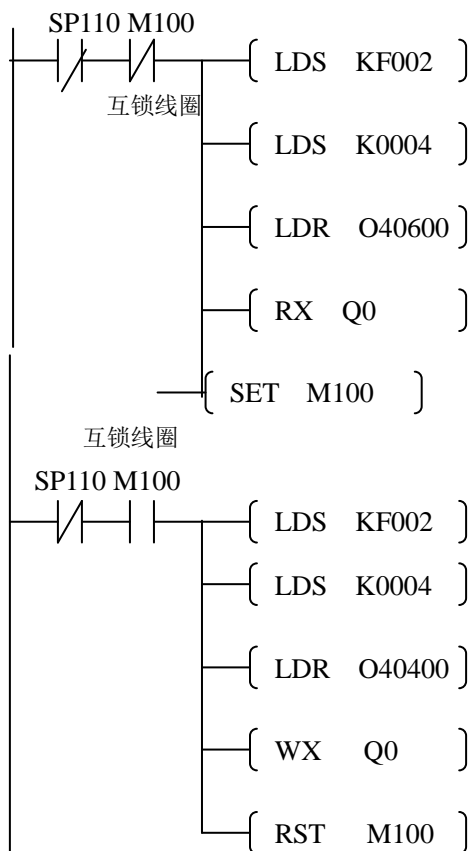
- 1) 子局发出“异常响应”时。
- 2) 接线错误。
- 3) 通信参数设置不正确。
- 4) 所连接的子局号不存在。

4-7-3 多个通信程序段的互锁

在有些情况下，你可能需要在程序中执行多次通信读写指令，而 NK0 的 RS-485 通信口在同一时间只能执行一条通信指令，处理一个通信。为了保证每个通信读写程序段都被执行，必须对各段通信程序采取互锁的方式；否则，NK0 仅能扫描执行排在最前面的通信程序段，而排在后面的通信程序段将永远不会被执行到。

右边的例子中，说明了如何处理这类问题的方法。用 M100 作为 2 段程序的互锁线圈，当执行完 RX 指令后，置位 M100；此后，执行 WX 指令，并复位 M100。

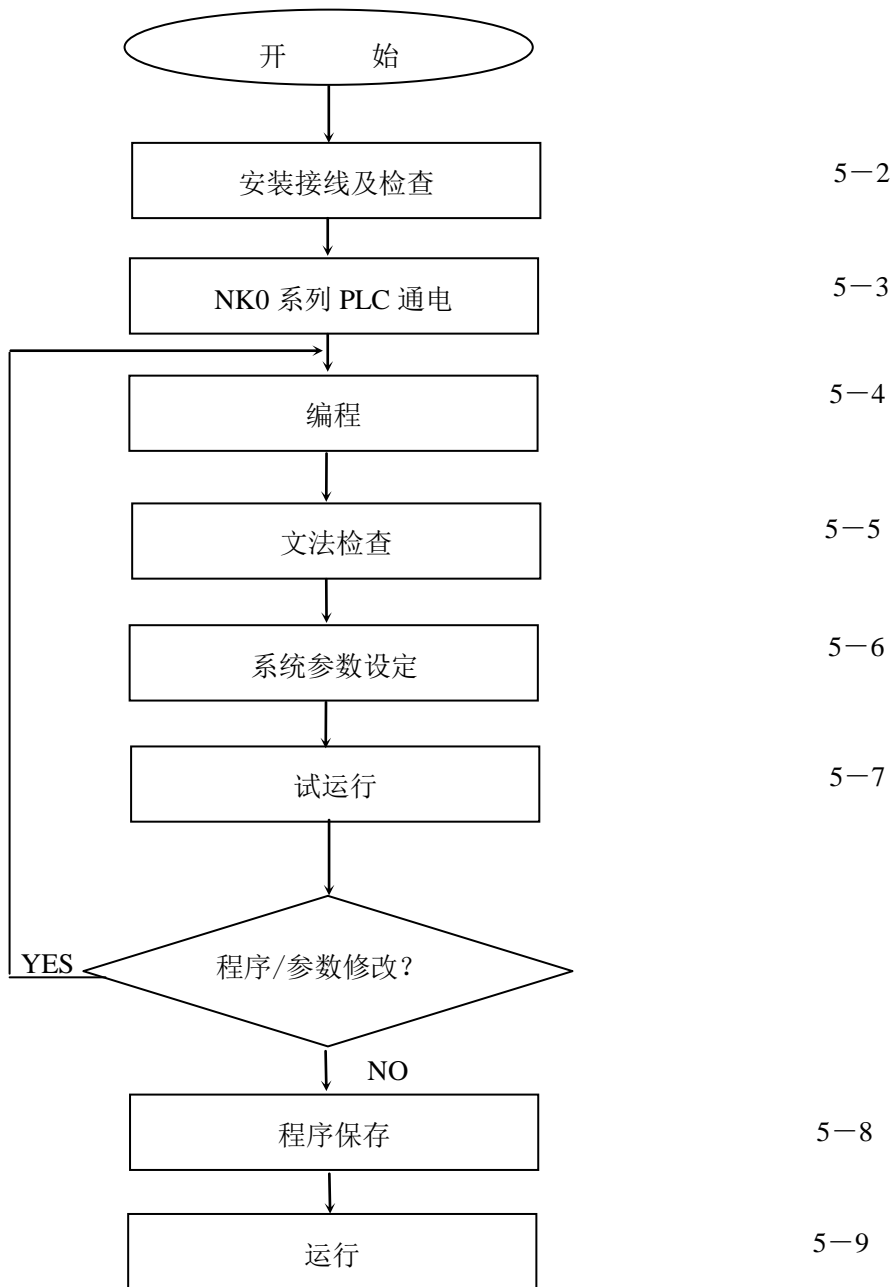
当然，如果你正在使用级式语言编程，就可以把每个通信程序段分别放在不同的级中，并顺次执行这些级。这样就保证某一刻只有一个通信程序的级被执。



第五章 NK0 系列 PLC 运行准备

5-1 运行步骤

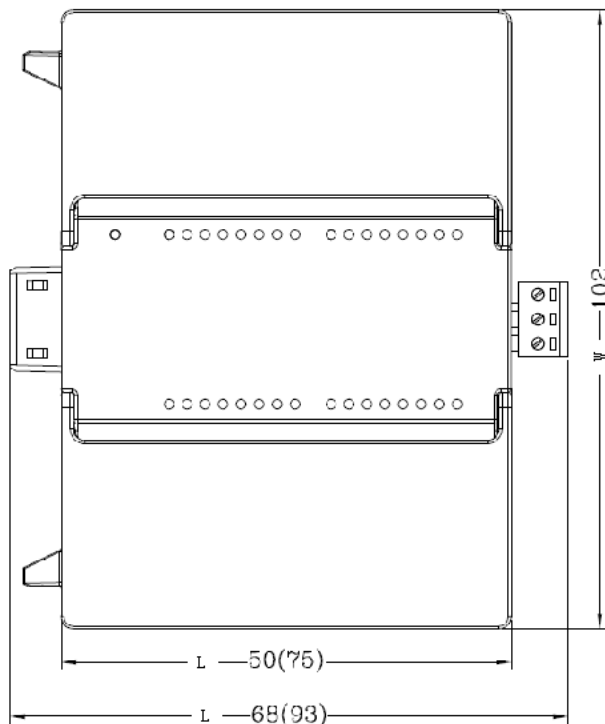
运行框图如下：



5-2 NK0 系列 PLC 安装接线及检查

5-2-1 NK0 系列 PLC 安装尺寸及安装方法

NK0 系列 PLC 可以螺钉安装和导轨安装，不管使用何种方法安装，首先需要了解 NK0 系列 PLC 各单元的长宽高尺寸 L、W、H。NK0 系列 PLC 分 16 点机型和 32 点机型，2 种机型的产品前后宽度、上下高度是一样的，就是左右长度不同如下（详见 1-5 节介绍）。



NK0 系列 PLC 各本体单元的长宽高尺寸如下表所示（单位：mm）

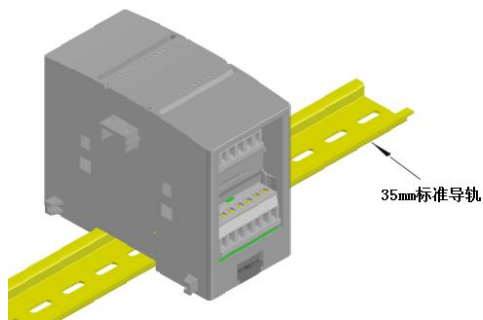
NK0 机型 类别	L	W	H
16 点机型	68	102	83
32 点机型	93	102	83

注：上表尺寸中的长度尺寸包含了左侧凸出和右侧 RS-485 端子的尺寸。

安装方法分述如下：

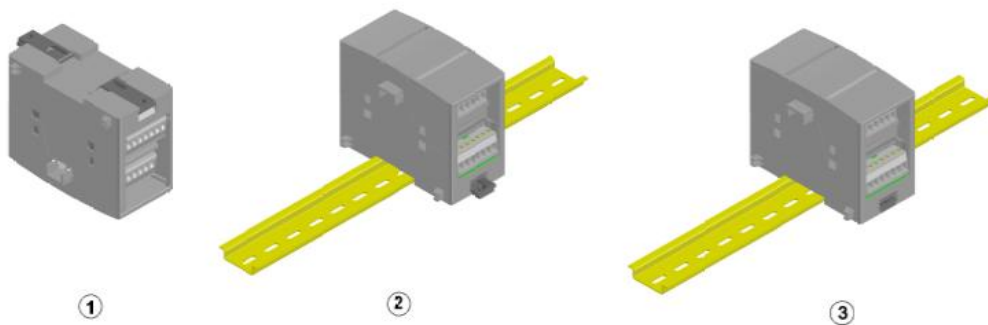
1、导轨安装

NK0 系列 PLC 可安装于导轨上，安装导轨为标准的 35 mm 导轨。



NK0 系列 PLC 导轨安装基本步骤如下：

- ① 先拉开本体后面的 2 个小耳扣；
 - ② 把 NK0 本体挂到导轨上，使 NK0 倾斜一定角度，先把导轨上边沿装入 NK0 背面导轨槽中，再用劲把 NK0 向下压，以便把整个导轨压入 NK0 背面的导轨槽中。
 - ③ 把所有小耳扣压紧到位，即完成导轨安装。
- 安装示意图如下页所示。



