

Omron NJ Ethernet 驱动程序

© 2019, PTC Inc. 保留所有权利。

目录

Omron NJ Ethernet 驱动程序	1
目录	2
Omron NJ Ethernet 驱动程序	7
概述	7
设置	8
通信路由和定时	8
连接路径规范	8
路由示例	9
通道属性 - 常规	12
通道属性 - 以太网通信	13
通道属性 - 写入优化	13
通道属性 - 高级	14
驱动程序设备属性	15
设备属性 - 常规	15
设备属性 - 扫描模式	16
设备属性 - 定时	17
设备属性 - 自动降级	17
设备属性 - 标记生成	18
设备属性 - 通信参数	19
设备属性 - 选项	20
设备属性 - 冗余	21
优化通信	22
通道属性 - 常规	22
通道属性 - 以太网通信	23
通道属性 - 写入优化	23
通道属性 - 高级	24
设备属性 - 常规	24
设备属性 - 扫描模式	26
设备属性 - 定时	26
设备属性 - 自动降级	27
设备属性 - 标记生成	27
设备属性 - 通信参数	29
设备属性 - 选项	29
设备属性 - 冗余	30
优化通信	32
优化应用程序	32
性能统计信息和调优	33
数据类型说明	34
地址说明	35
地址格式	36
标记范围	38

预定义术语标记	39
自动生成标记数据库	40
标记层次	40
事件日志消息	42
尝试写入标记时出现内部错误。意外的数据类型。 标记地址 = '<地址>', 数据类型 = '<类型>', DTRV = <代码>。	42
从设备上载控制器项目时出现以下错误。正在改为使用符号协议。	42
发生未知错误。	42
内存资源不足。	42
在同步时检测到无效或损坏的控制器项目。将立即重试同步。	42
同步时检测到项目下载。稍后将重试同步。	42
上传项目信息期间发生封装错误。 封装错误 = <代码>。	42
上传项目信息期间发生错误。 CIP 错误 = <代码>, 扩展错误 = <代码>。	43
上传项目信息期间发生帧错误。	43
上传项目信息期间 CIP 连接超时	43
数据库错误。上传项目信息期间 CIP 连接超时。	43
数据库错误。上载项目信息时发生错误。 CIP 错误 = <代码>, 扩展错误 = <代码>。	43
数据库错误。注册会话请求期间发生封装错误。 封装错误 = <代码>。	44
数据库错误。注册会话请求期间发生帧错误。	44
数据库项目。发生内部错误。	44
数据库错误。转发打开请求时发生封装错误。 封装错误 = <代码>。	44
数据库错误。fwd. 打开请求再无可连接。	44
数据库错误。在转发开放请求期间发生错误。 CIP 错误 = <代码>, 扩展错误 = <代码>。	44
数据库错误。在转发开放请求期间发生帧错误。	45
数据库错误。上载项目信息时发生封装错误。 封装错误 = <代码>。	45
数据库错误。上载项目信息时发生帧错误。	45
从设备接收的帧有误。	46
请求设备期间发生错误。 CIP 错误 = <代码>, 扩展错误 = <代码>。	46
对标记的写入请求由于帧错误而失败。 标记地址 = '<地址>'。	46
对标记的读取请求由于帧错误而失败。 标记地址 = '<地址>'。	46
块读取请求由于帧错误而失败。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	47
块读取请求由于帧错误而失败。 块大小 = <数字> (字节), 标记名称 = '<标记>'。	47
无法写入标记。 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。	47
无法读取标记。 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。	47
无法读取设备上的块。 块大小 = <数字> (元素), 起始标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。	48
无法读取设备上的块。 块大小 = <数字> (字节), 标记名称 = '<标记>', CIP 错误 = <代码>, 扩展错误 = <代码>。	48
无法写入标记。控制器标记数据类型未知。 标记地址 = '<地址>', 数据类型 = <类型>。	48
无法读取标记。控制器标记数据类型未知。标记已取消激活。 标记地址 = '<地址>', 数据类型 = <类型>。	48
无法读取设备上的块。控制器标记数据类型未知。标记已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = <类型>。	49
无法写入标记。不支持数据类型。 标记地址 = '<地址>', 数据类型 = <类型>。	49
无法读取标记。不支持数据类型。标记已取消激活。 标记地址 = '<地址>', 数据类型 = <类型>。	49

无法读取设备上的块。不支持数据类型。块已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = '<类型>'。	49
无法写入标记。此标记数据类型非法。 标记地址 = '<地址>', 数据类型 = <类型>。	49
无法读取标记。此标记数据类型非法。 标记地址 = '<地址>', 数据类型 = '<类型>'。	50
无法读取设备上的块。此块的数据类型是非法的。 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = '<类型>'。	50
无法写入标记。标记不支持多元素数组。 标记地址 = '<地址>'。	50
无法读取标记。标记不支持多元素数组。标记已取消激活。 标记地址 = '<地址>'。	50
无法读取块。块不支持多元素数组。块已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	51
无法写入标记。原生标记大小不匹配。 标记地址 = '<地址>'。	51
无法读取标记。原生标记大小不匹配。 标记地址 = '<地址>'。	51
无法读取设备上的块。原生标记大小不匹配。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	51
无法读取设备上的块。原生标记大小不匹配。 块大小 = <数字> (字节), 标记名称 = '<标记>'。	51
无法写入标记。 标记地址 = '<地址>'。	51
无法读取标记。标记已取消激活。 标记地址 = '<地址>'。	52
无法读取块。块已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	52
无法读取设备上的块。块已取消激活。 块大小 = <数字> (字节), 标记名称 = '<标记>'。	52
无法读取标记。内存无效。 标记地址 = '<地址>'。	52
无法读取标记。此标记数据类型非法。 标记地址 = '<地址>', 数据类型 = '<类型>'。	52
无法读取设备上的块。内存无效。标记已取消激活。 标记地址 = '<地址>'。	52
无法读取设备上的块。内存无效。块已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	52
无法写入地址。内存无效。 标记地址 = '<地址>'。	52
无法读取设备上的块。块已取消激活。 块大小 = <数字> (元素), 起始标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。	52
在读取标记时, 设备所返回的数据超过预期。请验证地址, 包括元素偏移和该偏移中的所有维度。 标记地址 = '<地址>'。	53
在读取块时, 设备所返回的数据超过预期。请验证地址, 包括元素偏移和该偏移中的所有维度。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	53
无法写入标记。地址超出了当前 CIP 连接大小。 标记地址 = '<地址>'。	53
无法读取设备上的块。地址超出了当前 CIP 连接大小。 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。	53
无法读取标记。地址超出了当前 CIP 连接大小。 标记地址 = '<地址>'。	54
本设备不支持请求的 CIP 连接大小。正在自动回退到最大大小。 请求的大小 = <数字> (字节), 最大大小 = <数字> (字节)。	54
对于 XML 元素, 当前值在此模型上不受支持。正在自动设置为新值。 当前值 = '<值>', XML 元素 = '{<命名空间><元素>', 模型 = '<模型>', 新值 = '<值>'。	54
数据库错误。复杂类型的数据类型不受支持。不会将此成员的标记添加到数据库。 数据类型 = <类型>, 复杂类型 = '<类型>', 成员 = '<名称>'。	54
数据库错误。无法解析标记的 CIP 数据类型。不会将标记添加到数据库。 数据类型 = <类型>, 标记名称 = '<标记>'。	54
数据库错误。标记的地址验证失败。不会将标记添加到数据库。 标记名称 = '<标记>', 标记地址 = '<地址>'。	55
无法检索设备的标识。 封装错误 = <代码>。	55
无法检索设备的标识。 CIP 错误 = <代码>, 扩展错误 = <代码>。	55
无法检索设备的标识。接收到的帧有错误。	55
请求期间发生封装错误。 封装错误 = <代码>。	56

