

mitsubishi

三菱可编程控制器

MELSEC **Q** 系列

MELSEC **L** 系列

MELSEC-Q/L MELSEC通讯协议 参考手册

QSERIES L SERIES

产品型号

-QJ71C24N

-QJ71C24N-R2

-QJ71C24N-R4

-QJ71C24

-QJ71C24N-R2

-QJ71E71-100

-QJ71E71-B5

-QJ71E71-B2

-LJ71C24

-LJ71C24-R2

● 安全注意事项 ●

(使用之前请务必阅读)

在使用本产品之前，应仔细阅读本手册及本手册中介绍的关联手册，同时在充分注意安全的前提下正确地操作。

本手册中的注意事项仅记载了 MELSEC 通信协议的相关内容。关于可编程控制器系统安全方面的注意事项请参阅所使用的 CPU 模块的用户手册。

在“安全注意事项”中，安全注意事项被分为“警告”和“注意”这二个等级。



警告

表示错误操作可能造成危险后果，导致死亡或重伤事故。



注意

表示错误操作可能造成危险后果，导致中度伤害、轻伤及设备损失。

注意根据情况不同，即使“注意”这一级别的事项也有可能引发严重后果。

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至关重要的。

请妥善保管本指南以备需要时阅读，并应将本手册交给最终用户。

[设计注意事项]

警告

- 如果把外部设备连接到 CPU 模块，或者将个人计算机等的外部设备连接到智能功能模块在运行中对可编程控制器进行控制(数据更改)时，应在程序中配置互锁电路，确保整个系统始终都会安全运行。

此外，在对运行中的可编程控制器执行其它控制(程序更改、运行状态更改(状态控制))之前，应仔细阅读手册并充分确认安全。

尤其是从外部设备对远程的可编程控制器进行上述控制时，由于数据通讯异常，可能不能对可编程控制器的故障立即采取措施。

应在程序中配置互锁电路的同时，预先在外部设备与 CPU 模块之间确定发生数据通信异常时系统方面的处理方法等。

- 不要对智能功能模块的缓冲存储器的“系统区”进行数据写入。

此外，在从 CPU 模块至智能功能模块的输出信号中，不要对标为“禁止使用”的信号进行输出(ON)操作。

如果对“系统区”进行了数据写入，对“禁止使用”的信号进行了输出，有可能导致可编程控制器系统误动作。

[运行时的注意事项]

注意

- 将个人计算机等的外部设备连接到智能功能模块在运行中对可编程控制器进行控制(特别是数据更改、程序更改、运行状态更改(状态控制))时,应仔细阅读用户手册并充分确认安全。
如果数据更改、状态控制错误,有可能导致系统误动作,造成设备损坏或引发事故。
- 将缓冲存储器的设置值登录到模块内的快闪卡中使用时,在登录过程中不要进行模块安装站的电源 OFF 以及 CPU 模块的复位操作。
如果在登录过程中进行了模块安装站的电源 OFF 以及 CPU 模块的复位操作,快闪卡内的数据内容将变为不定值,需要对至缓冲存储器的设置值进行重新设置后登录到快闪卡中。
否则有可能导致模块故障以及误动作。

●关于产品的应用●

(1) 在使用三菱可编程控制器时，应该符合以下条件:即使在可编程控制器设备出现问题或故障时也不会导致重大事故，并且应在设备外部系统地配备能应付任何问题或故障的备用设备及失效安全功能。

(2) 三菱可编程控制器是以一般工业用途等为对象设计和制造的通用产品。因此，三菱可编程控制器不应用于以下设备・系统等特殊用途。

如果用于以下特殊用途，对于三菱可编程控制器的质量、性能、安全等所有相关责任(包括但不限于债务未履行责任、瑕疵担保责任、质量保证责任、违法行为责任、制造物责任)，三菱电机将不负责。

- 面向各电力公司的核电站以及其它发电厂等对公众有较大影响的用途。
- 用于各铁路公司或公用设施目的等有特殊质量保证体系要求的用途。
- 航空航天、医疗、铁路、焚烧・燃料装置、载人移动设备、载人运输装置、娱乐设备、安全设备等预计对人身财产有较大影响的用途。

然而，对于上述应用，如果在限定于具体用途，无需特殊质量(超出一般规格的质量等)要求的条件下，经过三菱电机的判断也可以使用三菱可编程控制器，详细情况请与当地三菱电机代表机构协商。

修订记录

*本手册号在封底的左下角。

印刷日期	*手册编号	修改内容
2003 年 03 月	SH(NA)-080414CHN-A	第一版
2011 年 04 月	SH(NA)-080414CHN-B	第二版 全面改版
2012 年 10 月	SH(NA)-080414CHN-C	第三版 全面改版

日文手册原稿: SH-080003-V

本手册不授予工业产权或任何其它类型的权利，也不授予任何专利许可。三菱电机对由于使用了本手册中的内容而引起的涉及工业产权的任何问题不承担责任。

© 2003 MITSUBISHI ELECTRIC CORPORATION

前言

在此感谢贵方购买了三菱可编程控制器 MELSEC-Q/L 系列的产品。

本手册是用于让用户了解使用 MELSEC-Q/L 系列可编程控制器时必要的功能等有关内容的手册。

在使用之前应熟读本手册及关联手册，在充分了解 MELSEC-Q/L 系列可编程控制器的功能・性能的基础上正确地使用本产品。

请将本手册交付至最终用户。

目录

安全注意事项.....	A - 1
关于产品的应用.....	A - 3
修订记录	A - 4
前言	A - 5
目录	A - 5
关联手册	A - 11
手册的阅读方法・结构.....	A - 13
总称・略称	A - 15
术语	A - 17

1 概要

1 - 1 到 1 - 4

1.1 MELSEC 通信协议的概要	1 - 1
1.2 MELSEC 通信协议的特点	1 - 3

2 关于通过 MELSEC 通信协议进行的数据通信

2 - 1 到 2 - 35

2.1 数据通信用帧的类型及用途.....	2 - 1
2.2 各数据通信用帧的允许访问范围.....	2 - 4
2.3 MC 协议的控制步骤的思路	2 - 6
2.4 可编程控制器 CPU 侧访问时机.....	2 - 7
2.5 用于对可编程控制器 CPU 进行运行中写入的设置方法	2 - 9
2.6 关于其它站访问.....	2 - 10
2.6.1 可进行其它站访问的可编程控制器模块	2 - 10
2.6.2 使用各帧时的可访问站的示例	2 - 13
2.6.3 各数据通信帧内指定的其它站访问用数据项目的指定示例	2 - 17
2.7 数据通信时的注意事项.....	2 - 20
2.7.1 使用 E71 时.....	2 - 20
2.7.2 使用 C24 时.....	2 - 22
2.8 串行通信模块的传送顺控程序的时序图及通信时间	2 - 23
2.9 经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 进行其它站访问时的传送时间.....	2 - 25
2.10 与多 CPU 系统的兼容.....	2 - 28
2.11 与 Q00CPU、Q01CPU 的串行通信功能的兼容	2 - 32
2.12 与以太网端口内置 CPU 的兼容.....	2 - 34

3.1 报文格式.....	3 - 2
3.1.1 指令说明项的阅读方法.....	3 - 2
3.1.2 QNA 兼容 3E 帧、4E 帧的报文格式、控制步骤.....	3 - 3
3.1.3 QNA 兼容 3E 帧、4E 帧的数据指定项目的内容.....	3 - 13
3.1.4 QNA 兼容 3C 帧的控制步骤、报文格式.....	3 - 16
3.1.5 QNA 兼容 4C 帧的控制步骤、报文格式.....	3 - 25
3.1.6 QNA 兼容 3C/4C 帧的数据指定项目的内容.....	3 - 36
3.1.7 字符部分的传送数据的思路.....	3 - 48
3.2 QNA 兼容 3E/3C/4C 帧、4E 帧用指令及功能列表.....	3 - 53
3.3 软元件存储器的读取、写入.....	3 - 61
3.3.1 指令、字符部分的内容及软元件范围.....	3 - 61
3.3.2 多个块批量读取、批量写入.....	3 - 74
3.3.3 位单位的批量读取(指令: 0401).....	3 - 83
3.3.4 位单位的批量写入(指令: 1401).....	3 - 85
3.3.5 位单位的随机写入(测试)(指令: 1402).....	3 - 87
3.3.6 字单位的批量读取(指令: 0401).....	3 - 89
3.3.7 字单位的批量写入(指令: 1401).....	3 - 93
3.3.8 字单位的随机写入(测试)(指令: 1402).....	3 - 96
3.3.9 字单位的随机读取(指令: 0403).....	3 - 98
3.3.10 软元件存储器的监视.....	3 - 115
3.4 缓冲存储器的读取、写入.....	3 - 124
3.4.1 关于指令及缓冲存储器.....	3 - 124
3.4.2 缓冲存储器的读取(指令: 0613).....	3 - 126
3.4.3 至缓冲存储器的写入(指令: 1613).....	3 - 128
3.5 智能功能模块的缓冲存储器的读取、写入.....	3 - 129
3.5.1 关于指令及缓冲存储器.....	3 - 129
3.5.2 可以访问的智能功能模块.....	3 - 134
3.5.3 智能功能模块的缓冲存储器读取(指令: 0601).....	3 - 137
3.5.4 智能功能模块的缓冲存储器写入(指令: 1601).....	3 - 139
3.6 可编程控制器 CPU 的状态控制.....	3 - 140
3.6.1 指令、控制内容及字符部分的内容.....	3 - 140
3.6.2 远程 RUN(指令: 1001).....	3 - 142
3.6.3 远程 STOP(指令: 1002).....	3 - 143
3.6.4 远程 PAUSE(指令: 1003).....	3 - 144
3.6.5 远程 RESET(指令: 1006).....	3 - 145
3.6.6 远程锁存清除(指令: 1005).....	3 - 146
3.6.7 CPU 型号读取(指令: 0101).....	3 - 147
3.7 驱动器内存的整理(其它站 QNACPU 用).....	3 - 149
3.7.1 指令及字符部分的内容.....	3 - 150
3.7.2 驱动器内存使用状态的读取(指令: 0205).....	3 - 153
3.7.3 驱动器内存的碎片整理(指令: 1207).....	3 - 154
3.8 文件控制.....	3 - 155
3.8.1 指令及字符部分的内容.....	3 - 155
3.8.2 文件控制时的注意事项.....	3 - 170

3.8.3 用于 Q/LCPU 文件控制的执行步骤	3 - 172
3.8.4 用于 QNACPU 文件控制的执行步骤	3 - 177
3.8.5 目录 • 文件信息的读取 (指令: 1810): 用于 Q/LCPU	3 - 181
3.8.6 目录 • 文件信息的搜索 (指令: 1811): 用于 Q/LCPU	3 - 183
3.8.7 文件的打开 (指令: 1827): 用于 Q/LCPU	3 - 184
3.8.8 文件的关闭 (指令: 182A): 用于 Q/LCPU	3 - 186
3.8.9 文件的读取 (指令: 1828): 用于 Q/LCPU	3 - 187
3.8.10 文件的新建 (指令: 1820): 用于 Q/LCPU	3 - 189
3.8.11 至文件的写入 (指令: 1829): 用于 Q/LCPU	3 - 190
3.8.12 文件的删除 (指令: 1822): 用于 Q/LCPU	3 - 192
3.8.13 文件的复制 (指令: 1824): 用于 Q/LCPU	3 - 194
3.8.14 文件创建日期的更改 (指令: 1826): 用于 Q/LCPU	3 - 197
3.8.15 文件属性的更改 (指令: 1825): 用于 Q/LCPU	3 - 198
3.8.16 文件信息列表的读取: 用于 QNACPU	3 - 200
3.8.17 文件有无的读取 (文件搜索) (指令: 0203): 用于 QNACPU	3 - 206
3.8.18 文件锁定的登录、解除 (指令: 0808): 用于 QNACPU	3 - 208
3.8.19 文件内容的读取 (指令: 0206): 用于 QNACPU	3 - 210
3.8.20 文件的新建 (文件名登录) (指令: 1202): 用于 QNACPU	3 - 212
3.8.21 至文件的写入 (指令: 1203): 用于 QNACPU	3 - 214
3.8.22 文件的删除 (指令: 1205): 用于 QNACPU	3 - 218
3.8.23 文件的复制 (指令: 1206): 用于 QNACPU	3 - 220
3.8.24 文件信息的更改 (指令: 1204): 用于 QNACPU	3 - 222
3.9 用户登录帧的登录、删除、读取: 用于串行通信模块	3 - 228
3.9.1 指令及字符部分的内容	3 - 228
3.9.2 用户登录帧的登录、删除 (指令: 1610)	3 - 231
3.9.3 用户登录帧的读取 (指令: 0610)	3 - 233
3.10 全局功能: 用于串行通信模块	3 - 235
3.10.1 指令及字符部分的内容	3 - 235
3.10.2 全局功能的控制步骤 (指令: 1618)	3 - 237
3.11 至外部设备的数据发送 (接通请求功能): 用于串行通信模块	3 - 239
3.11.1 接通请求功能的执行步骤	3 - 240
3.11.2 通过接通请求功能进行数据发送的格式	3 - 243
3.11.3 接通请求功能的控制步骤 (指令: 2101)	3 - 245
3.12 传送顺控程序的初始化: 用于串行通信模块	3 - 250
3.12.1 指令	3 - 250
3.12.2 传送顺控程序初始化 (指令: 1615)	3 - 250
3.13 模式的切换: 用于串行通信模块	3 - 251
3.13.1 指令及字符部分的内容	3 - 251
3.13.2 模式切换 (指令: 1612)	3 - 255
3.14 显示 LED 的熄灯、通信出错信息 • 出错代码的初始化: 用于串行通信模块	3 - 257
3.14.1 指令及字符部分的内容	3 - 257
3.14.2 显示 LED 的熄灯、通信出错信息 • 出错代码的初始化 (指令: 1617)	3 - 259
3.15 COM.ERR.LED 的熄灯: 用于以太网模块	3 - 260
3.15.1 指令及字符部分的内容	3 - 260
3.15.2 COM.ERR.LED 的熄灯 (指令: 1617)	3 - 261
3.16 回送测试	3 - 262
3.16.1 指令及字符部分的内容	3 - 262
3.16.2 回送测试 (指令: 0619)	3 - 263

3.17 可编程控制器 CPU 监视登录/解除：用于串行通信模块	3 - 264
3.17.1 指令及字符部分的内容	3 - 266
3.17.2 可编程控制器 CPU 监视登录(指令：0630)	3 - 271
3.17.3 可编程控制器 CPU 监视解除(指令：0631)	3 - 274
3.17.4 通过可编程控制器 CPU 监视功能发送的数据	3 - 275
3.18 远程口令的解锁/锁定	3 - 286
3.18.1 指令及字符部分的内容	3 - 291
3.18.2 远程口令的解锁/锁定(指令：1630、1631)	3 - 292

4 通过 QnA 兼容 2C 帧进行通信时

4 - 1 到 4 - 15

4.1 控制步骤、报文格式	4 - 1
4.2 数据指定项目的内容	4 - 7
4.3 QnA 兼容 2C 帧用指令及功能列表	4 - 8
4.4 数据通信时的注意事项	4 - 9
4.5 通过 QnA 兼容 2C 帧进行数据通信的示例	4 - 10

5 通过 A 兼容 1C 帧进行通信时

5 - 1 到 5 - 55

5.1 控制步骤、报文格式	5 - 1
5.1.1 控制步骤、指令说明项的阅读方法	5 - 1
5.1.2 控制步骤、报文格式	5 - 3
5.1.3 A 兼容 1C 帧的数据指定项目的内容	5 - 8
5.1.4 字符部分的传送数据的思路	5 - 14
5.1.5 A 兼容 1C 帧用指令及功能列表	5 - 15
5.2 软元件存储器的读取、写入	5 - 18
5.2.1 指令及软元件范围	5 - 18
5.2.2 位单位的批量读取(指令：BR、JR)	5 - 23
5.2.3 位单位的批量写入(指令：BW、JW)	5 - 24
5.2.4 位单位的测试(随机写入)(指令：BT、JT)	5 - 25
5.2.5 字单位的批量读取(指令：WR、QR)	5 - 26
5.2.6 字单位的批量写入(指令：WW、QW)	5 - 28
5.2.7 字单位的测试(随机写入)(指令：WT、QT)	5 - 30
5.2.8 软元件存储器的监视	5 - 31
5.3 扩展文件寄存器的读取、写入	5 - 36
5.3.1 ACPU 公共指令及地址	5 - 36
5.3.2 AnA/AnUCPU 公共指令及软元件编号	5 - 37
5.3.3 扩展文件寄存器的读取、写入时的注意事项	5 - 40
5.3.4 扩展文件寄存器的批量读取(指令：ER)	5 - 41
5.3.5 扩展文件寄存器的批量写入(指令：EW)	5 - 42
5.3.6 扩展文件寄存器的直接读取(指令：NR)	5 - 43
5.3.7 扩展文件寄存器的直接写入(指令：NW)	5 - 44
5.3.8 扩展文件寄存器的测试(随机写入)(指令：ET)	5 - 45
5.3.9 扩展文件寄存器的监视	5 - 46

5.4 智能功能模块的缓冲存储器的读取、写入	5 - 49
5.4.1 指令及处理内容.....	5 - 49
5.4.2 控制步骤中智能功能模块编号的思路	5 - 50
5.4.3 智能功能模块的缓冲存储器的读取(指令: TR)	5 - 52
5.4.4 智能功能模块的缓冲存储器的写入(指令: TW)	5 - 53
5.5 回送测试.....	5 - 54

6 通过 A 兼容 1E 帧进行通信	6 - 1 到 6 - 60
---------------------------	-----------------------

6.1 报文格式、控制步骤.....	6 - 2
6.1.1 指令说明项的阅读方法.....	6 - 2
6.1.2 报文格式、控制步骤.....	6 - 3
6.1.3 A 兼容 1E 帧的数据指定项目的内容.....	6 - 7
6.1.4 字符部分的传送数据的思路.....	6 - 13
6.2 A 兼容 1E 帧用指令及功能列表	6 - 14
6.3 软元件存储器的读取、写入.....	6 - 17
6.3.1 指令及软元件范围.....	6 - 17
6.3.2 位单位的批量读取(指令: 00)	6 - 22
6.3.3 位单位的批量写入(指令: 02)	6 - 24
6.3.4 位单位的测试(随机写入)(指令: 04)	6 - 26
6.3.5 字单位的批量读取(指令: 01)	6 - 28
6.3.6 字单位的批量写入(指令: 03)	6 - 30
6.3.7 字单位的测试(随机写入)(指令: 05)	6 - 32
6.3.8 软元件存储器的监视.....	6 - 34
6.4 扩展文件寄存器的读取、写入.....	6 - 39
6.4.1 ACPU 公共指令及地址	6 - 39
6.4.2 AnA/AnUCPU 公共指令及软元件编号.....	6 - 40
6.4.3 扩展文件寄存器读取、写入时的注意事项	6 - 43
6.4.4 扩展文件寄存器的批量读取(指令: 17)	6 - 44
6.4.5 扩展文件寄存器的批量写入(指令: 18)	6 - 46
6.4.6 扩展文件寄存器的直接读取(指令: 3B)	6 - 48
6.4.7 扩展文件寄存器的直接写入(指令: 3C)	6 - 50
6.4.8 扩展文件寄存器的测试(随机写入)(指令: 19)	6 - 51
6.4.9 扩展文件寄存器的监视.....	6 - 52
6.5 智能功能模块的缓冲存储器的读取、写入	6 - 55
6.5.1 指令及处理内容.....	6 - 55
6.5.2 控制步骤中智能功能模块编号的思路	6 - 56
6.5.3 智能功能模块的缓冲存储器读取(指令: 0E)	6 - 58
6.5.4 智能功能模块的缓冲存储器写入(指令: 0F)	6 - 60

附录	附录 - 1 到附录 - 28
-----------	------------------------

附录 1 关于通过软元件存储器的扩展指定进行的读取、写入.....	附录 - 1
附录 1.1 软元件存储器扩展指定时的字符部分的数据的排列及内容.....	附录 - 3
附录 1.2 可进行扩展指定的软元件存储器及指定示例.....	附录 - 8
附录 1.3 软元件存储器的扩展指定时的限制.....	附录 - 15

附录 2 关于缓冲存储器的读取/写入 附录 - 16

附录 3 通过 MC 协议进行通信的可编程控制器侧的处理时间..... 附录 - 18

 附录 3.1 可编程控制器 CPU 的处理时间(扫描时间的延长时间)..... 附录 - 18

索引	索引 - 1 到索引 - 2
----	----------------

关联手册

关于 MELSEC 通信协议，可通过本手册进行确认。
关于其它内容请根据用途参阅下述手册。

(1) C24 的关联手册

手册名称	手册编号
Q 系列串行通信模块用户手册(基本篇) 对用于模块使用的概要、适用系统配置、规格、投行前的步骤、与外部设备的基本数据通信方法、维护、点检、故障排除等有关内容进行说明。 (另售)	SH-080238C
MELSEC-L 串行通信模块用户手册(基本篇) 对用于模块使用的概要、适用系统配置、规格、投行前的步骤、与外部设备的基本数据通信方法、维护、点检、故障排除等有关内容进行说明。 (另售)	SH-080949CHN
MELSEC-Q/L 串行通信模块用户手册(应用篇) 对通过模块的特殊功能与外部设备进行数据通信的方法等有关内容进行说明。 (另售)	SH-080284C

(2) E71 的关联手册

手册名称	手册编号
Q 系列以太网接口模块用户手册(基本篇) 对以太网模块的规格、与外部设备的数据通信步骤、线路连接(打开/关闭)、固定缓冲通信、随机访问用缓冲通信、故障排除等有关内容进行说明。 (另售)	SH-080235C
Q 系列以太网接口模块用户手册(Web 功能篇) 对使用以太网模块的 Web 功能时的有关内容进行说明。 (另售)	SH-080410C

(3) CPU 模块的用户手册

手册名称	手册编号
QnUCPU 用户手册 (功能解说/程序基础篇) 对程序创建所必需的功能、编程方法以及软元件等进行说明。 (另售)	SH-080812CHN
Qn (H) /QnPH/QnPRHCPU 用户手册 (功能解说/程序基础篇) 对程序创建所必需的功能、编程方法以及软元件等进行说明。 (另售)	SH-080808ENG
MELSEC-L CPU 模块用户手册 (功能解说/程序基础篇) 对程序创建所必需的功能、编程方法以及软元件等进行说明。 (另售)	SH-080942CHN
QnUCPU 用户手册 (内置以太网端口通信篇) 对 CPU 模块的内置以太网端口功能有关内容进行说明。 (另售)	SH-080813CHN
MELSEC-L CPU 模块用户手册 (内置以太网功能篇) 对 CPU 模块的内置以太网功能有关内容进行说明。 (另售)	SH-080944CHN

(4) 操作手册

手册名称	手册编号
GX Works2 Version1 操作手册 (公共篇) 记载了 GX Works2 的系统配置、参数设置、在线功能的操作方法等，简单工程及结构化工程的通用功能有关内容。 (另售)	SH-080932CHN
GX Developer Version8 操作手册 记载了 GX Developer 中的程序创建方法、打印输出方法、监视方法、调试方法等有关内容。 (另售)	SH-080311C

• **本手册的阅读方法**

以下将 MC 协议通信功能的有关内容按使用目的分类进行说明。

(1) 希望了解 MC 协议的通信功能时

- 1.1 节中记载了 MC 协议的数据通信概要。
- 1.2 节中记载了 MC 协议的主要数据通信功能有关内容。

(2) 希望了解 MC 协议的帧的类型及访问范围时

- (a) 希望了解 MC 协议的帧名的阅读方法时
 - 第 1 章中记载了帧名的阅读方法。
 - 2.1 节中记载了以前产品模块的通信帧与 MC 协议的帧的对应。
- (b) 希望了解各帧的类型及访问范围时
 - 2.1 节中记载了串行通信模块、以太网接口模块可使用的帧的类型。
 - 2.2 节以后记载了各帧的用途及允许访问范围。

(3) 通过 MC 协议进行通信时

- 第 2 章的 2.3 节以后记载了通过 MC 协议进行通信时的公共信息。
- 第 5 章中记载了 A 兼容 1C 帧用的指令、报文格式、控制步骤有关内容。
- 第 6 章中记载了 A 兼容 1E 帧用的指令、报文格式、控制步骤有关内容。
- 第 4 章中记载了 QnA 兼容 2C 帧用的指令、报文格式、控制步骤有关内容。
- 第 3 章中记载了 QnA 兼容 3E 帧、QnA 兼容 3C 帧、QnA 兼容 4C 帧、4E 帧用的指令、报文格式、控制步骤有关内容。

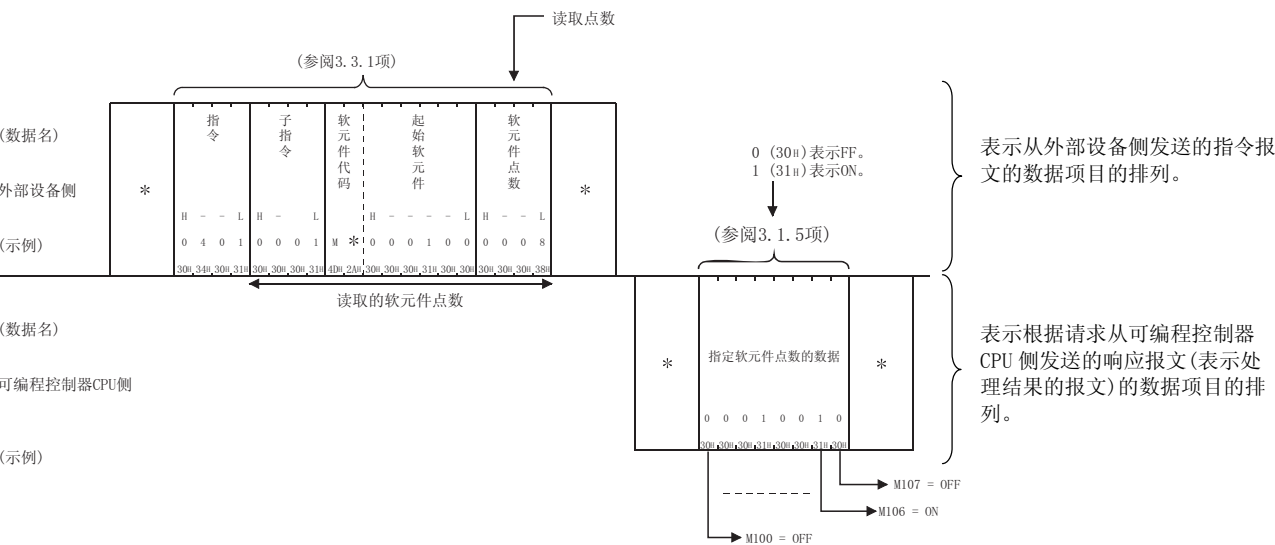
• 本手册的结构

对于通过 MC 协议进行通信时的报文格式、控制步骤有关内容，以下述格式进行说明。

[控制步骤]

(1) 通过由二进制代码进行的通信，对内部继电器 M100~M107 的 8 点进行读取的情况下

表示举例说明的下述控制步骤的来自于外部设备侧的请求内容。



在指令报文以及响应报文中，对于与使用其它指令时的控制步骤相同的数据排列部分以“*”符号表示。

(指令报文以及响应报文的起始部分及最后部分)

关于“*”符号部分的数据的排列，请通过下述说明项进行确认。

对于“*”符号部分的数据排列，根据各个以太网接口模块、串行通信模块而有所不同。

(1) 经由以太网接口模块进行通信时

- A 兼容 1E 帧 : 参阅 6.1 节
- QnA 兼容 3E 帧 : 参阅 3.1.2 项
- 4E 帧 : 参阅 3.1.2 项

(2) 经由串行通信模块进行通信时

- A 兼容 1C 帧 : 参阅 5.1 节
- QnA 兼容 2C 帧 : 参阅 4.1 项
- QnA 兼容 3C 帧 : 参阅 3.1.4 项
- QnA 兼容 4C 帧 : 参阅 3.1.5 项

总称・略称

在本手册中，除非特别标明，将使用下述总称・略称对串行通信模块、以太网接口模块以及数据通信设备等有关内容进行说明。

(1) 模块的总称・略称

关于模块型号的有关内容，请参阅所使用的模块的手册。

总称/略称	总称・略称的内容	
串行通信模块	下述模块的总称。	
	Q 系列	QJ71C24N、QJ71C24N-R2、QJ71C24N-R4、QJ71C24、QJ71C24-R2
	L 系列	LJ71C24、LJ71C24-R2
	QnA 系列	AJ71QC24、AJ71QC24-R2、AJ71QC24-R4、A1SJ71QC24、A1SJ71QC24-R2、AJ71QC24N、AJ71QC24N-R2、AJ71QC24N-R4、A1SJ71QC24N、A1SJ71QC24N-R2
C24	QJ71C24N、QJ71C24N-R2、QJ71C24N-R4、QJ71C24、QJ71C24-R2、LJ71C24、LJ71C24-R2 型串行通信模块的略称。	
E71	QJ71E71-100、QJ71E71-B5、QJ71E71-B2 形以太网接口模块的略称。	
QE71	AJ71QE71N3-T、AJ71QE71N-B5、AJ71QE71N-B2、AJ71QE71N-T、AJ71QE71N-B5T、A1SJ71QE71N3-T、A1SJ71QE71N-B5、A1SJ71QE71N-B2、A1SJ71QE71N-T、A1SJ71QE71N-B5T 的总称。	
LP25/BR15	AJ72LP25、AJ72BR15 的总称。	
QLP21/QBR11	AJ71QLP21、AJ71QBR11 的总称。	
QLP25/QBR15	AJ72QLP25(G)、AJ72QBR15、A1SJ72QLP25、A1SJ72QBR15 的总称。	
UC24 计算机链接模块	AJ71UC24、A1SJ71UC24-R2、A1SJ71UC24-R4、A1SJ71UC24-PRF、A1SJ71C24-R2、A1SJ71C24-R4、A1SJ71C24-PRF、A2CCPUC24、A2CCPUC24-PRF 的总称。 * A 系列计算机链接模块。	
ACPU	AnNCPU、AnACPU、AnUCPU、QCPU(A 模式) 的总称。 *在通过 MC 协议进行的通信功能中，可以通过外部设备进行访问的 A 系列可编程控制器 CPU。	
AnACPU	A2ACPU、A2ACPU-S1、A2ACPUP21/R21、A2ACPUP21/R21-S1、A3ACPU、A3ACPUP21/R21 的总称。	
AnNCPU	A1NCPU、A1NCPUP21/R21、A2NCPU、A2NCPU-S1、A2NCPUP21/R21、A2NCPUP21/R21-S1、A3NCPU、A3NCPUP21/R21 的总称。	
AnA/AnU/QnACPU	AnACPU、AnUCPU、QnACPU 的总称。	
AnUCPU	A2UCPU、A2UCPU-S1、A2ASCPU、A2ASCPU-S1、A3UCPU、A4UCPU 的总称。	
AnU/QnACPU	AnUCPU、QnACPU 的总称。	
QCPU	Q00JCPU、Q00CPU、Q01CPU、Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU、Q02PHCPU、Q06PHCPU、Q12PHCPU、Q25PHCPU、Q12PRHCPU、Q25PRHCPU、Q00UJCPU、Q00UCPU、Q01UCPU、Q02UCPU、Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、Q20UDHCPU、Q26UDHCPU、Q03UDECPU、Q04UDEHCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU、Q100UDEHCPU 的总称。	
QCPU 站	安装了 QCPU 的可编程控制器的略称。	
QCPU(A 模式)	Q02CPU-A、Q02HCPU-A、Q06HCPU-A 的总称。	
基本型 QCPU	Q00JCPU、Q00CPU、Q01CPU 的总称。	
高性能型 QCPU	Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU 的总称。	
过程 CPU	Q02PHCPU、Q06PHCPU、Q12PHCPU、Q25PHCPU 的总称。	
冗余 CPU	Q12PRHCPU、Q25PRHCPU 的总称。	

总称/略称	总称・略称的内容
通用型 QCPU	Q00UJCPU、Q00UCPU、Q01UCPU、Q02UCPU、Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、Q20UDHCPU、Q26UDHCPU、Q03UDECPU、Q04UDEHCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU、Q100UDEHCPU 的总称。
LCPU	L02CPU、L26CPU-BT、L02CPU-P、L26CPU-PBT 的总称。
安全 CPU	QS001CPU 的略称。但是，与 QCPU 的通用部分记载为 QCPU 或者 Q 系列 CPU。
QnACPU	Q2ACPU、Q2ACPU-S1、Q2ASCPU、Q2ASCPU-S1、Q2ASHCPU、Q2ASHCPU-S1、Q3ACPU、Q4ACPU、Q4ARCPU 的总称。
QnACPU 站	安装了 QnACPU 的可编程控制器的略称。

(2) 手册的总称・略称

总称/略称	总称・略称的内容
用户手册(基本篇)	- 串行通信模块 Q 系列串行通信模块用户手册(基本篇)MELSEC-L 串行通信模块用户手册(基本篇) - 以太网接口模块 Q 系列以太网接口模块用户手册(基本篇)
用户手册(应用篇)	- 串行通信模块 MELSEC-Q/L 串行通信模块用户手册(应用篇) - 以太网接口模块 Q 系列以太网接口模块用户手册(应用篇)

(3) 其它总称・略称

总称/略称	总称・略称的内容
计算机	外部设备中可以以 MC 协议或者双向协议进行数据通信的设备的总称。
开关设置	智能功能模块开关设置的略称。
数据通信功能	MC 协议、通信协议、无顺序协议、双向协议的总称。
缓冲存储器	用于存储与可编程控制器 CPU 发送接收的数据（设置值、监视值等）的，智能功能模块/特殊功能模块的存储器的总称。
I/F	接口的略称。
MELSECNET/10	MELSECNET/10 网络系统的略称。
MELSECNET/H	MELSECNET/H 网络系统的略称。
ONDEMAND	G. ONDEMAND、GP. ONDEMAND 的略称。
RS-232（接口）	RS-232 标准接口的略称。
RS-422/485（接口）	RS-422 以及 RS-485 标准接口的略称。

术语

包含本手册在内的关联手册中使用的术语的含义及内容如下所示。

术语	内容
4E 帧	是 MC 协议中用于通过 ASCII 代码或二进制代码数据进行通信的以太网接口模块用的报文格式之一。 是在 QnA 兼容 3E 帧中附加了报文识别用的任意编号(串行编号)的报文格式。 在外部设备中通过附加的报文识别用的串行编号，可以对对应于多个指令报文的响应报文进行确认。
A 兼容 1C 帧 (格式 1~格式 4)	是 MC 协议中用于通过 ASCII 代码数据进行通信的串行通信模块用的报文格式之一。 是与通过 A 系列计算机链接模块的专用协议进行通信时相同的报文格式。 对于 Q/LCPU，可以以 AnACPU 的软元件范围进行软元件存储器的读取、写入。
A 兼容 1E 帧	是 MC 协议中用于通过 ASCII 代码或二进制代码数据进行通信的以太网接口模块用的报文格式之一。 是与进行 A 系列以太网接口模块的可编程控制器 CPU 内数据的读取/写入通信时相同的报文格式。 对于 Q/LCPU，可以以 AnACPU 的软元件范围进行软元件存储器的读取、写入。
GX Developer	是进行设计、调试、维护的编程工具。
GX Works2	
GX Configurator-SC	
MELSEC 通信协议 (MC 协议)	是串行通信模块或以太网接口模块的通信步骤中，用于从外部设备对可编程控制器 CPU 进行访问的数据通信功能之一。 (在本手册中表示为 MC 协议) 有通过 ASCII 代码数据进行通信的方法及通过二进制代码数据进行通信的方法。
MX Component	是用于进行串行通信的 Active X [®] 控件库。
QnA 兼容 2C 帧 (格式 1~格式 4)	是 MC 协议中用于通过 ASCII 代码数据进行通信的串行通信模块用的报文格式之一。 与通过 QnA 系列串行通信模块的专用协议进行通信用帧相同的报文格式。 • QnA 兼容 2C 帧(格式 1~格式 4)：QnA 简易帧(格式 1~格式 4)
QnA 兼容 3C 帧 (格式 1~格式 4) QnA 兼容 4C 帧 (格式 1~格式 4)	是 MC 协议中用于通过 ASCII 代码数据进行通信的串行通信模块用的报文格式之一。 与通过 QnA 系列串行通信模块的专用协议进行通信用帧相同的报文格式。 • QnA 兼容 3C 帧(格式 1~格式 4)：QnA 帧(格式 1~格式 4) • QnA 兼容 4C 帧(格式 1~格式 4)：QnA 扩展帧(格式 1~格式 4)
QnA 兼容 3E 帧	是 MC 协议中用于通过 ASCII 代码或二进制代码数据进行通信的以太网接口模块用的报文格式之一。 与对 QnA 系列以太网接口模块的可编程控制器 CPU 内数据进行读取/写入通信时相同的报文格式。
QnA 兼容 4C 帧 (格式 5)	是 MC 协议中用于通过二进制代码数据进行通信的串行通信模块用的报文格式之一。 与通过 QnA 系列串行通信模块的专用协议进行通信用帧相同的报文格式。 • QnA 兼容 4C 帧(格式 5)：QnA 扩展帧(格式 5)

术语	内容
SD 存储卡	表示 Secure Digital Memory Card。是以快闪存储器构成的存储装置。有 L1MEM-2GBSD、L1MEM-4GBSD。
智能功能模块	是 A/D、D/A 转换模块等，具有除输入输出以外的功能的 MELSEC-Q/L 系列的模块。
智能功能模块软元件	是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。
双向协议	是串行通信模块的通信步骤中，用于外部设备与可编程控制器 CPU 之间进行任意数据通信的数据通信功能之一。
通信协议	是 C24 与外部设备之间进行数据通信时，以外部设备侧的协议进行数据发送接收的数据通信功能之一。通过 GX Works2 或 GX Configurator-SC(通信协议支持功能)进行设置。
特殊功能模块	是 A/D、D/A 转换模块等，具有除输入输出以外的功能的 MELSEC-A/QnA 系列的模块。
特殊功能模块软元件	是可通过 CPU 模块直接访问特殊功能模块的缓冲存储器的软元件。
独立动作	是与串行通信模块的 2 个接口无关，通过各通信协议设置指定的功能与外部设备进行数据通信时的各接口的动作。
多点连接	是使用串行通信模块的 RS-422/485 接口，将多个外部设备及其它串行通信模块等以 1:n、m:n 进行连接时的连接名称。
无顺序协议	是用户的通信步骤中，用于外部设备与可编程控制器 CPU 之间对任意数据进行通信的数据通信功能之一。
信息发送功能 (打印功能)	是将发送至外部设备(主要以打印机为对象)的字符数据(信息)作为用户登录帧预先登录到串行通信模块，通过无顺序协议对多个用户登录帧的登录数据进行发送的功能。 (通过来自于可编程控制器 CPU 的指示进行发送)
用户登录帧	是在下述功能中，将外部设备与串行通信模块之间进行发送接收的报文中的固定格式部分登录到模块中，作为数据的发送・接收使用时的数据名。 (用户登录帧的数据内容取决于外部设备的规格。) 将进行发送接收的报文中的起始部分、最终部分的数据排列(传送控制代码、C24 站编号、和校验、固定数据等)分别登录到串行通信模块中使用。 • MC 协议的接通请求功能。 • 通过无顺序协议进行数据发送接收的功能。
联动动作	是串行通信模块的 2 个接口分别连接了外部设备时，在 2 个接口联动的同时与外部设备进行数据通信时的各接口的动作。 2 个接口以相同的数据通信功能(MC 协议(相同格式)或无顺序协议)、相同的传送规格进行数据通信。 (不能进行双向协议的联动动作)

1 概要

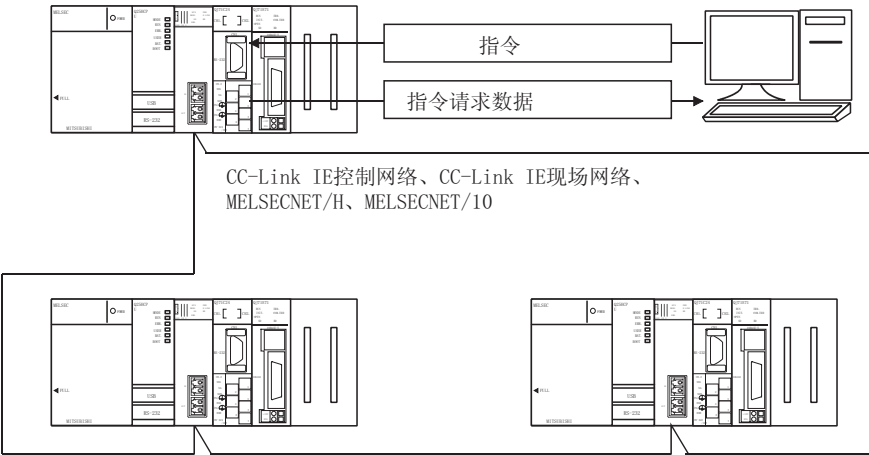
本手册对使用外部设备经由串行通信模块或以太网接口模块通过 MELSEC 通信协议(以下略称为 MC 协议)的数据通信功能对可编程控制器 CPU 的数据进行读取、写入等的方法有关内容进行说明。
将本手册中介绍的程序示例引用到实际系统的情况下，应充分验证对象系统中不会存在控制方面的问题。

通过 MC 协议进行数据通信时，务必阅读第 2 章。

1.1 MELSEC 通信协议的概要

以下对 MC 协议的概要进行说明。

- (1) MC 协议是指，用于外部设备经由 C24 或 E71 对可编程控制器 CPU 的软元件数据、程序进行读取/写入等的 Q/L 系列可编程控制器用的通信方式的名称。
只要是可安装应用程序且可通过 MC 协议的控制步骤进行数据发送接收的设备，便可通过 MC 协议进行通信。
- (2) 关于外部设备用于访问可编程控制器的报文格式、控制步骤，取决于各个 C24、E71。



- (3) 通过 MC 协议进行数据通信时的报文格式、控制步骤与通过以前产品 A/QnA 系列模块访问可编程控制器时相同。

ENQ	站编号		可编程序控制器 参数编号	指令	报文等待	起始软元件										软元件点数		和校验代码	
	H L					2 字符 () 16进制										H L			
	0	0				0	X	0	0	0	0	0	4	0	0	2	7	2	
05h	30h	30h	30h	31h	51h	52h	30h	38h	30h	30h	30h	30h	34h	30h	30h	32h	37h	32h	

(A兼容IC帧的指令报文)

STX	站编号		可编程序控制器 参数编号	软元件点数的数据 [软元件点数 × 4 字符]												ETX	和校验代码	
	H L			H L													H L	
	0	0		0	1	1	2	3	4	A	B	C	D				9	8
02h	30h	30h	30h	31h	31h	32h	32h	33h	34h	41h	42h	43h	44h	03h	39h	38h		

(A兼容IC帧的响应报文)

外部设备侧可以使用经由下述 A/QnA 系列模块访问可编程控制器时的程序，对 Q/L 系列可编程控制器进行访问。

1) 通过 C24 访问可编程控制器

可以使用下述 A/QnA 系列模块用的外部设备侧程序对可编程控制器进行访问。

- A 系列计算机链接模块
- QnA 系列串行通信模块

2) 通过 E71 访问可编程控制器

可以使用下述 A/QnA 系列模块用的外部设备侧程序对可编程控制器进行访问。

- A 系列以太网接口模块
- QnA 系列以太网接口模块

* MC 协议的数据通信功能相当于以前产品的下述数据通信功能。

· C24 的情况下

相当于通过 A 系列计算机链接模块、QnA 系列串行通信模块支持的专用协议进行的数据通信功能。

· E71 的情况下

相当于 A 系列/QnA 系列以太网接口模块支持的可编程控制器 CPU 内数据的读取/写入功能。

(4) 对于可编程控制器侧，C24/E71 根据来自于外部设备的指令进行数据发送接收。因此，可编程控制器 CPU 侧不需要数据通信用的顺控程序。

* C24 的情况下，使用用于从可编程控制器 CPU 发送数据的接通请求功能时，需要数据发送用的顺控程序。

(5) 对象设备为个人计算机的情况下，通过使用另售的通信支持工具(MX Component)，可以在无需理会 MC 协议的报文格式及发送接收步骤的状况下创建对象设备侧的通信程序。

* 对应的操作系统根据所使用的 MX Component 的版本而有所不同。

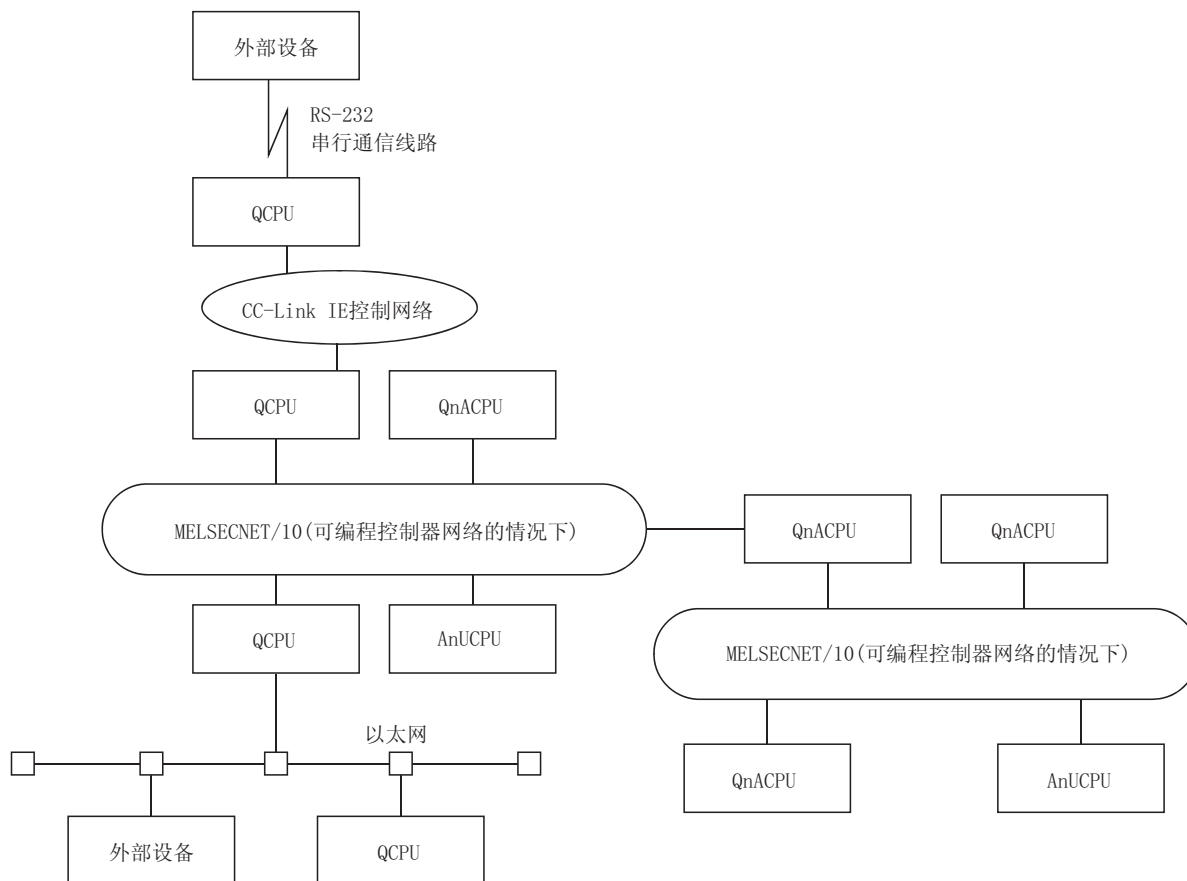
详细内容请参阅 MX Component 的手册。

1.2 MELSEC 通信协议的特点

以下对 MC 协议的特点进行说明。

- (1) 通过 MC 协议进行的数据通信是指，为了在外部设备侧对可编程控制器设备进行管理监视，对可编程控制器内的数据进行读取/写入等的功能。
可以从外部设备(个人计算机、显示器等)对可编程控制器的软件数据及程序文件进行读取/写入，对可编程控制器 CPU 的状态进行控制(远程 RUN/STOP)等。
 - (a) 数据的读取、写入
通过对可编程控制器 CPU 的软件存储器、智能功能模块的缓冲存储器进行数据的读取、写入，可以进行下述控制。
也可对 A/QnA 系列可编程控制器 CPU(其它站)、智能功能模块进行该数据的读取、写入。
 - 1) 数据的读取
在外部设备侧，可以进行可编程控制器 CPU 的动作监视、数据解析以及生产管理等。
 - 2) 数据的写入
可以在外部设备侧发布生产指示等。
 - (b) 文件的读取/写入
通过对可编程控制器 CPU 中存储的顺控程序及参数等文件进行读取、写入，可以进行下述控制。
 - 1) 文件的读取
在外部设备侧，可以进行本站 QCPU/LCPU 以及其它站 QCPU/LCPU/QnACPU 的文件管理。
 - 2) 文件的写入
通过将外部设备侧存储的文件数据根据需要写入到可编程控制器 CPU 中，可以对执行程序等进行更改(替换)。
 - (c) 可编程控制器 CPU 的远程控制
通过进行远程 RUN/STOP/PAUSE/锁存清除/复位，可以从外部设备对可编程控制器 CPU 进行远程操作。

- (2) 在存在有 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网的系统中，可以从外部设备经由各个网络对其它站可编程控制器进行访问。



- (3) C24 中具有可编程控制器 CPU 监视功能。
 通过使用该功能，可以在一定间隔下、以及在机械设备发生出错时或某个条件成立时将可编程控制器 CPU 状态及软元件存储器的数据发送至外部设备。
 可以减少外部设备侧的数据读取处理。
 * E71 中也具有通过电子邮件发送对可编程控制器 CPU 进行监视的功能。
 (请参阅用户手册(应用篇)。)

要点

关于 MC 协议的数据通信中外部设备可执行的内容，请通过第 3 章 ~ 第 6 章的各“指令及功能列表”说明项进行确认。

2 关于通过 MELSEC 通信协议进行的数据通信

以下对外部设备通过 C24/E71 对可编程控制器 CPU 进行数据读取、写入等时的 MC 协议的数据通信有关内容进行说明。

2.1 数据通信用帧的类型及用途

以下对用于外部设备通过 MC 协议访问可编程控制器 CPU 的帧(数据通信报文)的类型及各帧的用途有关内容进行说明。
外部设备经由 C24/E71 访问可编程控制器时，通过下述某个帧的指令报文(访问请求)及响应报文(响应)的发送接收进行数据通信。
应在对 2.2 项中所示的各帧中的允许访问范围进行考虑的基础上，选择使用的帧。

对象模块	可使用的通信帧		通信数据的代码	控制步骤说明项
C24	QnA 兼容 3C 帧	格式 1~4	ASCII 代码	第 3 章
	QnA 兼容 4C 帧	格式 5	二进制代码	
	QnA 兼容 2C 帧	格式 1~4	ASCII 代码	第 4 章
	A 兼容 1C 帧			第 5 章
E71	4E 帧		ASCII 代码或二进制代码	第 3 章
	QnA 兼容 3E 帧			第 6 章
	A 兼容 1E 帧			

(1) QnA 兼容 3C 帧、QnA 兼容 4C 帧、QnA 兼容 3E 帧、4E 帧

- (a) 是以从对象设备对 Q/L/QnACPU 的全部软元件、全部文件进行访问为主要目的的帧。
- (b) 经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网，也可对 A 系列可编程控制器 CPU 的软元件进行访问。(参阅 2.2 节(1))

(2) 4E 帧

- (a) 是以从对象设备对 Q/LCPU 的全部软元件、全部文件进行访问为主要目的的帧。
- (b) 经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网，也可对 A/QnA 系列可编程控制器 CPU 的软元件进行访问。(参阅 2.2 节(2))

(3) QnA 兼容 2C 帧

- (a) 可以对 C24 安装站的 QCPU/LCPU(本站)以及多点连接的 Q/L/QnACPU 站的软元件存储器进行访问。
- (b) 与 QnA 兼容 3E/3C/4C 帧相比报文格式可简单化。
- (c) 由于传送数据变少，因此外部设备侧的报文处理易于进行，可以缩短报文的传送时间。

(4) A 兼容 1C 帧、A 兼容 1E 帧

- (a) 是与使用 A 系列的计算机链接模块或以太网接口模块访问可编程控制器时相同的报文结构。
- (b) 可以使用用于 A 系列可编程控制器而创建的外部设备侧的数据通信用软件，可以使用相同的帧对多点连接、网络连接的 Q/L/QnACPU 及 Q/L/QnACPU 以外的可编程控制器 CPU 进行访问。
对于 Q/L/QnACPU，只能对与 AnCPU、AnNCPU、AnACPU、AnUCPU 中存在的软元件相同名称的软元件在 AnACPU 的软元件范围内进行访问。
· 使用 C24 时 : 参阅 5.2.1 项(2)
· 使用 E71 时 : 参阅 6.3.1 项(2)
不能对 Q/L/QnACPU 中新增的软元件进行访问。
- (c) 对通用型 QCPU 进行访问的情况下，应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。
序列号的前 5 位数为 10101 以前情况下，应以 QnA 兼容 2C/3C/4C 帧、QnA 兼容 3E 帧、4E 帧进行访问。

备注

- (1) 通过 MC 协议进行数据通信时的数据通信帧名的阅读方法如下所示。
数据通信帧名是由以下 3 部分组成：用于表示与以前产品模块的指令兼容性的对象可编程控制器 CPU、对应的以前产品模块的帧、对象模块。
xxx 兼容 n m 帧(示例：QnA 兼容 3C 帧、QnA 兼容 3E 帧)
a) b) c)
- (a) 用于表示与以前产品模块的指令兼容性的对象可编程控制器 CPU
A : A 系列可编程控制器 CPU
QnA : QnA 系列可编程控制器 CPU
- (b) 对应的以前产品模块的帧
1 : 兼容 A 系列的计算机链接模块、以太网接口模块支持的指令的通信帧
2 : 兼容 QnA 系列串行通信模块支持的 QnA 简易帧
3 : QnA 系列串行通信模块支持的 QnA 帧
兼容 QnA 系列以太网接口模块支持的通信帧
4 : 兼容 QnA 系列串行通信模块支持的 QnA 扩展帧
- (c) 本手册中介绍的可通过相应帧进行数据通信的对象模块
C : C24
E : E71

- (2) 经由 C24 进行访问时
外部设备以 GX Works2 或 GX Developer 的开关设置的“通信协议设置”中选择的格式 No. 的帧对可编程控制器 CPU 进行访问。
* 选择了格式 1 ~ 4 中之一时，可以以上述 4 种类型的帧进行访问，以所选择格式指令报文及响应报文的发送接收进行数据通信。
- (3) 经由 E71 进行访问时
外部设备可以以上述 2 种类型的帧进行访问，以 GX Works2 或 GX Developer 的动作设置中选择的代码的帧对指令报文及响应报文进行发送接收，对可编程控制器进行访问。
- (4) 通过二进制代码数据进行的通信与通过 ASCII 代码数据进行的通信相比，前者的通信数据量约为后者的二分之一，因此可缩短通信时间。

2.2 各数据通信用帧的允许访问范围

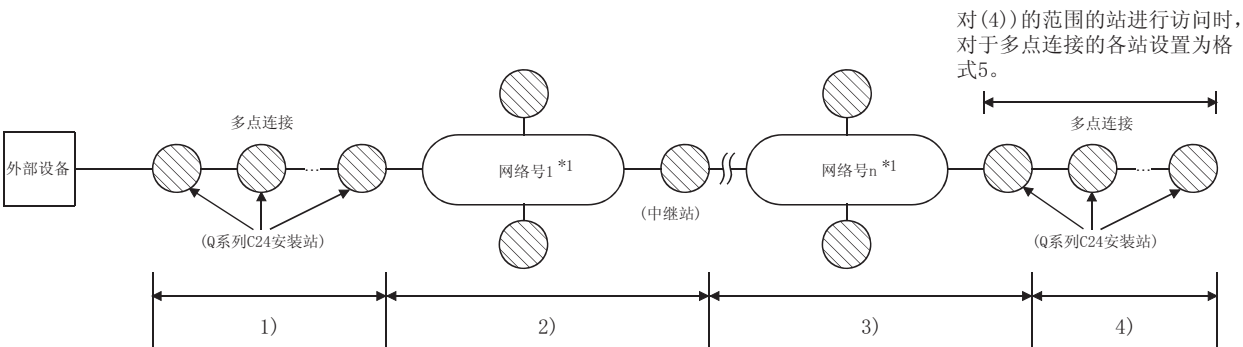
在通过 MC 协议进行的通信中，对于未安装 C24/E71 的可编程控制器站，可以经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 以及以太网进行访问。

在本节中，对访问目标可编程控制器未通过串行通信线路/以太网直接连接时，从通信对象设备可访问的其它站可编程控制器的范围的有关内容进行说明。

○：可以访问 ×：不能访问

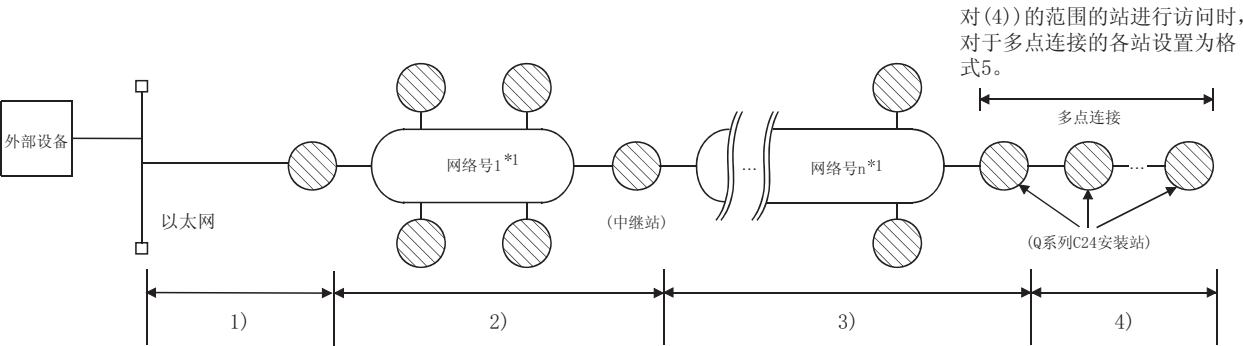
对象模块	使用的帧		可访问的站的范围			
			1)	2)	3)	4)
C24	QnA 兼容 3C 帧	格式 1 ~ 4	○	○	○	×
	QnA 兼容 4C 帧	格式 1 ~ 4	○	○	○	○
		格式 5	○	○	○	○
	QnA 兼容 2C 帧	格式 1 ~ 4	○	×	×	×
	A 兼容 1C 帧		○	○	×	×
E71	4E 帧		○	○	○	○
	QnA 兼容 3E 帧		○	○	○	○
	A 兼容 1E 帧		○	○	×	×

(1) 使用 C24 时



*1 网络表示 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网。

(2) 使用 E71 时



*1 网络表示 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网。

要点	
通过预先设置路由参数(通信路径)，也可访问下述网络系统的可编程控制器。	
· CC-Link IE 控制网络	
· CC-Link IE 现场网络	
· MELSECNET/H	
· MELSECNET/10	
· 以太网	
可访问范围最多为 8 个网络目标(中继站数：7 站)。	

2.3 MC 协议的控制步骤的思路

以下对外部设备通过 MC 协议对可编程控制器 CPU 进行访问时的步骤(控制步骤)的思路进行说明。

(1) 指令报文的发送

通过 MC 协议进行的数据通信是以半双工通信进行。(*1)
对可编程控制器 CPU 进行访问时，应在接收到来自于可编程控制器 CPU 侧的对应于之前发送的指令报文的响应报文后，再发送下一个指令报文。
(在响应报文的接收完毕之前，不能发送下一个指令报文。)



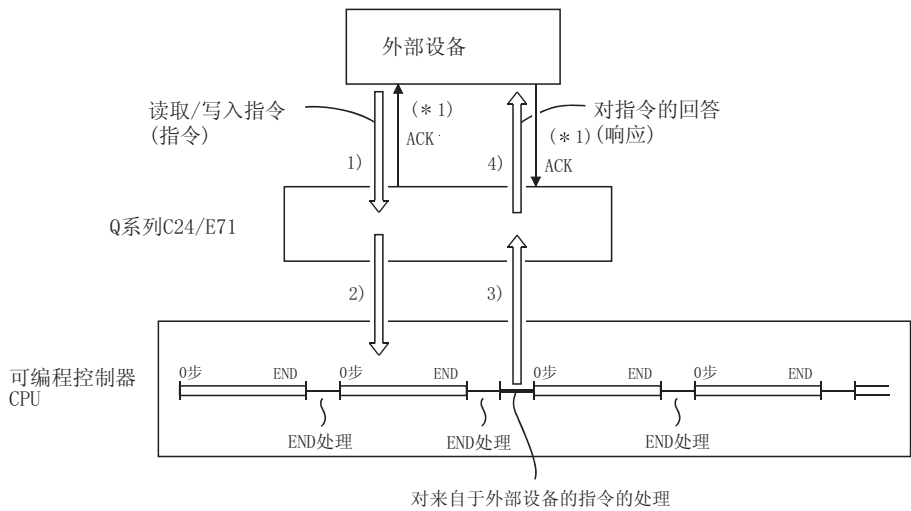
*1 经由 C24 进行访问的情况下，使用接通请求功能时根据用户设置进行全双工通信。
在外部设备与可编程控制器 CPU 的系统配置为 m:n 的情况下，在某个设备与可编程控制器 CPU 的数据通信完毕之前，不能发送下一个指令报文。

(2) 不能接收对应于指令报文的正常结束的响应报文的情况下

- (a) 接收了异常结束的响应报文时
应根据响应报文中的出错代码进行处理。
 - (b) 不能接收响应报文时，或不能全部接收时
 - 经由 E71 的情况下
应在经过了响应监视定时器值的监视时间后，对指令报文进行重新发送。
 - 经由 Q 系列 C24 的情况下
应在经过了响应监视时间(定时器 1)的监视时间后，对指令报文进行重新发送。
- * 应根据需要对监视时间的设置值进行更改。
经由 C24 时，对报文等待时间也应进行更改。

2.4 可编程控制器 CPU 侧访问时机

以下介绍从外部设备经由 C24/E71 对可编程控制器 CPU 进行访问时的可编程控制器 CPU 侧的访问时机。



- 1) 从外部设备向可编程控制器 CPU 侧发送指令报文。
 - 2) 可编程控制器侧的 C24/E71 接收了来自于外部设备的指令报文时，根据指令的指示内容对可编程控制器 CPU 发布数据读取/写入请求。
 - 3) 可编程控制器 CPU 在执行顺控程序的 END 指令时根据来自于外部设备请求的内容对数据进行读取/写入，将处理结果传递到 C24/E71 中。
 - 4) C24/E71 从可编程控制器 CPU 中获取处理的结果后，将包含处理结果在内的响应报文(响应)发送至作为请求源的外部设备中。
- *1 图中所示的 ACK 响应是指，通过 TCP/IP 通信进行经由 E71 的访问时，E71 与外部设备之间发送接收的响应。(是对于报文接收的响应。)是与指令报文中外部设备请求的处理所对应的响应(处理结果)不相同的其它响应。
通过 UDP/IP 通信进行经由 E71 的访问的情况下以及经由 C24 进行访问的情况下不执行*1 的 ACK 响应。

要点
(1) 如果在可编程控制器 CPU 的运行过程中从外部设备进行读取/写入，来自于外部设备的指令的处理需要耗费一定的时间，因此可编程控制器 CPU 的扫描时间将相应延长，应加以注意。 在扫描时间的延长对控制有影响的情况下，应以较少的点数分多次进行访问。
(2) 对于来自于外部设备的对 C24/E71 的缓冲存储器的访问请求将被立即受理。此外，对可编程控制器 CPU 的扫描时间无影响。

备注

关于可编程控制器 CPU 的扫描时间

- (1) 对 C24/E71 及可编程控制器 CPU 进行访问时，指令的请求是在可编程控制器 CPU 处于运行中状态的情况下，在各个 END 处理中进行处理。
(扫描时间将延长相当于指令的请求处理所需时间。)
- (2) 从多个外部设备同时对同一个站发出了访问请求的情况下，根据请求时机，来自于外部设备的请求处理有可能需要等待直至多个 END 处理被执行为止。
通过下述某个方法可以在 1 个扫描内对多个指令请求进行处理。但是，在安全 CPU 中不能使用下述方法。
 - (a) COM 指令的使用方法
应通过顺控程序执行 COM 指令。
关于 COM 指令的详细内容，请参阅所使用的 CPU 模块的编程手册
(公共指令篇)。
 - (b) 通信处理预留时间(SD315)的使用方法(基本型 QCPU、高性能型 QCPU、过程 CPU、冗余 CPU 的情况下)
使用通信处理预留时间(SD315)，将通信处理预留时间在 1 ~ 100ms 的范围内进行设置。
 - (c) 服务处理设置的使用方法(通用型 QCPU、LCPU 的情况下)
使用 GX Works2 或 GX Developer 的可编程控制器参数(可编程控制器系统设置)的服务处理设置，将服务处理时间在 1 ~ 100ms 的范围内进行预留。

2.5 用于对可编程控制器 CPU 进行运行中写入的设置方法

以下对用于从外部设备经由 C24/E71 将数据写入到处于运行状态的可编程控制器 CPU 中的设置有关内容进行说明。(运行中写入)

(1) C24 的设置

在通过 GX Works2 或 GX Developer 进行的智能功能模块开关设置中，将传送设置的“运行中写入”设置为允许。

* 关于通过 GX Works2 或 GX Developer 进行的开关设置，请参阅 C24 的用户手册(基本篇)。

(2) E71 的设置

在通过 GX Works2 或 GX Developer 进行的动作设置中，对“允许运行中写入”进行勾选。

* 关于通过 GX Works2 或 GX Developer 进行的动作设置，请参阅 Q 系列 E71 的用户手册(基本篇)4.7 节。

(3) 可编程控制器 CPU 侧的设置

通过系统保护开关，对“允许可编程控制器 CPU 的运行中写入”进行设置。
(QCPU(通用型 QCPU 除外)的情况下，将系统保护开关 SW1 设置为“OFF”。)

重要	
将外围设备连接到可编程控制器 CPU 上，或将个人计算机等连接到智能功能模块上对运行中的可编程控制器 CPU 进行控制(数据更改)时，应在顺控程序上设置互锁电路，以保障整个系统的安全。 此外，对运行中的可编程控制器 CPU 进行其它控制(程序更改、运行状态更改(状态控制))时，应在熟读手册并充分确认安全的基础上执行操作。 特别是从外部设备对远程的可编程控制器 CPU 进行上述控制时，由于数据通信异常有可能无法对可编程控制器 CPU 侧的故障进行立即处理。应在顺控程序上设置互锁电路的同时，预先在外部设备与可编程控制器 CPU 之间确定发生数据通信异常时系统方面的处理方法等。	

2.6 关于其它站访问

以下介绍经由 CC-Link IE 控制网络、CC-Link IE 现场网络，MELSECNET/H、MELSECNET/10、以太网对其它站可编程控制器进行访问的概要。

2.6.1 可进行其它站访问的可编程控制器模块

对其它站可编程控制器进行访问时，可访问的模块及可作为网络间中继的模块如下所示。（数据通信帧中的可编程控制器编号的指定值为 FFH 以外时的对象站）

(1) 可访问的模块

(a) 可编程控制器 CPU

		型号				
QCPU	基本型 QCPU	Q00JCPU	Q00CPU	Q01CPU		
	高性能型 QCPU	Q02CPU	Q02HCPU	Q06HCPU	Q12HCPU	Q25HCPU
	过程 CPU	Q02PHCPU	Q06PHCPU	Q12PHCPU	Q25PHCPU	
	冗余 CPU	Q12PRHCPU	Q25PRHCPU			
	通用型 QCPU ^{*1*2}	Q00UJCPU	Q00UCPU	Q01UCPU	Q02UCPU	Q03UDCPU
		Q04UDHCPU	Q06UDHCPU	Q10UDHCPU	Q13UDHCPU	Q20UDHCPU
		Q26UDHCPU	Q03UDECPU	Q04UDEHCPU	Q06UDEHCPU	Q10UDEHCPU
		Q13UDEHCPU	Q20UDEHCPU	Q26UDEHCPU	Q50UDEHCPU	Q100UDEHCPU
LCPU ^{*2}		L02CPU	L02CPU-P	L26CPU-BT	L26CPU-PBT	
安全 CPU ^{*3}		QS001CPU				
QnACPU		Q2ACPU	Q2ACPU-S1	Q2ASCPU	Q2ASCPU-S1	Q2ASHCPU
		Q2ASHCPU-S1	Q3ACPU	Q4ACPU		
	冗余 CPU	Q4ARCPU				
ACPU ^{*4}		A1NCP	A2NCP	A2NCP-S1	A3NCP	A2ACPU
		A2ACPU-S1	A3ACPU	A2UCPU	A2UCPU-S1	A3UCPU
		A4UCPU	A1SCPU	A1SJCPU(-S3)	A1SHCPU	A1SJHCPU
		A2SCPU	A2SHCPU	A2ASCPU	A2ASCPU-S1	A0J2HCPU
		Q02CPU-A	Q02HCPU-A	Q06HCPU-A		

- *1 使用 A 兼容 1C/1E 帧的情况下，应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。序列号的前 5 位数为 10101 以前的情况下，应通过 QnA 兼容 2C/3C/4C 帧、QnA 兼容 3E 帧、4E 帧进行访问。
- *2 对 CPU 模块的内置以太网端口进行访问的情况下，请参阅下述手册。
 - QnUCPU 用户手册(内置以太网端口通信篇)
 - MELSEC-L CPU 模块用户手册(内置以太网功能篇)
- *3 对于序列号的前 5 位数为 13041 以前的安全 CPU，只能作为访问站使用。不能作为中继站使用。
- *4 对于安全 CPU，不能对作为至 ACPU 的中继及将 ACPU/QnACPU 作为中继站的安全 CPU 进行访问。
- *5 对于除上述以外的可编程控制器 CPU，可以对与对象设备进行多点连接的 A2CCPUC24、A2CCPUC24-PRF 进行访问。

(b) CC-Link IE 现场网络

	型号
起始模块	LJ72GF15-T2

(c) MELSECNET/H、MELSECNET/10

	型号				
MELSECNET/H 远程 I/O 站	QJ72LP25-25	QJ72LP25GE	QJ72BR15		
MELSECNET/10 远程 I/O 站*6	AJ72QLP25 AJ72BR15	AJ72QBR15	A1SJ72QLP25	A1SJ72QBR15	AJ72LP25(G)

*6 对于 MELSECNET/10 远程 I/O 站，只能对智能功能模块的缓冲存储器进行读取/写入。

要点	对网络系统内的远程 I/O 站的软元件或远程 I/O 站中安装的智能功能模块(特殊功能模块)进行数据写入的情况下，应在 2.5 节(1)(2)中所示的 GX Works2 或 GX Developer 的设置中设置为“允许运行中写入”。
----	--

(2) 可作为网络间中继的模块

	型号				
CC-Link IE 控制网络	QJ71GP21-SX	QJ71GP21S-SX			
CC-Link IE 现场网络	QJ71GF11-T2	LJ71GF11-BT2	QS0J71GF11-T2		
MELSECNET/H	QJ71LP21 QJ71NT11B	QJ71LP21-25 (MELSECNET/H 模式)	QJ71LP21S-25	QJ71LP21GE	QJ71BR11
MELSECNET/10	QJ71LP21 (MELSECNET/10 模式)	QJ71LP21-25	QJ71LP21S-25	QJ71LP21GE	QJ71BR11
	AJ71QLP21(S/G)	AJ71QBR11	A1SJ71QLP21	A1SJ71QBR11	
	AJ71LP21(G)	AJ71LR21	AJ71BR11	A1SJ71LP21	A1SJ71BR11
以太网(*1)	QJ71E71-100	QJ71E71-B5	QJ71E71-B2	QJ71E71	
	AJ71QE71N3-T	AJ71QE71N-B5	AJ71QE71N-B2	AJ71QE71N-T	AJ71QE71N-B5T
	A1SJ71QE71N3-T	A1SJ71QE71N-B5	A1SJ71QE71N-B2	A1SJ71QE71N-T	A1SJ71QE71N-B5T
	AJ71QE71	AJ71QE71-B5	A1SJ71QE71-B2	A1SJ71QE71-B5	

*1 只有 QnA 系列以太网接口模块的功能版本为 B 版以后的模块才可以作为网络间的中继使用。
关于功能版本，请通过模块侧面的“额定铭牌的 DATE 栏”中所示的生产年月进行确认。
(生产年月：公历(低 2 位)，月(2 位)，功能版本(1 位))

要点

- 通过 A 兼容 1E/1C 帧进行其它站访问时，应通过 GX Works2 或 GX Developer 将以下参数设置到 C24/E71 安装站的可编程控制器 CPU 中。
· “其它站访问时的有效模块”设置
对其它站访问时经由的模块进行设置。
- C24/E71 安装站中安装了同相同络编号的多个网络模块时，指定了该网络编号的情况下，将经由基板的小编号的插槽中安装的网络模块进行其它站访问。
- 关于对同一个站通过多个外部设备及 GX Works2 或 GX Developer 等同时发出了访问请求时所必需的扫描次数，请参阅 2.4 节。
- 关于对网络系统上的其它站可编程控制器的允许访问范围的详细内容，请参阅网络系统的手册。
- 连接到 C24 上时，多点连接中包含有计算机链接模块时，应以通过 ASCII 代码进行的通信的各个帧进行访问。
对于通过二进制代码进行通信的 QnA 兼容 4C 帧(格式 5)，在包含至外部设备连接站的访问时在的经由多点连接的其它站访问中不能使用。
- 连接到 E71 上时，可以通过以下网络与其它站的可编程控制器 CPU 进行通信。
· CC-Link IE 控制网络
· CC-Link IE 现场网络
· MELSECNET/H
· MELSECNET/10
详细内容请参阅用户手册(应用篇)。

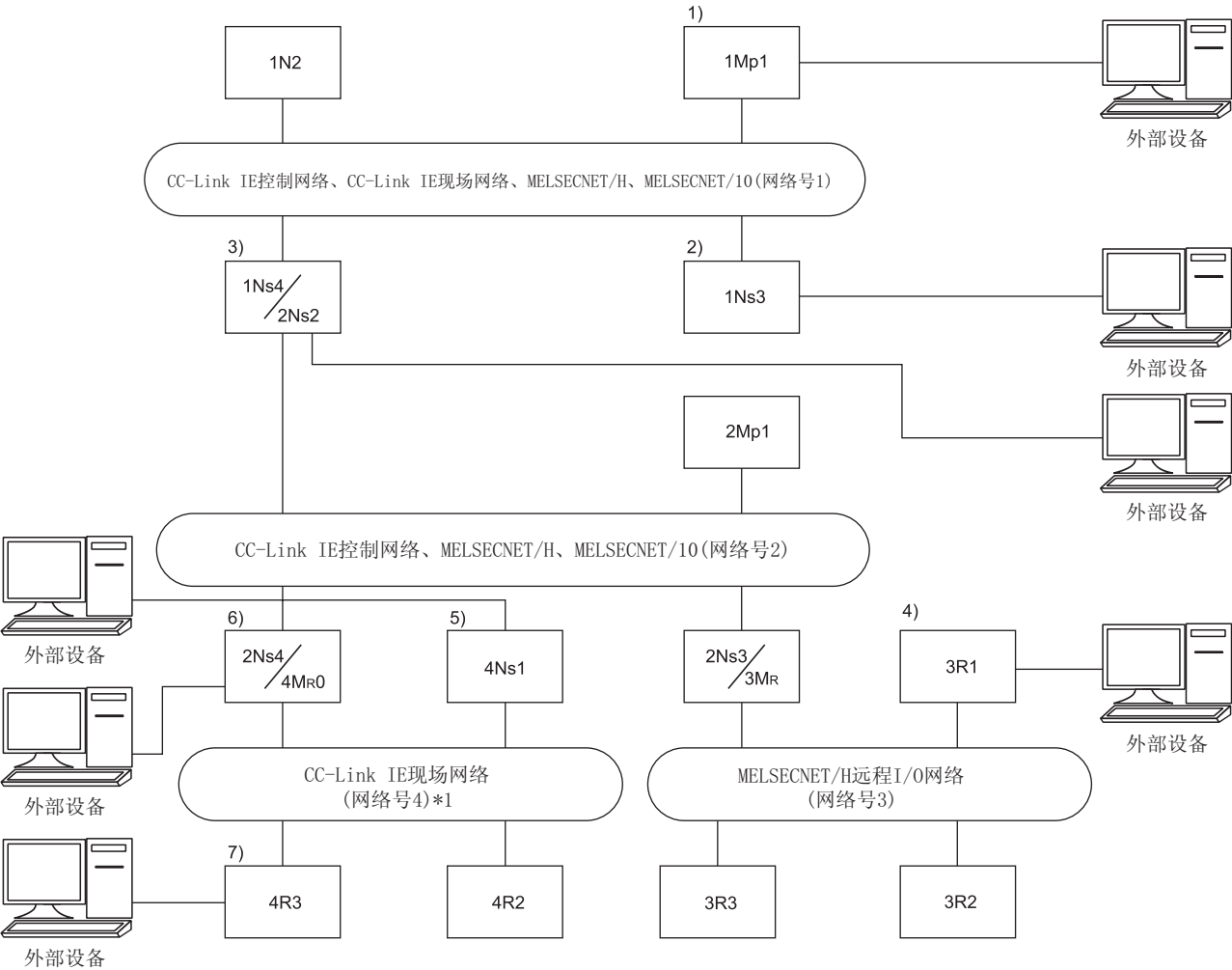
2.6.2 使用各帧时的可访问站的示例

以下介绍可对其它站可编程控制器进行访问的站的示例。
图中各站的符号如下所示。



* 对于网络系统中包含的 E71，定位于普通站(上述符号“Ns”)的位置处。

(与外部设备相连接的站为 Q/L/QnACPU 的情况下)



外部设备连接站		可从外部设备访问的站													
		1Mp1	1N2	1Ns3	1Ns4 2Ns2	2Mp1	2Ns3 3Mr	2Ns4 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3	
1)	A 兼容 1E 帧					×									
	A 兼容 1C 帧					×									
	QnA 兼容 2C 帧	○	×												
	QnA 兼容 3E 帧							○							
	4E 帧							○							
	QnA 兼容 3C 帧							○							
	QnA 兼容 4C 帧 (格式 1 ~ 4)							○							
	QnA 兼容 4C 帧 (格式 5)							○							
2)	A 兼容 1E 帧					×									
	A 兼容 1C 帧					×									
	QnA 兼容 2C 帧	×		○	×										
	QnA 兼容 3E 帧							○							
	4E 帧							○							
	QnA 兼容 3C 帧							○							
	QnA 兼容 4C 帧 (格式 1 ~ 4)							○							
	QnA 兼容 4C 帧 (格式 5)							○							
3)	A 兼容 1E 帧					×									
	A 兼容 1C 帧					×									
	QnA 兼容 2C 帧	×		○	×										
	QnA 兼容 3E 帧							○							
	4E 帧							○							
	QnA 兼容 3C 帧							○							
	QnA 兼容 4C 帧 (格式 1 ~ 4)							○							
	QnA 兼容 4C 帧 (格式 5)							○							
4)	A 兼容 1E 帧							×							
	A 兼容 1C 帧	×					×					×			
	QnA 兼容 2C 帧	×						○	×						
	QnA 兼容 3E 帧							○							
	4E 帧							○							
	QnA 兼容 3C 帧							○							
	QnA 兼容 4C 帧 (格式 1 ~ 4)							○							
	QnA 兼容 4C 帧 (格式 5)							○							

○：可以访问 (*1) ×：不能访问
(转下页)

外部设备连接站		可从外部设备访问的站											
		1Mp1	1N2	1Ns3	1Ns4 2Ns2	2Mp1	2Ns3 3Mr	2Ns4 4Mr0	3R1	3R2	3R3	4Ns1	4R2
5) ~ 6)	A 兼容 1E 帧	×											
	A 兼容 1C 帧	×											
	QnA 兼容 2C 帧	×											
	QnA 兼容 3E 帧	×											
	4E 帧	×											
	QnA 兼容 3C 帧	○											
	QnA 兼容 4C 帧 (格式 1~4)	○											
	QnA 兼容 4C 帧 (格式 5)	○											
7)	A 兼容 1E 帧	×										×	×
	A 兼容 1C 帧	×										×	×
	QnA 兼容 2C 帧	×										×	×
	QnA 兼容 3E 帧	×										×	×
	4E 帧	×										×	×
	QnA 兼容 3C 帧	○										×	○
	QnA 兼容 4C 帧 (格式 1~4)	○										×	○
	QnA 兼容 4C 帧 (格式 5)	○										×	○

○：可以访问 (*1) ×：不能访问

- *1 可通过 A 兼容 1E 帧或 A 兼容 1C 帧对 Q/L/QnACPU 进行访问的软件元件及其软件元件范围是有限制的。
 对于 Q/L/QnACPU，只能对与 AnCPU、AnNCPU、AnACPU、AnUCPU 中存在的软件元件相同名称的软件元件在 AnACPU 的软件元件范围内进行访问。
 ・使用 Q 系列 C24 时 参阅 5.2.1 项(2)
 ・使用 Q 系列 E71 时 参阅 6.3.1 项(2)
 不能对 Q/L/QnACPU 中新增的软件元件进行访问。

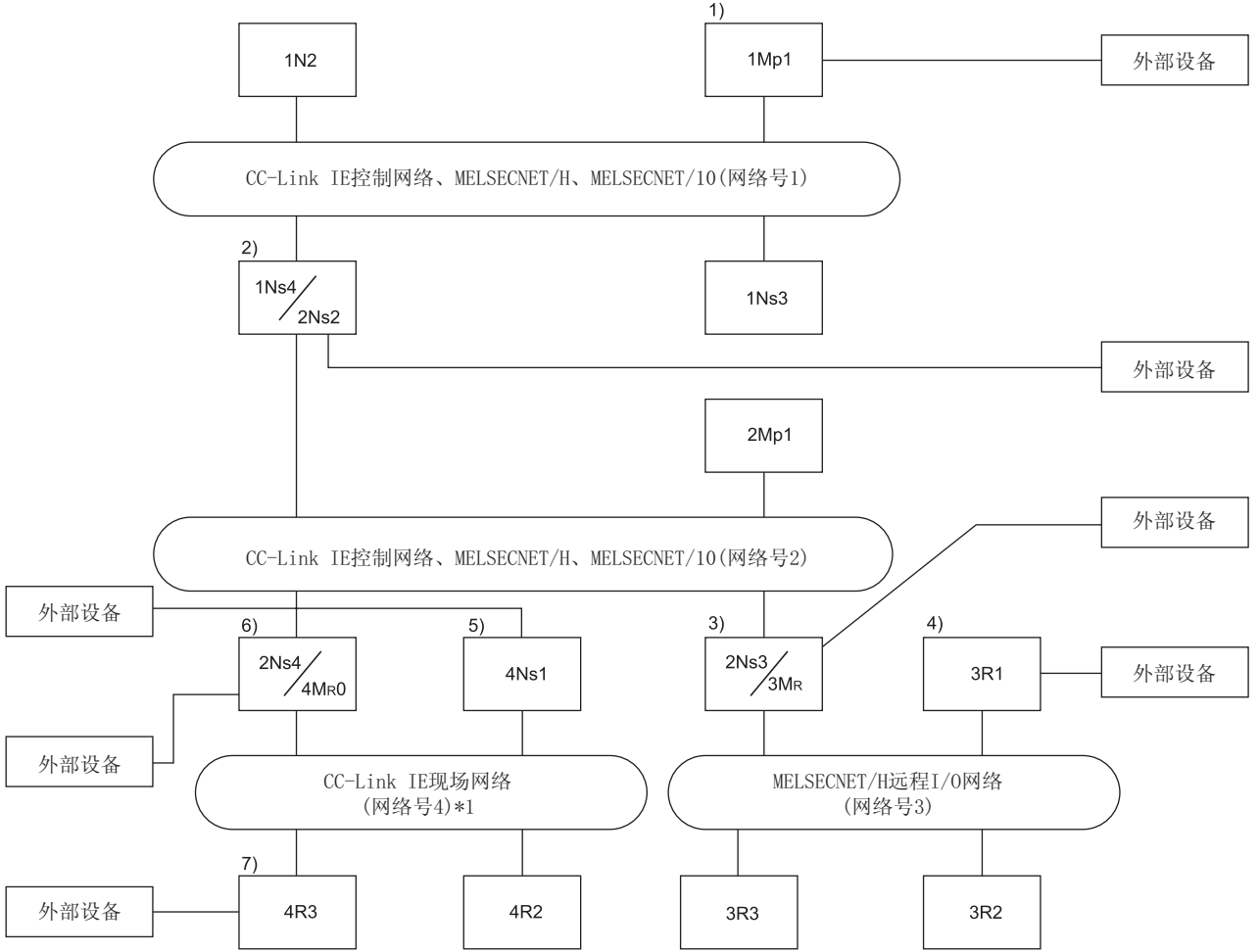
2.6.3 各数据通信帧内指定的其它站访问用数据项目的指定示例

以下介绍通过 MC 协议进行的通信中对 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的可编程控制器站进行访问时，各帧中指定的“网络编号”、“可编程控制器编号”、“请求目标模块 I/O 编号”、“请求目标模块站编号”的指定示例。

关于数据指定项目，请参阅 3.1 节。

(图·表的阅读方法)

- 关于说明符号的含义，请参阅 2.6.2 项。
- 表中的“从外部设备访问时的指定值”栏的数值是指，对表的上段显示的相应网络系统上的可编程控制器站进行访问时，对表的左侧中显示的数据项目的指定值。



外部设备连接站为 1) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □Mp	00 _H	01 _H			02 _H			03 _H			04 _H		
可编程控制器编号 Mp□	FF _H	02 _H	03 _H	04 _H	01 _H	03 _H	04 _H	01 _H	02 _H	03 _H	01 _H	02 _H	03 _H
请求目标模块 I/O 编号 □Spm	03FF _H												
请求目标模块站号 Spm□	00 _H												

外部设备连接站为 2) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □Mp	01 _H			00 _H	02 _H			03 _H			04 _H		
可编程控制器编号 Mp□	01 _H	02 _H	03 _H	FF _H	01 _H	03 _H	04 _H	01 _H	02 _H	03 _H	01 _H	02 _H	03 _H
请求目标模块 I/O 编号 □Spm	03FF _H												
请求目标模块站号 Spm□	00 _H												

外部设备连接站为 3) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □Mp	01 _H			02 _H		00 _H	02 _H	03 _H			04 _H		
可编程控制器编号 Mp□	01 _H	02 _H	03 _H	02 _H	01 _H	FF _H	04 _H	01 _H	02 _H	03 _H	01 _H	02 _H	03 _H
请求目标模块 I/O 编号 □Spm	03FF _H												
请求目标模块站号 Spm□	00 _H												

外部设备连接站为 4) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □Mp	01 _H			02 _H		03 _H	02 _H	00 _H	03 _H			04 _H	
可编程控制器编号 Mp□	01 _H	02 _H	03 _H	02 _H	01 _H	7D _H	04 _H	FF _H	02 _H	03 _H	01 _H	02 _H	03 _H
请求目标模块 I/O 编号 □Spm	03FF _H												
请求目标模块站号 Spm□	00 _H												

外部设备连接站为 5) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □ Mp	01 _H			02 _H		02 _H	04 _H	03 _H			00 _H	04 _H	
可编程控制器编号 Mp □	01 _H	02 _H	03 _H	02 _H	01 _H	03 _H	7D _H	01 _H	02 _H	03 _H	FF _H	02 _H	03 _H
请求目标模块 I/O 编号 □ Spm	03FF _H												
请求目标模块站号 Spm □	00 _H												

外部设备连接站为 6) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □ Mp	01 _H			02 _H		02 _H	00 _H	03 _H			04 _H		
可编程控制器编号 Mp □	01 _H	02 _H	03 _H	02 _H	01 _H	03 _H	FF _H	01 _H	02 _H	03 _H	01 _H	02 _H	03 _H
请求目标模块 I/O 编号 □ Spm	03FF _H												
请求目标模块站号 Spm □	00 _H												

外部设备连接站为 7) 时

数据名	从外部设备访问时的指定值												
	1Mp1	1N2	1Ns3	1Ns4/ 2Ns2	2Mp1	2Ns3/ 3Mr	2Ns4/ 4Mr0	3R1	3R2	3R3	4Ns1	4R2	4R3
网络编号 □ Mp	01 _H			02 _H		02 _H	04 _H	03 _H			04 _H		00 _H
可编程控制器编号 Mp □	01 _H	02 _H	03 _H	02 _H	01 _H	03 _H	7D _H	01 _H	02 _H	03 _H	01 _H	02 _H	FF _H
请求目标模块 I/O 编号 □ Spm	03FF _H												
请求目标模块站号 Spm □	00 _H												

2.7 数据通信时的注意事项

以下介绍外部设备与 C24/E71 之间数据通信时的注意事项。

2.7.1 使用 E71 时

(1) E71 的输入信号

E71 的以下信号为 ON 时进行读取/写入。

- (a) 使用自动打开 UDP 端口时
初始化正常完成信号(X19)
- (b) 使用用户打开端口时
初始化正常完成信号(X19)、使用的连接的打开完成信号(X10 ~ X17、缓冲存储器地址 5000H)

该信号为 ON 时，可以在无需顺控程序的状况下从外部设备通过 MC 协议进行通信。

(2) 在可编程控制器 CPU 的运行中进行数据写入的情况下

参阅 2.5 节，对用于执行可编程控制器 CPU 的运行中数据写入进行设置。

(3) 进行可编程控制器 CPU 的远程 STOP 的情况下

应使用自动打开 UDP 端口，或使用在 GX Works2 以及 GX Developer 的“动作设置” - “初始化时机设置”中设置为“常时打开等待”的 Passive(被动)打开的连接。

(4) 打开的连接用途设置为“无顺序”的情况下

不能通过 MC 协议进行通信。

用于通过 MC 协议进行通信的连接用途时，应设置为“有顺序”。

(5) 使用自动打开 UDP 端口的情况下

不能通过 ASCII 代码数据进行通信。

(6) 关于进行数据通信的其它站可编程控制器 CPU 的更换

E71 启动后对其它站可编程控制器 CPU 的信息进行获取及保持。

E71 启动后对进行数据通信的其它站可编程控制器 CPU 进行更换的情况下，可编程控制器 CPU 的型号改变时，应对 E71 进行重新启动。(本站可编程控制器的电源复位/CPU 复位)

(7) 关于 E71 的更换

以太网地址(MAC 地址)根据各设备而有所不同。

由于故障等对 E71 进行更换的情况下，应对外部设备也进行重新启动。

同样，对外部设备(个人计算机等)进行更换的情况下，对以太网模块也应进行重新启动。

(8) 将 E71 安装到 MELSECNET/H 远程 I/O 站中时

- (a) 应以 QnA 兼容 3E 帧、4E 帧进行通信。
不能通过 A 兼容 1E 帧进行通信。
- (b) 不能从 MELSECNET/H 远程主站侧经由安装在远程 I/O 站中的 E71 进行其它站访问，不能经由 E71 进行其它站访问。

(9) 使用 4E 帧，不等待以太网模块的响应报文，从外部设备连续发送指令报文的情况下

不要超过本项(10)中所示的指令报文的个数。

超过了本项(10)中所示的指令报文的个数的情况下，将变为下述状况。

- (a) TCP 协议的情况下
以太网模块的接收缓冲中将无可利用空间。(窗口大小=0)
因此，外部设备在以太网模块的接收缓冲中有可用空间之前，暂停以下指令报文的发送。
- (b) UDP 协议的情况下
以太网模块的接收缓冲将溢出，以太网模块中有可能发生通信出错，或以太网模块忽略指令报文，无法向外部设备侧返回响应报文。

(10) 使用 4E 帧，以太网模块可从外部设备连续接收指令报文并处理的个数

- (a) TCP 协议的情况下

	计算公式
1 个连接中可接收的指令报文个数	$(\text{TCP 窗口大小}[\text{字节}] \div \text{MC 协议的报文大小}[\text{字节}])^{*1*2}$

*1 将括号内的计算结果的小数点以下舍去化为整数。

*2 TCP 窗口大小如下所示。

QJ71E71-100 : 6144 字节
QJ71E71-B5, QJ71E71-B2 : 5120 字节

- (b) UDP 协议的情况下

	计算公式
1 个连接中可接收的指令报文个数	$1 + (\text{接收缓冲可存储的报文数} \div \text{使用的连接数})^{*1*2*3}$

*1 将括号内的计算结果的小数点以下舍去化为整数。

*2 接收缓冲中可存储的报文数如下所示。

QJ71E71-100 : 57
QJ71E71-B5, QJ71E71-B2 : 10

*3 使用的连接数如下所示。

1 ~ 17 个连接(用户打开端口+自动打开 UDP 端口)

2.7.2 使用 C24 时

(1) C24 的传送顺控程序变为初始状态的条件

C24 的传送顺控程序变为初始状态的条件如下所示。

- 投入电源时，操作 CPU 表面的复位开关时，或模式切换时。
- 对于指令报文接收的响应报文的发送完毕时。
- 接收了传送顺控程序的初始化请求时。
- 在 RS-232 侧的全双工通信中设置了“有 CD 端子检查”的状态下进行数据通信的情况下，CD 信号变为 OFF 时。

(2) 关于来自于 C24 的异常响应

在对本站地址的请求中检测出错误的时点，对 MC 协议的对象设备进行异常响应。因此，全双工通信时即使对象设备处于发送中的情况下也可能进行异常响应。

(3) 关于进行数据通信的其它站可编程控制器 CPU 的更换

C24 启动后对其它站可编程控制器 CPU 的信息进行获取及保持。

C24 启动后对执行数据通信的其它站可编程控制器 CPU 进行更换的情况下，可编程控制器 CPU 的型号改变时，应对 C24 进行重新启动。(本站可编程控制器的电源复位/CPU 复位)

(4) 关于外部设备侧发生的成帧出错

在未通过 RS-422/485 从 C24 向外部设备侧发送任何内容的状态下外部设备侧有可能发生成帧出错。(参阅用户手册(基本篇))

对于通过 C24 发送了 STX、ACK、NAK 之一之前的数据，在外部设备侧应将其跳过而不进行读取。

应在对用户手册(基本篇)中所示的 C24 侧的接口规格进行确认的基础上进行数据通信。

(5) 关于多个 C24 的安装

与各 C24 相连接的外部设备同时对可编程控制器 CPU 发出了访问请求时，对可编程控制器 CPU 的访问顺序是由可编程控制器 CPU 确定。

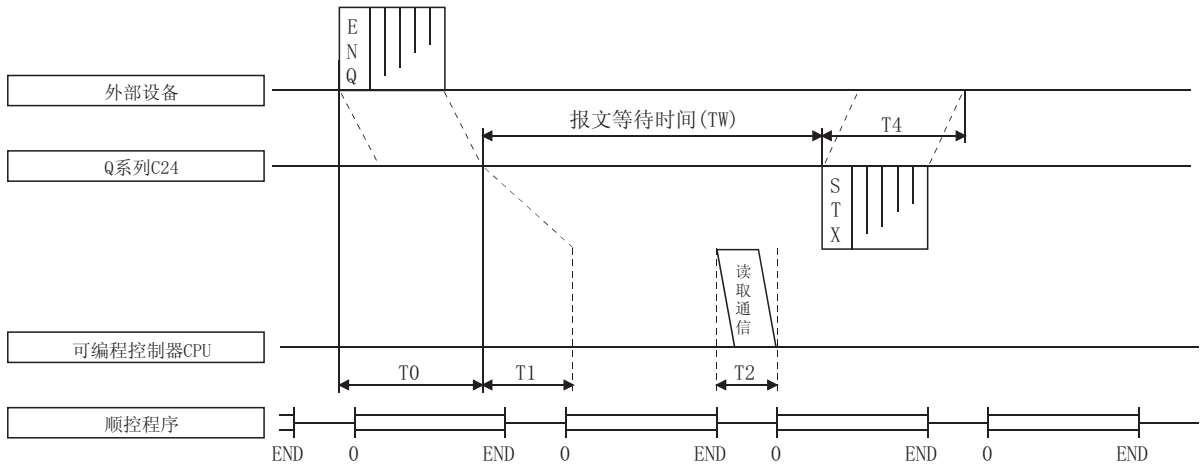
用户无法确定对该访问的优先顺序。

2.8 串行通信模块的传送顺控程序的时序图及通信时间

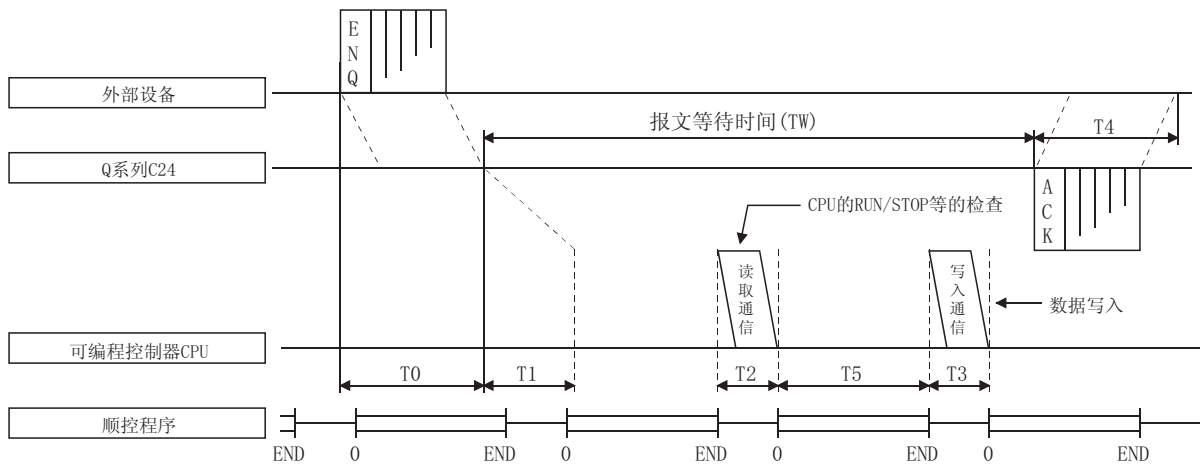
以下对外部设备经由 C24 访问可编程控制器时的通信时序图有关内容进行说明。

(1) 外部设备对 C24 安装站可编程控制器 CPU 进行访问的情况下

(a) 读取数据的情况下(设置了报文等待时间时的情况如下所示。)



(b) 写入数据的情况下(设置了报文等待时间时的情况如下所示。)



备注

- (1) C24 与可编程控制器 CPU 的通信总是在 END 后执行。
因此，扫描时间将延长相当于该通信时间(至可编程控制器 CPU 的中断时间)。
- (2) 对于至可编程控制器 CPU 的读取/写入请求时的处理所需的扫描次数，根据请求内容而有所不同。
例如，请求了需要 2 个扫描的读取的情况下，需要 1 个扫描+T2 的时间的余量。
- (3) 通用型 QCPU 的情况下，根据 GX Works2 或 GX Developer 的可编程控制器参数(可编程控制器系统设置)的服务处理设置，处理所需的扫描次数将改变。
关于服务处理设置的详细内容，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。
- (4) 关于读取/写入请求处理所需要的通信时间及扫描次数，请参阅附录 3。
关于 1 次通信中可执行的处理点数，请参阅第 3 ~ 6 章的指令列表说明项。

(2) 关于传送顺控程序的传送时间

以下对从外部设备开始进行数据传送起，至从 C24 返回结果为止的大致时间有关内容进行说明。

关于 T0 ~ T5 的内容，请参阅前页的 2.8(1)。

下述为上述大致时间的平均值，应将其作为处理时间的大致基准。

- (a) 外部设备侧从可编程控制器中读取数据的情况下(单位:ms)
通信时间 = 请求报文传送时间(T0)+{ (T1+(1.5 × 扫描时间)+T2)或 TW }+响应报文传送时间(T4)
(超过 TW 时)
- (b) 外部设备侧对可编程控制器进行数据写入的情况下
通信时间 = 请求报文传送时间(T0)+{ (T1+(1.5 × 扫描时间)+T2+T3+T5)或 TW }+响应报文传送时间(T4)
(超过 TW 时)

$$T0, T4 = (1000/\text{传送速度}) \times \text{传送时的 1 字节的位数} \\ \left(\underset{\substack{\uparrow \\ \text{启动位}}}{1} + \underset{\substack{\uparrow \\ \text{数据长度(7 或 8)}}}{7/8} + \underset{\substack{\uparrow \\ \text{奇偶位(0 或 1)}}}{0/1} + \underset{\substack{\uparrow \\ \text{停止位(1 或 2)}}}{1/2} \right) \times \text{字节数}$$

$$T1 = \text{Q 系列 C24 的处理时间} \\ \text{根据访问点数(MIN ~ MAX)变为下述范围的值。}$$

QJ71C24N (-R2/R4):	12.5 ~ 45.0ms
QJ71C24 (-R2):	13.0 ~ 70.0ms

$$T2, T3 = \text{CPU 中断时间(参阅附录 3)}^{*1}$$

$$T5 = \text{扫描时间}^{*1}$$

$$TW = \text{设置了报文等待时间时的设置时间}$$

*1 关于 1 个扫描中可处理的功能，变为 T3、T5=0。

2.9 经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 进行其它站访问时的传送时间

以下介绍对 CC-Link IE 控制网络、CC-Link IE 现场网络，MELSECNET/H、MELSECNET/10 上的其它站可编程控制器进行访问时的传送时间(T1)。

(1) CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H(可编程控制器网络)、MELSECNET/10(可编程控制器网络)的情况下

传送时间(T1) =
$$\frac{\text{传送延迟时间} + \text{C24/E71 安装站1扫描时间或内部处理时间}}{\text{超过内部处理时间时}} \times \frac{(n + 1)}{\text{}} \quad \begin{matrix} *1 \\ *2 \quad *3 \end{matrix}$$

内部处理时间 C24：约 50ms；E71：约 30ms

- *1 请参阅网络系统的手册内的传送延迟时间的说明项。
- *2
 - 对投入电源时或 CPU 复位操作后的相应站进行初次通信的情况下
 - 向除去最新通信的 16 站后的站进行通信的情况下
 - 通信站数为 16 站以下，进行第 2 次通信的情况下
 - 向最新通信的 16 站进行第 2 次通信的情况下
- *3 对 C24/E71 设置了“禁止运行中写入”的情况下，仅在从外部设备进行数据写入时进行加法运算。
(C24 是在 GX Works2 或 GX Developer 的开关设置中进行设置，E71 是在动作设置中进行设置。)

(示例)
MELSECNET/H 上的站中安装了 C24，对同一 MELSECNET/H 上的其它站的软元件存储器进行了读取的情况下。
(各处理时间/设置为以下情况下，进行通信站数为 8 站的第 2 次通信时)

- ST：发送侧扫描时间 12ms
- SR：接收侧扫描时间 10ms
- LS：链接扫描时间
- αT：发送侧链接刷新时间 5ms
- αR：接收侧链接刷新时间 5ms
- 同时瞬时请求：2
- 最大瞬时请求：1

(通常传送延迟时间 e: MELSECNET/H 模式)

传送时间(T1) =
$$\{(12 + 5 + 10 + 5) \times 2 + 9 \times 4 + ((2/1) - 1) \times 9 \times 2 + 50\} \times 1$$

(168ms) (ST • αT • SR • αR) (LS) (LS)

同时瞬时请求 —————

最大瞬时请求 —————

C24内部处理时间 —————

(最差传送延迟时间: MELSECNET/H 模式)

传送时间(T1) =
$$\{(12 + 5 + 10 + 5) \times 2 + 9 \times 6 + ((2/1) - 1) \times 9 \times 2 + 50\} \times 1$$

(186ms) (ST • αT • SR • αR) (LS) (LS)

同时瞬时请求 —————

最大瞬时请求 —————

C24内部处理时间 —————

- * 传送时间(T1)延迟原因
执行了需要 2 个扫描的传送指令的情况下，变为上述计算公式的 2 倍。
关于其它模块、GX Works2 或 GX Developer 等同时对同一个 Q/QnACPU 发出了访问请求时的有关内容，请参阅 2.4 节的[备注]。
- * 关于网络系统的详细内容请参阅网络系统的手册。

要点	
	至 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的本站以外的可编程控制器 CPU 的数据传送根据条件将会发生相应的延迟。 通过将外部设备设置为仅与 C24/E71 安装站的可编程控制器进行通信，与其它站可编程控制器的通信通过数据链接(LB、LW)进行，可以减少传送延迟时间。

(2) MELSECNET/H(远程 I/O 网络)、MELSECNET/10(远程 I/O 网络)的情况下

传送时间(T1) =
$$\frac{\text{传送延迟时间}}{*1} + \frac{1 \times \text{链接扫描时间或内部处理时间}}{\text{(超过内部处理时间时)}} \times \frac{(n + 1)}{*2 \quad *3}$$

内部处理时间 C24: 约 50ms , E71: 约 30ms

- *1 请参阅网络系统的参考手册(远程 I/O 网络篇)内的传送延迟时间的说明项。
- *2
 - 对链接开始后的相应站进行初次通信的情况下
 - 向除去最新通信的 16 站后的站进行通信的情况下

} n=6

- 通信站数为 16 站以下, 进行第 2 次通信的情况下
- 向最新通信的 16 站进行第 2 次通信的情况下

} n=1

- *3 对 C24/E71 设置了“禁止运行中写入”的情况下, 仅在从外部设备进行数据写入时进行加法运算。
(C24 是在 GX Works2 或 GX Developer 的开关设置中进行设置, E71 是在动作设置中进行设置。)

(示例)
MELSECNET/H(远程 I/O 网络)上的站中安装了 C24, 对同一 MELSECNET/H 上的其它站的软元件存储器进行了读取的情况下。
(各处理时间/设置为以下情况下, 进行通信站数为 8 站的第 2 次通信时)

- Sm : 远程主站的顺控程序的扫描时间 12ms
- αm : 远程主站的链接刷新时间 5ms
- LS : 链接扫描时间 9ms

(示例)的情况下, 由于(Sm) > (LS), 因此传送时间(T1)变为以下值。
(主站为 1 站时)

传送时间(T1) =
$$\{(12 + 5) \times 3 + 50\} \times 1$$

(101ms) (Sm · αm)

- * 传送时间(T1)延迟原因
执行了需要 2 个扫描的传送指令的情况下, 变为上述计算公式的 2 倍。
关于其它模块、GX Works2 或 GX Developer 等同时对同一个 Q/QnACPU 发出了访问请求时的有关内容, 请参阅 2.4 节的备注。
- * 关于网络系统的详细内容请参阅网络系统的参考手册(远程 I/O 网络篇)。

要点
至 MELSECNET/H 上的本站以外的可编程控制器 CPU 的数据传送根据条件将会发生相应的延迟。 通过将外部设备设置为仅与 C24/E71 安装站的可编程控制器进行通信, 与其它站可编程控制器的通信通过数据链接(LB、LW)进行, 可以减少传送延迟时间。

2.10 与多 CPU 系统的兼容

以下对 QCPU 为多 CPU 系统结构时通过 MC 协议进行通信的有关内容进行说明。

要点
QCPU 为多 CPU 系统的结构时请阅读本节。QCPU 的多 CPU 系统中使用 C24/E71 的情况下，请先参阅 QCPU 用户手册(多 CPU 系统篇)。

(1) 至本站多 CPU 系统的访问

- (a) 在多 CPU 系统中使用 C24/E71 时，应使用功能版本 B 的 C24/E71。
- (b) 在从外部设备对本站的多 CPU 系统进行访问时，可以对 C24/E71 的管理 CPU(通过 GX Works2 或 GX Developer 进行设置)以及非管理 CPU 进行访问。
 - * 根据访问目标管理 CPU、非管理 CPU 可使用的指令有所不同。
 - * 功能版本 A 的 C24/E71 也可使用，QCPU 的 1 号机将变为 C24/E71 的管理 CPU。
 在这种情况下，从外部设备对本站的 QCPU 进行访问时，只能对 C24/E71 的管理 CPU 进行访问。

(2) 至其它站多 CPU 系统的访问

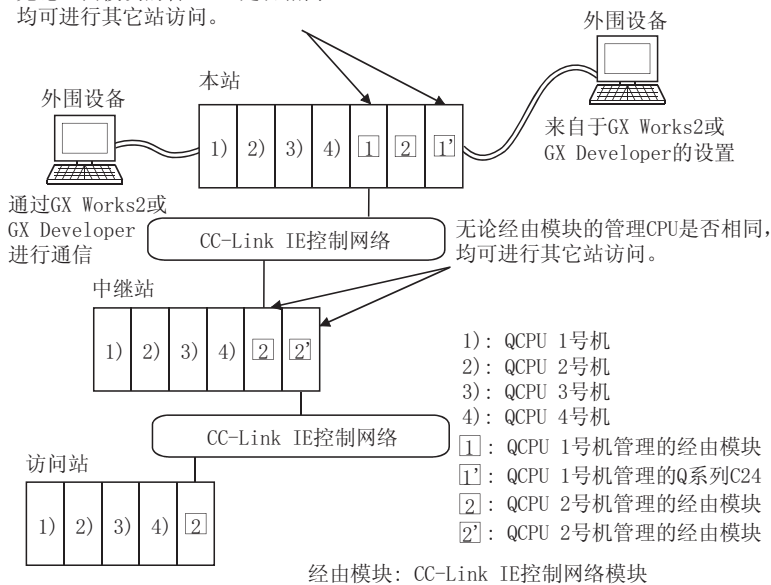
- (a) 对于以下各站的相应模块，应使用功能版本 B 的模块。

	模块	备注
本站	QCPU	—
	与外部设备连接的 C24/E71	将 C24/E71 与网络模块的管理 CPU 设置为同一个。
	与中继站/访问站连接的网络模块	
中继站	QCPU	—
	将本站侧与访问站侧的网络相连接的 2 个模块	
访问站	QCPU	—
	与中继站连接的网络模块	

- (b) 对其它站(访问站)的多 CPU 系统进行访问时，可以对与中继站相连接的网络模块的管理 CPU 以及非管理 CPU 进行访问。
 - * 根据访问目标 QCPU(管理 CPU、非管理 CPU)，可使用的指令有所不同。
 - * 从外部设备进行其它站访问时，在本站·中继站·访问站的上述模块中包含有功能版本 A 的模块时，只能对与中继站相连接的网络模块的管理 CPU 进行访问。此外，可以经由同一管理 CPU 所管理的模块进行其它站访问。
 - * 可以经由网络进行其它站访问的模块如下所示。
 - CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 网络模块
 - C24 · E71

(示例)

无论经由模块的管理CPU是否相同，
均可进行其它站访问。



要点

通过预先设置路由参数(通信路径)，也可访问下述网络系统的可编程控制器。

- CC-Link IE 控制网络
- CC-Link IE 现场网络
- MELSECNET/H
- MELSECNET/10
- 以太网

可访问范围最多为 8 个网络目标(中继站数：7 站)。

(3) 访问多 CPU 系统时的 QCPU 的处理时间

(a) 对管理 CPU 进行访问的情况下

关于访问下述管理 CPU 时的可编程控制器 CPU 侧的处理时间，与附录 3 中所示的访问单 CPU 系统时的处理时间相同。(*1)

- 本站 C24/E71 的管理 CPU
- 其它站网络模块的管理 CPU

(b) 对非管理 CPU 进行访问的情况下

关于访问下述非管理 CPU 时的可编程控制器 CPU 侧的处理时间，与访问单 CPU 系统时相比，需要额外的处理时间(每个指令约 9ms)。(*1)

- 本站 C24/E71 的非管理 CPU
- 其它站网络模块的非管理 CPU

*1 访问多 CPU 系统时每个指令的处理时间由于下述原因会有进一步的延迟。详细内容请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。

- QCPU 的动作状态(运行 N 中的情况下)
- QCPU 之间的自动刷新功能的使用状态、执行状态
- QCPU 与智能功能模块之间的访问状态、自动刷新状态

(4) MC 协议用通信帧及可访问的站

(a) 包括至多 CPU 系统的访问在内的 MC 协议用通信帧及可访问站如下所示。
对多 CPU 系统的非管理 CPU 进行访问时，应以下述帧进行访问。

- C24: QnA 兼容 4C 帧(格式 1 ~ 格式 5)
- E71: QnA 兼容 3E 帧、4E 帧

表中的范围 1 ~ 范围 4 与下图所示的范围 1 ~ 范围 4 相对应。

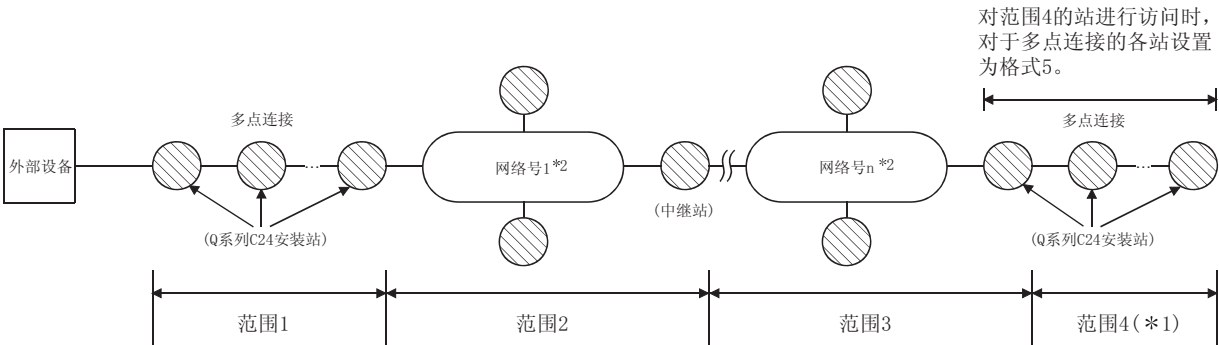
表中的管理/非管理表示访问站为多 CPU 系统时的 QCPU，其含义如下所示。

- 管理：表示管理与外部设备相连接的 C24/E71 或经由的网络模块的 QCPU。
- 非管理：表示未管理与外部设备相连接的 C24/E71 或经由网络模块的 QCPU。

对象模块	使用的帧		可以访问的站							
			范围 1		范围 2		范围 3		范围 4	
			管理	非管理	管理	非管理	管理	非管理	管理	非管理
C24	QnA 兼容 3C 帧	格式 1 ~ 4	○	×	○	×	○	×	×	
	QnA 兼容 4C 帧	格式 1 ~ 4	○		○		○		○	×
		格式 5	○		○		○		○	×
		QnA 兼容 2C 帧	格式 1 ~ 4	○	×	×		×		×
	QnA 兼容 1C 帧	○		×	○	×	×		×	
E71	4E 帧		○		○		○		○	×
	QnA 兼容 3E 帧		○		○		○		○	×
	A 兼容 1E 帧		○	×	○	×	×		×	

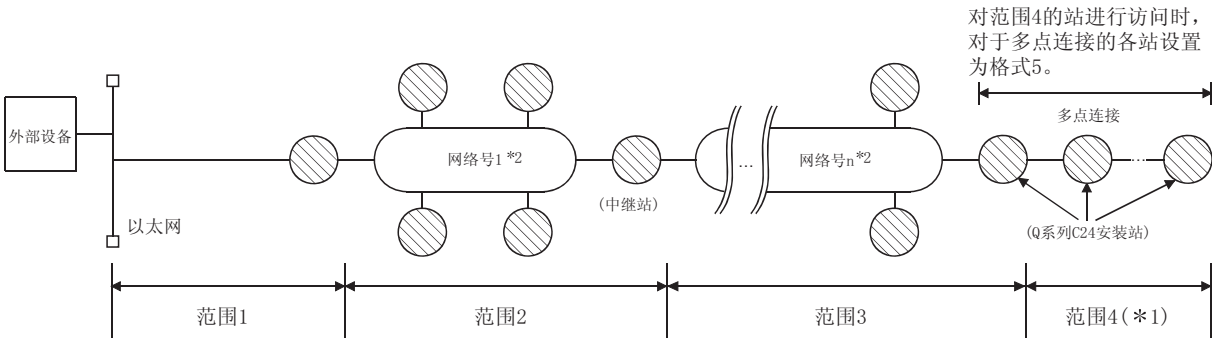
○：可以访问 ×：不能访问

[使用 C24 时]



- *1 在上图中，范围 4 的站为多 CPU 系统的情况下，只能对相应站的 C24 的管理 CPU 进行访问。
- *2 网络表示 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网。

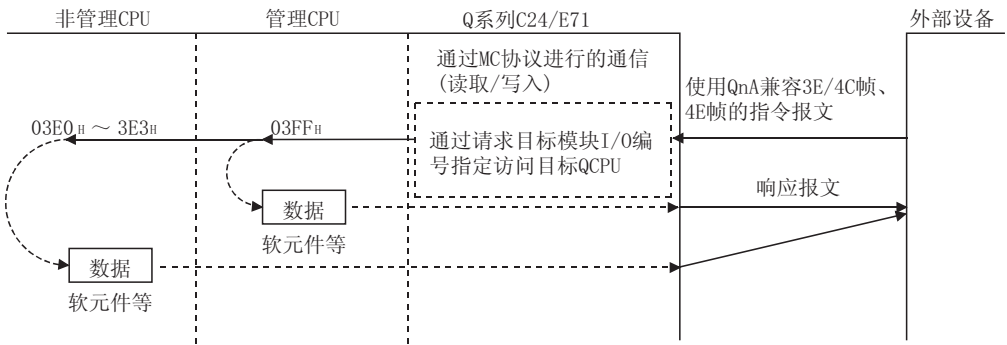
[使用 E71 时]



- *1 在上图中，范围 4 的站为多 CPU 系统的情况下，只能对相应站的 C24 的管理 CPU 进行访问。
- *2 网络表示 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网。

(b) 使用 MC 协议用 QnA 兼容 3E/4C 帧、4E 帧时，多 CPU 系统的访问对象 QCPU 是在 QnA 兼容 3E/4C 帧、4E 帧中的“请求目标模块 I/O 编号”的数据项目进行指定。

	外部设备的访问站	外部设备指定的请求目标模块 I/O 编号
1	管理 CPU	03FF _H
2	多 CPU 1 号机	03E0 _H
3	多 CPU 2 号机	03E1 _H
4	多 CPU 3 号机	03E2 _H
5	多 CPU 4 号机	03E3 _H

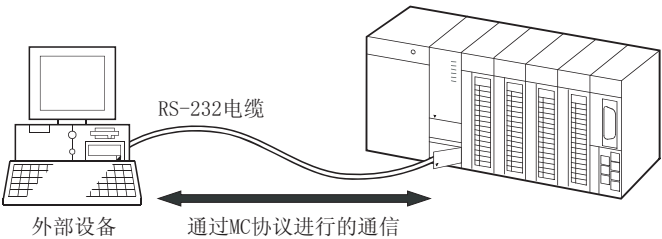


- (c) 可以通过 MC 协议对管理 CPU/非管理 CPU 以及智能功能模块的缓冲存储器进行访问。
根据管理 CPU、非管理 CPU 可使用的访问功能有所不同。
关于各指令的详细内容，请参阅 3.2 节的 QnA 兼容 3E/3C/4C 帧、4E 帧用指令及功能列表。

2.11 与 CPU 模块的串行通信功能的兼容

(1) 串行通信功能

- (a) 串行通信功能是指，将 CPU 模块的 RS-232 接口与对象设备相连接，通过 MC 协议的通信将 CPU 模块的动作通过对象设备侧进行监视、控制的功能。
关于串行通信功能，请参阅所使用的 CPU 模块的用户手册
(功能解说/程序基础篇)。



- (b) 串行通信功能可在下述 CPU 模块中使用。
- 基本型 QCPU(仅 Q00/Q01CPU)
 - 通用型 QCPU(仅具有 RS-232 连接器的 CPU 模块)
- 此外，可使用的 CPU 模块根据 CPU 模块的版本而有所不同。
关于 CPU 模块的版本，请参阅所使用的 CPU 模块的用户手册
(硬件设计/维护点检篇)。
- (c) 可通过串行通信功能进行通信仅为对象设备及通过 RS-232 接口连接的 CPU 模块。不能经由 C24 及 CC-Link IE 控制网络等对其它站进行访问。
- (d) 通过 GX Works2 或 GX GX Developer 进行设置可以使用串行通信功能。
此外，关于可使用的 GX Works2 或 GX GX Developer 的版本，请参阅所使用的 CPU 模块的用户手册(硬件设计/维护点检篇)。

(2) 关于 MC 协议用通信帧

- (a) 通过串行通信功能对 Q00/Q01CPU 进行访问的情况下，可以通过以下的帧进行访问。
关于各个帧的格式，请参阅 3.1.4 项、3.1.5 项。

功能		格式 4	格式 5
通过 ASCII 代码的通信	QnA 兼容 3C 帧	○	×
	QnA 兼容 4C 帧	○	×
通过二进制代码的通信	QnA 兼容 4C 帧	×	○

○：可以使用，×：不能使用

(b) 以下介绍串行通信功能中可使用的指令及功能。此外，关于可访问的软元件及软元件编号范围，请参阅 3.3.1 项(3)。

功能			指令(子指令)	处理内容	访问处理点数
软元件存储器	批量读取	位单位	0401(00□1)	将位软元件(X、Y、M等)以1点为单位进行读取。	ASCII:3584点 二进制:7168点
		字单位	0401(00□0)	将位软元件(X、Y、M等)以16点为单位进行读取。 将字软元件(D、R、T、C等)以1点为单位进行读取。	480字(7680点) 480点
	批量写入 ^{*1}	位单位	1401(00□1)	以1点为单位对位软元件(X、Y、M等)进行写入。	ASCII:3584点 二进制:7168点
		字单位	1401(00□0)	以16点为单位对位软元件(X、Y、M等)进行写入。 以1点为单位对字软元件(D、R、T、C等)进行写入。	480字(7680点) 480点
	随机读取 ^{*2}	字单位	0403(00□0)	将位软元件(X、Y、M等)以16点、32点为单位随机指定软元件、软元件编号进行读取。	96点
				将字软元件(D、R、T、C等)以1点、2点为单位随机指定软元件、软元件编号进行读取。	
	测试 ^{*1} (随机写入)	位单位	1402(00□1)	将位软元件(X、Y、M等)以1点为单位随机指定软元件、软元件编号进行设置/复位。	94点
		字单位 ^{*2}	1402(00□0)	将位软元件(X、Y、M等)以16点为单位随机指定软元件、软元件编号进行设置/复位。 对字软元件(D、R、T、C等)以1点、2点为单位随机指定软元件、软元件编号进行写入。	*5
	监视数据登录 ^{*2*3*4}	字单位	0801(00□0)	将进行监视的位软元件(X、Y、M等)以16点为单位进行登录。	96点
				对进行监视的字软元件(D、R、T、C等)以1点、2点为单位进行登录。	
	监视	字单位	0802(0000)	对进行了监视数据登录的软元件进行监视。	(登录点数)

*1 运行中写入的情况下，将“运行中写入设置”设置为“允许”。

*2 对于 TS、TC、SS、SC、CS、CC 的软元件，不能进行字单位的指定。
在监视登录中进行了指定的情况下，执行监视时将变为出错(4032H)状态。

*3 在监视登录中，不能对监视条件进行设置。

*4 不要从多个对象设备执行监视登录。
执行时最后的监视登录将有效。

*5 在以下范围内设置访问处理点数。

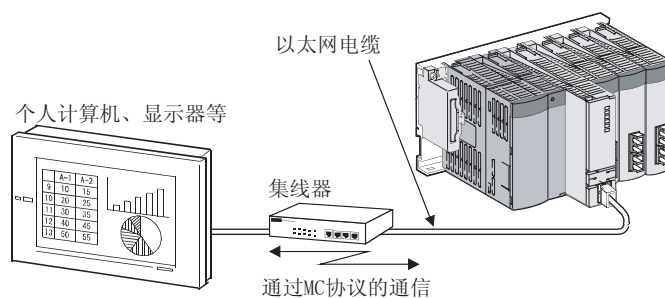
$$(\text{字访问点数}) \times 12 + (\text{双字访问点数}) \times 14 \leq 960$$

· 对于位软元件，在字访问时1点为16位，双字访问时1点为32位。

· 对于字软元件，在字访问时1点为1字，双字访问时1点为2字。

2.12 与以太网端口内置 CPU 的兼容

在以太网端口内置 CPU 中，可以使用内置以太网端口通过 MC 协议进行通信。
 可以从个人计算机、显示器等的外部设备使用 MC 协议对 CPU 模块的软元件数据进行读取/写入。
 通过对软元件数据进行读取/写入，在个人计算机、显示器等中进行 CPU 模块的动作监视及数据分析、生产管理等。
 此外，通过远程口令功能，可以防止来自于外部的非法访问。



要点

可以从个人计算机、显示器等通过 MC 协议进行通信的仅为处于连接状态的 CPU 模块。

不能对下述 CPU 模块进行访问。

- 经由网络对其它站 CPU 的访问
- 多 CPU 系统配置时对其它机号 CPU 的访问

备注

关于用于 MC 协议通信的设置、指令、注意事项等有关内容，请参阅下述手册。

- QnUCPU 用户手册(内置以太网端口通信篇)
- MELSEC-L CPU 模块用户手册(内置以太网功能篇)

备忘录

[illegible]

3 通过 QnA 兼容 3E/3C/4C 帧、4E 帧进行通信时

在本章中，对通过以下帧对 C24/E71 进行 MC 协议的数据通信时的报文的数据格式、数据指定方法、限制等有关内容进行说明。
各 C24/E71 的可通信帧如下所示。

帧的类型	E71	C24	备注
QnA 兼容 3E 帧	可以通信	不能通信	与 QnA 系列 QE71 的帧相同
4E 帧	可以通信	不能通信	-
QnA 兼容 3C 帧	不能通信	可以通信	与 QnA 系列 QC24(N) *1 的 QnA 帧相同
QnA 兼容 4C 帧	不能通信	可以通信	与 QnA 系列 QC24(N) *1 的 QnA 扩展帧相同

*1 表示 AJ71QC24、AJ71QC24-R2、AJ71QC24-R4、A1SJ71QC24、A1SJ71QC24-R2、AJ71QC24N、AJ71QC24N-R2、AJ71QC24N-R4、A1SJ71QC24N、A1SJ71QC24N-R2。

3.1 报文格式

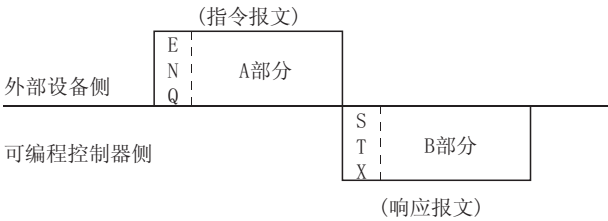
以下介绍以 QnA 兼容 3E/3C/4C 帧、4E 帧进行数据通信时，各指令的报文格式有关内容。

3.1.1 指令说明项的阅读方法

以下介绍 3.3.2 项 ~ 3.18 项中所示的各指令说明项的报文说明图的阅读方法。
关于以后各控制步骤的说明图中所示的传送数据的阅读方法，以与 C24 进行通信时为例进行说明。

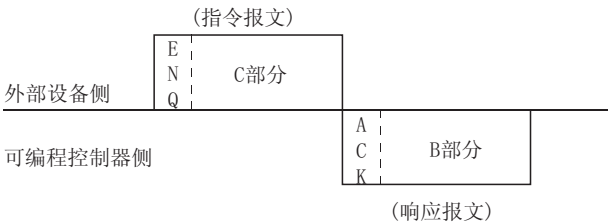
3

(1) 外部设备通过可编程控制器进行数据读取的情况下



- 1) A 部分表示由外部设备至可编程控制器 CPU 的传送。
- 2) B 部分表示由可编程控制器 CPU 至外部设备的传送。
- 3) 创建外部设备的程序时应编制为各数据按从左至右的顺序进行传送。
(示例：A 部分的情况下，按从 ENQ 开始向右的顺序进行数据发送。)

(2) 从外部设备对可编程控制器进行数据写入的情况下



- 1) C 部分表示由外部设备至可编程控制器 CPU 的传送。
- 2) B 部分表示由可编程控制器 CPU 至外部设备的传送。
- 3) 创建外部设备的程序时应编制为各数据按从左至右的顺序进行传送。
(示例：C 部分的情况下，按从 ENQ 开始向右的顺序进行数据发送。)

要点	
如果从外部设备接收指令报文，则可编程控制器 CPU 侧在对报文中的 A 部分/C 部分的处理完毕后，发送响应报文并进入下一个指令报文的接收等待(空档状态)。	

3.1.2 QnA 兼容 3E 帧、4E 帧的报文格式、控制步骤

以下介绍通过 E71 用的 QnA 兼容 3E 帧、4E 帧进行数据通信时的报文格式、控制步骤有关内容。

(1) 数据格式

用于 E71 与外部设备之间进行通信的数据格式是由“头部”及“应用数据”所构成。



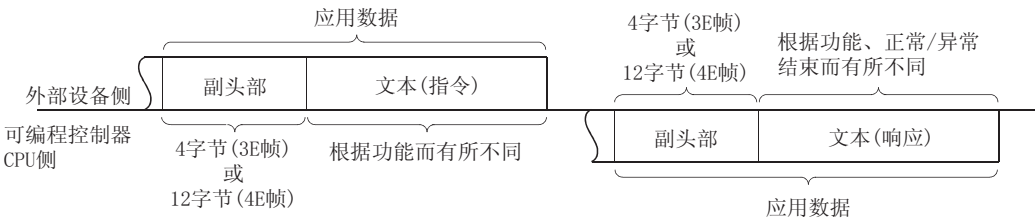
(2) 头部

是 TCP/IP、UDP/IP 用的头部。
对于外部设备侧→可编程控制器 CPU 侧(指令报文)的头部，应附加到外部设备侧进行发送。
对于可编程控制器 CPU 侧→外部设备侧(响应报文)的头部，由于 E71 会自动附加，因此无需由用户进行设置。

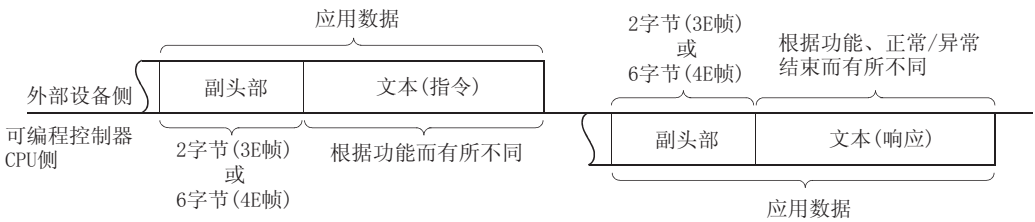
(3) 应用数据

应用数据大致分为“副头部”及“文本”。
副头部表示是指令报文还是响应报文。(请参阅本项(4)。)
文本是各功能中的请求数据(指令)、响应数据(响应)。(关于各功能的详细内容，请参阅 3.2 节以后的内容。)

(a) 通过 ASCII 代码进行通信时



(b) 通过二进制代码进行通信时



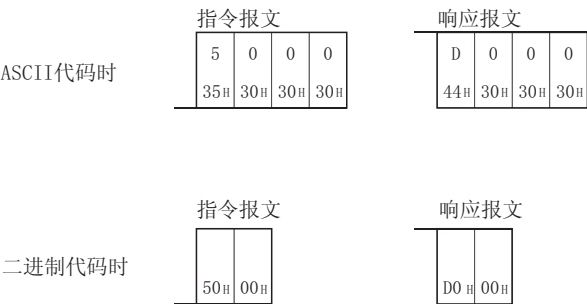
备注

对于来自于外部设备的指令的响应，是由 E71 创建并返送，因此无需由用户进行设置。

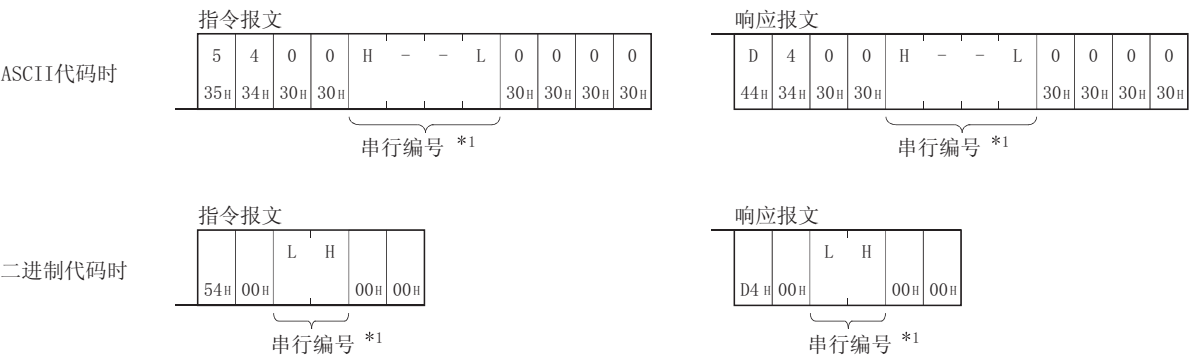
(4) 副头部的结构

副头部部分的结构如下所示。

(a) QnA 兼容 3E 帧



(b) 4E 帧



*1 串行编号(设置范围: 0000 ~ FFFF_H)
外部设备中附加的用于识别报文的任意编号。
对于以太网模块, 接收了指令时对串行编号进行存储, 返送响应时附加存储的串行编号后发送响应报文。

要点	
对于串行编号的使用, 应设置为由外部设备侧进行管理。 以太网模块进对串行编号的设置范围进行检查。	

(5) 控制步骤

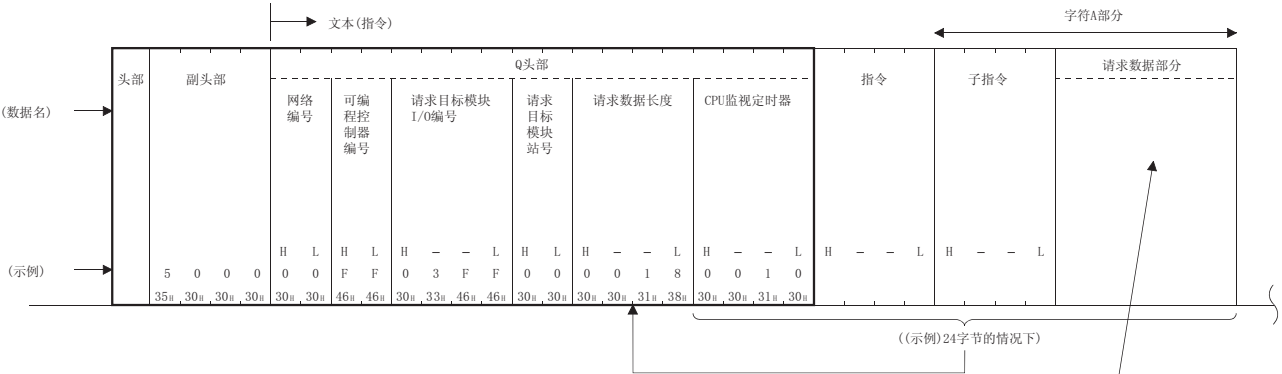
以下介绍 QnA 兼容 3E 帧、4E 帧中的控制步骤、应用数据部分的格式。
本项的报文说明图中所示的 □ 部分为各指令通用, 与本章的 3.3.2 项以后所示的报文说明图的*部分相对应。
关于 □ 部分的数据的内容、数据的指定方法, 请参阅 3.1.3 项。

要点	
通信时的数据代码是在 GX Works2 或 GX Developer 的 “ 以太网动作设置 ” 的通信数据代码设置中确定。	

(a) 以 QnA 兼容 3E 帧的 ASCII 代码进行通信时

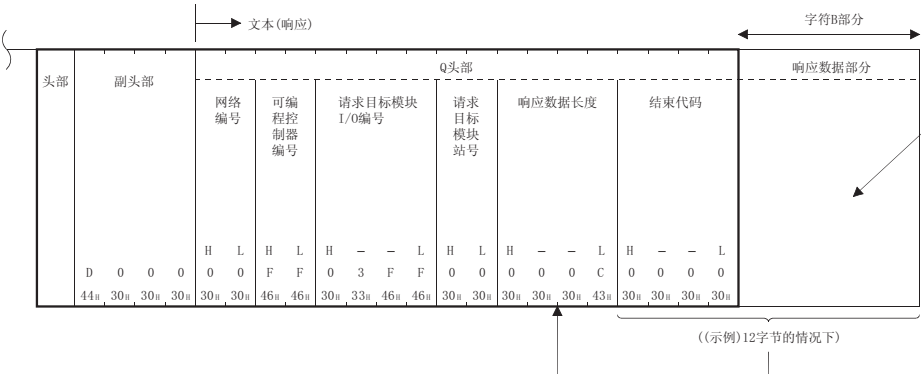
1) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

外部设备→可编程控制器CPU侧(指令报文)



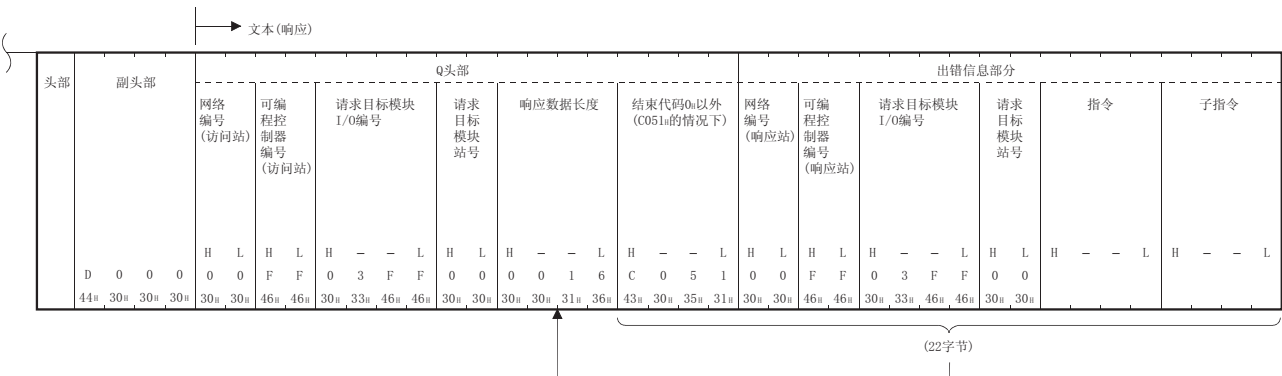
可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)



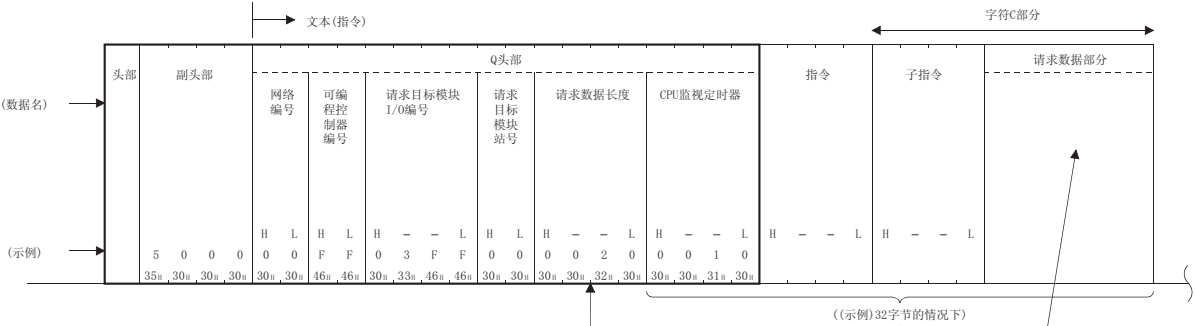
数据项目的排列根据指令、子指令而有所不同。详细内容请参阅本章3.3.2项以后所示的指令说明项。

(异常结束时)



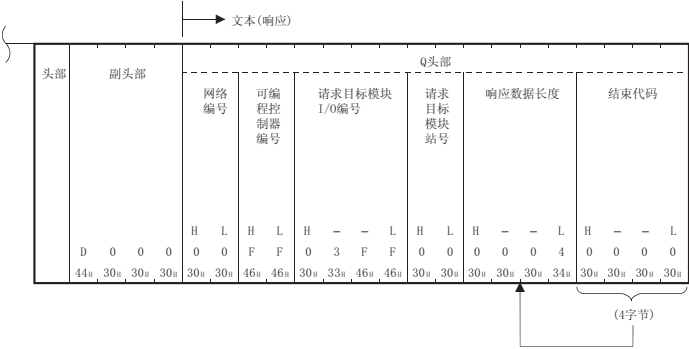
2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



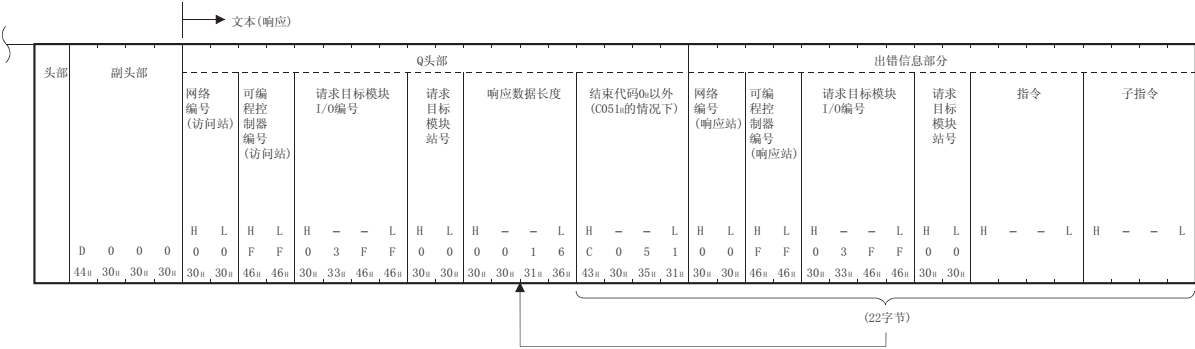
可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)



数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

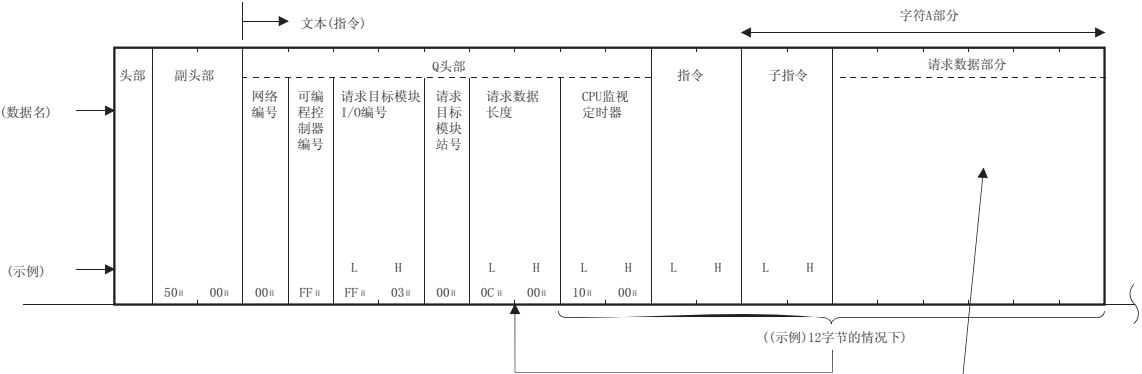
(异常结束时)



(b) 通过 QnA 兼容 3E 帧的二进制代码进行通信时

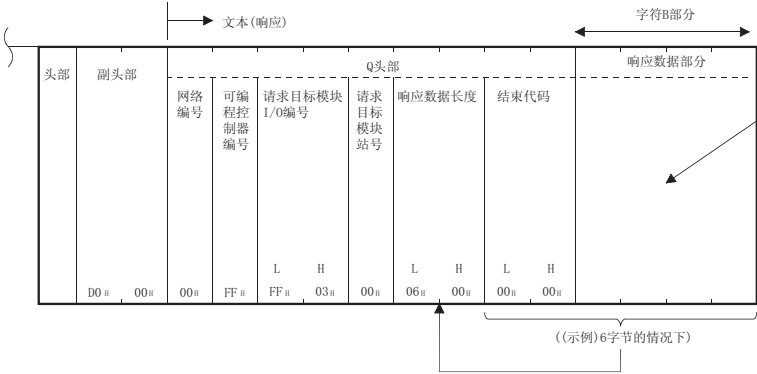
1) 外部设备对本站可编程控制器 CPU 进行数据读取的情况下

外部设备→可编程控制器CPU侧(指令报文)



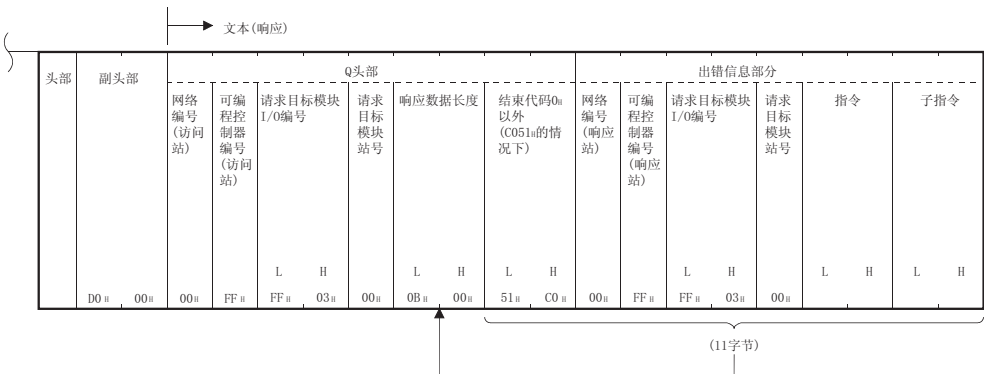
可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)

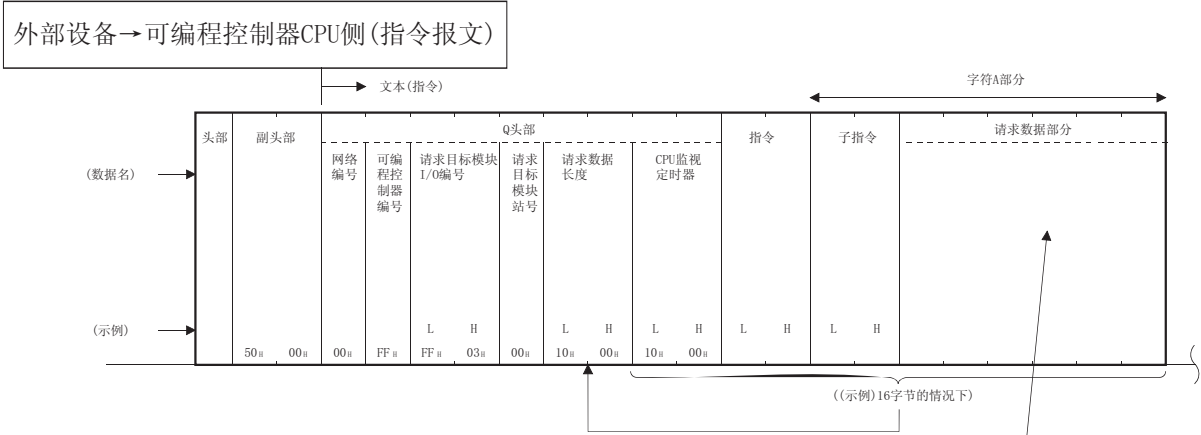


数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

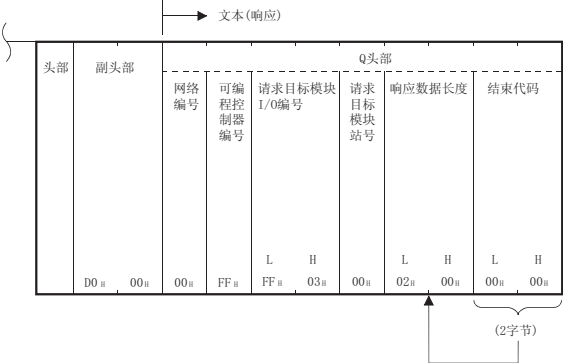
(异常结束时)



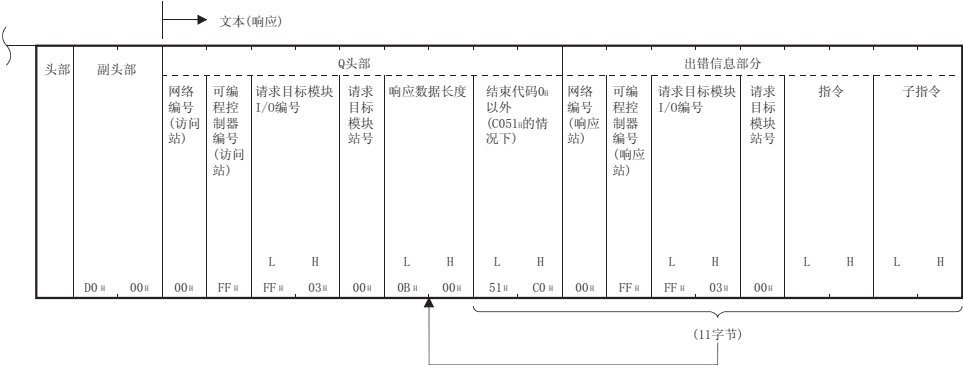
2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



(正常结束时)



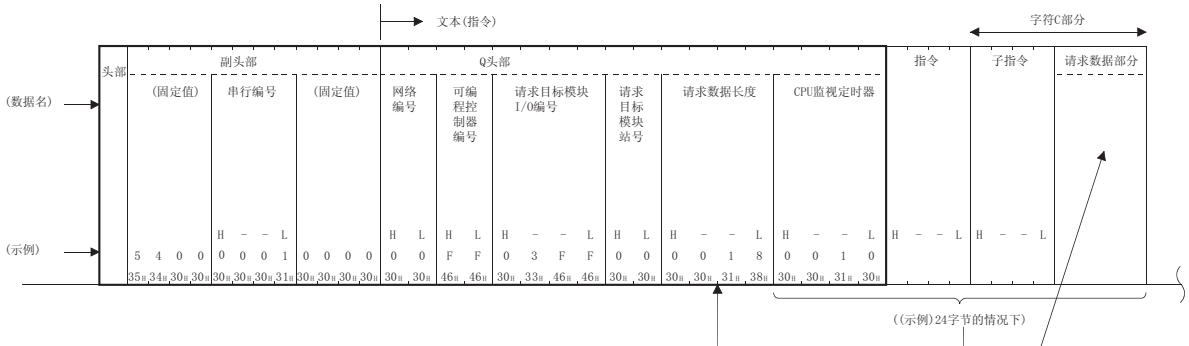
(异常结束时)



(c) 通过 4E 帧的 ASCII 代码进行通信时

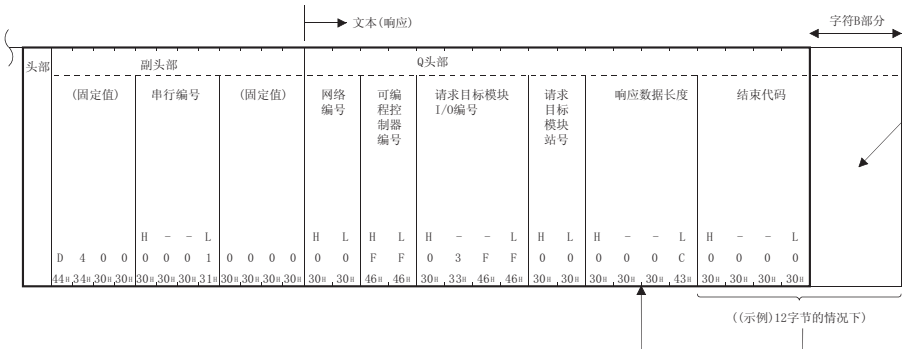
1) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

外部设备→可编程控制器CPU侧(指令报文)



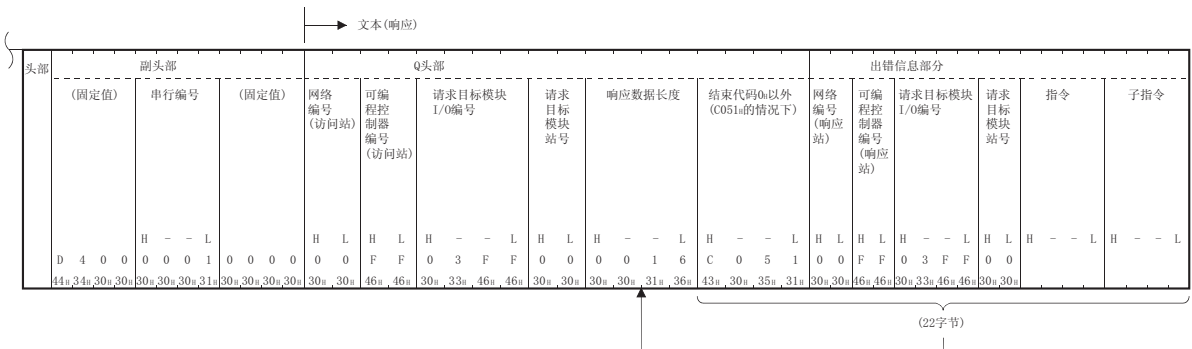
可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)

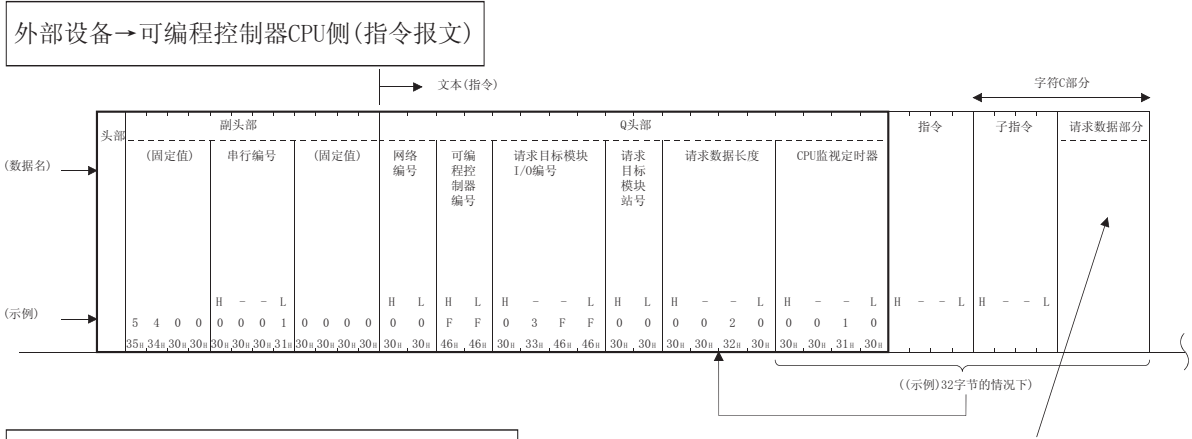


数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章 3.3.2 项以后所示的指令说明项。

(异常结束时)



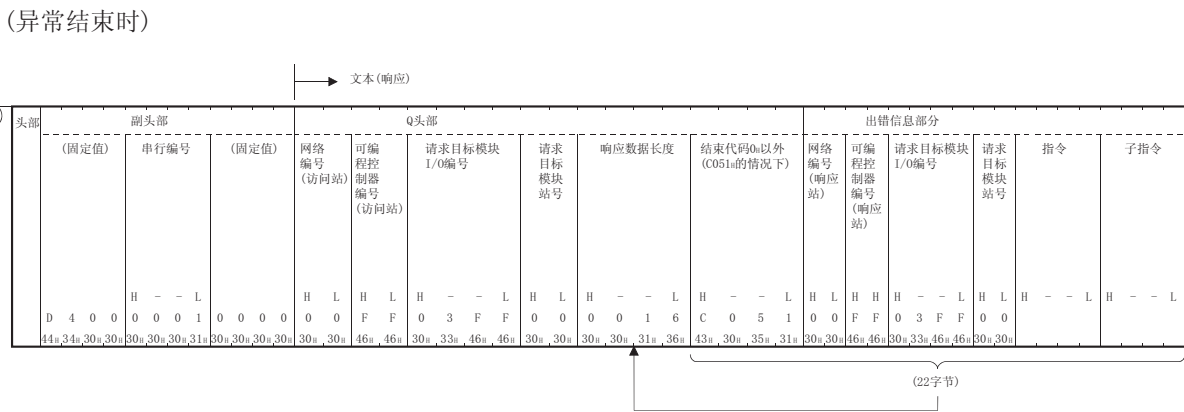
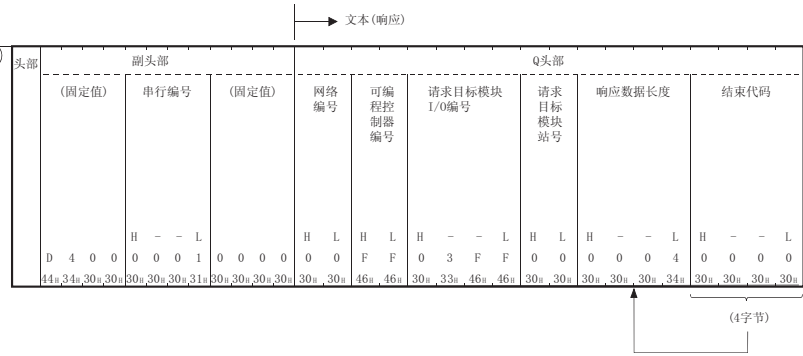
2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



可编程控制器CPU侧→外部设备侧(响应报文)

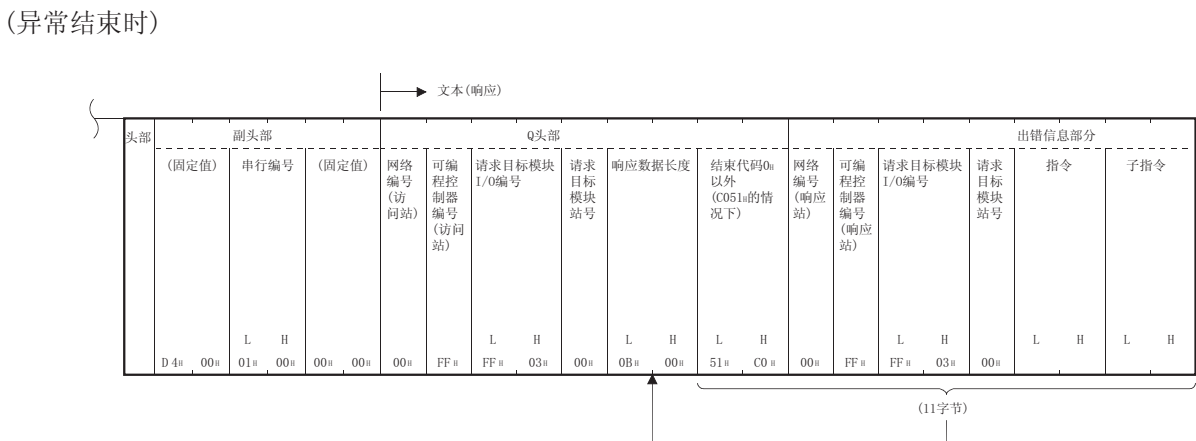
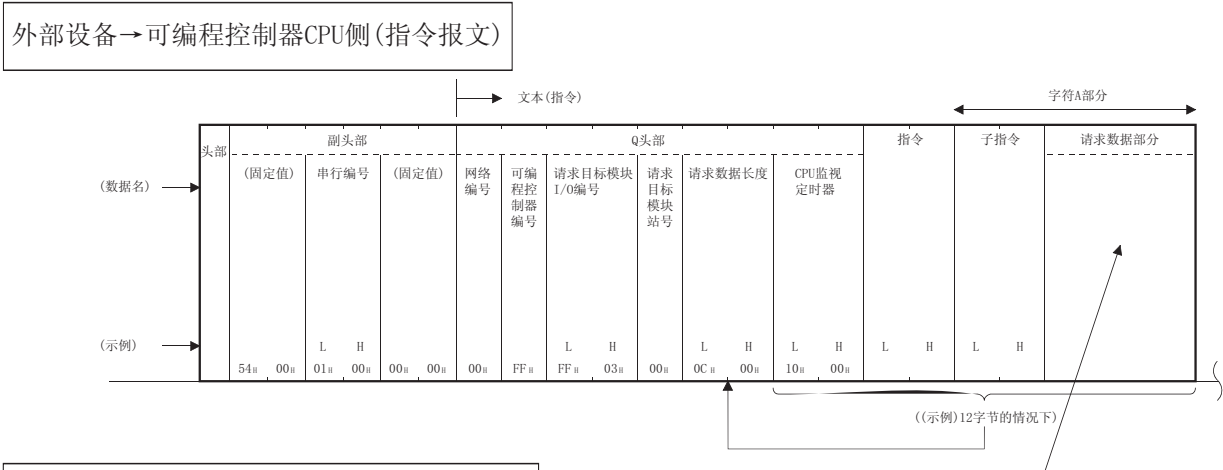
(正常结束时)

数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。



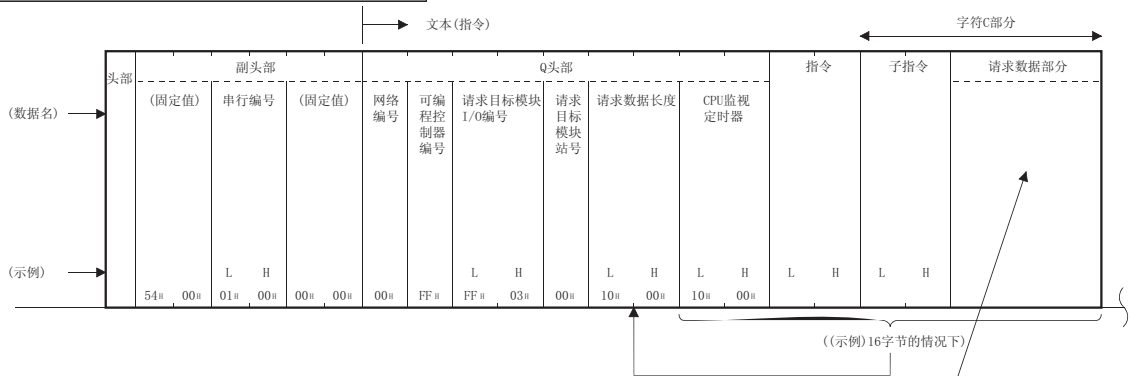
(d) 4E 帧的二进制代码进行通信时

1) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下



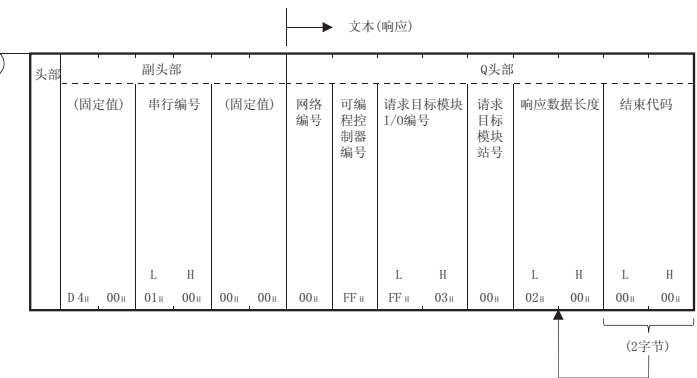
2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



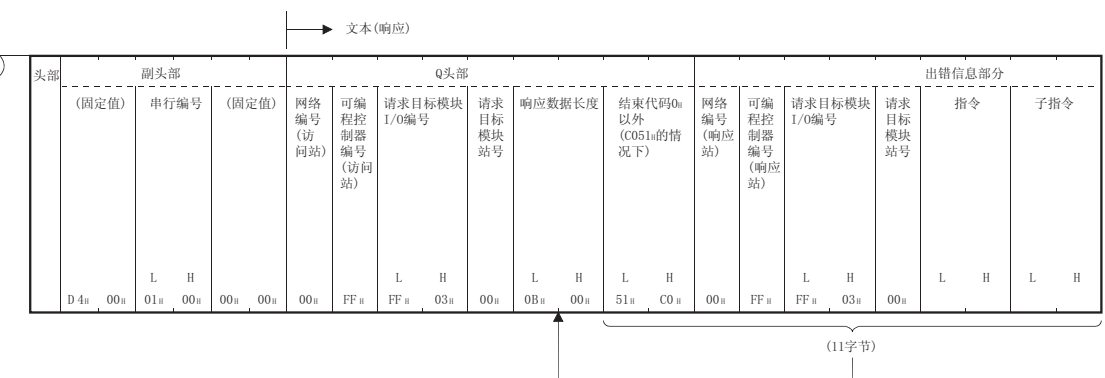
可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)



数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章 3.3.2 项以后所示的指令说明项。

(异常结束时)



3.1.3 QnA 兼容 3E 帧、4E 帧的数据指定项目的内容

以下对通过 E71 用的 QnA 兼容 3E 帧、4E 帧进行数据通信时，各报文的应用数据部分中的通用数据项目、内容及指定方法进行说明。

(1) 网络编号、可编程控制器编号

对其它站可编程控制器进行访问时，通过网络模块等的设置编号将最后经由的网络系统的编号及相应系统上的访问站的可编程控制器编号按以下方式进行指定。

	外部设备的访问站		外部设备指定的编号	
			网络编号	可编程控制器编号
1	E71 安装站(本站)		00 _H	FF _H (*1)
2	CC-Link IE 控制网络、CC-Link IE 现场网络、 MELSECNET/H、MELSECNET/10 上的管理站或主站(将 E71 安装到除管理站/主站以外中时)	除上述 1 以外	01 _H ~ EF _H (1 ~ 239) (*2)	7D _H : 指定管理站/主站 7E _H : 当前管理站/主站
3	MELSECNET/H 上的远程主站(将 E71 安装到远程 I/O 站中时)			
4	CC-Link IE 控制网络、CC-Link IE 现场网络上的站			
5	MELSECNET/H、MELSECNET/10 上的站	除上述 1、2、3、4 以外	FE _H (254) (*5)	01 _H ~ 78 _H (1 ~ 120) (*3) (*4)
6	“其它站访问时的有效模块”设置的网络模块经由站			01 _H ~ 78 _H (1 ~ 120) (*3) (*4)

