

Q系列以太网接口模块

用户手册

(基础篇)

MITSUBISHI



三菱可编程逻辑控制器

MELSEC-Q

QJ71E71-100

QJ71E71-B5

QJ71E71-B2

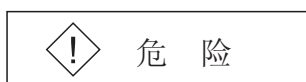
● 安全注意事项 ●

(使用设备前请阅读本说明)

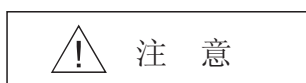
使用本产品前，请仔细阅读本手册及本手册提到的相关资料，正确操作并注意安全。

本手册中给出的说明均是关于本产品的。对于可编程控制器系统的安全注意事项，请阅读 CPU 模块的用户手册。

在本手册中，安全守则的等级分为“危险”和“注意”。



表示错误操作可能造成灾难性后果，引起死亡或重伤事故。



表示错误操作可能造成危险的后果，引起人员中等伤害或轻伤还可能使设备损坏。

注意根据情况不同，△注意这一级也能引发严重后果。

所以对两级注意事项，都须遵照执行，因为它们对于操作人员安全是非常重要的。

妥善保管本手册，请放在操作人员易于取阅的地方，并将本手册交给最终使用者。

[设计注意事项]

⚠ 危险

- 当数据链接出现通讯错误时，有关每个站的运行状态详情，参见各个数据链接的手册。
错误输出或故障可能引发事故。
不参照手册进行操作有可能由于错误输出或故障而引发事故。
- 为了防止外来非法电子邮件造成 PLC 系统故障，需要采取正确的防范措施(如病毒检测)，不让本模块的邮件服务器接收非法电子邮件。
- 为了确保 PLC 系统的安全，用户必须采取适当的措施，不让未经授权的人员通过互联网从外部设备访问 PLC 系统。
- 为了在运行中控制 PLC(修改数据)，可将外围设备连接到 CPU 模块，或将个人计算机连接到智能功能模块，然后在顺控程序中加入互锁回路，从而自始至终保持整个系统的安全。另外，对运行中的 PLC 进行其它控制操作(程序修改和操作状态修改(状态控制))之前，一定要仔细阅读本手册并确定是绝对安全的。
尤其当通过外部设备对远程 PLC 进行上述控制操作时，由于异常数据通讯，可能无法及时处理 PLC 上发生的任何问题。另应在顺控程序中加入互锁回路，确定系统如何在对方的设备和 PLC CPU 之间处理异常数据通讯。

[设计注意事项]

⚠ 危险

- 不要将任何数据写入智能功能模块缓冲存储器的“系统区”，也不要输出(打开)“禁用”信号，该信号是 PLC CPU 发给智能功能模块的信号之一。如果数据写入“系统区”或者输出“禁用”信号，PLC 系统有发生故障的危险。

⚠ 注意

- 不要将控制线或通讯电缆捆扎到主回路或电源线上，安装时也不要使它们靠得太近。安装时它们应彼此间隔 100mm (3.94in.) 或更远。否则可能会产生噪声，引起故障。
- 从外部设备上控制 PLC CPU 的状态(远程运行/停止等)时，选择由用户提前设定的“始终等待打开”参数。(用操作设置中的初始化计时设置进行选择)。如果选择了“不等待打开”，发出远程“停止”命令时，通讯回路关闭。自此不能再在 PLC CPU 上打开通讯回路，并且从外部设备上发出的远程“运行”命令也不能启动通讯回路。

[安装注意事项]

⚠ 危险

- 在符合本手册中规定的一般操作环境规格下使用 PLC。
在不符合本手册中规定的一般操作环境规格下使用 PLC 时，可能会引起电击、火灾、故障，并会损坏模块，或使模块性能变差。
- 安装模块时，按住模块下部的安装杆，将模块的锁紧扣插进基板安装孔中扣牢。然后，以安装孔为支撑点安全地安装模块。
如果模块安装得不正确，可能导致模块故障、失效或跌落。
尤其在可能一直振动的环境中使用，要用螺钉紧固模块。
- 确保按规定的扭矩紧固螺钉。如果螺钉松动，可能导致模块短路、故障或跌落。如果螺钉拧得过紧，可能损坏螺钉并导致模块短路、故障或跌落。
- 在安装或拆卸模块之前，确保断开外部电源的所有相位。
否则可能损坏模块。
- 不要直接触摸模块的导电部分。
否则可能会引起模块故障或失效。

[接线注意事项]

⚠ 注意

- 采用压接、压移或焊接方法在外部连接接上连接器时，必须正确使用制造商规定的工具。
如果连接不完善的话，可能导致模块短路，引起火灾，或故障。
- 在装有模块的站供电时，不要连接 AUI 电缆。
- 必须将连接模块的通讯电缆和电源电缆放置在电缆槽中或者用夹子固定。
如果电缆没有敷设在电缆槽中或没有用夹子固定，它们的位置就可能不稳定或来回晃动，并在不经意间被拉动。这种不正确的电缆连接会损坏模块和电缆，也有可能导致模块故障。
- 按规定扭矩紧固端子螺钉。如果端子螺钉松动，可能导致模块短路、故障或跌落。如果端子螺钉紧固过度，可能会损坏螺钉，造成模块短路、故障或跌落。
- 不要抓住电缆拆除连接在模块上的通讯电缆和电源电缆。
当拆除带有连接器的电缆时，抓住与模块相连的连接器，把连接器拔出来，拆下电缆。
当拆除连接到端子排的电缆时，在拆除之前，首先松开端子排上的螺钉。
拉动仍连接在模块上的电缆，可能会导致模块故障，损坏模块或电缆。
- 小心不要让任何异物(如接线碎片)进入模块内部。这些异物可能导致火灾，并破坏模块或使模块发生故障。
- 为了防止接线时异物(如线头)进入模块内部，在模块上部粘有一层防护膜。
接线过程中不要取下该防护膜。但是一定要在操作模块之前取下防护膜，以利操作期间产生的热量散发。
- 一定要在接线后重新接通电源开始操作之前，将端子盒盖安装在产品上。不盖上端子盖可能导致故障。
- 正确焊接同轴电缆连接器。焊接不良可能导致故障。

[装配和维护注意事项]

⚠ 注意

- 不要拆开及改造模块。这可能导致破损、故障、人身伤害及火灾。
- 在安装或拆卸模块之前，必须先切断外部电源的所有相。否则可能导致模块受损或故障。
- 首次使用本产品后，请不要从基板上安装/卸载模块超过 50 次。(符合 IEC61131-2) 否则可能因为接触不良而引起模块故障。
- 通电时不要触摸端子。否则可能会遭到电击，或导致模块故障。
- 在清理模块或重新紧固端子和模块固定螺钉之前，必须已切断外部电源的所有相位。否则可能导致模块受损或故障。
如果螺钉松动，可能导致模块短路、故障或跌落。
如果螺钉拧得过紧，可能损坏螺钉并导致模块短路、故障或跌落。
- 在接触设备之前，必须先接触已接地的金属，释放掉人体等所携带的静电。
如果不释放掉静电，有可能导致设备故障或误动作。

[操作注意事项]

⚠ 注意

- 在开始对 PLC 控制操作之前(特别是通过个人计算机等连接到智能功能模块的方法，对数据、程序和运行状态(状态控制)进行修改前)，请仔细阅读本手册并确保绝对安全。
对数据、程序和运行状态进行的不正确修改可能导致系统故障，损坏机器或引发事故。

[报废处理注意事项]

⚠ 注意

- 报废时，将本产品当作工业废料处理。

修订记录

*手册编号在封底的左下角

印刷日期	*手册编号	修改内容
2001年11月	SH(NA)-080235C-A	第一版
2006年11月	SH(NA)-080235C-B	第二版 部分修订

英文手册原稿: SH(NA)-080009-J

本手册未被授予工业知识产权或其他任何种类的权利，亦未被授予任何专利许可证。三菱电机株式会社对使用本手册中的内容造成的工业知识产权问题不承担责任。

© 2001 三菱电机

前言

感谢您购买 MELSEC-Q 系列 PLC。

使用设备前请认真阅读本手册，以对您购买的 Q 系列 PLC 的功能和性能有清晰的认识，从而确保正确地使用。请把本手册的拷贝件发给最终使用者。

目录 (本手册)

安全注意事项	A - 1
修订记录	A - 5
目录	A - 6
关于本手册	A - 14
符合 EMC 指令和低电压指令	A - 14
手册使用和手册结构	A - 15
关于通用术语和缩写	A - 19
产品配置	A - 20
1 概述	1-1 到 1-16
1.1 以太网模块的概述	1 - 1
1.2 以太网模块的特性	1 - 2
1.3 功能版本 B 或以上版本中新增的功能	1 - 12
1.4 软件配置	1 - 14
2 系统配置	2-1 到 2-20
2.1 可应用系统	2 - 1
2.2 网络配置需要的设备	2 - 3
2.3 用于多 CPU 系统中	2 - 8
2.4 与 Q00J/Q00/Q01CPU 一起使用时	2 - 10
2.5 与 Q12PRH/Q25PRHCPU 一起使用时	2 - 11
2.6 在 MELSECNET/H 远程 I/O 站上使用	2 - 14
2.7 检查功能版本和系列号	2 - 18
3 规格	3-1 到 3-28
3.1 性能规格	3 - 1
3.2 通讯用的数据代码	3 - 3
3.3 外部设备和每个通讯功能的附加功能之间的关系	3 - 5
3.4 以太网模块功能列表	3 - 6
3.5 专用指令列表	3 - 8
3.6 GX Developer 设置的以太网模块项目列表	3 - 9
3.7 从 PLC CPU 发出的输入/输出信号和发给 PLC CPU 的输入/输出信号列表	3 - 10
3.8 缓冲存储器的应用和分配列表	3 - 12
4 操作之前的设置和步骤	4-1 到 4-26
4.1 装载和安装	4 - 1

4.1.1 处理注意事项	4 - 1
4.1.2 安装环境	4 - 2
4.2 开始操作之前的设置和步骤	4 - 3
4.3 以太网模块的部件	4 - 5
4.4 连接到网络	4 - 7
4.4.1 连接到 10BASE-T/100BASE-TX 网络	4 - 8
4.4.2 连接到 10BASE5 网络	4 - 9
4.4.3 连接到 10BASE2 网络	4 - 10
4.5 用 GX Developer 进行设置	4 - 12
4.5.1 I/O 分配设置	4 - 12
4.5.2 其它设置	4 - 13
4.6 设置 MNET/10H 以太网卡号的网络参数	4 - 16
4.7 操作设置	4 - 19
4.8 自诊断测试	4 - 22
4.8.1 自抑制测试	4 - 22
4.8.2 硬件测试 (H/W 测试)	4 - 23
4.9 维护和检查	4 - 24
4.9.1 维护和检查	4 - 24
4.9.2 安装和拆卸模块	4 - 25

5 通讯步骤

5-1 到 5-112

5.1 通讯步骤概要	5 - 1
5.2 初始化处理	5 - 3
5.2.1 初始化处理	5 - 3
5.2.2 初始化设置	5 - 4
5.2.3 重新初始化处理	5 - 10
5.3 路由中继参数	5 - 16
5.4 确认初始化处理的完成	5 - 21
5.4.1 使用 GX Developer 进行 PING 测试 (通过以太网板)	5 - 21
5.4.2 使用 GX Developer 进行 PING 测试 (通过 CPU)	5 - 26
5.4.3 使用 GX Developer 进行环路回送测试	5 - 29
5.4.4 PING 命令 (个人计算机 → 以太网模块)	5 - 34
5.4.5 环路回送测试 (使用 MC 协议进行通讯)	5 - 35
5.5 开放设置	5 - 36
5.6 开放处理/关闭处理	5 - 43
5.6.1 主动开放处理/关闭处理	5 - 45
5.6.2 被动开放处理/关闭处理	5 - 52
5.6.3 UDP/IP 开放处理/关闭处理	5 - 60
5.7 成对开放	5 - 63
5.7.1 成对开放	5 - 63
5.7.2 使用 GX Developer 成对开放设置的范例	5 - 64
5.8 自动开放 UDP 端口	5 - 66
5.9 使用 QCPU 远程口令功能进行通讯	5 - 68
5.9.1 设置远程口令时的数据通讯	5 - 69
5.9.2 使用远程口令核对功能时的注意事项	5 - 76
5.9.3 数据通讯步骤	5 - 77
5.9.4 远程口令解锁处理或锁定处理异常完成时	5 - 79

5.9.5 如何设置远程口令核对的目标连接.....	5 - 81
5.9.6 远程口令核对功能所用的缓冲存储器.....	5 - 83
5.9.7 设置远程口令核对时的数据通讯.....	5 - 87
5.10 网络集线器连接状态监视功能.....	5 - 88
5.11 在冗余系统中配置网络(冗余系统支持功能).....	5 - 89
5.11.1 发送系统切换请求到控制系统 CPU.....	5 - 89
5.11.2 通讯路径迂回功能.....	5 - 95
5.11.3 冗余设置.....	5 - 97
5.11.4 用于冗余系统支持功能的缓冲存储器.....	5 - 99
5.11.5 用于在冗余系统中使用以太网模块的数据通讯.....	5 - 100

6 使用 MC 协议进行通讯	6-1 到 6-6
-----------------------	------------------

6.1 数据通讯功能.....	6 - 1
6.1.1 使用 MC 协议访问 PLC CPU.....	6 - 1
6.1.2 数据通讯的信息格式和控制步骤.....	6 - 2
6.1.3 执行数据通讯的 PLC CPU 设置.....	6 - 3
6.1.4 对多 CPU 系统或冗余系统的支持.....	6 - 4
6.1.5 对 QCPU 远程口令功能的支持.....	6 - 4
6.2 使用 MX Component.....	6 - 5

7 固定缓冲存储器通讯(有顺序控制方法)	7-1 到 7-22
-----------------------------	-------------------

7.1 控制方法.....	7 - 1
7.2 发送控制方法.....	7 - 3
7.3 接收控制方法.....	7 - 5
7.3.1 用主程序接收(专用指令: BUFRCV).....	7 - 5
7.3.2 用中断程序进行接收处理(专用指令: BUFRCVS).....	7 - 7
7.4 数据格式.....	7 - 12
7.4.1 标题.....	7 - 12
7.4.2 应用数据.....	7 - 13
7.5 编程.....	7 - 17
7.5.1 创建程序时的注意事项.....	7 - 17
7.5.2 固定缓冲存储器通讯程序范例(有顺序的控制方法).....	7 - 18

8 固定缓冲存储器通讯(无顺序的控制方法)	8-1 到 8-22
------------------------------	-------------------

8.1 控制方法.....	8 - 1
8.2 发送控制方法.....	8 - 4
8.3 接收控制方法.....	8 - 6
8.3.1 用主程序接收(专用指令: BUFRCV).....	8 - 6
8.3.2 用中断程序接收(专用指令: BUFRCVS).....	8 - 8
8.4 数据格式.....	8 - 10
8.5 使用 UDP/IP 进行同步广播.....	8 - 11
8.5.1 用同步广播发送.....	8 - 11
8.5.2 用同步广播接收.....	8 - 13
8.5.3 使用同步广播功能时的注意事项.....	8 - 16

8.6 编程	8 - 17
8.6.1 创建程序时的注意事项.....	8 - 17
8.6.2 固定缓冲存储器通讯程序范例(无顺序的控制方法)	8 - 18

9 使用随机访问缓冲存储器进行通讯	9-1 到 9-16
--------------------------	-------------------

9.1 控制方法	9 - 1
9.1.1 从外部设备读请求的控制方法.....	9 - 3
9.1.2 从外部设备写请求的控制方法.....	9 - 4
9.2 数据格式	9 - 5
9.2.1 标题	9 - 5
9.2.2 应用数据	9 - 6
9.2.3 命令/响应格式的范例.....	9 - 11
9.3 随机访问缓冲存储器的物理地址和逻辑地址.....	9 - 15
9.4 创建程序时的注意事项.....	9 - 16

10 专用指令	10-1 到 10-26
----------------	---------------------

10.1 专用指令列表	10 - 1
10.2 BUFRCV 指令.....	10 - 2
10.3 BUFRCVS 指令.....	10 - 5
10.4 BUFSND 指令.....	10 - 8
10.5 CLOSE 指令.....	10 - 11
10.6 ERRCLR 指令.....	10 - 14
10.7 ERRRD 指令.....	10 - 17
10.8 OPEN 指令.....	10 - 19
10.9 UINI 指令.....	10 - 23

11 故障排除	11-1 到 11-64
----------------	---------------------

11.1 如何使用 LED 显示来检查出错.....	11 - 2
11.1.1 检查出错显示	11 - 2
11.1.2 如何熄灭 COM.ERR LED 及如何读/清除出错信息	11 - 4
11.2 如何通过 GX Developer 检查出错.....	11 - 5
11.2.1 以太网诊断	11 - 6
11.2.2 系统监视	11 - 8
11.2.3 可用 GX Developer 诊断功能监视的缓冲存储器	11 - 10
11.2.4 使用缓冲存储器批量监视功能来检查出错信息.....	11 - 12
11.3 出错代码列表	11 - 13
11.3.1 在数据通讯期间返回至外部设备的结束代码(完成代码)	11 - 22
11.3.2 使用 A 兼容的 1E 帧进行通讯期间返回的异常代码	11 - 24
11.3.3 存储在缓冲存储器中的出错代码.....	11 - 25
11.4 故障排除流程图	11 - 42
11.4.1 固定缓冲存储器通讯期间发送出错 (有顺序和无顺序共用)	11 - 45
11.4.2 固定缓冲存储器通讯期间接收出错 (有顺序和无顺序共用)	11 - 47
11.4.3 随机访问缓冲存储器通讯期间出错.....	11 - 50
11.4.4 使用 MC 协议通讯出错.....	11 - 52
11.4.5 电子邮件通讯期间发送出错.....	11 - 54

11.4.6 电子邮件通讯期间接收出错.....	11 - 56
11.4.7 冗余系统中的错误.....	11 - 58

附录	附录-1 到附录-102
----	--------------

附录 1 以太网模块的功能升级.....	附录 - 1
附录 1.1 以太网模块功能的比较.....	附录 - 1
附录 1.2 将模块从功能版本 A 升级到 B 或以上版本的注意事项.....	附录 - 3
附录 2 QnA/A 系列模块.....	附录 - 4
附录 2.1 以太网模块和 QnA/A 系列模块之间的功能比较.....	附录 - 4
附录 2.2 使用专门为 QnA/A 系列模块设计的程序.....	附录 - 6
附录 3 将以太网模块安装在现有的系统上.....	附录 - 9
附录 4 处理时间.....	附录 - 9
附录 5 ASCII 代码代表.....	附录 - 18
附录 6 参考材料.....	附录 - 18
附录 7 外部尺寸.....	附录 - 19
附录 8 程序范例.....	附录 - 21
附录 8.1 使用 Visual Basic®.NET 和 Visual C++®.NET 的程序范例.....	附录 - 23
附录 8.1.1 使用 MC 协议-1 通讯的范例.....	附录 - 23
附录 8.1.2 使用 MC 协议-2 通讯的程序范例.....	附录 - 32
附录 8.1.3 使用 MC 协议-3 通讯的程序范例.....	附录 - 43
附录 8.2 使用 Visual Basic® 6.0/ Visual C++® 6.0 或更早版本的程序范例.....	附录 - 52
附录 8.2.1 使用 MC 协议-1 进行通讯的程序范例.....	附录 - 52
附录 8.2.2 使用 MC 协议-2 通讯的程序范例.....	附录 - 61
附录 8.2.3 使用 MC 协议-3 通讯的程序范例.....	附录 - 72
附录 9 通讯支持工具(MX Component).....	附录 - 75
附录 9.1 MX Component 概述.....	附录 - 75
附录 9.2 MX Component 的使用步骤.....	附录 - 79
附录 10 以太网和 IEEE802.3 之间的差异.....	附录 - 87
附录 11 以太网模块支持的 ICMP 协议.....	附录 - 87
附录 12 设置值记录表单.....	附录 - 88

索引	索引-1 到 索引-4
----	-------------

- 1 概述
 - 1.1 概述
 - 1.2 关于功能版本 B 或以后版本中增加的功能
- 2 使用电子邮件功能
 - 2.1 电子邮件功能
 - 2.2 应用系统的配置和环境
 - 2.3 使用电子邮件功能的注意事项
 - 2.4 电子邮件规格
 - 2.5 电子邮件功能的处理流程
 - 2.6 从 GX Developer 设置电子邮件
 - 2.7 由 PLC CPU 发送/接收电子邮件 (附件)
 - 2.8 由 PLC CPU 发送电子邮件正文
 - 2.9 使用 PLC CPU 监视功能发送电子邮件
- 3 与 MELSECNET/H、MELSECNET/10 中继通讯时
 - 3.1 MELSECNET/H、MELSECNET/10 中继通讯
 - 3.2 远程站 PLC 可访问范围和站
 - 3.3 设置访问其它站
 - 3.4 访问其它站的步骤
 - 3.5 访问其它站的注意事项
- 4 当 QCPU 使用数据链接指令访问其它站 PLC 时
 - 4.1 使用数据链接指令访问其它站
 - 4.2 访问其它站的注意事项
 - 4.3 使用数据链接指令
 - 4.4 数据链接指令
 - 4.5 数据发送/接收
 - 4.6 其它站的读/写字软元件 (READ/WRITE)
 - 4.7 其它站的读/写字软元件 (ZNRD/ZNWR)
 - 4.8 读/写时钟数据、远程 RUN/远程 STOP (REQ)
 - 4.9 数据链接指令的出错代码
- 5 使用文件传送功能 (FTP 服务器) 时
 - 5.1 文件传送功能
 - 5.2 文件可传送范围
 - 5.3 从 GX Developer 设置文件传送的 FTP 参数
 - 5.4 外部设备侧的步骤和需要的处理 (FTP 客户机)
 - 5.5 使用文件传送功能的注意事项
 - 5.6 FTP 命令
- 6 专用指令
 - 6.1 专用指令列表
 - 6.2 MRECV 指令
 - 6.3 MSEND 指令
 - 6.4 READ 指令
 - 6.5 RECV 指令 (用于主程序)
 - 6.6 RECVS 指令 (用于中断程序)
 - 6.7 REQ 指令 (远程 RUN/STOP)
 - 6.8 REQ 指令 (时钟数据读/写)
 - 6.9 SEND 指令
 - 6.10 SREAD 指令
 - 6.11 SWRITE 指令
 - 6.12 WRITE 指令
 - 6.13 ZNRD 指令
 - 6.14 ZNWR 指令

- 1 概述
 - 1.1 概述
- 2 系统配置
 - 2.1 系统配置
 - 2.2 使用 Web 功能的注意事项
- 3 操作步骤
 - 3.1 使用 Web 功能进行通讯的一般步骤
 - 3.2 如何获得和建立通讯库和示例屏幕
- 4 使用示例屏幕验证 Web 功能的操作
 - 4.1 可用在示例屏幕上的 Web 功能项目
 - 4.2 操作步骤
 - 4.3 示例屏幕的说明
 - 4.4 示例屏幕上的数据通讯实例
 - 4.5 示例屏幕上的文件配置
- 5 创建一个访问 PLC 文件的实例
- 6 通讯库功能

- 1 概述
 - 1.1 MELSEC 通讯协议概述
 - 1.2 MELSEC 通讯协议特性
 - 2 使用 MELSEC 通讯协议进行数据通讯
 - 2.1 数据通讯帧的类型和应用
 - 2.2 每个数据通讯帧的可访问范围
 - 2.3 如何读取 MC 协议的控制顺序
 - 2.4 PLC CPU 端的访问时序
 - 2.5 运行期间写入 PLC CPU 的设置方法
 - 2.6 访问其它站
 - 2.7 数据通讯的注意事项
 - 2.8 串行通信模块传送顺序的时间图和通讯时间
 - 2.9 通过 MELSECNET/H、MELSECNET/10 访问其它站的传送时间
 - 2.10 与多个 PLC 系统的兼容性
 - 2.11 与 Q00CPU、Q01 CPU 串行通信模块的兼容性
 - 3 使用 QnA 兼容 3E/3C/4C 帧或 4E 帧进行通讯时
 - 3.1 信息格式
 - 3.2 QnA 兼容 3E/3C/4C 帧或 4E 帧的命令和功能列表
 - 3.3 软元件内存的读/写
 - 3.4 缓冲存储器的读/写
 - 3.5 从智能功能模块的缓冲存储器读出数据和将数据写入智能功能模块的缓冲存储器
 - 3.6 PLC CPU 状态控制
 - 3.7 驱动存储器的碎片整理(用于其它站的 QnACPU)
 - 3.8 文件控制
 - 3.9 串行通信模块用户帧的寄存、删除和读取
 - 3.10 串行通信模块的全球化功能
 - 3.11 将串行通信模块的数据传送到外部设备(开启要求功能)
 - 3.12 初始化串行通信模块的传送步骤
 - 3.13 串行通信模块的模式切换
 - 3.14 关掉显示的 LED 并初始化串行通信模块的通讯出错信息和出错代码
 - 3.15 关闭以太网模块的 COM.ERR LED
 - 3.16 回路测试
 - 3.17 监视串行通信模块 PLC CPU 的寄存或删除
 - 3.18 远程口令解锁/锁定
 - 4 使用 QnA 兼容 ZC 帧通讯时
 - 4.1 控制顺序和信息格式
 - 4.2 数据指定项的内容
 - 4.3 QnA 兼容 2C 帧的命令和功能列表
 - 4.4 数据通讯的注意事项
 - 4.5 使用 QnA 兼容 2C 帧进行数据通讯的实例
 - 5 使用 A 兼容 1C 帧进行通讯时
 - 5.1 控制顺序和信息格式
 - 5.2 软元件内存的读/写
 - 5.3 扩展文件寄存器的读和写
 - 5.4 智能功能模块缓冲存储器中的读和写
 - 5.5 回路测试
 - 6 使用 A 兼容 1E 帧进行通讯
 - 6.1 信息格式和控制顺序
 - 6.2 A 兼容 1E 帧的命令和功能列表
 - 6.3 软元件内存的读/写
 - 6.4 扩展文件寄存器的读和写
 - 6.5 智能功能模块缓冲存储器中的读和写
- 附录
- 附录-1 通过指定软元件扩展存储器的读和写
 - 附录-2 从缓冲存储器读出数据和将数据写入缓冲存储器
 - 附录-3 使用 MC 协议进行通讯时 PLC CPU 端的处理时间

关于本手册

下列手册也与本产品有关。

如果需要，可按照下表所列的详细内容进行订购。

相关手册

手册名称	手册编号
Q 系列以太网接口模块用户手册(应用篇) 本手册说明了通过 MELSECNET/H、MELSECNET/10，使用数据链接指令的通讯功能如何使用以太网模块的电子 邮件功能、PLC CPU 状态监视功能、通讯功能和如何文件传送(FTP 服务器)等。 (另售)	SH(NA)-080285C
Q 系列以太网接口模块用户手册(Web 功能) 本手册说明了如何使用以太网模块的 Web 功能。 (另售)	SH(NA)- 080410C
Q 系列 MELSEC 通讯协议参考手册 本手册说明了外部设备使用 MC 协议，用串行通信模块/以太网模块从 PLC CPU 读取数据和将数据写入 PLC CPU 的通讯方法和控制步骤。 (另售)	SH(NA)- 080414C

符合 EMC 指令和低电压指令

当安装三菱 PLC 至其它工业机械或设备中组合使用时，应符合 EMC 指令和低电压指令。详细内容请参阅 CPU 模块用户手册(硬件篇)的第 3 章“EMC 指令和低电压指令”。

CE 标志印刷在符合 EMC 指令和低电压指令的 PLC 主体上的额定铭牌上。

要使此产品符合 EMC 指令和低电压指令，请参阅 CPU 模块用户手册(硬件篇)的第 3 章“EMC 指令和低电压指令”中的 3.1.3 节“电缆”部分。

● 如何使用本手册

在本手册中，介绍了以太网模块的各种应用(QJ71E71-100、QJ71E71-B5 和 QJ71E71-B2)。

在使用本手册时，请参考下列关键项目。

(1) 特性和应用列表

(a) 特性和功能

- 第 1 章说明以太网模块的特性。
- 第 3 章说明以太网模块的公共功能和规格

(b) 包装项目和网络配置项目

- 第 1 章“产品配置”之前的部分说明了与以太网模块一起提供的项目。
- 2.2 节说明了以太网模块的系统配置。
用户须单独购买没有与模块包装在一起的零部件。

(2) 开始操作以太网模块之前所需的处理

(a) 启动顺序

- 4.2 节概略地说明了开始操作以太网模块之前的步骤。

(b) 连接到以太网网络系统

- 2.2 节说明了需要连接到以太网网络系统的装置。
- 4.4 节说明了每种接口类型的连接方法。

(c) 开始操作以太网模块之前所需的参数设置

- 4.5 节说明了使用以太网模块所需的 GX Developer 设置参数画面的类型。
- 3.6 节说明了要使用每种功能所需的参数设置。

按照提供详细说明的相关章节来确认需要的参数，并进行设定，后将数值保存在安装有以太网模块的 PLC CPU 中。

(d) 如何核对以太网模块的故障

- 4.8 节说明了以太网模块的自诊断测试。

(e) 如何检查与外部设备的连接错误

- 5.4.1 节到 5.4.3 节说明了如何通过 GX Developer 执行 PING 测试和回路测试，检查连接错误。
- 5.4.4 节说明了使用“PING”命令检查连接错误。
- 5.4.5 节说明了如何在 MC 协议的基础上执行回路测试，检查连接错误。
* 通过 MC 协议使用回路测试命令的详情，请参考 Q 系列 MELSEC 通讯协议参考手册。

(3) 以太网模块和外部设备之间的连接

(a) 通讯步骤

5.1 节概略地说明了通讯步骤。

(b) 与外部设备的连接

- 5.6 节说明了每种通讯方法(TCP/IP、UDP/IP)的连接(建立连接和断开连接处理)和打开方式(主动、被动)，包括编程步骤。

(4) 数据通讯功能的详情

(a) 通讯功能

- 1.2 节概略地说明了以太网模块通讯功能和可参考更多详细说明的相关章节号和手册名称。
- 用户手册(应用篇)中说明了以太网模块的特殊功能。
- 用户手册(Web 功能篇)中说明了以太网模块的 Web 功能。

(5) 数据通讯功能和编程

(a) 如何从 PLC CPU 读出数据和将数据写入 PLC CPU。

- 从 PLC CPU 读出数据和将数据写入 PLC CPU 是通过使用 MC 协议的通讯功能实现的。
- 第 6 章概略地说明了使用 MC 协议的通讯功能。
* 详情请参见 Q 系列 MELSEC 通讯协议参考手册。

(b) 如何在 PLC CPU 和外部设备之间发送数据和接收数据。

- PLC CPU 和外部设备之间的数据通讯是通过使用固定缓冲存储器或随机访问缓冲存储器的通讯功能实现的。
- 第 7 章和第 8 章详细地说明了使用固定缓冲存储器的通讯功能和编程。
- 第 9 章详细地说明了使用随机访问缓冲存储器的通讯功能和编程。

(6) 如何检查出错和如何采取纠正措施

(a) 出错代码的内容

- 第 11 章说明了故障排除、如何检查出错及出错代码的内容及参考手册。

(b) 出错代码在以太网模块缓冲存储器中的存储区

- 11.3 节说明了出错代码在缓冲存储器中的存储区。

(7) 功能版本 B 或以后版本中增加的功能

- 1.3 节提供了增加的功能列表，及针对此点进行详细说明的手册。
- 附录 1.1 介绍了以太网模块功能的比较。

● 本手册的结构

(1) 用 GX Developer 设置参数

(a) 使用 GX Developer 设定参数，可以在以太网模块中简化与外部设备进行通讯的顺控程序。

(b) 在本手册中，用 GX Developer 设置参数时，按如下方式说明：

- 1) 4.5 节说明了设置画面的类型、设置对象、设置项目和概要。
- 2) 4.5 节画面上或文字中的具体说明部分说明详细内容。

(c) 按照相关的说明章节或手册确认需要的参数，并进行设定，然后将数值保存在安装有以太网模块的 PLC CPU 中。

(2) GX Developer 的设置画面

本手册以下列格式说明用 GX Developer 设置的参数：

4.7 操作设置

本节说明如何设置操作参数。
通过选择[设置 MNET/10H 以太网卡号]-[操作设置]，启动“以太网操作”画面。

1) 表示如何启动设置画面

2) 表示GX Developer设置画面

3) 表示设置项和设定值的存储目的地址(*1)。

4) 表示设定值和具体说明的相关章节。

(a) 与外部设备通讯时，选择通讯数据的格式。

设置名称	设定值	说明
二进制代码	0	使用二进制数据通讯。
ASCII 代码	1	使用 ASCII 数据通讯。

(b) 关于数据通讯代码的更多详细说明，请参阅第 3.2 节，“通讯用的数据代码”。

* 括号中的数值(地址: $\square\square\square\square_H \dots b00$)表示缓冲存储器地址和存储从 GX Developer 输入的设定值的以太网模块位置。

关于缓冲存储器的地址，请参见 3.8 节，“缓冲存储器的应用和分配列表”。

(续上页)

地址 十进制 (十六进制)	应用	名称	初始值 (十六进制)	GX Developer 设置适用性	参考章节
203 (Cb) $\square\square\square\square_H$	模块状态区 通讯状态存储区	用 GX Developer 设置的状态 <ul style="list-style-type: none"> 通讯数据代码设置(b1) <ul style="list-style-type: none"> 0: 以二进制代码进行通讯 1: 以 ASCII 代码进行通讯 初始化/开放方法设置(b2) <ul style="list-style-type: none"> 0: 无参数设置(按照程序启动) 1: 参数设置(按照程序设置) TCP 存在确认设置(b4) <ul style="list-style-type: none"> 0: 使用 Ping 1: 使用 KeepAlive 发送设置(b5) <ul style="list-style-type: none"> 0: 以太网帧 1: IEEE802.3 帧 使用 TCP 协议进行 CPU 通讯时序的设置(b6) <ul style="list-style-type: none"> 0: 禁止在运行时写入 1: 允许在运行时写入 被动开放系统(b8) <ul style="list-style-type: none"> 0: 不等待打开(不能在停止时间通讯) 1: 始终等待打开(能够在停止时间通讯) 非以上的位只供系统使用。	0 _h	0	4.7 节
204 (Cc)	发送/接收 指令区	系统区	-	-	-
205 (Cd)		RECV 指令执行请求	0 _h	×	应用篇的 第 4 章
206 (Ce)		系统区	-	-	-
207 (Cf)		ZNRD 指令的执行结果	0 _h	×	应用篇的 第 4 章
208 (Cg)		系统区	-	-	-
209 (Ch)		ZNRW 指令的执行结果	0 _h	×	应用篇的 第 4 章
210 至 223 (D2 至 Df)		系统区	-	-	-

(转下页)

关于通用术语和缩写

除非另外规定，本手册中使用下列通用术语和缩写来说明型号 QJ71E71-100、QJ71E71 和 QJ71E71-B2 的以太网接口模块。

通用术语/缩写	说明	
ACPU	AnNCPU、AnACPU 和 AnUCPU 的通用术语。	
AnACPU	A2ACPU、A2ACPU-S1、A2ACPUP21/R21、A2ACPUP21/R21-S1、A3ACPU、A3ACPUP21/R21 通用术语	
AnNCPU	A1NCPU、A1NCPUP21/R21、A2NCPU、A2NCPU-S1、A2NCPUP21/R21、A2NCPUP21/R21-S1、A3NCPU、A3NCPUP21/R21 的通用术语	
AnUCPU	A2UCPU、A2UCPU-S1、A2ASCPU、A2ASCPU-S1、A3UCPU 和 A4UCPU 的通用术语	
以太网地址	机器专有地址也称为 MAC (媒体存取控制) 地址。该地址用来识别网络以外的外部设备地址。 以太网模块的地址可以通过铭牌上的 MAC ADD 栏来进行核实。	
以太网模块	型号 QJ71E71-100、QJ71E71-B5 和 QJ71E71-B2 的以太网接口模块的缩写 (描述为以太网模块或 E71)。	
以太网网络系统	10BASE2、10BASE5、10BASE-T 和 100BASE-TX 网络系统中使用的缩写	
外部设备	通过以太网或数据通讯连接的个人计算机、计算机、工作站 (WS) 和以太网模块等的通用术语。	
GX Developer	GX Developer (SWnD5C-GPPW-E) 的缩写 (型号名称中的 n 是大于等于 4 的数字)	
MELSECNET/10	MELSECNET/10 网络系统的缩写	
MELSECNET/H	MELSECNET/H 网络系统的缩写	
MX Component	MX Component (SWD5C-ACT-E 或以后的版本) 的缩写	
网络模块 (N/W 模块)	与 MELSECNET/10 (H) 网络系统兼容的接口模块的缩写	
OPS	使用冗余系统兼容 EZSocket (操作站) 安装相关产品的通用术语。 OPS 可以使用用于 OPS 连接的以太网模块用户连接进行通讯。(参阅 5.5 节)	
个人计算机	IBM PC/AT (或 100% 兼容) 个人计算机的通用术语	
QCPU	Q 模式	Q00JCPU、Q00CPU、Q01CPU、Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU、Q12PHCPU、Q25PHCPU、Q12PRHCPU 和 Q25PRHCPU 的通用术语
QCPU-A	A 模式	Q02CPU-A、Q02HCPU-A 和 Q06HCPU-A 的通用术语
QCPU 站	安装了 QCPU 的 PLC 所使用的缩写	
QnACPU	Q2ACPU、Q2ACPU-S1、Q2ASCPU、Q2ASCPU-S1、Q2ASHCPU、Q2ASHCPU-S1、Q3ACPU、Q4ACPU 和 Q4ARCPU 的通用术语	
Q/QnA	QCPU 和 QnACPU 的通用术语	
参考手册	Q 系列 MELSEC 通讯协议参考手册的缩写	
用户手册 (应用篇)	Q 系列以太网接口模块用户手册的缩写 (应用篇)	
用户手册 (基础篇)	Q 系列以太网接口模块用户手册的缩写 (基础篇)	
用户手册 (Web 功能篇)	Q 系列以太网接口模块用户手册的缩写 (Web 功能篇)	

产品配置

下表列出了以太网接口模块的产品配置。

型号	项目名称	数量
QJ71E71-100	QJ71E71-100 以太网接口模块	1
QJ71E71-B5	QJ71E71-B5 以太网接口模块	1
QJ71E71-B2	QJ71E71-B2 以太网接口模块	1

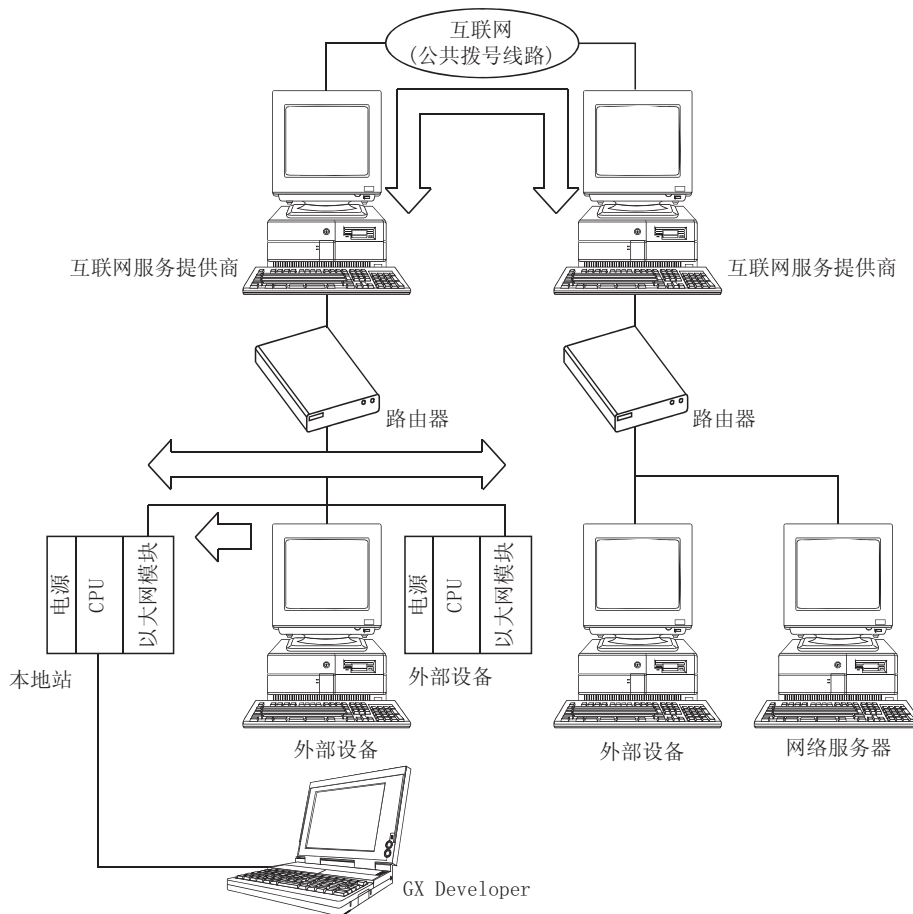
1 概述

本手册简要说明了 QJ71E71-100、QJ71E71-B5 和 QJ71E71-B2 以太网接口模块的规格信息(以下简称为以太网模块)，以及开始操作之前的步骤，与外部设备进行通讯的控制步骤和数据通讯方法、维护、检查和故障排除。

在实际系统中应用以下程序范例时，一定要检查其适用性并确认不会引起系统控制问题。

1.1 以太网模块的概述

以太网模块是 PLC 侧连接 Q 系列 PLC 与本站系统的接口模块，如个人计算机和工作站，也是通过以太网(10BASE-TX、10BASE-T、10BASE5、10BASE2)使用 TCP/IP 或 UDP/IP 通讯协议在 PLC 之间的接口模块。



- 1) PLC CPU 数据的收集和修改
(使用 MELSEC 通讯协议进行通讯)
- 2) 将任意数据传送到外部设备和接收来自外部设备的任意数据
(使用固定缓冲存储器或随机访问缓冲存储器进行通讯)
- 3) 通过电子邮件传送/接收数据
(使用电子邮件功能时)
- 4) 通过 Web 功能传送/接收数据
(仅用户手册(Web 功能))

* 使用 GX Developer (SW4D5C-GPPW-E 或以后的版本，以下简称为 GX Developer)，可以大大简化通讯所用的顺控程序。

1.2 以太网模块的特性

(1) 使用 MELSEC 通讯协议进行数据通讯 (详见 MELSEC 通讯协议参考手册的第 6 章中的说明)

在“使用 MELSEC 通讯协议 (以下简称 MC 协议) 进行数据通讯”中, 可以从本站系统读取 PLC 的软元件数据和程序文件, 或将 PLC 的软元件数据和程序文件写入本站系统。

该协议是按照本站系统的请求单独进行数据通讯的被动协议。在建立 1 号连接之后, 它不需要数据通讯的顺控程序。

如果本站系统是运行下列基本操作系统之一的个人计算机, 则有可能为本站系统创建一个顺控程序, 而不用考虑使用下列单独出售的通讯支持工具的具体 MC 协议 (传送/接收步骤)。

(支持的基本操作系统)

- Microsoft® Windows® 95 操作系统
- Microsoft® Windows® 98 操作系统
- Microsoft® Windows NT® Workstation 4.0 操作系统
- Microsoft® Windows® Millennium Edition 操作系统
- Microsoft® Windows® 2000 Professional 操作系统
- Microsoft® Windows® XP Home Edition 操作系统
- Microsoft® Windows® XP Home Edition 操作系统

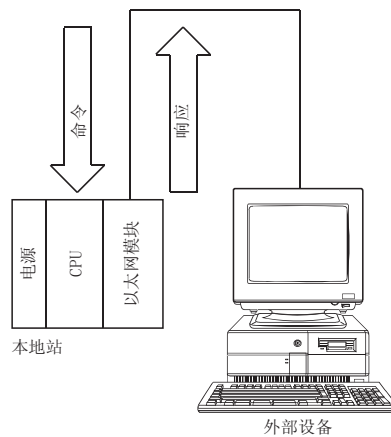
* 根据使用 MX Component 的不同版本, 支持不同的操作系统。

请参阅 MX Component 手册中的具体情况。

(通讯支持工具另售)

• MX Component (SW0D5C-ACT-E 或以后的版本, 以下简称为 MX Component)

* 有关 MX Component 的概述请参阅附录 9。



备注

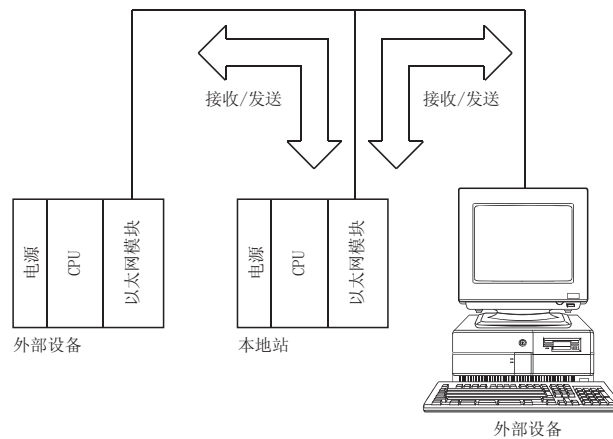
对 PLC CPU 进行读/写数据的通讯功能由 A/QnA 系列以太网模块 (A1SJ71E71/A1SJ71QE71 等) 支持, 此通讯功能与使用 MC 协议的通讯功能相对应。

(2) 使用固定缓冲存储器进行通讯(第 7 章和第 8 章中作了详细说明)

在“使用固定缓冲存储器进行通讯”中，PLC 之间或 PLC 和本站系统之间可以发送或接收最多 1k 字的任意数据。

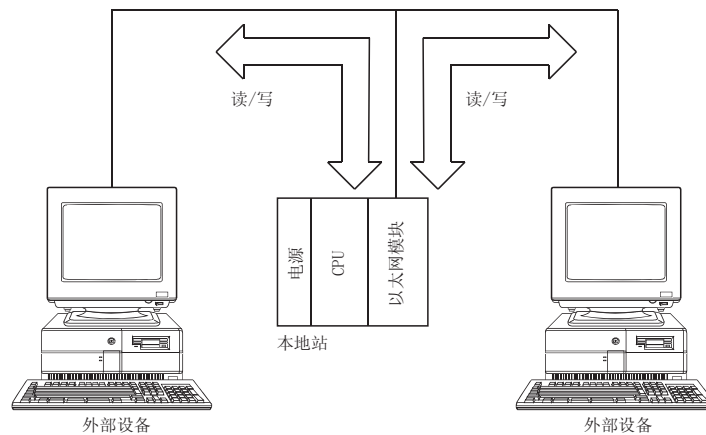
以太网模块提供 16 个 1k 字存储空间的固定缓冲存储器数据区，可以指定每一个缓冲存储器发送或接收任意软件的数据。

当使用的 MC 协议通讯为被动通讯时，使用固定缓冲存储器进行通讯是一个主动协议。当机器设备中出错，或当满足相同条件时，可以从 PLC 端向本站系统发送数据。而且，通过使用中断程序中的数据接收功能，可以加速恢复 PLC CPU 接收的数据。

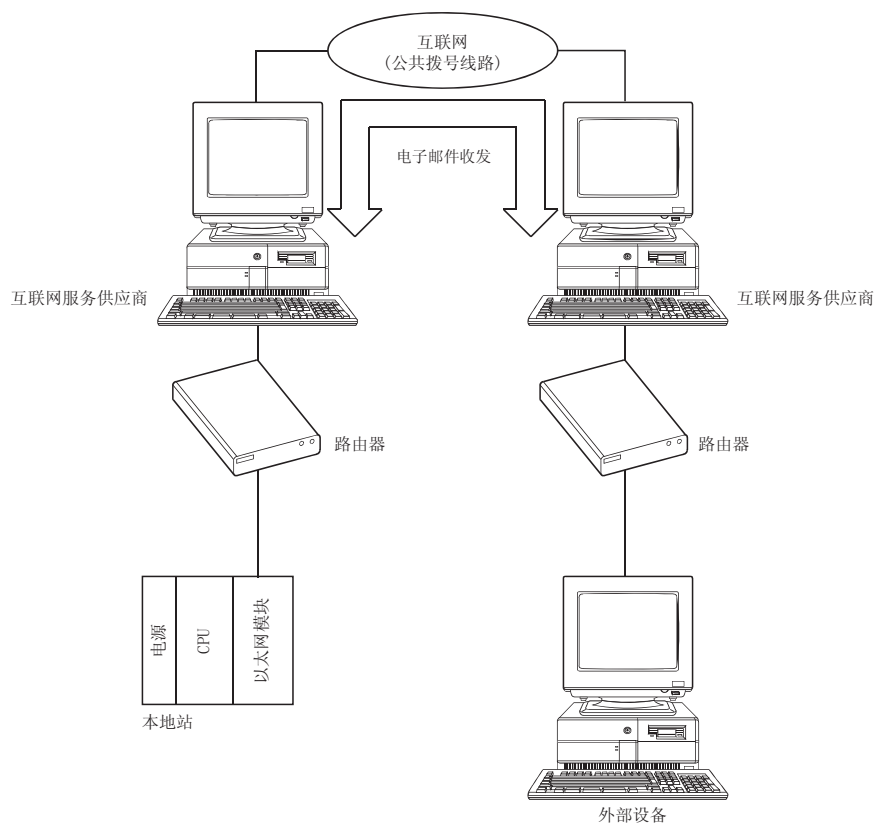


(3) 使用随机访问缓冲存储器进行通讯(第 9 章中作了详细说明)

在“使用随机访问缓冲存储器进行通讯”中，可以进行较大量数据(6k 字以下的数据)通讯。对于使用固定缓冲存储器进行通讯来说，过大的数据可以使用该协议(约达 1k 字的数据)。



- (4) 通过电子邮件进行通讯(用户手册(应用篇)中作了详细说明)
“收发电子邮件”时,可以在远程区域通过互联网、使用电子邮件功能把数据发送给外部设备,或者接收外部设备的数据。
- (a) 通过 PLC CPU 收发电子邮件
可以使用专用指令(MSEND、MRECV)发送/接收下列数据。
- 1) 作为附件发送/接收数据
6K 字以下的文件可以作为电子邮件的附件,发送到个人计算机或其它以太网模块中,或者从个人计算机或其它以太网模块中接收。
 - 2) 作为正文发送数据
960 个字以下的的数据可以作为电子邮件的正文发送给个人计算机或便携式终端。
- (b) 通过 PLC CPU 监视功能发送电子邮件
通过设置以太网参数,可以以恒定时间间隔来监视用户设置的通知条件(PLC CPU 状态或软元件值),并且在满足通知条件时,可以用下列方法之一来发送 960 个字以下的的数据:
- 1) 作为附件发送数据
 - 2) 作为正文发送文件



(5) 使用 Web 功能进行通讯(用户手册(Web 功能篇)中作了详细说明)
 “使用 Web 功能进行通讯”时，系统管理员可以通过互联网在远程区域使用市售的网络浏览器监视 Q 系列 CPU。

(a) 在网络服务器中安装一个通讯库，就可以与 PLC 进行数据通讯。
 网络浏览器显示的示例屏也可以供你使用。
 请与当地的代理商或销售部联系。

(b) 网络服务器和网络浏览器需要使用 Web 功能。

(基本操作系统)

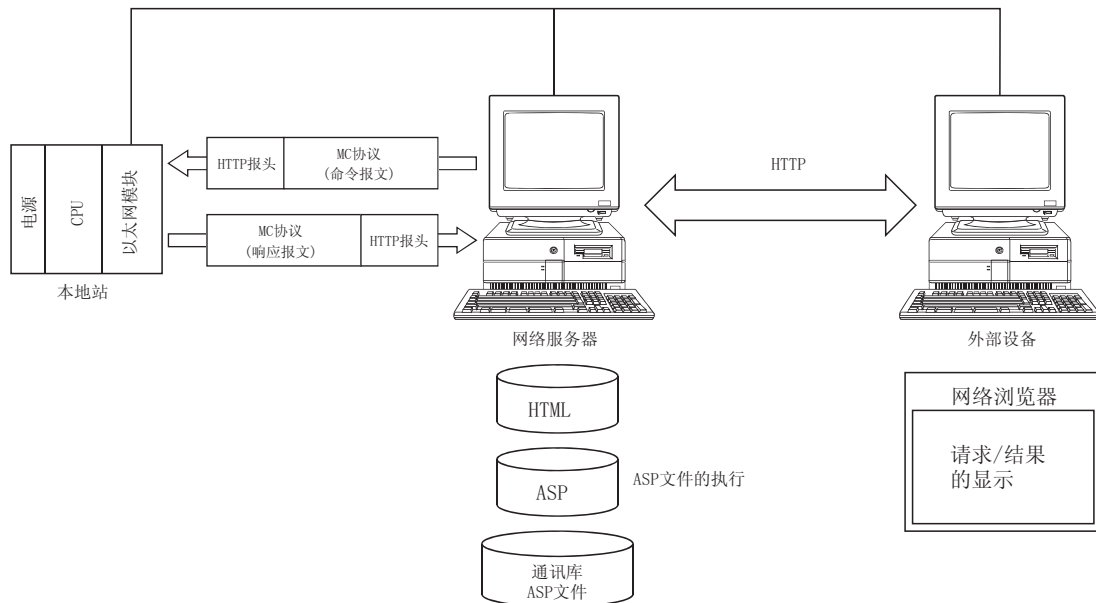
- Microsoft® Windows® 2000 Server 操作系统
- Microsoft® Windows® 2000 Professional 操作系统
- Microsoft® Windows NT® Server Network 4.0 操作系统
- Microsoft® Windows NT® Workstation 4.0 操作系统
- Microsoft® Windows® 98 操作系统

(网络服务器)

- Microsoft® Internet Information Server 5.0
- Microsoft® Internet Information Server 4.0
- Microsoft® Peer Web Services 4.0
- Microsoft® Personal Web Server 4.0

(网络浏览器)

- Internet Explorer 4.0 或以后的版本(Microsoft® 公司)
- Netscape® Communicator 4.05 或以后的版本(Netscape® 通讯公司)



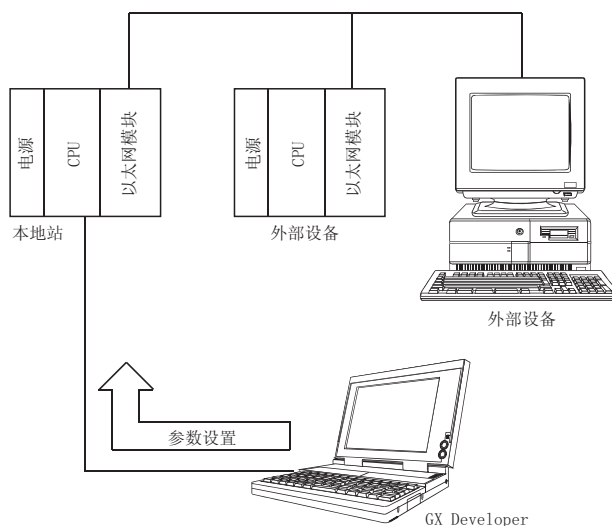
(6) 使用 GX Developer、GT SoftGOT 和 GOT (GX Developer 和 GT SoftGOT 的操作手册中及 GOT 的用户手册 (连接系统手册) 中作了详细说明)

(a) 使用 GX Developer 简化顺控程序

GX Developer 支持外部设备执行以太网模块初始化和开放处理的参数设置功能。通过使用 GX Developer 的“网络参数设置”来设置下列参数，能够从外部设备访问 PLC。它也可以大大地简化执行以太网模块通讯所使用的顺控程序。

- IP 地址设置
- 端口编号设置
- 协议类型设置
- 通知条件设置

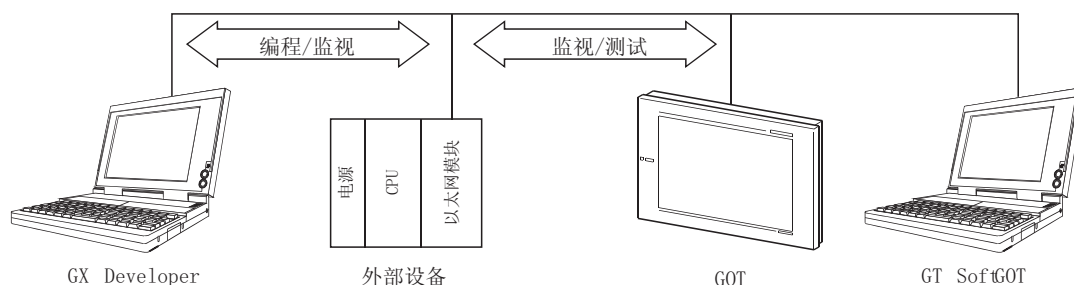
关于设置以太网模块的更多信息，参阅第 3.6 节，“以太网模块的 GX Developer 设置项目列表”和其它适用的参考章节。



(b) 通过以太网的编程和监视功能

通过建立以太网连接，可以使用 GX Developer 执行 PLC 的编程和监视，并可以使用 GOT 或 GT SoftGOT 执行 PLC 的监视和测试。

在上述的任一种情况下，都可利用长距离连接和以太网高速通讯进行远程操作。



(c) 连接多个 MELSOF 产品 (GX Developer、GT SoftGOT 和 MX Component) 或 GOT 该产品可以通过 TCP/IP 通过或 UDP/IP 通讯同时连接一个或多个 MELSOF 产品 (GX Developer、GX SoftGOT 和 MX Component) 或 GOT。*1 *2

1) 通过 TCP/IP 通讯进行连接

- 通过 TCP/IP 通讯, 使用一个专用的系统连接和最多 16 个的用户连接, 以太网模块端最多可以同时连接 17 个 MELSOF 产品。*3
- 如果只连接一个 MELSOF 产品, 则不需要使用 GX Developer 进行下列设置。

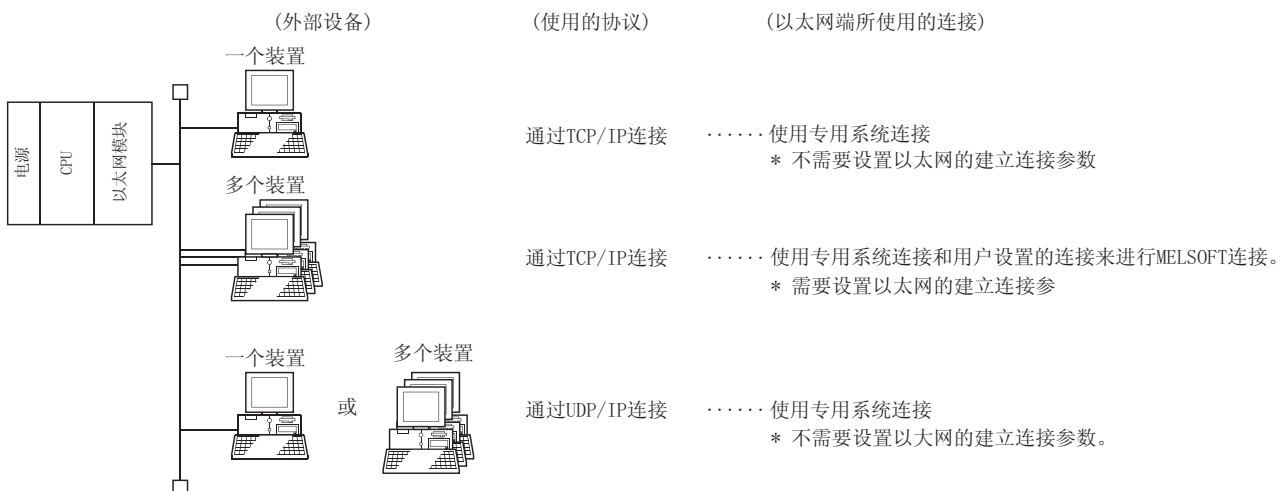
如果连接两个或更多的 MELSOF 产品, 由于需连接使用用户, 因此需要使用 GX Developer 进行下列设置。

在网络参数的“以太网开放设置”屏上, 在要使用的连接号的协议字段中设置“TCP”, 在开放方法字段中设置“MELSOF 连接”。(参阅第 5.5 节)

2) 通过 UDP/IP 通讯进行连接*4

使用一个专用的系统连接, 通过 UDP/IP 通讯, 以太网模块端可以连接 MELSOF 产品或 GOT。

[当连接 MELSOF 产品或 GOT 时, 以太网模块侧使用的连接]



*1 如果启动个人计算机中的两个或多个 MELSOF 产品以执行与以太网模块的 TCP/IP 通讯和 UDP/IP 通讯时, 可以使用相同的站号。(TCP/IP 和 UDP/IP 不必使用不同的站号)

*2 GT SoftGOT 和 GOT 只支持 UDP/IP 通讯。

*3 专用连接用来连接 MELSOF 产品进行数据通讯。除 MELSOF 产品以外, 这些专用连接不能用来与外部设备进行数据通讯。

*4 如果使用与 MELSECNET/H 或 MELSECNET/10 继电器通讯的以太网模块, 从 MELSOF 产品中访问其它站, 可以无需进行转换设置。(网络号、站号和 IP 地址)

关于 MELSECNET/H 和 MELSECNET/10 的继电器功能, 请参阅用户手册(应用篇)的第 3 章。

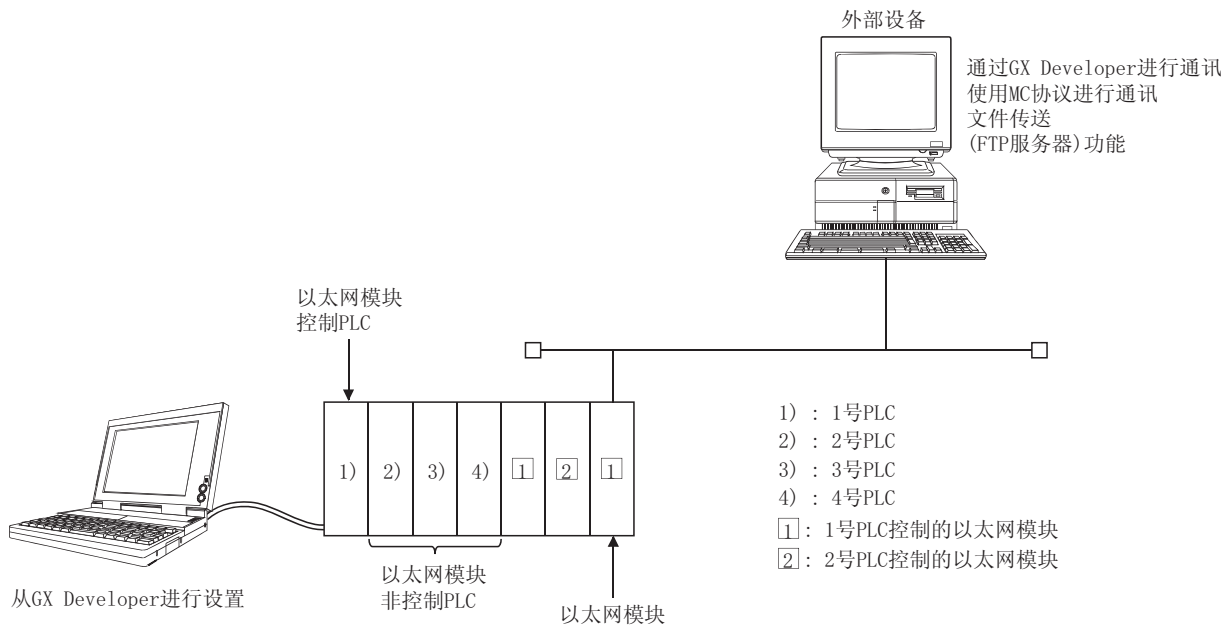
(7) 支持多 CPU 系统的功能(参考手册中作了详细说明。)

(a) 当与多 CPU 系统中的 QCPU 进行下列数据通讯时, 可以通过指定要访问的 QCPU, 执行数据通讯(如读/写数据或读/写文件)。

- 1) 使用 MC 协议进行通讯
- 2) 使用 GX Developer 进行通讯
- 3) 使用文件传送功能(FTP 服务器)时

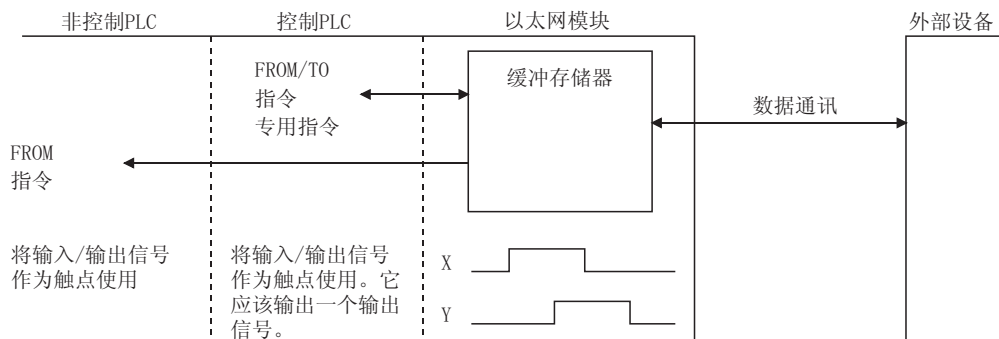
* 当使用多 CPU 系统中的以太网模块时, 应通过 GX Developer 指定控制以太网模块的 QCPU(以下称为控制 PLC)。

可以将功能版本 A 的以太网模块安装在多 CPU 系统中, 也可以访问唯一的控制 PLC(1 号 PLC)。



(b) 当在多 CPU 系统中使用功能版本 B 的以太网模块时, 可以用以太网模块执行下列数据通讯。

- 1) 使用固定缓冲存储器进行通讯时, 可以在控制 PLC 上使用数据链接指令通讯和收发电子邮件。
- 2) 从非控制 PLC 读取缓冲存储器也是可能的。可以将输入/输出信号作为触点使用。

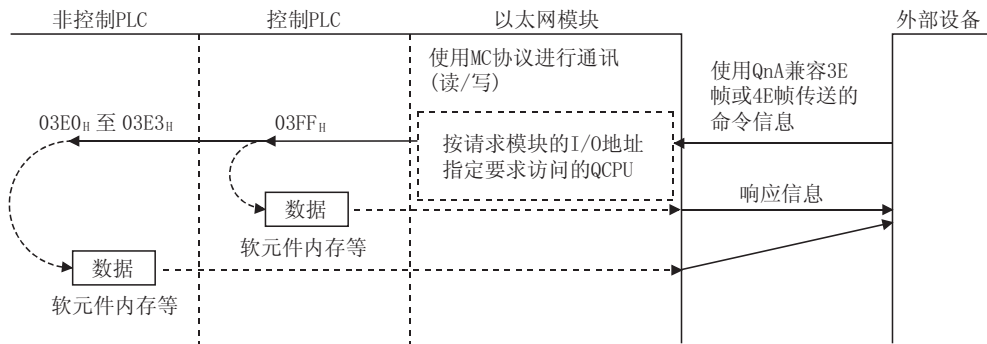


3) 可以使用 MC 协议和通过 GX Developer 和文件传送 (FTP 服务器) 从外部设备的互联网访问控制 PLC 和非控制 PLC。

另外, 对于以太网模块控制 PLC 来说, 也可能使用固定缓冲存储器进行通讯和收发电子邮件。

(范例)

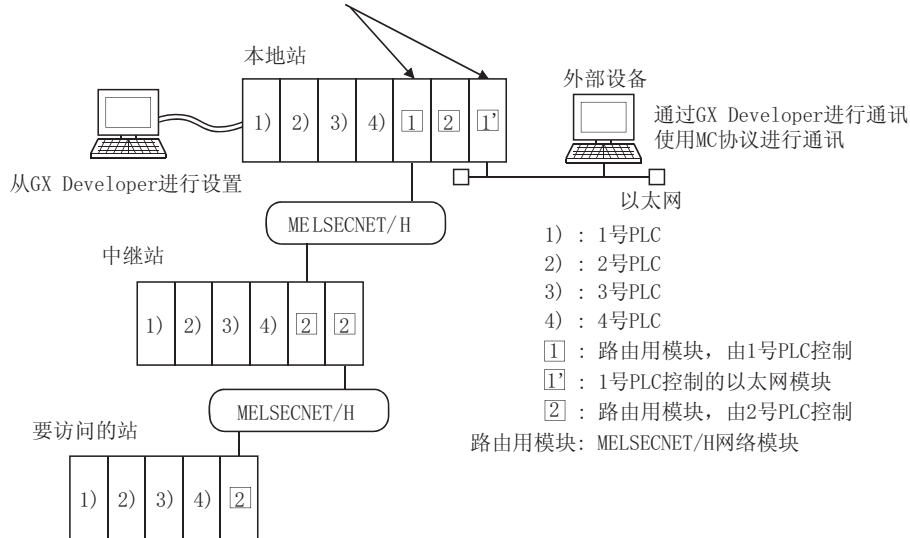
通过 MC 协议进行通讯时



如果使用 MC 协议或 GX Developer 通过互联网访问其它站, 即使中继站和互联网的站是多 CPU 系统, 也可以通过互联网访问该站的控制 PLC 和非控制 PLC。

(范例)

不论路由用模块的控制 PLC 是否相同, 都可以访问其它站。



* 当使用 MC 协议进行通讯时, 应使用 QnA 兼容 3E 帧或 4E 帧访问非控制 PLC。

但是要注意可用的功能会根据访问的 QCPU 不同而不同。(是控制 PLC 或是非控制 PLC)。

关于可用功能和可访问范围请参阅参考手册。

* 当访问其它站时, 路由用模块指以下模块:

- MELSECNET/H、MELSECNET/10 网络模块
- Q 系列 C24
- 以太网模块

* 如果在路由用模块当中有功能版本 A 的模块, 也可以只访问路由用模块的控制 PLC。另外, 也可以通过同一控制 PLC 控制的模块访问其它站。

(8) 远程口令核对功能(参考手册的第 5 章中作了详细说明。)

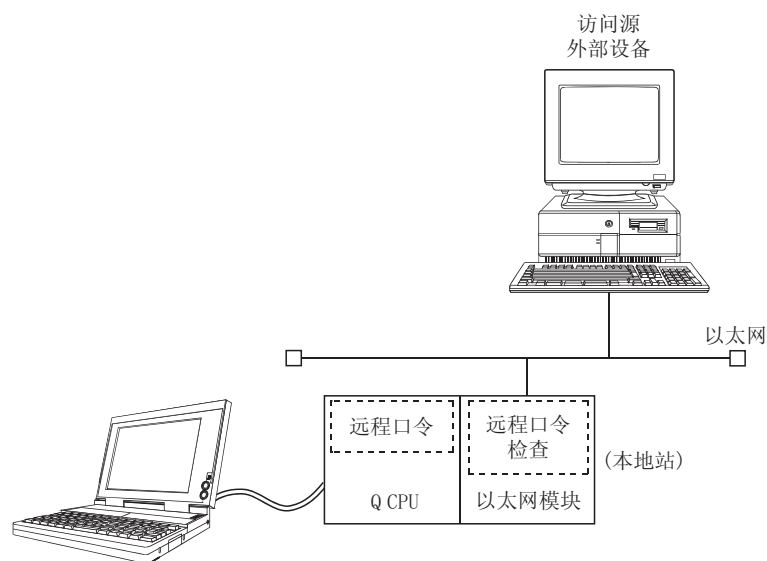
- (a) 以太网模块远程口令核对功能可防止用户在远程区域对 QCPU 进行不适当的访问。

针对从外部设备(使用作为 QCPU 中参数来设置的连接)进行的数据通讯,以太网模块执行远程口令核对。

* 远程口令功能是 QCPU 防止其它用户对 QCPU 进行不适当访问的功能之一。

当使用 GX Developer 设置 QCPU 的远程口令时,可以使用 QCPU 的远程口令功能。

- (b) 在进行数据通讯期间,从外部设备对本站或其它站的所有远程口令核对,都是对设置在本站 QCPU 中的远程口令执行的。
- (c) 当使用远程口令核对的适用连接进行数据通讯时,在完成开放处理之后,可以通过远程口令解锁(删除)处理执行来自外部设备的数据通讯。
- 1) 当使用 MC 协议、固定缓冲存储器或随机访问缓冲存储器时使用 MC 协议通讯的专用指令解锁来自外部设备的远程口令。
 - 2) 通过 GX Developer 访问 PLC 时
在线操作开始时,通过 GX Developer 解锁远程口令。
 - 3) 使用文件传送(FTP 服务器)功能时
使用专用的 FTP 命令解锁来自外部设备的远程口令。
 - 4) 使用 Web 功能时
当访问 QCPU 时,通过网络浏览器解锁显示在对话框中的远程口令。
在关闭连接之前,用户执行远程口令锁定处理。



从 GX Developer 设置
用参数设定远程口令检查的适用连接。

(9) 冗余系统支持功能(详细说明:第5章)

(a) 以太网通讯的备份

通过在冗余系统的主基板上安装功能版本 D 或以上版本的以太网通讯,可以备份以太网模块。如果以太网系统中出现故障或通讯出错,系统可以在控制系统和待机系统间进行切换,以保证以太网模块的连续通讯。*1

(b) 向控制系统 CPU 发出系统切换指令

如果安装在冗余系统中控制系统 CPU 主基板上的以太网模块检测到通讯错误或断开时,会向控制系统 CPU 发出系统切换指令。

(c) 访问冗余系统

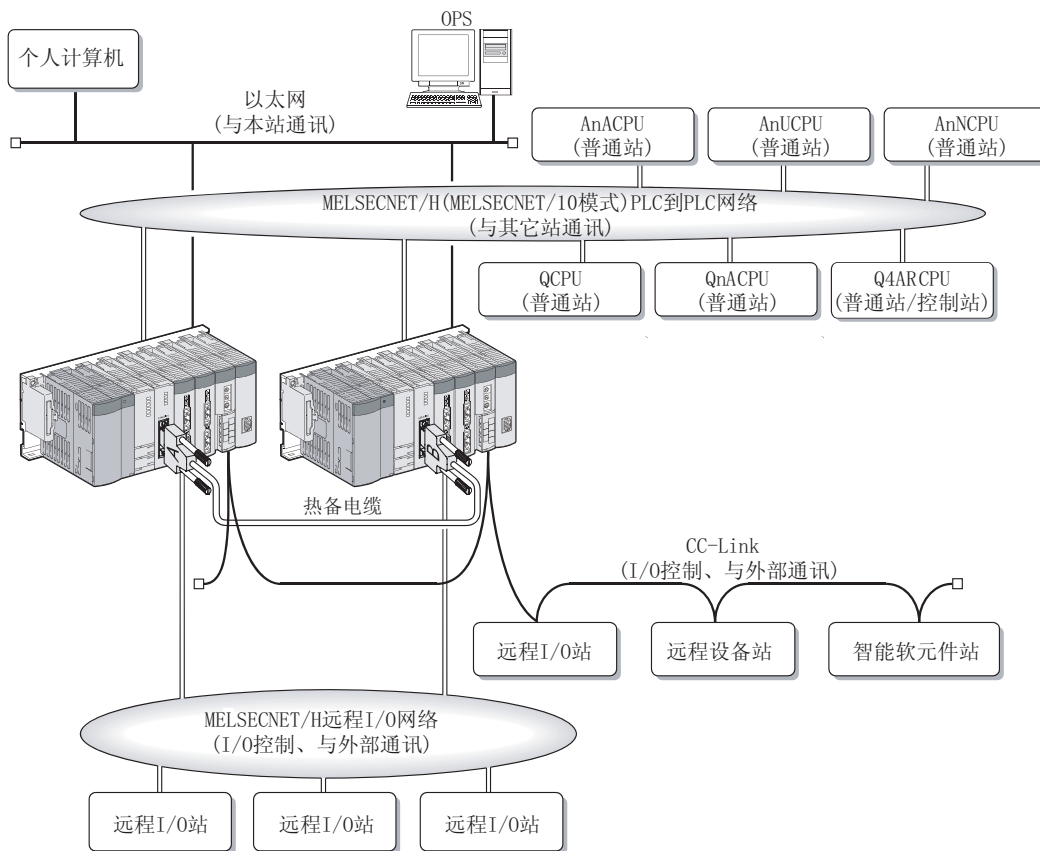
使用 MC 协议或数据链接指令的通讯可以进行数据通讯,例如:从冗余系统的控制系统或系统 A 读取软元件数据和文件或写入软元件数据和文件到冗余系统的待机系统或系统 B。

(d) 与 OPS 的连接

以太网模块可以使用 OPS 连接的用户连接与 OPS 进行通讯。

在 GX Developer 的开放设置中设置用于 OPS 连接的用户连接。(参阅第 5.5 节)

*1 模块也可以安装在冗余系统的 MELSECNET/H 远程 I/O 站上。但是,可使用的功能有所限制,所以在 MELSECNET/H 远程 I/O 站上安装模块时,请参阅 2.6 节。



1.3 功能版本 B 或以上版本中新增的功能

下面说明了功能版本 B 以太网模块中新增的功能。

对于可以使用新增功能的以太网模块和相关产品 (CPU 模块、GX Developer) 的功能版本、系列号和软件版本，参阅第 2.7 节。

对于以太网模块功能版本的功能比较，参阅附录 1.1。

(1) 功能版本 B 新增的功能

功能	概要	参考章节
支持 IEEE802.3 帧	使用 IEEE802.3 帧发送/接收数据 • 发送数据时，可以使用以太网帧或 IEEE802.3 帧选择和发送数据。 • 接收数据时，可以使用以太网帧或 IEEE802.3 帧选择和接收数据。	附录 9 第 4.7 节
以太网模块的重新初始化处理	在以下情况下进行重新初始化处理 • 对进行数据通讯的外部设备更新其以太网地址时。 • 更改数据通讯条件时。	5.2.3 节
存在确认功能	使用 KeepAlive 确认 通过发送 ACK 信息到远程设备来进行目标存在确认，并等待察看是否接收到响应。	5.2.2 节备注(5)
通过 TCP/IP 通讯最多同时连接 17 个 MELSOFT 产品	通过 TCP/IP 通讯最多同时连接 17 个 MELSOFT 产品 (如 GX Developer)。这可以通过使用最多 16 个用户连接和一个专用的系统连接得以实现。	适用 MELSOFT 产品手册的第 1.2(6) 和 5.5 节
简化同 MELSOFT 产品的连接	简化访问其它站 如果通过以太网模块从 MELSOFT 产品中访问其它站，可以不进行转换设置。(网络号、站号和 IP 地址)	每个 MELSOFT 产品手册的第 1.2(6) 节
	访问相同的站号 如果启动个人计算机中的两个或多个 MELSOFT 产品以进行与以太网模块的 TCP/IP 通讯和 UDP/IP 通讯时，可以使用相同的站号。	
GX Developer 的以太网诊断功能	GX Developer 的以太网诊断功能可以进行以下诊断： • 监视以太网模块的状态和出错状态。 • 通过执行 Ping 测试和环路回路回送测试，确认已完成以太网模块的初始化处理。	第 5.4 和 11.2 节
使用电子邮件功能时	作为附件发送 CSV 格式的文件 将 CSV 格式的文件作为附件，从以太网模块发送到电子邮件	用户手册(应用篇)的第 2 章
	发送正文 从以太网模块发送 960 个字以下的正文。	
	支持编码/解码 支持以下编码/解码： • 使用 7 位解码标题并从以太网模块发出。 • 解码并接收使用以太网模块 Quoted Printable 编码的电子邮件。	
使用 Web 功能进行通讯	使用市售的网络浏览器，通过互联网在远程位置用个人计算机访问 PLC。	用户手册(Web 功能篇)
远程口令核对	功能概要 在通过使用 QCPU 的参数开放的数据通讯中，该功能允许在解锁处理正常完成后，从外部设备设置 QCPU 中数据通讯的远程口令。	第 5.9 节
	使用 MC 系统进行通讯 文件传送(FTP 服务器)功能 解锁/锁定 QCPU 的远程口令。解锁处理可以允许使用各种命令访问 QCPU。	参考手册 3.18 节 用户手册(应用篇) 5.6 节
	使用 Web 功能进行通讯 访问 QCPU 时，通过网络浏览器显示的对话框解锁远程口令。	用户手册(Web 功能篇)
支持多 CPU 系统	使用 MC 协议进行通讯	参考手册的第 2.10 节
	从 GX Developer 访问 QCPU 文件传送(FTP 服务器)功能 与多 CPU 系统进行数据通讯时，该功能允许访问用户指定的控制 PLC/非控制 PLC。	GX Developer 操作手册 用户手册(应用篇)的第 5 章
	安装以太网模块到 MELSECNET/H 远程 I/O 站	第 2.6 节
安装以太网模块到 MELSECNET/H 远程 I/O 站	当以太网模块安装到 MELSECNET/H 远程 I/O 站时，允许从外部设备访问本站(安装了以太网模块的站)或其它站。	第 2.6 节

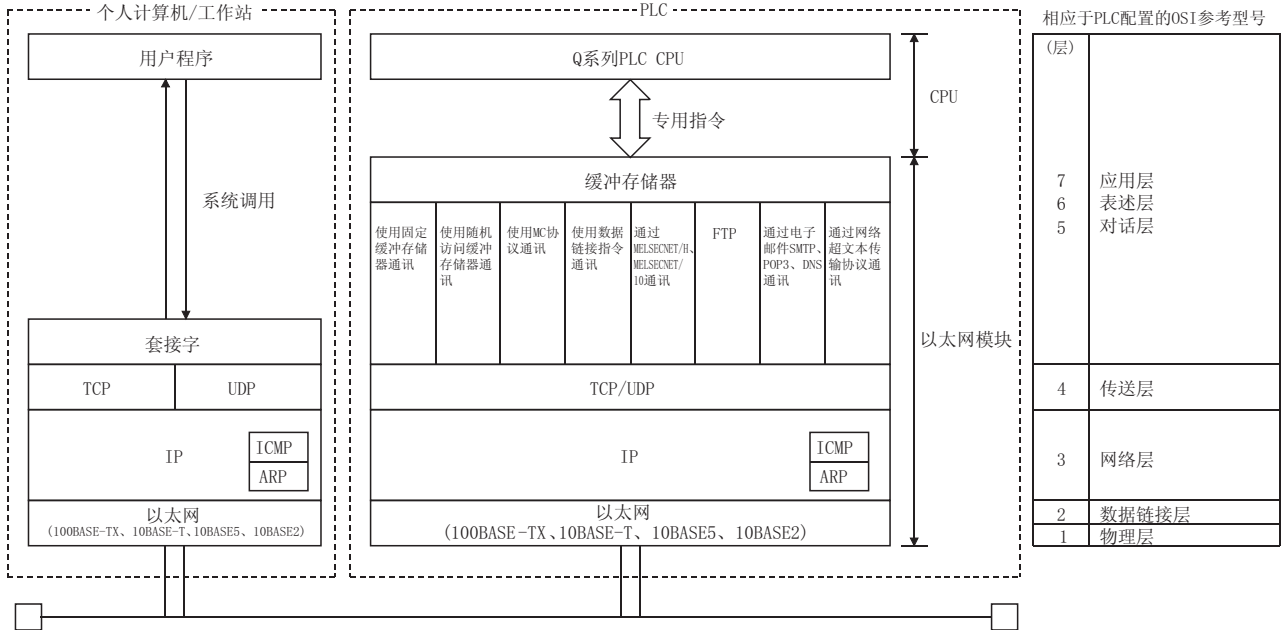
(2) 功能版本 D 新增的功能

功能	概要	参考章节	
冗余系统支持功能	<ul style="list-style-type: none"> 在冗余系统的主基板上可以安装并使用以太网模块。(冗余以太网通讯) 当检测到通讯错误或断开时,可以向控制系统 CPU 发出系统切换请求。 	第 5.11 节	
网络集线器连接状态监视功能	在缓冲存储器中,可以检查以太网模块和网络集线器的连接状态以及当前传输速度。	第 5.10 节	
使用电子邮件功能时	通过 PLC CPU 监视功能发送电子邮件正文中的字符串	PLC CPU 监视功能可以传送电子邮件正文存储在字软元件中的字符串。	用户手册(应用篇)的第 2 章
用于数据链接指令的目标站 CPU 类型	使用以下数据链接指令可以指定目标站 CPU 类型((S1)+3)。(可以指定冗余系统的控制系统/待机系统或系统 A/系统 B) <ul style="list-style-type: none"> READ/SREAD WRITE/SWRITE REQ 	用户手册(应用篇)的第 4 章、第 6 章	
数据链接指令增加的数据长度(从 480 到 960 字)	在以下数据链接指令中数据长度最多指定为 960 字: * ¹ <ul style="list-style-type: none"> SEND RECV/RECVS READ/SREAD WRITE/SWRITE 		
通过与 4E 帧兼容的 MC 协议通讯	对添加到用于信息识别的 QnA 兼容 3E 帧中任何给出编号(系列号)的信息格式(4E 帧)都兼容。 通过用于信息识别的外部设备添加系列号,可以检查多指令和响应信息间的通信。	参考手册的第 3 章	

*1 在多网络系统中,当传送到其它网络号的数据超过 480 字时,对所有的请求源、中继站和请求目标指定为 Q 系列型号。

1.4 软件配置

以太网模块支持 TCP/IP 和 UDP/IP 协议。



(1) TCP (传输控制协议)

该协议保证个人计算机/工作站和通过网络连接的 PLC 之间的通讯中数据的可信性和可靠性，并提供下列功能：

- 通过开放 (逻辑线路) 创建一个逻辑连接，就好像在外部设备之间建立一条专用线路。
- 最多可以建立 16 个连接，并且可以同时以太网模块中与多个缓冲存储器进行通讯。
- 通过使用序号、数据再次传送功能和校验的步骤控制来保持数据的可靠性。
- 可以使用 Windows 的操作来控制通讯数据流。
- 支持 Maximum Segment 选项。

对 TCP 传送或 TCP 重新传送，Maximum Segment 选项可以设置为有效。处理信息前，数据接收端必须检查接收数据的长度。

(2) UDP(用户数据包协议)

该协议不能保证个人计算机/工作站和通过网络连接的 PLC 之间的通讯中数据的可信性和可靠性。因此,即使数据没有到达目标节点,也不会再次传送。

- 因为它是无接点的,所以能高速传送。
- 通过校验增加通讯数据的可靠性。然而,当必须保持较大的可靠性时,必须利用用户应用程序或 TCP。

(3) IP(网际协议)

- 以数据报格式发送和接收通讯数据。
- 可以分割和重新汇编通讯数据。
- 不支持路由选项。

(4) ARP(地址解析协议)

- 该协议用于从 IP 地址获得以太网物理地址。

(5) ICMP(网间控制报文协议)

- 该协议用于交换 IP 网络上发生的各种错误和与网络相关的各种信息。
- 提供传送 IP 错误信息的功能。
- 关于选项支持类型(ICMP 协议)的信息,参阅附录。

(6) FTP(文件传送协议)

- 该协议用于传送文件。
- 可以上传和下载 PLC CPU 文件。

(7) DNS(域名系统)

- 该系统将 IP 地址翻译成用户容易记住的名字。

(8) SMTP(简单邮件传送协议)

- 该协议传送邮件。

(9) POP3(邮局协议版本 3)

- 该协议将邮件服务器收到的邮件传送给本地计算机。

(10) HTTP(超文本传送协议)

- 该协议用于执行全球网络的数据通讯。

2 系统配置

本章说明可以与以太网模块组合的系统配置。

2.1 可应用系统

以太网模块可以与下列系统一起使用：

(1) 适用的模块和可以安装的模块数

下表列出了 CPU 模块和可安装在远程 I/O 站的以太网模块以及可以安装的模块数。

适用模块	可以安装的模块数	备注
CPU 模块	Q00JCPU Q00CPU Q01CPU	最多 1 个 (* 1)
	Q02CPU Q02HCPU Q06HCPU Q12HCPU Q25HCPU	最多 4 个 只可以安装在 Q 模式中。 (* 1)
	Q12PHCPU Q25PHCPU	(* 1) (* 4)
	Q12PRHCPU Q25PRHCPU	(* 1) (* 4)
网络模块	QJ72LP25-25 QJ72LP25G QJ72LP25GE QJ72BR15	最多 4 个 MELSECNET/H 远程 I/O 站 (* 2) (* 3)

*1 参考将要使用的 CPU 模块的用户手册 (功能解说、程序基础)。

*2 参考 Q 系列 MELSECNET/H 网络系统参考手册 (远程 I/O 网络)。

*3 使用功能版本 B 或以上版本的以太网模块。

*4 使用功能版本 D 或以上版本的以太网模块。

(2) 适用的基板

以太网模块可以安装在基板的任意 I/O 槽对 (*1) 中。

*1 限定 CPU 模块的远程 I/O 点和 I/O 远程站的范围。

(用于远程 I/O 站)

(3) 多 CPU 系统

使用多 CPU 系统中的以太网模块时，请先参考 QCPU 用户手册 (多 CPU 系统)。

(a) 适用以太网模块

使用多 CPU 系统中的以太网模块时，使用功能版本 B 或以上版本的以太网模块。

(b) 网络参数

网络参数只可以写入到以太网模块的控制 PLC 中。

(4) 适用软件包

(a) 用于 PLC 的软件(*1)

下表列出了适用于以太网模块的系统和软件包。
使用以太网模块时，必须使用 GX Developer。

		软件版本
		GX Developer
Q00J/Q00/Q01CPU	单 CPU 系统	版本 7 或以上版本
	多 CPU 系统	版本 8 或以上版本
Q02/Q02H/Q06H/ Q12H/Q25HCPU	单 CPU 系统	版本 4 或以上版本
	多 CPU 系统	版本 6 或以上版本
Q12PH/Q25PHCPU	单 CPU 系统	版本 7.10L 或以上版本
	多 CPU 系统	
Q12PRH/Q25PRHCPU	冗余系统	版本 8.18U 或以上版本
安装到 MELSECNET/H 远程 I/O 站时		版本 6 或以上版本

*1 关于支持已改进以太网模块新增功能的 GX Developer 版本，请参阅 2.7 节。

(b) 外部设备的通讯支持工具

项目名	型号	备注
MX Component	SWnD5C-ACT-E	Active X 控制库。 型号名中的“n”表示 0 或以上数字。(*1)

*1 根据使用的 MX Component 版本不同，支持不同版本的以太网模块。
关于详情，请参阅 MX Component 手册。

2.2 网络配置需要的设备

本节说明了配置网络需要的设备。

网络安装工作需要足够的安全措施；安装时请网络专家进行指导。

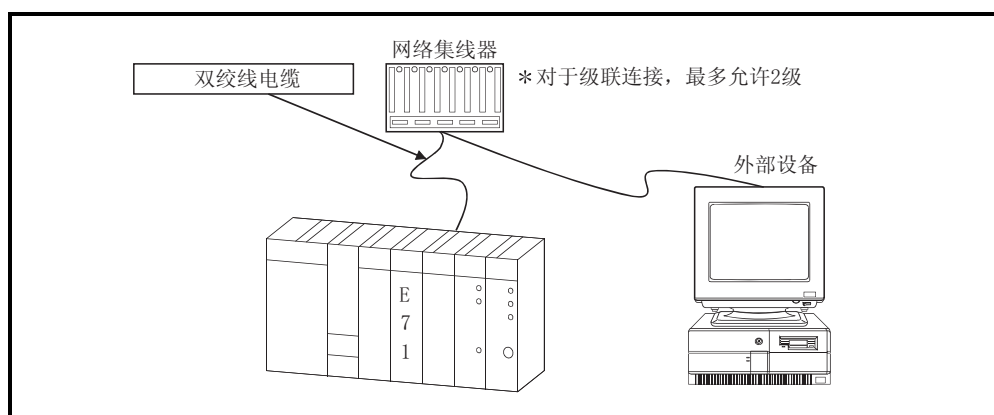
(1) 用 QJ71E71-100 配置以太网系统时

将 QJ71E71-100 连接到网络时，可以使用 10BASE-T，或者使用 100BASE-TX。

以太网模块根据网络集线器来检测是 10BASE-T 或是 100BASE-TX，是全双工或是半双工传送模式。

连接到无自动检测功能的网络集线器时，在网络集线器端设置为半双工传送模式。

(a) 使用 100BASE-TX 连接

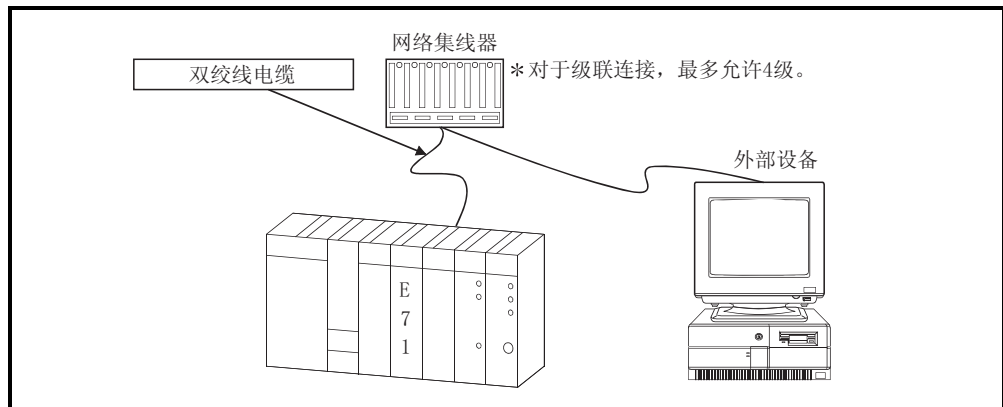


使用满足 IEEE802.3 和 100BASE-TX 标准的设备。

(上图中网络集线器下面的设备)

- 双绞线屏蔽电缆 (STP 电缆)，种类 5
 - * 可以使用直线电缆。(如果通过以太网模块的 10BASE-TX，使用交叉电缆连接到外部设备，则不能保证正确操作。但是，可以使用交叉电缆来连接两个以太网模块 (例如：两个 QJ71E71-100 模块)，用来进行数据通讯或连接以太网模块到 GOT。)
- RJ45 插头
- 100Mbps 网络集线器

(b) 使用 10BASE-T 连接



使用满足 IEEE802.3 和 100BASE-T 标准的设备。

(上图中网络集线器下面的设备)

- 双绞非屏蔽电缆 (UTP) 或双绞屏蔽电缆 (STP), 种类 3、4、5

* 可以使用直线电缆。

(如果通过以太网模块的 10BASE-TX, 使用交叉电缆连接到外部设备, 则不能保证正确操作。但是, 可以使用交叉电缆来连接两个以太网模块 (例如: 两个 QJ71E71-100 模块), 用来进行数据通讯或连接以太网模块到 GOT。)

- RJ45 插头
- 10Mbps 网络集线器

要点

通过 100BASE-TX 连接进行高速通讯 (100Mbps) 期间, 在给定的安装环境中由于高频噪声的影响, 可能使通讯出错。

以下就是针对 QJ71E71-100 端网络系统结构的对策。

(1) 接线

- 不要将双绞线电缆与主电路和电源线捆扎在一起, 也不要安装在将它们相互靠得太近。
- 一定要将双绞线电缆敷设在排线管中。

(2) 通讯方法

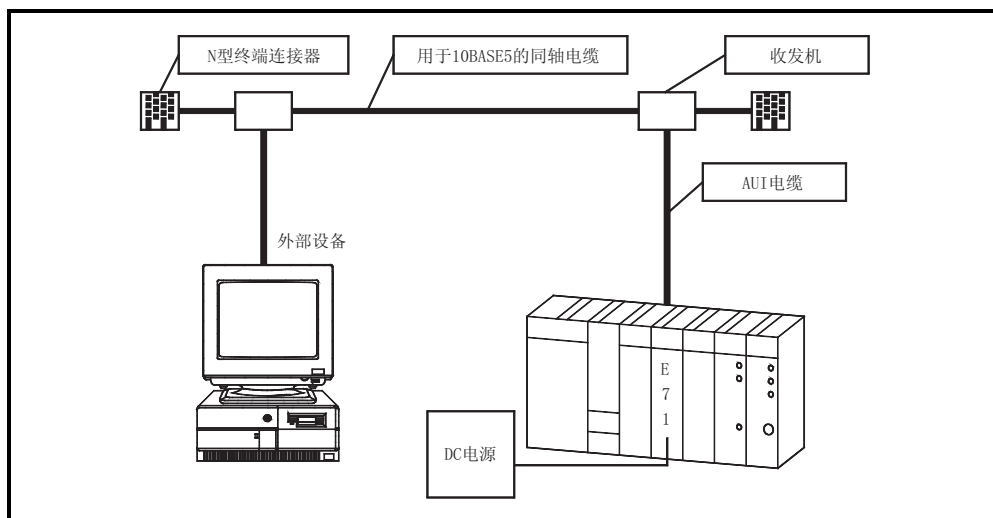
- 使用 TCP/IP 通讯与外部设备进行数据通讯。
- 需要时, 增加通讯重试次数。

(3) 10Mbps 通讯

- 将 QJ71E71-100 的网络集线器改为处理能力达 10Mbps 的网络集线器, 从而以 10Mbps 的数据传送速率进行通讯。

(2) 用 QJ71E71-B5 配置以太网系统时

(a) 使用 10BASE5 连接



- 1) 使用符合以太网标准的 10BASE5 同轴电缆、N 型终端连接器、收发器和 AUI 电缆 (收发器电缆)。
- 2) 使用具有 SQETEST (信号质量错误测试) 或心搏功能的收发器。
- 3) 使用符合收发器和 AUI 电缆规格的 DC 电源 (用于收发器的电源)。(参阅备注)

备注

收发机的电源特性如下：

- 输入终端电压：12V^{-6%}至 15V^{+5%}
- AUI 电缆直接电阻：40Ω/km 或以下，最大长度 50m (164ft.)
- 最大电流消耗：500mA 或以下

因此，适用的收发机电源在 13.28V 到 15.75V。

* 收发机电源电压 (V) 的计算。

$$\text{电压降 (V)} = \text{AUI 电缆直接电流电阻 } (\Omega/\text{m}) \times \text{AUI 电缆长度 (m)} \times 2 (\text{两个方向}) \times \text{收发机消耗电流 (A)}$$

(举例)

$$2.0 (\text{V}) = 0.04 (\Omega/\text{m}) \times 50 (\text{m}) \times 2 \times 0.5 (\text{A})$$

在这种情况下，建议收发机电源为 13.28V 以上。

$$13.28 (\text{V}) = 12 \text{ V}^{-6\%} (11.28\text{V}) + 2.0 (\text{V})$$

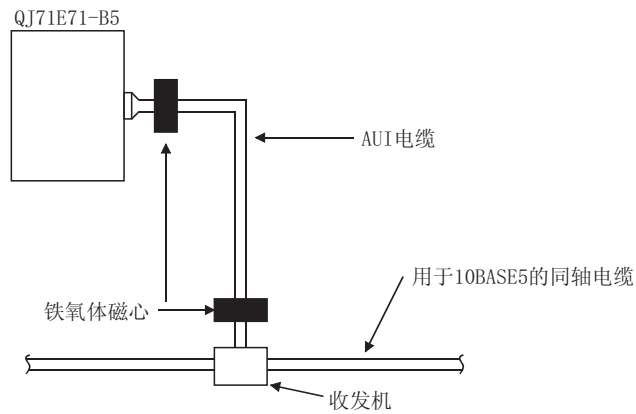
要点

- (1) 关于所需设备，请咨询网络专家。
- (2) 根据安装的环境，以下方法可以用来解决由于高频率噪音所引起的出错。
 - 使用以下(3)中显示的方法安装铁氧体磁心。
 - 使用 TCP/IP 通讯时增加重试次数。
- (3) 通过 10BASE5 连接到网络时，使用以下方法安装铁氧体磁心。

(安装铁氧体磁心)

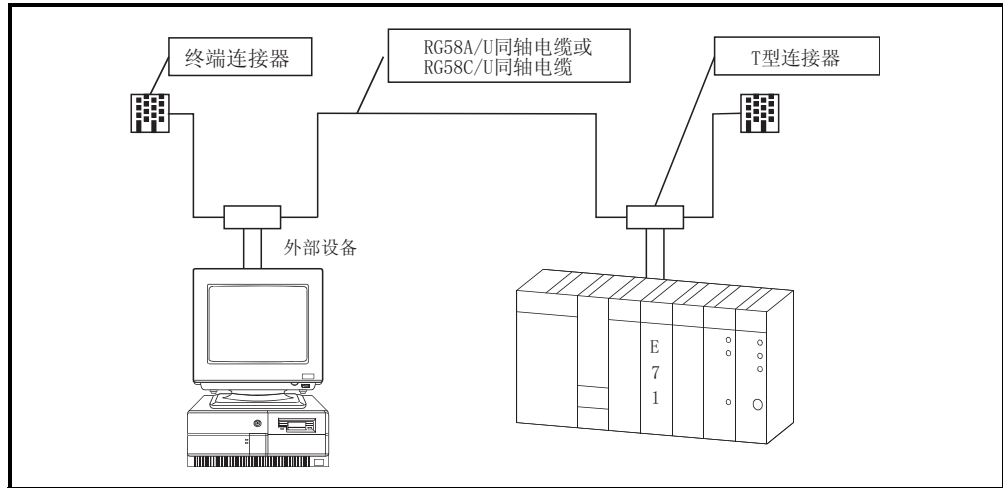
安装铁氧体磁心(*1)到用于 AUI 电缆的以太网模块端和外部设备端/收发机端。

*1 可以使用 TDK 公司生产的 ZCAT2032-0930。



(3) 用 QJ71E71-B2 配置以太网系统时

(a) 使用 10BASE2 连接



使用符合 IEEE802.3 和 10BASE2 标准的设备。

- RG58A/U 或 RG58C/U (50Ω同轴电缆)
- BNC 型终端连接器 (相当于 Tyco Electronics AMP K.K. 制造的 221629-4 产品)
- T 形适配器 (相当于 Hirose 电子公司制造的 UG-274/U (1S) 产品)

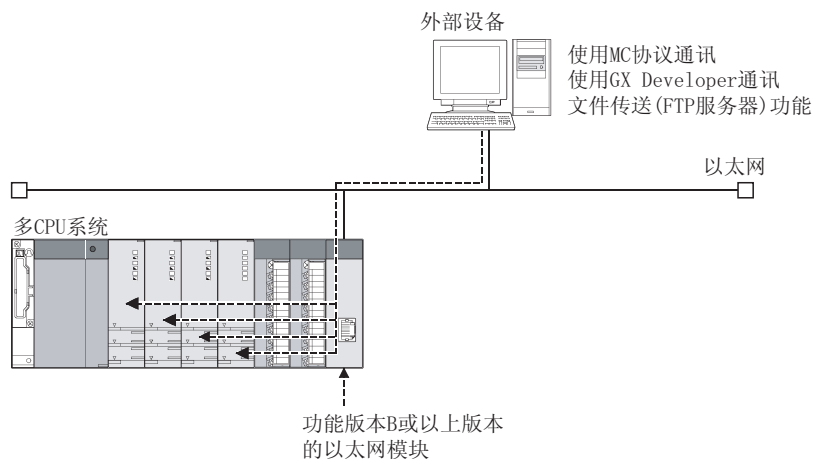
要点

有关需要的设备，请向网络专家咨询。

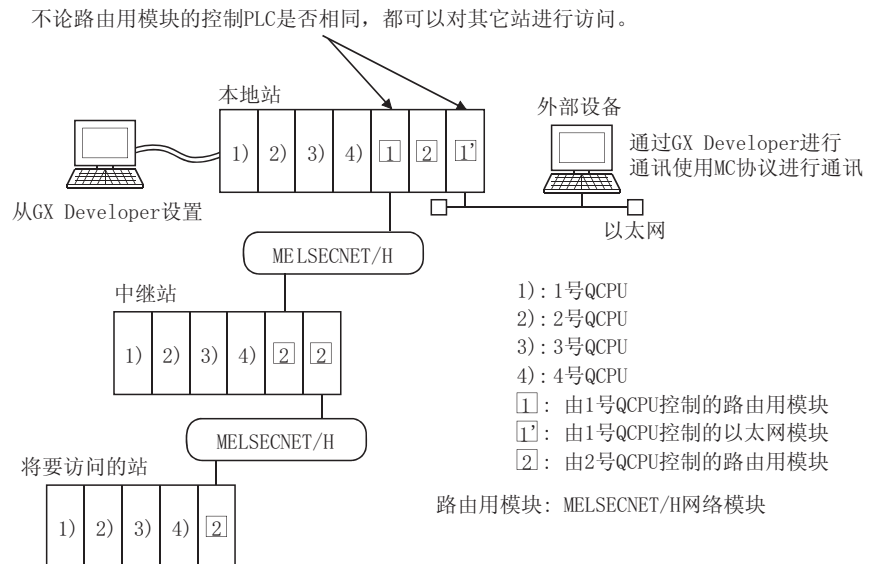
2.3 用于多 CPU 系统中

本节说明了多 CPU 系统中使用以太网模块。

- (1) 当从与以太网模块连接的设备访问以太网模块的非控制 CPU 时，应使用功能版本 B 或以后版本的以太网模块，使用下列所示功能
当使用功能版本 A 的以太网模块时，只可以访问控制 CPU。（访问非控制 CPU 会导致出错。）
- 使用 MC 协议通讯
 - 使用 GX Developer 通讯
 - 文件传送 (FTP 服务器) 功能



- (2) 如果要访问的其它站属于多 CPU 系统，用于路由、本地站处的 QCPU、所有中继站和访问站的模块应是功能版本 B 或以上版本的模块。所有情况下，在访问各站的路由用模块的非控制 CPU 时都应保持如此。*1
(范例)



- *1 当访问其它站时，路由用模块可以访问下列模块。
- MELSECNET/H、MELSECNET/10 网络模块
 - Q 系列 C24 • 以太网模块

2.4 与 Q00J/Q00/Q01CPU 一起使用时

本节说明与 Q00J/Q00/Q01CPU 一起使用时的以太网模块。

(1) 可应用的功能

下表列出了当以太网模块安装在 Q00J/Q00/Q01CPU 的主基板中时，可以使用的功能。

功能	可用性
使用 MC 协议通讯	○(*1)
使用固定缓冲器进行通讯	○
通过中断程序进行接收处理	○(*3)
使用随机访问缓冲器进行通讯	○
发送/接收电子邮件	○(*2)
使用数据链接指令进行通讯	△(*2)
通过中断程序进行接收处理	○(*3)
文件传送(FTP 服务器功能)	○
使用 Web 功能通讯	○
MELSECNET/H、MELSECNET/10 中继通讯	○
路由中继通讯(路由中继功能)	○
外部设备存在确认	○
成对开放通讯	○
通过自动开放 UDP 端口进行通讯	○
同时广播	○
支持 QCPU 远程口令功能	○(*3)
通过 GX Developer 设置以太网参数	○
用 GX Developer 访问 QCPU(TCP/IP 或 UDP/IP)	○

○：可用 △：使用受限制 ×：不可用

*1 关于处理时间，参阅参考手册。

根据数据通讯所使用的帧，可以访问的软元件及其范围是不同的。

*2 当 SREAD/SWRITE 指令的目标站是 Q00J/Q00/Q01CPU 时，忽略到目标站的通知软元件(D3)。

SREAD/SWRITE 指令的操作与 READ/WRITE 指令的操作相同。

关于 SREAD/SWRITE 指令，请参阅用户手册(应用篇)的第 4 章或第 6 章。

*3 可用于 Q00J/Q00/Q01CPU 功能版本 B 或以上版本。

要使用此功能，GX Developer 必须是版本 8 或以上版本。

2.5 与 Q12PRH/Q25PRHCPU 一起使用时

本节说明了以太网模块和 Q12PRH/Q25PRHCPU 一起使用。

(1) 系统配置

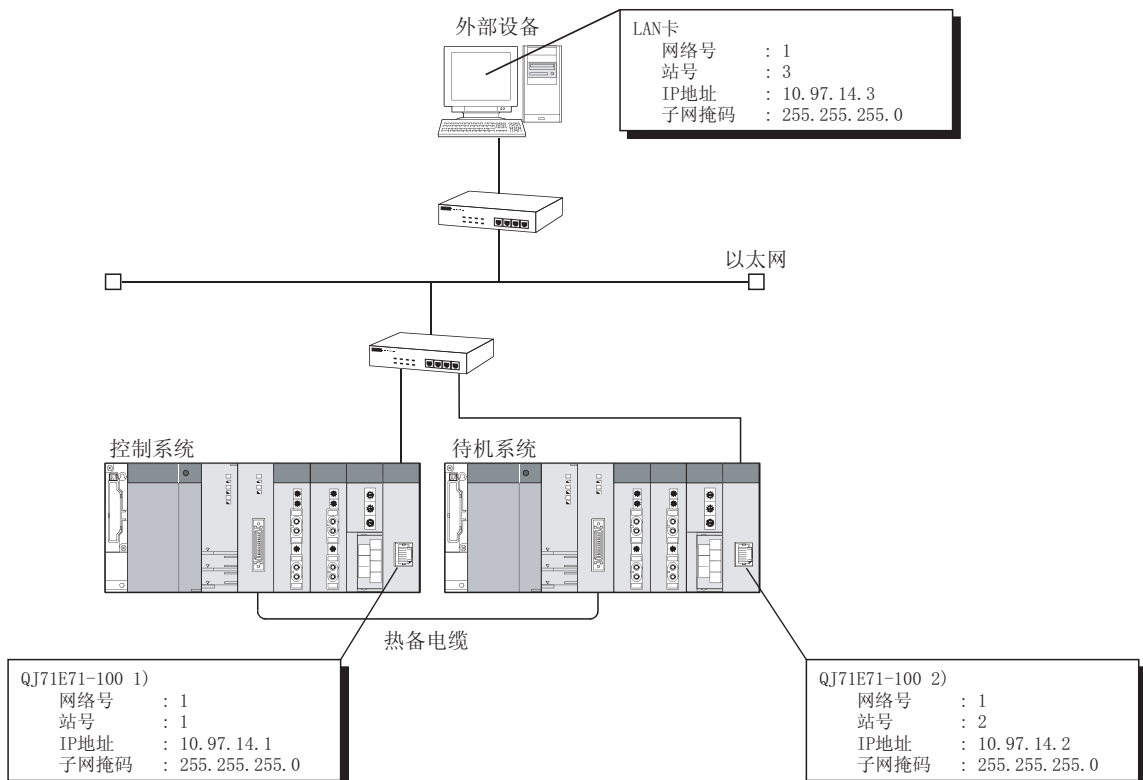
以下显示冗余系统的配置。

(a) 冗余系统的基本配置

以下显示冗余系统的基本系统配置。

外部设备可以访问冗余系统的控制系统/待机系统。

(使用冗余系统配置网络时, 请参阅 5.11 节。)



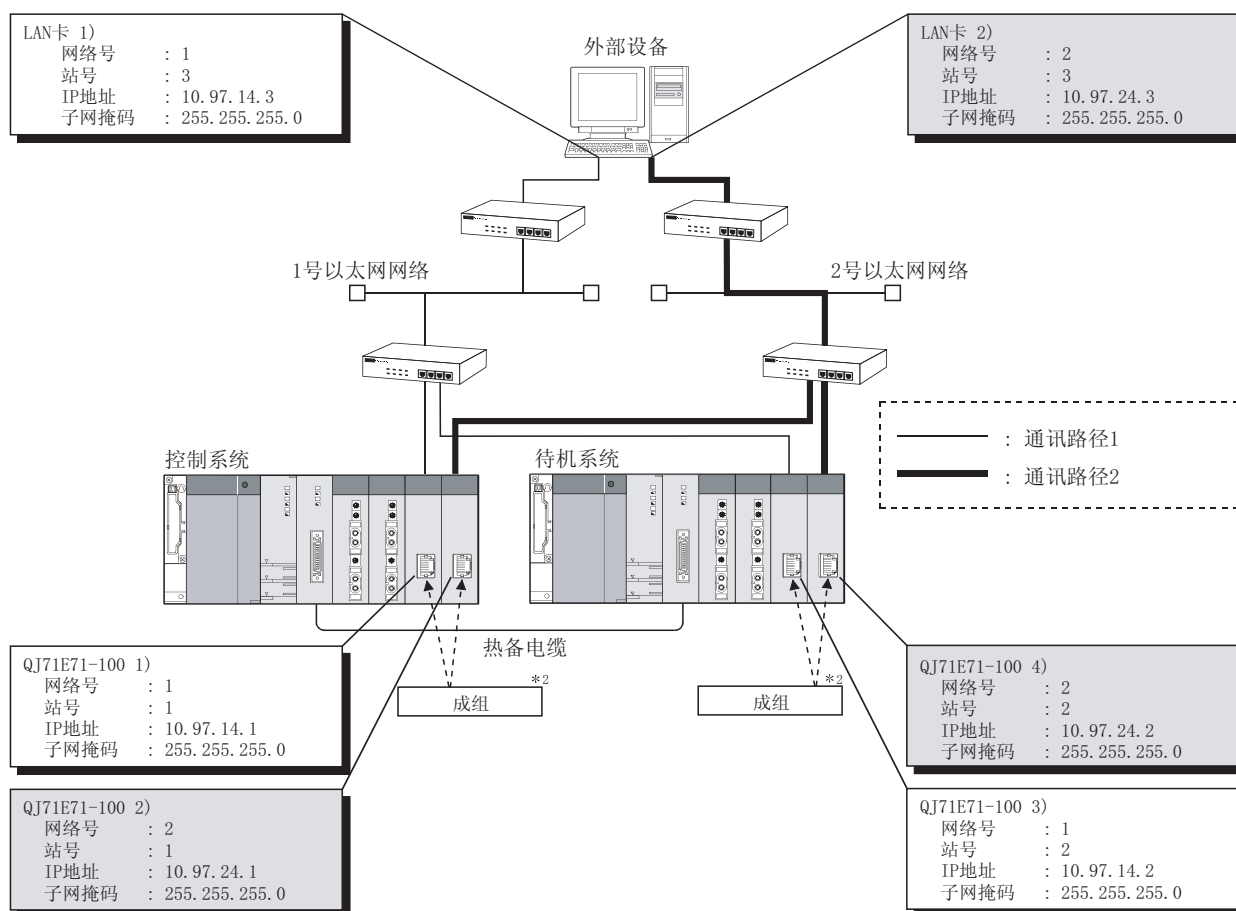
(b) 具有备份通讯路径的系统配置 *¹

以下显示带有成对通讯路径(备份)的系统配置, 该路径可以连接外部设备和以太网模块。

外部设备可以访问通讯路径 1 或通讯路径 2 中冗余系统的控制系统/待机系统。

(举例) 如果在访问通讯路径 1 中的控制系统时发生通讯错误, 则可以访问通讯路径 2 中的控制系统。*²

如果又在通讯路径 2 中发生通讯错误, 则在控制系统与待机系统间进行系统切换, 这样新控制系统就可以继续进行通讯。*³



*¹ 在外部设备中使用两个 LAN 卡时, 在不同的子网地址中使用以实现用子网掩码设置 IP 地址。

*² 进行 GX Developer 上的“网络模块冗余组设置”。
(关于“网络模块冗余组设置”, 参阅 QnPRHCPU 用户手册)
当两个成组的以太网模块都出故障时, 进行系统切换。

*³ 进行 GX Developer 的冗余设置, 以设置在检测到通讯错误或断开时是否从以太网模块发送系统切换请求。(参阅 5.11.3 节)

(2) 可用功能

以太网模块安装在 Q12PRH/Q25PRHCPU 的主基板时的可用功能如下：

功能	可用性
使用 MC 协议进行通讯	△
使用固定缓冲器进行通讯	△
使用随机访问缓冲器进行通讯	△
发送/接收电子邮件	△
使用数据链接指令进行通讯	△
文件传送(FTP 服务器功能)	△
使用 Web 功能进行通讯	○
MELSECNET/H、MELSECNET/10 中继通讯	△
路由中继通讯(路由中继功能)	○
外部设备存在确认	○
成对开放通讯	○
通过自动开放 UDP 端口进行通讯	○
同步广播	△
对 QCPU 远程口令功能的支持	○
通过 GX Developer 设置以太网参数	○
使用 GX Developer 访问 QCPU(TCP/IP 或 UDP/IP)	○

○：可用 △：使用受限制 ×：不可用

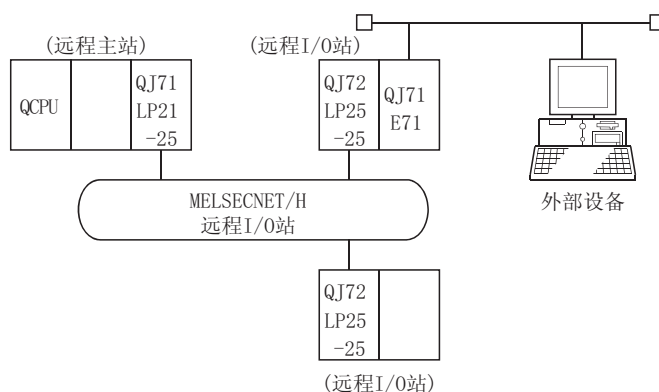
*1 关于使用 Q12PRH/Q25PRHCPU 功能的注意事项，请参阅 5.11.5 节。

2.6 在 MELSECNET/H 远程 I/O 站上使用

本节大致说明了在 MELSECNET/H 网络中的远程 I/O 站中使用以太网模块的各种目的。如果以太网模块安装在 QCPU 站上，则没有必要阅读本节。

(1) 系统配置

(举例)



(2) 可用功能

当以太网模块安装在 MELSECNET/H 远程 I/O 站时，可使用下列功能。

功能		可用性
初始化处理	顺控程序	×
	GX Developer 参数设置	○(参见(3)(4))
开放/关闭处理	顺控程序	×
	GX Developer 参数设置	○(参见(3)(4))
使用 MC 协议进行通讯		○(* ¹)
使用固定缓冲存储器进行通讯		×
使用随机访问缓冲器进行通讯		○
发送/接收电子邮件		×
使用数据链接指令进行通讯		×(可以中继)
文件传送(FTP 服务器)		×
使用 Web 功能进行通讯		×
MELSECNET/H、MELSECNET/10 中继通讯		○
路由中继通讯(路由中继功能)		○
外部设备存在确认		○
成对开放通讯		×
通过自动开放 UDP 端口进行通讯		○
同步广播		×
对 QCPU 远程口令功能的支持		○(* ²)
通过 GX Developer 设置以太网参数		○(参见(3))
用 GX Developer 访问 QCPU(TCP/IP 或 UDP/IP)		○

○：可用 ×：不可用

*1 以下说明使用 MC 协议访问 MELSECNET/H 远程 I/O 站及通过 MELSECNET/H 远程 I/O 站访问其它站。

(a) 使用 QnA 兼容 3E 帧进行通讯。(用 A 兼容 1E 帧不可以进行通讯)

- (b) 以下功能可用于 MELSECNET/H 远程 I/O 站。对于 QnA/A 系列兼容 MELSECNET/10 远程 I/O 站，只可以读/写智能功能模块缓冲存储器。

可用功能	功能
软元件存储器读/写	批量读和批量写
	随机读取、测试(随机写)
	监视数据寄存、监视
	多块批量读，多块批量写
缓冲存储器读/写	从以太网模块的缓冲存储器读取/写入到以太网模块的缓冲存储器中
读/写智能功能模块的缓冲存储器	对指定的智能功能模块的缓冲存储器读/写。

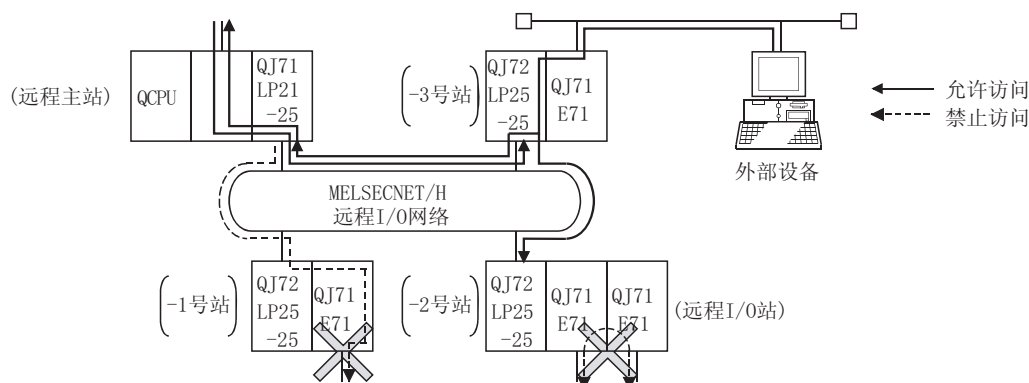
- (c) 以下 MELSECNET/H 远程 I/O 站的软元件可以用软元件存储器的读/写功能访问。

详细说明请参阅参考手册。

软元件名称	软元件符号	软元件名称	软元件符号
特殊继电器	SM	链接继电器	B
特殊寄存器	SD	数据寄存器	D
输入继电器	X	链接寄存器	W
输出继电器	Y	链接特殊继电器	SB
内部继电器	M	链接特殊寄存器	SW

- (d) 通过 MELSECNET/H 远程 I/O 站对其它站的访问，允许访问远程主站和远程 I/O 站。

- 1) 下图中，可以从外部设备访问 MELSECNET/H 远程主站和 MELSECNET/H 远程 I/O 站。



- 2) 不允许通过下列 MELSECNET/H 远程 I/O 站访问其它站
- 通过安装在 MELSECNET/H 远程 I/O 站上的以太网模块，从 MELSECNET/H 远程主站访问端访问其它站。(上图中的 1 号站)
 - 通过两个以太网模块访问其它站。(上图中的 2 号站)

*2 此功能可用于功能版本 D 版或以上版本的 MELSECNET/H 远程 I/O 站。
使用此功能需要 GX Developer 版本 8.18U 或以上版本。

(3) 通过 GX Developer 设置以太网模块的参数

为了使用安装在 MELSECNET/H 网远程 I/O 站上的以太网模块，应通过 GX Developer 设置下列参数。

在设置安装在 QCPU 站上的以太网模块参数时，可以以相同方法进行设置。参阅第 4.5 节和后面的章节。

关于如何显示每个设置画面的情况，参阅 GX Developer 操作手册。

(安装在 MELSECNET/H 网远程 I/O 站上以太网模块的参数设置项)

参数设置项	设置	备注
I/O 分配	设置模块的安装信息。	无须设置
用网络参数设置以太网模块数	执行将以太网模块用作网络模块的设置。	-
操作设置	执行以太网模块公共项的设置。	
初始设置	设置数据通讯的定时器值。	
开放设置	进行与开放/关闭处理有关的设置。	
路由中继参数	在使用路由中继器通讯时进行设置。	
站号<->IP 关联信息	在通过 MELSECNET/H、MELSECNET/10 通讯时进行设置。	
远程口令设置	对远程口令进行设置	

要点

- (1) 将 GX Developer 连接到 MELSECNET/H 网远程 I/O 站，并用它来设置参数。
- (2) 始终设置“设置 MNET/10H 以太网卡号的网络参数”和“操作设置”。
- (3) 更改设置后将 MELSECNET/H 网远程 I/O 站复位。

备注

没有必要使用 GX Developer I/O 分配来设置“智能功能模块开关设置”。

在上述“操作设置”、“初始设置”和“开放设置”中执行与开关设置对应的每种设置。

(4) 数据通讯步骤

- (a) 将 GX Developer 连接到 MELSECNET/H 远程 I/O 站并执行下列设置：
- 1) 使用 TCP/IP 通讯
 - 在“操作设置”下，针对“初始时间设置”选择“始终等待打开”。
 - 在“开放设置”下，针对“开放系统”设置选择“非被动或完全被动”。
 - 2) 使用 UDP/IP 通讯(当使用用户端口时)
在“操作设置”下，针对“初始时间设置”，选择“始终等待打开”。也可以使用以太网模块的自动开放 UDP 端口进行数据通讯。
- (b) 起动 MELSECNET/H 远程 I/O 站。
- (c) 执行开放处理。
- 1) 使用 TCP/IP 通讯时，必须从外部设备执行“主动开放”。(由于是“被动开放”，以太网模块端的状态为“等待开放”。)
 - 2) 使用 UDP/IP 进行通讯时，在外部设备端执行开放处理。(通过内部处理，以太网模块端的状态为“允许数据通讯”。)
- (d) 执行数据通讯。
- (e) 完成数据通讯后，执行关闭处理。
- 1) 使用 TCP/IP 进行通讯时，必须从外部设备执行关闭处理。(以太网模块端在外部设备发出请求时执行关闭处理，其状态变成可接收另一个开放请求。)
 - 2) 使用 UDP/IP 通讯时，执行外部设备的关闭处理。(不需要在以太网模块端进行关闭处理。)

2.7 检查功能版本和系列号

本节说明功能版本、系列号和使用以太网模块改进后新增功能的相关产品时，如何对其功能版本和系列号进行检查。

(1) 与专门设计的、使用以太网模块新增功能的相关产品的兼容性

新增功能		以太网模块的版本			相关产品的版本	
		QJ71E71-100	QJ71E71-B5	QJ71E71-B2	CPU 模块	GX Developer
支持 IEEE802.3 帧		○	○	功能版本 B 或以上版本系列号的前 5 个数字是 03102	○	版本 7 或以后的版本
以太网模块的重新初始化处理	使用顺控程序进行重新初始化	功能版本 B 或以上版本系列号的前 5 个数字是 03102	○	功能版本 B 或以上版本系列号的前 5 个数字是 03061	○	○
	通过专用指令进行重新初始化 (UINI 指令)			功能版本 B 或以上版本系列号的前 5 个数字是 03102		
	TCP 最大分段传送	功能版本 B 或以上版本系列号的前 5 个数字是 05051		功能版本 B 或以上版本系列号的前 5 个数字是 05051		
重新开放以太网模块 (*1) 的处理		功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	○
存在确认功能	用 KeepAlive 检查	功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	版本 8.05F 或以后版本
通过 TCP/IP 通讯最多连接 17 个 MELSOFT 产品		○	○	功能版本 B 或以上版本系列号的前 5 个数字是 02122	○	版本 6.05F 或以后版本
简化与 MELSOFT 产品的连接	简化访问其它站	功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	功能版本 B 或以上版本系列号的前 5 个数字是 05051	○	○
	用同一个站号访问					
GX Developer 的以太网诊断功能	监视不同的以太网模块状态	○	○	功能版本 B 或以后版本	功能版本 A 或以上版本系列号的前 5 个数字是 02092	版本 6 或以后版本
	通过以太网板 PING 测试/环路回送测试					版本 7 或以后版本
	通过 CPU 进行 PING 测试					版本 7 或以后版本
网络集线器连接状态监视功能		功能版本 D 或以后的版本	×	×	○	○
使用电子邮件功能时	按附件发送 CSV 格式的文件	○	○	功能版本 B 或以后版本	○	版本 6 后以后版本
	发送正文			功能版本 B 或以上版本系列号的前 5 个数字是 03102		版本 7 或以上版本
	支持编码/解码	功能版本 B 或以上版本系列号的前 5 个数字是 03102		功能版本 B 或以上版本系列号的前 5 个数字是 03102		○
	通过 PLC CPU 监视功能发送电子邮件正文中的字符串	功能版本 D 或以上版本系列号的前 5 个数字是 07082		功能版本 D 或以上版本系列号的前 5 个数字是 07082		功能版本 D 或以上版本系列号的前 5 个数字是 07082
用于数据链接指令的目标站 CPU 类型说明		功能版本 D 或以上版本	功能版本 D 或以上版本	功能版本 D 或以上版本	○	○
数据链接指令增加的数据长度 (480 到 960 字)		功能版本 D 或以上版本系列号的前 5 个数字是 07082	功能版本 D 或以上版本系列号的前 5 个数字是 07082	功能版本 D 或以上版本系列号的前 5 个数字是 07082	○	○
文件传送 (FTP 服务器) 功能的多 CPU 系统兼容性		○	○	功能版本 B 或以上版本系列号的前 5 个数字是 03102	功能版本 B 或以上版本	○

○：可用(无版本限制) ×：不可用

(接下页)

新增功能	以太网模块的版本			相关产品的版本	
	QJ71E71-100	QJ71E71-B5	QJ71E71-B2	CPU 模块	GX Developer
通过 MC 协议进行通讯 与 4E 帧兼容	功能版本 D 或以上版本 系列号的前 5 个数字是 07082	功能版本 D 或以上 版本系列号的前 5 个数字是 07082	功能版本 D 或以上版本系 列号的前 5 个数字是 07082	○	○
使用 Web 功能进行通讯	○	○	功能版本 B 或以上版本系 列号的前 5 个数字是 05051	○	○
远程口令核对	○	○	功能版本 B 或以上版本	功能版本 A 或以 上版本系列号的 前 5 个数字是 02092	版本 6 或以上版 本

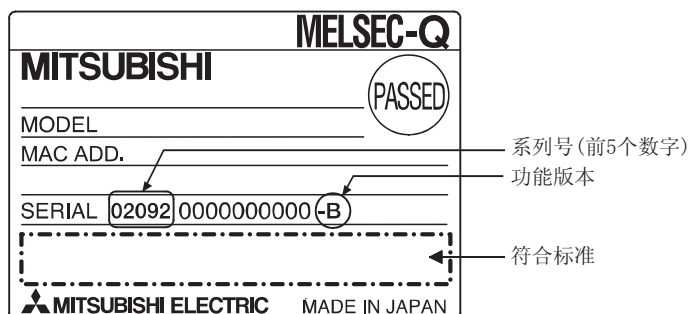
○: 可用(无版本限制) ×: 不可用

*1 在 TCP 开放完成状态再次接收外部设备的主动开放请求时, 对要执行的以太网进行更改操作。(参阅 5.6(2)中的要点)

(2) 检查 Q 系列 PLC 的功能版本和系列号

(a) 使用模块侧面的“铭牌”进行检查

主体模块的系列号和功能版本标示在铭牌的 SERIAL 一行上。



(b) 通过 GX Developer 检查

下面说明的是如何通过 GX Developer 检查目标模块的系列号和功能版本。系列号和功能版本显示在“产品信息列表”或 GX Developer 的“模块详细信息”屏上。

下图表示的是如何检查“产品信息列表”上的系列号和功能版本。(有关“模块的详细信息”屏, 参阅第 11.2 节。)

[起动步骤]

“诊断” → “系统监视” → “产品信息列表”

Slot	Type	Series	Model name	Points	I/O No.	Master PLC	Serial No	Ver
PLC	PLC	Q	Q02HCPU	-	-	-	0209200000000000	A
0-0	Intelli. Q	QJ71E71		32pt	0000	-	0209300000000000	B
0-1	Intelli. Q	QJ71C24-R2		32pt	0020	-	0205100000000000	B
0-2	-	-	None	-	-	-	-	-

[系列号、版本]

- 目标模块的系列号显示在系列号区。
- 目标模块的功能版本显示在版本区。

3 规格

本章介绍以太网模块的性能规格和传送规格。
关于通用规格的详细说明，请参见 QCPU 的用户手册 (Q 模式)。

3.1 性能规格

以下是以太网模块的性能规格。

项目		规格			
		QJ71E71-100		QJ71E71-B5	QJ71E71-B2
		100BASE-TX	10BASE-T	10BASE5	10BASE2
传送规格	数据传送速率	100 M bps	10 M bps		
	通讯模式	全双工/半双工		半双工	
	传送方法	基带			
	最大节点-节点距离	—		2500m (8202.10ft.)	925m (3034.77ft.)
	最大段长	100m (328.08ft.) (*1)		500m (1640.42ft.)	185m (606.96ft.)
	最多节点/连接数	级联连接 最多 2 级	级联连接 最多 4 级	100 个单元/段	30 个单元/段
	最小节点间隔	—		2.5m (8.20ft.)	0.5m (1.64ft.)
传送数据存储器	允许同时开放连接数	16 个连接 (顺控程序可用的连接)			
	固定缓冲存储器	1k 字 × 16			
	随机访问缓冲器	6k 字 × 1			
	电子邮件	附件	6k 字 × 1 (*4)		
		正文	960 字 × 1 (*4)		
占用的 I/O 点数		32 点/1 插槽 (I/O 分配: 智能)			
5VDC 内部电流消耗		0.50 A	0.50 A	0.60 A (*3)	
12VDC 外部电源功率容量 (收发器)		—		(*3)	—
外部尺寸		98 (3.86in.) (H) × 27.4 (1.08in.) (W) × 90 (3.54in.) (D) [mm]			
重量		0.11kg (0.24lb)	0.12kg (0.26lb)	0.13kg (0.29lb) (*3)	

*1 网络集线器和节点之间的长度。

*2 必须使用符合 AUI 电缆规格的收发器或软元件。(请参见第 2.2 节)

*3 产品系列号的前 5 位是“05049”或之前的产品有以下不同:

- 5VDC 内部电流消耗: 0.70A
- 重量: 0.14kg (0.31b)

*4 下表简要地概括了电子邮件传送和接收功能的规格。

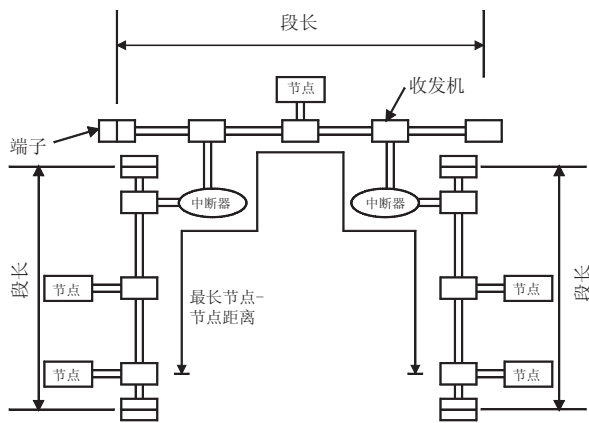
PLC CPU 的电子邮件传送和接收功能和随机访问缓冲存储器通讯功能不可一起使用; 一次只可以使用其中一项功能。(以太网模块 PLC CPU 监视功能的电子邮件传送和接收功能和随机访问缓冲存储器的通讯功能可以一起使用。)