无法为标记分配内存。 标记地址 = '<地址>'。	56
数据库错误。标记已重命名,因为它超出了最大字符长度。 标记名称 = '<标记>', 最大长度 = <数字>, 新标记名称 = '<标记>'。	56
数据库错误。数组标记已重命名,因为它超出了最大字符长度。 数组标记名称 = '<名称>', 最大长度 = <数字>, 新数组标记名称 = '<名称>'。	56
数据库状态: 标记已导入。 数据类型 = <类型>, 导入的标记数 = <数字>。	56
数据库状态: 正在生成 OPC 标记。	57
数据库状态: 正在构建标记项目, 请等待。 标记项目计数 = <数字>。	57
数据库状态: 正在检索控制器项目。	57
运行时间 = <数字> (秒)	57
符号设备读取次数 = <数字>	57
符号、数组块设备读取次数 = <数字>	57
符号、数组块缓存读取次数 = <数字>	57
符号实例非块设备读取次数 = <数字>	57
符号实例非块、数组块设备读取次数 = <数字>	57
符号实例非块、数组块缓存读取次数 = <数字>	57
符号实例块设备读取次数 = <数字>	57
符号实例块缓存读取次数 = <数字>	57
读取的标记数 = <数字>	58
发送的数据包数 = <数字>	58
接收的数据包数 = <数字>	58
初始化事务数 = <数字>	58
读取/写入事务数 = <数字>	58
平均每秒发送的数据包数 = <数字>	58
平均每秒接收的数据包数 = <数字>	58
平均每秒标记读取次数 = <数字>	58
平均每个事务的标记数 = <数字>	58
-----	58
%s 设备统计信息	58
平均设备回转时间 = <数字> (毫秒)。	58
%s 通道统计信息	59
驱动器统计信息	59
详细信息。 IP = '<地址>', 供应商 ID = <供应商>, 设备类型 = <类型>, 产品代码 = <代码>, 修订版本 = <修订版本>, 产品名称 = '<名称>', 产品序列号 = <数字>。	59
检索控制器项目出错。	59
内部驱动器出错。	59
在同步时检测到无效或损坏的控制器项目。请稍后重试。	59
在同步时检测到项目下载。请稍后重试。	59
内存资源不足。	59
发生未知错误。	59
错误代码	60
封装错误代码	60
CIP 错误代码	60
0x01 扩展错误代码	61
0x0C 扩展错误代码	63

0x1F 扩展错误代码	64
0x20 扩展错误代码	64
索引	65

Omron NJ Ethernet 驱动程序

帮助版本 1.036

目录

[概述](#)

什么是 Omron NJ Ethernet 驱动程序？

[设备设置](#)

如何配置使用此驱动程序的设备？

[优化通信](#)

如何提高此驱动程序的性能和系统通信？

[数据类型说明](#)

此驱动程序支持哪些数据类型？

[地址说明](#)

如何处理 Omron NJ Ethernet 设备上的数据位置？

[自动标记数据库生成](#)

如何在服务器中自动生成与设备特定数据相对应的标记列表？

[事件日志消息](#)

Omron NJ Ethernet 驱动程序 会产生哪些消息？

[错误代码](#)

Omron NJ Ethernet 错误代码是什么？

概述

Omron NJ Ethernet 驱动程序 提供将 Omron NJ Ethernet 控制器连接至客户端应用程序的可靠方式；其中包括 HMI、SCADA、Historian、MES、ERP 和无数自定义应用程序。

设置

支持的控制器

Omron NJ 301
Omron NJ 501

通信协议

使用 TCP/IP 的 EtherNet/IP (Ethernet 上的 CIP)。

支持的最大信道数量为 256。所支持设备的最大数量为每信道 1024 个。

信道属性

设备属性

通信路由和定时

路由可用于通过本地 NJ Ethernet/IP 单元与远程 NJ CPU 进行通信。有关路由的详细信息，请参阅 Omron NJ 系列 CPU 单元内置 Ethernet/IP 端口用户手册 W506 的第 8 章。虽然材料侧重于 NJ 到 NJ 的通信，但此原理在驱动程序与远程 NJ CPU 通信时同样适用。

路由定时

当与远程 CPU 的通信中断时，如果执行以下操作，驱动程序使用的请求超时属性将与设备定时设置中配置的有所不同：

- “标识请求”：未连接消息，用于确定远程 CPU 模型和固件版本。
- “转发开放请求”：未连接消息，用来与远程 CPU 建立高级 CIP 连接。

在这些情况下，本地设备将返回 CIP 错误 0x01 Ext. Err 0x204，此错误被定义为“未连接请求超时”，且表明无法访问远程 CPU。通常，在连接关闭后会发生这些请求，且随后将出现读取或写入请求超时。在这种情况下，该设备可能在读取或写入请求超时后快速进入错误状态，但失败标记却出现得很慢，因为它要待未连接请求超时后发生。虽然驱动程序会等待全部“未连接的请求超时”，但本地设备可能会更快地响应此错误。

此自定义超时取决于路由选择路径中的段数。每个请求仅尝试一次。有关详细信息，请参阅下表。

段数	请求超时 (秒)	示例
1	15	10.10.110.2\1\0
2	20	10.10.110.2\1\#10\2\172.16.1.3
3	25	10.10.110.2\1\#10\2\172.16.1.3\1\0
4	30	10.10.110.2\1\#10\2\192.168.1.3\1\0\2\172.16.1.4
5	35	10.10.110.2\1\#10\2\192.168.1.3\1\0\2\172.16.1.4\1\0

● **注意：**无论路由到远程 CPU 还是直接路由到本地 CPU，在执行读取、写入和自动标记生成操作时，驱动程序都将始终使用该设备的定时设置。

● 有关详细信息，请参阅 [设备属性 - 定时](#)。

连接路径规范

CIP 连接路径 (通常称为路由路径) 在“设备 ID”中进行指定。通信由 PC 上的 Omron NJ Ethernet 驱动程序发起，然后在本地 NJ CPU 或 Ethernet/IP 单元中进行定向。到达该本地单元后，“设备 ID”便会指定一条从该单元发出并到底面上的路径。随后，路由路径会将消息导向到所需的远程 NJ CPU 单元。

路由路径本身是一系列“端口/链接地址”对，这与“Omron NJ 系列 CPU 单元内置 Ethernet/IP 端口用户手册 W506”中第 8 章所述路由路径相同。在该文档中，“网络类型编号”等同于“端口”，而“远程地址”等同于“链接地址”。在路由路径中，“端口”和“链接地址”均由反斜杠 (无需空格) 来分隔。

● **注意：**当目标是本地 NJ CPU 单元时，不需要路由路径。“设备 ID”只需包含本地 CPU 单元的 IP 地址。

指示符类型	说明	格式	范围
端口 (网络类型编号)	用于指定从所讨论接口单元发出的路径。	十进制	0-65535 #0000-#FFFF

指示符类型	说明	格式	范围
	底面 (BP) 端口: 1 Ethernet/IP (EIP) 端口: 2	十六进制	
链接地址 (远程地址)	用于指定“端口”的目标。 底面 (BP) 端口: 目标单元的单元地址 Ethernet/IP (EIP) 端口: 远程 CPU 或 Ethernet/IP (EIP) 单元的 IP 地址	十进制 十六进制	单元: 0-255 IP: 有效的点分四段形式 IP* #00-#FF