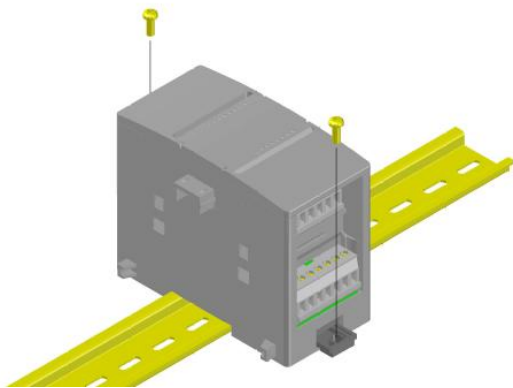
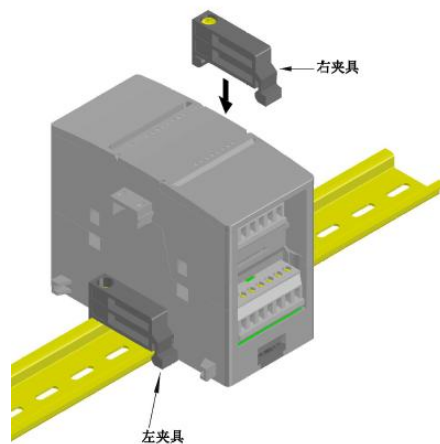
要把 NK0 从导轨上取下来，只要先把 NK0 上的 安装耳扣拉开，后直接取下 NK0 即可。

导轨安装固定夹具的使用

采用导轨安装方法安装 NK0 系列 PLC 时，为了更好的固定 NK0 产品，可以选择在已经安装好的 NK0 产品的两端，安装 35mm 导轨专用固定夹具，其安装示意图如下。

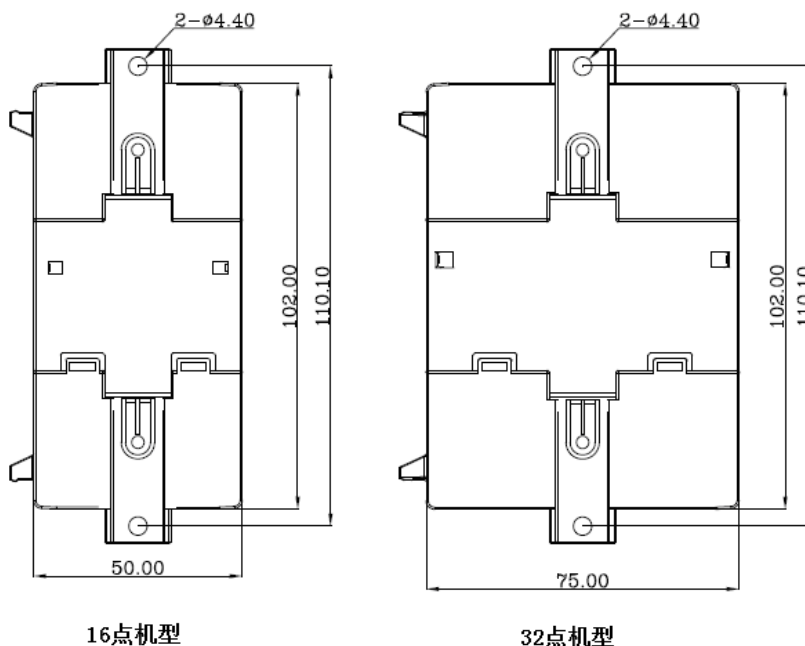
本公司备有 35mm 导轨安装专用固定夹具选件产品，型号：NK1-Fixer，欢迎选购使用！

如果你不使用专用固定夹具，为了固定 NK0 系列 PLC，建议对 NK0 系列 PLC 的安装耳扣，按下面介绍的螺钉安装方式进行螺钉固定安装处理（适用 M4 螺钉）。



2、螺钉安装

NK0 系列 PLC 的安装耳扣上都开有螺钉安装的安装口，安装孔直径 $\phi 4.40\text{mm}$ ，适用 M4 螺钉。
NK0 系列 PLC 安装耳扣上螺钉安装孔位置尺寸示意图如下：



NK0系列PLC螺钉安装孔位置示意图（单位：mm）

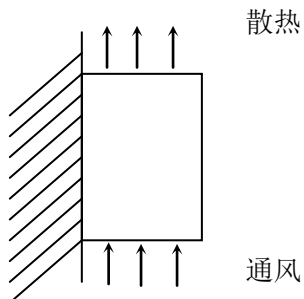
注：上图尺寸中的长度尺寸没有包含左侧凸出和右侧 RS-485 端子的尺寸。

安装时先按照安装孔位置开好安装孔或者攻上螺纹，后把 NK0 系列 PLC 的安装耳扣拉开，把耳扣对准安装孔位置，用螺钉固定即可。

3、安装散热

NK0 系列 PLC 可以水平或垂直安装，安装时请注意保证 PLC 良好的通风，在 NK0 系列 PLC 各单元上下侧面都开有通风散热孔，安装时请保证在 NK0 系列 PLC 的四周都留有足够的空间（最小距离为 50mm），以保证能够有效散热。

安装时检查确认没有连线屑或金属片从散热孔中掉入 PLC 单元里，以免引起短路事故。



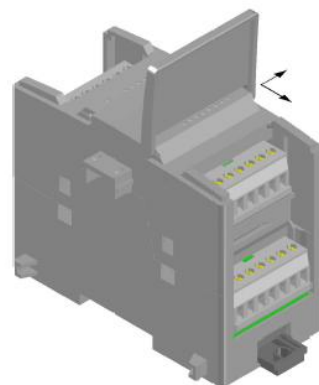
NK0 系列 PLC 安装散热示意图

5-2-2 NK0 系列 PLC 输入/输出接线端子盖板的装卸

1、输入/输出接线端子盖板的拆卸

按右面的示意图卸下 NK0 的接线端子盖板。

- ① 把盖板反转至垂直位置；
- ② 一只手扶住垂直盖板的一侧，另一只手把盖板另一边的下端轻轻往外拉，同时把盖板该端顺着盖板安装位侧面的导引槽往身边拉，就可以卸下该盖板了。
- ③ 用同样方法卸下另一块盖板。



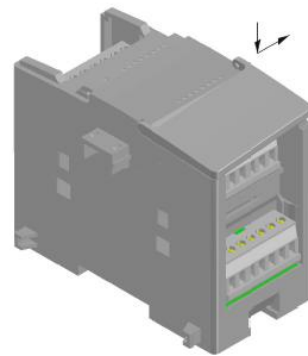
2、输入/输出接线端子盖板的安装

接线端子盖板安装方法：

- ① 把盖板一端安装到位（盖板支脚上的凸起装入前面板上凸起安装位置）；
- ② 把盖板另一端的支脚往外拉开一些；
- ③ 把该端支脚上的凸起顺着前面板盖板安装位侧面的导引槽往前推进，直到凸起滑到安装位置。

以上完成盖板的安装，如右图。

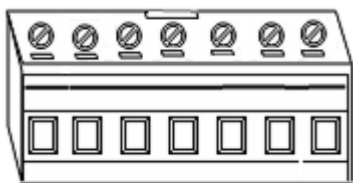
用同样的方法安装好另一块盖板。



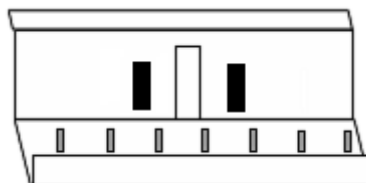
5-2-3 NK0 系列 PLC 输入/输出端子台的装卸

为方便用户的维护、保养工作，NK0 系列 PLC 的输入/输出接线端子设计成成组可拆式 I/O 端子台方式。

NK0 可拆式输入/输出端子台分成不同的组，以方便每组整体拆卸。NK0 输入/输出端子台一般以 8 个端子组成一组，每组端子台的内侧都有一凹槽，用于插入螺丝刀，拆下该端子台；对应于每组端子台，在 NK0 的本体/扩展单元上输入/输出点位置上有端子台插槽，安装时，把端子台安装到对应的槽位上即可。注意：安装端子台时，不同位置的端子台不要搞错。



端子台外形



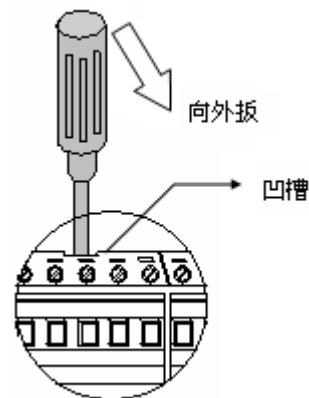
端子台插槽

● 可拆式 I/O 端子台的拆卸方法

每个端子台组的内侧中间都有一个凹槽。拆卸端子台组时将一字螺丝刀紧贴端子台组插入端子台组内侧的凹槽内，注意使用大小合适的螺丝刀，插入深度在 1 厘米左右，以保证螺丝刀头已插入端子台槽中。稍微用力向外扳螺丝刀，待端子台倾斜后，用手将端子台组继续向外扳，即可将端子台组卸下，见右图。