- *1 可编程控制器编号 FF_H 仅在网络编号为 00_H 时有效。
- *2 对访问站的网络编号进行指定。
- *3 对访问站的站号进行指定。
- *4 对于可编程控制器编号 41_H(65) 以后，E71 安装站的 QCPU(本站)为通用型 QCPU 的情况下可以进行指定。
- *5 经由安装在 MELSECNET/H 远程 I/O 站中的 C24/E71 进行其它站访问时，网络编号 FE_H 将被忽略。
指定了网络编号 FE_H 时，将对 MELSECNET/H 远程 I/O 站的可编程控制器编号中指定的其它站进行访问。

要点
(1) 网络模块的网络 No.、站号以 10 进制数进行设置，而通过 ASCII 代码进行通信时的各编号的指定是以 16 进制数进行指定。
(2) 不能对网络 No. 为 240 ~ 255 的站进行访问。
(3) 经由网络系统中安装的 E71 安装站，指定网络编号 FEH 进行其它站访问时，应将以下参数预先通过 GX Works2 或 GX Developer 设置到 E71 安装站的可编程控制器 CPU 中。 * 在以太网/CC IE/MELSECNET 个数设置的“其它站访问时的有效模块”中进行设置。

(2) 请求目标模块 I/O 编号、请求目标模块站号

访问站的可编程控制器 CPU 为以下 CPU 时进行指定。

- 多 CPU 系统的可编程控制器 CPU。
- 通过 C24 等多点连接上的可编程控制器 CPU。
- 冗余系统的可编程控制器 CPU。

指定方法与使用 QnA 兼容 4C 帧时相同。请参阅 3.1.6 项的[备注]。

* 访问目标可编程控制器 CPU 为除上述以外的 CPU 时，以固定值进行指定。

- 请求目标模块 I/O 编号：03FF_H
- 请求目标模块站号 ：00_H

(3) CPU 监视定时器

该定时器是 E71(从外部设备接收了请求数据的 E71)对可编程控制器 CPU 发出读取/写入请求后，至结果返回的等待时间设置定时器。

(a) 以下述值对设置值进行指定。

- 0000_H(0) ：无限等待*1
0001 ~ FFFF_H(1 ~ 65535) ：等待时间(单位 250ms)

*1 在来自于可编程控制器 CPU 的响应返回之前持续等待。

(b) 为了正常进行数据通信，建议根据通信目标在下表的设置范围内进行设置。

设置范围	通信目标
1 ~ 40 _H (0.25 ~ 10s)	本站
2 ~ 240 _H (0.5 ~ 60s)	经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 的其它站，或通过路由中继的其它站

要点

对 QnACPU、ACPU 进行访问的情况下，由于对 CPU 类型进行识别，仅初次响应报文返回之前需要相当于 CPU 监视定时器等待时间的时间，因此务必在上述(b)中所示的设置范围内进行设置。

(4) 请求数据长度、响应数据长度

对于请求数据长度，指定从文本内的 CPU 监视定时器项目起至请求数据部分的最后为止的字节长度。

作为响应数据长度，返回从文本内的结束代码项目起至响应数据部分/出错信息部分的最后为止的字节长度。

(5) 指令·子指令

对表示外部设备进行可编程控制器 CPU 内数据的读取/写入时的请求内容的指令以及子指令进行指定。

应将 3.2 节以后所示的各功能的指令及子指令根据读取/写入请求内容进行指定。

(6) 请求数据部分、响应数据部分

对于请求数据部分，指定外部设备指定上述指令·子指令通过 MC 协议进行通信时的对应数据(起始软元件、读取/写入范围、写入数据等)。

作为响应数据部分，返回来自于外部设备的请求内容对应的读取数据/写入结果等。

应进行 3.2 节以后所示各功能的指令及子指令对应的数据的指定、数据的读取。

(7) 结束代码

返回指令处理结果。正常结束时，返回图中所示的值。
异常结束时，返回出错代码。
关于出错代码，请参阅以下手册。

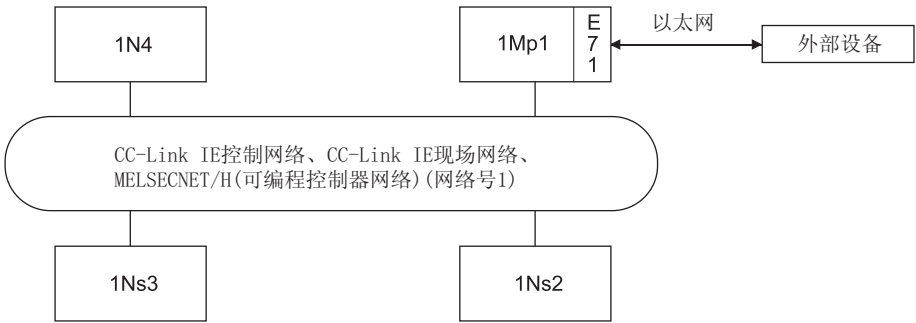
- 使用以太网端口内置的 CPU 时
QCPU 用户手册(硬件设计/维护点检篇)
MELSEC-L CPU 用户手册(硬件设计/维护点检篇)
- 使用 E71 时
Q 系列以太网接口模块用户手册(基本篇)

(8) 出错信息部分

返回进行了出错响应的可编程控制器、发生出错时的指令等。
· 网络编号、可编程控制器编号
进行了出错响应的可编程控制器的网络编号、可编程控制器编号。
· 指令、子指令
发生出错时的指令及子指令。

(9) 网络编号及可编程控制器编号示例

E71 安装站为 CC-Link IE 控制网络、CC-Link IE 现场网络或 MELSECNET/H(可编程控制器网络)的管理站/主站的情况下的网络编号及可编程控制器编号的示例如下所示。



网络编号及可编程控制器编号示例

将 E71 安装到管理站(1Mp1)中的情况下	可通信的可编程控制器及相应项目的指定值(16 进制数)				
	本站	1Mp1	1Ns2	1Ns3	1N4
网络编号	00	-	01		
可编程控制器编号	FF	-	02	03	04

* 关于可访问的站，请参阅 2.6.2 项。

3.1.4 QnA 兼容 3C 帧的控制步骤、报文格式

以下介绍通过 C24 用的 QnA 兼容 3C 帧进行数据通信时的各形式的控制步骤、报文格式。
本项的报文说明图中所示的 □ 部分与本章 3.3.2 项以后所示的报文说明图*部分相对应。
关于 □ 部分的数据内容、数据指定方法，请参阅 3.1.6 项。

(1) 数据通信的基本格式

外部设备通过 MC 协议对可编程控制器进行访问的控制步骤(指令报文·响应报文的
结构及发送接收步骤)有 5 种格式。
根据使用格式，通过 GX Works2 或 GX Developer 将可编程控制器 CPU 的对象接口的
通信协议设置设置为“1”~“5”，可以通过指定格式的各帧进行数据通信。

通信协议设置值	格式	可通信的帧			
		QnA 兼容 3C 帧	QnA 兼容 4C 帧	QnA 兼容 2C 帧	A 兼容 1C 帧
1	格式 1	○	○	○	○
2	格式 2	○	○	○	○
3	格式 3	○	○	○	○
4	格式 4	○	○	○	○
5	格式 5	×	○	×	×

* 格式 1~格式 4 ： 用于通过 ASCII 代码进行的通信
格式 5 ： 用于通过二进制代码进行的通信

将用于通过 ASCII 代码进行的通信的 4 种格式以格式 1 为基准仅区分时的情况如下所示。

- 格式 2 ： 各报文中附加了块编号的格式
- 格式 3 ： 将各报文用 STX、ETX 围住的格式
- 格式 4 ： 各报文中附加了 CR、LF 的格式

关于 QnA 兼容 3C 帧以外的内容，请参阅以下章节。

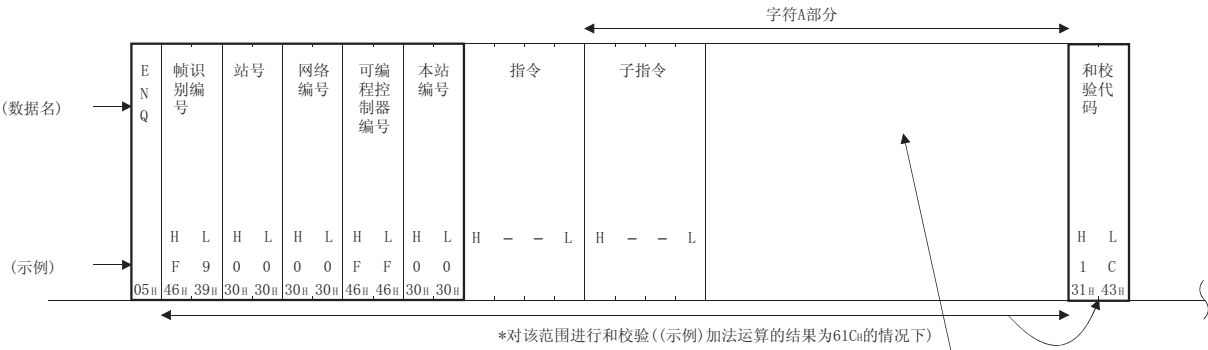
- QnA 兼容 4C 帧 ： 3.1.5 项
- QnA 兼容 2C 帧 ： 4.1 节
- A 兼容 1C 帧 ： 5.1.2 项

要点
本项所示的 QnA 兼容 4C 帧的格式 1~格式 4((2)~(5)图)如下所示。 (a) 仅在和校验被设置为“有”时才会有和校验代码。和校验被设置为“无”时没有和校验代码。 (b) 和校验被设置为“有”的情况下，仅对(2)~(5)图中有*符号部分的字符进行和校验。 (c) (2)~(5)图中的“字符 A 部分”、“字符 B 部分”、“字符 C 部分”的内容根据处理内容而有所不同。详细情况请参阅各指令的说明。此外，4 种格式的各字符部分的内容均相同。 (d) 对于模块接收了指令报文起至发送响应报文为止的时间间隔，可在 0ms~150ms 的范围内进行设置。(通过 GX Works2 或 GX Configurator-SC 进行设置) (e) 在外部设备与可编程控制器 CPU 的系统配置为 m:n 连接的状况下进行数据通信时，不能使用控制步骤格式 3。

(2) 以格式 1 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

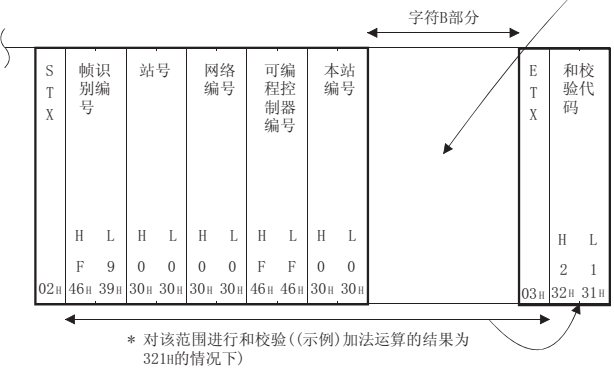
外部设备→可编程控制器CPU侧(指令报文)



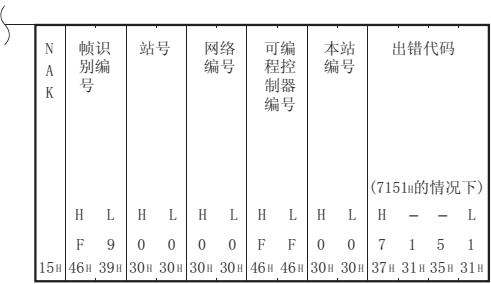
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

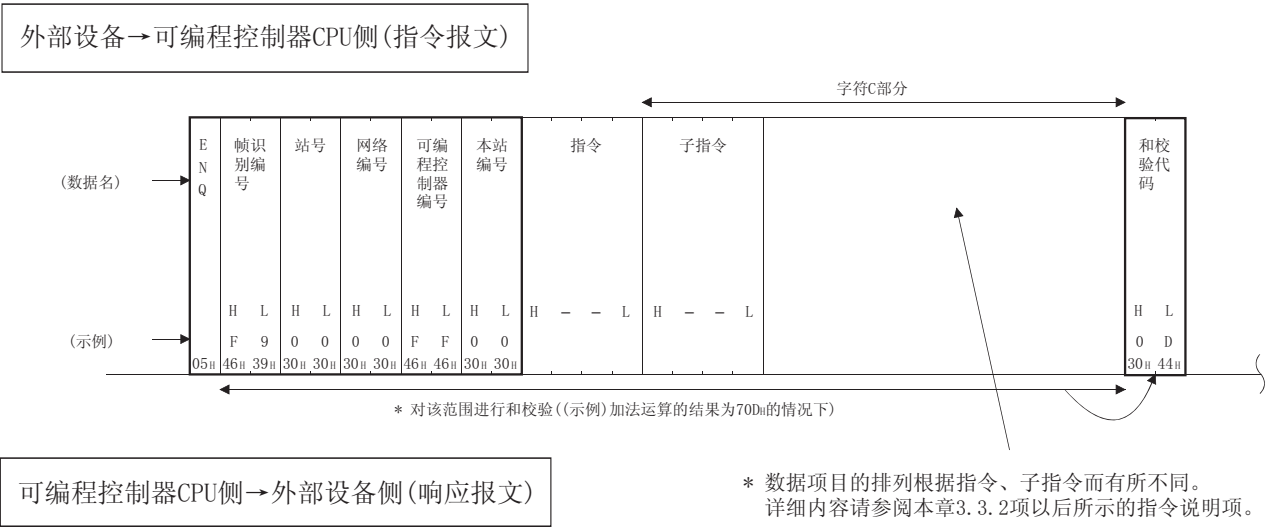
(正常结束时)



(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



(正常结束时)

A C K	帧识别 编号	站号	网络 编号	可编 程控 制器 编号	本站 编号
H L	H L	H L	H L	H L	H L
F 9	0 0	0 0	F F	0 0	0 0
06H	46H 39H	30H 30H	46H 46H	30H 30H	

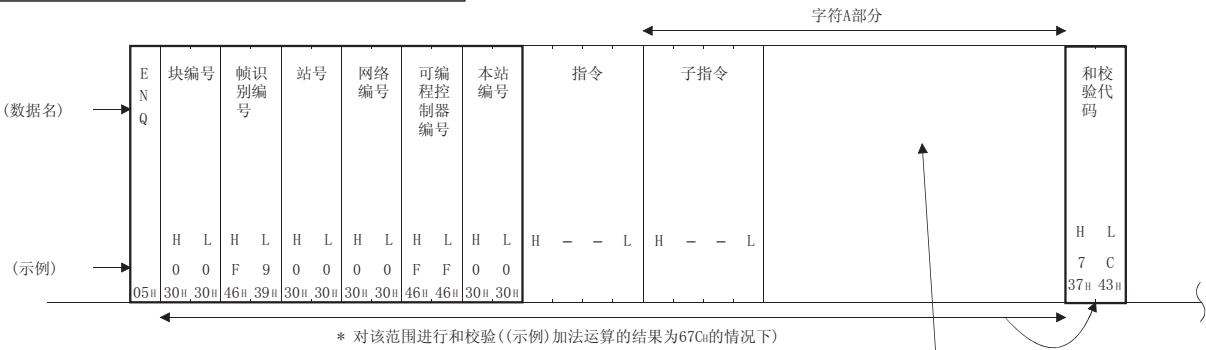
(异常结束时)

N A K	帧识别 编号	站号	网络 编号	可编 程控 制器 编号	本站 编号	出错代码
H L	H L	H L	H L	H L	H L	(7151H的情况下)
F 9	0 0	0 0	F F	0 0	7 1 5 1	
15H	46H 39H	30H 30H	46H 46H	30H 30H	37H 31H 35H 31H	

(3) 以格式 2 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

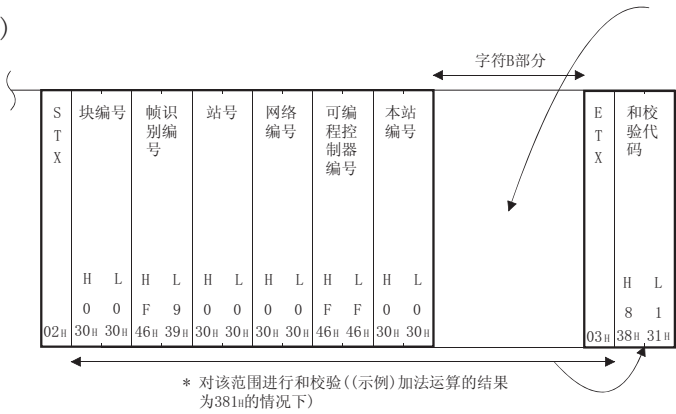
外部设备→可编程控制器CPU侧(指令报文)



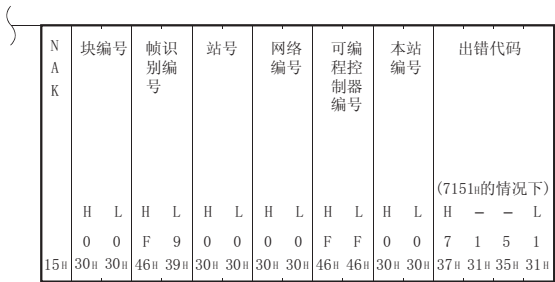
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

(正常结束时)

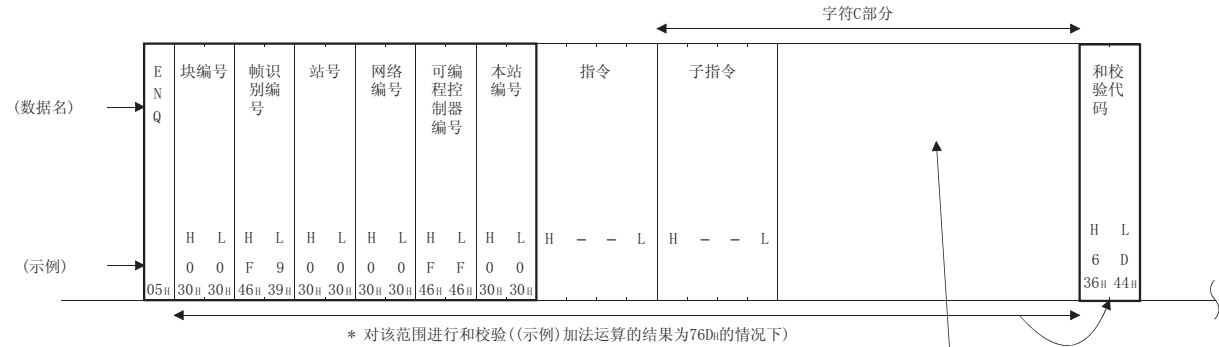


(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

ACK	块编号	帧识别编号	站号	网络编号	可编程控制器编号	本站编号
	H L	H L	H L	H L	H L	H L
	0 0	F 9	0 0	0 0	F F	0 0
	06H 30H 30H	46H 39H	30H 30H	30H 30H	46H 46H	30H 30H

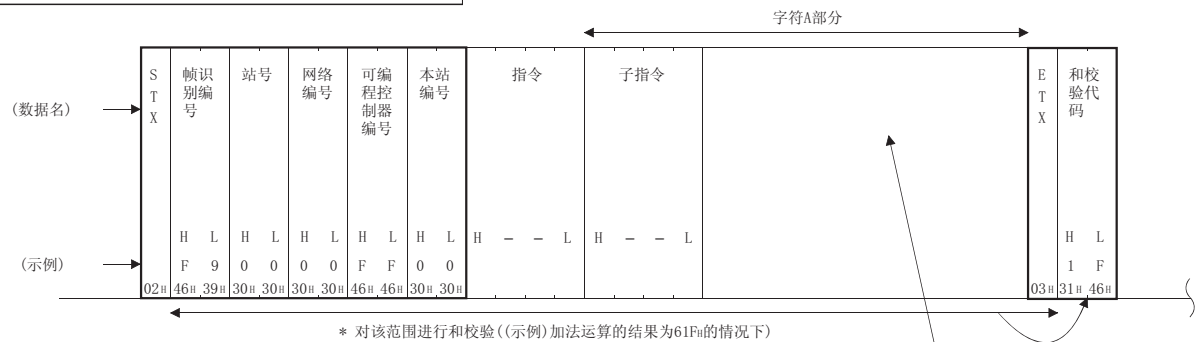
(异常结束时)

NAK	块编号	帧识别编号	站号	网络编号	可编程控制器编号	本站编号	出错代码
	H L	H L	H L	H L	H L	H L	(7151h的情况下)
	0 0	F 9	0 0	0 0	F F	0 0	7 1 5 1
	15H 30H 30H	46H 39H	30H 30H	30H 30H	46H 46H	30H 30H	37H 31H 35H 31H

(4) 以格式 3 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

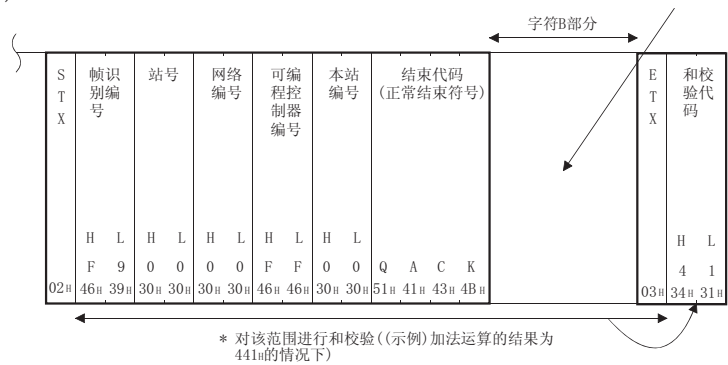
外部设备→可编程控制器CPU侧(指令报文)



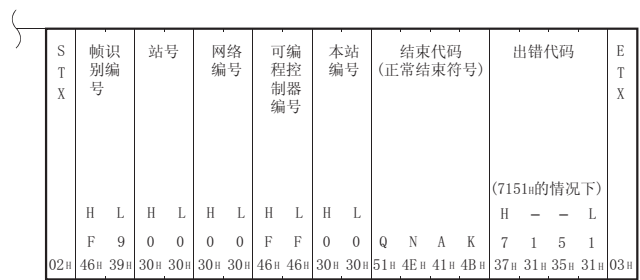
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

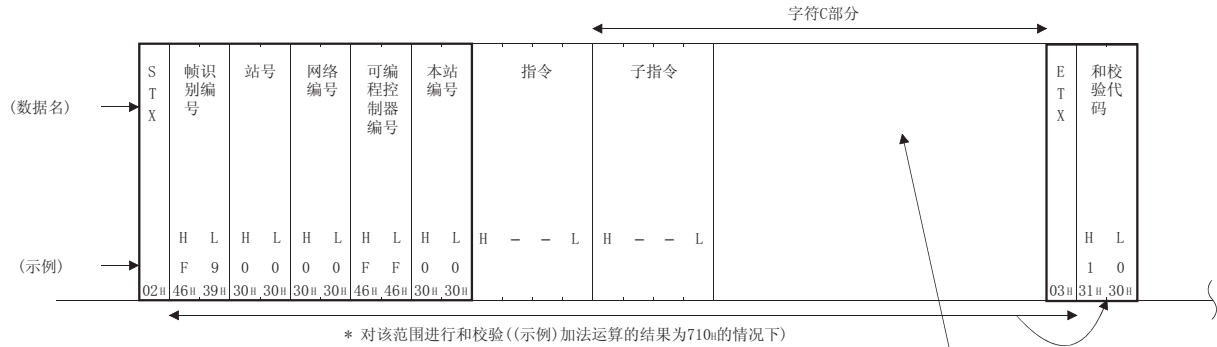


(异常结束时)



(b) 部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

块编号

STX	帧识别编号	站号	网络编号	可编程控制器编号	本站编号	结束代码 (正常结束符号)	ETX
	H L	H L	H L	H L	H L		
	F 9	0 0	0 0	F F	0 0	Q A C K	
	02H 46H 39H	30H 30H	30H 30H	46H 46H	30H 30H	51H 41H 43H 4BH	03H

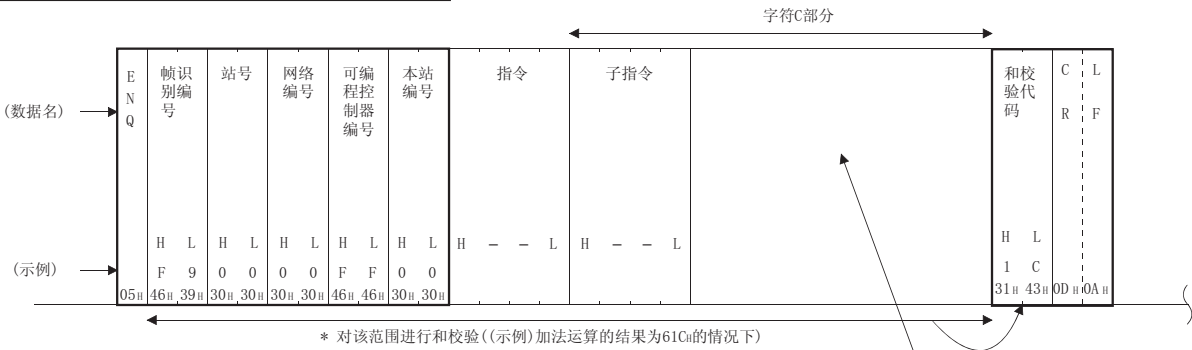
(异常结束时)

STX	帧识别编号	站号	网络编号	可编程控制器编号	本站编号	结束代码 (正常结束符号)	出错代码	ETX
	H L	H L	H L	H L	H L		(7151H的情况下) H - - L	
	F 9	0 0	0 0	F F	0 0	Q N A K	7 1 5 1	
	02H 46H 39H	30H 30H	30H 30H	46H 46H	30H 30H	51H 4EH 41H 4BH	37H 31H 35H 31H	03H

(5) 以格式 4 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

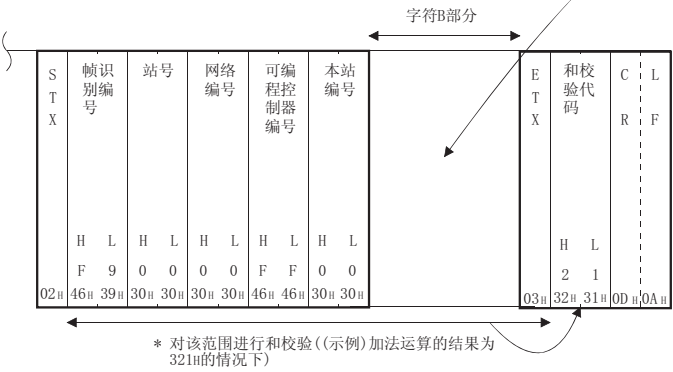
外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

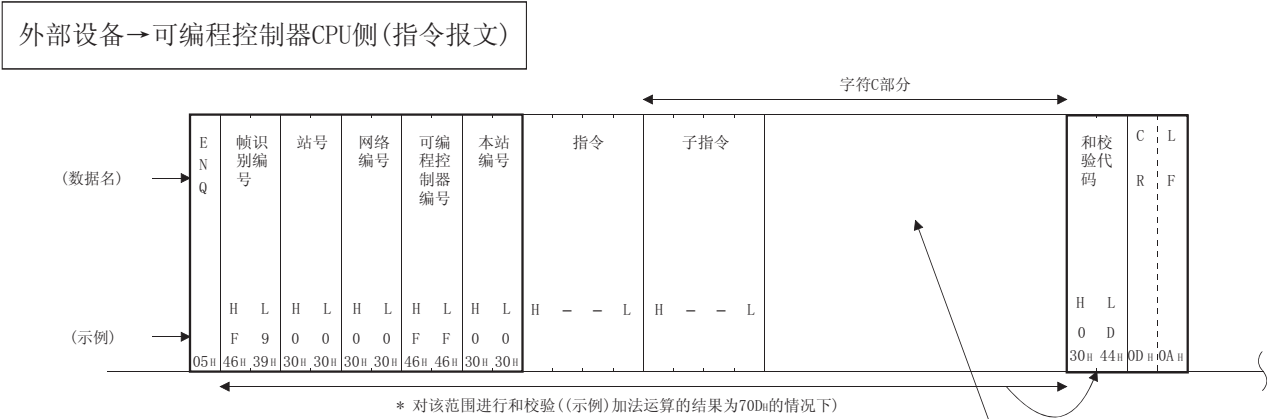
(正常结束时)



(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)

ACK	帧识别编号	站号	网络编号	可编程控制器编号	本站编号	C L	R F
	H L	H L	H L	H L	H L		
	F 9	0 0	0 0	F F	0 0		
	06H 46H 39H	30H 30H	30H 30H	46H 46H	30H 30H	0DH 0AH	

(异常结束时)

NAK	帧识别编号	站号	网络编号	可编程控制器编号	本站编号	出错代码	C L	R F
	H L	H L	H L	H L	H L	(7151H的情况下)		
	F 9	0 0	0 0	F F	0 0	H - - L		
	15H 46H 39H	30H 30H	30H 30H	46H 46H	30H 30H	7 1 5 1	0DH 0AH	

3.1.5 QnA 兼容 4C 帧的控制步骤、报文格式

以下介绍以 C24 用的 QnA 兼容 4C 帧进行数据通信时，各格式的控制步骤、报文格式。
本项的报文说明图中所示的□部分与本章 3.3.2 项以后所示的报文说明图的*部分相对应。

关于□部分的数据内容、数据指定方法，请参阅 3.1.6 项。

关于本项说明中的以下内容，请参阅 3.1.1 项或 3.1.4 项。

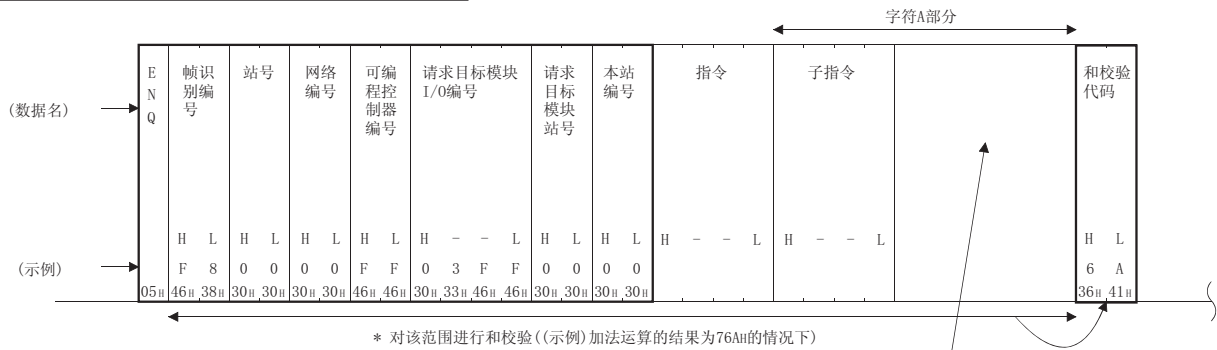
- MC 协议控制步骤的阅读方法 : 参阅 3.1.1 项
- 数据通信的基本格式 : 参阅 3.1.4 项

要点	
	本项所示的 QnA 兼容 4C 帧的格式 1 ~ 格式 5((1) ~ (5)图)如下所示。
(a)	仅在和校验被设置为“有”时才会有和校验代码。和校验被设置为“无”时没有和校验代码。
(b)	和校验被设置为“有”的情况下，仅对(1) ~ (4)图中有*符号部分的字符进行和校验。 通过二进制代码进行通信时((5)图)，仅对有*符号部分的数据(下述(E)中附加的“10H”除外)进行和校验，转换为 ASCII 代码后进行发送。
(c)	(1) ~ (5)的“字符 A 部分”、“字符 B 部分”、“字符 C 部分”的内容根据处理内容而有所不同。详细内容请参阅各指令的说明。此外，格式 1 ~ 格式 4 的各字符部分的内容均相同。
(d)	对于模块接收了指令报文起至发送响应报文为止的时间间隔，可在 0ms ~ 150ms 的范围内进行设置。(通过 GX Works2 或 GX Configurator-SC 进行设置)
(e)	(5)图的*部分的数据中包含有“10H”的数据时，在其前面附加“10H”(DLE 代码)后进行发送。(执行“10H”→“10H”+“10H”后进行发送。) 但是，发送的“数据字节数”中附加的“10H”不包含在内。
(f)	在外部设备与可编程控制器 CPU 的系统配置为 m: n 连接的状况下进行数据通信时，不能使用控制步骤格式 3 及格式 5。

(1) 以格式 1 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

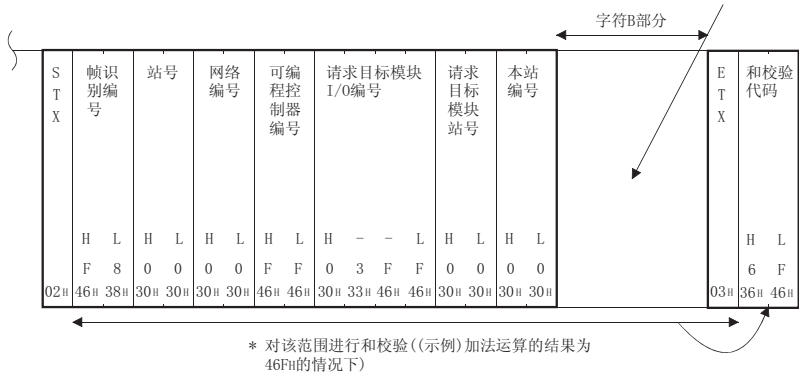
外部设备→可编程控制器CPU侧(指令报文)



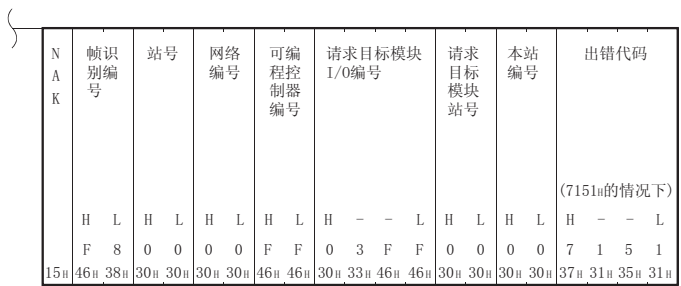
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

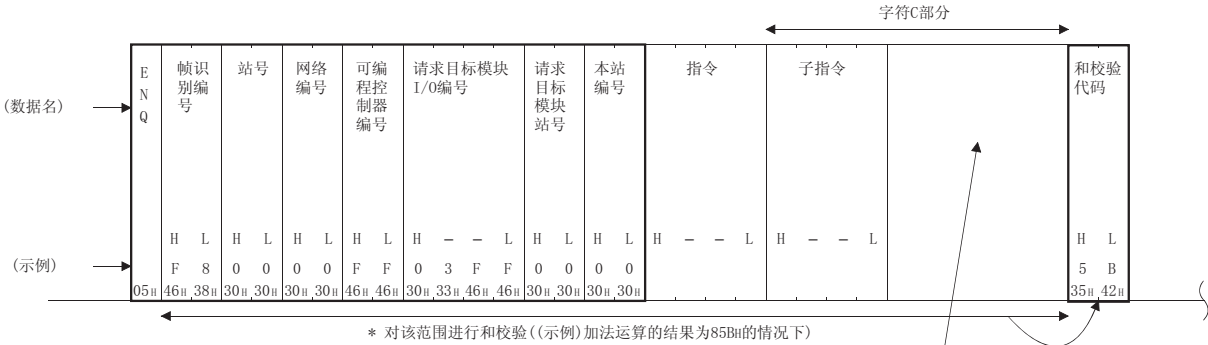


(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

ACK	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号
	H L	H L	H L	H L	H - - L	H L	H L
	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0
06H	46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H

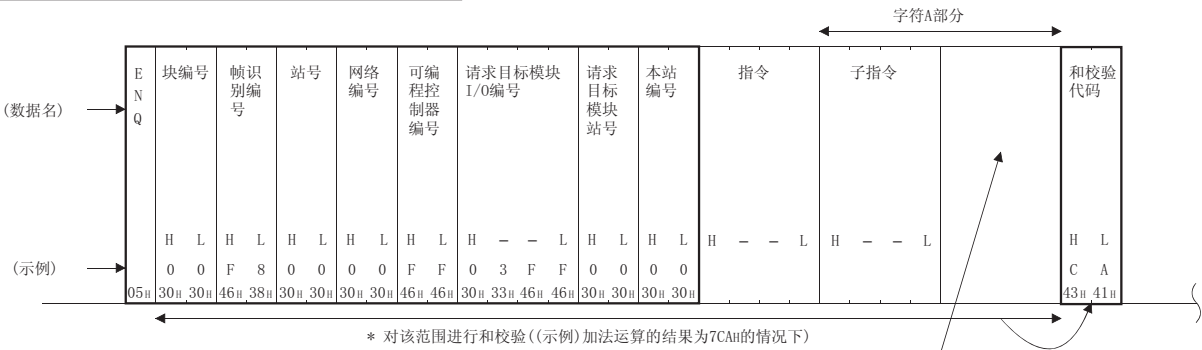
(异常结束时)

NAX	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	出错代码
	H L	H L	H L	H L	H - - L	H L	H L	(7151H的情况下)
	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0	7 1 5 1
15H	46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H	37H 31H 35H 31H

(2) 以格式 2 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

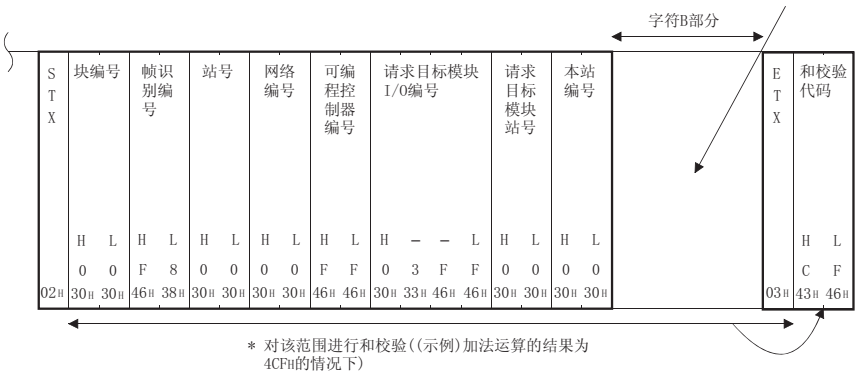
外部设备→可编程控制器CPU侧(指令报文)



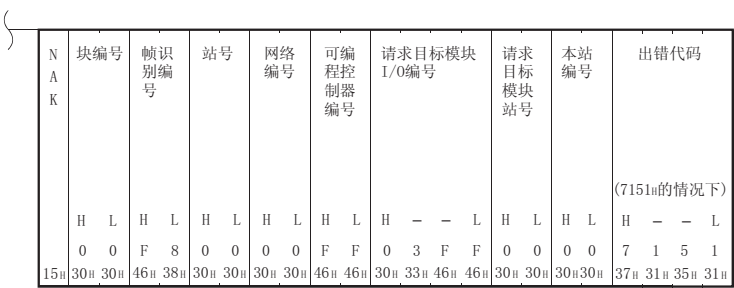
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

(正常结束时)

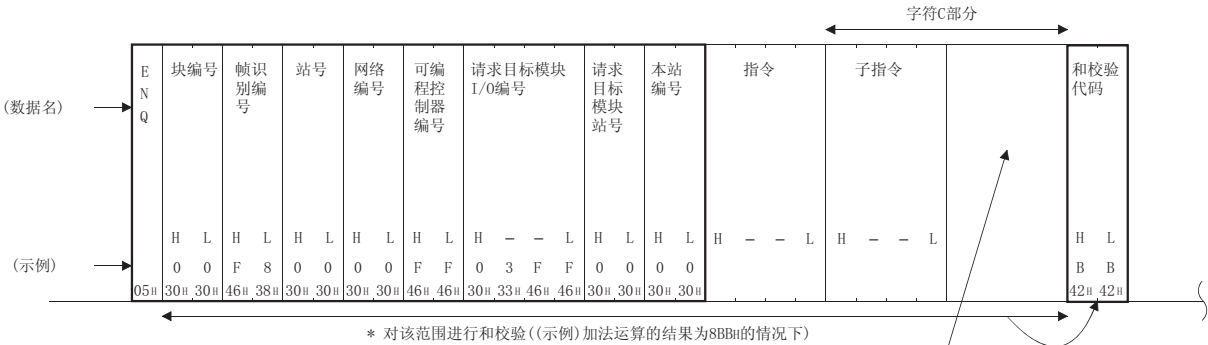


(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

(正常结束时)

ACK	块编号	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号
	H L	H L	H L	H L	H L	H - - L	H L	H L
	00H 30H 30H	46H 38H	30H 30H	30H 30H	46H 46H	03H FFH	30H 30H	30H 30H

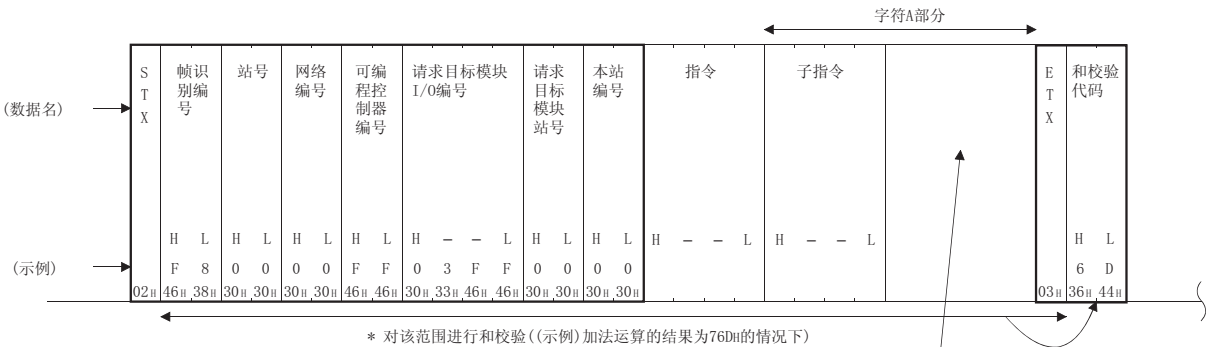
(异常结束时)

NAK	块编号	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	出错代码
	H L	H L	H L	H L	H L	H - - L	H L	H L	(7151H的情况下)
	00H 30H 30H	46H 38H	30H 30H	30H 30H	46H 46H	03H FFH	30H 30H	30H 30H	71H 51H

(3) 以格式 3 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

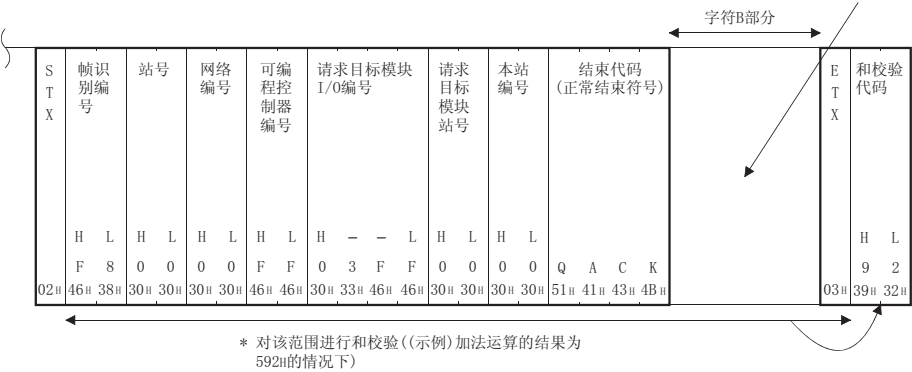
外部设备→可编程控制器CPU侧(指令报文)



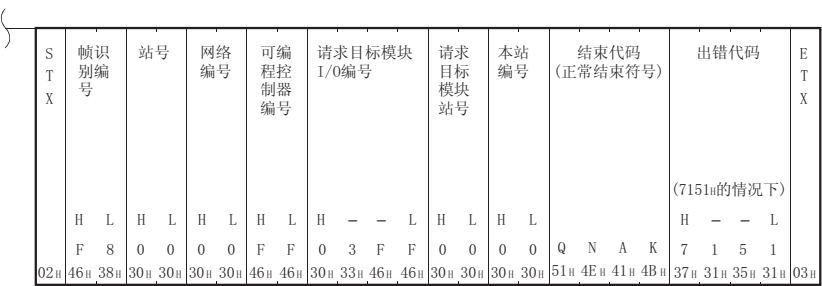
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

(正常结束时)

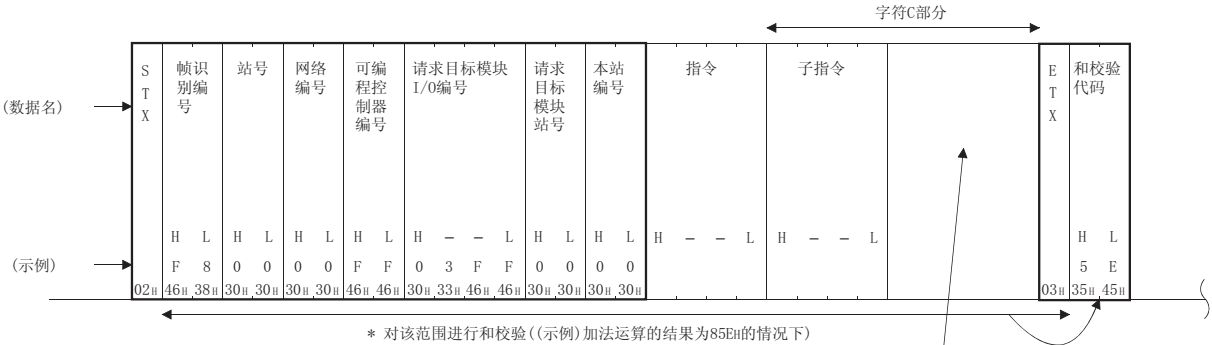


(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章 3.3.2 项以后所示的指令说明项。

(正常结束时)

STX	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	结束代码 (正常结束符号)	ETX
H L	H L	H L	H L	H L	H - - L	H L	H L		
02H	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0	Q A C K	03H
46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H	51H 41H 43H 4BH		

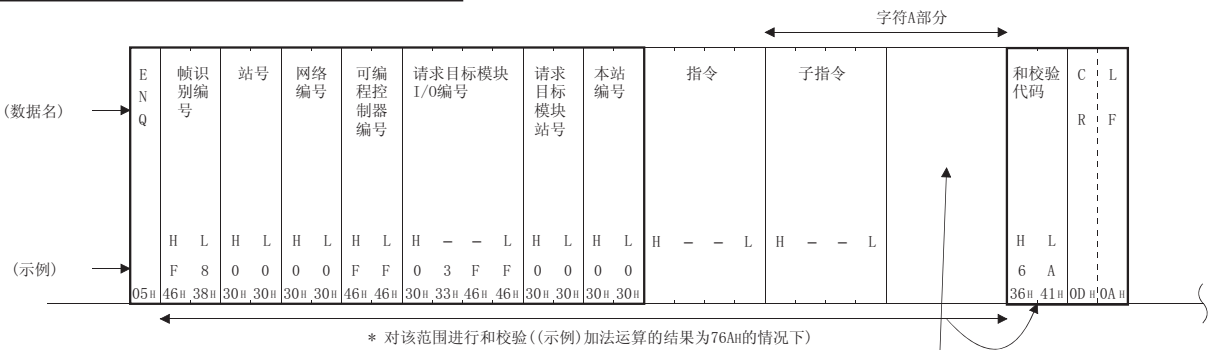
(异常结束时)

STX	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	结束代码 (正常结束符号)	出错代码	ETX
H L	H L	H L	H L	H L	H - - L	H L	H L		(7151H 的情况下)	
02H	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0	Q N A K	7 1 5 1	03H
46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H	51H 4EH 41H 4BH	37H 31H 35H 31H		

(4) 以格式 4 进行通信时(用于通过 ASCII 代码进行的通信)

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

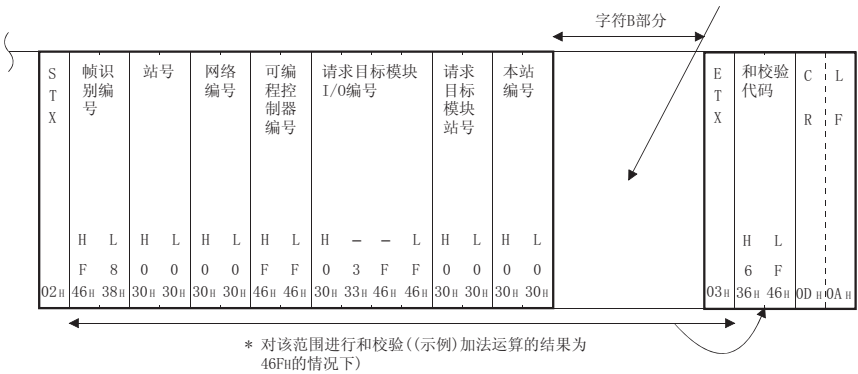
外部设备→可编程控制器CPU侧(指令报文)



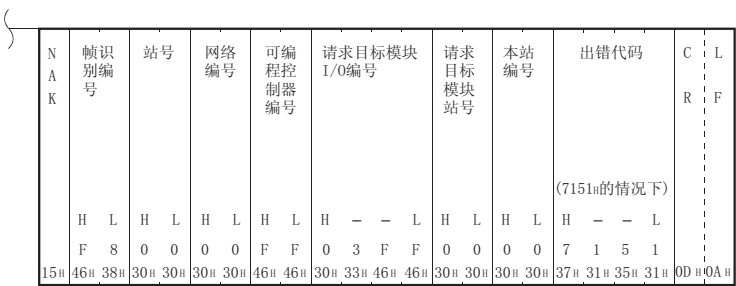
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3.3.2项以后所示的指令说明项。

(正常结束时)

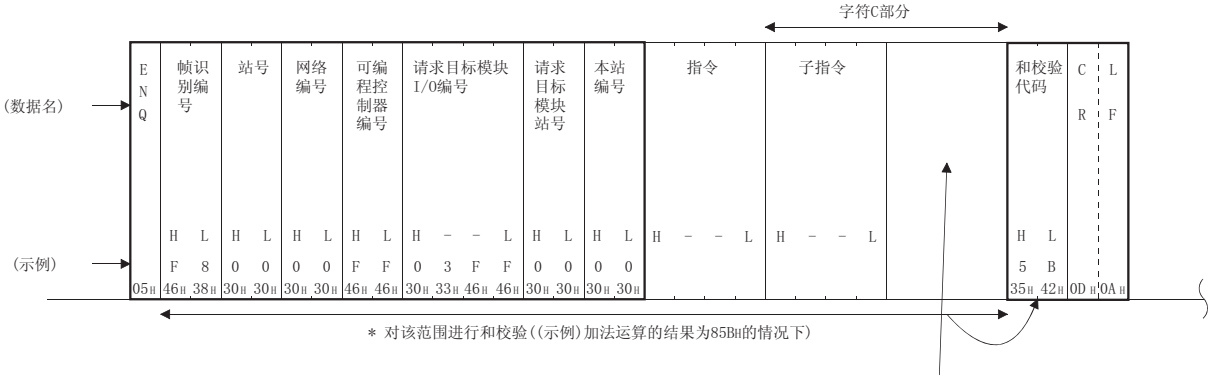


(异常结束时)



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

外部设备→可编程控制器CPU侧(指令报文)



可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

(正常结束时)

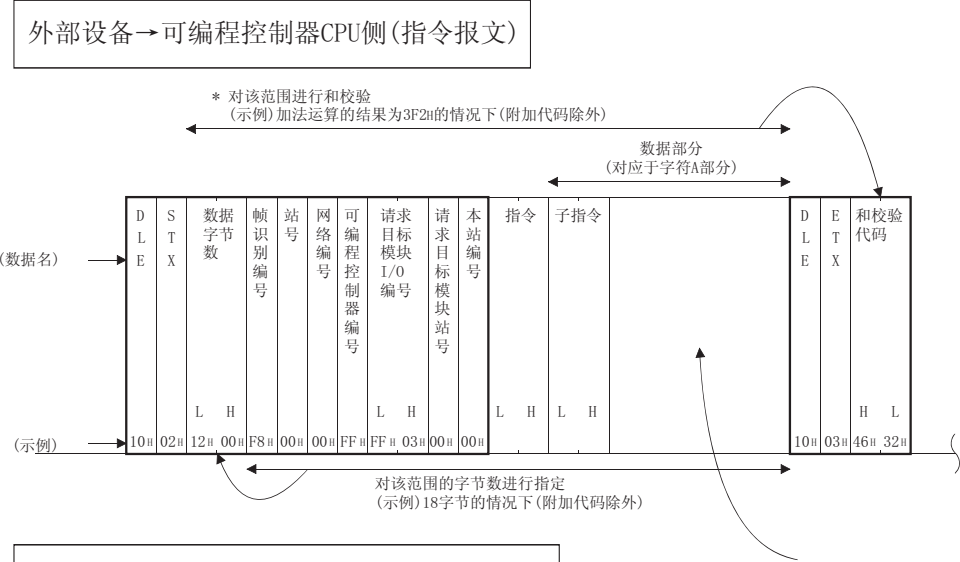
ACK	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	C	L
	H L	H L	H L	H L	H - - L	H L	H L	R	F
	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0		
	06H 46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H	0DH 0AH	

(异常结束时)

NAK	帧识别编号	站号	网络编号	可编程控制器编号	请求目标模块 I/O 编号	请求目标模块站号	本站编号	出错代码	C	L
	H L	H L	H L	H L	H - - L	H L	H L	(7151H的情况下)	R	F
	F 8	0 0	0 0	F F	0 3 F F	0 0	0 0	H - - L		
	15H 46H 38H	30H 30H	30H 30H	46H 46H	30H 33H 46H 46H	30H 30H	30H 30H	71H 51H	0DH 0AH	

(5) 以格式 5 进行通信时(用于通过二进制代码进行的通信)

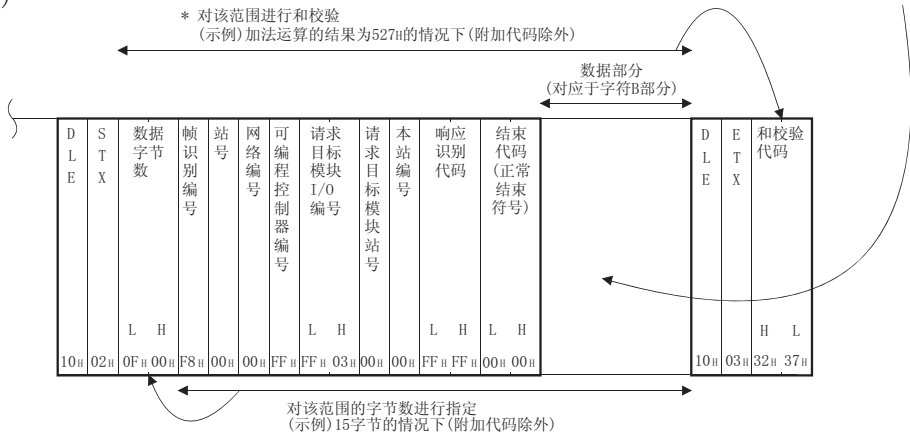
(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下



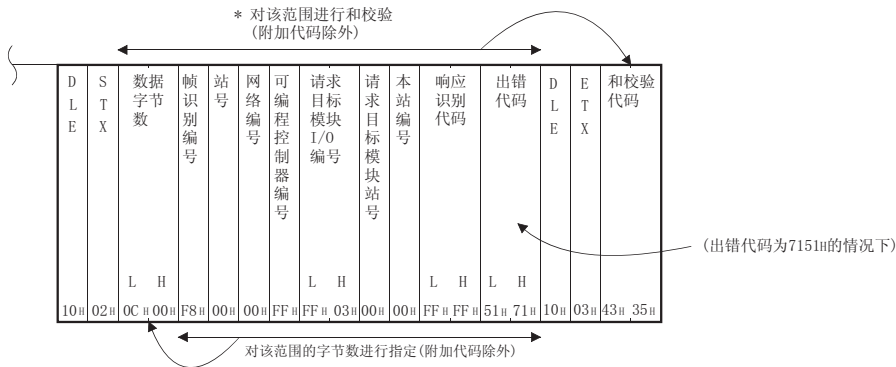
可编程控制器CPU侧→外部设备侧(响应报文)

* 数据项目的排列根据指令、子指令而有所不同。
详细内容请参阅本章3. 3. 2项以后所示的指令说明项。

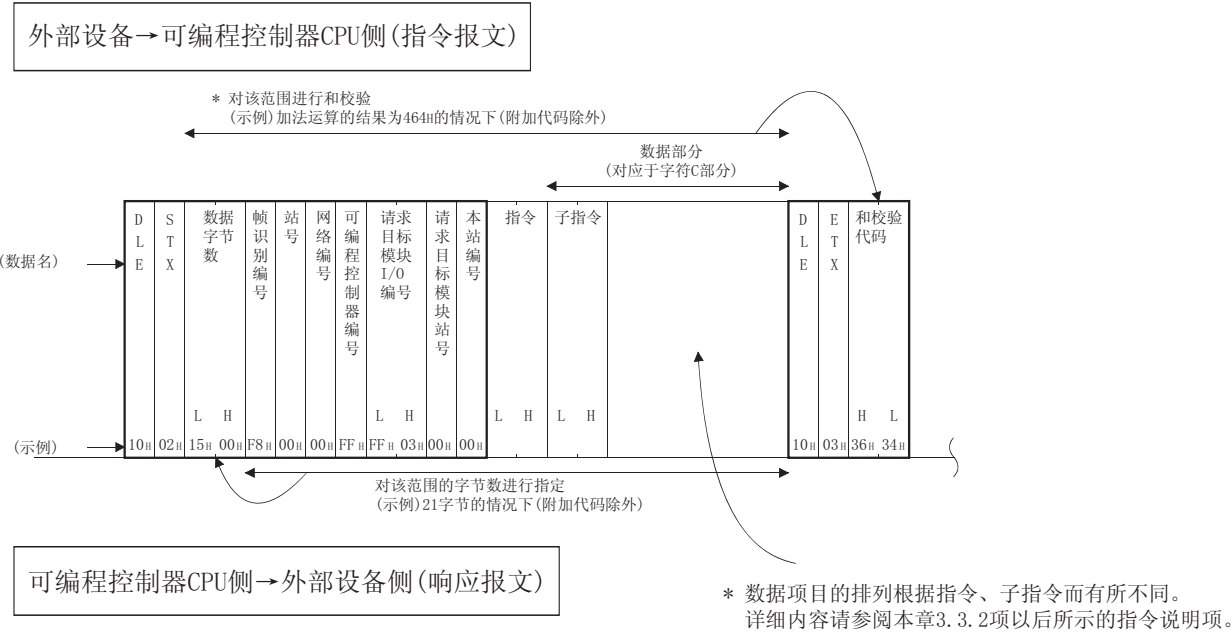
(正常结束时)



(异常结束时)

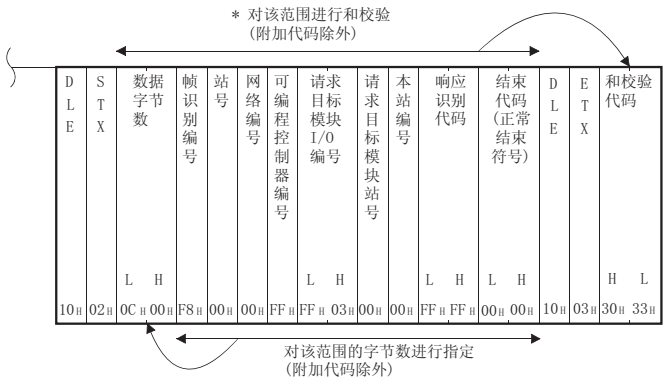


(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

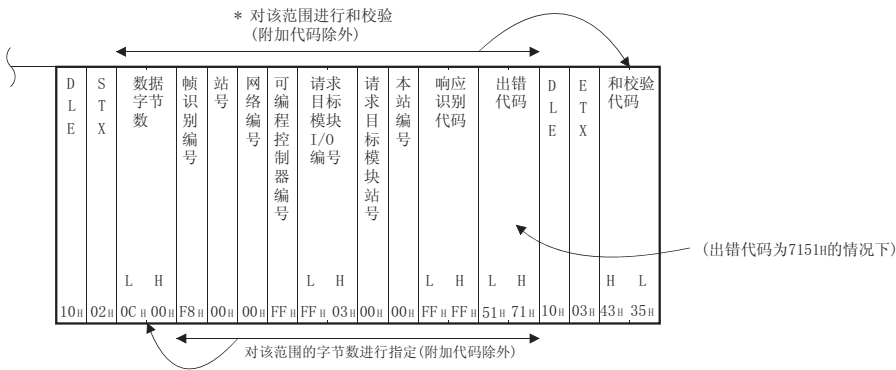


可编程控制器CPU侧→外部设备侧(响应报文)

(正常结束时)



(异常结束时)



3.1.6 QnA 兼容 3C/4C 帧的数据指定项目的内容

以下对以 C24 用的 QnA 兼容 3C 帧/QnA 兼容 4C 帧进行数据通信时，各报文中的通用数据项目、内容及指定方法进行说明。

(1) 控制代码

各控制步骤中，作为 C24 的传送控制用的具有特别含义的数据(报文的起始数据等)的代码及内容如下表所示。

对于 ASCII 代码的各帧中使用的控制代码，在表中的 ASCII 代码栏中以 ○ 符号表示。

对于二进制代码的 QnA 兼容 4C 帧中使用的控制代码，在表中的二进制代码栏中以 ○ 符号表示。

报文上的控制代码的使用方法相同。

符号名	代码 (16 进制)	内容	ASCII 代码	二进 制代 码	符号名	代码 (16 进制)	内容	ASCII 代码	二进 制代 码
NUL	00 _H	Null	○	○	CL	0C _H	Clear	○	
STX	02 _H	Start of Text	○	○	CR	0D _H	Carriage Return	○	
ETX	03 _H	End of Text	○		DLE	10 _H	Data Link Escape		○
EOT	04 _H	End of Transmission	○		NAK	15 _H	Negative Acknowledge	○	
ENQ	05 _H	Enquiry	○		QnA 兼 容帧识 别编号	F6 _H	(系统用)		○
ACK	06 _H	Acknowledge	○			F8 _H	(QnA 兼容 4C 帧 识别代码)	○	○
LF	0A _H	Line Feed	○			F9 _H	(QnA 兼容 3C 帧 识别代码)	○	
-	-	-	○			FA _H	(系统用)	○	○

要点
通过 QnA 兼容 4C 帧的格式 5 以二进制代码进行数据通信的情况下，报文中的“数据字节数”~“数据部分”中包含“10 _H ”的用户数据时，在其前面附加“10 _H ”的 DLE 代码(在本文中表示为附加代码。)后进行发送接收。 (通过“10 _H ”→“10 _H ”+“10 _H ”进行发送接收。) 对响应报文附加 C24。 关于报文结构示例，请参阅 3.1.7 项(2)。

- (a) Null 代码(00_H)在所有的报文中均被忽略。因此报文中即使有 Null 代码也将被作为虚无处理。
- (b) EOT、CL 是用于对通过 MC 协议的 ASCII 代码进行数据通信的传送顺控程序进行初始化，将 C24 置为对来自于外部设备的指令进行接收等待状态的代码。在外部设备侧执行以下操作时，应根据使用的格式将 EOT/CL 发送至 C24。
 - 1) 取消通过之前发送的指令进行的读取/写入请求。(发送了写入请求的情况下，数据已被写入到可编程控制器 CPU 中时，不能取消写入请求。)
 - 2) 发送指令之前，将 C24 预先置为指令接收等待状态。
 - 3) 无法正常进行数据通信时，将 C24 置为启动时的状态。

发送 EOT、CL 时的报文结构如下所示。

格式 1~3 的情况下	格式 4 的情况下	格式 5 的情况下
E O T 或者 C L 外部设备侧 可编程控制器侧	E O T C R F 或者 C C L L R F 外部设备侧 可编程控制器侧	通过传送顺控程序初始化(指令:1615)，对传送顺控程序进行初始化。
发送 EOT、CL 时，仅发送上图所示的数据。不要发送站号及可编程控制器编号等。		参阅 3.12.2 项

C24 接收了 EOT/CL 时，其情况如下所示。

- 通过来自于外部设备的请求，中止对可编程控制器 CPU 进行的读取/写入处理。
在这种情况下，不对之前接收的指令发送响应报文。
- 将接收了 EOT/CL 的接口侧的 MC 协议的传送顺控程序进行初始化后，置为对来自于外部设备的指令进行接收等待的状态。
- 对 EOT/CL 的接收不发送响应报文。(不向外部设备发送任何信息。)
- 接通请求功能(从可编程控制器 CPU 至外部设备的数据发送功能，参阅 3.11 项)的执行过程中接收了 EOT/CL 时，停止至外部设备的接通请求功能的数据发送。

(2) 块编号(QnA 兼容 3C/4C 帧的格式 2 用)

块编号是在外部设备侧对相应报文附加含义的任意编号，用于数据整理编号等。
块编号的范围为 00H ~ FFH，转换为 ASCII 代码 2 位(16 进制数)使用，从高位开始进行发送。

C24 仅对块编号是否指定在正确的范围内进行检查。
不对通过指令报文发送的块编号是否按编号顺序发送进行检查。

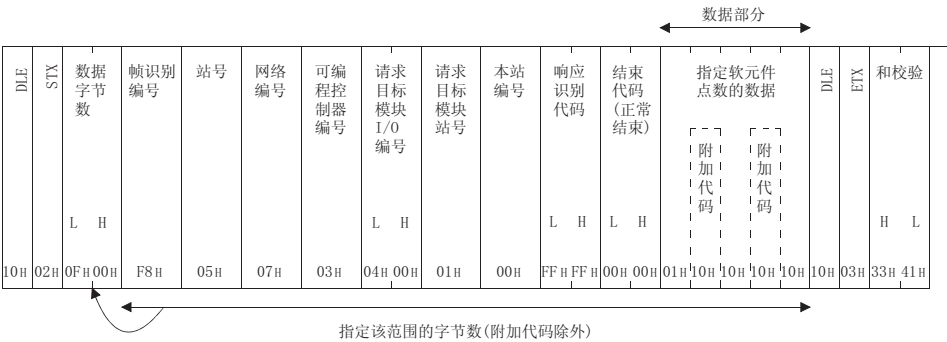
(3) 数据字节数(QnA 兼容 4C 帧的格式 5 用)

数据字节数是指，将帧识别编号～数据部分或帧识别编号～结束代码(出错代码)的除附加代码以外的数据的总字节数传送至外部设备的项目。(由于可编程控制器 CPU 不对从外部设备接收的指令报文中的数据字节数进行检查，因此也可以以虚拟数据 00H、00H 进行指定。)

数据字节数使用 2 字节的数值，按 Low 字节(L：位 0～7)、High 字节(H：位 8～15)的顺序进行发送。

(示例)

从可编程控制器 CPU 中进行了数据读取时的响应报文



(4) 帧识别编号

帧识别编号是指，用于对发送接收报文是 QnA 兼容 3C 帧还是 QnA 兼容 4C 帧进行识别的编号。

数据通信时指定的帧识别编号如下所示。

	帧	帧识别编号	备注
通过 ASCII 代码进行的通信	QnA 兼容 3C 帧	“ F9 ”	按 “ F ”、“ 9 ” 的顺序进行 2 字符发送。
	QnA 兼容 4C 帧	“ F8 ”	按 “ F ”、“ 8 ” 的顺序进行 2 字符发送。
通过二进制代码进行的通信	QnA 兼容 4C 帧	F8H	对 F8H 进行 1 字节发送。

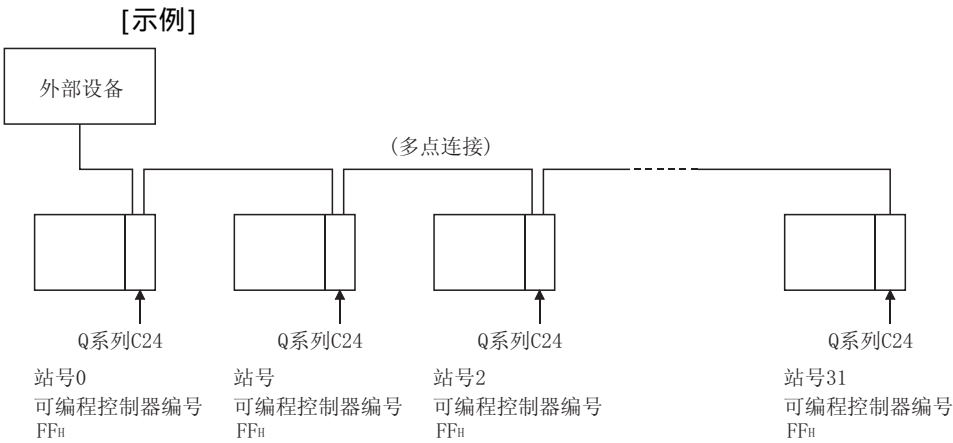
(5) 站号～本站编号

站号～本站编号是用于指定外部设备访问的可编程控制器站的数据，通过以下 4 组数据对至访问目标的路径进行指定。

关于进行其它站访问时的网络编号～请求目标模块站号的指定示例，请参阅 2.6.3 项。

· 站号

通过 GX Works2 或 GX Developer 中设置的 C24 的站号指定外部设备访问哪个 C24，或对 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的其它站可编程控制器进行访问时，经由哪个 C24。



备注

即使站号的设置未如上图那样按编号顺序进行设置也没有关系。先从站号 31 开始也能正常通信。

- 网络编号、可编程控制器编号
对其它站可编程控制器进行访问时，将最后经由的网络系统编号及相应系统上的可编程控制器编号(站号)通过网络模块的设置编号进行指定。
 - 请求目标模块 I/O 编号、请求目标模块站号
对其它站可编程控制器进行访问时，经由网络系统、再经由多点连接的可编程控制器 CPU 的情况下，对该多点连接的连接源可编程控制器 CPU 及访问目标可编程控制器 CPU 进行指定。
对于请求目标模块 I/O 编号，将多点连接源 C24 通过输入输出信号的起始编号进行指定。
对于请求目标模块站号，将访问目标 C24 通过 C24 的设置站号进行指定。
 - 本站编号
将外部设备的站号指定为“0”(默认值)。
- a) 站号
- 1) 通过 ASCII 代码进行数据通信时
在 00H ~ 1FH(0 ~ 31)的范围内，转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用 1 字节数值“00H” ~ “1FH”(0 ~ 31)进行发送。

要点

- (1) 使用 3.10 节所示的全局功能的情况下，协议上的站号指定为“FF”/FFH。
如果指定了 0 ~ 31(00H ~ 1FH)之一，仅指定的站的 X1A/X1B 将 ON，其它站不变为 ON。
- (2) 外部设备与可编程控制器 CPU 的系统配置为 m:n 的情况下，以 QnA 兼容 3C 帧、QnA 兼容 4C 帧进行数据通信时，外部设备侧的站号应在 00H ~ 1FH(0 ~ 31)的范围内且不与 C24 侧的重复。
关于外部设备侧的站号，请参阅用户手册(应用篇)。

(b) 网络编号

- 1) 通过 ASCII 代码进行数据通信时
在下述范围内，转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 数据通信时指定的网络编号如(c)中所示。

要点
(1) 不能对网络 No. 为 240 ~ 255 的站进行访问。 (2) 经由安装在网络系统中的可编程控制器 CPU 安装站，指定网络编号 FEH 进行其它站访问时，应预先将以下参数通过 GX Works2 或 GX Developer 设置到可编程控制器 CPU 安装站的可编程控制器 CPU 中。 “其它站访问时的有效模块”设置 在以太网/CC IE/MELSECNET 个数设置中进行设置，进行其它站访问时对所经由的模块进行设置 (3) 对于网络编号应以表中所示的编号进行指定。 指定了除表中所示编号以外的编号时，有可能不返回响应。

(c) 可编程控制器编号

- 1) 通过 ASCII 代码进行数据通信时
在下述范围内，转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 数据通信时指定的可编程控制器编号如下所示。

	外部设备的访问站	外部设备指定的编号	
		网络编号	可编程控制器编号
1	外部设备连接站(本站)	00 _H	FF _H (*1)
2	来自于外部设备连接站的多点连接站	00 _H	FF _H (*1)
3	CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的管理站或主站(将 C24 安装到除管理站/主站以外时)	01 _H ~ EF _H (1 ~ 239) (访问站的网络编号)	7D _H : 指定管理站/主站 7E _H : 当前管理站/主站
4	MELSECNET/H 上的远程主站(将 C24 安装到远程 I/O 站中时)		
5	CC-Link IE 控制网络、CC-Link IE 现场网络上的站		01 _H ~ 78 _H (1 ~ 120) (*3) (访问站的站号) (上述 1、3、4 除外)
6	MELSECNET/H、MELSECNET/10 上的站		01 _H ~ 40 _H (1 ~ 64) (访问站的站号) (上述 1、3、4、5 除外)
7	“其它站访问时的有效模块”设置的网络模块经由站	FE _H (254) (*2) 01 _H ~ EF _H (1 ~ 239) (经由的最终网络编号)	01 _H ~ 78 _H (1 ~ 120) (*3)
8	经由 CC-Link IE 控制网络、CC-Link IE 现场网络连接站的多点连接站		01 _H ~ 78 _H (1 ~ 120) (*3) (经由的最终站的站号)
9	经由 MELSECNET/H、MELSECNET/10 连接站的多点连接站		01 _H ~ 40 _H (1 ~ 64) (经由的最终站的站号)

- *1 对于可编程控制器编号 FF_H，只有(b)网络编号为 00_H 时才可以指定。
- *2 经由 MELSECNET/H 远程 I/O 站中安装的 C24/E71 进行其它站访问时，网络编号 FE_H 将被忽略。
指定了网络编号 FE_H 时，将对 MELSECNET/H 远程 I/O 站的可编程控制器编号中指定的其它站进行访问。
- *3 对于可编程控制器编号 41_H(65)以后的编号，只有 C24 安装站的 QCPU(本站)为通用型 QCPU 的情况下才可以指定。

要点

使用接通请求功能的情况下，可编程控制器 CPU 作为协议上的网络编号发送 00_H，
作为可编程控制器编号发送“FE”/FE_H。

(d) 请求目标模块 I/O 编号

1) 通过 ASCII 代码进行数据通信时

使用将对象可编程控制器 CPU 的输入输出信号以 4 位表示时的高 3 位，
或将 03FF_H 转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。

(示例)

可编程控制器 CPU 的输入输出信号为 0080_H ~ 009F_H 的情况下

请求目标模块 I/O 编号将变为“0008”，从“0”开始按顺序进行发送。

2) 通过二进制代码进行数据通信时

使用将对象可编程控制器 CPU 的输入输出信号以 4 位表示时的高 3 位的
数值，或 03FF_H 的 2 字节的数值，按 Low 字节(L: 位 0 ~ 7)、High 字节
(H: 位 8 ~ 15)的顺序进行发送。

(示例)

可编程控制器 CPU 的输入输出信号为 0080_H ~ 009F_H 的情况下

请求目标模块 I/O 编号将变为 0008_H，按 08_H、00_H 的顺序进行发送。

3) 数据通信时指定的请求目标模块 I/O 编号如下所示。

	外部设备的访问站	外部设备指定的请求目标模块 I/O 编号
1	控制系统 CPU	03D0 _H
2	待机系统 CPU	03D1 _H
3	A 系统 CPU	03D2 _H
4	B 系统 CPU	03D3 _H
5	多 CPU 1 号机	03E0 _H
6	多 CPU 2 号机	03E1 _H
7	多 CPU 3 号机	03E2 _H
8	多 CPU 4 号机	03E3 _H
9	本站 CPU/管理 CPU/自系 CPU	03FF _H
10	多点连接上的站的 C24 的管理 CPU(对最后经由的 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 连接站的可编程控制器进行访问时指定上述 1 ~ 9)	0000 _H ~ 01FF _H

要点

经由冗余系统的扩展基板上的 C24/E71 的情况下，根据不同指令可指定的访问站有所不同。

详细内容请参阅 QnPRHCPU 用户手册(冗余系统篇)。

(e) 请求目标模块站号

1) 通过 ASCII 代码进行数据通信时

在下述范围内，转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。

2) 通过二进制代码进行数据通信时

使用下述 1 字节的数值进行发送。

3) 数据通信时指定的请求目标模块站号如下所示。

	外部设备的访问站	外部设备指定的请求目标模块站号
1	下述以外的站	00H (0)
2	多点连接上的站 (对最后经由的 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 连接站的 Q/QnACPU 进行访问时使用上述 1 的站号)	00H ~ 1FH (0 ~ 31)

(f) 本站编号

- 1) 通过 ASCII 代码进行数据通信时，使用“00”进行发送。
- 2) 通过二进制代码进行数据通信时，使用 1 字节数值“00H”(0)进行发送。

要点
外部设备与可编程控制器 CPU 的系统配置为 m:n 的情况下，以 QnA 兼容 3C 帧，QnA 兼容 4C 帧进行数据通信时，外部设备侧的站号应在 00H ~ 1FH(0 ~ 31)的范围内且不与 C24 侧的重复。 关于外部设备侧的站号，请参阅用户手册(应用篇)。

(6) 指令

是从外部设备对访问目标可编程控制器进行读取、写入等，对访问内容进行指定的命令。

- 1) 通过 ASCII 代码进行数据通信时
将指令转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
(示例)
位单位的批量读取指令的情况下
指令“0401”从“0”开始按顺序进行发送。
- 2) 通过二进制代码进行数据通信时
将指令作为 2 字节数值(16 进制数)使用，按 Low 字节(L: 位 0 ~ 7)、High 字节(H: 位 8 ~ 15)的顺序进行发送。
(示例)
位单位的批量读取指令的情况下
指令 0401H 按 01H、04H 的顺序进行发送。

(7) 字符□部分(数据部分)

各字符部分的用途如下所示。

- 字符 A 部分 : 是用于使可编程控制器 CPU 执行指令中指定的读取请求的数据。
- 字符 B 部分 : 是对指令中指定的请求返回信息至外部设备的数据。
- 字符 C 部分 : 是用于使可编程控制器 CPU 执行指令中指定的写入请求的数据。

对于字符部分(数据部分)的内容，根据从外部设备发送的指令而有所不同。

- 1) 通过 ASCII 代码进行数据通信时，将字符部分转换为 ASCII 代码后进行发送。
- 2) 通过二进制代码进行数据通信时，将字符部分以二进制代码进行发送。
- 3) 关于字符部分的传送数据的思路，请参阅 3.1.7 项的说明。

(8) 和校验代码

和校验代码是指，对将报文中的和校验对象范围(参阅 3.1.4 项 ~ 3.1.5 项)数据作为二进制数据进行加法运算的结果(总数)的低位 1 字节(8 位)的数值进行表示的代码。

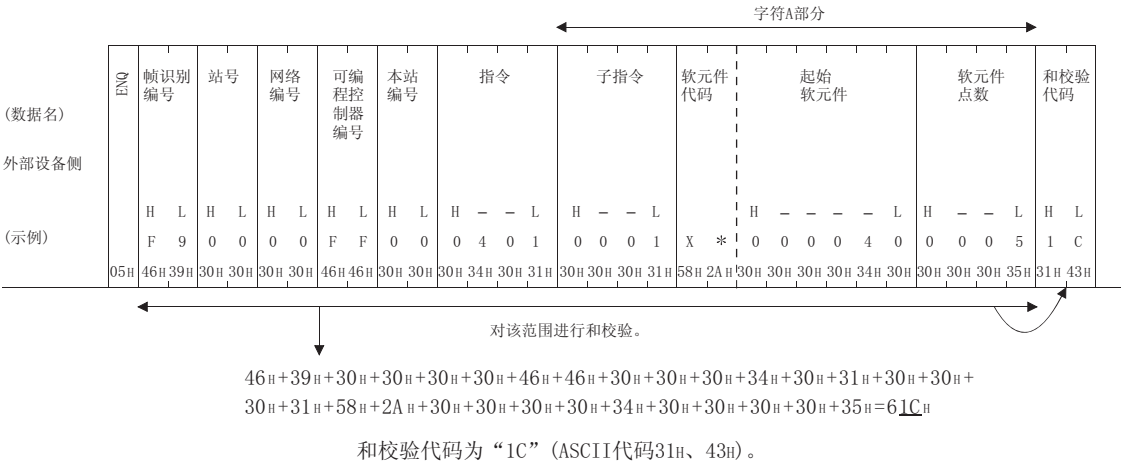
在 GX Works2 或 GX Developer 的传送规格设置中“和校验的有无设置”被设置为“有”时，需要附加和校验代码。

有和校验的情况下，可编程控制器 CPU 生成和校验代码并附加到发送报文中。此外，对接收报文中的和校验代码进行检查。

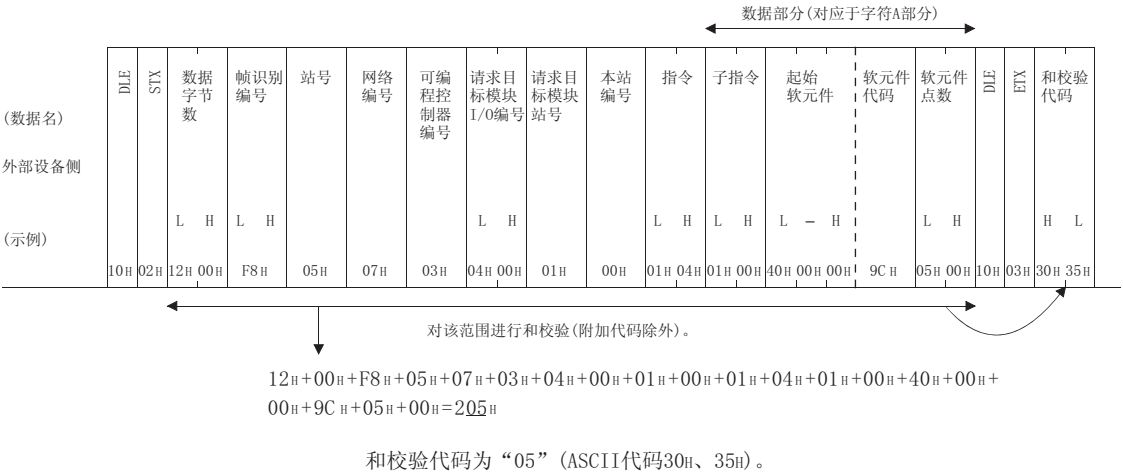
无和校验的情况下，可编程控制器 CPU 不在发送报文中附加和校验代码。此外，将接收报文作为无和校验代码处理。

- 1) 无论是通过 ASCII 代码进行数据通信时还是通过二进制代码进行数据通信时，均转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时，对除附加代码以外的和校验的对象范围(参阅 3.1.6 项(1)要点)进行和校验。
- 3) 和校验代码的内容示例如下所示。

(示例 1)
以 QnA 兼容 3C 帧格式 1 进行数据读取的情况下



(示例 2)
以 QnA 兼容 4C 帧格式 5 进行数据读取的情况下



(9) 响应识别代码(QnA 兼容 4C 帧的格式 5 用)

响应识别代码是指，用于外部设备识别是否对于外部设备的请求的响应报文的代码，使用 2 字节数值“FFFFH”进行发送。

(10) 结束代码(QnA 兼容 4C 帧的格式 5 的正常结束用)

结束代码是指，用于外部设备识别对于外部设备的请求可编程控制器 CPU 侧的处理是否正常进行的代码，使用 2 字节数值“0000H”进行发送。

(11) 出错代码

出错代码是指，用于外部设备识别对于外部设备的请求可编程控制器 CPU 侧的处理是否异常结束的代码。

1) 通过 ASCII 代码进行数据通信时

将出错代码转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。

(示例)

出错代码为 7151H 的情况下

发送至外部设备的出错代码变为“7151”，从“7”按顺序进行发送。

2) 通过二进制代码进行数据通信时

将出错代码作为 2 字节的数值使用，按照 Low 字节(L: 位 0 ~ 7)、High 字节(H: 位 8 ~ 15)的顺序进行发送。

(示例)

出错代码为 7151H 的情况下

按照 51H、71H 的顺序发送至外部设备。

3) 同时发生了多个出错时，可编程控制器 CPU 发送最先检测出的出错代码。**4) 关于出错代码的详细内容请参阅用户手册(基本篇)。**

备注

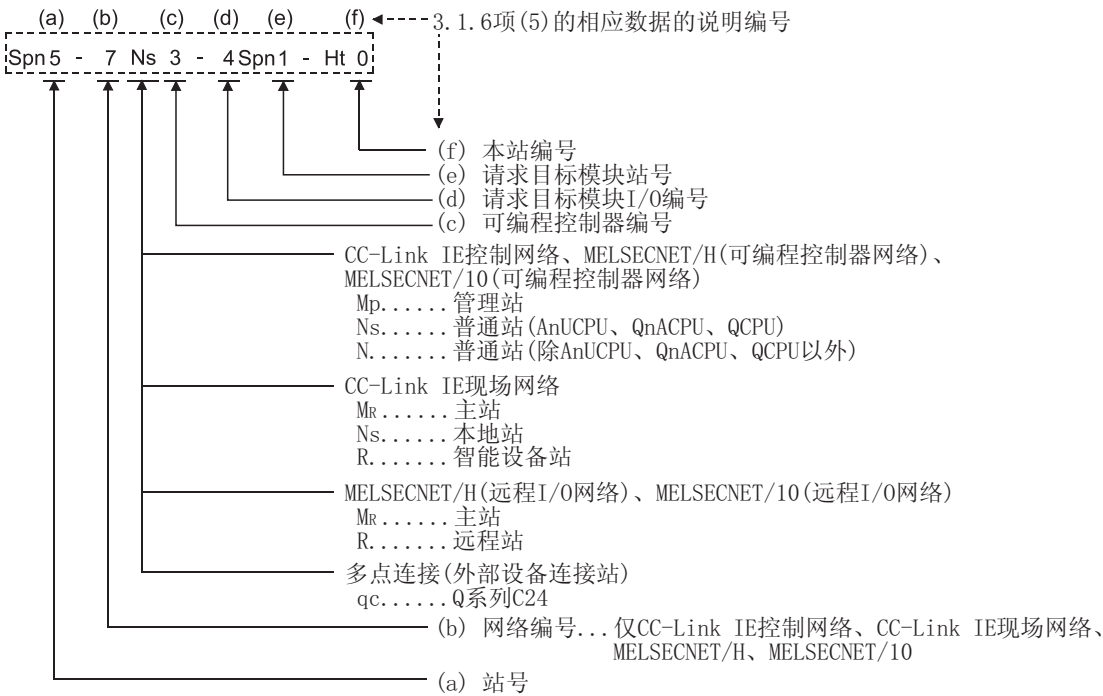
- (1) 在 QnA 兼容 3C 帧、QnA 兼容 4C 帧以及第 4 章中所示的 QnA 兼容 2C 帧的控制步骤内不指定“报文等待”时间。
以这些帧进行数据通信时的报文等待时间可通过 GX Works2 或 GX Configurator-SC 进行设置。

项目	设置方法
GX Works2	通过“各种控制指定”进行设置。
GX Configurator-SC	通过“MC 协议系统设置”画面的“报文等待时间指定”进行设置。

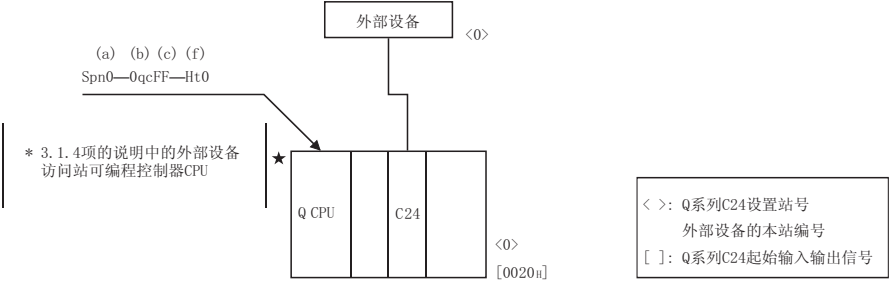
以 A 兼容 1C 帧进行数据通信时也包含在内，与可编程控制器 CPU 的 RS-422/485 接口上连接的外部设备进行数据通信时，应将报文等待时间指定为长于外部设备侧的硬件门 OFF 时间。

- (2) 以 QnA 兼容 3C 帧、QnA 兼容 4C 帧进行数据通信时，各控制步骤内指定的数据中“站号”~“本站编号”的指定示例如下所示。
(以下的 3)、4) 中所示的图·表的阅读方法)

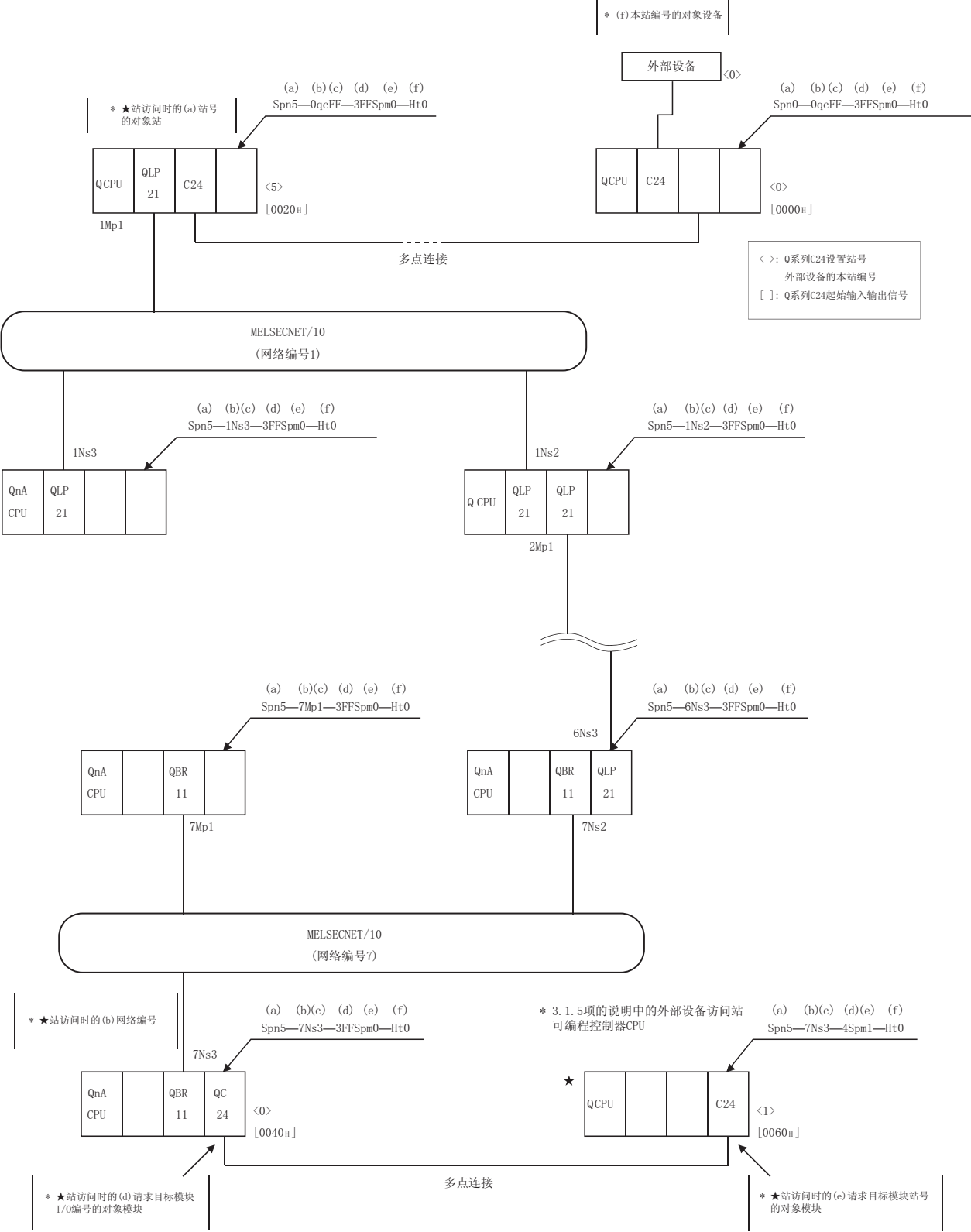
- 1) 图中的站的右侧所示的符号(< >、[]=内的数值的内容如下所示。
< >: 相应站中安装的 C24 的站号
[]: 相应站中安装的 C24 的起始输入输出信号
- 2) 至图中各站的箭头所表示符号的含义
符号的前或后的英文数字是指，外部设备对箭头所指站进行访问时，报文中的相应数据的内容(值)。



- 3) 以 QnA 兼容 3C 帧进行数据通信的格式说明项的外部设备连接站、访问目标
发送 3.1.4 项的各格式的说明中所示的传送数据(示例)的外部设备连接站、外部设备访问的可编程控制器 CPU 如下所示。



- 4) 以 QnA 兼容 4C 帧进行数据通信的格式说明项的外部设备连接站、访问目标
发送 3.1.5 项的各格式说明中所示的传送数据(示例)的外部设备连接站、
外部设备访问的可编程控制器 CPU 如下所示。



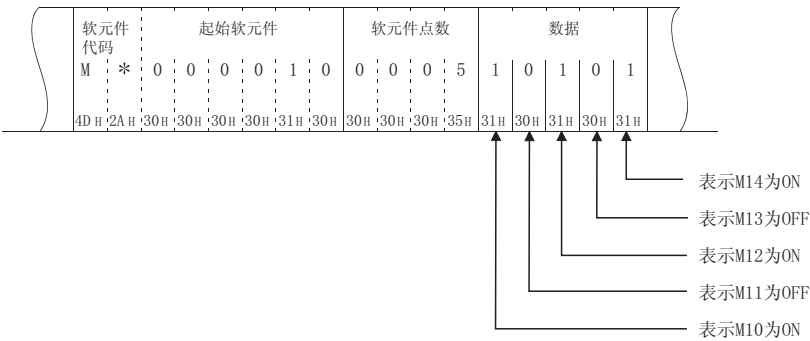
3.1.7 字符部分的传送数据的思路

以下对使用各指令在外部设备与可编程控制器 CPU 之间进行数据收发时的字符部分中使用的位软元件数据、字软元件数据的传送方法、传送时的排列思路有关内容进行说明。
示例中所示的传送数据在读取/监视的情况下为字符 B 部分，在写入/测试/监视数据的登录的情况下为字符 C 部分。

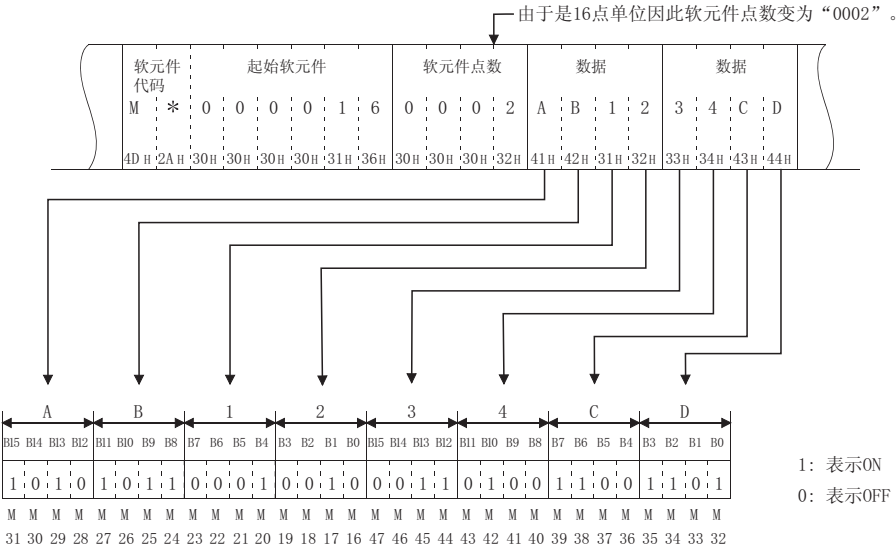
(1) 通过 ASCII 代码进行数据通信的情况下

- (a) 进行位软元件存储器的读取、写入时
位软元件存储器可以以位单位(1 点单位)进行处理以及以字单位(16 点)进行处理。
以下对各传送数据的思路进行说明。

- 1) 位单位(1 点单位)
将位软元件存储器以位单位进行处理时，从指定的起始软元件开始以指定软元件点数按从左至右的顺序为 ON 时表示“1”(31H)，为 OFF 时表示“0”(30H)。
(示例)
表示从 M10 开始 5 点的 ON/OFF 的情况下

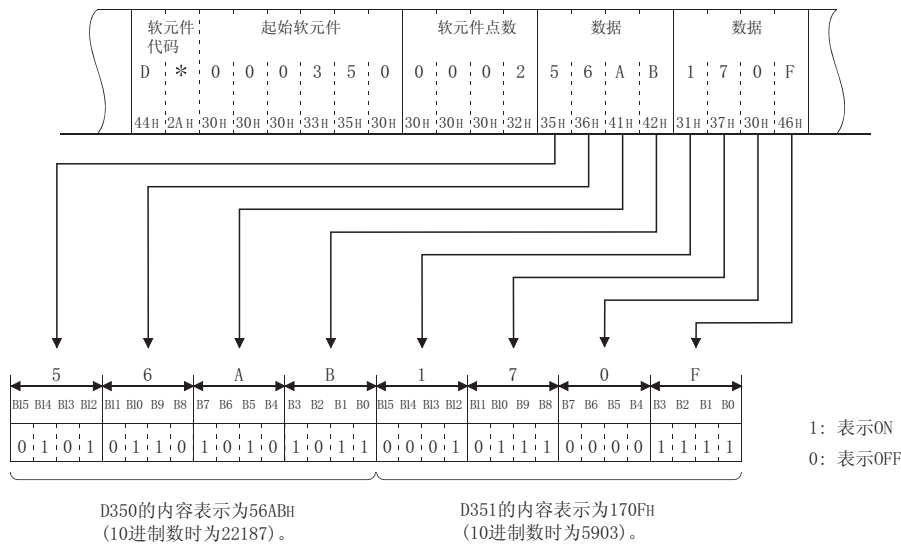


- 2) 字单位(16 点单位)
将位软元件存储器以字单位进行处理时，将 1 字以 4 位单位按从高位至低位的顺序以 16 进制数进行表示。
(示例)
表示从 M16 开始的 32 点的 ON/OFF 的情况下



- (b) 进行字软元件存储器的读取、写入时
字软元件存储器将 1 字以 4 位单位按从高位至低位的顺序以 16 进制数进行表示。
(示例)

表示 D350、D351 的数据寄存器的存储内容的情况下



要点

- (1) 字符部分中指定了英文的情况下，应使用大写字母。
(2) 数据读取字软元件存储器中存储了整数以外(实数、字符串)时，C24/E71 将存储值作为整数值进行读取。
(示例 1)
D0 ~ D1 中存储了实数(0.75)时，以下述整数值进行读取。
D0 = 0000H、D1 = 3F40H
(示例 2)
D2 ~ D3 存储了字符串(“ 12AB ”)时，以下述整数值进行读取。
D2 = 3231H、D3 = 4241H
(3) 对于本项(1)中所示的位软元件数据、字软元件数据的传送方法、传送时的数据的排列的思路，在通过下述帧进行通信时也与此相同。
· QnA 兼容 2C 帧(在第 4 章中介绍)
· A 兼容 1C 帧(在第 5 章中介绍)
· A 兼容 1E 帧(在第 6 章中介绍)

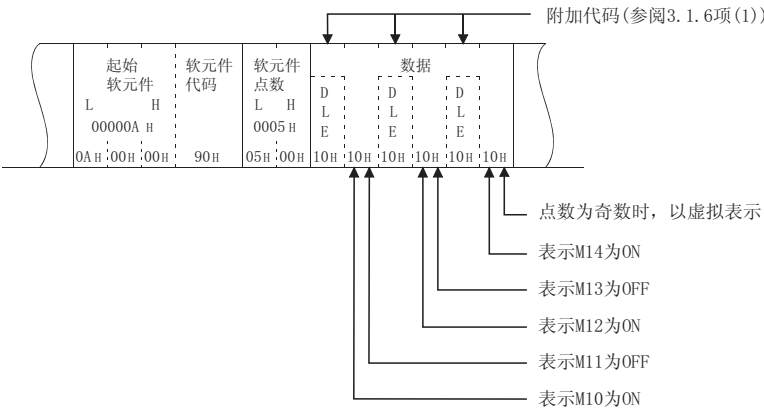
备注

缓冲存储器的读取、写入功能等中处理的字单位数据时的思路与字软元件存储器的相同。

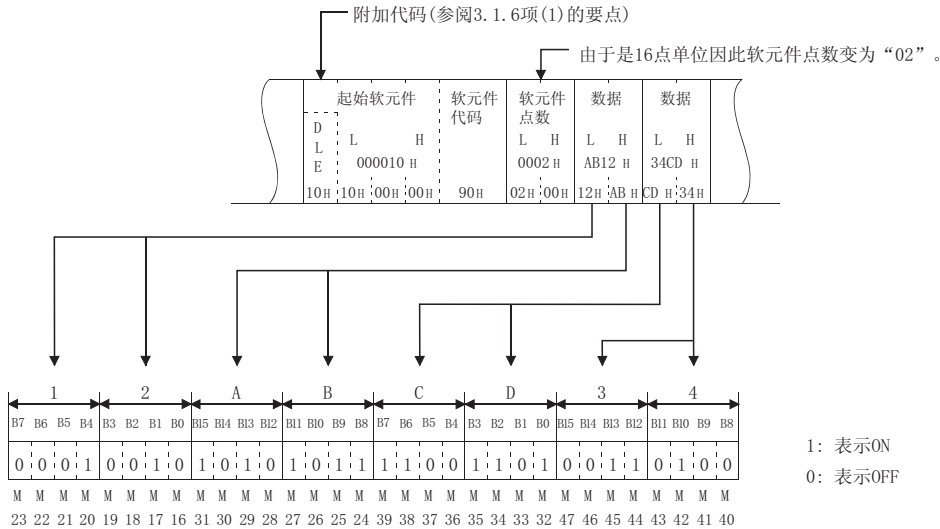
(2) 通过二进制代码进行数据通信的情况下

- (a) 进行位软元件存储器的读取、写入时
位软元件存储器可以以位单位(1 点单位)进行处理以及以字单位(16 点)进行处理。
以下对各传送数据的思路进行说明。

- 1) 位单位(1 点单位)
将位软元件存储器以位单位进行处理时，以 1 点 4 位进行指定，从指定的起始软元件开始以指定软元件点数按从高位至低位的顺序为 ON 时表示为“1”，为 OFF 时表示为“0”。
(示例)
表示从 M10 开始 5 点的 ON/OFF 的情况下



- 2) 字单位(16 点单位)
将位软元件存储器以字单位处理时，将 1 点以 1 位进行指定，从指定的起始软元件开始将指定软元件点数以 16 点单位按 Low 字节(L: 位 0~7)、High 字节(H: 位 8~15)的顺序进行表示。
(示例)
表示从 M16 开始 32 点的 ON/OFF 的情况下

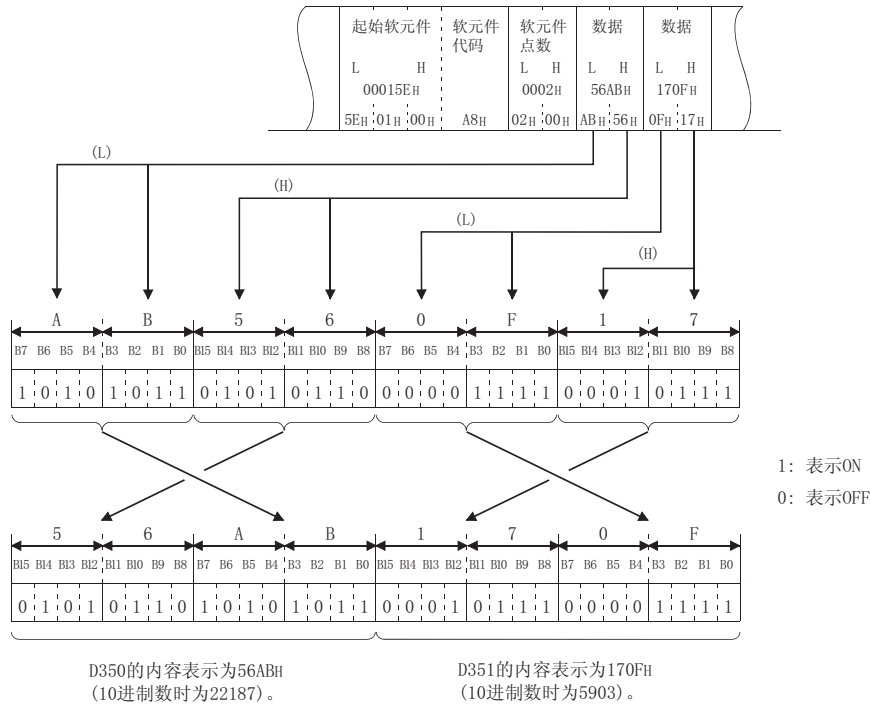


- (b) 进行字软元件存储器的读取、写入时

字软件存储器将 1 字以 16 位进行指定，从指定的起始软件开始将指定软件点数以 1 点单位按 Low 字节(L: 位 0~7)、High 字节(H: 位 8~15)的顺序进行表示。

(示例)

表示 D350、D351 的数据寄存器的存储内容的情况下



要点

- (1) 数据读取字软件元件存储器中存储了整数以外(实数、字符串)时,模块将存储值作为整数值进行读取。

(示例 1)

D0 ~ D1 中存储了实数(0.75)时，以下述整数值进行读取。

D0 = 0000_H, D1 = 3F40_H

(示例 2)

D2 ~ D3 存储了字符串(“12AB”)时,以下述整数值进行读取。

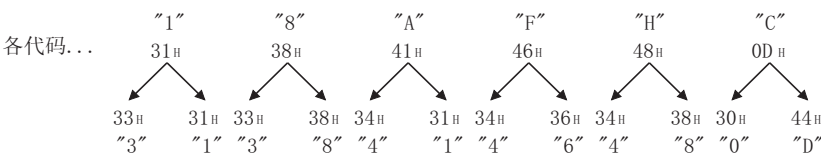
D2 = 3231_H, D3 = 4241_H

- (2) 对于本项中所示的位软元件数据、字软元件数据的传送方法、传送时的数据的排列的思路,在以 A 兼容 1E 帧(在第 6 章中介绍)进行通信时也相同。

备注

- (1) 扩展文件寄存器的读取/写入、缓冲存储器的读取/写入、通过字指定进行的接
通请求数据的思路与字软元件存储器的相同。
- (2) 通过 ASCII 数据进行通信的情况下，从外部设备向可编程控制器 CPU 传送字符
串，通过 PR 指令输出到外部时，应进行如下处理。

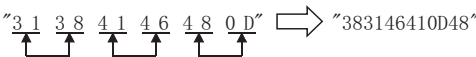
1) 将从外部设备发送的字符串展开为每个字符 2 字节的代码。



2) 将展开为 2 字节的字符串以 2 字符为单位进行排列变换后发送至
C24/E71。

(示例)

上述 1) 中生成的字符串的情况下

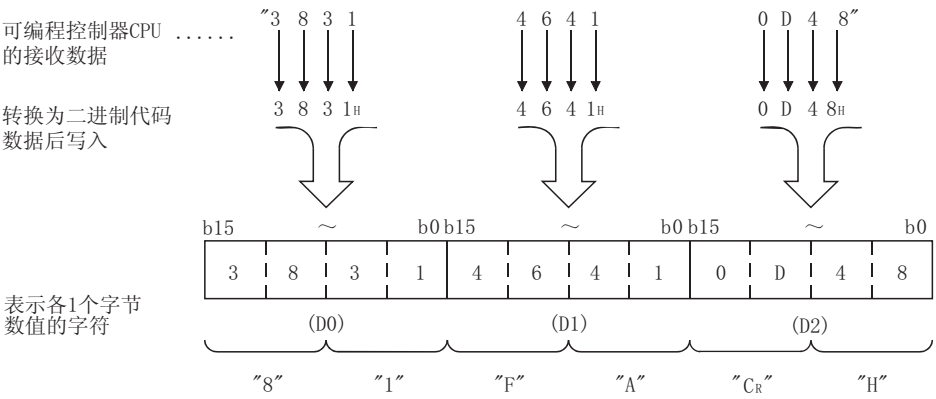


从外部设备将“383146410D48”发送至可编程控制器 CPU。

C24/E71 将从外部设备接收的数据转换为二进制代码数据后，写入到指定软元件中。

(示例)

将上述 2) 中生成的数据写入到可编程控制器 CPU 的 D0 ~ D2 中的情况下



3.2 QnA 兼容 3E/3C/4C 帧、4E 帧用指令及功能列表

以下介绍通过 QnA 兼容 3E/3C/4C 帧、4E 帧用指令从外部设备访问可编程控制器 CPU 时的指令及功能。

对于表中的(子指令)□的部分，根据指定的软元件、指令功能的使用方法等而有所不同。(请参阅 3.3 节以后的相应指令说明项。)

对于■部分的指令，是对多 CPU 系统的非管理 CPU 进行访问时可使用的指令。

至软元件存储器的访问指令对 Q/L/QnACPU 以外的可编程控制器 CPU 也可使用。

功能			指令(子指令)	处理内容	1 次通信中可处理的点数			
					访问站-1(*7)	访问站-2(*8)	访问站-3(*9)	
软元件存储器	多个块批量读取(*10)	字单位(*11)	0406 (00□0)	将字软元件及位软元件(1 点为 16 位)的 n 点作为 1 个块，随机指定多个块进行读取·写入。	960 点	480 点	(禁止)	
	多个块批量写入(*4)(*10)	字单位(*11)	1406 (00□0)			480 点	(禁止)	
	批量读取	位单位	0401 (00□1)	将位软元件(X、Y、M 等)以 1 点为单位进行读取。	3584 点 7168 点 7904 点	1792 点 3584 点 3952 点	256 点	
		字单位(*11)	0401 (00□0)	将位软元件(X、Y、M 等)以 16 点为单位进行读取。	960 字 (15360 点)	480 字 (7680 点)	32 字 (512 点)	
				将字软元件(D、R、T、C 等)以 1 点为单位进行读取。	960 点	480 点	64 点	
	批量写入(*4)	位单位	1401 (00□1)	以 1 点单位写入到位软元件(X、Y、M 等)中。	3584 点 7168 点 7904 点	1792 点 3584 点 3952 点	160 点	
		字单位(*11)	1401 (00□0)	以 16 点单位写入到位软元件(X、Y、M 等)中。	960 字 (15360 点)	480 字 (7680 点)	10 字 (160 点)	
				以 1 点单位写入到字软元件(D、R、T、C 等)中。	960 点	480 点	64 点	
	随机读取	字单位(*11)	0403 (00□0)	将位软元件(X、Y、M 等)以 16 点、32 点单位随机指定软元件、软元件编号后进行读取。 将字软元件(D、R、T、C 等)以 1 点、2 点单位随机指定软元件、软元件编号后 进行读取。	192 点	96 点	(禁止)	
	测试(随机写入)(*4)	位单位	1402 (00□1)	将位软元件(X、Y、M 等)以 1 点单位随机指定软元件、软元件编号后进行设置/复位。	188 点	94 点	20 点	
		字单位(*11)	1402 (00□0)	将位软元件(X、Y、M 等)以 16 点单位随机指定软元件、软元件编号后进行设置/复位。	1920 点	960 点	10 字 (160 点)	
				向字软元件(D、R、T、C 等)以 1 点、2 点单位随机指定软元件、软元件编号后进行写入。 但是，对于除 Q/L/QnACPU 以外只能以 1 点为单位写入。			10 点	
	监视数据登录	字单位(*11)	0801 (00□0)	将要监视的位软元件(X、Y、M 等)以 16 点单位进行登录。(*2) 对将要监视的字软元件(D、R、T、C 等)以 1 点、2 点单位进行登录。 但是，对于除 Q/L/QnACPU 以外只能以 1 点为单位写入。	192 点	96 点	(禁止)	
	监视	字单位(*11)	0802 (0000)	对进行了监视数据登录的软元件进行监视。	(登录点数)			

功能			访问站(参阅 2.6.1 项)									可编程控制器 CPU 的状态(*1)			指令执行允许模块		参阅章节
			ACPU	QnA CPU	QCPU (*12)	LCPU (*12)	安全 CPU	MELSEC NET/10 远程站	MELSEC NET/H 远程站	CC-Link IE 现场智能设备站	STOP 中	RUN 中		C24	E71		
												写入允许设	写入禁止设置				
	多个块批量读取	字单位(*11)	×	○	○	○	○	×	○	○	○	○	○	○	○	3.3.2 项	
	多个块批量写入	字单位(*11)	×	○	○	○	×	×	○	○	○	○	×				
	批量读取	位单位	○	○	○	○	○	×	○	○	○	○	○	○	○	3.3.3 项 3.3.6 项	
		字单位(*11)	○	○	○	○	○	○	×	○							○
			○	○	○	○	○	○	×	○							○
	批量写入	位单位	○	○	○	○	×	×	○	○	○	○	×	○	○	3.3.1 项 3.3.7 项	
		字单位(*11)	○	○	○	○	×	×	○	○							
			○	○	○	○	×	×	○	○							
	随机读取	字单位(*11)	×	○	○(*13)	○(*13)	○	×	○	○(*13)	○	○	○	○	○	3.3.9 项	
			×	○	○(*13)	○(*13)	○	×	○	○(*13)							
	测试(随机写入)	位单位	○	○	○	○	×	×	○	○	○	○	×	○	○	3.3.5 项 3.3.8 项	
		字单位(*11)	○	○	○	○	×	×	○	○							
			○	○	○	○	×	×	○	○							
	监视数据登录	字单位(*11)	×	○	○(*13)	○(*13)	×	×	○	○(*13)	○	○	○	○	○	3.3.10 项	
			×	○	○(*13)	○(*13)	×	×	○	○(*13)							
	监视	字单位(*11)	×	○	○	○	×	×	○	○	○	○	○				

— 功能			指令(子指令)	处理内容	1 次通信中可处理的点数					
					访问站-1(*7)	访问站-2(*8)	访问站-3(*9)			
缓冲存储器(*3)	批量读取		0613 (0000)	对与外部设备相连接的 C24/E71 的缓冲存储器的数据进行读取。	也可作为可编程控制器 CPU 与外部设备之间的数据通信使用。	960 字 (1920 字节)	480 字 (960 字节)			
	批量写入		1613 (0000)	对与外部设备相连接的 C24/E71 的缓冲存储器进行数据写入。						
智能功能模块	批量读取		0601 (0000)	对智能功能模块的缓冲存储器的数据进行读取。		1920 字节	960 字节			
	批量写入		1601 (0000)	对智能功能模块的缓冲存储器进行数据写入。						
可编程控制器 CPU(*4)	远程 RUN		1001 (0000)	对可编程控制器 CPU 发出远程 RUN 请求。		(1 站)	(1 站)			
	远程 STOP		1002 (0000)	对可编程控制器 CPU 发出远程 STOP 请求。		(1 站)	(1 站)			
	远程 PAUSE		1003 (0000)	对可编程控制器 CPU 发出远程 PAUSE 请求。		(1 站)	(1 站)			
	远程锁存清除		1005 (0000)	可编程控制器 CPU 处于 STOP 状态时, 对可编程控制器 CPU 发出远程锁存清除请求。		(1 站)	(1 站)			
	远程 RESET		1006 (0000)	为了对可编程控制器 CPU 的出错停止状态进行解除, 对可编程控制器 CPU 发出了远程 RESET 请求。		(1 站)	(1 站)			
	CPU 型号读取		0101 (0000)	通过可编程控制器 CPU 读取型号代码。		(1 站)	(1 站)			
驱动存储器	存储器使用状态读取		0205 (0000)	对驱动器的簇的使用状态进行读取。		(256 簇)				
	存储器碎片整理(*4)(*5)		1207 (0000)	进行驱动器内存的碎片整理, 增加连续可用空间。(文件存储位置的碎片整理)		(1 站)				
文件	文件信息列表读取	无索引	0201 (0000)	对文件列表(文件名、最后编辑日期、文件大小)进行读取。	(禁止)	(禁止)	(36 个)	(禁止)		
		有索引	0202 (0000)	对文件列表(文件附带的索引、文件名、最后编辑日期、文件大小)进行读取。			(16 个)			
		文件 No. 使用状况	0204 (0000)	对文件 No. 的使用状况进行读取。			(256 个)			
	文件信息更改(*4)(*6)	最后编辑日期更改	1204 (0000)	对文件的最后编辑日期进行更改。			(1 个)			
		文件名、大小更改	1204 (0001)	对文件名、文件大小进行更改。			(1 个)			
		批量更改	1204 (0002)	对文件名、文件大小、最后编辑日期进行更改。			(1 个)			
	文件搜索		0203 (0000)	指定文件的有无, 对文件 No.、文件大小进行读取。					(1 个)	
	文件内容读取(*6)		0206 (0000)	对文件内容进行读取。					960 字节	
	新登录(文件名登录)(*4)(*6)		1202 (0000)	预留指定文件名的文件区域。					(1 个)	
	文件内容写入(*4)(*6)	任意数据	1203 (0000)	将指定数据(n 字节)写入到文件中。					960 字节	
		同一数据(FILL)	1203 (0001)	将 n 字节的指定数据(1 字)写入到文件中。					文件大小	
	文件锁定登录/解除		0808 (000□)	对指定文件的访问过程中, 为了防止从其它设备对内容的更改, 对文件锁定进行登录。或对登录进行解除。					(1 个)	
	文件复制(*4)(*6)		1206 (0000)	将原有文件的内容写入到新登录的文件中(复制)					960 字节	480 字节
	文件删除(*4)(*6)		1205 (0000)	将文件删除。					(1 个)	

功能		访问站(参阅 2.6.1 项)									可编程控制器 CPU 的状态(*1)			指令执行允许模块		参阅章节
		A CPU	QnA CPU	QCPU (*12)	LCPU (*12)	安全 CPU	MELSECNET/10 远程站	MELSECNET/H 远程站	CC-Link 远端站	STOP 中	RUN 中					
											写入允许设置	写入禁止设置	C24	E71		
缓冲存储器批量读取		-	-	-	-	-	-	-	-	○	○	○	○	○	3.4 项	
缓冲存储器批量写入		-	-	-	-	-	-	-	-	○	○	○				
智能功能模块批量读取		×	○	○	○	○	○(仅 QnA 系列)	○	○	○	○	○	○	○	3.5 项	
智能功能模块批量写入		×	○	○	○	×	○(仅 QnA 系列)	○	○	○	○	○				
远程 RUN		×	○	○	○	×	×	×	○	○	○	○	○	×	3.6 项	
远程 STOP		×	○	○	○	×	×	×	○							
远程 PAUSE		×	○	○	○	×	×	×	×							
远程锁清除		×	○	○	○	×	×	×	×	○	×	×				
远程 RESET		×	○	○	○	×	×	×	○	○	×	×				
CPU 型号读取		×	×	○	○	○	×	×	○	○	○	○				
存储器使用状态读取		×	○	×	×	×	×	×	×	○	○	○	○	○	3.7 项	
存储器碎片整理		×	○	×	×	×	×	×	×	○	×	×				
文件信息列表读取	无索引	×	○	×	×	×	×	×	×	○	○	○	○	○	3.8.16 项	
	有索引	×	○	×	×	×	×	×	×							
	文件 No.使用状况	×	○	×	×	×	×	×	×							×
文件信息更改	最后编辑日期更改	×	○	×	×	×	×	×	×	○	○	×	○	○	3.8.24 项	
	文件名、大小更改	×	○	×	×	×	×	×	×							
	批量更改	×	○	×	×	×	×	×	×							×
文件搜索		×	○	×	×	×	×	×	×	○	○	○	○	○	3.8.17 项	
文件内容读取		×	○	×	×	×	×	×	×	○	○	○	○	○	3.8.19 项	
新登录(文件名登录)		×	○	×	×	×	×	×	×	○	○	×	○	○	3.8.20 项	
文件内容写入	任意数据	×	○	×	×	×	×	×	×	○	○	×	○	○	3.8.21 项	
	同一数据(FILL)	×	○	×	×	×	×	×	×							
文件锁定登录/解除		×	○	×	×	×	×	×	×	○	○	○	○	○	3.8.18 项	
文件复制		×	○	×	×	×	×	×	×	○	○	○	○	○	3.8.23 项	
文件删除		×	○	×	×	×	×	×	×	○	○	×	○	○	3.8.22 项	

— 功能		指令(子指令)	处理内容	1 次通信中可处理的点数					
				访问站-1(*7)	访问站-2(*8)	访问站-3(*9)			
文件	目录·文件信息读取	1810 (0000)	对文件列表信息进行读取。	(36 个)	(禁止)				
	目录·文件信息搜索(*6)	1811 (0000)	对指定文件的文件 No. 进行读取。	(1 个)					
	新建文件(*4)(*6)	1820 (0000)	预留指定文件的存储区域。	(1 个)					
	文件删除(*4)(*6)	1822 (0000)	删除文件。	(1 个)					
		1822 (0004)							
	文件复制(*4)(*6)	1824 (0000)	对指定文件进行复制。	(1 个)					
		1824 (0004)							
	文件属性更改(*4)(*6)	1825 (0000)	对文件的属性进行更改。	(1 个)					
		1825 (0004)							
	文件创建日期更改(*4)	1826 (0000)	对文件创建日期进行更改。	(1 个)					
	打开文件(*6)	1827 (0000)	为了防止来自于其它设备的内容更改对文件进行锁定。	(1 个)					
		1827 (0004)							
	文件读取(*6)	1828 (0000)	对文件的内容进行读取。	1920 字节					
文件写入(*4)(*6)	1829 (0000)	将内容写入到文件中。	1920 字节						
文件关闭	182A (0000)	解除通过打开处理进行的文件锁定。	(1 个)						
用户登录帧(*3)	登录数据读取	0610 (0000)	对指定帧 No. 的登录数据进行读取。	80 字节					
	数据登录	1610 (0000)	对以用户格式的报文格式进行数据通信时的起始帧/最终帧的数据排列进行登录。(写入)	(1 个)*7(1)5)					
	登录数据删除	1610 (0001)	将指定帧 No. 的登录数据删除。	(1 站/所有站)					
全局(*3)	1618 (000□)	对安装了 C24 的 Q/L/QnACPU 将全局信号(X1A/X1B)置为 ON/OFF。	同上						
接通请求	2101 (——)	从可编程控制器 CPU 发出发送请求, 将数据发送至外部设备。 可以进行 C24 缓冲存储器的用户自由区中的连续可用空间的最大区域大小的数据发送。 (系统配置为 1: 1 的情况下可以发送)	(1 站) *7 仅 1)站可以						
传送顺控程序初始化(仅二进制模式时可以使用)(*3)	1615 (0000)	中止当前的处理请求, 将 C24 置为指令接收等待。	(1 站) *7 仅 1)、3) 站可以						
模式切换(*3)	1612 (0000)	对指定接口的动作模式、传送规格进行切换。	(1 站)						
LED 熄灯、出错代码初始化(*3)	1617 (000□)	对显示出错 LED 的熄灯、出错代码进行初始化。	(1 站) *7 仅 1)、3) 站						
			(禁止)						
回送测试(*3)	0619 (0000)	对 C24/E71 与外部设备之间的数据通信是否正常进行确认。(用于检查连接状态及通信功能)	960 字节 (仅连接站可以通信)						
可编程控制器 CPU 监视	登录	0630 (0000)	对软元件存储器的监视、CPU 状态的监视进行登录, 开始可编程控制器 CPU 监视的动作。	960 点					
	解除	0631 (0000)	结束可编程控制器 CPU 监视的动作。	-					
远程口令	解锁	1630 (0000)	指定远程口令, 从锁定状态置为解锁状态。 (置为可以对可编程控制器 CPU 进行通信的状态。)	-					
	锁定	1631 (0000)	指定远程口令, 从解锁状态置为锁定状态。 (置为不能与可编程控制器 CPU 进行通信的状态。)	-					

功能	—		访问站(参阅 2.6.1 项)							可编程控制器 CPU 的状态(*1)			指令执行允许模块		参阅章节	
			A CPU	QnA CPU	QCPU (*12)	LCPU (*12)	安全 CPU	MELSEC NET/10 远程站	MELSEC NET/H 远程站	CC-Link 现场设备站	STOP 中	RUN 中		C24		E71
												写入允许设置	写入禁止设置			
文件	目录・文件信息读取	×	×	○	○	○	×	×	○	○	○	○	○	○	3.8.5 项	
	目录・文件信息搜索	×	×	○	○	○	×	×	○	○	○	○	○	○	3.8.6 项	
	新建文件	×	×	○	○	×	×	×	○	○	○	×	○	○	3.8.10 项	
	文件删除	×	×	○	×	×	×	×	×	○	○	×	○	○	3.8.12 项	
		×	×	×	○	×	×	×	○	○	○	×	○	○		
	文件复制	×	×	○	×	×	×	×	×	○	○	C24: ×	○	○	3.8.13 项	
		×	×	×	○	×	×	×	○	○	○	×	○	○		
	文件属性更改	×	×	○	×	×	×	×	×	○	○	×	○	○	3.8.15 项	
		×	×	×	○	×	×	×	○	○	○	×	○	○		
	文件创建日期更改	×	×	○	○	×	×	×	○	○	○	×	○	○	3.4.14 项	
	打开文件	×	×	○	×	○	×	×	×	○	○	○	○	○	3.8.7 项	
		×	×	×	○	×	×	×	○	○	○	○	○	○		
	文件读取	×	×	○	○	○	×	×	×	○	○	○	○	○	3.8.9 项	
文件写入	×	×	○	○	×	×	×	×	○	○	○	×	○	3.8.11 项		
文件关闭	×	×	○	○	○	×	×	×	○	○	○	○	○	3.8.8 项		
数据登录	登录数据读取	-	-	-	-	-	-	-	-	○	○	○	○	×	3.9 项	
	数据登录	-	-	-	-	-	-	-	-	○	○	○				
	登录数据删除	-	-	-	-	-	-	-	-	○	○	○				
全局		×	○	○	○	×	×	×	×	×	○	○	○	×	3.10 项	
接通请求		×	○	○	○	×	×	×	×	×	○	○	○	×	3.11 项	
传送顺控程序初始化 (仅二进制模式时可以使用)		-	-	-	-	-	-	-	-	○	○	○	○	×	3.12 项	
模式切换		-	-	-	-	-	-	-	-	○	○	○	○	×	3.13 项	
LED 熄灭、出错代码初始化		-	-	-	-	-	-	-	-	○	○	○	○	×	3.14 项	
													×	○	3.15 项	
回送测试		-	-	-	-	-	-	-	-	○	○	○	○	○	3.16 项	
可编程控制器 CPU 监视登录		-	-	-	-	-	-	-	-	○	○	○	○	×	3.17 项	
	可编程控制器 CPU 监视解除	-	-	-	-	-	-	-	-	○	○	○				
远程口令解锁		-	-	-	-	-	-	-	-	○	○	○	○	○	3.1.18 项	
远程口令锁定		-	-	-	-	-	-	-	-	○	○	○	×			

- *1 至可编程控制器 CPU 的运行中写入的允许/禁止设置是在 GX Works2 或 GX Developer 的以下画面中进行。
 - C24 的情况下：
“ I/O 模块、智能功能模块开关设置 ” 画面
 - E71 的情况下：
“ 以太网动作设置 ” 画面
- *2 AnA/AnU/QnA/LCPU/QCPU 以外的情况下，软元件 X(输入)的 1 点作为 2 点的处理点数。
指定软元件中包含有 X 时，应满足以下公式：

$$((X \text{ 的指定点数} \times 2) + \text{其它软元件指定点数}) \div 1 \text{ 次通信中可处理的点数}$$
 仅指定了 X 的情况下，1 次通信中可处理的点数将变为表中值的 1/2。
- *3 只能对与外部设备相连接的 C24(多点连接站也包含在内)或该 C24 安装站的 Q/QnACPU 执行指令。
不能对经由网络系统的其它站可编程控制器执行指令。
- *4 在执行指令的 Q/L/QnACPU 中实施了系统保护时将变为出错状态，将返回异常响应。
- *5 执行指令的 Q/L/QnACPU 中登录有数据写入用/读取用的关键字(口令)的情况下，应将相同的关键字指定到指令报文中。关键字不一致时将变为出错状态，返回异常响应。
- *6 对程序文件或参数文件执行指令时，相应 Q/L/QnACPU 中登录有数据写入用/读取用的关键字的情况下，应将相同的关键字指定到指令报文中。
关键字不一致时将变为出错状态，返回异常响应。
- *7 “ 访问站-1 ” 表示对以下某个站进行访问。
 - 1) C24/E71 安装站(本站)
 - 2) 经由 Q/L 系列对应网络系统(CC-Link IE 控制网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU 站(其它站)/CC-Link IE 现场网络的智能设备站/MELSECNET/H 远程 I/O 站
 - 3) 上述 1)、2)的 C24 及多点连接的 Q/LCPU 站
- *8 “ 访问站-2 ” 表示对以下某个站进行访问。
 - 1) 经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU 站(其它站)/MELSECNET/10 远程 I/O 站
 - 2) 上述 1)中安装的 C24 及多点连接的 Q/QnACPU 站(其它站)
 - 3) QnACPU 及经由网络系统(MELSECNET/H、MELSECNET/10、以太网)的 Q/QnACPU 站(其它站)/MELSECNET/H、MELSECNET/10 远程 I/O 站
- *9 “ 访问站-3 ” 表示对除上述*7、*8 以外的站进行访问。
(示例)
Q/L/QnACPU 站以外的可编程控制器 CPU(其它站)
- *10 关于可进行多个块批量读取·写入的 QnACPU，请参阅 3.3.10 项。

- *11 可对下述软元件进行访问的 C24/E71 的版本是有限制的。有关内容请参阅用户手册(基本篇)。
 - LW10000 以后的链接直接软元件
 - D65536 以后的扩展数据寄存器、W10000 以后的扩展链接寄存器
- *12 对 CPU 模块的内置以太网端口进行访问时，请参阅下述手册。
 - QnUCPU 用户手册(内置以太网端口通信篇)
 - MELSEC-L CPU 模块用户手册(内置以太网功能篇)
- *13 在通用型 QCPU、LCPU 以及 CC-Link IE 现场网络中，不能指定监视条件。
- *14 对象 CPU 模块中未设置关键字(口令)的情况下，可以执行指令并正常响应。

3.3 软元件存储器的读取、写入

以下对进行软元件存储器的读取、写入时的控制步骤的指定内容以及指定示例有关内容进行说明。

3.3.1 指令、字符部分的内容及软元件范围

以下对进行软元件存储器的读取、写入时的指令、控制步骤内的字符部分(二进制代码时为数据部分)、访问允许软元件范围有关内容进行说明。

(1) 指令

功能		指令(*1) (子指令)	处理内容	1 次通信中可处理的点数 (参阅 3.2 节*7~*9)			可编程控制器 CPU 的状态(参阅 3.2 项*1)			参阅章节
				访问站-1	访问站-2	访问站-3	STOP 中	RUN 中		
								写入允许设置	写入禁止设置	
多个块批量读取	字单位	0406 (00□0)	将 n 点的字软元件或位软元件(1 点为 16 位)作为 1 个块, 随机指定多个块进行读取。	960 点	480 点	(禁止)	○	○	○	3.3.2 项
多个块批量写入	字单位	1406 (00□0)	将 n 点的字软元件或位软元件(1 点为 16 位)作为 1 个块, 随机指定多个块进行写入。	960 点	480 点	(禁止)	○	○	×	
批量读取	位单位	0401 (00□1)	将位软元件以 1 点为单位进行读取。	3584 点 7168 点 7904 点	1792 点 3584 点 3952 点	256 点	○	○	○	3.3.3 项 3.3.6 项
	字单位	0401 (00□0)	将位软元件以 16 点为单位进行读取。	960 字 (15360 点)	480 字 (7680 点)	32 字 (512 点)				
			将字软元件以 1 点为单位进行读取。	960 点	480 点	64 点				
批量写入 (参阅 3.2 节*4)	位单位	1401 (00□1)	以 1 点单位写入到位软元件中。	3584 点 7168 点 7904 点	1792 点 3584 点 3952 点	160 点	○	○	×	3.3.4 项 3.3.7 项
	字单位	1401 (00□0)	以 16 点单位写入到字软元件中。	960 字 (15360 点)	480 字 (7680 点)	10 字 (160 点)				
			以 1 点单位写入到字软元件中。	960 点	480 点	64 点				
随机读取(*2)	字单位	0403 (00□0)	将随机指定的位软元件以 16 点、32 点单位进行读取。 将随机指定的字软元件以 1 点、2 点单位进行读取。	192 点	96 点	(禁止)	○	○	○	3.3.9 项
测试(*2) (随机写入) (参阅 3.2 项*4)	位单位	1402 (00□0)	以 1 点单位对随机指定的位软元件进行设置/复位。	188 点	94 点	20 点	○	○	×	3.3.5 项 3.3.8 项
	字单位	1402 (00□0)	以 16 点、32 点单位对随机指定的位软元件进行设置/复位。	(参阅 3.3.8 项)		10 字 (160 点)				
			以 1 点、2 点单位写入到随机指定的字软元件中。 (参阅 3.2 项*2)			10 点				
监视数据登录(*2)	字单位	0801 (00□0)	将进行监视的位软元件以 16 点、32 点单位进行登录。 将进行监视的字软元件以 1 点单位、2 点单位进行登录。	192 点	96 点	(禁止)	○	○	○	3.3.10 项
监视	字单位	0802 (0000)	对进行了监视数据登录的软元件执行监视。	(登录点数)			○	○	○	

上述表的可编程控制器 CPU 的状态栏的○符号表示可以执行。

*1 各指令可以访问的 CPU 模块有所不同。(参阅 3.2 节)

*2 对于 ACPU 的处理点数, 字软元件以 1 点为单位, 位软元件以 16 点为单位。

(2) 字符部分(二进制代码时为数据部分)

以下介绍外部设备对可编程控制器进行数据读取、写入时，各控制步骤内的字符部分的通用数据内容。

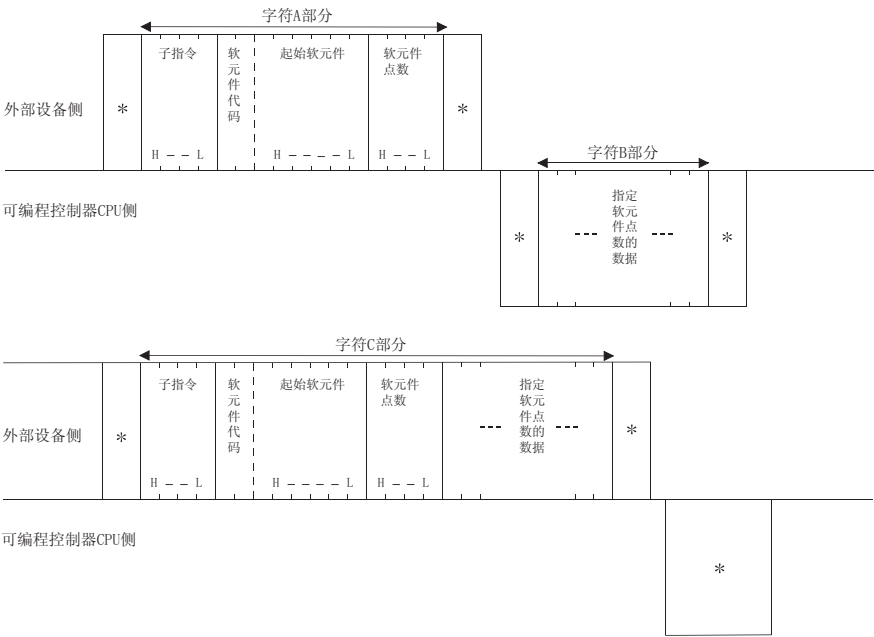
要点
(1) 字符部分根据所使用的指令及指定的内容而有所不同。 在本项中，对执行读取、写入的软元件存储器进行直接指定时的字符部分的通用数据内容进行说明。
(2) 关于仅通过任意指令处理的字符部分的数据有关内容，如相应指令的说明项中所示。
(3) 关于仅通过部分功能处理的数据、用于通过特殊表示对软元件存储器进行扩展指定的数据的有关内容，在以下章节中进行说明。 • 3.3.9 项：通过字单位的随机读取(指令：0403)、监视数据登录(指令：0801)功能指定的数据。 • 附录 1：对执行读取、写入的软元件存储器进行扩展指定时的数据。

(a) 通过 ASCII 代码进行通信时的字符部分的数据

在通过 ASCII 代码进行通信的控制步骤中，对于将同一指令在相同条件下使用时的字符 A 部分、字符 B 部分、字符 C 部分的数据的排列及内容均相同。

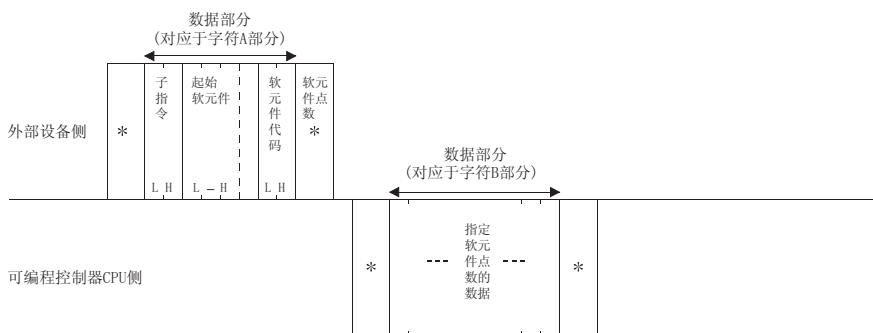
(示例)

读取时

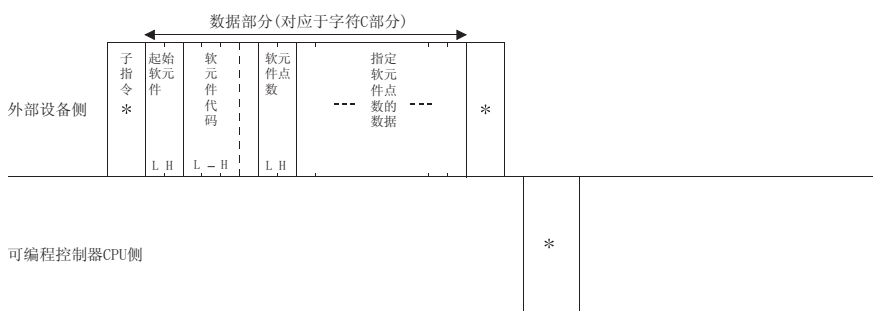


关于*符号部分的数据的排列及内容，请参阅 3.1 节。

(b) 二进制代码时的数据部分的数据
(示例)
读取时



(示例)
写入时

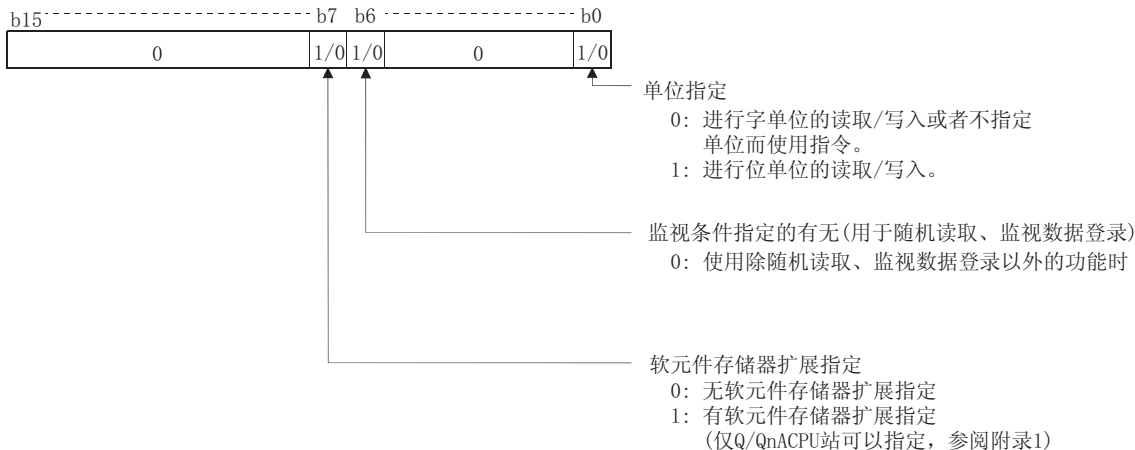


关于*符号部分的数据的排列及内容，请参阅 3.1 节。

(c) 字符部分的通用数据的内容
1) 子指令

是用于对读取/写入的单位、指定软元件的类型、数据读取条件等进行指定的数据。

- a) 通过 ASCII 代码进行数据通信时
将 0000_H(0)或下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- b) 通过二进制代码进行数据通信时
使用 0000_H或下述 2 字节的数值进行发送。
- c) 子指令的指定内容如下所示。



- d) 在以下情况下，子指令变为 0000_H/0001_H。
 - 选择无监视条件指定以及无软元件存储器扩展指定时。
 - 使用不能选择监视条件指定以及软元件存储器扩展指定的指令时。

要点
对于如下所示的 C24/E71 安装站以及其它站 Q/L/QnACPU 的软元件存储器，可以通过子指令位 7 的“软元件存储器扩展指定”进行访问。请参阅附录 1。 1) Q/L/QnACPU 的直接输入输出用的软元件存储器 2) 智能功能模块的缓冲寄存器(缓冲存储器)

- 2) 软元件代码
是用于识别进行数据读取/写入的软元件存储器的数据。
 - a) 软元件代码如(3)的表中所示。
 - b) 通过 ASCII 代码进行数据通信时
将软元件代码转换为 ASCII 代码 2 位后使用，从高位开始进行发送。
(示例)
输入继电器的情况下
对于输入继电器的软元件代码“X*”、从“X”开始按顺序进行发送。
此外，对于第 2 字符的“*”，也可以空格(代码: 20_H)进行指定。
 - c) 通过二进制代码进行数据通信时
使用(3)的表中所示的 1 字节的数值进行发送。
- 3) 起始软元件(软元件)
是用于指定进行数据读取/写入的软元件存储器的编号的数据。
连续指定软元件存储器时，对该软元件存储器范围的起始编号进行指定。
对于起始软元件的编号，根据对象软元件存储器通过(3)中表格中的“表示”栏中所示的表示方法(10 进制数或 16 进制数)进行指定。
 - a) 通过 ASCII 代码进行数据通信时
将表中所示的软元件编号转换为 ASCII 代码 6 位后使用，从高位开始进行发送。
此外，对于高位的“0”数字串(例如“001234”的起始 2 字符的“0”)，也可以以空格(代码: 20_H)进行指定。
(示例)
内部继电器 M1234，链接继电器 B1234 的情况下
内部继电器 M1234 及链接继电器 B1234 均变为“001234”或“₀₀ 1234 1234”，从“0”或“₀₀”开始按顺序进行发送。
 - b) 通过二进制代码进行数据通信时
使用表中所示的 3 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
(示例)
内部继电器 M1234，链接继电器 B1234 的情况下
内部继电器 M1234 变为 0004D2_H，按 D2_H、04_H、00_H 的顺序进行发送。
链接继电器 B1234 变为 001234_H，按 34_H、12_H、00_H 的顺序进行发送。

- 4) 软元件点数
- 是执行各指令时，用于指定进行读取/写入的点数的数据，在(1)的表中所示的1次通信中可处理的点数以内进行指定。
- a) 过 ASCII 代码进行数据通信时
- 将处理点数转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
- (示例)
- 5 点的情况：变为“0005”，从“0”开始按顺序进行发送。
下
20 点的情：变为“0014”，从“0”开始按顺序进行发送。
况下
- b) 通过二进制代码进行数据通信时
- 使用表示处理点数的 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
- (示例)
- 5 点的情况：变为 0005H，按 05H、00H 的顺序进行发送。
下
20 点的情：变为 0014H，按 14H、00H 的顺序进行发送。
况下
- 5) 指定软元件点数的数据
- 是表示向指定软元件存储器写入数据内容或从指定软元件存储器读取的数据内容的数据，根据处理单位(字/字节)数据的排列有所不同。关于数据的内容及排列(传送顺序)，请参阅 3.1 节。
- 6) 位访问点数
- 是用于对以位单位访问的点数进行指定的数据，在 3.3.1 项(1)的表中所示的1次通信中可处理的点数以内进行指定。
- a) 通过 ASCII 代码进行数据通信时
- 将点数转换为 ASCII 代码 2 位(16 进制数)后使用，从各高位进行发送。
- (示例)
- 5 点的情况：变为“05”，从“0”开始按顺序进行发送。
下
20 点的情：变为“14”，从“1”开始按顺序进行发送。
况下
- b) 通过二进制代码进行数据通信时
- 使用表示点数的 1 字节的数值进行发送。
- (示例)
- 5 点的情况：发送 05H。
下
20 点的情：发送 14H。
况下
- 7) 设置/复位
- 是用于指定写入到位软元件中的数据的数据，以如下所示的值进行指定。

	写入数据		备注
	ON	OFF	
ASCII 代码	“01”	“00”	从“0”开始按顺序发送 2 字符
二进制代码	01H	00H	发送如左所示的 1 字节的数值

(3) 软元件范围

以下介绍可访问的可编程控制器 CPU 的软元件及软元件编号范围。
应指定进行数据读取、写入的对象模块中存在的软元件、软元件编号范围。

(a) Q/L/QnACPU 的情况下*1

1) 基本型 QCPU

分类	软元件		类型	软元件代码		软元件编号范围(默认值)		备注	
				ASCII 代码时	二进制 代码时				
内部系统软 元件	功能输入		位	-	-	000000 ~ 00000F	16 进制	• 访问禁止	
	功能输出			-	-	000000 ~ 00000F	16 进制		
	功能寄存器		字	-	-	000000 ~ 000004	10 进制	-	
	特殊继电器		位	SM	91 _h	000000 ~ 001023	10 进制		
	特殊寄存器		字	SD	A9 _h	000000 ~ 001023	10 进制		
内部用户软 元件	输入		位	X *	9C _h	000000 ~ 0007FF	16 进制	• 更改分配时，可以访问至更改后的最大 软元件编号。 • 禁止访问局部软元件。	
	输出			Y *	9D _h	000000 ~ 0007FF	16 进制		
	内部继电器*2			M *	90 _h	000000 ~ 008191	10 进制		
	锁存继电器*2			L *	92 _h	000000 ~ 002027	10 进制		
	报警器			F *	93 _h	000000 ~ 001023	10 进制		
	变址继电器			V *	94 _h	000000 ~ 001023	10 进制		
	链接继电器			B *	A0 _h	000000 ~ 0007FF	16 进制		
	数据寄存器		字	D *	A8 _h	000000 ~ 011135	10 进制		
	链接寄存器			W *	B4 _h	000000 ~ 0007FF	16 进制		
	定时器*3	触点	位	TS	C1 _h	000000 ~ 000511	10 进制		• 与输入、输出相同(用于直接访问)
		线圈		TC	C0 _h				
		当前值	字	TN	C2 _h		10 进制		
		累计定时器 *3	触点	SS	C7 _h				
	线圈		位	SC	C6 _h				
	当前值	字		SN	C8 _h				
	计数器*3	触点	位	CS	C4 _h	000000 ~ 000511	10 进制		
		线圈		CC	C3 _h				
	当前值	字	CN	C5 _h					
	链接特殊继电器		位	SB	A1 _h	000000 ~ 0003FF	16 进制		
	链接特殊寄存器		字	SW	B5 _h	000000 ~ 0003FF	16 进制		
	步进继电器*2		位	S *	98 _h	000000 ~ 002027	10 进制		
	直接输入		位	DX	A2 _h	000000 ~ 0007FF	16 进制		
	直接输出			DY	A3 _h	000000 ~ 0007FF	16 进制		
变址寄存器		字	Z *	CC _h	000000 ~ 000009	10 进制	-		
文件寄存器*4、*6			字	R *	AF _h	000000 ~ 032767	10 进制	• 用于通过块切换进行的普通访问	
				ZR	B0 _h	000000 ~ 00FFFF	16 进制	• 用于连号访问	

- *1 对 Q/L/QnACPU 内的软元件存储器进行访问。
软元件范围根据 Q/L/QnACPU 的版本而有所不同。
关于软元件范围的有关内容，请参阅所使用的 CPU 的用户手册。
通过参数设置对软元件范围进行了更改时，可以对更改后的软元件范围进行访问。
但是，不能对各程序的局部软元件进行访问。

2) 高性能型 QCPU、过程 CPU、冗余 CPU、QnACPU

分类	软元件	类型	软元件代码		软元件编号范围(默认值)		备注	
			ASCII 代码时	二进制 代码时				
内部系统软 元件	功能输入	位	-	-	000000 ~ 00000F	16 进制	• 访问禁止	
	功能输出		-	-	000000 ~ 00000F	16 进制		
	功能寄存器	字	-	-	000000 ~ 000004	10 进制	-	
	特殊继电器	位	SM	91 _H	000000 ~ 002047	10 进制		
	特殊寄存器	字	SD	A9 _H	000000 ~ 002047	10 进制		
内部用户软 元件	输入	位	X *	9C _H	000000 ~ 001FFF	16 进制	• 更改分配时，可以访问至更改后的最大 软元件编号。 • 禁止访问局部软元件。	
	输出		Y *	9D _H	000000 ~ 001FFF	16 进制		
	内部继电器*2		M *	90 _H	000000 ~ 008191	10 进制		
	锁存继电器*2		L *	92 _H	000000 ~ 008191	10 进制		
	报警器		F *	93 _H	000000 ~ 002047	10 进制		
	变址继电器		V *	94 _H	000000 ~ 002047	10 进制		
	链接继电器		B *	A0 _H	000000 ~ 001FFF	16 进制		
	数据寄存器	字	D *	A8 _H	000000 ~ 012287	10 进制		
	链接寄存器		W *	B4 _H	000000 ~ 001FFF	16 进制		
	定时器*3	触点	位	TS	C1 _H	000000 ~ 002047		10 进制
				线圈	TC			
		当前值	字	TN	C2 _H			10 进制
	累计定时器 *3	触点	位	SS	C7 _H			
				线圈	SC			
		当前值	字	SN	C8 _H			
	计数器*3	触点	位	CS	C4 _H	000000 ~ 001023		10 进制
				线圈	CC			
		当前值	字	CN	C5 _H			
	链接特殊继电器		位	SB	A1 _H	000000 ~ 0007FF		16 进制
	链接特殊寄存器		字	SW	B5 _H	000000 ~ 0007FF		16 进制
	步进继电器*2		位	S *	98 _H	000000 ~ 008191		10 进制
	直接输入		位	DX	A2 _H	000000 ~ 001FFF	16 进制	• 与输入、输出相同(用于直接访问)
	直接输出			DY	A3 _H	000000 ~ 001FFF	16 进制	
变址寄存器		字	Z *	CC _H	000000 ~ 000015	10 进制	-	
文件寄存器*4、*5、*6		字	R *	AF _H	000000 ~ 032767	10 进制	• 用于通过块切换进行的普通访问	
			ZR	B0 _H	000000 ~ 0FE7FF	16 进制	• 用于连号访问	

3) 通用型 QCPU、LCPU

分类	软元件		类型	软元件代码		软元件编号范围(默认值)		备注	
				ASCII 代码时	二进制 代码时				
内部系统软 元件	功能输入		位	-	-	000000 ~ 00000F	16 进制	• 访问禁止	
	功能输出			-	-	000000 ~ 00000F	16 进制		
	功能寄存器		字	-	-	000000 ~ 000004	10 进制	-	
	特殊继电器		位	SM	91 _h	000000 ~ 002047	10 进制		
	特殊寄存器		字	SD	A9 _h	000000 ~ 002047	10 进制		
内部用户软 元件	输入		位	X *	9C _h	000000 ~ 001FFF	16 进制	• 更改分配时，可以访问至更改后的最大 软元件编号。 • 禁止访问局部软元件。	
	输出			Y *	9D _h	000000 ~ 001FFF	16 进制		
	内部继电器*2			M *	90 _h	000000 ~ 008191	10 进制		
	锁存继电器*2			L *	92 _h	000000 ~ 008191	10 进制		
	报警器			F *	93 _h	000000 ~ 002047	10 进制		
	变址继电器			V *	94 _h	000000 ~ 002047	10 进制		
	链接继电器			B *	A0 _h	000000 ~ 001FFF	16 进制		
	数据寄存器		字	D *	A8 _h	000000 ~ 012287	10 进制		
	链接寄存器			W *	B4 _h	000000 ~ 001FFF	16 进制		
	定时器*3	触点	位	TS	C1 _h	000000 ~ 002047	10 进制		
				TC	C0 _h				
		当前值	字	TN	C2 _h		10 进制		
	累计定时器 *3	触点	位	SS	C7 _h				
				SC	C6 _h				
		当前值	字	SN	C8 _h				
	计数器*3	触点	位	CS	C4 _h	000000 ~ 001023	10 进制		
				CC	C3 _h				
		当前值	字	CN	C5 _h				
	链接特殊继电器		位	SB	A1 _h	000000 ~ 0007FF	16 进制		-----
	链接特殊寄存器		字	SW	B5 _h	000000 ~ 0007FF	16 进制		
	步进继电器*2		位	S *	98 _h	000000 ~ 008191	10 进制	• 访问禁止。	
	直接输入		位	DX	A2 _h	000000 ~ 001FFF	16 进制	• 与输入、输出相同(直接访问用)	
	直接输出			DY	A3 _h	000000 ~ 001FFF	16 进制		
变址寄存器			字	Z *	CC _h	000000 ~ 000015 (000000 ~ 000019)	10 进制	-	
文件寄存器*4、*5、*6			字	R *	AF _h	000000 ~ 032767	10 进制	• 用于通过块切换进行的普通访问	
				ZR	B0 _h	000000 ~ 0FE7FF (000000 ~ 393216)	16 进制 (10 进制)	• 用于连号访问 • 括弧内为 LCPUCPU 的情况下。	
扩展数据寄存器*7、*8			字	D *	A8 _h	• 二进制代码: 000000 ~ 4184063 • ASCII 代码:: 000000 ~ 999999	10 进制	• 在可编程控制器文件设置中对使用点数 进行了设置的情况下，可以访问至设置 后的最大软元件编号。 但是，ASCII 代码时，左述点数将成为上 限。	
扩展链接寄存器*7			字	W *	B4 _h	000000 ~ 3FD7FF	16 进制	• 在可编程控制器文件设置中 对使用点数 进行了设置的情况下，可以访问至设置 后的最大软元件编号。	

4) 安全 CPU

分类	软元件	类型	软元件代码		软元件编号范围(默认值)		备注	
			ASCII 代码时	二进制 代码时				
内部系统软 元件	特殊继电器	位	SM	91 _H	000000 ~ 005119	10 进制	-	
	特殊寄存器	字	SD	A9 _H	000000 ~ 005119	10 进制		
内部用户软 元件	输入	位	X *	9C _H	000000 ~ 0017FF	16 进制	• 更改分配时，可以访问至更改后的最大 软元件编号。	
	输出		Y *	9D _H	000000 ~ 0017FF	16 进制		
	内部继电器		M *	90 _H	000000 ~ 006143	10 进制		
	报警器		F *	93 _H	000000 ~ 001023	10 进制		
	变址继电器		V *	94 _H	000000 ~ 001023	10 进制		
	链接继电器		B *	A0 _H	000000 ~ 0007FF	16 进制		
	数据寄存器	字	D *	A8 _H	000000 ~ 006143	10 进制		
	链接寄存器		W *	B4 _H	000000 ~ 0007FF	16 进制		
	定时器	触点	位	TS	C1 _H	000000 ~ 000511		10 进制
		线圈		TC	C0 _H			
	累计定时器	当前值	字	TN	C2 _H			10 进制
		触点	位	SS	C7 _H			
		线圈		SC	C6 _H			
		当前值	字	SN	C8 _H			
	计数器	触点	位	CS	C4 _H	000000 ~ 000511		10 进制
		线圈		CC	C3 _H			
		当前值	字	CN	C5 _H			
	链接特殊继电器	位	SB	A1 _H	000000 ~ 0005FF	16 进制		
	链接特殊寄存器	字	SW	B5 _H	000000 ~ 0005FF	16 进制		

- *2 Q/L/QnACPU 的情况下，内部用户的内部继电器(M)、锁存继电器(L)、步进继电器(S)为其它的软元件。
- *3 随机读取中，可指定定时器、累计定时器、计数器的各触点以及线圈的 Q/L/QnACPU 如下表所示。

功能	指令	Q/LCPU	QnACPU	
		功能版本	功能版本	
		A 以后	A	B 以后
随机读取	0403	○	×	○
监视数据登录	0801	×		

○：可以指定 ×：不能指定
对不能进行指定的 QnACPU 指定了定时器、累计定时器、计数器的各触点以及线圈时，将返回 4032_H 的出错信息。

- *4 从外部设备对可编程控制器 CPU 的文件寄存器进行访问时，应根据可编程控制器 CPU 的文件寄存器的结构，指定以下软元件代码进行访问。
- 1) 文件寄存器由多个块构成时
 - 指定用于连号访问的软元件代码时，通过 ASCII 代码进行通信的情况下应指定“ZR”；通过二进制代码进行通信的情况下应指定“B0H”。
 - 2) 仅由块 0 构成文件寄存器时
 - 指定用于连号访问或普通访问的软元件代码时，通过 ASCII 代码进行通信的情况下应指定“ZR” / “R*”，通过二进制代码进行通信的情况下应指定“B0H” / “AFH”。
 - 通过指定普通访问用的代码“R*”、“AFH”，可以将软元件号以 10 进制数进行指定。
 - 通过连号访问进行访问时，将软元件号以 16 进制数进行指定。
- 关于用于连号访问以及普通访问的文件寄存器的软元件号的思路、访问时的限制有关内容，请参阅各 CPU 模块的用户手册以及编程手册。

备注

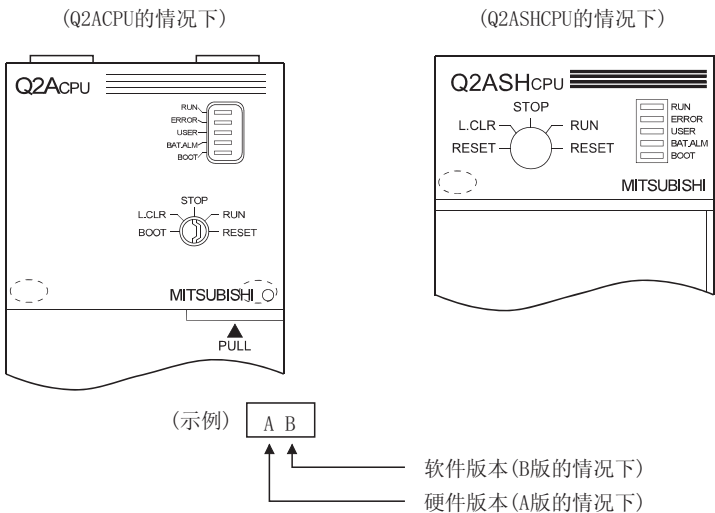
用于连号访问的软元件号的示例(QCPU 的情况下)

ZR000000 (0)	文件寄存器	对于连号访问用的软元件号，从实际存在的块的小块号的软元件开始按编号顺序自动分配。
~	块编号0的区域	
ZR032767 (32767)	(R0~R32767)	
ZR032768 (32768)	文件寄存器	
~	块编号1的区域	
ZR065535 (65535)	(R0~R32767)	
ZR065536 (65536)	文件寄存器	
~	块编号2的区域	

- *5 只有在以下的限制被全部清除时才可以对 QnACPU 的 EEPROM 中设置的文件寄存器进行数据写入。
- 以下限制未被全部清除时，向文件寄存器进行数据写入时将会返回异常结束报文。
- (用于对 EEPROM 的文件寄存器进行数据写入的限制)
- 只有通过批量写入功能(指令：1401)才可以进行写入。
 - 对象 QnACPU 的运行状态处于 STOP 中/PAUSE 中时可以进行写入。
 - 对生产日期为 1999 年 1 月以后的 QnACPU 及下述 QnACPU 可以进行写入。

QnACPU 的类型	生产日期	软件版本
Q2ACPU(-S1)、Q3ACPU、Q4ACPU	1998 年 9 月～12 月	L 版以后产品
Q4ARCPU		S 版以后产品
Q2ASCPU(-S1)、Q2ASHCPU(-S1)		T 版以后产品

- 关于除上述以外的 QnACPU 的有关内容，请参阅对象 QnACPU 的手册。
- * 模块的生产年(公历(低 2 位)、月(2 位))可以在模块侧面的“额定铭牌的 DATE 栏”中进行确认。
- 模块的软件版本可通过模块正面的封条(位于下述虚线圈部分的某个位置。)进行确认。



- *6 不能从外部设备访问通过参数指定了与程序相同文件名的各程序的文件寄存器。
- *7 不支持对 D65536 以后的扩展数据寄存器以及 W10000 以后的扩展链接寄存器的访问的情况下，应指定同一区域的文件寄存器(ZR)。
- 关于指定的文件寄存器(ZR)的计算方法，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。
- *8 在 Q50UDEHCPU、Q100UDEHCPU 中，变为 128k 点。

(b) Q/L/QnACPU 以外的情况下*9

软元件		类型	软元件代码		软元件编号范围(默认值)			备注
			ASCII 代码时	二进制代码时	QnA 兼容 3C/4C 帧	QnA 兼容 3E 帧、4E 帧	表示	
输入		位	X *	9C _h	000000 ~ 001FFF	000000 ~ 0007FF	16 进制	• 禁止扩展指定。 • 指定访问目标中存在的软元件、软元件范围进行访问。
输出			Y *	9D _h	000000 ~ 001FFF	000000 ~ 0007FF	16 进制	
内部继电器			M *	90 _h	000000 ~ 008191	000000 ~ 008191	10 进制	
锁存继电器			L *	92 _h	000000 ~ 008191	000000 ~ 008191	10 进制	
步进继电器			S *	98 _h	000000 ~ 008191	000000 ~ 008191	10 进制	
链接继电器			B *	A0 _h	000000 ~ 001FFF	000000 ~ 000FFF	16 进制	
报警器			F *	93 _h	000000 ~ 002047	000000 ~ 002047	10 进制	
特殊继电器			M *	90 _h	009000 ~ 009255	009000 ~ 009255	10 进制	
定时器 T	触点	位	TS	C1 _h	000000 ~ 002047	000000 ~ 002047	10 进制	
	线圈		TC	C0 _h				
	当前值	字	TN	C2 _h				
计数器 C	触点	位	CS	C4 _h	000000 ~ 001023	000000 ~ 001023	10 进制	
	线圈		CC	C3 _h				
	当前值	字	CN	C5 _h				
数据寄存器		字	D *	A8 _h	000000 ~ 008191	000000 ~ 006143	10 进制	
链接寄存器			W *	B4 _h	000000 ~ 001FFF	000000 ~ 000FFF	16 进制	
文件寄存器*10		字	R *	AF _h	000000 ~ 008191	000000 ~ 008191	10 进制	• 用于通过块切换进行的普通访问
			ZR	B0 _h	000000 ~ 07FFFF	000000 ~ 07FFFF	16 进制	• 用于连号访问
特殊寄存器		字	D *	A8 _h	009000 ~ 009255	009000 ~ 009255	10 进制	-

*9 对指定 CPU 内的软元件存储器进行访问。

对 Q/L/QnACPU 以外的可编程控制器 CPU 进行访问时，应注意以下内容。

- 1) 应在访问目标可编程控制器 CPU 中可使用的软元件编号范围内进行访问。
- 2) 对象设备连接站 Q/L/QnACPU、CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 经由 Q/L/QnACPU 除外，对可编程控制器 CPU 以字单位进行访问时，位软元件编号必须置为 16 的倍数(10 进制数的情况下 0、16、...)。
此外，对于特殊继电器 M 的 M9000 以后，也可以 (9000+16 的倍数) 进行指定。
- 3) 以 QnA 兼容 3C/4C 帧进行访问的情况下，M、L、S 被进行了范围指定，但将 M 的编号范围通过 L、S 进行指定，或者与之相反的情况下其处理也相同。
以 QnA 兼容 3E 帧、4E 帧进行访问的情况下，对于 L、S 应指定 M 后进行访问。
- 4) 特殊继电器(M9000~M9255)、特殊寄存器(D9000~D9255)分为只读、写入专用、系统用。
超出写入允许范围进行了写入时有可能导致可编程控制器 CPU 出错。
关于特殊继电器、特殊寄存器的详细内容请参阅 ACPU 的编程手册。

*10 请参阅(a)*4。

(C) CPU 以外的情况下

1) CC-Link IE 现场网络起始模块

软元件	类型	软元件代码		软元件编号范围		备注
		ASCII 代码时	二进制 代码时			
特殊继电器	位	SM	91 _H	000000 ~ 002047	10 进制	- • 分配更改禁止。
特殊寄存器	字	SD	A9 _H	000000 ~ 002047	10 进制	
输入	位	X *	9C _H	000000 ~ 008191	16 进制	
输出		Y *	9D _H	000000 ~ 008191	16 进制	
链接寄存器		W *	B4 _H	000000 ~ 008191	16 进制	
链接特殊继电器	位	SB	A1 _H	000000 ~ 008191	16 进制	
链接特殊寄存器	字	SW	B5 _H	000000 ~ 008191	16 进制	