*不允许为主机名。

包含路由路径

本地 CPU (0 个跃点)

本地 EIP IP\BP 端口\CPU 单元地址

远程 CPU (1 个跃点)

本地 CPU IP\BP 端口\EIP 单元地址\EIP 端口\远程 CPU IP 地址或本地 CPU IP\BP 端口\EIP 单元地址\EIP 端口\远程 EIP IP 地址\BP 端口\远程 CPU 单元地址

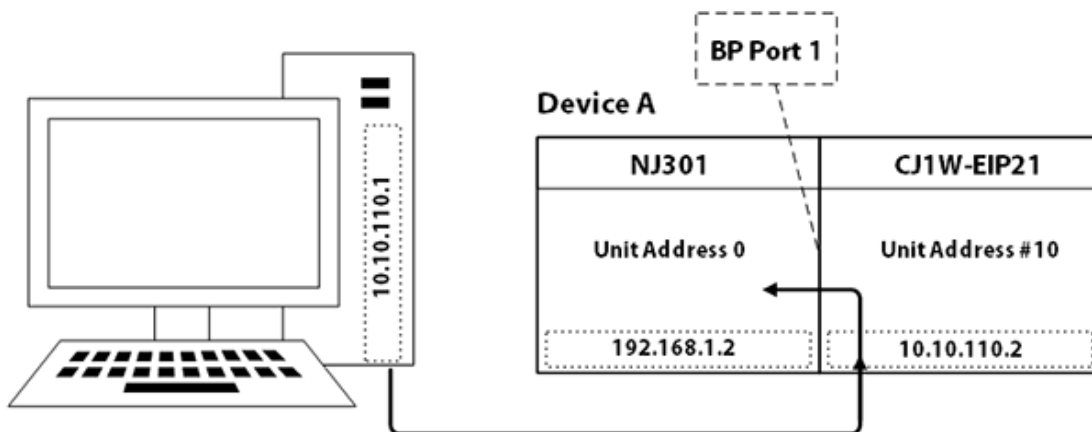
多跃点 (N 个跃点)

本地 CPU IP\BP 端口\EIP 单元地址\EIP 端口\远程 CPU IP 地址\BP 端口... \EIP 单元地址\EIP 端口\远程 CPU IP 地址或本地 CPU IP\BP 端口\EIP 单元地址\EIP 端口\远程 CPU IP 地址\BP 端口... \EIP 单元地址\EIP 端口\远程 EIP IP 地址\BP 端口\远程 CPU 单元地址

路由示例

下列路由示例中包括完整的设备 ID。

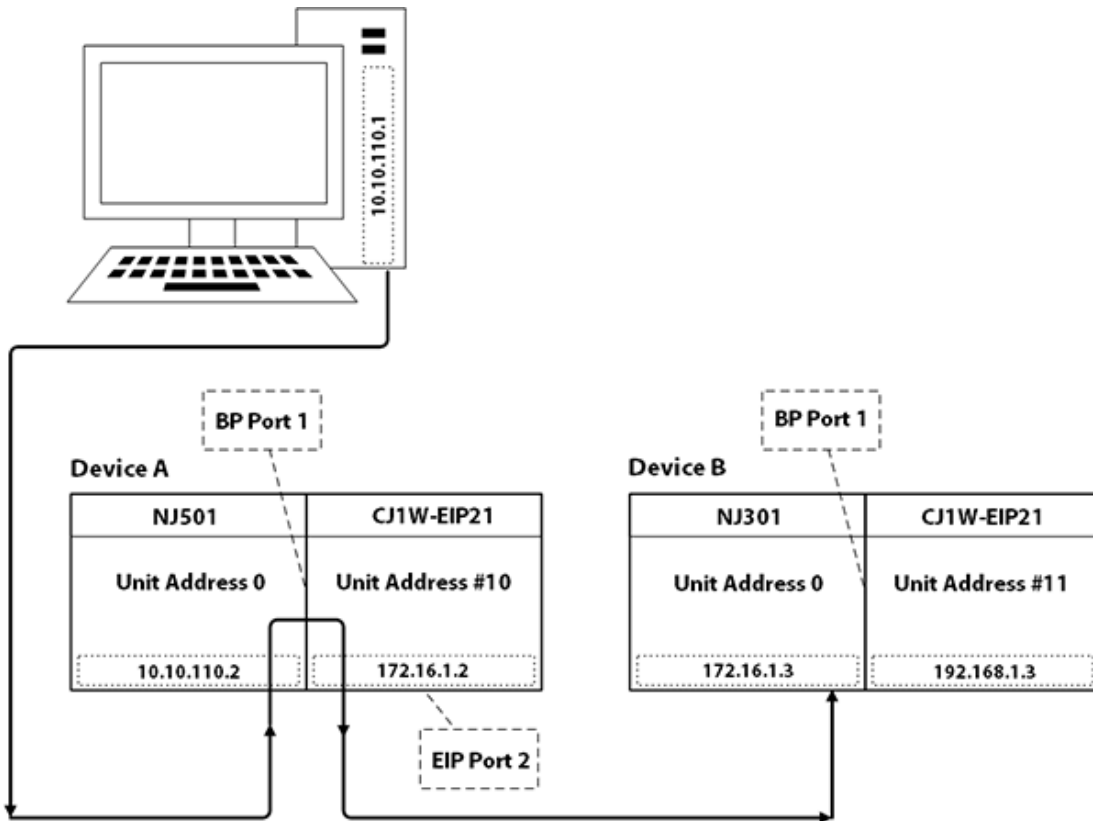
本地 Omron NJ (A): 10.10.110.2\10



10.10.110.2\110 路径的细目如下所示:

- 10.10.110.2: 设备 A 的 EIP 单元的 IP 地址
- \1: 设备 A 的 EIP 单元用以访问底面的端口编号
- \10: 设备 A 的 CPU 单元的单元地址