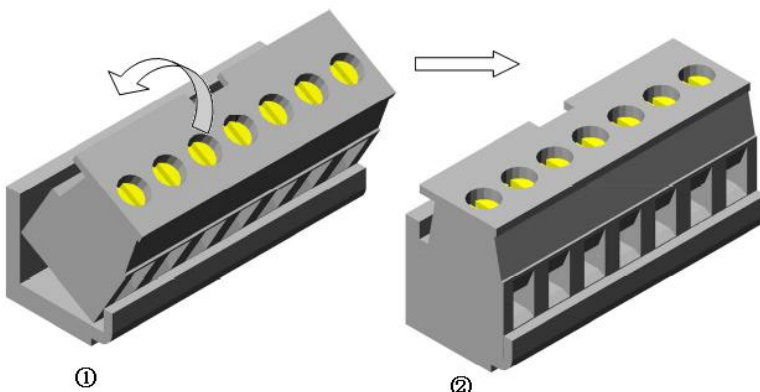
注意如果一次螺丝刀操作后，用手还扳不下端子台组的话，可以把螺丝刀插的更深一点，重复以上的外扳螺丝刀动作，直到端子台组倾斜脱出。

插入螺丝刀时，要注意确保螺丝刀正确插到端子台组和端子台插槽内壁之间，而不要插到端子台插槽内壁的外面去。



● 可拆式 I/O 端子台的安装方法

安装端子台组时，只需将端子台组上的每个插空对准端子台插槽上的插针，把端子台组背面的凸起对准插槽上的小空，端子台组稍微向外倾斜一点插下后，用力向内向下压端子台组，听见“咔哒”声即安装完毕，见右图。

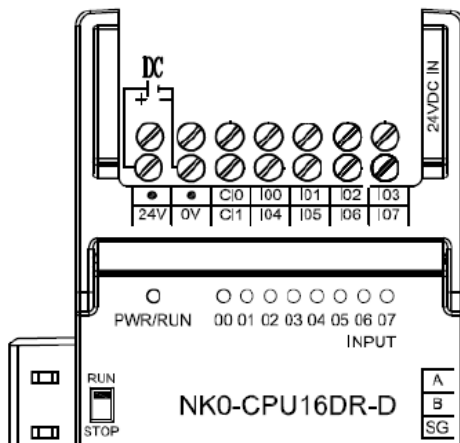


注意：NK0 本体右侧的 3 端子台 RS-485 通信端子没有凹槽，是直接用手插拔的。

5-2-4 NK0 系列 PLC 机器连线

1、电源连线

NK0 系列 PLC 为直流 24V 工作电源，其位置位于上侧前排端子最左面的 2 个，一般标有 24V, 0V 字样，其位置和接线示意图如下：

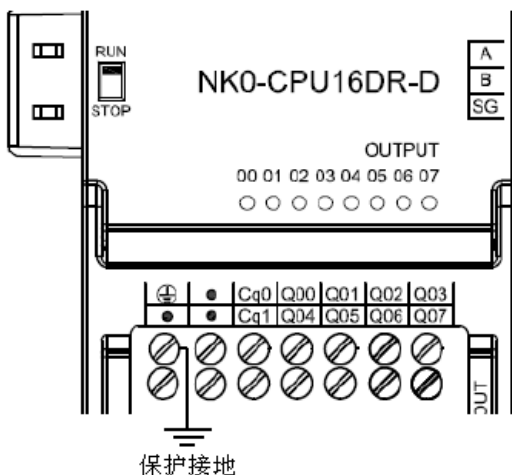


●关于电源干扰

NK0 系列 PLC 本身考虑到在通常工厂环境中的电磁噪声干扰问题，因此 NK0 系列 PLC 并不需要特别的电源噪声防护装置。但是，如果由于在附近有强电机，AC 电磁线圈或其它电感负载而产生很高的电磁噪声干扰时，建议使用直流稳压电源给 NK0 供电，并请在需要时使用静噪滤波器。

2、保护接地的连线

NK0 系列 PLC 上有  端用于保护接地，请在需要时按下图所示方式接地。



3、输入/输出信号线的连线

在连接输入/输出电线时，请选择适当粗细的电线，并注意正、负不要接反；输入/输出线应尽量分开连线；与主回路线和动力线不能分开的情况下，请用正交配线或使用整体屏蔽电缆，屏蔽层在 PLC 端接地；用导管连线时，务请把导管接地。

4、机器连线上的其他注意事项

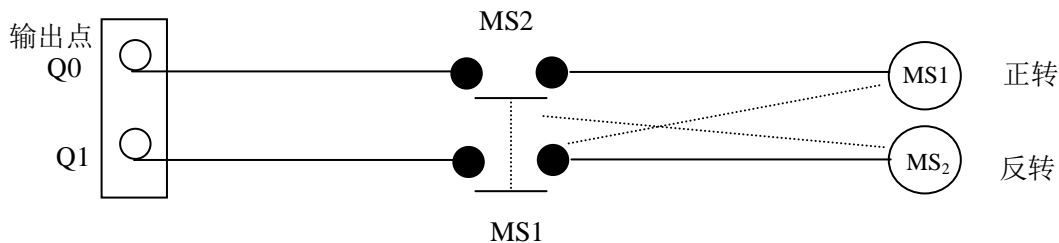
另外在给 NK0 系列 PLC 系统现场连线时，以下几点需要特别注意：

1) 电源系统的连线和紧急停止回路

请将电源部分、动力部分、控制部分、DC 输入/输出部分的电源线分开连线，另外，为使在 PLC 产生故障或异常动作时不造成整个系统的异常动作，请用外部继电器回路构成紧急停止回路。

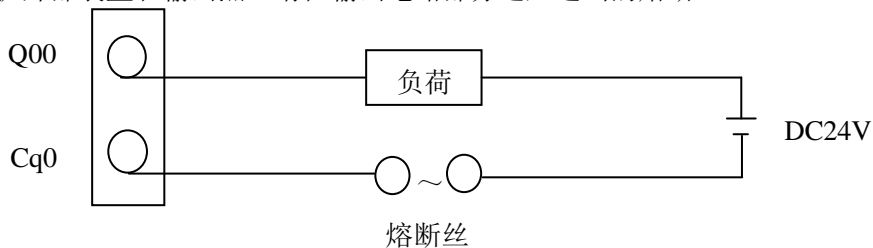
2) 互锁回路

用 PLC 输出控制相反动作或考虑到会由于 PLC 的误动作而产生严重事故或使装置损坏的情况下，请在外部设立互锁回路。



3) 保护熔断丝的接入

为了保护外部装置和输出点，请在输出电路部分连入适当的熔断丝。



5-2-5 NK0 系列 PLC 安装上的注意事项

- (1) 在安装 NK0 系列 PLC 系统前，应检查有无运输中的损坏以及是否符合订货要求。
- (2) 为确保通风和检修的间隙，在安装 NK0 系列 PLC 时，周围请充分保留些空间（50 mm 以上）。
- (3) 请安装于平整的表面上。安装表面有歪、斜、翘等现象时，将产生不必要的附加力，不利于产品的长久工作。
- (4) 请使用必要的配线槽。
- (5) 如果 PLC 安装于面板下，最好能够安装一个通风排气扇散热。
- (6) 请避免在以下环境中使用 NK0 系列 PLC。
 - 环境温度高于 55℃ 或低于 0℃ 的场合；
 - 相对湿度不在 30 ~ 95% 范围内，以及温度急剧变化导致凝结露的场合；
 - 环境中尘埃，铁粉，腐蚀性气体的场合；
 - PLC 本体直接受到振动或撞击的场合；
 - 有直射阳光的场合；
 - 有强电场，强磁场的场合。
- (7) I/O 线尽量分开布线。
 - 输入和输出，电源线之间应相互分开。

5-2-6 NK0 系列 PLC 安装检查

在安装连线时，请检查以下几点：

- (1) 电源系统接线端子和输入输出接线端子的固定情况；
- (2) PLC 本体的固定情况；
- (3) 电源系统和输入输出接线的检查；
- (4) 有无杂物混入的检查。
检查确认有没有连线屑或金属片从散热孔中掉入 PLC 单元中。

5-3 NK0 系列 PLC 通电

- (1) 上电前先请确认电源电压是否正常，并确认电源极性；
- (2) 上电前再次请确认各端子台的连线正确性；
- (3) 合上电源，给 PLC 通电；

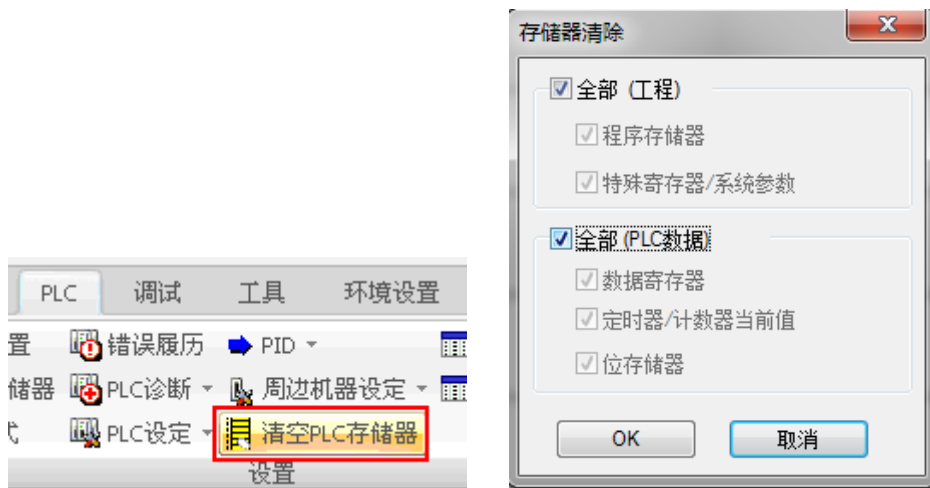
注意：若此时 NK0 系列 PLC 中已有无语法错误的程序存在，则 CPU 有可能会进入 RUN 状态（由运行模式开关位置决定）。为保证上电时系统处于 STOP 状态，请先把运行模式开关打到 STOP 位置（TERM-STOP 模式），然后上电。