2) MELSECNET/H 远程 I/O 站

软元件	类型	软元件代码		软元件编号范围		备注
		ASCII 代码时	二进制 代码时			
特殊继电器	位	SM	91 _H	000000 ~ 002047	10 进制	- • 分配更改禁止。
特殊寄存器	字	SD	A9 _H	000000 ~ 002047	10 进制	
输入	位	X *	9C _H	000000 ~ 001FFF	16 进制	
输出		Y *	9D _H	000000 ~ 001FFF	16 进制	
内部继电器		M *	90 _H	000000 ~ 008191	10 进制	
链接继电器		B *	A0 _H	000000 ~ 003FFF	16 进制	
数据寄存器	字	D *	A8 _H	000000 ~ 012287	10 进制	
链接寄存器		W *	B4 _H	000000 ~ 003FFF	16 进制	
链接特殊继电器	位	SB	A1 _H	000000 ~ 0001FF	16 进制	
链接特殊寄存器	字	SW	B5 _H	000000 ~ 0001FF	16 进制	

3.3.2 多个块批量读取、批量写入

将n点的位软元件存储器(1点=16位)或字软元件存储器(1点=1字)作为1个块,对随机指定多个块进行读取、写入的控制步骤进行举例说明。

本功能用于对 Q/L/QnACPU 进行访问。

可以对以下模块进行访问。

- C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络, CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)
- 下述 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 QCPU/下述 QnACPU(其它站)

功能	可编程控制器 CPU		
	QnA	Q2AS (H)	Q4AR
多个块的批量读取/写入	(9707B 以后产品)*		(均可以)

* 包装标识及本体的额定铭牌的 DATE 栏中记载有 9707 B 以后的为新增了本功能的产品。

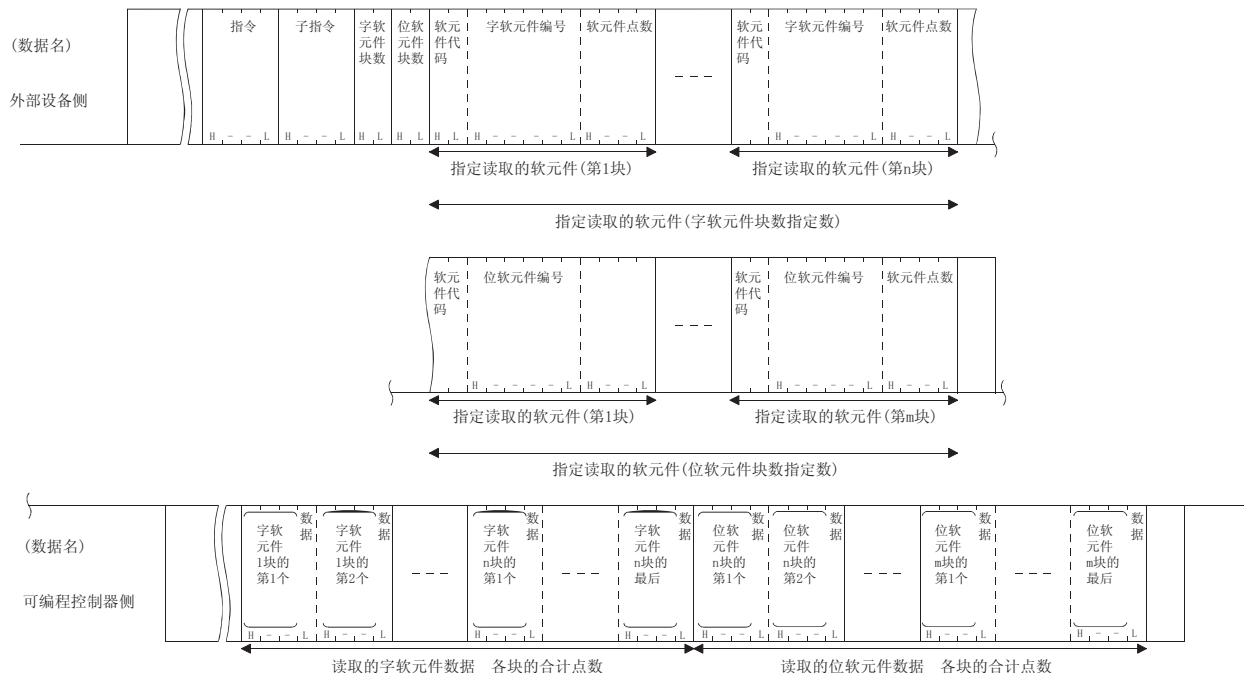
生产年月 功能版本(仅 B 版以后记载)

要点	对 T、ST、C 的触点及线圈进行读取/写入的情况下，应使用位软元件块数。
----	---------------------------------------

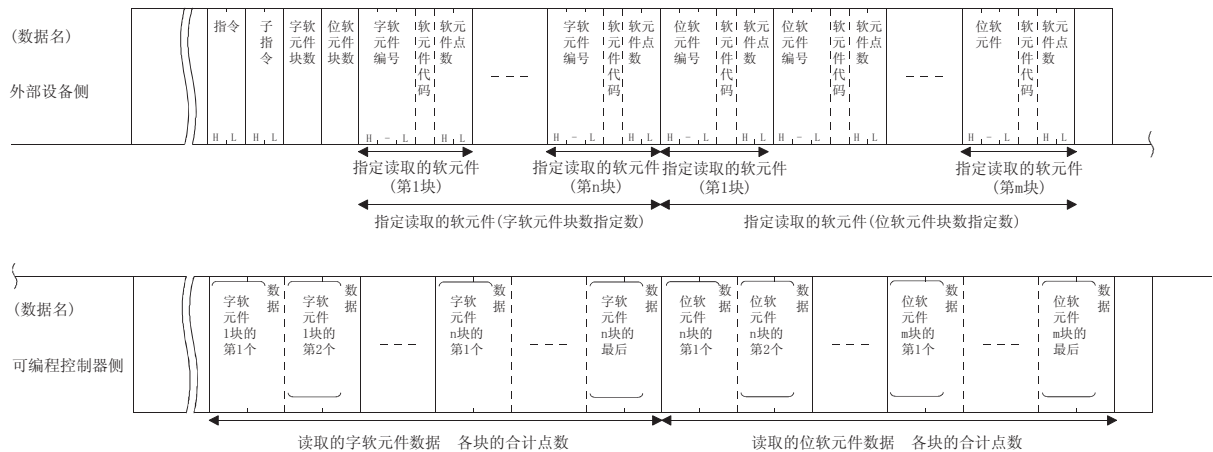
(1) 多个块批量读取时的字符部分的数据排列

以下对多个块批量读取时的字符部分的数据排列有关内容进行说明。

(a) 通过 ASCII 代码进行通信时的数据排列



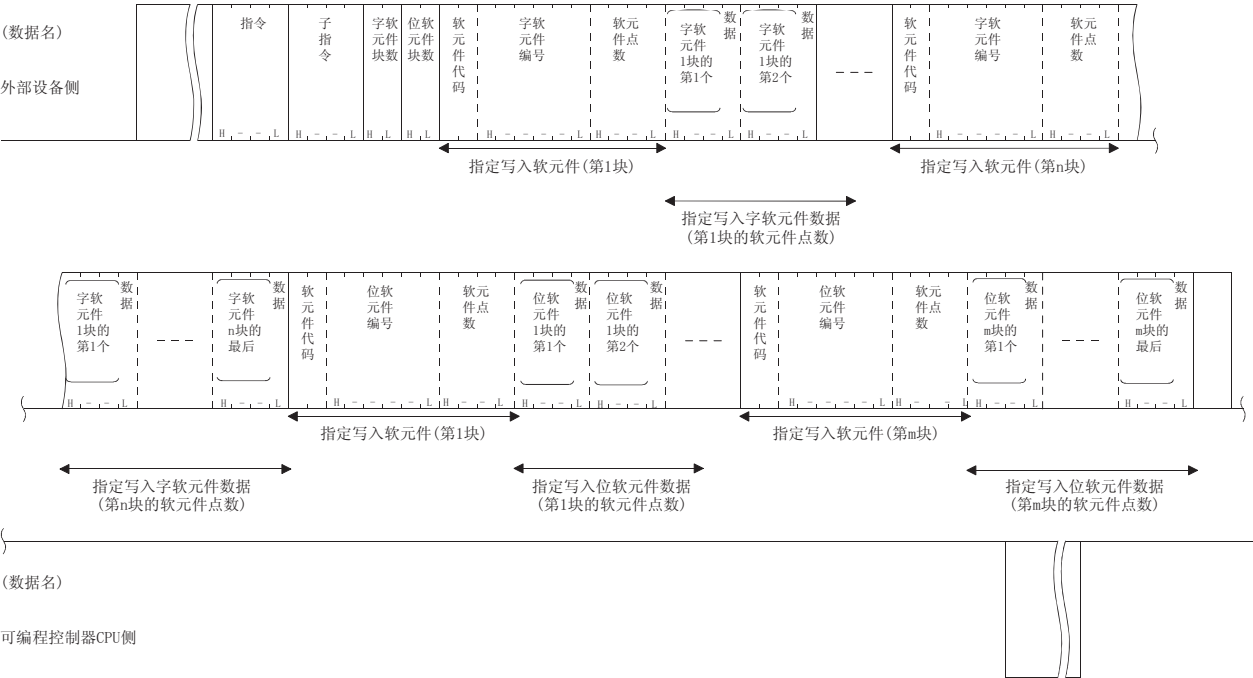
(b) 通过二进制代码进行通信时的数据排列



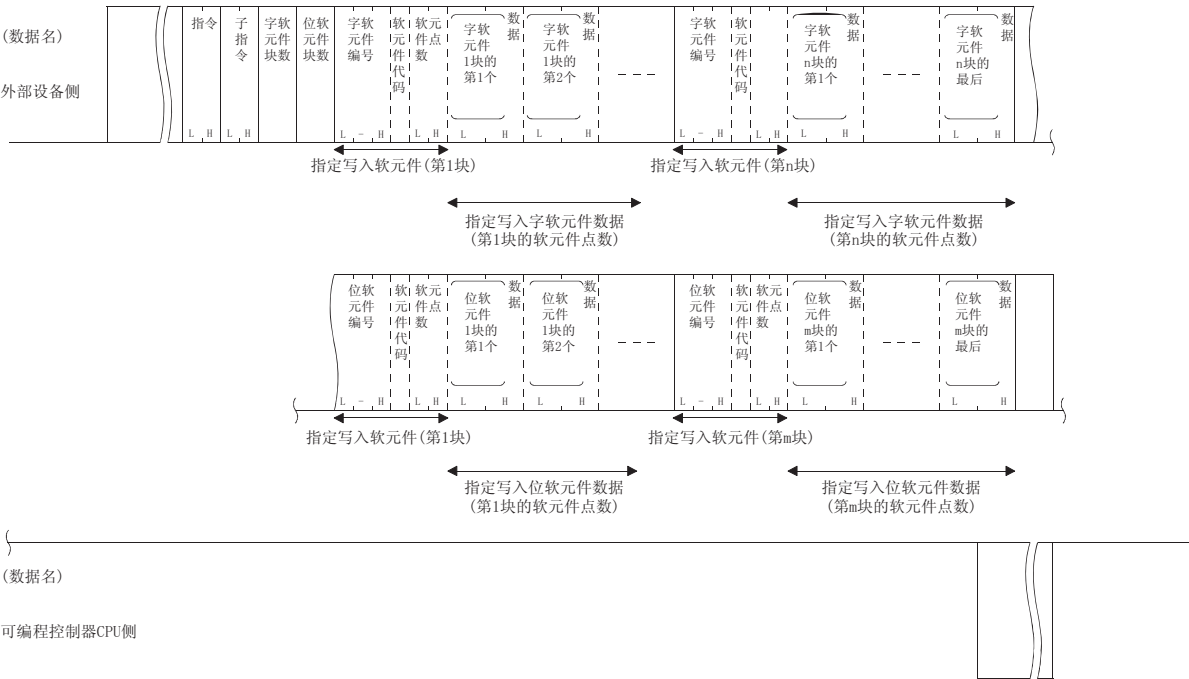
(2) 多个块批量写入时的字符部分的数据排列

以下对多个块批量写入时的字符部分的数据排列有关内容进行说明。

(a) 通过 ASCII 代码进行通信时的数据排列



(b) 通过二进制代码进行通信时的数据排列



(3) 多个块批量读取、批量写入时的字符部分的内容

以下对多个块批量读取、批量写入时的字符部分的内容进行说明。

对于如下所示的数据以外，与使用其它指令时的内容相同。

(a) 字软元件块数、位软元件块数

是用于在对字软元件及位软元件进行批量读取、批量写入中，对之后传送的字软元件的块数、位软元件的块数分别进行指定的数据。

1) 通过 ASCII 代码进行数据通信时

将各块数转换为 ASCII 代码 2 位 (16 进制数) 后进行发送。

(示例)

5 块的情况下 : 变为 “05”，从 “0” 开始按顺序进行发送。

20 块的情况下 : 变为 “14”，从 “1” 开始按顺序进行发送。

2) 通过二进制代码进行数据通信时

对表示各块数的 1 字节的数值进行发送。

(示例)

5 块的情况下 : 发送 05_H。

20 块的情况下 : 发送 14_H。

3) 对各块数按以下方式进行指定。

120 ≡ 字软元件块数 + 位软元件块数

4) 将某个块数置为 0 点时，不需要指定相应的软元件编号 · 软元件代码 · 软元件点数 · 数据。

(b) 字软元件编号、位软元件编号

是用于将字软元件及位软元件的连续软元件作为一个块，对批量读取、批量写入的各块的起始字软元件及位软元件分别进行指定的数据。

1) 通过 ASCII 代码进行数据通信时

将各块的起始软元件编号转换为 ASCII 代码 6 位后进行发送。

(示例)

内部继电器 M1234、链接寄存器 W1234 的情况下

内部继电器 M1234 时为 “001234” 或 “_□ _□ 1234”，链接寄存器

W1234 时为 “001234” 或 “_□ _□ 1234”，均从 “0” 或 “_□” 开始按顺序进行发送。

2) 通过二进制代码进行数据通信时

将各块的起始软元件编号以 3 字节的数值表示并进行发送。

(示例)

内部继电器 M1234、链接寄存器 W1234 的情况下

内部继电器 M1234 时为 0004D2_H，按 D2_H、04_H、00_H 的顺序进行发

送。链接寄存器 W1234 时为 001234_H，按 34_H、12_H、00_H 的顺序进行发送。

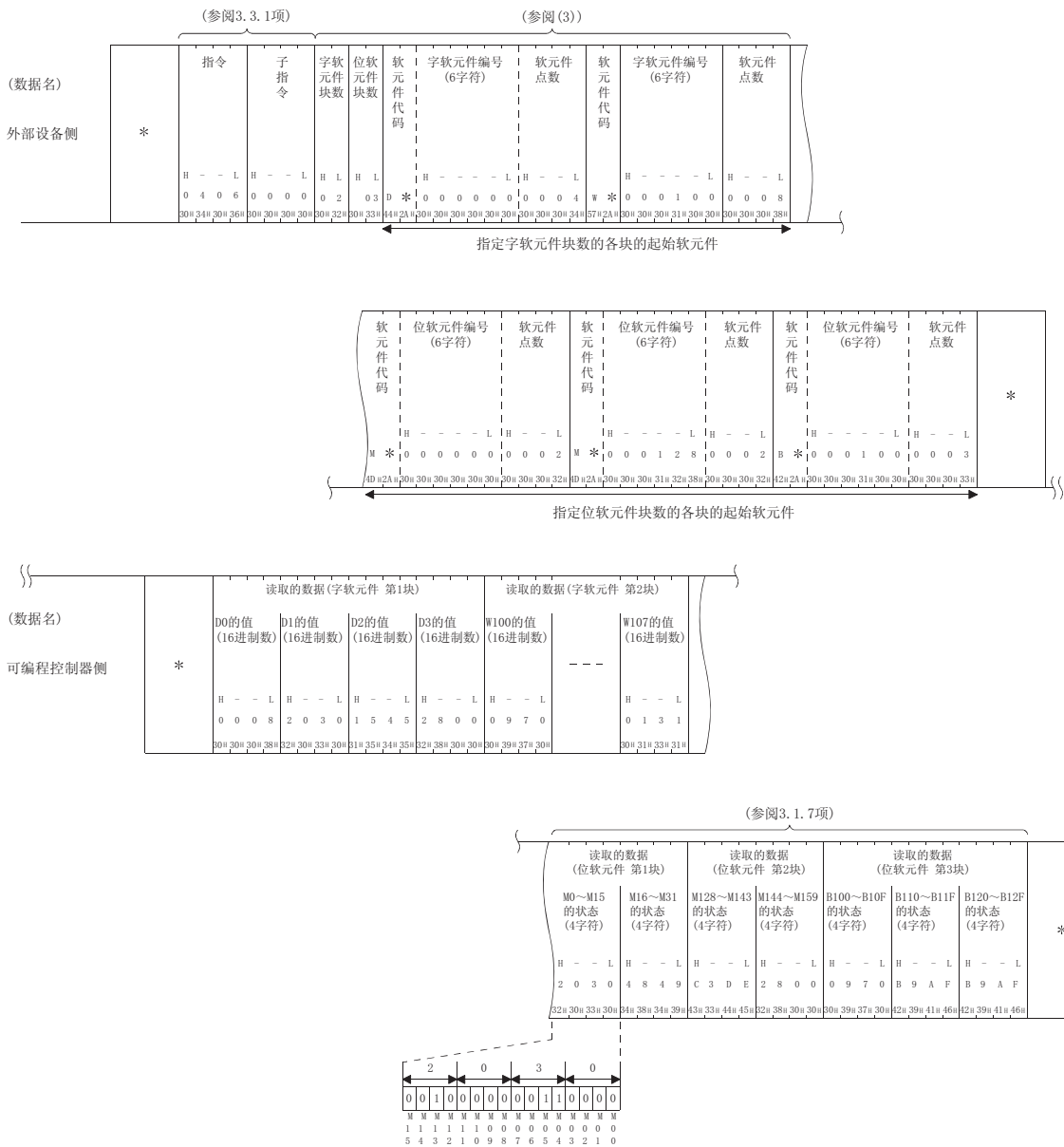
- (c) 软元件代码
是用于识别进行批量读取、批量写入的各块的起始软元件存储器的数据。
关于各软元件的软元件代码，在 3.3.1 项(3)中介绍。
- 1) 通过 ASCII 代码进行数据通信时
将各软元件代码转换为 ASCII 代码 2 位(16 进制数)后进行发送。
(示例)
内部继电器(M)、链接寄存器(W)的情况下
内部继电器(M)时为“M*”，链接寄存器(W)时为“W*”，从“M”、“W”开始按顺序进行发送。
- 2) 通过二进制代码进行数据通信时
对表示各软元件代码的 1 字节的数值进行发送。
(示例)
内部继电器(M)、链接寄存器(W)的情况下
内部继电器(M)发送 90H，链接寄存器(W)时发送 B4h。
- (d) 软元件点数
是用于将字软元件及位软元件的连续软元件作为一个块，对进行批量读取、批量写入的各块的连续软元件范围的点数(位软元件存储器时 1 点=16 位，字软元件存储器时 1 点=1 字)分别进行指定的数据。
- 1) 通过 ASCII 代码进行数据通信时
将各块的点数转换为 ASCII 代码 4 位(16 进制数)后进行发送。
(示例)
5 点的情况下：变为“0005”，从“0”开始按顺序进行发送。
20 点的情况下：变为“0014”，从“0”开始按顺序进行发送。
- 2) 通过二进制代码进行数据通信时
对表示各块的点数的 2 字节的数值进行发送。
(示例)
5 点的情况下：变为 0005h，从 05h 开始进行发送。
20 点的情况下：变为 0014h，从 14h 开始进行发送。
- 3) 对各软元件点数按以下条件进行指定。
- 多个块批量读取时
 $960 \geq \text{字软元件各块的合计点数} + \text{位软元件各块的合计点数}$
 - 多个块批量写入时
 $960 \geq 4 \times (\text{字软元件的块数} + \text{位软元件的块数}) + \text{字软元件各块的合计点数} + \text{位软元件各块的合计点数}$

要点	
对于通过多个块批量读取、批量写入功能进行读取、写入的软元件存储器，可以进行扩展指定。 对软元件存储器进行扩展指定时应根据附录 1 的说明进行操作。	

(4) 多个块批量读取(指令：0406)

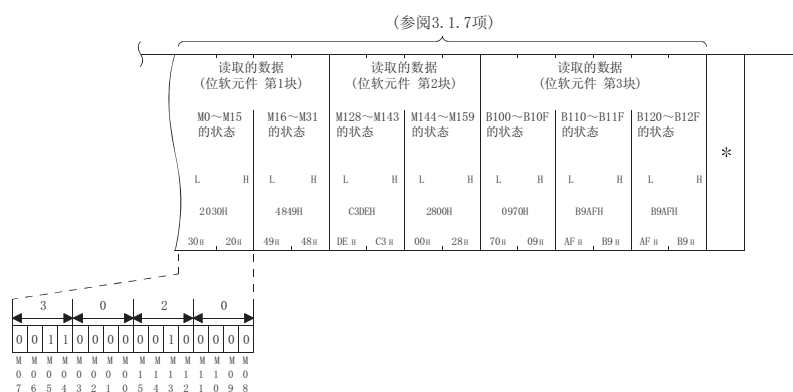
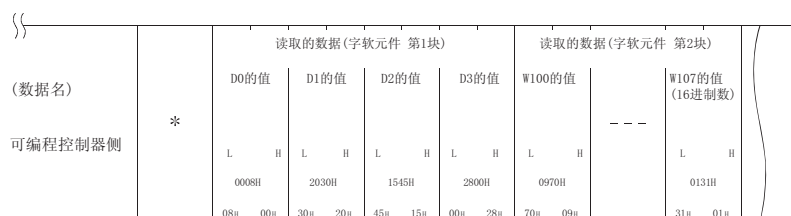
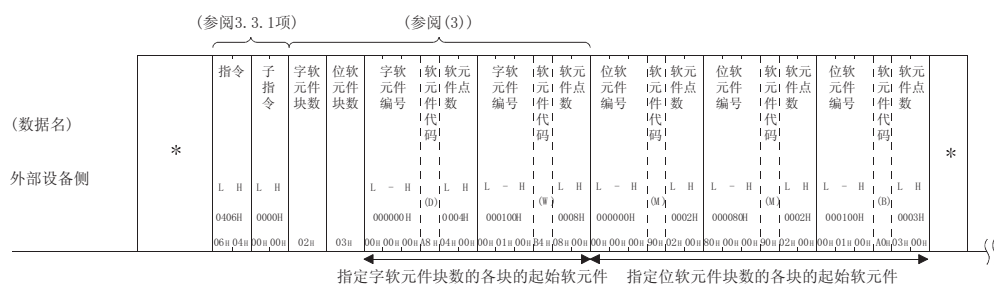
以下将连续的位软元件存储器(1 点=16 位)及字软元件存储器的 n 点作为 1 个块，对随机指定多个块进行读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

- (a) 通过由 ASCII 代码进行的通信，对以下的软元件存储器进行读取的情况下
- 字软元件存储器：2 块 D0 ～ D3(4 点)、
W100 ～ W107(8 点)
 - 位软元件存储器：3 块 M0 ～ M31(2 点)、
M128 ～ M159(2 点)、
B100 ～ B12F(3 点)



(b) 在通过二进制代码进行的通信中, 对以下的软件存储器进行读取的情况下

- 字软件存储器 : 2 块 D0 ~ D3(4 点)、
W100 ~ W107(8 点)
- 位软件存储器 : 3 块 M0 ~ M31(2 点)、
M128 ~ M159(2 点)、
B100 ~ B12F(3 点)



要点

- (1) 应按以下条件进行块数指定。
 $120 \cong \text{字软元件块数} + \text{位软元件块数}$
- (2) 应按以下条件对各软元件点数进行指定。
 $960 \cong \text{字软元件各块的合计点数} + \text{位软元件各块的合计点数}$

(5) 多个块批量写入(指令：1406)

以下将连续的位软元件存储器(1点=16位)及字软元件存储器的n点作为1个块，对随机指定多个块进行写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

- (a) 通过由 ASCII 代码进行的通信，对以下软元件存储器进行写入的情况下
- 字软元件存储器：2 块 D0 ～ D3(4 点)、
W100 ～ W107(8 点)
 - 位软元件存储器：3 块 M0 ～ M31(2 点)、
M128 ～ M159(2 点)、
B100 ～ B12F(3 点)



(b) 在通过二进制代码进行的通信中, 写入到以下软件存储器中的情况下

- 字软件存储器 : 2 块 D0 ~ D3(4 点)、
W100 ~ W107(8 点)
- 位软件存储器 : 3 块 M0 ~ M31(2 点)、
M128 ~ M159(2 点)、
B100 ~ B12F(3 点)

		(参阅 3.3.1项)				(参阅 (3))														
		指令		子指令	字软元件块数	位软元件块数	(字软元件 第1块)				(字软元件 第2块)									
							起始软元件		写入数据		起始软元件		写入数据							
(数据名)							字软元件编号	1软元ID0的值	写入至ID3的值	写入至ID3的值	字软元件编号	1软元ID0的值	写入至ID7的值	写入至ID7的值						
外部设备侧							1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)	1件数 (16进制数)						
							代码	---	---	---	代码	---	---	---						
		L	H	L	H		L	-	H	L	H	L	-	H	L	H				
		1406H		0000H			(D) 000000H		0004H	0008H		2800H	(D) 000100H	0008H	0970H		0131H			
		06n	14n	00n	00n	02n	03n	00n	00n	08n	00n	00n	28n	00n	01n	00n	08n	09n	31n	01n

(参阅3.1.7项)

(位软元件 第1块)					(位软元件 第2块)					(位软元件 第3块)					*
起始软元件		写入数据			起始软元件		写入数据			起始软元件		写入数据			
位软元 件编号	1软元 1元1点	1软元 1元1点	写入至 M0~M15 的值	写入至 M16~M31 的值	位软元 件编号	1软元 1元1点	1软元 1元1点	写入至 M128~M143 的值	写入至 M144的~ M159的值	位软元 件编号	1软元 1元1点	1软元 1元1点	写入至 B100~B10F 1的值	写入至 B120~B12F 1的值	
L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H
00000H	10001H	0002H	2030H	4849H	000080H	10001H	0002H	C3DEH	2800H	000100H	10001H	0003H	0970H	B9AFH	B9H
00H 00H 00H 00H 02H 00H	30H	20H	49H	48H	80H 00H 00H 00H 02H 00H	DEH	C3H	00H	28H	00H 01H 00H 00H 03H 00H	70H	09H	AFH	B9H	

3020

0011000001000000

HHHHHHHHHHHHHHHH

7 6 5 4 3 2 1 0 5 4 3 2 1 0 9 8

(数据名) 可编程控制器側	*	
----------------------	---	--

要点

应按以下条件对各软元件点数进行指定。

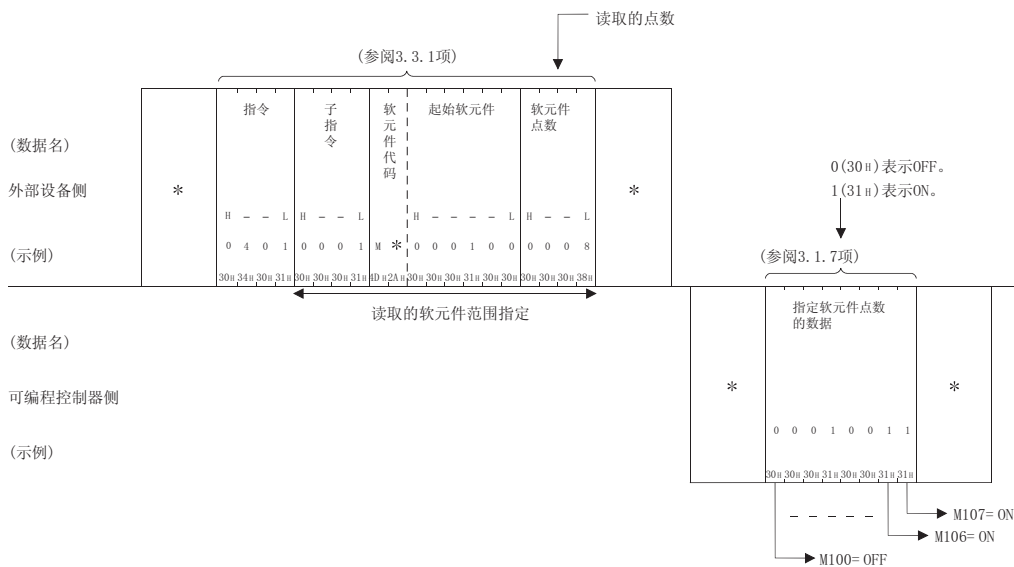
$$960 \geq 4 \times (\text{字软件元件块数} + \text{位软件元件块数}) + \text{字软件元件各块的合计点数} + \text{位软件元件各块的合计点数}$$

3.3.3 位单位的批量读取(指令：0401)

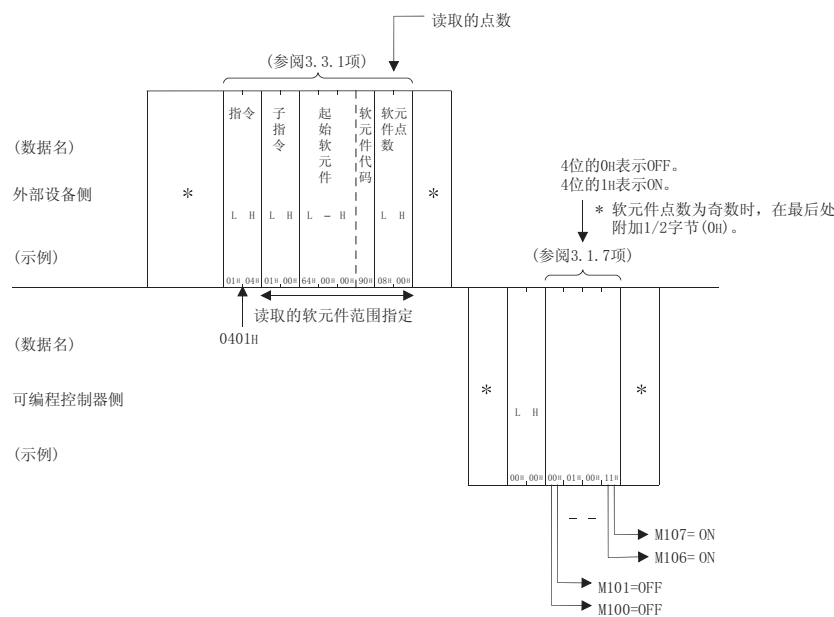
以下对位软元件存储器的批量读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对内部继电器 M100～M107 的 8 点进行读取的情况下



(2) 在通过二进制代码进行的通信中，对内部继电器 M100～M107 的 8 点进行读取的情况下



要点

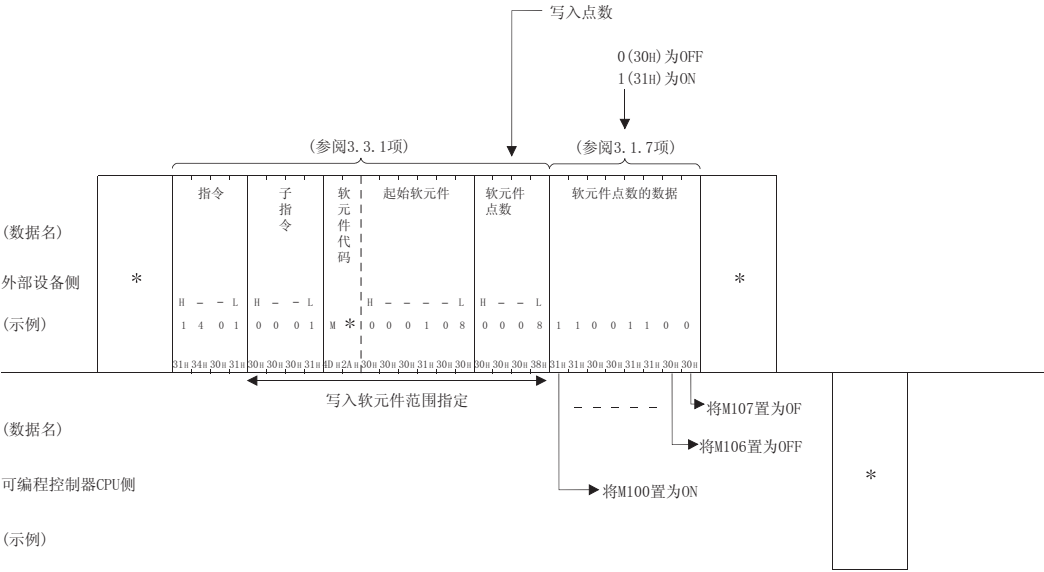
- (1) 软元件点数应在以下范围内进行指定。
- (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时
- 软元件点数 : $1 \leq \text{软元件点数} \leq 3584、7168 \text{ 或 } 7904$
- 1) 经由 E71 时 : 通过 ASCII 代码进行通信时 : 3584 点
通过二进制代码进行通信时 : 7168 点
- 2) 经由 C24 时 : 7904 点
- (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时
- 软元件点数 : $1 \leq \text{软元件点数} \leq 1792、3584 \text{ 或 } 3952$
- 1) E71 经由时 : 通过 ASCII 代码进行通信时 : 1792 点
通过二进制代码进行通信时 : 3584 点
- 2) C24 经由时 : 3952 点
- (c) 对除上述以外的可编程控制器 CPU(其它站)进行访问时
- 软元件点数 : $1 \leq \text{软元件点数} \leq 256$
- (2) 访问范围应在以下范围内。
- 访问范围 : $(\text{起始软元件号} + \text{软元件点数} - 1) \leq \text{最大软元件号}$
- (3) 通过以太网模块对 ACPU 进行访问时，如果将软元件点数指定为奇数，响应数据中将被附加虚拟数据。
- 通过 ASCII 代码进行通信时，在最后处将被附加 1 字节(30_H)。
 - 通过二进制代码进行通信时，在最后处将被附加 1/2 字节(0_H)。
- (示例) 通过 ASCII 代码进行通信时，读取 3 点时将返回 4 点的数据。最后的 1 字节为虚拟数据。

3.3.4 位单位的批量写入(指令：1401)

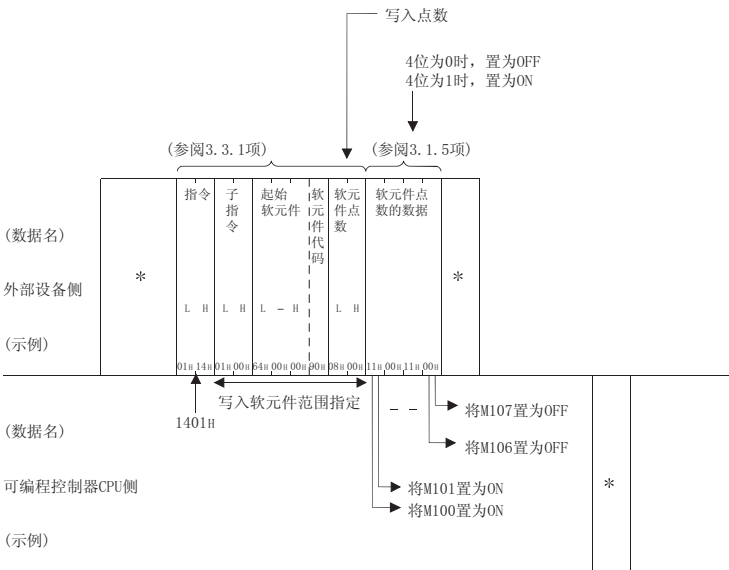
以下对至位软元件存储器的批量写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对内部继电器 M100~M107 进行 8 点的写入的情况下



(2) 在通过二进制代码进行的通信中，对内部继电器 M100～M107 写入 8 点的情况下



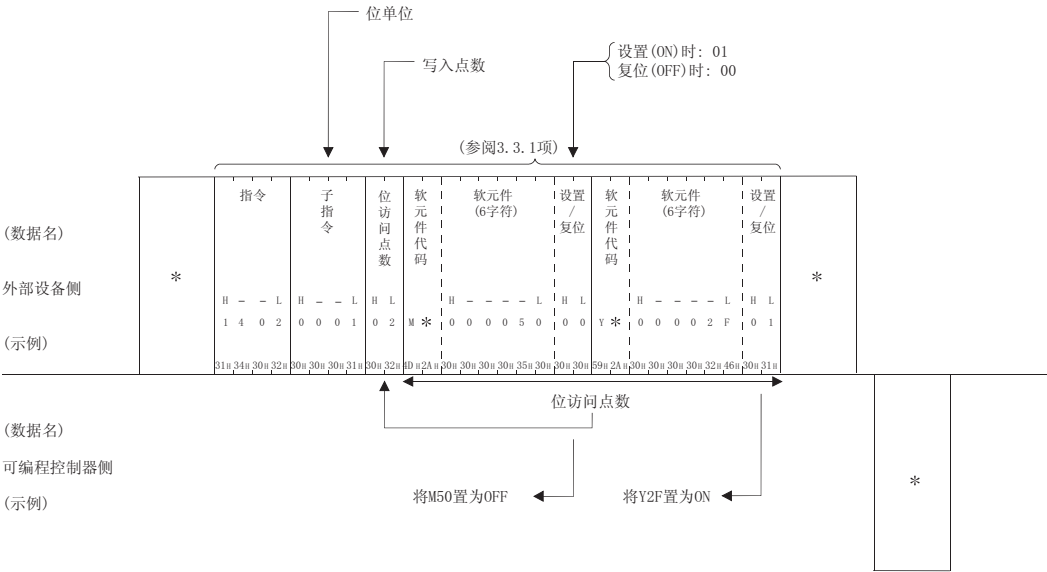
要点	
(1)	应在以下范围内对软元件点数进行指定。
(a)	对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 软元件点数 : 1 ≤ 软元件点数 ≤ 3584、7168 或 7904 1) 经由 E71 时 : 通过 ASCII 代码进行通信时 : 3584 点 通过二进制代码进行通信时 : 7168 点 2) 经由 C24 时 : 7904 点
(b)	QnACPU(其它站), QnA 系列对应网络系统(MELSECNET/10、以太网)经由 Q/QnACPU(其它站)访问时 软元件点数 : 1 ≤ 软元件点数 ≤ 1792、3584 或 3952 1) 经由 E71 时 : 通过 ASCII 代码进行通信时 : 1792 点 通过二进制代码进行通信时 : 3584 点 2) 经由 C24 时 : 3952 点
(c)	对除上述以外的可编程控制器 CPU(其它站)进行访问时 软元件点数 : 1 ≤ 软元件点数 ≤ 160
(2)	访问范围应在以下范围内。 访问范围 : (起始软元件号+软元件点数-1) ≤ 最大软元件号
(3)	对 Q/L/QnACPU 进行写入时, 如果 Q/L/QnACPU 被实施了系统保护将变为出错状态, 将返回出错时的结束代码。

3.3.5 位单位的随机写入(测试)(指令：1402)

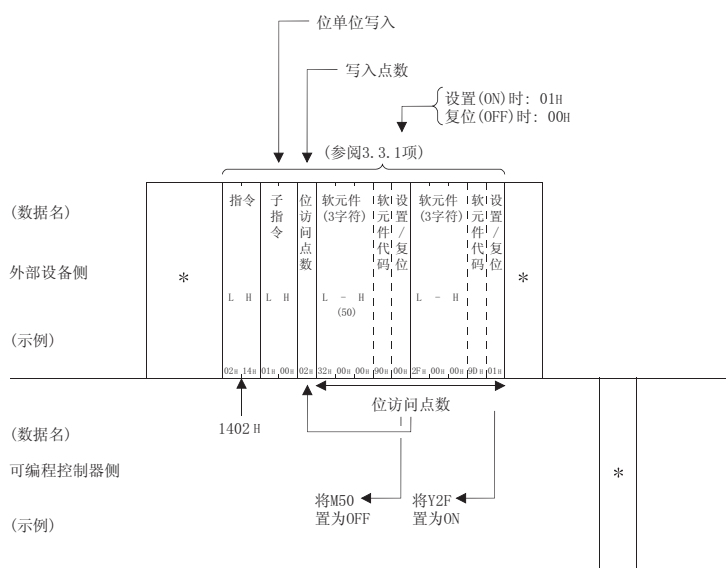
以下对随机指定位软元件存储器进行数据写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (1) 通过由 ASCII 代码进行的通信，将内部继电器 M50 置为 OFF，将输出继电器 Y2F 置为 ON 的情况下



(2) 在通过二进制代码进行的通信中, 将内部继电器 M50 置为 OFF, 将输出继电器 Y2F 置为 ON 的情况下



要点

- (1) 应将访问点数在以下范围内进行指定。
 - (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q/L 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时
软元件点数 : $1 \leq \text{软元件点数} \leq 188$
 - (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时
软元件点数 : $1 \leq \text{软元件点数} \leq 94$
 - (c) 对除上述以外的可编程控制器 CPU(其它站)进行访问时
软元件点数 : $1 \leq \text{软元件点数} \leq 20$
- (2) 对 Q/L/QnACPU 进行写入时, 如果 Q/L/QnACPU 被实施了系统保护将变为出错状态, 将返回出错时的结束代码。

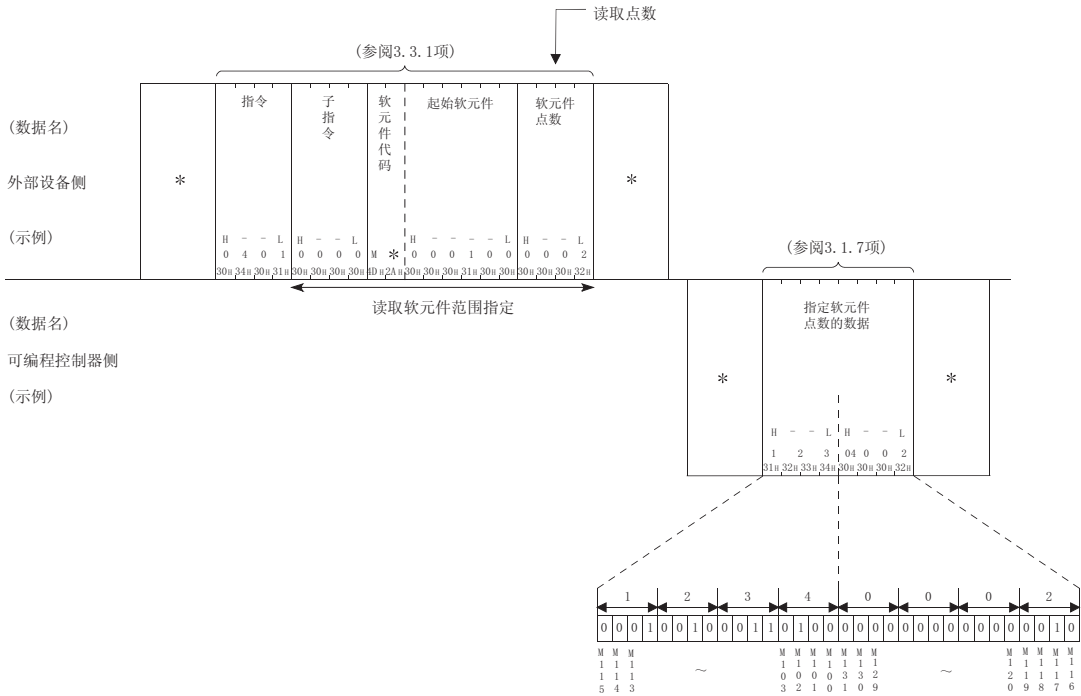
3.3.6 字单位的批量读取(指令：0401)

以下对位软元件存储器(16 位单位)、字软元件存储器(1 字单位)的批量读取的控制步骤进行举例说明。
对于控制步骤图中所示*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

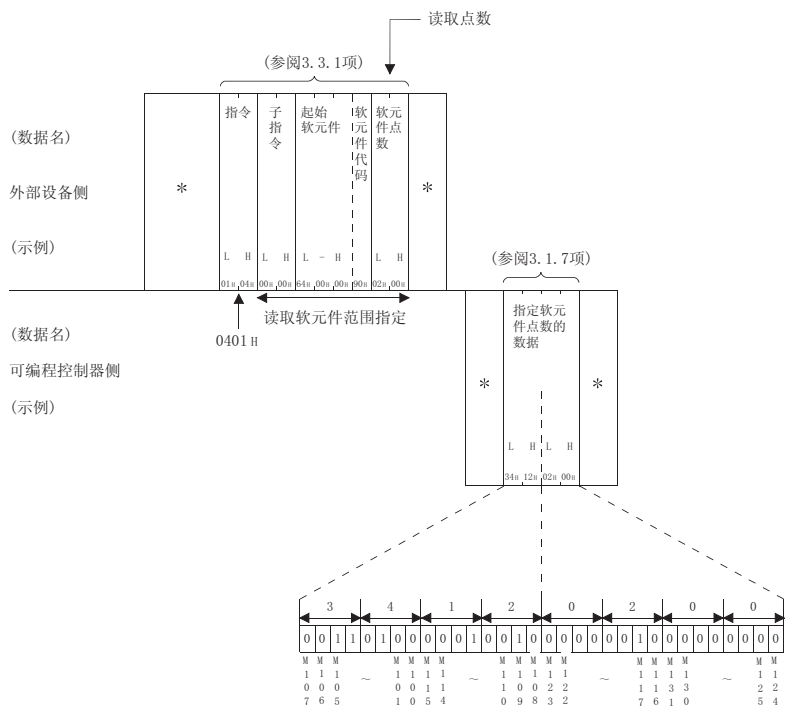
[控制步骤]

(1) 对位软元件存储器进行读取的情况下

(a) 通过由 ASCII 代码进行的通信，对内部继电器 M100~M131 的 2 点(32 位)进行读取的情况下



(b) 在通过二进制代码进行的通信中，对内部继电器 M100～M131 的 2 点(32 位)进行读取的情况下

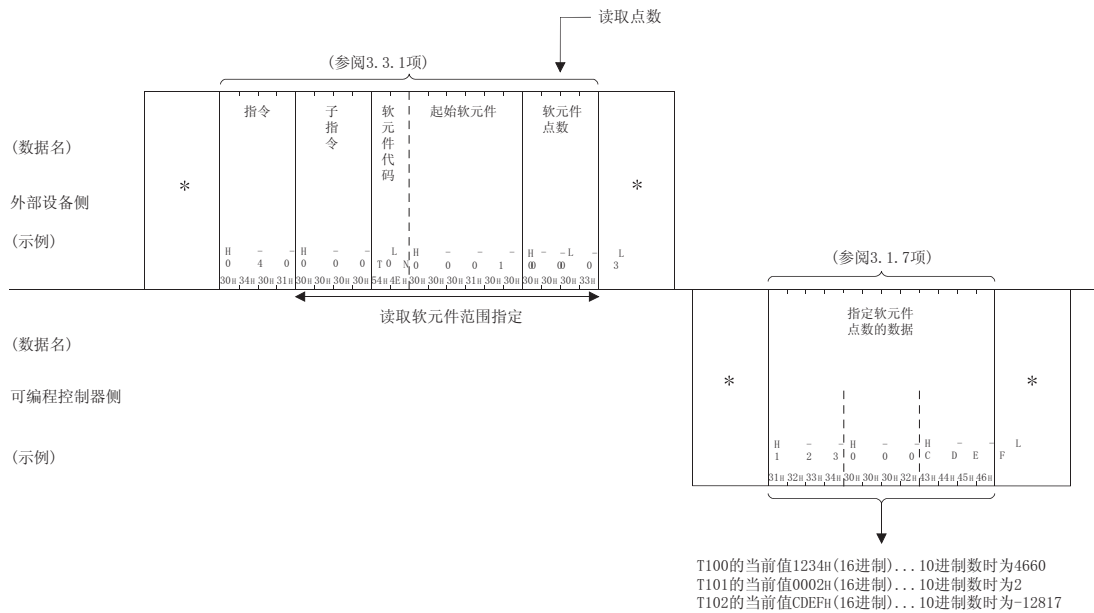


要点

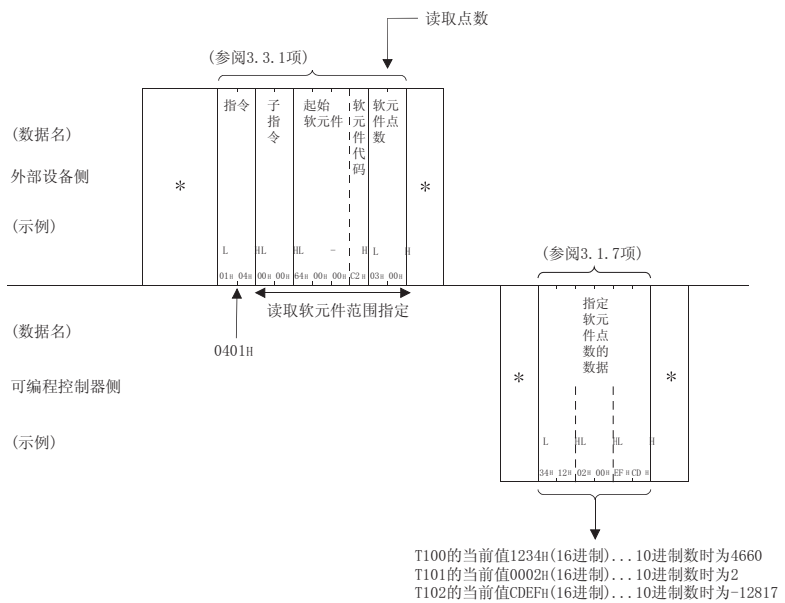
- (1) 应将软元件点数在以下范围内进行指定。(对于位软元件，1 点为 16 位。)
 - (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q/L 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时
软元件点数 : 1 软元件点数 960
 - (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时
软元件点数 : 1 软元件点数 480
 - (c) 对除上述以外的可编程控制器 CPU(其它站)进行访问时
软元件点数 : 1 软元件点数 32
- (2) 应将访问范围在以下范围内进行指定。
访问范围 : (起始软元件号+软元件点数×16-1) ≤ 最大软元件号
* 上述(c)可编程控制器 CPU 的起始软元件号必须设置为 16 的倍数。

(2) 对字软元件存储器进行读取的情况下

(a) 通过由 ASCII 代码进行的通信，对定时器 T100～T102 的 3 点的当前值进行读取的情况下



(b) 在通过二进制代码进行的通信中，对定时器 T100～T102 的 3 点的当前值进行读取的情况下



要点	
(1)	应将软元件点数在以下范围内进行指定。 (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 软元件点数 : 1 软元件点数 960 (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 软元件点数 : 1 软元件点数 480 (c) 对除上述以外的可编程控制器 CPU(其它站)进行访问时 软元件点数 : 1 软元件点数 64 (2) 应将访问范围在以下范围内进行指定。 访问范围 : (起始软元件号+软元件点数-1) ≤ 最大软元件号

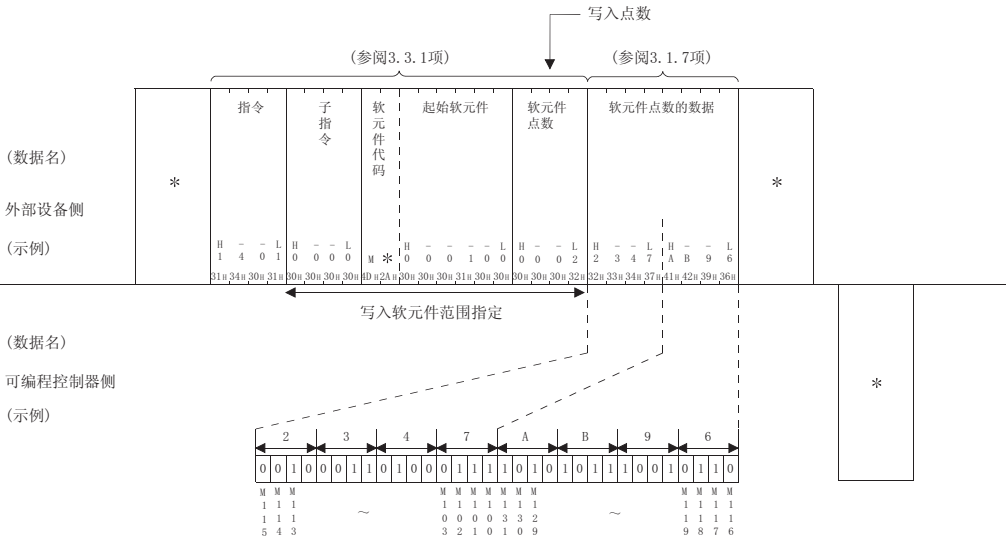
3.3.7 字单位的批量写入(指令：1401)

以下对位软元件存储器(16 位单位)、字软元件存储器(1 字单位)的批量写入的控制步骤进行举例说明。
对于控制步骤图中所示*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

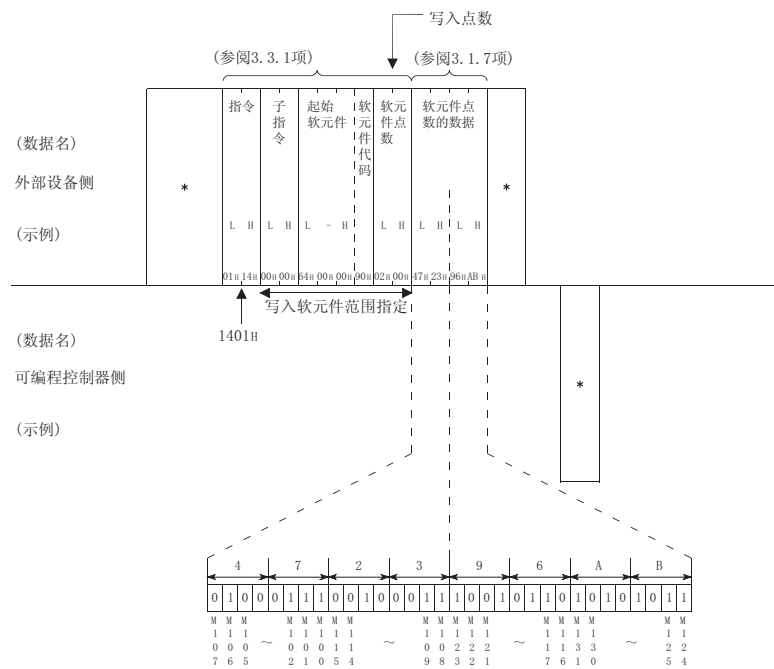
[控制步骤]

(1) 对位软元件存储器进行写入的情况下

(a) 通过由 ASCII 代码进行的通信，对内部继电器 M100~M131 进行 2 点(32 位)写入的情况下



(b) 在通过二进制代码进行的通信中，对内部继电器 M100～M131 写入 2 点 (32 位) 的情况下

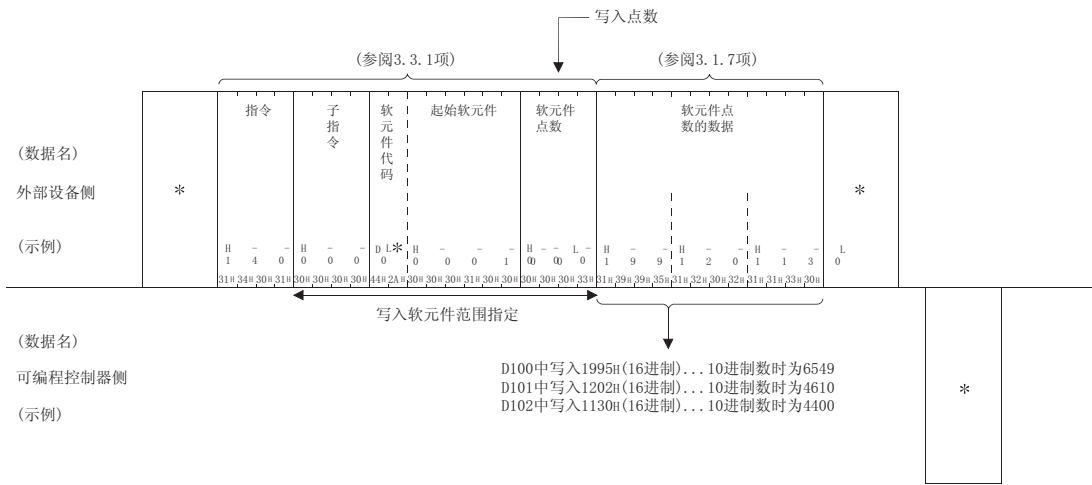


要点

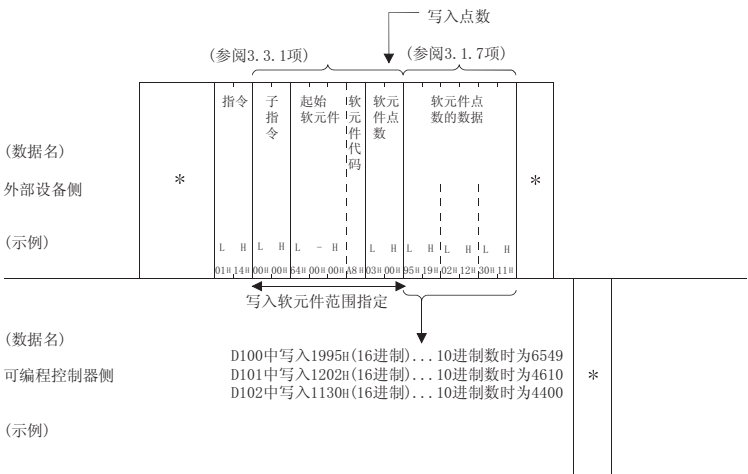
- (1) 应将软元件点数在以下范围内进行指定。
对于位软元件，1 点为 16 位。
 - (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q/L 系列对应网络系统 (CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网) 的 QCPU(其它站) 进行访问时
访问点数 : 1 软元件点数 960
 - (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统 (MELSECNET/10、以太网) 的 Q/QnACPU(其它站) 进行访问时
访问点数 : 1 软元件点数 480
 - (c) 对除上述以外的可编程控制器 CPU(其它站) 进行访问时
访问点数 : 1 软元件点数 10
- (2) 访问范围应在以下范围内。
访问范围 : (起始软元件号+软元件点数×16-1) ≤ 最大软元件号
- (3) 对上述(c)可编程控制器 CPU 的位软元件进行访问时，起始软元件的编号必须设置为 16 的倍数(10 进制数的情况下 0、16...)。
- (4) 对 Q/L/QnACPU 进行写入时，如果 Q/L/QnACPU 被实施了系统保护将变为出错状态，将返回出错时的结束代码。

(2) 对字软元件存储器进行写入的情况下

(a) 通过由 ASCII 代码进行的通信，从 D100 对 D102 进行 3 点的写入的情况下



(b) 在通过二进制代码进行的通信中，对 D100~D102 写入 3 点的情况下



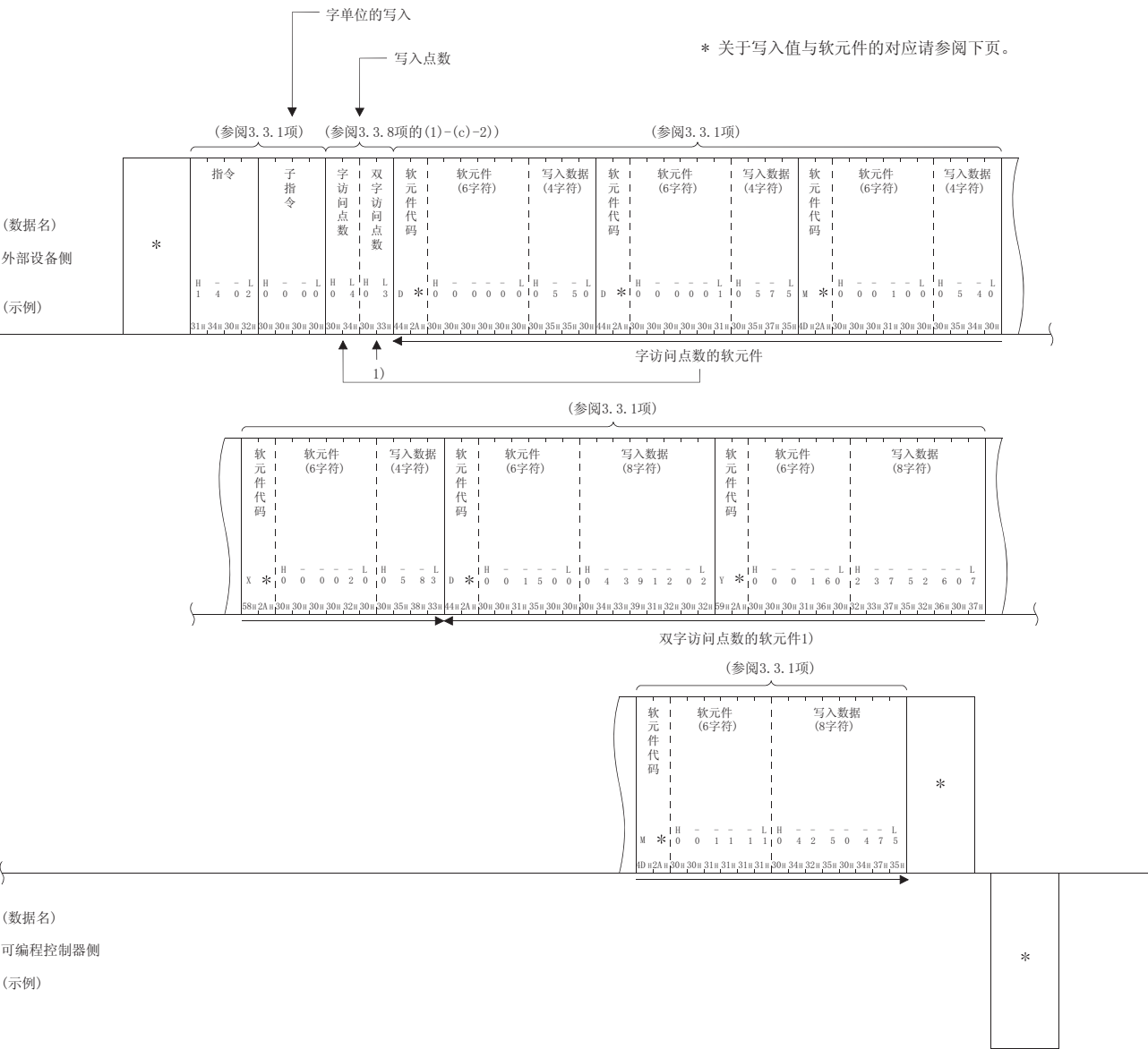
要点	
(1)	应将软元件点数在以下范围内进行指定。
(a)	对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 访问点数 : 1 软元件点数 960
(b)	对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 访问点数 : 1 软元件点数 480
(c)	对除上述以外的可编程控制器 CPU(其它站)进行访问时 访问点数 : 1 软元件点数 64
(2)	访问范围应在以下范围内。 访问范围 : (起始软元件号+软元件点数-1) ≤ 最大软元件号
(3)	对 Q/L/QnACPU 进行写入的情况下，如果 Q/L/QnACPU 被实施了系统保护将变为出错状态，将返回出错时的结束代码。

3.3.8 字单位的随机写入(测试)(指令：1402)

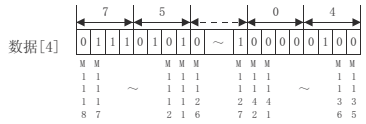
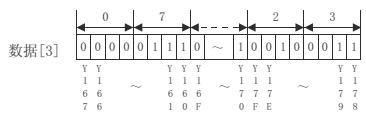
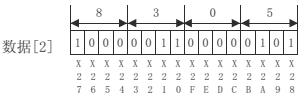
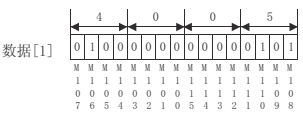
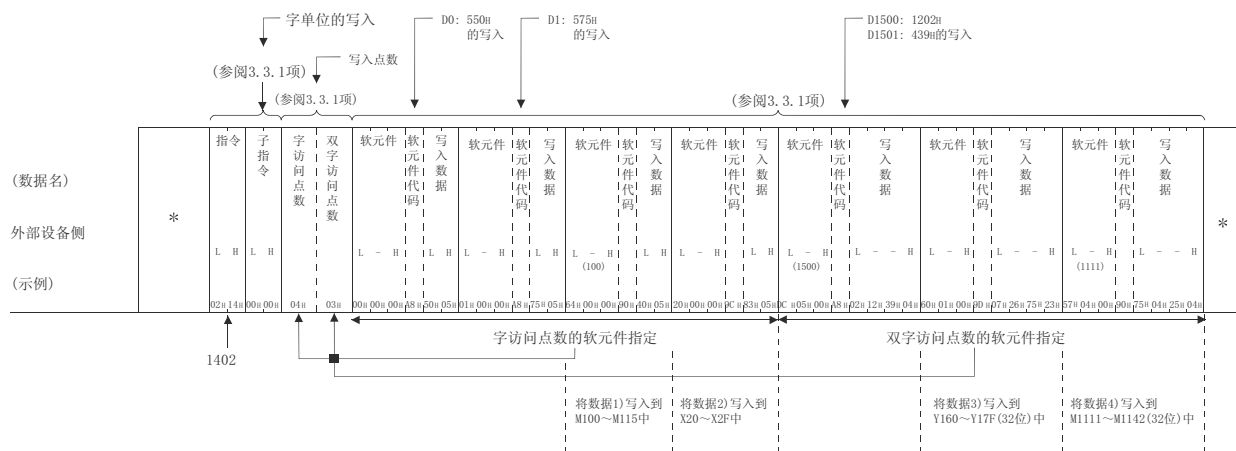
以下对随机指定位软元件存储器(16/32 位单位)、字软元件存储器(1/2 字单位)进行数据写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，写入到以下软元件存储器中的情况下
 - ・字访问 : D0、D1、M100~M115、X20~X2F
 - ・双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



- (b) 在通过二进制代码进行的通信中，对以下软元件存储器进行写入的情况下
- 字访问 : D0、D1、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



要点	
(1)	应将软元件点数在以下范围内进行指定。 位软元件在字访问中 1 点为 16 位，在双字访问中 1 点为 32 位。 (a) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 访问点数 : $1 \leq (\text{字访问点数} \times 12 + \text{双字访问点数} \times 14) \leq 1920$ (b) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 访问点数 : $1 \leq (\text{字访问点数} \times 12 + \text{双字访问点数} \times 14) \leq 960$ (c) 对除上述以外的可编程控制器 CPU(其它站)进行访问时 访问点数 : $1 \leq \text{字访问点数} \leq 10$
(2)	对上述(c)的可编程控制器 CPU 的位软元件进行访问时，必须将起始软元件的编号设置为 16 的倍数(10 进制数的情况下 0、16...)。
(3)	对 Q/L/QnACPU 进行写入的情况下，如果 Q/L/QnACPU 被实施了系统保护将变为出错状态，将返回出错时的结束代码。

3.3.9 字单位的随机读取(指令：0403)

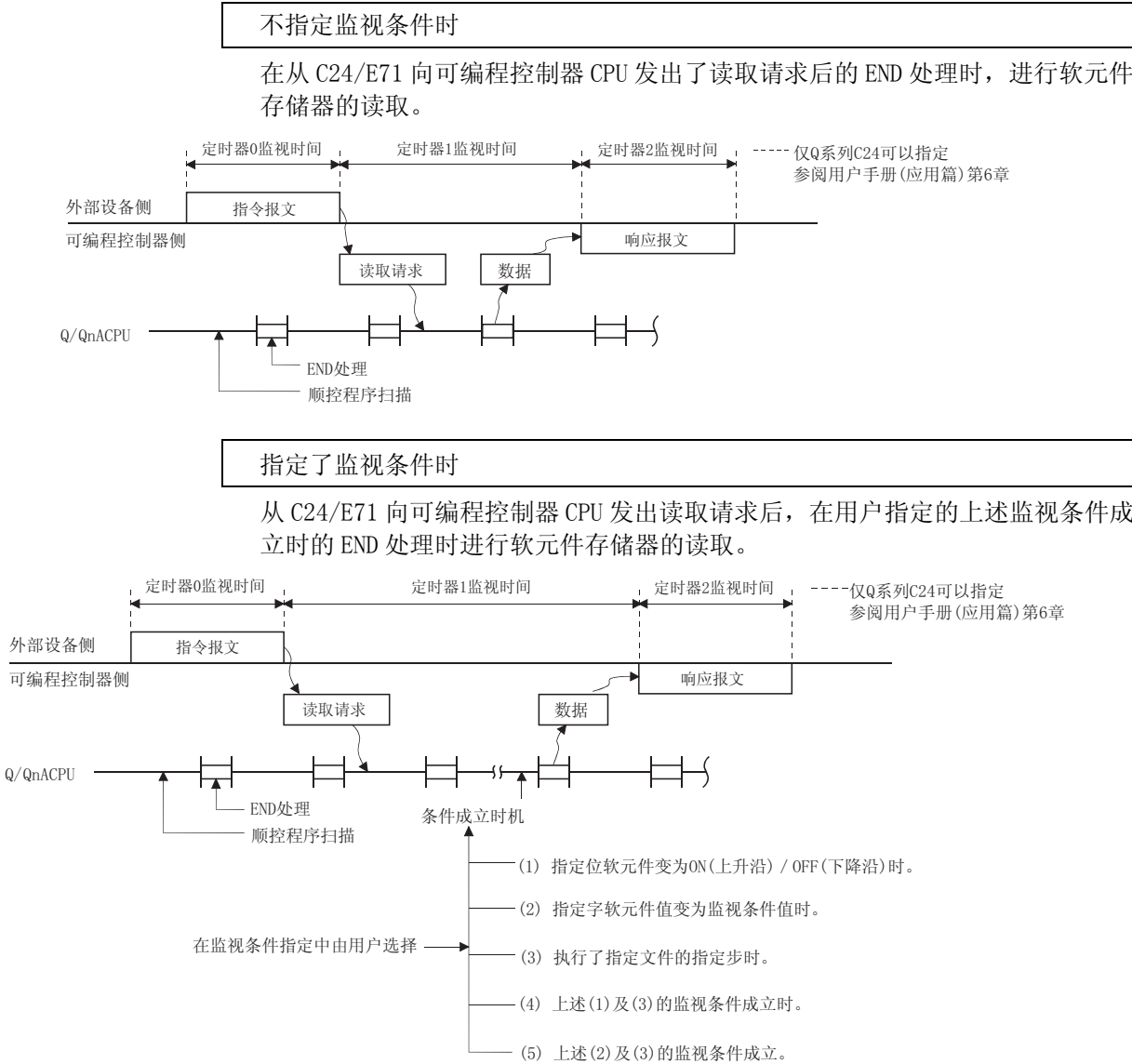
以下对随机指定软元件存储器(16/32 位单位)或字软元件存储器(1/2 字单位)进行数据读取的控制步骤进行举例说明。

在字单位的随机读取中，可以对数据读取时机的以下读取条件(以下略称为监视条件)进行指定。(也可进行组合指定)

- 在可编程控制器 CPU 的 END 处理时，对指定软元件存储器变为 ON 时或变为 OFF 时的软元件存储器进行读取。
- 在可编程控制器 CPU 的 END 处理时，对指定字软元件存储器的值变为监视条件值时的软元件存储器进行读取。(也可将掩码值指定为监视条件值)
- 在可编程控制器 CPU 的 END 处理时，对执行了指定文件的指定步时的软元件存储器进行读取。(也可指定 MELSAP3 的块编号、步号)

关于可指定监视条件的对象 CPU 请参阅 3.2 节。

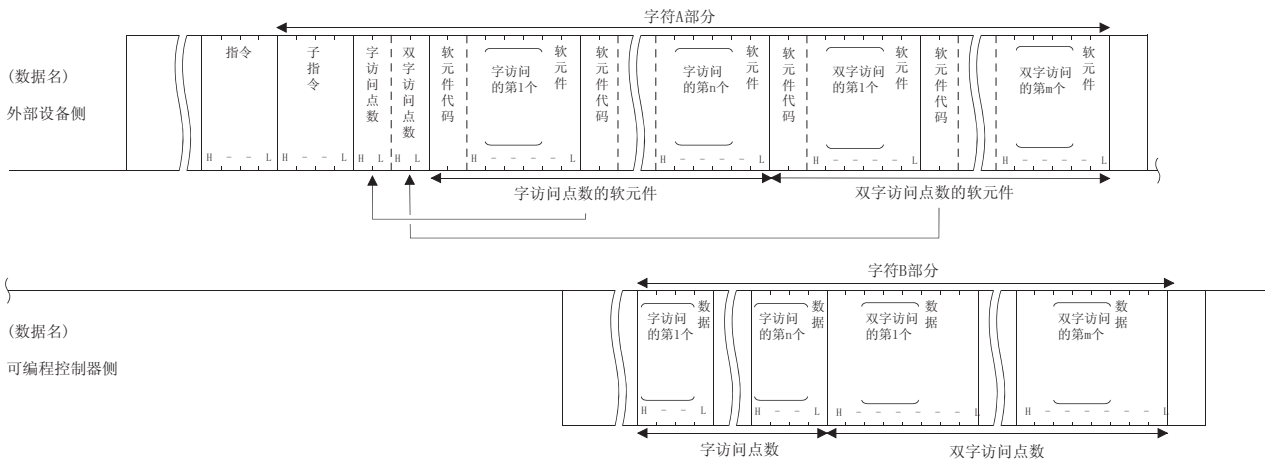
根据监视条件指定的软元件存储器的读取时机如下所示。



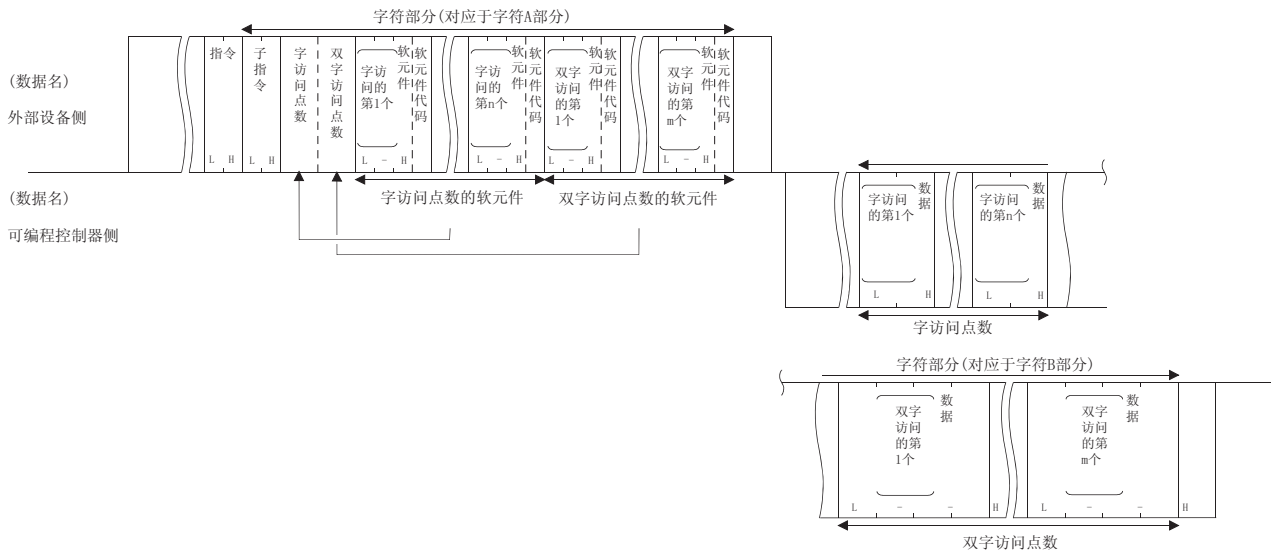
要点	不能从智能功能模块及 GX Works2 或 GX Developer 对 1 个 Q/QnACPU 的软元件存储器同时执行带条件监视。从外部设备向 C24/E71 发送了如下所示的指令报文时，其它智能功能模块及 GX Works2 或 GX Developer 对同一个 Q/QnACPU 进行了带条件监视的情况下，C24/E71 将向外部设备返回异常结束代码。（进行无条件监视时，可以从 C24/E71 进行带条件/无条件的监视。）	
指令	功能	功能说明项
0403	字单位的随机读取功能	本项
0802	登录软元件存储器的监视功能	3. 3. 9 项 (4)

(1) 随机读取时的字符部分的数据的排列及内容
以下对随机读取时的字符部分的数据的排列、内容进行说明。
对于字符部分，与使用其它指令时的排列有部分不同。

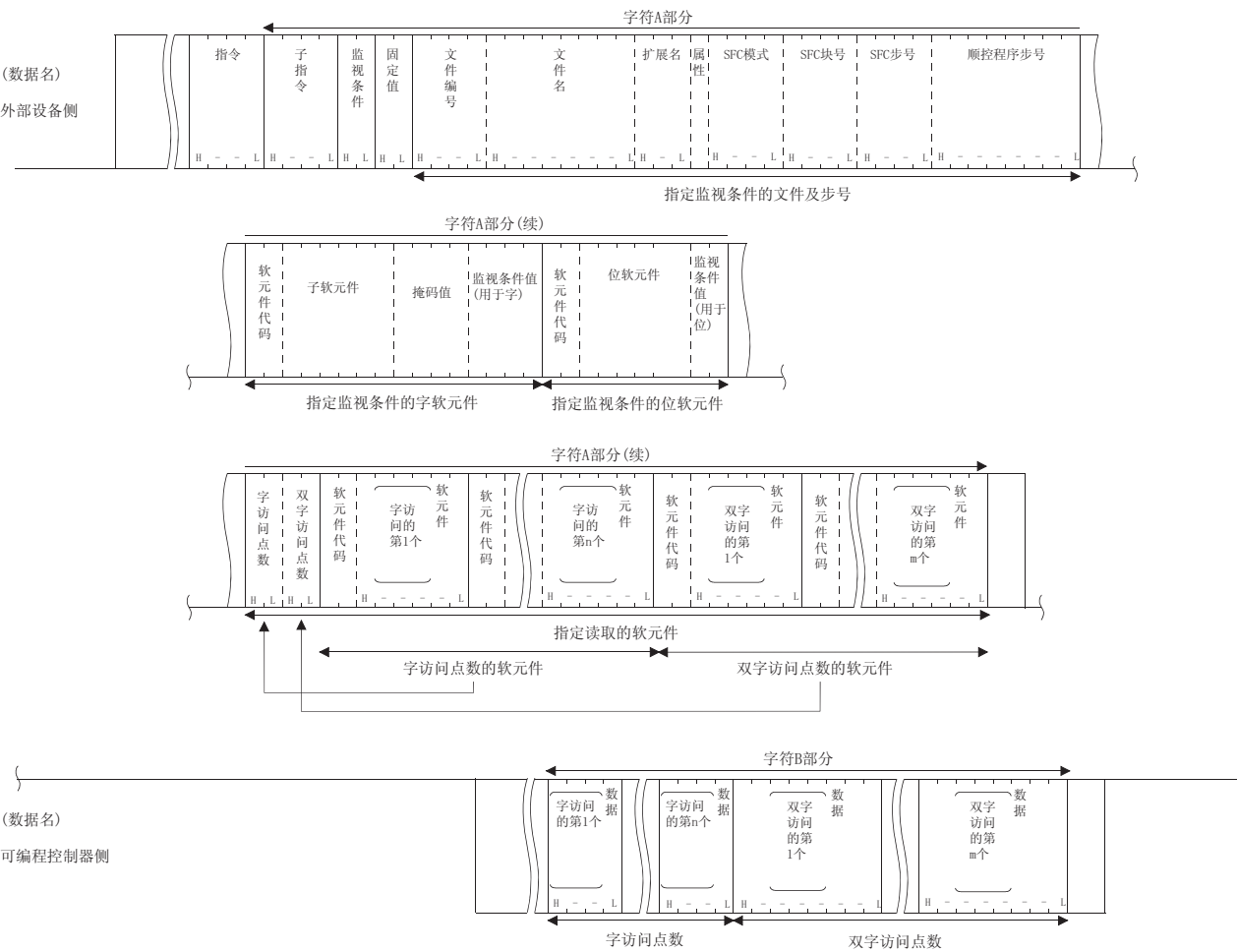
- (a) 未指定监视条件时的字符部分的数据排列
1) 通过 ASCII 代码进行通信时的数据排列



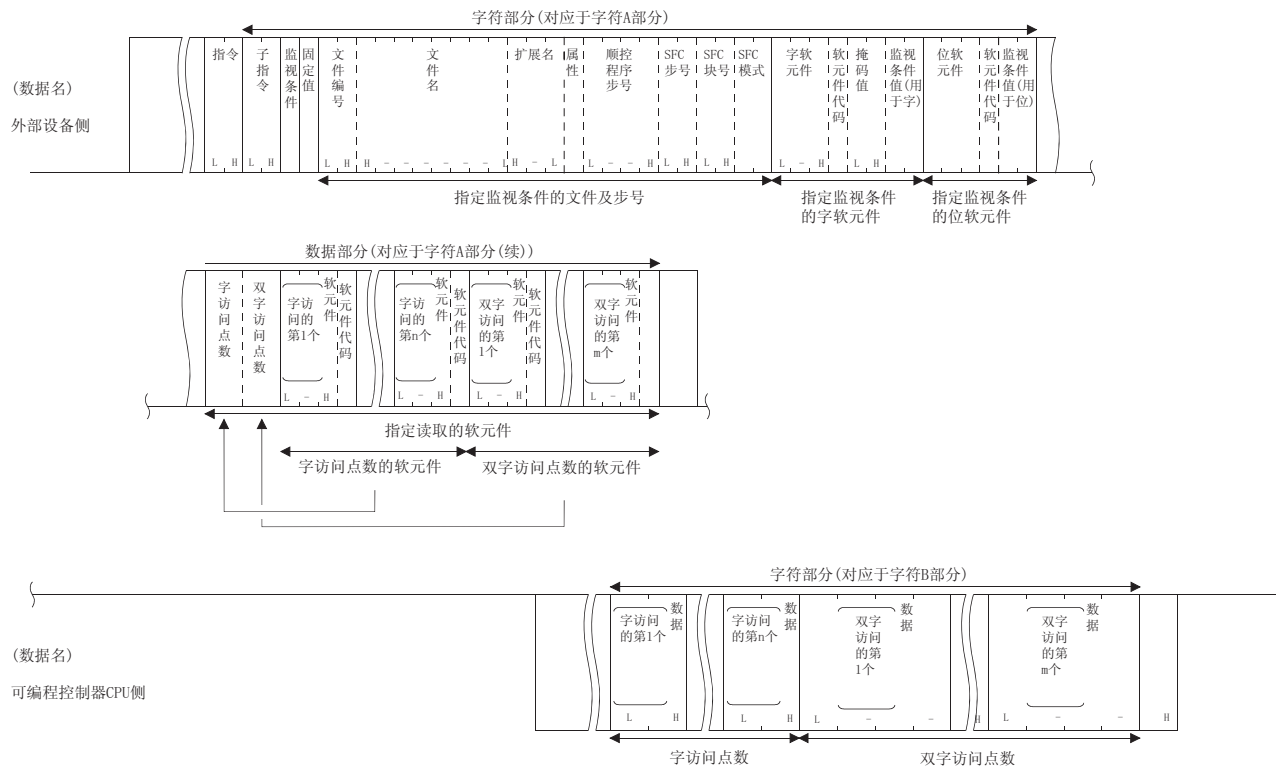
- 2) 通过二进制代码进行通信时的数据的排列



(b) 指定监视条件时的字符部分的数据排列
1) 通过 ASCII 代码进行通信时的数据的排列



2) 通过二进制代码进行通信时的数据的排列



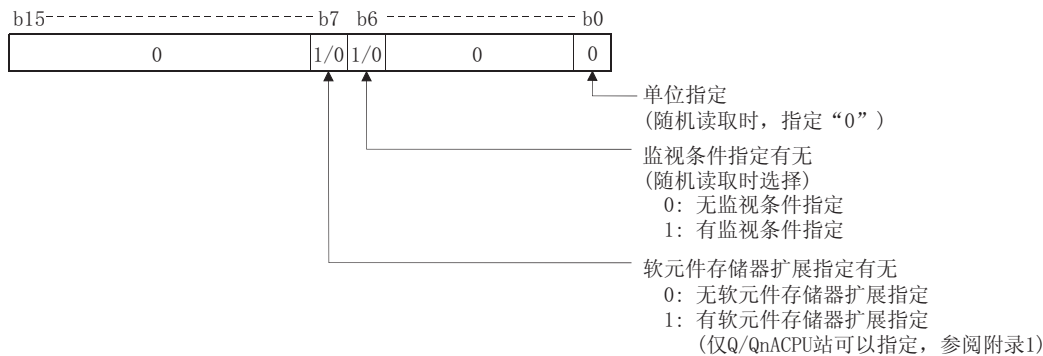
(c) 字符部分的内容

除如下所示的数据以外，与使用其它指令时的内容相同。

1) 子指令

是用于对读取/写入的单位、指定的软元件的类型、数据读取条件(时机)等进行指定的数据。

- a) 通过 ASCII 代码进行数据通信时
将 0000_H(0)或下述数值转换为 ASCII 代码 4 位(16 进制数)后从高位开始进行发送。
- b) 通过二进制代码进行数据通信时
使用 0000_H或下述 2 字节的数值进行发送。
- c) 子指令的指定内容如下所示。



- d) 选择无监视条件指定以及无软元件存储器扩展指定时，子指令将变为“0000_H”。

备注

根据子指令的监视条件指定有无，需要对以后的字符部分进行指定的数据的汇总如下表所示。

数据名	条件指定	无监视条件指定时	有监视条件指定时		
		-	指定文件及步号时	指定字软元件值时	指定位软元件时
字访问点数		●			
双字访问点数		●			
(用于读取软元件指定)		-	-	-	-
(用于字读取)		●			
软元件代码		(但是，字访问点数为 0 点时不需要进行指定)			
软元件					
(用于双字读取)		-	-	-	-
软元件代码		●			
软元件		(但是，双字访问点数为 0 点时不需要进行指定)			
监视条件		×	●		
固定值		×	●		
文件编号 属性		×	●	△	△
SFC 模式 SFC 步号		×	△	△	△
顺控程序步号		×	●	△	△
(用于字软元件值指定)		-	-	-	-
软元件代码 监视条件值(用于字)		×	△	●	△
(用于位软元件指定)		-	-	-	-
软元件代码 监视条件值(用于位)		×	△	△	●

●:需要指定 △: 选择(未指定时指定默认值) ×: 不需要指定

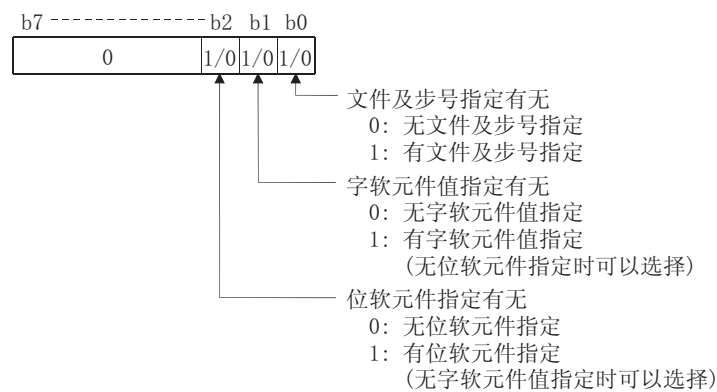
- 2) 字访问点数、双字访问点数
- 是用于以字单位进行读取点数指定、以双字单位进行读取点数指定的数据。
- 各点数的合计应在 3.2 项(1)的表中所示的 1 次通信中可处理的点数以内进行指定。
- a) 通过 ASCII 代码进行数据通信时
- 将各点数转换为 ASCII 代码 2 位(16 进制数)后使用，从各高位开始进行发送。
- (示例)
- 5 点的情况下：变为“05”，从“0”开始按顺序进行发送。
- 20 点的情况下：变为“14”，从“1”开始按顺序进行发送。
- b) 通过二进制代码进行数据通信时
- 使用表示各点数的 1 字节的数值进行发送。
- (示例)
- 5 点的情况下：发送 05_H。
- 20 点的情况下：发送 14_H。
- c) 将某个访问点数设置为 0 点时，不需要指定数据读取相应的软元件、软元件代码。

- 3) 监视条件
- 是用于对数据读取条件进行指定的数据。
- 可指定的监视条件的组合如下所示。
- (不能同时指定 3 个条件。)

监视条件 组合	指定文件及步号	指定字软元件值	指定位软元件
单一指定	○	○	○
组合指定	○	○	○
	○		○

○：可以指定

- a) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位 (16 进制数) 后使用，从高位开始进行发送。
- b) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- c) 监视条件的指定内容如下所示。



- d) 在子指令中选择了有监视条件指定时，监视条件将变为 “01_H” ～ “05_H”。
- 不能使用 “00_H”。
- e) 在子指令中选择了无监视条件指定时，不需要进行监视条件指定。

- 4) 固定值
- a) 通过 ASCII 代码进行数据通信时
使用 “0F”，从高位 (“0”) 开始进行发送。
- b) 通过二进制代码进行数据通信时
使用 1 字节的数值 “0F_H” 进行发送。
- c) 在子指令中选择了无监视条件指定时，不需要进行固定值的指定。

5) 文件编号

是用于指定将通过下述文件名及扩展名指定的文件登录(写入)到可编程控制器 CPU 中时的登录号的数据，通过下述某个数据进行指定。

文件编号	内容	指定内容
0000 _H	无文件名指定	在监视条件中选择了无文件及步号指定时进行此指定。
0001 _H ~ 0100 _H	文件编号	知道文件编号时进行此指定。
FFFF _H	文件编号不明	使 C24/E71 检索文件编号时进行此指定。 (从 C24/E71 至可编程控制器 CPU 的随机读取请求将导致 1 个顺控程序扫描时间以上的延迟。)

- a) 通过 ASCII 代码进行数据通信时
将上述文件编号转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
(示例)
1F_H 的情况下变为“001F”，从“0”开始按顺序进行发送。
 - b) 通过二进制代码进行数据通信时
使用上述 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
(示例)
1F_H 的情况下变为 001F_H，按 1F_H、00_H 的顺序进行发送。
 - c) 即使在子指令中选择了有监视条件指定，在监视条件中选择(使用 02_H、04_H)了无文件及步号指定时，文件编号将变为 0000_H。
 - d) 在子指令中选择了无监视条件指定时，不需要进行文件编号的指定。
 - e) 对于文件编号可通过 3.8.6 项、3.8.16 项、3.8.17 项中所示的功能进行确认。
- 6) 文件名、扩展名、属性
- 是用于对下述 9) 顺控程序步号的对象文件进行指定的数据，对对象文件被登录(写入)到可编程控制器 CPU 时的文件名、扩展名、属性进行指定。
- 用户创建文件的属性最初为 20_H(磁盘文件)，用户可对该属性进行更改。(参阅 3.8.15 项、3.8.24 项)
- a) 通过 ASCII 代码进行数据通信时
使用登录时的文件名、扩展名、属性，从各起始字符开始进行发送。
对属性使用空格(代码: 20_H)进行发送。
文件名少于 8 字符时，附加空格(代码: 20_H)。
(示例)
登录时的文件名为“ABCD12”的情况下
变为“ABCD12 _ _”，从“A”开始按顺序进行发送。
 - b) 通过二进制代码进行数据通信时
将文件名、扩展名登录时的各字符代码作为二进制值使用，从各起始字符开始进行发送。
对于属性，使用 1 字节的数值“20_H”进行发送。
文件名少于 8 字符时，附加 20_H。
(示例)
登录时的文件名为“ABCD12”的情况下
变为 41_H、42_H、43_H、44_H、31_H、32_H、20_H、20_H，从 41_H 开始按顺序进行发送。
 - c) 即使在子指令中选择了有监视条件指定，在监视条件中选择(使用 02_H、04_H)了无文件及步号指定时，文件名及扩展名将变为空格或 20_H 的排列。
属性将变为空格或 20_H。

- d) 在子指令中选择了无监视条件指定时，不需要进行文件名、扩展名、属性的指定。
 - e) 对于属性，可以通过 3.8.5 项、3.8.16 项中所示的文件信息的读取功能进行确认。
- 7) SFC 模式
- 作为数据读取时机，是用于执行 MELSAP3 程序(以下略称为 SFC。)的指定顺控程序步号时的数据之一。
- a) 在子指令中选择了有监视条件指定，在 3) 监视条件中选择了有文件及步号指定时，以如下所示的数据对 SFC 模式进行指定。
不能指定除下述以外的数据。

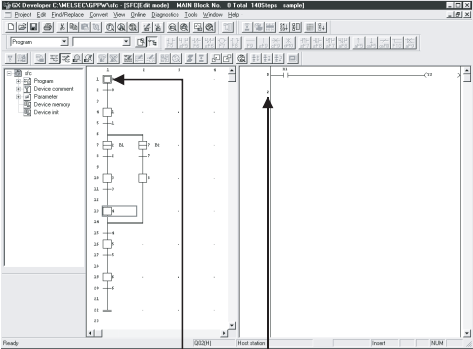
模式	写入数据		备注
ASCII 代码	指定 SFC 时	“0003”	从起始字符(“0”)开始按顺序发送 4 字节
	不指定 SFC 时	“0000”	
二进制代码	指定 SFC 时	0003h	将如左所示的 2 字节的数值从 Low 字节(L: 位 0~7)开始进行发送
	不指定 SFC 时	0000h	

- b) 即使在子指令中选择了有监视条件指定，但在监视条件中选择了无文件及步号指定时，SFC 模式将变为“0000”/“0000h”。
- c) 以“0003”/0003h 指定 SFC 模式的情况下，将在执行了下述 8) 中指定的块号、步号或 9) 中指定的顺控程序步号时的可编程控制器 CPU 的 END 处理中读取指定软元件存储器数据。
- d) 在子指令中选择了无监视条件指定时，不需要进行 SFC 模式的指定。

要点

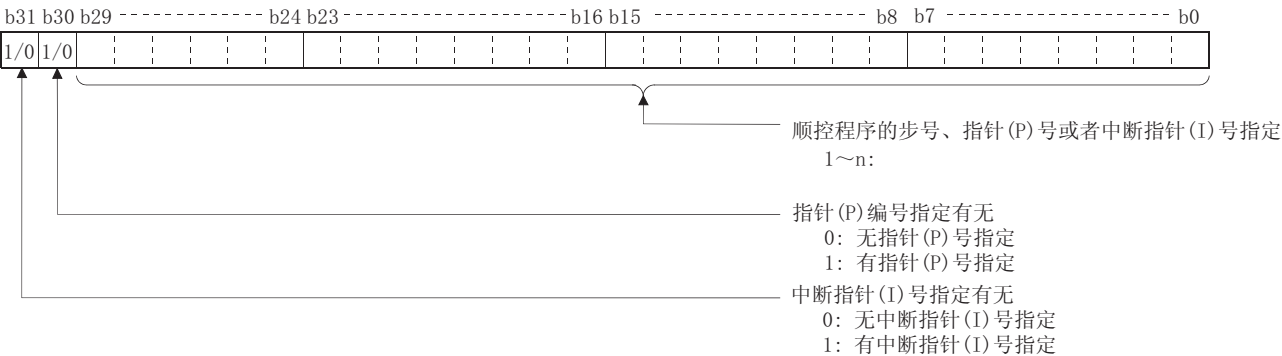
(1) 关于 MELSAP3 的详细内容，请参阅 Q/LCPU 对应的 MELSAP3 的编程手册、操作手册。

(2) 对于 8) 中指定的 SFC 步号、9) 中指定的顺控程序步号，与 MELSAP3 程序编辑时的以下显示部分相对应。



步号 顺控程序步号

- 8) SFC 块号、SFC 步号
- 是用于指定包含数据读取时机(执行指定顺控程序步时)的顺控程序步在的 SFC 块号、SFC 步号的数据。
- 在子指令中选择了有监视条件指定、在监视条件中选择了有文件及步号指定、在 SFC 模式指定中选择了有 SFC 块指定时，可以指定本数据。
- a) 通过 ASCII 代码进行数据通信时
- 将以下数值转换为 ASCII 代码 4 位(16 进制数)后使用，从各高位(“0”)开始进行发送。
- SFC 块号 : 0000H ~ 013FH(0 ~ 319)
 - SFC 步号 : 0000H ~ 01FFH(0 ~ 511)
- b) 通过二进制代码进行数据通信时
- 使用上述 2 字节的数值，从各 Low 字节(L: 位 0~7)开始进行发送。
- (示例)
- 0005H 的情况下按 05H、00H 的顺序进行发送。
- c) 在下述情况下，SFC 块号、SFC 步号均变为“0000H”。
- 即使在子指令中选择了有监视条件指定，但在监视条件中选择了无文件及步号指定时。
 - 在 SFC 模式中选择了无 SFC 块号指定时。
- d) 在子指令中选择了了无监视条件指定时，不需要进行 SFC 块号、SFC 步号的指定。
- 9) 顺控程序步号
- 是用于指定数据读取时机(执行指定顺控程序步时)的顺控程序的步号、指针(P)号或中断指针(I)号的数据。
- 在子指令中选择了有监视条件指定，在监视条件中选择了有文件及步号指定时，可以指定顺控程序步号。
- a) 通过 ASCII 代码进行数据通信时
- 将对象文件中存在范围的下述数值转换为 ASCII 代码 8 位(16 进制数)后使用，从高位开始进行发送。
- b) 通过二进制代码进行数据通信时
- 使用对象文件中存在范围的下述 4 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
- (示例)
- 00000005H 的情况下.....从 05H 开始按顺序进行发送。
- c) 顺控程序步号的指定内容如下所示。



(示例)

指定顺控程序的步号 28 时，变为 0000001CH。

指定中断指针 I28 时，变为 8000001CH。

- d) 即使在子指令中选择了有监视条件指定，但在监视条件中选择了无文件及步号指定时，顺控程序步号将变为“00000000_H”。
- e) 在子指令中选择了无监视条件指定时，不需要进行顺控程序步号的指定。

10) 掩码值、监视条件值(用于字)

是用于指定数据读取时机(指定字软元件变为监视条件值时)的字软元件值等的的数据。

- 掩码值

是用于对监视条件用字软元件的任意位范围的值进行抽取的数据。
(与顺控程序的“WAND”指令的作用相同。)

- 监视条件值(用于字)

是用于对数据读取时机的数值(监视条件用的字软元件值及掩码值的逻辑运算(与“WAND”指令相同)结果)进行指定的数据。

(示例)

将监视条件用的 D0 的位 0~14 的抽取结果变为 3E8_H(1000)的时间作为数据读取时机时，进行如下所示的指定。

掩码值 : 7FFF_H

监视条件值(用于字) : 03E8_H

在子指令中选择了有监视条件指定，在监视条件中选择了有字软元件值指定时，可以对掩码值及监视条件值(用于字)进行指定。

- a) 通过 ASCII 代码进行数据通信时
将上述掩码值、监视条件值的各值转换为 ASCII 代码 4 位(16 进制数)后使用，从各高位开始进行发送。
- b) 通过二进制代码进行数据通信时
使用表示上述掩码值、监视条件值的各 2 字节的数值，从各 Low 字节(L: 位 0~7)开始进行发送。

(示例)

0005_H 的情况下..... 从 05_H 开始按顺序进行发送。

- c) 即使在子指令中选择了有监视条件指定，但在监视条件中选择了无字软元件值指定时，掩码值及监视条件值将变为空格排列或“0000_H”。
在这种情况下，对于用于指定监视条件的字软元件值的字软元件及软元件代码，可指定任意的字软元件存储器及其软元件代码。
- d) 在子指令中选择了无监视条件指定时，不需要对掩码值及监视条件值进行指定。
(也不需要指定用于指定监视条件的字软元件值的字软元件及软元件代码。)

11) 监视条件值(用于位)

监视条件值(用于位)是用于指定数据读取时机条件(上升沿、下降沿)的数据。

在子指令中选择了有监视条件指定，在监视条件中选择了有位软元件指定时，可以对监视条件值(用于位)进行指定。

- a) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- b) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- c) 监视条件值(用于位)的指定内容如下所示。

指定值	读取时机
02 _n	指定位软元件变为上升沿时的可编程控制器 CPU 的 END 处理时
04 _n	指定位软元件变为下降沿时的可编程控制器 CPU 的 END 处理时

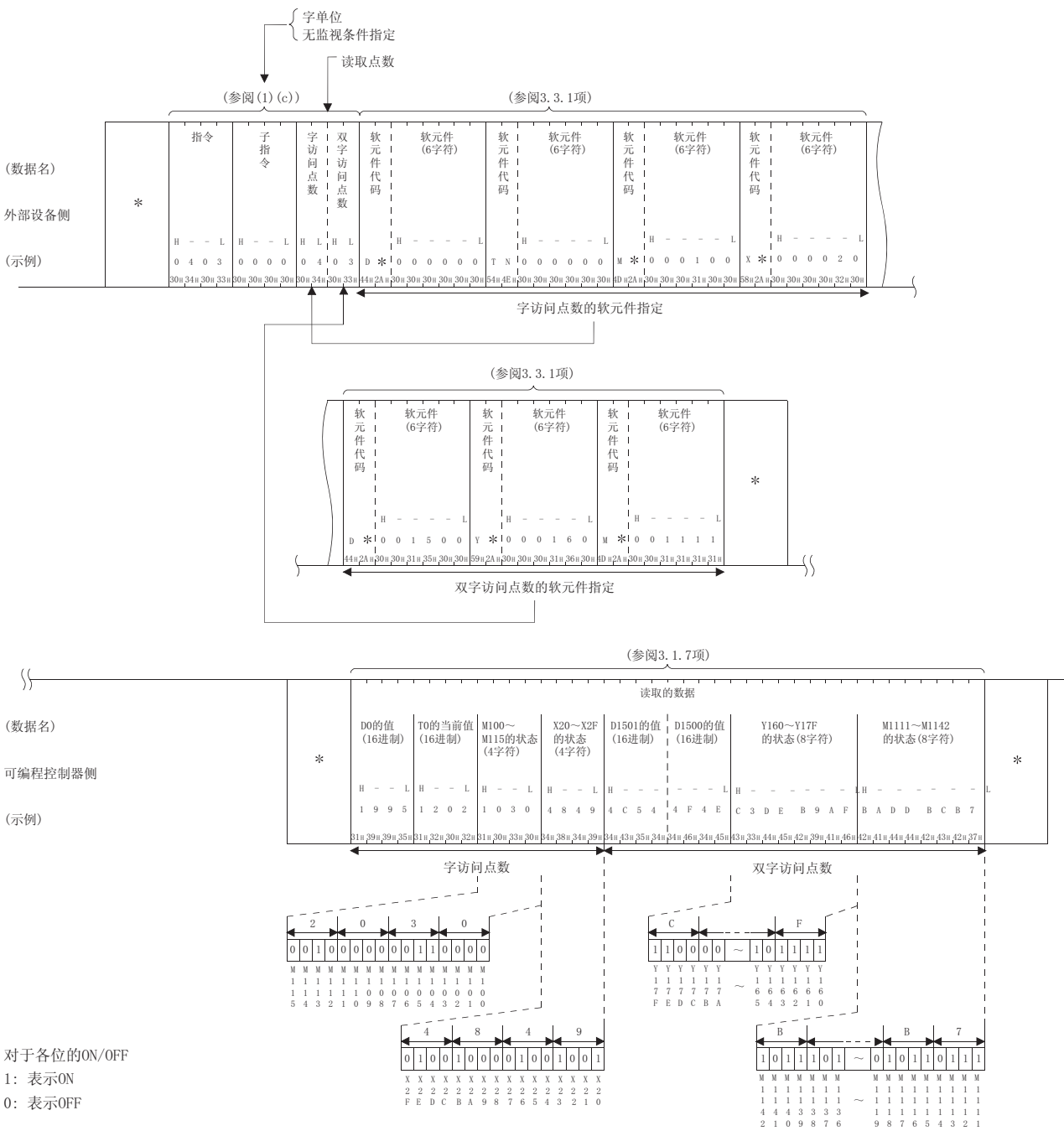
- d) 即使在子指令中选择了有监视条件指定，但在监视条件中选择了无位软元件指定时，监视条件值将变为“00”或“00_H”。
在这种情况下，对于用于指定监视条件的字软元件值的字软元件及软元件代码，可指定任意的字软元件存储器及其软元件代码。
- e) 在子指令中选择了无监视条件指定时，不需要进行监视条件值的指定。
(也不需要指定用于监视条件的位软元件指定的位软元件代码。)

(2) 字单位的随机读取(不指定监视条件的情况下)

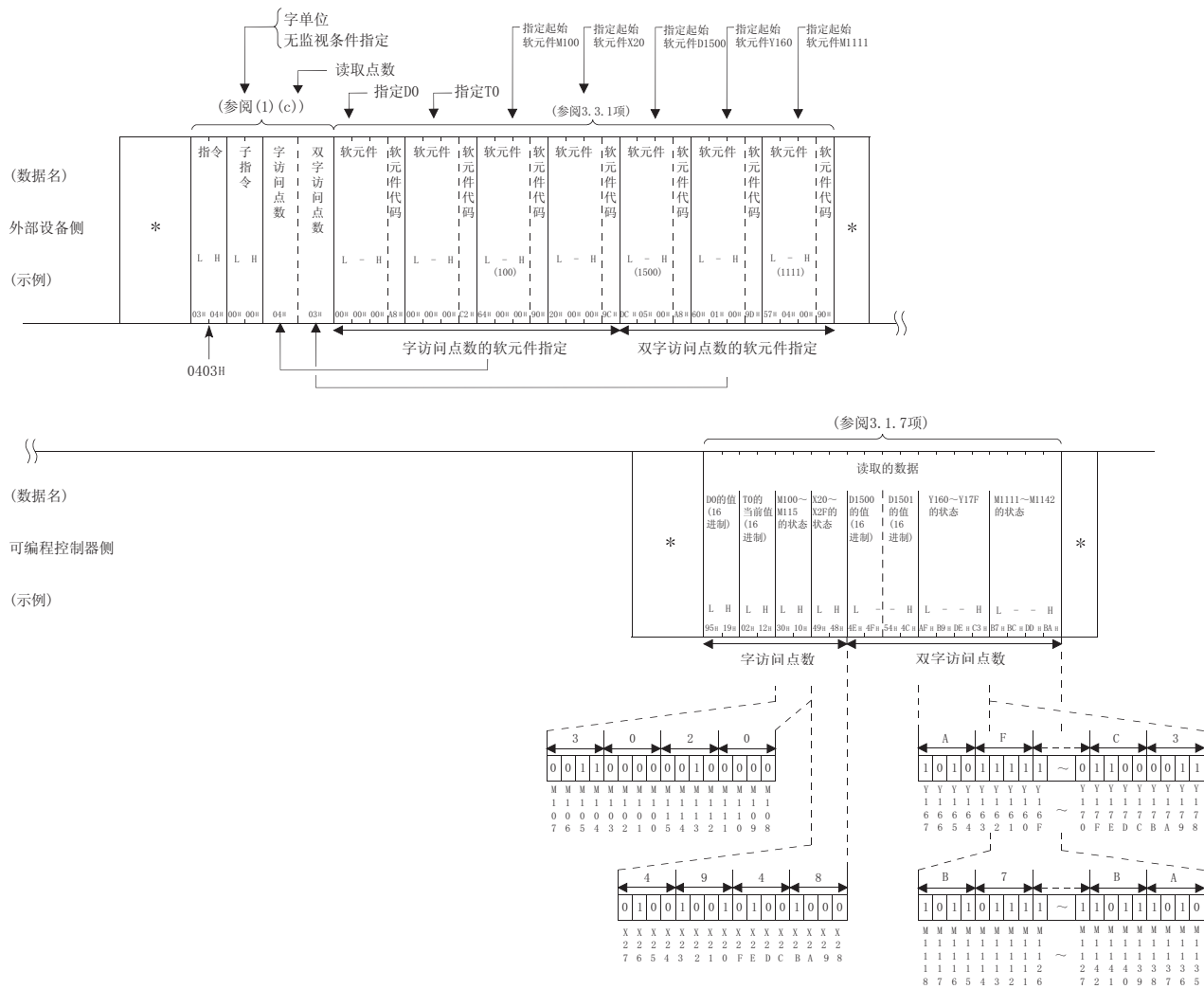
以下对无监视条件(读取条件)的状况下, 随机指定位软元件存储器(16 位单位)、字软元件存储器(1 字单位)进行数据读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容, 根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信, 对以下软元件存储器进行读取的情况下
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



- (b) 在通过二进制代码进行的通信中，对以下软元件存储器进行读取的情况下
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



要点	
	应将访问点数在以下范围内进行指定。 对于位软元件，字访问时 1 点为 16 位，在双字访问中 1 点为 32 位。 对于字软元件，在字访问中 1 点为 1 字，在双字访问中 1 点为 2 字。 (1) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 192$ 此外，对文件寄存器(ZR)进行登录的情况下，应以(访问点数) × 2 进行计算。 例如，登录 15 点的 ZR 的情况下，访问点数以 30 点进行计算。 (但是，基本型 QCPU 除外。) (2) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 96$ (3) 对除(1)、(2)以外的可编程控制器 CPU(其它站)进行访问时不能进行访问。

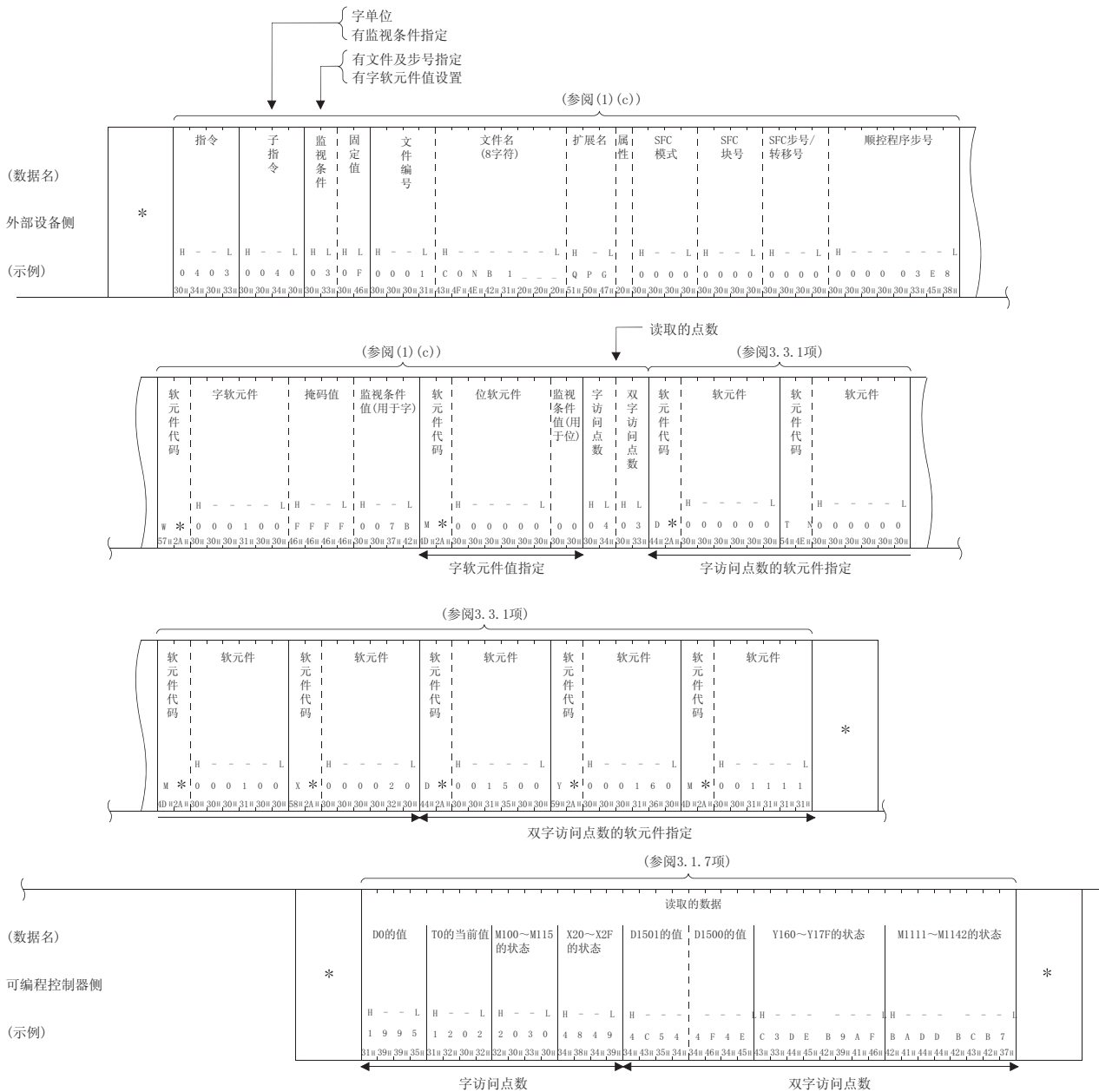
(3) 字单位的随机读取(指定监视条件的情况下)

以下对指定监视条件(读取条件)，随机指定位软元件存储器(16 位单位)、字软元件存储器(1 字单位)进行数据读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

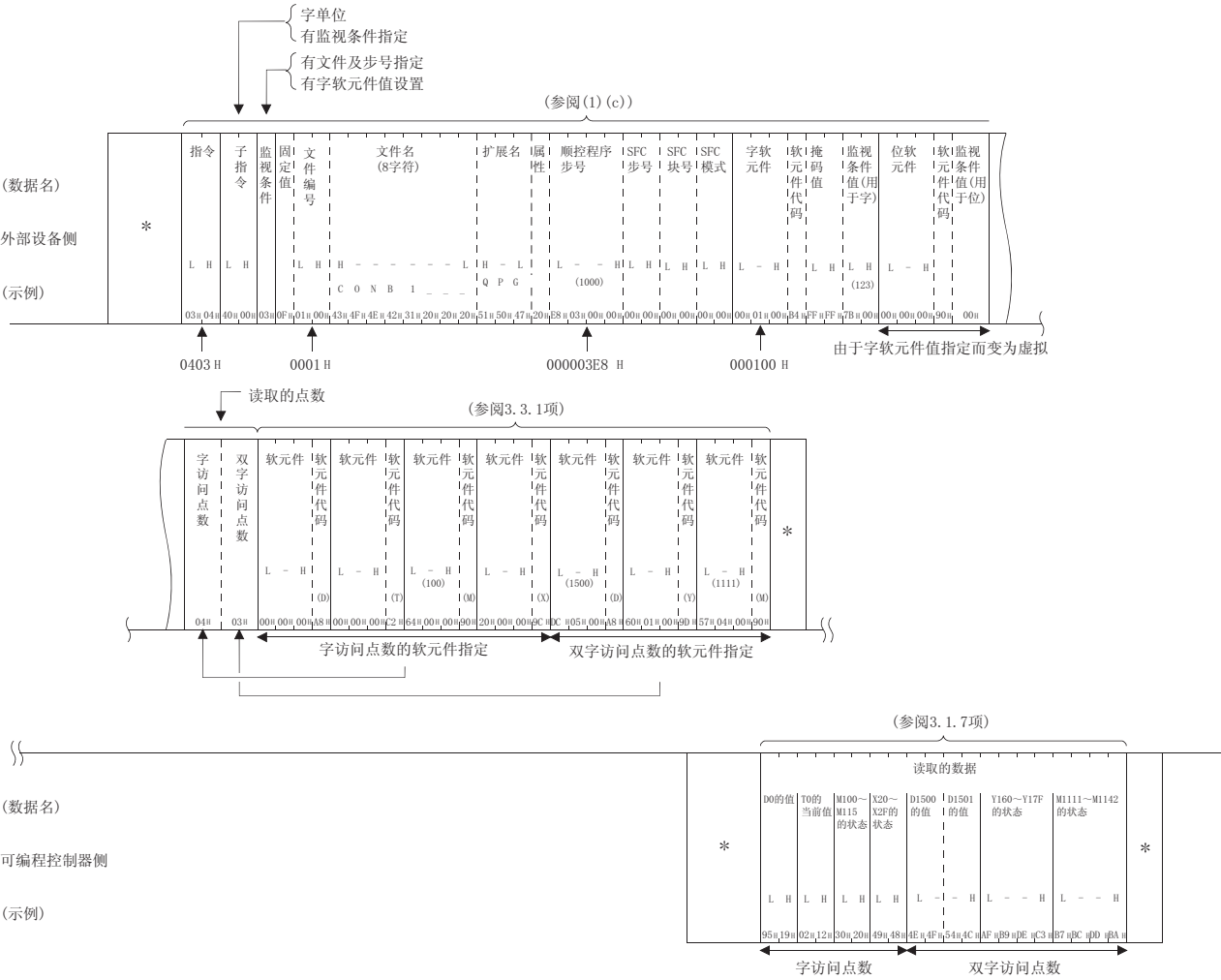
[控制步骤]

(a) 通过由 ASCII 代码进行的通信，对以下内容进行随机读取的情况下

- 1) 监视条件
执行程序文件 CONB1.QPG 的步号 1000 时，链接寄存器 W100 的值变为“7BH”(123)时。
- 2) 读取软元件存储器
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



- (b) 在通过二进制代码进行的通信中，对以下内容进行随机读取的情况下
- 1) 监视条件
执行程序文件 CONB1.QPG 的步号 1000 时，链接寄存器 W100 的值变为“7BH” (123) 时。
 - 2) 读取软元件存储器
 - 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



要点	
	应将访问点数在以下范围内进行指定。 对于位软元件，字访问时 1 点为 16 位，在双字访问中 1 点为 32 位。 对于字软元件，在字访问中 1 点为 1 字，在双字访问中 1 点为 2 字。 (1) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 192$ 此外，对文件寄存器(ZR)进行登录的情况下，应以(访问点数)×2 进行计算。例如，登录 15 点的 ZR 的情况下，访问点数以 30 点进行计算。(但是，基本型 QCPU 除外。) (2) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 96$ (3) 对除(1)、(2)以外的可编程控制器 CPU(其它站)进行访问时不能进行访问。

3.3.10 软元件存储器的监视

监视数据登录功能是指，将希望通过外部设备进行监视的软元件以及编号预先登录到 C24/E71 中的功能。

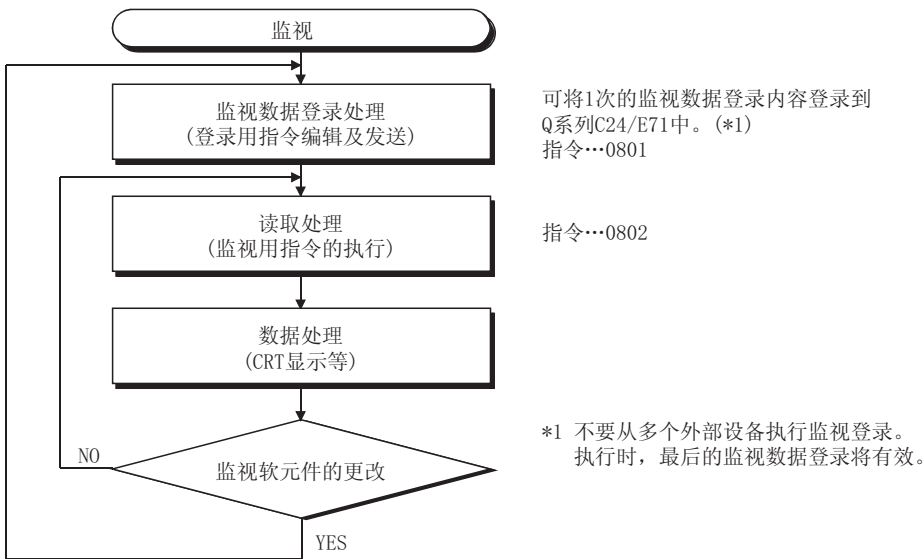
此外，监视功能是指，可以将监视数据登录的软元件的数据内容通过可编程控制器 CPU 进行读取，通过外部设备进行处理的功能。

通过批量读取功能进行读取时软元件编号为连续的编号，而使用本功能时可以随机指定编号进行监视。

关于可指定监视条件的对象 CPU 请参阅 3.2 节。

以下对监视控制步骤以及将希望监视的软元件以及编号登录到 C24/E71 中的控制步骤进行举例说明。

(1) 监视步骤



要点

(1)

在软元件存储器的监视功能中，可以通过以下方法进行数据读取。

关于指定方法及控制步骤内的字符部分的各数据内容以及数据监视(读取)时机，与使用字单位的随机读取功能时相同。详细内容请参阅 3.3.9 项。

1)

可以同时指定位软元件存储器及字软元件存储器。

2)

对于位软元件存储器可以以 16/32 位单位进行读取，对于字软元件存储器可以以 1/2 字单位进行读取。

3)

在进行监视数据登录时可以对数据监视(读取)时机的监视条件进行指定。

(也可进行多个条件的组合指定)

(2)

不能从智能功能模块及 GX Works2 或 GX Developer 同时对 1 个 Q/QnACPU 的软元件存储器进行带条件监视。从外部设备向 C24/E71 发送如下所示的指令报文时，对同一个 Q/QnACPU 通过其它智能功能模块及 GX Works2 或 GX Developer 进行了带条件监视的情况下，C24/E71 将向外部设备发送异常结束代码。(进行无条件监视时，可以从 C24/E71 进行带条件/无条件的监视。)

指令	功能	功能说明项
0403	字单位的随机读取功能	3.3.9 项
0802	登录软元件存储器的监视功能	本项(4)

(3)

按上述步骤执行监视时必须进行监视数据登录。如果在未进行监视数据登录的状况下执行监视，将返回出错结束代码。

(4)

进行了 C24/E71 安装站的重新启动时，监视数据登录的内容将被删除。

对同一个站进行了监视数据登录时，上次的监视数据登录的内容将被删除。

(2) 监视数据登录(指令：0801)(不指定监视条件的情况下)

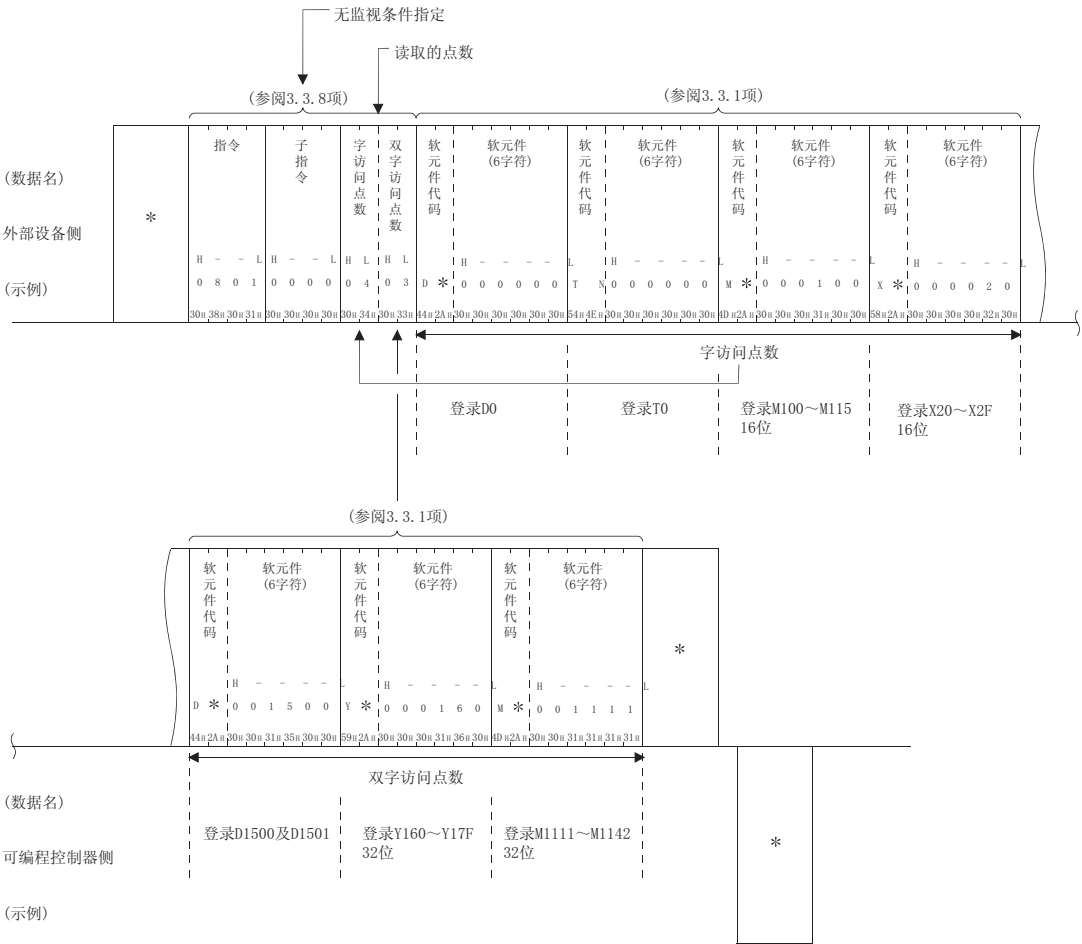
以下对用于随机指定位软元件存储器(16/32 位单位)、字软元件存储器(1/2 字单位)，在无监视条件(读取条件)的状况下进行软元件存储器监视的监视数据登录的控制步骤进行举例说明。

对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。

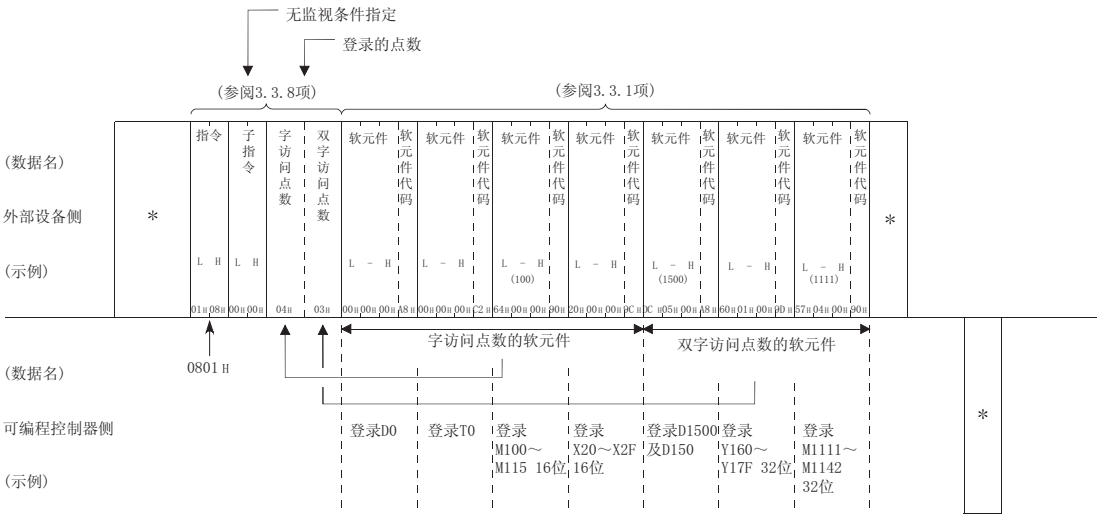
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对以下软元件存储器进行监视数据登录的情况下
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



- (b) 在通过二进制代码进行的通信中，对以下软元件存储器进行监视数据登录的情况下
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



要点

- 应将访问点数在以下范围内进行指定。
- 对于位软元件，字访问时 1 点为 16 位，在双字访问中 1 点为 32 位。
- 对于字软元件，在字访问中 1 点为 1 字，在双字访问中 1 点为 2 字。
- (1) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时
- 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 192$
- 此外，对文件寄存器(ZR)进行登录的情况下，应以(访问点数)×2 进行计算。例如，登录 15 点的 ZR 的情况下，访问点数以 30 点进行计算。(但是，基本型 QCPU 除外。)
- (2) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时
- 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 96$
- (3) 对除(1)、(2)以外的可编程控制器 CPU(其它站)进行访问时不能进行访问。

(3) 监视数据登录(指令：0801)(指定监视条件的情况下)

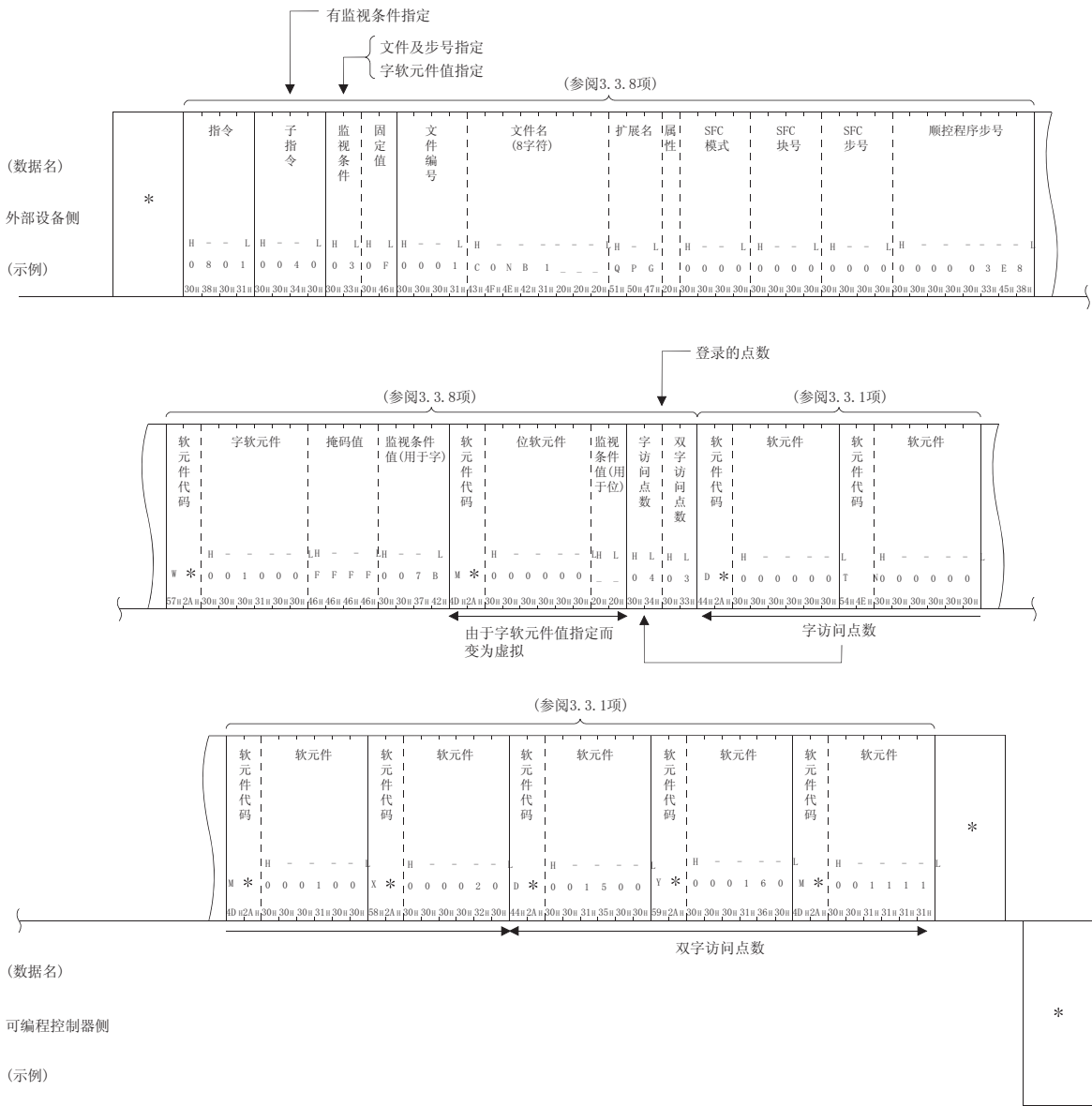
以下对用于随机指定位软元件存储器(16/32 位单位)、字软元件存储器(1/2 字单位)，指定监视条件(读取条件)进行软元件存储器监视的监视数据登录的控制步骤进行举例说明。

对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。

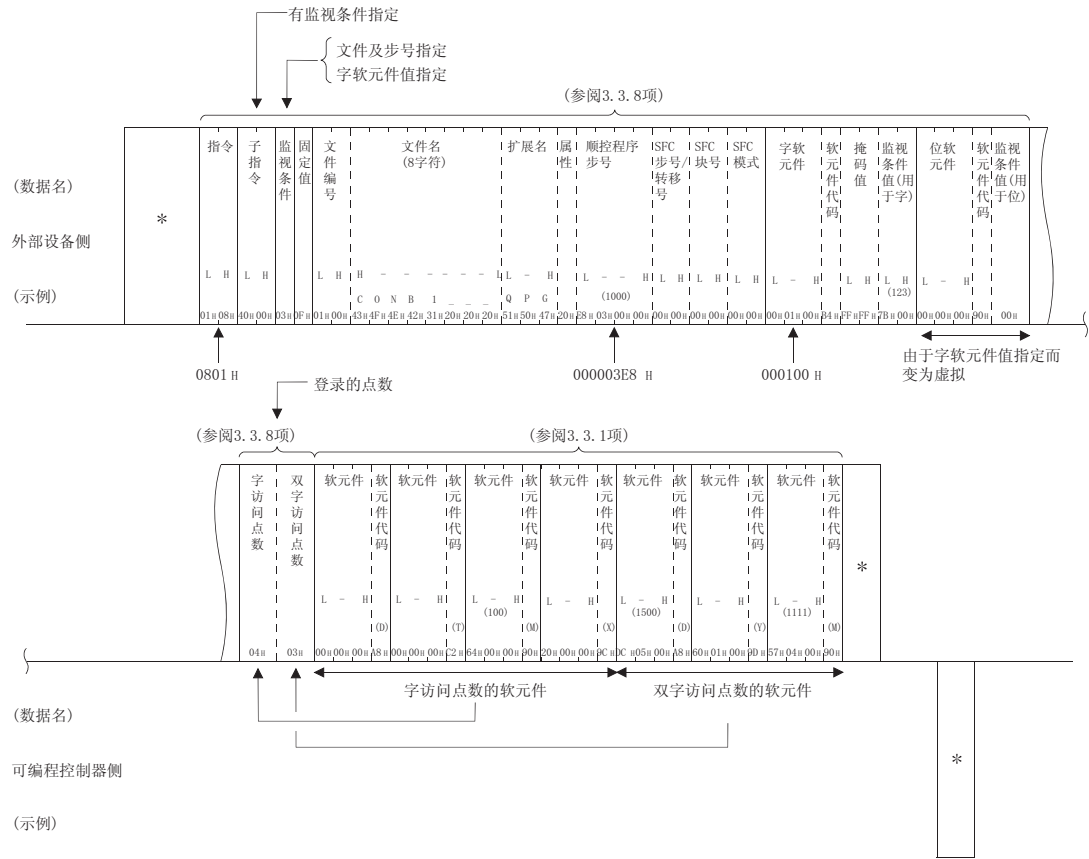
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对以下内容进行监视数据登录的情况下
- 1) 监视条件
执行程序文件 CONB1.QPG 的顺控程序的步号 1000 时，链接寄存器 W100 的值变为“7B_H”(123)时。
 - 2) 进行监视(读取)的软元件存储器
 - 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



- (b) 在通过二进制代码进行的通信中，对以下内容进行监视数据登录的情况下
- 1) 监视条件
执行程序文件 CONB1.QPG 的步号 1000 时，链接寄存器 W100 的值变为“7B_H” (123) 时。
 - 2) 进行监视(读取)的软元件存储器
 - 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



要点	
	应将访问点数在以下范围内进行指定。 对于位软元件，字访问时 1 点为 16 位，在双字访问中 1 点为 32 位。 对于字软元件，在字访问中 1 点为 1 字，在双字访问中 1 点为 2 字。 (1) 对 C24/E71 安装站 Q/LCPU(本站)、经由 Q 系列对应网络系统(CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10、以太网)的 QCPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 192$ 此外，对文件寄存器(ZR)进行登录的情况下，应以(访问点数)×2 进行计算。例如，登录 15 点的 ZR 的情况下，访问点数以 30 点进行计算。(但是，基本型 QCPU 除外。) (2) 对 QnACPU(其它站)、经由 QnA 系列对应网络系统(MELSECNET/10、以太网)的 Q/QnACPU(其它站)进行访问时 访问点数: $1 \leq (\text{字访问点数} + \text{双字访问点数}) \leq 96$ (3) 以外的可编程控制器 CPU(其它站)进行访问时 不能进行访问。

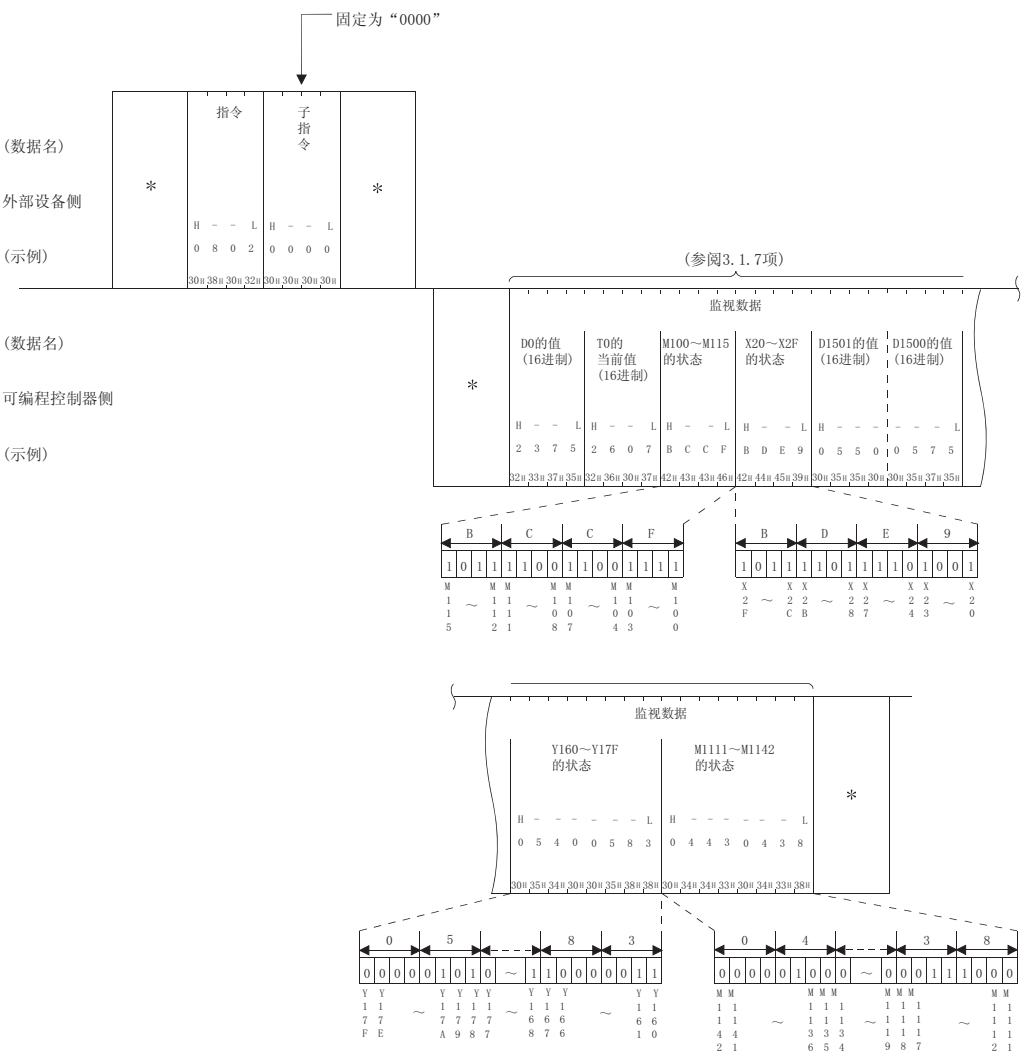
(4) 登录软元件存储器的监视(指令：0802)

以下对监视数据登录(指令：0801)中登录的软元件存储器的监视控制步骤进行举例说明。

与监视数据登录时的监视条件指定的有无无关，监视的控制步骤相同。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

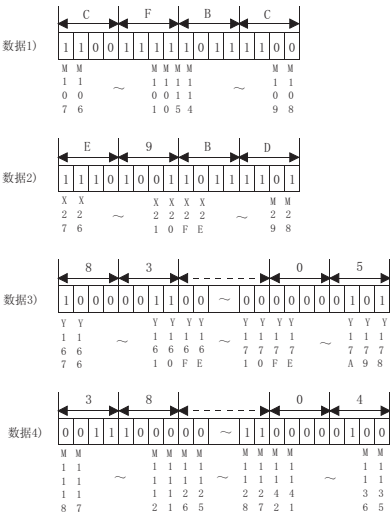
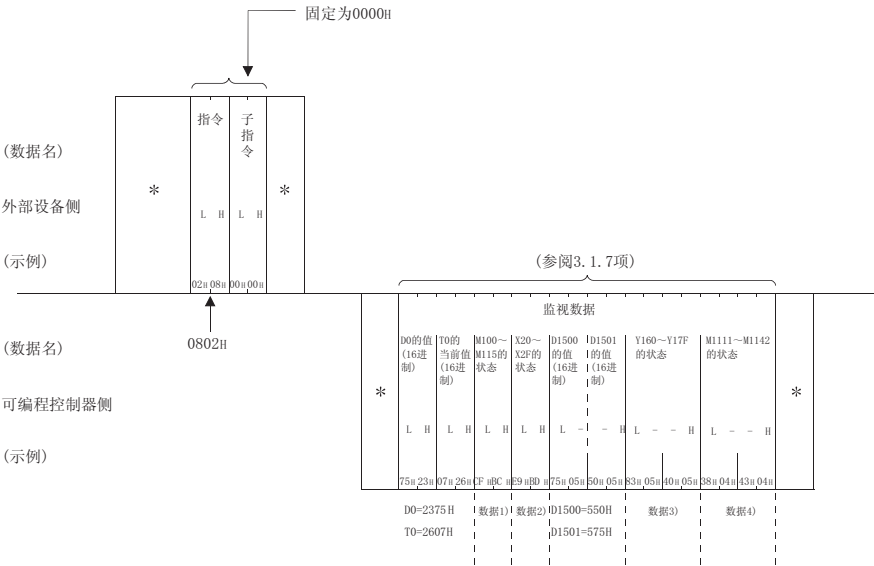
- (a) 对以下软元件存储器进行了监视数据登录时，通过 ASCII 代码通信进行监视的情况下
(进行了监视数据登录的软元件存储器)
- 字访问 : D0、T0、M100~M115、X20~X2F
 - 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



(b) 对以下软元件存储器进行了监视数据登录时，通过二进制代码通信进行监视的情况下

(进行了监视数据登录的软元件存储器)

- 字访问 : D0、T0、M100~M115、X20~X2F
- 双字访问 : D1500~D1501、Y160~Y17F、M1111~M1142



3.4 缓冲存储器的读取、写入

该功能是对与外部设备相连接的下述模块的缓冲存储器进行数据读取、写入的功能。

- C24(多点连接站也包括在内)
- E71

通过该功能进行外部设备与 C24/E71 的通信时，如果从外部设备发出了读取、写入请求，则不等待可编程控制器 CPU 的 END 处理而立即进行通信。

可编程控制器 CPU 侧通过 FROM/TO 指令等对缓冲存储器的数据(与外部设备的发送接收数据)进行读取、写入。

以下对该功能的控制步骤进行举例说明。

3.4.1 关于指令及缓冲存储器

以下对进行 C24/E71 的缓冲存储器的读取・写入时的指令以及控制步骤中指定的缓冲存储器地址等有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
				STOP 中	RUN 中		
					写入允许设置	写入禁止设置	
批量读取	0613 (0000)	对缓冲存储器的数据进行读取。	480 字 (960 字节)	○	○	○	3.4.2 项
批量写入	1613 (0000)	对缓冲存储器进行数据写入。		○	○	○	3.4.3 项

上表的可编程控制器 CPU 的状态栏的○符号表示可以执行。

(2) 关于缓冲存储器、访问单位

本功能中指定的缓冲存储器地址使用各模块的用户手册(基本篇)第 3 章中所示的缓冲存储器列表的地址。

1 地址由 1 字(16 位)所构成。

在本功能中，以字单位进行读取、写入。

- C24 的情况下，与字/字节单位指定无关以字单位进行读取、写入。

要点
• 缓冲存储器的用途是确定的。 • 如果无视规格对 C24、E71 的用途确定区域进行写入，C24/E71 将无法正常动作。

(3) 字符部分的内容

以下对外部设备对 C24/E71 的缓冲存储器进行读取、写入时字符部分的内容进行说明。

(a) 起始地址

是用于对数据读取范围(或写入范围)的起始区域地址进行指定的数据。

- 1) 通过 ASCII 代码进行数据通信时
将起始区域地址 0H~2307H 或 7FFFH 转换为 ASCII 代码 8 位(16 进制数)后使用, 从高位(“0”)开始进行发送。

(示例)

起始区域地址为 1E1H 的情况下变为“000001E1”, 从“0”开始按顺序进行发送。

- 2) 通过二进制代码进行数据通信时
使用表示起始区域地址 0H~2307H 或 7FFFH 的 4 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。

(示例)

起始区域地址为 1E1H 的情况下变为 000001E1H, 从 E1H 开始按顺序进行发送。

(b) 字长

是用于对数据读取范围(或写入范围)的地址数(字数)进行指定的数据。

- 1) 通过 ASCII 代码进行数据通信时
将地址数 1H~1E0H(1~480)转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位(“0”)开始进行发送。

- 2) 通过二进制代码进行数据通信时
使用表示地址数 1H~1E0H(1~480)的 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。

备注

对于报文中的数据指定项目的网络编号及可编程控制器编号, 应指定以本站为对象的数据。(网络编号: “00H” 可编程控制器编号: “FFH”)

(QnA兼容3C帧 格式1的情况下)

(QnA兼容3E帧的情况下)

QnA	帧识别编号		站号		网络站号		可编程控制器编号		本站编号	
	H	L	H	L	H	L	H	L	H	L
	F	9	0	0	0	0	F	F	0	0
05H	46H	39H	30H	30H	30H	30H	46H	46H	30H	30H

(通过ASCII代码进行通信时)

Q头部																	
网络编号		可编程 程控制 器编 号		请求目 标模 块I/O 编号				请求目 标模 块站号		请求数 据长度		CPU监视 定时器					
H	L	H	L	H	-	-	L	H	L	H	-	-	L	H	-	-	L
0	0	F	F	0	3	F	F	0	0	0	0	1	8	0	0	1	0
30H	30H	46H	46H	30H	33H	46H	46H	30H	30H	30H	30H	31H	38H	30H	30H	31H	30H

(通过二进制代码进行通信时)

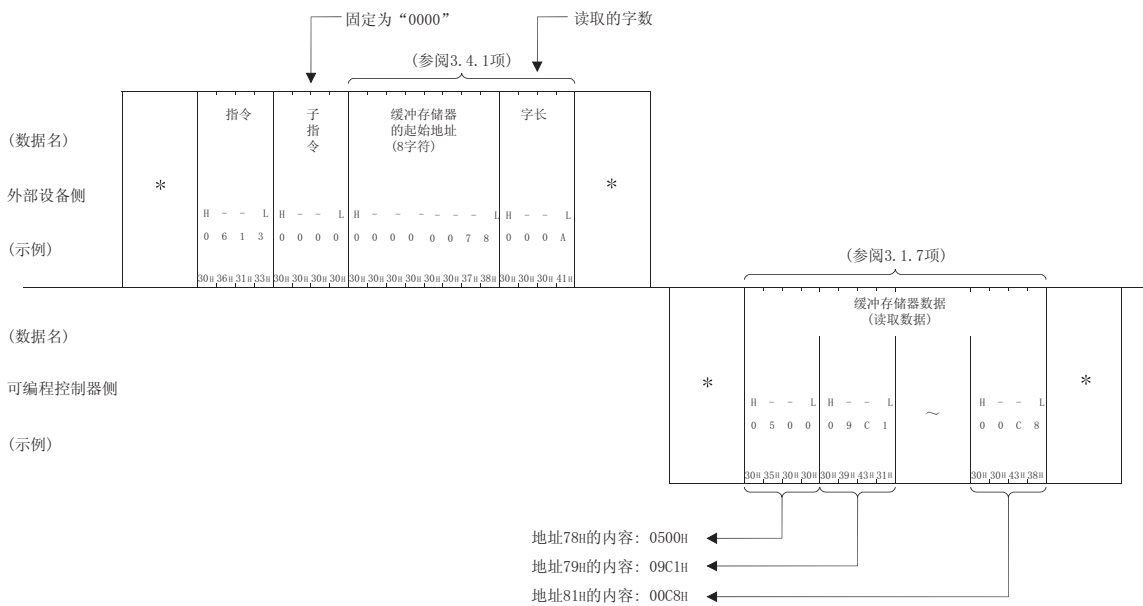
Q头部											
网络编号		可编程控制器编号		请求目标模块 I/O 编号		请求目标模块站号		请求数据长度		CPU监视定时器	
L	H	L	H	L	H	L	H	L	H	L	H
00H	FFH	FFH	03H	00H	0C	00H	10H	00H			

3.4.2 缓冲存储器的读取(指令：0613)

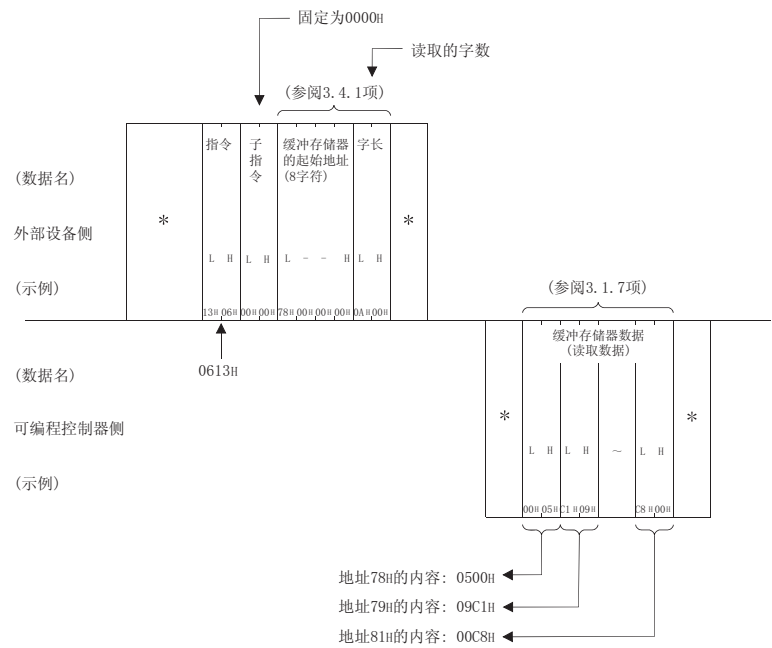
以下举例说明对 C24/E71 的缓冲存储器进行批量读取的控制步骤。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对缓冲存储器地址 78H~81H(120~129)区域的内容读取 10 字的情况下



(2) 在通过二进制代码进行的通信中，对缓冲存储器地址 78H~81H (120~129)区域的内容读取 10 字的情况下



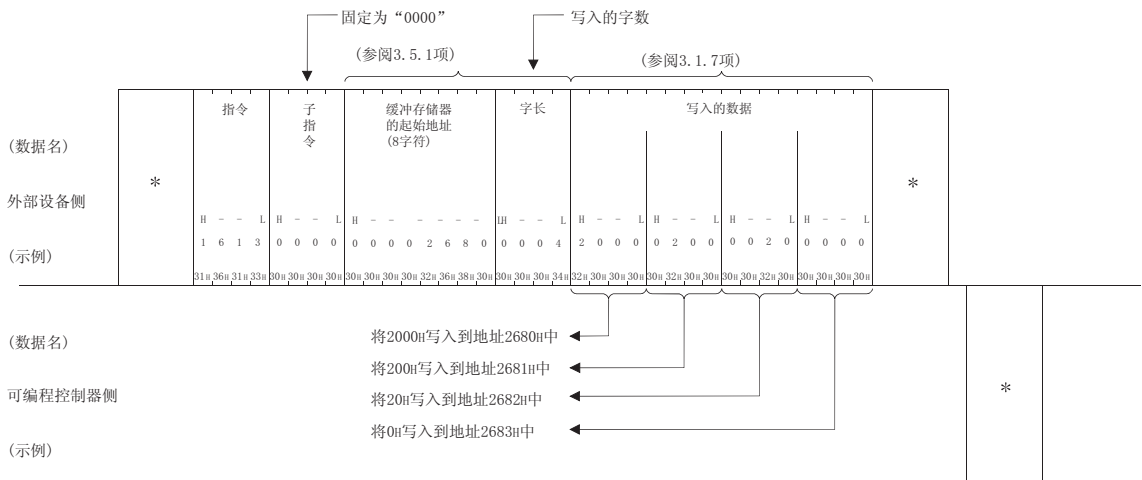
要点	
对起始地址及字长应在以下范围内进行指定。	
• 起始地址	: 0H ≤ 起始地址 ≤ 2307H 或 7FFFH
	C24 : 2307H
	E71 : 7FFFH
• 字长	: 1H ≤ 字长 ≤ 1E0H (480)
• 访问范围	: (起始地址 + 字长 - 1) ≤ 2307H 或 7FFFH

3.4.3 至缓冲存储器的写入(指令：1613)

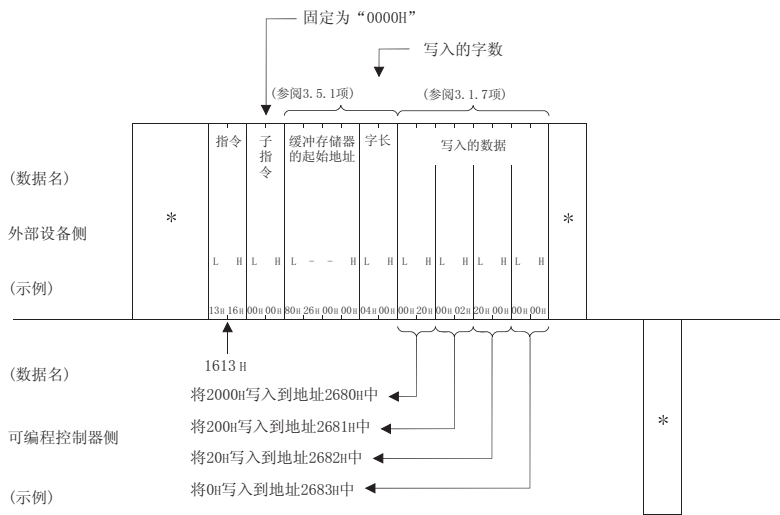
以下举例说明对 C24/E71 的缓冲存储器进行批量写入的控制步骤。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对缓冲存储器地址 2680H～2683H(9856～9859)区域写入 4 字的数据的情况下



(2) 在通过二进制代码进行的通信中，对缓冲存储器地址 2680H～2683H(9856～9859)区域写入 4 字的数据的情况下



要点	
对起始地址及字长应在以下范围内进行指定。	
・ 起始地址	: 0H ≤ 起始地址 ≤ 2307H 或 7FFFH
	C24 : 2307H
	E71 : 7FFFH
・ 字长	: 1H ≤ 字长 ≤ 1E0H (480)
・ 访问范围	: (起始地址 + 字长 - 1) ≤ 2307H 或 7FFFH

3.5 智能功能模块的缓冲存储器的读取、写入

以下举例说明对智能功能模块(*1)的缓冲存储器进行数据读取、写入时的控制步骤。
在本指令中对智能功能模块的缓冲存储器以字节单位进行访问。
*1 与 3.4 节中所示的外部设备相连接的下述模块以及 A/QnA 系列的特殊功能模块也包含在内。
• C24(多点连接也包含在内)
• E71

3.5.1 关于指令及缓冲存储器

以下对进行智能功能模块的缓冲存储器的读取・写入时的指令以及控制步骤中指定的缓冲存储器地址等有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
				STOP 中	RUN 中		
					写入允许设置	写入禁止设置	
批量读取	0601 (0000)	对缓冲存储器的数据进行读取。	1920 字节 (960 字)	☐	☐	☐	3.5.3 项
批量写入	1601 (0000)	对缓冲存储器进行数据写入。		☐	☐	☐	3.5.4 项

上表的可编程控制器 CPU 的状态栏的○符号表示可以执行。

(2) 关于缓冲存储器、访问单位

对于本功能中指定的缓冲存储器地址，以 3.5.2 项中所示的方法进行指定。
1 地址由 1 字(16 位)所构成，在本功能中以字节单位进行读取、写入。
* C24 的情况下，与字/字节单位指定无关，以字节单位进行读取、写入。

(3) 字符部分的内容

以下对外部设备进行智能功能模块的缓冲存储器的读取、写入时的字符部分的内容进行说明。

(a) 起始地址

- 是用于对数据读取范围(或写入范围)的起始区域地址进行指定的数据。
起始地址的指定方法如 3) 中所示。
访问可能模块及缓冲存储器起始地址如 3.5.2 项中所示。
- 1) 通过 ASCII 代码进行数据通信时
将起始区域地址转换为 ASCII 代码 8 位(16 进制数)后使用，从高位(“0”)开始进行发送。
(示例)
起始区域地址为 1E1H 的情况下变为“000001E1”，从“0”开始按顺序进行发送。
 - 2) 通过二进制代码进行数据通信时
使用表示起始区域地址的 4 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。

(示例)

起始区域地址为 1E1H 的情况下变为 000001E1H，从 E1H 开始按顺序进行发送。

- 3) 以下介绍进行智能功能模块的缓冲存储器的读取、写入时的起始地址的指定方法。

智能功能模块的缓冲存储器由 1 地址 16 位(1 字)所构成，在可编程控制器 CPU 与智能功能模块之间通过 FROM/TO 指令等进行读取、写入。通过 3. 5. 1 项中所示的指令从外部设备经由 C24/E71 对智能功能模块的缓冲存储器进行读取、写入时，以 1 地址=8 位(1 字节)单位进行。对于外部设备中指定的地址(16 进制)，以从用于 FROM/TO 指令的地址进行了下述换算后的地址进行指定。

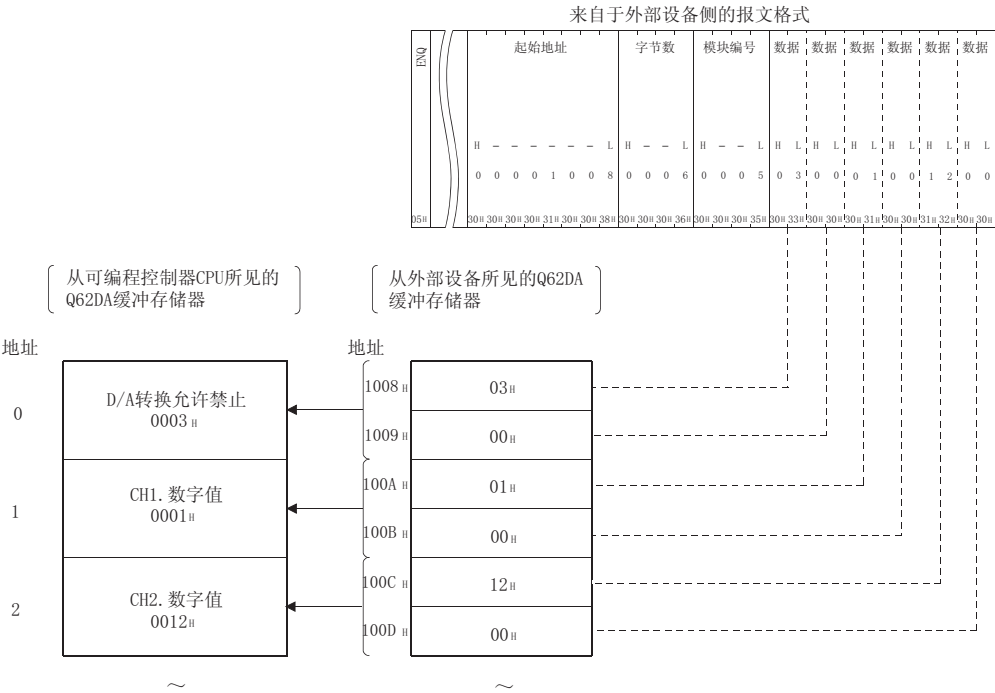
$$\text{起始地址 (16 进制)} = \{(\text{FROM/TO 指令用地址} \times 2)\} \text{ 的 16 进制化} + \text{缓冲存储器起始地址}$$

以下以 Q62DA 为例说明从外部设备对智能功能模块的缓冲存储器进行访问时的数据格式有关内容。

(示例)

指定 Q62DA 的 FROM/TO 指令地址 0(D/A 转换允许/禁止)的情况下

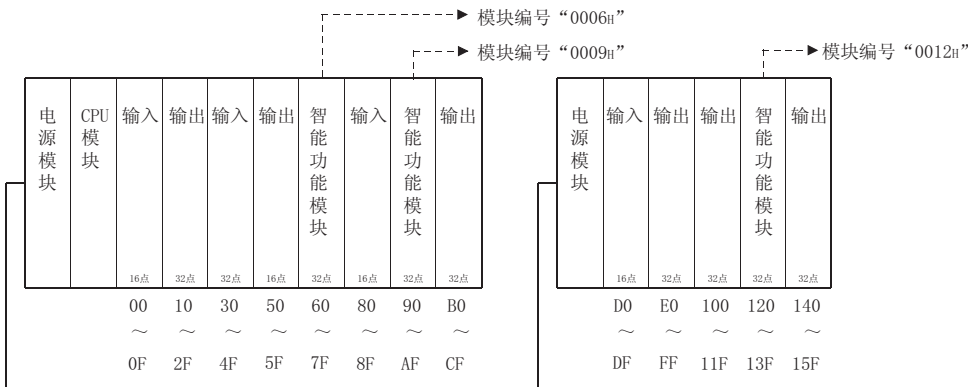
$$\begin{array}{ccc} \text{起始地址} & = & \text{FROM/TO 命令地址} \times 2 + \text{缓冲存储器起始地址} \\ 1008\text{H} & & 0\text{H} \qquad \qquad \qquad 1008\text{H} \end{array}$$



要点

指定报文中的“起始地址”时，各智能功能模块的缓冲存储器的起始地址(上述 Q62DA 的情况下为“1008H”)如 3. 5. 2 项中所示。

- (b) 字节数
是用于对数据读取范围(或写入范围)的地址数×2(字节数)进行指定的数据，以偶数字节进行指定。
- 1) 通过 ASCII 代码进行数据通信时
将地址数×2(2~1920)转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用表示地址数×2(2~1920)的 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
- (c) 模块编号
是用于对数据读取(或写入)智能功能模块进行指定的数据。
模块编号的指定方法如 3) 中所示。
关于允许访问模块、缓冲存储器起始地址以及将允许访问模块安装到 0 插槽中时的模块编号，在 3.5.2 项中进行介绍。
- 1) 通过 ASCII 代码进行数据通信时
将对象智能功能模块的输入输出信号以 4 位表示时的高 3 位转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
(示例)
智能功能模块的输入输出信号为 0080_H ~ 009F_H 的情况下
模块编号变为“0008”，从“0”开始按顺序进行发送。
 - 2) 通过二进制代码进行数据通信时
使用将对象智能功能模块的输入输出信号以 4 位表示时的高 3 位的 2 字节数值，按 Low 字节(L: 位 0~7)、High 字节(H: 位 8~15)的顺序进行发送。
(示例)
智能功能模块的输入输出信号为 0080_H ~ 009F_H 的情况下
模块编号变为 0008_H，按 08_H、00_H 的顺序进行发送。
 - 3) 以下介绍进行智能功能模块的缓冲存储器的读取、写入时的模块编号的指定方法。
 - 对于模块编号，以安装站中分配至对象智能功能模块的起始输入输出信号进行指定。
 - 占用 2 插槽的智能功能模块的情况下，以智能功能模块侧的插槽的起始输入输出信号进行指定。
(占用 1 插槽的智能功能模块的情况下)



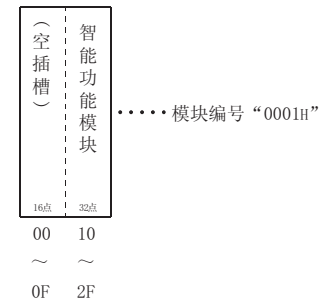
(占用 2 插槽的模块的情况下)

对于占用 2 插槽的智能功能模块，各模块中各插槽的占用点数是确定的。

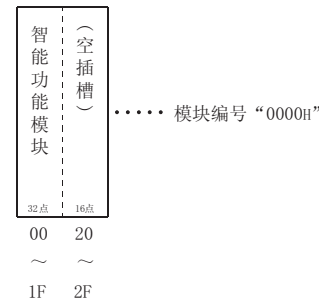
对于模块编号，变为将分配给智能功能模块的插槽侧的起始地址以 4 位进行表示时的高 3 位。

关于各模块插槽的分配有关内容，请参阅各自的智能功能模块用户手册。

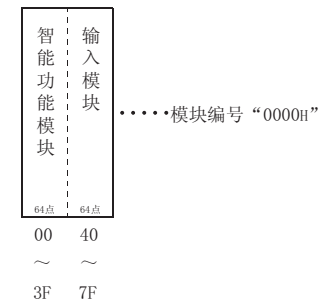
- a) 前半部分插槽分配为空插槽模块的情况下。
(AD72、A84AD 等)



- b) 后半部分插槽分配为空插槽模块的情况下。
(A61LS 等)



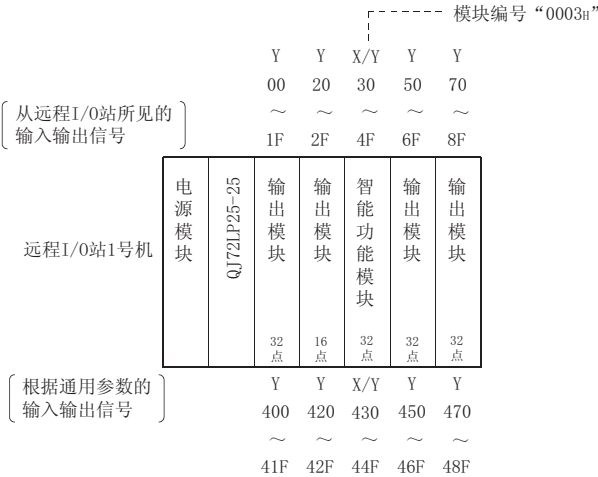
- c) 智能功能模块分配与输入输出分配混合在一起的情况下。
(A81CPU 等)