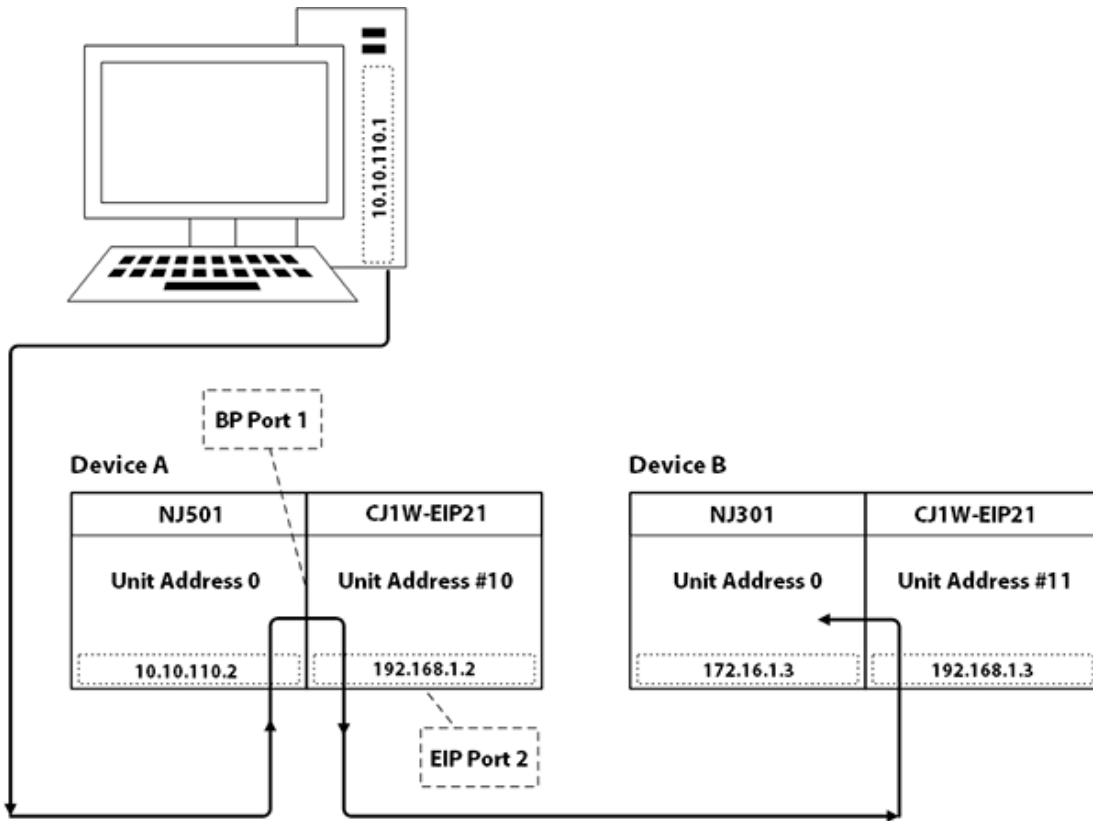
通过 Omron NJ (A) 的远程 Omron NJ (B): 10.10.110.2\1\#10\172.16.1.3



10.10.110.2\1\#10\2\172.16.1.3 路径的细目如下所示：

- 10.10.110.2: 设备 A 的 CPU 单元的 IP 地址
- \1: 设备 A 的 CPU 单元用以访问底面的端口编号
- \# 10: 设备 A 的 EIP 单元的单元地址 (10 十六进制、16 十进制)
- \2: 设备 A 的 EIP 单元用以访问 Ethernet/IP 的端口编号
- \172.16.1.3: 设备 B 的 CPU 单元的 IP 地址

通过 Omron NJ (A) 的远程 Omron NJ (B): 10.10.110.2\1\#10\2\172.16.1.3\1\0

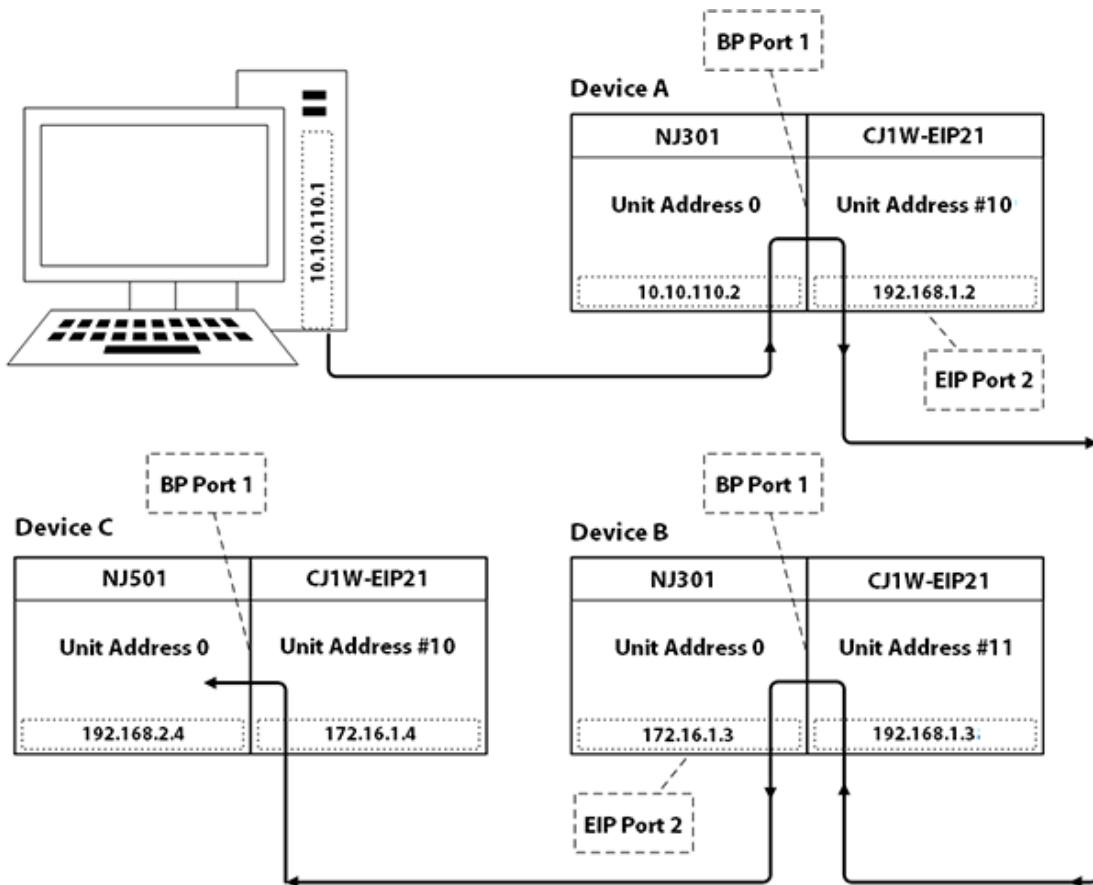


10.10.110.2\1\#10\2\192.168.1.3\1\0 路径的细目如下所示：

- 10.10.110.2: 设备 A 的 CPU 单元的 IP 地址
- \1: 设备 A 的 CPU 单元用以访问底面的端口编号
- \# 10: 设备 A 的 EIP 单元的单元地址 (10 十六进制、16 十进制)
- \2: 设备 A CJ1W-EIP21 单元用以访问 Ethernet/IP 的端口号
- 192.168.1.3: 设备 B 的 EIP 单元的 IP 地址
- \1: 设备 B 的 EIP 单元用以访问底面的端口编号
- \0: 设备 B 的 CPU 单元的单元地址

通过 Omron NJ (A) 的远程 Omron NJ

(C): 10.10.110.2\1\#10\2\192.168.1.3\1\0\2\172.16.1.4\1\0



10.10.110.2\11\#10\2\192.168.1.3\11\0\2\172.16.1.4\11\0 路径的细目如下所示：

- 10.10.110.2: 设备 A 的 CPU 单元的 IP 地址
- \1: 设备 A 的 CPU 单元用以访问底面的端口编号
- \# 10: 设备 A 的 EIP 单元的单元地址 (10 十六进制、16 十进制)
- \2: 设备 A 的 EIP 单元用以访问 Ethernet/IP 的端口编号
- \192.168.1.3: 设备 B 的 EIP 单元的 IP 地址
- \1: 设备 B 的 EIP 单元用以访问底面的端口编号
- \0: 设备 B 的 CPU 单元的单元地址
- \2: 设备 B 的 CPU 单元用以访问 Ethernet/IP 的端口号
- \172.16.1.4: 设备 C 的 EIP 单元的 IP 地址
- \1: 设备 C 的 EIP 单元用以访问背板的端口号
- \0: 设备 C 的 CPU 单元的单元地址

有关详细信息，请参阅[连接路径规范](#)。有关构建连接/路由路径的详细信息，请参阅 Omron 的 NJ 系列 CPU 单元内置 Ethernet/IP 端口用户手册 W506 的第 8 章。