- (4) 检查并确认 CPU 上的 PWR（绿色）指示灯点亮；
若 PWR 灯不亮，请立即切断电源，查找异常原因并改正。

5-4 程序编制

NK0 系列 PLC 可以通过计算机编程软件 KPPSoft 编制合适的用户程序。具体编程方法请参阅 KPPSoft 软件在线帮助资料、《KPPSoft 软件使用入门手册》以及 NK0 指令系统（《NK0 编程手册》）等有关资料。

注意：首次编程前，最好对 NK0 系列 PLC 做一下存储器全清处理（“清空 PLC 存储器”）。

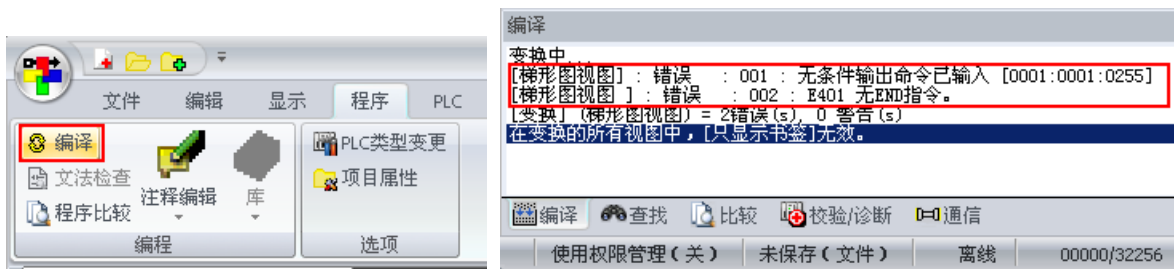


然后如下做一下“系统参数初始化”，以保证 PLC 为标准出厂状态。



5-5 程序文法检查

程序编好后要进行文法检查，文法检查可发现程序上存在的违反规则的部分。NK0 的用户程序文法检查工作是在 KPPSoft 软件上实现，每次在 KPPSoft 软件上通过“程序-->编译”菜单（快捷键 F8）操作对所编辑的程序部分进行编译时，如果发现程序中存在文法错误，KPPSoft 软件会在 KPPSoft 软件下边的“编译信息”窗口中，给出程序中有文法错误的提示信息。



NK0 系列 PLC 的文法检查出错信息分“错误”和“警告”2类。

“错误”类信息表示程序中存在需要改正的文法错误，需要修改程序后重新对程序进行编译，否则程序编译通不过，不能对编辑的用户程序进行存盘或送入 PLC 处理。

“警告”类信息表示程序中存在一些可能会引起程序动作异常的内容，例如输出线圈重复等。针对“警告”类信息，你可以确认一下是否程序输入错误，如果是输入错误，请修改；如果是有意为之，请保留。“警告”类问题并不影响用户程序的编译结果。

下面列出 KPPSoft 软件对 NK0 系列 PLC 程序进行编译时可能发现的主要文法错误和其产生原因以及对策方法：

(1) “错误”类文法检查出错

▲E401 无 END 指令

在主程序的最后写入 END 指令

▲E421 级重复

在 SG 或 ISG 指令中，相同的编号被重复使用。

▲E453 无定时器/计数器

虽然有定时器/计数器的接点指令，但相同编号的定时器/计数器的主体方面（线圈）指令没有被编程，接点方面和主体方面的编号不一致造成差错或者忘了在主体方面进行编程。

▲E455 计数器条件遗漏

在计数器指令中没有附加的接点条件（计数、复位等）或接点条件不全。应在被检查出错的指令前增加相应的条件。

▲E461 堆栈溢出

ANDLD 或者 ORLD 指令连续使用了 9 个以上。

▲E462 堆栈不够

ANDLD 或者 ORLD 指令的数目，超出了该连接的接点组所要求的数目。

▲E463 逻辑错误

以 AND 或 OR 指令开始而没有初始 LD 指令，请插入遗漏的 LD 指令或将出错的指令改写为 LD 指令。

▲E464 未形成回路

在自母线或级开始的回路中，没有用 OUT 类指令或 JMP 指令来结束回路。

(2) “警告”类文法检查出错，主要为一些重复检查出错

▲ E471 线圈重复

相同的线圈定义号被重复使用，由于本 PLC 的梯形图中允许重复使用线圈，因此，需要判别是否有意重复。

▲ E472 定时器重复

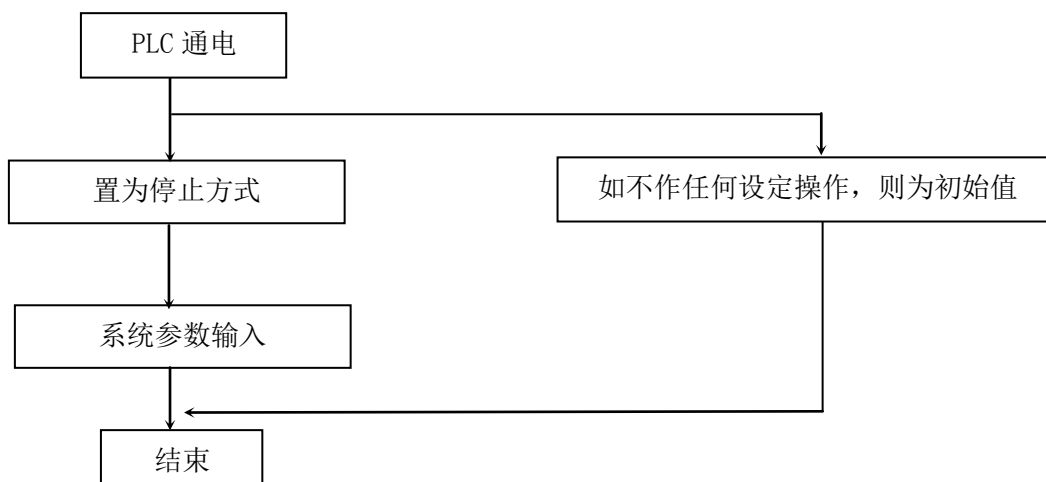
重复使用了相同的定时器定义号，通过检查确认重复地址后，改为正确的定义号。

▲ E473 计数器重复

重复使用了相同的计数器定义号，通过检查确认重复地址后，改为正确的定义号。

5-6 系统参数的设定

在系统运行前，需确认下表所示参数以及一些特殊功能寄存器是否需要设定。通常在未作任何设定时，这些参数有一个初始值。因此在初始值合适时，不需要进行任何设定操作。



注意：必要时在进行系统参数设置前先做一下“系统参数初始化”。

系统参数的初始值和可设定的范围：

项目	初期值	设定值	
用户程序名	未登录	8位英文、数字	
密码	00000000（未登录）	8位数字(BCD数)	
停电保护领域（*）	M	M300 – M377	M0000 – M1777
	R	R2000 – R2027	R2000 – R7777
	T	无	T000 – T177
	C	C000 – C17	C000 – C177
	S	无	S000 – S377
W · DOG 时间	200ms	2 – 9998ms	

（*）说明：

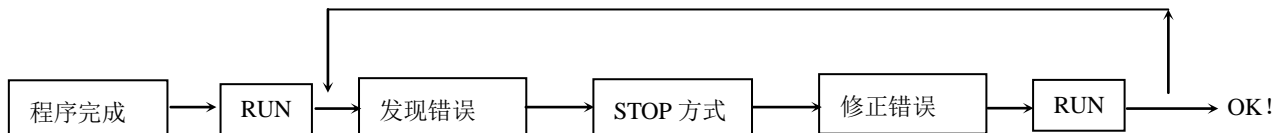
NK0 的停电保持区域无须另配电池，就可以实现停电保持功能。但注意 NK0 最大只支持 64 字（1 字=16bits）内容的停电保持，在设置 NK0 停电保持数据范围时，注意确保不要超过 64 字的范围。如果超过了 64 字的范围，那 NK0 将忽略本次设置内容并给出报警信息（通过 KPPSoft 软件给出）。

5-7 试运行

编制好程序，确认没有文法错误后，就可以把程序送入 NK0 系列 PLC 中，进行试运行了。

试运行的目的是要发现程序的执行逻辑是否符合控制的要求。如果发现问题，就要对程序进行修改，直到满足控制要求。

试运行的基本步骤如下：



KPPSoft 软件具有在线监视和离线模拟运行功能，可以更好地方便客户调试程序。

注意：不论 NK0 系列 PLC 的模式开关处于哪个位置，都可以通过 KPPSoft 改变 NK0 的运行模式！

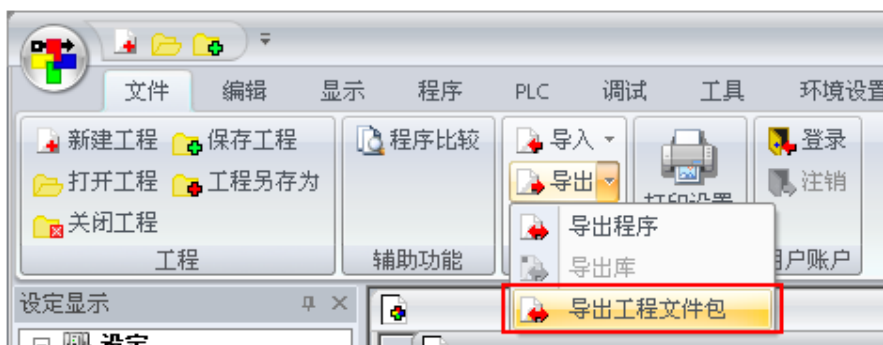
5-8 程序保存

在正式运行程序前，如有必要，应把程序保存起来，以备留档或将来使用。

你可以通过 KPPSoft 程序的“保存工程”菜单来把用户程序保存为计算机磁盘文件（.KPP 文件）。KPPSoft 保存的工程文件为单文件结构，该文件包含用户程序、注释以及从 PLC 中读出的寄存器内容、系统配置信息（如果有）。