项目		规格
传送规格 传送和接收数据	数据大小	附件 6k 字 × 1
		正文 960 字 × 1
	数据传送方法	发送时: 按附件或正文(二选一)发送一个文件 接收时: 按附件接收一个文件
	标题	Us-ASCII 格式或 ISO-2022-JP (Base64)
	附件格式	MIME 格式
	MIME	版本 1.0
	附件的数据格式	可以选择二进制/ASCII/CSV 文件名称: XXXX.bin (二进制), XXXX.asc (ASCII), XXXX.csv (CSV) (CSV: 逗号分隔的值)
	附件的分割	不能分割(只可以发送/接收一个文件) * 如果接收到任何分割文件, 则只会收到第一个文件, 剩下的文件将废除。
	发送时(编码)	标题: Base64/7bits 正文: 7 位 附件: Base64
	接收时(解码)	标题: (不解码) 正文: (无法接收) 附件: Base64/7 位/8 位/Quoted Printable * 如果从外部设备向 PLC 端发送电子邮件, 应指定附件的编码方法(Base64/7 位/8 位/Quoted Pintable)
	加密	无
	压缩	无
	使用邮件服务器通讯	SMTP(发送服务器) 端口编号 = 25 POP3(接收服务器) 端口编号 = 110
操作检查信箱	Microsoft® Corporation Internet Explorer 5.0 (Outlook Express 5.5/Outlook Express 5) Netscape® Communications Corporation Netscape® 4.05	

备注

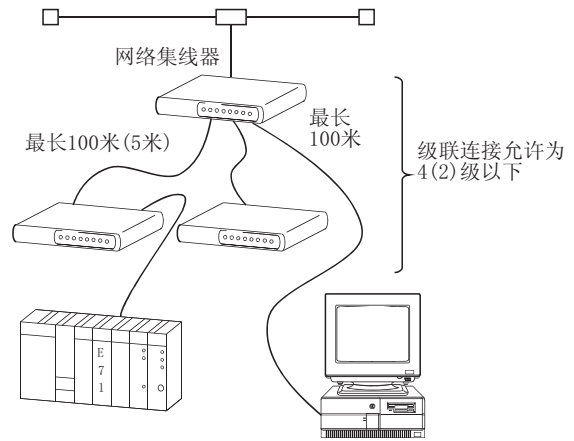
下面说明了每一项传送的规格。

[使用 10BASE2/10BASE5 连接]



* 当使用10BASE2连接时没有收发机。

[使用 10BASE-T/100BASE-TX 连接]



括号()中的项目表示使用100BASE-TX连接时的情况。

3.2 通讯用的数据代码

本节说明了以太网模块和外部设备或 PLC CPU 之间通讯时使用的数据代码。

(1) 通讯时使用的数据代码列表如下。

1) 以太网模块-外部设备

可以通讯的数据既可以选择数据代码设置中的二进制代码或 ASCII 代码，如下所示。

二进制代码和 ASCII 代码之间的切换是通过 GX Developer 进行设置的：

[GX Developer] - [网络参数] - [操作设置] - [通讯数据代码]

有关详细说明，请参见第 4.7 节“操作设置”。

数据通讯功能		通讯数据代码		参考章节
		二进制代码	ASCII 代码	
使用 MC 协议进行通讯	自动开放 UDP 端口	○(* ¹)	—	第 6 章
	用户开放端口	○	○	
使用固定缓冲存储器进行通讯	有顺序	○	○	第 7 章
	无顺序	○(* ¹)	—	第 8 章
使用随机访问缓冲存储器进行通讯		○	○	第 9 章

○：可选 ×：无法通讯

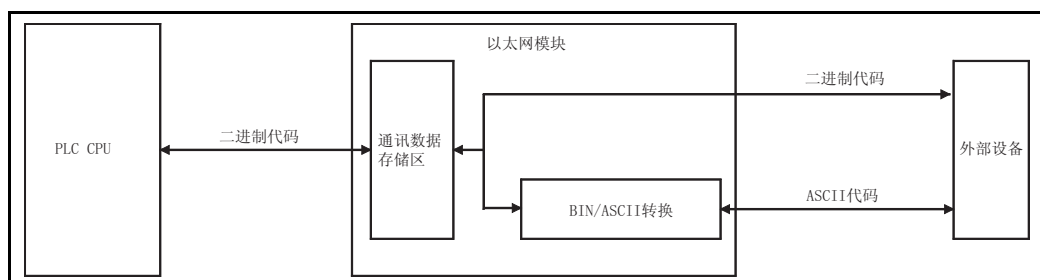
*1 使用二进制代码执行通讯，而不用考虑[操作设置]-[通讯数据代码]中的设置(请参见第 4.7 节)

备注

使用下列数据通讯功能时，可以使用每个功能处理的数据代码执行通讯，而不用考虑通讯数据代码的设置。

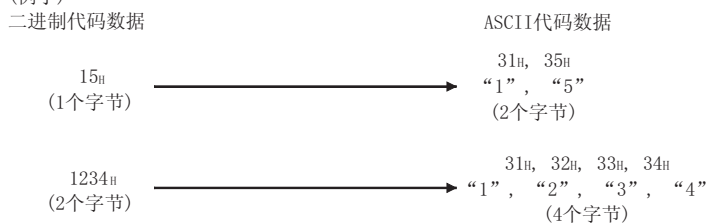
- 收发电子邮件
- MELSECNET/H、MELSECNET/10 中继通讯
- 使用数据链接指令进行通讯
- 文件传送(FTP 服务器功能)
- 使用 Web 功能进行通讯

- 2) 以太网模块 ↔ PLC CPU
以二进制代码发送和接收数据。



- (2) 当使用 ASCII 代码进行通讯时，1 个字节的二进制代码数据自动转换成 2 个字节的 ASCII 代码数据，然后传送。

(例子)



- (3) 以太网模块和外部设备之间一次可以通讯的数据量由使用的功能和通过选择 [GX Developer]-[通讯数据代码] 设置的数据代码 (二进制 ASCII) 确定。
下面内容表示的是每个数据通讯功能一次可以发送和接收的通讯数据的最大量。

数据通讯功能		可交换数据量	
使用 MC 协议进行通讯		可以用每个命令/指令 指定的最多点数是：最多 1920 个字节	
使用固定缓冲存储器进行通讯	有顺序	1017 个字 (二进制代码)	508 个字 (ASCII 代码)
	无顺序	2046 个字	
使用随机访问缓冲存储器进行通讯		1017 个字 (二进制代码)	508 个字 (ASCII 代码)
收发电子邮件		附件：最多 6k 个字 正文：最多 960 个字	
MELSECNET/H、MELSECNET/10 中继通讯		480 个字	
使用数据链接指令进行通讯		可以用每个命令/指令 指定的最多点数是：最多 960 个字节	
文件传送 (FTP 服务器功能)		一个文件的最多字数	
使用网络不能进行通讯		可以用每个命令/指令 指定的最多点数是：最多 1920 个字节	

3.3 外部设备和每个通讯功能的附加功能之间的关系

本节说明可以执行哪一个外部设备数据通讯和每个功能中可以使用的各项附加功能。

(1) 使用各种功能与外部设备进行通讯的可能性

下表列出了使用各种功能与外部设备进行通讯的可能性。

功能	外部设备				
	个人计算机 ↓ QJ71E71	个人计算机 ↑ QJ71E71	QJ71E71 ↓ ↑ QJ71E71	QJ71E71 ↓ 传统机型	传统机型 ↓ QJ71E71
使用 MC 协议进行通讯	○				×
使用固定缓冲存储器进行通讯			○		
使用随机访问缓冲存储器进行通讯	○				×
收发电子邮件		○			×
使用数据链接指令进行通讯		×	○	△(*1)	×
文件传送(FTP 服务器功能)	○				×
使用 Web 功能进行通讯	○				×

○: 可以通讯 △: 可以通讯(有限制) ×: 不能通讯

QJ71E71 : Q 系列以太网接口模块

传统机型 : QnA 或 A 系列以太网接口模块

*1 可以用 QnA 系列以太网接口模块进行通讯。

(2) 与附加功能的关系

下表列出了可以使用的功能及其与附加功能之间的对应关系。

通讯功能		附加功能							通讯方法	
		MELSECNET/H、 MELSECNET/10 中 继通讯	路由中继通 讯(路由中 继功能)	外部设备 的存在 检查	通过成对开放 进行通讯	通过自动开放 UDP 端口进行 通讯	远程口令 核对	同步广播	TCP/IP	UDP/IP
使用 MC 协 议进行通讯	QnA 兼容 3E 帧	○	○	○(*2)	×	○	○	○(*1)	○	○
	4E 帧									
	A 兼容 1E 帧	×	○	○	×	×	○	×	○	○
使用固定缓 冲存储器进 行通讯	有顺序	×	○	○	○	×	○	×	○	○
	无顺序	×	○	○	○	×	○	○(*1)	○	○
使用随机访问缓冲存储器 进行通讯		×	○	○	×	×	○	×	○	○
收发电子邮件		×	×	×	×	×	×	×	○	×
使用数据链接指令进行通 讯		○	○	×	×	○	×	○	×	○
文件传送 (FTP 服务器功能)		×	○	×	×	×	○	×	○	×
使用 Web 功能进行通讯		○	○	×	×	×	○	×	○	×

○: 可用 ×: 不可用或该功能与功能栏中的任何功能都不对应。

*1 仅对 UDP 是有效的。

*2 不包括自动开放 UDP 端口。

3.4 以太网模块功能列表

本节所示为以太网模块功能的列表。

(1) 以太网模块的基本功能

以太网模块可以通过 TCP/IP 或 UDP/IP 通讯执行下表所示的通讯。

功能	说明	参考章节	
使用 MC 协议进行通讯	4E 帧	从外部设备读取 PLC CPU 数据/将 PLC CPU 数据写入外部设备。	第 5 章 参考手册
	QnA 兼容		
	3E 帧		
	A 兼容 IE 帧		
使用固定缓冲存储器进行通讯	有顺序	使用以太网模块的固定缓冲存储器在 PLC CPU 和外部设备之间发送/接收任意数据。	第 7 章
	无顺序		第 8 章
使用随机访问缓冲存储器进行通讯	从多个外部设备读/写数据到以太网模块的随机访问缓冲存储器。	第 9 章	
收发电子邮件	通过电子邮件发送/接收数据。 • 通过 PLC CPU 发送/接收 • 通过 PLC CPU 监视功能发送 (自动通知功能)	用户手册(应用篇)的第 2 章	
使用数据链接指令进行通讯	使用数据链接指令, 通过以太网读/写其它站的 PLC CPU 数据。	用户手册(应用篇)的第 4 章	
文件传送(FTP 服务器功能)	以文件为单位, 使用 FTP 命令从外部设备读/写。	用户手册(应用篇)的第 5 章	
使用 Web 功能进行通讯	使用市售网络浏览器, 通过以太网读/写 PLC CPU 数据。	用户手册 (Web 功能篇)	

(2) 以太网模块的附加功能

下表列出了可以使用的以太网模块的附加功能。

功能	说明	参考章节
MELSECNET/H、MELSECNET/10 中继通讯	在以太网和 MELSECNET/H 或 MELSECNET/10 共存的网络系统, 或中继多个以太网的网络系统中, 数据通讯通过上述的多个网络执行。	用户手册(应用篇)的第 3 章
路由中继通讯(路由中继功能)	通过路由器或网关执行数据通讯。(路由中继功能不是以太网模块作为路由器工作时的功能)。	第 5.3 节
外部设备的存在检查	检查开放后外部设备是否正常工作(开放处理)。	第 5.2.2 和 5.5 节
通过成对开放进行通讯	组合成对, 然后开放一个接收连接和一个传送连接(用于固定缓冲存储器)。	第 5.7 节
通过自动开放 UDP 端口进行通讯	允许安装了以太网模块的站起动后进行通讯。(不需要主顺控程序执行开放/关闭处理)。	第 5.8 节
远程口令核对	防止远程用户未经授权访问 QCPU。	第 5.9 节
同步广播	发送/接收以太网上所有外部设备的数据, 与通过 UDP/IP 对以太网模块进行数据通讯相同。	第 8.5 节
连接 MELSOFT 产品(GX Developer、MX Component 等)和 GOT	通过 TCP/IP 通讯或 UDP/IP 通讯连接 MELSOFT 产品(GX Developer、MX Component 等)和 GOT。也可以同时连接多个 MELSOFT 产品和 GOT。	MELSOFT 产品手册的 1.2(6) 节

(3) 以太网模块的状态检查

检查以太网模块是否正常工作及它是否可以正常通讯。

功能	说明	参考章节
自抑制测试	检查以太网模块的发送/接收功能和线路连接状态。	第 4.8.1 节
硬件测试	测试以太网模块的 RAM 和 ROM。	第 4.8.2 节
通讯出错存储	当发生数据通讯出错时，该功能存储出错信息(出错日志)，包括缓冲存储器中最多 16 对的外部设备的信息副标题和 IP 地址。	第 11 章

3.5 专用指令列表

下表列出了可以主以太网模块使用的专用指令。

应用	指令名称	说明	参考章节	
用于建立和关闭连接	OPEN	开放连接。	第 10.8 节	
	CLOSE	关闭连接。	第 10.5 节	
用于重新初始化	UINI	重新初始化以太网模块。	第 10.9 节	
用于固定缓冲存储器通讯	BUFRCV	将接收数据读入固定缓冲存储器。	第 10.2 节	
	BUFRCVS	使用中断程序将接收读入固定缓冲存储器中。	第 10.3 节	
	BUFSND	使用固定缓冲存储器将数据发送到外部设备。	第 10.4 节	
用于读和清除出错信息	ERRCLR	清除出错(熄灭[COM.ERR]LED, 清除出错日志)。	第 10.6 节	
	ERRRD	读出错信息。	第 10.7 节	
用于发送和接收电子邮件	MRECV	接收电子邮件。	用户手册(应用篇)的第 6.2 节	
	MSEND	发送电子邮件。	用户手册(应用篇)的第 6.3 节	
用于其它站的 PLC 通讯(用于数据链接的指令)	用于读/写软元件数据	READ	读取其它站的字软元件。	用户手册(应用篇)的第 6.4 节
		SREAD	读取其它站的字软元件(带有完成软元件)。	用户手册(应用篇)的第 6.10 节
		WRITE	写入其它站的字软元件。	用户手册(应用篇)的第 6.12 节
		SWRITE	写入其它站的字软元件。(带有完成软元件)。	用户手册(应用篇)的第 6.11 节
		ZNRD	读取其它站的字软元件。	用户手册(应用篇)的第 6.13 节
		ZNWR	写入其它站的字软元件。	用户手册(应用篇)的第 6.14 节
	用于发送/接收信息(任意数据)	SEND	把数据发送到其它站。	用户手册(应用篇)的第 6.9 节
		RECV	从其它站读取接收数据(用于主程序)。	用户手册(应用篇)的第 6.5 节
		RECVS	从其它站读取接收数据(用于中断程序)。	用户手册(应用篇)的第 6.6 节
	读/写时钟数据, 远程运行/停止	REQ	以远程模式运行/停止其它站。	用户手册(应用篇)的第 6.7 节
			读/写其它站的时钟数据。	用户手册(应用篇)的第 6.8 节

3.6 GX Developer 设置的以太网模块项目列表

下表列出了使用 GX Developer 设置的参数设置项。

参数设置项	设置的说明	功能和参数设置要求 (*1)								参考章节 (*2)
		MC	固定	随机	数据链接	FTP	自动通知	邮件	网络	
PLC 参数	—	—								—
I/O 分配设置	设置模块的安装信息。这是以太网模块用在多 CPU 系统中时设置的。	△	△	△	△	△	△	△	△	第 4.5 节
中断指针设置	设置以太网模块端的控制号 (SI) 和 PLC CPU 上使用的中断指针之间的关系。	—	△	—	△	—	—	—	—	第 7.3 节
设置 MNET/10H 以太网卡号的网络参数	执行以太网模块用作网络模块的设置。	○	○	○	○	○	○	○	○	第 4.6 节
操作设置	设置模块的公共项。这些设置是初始化设置所要求的。	○	○	○	○	○	○	○	○	第 4.7 节
初始设置	设置数据通讯定时器值。	△	△	△	△	△	△	△	△	第 5.2 节
	设置 DNS 服务器的 IP 地址。	—	—	—	—	—	△	△	—	应用篇的第 2 章
开放设置	设置为了执行与外部设备进行数据通讯的开放处理。	○	○	○	—	—	—	—	—	第 5.5 节
路由中继参数	设置以太网的路由中继器。	△	△	△	△	△	△	△	△	第 5.3 节
站号 <-> IP 关联信息	执行设置, 使以太网可以同等处理 MELSECNET/H 或 MELSECNET/10 网络系统, 以便于达到与其它站 PLC 进行通讯的目的。	△	—	—	△	—	—	—	△	应用篇的第 3 章
FTP 参数	执行文件传送 (FTP) 的设置。	—	—	—	—	○	—	—	—	应用篇的第 5 章
电子邮件设置	执行发送/接收电子邮件的设置和使用自动通知功能的设置。	—	—	—	—	—	○	○	—	应用篇的第 2 章
发送邮件地址设置	设置目标邮件地址。	—	—	—	—	—	○	○	—	
新闻设置	设置通知条件。	—	—	—	—	—	○	—	—	
中断设置	设置请求 PLC CPU 中断时以太网模块端的控制号 (SI)。	—	△	—	△	—	—	—	—	第 7.3 节
冗余设置	进行设置以使得能够在冗余系统中使用以太网模块。	—	—	—	—	—	—	—	—	第 5.11.3 节
成组设置		—	—	—	—	—	—	—	—	第 4.6 节
路由参数	执行中继站的设置, 使以太网可以与 MELSECNET/H 或 MELSECNET/10 网络系统同等处理, 以便于达到与其它站 PLC 进行通讯的目的。	△	—	—	△	—	—	—	△	应用篇的第 3 章
远程口令设置	设置远程口令核对的目标连接。	△	△	△	△	△	—	—	△	第 5.9.5 节

○: 使用适用功能时必须设置 △: 按需要设置 ×: 不需要设置

*1 上表中使用的缩写含义为:

- MC : 使用 MC 协议通讯
- 随机 : 使用随机访问缓冲存储器通讯
- FTP : 文件传送协议
- 邮件 : 电子邮件
- 固定 : 使用固定缓冲存储器通讯
- 数据链接 : 使用数据链接指令通讯
- 自动通知 : 通过自动通知进行数据传送
- 网络 : 使用 Web 功能通讯

*2 参考章节一栏中的“应用篇”表示用户手册 (应用篇)。

3.7 从 PLC CPU 发出的输入/输出信号和发给 PLC CPU 的输入/输出信号列表

本节说明了以太网模块的输入/输出信号。

关于下表中所示的输入/输出信号的分配，假定以太网模块安装在基板的槽 0 中。

从×开始的软元件地址表示从以太网模块到 PLC CPU 的输入信号，从 Y 开始的软元件地址表示从 PLC CPU 到以太网模块的输出信号。

下面表示的是从 PLC CPU 发出的输入/输出信号或发给 PLC CPU 的输入/输出信号。

信号方向：以太网模块 → PLC CPU			信号方向：PLC CPU → 以太网模块		
软元件地址	信号名称	参考章节	软元件地址	信号名称	参考章节
X0	1号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	Y0	1号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X1	1号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	Y1	2号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X2	2号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	Y2	3号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X3	2号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	Y3	4号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X4	3号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	Y4	5号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X5	3号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	Y5	6号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X6	4号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	Y6	7号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X7	4号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	Y7	8号连接 ON：在发送请求或接收完成确认信号时 OFF：—	-
X8	5号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	Y8	1号连接 ON：开放请求 OFF：—	-
X9	5号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	Y9	2号连接 ON：开放请求 OFF：—	-
XA	6号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	YA	3号连接 ON：开放请求 OFF：—	-
XB	6号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	YB	4号连接 ON：开放请求 OFF：—	-
XC	7号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	YC	5号连接 ON：开放请求 OFF：—	-
XD	7号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	YD	6号连接 ON：开放请求 OFF：—	-
XE	8号连接的固定缓冲存储器通讯 ON：发送正常完成或接收完成 OFF：—	-	YE	7号连接 ON：开放请求 OFF：—	-
XF	8号连接的固定缓冲存储器发送 ON：发送异常或接收异常的检测 OFF：—	-	YF	8号连接 ON：开放请求 OFF：—	-

信号方向: 以太网模块 → PLC CPU			信号方向: PLC CPU → 以太网模块		
软元件地址	信号名称	参考章节	软元件地址	信号名称	参考章节
X10	1号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y10	禁用	-
X11	2号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y11		
X12	3号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y12		
X13	4号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y13		
X14	5号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y14		
X15	6号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y15		
X16	7号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y16		
X17	8号连接建立完成 ON: 连接建立完成信号 OFF: —	-	Y17	COM. ERR LED 熄灭请求 ON: 熄灭请求时 OFF: —	第 11.1 节
X18	开放异常检测信号 ON: 异常检测 OFF: —	第 5.6 节	Y18	禁用	-
X19	初始化正常完成信号 ON: 正常完成 OFF: —	第 5.1 节	Y19	初始化请求信号 ON: 请求时 OFF: —	-
X1A	初始化异常完成信号 ON: 异常完成 OFF: —	第 5.1 节	Y1A	禁用	-
X1B	禁用	-	Y1B		
X1C	COM. ERR LED 亮确认 ON: 亮(请参见第 11.1 节) OFF: 灭	第 11.1 节	Y1C		
X1D	禁用	-	Y1D		
X1E	禁用	-	Y1E		
X1F	WDT 出错检测 ON: WDT 出错 OFF: —	第 11.1 节	Y1F		

要点

在 PLC CPU 的输入/输出信号中，不要输出(接通)标记有“禁用”的信号。如果输出了任何“禁用”信号，PLC 系统可能发生故障。

要点

- 当 QnA 系列以太网接口模块程序也用于 Q 系列以太网模块时，使用本节中所示的输入/输出信号(请参见附录中的第 2 节)。在 QCPU 中，智能功能模块的输入/输出信号是用专用指令接通/关闭的。没有必要使用顺控程序接通/关闭信号，但是每个功能说明章节中程序范例中所示的除外。
- 另外，当 QnA 系列以太网接口模块程序也用于 Q 系列以太网模块时，推荐你使用 Q 系列以太网模块每个手册的相应功能说明章节中所示的专用指令，来代替该指令。

3.8 缓冲存储器的应用和分配列表

本节为缓冲存储器的详细说明。

(1) 缓冲存储器的配置

缓冲存储器由用户区和系统区组成，如下所示。

(a) 用户区

- 1) 该区域是用户写/读数据的位置。
- 2) 用户区由参数初始化处理参数区和数据通讯、数据通讯区、通讯状态和通讯出错数据存储区组成。
- 3) 对用户区数据的读/写应按照具体说明章节中相应的指令执行。
如果继续执行数据通讯可能会花更长的时间，因此，仅在需要时执行。

(b) 系统区

该区由以太网模块使用。

要点
<p>不要将数据写入缓冲存储器的“系统区”中。 如果数据写入任意系统区中，PLC 系统可能无法正常运行。 某些用户区部分地包括系统区，当读/写缓冲存储器时必须小心。</p>

要点
<p>(1) 当 QnA 系列以太网模块程序用于 Q 系列以太网模块时，使用 FROM/T0 指令或其它应用命令访问本节中所示的缓冲存储器(请参见附录中的第 2 节)。 在 QCPU 中，智能功能模块的缓冲存储器是由专用指令访问的。 没有必要使用顺控程序中的 FROM/T0 指令或其它指令直接访问，但是访问每个功能说明章节程序范例中所示的缓冲存储器除外。</p> <p>(2) 另外，当 QnA 系列以太网模块程序用于 Q 系列以太网模块时，推荐你使用 Q 系列以太网模块每个手册的相应功能说明章节中所示的专用指令，来代替该指令。</p>

(2) 缓冲存储器的分配

缓冲存储器是按每一地址 16 位组成的。

<位配置图>

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	----	----

下面表示的是缓冲存储器地址。

地址 十进制 (十六进制)	应用	名称	初始值 (十六进制)	GX Developer 设置适用性 (* ¹)	参考章节 (* ²)	
0 至 1 (0 至 1 _H)	初始化处理参数 设置区	本站以太网模块 IP 地址	C00001FE _H	○	4.7 节	
2 至 3 (2 至 3 _H)		系统区	-	-	-	
4 (4 _H)		特殊功能设置 <ul style="list-style-type: none"> 路由中继功能 (b5, b4) 00: 不使用 (默认) 01: 使用 MELSECNET/H、MELSECNET/10 中继功能的转换系统设置 (b7、b6) 00: 自动响应系统 (默认) 01: IP 地址计算系统 10: 图表交换系统 11: 共用系统 FTP 功能设置 (b9、b8) 00: 不使用 01: 使用 (默认) 非以上的位只供系统使用。	0100 _H	○	5.3 节 应用篇的 第 3 章	
5 至 10 (5 至 A _H)		系统区	-	-	-	
11 (B _H)		监视定时器	TCP ULP 定时器值 设置时间 = 设置值 × 500 ms	3C _H (60)	○	5.2 节
12 (C _H)			TCP 零窗口定时器值 设置时间 = 设置值 × 500 ms	14 _H (20)		
13 (D _H)			TCP 再次发送定时器值 设置时间 = 设置值 × 500 ms	14 _H (20)		
14 (E _H)			TCP 结束定时器值 设置时间 = 设置值 × 500 ms	28 _H (40)		
15 (F _H)			IP 组合定时器值 设置时间 = 设置值 × 500 ms	A _H (10)		
16 (10 _H)			向应 WDT value 设置时间 = 设置值 × 500 ms	3C _H (60)		
17 (11 _H)			目标存在确认启动时间间隔 设置时间 = 设置值 × 500 ms	4B0 _H (1200)		
18 (12 _H)			目标存在确认时间间隔定时器 设置时间 = 设置值 × 500 ms	14 _H (20)		
19 (13 _H)		目标存在确认再次发送定时器	3 _H (3)			
20 (14 _H)	自动开放 UDP 端口号	1388 _H	×	5.8 节		
21 至 29 (15 至 1D _H)	系统区	-	-	-		

(转下页)

*1 允许/禁止从 GX Developer 进行的设置

○: 允许设置 ×: 禁止设置

*2 参考章节一栏中的“应用篇”表示用户手册(应用篇)。

(续上页)

地址 十进制 (十六进制)	应用	名称	初始值 (十六进制)	GX Developer 设置适用性	参考章节
30 (1E _h)	初始化处理参数 设置区(用于重新 初始化)	TCP 最大段传送 设置区	0 _h : 允许 TCP 最大段传送的大小选项 8000 _h : 禁止 TCP 最大段传送的大小选项 • 重新初始化可使设置值有效。	×	5.2.3 节
31 (1F _h)		通讯条件设置区 (操作设置)	通讯条件设置区(操作设置) • 通讯数据代码设置(b1) 0: 以二进制代码通讯 1: 以 ASCII 代码通讯 • TCP 存在确认设置(b4) 0: 使用 Ping 1: 使用 KeepAlive • 发送帧设置(b5) 0: 以太网帧 1: IEEE 802.3 帧 • 在 RUN 时设置允许/禁止写(b6) 0: 禁止 1: 允许 • 初始化时间设置(b8) 0: 不等待打开(STOP 时不可以进行通讯) 1: 一定要等待 OPEN(STOP 时可以进行通讯) • 重新初始化的规格(B15) 0: 重新初始化的处理完成(系统重新设置) 1: 重新初始化的处理请求(用户设置) 保留除以上的位来用于系统使用。	○	
32 (20 _h)	通讯参数设置区	连接使用设置区	1号连接 • 固定缓冲存储器的使用(b0) 0: 用于发送或固定缓冲存储器的通讯未执行 1: 用于接收 • 目标存在确认(b1) 0: 不确认 1: 确认 • 成对开放(b7) 0: 不成对 1: 成对 • 通讯方法(协议)(b8) 0: TCP/IP 1: UDP/IP • 固定缓冲存储器通讯(b9) 0: 有顺序 1: 无顺序 • 开放系统(b15、b14) 00: 主动开放或 UDP/IP 10: 主动开放 11: 完全被动开放 非以上的位只供系统使用。	○	5.5 节
33 (21 _h)			2号连接 (位配置与1号连接相同)	(与1号连接相同)	
34 (22 _h)			3号连接 (位配置与1号连接相同)	(与1号连接相同)	
35 (23 _h)			4号连接 (位配置与1号连接相同)	(与1号连接相同)	

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称	初始值 (十六进制)	GX Developer 设置适用性	参考章节		
36 (24 _H)	通讯参数设置区	通讯地址设置区	5号连接 (位配置与1号连接相同)	(与1号连接相同)			
37 (25 _H)			6号连接 (位配置与1号连接相同)	(与1号连接相同)			
38 (26 _H)			7号连接 (位配置与1号连接相同)	(与1号连接相同)			
39 (27 _H)			8号连接 (位配置与1号连接相同)	(与1号连接相同)			
40 (28 _H)			1号连接	本地站端口编号	0 _H	○	5.5节
41至42 (29至2A _H)				目标IP地址	0 _H	○	5.5节
43 (2B _H)				目标端口编号	0 _H	○	5.5节
44至46 (2C至2E _H)				目标以太网地址	FFFFFFFF _H	×	-
47至53 (2F至35 _H)			2号连接	(与1号连接相同)			
54至60 (36至3C _H)			3号连接	(与1号连接相同)			
61至67 (3D至43 _H)			4号连接	(与1号连接相同)			
68至74 (44至4A _H)			5号连接	(与1号连接相同)			
75至81 (4B至51 _H)			6号连接	(与1号连接相同)			
82至88 (52至58 _H)			7号连接	(与1号连接相同)			
89至95 (59至5F _H)			8号连接	(与1号连接相同)			
96至102 (60至66 _H)			系统区		-	-	-
103至104 (67至68 _H)		系统区		-	-	-	
105 (69 _H)		通讯状态存储区	初始化处理范围	初始化出错代码	0 _H	×	11.3节
106至107 (6A至6B _H)				本地站IP地址	0 _H	×	-
108至110 (6C至6E _H)				本地站以太网地址	0 _H	×	-
111至115 (6F至73 _H)	系统区			-	×	-	
116 (74 _H)	自动开放UDP端口编号			0 _H	×	-	
117 (75 _H)	系统区			-	-	-	
118 (76 _H)	本地站网络号/站号			0 _H	×	-	
119 (77 _H)	本地站组号			0 _H	×	-	

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节							
120 (78 _H)	通讯状态存储区	连接信息区	1号连接	本地站端口编号	0 _H	×	-						
121至122 (79至7A _H)				目标IP地址	0 _H	×	-						
123 (7B _H)				目标端口编号	0 _H	×	-						
124 (7C _H)				开放出错代码	0 _H	×	11.3节						
125 (7D _H)				固定缓冲存储器发送出错代码	0 _H	×	11.3节						
126 (7E _H)				连接结束代码	0 _H	×	11.3节						
127 (7F _H)				固定缓冲存储器 通讯时间	最大值	0 _H	×	-					
128 (80 _H)					最小值	0 _H	×	-					
129 (81 _H)					当前值	0 _H	×	-					
130至139 (82至8B _H)				2号连接	(与1号连接相同)								
140至149 (8C至95 _H)				3号连接	(与1号连接相同)								
150至159 (96至9F _H)				4号连接	(与1号连接相同)								
160至169 (A0至A9 _H)				5号连接	(与1号连接相同)								
170至179 (AA至B3 _H)				6号连接	(与1号连接相同)								
180至189 (B4至BD _H)				7号连接	(与1号连接相同)								
190至199 (BE至C7 _H)				8号连接	(与1号连接相同)								
200 (C8 _H)				模块状态区	LED亮/灭状态 (存储以太网模块前表面上的LED的亮/灭状态)	<ul style="list-style-type: none"> • [INIT.] LED (b0) 0: 灭 1: 亮(初始化处理完成) • [OPEN] LED (b1) 0: 灭 1: 亮(连接建立处理完成) • [ERR.] LED (b3) 0: 灭 1: 亮(设置出错) • [COM. ERR.] LED (b4) 0: 灭 1: 亮(通讯出错) 非以上的位只供系统使用。				0 _H	×	-	
201 (C9 _H)						网络集线器连接状态区	<ul style="list-style-type: none"> • 通讯模式(b9) 0: 半双工 1: 全双工 • 网络集线器连接状态(b10) 0: 没有连接/断开网络集线器 1: 连接网络集线器 • 数据传送速率(b14) 0: 10BASE-T中运行 1: 100BASE-TX中运行 非以上的位只供系统使用。				0 _H	×	5.10节
202 (CA _H)						开关状态(操作模式设置)	<ul style="list-style-type: none"> 0: 在线 1: 离线 2: 自抑制测试 3: 硬件测试(H/W测试) 				0 _H	○	4.6节

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节		
203 (CB _H)	通讯状态存储区	模块状态区	用 GX Developer 设置的状态 <ul style="list-style-type: none"> • 通讯数据代码设置 (b1) 0: 以二进制代码进行通讯 1: 以 ASCII 代码进行通讯 • 初始化/开放方法设置 (b2) 0: 无参数设置 (按照顺控程序启动) 1: 参数设置 (按照顺控程序设置) • TCP 存在确认设置 (b4) 0: 使用 Ping 1: 使用 KeepAlive • 发送帧设置 (b5) 0: 以太网帧 1: IEEE802.3 帧 • 使用 MC 协议进行 CPU 通讯时序的设置 (b6) 0: 禁止在运行时间写入 1: 允许在运行时间写入 • 被动开放系统 (b8) 0: 不等待打开 (不能在停止时间通讯) 1: 始终等待打开 (能够在停止时间通讯) 非以上的位只供系统使用。	0 _H	○	4.7 节		
204 (CC _H)			发送/接收 指令区	系统区	-	-	-	
205 (CD _H)				RECV 指令执行请求	0 _H	×	应用篇的 第 4 章	
206 (CE _H)				系统区	-	-	-	
207 (CF _H)				执行每个通 讯的数据链 接指令	ZNRD 指令的执行结果	0 _H	×	应用篇的 第 4 章
208 (D0 _H)					系统区	-	-	-
209 (D1 _H)					ZNWR 指令的执行结果	0 _H	×	应用篇的 第 4 章
210 至 223 (D2 至 DF _H)					系统区	-	-	-

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节	
224 至 226 (E0 至 E2 _h)	出错日志区	系统区		-	-	-	
227 (E3 _h)		出错数		0 _h	×	11.3 节	
228 (E4 _h)		出错日志的写指针					
229 (E5 _h)		出错日志块区	1 号出错 记录块	出错代码/结束代码	0 _h	×	11.3 节
230 (E6 _h)				副标题			
231 (E7 _h)				命令代码			
232 (E8 _h)				连接号			
233 (E9 _h)				本地站端口编号			
234 至 235 (EA 至 EB _h)				目标 IP 地址			
236 (EC _h)				目标端口编号			
237 (ED _h)				系统区			
238 至 246 (EE 至 F6 _h)			2 号出错 记录块	(与 1 号出错日志块相同)			
247 至 255 (F7 至 FF _h)			3 号出错 记录块	(与 1 号出错日志块相同)			
256 至 264 (100 至 108 _h)			4 号出错 记录块	(与 1 号出错日志块相同)			
265 至 273 (109 至 111 _h)			5 号出错 记录块	(与 1 号出错日志块相同)			
274 至 282 (112 至 11A _h)			6 号出错 记录块	(与 1 号出错日志块相同)			
283 至 291 (11B 至 123 _h)			7 号出错 记录块	(与 1 号出错日志块相同)			
292 至 300 (124 至 12C _h)			8 号出错 记录块	(与 1 号出错日志块相同)			
301 至 309 (12D 至 135 _h)			9 号出错 记录块	(与 1 号出错日志块相同)			
310 至 318 (136 至 13E _h)		10 号出错 记录块	(与 1 号出错日志块相同)				
319 至 327 (13F 至 147 _h)		11 号出错 记录块	(与 1 号出错日志块相同)				
328 至 336 (148 至 150 _h)		12 号出错 记录块	(与 1 号出错日志块相同)				
337 至 345 (151 至 159 _h)		13 号出错 记录块	(与 1 号出错日志块相同)				
346 至 354 (15A 至 162 _h)		14 号出错 记录块	(与 1 号出错日志块相同)				
355 至 363 (163 至 16B _h)		15 号出错 记录块	(与 1 号出错日志块相同)				
364 至 372 (16C 至 174 _h)		16 号出错 记录块	(与 1 号出错日志块相同)				
373 至 375 (175 至 177 _h)		系统区		-	-	-	

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节		
376 至 377 (178 至 179 _h)	出错日志区	每个协议的状态	IP	接收的 IP 包计数	0 _h	×	11.3 节	
378 至 379 (17A 至 17B _h)				由于和数校验出错, 废除的已接收 IP 包计数				
380 至 381 (17C 至 17D _h)				发送的 IP 包总计数				
382 至 397 (17E 至 18D _h)					系统区	-	-	-
398 至 399 (18E 至 18F _h)					同时传送出错检测计数	0 _h	×	11.4 节 要点(3)
400 至 407 (190 至 197 _h)					系统区	-	-	
408 至 409 (198 至 199 _h)				ICMP	接收的 ICMP 包计数	0 _h	×	
410 至 411 (19A 至 19B _h)			由于和数校验出错, 废除的已接收 ICMP 包计数					
412 至 413 (19C 至 19D _h)			发送的 ICMP 包总计数					
414 至 415 (19E 至 19F _h)			接收 ICMP 包的回声请求总计数					
416 至 417 (1A0 至 1A1 _h)			发送的 ICMP 包的回声应答总计数					
418 至 419 (1A2 至 1A3 _h)			发送的 ICMP 包的回声请求总计数					
420 至 421 (1A4 至 1A5 _h)			接收的 ICMP 包的回声应答总计数					
422 至 439 (1A6 至 1B7 _h)			系统区		-			
440 至 441 (1B8 至 1B9 _h)				TCP	接收的 TCP 信息包计数	0 _h	×	-
442 至 443 (1BA 至 1BB _h)			由于和数校验出错, 废除的已接收 TCP 信息包计数					
444 至 445 (1BC 至 1BD _h)			发送的 TCP 信息包总计数					
446 至 471 (1BE 至 1D7 _h)			系统区		-			
472 至 473 (1D8 至 1D9 _h)				UDP	接收的 UDP 信息包计数	0 _h	×	
474 至 475 (1DA 至 1DB _h)			由于和数校验出错, 废除的已接收 UDP 信息包计数					
476 至 477 (1DC 至 1DD _h)			发送的 UDP 信息包总计数					
478 至 481 (1DE 至 1E1 _h)			系统区		-			
482 至 491 (1E2 至 1EB _h)					系统区	-	-	
492 至 493 (1EC 至 1ED _h)				接收出错	组帧出错计数	0 _h	×	
494 至 495 (1EE 至 1EF _h)			溢出计数		0 _h			
496 至 497 (1F0 至 1F1 _h)			crc 出错计数		0 _h			
498 至 511 (1F2 至 1FF _h)					系统区	-	-	

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称	初始值 (十六进制)	GX Developer 设置适用性	参考章节	
512 至 513 (200 至 201 _H)	路由器信息 设置区	子网掩码	0 _H	○	5.3 节	
514 至 515 (202 至 203 _H)		默认的路由器 IP 地址				
516 (204 _H)		注册的路由器编号				
517 至 518 (205 至 206 _H)		1 号路由器	子网地址	0 _H	○	5.3 节
519 至 520 (207 至 208 _H)			路由器 IP 地址			
521 至 524 (209 至 20C _H)		2 号路由器	(与 1 号路由器相同)			
525 至 528 (20D 至 210 _H)		3 号路由器	(与 1 号路由器相同)			
529 至 532 (211 至 214 _H)		4 号路由器	(与 1 号路由器相同)			
533 至 536 (215 至 218 _H)		5 号路由器	(与 1 号路由器相同)			
537 至 540 (219 至 21C _H)		6 号路由器	(与 1 号路由器相同)			
541 至 544 (21D 至 220 _H)		7 号路由器	(与 1 号路由器相同)			
545 至 548 (221 至 224 _H)		8 号路由器	(与 1 号路由器相同)			
549 (225 _H)		系统区		-	-	-
550 至 551 (226 至 227 _H)		系统区		-	-	-
552 (228 _H)	转换表数据编号		0 _H	○	-	
553 至 554 (229 至 22A _H)	站号<-> IP 关联信息 路由中继参数设置区	1 号转换信息	通讯请求目标/源站网络编号和站号	0 _H	○	应用篇的 第 3 章
555 至 556 (22B 至 22C _H)			以太网站中以太网模块的 IP 地址			
557 至 558 (22D 至 22E _H)		系统区	-	-	-	
559 至 564 (22F 至 234 _H)	2 号转换信息	(与 1 号转换信息相同)				
to	至					
931 至 936 (3A3 至 3A8 _H)	64 号转换信息	(与 1 号转换信息相同)				
937 至 938 (3A9 至 3AA _H)	MELSECNET/H、MELSECNET/10 路由网络掩码型式		0 _H	○	应用篇的 第 3 章	
939 至 943 (3AB 至 3AF _H)	系统区		-	-	-	
944 至 949 (3B0 至 3B5 _H)	FTP 设置区	FTP 注册名称	"QJ71E71"	○	应用篇的 第 3 章	
950 至 953 (3B6 至 3B9 _H)		口令	"QJ71E71"			
954 (3BA _H)		命令输入监视定时器	708 _H			
955 (3BB _H)		PLC CPU 监视定时器	A _H			
956 至 1663 (3BC 至 67F _H)		系统区	-			-

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节
1664 (680 _h)	固定缓冲存储器 数据区	1号固定缓冲 存储器	数据长度	0 _h	×	第7章、 第8章
1665至2687 (681至A7F _h)			固定缓冲存储器数据			
2688至3711 (A80至E7F _h)		2号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
3712至4735 (E80至127F _h)		3号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
4736至5759 (1280至167F _h)		4号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
5760至6783 (1680至1A7F _h)		5号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
6784至7807 (1A80至1E7F _h)		6号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
7808至8831 (1E80至227F _h)		7号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
8832至9855 (2280至267F _h)		8号固定缓冲 存储器	(与1号固定缓冲存储器相同)			
9856至16383 (2680至3FFF _h)	随机访问缓冲存 储器和电子邮件 缓冲存储器的共 享区	随机访问缓冲存储器和电子邮件缓冲存储器的共享区 1) 与随机访问缓冲存储器通讯时, 参考第9章。 2) 使用电子邮件功能时, 参考用户手册(应用篇)中的第2章。		0 _h	×	请参见左边
16384至20479 (4000至4FFF _h)		系统区		-	-	-
20480 (5000 _h)	连接状态存储区	连接状态信息区	开放完成信号 0: 开放未完成 1: 开放完成 • 1号连接(b0) • 2号连接(b1) 至 • 16号连接(b15)	0 _h	×	5.6节
20481 (5001 _h)			系统区	-	-	-
20482 (5002 _h)		连接状态信息区	开放请求信号 0: 无开放请求 1: 正在请求开放 • 1号连接(b0) • 2号连接(b1) 至 • 16号连接(b15)	0 _h	×	5.6节
20483至20484 (5003至5004 _h)		系统区	-	-	-	
20485 (5005 _h)		固定缓冲存储器 信息区	固定缓冲存储器接收状态信号 0: 数据未接收 1: 数据正在接收 • 1号连接(b0) • 2号连接(b1) 至 • 16号连接(b15)	0 _h	×	第7章
20486 (5006 _h)	远程口令状态存 储区	远程口令状态 0: 解锁状态/无远程口令设置 1: 锁定状态 • 1号连接(b0) • 2号连接(b1) 至 • 16号连接(b15)	0 _h	×	5.9节	

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节	
20487 (5007 _H)	系统端口信息区	远程口令状态 存储区	远程口令状态 0: 解锁状态/无远程口令设置 1: 锁定状态 • 自动开放 UDP 端口 (b0) • GX Developer (UDP 端口) (b1) • GX Developer (TCP 端口) (b2) • FTP 端口 (b3)	0 _H	×	5.9 节	
20488 (5008 _H)		系统端口禁用 指定区	系统端口禁用指定 0: 允许使用 1: 禁用 • 自动开放 UDP 端口 (b0) • GX Developer (UDP 端口) (b1) • GX Developer (TCP 端口) (b2)	0 _H	×		
20489 至 20591 (5009 至 506F _H)		系统区		-	-	-	
20592 (5070 _H)	监视区	远程口令功能 监视区	远程口令不相符的通知累计计数指定 (用于用户开放端口, 自动开放 UDP 端口) 0 : 未指定 1 或更高 : 通知累计的计数	1 _H	×	5.9.6 节	
20593 (5071 _H)			远程口令不相符的通知累计计数指定 (用于 GX Developer 通讯端口 (TCP、UDP), FTP 通讯 端口) 0 : 未指定 1 或更高 : 通知累计的计数	2 _H			
20594 (5072 _H)			1 号连接	解锁处理正常完成的累计计数			0 _H
20595 (5073 _H)				解锁处理异常完成的累计计数			
20596 (5074 _H)				锁定处理正常完成的累计计数			
20597 (5075 _H)				锁定处理异常完成的累计计数			
20598 (5076 _H)				依据关闭进行的锁定处理累计计数			
20599 至 20603 (5077 _H 至 507B _H)			2 号连接	(与 1 号连接相同)			
20604 至 20608 (507C _H 至 5080 _H)			3 号连接	(与 1 号连接相同)			
20609 至 20613 (5081 _H 至 5085 _H)			4 号连接	(与 1 号连接相同)			
20614 至 20618 (5086 _H 至 508A _H)			5 号连接	(与 1 号连接相同)			
20619 至 20623 (508B _H 至 508F _H)			6 号连接	(与 1 号连接相同)			
20624 至 20628 (5090 _H 至 5094 _H)			7 号连接	(与 1 号连接相同)			
20629 至 20633 (5095 _H 至 5099 _H)			8 号连接	(与 1 号连接相同)			
20634 至 20638 (509A _H 至 509E _H)			9 号连接	(与 1 号连接相同)			
20639 至 20643 (509F _H 至 50A3 _H)			10 号连接	(与 1 号连接相同)			
20644 至 20648 (50A4 _H 至 50A8 _H)			11 号连接	(与 1 号连接相同)			

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节	
20649 至 20653 (50A9 _h 至 50AD _h)	监视区	远程口令功能监视区	12号连接	(与1号连接相同)			
20654 至 20658 (50AE _h 至 50B2 _h)			13号连接	(与1号连接相同)			
20659 至 20663 (50B3 _h 至 50B7 _h)			14号连接	(与1号连接相同)			
20664 至 20668 (50B8 _h 至 50BC _h)			15号连接	(与1号连接相同)			
20669 至 20673 (50BD _h 至 50C1 _h)			16号连接	(与1号连接相同)			
20674 至 20678 (50C2 _h 至 50C6 _h)			自动开放UDP 端口	(与1号连接相同)			
20679 至 20683 (50C7 _h 至 50CB _h)			GX Developer 通讯UDP 端口	(与1号连接相同)			
20684 至 20688 (50CC _h 至 50D0 _h)			GX Developer 通讯UDP 端口	(与1号连接相同)			
20689 至 20693 (50D1 _h 至 50D5 _h)			FTP 通讯端口	(与1号连接相同)			
20694 至 20736 (50D6 _h 至 5100 _h)			状态存储区	系统区		—	—
20737 (5101 _h)	出错日志指针			0 _h	×	11.3 节	
20738 (5102 _h)	日志计数器(HTTP 响应代码 100 到 199)						
20739 (5103 _h)	日志计数器(HTTP 响应代码 200 到 299)						
20740 (5104 _h)	日志计数器(HTTP 响应代码 300 到 399)						
20741 (5105 _h)	日志计数器(HTTP 响应代码 400 到 499)						
20742 (5106 _h)	日志计数器(HTTP 响应代码 500 到 599)						
20743 (5107 _h)	系统区			—	—	—	
20744 (5108 _h)	1号出错日志块	HTTP 响应代码		0 _h	×	11.3 节	
20745 至 20746 (5109 至 510A _h)		目标 IP 地址					
20747 至 20750 (510B 至 510E _h)		出错时间					
20751 至 20757 (510F 至 5115 _h)	2号出错日志块			(与1号出错日志块相同)			
20758 至 20764 (5116 至 511C _h)	3号出错日志块			(与1号出错日志块相同)			
20765 至 20771 (511D 至 5123 _h)	4号出错日志块			(与1号出错日志块相同)			
20772 至 20778 (5124 至 512A _h)	5号出错日志块			(与1号出错日志块相同)			
20779 至 20785 (512B 至 5131 _h)	6号出错日志块			(与1号出错日志块相同)			
20786 至 20792 (5132 至 5138 _h)	7号出错日志块			(与1号出错日志块相同)			

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节	
20793 至 20799 (5139 至 513F _H)	状态存储区	8号出错日志块	(与1号出错日志块相同)				
20800 至 20806 (5140 至 5146 _H)		9号出错日志块	(与1号出错日志块相同)				
20807 至 20813 (5147 至 514D _H)		10号出错日志块	(与1号出错日志块相同)				
20814 至 20820 (514E 至 5154 _H)		11号出错日志块	(与1号出错日志块相同)				
20821 至 20827 (5155 至 515B _H)		12号出错日志块	(与1号出错日志块相同)				
20828 至 20834 (515C 至 5162 _H)		13号出错日志块	(与1号出错日志块相同)				
20835 至 20841 (5163 至 5169 _H)		14号出错日志块	(与1号出错日志块相同)				
20842 至 20848 (516A 至 5170 _H)		15号出错日志块	(与1号出错日志块相同)				
20849 至 20855 (5171 至 5177 _H)		16号出错日志块	(与1号出错日志块相同)				
20856 至 20991 (5178 至 51FF _H)	禁止使用	系统区		-	-	-	
20992 (5200 _H)	“断开检测超时的发送系统切换请求”状态存储区	“电缆断开时发送系统切换请求” 0: 未设置 1: 设置		1 _H	○	5.11 节	
20993 (5201 _H)		电缆断开超时设置 设置时间 = 设置值 × 500ms (设置范围: 0 至 60)		4 _H	○	5.11 节	
20994 (5202 _H)		系统区		-	-	-	
20995 (5203 _H)		断开检测计数		0 _H	×	5.10 节	
20996 至 21007 (5204 至 520F _H)	用户禁止	系统区		-	-	-	
21008 (5210 _H)	“发生通讯错误的系统切换设置”	“发生通讯错误的系统切换设置” (用户连接) 0: 未设置 1: 设置 • 1号连接(b0) • 2号连接(b1) 至 • 16号连接(b15)		0 _H	○	5.11 节	
21009 (5211 _H)		“发生通讯错误的系统切换设置” (用户连接) 0: 未设置 1: 设置 • 自动开放端口设置 • GX Developer (UDP 端口) (b1) • GX Developer (TCP 端口) (b2) * • FTP 端口 (b3) • HTTP 端口 (b4) * 包括 MELSOFT 连接		0 _H	○	5.11 节	
21010 至 22559 (5212 至 581F _H)	用户禁止	系统区		-	-	-	
22560 (5820 _H)	通讯状态存储区	连接信息区	9号连接	本地站端口号	0 _H	×	-
22561 至 22562 (5821 至 5822 _H)				目标 IP 地址	0 _H	×	-
22563 (5823 _H)				目标端口编号	0 _H	×	-
22564 (5824 _H)				开放出错代码	0 _H	×	11.3 节

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节					
22565 (5825 _H)	通讯状态存储区	连接信息区	9号连接	固定缓冲存储器发送出错代码	0 _H	×	11.3节				
22566 (5826 _H)				连接结束代码	0 _H	×	11.3节				
22567 (5827 _H)				固定缓冲存储器 通讯时间	最大值	0 _H	×	-			
22568 (5828 _H)					最小值	0 _H	×	-			
22569 (5829 _H)					当前值	0 _H	×	-			
22570至22579 (582A至5833 _H)			10号连接	(与9号连接相同)							
22580至22589 (5834至583D _H)			11号连接	(与9号连接相同)							
22590至22599 (583E至5847 _H)			12号连接	(与9号连接相同)							
22600至22609 (5848至5851 _H)			13号连接	(与9号连接相同)							
22610至22619 (5852至585B _H)			14号连接	(与9号连接相同)							
22620至22629 (585C至5865 _H)			15号连接	(与9号连接相同)							
22630至22639 (5866至586F _H)			16号连接	(与9号连接相同)							
22640 (5870 _H)			电子邮件状态存储区	接收	服务器上剩余的邮件数目		0 _H	×	11.3节		
22641 (5871 _H)					专用指令正常完成计数						
22642 (5872 _H)					专用指令异常完成计数						
22643 (5873 _H)					正常接收计数						
22644 (5874 _H)	附件接收计数										
22645 (5875 _H)	服务器查询计数										
22646 (5876 _H)	服务器通讯出错计数										
22647 (2877 _H)	出错日志写入计数										
22648 (5878 _H)	接收出错日志写入指针										
22649 (5879 _H)	1号出错 记录块	出错代码									
22650 (587A _H)		命令代码									
22651至22658 (587B至5882 _H)		来自									
22659至22662 (5883至5886 _H)		日期									
22663至22692 (5887至58A4 _H)	标题				0 _H	×	11.3节				
22693至22736 (58A5至58D0 _H)	2号出错 记录块	(与1号出错日志块相同)									
22737至22780 (58D1至58FC _H)	3号出错 记录块	(与1号出错日志块相同)									
22781至22824 (58FD至5928 _H)	4号出错 记录块	(与1号出错日志块相同)									

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节	
22825 至 22868 (5929 至 5954 _H)	接收	5号出错 记录块	(与1号出错日志块相同)				
22869 至 22912 (5955 至 5980 _H)		6号出错 记录块	(与1号出错日志块相同)				
22913 至 22956 (5981 至 59AC _H)		7号出错 记录块	(与1号出错日志块相同)				
22957 至 23000 (59AD 至 59D8 _H)		8号出错 记录块	(与1号出错日志块相同)				
23001 至 23044 (59D9 至 5A04 _H)		9号出错 记录块	(与1号出错日志块相同)				
23045 至 23088 (5A05 至 5A30 _H)		10号出错 记录块	(与1号出错日志块相同)				
23089 至 23132 (5A31 至 5A5C _H)		11号出错 记录块	(与1号出错日志块相同)				
23133 至 23176 (5A5D 至 5A88 _H)		12号出错 记录块	(与1号出错日志块相同)				
23177 至 23220 (5A89 至 5AB4 _H)		13号出错 记录块	(与1号出错日志块相同)				
23221 至 23264 (5AB5 至 5AE0 _H)		14号出错 记录块	(与1号出错日志块相同)				
23265 至 23308 (5AE1 至 5B0C _H)		15号出错 记录块	(与1号出错日志块相同)				
23309 至 23352 (5B0D 至 5B38 _H)		16号出错 记录块	(与1号出错日志块相同)				
23353 (5B39 _H)		电子邮件状态存储区	专用指令正常完成计数		0 _H	×	11.3节
23354 (5B3A _H)			专用指令异常完成计数				
23355 (5B3B _H)			正常完成的邮件数目				
23356 (5B3C _H)			附件发送计数				
23357 (5B3D _H)	发送到服务器计数						
23358 (5B3E _H)	异常完成的邮件数目						
23359 (5B3F _H)	出错日志写计数						
23360 (5B40 _H)	出错日志写指针						
23361 (5B41 _H)	1号出错 记录块		出错代码				
23362 (5B42 _H)			命令代码				
23363 至 23370 (5B43 至 5B4A _H)		到					
23371 至 23374 (5B4B 至 5B4E _H)		日期					
23375 至 23404 (5B4F 至 5B6C _H)	标题		0 _H	×	11.3节		
23405 至 23448 (5B6D 至 5B98 _H)	2号出错 记录块	(与1号出错日志块相同)					
23449 至 23492 (5B99 至 5BC4 _H)	3号出错 记录块	(与1号出错日志块相同)					
23493 至 23536 (5BC5 至 5BF0 _H)	4号出错 记录块	(与1号出错日志块相同)					
23537 至 23580 (5BF1 至 5C1C _H)	5号出错 记录块	(与1号出错日志块相同)					

(转下页)

(续上页)

地址 十进制 (十六进制)	应用	名称		初始值 (十六进制)	GX Developer 设置适用性	参考章节
23581 至 23624 (5C1D 至 5C48 _H)	电子邮件状态存储区	发送	6号出错记录块	(与1号出错日志块相同)		
23625 至 23668 (5C49 至 5C74 _H)			7号出错记录块	(与1号出错日志块相同)		
23669 至 23712 (5C75 至 5CA0 _H)			8号出错记录块	(与1号出错日志块相同)		
23713 至 24575 (5CA1 至 5FFF _H)		系统区		-	-	-
24576 (6000 _H)	固定缓冲存储数据区	9号固定缓冲存储器	数据长度	0 _H	×	-
24577 至 25599 (6001 至 63FF _H)			固定缓冲存储区			
25600 至 26623 (6400 至 67FF _H)		10号固定缓冲存储器	(与9号固定缓冲存储器相同)			
26624 至 27647 (6800 至 6BFF _H)		11号固定缓冲存储器	(与9号固定缓冲存储器相同)			
27648 至 28671 (6C00 至 6FFF _H)		12号固定缓冲存储器	(与9号固定缓冲存储器相同)			
28672 至 29695 (7000 至 73FF _H)		13号固定缓冲存储器	(与9号固定缓冲存储器相同)			
29696 至 30719 (7400 至 77FF _H)		14号固定缓冲存储器	(与9号固定缓冲存储器相同)			
30720 至 31743 (7800 至 7BFF _H)		15号固定缓冲存储器	(与9号固定缓冲存储器相同)			
31744 至 32767 (7C00 至 7FFF _H)		16号固定缓冲存储器	(与9号固定缓冲存储器相同)			

4 操作之前的设置和步骤

本章说明在系统中开始操作以太网模块之前所需的设置和步骤。

4.1 装载和安装

本节说明处理以太网模块时，包括从拆箱到安装及安装环境等的注意事项，对所有模块而言是通用的。

关于模块装载和安装的更多详细说明，请参阅使用 PLC CPU 模块的用户手册。

4.1.1 处理注意事项

以下说明处理以太网模块时的注意事项：

- (1) 由于以太网模块盒是由树脂制成的，因此应该防止跌落或受到任何震动。
- (2) 在下面的拧紧扭矩范围之内紧固模块端子和固定螺钉：

螺钉	夹紧扭矩范围
电源线连接端子螺钉 (M2.5 螺钉) (* ¹)	0.40N·m
模块固定螺钉 (标准使用时不需要) (M3 螺钉) (* ²)	0.36 至 0.48N·m

*1 该端子连接到 10BASE5 时，用作外部电源输入端子，对收发器供电。

*2 使用模块上部的挂钩，可以方便地将模块固定到基板上。但是，在可能有强烈震动或冲击的地方使用时，建议你使用模块固定螺钉来安装模块。



危险

- 通电时，不要触摸端子和连接器。
否则可能遭到电击或引起故障。
- 在清洁或重新拧紧螺钉之前，必须断开外部电源的所有相位。否则可能损坏模块或使模块发生故障。
如果螺钉松动，可能导致模块短路、故障或跌落。
螺钉拧得过紧可能损坏螺钉和/或模块，并导致模块短路、故障或跌落。
- 在开始对运行中的 PLC 进行控制操作之前，要彻底检查安全性。(尤其是对数据、程序和运行状态(状态控制)的修改)。



注意

- 安装模块时，按住模块下部的安装杆，将模块插入基板安装孔中的紧固锁扣中固定好。
如果模块安装不当，可能导致模块故障、失效或跌落。
在可能有连续震动的地方使用模块时，尤其要用螺钉紧固模块。
- 一定要小心，不要让任何异物如接线碎片等进入模块内部。
它们可能导致火灾，并有可能损坏模块并使模块故障。
- 不要拆开或改造模块。
否则可能损坏模块，引起故障、人身伤害或火灾。
- 在安装或拆卸模块之前，确保断开外部电源的所有相位。
否则可能导致损坏模块或使模块故障。
- 使用规定的扭矩拧紧端子螺钉。
如果端子螺钉松动，可能导致短路、故障或跌落。
螺钉拧得过紧，可能损坏螺钉和/或模块，并导致模块短路、故障或跌落。
- 不要直接触摸模块的导电部件和电子部件。
否则可能导致模块故障或失效。
- 当处置该产品时，把它作为工业废料处理。
- 接线时，为了防止异物如接线碎片等进入模块内部，在模块上部粘有一层防护膜。
接线期间不要取下该防护膜。
然而，一定要在操作模块之前拿走该防护膜，以使操作期间产生的热量散发。

4

4.1.2 安装环境

本节说明了 PLC 的安装环境。当安装 PLC 时，必须避免下列环境：

- 环境温度超过 0 到 55℃ 范围的地方。
- 环境湿度超过 5 到 95%RH 范围的地方。
- 由于温度骤变发生凝结的地方。
- 存在腐蚀或易燃气体的地方。
- 暴露在大量粉末状传导物质的地方，如灰尘和锉屑、油雾、盐份或有机溶剂。
- 暴露在阳光直射的地方。
- 暴露在强电场或强磁场的地方。
- 主模块会直接受到震动或冲击的地方。

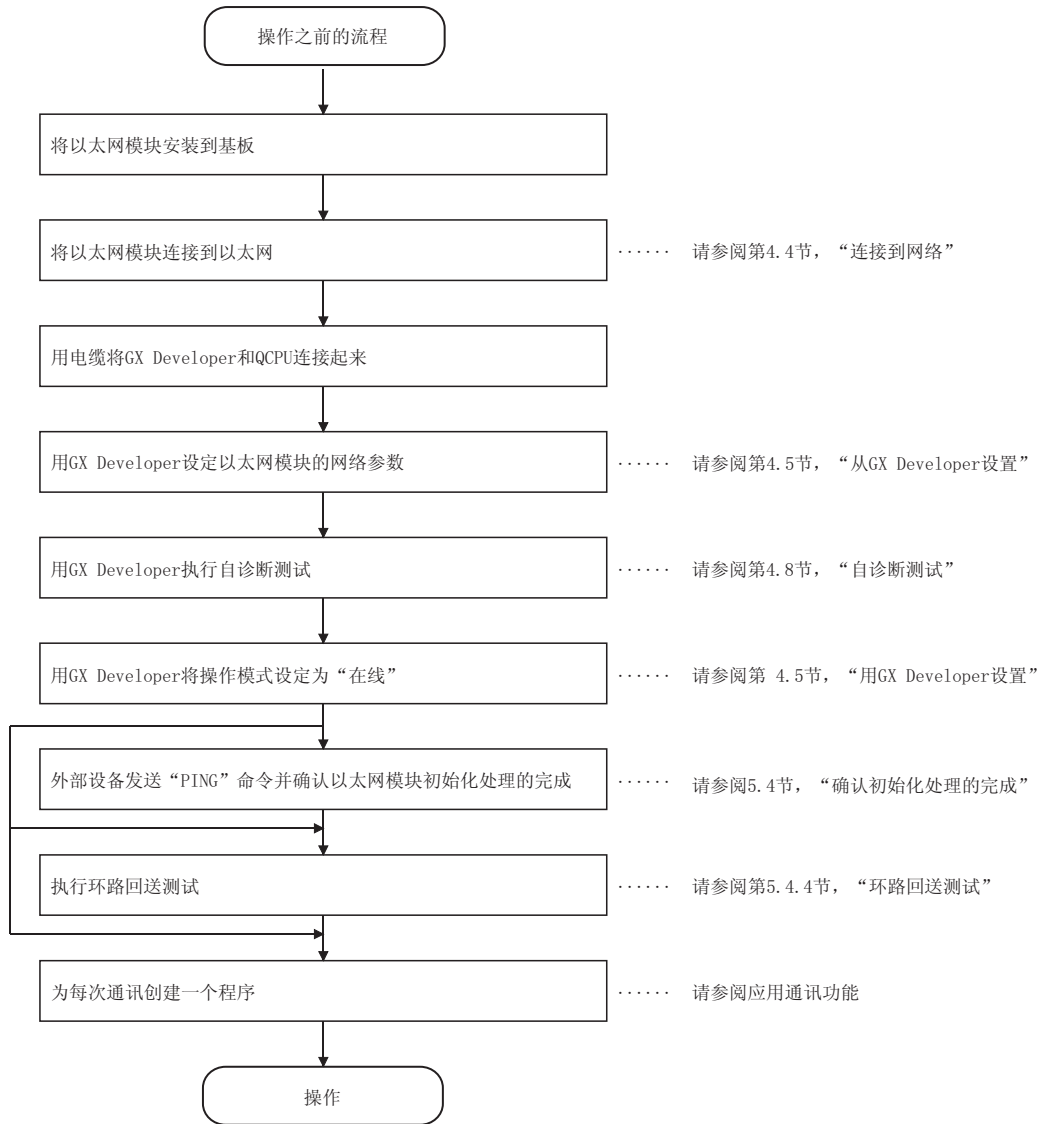


注意

- 在符合 CPU 模块用户手册中所述的通用规格的操作环境中使用 PLC。
在不符合通用规格的操作环境下使用 PLC，可能导致电击、火灾或故障，也可能损坏模块或使模块性能变差。

4.2 开始操作之前的设置和步骤

下面表示的是开始操作之前需要的流程：

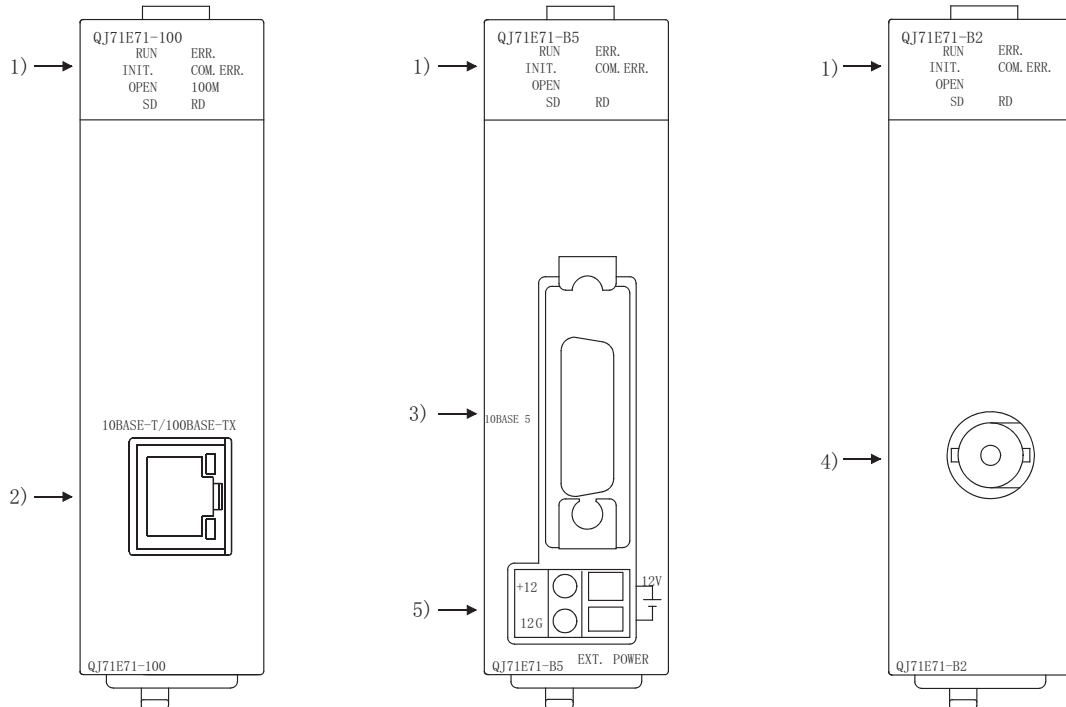


要点
<p>为了在使用 GX Developer 增加或修改参数设置之后能操作以太网模块，必须在参数值保存在 PLC CPU 之后，将 PLC CPU 复位。</p>

要点	
(1)	不要将任何数据写入智能功能模块缓冲存储器的“系统区”中。
(2)	不要输出(接通)“禁用”信号，它是输出信号之一。
(3)	当来自外部设备的状态控制(如远程运行/停止)用于 PLC CPU 时，用户应预先使用选择参数“始终等待打开”进行选择。(通过使用操作设置中的初始时间设置来进行选择。)
	如果选择了“不等待打开”，通讯线路将在远程停止期间断开。自此不能从 PLC CPU 端再次开放，也不能从外部设备起动远程运行。
(4)	当使用为传统机型以太网模块创建的顺控程序时，不要使用开放请求信号(Y8 到 YF)、固定缓冲存储器通讯信号(Y0 到 Y7)和同时用于程序中相同连接的 OPEN/CLOSE 和 BUFSND/BUFRCV 专用指令。它们可能导致故障。
(5)	当更换以太网模块时，也要将外部设备复位。(如果外部设备保留以太网地址，由于更换了模块，以太网地址已改变，所以无法继续通讯。)同样地，当更换了外部设备(个人计算机等)时，也要重新起动以太网模块。

4.3 以太网模块的部件

本节介绍以太网模块的部件。



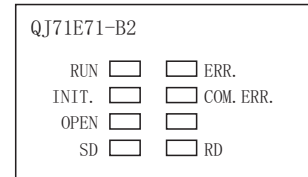
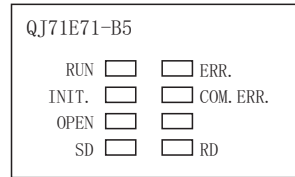
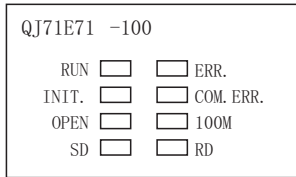
	名称	说明
1)	LED 显示	参阅 LED 显示(1)的内容。
2)	10BASE-T/100BASE-TX 连接器 (RJ45) (*1)	将以太网模块连接到 10BASE-T/100BASE-TX 的连接器。(以太网模块通过网络集线器把 10BASE-T 和 100BASE-TX 区分开来。)
3)	10BASE5 连接器	将以太网模块连接到 10BASE5 的连接器。(用于连接 10BASE5 的 AUI 电缆(收发器电缆))
4)	10BASE2 连接器	将以太网模块连接到 10BASE2 的连接器。(用于连接 10BASE2 同轴电缆)
5)	外部电源端子	当与 10BASE5 连接时, 向收发器供电的电源端子。(13.28V 至 15.75V)

*1 连接器上的 LED 显示不亮。

对于前 5 位系列号是“05059”或之前的连接器, 连接器的方向已经旋转了 180 度。

(1) LED 显示内容(*1)

1) LED显示



LED 名称		显示说明	LED 亮时	LED 灭时
QJ71E71-100	QJ71E71-B5, QJ71E71-B2			
RUN	RUN	正常运行显示	正常	异常
INIT.	INIT.	初始化处理状态显示	正常完成	未处理
OPEN *2	OPEN *2	开放处理状态显示	可使用正常建立的连接	不可用正常开放的连接
SD	SD	数据发送显示	数据正在发送	数据未被发送
ERR.	ERR.	设置异常显示	异常 *3	正常设置
COM. ERR.	COM. ERR.	通讯异常显示	发生通讯异常 *4	正在进行正常通讯
100M	(未使用)	传送速度显示	100Mbps	未连接时 10Mbps
RD	RD	数据接收状态显示	正在接收数据	没有接收到数据

*1 关于显示的出错原因和相应的更正方法，请参阅 11.1.1 节。

*2 根据用户连接 1 到 16 的开放状态，[OPEN]LED 开放/关闭。
(不包括系统连接的开放状态(例如自动开放 UDP 端口)。)

*3 下列情况下 [ERR.]LED 亮：

- 当使用 GX Developer 设置的值(模式、站号和或网络号)不正确时。
- 当以太网模块或 PLC CPU 中发生操作禁止出错时。

*4 关于“COM. ERR”LED 亮的时间，请参阅第 11.1.2 节。

4.4 连接到网络

下面说明如何将以太网模块连接到 100BASE-TX、10BASE-T、10BASE5 和 10BASE2 网络。以下内容也说明了连接以太网模块时应遵守的一些注意事项。一定要注意安全，并正确使用以太网模块。

- (1) 安装 100BASE-TX、10BASE-T、10BASE5 和 10BASE2 网络时，需要采取足够的安全措施。当连接电缆端子或安装中继线电缆等时，请向专家咨询。
- (2) 使用符合第 2.2 节中标准的连接电缆。
- (3) 预置允许的弯曲半径。当用弯曲同轴电缆来连接时，一定要留出比需要的同轴电缆允许半径还要大的空间。
关于同轴电缆允许弯曲半径的资料，请向电缆制造商咨询。



注意

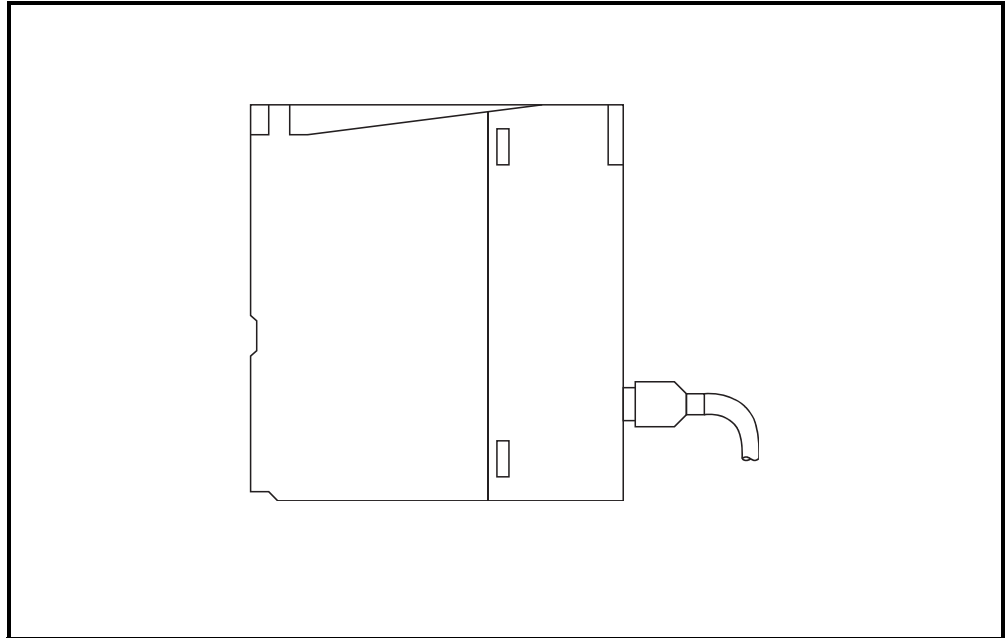
- 安装 AU 电缆 (收发机电缆) 或同轴电缆时，不要捆扎电缆线，或靠近主线路或电源线放置。
至少应相距 100mm (3.94in.) 安装。
否则可能产生噪声，导致故障。
- 在安装了模块的站通电时，不要连接 AUI 电缆。
- 一定要把连接到模块上的通讯电缆和电源电缆敷设在排线管中或用夹子紧固。
如果没有将电缆敷设在排线管中或用夹子紧固，它们的位置可能不稳或会移动，并会受到不经意的拉动。电缆连接不当，可能会损坏模块和电缆或导致模块故障。
- 当断开模块的通讯电缆和电源电缆时，不要用手拉电缆。
当断开带连接器的电缆时，用手抓住模块的连接器，拉出后拆掉电缆。
当断开不带连接器的电缆时，在拆掉电缆之前，首先松开端子排上的螺钉。
如果拉动还连接在模块上的电缆时，可能导致模块故障，或损坏模块和电缆。

4.4.1 连接到 10BASE-T/100BASE-TX 网络

本节说明如何将以太网模块连接到 10BASE-T、100BASE-TX 网络。

(说明的对象模块: QJ71E71-100)

下面表示的是双绞线的连接图。



<操作步骤>

(第 1 步) 将双绞线 (UTP) 连接到网络集线器上。

(第 2 步) 将双绞线 (UTP) 连接到以太网模块上。

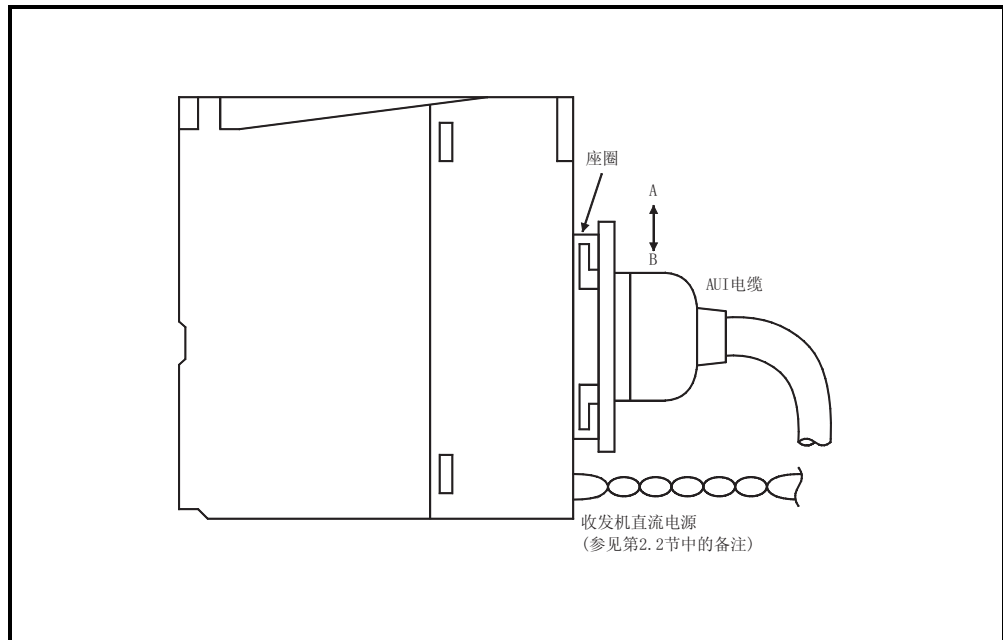
要点
(1) 根据网络集线器的不同, 以太网模块检测是 10BASE-T 还是 100BASE-TX, 全双工还是半双工传送模式。 对于无需自动检测功能就连接到网络集线器, 在网络集线器端选择半双工模式。
(2) 关于连接到 10BASE-T 或 100BASE-TX 时需要的设备以及示例系统配置, 请参阅第 2.2 节的 (1) 和 (2)。

4.4.2 连接到 10BASE5 网络

本节说明如何将以太网模块连接到 10BASE5 网络。

(说明用的对象模块为: QJ71E71-B5)

下面表示的是 AUI 电缆的连接图。



<操作步骤>

(第 1 步) 如上所示, 向着 B 方向滑动座圈。

(第 2 步) 完全按入 AUI 电缆连接器。

(第 3 步) 如上所示, 向着 A 方向滑动座圈。

(第 4 步) 确认 AUI 电缆锁定。

(第 5 步) 向收发机供电(*1)。

*1 使用具有一种通称为 SQE TEST 或“心搏”的功能(收发机功能: 发射信号, 通知通讯结束时收发机是否正常运行)的收发机。



注意

- 当安装了模块的站通电时, 不要连接 AUI 电缆。

要点

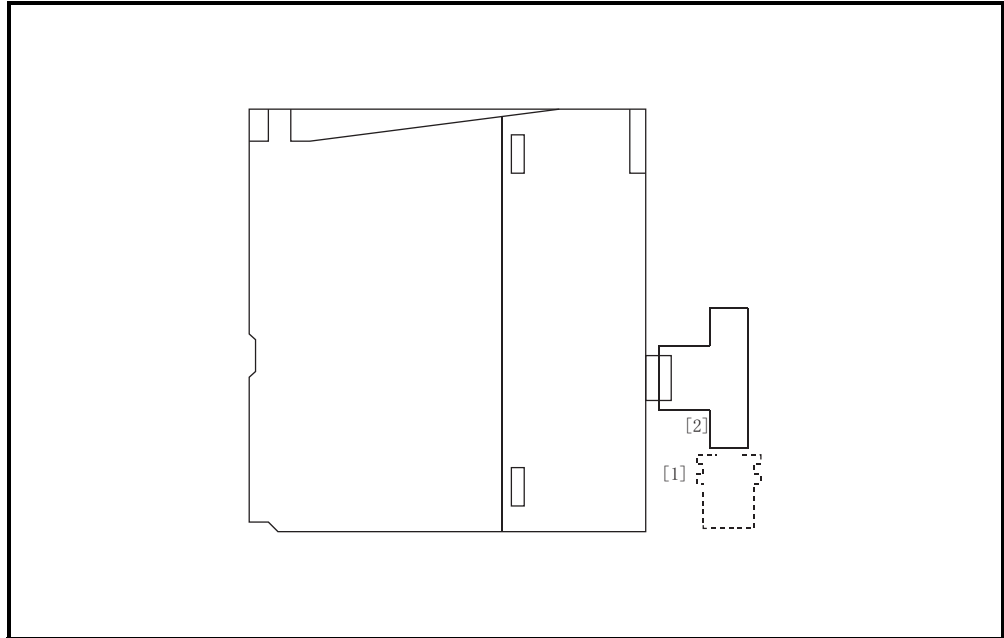
- (1) 当使用 10BASE5 执行与网络的连接时, 必需注意以太网模块安装环境中产生的高频噪声, 将一个铁芯连接到 AUI 电缆中以消除噪声影响。请参阅第 2.2 节 a(2) 中的要点。
- (2) 有关 10BASE5 连接需要的设备和系统配置范例, 请参阅第 2.2 节中的 (2) (b)。

4.4.3 连接到 10BASE2 网络

该节说明如何将以太网模块连接到 10BASE2 网络中。

(说明的对象模块为: QJ71E71-B2)

下面表示的是 10BASE-2 同轴电缆的连接图。



<操作步骤>

(第 1 步) 如上所示, 对准凹槽[1]和突起部[2]推入连接器。

(第 2 步) 推入连接器后, 顺时针旋转 1/4 圈。

(第 3 步) 转动直到连接器锁定。

(第 4 步) 检查连接器锁定。

要点

有关 10BASE2 连接需要的设备和系统配置范例, 请参阅第 2.2 节中的 (3)。
--

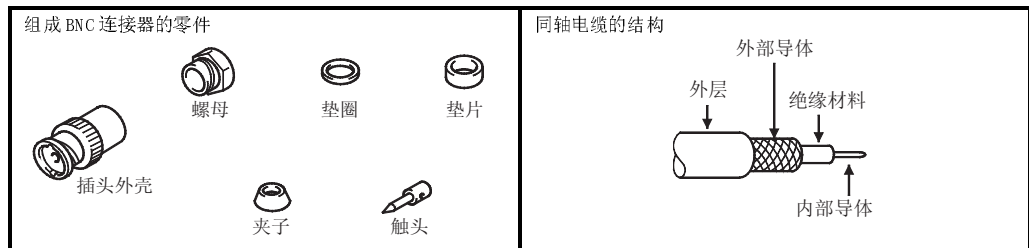
备注

连接同轴电缆的连接器。

以下部分说明了如何将 BNC 连接器(同轴电缆的连接器插头)连接到电缆上。

(1) BNC 连接器和同轴电缆的结构

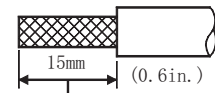
下面表示的是 BNC 连接器和同轴电缆的组成。



(2) 如何连接 BNC 连接器和同轴电缆

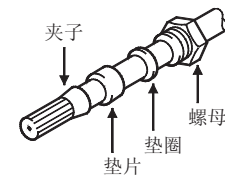
以下说明如何连接 BNC 连接器和同轴电缆。

- (a) 如右图所示将同轴电缆外层割去 15mm。
小心不要损坏外部导体。

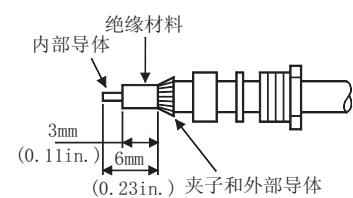


切去该部分的外层

- (b) 如右图所示将螺母、垫圈、垫片和夹子固定到同轴电缆上，并松开外部导体。



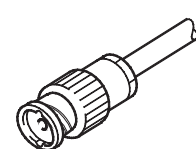
- (c) 按右图所示尺寸切割外部导体、绝缘材料和内部导体。
然而，要把外部导体割成与夹子锥形部分相等的尺寸，并要把它压平套在夹子上。



- (d) 将触头和内部导体焊接起来。



- (e) 将连接器组件(d)插入插头外壳并把螺母旋进插头外壳。



要点

焊接内部导体和触头时注意以下情形：

- (1) 确保焊接部分没有形成焊珠。
- (2) 确保连接器和电缆绝缘体之间没有间隙，并且它们没有互相切入。
- (3) 尽可能快地进行焊接以防绝缘材料变形。

4.5 用 GX Developer 进行设置

本节说明以太网模块适用的 GX Developer 设置屏名称。

按照第 3.6 节“以太网模块 GX Developer 设置项目列表”中使用的功能进行设置。

关于如何显示每个画面的详情，请参阅 GX Developer 操作手册。

4.5.1 I/O 分配设置

[设置目的]

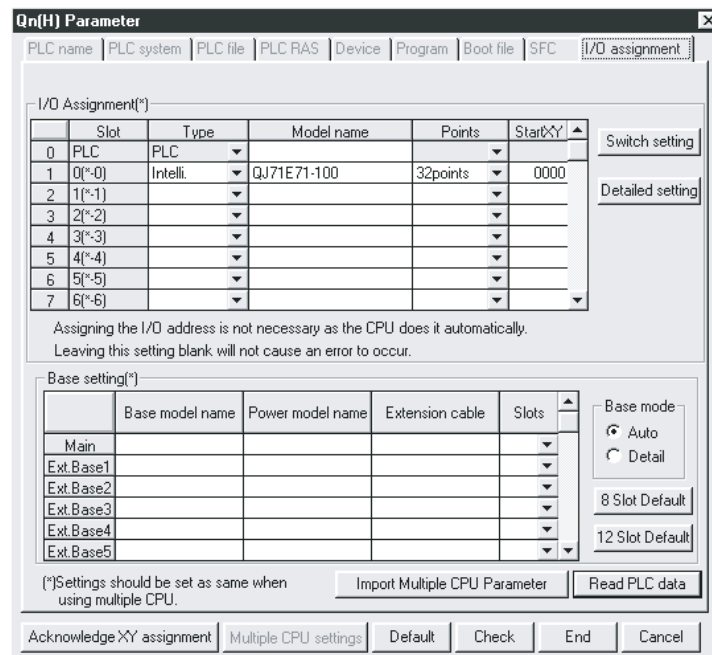
I/O 分配设置是对要安装在基板上的模块类型进行设置和对输入/输出信号范围进行设置。

[起动步骤]

[GX Developer] → [PLC 参数] → **I/O 分配**

关于画面显示，请参阅 GX Developer 的操作手册。

[设置画面]



[显示内容]

项目名称	项目的设置		备注
I/O 分配	类型	选择“智能”。	—
	类型名	输入要安装的模块型号名称(例如: QJ71E71-100)。	
	点数	选择 32 点。	
	起始 XY	输入目标模块的开始 I/O 信号(十六进制)。	
	详细设置	当采用多 CPU 系统时, 选择以太网模块的控制 CPU。	
多 CPU 设置	选择何时使用多 CPU 系统。		参阅 QCPU 用户手册(多 CPU 系统)

4.5.2 其它设置

本节说明了各种以太网模块功能的设置屏名称。关于设置内容，请参阅具体说明相应画面的章节。

- (1) “设置 MNET/10H 以太网卡号的网络参数” (第 4.6 节中作了详细说明。)

该画面用于将以太网模块作为网络模块进行设置。

它也是设置以太网模块“操作设置”和“初始化设置”的主画面。

为了使用以太网模块，必须对该画面上的各设置项进行设置。

	Module 1	Module 2	Module 3
Network type	Ethernet	None	None
Starting I/O No.			
Network No.			
Total stations			
Group No.	0		
Station No.			
Mode	On line		
Operational settings			
Initial settings			
Open settings			
Router relay parameter			
Station No. <-> IP information			
FTP Parameters			
E-mail settings			
Interrupt settings			
Redundant settings			

Necessary setting(No setting / Already set) Set if it is needed(No setting / Already set)

Start I/O No. : Valid module during other station access:

Please input the starting I/O No. of the module in HEX(16 bit) form

Buttons: Acknowledge XY assignment, Routing parameters, Assignment image, Group Settings, Check, End, Cancel

- (2) “操作设置” (第 4.7 节中作了详细说明。)

该画面用于当其它模块使用以太网模块时设置公共项。

因为以太网初始化处理需要该画面上的设置，所以必须对其进行设置。

- (3) “初始设置” (本手册 5.2 节和用户手册(应用篇)第 2 章中作了详细说明。)

该画面用于设置以太网模块中要使用 TCP/IP 通讯的公共定时器值；以及为了使用电子邮件功能，该画面也可用于设置 DNS 服务器。

当使用定时器默认值通讯时，不必设置定时器值。

- (4) “开放设置” (第 5.5 节中作了详细说明)
该画面用于用固定缓冲存储器通讯与外部设备数据通讯时的开放处理设置, 以及处理与缓冲存储器使用相关的设置。
- (5) “路由中继参数(路由中继参数)” (第 5.3 节中作了详细说明)
在该画面上执行下列与外部设备进行数据通讯时的设置:
 - 通过路由器与其它以太网连接的外部设备的数据通讯。
 - 将连接到以太网的设备分成组, 并使用数据链接指令与任意组中的外部设备进行数据通讯。
- (6) “站号<->IP 关联信息(MNET/10 路由中继参数)” (用户手册(应用篇)的第 3 章中作了详细说明)
该设置用于通过以太网或 MELSECNET/H、MELSECNET/10 与其它站上的 PLC CPU 进行的通讯。
- (7) “FTP 参数” (用户手册(应用篇)的第 5 章中作了详细说明)
该设置用于使用文件传送(FTP 服务器)功能。
通过使用文件传送功能, 外部设备可以从安装了以太网模块的 QCPU 的文件中读取数据或将数据写入文件。
- (8) “电子邮件设置” (用户手册(应用篇)的第 2 章中作了详细说明)
这些设置用于使用电子邮件传送/接收和自动消息(通知)功能。
- (9) “发送邮件地址设置” (用户手册(应用)的第 2 章中作了详细说明)
本节用于设置当使用电子邮件发送/接收功能时的发送目标邮件地址。
- (10) “消息设置” (用户手册(应用篇)的第 2 章中作了详细说明)
在该画面上, 设置使用电子邮件功能的消息(通知)PLC CPU 监视结果。

- (11) “中断设置”和“中断指针设置”(第 7.3 节中作了详细说明)
在该画面上,通过 PLC CPU 中断程序,使用下列指令,执行读接收数据的设置。
- 使用专用指令(BUFRCVS)读固定缓冲存储器通讯中的接收数据。
 - 使用数据链接指令(RECVS)读从另一站 PLC CPU 传送的数据。
- (12) “冗余设置”,“组设置”(详细说明:5.11 节,QCPU 用户手册)
冗余系统中使用以太网模块的设置。
- (13) “路由参数”(用户手册(应用篇)的第 3 章中作了详细说明)
该画面用于设置通过以太网或 MELSECNET/H、MELSECNET/10 与其它站的 PLC CPU 进行通讯时通过哪个站。
- (14) “多 CPU 设置”(QCPU 用户手册中作了详细说明)
该画面用于设置在多 CPU 系统中,使用以太网模块时,以太网模块的控制 CPU。
- (15) “远程口令”(第 5.9.5 节中作了详细说明)
该画面用于设置使用 QCPU 远程口令功能时,由以太网模块执行远程口令核对的连接。

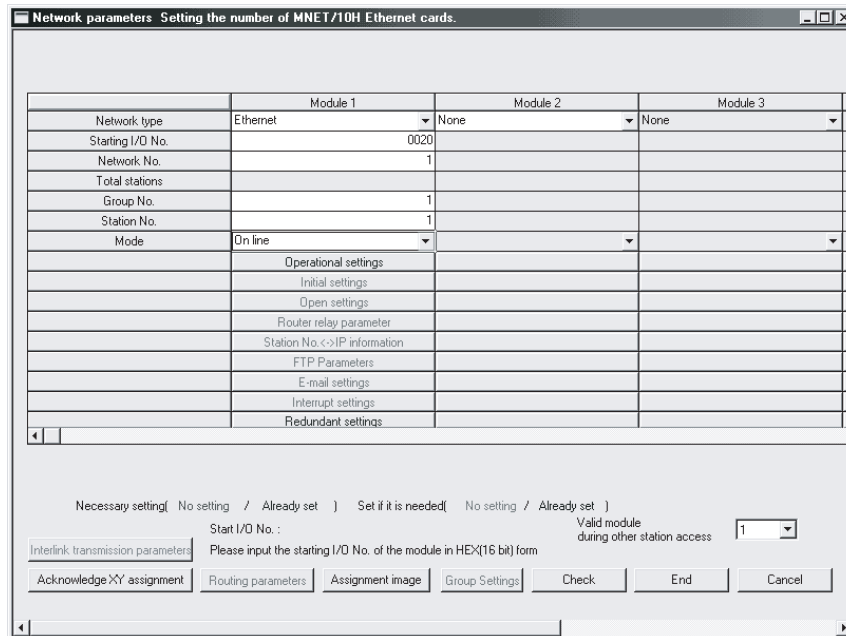
备注

无需用 GX Developer I/O 分配设置“智能功能模块开关设置”。
在上述“操作设置”、“初始设置”和“开放设置”中执行对应于开关设置的各种设置。

4.6 设置 MNET/10H 以太网卡号的网络参数

本节说明了设置 MNET1/10H 以太网卡号的网络参数之一。

通过选择[GX Developer]-[网络参数]启动“设置 MNET/100H 以太网卡号的网络参数”画面。关于如何显示画面的详细说明，请参阅 GX Developer 操作手册。



项目名称	设置的说明	设置范围/选择
其它站访问期间有效的模块	当访问请求没有指定网络号时，选择进行访问时使用的模块。	1 至 8
网络类型	选择安装模块。	以太网
起始 I/O 地址	设置模块的起始地址。	0000 至 0FE0 _H
网络号	设置模块的网络号。	1 至 239
组号	设置模块的组号。	1 至 32
站号	选择模块的站号。	1 至 64
模式	选择模块的操作模式。	<ul style="list-style-type: none"> • 在线 • 离线 • 自抑制测试 • H/W 测试
操作设置	设置模块的公共项。	—
初始化设置	设置模块的公共定时器值。	
开放设置	开放处理的设置。	
路由中继参数	使用路由中继功能的设置。	
站号 <-> IP 关联信息	使用 MELSECNET/H、MELSECNET/10 中继通讯功能的设置。	
FTP 参数	使用文件传送 (FTP) 功能的设置。	
电子邮件设置	传送/接收电子邮件的设置。	
中断设置	执行中断程序的设置。	
冗余设置	冗余系统中使用以太网模块的设置。	
组设置	对冗余系统中的外部设备和以太网模块使用冗余通讯路径时，如果通讯路径中发生出错，则执行禁止系统切换的设置。 (参阅 QnPRH CPU 用户手册 (冗余系统))	
路由参数	使用 MELSECNET/H、MELSECNET/10 中继通讯功能的设置。	

(1) 其它站访问期间的有效模块

- (a) 当本地站发出数据通讯请求，而它又无法指定访问目标 PLC 站的网络号时，该项指定直接请求哪一个网络模块。应为下列之一：
- 用于 MELSECNET/H、MELSECNET/10 的网络模块
 - 以太网模块

(2) 网络类型

- (a) 选择“以太网”，每一设置项就会如上画面中所示那样显示。必须设置下列项目。
- 起始 I/O 地址
 - 网络号
 - 组号
 - 站号
 - 模式
 - 操作设置
 - 初始化设置(使用默认值时不需要设置)

(3) 起始 I/O 地址

- (a) 设置 16 点单元中以太网模块的起始 I/O 地址(十六进制)。
(b) 设置的例子如下所示。

(范例)

- | | |
|-------------|---|
| 1) CPU 型号 | : Q25 (H) CPU |
| 2) 目前安装的模块 | : 串行通信模块 QJ71C24 |
| 3) I/O 使用状态 | : X/Y000 至 X/Y01F(QJ71C24 使用)
X/Y020 至 X/Y03F(以太网模块使用) |

以上环境中，起始 I/O 地址为“0020”。

(4) 网络号

- (a) 设置 MELSECNET/H、MELSECNET/10 的目标以太网模块(设置范围从 1 到 239)的网络号。
(b) 不要把网络号设置成已指定给现有系统和其它以太网及 MELSECNET/H、MELSECNET/10 网络系统的网络号。

(5) 组号

- (a) 设置 MELSECNET/H、MELSECNET/10 的目标以太网模块(设置范围从 1 到 32)的组号。
(b) 通过指定组号，可以使用相同组号与多 QCPU 站通讯数据。

(6) 站号

- (a) 设置 MELSECNET/H、MELSECNET/10 的目标以太网模块的站号(设置范围从 1 到 64)。
- (b) 不要把站号设置为已指定给现有系统和其它以太网模块的值。

(7) 模式(地址: CAH)

- (a) 选择以太网模块的操作模式。

设置名称	设置说明
在线	以正常操作模式与外部设备通讯。
离线	将本站从网络上断开。
自抑制测试	为了自诊断, 执行自抑制测试。(*1)
H/W 测试	执行 RAM 和 ROM 的测试。(*1)

*1 请参阅第 4.8 节, “自诊断测试”。

- (b) 当更改操作模式时, 将 PLC CPU 复位。

(8) 通过路由参数进行操作设置

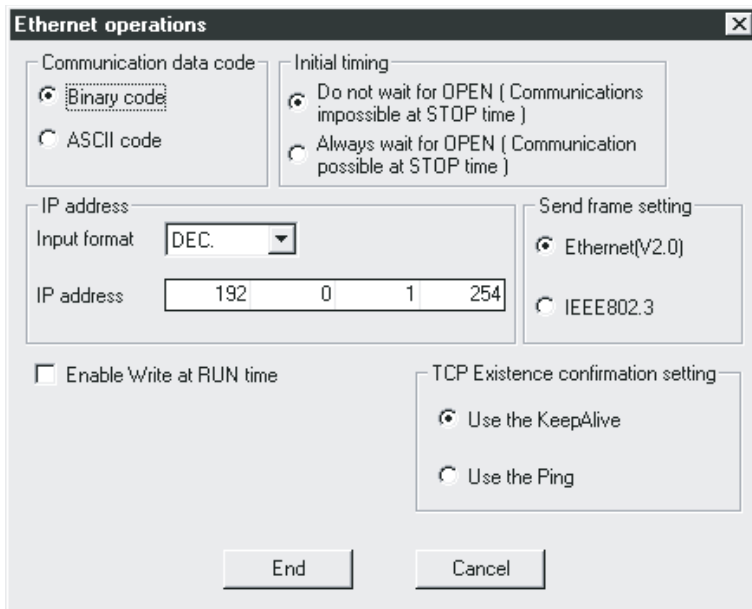
按照第 4.5 节中说明, 通过路由设置, 从操作设置中设置参数。

要点
(1) 必须指定“MNET/10H 以太网卡号的设置”和“操作设置”参数。如果更改设置, 必须将 QCPU 复位(当在多 CPU 系统中使用时为 1 号 PLC)。
(2) 当在多 CPU 系统中使用模块时, 将网络参数写入以太网模块的控制 PLC 中。

4.7 操作设置

本节说明如何设置操作参数。

通过选择[设置 MNET/10H 以太网卡号]-[操作设置]，启动“以太网操作”画面。



项目名称	项目设置说明	设置范围/选择
通讯数据代码	选择通讯数据代码。	<ul style="list-style-type: none"> 二进制代码 ASCII 代码
初始时间设置	执行开放的设置。	<ul style="list-style-type: none"> 不等待打开。 始终等待打开。
IP 地址	输入格式	<ul style="list-style-type: none"> 十进制 十六进制
	IP 地址	—
发送帧设置	选择要发送帧的格式。	以太网 (V2.0) <ul style="list-style-type: none"> IEEE 802.3
允许在运行时间写入 (用于 MC 协议)	设置允许在运行期间写入 PLC CPU。	<ul style="list-style-type: none"> 检查标记 (允许) 不检查标记 (禁止)
TCP 存在确认设置	选择存在确认方法进行 TCP 通讯。	<ul style="list-style-type: none"> 使用 KeepAlive 使用 Ping

(1) 通讯数据代码 (地址: CBH ... b1)

(a) 与外部设备通讯时，选择通讯数据的格式。

设置名称	设置说明
二进制代码	使用二进制数据通讯。
ASCII 代码	使用 ASCII 数据通讯。

(b) 关于数据通讯代码的更多详细说明，请参阅第 3.2 节，“通讯用的数据代码”。

(2) 初始时间设置(地址: CBH ... b8)

(a) 用“开放设置”参数选择 TCP-被动开放或 UDP 开放的时序(*1)。

*1 关于开放设置的更多详细说明, 请参阅第 5.5 节“开放设置”。

设置名称	设置说明
不等待打开 (在停止时间不能通讯)	<ul style="list-style-type: none"> 使用顺控程序执行开放/关闭处理。 PLC CPU 处于停止状态时, 不能执行通讯。
始终等待打开 (在停止时间可以通讯)	<ul style="list-style-type: none"> 按照参数设置, 被动开放和 UDP 开放始终等待打开(不需要开放/关闭处理的顺控程序)(*2)。 PLC CPU 处于停止状态时, 可以执行通讯。

*2 如果本站 PLC CPU 的顺控程序执行关闭处理, 连接断开之后, 站不处于开放请求等待状态。

(b) 在下列情况下, 需要专用指令来执行开放/关闭处理。

- 当在初始时间设置中选择“不等待打开”时。
- “开放设置”不执行连接时。
- 在连接的“开放设置”中选择“TCP-主动”时。

关于开放/关闭处理的详细说明, 请参阅第 5.6 节“开放/关闭处理”。

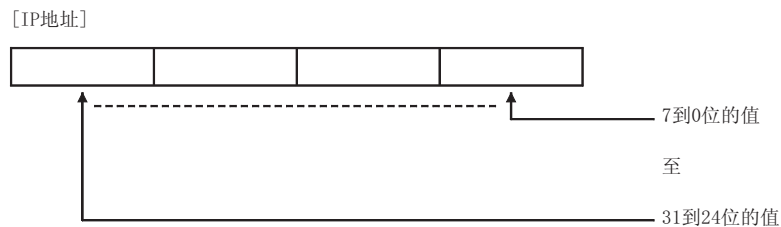
(3) IP 地址设置-输入格式

(a) 选择 IP 地址输入格式(十进制或十六进制)

(4) IP 地址设置-IP 地址(地址: 0H 至 1H)

(a) 按照指定的输入格式(十进制或十六进制)设置本站的 IP 地址。

应该设置 IP 地址, 使本站以太网模块和通讯的外部设备有相同的级和副网络 ID(2 个字)。



(b) 为了与其它以太网(不同副网络 ID)上的外部设备通讯, 有必要使用路由中继功能。

关于更多详细说明, 请参阅第 5.3 节, “路由中继参数”。

(c) 咨询网络管理员(规划网络和管理 IP 地址的人员)后, 设置 IP 地址。

(5) 发送帧设置

(a) 选择通过以太网模块发送的数据链接层以太网标题的帧。

设置项目	设置说明
以太网 (V2.0)	使用以太网帧传送。
IEEE802.3	使用 IEEE802.3 帧传送。

(b) 接收外部设备的数据时，不管是否使用了以太网帧或 IEEE802.3 帧，都应该接收。

备注

- (1) 一般推荐使用以太网帧传送。
- (2) 当与外部设备通讯失败时，检查是否可能使用 PING 命令通讯。

(6) 允许在运行时间写入 (地址: CBH ... b6)

(a) PLC CPU 运行时，在通过 MC 协议进行通讯时，选择允许/禁止外部设备写入数据。

设置名称	设置说明
检查标记 (允许在运行时间写入)	允许外部设备在 PLC CPU 运行时作写操作。
无检查标记 (禁止在运行时间写入)	禁止外部设备在 PLC CPU 运行时作写操作。

(7) TCP 存在确认设置

选择存在确认方法进行 TCP 通讯。*1

关于存在确认功能，请参阅 5.2.2 节。

设置名称	设置说明
使用 KeepAlive	用 KeepAlive 检查连接状态
使用 Ping	用 Ping 检查连接状态

- *1 不要让与本设置不兼容的 GX Developer 和兼容的 GX Developer 一起使用。
 (否则可能会把设置更改为“使用 Ping”)
 当以太网模块不支持 KeepAlive 检查功能时则忽略此设置。(Ping 用于存在确认)
 关于以太网模块和 GX Developer 的适用版本，请参阅 2.7 节。

要点

- (1) 必须始终设置“设置 MNET/10H 以太网卡号”和“操作设置”参数。
如果更改了设置，必须将 PLC CPU 复位。
- (2) 如果由于发生出错而必须重新初始化以太网模块，要使用顺控程序。(参阅 5.3.2 节)
- (3) 当对传统机型使用程序时，必须删除或禁止使用 I/O 信号进行初始化处理的程序。

4.8 自诊断测试

本节说明检查硬件和以太网模块的传送和接收功能的自诊断测试。

在 GX Developer “设置 MNET/10H 以太网卡号的网络参数” 画面上执行自诊断测试。

4.8.1 自抑制测试

以下说明了用于检查硬件(包括以太网模块的传送和接收电路)的自抑制测试。

自抑制测试将定址在以太网模块本地站的测试信息传送到线路中，并检查是否可以通过网络接收同一信息。

以下说明了执行自抑制测试的步骤。大约要花费 5 秒钟才能完成测试。

可以从以太网模块前表面的 LED 显示来确定测试结果。

步骤	操作说明	LED 的状态		
		[RUN]	[OPEN]	[ERR.]
1	将以太网模块连接到线路中。(* 1) (参阅第 4.4 节)	-	-	-
2	停止 QCPU。	-	-	-
3	选择[设置 MNET/10H 以太网卡号]-[模式]，以选择[自抑制测试]，并将参数值保存在参数值中。(参阅第 4.6 节)	-	-	-
4	复位 QCPU。(测试开始)	●	●	○
5	5 秒钟后检查每个 LED 的状态。	正常	○	○
		出错	○	●
6	选择[设置 MNET/10H 以太网卡号]-[模式]，将模式更改为“在线”或其它测试模式，并将参数值保存在 PLC CPU 中。(参阅第 4.6 节)	-	-	-
7	复位 QCPU。	-	-	-

●: 亮 ○: 熄灭

*1 对于 QJ71E71-100 来说，当没有连接线路时，不执行自抑制测试并正常结束。

以下是发生出错的可能原因：

- 以太网模块硬件出错
- 以太网线路出错
- 外部电源 12VDC 出错(仅 10BASE5)

出错代码存储在以太网模块缓冲存储器中的出错日志区(地址：E5_H)；因此可以从 GX Developer 检查出错内容。(参阅第 11.2 和 11.3 节)

要点
<p>当外部设备在线时，即使实施自抑制测试，也没有硬件干扰。如果线路上的信息包过于拥挤，可能发生信息包冲突，测试不会以出错结束完成或在预期的 5 秒时间帧之内完成。</p> <p>在该例中，终止其它设备之间的数据通讯后，再次执行测试。</p>

4.8.2 硬件测试(H/W 测试)

本节说明以太网模块的 RAM 和 ROM 测试。硬件测试的步骤如下所示。
根据以太网模块前表面上的 LED 显示来判断测试结果。

步骤	操作说明	LED 的状态		
		[RUN]	[OPEN]	[ERR.]
1	停止 QCPU。	-	-	-
2	选择[设置 MNET/10H 以太网卡号]-[模式]，选择[H/W 测试]，并将参数值保存在 PLC CPU 中。(参阅第 4.6 节)	-	-	-
3	复位 QCPU。(测试开始)	●	●	○
4	5 秒后检查每个 LED 的状态。	正常	●	○
		出错	●	○
5	选择[设置 MNET/10H 以太网卡号]-[模式]，将模式更改为“在线”或其它测试模式，并将参数值保存在 PLC CPU 中。(参阅第 4.6 节)	-	-	-
6	复位 QCPU。	-	-	-

●: 亮 ○: 灭

下面是发生出错的可能原因:

- 以太网模块 RAM/ROM 出错

出错代码存储在以太网模块缓冲存储器的出错日志区(地址: E5H)中; 因此可以用 GX Developer 检查出错内容。(参阅第 11.2 节和第 11.3 节)

要点

如果硬件测试结果显示出错，则再次进行测试。

如果再次测试的结果也显示出错，则说明以太网模块的硬件可能出了故障。

关于出错的详细情况请与当地的三菱电机 FA 中心、分公司或者代理商联系。

4.9 维护和检查

本节说明以太网模块的维护和检查以及安装和拆卸。

4.9.1 维护和检查

只需检查以太网模块的终端连接器和电缆的连接是否松动之外，无须检查其它任何项目。为了在最佳运行条件下使用以太网模块，应根据 PLC CPU 用户手册中所述的同一检查项目维护和检查系统。



危险

- 接通电源时不要触摸端子和连接器。
否则可能遭到电击或引起故障。
- 不要触摸模块上盖内部的连接器。
否则可能损坏模块或使模块发生故障。
- 确保在清洁或重新拧紧螺钉之前，断开外部电源的所有相位。否则可能损坏模块或使模块故障。
如果螺钉松动，可能导致模块短路、故障或跌落。
螺钉拧得过紧可能损坏螺钉和/或模块，并导致模块短路、故障或跌落。



注意

- 一定要小心，不要让任何异物如接线碎屑等进入模块内部。
它们可能导致火灾，有可能损坏模块并使模块发生故障。
- 不要拆开或修改模块。
否则可能损坏模块，引起故障、人身伤害或火灾。

4.9.2 安装和拆卸模块

安装和拆卸以太网模块之前，一定要仔细阅读第 4.1 节“处理注意事项”，确保安全并按照指令正确处理模块。

以下说明安装/拆卸以太网模块时的步骤。

〈更换以太网模块时的操作步骤〉

(第 1 步) 断开安装了以太网模块的站的电源。

(第 2 步) 拆掉网络电缆和以太网模块。

(第 3 步) 按照第 4.2 节“开始操作之前的设置和步骤”安装和起动新的以太网模块。

(第 4 步) 将外部设备复位(*1)。

〈更换 QCPU 时的操作步骤〉

(第 1 步) 从 PLC CPU 读取以太网模块的参数，并用 GX Developer 保存。(*2)

(第 2 步) 更换 QCPU。(请参阅相应的 QCPU 用户手册。)

(第 3 步) 将使用 GX Developer 保存的以太网模块参数输入到新的 QCPU 中。

(第 4 步) 将外部设备复位。

*1 当更换以太网模块时，也要将外部设备复位。(如果外部设备保留以太网地址，因为更换了模块，以太网地址已改变，所以无法继续通讯。)

同样的，当更换了外部设备(个人计算机等)时，应重新起动以太网模块。

*2 建议不仅要在更换 CPU 时，在创建或修改以太网模块参数时也要记录和保存参数。

5 通讯步骤

本章介绍了使用以太网模块的通讯步骤，以及为了与外部设备通讯，需要在开始数据通讯和开放处理之前对以太网进行的初始化处理。

通过设置 GX Developer 支持的以太网模块参数，可以大大简化设置执行通讯所需参数的顺控程序。

下面说明使用 GX Developer 的通讯步骤。

备注

当把顺控程序用于常规模块时，请参阅附录，“使用专门为常规模块设计的程序”。

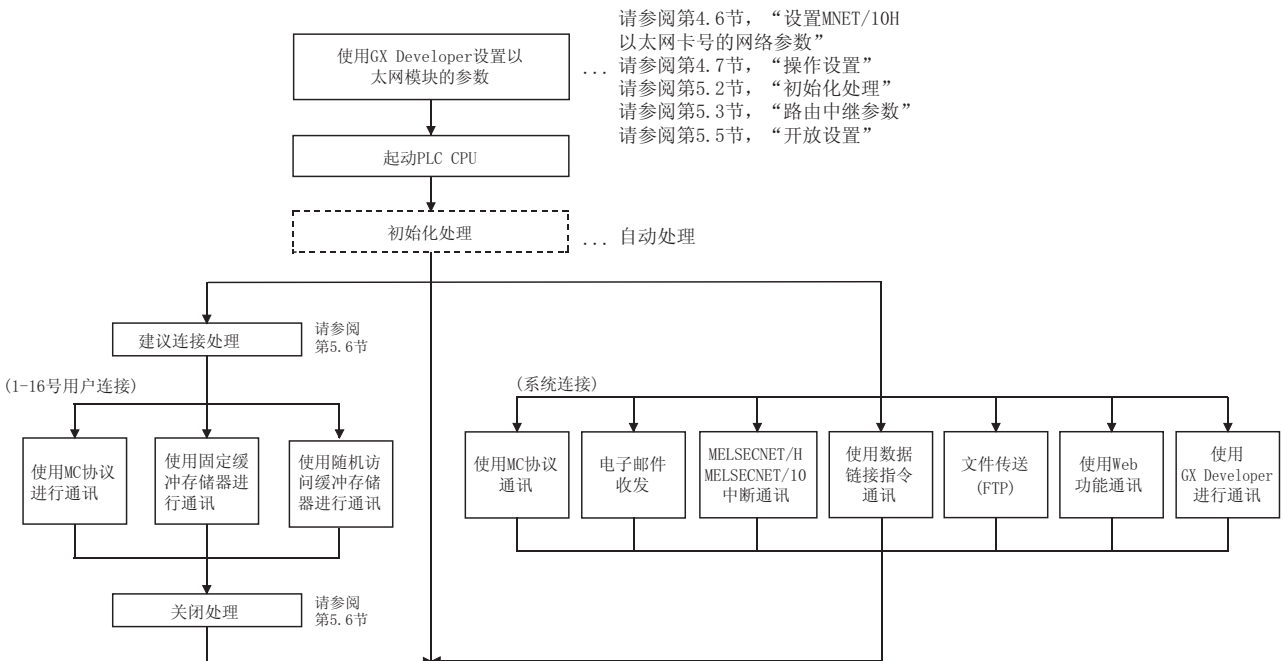
5.1 通讯步骤概要

本书说明了通过以太网模块与外部设备进行数据通讯的步骤。

需要起动数据通讯时，通过初始化或开放处理与外部设备建立连接。

需要结束数据通讯时，执行关闭和结束处理来断开连接，结果是终止所有的通讯处理。

下图说明通讯步骤：

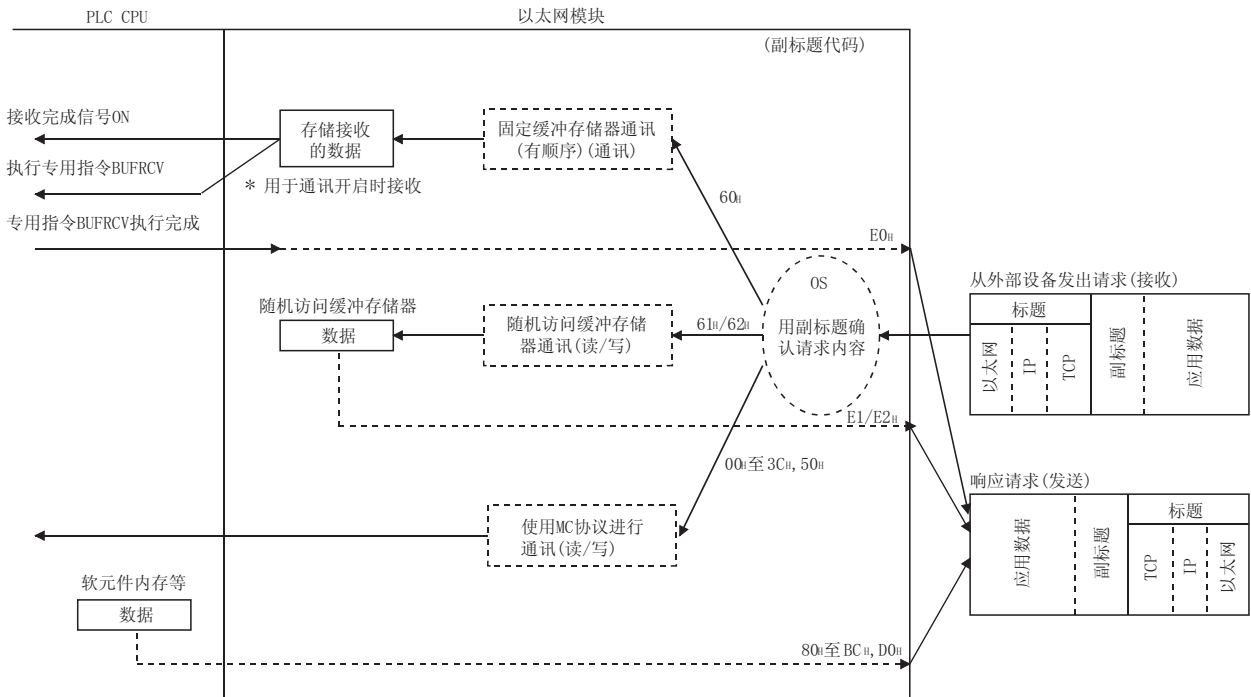


备注

用户可以与开放的外部设备进行以下各种通讯。

- 使用 MC 协议进行通讯
- 在固定缓冲存储器通讯中发送/接收(有顺序)
- 使用随机访问缓冲存储器进行通讯

当接收到从外部设备发出的通讯请求时



5.2 初始化处理

本节说明了以太网模块的初始化处理。

5.2.1 初始化处理

通过设置以太网模块进行数据通讯需要的最少参数，初始化处理使得与外部设备进行数据通讯成为可能。

使用 GX Developer 设置下列参数，将它们保存在 PLC CPU 中，并将 PLC CPU 复位；然后执行以太网模块的初始化处理。（不需要初始化处理的顺控程序。）

(1) 初始化处理需要的参数(使用 GX Developer 设置)

- “设置以太网卡号的网络参数” : (请参阅第 4.6 节)
- “操作设置” : (请参阅第 4.7 节)
- “初始设置” : (请参阅第 5.2.2 节)

(2) 初始化处理结果的确认

初始化处理	以太网模块		
	INIT. LED	I/O 信号	
		初始化正常完成信号 X19	初始化异常完成信号 X1A
正常完成时	● :亮	亮	灭
异常完成时	○ :灭	灭	亮

如果初始化处理没有正常完成，纠正以上参数设置值，并把它们写入 PLC CPU。然后将 PLC CPU 复位。

5.2.2 初始化设置

本节说明了使用 GX Developer 进行初始化设置的情况。

选择[设置 MNET1/10H 以太网卡号]-[初始设置]起动[以太网初始化设置]画面。

Network parameter Ethernet initial setting. Module No.1

Timer setting
Module will operate on default values if setting left blank

	Setting value	Default value	In module
TCP ULP timer		60	×500ms
TCP zero window timer		20	×500ms
TCP resend timer		20	×500ms
TCP end timer		40	×500ms
IP assembly timer		10	×500ms
Response monitoring timer		60	×500ms
Destination existence conformation starting interval		1200	×500ms
Destination existence conformation interval timer		20	×500ms
Destination existence conformation resend timer		3	Times

DNS setting
Input format: DEC.