通道属性 - 常规

此服务器支持同时使用多个通信驱动程序。服务器项目中使用的各个协议或驱动程序称为通道。服务器项目可以由具有相同通信驱动程序或具有唯一通信驱动程序的多个通道组成。通道充当 OPC 链路的基础构建块。此组用于指定常规通道属性，如标识属性和操作模式。

属性组	<input type="checkbox"/> 标识	
常规	名称	通道 1
写优化	说明	
高级	驱动程序	Simulator
持久存储	<input type="checkbox"/> 诊断	
	诊断数据捕获	禁用

标识

“名称”: 此通道的用户定义标识。在每个服务器项目中，每个通道名称都必须是唯一的。尽管名称最多可包含 256 个字符，但在浏览 OPC 服务器的标记空间时，一些客户端应用程序的显示窗口可能不够大。通道名称是 OPC 浏览器信息的一部分。该属性是创建通道所必需的。

● 有关保留字符的信息，请参阅服务器帮助中的“如何正确命名通道、设备、标记和标记组”。

“说明”: 有关此通道的用户定义信息。

● 这些属性 (包括 Description) 当中有很多具有关联的系统标记。

“驱动程序”(Driver): 为该通道选择的协议/驱动程序。该属性指定在通道创建期间选择的设备驱动程序。它在通道属性中为禁用设置。该属性是创建通道所必需的。

● **注意**: 服务器全天在线运行时，可以随时更改这些属性。其中包括更改通道名称以防止客户端向服务器注册数据。如果客户端在通道名称更改之前已从服务器中获取了项，那么这些项不会受到任何影响。如果客户端应用程序在通道名称更改之后发布项，并尝试通过原来的通道名称重新获取项，则该项将不被接受。考虑到这一点，一旦开发完成大型客户端应用程序，就不应对属性进行任何更改。利用“用户管理器”可防止操作员更改属性并限制对服务器功能的访问权限。

诊断

“诊断数据捕获”: 启用此选项后，通道的诊断信息即可用于 OPC 应用程序。由于服务器的诊断功能所需的开销处理量最少，因此建议在需要时使用这些功能，而在不需要时禁用这些功能。默认设置为禁用状态。

● **注意**: 如果驱动程序不支持诊断，则该属性不可用。

● 有关详细信息，请参阅服务器帮助中的“通信诊断”。

通道属性 - 以太网通信

以太网通信可用于与设备进行通信。

属性组	<input type="checkbox"/> 以太网设置	
常规	网络适配器	默认值 ...
以太网通信		
写优化		
高级		
通信序列化		

以太网设置

“网络适配器”(Network Adapter): 指定要绑定的网络适配器。如果留空或选择“默认”，则操作系统将选择默认适配器。

通道属性 - 写入优化

与任何服务器一样，将数据写入设备可能是应用程序应具备的最重要的功能。服务器旨在确保从客户端应用程序写入的数据能够准时发送到设备。为了达到此目标，服务器提供了用来满足特定需求以提高应用程序响应能力的优化属性。

属性组	<input type="checkbox"/> 写优化	
常规	优化方法	仅写入所有标记的最新值
写优化	占空比	10
高级		
持久存储		

写入优化

“优化方法”：控制如何将写入数据传递至底层通信驱动程序。选项包括：

- “写入所有标记的所有值”：此选项可强制服务器尝试将每个值均写入控制器。在此模式下，服务器将持续收集写入请求并将它们添加到服务器的内部写入队列。服务器将对写入队列进行处理并尝试通过将数据尽快写入设备来将其清空。此模式可确保从客户端应用程序写入的所有数据均可发送至目标设备。如果写入操作顺序或写入项的内容必须且仅能显示于目标设备上，则应选择此模式。
- “写入非布尔标记的最新值”：由于将数据实际发送至设备需要一段时间，因此对同一个值的多次连续写入会存留于写入队列中。如果服务器要更新已位于写入队列中的某个写入值，则需要大大减少写入操作才能获得相同的最终输出值。这样一来，便不会再有额外的写入数据存留于服务器队列中。几乎就在用户停止移动滑动开关时，设备中的值达到其正确值。根据此模式的规定，任何非布尔值都会在服务器的内部写入队列中更新，并在下一个可能的时机发送至设备。这可以大大提高应用性能。
- **注意**：该选项不会尝试优化布尔值的写入。它允许用户在不影响布尔运算的情况下优化 HMI 数据的操作，例如瞬时型按钮等。
- “写入所有标记的最新值”：该选项采用的是第二优化模式背后的理论并将其应用至所有标记。如果应用程序只需向设备发送最新值，则该选项尤为适用。此模式会通过当前写入队列中的标记发送前对其进行更新来优化所有的写入操作。此为默认模式。

“占空比”(Duty Cycle)：用于控制写操作与读操作的比率。该比率始终基于每一到十次写入操作对应一次读取操作。占空比的默认设置为 10，这意味着每次读取操作对应十次写入操作。即使在应用程序执行大量的连续写入操作时，也必须确保足够的读取数据处理时间。如果将占空比设置为 1，则每次读取操作对应一次写入操作。如果未执行任何写入操作，则会连续处理读取操作。相对于更加均衡的读写数据流而言，该特点使得应用程序的优化可通过连续的写入操作来实现。

● **注意**：建议在将应用程序投入生产环境前使其与写入优化增强功能相兼容。

通道属性 - 高级

此组用于指定高级通道属性。并非所有驱动程序都支持所有属性，因此不会针对不支持的设备显示“高级”组。

属性组	<input type="checkbox"/> 非规范浮点数处理	
常规	浮点值	替换为零
以太网通信	<input type="checkbox"/> 设备间延迟	
写优化	设备间延迟 (毫秒)	0
高级		
通信序列化		

“非规范浮点数处理”：非规范值定义为无穷大、非数字 (NaN) 或不正规编号。默认值为“替换为零”。具有原生浮点数处理功能的驱动程序可能会默认设置为“未修改”。通过非规范浮点数处理，用户可以指定驱动程序处理非规范 IEEE-754 浮点数据的方式。选项说明如下：

- “替换为零”：此选项允许驱动程序在将非规范 IEEE-754 浮点值传输到客户端之前，将其替换为零。
- “未修改”：此选项允许驱动程序向客户端传输 IEEE-754 不正规、规范、非数字和无穷大值，而不进行任何转换或更改。

● **注意**：如果驱动程序不支持浮点值或仅支持所显示的选项，则此属性不可用。根据通道的浮点规范化设置，将仅对实时驱动程序标记 (如值和数组) 进行浮点规范化。例如，此设置不会影响 EFM 数据。

● 有关浮点值的详细信息，请参阅服务器帮助中的“如何使用非规范化浮点值”。

“设备间延迟”: 指定在接收到同一通道上的当前设备发出的数据后, 通信通道向下一设备发送新请求前等待的时间。设置为零 (0) 将禁用延迟。

● **注意**: 此属性并不适用于所有驱动程序、型号和相关设置。

驱动程序设备属性

设备属性将被分成组。有关该组中设置的详细信息, 请单击下面的链接。

[常规](#)

[扫描模式](#)

[定时](#)

[自动降级](#)

[标记生成](#)

[通信参数](#)

[选件](#)

[冗余](#)