你还可以用 KPPSoft 软件的“导出工程文件包”操作，生成一个专门用于工程文件传送的文件包（.KAD 文件），该工程文件包可以通过 KPPSoft 软件传送到 NK0 中。

具体请参考《KPPSoft 软件使用入门手册》资料。



5-9 运行

在调试好用户程序后，就可以正式运行 NK0 上的程序了。正式运行时，请确保 NK0 的运行模式开关在“RUN”位置（如果模式开关在“STOP”位置，NK0 断电再上电后会自动进入停止模式）。

如果模式开关在“RUN”位置而 NK0 处于停止模式，可以通过以下方法使 NK0 运行用户程序：方法一、给 NK0 断电后再上电，NK0 自动进入运行模式；方法二、把模式开关先拨到“STOP”位置，再拨到“RUN”位置，NK0 也会进入运行模式。

第六章 NK0 系列 PLC 系统维护

NK0 系列 PLC 被设计成可以长期不间断地工作，其可靠性非常高，你几乎不需要维护它的运行。

在 PLC 运行过程中，你可以通过 KPPSoft 编程软件来观察整个 PLC 控制系统的状态。另外 NK0 系列 PLC 面板上的各指示灯（各 I/O 状态灯，PWR/RUN 灯）也有助于观察 NK0 系列 PLC 的运行状态和故障部位。

当 NK0 系列 PLC 发生运行故障或运行不正常时，可考虑以下原因：

- (1) 对于 PLC 系统的供给电源的问题
 - 电源没有供给；
 - 电源电压低；
 - 电源瞬时断开；
 - 电源里混有大的干扰。
- (2) 由于事故、差错等原因造成机器损坏
 - 由于叠加了高压（如雷电等）；
 - 由于机械故障引起动力装置的损坏（如阀门、马达等）；
 - 由于机械故障引起检测器件的损坏。
- (3) 控制回路不完备
 - 控制回路（PLC 程序等）和机械不同步；
 - 控制回路出现意外的情况。
- (4) 机械的老化、损耗
 - 接触不良（限位开关、继电器、电磁阀等）；
 - 后备电池不正常。
 - 强干扰造成 PLC 环境的恶化。
- (5) 由强干扰或误操作导致程序异常改变
 - 违背操作规定使程序发生改变；
 - 电源合上时更换存储器芯片；
 - 强电气干扰改变了程序。

注意：

- (1)：当由于 PLC 本身的原因引起严重故障时，请不要自己拆开 PLC 处理！请把问题产品送到本公司进行检查维修处理。

6-1 硬件系统的维护

维护标准：本产品的维护并没有什么特别的规则，然而，大约每隔一、二个月，建议对你的 PLC 和控制系统进行例行检查和维护，主要要包括以下几项：

- 空气温度——检查控制柜内空气温度，不要超出任何元件的操作温度范围。
- 空气过滤器——如果控制柜有空气过滤器，要定期清洗或更换。
- 保险丝或断路器——检验保险丝和断路器应完好。

清洁单元——检查所有的空气通风口是否干净。如果外箱体需要清洗，不要接入工作电源，用一块布仔细擦洗箱体。注意不要使水通过通风口进入箱体，不要用强清洁剂，因为这样可能损坏箱体油漆。

6-2 NK0 系列 PLC 状态 LED 显示

NK0 系列 PLC 前面板上有一个 PWR/RUN 双色 LED 状态指示灯，该组指示灯的显示状态可以帮助你初步判断系统出现的问题。一般在通常上电非运行模式下红色 PWR 灯亮（STOP 模式），在用户程序运行模式下，绿色 RUN 灯亮（RUN 模式），如果该组灯显示有异常，可能的问题如下：

指示状态		可能的问题
PWR（红）	RUN（绿）	
灭	灭	未上电/或者电源电压低
闪	亮	RUN 模式，系统有警告性错误时（显示效果：橙闪）
闪	灭	STOP 模式，系统有严重错误时
灭	闪	系统固件升级中

6-3 通信问题

如果你不能通过 RS-485 通信口和 NK0 建立通信连接，检查以下几项：

- 电缆没有连接。
- 电缆断线或连接不正确。
- 被连接的装置设定了不正确的波特率。
- 被连接的装置的端口不正确。
- 连接双方的数据位、停止位、奇偶校验位等通信参数设置不一致。
- 两个装置的接地不同。
- 电气干扰引起通信断断续续的错误。
- PLC 通信端口坏，这时 NK0 需要送修。

在计算机上使用 KPPSOFT 软件时的通信问题，请参考 KPPSOFT 软件在线帮助资料。该手册包括通信端口设置、地址或中断冲突等的诊断和问题的解决方法。

6-4 I/O 问题的解决

NK0 系列 PLC 的 I/O 错误的原因可能有以下几个方面引起：

- 在你的机器或面板上的保险丝熔断（NK0 无内部 I/O 保险丝）。
- 输入/输出接点的接线端子松动。
- NK0 内部输入/输出电路有问题。

阅读下面的内容，将有助于你理解产生 I/O 问题的可能原因以及可能的快速解决 NK0 系列 PLC I/O 问题的一些方法。

- 输出电路不能短路，输出点不能开路。如果你怀疑一个或更多的输出点有问题，要检测从公共点到怀疑点间的电压。当使用数字电压表时，必须注意可能从诸如可控硅管或晶体管类输出装置产生的漏电流。
- PLC 上的 I/O 状态指示灯是表示逻辑端的状态的。对于输入，其状态灯亮表示输入点正常，请确认当断开输入点连线时，其对应的状态指示灯是否熄灭；对于输出，其状态灯亮，仅表示内部输出信号正常，并不表示输出点一定正常。
- 当连接现场设备到 NK0 系列 PLC I/O 点时，漏电流可能是问题源。当某一输出设备所产生的漏电流大得足以使所连接的输入设备导通时，则会产生误输入信号。为避免这种情况，可在输入或输出上并上一个电阻。阻值的大小根据漏电流以及当时的电压而定。通常情况下采用 $10K\sim 20K\Omega$ 的电阻。请确认电阻的功率符合你的应用要求。
- 由于 NK0 的接线端子设计成多个端子整体可插拔的，因此，在你有备用的 NK0 时，可采用整体更换 PLC 的办法来判断解决问题。但是，当你认为某个现场设备有问题时，你必须首先检查现场设备的情况。否则，即使你换了 PLC，还是会产生同样的问题。

6-5 电气干扰问题

干扰是最难处理的问题之一，电气干扰可以以多种方式进入系统，它们主要分成 2 大类：传导型和放射型。很难判断电气干扰是如何进入系统的，但对这 2 种干扰的纠正措施是相似的。

- 传导型干扰是通过连接电线，操作面板等电气连接传入系统的，例如：I/O 连接回路、工作电源线、通信接地线、底板接地线等等。
- 放射型干扰是一种类似电波的干扰，无须通过电气连接就能进入系统。

电气干扰不能完全根除，但可以减小到不影响系统的程度。遵循以下几点可有效减少电气干扰。

- 大多数干扰是由于不正确的接地系统而引起的。一个好的接地系统可有效减小干扰问题，要确保所有接地线为单点接地，并且相互间不形成菊花链。
- 电气干扰可通过 PLC 或 I/O 回路的供电部分侵入系统。DC 电源必须为有良好接地系统的高质量电源系统。
- 接线时注意尽量把输入连线和输出连线隔离，低压信号线决不与高压信号线混布。

6-6 NK0 系列 PLC 错误代码一览表

NK0 系列 PLC 可能会产生的错误主要包括 2 类：一类是 NK0 系列 PLC 本身在运行中可能会产生的错误；另一类是在利用 KPPSoft 软件编制用户程序时可能会产生的程序语法错误。下面分别说明该 2 类错误的错误代码以及产生错误的原因说明。

1、 NK0 系列 PLC 使用中可能产生的错误

错误代码	原因说明
E003	内置看门狗时间溢出，SP51 为 ON。
E155	存储器检查错误，SP44 为 ON
E401	没有 END 指令（NK0 认为没有用户程序。），SP31 为 ON
E313	接收的地址出错，SP46 为 ON。
E316	通信模式出错，SP46 为 ON。

2、 使用 KPPSOFT 软件编制程序时可能产生的程序语法错误

错误代码	原因说明
E4××	没有用户程序
E401	缺少 END 指令
E402	未写入对应于 GOTO、CALL 指令的 GLBL、CLBL 指令
E403	在用 CLBL 指令调用的子程序中，未写 CEND 指令
E404	缺少 NEXT 指令对应的 FOR 指令
E405	缺少 FOR 指令对应的 NEXT 指令
E411	级超过允许数以上
E412	GLBL/CLBL/DLBL 等标记指令超过允许数以上
E413	FOR、NEXT 指令超过 64 条
E421	重复定义了 SG 或 ISG
E422	在同类标记指令中，重复使用同一定义号
E423	FOR/NEXT 循环指令中嵌套使用了 FOR/NEXT 循环指令
E431	在子程序或中断服务程序中使用了 SG，ISG 指令
E432	对应于 GOTO 指令的 GLBL 指令写在了不允许的程序范围内
E433	CLBL 写在主程序中
E434	RET 命令位置错误
E435	在主程序或者中断程序中写有 CEND 指令