IP address of DNS server 1				
IP address of DNS server 2				
IP address of DNS server 3				
IP address of DNS server 4				

End Cancel

项目名称	设置的说明	设置范围/选项
定时器设置	TCP ULP 定时器	设置 TCP 数据传送时信息包存在的时间。
	TCP 零窗口定时器	设置检查接收允许状态的间隔。
	TCP 再次发送定时器	设置在 TCP 数据传送时再次发送的时间。
	TCP 结束定时器	设置在 TCP 关闭处理时的确认等待时间。
	IP 组合定时器	设置部分数据信息包的等待时间。
	响应监视定时器	设置响应等待时间。
	目标存在确认起时间隔	设置与外部设备进行的通讯终止后，开始确认外部设备存在的时间。
	目标存在确认间隔定时器	设置再次确认存在之间的时间间隔。
DNS 设置 (*1)	目标存在确认再次发送定时器	设置没有接收到存在情况的响应时再次确认存在的次数。
	输入格式	选择 DNS 服务器的 IP 地址输入格式
	DNS 服务器 1 的 IP 地址	设置 DNS 服务器 1 的 IP 地址
	DNS 服务器 2 的 IP 地址	设置 DNS 服务器 2 的 IP 地址
	DNS 服务器 3 的 IP 地址	设置 DNS 服务器 3 的 IP 地址
DNS 服务器 4 的 IP 地址	设置 DNS 服务器 4 的 IP 地址	

*1: 详细说明请参阅用户手册(应用篇)的第2章。

- (1) 定时器设置-TCP ULP 定时器(地址: BH)
 - (a) 该项目设置 TCP 数据发送期间数据包存在的时间。
在 TCP 开放或数据发送期间, 由通过该定时器的一个参数来设置, 该设置也用作 ARP 功能的存在时间。
 - (b) 在 2 到 32767 的范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (c) 定时器设置 = 设置值 \times 500ms

- (2) 定时器设置-TCP 零窗口定时器(地址: CH)
 - (a) 这个窗口指示接收端方面的接收缓冲存储器。
 - (b) 当接收端的接收缓冲存储器(窗口尺寸 = 0)没有更多空间时, 数据通讯时必须等待, 直到有足够空间才能进行数据通讯。
当发生这种情况时, 达到 TCP 零窗口定时器值后, 发送端将发送窗口确认信息包发送到接收端, 并确认允许的接收状态。
 - (c) 在 2 到 32767 的范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (d) 定时器设置 = 设置值 \times 500ms

- (3) 定时器设置-TCP 再次发送定时器(地址: DH)
 - (a) 这个项目设置当 TCP 连接开放和发送数据时未返回 ACK 再次发送信息包的时间。该定时器也可以记录发出 ARP 后无响应时再次发送一个 ARP 请求的时间。
(在 TCP 再次发送 1/2 的定时器值时, 再次发送一个 ARP。)
它也用作记录数据链接指令达到监视时间的最小设置时间。
 - (b) 在 2 到 32767 的范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (c) 定时器设置 = 设置值 \times 500ms

- (4) 定时器设置-TCP 结束定时器(地址: EH)
 - (a) 当从本站断开 TCP 连接时, 该定时器在设置外部设备发送 FIN 请求且外部设备返回 ACK 后, 本站等待 FIN 请求的监视时间。
 - (b) 在 TCP 结束定时器设置指定时间之前, 如果无法从外部设备接收到 FIN 请求, 应发给外部设备一个 RST, 强制关闭。
 - (c) 在 2 到 32767 的范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (d) 定时器设置 = 设置值 \times 500ms

- (5) 定时器设置-IP 组合定时器(地址: Fh)
- (a) 由于发送和接收站的缓冲存储器的限制, 只能按 IP 级划分通讯数据。
 - (b) 在 1 到 32766 范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (c) 定时器设置 = 设置值 × 500ms
- (6) 定时器设置-响应监视定时器(地址: 10h)
- (a) 该定时器设置设置下列时间:
 - 1) 发送命令后, 等待响应的的时间。
 - 2) 当划分信息时, 接收第一条信息后, 等待最后一条信息的时间。
 - (b) 在 2 到 32767 的范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (c) 定时器设置 = 设置值 × 500ms
- (7) 定时器设置-目标存在确认的开始间隔(地址: 11h)
- (a) 该定时器设置, 当目标存在确认设置为确认停止的开放连接时, 开始确认外部目标存在前的时间间隔。
 - (b) 在 1 到 32767 范围中指定设置值。
(当使用默认值时, 不需要设置。)
 - (c) 定时器设置 = 设置值 × 500ms
- (8) 定时器设置-目标存在确认间隔定时器(地址: 12h)
- (a) 该定时器设置, 当存在确认设置为不响应确认的开放连接时, 再次确认外部设备存在前的时间间隔。
 - (b) 在 1 到 32767 范围中指定设置值。
(当使用默认值时, 无需设置。)
 - (c) 定时器设置 = 设置值 × 500ms
- (9) 定时器设置-目标存在确认再次发送定时器(地址: 13h)
- (a) 该定时器设置为, 当存在确认设置为确认的开放时, 未从外部设备得到响应时再次确认存在的次数。
 - (b) 在 1 到 32767 范围中指定设置值。
(当使用默认值时, 无需设置。)

(10) DNS 设置-输入格式(*1)

(11) DNS 设置-DNS 服务器 n 的 IP 地址(*1)

*1 当使用电子邮件发送/接收时, 设置 DNS 设置。

参考第 2 章, 用户手册(应用篇)的“电子邮件功能”。

备注

(1) 按符合以下关系的方式指定以太网模块上每个定时器的设置值。

$$\begin{aligned} & \bullet \left[\text{响应监视定时器值} \right] \geq \left[\text{TCP ULP 定时器值} \right] \geq \left[\text{TCP结束定时器} \right] \geq \left[\text{TCP再次发送定时器} \right] > \left[\text{IP汇编定时器值} \right] \\ & \bullet \left[\text{TCP再次发送定时器值} \right] = \left[\text{TCP零窗口定时器值} \right] \end{aligned}$$

而且, 使用三菱公司产品来连接电缆时, 你应该确保节点两头的设置相同。

(2) 在符合以下关系的条件下指定以太网模块上每个定时器的设置值。

如果没有设置定时器值以满足以下关系, 可能频繁发生发送超时之类的通讯错误。

$$\begin{aligned} & \bullet \left[\text{外部设备端的TCP重试定时器值} \right] > \left[\text{以太网模块端的TCP重试定时器值} \right] \\ & \bullet \left[\text{监视外部设备应用系统的定时器值} \right] > \left\{ \left[\text{以太网模块端的TCP ULP定时器值} \right] \times n \right\} \end{aligned}$$

*1 n是TCP段传送数量, 可以通过以下计算获得。

$$n = \left\lceil \frac{\text{以太网模块传送的信息大小}}{\text{最大段大小}} \right\rceil \text{ (小数点后小数四舍五入)}$$

范例 1: 通过相同电缆通讯时的 TCP 段传送数量

以下是通过相同电缆(无需通过路由器)的最大段大小为 1460 字节和 TCP 段传送数:

n = 1, 如果以太网模块传送的信息大小为 1460 字节或更小。

n = 2, 如果以太网模块传送的信息大小为 1460 字节以上。

范例 2: 通过分离电缆通讯时的 TCP 段传送数

以下是分离电缆上(通过拨号路由器)的最大段至少为 1460 字节和 TCP 段传送数:

n = 1, 如果以太网模块传送的信息大小为 536 字节或更小。

n = 2, 如果以太网模块传送的信息大小为 536 字节到 1072 字节之间。

n = 3, 如果以太网模块传送的信息大小为 1072 字节到 1608 字节之间。

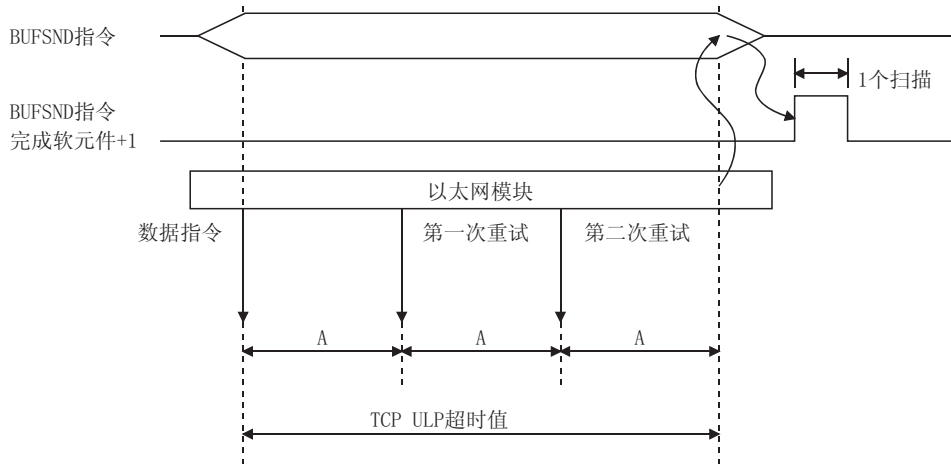
(3) 当由于噪音和其它因素而产生通讯错误时，需要更改设置以增加重试数。

重试数可以通过使用以下方程式得到。：

(假设是缺省值， $2 = (60/20) - 1$)

$$\bullet \left[\text{重试数} \right] = \left\{ \left[\frac{\text{TCP ULP定时器值}}{\text{TCP重新发送定时器值}} \right] - 1 \right\}$$

举例： 假设按此方法设置值，得到的重试数为2，那么数据传送出现故障时在表中显示的时间会发生数据通讯错误。(使用固定缓冲器通讯时。)



A: TCP重新发送定时器值
(如果在发送数据后没有返回“ACK”，需要重新传送数据)

(4) 进行以下设置以废除(3)中说明的重试。(例如：设置重试数为0)

$$\bullet \left[\text{TCP ULP定时器值} \right] = \left[\text{TCP结束定时器值} \right] = \left[\text{TCP重新发送定时器值} \right]$$

(每个定时器值都应该相同)

(5) 目标存在确认是以太网模块检查是否远程设备正常运作的功能，该功能通过发送存在确认信息，然后等待察看是否接收响应信息。如果连接到远程设备为开放状态但在某一时间段没有执行与远程设备的通讯，则使用此功能。

(a) 存在确认功能有两种确认方式：PING 和 KeepAlive。

根据本节(7)到(9)中说明的设置值和开放设置(参阅 5.5(6)节)的存在确认设置，以太网模块逐个进行存在确认。

在确认功能(Pink 或 KeepAlive)可以在操作设置或重新初始化时选择。

关于操作设置，请参阅 4.7 节。

关于重新初始化处理的说明，请参阅 5.2.3 节。

1) 用 KeepAlive 确认

此方法用于通过 TCP/IP 协议开放连接。

以太网模块通过向某一时间段没有进行通讯的远程设备发送存在确认 ACK 信息，然后等待并查看是否接收响应。(*¹)

*1 如果远程设备不支持 TCP KeepAlive 功能(响应 KeepAlive ACK 信息)，连接可能被中断。

2) 用 Ping 确认

此方法用于通过 TCP/IP 或 UDP/IP 协议打开的连接。

以太网模块通过向某一时间段没有进行通讯的远程设备发送 Ping 指令(使用 ICMP 回复请求/响应功能)，然后等待并查看是否接收响应。(*²)

*2 注意以太网模块接收 PING 响应指令时自动返回回复响应包。(即使关闭与远程设备的数据通讯，也可以发送响应到接收的 Ping 指令。)

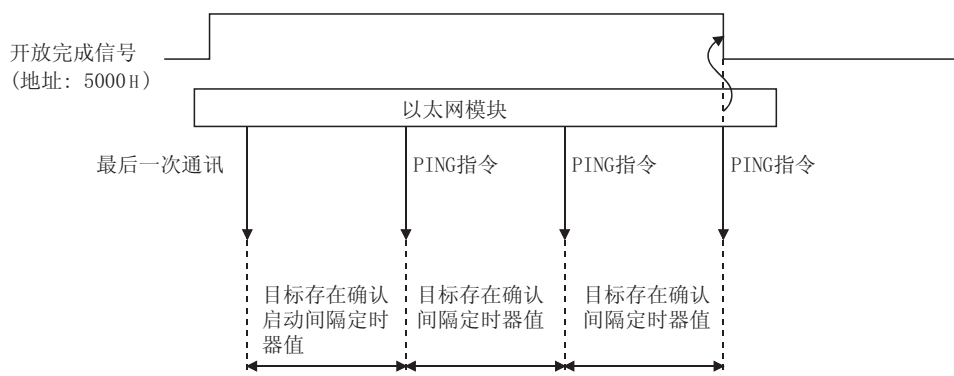
(b) 如果无法接收(检测到出错)远程设备发出的响应信息，请采取以下措施。

- 强制关闭相应连接(线路断开)。(*³)

*3 可使用用户程序重新开放。

- 以太网模块关闭开放完成信号(对应位地址：5000H)并存储出错代码(C035H)，例如在开放出错代码存储区。

举例： 假设在重试三次的情况下设置值，以太网模块在下表中的时间进行目标存在确认。(以 PING 存在确认为例)



5.2.3 重新初始化处理

重新初始化处理是让以太网模块处于启动状态，而不是真正重新启动 PLC (例如：复位 PLC CPU)。

在顺控程序中可以执行以太网模块的重新初始化处理。

以下说明了重新初始化的目的和如何计划重新初始化处理。

(1) 进行重新初始化处理的目的

(a) 更新以太网模块支持的外部设备地址信息

以太网模块支持与外部设备通讯的 IP 地址以及相应的以太网地址 (MAC 地址)。使用正常进行通讯的外部设备 IP 地址是为了避免其它设备非法访问 PLC。(*1)

为了防止外部设置上的模块和模板由于故障而替换，需要进行重新初始化处理以避免清除以太网模块支持的外部设备地址信息。

*1 每个设备只有唯一的以太网地址。没有设备具有相同的以太网地址。

(b) 更改本地站的以太网模块 IP 地址

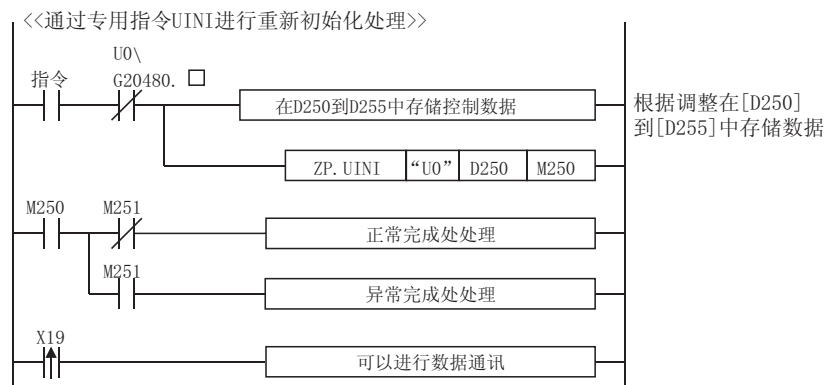
如果更换系统，可以通过更改 GX Developer 设置的操作设置 (参阅 4.7 节) 中的 IP 地址 (用于本地站的以太网模块) 来重新启动与外部设备的通讯。

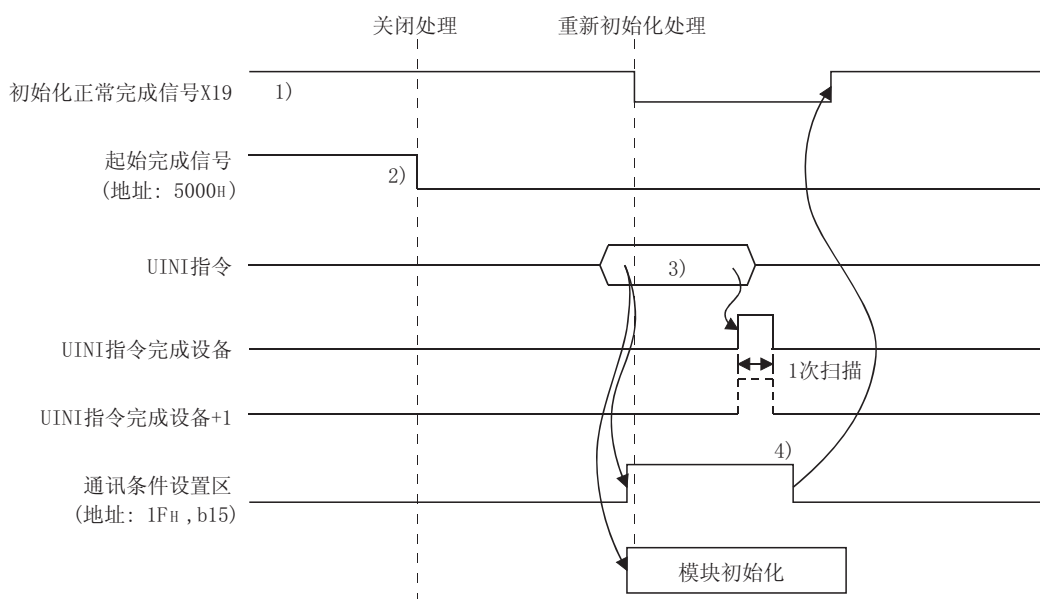
(c) 更改通讯条件设置值

通过更改 GX Developer 设置的操作设置 (参阅 4.7 节) 中的通讯条件来重新启动与外部设备的通讯。

(2) 进行重新初始化处理的编程和时序

以下显示进行重新初始化处理的编程和时序：





- 1) 确认是否已经正常完成初始化处理。
- 2) 停止当前与外部设备正在进行的所有数据通讯，并对所有连接进行关闭处理。
- 3) 执行专用指令 UINI。
(初始化正常完成信号(X19)：开)
(开放完成信号(地址：5000_H(20480))：所有关(0_H)
通过专用指令的控制数据指定参数(本地站 IP 地址，操作设置)后，初始化以太网模块。
- 4) 完成重新初始化处理时，重新初始化规格(地址：1F_H，位 15)变为“0”，初始化处理完成信号 X19 开放。
* 如果重新初始化处理完成异常，错误代码存储在以下区域。
初始化出错代码区(地址：69_H(105))

(3) 重新初始化处理时如果设置更改为“允许 TCP 最大段大小选项传送”，要注意与 MELSOFT 产品的组合

- (a) 支持 TCP 最大段传送功能的 MELSOFT 系列产品
关于设置“允许 TCP 最大段大小选项传送”，使用以下 MELSOFT 产品。
GX Developer : 版本8.07H 或以上版本
MX Component : 版本3.03D或以上版本
MX Links : 版本3.08J或以上版本
- (b) 以上(a)中没有列出 MELSOFT 系列产品
使用以上(a)中没有列出 MELSOFT 系列产品通过以太网进行通讯时，设置“禁止 TCP 最大段大小选项传送”或使用 UDP/IP 型通讯。
否则，顺控程序可能运行错误。(读/写出错)

要点

重新初始化以太网时，请注意以下几点：

(否则可能引起与外部设备的数据通讯错误。)

- (1) 在进行重新初始化处理前，一定要终止与外部设备的所有当前数据通讯并关闭所有连接。
- (2) 不要与直接写入缓冲存储器而产生的重新初始化处理相混淆，比如：使用 T0 指令，通过 UNI 指令的重新初始化处理。
而且，正在进行另一个重新初始化处理时不要请求另一个重新初始化处理。
- (3) 如果以太网模块的 IP 地址已经更改，一定要重新设置外部设备。(如果外部设备支持通讯设备的以太网地址，那么以太网模块的 IP 地址更改后可能无法继续通讯。)

备注

使用顺控程序进行重新初始化处理时可以更改操作设置。

对于以太网模块，在重新设置 QCPU(在多 CPU 系统中的 1 号 PLC)后，应该通过 GX Developer 对除了操作设置以外的其它参数、设置进行更改。

参数设置项目	适用缓冲存储器地址 (十六进制)	参数更改	参考章节
PLC 参数	-	×	-
I/O 分配设置	-	×	第 4.5 节
中断指针设置	-	×	7.3 节
网络参数设置	-	×	4.6 节
MNET/10 以太网卡号	-	×	4.6 节
操作设置	1 F _n	○	4.7, 5.2.3 节
初始化设置	定时器设置	△(*1)	5.2 节
	DNS 设置	×	用户手册(应用篇)第 2 章
开放设置	20 _n 至 5F _n	△(*1)	5.5 节
路由中继参数	4 _n , 200 _n 至 224 _n	△(*1)	5.3 节
站号<->IP 关联信息	4 _n , 229 _n 至 3A9 _n	△(*1)	用户手册(应用篇)第 3 章
FTP 参数	4 _n , 3B0 _n 至 3BB _n	△(*1)	用户手册(应用篇)第 5 章
电子邮件设置	-	×	用户手册(应用篇)第 2 章
发送邮件地址设置	-	×	
新闻设置	-	×	
中断设置	-	×	7.3 节
冗余设置	-	×	5.11.3 节
路由参数	-	×	用户手册(应用篇)第 3 章
远程口令设置	-	×	5.9.5 节

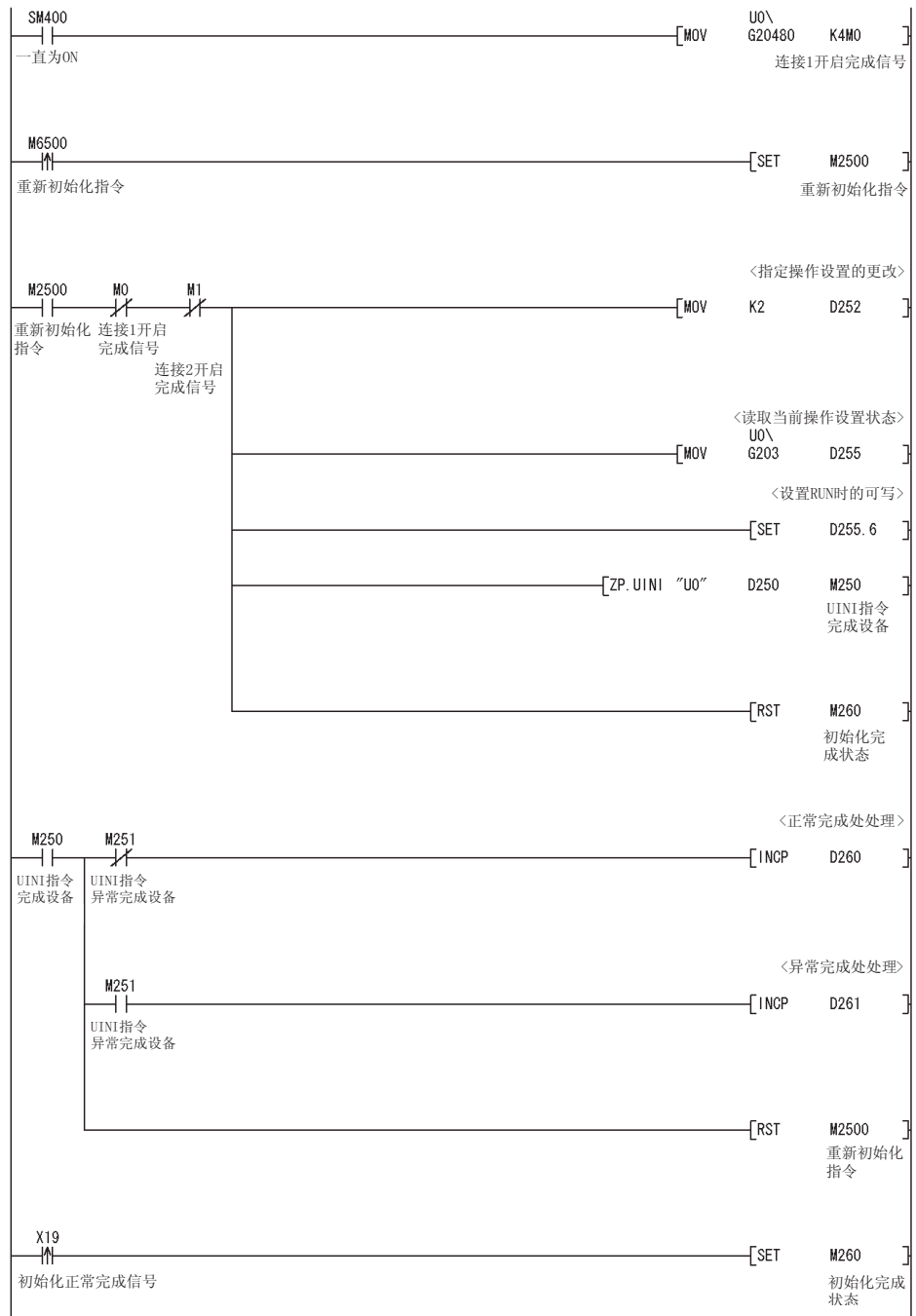
○ △：有效，×：无效

*1：PLC CPU 按照适用缓冲存储器中重新初始化处理时执行的设置运行。不要更改适用缓冲存储器中的设置。

(4) 示例程序

下图显示执行重新初始化处理的示例程序。

当以太网模块的 I/O 信号为 X/Y00 到 X/Y1F



要点

当与连接 1 和 2 通讯时，该范例程序可用来执行重新初始化处理。如果使用其它连接，则可以指定每个连接的相应号和位。

备注

直接写入到以下缓冲存储器地址可以重新初始化以太网模块的处理。
以下说明通过直接写入到缓冲存储器进行重新初始化处理的步骤：

- (a) 更改存储在 TCP 最大段传送区 (地址：1E_H) 中的值。
0000_H：允许 TCP 最大段大小选项传送
8000_H：禁止 TCP 最大段大小选项传送
- (b) 更改存储在通讯条件设置区 (地址：1F_H) 中的值。
位 0 至 14：指定操作设置中的值
位 15： “1”

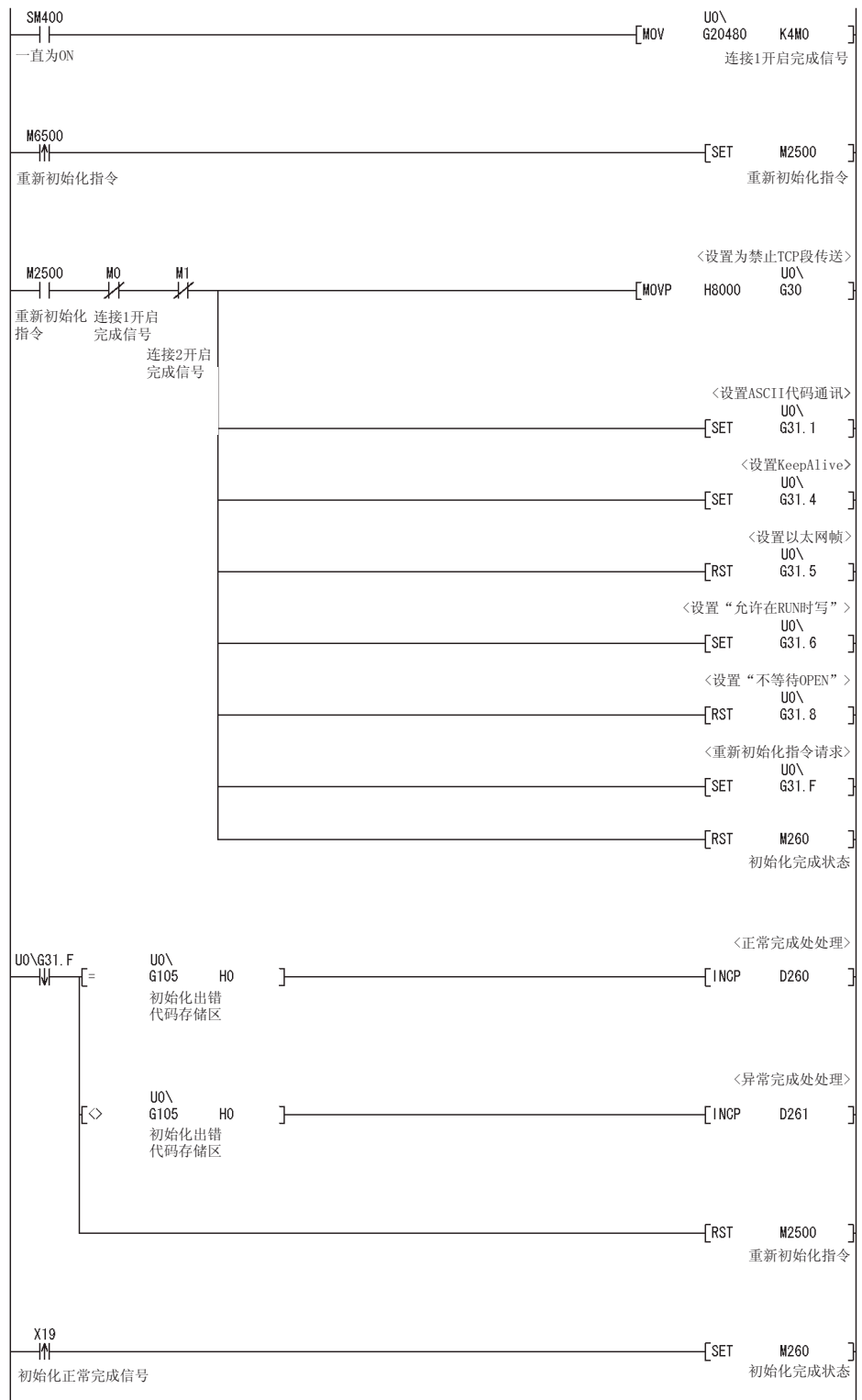
缓冲存储器 地址1F _H	b15	b8	b6	b5	b4	b1
(6)	0	5)	0	4)	3)	2)
						1)
						0

- 1) 通讯数据代码设置 (b1)
0: 以二进制代码通讯
1: 以 ASCII 代码通讯
 - 2) TCP 存在确认设置 (b4)
0: 使用 Ping
1: 使用 KeepAlive
 - 3) 发送帧设置 (b5)
0: 以太网帧
1: IEEE802.3 帧
 - 4) RUN 时 (b6) 读允许/禁止设置
0: 禁止
1: 允许
 - 5) 初始化时间设置 (b8)
0: 不等待打开 (STOP 时不可以通讯)
1: 等待 OPEN (STOP 时可以通讯)
 - 6) 重新初始化规格 (b15)
0: 重新初始化处理完成 (系统设置)
1: 重新初始化请求 (用户设置)
- (c) 一旦重新初始化处理完成，设置重新初始化规格 (地址：1F_H, 位 15) 为 “0”。
初始化处理完成信号 X19 也变为开。
* 如果重新初始化处理异常完成，出错代码会存储在以下区域中。：
起始出错代码存储区 (地址：69_H (105))

(示例程序)

下图显示了通过直接写入缓冲存储器进行重新初始化处理的示例程序。
创建一个带有触点 (用户标记等) 的程序，以此来表示顺控程序中显示的以下条件。

- 开放完成信号存储区 (地址：5000_H (20480))



要点

当与连接 1 和 2 通讯时，该范例程序可用来执行重新初始化处理。如果使用其它连接，则可以指定每个连接的相应号和位。

5.3 路由中继参数

本节说明了使用 GX Developer 设置路由中继参数的情况。

选择[设置 MNET/10H 以太网卡号]-[路由中继参数]启动[设置以太网路由中继参数]画面。

项目名称	设置的说明	设置范围/选项
路由中继功能	选择是否使用路由中继功能。	<ul style="list-style-type: none"> • 使用 • 不使用
子网掩码	设置子网掩码。	C0000000 _h 至 FFFFFFFC _h
默认路由器 IP 地址	设置使用默认路由器的 IP 地址。	00000000 _h 和 FFFFFFFF _h 除外
输入格式	选择设置项目的输入格式。	<ul style="list-style-type: none"> • 十进制 • 十六进制
子网地址	当通过非默认路由器与其它以太网上的外部设备进行通讯时，设置外部设备的子网地址或网络地址。	00000000 _h 和 FFFFFFFF _h 除外
路由器 IP 地址	当通过非默认路由器与其它以太网上的外部设备进行通讯时，设置使用的路由器的 IP 地址。	-

(1) 路由中继功能(地址: 4H ... b5, b4)

- (a) 设置是否使用路由中继功能。
当以太网模块与同一个以太网(IP地址的子网地址相同)上的目标设备通讯时, 无需路由中继功能。
- (b) 路由中继功能可以通过路由器或网关与其它以太网上的软元件通讯。
(路由中继功能不是指以太网模块可作为路由器来运行。)
- (c) 路由中继功能可以设置一个默认路由器和最多 8 个路由器。

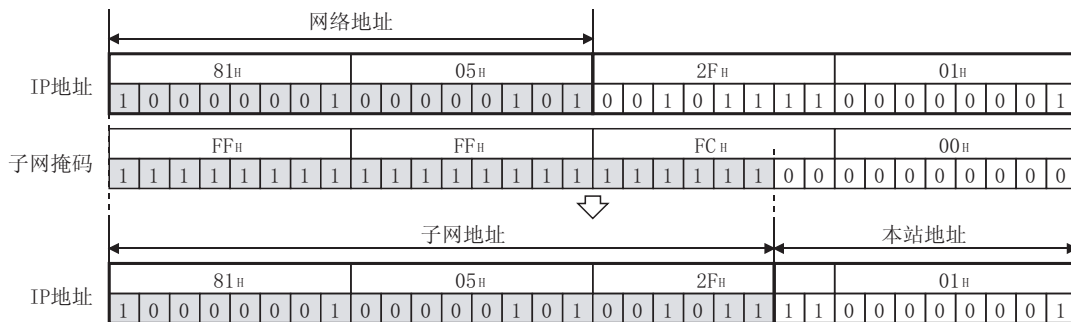
(2) 子网掩码(地址: 200H, 201H)

- (a) 设置子网掩码。*1 (设置范围: C0000000H 至 FFFFFFFCH)
关于设置请咨询网络管理员。
- (b) 不使用子网掩码时, 根据等级设置下表中的任何值。

等级	掩码值
A 级	FF000000H
B 级	FFFF0000H
C 级	FFFFFF00H

*1 以太网构建的网络包含小型网络、中型网络和大型网络系统; 小型网络系统中多个软元件连接到一个以太网, 中型网络和大型网络系统中多个小型网络通过路由器等相连。
理论上来说, 子网掩码把一个连接有许多软元件的网络分为多个子网络, 以便于管理。

(举例)等级B



要点
(1) 对于同一子网上的所有设备来说, 需要相同的公用子网掩码。
(2) 当不使用子网管理网络时, 连接的每台设备无需子网掩码。

(3) 默认路由器 IP 地址 (地址: 202H, 203H)

通过路由器信息 (参阅以下 (4)) 中非指定路由器, 在另一个以太网上以太网模块与目标设备通讯时, 设置要使用路由器 (默认路由器) 的 IP 地址。

设置的值要满足以下条件。

- 条件 1: IP 地址等级是 A、B 或 C。
- 条件 2: 默认路由器的子网地址与本站以太网模块的地址相同。
- 条件 3: 本站地址位不全是“0”或全是“1”。

要点

在开放连接或进行数据通讯时, 如果路由器信息 (参阅本节 (4)) 中不存在相应的子网地址, 可以通过缺省路由器进行通讯。

(4) 路由器信息: 子网地址 (地址: 205H, 206H 和随后的地址)

(a) 设置当通过非默认路由器, 在另一个以太网上与外部设备进行通讯时, 外部设备的子网地址 (*1) 或网络地址。 (*2)

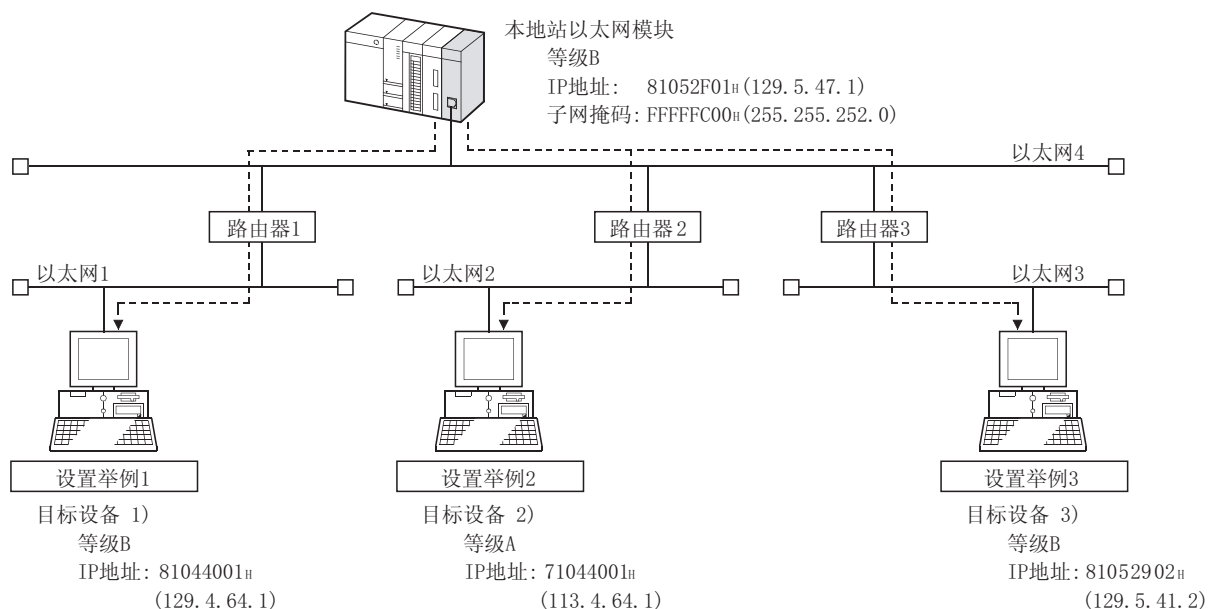
设置满足以下条件的值:

- 条件 1: IP 地址等级是 A、B 或 C
- 条件 2: 自地址位全都是“0”。

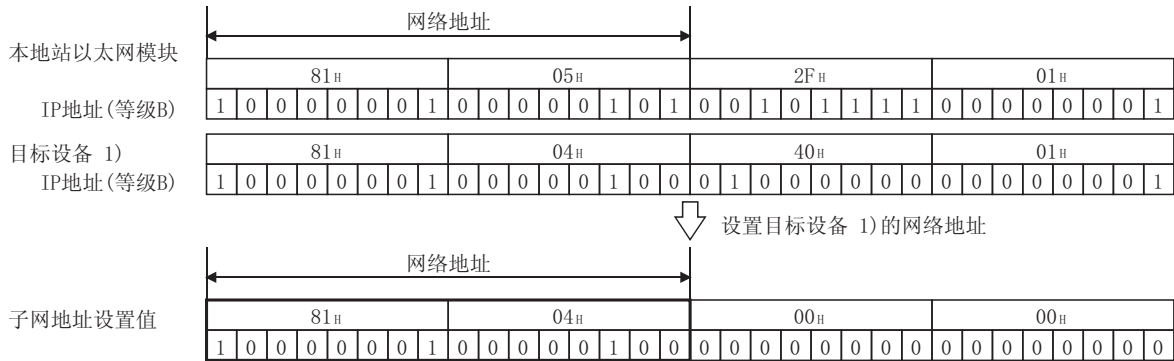
*1 如果本站的等级 (网络地址) 不同于外部设备的等级, 设置外部设备的网络地址。

*2 如果本站的等级 (网络地址) 与外部设备的等级相同, 设置外部设备的子网地址。

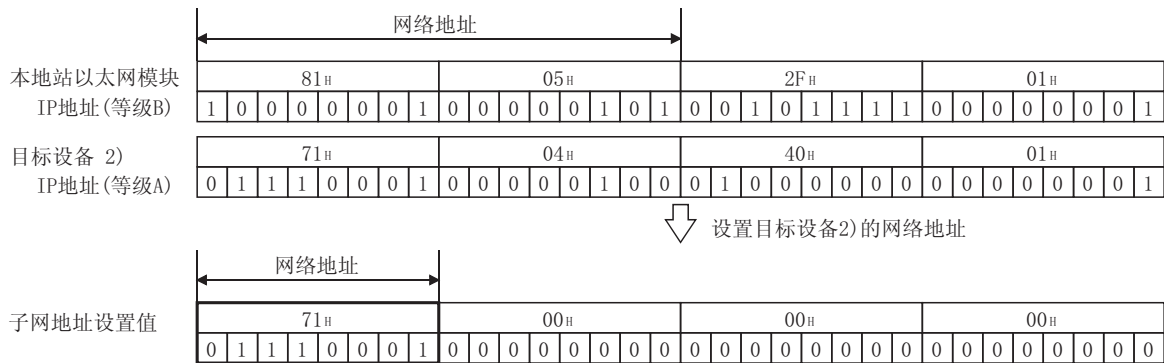
(b) 子网地址设置范例



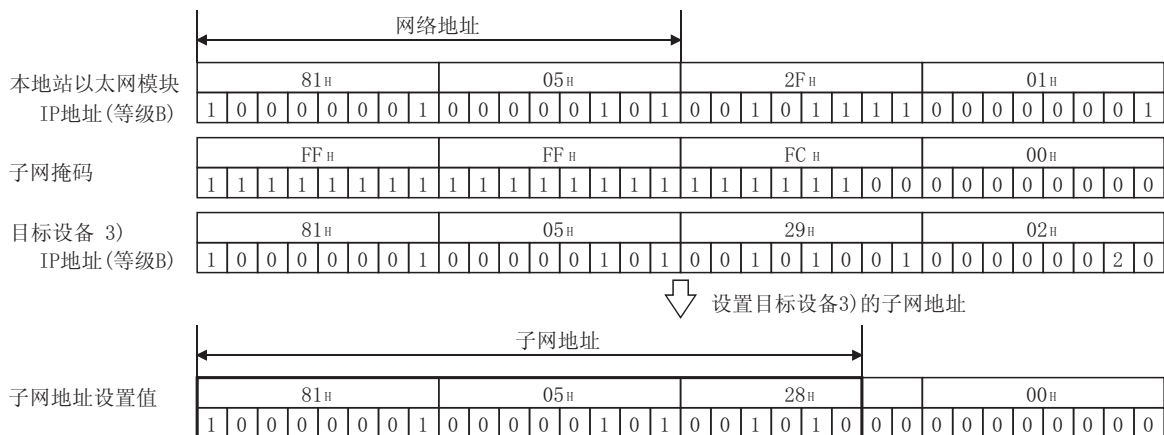
(设置例 1) 当本站以太网模块和目标设备的网络地址不同时



(设置例 2) 当本站以太网模块和目标设备的等级不同时



(设置例 3) 当本站以太网模块和目标设备的网络地址相同时



(5) 路由器信息：路由器 IP 地址(地址：207H、208H 和随后的地址)

设置当通过非默认路由器，在其它以太网上与目标设备进行通讯时，要使用得路由器的 IP 地址。

设置的值要满足以下条件。

- 条件 1: IP 地址等级是 A、B 或 C。
- 条件 2: 路由器的子网地址与本站以太网模块的地址相同。
- 条件 3: 自地址位不全是“0”或全是“1”。

要点
(1) 当通过被动开放协议(TCP/IP)，以太网模块使用路由中继与外部设备进行通讯时，不必使用路由中继功能进行通讯。
(2) 使用代理服务路由器的系统不需要路由中继功能。

5.4 确认初始化处理的完成

将下列参数保存在安装了以太网模块站的 PLC CPU 中，并重新启动 PLC CPU，以太网模块的初始化处理才算完成。

(当处理正常完成时，以太网模块前表面的 [INIT.]LED 亮。)

- “设置 MNET/10H 以太网卡数的网络参数”
- “操作设置”参数
- “初始化设置”参数

本节说明了如何检查初始化处理的完成。

要点
<p>当初始化处理正常完成时，以太网模块的状态变为允许通讯。请参阅执行通讯的每个通讯功能的参考章节。</p> <p>当初始化处理没有正常完成时，如下检查出错内容，采取纠正措施，然后再次执行初始化处理。</p> <ul style="list-style-type: none"> • 使用以太网诊断的“参数状态”检查出错代码。(请参阅第 11.2 节) • 检查对应出错代码的出错内容，然后采取纠正措施。(请参阅第 11.3.3 节)

5.4.1 使用 GX Developer 进行 PING 测试(通过以太网板)

本节说明了如何使用 GX Developer 以太网诊断的 PING 测试功能，来检查以太网模块的初始化处理完成状态。

(1) PING 测试

- (a) PING 测试用于通过 GX Developer，来检查已完成初始化处理(*1)的同一以太网线路上的以太网模块的存在情况，或检查带有指定 IP 地址的外部设备(如个人计算机)上的以太网模块的存在情况。

*1 也可以对 QnA/A 系列以太网模块执行 PING 测试。但只能对以下从软件版本 S 开始的 A 系列以太网模块执行 PING 测试：

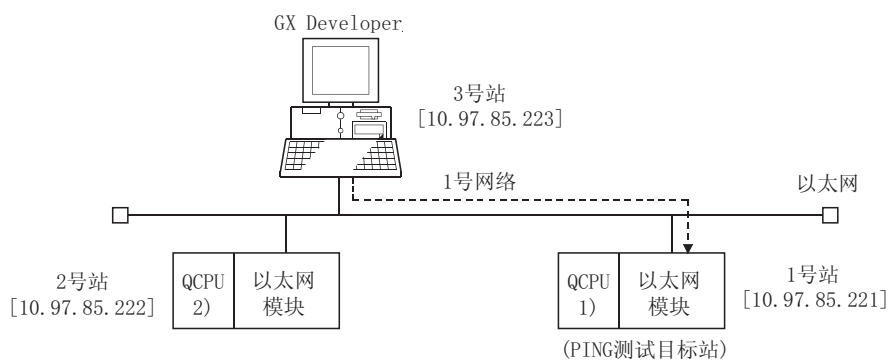
AJ71E71、AJ71E71-B2、AJ71E71-B5

- (b) 对以太网模块执行 PING 测试，可以检查以下项目：
- 线路是否正确连接到测试目标的以太网模块上
 - 是否正确设置了以太网模块的参数
 - 是否正常完成了以太网模块的初始化处理

- (c) 可以用路由中继方式对作为本站(相同的副网络 ID)的同一以太网中的以太网模块或另一以太网中的以太网模块执行 PING 测试。
- (d) 如果要进行 PING 测试的外部设备是以太网模块, 不要将以太网模块的用于 GX Developer 的 UDP 端口指定为远程口令核对的有效端口。如果把 UDP 端口指定为远程口令核对的有效端口, 则不能执行 PING 测试。

(2) 执行 PING 测试

下面的例子说明了从外围设备对相同以太网上的以太网模块执行 PING 测试时的方法和 GX Developer 的设置。



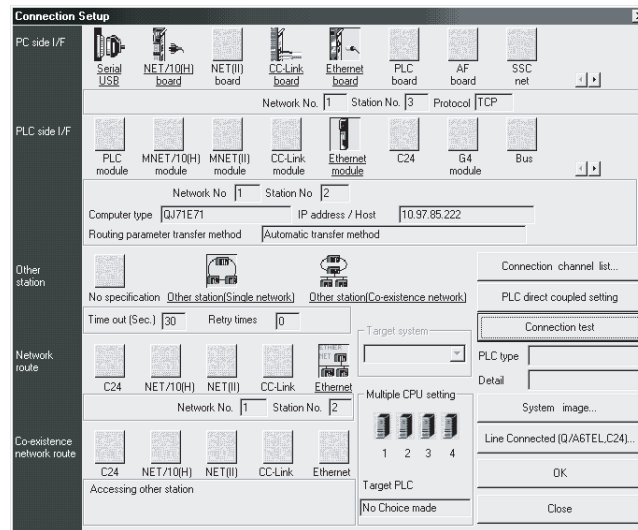
(a) 设置 PING 测试的目标站

- 1) 通过 GX Developer 设置下列 PING 测试目标站的以太网模块参数。使用除下列各项之外的设置项的默认值。

设置屏	设置项目	设置说明	
		QCPU 1)	QCPU 2)
设置 MNET/10H 以太网卡号的网络参数	网络类型	以太网	以太网
	起始 I/O 地址	0000	0000
	网络号	1	1
	组号	1	1
	站号	1	2
操作设置	IP 地址	[10.97.85.221]	[10.97.85.222]

- 2) 将参数写入应用站。
- 3) 当 PLC CPU 重新启动时, 初始化处理完成。
(当正常完成初始化处理时, 以太网模块的 [INIT.]LED 亮。)

(b) 指定 GX Developer 连接目标 (连接到 QCPU[2])



(c) 通过 GX Developer 执行 PING 测试

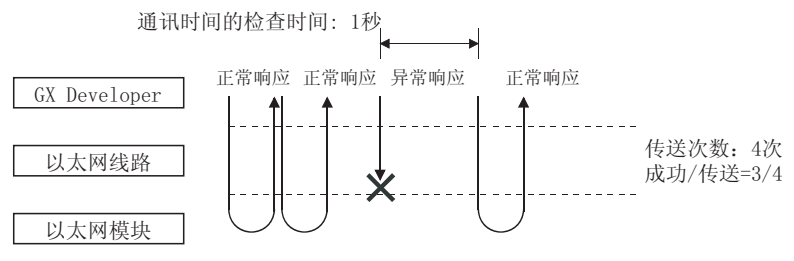
- 1) 在以太网诊断屏上选择 PING 测试。

GX Developer → [诊断] → [以太网诊断] → PING 测试

- 2) 执行以下指出的设置，然后单击执行按钮。

显示 PING 测试的执行结果。

(范例) 以下表示的是当指定“4”为传送计数时 PING 测试的步骤。



在通讯时间的检查时间内接收PING测试响应时发生正常响应。

[PING 测试画面]

(正常完成的例子)

(异常完成的例子)

[显示内容]

项目名称	项目设置说明	设置范围/选项
地址规格	IP 地址	指定 PING 测试目标站的 IP 地址。 (目标站 IP 地址)
	IP 地址输入形式	选择 IP 地址的输入格式。 十进制/十六进制
	主机名	指定 PING 测试目标站的主机名。 -
选项规格	显示主机名	使用结果显示字段中 IP 地址对应的主机名的显示结果。 -
	指定数据量	指定 PING 测试期间传送的系统数据量。 1 到 8192 个字节 (指定以太网模块为 1460 个字节或更少。)
	指定通讯时间的检查时间	指定 PING 测试的完成等待时间。 1 至 30s
	指定传送次数	指定传送次数。 • 执行到中断为止。
结果	显示 PING 测试的结果。 -	
成功/传送	显示 PING 测试期间信息包传送的总数和成功次数。 -	

(地址规格)

按 IP 地址或主机名指定 PING 测试目标站(进行 PING 测试的外部设备)。

- 1) 使用 IP 地址的规格
 - 选择 IP 地址的输入格式(选择: 十进制或十六进制)
 - 按照输入格式指定外部设备的 IP 地址(十进制或十六进制)
- 2) 使用主机名的规格

指定 DNS 服务器中设置的外部设备的主机名, 或安装了 GX Developer 的个人计算机的 HOSTS 文件。

* IP 地址也可以输入在主机名规格字段中。

(选项规格)

设置 PING 测试的详细说明。(如果使用默认规格, 不需要设置。)

- 1) 显示主机名。

选择该项, 则在结果显示字段中显示主机名, 而不是 PING 测试目标设备的 IP 地址。
- 2) 指定数据量。

指定 PING 测试期间要传送的系统数据量。

输入范围: 1 到 8192 个字节(默认: 32 个字节)

* 当指定 1460 个字节或更大的数据量传送到以太网模块时, 如果执行 PING 测试, 则以太网模块将返回 1460 个字节的响应。
- 3) 指定通讯时间的检查时间。

指定 PING 测试的响应等待时间。

输入范围: 1 到 30s(默认: 1s)
- 4) 指定传送次数。

指定要执行 PING 测试的次数。

选择项目	项目说明	备注
指定次数	按指定的次数执行 PING 测试。	传送计数: 1 到 50 次 (默认: 4 次)
执行到中断为止	执行 PING 测试, 直到按下中断按钮为止。	-

(结果)

显示 PING 测试的结果。

<当测试异常完成时>

检查下列项目 然后再次执行 PING 测试。

- 安装在基板上的以太网模块的状况。
- 到以太网的连接状态。
- 写入 PLC CPU 的参数内容。
- PLC CPU 的操作状态(是否发生了错误)。
- 用 GX Developer 设置的 IP 地址和 PING 测试目标站。
- 当更改了以太网模块时, 外部设备是否复位。

(成功/传送)

显示执行 PING 测试时的成功次数和信息包传送总数。

5.4.2 使用 GX Developer 进行 PING 测试(通过 CPU)

本节说明了如何使用 GX Developer 的以太网诊断 PING 测试功能，来检查以太网模块初始化处理的完成状态。

(1) PING 测试

- (a) PING 测试是对与已经完成初始化处理的 GX Developer 连接站(*1)或带有指定 IP 地址的外部设备(比如：个人计算机)具有相同以太网电缆上的以太网模块检查其存在情况。

*1 也可以对 QnA/A 系列以太网模块执行 PING 测试。但是，可以对以下从软件版本 S 开始的 A 系列以太网模块执行 PING 测试：AJ71E71、AJ71E71-B2、AJ71E71-B5。

- (b) 执行 PING 测试可以检查以太网模块的以下项目：

- 线路是否正确连接到了测试目标的以太网模块上。
- 是否正确设置了以太网模块的参数。
- 是否正常完成了以太网模块的初始化处理。

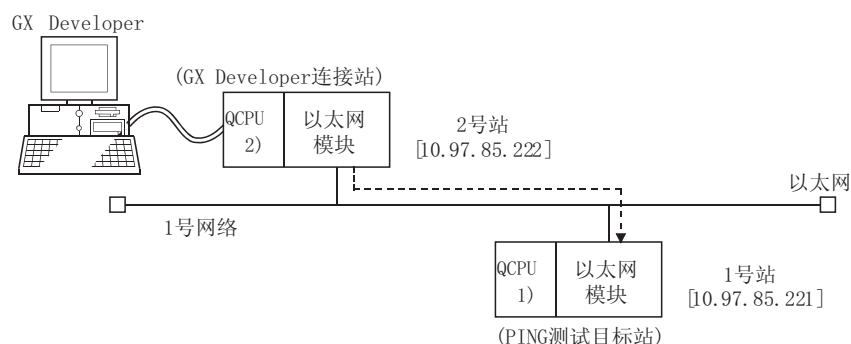
- (c) 可以对和本地站在相同的以太网(相同的副网地址)的以太网模块执行 PING 测试。

注意 PING 测试不可用于直接连接 GX Developer(本地站)的站上安装的以太网模块。(不可对本地站使用 PING 测试)

- (d) 如果要进行 PING 测试的外部设备是一个以太网模块，不要用 GX Developer 把以太网模块的 UDP 端口指定为远程口令核对的有效端口。如果把以太网模块的 UDP 端口指定为远程口令核对的有效端口，则不能执行 PING 测试。

(2) 执行 PING 测试

下面的范例说明了连接到 QCPU 的 GX Developer 对其它站的以太网模块进行 PING 测试时的方法和 GX Developer 设置。



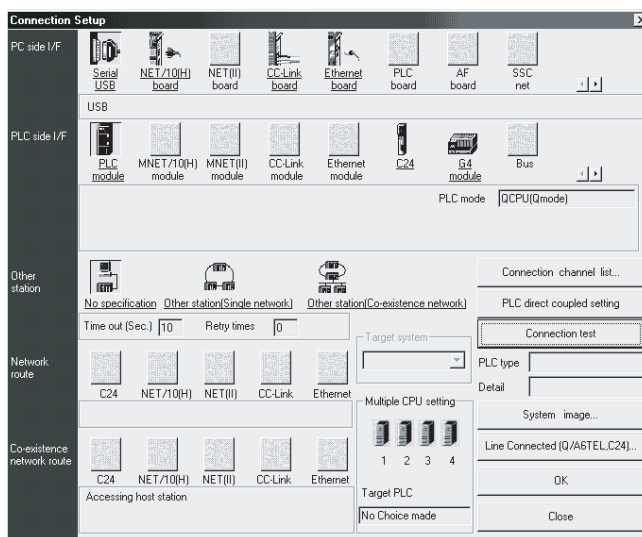
(a) 在 QCPU 站端设置

- 1) 使用 GX Developer 设置以下每个 QCPU 的以太网模块参数。
使用除以下列各项之外的设置项默认值。

设置画面	设置项目	设置说明	
		QCPU 1)	QCPU 2)
设置 MNET/10H 以太网卡号的网络参数	网络类型	以太网	以太网
	起始 I/O 地址	0000	0000
	网络号	1	1
	组号	1	1
	站号	1	2
操作设置	IP 地址	[10. 97. 85. 221]	[10. 97. 85. 222]

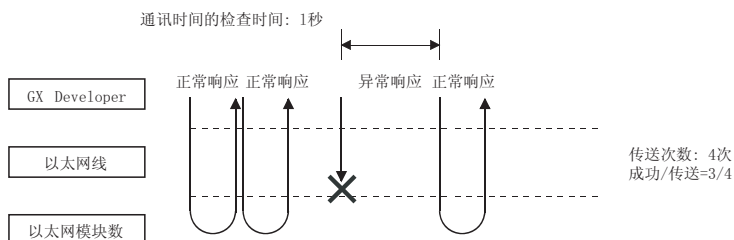
- 2) 将参数写入应用站。
- 3) 当 PLC CPU 重新启动时，完成初始化处理。
(当初始化处理正常完成时，以太网模块的[INIT.]LED 亮。)

(b) 指定 GX Developer 连接目标 (连接到 QCPU 2)



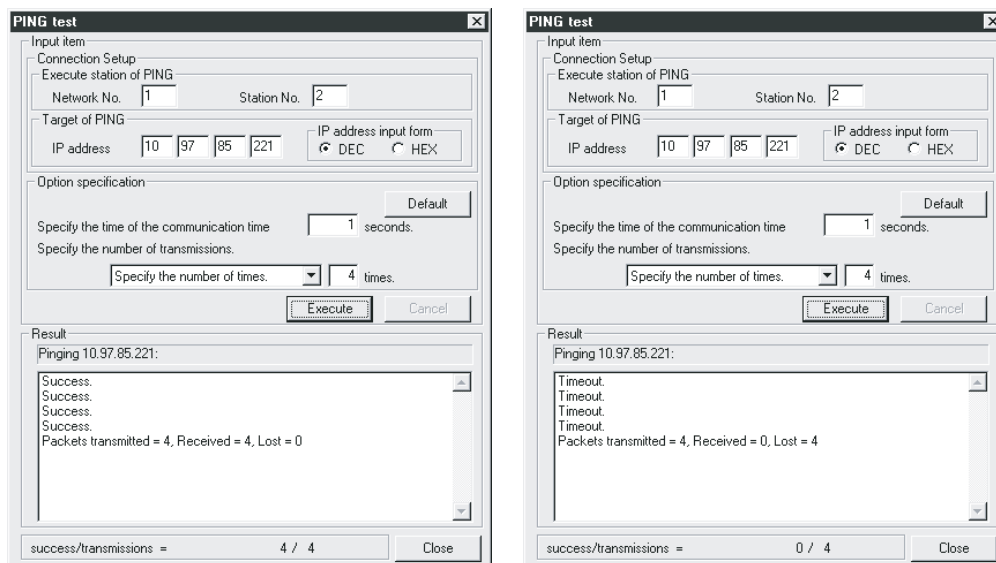
(c) 通过 GX Developer 执行 PING 测试

- 1) 在以太网诊断画面上选择 PING 测试。
GX Developer → [诊断] → [以太网诊断] → PING 测试
- 2) 执行以下所示的设置，然后单击执行按钮。
显示 PING 测试的执行结果。
(范例) 下例说明的是指定“4”为传送计数时的 PING 测试步骤。



当在通讯时间的检查时间内接收到PING测试响应时，发生正常响应。

[PING 测试画面(通过 CPU)]



(正常完成的例子)

(异常完成的例子)

[显示内容]

项目名称	项目设置说明		设置范围/选项
执行 PING 的站	网络号	指定 PING 执行站中以太网模块的网络号。	1 至 239
	站号	指定 PING 执行站中以太网模块的站号。	1 至 64
PING 的目标	IP 地址	指定 PING 测试目标站的 IP 地址。	0000001 _h 至 FFFFFFFE _h
	IP 地址输入形式	选择 IP 地址的输入格式。	十进制/十六进制
选项规格	指定通讯时间的时间	指定 PING 测试的响应等待时间。	1 至 30 秒
	指定传送的次数	指定传送计数。	<ul style="list-style-type: none"> 指定次数。 执行到中断为止。
结果	显示 PING 测试的结果。		-
成功/传送	显示 PING 测试执行期间信息包传送总数及其成功计数。		-

(连接设置)

- 1) 执行 PING 的站(连接到 GX Developer 的站)
指定执行 PING 测试的以太网模块的网络号和站号。
在 GX Developer 的“设置 MNET/10H 以太网卡号的网络参数”画面上设置网络号和站号。
- 2) PING 的目标
指定 PING 测试目标站的 IP 地址(外部设备要经受 PING 测试)。
 - 选择 IP 地址的输入格式(选择十进制或十六进制)。
 - 按照输入格式(十进制或十六进制)指定 PING 测试目标站的 IP 地址。

(选项规格、结果、成功/传送)

显示的信息与通过以太网板执行 PING 测试时显示的内容相同。请参阅第 5.4.1 节。

5.4.3 使用 GX Developer 进行环路回送测试

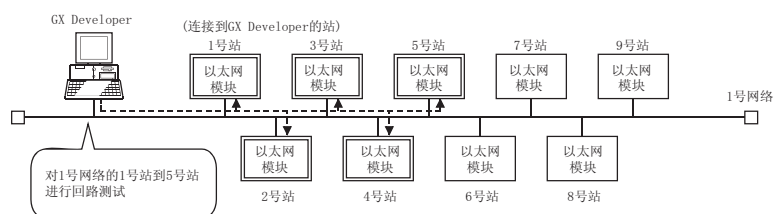
本节说明了如何使用 GX Developer 以太网诊断的环路回送测试功能，来检查以太网模块初始化处理的完成状态。

(1) 环路回送测试

- (a) 在网络中可以对 GX Developer 连接目标进行环路回送测试。测试包括按回路请求目标和所有站号范围之内 (功能版本 B 或以上版本) 的以太网模块指定的有关网络的环路回送测试信息的传输，检查每一模块的初始化处理是否已经完成。

* 可以通过以太网板 在连接网络上执行环路回送测试。

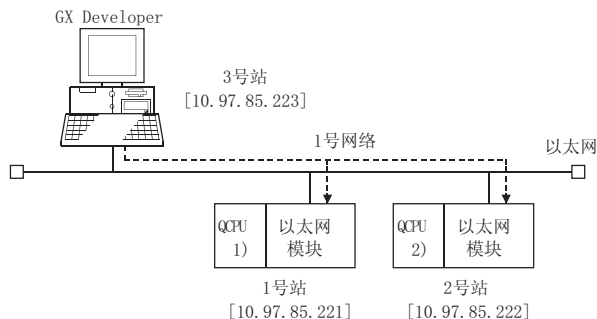
* 由于功能版本 A 以太网模块和 QnA/A 系列以太网模块没有响应此请求的功能，因此不能检查该测试结果。



- (b) 执行环路回送测试可以检查以下项目：
- 线路是否正确连接到了测试目标的以太网模块上。
 - 是否正确设置了以太网模块的参数。
 - 是否正常完成了以太网模块的初始化处理。
- (c) 可以对作为本地站的同一以太网 (同一副网络 ID) 上的以太网模块执行环路回送测试。
- (d) 在安装了要进行环路回送测试的以太网模块站中，不要使用 GX Developer 把以太网模块的 UDP 端口指定为远程口令核对的有效端口。如果把它指定为远程口令核对的有效端口，则不能执行环路回送测试。
- (e) 当使用 GX Developer 执行环路回送测试时，在 GX Developer 的路由中继参数设置中把路由器的中继功能设置为“不使用”。

(2) 执行环路回送测试

下面的例子说明了当使用 GX Developer 对同一以太网中的以太网模块执行环路回送测试时的方法和 GX Developer 设置。



(a) QCPU 站端的设置

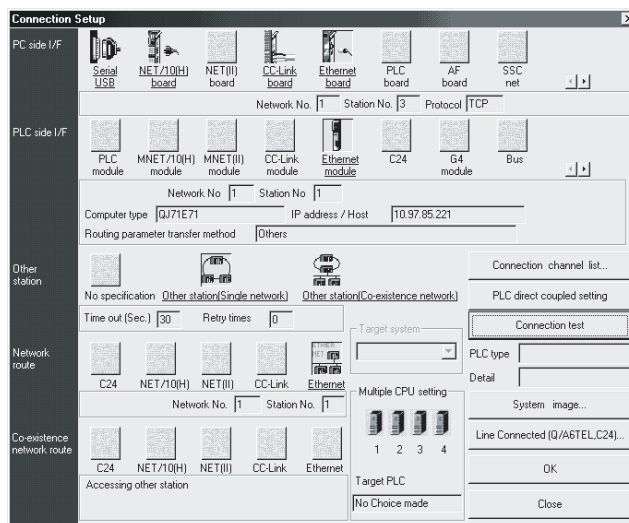
- 1) 通过 GX Developer 为每个 PLC CPU 设置以下以太网模块参数。
对于以下所列之外的设置项目, 使用默认值。

设置屏	设置项目	设置说明		备注
		QCPU 1)	QCPU 2)	
设置 MNET/10H 以太网卡号的网络参数	网络类型	以太网	以太网	请参阅第 4.6 节 设置本站的网络号等。
	起始 I/O 地址	0000	0000	
	网络号	1	1	
	组号	1	1	
	站号	1	2	
操作设置	IP 地址	[10.97.85.221]	[10.97.85.222]	请参阅第 4.7 节 输入本站的 IP 地址。
MNET/10 路由中继参数	MNET/10 路由系统	图表交换系统	图表交换系统	请参阅用户手册(应用篇)的第 3.3.1 节。
	网络号	1	1	
	站号	3	3	
	IP 地址	[10.97.85.223]	[10.97.85.223]	

* 当指定“自动响应系统”为 MNET/10 路由中继参数下的 MNET/10 路由系统时, 不需要为使用 GX Developer 而设置(执行环路回送测试的站)的网络号、站号和 IP 地址。

- 2) 将参数写入应用站。
- 3) 当重新启动 PLC CPU 时完成初始化处理。
(当初始化处理正常完成时, 以太网模块[INIT.]LED 亮。)

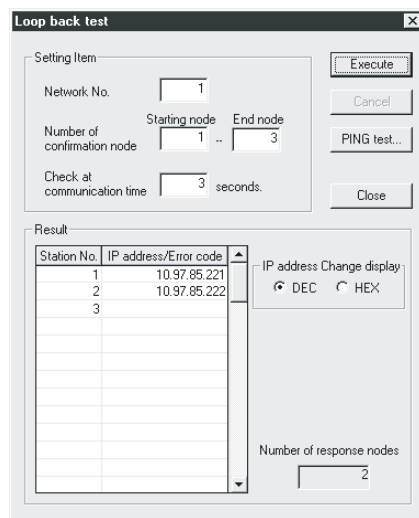
(b) 指定 GX Developer 的连接目标(连接到 QCPU[1])



(c) 通过 GX Developer 执行环路回送测试

- 1) 在以太网诊断画面上选择环路回送测试。
GX Developer → [诊断] → [以太网诊断] → 环路回送测试
- 2) 执行以下所示的设置，然后单击执行按钮。
显示环路回送测试的执行结果。

[环路回送测试画面]



[显示内容]

项目名称	项目设置说明		设置范围/选项
设置项目	网络号	指定环路回送测试目标模块的网络号。	1 至 239
	确认节点数	指定环路回送测试目标模块的站号范围。	1 至 64
	在通讯时间内检查	指定环路回送测试的响应等待时间。	1 至 99s

(网络号(1至239))

指定环路回送测试请求目标的网络号。

输入范围: 1至239

(确认节点数(1至64))

1) 使用站号指定要进行环路回送测试的以太网模块的范围(起始节点和结束节点)

输入范围: 1至64

2) 有顺序的从起始节点执行环路回送测试。

* 一旦接收到执行环路回送测试站的响应,或在通讯时间的检查时间过后,有顺序的对下一个目标站到结束站执行环路回送测试。

(在通讯时间内检查)

指定每个站环路回送测试的响应等待时间。

输入范围: 1至99s(默认: 10s)

(结果)

显示环路回送测试的结果。

1) 显示已正常完成环路回送测试的以太网模块的IP地址。

* 如果由于环路回送测试目标模块端的参数设置错误,多个站都设置了相同IP地址或站号,那么只会显示第一个发出响应的站的结果。

2) 将显示环路回送测试异常完成的以太网模块的“无响应”或出错代码。

* 使用GX Developer版本6.03D或早期产品不会显示“无响应”。

3) 如果指定本站为环路回送测试目标站(执行环路回送测试的站),本站环路回送测试的结果将显示“无响应”。

* 使用GX Developer版本6.03D或早期产品不会显示“无响应”。

(d) 当测试异常完成时的纠正措施。

下表表示的是进行异常完成环路回送测试时的显示内容、目标以太网模块的状态、原因和纠正措施。

环路回送测试结果的显示内容	目标以太网模块的状态	原因	纠正措施
“IP地址”	初始化处理正常完成状态(INIT.)LED亮	-	-
“无响应”	没有出错	目标以太网模块的初始化处理没有正常完成。	检查下列参数的设置值: • 设置MNET/IOH以太网卡号的网络参数 • 操作设置 • 初始化设置
		与目标以太网模块的连接有错。(电缆连接断开、断线等)	• 检查电缆 • 检查收发机

环路回送测试结果的显示内容	目标以太网模块的状态	原因	纠正措施
“无响应”	没有出错	目标以太网模块的 IP 地址不正确。(级或副网络 ID 与以太网模块的设置不同。)	检查操作设置的参数设置值。
		给多个目标以太网模块设置了相同的 IP 地址。	
		给多个目标以太网模块设置了相同的网络号和站号。	对“无响应”模块执行 PING 测试。如果测试正常完成,检查设置 MNET/10H 以太网卡号的网络参数的参数设置值。
	出错	以太网线路处于高负荷状态(包括发生相当于出错代码 C030H, C031H 的错误。)	当以太网线路上的负载太小时执行另一个测试。
未设置路由参数。(发生相当于出错代码 C080H 的错误。)		检查路由参数的设置值。(请参阅用户手册(应用篇)的第 3 章)。	
“出错代码”	没有出错	目标以太网模块所用的 GX Developer UDP 的远程口令状态处于锁定状态。	删除远程口令设置并将参数写入 PLC CPU。
		目标以太网模块是功能版本 A 模块。	检查主体模块的型号和功能版本。
	出错	以太网线路处于高负荷状态(包括发生相当于出错代码 C030H, C031H 的错误。)	当以太网线路上的负载太小时执行其它的测试。

要点

- (1) 环路回送测试只可以在功能版本 B 或以上版本的以太网模块上执行。无法检查下列模块的测试结果。(将显示“无响应”。)然而,使用版本 6.03D GX Developer 或前期产品时不显示。)另外,不可把它们计算在总站数中。
 - 功能版本 A 以太网模块, QnA/A 系列以太网模块
(这是因为没有响应该请求的功能。)
 - 没有完成初始化处理的以太网模块
(这是因为初始化处理还没有完成。)

* 如果没有响应环路回送测试功能的功能版本 A 以太网模块或 QnA/A 系列以太网模块接收环路回送测试请求,下列出错代码将存储在缓冲存储器的出错日志区中。

 - 在功能版本 A 以太网模块或功能版本 B QnA 系列以太网模块中,出错代码为 4080H。
 - 在功能版本 A QnA 系列以太网模块或 A 系列以太网模块中,出错代码为 50H。
- (2) 检查完出错内容,对异常完成环路回送测试的以太网模块采取纠正措施之后,重新起动安装了以太网模块的站。重新起动时,将执行以太网模块的初始化处理。使用 PING 测试检查以太网模块初始化处理的完成情况。也可以在“环路回送测试”屏上执行 PING 测试。
- (3) 如果由于环路回送测试异常完成而显示出错代码,检查对应出错代码的出错内容,然后参考第 11.3.3 节采取纠正措施。

5.4.4 PING 命令(个人计算机 → 以大网模块)

下面的例子说明了如何从连接到同一以太网的外部设备向本站以太网模块发布 PING 命令, 来确认初始化处理的完成。(本例中, 是在 IP 地址等级和子网地址完全相同的设备之间进行确认。)

<指定方法>

PING 的 IP 地址

<范例>

以太网模块的 IP 地址: 192.0.1.254

正常完成时的画面样本

```
C:\>ping 192.0.1.254...Execute the ping command

Pinging 192.0.1.254 with 32 bytes of data:

Reply from 192.0.1.254: bytes=32 time=1ms TTL=128
Reply from 192.0.1.254: bytes=32 time<10ms TTL=128
Reply from 192.0.1.254: bytes=32 time<10ms TTL=128
Reply from 192.0.1.254: bytes=32 time<10ms TTL=128

Ping statistics for 192.0.1.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss)
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>_
```

异常完成时的画面样本

```
C:\>ping 192.0.1.254...Execute the ping command

Pinging 192.0.1.254 with 32 bytes of data:

Request timed out:
Request timed out:
Request timed out:
Request timed out:

Ping statistics for 192.0.1.254:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>_
```

<当 PING 命令没有成功响应时>

检查下列几项并再次发送 PING 命令。

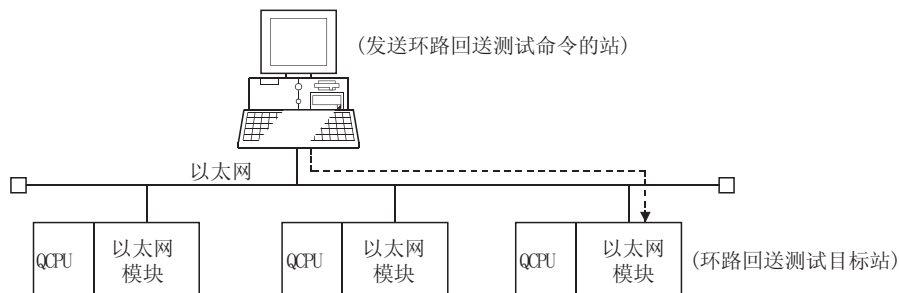
- 检查连接到基板上的以太网模块附件。
- 检查到以太网的连接。
- 检查写入 PLC CPU 的每个参数的内容。
- 检查 PLC CPU 的运行状况(有没有无规律的情况出现?)。
- 检查 PING 命令指定的以太网模块的 IP 地址。

5.4.5 环路回送测试(使用 MC 协议进行通讯)

为了检查目标以太网模块初始化处理的完成状态，可以使用 MC 协议进行的通讯来执行环路回送测试。

下面简要地说明了针对使用 MC 协议进行通讯的环路回送测试。详细说明请参阅 Q 系列 MELSEC 通讯协议参考手册。

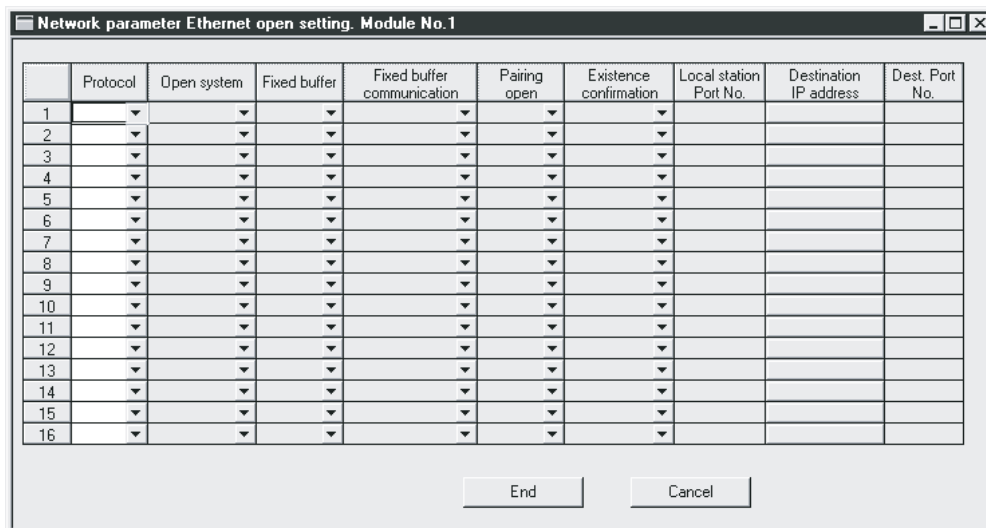
- (1) 针对使用 MC 协议进行通讯的环路回送测试
环路回送测试是检查下列情况的一种功能：
 - 是否将线路正确连接到了测试目标的以太网模块上。
 - 是否正确设置了以太网模块的参数。
 - 是否正常完成了以太网模块的初始化处理。
 - 外部设备的程序是否正常运行。
- (2) 当使用 MC 协议与以太网模块端的用户端口进行通讯时，有必要连接线路。对于用在以太网模块端的连接，执行开放处理。
- (3) 该功能只可以用于本地站以太网模块。不能通过网络系统把它用于其它站的以太网模块。



5.5 开放设置

本节说明了使用 GX Developer 进行的开放设置。

选择[设置 MNET/10H 以太网卡号] - [开放设置]起动[以太网开放设置]画面。



顺控程序最多可以执行与 16 个站的外部设备的开放处理 (建立连接)。

一旦与外部设备开放, 就可能使用 MC 协议、固定缓冲存储器通讯和随机访问缓冲存储器通讯进行通讯。

因而, 即使在使用 MC 协议进行通讯和随机访问缓冲存储器通讯时也需要开放处理。

项目名称	设置说明	设置范围/选项
协议	设置通讯方法(协议)。	<ul style="list-style-type: none"> • TCP/IP • UDP/IP
开放系统	选择连接建立系统。	<ul style="list-style-type: none"> • 主动开放 • 非被动开放 • 完全被动开放 • MELSOFT 连接 • OPS 连接
固定缓冲存储器	选择固定缓冲存储器的使用。	<ul style="list-style-type: none"> • 发送 • 接收
固定缓冲存储器通讯	选择固定缓冲存储器通讯使用哪一个协议。	<ul style="list-style-type: none"> • 有顺序 • 无顺序
成对开放	选择是否使用成对开放。	<ul style="list-style-type: none"> • 成对 • 不成对
存在确认	选择是否应该确认连接的目标站继续存在。	<ul style="list-style-type: none"> • 不确认 • 确认
本地站端口号	设置本地站的端口号。	401 _h 至 1387 _h or 138B _h 至 FFFE _h
目标 IP 地址	设置外部设备的 IP 地址。	1 _h 至 FFFFFFFF _h (FFFFFFF _h : 同步广播)
目标端口号	设置外部设备的端口号。	401 _h 至 FFFF _h (FFFF _h : 只可以在接收时设置同步广播)

(1) 协议

(1 到 8 号连接; 地址: 20H 至 27H ... b8)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 选择每个连接的协议。

设置名称	设置说明
TCP	使用 TCP/IP 进行通讯。
UDP	使用 UDP/IP 进行通讯。

(b) 关于协议(TCP/UDP), 请参阅 1.4 “软件配置”。

(2) 开放系统

(1 到 8 号连接; 地址: 20H 至 27H ... b15, b14)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 针对“(1)协议”中选择“TCP”的每个连接、选择开放连接系统。如果选择“UDP”, 不需要该项目规格。

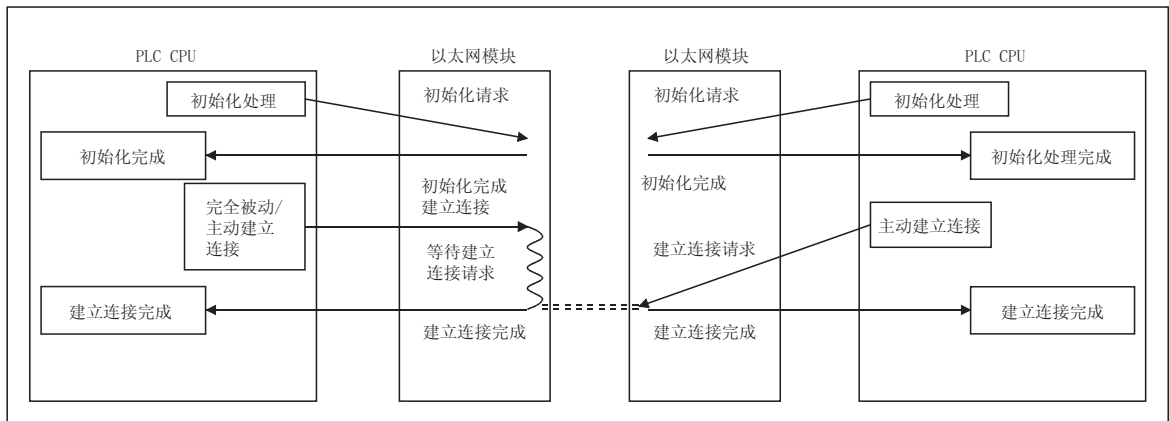
设置名称	设置说明
主动	对于在 TCT 连接上等待被动开放(完全被动/非被动)的外部设备执行主动开放处理。
非被动	对于在 TCP 连接上执行被动开放处理, 对所有连接到网络的设备定址。 (本站处于等待发出主动开放请求的等待状态)
完全被动	对于在 TCP 连接上执行被动开放处理, 只对指定的设备定址。 (本站处于等待发出主动开放请求的等待状态) 本站等待“(8)目标 IP 地址”中设置的对方站发出主动开放请求。
MELSOFT 连接 * 1 * 2 * 3	用于 TCP/IP 通讯连接 MELSOFT 产品 进行 TCP 连接上的被动开放处理, 定址连接到网络的所有 MELSOFT 产品。 (对于发送的主动打开请求, 本站处于等待状态)
OPS 连接 * 1 * 4	用于通过 TCP/IP 通讯连接 OPS 进行 TCP 连接上的被动开放处理, 只对指定 OPSs 定址。 (本站等待“(8)目标 IP 地址”中设置从 OPS 发出的主动开放请求)

*1 不管操作设置中的初始时间设置(请参阅 4.7 节)如何, 该连接将处于始终等待打开状态。

*2 设置连接专用于与 MELSOFT 产品的通讯。

*3 当同时连接到多 MELSOFT 产品时, 设置与 MELSOFT 产品相同数量的连接。(只连接一个产品时无需进行设置。使用专用于系统的连接。)

*4 使用 OPS 上的 MELSOFT 产品, 比如: GX Developer, 进行以太网模块的 TCP/IP 通讯时, 使用专用于系统的连接(GX Developer 通讯 TCP 端口)或在本设置中设置“MELSOFT 连接”。



(3) 固定缓冲存储器

(1 到 8 号连接; 地址: 20H 至 27H ... b0)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 在此处选择使用固定缓冲存储器进行通讯时, 是否使用对应每个适用连接号的固定缓冲存储器进行发送或接收。

设置名称	设置说明
发送	不执行用于发送或固定缓冲存储器的通讯。
接收	用于接收。

(b) 当外部设备使用固定缓冲存储器通讯执行发送和接收时, 发送需要一个缓冲存储器, 接收也需要一个缓冲存储器。因此, 应设置两个连接。

(c) 无论固定缓冲存储器设置为发送或接收, 外部设备都可以使用 MC 协议或随机访问缓冲存储器进行通讯。

(4) 固定缓冲存储器通讯

(1 到 8 号连接; 地址: 20H 至 27H ... b9)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 该项为使用固定缓冲存储器进行通讯时选择的通讯方法。

设置名称	设置说明
有顺序	<ul style="list-style-type: none"> 在固定缓冲存储器通讯中, 通过与外部设备同步交换信息, 数据以 1:1 进行通讯。 使用 MC 协议可以进行通讯, 使用随机访问缓冲存储器也可以进行通讯。
无顺序	<ul style="list-style-type: none"> 无顺序固定缓冲存储器通讯使用的专用连接。 通过同步广播, PLC CPU 和外部设备以 1:1 或 1:n 模式通讯数据。(*1) 必须使用顺控程序与外部设备进行信息交换。

*1 关于同步广播的详细说明, 请参阅第 8.3 节“当使用 UDP/IP 时的同步广播”。

(5) 成对开放

(1 到 8 号连接; 地址: 20H 至 27H ... b7)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 当使用固定缓冲存储器通讯时(可以指定有顺序和无顺序), 选择以太网模块的接收和发送连接是否应该组成一对, 并连接到外部设备的一个端口。

关于该项的详细说明, 请参阅第 5.7 节“成对开放”。

设置名称	设置说明
不成对	不使用成对开放方法。
成对	使用成对开放方法。

(6) 存在确认

(1 到 8 号连接; 地址: 20H 至 27H ... b1)

(9 到 16 号连接; 地址: 使用的系统区)

(a) 该设置选择以太网模块是否应该确认, 在固定期限没有通讯时外部设备仍然正常运行。必须完成对外部设备的开放处理。

设置名称	设置说明
不确认	不确认外部设备的存在。
确认	确认外部设备的存在。 关于设置存在确认时间和其它问题, 请参阅第 5.2 节“初始化设置”。

(b) 当存在确认时发生错误, 以太网模块执行以下处理。

- 强制断开线路, 并在缓冲存储器的出错日志区(地址: E0H 到 1FEH)中储存出错信息。
- 熄灭开放完成信号(地址: 5000H 的对应位), 开放异常检测信号(X18)亮。

(c) 如果 UDP/IP 连接建立时更改外部设备, 则应选择“不确认”。

如果选择“确认”, 则在建立 UDP/IP 连接后, 以太网模块将确认第一个目标的存在。对于更改的目标, 即重新选择的外部设备, 不执行存在确认。

(d) 当无顺序固定缓冲存储器通讯以同步广播传送数据时, 选择“不确认”。

(7) 本地站端口编号

(1 到 8 号连接; 地址: 28H 至 5FH)

(9 到 16 号连接; 地址: 使用系统区)

(a) 在该项中, 以太网模块每个连接的端口编号都是以十六进制设置的。

(b) 在 401H 到 1387H 和 138BH 到 FFFEh 范围中指定设置值。设置其它端口不使用的端口编号。

(因为 1388H 到 138Ah 范围是以太网模块操作系统的使用范围, 因此不能在该范围中指定端口编号。)

(c) 咨询网络管理员后, 再设置以太网模块的端口编号。

(8) 目标 IP 地址

(1 到 8 号连接; 地址: 28H 至 5FH)

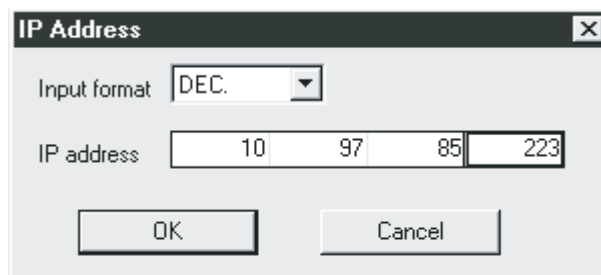
(9 到 16 号连接; 地址: 使用的系统区)

(a) 选择 IP 地址的输入格式(选项: 十进制/十六进制)。

(b) 以选择的输入格式(十进制/十六进制)设置外部设备的 IP 地址(两个字)。

(c) 给出的外部设备的 IP 地址必须是 0H 除外的值。而且, FFFFFFFH 是同步广播通讯用的设置值。

(d) 咨询网络管理员后, 再设置外部设备的 IP 地址。



(9) 目标端口编号

(1 到 8 号连接; 地址: 28H 至 5FH)

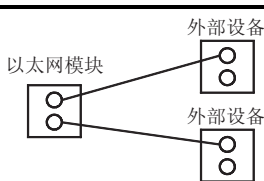
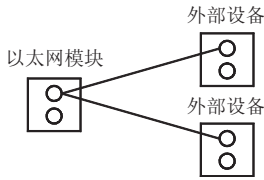
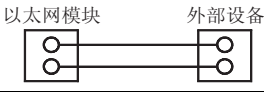
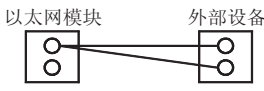
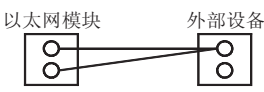
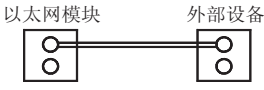
(9 到 16 号连接; 地址: 使用的系统区)

(a) 以十六进制设置每个连接的外部设备的端口编号。

(b) 在 401H 到 FFFFH 范围中设置外部设备的端口编号, 此外, FFFFH 是同步广播通讯用的设置值。

(c) 咨询网络管理员后, 再设置以太网模块的端口编号。

(d) 下表提供了设置端口编号时应遵守的一些注意事项。
(图中的□表示设备, ○表示端口编号。)

连接建立的状态 (○: 端口(表示端口号))	连接说明	通讯协议	
		TCP	UDP
	当连接多个外部设备时, 为本站设置多个端口号。	○	○
	当连接多个外部设备时, 为本站设置多个端口号。(但是, 有必要为每台外部设备开放) 当本站是非被动时不能选择这一项。	○	×
	当连接外部设备的多个端口时, 为以太网模块设置多个端口号。	○	○
	当连接外部设备的多个端口时, 为以太网模块设置单个端口号。(但是, 有必要为每台外部设备端口开放) 当本站是非被动时不能选择这一项。	○	×
	当连接到外部设备的相同端口时, 为以太网模块设置多个端口号。(但是, 有必要为每个以太网模块端口开放)	○	○
	当设置成对开放时, 只能为外部设备和以太网模块的同一端口设置多个端口号。	○	○

要点

当在操作设置中选择“始终等待打开(在停止时间可以进行通讯)”时, 必须在该画面上设置通过被动开放和 UDP 开放进行通讯的连接参数。(请参阅第 4.7 节)。

要点		按照开放连接所使用的开放方法设置参数。					
		通讯系统 开放系统		TCP		UDP	
参数		外部设备的 ARP 功能		非被动		外部设备的 ARP 功能	
		是	否	非被动	完全被动	是	否
通讯地址	本地站端口号	○	○	○	○	○	○
	目标 IP 地址	○	○	×	○	○	○
	目标端口号	○	○	×	○	○	○
	目标以太网地址 (* ²)	○(* ¹)	○	×	×	○(* ¹)	○

*1 使用默认值 (FFFFFFFFF₁₆) 或 “0”。

*2 当使用 GX Developer 的 “开放设置” 时，使用默认值。
 当与没有 ARP 功能的外部设备进行通讯时，使用专用 OPEN 指令并用控制数据设置外部设备的以太网地址。

5.6 开放处理/关闭处理

本节说明了使用顺控程序进行开放处理/关闭处理的情况。

(1) 开放处理

- (a) 开放处理的目的是为了执行下列形式的数据通讯而与外部设备建立连接。这些对外部设备开放的处理都可以由用户来执行。
 - 使用 MC 协议进行通讯
 - 使用固定缓冲存储器(有顺序)的发送/接收
 - 使用随机访问缓冲存储器进行通讯
- (b) 当使用 GX Developer 的参数设置设置下列项目时, 应该用顺控程序执行开放处理。
 - 1) 在操作设置中(第 4.7 节)
 当在“初始时间设置”中设置“不等待打开”时。
 - 2) 在开放设置中(第 5.5 节)
 当在“开放系统”设置中设置“主动”时。
- (c) 为了执行开放处理, 必须完成初始化处理。
- (d) 当使用 MC 协议、固定缓冲存储器或随机访问缓冲存储器进行通讯时, 必须建立(开放处理)与外部设备的连接。(*1)
 用户也可以与开放的外部设备进行上述三种类型的数据通讯。
 *1 由于以太网模块是按 IP 地址识别与之通讯的外部设备的, 所以 UDP 通讯需要开放处理。
- (e) 最多可以与外部设备建立 16 个连接。但是, 使用固定缓冲存储器通讯与相同外部设备进行通讯时, 需要两个缓冲存储器, 因此, 在这种情况下, 可以进行通讯的外部设备数目会较少。

要点
当使用 MC 协议或随机访问缓冲器进行通讯时, 如果安装了以太网模块的站中的 PLC CPU 处于停止状态时, 数据通讯仍继续, 则在“初始时间设置”中设置“始终等待打开(在停止时间可能进行通讯)”设置(请参阅 4.7“操作设置”。)

(2) 关闭处理

- (a) 关闭处理的目的在于断开(取消)前述通过开放处理已经与外部设备建立的连接。
- (b) 当终止与外部设备的连接、改变连接的外部设备、改变通讯条件等时, 使用关闭处理。
- (c) 断开用顺控程序建立的连接。
- (d) 确定断开与外部设备的连接的处理时序。

下面的例子说明通过对 1 号连接进行开放处理和关闭处理, 建立从以太网模块到外部设备的连接, 并在随后再次关闭的步骤。

- TCP/IP 主动开放 : 请参阅 5.6.1、“主动开放处理/关闭处理”。
- TCP/IP 被动开放 : 请参阅 5.6.2、“被动开放处理/关闭处理”。
- UDP/IP 开放 : 请参阅 5.6.3、“UDP/IP 开放处理/关闭处理”。

要点

不要使用输入/输出信号对同一连接执行开放/关闭处理; OPEN/CLOSE 处理时, 也不要一起使用专用 OPEN/CLOSE 指令。它可能导致故障。

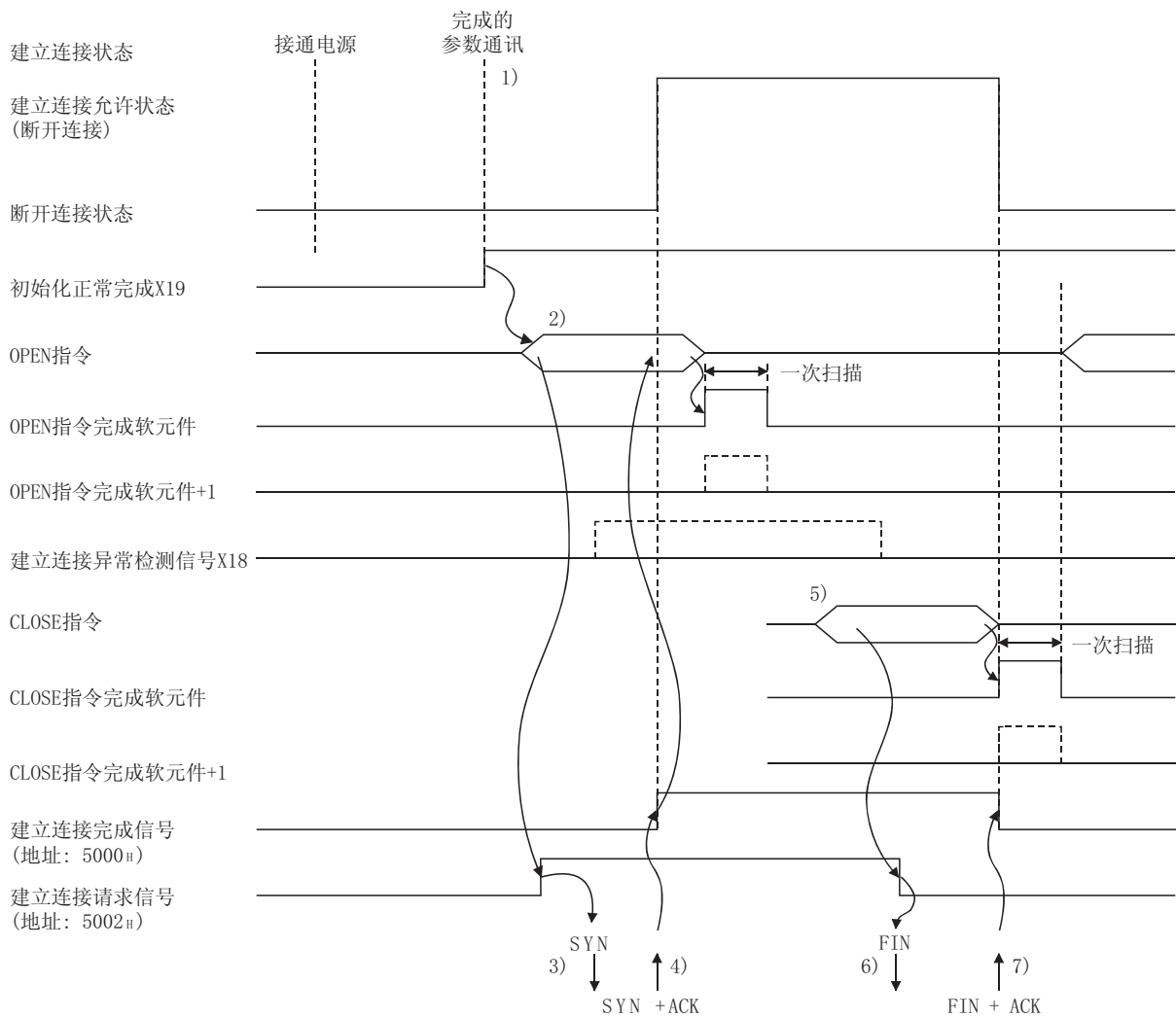
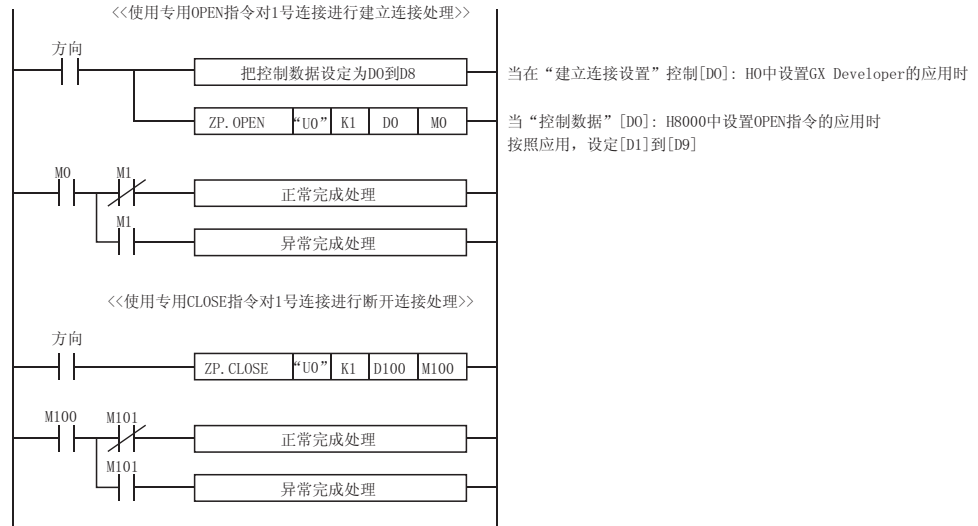
要点

除非请求关闭处理, 在下列情况下, 开放完成信号(地址: 5000H 的应用位)自动关闭, 通讯线路关闭:

- (1) 当存在确认功能发生超时(请参阅第 5.5 节)。(※1)
- (2) 当接收到外部设备发出的关闭或 ABORT (RST) 指令时。
- (3) 在 TCP 开放完成状态时再次接收外部设备发出主动开放请求。
根据其版本, 以太网模块进行如下操作。
 - (a) 以太网模块的前 5 位系列号为 05051 或以上数字。
返回 ACK 到外部设备后, 从外部设备接收 RST 指令时, 以太网模块切断连接。
 - (b) 以太网模块的前 5 位系列号为 05049 或之前数字。
发送 RST 指令后, 以太网模块切断连接。
 但是, 再次接收不同 IP 地址或端口号外的部设备发出主动开放请求时, 以太网模块只发送 RST 指令。(不切断连接。)

5.6.1 主动开放处理/关闭处理

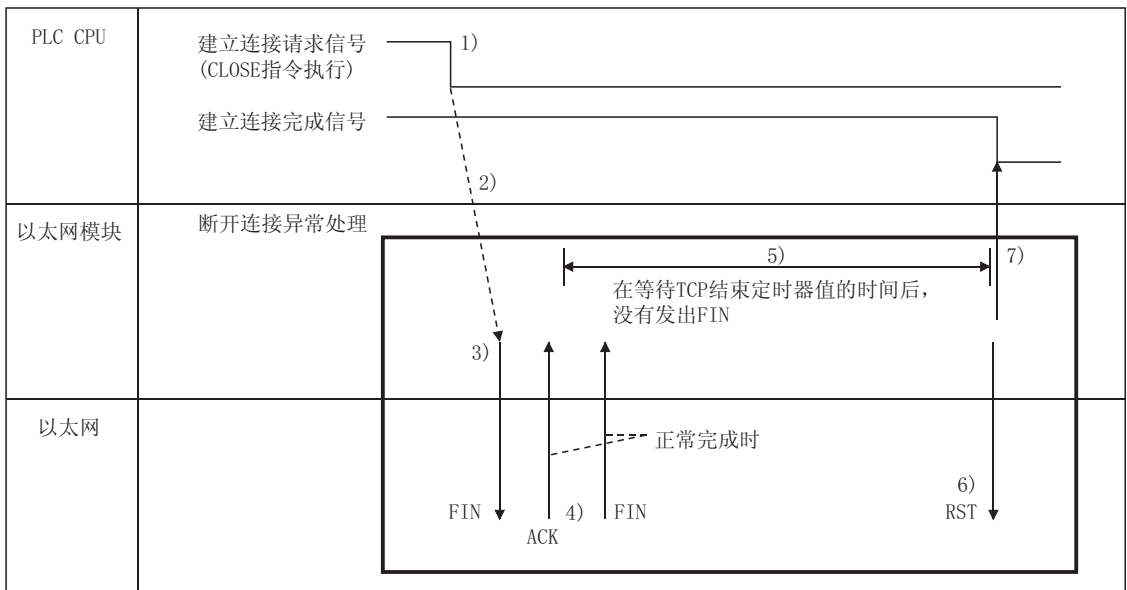
本节说明用以太网模块开放和关闭与外部设备的连接的步骤。



- 1) 参数设置通讯后，确认以太网模块初始化处理的正常完成。
(初始化正常完成信号(X19)：开)
- 2) 使用专用 OPEN 指令开始开放处理。
(开放请求信号(地址：5002_H ... b0)：开)
- 3) 以太网模块执行开放处理。
• 发出开放请求(SYN)。
- 4) 当开放处理正常完成时
 - 开放完成信号(地址：5000_H ... b0) : 开
 - OPEN 指令完成软元件 : 开
 - OPEN 指令完成软元件 + 1 : 关
 - OPEN 指令完成状态区(*1) : 0000_H
 允许数据通讯。
 当异常完成开放处理时。(*2)
 - 开放完成信号 : 关
 - OPEN 指令完成软元件 : 开
 - OPEN 指令完成软元件 + 1 : 开
 - 开放出错代码存储在缓冲存储器中。(*3)
 - OPEN 指令完成状态区(*1) : 除 0000_H 之外的值
 - 开放异常检测信号。(X18) : 开
- 5) 使用专用 CLOSE 指令开始关闭处理。
(开放请求信号：关)
- 6) 以太网模块执行关闭处理。
• 发送关闭处理(FIN)。
- 7) 当关闭处理正常完成时
 - 开放完成信号 : 关
 - CLOSE 指令完成软元件 : 开
 - CLOSE 指令完成软元件 + 1 : 关
 - CLOSE 指令完成状态区(*1) : 0000_H
 当关闭处理异常完成时(*4)
 - 开放完成信号 : 关
 - CLOSE 指令完成软元件 : 开
 - CLOSE 指令完成软元件 + 1 : 开
 - CLOSE 指令完成状态区(*1) : 除 0000_H 之外的值

要点	本例使用 1 号连接进行说明。其它编号的连接使用相应的信号和位。
-----------	----------------------------------

- *1 完成时的结束代码存储在专用指令完成状态区。关于专用指令的详细说明，请参阅第 10 章“专用指令”。
- *2 当异常完成开放处理 (TCP) 时进行处理
 当在正常情况下与以太网模块开放时，如果以太网模块发出 SYN，则外部设备返回 ACK 和 SYN。
 但是，如果以太网模块发出 SYN，然后外部设备返回 RST，则开放异常完成 (X18) 立即亮，开放处理结束。
- *3 可以用下列缓冲存储器确认异常结束时的开放状态和出错代码。
 - 通讯状态存储区的每个连接开放异常代码区：
 (1 到 8 号连接；地址：78_H 至 C7_H)
 (9 到 16 号连接；地址：5020_H 至 586F_H)
 - 出错日志区 (地址：E0_H 至 177_H)
 - 当再次执行专用 OPEN 指令时清除存储在开放异常代码区中的出错代码 (n → 0)。
- *4 当异常完成关闭处理 (TCP) 时进行处理
 当正常断开与以太网模块的连接时，以太网模块发出 FIN 请求，外部设备返回 ACK 和 FIN。
 然而，如果因为外部设备故障而没有返回 ACK 和 FIN，以太网模块强制断开连接 (发送 RST 信息)。



- 1) 使用专用 CLOSE 指令关闭开放请求信号。
- 2) 以太网模块执行关闭处理。
- 3) 以太网模块向外部设备发出 FIN 请求。
- 4) 外部设备发回 FIN 和 ACK 信息来答复以太网模块发出的 FIN 请求。
(如果没有回答, 以太网模块将再次发出 FIN 请求。)
- 5) 以太网模块等待外部设备发出 ACK 和 FIN。
(模块等待以 TCP 结束定时器值设置的时间量的时间。关于如何设置的详细说明, 请参阅第 5.2 节“初始化设置”。)
如果此时接收到 ACK 和 FIN 信息, 它返回一个 ACK, 表示处于正常处理中。
- 6) 如果在 TCP 结束定时器值指定的时间内没有接收到 ACK 和 FIN, 向外部设备发出 RST 信息。
- 7) 不管外部设备的状态如何, 以太网模块都可以确定关闭步骤的完成并关闭开放完成信号。

备注

- (1) 当执行上述步骤时, 以太网模块确定外部设备的关闭是否正确执行, 因而, 关闭处理的结果不存储在出错日志区中。
- (2) 以上所述的步骤是以太网模块的特殊功能; 对于通用的 TCP/IP 协议不适用。

程序范例

本例说明当在开放系统设置中选择主动开放时，开放处理/关闭处理用的程序。

(1) 程序范例的执行环境


- (a) 以太网模块安装在基板的“0”槽中。
- (b) 假定已使用 GX Developer 设置[设置 MNET/10H 以太网卡号的网络参数]设置，如下所示：
- 网络类型 : 以太网
 - 起始 I/O 地址 : 0000
 - 网络号 : 1
 - 组号 : 1
 - 站号 : 1
- (c) 假定已使用 GX Developer 设置了[操作设置]，如下所示：

本站 IP 地址 : 0A.61.55.DE_H(10.97.85.222)

- (d) 假定已使用 GX Developer 设置了[开放设置]参数，如下所示：

	Protocol	Open system	Fixed buffer	Fixed buffer communication	Pairing open	Existence confirmation	Local station Port No.	Destination IP address	Dest. Port No.
1	TCP	Active	Send	Procedure exist	No pairs	No confirm	1000	10.97.85.223	2000
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									

本站端口编号 : 1000_H
 目标 IP 地址 : 0A.61.55.DF_H(10.97.85.223)
 目标端口编号 : 2000_H

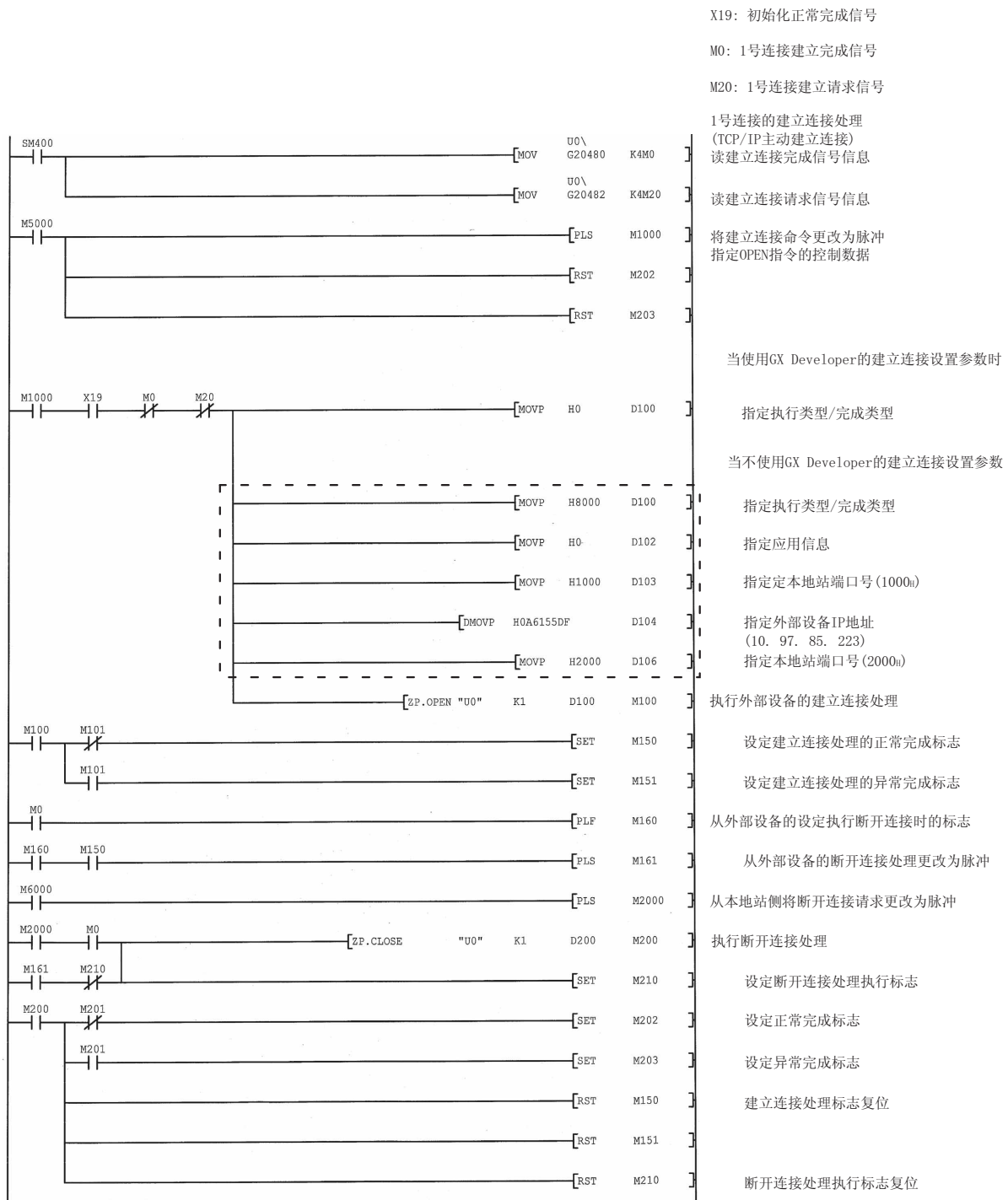
- (e) 程序中使用下列触点信号:
- | | |
|-------------------|---------------------|
| 1号连接开后完成信号 | : M0 |
| 1号连接开放请求信号 | : M20 |
| 1号连接 OPEN 指令控制数据 | : 存储在 D100 至 D109 中 |
| 1号连接 CLOSE 指令控制数据 | : 以 D200 和 D201 存储 |
- (f) 当[开放设置]以太网模块参数没有设置用于 GX Developer 时, 应该使用程序范例中  封住的区域。
当[开放设置]参数用于 GX Developer 时, 不需要这部分程序。
- (g) 关于专用 OPEN 指令的详细说明, 请参阅第 10 章“专用指令”。

(2) 程序范例概述

- (a) 使用 GX Developer 设置每一个参数, 将它写入 PLC CPU 后, 重新启动 PLC CPU 并确认初始化处理的完成。
- (b) 以太网模块对用[开放设置]或控制数据设置的外部设备的 1 号连接执行开放处理。
- (c) 按照发送给以太网模块的 CLOSE 指令或从外部设备发出的关闭请求, 对一号连接执行关闭处理。

备注

程序中所示的“U0\G20480”和“U0\G20482”代码指定缓冲存储器中的下列区:
 U0\G20480: 开放完成信号存储区(地址: 5000_H(20480))
 U0\G20482: 开放请求信号存储区(地址: 5002_H(20482))



5.6.2 被动开放处理/关闭处理

本节说明从外部设备与以太网模块开放和关闭的步骤。

如本节所述，被动开放处理/关闭处理的操作根据选择“始终等待打开”，或是选择“不等待打开”的不同而不同。

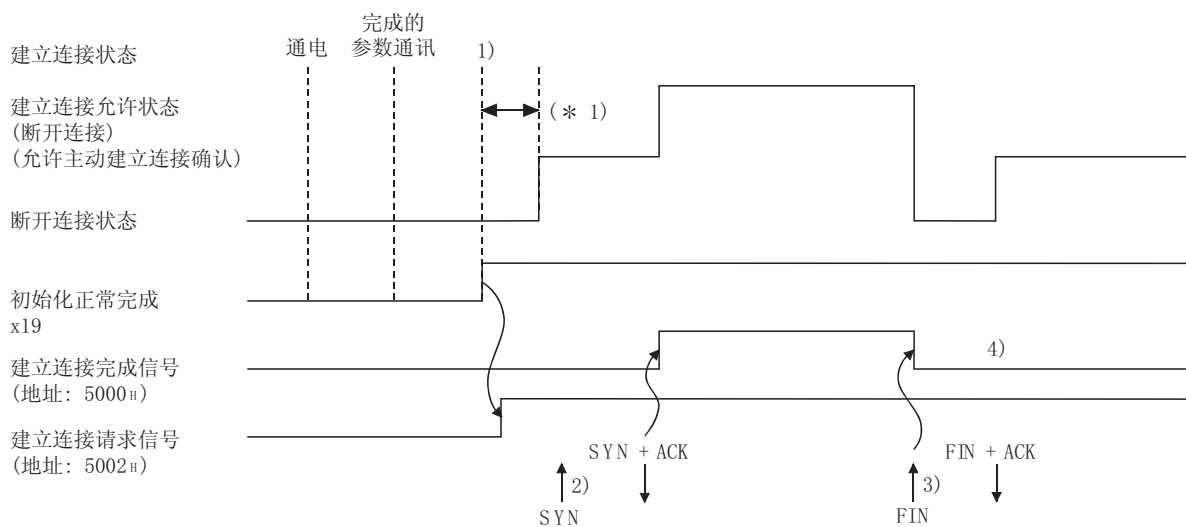
(1) 当在操作设置中选择“始终等待打开”时

当使用 GX Developer 选择[操作设置]-[初始时间设置]-[始终等待打开(在停止时间可能进行通讯)]时，开放处理/关闭处理按如下所述进行。

在本例中，因为按照[开放设置]参数设置，以太网模块保持连接处于始终等待打开

状态，所以不需要用于开放处理和关闭处理的顺控程序。

关于[开放设置]参数的详细说明，请参阅第 5.5 节，“开放设置”。



- 1) 发送出参数之后，确认以太网模块初始化处理的正常完成(初始化正常完成信号(X19)：开)
初始化处理正常完成之后，连接处于开放允许状态，以太网模块等待外部设备发出开放请求。
 - 2) 一旦接收到外部设备发出的开放请求(SYN)，以太网模块就开始开放处理。
当正常完成开放处理时，开放完成信号(地址：5000H... b0)亮并允许数据通讯。
 - 3) 一旦接收到外部设备发出的关闭请求(FIN)，以太网模块就开始关闭处理。
当关闭处理完成时，开放完成信号关闭，并禁止数据通讯。
 - 4) 以太网模块的内部处理完成后，连接返回到开放允许的确切状态。
- *1 在正常完成初始化处理后和以太网模块处于开放允许状态前接收的开放请求(SYN)会发生状态错误，以太网模块发出强制关闭命令(RST)。

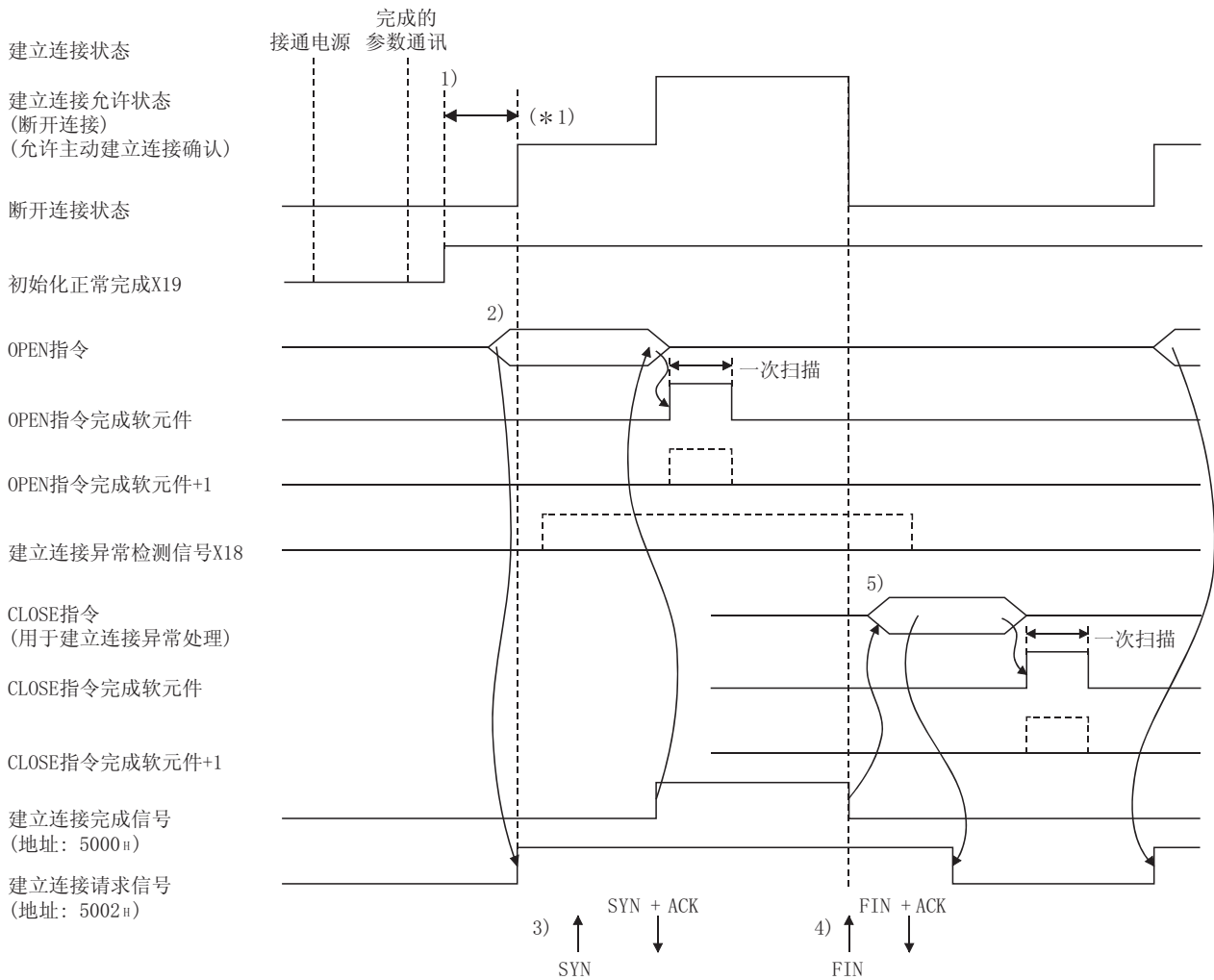
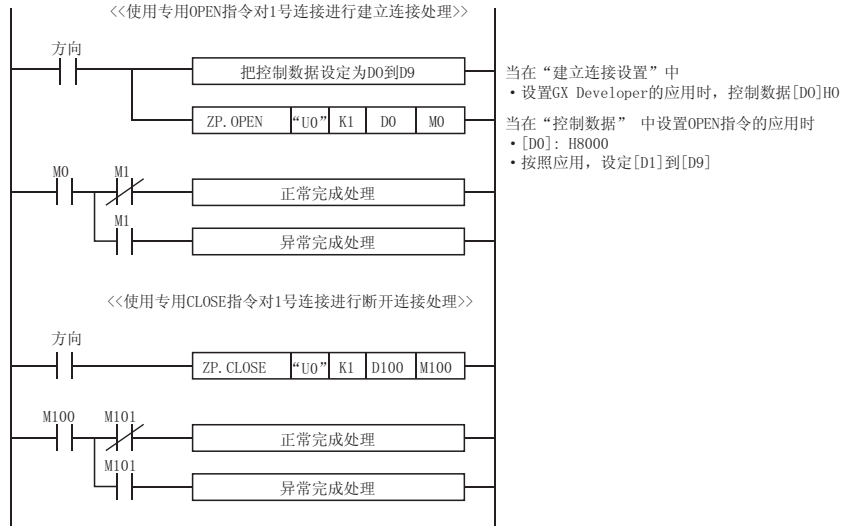
备注

对于在“操作设置”中选择“始终等待打开(在停止时间可以进行通讯)”的被动开放连接，按照从外部设备发出的开放/关闭请求，执行以太网模块端的开放/关闭处理。当从以太网模块端使用专用 CLOSE 指令执行关闭处理时，在关闭处理之后，应用连接将不会返回到开放允许确认状态。(对于选择了“不等待打开(在停止时间不能进行通讯)”的被动开放，需要进行相同的开放处理)。

(2) 当在操作设置中选择“不等待打开”时

当使用 GX Developer 选择[操作设置]-[初始时间设置]-[不等待打开(在停止时间内不能进行通讯)]时，开放处理/关闭处理按如下所述进行。

由于开放处理/关闭处理由顺控程序执行，因此在开放时可以更改外部设备。



- 1) 参数设置通讯后，确认以太网模块初始化处理的正常完成。
(初始化正常完成信号(X19)：开)
- 2) 使用专用 OPEN 指令开始开放处理。
(开放请求信号(地址：5002_H … b0)：开)
- 3) 一旦接收到外部设备的开放请求(SYN)，以太网模块就开始开放处理。

正常完成时

- 开放完成信号(地址：5000_H … b0) : 开
- OPEN 指令完成软元件 : 开
- OPEN 指令完成软元件+1 : 关
- OPEN 指令完成状态区(*2) : 0000_H

禁止数据通讯。

异常完成时

- 开放完成信号 : 关
- OPEN 指令完成软元件 : 开
- OPEN 指令完成软元件+1 : 开
- 开放出错代码存储在缓冲存储器中。
- OPEN 指令完成状态区(*2) : 除 0000_H 之外的值
- 开放异常检测信号(X18) : 开

- 4) 一旦接收到外部设备的关闭请求(FIN)，以太网模块就开始关闭处理。
当完成关闭处理时，开放完成信号关闭，禁止数据通讯。
- 5) 使用专用 CLOSE 指令开始关闭处理。
(开放请求信号：关)

内部处理的正常完成时

- 开放完成信号 : 关
- CLOSE 指令完成软元件 : 开
- CLOSE 指令完成软元件+1 : 关
- CLOSE 指令完成状态区(*2) : 0000_H

内部处理的异常完成时

- 开放完成信号 : 关
- CLOSE 指令完成软元件 : 开
- CLOSE 指令完成软元件+1 : 开
- CLOSE 指令完成状态区(*2) : 除 0000_H 之外的值

要点	该范例使用 1 号连接进行说明。其它编号的连接使用相应的信号和位。
-----------	-----------------------------------

- *1 在正常完成初始化处理后和以太网模块处于开放允许状态前接收的开放请求 (SYN) 会发生状态错误，以太网模块发出强制关闭命令 (RST)。
- *2 完成时的结束代码存储在专用指令完成状态区。关于专用指令的详情，请参阅第 10 章“专用指令”。

备注

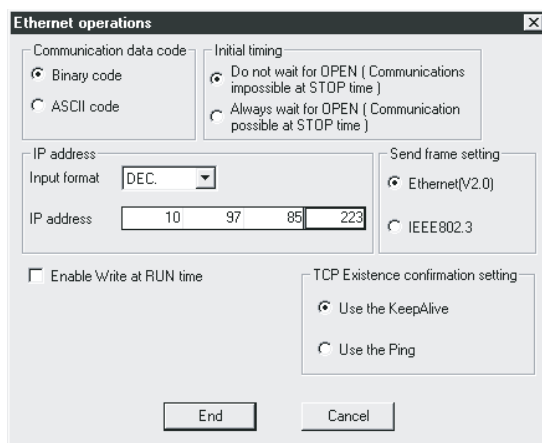
- (1) 如果连接的设置需要修改，则应在执行专用 OPEN 指令之前进行修改。
- (2) 一旦执行开放处理，就不能在开放处理完成之前取消开放请求。
开放处理完成后执行关闭处理 (CLOSE 指令)。

程序范例

本例说明当在开放系统设置中选择非被动开放时，开放处理/关闭处理用的程序。

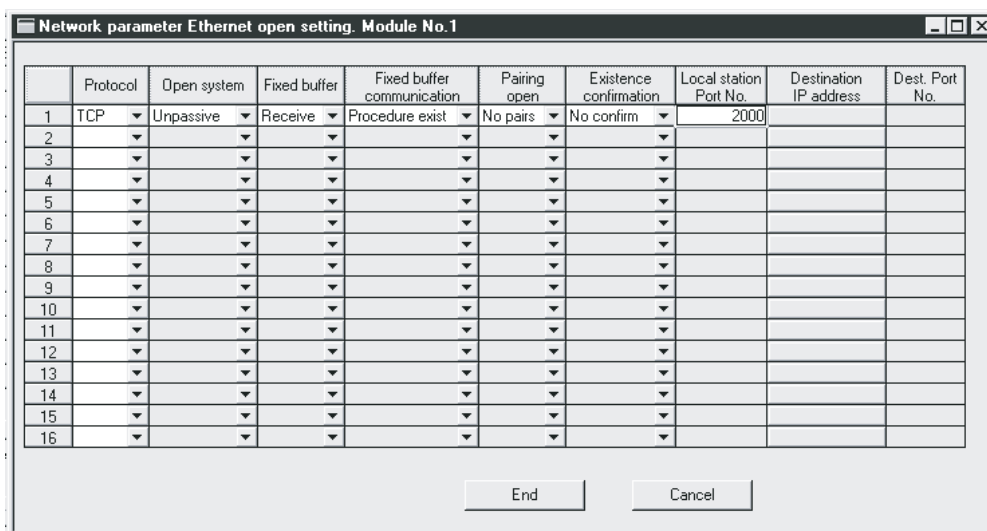
(1) 程序范例的执行环境

- (a) 以太网模块安装在基板的“0”槽中。
- (b) 假定已使用 GX Developer 设置了[设置 MNET/10H 以太网卡号的网络参数]设置，如下所示：
- 网络类型 : 以太网
 - 起始 I/O 地址 : 0000
 - 网络号 : 1
 - 组号 : 1
 - 站号 : 2
- (c) 假定已使用 GX Developer 设置了[操作设置]参数，如下所示




本站 IP 地址 : 0A.61.55.DF_H(10.97.85.223)

- (d) 假定已使用 GX Developer 设置了[开放设置]，如下所示：



本站端口编号 : 2000_H

- (e) 程序中使用下列触点信号：
- | | |
|-------------------|---------------------|
| 1号连接开放完成信号 | : M0 |
| 1号连接开放请求信号 | : M20 |
| 1号连接 OPEN 指令控制数据 | : 存储在 D100 至 D109 中 |
| 1号连接 CLOSE 指令控制数据 | : 存储在 D200 和 D201 中 |
- (f) 当未设置[开放设置]以太网模块参数用于 GX Developer 时，应该使用程序范例中  封住的区域。
当[开放设置]参数用于 GX Developer 时，不需要这部分程序。
- (g) 关于专用 OPEN 指令的详细说明，请参阅第 10 章“专用指令”。

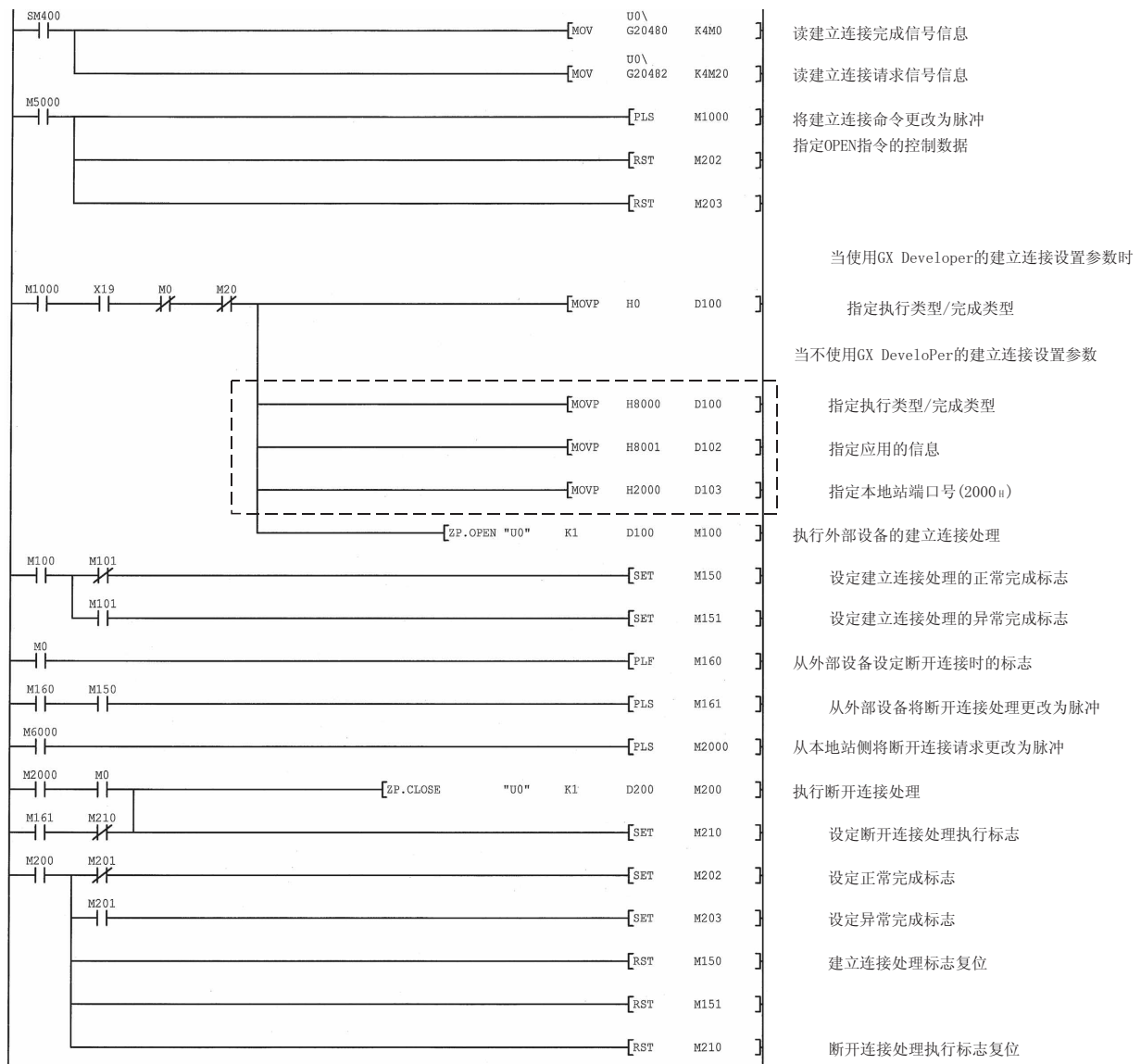
(2) 程序范例的概述

- (a) 使用 GX Developer 设置每个参数，将它写入 PLC CPU 后，重新启动 PLC CPU 并确认初始化处理的完成。
- (b) 对以太网模块的 1 号连接执行开放处理。
开放处理完成后，1 号连接等待外部设备发出开放请求。
- (c) 按照发给以太网模块的关闭指令或从外部设备发出的关闭请求，对 1 号连接执行关闭处理。

备注

程序中表示的“U0\G20480”和“U0\G20482”代码指定缓冲存储器中的下列区：
 U0\G20480：开放完成信号存储区(地址：5000_H(20480))
 U0\G20482：开放请求信号存储区(地址：5002_H(20482))

X19: 初始化正常完成信号
 M0: 1号连接建立完成信号
 M20: 1号连接建立请求信号
 1号连接的建立连接处理
 (TCP/IP非被动建立连接)



5.6.3 UDP/IP 开放处理/关闭处理

以下说明的是 UDP/IP 开放处理。

以 UDP/IP 进行开放处理/关闭处理的操作将根据用 GX Developer 在[操作设置]-[初始设置]中是选择“始终等待打开”或是选择“不等待打开”而有所不同，如下所述：

(1) 当在操作设置中选择“始终等待打开”时。(在停止时间可以进行通讯)

按照使用 GX Developer 进行的[开放设置]，在重新启动安装了以太网模块的站之后，自动建立选择 UDP/IP 通讯的连接，并且允许数据传送/接收。

不需要用于开放处理和关闭处理的顺控程序。

关于[开放设置]参数的详细说明，请参阅第 5.5 节“开放设置”。

备注

即使在“操作设置”时选择“始终等待打开(在停止时间可以进行通讯)”，如果以太网模块发出用于开放处理的专用 OPEN 指令，用 CLOSE 指令开放处理和关闭处理，但仍需要在使用顺控程序的有关连接之后，执行所有开放处理和关闭处理。

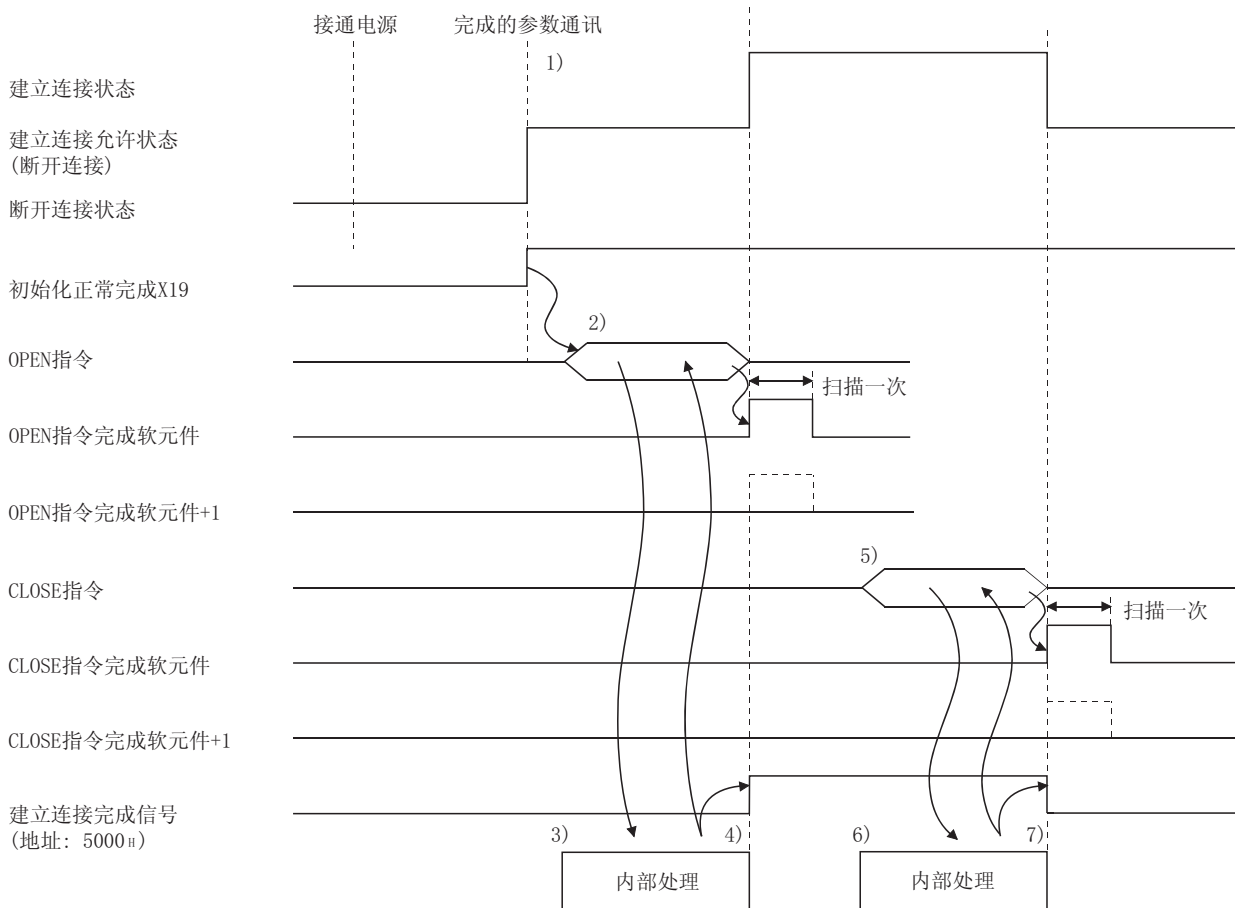
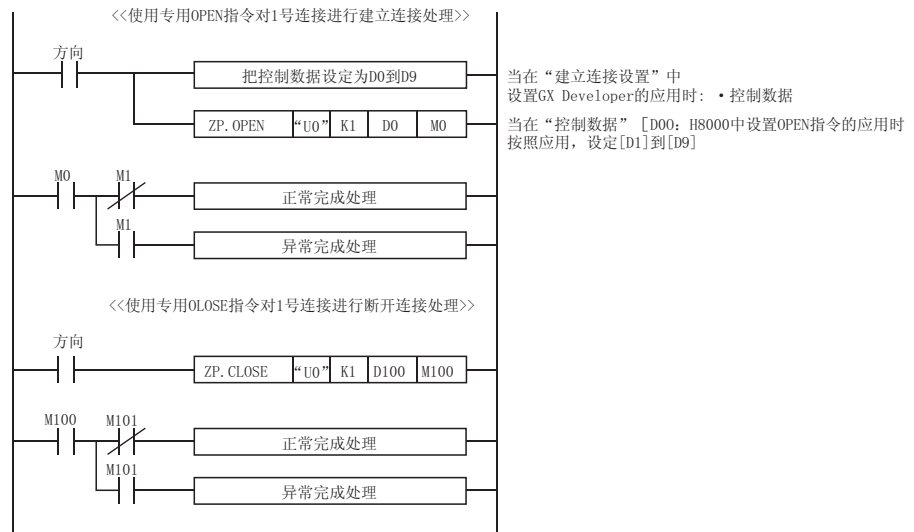
如果选择“不等待打开(在停止时间不能进行通讯)”，需要以与连接相同的方式执行开放处理和关闭处理。

(2) 当在操作设置中选择“不等待打开”时。(在停止时间不能进行通讯)

在该例中，需要下一页中显示的使用顺控程序来进行开放处理和关闭处理。开放处理正常完成后，允许数据传送和接收。

使用应用专用指令执行开放处理和关闭处理。

关于详细说明，请参阅第 10 章“专用指令”。



- 1) 通讯参数设置之后，确认以太网模块初始化处理的正常完成。
(初始化正常完成信号(X19)：开)
- 2) 使用专用 OPEN 指令开始开放处理。
(开放请求信号(地址：5002_H … b0)：开)
- 3) 以太网模块执行开放处理。(仅内部处理)
- 4) 当开放处理正常完成时
 - 开放完成信号(地址：5000_H … b0) : 开
 - OPEN 指令完成软元件 : 开
 - OPEN 指令完成软元件+1 : 关
 - OPEN 指令完成状态区(*1) : 0000_H
 允许数据通讯。
 当异常完成开放处理时
 - 开放完成信号 : 关
 - OPEN 指令完成软元件 : 开
 - OPEN 指令完成软元件+1 : 开
 - 开放出错代码存储在缓冲存储器中。
 - OPEN 指令完成状态区(*1) : 除 0000_H 之外的值
 - 开放异常检测信号(X18) : 开
- 5) 使用专用 CLOSE 指令开始关闭处理。
(开放请求信号：关)
- 6) 以太网模块执行关闭处理。(仅内部处理)
- 7) 当关闭处理正常完成时
 - 开放完成信号 : 关
 - CLOSE 指令完成软元件 : 开
 - CLOSE 指令完成软元件+1 : 关
 - CLOSE 指令完成状态区(*1) : 0000_H
 当关闭处理异常完成时
 - 开放完成信号 : 关
 - CLOSE 指令完成软元件 : 开
 - CLOSE 指令完成软元件+1 : 开
 - CLOSE 指令完成状态区(*1) : 除 0000_H 之外的值

*1 完成时的结束代码存储在专用指令完成状态区。关于专用指令的详情，请参阅第 10 章“专用指令”。

5.7 成对开放

下面说明的是通过以太网模块使用成对开放方法进行通讯。

5.7.1 成对开放

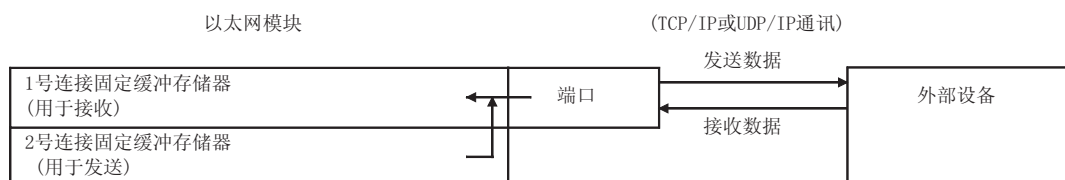
成对开放是开放的一种方法，它是将以太网模块的固定缓冲存储器通讯(有顺序和无顺序均允许)的接收连接和发送连接组合成对，建立一种使本地站和外部设备都用单个端口连接的方法。

通过指定成对开放方法，只在一个端口处执行开放处理就可以实现与两个连接进行数据通讯。

也可以使用已建立的成对连接来执行使用 MC 协议和随机访问缓冲存储器的通讯。

下面说明了执行成对开放的开放/关闭处理的步骤。

[范例]



要点

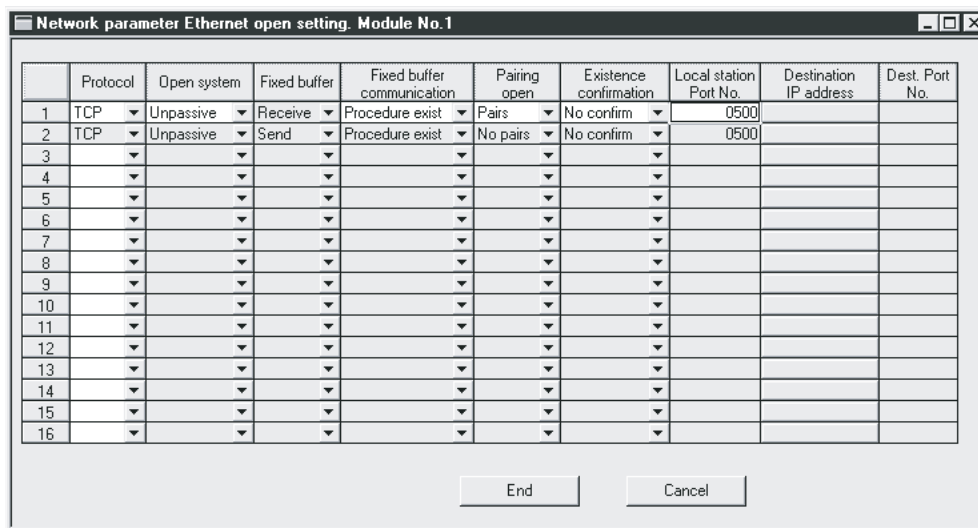
- (1) 当设置成对开放方法时，应用连接号(仅用于接收)的固定缓冲存储器和紧接着的连接号(仅用于发送)的固定缓冲存储器是成对的。
关于应用连接(仅用于接收)，在 1 号到 7 号和 9 号到 15 号连接范围中指定。
- (2) 使用成对开放方法通讯的外部设备范围的限定是，与以太网模块连接的设备(在以太网中)以及与具有中继功能的路由器连接的设备(请参阅第 5.3 节“路由中继参数”)。
- (3) 通过对设置了成对开放方法的应用连接(仅用于接收)进行开放/关闭处理，将自动执行下一次连接(仅用于发送)的开放/关闭处理。

5.7.2 使用 GX Developer 成对开放设置的范例

本节说明了为了按成对开放方法进行通讯，使用 GX Developer 进行设置的情况，下面给出了一个例子。

下面的画面表示的是在下列条件下设置的例子。

- 使用 1 号和 2 号连接。
- 以太网模块端口号是 0500H。
- 使用非被动开放系统。



(1) 协议

“TCP/IP”和“UDP/IP”都允许使用。

(2) 开放系统

所有开放系统：“主动”、“非被动”和“完全被动”，都可以进行设置。

(3) 固定缓冲存储器

在成对开放中，应用连接编号和后续的连接编号是成对的。把应用连接编号设置为“接收”，下一个连接编号设置为“发送”。

(4) 固定缓冲存储器通讯

“有顺序”和“无顺序”都可以选择。

(5) 成对开放

把接收连接设置为“成对”，发送连接设置为“不成对”。

(6) 存在确认

如果要执行存在确认，则把接收连接设置为“确认”，发送连接设置为“不确认”。

如果不要执行存在确认，则两者都选“不确认”。

- (7) 本地站端口编号
对于只用于接收的接收连接，设置该项。(发送连接不需要设置。)
咨询网络管理员后，设置端口号。

- (8) 目标 IP 地址
 - (a) 如果选择非被动系统
不需要设置。
 - (b) 如果选择主动或完全被动系统
需要设置。
咨询网络管理员后，只设置接收连接。

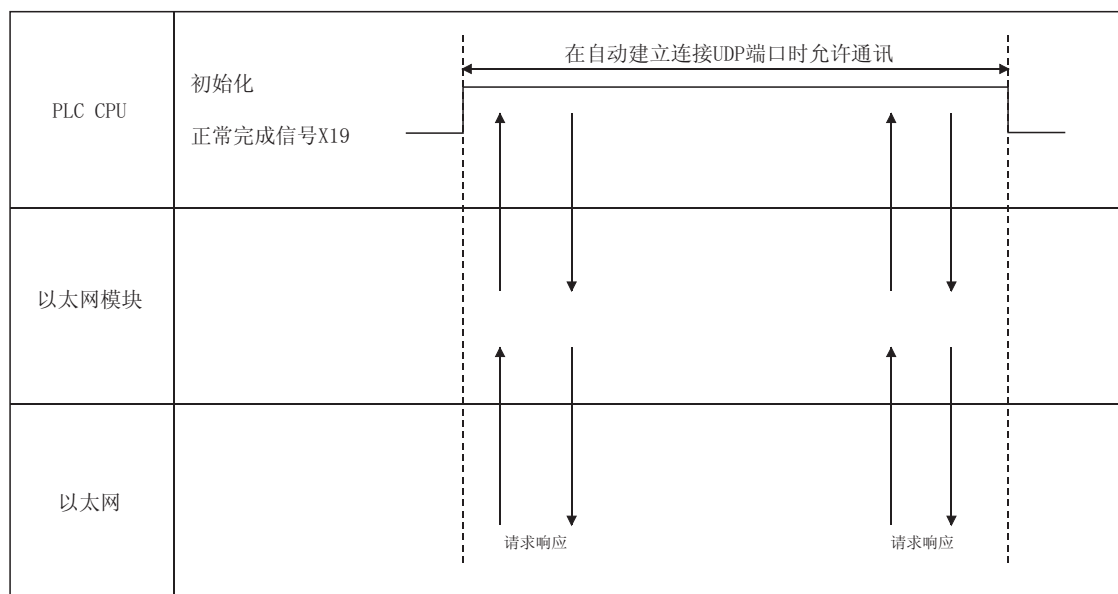
- (9) 目标端口号
 - (a) 如果选择非被动系统
不需要设置。
 - (b) 如果选择主动或完全被动系统。
需要设置。
咨询网络管理员后，只设置接收连接。

5.8 自动开放 UDP 端口

通常，当与外部设备进行通讯时，依照对外部设备的安排，有必要在开始数据通讯时与外部设备建立通讯线路；终止数据通讯时断开与外部设备的通讯线路。

在下列条件下，自动开放的 UDP 端口自动开放/关闭。

无需编程且不管 1 号到 16 号连接的开放状况，初始化处理完成之后，端口都处于允许通讯状态，并可以执行通讯。



(1) 自动开放的 UDP 端口开放和关闭的时序

(a) 开放时机

以太网模块完成初始化处理，并连接了通讯线路之后，按照用户注册的参数，端口自动开放。

(b) 关闭时机

当安装了以太网模块的站复位/断电时，关闭时序再次自动关闭。

(2) 使用自动开放的 UDP 端口适应数据通讯的功能

(a) 从外部设备通讯

1) 当使用 MC 协议 (QnA 兼容 3E 帧命令或 4E 帧命令) 进行通讯时，从 PLC CPU 读取数据/将数据写入 PLC CPU (端口号由用户指定；默认：1388H (5000))。

2) 不管使用 GX Developer 设置的通讯数据代码如何 (请参阅第 4.7 节)，均以二进制代码进行通讯。

(b) 从安装了以太网模块的站通讯

在该例中，可以使用数据链接指令进行通讯 (使用以太网模块操作系统的端口号)。

要点
(1) 初始化处理正常完成之后，以太网模块允许通过自动开放 UDP 端口号进行通讯，并等待发给本站以太网模块的通讯请求。(自动开放)
(2) 不管请求来自何处，只要它们是定址给以太网模块本身的，以太网模块都会应答并进行处理。
(3) 当肯定应答了通讯请求时，就占用了应用端口号，直到处理完成为止。即使在此期间，肯定应答了下一次通讯请求，通讯处理也必须等待上一个处理完成。
(4) 使用 MELSECNET/H、MELSECNET/10 中继通讯功能，自动开放的 UDP 端口号可以在以太网模块之间的通讯中。
(5) 必须进行重新初始化处理来更改自动开放 UDP 端口的端口号。(关于重新初始化处理，请参阅 5.2.3 节)

(3) 可访问范围

可访问范围视数据通讯功能的不同而不同。

数据通讯功能	可访问范围	参考章节
使用 MC 协议进行通讯(QnA 兼容 3E 帧或 4E 帧)	<ul style="list-style-type: none"> • 安装了以太网模块的站的 QCPU • 包含安装了以太网模块的网络系统中的 PLC CPU。 	-
数据链接指令	<ul style="list-style-type: none"> • 连接了以太网模块的以太网中的 PLC CPU。 • 通过路由器连接的以太网中的 PLC CPU。 • 包含安装了以太网模块的网络系统中的 PLC CPU。 	应用篇的 第 4 章