设备属性 - 常规

属性组	
常规	标识
扫描模式	名称
定时	说明
自动降级	驱动程序
标记生成	型号
通信参数	通道分配
选件	ID
冗余	操作模式
	数据收集
	模拟

名称
指定此对象的标识。

默认值 确定 取消 应用 帮助

标识

“名称”: 此设备的用户定义标识。

“说明”: 有关此设备的用户定义信息。

“通道分配”: 该设备当前所属通道的用户定义名称。

驱动程序: 为该设备选择的协议驱动程序。

“型号”: 设备的特定版本。有关支持 *FINS* 通信服务的型号列表, 请参阅制造商网站。

ID: ID 用于指定到目标 NJ CPU 单元的路径。

- 将本地 NJ CPU 单元的设备 ID 指定为本地 CPU 单元的 IP 或主机名。该设备 ID 必须是点分四段形式的有效 IP 地址或主机名。例如, "192.168.1.100" 或 "NJ001"。
- 将远程 NJ CPU 单元的设备 ID 指定为本地 CPU 或 Ethernet/IP 单元的 IP 或主机名, 外加指向远程 CPU 单元的 CIP 连接路径 (也称为路由路径)。例如, "192.168.1.100\1#102\10.10.110.2"。

有关连接路径语法的信息, 请参阅[连接路径规范](#)。

操作模式

“数据收集”: 此属性控制设备的活动状态。尽管默认情况下会启用设备通信, 但可使用此属性禁用物理设备。设备处于禁用状态时, 不会尝试进行通信。从客户端的角度来看, 数据将标记为无效, 且不接受写入操作。通过此属性或设备系统标记可随时更改此属性。

“模拟”: 此选项可将设备置于模拟模式。在此模式下, 驱动程序不会尝试与物理设备进行通信, 但服务器将继续返回有效的 OPC 数据。模拟停止与设备的物理通信, 但允许 OPC 数据作为有效数据返回到 OPC 客户端。在“模拟模式”下, 服务器将所有设备数据处理为反射型: 无论向模拟设备写入什么内容, 都会读取回来, 而且会单独处理每个 OPC 项。项的内存映射取决于组更新速率。如果服务器移除了项 (如服务器重新初始化时), 则不保存数据。默认值为“否”。

注意:

1. “系统”标记 (_Simulated) 为只读且无法写入, 从而达到运行时保护的目。 “系统”标记允许从客户端监控此属性。
2. 在“模拟”模式下, 项的内存映射取决于客户端更新速率 (OPC 客户端的“组更新速率”或本机和 DDE 接口的扫描速率)。这意味着, 参考相同项、而采用不同更新速率的两个客户端会返回不同的数据。

“模拟模式”仅用于测试和模拟目的。该模式永远不能用于生产环境。

设备属性 - 扫描模式

“扫描模式”为需要设备通信的标记指定订阅客户端请求的扫描速率。同步和异步设备的读取和写入会尽快处理; 不受“扫描模式”属性的影响。

属性组	扫描模式	
常规	扫描模式	遵循客户端指定的扫描速率
扫描模式	来自缓存的初始更新	禁用
定时		

“扫描模式”: 为发送到订阅客户端的更新指定在设备中扫描标记的方式。选项说明如下:

- **“遵循客户端指定的扫描速率”:** 此模式可使用客户端请求的扫描速率。
- **“不超过扫描速率请求数据”:** 此模式可将该数值集指定为最大扫描速率。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
 - **注意:** 当服务器有活动的客户端和设备项且扫描速率值有所提高时, 更改会立即生效。当扫描速率值减小时, 只有所有客户端应用程序都断开连接, 更改才会生效。
- **“以扫描速率请求所有数据”:** 此模式将以订阅客户端的指定速率强制扫描标记。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
- **“不扫描, 仅按需求轮询”:** 此模式不会定期轮询属于设备的标签, 也不会在一个项变为活动状态后为获得项的初始值而执行读取操作。客户端负责轮询以便更新, 方法为写入 _DemandPoll 标记或为各项发出显式设备读取。有关详细信息, 请参阅服务器帮助中的“设备需求轮询”。
- **“遵循标签指定的扫描速率”:** 此模式将以静态配置标记属性中指定的速率强制扫描静态标记。以客户端指定的扫描速率扫描动态标记。

“来自缓存的初始更新”: 启用后, 此选项允许服务器为存储 (缓存) 数据的新激活标签参考提供第一批更新。只有新项参考共用相同的地址、扫描速率、数据类型、客户端访问和缩放属性时, 才能提供缓存更新。设备读取仅用于第一个客户端参考的初始更新。默认设置为禁用; 只要客户端激活标记参考, 服务器就会尝试从设备读取初始值。

设备属性 - 定时

设备的“定时”属性允许调整驱动程序对错误条件的响应，以满足应用程序的需要。在很多情况下，需要更改环境的此类属性，以便获得最佳性能。由电气原因产生的噪音、调制解调器延迟以及较差的物理连接等因素都会影响通信驱动程序遇到的错误数或超时次数。“定时”属性特定于每个配置的设备。

属性组	<input type="checkbox"/> 通信超时	
常规	连接超时 (秒)	3
扫描模式	请求超时 (毫秒)	1000
定时	重试次数	3
自动降级	<input type="checkbox"/> 定时	
冗余	请求间延迟 (毫秒)	0

通信超时

“连接超时”(Connect Timeout): 此属性 (主要由基于驱动程序的以太网使用) 控制建立远程设备套接字连接所需的时间长度。设备的连接时间通常比针对同一设备的正常通信请求所花费时间更长。有效范围为 1 到 30 秒。默认值通常为 3 秒钟，但可能会因驱动程序的具体性质而异。如果驱动程序不支持此设置，则此设置将被禁用。

● **注意:** 鉴于 UDP 连接的性质，当通过 UDP 进行通信时，连接超时设置不适用。

“请求超时”(Request Timeout): 此属性可指定一个所有驱动程序使用的间隔来决定驱动程序等待目标设备完成响应的的时间。有效范围是 50 至 9,999,999 毫秒 (167.6667 分钟)。默认值通常是 1000 毫秒，但可能会因驱动程序而异。大多数串行驱动程序的默认超时是基于 9600 波特或更高的波特率来确定的。当以较低的波特率使用驱动程序时，请增加超时，以补偿获取数据所需增加的时间。

“超时前的尝试次数”: 此属性用于指定在认定请求失败以及设备出错之前，驱动程序发出通信请求的次数。有效范围为 1 到 10。默认值通常是 3，但可能会因驱动程序的具体性质而异。为应用程序配置的尝试次数很大程度上取决于通信环境。此属性适用于连接尝试和请求尝试。

定时

“请求间延迟”(Inter-Request Delay): 此属性指定驱动程序在将下一个请求发送到目标设备之前等待的时间。它会覆盖设备关联标记的一般轮询频率，以及一次性读取和写入次数。在处理周转时间慢的设备时，以及担心网络负载问题时，这种延迟很有用。为设备配置延迟会影响与通道上所有其他设备的通信。建议用户尽可能将所有需要请求间延迟的设备隔离至单独的通道。其他通信属性 (如通信序列化) 可以延长此延迟。有效范围是 0 至 300,000 毫秒；但是，某些驱动程序可能因某项特别设计的功能而限制最大值。默认值为 0，它表示对目标设备的请求之间没有延迟。

● **注意:** 不是所有的驱动程序都支持“请求间延迟”。如果不可用，则此设置不会出现。

设备属性 - 自动降级

自动降级属性可以在设备未响应的情况下使设备暂时处于关闭扫描状态。通过将特定时间段内无响应的设备脱机，驱动程序可以继续优化与同一通道上其他设备的通信。该时间段结束后，驱动程序将重新尝试与无响应设备进行通信。如果设备响应，则该设备会进入开启扫描状态；否则，设备将再次开始其关闭扫描时间段。