续

错误代码	原因说明
E440	DLBL 指令有逻辑条件存在
E441	数据定义块中有 ACON 或 NCON 以外的指令
E451	MLS 命令的母线号没从小到大的顺序
E452	对分配的给实装输入的功能存储器 I 编了输出指令
E453	没有对应于定时器/计数器接点的输出动作命令
E454	ATMR 和 AHTMR 的条件不是 2 个（计时条件、复位条件）
E455	计数器条件不足（CNT 命令需要 2 个，UDCNT 需要 3 个）
E456	SR 命令的条件不足 3 个（数据、时钟、复位条件）
E461	使用条件级联用栈超过了 9 级
E462	使用的 ANDLD、ORLD 命令多于条件级联数
E463	从母线开始的接点指令使用了 LD 关系以外的指令
E464	存在非连接回路
E471	对同一继电器线圈输出指令重复
E472	对同一定时器输出指令重复
E473	对同一计数器, 输出指令重复
E480	在子程序中使用了 CV 指令
E481	CV 指令序列间存在 CV 以外的指令
E482	CV 指令连续使用 17 个以上
E483	在子程序中使用了 CVJMP 指令
E484	在 CVJMP 指令前没有 CV 指令
E485	从 CV 命令开始至 SG、ISG、BSTART、BEND、END 指令间没有 CVJMP 指令存在
E486	在子程序中使用了 BREQ 指令
E487	BREQ 指令没有对应的 BSTART 指令
E488	在 BSTART 和 BEND 间重复使用了 BSTART 指令
E489	BSTART 使用了同一个定义号
E490	紧接在 BSTART 指令后的命令不是 SG 指令
E491	在 BSTART 指令和 BEND 间使用了 ISG 指令
E492	没有对应于 BEND 的 BSTART 指令
E493	紧接在 BEND 指令后的指令非 CV、SG、ISG、BSTART、END
E494	BSTART 指令 END 指令之间没有 BEND 指令

附录一 NK0 系列 PLC 指令语一览表

NK0 有丰富的指令可以用于编制用户程序。NK0 共有 298 种指令语，这些指令语按功能可分为多个大类，下面列出所有 NK0 支持的指令语。

1. 顺序指令（63 种）

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
接点指令	逻辑运算开始 NO 接点	LD		1	GI,GQ,I,Q,M,T,C,S,SP	
	逻辑运算开始 NC 接点	LDN		1	GI,GQ,I,Q,M,T,C,S,SP	
	逻辑与运算 NO 接点	AND		1	GI,GQ,I,Q,M,T,C,S,SP	
	逻辑与运算 NC 接点	ANDN		1	GI,GQ,I,Q,M,T,C,S,SP	
	逻辑或运算 NO 接点	OR		1	GI,GQ,I,Q,M,T,C,S,SP	
	逻辑或运算 NC 接点	ORN		1	GI,GQ,I,Q,M,T,C,S,SP	
直接输入接点指令	逻辑运算开始 NO 接点	LDDI		1	I	
	逻辑运算开始 NC 接点	LDNDI		1	I	
	逻辑与运算 NO 接点	ANDDI		1	I	
	逻辑与运算 NC 接点	ANDNDI		1	I	
	逻辑或运算 NO 接点	ORDI		1	I	
	逻辑或运算 NC 接点	ORNDI		1	I	
比较一致接点指令	逻辑运算开始比较一致接点	LDEQ		2	R,P [K,R,P]	*1
	逻辑运算开始不一致接点	LDNEQ		2	R,P [K,R,P]	*1
	逻辑与运算比较一致接点	ANDEQ		2	R,P [K,R,P]	*1
	逻辑与运算比较不一致接点	ANDEQ		2	R,P [K,R,P]	*1
	逻辑或运算比较一致接点	OREQ		2	R,P [K,R,P]	*1
	逻辑或运算比较不直接点	ORNEQ		2	R,P [K,R,P]	*1
比较一致大接点指令	逻辑运算开始一致大 NO 接点	LDGE		2	R,P [K,R,P]	*1
	逻辑运算开始一致大 NC 接点	LDNGE		2	R,P [K,R,P]	*1
	逻辑与运算一致大 NO 接点	ANDGE		2	R,P [K,R,P]	*1
	逻辑与运算一致大 NC 接点	ANDNGE		2	R,P [K,R,P]	*1
	逻辑或运算一致大 NO 接点	ORGE		2	R,P [K,R,P]	*1
	逻辑或运算一致大 NC 接点	ORNGE		2	R,P [K,R,P]	*1

*1 []: 第 2 操作数可使用的功能存储器., K:0~FFFF.

*2 直接输入指令仅对本体单元输入点有效。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
微分接点指令	逻辑运算开始上升沿接点	LDPD		1	GI,GQ,I,Q,M,T,C,S	
	逻辑运算开始下降沿接点	LDND		1	GI,GQ,I,Q,M,T,C,S	
	逻辑与运算上升沿接点	ANDPD		1	GI,GQ,I,Q,M,T,C,S	
	逻辑与运算下降沿接点	ANDND		1	GI,GQ,I,Q,M,T,C,S	
	逻辑或运算上升沿接点	ORPD		1	GI,GQ,I,Q,M,T,C,S	
	逻辑或运算下降沿接点	ORND		1	GI,GQ,I,Q,M,T,C,S	
寄存器操作指令	逻辑运算开始 NO 接点	BLD		2	R,P [K](BCD0 ~ 15)	*1
	逻辑运算开始 NC 接点	BLDN		2	R,P [K] (BCD0 ~ 15)	*1
	逻辑与运算开始 NO 接点	BAND		2	R,P [K] (BCD0 ~ 15)	*1
	逻辑与运算开始 NC 接点	BANDN		2	R,P [K] (BCD0 ~ 15)	*1
	逻辑或运算开始 NO 接点	BOR		2	R,P [K] (BCD0 ~ 15)	*1
	逻辑或运算开始 NC 接点	BORN		2	R,P [K] (BCD0 ~ 15)	*1
	BIT 接通	BOUT		2	R,P [K] (BCD0 ~ 15)	*1
	BIT 置位	BSET		2	R,P [K] (BCD0 ~ 15)	*1
	BIT 复位	BRST		2	R,P [K] (BCD0 ~ 15)	*1
逻辑组连接	逻辑组串联	ANDLD		1		
	逻辑组并联	ORLD		1		
输出指令	线圈 ON 动作(或)	OUT		1	GI,GQ,I,Q,M	
	线圈 ON 动作(后优先)	ZOUT		1	GI,GQ,I,Q,M	
	线圈 ON 动作(后优先)	ZOUTH		1	I,Q,M	
	取反输出	NOT		1		
	线圈 ON 保持动作	SET		1 (2)	GI,GQ,I,Q,M,S	
	线圈 OFF 保持动作	RST		1 (2)	GI,GQ,I,Q,M,S	
直接输出指令	线圈 ON 动作(或)	OUTDI		1	Q	
	线圈 ON 动作(后优先)	ZDI		1	Q	
	线圈 ON 保持动作	SETDI		1 (2)	Q	
	线圈 OFF 保持动作	RSTD		1 (2)	Q	

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
母线指令	新母线登记	MLS	—{MLS }—	1	K	
	母线复归	MLR	—{MLR }—	1	K	
微分输出指令	1 次扫描输出指令	PD	—{PD }—	1	GI,GQ,I,Q,M	
移位寄存器指令	移位寄存器	SR	—{SR }— —{CLOCK }— —{RESET }—	2	M,[M]	*1
定时器指令	0.1 秒定时器	TMR	—{TMR }—	2 (3)	T [K, R, P]	*1
	0.01 秒定时器	HTMR	—{HTMR }—	2 (3)	T [K, R, P]	*1
	0.1 秒累积计时器	ATMR	—{ATMR }— —{RESET }—	2 (3)	T [K, R, P]	*1
	0.01 秒累积计时器	AHTMR	—{AHTMR }— —{RESET }—	2 (3)	T [K, R, P]	*1
计数器指令	计数器（带复位端）	CNT	—{CNT }— —{RESET }—	2 (3)	C [K, R, P]	*1
	计数器（复位另设）	GCNT	—{GCNT }—	2 (3)	C [K, R, P]	*1
	加减计数器	UDCNT	—{UDCNT }— —{DOWN }— —{RESET }—	2 (3)	C [K, R, P]	*1
	定时器·计数器复位	RSTTC	—{RSTTC }—	1 (2)	T,C [T,C]	*1

*1 []: 第 2 操作数可使用的功能存储器。

2. 双字比较顺序指令（12种）

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
双字比较一致接点指令	逻辑运算开始比较一致接点	DLDEQ		2	R,P [K,R,P]	*1
	逻辑运算开始不一致接点	DLDNEQ		2	R,P [K,R,P]	*1
	逻辑与运算比较一致接点	DANDEQ		2	R,P [K,R,P]	*1
	逻辑与运算比较不一致接点	DANDEQ		2	R,P [K,R,P]	*1
	逻辑或运算比较一致接点	DOREQ		2	R,P [K,R,P]	*1
	逻辑或运算比较不直接点	DORNEQ		2	R,P [K,R,P]	*1
双字比较一致大接点指令	逻辑运算开始一致大 NO 接点	DLDGE		2	R,P [K,R,P]	*1
	逻辑运算开始一致大 NC 接点	DLDNGE		2	R,P [K,R,P]	*1
	逻辑与运算一致大 NO 接点	DANDGE		2	R,P [K,R,P]	*1
	逻辑与运算一致大 NC 接点	DANDNGE		2	R,P [K,R,P]	*1
	逻辑或运算一致大 NO 接点	DORGE		2	R,P [K,R,P]	*1
	逻辑或运算一致大 NC 接点	DORNGE		2	R,P [K,R,P]	*1