(4) 每次通讯的最大数据量

每次通讯的最大数据量视数据通讯功能的不同而不同。

数据通讯功能	每次通讯的最大数据量	参考章节
使用 MC 协议进行通讯(QnA 兼容 3E 帧或 4E 帧)	可以用 QnA 兼容 3E 帧命令指定的数据量。	第 6 章
数据链接指令	可以用数据链接命令指定的数据量。	应用篇的 第 4 章

5.9 使用 QCPU 远程口令功能进行通讯

远程口令功能是 QCPU 功能之一，它的作用在于防止用户在远地对 QCPU 作不适当的访问。

在 QCPU 中设置远程口令时，可以使用远程口令功能。

本节说明的是关于 QCPU 远程口令功能方面的以太网模块数据通讯。

远程口令功能是添加到 QCPU 中，防止来自外部设备的不适当访问(如毁坏程序或数据)的一种功能。但是，该功能不能防止不适当访问。

当需要保持 PLC 系统安全、防止来自外部设备不正确访问时，用户应结合采取自己的安全措施。

对于不正确访问导致系统故障的任何问题，公司概不负责。

* 防止不正确访问的措施例子：

- 建立防火墙。
- 把个人计算机设置成中继站，并使用应用程序控制发送/接收数据的中继。
- 把访问可控制外部设备设置成中继站。

关于访问可控制外部设备的情况，请向网络连接销售商或设备销售商咨询。

(2) 允许/禁止从外部设备访问 PLC 的处理

1) 访问允许处理 (解锁处理)

- 为了访问指定的 QCPU，外部设备执行直接连接站 (本站) 的以太网模块 (*) 的远程口令解锁处理。
 - 如果还没有执行解锁处理，禁止已接收通讯请求的以太网模块 (*) 通过远程口令核对访问特定站。(请参阅(3)。)
 - 解锁处理之前接收的所有数据将作为错误来处理。
- * 设置了远程口令的 QCPU 站的以太网模块。

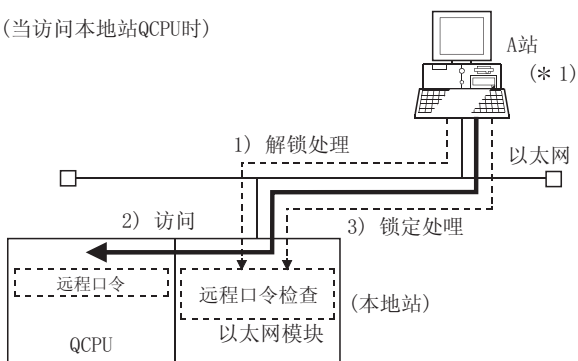
2) 访问处理

- 远程口令解锁的正常完成将允许访问特定站。
- 执行任意访问。

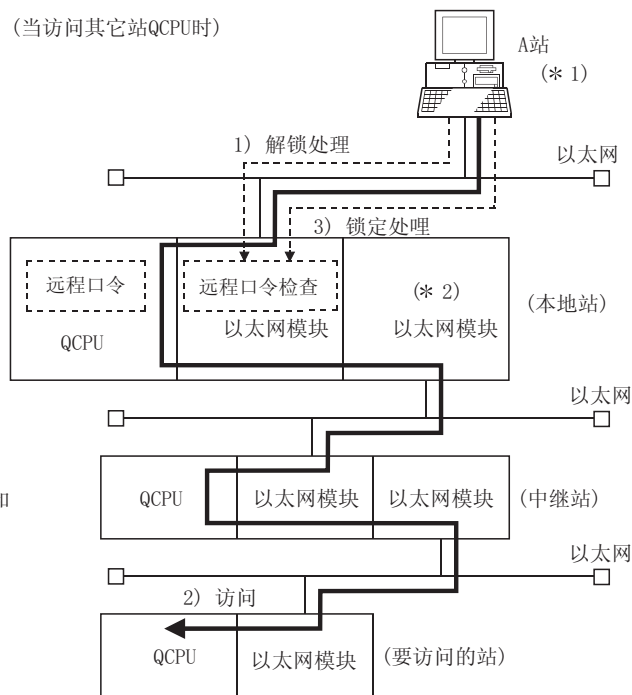
3) 访问禁止处理 (锁定处理)

- 当完成了对特定站的访问时，从外部设备执行远程口令锁定处理，禁止任何进一步的访问。

(当访问本站 QCPU 时)



(当访问其它站 QCPU 时)



*1 可以对本站，进行远程口令解锁处理和锁定处理。

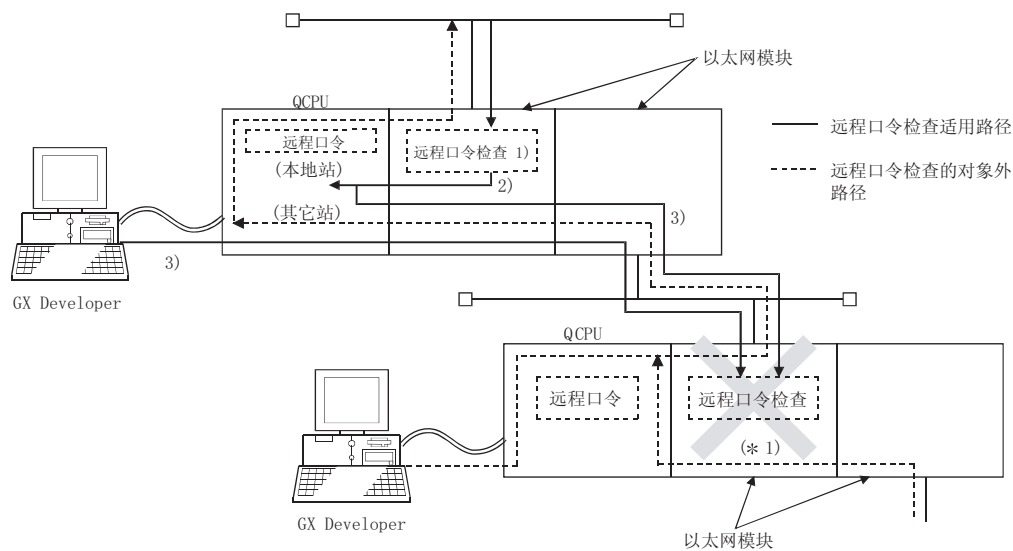
*2 以太网模块将通讯请求发送到其它以太网时，不需要解锁处理和锁。

要点	
	<p>(1) 远程口令解锁处理和锁定处理只是针对于直接连接到外部设备的本地站中的以太网模块进行的。 远程口令解锁处理和锁定处理不能针对任何其它站中的以太网模块处理(中继站和要访问的其它站)</p> <p>(2) 使用 MC 协议通讯的专用指令从外部设备执行远程口令解锁处理和锁定处理。 (当使用文件传送(FTP 服务器)功能时,用专用的 FTP 命令来执行。当使用 Web 功能或 GX Developer 时,用对话框输入远程口令。)</p>

(3) 以太网模块执行的远程口令核对

(a) 远程口令核对执行中的通讯

- 1) 为安装在 QCPU 站上的以太网模块设置下列参数时,以太网模块对通讯请求执行远程口令核对,如下:
 - 当在 QCPU 中设置远程口令时。
 - 当将正与外部设备通讯数据的连接设置为远程口令核对的目标时。
- 2) 本地站/其它站接收到外部设备通讯请求时,以太网模块执行远程口令核对。
- 3) 以太网模块执行关于下列传送请求的传送,而不执行远程口令核对:
 - 从本地站 QCPU 发出的传送请求(诸如使用固定缓冲存储器进行传送)。
 - 按照从 QCPU 发出的请求,将从外部设备(包括连接到本地站 QCPU 的 GX Developer)发出取得通讯请求发送到其它站。出的通讯请求发送到其它站。

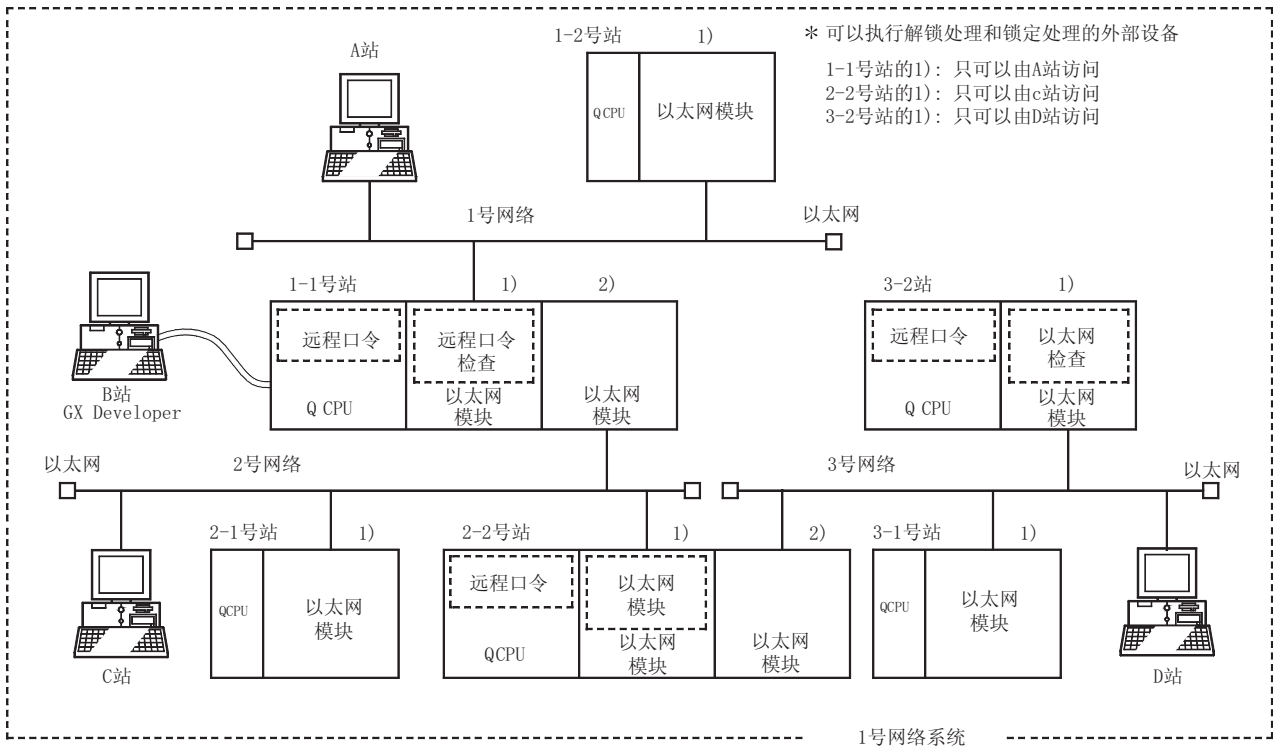


*1 上图中，由于设置了远程口令核对，故不能接收从外部设备发出的通讯请求。

如果没有设置远程口令核对，可以接收通讯请求，并且可以进行从外部设备发出的数据通讯。

- (b) 选择执行远程口令核对的连接
 用户可以选择以太网模块执行远程口令核对的任何连接，并使用 QCPU 参数来设置。(在 GX Developer 的“设置远程口令”画面上设置。)
- 1) 用户连接(1-16 号连接)
 - 2) 系统连接(如 GX Developer 通讯端口)
- (c) 当执行远程口令核对时可以访问的站
 表示当在 QCPU 中和在可以执行远程口令解锁处理和锁定处理的 QCPU 站中设置远程口令时，可以从外部设备访问的站。

(例 2)
 当在 PLC 系统中的多 QCPU 站中设置远程口令和远程口令核对应



(请求目标)		可使用的 PLC 站					
		1-1 号站 QCPU	1-2 号站 QCPU	2-1 号站 QCPU	2-2 号站 QCPU	3-1 号站 QCPU	3-2 号站 QCPU
外部设备 (* ³)	A 站	●	○	●	×	×	×
	B 站	○	○	○	×	×	×
	C 站	○	○	○	●	●	×
	D 站	○	○	○	○	○	●

- : 远程口令解锁处理后可以由外部设备访问的站
- : 不执行远程口令解锁处理就可以由外部设备访问的站
- ×: 不能由外部设备访问的站

*3 完成了 1-1 号站模块 1) 的远程口令解锁处理后, A 站可以访问 ● 站。如果开放通讯线路, 可以访问 ○ 站。
 如果与 ○ 站的通讯线路开放, B 站就可以访问这些站。
 完成了 2-2 号站模块 1) 的远程口令解锁处理后, C 站可以访问 ● 站。如果建立通讯线路连接, C 站可以访问 ○ 站。
 完成了 3-2 号站模块 1) 的远程口令解锁处理后, D 站可以访问 ● 站。如果开放通讯线路, D 站可以访问 ○ 站。

要点

为了禁止用以太网模块 MELSECNET/H、MELSECNET/10 中继通讯功能从外部设备访问其它的站，应把检查标记设在下列中继站和要访问站的远程口令设置项处：

“GX Developer 通讯端口 (UDP/IP) (*), 专用指令, MELSECNET/H、
MELSECNET/10 中继通讯端口”

* 在 GX Developer 的远程口令设置屏上设置。

* 如果没有把检查标记设在以上设置项目处，则可以访问其它站。

5.9.2 使用远程口令核对功能时的注意事项

以下是使用以太网模块的远程口令核对功能时的注意事项。

- (1) 在 QCPU 中设置远程口令后，重新启动 QCPU(1 号 PLC 用于多 CPU 系统)
(使用 RESET/L. CLR 开关或电源复位按钮复位)。
通过重新启动 QCPU，远程口令生效。
- (2) 只对可以执行解锁处理和锁定处理的外部设备进行数据通讯中要使用的连接设置远程口令核对。
(例子) 当通过固定缓冲存储器进行通讯时，不应在接收从 MELSEC PLC CPU 传送来的数据连接中设置远程口令核对。
*检查标记不应放在第 5.9.5 节中所示的“远程口令具体设置”屏上的应用连接处。
- (3) 由于使用无顺序固定缓冲存储器进行通讯连接未执行远程口令核对，因此不应设置远程口令核对。
- (4) 当外部设备通过以太网模块访问其它站的 PLC 时，如果在中继站或要访问的站的 QCPU 中设置了远程口令，就可能不能够访问 PLC。(请参阅第 5.9.1 节(1)和(3))
- (5) 当执行 UDP/IP 通讯时的注意事项：
 - 1) 决定将进行通讯的外部设备。
不要与指定外的外部设备进行数据通讯。
 - 2) 使用以太网模块存在确认功能。
同样，当完成数据通讯时，必须执行远程口令锁定处理。
* 如果没有执行锁定处理，允许从其它设备进行数据通讯，直到以太网模块存在确认功能发生超时为止。
为此原因，在用 GX Developer 设置应用连接时，必须规定如下：
 - 当执行初始化设置时，把起动间隔定时器值和存在确认功能的间隔定时器值设置得尽可能小。
 - 对于开放设置，在“存在确认”项选择“确认”。
* 当把使用自动开放 UDP 端口进行数据通讯的连接设置为远程口令核对的目标时，自动执行存在确认。
- (6) 尽可能使用 TCP/IP 通讯，以便通过使用以太网连接用 GX Developer 进行通讯。

5.9.3 数据通讯步骤

本节说明当外部设备使用执行远程口令核对的连接进行数据通讯时的步骤。

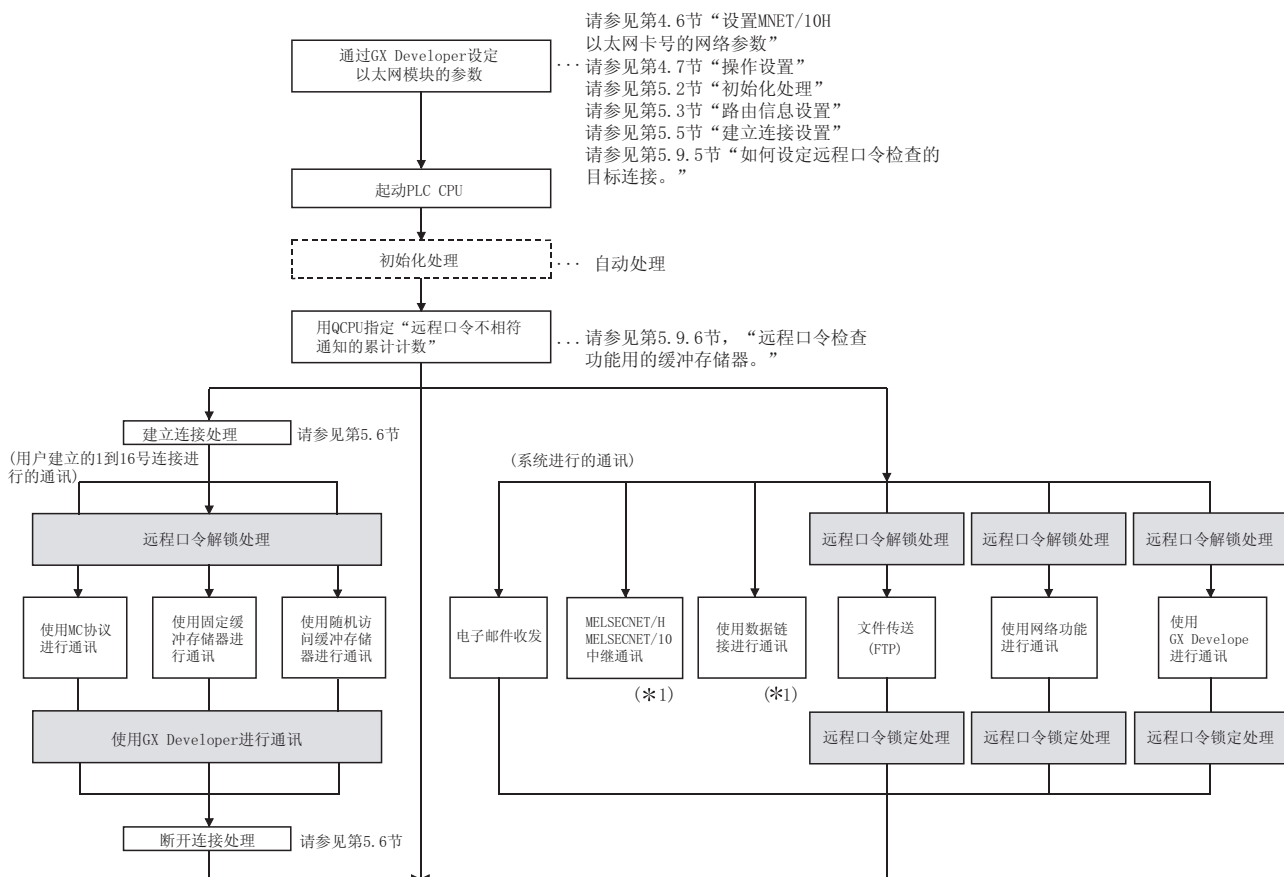
- (1) 通过 GX Developer 在“远程口令具体设置”中设置远程口令核对的目标连接并将参数写入 QCPU。(请参阅第 5.9.5 节)
- (2) 起动 QCPU 后, 按需要将设置值写入以太网模块的下列缓冲存储器中:(请参阅第 5.9.6 节(3))
远程口令不相符通知的累计计数指定区域
 • • • 地址 20592 至 20593 (5070_H 至 5071_H)
- (3) 开放处理完成后, 执行远程口令解锁(释放)处理。
解锁处理之前接收的所有数据将按错误来处理。(请参阅出错代码的故障排除一节)
- (4) 当从外部设备执行远程口令的解锁处理时, 如果用户指定的远程口令与本站 QCPU 中设置的远程口令相符, 则可以从外部设备访问指定站。
- (5) 完成数据通讯后, 在从外部设备执行远程口令锁定处理后关闭应用连接。
- (6) 使用 MC 协议通讯的专用指令, 在与以太网模块连接的外部设备中执行远程口令解锁处理和锁定处理。(当使用文件传送(FTP 服务器)功能时, 用专用 FTP 命令执行。当使用 Web 功能或 GX Developer 时, 用对话框输入远程口令。)

备注

当“GX Developer 通讯端口”设置为远程口令核对连接时, 在所示的 GX Developer 屏上执行解锁处理后, 当访问开始时, 访问 PLC。

- (7) 只可以对安装了以太网模块的本站 QCPU 执行远程口令解锁和锁定处理。不能对其它站的 QCPU 执行远程口令解锁和锁定处理。(请参阅第 5.9.1 节(2))

(数据通讯步骤)



*1 当访问其它站的 PLC 时，如果在安装了以太网模块的中继站和要访问站的 QCPU 中设置了远程口令，则可能不允许访问其它站。(请参阅第 5.9.1 节(1)和(3)。)

要点

- (1) 在 TCP/IP 通讯中，如果没有执行远程口令锁定处理就关闭连接，则在关闭连接时，以太网模块自动执行锁定处理。
- (2) 在 UDP/IP 通讯中，一定要在执行远程口令锁定处理后关闭连接。当连接关闭时，以太网模块不执行远程口令锁定处理。

5.9.4 远程口令解锁处理或锁定处理异常完成时

本节说明当远程口令解锁处理或锁定处理异常完成时用户要执行的步骤。

- (1) 在检查完 QCPU 中设置的远程口令之后，再次执行解锁/锁定处理。
- (2) 当解锁处理/锁定处理异常完成的发生次数大于缓冲存储器中设置的累计通知计数(*1)时，以太网模块出现下列情形：
 - (a) COM.ERR LED 亮。
 - (b) 代码 C200H 存储在缓冲存储器出错日志区的出错代码和结束代码存储区中(地址 227 至 372 (E3_H 至 174_H))。
*1 这是当以太网模块起动时，从 QCPU 设置在远程口令不相符通知累计计数指定区(地址 20592 (5070_H)、20593 (5071_H))中的计数。(计数是用 T0 指令等设置的)
- (3) 当以太网模块的 COM.ERR LED 亮时，监视以上缓冲存储器。
如果存储了出错代码 C200_H，监视缓冲存储器中异常完成累计计数存储区(地址 20595 (5073_H)、20597 (5075_H) ···)，并检查解锁处理或锁定处理中的哪一个连接异常完成。
- (4) 用户按需要执行下列事情：
 - (a) 断开应用连接。
 - (b) 将“0”写入缓冲存储器中的异常完成累计计数存储区。
* 如果通过写“0”没有清除累计计数，则每次在发生异常完成超过累计通知计数时执行上面(2)的过程。
 - (c) 如果发生的应用连接的解锁处理/锁定处理的异常完成次数大于以上累计的通知计数，这表明可能从外部设备进行了不适当访问。
禁止使用缓冲存储器的系统端口禁用指定区的应用连接(地址 20488 (5008_H))。(自此，不能对应用连接进行解锁处理，直到设置“允许使用”为止。)
 - (d) 通知系统管理员解锁处理/锁定处理异常完成发生的次数大于累计的通知计数，并采取适当措施。

备注

- (1) 在缓冲存储器中存储的以下累计计数可以在用户选项下清除。(从 QCPU 将“0”写入应用区。)
- 解锁处理正常完成累计计数的存储区: 地址 20594 (5072_H) . . .
 - 锁定处理正常完成累计计数的存储区: 地址 20596 (5074_H) . . .
- (2) 关于以太网模块用的 COM.ERR LED 亮了之后如何熄灭, 请参阅用户手册(基础篇)的第 11.1.2 节。

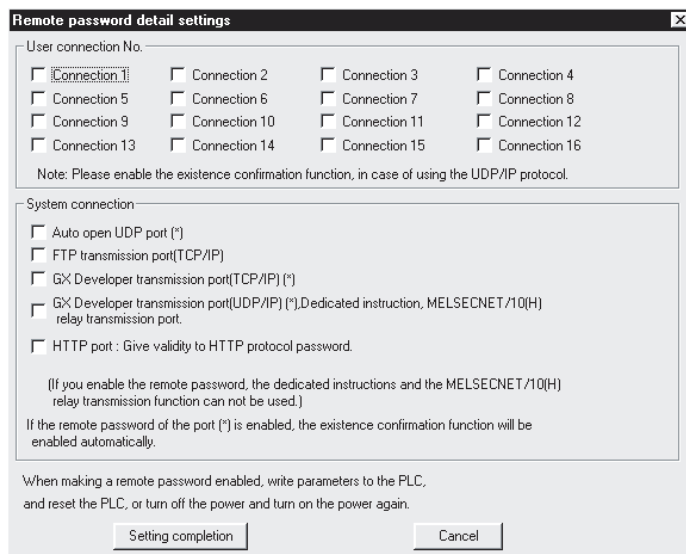
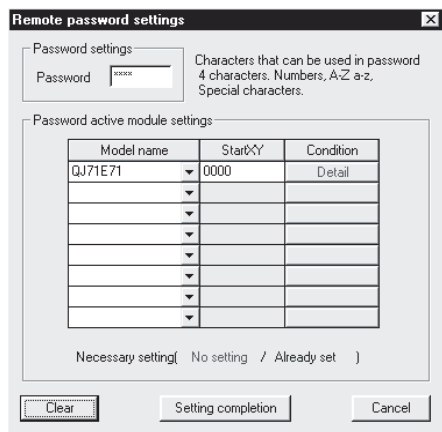
5.9.5 如何设置远程口令核对的目标连接

通过 GX Developer，用参数设置设置远程口令核对的连接。

[起动步骤]

“GX Developer” → **远程口令** → “远程口令设置”画面 → **详细** → “远程口令详细设置”画面

[设置画面]



[设置项目]

项目名称	项目设置说明	设置范围/选项
口令设置	输入要在 QCPU 中设置的口令 (*1)。	—
口令有效模块设置	型号名称	选择对在 QCPU 中设置的远程口令进行核对的模块型号名称。
	起始 XY	设置执行远程口令核对的起始地址。
	模块状况	(显示“远程口令具体设置”画面。)
用户连接号		具体设置
系统连接	自动开放 UDP 端口	设置要进行远程口令核对的连接。
	FTP 传送端口 (TCP/IP)	
	GX Developer 传送端口 (TCP/IP)	
	GX Developer 传送端口 (UDP/IP)	
	HTTP 端口	
		将一个检查标记置于目标连接处。

*1 设置远程口令时参考以下事项:

- 避免使用只有简单数字或字母的字符串。
- 数字、字母和特殊字符(?, !, %等)组合使用。
- 避免使用代表用户名或出生年月的字符串。

要点	
(1)	当在多 CPU 系统中使用以太网模块时，通过将远程口令写入以太网模块的控制 PLC 来设置远程口令。
(2)	在 QCPU 中设置远程口令后，重新启动 QCPU(用于多 CPU 系统的第一个 QCPU 模块) (使用 RESET/L. CLR 开关或电源复位进行复位)。通过重新启动 QCPU，远程口令生效。
(3)	功能版本 AQCPU 支持的口令，用于禁止通过 GX Developer 在 QCPU 中读/写的文件数据。 使用本节中说明的远程口令和文件访问口令可以实现双重访问控制。

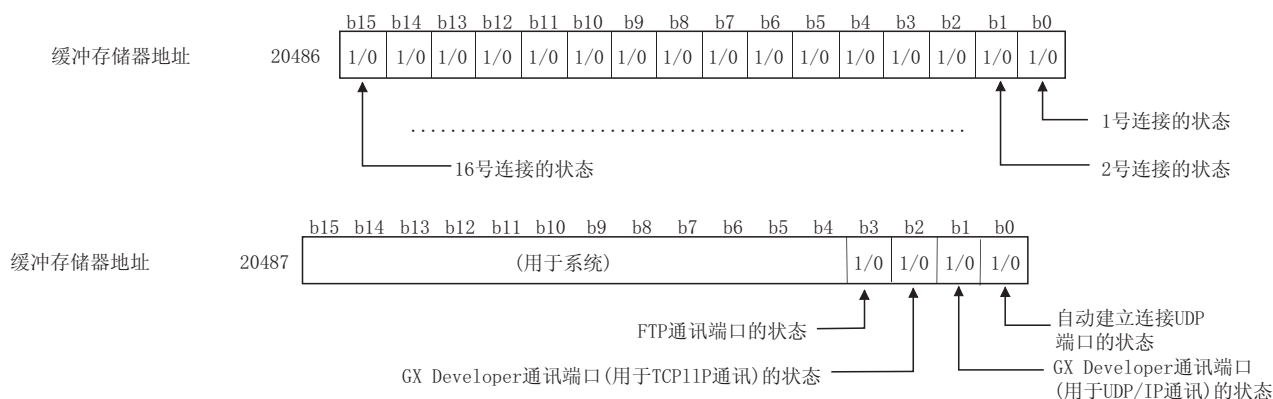
5.9.6 远程口令核对功能所用的缓冲存储器

本节说明了远程口令核对功能所用的缓冲存储器。

地址	远程口令核对功能所用的缓冲存储器		应用	
			设置	监视
20486 (5006H)	远程口令状态(用户端口)			○
20487 (5007H)	远程口令状态(系统端口)			○
20488 (5008H)	系统端口禁用指定		○	
20592 (5070H)	远程口令与通知累计的计数指定不匹配		○	
20593 (5071H)	远程口令与通知累计的计数指定不匹配		○	
20594 (5072H)	1号连接	解锁过程正常完成的累计计数		○
20595 (5073H)		解锁过程异常完成的累计计数		○
20596 (5074H)		锁定过程正常完成的累计计数		○
20597 (5075H)		锁定过程异常完成的累计计数		○
20598 (5076H)		以关闭为根据的锁定过程的累计计数		○
:	:	:		
20669 (50BDH)	16号连接	解锁过程正常完成的累计计数		○
20670 (50BEH)		解锁过程异常完成的累计计数		○
20671 (50BFH)		锁定过程正常完成的累计计数		○
20672 (50C0H)		锁定过程异常完成的累计计数		○
20673 (50C1H)		基于关闭基础上的锁定过程的累计计数		○
20674 (50C2H)	自动开放UDP端口的连接	解锁过程正常完成的累计计数		○
20675 (50C3H)		解锁过程异常完成的累计计数		○
20676 (50C4H)		锁定过程正常完成的累计计数		○
20677 (50C5H)		锁定过程异常完成的累计计数		○
20678 (50C6H)		以关闭为根据的锁定过程的累计计数		○
20679 (50C7H)	GX Developer通讯用的连接(用于UDP/IP通讯)	解锁过程正常完成的累计计数		○
20680 (50C8H)		解锁过程异常完成的累计计数		○
20681 (50C9H)		锁定过程正常完成的累计计数		○
20682 (50CAH)		锁定过程异常完成的累计计数		○
20683 (50CBH)		以关闭为根据的锁定过程的累计计数		○
20684 (50CCH)	GX Developer通讯用的连接(用于TCP/IP通讯)	解锁过程正常完成的累计计数		○
20685 (50CDH)		解锁过程异常完成的累计计数		○
20686 (50CEH)		锁定过程正常完成的累计计数		○
20687 (50CFH)		锁定过程异常完成的累计计数		○
20688 (50D0H)		以关闭为根据的锁定过程的累计计数		○
20689 (50D1H)	FTP 通讯用的连接	解锁过程正常完成的累计计数		○
20690 (50D2H)		解锁过程异常完成的累计计数		○
20691 (50D3H)		锁定过程正常完成的累计计数		○
20692 (50D4H)		锁定过程异常完成的累计计数		○
20693 (50D5H)		以关闭为根据的锁定过程的累计计数		○

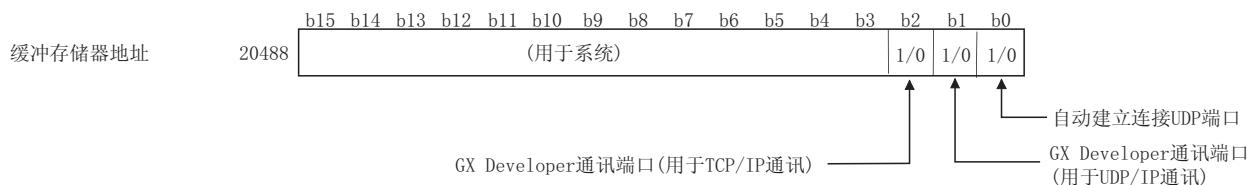
(1) 远程口令状态存储区(地址 20486 至 20487(5006H 至 5007H))

- (a) 存储每个连接的远程口令的当前解锁/锁定状态。
使用用户端口的 1 号到 16 号连接的解锁/锁定状态存储在地址 20486 中。
使用系统端口所连接的解锁/锁定状态存储在地址 20487 中。
- (b) 每个连接的远程口令状态都是按下面所示以位表示的：
1(开)：锁定状态
0(关)：解锁状态/无远程口令核对设置



(2) 系统端口禁用指定区(地址 20488(5008H))

- (a) 允许/禁止对方的软元件使用以太网模块系统所提供的端口进行下列数据通讯：
• 使用自动开放 UDP 端口进行通讯(使用 MC 协议进行通讯)
• 使用以太网连接从 GX Developer 进行通讯
- (b) 该指定由用户确定。
1(开)：禁用
0(关)：允许使用(默认)



- (c) 如果执行远程口令核对，且使用应用端口的连接的远程口令状态处于锁定状态，则可以禁止使用应用端口。

- (d) 如果在上面的数据通讯期间，已从对方的软件处执行了不正确的访问，则把应用端口设置为“禁用”。(详细说明请参阅第 5.9.4 节)

备注

为了禁止对方的软件使用以太网模块 FTP 功能进行数据通讯，通过 GX Developer 在“设置以太网 FTP 参数”下的 FTP 功能设置项目中设置“不使用”。(请参阅用户手册(应用篇)的第 5.3 节)

(3) 远程口令与通知累计的计数指定区(地址 20592 至 20593(5070H 至 5071H)) 不匹配

- (a) 以太网模块起动之后，从用户/外部设备进行解锁/锁定处理期间，远程口令不相符的发生数达到或超过允许值时，用 0 到 FFFFH 指定对用户进行定时通知的计数(该指定对于所有连接都是共同的。)

0H: 无指定

(当发生远程口令不相符时，不执行过程(c)。)

1 至 FFFFH: 通知累计的计数

- (b) 指定该区下列连接中每一个累计的通知计数:

地址 20592 区: 1 号到 16 号连接

地址 20593 区: 使用自动开放的 UDP 端口连接

使用以太网连接用的 GX Developer 通讯

从使用 FTP 功能的外部设备进行通讯

- (c) 解锁/锁定处理期间发生的远程口令不相符的次数超过累计的通知计数时，以太网模块执行下列情况(详细说明请参阅第 5.9.4 节):

- COM.ERR LED 亮。

- 代码 C200H 存储在缓冲存储器出错日志区的出错代码和结束代码存储区中(地址 227 至 372(E3H 至 174H))。

* 以太网模块系统未关闭。

- (d) 可以使用后面(4)中指示的异常完成累计计数存储区来检查到目前为止发生的远程口令不相符次数的累计数(根据以太网模块累计的计数值)。

- (4) (地址 20594 (5072H) . . .) 解锁过程正常完成累计计数的存储区
(地址 20595 (5073H) . . .) 解锁过程异常完成的累计计数的存储区
(地址 20596 (5074H) . . .) 锁定过程正常完成的累计计数的存储区
(地址 20597 (5075H) . . .) 锁定过程异常完成的累计计数的存储区
 - (a) 存储到目前为止已正常/异常完成应用连接的远程口令解锁/锁定处理的次数的累计数。
 - (b) 用户应该清除存储在该累计计数存储区中的值。(从 QCPU 将“0”写入应用区。)

- (5) 以关闭为根据的锁定过程累计计数的存储区
(地址 20598 (5076H) . . .)
 - (a) 当用户不执行远程口令锁定处理就断开应用连接时，以太网模块自动执行锁定处理。
以太网模块自动执行锁定处理的次数累计数存储在该区域中。
 - (b) 用户应该清除存储在该累计计数存储区中的值。(从 QCPU 将“0”写入应用区。)

要点

可以存储在(4)和(5)中列出的区中的最高值是 FFFF _H 。不存储超过 FFFF _H (65535) 的累计计数。

5.9.7 设置远程口令核对时的数据通讯

下面说明的是使用设置了远程口令核对的连接所进行的数据通讯。

功能		数据通讯		备注
		无远程口令核对设置	远程口令核对设置 ⁽¹⁾	
使用 MC 协议进行通讯	用户开放端口	开放处理完成之后，可以进行通讯。	开放处理完成之后，允许在接收到解锁命令时进行通讯，直到接收到锁定命令为止。	请参阅解锁和锁定命令的参考手册的第 3.18 节。
	自动开放 UDP 端口	初始化处理完成之后，可以进行通讯。	初始化处理完成之后，允许在接收到解锁命令时进行通讯，直到接收到锁定命令为止。	
使用固定缓冲存储器进行通讯	有顺序	开放处理完成之后，可以进行通讯。	开放处理完成之后，允许在接收到解锁命令时进行通讯，直到接收到锁定命令为止。	
	无顺序		开放处理完成之后，可以进行通讯。(※2) * 由于在无顺序通讯中未执行远程口令核对，所以不用为使用的连接设置远程口令核对。	
使用随机访问缓冲存储器进行通讯	开放处理完成之后，允许在接收到解锁命令时进行通讯，直到接收到锁定命令为止。			
电子邮件的收发		初始化处理之后，允许传送和接收。 * 电子邮件收发不执行远程口令核对。		-
使用 Web 功能进行通讯		初始化处理完成之后，可以进行通讯。	输入远程口令后允许进行通讯。当网络浏览器关闭时，自动执行远程口令锁定处理。	
MELSECNET/H, MELSECNET/10 中继通讯		初始化处理完成之后，可以进行通讯。	初始化处理完成之后，可以进行通讯。(※3)	
使用数据链接指令进行通讯				
文件传送(FTP 服务器)功能		在以太网中，可以与完成开放处理的外部设备进行通讯。	开放处理完成之后，允许在接收解锁命令时开始通讯，直到接收到锁定命令。	请参阅用户手册(应用篇)的第 5 章。
GX Developer TCP 通讯(※ ¹)		初始化处理完成后，允许 GX	允许在输入远程口令后进行通讯。	请参阅 GX Developer 操作手册。
GX Developer UDP 通讯(※ ¹)		Developer 端通过开后进行通讯。	当目标关闭时，自动执行远程口令锁定处理。	

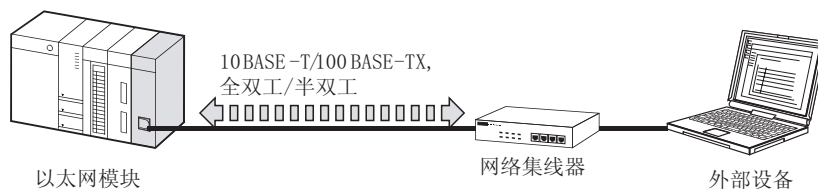
※1 关于访问其它站的 PLC 的情况，请参阅第 5.9.1 节(3)。

※2 无顺序固定缓冲存储器通讯使用专用连接。不要对应用连接执行远程口令设置。

※3 如果在安装了以太网模块的中继站和访问站中设置远程口令，当访问其它站 PLC 时，不能访问其它站。(请参阅第 5.9.1 节中的(1)和(3)。)

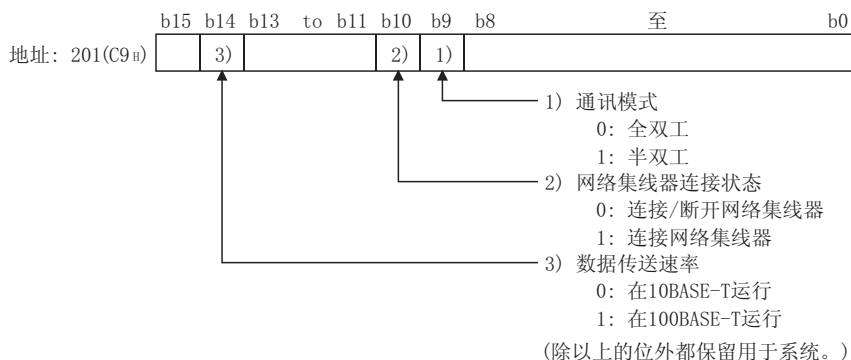
5.10 网络集线器连接状态监视功能

在以下缓冲存储器地址中可以检查当前以太网和网络集线器连接状态、传送速度和以太网模块检测断开的次数。(只用于 QJ71E71-100)



(1) 网络集线器连接状态区(地址: 201(C9H))

存储当前 QJ71E71-100 和网络集线器的连接状态和传送速率。



(2) 断开检测计数存储区(地址: 20995(5203H))

(a) 在完成初始化处理后存储断开检测次数。(X19 变为 0N)

在以下例子中检测断开。

- 以太网模块和网络集线器间的断开
- 从网络集线器端连接器上卸下电缆
- 网络集线器电源关闭
- 从以太网模块端连接器上卸下电缆

(b) 如果错误的发生次数超过 65536 次, 计数会在 FFFFH 时停止(655635)。使用顺控程序写入“0”到该区域以清除存储值。

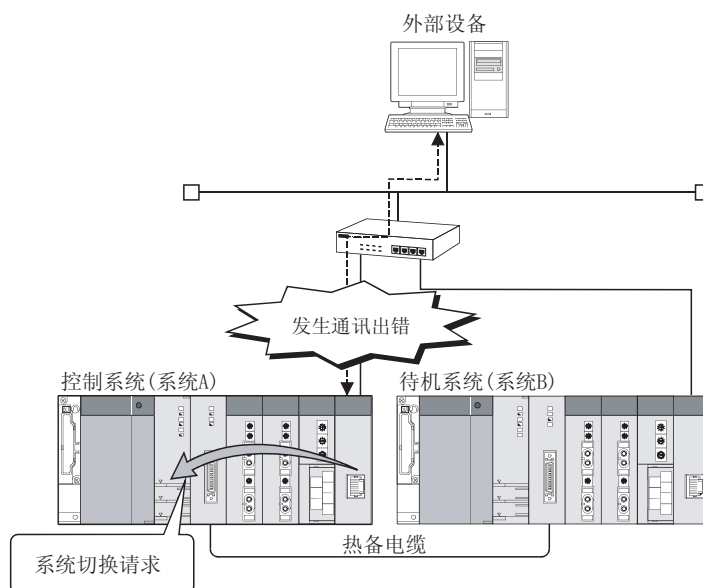
5.11 在冗余系统中配置网络(冗余系统支持功能)

本节说明冗余系统中以太网模块使用的功能和设置。关于系统配置请参阅 2.5 节。

5.11.1 发送系统切换请求到控制系统 CPU

(1) “发送系统切换请求到控制系统 CPU”

冗余系统中安装在控制系统 CPU 的主基板上的以太网模块检测到通讯错误或断开时，此功能向控制系统 CPU 发出系统切换请求。



(2) “系统切换请求发送条件”

当满足 GX Developer 冗余设置中设置的以下条件，以太网模块向控制系统 CPU 发送系统切换请求。(关于冗余设置，请参阅 5.11.3 节)

系统切换请求发送条件		说明
通讯错误检测	存在确认	连接开放后(开放处理后)，无法确认外部设备的存在。
	ULP 超时	TCP ULP 定时器值内没有外部设备发出的 ACK 响应。
断开检测		连接到以太网模块的电缆断开。(只用于 QJ71E7 1-100)

要点
<p>当满足本节(2)中指示发送条件的系统切换请求时，系统在控制系统和待机系统间切换。但是，在以下的任意情况下，如果从以太网模块中发出系统切换请求，系统不会切换。</p> <ul style="list-style-type: none"> • 待机系统已经处于出错状态。(比如:电源关闭, 复位或停止错误)(关于系统切换的原因, 请参阅 QnPRHCPU 用户手册(冗余系统)。) • 网络模块冗余组设置已经成为以太网模块模式, 并且任一站都运行正常。(关于“网络模块冗余组设置”, 请参阅 QnPRHCPU 用户手册(冗余系统))

(3) “通讯错误”时发送系统切换请求

安装在控制系统 CPU 主基板上的以太网模块在每个连接时监视与外部设备的通讯。当检测到通讯错误时, 发送系统切换请求到控制系统 CPU。

(a) “通讯错误时发送系统切换请求”

当发生以下任一通讯错误时, 发送系统切换请求。

通讯错误	说明
存在确认	连接开放后(开放处理后), 无法确认外部设备的存在。
ULP 超时	TCP ULP 定时器值内没有外部设备发出的 ACK 响应。

(b) “目标连接”

在 GX Developer 冗余设置中设置“发生通讯错误时的系统切换设置”的连接上检测通讯错误。(关于冗余设置, 请参阅 5.11.3 节)

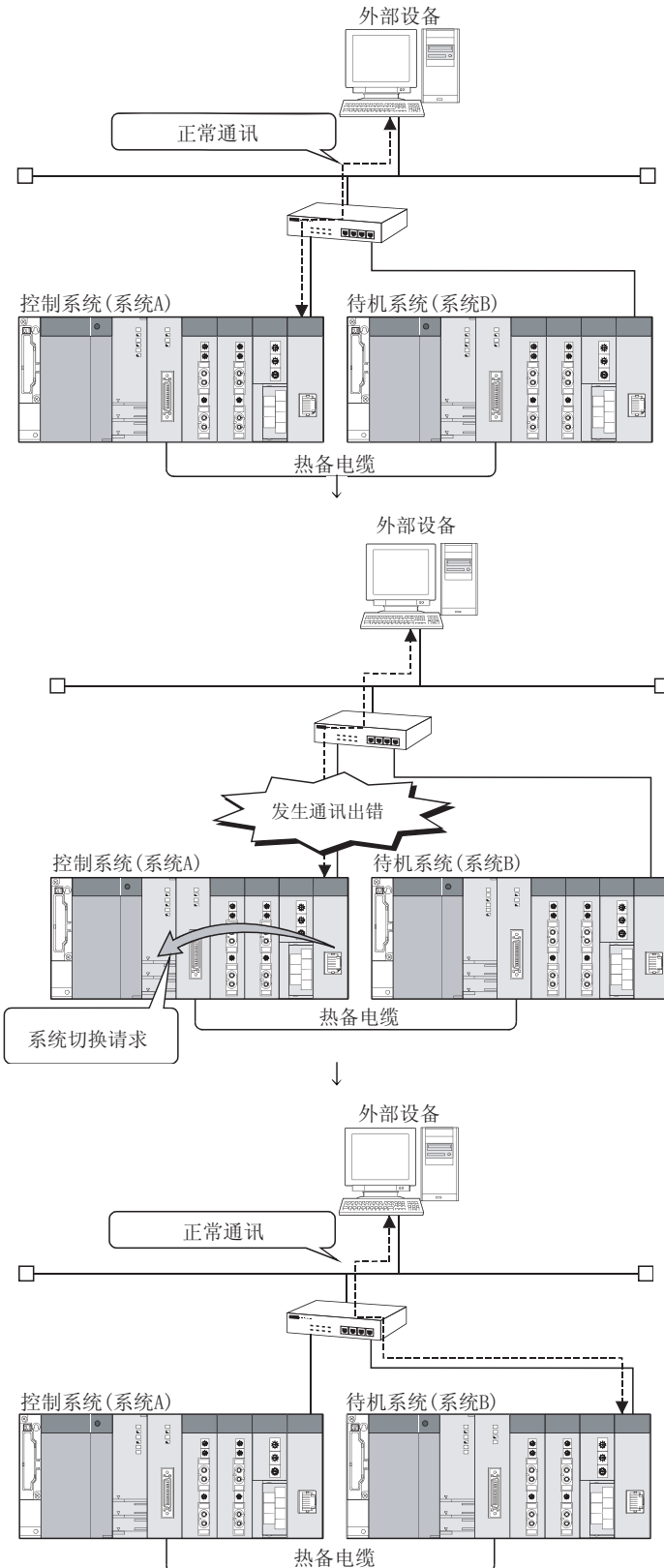
以下连接可成为监视目标。

- 连接号 1 到 16
- 自动开放 UDP 端口
- FTP 通讯端口
- GX Developer 通讯端口(TCP、UDP)
- HTTP 通讯端口

(c) “通讯错误”时的系统切换请求操作

以太网模块使用存在确认功能或 TCP ULP 定时器，检查与外部设备通讯的错误。

以下说明通讯错误时执行的系统切换请求操作。



1) 正常通讯时

系统 A 作为控制系统运行，系统 B 作为待机系统运行。

外部设备正在与安装在控制系统 CPU 主基板的以太网模块通讯。(*1)

2) 错误检测时(*2)

当安装在控制系统 CPU 主基板的以太网模块检测到外部设备和以太网模块间发生通讯错误时，会向控制系统 CPU 发送系统切换。(*3)

3) 系统切换后

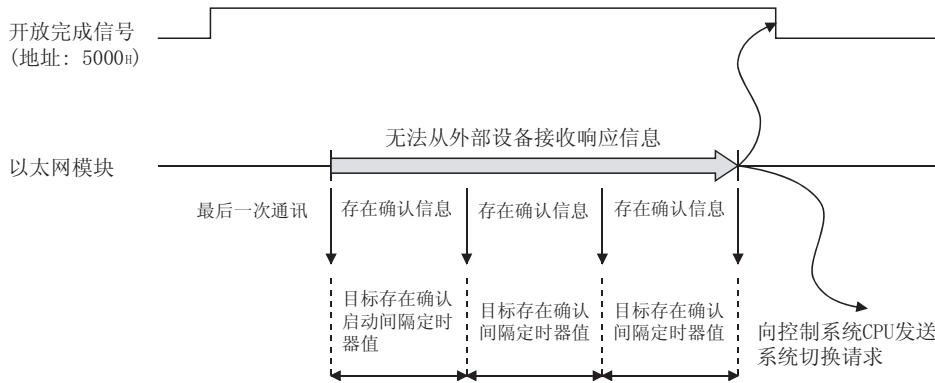
系统 A 作为控制系统运行，系统 B 作为待机系统运行。

外部设备更改连接目标，并且与安装在控制系统 CPU(系统 B)主基板的以太网模块进行通讯。

*1 通过 TCP 连接安装在待机系统 CPU 主基板上的外部设备和以太网模块，可以检测待机系统 CPU 端错误。

*2 通讯出错时的系统切换定时
以下显示在通讯错误时向控制系统 CPU 发送系统切换请求的定时。

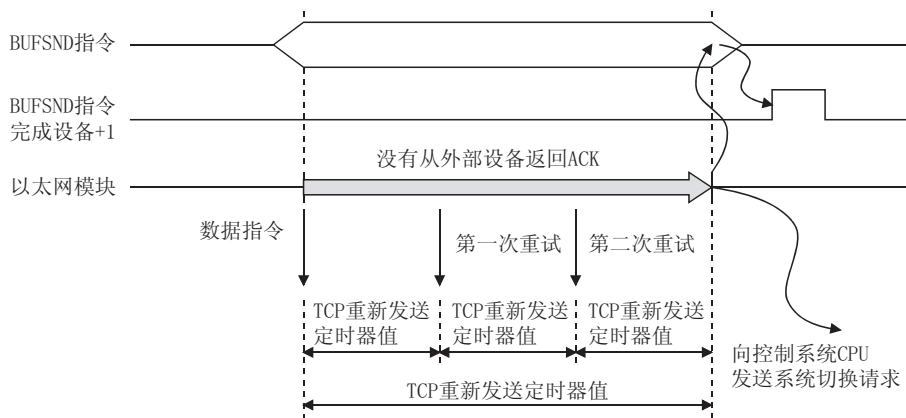
(a) 通过“存在确认”进行系统切换定时



- 1) 如果没有在预先设置的时间进行通讯，以太网模块对连接为开放的外部设备进行存在确认。(关于存在确认功能，请参阅 5.2.2 节)
- 2) 如果以太网模块无法接收外部设备发出的相应信息，以太网模块会切断相应连接并向控制系统 CPU 发送系统切换请求。

(上图显示两个重新发送时的存在确认范例。)

(b) “ULP 超时”系统切换定时



- 1) 如果在 TCP 开放或数据传送的 TCP ULP 定时器时期没有从外部设备返回 ACK, 会发生传送错误并且以太网向控制系统 CPU 发送系统切换请求。(关于 TCP ULP 定时器，请参阅 5.2.2 节) (上图显示两次重试的设置举例)

*3 设置是否向 GX Developer 冗余设置中的控制系统 CPU 发送系统切换请求。(请参阅 5.11.3 节)

(4) “断开检测”时发送系统切换

安装在控制系统 CPU 主基板上的以太网模块检测连接在以太网模块上电缆的连接状态，并在检测断开时向控制系统 CPU 发送系统切换请求。

此功能只可用于 QJ71E71-100。

(a) “断开检测时发送系统切换请求”

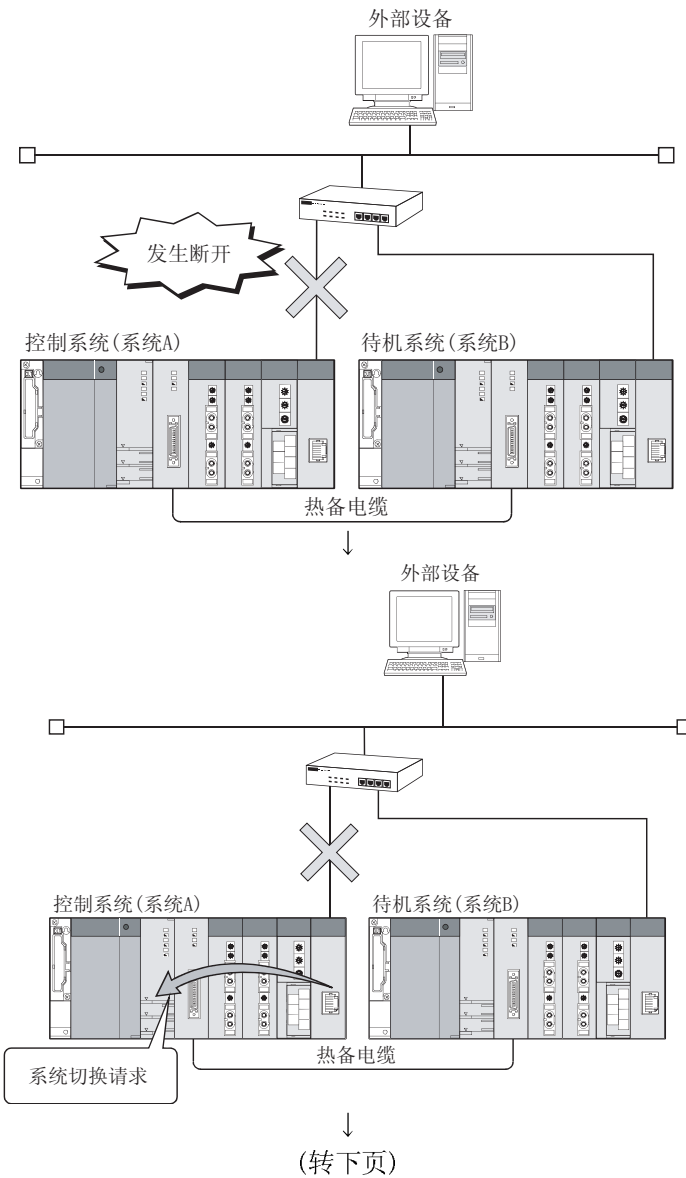
在以下情况下检测断开

- 以太网模块和网络集线器间断开
- 电缆从网络集线器端接口卸下
- 网络集线器断电
- 电缆从以太网模块端接口卸下

(b) “断开检测”时的系统切换请求操作

以太网模块检查连接电缆是否处于“断开”。

以下显示断开检测时执行的系统切换请求操作。



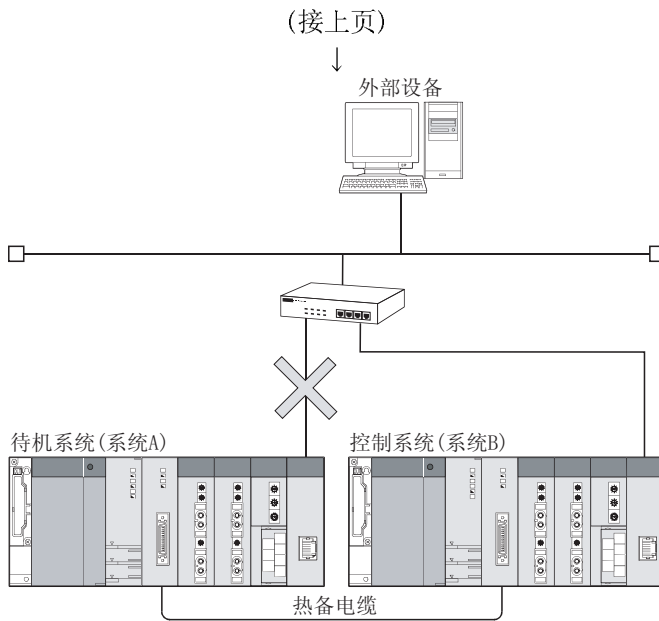
1) 断开监视

以太网模块检查连接电缆是否处于“断开”。(*1)

检测结果保存在缓冲存储区的网络集线器连接状态区(地址: 201(C9H))。(参阅 5.10 节)

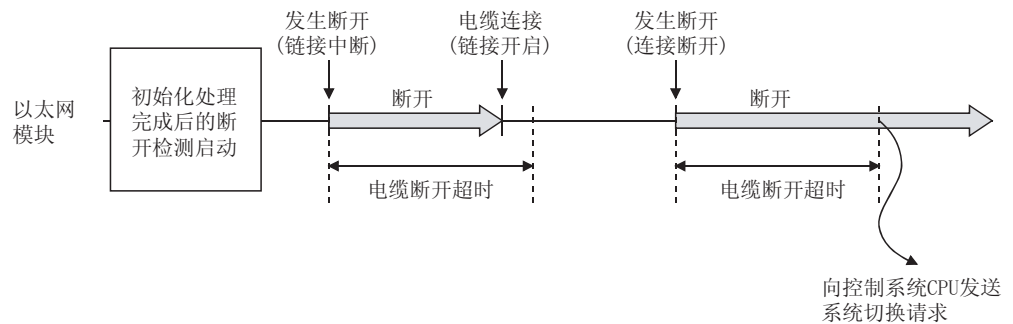
2) 断开检测时(*2)

当安装在控制系统 CPU 主基板上的以太网模块检测到断开时，执行断开状态计时检查；在电缆断开超时设置过程持续为断开状态时，以太网模块向控制系统 CPU 发送系统切换请求。(*3)



3) 系统切换后
系统 A 作为待机系统运行，系统 B 作为控制系统运行。

- *1 如果电缆不是从开始就连接，则以太网不视其为断开。(只有在正常状态变为异常状态时才检测断开。)
- *2 断开检测时的系统切换定时
以下显示发送系统切换请求到控制系统 CPU 的定时。



- 1) 以太网模块在初始化处理完成后启动断开检测。
 - 2) 检测断开时，以太网模块进行断开状态定时检查，并在电缆断开超时设置过程中持续为断开状态时，以太网模块向控制系统 CPU 发送系统切换请求。
 - 3) 当断开状态返回到电缆断开超时设置中的正常时，以太网模块不发送系统切换请求。
- *3 设置是否向 GX Developer 冗余设置中的控制系统 CPU 发送系统切换请求。(参阅 5.11.3 节)

5.11.2 通讯路径迂回功能

(1) 通讯路径迂回功能

当使用任何以下冗余系统支持的应用系统时，如果与以太网模块的通讯中发生错误时，发生通讯错误的路径会自动迂回以继续通讯。

用户不必从一个通讯路径更改到另一个路径。

- 在 OPS 上操作的应用程序。(除了 MELSOF T 产品，例如：GX Developer)*1
- GX Developer
- PX Developer 监视工具

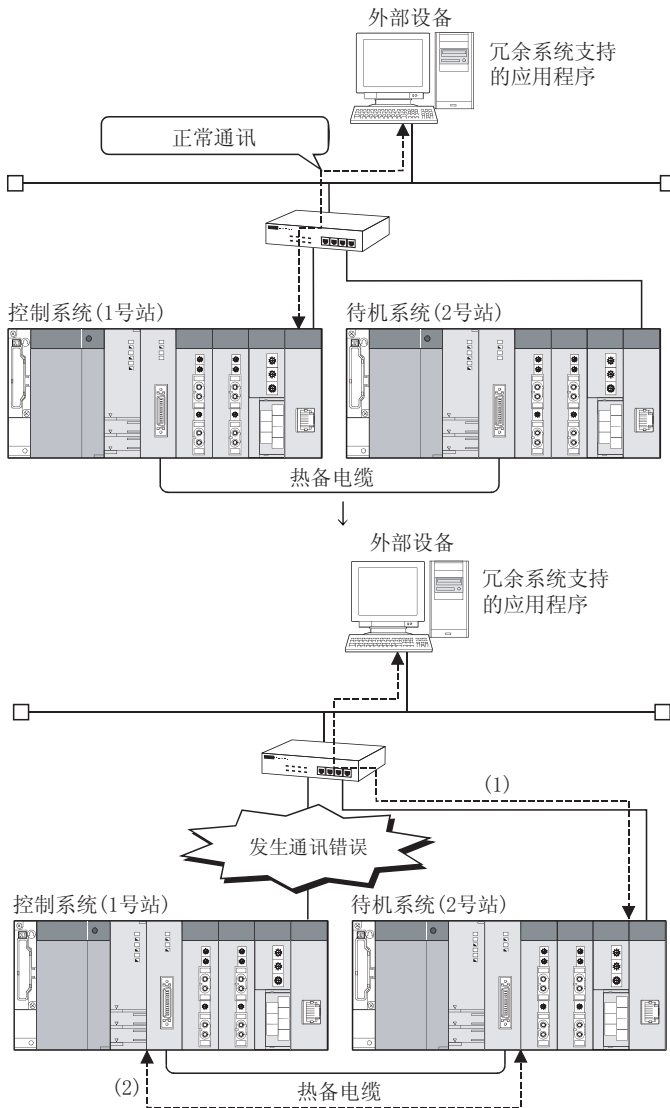
关于兼容版本和可用功能，请参阅相关应用程序手册。

*1 以太网模块可以使用用于 OPS 连接的用户连接来与 OPS 进行通讯。

对 GX Developer 开放设置中的 OPS 连接设置用户连接。(参阅 5.5 节)

(2) 发生通讯错误时的冗余系统支持应用程序操作范例

以下给出了与控制系统 CPU 通讯发生错误时的冗余系统支持应用程序操作范例。



- (1) 迂回路径以访问控制系统CPU。
- (2) 使用热备电缆继续与控制系统CPU进行通讯。

1) 正常通讯时

1号站作为控制系统运行，2号站作为待机系统运行。
冗余系统支持的应用程序正在与控制系统CPU进行通讯。

2) 发生通讯错误时

1号站作为控制系统运行，2号站作为待机系统运行。
(没有系统切换) (*1)

[冗余系统支持应用程序的操作]

因为冗余系统支持的应用程序与安装在控制系统CPU主基板上的以太网模块间发生通讯错误，冗余系统支持的应用程序会自动更改通讯路径并通过待机系统CPU与控制系统CPU通讯。

[待机系统CPU的操作]

因为要发送已接收数据到控制系统CPU，所以通过热备电缆中继数据到控制系统CPU。

*1 在 GX Developer 的冗余设置中设置系统切换请求是否发送到控制系统CPU。(参阅 5.11.3 节)

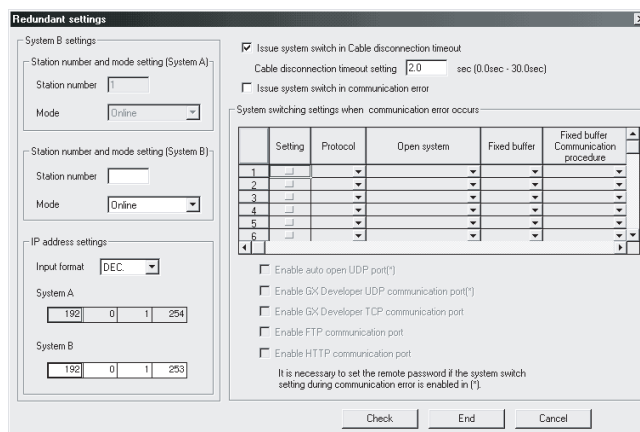
5.11.3 冗余设置

本节介绍 GX Developer 的冗余设置。

[启动步骤]

[设置 MNET/10H 以太网卡号] → **冗余设置** → “冗余设置画面”

[设置画面]



[设置项目]

项目名	说明
系统 B 设置	设置安装在系统 B 上以太网模块的站号和 IP 地址。
电缆断开超时发送系统切换	设置断开检测时是否对控制系统 CPU 发送系统切换请求。
通讯出错时发送系统切换	设置通讯出错时是否对控制系统 CPU 发送系统切换请求。

(1) 系统 B 设置

设置安装在系统 B 上以太网模块的站号、模式和 IP 地址。
设置方法与安装在系统 A 上以太网模块的方法相同。
参阅 4.6 节和 4.7 节。

要点

- (1) 对系统 A 和系统 B 设置不同的站号和 IP 地址。
- (2) 在备份模式下使用冗余系统时，设置系统 A 的操作模式与系统 B 相同。如果系统 A 与系统 B 间的以太网模式不同，则冗余 CPU 中会发生错误。
- (3) 对于除站号、模式和 IP 地址以外的以太网模块设置，系统 A 和系统 B 使用相同的数据。
- (4) 在“设置 MNET/10 以太网卡”和“操作设置”中设置安装在系统 A 上以太网模块的站号、模式和 IP 地址。

(2) “电缆断开超时时发送系统切换”

当此设置为有效时，如果在检测断开后的一段断开监视时期内仍然是断开状态，那么以太网模块向控制系统 CPU 发送系统切换请求。

关于断开检测时的系统切换，请参阅 5.11.1 节。

项目	说明	设置范围/选择
电缆断开超时时发送系统切换	设置是否在断开检测时发送系统切换请求。	<ul style="list-style-type: none"> • 检查(发送系统切换请求) • 未检查(不发送系统切换请求) (缺省值: 检查)
电缆断开超时设置	设置从检测断开到发送系统切换指令至控制系统 CPU 的时间。	0.0s 至 30.0s (缺省值: 2.0s)

要点

设置电缆断开超时设置为几秒到几十秒。

如果时间比以上范围短，则系统会由于噪音等原因而产生系统切换请求。

(3) “通讯错误时发送系统切换”

当此设置为有效时，如果在“发生通讯出错时的系统切换设置”中设置的连接上检测到通讯错误，则以太网模块向控制系统 CPU 发送系统切换请求。

关于通讯出错时的系统切换，请参阅 5.11.1 节。

项目	说明	设置范围/选择
通讯出错时发送系统切换	设置在通讯出错时是否发送系统切换请求。	<ul style="list-style-type: none"> • 检查(发送系统切换请求) • 未检查(不发送系统切换请求) (缺省值: 检查)
发生通讯出错时的系统切换请求 *1*2*3	设置通讯错误会引起发送系统切换请求的目标连接。	检查目标连接。 (缺省值: 不检查)

*1 当满足以下条件时，自动开放 UDP 端口和 GX Developer 通讯 UDP 端口的设置才为有效。

- 1) 远程口令设置为有效。
- 2) 取消远程口令。

满足以上条件时，如果在目标连接上发生通讯错误，则不发送系统切换请求。

*2 不要设置以下连接为目标连接。(不要确认其检查箱)

- 操作设置的初始时间设置 : 一定要等待 OPEN (STOP 时可以通讯)
- 开放设置的协议 : “UDP”
- 开放设置的存在确认 : “Check”

如果以上连接设置为目标连接，那么在电缆断开或外部设备断电引起通讯错误时，冗余系统中会连续发生系统切换。

*3 建议不要对已经在开放设置的开放系统中设置为“MELSOFT 连接”的连接设置为目标连接。(不要确认其检查箱)

如果以上连接设置为目标连接，则所有连接到网络的 MELSOFT 产品都成为目标，而且无法识别目标外部设备 (MELSOFT 产品)。

5.11.4 用于冗余系统支持功能的缓冲存储器

GX Developer 的冗余设置状态可以在以下缓冲存储器区中确认。

(1) “电缆断开时发送系统切换” (地址: 20992(5200H))

存储“电缆断开超时时发送系统切换”的设置状态。

0: 未设置

1: 设置

(2) 断开超时设置(地址: 20993(5201H))

存储“断开超时设置”的设置状态。

设置时间 = 设置值 × 500ms

(举例) 设置时间为 2s 时, 存储值为 4H。

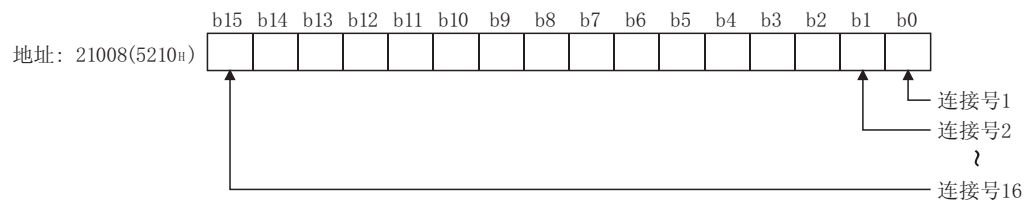
(3) “发生通讯出错时的系统切换设置” (用户连接)

(地址: 21008(5210H))

存储“发生通讯错误时的系统切换设置”的设置状态到用户连接。

0: 未设置

1: 设置



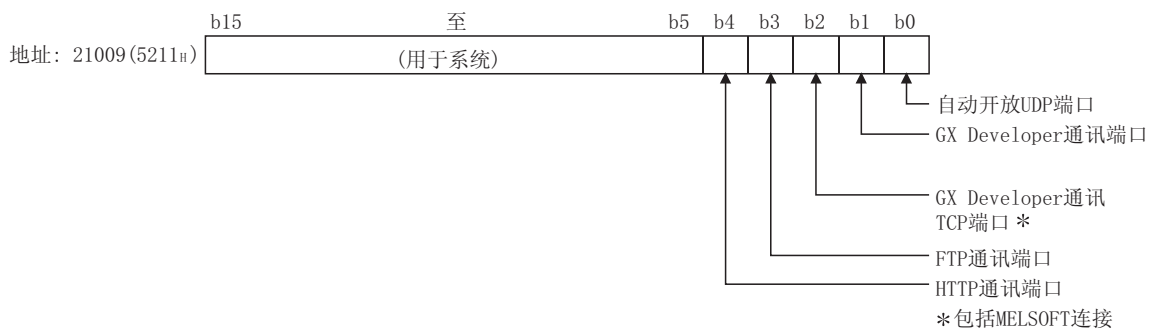
(4) “发生通讯出错时的系统切换设置” (系统连接)

(地址: 21009(5211H))

存储“发生通讯错误时的系统切换设置”的设置状态到系统连接。

0: 未设置

1: 设置



5.11.5 用于在冗余系统中使用以太网模块的数据通讯

本节说明了以太网模块安装在冗余系统主基板上时的数据通讯。

除以下外，数据通讯可以按照以太网模块安装在单 CPU 系统主基板上时的相同方法进行。

请参阅相应功能的说明项目。

(1) “初始化处理”

(a) 执行初始化处理

使用 GX Developer 对数据通讯进行设置，写入相同的参数到控制系统 CPU 和待机系统 CPU，然后同时复位两个冗余 CPU。

注意，对系统 A 和系统 B 必须设置不同的站号和 IP 地址。(参阅 5.11.3 节)

(b) 进行重新初始化处理时(参阅 5.2.3 节)

不要更改本地站 IP 地址和操作设置。

否则无法进行正常通讯。

1) 使用 UINI 指令时

对控制数据的更改目标(S1+2)指定“0H”后，执行指令。

2) 直接写入到缓冲存储器

写入“1”到通讯条件设置区(地址:1FH)的位 15，而不更改缓冲存储区的设置值。

(c) I/O 信号进行初始化处理

因为输出信号(Y)在待机系统 CPU 中关闭，所以不可用 I/O 信号进行的初始化处理。

设置 GX Developer 的网络参数并执行初始化处理。(参阅 5.2 节)

(2) “开放/关闭处理”

(a) 通过 TCP/IP 进行通讯时

使以太网模块处于开放等待状态(被动开放)，并执行外部设备中的开放/关闭处理。

在以太网模块中进行主动开放处理时，执行关闭处理；但如果在执行关闭处理前发生系统切换，则无法进行关闭处理。

- (b) 使用用户连接与待机系统进行通讯(使用 MC 协议或随机访问缓冲存储器进行通讯)
 - 1) 运行设置(参阅 4.7 节)
设置初始化定时设置为“始终等待打开”(STOP 时可以进行通讯)。
 - 2) 开放设置(参阅 5.5 节)
进行 TCP/IP 通讯时, 设置开放系统为“非被动或完全被动”
- (c) 用 I/O 信号来开放/关闭处理
因为在待机系统 CPU 中输出信号(Y)关闭, 所以不可用 I/O 信号进行开放/关闭处理。
设置操作设置的初始化定时设置为“始终等待打开(STOP 时可以进行通讯)”或专用指令(OPEN/CLOSE 指令)。(参阅 5.6 节)

要点

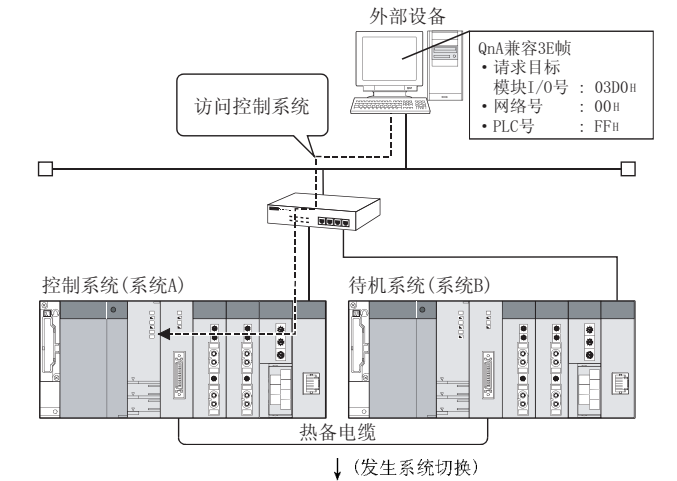
使用用户连接进行通讯时, 建议准备与系统 A 和系统 B 通讯的连接。 如果自系统中发生通讯出错或发生系统切换, 则以上可立即与其它系统进行通讯。 在以太网模块上, 最多可设置 16 个用户连接。
--

(3) “使用 MC 协议通讯”

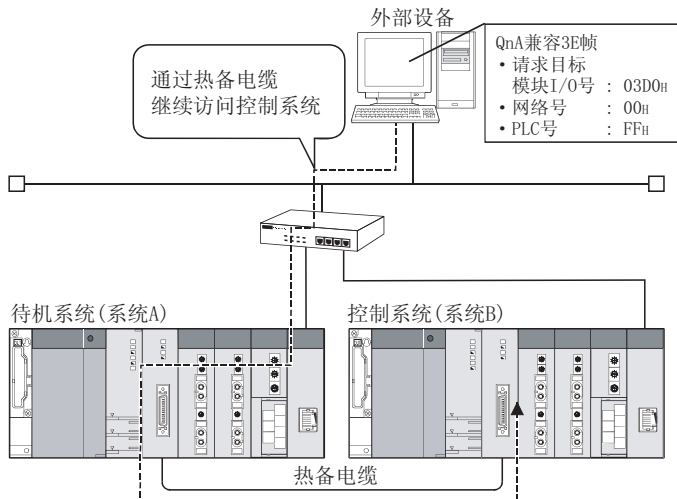
QnA 兼容 3E 帧或 4E 帧可用来访问控制系统/待机系统或系统 A/系统 B。
关于信息格式和数据说明方法，请参阅参考手册。

(a) 访问控制系统/待机系统或系统 A/系统 B 所执行的操作

1) 发生系统切换时(访问控制系统 CPU 的举例)



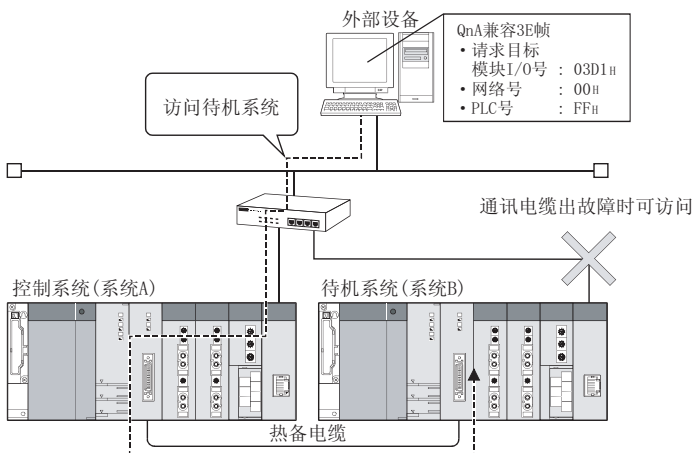
1) 外部设备连接到安装在控制系统 CPU 主基板上的以太网模块并访问控制系统 CPU。



2) 如果发生系统切换，外部设备通过热备电缆自动继续访问控制系统。但是，当带连接目标的通讯电缆出故障或待机系统关闭电源时，必须更改外部设备端的连接目标。

2) 访问非连接目标的系统

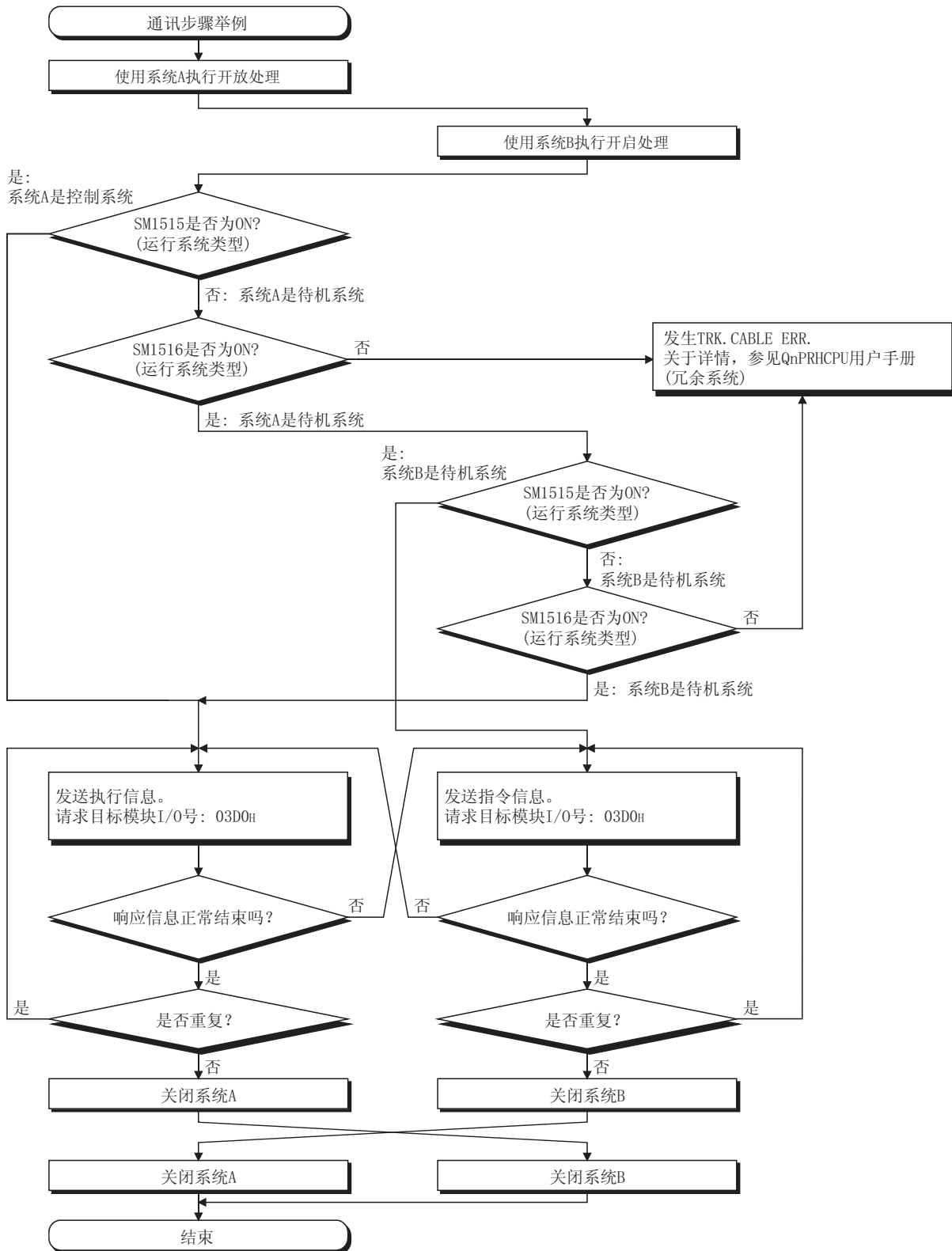
(连接到控制系统和访问待机系统 CPU 范例)



1) 外部设备连接到安装在控制系统 CPU 主基板上的以太网模块并通过热备电缆访问待机系统 CPU。

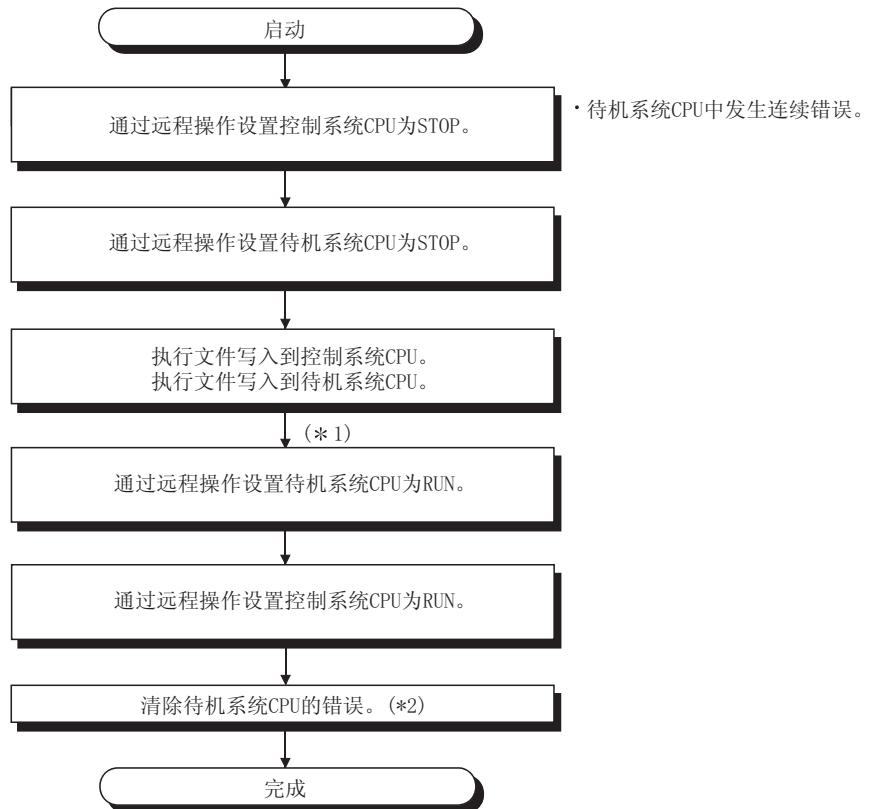
如果外部设备和待机系统间的电缆出故障时可以进行访问。

(b) 访问冗余系统中控制系统 CPU 的通讯步骤范例
 以下显示访问冗余系统中控制系统 CPU 的通讯步骤范例。
 使以太网模块处于启动等待状态并执行外部设备中的开放/关闭处理。



(c) 写入参数文件或程序文件的注意事项

- 1) 一定要写入相同的文件到系统 A 和系统 B。
如果写入的文件不同或者只写入到一个系统会发生错误。
- 2) 当 CPU 的运行状态为“STOP”时，写入参数文件或程序文件。
- 3) 按照以下步骤执行文件写入。



*1 写入参数文件后，同时复位两个冗余 CPU。

*2 当控制系统 CPU 的操作状态从 STOP 切换到 RUN，检查待机系统 CPU 的错误状态，如果已经发生错误，设置错误代码(6010H)为 SD50，然后开放 SM50 来清除错误。

(4) “通过固定缓冲器通讯”

(a) 接收待机系统中的处理

当数据发送到安装在待机系统上的以太网模块时，则废弃以太网模块发送的数据并且不执行数据接收处理。

(固定缓冲期接收完成信号(地址相应位: 5005H)不关闭。)

(b) 中断程序中的接收处理

执行中断程序前，通过系统切换把控制系统切换到待机系统时，保留中断因素。

当再次发生系统切换，待机系统通过中断程序切换到控制程序时，这些操作由保留的中断因素执行。

(中断因素不传送到其它系统。)

(c) 外部设备发送数据时

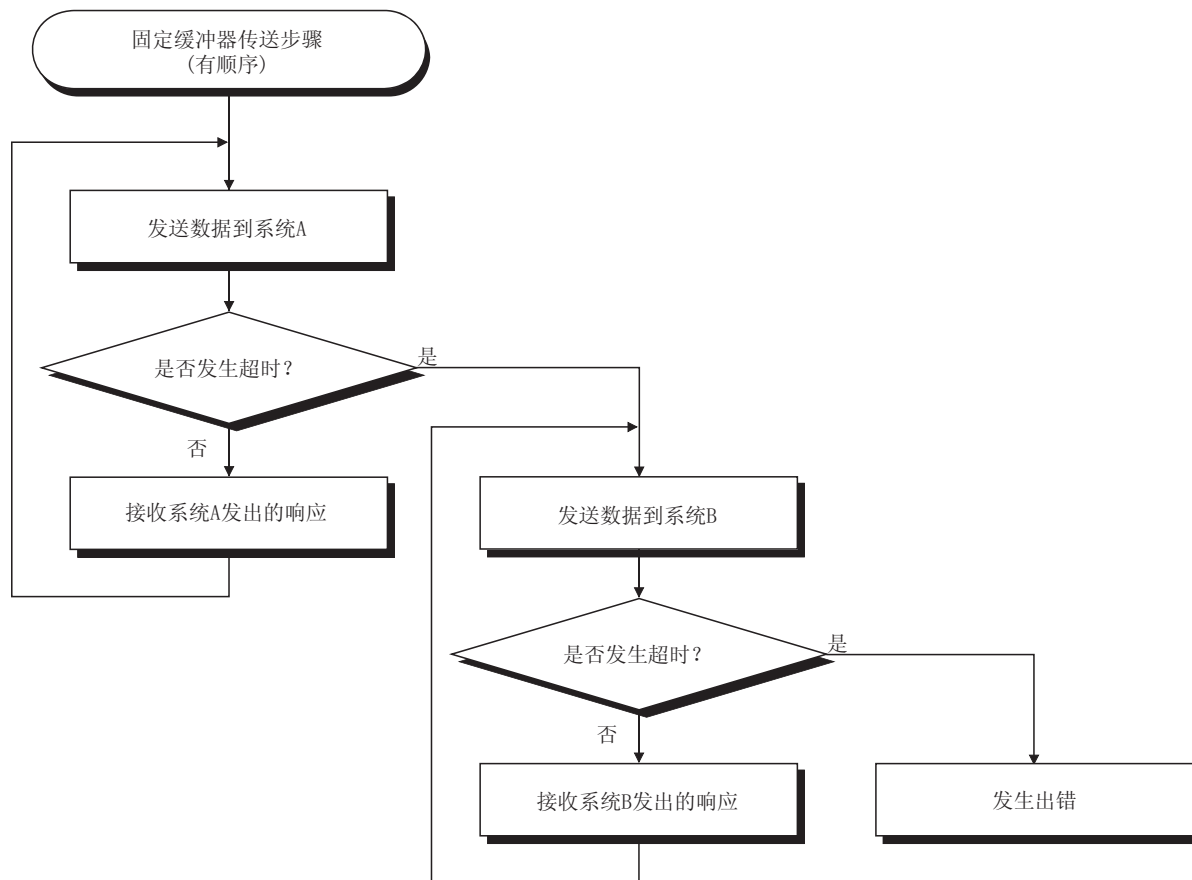
1) 当通讯顺序为“有顺序”时

如果发生以太网模块的响应超时，更改其它系统的连接目标并发送数据。

2) 当通讯顺序为“无顺序”时

发送相同的数据到控制系统和待机系统。

以下显示当数据通过固定缓冲器(有顺序)传送到冗余系统时的传送步骤举例。



要点
在系统切换时执行重新传送处理，请注意以下方面。 (1) 传送和接收同步时进行通讯 通讯时如果传送和接收不同步，则可能发生系统切换。 发生系统切换时，在初始化同步前重新开始通讯以保证安全。 (2) 使用专用指令时 因为无法判断是否完成写入指令的执行，相同的指令必须再次执行。 在这种情况下，注意相同的指令可能执行两次。

(5) “通过随机访问缓冲器通讯”

不追踪以太网模块的缓冲存储器。所以，当写入数据到随机访问缓冲器时，写入相同数据到控制系统和待机系统。

(6) “使用电子邮件功能时”

(a) 接收电子邮件

1) 通过以太网模块接收电子邮件后，发送响应邮件到邮件发送源。

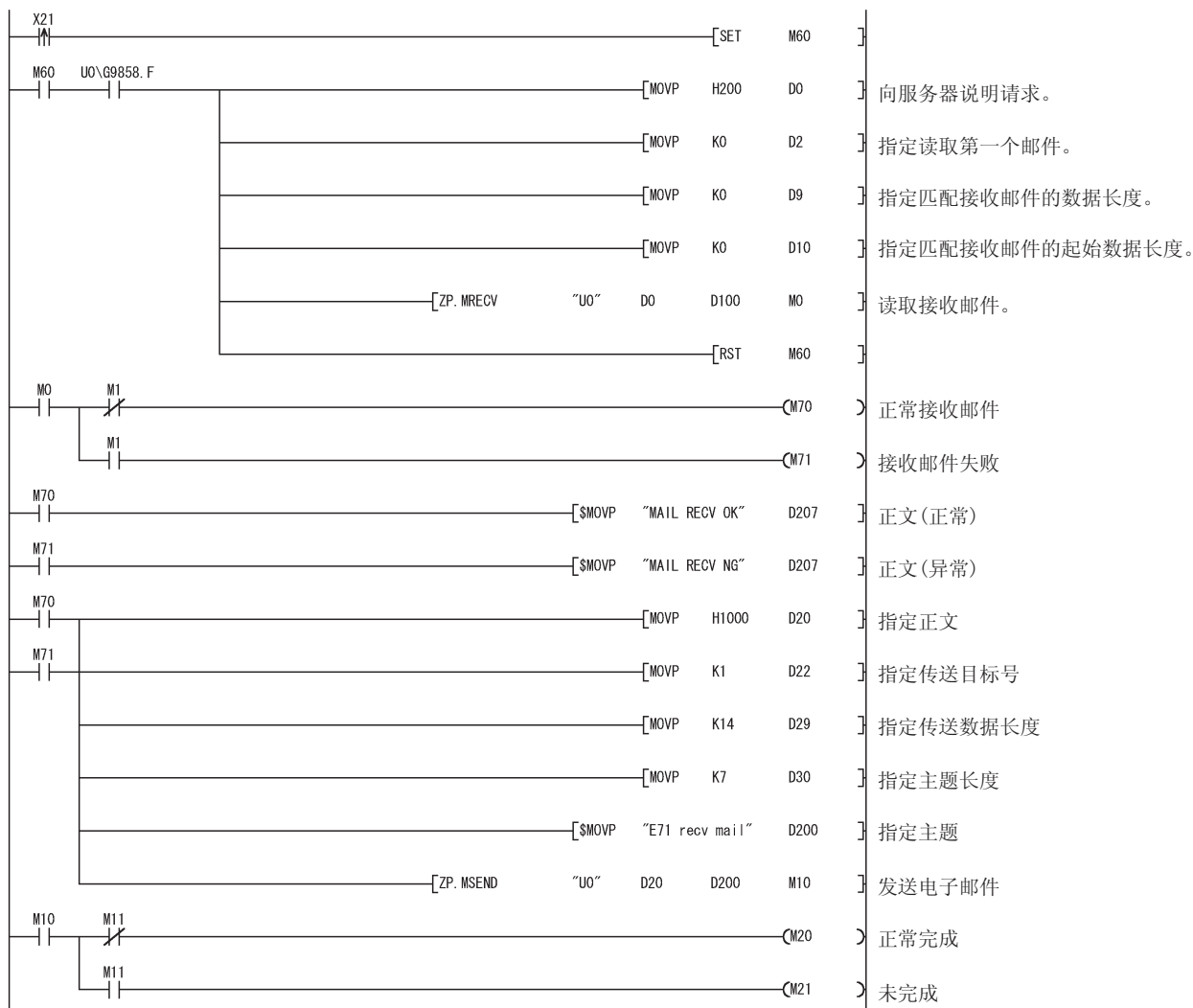
这样邮件发送源就可以确认收到电子邮件。

2) 执行 MRECV 指令后，从邮件服务器上删除已经读取的电子邮件。所以，如果在执行 MRECV 指令时发生系统切换，那么再次执行 MRECV 指令所发生系统切换后新控制系统 CPU 不会接收邮件。(电子邮件已经从邮件服务器中删除)

(b) 电子邮件接收程序举例

在以下程序中，开放 X21 可使安装在 I/O 信号 X/Y00 到 X/Y1F 位置上以太网模块通过执行 MRECV 指令接收电子邮件，并通过执行 MSEND 指令发送响应邮件到传送源。

关于电子邮件功能，请参阅用户手册(应用篇)中的第 2 章。



(c) 使用通知功能时

因为在以下任一情况下都可以从控制系统和待机系统中发送报告邮件，执行处理将丢弃个人计算机接收处重复的邮件。*1

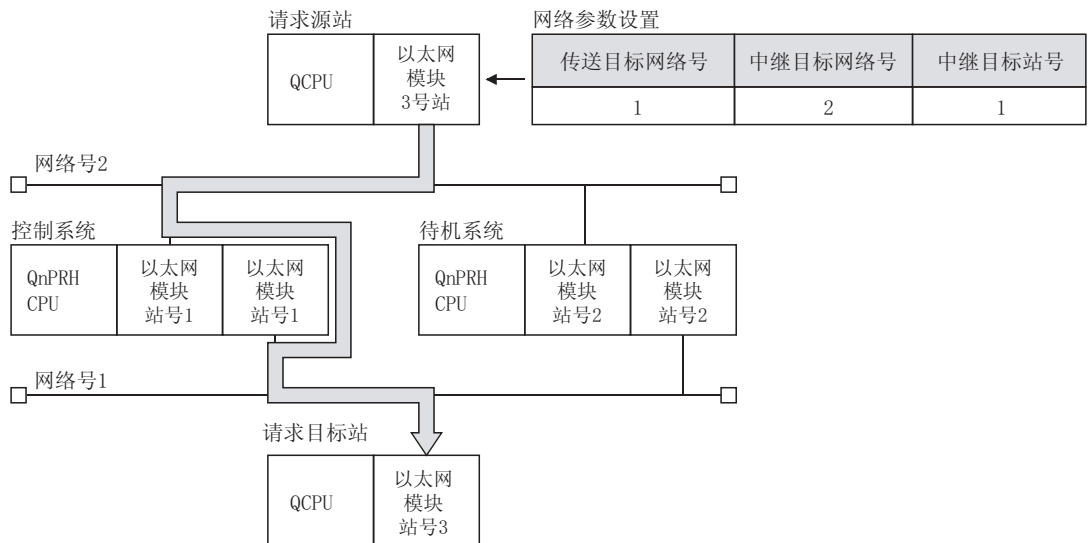
- 当 CPU 操作状态设置为报告环境。
- 当追踪设置为报告环境的设备数据。

*1 通过设置以下 SM 软元件为报告环境，报告邮件可以指定冗余系统的系统，并且设备可用作条件来识别哪个系统发送重复邮件。

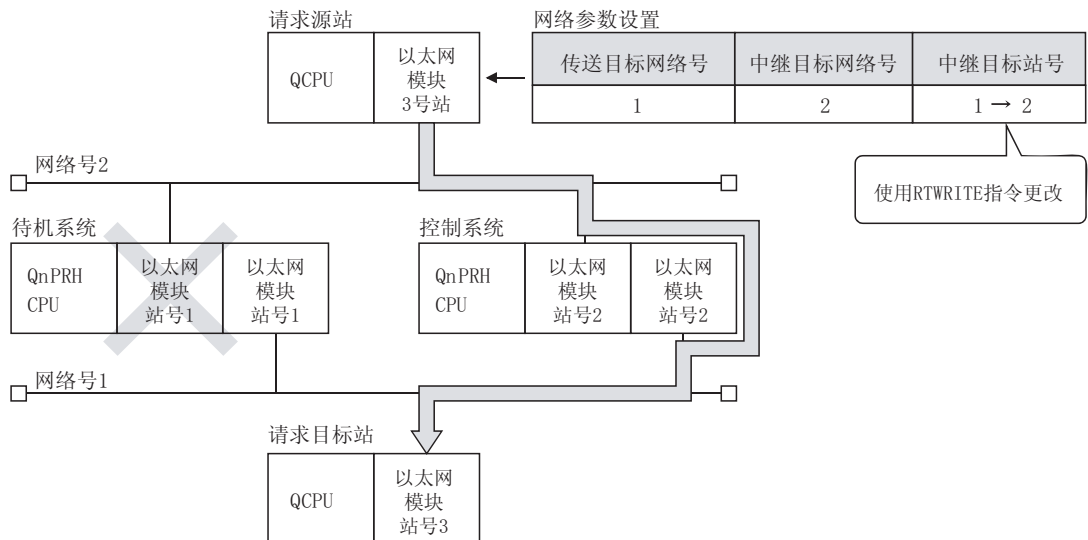
- SM1511(系统 A 识别标记)
- SM1512(系统 B 识别标记)
- SM1515、SM1516(运行系统类型)

(7) “通过 MELSECNET/H 或 MELSECNET/10 进行通讯时”

- (a) 当冗余系统在网络上时
通过冗余系统进行访问时，需要注意的是必须在系统切换时在请求源站或中继站中通过 RTWRITE 指令更改路由参数设置。(参阅以下(b))
- (b) 通过冗余系统访问时
通过使用 MELSECNET/H, MELSECNET/10 中继通讯功能的以太网，路由参数设置必须设置为请求源站或中继站来访问其它站。(参阅用户手册(应用篇)中的第三章)
当通过冗余系统进行访问时，设置控制系统的站点为路由站。



在中继源站或中继站上用 RTWRITE 指令更改路由参数设置，这样在发生系统切换后通过新控制系统的站点可以进行访问。(关于 RTWRITE 指令，请参阅 QCPU (Q 模式) / QnACPU 编程手册(公共指令))



(8) “当 QCPU 使用数据链接指令访问其它站 PLC”

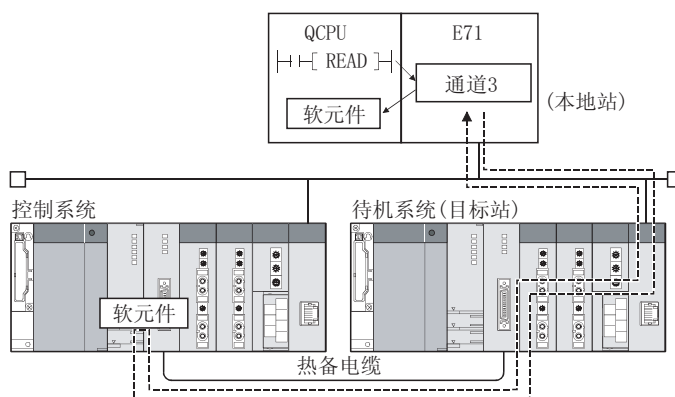
(a) 冗余系统兼容数据链接指令

1) 使用任何以下数据链接指令来指定控制数据的目标站 CPU 类型，可以访问控制系统/待机系统或系统 A/系统 B。(关于数据链接指令，请参阅用户手册(应用篇)的第 4 章或第 6 章。)

- READ/SREAD 指令
- WRITE/SWRITE 指令
- REQ 指令

2) 访问控制系统/待机系统或系统 A/系统 B 所执行的操作(执行 READ 指令的操作)

当目标站在待机系统中，接收指令发送到控制系统 CPU(目标站 CPU 类型:300H)，那么数据通过热备电缆中继到控制系统以读取控制系统 CPU 的数据。



(b) 发生出错时的处理

如果对冗余系统中的指定站(控制系统 CPU/待机系统 CPU)执行数据连接指令或目标站引起系统切换，数据连接指令可能引起错误。(错误代码:4244H、4248H)

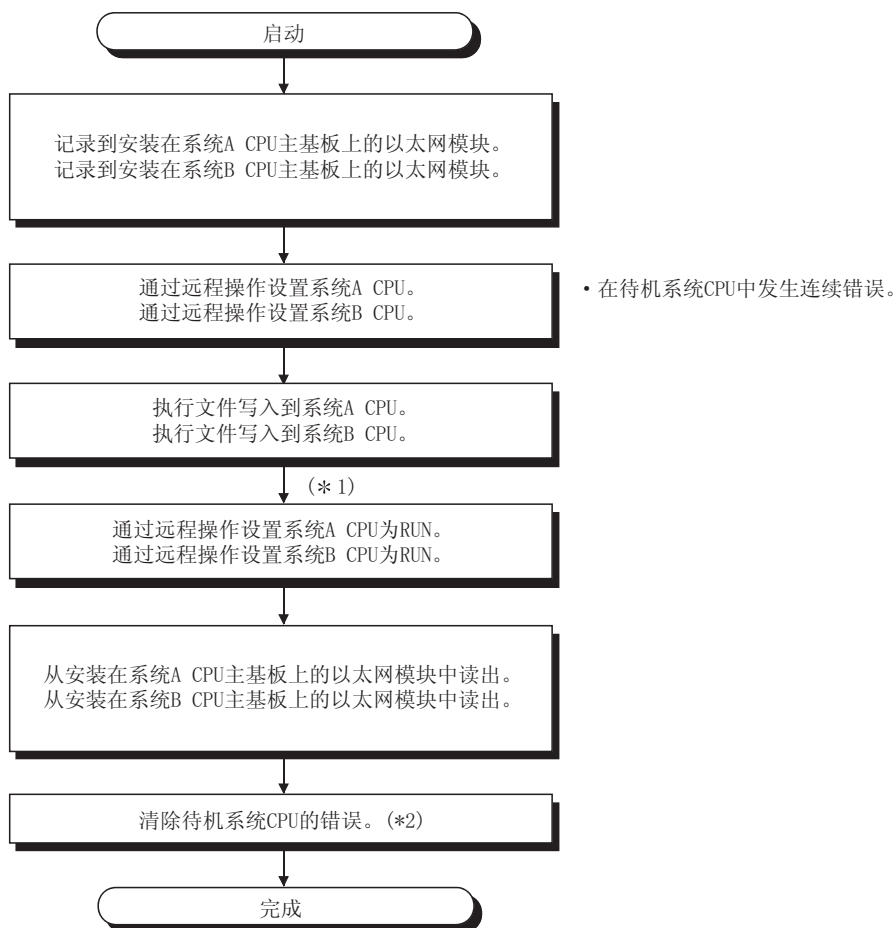
如果数据连接指令由于以上错误而引起出错，再次执行数据连接指令。

- (c) SEND 指令
 - 1) 执行 SEND 指令
当目标站在冗余系统中时，通讯请求源站必须识别执行 SEND 指令的目标站为控制站。
当目标站是待机系统时，因为在 SEND 指令发送数据后在目标站中不执行 RECV 指令，所以不使用目标站存储通道。(通道正在使用)
 - 2) 进行广播时
在进行广播的网络上有冗余系统时，因为不对待机系统执行 RECV 指令，所以无法使用存储通道。(通道正在使用)
- (d) RECV 指令、中断程序(RECVS 指令)
对冗余系统执行 SEND 指令时，RECV 指令或中断程序(RECVS 指令)的处理根据以下情况更改。
 - 1) 对控制系统执行 SEND 指令和对中断程序执行 RECV 指令间发生系统切换
在执行 RECV 指令或中断程序前控制系统通过系统切换到待机系统，会保持缓冲存储器的 RECV 指令执行请求区(地址:205(CDh))或中断程序的中断因素(中断指针)。
当再次发生系统切换并且待机系统切换到控制系统时，使用缓冲存储器的保持 RECV 指令执行请求区或中断程序的中断因素来执行 RECV 指令或中断程序。
 - 2) 对待机系统执行 SEND 指令时
当待机系统接收传送站发出的数据时，会保持缓冲存储器的 RECV 指令执行请求区(地址:205(CDh))和中断程序的中断因素(中断指针)。
所以，待机系统通过系统切换到控制系统时，使用缓冲存储器的保持 RECV 指令执行请求区和中断程序的中断因素来执行 RECV 指令或中断程序。

(9) “使用文件传送(FTP 服务器)功能时”

(a) 写入参数文件或程序文件

- 1) 一定要写入相同的文件到系统 A 和系统 B
如果写入不同文件或文件只写入一个系统会发生错误。
- 2) 当 CPU 的操作状态为“STOP”时，写入参数文件或程序文件。
- 3) 按照以下步骤执行文件写入。



*1 写入参数文件后，同时复位两个冗余 CPU。

*2 当控制系统 CPU 的操作状态从 STOP 切换到 RUN 时，检查待机系统 CPU 的出错状态。如果已经发生出错，设置出错代码为(6010H)到 SD50，并开放 SM50 来清除出错。

(10) “使用专用指令时”

如果执行专用指令时发生系统切换，可能无法完成专用指令。

这种情况下，在系统切换后再次执行新控制系统 CPU 中的专用指令。

6 使用 MC 协议进行通讯

本章概括地介绍了 MELSEC 通讯协议(后面简称为 MC 协议)。
关于使用 MC 协议的数据通讯功能的详情, 请参阅下列手册:
MELSEC 通讯协议参考手册 (SH(NA)-080414C)

6.1 数据通讯功能

MC 协议是 MELSEC 协议的简称, 它是 Q 系列 PLC 的通讯系统。使用该协议, 外部设备可以通过 Q 系列以太网模块或 Q 系列串行通信模块, 从 PLC CPU 读取数据和程序或将数据和程序写入 PLC CPU。

按照 MELSEC PLC 协议可安装应用程序, 且可以发送和接收数据的任何外部设备都可以使用 MC 协议与 PLC CPU 进行通讯。

6.1.1 使用 MC 协议访问 PLC CPU

该节说明使用 MC 协议访问 PLC CPU 的主要功能。
在 PLC 端, 以太网模块根据外部设备的命令发送和接收数据。
因此, PLC CPU 端无需进行数据通讯的顺控程序。

(1) 数据读/写

该功能对 MELSECNET/H 进行数据的读和写, MELSECNET/10 的本地站或其它站的 PLC CPU 软元件存储器及智能功能模块缓冲存储器。

通过读写数据, 可以在外部设备端执行 PLC CPU 操作监视、数据分析和生产管理。

另外, 可以从外部设备端执行生产指令。

(2) 文件读/写

该功能可以读取文件和写入文件诸如存储在 PLC CPU 中的顺控程序和参数文件。
通过读写这些文件, 可以在外部站端管理其它站上 QCPU 和 QnACPU 的文件。

另外, 可以从外部设备端更改(更换)执行程序。

(3) PLC CPU 的远程控制

该功能执行远程运行/停止/暂停/锁存清除/复位操作。
使用 PLC CPU 远程控制功能可以从外部设备端远程操作 PLC CPU。

		功能		
使用 MC 协议进行通讯	使用 4E 帧进行通讯	使用 ASCII 代码进行通讯 使用二进制代码进行通讯		
	使用 QnA 兼容 3E 帧进行通讯			
	使用 A 兼容 1E 帧进行通讯			
	软元件存储器读/写	以位/字为单位成批读/写		
		软元件存储器监视		
		多块成批读/写		
		通过扩展指定读/写		
		通过网络系统访问其它站		
	以太网模块缓冲存储器的读/写			
	智能功能模块缓冲存储器的读/写			
顺控程序文件的读/写				
PLC CPU 的状态控制 (远程运行/停止, 等)				

6.1.2 数据通讯的信息格式和控制步骤

使用 MC 协议的数据通讯功能与 A/QnA 系列以太网接口模块支持的 PLC CPU 中的读/写数据的功能一一对应。

因此, 信息格式和控制步骤与使用 A/QnA 系列以太网接口模块访问 PLC 时的相同。

- 使用 QnA 兼容 3E 帧进行通讯
该信息帧用于 QnA 系列以太网接口模块。
- 使用 A 兼容 1E 帧进行通讯
该信息帧用于 A 系列以太网接口模块。

外部设备端可以通过 A/QnA 系列以太网接口模块, 用访问 PLC 的程序访问 Q 系列 PLC。

(例子)

标题			副标题	文字(命令)				
以太网 (14字节)	IP (20字节)	TCP / UDP		PC 编号	ACPU 监控 定时器	起始软元件		软元件点数
			00h	FFh	L	H	L	H
			0Ah	1 00h	64h	00h	00h	20h
								0Ch
								00h

(A兼容1E帧的命令信息)

标题			副标题	文本(响应)	
以太网 (14字节)	IP (20字节)	TCP / UDP		完成代码	读取数据
			80h		
				00h	10h
					10h
					10h
					10h

(A兼容1E帧的响应信息)

要点

以下手册可用来执行使用 MC 协议的数据通讯, 即:
Q 系列 MELSEC 通讯协议参考手册 (另售)

6.1.3 执行数据通讯的 PLC CPU 设置

通过执行下面所示的在 PC 上的 GX Developer 设置，可以使用 MC 协议进行数据通讯并将参数写入 PLC CPU 中。

- 1) 设置以太网卡号(参阅第 4.6 节)
- 2) 初始化设置(参阅第 5.2 节)
- 3) 开放设置(参阅第 5.5 节)

要点
<p>(1) 通过使用以太网模块的自动开放 UDP 端口，可以执行使用 MC 协议的通讯，而无须考虑 PLC CPU 的运行/停止状态。 不使用自动开放 UDP 端口时，首先连接用户连接端口，并参照本手册第 4 章“开始操作之前的设置和步骤”和第 5 章“通讯步骤”执行数据通讯。</p> <p>(2) 从外部设备写入 PLC CPU 时，可以使用 GX Developer “以太网操作设置”参数设置允许禁止在 CPU RUN 时间写设置值。 * 将数据写入 MELSECNET/H 的远程 I/O 站时，将 CPU RUN 时间设置的允许/禁止写入参数设置为“允许”。</p>

6.2 使用 MX Component

如果外部设备是运行下列之一基本操作系统的 PC，则可以使用下列另售的通讯支持工具之一为外部设备创建一个通讯程序，而不用考虑具体的 MC 协议(传送/接收步骤)。关于 MX Component 的概述，请参阅附录 9。

(支持的基本操作系统)

- Microsoft® Windows® 95 操作系统
 - Microsoft® Windows® 98 操作系统
 - Microsoft® Windows NT® Workstation 4.0 操作系统
 - Microsoft® Windows® Millennium Edition 操作系统
 - Microsoft® Windows® 2000 Professional 操作系统
 - Microsoft® Windows® XP Professional 操作系统
 - Microsoft® Windows® XP Home Edition 操作系统
- * 根据使用 MX Component 的不同版本，支持不同的操作系统。
关于详情，请参阅 MX Component 手册。

(通讯支持工具)

- MX Component (SW0D5C-ACT-E 或以后的版本)

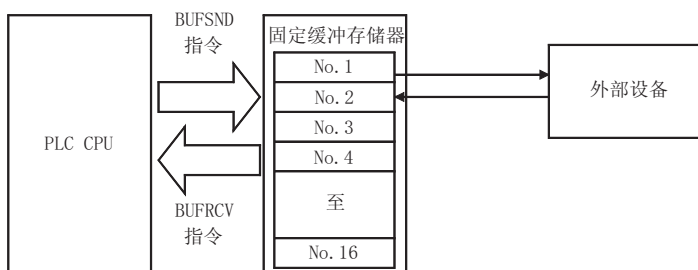
7 固定缓冲存储器通讯(有顺序控制方法)

本章介绍 PLC CPU 如何使用以太网模块固定缓冲存储器（有顺序控制方法），以 1:1 的模式与外部设备进行通讯。

7.1 控制方法

以下说明如何使用固定缓冲存储器和有顺序控制方法进行通讯。
在使用固定缓冲存储器进行通讯的过程中，从 PLC CPU 和外部设备的数据传送是通过交换信号的方式来实现的。

(1) 在通讯过程中的数据流如下所示：



(2) 数据可以通过下列外部设备来进行通讯。

- 以太网上与以太网模块连接的设备。
- 与具有中继功能的路由器连接的设备(见 5.3 部分)

如下图所示，在使用每个固定缓冲存储器(1 号到 16 号)时，应在通过开放以太网模块来固定每个缓冲存储器的外部设备时，对目标设备 and 应用条件(发送/接收、有顺序/无顺序等)进行设置。

(a) 在 TCP/IP 通讯时

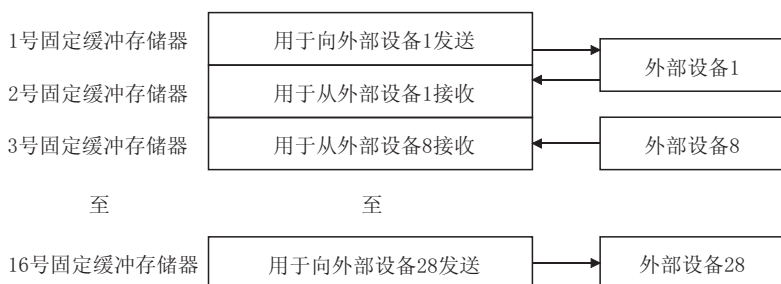
只有当应用连接的开放完成信号切断时，才可以改变外部设备。

(b) 在 UDP/IP 通讯中

不管应用连接的状态如何，外部设备都可以进行改变。

(通讯地址设置区域的“目标 IP 地址”和“目标端口编号”可以进行改变。然而，“本地站端口号”不能改变)

在改变外部设备时，不要使用“成对开放”和“存在确认”功能。



要点

在选择有顺序控制方法的通讯中，在开放过程完成之后，数据可以通过下列方法进行通讯。

- 用有顺序控制方法(发送/接收)进行固定缓冲存储器通讯
- 用随机访问缓冲存储器进行通讯
- 用 MC 协议进行通讯

(3) 当传送/接收数据时，以太网模块进行下列处理：

(a) 在发送数据时

当 PLC CPU 用顺控程序执行 BUFSND 专用指令(*1)时，以太网模块将可用固定缓冲存储器(n 号)的数据传送到与固定缓冲存储器 n 号(*2)相应的通讯地址设置区域(地址：28_H 至 5F_H 和 5038_H 至 507F_H)中规定的外部设备。

(b) 在数据接收时

如果数据是从与固定缓冲存储器 n 号(*2)相应的通讯设置区域设置的外部设备上收到的，那么以太网模块将处理收到的数据。

如果数据是从未在缓冲存储器的连接信息区域中作设置的外部设备中收到的，则以太网模块不请求将所收到的数据读到 PLC CPU 端。

*1 有关专用指令的详细情况，请参阅第 10 章“专用指令”部分。

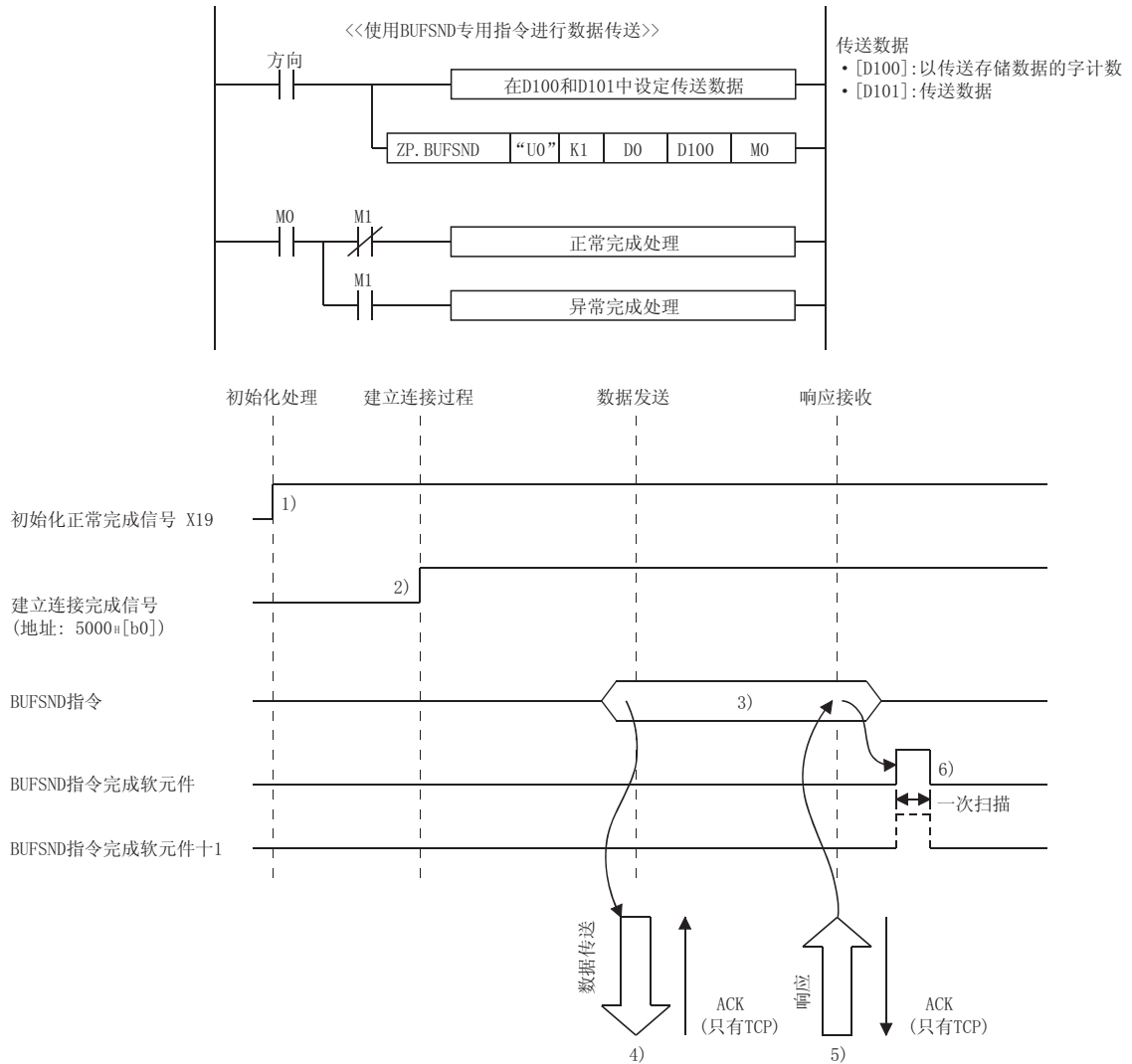
*2 在 TCP/IP 非被动开放的情况下，与外部设备进行通讯的数据存储在与 n 号固定缓冲存储器相对应的连接信息区中。

要点

当以太网模块在同步广播接收过程中相应的固定缓冲存储器中存储了接收数据后，它将更新与相应的 n 号固定缓冲存储器连接信息区域(地址：78_H 至 C7_H 和 5820_H 至 586F_H)中的目标 IP 地址以及目标端口号。

7.2 发送控制方法

本节说明当数据从以太网模块用 1 号固定缓冲存储器以及与 1 号连接相应的区域发往外部设备时的控制方法。



- 1) 确认初始化过程的正常完成。
- 2) 确认 1 号连接的开放过程正常完成。
- 3) 执行 BUFSND 专用指令。
以太网模块处理并传送下列数据。
 - 写入发送数据长度并发送数据给 (1 号) 固定缓冲存储器区域。
 - 发送数据长度 : 目标固定地址的 (*1) 起始地址区域
 - 发送数据 : 从目标固定缓冲存储器 (*1) 起始地址开始的区域
- *1 发送数据长度即字数。
- 4) 根据发送数据长度指定的固定缓冲存储器 (1 号) 中的发送数据量发送到指定的外部设备中 (在开放过程中设置)。
- 5) 当从以太网模块收到数据时, 外部设备向以太网模块返回一个“响应”。
- 6) 在从外部设备接收到“响应”后, 以太网模块结束数据传送。
如果在响应监视定时器时间里没有返回“响应”(见 5.2 部分), 即出现数据发送错误。

正常完成时

- BUFSND 指令完成软元件 : 开
- BUFSND 指令完成软元件+1 : 关
- BUFSND 指令完成软元件区域 (*2) : 0000_H
- 响应结束代码 (*3) : 00_H

异常完成时

- BUFSND 指令完成软元件 : 开
- BUFSND 指令完成软元件+1 : 开
- BUFSND 指令完成状态区域 (*2) : 除了 0000_H 之外的数值
- 响应结束代码 (3) : 除了 00_H 之外的数值

在数据传送异常完成后, 再次执行 BUFSND 专用指令以重复传送过程。

*2 完成时的状态存储在 BUFSND 专用指令完成状态区域。有关专用指令的详细情况, 请参阅第 10 章“专用指令”部分。

*3 响应结束代码存储在缓冲存储器的通讯存储状态区域。
有关响应结束代码的详细情况, 请参阅第 7.4.2 节“应用数据 (5) 结束代码”。

要点	
	在以太网模块开放完成信号 (地址: 5000 _H …对应位) 从关切换到开时, 用 GX Developer 设置参数的连接目标的设置 (参阅 5.5 节) 有效。

7.3 接收控制方法

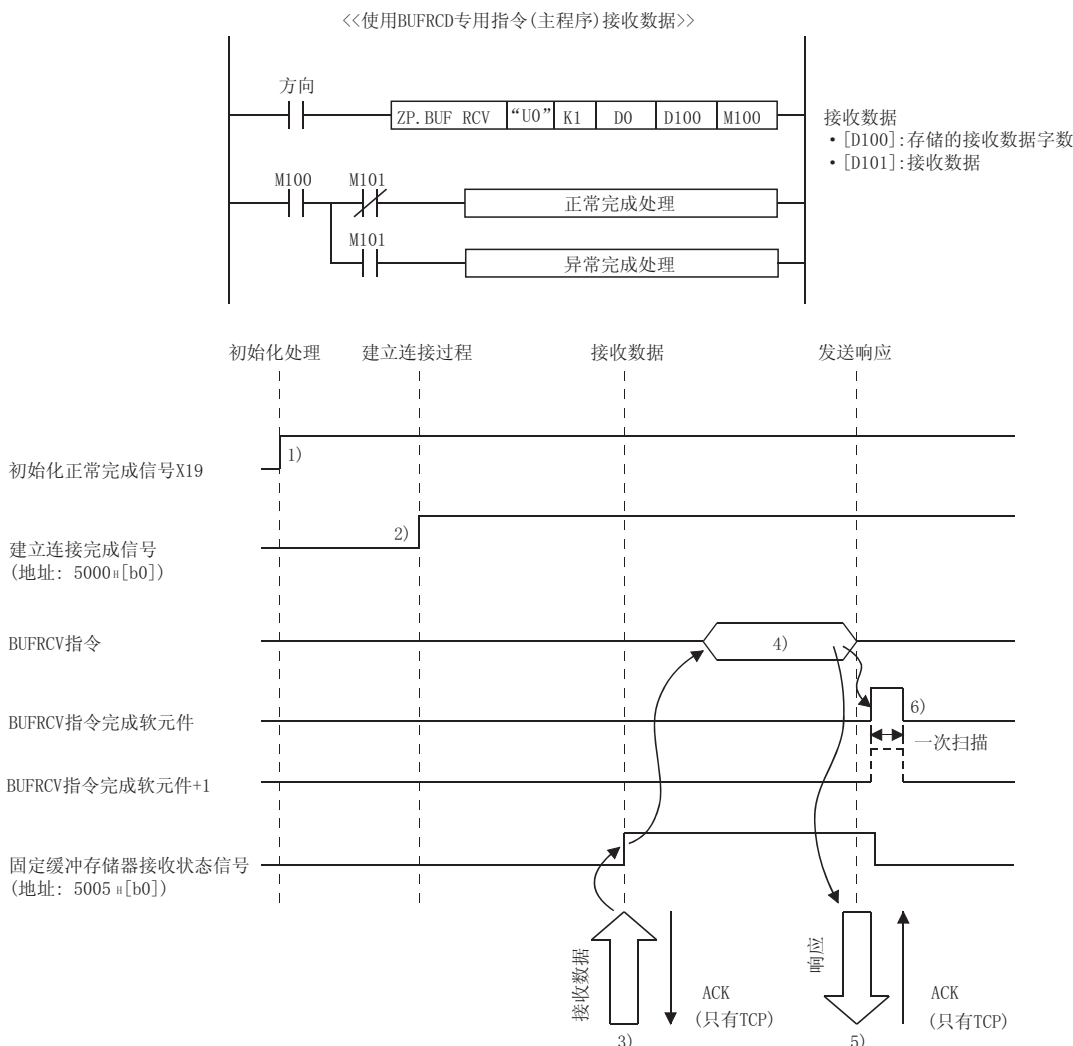
本节说明当以太网模块从外部设备接收数据时的控制方法。

固定缓冲存储器通讯使用下列接收处理方法：

- 用主程序接收处理(专用指令：BUFRCV)
：参阅第 7.3.1 节
- 用中断程序接收处理(专用指令：BUFRCVS)
：参阅第 7.3.2 节

7.3.1 用主程序接收(专用指令：BUFRCV)

本节用 1 号固定缓冲存储器和与 1 号连接相应的区域的例子，说明用主程序执行接收的过程。



- 1) 确认初始化处理正常完成。
- 2) 确认 1 号连接的开放处理正常完成。
- 3) 在从指定外部设备(在开放处理中设置)接收数据时,以太网模块进行下列处理。
 - 将接收数据存储到(1号)固定缓冲存储器区域。
 接收数据长度 : 目标固定地址的起始地址区域
 接收数据 : 从目标固定缓冲存储器+1 起始地址开始的区域
 - 固定缓冲存储器接收状态信号(地址: 5005_H ... b0) : 开
- 4) 执行 BUFRCV 专用指令读取接收数据长度以及接收来自(1号)固定缓冲存储器的数据。
- 5) 当读取接收数据长度和接收数据时,进行下列处理:

正常完成时

 - 将“响应”返回到通讯目标
 - BUFRCV 指令完成软元件 : 开
 - BUFRCV 指令完成软元件+1 : 关
 - BUFRCV 指令完成状态区域(*1) : 0000_H

异常完成时

 - 将“响应”返回到通讯目标
 - BUFRCV 指令完成软元件 : 开
 - BUFRCV 指令完成软元件+1 : 开
 - BUFRCV 指令完成状态区域(*1) : 除 0000_H 之外的其它值
- 6) 结束接收过程。

*1 完成时的状态存储在专用指令的完成状态区域。有关专用指令的详细说明,请参阅第 10 章“专用指令”。

要点	
(1)	在以太网模块开放完成信号(地址: 5000 _H ... 对应位)从关切换到开时,用 GX Developer 设置参数连接的目标的设置(参阅 5.5 节)有效。
(2)	当在固定缓冲存储器接受状态信号存储区域(地址 5005 _H)对应连接的位从关切换到开时,执行 BUFRCV 指令。
(3)	在接收异常数据时,固定缓冲存储器接收完成信号(地址: 5005 _H ... b0)不接通。 此外,数据也不存储在(1号)固定缓冲存储器中。

7.3.2 用中断程序进行接收处理(专用指令: BUFRCVS)

本节说明当使用中断程序时的数据接收过程。

当设置中断程序来处理数据接收时, 中断程序在从外部设备接收数据以及可以读取发往 PLC CPU 的接收数据时启动。

为了启动中断程序, 用 GX Developer 设置参数。

(1) 设置画面

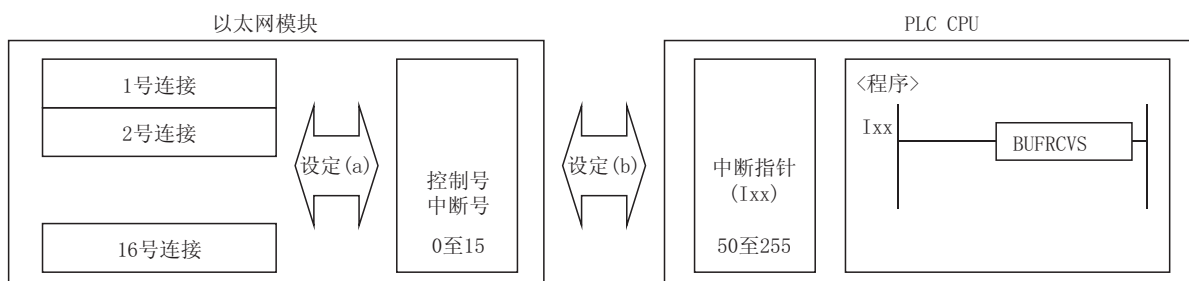
用 GX Developer 设置下列参数来启动中断程序。

- “网络参数” - “中断设置”

当以太网模块向 PLC CPU 发送中断请求时, 设置以太网模块端的控制号(SI)。

- “PLC 参数” - “中断指针设置”

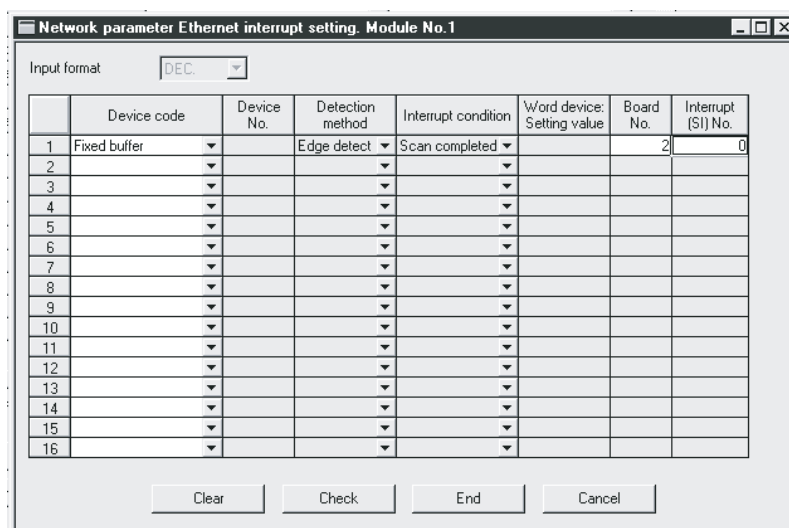
将“网络参数” - “中断设置”中的控制号与中断程序中使用的中断指针(Ixx)相结合。



(a) “网络参数” - “中断设置”

如下解释使用 GX Developer 的中断设置:

通过选择“设置 MNET/10H 以太网网卡号的网络参数” - “中断设置”, 来启动“以太网中断设置”画面



项目名称	设置描述	设置范围/选择
输入格式	选择每个数据的输入格式。	<ul style="list-style-type: none"> • 十进制 • 十六进制
软元件代码	选择软元件代码。	<ul style="list-style-type: none"> • 固定缓冲存储器 • RECV 指令
软元件号	-	-
检测方法	-	(边沿检测)
中断条件	-	(扫描完成)
字软元件设置值	-	-
板号	设置中断程序中使用的通道号/连接号。	<ul style="list-style-type: none"> • RECVS 指令 1-8 • BUFRCVS 指令 1-16
中断(SI)号	设置中断号。	0 至 15

- 1) 输入格式
 - 选择每个设置项目的输入格式(十进制或十六进制)。
 - 2) 软元件代码
 - 选择“固定缓冲存储器”。
 - 当接收数据存储于开放后面的“3)板号”中设置的端口的连接的固定缓冲存储器中时，中断程序开始启动。
 - 3) 板号
 - 设置初始化中断程序启动的固定缓冲存储器的连接号(1-16)。
 - 4) 中断(SI)号
 - 当以太网模块向 PLC CPU 发出中断请求时，设置在以太网模块一端的中断控制号(0-15)。
 - 设置不会与其他固定缓冲存储器通讯和 RECV 指令的中断号相重叠的专用的中断(SI)号。
- * 用户可以任意分配中断号(SI) (0-15) 来接收使用多达 16 个固定缓冲存储器的通讯数据、以及通过中断程序使用 RECV 指令接收数据。
用户必须管理分配给每个数据接收方法的中断(SI)号。
(例如)
用固定缓冲存储器通讯接收数据时，分配与目标数据通讯连接同样的中断(SI)号。
用 RECV 指令接收数据时，分配没有分配给固定缓冲存储器通讯的中断(SI)号。