(网络系统远程 I/O 站的智能功能模块的情况下)

MELSECNET/H、MELSECNET/10 远程 I/O 站的智能功能模块的模块编号为将所有下述“从远程 I/O 站所见的输入输出信号”的起始编号以 4 位表示时的高 3 位。

与 MELSECNET/H、MELSECNET/10 远程 I/O 网络主站中设置的通用参数的内容无关，应以“从远程 I/O 站所见的输入输出信号”进行指定。



对于占用 2 插槽的智能功能模块的模块编号，以前页所示(占用 2 插槽的模块的情况下)的方法进行指定。

- (d) 读取数据、写入数据
- 是对智能功能模块的读取数据或写入数据中，上述 (b) 中所示字节数的数据(最多 1920 字节)的排列。
- 1) 通过 ASCII 代码进行数据通信时
- 将各数据的代码分别转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- (示例)
- 读取数据/写入数据为 12h 的情况下
- 读取数据、写入数据变为“12”，从“1”开始进行发送。
- 2) 通过二进制代码进行数据通信时
- 将各数据的代码从起始部分开始进行发送。

要点

使用指令“0601”、“1601”的情况下，以 QnA 兼容 4C 帧进行通信时，应以下述数据指定报文中的数据指定项目的请求目标模块 I/O 编号及请求目标模块站号。

请求目标模块 I/O 编号：“03FFh”

请求目标模块站号：“00h”

(QnA 兼容 4C 帧格式 1 的情况下)

END	帧识别编号		站号		网络编号		可编程控制器编号		请求目标模块 I/O 编号				请求目标模块站号		本站编号	
	H	L	H	L	H	L	H	L	H	-	-	L	H	L	H	L
05H	F	9	0	0	0	0	F	F	0	3	F	F	0	0	0	0
	46H	39H	30H	30H	30H	30H	46H	46H	30H	33H	46H	46H	30H	30H	30H	30H

3.5.2 可以访问的智能功能模块

以下介绍可通过指令“0601”、“1601”对缓冲存储器进行读取/写入的智能功能模块及报文中指定的“起始地址”及“模块编号”有关内容。

对于下表中不存在的模块的缓冲存储器，应根据软元件存储器的批量读取(指令：0401)及批量写入(指令：1401)，使用智能功能模块的直接软元件进行访问。(参阅附录 1.2)

备注

对于表中所示的“缓冲存储器起始地址”及“安装在 0 插槽中时的模块编号”，是用于报文中指定的“起始地址”及“模块编号”。

(1) 可访问智能功能模块型号

模块型号	缓冲存储器起始地址(16 进制)	安装在 0 插槽中时的模块编号
Q62AD-DGH、Q64AD(-GH)、Q66AD-DG、Q68AD-G、Q68ADV/ADI 型模拟-数字转换模块	1008 _H	0000 _H
Q62DA(-FG)、Q62DAN、Q64DA、Q64DAN、Q66DA-G、Q68DAV/Q68DAI、Q68DAVN/Q68DAIN 型数字 - 模拟转换模块	1008 _H	
Q64AD2DA 模拟输入输出模块	2000 _H	
Q62HLC 型环路控制模块	10000 _H	
Q64TCTT/Q64TCRT 型温度调节模块	1000 _H	
Q61LD 负载传感器输入模块	2000 _H	
Q64TCTTBW/Q64TCRTBW 型温度调节模块	1000 _H	0001 _H
Q64TD、Q64RD 型温度输入模块(功能版本 B)	2000 _H	0000 _H
Q64TD、Q64TDV-GH、Q64RD(-G)型温度输入模块(功能版本 C)	8000 _H	
Q68TD-G-H01、Q68TD-G-H02 型通道间绝缘热电偶输入模块	1008 _H	
Q68RD3-G 型通道间绝缘测温电阻输入模块	1008 _H	
QD51(-R24)型智能通信模块	10000 _H	
QD60P8-G 型通道间绝缘脉冲输入模块	2000 _H	
QD62、QD62E、QD62D 型高速计数器模块	3C _H	0001 _H
QD63P6 型多通道高速计数器模块	2000 _H	
QD64D2 型 4Mpps 对应高速计数模块	2000 _H	
QD70P4/P8 型定位模块	5000 _H	
QD70D4/D8 型定位模块	5000 _H	
QD72P3C3 型计数器功能内置定位模块	5000 _H	
QD75P1/P2/P4、QD75D1/D2/D4、QD75M1/M2/M4、QD75MH1/MH2/MH4 型定位模块	10000 _H	0000 _H
QD81DL96 高速数据记录模块	10000 _H	
QJ61BT11(N)型 CC-Link 系统主站·本地站模块	10000 _H	
QJ61CL12 型 CC-Link/LT 主站模块	01B4 _H	
QJ71C24N(-R2/R4)、QJ71C24(-R2)型串行通信模块	10000 _H	
QJ71AS92 型 AS-i 主站模块	10000 _H	
QJ71DN91 型软元件网络主站·从站模块	10000 _H	
QJ71E71-100/-B5/-B2 型以太网接口模块	10000 _H	
QJ71FL71-T/-B5/-B2-F01 型 FL-net(OPCN-2)接口模块	10000 _H	
QJ71MB91 型 MODBUS 接口模块	10000 _H	
QJ71MES96 型 MES 接口模块	10000 _H	
QJ71MT91 型 MODBUS/TCP 接口模块	10000 _H	
QJ71PB92D 型 PROFIBUS-DP 接口模块	10000 _H	
QJ71PB92V 型 PROFIBUS-DP 主站模块	10000 _H	
QJ71PB93D 型 PROFIBUS-DP 从站模块	10004 _H	
QJ71WS96 型 Web 服务器模块	10000 _H	

模块型号	缓冲存储器起始地址 (16 进制)	安装在 0 插槽中时的模块编号
AD61 (S1) 型高速计数器模块	80 _H	0000 _H
A616AD 型模拟-数字转换模块	10 _H	
A616DAI/DAV 型数字-模拟转换模块	10 _H	
A616TD 型温度-数字转换模块	10 _H	
A62DA (S1) 型数字-模拟转换模块	10 _H	
A68AD (S2) 型模拟-数字转换模块	80 _H	
A68ADN 型模拟-数字转换模块	80 _H	
A68DAV/DAI 型数字-模拟转换模块	10 _H	
A68RD3/4 型温度-数字转换模块	10 _H	0001 _H
A84AD 型模拟/数字转换模块	10 _H	0000 _H
A81CPU 型 PID 控制模块	200 _H	
A61LS 型位置检出模块	80 _H	0001 _H
A62LS (S5) 型位置检出模块	80 _H	0000 _H
AJ71PT32 (S3)、AJ71T32-S3 型 MELSECNET/MINI 主站模块	20 _H	
AJ61BT11 型 CC-Link 系统主站・本地站模块	2000 _H	
AJ71C22 (S1) 型多点链接模块	1000 _H	
AJ71C24 (S3/S6/S8) 型计算机链接模块	1000 _H	
AJ71UC24 型计算机链接模块	400 _H	0001 _H
AD51 (S3) 型智能通信模块	800 _H	
AD51H (S3) 型智能通信模块	800 _H	0000 _H
AJ71C21 (S1) 型终端接口模块	400 _H	
AJ71B62 型 B/NET 接口模块	20 _H	
AJ71P41 型 SUMINET 接口模块	400 _H	
AJ71E71 (S3) 型以太网接口模块	400 _H	0001 _H
AD51FD (S3) 型外部故障诊断模块	280 _H	
AD57G (S3) 型图形控制模块	280 _H	
AS25VS 型视觉传感器模块	100 _H	
AS50VS 型视觉传感器模块	100 _H	
AS50VS-GN 型视觉传感器模块	80 _H	0000 _H
AD59 (S1) 型存储卡接口模块	1800 _H (* ¹)	
AD70 (D) (S2) 型定位模块	80 _H	0001 _H
AD71 (S1/S2/S7) 型定位模块	200 _H	
AD72 型定位模块	200 _H	0000 _H
AD75P1/P2/P3 (S3)、AD75M1/M2/M3 型定位模块	800 _H	
AJ61QBT11 型 CC-Link 系统主站・本地站模块	2000 _H	
AJ71QC24 (N) (R2、R4) 型串行通信模块	4000 _H	
AJ71QE71 (B5) 型以太网接口模块	4000 _H	
A1SD61、A1SD62 (E/D) 型高速计数器模块	10 _H	
A1S62DA 型数字-模拟转换模块	10 _H	
A1S62RD3/4 型温度-数字转换模块	10 _H	
A1S64AD 型模拟-数字转换模块	10 _H	
A1SJ71 (U) C24-R2 型计算机链接模块	400 _H	
A1SJ71 (U) C24-PRF 型计算机链接模块	400 _H	
A1SJ71 (U) C24-R4 型计算机链接模块	400 _H	
A1SJ71E71 (S3) 型以太网接口模块	400 _H	
A1SD70 型 1 轴定位模块	80 _H	
A1SD71-S2/S7 型定位模块	200 _H	0001 _H
A1SD75P1/P2/P3 (S3)、A1SD75M1/M2/M3 型定位模块	800 _H	0000 _H

模块型号	缓冲存储器起始地址(16 进制)	安装在 0 插槽中的模块编号
A1S63ADA 型模拟输入输出模块	10h	0000h
A1S64TCTT(BW)-S1 型温度调节模块	20h	
A1S64TCRT(BW)-S1 型温度调节模块	20h	
A1S64TCRT(BW) 型温度调节模块	20h	
A1S62TCTT(BW)-S2 型温度调节模块	20h	
A1S62TCRT(BW)-S2 型温度调节模块	20h	
A1SJ71PT32-S3 型 MELSECNET/MINI 主站模块	20h	
A1SJ61BT11 型 CC-Link 系统主站・本地站模块	2000h	
A1SJ71QC24(N)(R2) 型串行通信模块	4000h	
A1SJ71QE71-B2/B5 型以太网接口模块	4000h	
A1SJ61QBT11 型 CC-Link 系统主站・本地站模块	2000h	

*1 通过根据可编程控制器 CPU 与 AD59 (S1) 之间的输入输出信号 Y10・Y11 进行存储卡的扇区切换, 可仅对存储卡访问用存储区域进行读取/写入。

(2) 外部设备指定的智能功能模块的起始地址示例

以 Q62DA 为例，外部设备指定的起始地址如下所示。

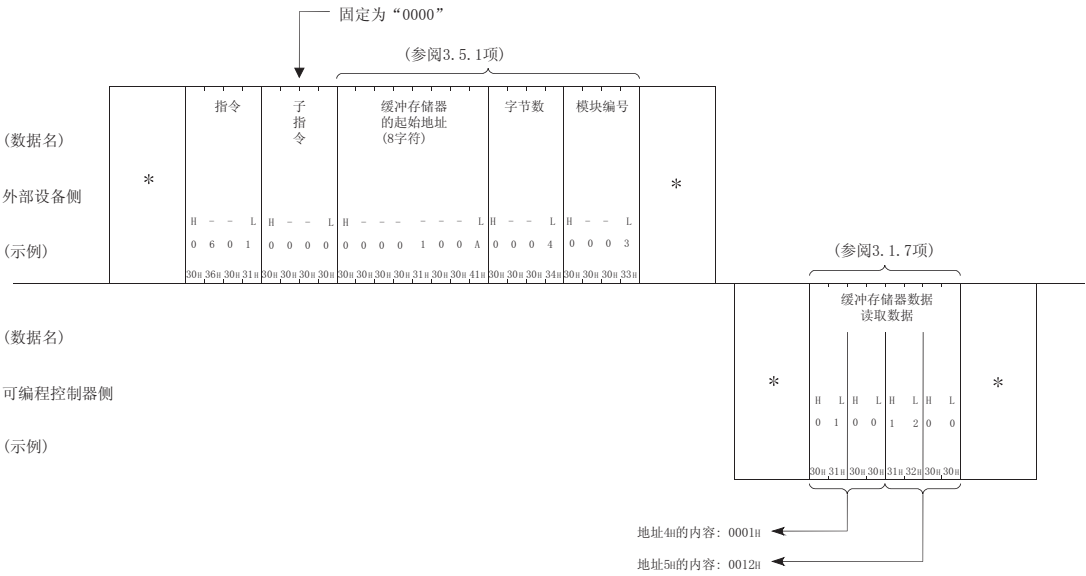
缓冲存储器内容	起始地址	根据 FROM/TO 指令的地址
D/A 转换允许/禁止	1008 _H	0 _H
	1009 _H	
CH. 1 数字值	100A _H	1 _H
	100B _H	
CH. 2 数字值	100C _H	2 _H
	100D _H	
系统区		3 _H ~ 10 _H
偏置・增益调整值指定	102C _H	18 _H

3.5.3 智能功能模块的缓冲存储器读取(指令：0601)

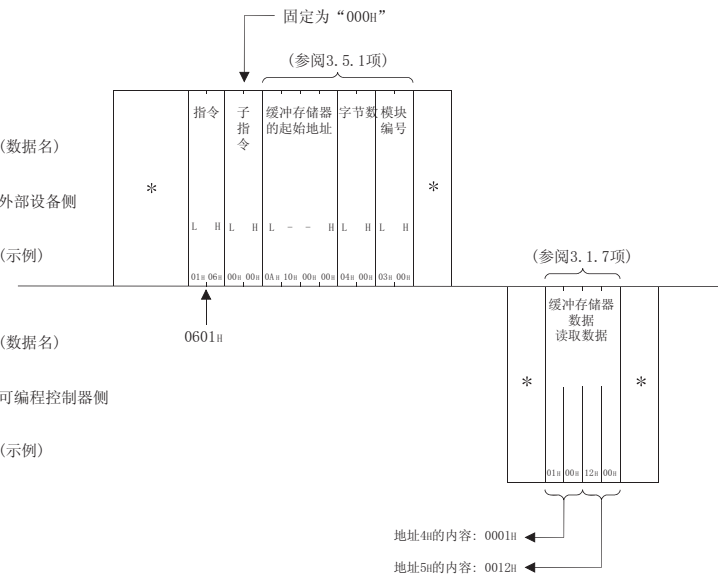
以下对进行智能功能模块的缓冲存储器的读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对输入输出信号为 30H~4FH(模块编号：03H)的 Q62DA 的缓冲存储器地址 1H~2H 的 4 字节进行读取的情况下



(2) 在通过二进制代码进行的通信中，对输入输出信号为 30H~4FH(模块编号: 03H) 的 Q62DA 的缓冲存储器地址 1H~2H 的 4 字节进行读取的情况下



要点

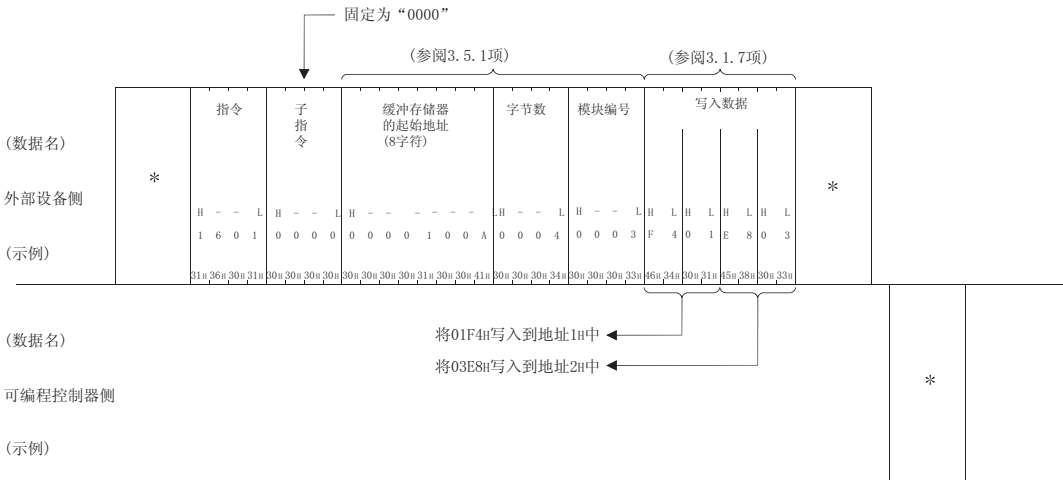
- (1) 起始地址及字节数应在以下范围内进行指定。
- 起始地址 : 对象智能功能模块的地址范围
 - 字节数 : $2(2H) \leq \text{字节数} \leq 1920(780H)$
- (2) 根据智能功能模块，有时 1 个数据会跨越 2~3 字节。关于字节数及读取数据，请参阅相应模块的手册。

3.5.4 智能功能模块的缓冲存储器写入(指令：1601)

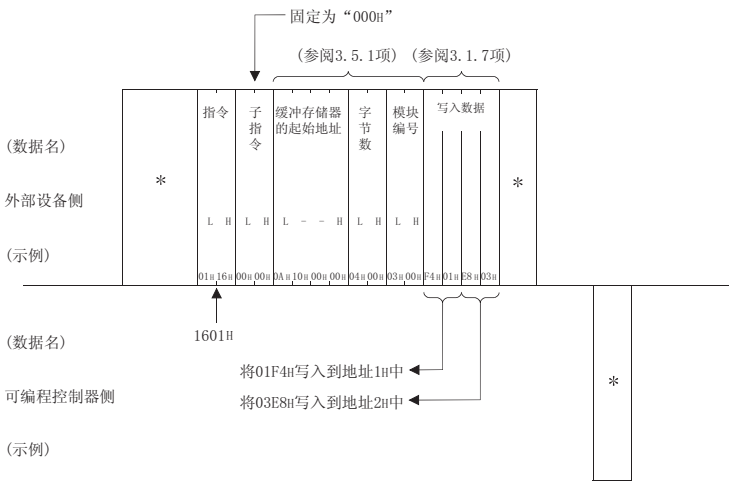
以下对进行智能功能模块的缓冲存储器的写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对输入输出信号为 30H~4FH(模块编号：03H) 的 Q62DA 的缓冲存储器地址 1H~2H 写入 4 字节的情况下



(2) 在通过二进制代码进行的通信中，对输入输出信号为 30H~4FH(模块编号：03H) 的 Q62DA 的缓冲存储器地址 1H~2H 写入 4 字节的情况下



要点
(1) 起始地址及字节数应在以下范围内进行指定。 • 起始地址 : 对象智能功能模块的地址范围 • 字节数 : 2 (2H) ≤ 字节数 ≤ 1920 (780H)
(2) 根据智能功能模块，有时 1 个数据会跨越 2~3 字节。关于字节数及写入数据，请参阅相应模块的手册。

3.6 可编程控制器 CPU 的状态控制

该功能是从外部设备对 Q/L/QnACPU 进行远程 RUN/STOP/PAUSE/RESET 操作、对 Q/L/QnACPU 的软元件存储器进行清除、对 Q/LCPU 的型号进行读取的功能。

3.6.1 指令、控制内容及字符部分的内容

以下对进行可编程控制器 CPU 的状态控制时的指令、控制内容以及控制步骤的字符部分(通过二进制代码进行通信时为数据部分)的有关内容进行说明。

(1) 指令

功能	指令(*1) (子指令)	处理内容	1 次通信中可处理的点数		可编程控制器 CPU 的状态			参阅章节
			Q/L/QnA CPU	Q/L/QnACPU 以外	STOP 中	RUN 中		
						写入允许 设置	写入禁止 设置	
远程 RUN	1001 (0000)	进行远程 RUN(执行运算) 请求。	(1 站)	(禁止)	○	○	○	3. 6. 2 项
远程 STOP	1002 (0000)	进行远程 STOP(停止运算) 请求。						3. 6. 3 项
远程 PAUSE	1003 (0000)	进行远程 PAUSE(停止运算) 请求。 (保持输出状态)						3. 6. 4 项
远程锁存清除	1005 (0000)	STOP 状态时, 进行远程锁存清除(软元件 存储器的清除) 请求。			○	×	×	3. 6. 6 项
远程 RESET	1006 (0000)	STOP 状态时, 进行远程 RESET(开始执行 运算) 请求。			○	×	×	3. 6. 5 项
CPU 型号读取	0101 (0000)	进行可编程控制器 CPU 的型号读取请求。			○	○	○	3. 6. 7 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

*1 根据各个指令可以访问的 CPU 模块有所不同。(参阅 3.2 节)

要点
(1) 通过外部设备执行了远程 RUN/STOP/PAUSE 后, 将 Q/L/QnACPU 的电源进行了 OFF→ON 或复位的情况下, 远程信息将丢失。 (2) Q/L/QnACPU 被实施了系统保护时不能从对象设备进行状态控制。对各请求将返回异常响应/异常结束的响应报文。 (3) 使用 E71 时, 对本站执行可编程控制器 CPU 的状态控制的情况下, 建议通过以下某个操作进行。 1) 使用自动打开 UDP 端口进行可编程控制器 CPU 的状态控制。 2) 在“以太网动作设置”的初始化时机设置中, 使用设置为“常时打开等待”的 Passive 打开连接, 进行可编程控制器 CPU 的状态控制。

(2) 字符部分的内容

以下对外部设备进行 Q/L/QnACPU 的状态控制时的字符部分的内容进行说明。

(a) 模式

是用于强制执行远程 RUN/远程 PAUSE 的数据。

强制执行用于以下情况：由于对 Q/L/QnACPU 进行了远程 STOP/PAUSE 请求的 C24/E71 站或外部设备发生了故障，无法对状态控制下的 Q/L/QnACPU 执行远程 RUN/远程 PAUSE 时，从其它外部设备强制执行远程 RUN/远程 PAUSE。

1) 通过 ASCII 代码进行数据通信时

将下述指定值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位 (“0”) 开始进行发送。

2) 通过二进制代码进行数据通信时

使用下述 2 字节的数值，从 Low 字节(L: 位 0~7) 开始进行发送。

3) 模式的指定内容如下所示。

指定值	处理内容
0001 _H	不强制执行。 从其它设备执行了远程 STOP/PAUSE 时，远程 RUN/远程 PAUSE 无效。
0003 _H	强制执行。 从其它设备执行远程 STOP/PAUSE 时，远程 RUN/远程 PAUSE 有效。(仅远程 RUN、远程 PAUSE 时可以指定)

4) 进行除远程 RUN、远程 PAUSE 以外的状态控制时、使用“0001”或“0001_H”进行发送。

(b) 清除模式

是用于通过远程 RUN 开始 Q/L/QnACPU 的运算时，指定 Q/L/QnACPU 的软元件存储器清除(初始化)处理的数据。

Q/L/QnACPU 进行了指定的清除后，按照参数设置(可编程控制器文件设置→软元件初始值)执行 RUN。

1) 通过 ASCII 代码进行数据通信时

将下述指定值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位 (“0”) 开始进行发送。

2) 通过二进制代码进行数据通信时

使用下述 1 字节的数值进行发送。

3) 清除模式的指定内容如下所示。

指定值	处理内容
00 _H	不对软元件存储器进行清除。
01 _H	对锁存范围外的软元件存储器进行清除。
02 _H	对包含锁存范围在内的全部软元件存储器进行清除。

4) 进行除远程 RUN 以外的状态控制时，不需要进行清除模式的指定。

(c) 固定值

1) 通过 ASCII 代码进行数据通信时，使用“00”进行发送。

2) 通过二进制代码进行数据通信时，使用 1 字节的数值“00_H”进行发送。

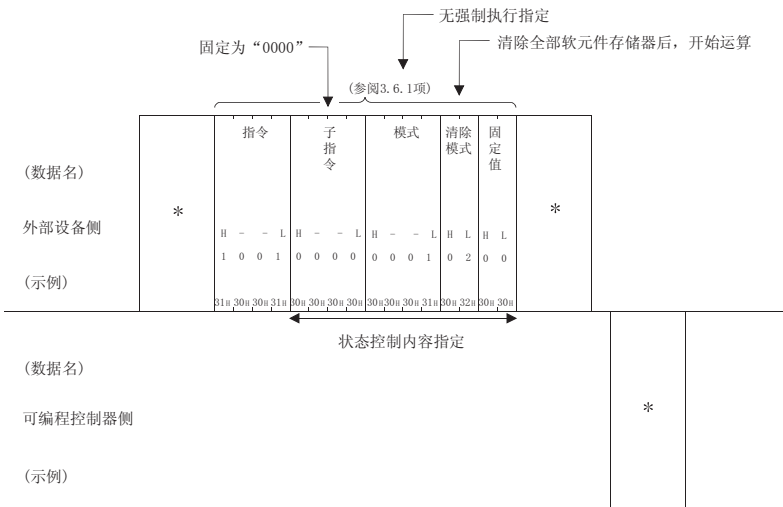
3) 进行除远程 RUN 以外的状态控制时，不需要进行固定值指定。

3.6.2 远程 RUN(指令：1001)

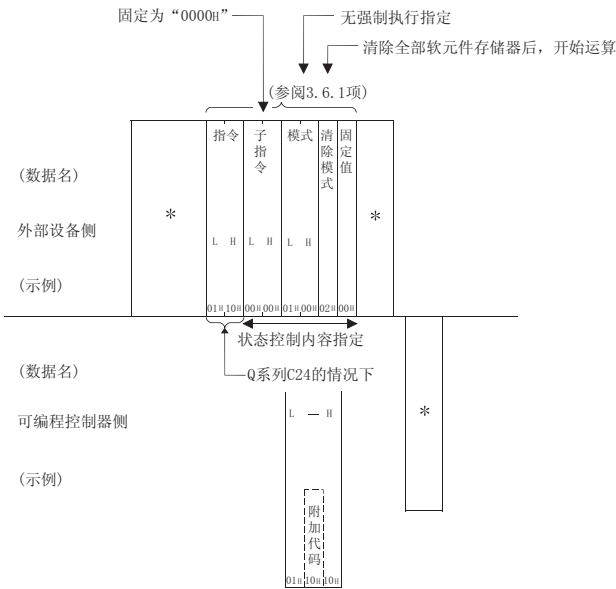
以下对远程 RUN 的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，进行远程 RUN 的情况下



(2) 在通过二进制代码进行的通信中，进行远程 RUN 的情况下



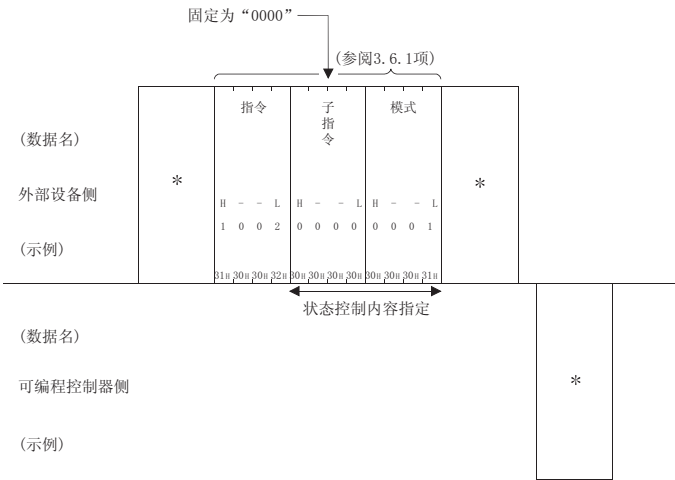
要点
“模式”为无强制执行指定时，已由其它外部设备等对对象 Q/L/QnACPU 进行了远程 STOP/PAUSE 的情况下，即使执行远程 RUN 也不会变为 RUN 状态。

3.6.3 远程 STOP(指令：1002)

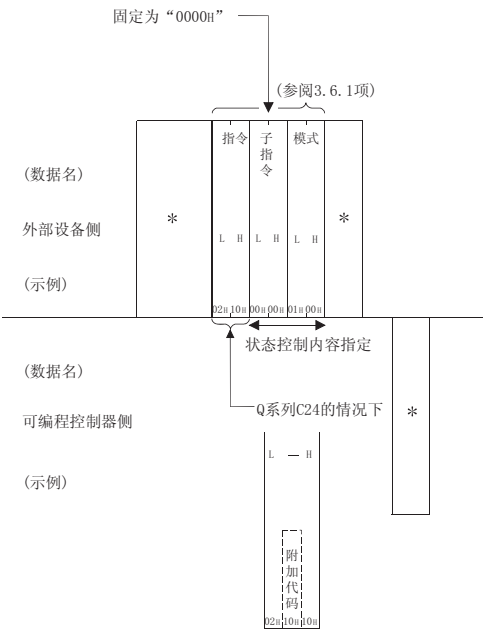
以下对远程 STOP 的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，进行远程 STOP 的情况下



(2) 在通过二进制代码进行的通信中，执行远程 STOP 的情况下

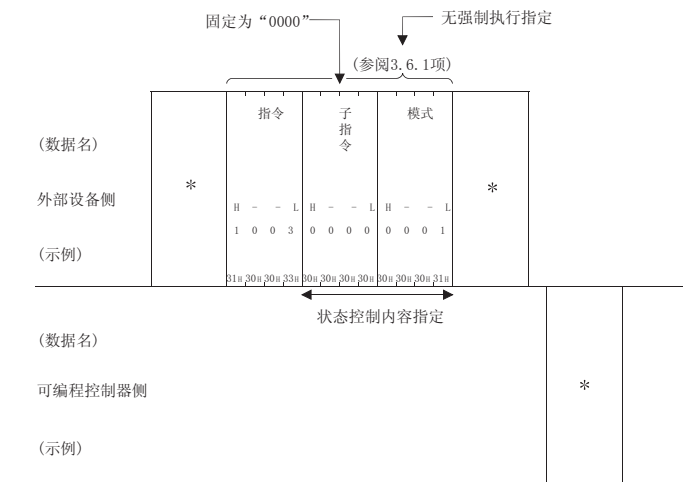


3.6.4 远程 PAUSE(指令：1003)

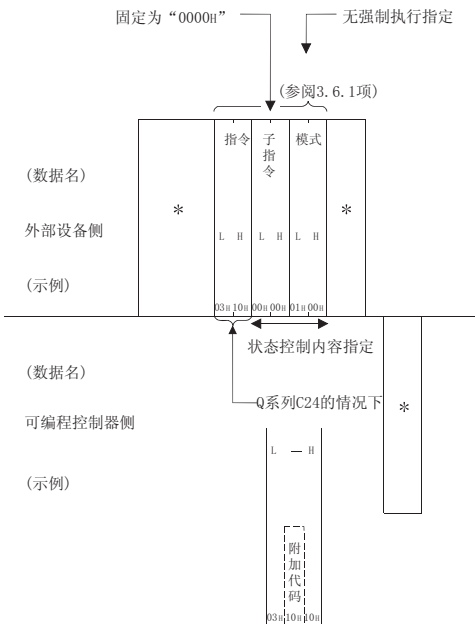
以下对远程 PAUSE 的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，执行远程 PAUSE 的情况下



(2) 在通过二进制代码进行的通信中，执行远程 PAUSE 的情况下



3.6.5 远程 RESET (指令: 1006)

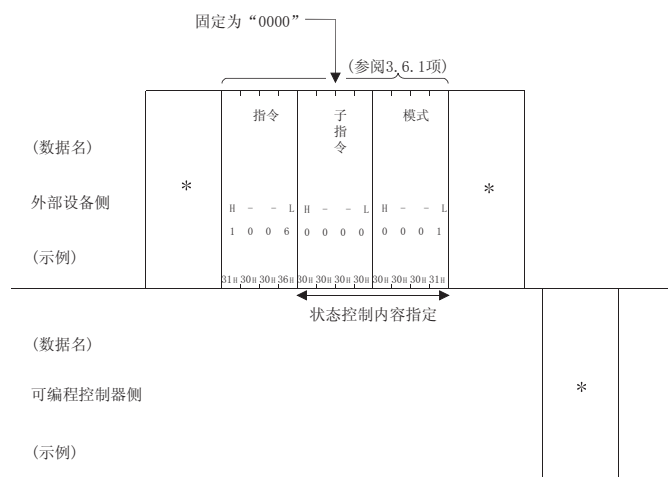
以下对远程 RESET 的控制步骤进行举例说明。

对于控制步骤图中所示*符号部分的数据项目的排列·内容,根据所使用的模块、通信时的帧·格式而有所不同。

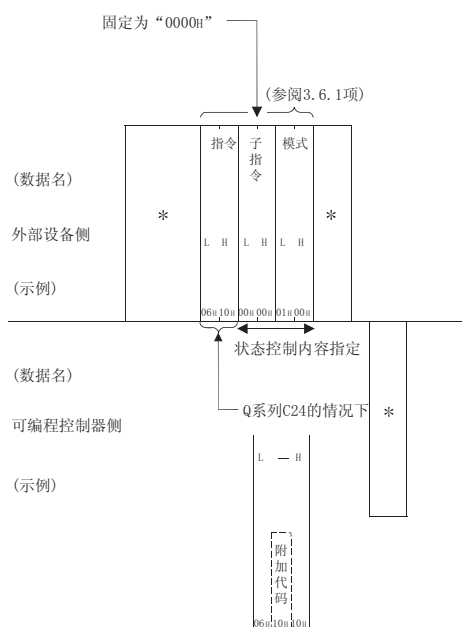
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信, 执行远程 RESET 的情况下



(2) 在通过二进制代码进行的通信中, 执行远程 RESET 的情况下



要点

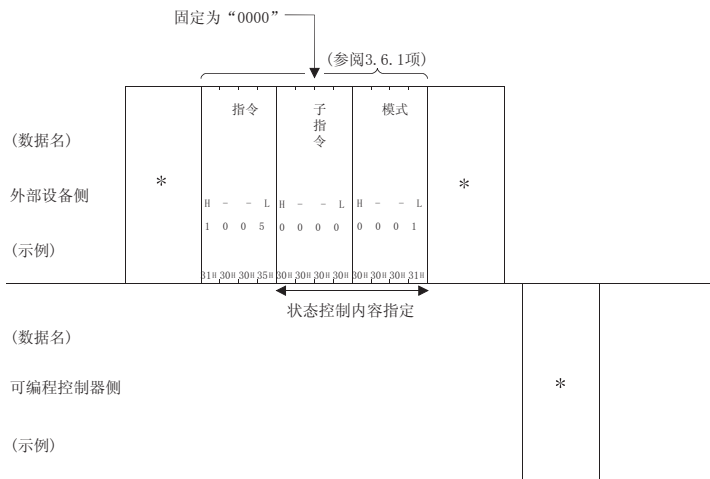
- (1) 由于对象 Q/L/QnACPU 发生出错导致进入 STOP 状态时应执行远程 RESET 操作。
- (2) 在 Q/L/QnACPU 正常动作时可以执行远程 RESET，如果执行了远程 RESET，C24/E71 也将被复位，以与电源投入时相同的状态进行再启动。
- (3) 进行远程 RESET 时，必须在 GX Works2 或 GX Developer 的可编程控制器参数设置(可编程控制器系统设置)中，将远程复位设置为“允许”。

3.6.6 远程锁存清除(指令：1005)

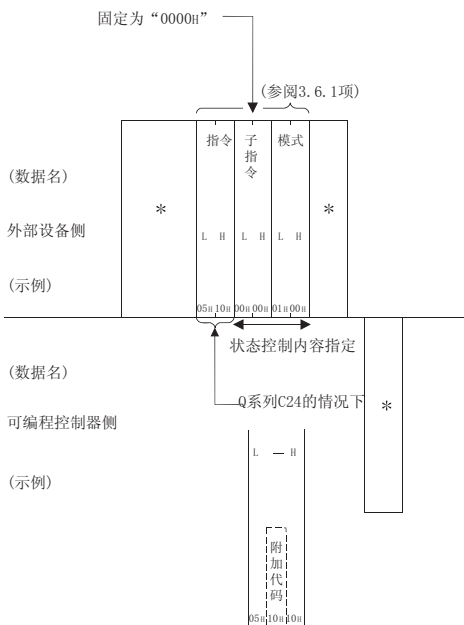
以下对远程锁存清除的控制步骤进行举例说明。
对于控制步骤图中所示*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，执行远程锁存清除的情况下



(2) 在通过二进制代码进行的通信中，执行远程锁存清除的情况下



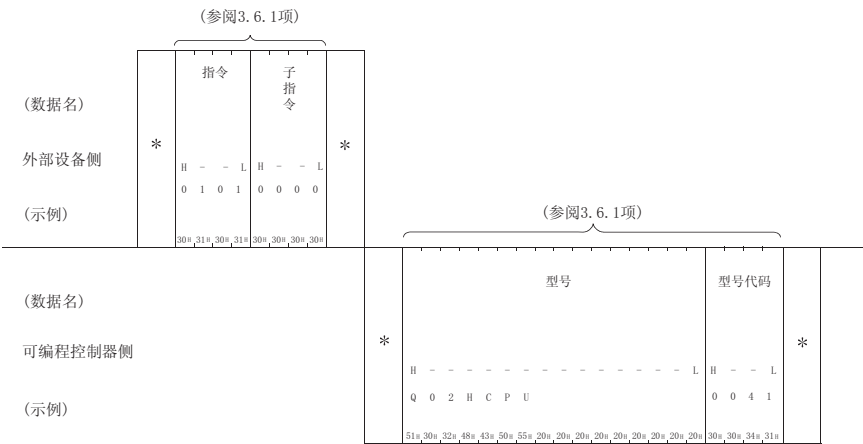
要点
(1) 应在将对象 Q/L/QnACPU 置为 STOP 状态之后再执行远程锁存清除。
(2) 在由于来自于其它外部设备等请求，对象 Q/L/QnACPU 处于远程 STOP/PAUSE 状态时，不能进行远程锁存清除。如果进行了此操作将异常结束。

3.6.7 CPU 型号读取(指令：0101)

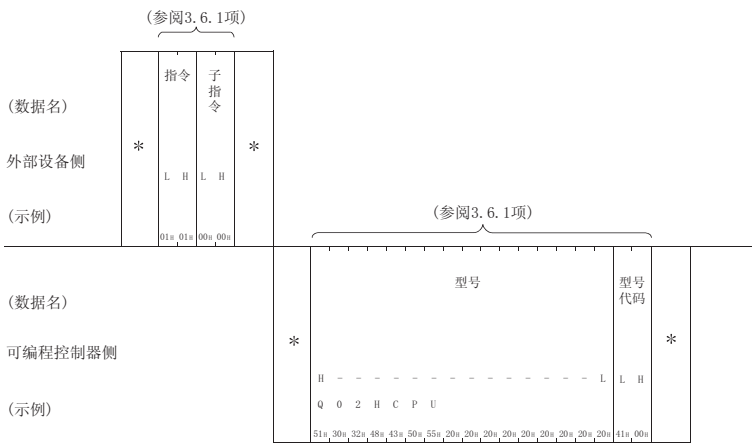
以下对 CPU 型号读取的控制步骤进行举例说明。
对于控制步骤图中所示*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过 ASCII 代码进行通信的情况下



(2) 通过二进制代码进行通信的情况下



* 该功能通过型号代码以及型号(用于监视)读取相应可编程控制器的型号。
读取的可编程控制器的型号代码、型号如下所示。

可编程控制器型号	读取内容
	型号代码(16 进制数)
Q00JCPU	250 _H
Q00CPU	251 _H
Q01CPU	252 _H
Q02CPU、Q02HCPU、Q02PHCPU	41 _H
Q06HCPU、Q06PHCPU	42 _H
Q12HCPU、Q12PHCPU	43 _H
Q25HCPU、Q25PHCPU	44 _H
Q12PRHCPU	4B _H
Q25PRHCPU	4C _H
Q00UJCPU	260 _H
Q00UCPU	261 _H
Q01UCPU	262 _H
Q02UCPU	263 _H
Q03UDCPU、Q03UDECPU	268 _H
Q04UDHCPU、Q04UDEHCPU	269 _H
Q06UDHCPU、Q06UDEHCPU	26A _H
Q10UDHCPU、Q10UDEHCPU	266 _H
Q13UDHCPU、Q13UDEHCPU	26B _H
Q20UDHCPU、Q20UDEHCPU	267 _H
Q26UDHCPU、Q26UDEHCPU	26C _H
Q50UDEHCPU	26D _H
Q100UDEHCPU	26E _H
L02CPU、L02CPU-P	541 _H
L26CPU-BT、L26CPU-PBT	542 _H
QS001CPU	230 _H
LJ72GF15-T2	0641 _H

要点
(1) 对于可编程控制器 CPU 的型号，应通过应型号代码进行判别。 (2) 读取的型号低于指定字节数时，C24/E71 将附加空白(20 _H)后返回。

3.7 驱动器内存的整理(其它站 QnACPU 用)

是外部设备对存储写入了参数及顺控程序的程序文件等的其它站 QnACPU 的驱动器进行以下处理的功能。

(1) 驱动器内存使用状态的读取

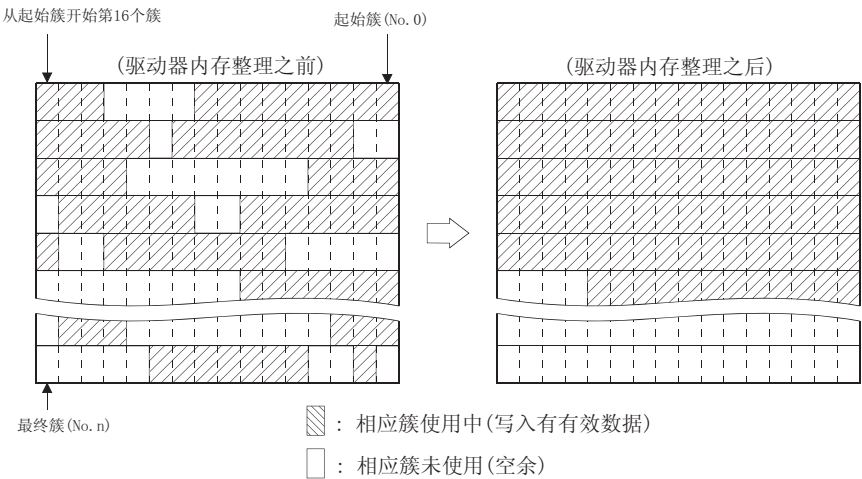
对指定驱动器的驱动器内存的使用状态(簇使用状态)进行确认。

(2) 驱动器内存的碎片整理

驱动器内存中写入了有效数据的内存散乱分布时，以簇单位进行存储器碎片整理以增加连续的空余区域。

要点
簇是指，将文件存储到驱动器内存(存储卡等)中时，由 FAT(*1)管理写入了数据的内存时的最小单位。 QnACPU 的各驱动器的 1 簇的大小如下所示。 • 内置内存 : 4096 字节 • 内置内存以外 : 512 字节 例如，存储卡中写入了 512 字节以下的数据时，作为写入数据的内存将使用 1 个簇的驱动器内存。 写入了 513 字节~1024 字节的数据时，作为写入数据的内存将使用 2 个簇的驱动器内存。 *1 FAT(File Allocation Table) 是操作系统对驱动器内存上的文件位置进行管理的表格。

(驱动器内存的碎片整理示意图)



3.7.1 指令及字符部分的内容

以下对进行驱动器内存整理时的指令以及控制步骤内的字符部分 (通过二进制代码进行通信时为数据部分) 有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
			访问站—2(3.2 项*8 中所 示的 QnACPU)	STOP 中	RUN 中 写入允许 设置	写入禁止 设置	
存储器使用状态 读取	0205 (0000)	对驱动器的簇使用状况进行读取。	(256 簇)	○	○	○	3.7.2 项
存储器碎片整理	1207 (0000)	进行驱动器内存的碎片整理, 增加连续的空余区域。	(1 站)	○	×	×	3.7.3 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下对外部设备进行 QnACPU 的驱动器内存整理时的字符部分的内容进行说明。

(a) 关键字

是用户登录到指定驱动器中的字符串(最多 6 字符), 是用于对该驱动器进行访问允许/禁止的数据。

登录了关键字时, 应指定相同的关键字。

- 1) 通过 ASCII 代码进行数据通信时
原样不变地使用指定驱动器中登录的关键字进行发送。
- 2) 通过二进制代码进行数据通信时
将指定驱动器中登录的关键字转换为 3 字节的二进制代码后使用, 从 Low 字节(L:位 0~7) 开始进行发送。

(示例)

登录关键字	至二进制代码的转换值	发送顺序	备注
“012345”	01 _H 、23 _H 、45 _H	45 _H 、23 _H 、01 _H	从 45 _H 开始按顺序进行发送
“012300”	01 _H 、23 _H 、00 _H	00 _H 、23 _H 、01 _H	从 00 _H 开始按顺序进行发送

- 3) 指定驱动器中未登录关键字时的字符部分的关键字如下所示。
- 通过 ASCII 代码进行数据通信时..... “000000”

• 通过二进制代码进行数据通信时..... 00_H、00_H、00_H

(b) 设置标志

是表示用户是否将指定驱动器中登录的关键字指定到上述 (a) 关键字中的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位 (16 进制数) 后使用, 从高位 (“0”) 开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 设置标志的指定内容如下所示。

指定值	指定内容
00 _H	关键字无效 (以虚拟数进行指定)
01 _H	关键字有效 (对指定驱动器中登录的关键字进行指定)

(c) 驱动器名

是用于对进行驱动器内存的使用状态读取、碎片整理的 QnACPU 的驱动器进行指定的数据。

- 1) 通过 ASCII 代码进行数据通信时
将表示访问目标驱动器的下述数值转换为 ASCII 代码 4 位 (16 进制数) 后使用, 从高位 (“0”) 开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示访问目标驱动器的下述 2 字节的数值、从 Low 字节 (L: 位 0~7) 开始进行发送。
- 3) 驱动器名的指定内容如下所示, 不能指定为除此以外的内容。

指定值	对象驱动器
0000 _H	内置存储器 (内置 RAM)
0001 _H	存储卡 A 的 RAM 区域
0002 _H	存储卡 A 的 ROM 区域
0003 _H	存储卡 B 的 RAM 区域
0004 _H	存储卡 B 的 ROM 区域
000F _H	存储当前使用的参数文件的 (通过 QnACPU 的拨杆开关进行指定) 驱动器

(d) 簇编号

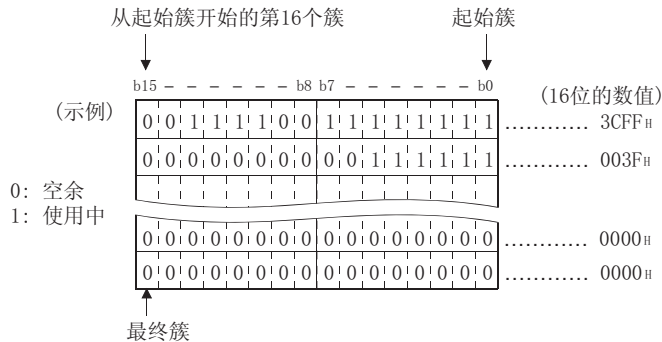
是用于对驱动器内存使用状态的读取范围的起始簇编号进行指定的数据, 以 16 的倍数 (16 进制数的情况下 00_H、10_H、20_H...) 进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将 00_H 以上的簇编号转换为 ASCII 代码 4 位 (16 进制数) 后使用, 从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示 00_H 以上簇编号的 2 字节的数值, 从 Low 字节 (L: 位 0~7) 开始进行发送。
- 3) 进行驱动器内存的碎片整理时, 不需要进行簇编号的指定。

- (e) 读取数
- 是用于对驱动器内存使用状态读取范围的簇数进行指定的数据，以 16 的倍数(16 进制数的情况下 10H、20H...)进行指定。
- 1) 通过 ASCII 代码进行数据通信时
将簇数 10H~100H(16~256)转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用表示簇数 10H~100H(16~256)的 2 字节的数值，从 Low 字节(L:位 0~7)开始进行发送。
 - 3) 进行驱动器内存的碎片整理时，不需要进行读取数的指定。

要点
应根据读取使用状态的驱动器的格式化后的可使用存储容量，对读取数进行指定。 (关于 1 簇的字节数请参阅 3.7 节) 簇数=可使用的存储容量/1 簇的字节数(4096 或 512)

- (f) 簇空余列表
- 是读取驱动器内存的使用状态并返回至外部设备侧的数据(表示簇的使用状态)。
- 1) 通过 ASCII 代码进行数据通信时
将表示使用状态的下述数值转换为 ASCII 代码 n 位(16 进制数)后发送至外部设备侧。(16 簇/4 位)
 - 2) 通过二进制代码进行数据通信时
将表示使用状态的 m 字节的数值发送至外部设备侧。
(16 簇/2 字节)
 - 3) 簇空余列表的内容如下所示。
1 簇/1 位时，各簇的使用状态如下所示。



上图所示使用状态时，返回至外部设备侧的簇空余列表的内容如下所示。

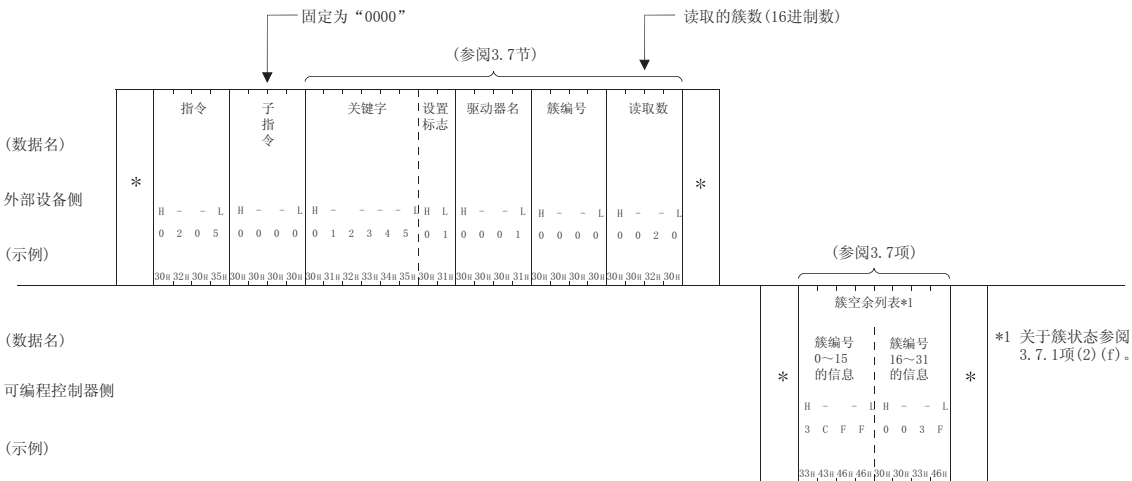
- 在 ASCII 代码的数据通信中，返回 32 簇的情况下
返回“3CF003F”、从“3”开始按顺序进行发送。
 - 在二进制代码的数据通信中，返回 32 簇的情况下
返回 FFH、3CH、3FH、00H，从 FFH 开始按顺序进行发送。
- 4) 进行驱动器内存的碎片整理时，不返回簇空余列表。

3.7.2 驱动器内存使用状态的读取 (指令：0205)

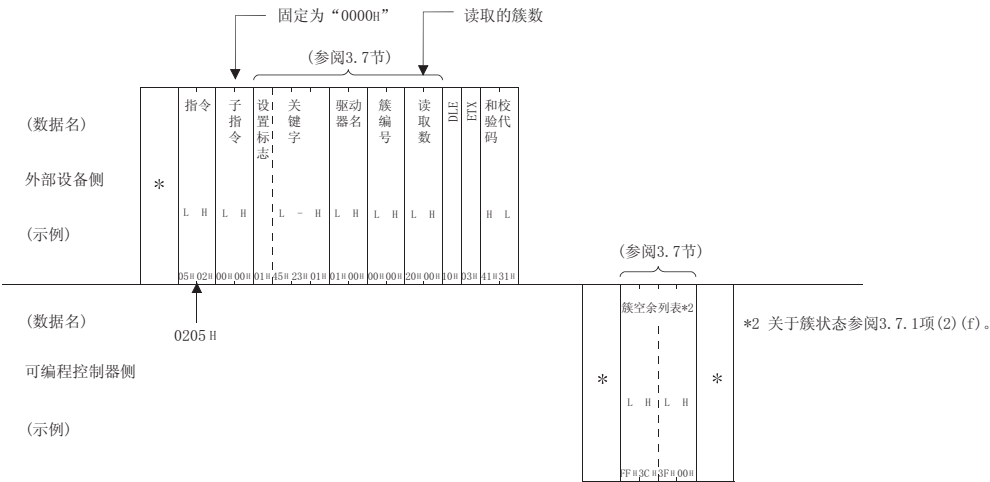
以下对驱动器内存使用状态的读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 以通过 ASCII 代码进行通信时的格式 1，对存储卡 A 的 RAM 区域(驱动器名:01H)的驱动器内存的使用状态读取 32 簇的情况下



(2) 以通过二进制代码进行通信时的格式 5，对存储卡 A 的 RAM 区域(驱动器名:01H)的驱动器内存的使用状态读取 32 簇的情况下



要点

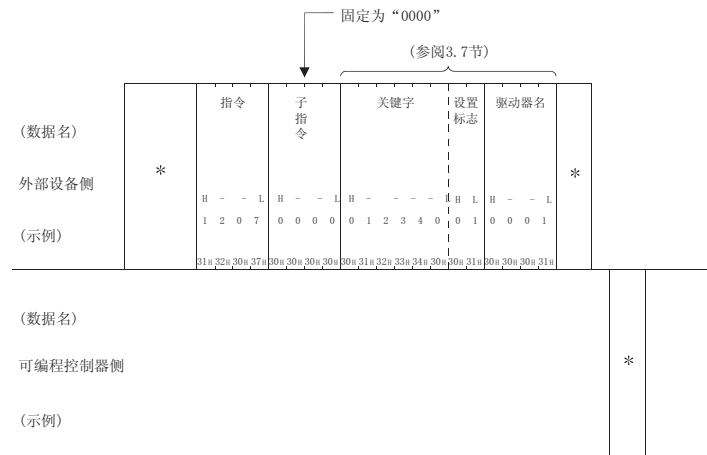
- (1) 对于读取数，应在 10H~100H(16~256)的范围内以 16 的倍数(16 进制数的情况下 10H、20H...)进行指定。
- (2) 创建(新登录)新文件时，需要有创建的文件容量的连续空余区域。
计算指定驱动器的连续空余区域的容量(大小)时，应通过本驱动器内存使用状态的读取对连续的空余簇数(0FF 位的排列数)进行确认。
连续空余区域容量(大小) = 连续空余簇数 × 4096 或 512(字节)
连续空余区域不足时，应按 3.7.3 项所示进行存储器的碎片整理。

3.7.3 驱动器内存的碎片整理(指令：1207)

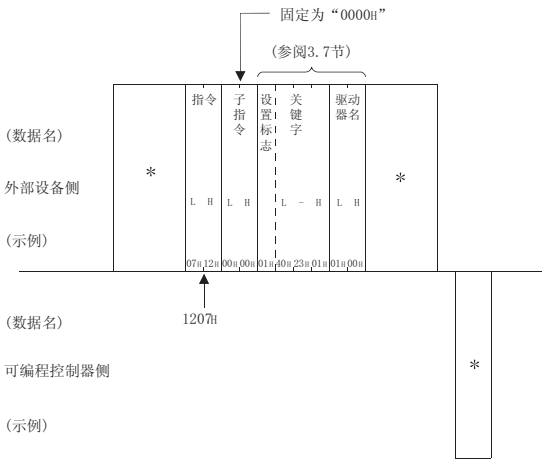
以下对进行驱动器内存碎片整理的控制步骤进行举例说明。
对于对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 以通过 ASCII 代码进行通信时的格式 1，对存储卡 A 的 RAM 区域(驱动器名:01H)的驱动器内存进行碎片整理的情况下



(2) 以通过二进制代码进行通信时的格式 5，对存储卡 A 的 RAM 区域(驱动器名:01H)的驱动器内存进行碎片整理的情况下



要点
(1) 驱动器内存的碎片整理应在以下情况下进行。 - 指定站 QnACPU 处于 STOP 状态时。 - 驱动器内存的使用状态(通过指令 0205 进行确认)散乱，无法存储文件时。
(2) 在以下情况下将变为出错状态，并返回异常响应。 1) QnACPU 被实施了系统保护时。 2) 未指定指定驱动器中登录的关键字时。 3) 驱动器内存有异常时。(有坏簇等) 4) 使用 IC 存储卡读/写器进行了以下操作时。 - 创建了子目录时。 1 个文件未存储在 1 个连续区域中时。

3.8 文件控制

是对 Q/L/QnA 内进行文件读取、新登录、删除等的功能。
该功能用于以下情况。

- 从对象设备中读取参数及顺控程序等。
- 将从 Q/L/QnACPU 中读取的文件根据控制内容从对象设备中写入到 Q/L/QnACPU 中。

关于 Q/L/QnACPU 中存储的文件的文件名、扩展名、存储位置等，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。

要点
在文件控制中，有时会看到所使用的 CPU 模块的用户手册(功能解说/程序基础篇)中记载的文件以外的文件，这是系统用的文件，因此请勿访问。

3.8.1 指令及字符部分的内容

以下对进行文件控制时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)有关内容进行说明。

(1) 指令

Q/L/QnACPU 用的文件控制指令如下所示。

(a) Q/LCPU 用文件控制指令

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	访问站(对象)		可编程控制器 CPU 的状态			参阅章节
			访问站-1 (3. 2 项*7 中所示的 Q/L CPU)	QCPU	LCPU	STOP 中	RUN 中		
							写入允许设置	写入禁止设置	
目录・文件信息读取	1810 (0000)	对文件列表信息进行读取。	(36 个)	○	○	○	○	○	3. 8. 5 项
目录・文件信息搜索	1811 (0000)	对指定文件的文件编号进行读取。	(1 个)	○	○			○	3. 8. 6 项
新建文件	1820 (0000)	对指定文件的存储区域进行预留。	(1 个)	○	○			×	3. 8. 10 项
文件删除	1822 (0000)	删除文件。	(1 个)	○	× (* 1)			×	3. 8. 12 项
	1822 (0004)			×	○				
文件复制	1824 (0000)	对指定文件进行复制。	(1 个)	○	× (* 1)			C24: × E71: ○	3. 8. 13 项
	1824 (0004)			×	○				
文件属性更改	1825 (0000)	对文件的属性进行更改。	(1 个)	○	× (* 1)			×	3. 8. 15 项
	1825 (0004)			×	○				
文件创建日期更改	1826 (0000)	对文件创建日期时间进行更改。	(1 个)	○	○			×	3. 8. 14 项

*1 对象 CPU 模块中未设置关键字(口令)的情况下，指令将被执行并返回正常应答。

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	访问站(对象)		可编程控制器 CPU 的状态			参阅章节
			访问站-1 (3.2 项*7 中所示的 Q/L CPU)	QCPU	LCPU	STOP 中	RUN 中		
							写入允许设置	写入禁止设置	
打开文件	1827 (0000)	为了防止来自于其它设备的内容更改而对文件进行锁定。	(1 个)	○	× (*1)	○	○	○	3.8.7 项
	1827 (0004)			×	○				
文件读取	1828 (0000)	对文件的内容进行读取。	1920 字节	○	○			○	3.8.9 项
文件写入	1829 (0000)	将内容写入到文件中。	1920 字节	○	○			×	3.8.11 项
文件关闭	182A (0000)	对打开文件时锁定的文件进行解除。	(1 个)	○	○			○	3.8.8 项

上表的可编程控制器 CPU 的状态栏中的○表示可以执行。

*1 对象 CPU 模块中未设置关键字(口令)的情况下，指令将被执行，并进行正常响应。

(b) QnACPU 用的文件控制指令

功能		指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
				访问站-2(3. 2 项*8 中所 示的 QnACPU)	STOP 中	RUN 中		
						写入允许 设置	写入禁止 设置	
文件信息列表读取	无索引	0201 (0000)	对文件列表(文件名、最后编辑日期、文件大小)进行读取。	(36 个)	○	○	○	3. 8. 16 项(1)
	有索引	0202 (0000)	通过文件附带的索引,对文件列表进行读取。	(16 个)				3. 8. 16 项(2)
	文件编号使用状况	0204 (0000)	对文件编号的使用状况进行读取。	(256 个)				3. 8. 16 项(3)
文件信息更改	最终编辑日時更改	1204 (0000)	对文件的最终编辑日期时间进行更改。	(1 个)	○	○	×	3. 8. 24 项(1)
	文件名、大小更改	1204 (0001)	对文件名、文件大小进行更改。					3. 8. 24 项(2)
	批量更改	1204 (0002)	对文件名、文件大小、最后编辑日期进行更改。					3. 8. 24 项(3)
文件搜索(文件的有无)		0203 (0000)	对指定文件的文件编号、文件大小进行读取。	(1 个)	○	○	○	3. 8. 17 项
文件内容读取(批量读取)		0206 (0000)	对文件内容进行读取。	960 字节	○	○	○	3. 8. 9 项
新登录(文件名登录)		1202 (0000)	对指定文件名的区域进行预留。	(1 个)	○	○	×	3. 8. 20 项
文件内容写入	任意数据(批量写入)	1203 (0000)	向文件写入指定数据(n 字节)。	960 字节	○	○	×	3. 8. 21 项(1)
	同一数据(FILL)	1203 (0001)	向文件写入 n 字节的指定数据(1 字)。	(文件大小)				3. 8. 21 项(2)
文件锁定	登录	0808 (0001)	为了防止来自于其它设备的内容更改,对文件锁定进行登录。或对登录进行解除。	(1 个)	○	○	○	3. 8. 18 项
	解除	0808 (0000)						
文件复制		1206 (0000)	将原有文件的内容写入到新登录的文件中。	480 字节	○	○	○	3. 8. 23 项
文件删除		1205 (0000)	将文件删除。	(1 个)	○	○	×	3. 8. 22 项

上表的可编程控制器 CPU 的状态栏中的○表示可以执行。

(2) 字符部分的内容

以下对外部设备进行 Q/L/QnACPU 的文件控制时，3.8.5 项以后所示的报文的字符部分的内容进行说明。

(a) 关键字(口令)

1) QCPU 的情况下(4 字符)

用户对至对象存储器的文件(程序文件、软元件注释文件、软元件初始值文件)的访问进行允许/禁止设置。

对各文件进行写入禁止或读取/写入禁止的设置。

2) LCPU 的情况下(4~32 字符)*1

用户对至对象存储器的文件(程序文件、软元件注释文件、软元件初始值文件、参数文件、标签程序可逆转换信息)的访问进行允许/禁止设置。

可对各文件分别进行读取、写入、读取/写入的口令。

*1 不足 32 字符的情况下附加空格(代码:20H)。

3) QnACPU 的情况下(6 字符)

用户对至指定的驱动器的访问进行允许/禁止设置。

登录了关键字时，应指定相同的关键字。

关于字符部分的内容，请参阅 3.7.1 项(2)(a)。

(b) 设置标志

是用于表示用户是否将指定驱动器中登录的关键字指定为上述(a)的关键字的数据。

关于字符部分的内容，请参阅 3.7.1 项(2)(b)。

- (c) 驱动器名、驱动器编号
是用于对进行文件控制的 Q/L/QnACPU 的驱动器进行指定的数据。
关于 QnACPU 的驱动器名的指定方法及对应的驱动器有关内容请参阅 3.7.1 项。
以下介绍 Q/LCPU 的驱动器编号的指定方法及对应的驱动器有关内容。

- 1) 通过 ASCII 代码进行数据通信时
将表示访问目标驱动器的下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示访问目标驱动器的下述 2 字节的数值，从 Low 字节(L:位 0~7)开始进行发送。
- 3) 驱动器编号及对象驱动器如下所示，不能指定除此以外的内容。

<Q/QnACPU 对象的情况下>

指定值	对象驱动器	指定值	对象驱动器
0000 _h	程序存储器	0003 _h	标准 RAM
0001 _h	SRAM 卡	0004 _h	标准 ROM
0002 _h	Flash 卡、ATA 卡	-	-

<LCPU 对象的情况下>

指定值	对象驱动器	指定值	对象驱动器
0000 _h	程序存储器	0003 _h	标准 RAM
0001 _h	-	0004 _h	标准 ROM
0002 _h	SD 存储卡	-	-

- (d) 文件编号
是通过下述文件名及扩展名指定的文件被登录(写入)到可编程控制器 CPU 时的登录号，或是用于对登录到可编程控制器 CPU 时的登录号进行指定的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述文件编号转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
(示例)
1F_h 的情况下变为“001F”，从“0”开始按顺序进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示下述文件编号的 2 字节的数值，从 Low 字节(L:位 0~7)开始进行发送。
(示例)
1F_h 的情况下变为 001F，按 1F_h、00_h 的顺序进行发送。
- 3) 文件编号通过以下方式之一进行指定。

指定值	内容	指定内容
01 _h ~ 100 _h	文件编号	知道文件编号时进行此指定。
FFFF _h	文件编号不明	使 C24/E71 检索文件编号时进行此指定。 (从 C24/E71 至可编程控制器 CPU 的读取、 写入请求将导致 1 个顺控程序扫描时间以上的延迟。)

- 4) 对于已登录的文件的文件编号，可通过 3.8.6 项、3.8.16 项、3.8.17 项中所示的功能进行确认。
QnACPU 的情况下，对于新登录时未使用的文件编号，可以通过 3.8.16 项(3)中所示的文件编号使用状况读取功能进行确认。(Q/LCPU 不能进行确认。)

- (e) 文件请求数、全登录文件数、文件信息数
是对文件信息进行读取时表示用户请求的文件数、指定驱动器中登录的文件数、文件信息返回的文件数的数据。
 - 1) 通过 ASCII 代码进行数据通信时
将相应功能说明项中所示的数值分别转换为 ASCII 代码 4 位 (16 进制数) 后使用, 从高位 (“0”) 开始进行发送。
 - 2) 通过二进制代码进行数据通信时
分别使用相应功能说明项中所示的 2 字节的数值, 从 Low 字节 (L: 位 0~7) 开始进行发送。
- (f) 文件名字符数、文件名、扩展名、属性
是用于指定进行读取、写入、登录等的文件的数据。
 - 1) 关于 Q/L/QnACPU 中可指定的文件的文件名、扩展名、存储位置等, 请参阅所使用的用户手册 (功能解说/程序基础篇)。

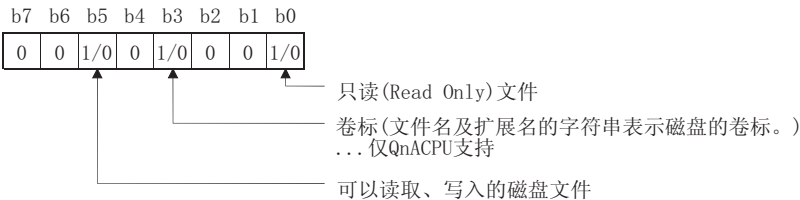
要点
在文件控制中, 有时会看到所使用的 CPU 模块的用户手册 (功能解说/程序基础篇) 中记载的文件以外的文件, 这是系统用的文件, 因此请勿访问。

- 2) 新建文件或更改文件名时, 通过 GX Works2 或 GX Developer 按照文件名附加方法规则对文件名 (最多半角 8 字符)、扩展名 (半角 3 字符) 进行指定。
- 3) 文件名字符数、文件名、扩展名、属性的处理如下所示, 数据通信时的排列根据指令而有所不同, 分别通过 ASCII 代码进行通信时与通过二进制代码进行通信时相同。
 - (模式 1) 用于 Q/LCPU 文件
 - 访问对象文件按下述排列进行指定。
文件名+ “.” +扩展名
 - 将上述指定的字符数以文件名字符数进行指定。
 - 文件的名称为 ABC.QPG 情况下的指定示例如下所示。
文件名字符数: 7
文件名: “ABC.QPG”
 - (模式 2) 主要用于 QnACPU 文件
 - 访问对象文件按下述排列进行指定。
文件名+扩展名+属性
不足半角 8 字符时, 在文件名的后面附加空格 (代码: 20H), 使其达到半角 8 字符。
 - 文件的名称为 ABC.QPG 的情况下的指定示例如下所示。
文件名: “ABC QPG”
(“ ” : 表示空格)

- 4) 用户新创建的文件的属性以及虚拟指定时的属性使用 20h(可以读取/写入的磁盘文件)。
用户可对该属性进行更改。(参阅 备注)
(Q/LCPU 用文件:参阅 3.8.15 项, QnACPU 用文件:参阅 3.8.24 项)
* 对于现有文件的属性, 可以通过如下所示的功能进行确认。
Q/LCPU 用文件:目录・文件信息的读取功能(参阅 3.8.5 项)
QnACPU 用文件:文件信息列表的读取功能(参阅 3.8.16 项)
- 5) 关于数据通信时表示文件名称的数据的发送方法, 请参阅各指令的说明项。

备注

以下介绍 Q/L/QnACPU 的各磁盘中存储的文件属性的阅读方法有关内容的概要。
表示属性的数值的各个位均带有含义。
通过将相应位置为 0N(1), 附加对应的属性。



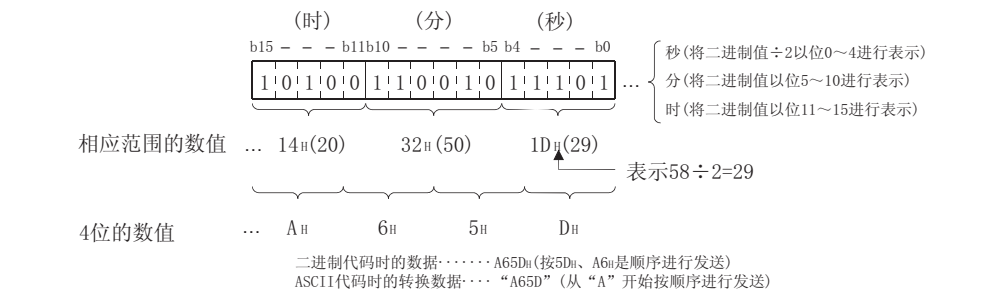
- * 对于未进行属性更改的用户创建文件, 附加了可以读取、写入的磁盘文件的属性。
对于用户创建文件的属性, 可以在 01h(只读(Read Only)文件)・20h(可以读取、写入的磁盘文件)之间进行更改。(参阅 3.8.15 项、3.8.24 项)
- (g) 最后编辑时间、最后编辑日期
是表示当前内容被登录时的日期的数据。
 - 1) 通过 ASCII 代码进行数据通信时
将下述数值分别转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位(时刻, 年)开始进行发送。
以虚拟键值进行指定时发送“0000”。
 - 2) 通过二进制代码进行数据通信时
分别使用下述 2 字节的数值, 从 Low 字节(L:位 0~7)开始进行发送。
以虚拟键值进行指定时发送 0000h。

3) 表示时间、日期的数值的内容及发送顺序如下所示。

• 时间(时、分、秒)

(示例)

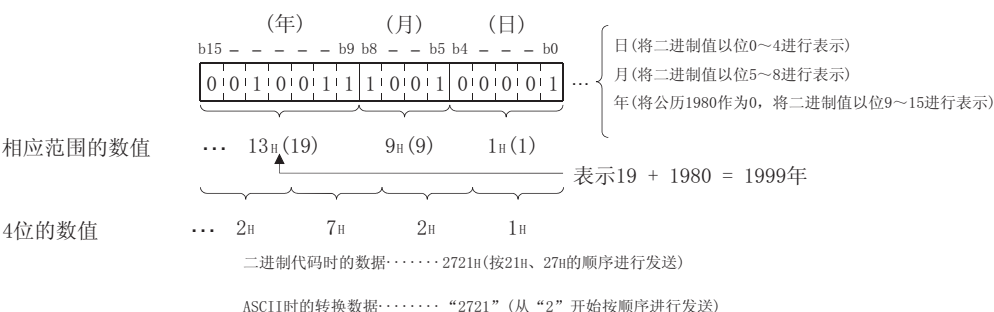
20 时 50 分 58 秒的情况下



• 日期(年、月、日)

(示例)

1999 年 9 月 1 日的情况下



(h) 文件大小

是将当前文件的容量以字节数表示的数据。

1) 通过 ASCII 代码进行数据通信时

将 2 字的数值转换为 ASCII 代码 8 位(16 进制数)后使用, 从高位开始进行发送。

(示例)

文件大小为 7168 字节的情况下

变为“00001C00”, 从起始“0”开始按顺序进行发送。

↑ 起始

2) 通过二进制代码进行数据通信时

使用 2 字的数值, 从 Low 字节(L:位 0~7)开始进行发送。

(示例)

文件大小为 7168 字节的情况下

变为 00001C00H, 按 00H、1CH、00H、00H 的顺序进行发送。

(i) 索引

是通过 Q/L/QnACPU 对应的 GX Works2 或 GX Developer 附加到指定文件上的索引。(最多 32 字符(半角时))

1) 通过 ASCII 代码进行数据通信时

从各个起始字符开始进行发送。

索引不足 32 字符时, 附加空格(代码:20H)。

(示例)

登录时的索引为“1 生产线-可编程控制器 5”的情况下

变为“1 生产线-可编程控制器 5...”, 从“1”开始按顺序进行发送。

2) 通过二进制代码进行数据通信时

将索引的各字符的字符代码作为二进制值使用, 从起始字符开始进行发送。

索引不足 32 字符时, 附加 20H 使其达到 32 字符。

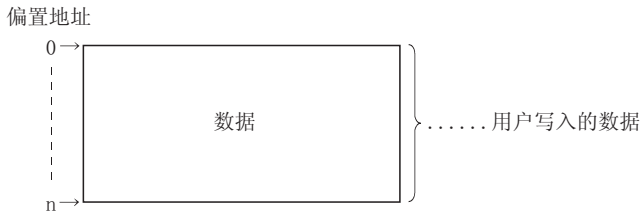
(示例)

登录时的索引为“1 生产线-可编程控制器 5”的情况下

变为 31H、D7H、B2H、DDH、2DH、50H、43H、35H、20H、20H、..., 从 31H 开始按顺序进行发送。

(j) 偏置地址

是用于对文件数据读取、写入范围的起始地址进行指定的数据。
将从各文件的起始(偏置地址: 0H)开始的地址(1 地址/1 字节)以偶数地址进行指定。



- 1) 通过 ASCII 代码进行数据通信时
将相应功能说明项中所示的地址转换为 ASCII 代码 8 位(16 进制数)后使用, 从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示相应功能说明项中所示的地址的 4 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。
- 3) 对于可指定的偏置地址, 应通过 3. 8. 16 项中所示的文件信息列表的读取功能对文件的大小(字节数)进行确认后, 从该大小求出偏置地址(0H ~nH)。

(k) 读取字节数、写入字节数

是用于对文件数据读取、写入范围的字节数进行指定的数据, 以 1 地址/1 字节进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将相应功能说明项中所示的数值转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用相应功能说明项中所示的 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。

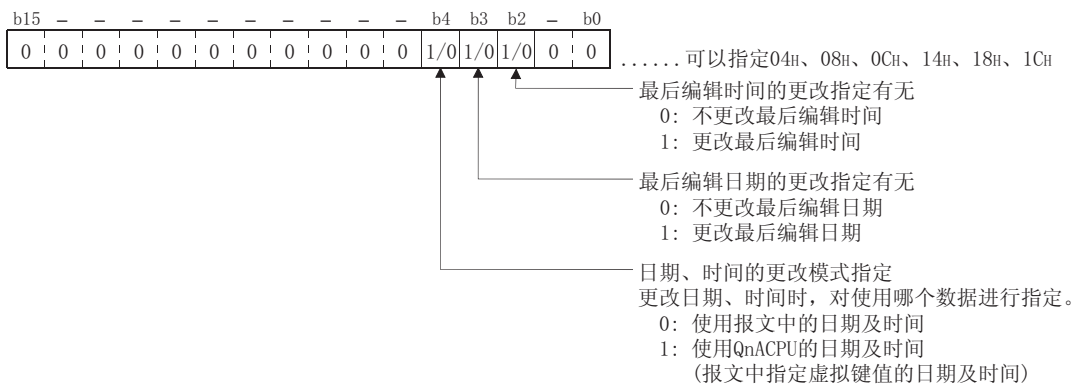
(l) 读取数据、写入数据(用于批量读取、批量写入功能)

是从 Q/L/QnACPU 的文件中读取的数据或写入到 Q/L/QnACPU 的文件中的数据, 从偏置地址开始的排列。

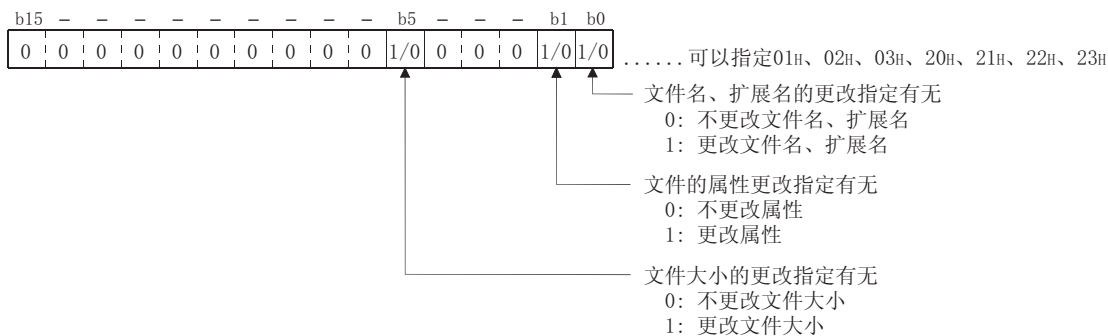
- 1) 通过 ASCII 代码进行数据通信时
将 1 字节(1 地址)转换为 ASCII 代码 2 位(16 进制数)的数据, 以指定字节数从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
将 1 个地址作为 1 个字节, 对指定字节数进行发送。
- 3) 读取时, 以从 Q/L/QnACPU 中读取时的排列原样不变地保存到外部设备中。
写入时, 以从 Q/L/QnACPU 中读取时的排列原样不变地进行指定。

- (m) 写入数据(同一数据写入功能用): 用于 QnACPU 文件
是向现有的 QnACPU 的文件写入同一数据时的同一数据写入功能用的数据。
 - 1) 通过 ASCII 代码进行数据通信时
将 1 字的数值转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用 1 字的数值, 从 Low 字节(L: 位 0~7)开始进行发送。
- (n) 容量
是新登录文件时用于在指定磁盘上预留文件区域的数据, 以字节数进行指定。
 - 1) 通过 ASCII 代码进行数据通信时
将预留的指定文件用的区域以 2 字表示时的数值转换为 ASCII 代码 8 位(16 进制数)后使用, 从高位开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用将预留的指定文件用的区域以 2 字表示时的数值, 从 Low 字节(L: 位 0~7)开始进行发送。
 - 3) 可以从外部设备对与现有文件相同内容的文件进行新登录。
对于对象现有文件的大小, 需要通过文件信息的读取功能进行确认。
(参阅 3.8.5 项、3.8.16 项、3.8.17 项)
- (o) 固定值
 - 1) 通过 ASCII 代码进行数据通信时, 使用“0000”进行发送。
 - 2) 通过二进制代码进行数据通信时, 使用 2 字节的数值“0000H”进行发送。

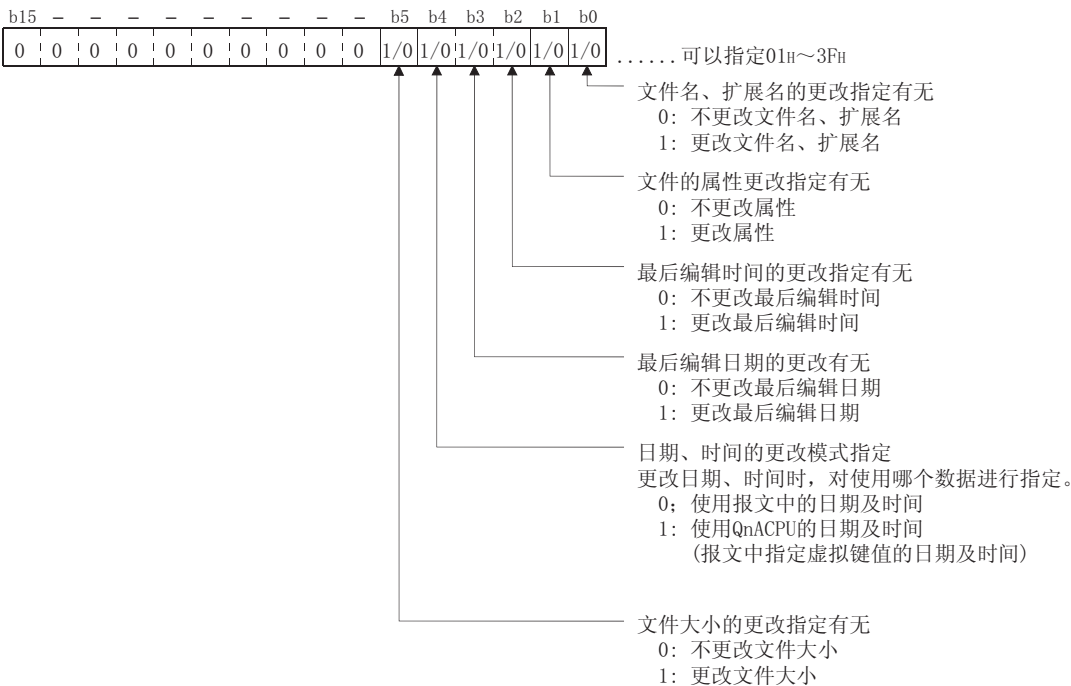
- (p) 更改模式(用于文件名、文件大小的更改)：用于 QnACPU 文件
是对现有文件的信息(文件名、大小、创建日期、时间)进行更改时，用于指定对哪个信息进行更改的数据。
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L：位 0~7)开始进行发送。
 - 3) 更改模式的指定值及内容如下所示。
 - 对文件创建日期、时间进行更改的情况下
(指令：1204，子指令：0000)



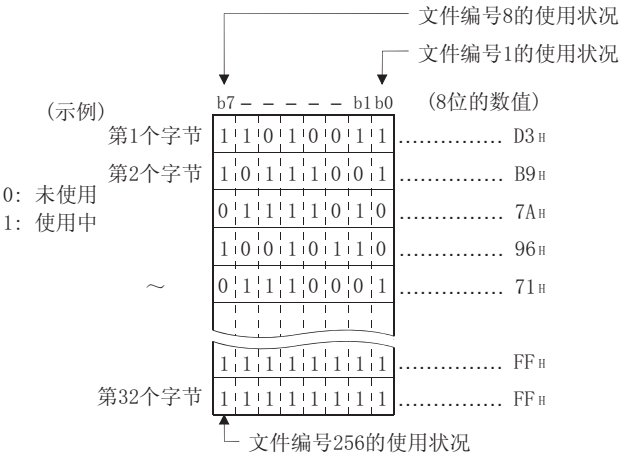
- 对文件名、文件大小进行更改的情况下
(指令：1204，子指令：0001)



- 对文件信息进行批量更改的情况下
(指令: 1204, 子指令: 0002)



- (q) 文件编号使用状况：用于 QnACPU 文件
- 是在文件编号使用状况的读取中，表示返回至外部设备侧的 256 个文件编号的使用状况的数据，从表示文件编号使用状况的下图第 1 字节开始进行排列。
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 64 位 (16 进制数) 后发送至外部设备侧。
(8 个文件编号：2 位)
 - 2) 通过二进制代码进行数据通信时
表示使用状况的下述 32 个字节的数值从 Low 字节 (L：位 0~7) 开始发送至外部设备侧。(8 个文件编号：1 字节)
 - 3) 文件编号使用状况的内容如下所示。
1 文件编号/1 位，各文件编号的使用状况如下所示。



上图所示的使用状况中，返回至外部设备侧的文件编号使用状况的内容如下所示。

- ASCII 代码数据的通信时、返回 “D3B97A…FFFF”、从 “D” 开始按顺序进行发送。
- 通过二进制代码进行数据通信时，返回 D3_H、B9_H、7A_H…FF_H、FF_H，从 D3_H 开始按顺序进行发送。

- (r) 文件锁定模式：用于 QnACPU 文件
是允许从其它设备对指定文件进行访问时，用于指定是否强制执行文件锁定解除的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L：位 0~7)开始进行发送。
- 3) 文件锁定模式的指定值及指定内容如下所示，不能进行除此以外的指定。

指定值	指定内容
0000 _H	正常执行文件锁定解除。
0002 _H	强制执行文件锁定解除。

- 4) 对指定文件进行文件锁定解除时，正常执行与强制执行的差别如下所示。
 - 正常执行
其它设备进行了文件锁定登录时，不进行文件锁定的解除。
发出解除请求时将变为出错状态，并返回异常响应。
 - 强制执行
即使其它设备进行了文件锁定登录，也强制执行文件锁定的解除。
该强制执行功能可用于以下情况：由于进行了文件锁定登录的设备发生了故障而无法进行文件锁定解除的情况下。

- (s) 复制模式：用于 QnACPU 文件
是在文件的复制中，用于指定是否在复制完毕时将复制源文件的最后编辑日期复制到复制目标文件中的数据。
不复制时，新建文件时的 QnACPU 的管理时间将原样不变地保留。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L：位 0~7)开始进行发送。
- 3) 复制模式的指定值及内容如下所示。

指定值	指定内容
0000 _H	复制完毕时，不对复制源文件的最后编辑日期时间进行复制。
0001 _H	复制完毕时，对复制源文件的最后编辑日期时间进行复制。

- (t) 目录字符数：用于 Q/LCPU 文件
是用于指定进行读取、写入、登录等的文件的存储位置的数据。
- 1) 通过 ASCII 代码进行数据通信时
使用“0000”进行发送。
 - 2) 通过二进制代码进行数据通信时
使用 2 字节的数值“0000h”进行发送。
- (u) 备用数据 n：用于 QnACPU 文件
- 1) 通过 ASCII 代码进行数据通信时
使用 4 字符的 NuLL 数据(代码：00h)进行发送。
 - 2) 通过二进制代码进行数据通信时
使用 2 字节的数值“0000h”进行发送。
- (v) 文件指针号：用于 Q/LCPU 文件
是可编程控制器 CPU 用于管理文件的编号。
将打开文件时返回的数据原样不变地使用。
- 1) 通过 ASCII 代码进行数据通信时
对 4 字符的数据进行发送。
 - 2) 通过二进制代码进行数据通信时
对 2 字节的数值进行发送。
- (w) 关闭类型：用于 Q/LCPU 文件
是允许从其它设备对指定文件进行访问时，用于指定是否强制执行文件关闭的数据。
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L：位 0~7)开始进行发送。
 - 3) 关闭类型的指定值及处理内容如下所示，不能进行除此以外的指定。

指定值	指定内容	处理内容
0000h	正常关闭	仅关闭对象文件。
0001h	强制关闭-1	对包括打开了对象文件的模块/设备所打开的其它文件在内，进行强制文件关闭。
0002h	强制关闭-2	对打开的所有文件进行强制关闭。

- 4) 对指定文件进行文件关闭时，正常关闭与强制关闭的区别如下所示。
- 正常关闭
不关闭模块/设备打开的其它文件。
发出关闭请求时，将异常结束。
 - 强制关闭-1
关闭同一模块/设备所打开的其它所有文件。(*1)
 - 强制关闭-2
关闭当前打开的所有文件。(*1)
- *1 在以下情况下进行此指定：由于故障等，打开了文件的模块/设备无法关闭文件时，通过其它模块・设备关闭打开中的文件时。
- (x) 打开模式：用于 Q/LCPU 文件
是用于对指定文件是以读取用方式打开还是以写入用方式打开进行指定的数据。
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L：位 0~7)开始进行发送。
 - 3) 打开模式的指定值及处理内容如下所示，不能进行除此以外的指定。

指定值	指定内容	处理内容
0000 _h	读取打开	作为数据的读取用打开对象文件。
0100 _h	写入打开	作为数据的写入用打开对象文件。

3.8.2 文件控制时的注意事项

- 进行 Q/LCPU 或 QnACPU 的文件控制时的注意事项如下所示。
- (1) 从 Q/L/QnACPU 中读取的文件是用于存放到外部设备中。
对于从 Q/L/QnACPU 中读取的文件内容，不能在外部设备侧进行编辑。
- (2) 进行数据的读取/写入的情况下，一次通信中无法对整个文件容量进行数据的读取/写入时，应分数次通信进行数据的读取/写入。
对于文件大小，可以通过以下功能进行确认。

功能	说明项	
	用于 Q/LCPU	用于 QnACPU
文件信息列表的读取功能	3.8.5 项	3.8.16 项
文件有无的读取功能	3.8.6 项	3.8.17 项

- (3) 使用以下功能时，Q/L/QnACPU 被实施了系统保护的情况下将变为出错状态，返回异常结束的报文。

功能	说明项	
	Q/LCPU 用	QnACPU 用
新建文件(文件名登录)	3. 8. 10 项	3. 8. 20 项
文件写入	3. 8. 11 项	3. 8. 21 项
文件删除	3. 8. 12 项	3. 8. 22 项
文件复制	3. 8. 13 项	3. 8. 23 项
文件信息(创建日期、属性)更改	3. 8. 14 项 3. 8. 15 项	3. 8. 24 项

- (4) 文件中登录了关键字时，应对登录的关键字进行备忘。
对以下文件进行访问的情况下，进行相应文件的打开时或文件的读取/写入时，需要指定登录的关键字。
- 参数文件
 - 程序文件

- (5) 对于文件的属性，仅在使用以下功能时有效。

功能	说明项	
	用于 Q/LCPU	用于 QnACPU*1
目录・文件信息的读取	3. 8. 5 项	3. 8. 16 项
新登录文件	—	3. 8. 20 项
文件信息的更改(创建日期、属性)	3. 8. 15 项	3. 8. 24 项

*1 在其它指令中将被作为虚拟键值处理。

- (6) 新建用于 Q/LCPU 的下述文件时，应按以下处理步骤创建最终目标文件。

- 1) 指定除下述扩展名以外的任意扩展名，新建临时文件。
- 2) 打开新建文件进行数据写入后，关闭文件。
- 3) 通过复制功能创建最终目标扩展名的文件。
- 4) 复制处理完毕后，根据需要删除复制源文件。

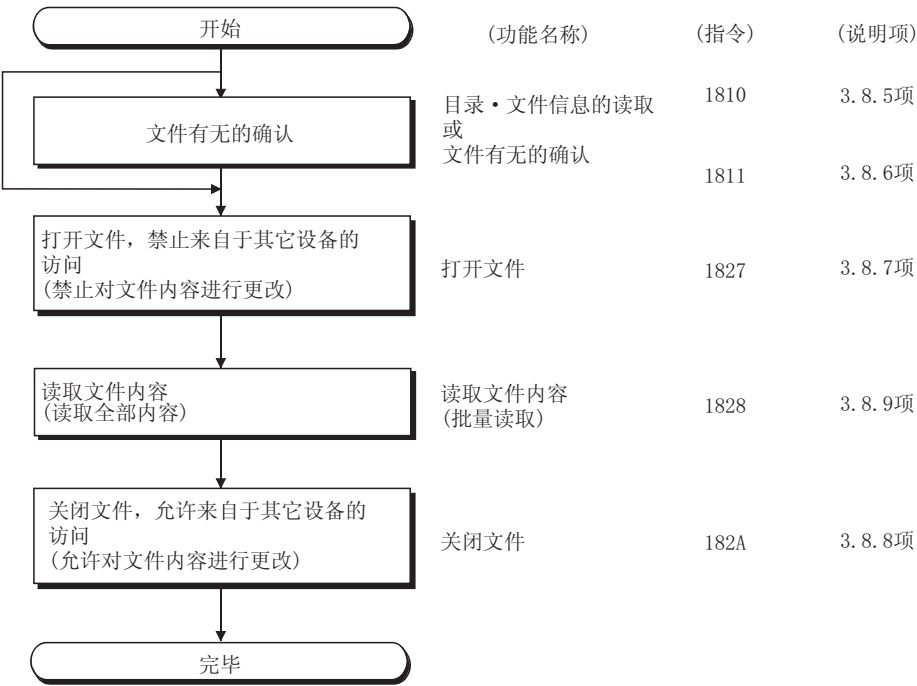
文件的类型	文件的扩展名
索引文件	DAT
顺控程序文件	QPG
软元件注释文件	QCD
软元件初始值文件	QDI

- (7) 关于除上述以外的注意事项，请参阅各功能说明项。

3.8.3 用于 Q/LCPU 文件控制的执行步骤

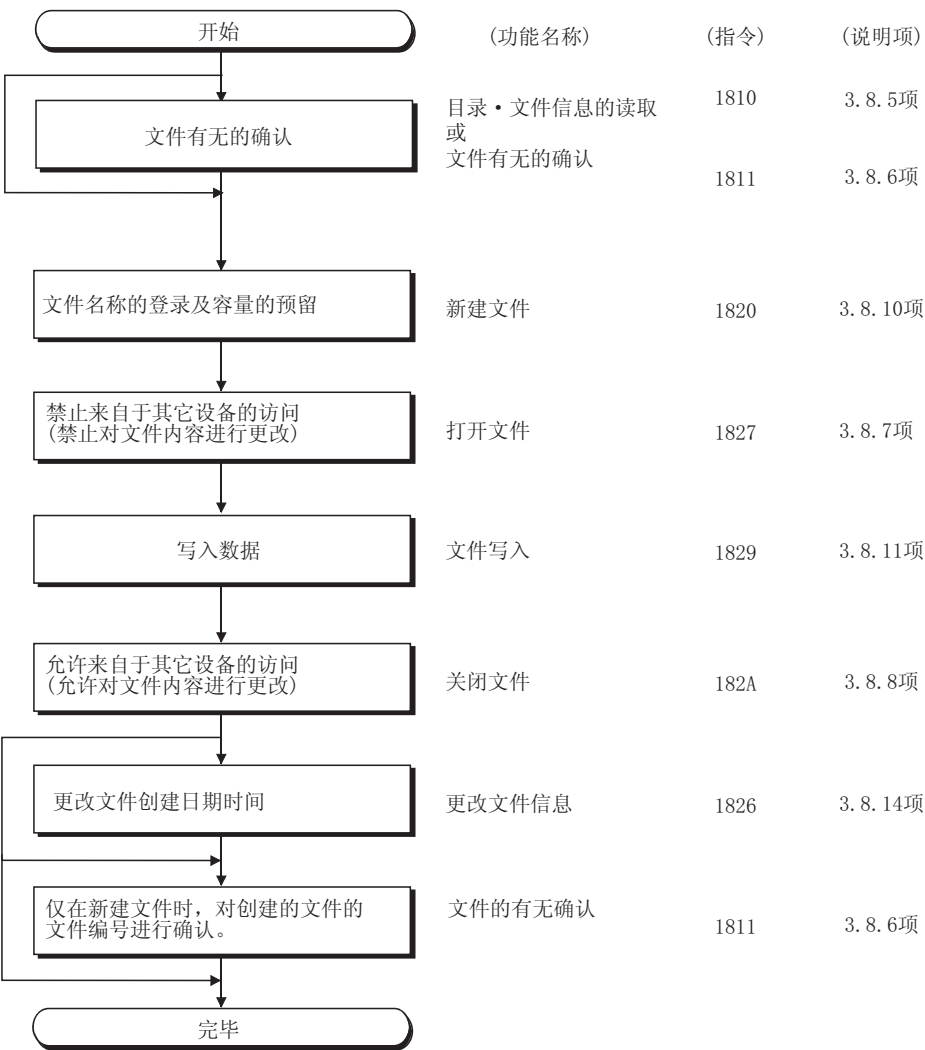
对 Q/LCPU 进行文件控制时的步骤如以下流程所示。

(1) 读取文件内容时的步骤



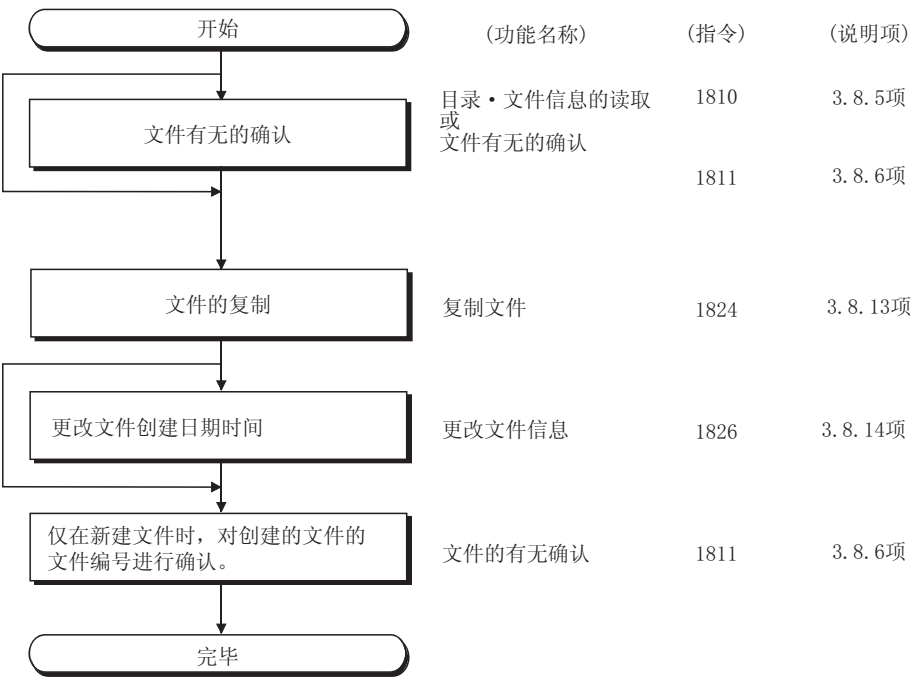
要点	
应对外部设备侧读取的文件(存放用)的以下文件信息进行预先存储。	
• 文件编号	
• 文件的名称及属性	
• 文件大小	

(2) 新建文件，进行数据写入时的步骤



要点
(1) Q/LCPU 中存储的各文件被存储在程序区域(存储器)的连续区域中。 为了防止从外部设备新建文件时发生空余区域不足出错，在运行 Q/LCPU 前应在 GX Works2 或 GX Developer 中通过下述某个操作预留出必要容量以上的连续区域。 1) 删除不需要的文件。 2) 进行可编程控制器存储器整理操作。 (2) 新建 Q/LCPU 的文件时，请参阅 3.8.2 项。

(3) 文件的复制步骤



要点	
Q/LCPU 中存储的各文件被存储在程序区域(存储器)的连续区域中。 为了防止从外部设备复制文件时发生空余区域不足出错，在运行 Q/LCPU 前应在 GX Works2 或 GX Developer 中通过下述某个操作预留出必要容量以上的连续区域。 1) 删除不需要的文件。 2) 进行可编程控制器存储器整理操作。	

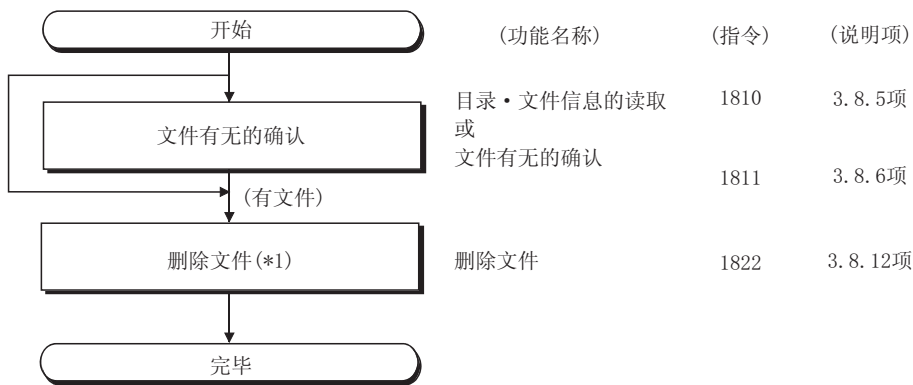
(4) 将数据覆盖到现有文件中时的步骤

对文件信息进行更改时的步骤



- *1 Q/LCPU 的现有文件的大小不能更改。
需要进行文件大小的更改的情况下，应通过以下步骤重新创建文件。
- 1) 应通过本项(1)中所示步骤，对对象文件的全部数据进行读取。
 - 2) 应通过本项(5)中所示步骤，删除对象文件。
 - 3) 应通过本项(2)中所示步骤，新建文件并写入全部数据。

(5) 删除文件时的步骤

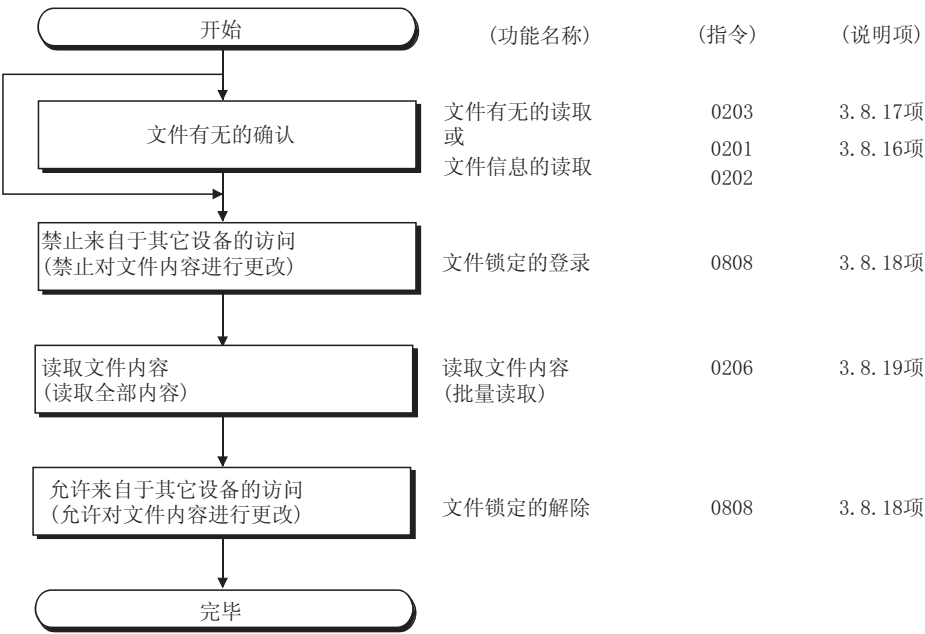


*1 对于文件删除时机，应由包括 Q/LCPU 及关联设备在内的整个系统确定。

3.8.4 用于 QnACPU 文件控制的执行步骤

对 QnACPU 进行文件控制时的步骤如以下流程所示。

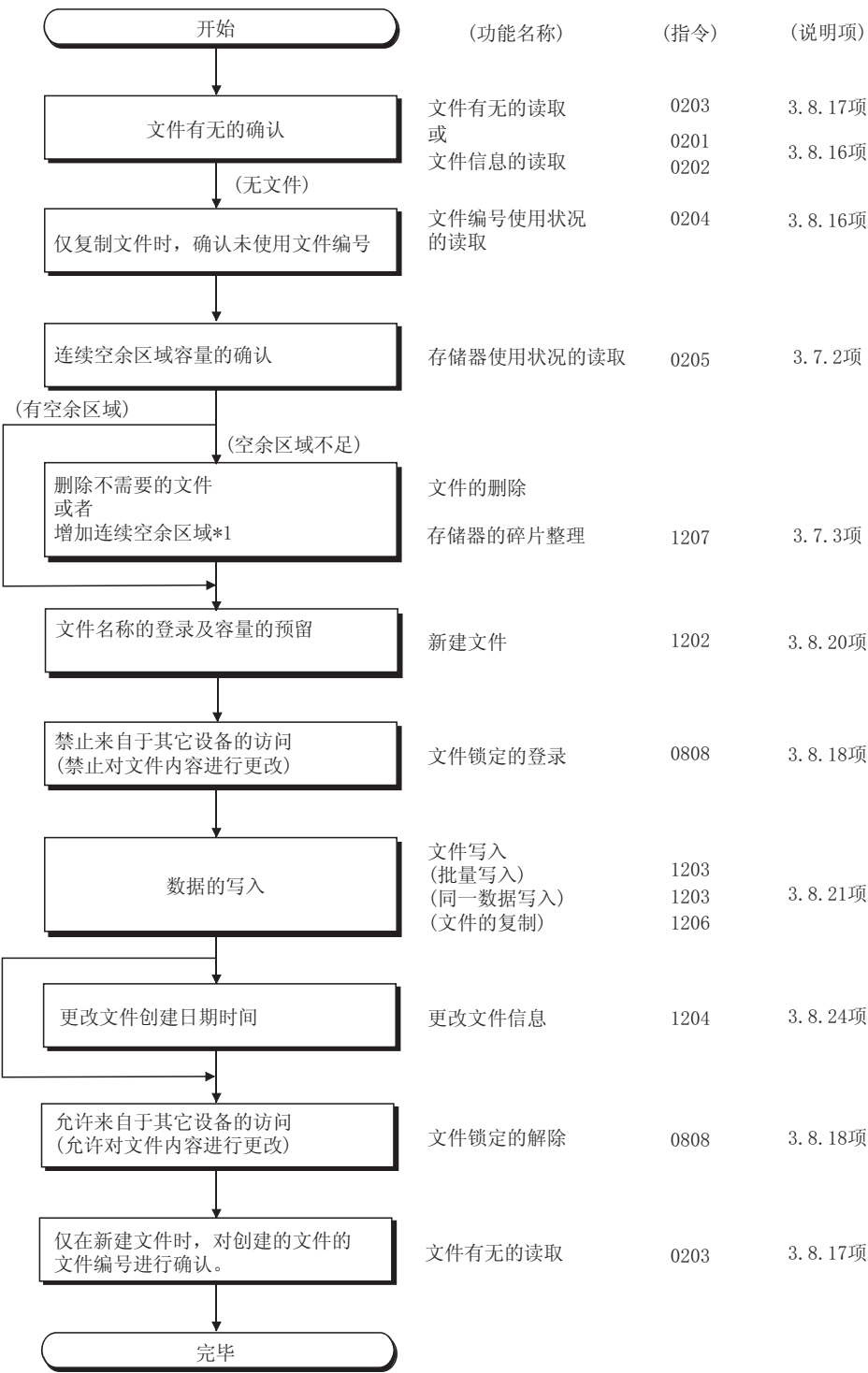
(1) 读取文件内容时的步骤



要点
应对外部设备侧读取的文件(存放用)的以下文件信息进行预先存储。 • 文件编号 • 文件的名称及属性 • 文件大小

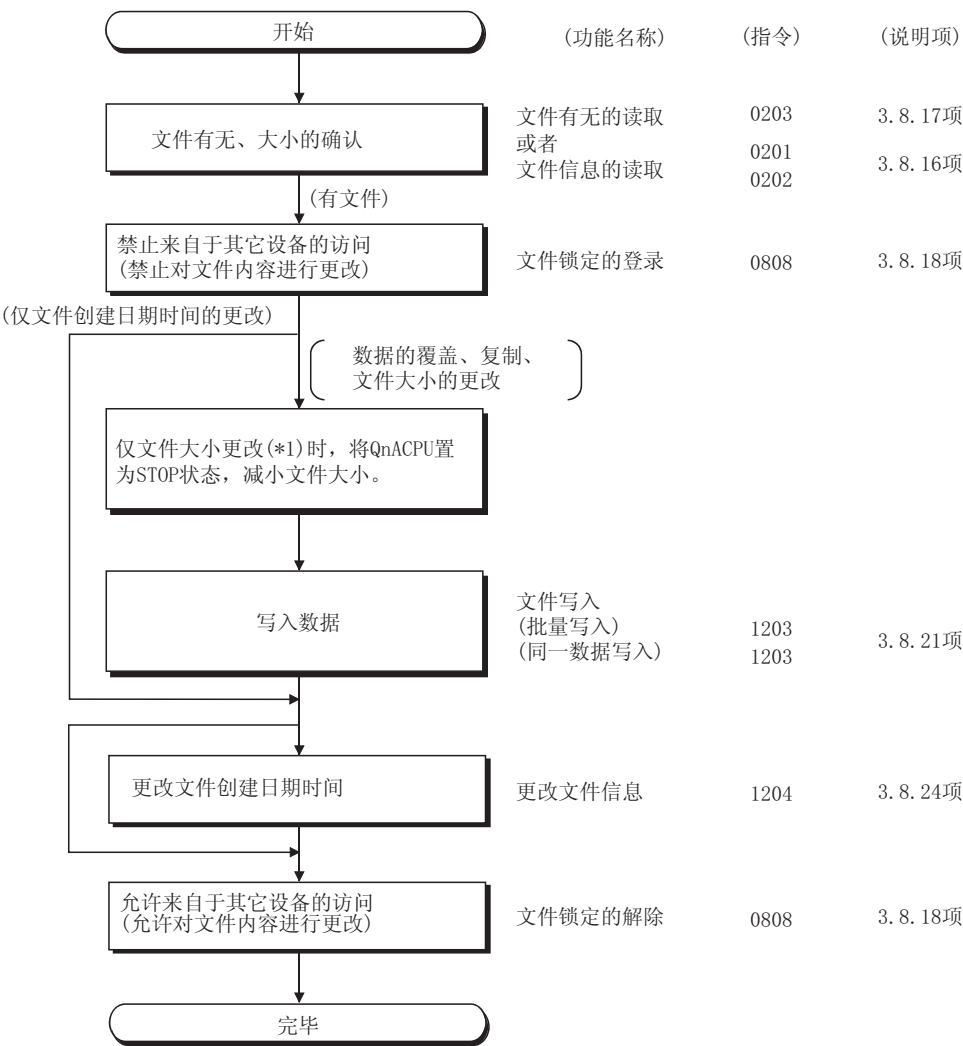
(2) 新建文件，进行数据写入时的步骤

将现有文件的数据复制到新建文件中时的步骤



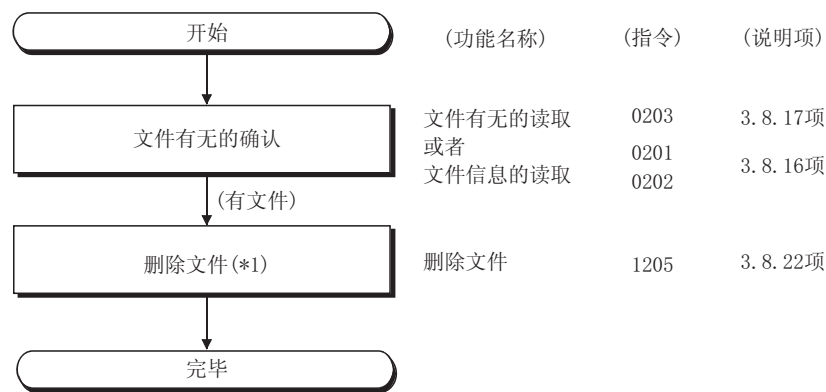
*1 应通过 3. 6. 3 项中所示的远程 STOP(指令：1002) 功能等将 QnACPU 置为 STOP 状态之后再行存储器的碎片整理。
本项的处理完成后应通过 3. 6. 2 项中所示的远程 RUN(指令：1001) 功能等将 QnACPU 置为 RUN 状态。

(3) 将数据覆盖到现有文件中时的步骤
对文件信息进行更改时的步骤



*1 仅在减小文件的大小时，可通过 3.8.24 项中所示的文件信息的更改(指令：1204) 功能对文件大小进行更改。
需要增大文件大小时，应通过本项(2)中所示的步骤新建文件，写入数据。

(4) 删除文件时的步骤



*1 对于文件删除时机，应由包括 QnACPU 及关联设备在内的整个系统确定。

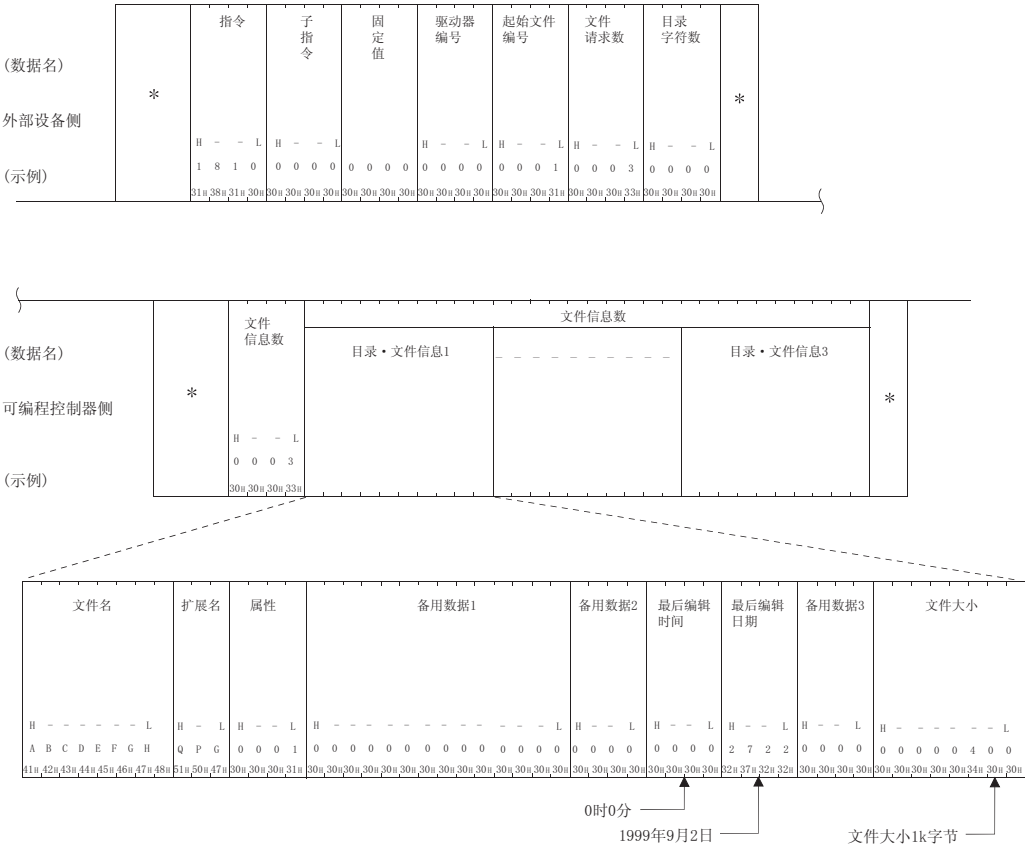
3.8.5 目录・文件信息的读取(指令：1810)：用于 Q/LCPU

以下对目录・文件信息读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

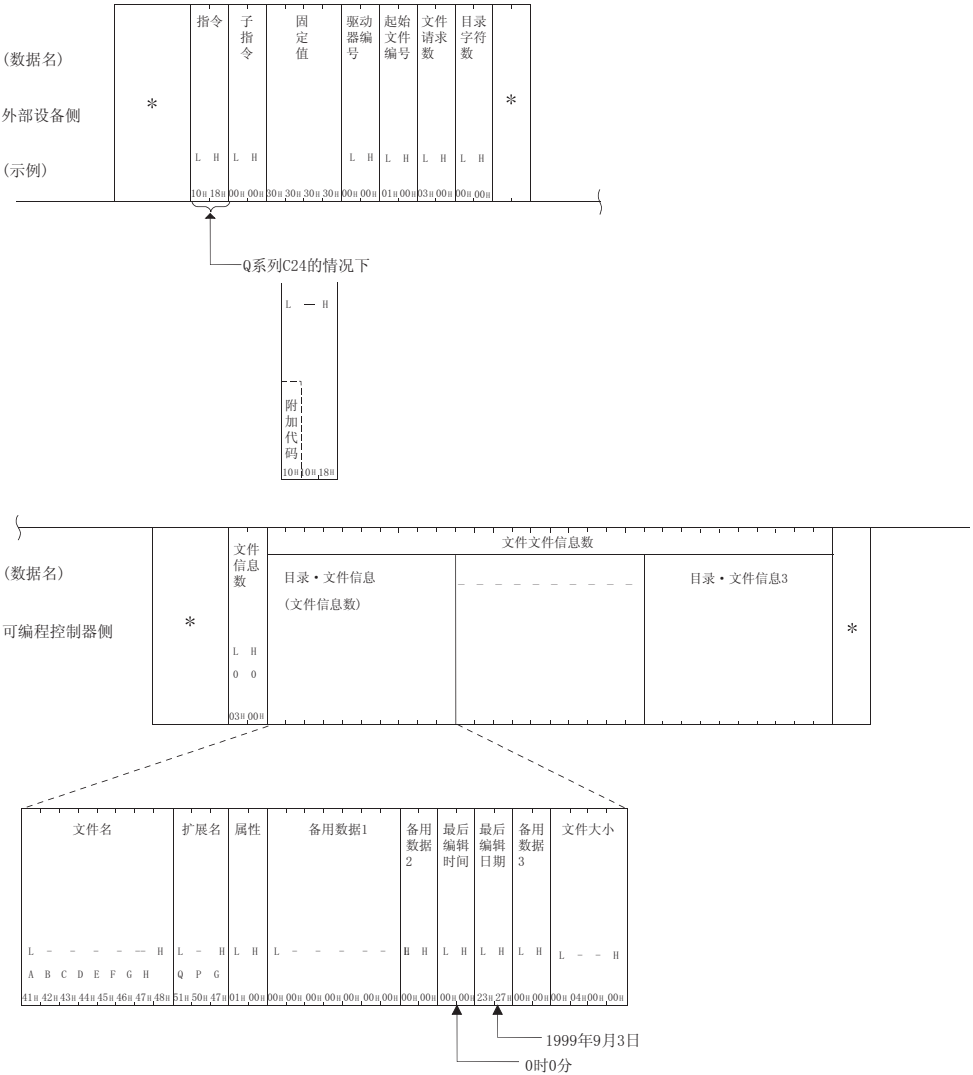
[控制步骤]

以下条件下的目录・文件读取示例如下所示。
驱动器编号 0
起始文件编号 1
文件请求数 3

(1) 通过 ASCII 代码进行通信时



(2) 通过二进制代码进行通信时



要点

(1) 各指定位置应在以下范围内进行指定或返回。

- 起始文件编号 : 1≦文件编号≦256
- 文件请求数 : 1≦文件编号≦36
- 文件信息数 : 0≦文件数≦文件请求数 (0: 指定的起始文件编号以后无登录文件)

(2) 指定的文件编号范围内未能登录所有文件时, 文件信息数将变为指定范围内登录的文件数(返回的文件信息数)。

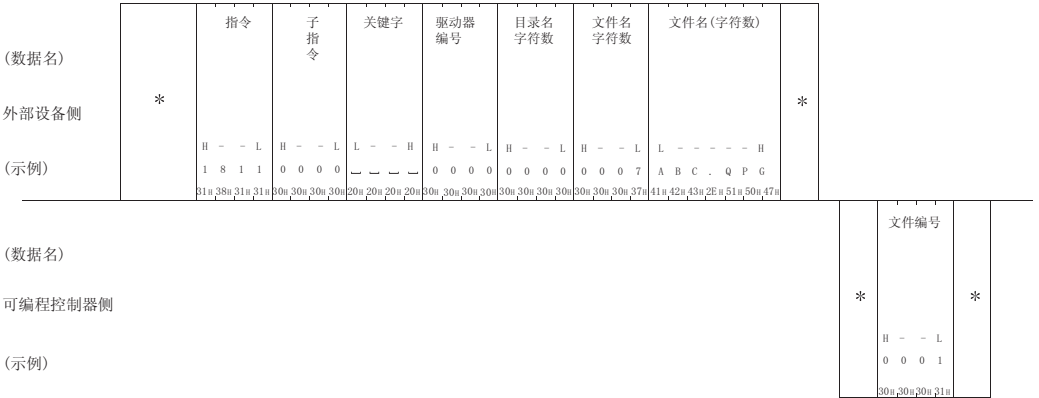
3.8.6 目录・文件信息的搜索(指令：1811)：用于 Q/LCPU

以下对目录・文件信息搜索的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

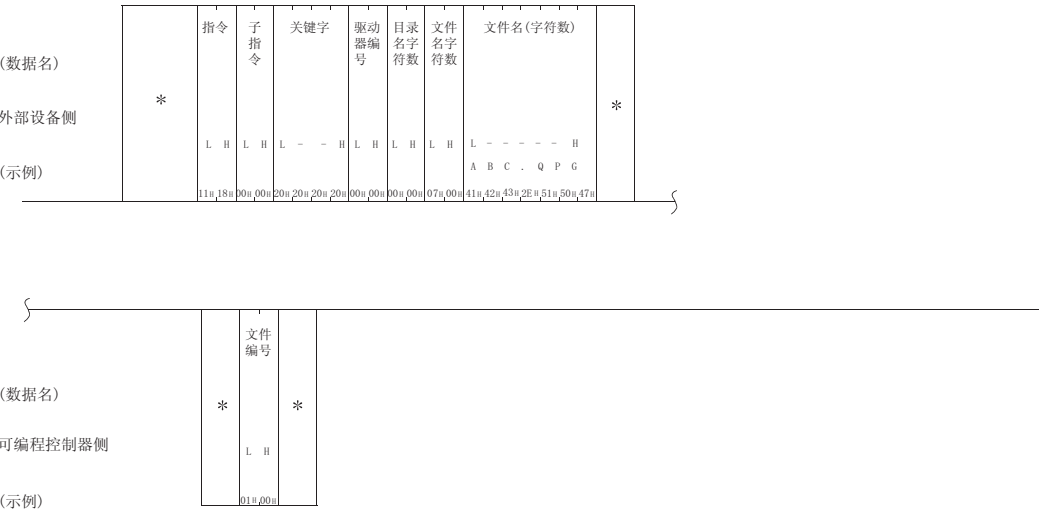
[控制步骤]

以下条件下的目录・文件信息搜索示例如下所示。
关键字 _ _ _ _ (空格 4 字符，代码：20H)
驱动器编号 0
文件名 ABC.QPG

(1) 通过 ASCII 代码进行通信时



(2) 通过二进制代码进行通信时



要点	指定文件不存在时将变为出错状态，将返回出错时的结束代码。
----	------------------------------

3.8.7 文件的打开(指令：1827)：用于 Q/LCPU

以下对打开文件的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 打开 QCPU 的文件的情况下(子指令“0000”)

以下条件下的文件的打开示例如下所示。
关键字 1234
驱动器编号 0
文件名 ABCMAIN.QPG
打开模式 写入打开

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字	打开模式	驱动器编号	文件名 字符数	文件名(字符数)	
外部设备侧	*								*
(示例)		H - - L 1 8 2 7	H - - L 0 0 0 0	H - - L 1 2 3 4	H - - L 0 1 0 0	H - - L 0 0 0 0	H - - L 0 0 0 7	L - - - - H A B C . Q P G	
		31h 38h 32h 37h	30h 30h 30h 30h	31h 32h 33h 34h	30h 31h 30h 30h	30h 30h 30h 30h	30h 30h 30h 37h	41h 42h 43h 2Eh 51h 50h 47h	
(数据名)								文件指针号	
可编程控制器侧								*	*
(示例)								H - - L 0 0 0 0	
								30h 30h 30h 30h	

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	关键字	打开模式	驱动器编号	文件名 字符数	文件名(字符数)	
外部设备侧	*								*
(示例)		L H 27h 18h	L H 00h 00h	L - - H 31h 32h 33h 34h	L H 00h 01h	L H 00h 00h	L H 07h 00h	L - - - - H A B C . Q P G	
								41h 42h 43h 2Eh 51h 50h 47h	
(数据名)								文件指针号	
可编程控制器侧								*	*
(示例)								L H 00h 00h	

(2) 打开 LCPU 的文件的情况下(子指令“0004”)

以下条件下的文件的打开示例如下所示。
 关键字 AbCd1234L• • • L(空格 24 字符，代码：20H)
 驱动器编号 0
 文件名 MAIN.QPG
 打开模式 写入打开

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字(固定为32字符)	打开模式	驱动器编号	文件名字符数	文件名(字符数)	
外部设备侧	*								*
(示例)		H - - L 1 8 2 7	H - - L 0 0 0 4	L - - - - - A b C d 1 2 3 4 L	H - - L 0 1 0 0	H - - L 0 0 0 0	H - - L 0 0 0 8	L - - - - - M A I N . Q P G	
		31H 38H 32H 37H	30H 30H 30H 34H	41H 62H 43H 64H 31H 32H 33H 34H 20H - - - 20H	30H 31H 30H 30H	30H 30H 30H 30H	30H 30H 30H 38H	4DH 41H 49H 4EH 2EH 51H 50H 47H	

(数据名)		文件指针号	
可编程控制器侧	*		*
(示例)		H - - L 0 0 0 0 30H 30H 30H 30H	

(b) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字(固定为32字符)	打开模式	驱动器编号	文件名字符数	文件名(字符数)	
外部设备侧	*								*
(示例)		L H L H L	L - - - - -	L - - - - -	L H L H L	L H L H L	L - - - - -	L - - - - -	
		27H 18H 04H 00H	41H 62H 43H 64H 31H 32H 33H 34H 20H - - - 20H	00H 01H 00H 00H 08H 00H 4DH 41H 49H 4EH 2EH 51H 50H 47H					

(数据名)		文件指针号	
可编程控制器侧	*		*
(示例)		L H 00H 00H	

要点
对于处于打开状态的文件，Q/LCPU 的再启动(CPU 复位等)将使文件变为关闭状态。

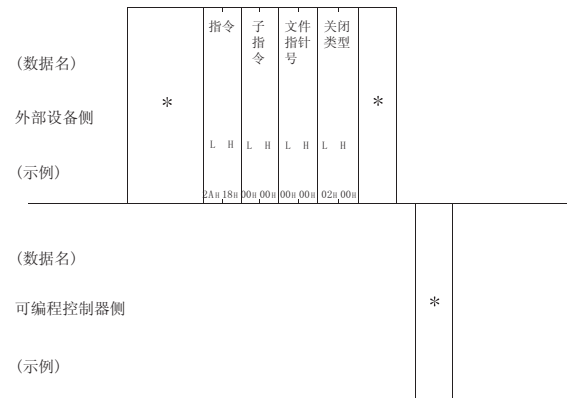
3.8.8 文件的关闭(指令：182A)：用于 Q/LCPU

以下对文件关闭的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

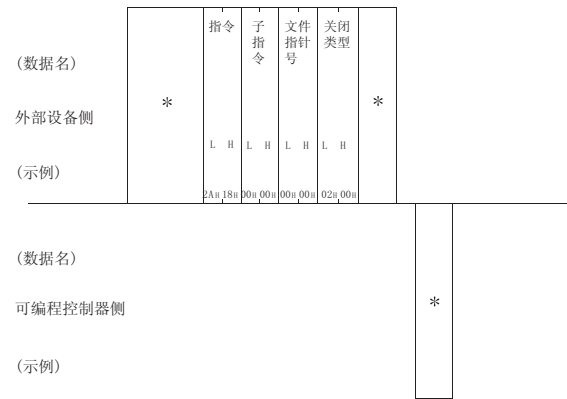
[控制步骤]

以下条件下的文件的关闭示例如下所示。
文件指针号 0
关闭类型 2

(1) 通过 ASCII 代码进行通信时



(2) 通过二进制代码进行通信时



要点
(1) 正在进行文件的打开时，Q/LCPU 的再启动(CPU 复位等)将使文件变为关闭状态。

3.8.9 文件的读取(指令：1828)：用于 Q/LCPU

以下对文件读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

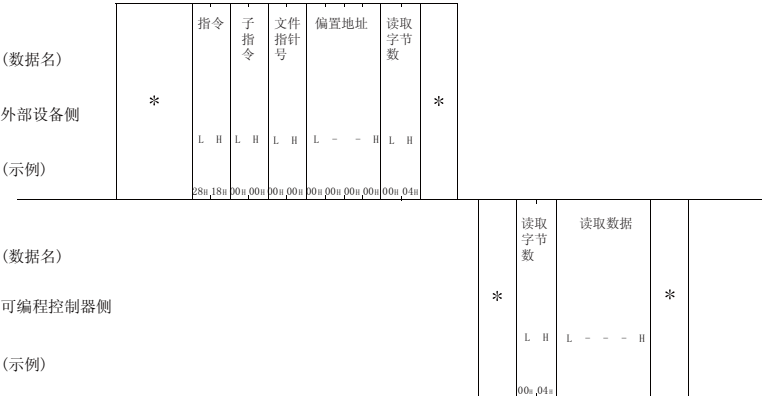
[控制步骤]

以下条件下的文件读取示例如下所示。
文件指针号 0
读取字节数 1K 字节

(1) 通过二进制代码进行通信时

(数据名)		指令	子指令	文件指针号	偏置地址	读取字节数	
外部设备侧	*						
(示例)		H - - L 1 8 2 8	H - - L 0 0 0 0	H - - L 0 0 0 0	H - - - - - L 0 0 0 0 0 0 0 0 0 0	H - - L 0 4 0 0	
		31n 38n 32n 38n	80n 30n 30n 30n	30n 30n 30n 30n	30n 30n 30n 30n 30n 30n 30n 30n	30n 34n 30n 30n	
(数据名)		读取字节数	读取数据				
可编程控制器侧							
(示例)		H - - L 0 4 0 0	L - - - - - H				
		30n 34n 30n 30n					

(2) 通过二进制代码进行通信时



要点

(1) 读取数据时每次可读取的最多字节数是确定的。
对于写入到指定文件中的数据，应对偏置地址及读取字节数进行调整，分为若干次进行全部读取。
此外，对于读取到外部设备中的数据，应原样不变地进行存放保管。
对于文件的大小，可通过以下功能进行确认。

- 目录・文件信息列表的读取功能 : 参阅 3.8.5 项
- 文件有无的读取功能 : 参阅 3.8.6 项

(2) 各指定值应在以下范围内进行指定。

- 偏置地址 : 在以下范围内以偶数地址进行指定
 0≤地址≤(文件大小-1)
- 读取字节数 : 0≤字节数≤1920

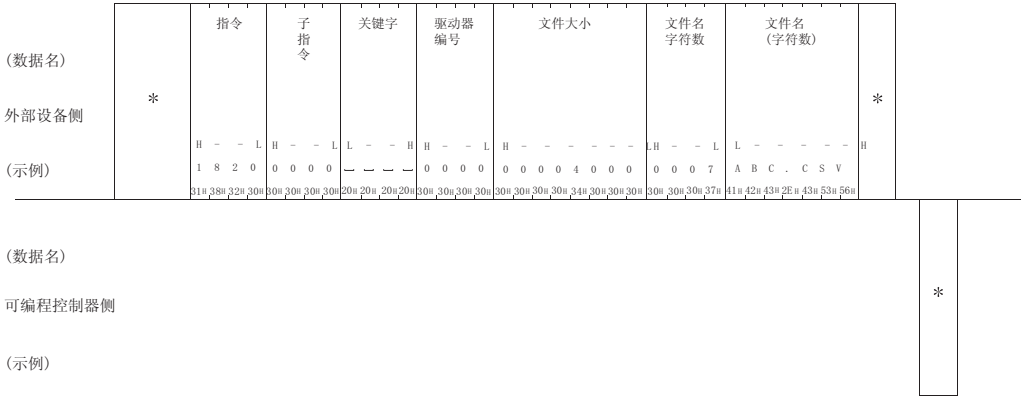
3.8.10 文件的新建(指令：1820)：用于 Q/LCPU

以下对新建文件的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

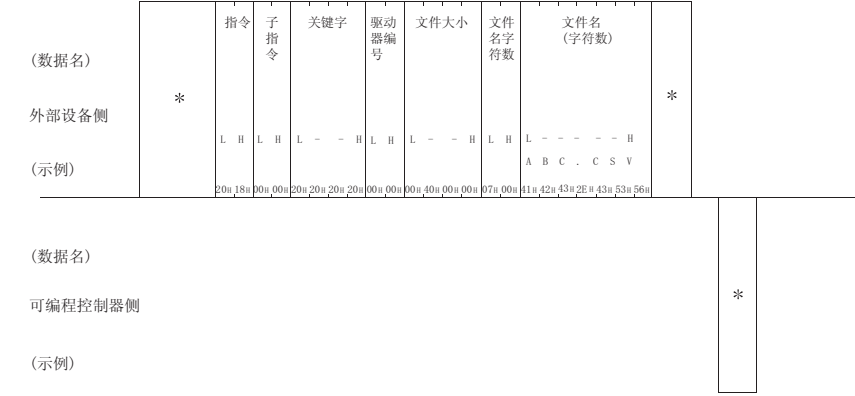
[控制步骤]

以下条件下的新建文件示例如下所示。
关键字 _ _ _ _ (空格 4 字符，代码：20H)
驱动器编号 0
文件名 ABC.CSV
文件大小 1k 字节

(1) 通过 ASCII 代码进行通信时



(2) 通过二进制代码进行通信时



要点
(1) 可以在指定驱动器存储器中连续的未使用簇的容量以内新建文件。 新建文件时，建议设置考虑了将来数据添加的容量。
(2) 新建文件的属性应指定为“20H”（可读取、写入的磁盘文件）。
(3) 在通过本功能新建的文件中，作为最后编辑日期时间将登录 QCPU 的管理时间。

3.8.11 至文件的写入(指令：1829)：用于 Q/LCPU

以下对文件写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

以下条件下的文件写入示例如下所示。
文件指针号 0
偏置地址 0
写入字节数 1K 字节

(1) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	文件指针号	偏置地址	写入字节数	写入数据	
外部设备侧	*							*
(示例)		H - - L	H - - L	H - - L	H - - - - - L	H - - L	H - - - - - L	L
		1 8 2 9	0 0 0 0	0 0 0 0	0 0 0 0 0 0 0 0 0 0	0 4 0 0		
		31h 38h 32h 39h	30h 30h 30h 30h	30h 30h 30h 30h	30h 30h 30h 30h 30h 30h 30h 30h 30h 30h	30h 34h 30h 30h		

(数据名)		写入字节数	
可编程控制器侧	*		*
(示例)		H - - L	
		0 4 0 0	
		30h 34h 30h 20h	

(2) 通过二进制代码进行通信时

(数据名)		指令	子指令	文件指针号	偏置地址	写入字节数	写入数据	
外部设备侧	*							*
(示例)		L H	L H	L H	L - - H	L H	L - - - H	
		29h 18h	00h 00h	00h 00h	00h 00h 00h 00h	00h 04h		

(数据名)		写入字节数	
可编程控制器侧	*		*
(示例)		L H	
		00h 04h	

要点	
(1)	写入数据时每次的最多字节数是确定的。 对于从 Q/LCPU 中读取后存放到外部设备中的数据，应对偏置地址及写入字节数进行调整，分为若干次全部写入到指定文件中。
(2)	Q/LCPU 处于运行中的情况下，如果指定以下文件将变为出错状态，将返回出错时的结束代码。 • 参数文件 • 程序存储器 (驱动器名: 00H) 中当前正在执行的文件
(3)	各指定值应在以下范围内进行指定。 • 偏置地址 : $0 \leq \text{地址} \leq (\text{文件大小}-1)$ 对驱动器名为“00H”(程序存储器)的文件进行写入时，在上述范围内以 4 的倍数(10 进制数的情况下，0、4、8、...)的地址进行指定。 对驱动器名为“00H”以外的文件进行写入时，在上述范围内以偶数地址(10 进制数的情况下，0、2、4、6、8、...)进行指定。 • 写入字节数 : $0 \leq \text{字节数} \leq 1920$
(4)	应在新建时预留的文件大小以内进行文件写入。 超过文件大小进行写入的情况下，有可能变为写入出错(出错代码: 402BH)。

3.8.12 文件的删除(指令：1822)：用于 Q/LCPU

以下对文件删除的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 对 QCPU 的文件进行删除时(子指令“0000”)

以下条件下的文件删除示例如下所示。
关键字 1234
驱动器编号 0
删除文件名 ABC.QPG

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字	驱动器编号	文件名 字符数	文件名 (字符数)	
外部设备侧	*							*
(示例)		H - - L	H - - L	L - - H	H - - L	H - - L	L - - - - H	
		1 8 2 2	0 0 0 0	1 2 3 4	0 0 0 0	0 0 0 7	A B C . Q P G	
		31# 38# 32# 32#	30# 30# 30# 30#	31# 32# 33# 34#	30# 30# 30# 30#	30# 30# 30# 37#	41# 42# 43# 2E# 51# 50# 47#	
(数据名)								
可编程控制器侧								*
(示例)								

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	关键字	驱动器编号	文件名 字符数	文件名 (字符数)	
外部设备侧	*							*
(示例)		L H	L H	L - - H	L H	L H	L - - - - H	
		1 2 3 4					A B C . Q P G	
		22# 18# 00# 00#	31# 32# 33# 34#	00# 00#	07# 00#	41# 42# 43# 2E# 51# 50# 47#		
(数据名)								
可编程控制器侧								*
(示例)								

(2) 对 LCPU 的文件进行删除时(子指令“0004”)

以下条件下的文件删除示例如下所示。
 关键字 AbCd1234 (空格 24 字符，代码：20H)
 驱动器编号 0
 删除文件名 MAIN.QPG

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字(固定为32字符)	驱动器编号	文件名 字符数	文件名 (字符数)	
外部设备侧	*							*
(示例)		H 1 8 2 2	L 0 0 0 4	A b C d 1 2 3 4	H 0 0 0 0	L 0 0 0 8	M A I N . Q P G	
		31h:38h:32h:32h:30h:30h:30h:34h:41h:62h:43h:64h:31h:32h:33h:34h:20h:-- -- -- 20h:30h:30h:30h:30h:30h:30h:30h:38h:4Dh:41h:49h:4Eh:2Eh:51h:50h:47h:-- -- --						
(数据名)								
可编程控制器侧								*
(示例)								

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	关键字(固定为32字符)	驱动器编号	文件名 字符数	文件名 (字符数)	
外部设备侧	*							*
(示例)		L 22h:18h:04h:00h:41h:62h:43h:64h:31h:32h:33h:34h:20h:-- -- -- 20h:00h:00h:08h:00h:4Dh:41h:49h:4Eh:2Eh:51h:50h:47h:-- -- --	H 04h:00h:41h:62h:43h:64h:31h:32h:33h:34h:20h:-- -- -- 20h:00h:00h:08h:00h:4Dh:41h:49h:4Eh:2Eh:51h:50h:47h:-- -- --					
(数据名)								
可编程控制器侧								*
(示例)								

要点
(1) 对于文件删除时机，应由包括 Q/LCPU 及关联设备在内的整个系统确定。 (2) 对于处于打开状态的文件，不能进行删除。 (3) Q/LCPU 处于“运行 N 中”时，不能删除以下文件。 • 程序文件(□.QPG) • 参数文件(□.QPA) • 引导设置文件(□.QBT)

3.8.13 文件的复制(指令：1824)：用于 Q/LCPU

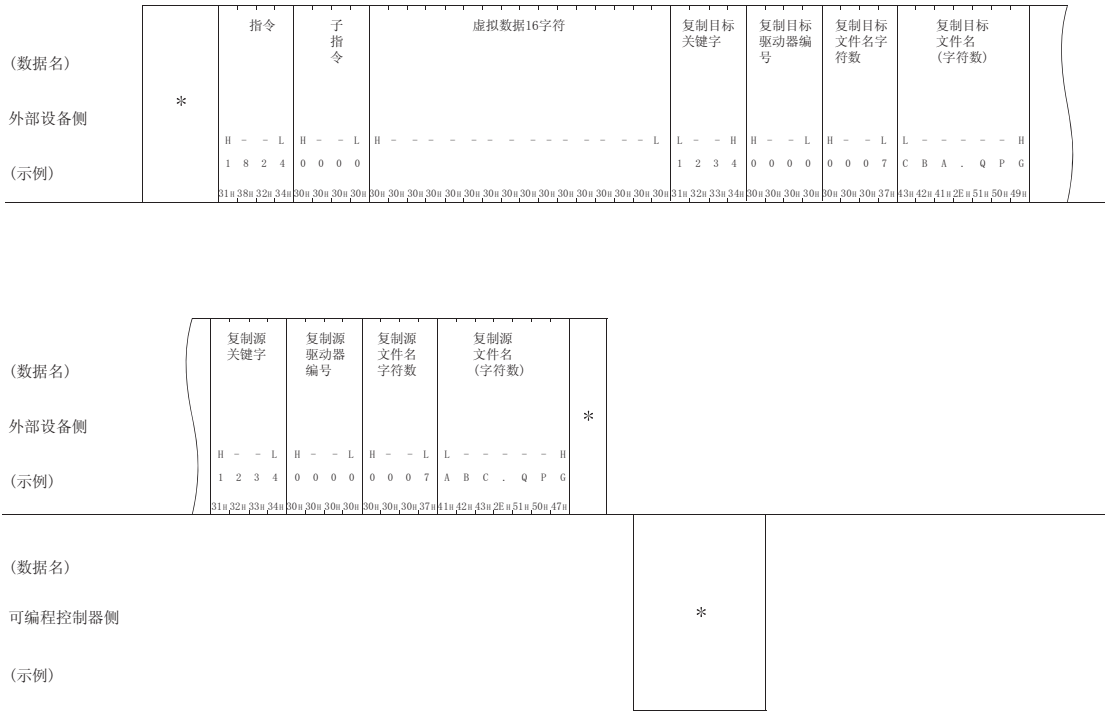
以下对文件复制的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 对 QCPU 的文件进行复制时(子指令“0000”)

以下条件下的文件复制示例如下所示。
关键字复制目标/源 1234
复制源驱动器编号 0
复制目标驱动器编号 1
复制源文件名 ABC.QPG
复制目标文件名 CBA.QPG

(a) 通过 ASCII 代码进行通信时



(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	虚拟数据(8字符)	复制目标关键字	复制目标驱动器号	复制目标文件名字符数	复制目标文件名(字符数)	复制源关键字	复制源驱动器编号	复制源文件名字符数	复制源文件名(字符数)	
外部设备侧	*	L H	L H	L - - - - - H	L - - - H	L H	L H	L - - - - - H C B A , Q P G	H - - - L	L H	L H	L - - - - - H A B C , Q P G	*
(示例)		24H 18H 00H 00H	00H 00H 00H 00H 00H 00H 00H	31H 32H 33H 34H 00H 00H 07H 00H	43H 42H 41H 2EH 51H 50H 47H	31H 32H 33H 34H 00H 00H 07H 00H	41H 42H 43H 2EH 51H 50H 47H						

*

(数据名)													
可编程控制器侧													*
(示例)													

(2) 对 LCPU 的文件进行复制时(子指令“0004”)

以下条件下的文件复制示例如下所示。

关键字复制目标/源 AbCd1234• • • (空格 24 字符, 代码: 20H)

复制源驱动器编号 0

复制目标驱动器编号 0

复制源文件名 MAIN.QPG

复制目标文件名 DEST. QPG

(a) 通过 ASCII 代码进行通信时

(数据名)	指令	子指令	虚拟数据16字符	关键字 (固定为32字符)	复制目标驱动器编号	复制目标文件名字符数
外部设备侧	*					
(示例)	H - - L 1 8 2 4	H - - L 0 0 0 4	L - - - - - H 0 0 0 0 - - -	L - - - - - - - - - H A b C d 1 2 3 4 _ _ _	H - - - L 0 0 0 0	H - - - L 0 0 0 8
	31:28:27:26:25:24:23:22:21:20:19:18:17:16:15:14:13:12:11:10:9:8:7:6:5:4:3:2:1:0	31:30:29:28:27:26:25:24:23:22:21:20:19:18:17:16:15:14:13:12:11:10:9:8:7:6:5:4:3:2:1:0	31:30:29:28:27:26:25:24:23:22:21:20:19:18:17:16:15:14:13:12:11:10:9:8:7:6:5:4:3:2:1:0	44:63:43:64:31:32:33:34:35:36:37:38:39:40:41:42:43:44:45:46:47:48:49:50:51:52:53:54:55:56:57:58:59	30:29:28:27:26:25:24:23:22:21:20:19:18:17:16:15:14:13:12:11:10:9:8:7:6:5:4:3:2:1:0	30:29:28:27:26:25:24:23:22:21:20:19:18:17:16:15:14:13:12:11:10:9:8:7:6:5:4:3:2:1:0

外部设备側	(数据名) 复制目标 文件名 (字符数) 关键字 (固定为32字符) 复制源 驱动器 编号 复制源 文件名 字符数 复制源 文件名 (字符数) *
(示例)	L - - - - - H L - - - - - H H - - L H - - L L - - - - - H D E S T . Q P G A b c d 1 2 3 4 _ _ 0 0 0 0 0 0 0 8 M A I N . Q P G 44_h45_h53_h54_h2E_h51_h50_h47_h11_h62_h43_h64_h31_h32_h33_h34_h20_h- - - -20_h30_h30_h30_h30_h30_h30_h38_h4D_h41_h49_h4E_h2E_h51_h50_h47_h
可编程控制器側	(数据名) *
(示例)	

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	虚拟数据8字符	关键字 (固定为32字符)	复制 目标 驱动器 编号	复制 目标 文件名 字符数	
外部设备侧	*							
(示例)		L	H	L	H	L	H	L
				0 0 0 0				
		24h	18h	04h	00h	00h	00h	00h
					41h	62h	43h	64h
					31h	32h	33h	34h
					20h			
						20h	00h	00h
							08h	00h

(数据名)		复制目标 文件名 (字符数)	复制源关键字 (固定为32字符)	复制 源 驱动器 编号	复制 源 文件 名 字符 数	复制源 文件名 (字符数)	*
外部设备侧							
(示例)		L	H	L	H	L	H
		D	E	S	T	.	Q
		P				G	
		44h	45h	53h	54h	2Eh	51h
				50h	47h	41h	62h
				43h	64h	31h	32h
				33h	34h	20h	
					20h	00h	00h
					08h	00h	
					4Dh	41h	49h
					4Eh	2Eh	51h
					50h	47h	

(数据名)		
可编程控制器侧		*
(示例)		

要点

- (1) 进行复制时每次的最多字节数是确定的。
对于写入到现有文件中的数据，应对偏置地址及复制字节数进行调整，分为若干次全部写入到新登录文件中。
- (2) Q/LCPU 处于运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。
 - 参数文件
 - 程序存储器 (驱动器名: 00H) 中当前正在执行的文件

3.8.14 文件创建日期的更改(指令：1826)：用于 Q/LCPU

以下对更改文件创建日期的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

以下条件下的文件创建日期的更改示例如下所示。
驱动器编号 0
更改日期 1999/09/02
更改时间 上午 0 时
文件名 ABC.QPG

(1) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	固定值	驱动器编号	更改日期	更改时间	文件名 字符数	文件名 (字符数)	
外部设备侧	*									*
(示例)		H - - L	H - - L		H - - L	H - - L	H - - L	H - - L	L - - - - H	
		1 8 2 6	0 0 0 0	0 0 0 0	0 0 0 0	2 7 2 2	0 0 0 0	0 0 0 7	A B C . Q P G	
		31h 38h 32h 36h	30h 30h 30h 30h	30h 30h 30h 30h	30h 30h 30h 30h	32h 37h 32h 32h	30h 30h 30h 30h	30h 30h 30h 37h	41h 42h 43h 2Eh 51h 50h 47h	
(数据名)										
可编程控制器侧										*
(示例)										

(2) 通过二进制代码进行通信时

(数据名)		指令	子指令	固定值	驱动器编号	更改日期	更改时间	文件名 字符数	文件名 (字符数)	
外部设备侧	*									*
(示例)		L H	L H		L H	L H	L H	L - - - - H	A B C . Q P G	
		26h 18h	00h 00h	30h 30h 30h 30h	00h 00h	22h 27h	00h 00h	07h 00h	41h 42h 43h 2Eh 51h 50h 47h	
(数据名)										
可编程控制器侧										*
(示例)										

要点

Q/LCPU 处于运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。

- 参数文件
- 程序存储器(驱动器名：00h)中当前正在执行的文件

3.8.15 文件属性的更改(指令：1825)：用于 Q/LCPU

以下对更改文件属性的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 对 QCPU 的文件属性进行更改时(子指令“0000”)

以下条件下的文件属性更改示例如下所示。
关键字 1234
驱动器 No. 0
属性更改文件名 ABC.QPG
更改属性 只读

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字	驱动器编号	更改属性	文件名 字符数	文件名 (字符数)	
外部设备侧	*								*
(示例)		H - - L	H - - L	L - - H	H - - L	H - - L	H - - L	L - - - - H	
		1 8 2 5	0 0 0 0	1 2 3 4	0 0 0 0	0 0 0 1	0 0 0 7	A B C . Q P G	
		31H 38H 32H 35H	30H 30H 30H 30H	31H 32H 33H 34H	30H 30H 30H 30H	30H 30H 30H 31H	30H 30H 30H 37H	41H 42H 43H 2EH 51H 50H 47H	
(数据名)									
可编程控制器侧									*
(示例)									

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	关键字	驱动器编号	更改属性	文件名 字符数	文件名 (字符数)	
外部设备侧	*								*
(示例)		L H	L H	L - - H	L H	L H	L H	L - - - - H	
		25H 18H	00H 00H	31H 32H 33H 34H	00H 00H	01H 00H	07H 00H	41H 42H 43H 2EH 51H 50H 47H	
(数据名)									
可编程控制器侧									*
(示例)									

(2) 对 LCPU 的文件属性进行更改时(子指令“0004”)

以下条件下的文件属性更改示例如下所示。
 关键字 AbCd1234_ . . . _ (空格 24 字符, 代码: 20H)
 驱动器 No. 0
 属性更改文件名 MAIN.QPG
 更改属性 只读

(a) 通过 ASCII 代码进行通信时

(数据名)		指令	子指令	关键字 (固定为32字符)	驱动器编号	更改属性	文件名 字符数	文件名 (字符数)	
外部设备侧	*								*
(示例)		H 1 8 2 5	L 0 0 0 4	L A b C d 1 2 3 4 _ . . . _	H 0 0 0 0	H 0 0 0 1	L 0 0 0 8	L M A I N . Q P G	
		31h 38h 32h 35h	30h 30h 30h 34h	41h 62h 43h 64h 31h 32h 33h 34h 20h	20h 30h 30h 30h 30h 30h 30h 31h	30h 30h 30h 38h	4Dh 41h 49h 4Eh 2Eh 51h 50h 47h		
(数据名)									
可编程控制器侧									*
(示例)									

(b) 通过二进制代码进行通信时

(数据名)		指令	子指令	关键字 (固定为32字符)	驱动器编号	更改属性	文件名 字符数	文件名 (字符数)	
外部设备侧	*								*
(示例)		L H L H	L	L	H	L H	L H L H	L	
		25h 18h 04h 00h	41h 62h 43h 64h	31h 32h 33h 34h 20h	20h	00h 00h 01h 00h 08h 00h	4Dh 41h 49h 4Eh 2Eh 51h 50h 47h		
(数据名)									
可编程控制器侧									*
(示例)									

要点
(1) Q/LCPU 处于运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 • 参数文件 • 程序存储器(驱动器名: 00H)中当前正在执行的文件
(2) 属性只能在 01H(只读文件 ↔ 20H(可读取、写入的磁盘文件)之间进行更改。

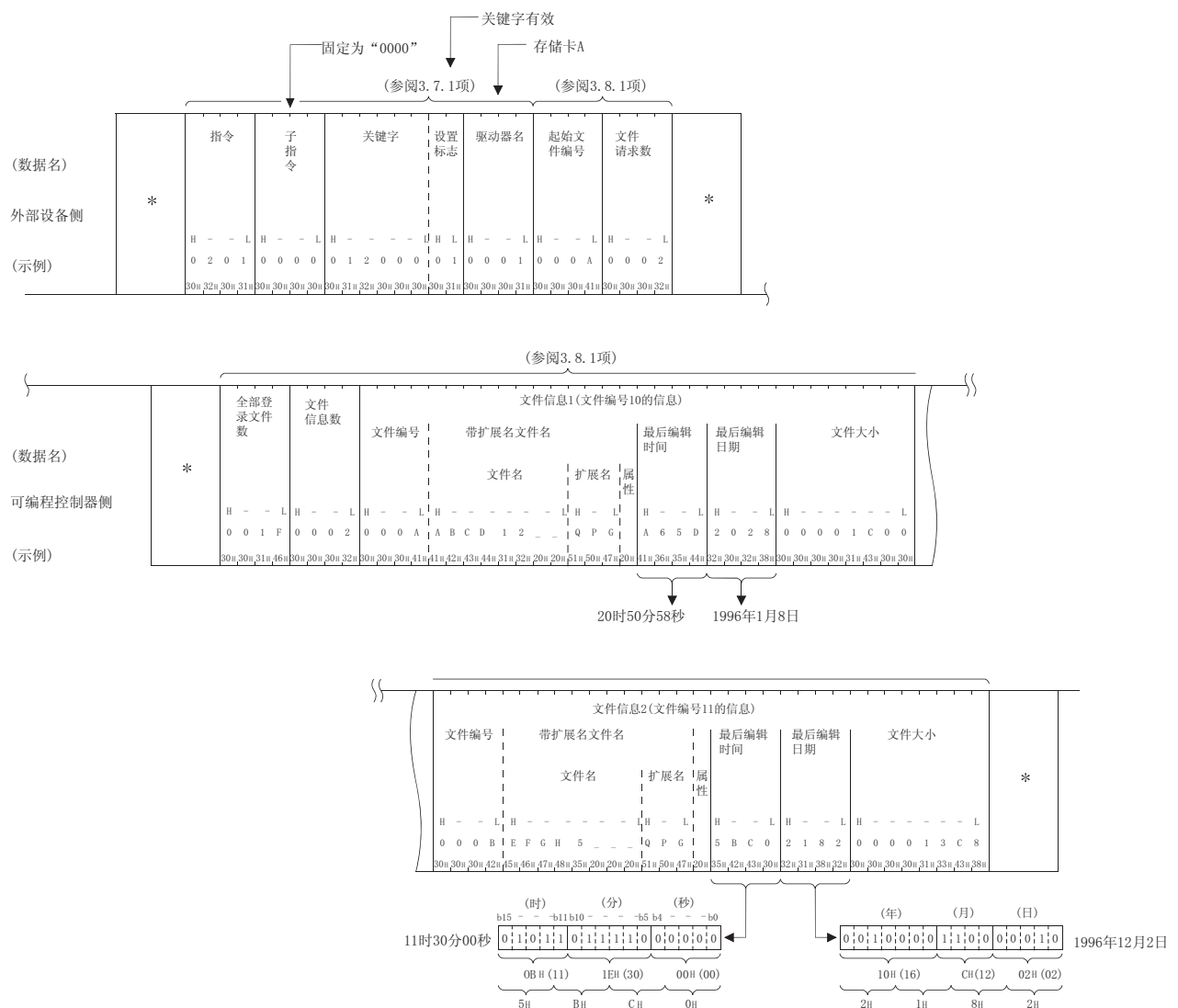
3.8.16 文件信息列表的读取：用于 QnACPU

(1) 无索引文件信息列表的读取(指令：0201)：用于 QnACPU

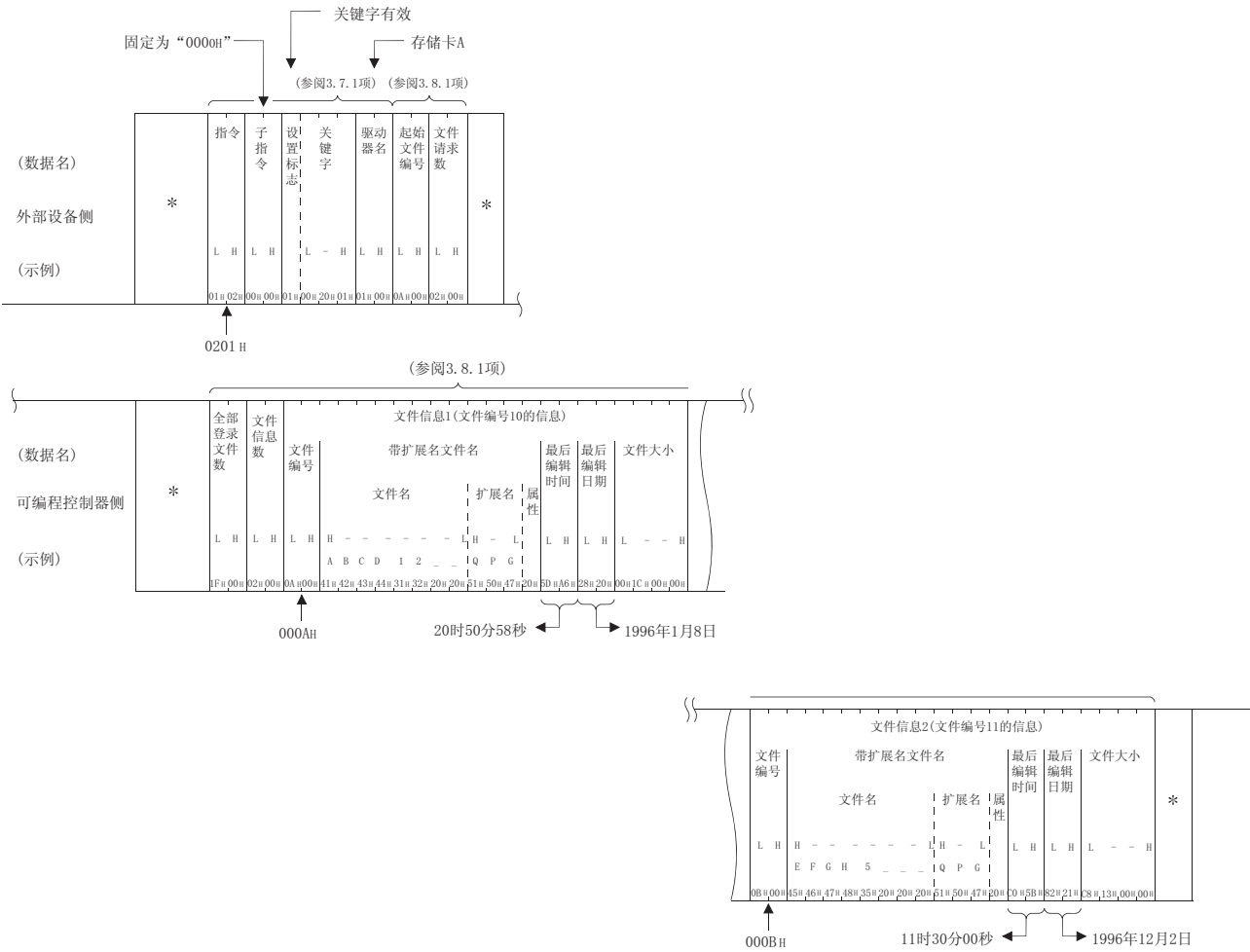
以下对指定文件编号范围的文件信息的读取控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(a) 通过由 ASCII 代码进行的通信，从存储卡 A(RAM 区域，驱动器名：01h)的文件编号 10(Ah)中读取 2 个文件信息的情况下



(b) 通过由二进制代码进行的通信，从存储卡 A(RAM 区域，驱动器名：01h)的文件编号 10(Ah)中读取 2 个文件信息的取情况下



要点

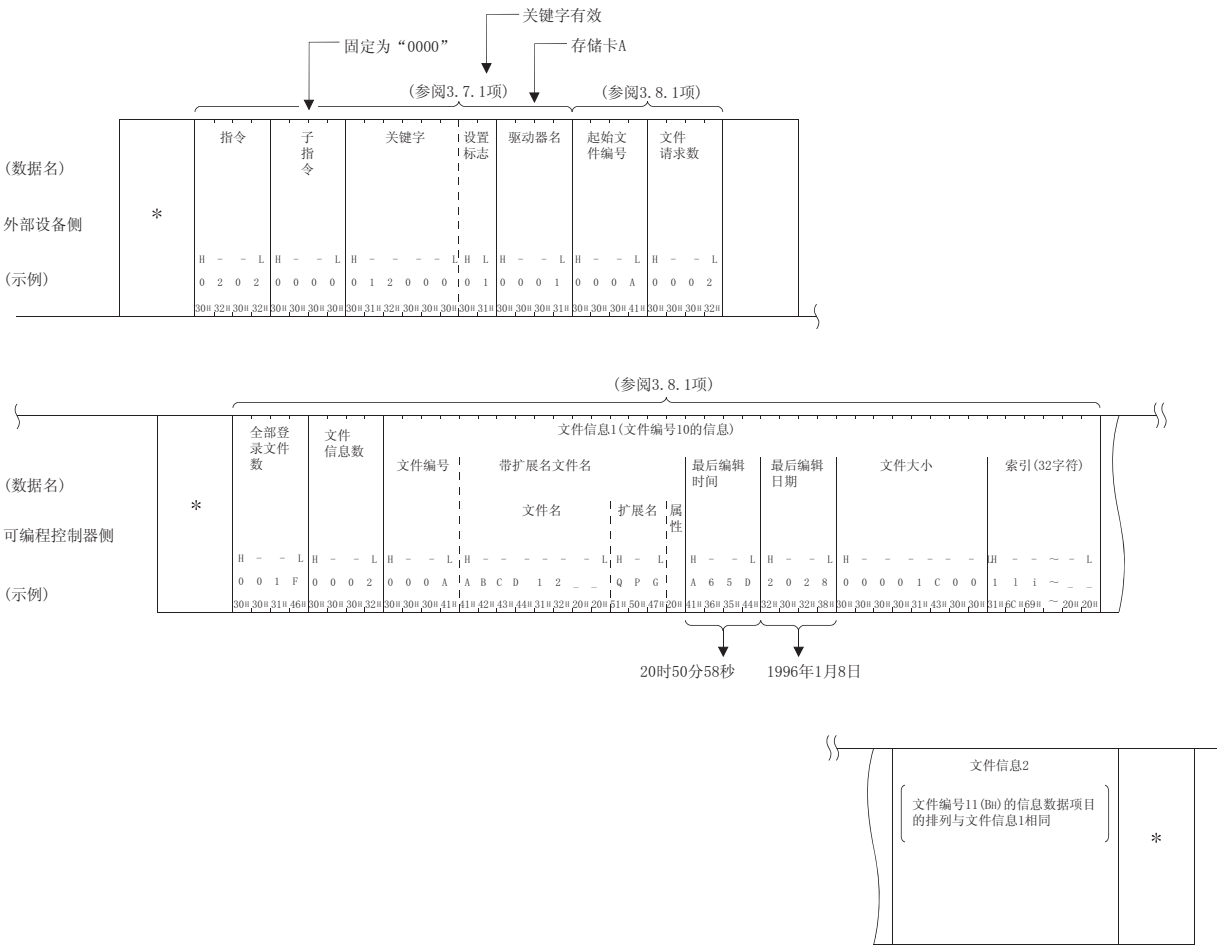
- (1) 各指定值应在以下范围内进行指定或返回。
 - 起始文件编号 : $1 \leq \text{文件编号} \leq 256$
 - 文件请求数 : $1 \leq \text{文件编号} \leq 36$
 - 全登录文件数 : $1 \leq \text{文件数} \leq 256$
 - 文件信息数 : $0 \leq \text{文件数} \leq \text{文件请求数}$ (0: 指定的起始文件编号以后无登录文件)
- (2) 全部登录文件数是指，当前指定驱动器中登录的全部文件数。
- (3) 指定的文件编号范围内未能登录所有文件时，文件信息数将变为指定范围内登录的文件数(返回的文件信息数)。

(2) 带索引文件信息列表的读取(指令：0202)：用于 QnACPU

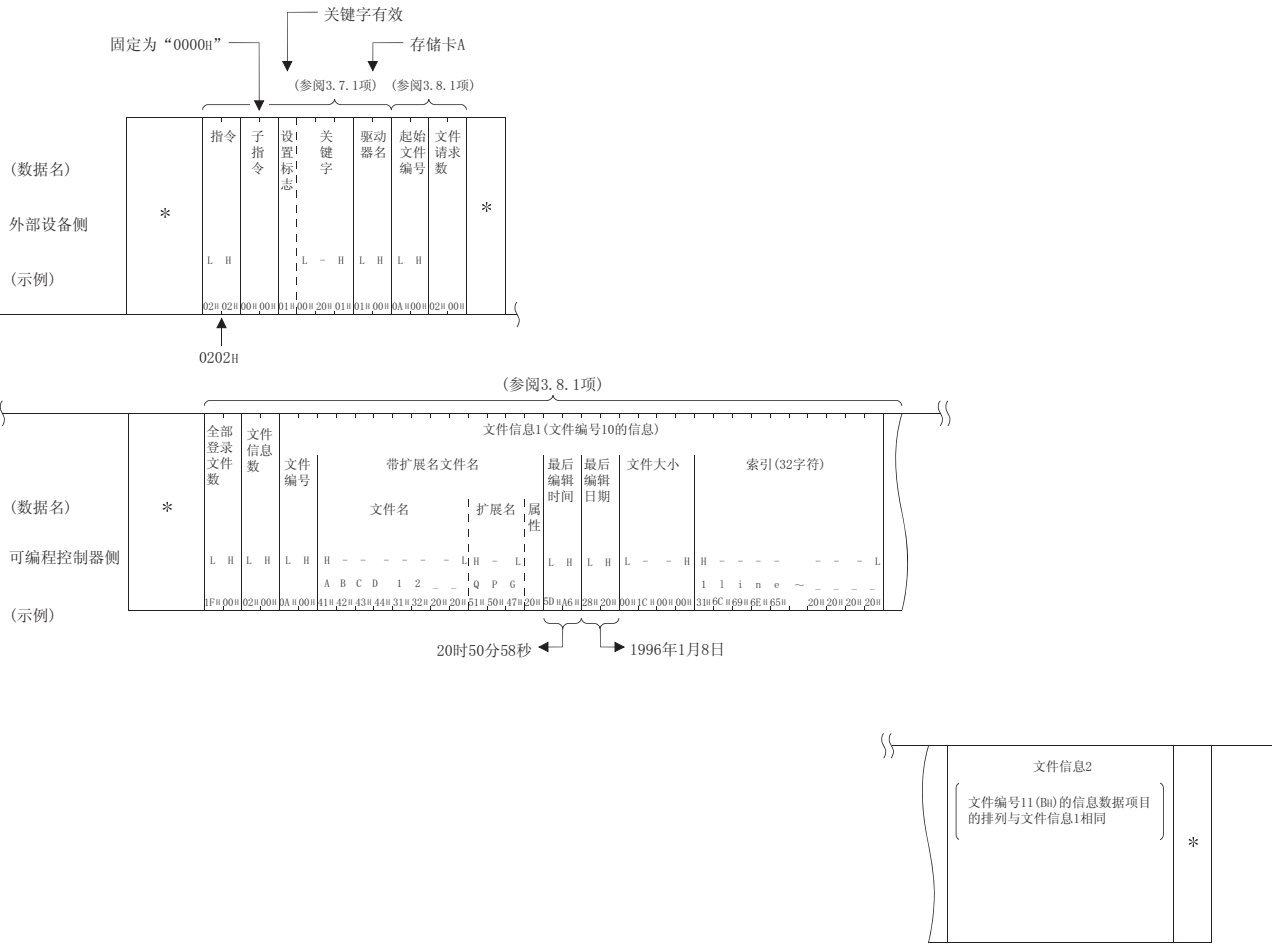
以下对指定文件编号范围的带索引文件信息的读取控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(a) 通过由 ASCII 代码进行的通信，从存储卡 A(RAM 区域，驱动器名：01h)的文件编号 10(Ah)中读取 2 个文件信息的情况下