*1[]: 第2操作数可使用的功能存储器；R/P 连续2寄存器；K0~FFFFFF。

3. 控制程序执行指令（21 种）

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
级式指令	级登记	SG		2	S	
	初始级登记	ISG		2	S	
	条件成立时级转移	JMP	—{ JMP }—	1	S	
	条件不成立时级转移	NJMP	—{ NJMP }—	1	S	
	合流级登记	CV	—{ CV }—	1	S	
	合流级转移	CVJMP	—{ CVJMP }—	1	S	
	级组请求	BREQ	—{ BREQ }—	1	M	
	级组开始	BSTART		2	M	
	级组结束	BEND	—{ BEND }—	1		
跳转	跳转	GOTO	—{ GOTO }—	2	K(1-FFFF)	
	跳转标号	GLBL		2	K(1-FFFF)	
循环	循环起点	FOR	—{ FOR }—	1	K,R(1-9999)	
	循环返回点	NEXT	—{ NEXT }—	1		
子程序	子程序调用	CAL	—{ CAL }—	2	K(1-FFFF)	
	子程序标号	CLBL		2	K(1-FFFF)	
	返回	RET	—{ RET }—	1		
	无条件返回	CEND	—{ CEND }—	1		
其他						
	监控定时器复位	WDOGR	—{ WDOGR }—	1		
	扫描停止	STOP	—{ STOP }—	1		
	无功能	NOP		1		
	主程序结束	END	—{ END }—	1		

4. 数据处理指令（159 种）

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
读入指令	16 位读入	LDW	—{LDW }—	1	R,P	
	32 位读入	LDD	—{LDD }—	1	R,P	
	任意位长读入	LDF	—{LDF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	直接 16 位读入 *2	LDDW	—{LDDW }—	1	R(I: 输入区域)	
	直接任意位读入(1~32) *2	LDDF	—{LDDF }—	2	I,[K](位长)	*1
	索引 16Bit 读入	LDIX	—{LDIX }—	2	R,P	
	数据堆栈弹出	POP	—{POP }—	1		
	4 位常数读入(十进制,十六进制)	LDS	—{LDS }—	1	K	
	8 位常数读入(十进制,十六进制)	LDC	—{LDC }—	2	K	
	浮动小数读入(32 位)	RLDD	—{RLDD }—	3	R,P	
	浮动小数点常数读入	RLDC	—{RLDC }—	3	F	
	8 进制常数读入	LDR	—{LDR }—	1	O	
写入指令	16 位写入	OUTW	—{OUTW }—	1	R,P	
	32 位写入	OUTD	—{OUTD }—	1	R,P	
	任意位长写入	OUTF	—{OUTF }—	2	GI,GQ,I,Q,M [K] (位长)	*1
	直接 16 位写入 *2	OUTDW	—{OUTDW }—	1	R(Q: 输出区域)	
	直接任意位长写入 *2	OUTDF	—{OUTDF }—	2	Q, [K](位长)	*1
	高 8 位写入	OUTM	—{OUTM }—	2	R	
	低 8 位写入	OUTL	—{OUTL }—	2	R	
	索引 16Bit 写入	OUTIX	—{OUTIX }—	1	R,P	
BCD 加算 命令	4 位 BCD	ADD	—{ADD }—	1	R,P	*1
	8 位 BCD	ADDD	—{ADDD }—	1	R,P	*1
	任意位长(1~32bit)	ADDF	—{ADDF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SADD	—{SADD }—	1		
	8 位常数	ADDC	—{ADDC }—	2	K	

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
BCD 减算 指令	4 位 BCD	SUB	—{SUB }—	1	R,P	*1
	8 位 BCD	SUBD	—{SUBD }—	1	R,P	*1
	任意位长(1~32bit)	SUBF	—{SUBF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SSUB	—{SSUB }—	1		
	8 位常数	SUBC	—{SUBC }—	2	K	
BCD 乘算 指令	4 位 BCD	MUL	—{MUL }—	1	R,P	*1
	8 位 BCD	MULD	—{MULD }—	2	R,P	
	任意位长(1~16bit)	MULF	—{MULF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	
	堆栈	SMUL	—{SMUL }—	1		
	4 位常数	MULS	—{MULS }—	1	K	
BCD 除算 指令	4 位 BCD	DIV	—{DIV }—	1	R,P	*1
	8 位 BCD	DIVD	—{DIVD }—	2	R,P	
	任意位长(1~16bit)	DIVF	—{DIVF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	
	堆栈	SDIV	—{SDIV }—	1		
	4 位常数	DIVS	—{DIVS }—	1	K	
BIN 加算 指令	16BIT	BADD	—{BADD }—	1	R,P	
	32BIT	BADDD	—{BADDD }—	1	R,P	
	堆栈	SBADD	—{SBADD }—	1		
	4 位常数	BADDS	—{BADDS }—	1	K	
	8 位常数	BADDC	—{BADDC }—	1	K	
BIN 减算 指令	16BIT	BSUB	—{BSUB }—	1	R,P	
	32BIT	BSUBD	—{BSUBD }—	1	R,P	
	堆栈	SBSUB	—{SBSUB }—	1		
	4 位常数	BSUBS	—{BSUBS }—	1	K	
	8 位常数	BSUBC	—{BSUBC }—	1	K	

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
BIN 乘算 指令	16BIT	BMUL	—{BMUL }—	1	R,P	
	堆栈	SBMUL	—{SBMUL }—	1		
	4 位常数	BMULS	—{BMULS }—	1	K	
BIN 除算 指令	16BIT	BDIV	—{BDIV }—	1	R,P	
	堆栈	SBDIV	—{SBDIV }—	1		
	4 位常数	BDIVS	—{BDIVS }—	1	K	
寄 存 器 变 换 指 令	BCD 加 1	INCR	—{INCR }—	2	R,P	
	BCD 减 1	DECR	—{DECR }—	2	R,P	
	BIN 加 1	BINC	—{BINC }—	2	R,P	
	BIN 减 1	BDEC	—{BDEC }—	2	R,P	
浮 动 小 数 点 演 算 指 令	浮动小数点加法	RADD	—{RADD }—	1	R,P	
	浮动小数点常数加法	RADDC	—{RADDC }—	3	F	
	浮动小数点减法	RSUB	—{RSUB }—	1	R,P	
	浮动小数点常数减法	RSUBC	—{RSUBC }—	3	F	
	浮动小数点乘法	RMUL	—{RMUL }—	1	R,P	
	浮动小数点常数乘法	RMULC	—{RMULC }—	3	F	
	浮动小数点除法	RDIV	—{RDIV }—	1	R,P	
	浮动小数点常数除法	RDIVC	—{RDIVC }—	3	F	
逻 辑 与	16BIT	ANDW	—{ANDW }—	1	R, P	
	32BIT	ANDD	—{ANDD }—	1	R, P	
	任意 BIT 长	ANDF	—{ANDF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SAND	—{SAND }—	1		
	8 位常数	ANDC	—{ANDC }—	2	K	
逻 辑 或	16BIT	ORW	—{ORW }—	1	R, P	
	32BIT	ORD	—{ORD }—	1	R, P	
	任意 BIT 长	ORF	—{ORF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SOR	—{SOR }—	1		
	8 位常数	ORC	—{ORC }—	2	K	

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
逻辑异或	16BIT	XORW	—{XORW }—	1	R, P	
	32BIT	XORD	—{XORD }—	1	R, P	
	任意 BIT 长	XORF	—{XORF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SXOR	—{SXOR }—	1		
	8 位常数	XORC	—{XORC }—	2	K	
比较指令	16BIT	CMPR	—{CMPR }—	1	R, P	
	32BIT	CMPRD	—{CMPRD }—	1	R, P	
	任意位长比较	CMPRF	—{CMPRF }—	2	GI,GQ,I,Q,M,S,T,C,SP [K(1-32)]	*1
	堆栈	SCMPR	—{SCMPR }—	1		
	8 位常数比较	CMPRC	—{CMPRC }—	2	K	
	浮动小数点比较	RCMPR	—{RCMPR }—	1	R, P	
	浮动小数点常数比较	RCMPRC	—{RCMPRC }—	3	F	
A C C 变 换 指 令	取返	INV	—{INV }—	1		
	BCD→BIN 码变换	BIN	—{BIN }—	1		
	BIN→BCD 码变换	BCD	—{BCD }—	1		
	10 进制补码变换	BCDCPL	—{BCDCPL }—	1		
	编码	ENCO	—{ENCO }—	1		
	译码	DECO	—{ENCO }—	1		
	7 段译码	SEG	—{SEG }—	1		
	右移	SHFR	—{SHFR }—	2	R,K	
	左移	SHFL	—{SHFL }—	2	R,K	
	循环右移	ROTR	—{ROTR }—	2	R,K	
	循环左移	ROTL	—{ROTL }—	2	R,K	
	ON 位求和	SUM	—{SUM }—	1		
	平方根	SQRT	—{SQRT }—	1		
GRAY→BCD 变换	GRAY	—{GRAY }—	1			
位替换指令	SFLDGT	—{SFLDGT }—	1			

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
接上	弧度变换	RAD	—{RAD }—	1		
	角度变换	DEG	—{DEG }—	1		
三角函数反三角函数运算	正弦	SIN	—{SIN }—	1		
	余弦	COS	—{COS }—	1		
	正切	TAN	—{TAN }—	1		
	反正弦	ASIN	—{ASIN }—	1		
	反余弦	ACOS	—{ACOS }—	1		
	反正切	ATAN	—{ATAN }—	1		
浮点三角函数反三角函数运算	正弦	RSIN	—{RSIN }—	1		
	余弦	RCOS	—{RCOS }—	1		
	正切	RTAN	—{RTAN }—	1		
	反正弦	RASIN	—{RASIN }—	1		
	反余弦	RACOS	—{RACOS }—	1		
	反正切	RATAN	—{RATAN }—	1		
浮点数 ACC 变换	浮动小数点变换	REAL	—{REAL }—	1		
	整数	INT	—{INT }—	1		
	平方根	RSQRT	—{RSQRT }—	1		
	弧度变换	RRAD	—{RRAD }—	1		
	角度变换	RDEG	—{RDEG }—	1		
浮点指数对数运算	幂运算	RPOW	—{RPOW }—	1		
	指数运算	REXP	—{REXP }—	1		
	自然对数运算	RLN	—{RLN }—	1		
	常用对数运算	RLG	—{RLG }—	1		
位 / 位置	表任意位置位	BITSET	—{BITSET }—	2	R,O	
	表任意位置复位	BITRST	—{BITRST }—	2	R,O	