备注

以上没有提及的其他项目不需要用户在“以太网中断设置”画面上进行设置。
上表中所示的与检测方法和事件状态有关的项目的设置值是自动显示的。

要点

<p>为了启动中断程序，需要进行“设置以太网中断网络参数”和“PLC 参数”-“智能功能模块中断指针设置”。</p>
--

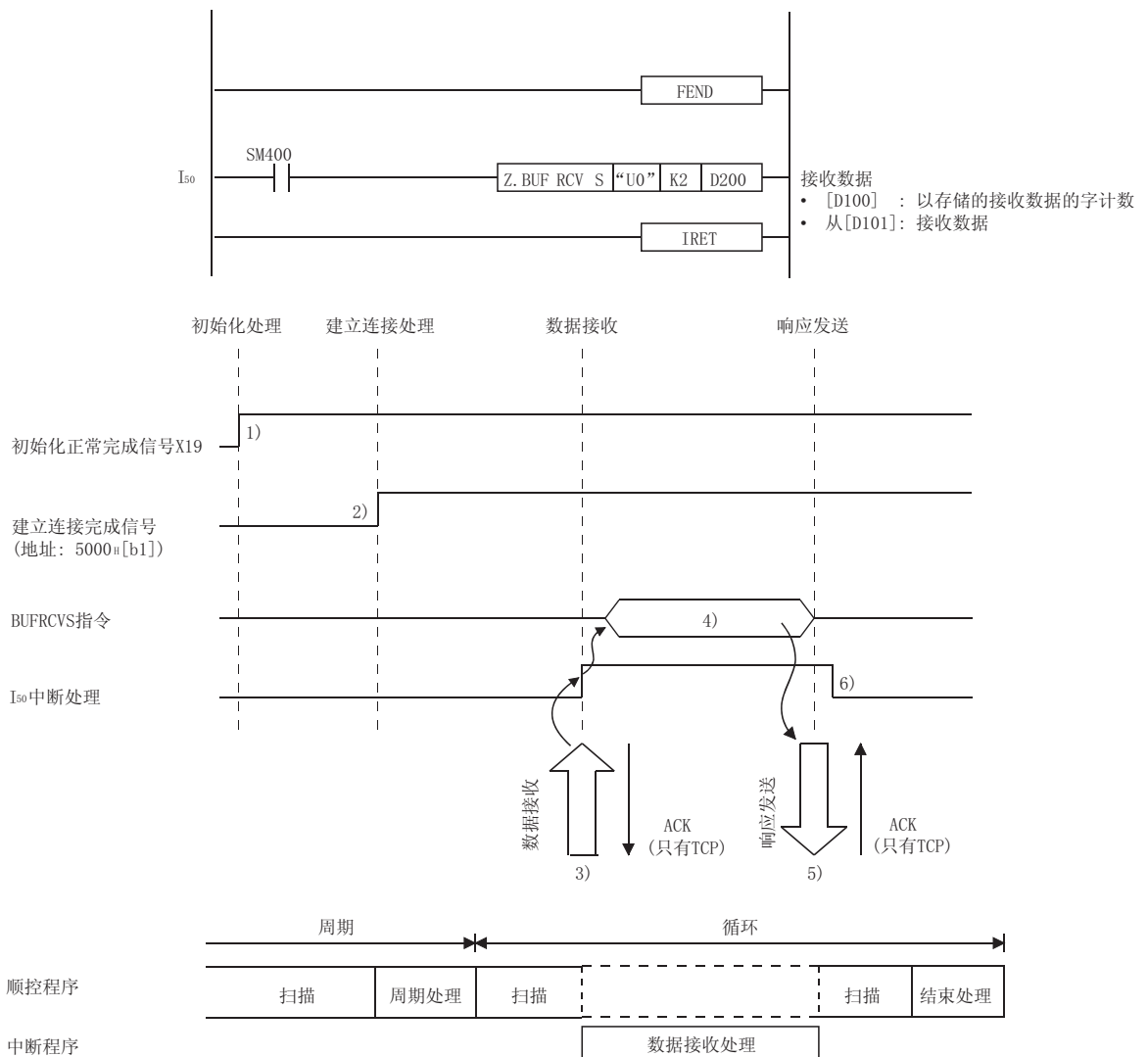
要点
 为启动中断程序，需要“设置以太网中断网络参数”和“PLC 参数”-“智能功能模块中断指针设置”。

(2) 控制方法

用 2 号固定缓冲存储器以及 2 号连接的相应区域为例，来说明中断程序启动时的控制方法。

* 使用下述中断程序的例子，在(1)所示的显示画面中说明通过 GX Developer 对读取接收数据进行的中断设置。

<<用BUFRCVS专用指令(中断程序)接收数据>>



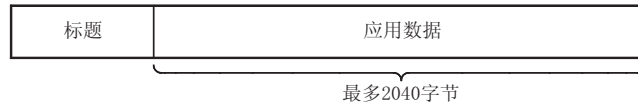
- 1) 确认初始化过程正常完成。
 - 2) 确认 2 号连接的开放过程正常完成。
 - 3) 从专用外部设备(在开放中设置)接收数据后,以太网模块进行下列处理:
 - 将接收数据存储到 2 号固定缓冲存储器区域
接收数据长度 : 目标固定地址的起始地址区域
接收数据 : 从固定缓冲存储器+1 目标起始地址开始的区域
 - 固定缓冲存储器接收状态信号(地址: 5005_H ... b1) : 开
 - 请求 PLC CPU 启动中断程序。
 - 4) 中断程序启动。
执行 BUFRCVS 专用指令从 2 号固定缓冲存储器中读接收数据长度和接收数据。
 - 5) 读接收数据长度和接收数据时,进行下列处理。
 - 正常完成时
 - 将“响应”返回到目标。
 - PLC CPU 出错标记(SM0) (*1) : 关
 - 在异常完成时
 - PLC CPU 出错标记(SM0) (*1) : 开
 - PLC CPU 出错代码(SD0) (*1) : 出错代码
 - 6) 中断程序执行结束,主程序开始重新启动。
- *1 有关 PLC CPU 出错标记(SM0)和出错代码(SD0)的信息,请参阅 PLC CPU 手册。

备注

若要启动中断程序,应在主程序中创建中断允许/禁止程序。
使用指令为 EI 和 DI。

7.4 数据格式

当以太网模块和外部设备进行数据通讯时，使用下面的数据格式：
通讯数据包括“标题”和“应用数据”，如下所示：



7.4.1 标题

使用 TCP/IP 或 UDP/IP 协议的标题。在以太网模块情况下，以太网模块能添加和删除标题。因此用户无须对其进行设置。

(标题部分的详细大小)

1) 使用 TCP/IP 协议时

以太网 14个字节	IP 20个字节	TCP 20个字节
--------------	-------------	--------------

2) 使用 UDP/IP 协议时

以太网 14个字节	IP 20个字节	UDP 8个字节
--------------	-------------	-------------

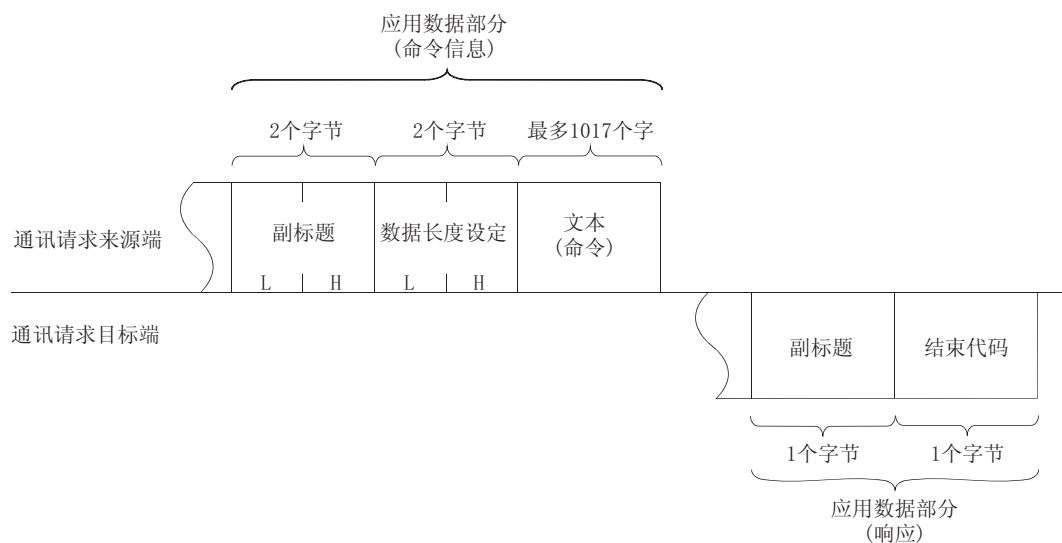
7.4.2 应用数据

如下所示，应用数据中的数据代码可以以二进制或 ASCII 代码来表示。二进制代码和 ASCII 码之间的切换是用 GX Developer 来进行的，如下所示：

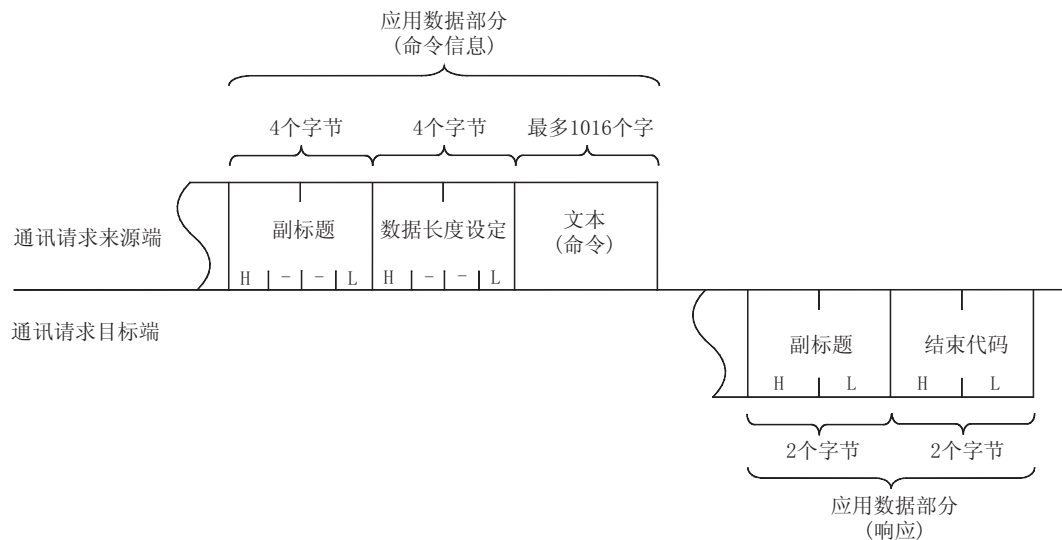
[GX Developer] - [网络参数] - [操作设置] - [通讯数据代码]
 详细说明见 4.7 节“操作设置”。

(1) 格式

(a) 用二进制代码通讯



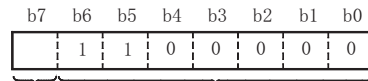
(b) 用 ASCII 代码通讯



(2) 副标题

副标题格式如下所示:

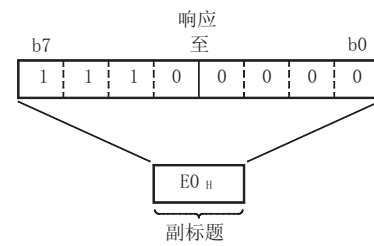
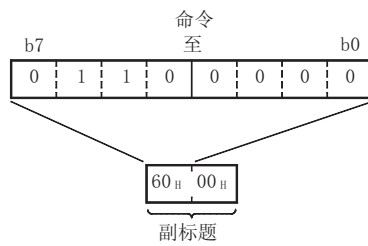
由于以太网模块添加或删除副标题, 因此在使用以太网模块时用户无需设置副标题。



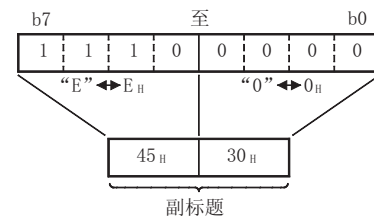
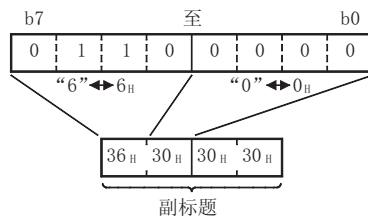
命令/响应类型
(图中指示的位状态显示正在用固定缓冲存储器进行通讯)

命令/响应标记
命令...0
响应...1

用二进制代码通讯



用ASCII代码通讯



(3) 数据长度设置

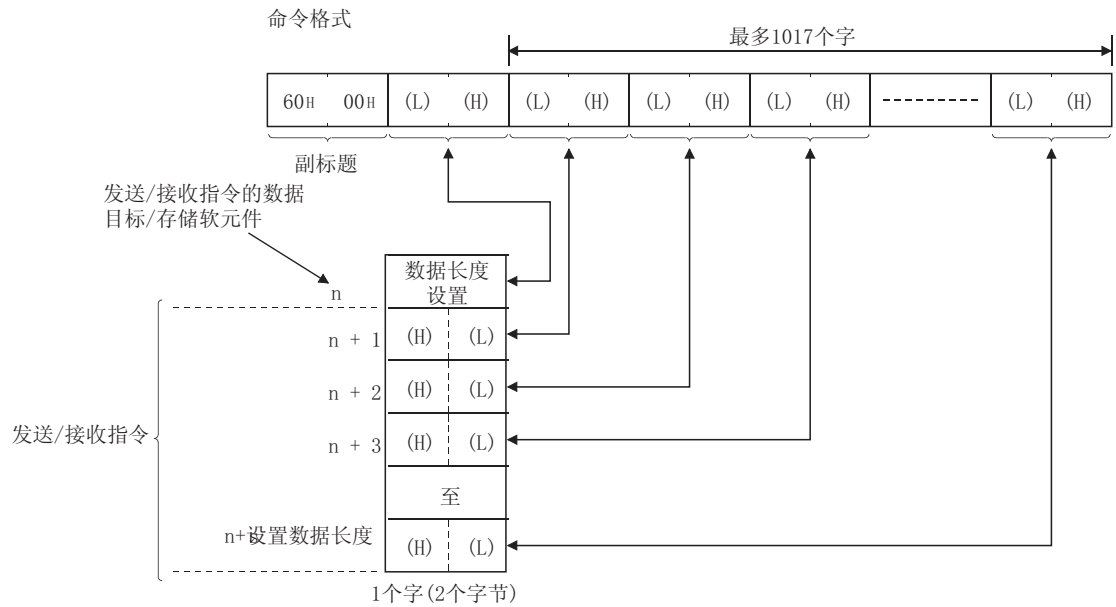
该数值表示文本(命令)部分的数据量。

要点
数据长度可以在下列范围内指定： • 使用二进制代码通讯：最多 1017 个字 • 使用 ASCII 代码通讯：最多 508 个字(*1) *1 由于数据是作为 ASCII 数据发送/接收的，通讯数据量大约为使用二进制代码时数据量的一半。

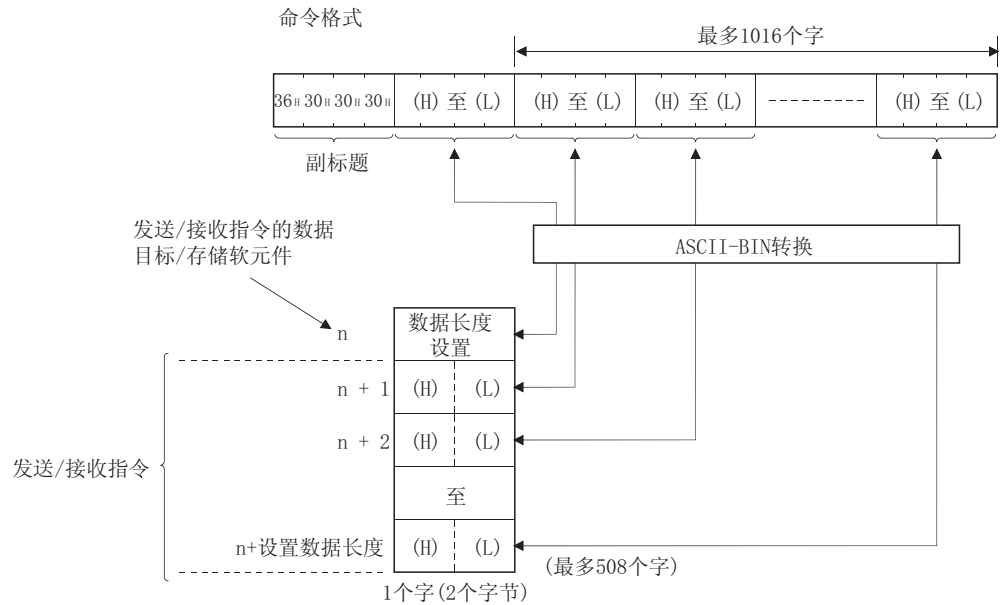
(4) 文本(命令)

下面是使用固定缓冲存储器进行通讯时的命令/响应格式：

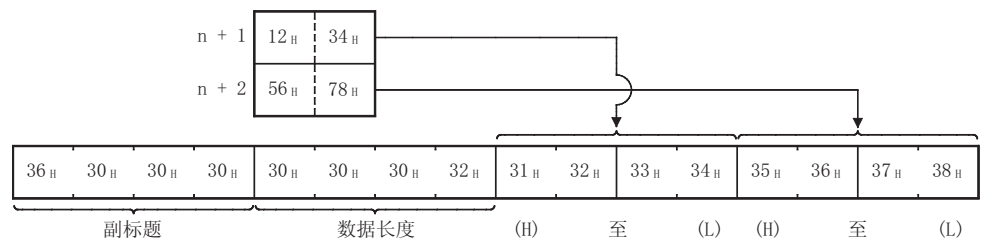
(a) 使用二进制代码通讯



(b) 使用 ASCII 代码通讯



(举例)



(5) 结束代码

有关使用固定缓冲存储器进行通讯时，有关添加结束代码到响应中的详细介绍，请参阅 11.3.1 节。

结束代码存储在 BUFSND 和 BUFRCV 指令的完成状态区域(在控制数据中)，以及缓冲存储器的通讯状态存储区域。

7.5 编程

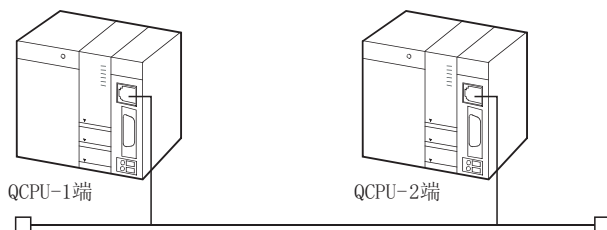
本节说明在以太网模块和外部设备使用固定缓冲存储器进行通讯时，创建程序的方法以及有顺序的控制方法。

7.5.1 创建程序时的注意事项

- (1) 为使用固定缓冲存储器进行通讯，一定要完成初始化过程和开放过程。
- (2) 在以太网模块开放完成信号(地址: 5000_H ... 对应位)从关切换到开时，参数设置的内容已经装载到以太网模块中。
- (3) 用有顺序的控制方法进行通讯的专用指令指定(存储)以字符数表示的数据长度。如果发送数据的长度在数据发送时超过范围，出现通讯出错信号，则无法发送数据。
- (4) 用下列专用程序进行固定缓冲存储器通讯。
 - 数据发送 : BUFSND 指令
 - 数据接收 : BUFRCV 指令(用于主程序)
: BUFRCVS 指令(用于中断程序)有关专用程序的详细说明，参阅第 10 章“专用指令”部分。
- (5) 使用 UDP 打开连接时需要注意以下方面。
 - 在发送/接收数据前，通过对参数设置区的通讯地址设置区中的设置值进行修改可以切换外部设备。因此，数据可以连续的发送到多个外部设备。执行发送/接收时，一定要在外部设备间进行切换以保证不会发生通讯问题。
- (6) 当从同一连接读接收到的数据时，读主程序接收的数据和读中断程序接收的数据无法同时进行。
只使用上述程序读接收数据。
* 当用 GX Developer 设置对中断程序读接收数据时，无法读用主程序接收的数据。
- (7) 关于数据(指令)传送，下一个数据(指令)应该在上一个数据传送(指令)的数据通讯(比如在接收响应后)完成后。

7.5.2 固定缓冲存储器通讯程序范例(有顺序的控制方法)

本节讲述了使用固定缓冲存储器的外部设备进行数据通讯(有顺序的控制方法)的编程方法。



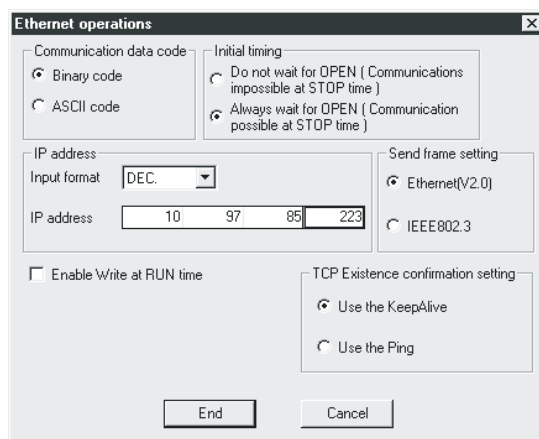
(1) 程序执行环境范例

(a) 发送程序(QCPU-1 站端)

- 1) 1号连接用于固定缓冲存储器的发送。
- 2) 假定通讯参数设置为如第5.6.1节中描述的“主动开放/关闭处理”。
- 3) 1号固定缓冲存储器发送数据
: 存储在 D300-D303 中
- 4) 1号固定缓冲存储器发送指令正常完成软元件
: M300
- 5) 1号固定缓冲存储器发送指令异常完成软元件
: M301
- 6) 1号固定缓冲存储器发送指令完成状态
: D3001

(b) 接收程序(QCPU-2 站端)

- 1) 1号连接用于处理在主程序中接收的固定缓冲存储器。
- 2) 2号连接用于处理在中断程序中接收的固定缓冲存储器。
- 3) 以太网模块安装在主基板的“0”槽中。
- 4) 假设已经用 GX Developer 设置“设置 MNET/10H 以太网卡号的网络参数”为如下参数：(见第 4.6 部分)
 - 网络类型 : 以太网
 - 首 I/O 号 : 0000
 - 网络号 : 1
 - 组号 : 1
 - 站号 : 2
- 5) 假设已经用 GX Developer 设置“操作设置”为如下参数：



本站 IP 地址：0A.61.55.DFH (10.97.85.223)

6) 假设已经用 GX Developer 设置“开放设置”为如下参数:

	Protocol	Open system	Fixed buffer	Fixed buffer communication	Pairing open	Existence confirmation	Local station Port No.	Destination IP address	Dest. Port No.
1	TCP	Unpassive	Receive	Procedure exist	No pairs	No confirm	2000		
2	TCP	Unpassive	Receive	Procedure exist	No pairs	No confirm	3000		
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									

End Cancel

1号连接本地站端口号 : 2000H
(用于主程序)

2号连接本地站端口号 : 3000H
(用于中断程序)

- 7) 1号固定缓冲存储器接收数据
: 存储在 D500 至 D503 中
- 8) 2号固定缓冲存储器接收数据
: 存储在 D700 至 D703 中
- 9) 1号固定缓冲存储器接收指令完成软元件
: M500
- 10) 1号固定缓冲存储器接收指令异常完成软元件
: M501
- 11) 1号固定缓冲存储器发送指令完成状态
: D5001
- 12) 1号固定缓冲存储器接收状态信号
: M40
- 13) 假设已经如第 7.3.2 节“用中断程序进行数据接收处理”中的画面设置中所叙述, 用 GX Developer 设置启动中断程序的参数。
2号固定缓冲存储器中断指针 r
: I50 (SI 0 号)

要点

根据源发送的数据最大长度来确保足够的设备存储, 以避免用于其它用途的设备区被接收数据覆盖。

(2) 程序范例概述

(a) 发送程序(QCPU-1 站端)

- 1) 在用 GX Developer 设置每个参数并写到 PLC CPU 中后, 将 PLC CPU 复位并确认初始化过程的完成。
- 2) 执行 1 号连接(*1)的开放过程(主动开放)
- 3) 用固定缓冲存储器通讯的(步骤存在发送)从 PLC CPU 作数据通讯。
- 4) 在数据发送完成后, 执行 1 号连接(*1)的关闭过程。

*1 用第 5.6.1 部分描述的程序范例执行开放处理/关闭处理的顺控程序的“主动开放处理/关闭处理”。

(b) 接收程序(QCPU-2 站端)

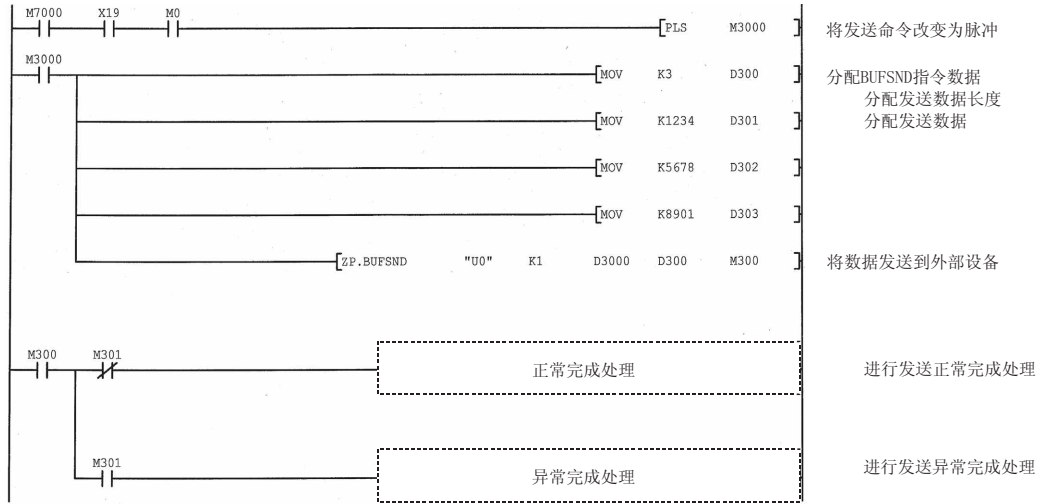
- 1) 在用 GX Developer 设置每个参数并写入 PLC CPU 之后, 将 PLC CPU 复位并确认初始化处理完成。
当初始化处理正常完成时, 1 号连接和 2 号连接等待来自外部设备的主动开放请求。
- 2) 固定缓冲存储器通讯(有顺序发送)对来自外部设备的数据进行通讯。
- 3) 将以太网模块中的相应固定缓冲存储器数据区域中接收的数据读到 PLC CPU。

(发送程序)

开放处理程序

(见第 5.6.1 节)

X19: 初始化正常完成信号
 M0: 1号连接开启完成信号
 1号固定缓冲存储器发送程序



将发送命令改变为脉冲

分配BUFSND指令数据
 分配发送数据长度
 分配发送数据

将数据发送到外部设备

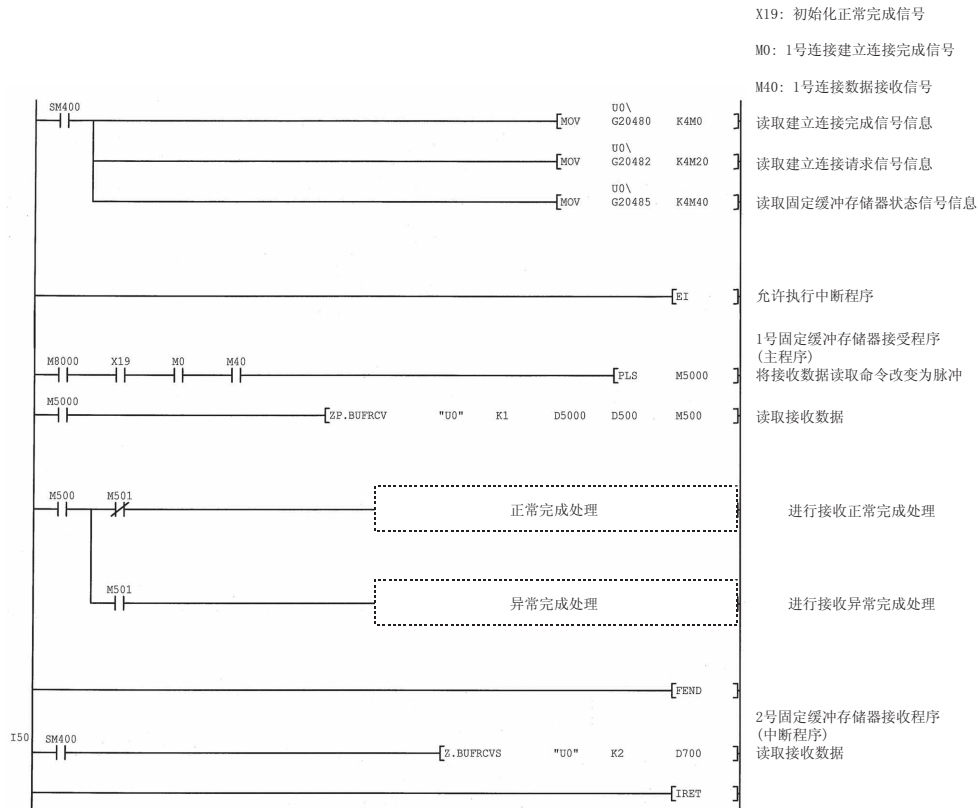
进行发送正常完成处理

进行发送异常完成处理

关闭处理程序

(见第 5.6.1 节)

(接收程序)



8 固定缓冲存储器通讯(无顺序的控制方法)

本章说明 PLC CPU 和外部设备如何使用以太网模块的固定缓冲存储器(无顺序的控制方法)进行通讯。

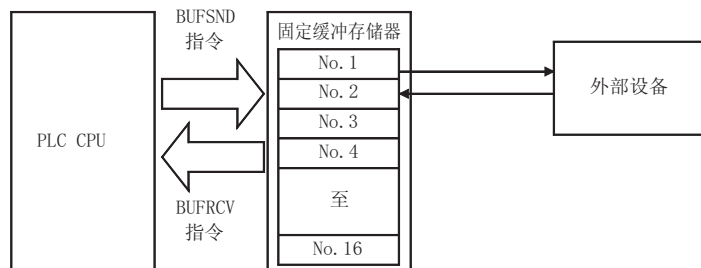
要点
<p>以下各点是当使用固定缓冲存储器进行通讯时，与“有顺序”不相同处的说明：</p> <ol style="list-style-type: none"> 1) 可以发送和接收符合外部设备信息格式的数据。 在发送数据时，副标题、数据长度等不包含在信息的应用数据字段中。只有在固定缓冲存储器中的数据才被发送。在接收数据时，信息中的所有数据，除了标题之外，都存储在固定缓冲存储器里。 2) 不发送数据接收响应。 3) 虽然通讯数据是用 GX Developer 参数码设置的，通讯仍用二进制代码进行。(见 4.7 节“操作设置”。) 4) 每次通讯的最大应用数据区域为 2046 个字节。 5) 应用连接专门用于无顺序固定缓冲存储器的通讯中。 对于有顺序固定缓冲存储器通讯，随机访问缓冲存储器通讯和使用 MC 协议的通讯不能像无顺序固定缓冲存储器通讯那样同时进行。

8.1 控制方法

下面介绍如何使用固定缓冲存储器和无顺序控制方法进行通讯。

在使用固定缓冲存储器的通讯处理中，PLC CPU 和外部设备的数据传送是用无顺序控制方法进行的。

(1) 通讯处理中的数据流如下：



- (2) 数据可以通过下列外部设备进行通讯。
- 以太网上与以太网模块相连接的设备。
 - 通过路由中继功能相连接的设备。(见 5.3 节)

如下图所示, 当使用每个固定缓冲存储器(1 号到 16 号)时, 目标设备和使用条件(发送/接收、有顺序/无顺序等)应在通过以太网模块的连接开放时设置, 以便为每个缓冲存储器设置外部设备。

- (a) 在 TCP/IP 通讯时

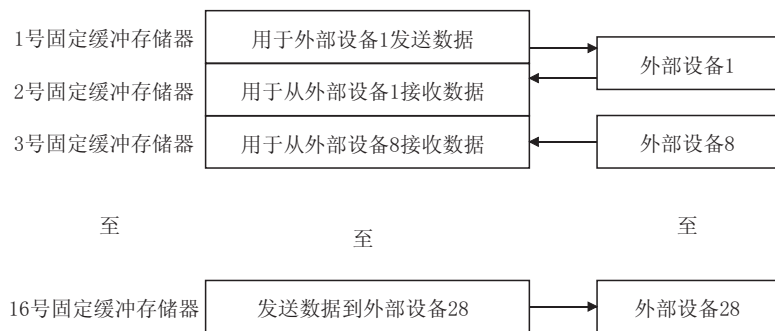
只有当应用连接的开放完成信号关闭时, 才可以改变外部设备。

- (b) 在 UDP/IP 通讯时

不管应用连接的状态是什么, 都可以改变外部设备。

(在通讯地址设置区域内的“目标 IP 地址”和“目标端口号”可以改变。然而, “本地站端口号”不能进行改变。)

在改变外部设备时, 不要使用“成对开放”和“存在确认”功能。



要点

所选的无顺序连接专用于开放处理完成后的固定缓冲存储器发送或接收。

(3) 在数据发送/接收时，以太网模块进行下列处理：

1) 在发送数据时

当 PLC CPU 在顺控程序中执行 BUFSND 专用指令(*1)时，以太网模块将可用固定缓冲存储器(n 号)中的数据发送到与 n 号固定缓冲存储器(*2)相应的通讯地址设置区域所规定的外部设备中(地址:28H-5FH 和 5038H-507FH)。

2) 在接收数据时

如果数据是从与 n 号固定缓冲存储器件(*2)相应的通讯地址设置区域所设置的外部设备中接收的，那么以太网模块处理接收数据。

而且，当以太网模块将接收数据存储在接受处理中的通讯固定缓冲存储器中时，它会更新与 n 号固定缓冲存储器相应的连接信息区域(地址:78H-C7H 和 5820H-586FH)的目标 IP 地址和目标端口号。

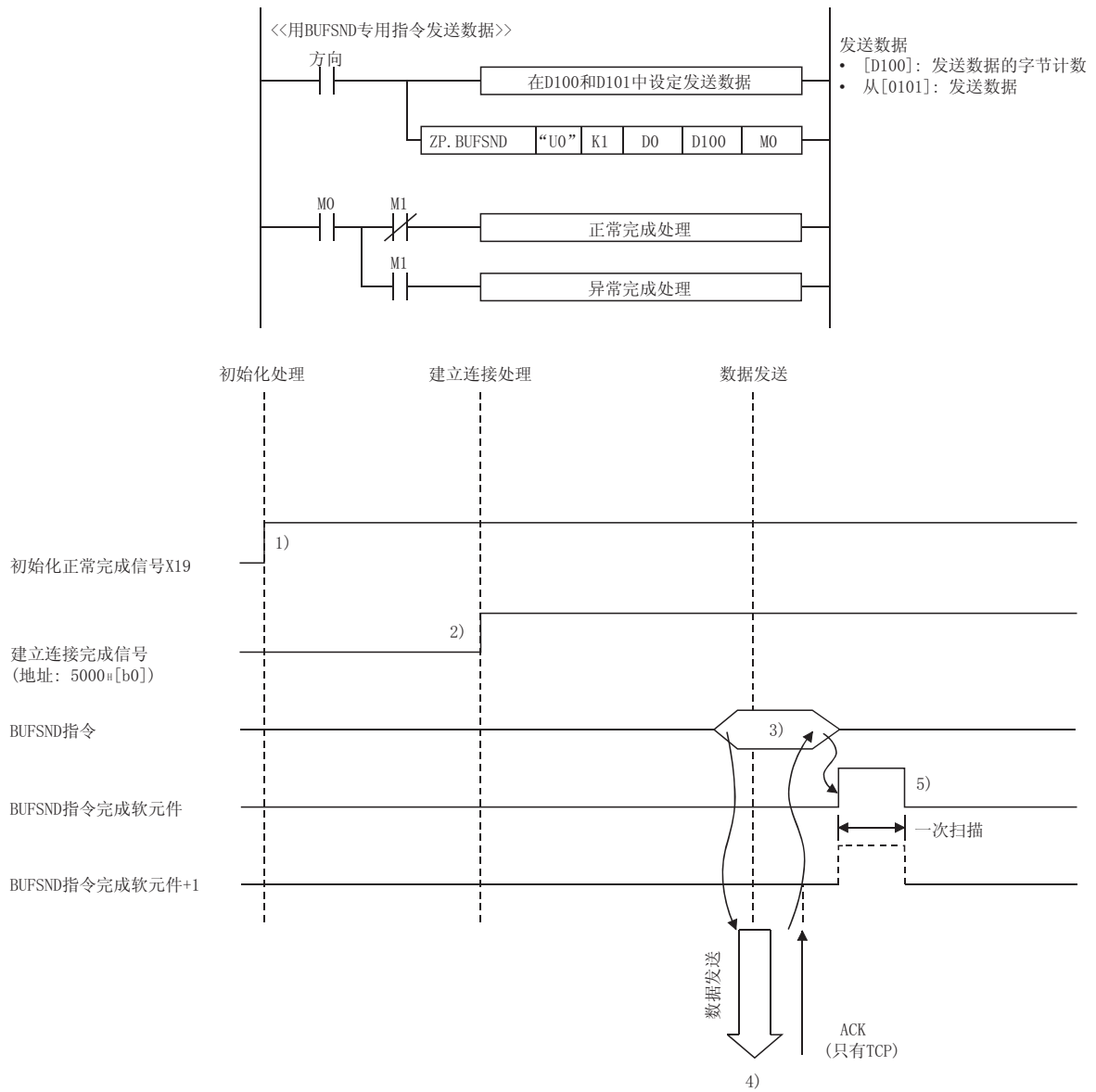
如果从外部设备接收的数据未设置在缓冲存储器连接信息区域，则以太网模块忽略接收数据。

*1 有关专用指令的详细情况，请参见第 10 章“专用指令”部分。

*2 当 TCP/IP 非被动开放，与外部设备进行通讯的数据通过存储在缓冲存储器的连接信息区。

8.2 发送控制方法

本节用 1 号固定缓冲存储器和与 1 号连接相应的区域作为例子，说明当数据从以太网模块发送到外部设备时的控制方法。



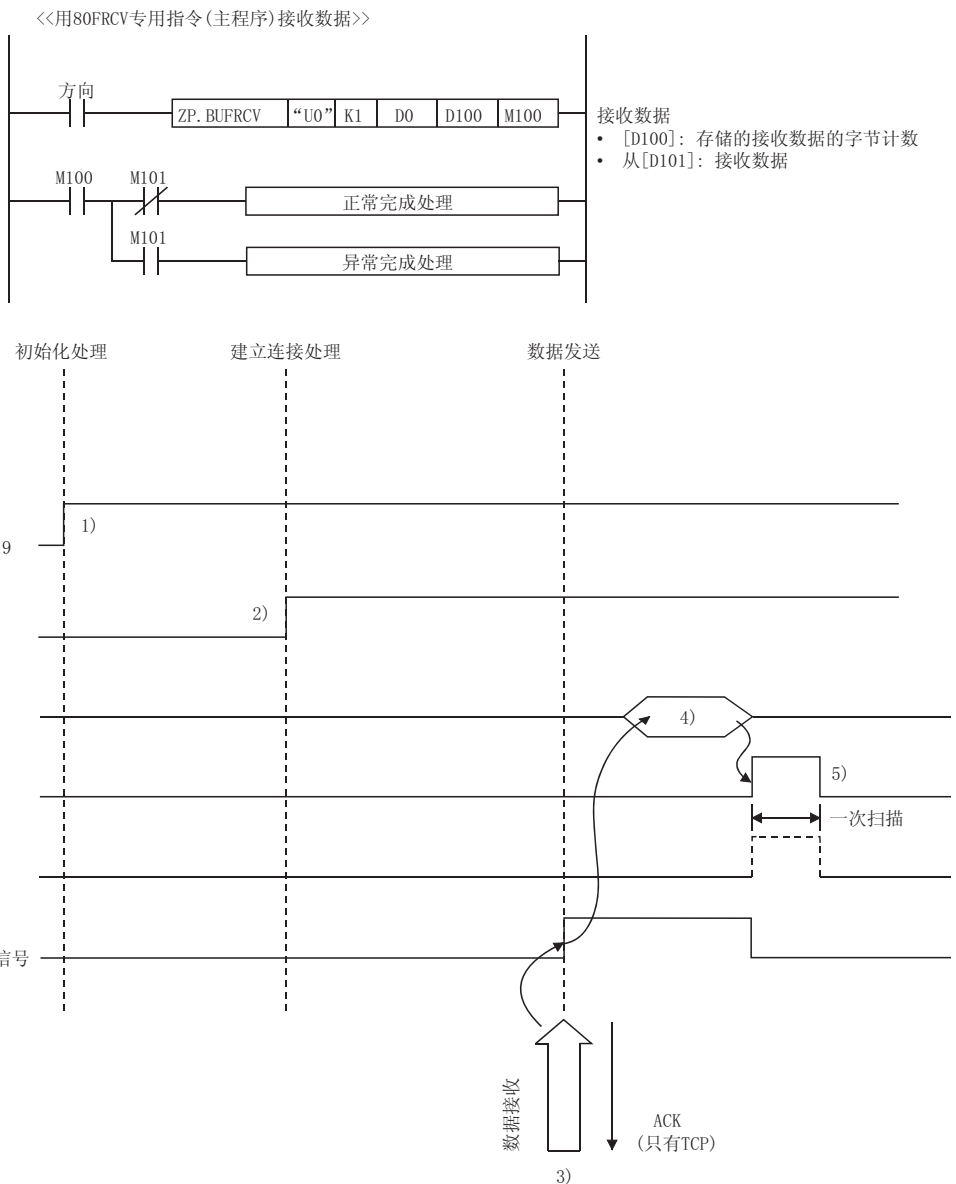
8.3 接收控制方法

本节说明当以太网模块从外部设备接收数据时的控制方法。
 固定缓冲存储器通讯采用下列接收处理方法：

- 用主程序(专用指令:BUFRCV)处理接收
 : 参阅第 8.3.1 节
- 用中断程序(专用指令:BUFRCVS)处理接收
 : 参阅第 8.3.2 节

8.3.1 用主程序接收(专用指令: BUFRCV)

本节通过 1 号固定缓冲存储器与与 1 号连接相应的区域的例子, 说明如何用主程序进行接收。



- 1) 确认初始化处理的正常完成。
- 2) 确认 1 号连接的开放处理正常完成。
- 3) 如果从指定外部设备(设置于开放处理中)接收到数据,以以太网进行下列处理。
 - 将接收数据存储到固定缓冲存储器(1号)区域中。
从目标固定缓冲存储器+1 起始地址开始的区域
 - 将数据长度存储到目标固定地址(*1)的 1 起始地址区域中。
 - 固定缓冲存储器接收状态信号(地址: 5005H ... b0) : 开
- *1 接收数据长度由字节数来表示。
当接收到数据字节数时,最后接收的数据存储在最后数据存储区域的低位字节中。(高位字节成为非常数数值)
- 4) 执行 BUFRCV 专用指令,并从固定缓冲存储器(1号)中读取接收数据长度和接收数据。
- 5) 结束接收处理。

当正常完成时

- BUFRCV 指令完成软元件 : 0N
- BUFRCV 指令完成软元件+1 : 0FF
- BUFRCV 指令完成状态区域(*2) : 0000H

当异常完成时

- BUFRCV 指令完成软元件 : 0N
- BUFRCV 指令完成软元件+1 : 0N
- BUFRCV 指令完成状态区域(*2) : 除 0000H 之外的数值

*2 完成状态存储在专用指令的完成状态区域中。有关专用指令的详细说明,请参见第 10 章“专用指令”部分。

要点
(1) 在以太网模块开放完成信号(地址: 5005H ... 相应位)从关切换到开时,用 GX Developer 设置的参数连接目标设置(见第 5.5 节)有效。 (2) 当固定缓冲存储器接收状态信号存储区域(地址: 5005H)中与缓冲存储器相应的连接的位从关切换到开时,执行 BUFRCV 指令。 (3) 在接收异常数据时,不接通固定缓冲存储器的接收完成信号(地址: 5005H ... b0)。 此外,数据不存储在固定缓冲存储器(1号)区域中。

8.3.2 用中断程序接收(专用指令: BUFRCVS)

本节说明使用中断程序时的接收处理。

当安装中断程序进行接收处理时, 中断程序在从外部设备接收数据时, 启动且读取发送给 PLC CPU 的接收数据。

为了启动中断程序, 用 GX Developer 设置参数。

(1) 设置画面

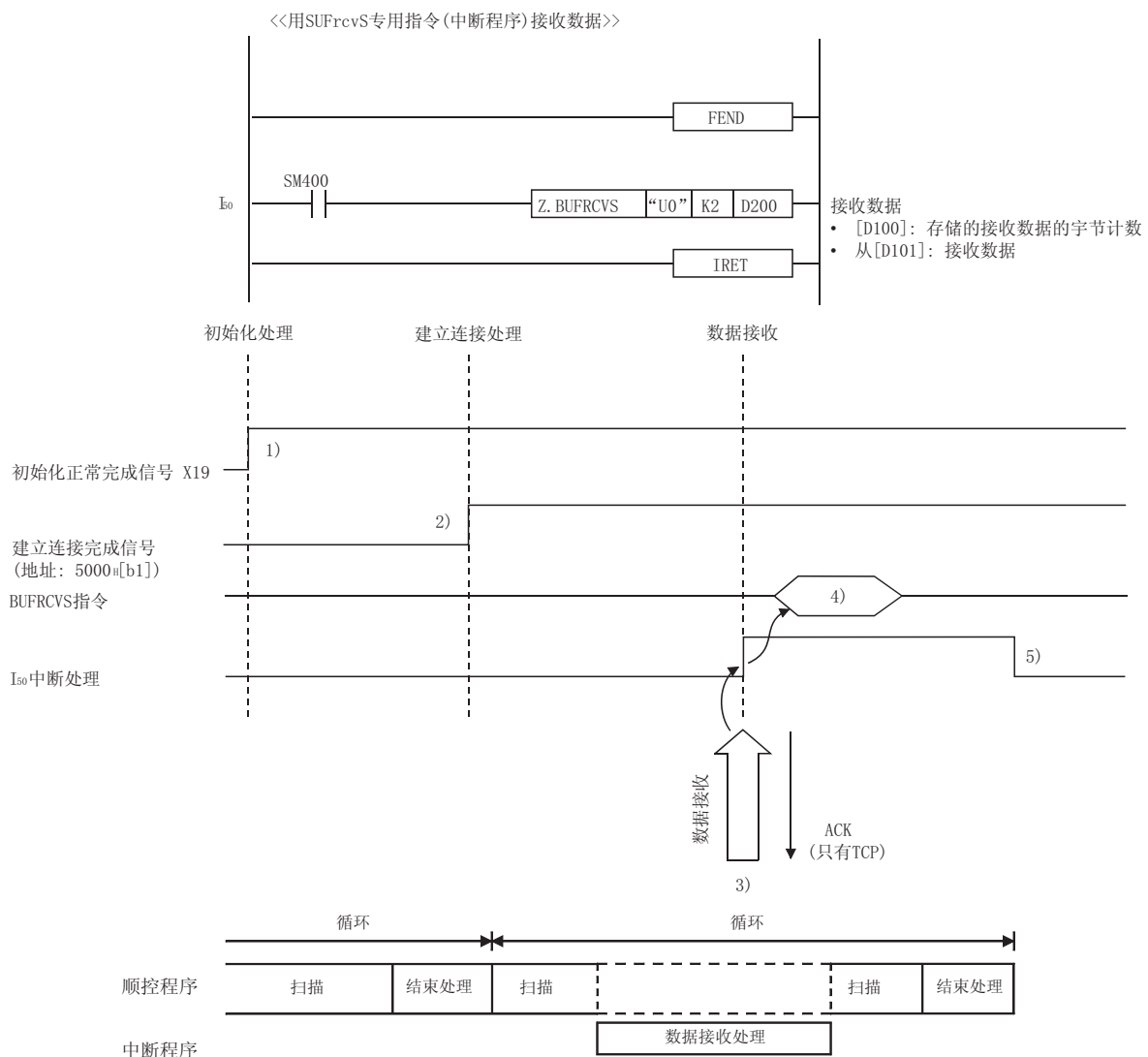
用 GX Developer 设置下列参数, 以启动中断程序。

设置画面和设置方法与固定缓冲存储器的通讯(有顺序)相同。参阅 7.3.2 节的 (1)。

(2) 控制方法

用 2 号固定缓冲存储器与 2 号连接相应的区域为例, 说明在中断顺控程序启动时的控制方法。

*用 GX Developer 进行的中断设置以使用中断程序读接收数据的例子, 见第 7.3.2 节 (1) 中所示的画面显示中的图示说明。



- 1) 确认初始化处理的正常完成。
- 2) 确认 2 号连接的开放处理正常完成。
- 3) 当从专用外部设备(在开放处理中设置)接收到数据时, 以太网模块进行下列处理:
 - 在固定缓冲存储器(2 号)区域存储接收数据。
(从目标固定缓冲存储器+1 起始地址开始的区域)
 - 在目标固定地址件(*1)的起始地址区域中存储数据长度
 - 要求 PLC CPU 启动中断程序。

*1 接收数据长度以字节数表示。
当接收到的数据字节数为奇数时, 最后接收的数据存储在最后数据存储区域的下位字节中。(高位字节成为一个非常数的数值。)
- 4) 中断程序启动。
执行专用 BUFRCVS 指令, 且从 2 号固定缓冲存储器中读取接收数据长度和接收数据。
在正常完成时
 - PLC CPU 出错标记(SM0) (*2) : 关
 在异常完成时
 - PLC CPU 出错标记(SM0) (*2) : 开
 - PLC CPU 出错标记(SD0) (*2) : 错误代码

*2 有关 PLC CPU 出错标记(SM0) 出错代码(SD0)的详细说明, 请参阅所使用的 PLC CPU 的手册。
- 5) 中断程序执行结束, 主程序开始执行。

备注

- (1) 启动中断程序, 用 EI 和 DI 指令在主程序中创建中断允许/禁止程序。
- (2) 用中断程序读取接收数据的编程和通过固定缓冲存储器所进行的通讯(有顺序)相同。编程时请参考第 7.5.2 节中所举的程序例子。

8.4 数据格式

当在以太网模块和外部设备之间进行数据通讯时，使用下述数据格式。
通讯数据包括“标题”和“应用数据”，如下所示：



(1) 标题

使用 TCP/IP 或 UDP/IP 的标题。在使用以太网模块时，以太网模块添加和删除标题。因此，用户无须进行设置。

(标题部分的详细大小)

1) 在使用 TCP/IP 协议时



2) 在使用 UDP/IP 协议时



(2) 应用数据

应用数据中的数据代码用二进制代码表示。

无论如何设置通讯数据(见第 4.7 节)，都使用二进制代码进行通讯。



备注

用无顺序控制方法中的固定缓冲存储器为通讯添加的副标题和数据长度，在有顺序控制方法的通讯中不提供。所有数据作为有效文本来处理。

8.5 使用 UDP/IP 进行同步广播

当 UDP/IP 用于无顺序固定缓冲存储器通讯时，数据可以同步广播到所有同一以太网中装有以太网模块的站点中。这样就可以将同样的数据写入所有的站点。

要点

- (1) 在同步广播时，如果不需要同时传送接收信息，则一有机会，连接到同一以太网的外部设备就会将其删除。
- (2) 用户需要确定通过同步广播进行的传送和接收专用的端口号，此后，同步广播可以通过分配这些端口号来进行。

8.5.1 用同步广播发送

可以通过将外部设备 IP 地址设置为 FFFFFFFFH 以及进行开放处理来发送。通过同步广播进行数据发送，以太网设置需要的目标 IP 地址到 FFFFFFFFH 并通过以太网发送数据。

(1) 设置画面

该部分说明通过同步广播进行发送的“开放设置”。

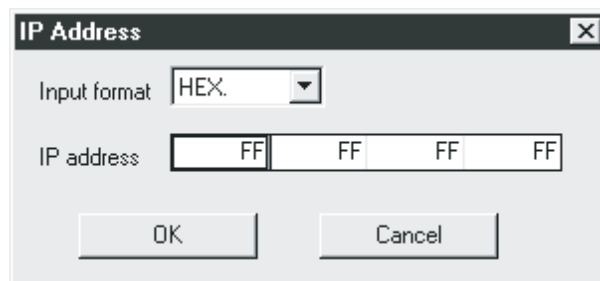
- 使用第 15 号连接。
- 以太网模块端口号为 0800H。

	Protocol	Open system	Fixed buffer	Fixed buffer communication	Pairing open	Existence confirmation	Local station Port No.	Destination IP address	Dest. Port No.
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15	UDP		Send	No procedure	No pairs	No confirm	0800	Simultan	0801
16									

End Cancel

- (a) 协议
选择“UDP/IP”协议。
- (b) 开放系统
无需设置。
- (c) 固定缓冲存储器
选择“发送”。

- (d) 固定缓冲存储器通讯
选择“无顺序”。
- (e) 成对开放
选择“不成对”。
- (f) 存在确认
选择“不确认”。
- (g) 本地站端口号
 - 1) 用十六进制设置以太网模块的端口号。
 - 2) 根据向网络管理员询问，在 401_H 到 1387_H 以及 138B_H 到 FFFE_H 的范围内分配设置值。选择其他端口不用的端口号。
- (h) 目标 IP 地址
设置为“FFFFFFF_H”。



- (i) 目标端口号
 - 1) 用十六进制设置外部设备端的端口号。
 - 2) 根据向网络管理员询问，在 401_H 到 FFFE_H 的范围内分配。
 - 3) 检查第 8.1(2) 中显示的通讯范围。
- (2) 发送控制方法
发送控制方法与使用固定缓冲存储器进行的正常无顺序通讯相同。详见第 8.2 节。

8.5.2 用同步广播接收

将发送数据到以太网模块的外部设备 IP 地址设置为 FFFFFFFFH 以及设置端口号为 FFFFH，然后进行开放处理，则数据可以通过同步广播接收。

(1) 设置画面

该部分讲述了通过同步广播接收的“开放设置”。

- 使用第 16 号连接。
- 以太网模块端口号为 0801H。

	Protocol	Open system	Fixed buffer	Fixed buffer communication	Pairing open	Existence confirmation	Local station Port No.	Destination IP address	Dest. Port No.
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16	UDP		Receive	No procedure	No pairs	No confirm	0801	Simultan	FFFF

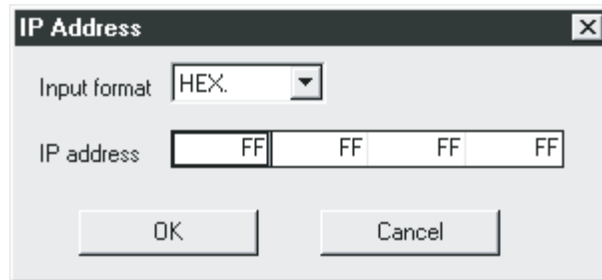
- 协议
选择“UDP/IP”协议。
- 开放系统
无需设置。
- 固定缓冲存储器
选择“接收”。
- 固定缓冲存储器通讯
选择“无顺序”。
- 成对开放
选择“不成对”。
- 存在确认
选择“不确认”。

(g) 本地站端口号

- 1) 用十六进制设置以太网模块的端口号。
- 2) 在 401_H 到 1387_H 和 138B_H 到 FFFE_H 的范围内分配设置值。
- 3) 如果要处理从特定的外部设备处接收的数据, 应根据向网络管理员咨询来分配用户定义端口号。

(h) 目标 IP 地址

设置“FFFFFFFF_H”。



(i) 目标端口号

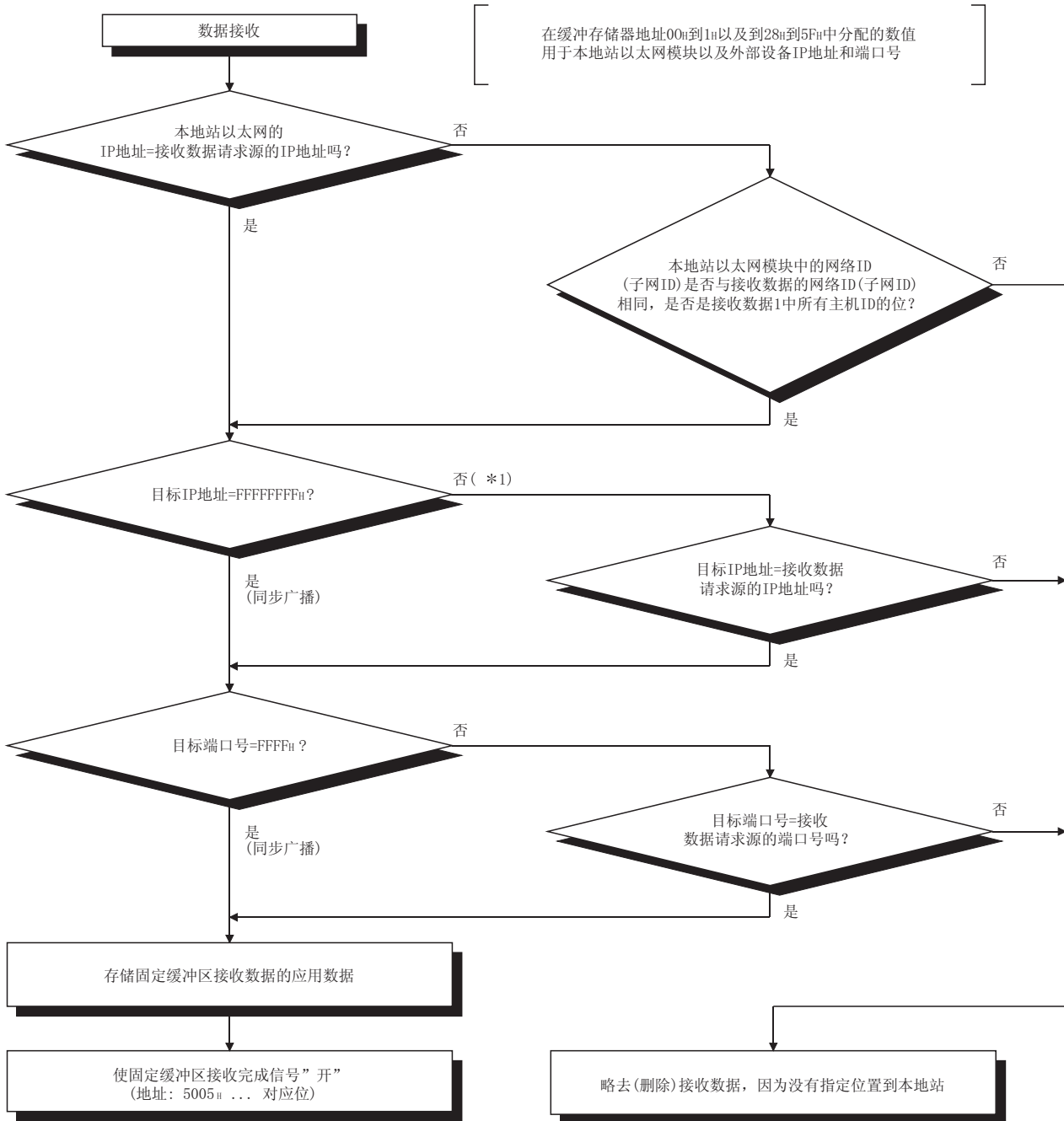
设置为“FFFF_H”。

(2) 接收控制方法

接收控制方法与使用固定缓冲存储器的正常无顺序通讯相同。详见第 8.3 节。

备注

下图为当采用无顺序控制方法通过同步广播接收数据时，以太网模块的内部处理流程略图。



*1 如果指示接收数据请求目标IP地址的主机ID范围中的位都是“1”，那么按“是”来处理。

8.5.3 使用同步广播功能时的注意事项

在使用固定缓冲存储器采用无顺序通讯方式进行同步广播的注意事项如下。

- (1) 用户需要确定同步广播的发送和接收专用端口号，此后就可以通过分配这些端口号来进行同步广播。
- (2) 将同步广播的发送信息发送到所有连接以太网模块的同一以太网上的外部设备中。
如果不需要用同步广播接收的信息，连接到同一以太网的外部设备应将其删除。
* 如果不需要这些信息的话，外部设备应当将其删除。此外，即使外部设备是接收信息发往的站点，也不返回响应。以太网模块会自动进行这些处理。
- (3) 每次发送/接收，在应用数据字段中最多可以处理 2046 个字节的数据。
当需要发送/接收超过 2046 个字节的数据时，将发送源文件分割成几部分。
- (4) 在进行同步广播时，将相应连接的“开放设置”中的“存在确认”设置为“无确认”。

备注

完成当前接收处理前，以太网模块为操作系统临时在其内部缓冲器中存储接收数据。

如果通过同时广播所接收到的数据超出内部缓冲器(大约 40 字节)的容量，超出的数据会被废弃。

通过缓冲存储器(有顺序)进行的通讯中，以太网模块发送指令信息到外部设备、等待接收响应信息，然后发送下一个指令信息。所以，用户无需考虑以上提及的用于操作系统的内部缓冲器。

8.6 编程

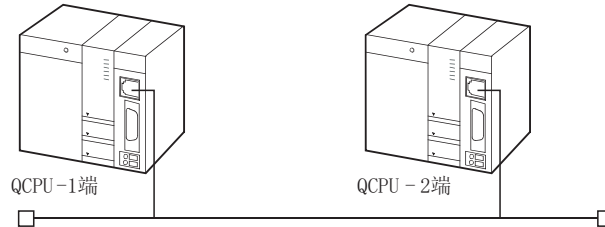
本节说明当以太网模块用固定缓冲存储器和无顺序控制方法与外部设备进行通讯时的编程方法。

8.6.1 创建程序时的注意事项

- (1) 为了用固定缓冲存储器进行通讯，必须完成初始化处理和连接的开放处理。
- (2) 在以太网模块的开放完成信号(地址：5000_H...相应位)从关切换到开时，参数设置内容装载到以太网模块中。
- (3) 以字节计数的数据长度用专用指令为无顺序通讯进行分配和存储。
如果在数据发送时发送数据长度超过此范围，会出现通讯错误，数据无法发送。
- (4) 固定缓冲存储器通讯使用下列专用指令。
 - 数据发送 : BUFSND 指令
 - 数据接收 : BUFRVC 指令(主程序)
: BUFRVCS 指令(中断程序)有关专用指令的详细说明，请参阅第 10 章“专用指令”。
- (5) 在使用 UDP 开放时，应注意下列事项。
 - 在发送/接收数据之前，外部设备可以通过通讯参数设置区域中的通讯地址设置区域的修改设置值来进行更改。因此，数据可以按序发送给多个外部设备。当发送/接收时，注意恰当更改外部设备，以免发生通讯故障。
- (6) 所选的无顺序连接是专用于无顺序固定缓冲存储器发送/接收的。因此，用有顺序控制方法来任意访问缓冲存储器和 MC 协议的固定缓冲存储器的通讯不能与用无顺序控制方法的固定缓冲存储器的通讯同时进行。
- (7) 信息数据长度不用于无顺序通讯。
以太网模块将所接收信息(信息包)的大小存储到接收数据长度存储区域，并打开固定缓冲存储器的接收状态信号(地址：5005_H ...相应位)。
建议使用检查系统，例如在信息的应用数据中包括数据长度和数据类型代码，这样接收方可以识别在应用数据中的字节数和数据类型。
- (8) 当读从同一连接接收的数据时，不能同时读主程序接收的数据和中断程序接收的数据。
只可使用上述一种程序读接收数据。
* 当用 GX Developer 进行中断程序读接收数据的设置时，无法读主程序接收的数据。

8.6.2 固定缓冲存储器通讯程序范例(无顺序的控制方法)

本节说明用固定缓冲存储器的外部设备(通过无顺序的控制方法)进行数据通讯的编程方法。



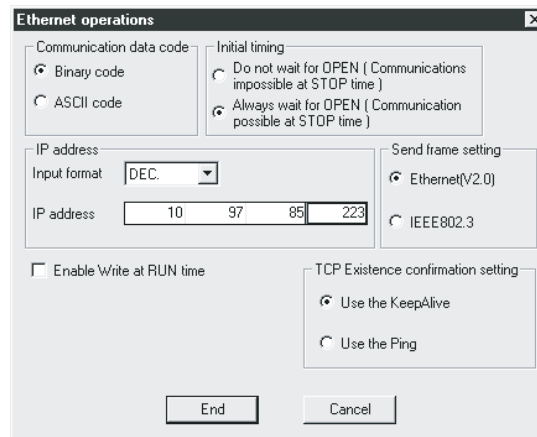
(1) 程序范例的执行环境

(a) 发送程序(QCPU-1 站端)

- 1) 1号连接用于固定缓冲存储器的发送。
- 2) 除“固定缓冲存储器通讯”参数外,假设通讯参数设置为第5.6.1节中所描述的“主动开放处理/关闭处理”。
“固定缓冲存储器通讯”设置应从“有顺序”改变为“无顺序”。
- 3) 1号固定缓冲存储器发送数据。
: 存储在 D300 到 D303 中
- 4) 1号固定缓冲存储器发送指令完成软元件。
: M300
- 5) 1号固定缓冲存储器发送指令异常完成软元件。
: M301
- 6) 1号固定缓冲存储器发送指令完成状态。
: D3001

(b) 接收程序(QCPU-2 站端)

- 1) 1号连接用于处理固定缓冲存储器的接收。
- 2) 以太网模块装载到主基板的“0”槽上。
- 3) 假定为用 GX Developer 如下设置“设置 MNET/10H 以太网网卡号的网络参数”(见第4.6节)
 - 网络类型 : 以太网
 - 起始 I/O 号 : 0000
 - 网络号 : 1
 - 组号 : 1
 - 站号 : 2
- 4) 假定用 GX Developer 设置了如下的“操作设置”参数。



本站 IP 地址: 0A.61.55.DFH (10.97.85.223)

5) 假定已按如下所示用 GX Developer 设置“开放设置”参数：

	Protocol	Open system	Fixed buffer	Fixed buffer communication	Pairing open	Existence confirmation	Local station Port No.	Destination IP address	Dest. Port No.
1	TCP	Unpassive	Receive	No procedure	No pairs	No confirm	2000		
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									

End Cancel

1 号连接本地站端口号：2000_H(用于主程序)

- 6) 1 号固定缓冲存储器接收数据
：存储在 D500 至 D503 中
- 7) 1 号固定缓冲存储器接收指令完成软元件
：M500
- 8) 1 号固定缓冲存储器接收指令异常完成软元件
：M501
- 9) 1 号固定缓冲存储器发送指令完成状态
：D5001
- 10) 1 号固定缓冲存储器接收状态信号
：M40

要点

根据源发送的数据最大长度来确保足够的设备存储，以避免用于其它用途的设备区被接收数据覆盖。

(2) 程序范例概要

(a) 发送程序(QCPU-1 站端)

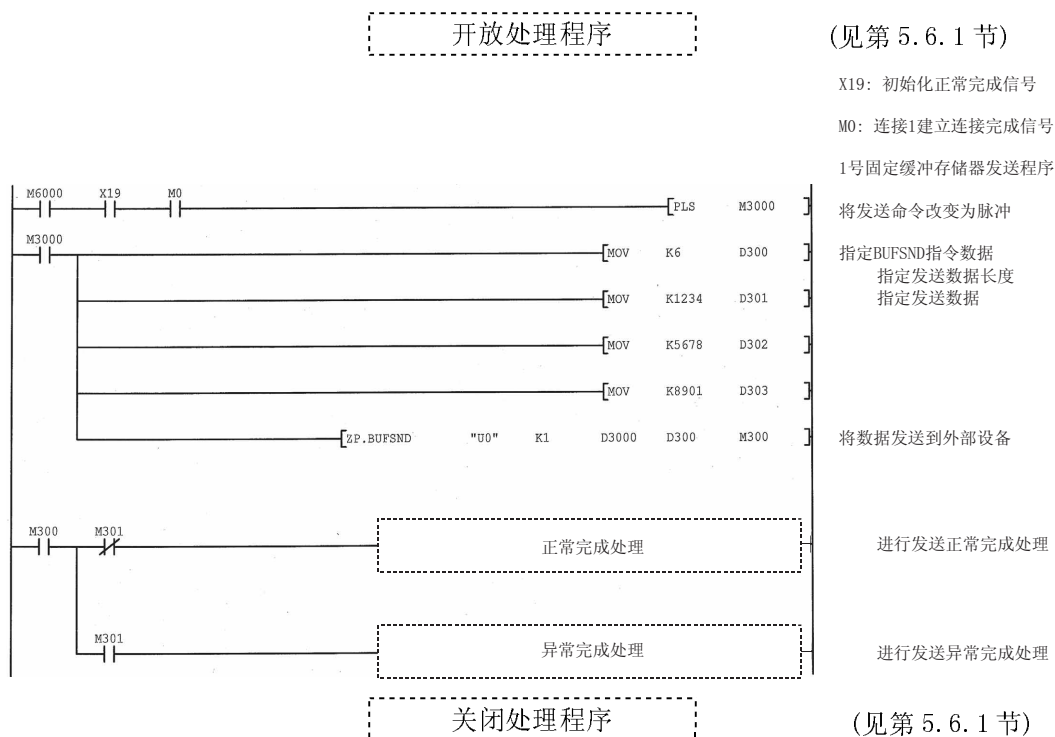
- 1) 在用 GX Developer 设置每个参数，并写到 PLC CPU 之后，将 PLC CPU 复位，然后确认初始化处理完成。
- 2) 进行 1 号连接的开放处理(*1)(主动开放)
- 3) 用通讯数据的固定缓冲存储器通讯(无顺序发送)从 PLC CPU 进行数据通讯。
- 4) 数据发送完成后，进行 1 号连接(*1)关闭处理。

*1 使用程序范例描述第 5.6.1 节“主动开放处理/关闭处理”(用于进行主动开放处理/关闭处理的顺控程序)。
注意应当改变“固定缓冲存储器通讯”设置，从“有顺序”改变为“无顺序”。

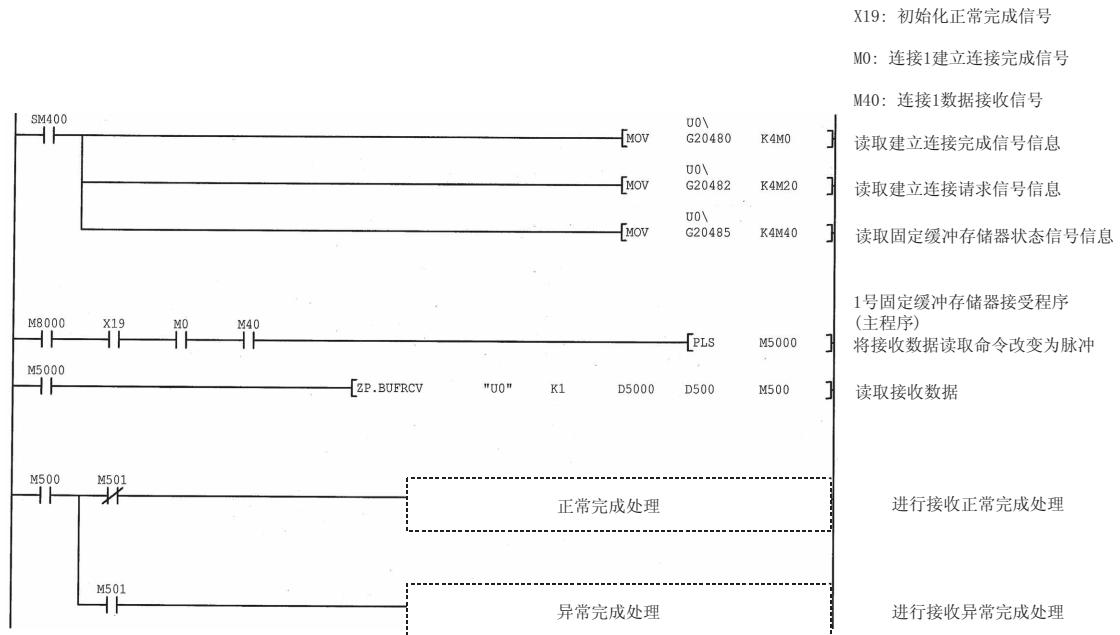
(b) 接收程序(QCPU-2 站端)

- 1) 用 GX Developer 设置了每个参数并写到 PLC CPU 之后, 将 PLC CPU 复位, 并确认初始化处理完成。
当初始化处理正常完成时, 1 号连接等待外部设备主动开放请求。
- 2) 用固定缓冲存储器通讯(无顺序发送)从外部设备对数据进行通讯。
- 3) 将以太网模块中相应的固定缓冲存储器数据区域所接收的数据读取到 PLC CPU 中。

(发送程序)



(接收程序)



9 使用随机访问缓冲存储器进行通讯

本章介绍如何使用以太网模块的随机访问缓冲存储器，在以太网模块和外部设备之间进行通讯数据。

要点
随机访问缓冲存储器的通讯功能不能和 PLC CPU 的电子邮件发送/接收功能一起使用。 但是，随机访问缓冲存储器的通讯功能可以和 PLC CPU 监视电子邮件发送功能一起使用。

9.1 控制方法

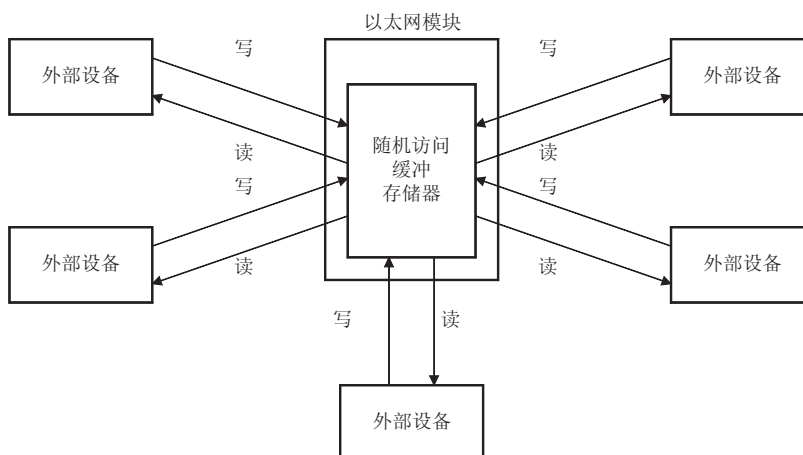
本节说明了随机访问缓冲存储器通讯的控制方法。

在随机访问缓冲存储器通讯中，数据是按照外部设备的指令(请求)写入随机访问缓冲存储器中，及按照外部设备的指令(请求)从随机访问缓冲存储器中读取数据的。

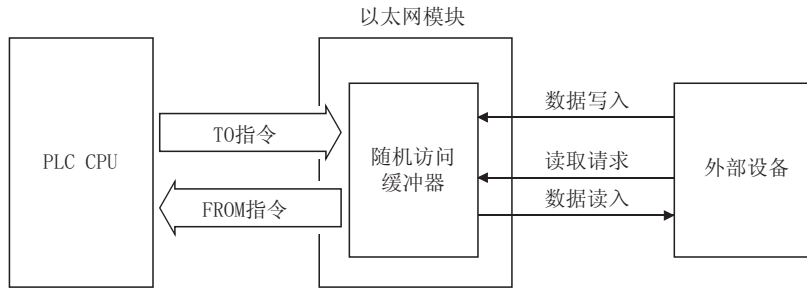
外部设备将数据写入以太网模块的随机访问缓冲存储器中及从以太网模块的随机访问缓冲存储器中读取数据与 PLC CPU 的顺控程序异步。

- (1) 任何外部设备都可以自主地将数据写入随机访问缓冲存储器中及从随机访问缓冲存储器中读取数据(除了以太网模块)，而不用把访问权指定给一个特殊的外部设备。

因此，它可以连接到以太网的所有外部设备的公共缓冲区中。



(2) 下列所示的是在使用随机访问缓冲器进行通讯处理时的数据流。



(3) 除了以太网模块本身(包括常规模块)之外,可以从所有外部设备进行随机访问缓冲存储器通讯。

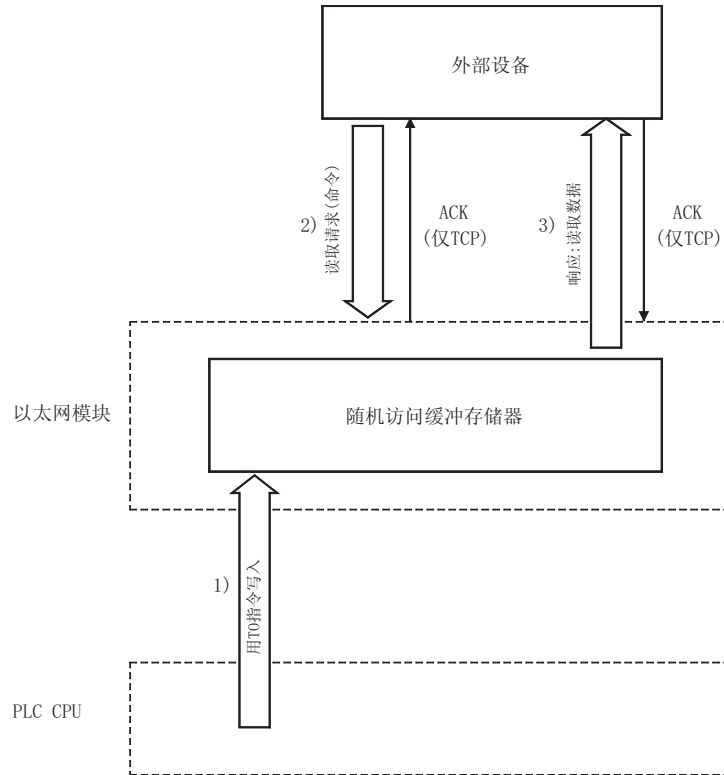
随机访问缓冲存储器通讯不能用于 PLC CPU 之间的通讯。

(可以使用随机访问缓冲存储器进行通讯的外部设备)

- 在以太网上与以太网模块连接的外部设备。
- 通过路由中继功能与以太网相连接的外部设备(请参见第 5.3 节)。

9.1.1 从外部设备读请求的控制方法

下图说明了从以太网模块发送数据来响应外部设备发出的读请求的控制方法。

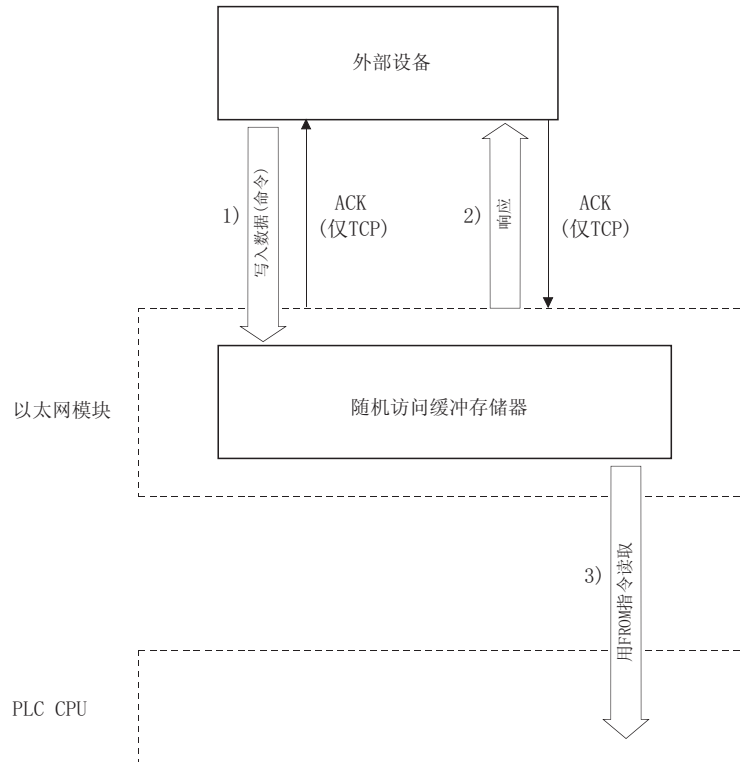


- 1) PLC CPU 按照顺控程序的 T0 指令，将数据写入以太网模块的随机访问缓冲存储器中。
或者，外部设备将数据写入以太网模块的随机访问缓冲存储器中。
- 2) 与上面的们处理异步，外部设备将读请求发送给以太网模块。
(以太网模块端：接收命令)
- 3) 一旦接收到外部设备的读请求，以太网模块就将写入随机访问缓冲存储器的数据发送到发出读请求的外部设备中。
(以太网模块端：发送响应)

要点
<p>(1) 在随机访问缓冲存储器通讯中，由于在固定缓冲存储器通讯步骤设置(请参见第 5.5 节)中选择了有顺序的控制方法，及在以太网开放完成信号(地址：5000_H …相应位)接通处使用连接，则数据可以与外部设备进行通讯。</p> <p>(2) 使用随机访问缓冲存储器进行的通讯与顺控程序异步执行。当必须同步时，使用固定缓冲存储器的通讯功能。</p>

9.1.2 从外部设备写请求的控制方法

下图说明了外部设备将数据写入以太网模块随机访问缓冲存储器中的控制方法。



- 1) 外部设备将数据写入以太网模块的随机访问缓冲存储器中。
(以太网模块端：接收命令)
- 2) 以太网处理外部设备发出的写入请求，并将写入结果返回给发出写入请求的外部设备。
(以太网模块端：发送响应)
- 3) 当与以上 1) 和 2) 处理异步时，用顺控程序的 FROM 指令读写入随机访问缓冲存储器的数据。

要点
<p>(1) 在随机访问缓冲存储器通讯中，由于固定缓冲存储器通讯的步骤设置(请参见第 5.5 节)中选择了有顺序的控制方法，及在以太网开放完成信号(地址：5000_H … 相应位)接通处使用连接，则数据可以与外部设备进行通讯。</p> <p>(2) 使用随机访问缓冲存储器进行的通讯与顺控程序异步执行。当必须同步时，使用固定缓冲存储器的通讯功能。</p>

9.2.2 应用数据

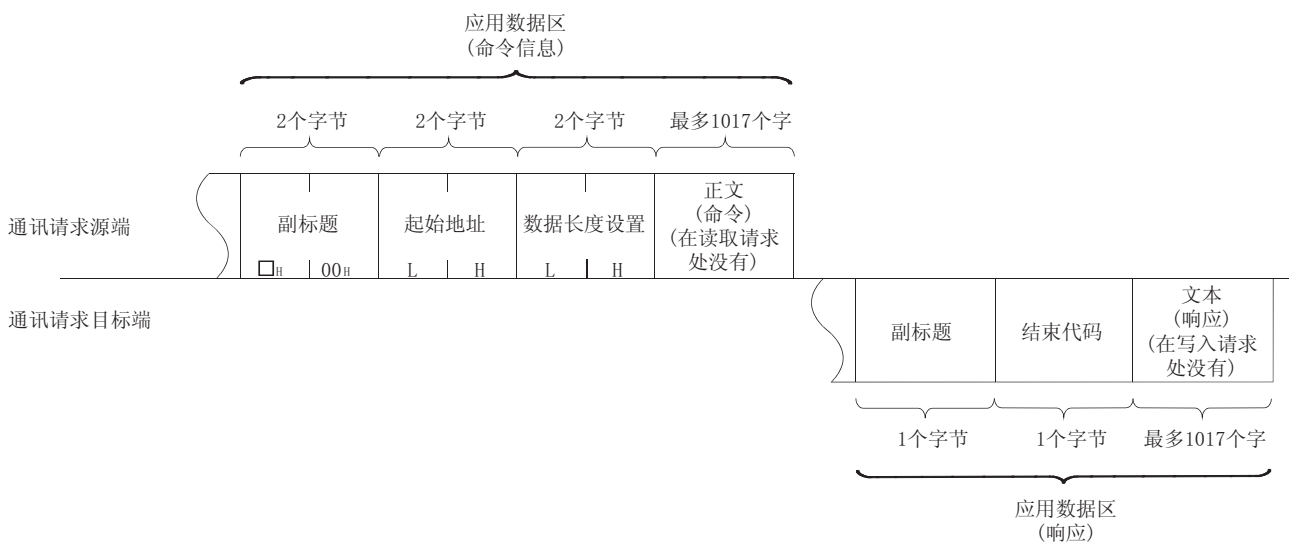
如下所示，应用数据中的数据代码既可以用二进制代码表示，又可以用 ASCII 代码表示。通过 GX Developer 执行二进制代码和 ASCII 代码之间的切换，如下所示：

[GX Developer] - [网络参数] - [操作设置] - [通讯数据代码]

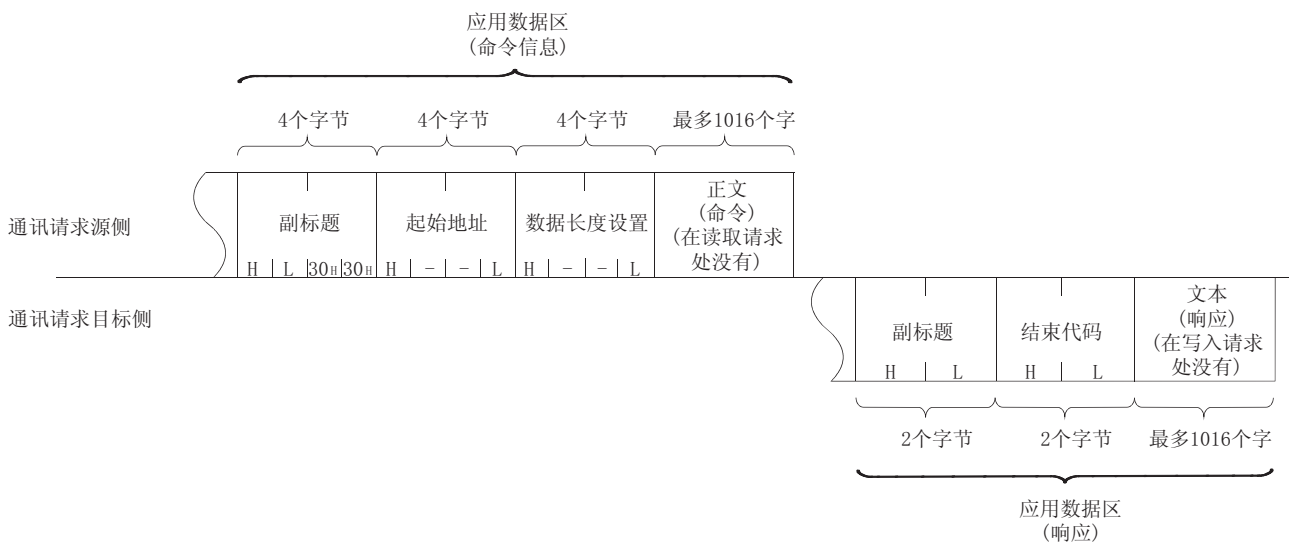
有关详细说明，请参见 4.7 节的“操作设置”。

(1) 格式

(a) 使用二进制代码进行通讯



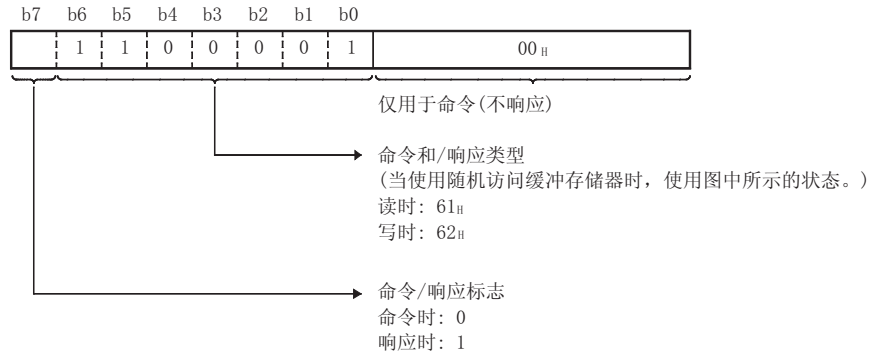
(b) 使用 ASCII 代码进行通讯



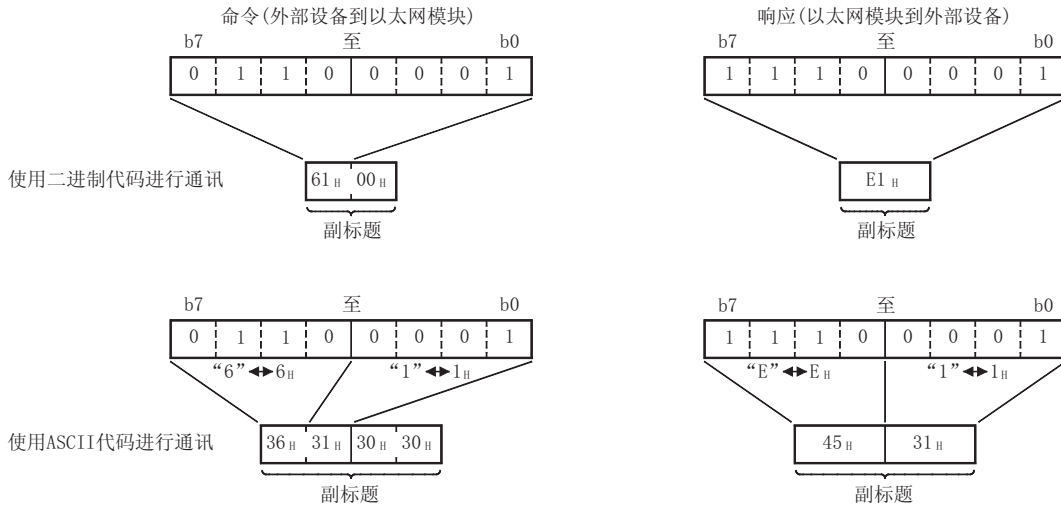
(2) 副标题

副标题的格式如下所示：

使用以太网模块时，由于以太网模块自动把副标题加到正文中或从正文中删除，用户不需要设置副标题。



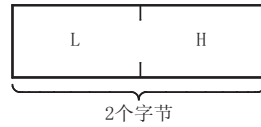
● 读取时



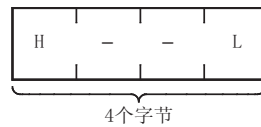
(3) 起始地址

表示使用逻辑地址(范围: 从 0H 到 17FFH, 请参见第 9.3 节)从中读数据/写入数据的随机访问缓冲存储器的起始地址(范围: 从 2680H 到 3F7FH)。

(a) 使用二进制代码进行通讯 : 以二进制值指定起始地址。



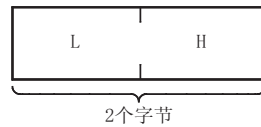
(b) 使用 ASCII 代码进行通讯 : 指定以十六进制表示起始地址的 ASCII 代码值。



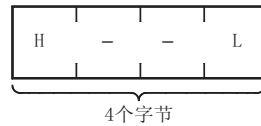
(4) 数据长度设置

表示在随机访问缓冲存储器范围中读取/写入数据的字数。

(a) 使用二进制代码进行通讯 : 以二进制值指定字数。



(b) 使用 ASCII 代码进行通讯 : 指定以十六进制表示起始地址的 ASCII 代码值。



要点

可以在下列范围指定数据长度:

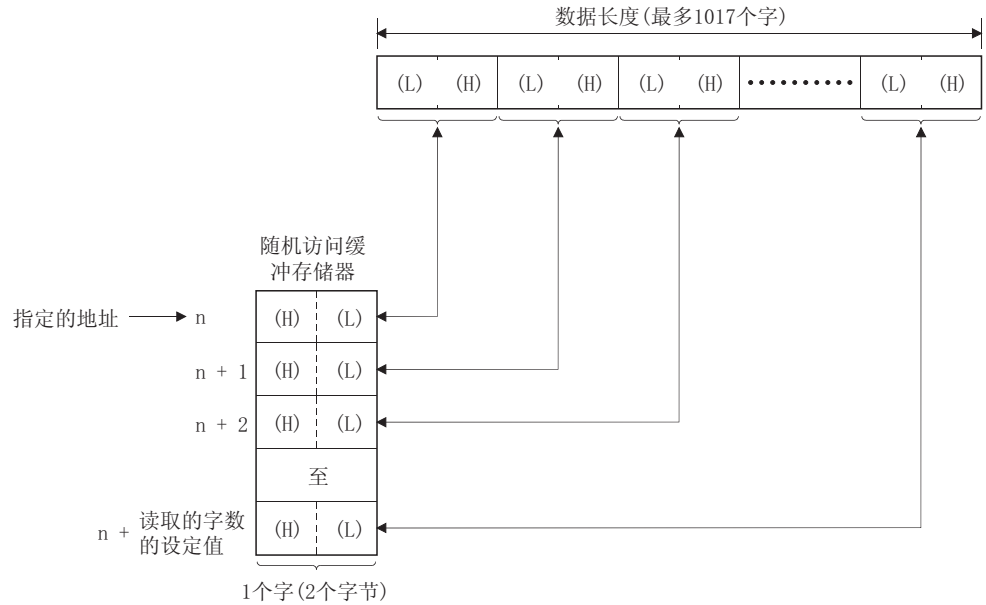
- 使用二进制代码进行通讯: 最多 1017 个字
- 使用 ASCII 代码进行通讯: 最多 508 个字(*1)

*1 由于数据是按 ASCII 数据发送/接收的, 通讯数据量约是使用二进制代码时数据量的一半。

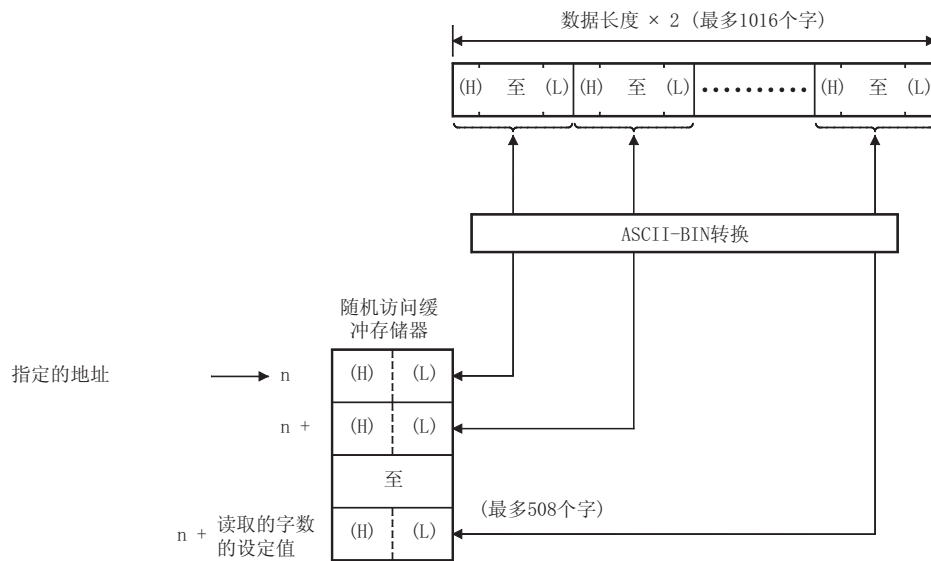
(5) 正文

表示写入随机访问缓冲存储器的数据和从随机访问缓冲存储器读取的数据。

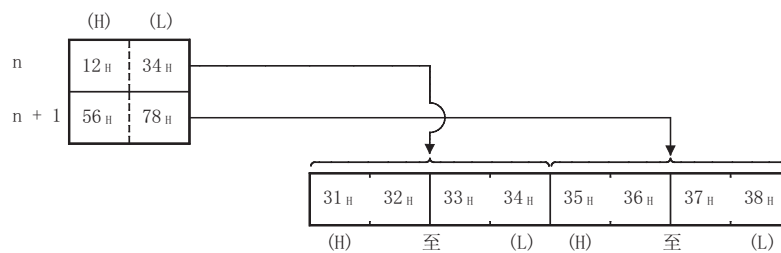
(a) 使用二进制代码进行通讯



(b) 使用 ASCII 代码进行通讯



(范例)



(6) 结束代码

下列结束代码加到随机访问缓冲存储器通讯的响应中。

- 正常完成 : 00_H
- 异常完成 : 除 00_H 以外的值 (请参见第 11.3.1 节)

结束代码存储在缓冲存储器的通讯状态存储区中。

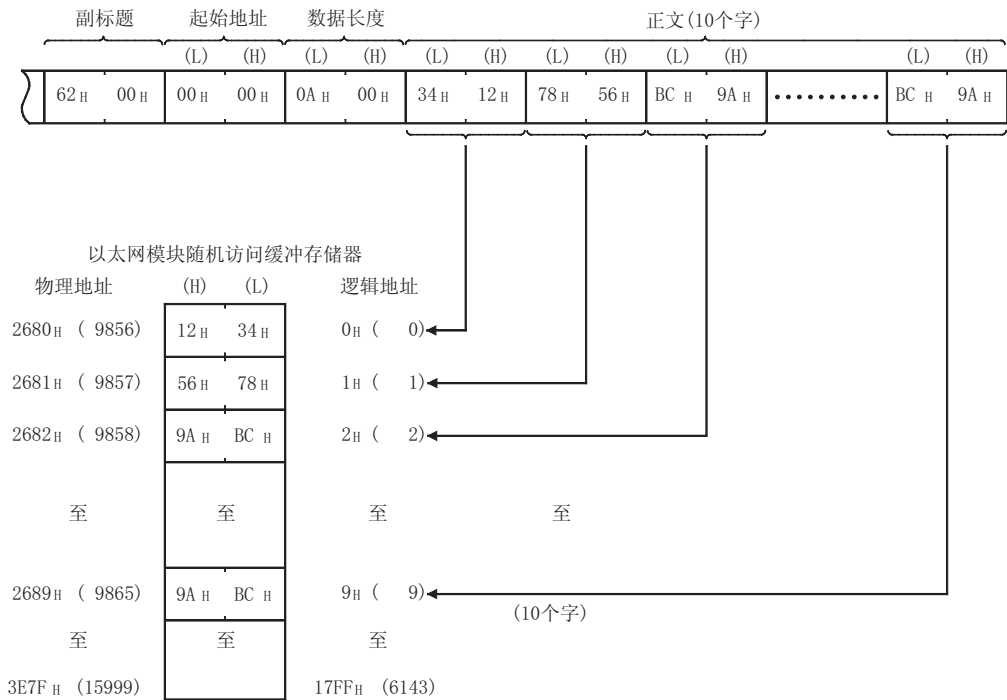
9.2.3 命令/响应格式的范例

本节说明了使用随机访问缓冲存储器进行通讯期间的命令/响应格式的范例。

(1) 通过从外部设备的写请求，写入随机访问缓冲存储器中

(a) 使用二进制代码进行通讯

1) 命令格式(外部设备到以太网模块)

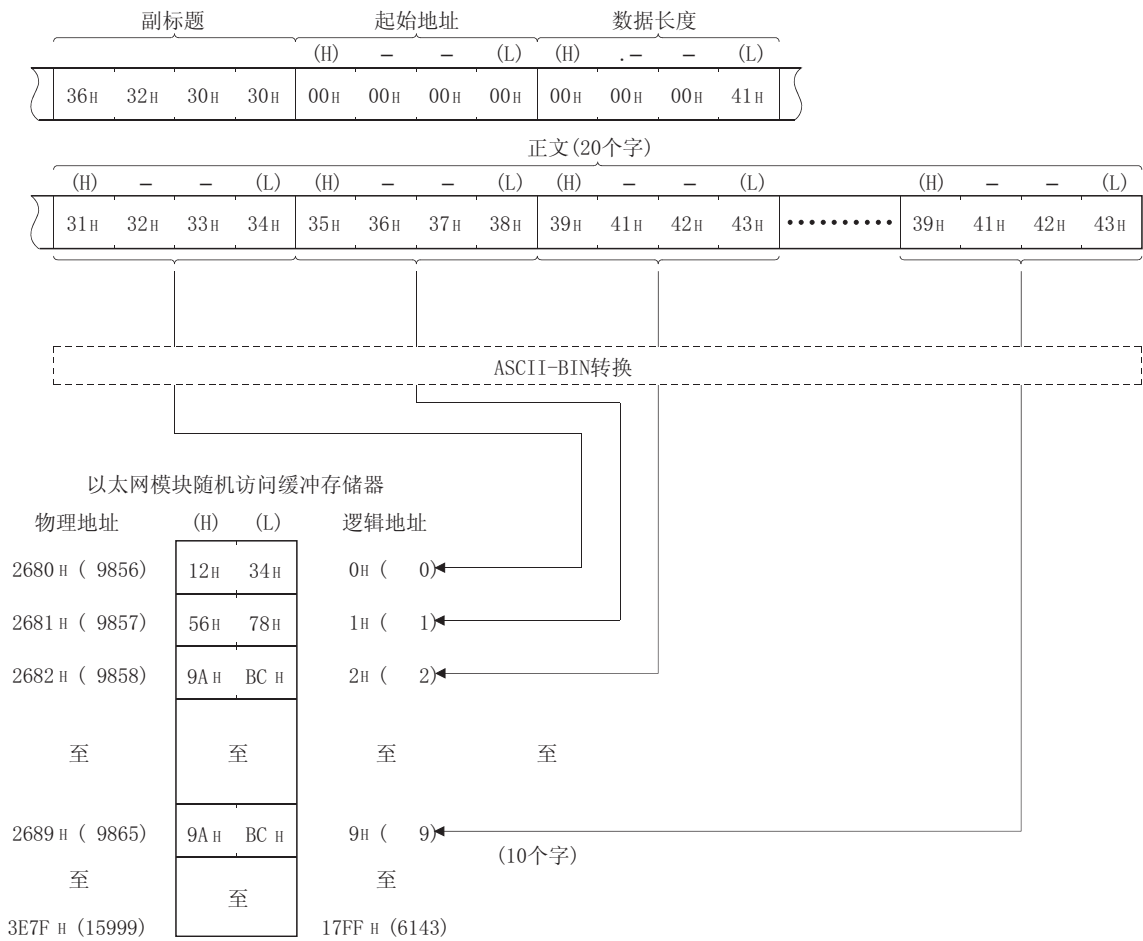


2) 响应格式(以太网模块到外部设备)

副标题	结束代码
E2 _H	00 _H

(b) 使用 ASCII 代码进行通讯

1) 命令格式(外部设备到以太网模块)



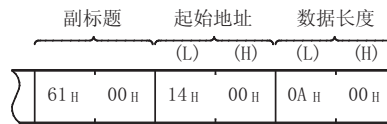
2) 响应格式(以太网模块到外部设备)

副标题	结束代码
45H 32H	30H 30H

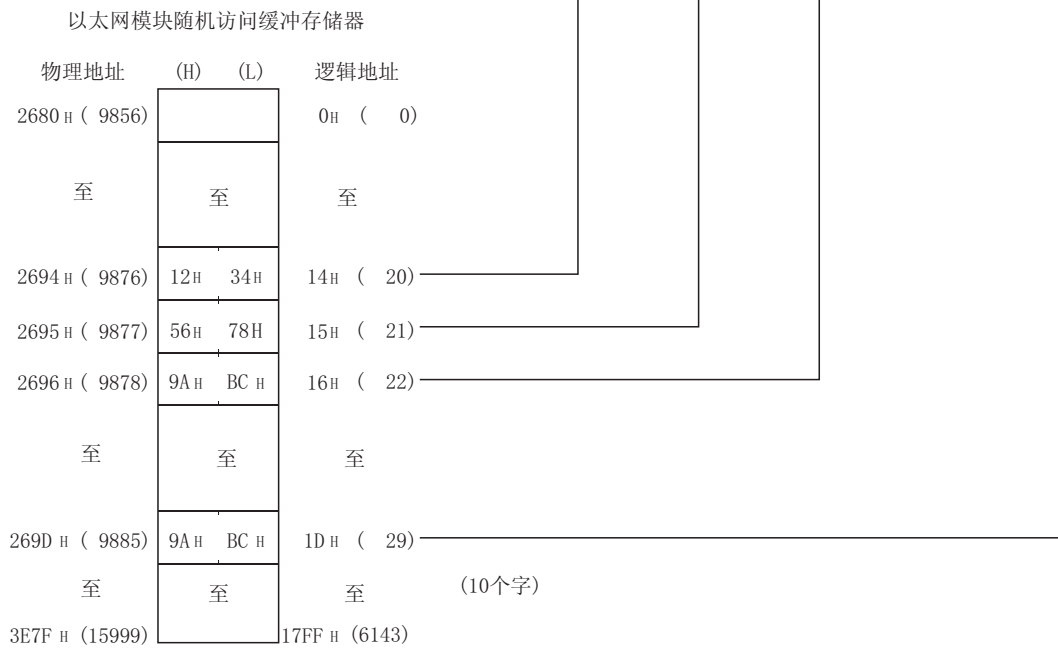
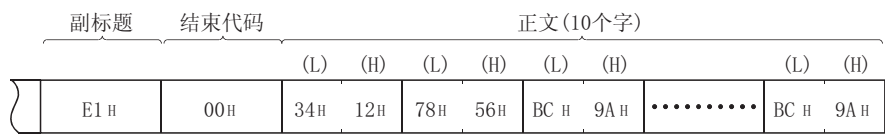
(2) 通过从外部设备的读请求，从随机访问缓冲存储器中读取数据

(a) 使用二进制代码进行通讯

1) 命令格式(外部设备到以太网模块)

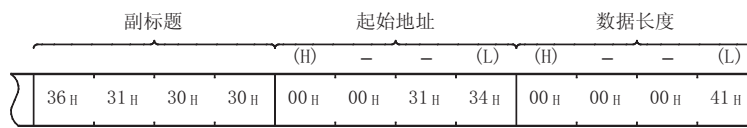


2) 响应格式(以太网模块到外部设备)

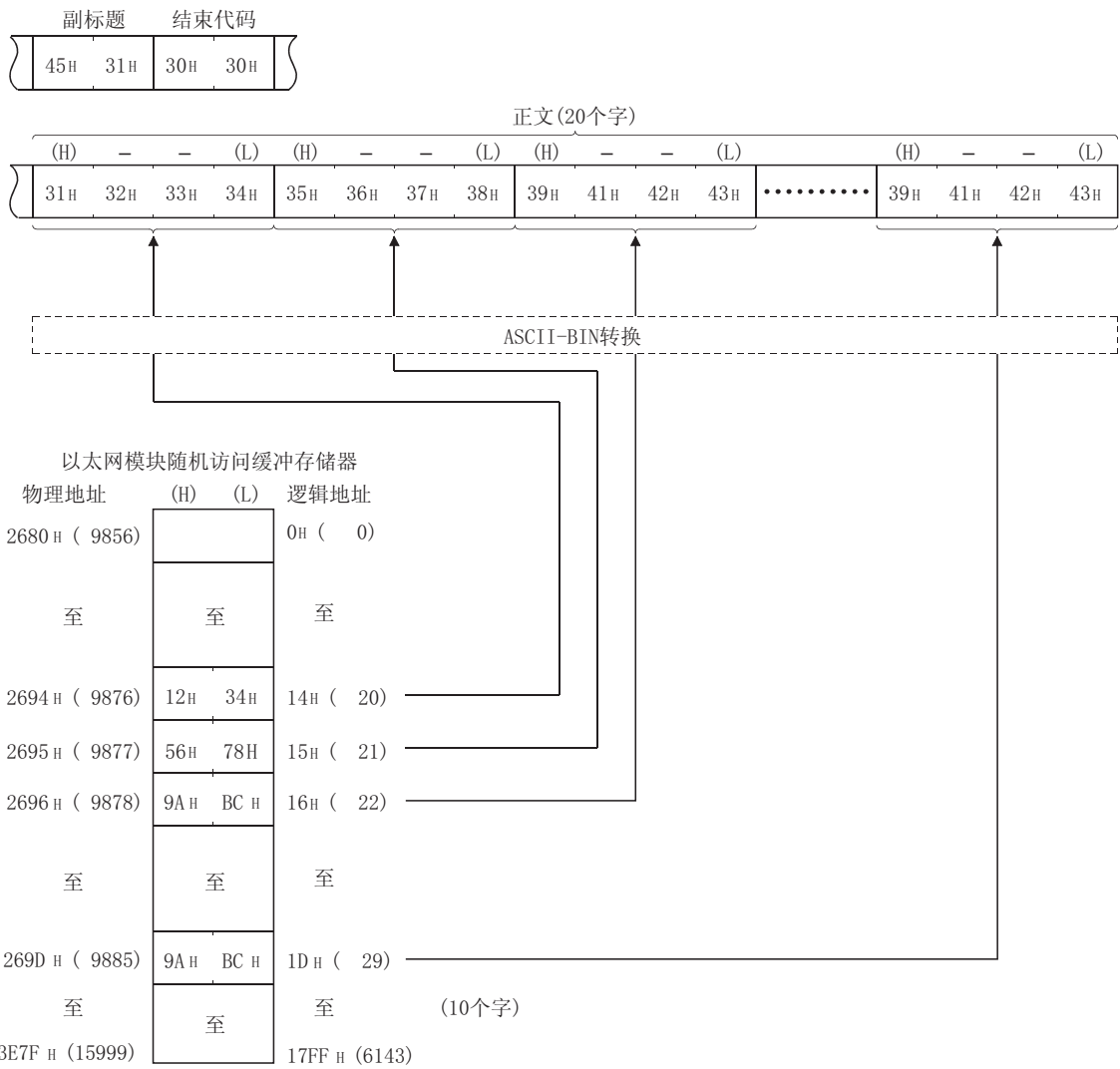


(a) 使用 ASCII 代码进行通讯

1) 命令格式(外部设备到以太网模块)



2) 响应格式(以太网模块到外部设备)



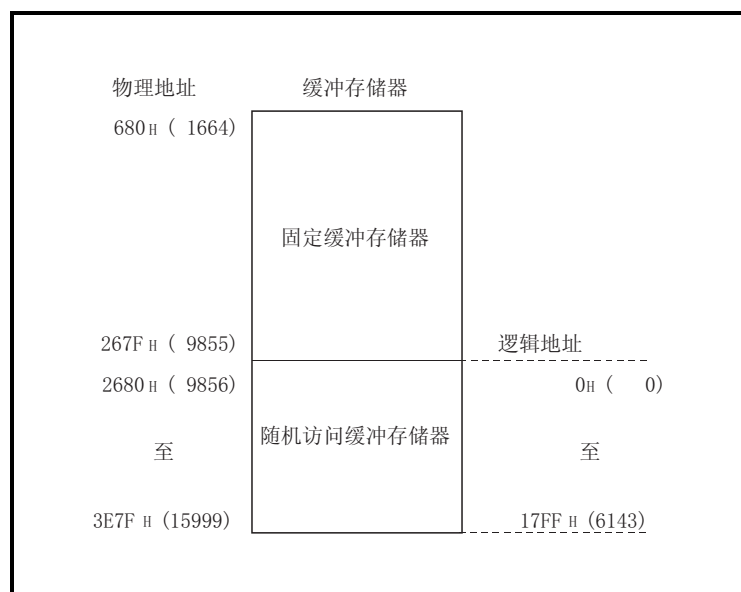
9.3 随机访问缓冲存储器的物理地址和逻辑地址

本节说明了以太网随机访问缓冲存储器(无电池备份)的起始地址，它是用随机访问缓冲存储器通讯的命令指定的。

随机访问缓冲存储器指定的地址如下所示。

注意指定随机访问缓冲存储器的起始地址时，外部设备指定的地址与顺控程序中的 FROM/T0 指令指定的地址不同。

物理地址 : 用顺控程序中的 FROM/T0 指令指定的地址
逻辑地址 : 外部设备指定作为命令中起始地址项的地址



9.4 创建程序时的注意事项

本节简要地说明，使用随机访问缓冲存储器创建在以太网模块和外部设备之间进行数据通讯程序时应遵守的注意事项。

- (1) 为了使用随机访问缓冲存储器的通讯，必须完成初始化处理和开放处理。
- (2) 当使用随机访问缓冲存储器通讯时，PLC CPU 不能发布发送请求。
另外，PLC CPU 不能确认接收完成。
当需要同步进行 PLC CPU 和外部设备之间的数据发送/接收时，使用固定缓冲存储器的通讯功能。
- (3) 用外部设备为随机访问缓冲存储器指定的地址和用顺控程序中的 FROM/T0 指令指定的地址是不同的。
有关详细说明，请参见第 9.3 节。

10 专用指令

专用指令用于在使用智能型功能模块的功能时简化编程。

本章介绍了本手册中有关功能的专用指令，其中用于以太网模块的专用指令也可用于QCPU。

10.1 专用指令列表

本表列出了此章中所介绍的专用指令：

应用	专用指令	功能说明	参考小节
用于开放和关闭	OPEN	开放。	10.8 节
	CLOSE	关闭。	10.5 节
用于固定缓冲存储器之间的通讯	BUFRCV	读取接收到的数据。 (用于主程序)	10.2 节
	BUERCVS	读取接收到的数据。 (用于中断程序)	10.3 节
	BURSND	发送数据。	10.4 节
用于读取和清除错误信息	ERRCLR	清除出错信息。	10.6 节
	ERRRD	读取出错信息。	10.7 节
用于重新初始化	UINI	重新初始化以太网模块	10.9 节

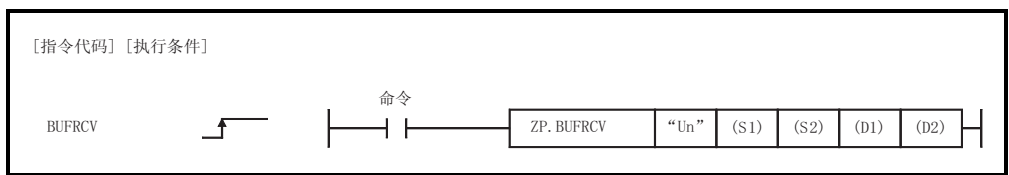
要点

- | |
|---|
| <p>(1) 直到专用指令执行完毕，用户才可更改任何带有专用指令的规定数据(控制数据、要求数据等)。</p> <p>(2) 所有的专用指令都必须在线执行。
如果有任何专用指令离线执行，虽不会发生错误，但是专用指令的执行将不能完成。</p> |
|---|

10.2 BUFRCV 指令

该指令可读取通过固定缓冲存储器的通讯，从外部设备接收到的数据。
该指令用于主程序。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○						○	-	-
(S2)	-	○						-	-	-
(D1)	-	○						-	-	-
(D2)	○	○						-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
"Un"	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	连接号 (1 至 16)	系统	16 位二进制
(S2)	指定控制数据的软元件的起始编号		16 位二进制
(D1)	储存已接收数据的软元件的起始编号		16 位二进制
(D2)	指令完成时变为 ON 并持续扫描周期的本地站位软元件的起始编号。 如果指令的执行异常结束，(D2)+1 也变为 ON。		位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

软元件	项目	设置数据	设置范围	由(*)设置
(S2) + 0	系统区域	-	-	-
(S2) + 1	完成状态	• 在完成时 储存状态。 0000h: 正常完成 非 0000h : 异常完成(出错代码) (* 2)	-	系统

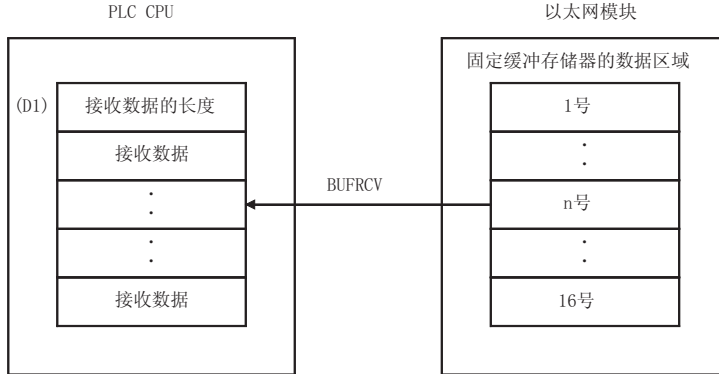
接收数据

软元件	项目	设置数据	设置范围	由(*)设置
(D1) + 0	接收数据的长度	• 以字为单位来储存从固定缓冲存储器的数据区域中读数据的数据长度。 有顺序(用于使用二进制代码的通讯): 字数 有顺序(用于使用 ASCII 代码的通讯): 字数 无顺序(用于使用二进制代码的通讯): 字节数	- 1 至 1017 1 至 508 1 至 2046	系统
(D1) + 1 至 (D2) + n	接收数据	按升序依次储存从固定缓冲存储器的数据区域中读取的数据。	-	系统

*1 “由……设置”一栏如下指示：
 • 用户 : 在执行专用指令之前，由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。
 *2 异常完成时，有关出错代码的详细说明，请参见 11.3 节“出错代码列表”。

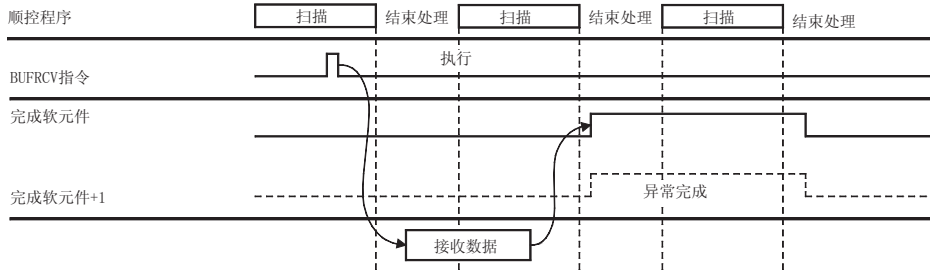
功能

- (1) 该指令用于为用 Un 指定的模块 (通过固定缓冲存储器) 读取从 S1 规定的连接中接收的数据。



- (2) 可以通过完成位软元件 (D2)+0 和 (D2)+1 来检查 BUFRCV 指令是否已经完成。
 - (a) 完成位软元件 (D2)+0
在 BUFRCV 指令完成时, 扫描结束处理开放, 且在下一个结束处理时关闭。
 - (b) 完成位软元件 (D2)+1
根据 BUFRCV 指令的完成状态来开放或关闭。
 - 正常完成 : 保持关闭不变。
 - 异常完成 : 在 BUFRCV 指令完成时, 扫描结束处理开放, 且在下一个结束处理时关闭。

[执行 BUFRCV 指令时的操作]



- (3) 当读取指令 (由缓冲存储器的固定缓冲接收状态信号存储区域 (地址: 5005H) 中的应用连接位来表示) 从 OFF 切换到 ON 时, 执行 ZP.BUFRCVCV 指令。
- (4) 当从同样的连接中读取接收的数据时, 不可与 BUFRCVS 指令 (用于中断程序) 一起使用。

出错

- (1) 当专用指令由于出错而结束时，异常完成信号(D2)+1 开放，且将出错代码储存在完成状态区域(S2)+1 中。

有关出错代码，检查出错及纠正措施，参见以下手册。

<出错代码>

小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)

大于等于 C000_H : 本手册 11.3.3 节

程序范例

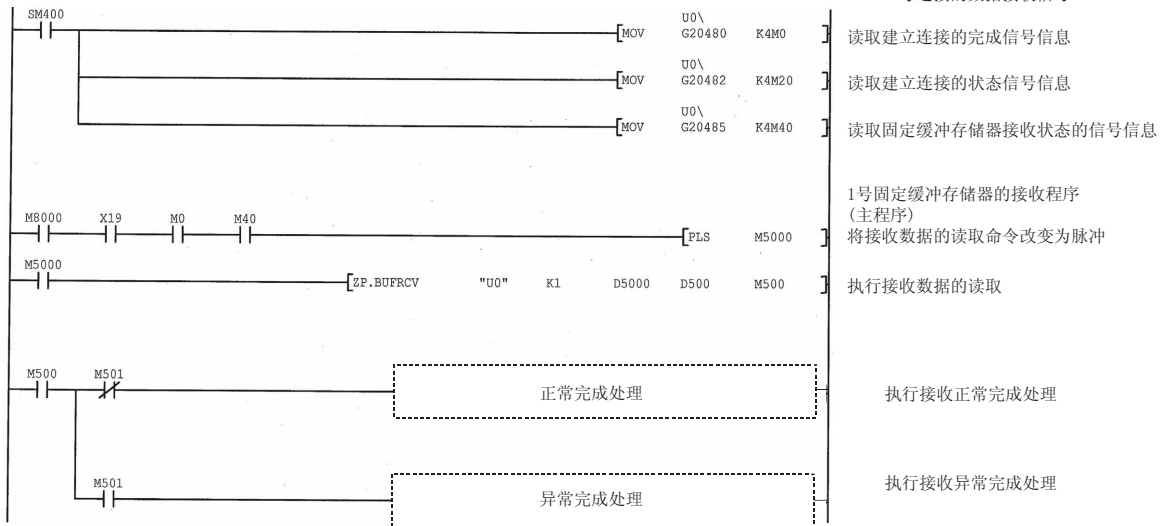
从 1 号连接的固定缓冲储存器中读取接收数据的程序：

当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F 时

X19: 初始化正常完成的信号

M0: 1号连接建立连接完成信号

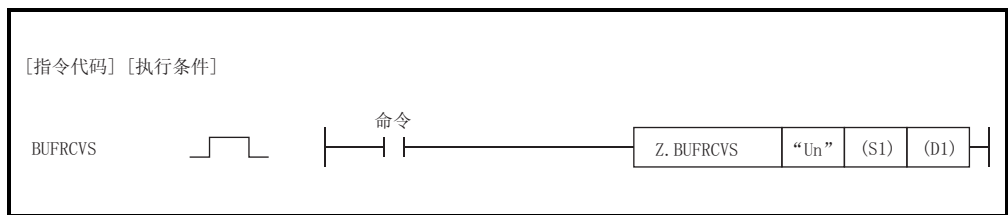
M40: 1号连接的数据接收信号



10.3 BUFRCVS 指令

该指令可读取通过固定缓冲存储器的通讯，从外部设备接收到的数据。
该指令用于中断程序。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○						○	-	-
(D1)	-	○						-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
“Un”	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	连接号 (1 至 16)		16 位二进制
(D1)	储存已接收数据的软元件的起始编号	系统	16 位二进制

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

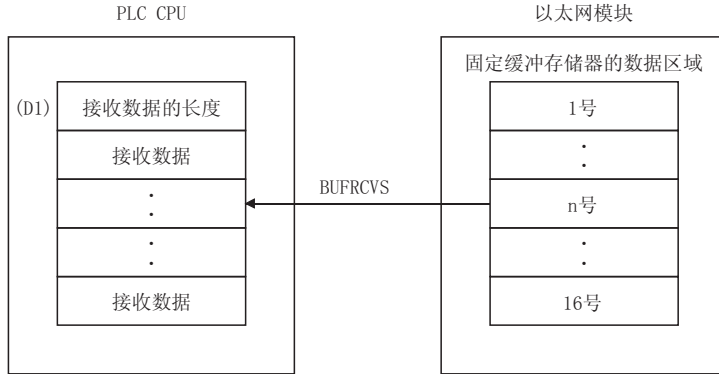
接收数据

软元件	项目	设置数据	设置范围	由(*)设置
(D1) + 0	接收数据的长度	• 以字为单位来储存从固定缓冲存储器的数据区域中读数据的数据长度。	-	系统
		有顺序(用于使用二进制代码的通讯): 字数	1 至 1017	
		有顺序(用于使用 ASCII 代码的通讯): 字数 无顺序(用于使用二进制代码的通讯): 字节数	1 至 508 1 至 2046	
(D1) + 1 至 (D1) + n	接收数据	• 按升序依次储存从固定缓冲存储器的数据区域中读取的数据。	-	系统

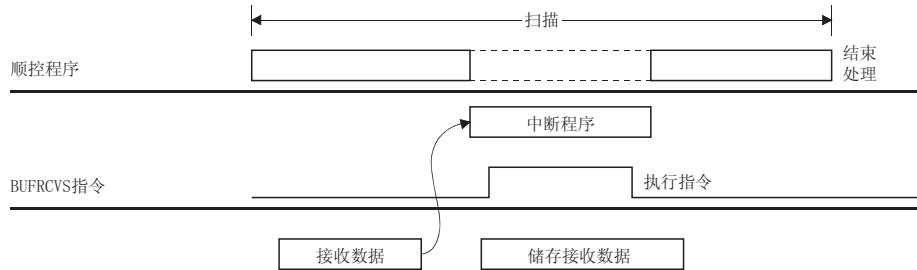
*1 “由……设置”一栏如下指示:
 • 用户 : 在执行专用指令之前, 由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。

功能

- (1) 该指令用于为用 Un 指定的模块 (通过固定缓冲存储器) 读取 S1 规定的连接中接收的数据。



[当执行 BUFRCVS 指令时的操作]



- (2) 通过中断程序来执行 Z.BUFRCVS 指令，且在一次扫描之内完成处理。
- (3) 为了能通过中断程序来读接收的数据，必须用 GX Developer 参数设置来执行中断设置和中断指针。
- (4) 当从同样的连接中读取接收的数据时，不可与 BUFRCV 指令 (用于主程序) 一起使用。

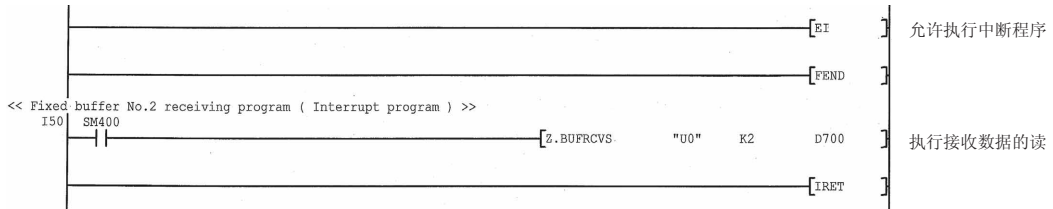
错误

- (1) 当专用指令异常完成时，出错标志 (SM0) 接通，且将出错代码存储在 SD0 中。有关出错代码、检查出错和纠正措施，参见以下手册。
 <出错代码>
 小于等于 4FFF_H : QCPU (Q 模式) 用户手册 (硬件设计、维护和检查)
 大于等于 C000_H : 本手册 11.3.3 节

程序范例

从 2 号连接的固定缓冲存储器中读取接收数据的程序：

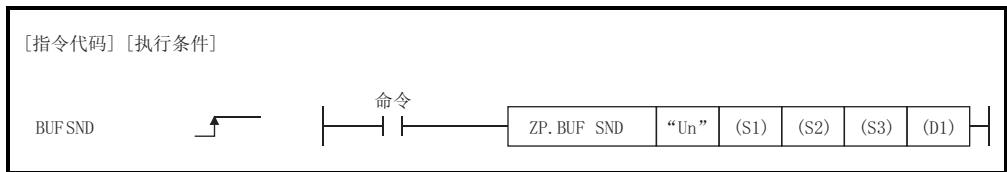
当以太网的输入/输出信号为 X/Y00 到 X/Y11F 时



10.4 BUFSND 指令

该指令通过固定缓冲存储器的通讯，将数据发送给外部设备。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○					○	-	-	
(S2)	-	○					-	-	-	
(S3)	-	○					-	-	-	
(D1)	○	○					-	-	-	



设置数据

设置数据	说明	由(*)设置	数据类型
"Un"	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	连接号 (1 至 16)	用户	16 位二进制
(S2)	存储控制数据的软元件的起始编号	系统	16 位二进制
(S3)	存储发送数据的软元件的起始编号	用户	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站软元件的起始编号。 如果指令的执行异常结束, (D1) + 1 也开放。	系统	位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

软元件	项目	设置数据	设置范围	由(*)设置
(S2) + 0	系统区域	-	-	-
(S2) + 1	完成状态	<ul style="list-style-type: none"> 在完成时储存状态。 0000_n: 正常完成 非 0000_n: 异常完成(出错代码) (*2) 	-	系统

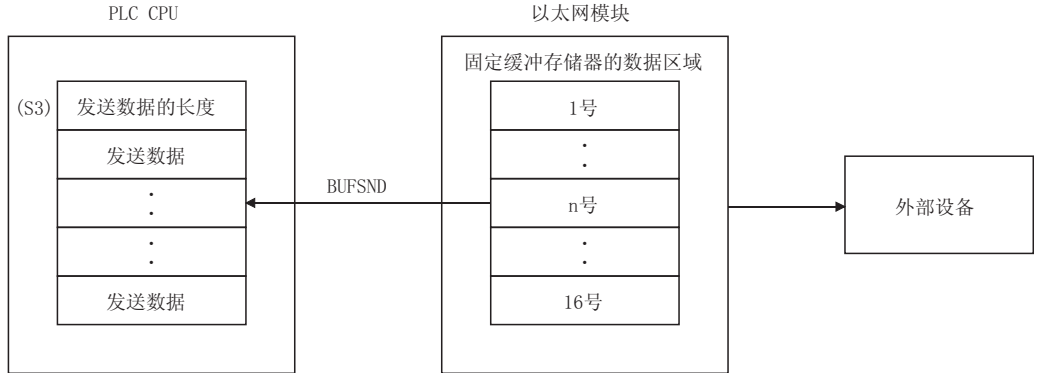
发送数据

软元件	项目	设置数据	设置范围	由(*)设置
(S3) + 0	发送数据的长度	以字为单位来指定发送数据的长度。	-	用户
		有顺序(用于使用二进制代码的通讯): 字数	1 至 1017	
		有顺序(用于使用 ASCII 代码的通讯): 字数	1 至 508	
(S3) + 1 至 (S3) + n	发送数据	<ul style="list-style-type: none"> 指定发送的数据。 	-	用户

*1 “由……设置”一栏如下指示:
 • 用户 : 在执行专用指令之前, 由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。
 *2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

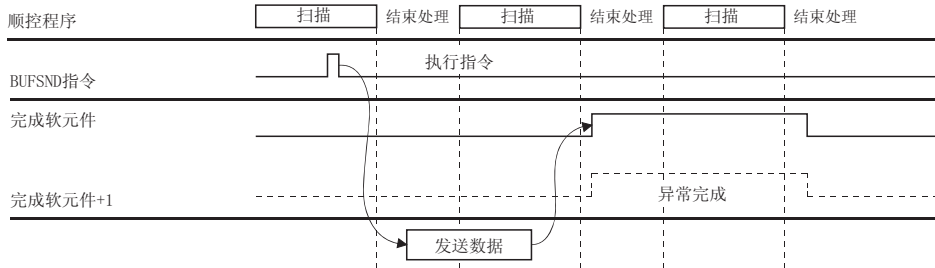
功能

- (1) 该指令用于为用 Un 指定的模块将 (S3) 指定的数据发送到 (S1) 规定连接的外部设备中。



- (2) 可以通过完成位软元件 (D1)+0 和 (D1)+1 来检查 BUFSND 指令是否已经完成。
 - (a) 完成位软元件 (D1)+0
在 BUFSND 指令完成时，扫描结束处理开放，且在下一个结束处理时关闭。
 - (b) 完成位软元件 (D1)+1
根据 BUFSND 指令的完成状态开放或关闭。
 - 正常完成 : 保持关闭。
 - 异常完成 : 在 BUFSND 指令完成时，扫描结束处理开放，且在下一个结束处理时关闭。

[当执行 BUFSND 指令时的操作]



- (3) 当发送指令从 OFF 切换到 ON 时，执行 ZP.BUFSND 指令。

出错

- (1) 当专用指令由于出错而结束时，异常完成信号(D1)+1 接通，且将出错代码储存在完成状态区域(S2)+1 中。

有关出错代码，检查错误及纠正措施，参见以下手册。

<出错代码>

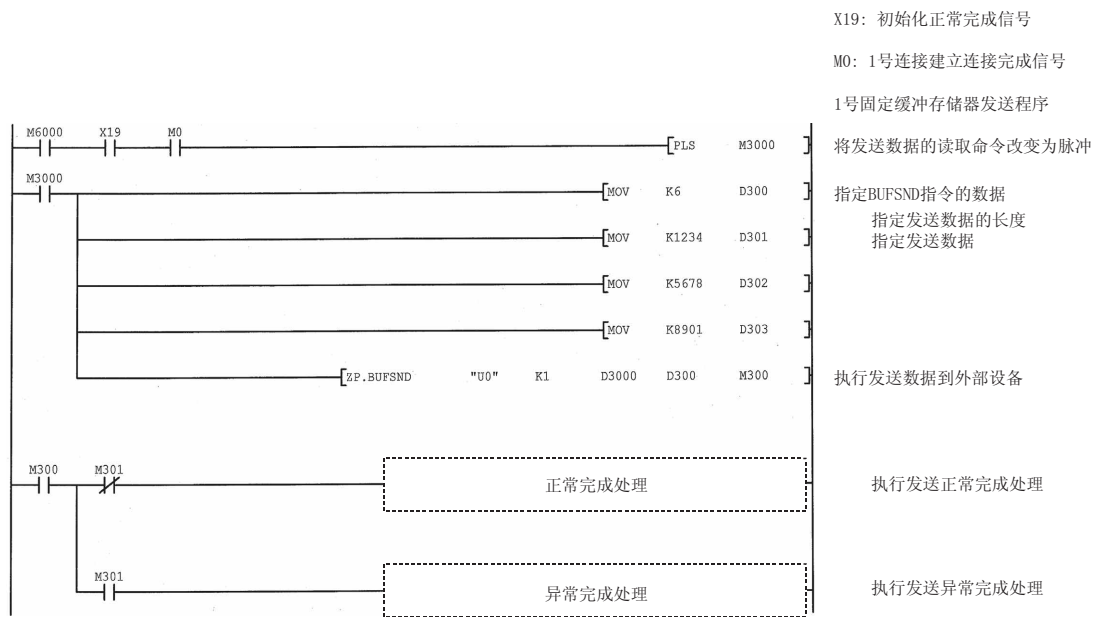
小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)

大于等于 C000_H : 本手册 11.3.3 节

程序范例

从 1 号连接的固定缓冲存储器中发送数据的程序：

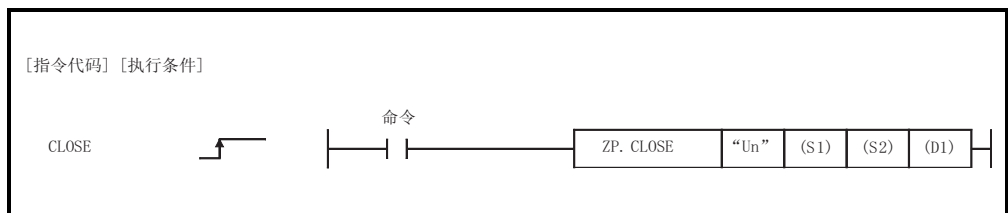
当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F 时



10.5 CLOSE 指令

该指令断开(关闭)与外部设备进行数据通讯的连接。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○						○	-	-
(S2)	-	○						-	-	-
(D1)	○	○						-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
“Un”	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	连接号 (1 至 16)		16 位二进制
(S2)	存储控制数据的软元件的起始编号	系统	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站软元件的起始编号。 如果指令的执行异常结束, (D1)+1 也变为 ON。		位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

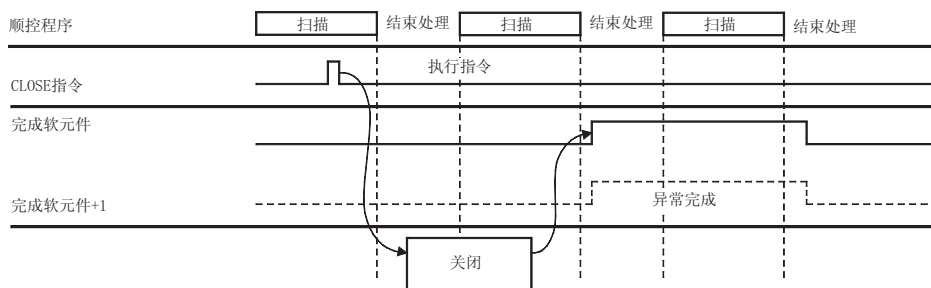
软元件	项目	设置数据	设置范围	由(*)设置
(S2) + 0	系统区域	-	-	-
(S2) + 1	完成状态	• 在完成时储存状态。 0000 _n : 正常完成 非 0000 _n : 异常完成(出错代码)(* ²)	-	系统

*1 “由……设置”一栏如下指示:
 • 用户 : 在执行专用指令之前, 由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。
 *2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

功能

- (1) 该指令用于为用 Un 指定的模块关闭由 (S1) 规定的连接(关闭)。
- (2) 通过完成的位软元件 (D1)+0 和 (D1)+1 来检查 CLOSE 指令是否已经完成。
 - (a) 完成位软元件 (D1)+0
在 CLOSE 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。
 - (b) 完成位软元件 (D1)+1
根据 CLOSE 指令的完成状态接通或断开。
 - 正常完成 : 保持关闭不变。
 - 异常完成 : 在 CLOSE 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。

[执行 CLOSE 指令时的操作]



- (3) 当断开指令从 OFF 切换到 ON 时, 执行 ZP.CLOSE。

要点
不要对同一的连接, 同时使用输入/输出信号执行接通/断开处理和使用 OPEN 或 CLOSE 专用指令执行接通/断开处理, 否则将导致故障。

出错

- (1) 当专用指令由于出错而结束时, 异常完成信号 (D1)+1 开放, 且将出错代码储存在完成状态区域 (S2)+1 中。
有关出错代码, 检查出错及纠正措施, 参见以下手册。
<出错代码>
小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)
大于等于 C000_H : 本手册 11.3.3 节

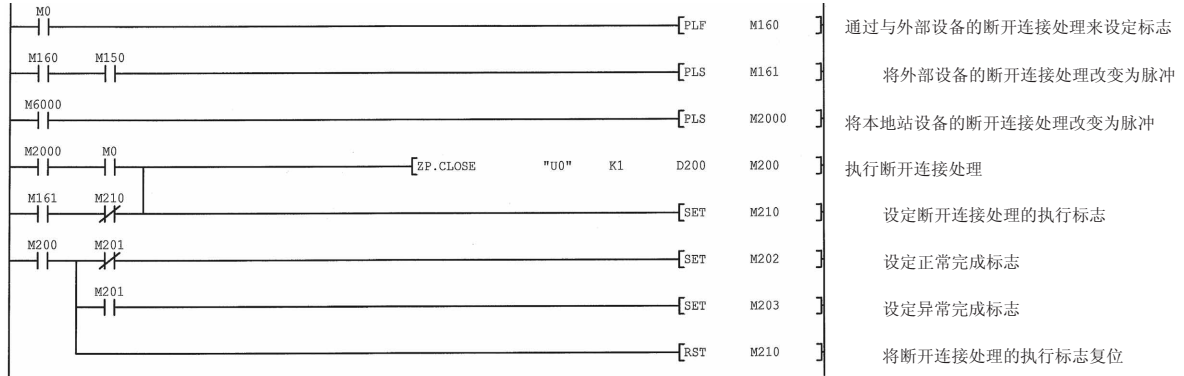
程序范例

1号连接关闭的程序：

当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F

M0: 1号连接建立连接完成信号

M150: OPEN指令执行正常完成的标志



10.6 ERRCLR 指令

该指令可关闭以太网模块的 LED 灯和清除储存在缓冲存储器中的数据信息。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○						-	-	-
(D1)	○	○						-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
“Un”	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	存储控制数据的软元件的起始编号	用户、系统	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站位软元件的起始编号。 如果指令的执行异常结束，(D1)+1 也变为 ON。	系统	位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

软元件	项目	设置数据	设置范围	由(*)设置
(S1) + 0	系统区域	-	-	-
(S1) + 1	完成状态	<ul style="list-style-type: none"> 在完成时 储存状态。 0000_h: 正常完成 非 0000_h: 异常完成(出错代码) (*²) 	-	系统
(S1) + 2	清除目标指定	<ul style="list-style-type: none"> 指定要清除的错误信息。 0000_h: 初始化异常代码 0001_h 至 0016_h: 应用连接的开放异常代码 0100_h: 出错日志块的区域 0101_h: 通讯状态 - 每份协议决定的状态 0102_h: 通讯状态 - 电子邮件的接收状态 0103_h: 通讯状态 - 电子邮件的发送状态 FFFF_h: 清除上述内容 	(如左栏所描述)	用户
(S1) + 3	清除功能指定	<ul style="list-style-type: none"> 指定要清除的功能。 0000_h: [COM.ERR] LED 灯关, 清除出错代码 FFFF_h: 清除出错日志 	0000 _h FFFF _h	用户
(S1) + 4 至 (S1) + 7	系统区域	-	-	-

*1 “由……设置”一栏如下指示:
 • 用户 : 在执行专用指令之前, 由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。
 *2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

功能

- (1) 该指令可关闭[COM. ERR]LED 灯和清除由下表所列的、用于 Un 指定模块的出错信息。

目标项目	目标指定 (S1) + 2	功能指定 (S1) + 3	要清除的错误信息 (缓冲存储器)
初始化错误	0000 _h	0000 _h	<ul style="list-style-type: none"> 初始化异常代码(地址: 69_h) [COM. ERR] LED 灯关闭
接通出错	0001 _h 至 0016 _h	0000 _h	<ul style="list-style-type: none"> 应用连接的开放异常代码(地址: 7C_h, 86_h...) [COM. ERR] LED 灯关闭
出错日志	0100 _h	FFFF _h	<ul style="list-style-type: none"> 出错日志(地址: E3_h 至 174_h)
通讯状态	每份协议决定的 状态	0101 _h	<ul style="list-style-type: none"> 清除通讯状态 (地址: 178_h 至 1FF_h)
	电子邮件的收状态	0102 _h	<ul style="list-style-type: none"> 接收电子邮件(地址: 5871_h 至 5B38_h)
	电子邮件的发状态	0103 _h	<ul style="list-style-type: none"> 发送电子邮件 (地址: 5B39_h 至 5CA0_h)
全部	FFFF _h	FFFF _h	<ul style="list-style-type: none"> 清除上述内容。

- (2) 通过完成位软元件(D1)+0 和 (D1)+1 来检查 ERRCLR 指令是否已经完成。

- (a) 完成位软元件(D1)+0

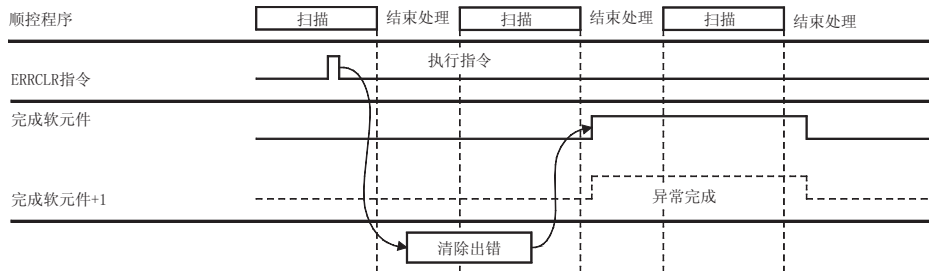
在 ERRCLR 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。

- (b) 完成位软元件(D1)+1

根据 ERRCLR 指令的完成状态接通或断开。

- 正常完成 : 保持关闭不变。
- 异常完成 : 在 ERRCLR 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。

[执行 ERRCLR 指令时的操作]



- (3) 当清除指令从 OFF 切换到 ON 时, 执行 ZP.ERRCLR 功能。

出错

- (1) 当专用指令由于出错而结束时，异常完成信号(D1)+1 接通，且将出错代码储存在完成状态区域(S1)+1 中。

有关出错代码，检查出错及纠正措施，参见以下手册。

<出错代码>

小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)

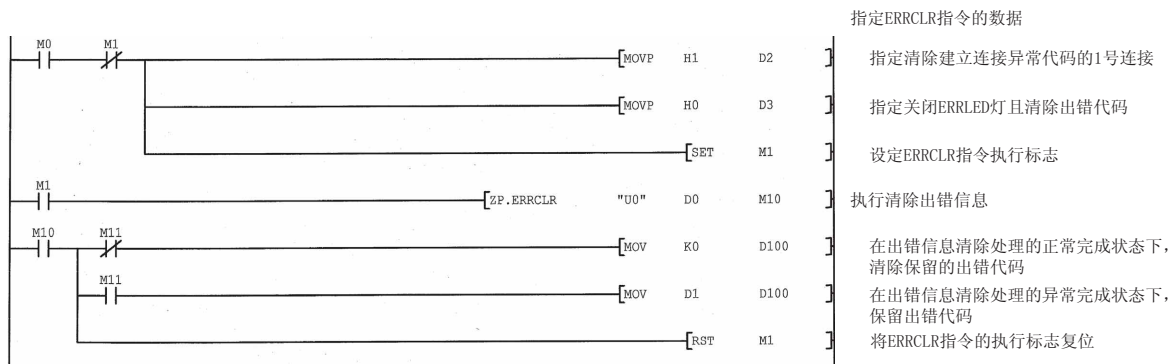
大于等于 C000_H : 本手册 11.3.3 节

程序范例

清除用于 1 号连接的开放异常代码程序：

当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F

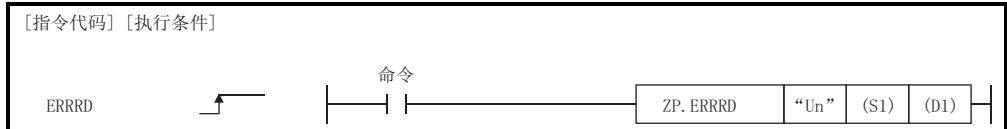
M0: 1号连接接通完成信号



10.7 ERRRD 指令

该指令可读取储存在以太网模块的缓冲存储器中的出错信息。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-		○					-	-	-
(D1)	○		○					-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
“Un”	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	存储控制数据的软元件的起始编号	用户、系统	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站位软元件的起始编号。 如果指令的执行异常结束，(D1)+1 也变为 ON。	系统	位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

软元件	项目	设置数据	设置范围	由(*)设置
(S1) + 0	系统区域	-	-	-
(S1) + 1	完成状态	• 在完成时 储存状态。 0000h: 正常完成 非 0000h : 异常完成(出错代码)(* ²)	-	系统
(S1) + 2	读取指定的信息	• 指定要读取的出错信息。 0 : 初始化异常代码 1 至 16 : 应用连接的开放异常代码	0 1 至 16	用户
(S1) + 3	读取指定的目标 信息	• 指定要读取的出错信息的目标。 0000h: 已发生的最后出错信息	0000h	用户
(S1) + 4	出错信息	• 存储已读取的出错信息。 000h : 无出错 非 0000h: 出错代码(* ²)	0000h	系统
(S1) + 5 至 (S1) + 7	系统区域	-	-	-

- *1 “由……设置”一栏如下指示:
 - 用户 : 在执行专用指令之前, 由用户设置数据。
 - 系统 : PLC CPU 储存专用指令的执行结果。
- *2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

功能

(1) 该指令可读取 Un 指定模块的出错日志中的出错代码和信息。

目标项目	目标指定(S1)+2	功能指定(S1)+3	要清除的错误信息 (缓冲存储器)
初始化错误	0000h	0h	• 初始化异常代码(地址: 69h)
接通错误	0001h 至 0016h	0h	• 应用连接的开放异常代码 (地址: 7C _n , 86 _n ...)