(b) 通过由二进制代码进行的通信，从存储卡 A(RAM 区域，驱动器名: 01h)的文件编号 10(Ah)中读取 2 个文件信息的取情况下



要点

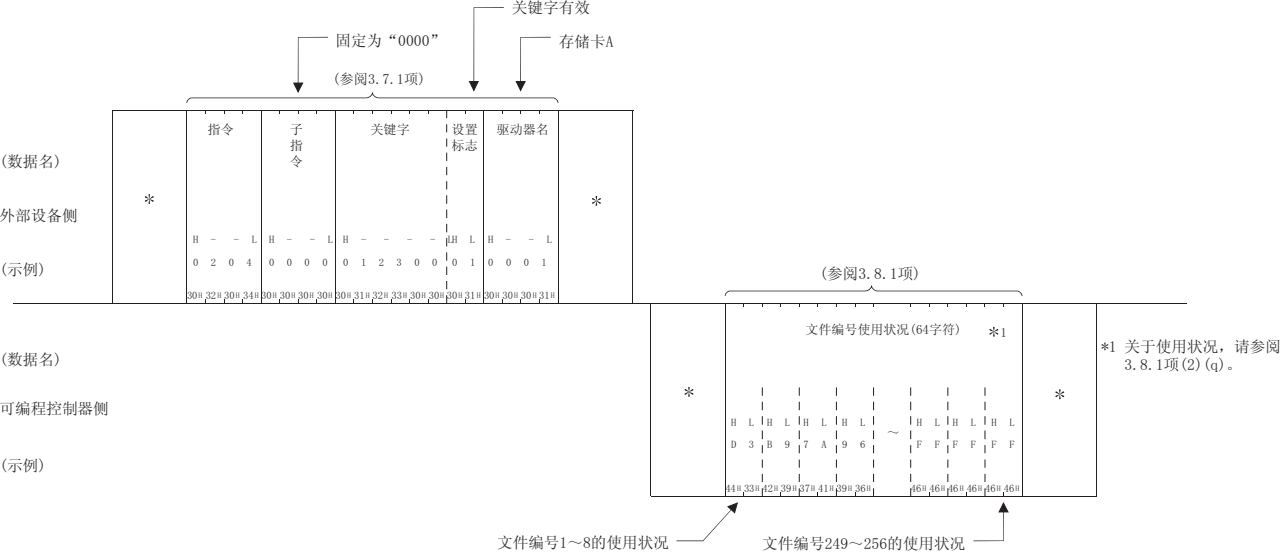
- 各指定值应在以下范围内进行指定或返回。
 - 起始文件编号 : $1 \leq \text{文件编号} \leq 256$
 - 文件请求数 : $1 \leq \text{文件编号} \leq 16$
 - 全登录文件数 : $1 \leq \text{文件数} \leq 256$
 - 文件信息数 : $0 \leq \text{文件数} \leq \text{文件请求数}$ (0: 指定的始文件编号以后无登录文件)
- 全部登录文件数是指，当前指定驱动器中登录的全部文件数。
- 指定的文件编号范围内未能登录所有文件时，文件信息数将变为指定范围内登录的文件数(返回的文件信息数)。

(3) 文件编号使用状况的读取(指令：0204)：用于 QnACPU

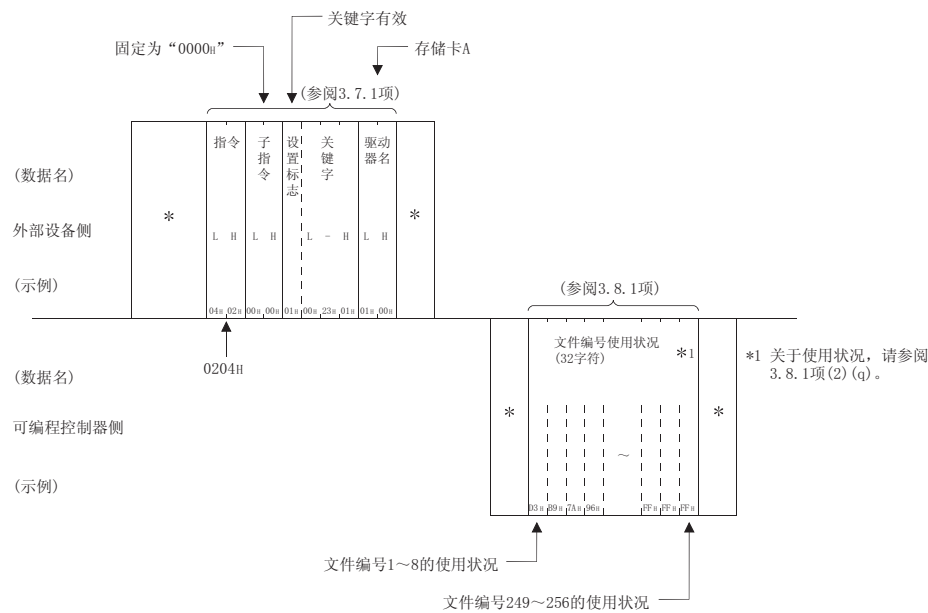
以下对文件编号使用状况的读取控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)的文件编号的使用状况进行读取的情况下



(b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)的文件编号的使用状况进行读取的情况下



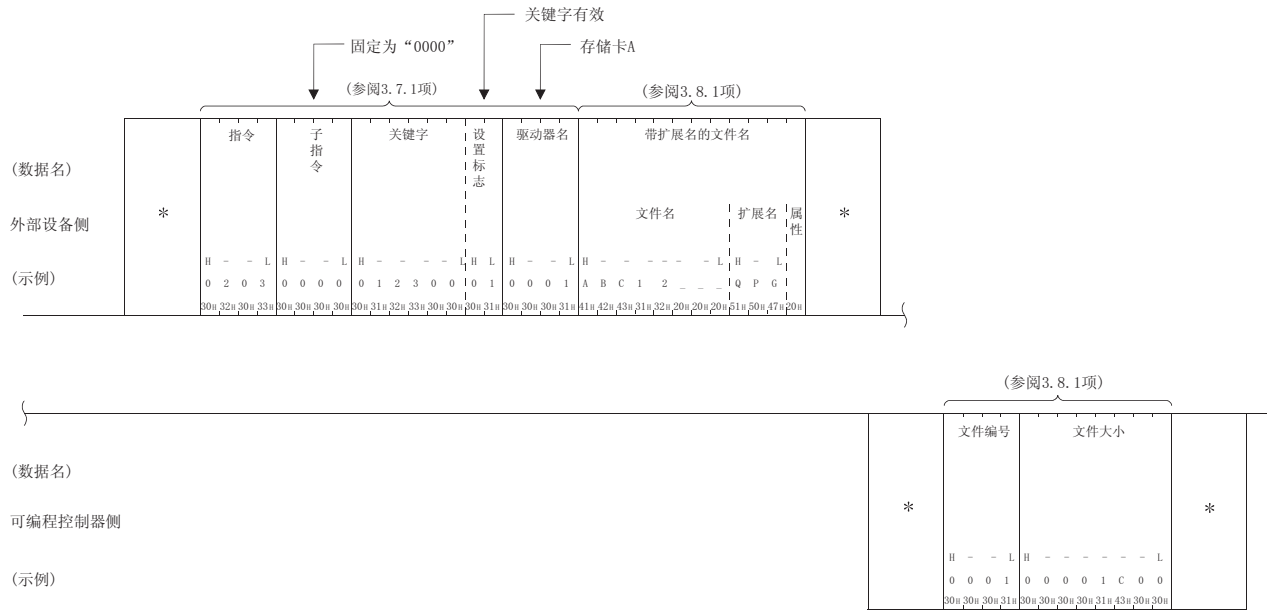
要点	
指定了不能存储最多 256 个文件的驱动器内存时，未能存储的文件个数部分(不足部分)的文件编号将变为使用中(相应位为 1)状态。	

3. 8. 17 文件有无的读取(文件搜索)(指令：0203)：用于 QnACPU

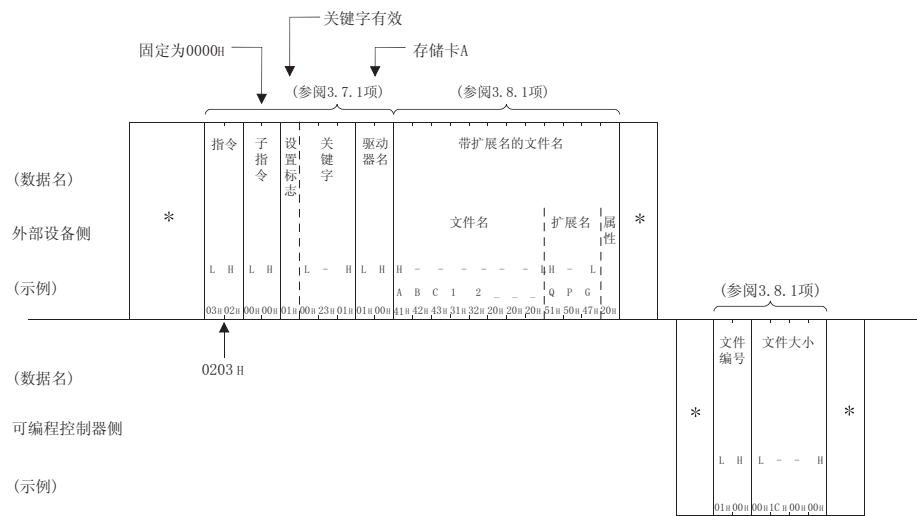
以下对指定文件的有无、存在时的文件编号、文件大小的读取控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3. 1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H) 中有无文件名为“ABC12. QPG”的文件进行读取的情况下



(2) 通过由二进制代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H) 中有无文件名为“ABC12. QPG”的文件进行读取的情况下



要点
(1) 读取文件的属性应作为虚拟键值处理。
(2) 指定文件不存在时将变为出错状态，将返回出错时的结束代码。

3. 8. 18 文件锁定的登录、解除(指令：0808)：用于 QnACPU

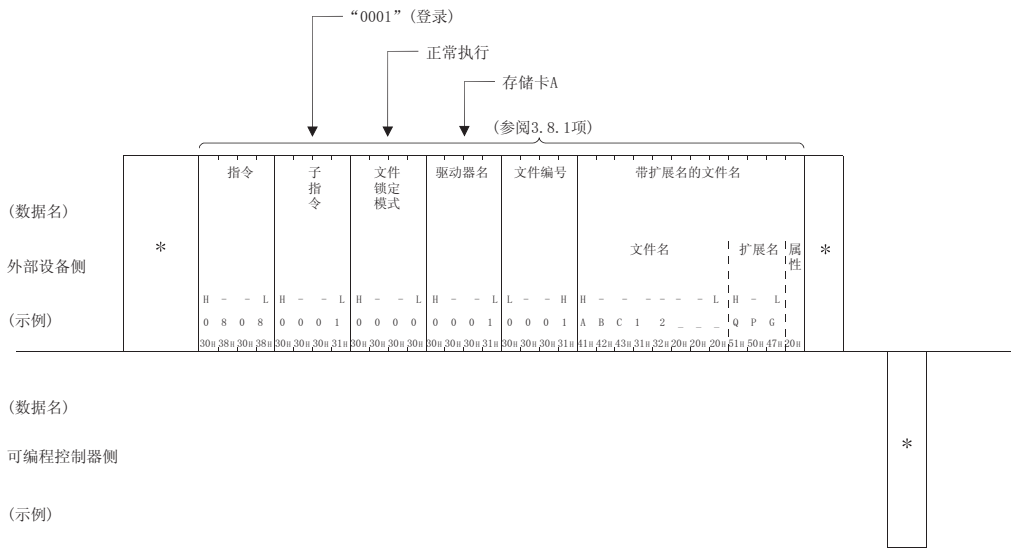
以下对进行如下所示的文件锁定的登录、解除时的控制步骤进行举例说明。

- 文件锁定登录(禁止来自于其它设备的访问)
在对指定文件进行访问的过程中，进行文件锁定登录，防止来自于其它设备等对文件内容进行更改，阻止来自于其它设备对同一文件进行访问。
- 文件锁定解除(允许来自于其它设备的访问)
对进行了文件锁定登录的文件进行文件锁定解除，允许从其它设备对同一文件进行访问。

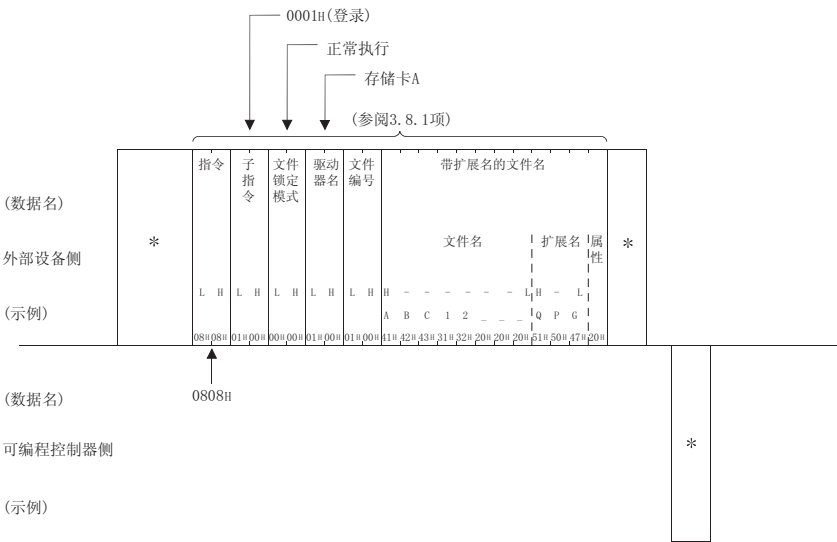
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3. 1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H)中的文件编号为 1、文件名为“ABC12.QPG”的文件以正常执行模式进行文件锁定登录的情况下



(2) 通过由二进制代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H) 中的文件编号为 1、文件名为“ABC12. QPG”的文件以正常执行模式进行文件锁定的情况下



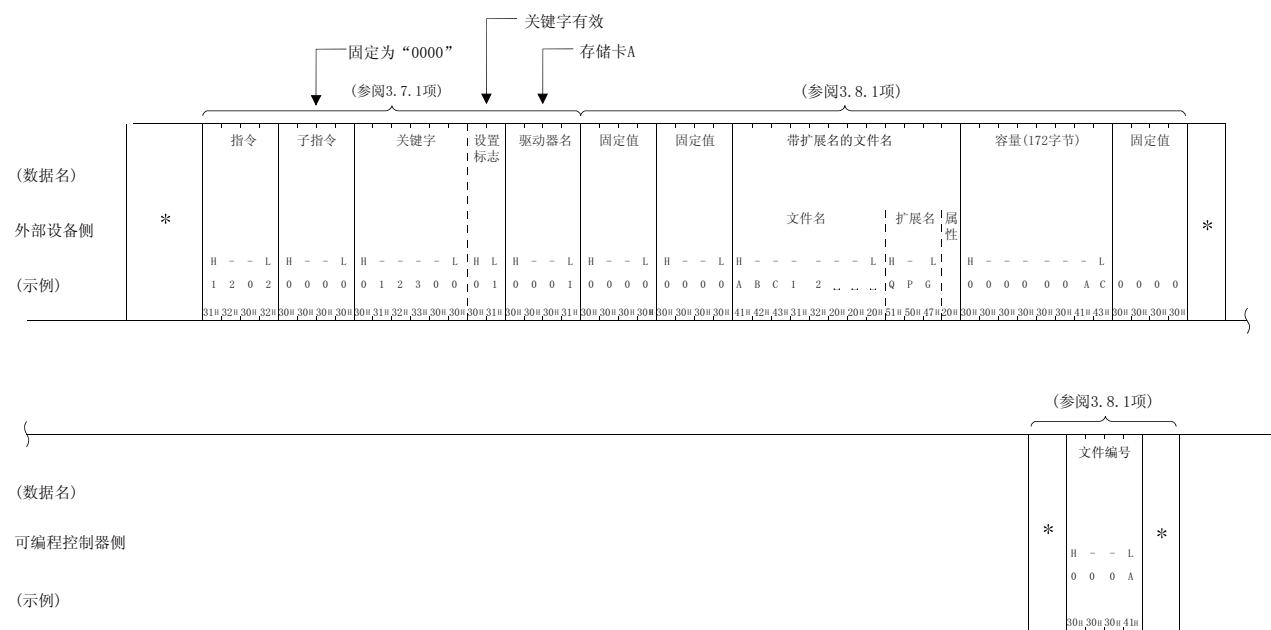
要点
(1) 对于进行文件锁定的登录、解除的文件的属性，新建等时附加的属性仍然有效。 进行文件锁定的登录、解除时，应作为虚拟键值处理。
(2) 处于文件锁定登录的状况下，QnACPU 的再启动 (CPU 复位等) 将使文件变为文件锁定解除状态。

3.8.19 文件内容的读取(指令：0206)：用于 QnACPU

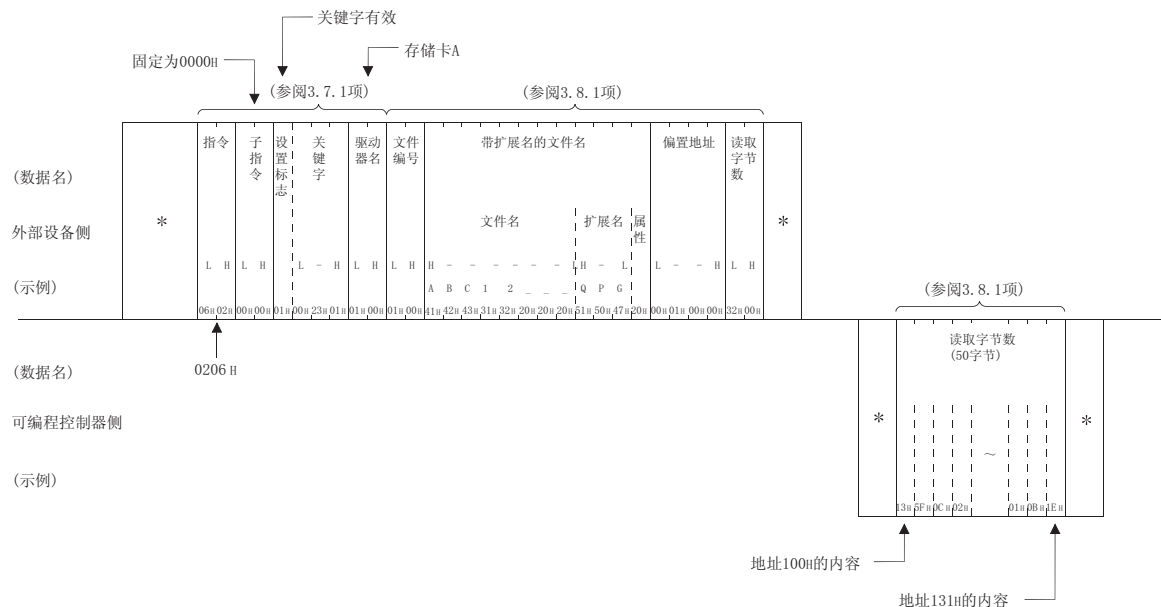
以下对读取指定文件中写入的数据的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，从存储卡 A (RAM 区域，驱动器名：01H) 的文件编号为 1、文件名为“ABC12.QPG”的偏置地址 100H 中读取 50 字节内容的情况下



(2) 通过由二进制代码进行的通信，从存储卡 A (RAM 区域，驱动器名：01H) 的文件编号为 1、文件名为“ABC12.QPG”的偏置地址 100H 中读取 50 字节的内容的情况下



要点

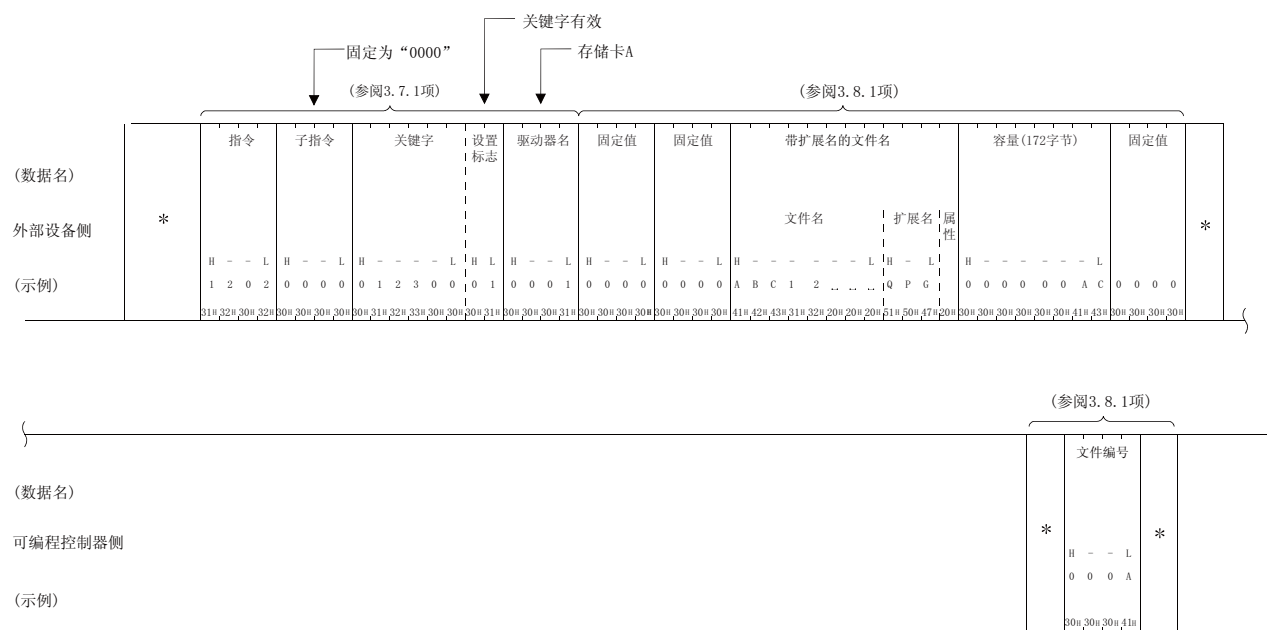
- (1) 进行数据读取时每次可读取的最多字节数是确定的。
对于写入到指定文件中的数据，应对偏置地址及读取字节数进行调整，分为若干次进行全部读取。
此外，对于读取到外部设备中的数据，应原样不变地进行存放保管。
对于文件的大小，可通过以下功能进行确认。
- 文件信息列表的读取功能 : 参阅 3.8.16 项
 - 文件有无的读取功能 : 参阅 3.8.17 项
- (2) 对于读取文件的属性，应作为虚拟键值处理。
- (3) 各指定值应在以下范围内进行指定。
- 文件编号 : $0 \leq \text{文件编号} \leq 256$
 - 偏置地址 : 在以下范围内以偶数地址进行指定
 $0 \leq \text{地址} \leq (\text{文件大小} - 1)$
 - 读取字节数 : $0 \leq \text{字节数} \leq 960$

3. 8. 20 文件的新建(文件名登录)(指令：1202)：用于 QnACPU

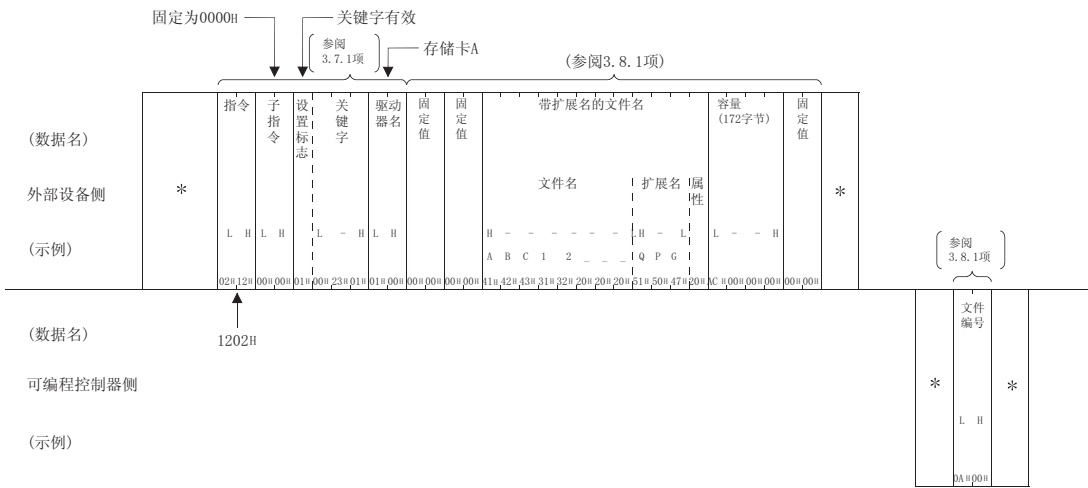
以下对将文件新登录到指定磁盘中，对文件区域进行预留时的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3. 1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，在存储卡 A (RAM 区域，驱动器名：01H) 中对文件名为“ABC12.QPG”、容量为 172 字节的文件进行新建登录，对文件区域进行预留的情况下



(2) 通过由二进制代码进行的通信，在存储卡 A (RAM 区域，驱动器名：01H) 中对文件名为“ABC12.QPG”、容量为 172 字节的文件进行新建登录，对文件区域进行预留的情况下



要点	
(1)	对于文件，可以在指定驱动器内存中连续的未使用簇的容量以内进行新建。新建时，建议对以后数据的添加容量也考虑在内。
(2)	新建文件的属性应指定为“20H”（允许读取、写入的磁盘文件）。
(3)	对于通过本功能新建的文件，应通过 3.8.21 项中所示的至文件的写入 (指令：1203) 功能进行数据写入。 不能对未写入数据的文件的内容进行读取。
(4)	在通过本功能新建的文件中，作为最后编辑日期时间将登录 QCPU 的管理时间。

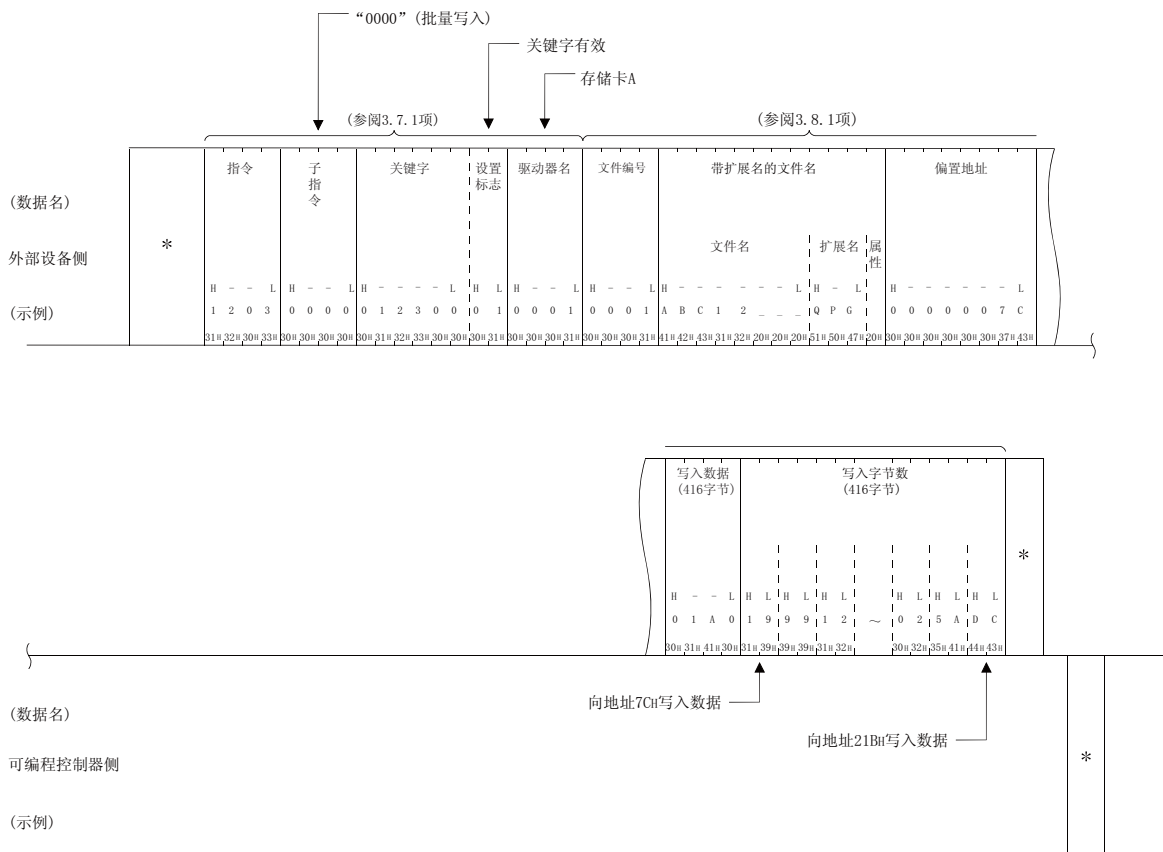
3. 8. 21 至文件的写入(指令：1203)：用于 QnACPU

(1) 批量写入

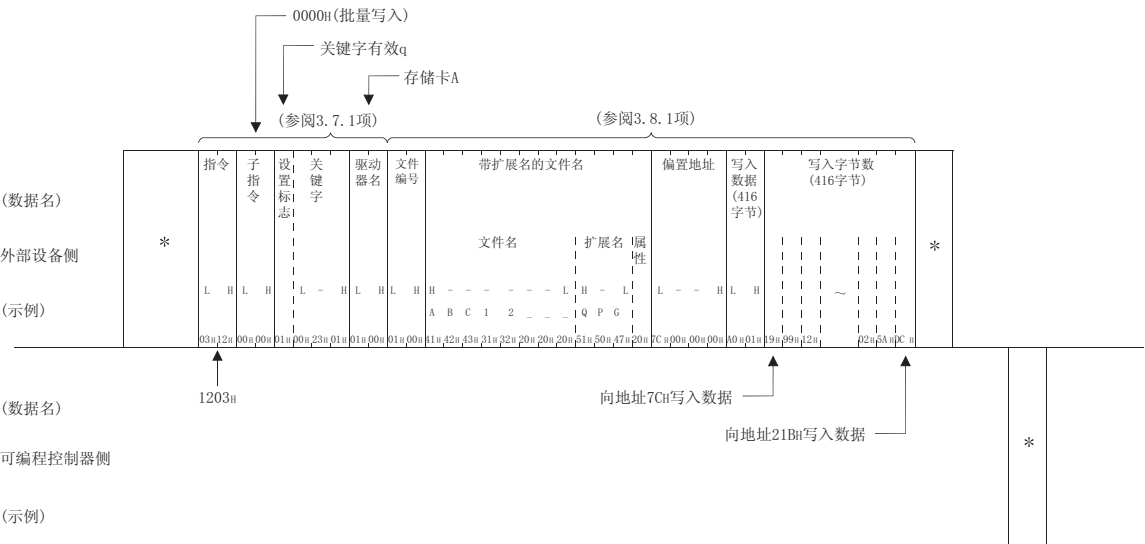
以下对将从 QnACPU 中读取的存放在外部设备中的文件数据写入到指定文件中的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3. 1 节中所示的详细说明。

[控制步骤]

(a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)中的文件编号为 1、文件名为“ABC12. QPG”的文件从偏置地址 7CH 开始写入 416 字节的数据的情况下



(b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名: 01h)中的文件编号为 1、文件名为“ABC12.QPG”的文件从偏置地址 7Ch 开始写入 416 字节的数据的情况下



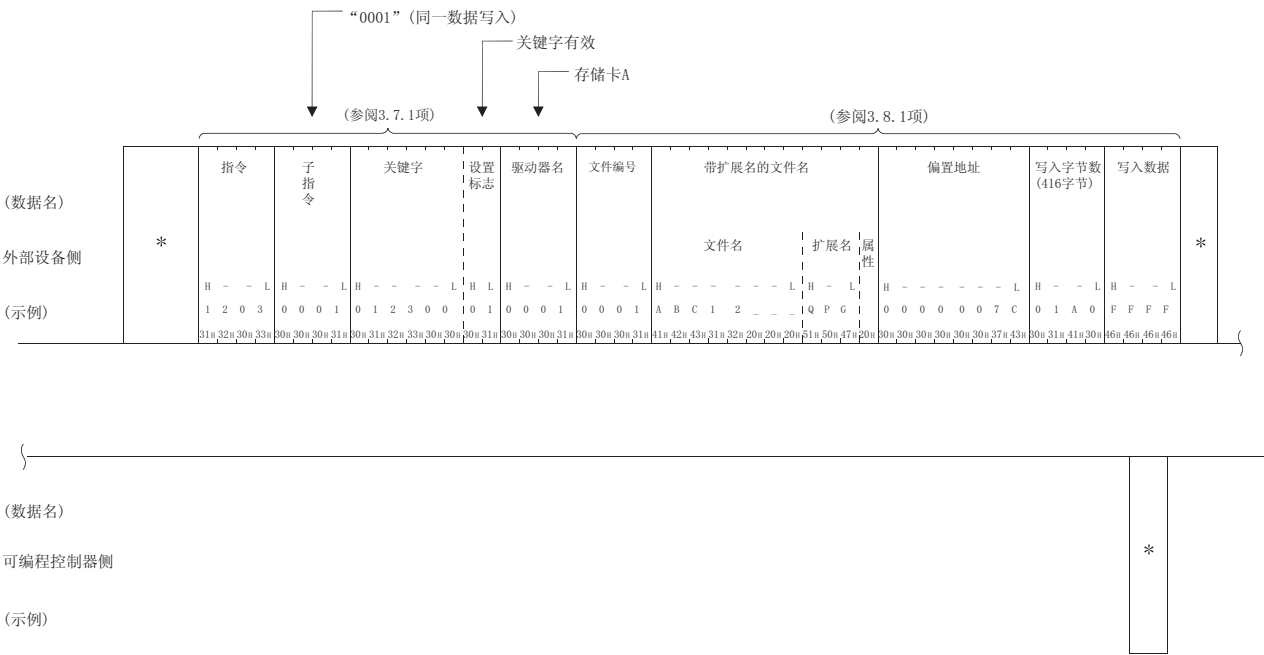
要点	
(1)	写入数据时每次可写入的最多字节数是确定的。 对于从 QnACPU 中读取后存放 to 外部设备中的数据，应对偏置地址及写入字节数进行调整，分为若干次全部写入到指定文件中。
(2)	对于写入了数据的文件的属性，新建等时附加的属性仍然有效。 数据写入时，应作为虚拟键值处理。
(3)	QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 • 参数文件 • 内置 RAM(驱动器名: 00h)的当前正在执行的文件
(4)	各指定值应在以下范围内进行指定。 • 偏置地址 : 0≤地址≤(文件大小-1) 对驱动器名为“00h”(内置 RAM)的文件进行写入时，在上述范围内以 4 的倍数(10 进制数的情况下，0、4、8、...)的地址进行指定。 对驱动器名为除“00h”以外的文件进行写入时，在上述范围内以偶数地址(10 进制数的情况下，0、2、4、6、8、...)进行指定。 • 写入字节数 : 0≤字节数≤960

(2) 同一数据写入(FILL)

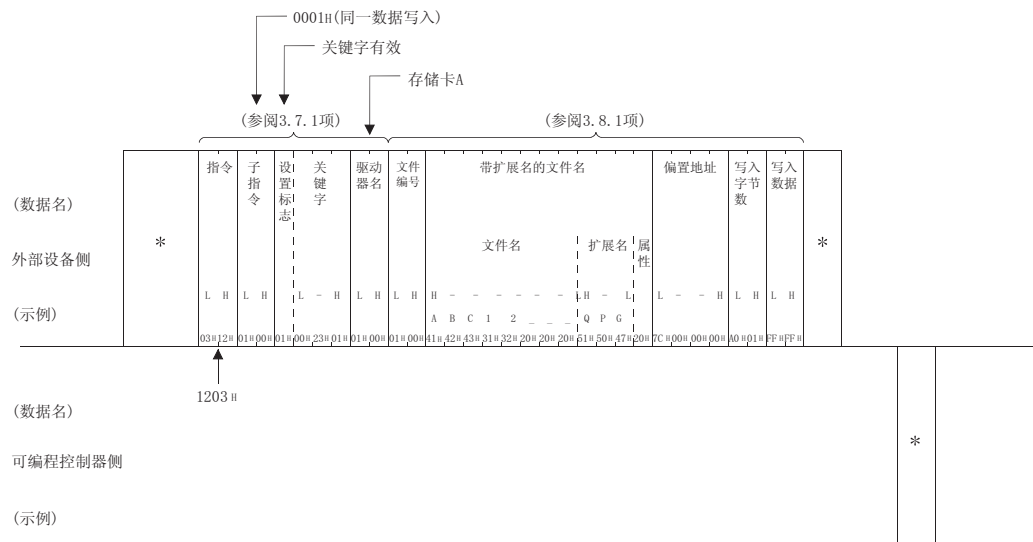
以下对将 n 字节的任意的 1 字数据写入到指定文件中的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名: 01h)中的文件编号为 1、文件名为“ABC12.QPG”的文件，从偏置地址 7CH 开始写入 416 字节的“FFFFh”的情况下



(b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名: 01h)中的文件编号为 1、文件名为“ABC12.QPG”文件，从偏置地址 7Ch 开始写入 416 字节的“FFFFh”的情况下



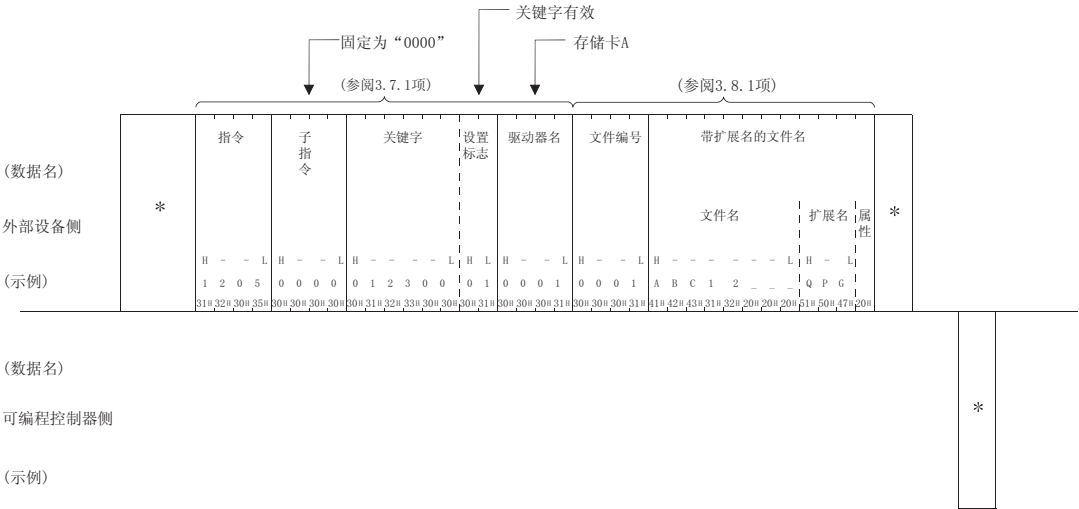
要点	
(1)	写入数据时每次可写入的最多字节数是确定的。 对于任意的 1 字数据，应对偏置地址及写入字节数进行调整，分为若干次在指定文件的大小以内进行写入。 此外，由于数据的写入是以字节单位进行，因此数据写入剩余的容量为 1 字节时，1 字数据的高位字节(位 8~15)的数值将不被写入。
(2)	对于写入了数据的文件的属性，新建等时附加的属性仍然有效。 数据写入时，应作为虚拟键值处理。
(3)	QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 - 参数文件 - 内置 RAM(驱动器名: 00h)的当前正在执行的文件
(4)	各指定值应在以下范围内进行指定。 - 偏置地址 : 0≤地址≤(文件大小-1) 对驱动器名为“00h”(内置 RAM)的文件进行写入时，在上述范围内以 4 的倍数(10 进制数的情况下，0、4、8、...)的地址进行指定。 对驱动器名为除“00h”以外的文件进行写入时，在上述范围内以偶数地址(10 进制数的情况下，0、2、4、6、8、...)进行指定。 - 写入字节数 : 0≤字节数≤960

3.8.22 文件的删除(指令：1205)：用于 QnACPU

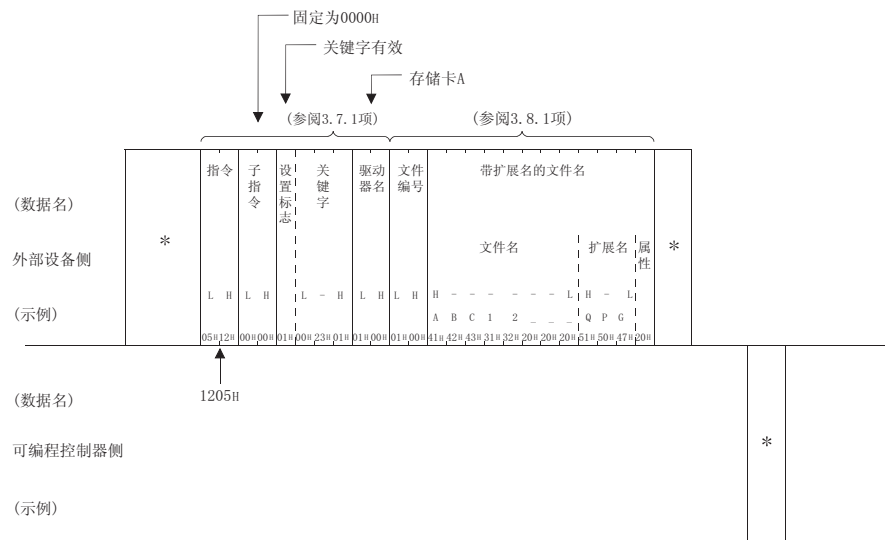
以下对删除现有文件的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H) 中的文件名为“ABC12. QPG”的现有文件进行删除的情况下



(2) 通过由二进制代码进行的通信，对存储卡 A (RAM 区域，驱动器名：01H) 中的文件名为“ABC12. QPG”的现有文件进行删除的情况下



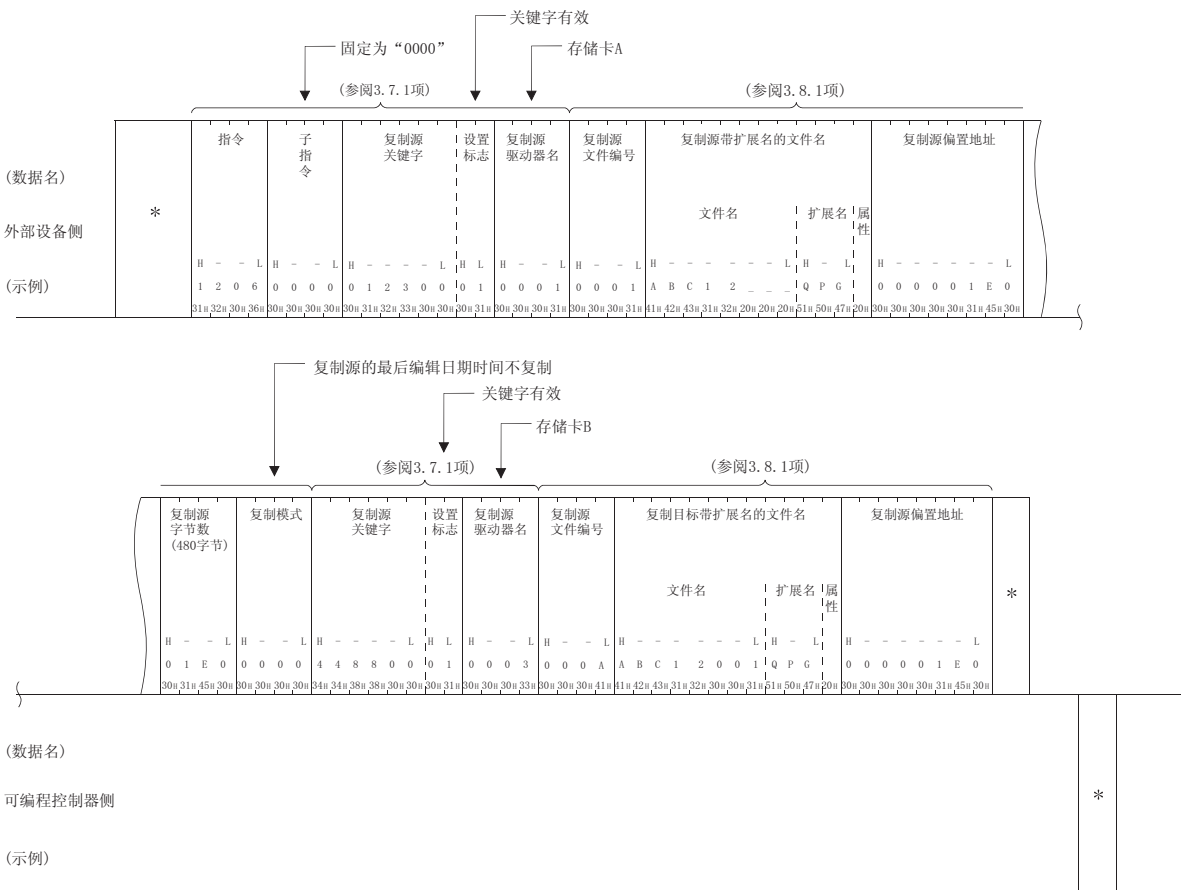
要点
(1) 对于删除的文件的属性，新建等时附加的属性仍然有效。 删除时，应作为虚拟键值处理。
(2) 对于文件的删除时机，应由包含 QnACPU 及关联设备在内的整个系统确定。
(3) 无法删除进行了文件锁定登录的文件。
(4) QnACPU 处于“运行中”时，无法删除以下文件。 • 程序文件(□. QPG) • 参数文件(□. QPA) • 引导设置文件(□. QBT)

3.8.23 文件的复制(指令：1206)：用于 QnACPU

以下介绍将写入到现有文件中的数据写入(覆盖)到新登录的文件中的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (1) 通过由 ASCII 代码进行的通信，将存储卡 A (RAM 区域，驱动器名：01H) 内的现有文件的内容复制到新登录文件中的情况下
无论在何种情况下均指定偏置地址 1E0H，复制 480 字节。
现有文件 文件编号：0001H (1)，文件名：“ABC12.QPG”
新登录文件 文件编号：000AH (10)，文件名：“ABC12001.QPG”
(复制源文件的最后编辑日期时间不复制。)

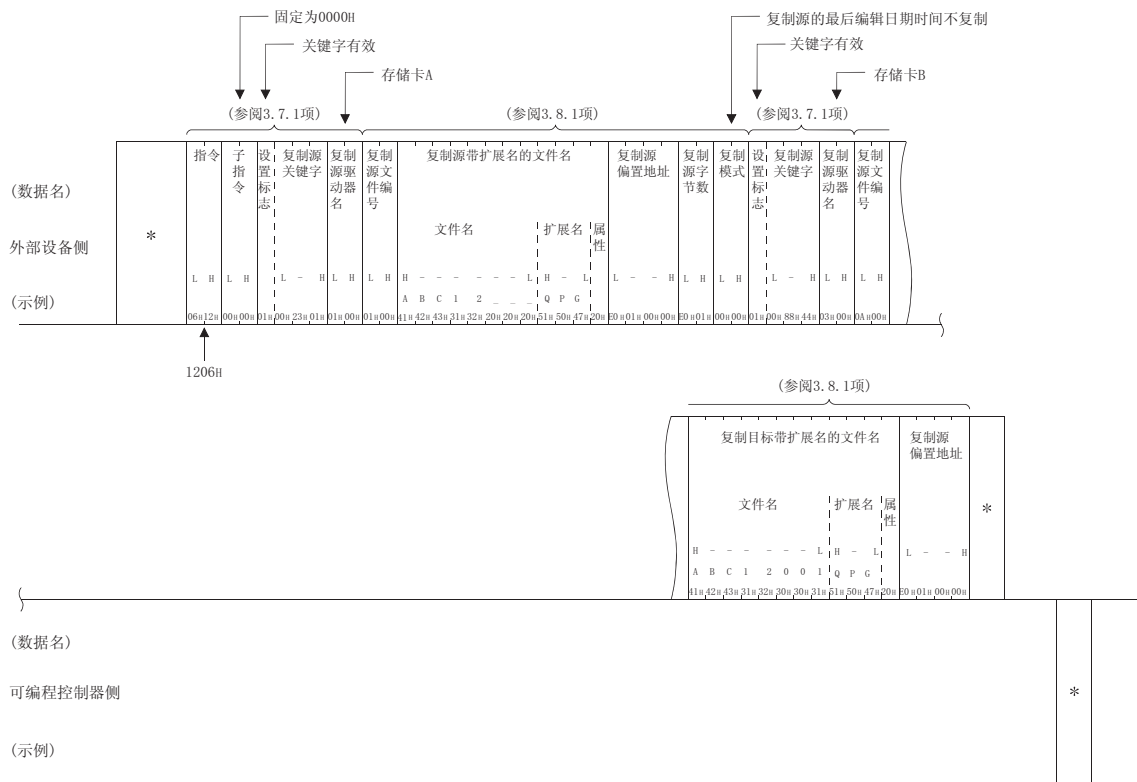


(2) 通过由二进制代码进行的通信，将存储卡 A(RAM 区域，驱动器名：01H)内的现有文件的内容复制到新登录文件中的情况下

无论在何种情况下均指定偏置地址 1E0H，复制 480 字节。

现有文件 文件编号：0001H (1)， 文件名 ：“ABC12.QPG”

新登录文件 文件编号：000AH (10)，文件名 ：“ABC12001.QPG”



要点	
(1)	每次复制时的最多字节数是确定的。 对于写入到现有文件中的数据，应对偏置地址及复制字节数进行调整，分为若干次全部写入到新登录文件中。
(2)	对于复制源以及复制目标文件的属性，新建等时附加的属性仍然有效。 复制时，应作为虚拟键值处理。
(3)	QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 - 参数文件 - 内置 RAM(驱动器名：00H)的当前正在执行的文件
(4)	各指定值应在以下范围内进行指定。 - 偏置地址 : 0 ≤ 地址 ≤ (文件大小-1) 指定驱动器名为“00H”(内置 RAM)的文件时，在上述范围内以 4 的倍数(10 进制数的情况下，0、4、8、...)的地址进行指定。 指定驱动器名为除“00H”以外的文件时，在上述范围内以偶数地址(10 进制数的情况下，0、2、4、6、8、...)进行指定。 - 复制字节数 : 0 ≤ 字节数 ≤ 480

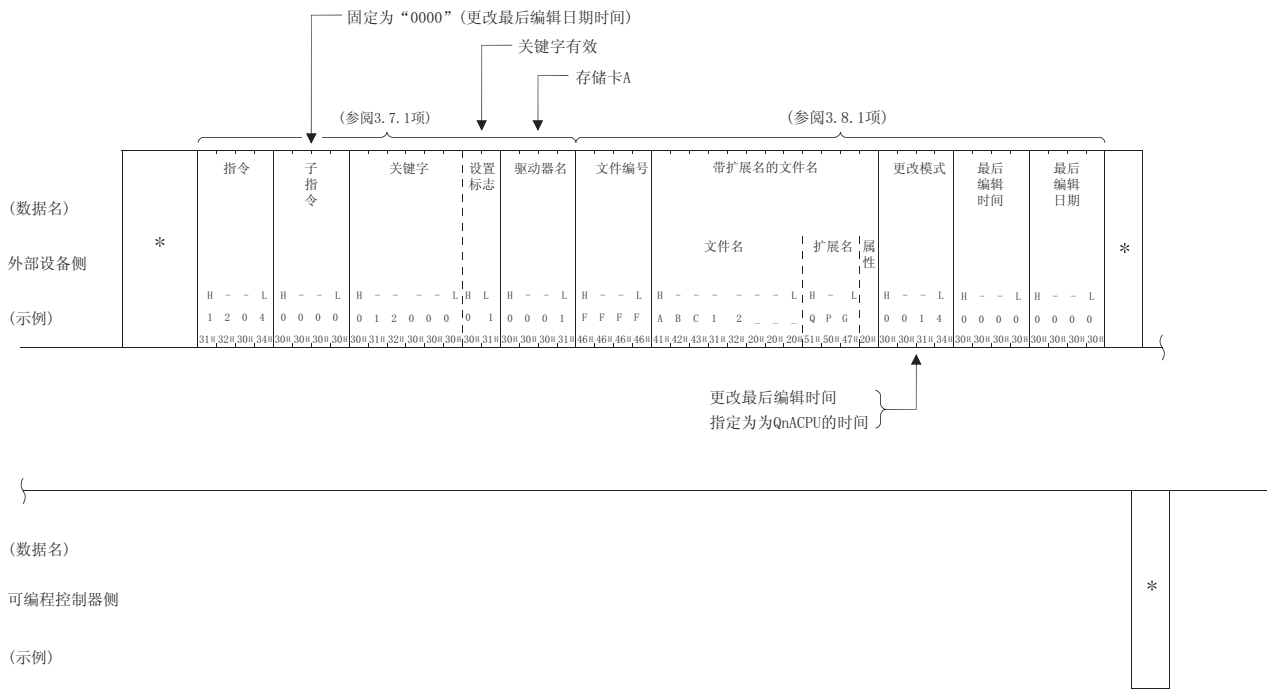
3.8.24 文件信息的更改(指令：1204)：用于 QnACPU

(1) 文件创建日期时的更改(指令：1204，子指令：0000)

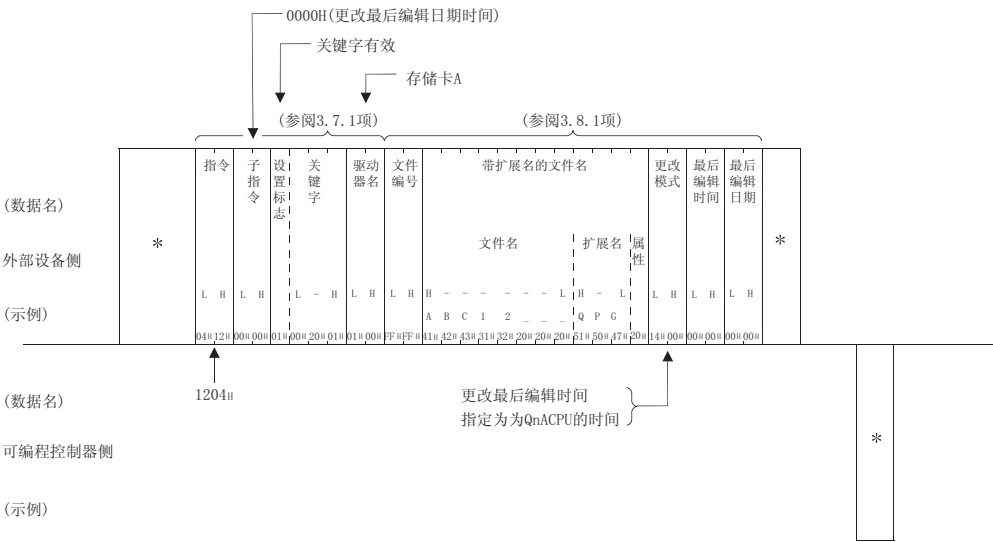
以下对更改指定文件的最后编辑日期及时间的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)中的文件名为“ABC12.QPG”的最后编辑时间进行更改的情况下
(文件编号指定为不明，最后编辑时间指定为 QnACPU 的时间)



(b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)中的文件名为“ABC12.QPG”的最后编辑时间进行更改的情况下
(文件编号指定为不明，最后编辑时间指定为 QnACPU 的时间)



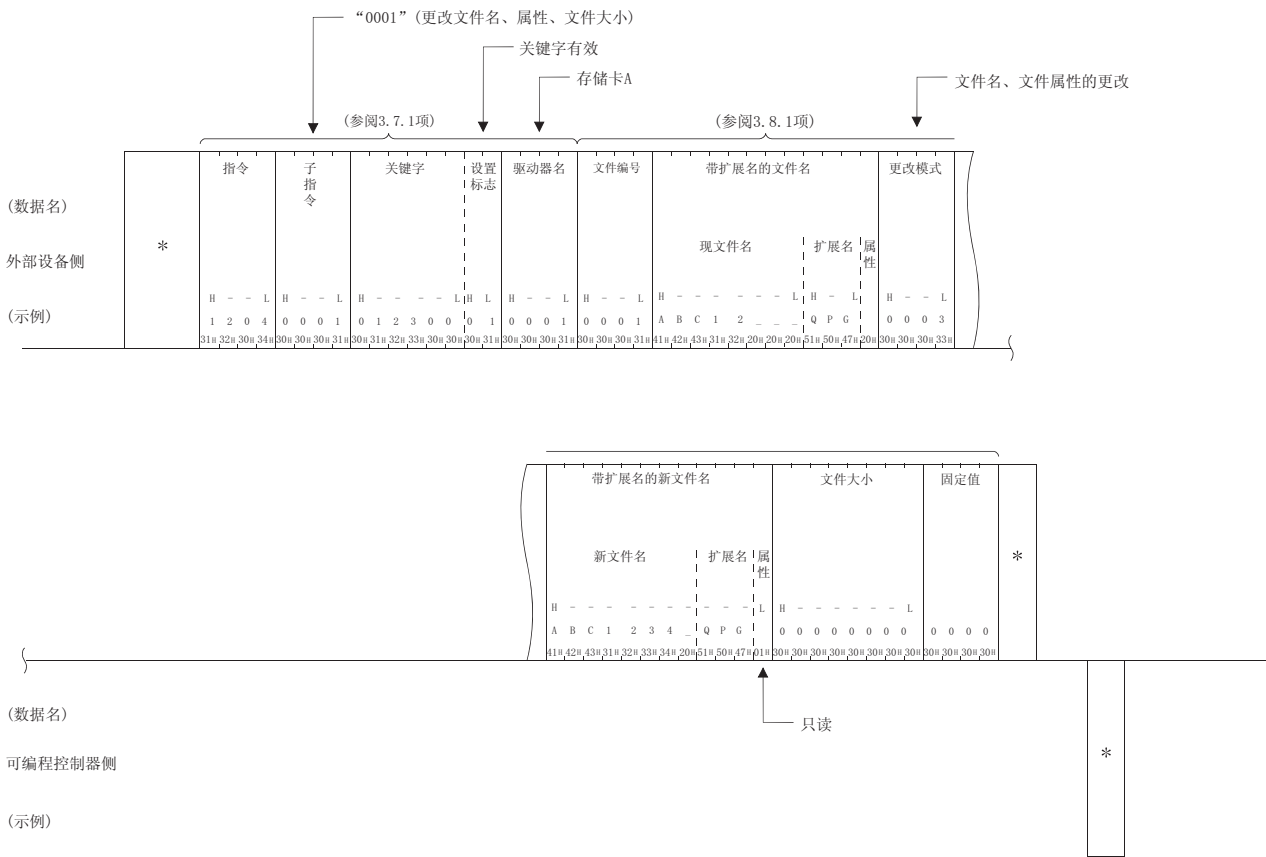
要点
(1) 各指定值应在以下范围内进行指定或返回。 - 文件编号 : 1≤文件编号≤256 - 更改模式 : 04h、08h、0Ch、14h、18h 或 1Ch
(2) QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 - 参数文件 - 内置 RAM(驱动器名：00h)的当前正在执行的文件

(2) 文件名、属性、文件大小的更改(指令：1204，子指令：0001)

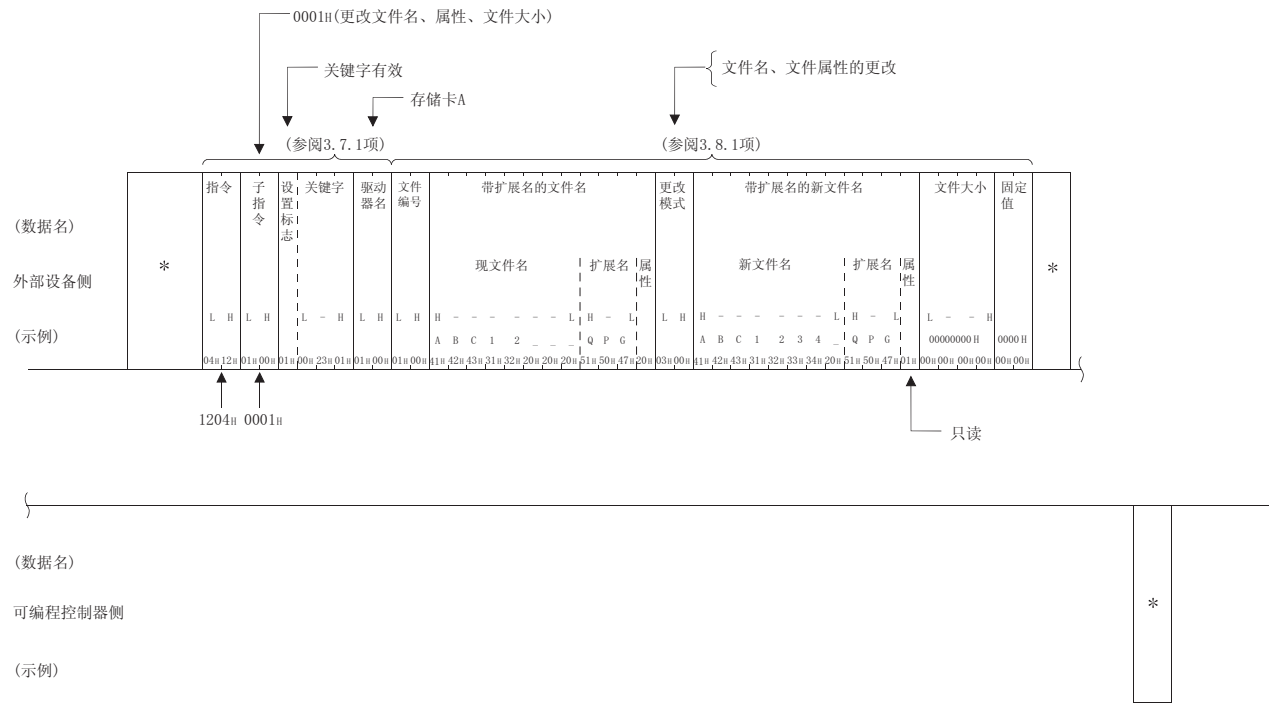
以下对更改指定文件的文件名、属性、文件大小的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)中的文件编号为 1、文件名为“ABC12. QPG”的文件名、属性进行更改的情况下(文件名更改为“ABC1234. QPG”，属性更改为只读文件)



(b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名: 01h)中的文件编号为 1、文件名为“ABC12. QPG”的文件名、属性进行更改的情况下(文件名更改为“ABC1234. QPG”，属性更改为只读文件)



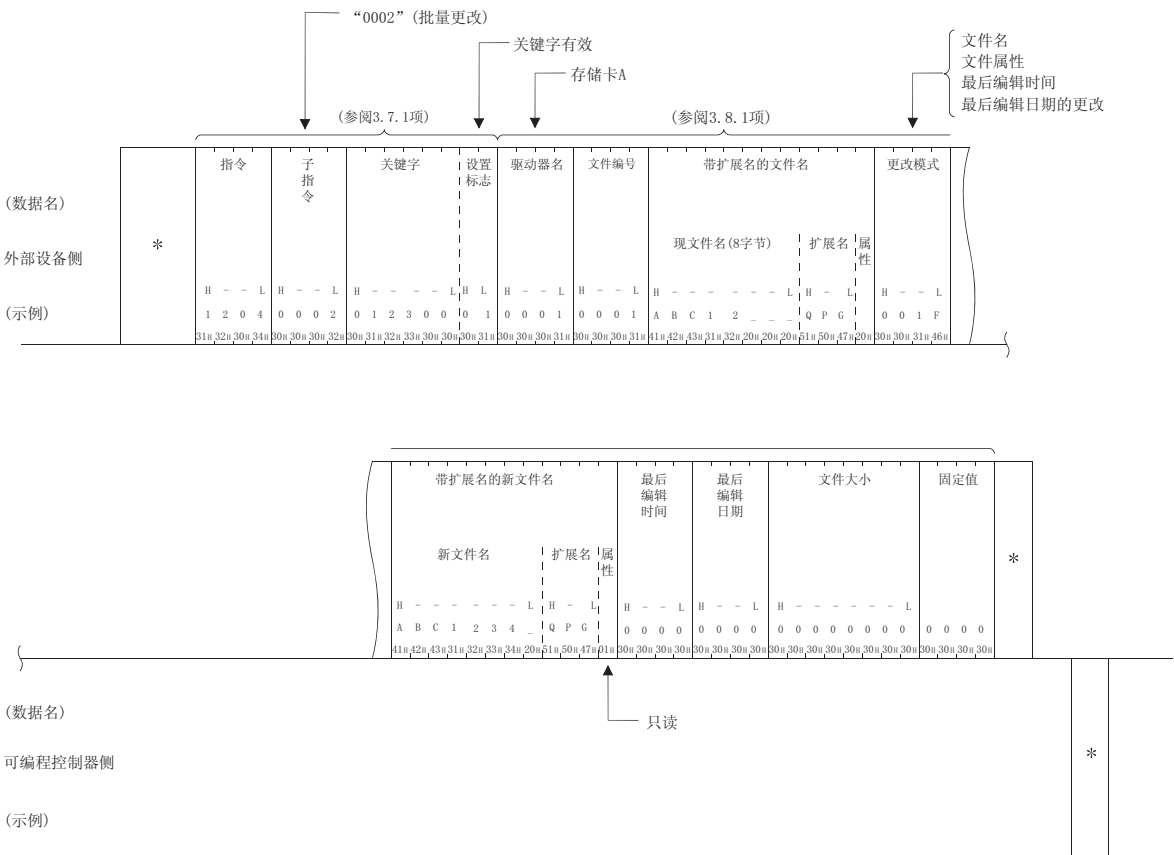
要点
(1) 各指定值应在以下范围内进行指定或返回。 • 文件编号 : 1≤文件编号≤256 • 更改模式 : 01h、02h、03h、20h、21h、22h 或 23h
(2) QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。 • 参数文件 • 内置 RAM(驱动器名: 00h)的当前正在执行的文件
(3) 属性只能在 01h(只读文件) ←→ 20h(允许读取、写入的磁盘文件)之间进行更改。
(4) 大小更改只能在 QnACPU 处于 STOP 中时才可进行，指定驱动器中需要有指定容量的连续空余区域。 对于空余区域，可通过 3.7.2 项中所示的存储器使用状态的读取进行确认。

(3) 文件信息的批量更改(指令：1204，子指令：0002)

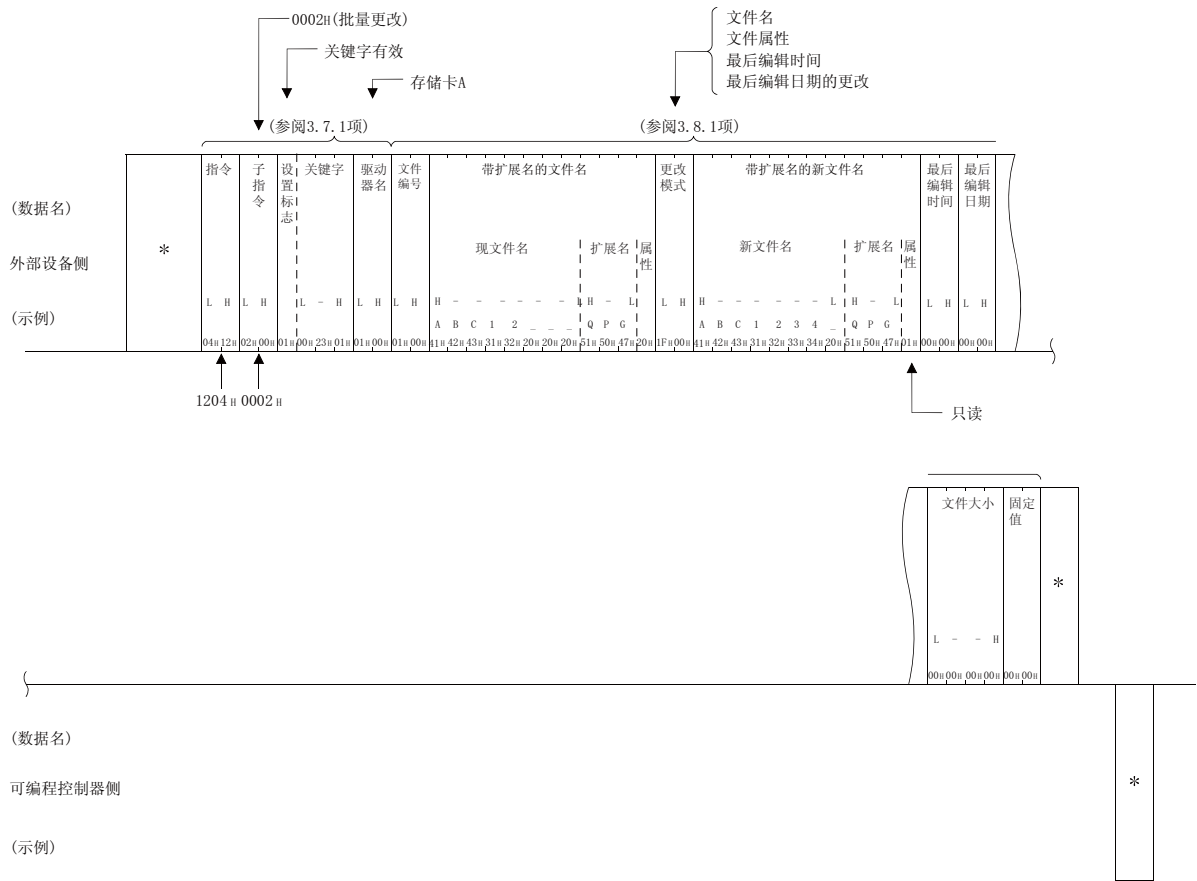
以下对指定文件的文件信息批量更改的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

- (a) 通过由 ASCII 代码进行的通信，对存储卡 A(RAM 区域，驱动器名：01h)中的文件编号为 1、文件名为“ABC12.QPG”的文件名、属性、最后编辑日期时间进行更改的情况下
(最后编辑日期时间使用 QnACPU 的日期及时间。属性更改为只读文件。)



- (b) 通过由二进制代码进行的通信，对存储卡 A(RAM 区域，驱动器名: 01H)中的文件编号为 1、文件名为“ABC12.QPG”的文件名、属性、最后编辑日期时间进行更改的情况下
(最后编辑日期时间使用 QnACPU 的日期及时间。属性更改为只读文件。)



要点

- (1) 各指定值应在以下范围内进行指定或返回。
- 文件编号 : $1 \leq \text{文件编号} \leq 256$
 - 更改模式 : $01H \leq \text{更改模式} \leq 3FH$
- (2) QnACPU 为运行中的情况下，如果指定以下文件将变为出错状态，并返回出错时的结束代码。
- 参数文件
 - 内置 RAM(驱动器名: 00H)的当前正在执行的文件
- (3) 属性只能在 01H(只读文件) $\leftrightarrow$ 20H(允许读取、写入的磁盘文件)之间进行更改。
- (4) 大小更改只能在 QnACPU 处于 STOP 中时才可进行，指定驱动器中需要有指定容量的连续空余区域。
对于空余区域，可通过 3.7.2 项中所示的存储器使用状态的读取进行确认。

3.9 用户登录帧的登录、删除、读取：用于串行通信模块

用户登录帧是用于将通过如下所示功能进行外部设备与 Q/LCPU 的数据通信时的报文的起始部分以及最终部分设置为用户确定的数据格式。

- 通过 MC 协议进行的接通请求数据的发送
- 通过无顺序协议进行的数据的发送、接收

在本节中，对用于外部设备对 C24 进行用户登录帧的登录、删除、读取的指令的使用方法有关内容进行说明。

关于通过用户登录帧进行的数据的发送、接收的有关内容，请参阅 C24 的用户手册(应用篇)。

要点
(1) 关于用户登录帧的内容，请参阅 C24 的用户手册(应用篇)。
(2) 只能对与外部设备相连接的 C24(多点连接站也包含在内)使用本功能。 对于经由网络系统的其它站 C24，不能使用本功能。
(3) 对于本节的通信，从外部设备发出了登录/删除/读取请求时将立即执行，不等待可编程控制器 CPU 的 END 处理。

3.9.1 指令及字符部分的内容

以下介绍外部设备对 C24 进行用户登录帧的登录等时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)有关内容。

(1) 指令

功能		指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
					STOP 中	RUN 中		
						写入允许 设置	写入禁止 设置	
用户登录帧	数据登录	1610 (0000)	对用户登录帧(数据的排列) 进行登录。	80 字节	○	○	○	3.9.2 项
	登录数据删除	1610 (0001)	对指定帧编号的用户登录帧进行删除。	(1 个)				
	登录数据读取	0610 (0000)	对指定帧编号的登录帧进行读取。	80 字节				3.9.3 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下介绍外部设备对 C24 进行用户登录帧的登录、删除、读取时字符部分的内容。

(a) 帧编号

是用于表示进行登录、删除、读取的用户登录帧的编号的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位开始进行发送。

(示例)

用户登录帧的编号为 3E8H(用户登录的帧)的情况下变为“03E8”, 从“0”开始按顺序进行发送。

- 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。
- 3) 帧编号的指定值及指定内容如下所示, 不能进行除此以外的指定。

指定值	指定内容	登录目标
1H~3E7H	默认登录帧	C24 的操作系统用 ROM(只能读取)
3E8H~4AFH	用户登录帧	C24 的快闪卡(可以读取、写入, 删除)
8001H~801FH	用户登录帧	C24 的缓冲存储器(地址 1B00H~1FF6H) (可以读取、写入, 删除)

(b) 登录数据字节数

是用于表示登录数据排列的字节数的数据。

用户登录帧中包含有可更改数据(用于将用户登录帧的一部分替换为和校验代码等的控制数据)时, 与下述帧字节数有所不同。

关于登录时的数据的排列及字节数、通信时的帧的字节数的有关内容, 请参阅用户手册(应用篇)。

- 1) 通过 ASCII 代码进行数据通信时
将 0H(删除时的指定值)或 1H~50H(1~80)转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位开始进行发送。

(示例)

字节数为 10 字节的情况下
变为“000A”, 从“0”开始按顺序进行发送。

- 2) 通过二进制代码进行数据通信时
使用字节数 0H(删除时的指定值)或表示 1H~50H(1~80)的 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。

(c) 帧字节数

是用于表示登录/读取的帧的字节数的数据。

对于可更改的数据部分, 将 FFH+□H 的 2 字节作为 1 字节计算。

- 1) 通过 ASCII 代码进行数据通信时
将 0H(删除时的指定值)或 1H~50H(1~80)转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用字节数 0H(删除时的指定值)或表示 1H~50H(1~80)的 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。

(d) 登录数据

是用于表示 C24 中登录的(或已登录的)帧的数据的排列的数据，是上述(b)中所示的读取/写入字节数的数据(最多 80 字节)的排列。

删除已登录的用户登录帧时，不需要进行本登录数据的指定。

1) 通过 ASCII 代码进行数据通信时

将构成帧的数据的代码分别转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。

(示例)

指定用于对 ENQ+模块站号+空格进行发送/接收的帧的情况下
变为“05FF0120”，从“0”（起始数据）开始按顺序进行发送。

2) 通过二进制代码进行数据通信时

将构成帧的数据的代码从起始部分开始进行发送。

(示例)

指定用于对 ENQ+模块站号+空格进行发送/接收的帧的情况下
变为 05H、FFH、01H、20H，从 05H 开始按顺序进行发送。

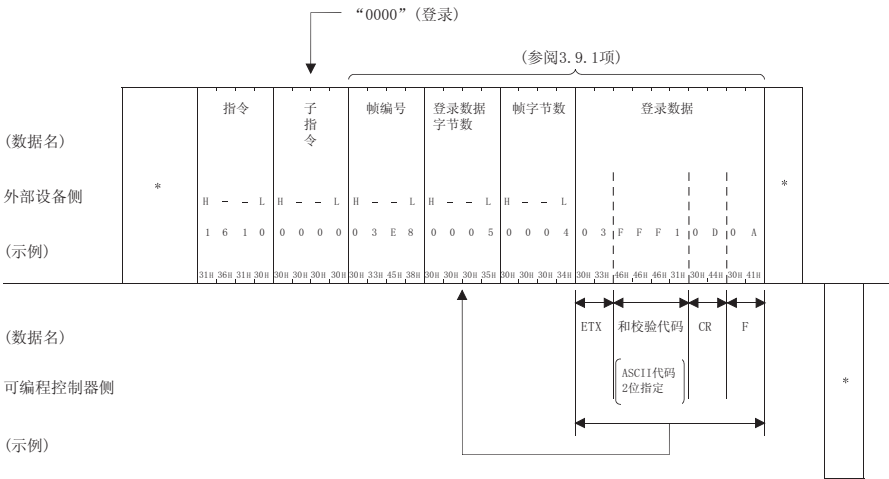
3.9.2 用户登录帧的登录、删除(指令：1610)

以下对将用户登录帧登录到 C24 的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

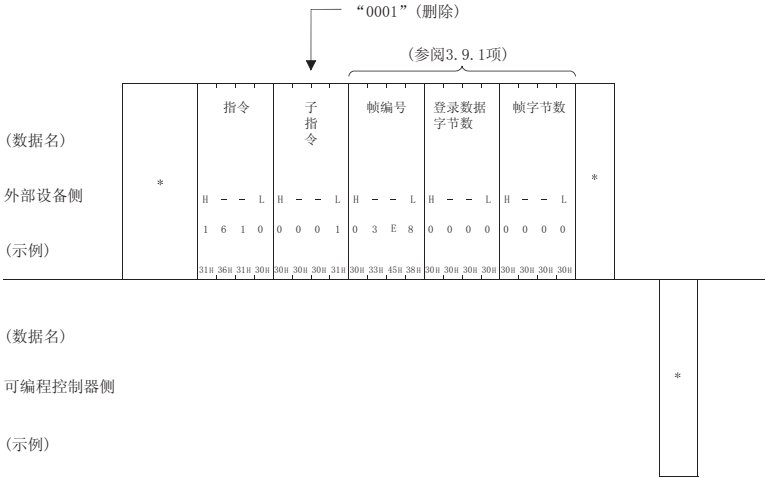
[控制步骤]

(1) 通过由 ASCII 代码进行的通信，进行登录、删除的情况下

- (a) 将用于对 ETX+和校验代码+CR+LF 进行发送/接收的帧以帧编号 3E8H 进行登录时(登录后的代码及排列：03H、FFH、F1H、0DH、0AH)
和校验代码指定为将除去起始帧部分的数据作为二进制数据进行加法运算的结果的低位 1 字节(8 位)的数值转换为 ASCII 代码 2 位(16 进制数)的字符串时。

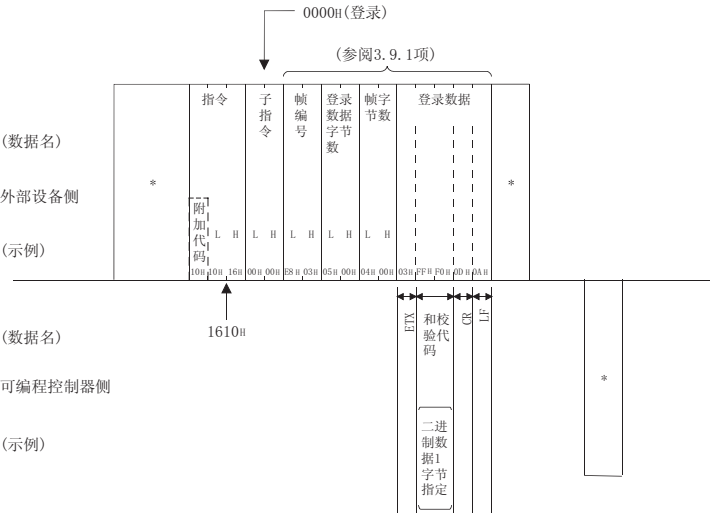


(b) 对以帧编号 3E8H 登录的用户登录帧进行删除的情况下

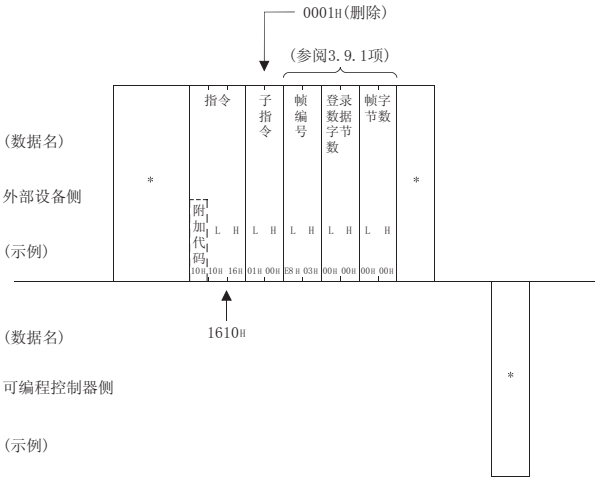


(2) 通过由二进制代码进行的通信，进行登录、删除的情况下

- (a) 将用于对 ETX+和校验代码+CR+LF 进行发送/接收的帧以帧编号 3E8H 进行登录时(登录后的代码及排列：03H、FFH、F0H、0DH、0AH)
和校验代码指定为原样不变地使用将除去起始帧部分的数据作为二进制数据进行加法运算的结果的低位 1 字节(8 位)的数值时。



- (b) 对以帧编号 3E8H 登录的用户登录帧进行删除的情况下



要点	
(1)	指定同一帧编号进行用户登录帧的重新登录时，应在进行了用户登录帧的删除之后再行重新登录。 指定已登录的帧编号进行用户登录帧的重新登录时，将变为出错状态，并返回异常响应。
(2)	指定未进行用户登录帧的登录的帧编号进行删除时，将变为出错状态并返回异常响应。
(3)	对于帧编号，应在以下范围内进行指定。 • 登录至 C24 的快闪卡中时 : 3E8H ≤ 帧编号 ≤ 4AFH • 登录至 C24 的缓冲存储器时 : 8001H ≤ 帧编号 ≤ 801FH

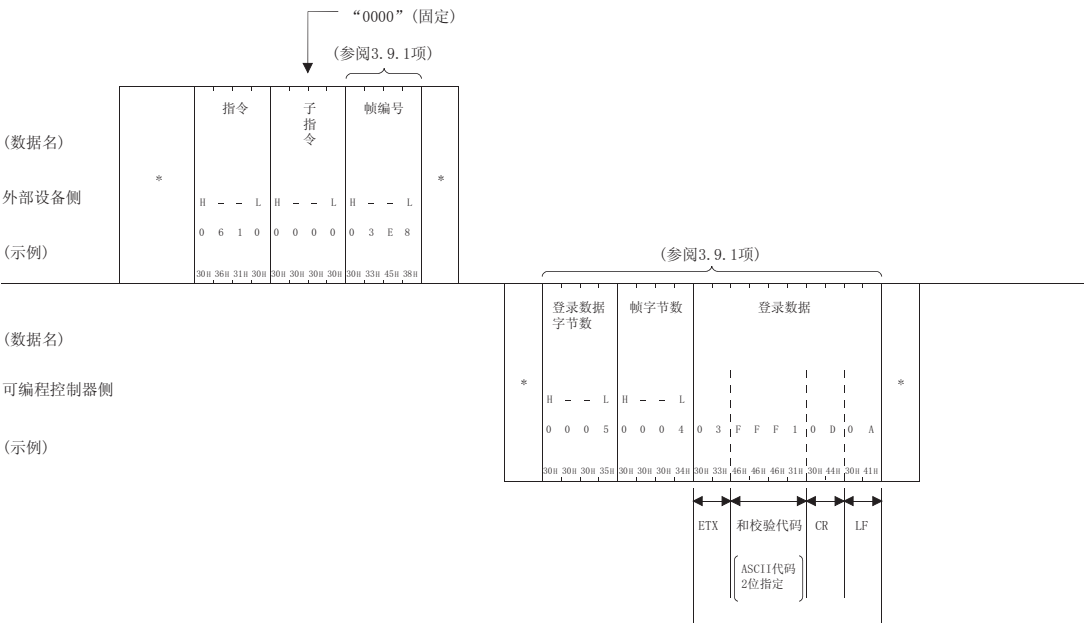
3.9.3 用户登录帧的读取(指令：0610)

以下对从 C24 中读取用户登录帧的登录内容(登录数据的排列)的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

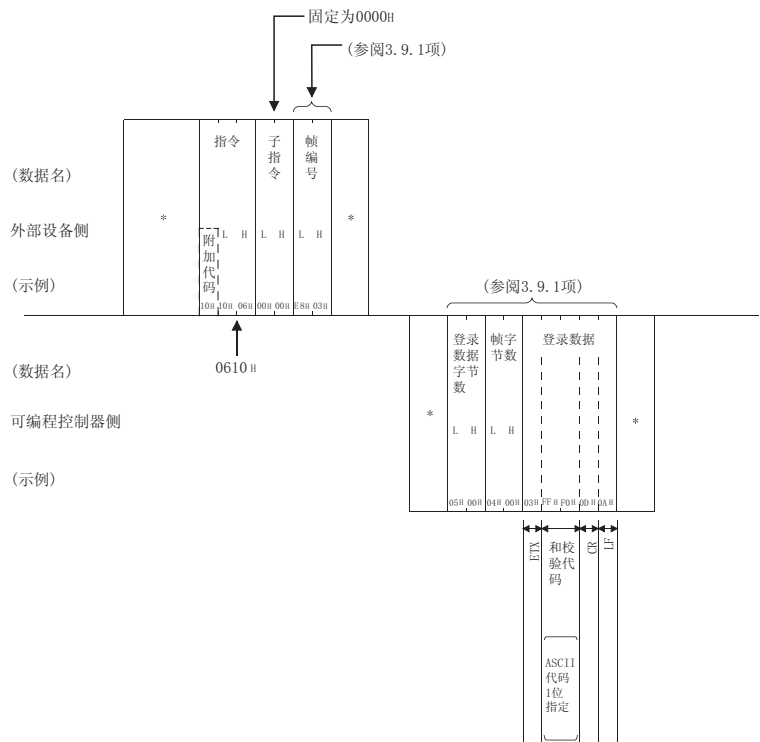
(1) 通过由 ASCII 代码进行的通信，对以帧编号 3E8H 登录的内容进行读取的情况下

登录内容是用于对 ETX+和校验代码+CR+LF 进行发送/接收的帧。
和校验代码指定为，将除去起始帧部分的数据作为二进制数据进行加法运算的结果的低位 1 字节(8 位)的数值转换为 ASCII 代码 2 位(16 进制数)的字符串时。



(2) 通过由二进制代码进行的通信，对以帧编号 3E8H 登录的内容进行读取的情况下

登录内容是用于对 ETX+和校验代码+CR+LF 进行发送/接收的帧。
和校验代码指定为，原样不变地使用将除去起始帧部分的数据作为二进制数据进行加法运算的结果的低位 1 字节 (8 位) 的数值时。



要点
(1) 指定未进行用户登录帧登录的帧编号时将变为出错状态，并返回异常响应。
(2) 对于帧编号，应在以下范围内进行指定。 • C24 的操作系统用 ROM 中登录的默认登录帧 1H ≤ 帧编号 ≤ 3E7H • C24 的快闪卡中登录的用户登录帧 3E8H ≤ 帧编号 ≤ 4AFH • C24 的缓冲存储器中登录的用户登录帧 8001H ≤ 帧编号 ≤ 801FH

3.10 全局功能：用于串行通信模块

全局功能是指，对与外部设备多点连接的 C24 安装站可编程控制器 CPU 的全局信号 (X1A/X1B) 进行 ON/OFF 的功能。
该功能用于对可编程控制器 CPU 的紧急指令、同时启动、数据发送接收可否的互锁信号等。
以下对使用全局功能时的控制步骤进行举例说明。

要点
(1) 外部设备对多点连接站上的计算机链接模块使用了 A 兼容 1C 帧的 GW 指令的情况下，对于 Q/L/QnACPU 站的串行通信，接收了 GW 指令的接口侧的全局信号 (输入信号) X1A 或 X1B 将变为 ON/OFF。 (示例) 从 C24 的 CH1 侧接口接收了 GW 指令时，C24 对 X1A 进行 ON/OFF。
(2) 对多点连接的 ACPU+C24 站使用了全局功能时，ACPU 的 X2 将变为 ON/OFF。
(3) 将全局信号置为 ON/OFF 后对可编程控制器 CPU 进行了重新启动时，全局信号将变为 OFF。
(4) 只有对于与外部设备相连接的 Q/L/QnACPU+C24 站 (多点连接站也包含在内)，才可以使用本功能。 对于经由网络系统的其它站 ACPU+C24 站，不能使用本功能。

3.10.1 指令及字符部分的内容

以下对外部设备将 C24 的全局信号置为 ON/OFF 时的指令以及控制步骤内的字符部分 (通过二进制代码进行通信时为数据部分) 的有关内容进行说明。

(1) 指令

功能		指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
				STOP 中	RUN 中		
					写入允许 设置	写入禁止 设置	
全局	全局信号 OFF	1618 (0000)	将全局信号置为 OFF。	○	○	○	3. 10. 2 项
	全局信号 ON	1618 (0001)	将全局信号置为 ON。				

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

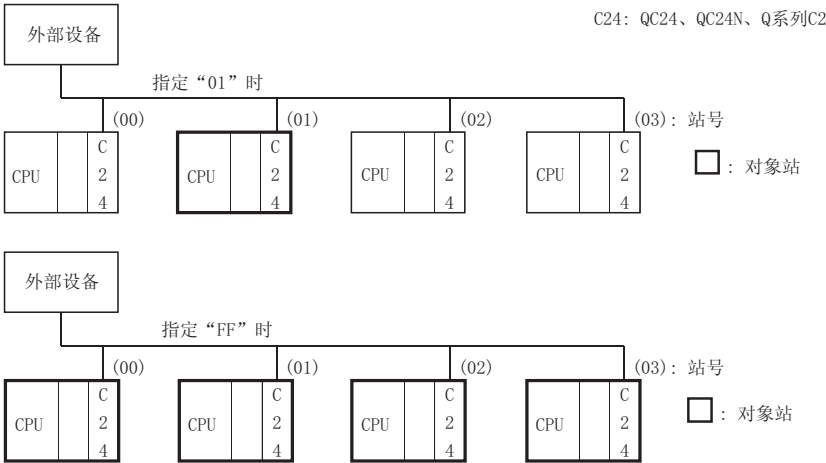
以下对外部设备将全局信号置为 ON/OFF 时的字符部分的全局信号指定数据的内容进行说明。

(a) 站号

对于外部设备将全局信号置为 ON/OFF 的站，以站号进行指定。
应指定以下之一。

指定值		全局功能的处理内容
ASCII 代码时	二进制代码时	
“00” ~ “1F”	00 _h ~ 1F _h	只有对多点连接中的指定站，才可以对全局信号进行 ON/OFF。
“FF”	FF _h	对多点连接的所有站的全局信号进行 ON/OFF。

(示例)



(b) 全局信号指定

是用于指定将至 Q/L/QnACPU 的哪个全局信号置为 ON/OFF 的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
- 3) 全局信号指定的指定值及指定内容如下所示，不能进行除此以外的指定。

指定值	指定内容
0000 _h	将接收了全局功能用的指令的接口侧的全局信号置为 ON/OFF。 • 从 CH1 侧接口接收了指令时将 X1A 置为 ON/OFF。 • 从 CH2 侧接口接收了指令时将 X1B 置为 ON/OFF。
0001 _h	与接收了全局功能用的指令的接口无关，将 X1A 置为 ON/OFF。
0002 _h	与接收了全局功能用的指令的接口无关，将 X1B 置为 ON/OFF。

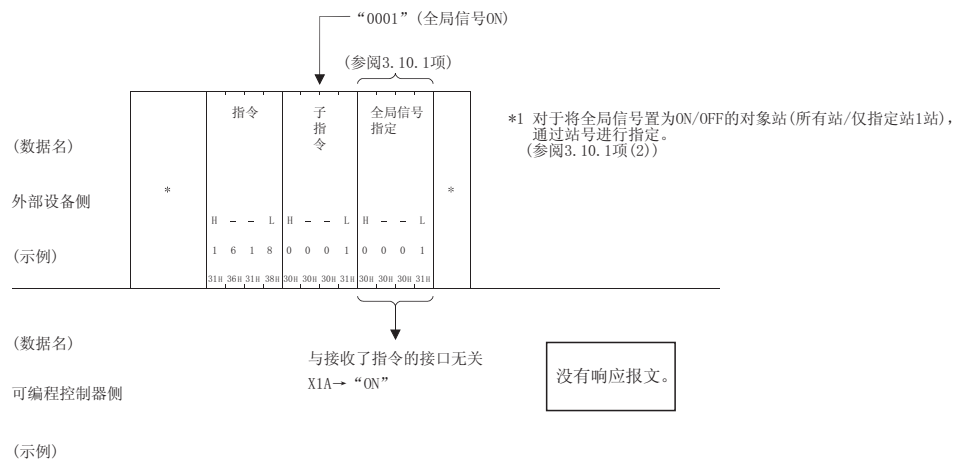
3.10.2 全局功能的控制步骤(指令：1618)

以下对将至可编程控制器 CPU 的全局信号通过外部设备置为 ON/OFF 的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

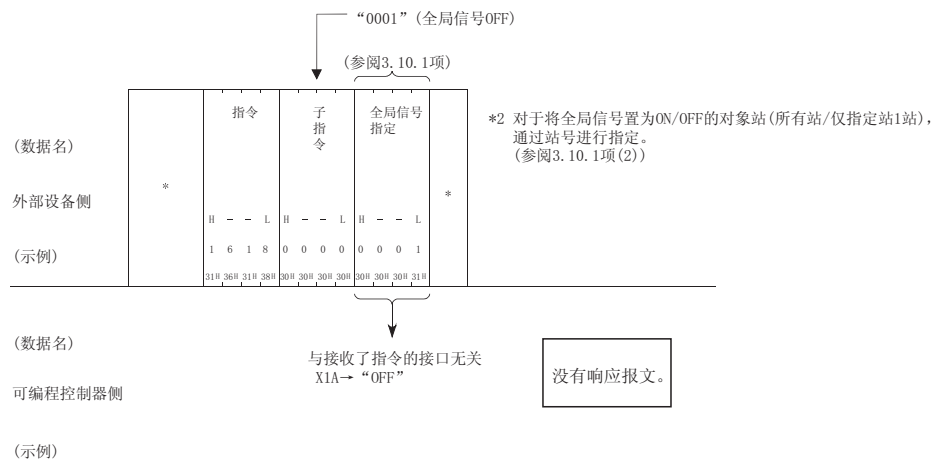
[控制步骤]

(1) 通过 ASCII 代码的格式 1，将全局信号置为 ON/OFF 的情况下

(a) 将多点连接的所有站的全局信号 X1A 置为 ON 的情况下

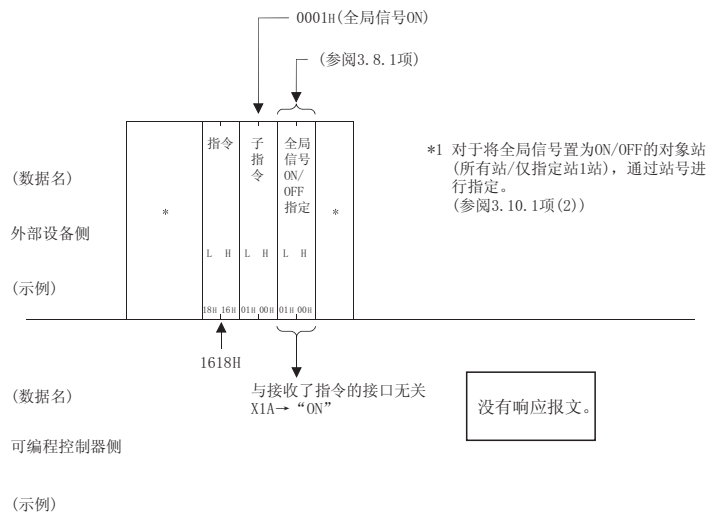


(b) 将多点连接的所有站的全局信号 X1A 置为 OFF 的情况下

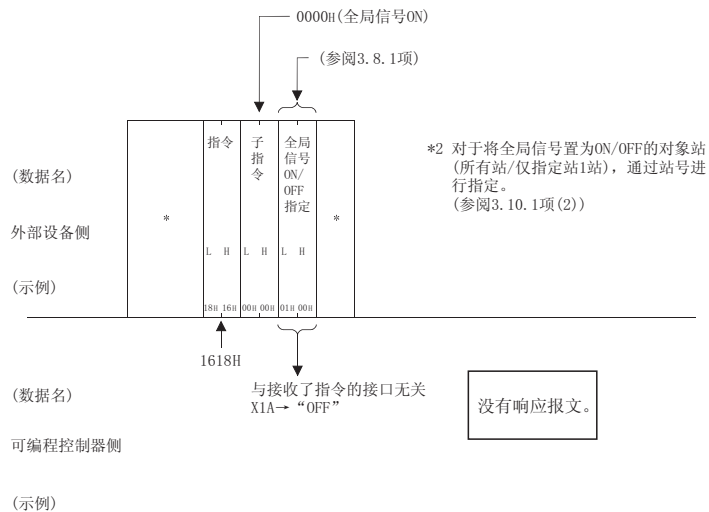


(2) 通过二进制代码的格式 5，将全局信号置为 ON/OFF 的情况下

(a) 将多点连接的所有站的全局信号 X1A 置为 ON 的情况下



(b) 将多点连接的所有站的全局信号 X1A 置为 OFF 的情况下



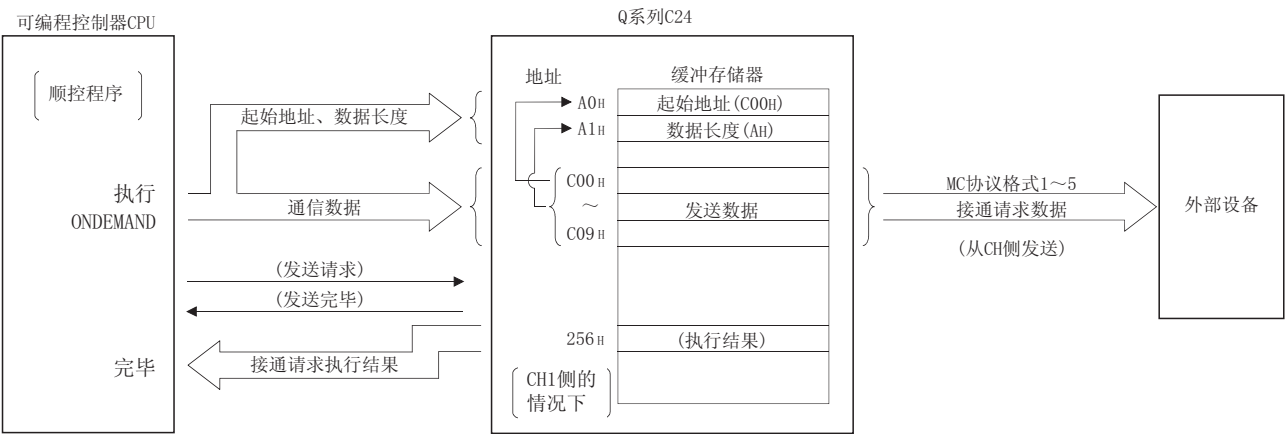
3.11 至外部设备的数据发送(接通请求功能)：用于串行通信模块

通过由 MC 协议进行的数据通信，从可编程控制器 CPU 向外部设备发出数据发送请求的功能称为接通请求功能。

通过 MC 协议进行数据通信时，根据来自于正常通信的外部设备的指令进行通信。

接通请求功能可用于以下情况：将必须通知至外部设备的紧急数据等作为接通请求数据，通过从可编程控制器 CPU 的启动发送至外部设备。

(从 CH1 侧接口进行发送的情况下)



要点

该接通请求功能可在外部设备与可编程控制器 CPU 的系统配置为 1：1 的情况下使用。

(外部设备为1个的情况下)

外部设备

可编程
控制器CPU

CH1或者
CH2
Q系列C24

(外部设备为2个的情况下)…1:1连接的组合

外部设备

外部设备

可编程
控制器CPU

CH1
CH2
Q系列C24

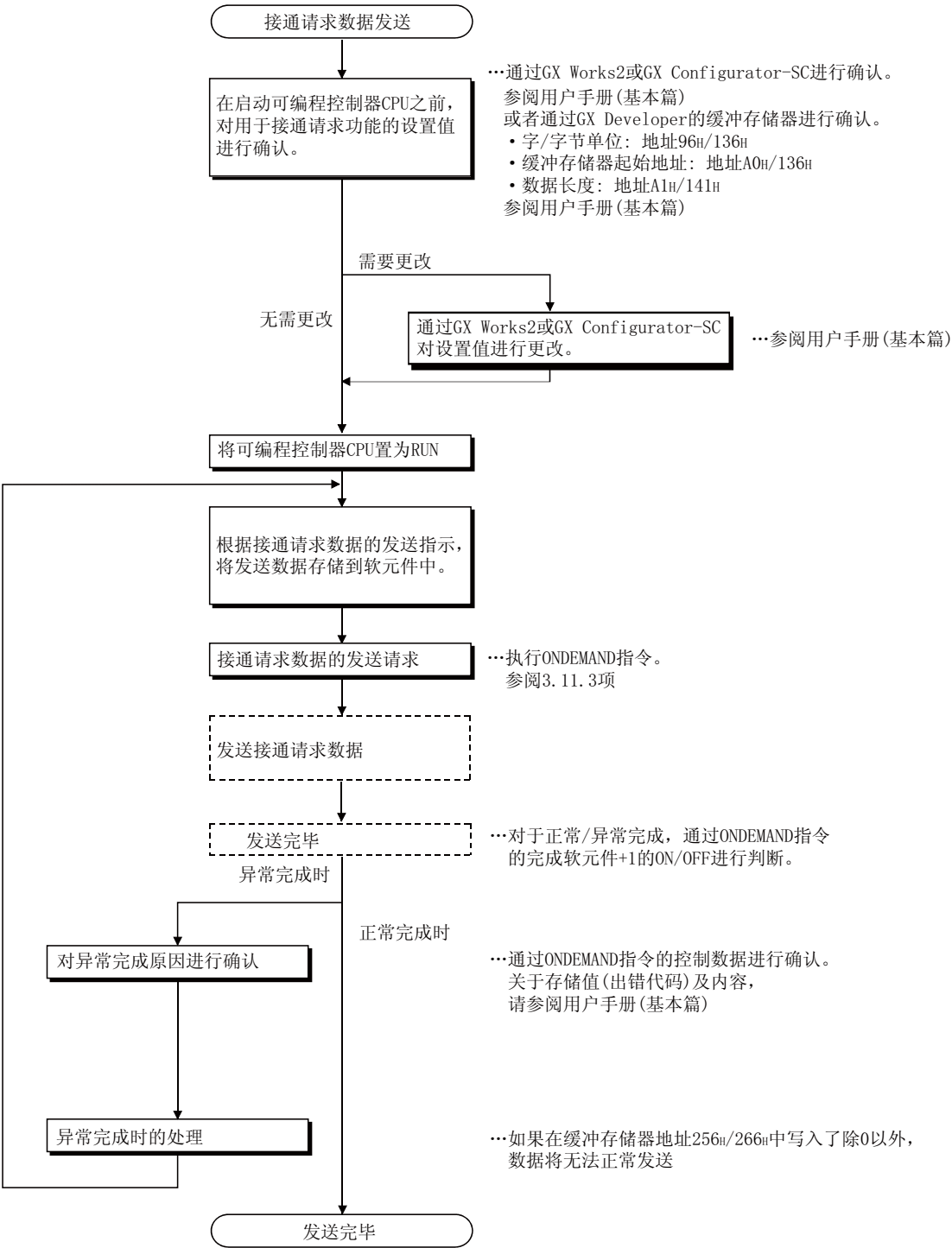
(独立动作)

系统配置为 1：1 以外时，不要使用接通请求功能。

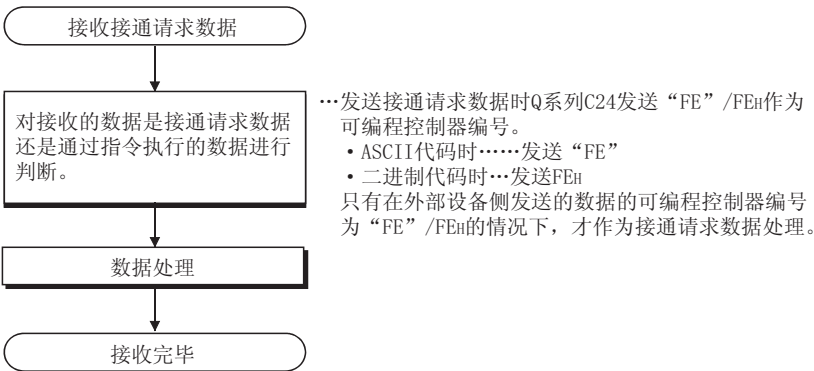
将外部设备与可编程控制器 CPU 以 1：n 站或 m：n 站进行多点连接时如果使用接通请求功能，控制步骤格式 1～5 的通信数据、接通请求数据将损坏，导致无法正常进行数据发送。

3.11.1 接通请求功能的执行步骤

(1) 可编程控制器 CPU 侧的执行步骤

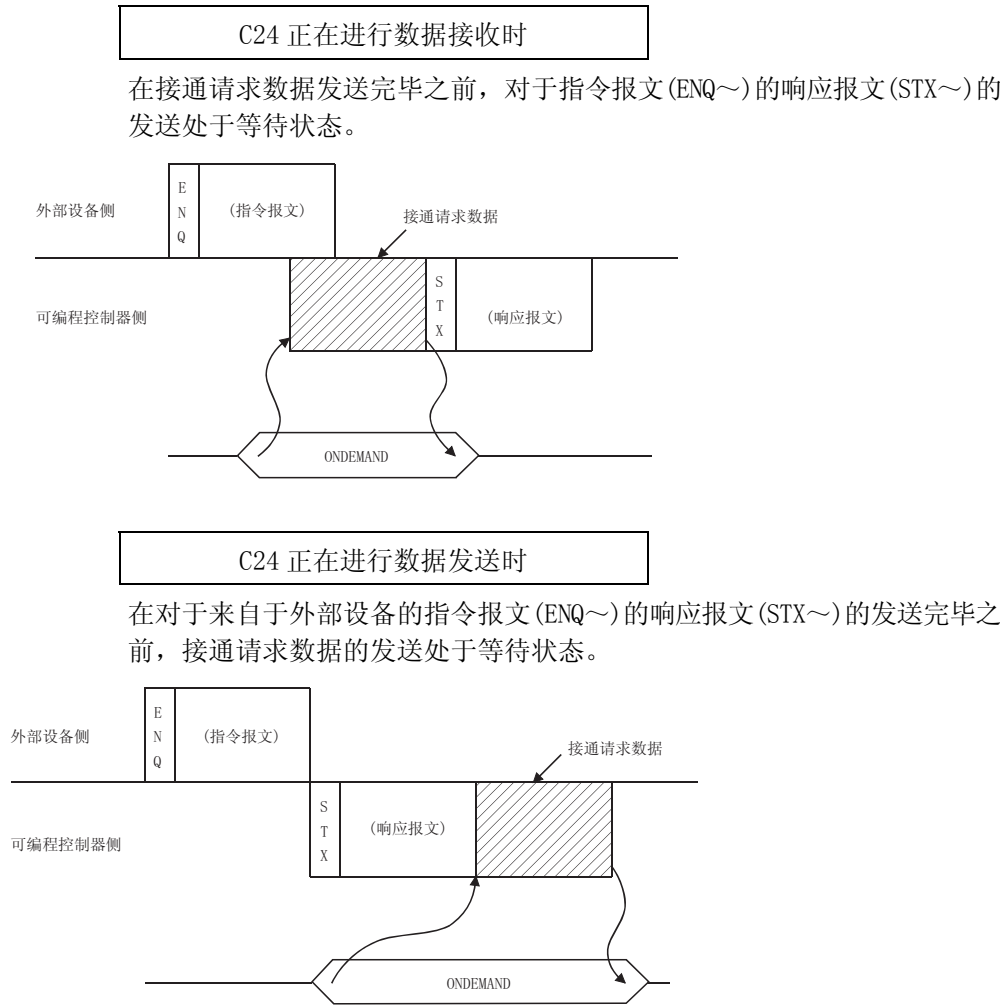


(2) 外部设备侧的执行步骤



(3) 以下对发出接通请求数据的发送请求时的时序图进行说明。

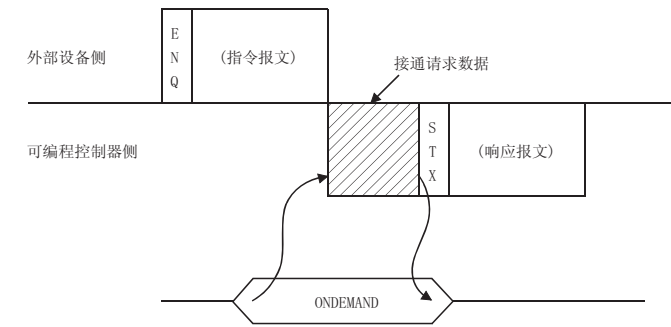
(a) 全双工通信方式的情况下



(b) 双工通信方式的情况下：参阅用户手册(应用篇)

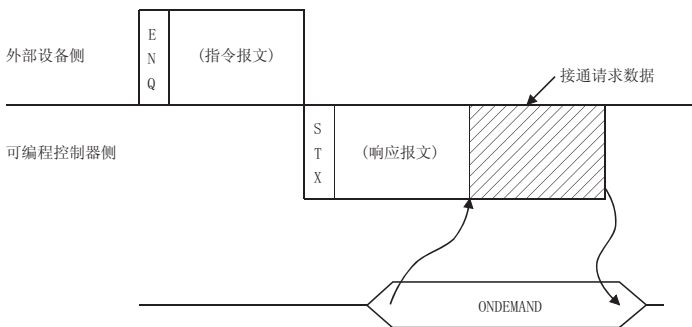
C24 正在进行数据接收时

在来自于外部设备的指令报文(ENQ~)的接收完毕之前，接通请求数据的发送处于等待状态。



C24 正在进行数据发送时

在对于来自于外部设备的指令报文(ENQ~)的响应报文(STX~)的发送完毕之前，接通请求数据的发送处于等待状态。



备注

进行接通请求数据以及响应数据的发送时，分别根据用户手册(应用篇)中所示的发送监视时间(定时器 2)进行超时检查。
发生了该超时出错时，应重新设置发送监视时间，使得在监视时间内能正常完成发送。

3.11.2 通过接通请求功能进行数据发送的格式

对于接通请求功能的接通请求数据的排列，根据通过 GX Works2 或 GX Developer 设置的 C24 的通信协议的格式编号，以如下所示的帧的设置格式的排列进行发送。

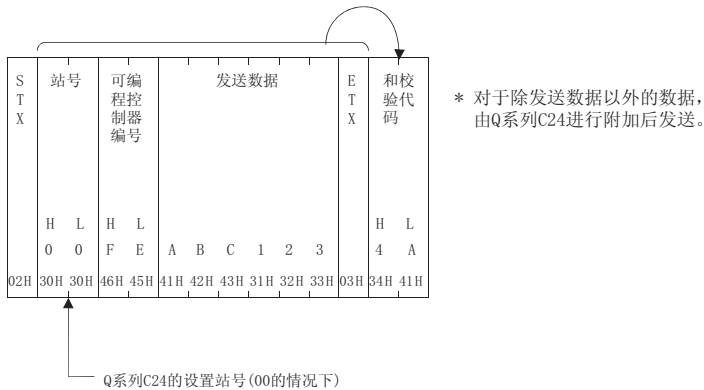
- 设置格式为“1”～“4”时：A 兼容 1C 帧的“格式 1”～“格式 4”的排列。
- 设置格式为“5”时：QnA 兼容 4C 帧的“格式 5”的排列。

以下对发送的接通请求数据的排列及内容进行举例说明。

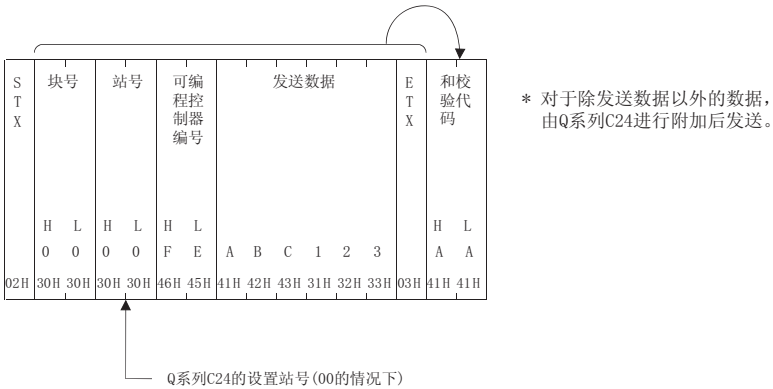
对于接通请求数据中的发送数据、数据字节数、和校验代码以外的各数据项目，按图中所示以 ASCII 代码/二进制代码数据进行发送。

(站号为表示 C24 的设置站号 (00H～1FH) 的数据。)

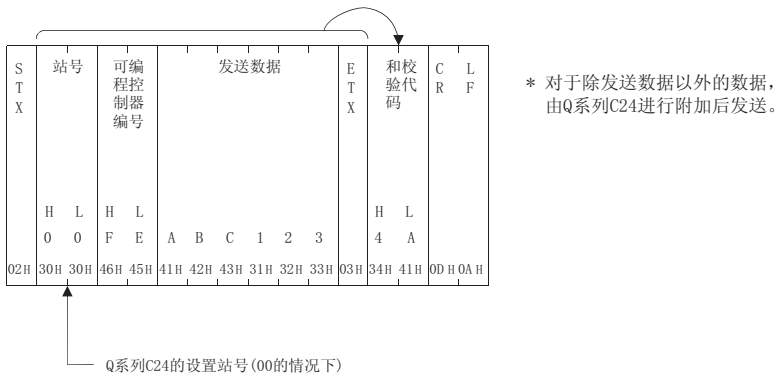
(1) 通过 A 兼容 1C 帧格式 1、格式 3 的接通请求数据发送格式



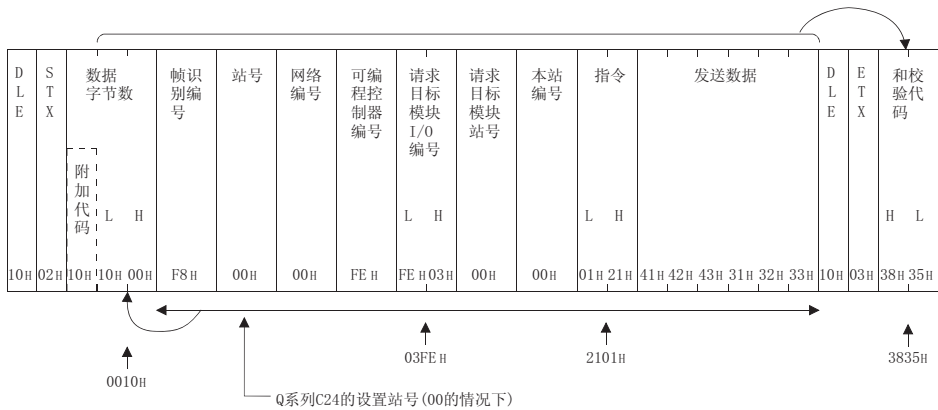
(2) 通过 A 兼容 1C 帧格式 2 的接通请求数据发送格式



(3) 通过 A 兼容 1C 帧格式 4 的接通请求数据发送格式



(4) 通过 QnA 兼容 4C 帧格式 5 的接通请求数据发送格式



要点
希望通过除上述以外的用户选择的 QnA 兼容 3C/4C 帧的格式等的排列进行发送时，应使用用户手册(应用篇)中所示的通过用户登录帧进行的数据通信功能。

3.11.3 接通请求功能的控制步骤(指令: 2101)

以下对通过接通请求功能向外部设备发送接通请求数据时的控制步骤进行举例说明。控制步骤图中所示的报文的数据项目的排列·内容根据通信时的帧·格式而有所不同。通过 GX Works2 或 GX Developer 进行的通信协议设置为“1~4”时，将以 A 兼容 1C 帧发送接通请求数据。关于 A 兼容 1C 帧的各格式的数据项目的排列·内容，请参阅 5.1 节。

通过 GX Works2 或 GX Developer 进行的通信协议设置为“5”时，将以 QnA 兼容 4C 帧的格式 5 发送接通请求数据。

关于 QnA 兼容 4C 帧的数据项目的排列·内容, 请参阅 3.1 节。

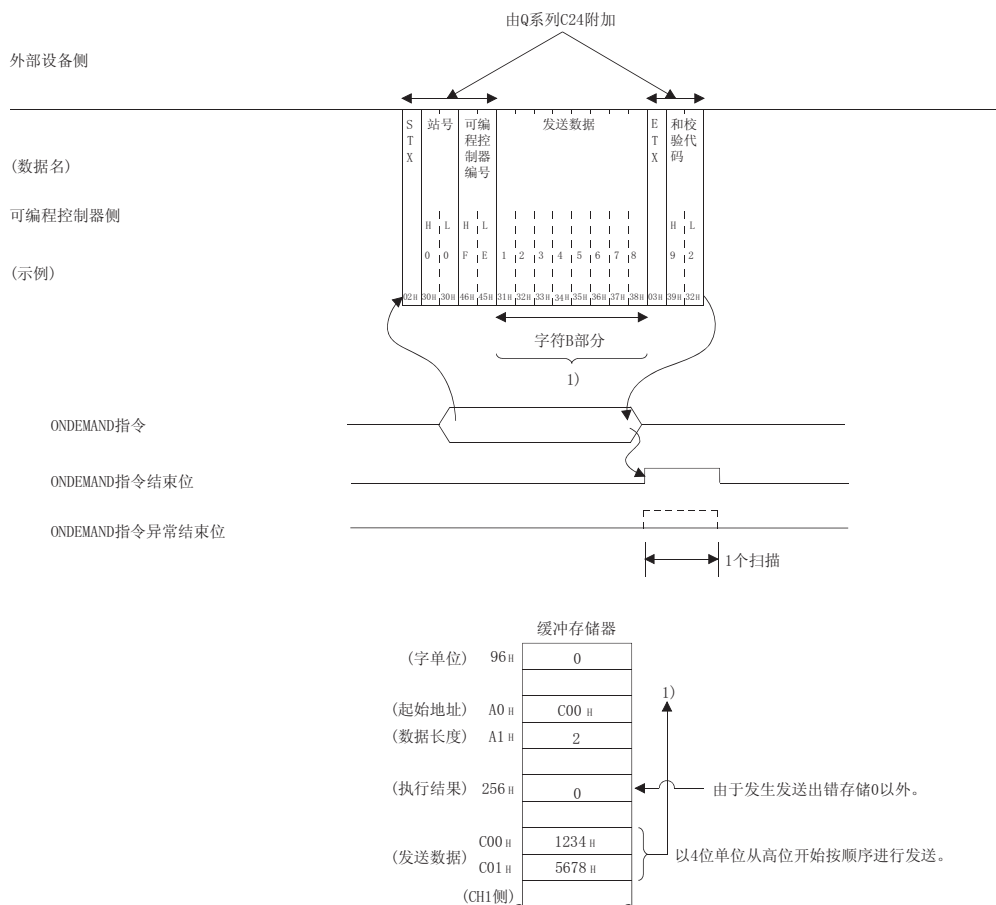
[控制步骤]

以 A 兼容 1C 帧的格式 1、QnA 兼容 4C 帧的格式 5 进行通信的情况如下所示。

(1) 格式 1 的情况下

(a) GX Works2 或 GX Configurator-SC 的“字/字节单位指定”为“0”（字单位）的情况下

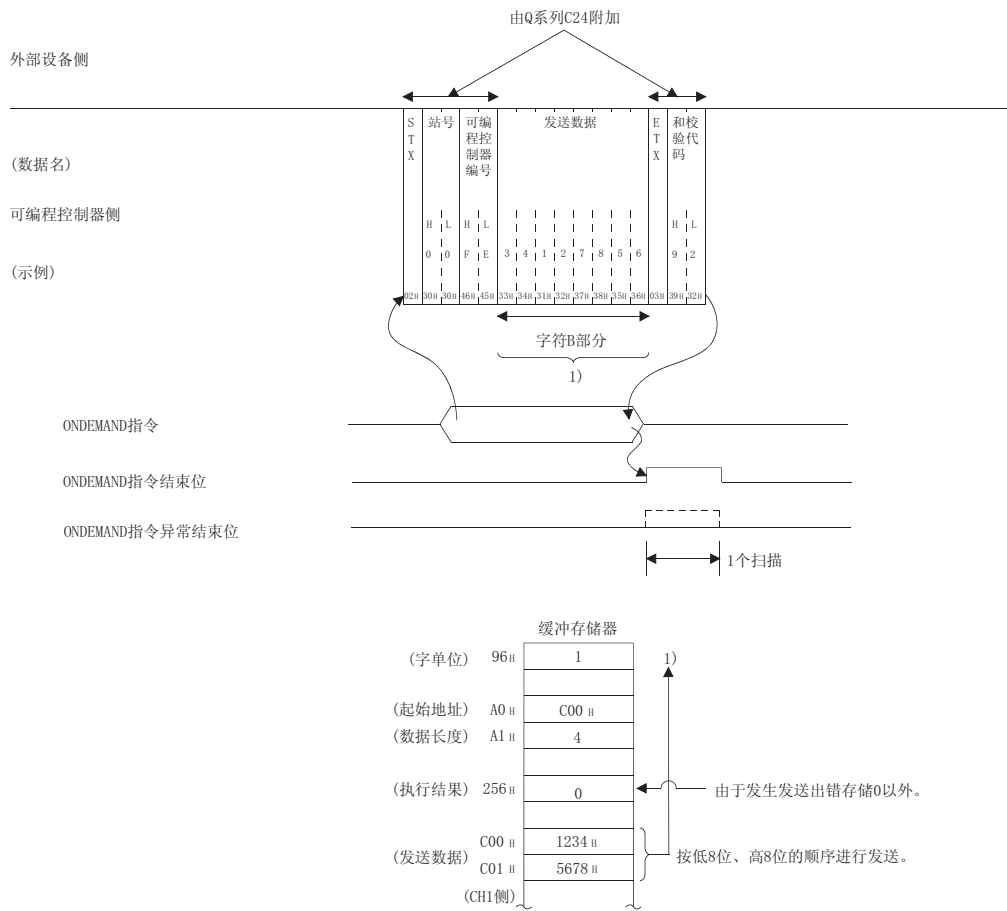
下图为指定了 2 字的发送数据时的情况。



要点

- (1) 以控制步骤格式 2 进行的情况下, 块编号将变为“00”。
- (2) 接通请求数据的发送数据部分的字符数为: 数据长度 $\times 4$ 。
1 个字节数据使用 4 个字符。
因此, 以 4 位(16 进制)表示 1 个字节数据。

- (b) GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）的情况下
下图为指定了 2 字(4 字节)的发送数据时的情况。



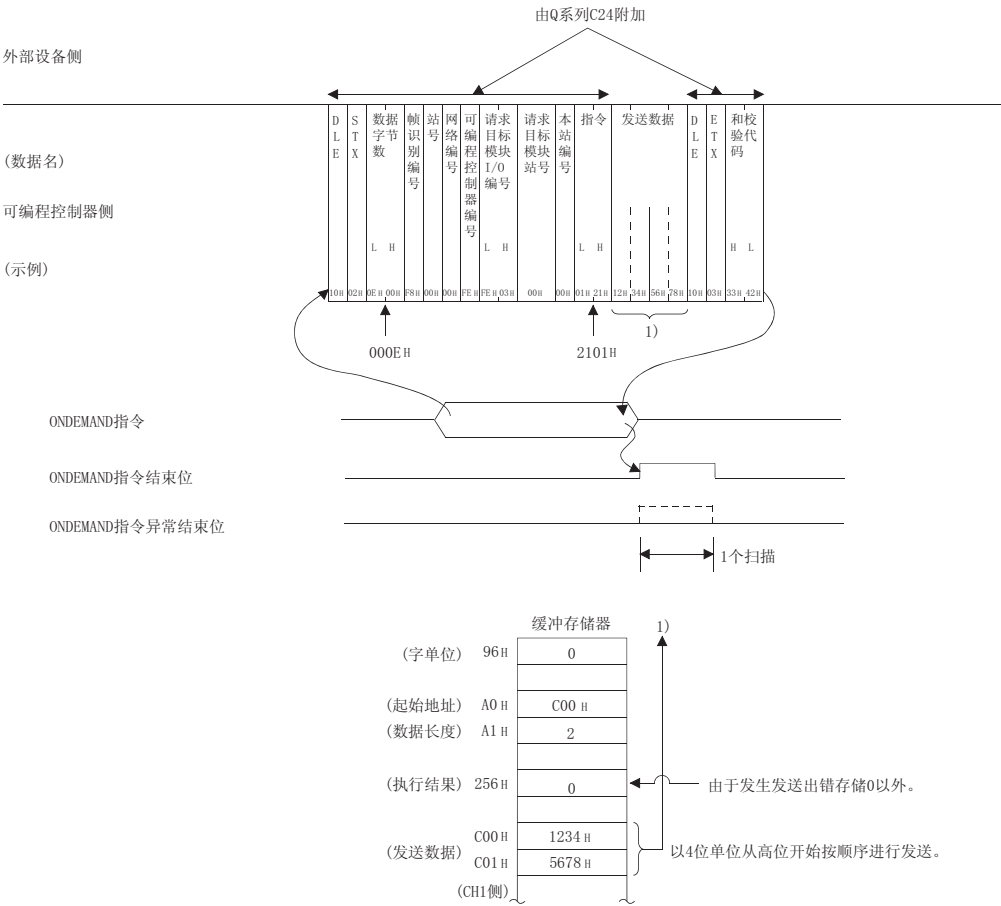
要点

- (1) 以控制步骤格式 2 进行的情况下，块编号将变为“00”。
- (2) 接通请求数据的发送数据部分的字符数为：数据长度×2。
1 个字节数据使用 2 个字符。
因此，以 2 位(16 进制)表示 1 个字节数据。
- (3) 数据长度为奇数时，最终数据存储软元件的低位字节(位 0~位 7)的数据将被发送。

(2) 格式 5 的情况下

(a) GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”（字单位）的情况下

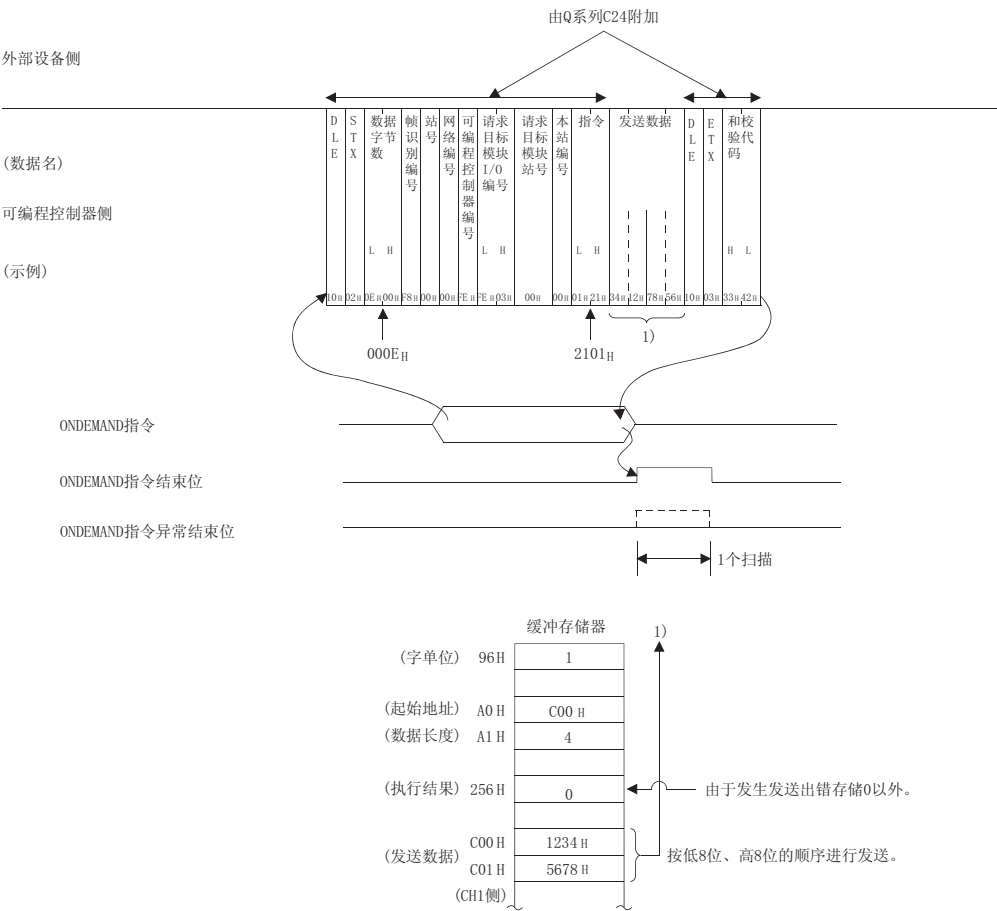
下图为指定了 2 字的发送数据时的情况。



要点

接通请求数据的发送数据部分的字节数为：数据长度×2。
(1 个字数数据使用 2 个字节。)

- (b) GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）的情况下
下图为指定了 2 字(4 字节)的发送数据时的情况。



要点

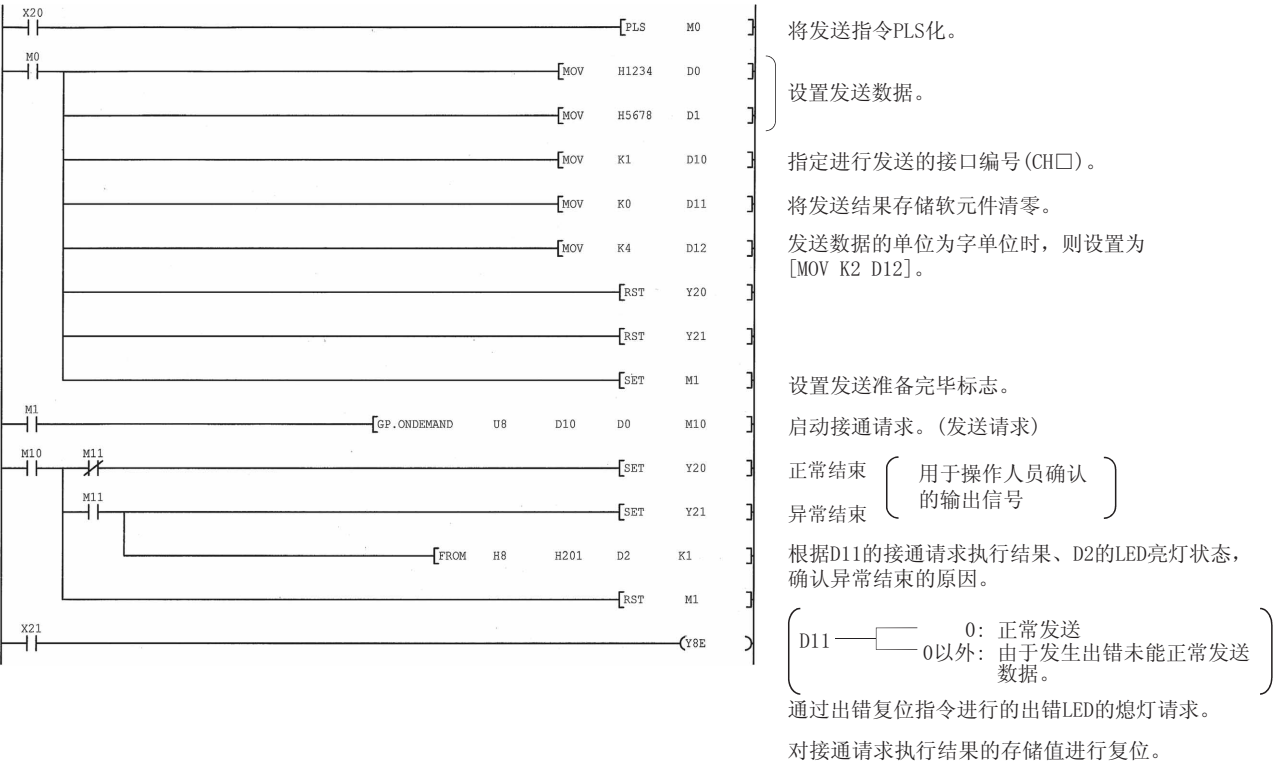
- (1) 接通请求数据的发送数据部分的字节数与数据长度相同。
(1 个字节数据使用 1 个字节。)
- (2) 数据长度为奇数时，最终数据存储软元件的低位字节(位 0～位 7)的数据将被发送。

(3) 使用接通请求功能时的顺控程序示例(使用 ONDEMAND 指令)

作为本项(1)(2)中所示的控制的示例，使用接通请求功能时的顺控程序示例如下所示。

(C24 的输入输出信号为 X/Y80~X/Y9F，从 CH1 侧发送)

关于 ONDEMAND 指令的详细内容，请参阅用户手册(基本篇)。



要点

- (1) 由专用指令进行的通信状态的读取可通过 SPBUSY 指令进行。
- (2) 关于上述(1)的详细内容，请参阅用户手册(基本篇)。
- (3) 对发送数据的存储容量(存储在上述程序示例中的 D0~D1 中)、数据长度(存储在上述程序示例中的 D12 中)进行指定时，不要超出用户分配给接通请求功能用的缓冲存储器范围。

3.12 传送顺控程序的初始化：用于串行通信模块

该功能是指，用于对通过 MC 协议的 QnA 兼容 4C 帧的格式 5 进行的数据通信的传送顺控程序进行初始化，将 C24 置为等待接收来自于外部设备的指令的状态的功能。

通过格式 5 进行数据通信的情况下，在外部设备侧对 C24 的传送顺控程序进行初始化时，应将本节中所示的指令报文发送至 C24。

本节的功能与通过 ASCII 代码进行数据通信时的 EOT、CL 的功能相同。

关于以下内容，请参阅 3.1.6 项 (1) (b) 的 EOT、CL 的说明部分。

- 何时外部设备对 C24 的传送顺控程序进行初始化
- C24 进行传送顺控程序初始化时的处理/动作
- 通过 ASCII 代码进行数据通信时的传送顺控程序的初始化方法

要点
只能对与外部设备相连接的 C24(多点连接站也包含在内)使用本功能。 对于经由网络系统的其它站 C24，不能使用本功能。

3.12.1 指令

以下对外部设备对 C24 的传送顺控程序进行初始化时的指令有关内容进行说明。

功能	指令(子指令)	处理内容	可编程控制器 CPU 的状态		
			STOP 中	RUN 中	
				写入允许设置	写入禁止设置
传送顺控程序初始化	1615 (0000)	中止当前的处理请求，置为指令接收等待。	○	○	○

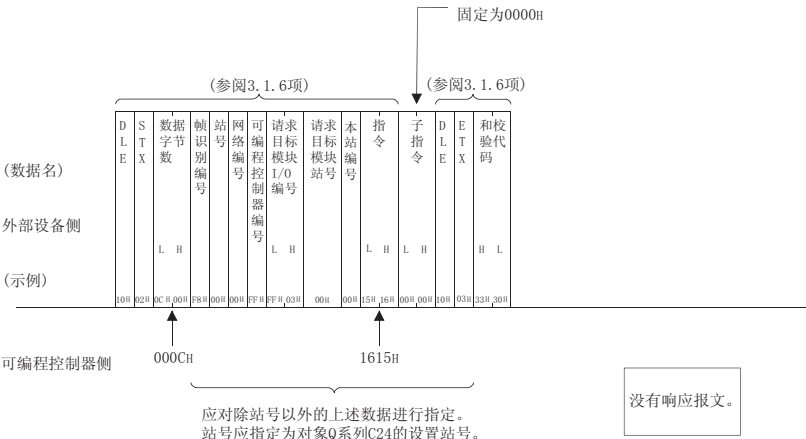
上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

3.12.2 传送顺控程序初始化(指令：1615)

以下对传送顺控程序初始化的控制步骤进行举例说明。

[控制步骤]

以 QnA 兼容 4C 帧的格式 5 进行通信时的情况如下所示。



3.13 模式的切换：用于串行通信模块

该功能是启动 C24 后，从外部设备对指定接口当前的通信协议(动作模式)及传送规格进行强制切换的功能。

通过使用本功能，可以在对指定接口的通信协议及传送规格进行了更改时无需重启 Q/LCPU，继续进行数据通信。

在本节中，仅对用于从外部设备进行 C24 的模式切换的指令的使用方法进行说明。

要点
(1) 进行模式切换时，请参阅用户手册(应用篇)。对通过可编程控制器 CPU 对 C24 的模式进行切换的方法也进行了说明。
(2) 只能对与外部设备相连接的 C24(多点连接站也包含在内)使用从外部设备进行模式切换的功能。 对于经由网络系统的其它站 C24，不能使用该功能。
(3) 对于 C24 的模式切换，一旦发出请求便立即开始执行。 在有模式切换请求时，如果 C24 正在进行某个处理，该处理将被中止。

3.13.1 指令及字符部分的内容

以下对外部设备对 C24 的模式进行切换时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)的有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
			访问站-1(参阅 3.2 节*7)	STOP 中	RUN 中		
					写入允许设置	写入禁止设置	
模式切换	1612 (0000)	对指定接口的动作模式、传送规格进行切换。	(1 站)	○	○	○	3.13.2 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下对外部设备对 C24 的模式进行切换时的字符部分的内容进行说明。
关于各设置值的内容、限制，请参阅用户手册(基本篇)。

(a) 通道编号

是用于表示模式切换的对象接口(CH□)的数据。

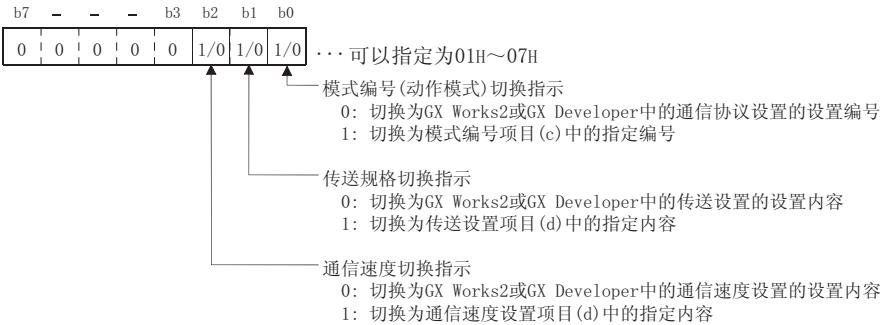
- 1) 通过 ASCII 代码进行数据通信时
将 3) 中所示的数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位 (“0”) 开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用 3) 中所示的 1 字节的数值进行发送。
- 3) 通道编号的指定值及指定内容如下所示。

指定值	对象接口
1 _H	C24 的 CH1 侧接口
2 _H	C24 的 CH2 侧接口

(b) 切换指示

是用于表示模式切换的切换内容的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 切换指示的指定值及指定内容如下所示。
所有的位均为 0(OFF)时，通过 GX Works2 或 GX Developer 中的各开关设置内容进行模式切换。



(c) 模式编号

是模式切换时用于指定切换后的通信协议设置的数据。

- 1) 通过 ASCII 代码进行数据通信时
通过 ASCII 代码 2 位(16 进制数)进行下述指定，从高位开始进行发送。
(示例：01 的情况下，从“0”开始进行发送)

指定值		模式编号(动作模式)
ASCII	二进制	
01	01 _H	MC 协议(格式 1)
02	02 _H	MC 协议(格式 2)
03	03 _H	MC 协议(格式 3)
04	04 _H	MC 协议(格式 4)
05	05 _H	MC 协议(格式 5)
06	06 _H	无顺序协议
07	07 _H	双向协议
09	09 _H	通信协议
FF	FF _H	MELSOFT 连接、GX Developer 连接*1

*1 在 GX Works2 或 GX Developer 的智能功能模块开关设置中，设置为 MELSOFT 连接模式或 GX Developer 连接模式的情况下，应将通信协议设置指定为“00_H”。(参阅用户手册(基本篇))

- 2) 通过二进制代码进行数据通信时
使用上述 1) 中所示的 1 字节的数值(01_H~07_H、09_H、FF_H)进行发送。
- 3) (b)中所示的切换指示的“模式编号(动作模式)切换指示”为“0”时，对本项目以虚拟键值“01~07、09、FF”(ASCII 代码的情况下)进行指定。(不要指定为“0”。)

(d) 传送设置、通信速度设置

是模式切换时用于指定切换后的传送规格的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 传送设置、通信速度设置的指定值及指定内容如下所示。

[传送设置]



*1 对于模式编号中设置为“MELSOFT 连接”或“GX Developer 连接”的接口侧，应全部设置为 OFF 侧。

与 GX Works2 或 GX Developer 直接连接的情况下，以 GX Works2 或 GX Developer 侧的设置值执行动作。（参阅下述内容）

传送设置	GX Works2 或 GX Developer 侧设置内容
动作设置	独立
数据位	8
奇偶校验位	有
偶数/奇数校验	奇数
停止位	1
和校验代码	有
运行中写入	允许
设置更改	允许/禁止

[通信速度设置] *2*3*4

通信速度(单位: bps)	位位置	通信速度(单位: bps)	位位置	备注
	b15~b8		b15~b8	
50bps	0FH	14400bps	06H	在与外部设备的数据通信中, 由于发生了溢出出错、成帧出错等导致无法正常进行数据通信时, 应降低通信速度。
300bps	00H	19200bps	07H	
600bps	01H	28800bps	08H	
1200bps	02H	38400bps	09H	
2400bps	03H	57600bps	0AH	
4800bps	04H	115200bps	0BH	
9600bps	05H	230400bps	0CH	

*2 对于 230400bps, 仅在 QJ71C24N(-R2/R4) 或 LJ71C24(-R2) 的 CH1 侧可以使用。

*3 2个接口上分别连接了外部设备时,2个接口的通信速度的合计应设置为115200bps以内(QJ71C24N(-R2/R4)的情况下为230400bps以内)。

仅某个接口与外部设备相连接时，连接接口侧最大可设置为 115200bps(QJ71C24N(-R2/R4) 或 LJ71C24(-R2) 的 CH1 侧的情况下，最大 230400bps)。在这种情况下，对于未连接外部设备的接口侧应设置为 300bps。

*4 对于模式编号中设置为“MELSOFT 连接”或“GX Developer 连接”的接口侧，应设置为“00H”。
以 GX Works2 或 GX Developer 侧的设置速度执行动作。

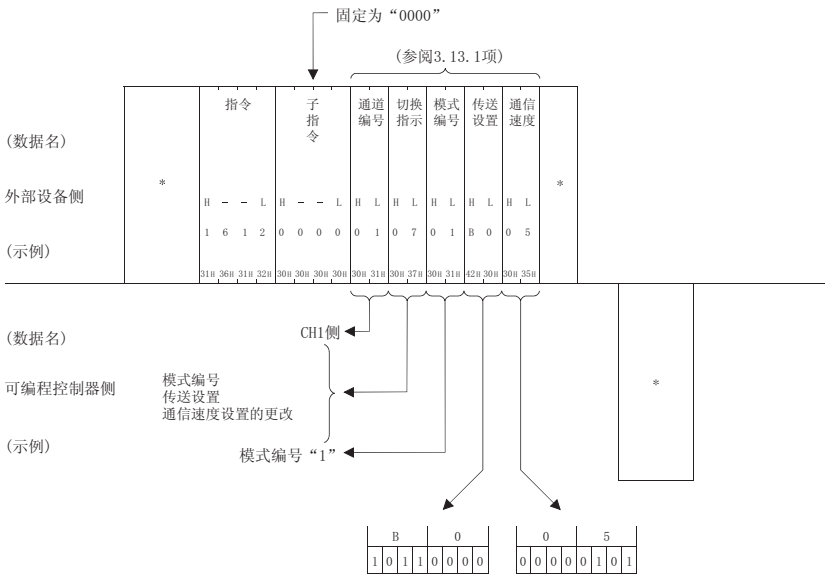
4) (b)中所示的切换指示的“传送规格切换、通信速度切换指示”为“0”时,对本项目以虚拟键值“00”(ASCII代码的情况下)进行指定。

3.13.2 模式切换(指令：1612)

以下对从外部设备进行 C24 的模式切换的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的帧・格式而有所不同。
请参阅 3.1.4 项～3.1.6 项中所示的详细说明。

[控制步骤]

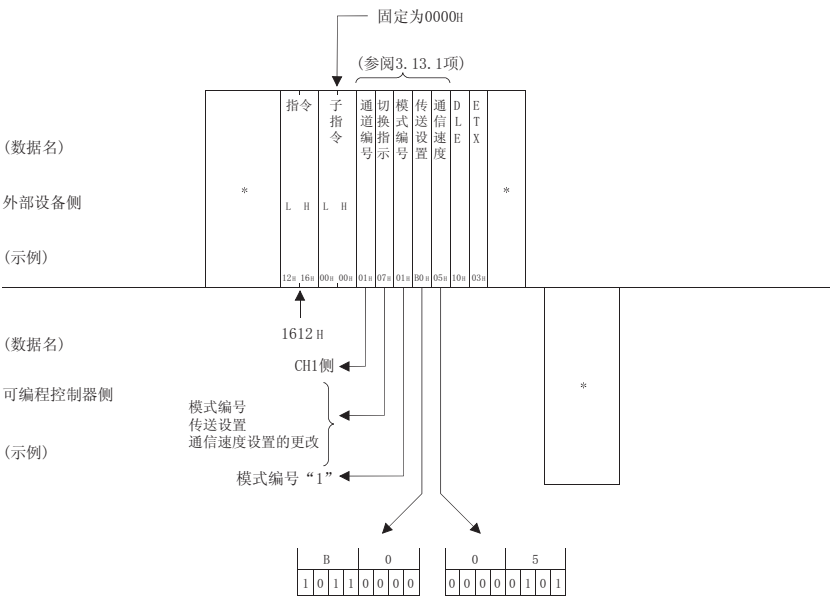
(1) 通过由 ASCII 代码进行的通信，对 CH1 侧接口进行模式切换的情况下



* 上图的模式切换内容

指定项目	指定内容	对应开关(设置值)
动作模式	MC 协议的格式 1	通信协议设置(1)
动作设置	独立动作	位 0=OFF
数据位设置	7 位	位 1=OFF
奇偶校验位的有无设置	无	位 2=OFF
偶数奇偶校验/奇数奇偶校验的设置	奇数	位 3=OFF
停止位设置	2 位	位 4=ON
和校验的有无设置	有	位 5=ON
运行中写入允许・禁止设置	禁止运行中写入	位 6=OFF
设置更改允许・禁止设置	更改允许	位 7=ON
通信速度	9600bps	通信速度设置(5n)

(2) 通过由二进制代码进行的通信，对 CH1 侧接口进行模式切换的情况下



* 上图的模式切换内容如(1)中所示。

3.14 显示 LED 的熄灯、通信出错信息・出错代码的初始化：用于串行通信模块

该功能是外部设备使 C24 的 ERR LED 熄灯，对缓冲存储器中存储的通信出错信息、出错代码进行初始化的功能。

通过对于指令报文的异常响应的响应等对当前的出错信息进行初始化以返回正常时的信息、对缓冲存储器的出错代码存储区域进行初始化时，使用本功能。

通过本功能进行的 C24 的处理与通过下述输出信号及缓冲存储器进行的处理相同。

根据 3.14.1 项 (2) (a) 中说明的表示子指令的值的位 0~3 的 ON/OFF 进行对应处理。请在参阅用户手册 (基本篇) 之后使用。

在本节中仅对指令的使用方法有关内容进行说明。

表示子指令的指定值的存储器的位状态	进行相同处理的输出信号、缓冲存储器	处理概要	参照
位 0 为 1 (ON) 时	出错信息初始化请求信号 (YE)	由于 CH1 侧的出错发生而亮灯的 ERR LED 的熄灯、输入信号 XE 的 OFF、缓冲存储器的 CH1 侧出错代码的初始化	用户手册 (基本篇)
位 1 为 1 (ON) 时	出错信息初始化请求信号 (YF)	由于 CH2 侧的出错发生而亮灯的 ERR LED 的熄灯、输入信号 XF 的 OFF、缓冲存储器的 CH1 侧出错代码的初始化	
位 2 为 1 (ON) 时	CH1 侧通信出错信息初始化请求区域 (缓冲存储器地址 0h)	通信出错信息的初始化	
位 3 为 1 (ON) 时	CH2 侧通信出错信息初始化请求区域 (缓冲存储器地址 1h)	通信出错信息的初始化	

要点
(1) 应在对根据表示子指令的值的位 0~3 的 ON/OFF 而进行的 C24 的处理概要进行确认的基础上，进行对象通信出错信息、出错代码的初始化。 建议对各接口分别进行或两个接口同时进行。 (以使子指令变为 0005h/000Ah/000Fh 的数值进行指定。)
(2) 只有对与外部设备相连接的 C24 (多点连接站也包含在内) 才可以使用本功能。 对于经由网络系统的其它站的 C24，不能使用本功能。

3.14.1 指令及字符部分的内容

以下对从外部设备进行 C24 的显示 LED 的熄灯、通信出错信息的初始化时的指令以及控制步骤内的字符部分 (通过二进制代码进行通信时为数据部分) 有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
			STOP 中	RUN 中		
				写入允许 设置	写入禁止 设置	
显示 LED 的熄灯、出错代码的初始化	1617 (000□)	进行显示 LED 的熄灯、出错代码的初始化等。	○	○	○	3. 14. 2 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

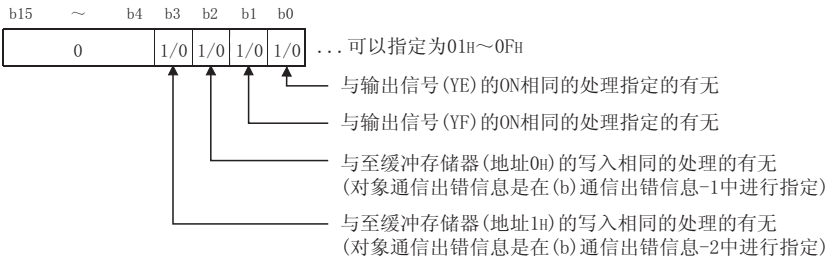
(2) 字符部分的内容

以下对从外部设备进行 C24 的通信出错信息・出错代码的初始化时的字符部分的内容进行说明。

(a) 子指令

是用于指定 C24 侧的处理内容的数据。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 子指令的指定值及指定内容如下所示。
通过相应位的 1(ON)，可以指定对应处理的执行。

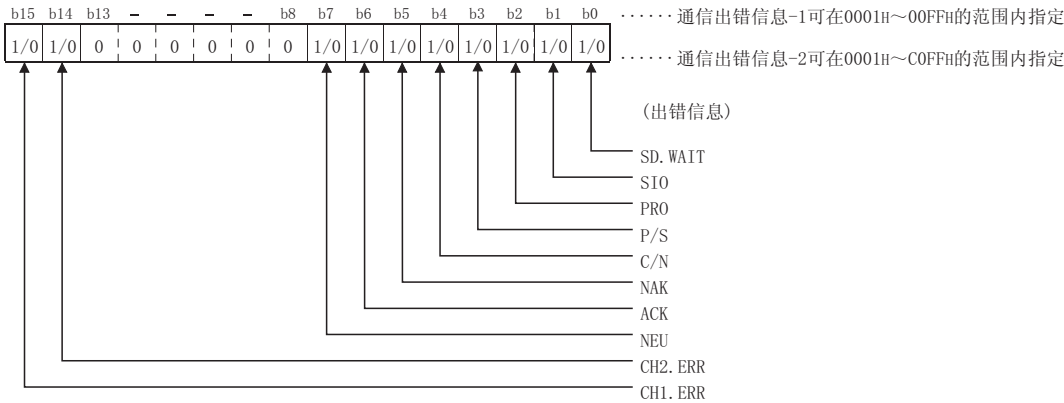


(b) 通信出错信息-1、通信出错信息-2

是用于表示进行初始化的通信出错信息的数据。

- 通信出错信息-1(主要是对 CH1 侧的指定)
- 通信出错信息-2(主要是对 CH2 侧的指定)
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用，从高位开始进行发送。
(示例)
将 CH2 侧的通信出错信息全部作为对象时变为“COFF”，从“C”开始按顺序进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。
- 3) 通信出错信息-1、通信出错信息-2 项目的各指定值及指定内容如下所示。
通过相应位的 1(ON)，可以指定进行初始化的通信出错信息。

子指令	通信出错信息-1	通信出错信息-2	备注
0001H~0003H	00H	00H	00H 以外根据下图进行指定
0004H~0007H	0001H~00FFH	00H	
0008H~000BH	00H	0001H~C0FFH	
000CH~000FH	0001H~00FFH	0001H~C0FFH	

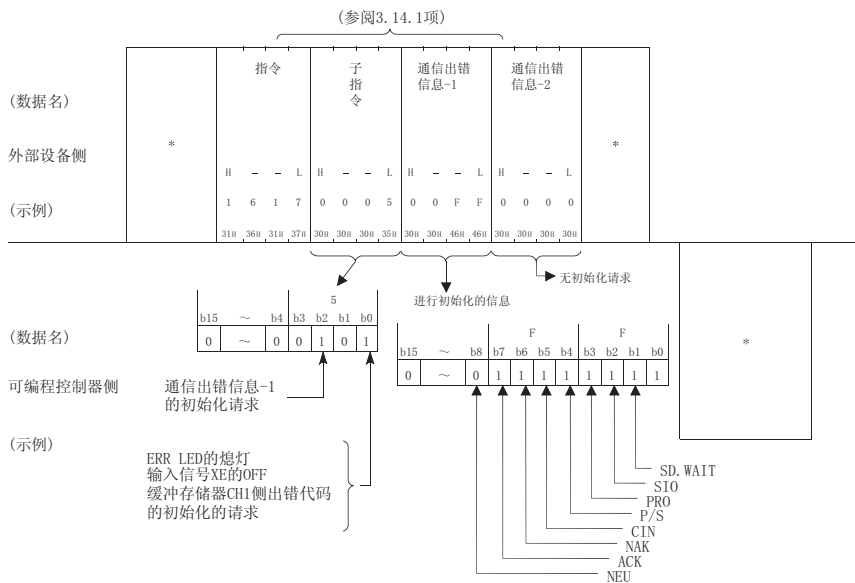


3.14.2 显示 LED 的熄灯、通信出错信息・出错代码的初始化(指令：1617)

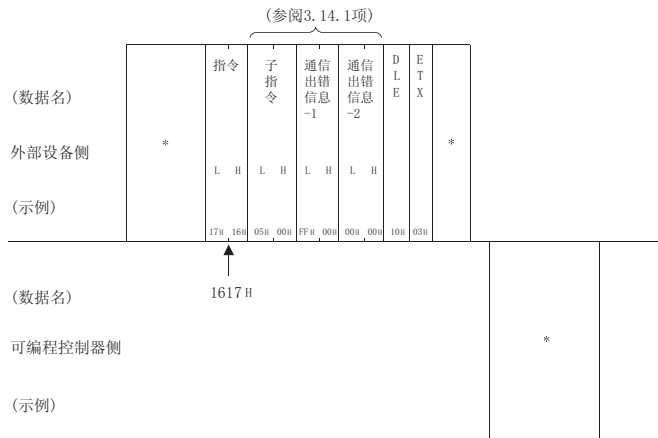
以下对从外部设备进行 C24 的 ERR LED 熄灯、通信出错信息・出错代码的初始化的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的帧・格式而有所不同。
请参阅 3.1.4 项～3.1.6 项中所示的详细说明项。

[控制步骤]

- (1) 通过由 ASCII 代码进行的通信，进行 CH1 侧接口相关 ERR LED 的熄灯、输入信号 XE 的 OFF、缓冲存储器的 CH1 侧出错代码的初始化、通信出错信息-1 项目中指定的出错信息的初始化的情况下



- (2) 通过由二进制代码进行的通信，进行 CH1 侧接口相关 ERR LED 的熄灯、输入信号 XE 的 OFF，缓冲存储器的 CH1 侧出错代码的初始化、通信出错信息-1 项目中指定的出错信息的初始化的情况下



3.15 COM.ERR.LED 的熄灯：用于以太网模块

是外部设备使 E71 前面的 COM.ERR.LED 熄灯的功能。

要点
只能对与进行 COM.ERR.LED 的熄灯请求的外部设备处于同一个以太网内的 E71 使用本功能。 对于经由网络系统的其它站 E71，不能使用本功能。

3.15.1 指令及字符部分的内容

以下对从外部设备进行 E71 的显示 LED 的熄灯时的指令以及控制步骤内的字符部分有关内容进行说明。

(1) 指令

— 功能	指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
			STOP 中	RUN 中		
				写入允许 设置	写入禁止 设置	
显示 LED 的熄灯	1617 (0000)	进行显示 LED 的熄灯。	○	○	○	3.15.2 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

根据通信代码仅对子指令进行发送。

(a) 子指令

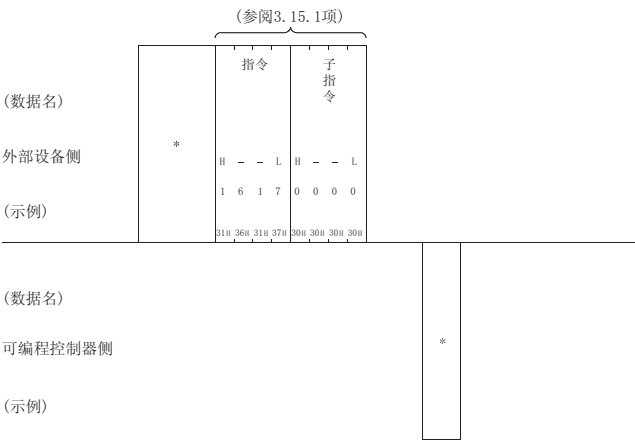
- 1) 通过 ASCII 代码进行数据通信时
将“0000”转换为 ASCII 代码 2 位(16 进制数)后从高位开始进行发送。
- 2) 通过二进制代码进行数据通信时
对 0000_H 进行发送。

3.15.2 COM.ERR.LED 的熄灯 (指令: 1617)

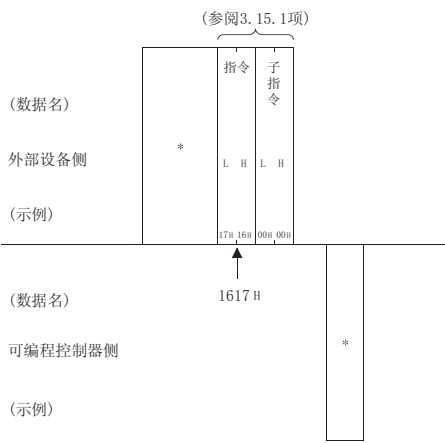
以下对从外部设备进行 E71 前面的显示 LED 的熄灯的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 3.1.2 项~3.1.3 项中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信执行 COM.ERR.LED 的熄灯的情况下



(2) 通过由二进制代码进行的通信执行 COM.ERR.LED 的熄灯的情况下



3.16 回送测试

回送测试是指，对外部设备与 C24/E71 的通信功能能否正常动作进行测试的功能。以下对该功能时的控制步骤进行举例说明。

要点
(1) 在 C24/E71 启动时或发生故障时，通过本回送测试可以确认外部设备与 C24/E71 的连接是否正确，数据通信功能是否正常动作。
(2) 只能对与外部设备相连接的 C24/E71 (C24 也包括多点连接站在内) 使用本功能。
对于经由网络系统的其它站 C24/E71，不能使用本功能。

3.16.1 指令及字符部分的内容

以下对从外部设备进行与 C24/E71 的回送测试时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)的有关内容进行说明。

(1) 指令

功能	指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
			STOP 中	RUN 中		
				写入允许 设置	写入禁止 设置	
回送测试	0619 (0000)	确认数据通信是否正常进行。	○	○	○	3.16.2 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下对外部设备进行与 C24/E71 的回送测试时的字符部分的内容进行说明。

(a) 回送数据数

是用于表示回送数据部分的字节数的数据。

- 1) 通过 ASCII 代码进行数据通信时
将字节数转换为 ASCII 代码 4 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示字节数的 2 字节的数值，从 Low 字节(L: 位 0~7)开始进行发送。

(b) 回送数据

是用于回送测试中用于指定进行数据发送接收的报文的用户数据的数据。

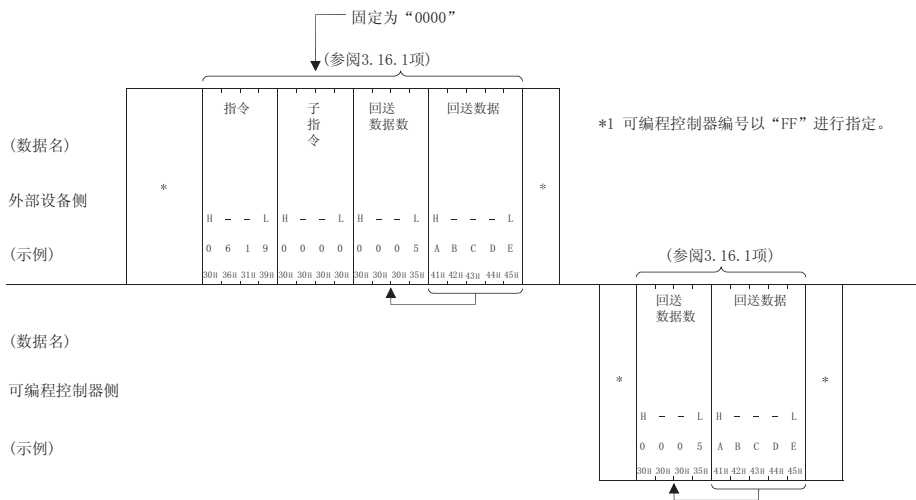
- 1) 通过 ASCII 代码进行数据通信时
将最多 960 字符的半角字符串(“0”~“9”、“A”~“F”)的排列从起始开始进行发送。
- 2) 通过二进制代码进行数据通信时
将半角字符(“0”~“9”、“A”~“F”)的排列的各字符代码作为 1 字节的数值，将最多 960 字节从起始字符代码开始进行发送。

3.16.2 回送测试(指令：0619)

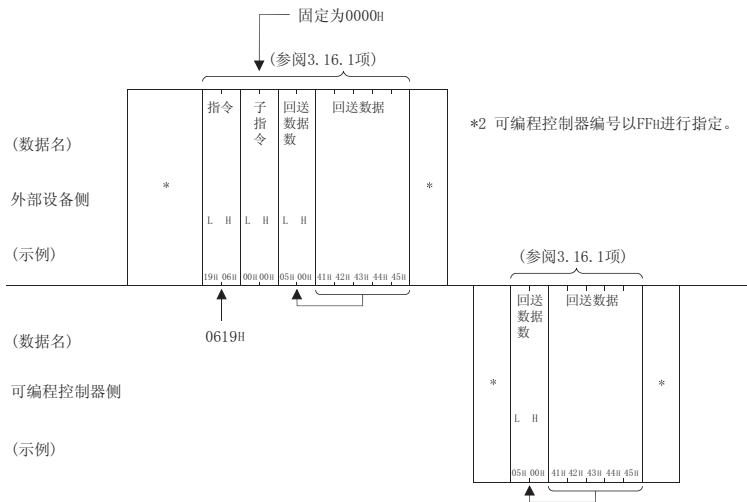
以下对外部设备进行与 C24/E71 的回送测试的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，进行回送测试的情况下(回送数据为“ABCDE”)



(2) 通过由二进制代码进行的通信，进行回送测试的情况下(回送数据为“ABCDE”)



要点
对于外部设备发送的回送数据数及回送数据，原样不变地以相同内容返回至外部设备。

3.17 可编程控制器 CPU 监视登录/解除：用于串行通信模块

可编程控制器 CPU 监视登录/解除是指，为了使用 C24 的可编程控制器 CPU 监视功能，从外部设备/可编程控制器 CPU 进行可编程控制器 CPU 监视登录(包括开始指示)或可编程控制器 CPU 监视解除的功能。

通过使用该可编程控制器 CPU 监视功能，外部设备可以在用户指定的时机接收事先登录的软元件数据、CPU 异常信息或信息。

在本节中，对可编程控制器 CPU 监视功能的概要、进行 MC 协议通信的外部设备为了使用可编程控制器 CPU 监视功能而进行可编程控制器 CPU 监视登录/解除的指令的使用方法有关内容进行说明。

使用可编程控制器 CPU 监视功能时，应在参阅了用户手册(应用篇)之后，进行本节中所示的可编程控制器 CPU 监视登录/解除。

在用户手册(应用篇)中，对可编程控制器 CPU 监视功能的详细内容、执行步骤、至外部设备的信息发送的报文格式以及从可编程控制器 CPU 进行可编程控制器 CPU 监视登录/解除的方法等有关内容进行了说明。