属性组	<input type="checkbox"/> 自动降级	
常规	故障时降级	启用
扫描模式	降级超时	3
定时	降级期间 (毫秒)	10000
自动降级	降级时放弃请求	禁用
标记生成		

“故障时降级”: 启用后，将自动对设备取消扫描，直到该设备再次响应。

● **提示:** 使用 _AutoDemoted 系统标记来监视设备的降级状态，确定何时对设备取消扫描。

“降级超时”: 指定在对设备取消扫描之前，请求超时和重试的连续周期数。有效范围是 1 到 30 次连续失败。默认值为 3。

“降级期间”: 指示当达到超时值时, 对设备取消扫描多长时间。在此期间, 读取请求不会被发送到设备, 与读取请求关联的所有数据都被设置为不良质量。当此期间到期时, 驱动程序将对设备进行扫描, 并允许进行通信尝试。有效范围为 100 至 3600000 毫秒。默认值为 10000 毫秒。

“降级时放弃请求”: 选择是否在取消扫描期间尝试写入请求。如果禁用, 则无论是否处于降级期间都始终发送写入请求。如果启用, 则放弃写入; 服务器自动将接收自客户端的写入请求视为失败, 且不会在事件日志中记录消息。

设备属性 - 标记生成

自动标记数据库生成功能使设置应用程序成为一项即插即用操作。选择可以配置为自动构建标记列表的通信驱动程序 (标记与特定于设备的数据相对应)。可以从客户端浏览这些自动生成的标记 (这取决于支持驱动程序的性质)。

并非所有设备和驱动程序都支持全自动标记数据库生成, 也并非所有都支持相同的数据类型。有关具体信息, 请参阅各驱动程序的数据类型说明或支持的数据类型列表。

如果目标设备支持其自身的本地标记数据库, 则驱动程序会读取设备的标记信息, 并使用该数据来在服务器中生成标记。如果该设备本身不支持已命名的标记, 则驱动程序会根据特定于驱动程序的信息来创建标记列表。这两个条件的示例如下:

1. 如果数据采集系统支持其自身的本地标记数据库, 则通信驱动程序将使用在设备中发现的标记名称来构建服务器的标记。
2. 如果以太网 I/O 系统支持其自身可用 I/O 模块类型的检测, 则通信驱动程序会基于插入以太网 I/O 机架的 I/O 模块类型在服务器中自动生成标记。

注意: 自动标记数据库生成的操作模式可进行完全配置。有关详细信息, 请参阅下方的属性说明。

属性组	<input type="checkbox"/> 标记生成	
常规	设备启动时	启动时不生成
扫描模式	对于重复标记	创建时删除
定时	父组	
自动降级	允许自动生成的子组	启用
标记生成	创建	创建标记
冗余		

“属性更改时”: 如果设备支持在特定属性更改时自动生成标记, 系统会显示“属性更改时”选项。默认情况下, 该选项设置为“是”, 但可以将其设置为“否”来控制何时生成标记。在此情况下, 必须手动执行“创建标记”操作以执行标记生成。

“设备启动时”: 此属性指定何时自动生成 OPC 标记。选项说明如下:

- **“启动时不生成”**: 此选项可防止驱动程序向服务器的标记空间添加任何 OPC 标记。这是默认设置。
- **“始终在启动时生成”**: 此选项可使驱动程序评估设备, 以便获得标记信息。每次启动服务器时, 它还会向服务器的标记空间添加标记。
- **“首次启动时生成”**: 此选项可使驱动程序在首次运行项目时评估目标设备, 以便获得标记信息。它还可以根据需要向服务器标记空间添加任何 OPC 标记。

注意: 如果选择自动生成 OPC 标记的选项, 添加到服务器标记空间的任何标记都必须随项目保存。用户可以在“工具”|“选项”菜单中将项目配置为自动保存。

“对于重复标记”: 启用自动标记数据库生成后, 服务器需要了解如何处理先前已添加的标记, 或在初始创建通信驱动程序后添加或修改的标记。此设置可控制服务器处理自动生成的以及当前存在于项目中的 OPC 标记的方式。它还可以防止自动生成的标记在服务器中累积。

例如, 如果用户更改机架中的 I/O 模块, 并且服务器配置为“始终在启动时生成”, 则每当通信驱动程序检测到新的 I/O 模块时, 新标记就会添加到服务器。如果未移除旧标记, 则许多未使用的标记可能会在服务器的标记空间中累积。选项包括:

- **“创建时删除”**: 此选项可在添加任何新标记之前, 将先前添加到标记空间的任何标记删除。这是默认设置。
- **“根据需要覆盖”**: 此选项可以指示服务器仅移除通信驱动程序要用新标记替换掉的标记。所有未被覆盖的标记仍将保留在服务器的标记空间中。
- **“不覆盖”**: 此选项可以防止服务器移除任何之前生成的标记或服务器中已存在的标记。通信驱动程序只能添加全新的标记。
- **“不覆盖, 记录错误”**: 此选项与前一选项有相同效果, 并且在发生标记覆盖时, 也会将错误消息发布到服务器的事件日志。

● **注意**: 删除 OPC 标记会影响通信驱动程序已自动生成的标记以及使用匹配已生成标记的名称添加的任何标记。如果标记所使用的名称可能与驱动程序自动生成的标记相匹配, 则用户应避免将此类标记添加到服务器。

“父组”: 此属性通过指定将要用于自动生成标记的组, 来防止自动生成的标记与已手动输入的标记发生混淆。组名称最多可包含 256 个字符。此父组具有一个根分支, 可将所有自动生成的标记添加到其中。

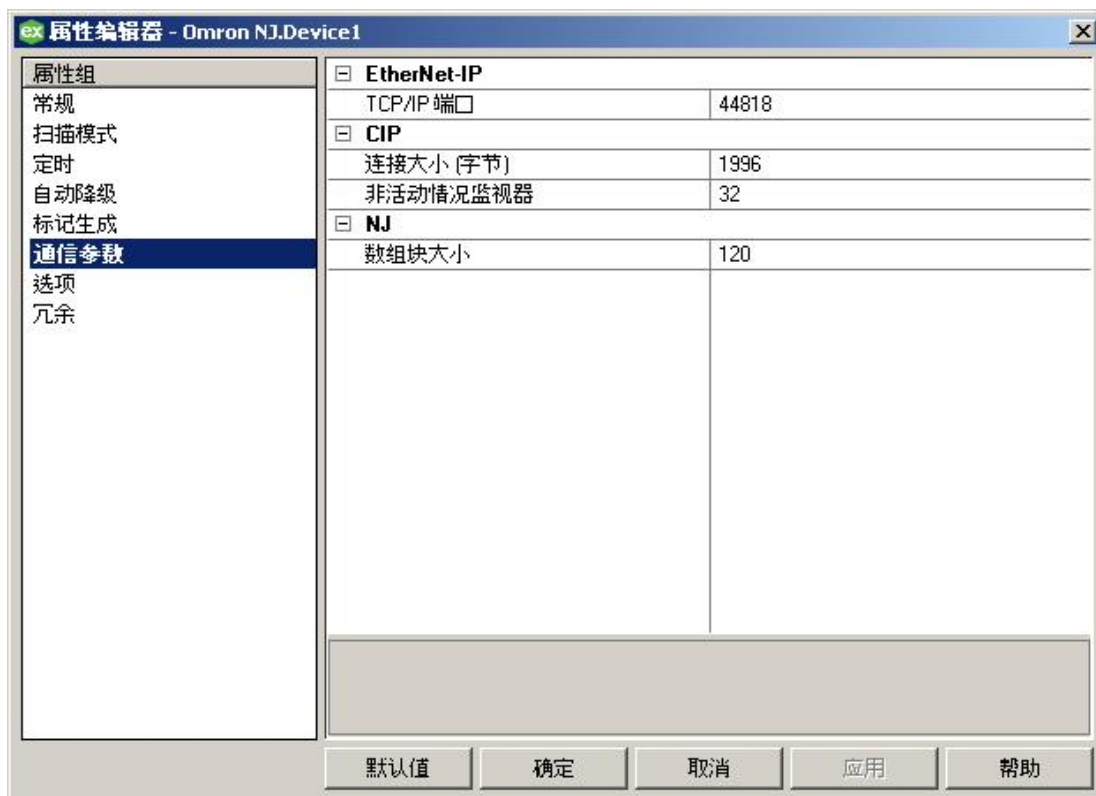
“允许自动生成的子组”: 此属性用于控制服务器是否为自动生成的标记自动创建子组。这是默认设置。如果禁用, 则服务器会在没有任何分组的简单列表中生成设备标记。在服务器项目中, 生成的标记使用地址值命名。例如, 生成过程中不会保留标记名称。