(2) 通过完成的位软元件 (D1)+0 和 (D1)+1 来检查 ERRRD 指令是否已经完成。

(a) 完成位软元件 (D1)+0

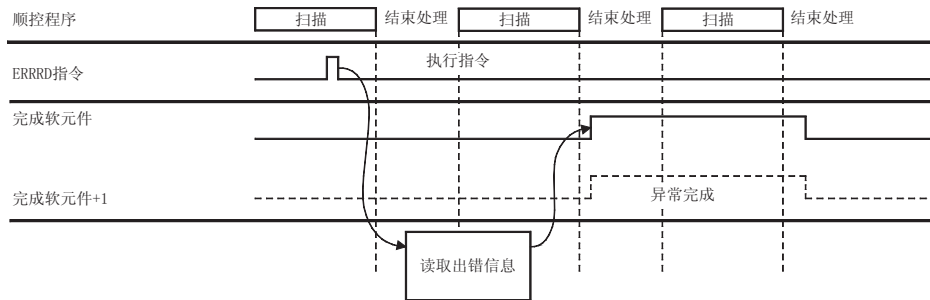
在 ERRRD 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。

(b) 完成位软元件 (D1)+1

根据 ERRRD 指令的完成状态接通或断开。

- 正常完成 : 保持关闭不变。
- 异常完成 : 在 ERRRD 指令完成时, 扫描结束处理时接通, 且在下一个结束处理时断开。

[执行 ERRRD 指令时的操作]



(3) 当读指令从 OFF 切换到 ON 时, 执行 ZP.ERRRD。

出错

(1) 当专用指令由于出错而结束时, 异常完成信号 (D1)+1 开放, 且将出错代码储存在完成状态区域 (S1)+1 中。

有关出错代码, 检查出错及纠正措施, 参见以下手册。

<出错代码>

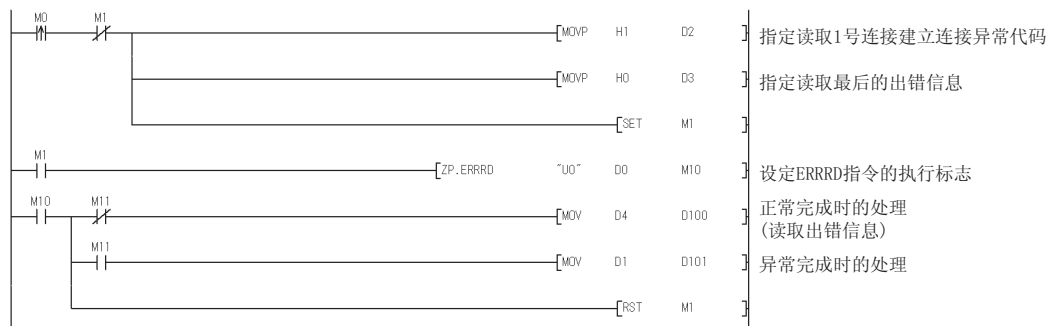
小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)

大于等于 C000_H : 本手册 11.3.3 节

程序范例

读用于 1 号连接的开放异常代码程序:

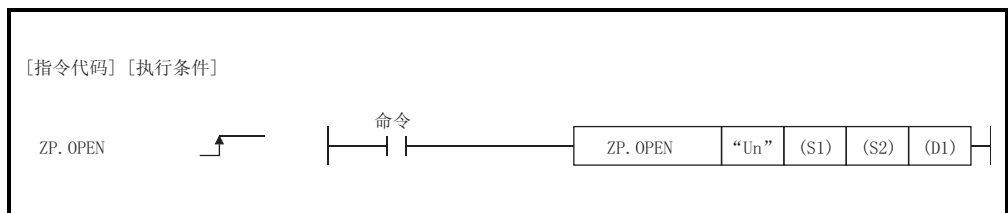
当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F 时



10.8 OPEN 指令

该指令可通过与外部设备建立连接（开放处理）来执行数据通讯。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-	○			-			○	-	-
(S2)	-	○			-			-	-	-
(D1)	○	○			-			-	-	-



设置数据

设置数据	说明	由(*)设置	数据类型
“Un”	以太网模块的输入/输出启动信号 (00 至 FE: 三数字输入/输出信号中的两个最重要的数字)	用户	16 位二进制
(S1)	连接号 (1 至 16)		16 位二进制
(S2)	存储控制数据的软元件的起始编号	用户、系统	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站软元件的起始编号。 如果指令的执行异常结束，(D1)+1 也变为 ON。	系统	位

用于每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

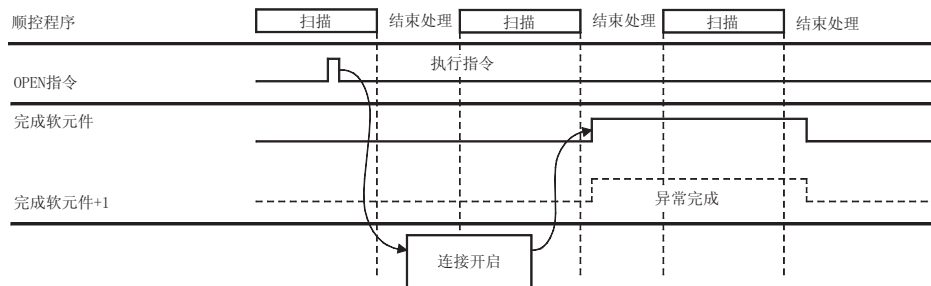
软元件	项目	设置数据	设置范围	由(*)设置																										
(S2) + 0	执行类型/完成类型	<ul style="list-style-type: none"> 指定在连接的开放处理中该使用何种设置，以及 GX Developer 的参数设置值或从 (S2)+2 开始的控制数据的设置值。 0000_h: 使用 GX Developer [开放设置] 中所设置的参数来开放处理。 8000_h: 使用从 (S2)+2 到 (S2)+9 控制数据中所指定的参数来开放处理。 	0000 _h 8000 _h	用户																										
(S2) + 1	完成状态	<ul style="list-style-type: none"> 在完成时 储存状态。 0000_h: 正常完成 非 0000_h : 异常完成 (出错代码) (*²) 	-	系统																										
(S2) + 2	应用设置区域	<ul style="list-style-type: none"> 指定如何使用开放。 <table border="1" style="margin-left: 20px;"> <tr> <td>b15</td><td>b14</td><td>b13</td><td>至</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>至</td><td>b2</td><td>b1</td><td>b0</td> </tr> <tr> <td>6)</td><td></td><td>0</td><td></td><td>5)</td><td>4)</td><td>3)</td><td></td><td>0</td><td></td><td>2)</td><td>1)</td><td></td> </tr> </table> <ol style="list-style-type: none"> 1) 固定缓冲存储器的使用 0: 不执行发送或固定缓冲存储器的通讯 1: 用于接收 2) 指定现存的确认状态 0: 不确认 1: 确认 3) 成对开放设置 0: 不成对 1: 成对 4) 通讯方法 (协议) 0: TCP/IP 1: UDP/IP 5) 固定缓冲存储器的通讯 0: 有顺序 1: 无顺序 6) 开放系统 00: 主动开放或 UDP/IP 10: 非被动开放 11: 被动开放 	b15	b14	b13	至	b10	b9	b8	b7	b6	至	b2	b1	b0	6)		0		5)	4)	3)		0		2)	1)		(如左栏所描述)	用户
b15	b14	b13	至	b10	b9	b8	b7	b6	至	b2	b1	b0																		
6)		0		5)	4)	3)		0		2)	1)																			
(S2) + 3	本地站的端口号	<ul style="list-style-type: none"> 指定本地站的端口号。 	407 _h 至 1387 _h 138B _h 至 FFFE _h	用户																										
(S2) + 4 (S2) + 5	指定 IP 地址	<ul style="list-style-type: none"> 指定外部设备的 IP 地址。 	1 _h 至 FFFFFFFF _h (FFFFFFF _h : 同步广播)	用户																										
(S2) + 6	指定端口号。	<ul style="list-style-type: none"> 指定外部设备的端口号。 	401 _h 至 FFFF _h (FFF _h : 同步广播)	用户																										
(S2) + 7 to (S2) + 9	指定以太网地址	<ul style="list-style-type: none"> 指定外部设备的以太网地址。 (参阅 5.5 节的要点) 	n 000000000000 _h FFFFFFFFFFFF _h	用户																										

*1 “由……设置”一栏如下指示:
 • 用户 : 在执行专用指令之前, 由用户设置数据。
 • 系统 : PLC CPU 储存专用指令的执行结果。
 *2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

功能

- (1) 该指令可执行由 Un 指定的模块与由 (S1) 规定的连接开放处理。
由 (S2)+0 指定开放处理所使用的设置值。
- (2) 通过完成的位软元件 (D1)+0 和 (D1)+1 来检查 OPEN 指令是否已经完成。
 - (a) 完成位软元件 (D1)+0
在 OPEN 指令完成时，扫描的结束处理时接通，而在下一个结束处理时断开。
 - (b) 完成位软元件 (D1)+1
根据 OPEN 指令的完成状态接通或断开。
 - 正常完成 : 保持关闭不变。
 - 异常完成 : 在 OPEN 指令完成时扫描的结束处理时接通，而在下一个结束处理时断开。

[当执行 OPEN 指令时的操作]



- (3) 当开放指令从 OFF 切换到 ON 时，执行 ZP.OPEN。

要点

不要对同一连接同时使用输入/输出信号执行接通/断开处理和使用 OPEN 或 CLOSE 专用指令执行接通/断开处理，否则将导致故障。

出错

- (1) 当专用指令由于出错而结束时，异常完成信号 (D1)+1 接通，且将出错代码储存在完成状态区域 (S2)+1 中。

有关出错代码，检查出错及纠正措施，参见以下手册。

<出错代码>

小于等于 4FFF_H : QCPU(Q 模式)用户手册(硬件设计、维护和检查)

大于等于 C000_H : 本手册 11.3.3 节

程序范例

为 TCP/IP 通讯主动开放 1 号连接的程序：

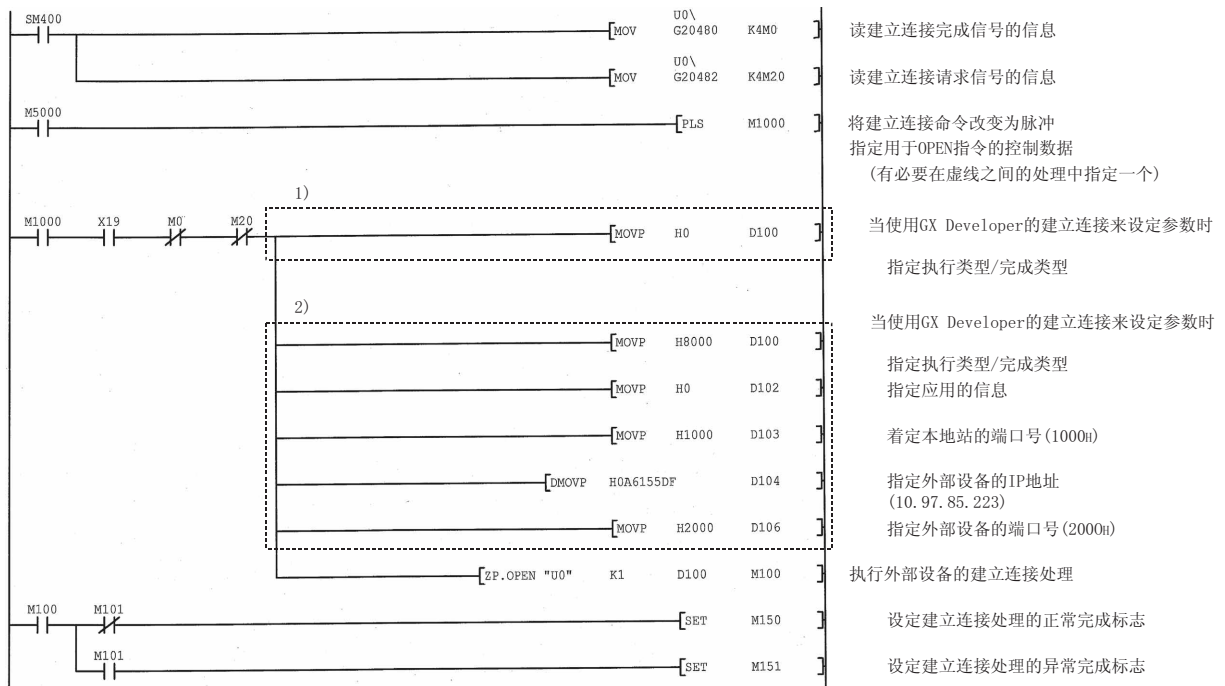
当以太网模块的输入/输出信号为 X/Y00 到 X/Y1F 时

X19: 初始化正常完成的信号

M0: 1号连接建立连接完成的信号

M20: 1号连接建立连接请求信号

1号连接建立连接处理
(TCP/IP主动建立连接)

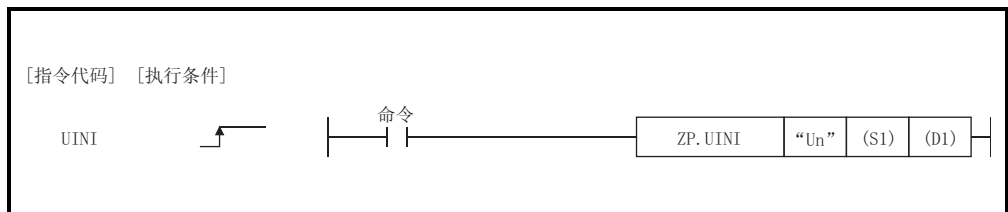


* 有关程序的 1) 部分和 2) 部分，当使用 GX Developer 的“开放设置”参数时，1) 是必要的。当不使用 GX Developer 的“开放设置”参数时，2) 是必要的。

10.9 UINI 指令

该指令用来执行以太网模块的重新初始化处理。。

设置数据	适用软元件									
	内部软元件 (系统、用户)		文件 寄存器	MELSECNET/H、 MELSECNET/10 直接 J□\□		特殊模块 U□\G□	变址寄存器 Zn	常数		其它
	位	字		位	字			K、H	\$	
(S1)	-		○			-		-	-	-
(D1)	○		○			-		-	-	-



设置数据

设置数据	说明	由(‘)设置	数据类型
“Un”	启动以太网模块的输入/输出信号 (00 至 FE: 三位输入/输出信号中两个最重要的数字)	用户	16 位二进制
(S1)	存储控制数据的软元件起始号	用户、系统	16 位二进制
(D1)	指令完成时变为 ON 并持续扫描周期的本地站位软元件的起始编号。 如果指令的执行异常结束, (D1)+1 也变为 ON。	系统	位

每个局部软元件和程序的文件寄存器不能在设置数据时作为软元件使用。

控制数据

软元件	项目	设置数据	设置范围	由(*)设置																								
(S1) + 0	系统区域	-	-	-																								
(S1) + 1	完成状态	<ul style="list-style-type: none"> 在完成时 储存状态。 0000h: 正常完成 非 0000h : 异常完成(出错代码) (*²) 	-	系统																								
(S1) + 2	目标更改说明	<ul style="list-style-type: none"> 指定要更改的参数。 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">b15</td> <td style="text-align: center;">至</td> <td style="text-align: center;">b2</td> <td style="text-align: center;">b1</td> <td style="text-align: center;">b0</td> </tr> <tr> <td style="text-align: center;">0</td> <td></td> <td style="text-align: center;">0</td> <td style="text-align: center;">2)</td> <td style="text-align: center;">1)</td> </tr> </table> <p>1) 指定是否更改本地站 IP 地址。 指定是否更改本地站 IP 地址。 (如果要更改, 则指定使用 (S1)+3 和 (S1)+4 的地址。) 0: 不更改 1: 更改</p> <p>2) 指定是否更改操作设置 指定是否更改操作设置。 (如果要更改, 则指定使用 (S1)+5 的设置。) 0: 不更改 1: 更改</p>	b15	至	b2	b1	b0	0		0	2)	1)	0h 至 3h	用户														
b15	至	b2	b1	b0																								
0		0	2)	1)																								
(S1) + 3 (S1) + 4	本地站 IP 地址	<ul style="list-style-type: none"> 指定本地站的 IP 地址。 	00000001h 至 FFFFFFFEh	用户																								
(S1) + 5	操作设置 (参见 4.7 节)	<ul style="list-style-type: none"> 指定操作设置。 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">b15</td> <td style="text-align: center;">至</td> <td style="text-align: center;">b9</td> <td style="text-align: center;">b8</td> <td style="text-align: center;">b7</td> <td style="text-align: center;">b6</td> <td style="text-align: center;">b5</td> <td style="text-align: center;">b4</td> <td style="text-align: center;">b3</td> <td style="text-align: center;">b2</td> <td style="text-align: center;">b1</td> <td style="text-align: center;">b0</td> </tr> <tr> <td style="text-align: center;">0</td> <td></td> <td style="text-align: center;">0</td> <td style="text-align: center;">5)</td> <td style="text-align: center;">0</td> <td style="text-align: center;">4)</td> <td style="text-align: center;">3)</td> <td style="text-align: center;">2)</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1)</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> </table> <p>1) 通讯数据代码设置 0: 以二进制代码通讯 1: 以 ASCII 代码通讯</p> <p>2) TCP 存在确认设置 0: 使用 Ping 1: 使用 KeepAlive</p> <p>3) 发送帧设置 0: 以太网帧 1: IEEE802.3 帧</p> <p>4) RUN 状态下设置允许/禁止写入 0: 禁止 1: 允许</p> <p>5) 初始化设置 0: 不要等待 OPEN (STOP 状态不可以通讯) 1: 一定要等待 OPEN (STOP 状态可以通讯)</p>	b15	至	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	0		0	5)	0	4)	3)	2)	0	1)	0	0	(如左栏所描述)	用户
b15	至	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																	
0		0	5)	0	4)	3)	2)	0	1)	0	0																	

*1 “由……设置”一栏如下指示:

- 用户 : 在执行专用指令之前, 由用户设置数据。
- 系统 : PLC CPU 储存专用指令的执行结果。

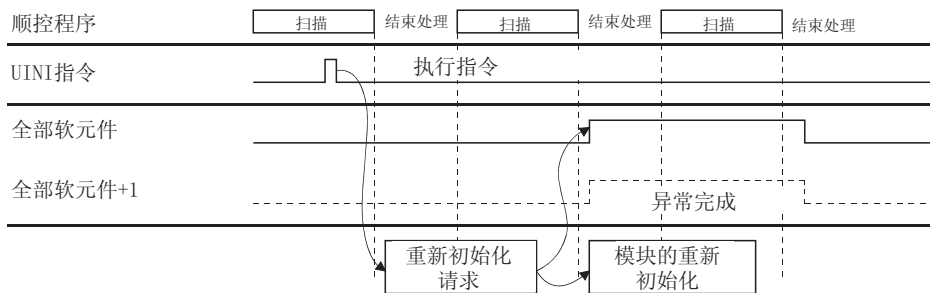
*2 有关异常完成时出错代码的详细说明, 见 11.3 节“出错代码列表”。

要点
<p>只执行以太网模块的重新初始化处理时, 比如: 不更改本地站 IP 地址和操作设置, 应该指定控制数据以使得在执行 UINI 指令前把值 (0H) 存储在 ((S1)+2), 指定更改目标。</p> <p>以太网模块清除之前支持的外部设备地址信息并执行重新初始化处理, 以使得数据通讯重新启动。(初始化正常完成信号 (X19) 为开。)</p>

功能

- (1) 执行 Un 中指定以太网的重新初始化处理。
- (2) 可以通过完成位软元件 (D1)+0 和 (D1)+1 来确认是否完成 UINI 指令。
 - (a) 完成位软元件 (D1)+0
开放完成 UINI 指令扫描的 END 处理，并在下一个 END 处理时关闭。
 - (b) 完成位软元件 (D1)+1
根据 UINI 指令的完成状态开放和关闭。
 - 正常完成 : 保持关闭不变。
 - 异常完成 : 开放完成 UINI 指令扫描的 END 处理，并在下一个 END 处理时关闭。

[正在执行 UINI 指令的操作]



- (3) 当开放指令从 OFF 切换到 ON 时，执行 ZP.UINI 指令。

要点

重新初始化以太网模块时，请注意以下几点。(否则可能引起与外部设备数据通讯的出错。)

- (1) 在执行重新初始化处理前一定要结束所有与外部设备的当前数据通讯并且关闭所有连接
- (2) 不要混淆通过直接写入到缓冲存储器生成的重新初始化处理，比如，通过使用 T0 指令、通过 UINI 指令的重新初始化处理。
并且，当正在进行重新初始化处理时，不要请求另一个重新初始化处理。
- (3) 如果已经更改了以太网模块的 IP 地址，一定要复位外部设备。(如果外部设备仍然保持与通讯设备的以太网地址，那么在更改以太网模块的 IP 地址后可能无法继续通讯。)
- (4) 在冗余系统中，不要用 UINI 指令更改 IP 地址和操作设置。否则可能无法进行正常通讯。
使用 GX Developer 来更改 IP 地址和操作设置。

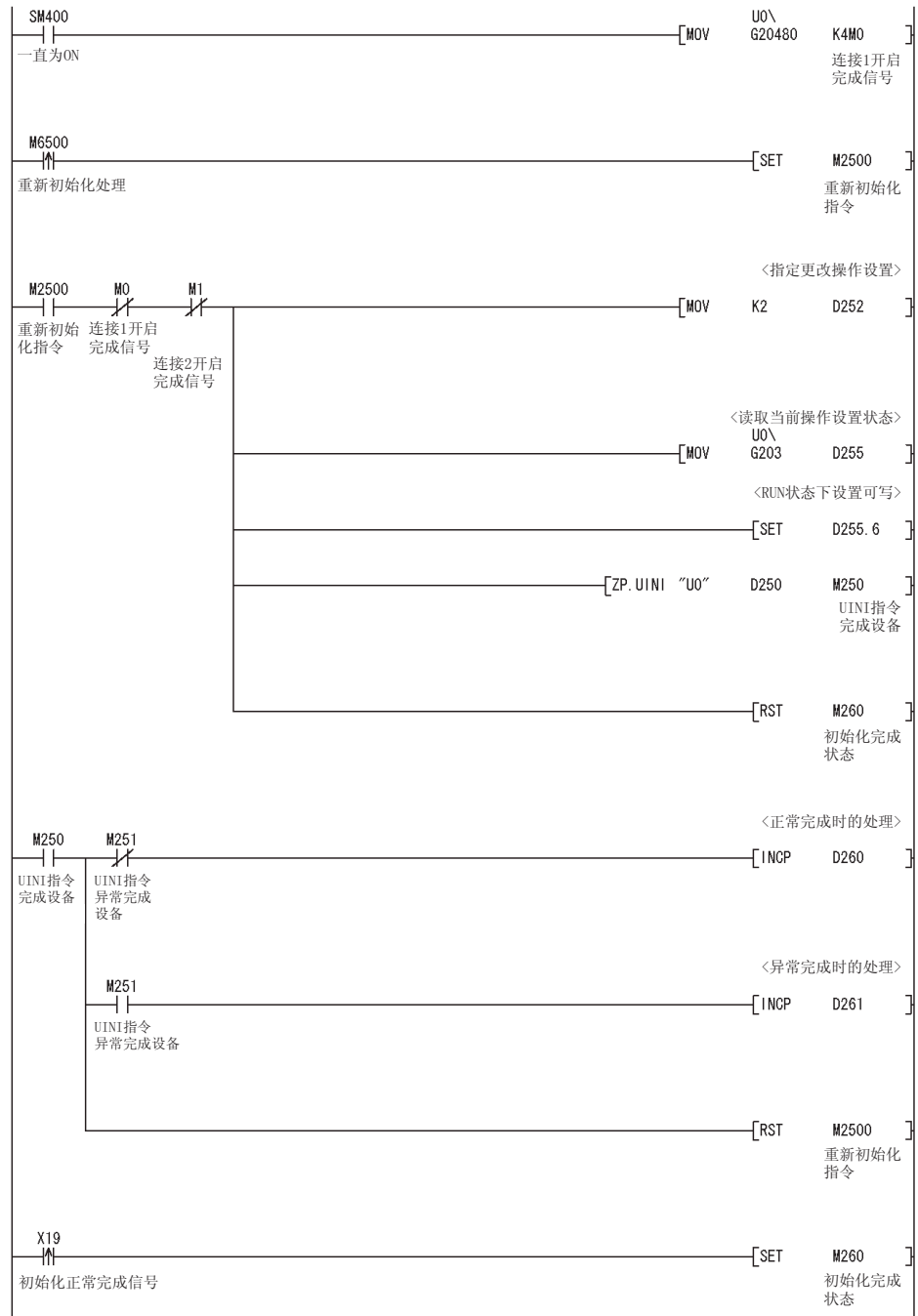
出错

- (1) 当专用指令出现错误，异常完成信号 (D1)+1 开放，并且在完成状态区 (S1)+1 中存储错误代码。
参阅以下关于出错代码的手册，检查错误并采取更正方法。
<错误代码>
小于等于 4FFF_H : QCPU(Q 模式)用户手册 (硬件设计、维护和检查)
大于等于 C000_H : 本手册 11.3.3 节

程序范例

下图显示了执行重新初始化处理的示例程序。

当以太网模块的 I/O 信号为 X/Y00 至 X/Y1F。



要点

该程序范例可在与连接 1 和连接 2 通讯时用来执行重新初始化处理。如果使用其它连接，需要指定每个连接的相应号和位。

11 故障排除

本章说明以太网和外部设备通讯时可能发生的出错内容以及故障检测步骤。
以下方法用来检查以太网端是否存在出错以及出错内容。
使用以下方法之一检查是否存在出错以及出错内容，然后实施纠正措施。

(1) 使用以太网模块前表面上的 LED 显示来检查出错 (见 11.1 节)

LED 开/关状态的显示可以用来检查以太网模块是否出错。

(2) 通过 GX Developer 检查

GX Developer 能用来检查以太网模块的各种状态，以及相应于发生出错内容的出错代码并执行测试。

(a) 以太网诊断 (使用专用画面)

- 1) 用于各种设置状态的监视 (请参阅 11.2 节)
- 2) PING 测试 (请参阅 5.4.1. 节)
- 3) 环路回送测试 (请参阅 5.4.2. 节)
- 4) COM.ERR 关闭 (请参阅 11.1.2 和 11.2.1 节)

(b) 系统监视 (使用专用画面: 请参阅 11.2.2 节)

- 1) 模块的详细信息 模块状态、出错代码等。
- 2) H/W 信息 LED 开/关状态、开关状态等。

(c) 批量缓冲存储器的监视

通过监视以太网模块缓冲存储器来检查出错代码。

(3) 用出错代码来检查出错内容 (请参阅 11.3 节)

可以用上述专用画面中已确认的出错代码或 11.3 中缓冲存储器的监视来检查出错内容。

注意

如果连接多家制造商的设备时发生线路出错或其它出错，用户需用线路分析器等器件来隔离故障部件。

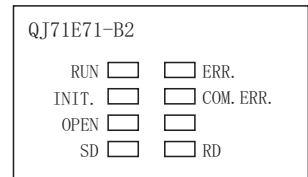
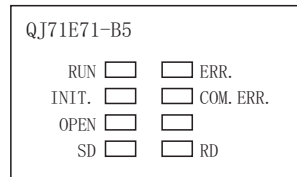
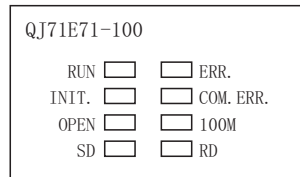
11.1 如何使用 LED 显示来检查出错

本节说明可以通过以太网模块前表面上的 LED 的显示状态来检查出错。

11.1.1 检查出错显示

以下内容可通过以太网模块前表面上的 LED 的显示状态来检查。

<以太网模块的 LED 灯>



	LED 名	检查状态	原因/纠正措施
1	[RUN]	在关闭以太网模块电源后熄灭。(*1)	1) WDT 出错 • 当 WDT (约 300ms) 出错时, 由以太网模块的自诊断功能将 WDT 出错检测信号 (X1F) 接通。 2) 以太网安装故障 • 检查电源模块的电源能量 (5VDC) 是否已不足。 • 断开电源且重新安装模块。
2	[ERR.]	在接通以太网模块电源后点亮。(*1)	1) 模块参数的设置出错 • 用 GX Developer 来检查/纠正以太网模块的参数设置值。 2) PLC CPU 出错 • 当 PLC CPU 的 [RUN] LED 熄灭/闪烁, 或 [ERR.] LED 灯亮时, 检查发生在 PLC CPU 中的出错内容, 且纠正问题。 • 检查以太网模块是否安装在 Q 模式的 PLC CPU。 3) 以太网模块出错 (H/W 出错)
3	[COM. ERR]	在接通以太网模块电源后点亮。	1) 通过以下处理使用存储的出错代码, 来检查出错内容并消除原因。 • 初始化处理 • 固定缓冲存储器的发送处理 • 电子邮件的发送/接收处理 • 开放处理 • 数据通讯处理 • 其它处理 (用于存储在出错日志区中出错代码的处理) 2) 关于出错代码列表, 请参阅 11.3 节。

(转下页)

(接上页)

	LED 名	检查状态	原因/纠正措施
4	[SD]	在数据发送时[SD]LED 不闪烁。	1) [ERR.]或[COM. ERR]LED 灯亮。 • 消除使[ERR.]或[COM. ERR]LED 灯亮的因素。 2) 电缆连接差 • 检查电缆的连接。(*2) 3) 要求核查程序 • 核查发送用顺控程序。
5	[RD]	[RD]LED 灯保持熄灭且不能接收数据。	1) [ERR.]或[COM. ERR]LED 灯亮。 • 消除使[ERR.]或[COM. ERR]LED 灯亮的因素。 2) 电缆连接差 • 检查电缆的连接。(*2) 3) 本地站的 IP 地址设置出错 • 如果电缆连接是正确的,用 GX Developer 核查本地站 IP 地址的每一个设置值、路由设置及子网掩码设置。 4) 要求核查程序 • 核查要发送的顺控程序。

*1 进行硬件测试(H/W 测试)且检查以太网模块是否正常工作。

有关硬件测试的详细说明,请参阅 4.8.2 节“硬件测试”。

*2 对初始化处理完成进行确认,且检查电缆连接和以太网线路中是否存在问题。

有关确认初始化处理完成的详细说明,请参阅 5.4 节“确认初始化处理完成”。(如 5.4 节所述来“确认初始化处理完成”的操作)

要点
[INT]、[OPEN]、[ERR.]和[COM. ERR]LED 灯的亮/灭状态存储在缓冲存储器的模块状态区里(地址: C8H)。 有关详细说明,请参阅 3.8 节“缓冲存储器的应用和分配列表”。

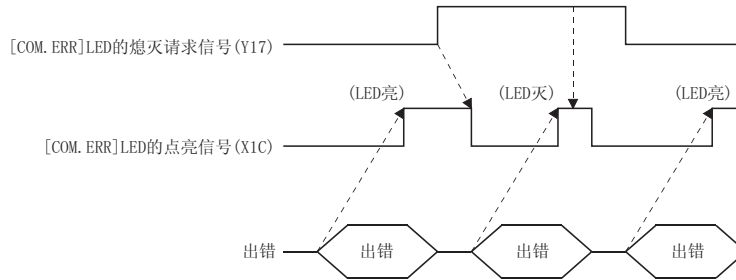
11.1.2 如何熄灭 COM. ERR LED 及如何读/清除出错信息

本节说明如何使[COM. ERR]LED 熄灭及读取/清除出错信息。

(1) 如何用输入/输出信号来使[COM. ERR]LED 熄灭

当外部设备发生通讯出错时，以太网模块前表面[COM. ERR]LED 点亮。(输入/输出信号 X1C: 0N)

(a) 通过接通熄灭请求信号(Y17)来使[COM. ERR]LED 熄灭。



(b) 熄灭请求信号(Y17)点亮的同时，处理熄灭请求信号。

(c) 在缓冲存储器出错日志区中的出错信息，不会因点亮熄灭信号(Y17)而被清除(删除)。

(2) 如何使 GX Developer “以大网诊断” 画面上的[COM. ERR]LED 熄灭 (请参阅 11.2.1 节)

(a) 按下 按钮来使[COM. ERR]LED 熄灭。

(b) 在缓冲存储器的记录区中的出错信息未清除(删除)。

(3) 如何使用专用指令来读/清除出错信息

通过使用以下专用指令，可在任意时间读/清除出错信息。

(a) ERRRD 专用指令

使用此指令，可读初始化异常代码信息或开放异常代码信息。

(b) ERRCLR 专用指令

使用此指令，可以使[COM. ERR]LED 熄灭及清除初始化异常代码/开放异常代码或出错日志。

* 有关专用指令的详细说明，请参阅第 10 章“专用指令”。

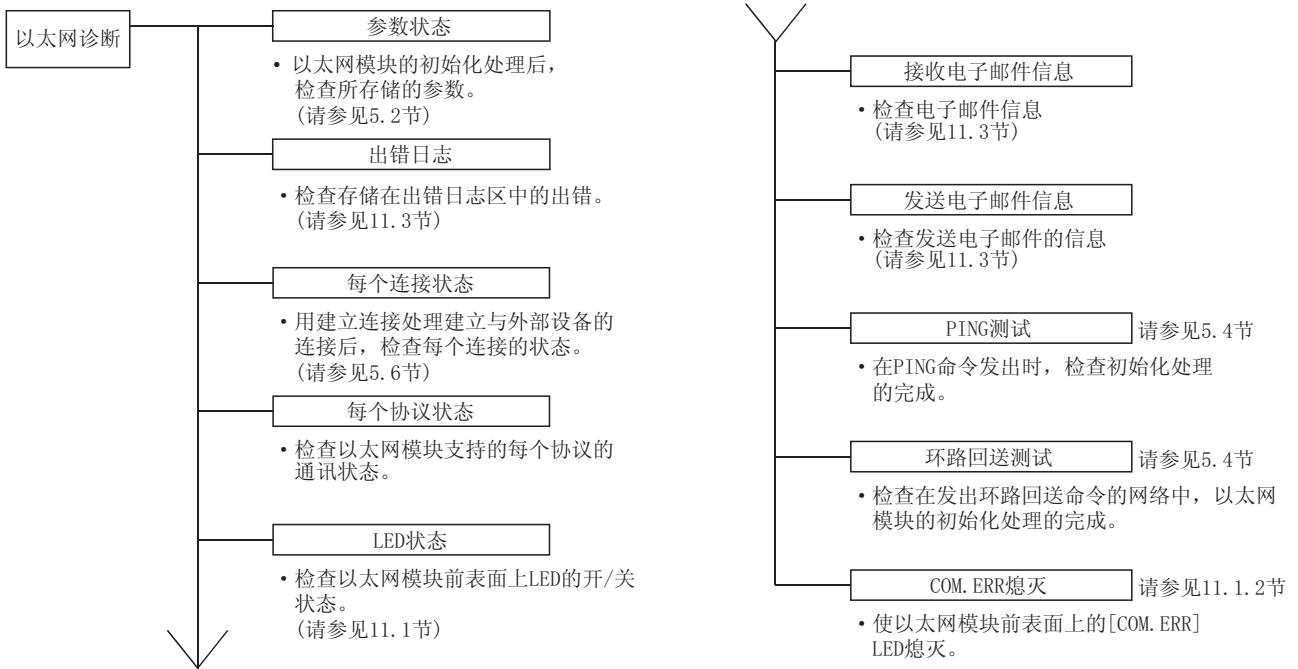
11.2 如何通过 GX Developer 检查出错

可以使用 GX Developer 的功能来检查用于以太网模块的各种设置状态。

(1) 以太网诊断(请参阅 11.2.1 节)

可以使用以太网的诊断功能来检查以太网的模块状态、参数设置、通讯状态、出错记录及其它。

下列为以太网诊断的功能。



(2) 系统监视(请参阅 11.2.2 节)

以太网模块的模块状态可以由系统监视来检查。

(a) 模块的详细信息

可以检查功能版本和出错代码。

(b) H/W 信息

可以检查以太网模块的 LED 的亮/灭状态、连接状态和参数状态。

(3) 缓冲存储器批量监视(请参阅 11.2.4 节)

监视以太网模块的缓冲存储器。

要点
有关可以在“以太网诊断”画面和“系统监视”画面上检查的缓冲存储器，请参阅 11.2.3 节。

11.2.1 以太网诊断

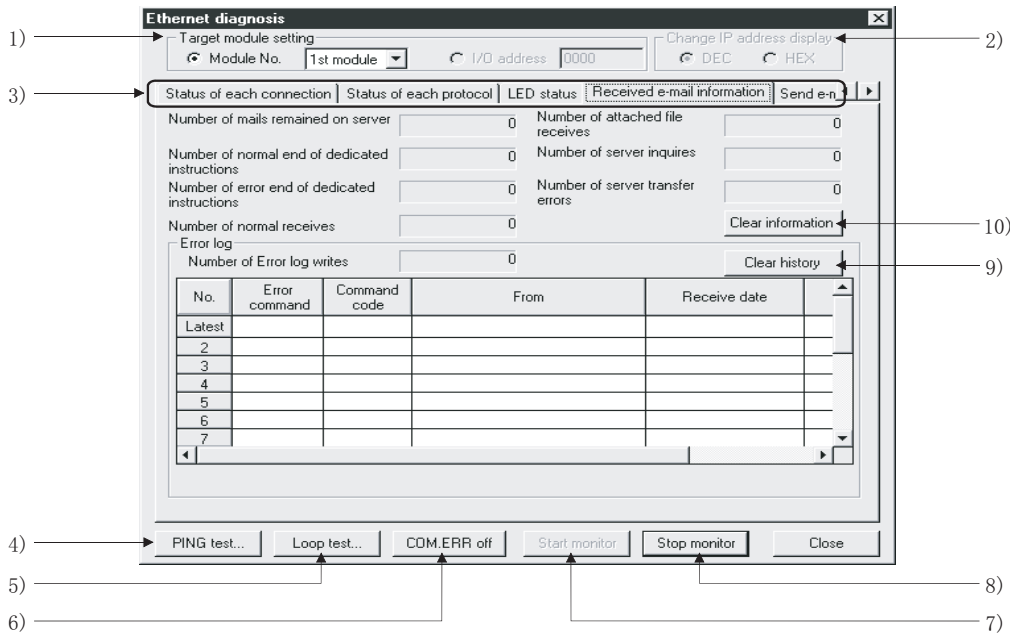
[目的]

可以使用 GX Developer 的以太网诊断功能来检查以太网模块的模块状态、参数状态、通讯状态、出错日志及其它。

[操作步骤]

GX Developer → [诊断] → 以太网诊断

[以太网诊断画面]



[项目说明]

编号	项目	说明	设置范围
1	目标模块设置	指定监视的目标以太网模块。 * 不包括 MELSECNET/H 模块的卡数。	卡 1 到卡 4
2	显示更改的 IP 地址	将显示的 IP 地址在十进制和十六进制之间切换。	十进制/十六进制
3	从各种信息监视之间选择	可以监视以太网模块的各种信息类型。 (有关与显示内容相对应的缓冲存储器, 请参阅 11.2.3 节。)	
4	<input type="button" value="PING test"/>	在外部设备上执行 PING 测试。(请参阅 5.4.1 节)	
5	<input type="button" value="Loop test"/>	执行网络的环路回送测试。(请参阅 5.4.2 节)	
6	<input type="button" value="COM. ERR off"/>	按此按钮可使 [COM. ERR] LED 熄灭。(请参阅 11.1.2 节)	
7	<input type="button" value="Start monitor"/>	按此按钮可执行以太网诊断。 监视期间的更新显示。	
8	<input type="button" value="Stop monitor"/>	按此按钮可停止以太网诊断。 当监视停止时保持显示。	
9	<input type="button" value="Clear history"/>	清除记录。	
10	<input type="button" value="Clear information"/>	清除各类数值。	

要点

在以太网诊断过程中用数据链接指令来访问其它站上的 PLC 时，执行数据指令可能需要一段时间。

执行数据链接指令时采取以下方法并执行以太网诊断。

- 执行 COM 指令。
- 使用特殊寄存器 SD315，确保通讯处理的时间为 2 至 3ms。

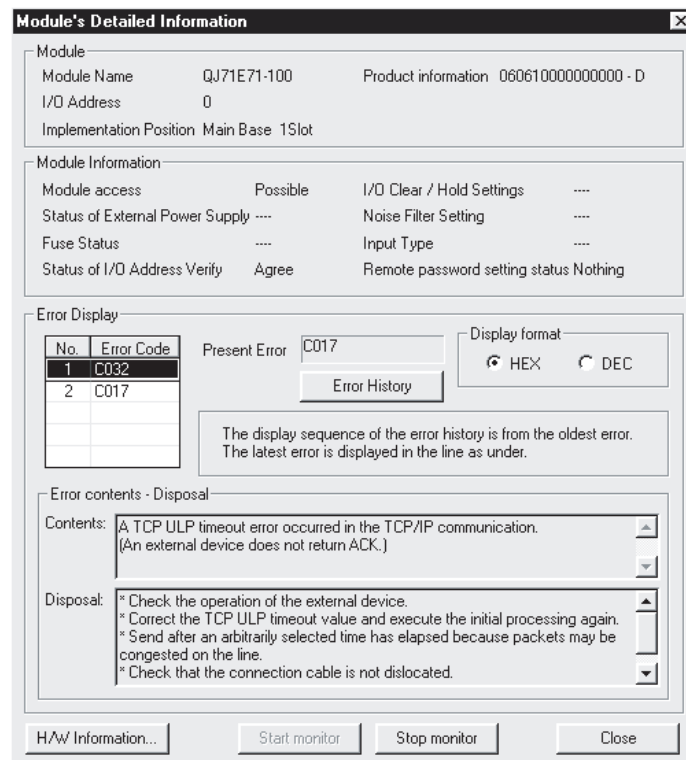
11.2.2 系统监视

可通过系统监视来检查以太网模块的模块状态。

- (1) 可检查用于诊断功能的模块详细信息画面上的模块状态和出错代码。

[起动步骤]

GX Developer → [诊断] → [系统监视] → 模块详细信息



[显示]

- 模块

显示以下信息：

型号名称 ：安装模块的型号名称
 起始 I/O 号 ：目标模块开始输入/输出信号的编号
 安装位置 ：安装模块槽的位置
 产品信息 ：产品信息

* 模块的功能版本显示在产品信息的末尾。
 (举例) 产品信息末尾的“B”表示这是功能版本 B 的模块。

- 装置访问 (模块访问)

当 WDT 的出错信号 (X1F) 断开时，显示允许访问。

- 验证 I/O 地址的状态

显示用户已设置参数的模块是否与安装的模块相匹配。

- 远程口令的设置状态

显示远程口令的设置状态。

- 当前出错

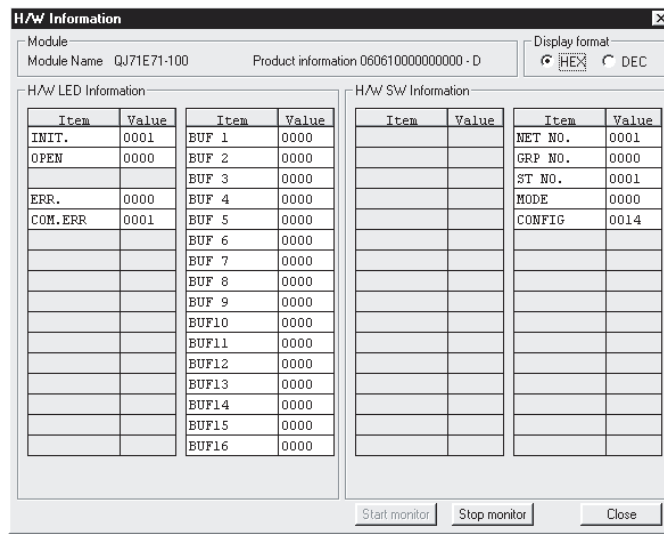
显示最近发生出错的出错代码。

- 出错显示
显示存储在缓冲存储器的出错日志区(地址: E5H, EEH ..., 16CH)中的出错代码。
- 错误详情, 更正方法
对于错误显示中选择的错误代码, 显示错误详情和更正方法。

(2) 在诊断功能的H/W信息画面上检查LED亮/灭状态及操作模式号

[起动步骤]

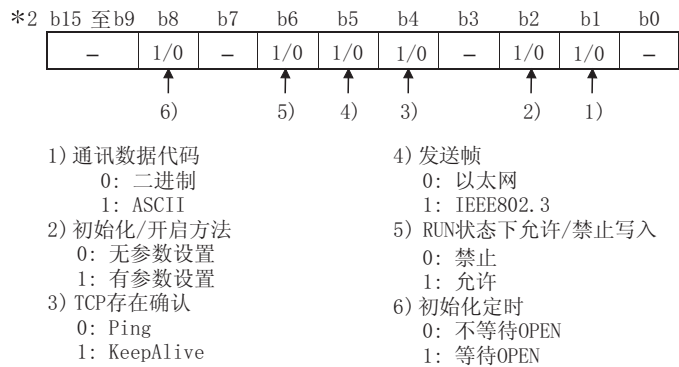
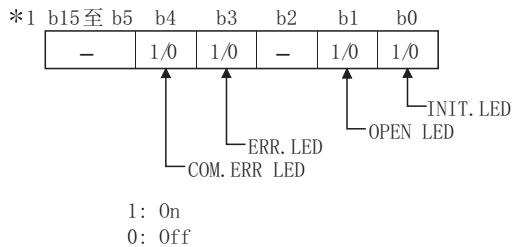
GX Developer → [诊断] → [系统监视] → 模块详细信息 → H/W信息



[显示]

该画面显示存储在下列缓冲存储器存储区中的以太网模块信息。

编号	显示内容	相应的缓冲存储器	地址
1	H/W LED灯的左侧信息(*1)	站中的LED存储区域	C8H(200)
2	H/W LED灯的右侧信息	连接状态的存储区域	5000H(20480)
1	H/W的开关信息	网络编号	本站站网络号/站号的存储区域
2		组号	本站站组号的存储区域
3		站号	本站站网络号/站号的存储区域
4		模式号	开关状态(操作模式设置)的存储区域
5		通讯状态(*2)	通讯状态的存储区域



11.2.3 可用 GX Developer 诊断功能监视的缓冲存储器

下表为可在 GX Developer “以太网诊断”画面和“系统监视”画面上显示的缓冲存储器地址列表。

所提及的有关以太网诊断的显示画面和显示内容，请参阅 11.2.2 节“如何显示系统监视的画面”。

地址 十进制 (十六进制)	可应用的缓冲存储器		以太网的诊断 显示画面	显示内容		
32 (20 _h)	1号连接	目标存在确认 (b1)	每个连接的状态	1号连接	存在确认	
		成对开放 (b7)			成对开放	
		通讯方法 (协议) (b8)			协议	
		开放系统 (b15、b14)			开放系统	
33 至 39 (21 _h 至 27 _h)	2-8号连接 (与1号连接相同)		参数状态	2-8号连接		
105 (69 _h)	初始化出错代码			初始化出错代码		
106 至 107 (6A _h 至 6B _h)	本地站 IP 地址			IP 地址		
108 至 110 (6C _h 至 6E _h)	本地站以太网地址			以太网地址		
116 (74 _h)	自动开放的 UDP 端口号			自动开放 UDP 端口号		
118 (76 _h) * ¹	本地站的网络号/站号			网络号		
				站号		
119 (77 _h) * ¹	本地站的组号			组号		
120 (78 _h)	1号连接	本地站的端口号 L		每个连接的状态	1号连接	本地站的端口号
121 至 122 (79 _h 至 7A _h)		指定 IP 地址				目标 IP 地址
123 (7B _h)		目标端口号	目标端口号			
124 (7C _h)		开放出错代码	开放出错代码			
125 (7D _h)		固定缓冲存储器发送的出错代码	固定缓冲存储器发送的出错代码			
126 (7E _h)		连接的结束代码	连接的结束代码			
130 至 199 (82 _h 至 C7 _h)		2-8号连接 (与1号连接相同)				2-8号连接
200 (C8 _h) * ¹	LED 的亮/灭状态	[INIT.] LED (b0)	LED 的状态	LED 的显示状态	INIT.	
		[OPEN] LED (b1)			OPEN	
		[ERR.] LED (b3)			ERR.	
		[COM. ERR] LED (b4)			COM. ERR	
202 (CA _h) * ¹	开关状态 (操作模式设置)		驱动模式			
203 (CB _h) * ¹	用 GX Developer 设置的状态		-	(检查系统的监视画面。)		
227 (E3 _h)	出错日志块 1	发生的出错号	出错日志	最新	发生的出错号	
229 (E5 _h)		出错代码/结束代码			出错代码/结束代码	
230 (E6 _h)		副标题			副标题	
231 (E7 _h)		命令代码			命令代码	
232 (E8 _h)		连接号			连接号	
233 (E9 _h)		本地站的端口号			本地站的端口号	
234 至 235 (EA _h 至 EB _h)		目标 IP 地址			目标 IP 地址	
236 (EC _h)		目标端口号			目标端口号	
238 至 372 (EE _h 至 174 _h)	出错日志块 2-16 (与出错日志块 1 相同)		2-16号连接			
376 至 377 (178 _h 至 179 _h)	IP	接收 IP 包的计数	每个协议的状态	IP 包	接收总数	
378 至 379 (17A _h 至 17B _h)		由于校验和出错而删除的已接收 IP 包的计数			因校验和出错而注销的总数	
380 至 381 (17C _h 至 17D _h)		发送 IP 包总数			发送总数	
408 至 409 (198 _h 至 199 _h)	ICMP	接收 ICMP 包数	每个协议的状态	ICMP 包	接收总数	
410 至 411 (19A _h 至 19B _h)		由于校验和出错而删除的已接收 ICMP 包数			因校验和出错而注销的总数	
412 至 413 (19C _h 至 19D _h)		发送 ICMP 包数			发送总数	
414 至 415 (19E _h 至 19F _h)		接收 ICMP 包的回声请求总数			回声请求接收的总数	
416 至 417 (1A0 _h 至 1A1 _h)		发送 ICMP 包的回声应答总数			回声应答发送的总数	
418 至 419 (1A2 _h 至 1A3 _h)		发送 ICMP 包的回声请求总数			回声请求发送的总数	

地址 十进制 (十六进制)	可应用的缓冲存储器		以太网的诊断 显示画面	显示内容		
420 至 421 (1A4 _h 至 1A5 _h)	ICMP	接收 ICMP 包的回声应答总数	每个协议的状态	ICMP 包	回声应答接收的总数	
440 至 441 (1B8 _h 至 1B9 _h)		接收 TCP 信息包数			TCP 信息包	接收总数
442 至 443 (1BA _h 至 1BB _h)	TCP	由于校验和出错而删除的已接收 TCP 信息包数		因校验和出错而注销的总数		
444 至 445 (1BC _h 至 1BD _h)		发送 TCP 信息包总数		发送总数		
472 至 473 (1D8 _h 至 1D9 _h)	UDP	接收 UDP 信息包总数		UDP 信息包	接收总数	
474 至 475 (1DA _h 至 1DB _h)		由于校验和出错而删除的接收 UDP 信息包数			因校验和出错而注销的总数	
476 至 477 (1DC _h 至 1DD _h)		发送 UDP 信息包的总计数			发送总数	
20480(5000 _h) *1	1-16 号连接的开放完成信号			-	(检查系统的监视画面)	
22560 至 22639 (5820 _h 至 586F _h)	9-16 号连接 (与 1 号连接相同 (地址: 120 至 126(78 _h 至 7E _h)))		每个连接的状态	9-16 号连接		
22640(5870 _h)	接收	保存在服务器中的邮件数	接收的电子邮件信息	保存在服务器中的邮件数		
22641(5871 _h)		专用指令正常完成数		专用指令正常结束数		
22642(5872 _h)		专用指令异常完成数		专用指令出错结束数		
22643(5873 _h)		正常接收数		正常接收数		
22644(5874 _h)		附件接收数		附件接收数		
22645(5875 _h)		服务器询问数		服务器的询问数		
22646(5876 _h)		服务器通讯出错数		服务器传输出错数		
22647(5877 _h)		出错日志写入数		写入的出错日志数		
22649(5879 _h)		出错日志块 1		出错代码 命令代码 发件人 日期 主题	最近的出错日志	出错代码
22650(587A _h)						命令代码
22651 至 22658 (587B _h 至 5882 _h)						发件人
22659 至 22662 (5883 _h 至 5886 _h)						接收日期
22663 至 22692 (5887 _h 至 58A4 _h)						主题
22693 至 23352 (58A5 _h 至 5B38 _h)		出错日志块 2-16 (与出错日志块 1 相同)		出错日志 2-16		
23353(5B39 _h)	发送	专用指令正常完成数	发送的电子邮件信息	专用指令正常结束数		
23354(5B3A _h)		专用指令异常完成数		专用指令出错结束数		
23355(5B3B _h)		正常完成的邮件数		正常结束邮件数		
23356(5B3C _h)		附件发送数		发送附件数		
23357(5B3D _h)		发送至服务器的数		发送服务器数		
23358(5B3E _h)		异常完成的邮件数		邮件的出错结束数		
23359(5B3F _h)		出错日志写入数		出错日志写入数		
23361(5B41 _h)		出错日志块 1		出错代码 命令代码 收件人 日期 主题	最新的出错日志	出错代码
23362(5B42 _h)						命令代码
23363 至 23370 (5B43 _h 至 5B4A _h)						收件人
23371 至 23374 (5B4B _h 至 5B4E _h)						发送日期
23375 至 23404 (5B4F _h 至 5B6C _h)						主题
23405 至 23712 (5B6D _h 至 5CA0 _h)		出错日志块 2-16 (与出错日志块 1 相同)		出错日志 2-16		

*1 可以通过 GX Developer 的系统监视画面来监视。请参阅 11. 2. 2 节。

11.2.4 使用缓冲存储器批量监视功能来检查出错信息

本节说明如何通过 GX Developer 来检查以太网模块的出错。

存储在以太网模块的缓冲存储器中的出错代码能通过使用 GX Developer 的“缓冲存储器批量监视”功能来监视。

[操作步骤]

(第 1 步)选择 GX Developer 菜单栏的[在线]-[监视]-[缓冲存储器批量]，显示“缓冲存储器批量监视”画面。

(第 2 步)输入[模块的开始地址:]。

对模块的开始地址，要输入被监视以太网模块的启动 I/O 信号的前 3 位数(主 4 位数表示)。

(举例)

启动 I/O 信号：当监视 X/Y0020 到 003F 模块时，输入“002”

(第 3 步)输入[缓冲存储器的开始地址:]。

使用选定的输入格式(十进制/十六进制)，输入要监视的缓冲存储器的地址。有关存储出错代码的缓冲存储器地址列表，请参阅 11.3 节“出错代码列表”。

(范例)

当监视初始化异常代码时(缓冲存储器的存储地址：69H)：

输入“69”+“十六进制数”

(第 4 步)按下 按钮。

在指定地址后，显示缓冲存储器的内容。(在上述例子中，会显示 69H 及随后的地址。)

注意

显示格式能修改如下：

监视的格式 ：位&字/多点位/多点字

显示 ：16 位整数/32 位整数/实数 IASCII 字符

数值 ：十进制/十六进制

有关详细说明，请参阅 GX Developer 的“操作手册”。

11.3 出错代码列表

本节介绍的出错代码(异常代码)是代表当以太网模块和外部设备通讯时,在每个处理中发生的出错,以及那些来自于本地站的 QCPU 请求处理中所产生的出错的。
并说明出错内容和出错处理步骤。

	出错类型	说明	存储在缓冲存储器中的出错代码	阐述	
1	在初始化处理中所发生的出错	<ul style="list-style-type: none"> 设置值出错 初始化处理出错 	69 _H : 初始化出错代码 (通讯状态的存储区域)	11.3.3 节	
2	在开放处理中所发生的出错	<ul style="list-style-type: none"> 设置值出错 连接处理出错 	7C _H : 5824 _H : 连接出错代码 (通讯状态的存储区域)		
3	发送至外部设备的固定缓冲存储器中所发生的出错	<ul style="list-style-type: none"> 指定数据出错 发送出错 	7D _H : 5825 _H : 固定缓冲存储器发送的出错代码 7E _H : 5826 _H : 连接的结束代码 (通讯状态的存储区域)		
4	与外部设备通讯的固定缓冲存储器中所发生的出错	<ul style="list-style-type: none"> 指定数据出错 通讯出错 (不包括上述第 3 条)	7E _H : 5826 _H : 连接的结束代码 (通讯状态的存储区域)		
5	当与外部设备通讯时,返回至外部设备的出错	<ul style="list-style-type: none"> 返回至固定缓冲存储器通过程的出错(结束代码) 		11.3.1 节	
		<ul style="list-style-type: none"> 返回至随机访问缓冲存储器通过程的出错(结束代码) 			
		<ul style="list-style-type: none"> 返回至使用 MC 协议通过程的出错 	当使用与 3E 帧兼容命令时的结束代码	11.3.3 节	
			当使用与 1E 帧兼容命令时的结束代码	11.3.1 节	
6	当与外部设备通讯时所发生的出错(包括说明栏中解释的原因,那些存储在出错日志区中的出错代码。	<ul style="list-style-type: none"> 指定数据出错 不能确认识错源的出错 当使用随机访问缓冲存储器通讯时所发生的出错 当使用 MC 协议通讯时所发生的出错 	E5 _H : (出错日志区域)	11.3.3 节	
				CPU 模块手册	
7	当与使用文件传送(FTP)功能的外部设备通讯时所发生的出错(响应命令)	<ul style="list-style-type: none"> 指定数据出错 通讯出错,等 			
8	当与 Web 功能通讯时所发生的出错	<ul style="list-style-type: none"> 通讯出错 	5101 _H . . . (HTTP 状态存储区域 a)	11.3.3 节	
9	当接收电子邮件时所发生的出错	<ul style="list-style-type: none"> 设置数据出错 接收出错 	5870 _H : 接收 (电子邮件的状态存储区域)		
10	当发送电子邮件时所发生的出错	<ul style="list-style-type: none"> 设置数据出错 发送出错 	5B39 _H : 发送 (电子邮件的状态存储区域)		
11	当在通讯时使用来自于本地站 QCPU 的数据链接指令所发生的出错	<ul style="list-style-type: none"> 指定数据出错 通讯出错 	未存储 (在指令的完成状态区域存储)		
12	当在通讯时使用来自于本地站 QCPU 的专用指令所发生的出错	<ul style="list-style-type: none"> 指定数据出错 通讯数据出错 	未存储 (在指令的完成状态区域存储)		

* 有关当使用 MX Component 进行数据通讯时,通过 MX Component 所返回的出错代码的详细说明,请参阅 MX Component 编程手册。

- (1) 初始化出错代码 (地址: 69H)
 - (a) 当执行初始化处理时, 此地址存储产生的出错代码。
 - (b) 当初始化异常完成信号 (X1A) 0N 时, 以二进制数值来存储出错代码。
 - (c) 当初始化完成信号 (X19) 0N 时, 可清除出错代码, 也可通过以下操作来清除出错代码。
 - 1) PLC CPU 复位或断开 PLC 电源
 - 2) 通过顺控程序将“0”写入初始化出错代码的存储区。

- (2) 打开出错代码
(连接号: 1 至 8; 地址: 7CH 至 C1H)
(连接号: 9 至 16; 地址: 5824H 至 5869H)
 - (a) 这些地址存储应用连接的开放处理结果。
 - (b) 开放处理的结果以二进制数值存储。
0 : 正常完成
除 0 外 : 异常完成 (开放异常检测信号 (X18): 连接)
 - (c) 出错代码由以下操作来清除。
 - 1) 将开放的出错重新开放
 - 2) 将 PLC CPU 复位或断开 PLC 电源

- (3) 固定缓冲存储器发送出错代码
(连接号: 1 至 8; 地址: 7DH 至 C2H)
(连接号: 9 至 16; 地址: 5825H 至 586AH)
 - (a) 在使用可应用连接的固定缓冲器通讯期间, 当发送至外部设备的数据发生出错时, 这些地址存储产生的出错代码。
 - (b) 当下一个数据发送正常完成时, 清除发送的出错数据。

- (4) 连接的结束代码
(连接号: 1 至 8; 地址: 7EH 至 C3H)
(连接号: 9 至 16; 地址: 5826H 至 586BH)
 - (a) 在使用可应用连接的固定缓冲器通讯期间, 这些地址将作为响应存储从外部设备返回的结束代码。
 - (b) 通过布置特殊的外部设备, 决定如何处理响应的结束代码。

(5) 出错日志区域(地址: E0H 至 1FFH)

此区域存储以下出错。

- 不能确认出错源的出错
- 在使用随机访问缓冲存储器通讯期间所发生的出错
- 在使用 MC 协议通讯期间所发生的出错

(a) 发生的出错数(地址: E3H)

- 1) 此地址存储寄存在出错日志块区中的出错数。
- 2) 当出错的发生超过 65536 次时, 计数停止在 FFFFH (65535)。

(b) 出错日志的写入指针(地址: E4H)

- 1) 此地址存储最新出错日志寄存的出错日志块数。
 0 : 无错。(无出错日志寄存)
 1 以上 : 最新出错日志寄存的出错日志块数

*如果指针值为“16”, 则意味着最新的出错日志寄存在出错日志块的 16 区中。

- 2) 如果发生的出错大于等于 17 个, 则从出错日志块的 1 区重新开始寄存出错日志。

要点	
	(1) 一个出错日志区包含有相同数据顺序的 16 个出错日志区。 (2) 如果停止出错计数并且不再存储, 则出错信息仍然存储在以下区域: <ul style="list-style-type: none"> • 出错日志写入指针存储区 • 出错日志区

(c) 出错日志块-出错代码/结束代码(地址: 从 E5H 开始)

- 1) 在出错代码/结束代码区, 存储表示出错内容的出错代码。(请参阅 11.3 节)

(d) 出错小副标题(地址: 从 E6H 开始)

- 1) 异常信息的副标题代码存储在相应区域的 0-7 位。(“0”存储在 8-15 位)。
- 2) 对于低于 TCP 和 UDP 等级的出错, 存储“0”。

(e) 出错日志块-命令代码(地址: 从 E7H 开始)

- 1) 此块存储异常信息的命令代码、或需求类型的每个较低字节值和数据链接指令的子请求类型。



- 2) 在以下情况存储“0”。
 - 不包含命令代码的信息
 - 低于 TCP 和 UDP 级的出错(因为它们的命令未知)

- (f) 出错日志块-连接号(地址: 从 E8_H 开始)
- 1) 异常连接号存储在相应区域的 0-7 位中。(“0”存储在 8-15 位中)。
 - 2) 对于低于 TCP 和 UDP 等级的出错, 存储“0”。
- (g) 出错日志块-本地站的端口编号(地址: 从 E9_H 开始)
- 1) 当发生出错时, 此块存储本地站的端口号。
 - 2) 对于低于 TCP 和 UDP 等级的出错, 存储“0”。
- (h) 出错日志块-指定 IP 地址(地址: EA_H 及从 EB_H 开始)
- 1) 当发生出错时, 此块存储外部设备的 IP 地址。
 - 2) 在以下场合存储“0”。
 - 对于低于 IP 级的出错
 - 当通过 PLC CPU 中继来执行出错响应时
- (i) 出错日志块-指定端口编号(地址: 从 EC_H 开始)
- 1) 当发生出错时, 此块存储外部设备的端口号。
 - 2) 对于低于 TCP 和 UDP 等级的出错, 存储“0”。
- (j) 用于每个协议的状态(地址: 178_H 至 1FF_H)
- 1) 此地址存储用于每个协议条件可应用内容所发生的计数。
(由以太网模块计数。)
 - 2) 当计数值超过两个字时, 计数停止在 FFFFFFFF_H (4294967295) 处。

要点

<p>当安装以太网模块的站通电或复位时, 清除存储在缓冲存储器中的数值。(在初始化处理中, 这些数值未清除。)</p>

<p>通常没有必要读此区域; 所以, 在维护期间需要时可读。</p>

(6) HTTP 状态存储区域(地址: 5101H 至 5177H)

(a) 出错日志指针(地址: 5101H)

- 此地址存储最新出错日志寄存的出错日志块号。

0 : 无错。(无出错日志寄存)

1 或 1 以上: 最新出错日志寄存的出错日志块号

* 如果指针值为“16”，则意味着最新的出错日志寄存在出错日志块 16 区中。

- 当发生的出错大于等于 17 时，则从出错日志块 1 区重新开始寄存出错日志。

要点	
	一个出错日志块区域包括有同样数据步骤的 16 个出错日志块。

(b) 日志计数器(HTTP 响应代码 100 至 199)(地址: 5102H)

日志计数器(HTTP 响应代码 200 至 299)(地址: 5103H)

日志计数器(HTTP 响应代码 300 至 399)(地址: 5104H)

日志计数器(HTTP 响应代码 400 至 499)(地址: 5105H)

日志计数器(HTTP 响应代码 500 至 599)(地址: 5106H)

日志计数器存储以太网模块将 HTTP 响应代码发送给网络浏览器的次数。

(c) 出错日志块: HTTP 响应代码(地址: 从 5108H 开始)

当发生出错时，此出错日志块存储 HTTP 的响应代码。(请参阅 11.3 节)

(d) 出错日志块: 通讯对应方的 IP 地址(地址: 5109H、510AH 及以上)

当发生出错时，此出错日志块存储服务器的 IP 地址。

(e) 出错日志块: 出错发生时间(地址: 从 510BH 开始)

当在 BCD 代码中发生出错时，此块存储时间。

b15 至 b8	b7 至 b0
--------------------------------	-------------------------------

月(01到12H)	年的低2位数(00H到99H)
-----------	-----------------

b15 至 b8	b7 至 b0
--------------------------------	-------------------------------

小时(00H到23H)	日期(01H到31H)
-------------	-------------

b15 至 b8	b7 至 b0
--------------------------------	-------------------------------

秒(00H到59H)	分(00H至59H)
------------	------------

b15 至 b8	b7 至 b0
--------------------------------	-------------------------------

年的高2位数(00H到99H)	星期几(0-6)
-----------------	----------

(7) 电子邮件状态的存储区域(地址: 5870H 至 5FFFH)

*当存储计数超过 FFFF_H 次时, 计数重新由 0_H 开始。

(a) 接收电子邮件状态的存储区域(地址: 5870H 至 5B38H)

1) 保存在服务器中的邮件数(地址: 5870H)

- 当以太网模块向接收邮件的服务器询问时, 此区域存储保存的邮件数。

0 : 服务器未收到邮件

1 至 15 : 保存在服务器中的邮件数

16 : 服务器中的邮件数为 16 或更多

2) 专用指令正常完成时的计数(地址: 5871H)

- 此区域存储正常完成的专用指令次数的累积计数。

0 : MRECV 指令未在正常完成时执行或执行无结果。

1 或 1 以上 : MRECV 指令正常完成时的累积计数

3) 专用指令异常完成时的计数(地址: 5872H)

- 此区域存储异常完成的专用指令(MRECV)次数的累积计数。

0 : MRECV 指令未执行或执行异常完成无结果。

1 或 1 以上 : MRECV 指令异常完成的累积计数

4) 正常接收计数(地址: 5873H)

- 当以太网向邮件缓冲数据区传送接收到的邮件时, 此地址存储累积的计数。

0 : 无邮件传送。

1 或 1 以上 : 传送完成的正常邮件数。

5) 接收的附件计数(地址: 5874H)

- 此地址存储以太网模块接收带有附件邮件的次数的累积计数。

0 : 未收到带有附件的邮件。

1 或 1 以上 : 接收到的带有附件的正常完成邮件数。

6) 询问服务器计数(地址: 5875H)

- 此地址存储根据参数设置, 询问接收邮件服务器的累积计数。(请参阅用户手册(适用篇)的第 2 章。)

0 : 对服务器无询问。

1 或 1 以上 : 询问服务器的累积计数。

7) 服务器通讯出错的计数(地址: 5876H)

- 当询问接收邮件的服务器时, 此地址存储发生返回通讯出错的累积计数。

0 : 服务器之间无通讯出错或无询问。

1 或 1 以上 : 发生通讯出错的累积计数。

- 8) 出错日志写计数(地址: 5877H)
 - 此地址存储在接收出错日志区域中寄存的累积计数。
 - 0 : 无出错或未询问服务器。
 - 1或1以上 : 出错日志块区域中寄存的累积计数。
- 9) 接收写入出错日志的指针(地址: 5878H)
 - 此地址存储最新出错日志寄存的出错日志块号。
 - 0 : 无出错(出错日志未寄存)
 - 1或1以上 : 最新的出错日志寄存的出错日志块号
 - * 如果指针值为“16”，则意味着最新的出错日志寄存在出错日志块16区中。
 - 当发生的出错大于等于17时，则从出错日志块的1区重新开始寄存出错日志。

要点	一个出错日志块区域包括有同样数据步骤的16个出错日志块。
-----------	------------------------------

- 10) 出错日志块-出错代码(地址: 从 5879H 开始)
 - 此块存储表示出错内容的出错代码。(请参阅 11.3. 节)
- 11) 出错日志块-命令代码(地址: 从 587AH 开始)
 - 此块存储引起信息出错的系统命令代码。
- 12) 出错日志块-从(地址: 从 587BH 开始)
 - 此块以 ASCII 代码的字符来存储在与邮件服务器的通讯期间，从引起邮件出错的发送源邮件地址开始的八个字。
 - (范例)
 - 如果发送源的邮件地址是 use@from.add.sample.co.jp，
 - 则 use@from.add.sam 作为 ASCII 代码的字符存储。
- 13) 出错日志块-日期(地址: 从 5883H 开始)
 - 此块以 BCD 码存储电子邮件接收的日期和时间。

b15	至	b8	b7	至	b0
月 (01h到12h)			年的低2位数 (00h到99h)		
b15	至	b8	b7	至	b0
小时 (00h到23h)			日 (01h到31h)		
b15	至	b8	b7	至	b0
秒 (00h到59h)			分 (00h到59h)		
b15	至	b8	b7	至	b0
年的高2位数 (00到99)			星期 (0到6)		

- 14) 出错日志块-主题(地址: 从 5887H 开始)
 - 此块可存储电子邮件标题开始的 30 个字。
 - 如果标题中包含有字母和 ASCII 码以外的字符，则标题不能存储。

- (b) 发送电子邮件状态的存储区域(地址: 从 5B39H 至 5CA0H)
- 1) 专用指令的正常完成计数(地址: 5B39H)
 - 此区域存储专用指令(MSEND)正常完成次数的累计计数。
0 : MSEND 指令未执行或执行无正常完成结果。
1 或 1 以上 : MSEND 指令正常完成时的累计计数。
 - 2) 专用指令的异常完成计数(地址: 5B3AH)
 - 此区域存储专用指令(MSEND)异常完成次数的累积计数。
0 : MSEND 指令未执行或执行无异常完成结果。
1 或 1 以上 : MSEND 指令异常完成时的累计计数。
 - 3) 正常完成的邮件数(地址: 5B3BH)
 - 此地址存储以太网模块将发送邮件传送给发送邮件服务器的次数的累积计数。
0 : 无邮件发送。
1 或 1 以上 : 发送邮件的正常完成数。
 - 4) 发送附件的计数(地址: 5B3CH)
 - 此地址存储以太网模块发送带有附件邮件的次数的累积计数。
0 : 无带有附件的邮件发送。
1 或 1 以上 : 带有附件邮件的正常发送次数。
 - 5) 发送至服务器的计数(地址: 5B3DH)
 - 此地址存储发送至发送邮件服务器的累积计数。
0 : 无邮件发送至服务器。
1 或 1 以上 : 发送至服务器的累积计数。
 - 6) 异常完成的邮件数(地址: 5B3EH)
 - 此地址存储当请求发送至传送邮件服务器时, 返回所发生通讯出错的累积计数。
0 : 服务器之间无通讯出错, 或未传送。
1 或 1 以上 : 发生的通讯出错的累积计数。
 - 7) 写入的出错日志计数(地址: 5B3FH)
 - 此地址存储对发送出错日志块区域寄存的累积计数。
0 : 无出错, 或未询问服务器。
1 或 1 以上 : 对出错日志块区域寄存的累积计数。

- 8) 发送出错日志写入指针(地址: 5B40_H)
- 此地址存储最新发送出错日志寄存的出错日志块区域号。
 - 0 : 无出错。(发送的出错日志未寄存)
 - 1 或 1 以上 : 最新发送出错寄存的出错日志块的区域号。
 - * 如果指针值为“8”，则意味着最新出错日志寄存在发送出错日志块的 8 区中。
 - 当发生的发送出错次数大于等于 9 时，则从发送出错日志块的 1 区重新开始寄存发送出错日志。

要点	发送出错日志块区域包括有同样数据项目步骤的 8 个出错日志块。
-----------	---------------------------------

- 9) 出错日志块-出错代码(地址: 从 5B41_H 开始)
- 此块存储代表出错内容的出错代码。(请参阅 11.3 节)
- 10) 出错日志块-命令代码(地址: 从 5B42_H 开始)
- 此块存储引起出错信息的系统命令代码。
- 11) 出错日志块-收件人(地址: 从 5B43_H 开始)
- 当以 ASCII 码字符与邮件服务器的通讯期间，此块存储引起电子邮件出错的发送源邮件地址的前 8 个字。
- (举例)
- 如果发送源邮件的地址是
 “use@from.add.sample.co.jp”，
 “use@from.add.sam” 作为 ASCII 码字符存储。
- 12) 出错日志块-日期(地址: 从 5B4B_H 开始)
- 此块以 BCD 码来存储发送电子邮件的时间和日期。
 - 日期和时间的存储步骤与 (a) 13) 中所示的接收电子邮件的日期和时间的存储步骤相同。
- 13) 出错日志块-主题(地址: 从 5B4F_H 开始)
- 此块存储电子邮件的标题最前的 15 个字。

结束代码	说明	处理	通讯功能		
			固定	随机	MC
56 _n	<ul style="list-style-type: none"> 从外部进行的软元件指定不正确。 	<ul style="list-style-type: none"> 纠正指定的软元件。 			○
57 _n	<ul style="list-style-type: none"> 用于外部设备指定命令的点数超过了用于每次处理的处理点(每次通讯中能执行的处理数)的最大数。 从起始地址到(起始软元件号和起始步号)指定点的地址个数超过了最大地址个数(软元件号和步号)。 	<ul style="list-style-type: none"> 纠正指定点或起始地址(软元件号和步号)。 			○
	<ul style="list-style-type: none"> 命令的字节长度与规格不一致。 当写入数据时, 写入数据点的设置号与指定号的数值不同。 	<ul style="list-style-type: none"> 检查命令的数据长度且重新进行数据设置。 			○
	<ul style="list-style-type: none"> 即使监视数据未寄存, 也请求监视。 	<ul style="list-style-type: none"> 寄存监视数据。 			○
	<ul style="list-style-type: none"> 当用微机程序来读/写时, 指定参数设置范围末尾后的地址。 	<ul style="list-style-type: none"> 最终地址后的地址不能读/写, 纠正指定地址。 			○
	<ul style="list-style-type: none"> 在扩展文件寄存器的指定块号中, 指定超过相应存储盒大小范围的块号。 	<ul style="list-style-type: none"> 纠正块号。 			○
58 _n	<ul style="list-style-type: none"> 由外部设备指定的命令起始地址(起始软元件号和起始步号)可设置在指定范围外。 当从微机程序或文件寄存器(R)读入或写入到微机程序或文件寄存器(R)中时, 指定超出PLC CPU参数设置范围的数值。 	<ul style="list-style-type: none"> 指定每个处理的允许范围内的适合数值。 			○
	<ul style="list-style-type: none"> 为扩展文件寄存器指定的块号不存在。 	<ul style="list-style-type: none"> 纠正块号。 			○
	<ul style="list-style-type: none"> 不能指定文件寄存器(R)。 	<ul style="list-style-type: none"> 检查软元件。 			○
	<ul style="list-style-type: none"> 为位软元件的命令指定字软元件。 位软元件的起始号由某一个数值指定, 此数值不是字软元件命令中16的倍数。 	<ul style="list-style-type: none"> 纠正命令或指定的软元件。 			○
59 _n	<ul style="list-style-type: none"> 不能指定扩展文件的寄存器。 	<ul style="list-style-type: none"> 检查软元件。 			○
5B _n	<ul style="list-style-type: none"> PLC CPU 和以太网模块不能通讯。 PLC CPU 不能处理外部设备的请求。 	<ul style="list-style-type: none"> 通过参考结束代码中附加的异常代码, 修理有故障的端口(请参阅11.3.2节)。 			○
60 _n	<ul style="list-style-type: none"> 以太网模块和PLC CPU之间的通讯超过了CPU监视定时器的数值。 	<ul style="list-style-type: none"> 提高CPU监视定时器的数值。 			○
63 _n	<ul style="list-style-type: none"> 在固定缓冲器的通讯期间, 目标以太网模块端口的远程口令状态为锁定状态。 	<ul style="list-style-type: none"> 在使用MC协议解开远程口令后, 执行固定缓冲存储器的通讯。 不要如远程口令核对的目标一样来设置固定缓冲器通讯的端口。 	○	○	
A0 _n 至 FFFF _n	<ul style="list-style-type: none"> 每个结束代码的出错内容及出错处理与存储在缓冲存储器中的、从A0_n到FFFF_n的出错代码相同。参考11.3.3节所示适用出错代码的说明来采取必要的措施。 				

11.3.2 使用 A 兼容的 1E 帧进行通讯期间返回的异常代码

本节说明当使用 MC 协议和兼容 1E 帧通讯时，添加入响应中的异常代码(出错代码)。(仅当结束代码为“5B”时，添加异常代码。)

有关重新添加入响应中的结束代码(出错代码)的详细说明，请参阅 11.3.1 节。

有关存储在以太网模块缓冲存储器中的结束代码(出错代码)的详细说明，请参阅 11.3.3 节。



出错代码 (十六进制)	出错	出错说明	纠正措施
10 _h	PC 号出错 (PLC 号出错)	规定 PC 号的站不存在。 (1) 由命令指定的 PC 号既不是本地站的“FF”，也不是由 MELSECNET 链接参数指定的任何站号。	(1) 将 PC 号更改为本地站的“FF”或更改为用链接参数设置的站号，且重新通讯。
11 _h	模式出错	以太网模块和 PLC CPU 之间的通讯差 (1) 在以太网模块成功的从外部设备接收请求后，以太网模块和 PLC CPU 之间仍然由于同样的原因而不能通讯(噪音等)。	(1) 重新通讯。如果出错再次发生，检查噪音等，且更换以太网模块，然后重新通讯。
12 _h	智能功能模块的指定出错	智能功能模块的出错 (1) 由智能功能模块号指定的位置处，不存在带有可通讯缓冲存储器的智能功能模块。(举个例子，相应处是输入/输出模块或空槽。)	(1) 更改在控制步骤中指定的数据内容或更改智能功能模块的安装位置，然后重新通讯。
18 _h	远程出错	远程运行/停止无法通过 其它模块(其它以太网模块等)已执行远程停止/暂停命令。	(1) 检查是否其它模块已执行远程停止/暂停命令。如果命令之一已执行，则删除后重新通讯。
1F _h	软元件出错	软元件规格无效	(1) 核查特殊软元件。 (2) 不要访问任何不存在的软元件。
20 _h	链接出错	请求目标的 CPU 模块未与数据链接系统连接。	检查如 PC 号一般设置站号的 PLC CPU 是否未连接。消除引起关闭的原因后重新连接。
21 _h	智能功能模块的总线出错	不能访问智能功能模块的存储器。 (1) 智能功能模块的控制总线发生故障。 (2) 智能功能模块发生故障。	以下硬件之一发生故障：PLC CPU、基板、智能功能模块或以太网模块。请与您最近的经销商联系。

11.3.3 存储在缓冲存储器中的出错代码

当以太网和外部设备之间的每个数据通讯处理中发生出错时，出错代码(异常代码)存储在以太网模块的缓冲存储器中。本节说明出错类型的内容及出错处理步骤。出错代码列表中的“存储目标”一栏表示存储相应出错代码的缓冲存储器。在“存储目标”一栏中所列的名称(解释时用)与下列表中所示的缓冲存储器出错代码的存储区域相对应。(没有写出存储目标的出错代码会返回亚外部设备。)

解释时所用的名称	缓冲存储器	缓冲存储器的地址
初始化	初始化异常代码区	69 _h (105)
开放	开放异常代码区	7C _h (124)...
固定发送	固定缓冲存储器发送的异常代码区	7D _h (125)...
连接	连接结束代码/出错日志区	7E _h (126)...
出错代码	出错代码结束代码区	E5 _h (229)...
数据链接	((S1)+1 用于数据链接指令)	-
HTTP 记录	HTTP 响应代码区	5108 _h (20744)
电子邮件记录	电子邮件记录区	5879 _h (22649)...
专用指令	(专用指令的完成状态区域)	-

出错代码 (异常代码)	出错说明	出错处理	存储目标									
			初始化	开放	固定发送	连接	出错代码	数据链接	HTTP 记录	电子邮件记录	专用指令	
			69 _h	7C _h	7D _h	7E _h	E5 _h	-	5108 _h	5879 _h	-	
02 _h	每个出错代码的出错内容和出错处理与返回到外部设备的出错代码(02 _h 到0063 _h)相同。 检查 11.3.1 节中所示的可应用代码的解释，且采取相应的措施。	读和处理出错代码和结束代码区。					○	○				
0050 _h						○	○					
0051 _h							○	○				
0052 _h						○	○	○				
0054 _h							○	○				
0055 _h							○	○				
0056 _h							○	○				
0057 _h							○	○				
0058 _h							○	○				
0059 _h							○	○				
005B _h								○	○			
0060 _h						○	○					
0063 _h				○								
00A0 _h	不能指定用于可应用连接的请求。	<ul style="list-style-type: none"> • 核查请求的内容。 • 纠正开放设置。(请参阅 5.5 节) 				○						
00A1 _h	不能分析请求的内容，因为文本区的长度或请求数据的长度太短。	<ul style="list-style-type: none"> • 核查及纠正文本区的长度或 Qn 标题的请求数据长度，且再次发送给以太网模块。 				○						
00A2 _h	不能处理请求。	<ul style="list-style-type: none"> • 纠正请求的内容和命令。 					○					
3E8 _h 至4FFF _h	(由 PLC CPU 检测出的出错)	<ul style="list-style-type: none"> • 参考 CPU 用户手册中的故障排除来处理(硬件设计、维护和检查)。 				○	○					

出错代码 (异常代码)	出错说明	出错处理	存储目标										
			初始化 69 _h	开放 7C _h	固定发送 7D _h	连接 7E _h	出错代码 E5 _h	数据链接 -	HTTP 记录 5108 _h	电子邮件 记录 5879 _h	专用指令 -		
7000 _h 至 7FFF _h	(串行通信模块检测到的错误等)	• 参阅串行通信模块用户手册等，然后采取更正措施。					○						
B000 _h 至 BFFF _h	(CC-Link 模块检测到的错误)	• 参阅 CC-Link 系统主站/本地站模块用户手册等，然后采取更正措施。					○						
C001 _h	• 初始化处理时，以大网模块的 IP 地址设置值不正确。 • 当使用路由器中断功能时，子网掩码域的设置值不正确。	• 纠正 IP 地址。 • 将级别设置为 A/B/C。 • 纠正子网掩码。	○				○						
C002 _h	初始化设置时，某些定时器的设置值在允许范围之外。	• 在初始化处理中，核查及纠正定时器必要的数值。	○				○						
C003 _h	初始化设置时，自动开放 UDP 端口号的设置值在允许范围之外。	• 纠正自动开放 UDP 端口编号。	○				○						
C004 _h	子网掩码域的设置值不正确。	• 纠正子网掩码且再次执行初始化处理。	○				○						
C005 _h	• 用于路由器中断功能的故障 IP 地址设置值不正确。 • 故障路由器 IP 地址的网络 ID (子网掩码后的网络 ID) 与本地站以大网模块 IP 地址的网络 ID 不同。	• 纠正故障路由器 IP 地址且再次执行初始化处理。 • 将网络 ID 与本地站以太网模块 IP 地址的网络 ID 设置一致。	○				○						
C006 _h	路由器中断功能的子网地址设置值不正确。	• 纠正子网地址且再次执行初始化处理。	○				○						
C007 _h	• 用于路由器中断功能的路由器 IP 地址设置值不正确。 • 路由器 IP 地址的网络 ID (子网掩码后的网络 ID) 与本地站以太网模块 IP 地址的网络 ID 不同。	• 纠正路由器 IP 地址且再次执行初始化处理。 • 将网络 ID 与本地站以太网模块 IP 地址的网络 ID 设置一致。	○				○						
C010 _h	在开放处理时，以太网模块端口号的设置值不正确。	• 纠正端口号。	○	○			○						
C011 _h	在开放处理时，外部设备端口号的设置值不正确。	• 纠正端口号。		○	○		○						
C012 _h	接线中所使用的端口号设置已由 TCP/IP 开放。	• 核查及纠正以太网模块和外部设备的端口号。		○			○						
C013 _h	接线中所使用的端口号已在 UDP/IP 开放处理中设置为开放。	• 核查及纠正以太网模块的端口号。		○			○						
C014 _h	以太网模块的初始化处理和开放处理未完成。	• 执行初始化处理和开放处理。			○		○						

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _h	开放 7C _h	固定发送 7D _h	连接 7E _h	出错代码 E5 _h	数据链接 -	HTTP 记录 5108 _h	电子邮件 记录 5879 _h	专用指令 -
C015 _h	在开放处理时，外部设备 IP 地址的设置值不正确。	<ul style="list-style-type: none"> 纠正 IP 地址。 将级别设置为 A/B/C。 		○	○		○				
C016 _h	指定用于成对开放的接线(或下一个接线)中的成对开放处理已完成。	<ul style="list-style-type: none"> 核查用于成对开放的目标连接的开放处理是否未执行。 核查成对开放的组合。 		○			○				
C017 _h	在 TCP 连接的开放处理中不能开放。	<ul style="list-style-type: none"> 检查外部设备的操作。 检查外部设备的开放处理。 纠正通讯参数中的开放设置。 核查以太网模块的端口编号、IP 地址/端口编号和外部设备的开放系统。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 		○			○				
C018 _h	外部设备 IP 地址的设置值不正确。 * 当使用 TCP 时，不能将 FFFFFFF _h 设置为 IP 地址。	<ul style="list-style-type: none"> 纠正 IP 地址。 		○			○				
C020 _h	数据的长度超过了允许的范围。	<ul style="list-style-type: none"> 纠正数据长度。 如果传送的数据大于允许值，则分割后再传送。 			○		○				
C021 _h	在使用固定缓冲存储器传送后，接收到异常结束指令。	<ul style="list-style-type: none"> 从开放的结束指令/出错日志区中读入响应的结束指令，且根据需要处理。 			○		○				
C022 _h	<ul style="list-style-type: none"> 在响应监视定时器的数值范围内不能接收到响应。 等待响应时关闭可应用连接。 	<ul style="list-style-type: none"> 检查外部设备的操作。 核查及纠正响应监视定时器的数值。 检查可应用连接的状态。 			○		○				
C023 _h	<ul style="list-style-type: none"> 可应用连接的开放处理未完成。 关闭可应用连接。 	<ul style="list-style-type: none"> 执行可应用连接的开放处理。 			○		○				
C030 _h	发生发送出错。	<ul style="list-style-type: none"> 检查收发器和外部设备的操作。 * 使用允许进行 SQE 测试的收发器。 经过任意选择时间后发送，因为该线上信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 进行自诊断测试，查看以太网模块是否出故障。 		○	○		○				
C031 _h	发生发送出错。										

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化	开放	固定发送	连接	出错代码	数据链接	HTTP 记录	电子邮件 记录	专用指令
			69 _H	7C _H	7D _H	7E _H	E5 _H	-	5108 _H	5879 _H	-
C032 _H	在 TCP/IP 通讯中发生 TCP 的 ULP 超时出错。 (外部设备未返回 ACK)	<ul style="list-style-type: none"> 检查外部设备的操作。 纠正 TCP 的 ULP 的超时值及再次执行初始化处理。 经过任意选择时间后发送，因为该线上信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 		○	○		○				
C033 _H	外部设备侧设置的 IP 地址不存在。	<ul style="list-style-type: none"> 核查及纠正外部设备的 IP 地址及以太网地址。 如果外部设备具有 ARP 功能，则设置默认值。如果不具有，则设置外部设备的以太网地址。 检查外部设备的操作。 经过任意选择时间后发送，因为该线上的信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 		○	○		○				
C035 _H	外部设备的存在不能在响应监视定时器的数值范围内得到确认。	<ul style="list-style-type: none"> 检查外部设备的操作。 核查及纠正目前确认的每个设置值。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 		○	○		○				
C036 _H	不能执行发送处理，因为电缆未连接或已断开。	<ul style="list-style-type: none"> 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 进行环路回送测试，查看线路是否有故障。 进行自诊断测试，查看以太网模块是否故障。 					○				
C040 _H	<ul style="list-style-type: none"> 在响应监视定时器的数值范围内，没有接收到所有的数据。 没有接收到达到数据长度的足够数据。 在响应监视定时器的数值范围内，不能接收 TCP/IP 级别的，已分割信息的解除部分。 	<ul style="list-style-type: none"> 核查及纠正通讯数据的数据长度。 核查及纠正初始化处理中的每个设置值因为在线时信息包有可能已满。 再次从外部设备发送相同的数据。 		○	○		○				
C041 _H	当使用 TCP 时，接收数据的校验和不正确。	<ul style="list-style-type: none"> 检查外部设备侧的校验和，且发送正确的数值。 调查线路条件。(噪音、线路和电源线之间的距离、每个设备的传导性等) 			○		○				
C042 _H	当使用 UDP 时，接收数据的校验和不正确。				○		○				
C043 _H	接收到的 IP 包的标题求和不正确。				○		○				

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 —	HTTP 记录 5108 _H	电子邮件 记录 5879 _H	专用指令 —
C044 _H 至 C048 _H	接收到 ICMP 的出错信息包。	<ul style="list-style-type: none"> 检查外部设备的操作。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 			○		○				
C049 _H	接收到 ICMP 的出错信息包。	<ul style="list-style-type: none"> 检查外部设备的操作。 经过任意选择时间后发送，因为该线上的信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 			○		○				
C04A _H	接收到 ICMP 的出错信息包。 (在外部设备发生 IP 组合超时出错。)	<ul style="list-style-type: none"> 纠正外部设备现存定时器超时的 IP 组合定时器数值。 			○		○				
C04B _H	发生 IP 组合超时出错。 (不能接收分割数据的剩余部分发生超时。)	<ul style="list-style-type: none"> 检查外部设备的操作。 经过任意选择时间向后发送，因为在线时信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子的接线是否有故障。 纠正 IP 组合定时器的数值且再次执行初始化处理。 			○		○				
C04C _H	无法发送，因为内部缓冲存储器如 IP 标题缓冲存储器中已无空间。	<ul style="list-style-type: none"> 再次发送相同的数据且检查已收到的响应。 			○		○				
C04D _H	<ul style="list-style-type: none"> 在以太网模块通过自动开放 UDP 端口通讯或无序固定缓冲存储器通讯接收的信息中，应用领域中指定的数据长度不正确。 不能储存所有的接收数据。 	<ul style="list-style-type: none"> 核查数据长度。 核查文本大小，使文本数据小于接收缓冲存储器的大小。 				○	○				
C050 _H	当在以太网模块中进行 ASCII 代码通讯的操作设置时，接收不能转化为二进制代码的 ASCII 代码数据。	<ul style="list-style-type: none"> 在操作设置中选择二进制代码通讯，再启动以太网模块。 纠正从外端发送来的数据且再次发送。 			○		○				
C051 _H 至 C054 _H	读/写点的数目在允许范围之外。	<ul style="list-style-type: none"> 纠正读/写点的数目且再次发送至以太网模块。 			○		○				
C055 _H	文件数据读/写点的数目在允许范围之外。	<ul style="list-style-type: none"> 纠正读/写点(或字节)的数目，且再次发送至以太网模块。 			○		○				
C056 _H	<ul style="list-style-type: none"> 读/写请求超过了最大地址。 地址为 0。 	<ul style="list-style-type: none"> 纠正起始地址或写/读点的数目，且再次发送至以太网模块。 (一定不能超过最大地址。) 			○		○				

出错代码 (异常代码)	出错说明	出错处理	存储目标										
			初始化 69 _h	开放 7C _h	固定发送 7D _h	连接 7E _h	出错代码 E5 _h	数据链接 -	HTTP 记录 5108 _h	电子邮件 记录 5879 _h	专用指令 -		
C057 _h	请求数据的长度与字符区域(部分文本)的数据计数不匹配。	• 核查及纠正文本区域的内容或 Qn 标题的请求数据长度, 且再次发送至以太网模块。			○		○						
C058 _h	在经过 ASCII 二进制转换后, 请求数据的长度与字符区域(部分文本)的数据计数不相符。	• 核查及纠正文本区域的内容或 on 标题的请求数据长度, 且再次发送至以太网模块。			○		○						
C059 _h	命令和子命令的指定不正确。	• 核查请求内容。			○		○						
C05A _h	以太网模块不能对指定软元件进行读出和写入。	• 检查软元件的读/写。			○		○						
C05B _h					○		○						
C05C _h	请求内容不正确。 (以位为单元请求读/写至软元件。)	• 纠正请求内容且再次发送至以太网模块。 (纠正子命令等。)			○		○						
C05D _h	不执行监视注册。	• 在执行监视注册后进行监视。			○		○						
C05E _h	以太网模块和 PLC CPU 之间的通讯时间超过了 CPU 监视定时器的时间。	• 增加 CPU 监视定时器的数值。 • 检查 PLC CPU 是否正常的操作。			○		○						
C05F _h	目标 PLC 上不能执行请求。	• 纠正网络号和 PC 号。 • 纠正读/写请求的内容。			○		○						
C060 _h	请求内容不正确。 (对位软元件等指定了不正确的数据。)	• 纠正请求内容且再次发送至以太网模块。 (纠正数据等。)			○		○						
C061 _h	请求数据的长度与字符区域(部分文本)中的数据数目不相符。	• 核查及纠正文本区域的内容或 Qn 标题的请求数据长度, 且再次发送至以太网模块。			○		○						
C062 _h	禁止在线更正时, 通过 MC 协议对远程 I/O 站执行(QnA 兼容 3E 帧或 4E 帧)写入操作。	• 当通过 MC 协议(QnA 兼容 3E 帧或 4E 帧)对远程 I/O 站执行写入操作时, “允许”操作设置的在线更正设置。					○						
C070 _h	不能为目标站指定软元件存储器的范围。	• 不指定读/写的范围。 * 只能为安装以太网模块的站和通过 MELSECNET/H, MELSECNET/10 开放的 Q/QnACPU 指定软元件的存储器范围。			○		○						
C071 _h	除了 Q/QnACPU 以外, 其它站也有过多的读/写软元件点。	• 纠正读/写的软元件点数目, 且再次发送至以太网模块。			○		○						
C072 _h	请求内容不正确。 (以位为单元请求调写至软元件。)	• 检查是否能向目标 PLC CPU 请求内容。 • 纠正请求内容且再次发送至以太网模块。 (纠正子命令等)			○		○						
C073 _h	目标 PLC CPU 的以太网模块不支持请求。 (除了 Q/QnACPU 等, 其它站也指定了两个地址)	• 核查请求内容。			○		○						

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _h	开放 7C _h	固定发送 7D _h	连接 7E _h	出错代码 E5 _h	数据链接 -	HTTP 记录 5108 _h	电子邮件 记录 5879 _h	专用指令 -
C074 _h	目标 PLC 不执行请求。	<ul style="list-style-type: none"> 纠正网络号和 PC 号。 纠正读/写请求的内容。 			○		○				
C080 _h	在 MELSECNET/H, MELSECNET/10 继电器通讯或数据链接指令通讯期间, 无法获得外部设备的 IP 地址。	<ul style="list-style-type: none"> 设置以太网模块的以太网 MNET/10 路由器信息。 更改 MELSECNET/H, MELSECNET/10 的参数转换方法。 					○	○			
C081 _h	已完成以太网模块的结束处理, 且无法检查数据链接指令通讯。	<ul style="list-style-type: none"> 在所有的通讯结束后再执行以太网模块的结束处理。 					○	○			
C082 _h	在以下通讯中, 通讯处理异常完成。 <ul style="list-style-type: none"> 与 GX Developer (UDP/IP) 的通讯 MELSECNET/H, MELSECNET/10 中继。 	<ul style="list-style-type: none"> 检查异常操作的中继站/外部站。(如果通讯继续进行, 则没有必要对出错采取措施。) 检查本站站和外部站之间的电缆连接是否有故障。 					○				
C083 _h	在数据链接指令通讯中, 通讯处理异常完成。	<ul style="list-style-type: none"> 检查本站站/中继站及外部站是否异常操作。 检查本站站和外部站之间的电缆开放是否有故障。 使现存确认目标的内部定时器的数值变长。 					○	○			
C084 _h	在数据链接指令通讯中, 通讯处理异常完成。	<ul style="list-style-type: none"> 检查本站站/中继站及外部站是否异常操作。 检查本站站和外部站之间的电缆开放是否有故障。 使现存确认目标的内部定时器的数值变长。 					○				
C085 _h	数据链接指令通讯中另一个站指定的本站站通道目前正在使用。	<ul style="list-style-type: none"> 从另一个站重新执行请求。 					○	○			
C086 _h	接收到超过接收信息大小的信息。	<ul style="list-style-type: none"> 从请求源纠正发送信息的大小。 					○				
C087 _h	在 IP 地址中出现为 MELSECNET/H, MELSECNET/10 中继通讯设置的路由器信息出错。	<ul style="list-style-type: none"> 在路由器信息中, 为 MELSECNET/H, MELSECNET/10 中继通讯设置目标软元件的 IP 地址。 					○				○
C0B2 _h	用于 MELSOFT 连接、数据链接指令或通讯请求站的中继站接收缓冲器中, 剩余的空间不足。(接收缓冲期完全错误)	<ul style="list-style-type: none"> 给予请求间隔间充足的时间。 减少请求代码数。 等待请求的响应并在接受响应后发送下一个请求。 重新查看超时值。 					○	○			
C0B3 _h	作不能由 PLC CPU 处理的请求。	<ul style="list-style-type: none"> 核查请求内容。 纠正网络号和 PC 号。 					○	○			
C0B5 _h	指定了不能由 PLC CPU/以太网模块处理的数据。	<ul style="list-style-type: none"> 核查请求内容。 取消目前的请求。 					○	○			
C0B6 _h	通道号在允许范围之外。	<ul style="list-style-type: none"> 在范围 1-8 之间指定通道号。 					○	○			
C0B7 _h	指定目前正在使用的通道号。	<ul style="list-style-type: none"> 更改通道号。 在完成目前的通讯后执行。 					○	○			

出错代码 (异常代码)	出错说明	出错处理	存储目标							
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 -	HTTP 记录 5108 _H	电子邮件 记录 5879 _H
COB8 _H	网络号和 PC 号在允许范围之外。 PLC CPU 的响应有故障。	<ul style="list-style-type: none"> 纠正网络号和 PC 号。 检查 PLC CPU 的操作。 					○			
COB9 _H	可应用连接的开放处理未完成。	<ul style="list-style-type: none"> 执行开放处理。 检查外部设备的操作。 			○		○			
COBA _H	在通过 CLOSE 指令执行关闭处理后，不能确认发送请求。	<ul style="list-style-type: none"> 执行开放处理且产生一个发送请求。 			○		○			
COBB _H	系统错误 • OS 检测到任何错误。	(*1)								
COBC _H	指定的通讯线路关闭。	<ul style="list-style-type: none"> 通讯线路开放。 核查目标连接数。 			○	○	○			
COBD _H	通过对连续请求的确认，而无法发送。	<ul style="list-style-type: none"> 检查是否无须等待响应而连续产生请求。 			○	○	○			
COBE _H	系统错误	(*1)								
COBF _H	• OS 检测到任何错误。									
COCO _H	当接收完成信号未打开时，开放接收完成确认信号。	<ul style="list-style-type: none"> 执行开放处理。 检查外部设备的操作。 			○		○			
COC1 _H	UDP 的传送间隔过短。	<ul style="list-style-type: none"> 检查是否重复发送亲请求。 使发送间隔更长。 					○			
COC2 _H	系统错误	(*1)								
COC3 _H	• OS 检测到任何错误。									
COC4 _H	通讯时已经执行 UINI 指令。									
COC5 _H	<ul style="list-style-type: none"> 当不使用路由器功能时，向外部设备产生发送请求，这些外部设备的级别/网络 ID 与本地站的级别/网络不同。 以太网路由器信息的设置不正确。 	<ul style="list-style-type: none"> 通过设置应该使用的功能来执行初始化处理。 对参数设置正确数据。 更正外部设备的 IP 地址并执行开放处理。 检查网络地址是否正确地址。更改时，再次执行初始化处理。 		○	○		○			
COC6 _H	系统错误 • OS 检测到任何错误。	(*1)								
COC7 _H	以太网模块系统出错。	<ul style="list-style-type: none"> 再次执行初始化处理。 参见 11.1 节要点(3)来执行处理。 	○		○		○			
COCS _H 至 COCA _H	系统错误 • OS 检测到任何错误。	(*1)								
COCB _H	当发送处理未完成时，又作了另一个发送请求。	<ul style="list-style-type: none"> 在完成先前发送后进行下一个发送请求。 			○		○			
COCC _H	系统错误	(*1)								
COCF _H	• OS 检测到任何错误。									
COD0 _H	指定了不正确的数据长度。	<ul style="list-style-type: none"> 检查数据长度的指定值。 					○	○		
COD1 _H	指定了不正确的重新发送次数。	<ul style="list-style-type: none"> 检查重新发送次数的指定。 					○	○		

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 -	HTTP 记录 5108 _H	电子邮件 记录 5879 _H	专用指令 -
COD2 _H	指定了不正确的 WTD。	• 核查 WTD 的指定值。					○	○			
COD3 _H	MELSECNET/H, MELSECNET/10 中的中 继站数超过了允许的计数范围。	• 核查目标的指定值。					○				
COD4 _H		• 核查本站点和目标之间的以太网路由信息。 • 核查系统规格。					○				
COD5 _H	指定了不正确的重试数。	• 核查重试计数。					○	○			
COD6 _H	指定的网络号或站号不正确。	• 核查指定的目标内容。 • 核查目标的指定值。					○	○			
COD7 _H	由 GX Developer 设置的参数值未装 载。	• 用 GX Developer 设置参数值且向 CPU 模块装载这些参数。 • 完成初始化处理后开始与外部设备进行通 讯。					○	○			
COD8 _H	块数超出了范围。	• 纠正块数的指定值。					○				
COD9 _H	指定了不正确的子命令。	• 纠正子命令的指定值。					○				
CODA _H	在通讯时间的检查期内，未接收到对 PING 测试的响应。	• 核查用于 PING 测试的以太网模块的 IP 地 址/主站名称。 • 通过更改用于 PING 测试的以太网模块的 状态来允许通讯。（完成初始化处理后的 状态）									○
CODB _H	用于 PING 测试的以太网模块的 IP 地 址/主站名称出错。	• 核查用于 PING 测试的以太网模块的 IP 地 址/主站名称。									○
CODC _H	系统错误	(*1)									
CODD _H	• OS 检测到任何错误。										
CODE _H	在指定的监视时间内不能接收到数 据。	• 核查 WTD 的指定值。 • 核查通道数的指定值。 • 检查发送源站和中继站的状态。					○	○			
CODF _H	系统错误	(*1)									
	• OS 检测到任何错误。										
COE0 _H 至 COEF _H	PLC CPU 中检测到错误。	• 检查 PLC CPU 和智能功能模块是否正确安 装到基板上。 • 检查 PLC CPU 是否仍然处于复位状态。 • 检查 PLC CPU 中是否没有发生过出错。如 果发现错误，根据 PLC CPU 的错误说明采 取更正措施。 • 替换电源模块，PLC CPU 和/或智能功能 模块。					○				
COF0 _H	在硬件测试中检测出以太网模块 RAM 异常。	• 再次进行硬件测试。如果再次检测出异 常，则以太网模块的硬件有可能有故障。 向最近的分部或经销商咨询出错详细说 明。					○				

出错代码 (异常代码)	出错说明	出错处理	存储目标							
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 -	HTTP 记录 5108 _H	电子邮件 记录 5879 _H
C0F1 _H	在硬件测试中检测出以太网模块 ROM 异常。	<ul style="list-style-type: none"> 再次进行硬件测试。如果再次检测出异常，则以太网模块的硬件有可能有故障。向最近的分部或经销商咨询出错详细说明。 					○			
C0F3 _H	在 CPU 中检测出模块系统出错(异常情况严重)。	<ul style="list-style-type: none"> 纠正本站 CPU 中的出错原因。 					○			
C0F4 _H 至 C0F6 _H	系统错误 <ul style="list-style-type: none"> OS 检测到任何错误。 	(*1)								
C0F7 _H	自抑制测试中发生出错。	<ul style="list-style-type: none"> 经过任意选择时间后发送，因为该线路上信息包有可能已满。 检查连接电缆是否错位。 检查收发器和端子之间的连接是否有故障。 如果不是以上问题，则以太网模块硬件可能有故障。向您最近的经销商咨询出错详细说明。 					○			
C100 _H	系统错误 <ul style="list-style-type: none"> OS 检测到任何错误。 	(*1)								
C101 _H	不能从 DNS 客户中接收响应。	<ul style="list-style-type: none"> 检查 DNS 服务器地址。 检查使用 PING 命令，是否有可能与 DNS 服务器通讯。 检查本站的 IP 地址是否和 DNS 服务器在同一级别，如果级别不同，检查路由器设置。)) 							○	
C102 _H	不能接收来自 SMTP 层的响应。	<ul style="list-style-type: none"> 检查注册在 DNS 中的 SWP 服务器的名称。 删除 SWP 服务器名称，更改 IP 地址的设置，且检查操作。 检查使用 PING 命令，是否有可能与 DNS 服务器通讯。 							○	
C103 _H	DNS 设置不正确。	<ul style="list-style-type: none"> 检查 DNS 邮件地址。 检查 DNS 设置的内容。 								
C104 _H 至 C106 _H C110 _H	系统错误 <ul style="list-style-type: none"> OS 检测到任何错误。 	(*1)								
C111 _H	不能从 DNS 客户中接收响应。	<ul style="list-style-type: none"> 检查电缆、网络集线器等。 检查使用 PING 命令，是否有可能与 DNS 服务器通讯。 							○	
C112 _H	不能接收来自 POP3 层的响应。	<ul style="list-style-type: none"> 检查注册在 DNS 中的 SWP 服务器的名称。 删除 POP3 服务器名称，更改 IP 地址的设置，且检查操作。 检查使用 PING 命令，是否有可能与 POP3 服务器通讯。 							○	

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 -	HTTP 记录 5108 _H	电子邮件 记录 5879 _H	专用指令 -
C113 _H	接收到没有附件的电子邮件。 (当异常读附件时会产生)	<ul style="list-style-type: none"> 在发送侧指定附件。 在发送侧检查程序。 当发送源为邮件服务器时，通过先前的 MSEND 指令发送失败。检查 MSEND 指令的目标等。 检查发送侧是否和以太网模块有相同的电子邮件规格。(编码/解码、文件格式等) 从 SMTP 服务器中接收到目标未知的服务器。 								○	
C114 _H	接收到附件名无效的电子邮件。	<ul style="list-style-type: none"> 在发送侧检查附件的“扩展名”是“bin”或“asc”。 检查附件是压缩的还是加密的。 检查 MSEND 指令等的目标。 从 SMTP 服务器中接收到目标未知的服务器。 								○	
C115 _H 至 C118 _H	系统错误 • OS 检测到任何错误。	(*1)									
C119 _H	未接收到邮件。	<ul style="list-style-type: none"> 检查发送端是否和以太网模块有相同的电子邮件规格。(编码解码、文件格式等) 检查缓冲存储器的邮件信息储存区(地址: 2682_H)，然后读取服务器中的任何接收邮件。 									○
C11A _H	未转换准备接收的电子邮件。	<ul style="list-style-type: none"> 检查附件是压缩的还是加密的。 检查发送侧是否和以太网模块有相同的电子邮件规格。(编码解码、文件格式等) 检查文件是否在发送侧分割。 								○	
C11B _H	发送电子邮件后，从目标邮件服务器中接收到出错邮件。	<ul style="list-style-type: none"> 从 SWP 服务器中接收到目标未知的服务器。(接收邮件储存在邮件缓冲存储器中。) 检查“@”之前的部分在参数设置的邮件地址设置中是否正确。 检查“@”之前的部分是否在目标邮件服务器中注册。 								○	○
C11C _H	没有发现邮件地址	<ul style="list-style-type: none"> 检查参数设置的邮件设置是否正确。 当邮件服务器和以太网模块通过路由器连接，检查路由器是否正确。 测试发送电子邮件到可以接收但不会出错的地址。当正常完成测试时，重新检查“@”后的域名是否正确。 								○	○
C11D _H	附件大小超过了允许范围。	<ul style="list-style-type: none"> 检查附件是否小于 6k 字。 检查发送侧是否分割文件。 								○	

出错代码 (异常代码)	出错说明	出错处理	存储目标									
			初始化 69#	开放 7C#	固定发送 7D#	连接 7E#	出错代码 E5#	数据链接 -	HTTP 记录 5108#	电子邮件 记录 5879#	专用指令 -	
C120#	不能连接 SMTP 服务器。	<ul style="list-style-type: none"> 检查 SMTP 服务器的端口编号是否=25。 检查是否有可能使用 PING 命令来与 SMTP 服务器通讯。 									○	
C121#	不能与 SMTP 服务器通讯。 (出错响应)	<ul style="list-style-type: none"> 检查 SMTP 服务器是否繁忙。 									○	
C122#	不能与 SMTP 服务器通讯。 (异常中断)	<ul style="list-style-type: none"> 检查 SMTP 服务器是否繁忙。 									○	
C123#	不能与 SMTP 服务器通讯。 (复位响应)	<ul style="list-style-type: none"> 检查 SMTP 服务器是否繁忙。 									○	
C124#	从 SMTP 服务器来的响应超时。	<ul style="list-style-type: none"> 检查 SMTP 服务器是否有故障。 检查网络上是否负载过多。 									○	
C125#	从 SMTP 服务器上强制关闭。	<ul style="list-style-type: none"> 检查 SMTP 服务器是否有故障。 									○	
C126#	不能断开 SMTP 服务器。	<ul style="list-style-type: none"> 检查 SMTP 服务器是否有故障。 检查网络上是否负载过多。 									○	
C127#	关闭 SMTP 服务器时，给出出错的响应。	<ul style="list-style-type: none"> 检查 SMTP 服务器是否有故障。 									○	
C130#	由于服务器无效，通讯通道关闭。	<ul style="list-style-type: none"> 检查 SMTP 服务器的状态。 									○	
C131#	SMTP 服务器在执行处理时，接收到出错响应。	<ul style="list-style-type: none"> 检查是否指定了未在服务器中注册的用户名。 经过任意的设置时间后，再次发送。 									○	
C132#	SMTP 服务器在执行处理时，接收到出错响应。 (本地出错)	<ul style="list-style-type: none"> 检查 SMTP 服务器的状态。 									○	
C133#	SMTP 服务器在执行处理时，接收到出错响应。 (存储区域不足)	<ul style="list-style-type: none"> 检查 SMTP 服务器的状态。 									○	
C134# 至 C137#	系统错误 • OS 检测到任何错误。	(*1)										
C138#	SMTP 服务器在执行处理时，接收到出错响应。 (未找到邮箱)	<ul style="list-style-type: none"> 检查以太网模块的邮件地址是否设置正确。 									○	
C139#	系统错误 • OS 检测到任何错误。	(*1)										
C13A#	SMTP 服务器在执行处理时，接收到出错响应。 (超过了存储区域的分配)	<ul style="list-style-type: none"> 检查 SMTP 服务器的状态。 									○	
C13B#	SMTP 服务器在执行处理时，接收到出错响应。 (邮箱名无效)	<ul style="list-style-type: none"> 检查以太网模块的邮件地址是否设置正确。 									○	
C13C#	系统错误 • OS 检测到任何错误。	(*1)										

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化	开放	固定发送	连接	出错代码	数据链接	HTTP 记录	电子邮件记录	专用指令
			69 _H	7C _H	7D _H	7E _H	E5 _H	-	5108 _H	5879 _H	-
C140 _H	不能开放 POP3 服务器。	<ul style="list-style-type: none"> 检查 POP3 服务器的端口号是否为 110。(对于以太网模块, 固定为 110。) 检查是否可以使用 PING 命令来与 POP3 服务器通讯。 								○	
C141 _H	不能与 POP3 服务器通讯 (出错响应)	<ul style="list-style-type: none"> 检查 POP3 服务器是否忙。 								○	
C142 _H	不能与 POP3 服务器通讯 (异常中断)	<ul style="list-style-type: none"> 检查 POP3 服务器是否繁忙。 								○	
C143 _H	不能与 POP3 服务器通讯 (复位响应)	<ul style="list-style-type: none"> 检查 POP3 服务器是否繁忙。 								○	
C144 _H	不能从 POP3 服务器中接收响应。	<ul style="list-style-type: none"> 检查 POP3 服务器是否有故障。 检查网络上是否负载过多。 								○	
C145 _H	从 POP3 服务器上强制关闭。	<ul style="list-style-type: none"> 检查 POP3 服务器是否有故障。 								○	
C146 _H	不能断开 POP3 服务器。	<ul style="list-style-type: none"> 检查 POP3 服务器是否有故障。 检查网络上是否负载过多。 								○	
C147 _H	断开 POP3 服务器时, 给出出错响应。	<ul style="list-style-type: none"> 检查 POP3 服务器是否有故障。 								○	
C150 _H	POP3 服务器验证出错。	<ul style="list-style-type: none"> 检查 POP3 服务器的状态。 								○	
C151 _H	以太网模块的邮件地址(电子邮件地址设置参数)与服务器侧邮箱的帐户名不同。	<ul style="list-style-type: none"> 检查服务器侧邮箱的帐户名及纠正为以太网模块设置的邮箱帐户。 								○	
C152 _H	以太网模块的密码(电子邮件设置参数)与服务器侧的密码不同。	<ul style="list-style-type: none"> 检查服务器侧的密码及纠正为以太网模块设置的密码。 								○	
C153 _H	当给出接收邮件清单时, 发生出错。(无法获得到达 POP3 服务器的邮件清单。)	<ul style="list-style-type: none"> 将服务器查询时间重新设置为默认值, 且复位 PLC CPU 的本地站。 								○	
C154 _H	接收邮件时发生出错。(不能从 POP3 服务器中读电子邮件。)	<ul style="list-style-type: none"> 检查邮件是压缩的还是加密的。 检查发送侧是否和以太网模块有相同的电子邮件规格。(编码懈码、文件格式等) 								○	
C160 _H	超时后从 DNS 服务器中接收到响应。	<ul style="list-style-type: none"> 检查网络上是否负载过多。 								○	
C161 _H	不能从 DNS 服务器中接收到邮件。	<ul style="list-style-type: none"> 检查 DNS 服务器的状态。 								○	

出错代码 (异常代码)	出错说明	出错处理	存储目标								
			初始化 69 _H	开放 7C _H	固定发送 7D _H	连接 7E _H	出错代码 E5 _H	数据链接 -	HTTP 记录 5108 _H	电子邮件 记录 5879 _H	专用指令 -
C162 _H C163 _H C171 _H 至 C17F _H	从 DNS 服务器中返回错误	<ul style="list-style-type: none"> 检查 DNS 服务器的 IP 地址设置是否正确。 检查邮件服务器名设置(SMTP 服务器名, POP 服务器名)是否正确。 与网络管理员或相关人员检查 DNS 设置中的服务器 DNS 功能是否正在执行。 								○	
C1A0 _H	作了非法请求	<ul style="list-style-type: none"> 再次执行。 如果又发生同样的出错, 则以太网模块的硬件有可能有故障。 向您最近的经销上咨询出错详细说明。 									○
C1A2 _H	不能接收到请求的响应。	<ul style="list-style-type: none"> 核查及纠正响应等待时间。 									○
C1A4 _H	请求或子请求不正确。	<ul style="list-style-type: none"> 纠正请求或子请求。 									○
C1A5 _H	目标站或清除目标的指定不正确。	<ul style="list-style-type: none"> 纠正目标站或清除目标的指定值。 									○
C1A6 _H	指定了不正确的开放号。	<ul style="list-style-type: none"> 在 1-16 范围内指定连接号。 									○
C1A7 _H	指定了不正确的网络号。	<ul style="list-style-type: none"> 纠正网络号的指定值。 									○
C1A8 _H	指定了不正确的站号。	<ul style="list-style-type: none"> 纠正站号的指定值。 									○
C1A9 _H	指定了不正确的软元件号。	<ul style="list-style-type: none"> 纠正软元件号的指定值。 									○
1AA _H	指定了不正确的软元件名。	<ul style="list-style-type: none"> 纠正软元件名的指定值。 									○
C1AC _H	指定了不正确的不良计数。	<ul style="list-style-type: none"> 纠正不良计数的指定值。 									○
C1AD _H	指定了不正确的数据长度。	<ul style="list-style-type: none"> 纠正数据长度的指定值。 									○
C1AE _H	指定了不正确的邮件发送/接收数据长度及标题长度。	<ul style="list-style-type: none"> 纠正发送/接收数据长度及标题长度的指定值。 发送/接收的数据长度应等同与或大于标题的长度。 									○
C1AF _H	指定了不正确的端口编号。	<ul style="list-style-type: none"> 纠正端口号的指定值。 									○
C1B0 _H	指定连接的开放处理已完成。	<ul style="list-style-type: none"> 在执行关闭处理之后执行开放处理。 									○
C1B1 _H	指定连接的开放处理未完成。	<ul style="list-style-type: none"> 执行关闭处理。 									○
C1B2 _H	在指定连接中执行 OPEN/CLOSE 指令。	<ul style="list-style-type: none"> 在完成 OPEN/CLOSE 指令后执行。 									○
C1B3 _H	在指定通道中执行另一个发送/接收指令。	<ul style="list-style-type: none"> 更改通道号, 在完成发送/接收指令后执行。 									○
C1B4 _H	指定了不正确的到达时间。	<ul style="list-style-type: none"> 在适当的范围内设置 WDT。 									○
C1B5 _H	在指定的监视时间内不能接收到数据。	<ul style="list-style-type: none"> 核查 WDT 的指定值。 									○
C1B6 _H	指定了不正确的邮件目标号。	<ul style="list-style-type: none"> 核查邮件指定号的指定值。 核查发送邮件地址的设置参数。 									○

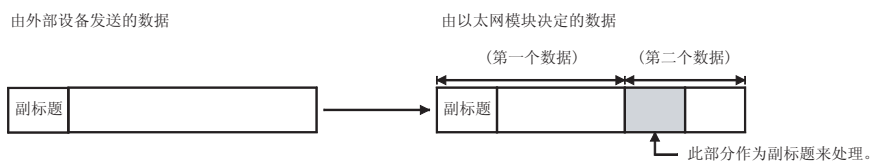
出错代码 (异常代码)	出错说明	出错处理	存储目标										
			初始化 69 _h	开放 7C _h	固定发送 7D _h	连接 7E _h	出错代码 E5 _h	数据链接 -	HTTP 记录 5108 _h	电子邮件 记录 5879 _h	专用指令 -		
C1B7 _h	当无电子邮件储存在邮件缓冲存储数据区内时执行读操作。	• 当邮件信息中邮件接收标志为“是”时，执行 MRECV 指令。											○
C1B8 _h	对没有接收数据的通道执行 RECV 指令。	• 重新检查 RECV 指令的执行情况。 • 重新检查通道号。											○
C1B9 _h	对指定连接不可以执行 OPEN 指令。	• 重新检查连接号。											○
C1BA _h	在初始化未完成情况下执行专用指令。	• 在初始化处理完成后才能执行专用指令。											○
1BB _h	目标站 CPU 类型错误。	• 重新检查对目标站 CPU 类型指定的值。											○
C200 _h	远程密码出错。	• 核查远程密码，然后再次执行远程密码的解锁/锁定处理。						○					
C201 _h	用于通讯的端口的远程密码状态为锁定状态。	• 解锁远程密码，然后执行通讯。						○					○
C202 _h	当访问另一个站时，无法解锁远程密码。	• 当访问另一个站时，不要在中继站或要访问的站中设置远程密码，如果这样做，他们将不受远程密码检查的控制。						○					
C203 _h	系统错误 • OS 检测到任何错误。	(*1)											
C204 _h	设备与请求远程密码解锁的设备不同。	• 从己请求远程密码解锁处理的对方设备来请求远程密码锁定处理。						○					
C205 _h	当访问另一个站时，无法解锁远程密码。	• 当访问另一个站时，不要在中继站或要访问的站中设置远程密码，如果这样做，他们将不受远程密码检查的控制。						○					
C206 _h	系统错误 • OS 检测到任何错误。	(*1)											
C207 _h	文件名字符过多。	• 把文件名字符减少到 255 个字符的范围。						○					
C300 _h	在响应 WDT 数值内，无法接收到响应。	• 检查外部设备的操作。 • 核查及纠正响应 WDT 的数值。						○					
F000 _h 至 FFFF _h	(由 MELSECNET/H, MELSECNET/10 网络系统检测出错)	• 有关 MELSECNET/H、MELSECNET/10 网络系统的详细说明及出错处理，请参阅手册。						○					

- *1 按照以下步骤进行更正处理。
- 1) 检查以太网模块，电源模块和 CPU 模块是否正确安装在基板上。
 - 2) 检查以太网模块的操作环境是否在 CPU 模块的一般说明范围内。
 - 3) 检查电源功率是否足够。
 - 4) 根据相应模块的手册检查以太网模块、电源模块和 CPU 模块的硬件是否正常。如果任何模块发生故障，要求您当地的三菱服务中心或代表处进行修理。
 - 5) 如果以上方法无法解决问题，请与您当地的三菱服务中心或代表处联系。说明发生错误时的操作/通讯环境和以太网模块的缓冲存储器中出错日志所存的信息。

备注

根据本地站和外部设备缓冲存储器的限制条件，数据会因通讯而分割。
 分别接收到的数据通过使用固定缓冲存储器或随机访问缓冲存储器等进行通讯的以太网模块来还原(再组合)。接收数据根据在通讯数据中的数据长度来还原(再组合)。当通讯数据的设置值和实际数据长度不同时，发生出错。

- (1) 当使用固定缓冲器(有步骤)和随机缓冲器进行通讯时
- (a) 在副标题小于接收的正文大小时立即指定数据长度
- 1) 与副标题后指定数据长度相对应的紧接在正文后数据被认为是下一个信息。
 - 2) 因为每个信息的标题都有副标题，所以以太网根据副标题中的代码执行操作。
 - 3) 如果以太网模块无法识别副标题包含的代码，以太网模块会向外部设备发送异常完成的响应通知。



本例的响应，作为副标题的代码处理最高位会作为设置为 1 的代码而返回。
 例如，如果命令的副标题部分是 65H，则响应的副标题是 E5H。

- (b) 如果副标题后指定的数据长度超出接收的正文大小
- 1) 以太网模块等待接收剩余的丢失数据。
 - 2) 如果剩余数据要在响应监视定时器分配的时间内接收，以太网模块根据副标题中的代码执行操作。
 - 3) 如果剩余数据无法在响应监视定时器分配的时间内接收，以太网模块执行以下操作。
 - 向外部设备发送 ABORT (RST) 指令并切断电源。
 - 通过开放错误检测信号 (X18=0N) 向 PLC CPU 通知发生开放错误。
 - 在开放错误代码存储区存储出错代码。(出错代码不存储在出错日志存储区。)