(1) 可编程控制器 CPU 监视功能的概要

- (a) C24 以用户事前登录的下述监视信息为对象，对可编程控制器 CPU 进行监视。
 - 字软元件中存储的数值
 - 位软元件的 ON/OFF 状态
 - 本站可编程控制器 CPU 的状态(异常发生的有无)
- (b) 监视信息达到用户指定的信息发送条件时，或以恒定的时间间隔，C24 将用户指定的下述监视信息等发送至外部设备。
(不需要通过顺控程序进行发送处理。)
 - 用户登录的多个软元件数据或 CPU 异常信息。
(也可通过与调制解调器功能并用进行监视信息的发送。)
- * 对于通过可编程控制器 CPU 监视功能进行的至外部设备的信息发送，与下述时机重叠时，将以与通过 3.11 项中所示的接通请求功能进行的数据发送相同的时机进行发送。(参阅 3.11.1 项(3))
 - 通过本功能进行的信息发送与通过 MC 协议进行的通信的响应报文的发送时机重叠时。
 - 进行半双工通信的情况下，正在接收来自于外部设备的数据时。
- (c) 通过对从外部设备或可编程控制器 CPU 进行的可编程控制器 CPU 监视进行登录，开始由 C24 进行的可编程控制器 CPU 的监视。此外，通过从外部设备或可编程控制器 CPU 进行可编程控制器 CPU 的监视解除，可编程控制器 CPU 的监视将结束。

(2) 通过可编程控制器 CPU 监视功能进行信息发送的对象接口

信息发送是对设置了 MC 协议或无顺序协议的接口进行的。

* 关于通过无顺序协议进行的可编程控制器 CPU 监视功能，请参阅用户手册(应用篇)。

在本手册中，省略了通过无顺序协议进行的可编程控制器 CPU 监视功能的有关内容。

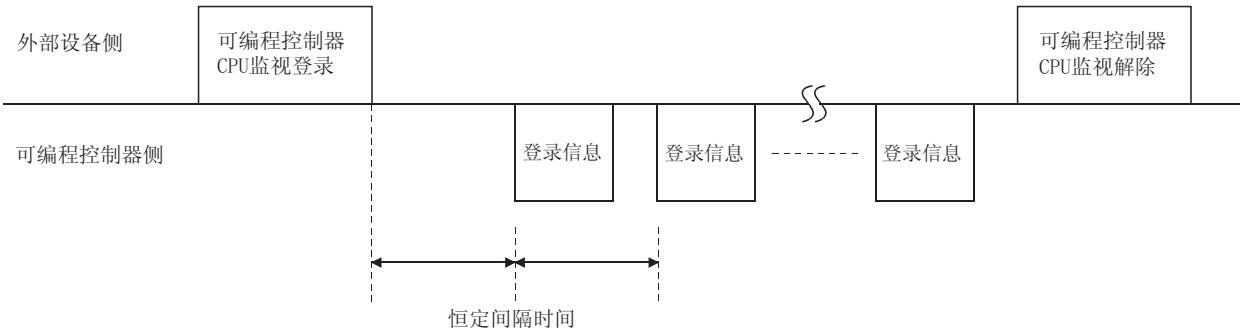
(3) 通过可编程控制器 CPU 监视功能进行的信息发送的报文格式

关于信息发送的报文格式，与通过 3. 11 项中所示的接通请求功能进行发送的报文格式相同。(参阅 3. 11. 2 项)
* 接通请求数据的发送数据部分即为本功能使用的数据。

(4) 控制步骤

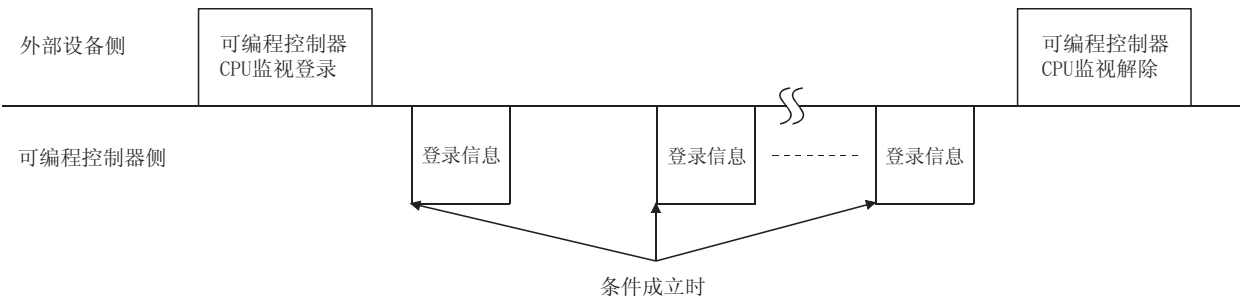
(示例)

以恒定周期发送的情况下(登录 1 块时)



(示例)

条件一致发送的情况下(登录 1 块时)



* 对于登录信息的发送，也可以与通过接通请求功能用的用户登录帧进行的发送相同的方法进行。
请参阅用户手册(应用篇)。

要点	
(1)	对于可编程控制器 CPU 监视功能的可编程控制器 CPU 监视登录，可以通过外部设备、可编程控制器 CPU、GX Works2 或 GX Configurator-SC 进行。
(2)	进行用于监视软元件等登录的，从外部设备・可编程控制器 CPU 至 C24 的可编程控制器 CPU 监视登录时，不能进行重复登录。 已进行了可编程控制器 CPU 监视登录时，应在进行可编程控制器 CPU 监视解除之后再行可编程控制器 CPU 监视登录。
(3)	只能对与外部设备相连接的 C24(多点连接站也包含在内)使用本功能。 对于经由网络系统的其它站的 C24，不能使用本功能。

3.17.1 指令及字符部分的内容

以下对通过外部设备使用可编程控制器 CPU 监视功能时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)的有关内容进行说明。

(1) 指令

功能		指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
				STOP 中	RUN 中		
					写入允许 设置	写入禁止 设置	
可编程控制器 CPU 监视	登录	0630 (0000)	对软元件存储器的监视、CPU 状态的监视进行登录， 开始可编程控制器 CPU 监视的动作。	○	○	○	3. 17. 2 项
	解除	0631 (0000)	结束可编程控制器 CPU 监视的动作。				3. 17. 3 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下对通过外部设备使用可编程控制器 CPU 监视功能时的字符部分的内容进行说明。

(a) 周期时间单位

对周期时间(b)的单位进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用，从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 指定值与时间单位的对应如下所示。

指定值	时间单位
00 _h	100ms
01 _h	1 秒
02 _h	1 分

- 4) 对于以本周期时间单位及(b)周期时间指定的时间，将成为 C24 从可编程控制器 CPU 中读取监视信息的时间间隔，也将作为恒定周期发送的发送时间间隔使用。

(b) 周期时间

对 C24 从可编程控制器 CPU 中读取监视信息的时间间隔(1 个周期的时间)进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将周期时间(1~65535)转换为 ASCII 代码 4 位(16 进制数：“0001”~“FFFF”)后使用，从高位(“0001”的情况下为“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用表示周期时间(1~65535)的 2 字节的数值(0001_h~FFFF_h)，从 Low 字节(L: 位 0~7)开始进行发送。

(c) 可编程控制器 CPU 监视功能(周期监视功能)

对将可编程控制器 CPU 的监视结果信息(软元件信息/CPU 状态信息)发送至外部设备时的发送时机(恒定周期发送/条件一致发送)进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
使用下述 1 字节的数值进行发送。
- 3) 指定值与通过可编程控制器 CPU 监视功能进行发送的方法的对应如下所示。

指定值	通过可编程控制器 CPU 监视功能进行发送的方法	信息的发送时机
01 _H	恒定周期发送	以周期时间间隔进行信息发送
02 _H	条件一致发送	与指定条件一致时进行信息发送

- 4) 对于上述(a)、(b)的数据项目中指定的从可编程控制器 CPU 中读取信息的时间间隔, 也将作为恒定周期发送的发送间隔时间使用。
- 5) 对于条件一致发送的发送时机, 在下述(h)的数据项目中进行指定。
对于条件一致发送的条件, 在下述(i)、(j)的数据项目中进行指定。
- 6) 对软元件监视进行条件一致发送时, 下述(g)中指定的各块的起始软元件将成为监视对象。

(d) 可编程控制器 CPU 监视发送方式

对将可编程控制器 CPU 的监视结果传送至外部设备的方式进行指定。

在通过 MC 协议进行的通信中, 只能指定用于“数据发送”的下述数据。

(数据发送: 软元件数据、CPU 状态数据的发送)

- 1) 通过 ASCII 代码进行数据通信时
对“00”进行发送。
- 2) 通过二进制代码进行数据通信时
对 1 字节的数值(0_H)进行发送。

(e) 登录字块数、登录位块数

作为进行软元件数据的监视以及发送时的对象, 对 C24 中登录的字软元件的块数(登录字块数)及位软元件的块数(登录位块数)进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将块数转换为 ASCII 代码 2 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
对表示块数的 1 字节的数值进行发送。
- 3) 对于通过本功能进行的至 C24 的软元件登录, 应按如下所示进行指定。
10 块 ≥ (全部字块数+全部位块数)
960 点 ≥ (全部字块的合计软元件点数+全部位块的合计软元件点数)
- 4) 对于各块的登录软元件范围, 在下述(g)的项目中进行指定。

(f) CPU 异常监视

指定进行可编程控制器 CPU 监视时是否包含本站可编程控制器 CPU 的异常监视在内。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
对下述 1 字节的数值进行发送。
- 3) 指定值与 CPU 异常监视指定的对应如下所示。

指定值	CPU 异常监视指定
00 _H	不包含本站可编程控制器 CPU 的异常监视在内。
01 _H	包含本站可编程控制器 CPU 的异常监视在内。

(g) 软元件代码、监视起始软元件、登录点数(读取点数)

作为可编程控制器 CPU 监视进行软元件数据的监视以及发送时, 对(e)登录字块数、登录位块数中指定的块数的各块的软元件范围进行指定。

* 对于通过可编程控制器 CPU 监视功能进行的软元件数据的监视, 各块的起始软元件将成为对象。

字软元件指定的块 : 起始字软元件(1 字)

位软元件指定的块 : 起始位软元件(1 位)

- 1) 软元件代码是用于识别相应块的对象软元件的数据。
- 2) 监视起始软元件是用于指定相应块的对象软元件范围的起始位置的数据。
- 3) 登录点数(读取点数)是用于指定相应块的对象软元件范围从起始开始的点数的数据。
对于位软元件, 以字单位(1 点=16 位)进行点数指定。
- 4) 关于各数据的指定方法, 与 3.3 节中所示的进行软元件存储器的读取・写入时的指定方法相同。
应按照 3.3.1 项(2)(c)2)~4)的说明进行指定。

(h) 条件一致发送的发送方法

在可编程控制器 CPU 监视功能(c)中指定条件一致发送时, 对发送可编程控制器 CPU 的监视结果时的发送方法进行指定。

- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
- 2) 通过二进制代码进行数据通信时
对下述 1 字节的数值进行发送。
- 3) 在可编程控制器 CPU 监视功能(c)中指定恒定周期发送时, 指定为“00”或 00_H。
- 4) 指定值与条件一致发送时的发送方法的对应如下所示。

指定值	条件一致发送的发送方法	发送方法
00 _H	边沿触发发送	仅监视条件与读取的数据一致时, 对监视结果进行发送。
01 _H	电平触发发送	在监视条件与读取的数据一致期间, 以周期时间间隔对监视结果进行发送。

- (i) 监视条件
- 在可编程控制器 CPU 监视功能(c)中指定条件一致发送时, 对监视条件值(j)的条件(监视结果的发送条件)进行指定。
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 2 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
对下述 1 字节的数值进行发送。
 - 3) 在可编程控制器 CPU 监视功能(c)中指定恒定周期发送时, 指定为“00”或 00H。
 - 4) 指定值与监视条件的对应如下所示。

指定 值	监视条件(判定内容)		指定有效软元件	
			位	字
01H	软元件值或状态 = 软元件监视条件值或状态		○	○
02H	软元件值或状态 ≠ 软元件监视条件值或状态			
03H	无符号	软元件值 ≤ 软元件监视条件值	×	
04H		软元件值 < 软元件监视条件值		
05H		软元件值 ≥ 软元件监视条件值		
06H		软元件值 > 软元件监视条件值		
07H	有符号	软元件值 ≤ 软元件监视条件值		
08H		软元件值 < 软元件监视条件值		
09H		软元件值 ≥ 软元件监视条件值		
0AH		软元件值 > 软元件监视条件值		

- (j) 监视条件值
- 在可编程控制器 CPU 监视功能(c)中指定条件一致发送时, 对监视条件(i)的对象状态/数值进行指定。
- 监视软元件为字软元件的情况下: 监视条件值以数值进行指定
 - 监视软元件为位软元件的情况下: 监视条件以对应于 ON/OFF 的数值进行指定
- 1) 通过 ASCII 代码进行数据通信时
将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用, 从高位(“0”)开始进行发送。
 - 2) 通过二进制代码进行数据通信时
使用下述 2 字节的数值, 从 Low 字节(L: 位 0~7)开始进行发送。
 - 3) 在可编程控制器 CPU 监视功能(c)中指定恒定周期发送时, 指定为“0000”或 0000H。
 - 4) 指定值与监视条件值的对应如下所示。

指定值	监视条件值或状态	监视软元件的类型
0000H	OFF	位软元件
0001H	ON	
000H~FFFFH	数值	字软元件

- (k) 读取数据(作为监视结果发送至外部设备的软元件数据)
 - 1) 是从可编程控制器 CPU 中读取的 1 个块的监视对象的软元件数据。
 - 2) 各数据的排列及内容如 3. 17. 4 项中所示。
对于位软元件，以字单位(1 点=16 位)读取登录的点数。
对于字软元件，以 1 点单位读取登录的点数。
- (1) 读取数据(作为监视结果发送至外部设备的 CPU 状态数据)
 - 1) 是从可编程控制器 CPU 中读取的可编程控制器 CPU 的状态信息。
 - 2) 通过 ASCII 代码进行数据通信时
从可编程控制器 CPU 侧以 ASCII 代码 4 位(16 进制数)发送下述数值。
(参阅 3. 17. 4 项)
 - 3) 通过二进制代码进行数据通信时
从可编程控制器 CPU 侧发送下述 2 字节的数值。(参阅 3. 17. 4 项)
 - 4) 读取数据与可编程控制器 CPU 的状态的对应如下所示。

读取数据	可编程控制器 CPU 的状态	备注
0000 _H	正常动作中	—
0001 _H	模块报警发生中	根据 Q/LCPU 的手册对出错内容进行确认并处理。
0002 _H	模块出错/模块系统出错发生中	

备注

对于除上述以外的用于指定访问目标站的指令报文中的数据项目，应指定以本站为对象的以下数据。

数据项目	指定数据	
	通过 ASCII 代码进行数据通信时	通过二进制代码进行数据通信时
网络编号	“00”	00 _H
可编程控制器编号	“FF”	FF _H
请求目标模块 I/O 编号	“03FF”	03FF _H
请求目标模块站号	“00”	00 _H
本站编号	“00”	00 _H

3.17.2 可编程控制器 CPU 监视登录(指令：0630)

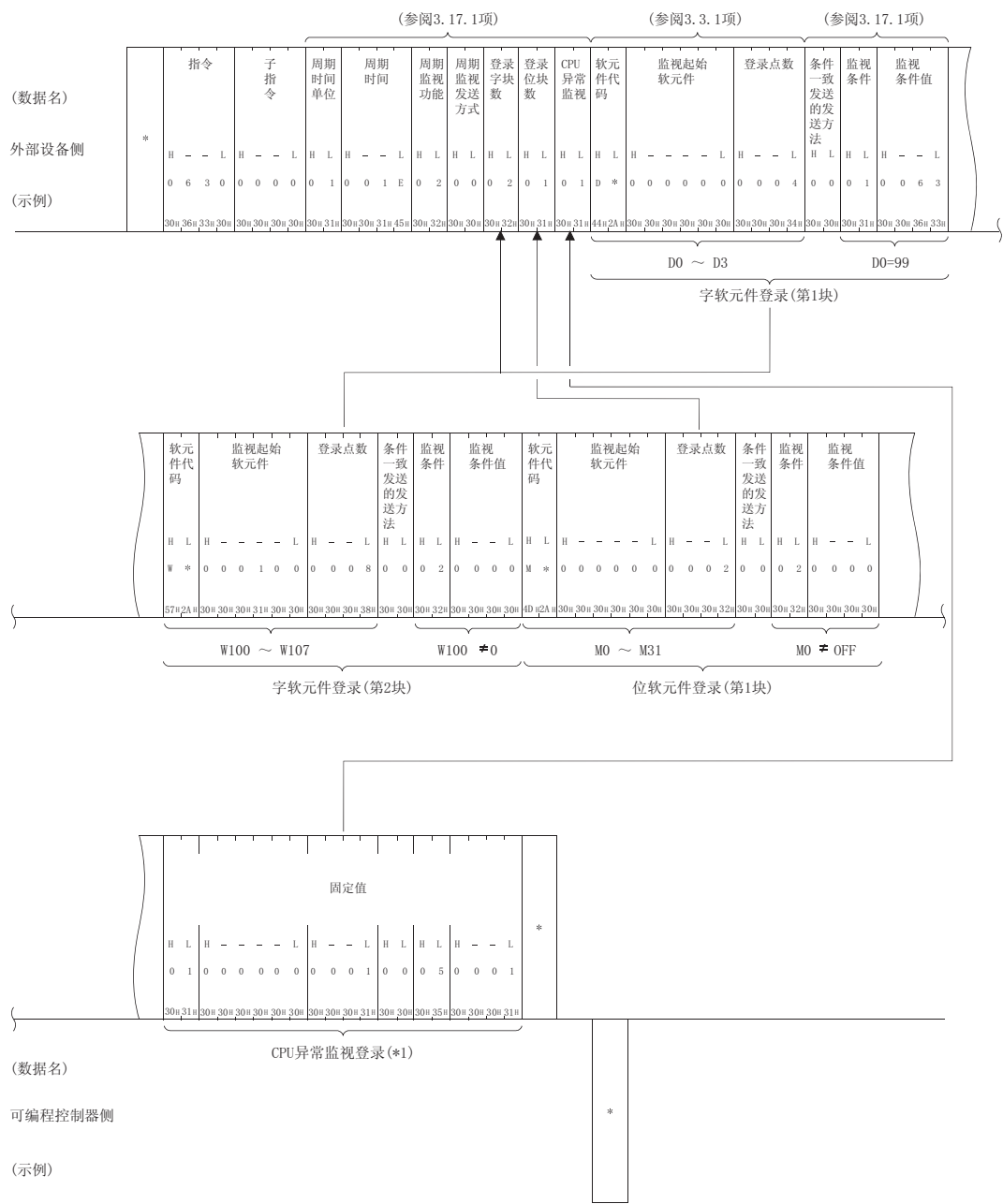
以下对从通信外部设备进行可编程控制器 CPU 监视登录时的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的帧・格式而有所不同。
请参阅 3.1.4 项～3.1.6 项中所示的详细说明。

[控制步骤]

进行以下可编程控制器 CPU 监视登录的情况下

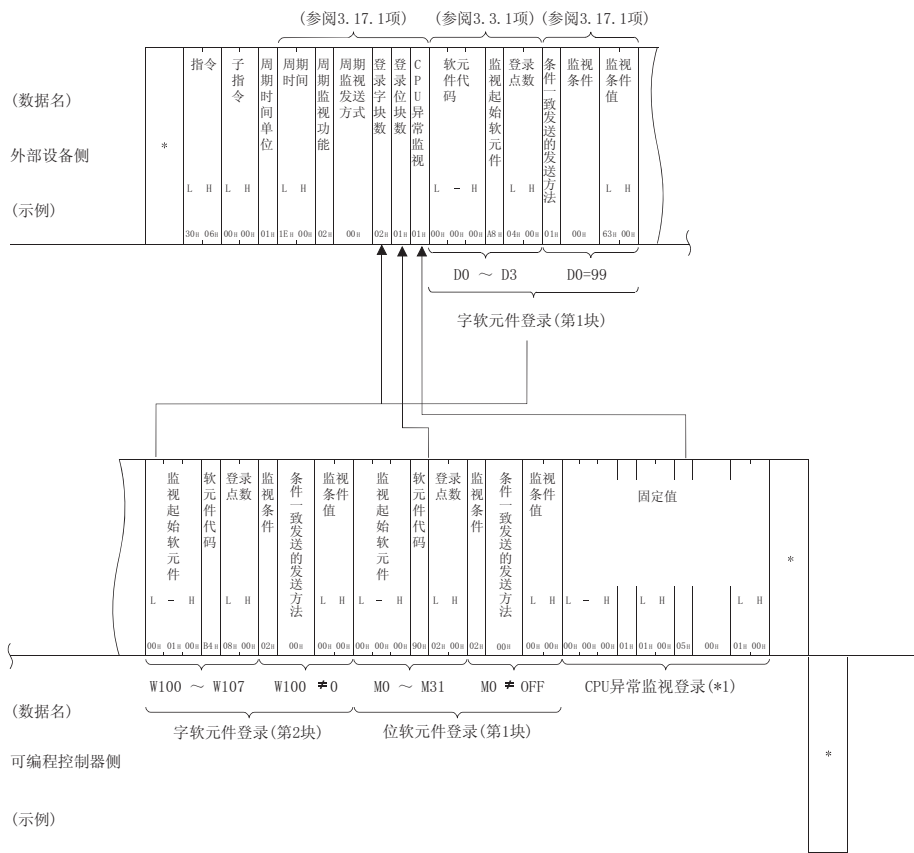
数据名		登录值	数据名		登录值
周期时间单位\周期时间		30 秒	可编程控制器 CPU 监视功能		条件一致发送
可编程控制器 CPU 监视发送方式		软元件数据、CPU 状态数据的发送	登录字块数		2 块
			登录位块数		1 块
CPU 异常监视		包含	-		-
登录字软元件		(监视软元件=D0)	登录位软元件		(监视软元件=M0)
第 1 块	软元件	D0～D3 (4 点)	第 3 块	软元件	M0～M31 (2 点)
	发送方法	边沿触发发送		发送方法	边沿触发发送
	监视条件	软元件值=监视条件值		监视条件	软元件状态≠监视状态
	监视条件值	99		监视状态	OFF
登录字软元件		(监视软元件=W100)	-		-
第 2 块	软元件	W100～W107 (8 点)	-		-
	发送方法	边沿触发发送	-		-
	监视条件	软元件值≠监视条件值	-		-
	监视条件值	0	-		-

(1) 通过 ASCII 代码进行通信时



- *1 CPU 异常监视包含在监视条件中的情况下，软元件代码～监视条件值的项目中应指定如下所示的固定值。
- 软元件代码 : “01”
 - 监视起始软元件 : “000000”
 - 登录点数 : “0001”
 - 条件一致发送的发送方法 : “00”
 - 监视条件 : “05”
 - 监视条件值 : “0001”

(2) 通过二进制代码进行通信时



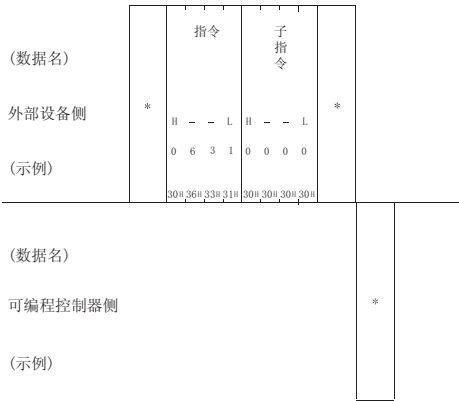
- *1 CPU 异常监视包含在监视条件中的情况下，软元件代码～监视条件值的项目中应指定如下所示的固定值。
- 监视起始软元件 : 000000H
 - 软元件代码 : 01H
 - 登录点数 : 0001H
 - 监视条件 : 05H
 - 条件一致发送的发送方法 : 00H
 - 监视条件值 : 0001H

3.17.3 可编程控制器 CPU 监视解除(指令：0631)

以下对从外部设备进行可编程控制器 CPU 监视解除时的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的帧・格式而有所不同。
请参阅 3.1.4 项～3.1.6 项中所示的详细说明。

[控制步骤]

(1) 通过 ASCII 代码进行通信时



(2) 通过二进制代码进行通信时



3.17.4 通过可编程控制器 CPU 监视功能发送的数据

以下举例说明通过外部设备进行可编程控制器 CPU 监视登录，从 C24 发送监视信息。

对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的帧・格式而有所不同。

GX Works2 或 GX Developer 中的通信协议设置为“1~4”时，以 A 兼容 1C 帧进行数据发送。关于 A 兼容 1C 帧的各格式的数据项目的排列・内容，请参阅 5.1 节。

GX Works2 或 GX Developer 中的通信协议设置为“5”时，以 QnA 兼容 4C 帧的格式 5 进行数据发送。关于 QnA 兼容 4C 帧的数据项目的排列・内容，请参阅 3.1.5 项、3.1.6 项。

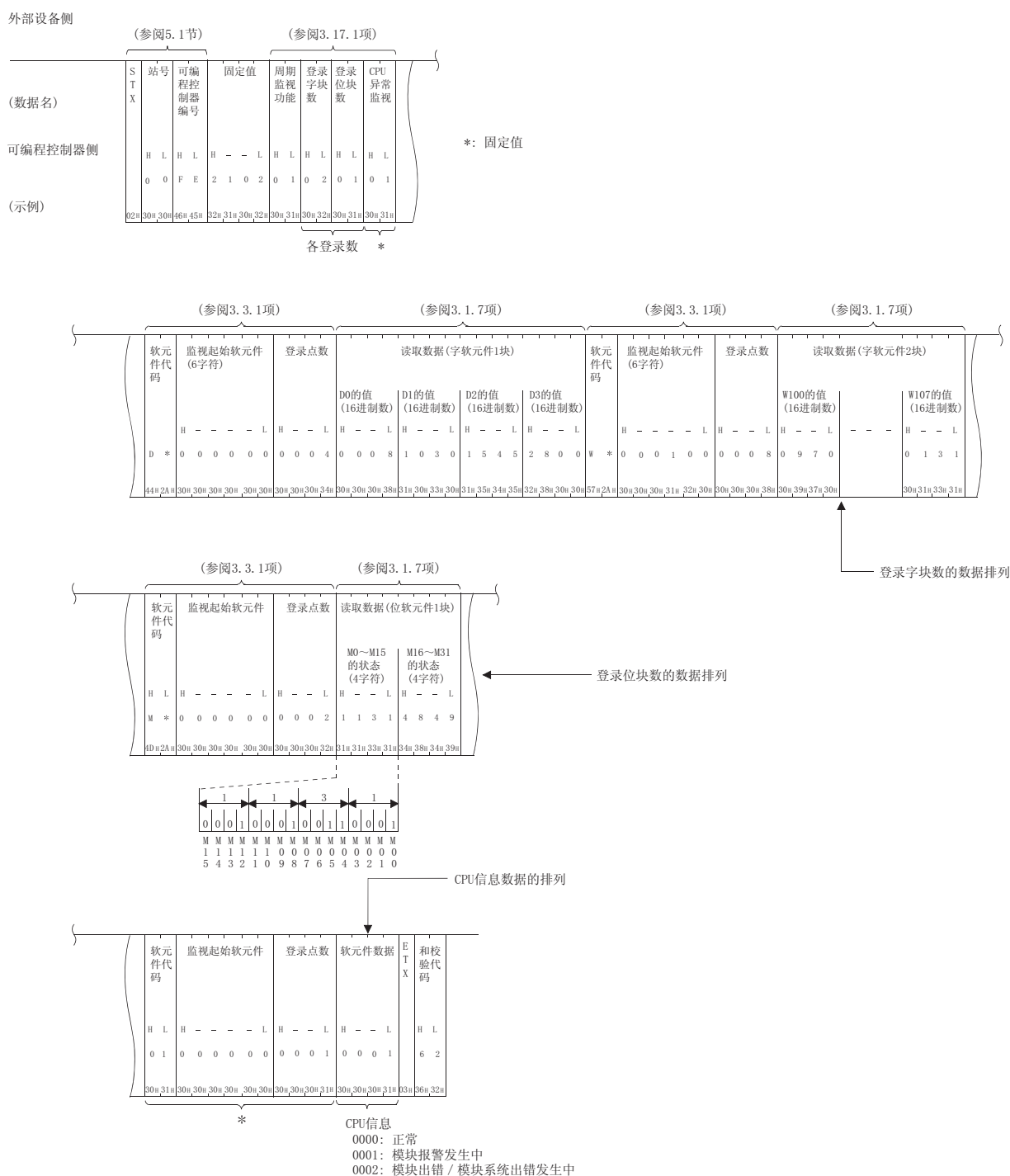
(1) 通过恒定周期发送对 CPU 状态数据及软元件数据进行发送的情况下

以下对以 A 兼容 1C 帧的格式 1 进行 3.17.2 项中所示的软元件数据登录时的发送示例进行说明。

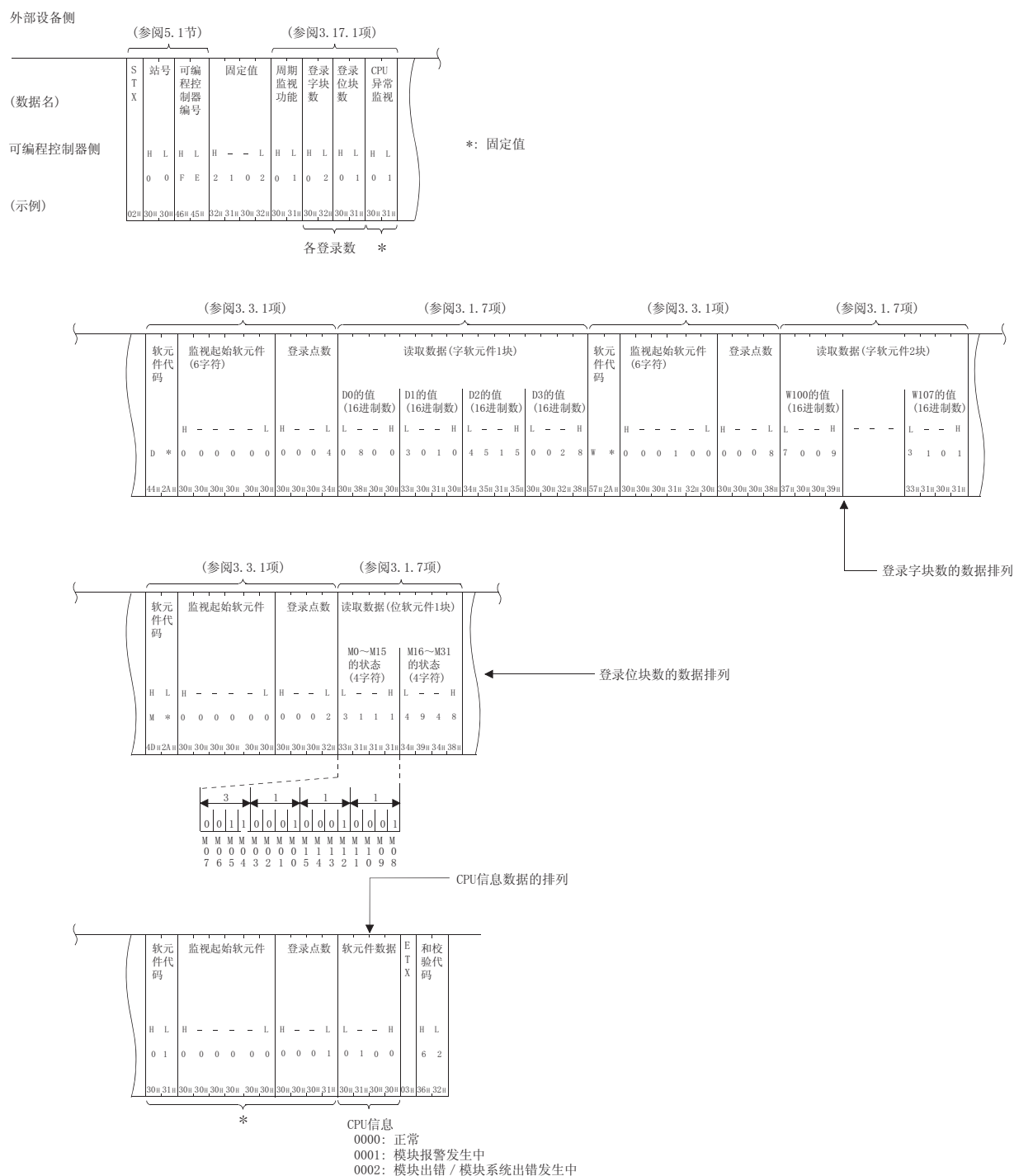
进行可编程控制器 CPU 监视登录时，在“可编程控制器 CPU 监视功能”的数据项目中指定了“恒定周期发送”时，登录数据的信息将被批量发送。

(a) 通过二进制代码进行通信时

- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”(字单位)时

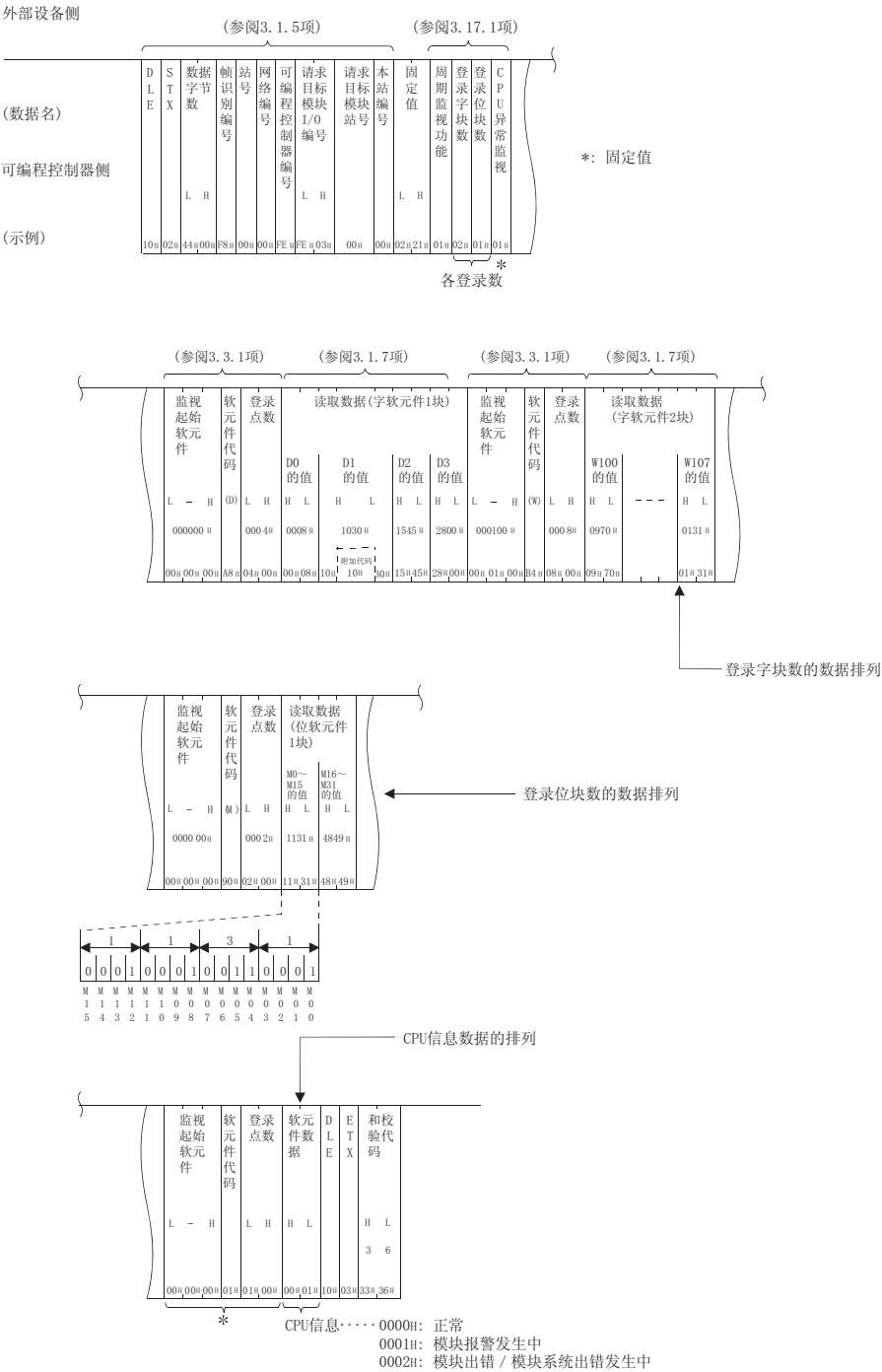


- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）时

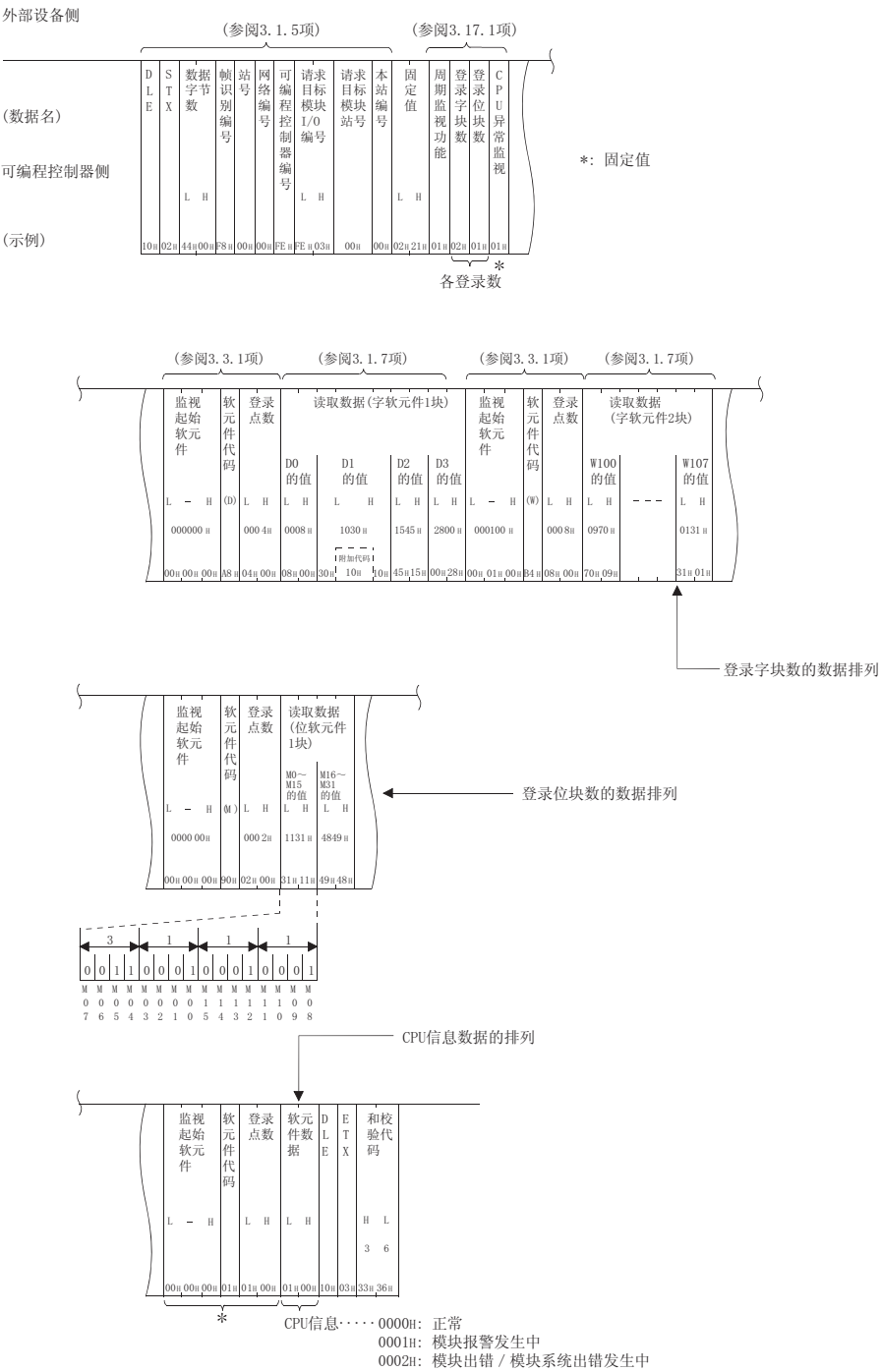


(b) 通过二进制代码进行通信时

- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”（字单位）时



- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）时



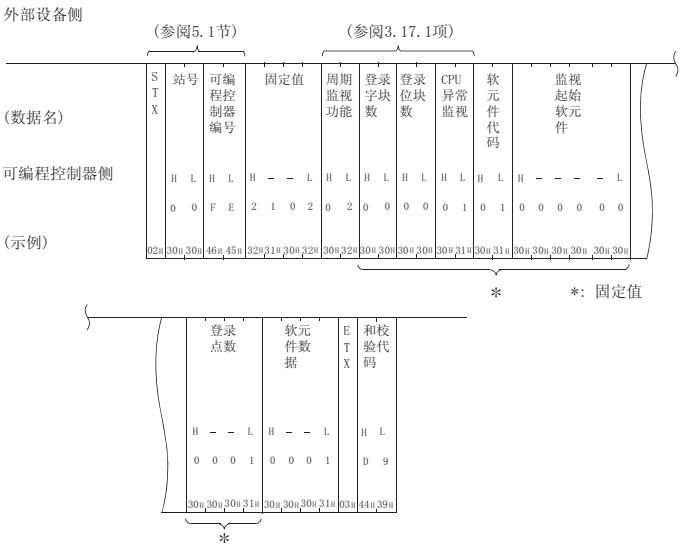
(2) 通过条件一致发送，对 CPU 状态数据或软元件数据进行发送的情况下

以下对以 A 兼容 1C 帧的格式 1 进行 3. 17. 2 项中所示的可编程控制器 CPU 监视登录时的发送示例进行说明。
进行可编程控制器 CPU 监视登录时，在“可编程控制器 CPU 监视功能”的数据项目中指定了“条件一致发送”的情况下，登录数据的信息将被逐个发送。
(软元件数据按块进行发送。)
关于各信息的内容，请参阅本项(1)。

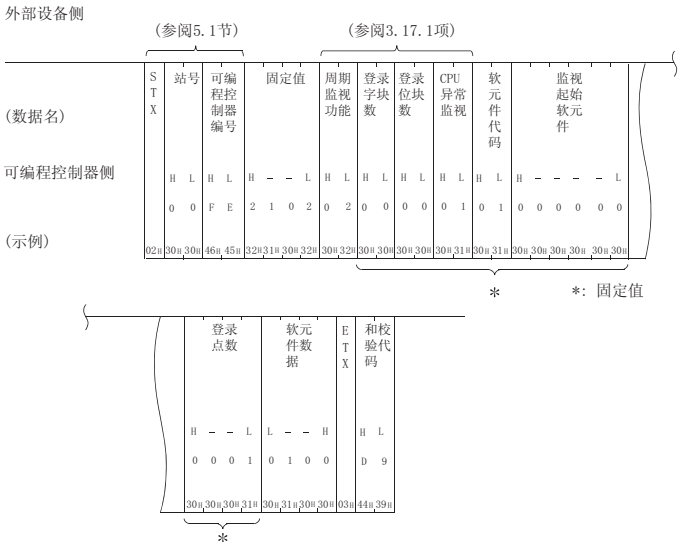
(a) 发送 CPU 状态数据时

1) 通过 ASCII 代码进行通信时

- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”(字单位)时



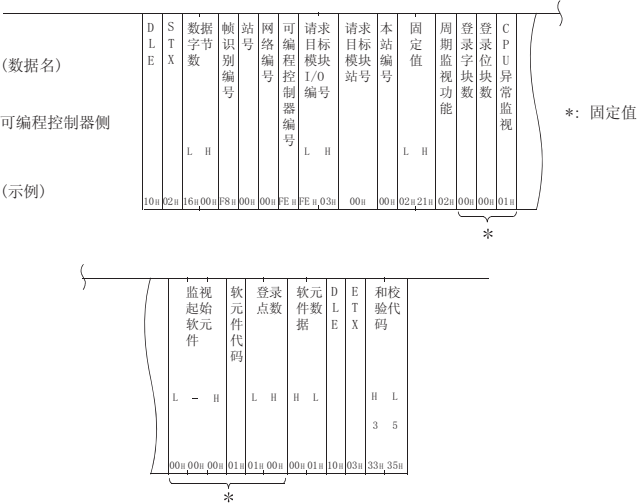
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”(字节单位)时



2) 通过二进制代码进行通信时

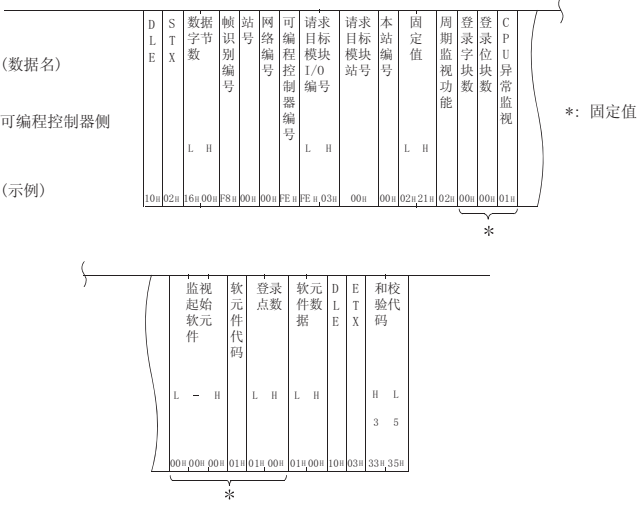
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”（字单位）时

外部设备侧



- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）时

外部设备侧



(b) 发送字软件元件数据时

进行了 3.17.2 项中所示的可编程控制器 CPU 监视登录时，由于 D0=99 进行的字软件元件数据 D0~D3 (4 点) 的发送示例如下所示。

1) 通过 ASCII 代码进行通信时

- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”(字单位)时

外部设备侧

(数据名)

S T X	站号	可编 程控 制器 编号	固定值	周期 监视 功能	登录 字块 数	登录 位块 数	CPU 异常 监视
H L	H L	H L	- - L	H L	H L	H L	H L
0 0	F E	2 1 0 2	0 2	0 1	0 0	0 0	

(示例)

*: 固定值

登录数

- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”(字节单位)时

外部设备侧

(数据名)

可编程序控制器側

(示例)

S T X	站号	可编 程控 制器 编号	固定值	周期 监视 功能	登录 字块 数	登录 位块 数	CPU 异常 监视
H L	H L	H L	- - - L	H L	H L	H L	H L
0 0	F E	2 1 0 2	0 2	0 2	0 1	0 0	0 0
02#	30# 30#	46# 45#	32# 31# 30# 32#	30# 32#	30# 31#	30# 30#	30# 30#

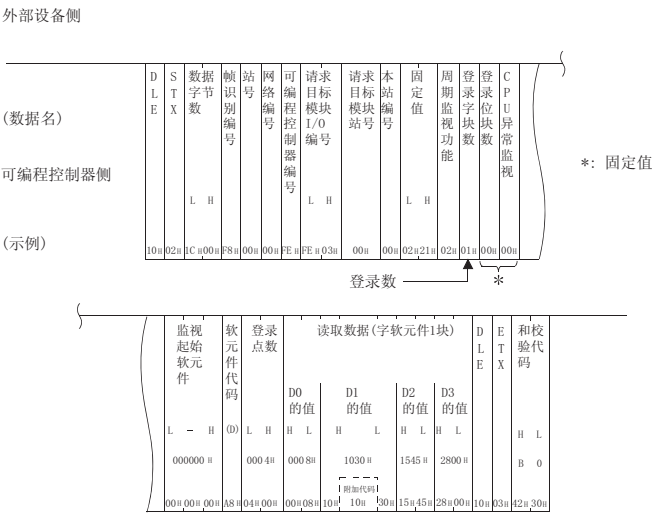
*: 固定值

登录数

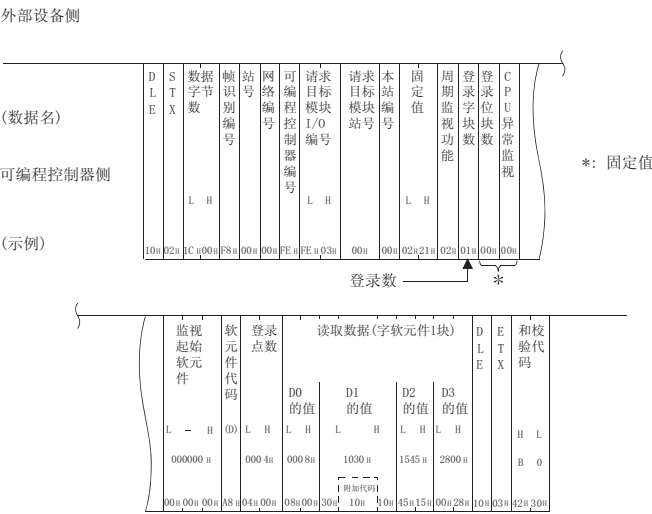
*

软元 件代 码	监视起始软元件 (6字符)	登录点数	读取数据(字软元件1块)				E T X	和校 验代 码
			D0的值 (16进制数)	D1的值 (16进制数)	D2的值 (16进制数)	D3的值 (16进制数)		
H - - - - L	H - - - L	L - - H	L - - H	L - - H	L - - H	L - - H	H L	
D * 0 0 0 0 0 0	0 0 0 4	0 8 0 0	3 0 1 0	4 5 1 5	0 0 2 8	C 2		
44# 2A#	30# 30# 30# 30# 30# 30#	30# 30# 30# 34#	30# 38# 30# 30#	33# 30# 31# 30#	34# 35# 31# 35#	30# 30# 32# 38#	03# 43# 32#	

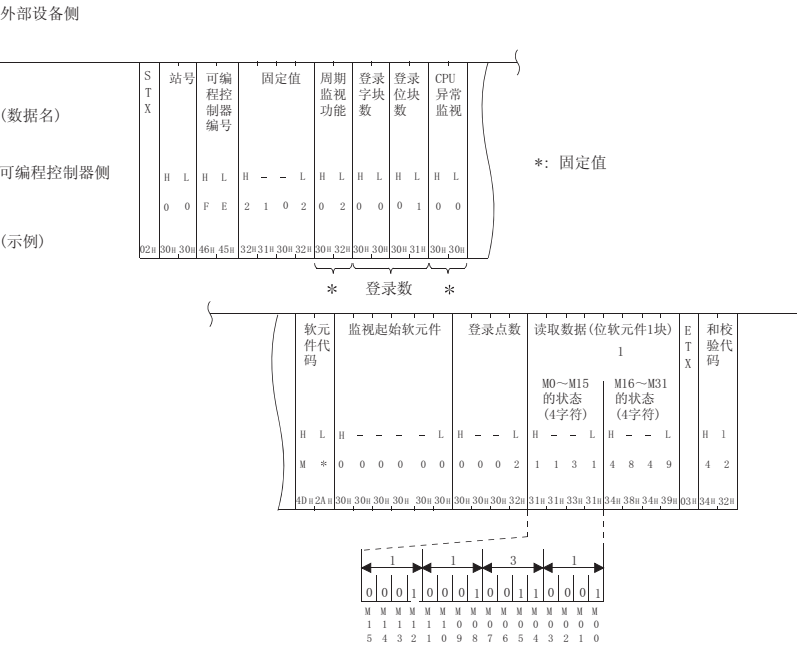
- 2) 通过二进制代码进行通信时
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”（字单位）时



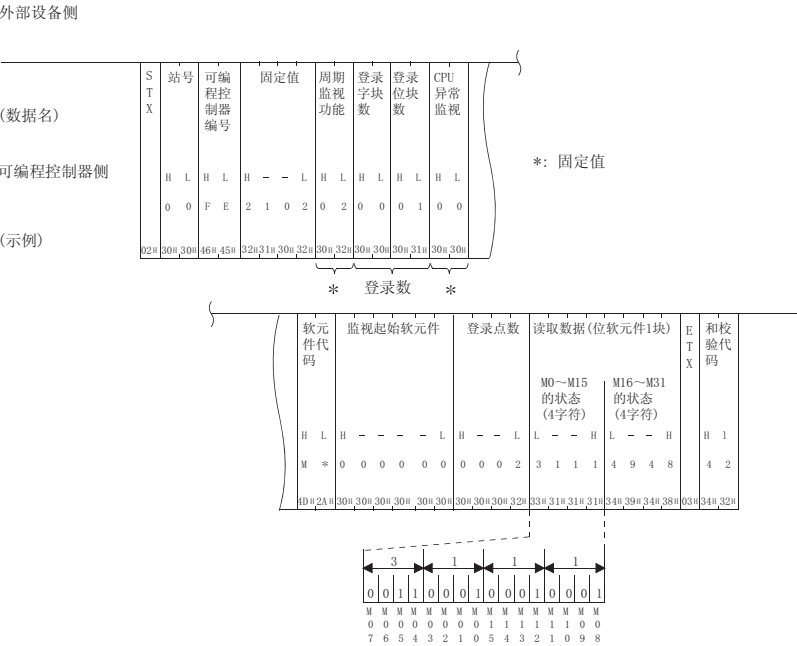
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）时



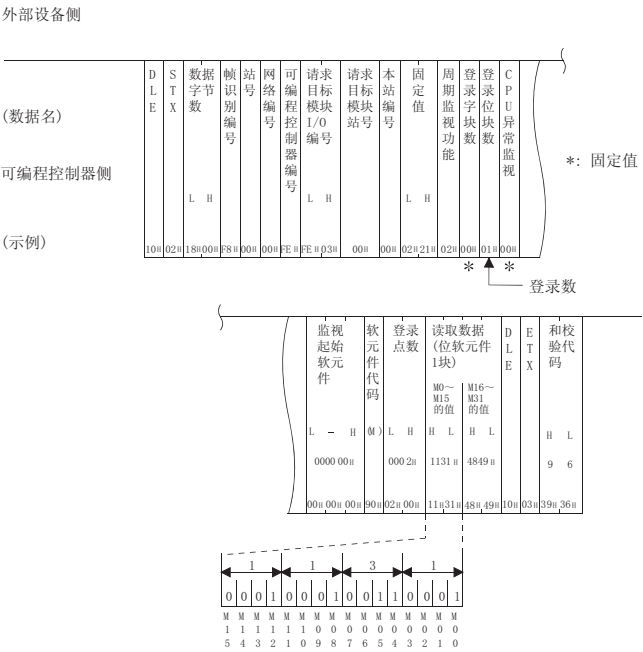
- (c) 发送位软元件数据时
- 进行了 3. 17. 2 项中所示的可编程控制器 CPU 监视登录时，由于 M0=ON 进行的位软元件数据 M0~M31 (2 点) 的发送示例如下所示。
- 1) 通过 ASCII 代码进行通信时
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0” (字单位) 时



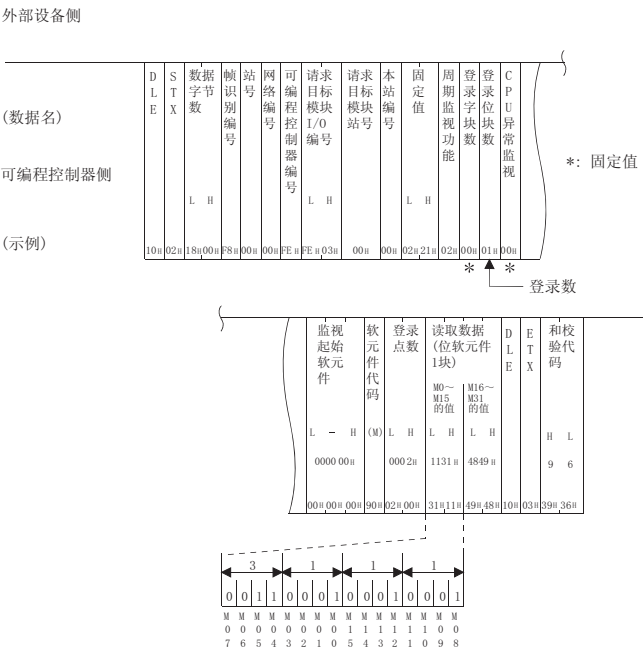
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1” (字节单位) 时



- 2) 通过二进制代码进行通信时
- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“0”（字单位）时



- GX Works2 或 GX Configurator-SC 中的“字/字节单位指定”为“1”（字节单位）时



3.18 远程口令的解锁/锁定

远程口令是用于防止来自于远程的用户对 CPU 模块进行的非法访问的功能。

设置了远程口令时，有来自于远程的用户对 CPU 模块进行访问的情况下，将进行远程口令的检查。

在本节中，对通过 MC 协议进行远程口令的解锁处理/锁定处理的指令的使用方法有关内容进行说明。

备注

- (1) 关于远程口令功能的详细内容，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。
- (2) 关于可设置远程口令的模块与远程口令的对应，请参阅下述手册。
 - 使用以太网端口内置 CPU 时
QnUCPU 用户手册(内置以太网端口通信篇)
MELSEC-L CPU 模块用户手册(内置以太网功能篇)
 - 使用 C24 时
MELSEC-Q/L 用于串行通信模块用户手册(应用篇)
 - 使用 E71 时
Q 系列以太网接口模块用户手册(基本篇)

(1) 关于远程口令检查功能的对象

对 Q/LCPU 中安装的 C24/E71 设置了远程口令时，在进行数据通信之前应通过本节中所示的指令进行远程口令的解锁处理之后再进行数据通信。

- (a) 使用 E71 的情况下
通过外部设备的请求将进行数据通信的连接设置为远程口令检查的对象的情况下，需要进行远程口令的解锁处理。
(示例) 通过 MC 协议进行的通信
通过固定缓冲进行的通信(*1)
通过随机访问用缓冲进行的通信(*1)
*1 进行通信的外部设备为 E71 时，不能进行通信。(对通信请求将返回出错应答的响应。)
- (b) 使用 C24 的情况下
从使用调制解调器功能的接口侧通过 MC 协议通信对 Q/LCPU 进行访问的情况下，应进行远程口令的解锁处理之后再进行数据通信。

备注

关于使用以太网端口内置 CPU 模块时的远程口令检查功能的对象，请参阅以下手册。

- QnUCPU 用户手册(内置以太网端口通信篇)
- MELSEC-L CPU 模块用户手册(内置以太网功能篇)

(2) 控制步骤

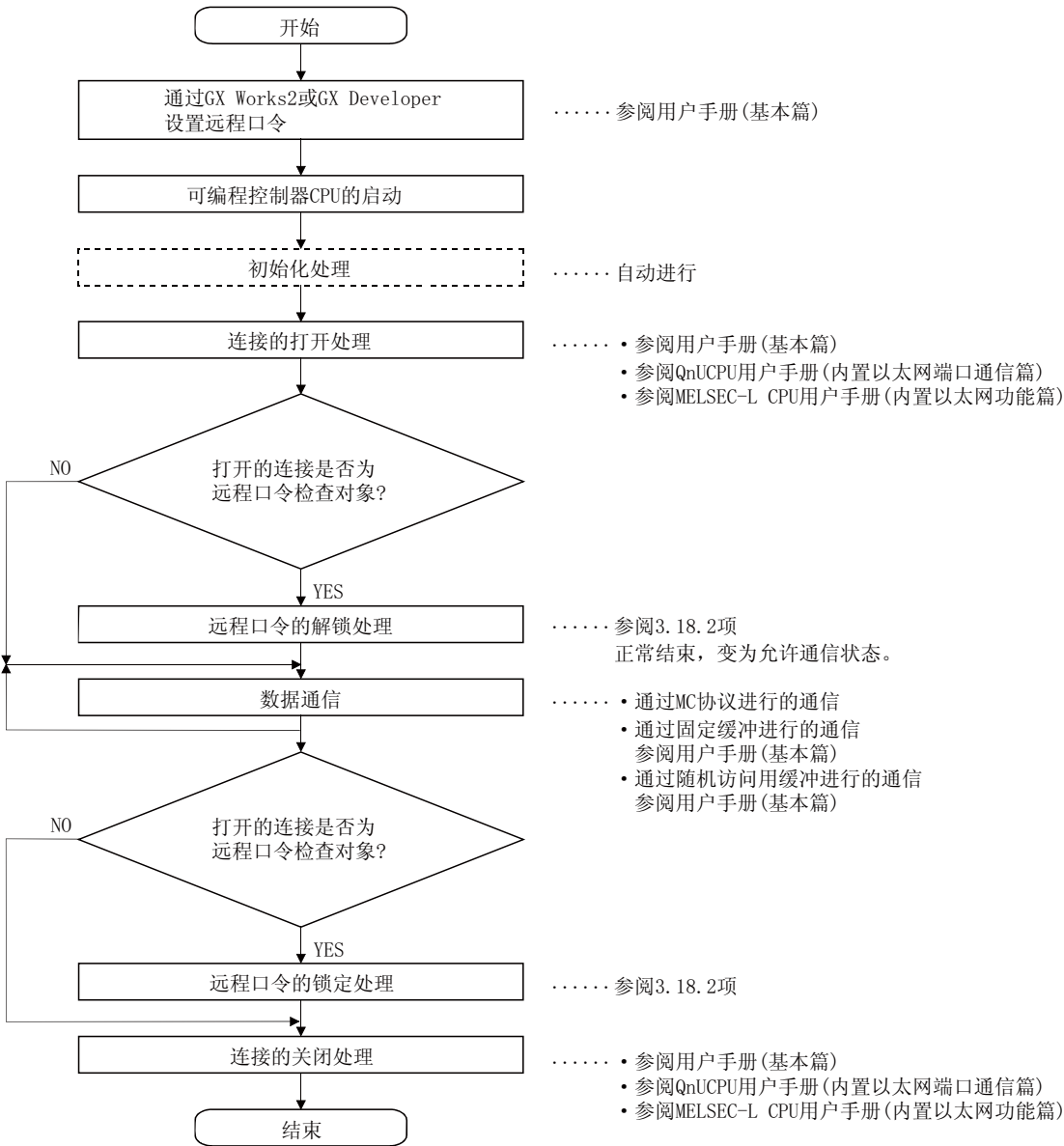
C24/E71 安装站的 Q/LCPU 中设置了远程口令时的控制步骤如下所示。

(a) 经由 E71 以及 CPU 模块的内置以太网端口进行访问的情况下

备注

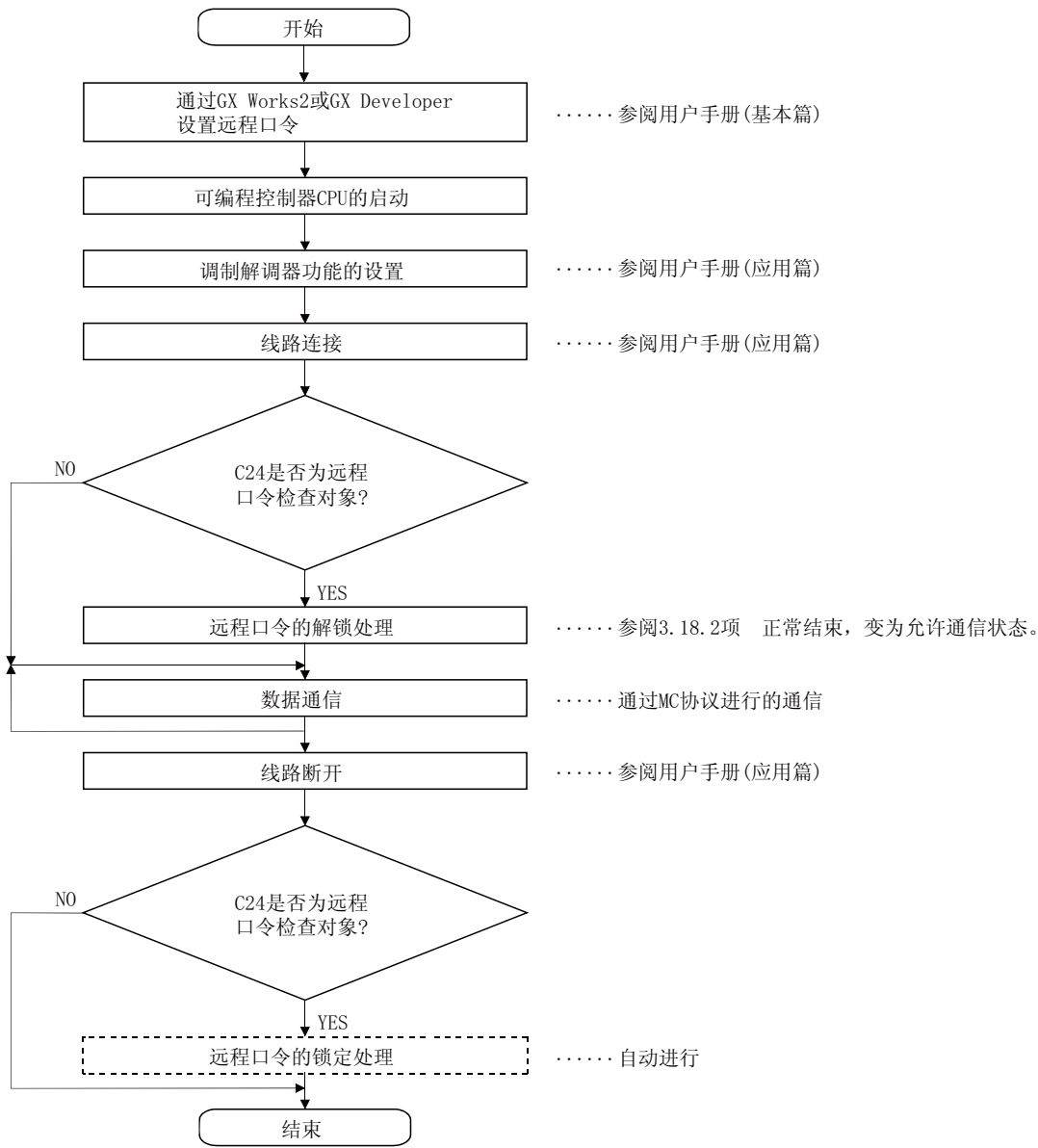
关于使用以太网端口内置 CPU 模块时的控制步骤，请参阅下述手册。

- QnUCPU 用户手册(内置以太网端口通信篇)
- MELSEC-L CPU 模块用户手册(内置以太网功能篇)



要点	
(1)	数据通信中使用的连接被设置为远程口令检查的对象的情况下，在从解锁处理完毕后开始至进行了锁定处理为止期间处于可以通信状态。
(2)	处于远程口令锁定状态时接收的指令将全部变为出错响应。应在进行了远程口令的解锁处理之后再行通信。
(3)	TCP/IP 通信时，在未进行远程口令的锁定处理的状况下进行了连接的关闭处理的情况下，关闭时将自动进行远程口令的锁定处理。(E71 自动进行。)

(b) 经由 C24 进行访问的情况下(使用调制解调器功能的接口侧为对象)



要点

- (1) 使用调制解调器功能进行数据通信的 C24 被设置为远程口令检查的对象的情况下, 从解锁处理完毕后开始至进行了线路断开为止期间处于可以通信状态。
- (2) 处于远程口令锁定状态时接收的指令将全部变为出错响应。应在进行了远程口令的解锁处理之后再行通信。
- (3) 线路断开时将自动进行远程口令的锁定处理。

3.18.1 指令及字符部分的内容

以下对从外部设备进行远程口令的解锁处理/锁定处理时的指令以及控制步骤内的字符部分(通过二进制代码进行通信时为数据部分)有关内容进行说明。

(1) 指令

功能		指令(子指令)	处理内容	可编程控制器 CPU 的状态			参阅章节
				STOP 中	RUN 中		
					写入允许 设置	写入禁止 设置	
远程口令	解锁	1630 (0000)	指定远程口令, 从锁定状态置为解锁状态。(置为可以与可编程控制器 CPU 进行通信的状态。)	○	○	○	3. 18. 2 项
	锁定	1631 (0000)	指定远程口令, 从解锁状态置为锁定状态。(置为无法与可编程控制器 CPU 进行通信的状态。)				3. 18. 2 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行。

(2) 字符部分的内容

以下对从外部设备进行远程口令的解锁处理/锁定处理时的字符部分的内容进行说明。

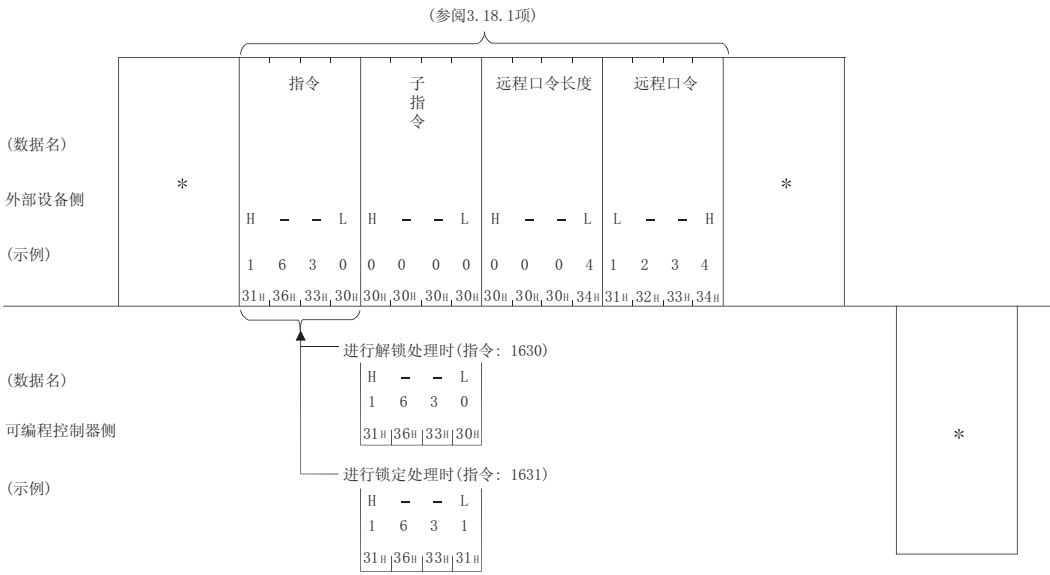
- (a) 远程口令长度
是表示远程口令的字节数(4)的数据。
- (b) 远程口令
 - 1) 是通过 GX Works2 或 GX Developer 设置到 Q/LCPU 中的远程口令。
 - 2) 通过 ASCII/二进制代码进行数据通信时, 将设置在 Q/LCPU 中的远程口令原样不变地从起始字符开始进行发送。

3.18.2 远程口令的解锁/锁定(指令：1630、1631)

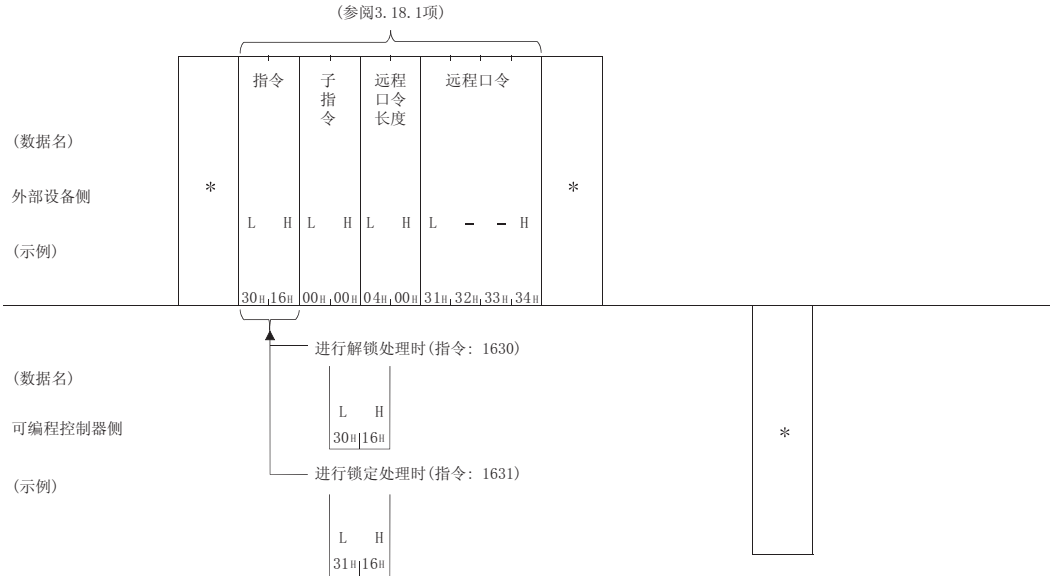
以下对进行远程口令的解锁处理/锁定处理时的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据所使用的模块、通信时的帧・格式而有所不同。
请参阅 3.1 节中所示的详细说明。

[控制步骤]

(1) 通过由 ASCII 代码进行的通信，进行解锁处理/锁定处理的情况下
设置在 Q/LCPU 中的远程口令为“1234”时



(2) 通过由二进制代码进行的通信，进行解锁处理/锁定处理的情况下
设置在 Q/LCPU 中的远程口令为“1234”时



要点

(1) 对于进行远程口令解锁处理/锁定处理时的报文格式，应指定为本站。不能对其它站 Q/LCPU 进行解锁处理/锁定处理。

(示例) 通过 QnA 兼容 4C 帧(格式 1)进行的远程口令解锁处理指令的报文示例

E N Q	帧 识 别 编 号	站 号	网 络 编 号	可 编 程 控 制 器 编 号	请 求 目 标 模 块 I/O 编 号	请 求 目 标 模 块 站 号	本 站 编 号	指 令	子 指 令	远 程 口 令 长 度	远 程 口 令
	H L F 8	H L 0 0	H L 0 0	H L F F	H - - L 0 3 F F	H L 0 0	H L 0 0	H - - L 1 6 3 0	H - - L 0 0 0 0	H - - L 0 0 0 4	H - - L 1 2 3 4
	05H46H88H	30H30H30H30H	46H46H	30H33H46H46H	30H30H30H30H	31H36H33H30H	30H30H60H30H	30H30H34H	31H62H33H34H		

(2) 用户无需对 C24 进行远程口令的锁定处理。通过断开调制解调器的线路即可自动进行锁定处理。

4 通过 QnA 兼容 2C 帧进行通信时

在本章中，对以 QnA 兼容 2C 帧进行至 C24 的 MC 协议的数据通信时的报文的数据格式、数据指定方法、限制等有关内容进行说明。
* QnA 兼容 2C 帧与 QnA 系列串行通信模块的 QnA 简易帧为相同的报文格式。

要点	
使用 E71 时，无需阅读本章。	

4.1 控制步骤、报文格式

以 QnA 兼容 2C 帧进行数据通信时，各格式的控制步骤、报文格式如下所示。

(1) 数据通信的基本格式

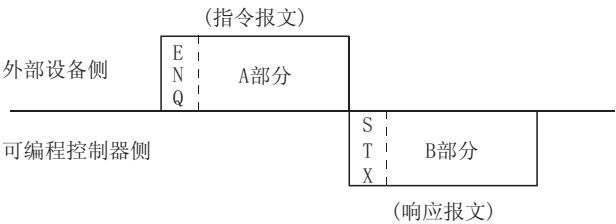
作为外部设备用于通过 MC 协议访问可编程控制器的控制步骤(指令报文·响应报文的构成及发送接收步骤)有 4 种格式。
根据使用的格式，通过在 GX Works2 或 GX Developer 中将 C24 的对象接口的通信协议设置设置为“1”~“4”，可以以指定格式进行数据通信。
无论哪种格式均是通过 ASCII 代码数据进行通信。
以格式 1 为基准 4 种格式的区别如下所示。

- 格式 2 : 是各报文中附加了块编号的格式
- 格式 3 : 是将各报文用 STX、ETX 围住的格式
- 格式 4 : 是各报文中附加了 CR、LF 的格式

(2) MC 协议控制步骤的阅读方法

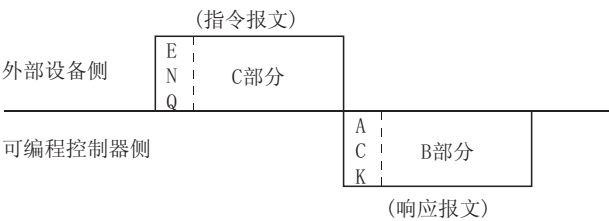
以下对各控制步骤格式的说明中所示的传送数据的阅读方法进行说明。

(a) 外部设备通过可编程控制器进行数据读取的情况下



- 1) A 部分表示由外部设备至 C24 的传送。
- 2) B 部分表示由 C24 至外部设备的传送。
- 3) 创建外部设备的程序时应使各数据按从左至右的顺序进行传送。
(示例：A 部分的情况下，按从 ENQ 至右的顺序进行数据发送。)

(b) 从外部设备对可编程控制器进行数据写入的情况下



- 1) C 部分表示由外部设备至 C24 的传送。
- 2) B 部分表示由 C24 至外部设备的传送。
- 3) 创建外部设备的程序时应使各数据按从左至右的顺序进行传送。
(示例：C 部分的情况下，按从 ENQ 至右的顺序进行数据发送。)

要点

从外部设备接收指令报文时，C24 对报文中 A 部分的处理完毕后，发送响应报文并变为待机状态。
C24 处于待机状态时，对下一个指令报文的接收以及来自于可编程控制器 CPU 的接通请求数据的发送请求进行等待。

(3) 控制步骤

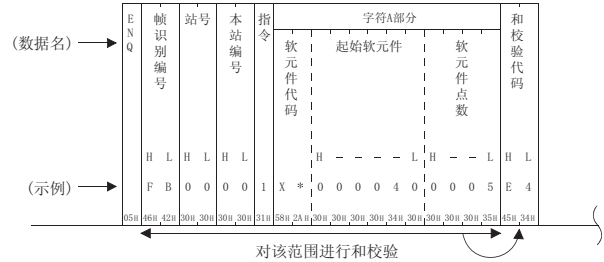
(a) 以格式 1 进行通信时

1) 外部设备从 Q/LCPU 中进行数据读取的情况下

(示例)

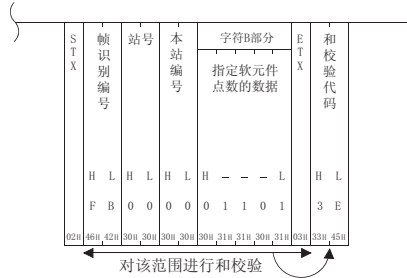
外部设备通过指令 1 从 Q/LCPU 的 X40 ~ X44 中读取 5 点的的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

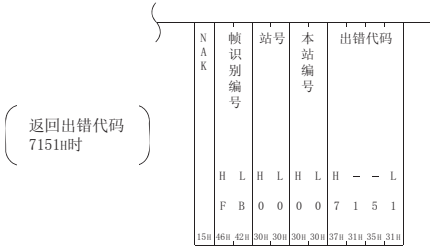


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)

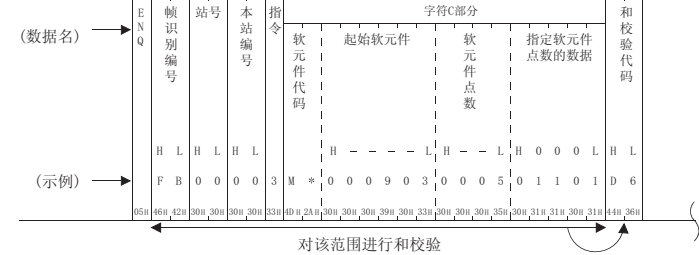


2) 外部设备对 Q/LCPU 进行数据写入的情况下

(示例)

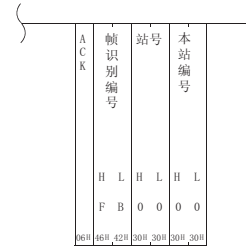
外部设备通过指令 3 向 Q/LCPU 的 M903 ~ M907 写入 5 点的的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

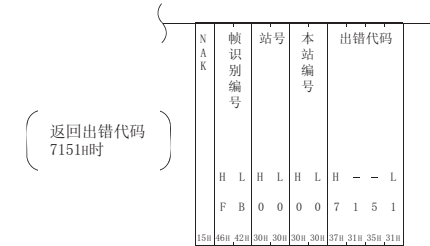


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)



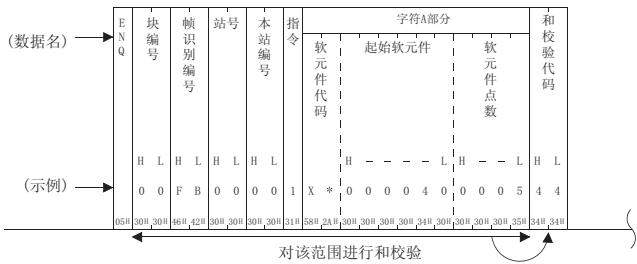
(b) 以格式 2 进行通信时

1) 外部设备从 Q/LCPU 中进行数据读取的情况下

(示例)

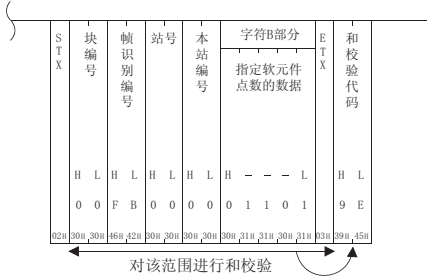
外部设备通过指令 1 从 Q/LCPU 的 X40 ~ X44 中读取 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

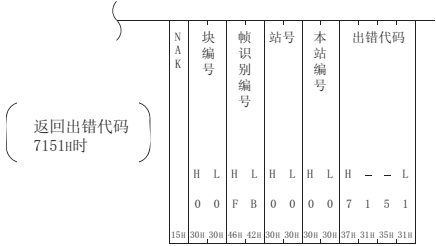


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)

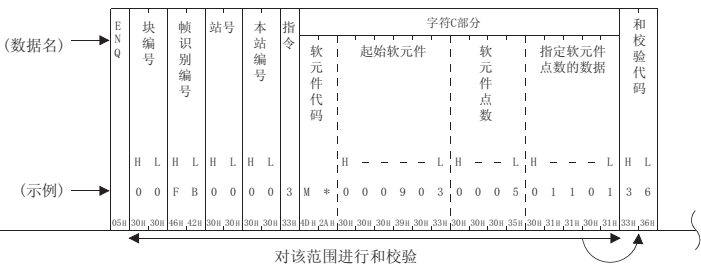


2) 外部设备对 Q/LCPU 进行数据写入的情况下

(示例)

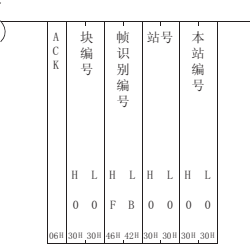
外部设备通过指令 3 向 Q/LCPU 的 M903 ~ M907 写入 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

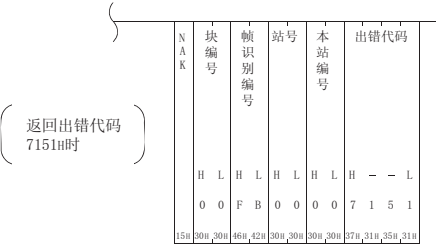


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)



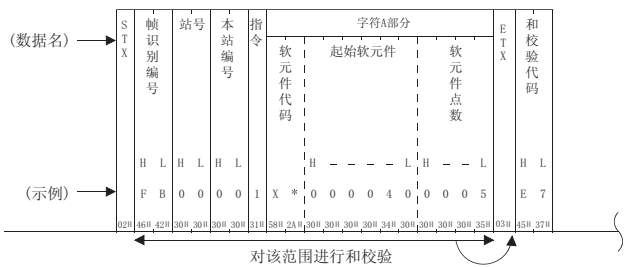
(c) 以格式 3 进行通信时

1) 外部设备从 Q/LCPU 中进行数据读取的情况下

(示例)

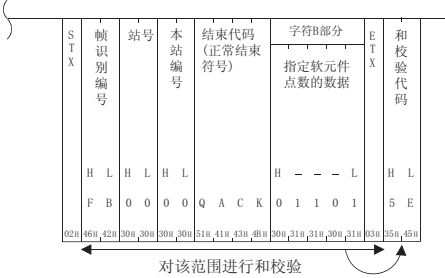
外部设备通过指令 1 从 Q/LCPU 的 X40 ~ X44 读取 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

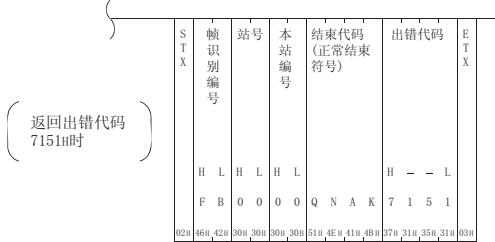


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)

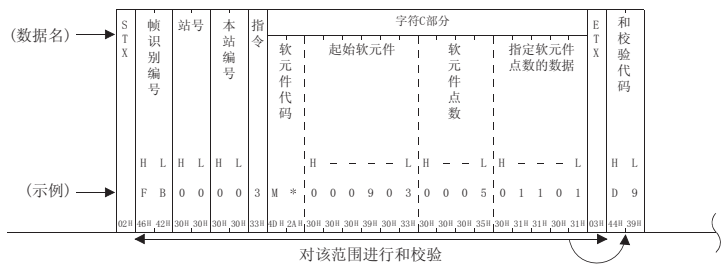


2) 外部设备对 Q/LCPU 进行数据写入的情况下

(示例)

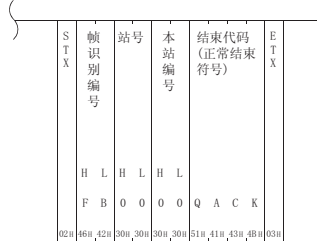
外部设备通过指令 3 向 Q/LCPU 的 M903 ~ M907 写入 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

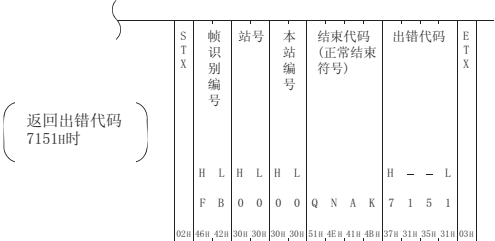


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)



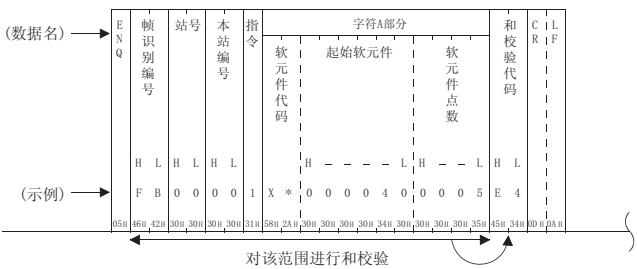
(d) 以格式 4 进行通信时

1) 外部设备从 Q/LCPU 中进行数据读取的情况下

(示例)

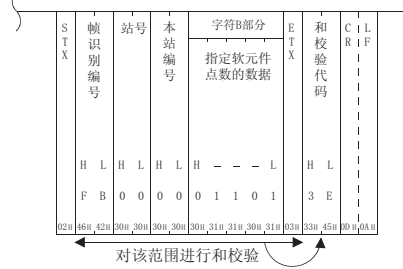
外部设备通过指令 1 从 Q/LCPU 的 X40 ~ X44 中读取 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

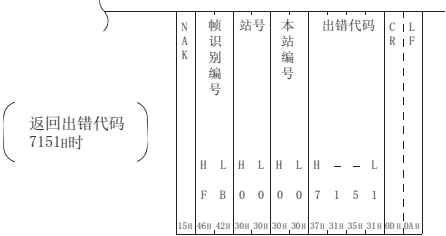


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)

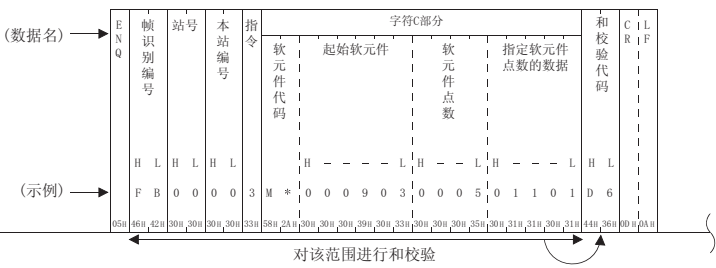


2) 外部设备对 Q/LCPU 进行数据写入的情况下

(示例)

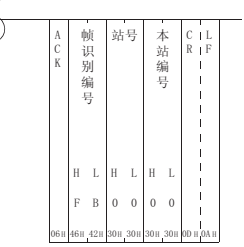
外部设备通过指令 3 向 Q/LCPU 的 M903 ~ M907 写入 5 点的数据的情况下

外部设备侧→可编程控制器侧(指令报文)

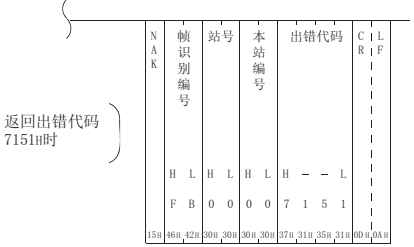


可编程控制器侧→外部设备侧(指令报文)

(正常结束时)



(异常结束时)



4.2 数据指定项目的内容

以下对 QnA 兼容 2C 帧各格式的控制步骤内指定的各数据名的指定内容中，与 QnA 兼容 3C/4C 帧不相同的 QnA 兼容 2C 帧专用的数据项目的内容进行说明。

要点
关于本节中所示的数据项目以外，与以 QnA 兼容 3C/4C 帧进行通信时的指定方法相同。请参阅 3.1.6 项。
关于用于数据通信的“报文等待”时间，请参阅 3.1.6 项(11)下的备注。

(1) 帧识别编号

以 QnA 兼容 2C 帧进行通信时，将“FB”用作帧识别编号。

(2) 指令

将所使用功能对应的 QnA 兼容 2C 帧用指令“1~9”(参阅 4.3 节)之一，通过 ASCII 代码 1 位进行发送。

(3) 字符部分(A 部分、B 部分、C 部分)

字符部分与以 QnA 兼容 3C/4C 帧进行通信时的指定方法·内容相同，取決与从外部设备发送的指令。

QnA 兼容 3C/4C 帧用指令与 QnA 兼容 2C 帧用指令的字符部分的指定方法·内容相同，其对应关系如下所示。

指令		QnA 兼容 2C 帧的指令	与如左所示指令对应的 QnA 兼容 3C/4C 帧的指令	
			指令	子指令
批量读取	位单位	1	0401	0001
	字单位	2	0401	0000
批量写入	位单位	3	1401	0001
	字单位	4	1401	0000
随机读取	字单位	5	0403	0000
测试(随机写入)	位单位	6	1402	0001
	字单位	7	1402	0000
监视数据登录	字单位	8	0801	0000
监视	字单位	9	0802	0000

应根据 3.3 节，对所使用的指令用的字符部分进行指定。

4.3 QnA 兼容 2C 帧用指令及功能列表

QnA 兼容 2C 帧通信的指令及功能等如下所示。

用于通过 MC 协议对 C24 安装站可编程控制器 CPU 的软元件存储器进行数据读取、写入的指令如下表所示。

功能		指令	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态(*1)		
					STOP 中	RUN 中	
						写入允许设置	写入禁止设置
批量读取	位单位	1	对位软元件以 1 点为单位进行读取。 (1 点=1 位)	7904 点	○	○	○
	字单位	2	对位软元件以 1 点为单位进行读取。 (1 点=16 位)				
			对字软元件以 1 点为单位进行读取。				
批量写入(*2)	位单位	3	向位软元件以 1 点单位进行写入。 (1 点=1 位)	7904 点	○	○	×
	字单位	4	向位软元件以 1 点单位进行写入。 (1 点=16 位)	960 点			
			向字软元件以 1 点单位进行写入。				
随机读取	字单位	5	将位软元件以 1 点为单位，随机指定软元件、 软元件编号进行读取。(1 点=16 位)	192 点	○	○	○
			将字软元件以 1 点为单位，随机指定软元件、 软元件编号进行读取。				
测试(*2) [随机写入]	位单位	6	将位软元件以 1 点为单位，随机指定软元件、 软元件编号进行写入。(1 点=1 位)	188 点	○	○	×
	字单位	7	将位软元件以 1 点为单位，随机指定软元件、 软元件编号进行写入。(1 点=16 位)	1920 点(*4)			
			将字软元件以 1 点为单位，随机指定软元件、 软元件编号进行写入。				
监视数据登录(*3)	字单位	8	将要监视的位软元件以 1 点单位进行登录。 (1 点=16 位)	192 点	○	○	○
			将要监视的字软元件以 1 点单位进行登录。				
监视(*3)	字单位	9	对进行了监视数据登录的软元件进行监视。	按登录点数			

*1 对可编程控制器 CPU 的运行中写入的可否设置是在 GX Works2 或 GX Developer 中的传送设置中进行设置。

*2 执行指令的 Q/LCPU 中被实施了系统保护时，将变为出错状态并返回异常响应。

*3 进行监视时的步骤与以 QnA 兼容 3C/4C 帧进行通信时的监视步骤相同。

*4 1 次通信中可写入的点数(处理点数)应在以下范围内进行指定。
 $1920 \geq (\text{字访问点数} \times 12) + (\text{双字访问点数} \times 14) \geq 1$

4.4 数据通信时的注意事项

通过 QnA 兼容 2C 帧进行数据通信时的注意事项如下所示。

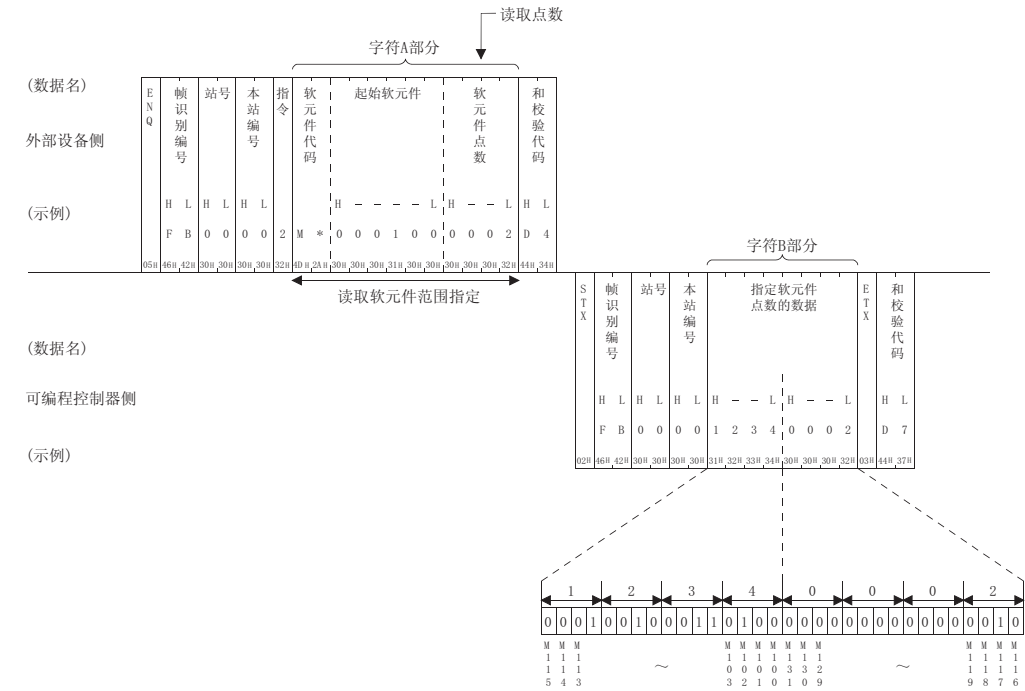
- (1) 各格式的的注意事项与以 QnA 兼容 3C/4C 帧进行通信时相同。请参阅 3.1.4 项 ~ 3.1.5 项。
- (2) 对于在以 QnA 兼容 3C/4C 帧通信中可指定的读取数据时的监视条件，在以 QnA 兼容 2C 帧进行的通信中不能指定。
此外，也不能进行附录 1 中所示的读取、写入的软件存储器的扩展设置。
- (3) 对于以 QnA 兼容 2C 帧进行数据通信时的各指令的读取、写入点数及范围，与使用对应于 QnA 兼容 2C 帧指令的 QnA 兼容 3C/4C 帧指令时相同。(字符部分的指定方法 · 内容相同)
关于对应于 QnA 兼容 2C 帧指令的 QnA 兼容 3C/4C 帧指令，请参阅本章的 4.2 节。
- (4) 可以对与外部设备相连接的 C24 安装站(多点连接的 Q/QnACPU 站也包含在内)进行访问。

4.5 通过 QnA 兼容 2C 帧进行数据通信的示例

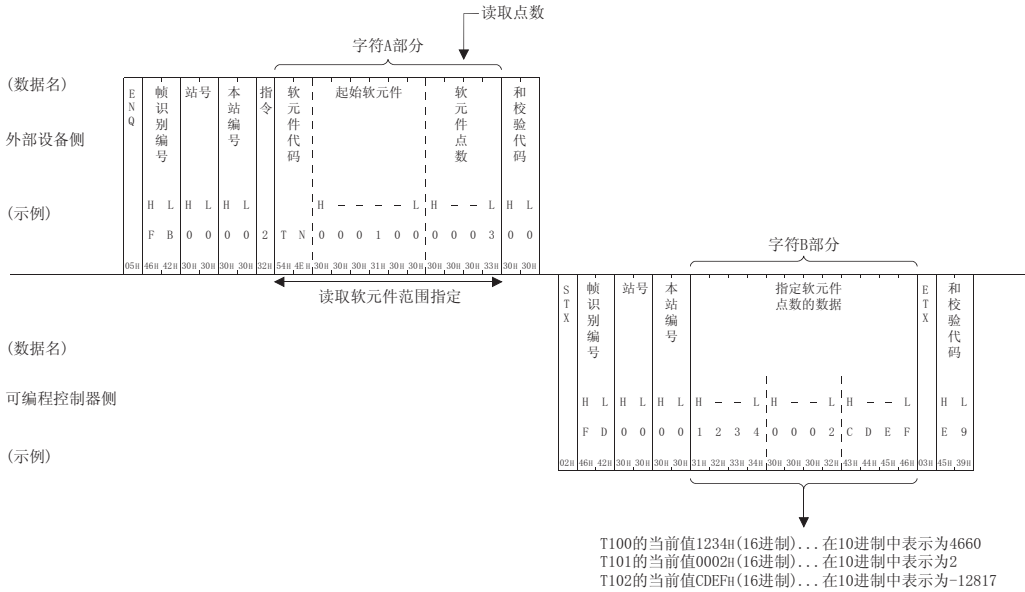
通过 QnA 兼容 2C 帧格式 1 进行数据通信的控制步骤示例如下所示。
关于指令 1、3 的控制步骤示例，请参阅 4.1 节(3)。

(1) 字单位的批量读取(指令：2)

(a) 对内部继电器 M100 ~ M131 的 2 点(32 位)进行读取的情况下



(b) 对定时器 T100 ~ T102 的当前值读取 3 点的情况下



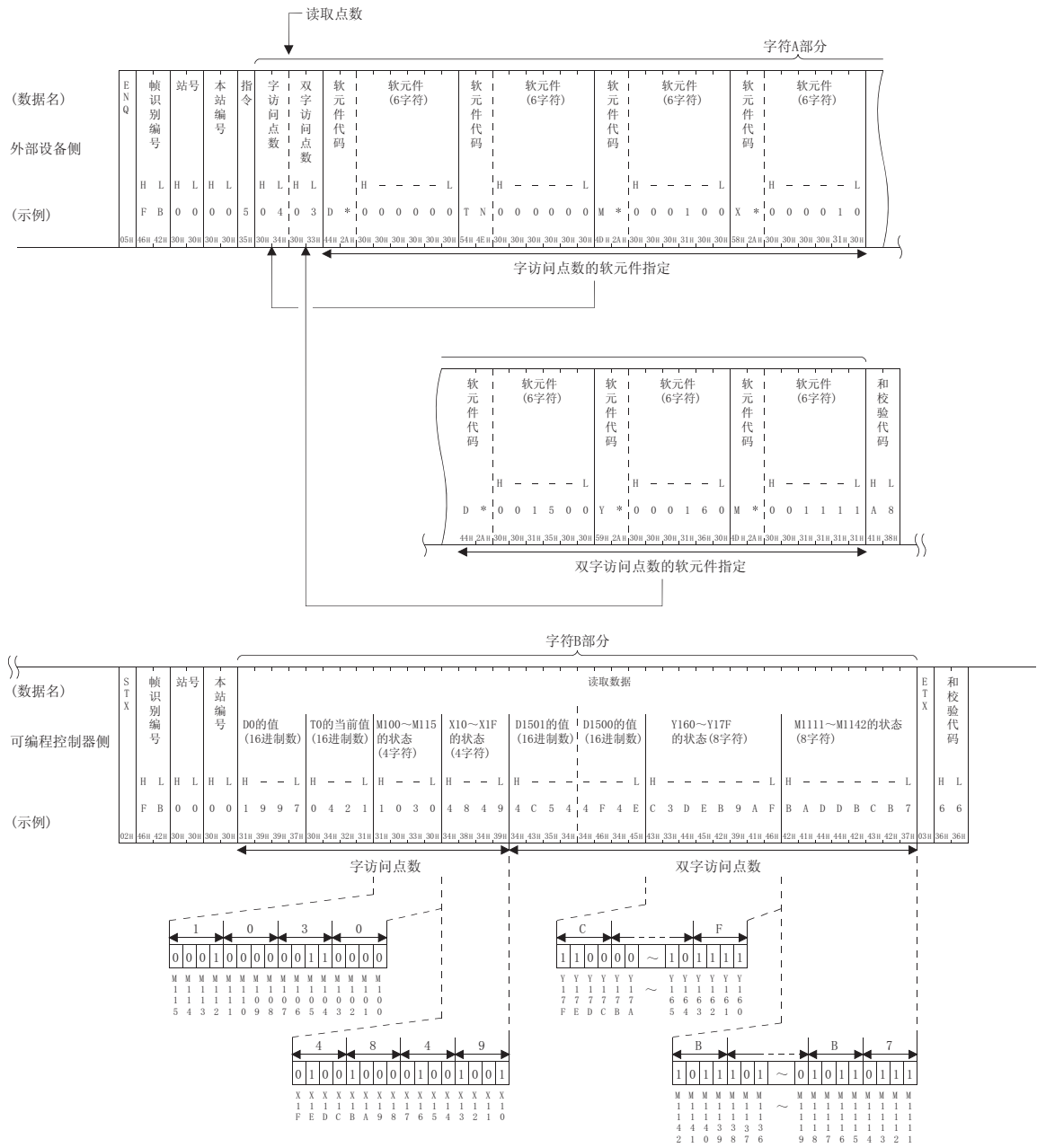
T100的当前值1234h(16进制)...在10进制中表示为4660
T101的当前值0002h(16进制)...在10进制中表示为2
T102的当前值CDEFh(16进制)...在10进制中表示为-12817

* 关于指令 2 的字符部分的指定方法·内容，与 QnA 兼容 3C/4C 帧的指令 0401(子指令 0000)相同。

(3) 字单位的随机读取(指令：5)

对以下软元件存储器进行读取的情况下

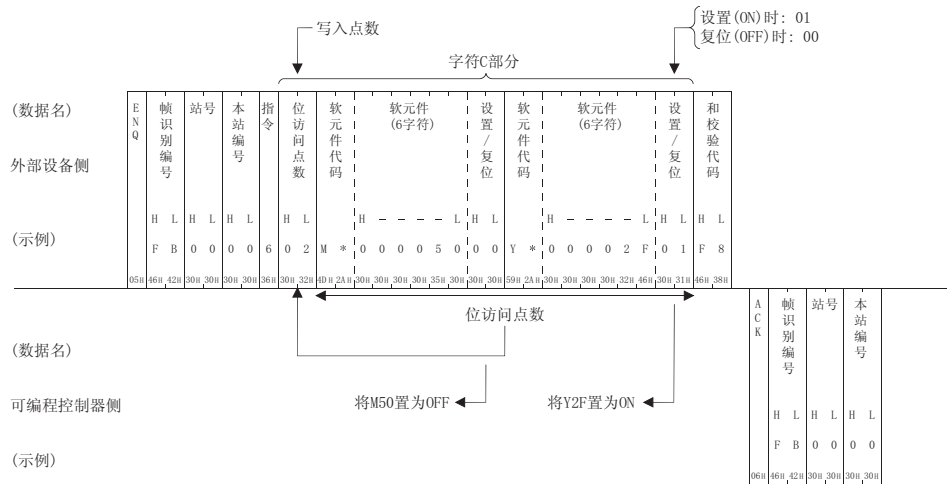
- 字访问 : D0、T0、M100 ~ M115、X10 ~ X1F
- 双字访问 : D1500 ~ D1501、Y160 ~ Y17F、M1111 ~ M1142



* 关于指令5的字符部分的指定方法·内容，与QnA兼容3C/4C帧的指令0403(子指令0000)相同。

(4) 位单位的测试(指令: 6)

将内部继电器 M50 置为 OFF，将输出继电器 Y2F 置为 ON 的情况下

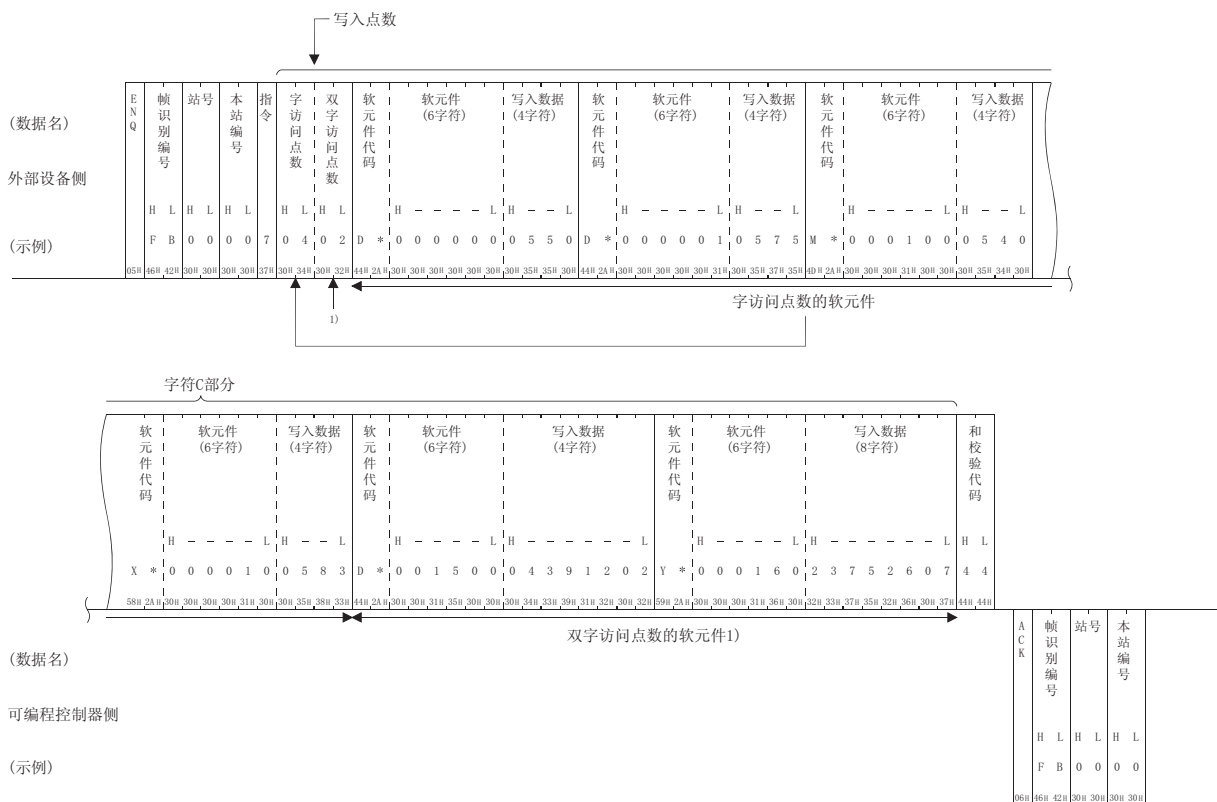


* 关于指令 6 的字符部分的指定方法·内容，与 QnA 兼容 3C/4C 帧的指令 1402(子指令 0001)相同。

(5) 字单位的测试(指令: 7)

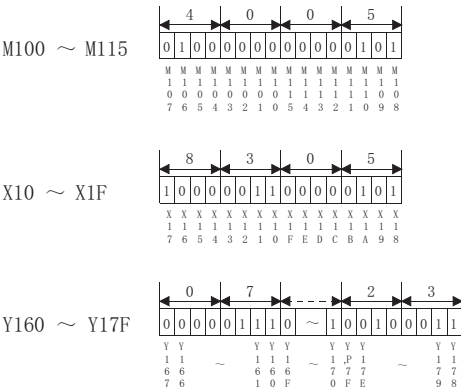
写入到以下软元件存储器中的情况下

- 字访问 : D0、D1、M100 ~ M115、X10 ~ X1F
- 双字访问 : D1500 ~ D1501、Y160 ~ Y17F



* 关于指令 7 的字符部分的指定方法·内容,与 QnA 兼容 3C/4C 帧的指令 1402(子指令 0000)相同。

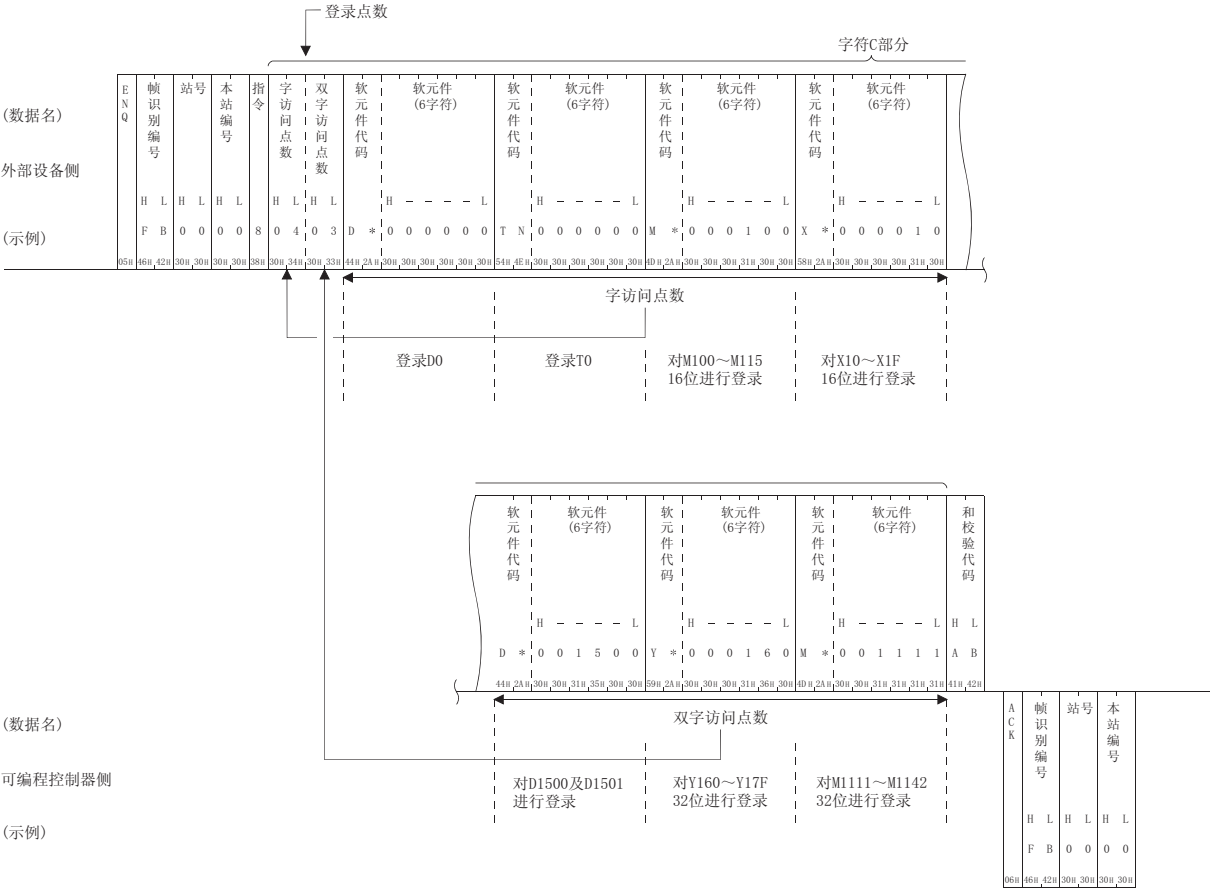
通过字单位测试进行测试的数据与位软元件的对应如下所示。



(6) 字单位的监视数据登录(指令：8)

对以下软元件存储器进行监视数据登录的情况下

- 字访问 : D0、T0、M100 ~ M115、X10 ~ X1F
- 双字访问 : D1500 ~ D1501、Y160 ~ Y17F、M1111 ~ M1142



* 关于指令 8 的字符部分的指定方法·内容，与 QnA 兼容 3C/4C 帧的指令 0801(子指令 0000)相同。

(7) 字单位的监视(指令：9)

对以下进行了监视数据登录的软元件存储器进行监视的情况下
(进行了监视数据登录的软元件存储器)

- 字访问 : D0、T0、M100 ~ M115、X10 ~ X1F
- 双字访问 : D1500 ~ D1501、Y160 ~ Y17F、M1111 ~ M1142

(数据名)	ENQ	帧识别编号	站号	本站编号	指令	和校验代码
外部设备侧						
(示例)		H L	H L	H L	H L	H L
		F B	0 0	0 0	9 8	1
	05H	46H, 42H	30H, 30H	30H, 30H	39H	38H, 31H

(数据名)	STX	帧识别编号	站号	本站编号	监视数据													
					D0的值 (16进制数)	T0的当前值 (16进制数)	M100~M115 的状态	X10~X1F 的状态	D1501的值 (16进制数) ¹	D1500的值 (16进制数) ¹								
可编程控制器侧		H L	H L	H L	H - - - L	H - - - L	H - - - L	H - - - L	H - - - L	H - - - L								
(示例)		F B	0 0	0 0	2 3 7 5	2 6 0 7	B C C F	B D E 9	0 5 5 0	0 5 7 5								
	02H	46H, 42H	30H, 30H	30H, 30H	32H, 33H, 37H, 35H	32H, 36H, 30H, 37H	42H, 43H, 43H, 46H	42H, 44H, 45H, 39H	30H, 35H, 35H, 30H	30H, 35H, 37H, 35H								

监视数据												STX	和校验代码
Y160~Y17F 的状态						M1111~M1142 的状态						H L	
H - - - - - L						H - - - - - L						H L	
0 5 4 0 0 5 8 3						0 4 4 3 0 4 3 8						C B	
30H, 35H, 34H, 30H, 30H, 35H, 38H, 23H												03H	43H, 42H

进行了监视数据登录的位软元件与读取的数据的对应如下所示。

M100 ~ M115	<table><tr><td colspan="4">B</td><td colspan="4">C</td><td colspan="4">C</td><td colspan="4">F</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td></td></tr><tr><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td></tr><tr><td>5</td><td></td><td>2</td><td>1</td><td></td><td>8</td><td>7</td><td></td><td>4</td><td>3</td><td></td><td>0</td><td></td><td></td><td>0</td><td></td></tr></table>	B				C				C				F				1	0	1	1	1	0	0	1	1	0	0	1	1	1	1	1	M		M	M		M	M		M	M		M	M		M		1	~	1	1	~	1	1	~	1	1	~	1	1	~	1	1	5		2	1		8	7		4	3		0			0																																																							
B				C				C				F																																																																																																																											
1	0	1	1	1	0	0	1	1	0	0	1	1	1	1	1																																																																																																																								
M		M	M		M	M		M	M		M	M		M																																																																																																																									
1	~	1	1	~	1	1	~	1	1	~	1	1	~	1	1																																																																																																																								
5		2	1		8	7		4	3		0			0																																																																																																																									
X10 ~ X1F	<table><tr><td colspan="4">B</td><td colspan="4">D</td><td colspan="4">E</td><td colspan="4">9</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>X</td><td></td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td></td><td>X</td><td></td></tr><tr><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td><td>~</td><td>1</td><td>1</td></tr><tr><td>F</td><td></td><td>C</td><td>B</td><td></td><td>8</td><td>7</td><td></td><td>4</td><td>3</td><td></td><td>0</td><td></td><td></td><td>0</td><td></td></tr></table>	B				D				E				9				1	0	1	1	1	0	1	1	0	1	1	0	1	0	0	1	X		X	X		X	X		X	X		X	X		X		1	~	1	1	~	1	1	~	1	1	~	1	1	~	1	1	F		C	B		8	7		4	3		0			0																																																							
B				D				E				9																																																																																																																											
1	0	1	1	1	0	1	1	0	1	1	0	1	0	0	1																																																																																																																								
X		X	X		X	X		X	X		X	X		X																																																																																																																									
1	~	1	1	~	1	1	~	1	1	~	1	1	~	1	1																																																																																																																								
F		C	B		8	7		4	3		0			0																																																																																																																									
Y160 ~ Y17F	<table><tr><td colspan="4">0</td><td colspan="4">5</td><td colspan="4">---</td><td colspan="4">8</td><td colspan="4">3</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>~</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>X</td><td>X</td><td></td><td></td><td></td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td>X</td><td>X</td><td></td><td>X</td><td>X</td></tr><tr><td>1</td><td>1</td><td></td><td>~</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td></tr><tr><td>7</td><td>7</td><td></td><td></td><td></td><td>7</td><td>7</td><td>7</td><td></td><td>6</td><td>6</td><td>6</td><td></td><td>~</td><td></td><td>6</td><td>6</td><td></td><td>6</td></tr><tr><td>F</td><td>E</td><td></td><td></td><td></td><td>9</td><td>8</td><td>7</td><td></td><td>8</td><td>7</td><td>6</td><td></td><td></td><td></td><td>7</td><td>6</td><td></td><td>1</td></tr></table>	0				5				---				8				3				0	0	0	0	0	1	0	1	0	0	~	1	1	0	0	0	0	1	1	X	X				X	X	X		X	X	X		X	X	X		X	X	1	1		~		1	1	1		1	1	1		1	1	1		1	1	7	7				7	7	7		6	6	6		~		6	6		6	F	E				9	8	7		8	7	6				7	6		1																			
0				5				---				8				3																																																																																																																							
0	0	0	0	0	1	0	1	0	0	~	1	1	0	0	0	0	1	1																																																																																																																					
X	X				X	X	X		X	X	X		X	X	X		X	X																																																																																																																					
1	1		~		1	1	1		1	1	1		1	1	1		1	1																																																																																																																					
7	7				7	7	7		6	6	6		~		6	6		6																																																																																																																					
F	E				9	8	7		8	7	6				7	6		1																																																																																																																					
M1111 ~ M1142	<table><tr><td colspan="4">0</td><td colspan="4">4</td><td colspan="4">---</td><td colspan="4">3</td><td colspan="4">8</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>~</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>M</td><td></td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td>M</td><td></td><td>M</td><td></td><td>M</td></tr><tr><td>1</td><td>1</td><td></td><td>~</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td></tr><tr><td>4</td><td>4</td><td></td><td></td><td></td><td>3</td><td>3</td><td>3</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td><td>1</td><td></td><td>1</td><td>1</td></tr><tr><td>2</td><td>1</td><td></td><td></td><td></td><td>6</td><td>5</td><td>4</td><td></td><td>9</td><td>8</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td>2</td><td></td></tr></table>	0				4				---				3				8				0	0	0	0	0	1	0	0	0	0	~	0	0	0	1	1	0	0	0	M		M		M	M		M	M		M	M		M	M		M		M	1	1		~		1	1	1		1	1	1		1	1	1		1	1	1	1				1	1	1		1	1	1		1	1	1		1	1	4	4				3	3	3		1	1	1		1	1	1		1	1	2	1				6	5	4		9	8	7						2	
0				4				---				3				8																																																																																																																							
0	0	0	0	0	1	0	0	0	0	~	0	0	0	1	1	0	0	0																																																																																																																					
M		M		M	M		M	M		M	M		M	M		M		M																																																																																																																					
1	1		~		1	1	1		1	1	1		1	1	1		1	1																																																																																																																					
1	1				1	1	1		1	1	1		1	1	1		1	1																																																																																																																					
4	4				3	3	3		1	1	1		1	1	1		1	1																																																																																																																					
2	1				6	5	4		9	8	7						2																																																																																																																						

* 关于指令 9 的字符部分的内容，与 QnA 兼容 3C/4C 帧的指令 0802(子指令 0000)相同。

5 通过 A 兼容 1C 帧进行通信时

在本章中，对 C24 以 A 兼容 1C 帧进行 MC 协议数据通信时的报文数据格式、数据指定方法、限制等有关内容进行说明。

要点	
使用 E71 时，无需阅读本章。	

5.1 控制步骤、报文格式

以 A 兼容 1C 帧进行数据通信时的控制步骤及各指令的报文格式如下所示。
对于以 A 兼容 1C 帧进行的数据通信，与通过 A 系列计算机链接模块支持的专用协议进行的通信功能相同，只能使用 5.1.5 项中所示的指令。无论哪种均通过 ASCII 代码数据进行通信。

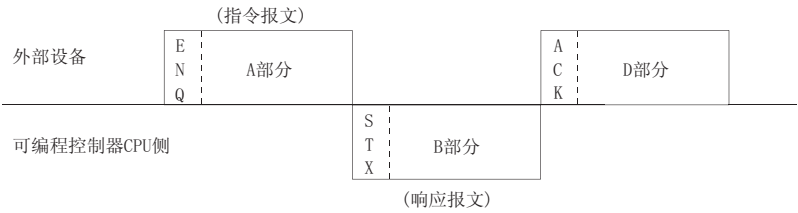
5.1.1 控制步骤、指令说明项的阅读方法

5.2 节～5.4 节中所示的各指令的控制步骤以及说明项的报文说明图的阅读方法如下所示。

(1) MC 协议控制步骤的阅读方法

以下对 5.2 节以后的各控制步骤格式的说明中所示的传送数据的阅读方法进行说明。

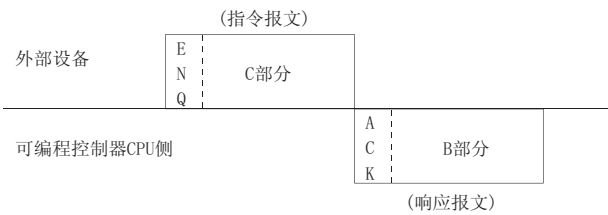
(a) 外部设备通过可编程控制器进行数据读取的情况下



- 1) A 部分表示由外部设备至 C24 的传送。
- 2) B 部分表示由 C24 至外部设备的传送。
- 3) 创建外部设备的程序时应使各数据按从左至右的顺序进行传送。
(示例：A 部分的情况下，按从 ENQ 至右的顺序进行数据发送。)

要点	
外部设备通过可编程控制器 CPU 进行数据读取时，对于上图的“D 部分”无需由外部设备进行发送。(可以省略。) C24 将上图的“B 部分”发送至外部设备中，在对读取请求“A 部分”的处理完毕后将等待下一个“A 部分”、下述数据写入请求的“C 部分”的接收。(待机状态) 因此，C24 即使从外部设备接收了数据通信最后的“D 部分”也不进行任何处理。	

(b) 从外部设备对可编程控制器进行数据写入的情况下



- 1) C 部分表示由外部设备至 C24 的传送。
- 2) B 部分表示由 C24 至外部设备的传送。
- 3) 创建外部设备的程序时应使各数据按从左至右的顺序进行传送。
(示例: C 部分的情况下, 按从 ENQ 至右的顺序进行数据发送。)

要点	
从外部设备接收指令报文时, C24 对报文中 A 部分/C 部分的处理完毕后, 发送响应报文并变为待机状态。 C24 处于待机状态时, 对下一个指令报文的接收以及来自于可编程控制器 CPU 的接通请求数据的发送请求进行等待。	

(2) 数据通信的基本格式

作为外部设备用于通过 MC 协议用的 A 兼容 1C 帧访问可编程控制器的控制步骤(指令报文・响应报文的构成及发送接收步骤)有 4 种格式。
根据使用的格式, 通过在 GX Works2 或 GX Developer 中将 C24 的对象接口的通信协议设置设置为“1”~“4”, 可以以指定格式进行数据通信。
以格式 1 为基准 4 种格式的区别如下所示。

- 格式 2 : 是各报文中附加了块编号的格式
- 格式 3 : 是将各报文用 STX、ETX 围住的格式
- 格式 4 : 是各报文中附加了 CR、LF 的格式

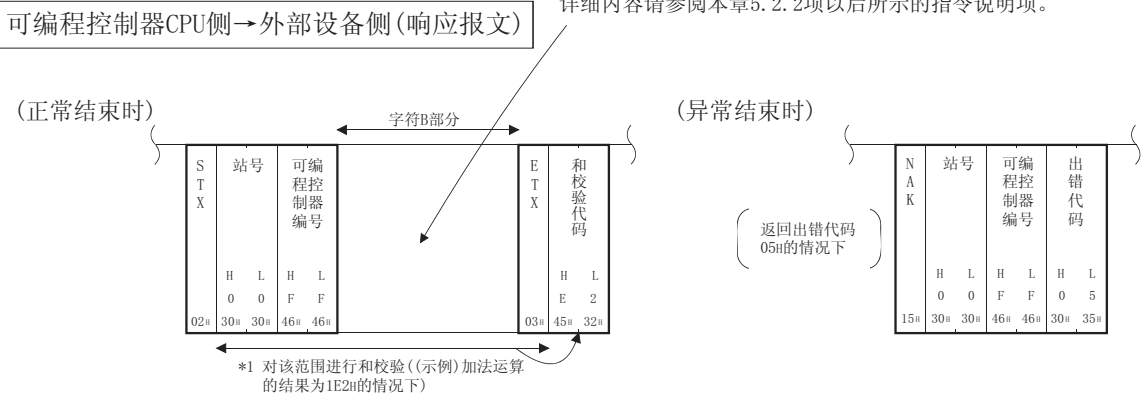
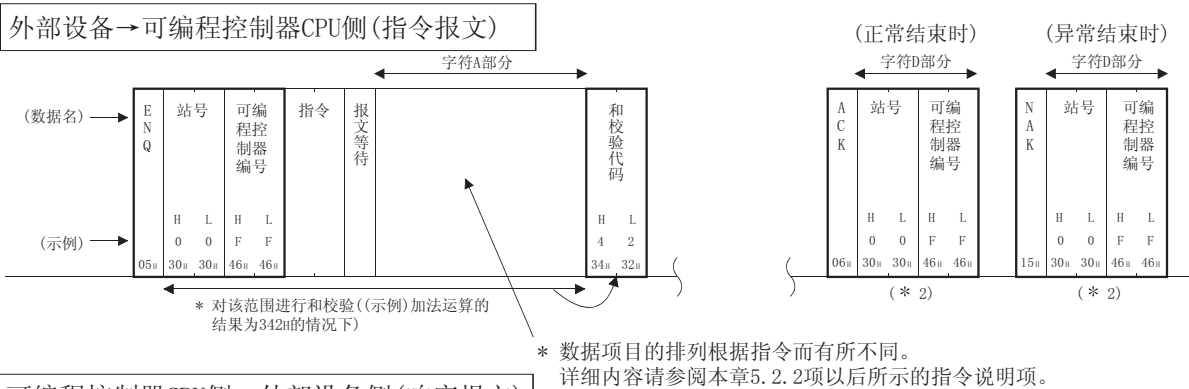
5.1.2 控制步骤、报文格式

以下介绍以 A 兼容 1C 帧进行数据通信时各格式的控制步骤及各指令的报文格式。
本项的报文说明图中所示的□部分为各指令所共用，与本章 5.2.2 项以后中所示的报文说明图的*部分相对应。
关于□部分的数据内容、数据指定方法，请参阅 5.1.3 项。

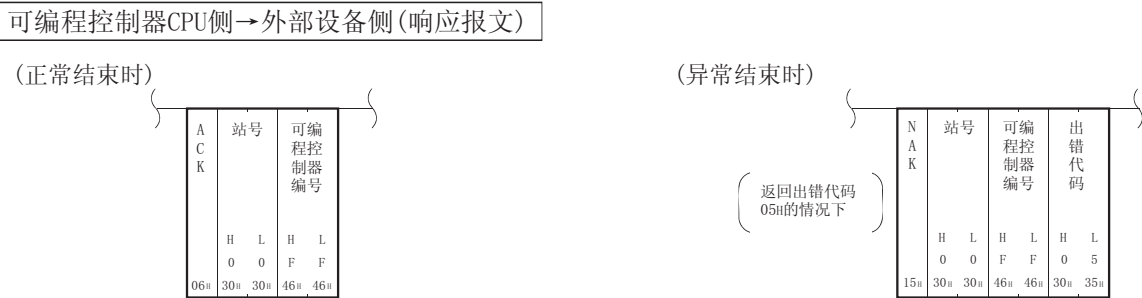
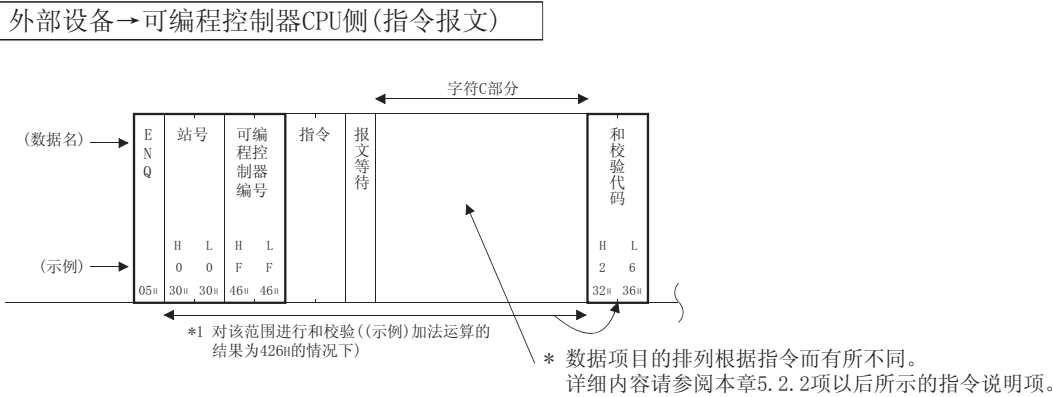
要点	
	本项中所示的 A 兼容 1C 帧的格式 1～格式 4((1)～(4)图)如下所示。 (a) 和校验设置为“有”的情况下，仅对图(*1)部分的字符进行和校验。 (b) 图中的“字符 A 部分”、“字符 B 部分”、“字符 C 部分”的内容根据处理内容而有所不同。详细内容请参阅各指令的说明。此外，各字符部分的内容 4 种格式均相同。 (c) 在外部设备侧对可编程控制器的数据进行读取的情况下，图中(*2)部分的报文发送可以省略。 (d) 外部设备与可编程控制器 CPU 的系统配置以 m:n 连接进行数据通信时，不能使用控制步骤格式 3。

(1) 以格式 1 进行通信时

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

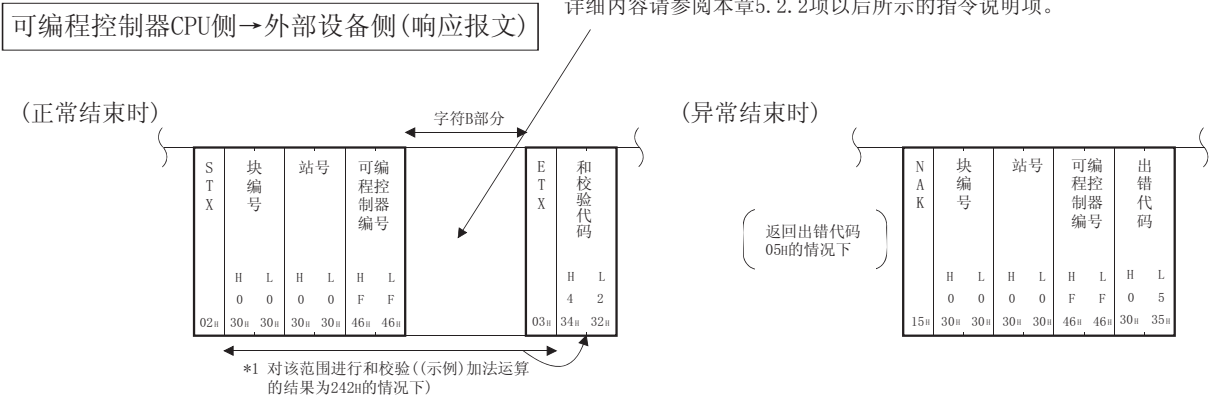
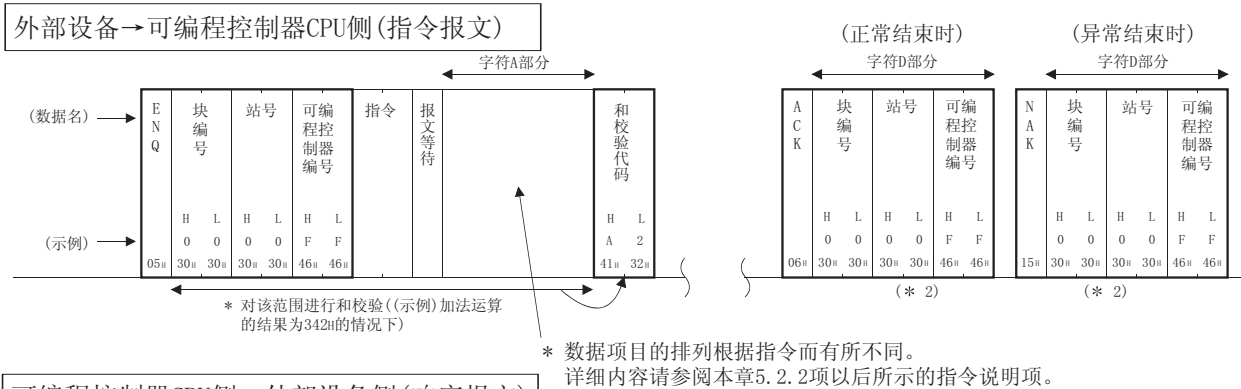


(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

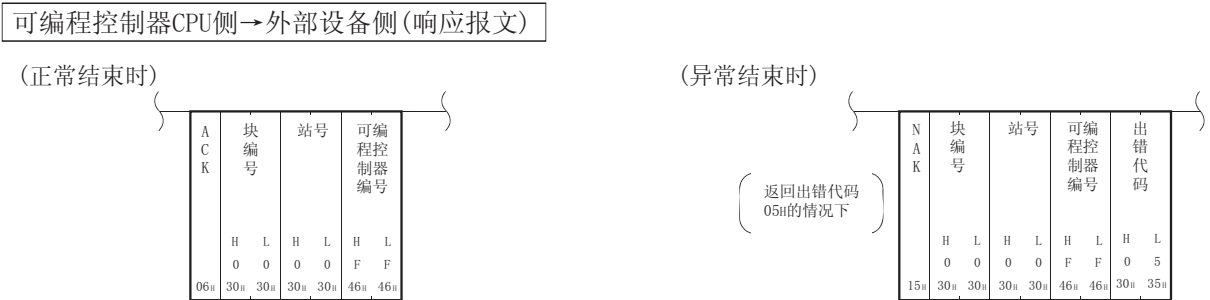
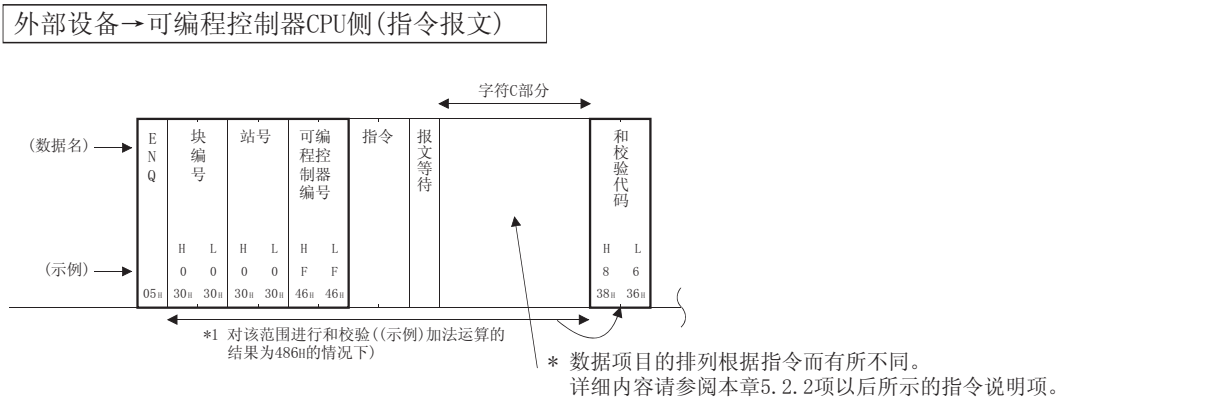


(2) 以格式 2 进行通信时

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

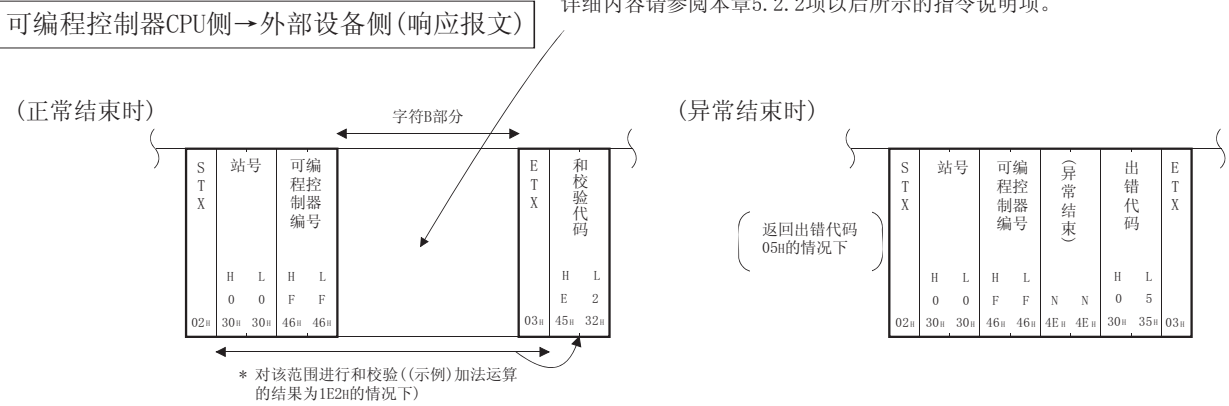
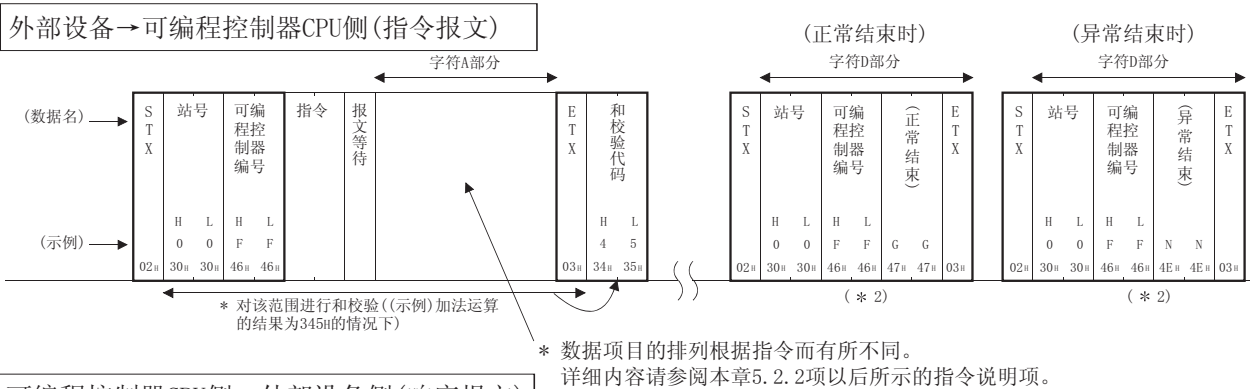


(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

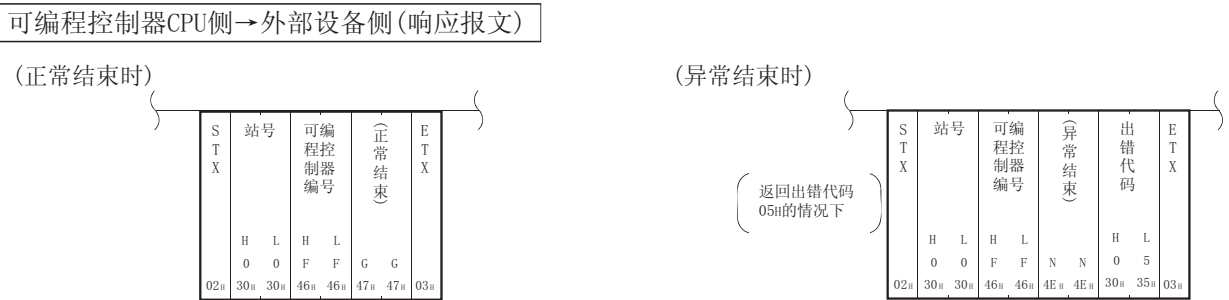
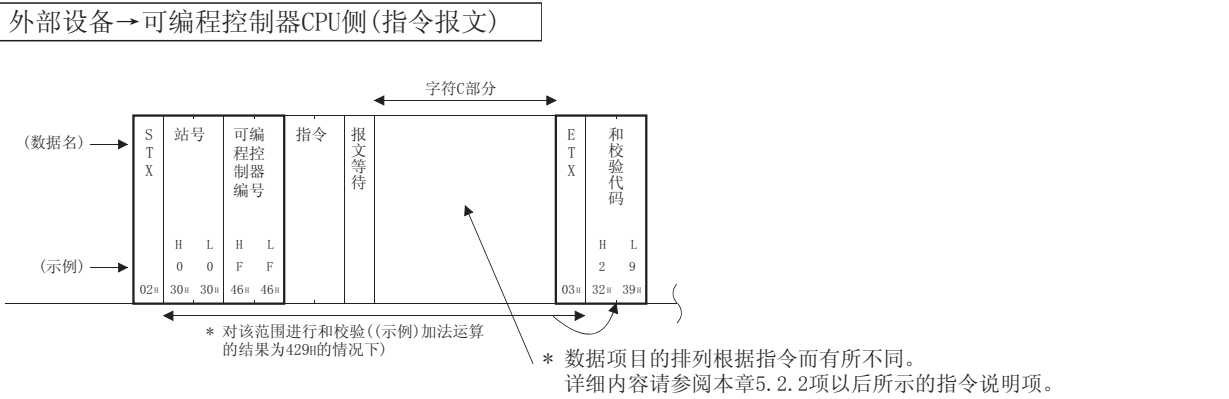


(3) 以格式 3 进行通信时

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

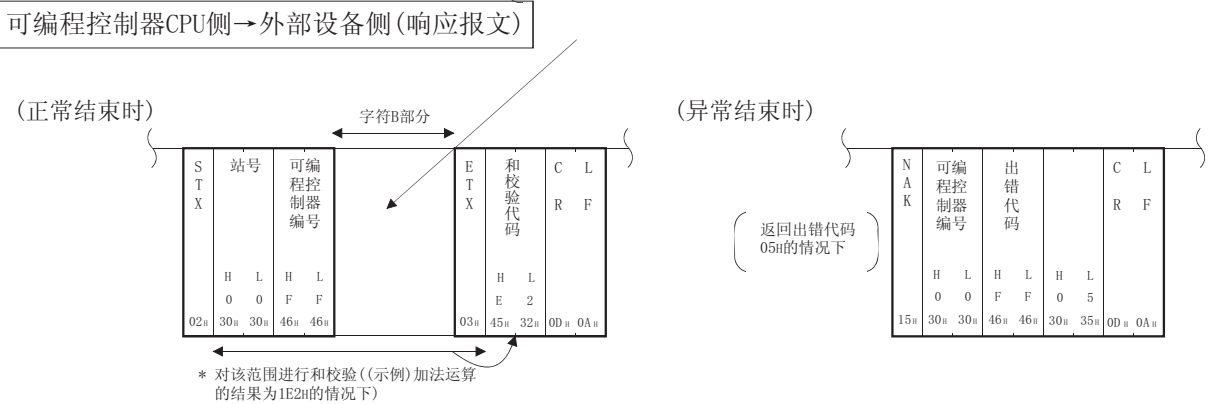
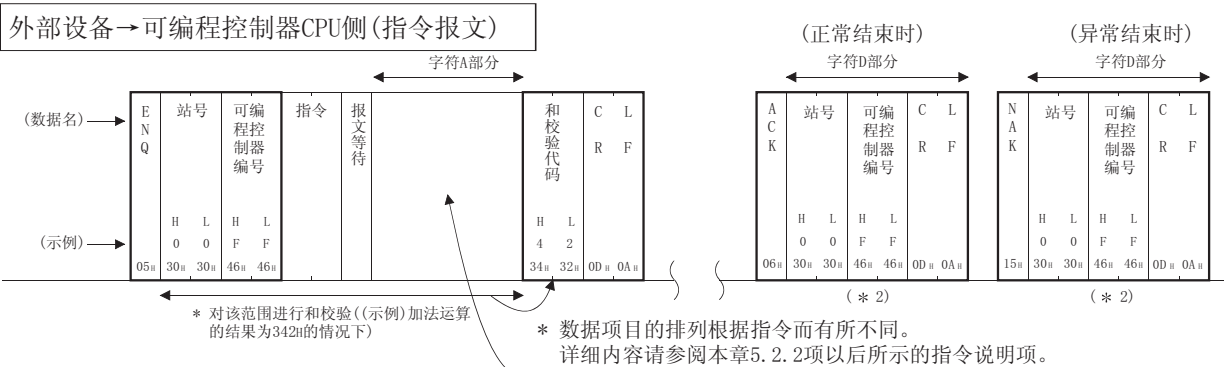


(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

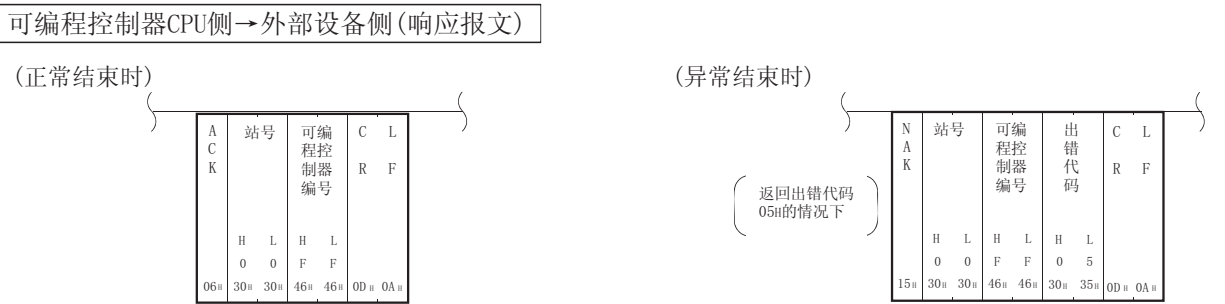
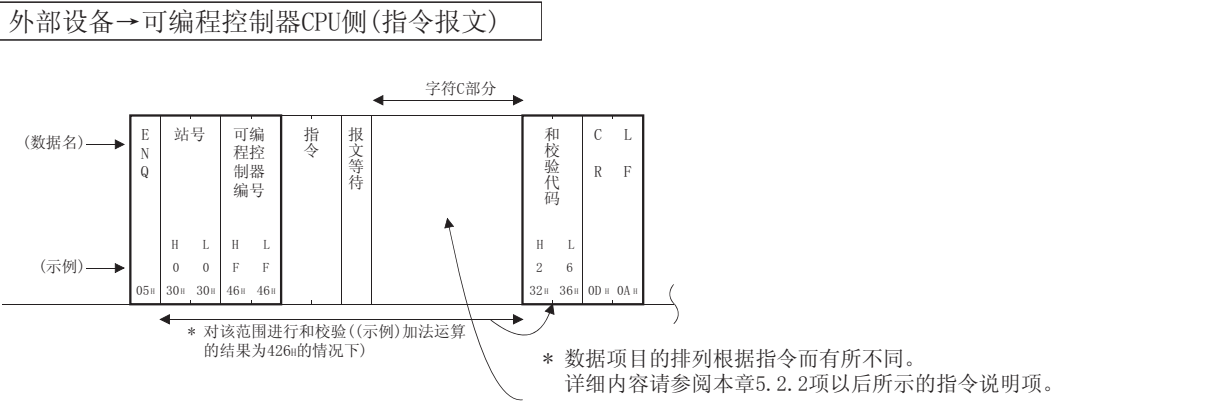


(4) 以格式 4 进行通信时

(a) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下



(b) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



5.1.3 A 兼容 1C 帧的数据指定项目的内容

以下对以 C24 用的 A 兼容 1C 帧进行数据通信时，与 QnA 兼容 3C/4C 帧的指定内容・指定方法不相同的数据项目有关内容进行说明。

(1) 控制代码

控制代码如下表所示。

符号名	代码(16 进制数)	内容	符号名	代码(16 进制数)	内容	符号名	代码(16 进制数)	内容
NUL	00 _H	Null	ENQ	05 _H	Enquiry	CR	0D _H	Carriage Return
STX	02 _H	Start of Text	ACK	06 _H	Acknowledge	NAK	15 _H	Negative Acknowledge
ETX	03 _H	End of Text	LF	0A _H	Line Feed	G	47 _H	Good
EOT	04 _H	End of Transmission	CL	0C _H	Clear	N	4E _H	No Good

- (a) Null 代码(00_H) 在所有的报文中均被忽略。因此即使报文中存在有 Null 代码，也不进行任何处理。
- (b) 格式 3 的控制步骤中的控制代码“GG”表示 ACK(正常结束)，“NN”表示 NAK(异常结束)。
- (c) EOT、CL 是用于对 MC 协议数据通信的传送顺控程序进行初始化，将 C24 置为等待接收来自于外部设备的指令的状态的代码。

C24 接收了 EOT/CL 时，其情况如下所示。(*1)

- 根据来自于外部设备的请求，中止对可编程控制器 CPU 进行的读取/写入处理。
在这种情况下，不对之前接收的指令发送响应报文(5.1.2 项中所示的 B 部分)。
- 对 MC 协议的传送顺控程序进行初始化，置为等待接收来自于外部设备的指令的状态。
- 没有对于 EOT/CL 的接收的响应报文。(不向外部设备发送任何内容。)
- 在接通请求功能(3.11 项中所示的从可编程控制器 CPU 至外部设备的数据发送功能)的执行过程中接收了 EOT/CL 时，至外部设备的接通请求数据的发送将停止。

通过外部设备侧进行以下动作时，应根据所使用的格式以下述报文格式将 EOT/CL 从外部设备发送至 C24。

- 1) 取消根据之前发送的指令进行的读取/写入请求。
(发送了写入请求的情况下，数据已被写入到可编程控制器 CPU 中时，将无法取消写入请求。)
- 2) 在发送指令之前，预先将 C24 置为指令接收等待状态。
- 3) 无法正常进行数据通信时，将 C24 置为启动时的状态。

格式 1~3 的情况下			格式 4 的情况下		
外部设备	C L	或	E O T	外部设备	C L 或 C R 或 L R F
可编程控制器侧			可编程控制器侧		

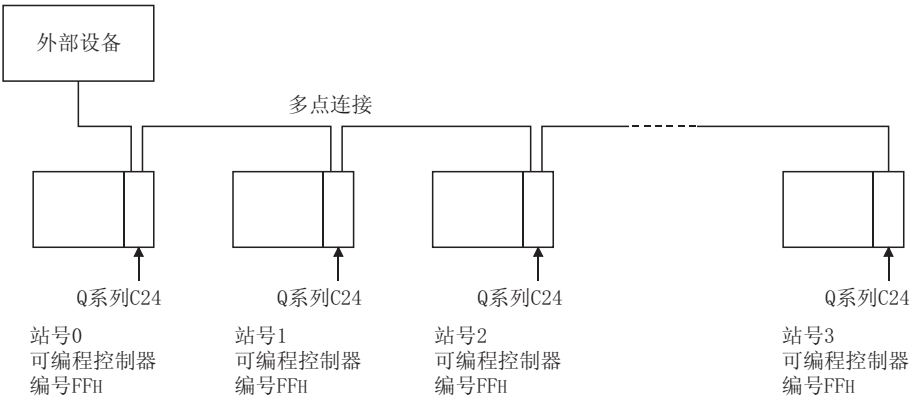
* 发送 EOT、CL 时，仅发送左图中所示的数据。
不需要发送站号及可编程控制器编号等。

(2) 块编号

块编号是在外部设备侧对相应报文附加了某种含义的任意编号，用作数据整理编号等。
指定的内容及指定方法与以 QnA 兼容 3C/4C 帧进行通信时相同。请参阅 3.1.6 项。

(3) 站号

站号是通过 GX Works2 或 GX Developer 设置的 C24 的站号，用于识别外部设备对哪个站的 C24 进行访问。
(示例)



指定的内容及指定方法与以 QnA 兼容 3C/4C 帧进行通信时相同。请参阅 3.1.6 项。

(4) 可编程控制器编号

可编程控制器编号是用于识别对 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的哪个可编程控制器进行访问。
对于可编程控制器编号，使用 FFH 或将网络模块上连接的站号在 00H~40H 范围内转换为 ASCII 代码 2 位(16 进制)后使用。

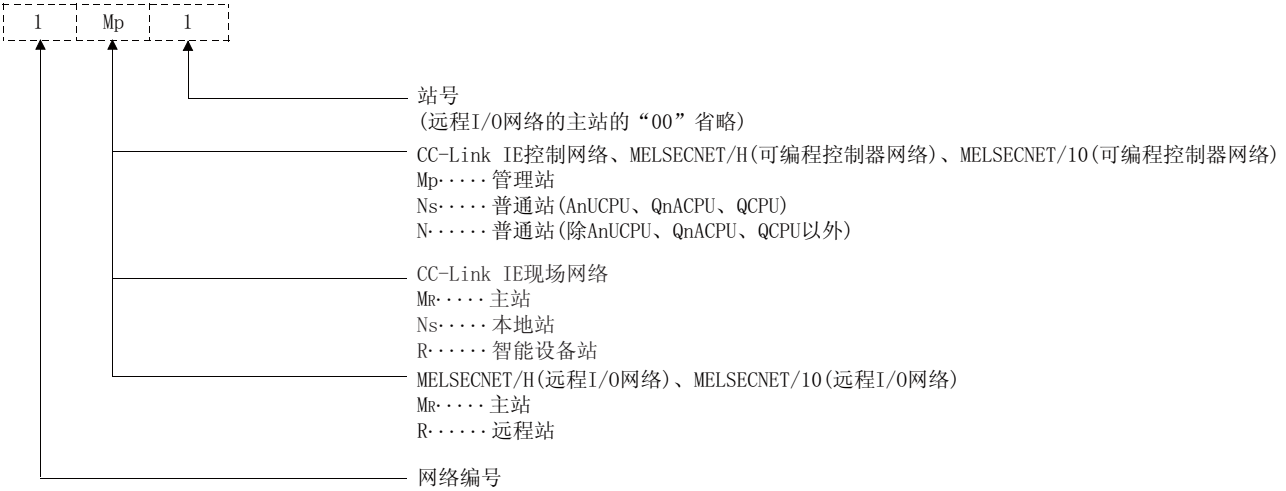
	外部设备的访问站	外部设备指定的可编程控制器编号
1	外部设备连接站(本站)	FF _H
2	来自于外部设备连接站的多点连接站	FF _H
3	CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的站(上述 1、2 除外) (*1)	01 _H ~40 _H (1~64) 访问站的站号 (*2)
4	MELSECNET/H 上的远程主站(将外部设备与远程 I/O 站的 C24 相连接时)	00 _H

*1 对网络模块(以太网、CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10)的“其它站访问时的有效模块”设置中指定的网络上的模块进行访问。
但是，对 MELSECNET/H 远程 I/O 站上设置的“其它站访问时的有效模块”的设置将被忽略，将对 MELSECNET/H 远程 I/O 站的可编程控制器编号中指定的其它站进行访问。
*2 不能指定 CC-Link IE 控制网络、CC-Link IE 现场网络的站号 41H~78H (65~120)。

要点	
	(1) 对于网络模块(以太网、CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10) 安装站，必须设置“网络参数”。 (2) 通过 A 兼容 1C 帧对其它站可编程控制器 CPU 进行访问的情况下，需要对网络模块安装站(MELSECNET/H 远程 I/O 站除外)进行“其它站访问时的有效模块”的设置。

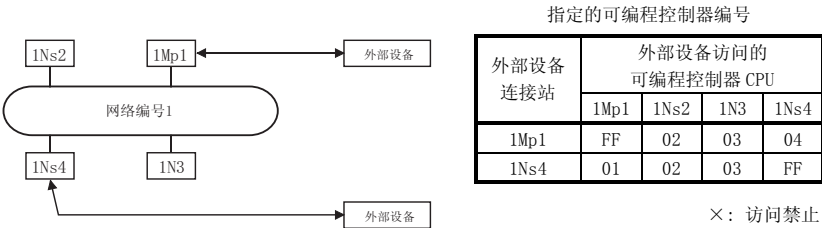
[可编程控制器编号的指定示例]
(图中所示各站的符号的含义)

- 网络系统 (CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10)

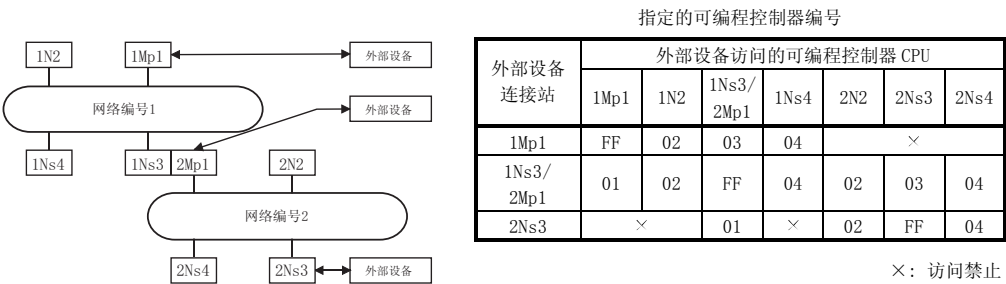


(a) CC-Link IE 控制网络、MELSECNET/H(可编程控制器网络)、MELSECNET/10(可编程控制器网络)的情况下

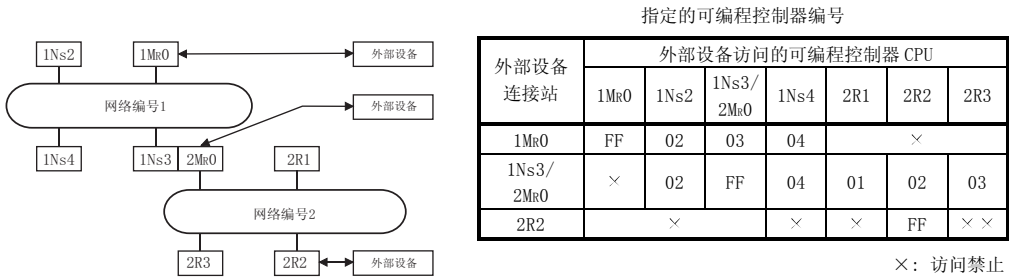
- 二级系统时



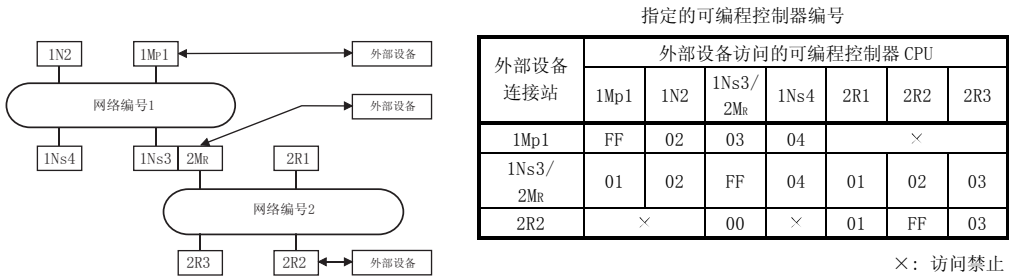
- 多级系统时



(b) CC-Link IE 现场网络的情况下
• 多级系统时



(c) MELSECNET/H(远程 I/O 网络)的情况下
• 多级系统时



(5) 指令

是用于指定从外部设备对相应可编程控制器进行读取、写入等的访问内容的命令。将 5.2 节~5.5 节中所示的指令转换为 ASCII 代码 2 位 (16 进制) 后使用。

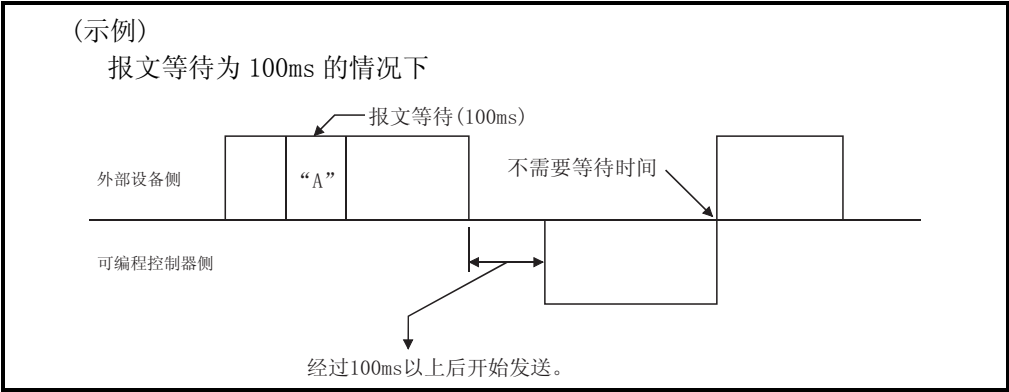
(6) 报文等待

报文等待是用于指定响应发送的延迟时间的数据。根据外部设备，从指令发送起至变为接收状态为止有可能需要一定的时间。

对 C24 从外部设备接收指令后，至结果发送为止的最短等待时间进行指定。应根据外部设备的规格指定等待时间。

等待时间在 0~150ms 的范围内以 10ms 为单位进行指定，将 10ms 作为 1H 将 0H~FH (0~15) 转换为 ASCII 代码 1 位 (16 进制) 后使用。

报文等待的指定示例如下所示。



(7) 字符 A 部分

是用于从 C24 中对可编程控制器 CPU 执行指令中指定的读取请求的数据，通过 ASCII 代码进行发送。

对于字符 A 部分的内容，根据从外部设备发送的指令而有所不同。

详细内容请参阅 5.2 节~5.5 节。

(8) 字符 B 部分

是 C24 对指令中指定的请求返回至外部设备的数据，通过 ASCII 代码进行发送。

对于字符 B 部分的内容，根据之前从外部设备发送的指令而有所不同。详细内容请参阅 5.2 节~5.5 节。

(9) 字符 C 部分

是用于从 C24 中对可编程控制器 CPU 执行指令中指定的写入请求的数据，通过 ASCII 代码进行发送。

对于字符 C 部分的内容，根据从外部设备发送的指令而有所不同。详细内容请参阅 5.2 节~5.5 节。

(10)和校验代码

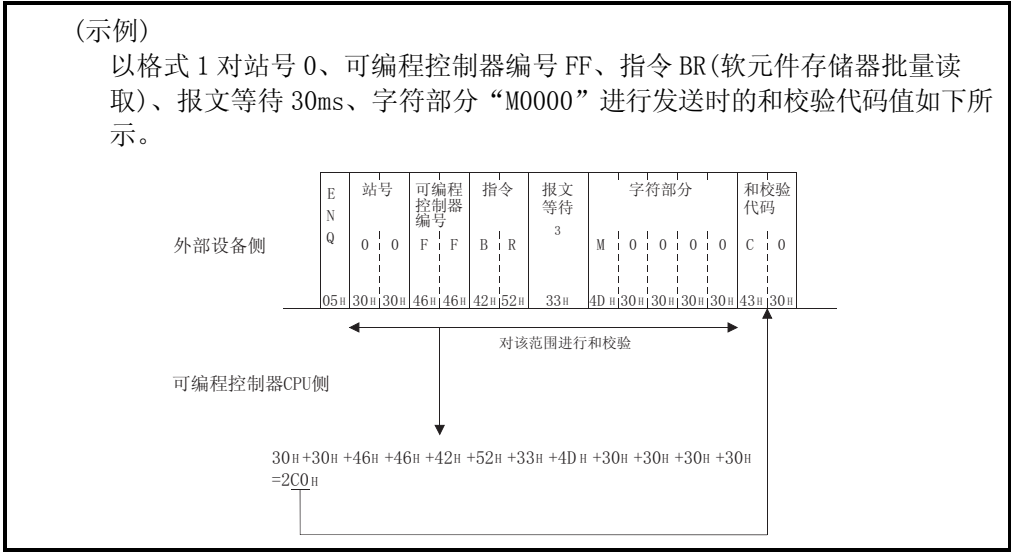
和校验代码是指，将和校验的对象数据作为二进制数据进行加法运算的结果(和)的低位 1 字节(8 位)转换为 ASCII 代码 2 位(16 进制)的数据。

在 GX Works2 或 GX Developer 的设置中设置了“无和校验”时，C24 不对发送报文附加和校验代码。此外，接收数据中作为无和校验代码的数据处理。

设置了“有和校验”时，C24 对发送报文附加和校验代码。

此外，接收数据中作为有和校验代码的数据进行检查。

和校验代码的内容示例如下。



(11)出错代码

- 出错代码是表示异常响应时的出错内容的代码。
- 对于出错代码，在 00H~FFH 的范围内通过 ASCII 代码 2 位(16 进制)进行发送。
- 同时发生了多个出错时，C24 对最先检测出的出错代码进行发送。
- 关于出错代码的详细内容请参阅用户手册(基本篇)。