*1 []: 第 2 操作数可使用的功能存储器。

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
数据块处理	数据块传送	MOVE	—{MOVE }—	2	R,P	
	ACC 逻辑与传送	ANDMOV	—{ANDMOV}—	2	R,P	
	ACC 逻辑或传送	ORMOV	—{ORMOV }—	2	R,P	
	ACC 逻辑异或传送	XORMOV	—{XORMOV}—	2	R,P	
	交换	SWAP	—{SWAP }—	2	R	
	ASC II →HEX 码变换	ATH	—{ATH }—	2	R	
	HEX →ASC II 码变换	HTA	—{HTA }—	2	R	
	同一数据的块写入	FILL	—{FILL }—	2	R,K	
	表右移	TSHFR	—{TSHFR }—	2	R,O	
	表左移	TSHFL	—{TSHFL }—	2	R,O	
表检索	同一数据块检索	SRCH	—{SRCH }—	2	R,K	
	多字节数据检索	BSRCH	—{BSRCH }—	2	R	
	数据分类	CLASS	—{CLASS }—	2	R,K	
带指针表处理	指针加算取出	TTD	—{TTD }—	2	R	
	指针减算取出	RFB	—{RFB }—	2	R	
	上托取出	RFT	—{RFT }—	2	R	
	指针加算存入	STT	—{STT }—	2	R,K	
	下推存入	ATT	—{ATT }—	2	R,K	
数据登记	数据区标号	DLBL	—{DLBL }—	2	K	
	数值数据登录	NCON	—{NCON }—	1	K	
	ASC II 数据登录	ACON	—{ACON }—	1	A	
	程序块索引读入	LDSIX	—{LDSIX }—	2	K	
	数据标号地址读出	LDLBL	—{LDLBL }—	2	K	
	登记数据寄存器传送	MOVAS	—{MOVAS }—	2		
	程序区·寄存器间传送	MOVMC	—{MOVMC }—	2	R,K	
鼓形控制指令	时间驱动型凸轮 (离散输出)	DRUM	—{DRUM }—			
	时间/事件型凸轮 (离散输出)	EDRUM	—{EDRUM }—			
	带掩膜事件驱动型凸轮(离散输出)	MDRMD	—{MDRMD }—			
	带掩膜事件驱动型凸轮(字输出)	MDRMW	—{MDRMW }—			

5 新增数据处理指令(8种)

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
32位运算指令	BCD 常数乘法运算	MULC	—{MULC }—	1	R,P,K	*1
	BCD 常数乘法运算	DIVC	—{DIVC }—	1	R,P,K	*1
	BIN 乘法运算（可带符号）	BMULD	—{BMULD }—	1	R,P,K	*1
	BIN 除法运算（可带符号）	BDIVD	—{BDIVD }—	1	R,P,K	*1
二进制比较指令	16BIT	BCMPR	—{BCMPR }—	1	R, P	*2
	32BIT	BCMPRD	—{BCMPRD }—	1	R, P	*2
	堆栈	BSCMPR	—{BSCMPR }—	1		*2
	8位常数比较	BCMPRC	—{BCMPRC }—	2	K	*2

*1[]: 第2操作数可使用的功能存储器；R/P连续2寄存器；K0~FFFFFF。

*2: 根据R7633的BIT15位内容决定是进行无符号位比较还是带符号位比较。

6 特殊对象指令(7种)

分类	指令名称	指令	符号	语数	可使用功能存储器	备注
通信指令	数据读出	RX	—{RX }—	2	GI,GQ,I,Q,M,S,T,C,SP,R,P	
	数据写入	WX	—{WX }—	2	GI,GQ,I,Q,M,S,T,C,SP,R,P	
MODBUS通信指令	数据读出	MRX	—{MRX }—		GI,GQ,I,Q,M,S,T,C,SP,R,P	
	数据写入	MWX	—{MWX }—		GI,GQ,I,Q,M,S,T,C,SP,R,P	
Email发信	Email 邮件发送	EMAIL	—{EMAIL }—		GI,GQ,I,Q,M,S,T,C,SP,R,P	注1
ASCII码处理指令	ASCII码输入	AIN	—{AIN }—			
	从存储器打印ASCII码	PRINTV	—{PRINTV }—			

注1: EMAIL 邮件发送指令仅NK0-CPU28DR-D机型支持。

注意

1. 用MOVMC指令从数据寄存器向用户存储器传送时,会加长PLC扫描时间。
2. 语数栏为空白的指令,其指令所用命令语数与指令内容有关。

7 IBOX 指令(28 种)

分类	指令名称	IBOX 指令代号	指令	备注
存储器指令	字传送指令	IB-200	MOVEW	
	双字传送指令	IB-201	MOVED	
开关量指令	ON/OFF 线圈指令	IB-300	PONOFF	
	ON 延时定时器	IB-301	ONDTMR	
	OFF 延时定时器	IB-302	OFFDTMR	
	单次扫描输出指令	IB-303	ONESHOT	
模拟量指令	高/低限报警指令-BIN 数	IB-401	HILOALB	
	采样值过滤指令-BIN 数	IB-402	FILTERB	
	工程量换算指令-12 位 BIN 数	IB-403	ANSCLB	
	高/低限报警指令-BCD 数	IB-421	HILOAL	
	采样值过滤指令-BCD 数	IB-422	FILTER	
	工程量换算指令-12 位 BCD 数	IB-423	ANSCL	
算术相关指令	BIN 算术运算指令	IB-501	MATHBIN	
	BIN 求和指令	IB-502	SUMBIN	
	BIN 平方指令	IB-503	SQUAREB	
	BCD 算术运算指令	IB-521	MATHBCD	
	BCD 求和指令	IB-522	SUMBCD	
	BCD 平方指令	IB-523	SQUARE	
	浮点数算术运算指令	IB-541	MATHR	
	浮点数求和指令	IB-542	SUMR	
	浮点数平方指令	IB-543	SQUARER	
	BCD→浮点数转换指令	IB-560	BCDTOR	
	浮点数→BCD 转换指令	IB-561	RTOBCD	
	双字 BCD→浮点数转换指令	IB-562	BCDTORD	
	浮点数→双字 BCD 转换指令	IB-563	RTOBCDD	
数据运算	BCD 运算	IB-550	MATHBCDD	结果 32 位
	BIN 运算	IB-551	MATHBIND	结果 32 位
	CRC 校验码生成指令	IB-552	RTOBCDD	

附录二 有关软件专利权的说明

本产品使用了 TOPPERS 协会的 TOPPERS 软件技术和 TINET 协议栈技术,而 TINET 是在 FreeBSD 的基础上开发的。

根据以上软件技术著作权所有人的要求,对以上软件技术的使用作以下版权说明。

(1)FreeBSD

Copyright (c) 1980, 1986, 1993

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(2)KAME

Copyright (C) 1995, 1996, 1997, and 1998 WIDE Project.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the project nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE PROJECT AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE PROJECT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(3) Ethernet • device driver

Copyright (c) 1995, David Greenman

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice unmodified, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Device driver for National Semiconductor DS8390/WD83C690 based ethernet adapters. By David Greenman, 29-April-1993

Currently supports the Western Digital/SMC 8003 and 8013 series, the SMC Elite Ultra (8216), the 3Com 3c503, the NE1000 and NE2000, and a variety of similar clones.

(4)/usr/sbin/ppp

User Process PPP

Written by Toshiharu OHNO (tony-o@ij.ad.jp)

Copyright (C) 1993, Internet Initiative Japan, Inc. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the Internet Initiative Japan, Inc. The name of the IJ may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

(5)/usr/sbin/pppd

main.c - Point-to-Point Protocol main module

Copyright (c) 1989 Carnegie Mellon University.

All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by Carnegie Mellon University. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

(6)TINET 和 TOPPERS**TINET (TCP/IP Protocol Stack)**

Copyright (C) 2001-2006 by Dep. of Computer Science and Engineering
Tomakomai National College of Technology, JAPAN

只要符合以下的(1) ~ (4)条件,或者符合 Free Software Foundation 机构公布的 GNU General Public License 第2版中记述的条件,著作权所有者将无偿同意本软件(含本软件修正品,以下同)的使用,复制,修改及再分发(以下统称为使用)。

(1) 以源代码形式使用本软件时,须原封不动地把著作权申明,使用条件及以下的无保障规定放入源代码中。

(2) 以程序库等能被其它软件调用的形式再分发时,须在再分发说明书(使用说明等)中,写入上述的著作权申明,使用条件及以下的无保障规定。

(3) 把本软件嵌入机器等,以不能被其它软件调用的形式使用本软件时,须满足以下条件。

(a) 再分发时使用书(使用说明等)中,须写入上述著作权申明,使用条件及以下的无保障规定。

(4) 因使用本软件而引起的直接或间接的损失,均不得向上述著作权所有者及 TOPPERS 协会要求承担责任。

TINET 和 TOPPERS 软件以无保障方式提供给社会。软件著作权所有者及 TOPPERS 协会对本软件及能否使用本软件的探讨均不负任何责任。同时,对因使用本软件而引起的直接或间接的损失,均不负任何责任。

光洋电子(无锡)有限公司

Koyo ELECTRONICS (WUXI) CO., LTD.

地址：江苏省无锡市滨湖区建筑西路 599 号1栋21 层

邮编：214072

电话：0510-85167888 传真：0510-85161393

<http://www.koyoele.com.cn>

KEW-M2611B

2023年03月