要点

对外部设备发送到以太网模块的信息，其应用数据区中说明的“数据长度”指定正文区的实际数据大小。
--

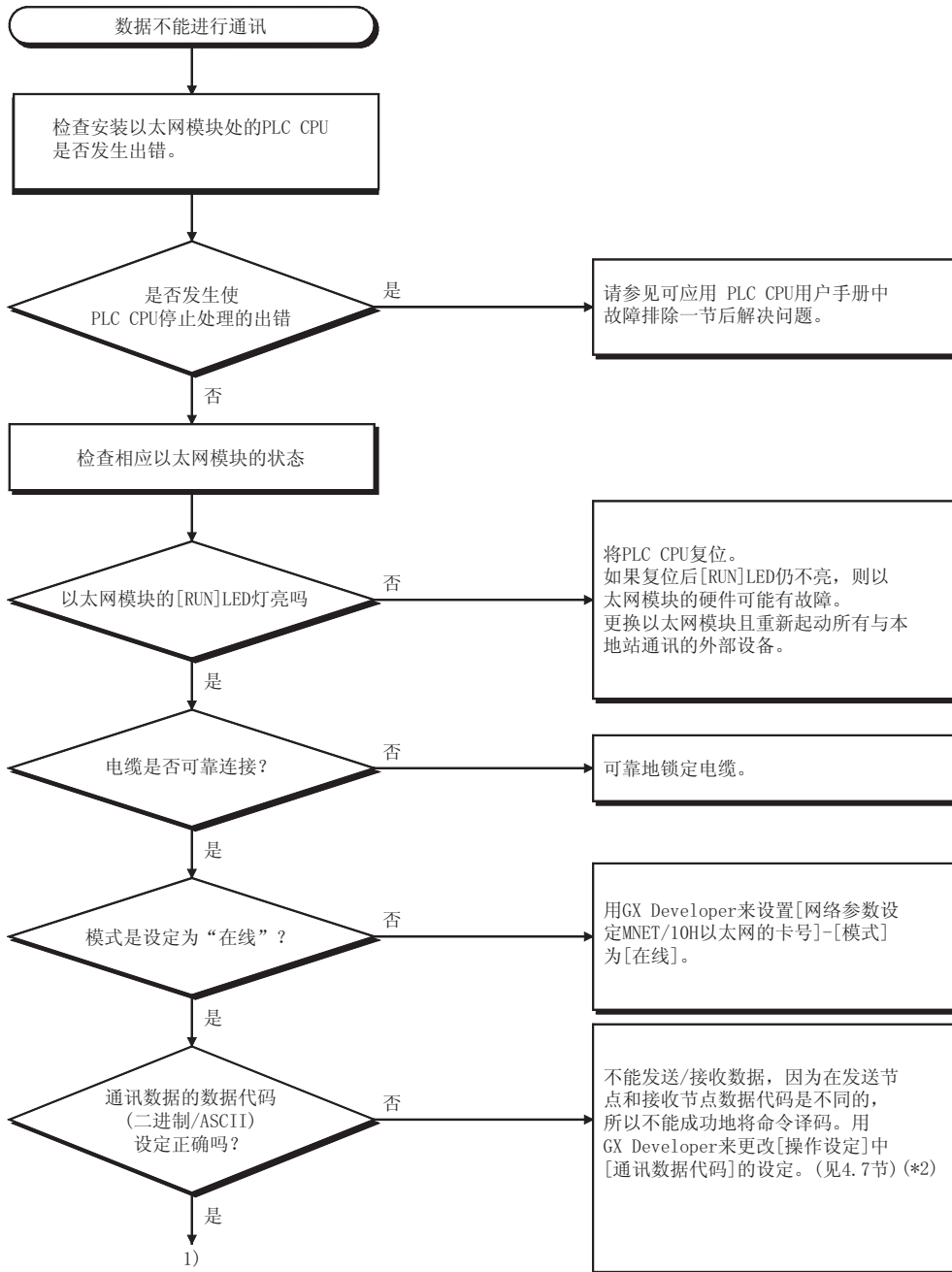
以太网模块不会发送大小不同于指定数据长度的正文到外部设备。

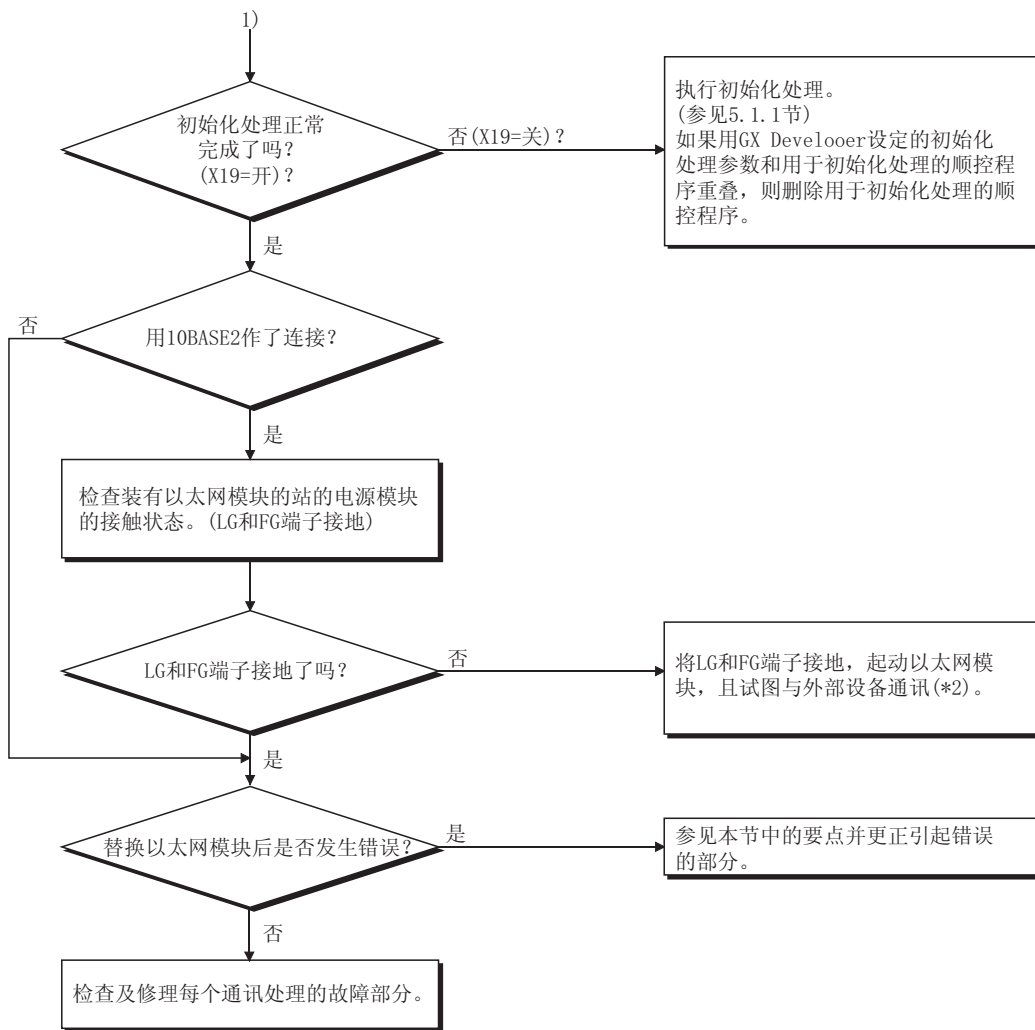
- (2) 当使用缓冲存储器进行通讯(无顺序)
- 因为在无顺序通讯中不指定信息数据长度，接收的数据存储在接收缓冲区中。建议设计一些方法来检查正确接收的数据。比如可以在信息的应用数据中包含数据长度和数据类型代码，这样在接受端就可以识别应用数据的字节数和数据类型。

11.4 故障排除流程图

本节以流程图的形式说明当以太网模块和外部设备存在通讯问题时，一些简单的故障排除步骤。（*1）

要点
 当使用以太网模块时发生故障，则如 11.2 所述，使用 GX Developer 以太网诊断功能检查以太网模块的模块状态、出错状态及其它状态。

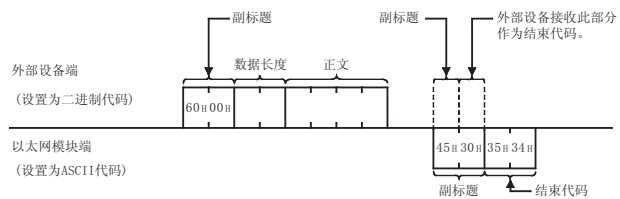




在固定缓冲存储器通讯时发送出错: 参阅11.4.1节。
 在固定缓冲存储器通讯时接收出错: 参阅11.4.2节。
 随机访问缓冲存储器通讯时的出错: 参阅11.4.3节。
 MC协议通讯时的出错: 参阅11.4.4节。
 电子邮件通讯时的发送错误: 参阅11.4.5节。
 电子邮件通讯时的接收错误: 参阅11.4.6节。
 冗余系统中的错误: 参阅11.4.7节

- *1 当 I/O 信号 X1C (COM. ERR LED ON 确认信号) 接通或当显示 LED COM. ERR (通讯出错检测显示) 亮时, 可参阅 11.1 节。
 (在出错发生期或出错代码储存在缓冲存储器中时, 核对处理用的专用指令控制表, 然后检查出错内容, 参阅 11.3 节, 采取正确的措施。)
- *2 如果以太网模块端 (参阅 4.7 节) 上的通讯数据设置和外部设备端的数据代码设置不同, 出错代码中没有发现的出错代码可能会返回到外部设备。
 如果接收的数据具有不同的数据代码, 以太网模块无法正确解码指令。以太网模块根据通讯数据代码设置返回出错响应。

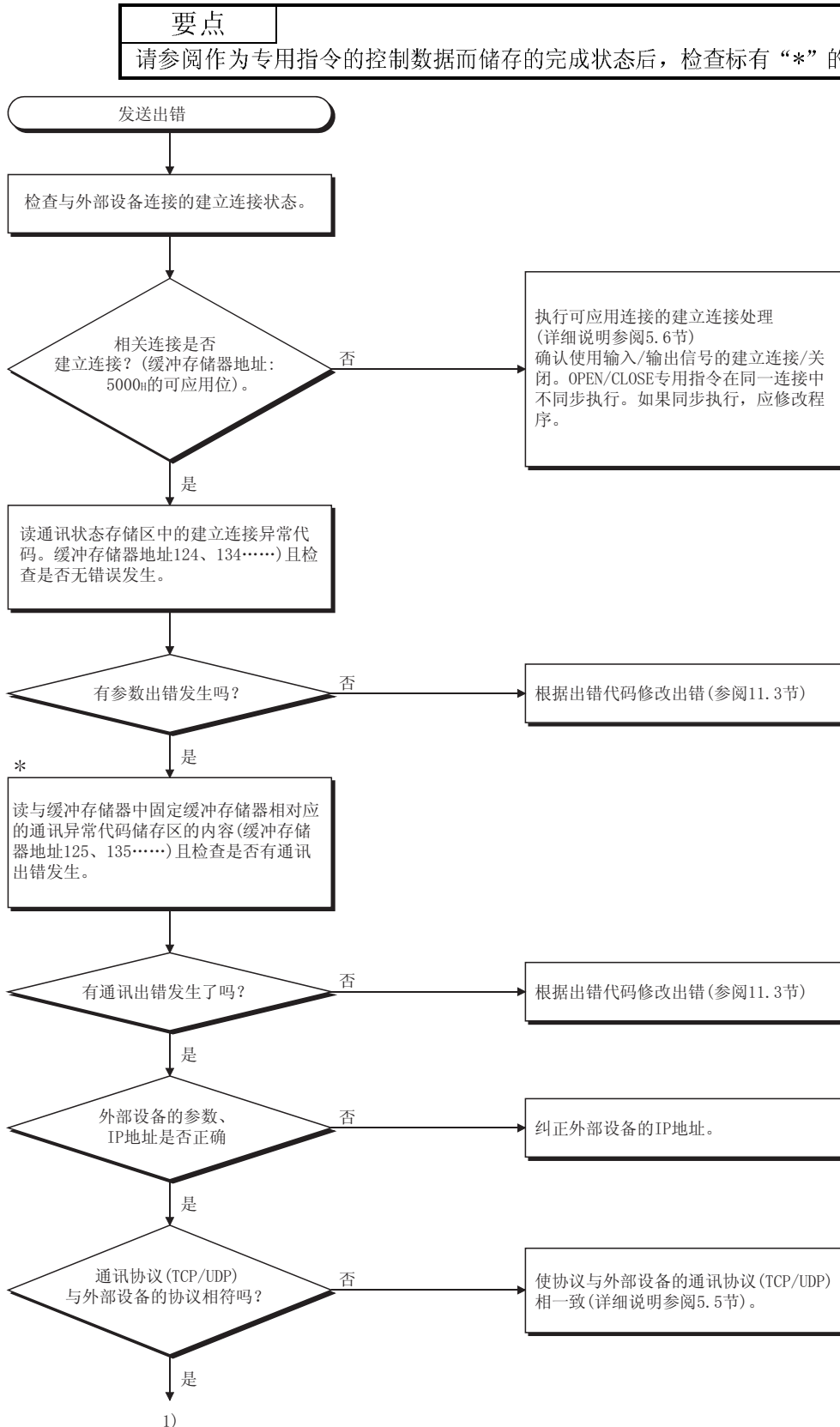
[举例] 使用固定缓冲器执行通讯时

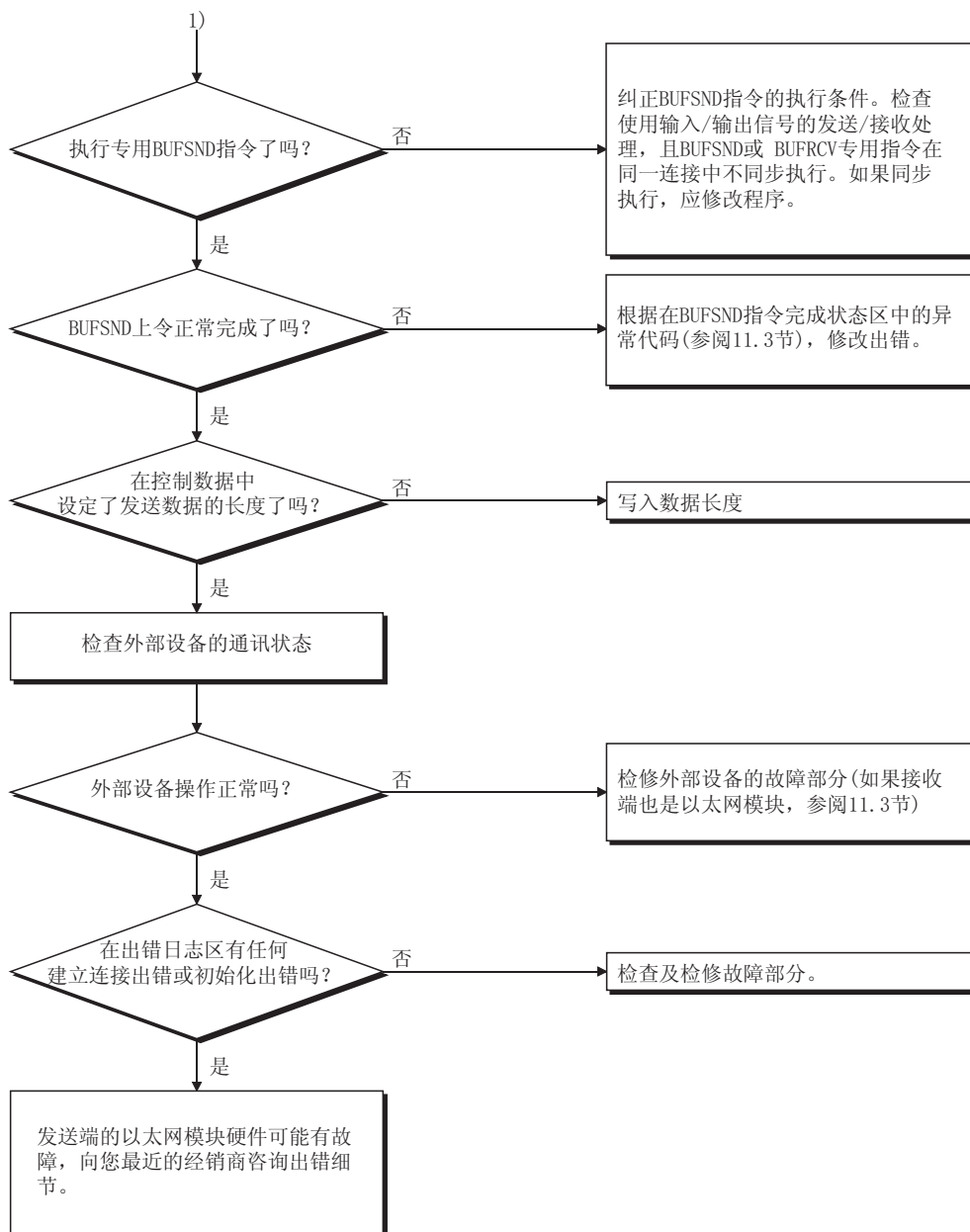


- *3 如果装有以太网模块的站的电源模块中的 LG 和 FG 端子未接地，则由于噪音的影响和外部设备的通讯线路关闭(连接断开)。
在参阅阐述接线步骤的 PLC CPU 用户手册中的相关章节后，断开装有以太网模块的站的电源，且将电源模块的 LG 和 FG 端子接地。

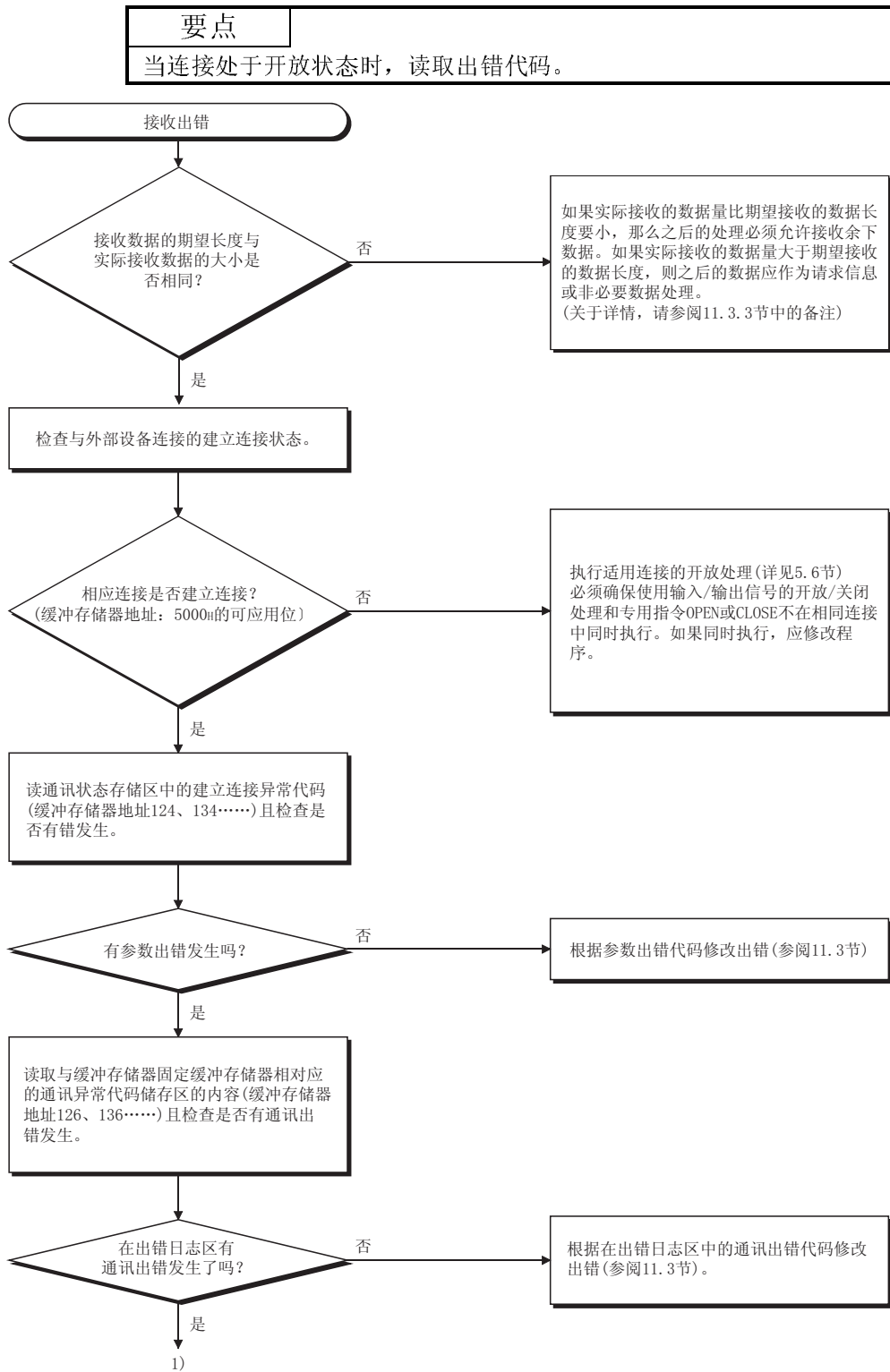
要点
<p>(1) 当由于出错而更换以太网模块时，重新启动以下的外部设备和数据通讯：</p> <ul style="list-style-type: none"> • 所有与以太网模块已更换的站进行通讯的外部设备。 • 所有通过以太网模块已更换的站，与其它站的 PLC CPU 进行通讯的外部设备。 <p>(2) 当将设备与以太网模块开放时，所要的设备和连接方法可参阅以下小节： 2.2 节：网络配所需要的设备 4.4 节：开放方式检查</p> <p>(3) 如果以太网模块无法接收外部设备多次发送的信息，则检查以下缓冲存储器中存储的数值。</p> <p>(a) 同步传送出错检查计数存储区(地址：18E_H 至 18F_H) 和用于每个出错日志块(地址：E5_H...) 的出错/结束代码存储区。 如果出错检测计数很高，或已存储了出错代码 C0C7H，这可能是由于所连接设备的数据发送所引起以太网线路的高负载导致。 为了减少以太网线路上的负载，应采取必要的措施，如分隔网络或减少数据发送计数。请在咨询了网络管理员后，再采取必要的措施。</p> <p>(b) TCP 包接收计数存储区(地址：1B8_H, 1B9_H) 如果即使更新 TCP 包接收计数也无法接收数据，在 TCP 最大段传送设置区(地址：1E_H)中设置 8000_H(禁止 TCP 最大段传送大小选项)并执行重新初始化。 (关于重新初始化，请参阅 5.2.3 节。)</p> <p>(4) 所有的专用指令都必须在线执行。 如果有任何的专用指令在离线时执行，虽然不会发生出错，但是专用指令的执行将不能完成。</p>

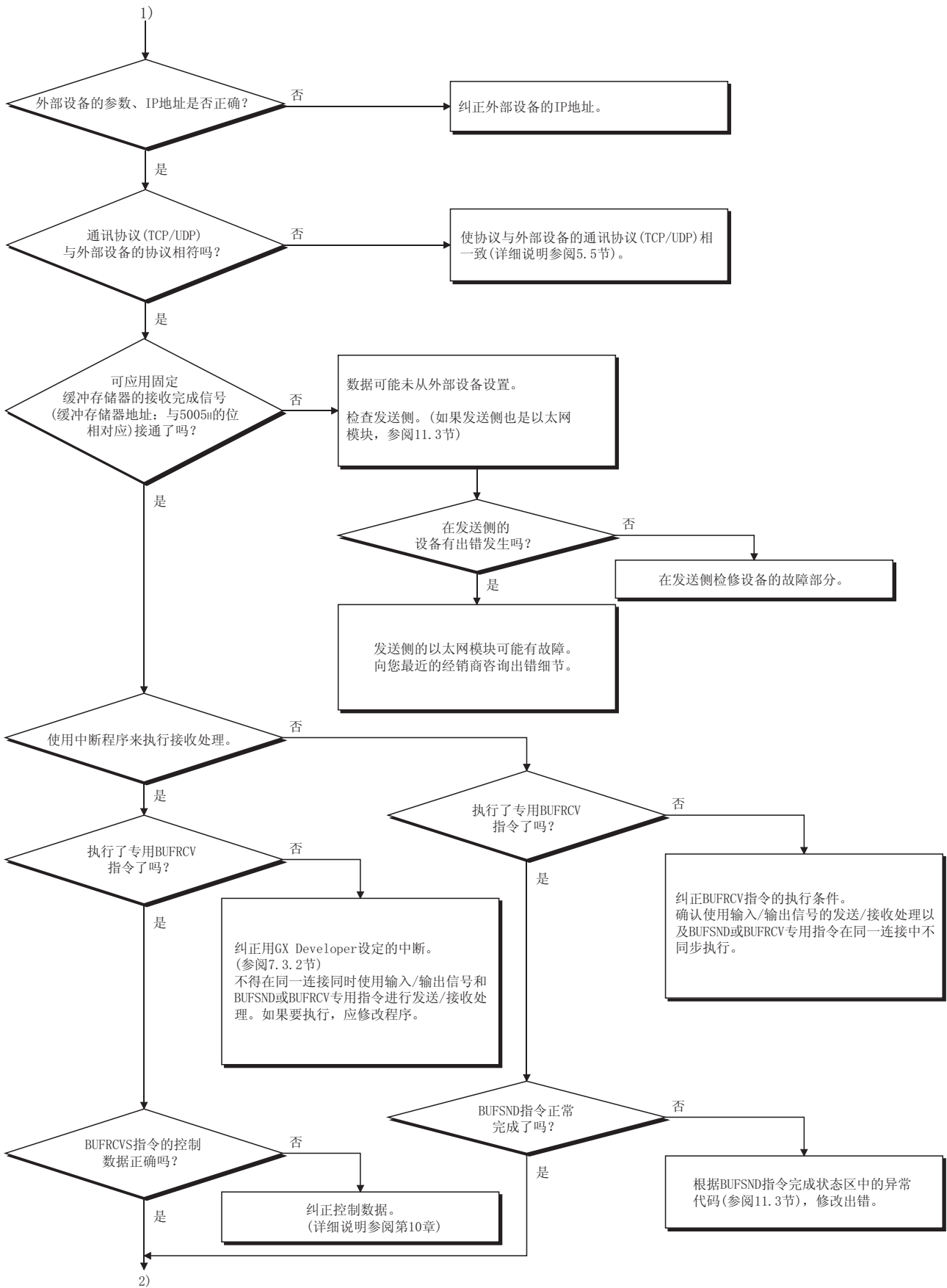
11.4.1 固定缓冲存储器通讯期间发送出错
(有顺序和无顺序共用)

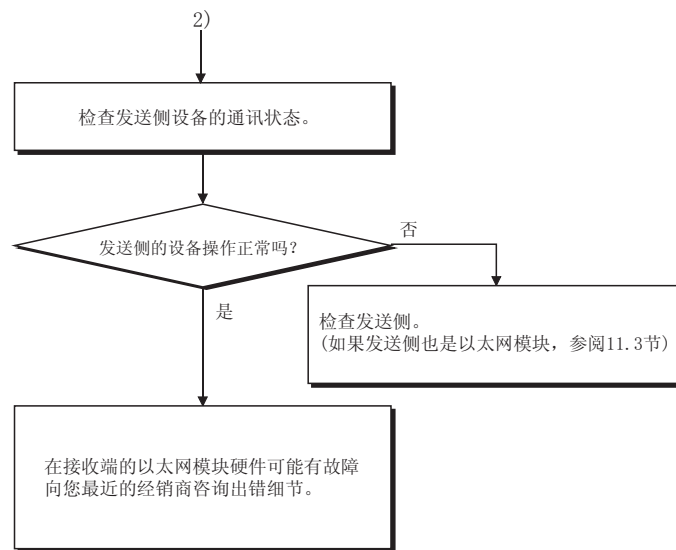




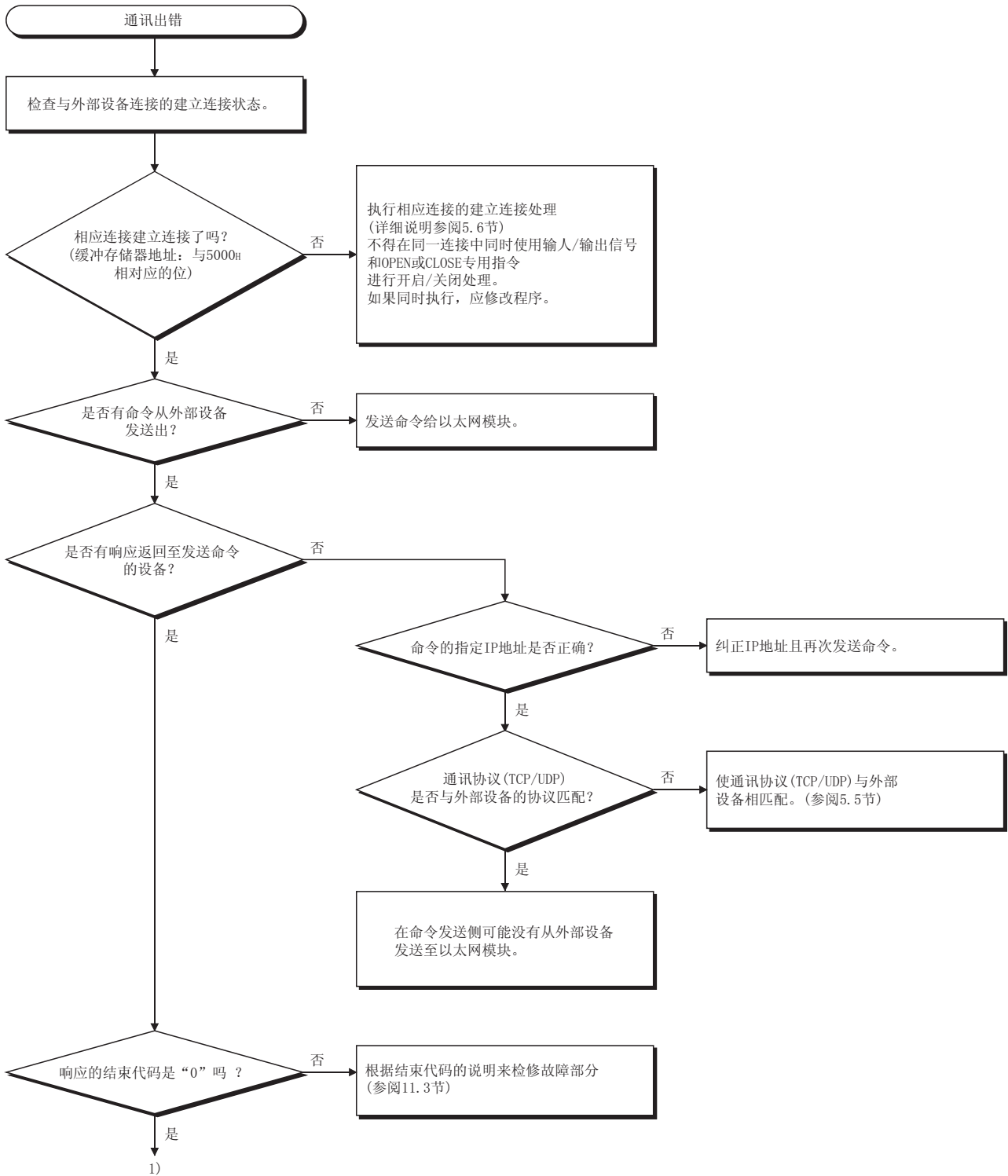
11.4.2 固定缓冲存储器通讯期间接收出错
(有顺序和无顺序共用)





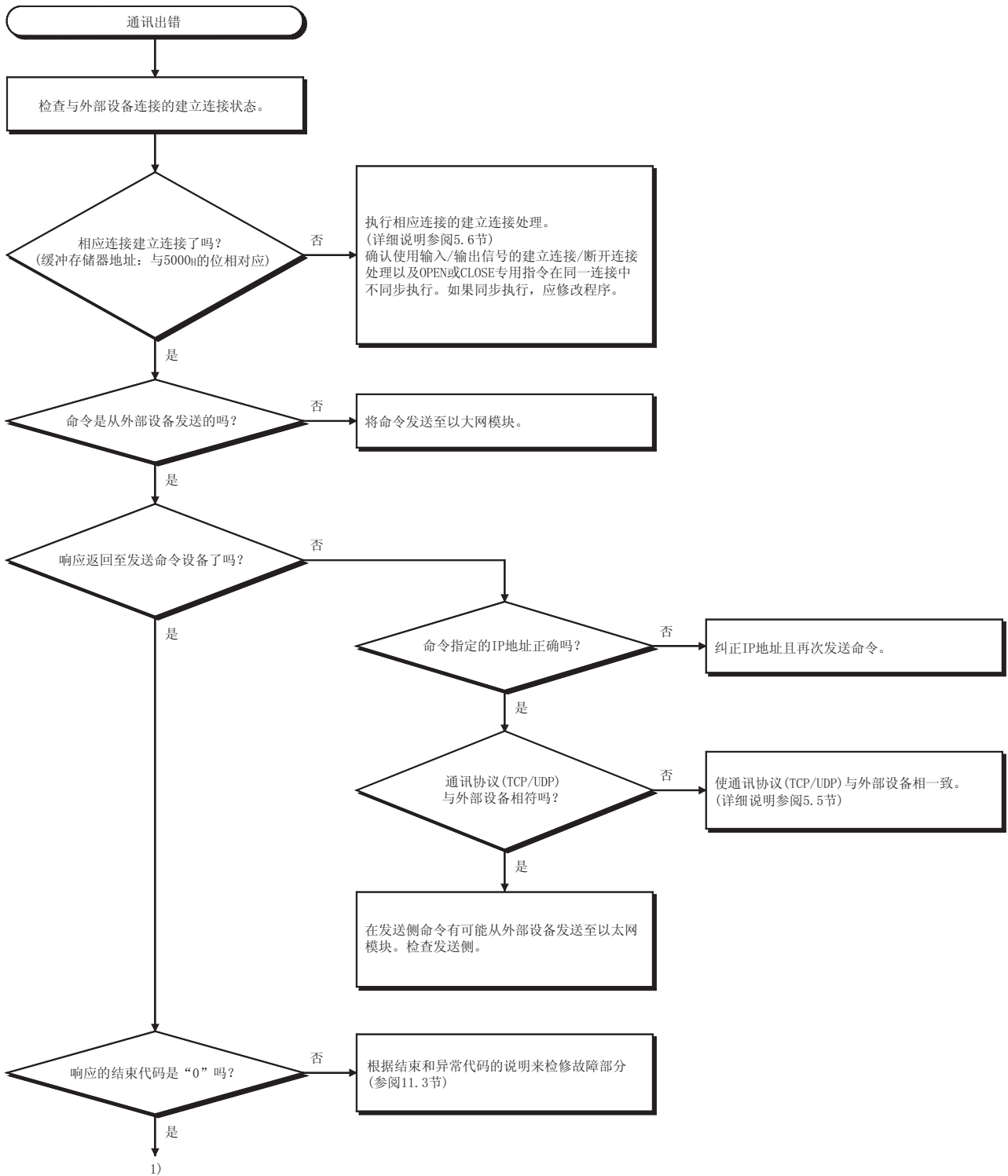


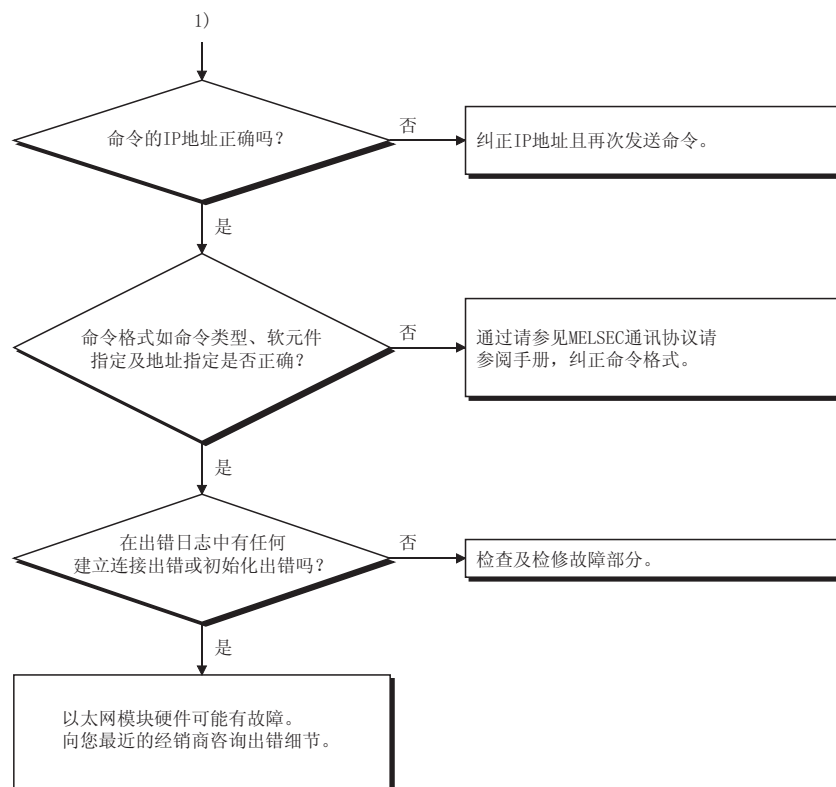
11.4.3 随机访问缓冲存储器通讯期间出错



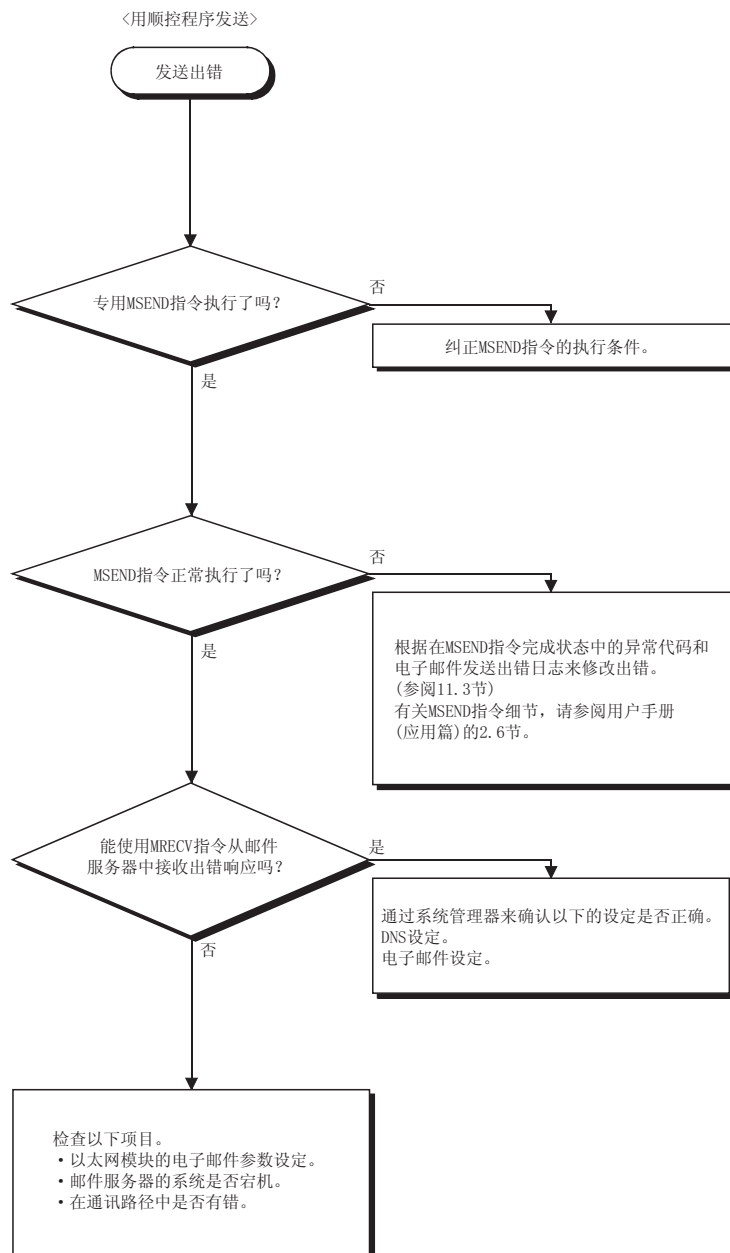


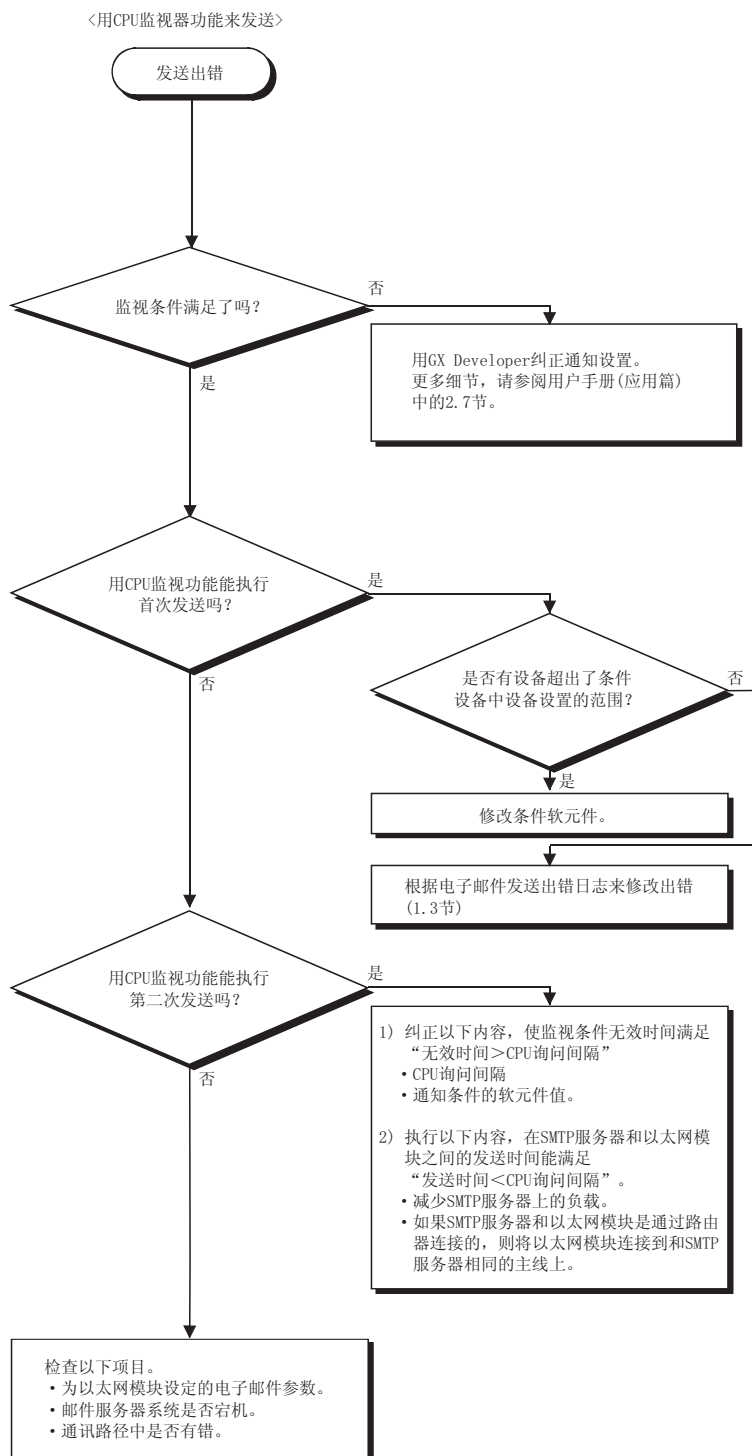
11.4.4 使用 MC 协议通讯出错





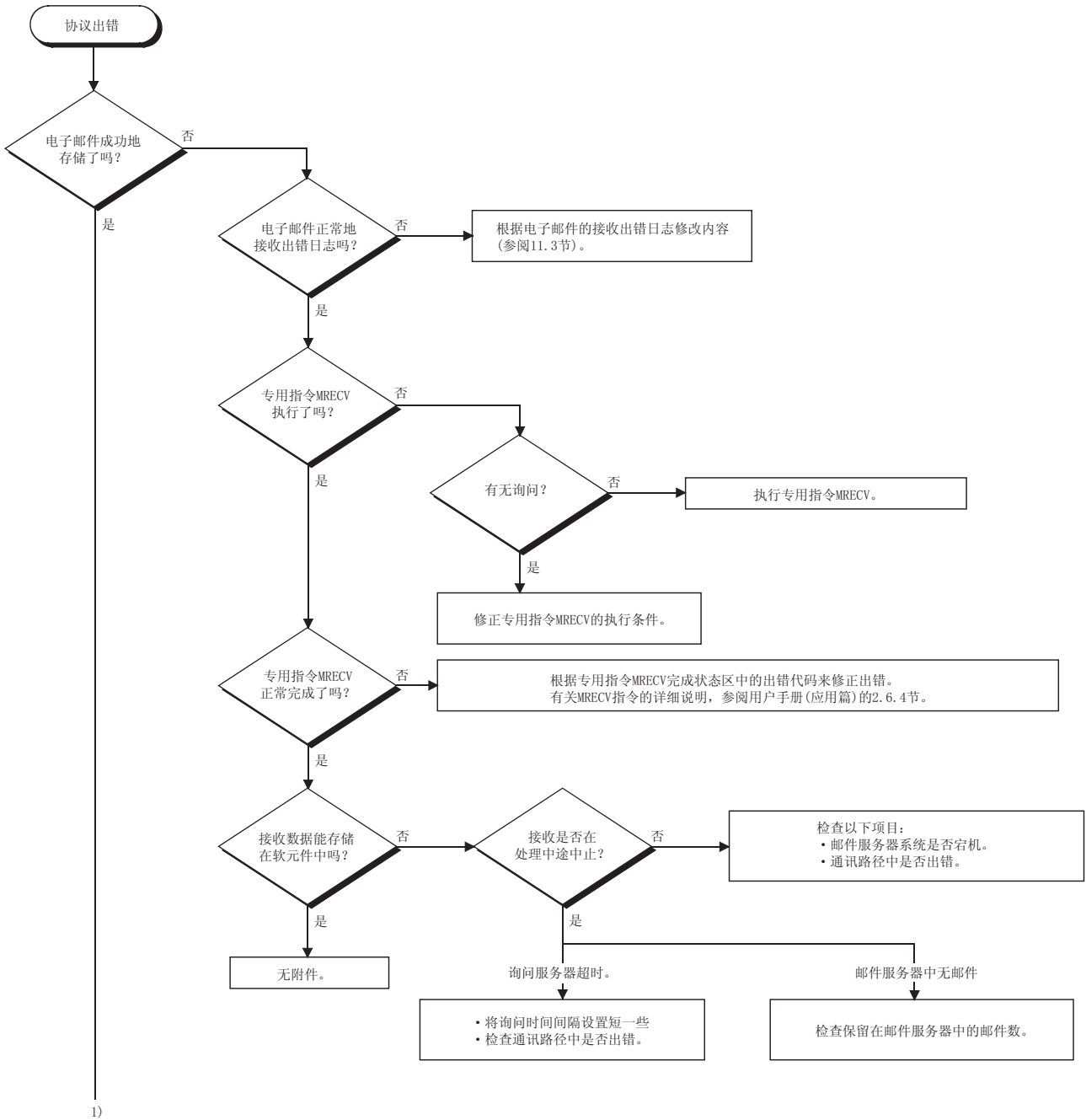
11.4.5 电子邮件通讯期间发送出错

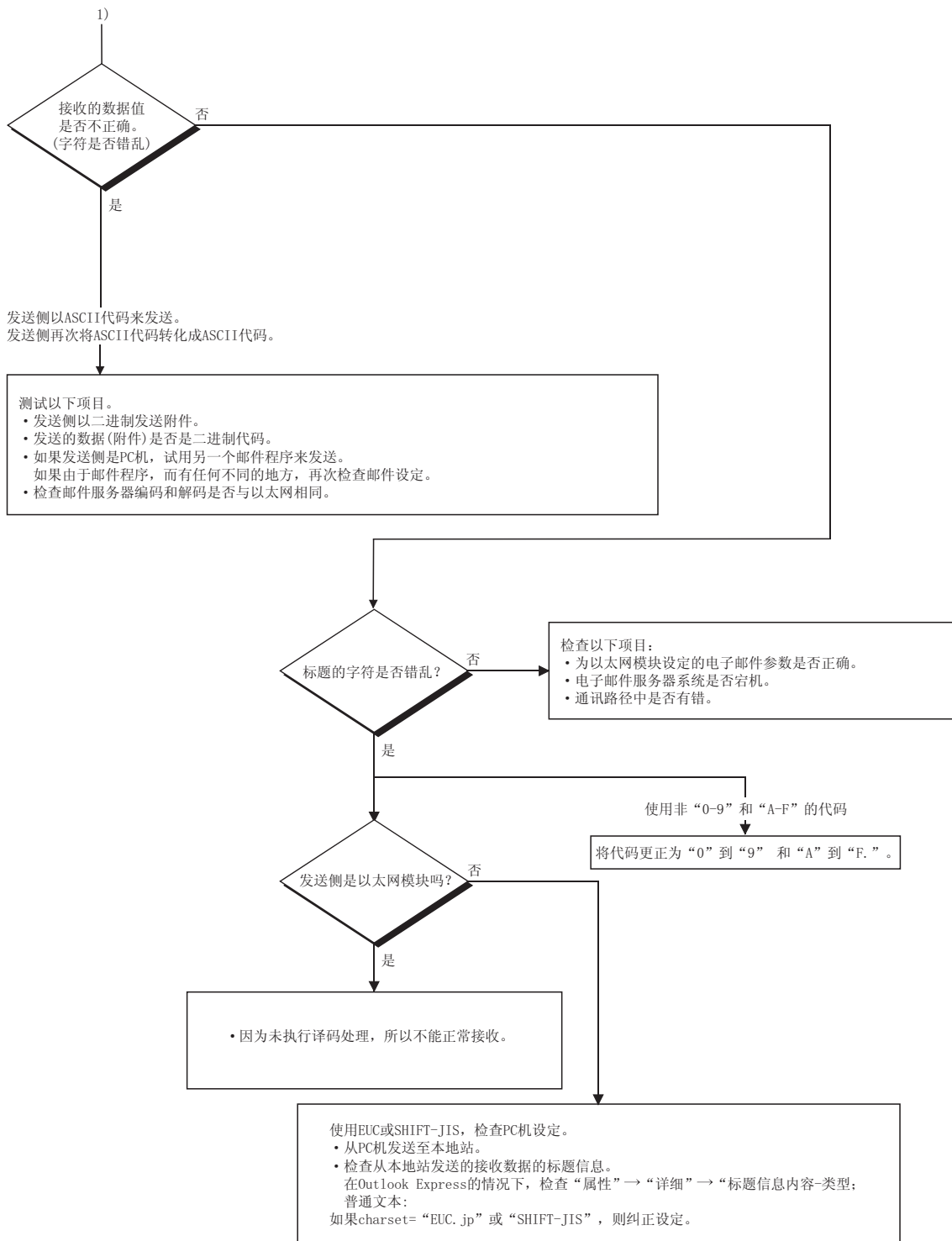




要点
<p>在使用电子邮件功能之前，检查以下项目：</p> <p>(1) 系统配置和环境： 用户手册(应用篇)中的 2.2 节</p> <p>(2) 使用电子邮件功能时的注意事项： 用户手册(应用篇)中的 2.3 节</p> <p>(3) 由 GX Developer 设置的电子邮件： 用户手册(应用篇)中的 2.5 节</p>

11.4.6 电子邮件通讯期间接收出错

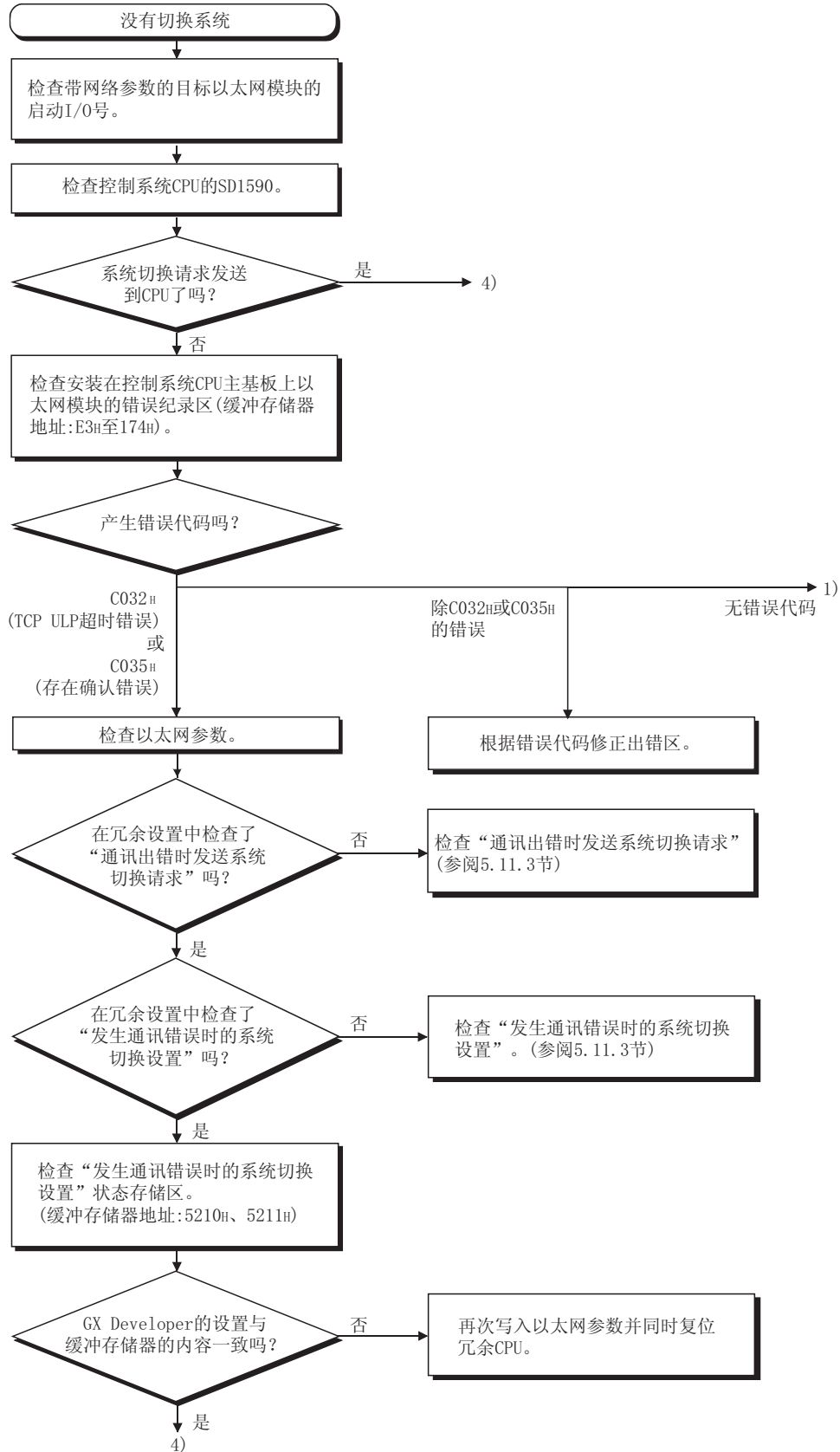


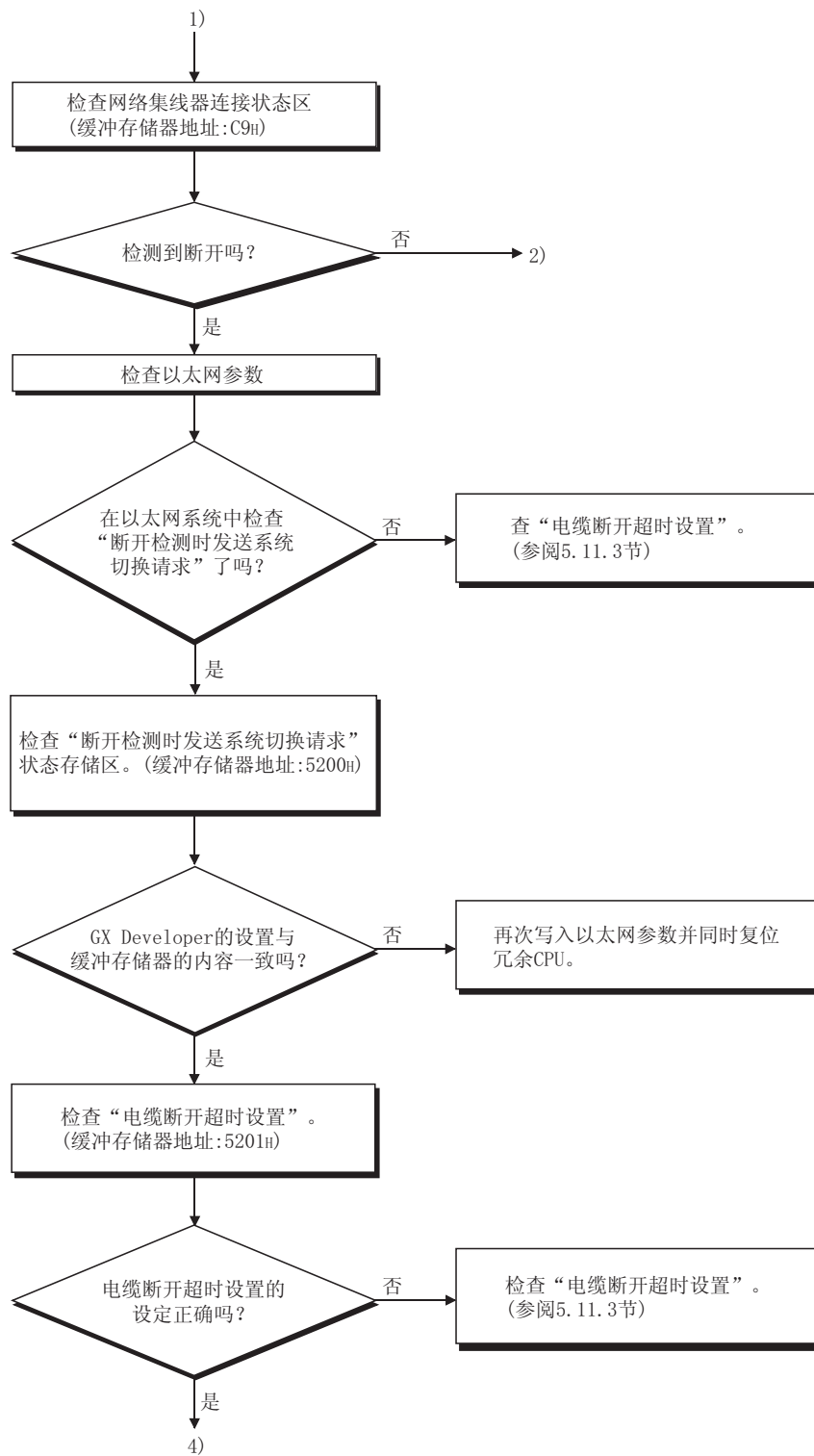


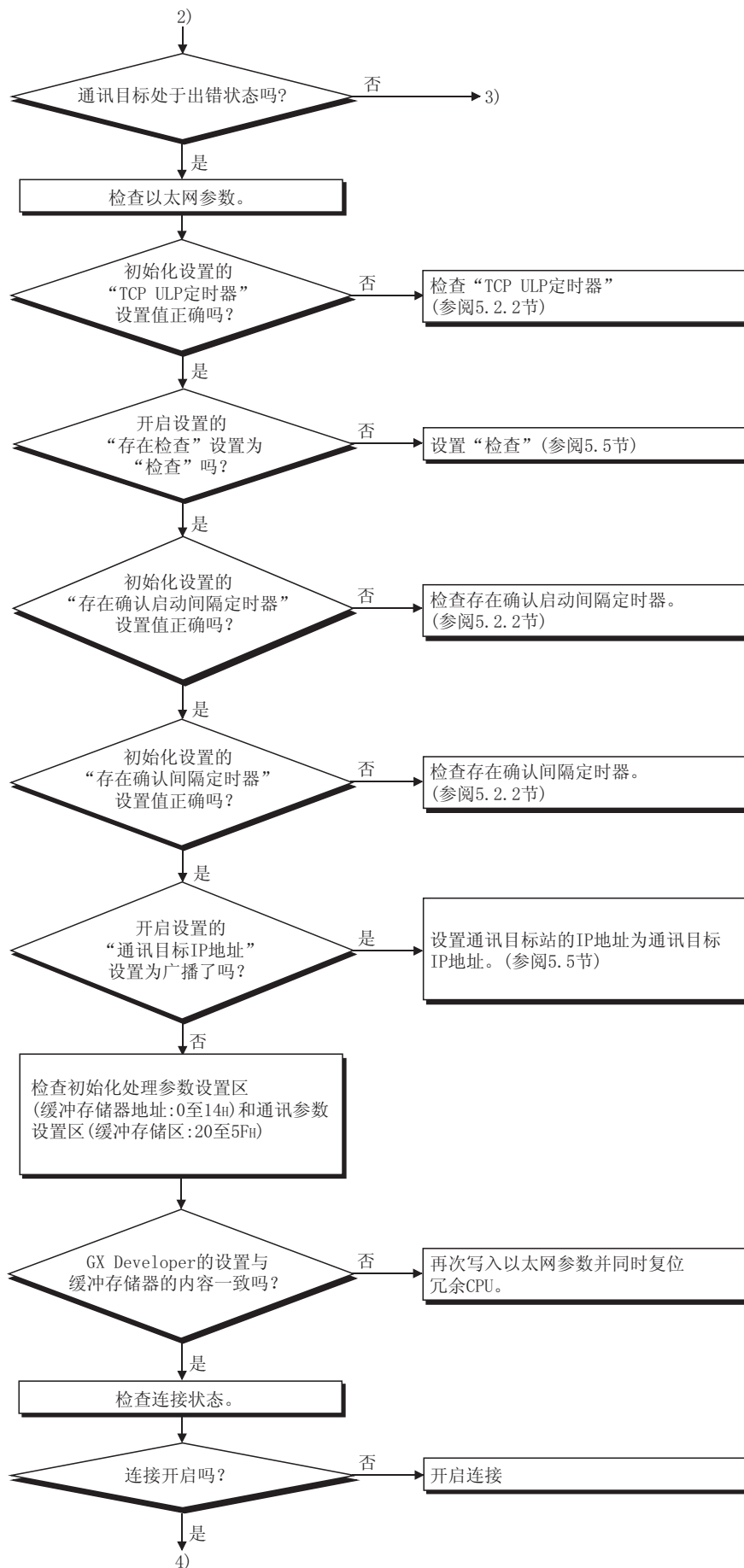
11.4.7 冗余系统中的错误

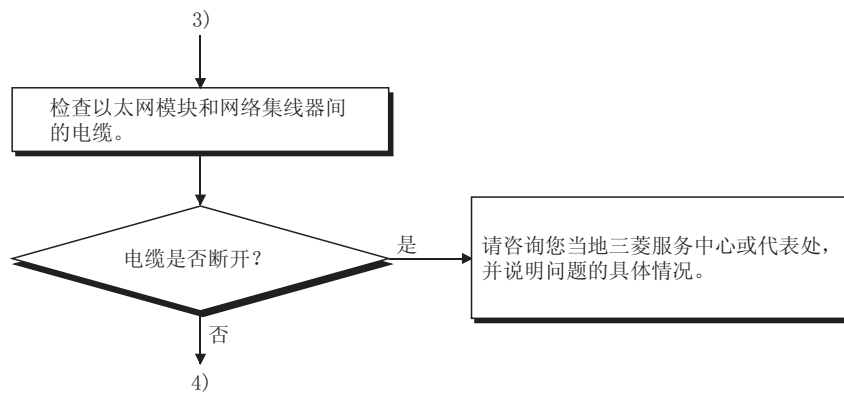
(1) 系统切换出错

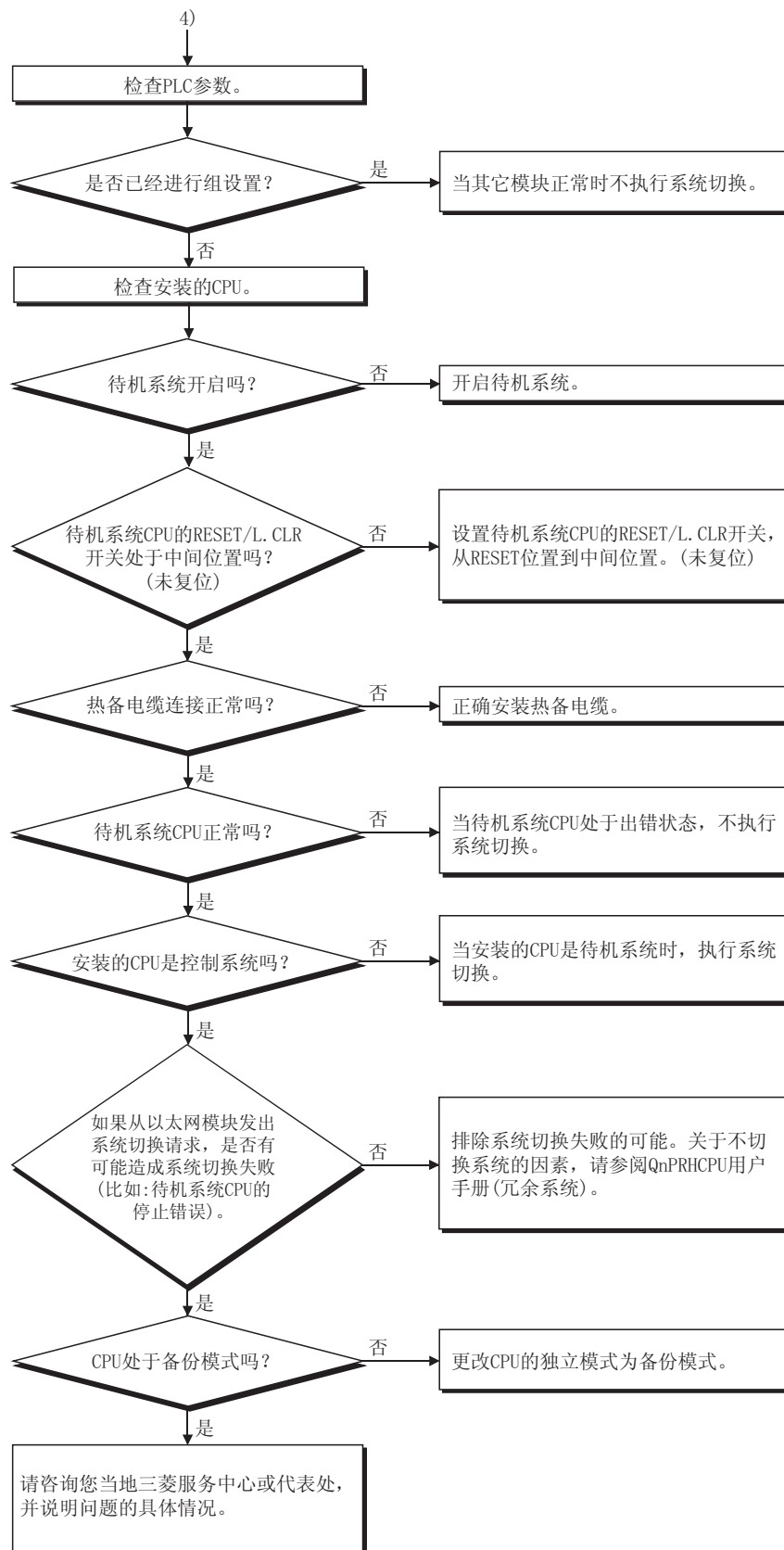
以下显示在通讯出错或断开检测时不发生系统切换的情况下的故障排除流程图。





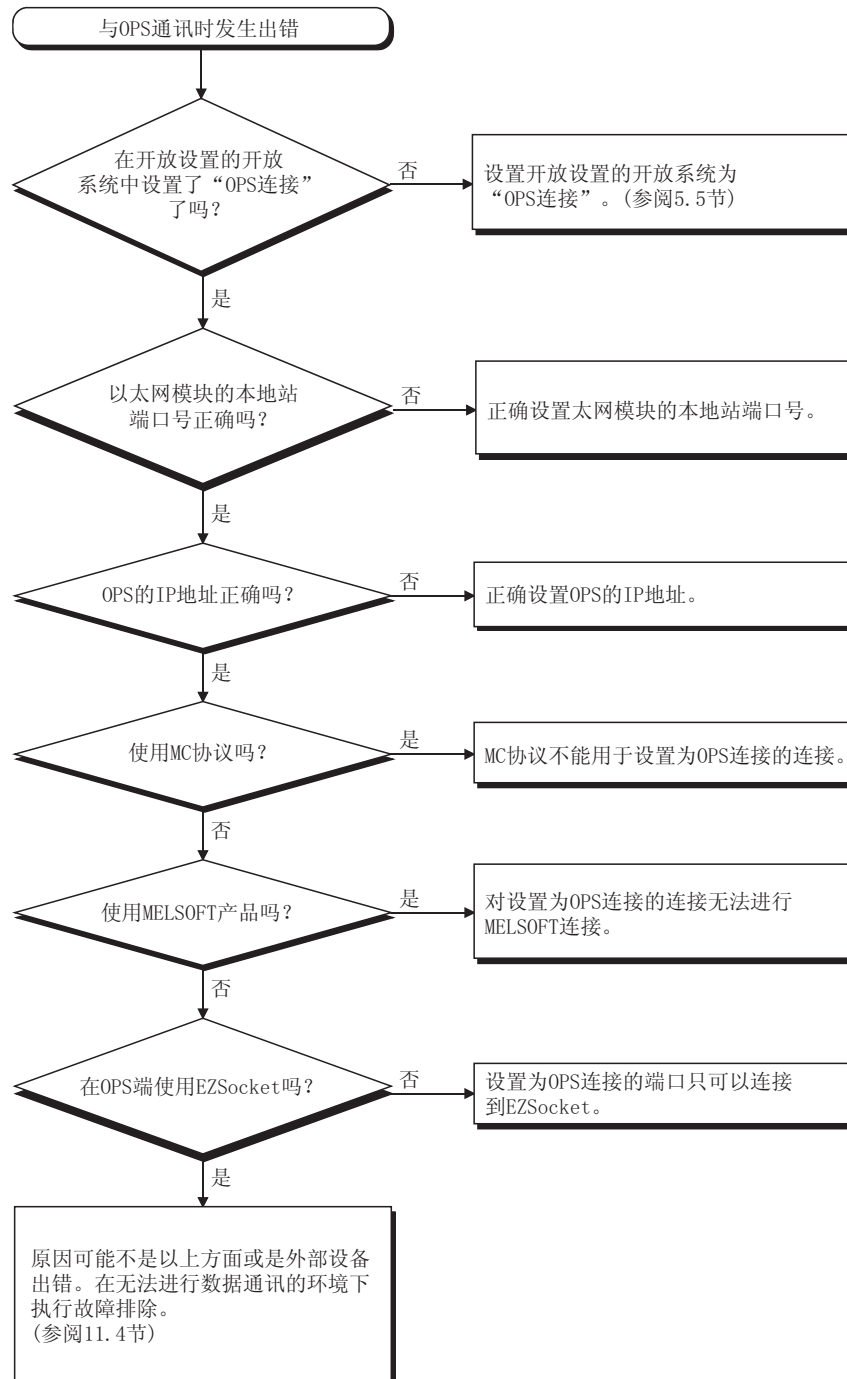






(2) OPS 连接通讯错误

以下显示在与 OPS 通讯时发生通讯错误的故障排除流程图。



附录

附录 1 以太网模块的功能升级

功能版本 B 或以上版本的以太网模块包含早期产品 (功能版本 A) 所没有的几种功能。本节以改变或增加的功能为基础, 对以太网模块的功能进行比较, 并说明程序的使用情况, 和把使用的程序结合进现有的系统中。

附录 1.1 以太网模块功能的比较

(1) 以太网模块功能的比较

下面是对以太网模块功能的比较。

功能		QJ71E71-100	QJ71E71-B5	QJ71E71-B2	
数据传送速率	100 Mbps	○	×	×	
	10 Mbps	○	○	○	
初始化处理	顺控程序	○	○	○	
	GX Developer 网络参数设置	○	○	○	
重新初始化处理	顺控程序	△	○	△	
	专用指令	△	○	△	
开放处理	顺控程序	○	○	○	
	GX Developer 网络参数设置	○	○	○	
	4E 帧	△	△	△	
使用 MC 协议通讯	QnA 兼容 3E 帧	○	○	○	
	A 兼容 1E 帧	○	○	○	
使用固定缓冲存储器通讯	有顺序	○	○	○	
	无顺序	○	○	○	
使用随机访问缓冲存储器通讯		○	○	○	
发送/接收电子邮件	• 通过 PLC CPU 发送/接收 • 通过 PLC 监视功能发送。(自动通知功能)	○	○	○	
	作为附件发送 CSV 格式文件	○	○	△	
	发送正文	○	○	△	
	支持解码/编码	△	○	△	
使用数据连接指令通讯	通过 PLC CPU 监视功能发送电子邮件正文中的字符串。	△	△	△	
	○	○	○		
使用数据连接指令通讯	目标站 CPU 类型规格	△	△	△	
	增加的数据长度(从 480 到 960 字)	△	△	△	
文件传送(FTP 服务器功能)		○	○	○	
使用 Web 功能通讯		○	○	△	
MELSECNET/H、MELSECNET/10 中继通讯		○	○	○	
路由中继通讯(路由中继功能)		○	○	○	
远程设备的存在确认(存在确认功能)	通过 PING 确认	○	○	○	
	通过 KeepAlive 确认	△	○	△	
通过成对开放通讯		○	○	○	
通过自动开放 UDP 端口通讯		○	○	○	
同时广播		○	○	○	
支持 QCPU 远程口令功能	远程口令检查	○	○	△	
	解锁处理/锁定处理	使用 MC 协议通讯	○	○	△
		GX Developer	○	○	△
		文件传送(FTP 服务器)功能	○	○	△
	使用 Web 功能通讯	○	○	△	

功能			QJ71E71-100	QJ71E71-B5	QJ71E71-B2
支持多 CPU 系统	安装到多 CPU 系统站		○	○	△
	访问非控制 PLC	使用 MC 协议进行	○	○	△
		GX Developer	○	○	△
		文件传送(FTP 服务器)功能	○	○	△
安装以太网模块到冗余系统			△	△	△
安装以太网模块到 MELSECNET/H 远程 I/O 站			○	○	△
设置参数以使用以太网模块功能			○	○	○
通过以太网模块(TCP/IP 或 UDP/IP)访问 QCPU			○	○	○
以太网诊断功能	监视不同的以太网模块状态		○	○	○
	通过以太网板	PING 测试	○	○	○
		自回路回送测试	○	○	△
	通过 CPU	PING 测试	○	○	△
支持 IEEE802.3 帧			○	○	△
MELSOFT 产品的连接(比如: GX Developer)			○	○	○
	通过 TCP/IP 通讯同时与多个装置连接		○	○	△
	简化访问其它站		△	○	△
	使用相同的站号访问		△	○	△
网络集线器连接状态监视功能			△	×	×

○: 可以使用。

△: 可以使用(然而, 产品系列号是有限制的(制造日期))(*1)。

×: 不能使用。

*1: 对支持此功能的以太网模块, 其生产日期有时间限制。关于如何检查功能版本, 请参见 2.7 节。

要点

关于以下项目的说明, 请参见 2.7 节。

- 如何检查以太网模块的功能版本。
- 与可以使用增加功能的相关产品(CPU 模块、GX Developer)进行通讯。

附录 1.2 将模块从功能版本 A 升级到 B 或以上版本的注意事项

本节说明为功能版本 A 的以太网模块创建的程序的利用、以及把该程序结合进现有系统。

(1) 程序利用

为功能版本 A 的以太网模块创建的程序可以用于功能版本 B 或以上版本。

(2) 结合进现有的系统

功能版本 A 的以太网模块接线方式适用于功能版本 B 或以上版本。

附录 2 QnA/A 系列模块

本节对以太网模块功能和 QnA/A 系列模块功能进行比较，并说明程序的使用及与现有的系统相结合。


QnA/A 系列模块参考下列产品：

型号名称	产品名称
AJ71E71	AJ71E71、A1SJ71E71-B2、A1SJ71E71-B5
AJ71E71-S3	AJ71E71-S3、A1SJ71E71-B2-S3、A1SJ71E71-B5-S3
AJ71E71N	AJ71E71N3-T、AJ71E71N-B5、AJ71E71N-B2、AJ71E71N-T、AJ71E71N-B5T、A1SJ71E71N3-T、A1SJ71E71N-B5、A1SJ71E71N-B2、A1SJ71E71N-T、A1SJ71E71N-B5T
QE71	AJ71QE71、AJ71QE71-B5、A1SJ71QE71-B2、A1SJ71QE71-B5
QE71N	AJ71QE71N3-T、AJ71QE71N-B5、AJ71QE71N-B2、AJ71QE71N-T、AJ71QE71N-B5T、A1SJ71QE71N3-T、A1SJ71QE71N-B5、A1SJ71QE71N-B2、A1SJ71QE71N-T、A1SJ71QE71N-B5T

附录 2.1 以太网模块和 QnA/A 系列模块之间的功能比较

下表列出了以太网模块和 QnA/A 系列模块之间的功能比较。

型号名称栏中的○符号表示相应的功能在可提供的型号中是兼容的。(详情参见每个模块的手册。)

 表示该功能已加到 QnA/A 系列模块中了，或者已对 QnA/A 系列模块进行了改进。

	功能	AJ71E71	AJ71E71-S3、AJ71E71N	QE71、QE71N		QJ71E71-100、QJ71E71-B5、QJ71E71-B2	备注
				9706 之前	9706B 之后		
1	初始化处理	○	○	○	○	○(* ³)	(* ⁹)
	顺控程序						
2	开放处理	×	×	×	○	○(* ³)	(* ⁹)
	GX Developer 参数设置						
3	使用固定缓冲器通讯	○	○	○	○	○(* ⁴)	-
	有顺序						
4	使用随机访问缓冲器通讯	×	○	○	○	○(* ⁴)	-
	无顺序						
5	PLC CPU 中的数据读/写 (使用 MC 协议通讯)	○	○	○	○	○	(* ¹⁰)
6	使用数据连接指令通讯	×	×	×	○(* ²)	○	用于 PLC CPU 间的通讯
7	中断处理 (在数据接收时)	×	×	×	×	○	BUFRCVS 指令
	固定缓冲通讯						
8	发送/接收电子邮件	×	×	×	×	○	(* ¹¹)
	数据连接通讯						
9	文件传送	×	×	×	○	○	FTP 服务器功能
	用自动通知功能发送						
10	用 Web 功能传送	×	×	×	×	○	-
11	同步广播	×	○	○	○	○	同步广播功能
12	PLC CPU 处于 STOP 状态时的通讯	×	○	×	○	○(* ⁶)	-
13	选择通讯数据代码 (ASCII/二进制)	○	○	○	○	○	-
14	MELSECNET/H、MELSECNET/10 中继通讯	×	×	×	○(* ²)	○	-
15	路由中继功能	×	○	○	○	○	路由中继功能
16	远程设备的存在确认	×	○	○	○	○	-
	Ping						
17	成对开放通讯	×	○	○	○	○	用于使用固定缓冲器的通讯
	KeepAlive						
18	每个定时器数据通讯设置值的单位	×(* ¹)	○	○(固定)	○(固定)	○(固定)	-
	500ms						
19	与 GX Developer 连接	×	×	×	×	○	用于 1:1 通讯
	2s						
20	安装 EEPROM	×	×	○	○	×(* ⁷)	通讯参数注册
	UDP/IP						
21	TCP 最大段大小选项传送	×	○(* ¹³)	×	○(* ¹³)	○(* ¹²)	-

○：该功能可用。×：该功能不可用。

- *1 可应用比“Q版本”还早的模块软件版本。
- *2 功能的应用性依据 PLC CPU 和 SWn1VD/NX-GPPQ 型 GPP 功能软件包的制造日期/软件版本而定。
- *3 使用输入/输出信号的顺控程序不能与用 GX Developer 设置的参数一起使用。
- *4 与输入/输出信号和 QE71 的缓冲存储器兼容。
- *5 功能是兼容的，但是不能与顺控程序的电子邮件发送/接收功能一起使用。
- *6 该功能由 GX Developer 设置的参数执行，因此不需要常规的顺控程序。
在常规程序中使用该功能时，在下列条件下不能应用该功能。（不能进行正确操作，因为使用了相同区域）
 - “始终等待打开”设置于网络参数的操作设置中。
 - “MELSOFT 连接”设置于网络参数的开放设置中。
 - 重新初始化处理（用 UINI 指令，缓冲存储器）用于顺控程序中。
- *7 没有安装 EEPROM。先前存储在 QE71 EEPROM 中的项目，现在由 GX Developer 设置的参数来存储。
- *8 执行用 GX Developer 设置的参数时，初始化处理在以太网模块启动时执行。无需常规的顺控程序。
- *9 可以从 PLC CPU 开放的以太网模块连接数已增加到 16 个。同时，在执行被动开放处理 TCP/IP 通讯和开放处理 UDP/IP 时，开放处理就会在以太网模块启动时，通过 GX Developer 设置的“始终等待打开”参数执行。无需常规的顺控程序。
- *10 使用以太网模块时，最多可以读/写 960 个字（QE71 最多可以读/写 480 个字）。
- *11 不能与使用随机存取缓冲存储器的通讯功能一起使用。
- *12 前 5 位系列号为 05051 或之后的产品可使用 TCP 最大段大小选项传送。
 - 前 5 位系列号为 05051 至 05081 的 QJ71E71-100、QJ71E71-B5、QJ71E71-B2，缺省值预先设置为“允许 TCP 最大段大小选项传送”。如果由于与其它代码相混淆而无法进行正常通讯，更改设置为“禁止 TCP 最大段大小选项传送”。
 - 要更改 TCP 最大段大小选项传送，需要更改 TCP 最大段大小选项传送并执行重新初始化。（关于重新初始化的详情，请参见 5.2.3 节）
- *13 当 QE71N 或 AJ71E71N 的软件版本为“E”或以上时，可使用 TCP 最大段大小选项传送。
注意，缺省值设置为禁止 TCP 最大段大小选项传送。（重新传送 TCP 数据时可以）

要点

该以太网模块对外部设备的响应速度比用于 A/QnA 系列的以太网模块的响应速度要快。

使用该以太网模块时，无法保证与用于 A/QnA 系列的以太网模块严格兼容。如果在考虑外部设备性能时存在疑问的话，请试着在使用 QCPU 恒定扫描设置和其它应用设置的常规系统的模块中设置定时关闭程序。

附录 2.2 使用专门为 QnA/A 系列模块设计的程序

先前在 PLC CPU 和使用 QnA/A 系列以太网接口模块 (如 AJ71E71) 的以太网上的外部设备之间执行的数据通讯也可以使用以太网模块来执行。

下面详细说明如何使用专门为传统模块设计的程序利用以太网模块执行数据通讯。

(1) 使用专门为 AJ71E71 (-S3) 和 AJ71E71N 设计的程序

(a) 使用专门为外部设备端设计的程序

以下部分为 AJ71E71 (-S3) 和 AJ71E71N (以下简称为 E71) 使用的外部设备的通讯功能程序, 它们可以用于与以太网模块的通讯。

但是, 由于以太网模块和 E71 的响应速度不同, 所以这些程序无法按照现状使用。首先要保证测试运行情况。

专门用于 E71 设计的程序		目的地			
		外部设备→ 以太网模块	以太网模块→ 外部设备	E71→ 以太网模块	以太网模块 → E71
通讯功能	使用固定缓冲存储器通讯 (有顺序)	○	○	○	○
	使用随机存取缓冲器通讯	○			
	对 PLC CPU 读/写数据(*1)	○			

○: 可以利用用于外部设备的 E71 程序进行通讯。

*1 只有 A 兼容 1E 帧命令可以用于数据通讯。
关于 E71 命令的详情, 请参见 MELSEC 通讯协议参考手册。
要使用除 A 兼容 1E 帧以外的命令执行数据通讯, 须创建一个新程序。

(b) 使用专门为本地站的 E71 设计的顺控程序。

以太网模块和 E71 分配有不同的缓冲存储器, 因此专门为 E71 设计的顺控程序不能用于以太网模块。

参考每个应用功能章节, 创建新程序。

(2) 使用专门为 AJ71QE71 (N) 设计的程序

(a) 使用专门为外部设备设计的程序

专门为 AJ71QE71 (N) (以下简称为 QE71) 设计的外部设备的顺控程序可以用于使用以太网模块的通讯中; 但是以下所列的程序除外:

- 与文件处理相关的命令的程序
(参考 MELSEC 通讯协议参考手册)
- 访问数据链接系统的程序
(QCPU(Q 模式) 模块无法连接到 MELSECNET (II) 和 MELSECNET/B。)

然而, 由于以太网模块和 QE71 具有不同的响应速度, 程序无法按现状使用。首先要保证测试运行情况。

(b) 使用专门为本地站 QCPU 设计的程序

- 1) 如果未对初始化处理和结束处理使用顺控程序，则不能写入 GX Developer (网络参数) 设置的以太网模块参数到 QCPU。
如果没有使用 GX Developer 以太网模块的参数设置，那么在进行通讯时需要注意以下方面：
 - QE71 通讯条件设置开关的所有设置值操作在关闭状态下进行。通过 5.2.3 节中说明的重新初始化处理来设置通讯条件。
 - MELSOFT 产品 (例如: GX Developer) 不能通过 MELSOFT 产品 (例如: GX Developer) 和以太网模块间的直接连接来访问 QCPU。
- 2) 如果使用 GX Developer 执行以太网模块的参数设置，删除初始化处理和结束处理的顺控程序。
- 3) 专门为本地站 QE71 设计的顺控程序可以与以太网模块通讯，但是以下列出的程序除外：
 - 访问数据链接系统的顺控程序。
(参考 QCPU 用户手册。)
 - 与 EEPROM 有关的顺控程序。
 - 建立 8 号连接的成对连接设置。
(参见第 5.7.1 节“成对开放”。)
 - 使用 EPRSET 指令的参数设置程序。

然而，由于以太网模块和 QE71 有不同的响应速度，实际情况下可能无法使用程序。必须首先测试运行情况。

备注

利用程序时，要谨记下列要点：

- 初始化处理的顺控程序不要与 GX Developer 用参数设置进行的初始化处理一起使用。
- 不要对同一连接同时执行下列操作：使用输入/输出信号进行的开放/关闭处理和使用 OPEN/CLOSE 专用指令进行的开放/关闭处理，以及使用 BUFSN/BUFRCV/BUFRCVS 指令进行的固定缓冲存储器发送/接收和发送/接收处理。

此外，设置以太网模块参数和设置与 QCPU 进行通讯时，一定要使用 GX Developer (SW4DSC-GPPW-E 或以后的版本)。

要点
<p>(1) 使用 Q 系列以太网模块时，毫无例外地要在参数设置画面上进行下列操作模式和通讯条件的 GX Developer 设置：</p> <ul style="list-style-type: none">• “设置 MNET/10H 以太网卡号的网络参数”画面。• “以太网操作设置”画面。 <p>(2) Q 系列以太网模块上没有进行通讯条件设置和操作模式设置的设置开关，而 QnA/A 系列以太网接口模块上有。</p> <p>(3) 执行被动开放处理时，在完成开放处理之前，Q 系列以太网模块不能取消开放请求。 在开放处理完成后执行关闭处理。</p>

附录 3 将以太网模块安装在现有的系统上

以太网模块和 QnA/A 系列以太网模块可以共存在同一个以太网上。使用 QnA/A 系列以太网接口模块的现有接线，可以将以太网模块连接在现有系统的以太网上。

附录 4 处理时间

使用下面的公式计算每个功能的最短处理时间。

注意，依据网络上的负载系数(线路上的拥挤程度)、连接的每个设备的窗口大小、同时使用的连接数以及系统的配置情况，处理时间可能会更长。当只使用一个连接进行通讯时，可以把下面公式计算出的数值当作处理时间的指导值。

(1) 使用固定缓冲存储器通讯的最短处理时间(以太网模块之间通讯)

(a) 使用固定缓冲存储器通讯(有顺序)

$$Tfs = St + Ke + (Kdf \times Df) + Sr$$

Tfs : 从发送开始到发送完成的时间(单位: ms)

St : 发送站扫描时间

Ke、Kdf : 常数(参见下表)

Df : 按字计数的发送数据

Sr : 接收站扫描时间

	QJ71E71-100				QJ71E71-B5、QJ71E71-B2			
	使用 TCP/IP 通讯		使用 UDP/IP 通讯		使用 TCP/IP 通讯		使用 UDP/IP 通讯	
	Ke	Kdf	Ke	Kdf	Ke	Kdf	Ke	Kdf
使用二进制代码进行数据通讯	12	0.0065	10	0.0069	25	0.020	20	0.019
使用 ASCII 代码进行数据通讯	12	0.030	10	0.029	26	0.068	21	0.068

(b) 使用固定缓冲存储器通讯(无顺序)

$$Tfs = St + Ke + (Kdf \times Df)$$

Tfs : 从发送开始到发送完成的时间(单位: ms)

St : 发送站扫描时间

Ke、Kdf : 常数(参见下表)

Df : 按字节计数发送数据

	QJ71E71-100				QJ71E71-B5、QJ71E71-B2			
	使用 TCP/IP 通讯		使用 UDP/IP 通讯		使用 TCP/IP 通讯		使用 UDP/IP 通讯	
	Ke	Kdf	Ke	Kdf	Ke	Kdf	Ke	Kdf
使用二进制代码进行数据通讯	7	0.0018	4	0.0014	16	0.0057	9	0.0025

[计算范例]

下面以 QJ71E71 使用 TCP/IP 通讯和使用固定缓冲存储器通讯(有顺序)发送 1017 个字的二进制代码数据为例，计算从发送开始到发送完成的时间(单位: ms)。

- 假定接收方面的扫描时间为 10ms，发送方面的扫描时间为 8ms:

$$63.34 \text{ (ms)} \cong 10 + 25 + (0.020 \times 1017) + 8$$

(2) 使用随机访问缓冲存储器进行通讯的最短处理时间

$$Trs = Kr + (Kdr \times Df) + \text{外部设备的 ACK 处理时间}$$

(仅在 TCP/IP 通讯时加上)

Trs : 以太网模块从接收 PC 发出的数据请求到处理完成所花费的时间(单位: ms)

Kr、Kdr : 常数(参见下表)

Df : 按字计数的请求数据

外部设备的 ACK 处理时间 :

在随机存取缓冲存储器的读取或写入完成之后, 直到外部设备返回一个 ACK 时需要的时间。

		QJ71E71-100				QJ71E71-B5、QJ71E71-B2			
		使用 TCP/IP 通讯		使用 UDP/IP 通讯		使用 TCP/IP 通讯		使用 UDP/IP 通讯	
		Kr	Kdr	Kr	Kdr	Kr	Kdr	Kr	Kdr
读取	使用二进制代码进行数据通讯	3.1	0.004	2.1	0.005	9.4	0.008	6.6	0.008
	使用 ASCII 代码进行数据通讯	3.1	0.016	2.2	0.016	9.1	0.030	6.5	0.030
写入	使用二进制代码进行数据通讯	3.1	0.006	2.1	0.005	9.5	0.014	6.6	0.012
	使用 ASCII 代码进行数据通讯	3.1	0.017	2.2	0.015	9.6	0.042	6.7	0.036

[计算范例 1]

下面以 QJ71E71-B5 和 PC 使用 TCP/IP 进行通讯, 并从随机存取缓冲存储器读取 508 个字的二进制代码数据为例, 计算 QJ71E71-B5 从接收 PC 发出的数据请求到处理完成所花费的时间(单位: ms)。

$$13.46 + \text{ACK 外部设备的处理时间(ms)} \doteq 9.4(0.008 \times 508) + \text{外部设备的 ACK 处理时间}$$

[计算范例 2]

下面以 QJ71E71-B5 和 PC 使用 TCP/IP 进行通讯, 将 508 个字的二进制代码数据写入随机存取缓冲存储器为例, 计算 QJ71E71-B5 从接收 PC 发出的数据请求到处理完成所花费的时间(单位: ms)。

$$16.61 + \text{外部设备的 ACK 处理时间(ms)} \doteq 9.5 + (0.014 \times 508) + \text{外部设备的 ACK 处理时间}$$

(3) 使用 MC 协议进行通讯的最短处理时间(批量读取和批量写入)

$Tfs = Ke + (Kdt \times Df) + Scr \times$ 处理要求的扫描次数+外部设备的 ACK 处理时间。

Tfs : 以太网模块从接收个人计算机发出的请求数据到处理完成所花费的时间。(单位: ms) *1

Ke 、 Kdt : 常数(参见下表)

Df : 请求数据的字数+响应数据的字数(应用数据部分)

Scr : PLC CPU 处理时间

(a) 当目标站为 QCPU

- 本地站访问:

- 安装以太网模块站的扫描时间

- 通过 MELSECNET/10 访问其它站

- 传输延迟时间+以太网模块-安装了以太网模块站的扫描时间。

(b) 当目标站在冗余系统中, 并且通过热备电缆中继数据

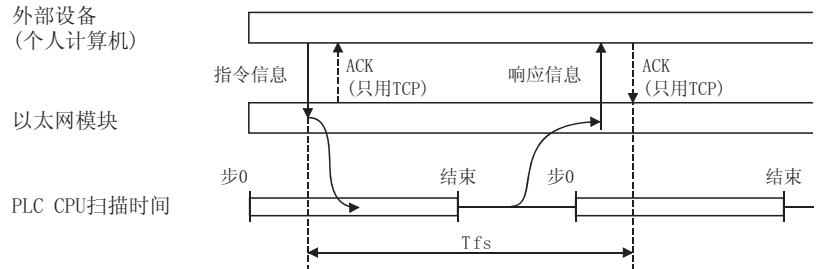
- 本地站访问:

- 控制系统 CPU 扫描时间+搜寻时间 *2

- 通过 MELSECNET/10 访问其它站

- 传输延迟时间+控制系统 CPU 扫描时间+搜寻时间 *2

*1 以下显示从以太网接收个人计算机发出的请求数据到完成处理的时间定时。



*2 通过冗余系统的热备电缆中继数据时, 会增加搜寻时间。关于热备电缆中数据传送时间, 请参见 QnPRHCPU 用户手册(冗余系统)。

		QJ71E71-100				QJ71E71-B5、QJ71E71-B2			
		使用 TCP/IP 通讯		使用 UDP/IP 通讯		使用 TCP/IP 通讯		使用 UDP/IP 通讯	
		Ke	Kdt	Ke	Kdt	Ke	Kdt	Ke	Kdt
批量 读取	使用二进制代码进行数据 通讯	14	0.009	13	0.008	21	0.012	19	0.011
	使用 ASCII 代码进行数据 通讯	18	0.015	13	0.017	23	0.020	18	0.020
批量 写入	使用二进制代码进行数据 通讯	14	0.009	13	0.008	21	0.020	19	0.013
	使用 ASCII 代码进行数据 通讯	16	0.027	14	0.027	22	0.037	20	0.033

[计算范例 1]

下面以 QJ71E71-B5 和 PC 使用 MC 协议通讯执行 TCP/IP 通讯，并从本站的数据寄存器 (D) 读取 100 点的 ASCII 代码数据为例，计算 QJ71E71-B5 从接收 PC 发出的数据请求到处理完成所花费的时间(单位: ms)。

- 假定安装了 QJ71E71-B5 的站的扫描时间是 10ms:

$$37.64 + \text{外部设备的 ACK 处理时间(ms)} \doteq$$

$$23 + (0.020 \times (21 + 211)) + 10 \times 1 + \text{外部设备的 ACK 处理时间}$$

$$\text{命令数据长度} = 21 \text{ 个字}$$

$$\text{响应数据长度} = 211 \text{ 个字}$$

[计算范例 2]

下面以 QJ71E71-B5 和 PC 使用 MC 协议通讯执行 TCP/IP 通讯，并将 100 点的 ASCII 代码数据写入本站的数据寄存器 (D) 为例，计算 QJ71E71-B5 从接收 PC 发出的数据请求到处理完成所花费的时间(单位: ms)。

- 当设置“允许在运行时间写入”
- 假定安装了 QJ71E71 的站的扫描时间是 10ms:

$$40.58 \text{ (ms)} \doteq 22 + (0.037 \times (221 + 11)) + 10 \times 1$$

$$\text{命令数据长度}=221 \text{ 个字}$$

$$\text{响应数据长度}=21 \text{ 个字}$$

(4) 专用指令的处理时间

下表表示了每个专用指令的大约操作处理时间。根据系统配置和通讯协议的不同，操作处理时间也有所不同。

(a) QJ71E71-100

指令名称	存取点数		处理时间(单位: ms)						指令执行条件		
			Q02CPU		Q02H/Q06H/Q12H/ Q25H/Q12PH/Q25PH/ Q12PRH/Q25PRHCPU		Q00J/Q00/Q01CPU				
	1)	2)	存取点数 1)	存取点数 2)	存取点数 1)	存取点数 2)	存取点数 1)	存取点数 2)			
BUFRCV	1 个字	1017 个字	1.3	1.8	0.9	1.4	2.2	5.8	TCP/IP 通讯, 二进制代码通讯 (固定缓冲存储器通讯 (有顺序))		
BUFRCVS			0.5	0.9	0.3	0.7	0.8	2.9			
BUFSND			12.8	19.2	11.5	18.1	14.0	23.5			
CLOSE	1 个端口		3.3		3.2		4.2		关闭 UDP/IP 通讯端口		
ERRCLR	清除所有出错信息		2.2		2.0		3.4		-		
ERRRD	读取初始异常代码		1.2		0.8		2.4		-		
OPEN	1 个端口		3.8		3.0		4.2		开放 UDP/IP 通讯端口		
RECVS	1 个字	960 个字	0.6	0.9	0.3	0.7	0.8	1.5	安装了以太网模块的站之间的 通讯		
		480 个字		0.8		0.5		1.2			
READ, SREAD		960 个字	17.2	28.8	17.1 *1	28.2 *1	14.7	24.3			
		480 个字								22.7	21.7 *1
RECV		960 个字	2.1	4.3	2.0	3.8	1.8	6.8			
		480 个字								3.2	2.9
发送		960 个字	7.9	15.7	7.5	15.4	11.5	16.4			
		480 个字								11.2	10.8
WRITE, SWRITE		960 个字	17.3	28.8	17.0 *1	28.4 *1	14.5	24.4			
		480 个字								23.0	22.2 *1
ZNRD		230 个字		14.4	17.1	13.8	16.6	12.1		14.1	
ZNWR				14.2	17.5	13.9	16.4	12.0		14.8	
UINI		-		21.5		21.2		21.8		从接收 UINI 指令到完成重新 初始化处理的时间。(开放 X19)	

*1 当通过冗余系统的热备电缆中继数据时, 会增加搜寻时间。关于热备电缆中的数据传送时间, 请参见 QnPRHCPU 用户手册(冗余系统)。

(b) QJ71E71-B5、QJ71E71-B2

指令名称	存取点数		处理时间(单位: ms)						指令执行条件		
			Q02CPU		Q02H/Q06H/Q12H/ Q25H/Q12PH/Q25PH/ Q12PRH/Q25PRHCPU		Q00J/Q00/Q01CPU				
	1)	2)	存取点数 1)	存取点数 2)	存取点数 1)	存取点数 2)	存取点数 1)	存取点数 2)			
BUFRCV	1 个字	1017 个字	1.9	2.4	1.2	1.6	2.3	5.8	TCP/IP 通讯, 二进制代码通讯, 固定缓冲存储器通讯 (有顺序)		
BUFRCVS			0.5	0.9	0.3	0.7	0.8	2.9			
BUFSND			27.6	45.3	24.5	45.0	28.2	50.0			
CLOSE	1 个端口		4.5		4.5		6.0		关闭 UDP/IP 通讯端口		
ERRCLR	清除所有出错信息		2.7		2.2		3.4		-		
ERRRD	读初始异常代码		1.7		1.1		2.5		-		
OPEN	1 个端口		4.3		3.3		5.2		开放 UDP/IP 通讯端口		
RECVS	1 个字	960 个字	0.6	1.0	0.3	0.7	0.8	1.6	安装了以太网模块的站之间的通讯		
		480 个字		0.8		0.5		1.2			
READ、SREAD		960 个字	30.1	52.1	27.7 *1	52.3 *1	27.7	50.1			
		480 个字		41.1		40.0 *1		38.9			
RECV		960 个字	5.3	7.9	5.2	7.4	5.3	11.1			
		480 个字		6.6		6.3		8.2			
发送		960 个字	21.4	39.4	20.3	37.9	22.8	38.8			
		480 个字		30.4		29.1		30.8			
WRITE、SWRITE		960 个字	30.0	53.6	29.4 *1	52.4 *1	28.2	47.6			
		480 个字		41.8		40.9 *1		37.9			
ZNRD		230 个字	29.0	34.3	29.0	34.7	27.8	33.2			
ZNWR			29.7	36.4	29.4	35.2	27.6	33.4			
UINI		-		21.5		21.2		21.8		从接收 UINI 指令到完成重新初始化处理的时间。(开放 X19)	

*1 当通过冗余系统的热备电缆中继数据时, 会增加搜寻时间。关于热备电缆中的数据传送时间, 请参见 QnPRHCPU 用户手册(冗余系统)。

(5) 冗余系统的系统切换时间

以下表示在发生通讯出错或断开诊断时安装在冗余系统中控制系统 CPU 主基板上的以太网模块向控制系统 CPU 发出系统切换请求时，所需的系统切换时间。
系统切换时间是从通讯出错或检测断开直到控制系统 CPU 切换到待机系统 CPU。

(a) 当检测通讯出错时

1) 发生 ULP 超时

$$T_{nc} = T_{tu} + S_t + T_{cc}$$

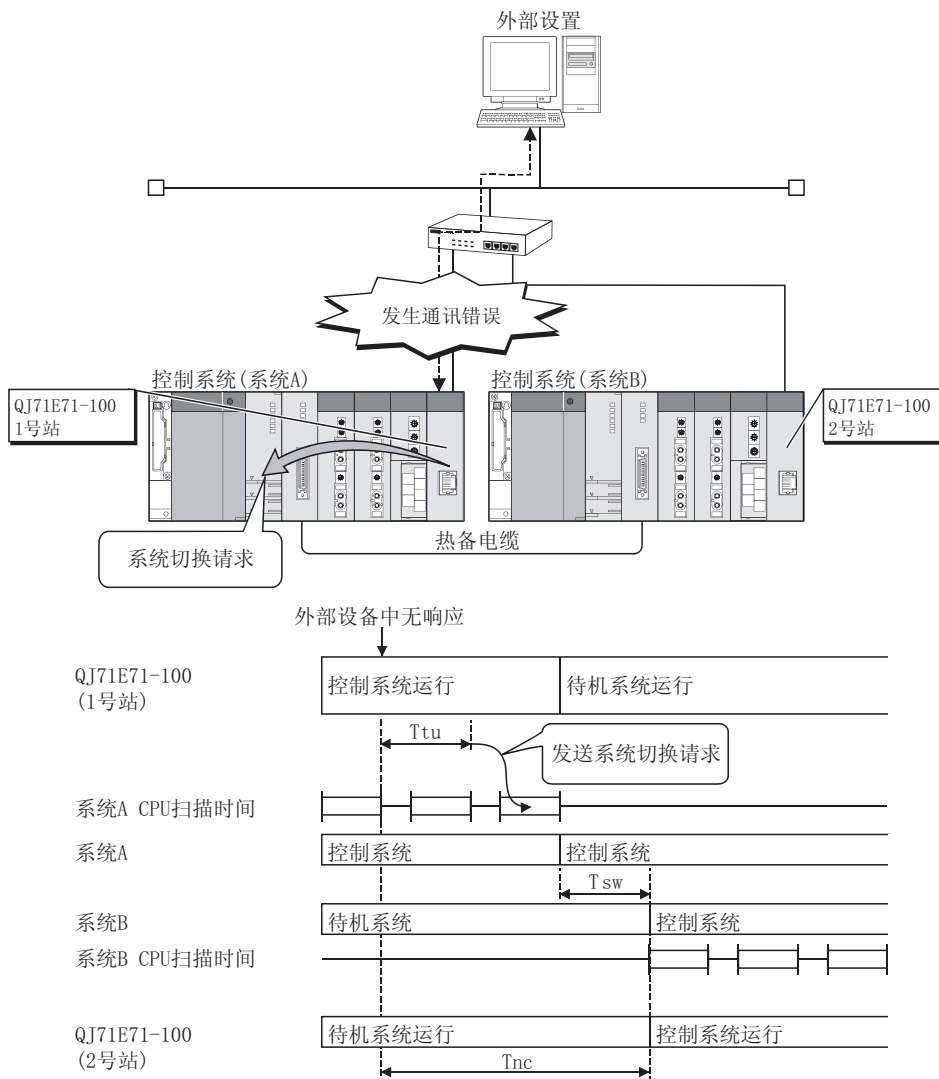
T_{nc} : 系统切换时间

T_{tu} : TCP ULP 定时器值

S_t : 1 次扫描时间

T_{sw} : CPU 系统切换时间(参见 QnPRHCPU 用户手册(冗余系统)。)

以下显示发生 ULP 超时时的系统切换操作定时。

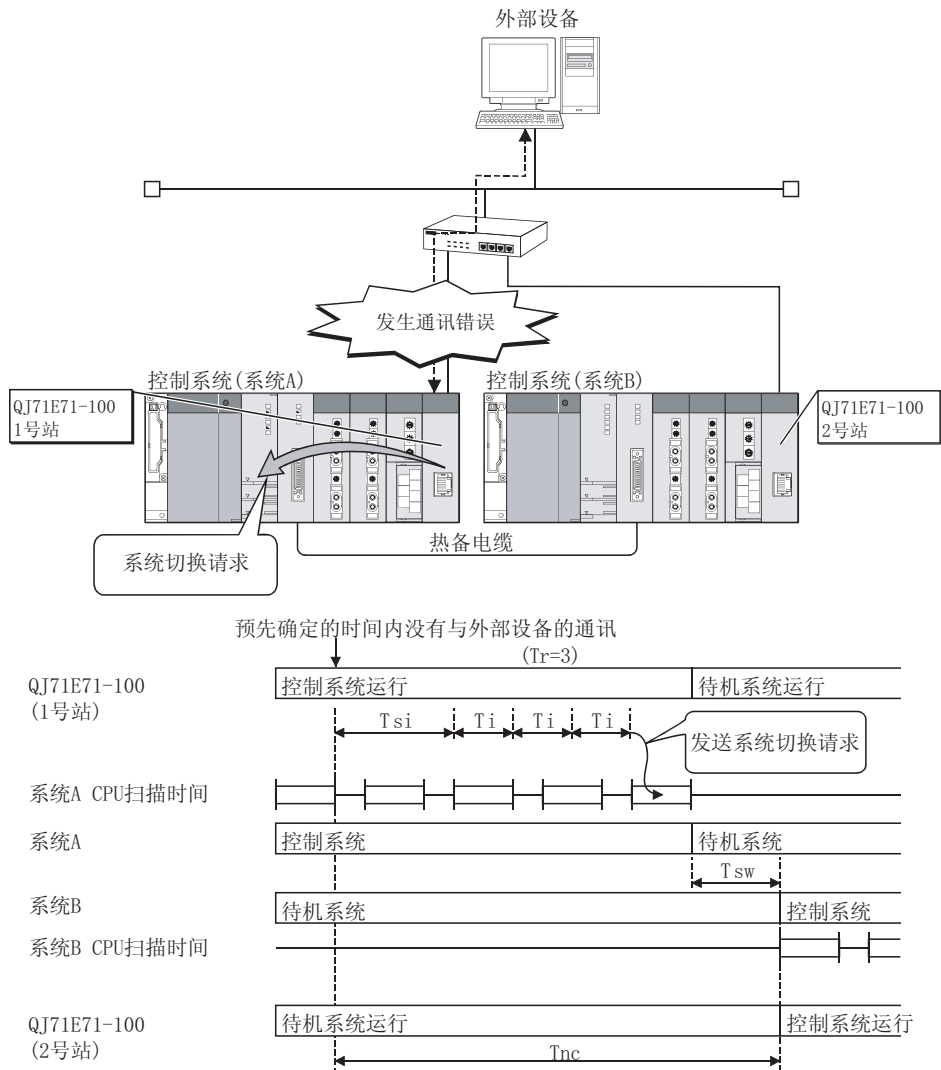


2) 发生存在确认错误时

$$T_{nc} = T_{si} + T_i \times T_r + T_s + T_{cc}$$

- T_{nc} : 系统切换时间
- T_{si} : 存在确认启动间隔定时器值
- T_i : 存在确认定时器值
- T_r : 存在确认重新发送计数
- T_s : 1次扫描时间
- T_{sw} : CPU系统切换时间(参见 QnPRH CPU 用户手册(冗余系统)。)

以下显示发生存在确认错误时的系统切换操作定时。

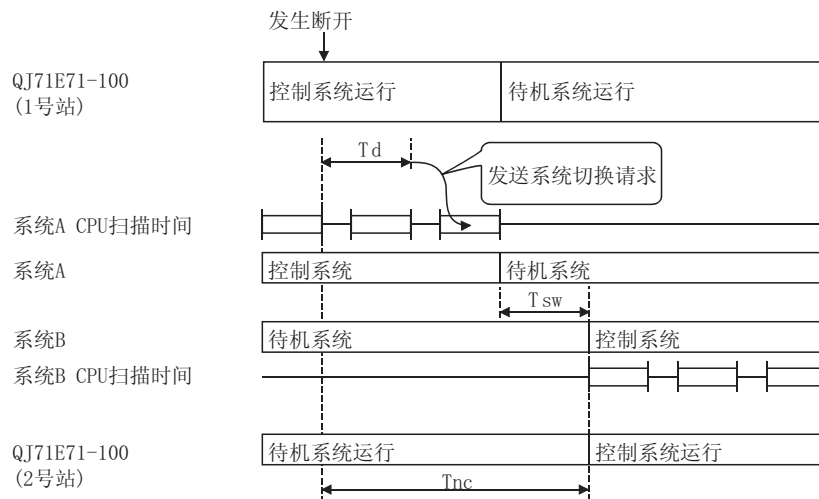
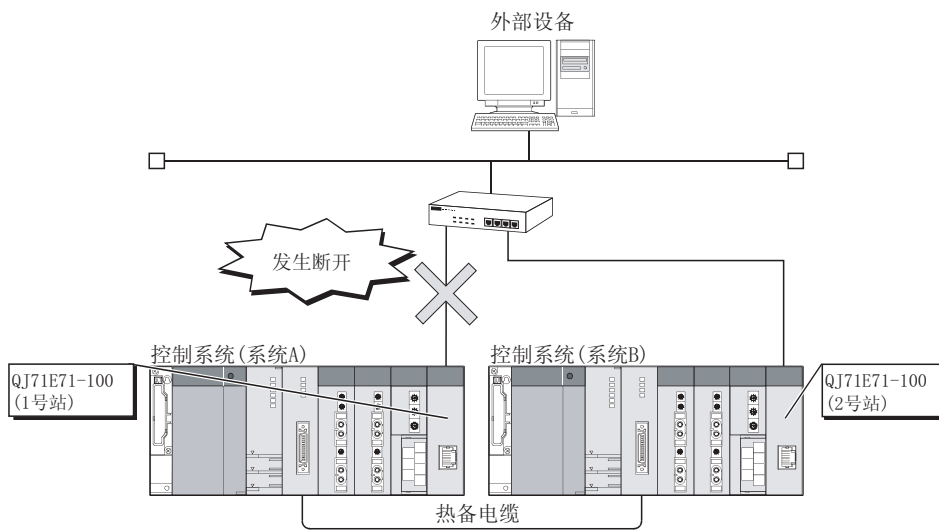


(b) 检测断开时

$$T_{nc} = T_d + S_t + T_{ec}$$

- T_{nc} : 系统切换时间
- T_d : 电缆断开超时
- S_t : 1次扫描时间
- T_{sw} : CPU系统切换时间(参见 QnPRHCPU 用户手册(冗余系统)。)

以下显示检测断开时的系统切换操作定时。



附录 5 ASCII 代码代表

LSD \ MSD		0	1	2	3	4	5	6	7
		000	001	010	011	100	101	110	111
0	0000	NUL	DLE	(SP)	0	@	P	~	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(8	H	X	h	x
9	1001	HT	EM)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	—	=	M]	m	}
E	1110	SO	RS	.	>	N	^	n	~
F	1111	SI	US	/	?	O	-	o	DEL

附录 6 参考材料

关于 TCP/IP 的详细情况，请参见 DDN 协议手册 (3 卷)。

出版商

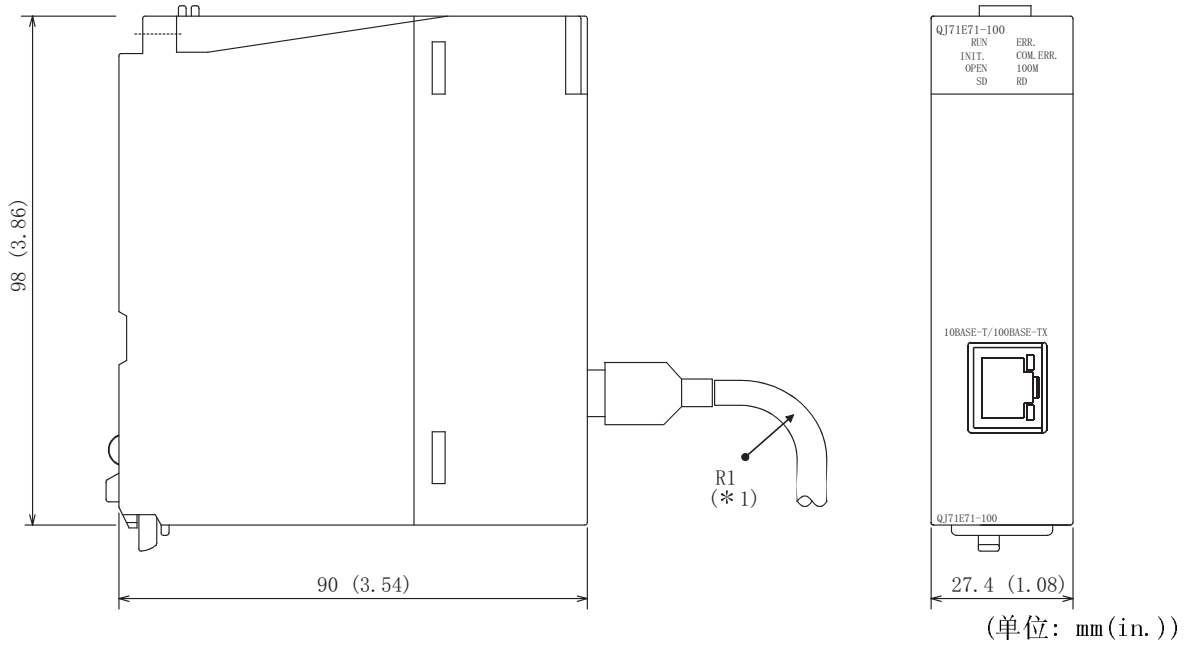
DDN 网络中心
SRI International
333 Ravenswood Avenue, EJ291
Menlo Park, 加利福尼亚 94025

RFC 编号

TCP RFC 793
UDP RFC 768
IP RFC 791
ICMP RFC 792
ARP RFC 826

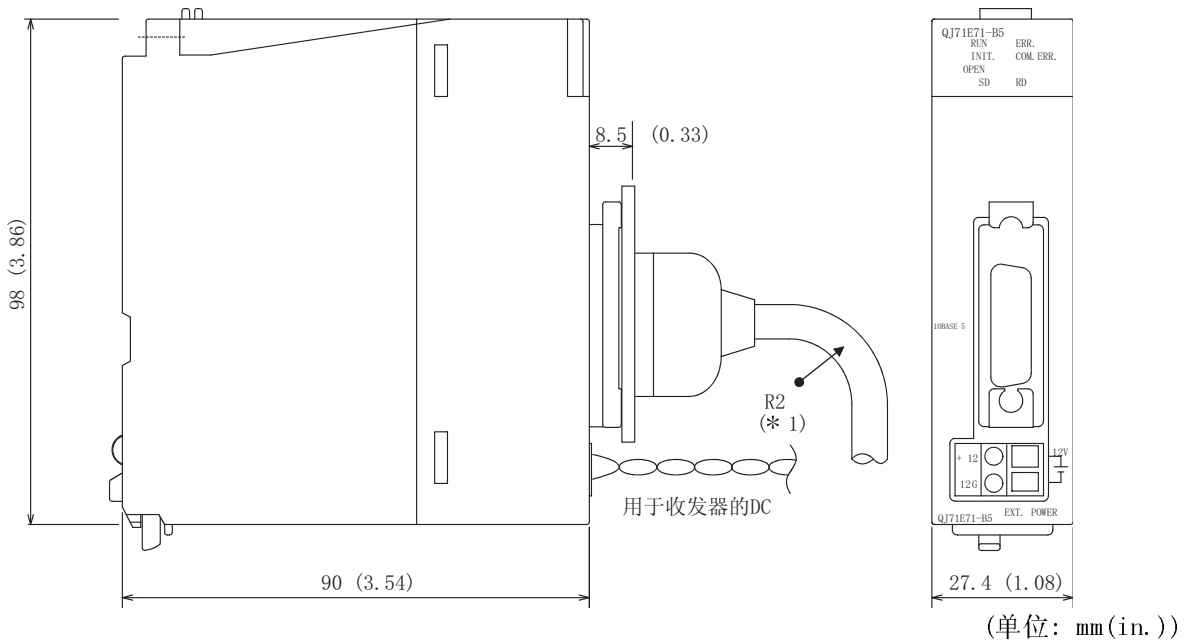
附录 7 外部尺寸

(1) QJ71E71-100



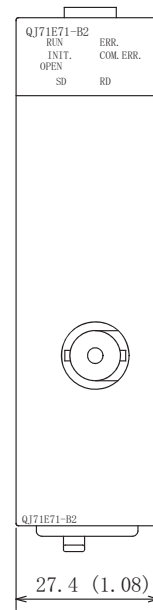
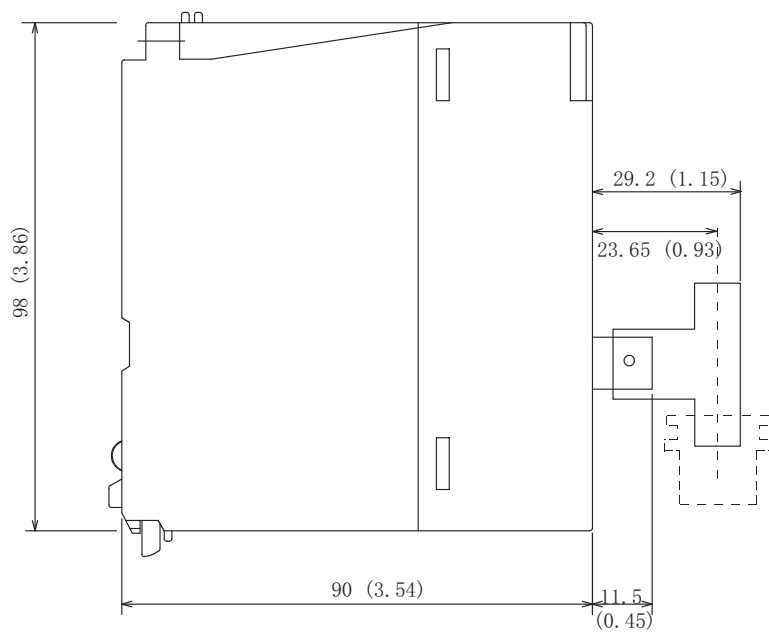
*1 当连接双绞线时，确保靠近接头的弯曲半径(参考值：R1)是电缆外径的 4 倍或更大。

(2) QJ71E71-B5



*1 连接 AUI 电缆时，设置接口旁的弯曲半径(参考值：R2)为电缆外部直径的 4 倍或更大。

(3) QJ71E71-B2



(单位: mm (in.))

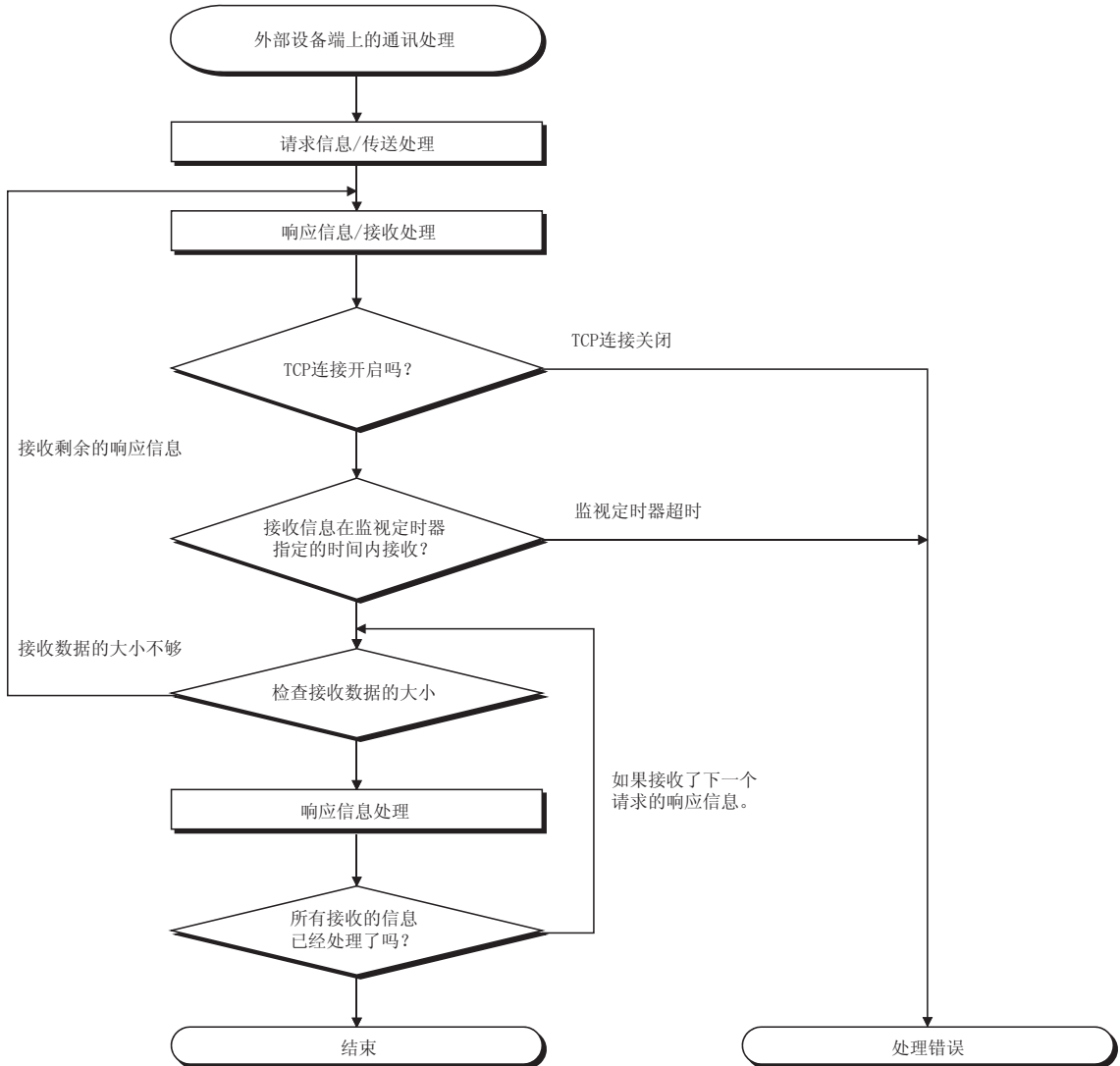
附录 8 程序范例

本节中介绍的程序范例为测试以太网模块和同在一个以太网上连接的外部设备(IBM PC/AT)之间的连接。

对于每一个程序来说, 执行通讯测试只需极少量的编程。按照系统配置修改 IP 地址、端口号和其它数值。另外, 可以添加异常处理。

(1) 目标设备的接收处理

以下显示目标设备端上的接收处理例子。



背景

对于以太网通讯, 在个人计算机内部使用 TCP 插件功能。但是, 这些功能没有任何限制。所以, 当执行一次“发送”功能来传送数据时, 接受结果(代码)需要执行一次或多次的“接收”功能来读取数据(“发送”和“接收”不是成 1:1 来执行)。由于此原因, 必须按照以上说明的接收步骤。

(2) 当传送目标不支持(1)节中显示的接收处理流程图时

如果传送目标不支持(1)节中显示的接收处理流程图时，设置允许与“TCP 最大段大小选项传送”进行通讯会出现以下结果。

- 使用MC协议从传送目标中执行批量读取时，数据读取出错。
- 用支持此功能的模块替换以太网模块(不支持TCP最大段大小选项传送功能)后，数据读取出错。
- 即使更改缓冲存储器中的“接收包数”区(地址:1B8H, 1B9H)，也无法接收数据。

如果发生这样的情况，设置“TCP 最大段大小选项传送”设置为禁止。

附录 8.1 使用 Visual Basic®.NET 和 Visual C++®.NET 的程序范例

附录 8.1.1 使用 MC 协议-1 通讯的范例

下面说明一示例程序，它的执行环境和数据通讯的内容。

(1) 程序范例的执行环境

(a) PLC CPU 端

- | | |
|------------------------|-----------------------------------|
| 1) 安装有以太网的站的 QCPU 型号名称 | : Q25HCPU |
| 2) 以太网模块 I/O 信号 | : X/Y000 至 X/Y01F |
| 3) 以太网模块 IP 地址 | : C0.00.01.FDH (192.00.01.253) |
| 4) 以太网模块端口号 | : 2000 _H |
| 5) GX Developer 设置 | |
| • 操作设置 | : 参阅下一页的“(3) GX Developer 设置 (a)” |
| • 开放设置 | : 参阅下一页的“(3) GX Developer 设置 (b)” |

(b) 外部设备端

- | | |
|---------------|---|
| 1) 操作环境 | : Microsoft® Windows® XP 操作系统 2002 版本服务包 2 |
| 2) 以太网接口板型号名称 | : WINSOCK 兼容板 |
| 3) 库 | : WSOCK32.LIB |
| 4) 软件开发环境 | : Microsoft® Corporation Visual C++® .NET 2003 版本 |
| 5) 以太网地址 | : 不需要设置，因为 ARP 功能可用。 |
| 6) IP 地址 | : 在主动开放时接收。 |
| 7) 端口号 | : 在主动开放时接收。 |

(c) 通讯协议

: TCP/IP

(2) 程序范例的概要

(a) PLC CPU 端的顺控程序

参数用 GX Developer 设置。
(无需顺控程序)

(b) 外部设备端的程序

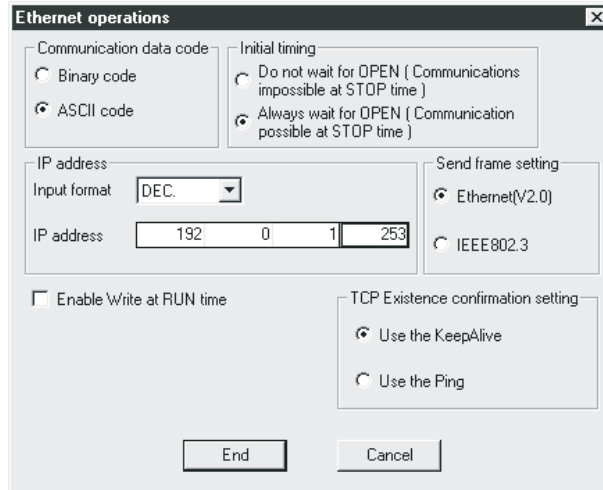
使用上述库与 PLC CPU 进行读/写数据通讯。

- 以字为单位写(从 D0 到 D4 共 5 个点)
- 以字为单位读(从 D0 到 D4 共 5 个点)

(3) GX Developer 设置

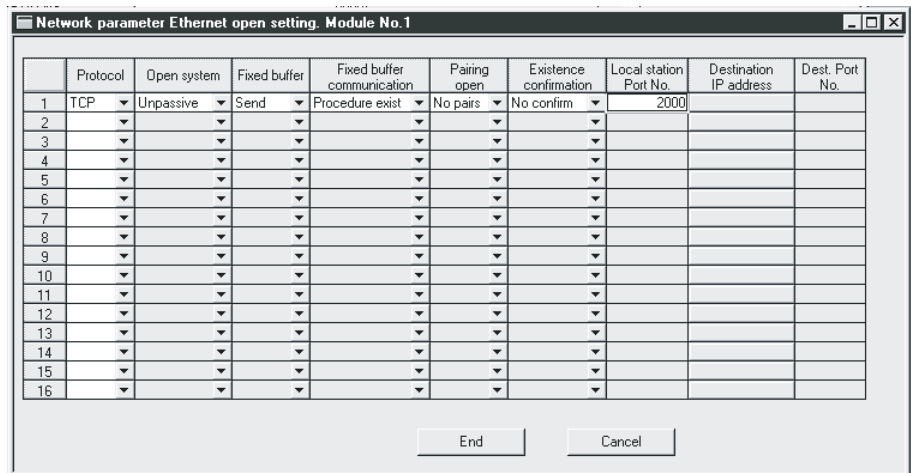
按如下设置 PLC CPU 参数。

(a) 操作设置



本地站 IP 地址：C0.00.01.FDH (192.00.01.253)

(b) 开放设置



本地站端口号：2000H

(4) 外部设备端的程序

下面所示的是访问安装有以太网模块的站中的 Q25H CPU 的外部设备的程序范例。执行程序时，下列通讯信息内容显示在可编程控制器中：

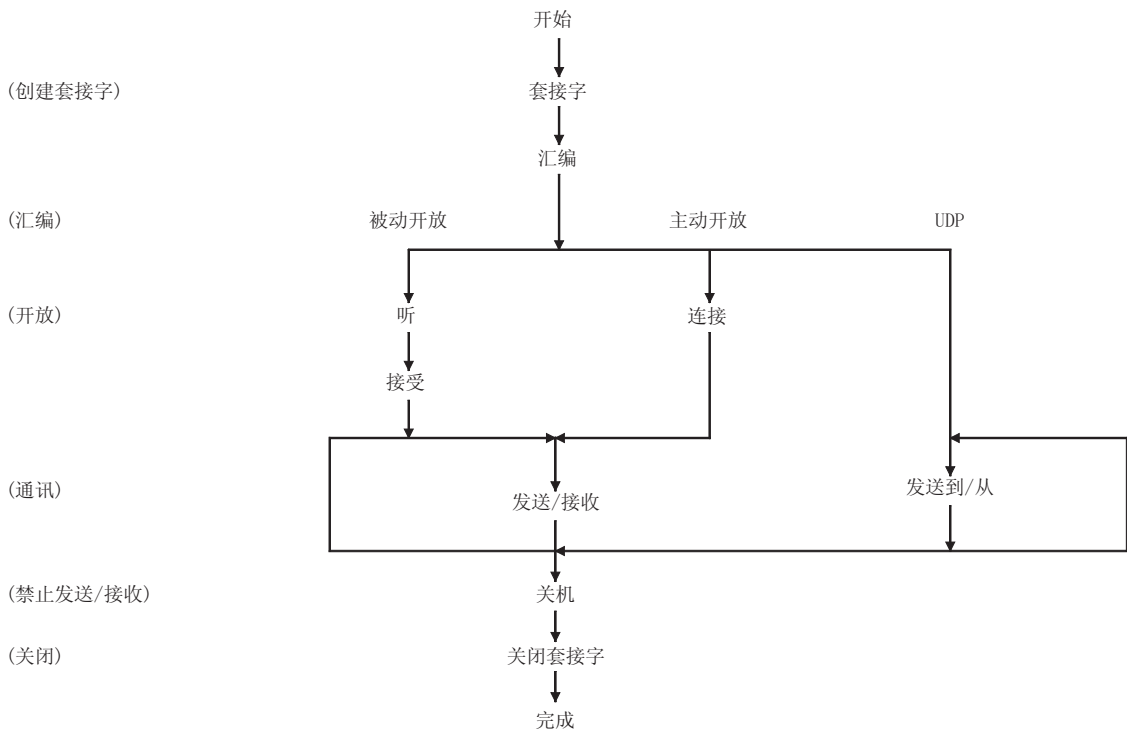
- 1) 以字为单位批量写入命令信息
- 2) 以字为单位批量写入响应信息
- 3) 以字为单位批量读命令信息
- 4) 以字为单位批量读响应信息

备注

(1) 下面大致说明使用 Microsoft® Corporation Visual C⁺⁺.NET. 创建一个程序的汇编步骤。

- 1) 启动 Visual C⁺⁺.NET.
- 2) 创建工程
选择 [File] → [New] → [Project] 菜单，在 “Project Types” 中选择 “.NET”，在 “Templates” 中选择 “Empty Project”，并设置工程名 (比如:AJSAMP) 和位置。
- 3) 创建源文件
显示解答浏览器，右击源文件并选择 [添加] → [添加新项目]。设置文件名 (比如:AJSAMP.cpp) 和保存位置，并根据程序范例创建程序。(请参见下一页)
- 4) 从程序设置画面中进行 WSOCK32.LIB 链接。
显示解答浏览器，右击程序名 (AJSAMP) 并选择 [属性] → [配置属性] → [连接器] → [指令链接]。在附加选项中选择 WSOCK32.LIB 并按 **OK** 键。
- 5) 在创建菜单中，点击创建方法来创建一个执行文件。(AJSAMP.EXE)
- 6) 结束 Visual C⁺⁺.NET.
- 7) 执行 AJSAMP.EXE.