5.1.4 字符部分的传送数据的思路

对于使用各指令在外部设备与可编程控制器 CPU 之间进行数据收发时的字符部分处理的传送数据的思路，与以 QnA 兼容 3C/4C 帧进行数据通信时的情况相同。

关于字符部分处理的传送数据中的位软元件数据、字软元件数据的思路，请参阅 3.1.7 项。

5.1.5 A 兼容 1C 帧用指令及功能列表

以 A 兼容 1C 帧从外部设备对可编程控制器进行访问时的指令及其功能等如下所示。

功能			指令		处理内容	1 次通信中可处理的点数	
			符号	ASCII 代码			
软元件存储器(*6)	批量读取	位单位	BR JR	42 _H , 52 _H 4A _H , 52 _H	对位软元件(X、Y、M 等)以 1 点为单位进行读取。	256 点	
		字单位	WR QR	57 _H , 52 _H 51 _H , 52 _H	对位软元件(X、Y、M 等)以 16 点为单位进行读取。	32 字(512 点)	
					对字软元件(D、T、C 等)以 1 点为单位进行读取。	64 点	
	批量写入(*5)	位单位	BW JW	42 _H , 57 _H 4A _H , 57 _H	向位软元件(X、Y、M 等)以 1 点单位进行写入。	160 点	
		字单位	WW QW	57 _H , 57 _H 51 _H , 57 _H	向位软元件(X、Y、M 等)以 16 点单位进行写入。	10 字(160 点)	
					向字软元件(D、T、C 等)以 1 点单位进行写入。	64 点	
	测试 (随机写入)(*5)	位单位	BT JT	42 _H , 54 _H 4A _H , 54 _H	将位软元件(X、Y、M 等)以 1 点为单位随机指定软元件、 软元件编号进行设置/复位。	20 点	
		字单位	WT QT	57 _H , 54 _H 51 _H , 54 _H	向位软元件(X、Y、M 等)以 16 点为单位随机指定软元 件、软元件编号进行设置/复位。	10 字(160 点)	
					向字软元件(D、T、C 等)以 1 点为单位随机指定软元件， 软元件编号进行写入。	10 点	
	监视数据登录 (*3)	位单位	BM JM	42 _H , 4D _H 4A _H , 4D _H	将进行监视的位软元件(X、Y、M 等)以 1 点为单位进行 登录。(*2)	40 点	
		字单位	WM QM	57 _H , 4D _H 51 _H , 4D _H	将进行监视的位软元件(X、Y、M 等)以 16 点为单位进行 登录。(*2)	20 字(320 点)	
					将进行监视的字软元件(D、T、C 等)以 1 点为单位进行登 录。	20 点	
	监视	位单位	MB MJ	4D _H , 42 _H 4D _H , 4A _H	对进行了监视数据登录的软元件进行监视。	(登录点数)	
		字单位	MN MQ	4D _H , 4E _H 4D _H , 51 _H			
扩展文件寄存 器	批量读取		ER	45 _H , 52 _H	对扩展文件寄存器(R)以 1 点为单位进行读取。	64 点	
	批量写入		EW	45 _H , 57 _H	对扩展文件寄存器(R)以 1 点为单位进行写入。	64 点	
	测试(随机写入)		ET	45 _H , 54 _H	对扩展文件寄存器(R)以 1 点为单位随机指定块编号、软 元件编号进行写入。	10 点	
	监视数据登录(*3)		EM	45 _H , 4D _H	对进行监视的扩展文件寄存器(R)以 1 点为单位进行登 录。	20 点	
	监视	字单位	ME	4D _H , 45 _H	对进行了监视数据登录的扩展文件寄存器(R)进行监视。	(登录点数)	
	直接读取	字单位	NR	4E _H , 52 _H	在无需在意扩展文件寄存器的块编号的状况下通过连号指 定软元件编号以 1 点为单位进行读取。	64 点	
	直接写入	字单位	NW	4E _H , 57 _H	在无需在意扩展文件寄存器的块编号的状况下通过连号指 定软元件编号以 1 点为单位进行写入。	64 点	
智能功能模 块	批量读取		TR	54 _H , 52 _H	对智能功能模块的缓冲存储器的数据进行读取。	128 字节	
	批量写入		TW	54 _H , 57 _H	对智能功能模块的缓冲存储器的数据进行写入。		
回送测试			TT	54 _H , 54 _H	将从外部设备接收的字符原样不变地发送至外部设备(返 回)。	254 字节	

功能	—			访问站(*4)						可编程控制器 CPU 的状态(*1)			参阅章节
				ACPU	QnA CPU	QCPU (*7)	LCPU	安全 CPU	MELSECNET/H 远程站	STOP 中	RUN 中		
											设置为允许时	设置为不允许时	
软件元件存储器	批量读取	位单位	○	○	○	○	○	○	○	○	○	5.2.2 项	
		字单位	○	○	○	○	○	○	○	○	○	5.2.5 项	
	批量写入	位单位	○	○	○	○	×	○	○	○	×	5.2.3 项	
		字单位	○	○	○	○	×	○	○	○	×	5.2.6 项	
	测试 (随机写入)	位单位	○	○	○	○	×	○	○	○	×	5.2.4 项	
		字单位	○	○	○	○	×	○	○	○	×	5.2.7 项	
	监视数据登录	位单位	○	○	○	○	×	○	○	○	○	5.2.8 项	
		字单位	○	○	○	○	×	○	○	○	○		
	监视	位单位	○	○	○	○	×	○	○	○	○	5.2.8 项	
		字单位											
扩展文件寄存器	批量读取		○	×	×	×	×	×	○	○	○	5.3.4 项	
	批量写入		○	×	×	×	×	×	○	○	×	5.3.5 项	
	测试 (随机写入)		○	×	×	×	×	×	○	○	×	5.3.8 项	
	监视数据登录		○	×	×	×	×	×	○	○	○	5.3.9 项	
	监视		○	×	×	×	×	×	○	○	○	5.3.9 项	
	直接读取	字单位	○	×	×	×	×	×	○	○	○	5.3.6 项	
	直接写入	字单位	○	×	×	×	×	×	○	○	×	5.3.7 项	
智能功能模块	批量读取		○	×	×	×	×	×	○	○	○	5.4.3 项	
	批量写入		○	×	×	×	×	×	○	○	×	5.4.4 项	
回送测试			-	-	-	-	-	-	-	○	○	5.5 节	

- *1 至可编程控制器 CPU 的运行中写入的允许/禁止设置是在 GX Works2 或 GX Developer 的以下画面中进行。
 - C24 的情况下:
“I/O 模块、智能功能模块开关设置”画面
- *2 除 AnA/AnU/QnA/L/QCPU 以外的情况下, 每 1 点软元件 X(输入)变为 2 点的处理点数。指定软元件中包含有 X 时, 应满足以下公式。
$$((X \text{ 的指定点数} \times 2) + \text{其它软元件指定点数}) \leq 1 \text{ 次通信中可处理的点数}$$

仅指定了 X 的情况下, 1 次通信中可处理的点数将为表中的 1/2。
- *3 在各接口中可以将监视数据登录用的 5 种类型的指令(BM、JM、WM、QM、EM)的软元件同时登录到 C24 中。
- *4 关于表中所示的访问站的详细内容, 请参阅 2.6.1 项。
- *5 执行指令的 Q/L/QnACPU 中被实施了系统保护时将变为出错状态, 并返回异常响应。
- *6 对 Q/L/QnACPU 以外的扩展文件寄存器进行读取/写入的情况下, 应使用扩展寄存器的专用指令。
- *7 对通用型 QCPU 进行访问的情况下, 应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。
序列号的前 5 位数为 10101 以前的情况下, 应以 QnA 兼容 2C/3C/4C 帧进行访问。

5.2 软元件存储器的读取、写入

以下对进行软元件存储器的读取、写入时的控制步骤的指定内容以及指定示例有关内容进行说明。

此外，进行扩展文件寄存器的读取、写入时，应使用扩展文件寄存器专用的指令。
(参阅 5.3 节)

5.2.1 指令及软元件范围

(1) 用于软元件存储器的读取、写入的指令

(a) ACPU 公共指令

项目		指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态(*1)			参阅章节
		符号	ASCII 代码			STOP 中	RUN 中		
							设置为允许时	设置为不允许时	
批量读取	位单位	BR	45 _H , 52 _H	对位软元件(X、Y、M 等)以 1 点为单位进行读取。	256 点	○	○	○	5.2.2 项
	字单位	WR	57 _H , 52 _H	对位软元件(X、Y、M 等)以 16 点为单位进行读取。	32 字(512 点)				5.2.5 项
				对字软元件(D、T、C 等)以 1 点为单位进行读取。	64 点				
批量写入(*4)	位单位	BW	42 _H , 57 _H	对位软元件(X、Y、M 等)以 1 点为单位进行写入。	160 点	○	○	×	5.2.3 项
	字单位	WW	57 _H , 57 _H	对位软元件(X、Y、M 等)以 16 点为单位进行写入。	10 字(160 点)				5.2.6 项
				对字软元件(D、T、C 等)以 1 点为单位进行写入。	64 点				
测试 (随机写入)(*4)	位单位	BT	42 _H , 54 _H	将位软元件(X、Y、M 等)以 1 点为单位随机指定软元件 • 软元件编号进行设置/复位。	20 点	○	○	×	5.2.4 项
	字单位	WT	57 _H , 54 _H	将位软元件(X、Y、M 等)以 16 点为单位随机指定软元件 • 软元件编号进行设置/复位。	10 字(160 点)				5.2.7 项
				对字软元件(D、T、C 等)以 1 点为单位随机指定软元件 • 软元件编号进行写入。	10 点				
监视数据登录(*3)	位单位	BM	42 _H , 4D _H	对进行监视的位软元件(X、Y、M 等)以 1 点为单位进行设置。(*2)	40 点	○	○	○	5.2.8 项
	字单位	WM	57 _H , 4D _H	对进行监视的位软元件(X、Y、M 等)以 16 点为单位进行设置。(*2)	20 字(320 点)				
				对进行监视的字软元件(D、T、C 等)以 1 点为单位进行设置。	20 点				
监视	位单位	MB	4D _H , 42 _H	对进行了监视数据登录的软元件进行监视。	-	○	○	○	5.2.8 项
	字单位	MN	4D _H , 4E _H						

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

关于*1、*2、*3 的有关内容，请参阅 5.1.5 项的*1、*2、*3。

关于*4，请参阅 5.1.5 项的*5。

(b) AnA/AnUCPU 公共指令

项目		指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态(*1)			参阅章节
		符号	ASCII 代码			STOP 中	RUN 中		
							设置为允许时	设置为不允许时	
批量读取	位单位	JR	4A _H , 52 _H	对位软元件(X、Y、M 等)以 1 点为单位进行读取。	256 点	○	○	○	5.2.2 项
	字单位	QR	51 _H , 52 _H	对位软元件(X、Y、M 等)以 16 点为单位进行读取。	32 字(512 点)				5.2.5 项
				对字软元件(D、T、C 等)以 1 点为单位进行读取。	64 点				
批量写入	位单位	JW	4A _H , 57 _H	对位软元件(X、Y、M 等)以 1 点为单位进行写入。	160 点	○	○	×	5.2.3 项
	字单位	QW	51 _H , 57 _H	对位软元件(X、Y、M 等)以 16 点为单位进行写入。	10 字(160 点)				5.2.6 项
				对字软元件(D、T、C 等)以 1 点为单位进行写入。	64 点				
测试 (随机写入)	位单位	JT	4A _H , 54 _H	将位软元件(X、Y、M 等)以 1 点为单位随机指定软元件 • 软元件编号进行设置/复位。	20 点	○	○	×	5.2.4 项
	字单位	QT	51 _H , 54 _H	将位软元件(X、Y、M 等)以 16 点为单位随机指定软元件 • 软元件编号进行设置/复位。	10 字(160 点)				5.2.7 项
				对字软元件(D、T、C 等)以 1 点为单位随机指定软元件 • 软元件编号进行写入。	10 点				
监视数据登录(*3)	位单位	JM	4A _H , 4D _H	将进行监视的位软元件(X、Y、M 等)以 1 点为单位进行设置。(*2)	40 点	○	○	○	5.2.8 项
	字单位	QM	51 _H , 4D _H	将进行监视的位软元件(X、Y、M 等)以 16 点为单位进行设置。(*2)	20 字(320 点)				
				将进行监视的字软元件(D、T、C 等)以 1 点为单位进行设置	20 点				
监视	位单位	MJ	4D _H , 4A _H	对进行了监视数据登录的软元件进行监视。	-	○	○	○	5.2.8 项
	字单位	MQ	4D _H , 51 _H						

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

关于*1、*2、*3 的有关内容，请参阅 5.1.5 项的*1、*2、*3。

(2) 可访问的软元件范围

(a) 使用 ACPU 公共指令时的软元件范围

软元件存储器的访问中可指定的软元件以及软元件编号范围如下所示。

- 1) 各软元件由 5 个字符所构成，以下表中所示的排列进行指定。
对于软元件编号，可以将高位的“0”列(示例表示 X0070 的一部分)以空格(20H)进行指定。

$$\begin{array}{c} \text{软元件} \\ \left(\begin{array}{c} 1 \text{ 个字符} \\ (\text{T/C 时为 2 个字符}) \end{array} \right) \end{array} + \begin{array}{c} \text{软元件编号} \\ \left(\begin{array}{c} 4 \text{ 个字符} \\ (\text{T/C 时为 3 个字符}) \end{array} \right) = 5 \text{ 个字符}$$

软元件(*1)	字符编号范围(字符)	10 进制/16 进制表示	软元件(*1)	字符编号范围(字符)	10 进制/16 进制表示
输入 X	X0000 ~ X07FF	16 进制表示	定时器(触点) T	TS000 ~ TS255	10 进制表示
输出 Y	Y0000 ~ Y07FF		定时器(线圈) T	TC000 ~ TC255	
内部继电器 M	M0000 ~ M2047	10 进制表示	定时器(当前值) T	TN000 ~ TN255	
锁存继电器 L	L0000 ~ L2047		计数器(触点) C	CS000 ~ CS255	
步进继电器 S	S0000 ~ S2047	16 进制表示	计数器(线圈) C	CC000 ~ CC255	
链接继电器 B	B0000 ~ B03FF		计数器(当前值) C	CN000 ~ CN255	
报警器 F	F0000 ~ F0255	10 进制表示	数据寄存器 D	D0000 ~ D1023	16 进制表示
特殊继电器 M	M9000 ~ M9255		链接寄存器 W	W0000 ~ W03FF	
特殊寄存器 D	D9000 ~ D9255		文件寄存器 R	R0000 ~ R8191	10 进制表示

(b) 使用 AnA/AnUCPU 公共指令时的软元件范围

软元件存储器的访问中可使用的软元件以及软元件编号范围如下所示。

- 1) 各软元件由 7 个字符构成，以下表中所示的排列进行指定。
对于软元件编号，可以将高位的“0”列(表示 X000070 的一部分)以空格(20H)进行指定。

$$\begin{array}{c} \text{软元件} \\ \left(\begin{array}{c} 1 \text{ 个字符} \\ (\text{T/C 时为 2 个字符}) \end{array} \right) \end{array} + \begin{array}{c} \text{软元件编号} \\ \left(\begin{array}{c} 6 \text{ 个字符} \\ (\text{T/C 时为 5 个字符}) \end{array} \right) = 7 \text{ 个字符}$$

软元件(*1)	字符编号范围(字符)	10 进制/16 进制表示	软元件(*1)	字符编号范围(字符)	10 进制/16 进制表示
输入 X	X000000 ~ X001FFF	16 进制表示	定时器(触点) T	TS00000 ~ TS02047	10 进制表示
输出 Y	Y000000 ~ Y001FFF		定时器(线圈) T	TC00000 ~ TC02047	
内部继电器 M	M000000 ~ M008191	10 进制表示	定时器(当前值) T	TN00000 ~ TN02047	
锁存继电器 L	L000000 ~ L008191		计数器(触点) C	CS00000 ~ CS01023	
步进继电器 S	S000000 ~ S008191	16 进制表示	计数器(线圈) C	CC00000 ~ CC01023	
链接继电器 B	B000000 ~ B001FFF		计数器(当前值) C	CN00000 ~ CN01023	
报警器 F	F000000 ~ F002047	10 进制表示	数据寄存器 D	D000000 ~ D008191	16 进制表示
特殊继电器 M	M009000 ~ M009255		链接寄存器 W	W000000 ~ W001FFF	
特殊寄存器 D	D009000 ~ D009255		文件寄存器 R	R000000 ~ R008191	10 进制表示

2) 对 MELSECNET/H 远程 I/O 站进行访问时的可访问软元件范围如下所示。
(对于链接继电器(B)、数据寄存器(D)、链接寄存器(W)，不能对下表中所
示的编号后面的软元件进行访问。)

软元件	字符编号范围(字符)	10 进制/16 进制表示	软元件	字符编号范围(字符)	10 进制/16 进制表示
输入	X	X000000 ~ X001FFF	链接继电器	B	B000000 ~ B001FFF
输出	Y	Y000000 ~ Y001FFF	数据寄存器	D	D000000 ~ D008191
内部继电器	M	M000000 ~ M008191	链接寄存器	W	W000000 ~ W001FFF

*1 对 Q/L/QnACPU 进行读取/写入时，存在有无法访问的软元件及可编程控制器 CPU。
有关详细内容请参阅 5.2.1 项(3)。

要点								
(1) 应在 ACPU 公共指令、AnA/AnU 公共指令中可使用的软元件编号范围以及访问目标可编程控制器 CPU 中可使用的软元件编号范围的范围内进行访问。 (2) 对于字单位指定时的位软元件的起始软元件编号，必须设置为 16 的倍数(10 进制数的情况下为 0、16、. . .)。 此外，对于特殊继电器 M 的 M9000 以后，也可以以 (9000+16 的倍数) 进行指定。 (3) 对于 Q/L/QnACPU 以外，可以对 M、L、S 进行范围指定，但将 M 的编号范围以 L、S 进行指定，或相反的情况下也将进行相同的处理。 (4) 特殊继电器 (M9000~M9255)、特殊寄存器 (D9000~D9255) 被分为只读、写入专用、系统用。 如果对允许写入范围以外进行了写入，有可能导致可编程控制器 CPU 出错。关于特殊继电器、特殊寄存器的详细内容请参阅 ACPU 的编程手册。 (5) 使用 AnACPU、AnUCPU 扩展文件寄存器用专用指令的情况下，文件寄存器(R)的读取・写入应通过 5.4 节中说明的指令进行。 (6) 对于指令的执行过程中进行读取/写入的软元件点数，在 5.2.1 项的表中所示的处理点数(1 次通信中可处理的点数)以内，转换为 ASCII 代码 2 位(16 进制)后进行指定。 但是，仅在指定 256 点时，以“00”进行指定。 (示例：括号内表示 ASCII 代码。) <table><tr><td>5 点的情况下</td><td>: 05(30_H、35_H)</td></tr><tr><td>10 点的情况下</td><td>: 0A(30_H、41_H)</td></tr><tr><td>20 点的情况下</td><td>: 14(31_H、34_H)</td></tr><tr><td>256 点的情况下</td><td>: 00(30_H、30_H)</td></tr></table>	5 点的情况下	: 05(30 _H 、35 _H)	10 点的情况下	: 0A(30 _H 、41 _H)	20 点的情况下	: 14(31 _H 、34 _H)	256 点的情况下	: 00(30 _H 、30 _H)
5 点的情况下	: 05(30 _H 、35 _H)							
10 点的情况下	: 0A(30 _H 、41 _H)							
20 点的情况下	: 14(31 _H 、34 _H)							
256 点的情况下	: 00(30 _H 、30 _H)							

(3) 对 Q/L/QnACPU 进行读取/写入时的注意事项。

- (a) 只有与 AnCPU、AnNCPU、AnACPU、AnUCPU 中存在的软元件相同名称的软元件，才可以在表中所示的范围内进行访问。

对于以下软元件，不能从对象设备进行访问。

- Q/L/QnACPU 中新增的软元件
- 锁存继电器 (L) 以及步进继电器 (S)*1
- 文件寄存器 (R)

*1 即使指定锁存继电器 (L) 以及步进继电器 (S)，也将对内部继电器 (M) 进行访问。

- (b) 对特殊继电器、特殊寄存器按以下方式进行访问。

- 指定为 M9000～M9255 时，对 SM1000～SM1255 进行访问
- 指定为 D9000～D9255 时，对 SD1000～SD1255 进行访问

- (c) 对通用型 QCPU 进行访问的情况下，应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。

使用序列号的前 5 位数为 10101 以前的情况下，应以 QnA 兼容 2C/3C/4C 帧进行访问。

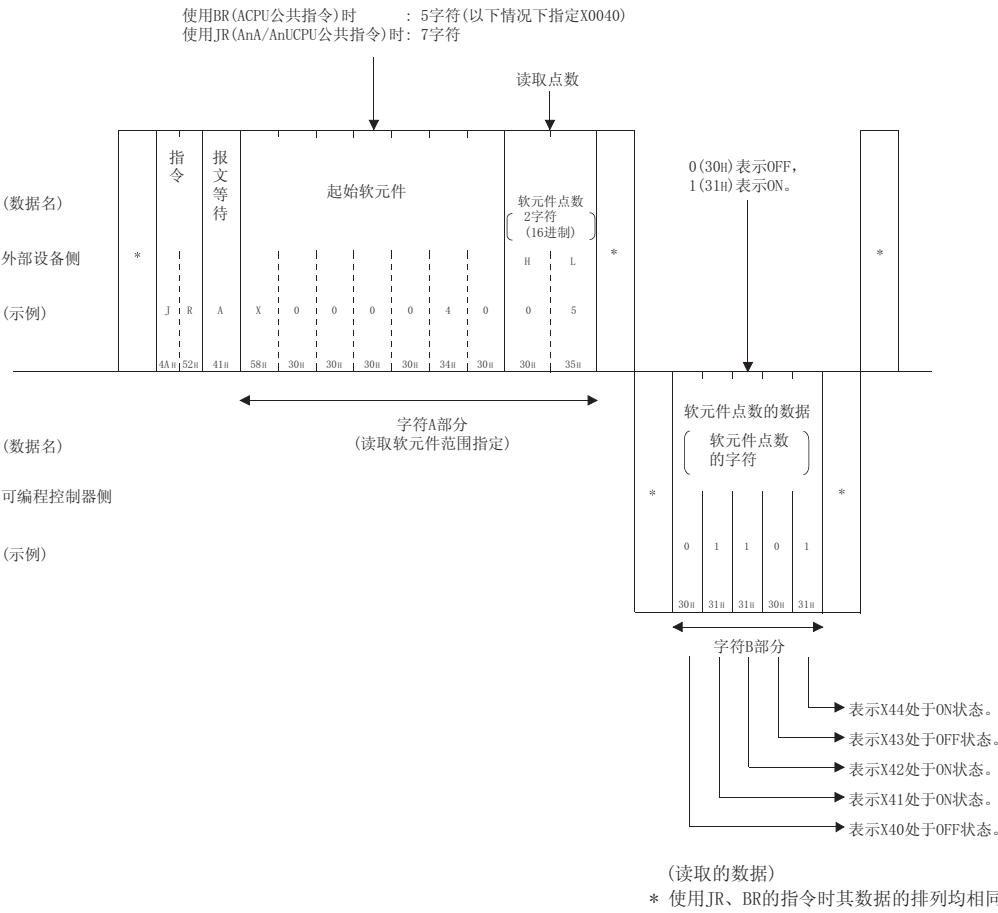
5.2.2 位单位的批量读取(指令：BR、JR)

以下对通过 BR、JR 指令进行位软元件存储器的批量读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

(读取条件)

- 1) 报文等待为 100ms。
- 2) 对 X040~X044 的 5 点进行读取。



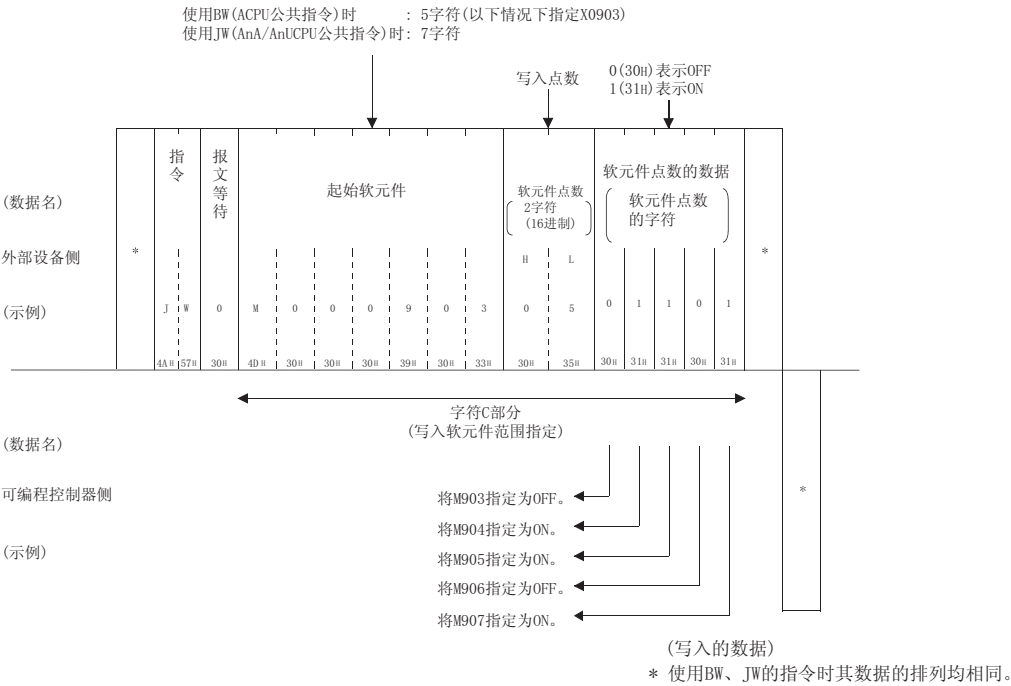
要点
(1) 软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1≦软元件点数≦256 (256 点的设置以 00h 进行指定) • 起始软元件号+软元件点数-1≦最大软元件号
(2) 报文等待时间在 0~150ms 的范围内以 10ms 为单位进行指定，以 0~Fh (16 进制) 进行表示。 因此 100ms 变为 “A”。

5.2.3 位单位的批量写入(指令：BW、JW)

以下对通过 BW、JW 指令进行位软元件存储器的批量写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (写入条件)
- 1) 报文等待为 0ms。
 - 2) 向 M903~M907 写入 5 点。



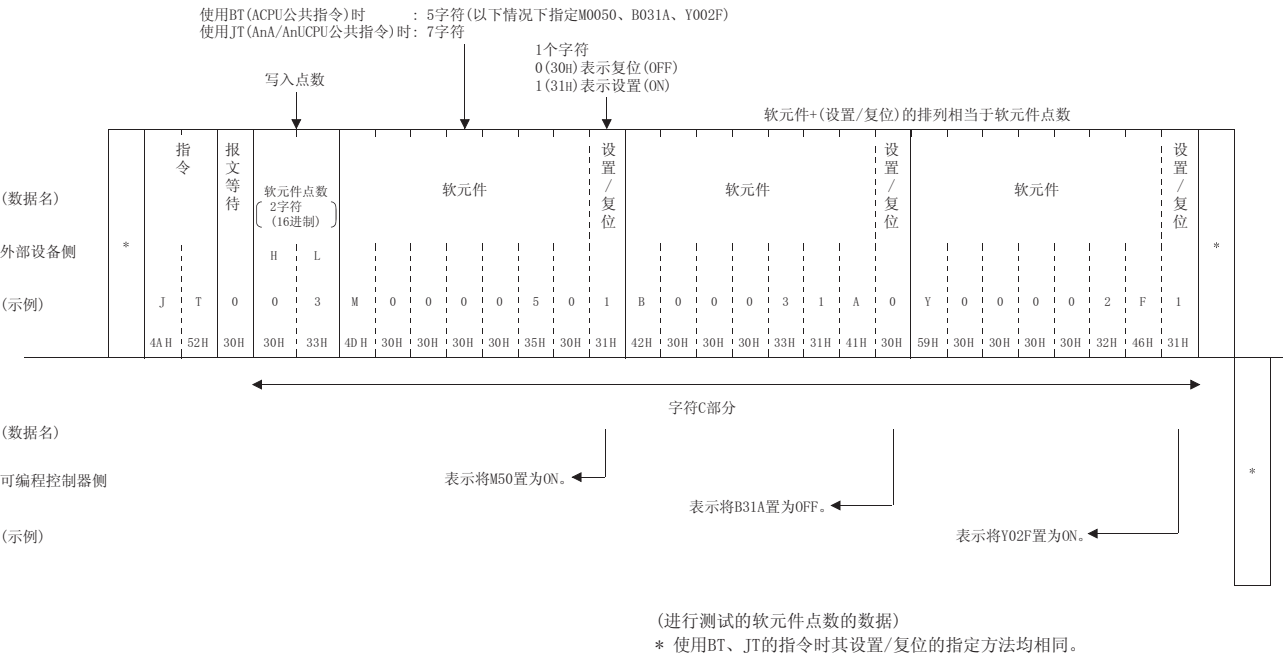
要点	软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1 ≦ 软元件点数 ≦ 160 • 起始软元件号+软元件点数-1 ≦ 最大软元件号
----	---

5.2.4 位单位的测试(随机写入)(指令：BT、JT)

以下对通过 BT、JT 指令随机指定位软元件存储器进行写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (写入条件)
- 1) 报文等待为 0ms。
 - 2) 将 M50 置为 ON、将 B31A 置为 OFF、将 Y02F 置为 ON 的 3 点(3 位)的测试。



要点	软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1 ≤ 软元件点数 ≤ 20
----	---

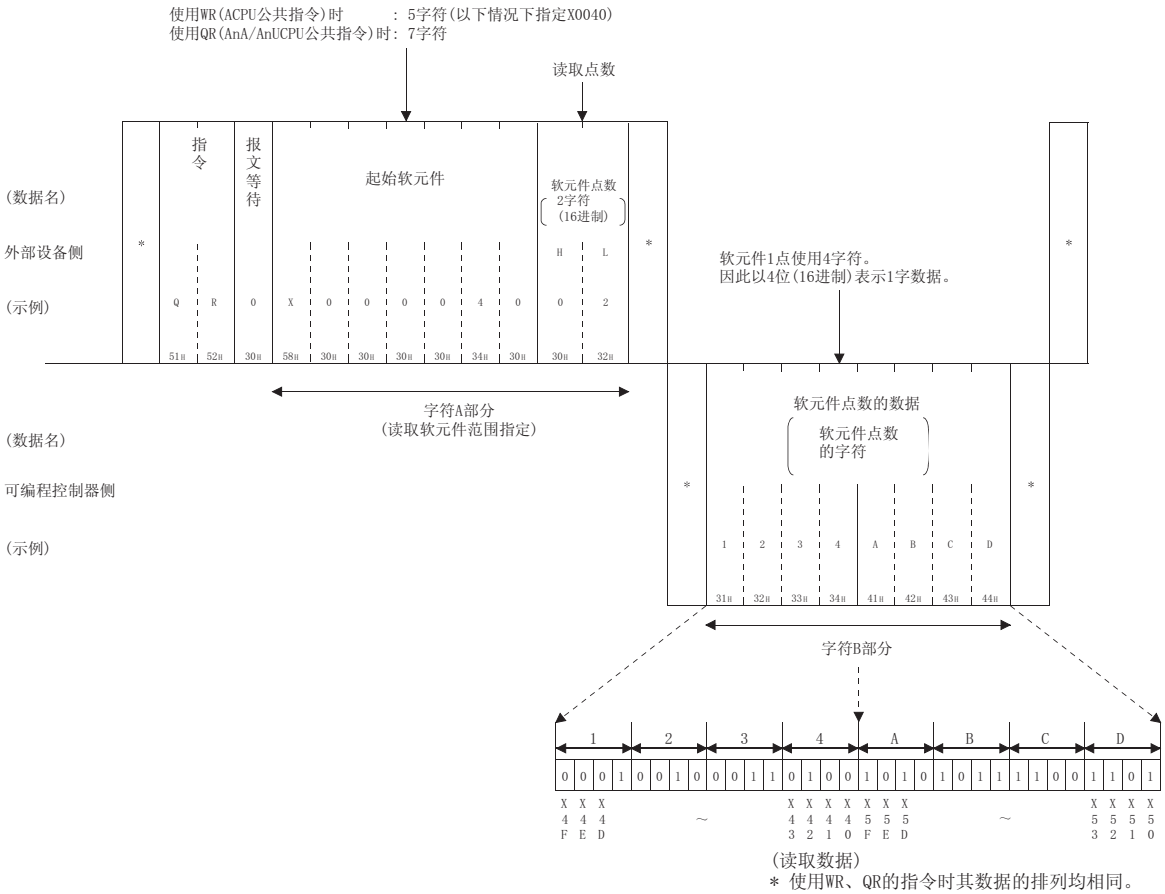
5.2.5 字单位的批量读取(指令：WR、QR)

以下对通过 WR、QR 指令进行的位软元件存储器(16 点单位)的批量读取以及字软元件存储器的批量读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (读取条件)
- 1) 报文等待为 0ms。
 - 2) 对 X040~X05F 的 32 点、T123~T124 当前值的 2 点(均为 2 字)进行读取。

(1) 对位软元件存储器进行读取的情况下

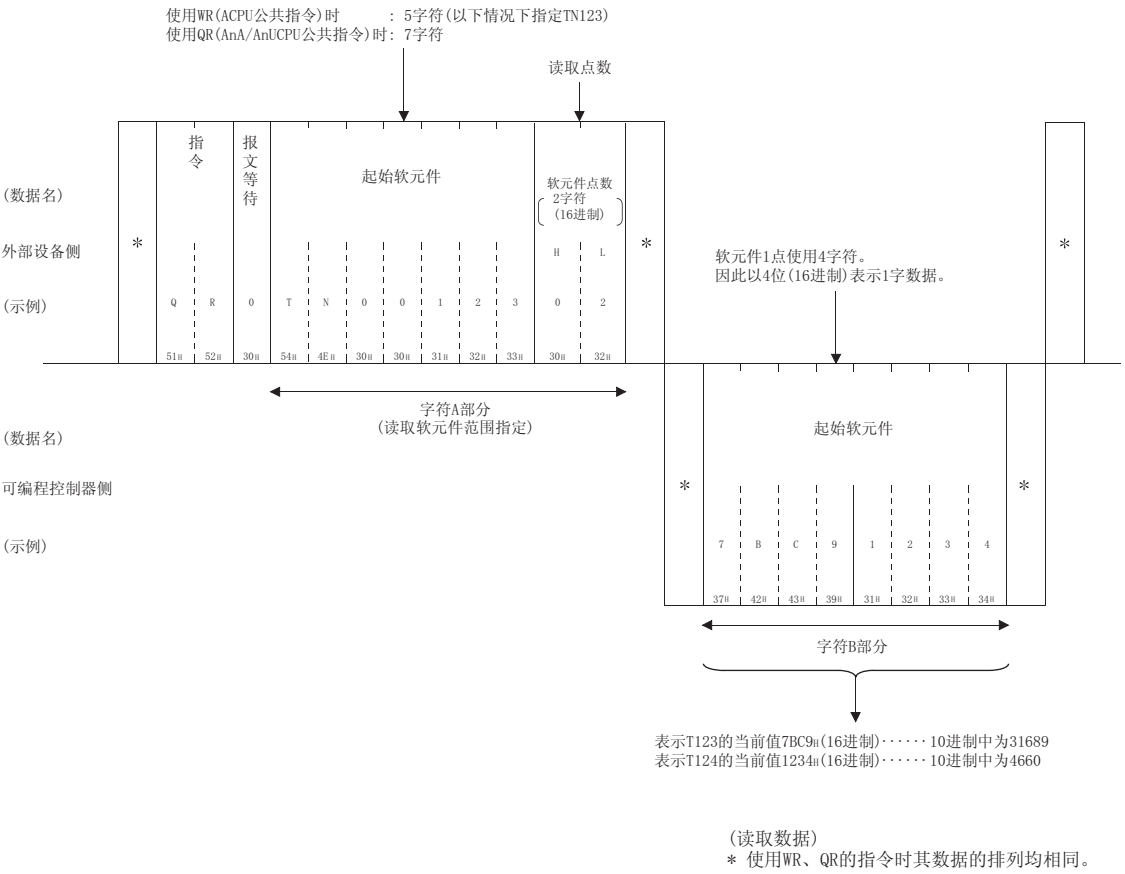


要点
(1) 对位软元件存储器进行读取时，软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1 ≦ 软元件点数 ≦ 32 • 起始软元件号 + 软元件点数 × 16 - 1 ≦ 最大软元件号
(2) 指令 WR、QR 为字单位。对 X40~X5F 的 32 点进行读取的情况下软元件点数的指定为“02”(将 16 点指定为 1)。

备注

指定软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(2) 对字软元件存储器进行读取的情况下



要点

- (1) 对字软元件存储器进行读取时，软元件点数的指定范围应在满足以下条件的基础上进行指定。
 - $1 \leq \text{软元件点数} \leq 64$
 - $\text{起始软元件号} + \text{软元件点数} - 1 \leq \text{最大软元件号}$
- (2) 指令 WR、QR 为字单位。对 T123~T124 的当前值进行读取的情况下软元件点数的指定为“02”(将 1 点指定为 1)。

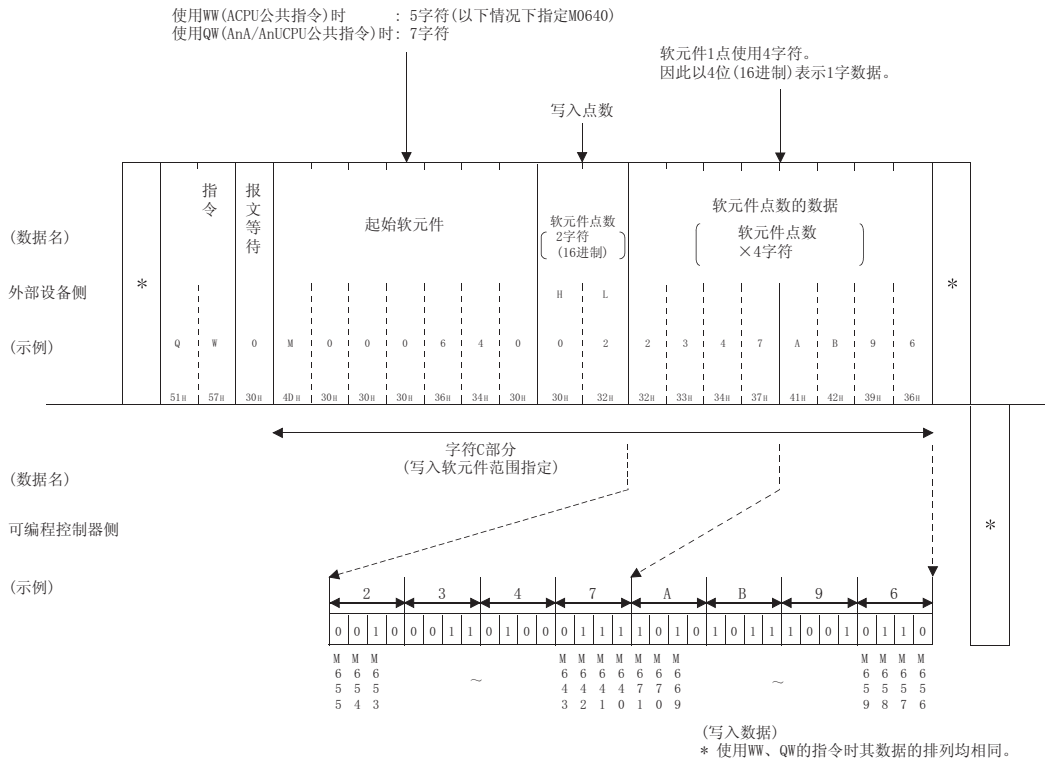
5.2.6 字单位的批量写入(指令：WW、QW)

以下对通过 WW、QW 指令进行位软元件存储器(16 点单位)的批量写入以及字软元件存储器的批量写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (写入条件)
- 1) 报文等待为 0ms。
 - 2) 向 M640~M671 写入 32 点，向 D0~D1 写入 2 点(均为 2 字)。

(1) 对位软元件存储器进行写入的情况下

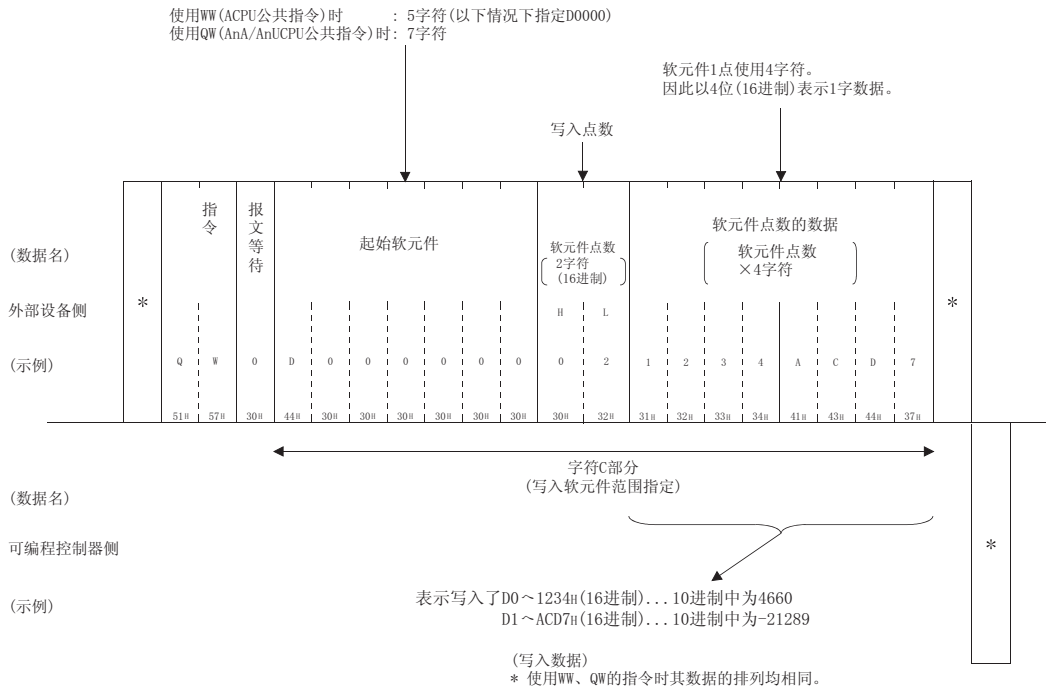


要点
(1) 对位软元件存储器进行写入时，软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1 ≦ 软元件点数 ≦ 10 • 起始软元件号 + 软元件点数 × 16 - 1 ≦ 最大软元件号
(2) 指令 WW、QW 为字单位。写入 M640~M671 的 32 点的情况下软元件点数的指定为“02”(将 16 点指定为 1)。

备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(2) 对字软元件存储器进行写入的情况下



要点

- (1) 对字软元件存储器进行写入时，软元件点数的指定范围应在满足以下条件的基础上进行指定。
 - $1 \leq \text{软元件点数} \leq 64$
 - $\text{起始软元件号} + \text{软元件点数} - 1 \leq \text{最大软元件号}$
- (2) 指令 WW、QW 为字单位。对 D0~D1 的 2 点进行写入的情况下软元件点数的指定为“02”(将 1 点指定为 1)。

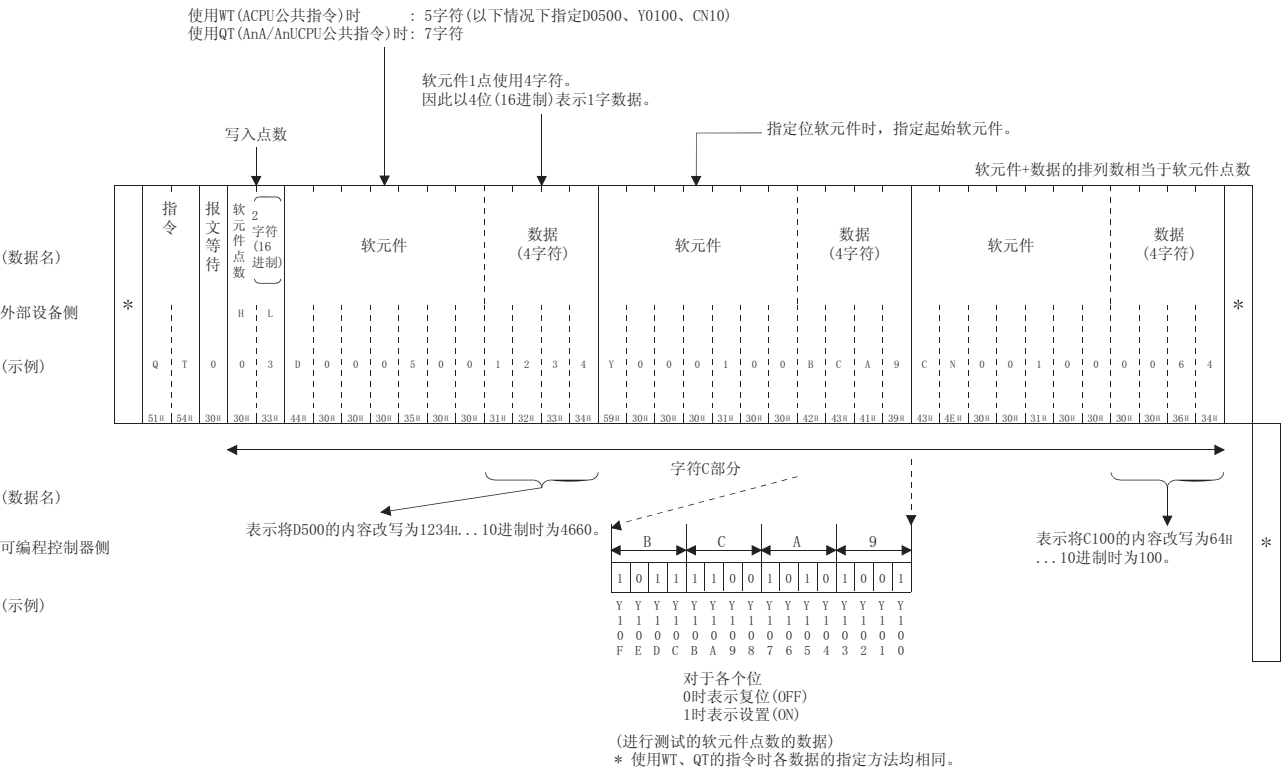
5.2.7 字单位的测试(随机写入)(指令：WT、QT)

以下对通过 WT、QT 指令随机指定字软元件存储器以及位软元件存储器(16 点单位)进行写入的控制步骤进行举例说明。
可以对字软元件与位软元件(16 点单位)进行混合指定。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

(写入条件)

- 1) 报文等待为 0ms。
- 2) 将 D500 置为 1234H、将 Y100~Y10F 置为 BCA9H、将 C100 的当前值置为 64H 的 3 点(3 字)的测试。



要点

软元件点数的指定范围应在满足以下条件的基础上进行指定。

- 1 ≤ 软元件点数 ≤ 10 (位软元件时 10 (将 16 点指定为 1))

备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数 (10 进制数的情况下为 0、16、...)。

5.2.8 软元件存储器的监视

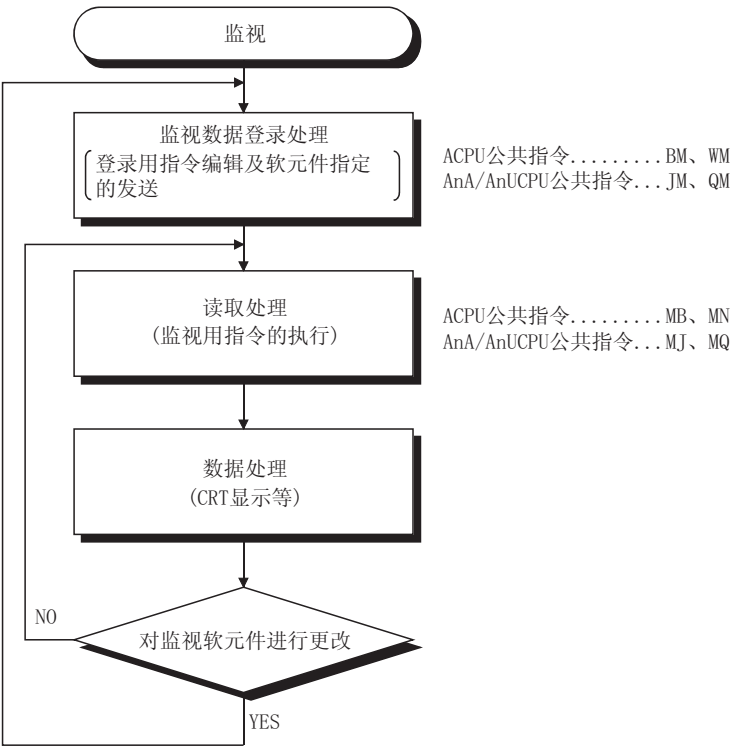
将希望通过外部设备进行监视的软元件以及编号预先登录到 C24 中的功能称为监视数据登录。

此外，将进行了监视登录的软元件的数据内容通过可编程控制器 CPU 进行读取，通过外部设备进行处理的功能称为监视。

通过批量读取 (BR • WR/JR • QR) 进行读取时软元件编号为连续的编号，如果使用本功能可以随机指定编号进行监视。

以下对进行监视时的控制步骤以及将希望监视的软元件以及编号登录到 C24 中的控制步骤进行举例说明。

(1) 监视的步骤



要点
(1) 如上述步骤所示，执行监视的情况下必须进行监视数据登录。如果在不进行监视数据登录的状况下执行监视，将变为协议出错状态。
(2) 监视数据登录的内容在 C24 的重新启动时将丢失。
(3) 可以对位单位 (BM 或 JM)、字单位 (WM 或 QM)、扩展文件寄存器 (EM) 的各指令将各指定软元件预先登录到 C24 中。
(4) 从多个 C24 对同一站的可编程控制器 CPU 的软元件进行监视数据登录时，登录数据将被覆盖，因此最后登录的软元件存储器将有效。
(5) 关于扩展文件寄存器的监视，请参阅 5.3.9 项。

(2) 软元件存储器的监视数据登录(指令：BM、JM、WM、QM)

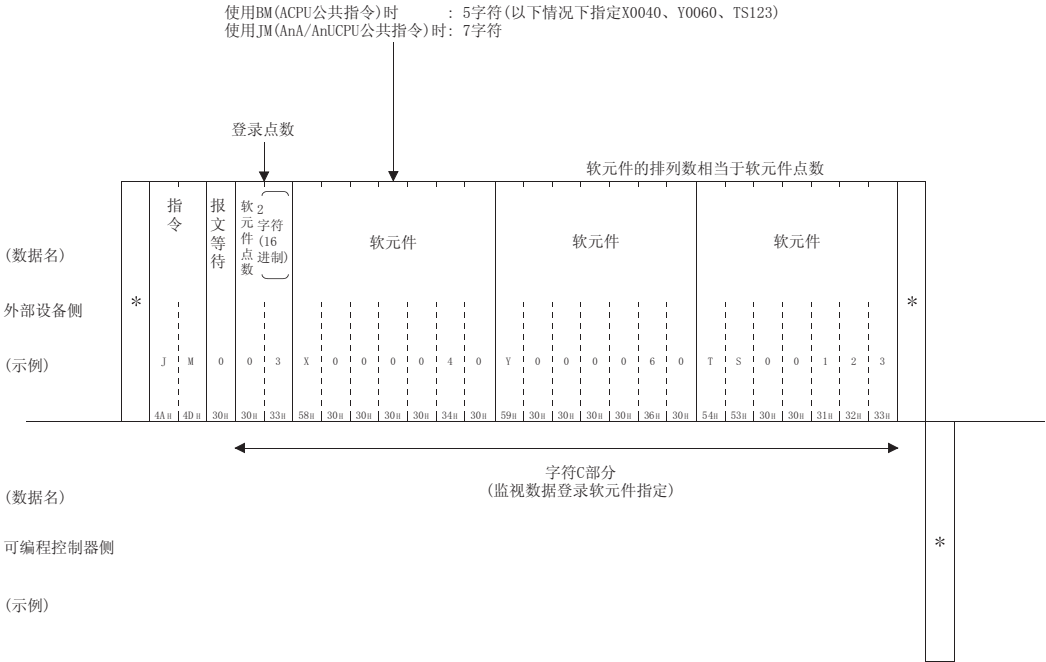
以下对将希望监视的软元件以及编号通过 BM/JM/WM/QM 指令登录到 C24 中的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

(a) 将位软元件存储器以位单位进行监视数据登录的情况下

[控制步骤]

(监视数据登录条件)

- 1) 报文等待为 0ms。
- 2) 对 X40、Y060、T123 的 3 点(3 位)触点进行监视数据登录。



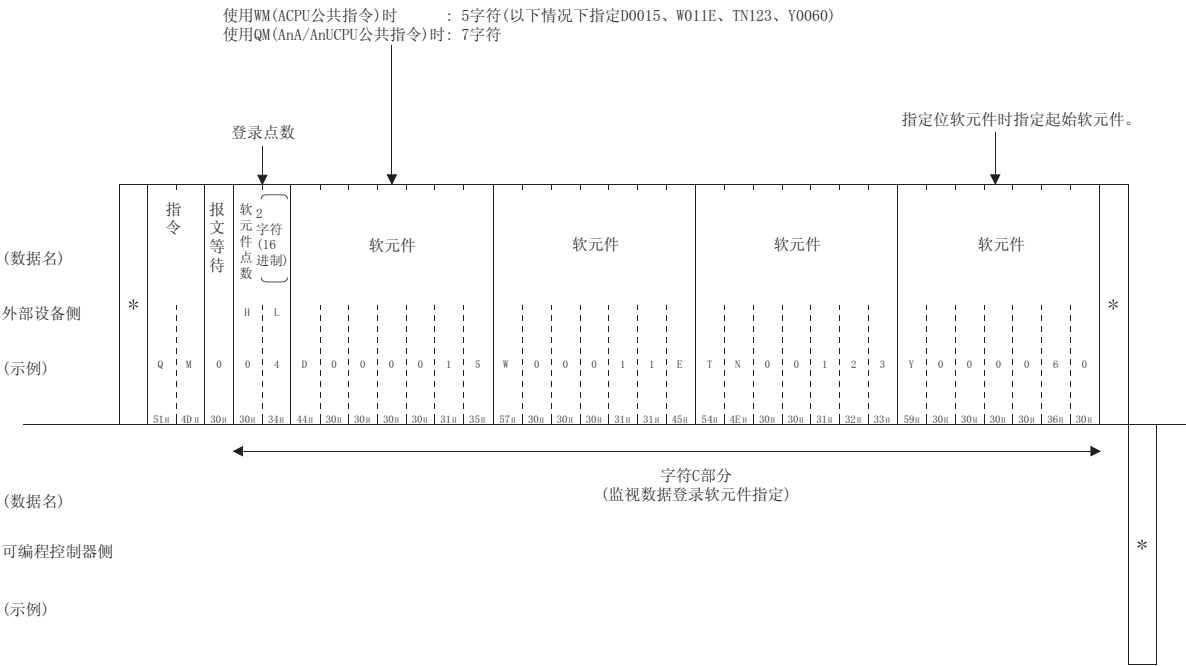
要点
(1) 软元件点数的指定范围应在满足以下条件的基础上进行指定。 此外，使用 BM 指令时，进行访问的可编程控制器 CPU 为除 AnACPU、AnUCPU、QnACPU、LCPU、QCPU 以外的情况下，每 1 点的软元件 X(输入)将变为 2 点的处理点数。 • 1≦软元件点数≦40
(2) 对于以位单位登录的软元件存储器的监视，应按本项 (3) (a) 的说明进行。

(b) 将位软元件存储器以及字软元件存储器以字单位进行监视数据登录的情况下

[控制步骤]

(监视数据登录条件)

- 1) 报文等待为 0ms。
- 2) 对 D15、W11E、T123 的当前值以及 Y060~Y06F 的 4 点(4 字)进行监视数据登录。



要点
(1) 软元件点数的指定范围应在满足以下条件的基础上进行指定。 此外，使用 WM 指令时，访问的可编程控制器 CPU 为除 A3HCPU、AnACPU、AnUCPU、QnACPU、LCPU、QCPU 以外的情况下，每 1 点的软元件 X(输入)将变为 2 点的处理点数。 • 1≦软元件点数≦20 (2) 在字单位的监视登录中如上图所示，可以对字软元件、位软元件(16 点单位)进行混合指定。 (3) 对于以字单位进行了登录的软元件存储器的监视，应按照本项(3) (b)的说明进行操作。

备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(3) 进行了监视数据登录的软元件存储器的监视(指令：MB、MJ、MN、MQ)

以下对通过上述(2)监视数据登录登录到 C24 中的软元件通过 MB/MJ/MN/MQ 指令进行监视的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

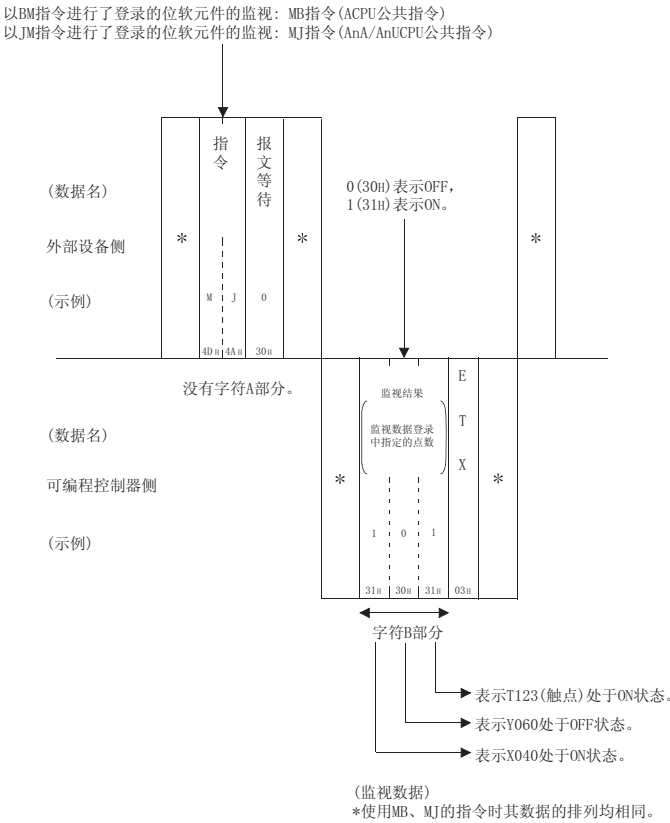
(a) 对以位单位进行了监视数据登录的位软元件存储器进行监视的情况下

[控制步骤]

对于以 BM 指令进行了登录的位软元件存储器，通过 MB 指令进行监视。
对于以 JM 指令进行了登录的位软元件存储器，通过 MJ 指令进行监视。

(监视条件)

- 1) 报文等待为 0ms。
- 2) 对进行了监视数据登录的 X040、Y060、T123 的触点的 3 点(3 位)进行监视。



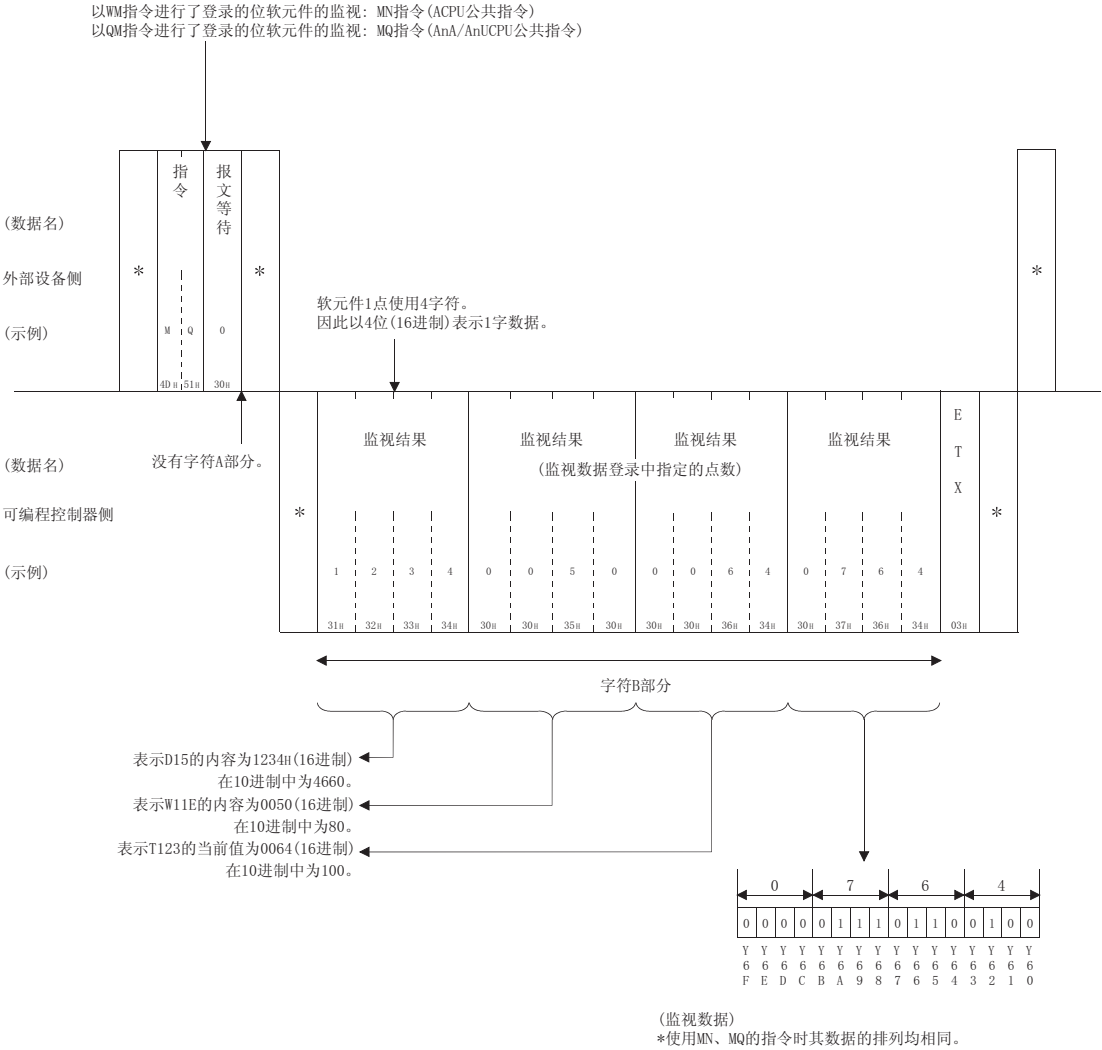
(b) 对以字单位进行了监视数据登录的位软元件存储器以及字软元件存储器进行监视的情况下

[控制步骤]

对于以 WM 指令进行了登录的软元件存储器，通过 MN 指令进行监视。
对于以 QM 指令进行了登录的软元件存储器，通过 MQ 指令进行监视。

(监视条件)

- 1) 报文等待为 0ms。
- 2) 对进行了监视数据登录的 D15、W11E、T123 的当前值以及 Y060~Y06F 的 4 点(4 字)进行监视。



5.3 扩展文件寄存器的读取、写入

扩展文件寄存器是指，将可编程控制器 CPU 的用户存储器区域的空余区域作为文件寄存器使用的寄存器，是使用扩展文件寄存器用软件包“SWOGHP-UTLPC-FN1、SWOSRX-FNUP”（以下略称为 UTLPC-FN1、FNUP。）进行各种数据处理以及 AnACPU、AnUCPU 的扩展文件寄存器用专用指令中，用于存储必要数据以及运算结果的存储器区域。
以下对进行扩展文件寄存器的读取、写入等的控制步骤进行举例说明。

5.3.1 ACPU 公共指令及地址

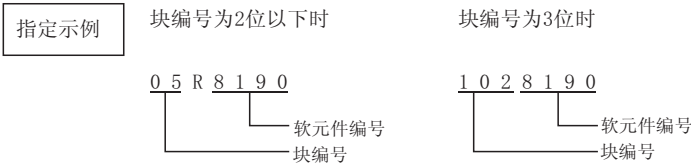
(1) 用于扩展文件寄存器的读取、写入等的 ACPU 公共指令如下所示。

项目	指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
	符号	ASCII 代码			STOP 中	RUN 中		
						写入允许设置	写入禁止设置	
批量读取	ER	45H, 52H	对扩展文件寄存器(R)以 1 点为单位进行读取。	64 点	○	○	○	5.3.4 项
批量写入	EW	45H, 57H	对扩展文件寄存器(R)以 1 点为单位进行写入。	64 点	○	○	×	5.3.5 项
测试 (随机写入)	ET	45H, 54H	对扩展文件寄存器(R)以 1 点为单位随机指定块编号・软元件编号进行写入。	10 点	○	○	×	5.3.8 项
监视数据 登录	EM	45H, 4DH	对进行监视的软元件编号以 1 点为单位进行登录。	20 点	○	○	○	5.3.9 项(2)
监视	ME	4DH, 45H	对进行了监视数据登录的扩展文件寄存器进行监视。	-	○	○	○	5.3.9 项(3)

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 扩展文件寄存器的地址

- (a) 扩展文件寄存器中，有 0~n(n 根据存储卡盒而有所不同)号块，0 号块中有通过可编程控制器 CPU 参数指定的点数的寄存器，块编号 1~n 的各块中有 8192 点的寄存器。
但是，可通过可编程控制器 CPU 进行读取、写入的范围为 0 号块的参数中指定的点数的范围。
- (b) 可指定的块号范围根据存储卡盒的类型以及可编程控制器 CPU 的参数设置而变化。
详细内容请参阅 UTLPC-FN1 或 FNUP 的操作手册或 AnACPU、AnUCPU 的用户手册。
- (c) 进行地址指定时，将块编号及软元件编号以 7 字符进行指定。
- 块编号为 2 位以下的情况下
“块编号(2 位)” + “R” + “软元件编号(4 位)”
 - 块编号为 3 位的情况下
“块编号(3 位)” + “软元件编号(4 位)”



5.3.2 AnA/AnUCPU 公共指令及软元件编号

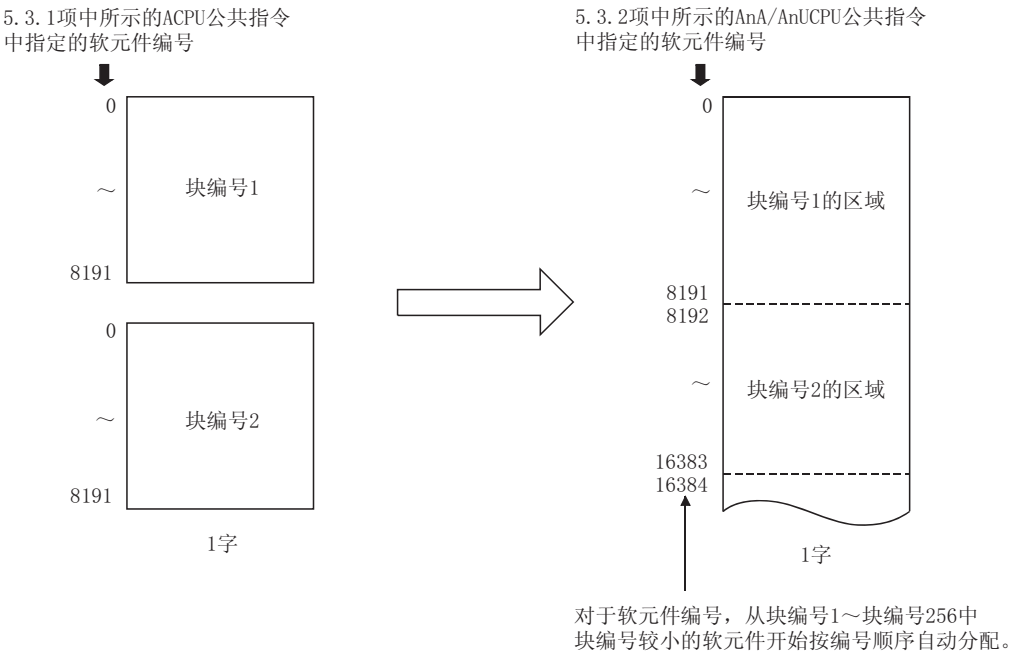
- (1) 扩展文件寄存器的直接读取、直接写入中使用的 AnA/AnUCPU 公共指令如下所示。
该指令的功能为，在对块编号 1~256 的扩展文件寄存器进行访问时无需在意各块编号，将从块编号 1 的软元件编号 0 开始的地址指定为软元件编号进行访问。
(对可使用的块的个数×8192 点的扩展文件寄存器以连续的软元件编号进行访问。)

项目	指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
	符号	ASCII 代码			STOP 中	RUN 中		
						写入允许设置	写入禁止设置	
直接读取	NR	4E _H , 52 _H	通过对扩展文件寄存器的软元件编号进行连号指定以 1 点(1 字)为单位进行读取。	64 点	○	○	○	5.3.6 项
直接写入	NW	4E _H , 57 _H	通过对扩展文件寄存器的软元件编号进行连号指定以 1 点(1 字)为单位进行写入。	64 点	○	○	×	5.3.7 项

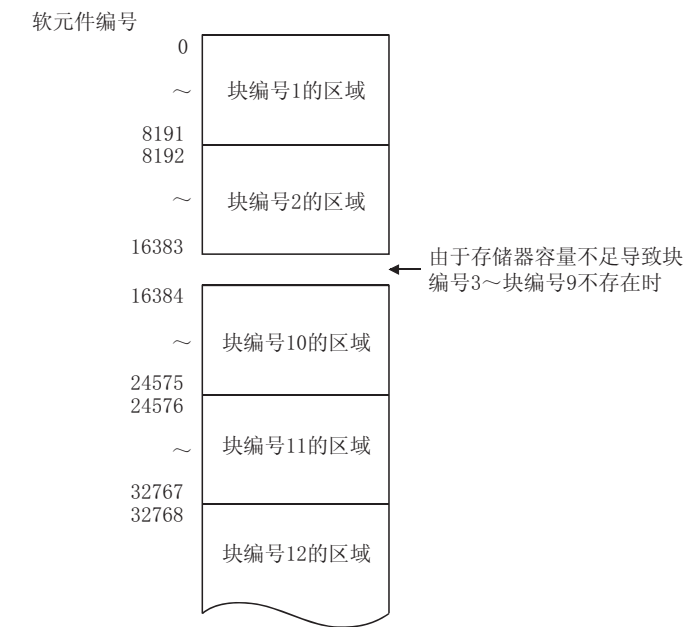
上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 扩展文件寄存器的软元件编号

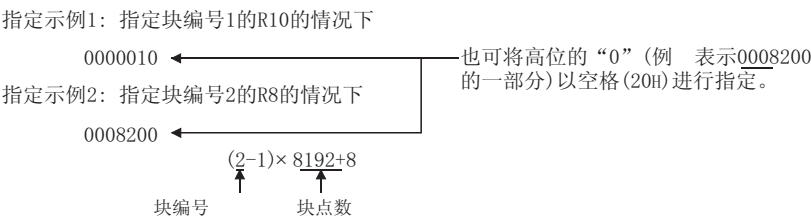
- (a) 可指定的软元件编号的范围如下所示。
0~(可使用的块数×8192)-1



此外，可指定的软元件编号根据存储卡盒的类型以及可编程控制器 CPU 的参数设置而变化。
(详细内容请参阅 UTLP-FN1 或 FNUP 的操作手册或访问目标可编程控制器 CPU 的用户手册。)
对于存储卡盒内不存在的块编号，不进行软元件编号分配。如下所示，将跳过存储卡盒内不存在的块编号进行软元件编号的自动分配。



(b) 软元件编号以 7 个字符进行指定。



要点	
(1)	对于 AnA/AnUCPU 公共指令 NR・NW，只有在对块编号 1~256 的扩展文件寄存器进行数据读写时才可以使用。 此外，可在与参数的文件寄存器设置的有无无关的状况下使用。
(2)	对参数中设置的文件寄存器(R)进行访问时，或指定块编号进行访问时，应使用 5.3.1 项中所示的指令。
(3)	AnA/AnUCPU 公共指令 NR・NW 中指定的起始软元件编号的计算公式如下所示。 (指定从起始开始的第 n(1 以上)块的软元件编号 m(0~8191)的情况下) 起始软元件编号 = (n - 1) × 8192 + m

备注

将使用 AnA/AnUCPU 公共指令 NR・NW 时指定的软元件编号范围分为 28 块，各块如下所示。

软元件编号	对象块的位置	软元件编号	对象块的位置
0 ～ 8191	第 1 块 R0 ～ R8191	114688 ～ 122879	第 15 块 R0 ～ R8191
8192 ～ 16383	第 2 块 R0 ～ R8191	122880 ～ 131071	第 16 块 R0 ～ R8191
16384 ～ 24575	第 3 块 R0 ～ R8191	131072 ～ 139263	第 17 块 R0 ～ R8191
24576 ～ 32767	第 4 块 R0 ～ R8191	139264 ～ 147455	第 18 块 R0 ～ R8191
32768 ～ 40959	第 5 块 R0 ～ R8191	147456 ～ 155647	第 19 块 R0 ～ R8191
40960 ～ 49151	第 6 块 R0 ～ R8191	155648 ～ 163839	第 20 块 R0 ～ R8191
49152 ～ 57343	第 7 块 R0 ～ R8191	163840 ～ 172031	第 21 块 R0 ～ R8191
57344 ～ 65535	第 8 块 R0 ～ R8191	172032 ～ 180223	第 22 块 R0 ～ R8191
65536 ～ 73727	第 9 块 R0 ～ R8191	180224 ～ 188415	第 23 块 R0 ～ R8191
73728 ～ 81919	第 10 块 R0 ～ R8191	188416 ～ 196607	第 24 块 R0 ～ R8191
81920 ～ 90111	第 11 块 R0 ～ R8191	196608 ～ 204799	第 25 块 R0 ～ R8191
90112 ～ 98303	第 12 块 R0 ～ R8191	204800 ～ 212991	第 26 块 R0 ～ R8191
98304 ～ 106495	第 13 块 R0 ～ R8191	212992 ～ 221183	第 27 块 R0 ～ R8191
106496 ～ 114687	第 14 块 R0 ～ R8191	221184 ～ 229375	第 28 块 R0 ～ R8191

5.3.3 扩展文件寄存器的读取、写入时的注意事项

以下对根据 5.3.4 项～5.3.9 项中所示的指令进行扩展文件寄存器的读取、写入等时的注意事项有关内容进行说明。

- (1) 可以对支持扩展文件寄存器的可编程控制器 CPU 进行访问。
对于不支持扩展文件寄存器的可编程控制器 CPU(A1N 等)，不能使用本功能。
- (2) 即使在指定了不存在的块编号进行了读取、写入的情况下，根据可编程控制器 CPU 中安装的存储卡盒的类型，也可能无法检测出出错(字符部分出错 06h)。在这种情况下，读取的数据将为不正确的数据。此外，如果进行写入，有可能会损坏可编程控制器 CPU 的用户存储器。
应对存储卡盒的类型、参数设置内容等进行确认之后再执行本功能。

存储卡盒型号	可能无法检测出字符部分出错(06h)的块编号		
	A0J2H、A2、A3CPU	A2N、A3NCPU	A3H、AnA、AnUCPU
A3NMCA-12	No. 10 ～ No. 11		
A3NMCA-18	-	No. 10 ～ No. 28	
A3NMCA-24	-	No. 13 ～ No. 20	No. 13 ～ No. 28
A3NMCA-40	-		No. 21 ～ No. 28
A3AMCA-96	-		No. 21 ～ No. 48 (*1)

*1 A3AMCA-96 可用于 A3A、A3U、A4UCPU。

(详细内容请参阅 UTLP-FN1 或 FNUP 的手册或访问目标可编程控制器 CPU 的用户手册。)

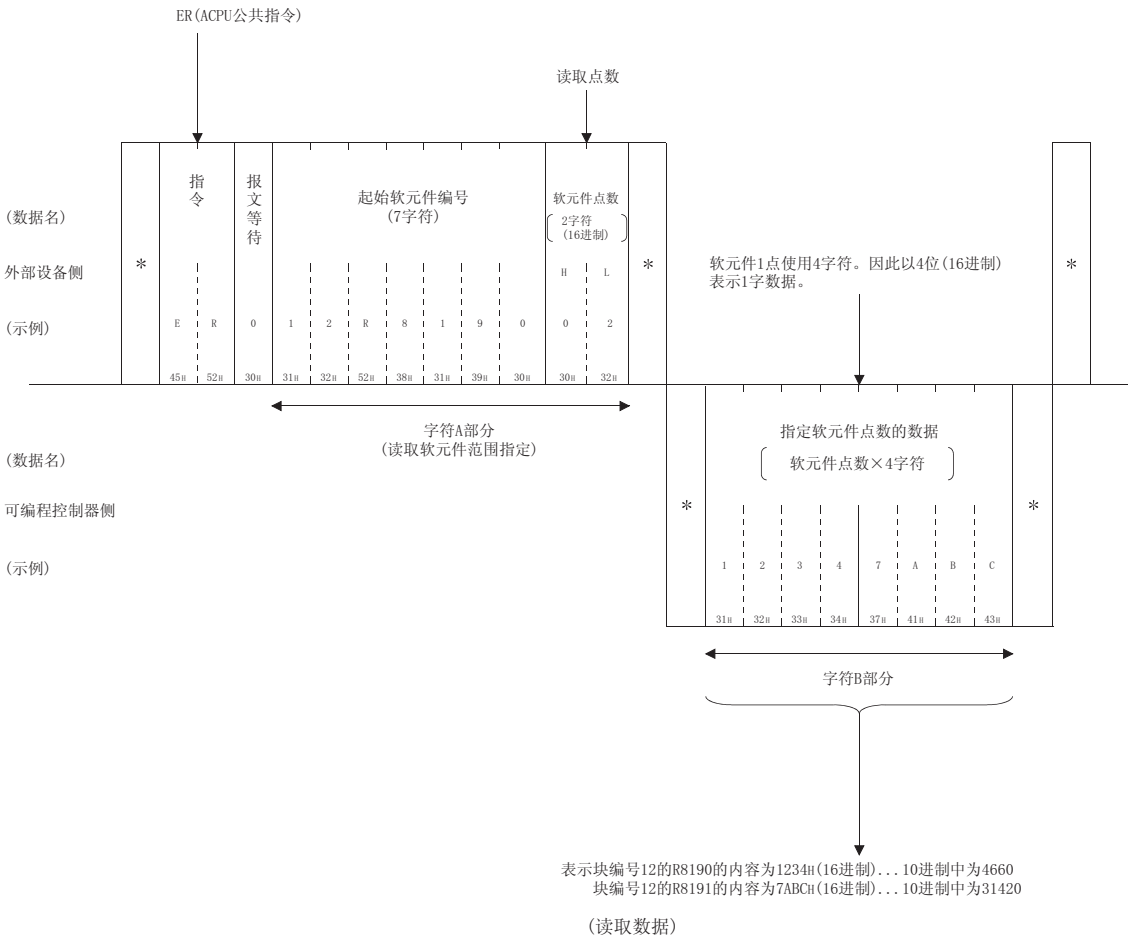
- (3) A2USCPU(S1)中可处理的扩展文件寄存器的块编号如下所示。
 - A2USCPU : No. 1～3
 - A2USCPU-S1 : No. 1～8, No. 10～16
- (4) 不能对 Q/L/QnACPU 的扩展文件寄存器进行读取/写入。

5.3.4 扩展文件寄存器的批量读取(指令：ER)

以下对通过 ER 指令进行扩展文件寄存器的批量读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (读取条件)
- 1) 报文等待为 0ms。
 - 2) 对块编号 12 的 R8190、R8191 的 2 点进行读取。



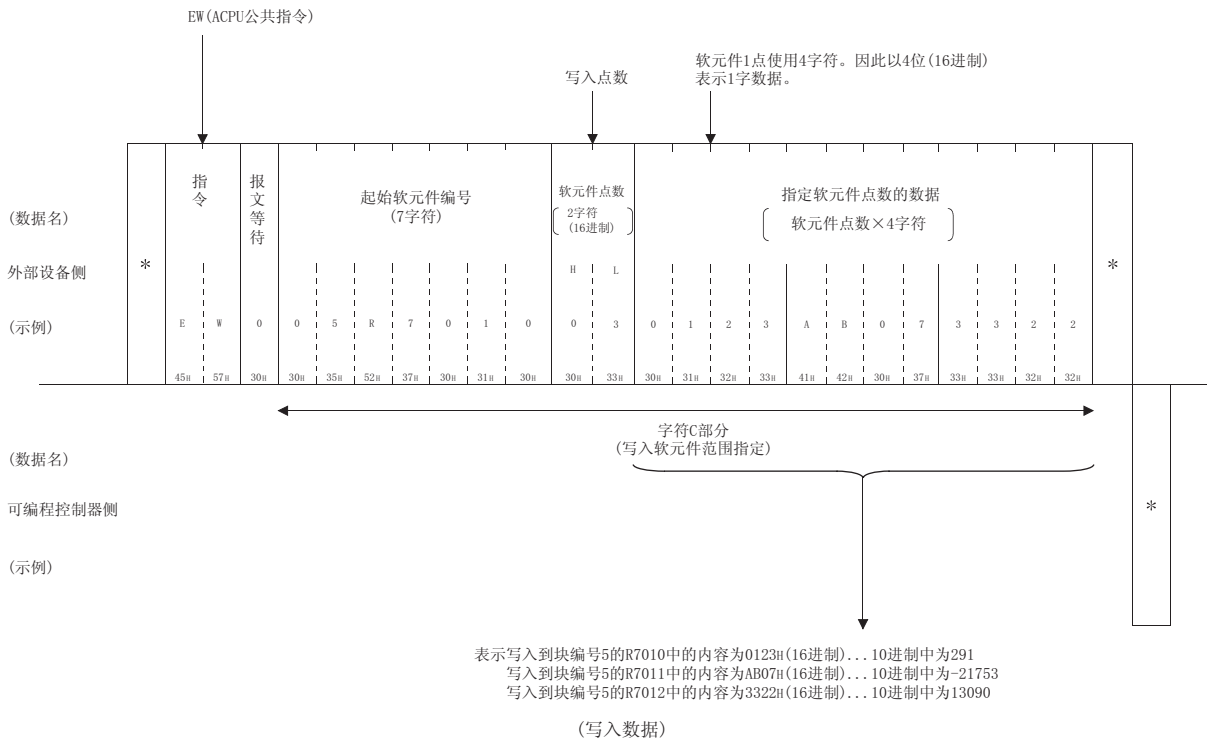
要点
软元件点数的指定范围应在满足以下条件的基础上进行指定。 • $1 \leq \text{软元件点数} \leq 64$ • $\text{起始软元件编号} + \text{软元件点数} - 1 \leq \text{最大软元件编号}$

5.3.5 扩展文件寄存器的批量写入(指令：EW)

以下对通过 EW 指令进行扩展文件寄存器的批量写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (写入条件)
- 1) 报文等待为 0ms。
 - 2) 对块编号 5 的 R7010~R7012 的 3 点进行写入。



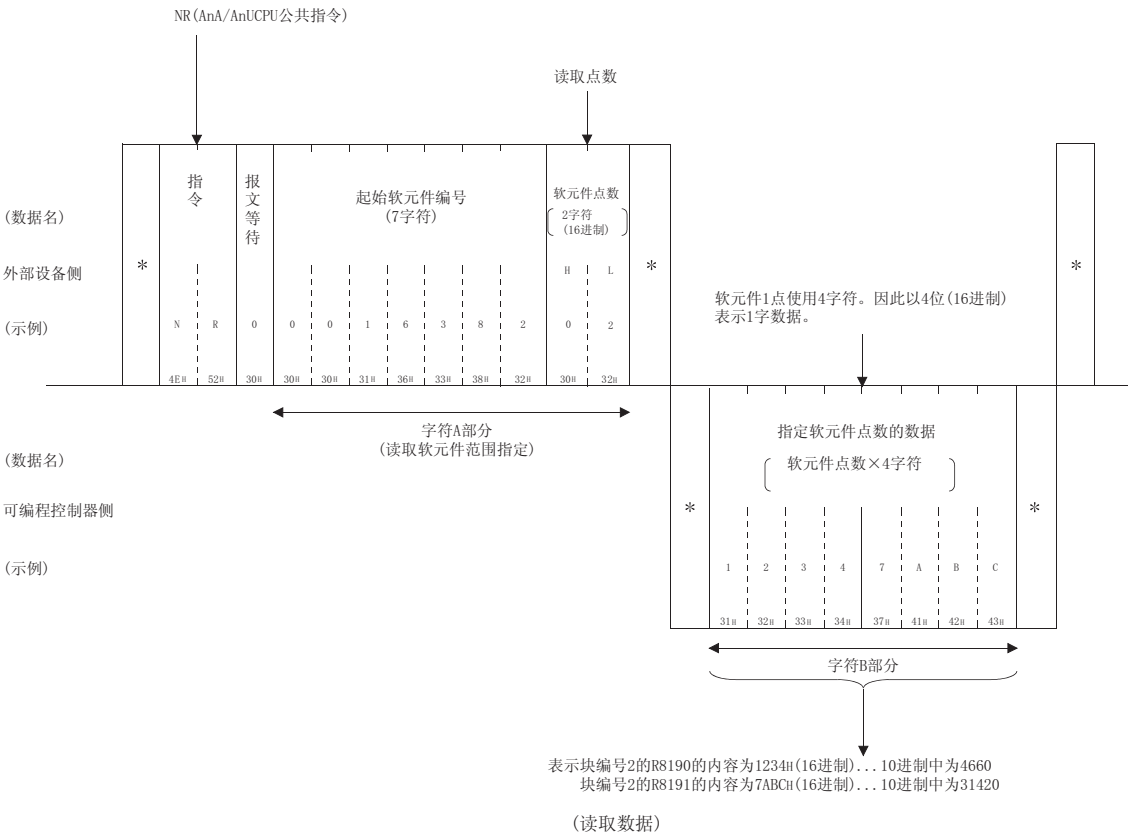
要点
软件元件点数的指定范围应在满足以下条件的基础上进行指定。
• 1≦软件元件点数≦64
• 起始软件元件编号+软件元件点数-1≦最大软件元件编号

5.3.6 扩展文件寄存器的直接读取(指令：NR)

以下对通过 NR 指令进行扩展文件寄存器的直接读取的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (读取条件)
- 1) 报文等待为 0ms。
 - 2) 对块编号 2 的 R8190、R8191 的 2 点进行读取。



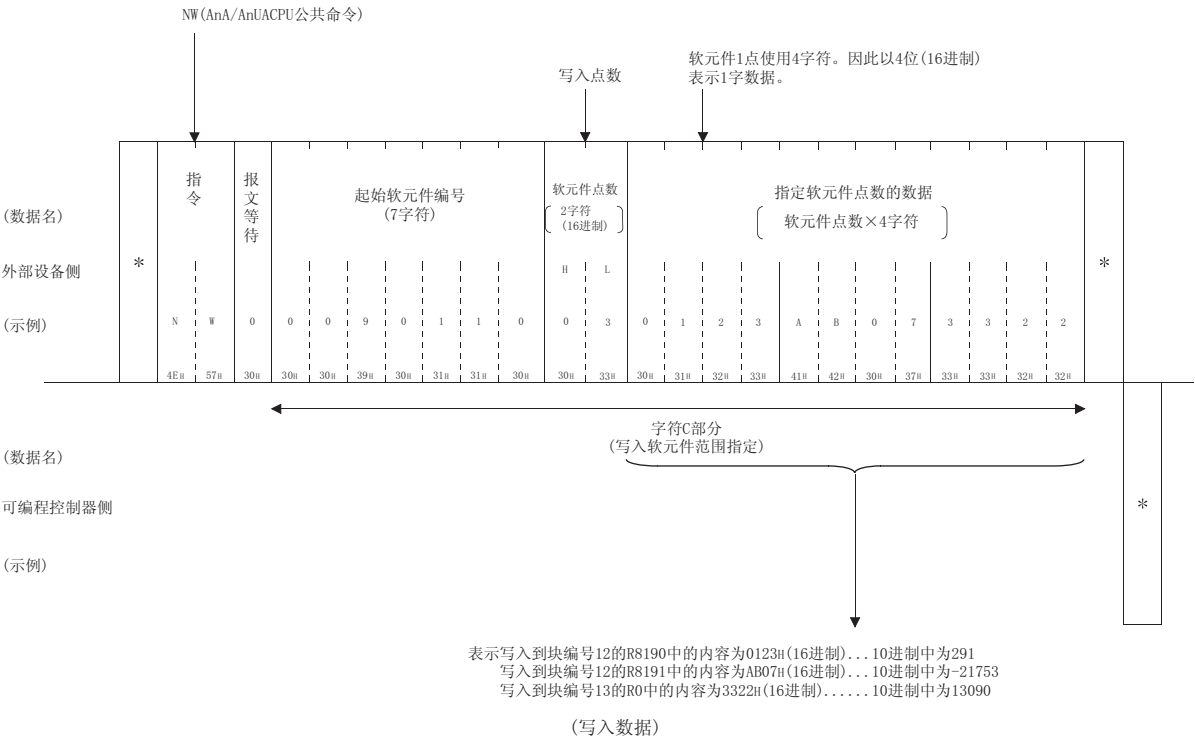
要点
软元件点数的指定范围应在满足以下条件的基础上进行指定。
• 1 ≦ 软元件点数 ≦ 64
• 起始软元件编号+软元件点数-1 ≦ 最大软元件编号

5.3.7 扩展文件寄存器的直接写入(指令：NW)

以下对通过 NW 指令进行扩展文件寄存器的直接写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (写入条件)
- 1) 报文等待为 0ms。
 - 2) 向块编号 12 的 R8190、R8191 以及块编号 13 的 R0 写入 3 点。
(块编号 1~8、块编号 10~13 的扩展文件寄存器存在的情况下)



要点
软元件点数的指定范围应在满足以下条件的基础上进行指定。 • 1≦软元件点数≦64 • 起始软元件编号+软元件点数-1≦最大软元件编号

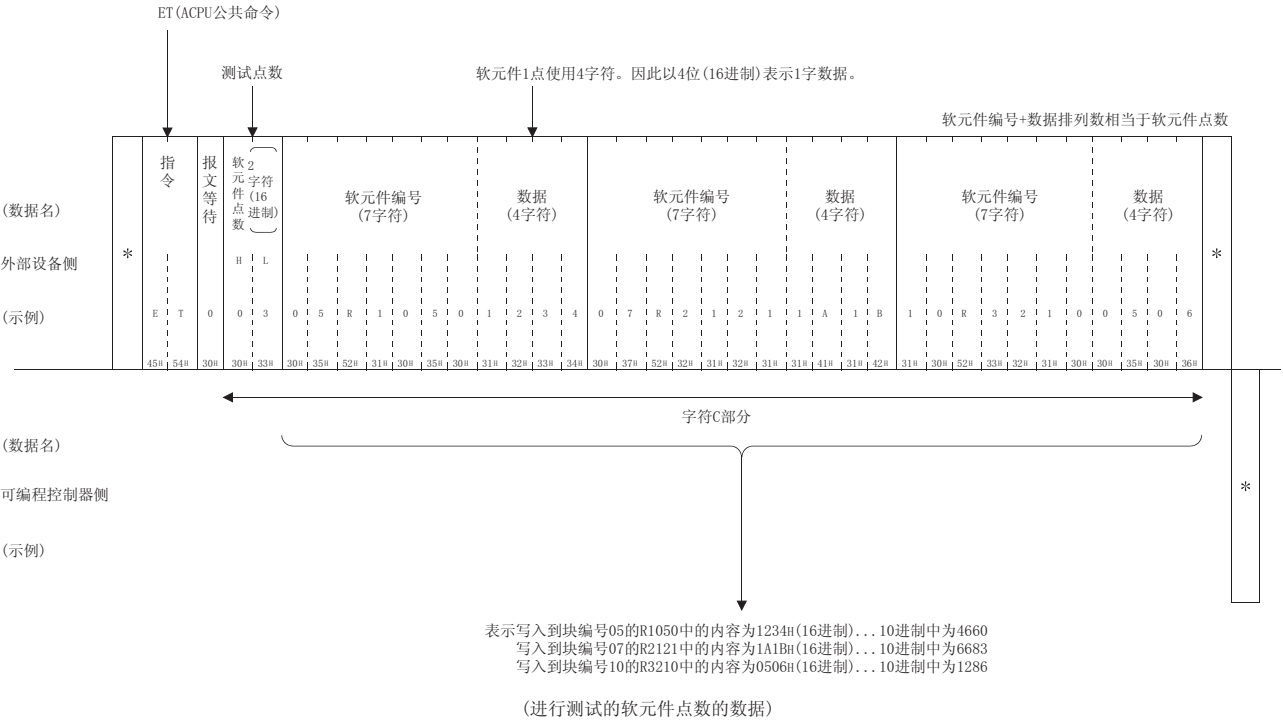
5.3.8 扩展文件寄存器的测试(随机写入)(指令：ET)

以下对通过 ET 指令随机指定扩展文件寄存器的软元件编号进行数据写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

(写入条件)

- 1) 报文等待为 0ms。
- 2) 将块编号 5 的 R1050 置为 R1234H、将块编号 7 的 R2121 置为 1A1BH、将块编号 10 的 R3210 置为 506H 的 3 点(3 字)的测试。
(存在块编号 1~8、块编号 10 的扩展文件寄存器的情况下)



要点

软元件点数的指定范围应在满足以下条件的基础上进行指定。
• 1≦软元件点数≦10

5.3.9 扩展文件寄存器的监视

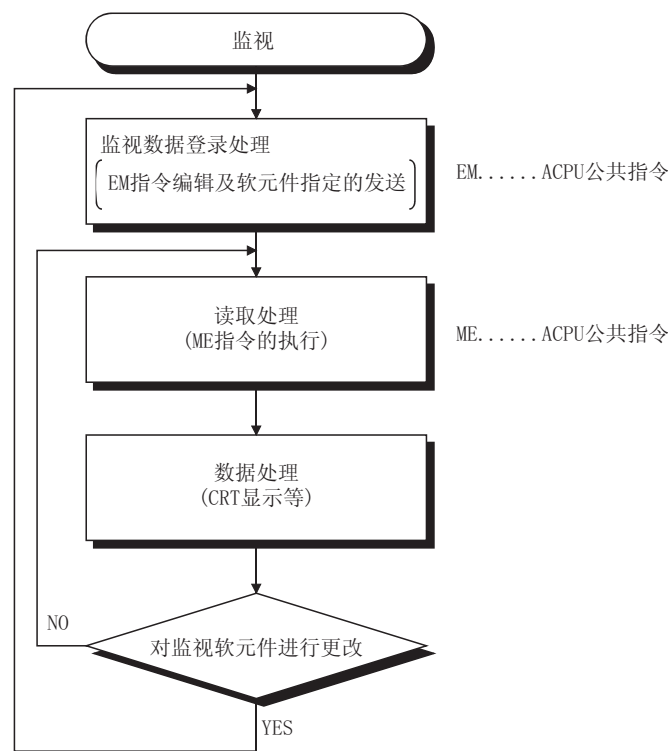
监视数据登录功能是指，将希望通过外部设备进行监视的软元件以及编号预先登录到 C24 中的功能。

此外，监视功能是指，可以将监视数据登录的软元件的数据内容通过可编程控制器 CPU 进行读取，通过外部设备进行处理的功能。

通过批量读取 (ER) 或直接读取 (NR) 进行读取时软元件编号为连续的编号，而使用本功能时可以随机指定编号进行监视。

以下对进行监视时的控制步骤以及将希望监视的软元件以及编号登录到 C24 中的控制步骤进行举例说明。

(1) 监视步骤



要点	
(1)	如上述步骤所示，执行监视的情况下必须进行监视数据登录。如果在不进行监视数据登录的状况下执行监视，将变为协议出错状态。
(2)	监视数据登录的内容在 C24 的重新启动时将丢失。
(3)	进行监视数据登录时可以对扩展文件寄存器 (EM)、软元件存储器・位单位 (BM 或 JM)、字单位 (WM 或 QM) 的各指令预先登录 5 种类型。
(4)	从多个外部设备对同一站的可编程控制器 CPU 的软元件存储器进行监视数据登录时，登录数据将被覆盖，因此最后登录的软元件存储器将有效。 关于软元件存储器的监视，请参阅 5.2.8 项。

(2) 扩展文件寄存器的监视数据登录(指令：EM)

以下对将进行监视的扩展文件寄存器的软件元件编号通过 EM 指令登录到 C24 中的控制步骤进行举例说明。

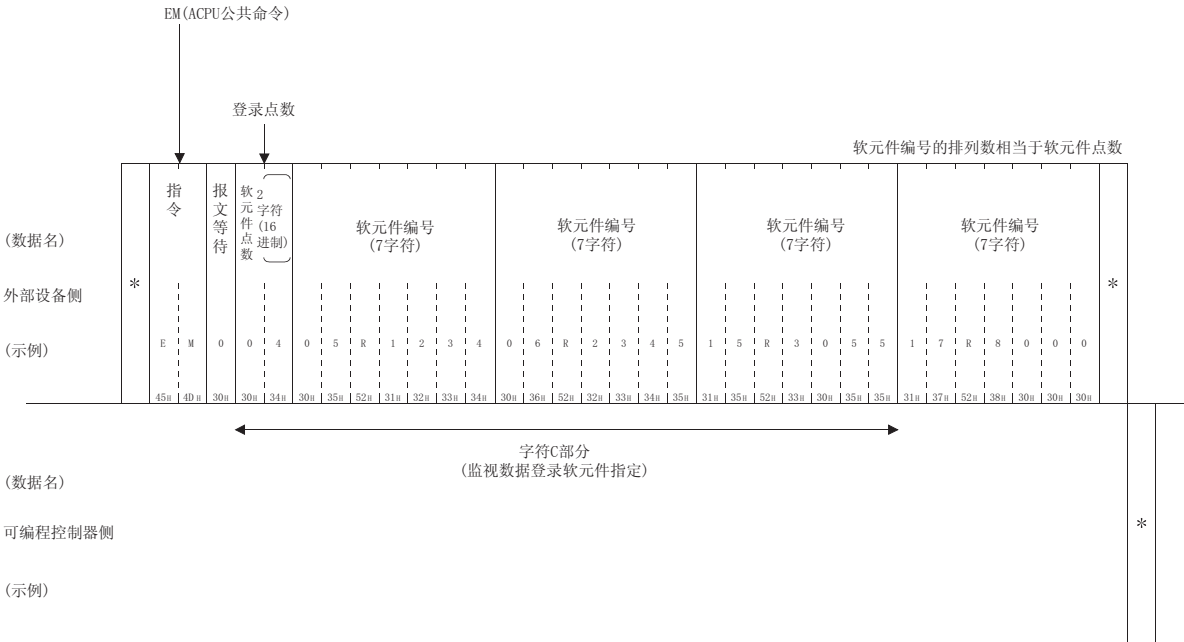
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。

请参阅 5.1 节中所示的详细说明。

[控制步骤]

(监视数据登录条件)

- 1) 报文等待为 0ms。
 - 2) 对块编号 5 的 R1234、块编号 6 的 R2345、块编号 15 的 R3055、块编号 17 的 R8000 的 4 点(4 字)进行监视数据登录。
- (存在块编号 1~8、块编号 10~17 的扩展文件寄存器的情况下)



要点

- (1) 软元件点数的指定范围应在满足以下条件的基础上进行指定。
 - $1 \leq \text{软元件点数} \leq 20$
- (2) 对于通过 EM 指令登录的扩展文件寄存器的监视，应根据本项(3)的说明进行操作。

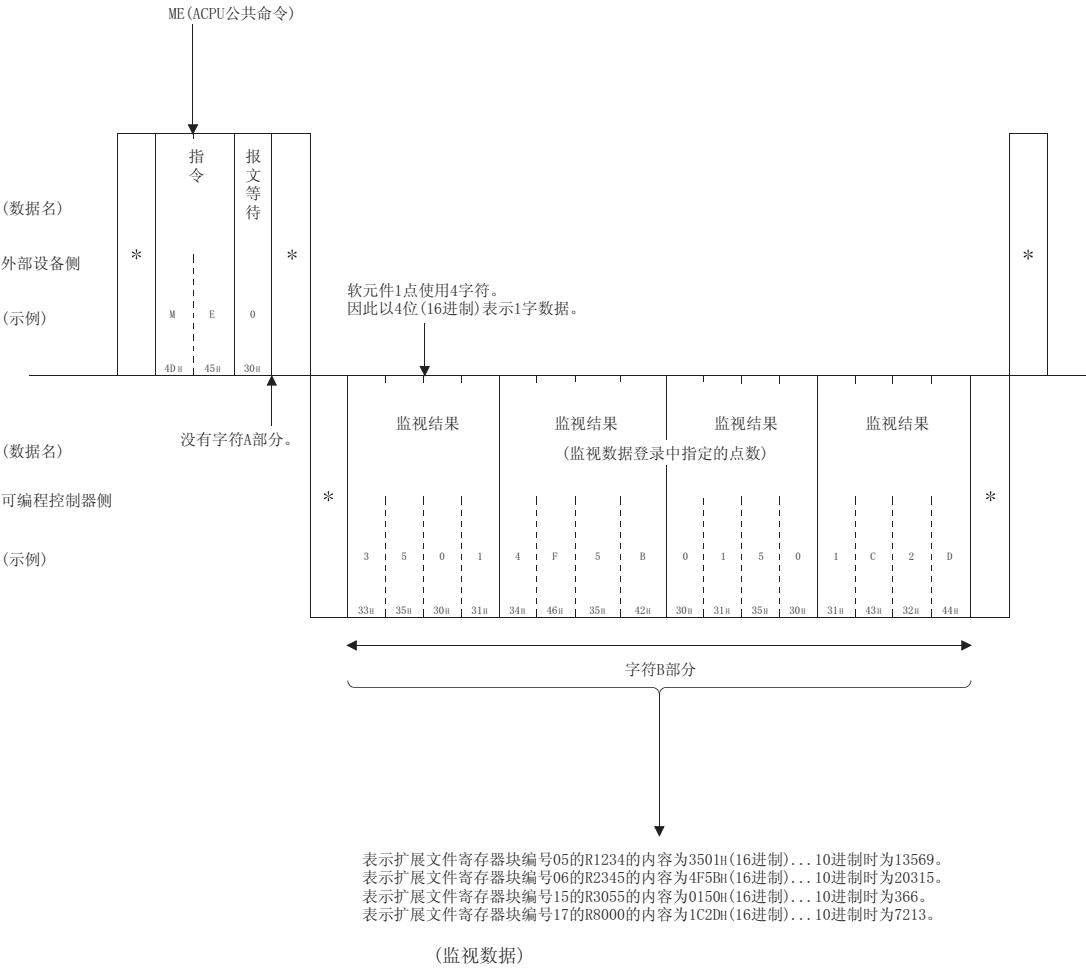
(3) 扩展文件寄存器的监视(指令：ME)

以下对通过上述(2)的监视数据登录(EM)登录到 C24 中的扩展文件寄存器通过 ME 指令进行监视的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

(监视条件)

- 1) 报文等待为 0ms。
- 2) 对进行了监视数据登录的块编号 5 的 R1234、块编号 6 的 R2345、块编号 15 的 R3055、块编号 17 的 R8000 的 4 点(4 字)进行监视。
(存在块编号 1~8、块编号 10~17 的扩展文件寄存器的情况下)



5.4 智能功能模块的缓冲存储器的读取、写入

以下举例说明对 A 系列可编程控制器 CPU 站或 MELSECNET/H、MELSECNET/10 远程 I/O 站的智能功能模块(特殊功能模块)的缓冲存储器进行数据读取、写入时的控制步骤。
在本指令中对智能功能模块的缓冲存储器以字节单位进行访问。

5.4.1 指令及处理内容

(1) ACPU 公共指令

项目	指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态			参阅章节
	符号	ASCII 代码			STOP 中	RUN 中		
						写入允许设置	写入禁止设置	
批量读取	TR	54 _H , 52 _H	对智能功能模块的缓冲存储器内容进行读取。	128 字节 (64 字)	○	○	○	5.4.3 项
批量写入	TW	54 _H , 57 _H	对智能功能模块的缓冲存储器进行数据写入。		○	○	×	5.4.4 项

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 关于可访问的模块及缓冲存储器的地址

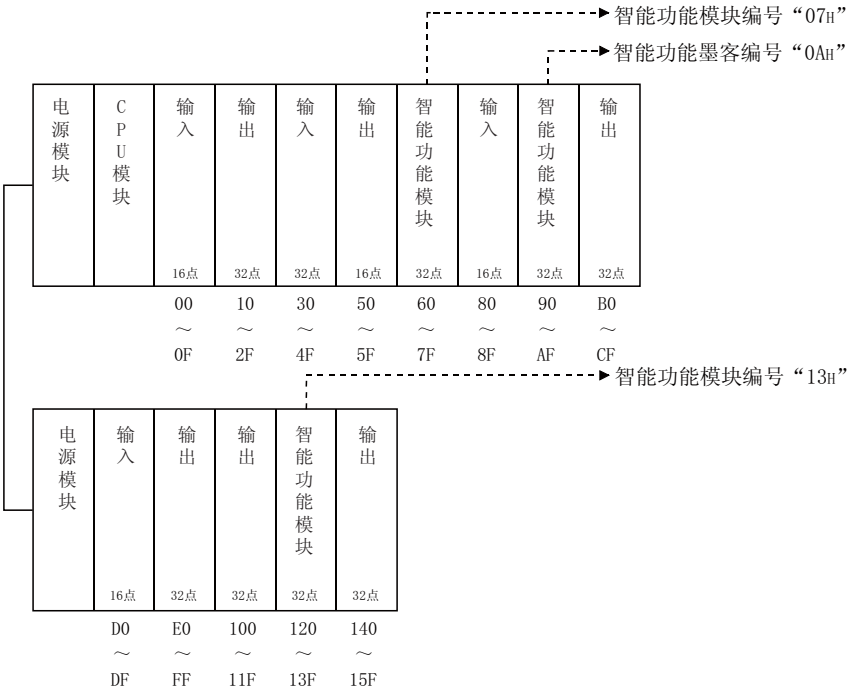
关于通过本功能可访问的 A/QnA 系列的模块型号及指定的缓冲存储器起始地址，
与以 QnA 兼容 3E/3C/4C 帧、4E 帧进行的通信中对智能功能模块进行访问时相同。
请参阅 3.5.1 节及 3.5.2 项。(不能对 Q/L 系列的智能功能模块进行访问。)
在本功能中，与通过 GX Works2 或 GX Configurator-SC 进行的字/字节单位指定无关，以字节单位进行读取、写入。

要点
对于智能功能模块的缓冲存储器，各模块中有可进行读取・写入的区域、只读区域、写入专用区域、用于操作系统的用户使用禁止区域。 应按照各模块的手册的说明执行本功能。 如果进行了错误的读取・写入，可编程控制器 CPU、各智能功能模块中有可能发生出错。

5. 4. 2 控制步骤中智能功能模块编号的思路

(1) 占用 1 个插槽的智能功能模块的模块编号

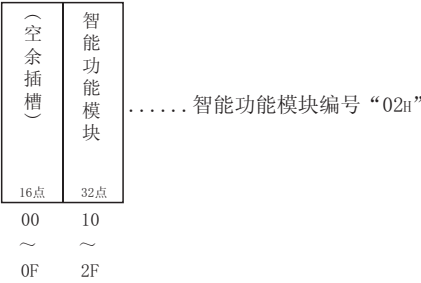
对于控制步骤中指定的智能功能模块编号，为将安装了模块的插槽的输入输出信号(I/O 地址)的最终编号以 3 位表示时的高 2 位。



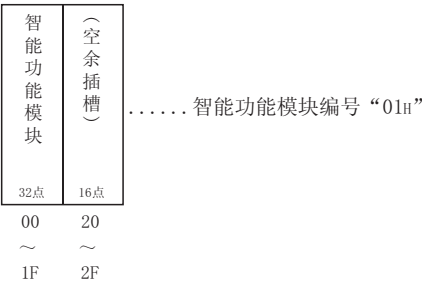
(2) 占用 2 个插槽的智能功能模块的模块编号

对于占用 2 个插槽的智能功能模块，每个模块中各插槽的占用点数是确定的。
对于控制步骤中指定的智能功能模块编号，为将安装了模块的插槽的作为智能功能模块分配的插槽侧的输入输出信号(I/O 地址)的最终编号以 3 位表示时的高 2 位。
关于各模块的插槽的分配有关内容，请参阅相应模块的手册。

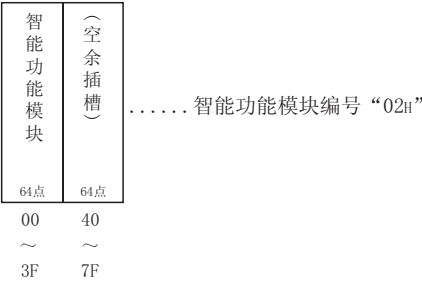
- 1) 将前半部分插槽分配为空余插槽的模块的情况下。
(AD72、A84AD 等)



- 2) 将后半部分插槽分配为空余插槽的模块的情况下。
(A61LS 等)



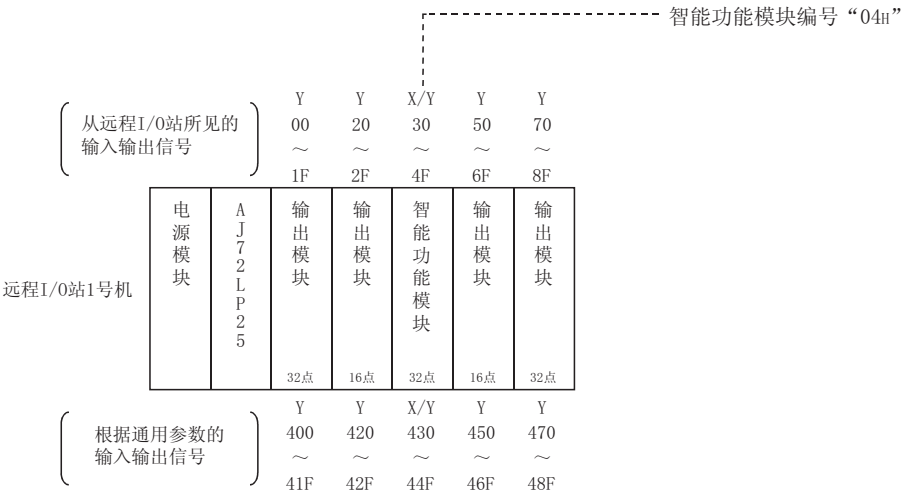
- 3) 智能功能模块分配与输入输出分配混合的模块的情况下。
(A81CPU 的情况下)



(3) MELSECNET/H、MELSECNET/10 远程 I/O 站的智能功能模块的模块编号

对于 MELSECNET/H、MELSECNET/10 远程 I/O 站的智能功能模块的模块编号，为将
所有下述“从远程 I/O 站所见的输入输出信号”的最终编号以 3 位表示时的高 2
位。

与 MELSECNET/H、MELSECNET/10 远程 I/O 网络的主站中设置的通用参数的内容无
关，应以“从远程 I/O 站所见的输入输出信号”进行指定。

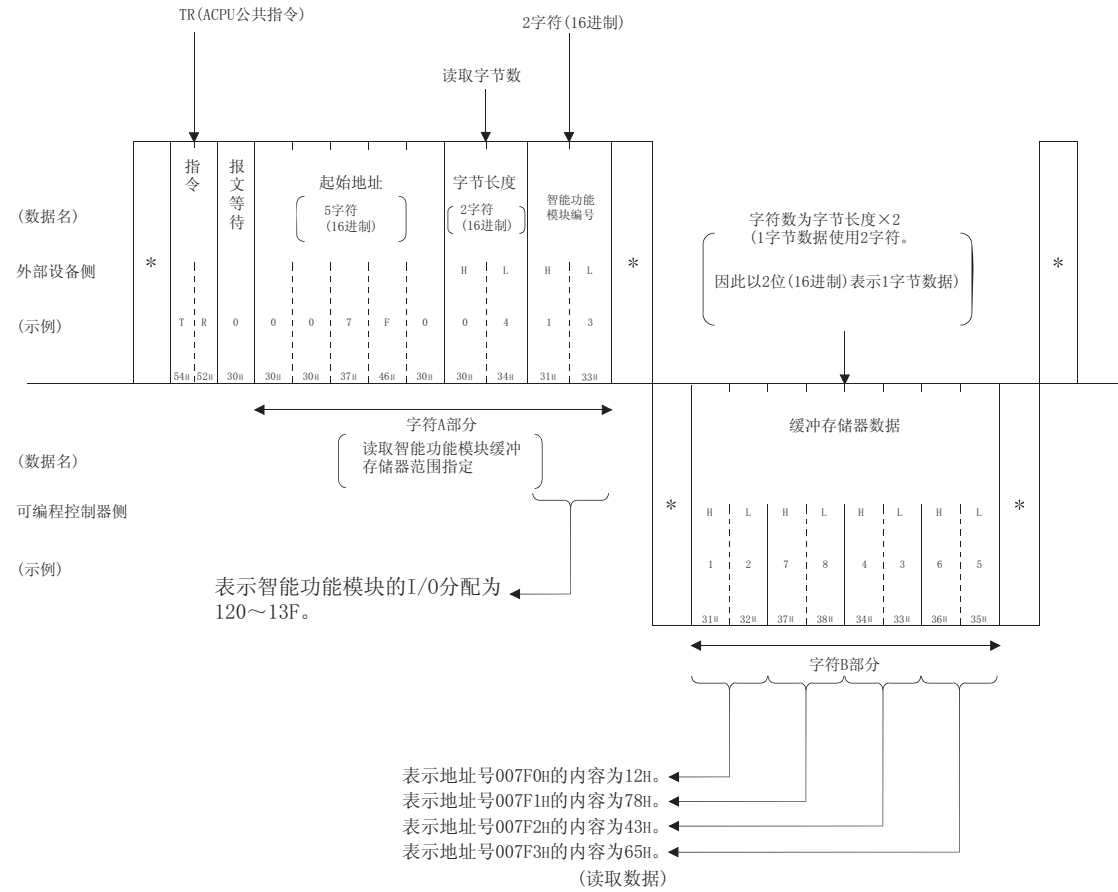


5.4.3 智能功能模块的缓冲存储器的读取(指令：TR)

以下对通过 TR 指令读取智能功能模块的缓冲存储器的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

[控制步骤]

- (读取条件)
- 1) 报文等待为 0ms。
 - 2) 对输入输出信号为 120H~13FH(模块编号 13H)的智能功能模块的缓冲存储器地址 7F0H~7F3H 的 4 字节进行读取。



要点
(1) 对于字节长度的指定范围应在满足以下条件的基础上进行指定。 • 1 ≤ 字节长度 ≤ 128
(2) 根据智能功能模块 1 个数据内容有时会跨越 2~3 字节，因此进行字节长度的指定时请参阅各模块的手册进行指定。

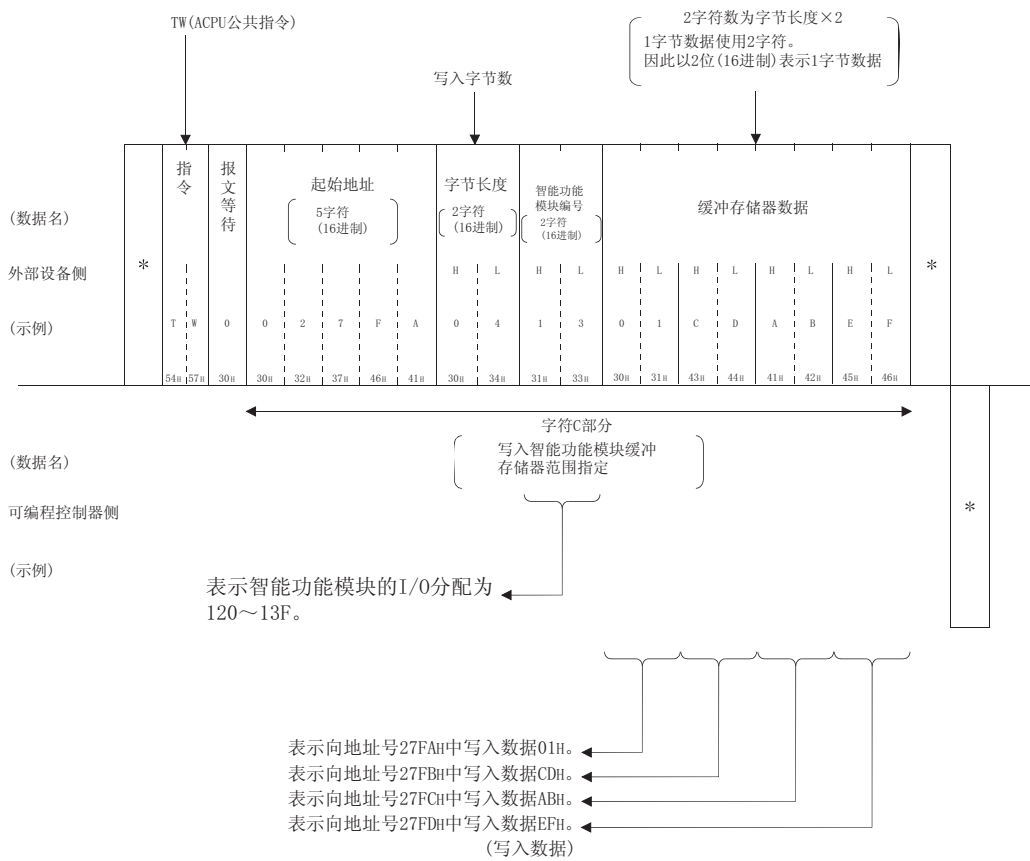
5. 4. 4 智能功能模块的缓冲存储器的写入(指令：TW)

以下对通过 TW 指令向智能功能模块的缓冲存储器进行数据写入的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5. 1 节中所示的详细说明。

[控制步骤]

(读取条件)

- 1) 报文等待为 0ms。
- 2) 向输入输出信号为 120H~13FH(模块编号 13H)的智能功能模块的缓冲存储器地址 27FAH~27FDH 写入 4 字节。



要点

- (1) 对于字节长度的指定范围应在满足以下条件的基础上进行指定。
 - 1 ≤ 字节长度 ≤ 128
- (2) 根据智能功能模块 1 个数据内容有时会跨越 2~3 字节，因此进行字节长度的指定时请参阅各模块的手册进行指定。

5.5 回送测试

回送测试是用于对外部设备与 C24 的通信功能是否正常动作进行测试的功能。以下对使用该功能时的控制步骤进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，根据通信时的格式而有所不同。
请参阅 5.1 节中所示的详细说明。

(1) ACPU 公共指令及处理内容

通过 TT 指令进行回送测试时的指令如下所示。

项目	指令		处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态		
	符号	ASCII 代码			STOP 中	RUN 中	
						写入允许设置	写入禁止设置
回送测试	TT	54H, 54H	将通过外部设备接收的字符原样不变地返回至外部设备。	254 字符	○	○	

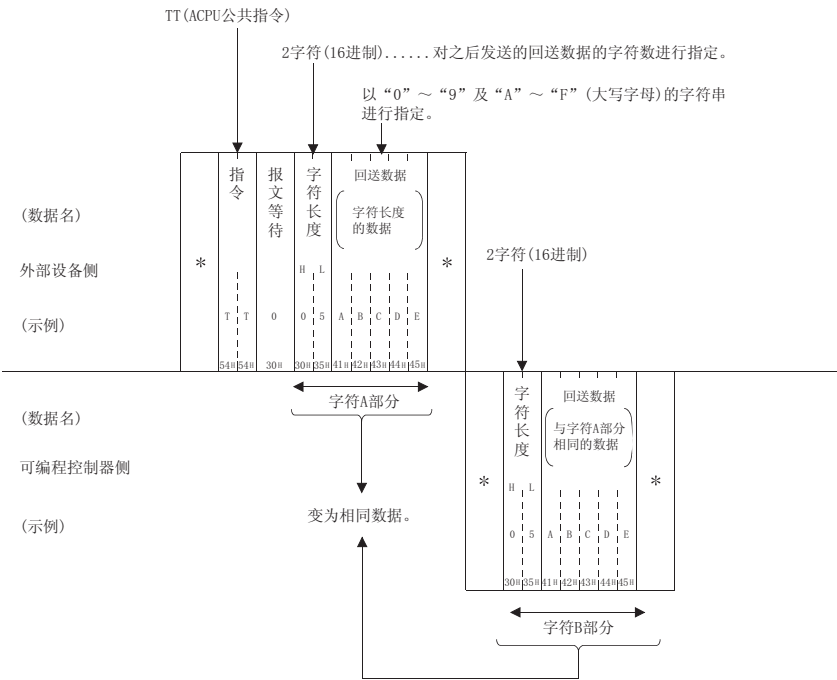
上表的可编程控制器 CPU 状态栏的○符号表示可以执行。

(2) 回送测试的控制步骤

[控制步骤]

(回送测试条件)

- 1) 报文等待为 0ms。
- 2) 作为回送数据，对“ABCDE”的 5 个字符进行发送接收。



要点
(1) 字符长度的指定范围应在满足以下条件的基础上进行指定。 • 1 ≦ 字符长度 ≦ 254
(2) 可编程控制器编号应指定为“FF”。

备忘录

[illegible]

6 通过 A 兼容 1E 帧进行通信时

在本章中，介绍对 E71 以 A 兼容 1E 帧进行基于 MC 协议的数据通信时的报文格式、报文中的数据项目的指定方法、限制等有关内容。

要点	
	使用 C24 时，无需阅读本章。

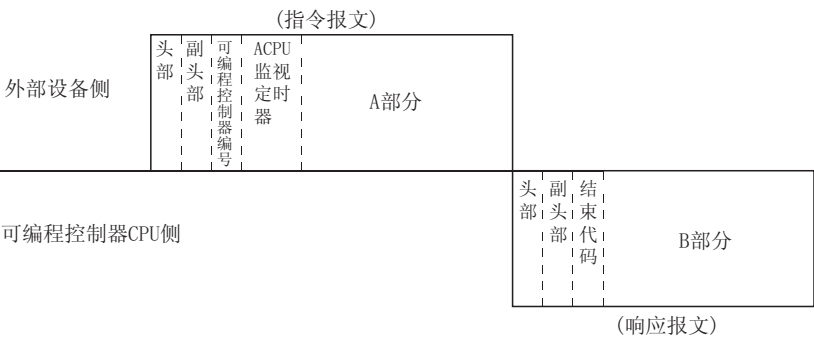
6.1 报文格式、控制步骤

以下介绍以 A 兼容 1E 帧进行数据通信时的各指令的报文格式及控制步骤有关内容。
以 A 兼容 1E 帧进行的数据通信与 A 系列以太网模块支持的可编程控制器 CPU 内数据的
读取/写入通信功能相同，只能使用 6.2 节中所示的指令。

6.1.1 指令说明项的阅读方法

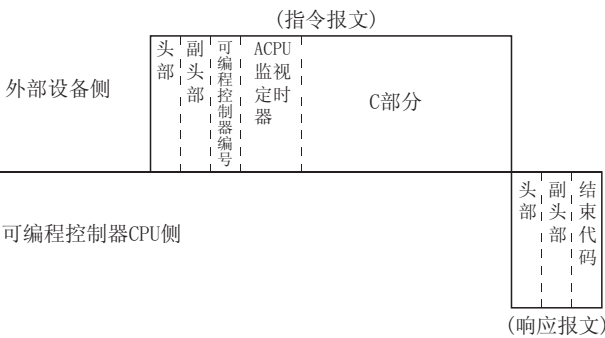
6.3 节～6.5 节中所示的各指令说明项的报文说明图的阅读方法如下所示。

(1) 外部设备通过可编程控制器进行数据读取的情况下



- 6
- 1) A 部分表示由外部设备至可编程控制器 CPU 的传送。
 - 2) B 部分表示由可编程控制器 CPU 至外部设备的传送。
 - 3) 创建外部设备的程序时应编制为各数据按从左至右的顺序进行传送。
(示例：A 部分的情况下，按从头部开始向右的顺序进行数据发送。)

(2) 从外部设备对可编程控制器进行数据写入的情况下



- 1) C 部分表示由外部设备至可编程控制器 CPU 传送。
- 2) B 部分表示由可编程控制器 CPU 至外部设备的传送。
- 3) 创建外部设备的程序时应编制为各数据按从左至右的顺序进行传送。
(示例：C 部分的情况下，按从头部开始向右的顺序进行数据发送。)

要点	
从外部设备接收指令报文时，可编程控制器 CPU 侧对报文中的 A 部分/C 部分的处理 完毕后，发送响应报文并进入下一个指令报文的接收等待(待机状态)状态。	

6.1.2 报文格式、控制步骤

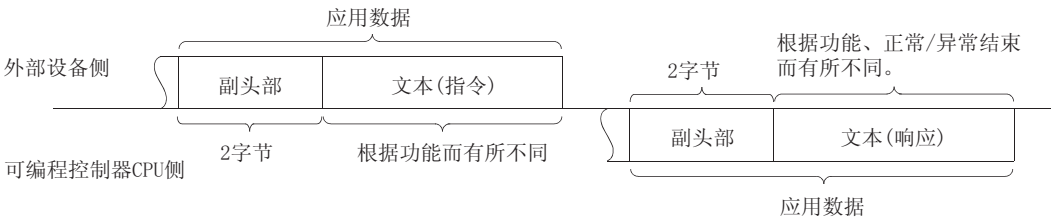
以下介绍通过 E71 用的 A 兼容 1E 帧进行数据通信时的报文格式、控制步骤有关内容。

(1) 报文格式

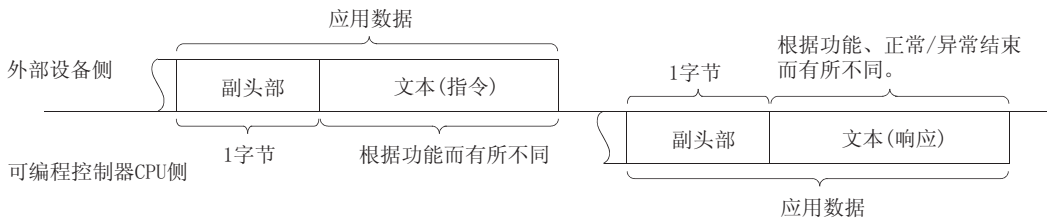
以下对 E71 与外部设备之间发送接收的报文格式有关内容进行说明。
通信数据由如下所示的“头部”及“应用数据”所构成。

头部	应用程序
----	------

- (a) 头部
头部是 TCP/IP、UDP/IP 用的头部。
对于外部设备侧→可编程控制器 CPU 侧(指令报文)的头部，应在外部设备侧附加后进行发送。
对于可编程控制器 CPU 侧→外部设备侧(响应报文)的头部，由于 E71 会自动附加，因此无需由用户进行设置。
- (b) 应用数据
应用数据按如下所示大致分为“副头部”及“文本”。
副头部是用于表示是指令报文还是响应报文的数据，设置值是确定的。
文本是用于设置各功能中的请求数据(指令)、响应数据(响应)的数据，以规定的格式进行数据设置。(有关详细内容请参阅 6.3 节及以后的内容。)
- (c) 应用数据部分的格式
 - 通过 ASCII 代码进行通信时



- 通过二进制代码进行通信时



备注

在基于 MC 协议进行的通信中，对于来自于外部设备的指令的响应是由 E71 创建并返送，因此用户无需进行设置。

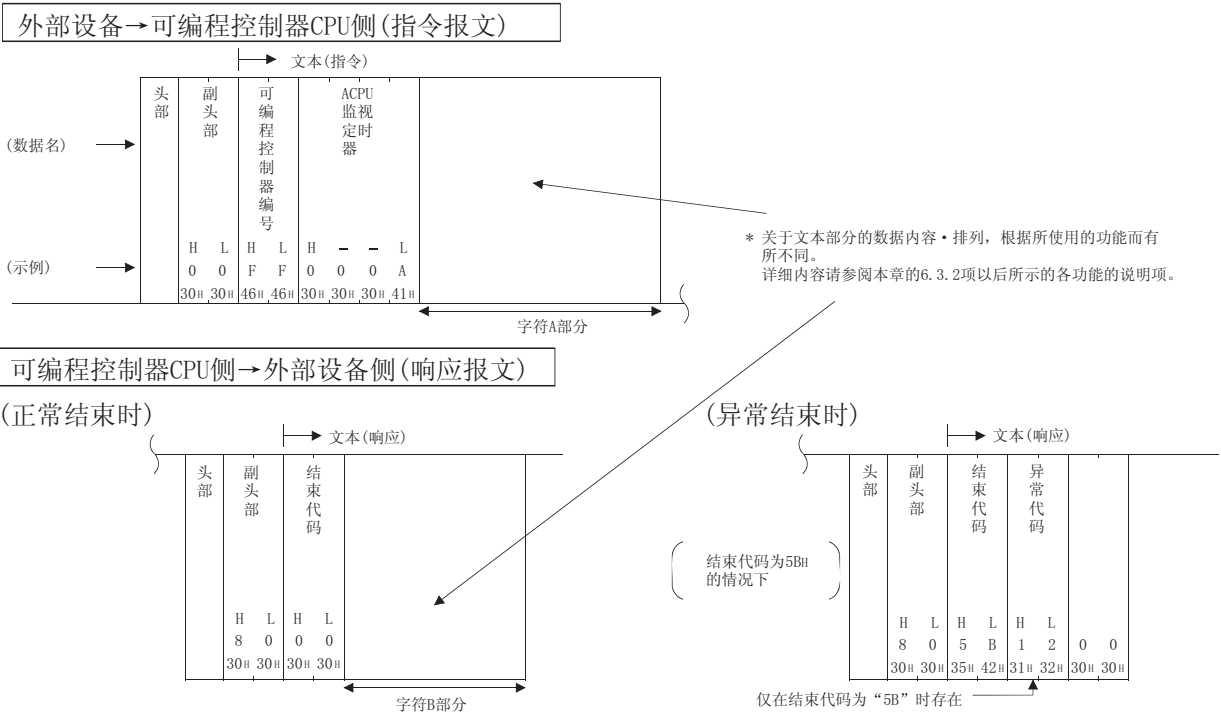
(2) 控制步骤

以下介绍通过 A 兼容 1E 帧进行通信时的控制步骤、应用数据部分的数据项目排列。

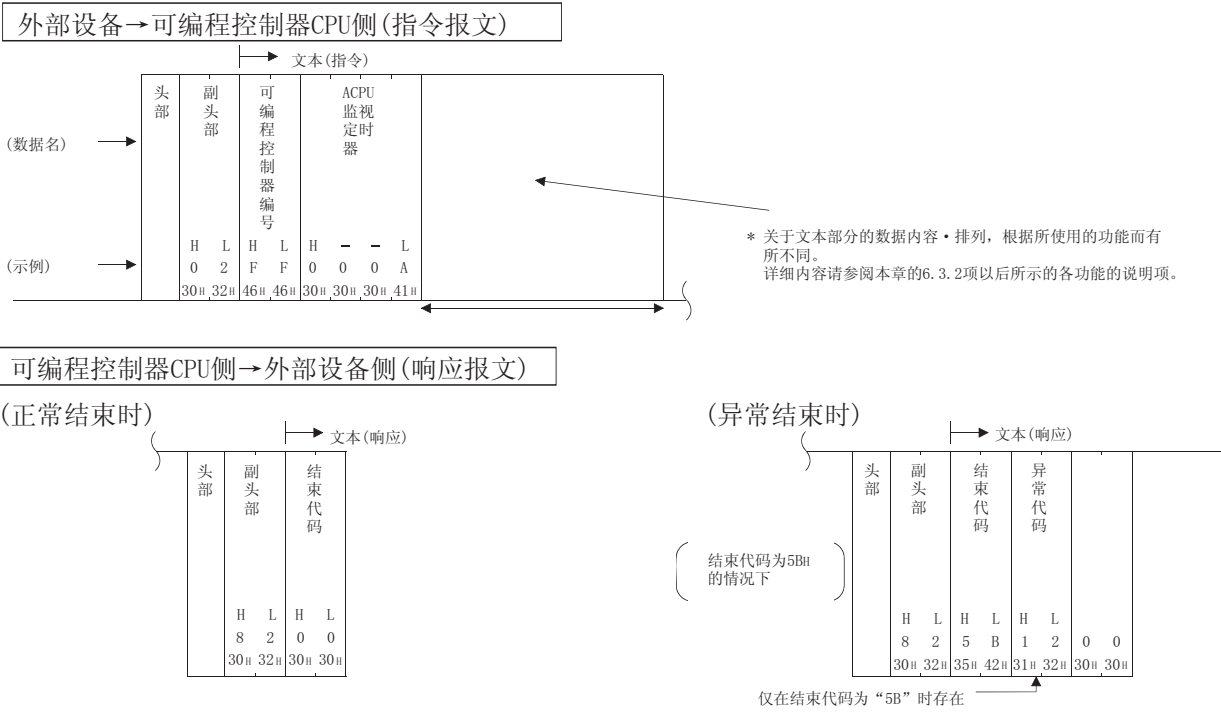
本项的报文说明图中所示的头部部分与本章 6.3.2 项以后中所示的报文说明图的*部分相对应。

关于报文格式的数据项目内容、数据指定方法，请参阅 6.1.3 项。

(a) 通过 ASCII 代码进行通信时
1) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下

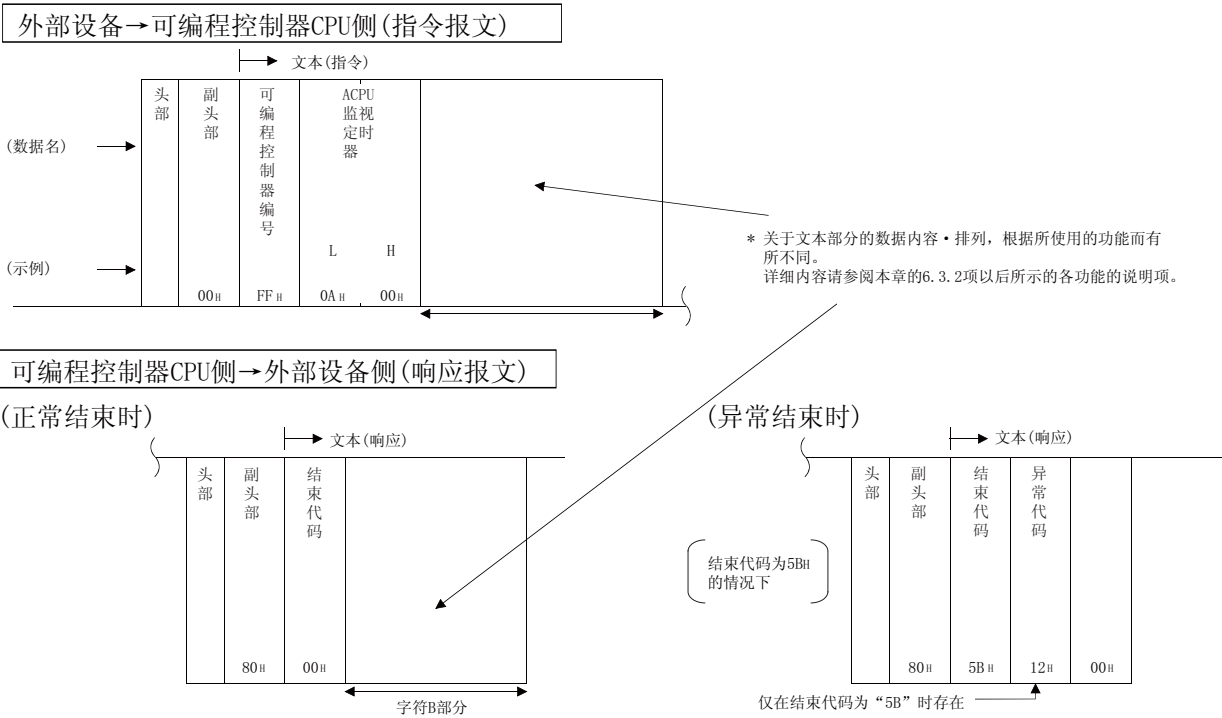


2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下

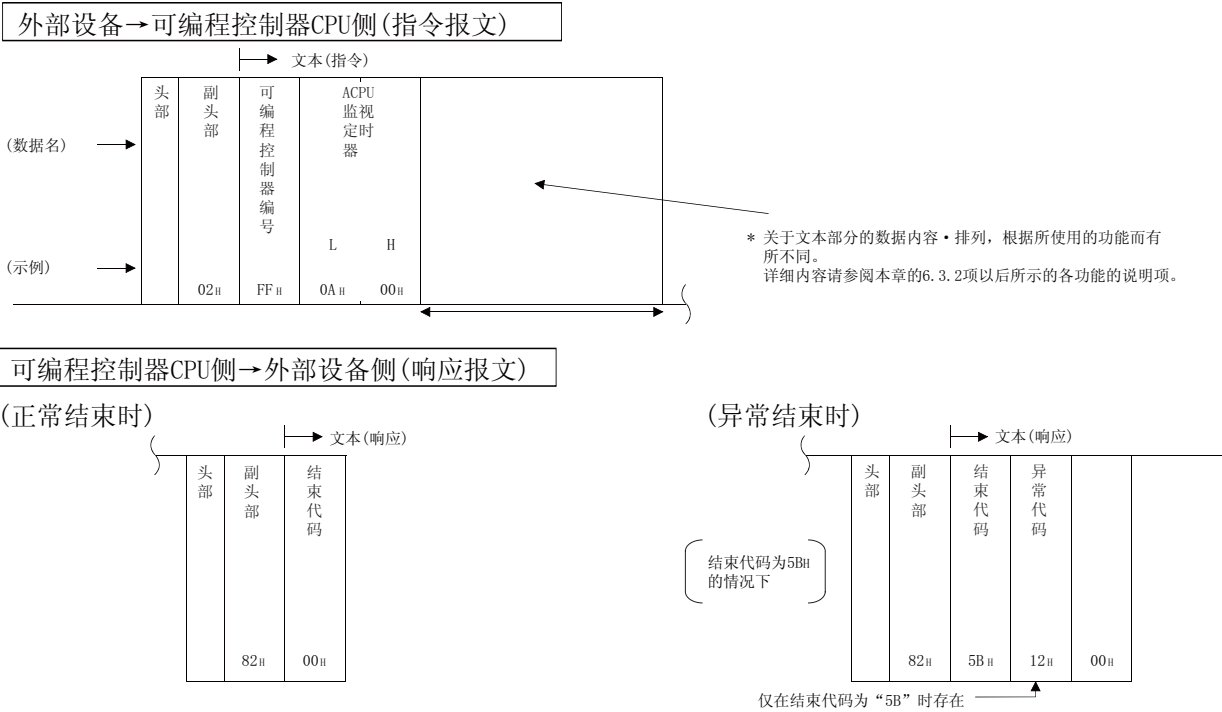


(b) 通过二进制代码进行通信时

1) 外部设备对本站可编程控制器 CPU 的数据进行读取的情况下



2) 外部设备对本站可编程控制器 CPU 进行数据写入的情况下



6.1.3 A 兼容 1E 帧的数据指定项目的内容

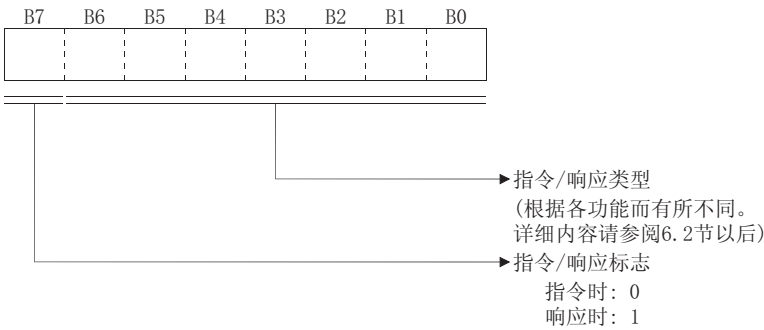
以下介绍以 A 兼容 1E 帧进行基于 MC 协议的通信时的指令以及响应的数据项目的内容。
对于从 E71 至外部设备的返送响应，E71 会自动进行数据设置，因此用户无需进行设置。

(1) 头部

头部是 TCP/IP、UDP/IP 用的头部。
对于外部设备侧→可编程控制器 CPU 侧(指令报文)的头部，应在外部设备侧附加后进行发送。
对于可编程控制器 CPU 侧→外部设备侧(响应报文)的头部，由于 E71 会自动附加，因此无需由用户进行设置。

(2) 副头部

副头部的格式构成如下所示。



(3) 可编程控制器编号

可编程控制器编号是用于识别对 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的哪个可编程控制器进行访问。
对于可编程控制器编号，使用 FFH 或将网络模块上连接的站号在 00H~40H 范围内进行指定。

	外部设备的访问站	外部设备指定的可编程控制器编号
1	外部设备连接站(本站)	FFH
2	来自于外部设备连接站的多点连接站	FFH
3	CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的站(上述 1、2 除外) (*1)	01H ~ 40H (1 ~ 64): 访问站的站号(*2)
4	MELSECNET/H 上的远程主站(将外部设备与远程 I/O 站的 E71 相连接时)	(访问禁止)

- *1 对网络模块(以太网、CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10)的“其它站访问时的有效模块”的设置中指定的网络上的模块进行访问。
- *2 不能指定 CC-Link IE 控制网络、CC-Link IE 现场网络的站号 41H ~ 78H(65 ~ 120)。
 - (a) 通过二进制代码进行通信时，将可编程控制器编号以二进制值进行表示。
 - (b) 通过 ASCII 代码进行通信时，以将可编程控制器编号以 16 进制数表示时的 ASCII 代码进行表示。

要点

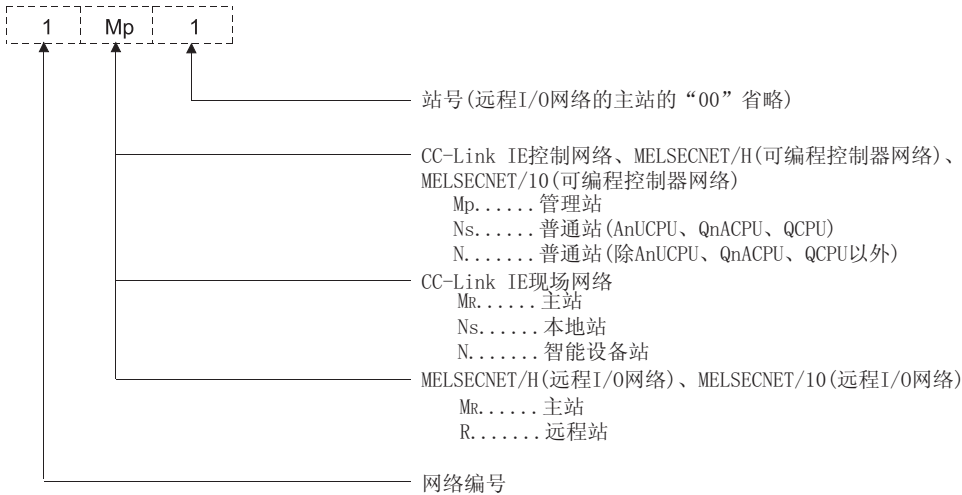
为了对其它站的可编程控制器 CPU 进行访问，需要对以下网络系统进行网络参数设置以及“其它站访问时的有效模块”的设置。

- CC-Link IE 控制网络
- CC-Link IE 现场网络
- MELSECNET/H
- MELSECNET/10
- 以太网

*有关详细内容请参阅各网络模块的手册。

[可编程控制器编号的指定示例]
(图中所示各站的符号的含义)

- 网络系统 (CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10)



(a) CC-Link IE 控制网络、MELSECNET/H(可编程控制器网络)、MELSECNET/10(可编程控制器网络)的情况下

- 二级系统时

指定的可编程控制器编号

外部设备 连接站	外部设备访问的可编程控制器 CPU			
	1Mp1	1Ns2	1N3	1Ns4
1Mp1	FF	02	03	04
1Ns4	01	02	03	FF

×: 访问禁止

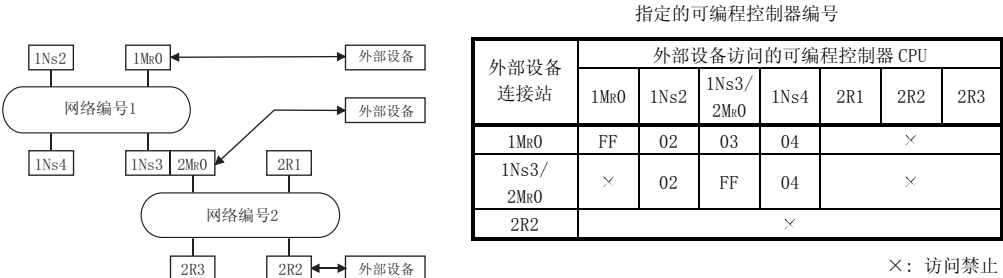
• 多级系统时

指定的可编程控制器编号

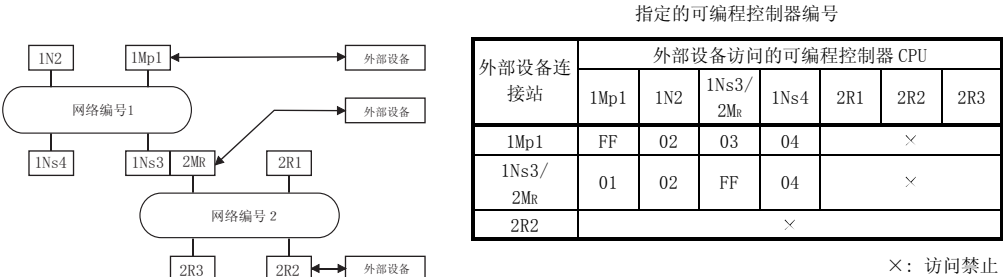
外部设备 连接站	外部设备访问的可编程控制器 CPU						
	1Mp1	1N2	1Ns3/ 2Mp1	1Ns4	2N2	2Ns3	2Ns4
1Mp1	FF	02	03	04	×		
1Ns3/ 2Mp1	01	02	FF	04	02	03	04
2Ns3	×		01	×	02	FF	04

×: 访问禁止

(b) CC-Link IE 现场网络的情况下
• 多级系统时



(c) MELSECNET/H(远程 I/O 网络)的情况下
• 多级系统时



(4) ACPU 监视定时器

是 E71(从外部设备接收了请求数据的 E71)对可编程控制器 CPU 发出读取/写入请求后，至返回结果为止的等待时间设置定时器。

(a) 设置值以下述值进行指定。

- 0000H(0)：：无限等待*1
- 0001 ～ FFFFH(1 ～ 65535)：等待时间(单位 250ms)

*1 持续等待直至从可编程控制器 CPU 返回响应为止。

(b) 为了正常进行数据通信，建议根据通信目标在下表的设置范围内使用。

设置范围	通信目标
1 ～ 40h(0.25 ～ 10s)	本站
2 ～ 240h(0.5 ～ 60s)	经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 的其它站或通过路由器中继的其它站

要点
对 QnACPU、ACPU 进行访问的情况下，由于对 CPU 类型进行判别，仅初次返回响应报文之前需要耗费相当于 CPU 监视定时器等待时间的时间，因此必须在上述(b)中所示的设置范围内使用。

(5) 文本(指令)

显示外部设备向对象可编程控制器站进行数据的读取/写入时，表示所使用的功能的 E71 指令等。

对于文本(指令)部分数据的内容・排列，根据所使用的功能而有所不同。

关于各功能的数据排列，请参阅 6.2 节以后的各功能说明项。

(6) 文本(响应)

显示外部设备向对象可编程控制器站进行数据的读取/写入时的读取数据/处理结果等。

关于文本(响应)部分的数据内容・排列，根据所使用的功能而有所不同。

关于各功能正常结束时的数据排列，请参阅 6.2 节以后的各功能说明项。

(7) 结束代码

将外部设备向对象可编程控制器站进行数据的读取/写入时的处理结果以下述值进行显示。

- 00H : 正常结束
- 00H 以外 : 异常结束 (01H ~ B001H)

- (a) 通过二进制代码进行通信时，将结束代码以二进制值进行显示。
- (b) 通过 ASCII 代码进行通信时，以将结束代码以 16 进制数表示时的 ASCII 代码进行显示。
- (c) 异常结束时，应根据以太网用户手册(基本篇)故障排除的进行内容确认及处理。
结束代码为 5BH/“5B”的情况下，其后面将包含有异常代码(10H~21H)数据及 00H/“00”。

(8) 异常代码

外部设备向对象可编程控制器站进行数据的读取/写入时的处理结果异常，结束代码为 5BH/“5B”的情况下显示异常内容。(异常代码: 10H~21H)

- (a) 通过二进制代码进行通信时，将异常代码以二进制值进行显示。
- (b) 通过 ASCII 代码进行通信时，以将异常代码以 16 进制数表示时的 ASCII 代码进行显示。
- (c) 应根据以太网用户手册(基本篇)故障排除进行内容确认及处理。

要点	
对于 E71 与外部设备之间进行指令、响应的发送接收时的数据代码(ASCII/二进制)，是通过 GX Works2 或 GX Developer 的动作设置进行确定。 对于与 E71 进行通信的外部设备侧，应将指令、响应中的各项目中使用的值根据上述设置以如下所示的代码进行发送。 此外，应以如下所示的代码进行接收。在本节以后的说明中，将指令、响应中的各项目中使用的值以二进制值表示。	
(1) 通过二进制代码进行通信时 在未特别说明的情况下，将各说明中所示的值原样不变地以二进制值及指定的顺序(L-H)进行发送接收。	
(2) 通过 ASCII 代码进行通信时 在未特别说明的情况下，将各说明中所示的值转换为 16 进制数的 ASCII 代码后，以指定顺序(H-L)进行发送接收。	

备注

以下述条件进行基于 MC 协议的通信时的副头部～ACPU 监视定时器的指定示例如下所示。

- (指定值)
- 对象站

• 使用的功能

• ACPU 监视定时器值

: 安装 E71 的可编程控制器 CPU 站(本站)

: 软元件存储器的批量读取(位单位)(A 兼容 1E 帧用指令)

: 2500 ms

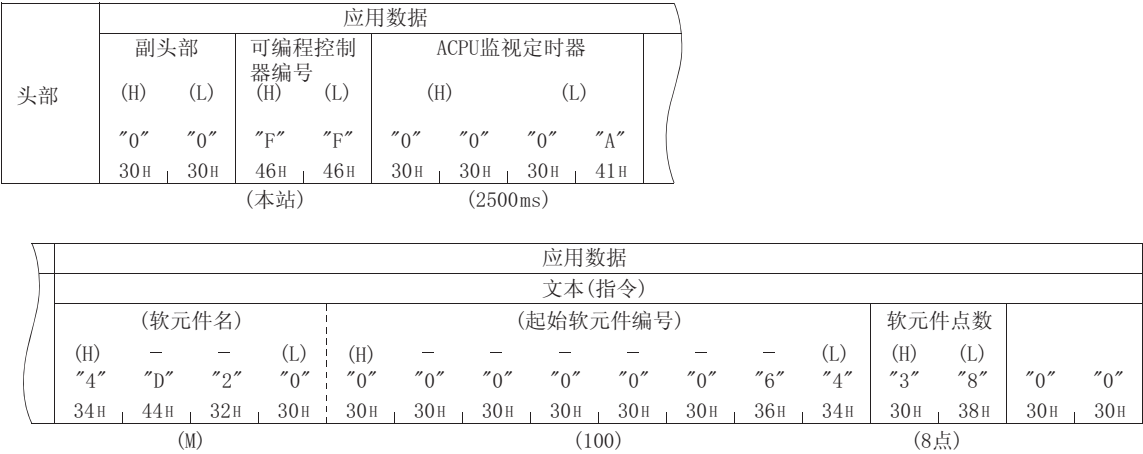
: FF_H

: 00_H

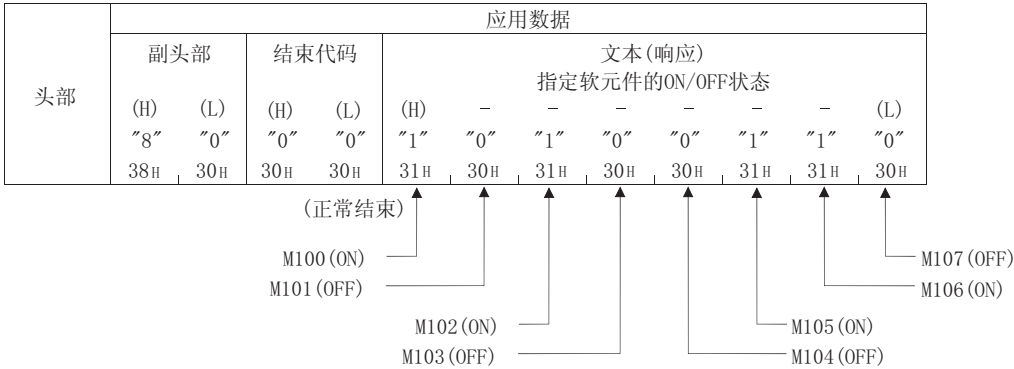
: 000A_H
- (1) 通过二进制代码进行通信时的格式
- (a) 发送指令时的排列(外部设备→E71)
- | 头部 | 应用数据 | | | | | | | | | |
|----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | 副头部 | 可编程控制器
器编号 | 文本(指令) | | | | | | | |
| | | | ACPU监视
定时器 | | (起始软元件编号) | | | | (软元件名) | |
| | | | (L) | (H) | (L) | — | — | (H) | (L) | (H) |
| | 00 _H | FF _H | 0A _H | 00 _H | 64 _H | 00 _H | 00 _H | 00 _H | 20 _H | 4D _H |
| | | (本站) | (2500ms) | | (100) | | | | (M) | |
| | | | | | | | | | (8点) | |
- (b) 接收响应时的排列(外部设备←E71)
- | 头部 | 应用数据 | | | |
|----|-----------------|-----------------|-----------------|-----------------|
| | 副头部 | 结束代码 | 文本(响应) | |
| | | | 指定软元件的ON/OFF状态 | |
| | 80 _H | 00 _H | 10 _H | 10 _H |
| | | (正常结束) | 10 _H | 01 _H |
| | | M100 (ON) | 10 _H | 10 _H |
| | | M101 (OFF) | 10 _H | 01 _H |
| | | M102 (ON) | 10 _H | 10 _H |
| | | M103 (OFF) | 10 _H | 01 _H |
| | | M104 (OFF) | 10 _H | 01 _H |
| | | M105 (ON) | 10 _H | 10 _H |
| | | M106 (ON) | 10 _H | 01 _H |
| | | M107 (OFF) | 10 _H | 10 _H |
- 6 - 12
- 6 - 12

(2) 通过 ASCII 代码进行通信时的格式

(a) 指令发送时的排列(外部设备→E71)



(b) 响应接收时的排列(外部设备←E71)



6.1.4 字符部分的传送数据的思路

关于以 A 兼容 1E 帧进行基于 MC 协议的通信时的传送数据(各字符部分)中表示的位软元件数据、字软元件数据的传送顺序的思路，与以 QnA 兼容 3E/3C/4C 帧、4E 帧进行通信时相同。
请参阅 3.1.7 项。

要点	
对于 3.1.7 项中所示的传送数据的思路的说明中所示的“附加代码”数据，是 C24 使用的数据。	
对 E71 进行数据发送接收时，应忽略“附加代码”相关说明以及图的部分。	

6.2 A 兼容 1E 帧用指令及功能列表

以 A 兼容 1E 帧从外部设备对可编程控制器进行访问时的指令及其功能等如下所示。

功能			指令/响应类型	处理内容	1 次通信中可处理的点数	
软件元件存储器(*6)	批量读取	位单位	00H	对位软元件(X、Y、M等)以1点为单位进行读取。	256点	
		字单位	01H	对位软元件(X、Y、M等)以16点为单位进行读取。 对字软元件(D、T、C等)以1点为单位进行读取。	128字(2048点) 256点	
	批量写入(*5)	位单位	02H	对位软元件(X、Y、M等)以1点为单位进行写入。	256点	
		字单位	03H	对位软元件(X、Y、M等)以16点为单位进行写入。 对字软元件(D、T、C等)以1点为单位进行写入。	40字(640点) 256点	
	测试 (随机写入)(*5)	位单位	04H	对位软元件(X、Y、M等)以1点为单位随机指定 软元件• 软元件编号进行设置/复位。	80点	
		字单位	05H	对位软元件(X、Y、M等)以16点为单位随机指定 软元件• 软元件编号进行设置/复位。 对字软元件(D、T、C等)以1点为单位随机指定 软元件• 软元件编号进行写入。	40字(640点) 40点	
	监视数据登录 (*3)	位单位	06H	对进行监视的位软元件(X、Y、M等)以1点为单位进行 登录。(*2)	40点	
		字单位	07H	对进行监视的位软元件(X、Y、M等)以16点为单位进行 登录。(*2) 对进行监视的字软元件(D、T、C等)以1点为单位进行 登录。	20字(320点) 20点	
	监视	位单位	08H	对进行了监视数据登录的软元件进行监视。	(登录点数)	
		字单位	09H			
扩展文件寄存器	批量读取		17H	对扩展文件寄存器(R)以1点为单位进行读取。	对扩展文件寄存器(R) 以1点为单位	
	批量写入		18H	对扩展文件寄存器(R)以1点为单位进行写入。	对扩展文件寄存器(R) 以1点为单位	
	测试(随机写入)		19H	对扩展文件寄存器(R)以1点为单位随机指定块编号软元 件编号进行写入。	对扩展文件寄存器(R) 以1点为单位	
	监视数据登录(*3)		1AH	对进行监视的扩展文件寄存器(R)以1点为单位进行登 录。	进行监视的扩展文件 寄存器	
	监视		1BH	对进行了监视数据登录的扩展文件寄存器(R)进行监视。	进行了监视数据登录的 扩展文件寄存器	
	直接读取		3BH	对扩展文件寄存器(R)通过直接指定以1点为单位进行读 取。	对扩展文件寄存器(R) 进行直接指定	
	直接写入		3CH	对扩展文件寄存器(R)通过直接指定以1点为单位进行写 入。	对扩展文件寄存器(R) 进行直接指定	
智能功能 模块	批量读取		0EH	对智能功能模块的缓冲存储器的内容进行读取。	对智能功能模块的缓冲存 储器的内容进行读取。	
	批量写入		0FH	对智能功能模块的缓冲存储器进行数据写入。		

功能	-			访问站(*4)						可编程控制器 CPU 的状态			参阅章节
				ACPU	QnA CPU	QCPU (*7)	安全 CPU	MELSECNET/10 远程站	MELSECNET/H 远程站	STOP 中	RUN 中		
											写入允许设置	写入禁止设置	
软件元件存储器(*5)	批量读取	位单位	○	○	○	○	×	×	○	○	○	6.3.2 项	
		字单位	○	○	○	○	×	×				6.3.5 项	
	批量写入	位单位	○	○	○	×	×	×	○	○	×	6.3.3 项	
		字单位	○	○	○	×	×	×				6.3.6 项	
	测试(随机写入)	位单位	○	○	○	×	×	×	○	○	×	6.3.4 项	
		字单位	○	○	○	×	×	×				6.3.7 项	
	监视数据登录(*3)	位单位	○	○	○	×	×	×	○	○	○	6.3.8 项	
		字单位	○	○	○	×	×	×					
	监视	位单位	○	○	○	×	×	×	○	○	○		
		字单位	○	○	○	×	×	×					
扩展文件寄存器	批量读取		○	×	×	×	×	×	○	○	○	6.4.3 项	
	批量写入		○	×	×	×	×	×	○	○	×	6.4.4 项	
	测试(随机写入)		○	×	×	×	×	×	○	○	×	6.4.5 项	
	监视数据登录(*3)		○	×	×	×	×	×	○	○	○	6.4.6 项	
	监视		○	×	×	×	×	×					
	直接读取		○	×	×	×	×	×				6.4.7 项	
	直接写入		○	×	×	×	×	×				6.4.7 项	
智能功能模块	批量读取		○	×	×	×	○	×	○	○	○	6.5.3 项	
	批量写入		○	×	×	×	○	×	○	○	×	6.5.4 项	

- *1 至可编程控制器 CPU 的运行中写入的允许/禁止设置是在 GX Works2 或 GX Developer 的以下画面中进行。
 - E71 的情况下: “以太网动作设置”画面
- *2 AnA/AnU/QnACPU/QCPU 以外的情况下, 软元件 X(输入)的 1 点作为 2 点的处理点数。指定软元件中包含有 X 时, 应满足以下公式:
$$((X \text{ 的指定点数} \times 2) + \text{其它软元件指定点数}) \leq 1 \text{ 次通信中可处理的点数}$$

仅指定了 X 的情况下, 1 次通信中可处理的点数将变为表中值的 1/2。
- *3 可登录到 E71 中的软元件为 3 种类型指令(06H、07H、1AH)中的 1 个指令。
上述某个最后使用的指令中的指定软元件将被登录到 E71 中。
- *4 关于表中所示的访问站的详细内容, 请参阅 2.6.1 项。
- *5 执行指令的 Q/QnACPU 中被实施了系统保护时将变为出错状态, 并返回异常结束的响应报文。
- *6 对 Q/QnACPU 以外的扩展文件寄存器进行读取/写入的情况下, 应使用扩展寄存器的专用指令。
- *7 对通用型 QCPU 进行访问的情况下, 应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。
序列号的前 5 位数为 10101 以前的情况下, 应以 QnA 兼容 3E 帧、4E 帧进行访问。
此外, 对 CPU 模块的内置以太网端口进行访问的情况下, 请参阅以下手册。
 - QnUCPU 用户手册(内置以太网端口通信篇)

6.3 软元件存储器的读取、写入

以下对进行软元件存储器的读取、写入时的控制步骤的指定内容及指令示例进行说明。
此外，进行扩展文件寄存器的读取、写入时，应使用扩展文件寄存器专用的指令。
(参阅 6.4 节)

6.3.1 指令及软元件范围

(1) 用于软元件存储器的读取、写入的指令

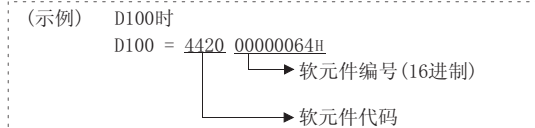
项目		指令/响应类型	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态(*1)		
					STOP 中	RUN 中	
						写入允许设置	写入禁止设置
批量读取	位单位	00H	对位软元件(X、Y、M 等)以 1 点为单位进行读取。	256 点	○	○	○
	字单位	01H	对位软元件(X、Y、M 等)以 16 点为单位进行读取。	128 字(2048 点)			
			对字软元件(D、T、C 等)以 1 点为单位进行读取。	256 点			
批量写入(*4)	位单位	02H	对位软元件(X、Y、M 等)以 1 点为单位进行写入。	256 点	○	○	×
	字单位	03H	对位软元件(X、Y、M 等)以 16 点为单位进行写入。	40 字(640 点)			
			对字软元件(D、T、C 等)以 1 点为单位进行写入。	256 点			
测试(随机写入)(*4)	位单位	04H	对位软元件(X、Y、M 等)以 1 点为单位随机指定软元件・软元件编号进行设置/复位。	80 点	○	○	×
	字单位	05H	对位软元件(X、Y、M 等)以 16 点为单位随机指定软元件・软元件编号进行设置/复位。	40 字(640 点)			
			对字软元件(D、T、C 等)以 1 点为单位随机指定软元件・软元件编号进行写入。	40 点			
监视数据登录(*3)	位单位	06H	对进行监视的位软元件(X、Y、M 等)以 1 点为单位进行设置。(*2)	40 点	○	○	○
	字单位	07H	对进行监视的位软元件(X、Y、M 等)以 16 点为单位进行设置。(*2)	20 字(320 点)			
			对进行监视的字软元件(D、T、C 等)以 1 点为单位进行设置。	20 点			
监视	位单位	08H	对进行了监视数据登录的软元件进行监视。	(登录点数分)	○	○	○
	字单位	09H					

关于*1、*2、*3 的有关内容，请参阅 6.2 节的*1、*2、*3。

关于*4 的有关内容，请参阅 6.2 节的*5。

(2) 软元件的指定方法以及允许访问范围

(a) 对于软元件存储器的读取/写入中各软元件的设置方法，按下图所示通过软元件代码及软元件编号进行设置。



(b) 软元件代码、软元件编号如下所示。

软元件列表(无限制 CPU 模块)