(2) 调用套接字 (socket) 程序的步骤概要




```

/ *****/
/ ** ** /
/ ** 示例程序(程序名:AJSAMP.CPP) ** /
/ ** ** /
/ ** 该示例程序执行以太网模块和目标设备间的连接 ** /
/ ** 测试。 ** /
/ ** 该程序访问装有以太网模块的 PLC CPU 的数据寄 ** /
/ ** 存器(D)。 ** /
/ ** ** /
/ ** 版权(C) 2005 三菱电机 ** /
/ ** 保留所有版权 ** /
/ ** ** /
/ ** ** /
/ ** ** /
/ ** ** /
/ *****/

#include <stdio.h>
#include <winsock.h>

#define FLAG_OFF 0 // 完成标志 OFF
#define FLAG_ON 1 // 完成标志 ON
#define SOCK_OK 0 // 正常完成
#define SOCK_NG -1 // 异常完成
#define BUF_SIZE 4096 // 缓冲存储器规格

#define ERROR_INITIAL 0 // 初始化出错
#define ERROR_SOCKET 1 // 套接字创建出错
#define ERROR_BIND 2 // 汇编出错
#define ERROR_CONNECT 3 // 连接出错
#define ERROR_发送 4 // 发送出错
#define ERROR_RECEIVE 5 // 接收出错
#define ERROR_SHUTDOWN 6 // 关机出错
#define ERROR_CLOSE 7 // 线路关闭出错

//定义检查接收大小
//#define RECV_ANS_1 4// 回复软元件写入(1E 帧)的响应信息接收大小
#define RECV_ANS_1 22// 回复软元件写入(3E 帧)的响应信息接收大小
//#define RECV_ANS_2 24// 回复软元件读取(1E 帧)的响应信息接收大小
#define RECV_ANS_2 42// 回复软元件读取(3E 帧)的响应信息接收大小

typedef struct sck_inf{
    struct in_addr my_addr;
    unsigned short my_port;
    struct in_addr aj_addr;
    unsigned short aj_port;
}sck_inf;

```

```

int nErrorStatus; // 出错信息存储值
int Dmykeyin; // 虚拟键入
int Closeflag; // 连接完成标志
int socketno;

int main()
{
    WORD wVersionRequested=MAKEWORD(1, 1); // Winsock Ver 1.1 请求
    WSADATA wsaData;
    int length; // 通讯数据长度
    unsigned char s_buf[BUF_SIZE]; // 发送缓冲存储器
    unsigned char r_buf[BUF_SIZE]; // 接收缓冲存储器
    int rbuf_idx; // 接受数据存储器起始索引
    int recv_size; // 接受数据数
    struct sock_inf sc;
    struct sockaddr_in hostdata; // 外部设备端数据
    struct sockaddr_in aj71e71; // 以太网模块端数据
    void Sockerror(int); // 错误处理功能

    unsigned long ulCmdArg ; // 无块模式设置标志

    sc.my_addr.s_addr=htonl(INADDR_ANY); // 以太网设备端 IP 地址
    sc.my_port=htons(0); // 以太网设备端端口号
    sc.aj_addr.s_addr=inet_addr("192.0.1.253"); // 以太网模块端 IP 地址
    // (C00001FDH)
    sc.aj_port=htons(0x2000); // 以太网模块端口号

    Closeflag=FLAG_OFF; // 连接完成标志关

    nErrorStatus=WSAStartup(wVersionRequested, &wsaData); // Winsock 初始化处理

    if (nErrorStatus!=SOCK_OK) {
        Sockerror(ERROR_INITIAL); // 错误处理
        return(SOCK_NG);
    }

    printf("Winsock Version is %ld.%ld\n", HIBYTE(wsaData.wVersion), LOBYTE(wsaData.wVersion));
    printf("AJ_test Start\n");

    socketno=socket(AF_INET, SOCK_STREAM, 0); // 创建 TCP/IP 的套接字

    if (socketno==INVALID_SOCKET) {
        Sockerror(ERROR_SOCKET); // 错误处理
        return(SOCK_NG);
    }

    hostdata.sin_family=AF_INET;
    hostdata.sin_port=sc.my_port;
    hostdata.sin_addr.s_addr=sc.my_addr.s_addr;

```

```

if (bind(socketno, (LPSOCKADDR)&hostdata, sizeof(hostdata))!=SOCK_OK) {
    Sockerror(ERROR_BIND);           // 汇编
    return(SOCK_NG);                 // 错误处理
}

aj71e71.sin_family=AF_INET;
aj71e71.sin_port=sc.aj_port;
aj71e71.sin_addr.s_addr=sc.aj_addr.s_addr;

if (connect(socketno, (LPSOCKADDR)&aj71e71, sizeof(aj71e71))!=SOCK_OK) {
    Sockerror(ERROR_CONNECT);       // 连接(主动开放)
    return(SOCK_NG);                 // 错误处理
}

Closeflag=FLAG_ON;                  // 连接完成标志开

// 设置为无块模式
ulCmdArg = 1;
ioctlsocket(socketno, FIONBIO, &ulCmdArg);           // 设置为无块模式

// strcpy((char *) (s_buf), "03FF000A4420000000000500112233445566778899AA");
// D0 至 D4 批量写入请求(1E 帧)
strcpy((char *) (s_buf), "500000FF03FF00002C000A1401000D*0000000005112233445566778899AA");
// D0 至 D4 批量写入请求(3E 帧)

length = strlen((char *) (s_buf));

if (send(socketno, (char *) (s_buf), length, 0) == SOCKET_ERROR) {
    Sockerror(ERROR_send);           // 数据发送
    return (SOCK_NG);                 // 错误处理
}
printf("\n send data\n%s\n", s_buf);

// 同时执行接收大小检查和接收处理
rbuf_idx = 0;                         // 接收数据存储器起始索引初始化
recv_size = 0;                         // 初始化数据接收数
while(1) {
    length = recv(socketno, (char*) (&r_buf[rbuf_idx]), (BUF_SIZE - rbuf_idx), 0);
    // 响应数据接收
    if (length == 0) {                 // 连接是否断开?
        Sockerror(ERROR_RECIEVE);    // 错误处理
        return (SOCK_NG);
    }
}

```

```

    if (length == SOCKET_ERROR) {
        nErrorStatus = WSAGetLastError();
        if (nErrorStatus != WSAEWOULDBLOCK) {
            Sockerror(ERROR_RECEIVE); // 错误处理
            return (SOCK_NG);
        } else {
            continue; // 接收数据前重复
        }
    } else {
        rbuf_idx += length; // 更新接收数据存储器位置
        // 位置
        recv_size += length; // 更新接收数据数
        if (recv_size >= RECV_ANS_1) // 已经接收所有响应信息吗?
            // 接收?
            break; // 接收完信息后停止重复
            // 已接收
    }
}
r_buf[rbuf_idx] = '\0' ; // 在接收数据结束时设置 NULL

printf("\n receive data\n%s\n", r_buf);

// strcpy((char *) (s_buf), "01FF000A4420000000000500"); // D0 至 D4 批量读取请求
// (1E 帧)
strcpy((char *) (s_buf), "500000FF03FF000018000A0401000D*000000005");
// D0 至 D4 批量读取请求
// (3E 帧)

length = strlen((char *) (s_buf));

if (send(socketno, (char *) (s_buf), length, 0) == SOCKET_ERROR) {
    // 数据发送
    Sockerror(ERROR_SEND); // 错误处理
    return (SOCK_NG);
}
printf("\n send data\n%s\n", s_buf);

// 同时执行接收大小检查和接收处理
rbuf_idx = 0; // 接收数据存储器起始索引
// 初始化
recv_size = 0; // 初始化数据接收数
while(1) {
    length = recv(socketno, (char*) (&r_buf[rbuf_idx]), (BUF_SIZE - rbuf_idx), 0);
    // 响应数据接收
    if (length == 0) { // 连接是否断开?
        Sockerror(ERROR_RECEIVE); // 错误处理
        return (SOCK_NG);
    }
}

```

```

    if (length == SOCKET_ERROR) {
        nErrorStatus = WSAGetLastError();
        if (nErrorStatus != WSAEWOULDBLOCK) {
            Sockerror(ERROR_RECIEVE); // 错误处理
            return (SOCK_NG);
        } else {
            continue; // 接收数据前重复
        }
    } else {
        rbuf_idx += length; // 更新接收数据存储位置
        recv_size += length; // 更新接收数据数
        if (recv_size >= RECV_ANS_2) // 已经接受所有响应信息吗?
            break; // 接收完信息后停止重复
    }
}
r_buf[rbuf_idx] = '\0' ; // 在接收数据结束时设置 NULL

printf("\receive data\n%s\n", r_buf);

if (shutdown(socketno, 2) !=SOCK_OK) { // 禁止发送/接收的处理
    Sockerror(ERROR_SHUTDOWN); // 发送/接收
    return (SOCK_NG); // 错误处理
}

if (closesocket(socketno) !=SOCK_OK) { // 关闭处理
    Sockerror(ERROR_CLOSE); // 错误处理
    return (SOCK_NG);
}

Closeflag=FLAG_OFF; // 连接完成标志关
WSACleanup(); // 释放 Winsock.DLL

printf("\nAJ_test End.\n\n Normally completed. \n");
printf("Press any key 至 exit the program.\n");
Dmykeyin=getchar(); // 等待键入

return (SOCK_OK);
}

void Sockerror(int error_kind) // 错误处理功能
{
    if (error_kind==ERROR_INITIAL) {
        printf("Initial processing is abnormal.");
    }
}

```

```
else{
    nErrorStatus=WSAGetLastError();
    switch(error_kind){
    case ERROR_SOCKET:
        printf("Failed to create socket.");
        break;
    case ERROR_BIND:
        printf("Failed to bind.");
        break;
    case ERROR_CONNECT:
        printf("Failed to establish connection.");
        break;
    case ERROR_SEND:
        printf("Sending failed.");
        break;
    case ERROR_RECEIVE:
        printf("Receiving failed.");
        break;
    case ERROR_SHUTDOWN:
        printf("Failed to shutdown.");
        break;
    case ERROR_CLOSE:
        printf("Failed to close normally.");
        break;
    }
}

printf("Error code is %d.\n", nErrorStatus);

if(Closeflag==FLAG_ON){
    nErrorStatus=shutdown(socketno, 2); // 关机处理
    nErrorStatus=closesocket(socketno); // 关闭处理
    Closeflag=FLAG_OFF; // 连接完成标志关
}

printf("Press any key to exit the program.\n");
Dmykeyin=getchar(); // 等待键入
WSACleanup(); // 释放 Winsock. DLL
return;
}
```

附录 8.1.2 使用 MC 协议-2 通讯的程序范例

本节说明了程序范例，执行环境和数据执行内容。

(1) 程序范例的执行环境

(a) PLC CPU 端

- 1) 安装站 QCPU 的以太网模块型号名 : Q25PRHCPU
- 2) 以太网模块的 I/O 信号 : X/Y000 至 X/Y01F
- 3) 以太网模块 IP 地址(系统 A) : C0.00.01.FC_H
(192.00.01.252)
- 以太网模块 IP 地址(系统 B) : C0.00.01.FD_H
(192.00.01.253)
- 4) 以太网模块端口编号 : 2000_H
- 5) GX Developer 设置
 - 操作设置:参见下一页上的“(3)GX Developer 设置(a)”
 - 开放设置:参见下一页上的“(3)GX Developer 设置(b)”
 - 冗余设置:参见下一页上的“(3)GX Developer 设置(c)”

(b) 外部设备端

- 1) 操作环境:
Microsoft® Windows® XP Professional 操作系统版本 2002 服务包 2。
- 2) 以太网接口板型号名 : WINSOCK 兼容板
- 3) 库 : WSOCK32.LIB
- 4) 软件开发环境:
使用微软公司制造的 Microsoft® Visual C++®.NET 2003。
- 5) 以太网地址 : 可以不设置为 APP 功能。
- 6) IP 地址 : 在主动开放时接收。
- 7) 端口编号 : 在主动开放时接收。

(c) 通讯系统 : TCP/IP

(2) 程序范例概述

(a) PLC CPU 端顺控程序

必须使用 GX Developer 进行参数设置。
(无需顺控程序)

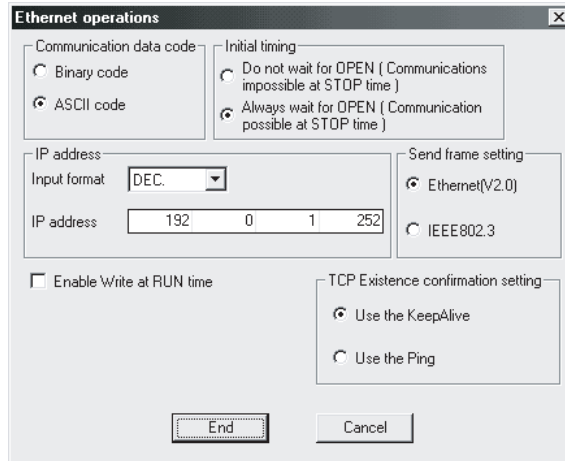
(b) 外部设备端程序

使用以上库按照字单位写入数据到冗余 CPU(控制系统)的数据寄存器 D0 到 D4(5 点)。
如果此时由于通讯出错或相关原因无法从系统 A 端写入数据，数据则从系统 B 端写入到冗余系统(控制系统)的数据寄存器中。

(3) GX Developer 设置

按以下说明设置 PLC CPU 参数。

(a) 操作设置



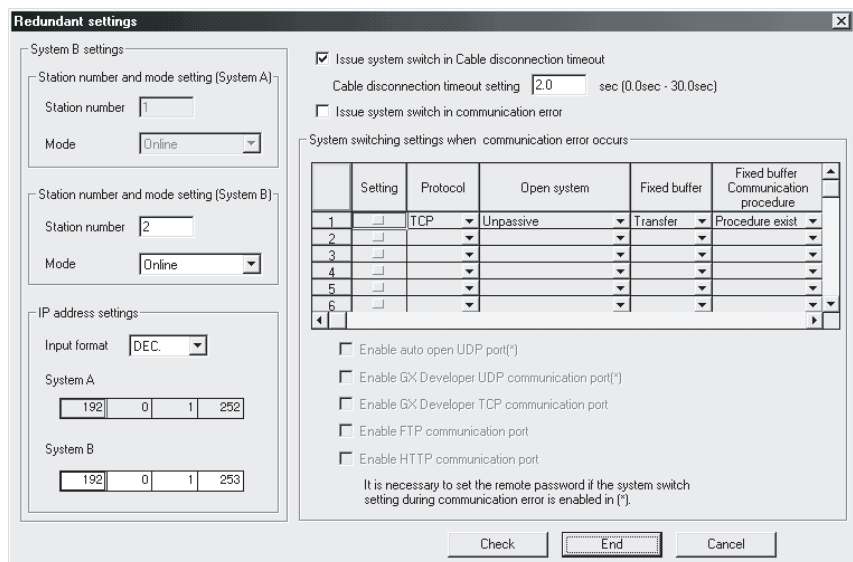
以太网模块 IP 地址(系统 A) : C0.00.01.FC_H (192.00.01.252)

(b) 开放设置

	Protocol	Open system	Fixed buffer	Fixed buffer communication procedure	Pairing open	Existence confirmation	Host station Port No.	Transmission Target device IP address	Transmission target device Port No.
1	TCP	Unpassive	Send	Procedure exist	Disable	No confirm	2000		
2									
3									

本地端口号 : 2000_H

(c) 冗余设置



以太网模块 IP 地址(系统 B) : C0.00.01.FD_H (192.00.01.253)

(4) 外部设备端程序

以下显示访问安装以太网模块站 Q25PRHCPU 的外部设备程序范例。
执行程序时，按照预定顺序显示以下内容。

- 1) 使用 Winsock 的版本
- 2) 测试启动信息。
- 3) 以字单位写入批量指令信息。
- 4) 以字单位写入批量响应信息。
- 5) 测试结束信息

备注

以下说明使用 Microsoft® Visual C++®.NET 创建程序的一般编辑步骤。

- 1) 启动 Visual C++®.NET。
- 2) 创建工程。
选择 [File] → [New] → [Project] 菜单，在 “Project Types” 中选择 “.NET”，在 “Templates” 中选择 “Empty Project”，并设置工程名 (比如:QJSAMP) 和位置。
- 3) 创建源文件。
显示 Solution Explorer，右击源文件并选择 [添加] → [添加新项目]。设置文件名 (比如:QJSAMP.cpp) 和保存位置，并根据程序范例创建程序。(请参见下一页)
- 4) 从程序设置画面中进行 WSOCK32.LIB 链接。
显示 Solution Explorer，右击程序名 (AJSAMP) 并选择 [属性] → [配置属性] → [连接器] → [指令链接]。在附加选项中选择 WSOCK32.LIB 并按 键。
- 5) 在创建菜单中，点击创建方法来创建一个执行文件。(QJSAMP.EXE)
- 6) 退出 Visual C++®.NET。
- 7) 执行 QJSAMP.EXE。

```

/ *****/
/ ** ** /
/ ** 示例程序(程序名:QJSAMP.CPP) ** /
/ ** ** /
/ ** 该示例程序执行以太网模块和外部设备间的连接 ** /
/ ** 测试。 ** /
/ ** 该程序访问装有以太网模块的冗余 CPU(控制程序) ** /
/ ** 的数据寄存器(D)。 ** /
/ ** ** /
/ ** 版权(C) 2005 三菱电机 ** /
/ ** 保留所有版权 ** /
/ ** ** /
/ ** ** /
/ ** ** /
/ *****/

#include <stdio.h>
#include <winsock.h>

#define FLAG_OFF 0 // 结束标志 OFF
#define FLAG_ON 1 // 结束标志 ON
#define SOCK_OK 0 // 正常结束
#define SOCK_NG -1 // 未结束
#define BUF_SIZE 4096 // 接收缓冲器大小

#define ERROR_NO_ERROR 0 // 无出错
#define ERROR_INITIAL 1 // 初始化错误
#define ERROR_SOCKET 2 // 接口创建错误
#define ERROR_BIND 3 // 汇编错误
#define ERROR_CONNECT 4 // 连接错误
#define ERROR_SEND 5 // 发送错误
#define ERROR_SHUTDOWN 6 // 关闭错误
#define ERROR_CLOSE 7 // 线路关闭错误

//检查接收大小的定义
#define RECV_ANS_1 22 // 答复设备写入的响应信息接收大小(3E 帧)

typedef struct sck_inf {
    struct in_addr my_addr;
    unsigned short my_port;
    struct in_addr qj_addr;
    unsigned short qj_port;
} sck_inf;

int nErrorStatus; // 出错信息存储变量
int Dmykeyin; // 虚拟键输入
int ShutdownflagA; // 关闭标志(用于系统 A 连接)
int ShutdownflagB; // 关闭标志(用于系统 B 连接)

```

```

int CloseflagA; // 连接结束标志 (用于系统 A 连接)
int CloseflagB; // 连接结束标志 (用于系统 B 连接)
int socketnoA;
int socketnoB;
int ConnectLastErrorA; //连接处理出错信息(用于系统 A 连接)
int ConnectLastErrorB; //连接处理出错信息(用于系统 B 连接)
int SendFlag; // 发送完成标志

int main()
{
    WORD wVersionRequested = MAKEWORD(1, 1); // Winsock 版本 1.1 请求
    WSADATA wsaData;
    int length; // 通讯数据长度
    unsigned char s_buf[BUF_SIZE]; // 发送缓冲器
    unsigned char r_bufA[BUF_SIZE], r_bufB[BUF_SIZE]; // 接收缓冲器
    struct sock_inf scA, scB;
    struct sockaddr_in hostdataA, hostdataB; // 外部设备端数据
    struct sockaddr_in qj71e71A, qj71e71B; // 以太网模块端数据
    BOOL DataRecv(int, unsigned char *, int); // 接收处理功能
    void Sockerror(int, int); // 错误处理功能

    unsigned long ulCmdArgA, ulCmdArgB; // 无块模式设置标志

    scA.my_addr.s_addr = scB.my_addr.s_addr = htonl(INADDR_ANY); // 外部设备端 IP 地址
    scA.my_port = scB.my_port = htons(0); // 外部设备端端口号
    scA.qj_addr.s_addr = inet_addr("192.0.1.252"); // 以太网模块端 IP 地址(系统 A: C00001FCh)
    scB.qj_addr.s_addr = inet_addr("192.0.1.253"); // 以太网模块端 IP 地址(系统 B: C00001FDh)
    scA.qj_port = scB.qj_port = htons(0x2000); // 外部设备端端口号

    ShutdownflagA = ShutdownflagB = FLAG_OFF; // 关闭标志 OFF
    CloseflagA = CloseflagB = FLAG_OFF; // 连接结束标志 OFF

    nErrorStatus = WSASStartup(wVersionRequested, &wsaData); // Winsock 初始化处理

    ConnectLastErrorA = ERROR_NO_ERROR; // 连接处理出错信息初始化(用于系统 A)
    ConnectLastErrorB = ERROR_NO_ERROR; // 连接处理出错信息初始化(用于系统 B)

    if(nErrorStatus != SOCK_OK) {
        Sockerror(ERROR_INITIAL, ERROR_INITIAL); // 错误处理
        return (SOCK_NG);
    }

    printf("Winsock Version is %ld.%ld\n", HIBYTE(wsaData.wVersion), LOBYTE(wsaData.wVersion));
    printf("QJ_test Start\n");
}

```

```

// 系统 A 连接处理
socketnoA = socket(AF_INET, SOCK_STREAM, 0); // TCP/IP 接口 (用于系统 A 连接) 创建
if(socketnoA != INVALID_SOCKET) {
    hostdataA.sin_family    = AF_INET;
    hostdataA.sin_port      = scA.my_port;
    hostdataA.sin_addr.s_addr= scA.my_addr.s_addr;

    if(bind(socketnoA, (LPSOCKADDR)&hostdataA, sizeof(hostdataA)) == SOCK_OK) {
                                                // 连接(系统 A)

        qj71e71A.sin_family    = AF_INET;
        qj71e71A.sin_port      = scA.qj_port;
        qj71e71A.sin_addr.s_addr= scA.qj_addr.s_addr;

        if(connect(socketnoA, (LPSOCKADDR)&qj71e71A, sizeof(qj71e71A)) == SOCK_OK) {
                                                    // 汇编(主动开放:系统 A)
            ShutdownflagA = FLAG_ON;           // 关闭标志 ON
            CloseflagA = FLAG_ON;             // 连接结束标志 ON
            // 设置为无块模式
            ulCmdArgA = 1;
            ioctlsocket(socketnoA, FIONBIO, &ulCmdArgA);
                                                    // 设置为无块模式(用于系统 A 连接)
        } else {
            ConnectLastErrorA = ERROR_CONNECT; // 建立连接失败
        }
    } else {
        ConnectLastErrorA = ERROR_BIND;       // 连接失败
    }
} else {
    ConnectLastErrorA = ERROR_SOCKET;        // 创建接口失败
}

// 系统 B 连接处理
socketnoB = socket(AF_INET, SOCK_STREAM, 0); // TCP/IP 接口 (用于系统 B 连接) 创建
if(socketnoB != INVALID_SOCKET) {
    hostdataB.sin_family    = AF_INET;
    hostdataB.sin_port      = scB.my_port;
    hostdataB.sin_addr.s_addr= scB.my_addr.s_addr;

    if(bind(socketnoB, (LPSOCKADDR)&hostdataB, sizeof(hostdataB)) == SOCK_OK) {
                                                // 连接(系统 A)

        qj71e71B.sin_family    = AF_INET;
        qj71e71B.sin_port      = scB.qj_port;
        qj71e71B.sin_addr.s_addr= scB.qj_addr.s_addr;
    }
}

```

```

        if(connect(socketnoB, (LPSOCKADDR)&qj71e71B, sizeof(qj71e71B)) == SOCK_OK) {
                                                    // 连接(主动开放: 系统 B)
            ShutdownflagB = FLAG_ON;                // 关闭标志 ON
            CloseflagB = FLAG_ON;                    // 连接结束标志 ON
            // 设置为无块模式
            uICmdArgB = 1;
            ioctlsocket(socketnoB, FIONBIO, &uICmdArgB);
                                                    // 设置为无块模式(用于 B 连接)
        } else {
            ConnectLastErrorB = ERROR_CONNECT;        // 建立连接失败
        }
    } else {
        ConnectLastErrorB = ERROR_BIND;              // 连接失败
    }
} else {
    ConnectLastErrorB = ERROR_SOCKET;                // 创建接口失败
}

// 连接完成处理
if( (CloseflagA == FLAG_OFF) && (CloseflagB == FLAG_OFF) ){// 两个系统都为异常
    Sockerror(ConnectLastErrorA, ConnectLastErrorB); // 错误处理
    return (SOCK_NG);
}

strcpy((char*)(s_buf), "500000FF03D000002C000A1401000D*0000000005112233445566778899AA");
                                                    // D0-D4 批量写入请求 (3E 帧, 发送到控制系统)
length = strlen((char*)(s_buf));

printf("Send starts. Press any key. \n");
Dmykeyin = getchar();                            // 等待键输入

SendFlag = FLAG_OFF;                              // 发送完成标志 OFF
// 系统 A 发送处理
if( CloseflagA == FLAG_ON && (SendFlag == FLAG_OFF) ){
    if(send(socketnoA, (char*)(s_buf), length, 0) != SOCKET_ERROR) {
                                                    // 数据发送(系统 A)
        printf("\n Send data (System A) \n%s\n", s_buf); // 发送数据显示(系统 A)
        SendFlag = FLAG_ON;                            // 发送完成标志 ON
        // Receive processing
        if(DataRecv(socketnoA, r_bufA, RECV_ANS_1) == TRUE) { // 接收处理
            printf("\n Receive data (System A) \n%s\n", r_bufA); // 数据接收
        } else {
            printf("Receive failure (System A) \n");
        }
    } else {
        printf("Send failure (System A) \n");
    }
}
}
}

```

```
// 系统 B 发送处理
if( (CloseflagB == FLAG_ON) && (SendFlag == FLAG_OFF) ){
    if(send(socketnoB, (char*)(s_buf), length, 0) != SOCKET_ERROR) { // 数据发送(系统 B)
        printf("\n Send data (System B) \n%s\n", s_buf); // 发送数据显示(系统 B)
        SendFlag = FLAG_ON; // 发送完成标志 ON
        // Receive processing
        if(DataRecv(socketnoB, r_bufB, RECV_ANS_1) == TRUE) { // 接收处理
            printf("\n Receive data (System B) \n%s\n", r_bufB); // 数据接收
        } else {
            printf("Receive failure (System B) \n");
        }
    }else{
        printf("Send failure (System B) \n");
    }
}

// 发送完成处理
if( SendFlag == FLAG_OFF ){
    Sockerror(ERROR_SEND, ERROR_SEND); // 错误处理
    return (SOCK_NG);
}

if(CloseflagA == FLAG_ON) {
    ShutdownflagA = FLAG_OFF; // 关闭标志 OFF
    if(shutdown(socketnoA, 2) != SOCK_OK) { // 发送/接收禁止处理(系统 A)
        Sockerror(ERROR_SHUTDOWN, ERROR_NO_ERROR); // 错误处理
        return (SOCK_NG);
    }
}

if(CloseflagB == FLAG_ON) {
    ShutdownflagB = FLAG_OFF; // 关闭标志 OFF
    if(shutdown(socketnoB, 2) != SOCK_OK) { // 发送/接收禁止处理(系统 B)
        Sockerror(ERROR_NO_ERROR, ERROR_SHUTDOWN); // 错误处理
        return (SOCK_NG);
    }
}

CloseflagA = FLAG_OFF; // 连接结束标志 OFF
if(closesocket(socketnoA) != SOCK_OK) { // 关闭处理(系统 A)
    Sockerror(ERROR_CLOSE, ERROR_NO_ERROR); // 错误处理
    return (SOCK_NG);
}
```

```

CloseflagB = FLAG_OFF; // 连接结束标志 OFF
if(closesocket(socketnoB) != SOCK_OK) { // 结束处理(系统 B)
    Sockerror(ERROR_NO_ERROR, ERROR_CLOSE); // 错误处理
    return (SOCK_NG);
}

WSACleanup(); // Winsock.DLL 释放

printf("\nQJ_test End. \n\nNormally completed. \n");
printf("Program is closed. Press any key. \n");
Dmykeyin = getchar(); // 等待键输入

return (SOCK_OK);
}

BOOL DataRecv(int socketno, unsigned char *pR_buf, int size_max) // 接收处理功能
{
    int length; // 通讯数据长度
    int rbuf_idx; // 接收数据存储启动索引
    int recv_size; // 接收数据数

    // 同时进行大小检查时进行接收处理
    rbuf_idx = 0; // 接收启动索引初始化的数据存储
    recv_size = 0; // 初始化接收数据量
    while(1) {
        length = recv(socketno, ((char*)(pR_buf + rbuf_idx)), (BUF_SIZE - rbuf_idx), 0); // 响应数据接收
        // 连接中断了吗?
        if(length == 0) { // 错误处理
            return (FALSE);
        }
        if(length == SOCKET_ERROR) {
            nErrorStatus = WSAGetLastError();
            if(nErrorStatus != WSAEWOULDBLOCK) { // 错误处理
                return (FALSE);
            } else {
                continue; // 接收数据前一直重复
            }
        } else {
            rbuf_idx += length; // 更新接收数据存储位置
            recv_size += length; // 更新接收数据量
            if(recv_size >= size_max) // 接收到所有响应信息吗?
                break; // 接收数据时停止重复
        }
    }
    *(pR_buf + rbuf_idx) = '\0'; // 接收数据末尾
    // 设置 NULL
}

```

```
    return (TRUE);                // 正常完成
}

void Sockerror(int error_kind_A, int error_kind_B)    // 错误处理功能
{
    if (error_kind_A == ERROR_INITIAL) {
        printf("Initial processing is abnormal. \n");
    }
    else{
        nErrorStatus = WSAGetLastError();
        switch(error_kind_A) {
            case ERROR_SOCKET:
                printf("Socket could not be created. (System A)\n");
                break;
            case ERROR_BIND:
                printf("Bind could not be executed. (System A)\n");
                break;
            case ERROR_CONNECT:
                printf("Connection could not be established. (System A)\n");
                break;
            case ERROR_SEND:
                printf("Send could not be executed. \n");
                break;
            case ERROR_SHUTDOWN:
                printf("Shutdown could not be executed. (System A)\n");
                break;
            case ERROR_CLOSE:
                printf("Normal close could not be executed. (System A)\n");
                break;
        }
        switch(error_kind_B) {
            case ERROR_SOCKET:
                printf("Socket could not be created. (System B)\n");
                break;
            case ERROR_BIND:
                printf("Bind could not be executed. (System B)\n");
                break;
            case ERROR_CONNECT:
                printf("Connection could not be established. (System B)\n");
                break;
            case ERROR_SHUTDOWN:
                printf("Shutdown could not be executed. (System B)\n");
                break;
            case ERROR_CLOSE:
                printf("Normal close could not be executed. (System B)\n");
                break;
        }
    }
}
```



```
    }
}
printf("Error code is %d. \n", nErrorStatus);

if (ShutdownflagA == FLAG_ON) {
    nErrorStatus = shutdown(socketnoA, 2);           // 关闭处理(系统 A)
    ShutdownflagA = FLAG_OFF;                       // 关闭标志 OFF(系统 A)
}
if (ShutdownflagB == FLAG_ON) {
    nErrorStatus = shutdown(socketnoB, 2);           // 关闭处理(系统 B)
    ShutdownflagB = FLAG_OFF;                       // 关闭标志 OFF(系统 B)
}

if (CloseflagA == FLAG_ON) {
    nErrorStatus = closesocket(socketnoA);           // 关闭处理(系统 A)
    CloseflagA = FLAG_OFF;                          // 连接结束标志 OFF(系统 A)
}
if (CloseflagB == FLAG_ON) {
    nErrorStatus = closesocket(socketnoB);           // 结束处理(系统 B)
    CloseflagB = FLAG_OFF;                          // 连接结束标志 OFF(系统 B)
}

printf("Program is closed. Press any key. \n");
Dmykeyin = getchar();                               // 等待键输入
WSACleanup();                                       // Winsock.DLL 释放
return;
}
```

附录 8.1.3 使用 MC 协议-3 通讯的程序范例

本节说明了从 PLC CPU 中读取数据的外部设备程序的范例。
以下显示一个范例程序、其执行环境和数据通讯的内容。

(1) 程序范例的执行环境

(a) PLC CPU 端

- 1) 安装站 QCPU 的以太网模块型号名 : Q25HCPU
- 2) 以太网模块的 I/O 信号 : X/Y000 至 X/Y01F
- 3) 以太网模块 IP 地址 : C0.00.01.FD_H
(192.00.01.253)
- 4) 以太网模块端口编号 : 2000_H
- 5) GX Developer 设置
 - 操作设置:参阅下一页上的“(3)GX Developer 设置(a)”
 - 开放设置:参阅下一页上的“(3)GX Developer 设置(b)”

(b) 外部设备端

- 1) 操作环境:
Microsoft® Windows® XP Professional 操作系统版本 2002 服务包 2。
- 2) 以太网接口板型号名 : WINSOCK 兼容板
- 3) 软件开发环境:
使用微软公司制造的 Visual Basic® .NET 2003
- 4) 以太网地址 : 不必设置为 APP 功能。
- 5) IP 地址 : 分配任何给定序号。
- 6) 端口编号 : 分配任何给定序号。

(c) 通讯系统 : TCP/IP

(2) 程序范例概述

(a) PLC CPU 端顺控程序

必须使用 GX Developer 进行参数设置。
(无需顺控程序)

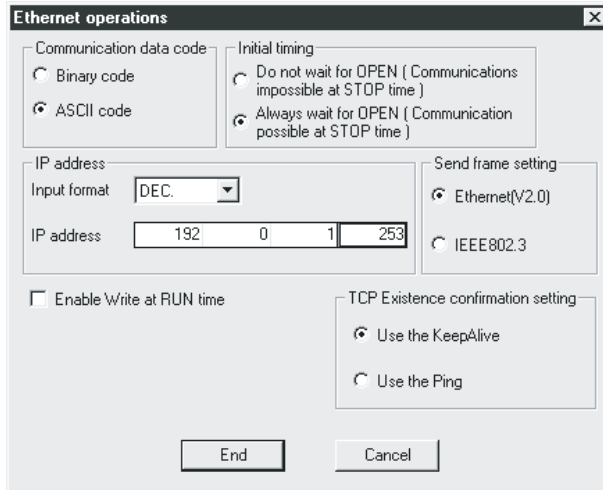
(b) 外部设备端程序

读取 PLC CPU 中的数据 (D0 到 D4)。

(3) GX Developer 设置

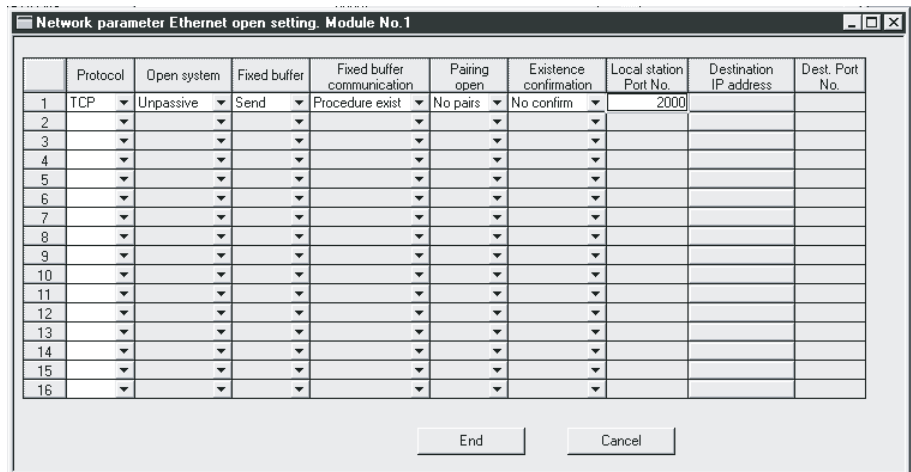
按以下说明设置 PLC CPU 参数。

(a) 操作设置



本站 IP 地址：C0.00.01.FD_H(192.00.01.253)

(b) 开放设置



本地端口号：2000_H

(4) 外部设备端程序

以下显示访问安装以太网模块站上 Q25PRHCPU 的外部设备的程序范例。

在此程序中，安装站的以太网模块上 QCPU 中的数据 D0 到 D4(5 点)用 A 兼容 1E 帧指令读取。(01:以字单位批量读取)

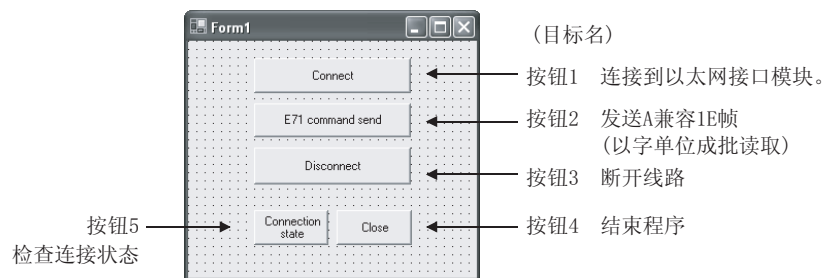
以下是基本操作步骤：

- 线路断开时发送指令。
连接线路，并在完成连接后发送 E71 指令。
- 线路连接时重新连接。
断开线路，并在断开后重新连接。

备注

- 按以下说明使用 Microsoft® Corporation's Visual Basic®.NET 创建程序:
 - 1) 启动 Visual Basic®.NET。
 - 2) 创建工程。
选择 [File] → [New] → [Project] 菜单，在 “Project Types” 中选择 “Visual Basic Project”，在 “Templates” 中选择 “Windows Application”，并设置工程名 (比如: AJSAMP) 和位置。
 - 3) 创建表格和程序。
使用工具箱中的 “按钮” 来创建 (5) 中显示的画面范例 (表 1)，并根据 (6) 中给出的程序范例创建程序。
 - 4) 在创建菜单上，点击创建方法来创建一个执行文件 (AJSAMP.EXE)。
 - 5) 退出 Visual Basic®.NET。
 - 6) 执行 AJSAMP.EXE。

(5) 窗口范例 (Form 1. vb)



(6) 示例程序 (Form 1. vb)

```

Option Strict Off
Option Explicit ON

Imports System
Imports System.Text
Imports System.Net

Friend Class Form1
    Inherits System.Windows.Forms.Form
    #Region "Windows Form Designer generated code"
    Public Sub New()
        MyBase.New()
        If m_vb6FormDefInstance Is Nothing Then
            If m_InitializingDefInstance Then
                m_vb6FormDefInstance = Me
            Else
                Try
                    'For the start-up form, the first instance created is the default instance.
                    If System.Reflection.Assembly.GetExecutingAssembly().EntryPoint
                        .DeclaringType Is Me.GetType Then
                        m_vb6FormDefInstance = Me
                    EndIf
                Catch
                End Try
            End If
        End If
        ' This call is required by the Windows form designer.
        InitializeComponent()
    End Sub

```

```

'Form overrides dispose to clean up the component list.
Protected Overloads Overrides Sub Dispose(ByVal Disposing As Boolean)
    If Disposing Then
        If Not components Is Nothing Then
            components.Dispose()
        End If
    End If
    MyBase.Dispose(Disposing)
End Sub
'Required by the Windows Form Designer.
Private components As System.ComponentModel.IContainer
Public WithEvents Command5 As System.Windows.Forms.Button
Public WithEvents Command4 As System.Windows.Forms.Button
Public WithEvents Command3 As System.Windows.Forms.Button
Public WithEvents Command2 As System.Windows.Forms.Button
Public WithEvents Command1 As System.Windows.Forms.Button
Dim Ajsock As Sockets.Socket
Private State As Boolean = False

```

'NOTE: The following procedure is required by the Windows Form Designer.

'It can be modified using the Windows Form Designer.

'Do not modify it using the code editor.

```

<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
    Me.Command5 = New System.Windows.Forms.Button
    Me.Command4 = New System.Windows.Forms.Button
    Me.Command3 = New System.Windows.Forms.Button
    Me.Command2 = New System.Windows.Forms.Button
    Me.Command1 = New System.Windows.Forms.Button
    Me.SuspendLayout()
    ,
' Command5
,
    Me.Command5.BackColor = System.Drawing.SystemColors.Control
    Me.Command5.Cursor = System.Windows.Forms.Cursors.Default
    Me.Command5.ForeColor = System.Drawing.SystemColors.ControlText
    Me.Command5.Location = New System.Drawing.Point(64, 152)
    Me.Command5.Name = "Command5"
    Me.Command5.RightToLeft = System.Windows.Forms.RightToLeft.No
    Me.Command5.Size = New System.Drawing.Size(72, 32)
    Me.Command5.TabIndex = 4
    Me.Command5.Text = "Connection status"
    ,
' Command4
,

```

```
Me.Command4.BackColor = System.Drawing.SystemColors.Control
Me.Command4.Cursor = System.Windows.Forms.Cursors.Default
Me.Command4.ForeColor = System.Drawing.SystemColors.ControlText
Me.Command4.Location = New System.Drawing.Point(144, 152)
Me.Command4.Name = "Command4"
Me.Command4.RightToLeft = System.Windows.Forms.RightToLeft.No
Me.Command4.Size = New System.Drawing.Size(73, 32)
Me.Command4.TabIndex = 3
Me.Command4.Text = "Close"
,
' Command3
,

Me.Command3.BackColor = System.Drawing.SystemColors.Control
Me.Command3.Cursor = System.Windows.Forms.Cursors.Default

Me.Command3.ForeColor = System.Drawing.SystemColors.ControlText
Me.Command3.Location = New System.Drawing.Point(64, 96)
Me.Command3.Name = "Command3"
Me.Command3.RightToLeft = System.Windows.Forms.RightToLeft.No
Me.Command3.Size = New System.Drawing.Size(152, 33)
Me.Command3.TabIndex = 2
Me.Command3.Text = "disconnect"
,
' Command2
,

Me.Command2.BackColor = System.Drawing.SystemColors.Control
Me.Command2.Cursor = System.Windows.Forms.Cursors.Default
Me.Command2.ForeColor = System.Drawing.SystemColors.ControlText
Me.Command2.Location = New System.Drawing.Point(64, 56)
Me.Command2.Name = "Command2"
Me.Command2.RightToLeft = System.Windows.Forms.RightToLeft.No
Me.Command2.Size = New System.Drawing.Size(152, 31)
Me.Command2.TabIndex = 1
Me.Command2.Text = "Sending a E71 command"
,
' Command1
,

Me.Command1.BackColor = System.Drawing.SystemColors.Control
Me.Command1.Cursor = System.Windows.Forms.Cursors.Default
Me.Command1.ForeColor = System.Drawing.SystemColors.ControlText
Me.Command1.Location = New System.Drawing.Point(64, 16)
Me.Command1.Name = "Command1"
Me.Command1.RightToLeft = System.Windows.Forms.RightToLeft.No
Me.Command1.Size = New System.Drawing.Size(152, 31)
Me.Command1.TabIndex = 0
Me.Command1.Text = "connect"
```

```
,
' Form1
,

Me.AutoScaleBaseSize = New System.Drawing.Size(5, 12)
Me.BackColor = System.Drawing.SystemColors.Control
Me.ClientSize = New System.Drawing.Size(280, 214)
Me.Controls.Add(Me.Command5)
Me.Controls.Add(Me.Command4)
Me.Controls.Add(Me.Command3)
Me.Controls.Add(Me.Command2)
Me.Controls.Add(Me.Command1)
Me.Cursor = System.Windows.Forms.Cursors.Default
Me.Location = New System.Drawing.Point(329, 189)
Me.Name = "Form1"
Me.RightToLeft = System.Windows.Forms.RightToLeft.No
Me.StartPosition = System.Windows.Forms.FormStartPosition.Manual
Me.Text = "Form1"
Me.ResumeLayout(False)

End Sub
#End Region

#Region "Upgrade Wizard support code"
Private Shared m_vb6FormDefInstance As Form1
Private Shared m_InitializingDefInstance As Boolean
Public Shared Property DefInstance() As Form1
Get
    If m_vb6FormDefInstance Is Nothing OrElse m_vb6FormDefInstance.IsDisposed Then
        m_InitializingDefInstance = True
        m_vb6FormDefInstance = New Form1()
        m_InitializingDefInstance = False
    End If
    DefInstance = m_vb6FormDefInstance
End Get
Set
    m_vb6FormDefInstance = Value
End Set
End Property
#End Region
```



```
Private Sub Command1_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Command1.Click
    'Connect to the Ethernet interfece module.
    Dim sock As New Sockets.Socket(Sockets.AddressFamily.InterNetwork, _
    Sockets.SocketType.Stream, Sockets.ProtocolType.Tcp)
    Ajsock = sock
    Dim ip As IPAddress = Dns.Resolve("192.0.1.253").AddressList(0)
    Dim ipend As IPEndPoint = New IPEndPoint(ip, "8192")

    Me.Ajsock.Connect(ipend)
    MsgBox("Connection Completed")
    State = Me.Ajsock.Connected()

End Sub

Private Sub Command2_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Command2.Click
    Dim SData As Byte()
    Dim RData(256) As Byte

    'Rend D0 to D4 (5 points) with the A-compatible 1E frame command.
    SData = Encoding.ASCII.GetBytes("01FF000A4420000000000500")
    'Read D0 to D4 (5 points) with the QnA-compatible 3E frame command.
    'SData = Encoding.ASCII.GetBytes("500000FF03FF000018000A04010000D*0000000005")
    'Send the data.
    Me.Ajsock.Send(SData)
    MsgBox("Send completion", MsgBoxStyle.Information)

    'Read the response from the PLC CPU.
    Me.Ajsock.Receive(RData)
    MsgBox(Encoding.ASCII.GetString(RData), MsgBoxStyle.Information)

End Sub

Private Sub Command3_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Command3.Click
    'Close the TCP (UDP) connection socket (disconnect the line).
    Me.Ajsock.Shutdown(Net.Sockets.SocketShutdown.Both)
    Me.Ajsock.Close()
    MsgBox("The disconnection was successful", MsgBoxStyle.Information)
    State = Me.Ajsock.Connected()

End Sub
```

```
Private Sub Command4_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Command4.Click
    'End the program.
    End
End Sub
```

```
Private Sub Command5_Click(ByVal eventSender As System.Object, ByVal eventArgs
As System.EventArgs) Handles Command5.Click
    'Check the connection state.
    If State Then
        MsgBox("Connected")
    Else
        MsgBox("Closed")
    End If
End Sub
```

```
End Class
```

附录 8.2 使用 Visual Basic® 6.0/ Visual C++® 6.0 或更早版本的程序范例

附录 8.2.1 使用 MC 协议-1 进行通讯的程序范例

以下说明了程序、其执行环境和数据通讯的内容。

(1) 程序范例的执行环境

(a) PLC CPU 端

- 1) 安装站的以太网 QCPU 模块型号名 : Q25HCPU
- 2) 以太网模块的 I/O 信号 : X/Y000 至 X/Y01F
- 3) 以太网模块 IP 地址 : C0.00.01.FD_H (192.00.01.253)
- 4) 以太网模块端口编号 : 2000_H
- 5) GX Developer 设置
 - 操作设置 : 参阅下一页上的“(3) GX Developer 设置(a)”
 - 开放设置 : 参阅下一页上的“(3) GX Developer 设置(b)”

(b) 外部设备端

- 1) 操作环境 : Microsoft® Windows® 95 操作系统
- 2) 以太网接口板型号名 : WINSOCK 兼容板
- 3) 库 : WSOCK32.LIB
- 4) 软件开发环境 : Microsoft® Corporation Visual C++® (版本 4.0)
- 5) 以太网地址 : 可用 ARP 功能, 所以无需设置。
- 6) IP 地址 : 主动开放时接收。
- 7) 端口编号 : 主动开放时接收。

(c) 通讯系统

: TCP/IP

(2) 程序范例概述

(a) PLC CPU 端顺控程序

从 GX Developer 中设置参数。
(无需顺控程序)

(b) 外部设备端程序

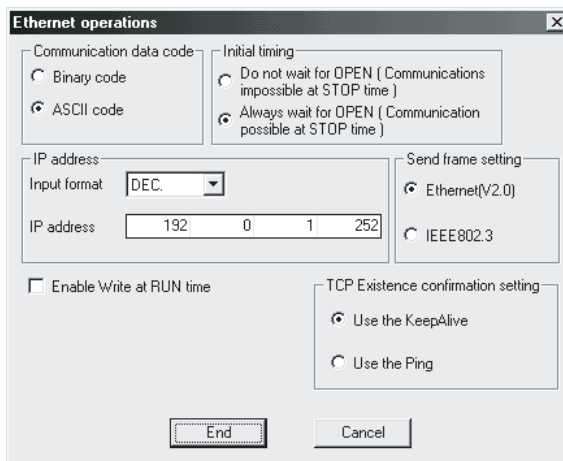
使用以上所述的库与 PLC CPU 执行以下读/写数据通讯。

- 以字单位写入 (D0 至 D4 的 5 点)
- 以字单位读出 (D0 至 D4 的 5 点)

(3) GX Developer 设置

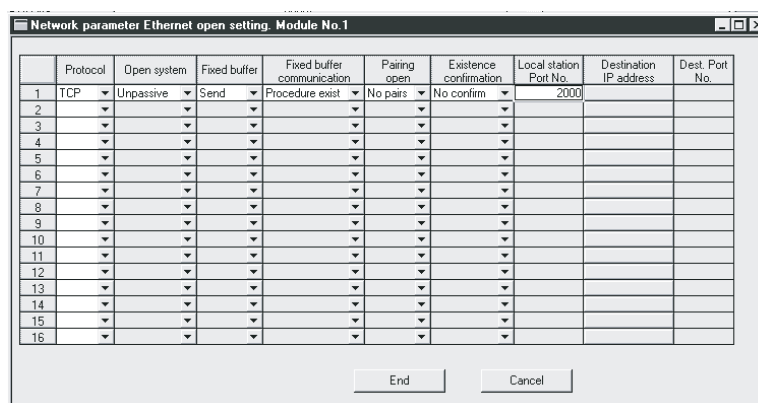
按以下说明设置 PLC CPU 参数。

(a) 操作设置



本地站 IP 地址：C0.00.01.FD_H (192.00.01.253)

(b) 开放设置



本地端口号：2000_H

(4) 外部设备端程序

以下显示的外部设备的程序范例访问安装有以太网模块的站的 Q25HCPU。
 执行此程序时，在序列发生器中显示以下通讯信息：

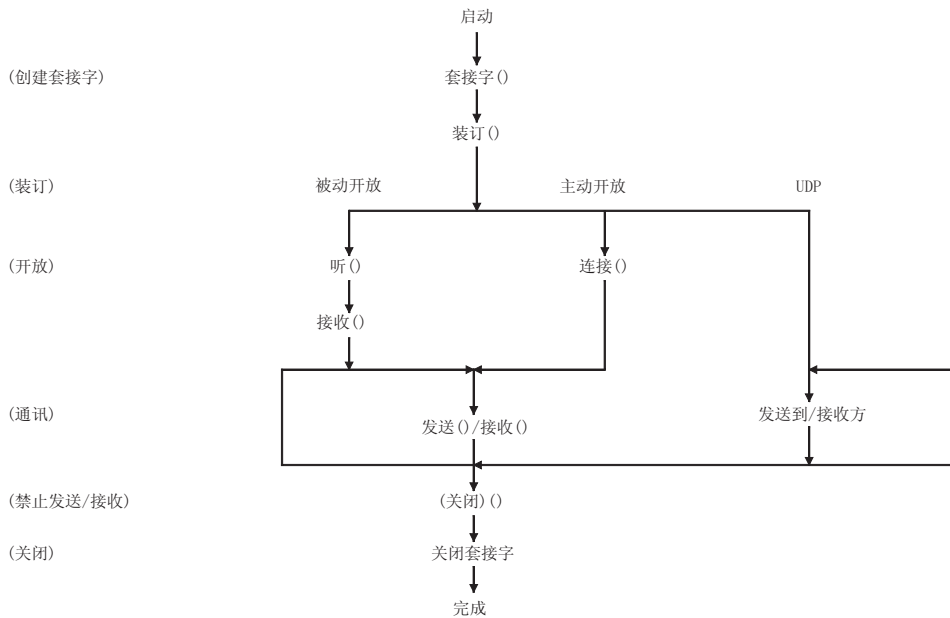
- 1) 以字单位批量写入指令信息
- 2) 以字单位批量写入响应信息
- 3) 以字单位批量读取指令信息
- 4) 以字单位批量读取响应信息

备注

(1) 以下概述了使用 Microsoft® Corporation Visual C++® 创建程序的编辑步骤。
 (版本 4.0)。

- 1) 启动 Visual C++®。
- 2) 准备创建文件。
 选择文件 → 新建，然后从程序工作区画面中启动控制台应用程序。
- 3) 打开文件名 AJSAMP.C 并创建一个程序。
 (参见下一页上的程序范例。)
- 4) 从创建菜单的编辑画面中编辑创建文件。
- 5) 从创建画面的设置画面中链接 WSOCK32.LIB。
- 6) 在创建菜单的创建画面上创建可执行文件 (AJSAMP.EXE)。
- 7) 结束 Visual C++®。
- 8) 执行 AJSAMP.EXE。

(2) 调用套接字路径的步骤概述



```

/ *****/
/ ** ** /
/ ** 示例程序 ** /
/ ** ** /
/ ** 该示例程序执行以太网模块和目标设备间的连接 ** /
/ ** 测试。 ** /
/ ** 该程序访问装有以太网模块的 PLC CPU 的数据寄 ** /
/ ** 存器(D)。 ** /
/ ** ** /
/ ** 版权(C) 2005 三菱电机 ** /
/ ** 保留所有版权 ** /
/ ** ** /
/ ** ** /
/ ** ** /
/ ** ** /
/ *****/

#include <stdio.h>
#include <winsock.h>

#define FLAG_OFF 0 // 完成标志 OFF
#define FLAG_ON 1 // 完成标志 ON
#define SOCK_OK 0 // 正常完成
#define SOCK_NG -1 // 异常完成
#define BUF_SIZE 4096 // 接收缓冲器大小

#define ERROR_INITIAL 0 // 初始化错误
#define ERROR_SOCKET 1 // 接口创建错误
#define ERROR_BIND 2 // 汇编错误
#define ERROR_CONNECT 3 // 连接错误
#define ERROR_SEND 4 // 发送错误
#define ERROR_RECEIVE 5 // 接收错误
#define ERROR_SHUTDOWN 6 // 关闭错误
#define ERROR_CLOSE 7 // 线路关闭错误

//用于检查接收大小的定义
//#define RECV_ANS_1 4 // 回复软元件写入(1E 帧)的响应信息接收大小
#define RECV_ANS_1 22 // 回复软元件写入(3E 帧)的响应信息接收大小
//#define RECV_ANS_2 24 // 回复软元件读取(1E 帧)的响应信息接收大小
#define RECV_ANS_2 42 // 回复软元件读取(3E 帧)的响应信息接收大小

typedef struct sck_inf{
    struct in_addr my_addr;
    unsigned short my_port;
    struct in_addr aj_addr;
    unsigned short aj_port;
};

```

```

int nErrorStatus; // 出错信息存储变量
int Dmykeyin; // 虚拟键输入
int Closeflag; // 连接完成标志
int socketno;

int main()
{
    WORD wVersionRequested=MAKEWORD(1, 1); // Winsock 版本 1.1 请求
    WSADATA wsaData;
    int length; // 通讯数据长度
    unsigned char s_buf[BUF_SIZE]; // 发送缓冲器
    unsigned char r_buf[BUF_SIZE]; // 接收缓冲器
    int rbuf_idx; // 接收数据存储起始索引
    int recv_size; // 接收数据量
    struct sock_inf sc;
    struct sockaddr_in hostdata; // 外部设备端数据
    struct sockaddr_in aj71e71; // 以太网模块端数据
    void Sockerror(int); // 错误处理功能

    unsigned long ulCmdArg ; // 无块模式设置标志

    sc.my_addr.s_addr=htonl(INADDR_ANY); // 外部设备端 IP 地址
    sc.my_port=htons(0); // 外部设备端端口号
    sc.aj_addr.s_addr=inet_addr("192.0.1.253"); // 以太网模块端 IP 地址
    // (C00001FDH)
    sc.aj_port=htons(0x2000); // 以太网模块端端口号

    Closeflag=FLAG_OFF; // 连接完成标志 OFF

    nErrorStatus=WSAStartup(wVersionRequested, &wsaData); // Winsock 初始化处理

    if (nErrorStatus!=SOCK_OK) {
        Sockerror(ERROR_INITIAL); // 错误处理
        return(SOCK_NG);
    }

    printf("Winsock Version is %ld.%ld\n", HIBYTE(wsaData.wVersion), LOBYTE(wsaData.wVersion));
    printf("AJ_test Start\n");

    socketno=socket(AF_INET, SOCK_STREAM, 0); // 为 TCP/IP 创建接口

    if (socketno==INVALID_SOCKET) {
        Sockerror(ERROR_SOCKET); // 错误处理
        return(SOCK_NG);
    }

    hostdata.sin_family=AF_INET;
    hostdata.sin_port=sc.my_port;
    hostdata.sin_addr.s_addr=sc.my_addr.s_addr;

```

```

if (bind(socketno, (LPSOCKADDR)&hostdata, sizeof(hostdata))!=SOCK_OK) {
    // 汇编
    Sockerror(ERROR_BIND); // 错误处理
    return(SOCK_NG);
}

aj71e71.sin_family=AF_INET;
aj71e71.sin_port=sc.aj_port;
aj71e71.sin_addr.s_addr=sc.aj_addr.s_addr;

if (connect(socketno, (LPSOCKADDR)&aj71e71, sizeof(aj71e71))!=SOCK_OK) {
    // 连接(主动开放)
    Sockerror(ERROR_CONNECT); // 错误处理
    return(SOCK_NG);
}

Closeflag=FLAG_ON; // 连接完成标志 ON

// 设置为无块模式
ulCmdArg = 1;
ioctlsocket(socketno, FIONBIO, &ulCmdArg); // 设置为无块模式

// strcpy(s_buf, "03FF000A442000000000500112233445566778899AA");
// D0 至 D4 批量写入请求(1E 帧)
strcpy(s_buf, "500000FF03FF00002C000A14010000D*000000005112233445566778899AA");
// D0 至 D4 批量写入请求(3E 帧)

length=strlen(s_buf);

if (send(socketno, s_buf, length, 0)==SOCKET_ERROR) { // 数据发送
    Sockerror(ERROR_SEND); // 错误处理
    return (SOCK_NG);
}
printf("\n send data\n%s\n", s_buf);

// Perform receiving size check and receiving processing simultaneously
rbuf_idx = 0; // 接收数据存储起始索引初始化
recv_size = 0; // 初始化接收数据量
while(1) {
    length = recv(socketno, &r_buf[rbuf_idx], (BUF_SIZE - rbuf_idx), 0);
    // 响应数据接收
    if (length == 0) { // 断开连接吗?
        Sockerror(ERROR_RECIEVE); // 错误处理
        return (SOCK_NG);
    }
}

```



```

    if (length == SOCKET_ERROR) {
        nErrorStatus = WSAGetLastError();
        if (nErrorStatus != WSAEWOULDBLOCK) {
            Sockerror(ERROR_RECEIVE); // 错误处理
            return (SOCK_NG);
        } else {
            continue; // 接收数据前重复
        }
    } else {
        rbuf_idx += length; // 更新接收数据存储位置
        recv_size += length; // 更新接收数据量
        if (recv_size >= RECV_ANS_1) // 接收了所有响应信息吗?
            break; // 接收?
        // 已经接收信息时停止重复
        // 已接收
    }
}
r_buf[rbuf_idx] = '\0' ; // 在接收数据末端设置 NULL

printf("\n receive data\n%s\n", r_buf);

// strcpy(s_buf, "01FF000A442000000000500"); // D0 至 D4 批量写入请求
// (1E 帧)
strcpy(s_buf, "50000FF03FF000018000A04010000D*000000005"); // D0 至 D4 批量读取请求
// (3E 帧)

length=strlen(s_buf);

if (send(socketno, s_buf, length, 0)==SOCKET_ERROR) { // 数据发送
    Sockerror(ERROR_SEND); // 错误处理
    return (SOCK_NG);
}
printf("\n send data\n%s\n", s_buf);

// 同时执行接收大小检查和接收处理
rbuf_idx = 0; // 接收数据存储起始索引
// 初始化
recv_size = 0; // 初始化接收数据量
while(1) {
    length = recv(socketno, &r_buf[rbuf_idx], (BUF_SIZE - rbuf_idx), 0);
    // 响应数据接收
    if (length == 0) { // 连接断开了吗?
        Sockerror(ERROR_RECEIVE); // 错误处理
        return (SOCK_NG);
    }
}

```

```

    if (length == SOCKET_ERROR) {
        nErrorStatus = WSAGetLastError();
        if (nErrorStatus != WSAEWOULDBLOCK) {
            Sockerror(ERROR_RECIEVE); // 错误处理
            return (SOCK_NG);
        } else {
            continue; // 接收信息前重复
        }
    } else {
        rbuf_idx += length; // 更新接收数据存储
                           // 位置
        recv_size += length; // 更新接收数据量
        if (recv_size >= RECV_ANS_2) // 接收了所有响应吗?
            // 接收?
            break; // 已经接收好信息时停止
                  // 已接收
    }
}
r_buf[rbuf_idx] = '\0' ; // 在接收数据末端设置 NULL

printf("\receive data\n%s\n", r_buf);

    if (shutdown(socketno, 2) !=SOCK_OK) { // 无法进行
                                           // 发送/接收处理
                                           // 错误处理
        Sockerror(ERROR_SHUTDOWN);
        return (SOCK_NG);
    }

    if (closesocket(socketno) !=SOCK_OK) { // 关闭处理
                                           // 错误处理
        Sockerror(ERROR_CLOSE);
        return (SOCK_NG);
    }

    Closeflag=FLAG_OFF; // 连接完成标志 OFF
    WSACleanup(); // 释放 Winsock.DLL

    printf("\nAJ_test End.\n\n Normally completed. \n");
    printf("Press any key 至 exit the program.\n");
    Dmykeyin=getchar(); // 等待键输入

    return (SOCK_OK);
}

void Sockerror(int error_kind) // 错误处理功能
{
    if (error_kind==ERROR_INITIAL) {
        printf("Initial processing is abnormal.");
    }
}

```

```
else {
    nErrorStatus=WSAGetLastError();
    switch(error_kind) {
    case ERROR_SOCKET:
        printf("Failed to create socket.");
        break;
    case ERROR_BIND:
        printf("Failed to bind.");
        break;
    case ERROR_CONNECT:
        printf("Failed to establish connection.");
        break;
    case ERROR_SEND:
        printf("Sending failed.");
        break;
    case ERROR_RECEIVE:
        printf("Receiving failed.");
        break;
    case ERROR_SHUTDOWN:
        printf("Failed to shutdown.");
        break;
    case ERROR_CLOSE:
        printf("Failed to close normally.");
        break;
    }
}

printf("Error code is %d.\n", nErrorStatus);

if (Closeflag==FLAG_ON) {
    nErrorStatus=shutdown(socketno, 2); // 关机处理
    nErrorStatus=closesocket(socketno); // 关闭处理
    Closeflag=FLAG_OFF; // 连接完成标志 OFF
}

printf("Press any key 至 exit the program.\n");
Dmykeyin=getchar(); // 等待键输入
WSACleanup(); // 释放 Winsock.DLL
return;
}
```

附录 8.2.2 使用 MC 协议-2 通讯的程序范例

本节说明了程序范例、执行环境和数据通讯内容。

(1) 程序范例的执行环境

(a) PLC CPU 端

- 1) 安装站的以太网模块 QCPU 型号名 : Q25PRHCPU
- 2) 以太网模块的 I/O 信号 : X/Y000 至 X/Y01F
- 3) 以太网模块 IP 地址(系统 A) : C0.00.01.FC_H
(192.00.01.252)
- 以太网模块 IP 地址(系统 B) : C0.00.01.FD_H
(192.00.01.253)
- 4) 以太网模块端口编号 : 2000_H
- 5) GX Developer 设置
 - 操作设置:参阅下一页上的“(3)GX Developer 设置(a)”。
 - 开放设置:参阅下一页上的“(3)GX Developer 设置(b)”。
 - 冗余设置:参阅下一页上的“(3)GX Developer 设置(c)”。

(b) 外部设备端

- 1) 操作环境:
 - Microsoft® Windows® XP Professional 操作系统。
 - Microsoft® Windows® XP Home Edition 操作系统。
- 2) 以太网接口板型号名 : WINSOCK 兼容板
- 3) 库 : WSOCK32.LIB
- 4) 软件开发环境 :
使用微软公司制造的 Visual C++® (版本 6.0)。
- 5) 以太网地址 : 不必设置为 APP 功能。
- 6) IP 地址 : 在 Active 开放时接收。
- 7) 端口编号 : 在 Active 开放时接收。

(c) 通讯系统 : TCP/IP

(2) 程序范例概述

(a) PLC CPU 端顺控程序

必须使用 GX Developer 进行参数设置。
(无需顺控程序)

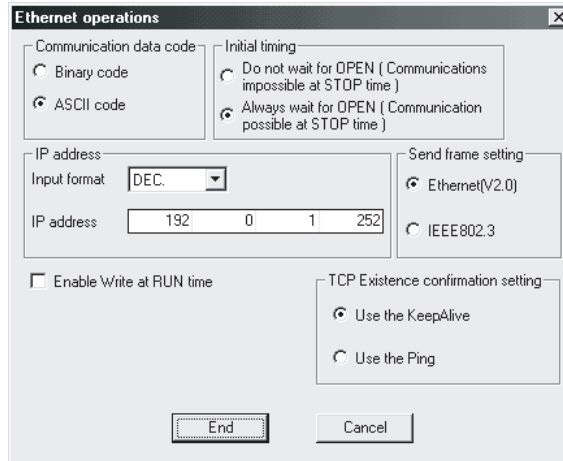
(b) 外部设备端程序

使用以上库按照字单位写入数据到冗余 CPU(控制系统)的数据寄存器 D0 至 D4(5 点)。
如果此时由于通讯出错或相关原因无法从系统 A 端写入数据,数据则从系统 B 端写入到冗余系统(控制系统)的数据寄存器中。

(3) GX Developer 设置

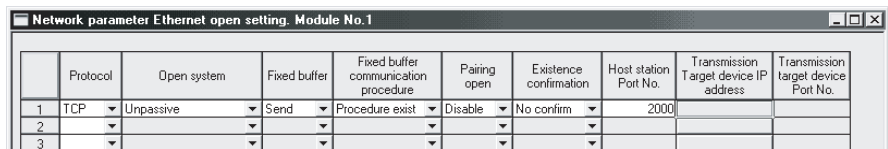
按以下说明设置 PLC CPU 参数。

(a) 操作设置



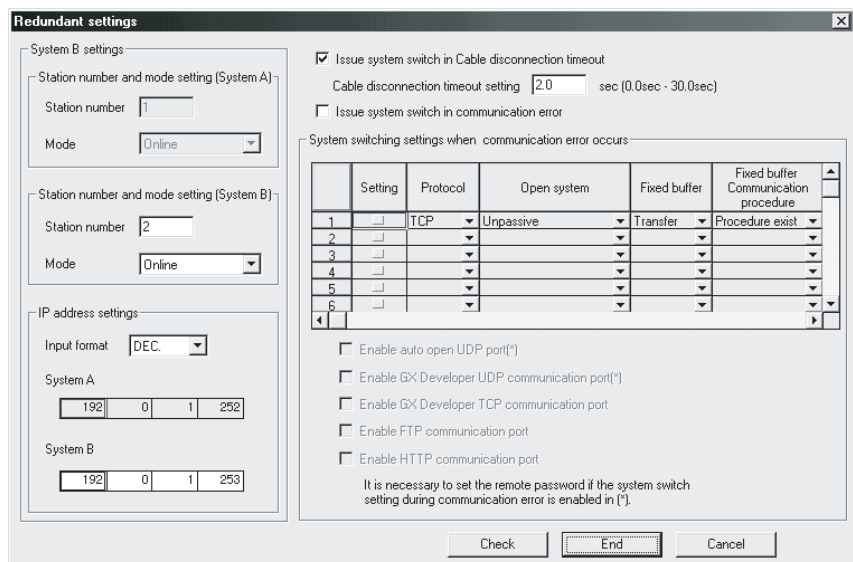
以太网模块 IP 地址(系统 A) : C0.00.01.FCH (192.00.01.252)

(b) 开放设置



本地端口号 : 2000H

(c) 冗余设置



以太网模块 IP 地址(系统 B) : C0.00.01.FDH (192.00.01.253)

(4) 外部设备端程序

以下显示访问安装站的以太网模块 Q25PRHCPU 外部设备的程序范例。
执行此程序时，按顺序显示以下内容。

- 1) 使用的 Winsock 版本。
- 2) 测试启动信息。
- 3) 以字单位写入批量指令信息
- 4) 以字单位写入批量响应信息
- 5) 测试结束信息

备注

以下说明使用 Microsoft® Visual C++® (版本 6.0) 创建程序的一般编辑步骤。

- 1) 使用文本编辑器创建“QJSAMP.C”文件。
(参见下一页的程序范例)
- 2) 启动 Visual C++®。
- 3) 打开“QJSAMP.C”文件。
选择文件 → 打开，指定步骤 1) 中创建的“QJSAMP.C”，并打开它。
- 4) 在创建菜单的编辑画面上执行创建程序的编辑。
选择创建 → 编辑。
在显示的信息后创建一个缺省值的程序工作区。
- 5) 从程序设置画面上连接 WSOCK32.LIB。
选择程序 → 设置来显示“程序设置”对话框。
选择链接标签并添加 WSOCK32.LIB 到“目标/库模块”。
- 6) 在创建菜单的创建画面上新建一个执行文件(QJSAMP.EXE)。
选择创建 → 创建。
- 7) 退出 Visual C++®。
- 8) 执行 QJSAMP.EXE。

```

/ *****/
/ **          ** /
/ **  示例程序          ** /
/ **          ** /
/ **  该示例程序执行以太网模块和外部设备间的连接 ** /
/ **  测试。          ** /
/ **  该程序访问装有以太网模块的冗余 CPU(控制系统) ** /
/ **  的数据寄存器(D)。          ** /
/ **          ** /
/ **  版权(C) 2005 三菱电机          ** /
/ **  保留所有版权          ** /
/ **          ** /
/ **          ** /
/ **          ** /
/ *****/

#include <stdio.h>
#include <winsock.h>

#define FLAG_OFF          0          // 结束标志 OFF
#define FLAG_ON          1          // 结束标志 ON
#define SOCK_OK          0          // 正常结束
#define SOCK_NG          -1         // 未结束
#define BUF_SIZE          4096      // 接收缓冲器大小

#define ERROR_NO_ERROR    0          // 无错误
#define ERROR_INITIAL    1          // 初始化错误
#define ERROR_SOCKET      2          // 接口创建错误
#define ERROR_BIND        3          // 汇编错误
#define ERROR_CONNECT     4          // 连接错误
#define ERROR_SEND        5          // 发送错误
#define ERROR_SHUTDOWN    6          // 关闭错误
#define ERROR_CLOSE       7          // 线路关闭错误

// 检查接收大小的定义
#define RECV_ANS_1          22      // 答复软元件写入的响应信息接收大小(3E 帧)

typedef struct sck_inf {
    struct in_addr my_addr;
    unsigned short my_port;
    struct in_addr qj_addr;
    unsigned short qj_port;
};

int nErrorStatus;          // 出错信息存储变量
int Dmykeyin;             // 虚拟键输入
int ShutdownflagA;        // 关闭标志(用于系统 A 连接)
int ShutdownflagB;        // 关闭标志(用于系统 B 连接)

```

```

int CloseflagA; // 连接结束标志 (用于系统 A 连接)
int CloseflagB; // 连接结束标志 (用于系统 B 连接)
int socketnoA;
int socketnoB;
int ConnectLastErrorA; // 连接处理错误信息(用于系统 A 连接)
int ConnectLastErrorB; // 连接处理错误信息(用于系统 B 连接)
int SendFlag; // 发送完成标志

int main()
{
    WORD wVersionRequested = MAKEWORD(1, 1); // Winsock 版本 1.1 请求
    WSADATA wsaData;
    int length; // 通讯数据长度
    unsigned char s_buf[BUF_SIZE]; // 发送缓冲器
    unsigned char r_bufA[BUF_SIZE], r_bufB[BUF_SIZE]; // 接收缓冲器
    struct sock_inf scA, scB;
    struct sockaddr_in hostdataA, hostdataB; // 外部设备端数据
    struct sockaddr_in qj71e71A, qj71e71B; // 以太网模块端数据
    BOOL DataRecv(int, unsigned char *, int); // 接收处理功能
    void Sockerror(int, int); // 错误处理功能

    unsigned long ulCmdArgA, ulCmdArgB; // 无块模式设置标志

    scA.my_addr.s_addr = scB.my_addr.s_addr = htonl(INADDR_ANY); // 外部设备端 IP 地址
    scA.my_port = scB.my_port = htons(0); // 外部设备端端口号
    scA.qj_addr.s_addr = inet_addr("192.0.1.252"); // 以太网模块端 IP 地址(系统 A: C00001FCh)
    scB.qj_addr.s_addr = inet_addr("192.0.1.253"); // 以太网模块端 IP 地址(系统 B: C00001FDh)
    scA.qj_port = scB.qj_port = htons(0x2000); // 外部设备端端口号

    ShutdownflagA = ShutdownflagB = FLAG_OFF; // 关闭标志 OFF
    CloseflagA = CloseflagB = FLAG_OFF; // 连接结束标志 OFF

    nErrorStatus = WSASStartup(wVersionRequested, &wsaData); // Winsock 初始化处理

    ConnectLastErrorA = ERROR_NO_ERROR; // 连接处理出错信息初始化(用于系统 A)
    ConnectLastErrorB = ERROR_NO_ERROR; // 连接处理出错信息初始化 (用于系统 B)

    if(nErrorStatus != SOCK_OK) {
        Sockerror(ERROR_INITIAL, ERROR_INITIAL); // 错误处理
        return (SOCK_NG);
    }

    printf("Winsock Version is %1d.%1d\n", HIBYTE(wsaData.wVersion), LOBYTE(wsaData.wVersion));
    printf("QJ_test Start\n");
}

```



```

// 系统 A 连接处理
socketnoA = socket(AF_INET, SOCK_STREAM, 0); // TCP/IP 接口 (用于系统 A 连接) 创建
if(socketnoA != INVALID_SOCKET) {
    hostdataA.sin_family    = AF_INET;
    hostdataA.sin_port      = scA.my_port;
    hostdataA.sin_addr.s_addr= scA.my_addr.s_addr;

    if(bind(socketnoA, (LPSOCKADDR)&hostdataA, sizeof(hostdataA)) == SOCK_OK) {
                                                // 汇编(系统 A)

        qj71e71A.sin_family    = AF_INET;
        qj71e71A.sin_port      = scA.qj_port;
        qj71e71A.sin_addr.s_addr= scA.qj_addr.s_addr;

        if(connect(socketnoA, (LPSOCKADDR)&qj71e71A, sizeof(qj71e71A)) == SOCK_OK) {
                                                // 汇编(主动开放:系统 A)
            ShutdownflagA = FLAG_ON;           // 关闭标志 ON
            CloseflagA = FLAG_ON;             // 连接结束标志 ON
            // 设置为无块模式
            ulCmdArgA = 1;
            ioctlsocket(socketnoA, FIONBIO, &ulCmdArgA);
                                                // 设置为无块模式(用于系统 A 连接)
        } else {
            ConnectLastErrorA = ERROR_CONNECT; // 建立连接失败
        }
    } else {
        ConnectLastErrorA = ERROR_BIND;       // 汇编失败
    }
} else {
    ConnectLastErrorA = ERROR_SOCKET;         // 汇编创建失败
}

// 系统 B 连接处理
socketnoB = socket(AF_INET, SOCK_STREAM, 0); // TCP/IP 接口 (用于系统 B 连接) 创建
if(socketnoB != INVALID_SOCKET) {
    hostdataB.sin_family    = AF_INET;
    hostdataB.sin_port      = scB.my_port;
    hostdataB.sin_addr.s_addr= scB.my_addr.s_addr;

    if(bind(socketnoB, (LPSOCKADDR)&hostdataB, sizeof(hostdataB)) == SOCK_OK) {
                                                // 汇编(系统 A)

        qj71e71B.sin_family    = AF_INET;
        qj71e71B.sin_port      = scB.qj_port;
        qj71e71B.sin_addr.s_addr= scB.qj_addr.s_addr;

```

```

        if(connect(socketnoB, (LPSOCKADDR)&qj71e71B, sizeof(qj71e71B)) == SOCK_OK) {
                                                    // 连接(主动开放: 系统 B)
            ShutdownflagB = FLAG_ON;                // 关闭标志 ON
            CloseflagB = FLAG_ON;                  // 连接结束标志 ON
            // 设置为无块模式
            uICmdArgB = 1;
            ioctlsocket(socketnoB, FIONBIO, &uICmdArgB);
                                                    // 设置为无块模式(用于 B 连接)
        } else {
            ConnectLastErrorB = ERROR_CONNECT;     // 建立连接失败
        }
    } else {
        ConnectLastErrorB = ERROR_BIND;           // 汇编失败
    }
} else {
    ConnectLastErrorB = ERROR_SOCKET;            // 创建接口失败
}

// 连接完成处理
if( (CloseflagA == FLAG_OFF) && (CloseflagB == FLAG_OFF) ){// 两个站都为异常时
    Sockerror(ConnectLastErrorA, ConnectLastErrorB); // 错误处理
    return (SOCK_NG);
}

strcpy(s_buf, "500000FF03D000002C000A14010000D*0000000005112233445566778899AA");
                                                    // D0-D4 批量写入请求 (3E 帧, 发送到控制系统)

length = strlen(s_buf);

printf("Send starts. Press any key. \n");
Dmykeyin = getchar();                            // 等待键输入

SendFlag = FLAG_OFF;                             // 发送完成标志 OFF
// 系统 A 发送处理
if( CloseflagA == FLAG_ON && (SendFlag == FLAG_OFF) ){
    if(send(socketnoA, s_buf, length, 0) != SOCKET_ERROR) { // 数据发送(系统 A)
        printf("\n Send data (System A) \n%s\n", s_buf); // 发送数据显示(系统 A)
        SendFlag = FLAG_ON;                             // 发送完成标志 ON
        // Receive processing
        if(DataRecv(socketnoA, r_bufA, RECV_ANS_1) == TRUE) { // 接收处理
            printf("\n Receive data (System A) \n%s\n", r_bufA); // 数据接收
        } else {
            printf("Receive failure (System A) \n");
        }
    } else {
        printf("Send failure (System A) \n");
    }
}
}

```

```

// 系统 B 发送处理
if( (CloseflagB == FLAG_ON) && (SendFlag == FLAG_OFF) ){
    if(send(socketnoB, s_buf, length, 0) != SOCKET_ERROR) { // 数据发送(系统 B)
        printf("\n Send data (System B) \n%s\n", s_buf); // 发送数据显示(系统 B)
        SendFlag = FLAG_ON; // 发送完成标志 ON
        // Receive processing
        if(DataRecv(socketnoB, r_bufB, RECV_ANS_1) == TRUE) { // 接收处理
            printf("\n Receive data (System B) \n%s\n", r_bufB); // 数据接收
        } else {
            printf("Receive failure (System B) \n");
        }
    }else{
        printf("Send failure (System B) \n");
    }
}

// 发送完成处理
if( SendFlag == FLAG_OFF ){
    Sockerror(ERROR_SEND, ERROR_SEND); // 错误处理
    return (SOCK_NG);
}

if(CloseflagA == FLAG_ON) {
    ShutdownflagA = FLAG_OFF; // 关闭标志 OFF
    if(shutdown(socketnoA, 2) != SOCK_OK) { // 发送/接收禁止处理(系统 A)
        Sockerror(ERROR_SHUTDOWN, ERROR_NO_ERROR); // 错误处理
        return (SOCK_NG);
    }
}

if(CloseflagB == FLAG_ON) {
    ShutdownflagB = FLAG_OFF; // 关闭标志 OFF
    if(shutdown(socketnoB, 2) != SOCK_OK) { // 发送/接收禁止处理(系统 B)
        Sockerror(ERROR_NO_ERROR, ERROR_SHUTDOWN); // 错误处理
        return (SOCK_NG);
    }
}

CloseflagA = FLAG_OFF; // 连接结束标志 OFF
if(closesocket(socketnoA) != SOCK_OK) { // 关闭处理(系统 A)
    Sockerror(ERROR_CLOSE, ERROR_NO_ERROR); // 错误处理
    return (SOCK_NG);
}

```

```

CloseflagB = FLAG_OFF; // 连接结束标志 OFF
if(closesocket(socketnoB) != SOCK_OK) { // 结束处理(系统 B)
    Sockerror(ERROR_NO_ERROR, ERROR_CLOSE); // 错误处理
    return (SOCK_NG);
}

WSACleanup(); // Winsock.DLL 释放

printf("\nQJ_test End. \n\nNormally completed. \n");
printf("Program is closed. Press any key. \n");
Dmykeyin = getchar(); // 等待键输入

return (SOCK_OK);
}

BOOL DataRecv(int socketno, unsigned char *pR_buf, int size_max) // 接收处理功能
{
    int length; // 通讯数据程度
    int rbuf_idx; // 接收数据存储器起始索引
    int recv_size; // 接收数据量

    // 同时检查大小执行接收处理
    rbuf_idx = 0; // 接收数据存储器启动索引初始化
    recv_size = 0; // 初始化接收数据量
    while(1) {
        length = recv(socketno, (pR_buf + rbuf_idx), (BUF_SIZE - rbuf_idx), 0);
        // 响应数据接收
        if(length == 0) { // 连接断开了吗?
            return (FALSE); // 错误处理
        }
        if(length == SOCKET_ERROR) {
            nErrorStatus = WSAGetLastError();
            if(nErrorStatus != WSAEWOULDBLOCK) {
                return (FALSE); // 错误处理
            } else {
                continue; // 接收数据前重复
            }
        } else {
            rbuf_idx += length; // 更新接收数据存储器位置
            recv_size += length; // 更新接收数据量
            if(recv_size >= size_max) // 接收所有响应信息了吗?
                break; // 接收数据时停止重复
        }
    }
    *(pR_buf + rbuf_idx) = '\0'; // 在接收数据末端
    // 设置为 NULL
}

```

```
    return (TRUE);                // 正常结束
}

void Sockerror(int error_kind_A, int error_kind_B)    // 错误处理功能
{
    if (error_kind_A == ERROR_INITIAL) {
        printf("Initial processing is abnormal. \n");
    }
    else{
        nErrorStatus = WSAGetLastError();
        switch(error_kind_A) {
            case ERROR_SOCKET:
                printf("Socket could not be created. (System A)\n");
                break;
            case ERROR_BIND:
                printf("Bind could not be executed. (System A)\n");
                break;
            case ERROR_CONNECT:
                printf("Connection could not be established. (System A)\n");
                break;
            case ERROR_SEND:
                printf("Send could not be executed. \n");
                break;
            case ERROR_SHUTDOWN:
                printf("Shutdown could not be executed. (System A)\n");
                break;
            case ERROR_CLOSE:
                printf("Normal close could not be executed. (System A)\n");
                break;
        }
        switch(error_kind_B) {
            case ERROR_SOCKET:
                printf("Socket could not be created. (System B)\n");
                break;
            case ERROR_BIND:
                printf("Bind could not be executed. (System B)\n");
                break;
            case ERROR_CONNECT:
                printf("Connection could not be established. (System B)\n");
                break;
            case ERROR_SHUTDOWN:
                printf("Shutdown could not be executed. (System B)\n");
                break;
            case ERROR_CLOSE:
                printf("Normal close could not be executed. (System B)\n");
                break;
        }
    }
}
```

```
    }  
}  
printf("Error code is %d. \n", nErrorStatus);  
  
if (ShutdownflagA == FLAG_ON) {  
    nErrorStatus = shutdown(socketnoA, 2);           // 关闭处理(系统 A)  
    ShutdownflagA = FLAG_OFF;                       // 关闭标志 OFF (系统 A)  
}  
if (ShutdownflagB == FLAG_ON) {  
    nErrorStatus = shutdown(socketnoB, 2);           // 关闭处理(系统 B)  
    ShutdownflagB = FLAG_OFF;                       // 关闭标志 OFF(系统 B)  
}  
  
if (CloseflagA == FLAG_ON) {  
    nErrorStatus = closesocket(socketnoA);           // 关闭处理(系统 A)  
    CloseflagA = FLAG_OFF;                          // 连接结束标志 OFF(系统 A)  
}  
if (CloseflagB == FLAG_ON) {  
    nErrorStatus = closesocket(socketnoB);           // 结束处理(系统 B)  
    CloseflagB = FLAG_OFF;                          // 连接结束标志 OFF(系统 B)  
}  
  
printf("Program is closed. Press any key. \n");  
Dmykeyin = getchar();                               // 等待键输入  
WSACleanup();                                       // Winsock.DLL 释放  
return;  
}
```

附录 8.2.3 使用 MC 协议-3 通讯的程序范例

本节说明了从 PLC CPU 中读取数据和写入数据到 PLC CPU 的外部设备程序的范例。
以下显示一个示例程序、其执行环境和数据通讯的内容。

(1) 程序范例的执行环境

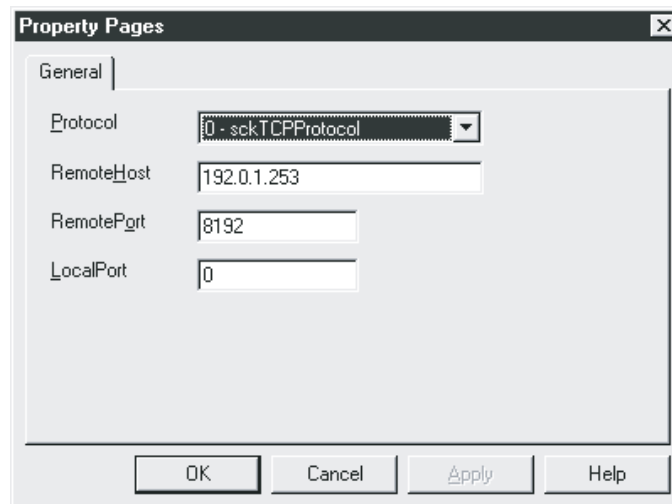
- 1) PLC CPU 端的设置与附录 8.2.1 节(1) (a)和(3)中说明的执行环境相同。
- 2) 除了以下所含的软件开发，外部设备端的设置与附录 8.2.1 节(1) (b)中说明的执行环境相同：
 - 软件开发环境：Microsoft® 公司 Visual Basic® (版本 6.0)
 - 对 IP 地址和端口号分配任意值。
- 3) 通讯协议为 TCP/IP

(2) 程序范例概述

使用 A 兼容 1E 帧指令(01:以字单位批量读取)，该程序从安装有以太网模块的 QCPU 站上 D0 至 D4(5 点)中读取数据。

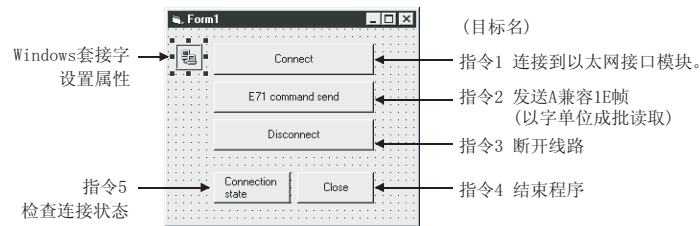
(3) 示例程序概述

- (a) 创建一个新的程序和表格。
- (b) 使用工具箱中的“指令按钮”来新建(4)中显示的(示例)窗口。
- (c) 使用元件控制添加“Microsoft Winsock Control 6.0”。
从工具箱中添加“Winsock”到表格并按照如下设置属性页：



- (d) 新建(5)中显示的程序。

(4) 窗口范例 (Form 1)



(5) 示例程序 (Form 1)

```
Option Explicit
```

```
Private Sub Command1_Click()
```

```
' Connect to the Ethernet interface module
  Me.Winsock1.Connect
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
' When calling an Active open method from a PC to the Ethernet interface module,
' The property screen should be used or the settings performed as follows.
' Specify the protocol to be used.
' Winsock1.Protocol = sckTCPProtocol / sckUDPProtocol
' Specify the IP address of the Ethernet interface module.
' Winsock1.RemoteHost = "192.0.1.253"
' Specify the port No. used by the Ethernet interface module.
' Winsock1.RemotePort = 8192 :H2000
' If the open method of the Ethernet interface module is Fullpassive open,
' specify the set port No.
' If the open method of the Ethernet interface module is Unpassive open,
' "0" - use any port No.
' Winsock1.LocalPort = 0 :Unpassive open
```

```
End Sub
```

```
Private Sub winsock1_connect()
```

```
' Use the Connect event to perform confirmation processing at the time when the connection
processing is normally completed.
```

```
' The Connect event occurs when the connection processing is completed.
```

```
  MsgBox "Connection Completed"
```

```
End Sub
```



```
Private Sub Command2_Click()  
Dim SData As String  
' Read D0 to D4 (5 points) with the A-compatible 1E frame command.  
    SData = "01ff000a4420000000000500"  
' Send the data.  
    Me.Winsock1.SendData SData  
  
End Sub  
  
Private Sub Command3_Click()  
' Close the TCP connection socket (disconnect the line).  
    Me.Winsock1.Close  
  
End Sub  
  
Private Sub Command4_Click()  
' End the program.  
    End  
  
End Sub  
  
Private Sub Command5_Click()  
' Check the state of Winsock.  
' See the Help of Visual Basic for details.  
    MsgBox Winsock1.State  
  
End Sub  
  
Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long)  
' The DataArrival event occurs when new data arrives.  
  
Dim RData As String  
' Obtain the current data block and save it as a variant type variable.  
' Read the response from the PLC CPU.  
    Me.Winsock1.GetData RData  
    MsgBox RData  
  
End Sub
```

附录 9 通讯支持工具(MX Component)

MX Component 是支持 IBM PC/AT 兼容个人计算机和 PLC 间任何通讯路径类型的 ActiveX 控制库。只需简单处理就可使用户建立通讯，无需了解每个通讯中使用的通讯协议的差异。

而且，它支持不同的编程语言，允许大范围的应用开发。

本节说明了 MX Component 提供的功能概述并说明了创建应用程序的步骤。

* 关于详情，请参见 MX Component 的操作手册和编程手册。

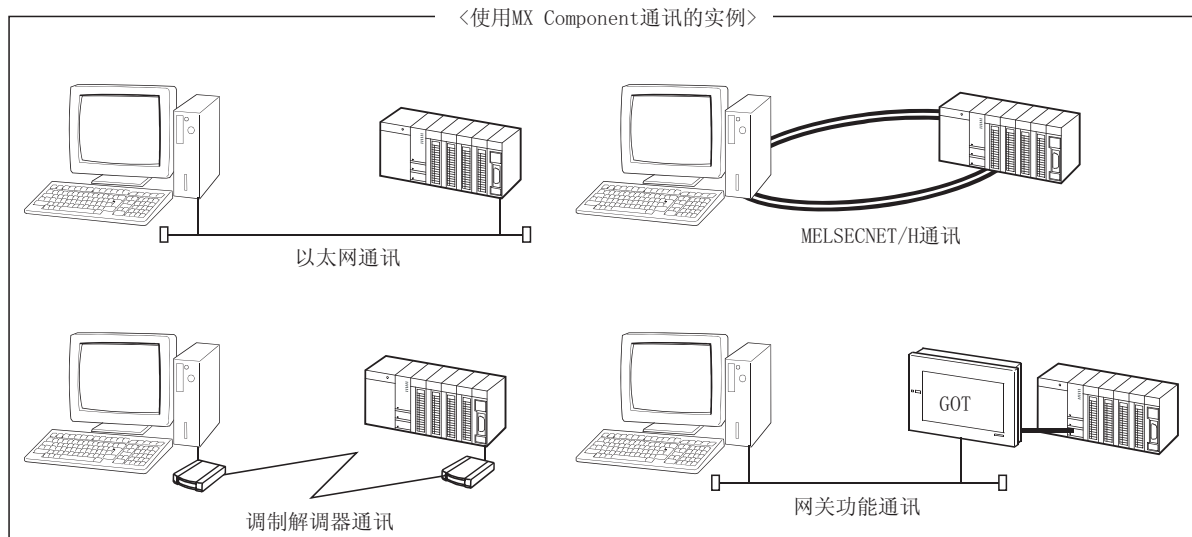
附录 9.1 MX Component 概述

本节说明了 MX Component 的概述

根据使用的 MX Component 版本不同，支持不同的通讯路径、操作系统、编程语言和功能。

(1) 支持到 PLC 的多条通讯路径

MX Component 支持到 PLC 的多条通讯路径。可以根据用户需要建立系统。



(2) 大大提高应用程序开发效率

MX Component 给出了向导型的通讯设置效用。

用户只需以交互方式在画面上显示的菜单中选择设置来完成访问目标 PLC CPU 所需的通讯设置。

而且，一旦执行通讯设置，只要通过通讯设置效用指定存储的逻辑站号就可以访问 PLC CPU。

(3) 支持基本操作系统的多种选择

MX Component 可以在运行以下基本操作系统的 IBM PC/AT 个人计算机上运行：

- Microsoft® Windows® 95 操作系统
- Microsoft® Windows® 98 操作系统
- Microsoft® Windows NT® Workstation 操作系统版本 4.0
- Microsoft® Windows® Millennium Edition 操作系统
- Microsoft® Windows® 2000 Professional 操作系统
- Microsoft® Windows® XP Professional 操作系统
- Microsoft® Windows® XP Home Edition 操作系统

(4) 支持多种编程语言

MX Component 支持以下编程语言。

允许用户开发多用户化应用程序。

编程语言	开发软件
Visual Basic®	Microsoft® Visual Basic® 6.0、Microsoft® Visual Basic® .NET 2003。
Visual C++®	Microsoft® Visual C++® 6.0、Microsoft® Visual C++® .NET 2003。
VBScript	文本编辑器和市售的 HTML 工具。
VBA	Microsoft® Excel 2000、Microsoft® Excel 2002, Microsoft® Access 2000 或 Microsoft® Access 2002

(5) 支持专用于与 PLC 进行数据通讯的功能

MX Component 提供与 PLC 进行数据通讯所需的功能，包括开放/关闭通讯线功能和读取/写入软元件功能。

因此，MX Component 可以方便的开发多功能通讯程序。

- (a) 使用 Microsoft® Visual Basic®.NET 2003 或 Microsoft® Visual C++®.NET 2003 时。

函数名	功能
Connect	连接电话线。
Open	开放通讯线。
Close	关闭通讯线。
Disconnect	断开电话线。
GetErrorMessage	显示出错定以及更正方法。
ReadDeviceBlock	批量读取软元件中数据。(INT 类型)
WriteDeviceBlock	批量写入数据到软元件。(INT 类型)
ReadDeviceBlock2	批量读取软元件中数据。(SHORT 类型)
WriteDeviceBlock2	批量写入数据到软元件。(SHORT 类型)
ReadDeviceRandom	随机从软元件中读取数据。(INT 类型)
WriteDeviceRandom	随机写入数据到软元件。(INT 类型)
ReadDeviceRandom2	随机从软元件中读取数据。(SHORT 类型)
WriteDeviceRandom2	随机写入数据到软元件。(SHORT 类型)
SetDevice	设置一个软元件。(INT 类型)
GetDevice	获取一个软元件的数据。(INT 类型)
SetDevice2	设置一个软元件。(SHORT 类型)
GetDevice2	获取一个软元件的数据。(SHORT 类型)
ReadBuffer	从缓冲存储器中读取。
WriteBuffer	写入到缓冲存储器。
GetClockData	读取 PLC CPU 中的时钟数据。
SetClockData	写入时钟数据到 PLC CPU。
GetCpuType	读取 PLC CPU 类型。
SetCpuStatus	PLC CPU 的远程 RUN/STOP/PAUSE。
EntryDeviceStatus	登录软元件状态监视。
FreeDeviceStatus	撤销软元件状态监视。
OnDeviceStatus	通知事件。

(b) 使用 Microsoft® Visual Basic®6.0 或 Microsoft® Visual C++®6.0 时。

函数名	功能
Connect	连接电话线。
Open	开放通讯线。
Close	关闭通讯线。
Disconnect	断开电话线。
GetErrorMessage	显示出错定以及更正方法。
ReadDeviceBlock	批量读取软元件中数据。(INT 类型)
WriteDeviceBlock	批量写入数据到软元件。(LONG 型)
ReadDeviceBlock2	批量读取软元件中数据。(SHORT 型/INT 型)
WriteDeviceBlock2	批量写入数据到软元件。(SHORT 型/INT 型)
ReadDeviceRandom	随机从软元件中读取数据。(LONG 型)
WriteDeviceRandom	随机写入数据到软元件。(LONG 型)
ReadDeviceRandom2	随机从软元件中读取数据。(SHORT 型/INT 型)
WriteDeviceRandom2	随机写入数据到软元件。(SHORT 型/INT 型)
SetDevice	设置一个软元件。(LONG 型)
GetDevice	获取一个软元件的数据。(LONG 型)
SetDevice2	设置一个软元件。(SHORT 型/INT 型)
GetDevice2	获取一个软元件的数据。(SHORT 型/INT 型)
ReadBuffer	读取缓冲存储器
WriteBuffer	写入到缓冲存储器
GetClockData	读取 PLC CPU 中的时钟数据。
SetClockData	写入时钟数据到 PLC CPU。
GetCpuType	读取 PLC CPU 类型。
SetCpuStatus	PLC CPU 的远程 RUN/STOP/PAUSE。
EntryDeviceStatus	记录软元件状态监视。
FreeDeviceStatus	撤销软元件状态监视。
OnDeviceStatus	通知事件。

(6) 采集 Excel 上数据无需编程

使用 MX component 和 MX Sheet (SWnd5C-SHEET-E) 可以使用户只需简单设置而无需编程就在 Excel 上采集 PLC 的软元件数据。

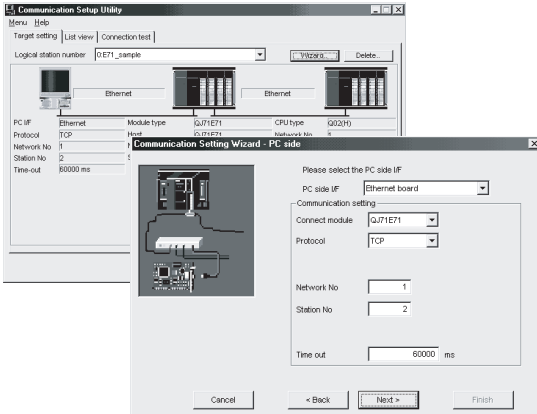
附录 9.2 MX Component 的使用步骤

本节说明了使用 MX Component 创建程序和示例程序的步骤。

(1) 创建程序的步骤

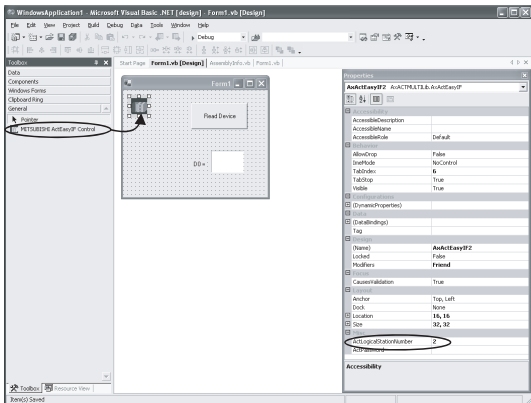
创建程序的步骤如下。

以下使用步骤以 Visual Basic® .NET 2003 作为范例。



- 1) 根据向导执行从 IBM PC/AT 兼容个人计算机到 PLC 的通讯设置。(部分控制类型只由程序设置而不使用向导。)

该向导可使用户执行通讯所需的设置，比如：逻辑站号、连接模块类型和要连接的 PLC。

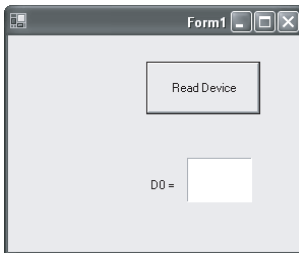


- 2) 粘贴 ACT 控制图标至表格并分配步骤 1 中设置的逻辑站号至粘贴控制的属性。

```
Private Sub Button1_Click(ByVal sender As System.Object)
    Dim rtn As Integer
    Dim iData As Integer
    rtn = AxActEasyIF1.Open()

    rtn = AxActEasyIF1.GetDevice("D0", iData)
    Label1.Text = iData
End Sub
```

- 3) 使用软件提供的函数写入读取软元件的程序。

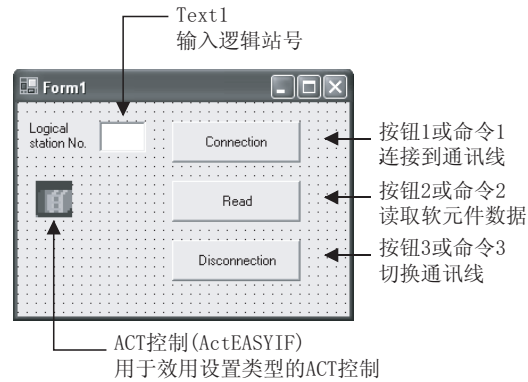


完成

(2) 示例程序

以下示例程序使用逻辑站号读取目标 PLC 的 D0 到 D4 (5 个点)。

(a) 画面范例 (Form 1)



(b) 范例程序

以下显示用于开发软件的每个范例程序。

- 1) Visual Basic®.NET 2003
- 2) Visual C++®.NET 2003
- 3) Visual Basic® 6.0
- 4) Visual C++® 6.0

1) 使用 Visual Basic®.NET 2003 时

```
Private Sub Command1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Command1.Click
' *****
' Connection
' *****
Dim rtn As Integer

' Get LogicalStationNumber
AxActEasyIF1.ActLogicalStationNumber = Val(Text1.Text)

' Connection
rtn = AxActEasyIF1.Open()
If rtn = 0 Then
    MsgBox("The Connection was successful")
Else
    MsgBox("Connection Error:" & Hex(rtn))
End If

End Sub
```

```
Private Sub Command2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Command2.Click
' *****
'Read
' *****
Dim rtn As Integer
Dim idata(5) As Short

'D0-D4 are read
rtn = AxActEasyIF1.ReadDeviceBlock2("D0", 5, idata(0))
If rtn = 0 Then
    MsgBox("D0-D4 = " & idata(0) & ", " & idata(1) & ", " & idata(2) & ", " & idata(3) & ", " &
idata(4))
Else
    MsgBox("Read Error:" & Hex(rtn))
End If

End Sub

Private Sub Command3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Command3.Click
' *****
'Disconnection
' *****
Dim rtn As Integer

'Disconnection
rtn = AxActEasyIF1.Close()
If rtn = 0 Then
    MsgBox("The disconnection was successful")
Else
    MsgBox("Disconnection Error:" & Hex(rtn))
End If

End Sub
```


2) 使用 Visual C++[®].NET 2003 时

```

//*****
//Connection
//*****
private: System::Void button1_Click(System::Object * sender, System::EventArgs * e)
{
    int iRet

    //Get LogicalStationNumber
    axActEasyIF1->ActLogicalStationNumber=Convert::ToInt32(textBox1->Text);

    //Connection
    iRet = axActEasyIF1->Open();
    if( iRet == 0 ){
        MessageBox::Show("The connection was successful");
    } else {
        MessageBox::Show( String::Format( "Connection Error:0x{0:x8} [HEX]", __box(iRet) ) );
    }
}

//*****
//Read
//*****
private: System::Void button2_Click(System::Object * sender, System::EventArgs * e)
{
    int iRet;
    short sData[5];
    String* szMessage= "";
    String* lpszarrData[];
    int iNumber;
    String* szReadData

    //D0-D4 are read
    iRet = axActEasyIF1->ReadDeviceBlock2( "D0", 5, sData );
    if( iRet == 0 ){
        lpszarrData = new String * [ 5 ];
        lpszarrData[0] = "D0-D4 = ";

        // 存储结果显示数据
        for( iNumber = 0 ; iNumber < 5 ; iNumber++ )
        {
            lpszarrData[ iNumber ] = sData[ iNumber ].ToString();
        }
        szReadData = String::Join(", ", lpszarrData);
        MessageBox::Show(String::Format("D0-D4 = {0}", szReadData));
    } else {
        MessageBox::Show( String::Format( "Read Error:0x{0:x8} [HEX]", __box(iRet) ) );
    }
}

```

```
/**/
//Disconnection
/**/
private: System::Void button3_Click(System::Object * sender, System::EventArgs * e)
{
    int iRet;

    //Disconnection
    iRet = axActEasyIF1->Close();
    if( iRet == 0 ){
        MessageBox::Show( "The disconnection was successful" );
    } else {
        MessageBox::Show( String::Format( "Disconnection Error:0x{0:x8} [HEX]", __box(iRet) ) );
    }
}
}
```

3) 使用 Visual Basic® 6.0 时

```
Private Sub Command1_Click()  
' *****  
'      Connection  
' *****  
Dim rtn As Long  
    ' Get LogicalStationNumber  
    ActEasyIF1. ActLogicalStationNumber = Val(Text1.Text)  
    ' Connection  
    rtn = ActEasyIF1. Open()  
    If rtn = 0 Then  
        MsgBox "The connection was successful"  
    Else  
        MsgBox "Connection Error:" & Hex(rtn)  
    End If  
End Sub  
  
Private Sub Command2_Click()  
' *****  
'      Read  
' *****  
Dim rtn As Long  
Dim idata(5) As Integer  
    ' D0-D4 are read  
    rtn = ActEasyIF1. ReadDeviceBlock2 ("D0", 5, idata(0))  
    If rtn = 0 Then  
        MsgBox "D0-D5 = " & idata(0) & ", " & idata(1) & ", " & idata(2) & ", " & idata(3) &  
        ", " & idata(4)  
    Else  
        MsgBox "Read Error:" & Hex(rtn)  
    End If  
End Sub  
  
Private Sub Command3_Click()  
' *****  
'      Disconnection  
' *****  
Dim rtn As Long  
    ' Disconnection  
    rtn = ActEasyIF1. Close()  
    If rtn = 0 Then  
        MsgBox "The disconnection was successful"  
    Else  
        MsgBox "Disconnection Error:" & Hex(rtn)  
    End If  
End Sub
```

4) 使用 Visual C++[®]6.0 时

```
// *****
//      Connection
// *****
void CVCDlg::OnOpen()
{
    long lRet;
    CString szMessage;

    UpdateData();
    // Get LogicalStationNumber
    m_actEasyIF. SetActLogicalStationNumber ( m_lLogicalStationNumber );
    // Connection
    lRet = m_actEasyIF. Open();
    if ( lRet == 0 ) {
        MessageBox ( "The connection was successful" )
    } else {
        szMessage. Format ( "Connection Error: %x", lRet );
        MessageBox ( szMessage )
    }
}

// *****
//      Read
// *****
void CVCDlg::OnRead()
{
    long lRet;
    short sData[5];
    CString szMessage;
    // D0-D4 are read
    lRet = m_actEasyIF. ReadDeviceBlock2 ( "D0", 5, sData );
    if ( lRet == 0 ) {
        szMessage. Format ( "D0-D5 = %d, %d, %d, %d, %d",
                           sData[0], sData[1], sData[2],
sData[3], sData[4] );
        MessageBox ( szMessage );
    } else {
        szMessage. Format ( "Read Error: %x", lRet );
        MessageBox ( szMessage )
    }
}
}
```

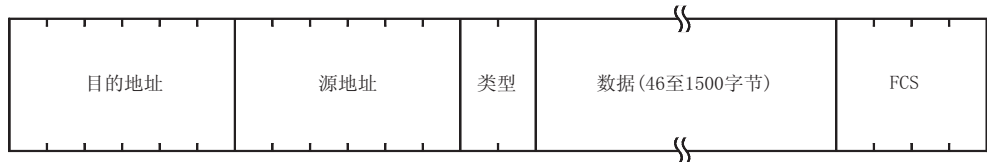
```
// *****  
//      Disconnection  
// *****  
void CVCDlg::OnOpen()  
{  
    long lRet;  
    CString szMessage;  
    // Disconnection  
    lRet = m_actEasyIF. Close();  
    if ( lRet == 0) {  
        MessageBox ( "The disconnection was successful" )  
    } else {  
        szMessage. Format ( "Disconnection Error: %x", lRet );  
        MessageBox ( szMessage )  
    }  
}
```

附录 10 以太网和 IEEE802.3 之间的差异

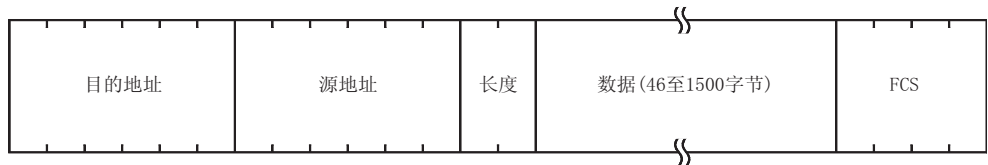
以下内容表示以太网模块支持的数据链接层中的以太网标题。

以太网数据链接层标题	以太网模块
以太网帧 (V2.0) 的规格	○
IEEE802.3 (ISO/IEC8802.3) 帧规格	○

(1) 以太网



(2) IEEE802.3



附录 11 以太网模块支持的 ICMP 协议

下表大致描述了以太网模块支持的 ICMP 类型和以太网模块执行的处理。

ICMP 类型	ICMP 名称/说明	以太网模块执行的处理
0	回声请求 请求 IP 包的回送	接收回声请求时，以太网模块发送一个响应 (回声回答)。
3	目标不能达到 无法将 IP 包传送到目的地	如果在没有连接未开放的地方接收数据，以太网模块发送一个出错响应。
8	回声请求 请求 IP 包回送	如果在缓冲存储器中设置目标存在确认，则在尝试确认目标存在时，以太网模块发送该命令。(*1)
其它	—	以太网模块忽略 (不支持)

*1 以太网模块可以同时接收两个 ICMP 回声请求 (类型 8, PING)，用来作存在确认，并对它们进行相应处理。
 当同时接收到三个或更多的 ICMP 回声请求时，第三个及后面的请求将忽略。
 当 ICMP 回声请求发送到以太网模块时，如果响应没有返回到外部设备，则再次将 ICMP 回声请求发送到以太网模块。
 以太网模块最多一次可接收 1460 字节 ICMP 信息。
 不要向以太网模块发送超过 1460 字节大小的 ICMP 信息请求。

附录 12 设置值记录表单

本节提供了用 GX Developer 设置参数的设置值记录表单。如有需要，可以制作一份副本。

设置值记录表单编号	GX Developer 设置画面
记录表单 1	设置 MNET/10H 以太网卡号的网络参数 以太网运行
记录表单 2	设置以太网初始化的网络参数
记录表单 3	设置以太网开放的网络参数
记录表单 4	设置以太网路由中继参数的网络参数
记录表单 5	设置以太网站号<->IP 关联信息的网络参数
记录表单 6	设置以太网 FTP 参数的网络参数
记录表单 7	设置以太网电子邮件的网络参数 设置以太网邮件地址
记录表单 8	设置以太网消息
记录表单 9	设置以太网中断的网络参数
记录表单 10	设置智能功能模块中断指针
记录表单 11	设置 MNET/10 以太网路由中继参数的网络参数
记录表单 12	设置远程口令
记录表单 13	网络参数以太网冗余设置

记录表单 1

[模块编号]

GX Developer 设置画面	数据项	设置数据		
		设置值	备注	
设置 MNET/10H 以太网卡号的网络参数	起始 I/O 地址			
	网络号			
	组号			
	站号			
	模式	<input type="checkbox"/> 在线 <input type="checkbox"/> 离线 <input type="checkbox"/> 自抑制测试 <input type="checkbox"/> 硬件(H/W)测试		
以太网操作	通讯数据代码设置		<input type="checkbox"/> 二进制代码通讯	
			<input type="checkbox"/> ASCII 代码通讯	
	初始时间设置		<input type="checkbox"/> 不等待打开(不可以在停止时间进行通讯)	
			<input type="checkbox"/> 始终等待打开(可以在停止时间进行通讯)	
	IP 地址设置	输入格式	<input type="checkbox"/> 十进制 <input type="checkbox"/> 十六进制	IP 地址 输入格式
		IP 地址		调节为输入格式
	发送帧设置		<input type="checkbox"/> 以太网(V2.0)	
			<input type="checkbox"/> IEEE802.3	
运行中允许写入		<input type="checkbox"/> 运行中允许写入	允许: 选中复选框	
TCP 存在确认设置		<input type="checkbox"/> 使用 KeepAlive		
		<input type="checkbox"/> 使用 Ping		

记录表单 2

[模块编号]

GX Developer 设置画面	数据项		设置数据	
			设置值	备注
设置以太网初始化的网络参数	定时器设置	TCP ULP 定时器		默认: 60(500 ms)
		TCP 零窗口定时器		默认: 20(500 ms)
		TCP 再次发送定时器		默认: 20(500 ms)
		TCP 结束定时器		默认: 40(500 ms)
		IP 组合定时器		默认: 10(500 ms)
		响应监视定时器		默认: 60(500 ms)
		目标存在确认开始间隔		默认: 1200(500 ms)
		目标存在确认间隔定时器		默认: 20(500 ms)
		目标存在确认再次发送定时器		默认: 3(次)
	DNS 设置	输入格式	十进制	DNS 服务器 IP 地址 输入格式
			十六进制	
		DNS 服务器 1 IP 地址		调节为输入格式
		DNS 服务器 2 IP 地址		
		DNS 服务器 3 IP 地址		
DNS 服务器 4 IP 地址				

记录表单 3

[模块编号]

GX Developer 设置画面	数据项		设置数据		
			设置值	备注	
设置以太网开放的 网络参数	连接号	协议	<input type="checkbox"/>	TCP	如果选择 UDP, 则无需设置
			<input type="checkbox"/>	UDP	
		开放系统	<input type="checkbox"/>	主动	
			<input type="checkbox"/>	非被动	
			<input type="checkbox"/>	完全被动	
		固定缓冲存储器	<input type="checkbox"/>	发送	
			<input type="checkbox"/>	接收	
		固定缓冲存储器通讯	<input type="checkbox"/>	有顺序	
			<input type="checkbox"/>	无顺序	
		建立成对开放	<input type="checkbox"/>	不成对	
			<input type="checkbox"/>	成对	
		存在确认	<input type="checkbox"/>	未确认	
	<input type="checkbox"/>		确认		
	本地站端口号			输入格式: 十六进制	
	目标 IP 地址	<input type="checkbox"/>	十进制	目标 IP 地址输入格式	
		<input type="checkbox"/>	十六进制		
				调节为输入格式	
	目标端口号			输入格式: 十六进制	
	连接号	协议	<input type="checkbox"/>	TCP	
			<input type="checkbox"/>	UDP	
		开放系统	<input type="checkbox"/>	主动	
			<input type="checkbox"/>	非被动	
			<input type="checkbox"/>	完全被动	
		固定缓冲存储器	<input type="checkbox"/>	发送	
<input type="checkbox"/>			接收		
固定缓冲存储器通讯		<input type="checkbox"/>	有顺序		
		<input type="checkbox"/>	无顺序		
建立成对开放		<input type="checkbox"/>	不成对		
		<input type="checkbox"/>	成对		
存在确认		<input type="checkbox"/>	未确认		
	<input type="checkbox"/>	确认			
本地站端口号			输入格式: 十六进制		
目标 IP 地址	<input type="checkbox"/>	十进制	目标 IP 地址输入格式		
	<input type="checkbox"/>	十六进制			
			调节为输入格式		
目标端口号			输入格式: 十六进制		

记录表单 7

[模块编号]

GX Developer 设置画面	数据项		设置数据			
			设置值	备注		
设置以太网电子邮件的网络参数	一般设置	口令				
		邮件地址				
		接收邮件的检查	<input type="checkbox"/>	检查接收的邮件	检查: 检查标记	
		查询的时间间隔	<input type="checkbox"/>	秒	设置检查接收邮件的间隔时间 选择检查接收邮件的时间间隔单位	
	<input type="checkbox"/>		分钟			
	<input type="checkbox"/>		小时			
	邮件服务器名称	发送设置	SMTP 服务器名称			
			IP 地址	<input type="checkbox"/>	十进制	IP 地址输入格式
		<input type="checkbox"/>		十六进制		
		POP 服务器名称				调节为输入格式
			IP 地址	<input type="checkbox"/>	十进制	IP 地址输入格式
		<input type="checkbox"/>		十六进制		
	设置以太网发送邮件地址	编号	发送邮件地址			
			消息设置	<input type="checkbox"/>	未执行 执行消息	
编号		发送邮件地址				
		消息设置	<input type="checkbox"/>	未执行 执行消息		
编号		发送邮件地址				
		消息设置	<input type="checkbox"/>	未执行 执行消息		
编号		发送邮件地址				
		消息设置	<input type="checkbox"/>	未执行 执行消息		
编号		发送邮件地址				
		消息设置	<input type="checkbox"/>	未执行 执行消息		

记录表单 8

[模块编号]

GX Developer 设置画面	数据项	设置数据		
		设置值	备注	
设置以太网消息	PLC 检查的条件	<input type="checkbox"/>	无设置	
		<input type="checkbox"/>	正常停止	
		<input type="checkbox"/>	模块错误/模块系统错误 (中等/严重无序)	
		<input type="checkbox"/>	警告停止(少量无序停止)	
		<input type="checkbox"/>	正常运行	
		<input type="checkbox"/>	警告运行(少量无序运行)	
		<input type="checkbox"/>	暂停	
	发送方法	<input type="checkbox"/>	发送附件	
		<input type="checkbox"/>	发送文本邮件	
	附件格式	<input type="checkbox"/>	二进制	调节为输入格式
		<input type="checkbox"/>	ASCII	
		<input type="checkbox"/>	CSV	
	附件名称	<input type="checkbox"/>		
	PLC 查询间隔	<input type="checkbox"/>		CPU 查询数值设置的时间间隔
		<input type="checkbox"/>	秒	CPU 查询时间单位选择的间隔时间
		<input type="checkbox"/>	分钟	
	监视数值输入格式	<input type="checkbox"/>	十进制	监视值输入格式
		<input type="checkbox"/>	十六进制	
	编号	条件软元件	<input type="checkbox"/>	
		检测条件	<input type="checkbox"/>	
		监视值	<input type="checkbox"/>	调节为输入格式
		消息数据存储软元件	<input type="checkbox"/>	
		消息数据	<input type="checkbox"/>	
	编号	条件软元件	<input type="checkbox"/>	
		检测条件	<input type="checkbox"/>	
		监视值	<input type="checkbox"/>	调节为输入格式
		消息数据存储软元件	<input type="checkbox"/>	
		消息数据	<input type="checkbox"/>	
	编号	条件软元件	<input type="checkbox"/>	
		检测条件	<input type="checkbox"/>	
监视值		<input type="checkbox"/>	调节为输入格式	
消息数据存储软元件		<input type="checkbox"/>		
消息数据		<input type="checkbox"/>		

记录表单 9

[模块编号]

GX Developer 设置画面	数据项	设置数据		备注	
		设置值			
设置以太网中断的网络参数	输入格式		十进制	字软件设置值 输入格式	
			十六进制		
			REC V 指令		
	编号	软件代码		固定缓冲存储器	
		软件编号		-	不需要设置
		检测方法		边沿检测	自动设置
		中断条件		扫描完成	自动设置
		字软件设置值		-	不需要设置
		板号			输入格式: 十进制
		中断(SI)编号			输入格式: 十进制
		编号	软件代码		REC V 指令
				固定缓冲存储器	
	软件编号			-	不需要设置
	检测方法			边沿检测	自动设置
	中断条件			扫描完成	自动设置
	字软件设置值			-	不需要设置
	板号				输入格式: 十进制
	中断(SI)编号				输入格式: 十进制
	编号	软件代码		REC V 指令	
				固定缓冲存储器	
		软件编号		-	不需要设置
		检测方法		边沿检测	自动设置
		中断条件		扫描完成	自动设置
		字软件设置值		-	不需要设置
		板号			输入格式: 十进制
		中断(SI)编号			输入格式: 十进制
	编号	软件代码		REC V 指令	
				固定缓冲存储器	
		软件编号		-	不需要设置
		检测方法		边沿检测	自动设置
中断条件			扫描完成	自动设置	
字软件设置值			-	不需要设置	
板号				输入格式: 十进制	
中断(SI)编号				输入格式: 十进制	

记录表单 10

GX Developer 设置画面	数据项		设置数据		
			设置值	备注	
设置智能功能模块的中断指针	编号	PLC CPU 端	中断指针起始号		输入格式: 十进制
			模块的中断指针号		
		智能模块端	起始 I/O 地址		输入格式: 十六进制
			起始 SI 号		
	编号	PLC CPU 端	中断指针起始号		输入格式: 十进制
			模块的中断指针号		
		智能模块端	起始 I/O 地址		输入格式: 十六进制
			起始 SI 号		
	编号	PLC CPU 端	中断指针起始号		输入格式: 十进制
			模块的中断指针号		
		智能模块端	起始 I/O 地址		输入格式: 十六进制
			起始 SI 号		
	编号	PLC CPU 端	中断指针起始号		输入格式: 十进制
			模块的中断指针号		
		智能模块端	起始 I/O 地址		输入格式: 十六进制
			起始 SI 号		输入格式: 十进制

记录表单 11

GX Developer 设置画面	数据项		设置数据	
			设置值	备注
设置 MNET/10H 以太网路由中继参数的网络参数	编号	传送目标网络号		输入格式：十进制
		中继目标网络号		
		中继目标站号		
	编号	传送目标网络号		
		中继目标网络号		
		中继目标站号		
	编号	传送目标网络号		
		中继目标网络号		
		中继目标站号		
	编号	传送目标网络号		
		中继目标网络号		
		中继目标站号		
	编号	传送目标网络号		
		中继目标网络号		
		中继目标站号		
	编号	传送目标网络号		
		中继目标网络号		
		中继目标站号		

记录表单 12

GX Developer 设置画面	数据项		设置数据		
			设置值	备注	
设置远程口令	口令设置				
	口令有效模块设置	型号名称	QJ71E71		
		起始 X/Y			
远程口令 详细设置	用户连接号			1号连接	选择目标连接
				2号连接	
				3号连接	
				4号连接	
				5号连接	
				6号连接	
				7号连接	
				8号连接	
				9号连接	
				10号连接	
				11号连接	
				12号连接	
				13号连接	
				14号连接	
				15号连接	
				16号连接	
	系统连接			自动打开 UDP 端口	
				FTP 传送端口 (TCP/IP)	
				GX Developer 传送端口 (TCP/IP)	
				GX Developer 传送端口 (UDP/IP)	
			HTTP 端口		

记录表单 13

[模块编号]

GX Developer 设置画面	数据项		设置数据		
			设置值	备注	
冗余设置	站号和模式设置 (系统B)	站号			
		模式		在线	
				离线	
				自回路回送测试	
	IP 地址设置	输入格式		H/W 测试	
				十进制	
		系统B		十六进制	IP 地址输入格式
					调节为输入格式
		电缆断开超时时发送系统切换		电缆断开超时时发送系统切换	发送: 检查标记
					默认: 2.0s
		通讯出错时发送系统切换		通讯出错时发送系统切换	发送: 检查标记
		通讯出错时设置系统切换		1号连接	选择目标连接
			2号连接		
			3号连接		
			4号连接		
			5号连接		
			6号连接		
			7号连接		
			8号连接		
			9号连接		
			10号连接		
			11号连接		
			12号连接		
			13号连接		
	14号连接				
	15号连接				
	16号连接				
		自动打开 UDP 端口			
		建立 GX Developer UDP 通讯端口			
		建立 GX Developer TCP 通讯端口			
		建立 FIP 通讯端口			
		建立 HTTP 通讯端口			

索引

[1 至 10]

- 100BASE-TX 2-3、4-8
- 10BASE2 2-7、4-10
- 10BASE5 2-5、4-9
- 10BASE-T 2-4、4-8

[A]

- AJ71E71 附录-4
- AJ71E71-S3 附录-4
- ARP 1-15、5-42
- AUI 电缆(收发器电缆) 2-5
- A 兼容 1E 帧..... 6- 2
 - (结束代码) 11-24
 - (异常代码) 11-24

[B]

- BUFRCVS 指令..... 7-7、8-8、10-5
- BUFRVCV 指令..... 7-5、8-6、10-2
- BUFSND 指令..... 7-3、8-4、10-8
- 被动开放 5-37、5-42、5-52
- 本地站(以太网模块) IP 地址 4-20
- 本地站端口号(设置) 5-39
- 标题地址 9-8、9-15
- 标题
 - 固定缓冲器通讯(无顺序) 8-10
 - 固定缓冲器通讯(有顺序) 7-12
 - 使用随机访问缓冲器通讯 9-5

[C]

- 参数设置
 - 设置画面列表 4-13
 - 设置项目列表 3-9
 - 设置值记录表 附录-88
- 操作设置 4-19
- 测试
 - PING 指令测试..... 5-34
 - 环路测试 5-29
 - 硬件测试(H/W 测试) 4-23
 - 自抑制测试 4-22
- 成对开放 5-63
- 程序范例
 - 固定缓冲器通讯(无顺序) 8-18
 - 固定缓冲器通讯(有顺序) 7-18

- 开放/关闭处理(被动)..... 5-57
- 开放/关闭处理(主动)..... 5-49
- 出错代码 11-25
- 出错日志区 11-15
- 初始化处理 5-3
- 初始化定时设置 4-20
- 初始化设置 5-4
- 存储器 3-12
- 存在确认 5-36

[D]

- DNS 1-15
- DNS 设置..... 5-7
- 定时器设置 5-4
- 端口 5-40、5-41
- 多 CPU 系统 1-8、2-8、6-4

[E]

- ERRCLR 指令 10-14、11-4
- ERRRD 指令 10-17、11-4

[F]

- FTP 1-15
- FTP 参数 4-14
- 访问不当 5-68
- 非被动 5-37、5-52
- 副标题
 - 固定缓冲器通讯(有顺序)..... 7-14
 - 使用随机访问缓冲器通讯..... 9-7

[G]

- GX Developer..... A-21
 - 设置项目列表 3-9
 - 设置值纪录表 附录-88
- 固定缓冲器(设置)..... 5-38
- 固定缓冲器通讯(设置)..... 5-38
- 固定缓冲器通讯(无顺序)..... 第 8 章
- 固定缓冲器通讯(有顺序)..... 第 7 章
- 关闭处理 5-44
- 关闭指令 5-44、10-11
- 规格 第 3 章

[H]	缓冲存储器列表	3-13
[I]	ICMP	1-15、附录-87
	IP	1-15
	IP 地址设置	
	本地站 IP 地址	4-20
	目标 IP 地址	5-40
[J]	监视通讯	1-6
	结束代码	
	固定缓冲器通讯(有步骤)	7-16
	结束代码列表	11-22
	使用随机访问缓冲器通讯	9-10
[K]	开放处理	
	被动开放处理	5-52
	成对开放处理	5-63
	主动开放处理	5-45
	开放设置	5-36
	成对开放设置	5-64
	同时广播发送	8-11
	同时广播接受	8-13
	开放系统	5-37
	开始操作前的设置和顺序	4-3
	可访问范围	
	固定缓冲器通讯(无顺序)	8-1
	固定缓冲器通讯(有顺序)	7-1
	使用随机访问缓冲器通讯	9-15
[L]	LED 显示器	4-6、11-9
	路由器 IP 地址	5-20
	路由中继参数	5-16
	路由中继功能	5-17
	逻辑地址	9-15
[M]	MAC 地址	A-21
	MELSECNET/H、MELSECNET/10 继电器通讯 ...	3-6
	MX Component	6-5、附录-75
	模式	4-18

	默认路由器 IP 地址	5-18
	目标 IP 地址	5-40、5-42
	目标端口号	5-41
[O]	OPEN 指令	5-43、10-19
[P]	PLC CPU 处于 STOP 状态时通讯	4-20
	POP3	1-15
[Q]	QE71	附录-4
	QnA/A 系列模块间的功能比较	附录-4
	QnA 兼容 3E 帧或 4E 帧	6-2
	其它站访问时的有效模块	4-17
	启动 I/O 号	4-17
[R]	RUN 时允许写入	4-21
	冗余系统	1-11、5-89
[S]	SMTIP	1-15
	设置卡数	4-13、4-16
	使用 MC 协议通讯	第 6 章
	使用 MELSEC 通讯协议的通讯	1-2
	使用电子邮件通讯	1-4
	使用随机访问缓冲器通讯	第 9 章
	适用数据	
	固定缓冲器通讯(无顺序)	8-10
	固定缓冲器通讯(有顺序)	7-13
	使用随机访问缓冲器通讯	9-6
	适用系统	2-1
	输入/输出信号列表	3-10
	数据长度	
	固定缓冲器通讯(有顺序)	7-15
	使用随机访问缓冲器通讯	9-8
	数据格式	
	固定缓冲器通讯(无顺序)	8-10
	固定缓冲器通讯(有顺序)	7-12
	使用随机访问缓冲器通讯	9-5
[T]	TCP	1-14、5-37
	通讯步骤	5-1

通讯参数 3-14、5-36
 通讯方法设置(TCP/UDP)..... 5-37
 通讯数据代码 4-19
 通讯状态存储区 3-15
 同时广播 8-11
 同轴电缆的接头连接 4-11

[U]

UDP 1-15、5-36
 UINI 指令 5-13、10-23

[W]

完全被动开放 5-37、5-52
 网络参数
 MNET/10H 以太网卡数 4-16
 网络号 4-17
 网络类型 4-17
 物理地址 9-15

[X]

系统配置 第 2 章
 相关手册 A-13
 响应形式
 固定缓冲器通讯(有顺序) 7-13
 使用随机访问缓冲器通讯 9-6
 协议 5-37

[Y]

以太网 1-1
 以太网地址 A-21
 以太网模块 A-21、1-1
 异常代码 11-24
 用法设置 3-14、5-36
 远程口令出错通知累计数 5-85
 远程口令确认 5-68

[Z]

站号 4-18
 正文(指令)
 固定缓冲器通讯(有步骤) 7-15
 使用随机访问缓冲器通讯 9-9
 中断功能
 固定缓冲器通讯(无顺序) 8-8
 固定缓冲器通讯(有顺序) 7-7
 中断设置 7-7

中断指针设置 7-9
 重新初始化处理 5-9
 主动开放 5-37、5-42、5-45
 专用指令列表 3-8
 专有名词和简写 A-21
 子网地址 5-18
 子网掩码样式 5-17
 自动开放 UDP 端口 5-66
 组号 4-17

质保

使用之前敬请确认下述产品质保的内容。

1. 免费质保期限和免费质保范围

如果是在质保期内使用本产品时发现因三菱电机的责任而导致的异常或缺陷下文简称为“故障”，则由经销商或三菱电机维修公司免费维修。

但如果需要到国外或偏远地方出差修理时，则要收取派遣技术人员所需的费用。

[免费质保期]

本产品的免费质保期为一年，自购买或货到目的地的日期起算。

但从出厂开始最长分销时间不得超过 6 个月，从制造之日开始的最长免费质保期不得超过 18 个月。修理产品的免费质保期不得超过修理以前的免费质保期。

[免费质保范围]

- (1) 仅限于是在按照使用手册、用户手册和产品上的警示标贴上规定的使用状态、使用方法和使用环境等正常使用的条件下。
- (2) 即使在免费质保期内，以下情况也属于有偿修理。
 1. 因用户的不合理存储或搬运、大意或疏忽而导致的故障以及因用户的硬件或软件设计而导致的故障。
 2. 因用户在三菱电机不知情的情况下对该产品进行改造而引起的故障。
 3. 三菱电机产品被组装到用户的设备上时，如果用户的设备配备了相应法规所要求的安全装置或按行业惯例应具备的功能-构造等则可以避免的故障。
 4. 使用说明书上指定的消耗部件(电池、背光灯、保险丝等)得到正常维护和更换便可防止的故障。
 5. 因火灾、不正常电压和因地震、雷电、大风和水灾等不可抗力引发的故障。
 6. 以三菱电机交货时的科学技术水准不可能预见的事由导致的故障。
 7. 非三菱电机责任或用户承认的非三菱电机责任的其它故障。

2. 停止生产该产品后的有偿修理条款

(1) 三菱电机在本产品停止生产后的 7 年内受理对该产品的有偿修理。

停止生产的信息将以三菱电机技术公告等方式予以通知。

(2) 生产停止以后不再提供产品(包括备件)。

3. 海外服务

在海外，修理由三菱电机在当地的海外 FA 中心受理。请注意各个 FA 中心的修理条件等可能会有所不同。

4. 机会损失和间接损失不在质保责任范围内

无论是否处于无偿保证期间内，对于非三菱电机责任的事由导致的损害，三菱电机产品故障导致的用户的机会损失和利益损失，与三菱电机预见与否无关的特别情况导致的损害、间接损害、事故赔偿、三菱电机产品以外的损坏以及其它业务的赔偿，三菱电机概难负责。

5. 产品规格的改变

产品目录、手册或技术资料上记载的规格可能会在未通知的情况下进行变更，敬请谅解。

6. 产品应用

(1) 使用三菱电机 MELSEC 可编程逻辑控制器时，应满足以下使用条件: 只应用于即使可编程逻辑控制器发生故障-异常等也不会导致重大事故的用途，以及在发生故障-异常时设备外部将系统地实施备份及失效安全功能。

(2) 三菱电机的图形操作终端是以一般工业等用途为对象而设计-制造的通用产品。因此，不适用于各电力公司的核电站以及其它发电站等对公众有较大影响的用途、以及各铁路公司和国防部门等要求特别质量保证体制的用途。

此外，三菱电机的可编程控制器也不适用于航空、医疗、焚烧、燃料装置、载人搬运装置、娱乐设备、安全设备等对人身和财产有较大影响的用途。

但是，如果事先与三菱电机协商，用户承认限定用途且无特别质量要求时，交换必要的文件后也可以用于上述用途。

Microsoft、Windows、WindowsNT 是 Microsoft Corporation 在美国以及其它国家的注册商标。
Netscape 是 Netscape Communication Corporation 在美国以及其它国家的注册商标。
Adobe、Acrobat 是 Adobe Systems Incorporated 的注册商标。
Pentium、Celeron 是 Intel Corporation 在美国以及其它国家的商标及注册商标。
Ethernet 是美国 Xerox. co. ltd 的注册商标。
本手册中使用的其它的公司名称和产品名称是相应公司的商标和注册商标。

Q系列以太网接口模块

用户手册(基础篇)

技术服务热线:

800-828-9910

服务时间: **9:00~12:00**

13:00~17:00 (节假日除外)

三菱电机自动化(上海)有限公司

地址: 上海市黄浦区新昌路80号智富广场4楼

邮编: 200003

电话: 021-61200808 传真: 021-61212444

网址: www.mitsubishielectric-automation.cn

书号	SH(NA)-080235C-B(0611)STC
印号	STC-QEthernet(B)-UM(0611)

内容如有更改
恕不另行通知