○：可以访问 ×：禁止访问 —：无软元件

软元件(*1)		软元件代码	软元件范围	软元件编号	A1S A1SH A1SJ A1SJH A1 A1N	A2S A2SH A2 A2N A2C A2CJ A0J2H	A2-S1 A2N-S1	A3 A3N	A2A	A2A-S1	A3A
数据寄存器	D0 (44 _H , 20 _H)	D0 ~ D1023	0000 _H ~ 03FF _H	0000 _H ~ 03FF _H	○	○	○	○	○	○	○
		D1024 ~ D6143	0400 _H ~ 17FF _H	0400 _H ~ 17FF _H	—	—	—	—	—	—	—
		D9000 ~ D9255	2328 _H ~ 2427 _H	2328 _H ~ 2427 _H	○	○	○	○	○	○	○
链接寄存器	W0 (57 _H , 20 _H)	W0 ~ W3FF	0000 _H ~ 03FF _H	0000 _H ~ 03FF _H	○	○	○	○	○	○	○
		W400 ~ WFFF	0400 _H ~ 0FFF _H	0400 _H ~ 0FFF _H	—	—	—	—	—	—	—
文件寄存器	R0 (52 _H , 20 _H)	R0 ~ R4095	0000 _H ~ 0FFF _H	0000 _H ~ 0FFF _H	—	—	○	○	○	○	○
		R4096 ~ R8191	1000 _H ~ 1FFF _H	1000 _H ~ 1FFF _H	—	—	—	○	—	—	—
定时器	当前值	TN (54 _H , 4E _H)	T0 ~ T255	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
			T256 ~ T2047	0100 _H ~ 07FF _H	—	—	—	—	—	—	—
	触点	TS (54 _H , 53 _H)	T0 ~ T255	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
			T256 ~ T2047	0100 _H ~ 07FF _H	—	—	—	—	—	—	—
计数器	当前值	CN (43 _H , 4E _H)	C0 ~ C255	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
			C256 ~ C1023	0100 _H ~ 03FF _H	—	—	—	—	—	—	—
	触点	CS (43 _H , 53 _H)	C0 ~ C255	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
			C256 ~ C1023	0100 _H ~ 03FF _H	—	—	—	—	—	—	—
输入	X0 (58 _H , 20 _H)	X0 ~ X0FF	0000 _H ~ 00FF _H	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
		X100 ~ X1FF	0100 _H ~ 01FF _H	0100 _H ~ 01FF _H	—	—	○	○	—	—	—
		X200 ~ X3FF	0200 _H ~ 03FF _H	0200 _H ~ 03FF _H	—	—	—	○	—	—	—
		X400 ~ X7FF	0400 _H ~ 07FF _H	0400 _H ~ 07FF _H	—	—	—	○	—	—	—
输出	Y0 (59 _H , 20 _H)	Y0 ~ Y0FF	0000 _H ~ 00FF _H	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
		Y100 ~ Y1FF	0100 _H ~ 01FF _H	0100 _H ~ 01FF _H	—	—	○	○	—	—	—
		Y200 ~ Y3FF	0200 _H ~ 03FF _H	0200 _H ~ 03FF _H	—	—	—	○	—	—	—
		Y400 ~ Y7FF	0400 _H ~ 07FF _H	0400 _H ~ 07FF _H	—	—	—	○	—	—	—
内部继电器 * 包括锁存继电器、步进继电器	M0 (4D _H , 20 _H)	M0 ~ M2047	0000 _H ~ 07FF _H	0000 _H ~ 07FF _H	○	○	○	○	○	○	○
		M2048 ~ M8191	0800 _H ~ 1FFF _H	0800 _H ~ 1FFF _H	—	—	—	—	—	—	—
		M9000 ~ M9255	2328 _H ~ 2427 _H	2328 _H ~ 2427 _H	○	○	○	○	○	○	○
链接继电器	B0 (42 _H , 20 _H)	B0 ~ B3FF	0000 _H ~ 03FF _H	0000 _H ~ 03FF _H	○	○	○	○	○	○	○
		B400 ~ BFFF	0400 _H ~ 0FFF _H	0400 _H ~ 0FFF _H	—	—	—	—	—	—	—
报警器	F0 (46 _H , 20 _H)	F0 ~ F255	0000 _H ~ 00FF _H	0000 _H ~ 00FF _H	○	○	○	○	○	○	○
		F256 ~ F2047	0100 _H ~ 07FF _H	0100 _H ~ 07FF _H	—	—	—	—	—	—	—

(转下页)

软元件列表(带限制 CPU 模块)

○：可以访问 ×：禁止访问 —：无软元件

软元件(*1)		软元件代码	软元件范围	软元件编号	基本型 QCPU，安全 CPU	高性能型 QCPU、过程 CPU、 冗余 CPU、通用型 QCPU(*1)	QJ72LP25-25 QJ72LP25GE QJ72BR15
数据寄存器		D0 (44 _h , 20 _h)	D0 ~ D6143	0000 _h ~ 17FF _h	○		×
			D6144 以上	1800 _h 以上	×		
			D9000 ~ D9255 (SD1000 ~ SD1255)	2328 _h ~ 2427 _h	-	○	-
			(SD1256 ~ SD2047)		×		
链接寄存器		W0 (57 _h , 20 _h)	W0 ~ W7FF	0000 _h ~ 7FFF _h	○		×
			W800 ~ WFFF	0800 _h ~ 0FFF _h	-	○	
			W1000 以上	1000 _h 以上	×		
文件寄存器		R0 (52 _h , 20 _h)	R0 以上	0000 _h 以上	×		-
定时器	当前值	TN (54 _h , 4E _h)	T0 ~ T511	0000 _h ~ 01FF _h	○		
			T512 ~ T2047	0200 _h ~ 07FF _h	-	○	
			T2048 以上	0800 _h 以上	×		
	触点	TS (54 _h , 53 _h)	T0 ~ T511	0000 _h ~ 01FF _h	○		
			T512 ~ T2047	0200 _h ~ 07FF _h	-	○	
			T2048 以上	0800 _h 以上	×		
	线圈	TC (54 _h , 43 _h)	T0 ~ T511	0000 _h ~ 01FF _h	○		
			T512 ~ T2047	0200 _h ~ 07FF _h	-	○	
			T2048 以上	0800 _h 以上	×		
	计数器	当前值	CN (43 _h , 4E _h)	C0 ~ C511	0000 _h ~ 01FF _h	○	
C512 ~ C1023				0200 _h ~ 03FF _h	-	○	
C1024 以上				0400 _h 以上	×		
触点		CS (43 _h , 53 _h)	C0 ~ C511	0000 _h ~ 01FF _h	○		
			C512 ~ C1023	0200 _h ~ 03FF _h	-	○	
			C1024 以上	0400 _h 以上	×		
线圈		CC (43 _h , 43 _h)	C0 ~ C511	0000 _h ~ 01FF _h	○		
			C512 ~ C1023	0200 _h ~ 03FF _h	-	○	
			C1024 以上	0400 _h 以上	×		
输入		X0 (58 _h , 20 _h)	X0 ~ X7FF	0000 _h ~ 07FF _h	○		×
			X800 以上	0800 _h 以上	×		-
输出		Y0 (59 _h , 20 _h)	Y0 ~ Y7FF	0000 _h ~ 07FF _h	○		×
			Y800 以上	0800 _h 以上	×		-
内部继电器		M0 (4D _h , 20 _h)	M0 ~ M8191	0000 _h ~ 1FFF _h	○		×
			M8192 以上	2000 _h 以上	×		-
			M9000 ~ M9255 (SM1000 ~ SM1255)	2328 _h ~ 2427 _h	-	○	
			(SM1256 ~ SM2047)		×		
锁存继电器					×		* 即使指定锁存继电器(L)、步进继电器(S) 也将对内部继电器(M)进行访问。
步进继电器							
链接继电器		B0 (42 _h , 20 _h)	B0 ~ B7FF	0000 _h ~ 07FF _h	○		×
			B800 ~ BFFF	0800 _h ~ 0FFF _h	-	○	
			B1000 以上	1000 _h 以上	×		
报警器		F0 (46 _h , 20 _h)	F0 ~ F1023	0000 _h ~ 03FF _h	○		-
			F1024 ~ F2047	0400 _h ~ 07FF _h	-	○	
			F2048 以上	0800 _h 以上	×		

(转下页)

软元件列表(带限制 CPU 模块)

○：可以访问 ×：禁止访问 -：无软元件

软元件(*1)		软元件代码	软元件范围	软元件编号	A2AS A2U	A2AS-S1 A2U-S1	A3U A4U	Q2A Q2AS Q2ASH	Q2A-S1 Q2AS-S1 Q2ASH-S1	Q3A Q4A Q4AR
数据寄存器		D0 (44 _H , 20 _H)	D0 ~ D6143	0000 _H ~ 17FF _H	○			○		
			D6144 ~ D8191	1800 _H ~ 1FFF _H	×			×		
			D8192 以上	2000 _H 以上	-			×		
			D9000 ~ D9255 (SD1000 ~ SD1255)	2328 _H ~ 2427 _H	○			○		
			(SD1256 ~ SD2047)		-			×		
链接寄存器		W0 (57 _H , 20 _H)	W0 ~ WFFF	0000 _H ~ 0FFF _H	○			○		
			W1000 ~ W1FFF	1000 _H ~ 1FFF _H	×			×		
			W2000 以上	2000 _H 以上	-			×		
文件寄存器		R0 (52 _H , 20 _H)	R0 ~ R8191	0000 _H ~ 1FFF _H	○			×		
			R8192 以上	2000 _H 以上	-			×		
定时器	当前值	TN (54 _H , 4E _H)	T0 ~ T2047	0000 _H ~ 07FF _H	○			○		
			T2048 以上	0800 _H 以上	-			×		
	触点	TS (54 _H , 53 _H)	T0 ~ T2047	0000 _H ~ 07FF _H	○			○		
			T2048 以上	0800 _H 以上	-			×		
	线圈	TC (54 _H , 43 _H)	T0 ~ T2047	0000 _H ~ 07FF _H	○			○		
T2048 以上			0800 _H 以上	-			×			
计数器	当前值	CN (43 _H , 4E _H)	C0 ~ C1023	0000 _H ~ 03FF _H	○			○		
			C1024 以上	0400 _H 以上	-			×		
	触点	CS (43 _H , 53 _H)	C0 ~ C1023	0000 _H ~ 03FF _H	○			○		
			C1024 以上	0400 _H 以上	-			×		
	线圈	CC (43 _H , 43 _H)	C0 ~ C1023	0000 _H ~ 03FF _H	○			○		
输入		X0 (58 _H , 20 _H)	X0 ~ X1FF	0000 _H ~ 01FF _H	○			○		
			X200 ~ X3FF	0200 _H ~ 03FF _H	×	○	×	○		
			X400 ~ X7FF	0400 _H ~ 07FF _H	×		○	×	○	
			X800 ~ X1FFF	0800 _H ~ 1FFF _H	×			×		
			X2000 以上	2000 _H 以上	-			×		
输出		Y0 (59 _H , 20 _H)	Y0 ~ Y1FF	0000 _H ~ 01FF _H	○			○		
			Y200 ~ Y3FF	0200 _H ~ 03FF _H	×	○	×	○		
			Y400 ~ Y7FF	0400 _H ~ 07FF _H	×	○	×	○		
			Y800 ~ Y1FFF	0800 _H ~ 1FFF _H	×			×		
			Y2000 以上	2000 _H 以上	-			×		
内部继电器		M0 (4D _H , 20 _H)	M0 ~ M8191	0000 _H ~ 1FFF _H	○		* 锁存继电器(L)/步进继电器(S) 也包含在内		○	
			M8192 以上	2000 _H 以上	-				×	
			M9000 ~ M9255 (SM1000 ~ SM1255)	2328 _H ~ 2427 _H	○				○	
			(SM1256 ~ SM2047)		-				×	
锁存继电器					(根据上述内容)		× * 即使指定锁存继电器(L)、步进继电器(S)也将访问内部继电器(M)。			
步进继电器										
链接继电器		B0 (42 _H , 20 _H)	B0 ~ BFFF	0000 _H ~ 0FFF _H	○		○			
			B1000 ~ B1FFF	1000 _H ~ 1FFF _H	×		×			
			B2000 以上	2000 _H 以上	-		×			
报警器		F0 (46 _H , 20 _H)	F0 ~ F2047	0000 _H ~ 07FF _H	○		○			
			F2048 以上	0800 _H 以上	-		×			

*1 对 Q/L/QnACPU 进行读取/写入时，存在有无法访问的软元件及可编程控制器 CPU。
有关详细内容请参阅 6.3.1 项(3)。

要点
(1) 应在访问目标可编程控制器 CPU 中可使用的软元件编号范围内进行访问。 (Q/QnACPU 时为 AnACPU 的范围) (2) 位软元件、字软元件的区分如下所示。 位软元件 : X、Y、M、L、B、F、T(触点)、T(线圈)、C(触点)、C(线圈) 字软元件 : T(当前值)、C(当前值)、D、W、R (3) 字单位指定时位软元件的起始软元件编号必须为 16 的倍数(10 进制数的情况下 0、16、. . .)。 此外, 对于特殊继电器 M 的 M9000 以后, 也可以以 (9000+16 的倍数) 进行指定。 (4) 对于除 Q/QnACPU 以外, 对 M、L、S 进行访问时应指定 M 及访问对象软元件编号。(对 L100 进行访问时, 对 M100 进行指定。) (5) 特殊继电器 (M9000~M9255)、特殊寄存器 (D9000~D9255) 分为只读、写入专用、系统用。 进行了超出写入允许范围的写入时有可能导致可编程控制器 CPU 发生出错。 关于特殊继电器、特殊寄存器的详细内容请参阅 ACPU 编程手册。 (6) 使用 AnACPU、AnUCPU 扩展文件寄存器用专用指令的情况下, 文件寄存器 (R) 的读取・写入应通过 6.4 节中说明的指令进行。 (7) 对于通过执行指令进行读取/写入的软元件点数, 应在 6.3.1 项的表中所示的处理点数(1 次通信中可处理的点数)以内进行指定。 但是, 仅在指定 256 点时, 以 00H/“00”进行指定。 (示例: 引号内表示通过 ASCII 代码进行通信时的指定数据。) 5 点的情况下 : 05H/“05” 10 点的情况下 : 0AH /“0A” 20 点的情况下 : 14H/“14” 256 点的情况下 : 00H/“00”

(3) 对 Q/L/QnACPU 进行读取/写入时的注意事项

- (a) 只有与 AnCPU、AnNCPU、AnACPU、AnUCPU 中存在的软元件相同名称的软元件, 才可以在 AnACPU 的软元件范围内进行读取/写入。
对于以下软元件, 不能从对象设备进行访问。
 - Q/QnACPU 中新增的软元件
 - 锁存继电器 (L) 以及步进继电器 (S)*1
 - 文件寄存器 (R)

*1 Q/QnACPU 的情况下, 对于内部继电器 (M) 及其它软元件的锁存继电器 (L) 以及步进继电器 (S), 不能指定为访问对象软元件。
- (b) 对特殊继电器、特殊寄存器的访问如下所示。
 - 通过 M9000~M9255 的指定, 对 SM1000~SM1255 进行访问
 - 通过 D9000~D9255 的指定, 对 SD1000~SD1255 进行访问
- (c) 对通用型 QCPU 进行访问的情况下, 应使用序列号的前 5 位数为 10102 以后的通用型 QCPU。
序列号的前 5 位数为 10101 以前的情况下, 应以 QnA 兼容 3E 帧、4E 帧进行访问。
此外, 对 CPU 模块的内置以太网端口进行访问的情况下, 请参阅以下手册。
 - QnUCPU 用户手册(内置以太网端口通信篇)

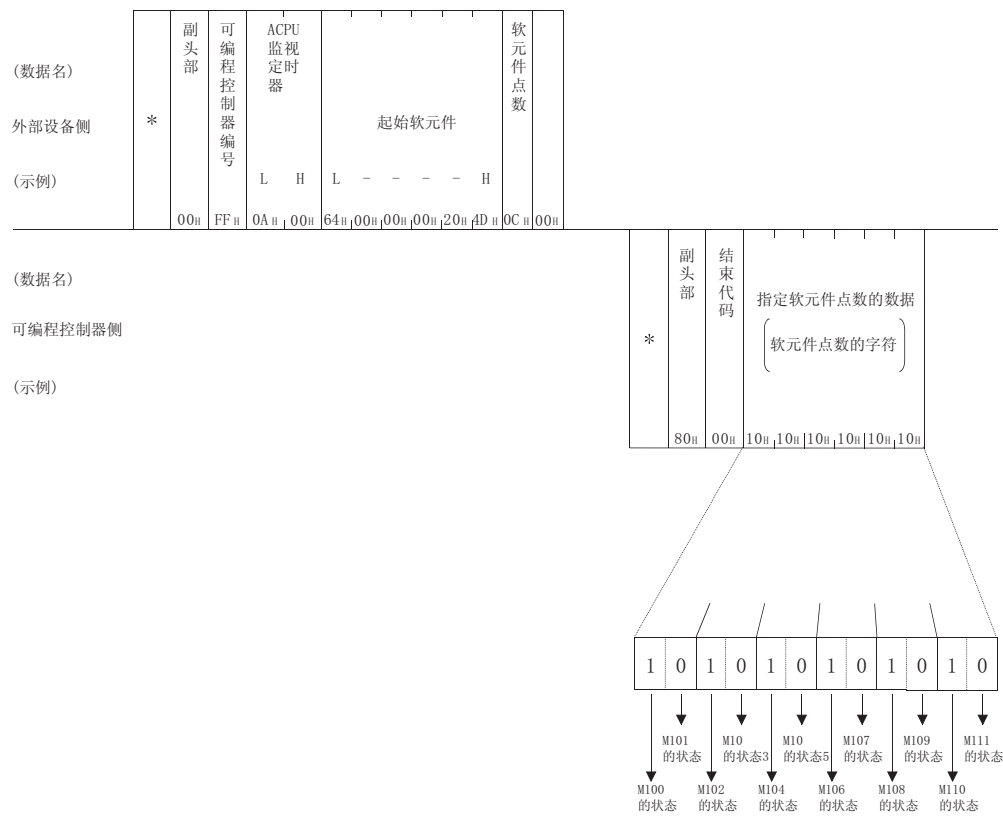
6.3.2 位单位的批量读取(指令：00)

以下对进行位软件元件存储器的批量读取时的指令/响应格式进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 M100～M111 的 ON/OFF 状态进行读取的情况下。

(1) 通过二进制代码进行通信时



备注

将软元件点数置为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时

[illegible][illegible]

备注

- (1) 将软件点数置为 256 点的情况下，以“00”进行指定。
- (2) 将软件点数以奇数进行指定时，响应数据中将附加 1 字节的虚拟数据(30H)。例如，读取 3 点时将返回 4 点的数据。最后的 1 字节为虚拟数据。

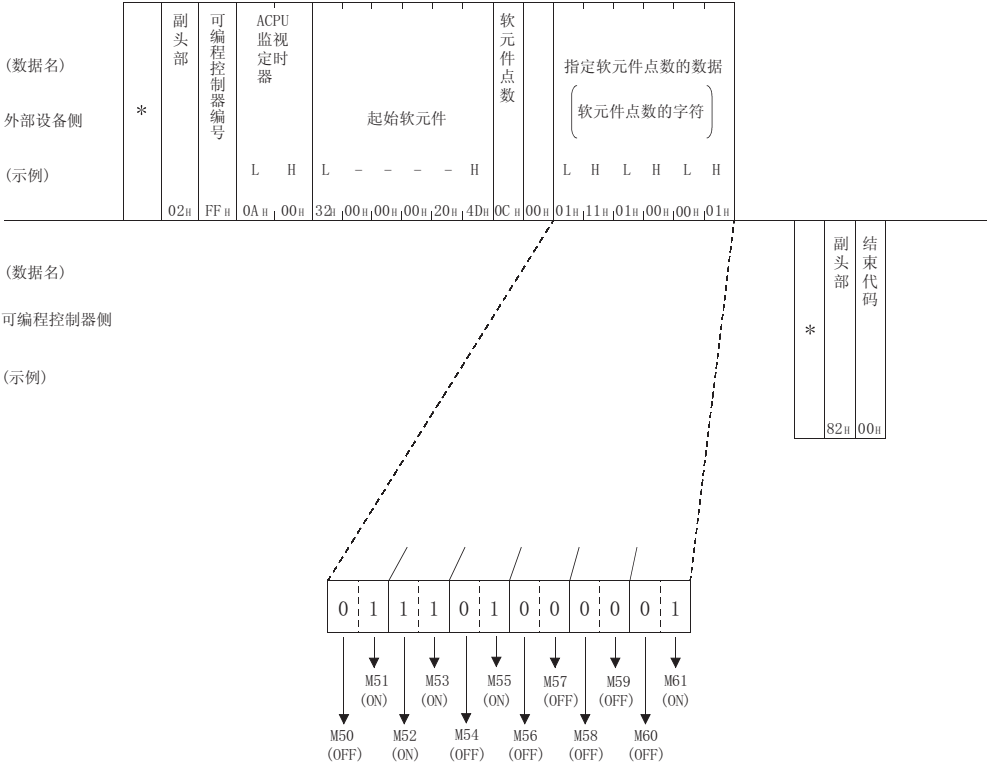
6.3.3 位单位的批量写入(指令：02)

以下对进行位软元件存储器的批量写入时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 M50～M61 进行 ON/OFF 状态的写入的情况下。

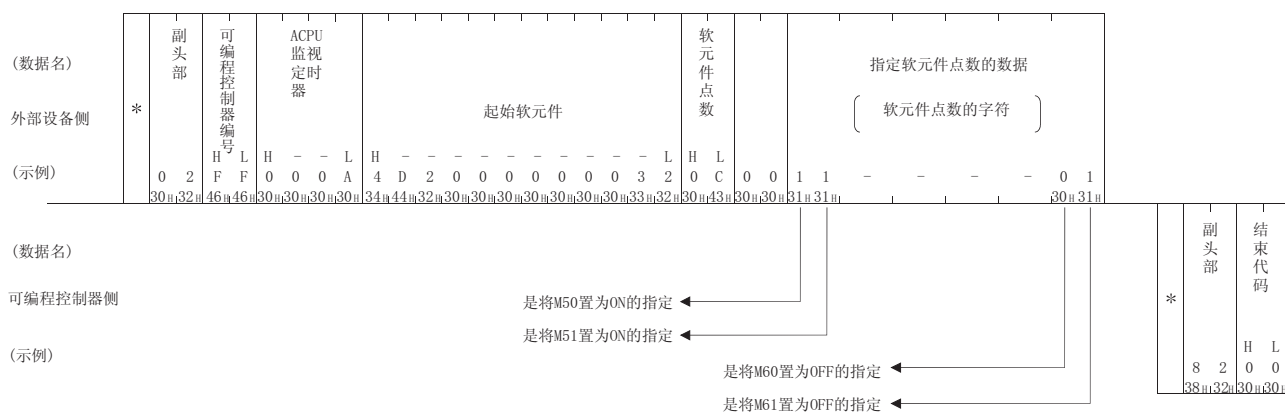
(1) 通过二进制代码进行通信时



备注

将软元件点数为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

- (1) 将软元件点数为 256 点的情况下，以“00”进行指定。
- (2) 将软元件点数以奇数进行指定时，应在写入数据的最后附加 1 字节的虚拟数据 (30H)。
例如，写入 3 点时应在最后附加虚拟数据 (30H)。

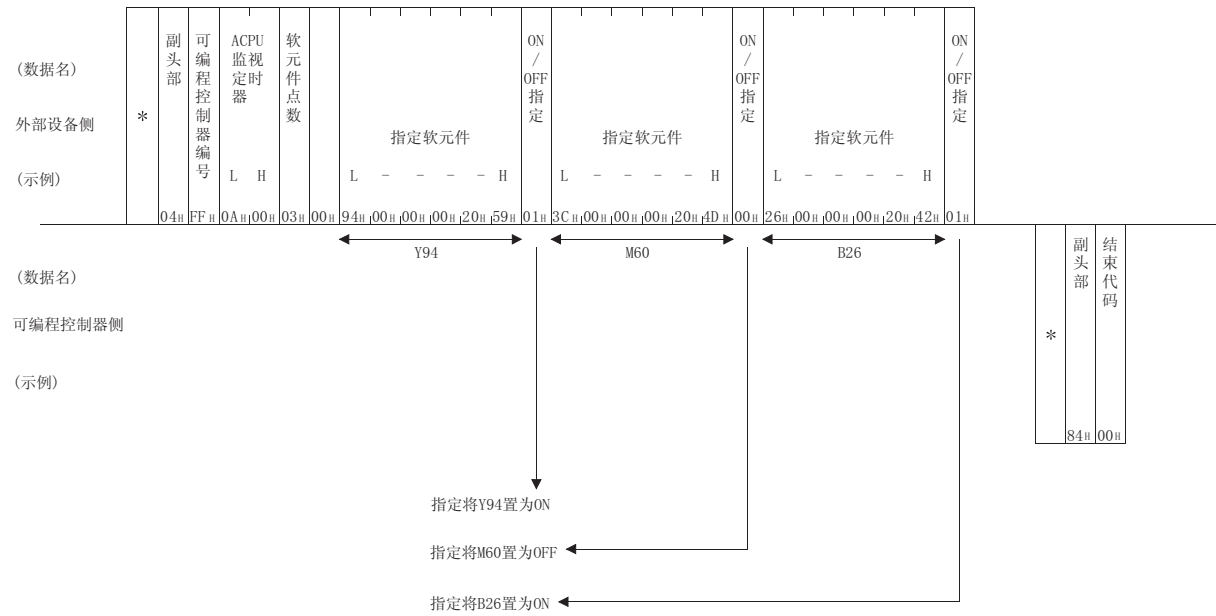
6.3.4 位单位的测试(随机写入)(指令：04)

以下对随机指定位软元件存储器进行数据写入时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

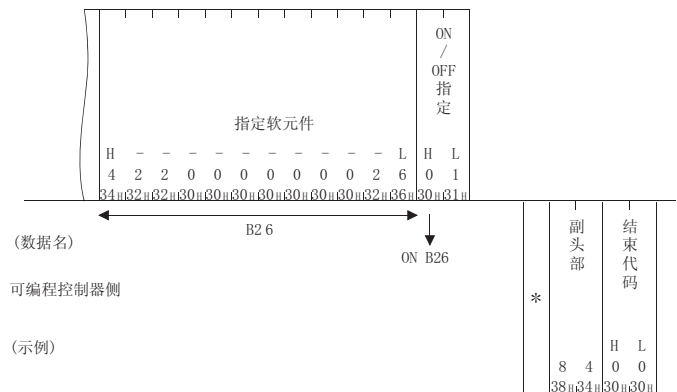
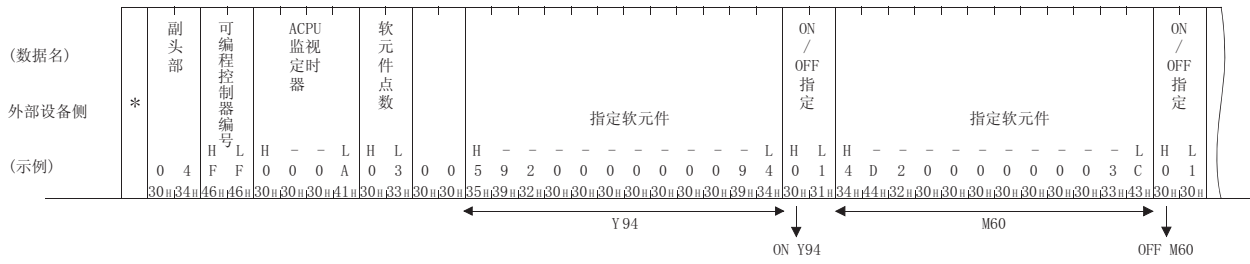
[控制步骤]

将安装了 E71 的可编程控制器 CPU 的 Y94 置为 ON、M60 置为 OFF、B26 置为 ON 的情况下。

(1) 通过二进制代码进行通信时



(2) 通过 ASCII 代码进行通信时



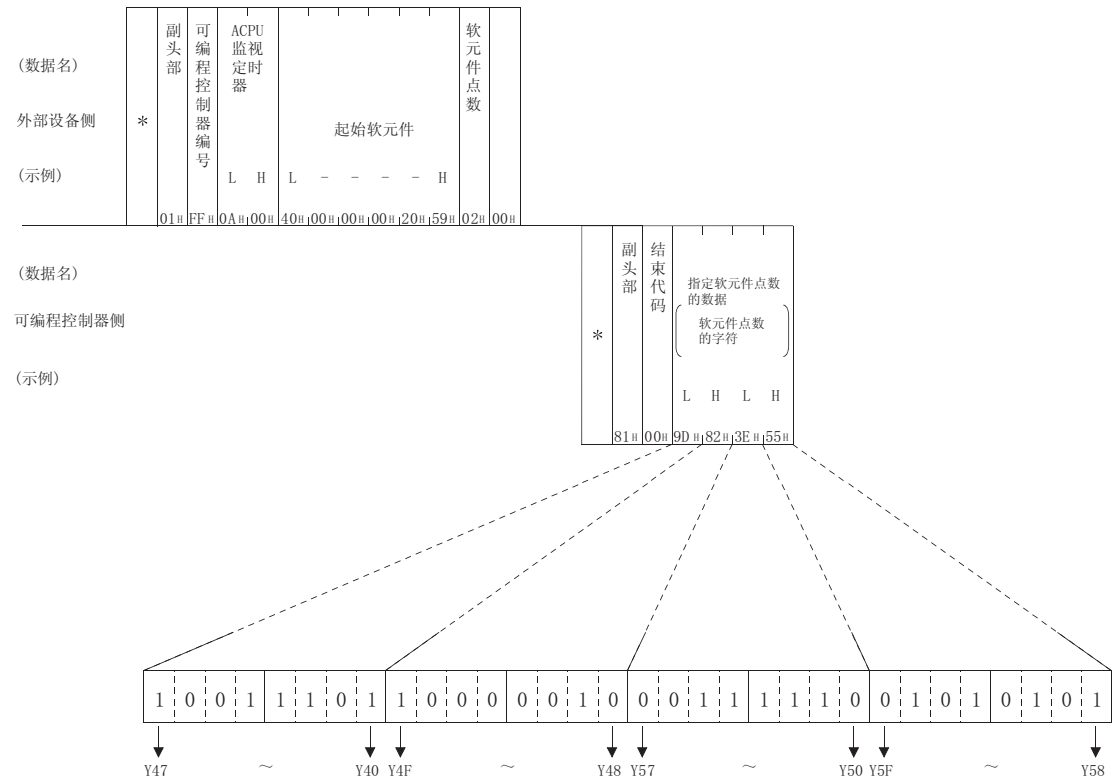
6.3.5 字单位的批量读取(指令：01)

以下对进行字软元件存储器的批量读取以及位软元件存储器(16 单位)的批量读取时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 Y40～5F(32 点)的 ON/OFF 状态进行读取的情况下。

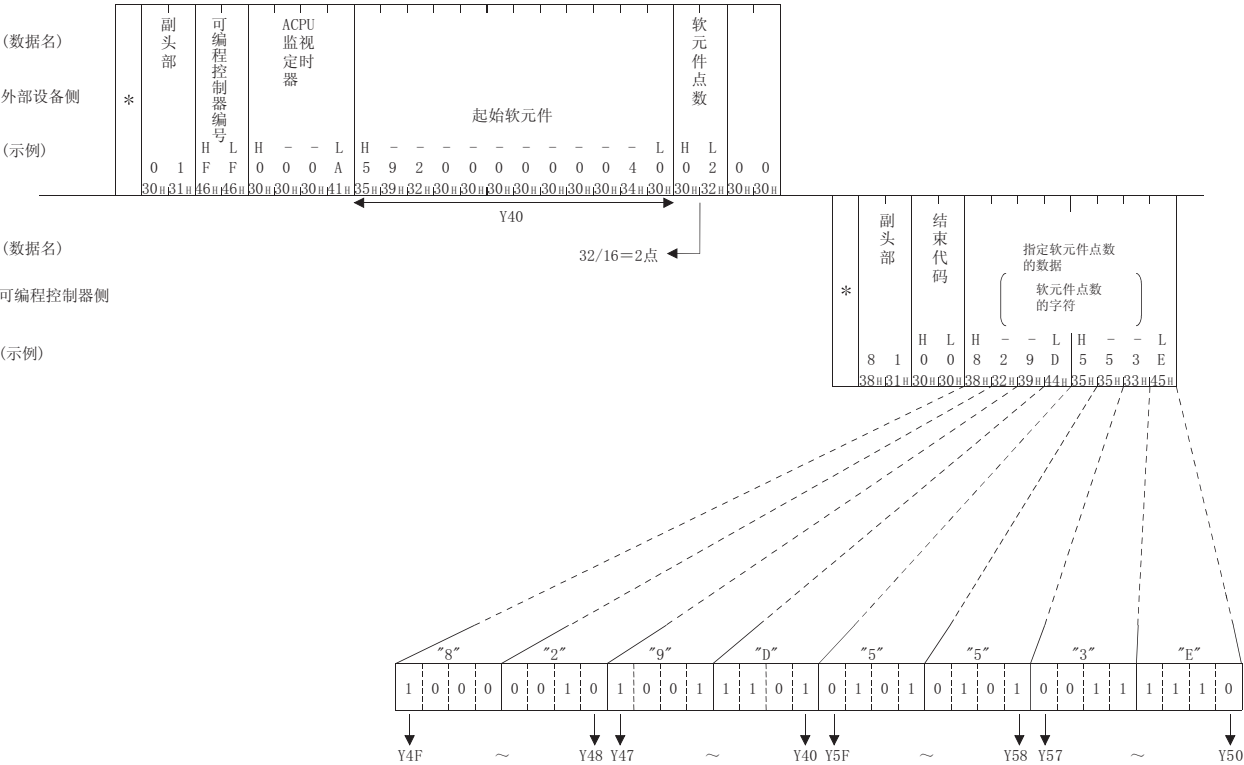
(1) 通过二进制代码进行通信时



备注

- 将软元件点数置为 256 点的情况下，以“00H”进行指定。
- 指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(2) 通过 ASCII 代码进行通信时



备注

- 将软元件点数为 256 点的情况下，以“00h”进行指定。
- 指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

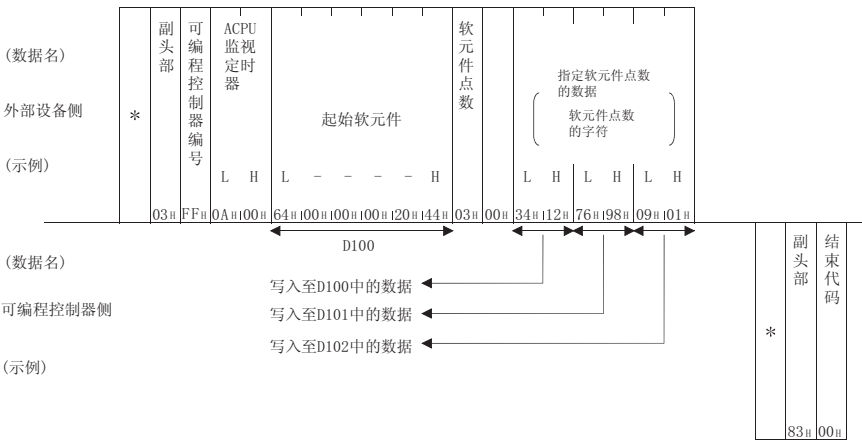
6.3.6 字单位的批量写入(指令：03)

以下对进行字软元件存储器的批量写入以及位软元件存储器(16 点单位)的批量写入时指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 D100~102 进行数据写入的情况下。

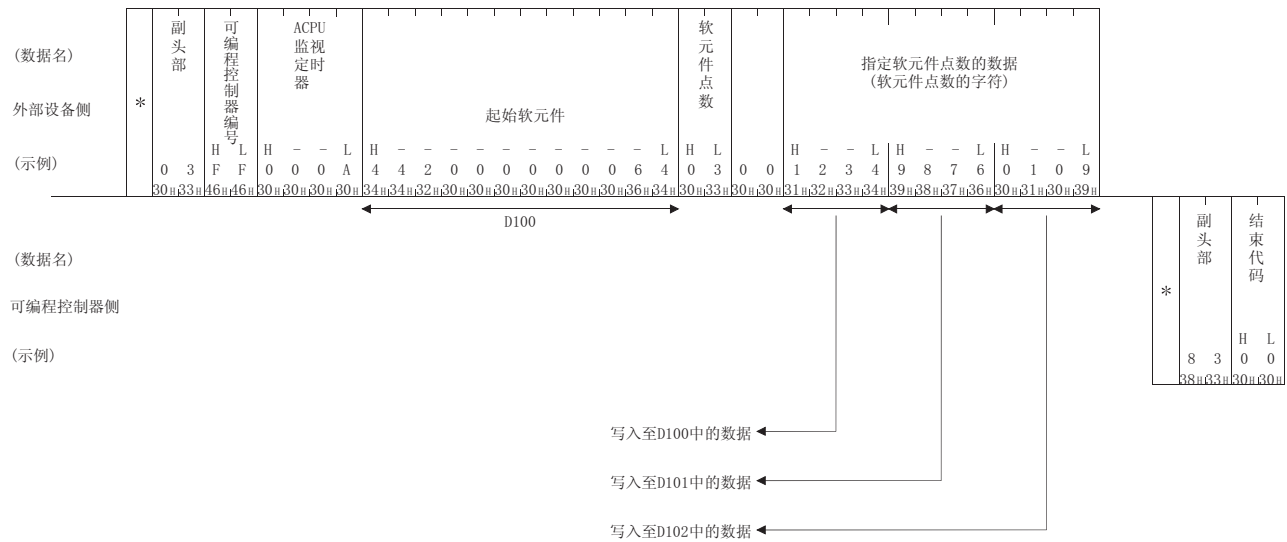
(1) 通过二进制代码进行通信时



备注

- 将软元件点数置为 256 点的情况下，以“00H”进行指定。
- 指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(2) 通过 ASCII 代码进行通信时



备注

- 将软元件点数置为 256 点的情况下，以“00H”进行指定。
- 指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

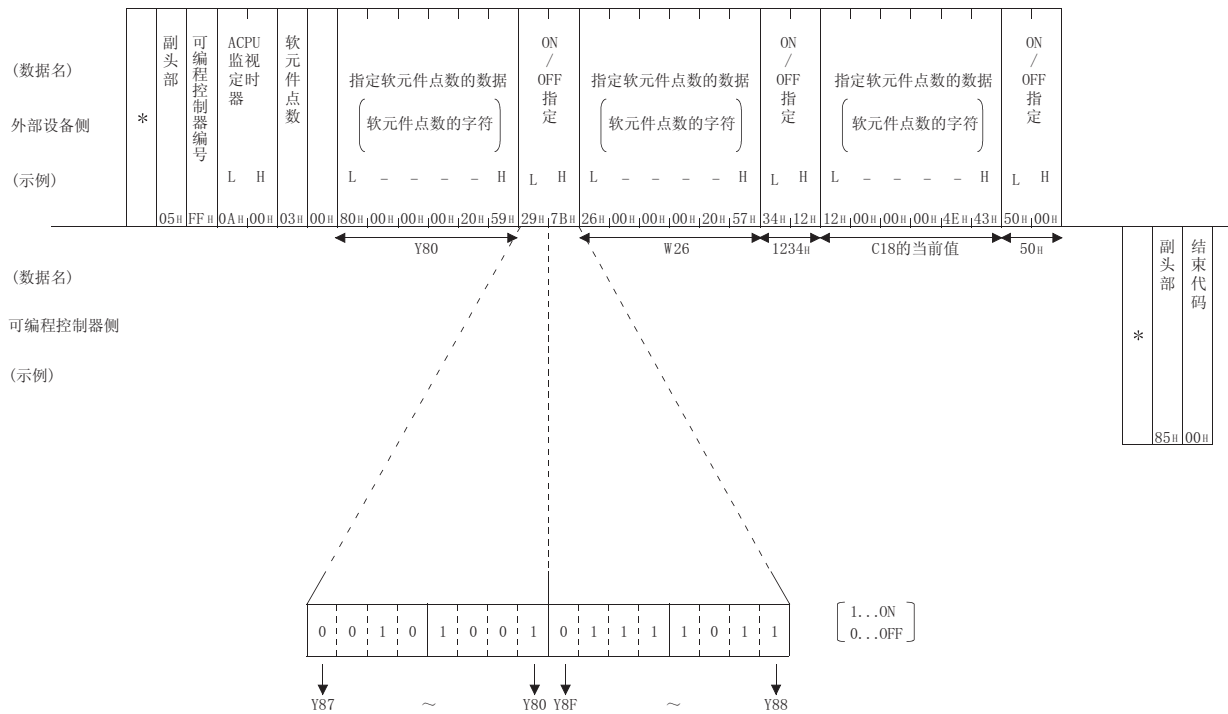
6.3.7 字单位的测试(随机写入)(指令：05)

以下对随即指定字软元件存储器以及位软元件存储器(16 点单位)进行写入时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 Y80~8F 进行 ON/OFF 改写、将 W26 改写为“1234_h”、将 C18 的当前值改写为“50_h”的情况下。

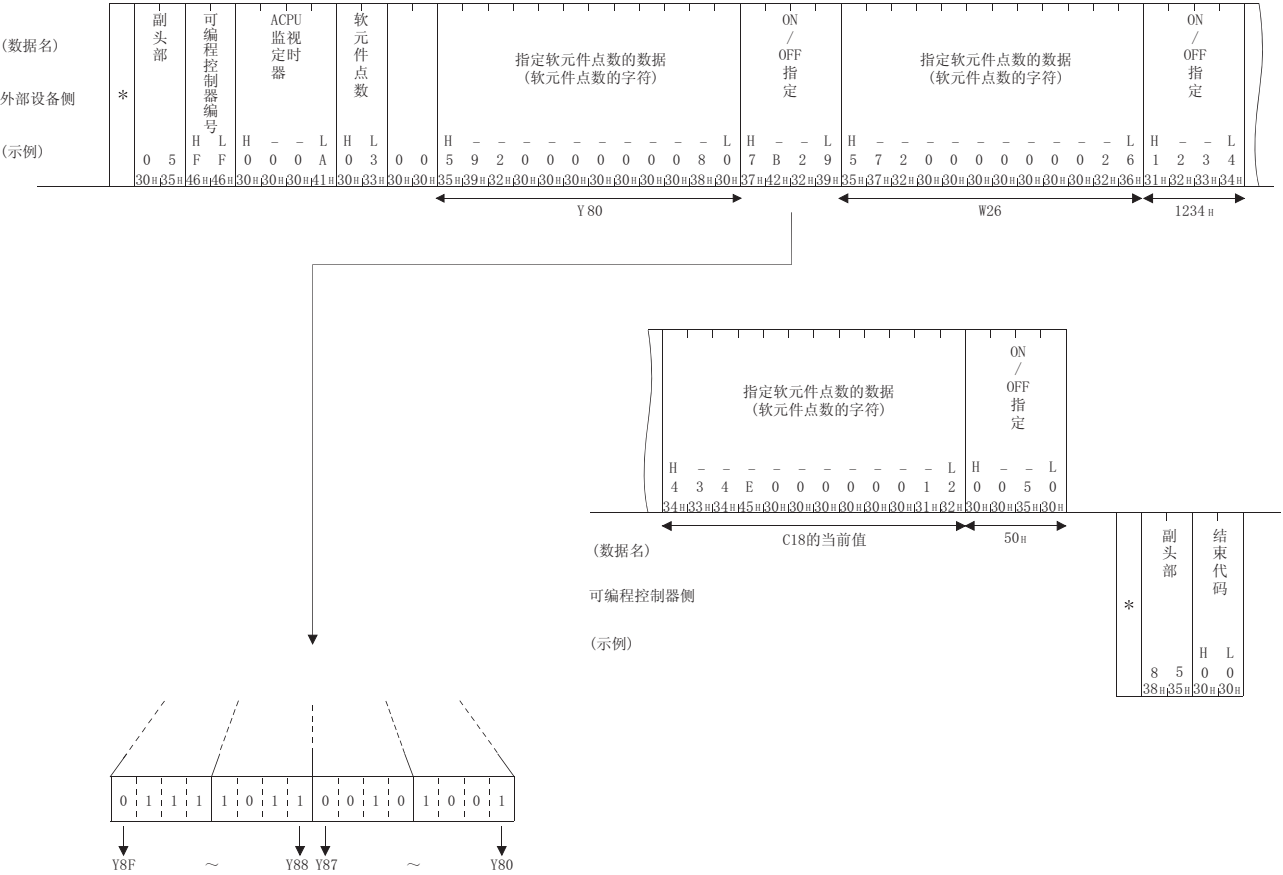
(1) 通过二进制代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(2) 通过 ASCII 代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数 (10 进制数的情况下为 0、16、...)。

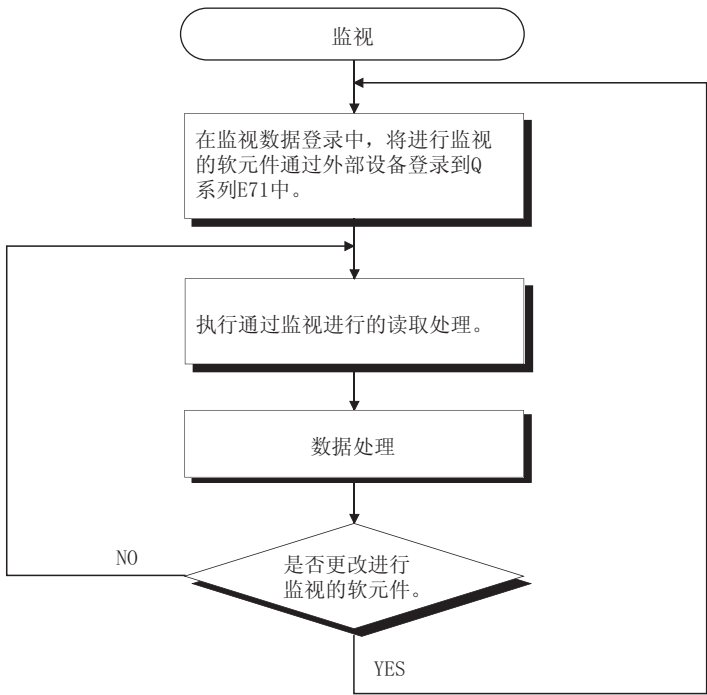
6.3.8 软元件存储器的监视

通过将希望在外设上进行监视的软元件以及软元件编号预先登录到 E71 中，从外部设备执行监视指令，可以将可编程控制器 CPU 内的软元件的 ON/OFF 状态或内容通过外部设备进行监视。

在通过软元件存储器的批量读取进行的读取中，以连续的软元件编号进行处理，而在通过监视进行的读取中，可以随机指定任意的软元件以及编号进行读取。

(1) 监视的操作步骤

进行监视时的操作步骤如下所示。



要点	
(1)	按上述操作步骤执行监视时必须进行监视数据登录操作。如果在未进行监视数据登录的状况下执行监视，将变为出错状态。
(2)	在进行了电源 OFF 或可编程控制器 CPU 的复位的情况下，监视数据登录的内容将被删除。
(3)	进行监视数据登录时，可以将软元件存储器位单位、字单位以及扩展文件寄存器的 3 种类型的指令中的最后登录的 1 个指令预先登录到 E71 中。
(4)	从多个外部设备对同一站的可编程控制器 CPU 的软元件存储器进行监视数据登录时，登录数据将被覆盖登录，因此最后登录的软元件存储器将有效。
(5)	关于扩展文件寄存器的监视，请参阅 6.4.6 项。

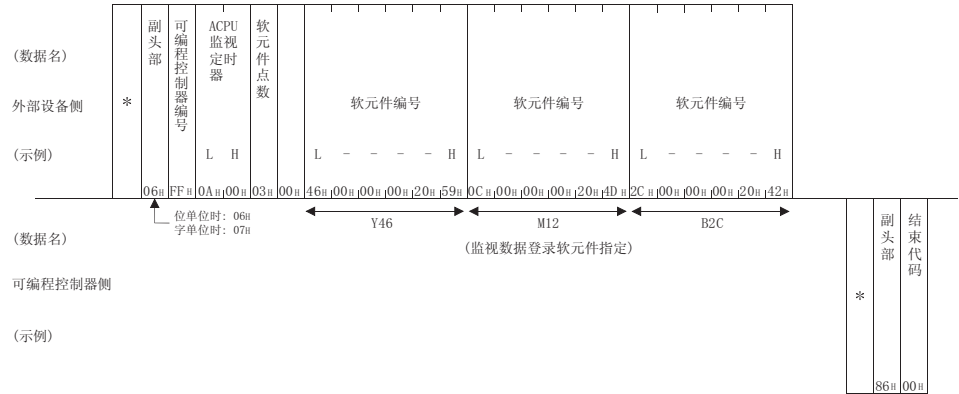
(2) 监视数据登录(指令：06、07)

以下对要监视的软元件进行登录时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所
示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的 Y46、M12、B2C 进行登录的情况下。

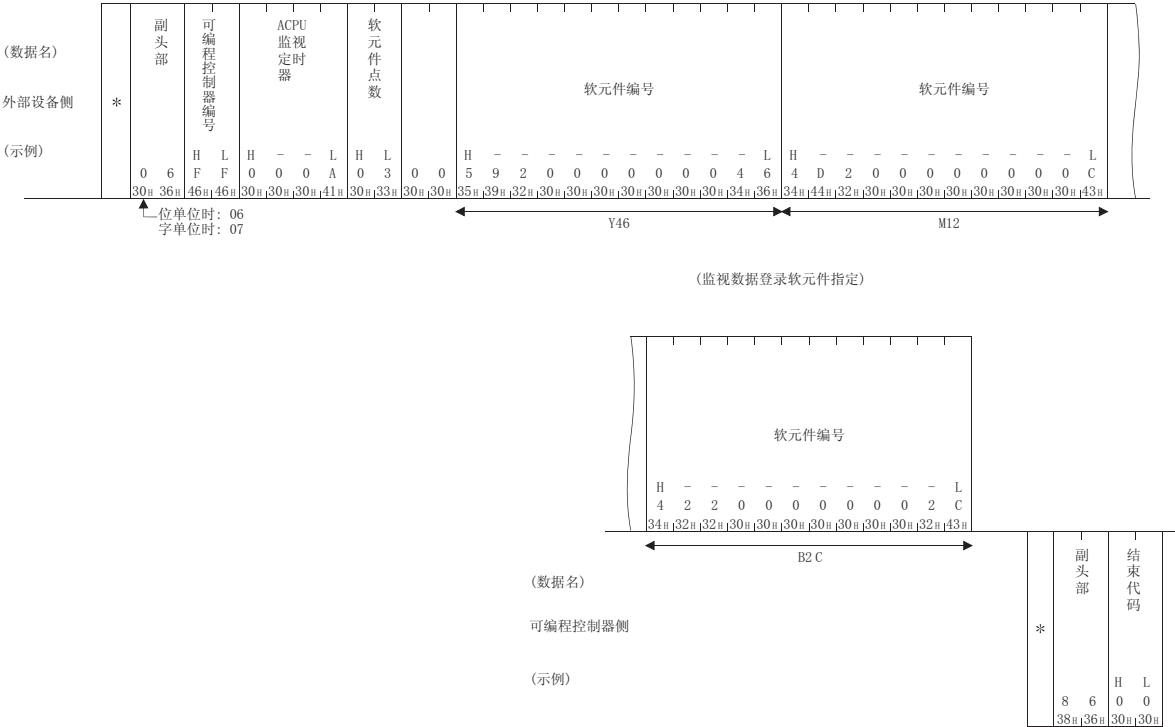
(a) 通过二进制代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(b) 通过 ASCII 代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

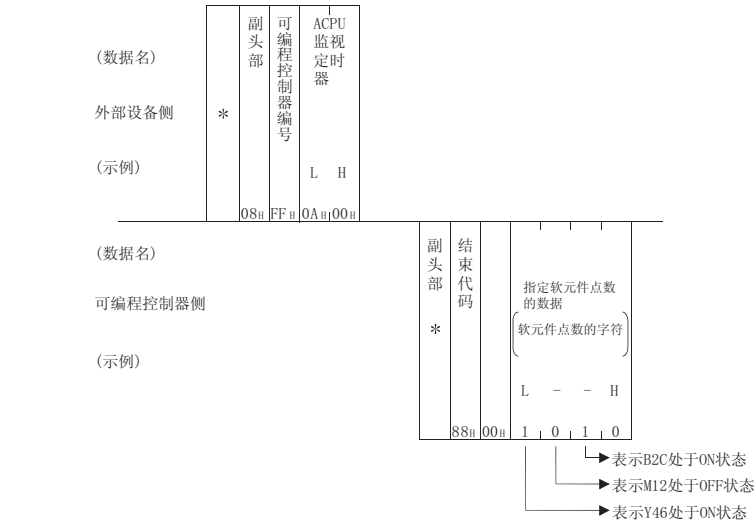
(3) 位单位的监视(指令：08)

以下介绍对进行了监视数据登录的位软元件进行监视时的指令/响应格式有关内容。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

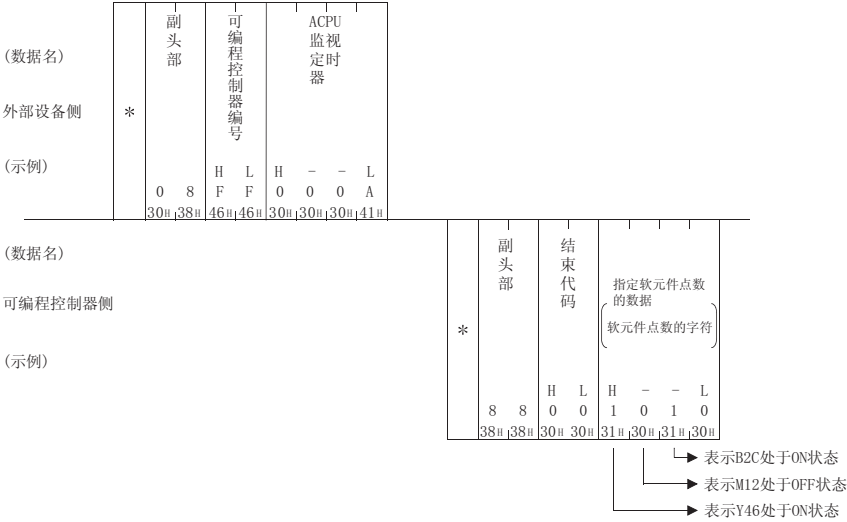
[控制步骤]

对安装了 E71 的可编程控制器 CPU 中进行了监视数据登录的“Y46”、“M12”、“B2C”进行监视的情况下。

(a) 通过二进制代码进行通信时



(b) 通过 ASCII 代码进行通信时



备注

监视登录的软元件点数为奇数时，执行监视时将被附加虚拟数据(30H)。
例如，监视登录的软元件点数为 3 点时，将返回 4 点的数据。最后 1 字节为虚拟数据。

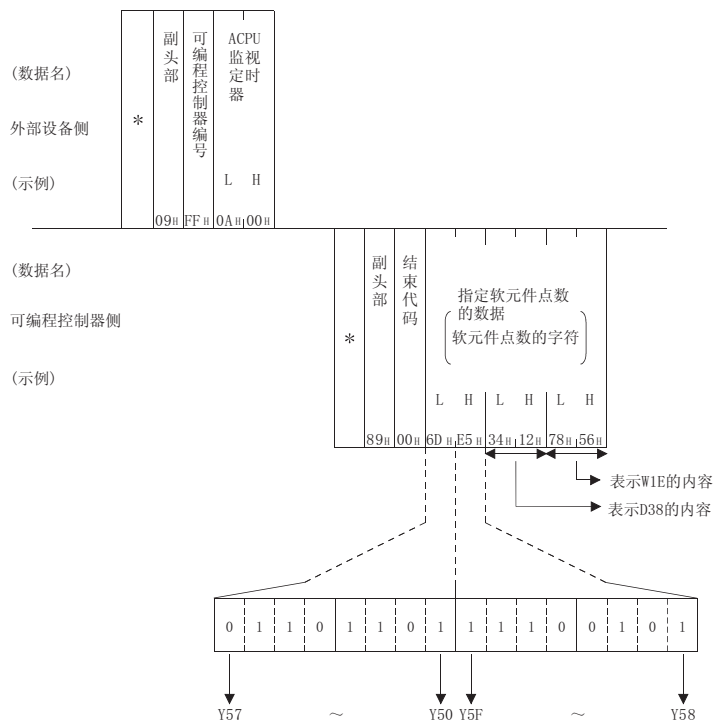
(4) 字单位的监视(指令：09)

以下对进行了监视数据登录的字软元件、以及位软元件(16 点单位)进行监视时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的进行了监视数据登录的“Y50～5F”、“D38”、“W1E”进行监视的情况下。

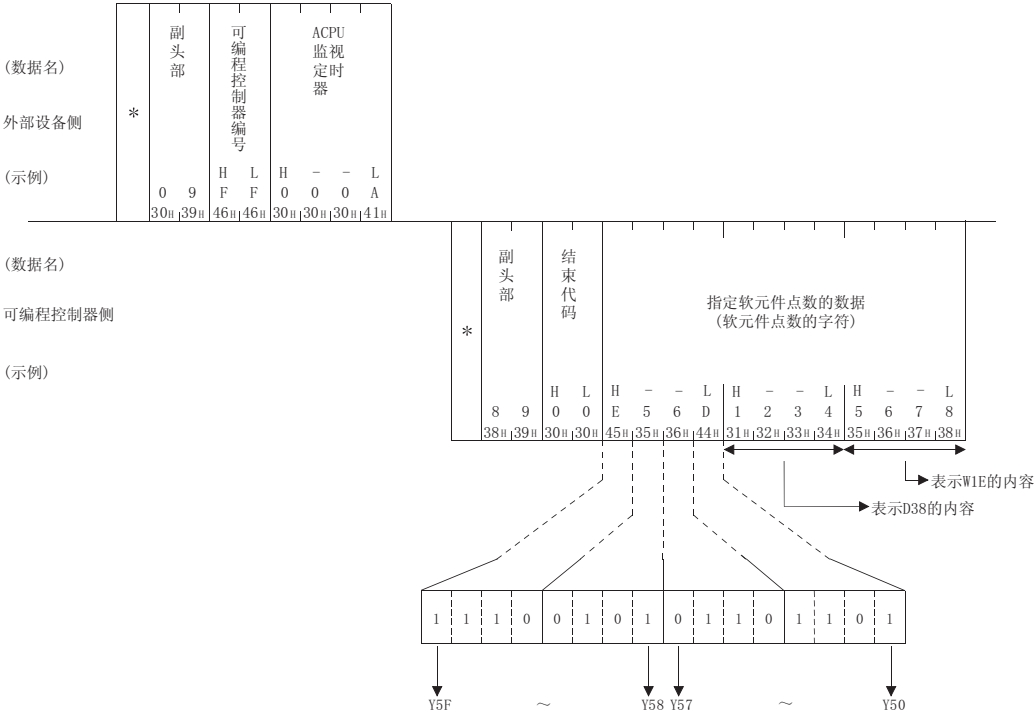
(a) 通过二进制代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

(b) 通过 ASCII 代码进行通信时



备注

指定位软元件的情况下，起始软元件 No. 必须设置为 16 的倍数(10 进制数的情况下为 0、16、...)。

6.4 扩展文件寄存器的读取、写入

扩展文件寄存器是指，将可编程控制器 CPU 的用户存储器区域的空余区域作为文件寄存器使用的寄存器，是使用扩展文件寄存器用软件包“SWOGHP-UTLPC-FN1、SWOSRX-FNUP”（以下略称为 UTLPC-FN1、FNUP。）进行各种数据处理以及 AnACPU、AnUCPU 的扩展文件寄存器用专用指令中，用于存储必要数据以及运算结果的存储器区域。
以下对进行扩展文件寄存器的读取、写入等的控制步骤进行举例说明。

6.4.1 ACPU 公共指令及地址

(1) 用于扩展文件寄存器的读取、写入等的 ACPU 公共指令如下所示。

项目	指令/响应格式	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态		
				STOP 中	RUN 中	
					写入允许设置	写入禁止设置
批量读取	17n	对扩展文件寄存器(R)以1点为单位进行读取。	256 点	○	○	○
批量写入	18n	对扩展文件寄存器(R)以1点为单位进行写入。	256 点	○	○	×
测试(随机写入)	19n	对扩展文件寄存器(R)以1点为单位随机指定块编号・软元件编号进行写入。	40 点	○	○	×
监视数据登录	1An	对进行监视的软元件编号以1点为单位进行登录。	20 点	○	○	○
监视	1Bn	对进行了监视数据登录的扩展文件寄存器进行监视。	-	○	○	○

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 扩展文件寄存器的地址

- (a) 扩展文件寄存器中，有 0~n(n 根据存储卡盒而有所不同)号块，0 号块中有通过可编程控制器 CPU 参数指定的点数的寄存器，块编号 1~n 的各块中有 8192 点的寄存器。
但是，可通过可编程控制器 CPU 进行读取、写入的范围为 0 号块的参数中指定的点数的范围。
- (b) 可指定的块号范围根据存储卡盒的类型以及可编程控制器 CPU 的参数设置而变化。
详细内容请参阅 UTLPC-FN1 或 FNUP 的操作手册或 AnACPU、AnUCPU 的用户手册。

6. 4. 2 AnA/AnUCPU 公共指令及软元件编号

(1) 扩展文件寄存器的直接读取、直接写入中使用的 AnA/AnUCPU 公共指令如下所示。

该指令的功能为，在对块编号 1~256 的扩展文件寄存器进行访问时无需在意各块编号，将从块编号 1 的软元件编号 0 开始的地址指定为软元件编号进行访问。（对可使用的块的个数×8192 点的扩展文件寄存器以连续的软元件编号进行访问。）

项目	指令/响应格式	处理内容	1 次通信中可处理的点数	顺控程序 CPU 的状态		
				STOP 中	RUN 中	
					写入允许设置	写入禁止设置
直接读取	3B _H	对扩展文件寄存器(R)以 1 点为单位进行读取。	256 点	○	○	○
直接写入	3C _H	对扩展文件寄存器(R)以 1 点为单位进行写入。	256 点	○	○	×

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 扩展文件寄存器的软元件编号

可指定的软元件编号的范围如下所示。

0~(使用可能な块数×8192)-1

6. 4. 1项中所示的ACPU公共指令中指定的软元件编号

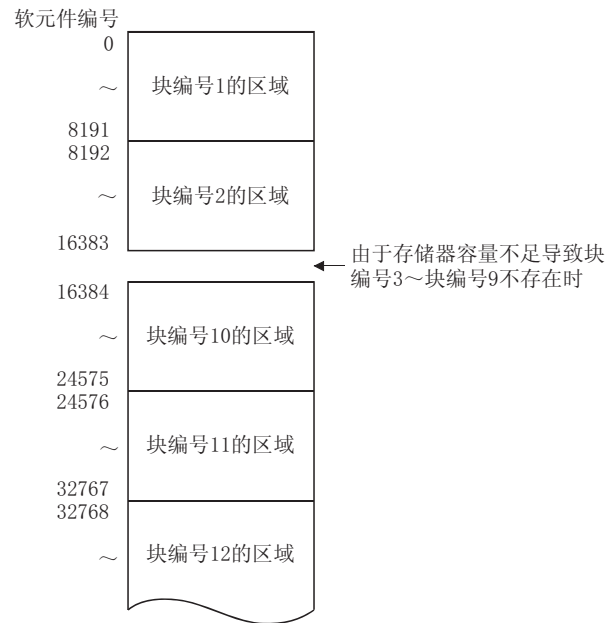


6. 4. 2项中所示的AnA/AnUCPU公共指令中指定的软元件编号



对于软元件编号，从块编号1~块编号256中块编号较小的软元件开始按编号顺序自动分配。

此外，可指定的软元件编号根据存储卡盒的类型以及可编程控制器 CPU 的参数设置而变化。
(详细内容请参阅 UTLP-FN1 或 FNUP 的操作手册或访问目标可编程控制器 CPU 的用户手册。)
对于存储卡盒内不存在的块编号，不进行软元件编号分配。如下所示，将跳过存储卡盒内不存在的块编号进行软元件编号的自动分配。



要点	
(1)	对于 AnA/AnUCPU 共用指令，只有在对块编号 1~256 的扩展文件寄存器进行数据的读取写入时才可以使用。 此外，可在与参数的文件寄存器设置的有无无关的状况下使用。
(2)	对参数中设置的文件寄存器 (R) 进行访问时，或指定块编号进行访问时，应使用 6. 4. 1 项中所示的指令。
(3)	AnA/AnUCPU 公共指令中指定的起始软元件编号的计算公式如下所示。 指定从起始开始的第 n 块的软元件编号 m (0~8191) 的情况下 起始软元件编号 = (n - 1) × 8192 + m

备注

使用 AnA/AnUCPU 公共指令 (3BH、3CH) 时指定的软元件编号范围分为 28 块，各块如下所示。

软元件编号	对象块的位置	软元件编号	对象块的位置
0 ～ 8191	第 1 块 R0 ～ R8191	114688 ～ 122879	第 15 块 R0 ～ R8191
8192 ～ 16383	第 2 块 R0 ～ R8191	122880 ～ 131071	第 16 块 R0 ～ R8191
16384 ～ 24575	第 3 块 R0 ～ R8191	131072 ～ 139263	第 17 块 R0 ～ R8191
24576 ～ 32767	第 4 块 R0 ～ R8191	139264 ～ 147455	第 18 块 R0 ～ R8191
32768 ～ 40959	第 5 块 R0 ～ R8191	147456 ～ 155647	第 19 块 R0 ～ R8191
40960 ～ 49151	第 6 块 R0 ～ R8191	155648 ～ 163839	第 20 块 R0 ～ R8191
49152 ～ 57343	第 7 块 R0 ～ R8191	163840 ～ 172031	第 21 块 R0 ～ R8191
57344 ～ 65535	第 8 块 R0 ～ R8191	172032 ～ 180223	第 22 块 R0 ～ R8191
65536 ～ 73727	第 9 块 R0 ～ R8191	180224 ～ 188415	第 23 块 R0 ～ R8191
73728 ～ 81919	第 10 块 R0 ～ R8191	188416 ～ 196607	第 24 块 R0 ～ R8191
81920 ～ 90111	第 11 块 R0 ～ R8191	196608 ～ 204799	第 25 块 R0 ～ R8191
90112 ～ 98303	第 12 块 R0 ～ R8191	204800 ～ 212991	第 26 块 R0 ～ R8191
98304 ～ 106495	第 13 块 R0 ～ R8191	212992 ～ 221183	第 27 块 R0 ～ R8191
106496 ～ 114687	第 14 块 R0 ～ R8191	221184 ～ 229375	第 28 块 R0 ～ R8191

6.4.3 扩展文件寄存器读取、写入时的注意事项

以下对根据 6.4.4 项~6.4.9 项中所示的指令进行扩展文件寄存器的读取、写入等时的注意事项有关内容说明。

- (1) 可以对支持扩展文件寄存器的可编程控制器 CPU 进行访问。
对于不支持扩展文件寄存器的可编程控制器 CPU(A1N 等)，不能使用本功能。
- (2) 即使在指定了不存在的块编号进行了读取、写入的情况下，根据可编程控制器 CPU 中安装的存储卡盒的类型，也可能无法检测出出错(字符部分出错 58H)。在这种情况下，读取的数据将为不正确的数据。此外，如果进行写入，有可能会损坏可编程控制器 CPU 的用户存储器。
应对存储卡盒的类型、参数设置内容等进行确认之后再执行本功能。

存储卡盒 型号	可能无法检测出字符部分出错(58H)的块编号		
	A0J2H、A2、A3CPU	A2N、A3NCPU	A3H、AnA、AnUCPU
A3NMCA-12	No. 10 ~ No. 11		
A3NMCA-18	-	No. 10 ~ No. 28	
A3NMCA-24	-	No. 13 ~ No. 20	No. 13 ~ No. 28
A3NMCA-40	-		No. 21 ~ No. 28
A3AMCA-96	-		No. 21 ~ No. 48(*1)

*1 A3AMCA-96 可用于 A3A、A3U、A4UCPU。

(详细内容请参阅 UTLP-FN1 或 FNUP 的手册或访问目标可编程控制器 CPU 的用户手册。)

- (3) A2USCPU(S1)中可处理的扩展文件寄存器的块编号如下所示。
 - A2USCPU : No. 1~3
 - A2USCPU-S1 : No. 1~8、No. 10~16
- (4) 不能对 Q/QnACPU 的扩展文件寄存器进行读取/写入。

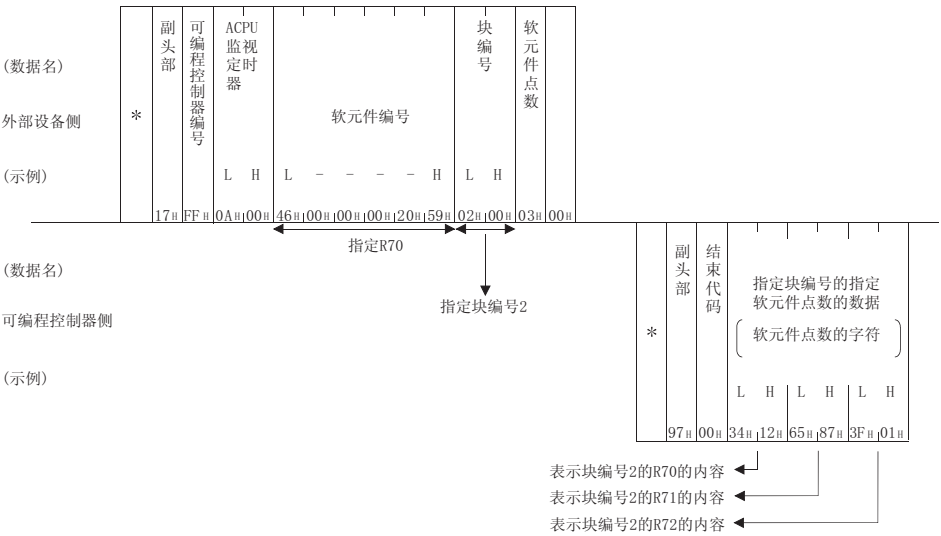
6.4.4 扩展文件寄存器的批量读取(指令：17)

以下对进行扩展文件寄存器的批量读取时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 中的扩展文件寄存器的块编号 2 的 R70~72 的内容进行读取的情况下。

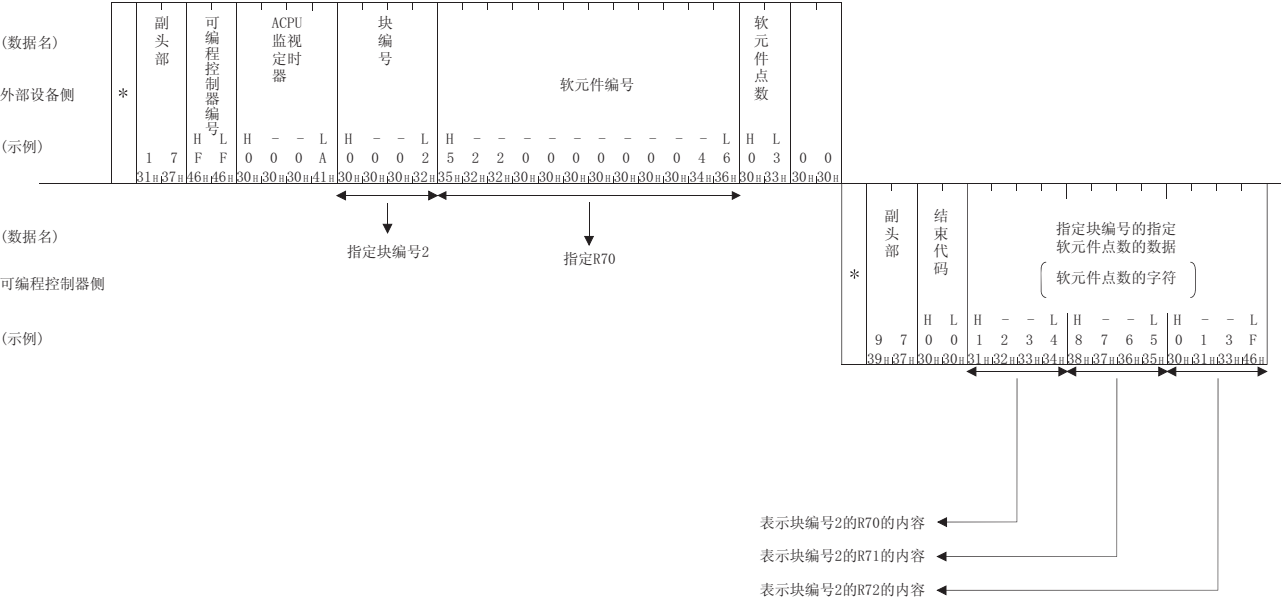
(1) 通过二进制代码进行通信时



备注

将软元件点数置为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将软元件点数置为 256 点的情况下，以“00”进行指定。

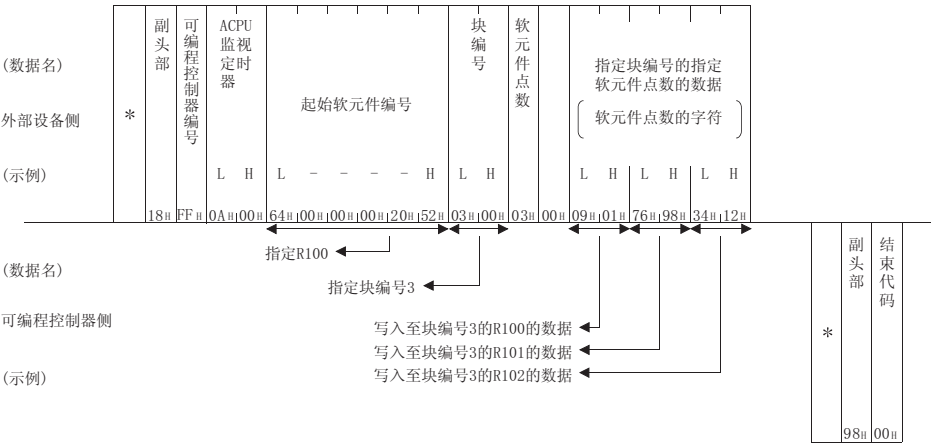
6.4.5 扩展文件寄存器的批量写入(指令：18)

以下对进行扩展文件寄存器的批量写入时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 中的扩展文件寄存器块编号 3 的 R100~102 进行数据写入的情况下。

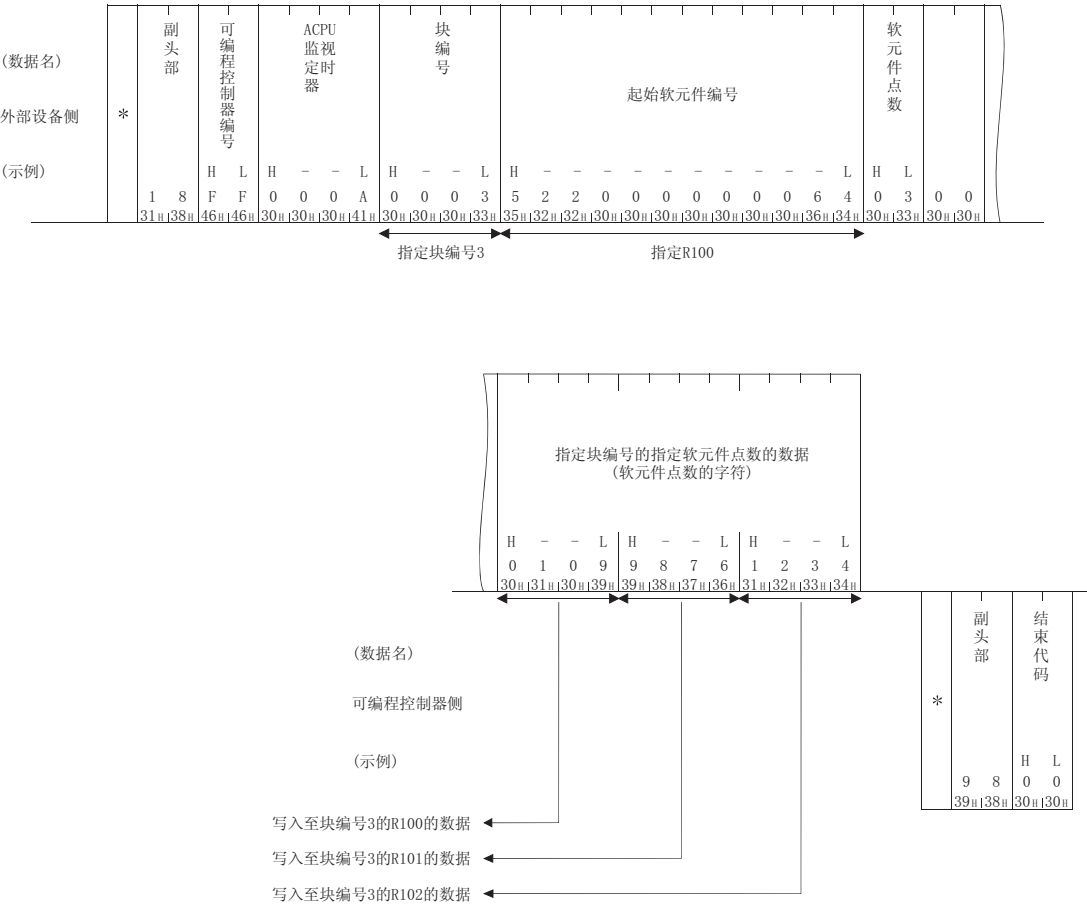
(1) 通过二进制代码进行通信时



备注

将软件元件点数置为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将软元件点数置为 256 点的情况下，以“00”进行指定。

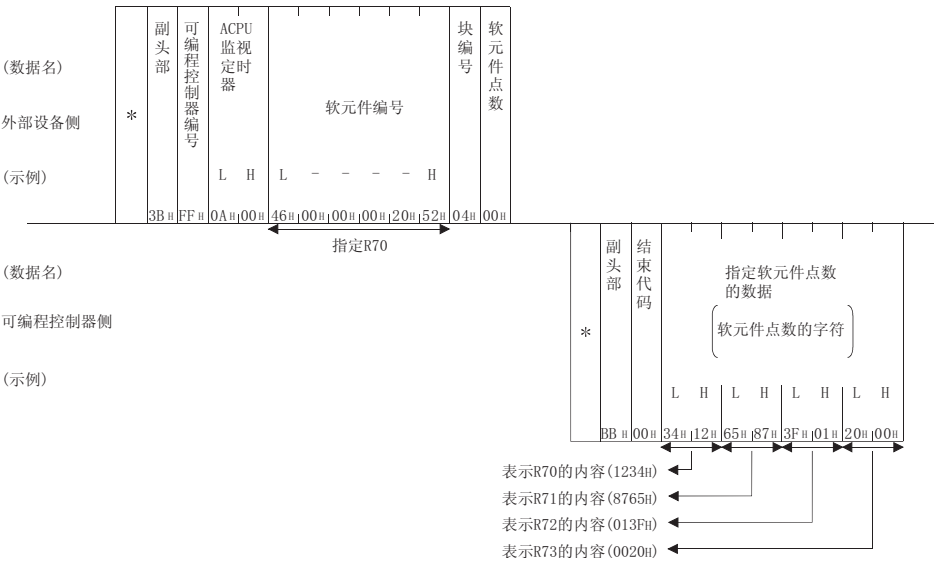
6.4.6 扩展文件寄存器的直接读取(指令：3B)

以下对进行扩展文件寄存器的直接读取时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的扩展文件寄存器 R70~73 的内容进行读取的情况下。

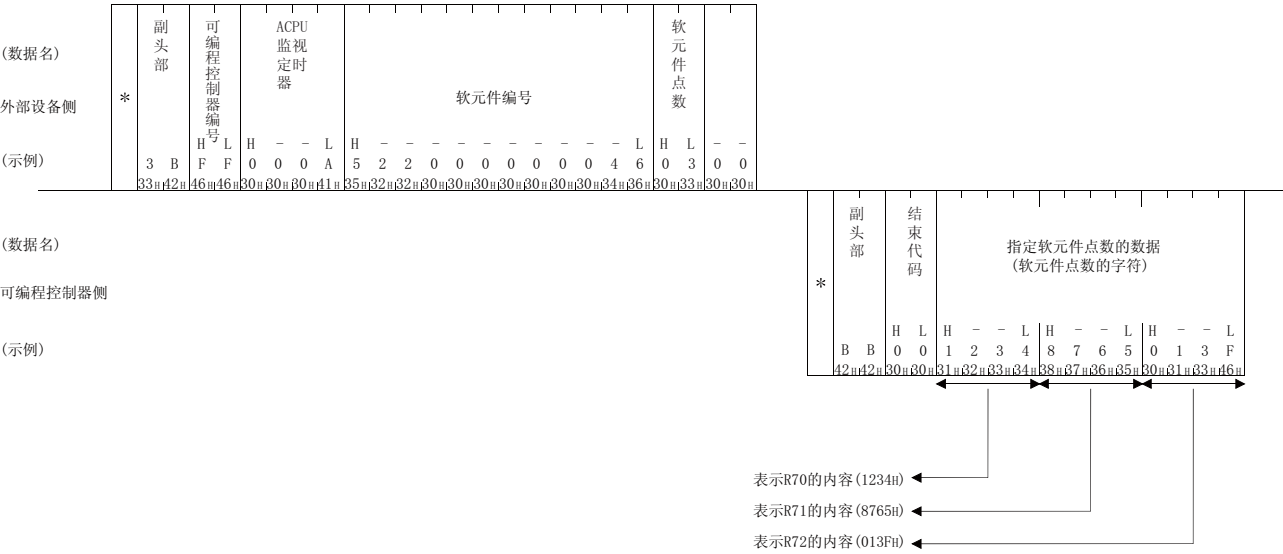
(1) 通过二进制代码进行通信时



备注

将软元件点数为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将软元件点数置为 256 点的情况下，以“00”进行指定。

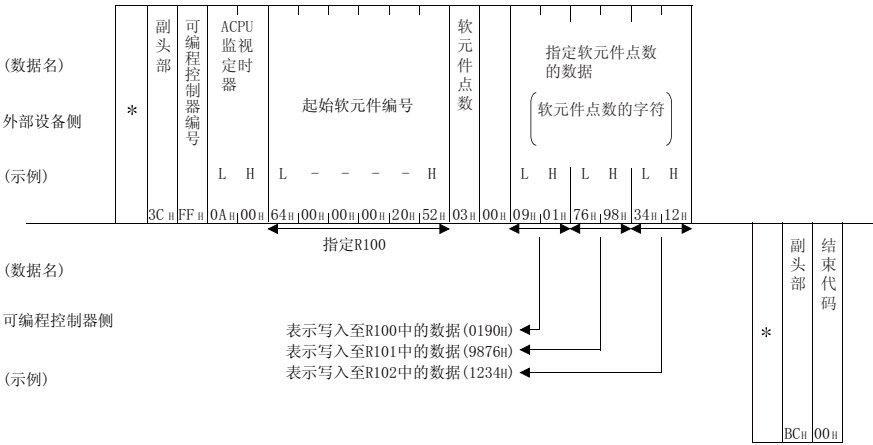
6.4.7 扩展文件寄存器的直接写入(指令：3C)

以下对进行扩展文件寄存器的直接写入时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的可编程控制器 CPU 的扩展文件寄存器 R100~102 进行数据写入的情况下。

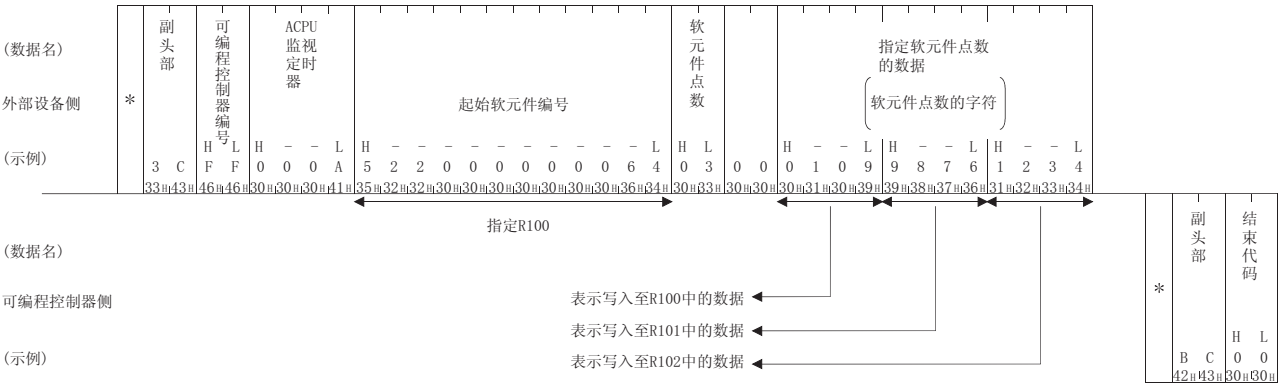
(1) 通过二进制代码进行通信时



备注

将软件点数为 256 点的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将软件点数为 256 点的情况下，以“00”进行指定。

6.4.8 扩展文件寄存器的测试(随机写入)(指令: 19)

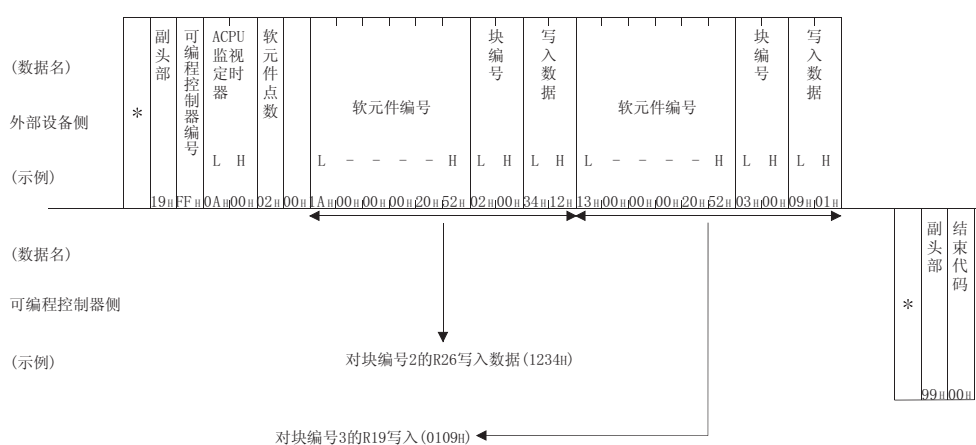
以下对随机指定扩展文件寄存器进行数据写入时的指令/响应格式有关内容进行举例说明。

对于控制步骤图中所示的*符号部分的数据项目的排列・内容, 请参阅 6.1 节中所示的详细说明。

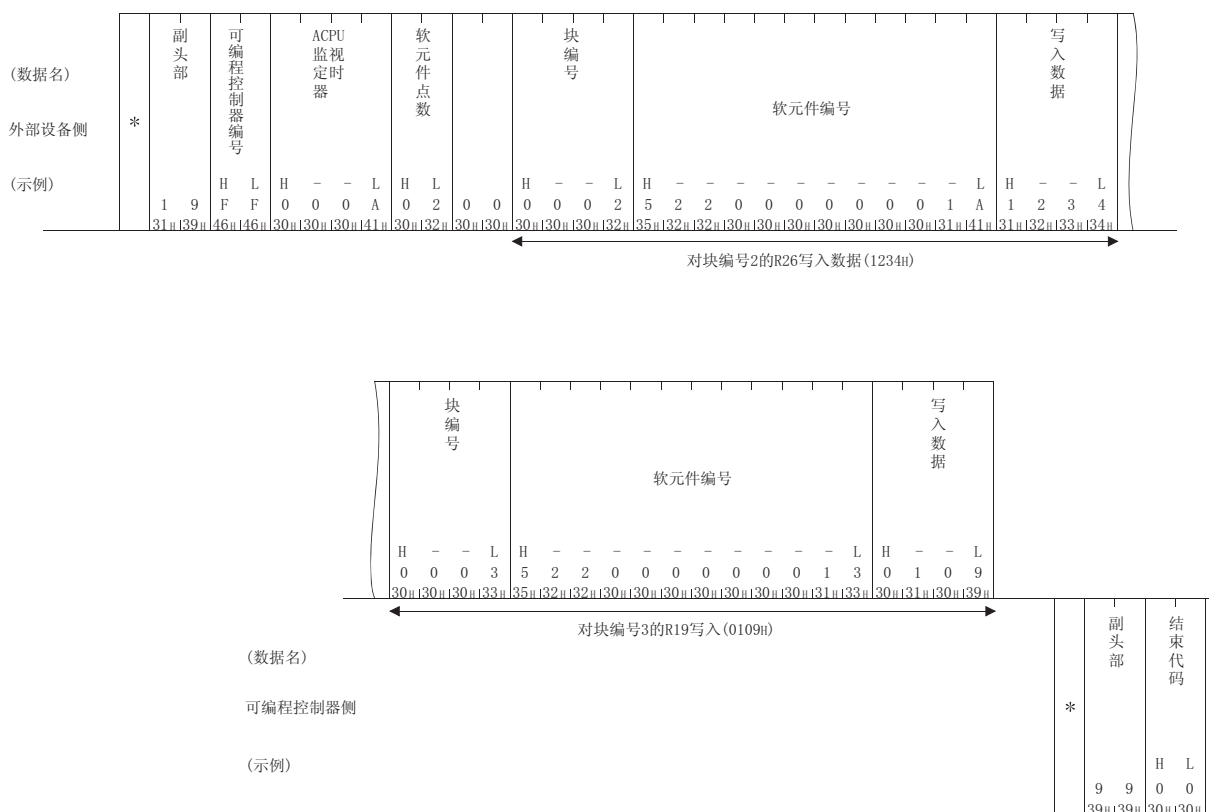
[控制步骤]

对安装了 E71 的可编程控制器 CPU 的块编号 2 的 R26 及块编号 3 的 R19 进行数据写入的情况下。

(1) 通过二进制代码进行通信时



(2) 通过 ASCII 代码进行通信时



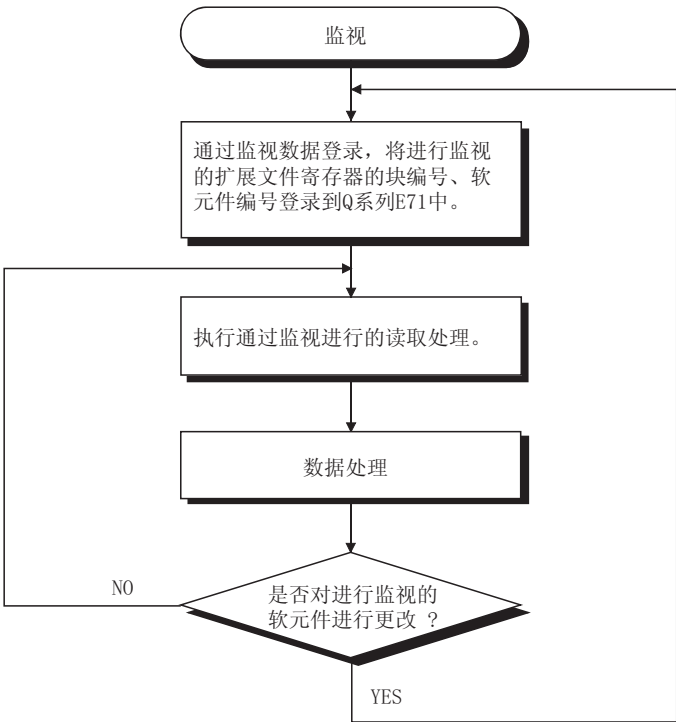
6.4.9 扩展文件寄存器的监视

将希望在外部设备中进行监视的扩展文件寄存器的块编号以及软元件编号预先登录到 E71 中，通过从外部设备执行监视指令，可以将可编程控制器 CPU 内的扩展文件寄存器的内容在外部设备中进行监视。

在通过扩展文件寄存器的批量读取进行的读取中，以连续的软元件编号进行处理，而在通过监视进行的读取中，可以随机指定任意的块编号、软元件编号的文件寄存器进行读取。

(1) 监视的操作步骤

进行监视时的操作步骤如下所示。



要点	
(1)	按上述操作步骤执行监视时必须进行监视数据登录操作。 如果在未进行监视数据登录的状况下执行监视，将变为出错(结束代码 57H)状态。
(2)	在进行了电源 OFF 或可编程控制器 CPU 的复位的情况下，监视数据登录的内容将被删除。
(3)	进行监视数据登录时，可以将扩展文件寄存器、软元件存储器・位单位以及字单位的 3 种类型的指令的中最后登录的 1 个指令预先登录到 E71 中。 关于软元件存储器的监视，请参阅 6.3.8 项。
(4)	从多个外部设备对同一站的可编程控制器 CPU 的软元件存储器进行监视数据登录时，登录数据将被覆盖登录，因此最后登录的软元件存储器将有效。

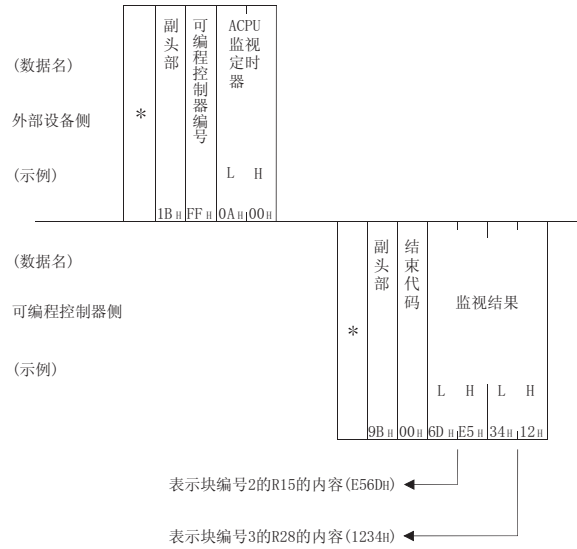
(3) 监视(指令：1B)

以下对通过监视数据登录进行了登录的扩展文件寄存器进行监视时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

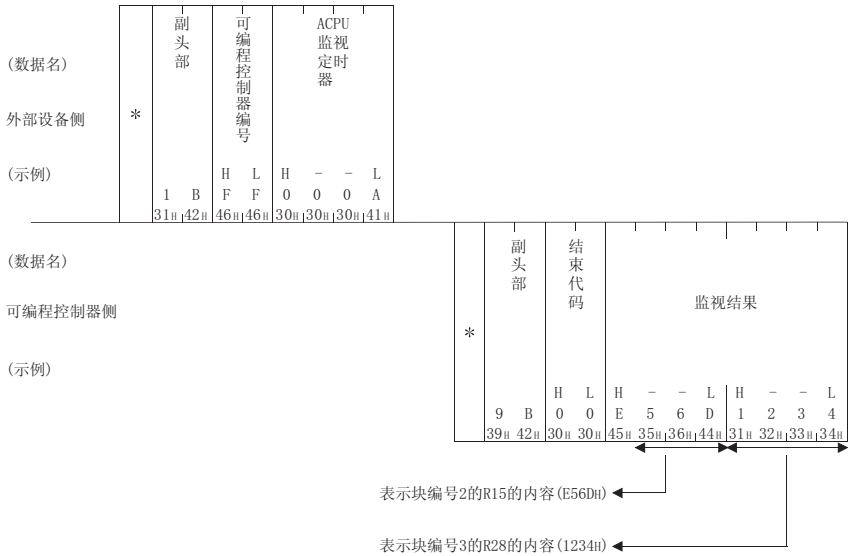
[控制步骤]

对安装了 E71 的可编程控制器 CPU 中进行了监视数据登录的块编号 2 块的 R15、块编号 3 的 R28 进行监视的情况下。

(a) 通过二进制代码进行通信时



(b) 通过 ASCII 代码进行通信时



6.5 智能功能模块的缓冲存储器的读取、写入

以下介绍对其它站的 A 系列可编程控制器 CPU 站或 MELSECNET/10 远程 I/O 站的智能功能模块(特殊功能模块)的缓冲存储器进行数据读取、写入时的控制步骤的指定内容・方法以及指定示例有关内容。
在本指令中对智能功能模块的缓冲存储器以字节单位进行访问。

6.5.1 指令及处理内容

(1) 指令

项目	指令/响应类型	处理内容	1 次通信中可处理的点数	可编程控制器 CPU 的状态		
				STOP 中	RUN 中	
					写入允许设置	写入禁止设置
批量读取	0Er	对智能功能模块的缓冲存储器内容进行读取。	256 字节 (128 字)	○	○	○
批量写入	0Fr	对智能功能模块的缓冲存储器进行数据写入。		○	○	×

上表的可编程控制器 CPU 的状态栏中的○符号表示可以执行，×符号表示不能执行。

(2) 关于可访问的模块及缓冲存储器的地址

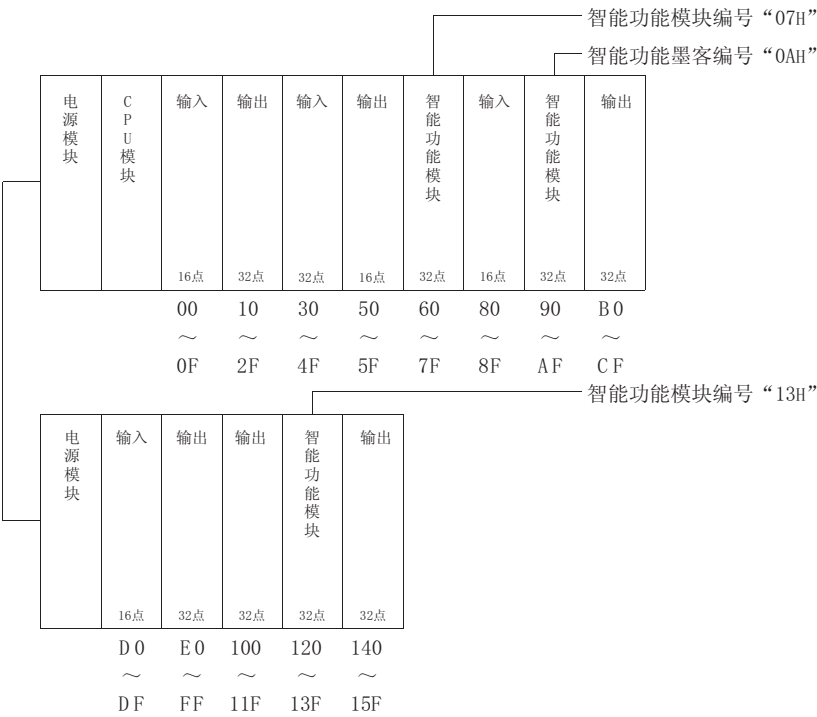
关于通过本功能可访问的 A/QnA 系列的模块型号及指定的缓冲存储器起始地址，与以 QnA 兼容 3E/3C/4C 帧、4E 帧进行通信时对智能功能模块进行访问时相同。请参阅 3.5.1 项及 3.5.2 项。(不能对 Q 系列的智能功能模块进行访问。)
在本功能中，以字节单位进行读取、写入。

要点
对于智能功能模块的缓冲存储器，各模块有可进行读取/写入的区域、只读区域、写入专用区域、用于操作系统的用户使用禁止区域。 应按照各模块的手册的说明执行本功能。 如果进行了错误的读取/写入，可编程控制器 CPU、各智能功能模块中有可能发生出错。

6.5.2 控制步骤中智能功能模块编号的思路

(1) 占用 1 个插槽的智能功能模块的模块编号

对于控制步骤中指定的智能功能模块编号，为将智能功能模块的输入输出信号的最终编号以 3 位表示时的高 2 位。(I/O 地址)的最终编号以 3 位表示时的高 2 位。



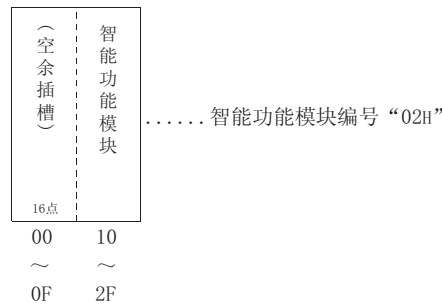
(2) 占用 2 个插槽的智能模块的模块编号

对于占用 2 个插槽的智能功能模块，每个模块中各插槽的占用点数是确定的。

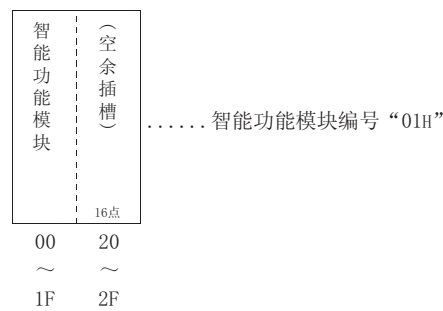
对于智能功能模块编号，为将作为智能功能模块分配的插槽的输入输出信号的最终编号以 3 位表示时的高 2 位。

关于各模块的各插槽的分配，请参阅相应的智能功能模块用户手册。

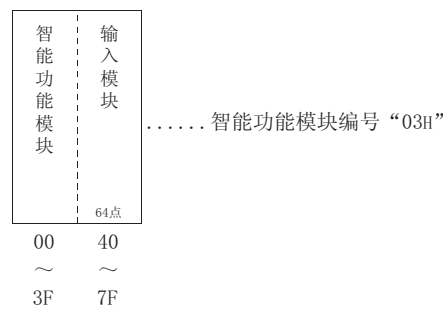
- 1) 将前半部分插槽分配为空余插槽的模块的情况下。
(AD72、A84AD 等)



- 2) 将后半部分插槽分配为空余插槽的模块的情况下。
(A61LS 等)

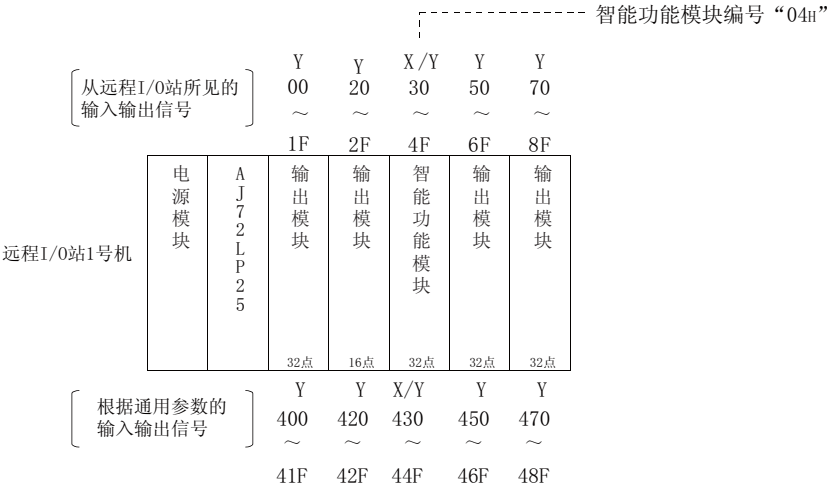


- 3) 智能功能模块分配与输入输出分配混合的模块的情况下。
(A81CPU 的情况下)



(3) MELSECNET/10 远程 I/O 站的智能功能模块的模块编号

对于远程 I/O 站的智能功能模块的模块编号，为将所以下述“从远程 I/O 站所见的输入输出信号”的最终编号以 3 位表示时的高 2 位。
与 MELSECNET/10 远程 I/O 网络的主站中设置的通用参数的内容无关，应以“从远程 I/O 站所见的输入输出信号”进行指定。



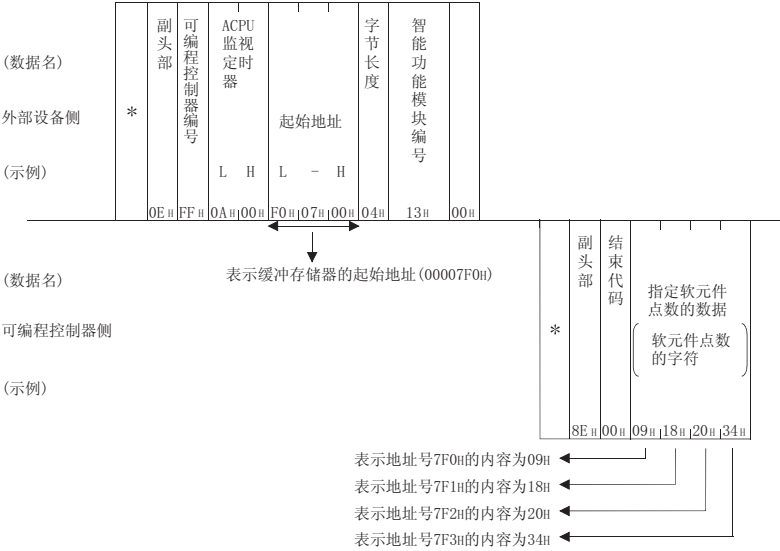
6.5.3 智能功能模块的缓冲存储器读取(指令：0E)

以下对从智能功能模块的缓冲存储器中进行数据读取时的指令/响应格式有关内容进行举例说明。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的同一站内的智能功能模块(输入输出信号为 120H~13FH(模块编号 13H))
的缓冲存储器地址 7F0H~7F3H 的内容进行读取的情况下。

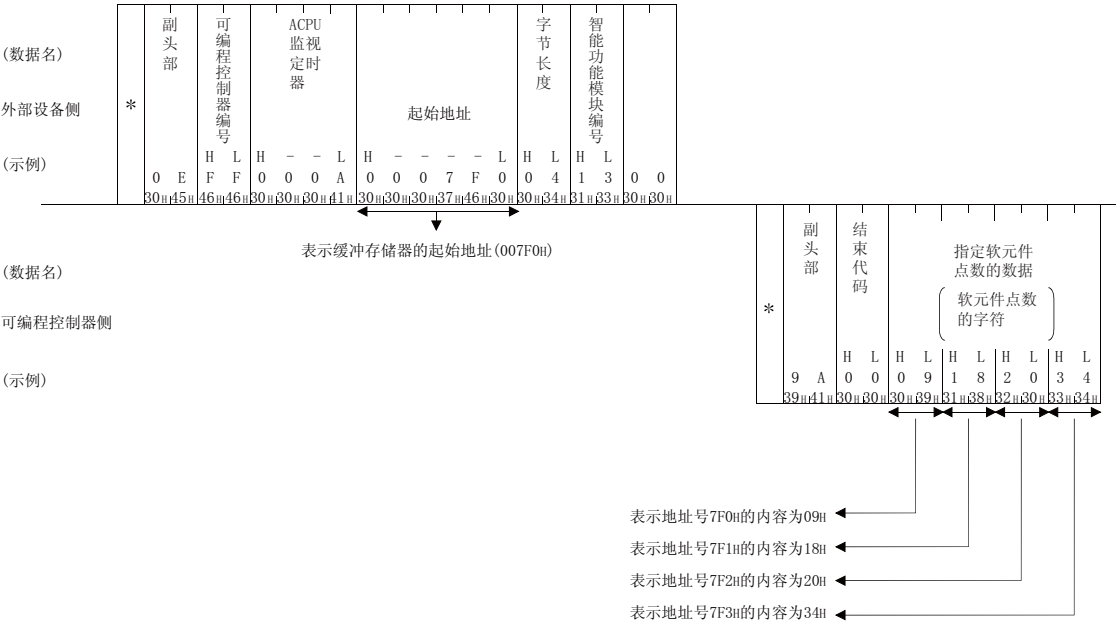
(1) 通过二进制代码进行通信时



备注

将字节长度设置为 256 字节的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将字节长度设置为 256 字节的情况下，以“00”进行指定。

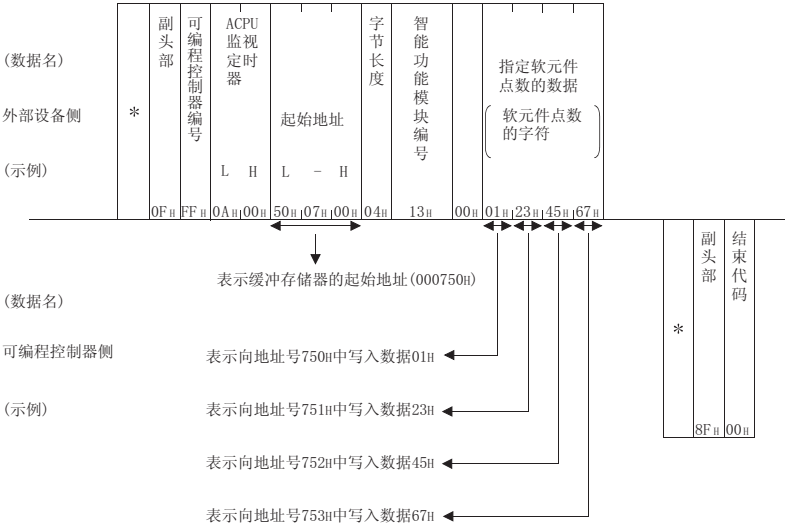
6.5.4 智能功能模块的缓冲存储器写入(指令：0F)

以下举例说明对智能功能模块的缓冲存储器进行数据写入时的指令/响应格式有关内容。
对于控制步骤图中所示的*符号部分的数据项目的排列・内容，请参阅 6.1 节中所示的详细说明。

[控制步骤]

对安装了 E71 的同一站内的智能功能模块(输入输出信号为 120H~13FH(模块编号 13H))
的缓冲存储器地址 750H~753H 进行数据写入的情况下。

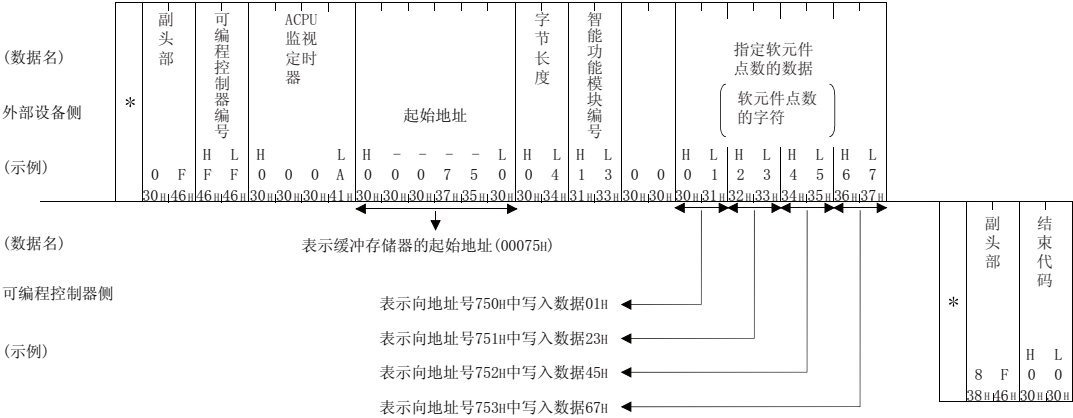
(1) 通过二进制代码进行通信时



备注

将字节长度设置为 256 字节的情况下，以“00H”进行指定。

(2) 通过 ASCII 代码进行通信时



备注

将字节长度设置为 256 字节的情况下，以“00”进行指定。

附录

附录 1 关于通过软元件存储器的扩展指定进行的读取、写入

软元件存储器的扩展指定是指，外部设备通过 3.3.1 项(1)中所示的指令对 3.3.1 项(3)中所示的软元件以外进行指定，
对访问的软元件存储器的软元件编号及网络编号等进行修饰指定。
以下介绍软元件存储器的扩展指定概要。
在本节以后的说明中将这些指定表示为[指定-1] ~ [指定-5]。

[指定-1]

是用于对 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 的链接直接软元件进行访问的指定。(关于链接输入、链接输出、链接特殊继电器等，参阅附录 1.2(2)的表)

[指定-2]

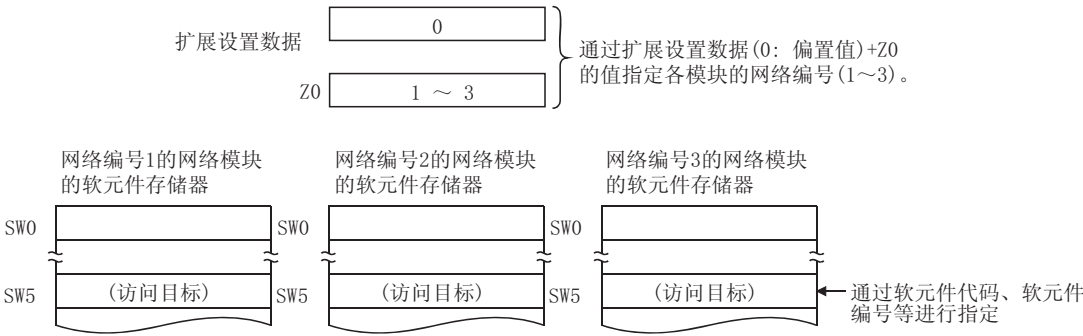
是用于对 Q/L/QnACPU 站、MELSECNET/H、MELSECNET/10 远程 I/O 站的智能功能模块(包括 A/QnA 系列的特殊功能模块)的智能功能模块软元件(包括特殊功能模块软元件)进行访问的指定。(关于缓冲寄存器，参阅附录 1.2(2)的表)

[指定-3]

是根据以下 1)、2)的网络编号、输入输出信号的扩展设置修饰进行的对象模块的指定。

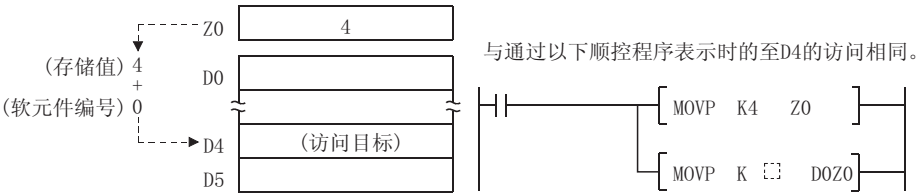
- 1) 上述[指定-1]的对象模块的指定。
(任意的网络编号)+(变址寄存器中指定的网络编号)的网络模块
- 2) 上述[指定-2]的对象模块的指定。
(任意的输入输出信号编号)+(变址寄存器中指定的输入输出信号编号)的智能功能模块

例如访问站中安装了多个网络模块时，可以由外部设备指定如下所示的说明图中的扩展设置数据及变址寄存器“Z0”，对各模块的相同软元件存储器(SW5)进行访问。

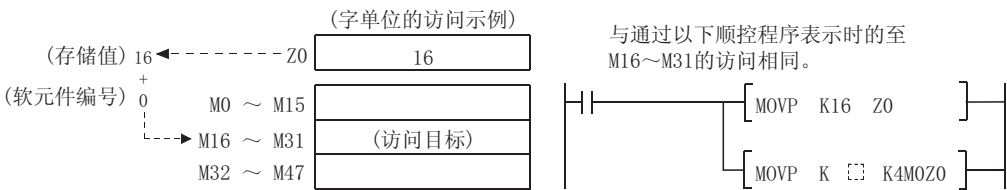


[指定-4]

通过软元件编号及变址寄存器中的软元件修饰对上述[指定-1]、[指定 - 2]及 3.3.1 项(3)的软元件进行的对象软元件编号的指定。
例如通过 D0 及 Z0 的指定，可以对软元件存储器(D4)进行访问。

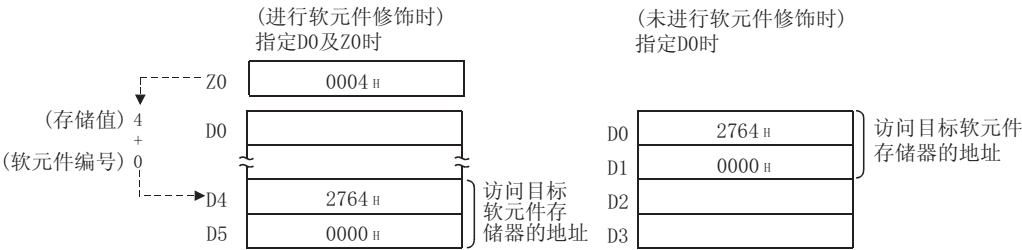


此外，通过 M0 及 Z0 的指定，可以对软元件存储器(M16 或 M16 ~ M31 等)进行访问。



[指定-5]

是指定的字软元件的存储值作为访问目标软元件存储器的地址的字软元件的间接指定。
在以下情况下，对软元件存储器地址 00002764h 存储器中分配的软元件存储器进行访问。



要点
外部设备通过间接指定对软元件存储器进行访问时，在访问之前应通过与可编程控制器 CPU 侧的约定执行以下操作。
(1) 通过可编程控制器 CPU 的 ADRSET 指令对访问的软元件存储器分配的存储器上的地址进行确认。
(2) 将上述 1)中确认的地址通过间接指定存储到指定的软元件存储器中。
(示例)
外部设备访问 D100 时，将 D100 的地址存储到 D0、D1 中的顺控程序示例 (外部设备通过对 D0 进行间接指定，可以对 D100 进行访问。)

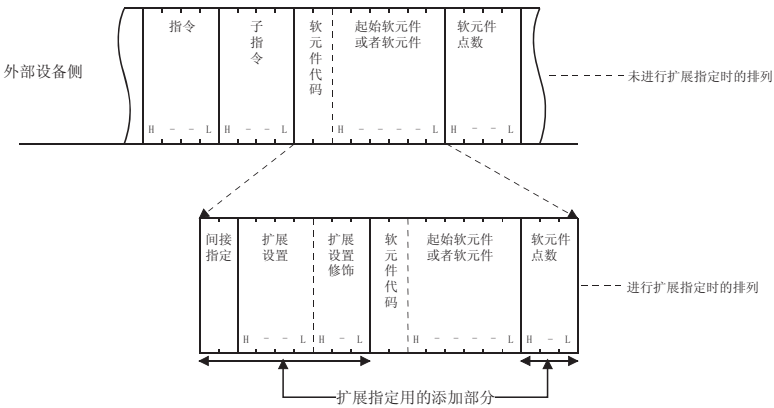
附录 1.1 软元件存储器扩展指定时的字符部分的数据的排列及内容

以下介绍对进行数据的读取、写入的软元件存储器进行扩展指定时的扩展指定用的添加部分的指定内容等。
关于与不对软元件存储器进行扩展指定时相同的数据项目，请参阅 3.3.1 项(2)(3)。

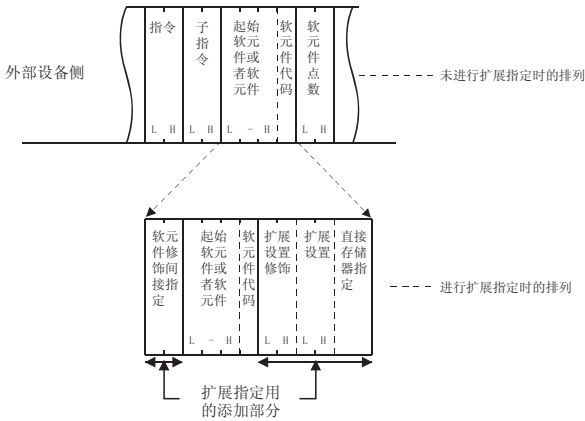
要点
(1) 扩展指定时，将不进行扩展指定时的软元件存储器指定部分替换为扩展指定用的指定。
(2) 指定多个软元件存储器时应全部进行扩展指定。

(a) 对软元件存储器进行扩展指定时的软元件存储器指定部分的数据的排列

1) 通过 ASCII 代码进行数据通信时的 QnA 兼容 3E/3C/4C 帧、4E 帧的数据的排列



2) 通过二进制代码进行数据通信时的 QnA 兼容 3E/3C/4C 帧、4E 帧的数据的排列



- (b) 对软元件存储器进行扩展指定时的字符部分的内容
对软元件存储器进行扩展指定时至字符部分的各指定值如下所示。
(通过 ASCII 代码进行数据通信时)

		通过外部设备进行的指定值					指定字符数
		[指定-1]	[指定-2]	[指定-3]	[指定-4]	[指定-5]	
子指令	无监视条件	“ 0080 ” / “ 0081 ” (参阅 1))					4
	有监视条件	“ 00C0 ” / “ 00C1 ” (参阅 1))					
间接指定		“ 00 ”				“ 0@ ”	2
扩展设置		“ J ☐ ” (参阅 2))	“ U ☐ ” (参阅 2))	左述之一 / “ 0000 ” (参阅 2))		“ 0000 ”	4
扩展设置修饰		“ 000 ”		“ Z ☐ ” (参阅 3))	左述之一 (参阅 3))	“ 000 ”	3
软元件代码		(参阅附录 1.2(2)(a)以及 3.3.1 项(3))					2
起始软元件或软元件							6
软元件修饰		“ 000 ”			“ Z ☐ ” (参阅 4))	左述之一 (参阅 4))	3

(通过二进制代码进行数据通信时)

		通过外部设备进行的指定值					指定字符数
		[指定-1]	[指定-2]	[指定-3]	[指定-4]	[指定-5]	
子指令	无监视条件	" 0080 _H "/"0081 _H " (参阅 1))					2
	有监视条件	" 00C0 _H "/"00C1 _H " (参阅 1))					
软元件修饰间接指定		0000 _H			40 ☐ _H (参阅 4))	0800 _H / ☐ 8 ☐ _H (参阅 4))	2
起始软元件或软元件		(参阅附录 1.2(2)(a)以及 3.3.1 项(3))					3
软元件代码							1
扩展设置修饰		0000 _H		40 ☐ _H (参阅 3))	左述之一 (参阅 3))	0000 _H	2
扩展设置		☐ _H (参阅 2))				0000 _H	2
直接存储器指定		F9 _H	F8 _H	左述之一 / 00 _H (参阅 2))		00 _H	1

1) 子指令

是用于对读取/写入的单位、指定软元件的类型、数据读取条件等进行指定的数据。

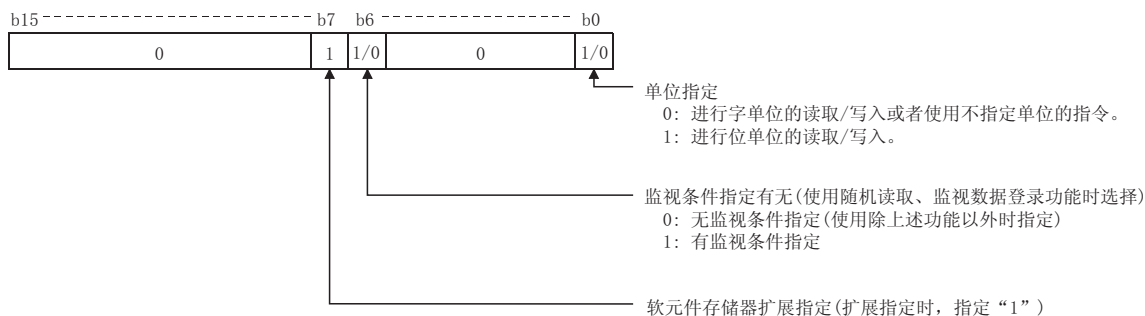
a) 通过 ASCII 代码进行数据通信时

将下述数值转换为 ASCII 代码 4 位(16 进制数)后使用,从高位开始进行发送。

b) 通过二进制代码进行数据通信时

使用下述 2 字节的数值进行发送。

c) 子指令的指定内容如下所示。



- d) 关于监视条件指定有无(随机读取、监视数据登录用)以及有监视条件指定时的软元件存储器的读取时机等有关内容,请参阅 3.3.8 项。
- e) 进行软元件存储器扩展指定时,根据监视条件指定及单位指定,子指令将变为 0080_H、0081_H、00C0_H、00C1_H。
- 2) 扩展设置、直接存储器指定
是用于外部设备对如下所示的软元件存储器进行访问的软元件存储器指定用的数据。
- 链接直接软元件 : [指定-1]
 - 智能功能模块的直接软元件 : [指定-2]
- a) 通过 ASCII 代码进行数据通信时
仅指定扩展设置数据,使用以下值从高位开始进行发送。

指定值	对象软元件存储器	备注
"0000"	(无扩展设置)	—
"J □"	链接直接软元件	对于在 □ 部分,将访问对象网络编号转换为 ASCII 代码 3 位(16 进制数)后进行指定。
"U □"	智能功能模块的智能功能模块软元件	对于 □ 部分,以将访问对象智能功能模块的起始输入输出信号转换为 ASCII 代码 4 位(16 进制数)时的高 3 位进行指定。

- d) 通过二进制代码进行数据通信时
对扩展设置、直接存储器指定数据进行指定后,使用以下值进行发送。(对于扩展设置数据,从 Low 字节(L: 位 0~7)开始进行发送)

指定值		对象软元件存储器	备注
扩展设置	直接存储器指定		
0000 _H	00 _H	(无扩展设置)	—
□ _H	F9 _H	链接直接软元件	对于 □ 部分,以访问对象网络编号(16 进制数)进行指定。
□ _H	F8 _H	智能功能模块的智能功能模块软元件	对于 □ 部分,以将访问对象智能功能模块的起始输入输出信号(16 进制数)以 4 位表示时的高 3 位进行指定。

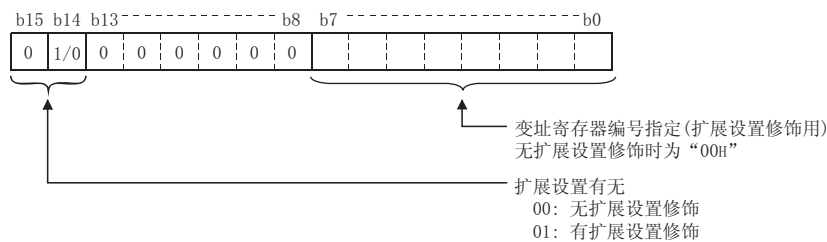
3) 扩展设置修饰([指定 - 3]用)

是用于将扩展设置中的指定值作为网络编号/输入输出信号的偏置值，然后使用变址寄存器指定任意网络编号/输入输出信号的模块的数据。

- a) 通过 ASCII 代码进行数据通信时
使用以下值从高位开始进行发送。

指定值	内容	备注
" 000 "	(无扩展设置修饰)	—
" Z "	扩展设置修饰用变址寄存器	对于 部分，将所使用的变址寄存器编号转换为 ASCII 代码 2 位(10 进制数)后进行指定。

- b) 通过二进制代码进行数据通信时，使用以下数值，从 Low 字节(L: 位 0 ~ 7)开始进行发送。



- c) 可以将变址寄存器(Z0 ~ Z15)作为扩展设置修饰使用。
d) 将输入输出信号存储到扩展设置修饰用变址寄存器中时，以下述减法运算值进行指定。

(将访问对象模块的起始输入输出信号以 4 位表示时的高 3 位的值) - (扩展设置的指定值)
(至扩展设置修饰用变址寄存器的存储值)

4) 软元件修饰、间接指定

(软元件修饰: [指定 - 4]用)

是用于将起始软元件(或软元件)中的指定值作为软元件编号的偏置值，然后使用变址寄存器对同一软元件的任意软元件编号进行指定的数据。

(间接指定: [指定 - 5]用)

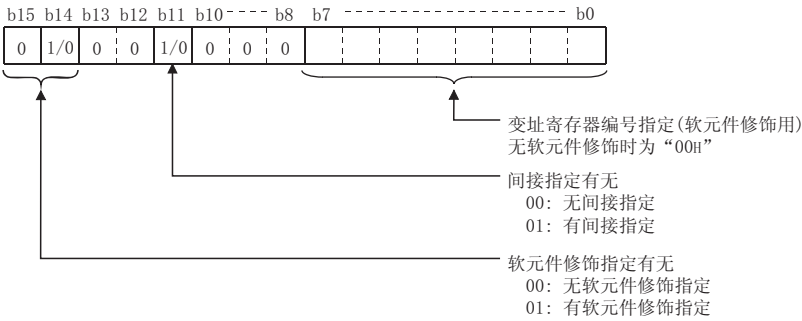
是存储在起始软元件(或软元件)中指定的软元件存储器及以下软元件存储器中的值，是外部设备进行访问的软元件存储器的地址数据。

对字软元件进行访问时，可以进行间接指定。

- a) 通过 ASCII 代码进行数据通信时
仅指定软元件修饰数据，使用以下值从高位开始进行发送。

指定值	内容	备注
" 000 "	(无软元件修饰)	—
" Z "	软元件修饰用变址寄存器	对于 部分，将所使用的变址寄存器的编号转换为 ASCII 代码 2 位(10 进制数)后进行指定。

b) 以二进制模式进行数据通信时
对软元件修饰、间接指定数据进行指定后，使用以下值从 Low 字节(L: 位 0 ~ 7)开始进行发送。



c) 可以将变址寄存器(Z0 ~ Z15)作为扩展设置修饰使用。

附录 1.2 可进行扩展指定的软元件存储器及指定示例

(a) 可进行扩展指定的软元件存储器

可以对 C24/E71 安装站以及经由 CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 的 Q/QnACPU 站的如下所示的软元件存储器进行扩展指定。

- 1) Q/L/QnACPU 的软元件存储器(参阅 3.3.1 项(3))
- 2) 网络模块的链接直接软元件、智能功能模块的智能功能模块软元件(参阅下表)

分类	软元件	类型	扩展设置		软元件代码*1		软元件编号范围*1		备注		
			ASCII 代码时	二进制 代码时	ASCII 代码时	二进制 代码时					
链接直接软元件	链接输入	位	J 	F9 _H	X *	9C _H	000000 ~ 001FFF	16 进制	· 在扩展设置的 中指定网络编号。 · 括弧内是指通用型 QCPU 的情况下。		
	链接输出				Y *	9D _H	000000 ~ 001FFF	16 进制			
	链接继电器				B *	A0 _H	000000 ~ 001FFF (000000 ~ 007FFF)	16 进制			
	链接特殊继电器				SB	A1 _H	000000 ~ 0001FF	16 进制			
	链接寄存器	字			W *	B4 _H	000000 ~ 001FFF (000000 ~ 01FFFF)	16 进制			
	链接特殊寄存器				S *	B5 _H	000000 ~ 0001FF	16 进制			
智能功能模块软元件	缓冲寄存器 (缓冲存储器)	字	U 	F8 _H	G *	AB _H	000000 ~ 016383	10 进制	· 在扩展设置的 中指定对象模块的输入输出信号。		

*1 对于软元件代码指定以及起始软元件(或软元件)指定的“00...0”，可以如 3.3.1 项(2)(c) 2) 3) 中所示以空格(代码：20_H)进行指定。

(b) 软元件存储器的扩展指定示例

对软元件存储器进行扩展指定时的指定示例如下所示。(子指令为无监视条件指定的情况下)

[指定-1]的情况下

a) 对以下软元件存储器进行访问的情况下

- 对象模块 : 网络编号为(008_H)的网络模块
- 软元件编号 : 以位单位进行访问

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 1	0 0	J 0 0 8	0 0 0	X *	0 0 0 1 0 0	0 0 0
	30 _H 30 _H 38 _H 31 _H	30 _H 30 _H	4A _H 30 _H 30 _H 38 _H	30 _H 30 _H 30 _H	F8 _H 2A _H	30 _H 30 _H 30 _H 31 _H 30 _H 30 _H	30 _H 30 _H 30 _H

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H	(X)	L H	L H	
(示例)	31 _H 00 _H	30 _H 00 _H	30 _H 01 _H 00 _H	9C _H	00 _H 00 _H	38 _H 00 _H	F9 _H

b) 对以下软元件存储器进行访问的情况下

- 对象模块 : 网络编号为 8(008H)的网络模块
- 软元件编号 : W100: 以字单位进行访问

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L H - - L	H - L	H L H - - L	H L H - - L	H - L	H - L
(示例)	0 0 8 0	0 0 J 0 0 8	0 0 0	W *	0 0 0 1 0 0	0 0 0	0 0 0
	30h 30h 38h 30h	30h 30h 4Ah 30h 30h 38h	30h 30h 30h	57h 2Ah	30h 30h 30h 31h 30h 30h	30h 30h 30h	

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H L H	L - H	(W)	L H L H	L H L H		
(示例)	30h 00h	00h 00h	00h 01h 00h	B4h	00h 00h	08h 00h	F9h

[指定-2]的情况下

a) 对以下缓冲存储器进行访问的情况下

- 对象模块 : 起始输入输出信号为 010H 的智能功能模块
- 起始地址 : 3072(C00H)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L H - - L	H - L	H L H - - L	H L H - - L	H - L	H - L
(示例)	0 0 8 0	0 0 U 0 0 1	0 0 0	G *	0 0 3 0 7 2	0 0 0	0 0 0
	30h 30h 38h 30h	30h 30h 55h 30h 30h 31h	30h 30h 30h	47h 2Ah	30h 30h 33h 30h 37h 32h	30h 30h 30h	

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H L H	L - H		L H L H	L H L H		
(示例)	30h 00h	30h 00h	00h 0C 00h	ABh	00h 00h	01h 00h	F8h

[指定-3]的情况下

a) 对以下软元件存储器进行访问的情况下

- 对象模块 : 网络编号为 12(0CH) + Z0 的网络模块
- 起始地址 : W100

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	J 0 0 C	Z 0 0	# *	0 0 0 1 0 0	0 0 0
	30h 30h 38h 30h	30h 30h	4Ah 30h 30h 43h	5Ah 30h 30h	57h 2Ah	30h 30h 31h 30h 30h	30h 30h 30h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H		L H	L H	
(示例)			(0)				
	80h 00h	00h 00h	30h 01h 00h	B4h	00h 40h	C h 00h	F9h

b) 对以下缓冲存储器进行访问的情况下

- 对象模块 : 起始输入输出信号为 010H + Z1 的智能功能模块
- 起始地址 : 3072(C00H)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	U 0 0 1	Z 0 1	G *	0 0 3 0 7 2	0 0 0
	30h 30h 38h 30h	30h 30h	55h 30h 30h 31h	5Ah 30h 31h	47h 2Ah	30h 30h 33h 30h 27h 32h	40h 30h 30h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H		L H	L H	
(示例)							
	80h 00h	30h 00h	00h 0C h 00h	A8h	01h 40h	01h 00h	F8h

[指定-4]的情况下

a) 对以下软元件存储器进行访问的情况下

· 软元件编号 : 以 M200 + Z3 指定 : 以位单位进行访问的内部继电器(M)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 1	0 0	0 0 0 0	0 0 0	M *	0 0 0 2 0 0	Z 0 3
	30h 30h 38h 31h	30h 30h	30h 30h 30h 30h	30h 30h 30h	40h 2Ah	30h 30h 30h 32h 30h 30h	5Ah 30h 33h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H	(0)	L H	L H	
(示例)	81h 00h	03h 40h	28h 00h 00h	90h	30h 00h	00h 00h	00h

b) 对以下软元件存储器进行访问的情况下

· 软元件编号 : 以 D100 + Z4 指定 : 以字单位进行访问的数据寄存器(D)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	0 0 0 0	0 0 0	D *	0 0 0 1 0 0	Z 0 4
	30h 30h 38h 30h	30h 30h	30h 30h 30h 30h	30h 30h 30h	44h 2Ah	30h 30h 30h 31h 30h 30h	5Ah 30h 34h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H	(D)	L H	L H	
(示例)	80h 00h	04h 40h	54h 00h 00h	A8h	30h 00h	08h 00h	00h

c) 对以下软元件存储器进行访问的情况下

- 对象模块 : 网络编号为 8(008H)的网络模块
- 软元件编号 : 以 X100 + Z5 表示 : 以位单位进行访问的链接输入(X)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 1	0 0	J 0 0 8	0 0 0	X *	0 0 0 1 0 0	Z 0 5
	30h 30h 38h 31h	30h 30h	4Ah 30h 30h 38h	30h 30h 30h	58h 2Ah	30h 30h 30h 31h 30h 30h	5Ah 30h 35h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H L H	L - H			L H	L H	
(示例)			(X)				
	81h 00h	05h 40h	00h 01h 00h	9C h	00h 00h	08h 00h	F9h

d) 对以下软元件存储器进行访问的情况下

- 对象模块 : 网络编号为 8(008H) + Z11 的网络模块
- 软元件编号 : 以 W10 + Z6 表示的 : 以字单位进行访问的链接寄存器(W)

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	J 0 0 8	Z 1 1	W *	0 0 0 0 1 0	Z 0 6
	30h 30h 38h 30h	30h 30h	4Ah 30h 30h 38h	5Ah 31h 31h	57h 2Ah	30h 30h 30h 31h 30h	5Ah 30h 36h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件修饰间接指定	起始软元件	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H L H	L - H			L H	L H	
(示例)			(W)				
	80h 00h	06h 40h	00h 01h 00h 00h	B4 h	0B h 40h	08h 00h	F9h

e) 对以下缓冲存储器进行访问的情况下

- 对象模块 : 起始输入输出信号为 010_H 的智能功能模块
- 起始地址 : 以 3072(C00_H) + Z7 表示的地址

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	U 0 0 1	0 0 0	G *	! 0 0 3 0 7 2	Z 0 7
	30H 30H 38H 30H	30H 30H	55H 30H 30H 31H	30H 30H 30H	47H 2AH	30H 30H 33H 30H 37H 32H	5AH 30H 37H

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H		L H	L H	
(示例)	80H 00H	37H 40H	00H 0C H 00H	A8 H	30H 00H	01H 00H	F8 H

f) 对以下缓冲存储器进行访问的情况下

- 对象模块 : 起始输入输出信号为 010_H + Z12 的智能功能模块
- 起始地址 : 以 3072(C00_H) + Z7 表示的地址

(通过ASCII代码进行通信时的指定方法)

(数据名)	子指令	间接指定	扩展设置	扩展设置修饰	软元件代码	起始软元件(软元件)	软元件修饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 0	U 0 0 1	Z 1 2	G *	! 0 0 3 0 7 2	Z 0 7
	30H 30H 38H 30H	30H 30H	55H 30H 30H 31H	5AH 31H 32H	47H 2AH	30H 30H 33H 30H 37H 32H	5AH 30H 37H

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子指令	软元件间接指定	起始软元件(软元件)	软元件代码	扩展设置修饰	扩展设置	直接存储器设置
可编程控制器侧	L H	L H	L - H		L H	L H	
(示例)	80H 00H	37H 40H	00H 00H 00H	A8 H	3C H 40H	01H 00H	F8 H

[指定-5]的情况下

对以下软元件存储器进行访问的情况下

- 软元件编号：以 D100 + Z4 表示的数据寄存器：以字单位进行访问
(D)中存储的地址的软元件存储器

(通过ASCII代码进行通信时的指定方法)

(数据名)	子 指 令	间 接 指 定	扩 展 设 置	扩 展 设 置 修 饰	软 元 件 代 码	起始软元件 (软元件)	软 元 件 修 饰
外部设备侧	H - - L	H L	H - - L	H - L	H L	H - - - L	H - L
(示例)	0 0 8 0	0 #	0 0 0 0	0 0 0 0	D *	0 0 0 1 0 0	Z 0 4
	30h 30h 38h 30h	30h 40h	30h 30h 30h 30h	30h 30h 30h	44h 2Ah	30h 30h 30h 31h 30h 30h	5Ah 30h 34h

(通过二进制代码进行数据通信时的指定方法)

(数据名)	子 指 令	软 元 件 修 饰 间 接 指 定	起 始 软 元 件 (软 元 件)	软 元 件 代 码	扩 展 设 置 修 饰	扩 展 设 置	直 接 存 储 器 设 置
可编程控制器侧	L H	L H	L - H		L H	L H	
(示例)	80h 00h	04h 48h	64h 00h 00h	ASh	00h 00h	00h 00h	00h
			(D)				

附录 1.3 软元件存储器的扩展指定时的限制

以下介绍对软元件存储器进行扩展指定时的限制。

(a) 可进行软元件存储器的扩展指定的指令

将可进行软元件存储器的扩展指定的内容按 QnA 兼容 3E/3C/4C 帧、4E 帧用的软元件存储器的读取、写入指令分类表示如下。

功能		访问软元件	指令	软元件存储器扩展指定项目					指定表示				
				间接指定	扩展设置	扩展设置修饰	软元件修饰	直接存储器指定	指定-1	指定-2	指定-3	指定-4	指定-5
多个块批量读取 多个块批量写入	字单位	位	0406	×	○	×	×	○	○	○	×	×	×
		字	1406	×	○	×	×	○	○	○	×	×	×
批量读取 批量写入	字单位	位	0401	×	○	×	×	○	○	○	×	×	×
		字	1401	×	○	×	×	○	○	○	×	×	×
随机读取	字单位	位	0403	×	○	○	○	○	○	○	○	○	○
		字	0403	○	○	○	○	○	○	○	○	○	○
测试(随机写入)	字单位	位	1402	×	○	○	○	○	○	○	○	○	○
		字	1402	×	○	○	○	○	○	○	○	○	○
监视数据登录	字单位	位	0801	×	○	○	○	○	○	○	○	○	○
		字	0801	○	○	○	○	○	○	○	○	○	○
监视	字单位	—	0802	×	×	×	×	×	—	—	—	—	—

↑
位：位软元件
字：字软元件

○：可以指定 ×：不能指定

(b) 软元件存储器扩展指定有无的混合指定

使用以下功能时，可以将多个软元件名指定到指令报文中。在这种情况下，对软元件存储器进行扩展指定时，应将指令报文中指定的软元件全部进行扩展指定。

软元件存储器的“有扩展指定”与“无扩展指定”不能混合在一起。

- 随机读取功能 : (指令: 0403)
- 测试(随机写入)功能 : (指令: 1402)
- 监视数据登录功能 : (指令: 0801)
- 多个块批量读取/写入功能 : (指令: 0406, 1406)

(c) 至智能功能模块的访问(包括智能功能模块)

- 1) 对于以下站中安装的智能功能模块，可以通过软元件存储器扩展指定进行智能功能模块的缓冲存储器的读取/写入。
 - 与外部设备相连接的 C24/E71 安装站以及与该 C24 多点连接的 Q/QnACPU 站。
 - CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的 Q/QnACPU 站以及经由该 Q/L/QnACPU 站的 C24 进行的多点连接的 Q/QnA 相应站。
- 2) 关于对除上述以外的站中安装的智能功能模块的读取/写入，请按照 3.5 节的说明进行操作。

附录 2 关于缓冲存储器的读取/写入

以下介绍通过外部设备对安装了 C24/E71 的 Q/LCPU 站(本站)或未安装 C24/E71 的网络系统上的可编程控制器站(其它站)中安装的智能功能模块(包括特殊功能模块)的缓冲存储器进行数据读取、写入的功能有关内容。

可进行智能功能模块的缓冲存储器的读取/写入的可编程控制器站、使用的功能及指令如下所示。

表中所示的访问-1 ~ 访问-4 的对象站及对象模块、读取/写入方法的对象功能/指令如下所示。

	读取/写入对象			读取/写入方法			
	No.	对象站	对象模块	QnA 扩展指定 (附录 1)	QnA 兼容 3E/3C/4C 帧、4E 帧		A 兼容 1E/1C 帧 用指令“OE”、 “OF”“TR”、 “TW”
					“0601”、 “1601”	“0613”、 “1613”	
访问-1(本站)	(1)	外部设备连接站	Q/LCPU 站	×	×	○	×
			Q 系列远程 I/O 站				
访问-2(本站)	(2)	外部设备连接站	Q/LCPU 站	○	○	×	×
			Q 系列远程 I/O 站	×	○	×	×
访问-3(其它站)	(3)	CC-Link IE 控制 网络、CC-Link	Q/QnACPU 站	○	○	×	×
	(4)		Q/QnA 对应远程 I/O 站	×	○	×	×
访问-4(其它站)	(5)	IE 现场网络、 MELSECNET/H、 MELSECNET/10 上 的站	Q/QnACPU 以外的 可编程控制器 CPU 站	×	×	×	×
	(6)		AnU 对应的远程 I/O 站	×	×	×	×

○：可以读取/写入 ×：不能读取/写入

(1) 表中所示的访问-1 ~ 访问-6 的对象站及对象模块

- 访问-1(本站)
 - 1) 与外部设备相连接的 C24/E71 以及与该 C24 多点连接的 C24。
- 访问-2(本站)
 - 2) 与外部设备相连接的 C24/E71 安装站的智能功能模块以及与该 C24 多点连接的 C24 安装站的智能功能模块。
- 访问-3(其它站)
 - 3) CC-Link IE 控制网络、CC-Link IE 现场网络、MELSECNET/H、MELSECNET/10 上的 Q/QnACPU 站中安装的智能功能模块。
 - 4) MELSECNET/H、MELSECNET/10 上的 Q/QnA 对应远程 I/O 站中安装的智能功能模块。
- 访问-4(其它站)
 - 5) MELSECNET/10 上的 Q/QnACPU 以外的站中安装的智能功能模块。
 - 6) MELSECNET/10 上的 AnU 对应远程 I/O 站中安装的智能功能模块。

(2) 表中所示的读取/写入方法的对象功能/指令

- QnA 扩展指定：应通过附录 1 中所示的软元件存储器扩展指定进行读取/写入。
- QnA 兼容 3E 帧、QnA 兼容 3C 帧、QnA 兼容 4C 帧、4E 帧用指令
 - “ 0613 ”、 “ 1613 ” : 应通过 3.4 节中所示的 QnA 兼容 3E/3C/4C 帧、4E 帧用指令 “ 0613 ”、“ 1613 ” 进行读取/写入。
 - “ 0601 ”、 “ 1601 ” : 应通过 3.5 节中所示的 QnA 兼容 3E/3C/4C 帧、4E 帧用指令 “ 0601 ”、“ 1601 ” 进行读取/写入。
- A 兼容 1E/1C 帧用指令
 - “ 0E ”、“ 0F ” : 应通过第 6 章中所示的 A 兼容 1E 帧用指令 “ 0E ”、“ 0F ” 进行读取/写入。
 - “ TR ”、“ TW ” : 应通过第 5 章中所示的 A 兼容 1C 帧用指令 “ TR ”、“ TW ” 进行读取/写入。

要点	
C24/E71 也包含在内，智能功能模块的缓冲存储器可分为可对各模块进行读取·写入的区域、只读区域，写入专用区域、操作系统用的用户使用禁止区域。应按照所使用模块的手册中的说明执行本功能。 如果进行了错误的读取·写入，可编程控制器 CPU、各智能功能模块中有可能发生出错。	

附录 3 通过 MC 协议进行通信的可编程控制器侧的处理时间

以下介绍通过 MC 协议进行通信时的可编程控制器侧的处理时间有关内容。

附录 3.1 可编程控制器 CPU 的处理时间(扫描时间的延长时间)

在基于 MC 协议进行的数据通信中，对于来自于外部设备的处理请求，在可编程控制器 CPU 处于运行中的情况下每次 END 时对 1 次可处理的点数进行处理。

此时的扫描时间的延迟时间以及处理所需的扫描次数如下所示。

(1) 经由 C24/E71 的可编程控制器 CPU 的处理时间

(QnA 兼容 3E/3C/4C 帧、4E 帧用指令的情况下)

(a) Q01CPU 的情况下

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms] (扫描时间的延长)		处理所需扫描次数	
						Q01CPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
						访问点数 1) 时	访问点数 2) 时		
软件元件存储器	多个块批量读取	字单位	0406	0000	1/960	1.44	9.10	1	1
	多个块批量写入	字单位	1406	0000	1/960	1.46	8.90	1	2
	批量读取	位单位	0401	0001	1/7904	1.44	2.25	1	1
		字单位		0000	1/960	1.45	2.54	1	1
	批量写入	位单位	1401	0001	1/7904	1.46	2.89	1	2
		字单位		0000	1/960	1.47	2.66	1	2
	随机读取	字单位	0403	0000	1/192	1.46	4.74	1	1
	测试 * 随机写入	位单位	1402	0001	1/188	1.46	7.40	1	2
		字单位		0000	1/160 (* ¹)	1.48	4.32	1	2
	监视数据登录	字单位	0801	0000	1/192 (* ²)	0.00	0.00	0	0
智能功能模块	监视		0802	0000	1/192 (* ²)	1.44	5.92	1	1
	批量读取		0601	0000	1/1920	1.71	3.95	1	1
	批量写入		1601	0000		1.78	3.97	1	2

(转下页)

项目		指令	子指令	访问点数 1)/2)	延迟时间[ms] (扫描时间的延长)		处理所需扫描次数	
					Q01CPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
					访问点数 1) 时	访问点数 2) 时		
可编程控制器 CPU	CPU 型号读取	0101	0000	(1 站)	1.68	—	1	1
全局		1618	000n	—	—		—	
接通请求		2101	—	—	—			

*1 是仅指定字访问点数进行了访问时的处理时间。

*2 是仅指定双字访问点数进行了访问时的处理时间。

要点
(1) 对于可编程控制器 CPU，通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下，需要等待直至其它处理结束为止，处理所需的扫描次数也将增加。 (2) 扫描时间的延长对控制有所影响的情况下，应以较少的点数分为多次进行访问。

(b) Q02HCPU 的情况下

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms] (扫描时间的延长)		处理所需的扫描次数	
						Q02HCPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
						访问点数 1) 时	访问点数 2) 时		
软元件存储器	多个块批量读取	字单位	0406	0000	1/960	0.11	2.80	1	1
	多个块批量写入	字单位	1406	0000	1/960	0.13	3.30	1	2
	批量读取	位单位	0401	0001	1/7904	0.10	0.24	1	1
		字单位		0000	1/960	0.09	0.16	1	1
	批量写入	位单位	1401	0001	1/7904	0.11	0.26	1	2
		字单位		0000	1/960	0.09	0.16	1	2
	随机读取	字单位	0403	0000	1/192	0.14	0.33	1	1
	测试 * 随机写入	位单位	1402	0001	1/188	0.14	3.10	1	2
		字单位		0000	1/160 (* ¹)	0.15	2.05	1	2
	监视数据登录	字单位	0801	0000	1/192 (* ²)	0.00	0.00	0	0
	监视		0802	0000	1/192 (* ²)	0.16	0.92	1	1
智能功能模块	批量读取		0601	0000	1/1920	0.00	0.00	0	0
	批量写入		1601	0000				0	1
可编程控制器 CPU	CPU 型号读取		0101	0000	(1 站)		—	0	0

(转下页)

项目	指令	子指令	访问点数 1)/2)	延迟时间[ms] (扫描时间的延长)		处理所需的扫描次数	
				Q02HCPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
				访问点数 1)时	访问点数 2)时		
目录・文件信息读取	1810	0000	1/30	0.00	0.00	0	0
目录・文件信息搜索	1811	0000	—		—	0	0
新建文件	1820	0000	1/—	0.07 (*3)		1	2
文件删除	1822	0000		0.06 (*3)		0	1
文件复制	1824	0000		0.07		1	2
文件属性更改	1825	0000				0	1
文件创建日期更改	1826	0000				0	1
打开文件	1827	0000	0.00 (*3)	0.07		0	0
文件读取	1828	0000	1/1920		0.07	1	1
文件写入	1829	0000				1	2
文件关闭	182A	0000			0	0	
全局	1618	000n	1/—	0.00	—		
接通请求	2101	—					

*1 是仅指定字访问点数进行了访问时的处理时间。

*2 是仅指定双字访问点数进行了访问时的处理时间。

*3 以大小为“0 字节”的文件为对象。

要点
(1) 对于可编程控制器 CPU，通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下，需要等待直至其它处理结束为止，处理所需的扫描次数也将增加。 (2) 扫描时间的延长对控制有所影响的情况下，应以较少的点数分为多次进行访问。

(c) Q06UDHCPU 的情况下

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms](*3) (扫描时间的延长)		处理所需的扫描次数(*3)	
						Q06UDHCPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
						访问点数 1) 时	访问点数 2) 时		
软元件存储器	多个块批量读取	字单位	0406	0000	1/960	0.45	3.10	1	1
	多个块批量写入	字单位	1406	0000	1/960	0.44	2.85	1	2
	批量读取	位单位	0401	0001	1/7904	0.46	1.22	1	1
		字单位		0000	1/960	0.45	1.05	1	1
	批量写入	位单位	1401	0001	1/7904	0.45	1.23	1	2
		字单位		0000	1/960	0.46	0.98	1	2
	随机读取	字单位	0403	0000	1/192	0.46	3.03	1	1
	测试 * 随机写入	位单位	1402	0001	1/188	0.47	3.10	1	2
		字单位		0000	1/160 (* ¹)	0.47	2.44	1	2
	监视数据登录	字单位	0801	0000	1/192 (* ²)	0.00	0.00	0	0
	监视		0802	0000	1/192 (* ²)	0.46	3.34	1	1
智能功能模块	批量读取		0601	0000	1/1920	0.44	1.43	1	1
	批量写入		1601	0000		0.48	1.38	1	2
可编程控制器 CPU	CPU 型号读取		0101	0000	(1 站)	0.42	—	1	1

(转下页)

项目	指令	子指令	访问点数 1)/2)	延迟时间[ms](*) (扫描时间的延长)		处理所需的扫描次数(*)	
				Q06UDHCPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
				访问点数 1) 时	访问点数 2) 时		
目录·文件信息读取	1810	0000	1/30	0.49	0.95	1	1
目录·文件信息搜索	1811	0000	—	0.51	—	1	1
新建文件	1820	0000	1/—	0.66		1	2
文件删除	1822	0000		0.58		1	2
文件复制	1824	0000		0.86		1	2
文件属性更改	1825	0000		0.55		1	2
文件创建日期更改	1826	0000		0.52		1	2
打开文件	1827	0000		0.60		1	1
文件读取	1828	0000	1/1920	0.47	0.95	1	1
文件写入	1829	0000		0.50	0.93	1	2
文件关闭	182A	0000	1/—	0.46	—	1	1
全局	1618	000n		0.00		—	
接通请求	2101	—		0.00			

*1 是仅指定字访问点数进行了访问时的处理时间。

*2 是仅指定双字访问点数进行了访问时的处理时间。

*3 是在 GX Works2 或 GX Developer 的可编程控制器参数(可编程控制器系统设置)中, 将“服务处理设置”的“指定服务处理次数”设置为 1 次时的处理时间。

要点
(1) 对于可编程控制器 CPU, 通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下, 需要等待直至其它处理结束为止, 处理所需的扫描次数也将增加。 (2) 希望减少扫描时间的延迟时间的情况下, 应在可编程控制器参数(可编程控制器系统设置)的“服务处理设置”中, 对可编程控制器 CPU 的延迟时间进行调整。 关于“服务处理设置”, 请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。 (3) 扫描时间的延长对控制有所影响的情况下, 应以较少的点数分为多次进行访问。

(d) L02CPU、L02CPU-P 的情况下

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms](*4) (扫描时间的延长)		处理所需的扫描次数(*4)	
						L02CPU、L02CPU-P		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
						访问点数 1) 时	访问点数 2) 时		
软元件存储器	多个块批量读取	字单位	0406	0000	1/960 (* 3)	0.49	3.22	1	1
	多个块批量写入	字单位	1406	0000	1/960 (* 3)	0.49	2.93	1	2
	批量读取	位单位	0401	0001	1/7904	0.74	1.34	1	1
		字单位		0000	1/960	0.50	1.17	1	1
	批量写入	位单位	1401	0001	1/7904	0.50	1.37	1	2
		字单位		0000	1/960	0.50	1.11	1	2
	随机读取	字单位	0403	0000	1/192	0.55	3.18	1	1
	测试 * 随机写入	位单位	1402	0001	1/188	0.52	3.18	1	2
		字单位		0000	1/160 (* 1)	0.53	2.64	1	2
	监视数据登录	字单位	0801	0000	1/192 (* 2)	0.00	0.00	0	0
	监视		0802	0000	1/192 (* 2)	0.53	3.46	1	1
智能功能模块	批量读取		0601	0000	1/1920	0.47	1.50	1	1
	批量写入		1601	0000		0.51	1.53	1	2
可编程控制器 CPU	CPU 型号读取		0101	0000	(1 站)	0.46	—	1	1

(转下页)

项目	指令	子指令	访问点数 1)/2)	延迟时间[ms](*4) (扫描时间的延长)		处理所需的扫描次数(*4)	
				L02CPU、L02CPU-P		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
				访问点数 1)时	访问点数 2)时		
目录·文件信息读取	1810	0000	1/30	0.56	0.88	1	1
目录·文件信息搜索	1811	0000	—	0.56	—	1	1
新建文件	1820	0000	1/—	0.86		1	2
文件删除	1822	0000		0.87		1	2
文件复制	1824	0000		0.84		1	2
文件属性更改	1825	0000		0.64		1	2
文件创建日期更改	1826	0000		0.60		1	2
打开文件	1827	0000		0.73		1	1
文件读取	1828	0000	1/1920	0.55	0.89	1	1
文件写入	1829	0000	1/—	0.56	0.92	1	2
文件关闭	182A	0000		0.50	—	1	1
全局	1618	000n		0.00		—	
接通请求	2101	—		0.00			

- *1 是仅指定字访问点数进行了访问时的处理时间。
- *2 是仅指定双字访问点数进行了访问时的处理时间。
- *3 是将块数指定为 120 后进行了访问时的处理时间。
- *4 是在 GX Works2 或 GX Developer 的可编程控制器参数(可编程控制器系统设置)中,将“服务处理设置”的“指定服务处理次数”设置为 1 次时的处理时间。

要点
(1) 对于可编程控制器 CPU,通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下,需要等待直至其它处理结束为止,处理所需的扫描次数也将增加。 (2) 希望减少扫描时间的延迟时间的情况下,应在可编程控制器参数(可编程控制器系统设置)的“服务处理设置”中,对可编程控制器 CPU 的延迟时间进行调整。 关于“服务处理设置”,请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。 (3) 扫描时间的延长对控制有所影响的情况下,应以较少的点数分为多次进行访问。

(e) L26CPU-BT、L26CPU-PBT 的情况下

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms] (*4) (扫描时间的延长)		处理所需的扫描次数(*4)	
						L26CPU-BT、L26CPU-PBT		设置为“RUN 中 允许”时	设置为“RUN 中 禁止”时
						访问点数 1) 时	访问点数 2) 时		
软元件存储器	多个块批量读取	字单位	0406	0000	1/960 (* 3)	0.40	3.05	1	1
	多个块批量写入	字单位	1406	0000	1/960 (* 3)	0.41	2.76	1	2
	批量读取	位单位	0401	0001	1/7904	0.40	1.14	1	1
		字单位		0000	1/960	0.41	0.98	1	1
	批量写入	位单位	1401	0001	1/7904	0.41	1.20	1	2
		字单位		0000	1/960	0.42	0.94	1	2
	随机读取	字单位	0403	0000	1/192	0.43	3.04	1	1
	测试 * 随机写入	位单位	1402	0001	1/188	0.44	3.04	1	2
		字单位		0000	1/160 (* 1)	0.42	2.49	1	2
智能功能模块	监视数据登录	字单位	0801	0000	1/192 (* 2)	0.00	0.00	0	0
	监视		0802	0000	1/192 (* 2)	0.43	3.29	1	1
可编程控制器 CPU	批量读取		0601	0000	1/1920	0.39	1.38	1	1
	批量写入		1601	0000		0.41	1.44	1	2
CPU 型号读取			0101	0000	(1 站)	0.38	—	1	1

(转下页)

项目	指令	子指令	访问点数 1)/2)	延迟时间[ms] (*4) (扫描时间的延长)		处理所需的扫描次数(*4)	
				L26CPU-BT、L26CPU-PBT		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
				访问点数 1)时	访问点数 2)时		
目录·文件信息读取	1810	0000	1/30	0.45	0.84	1	1
目录·文件信息搜索	1811	0000	—	0.46	—	1	1
新建文件	1820	0000	1/—	0.72		1	2
文件删除	1822	0000		0.86		1	2
文件复制	1824	0000		0.82		1	2
文件属性更改	1825	0000		0.52		1	2
文件创建日期更改	1826	0000		0.48		1	2
打开文件	1827	0000		0.62		1	1
文件读取	1828	0000	1/1920	0.43	0.84	1	1
文件写入	1829	0000		0.46	0.81	1	2
File close	182A	0000	1/—	0.42	—	1	1
全局	1618	000n		0.00		—	
接通请求	2101	—		0.00			

- *1 是仅指定字访问点数进行了访问时的处理时间。
- *2 是仅指定双字访问点数进行了访问时的处理时间。
- *3 是将块数指定为 120 后进行了访问时的处理时间。
- *4 是在 GX Works2 或 GX Developer 的可编程控制器参数(可编程控制器系统设置)中，将“服务处理设置”的“指定服务处理次数”设置为 1 次时的处理时间。

要点
(1) 对于可编程控制器 CPU，通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下，需要等待直至其它处理结束为止，处理所需的扫描次数也将增加。 (2) 希望减少扫描时间的延迟时间的情况下，应在可编程控制器参数(可编程控制器系统设置)的“服务处理设置”中，对可编程控制器 CPU 的延迟时间进行调整。 关于“服务处理设置”，请参阅所使用的 CPU 模块的用户手册(功能解说/程序基础篇)。 (3) 扫描时间的延长对控制有所影响的情况下，应以较少的点数分为多次进行访问。

(2) 使用串行通信功能时的可编程控制器 CPU 的处理时间(QnA 兼容 4C 帧用指令(格式 4)的情况下)

项目			指令	子指令	访问点数 1)/2)	延迟时间[ms] (扫描时间的延长)		处理所需的扫描次数	
						Q00CPU、Q01CPU		设置为“RUN 中允许”时	设置为“RUN 中禁止”时
						访问点数 1)时	访问点数 2)时		
软元件存储器	批量读取	位单位	0401	0001	1/3584	0.8	13.1	3	
		字单位		0000	1/480	0.8	6.2	3	
	批量写入	位单位	1401	0001	1/3584	0.9	7.5	3	
		字单位		0000	1/480	0.8	9.3	3	
	随机读取	字单位	0403	0000	1/96	0.8	7.5	3	
	测试 * 随机写入	位单位	1402	0001	1/94	0.8	7.1	3	
		字单位		0000	1/80 (* ¹)	0.9	8.2	3	
	监视数据登录	字单位	0801	0000	1/96 (* ²)	0.8	7.3	3	
	监视		0802	0000	1/96 (* ²)	0.7	2.4	3	

*1 是仅指定字访问点数进行了访问时的处理时间。

*2 是仅指定双字访问点数进行了访问时的处理时间。

要点
(1) 对于可编程控制器 CPU，通过 END 处理仅进行上述项目之一的处理。 GX Works2 或 GX Developer、各模块等对相应的可编程控制器 CPU 同时进行了访问的情况下，需要等待直至其它处理结束为止，处理所需的扫描次数也将增加。 (2) 扫描时间的延长对控制有所影响的情况下，应以较少的点数分为多次进行访问。

索引

[B]

报文格式

(A 兼容 1C 帧)	5-1
(A 兼容 1E 帧)	6-2
(QnA 兼容 2C 帧)	4-1
(QnA 兼容 3C 帧)	3-16
(QnA 兼容 3E 帧)	3-3
(QnA 兼容 4C 帧)	3-25
(4E 帧)	3-3

[C]

传送顺控程序的初始化	3-250
传送顺控程序的传送时间	2-24
串行通信功能	2-32
簇	3-151

[D]

多 CPU 系统	2-28
----------------	------

[F]

FAT	3-149
-----------	-------

[G]

关键字(口令)	3-158
---------------	-------

[H]

和校验代码	3-43
缓冲存储器的读取、写入	附录-16
回送测试	3-262

[J]

接通请求功能	3-240
--------------	-------

[K]

可编程控制器 CPU 的处理时间	附录-18
可编程控制器 CPU 的状态控制	3-140
可编程控制器侧访问时机	2-7

可访问站	2-13
------------	------

可以访问的智能功能模块	3-134
-------------------	-------

控制步骤

(A 兼容 1C 帧)	5-1
(A 兼容 1E 帧)	6-2
(QnA 兼容 2C 帧)	4-1
(QnA 兼容 3C 帧)	3-16
(QnA 兼容 3E 帧)	3-3
(QnA 兼容 4C 帧)	3-25
(4E 帧)	3-3

控制代码	3-36
------------	------

扩展文件寄存器的读取、写入	5-36
---------------------	------

[Q]

其它站访问时的传送时间	2-23
驱动器内存的整理	3-154
全局功能	3-235

[R]

软元件存储器的扩展指定	附录-1
-------------------	------

[S]

数据通信用帧类型	2-1
----------------	-----

[T]

头部	3-3
----------	-----

[W]

文件的打开	3-184
关闭	3-186
复制	3-194
删除	3-192
新建	3-189
读取	3-187

[Y]

应用数据	3-3
远程口令	3-286

允许访问范围.....2-4

运行中写入2-9

[Z]

指令(A 兼容 1C 帧)5-15

 (A 兼容 1E 帧)6-14

 (QnA 兼容 2C 帧)4-8

 (QnA 兼容 3E/3C/4C 帧、4E 帧)3-53

至文件的写入.....3-190

 创建日期的更改3-197

 信息列表的读取3-200

 信息读取3-181

 控制3-155

 属性的更改3-198

 有无的读取3-206

质保

使用之前请确认以下产品质保的详细说明。

1. 免费质保期限和免费质保范围

在免费质保期内使用本产品时如果出现任何属于三菱电机责任的故障或缺陷(以下称“故障”),则经销商或三菱电机服务公司将负责免费维修。

但是如果需要在国内现场或海外维修时,则要收取派遣工程师的费用。对于涉及到更换故障模块后的任何再试运转、维护或现场测试,三菱电机将不负任何责任。

[免费质保期限]

免费质保期限为自购买日或交货的一年内。

注意产品从三菱电机生产并出货之后,最长分销时间为6个月,生产后最长的免费质保期为18个月。维修零部件的免费质保期不得超过修理前的免费质保期。

[免费质保范围]

(1) 范围局限于按照使用手册、用户手册及产品上的警示标签规定的使用状态、使用方法和使用环境正常使用的情况下。

(2) 以下情况下,即使在免费质保期内,也要收取维修费用。

- 1) 因不当存储或搬运、用户过失或疏忽而引起的故障。因用户的硬件或软件设计而导致的故障。
- 2) 因用户未经批准对产品进行改造而导致的故障等。
- 3) 对于装有三菱电机产品的用户设备,如果根据现有的法定安全措施或工业标准要求配备必需的功能或结构后本可以避免的故障。
- 4) 如果正确维护或更换了使用手册中指定的耗材(电池、背光灯、保险丝等)后本可以避免的故障。
- 5) 因火灾或异常电压等外部因素以及因地震、雷电、大风和水灾等不可抗力而导致的故障。
- 6) 根据从三菱出货时的科技标准还无法预知的原因而导致的故障。
- 7) 任何非三菱电机或用户责任而导致的故障。

2. 产品停产后的有偿维修期限

(1) 三菱电机在本产品停产后的7年内受理该产品的有偿维修。

停产的消息将以三菱电机技术公告等方式予以通告。

(2) 产品停产,将不再提供产品(包括维修零件)。

3. 海外服务

在海外,维修由三菱电机在当地的海外FA中心受理。注意各个FA中心的维修条件可能会不同。

4. 意外损失和间接损失不在质保责任范围内

无论是否在免费质保期内,对于任何非三菱电机责任的原因而导致的损失、机会损失、因三菱电机产品故障而引起的用户利润损失、无论能否预测的特殊损失和间接损失、事故赔偿、除三菱电机以外产品的损失赔偿、用户更换设备、现场机械设备的再调试、运行测试及其它作业等,三菱电机将不承担责任。

5. 产品规格的改变

目录、手册或技术文档中的规格如有改变,恕不另行通知。

Microsoft、Windows、Windows NT、Windows Vista、Excel 是 Microsoft Corporation 公司在美国及其它国家的注册商标。

Pentium 是 Intel Corporation 在美国及其它国家的商标。

Ethernet 是美国 Xerox Corporation 公司的商标。

MODBUS 是 Schneider Electric SA 的注册商标。

本手册中使用的其它公司名和产品名是相应公司的商标或注册商标。

MELSEC-Q/L MELSEC通讯协议 参考手册



三菱电机自动化(中国)有限公司

地址：上海市虹桥路1386号三菱电机自动化中心
邮编：200336
电话：021-23223030 传真：021-23223000
网址：www.meach.cn

书号	SH(NA)-080414CHN-C(1210)MEACH
印号	MEACH-Q/LCPU-SCP-RM(1210)

内容如有更改
恕不另行通知