● **注意**: 如果在服务器生成标记的过程中, 分配给标记的名称与现有标记的名称相同, 则系统会自动递增到下一个最高数字, 以免标记名称发生重复。例如, 如果生成过程中创建了名为 "AI22" 的标记且该名称已存在, 则会将标记创建为 "AI23"。

“创建”: 开始创建自动生成的 OPC 标记。如果已修改设备的配置, 则**“创建标记”**可强制驱动程序重新评估设备以发现可能的标记更改。由于该选项可以通过系统标记进行访问, 这使得客户端应用程序能够启动标记数据库创建。

● **注意**: 当“配置”对项目进行离线编辑时, 会禁用**“创建标记”**。

设备属性 - 通信参数



EtherNet-IP

“TCP/IP 端口”(TCP/IP Port): 指定对设备进行配置后, 该设备所使用的 TCP/IP 端口号。默认设置为 44818。

CIP

“连接大小”(Connection Size): 指定 CIP 连接上可用于数据请求和响应的字节数。有效范围为 500 到 1996 字节。默认设置为 1996 字节。

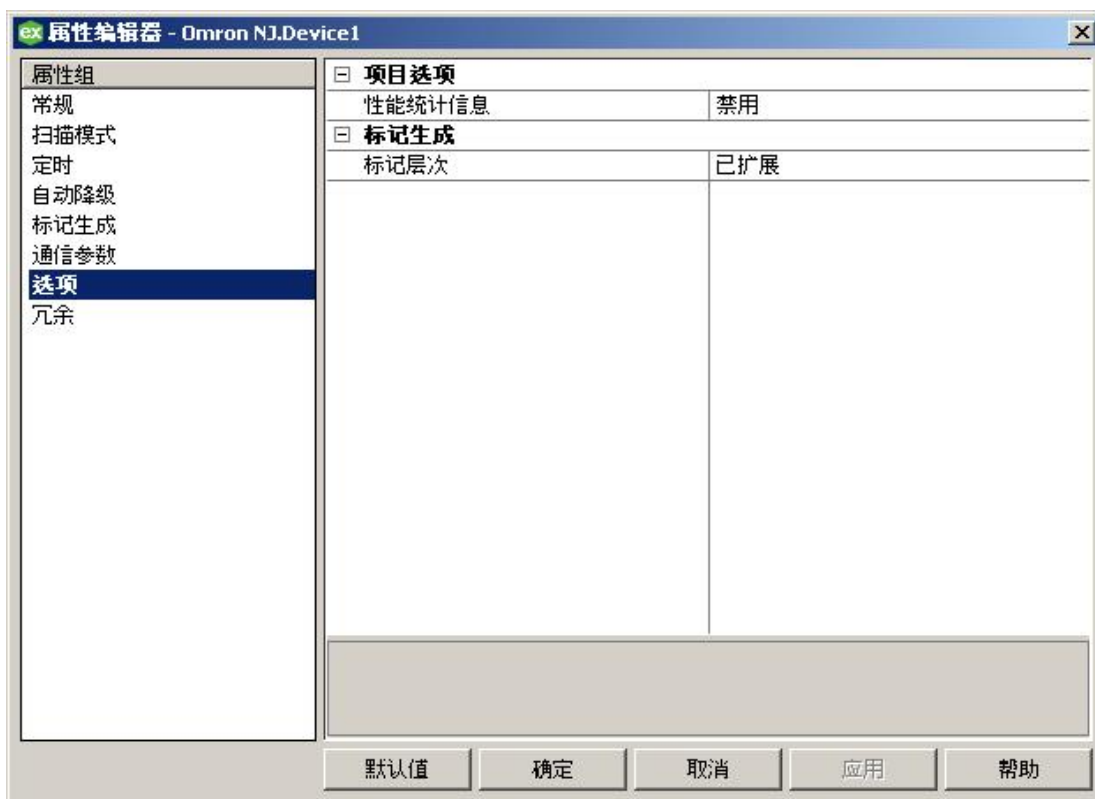
● **提示:** “连接大小”值可通过系统标记 `_CIPConnectionSizeRequested` 来请求。

“非活动情况监视器”(Inactivity Watchdog): 指定连接在由控制器关闭之前保持空闲状态 (不进行读/写事务处理) 的时长。通常情况下, 监视器值越大, 控制器释放连接资源所需的时间越长。默认设置为 32 字节。

NJ

“数组块大小”(Array Block Size): 指定在单个事务处理中要读取的数组元素的最大数量。值范围介于 30 到 3840 个元素之间。默认设置为 120 个元素。

设备属性 - 选项



项目 选项

“性能统计信息”: 选择此选项可收集通信统计数据, 以分析驱动程序的性能。启用时, 驱动程序可跟踪客户端-服务器标记更新的数量和类型。重新启动服务器应用程序后, 结果会显示在“事件日志”中。为了能够正常操作, 默认设置为处于禁用状态。

● **注意:**

1. 在对项目进行配置以获得最佳性能后, 禁用“性能统计信息”以获得最佳性能。
2. 统计信息在关闭时被发送到“事件日志”, 因此, 必须重新启动服务器才能查看结果。

● 有关性能和诊断的详细信息, 请参阅 [性能统计信息和调优](#)。

“标记生成”

“标记层次”: 选择标记层次的显示方式。默认设置为“已扩展”。

- **“已压缩”**在此模式下，由自动标记生成创建的服务器标记遵循符合标记地址的组/标记层次。会为前面带有句点的每个段创建组。
- **“已扩展”**在“已扩展”模式下，会为前面带有句点的每个段创建标记组 (和“已压缩”模式一样)，但还会为数组标记创建组。这是默认设置。

● 有关如何创建组的详细信息，请参阅[标记层次](#)。

● **提示**: 要使用此功能，请启用“允许子组”。

设备属性 - 冗余

属性组 常规 扫描模式 定时 自动降级 冗余	冗余	
	次级路径	...
	操作模式	故障切换
	监视器项目	
	监视器间隔 (秒)	300
	尽快返回至主要设备	是

Media-Level Redundancy 插件提供冗余。

● 有关详细信息，请参阅[网站](#)、[向销售代表咨询](#)或[查阅用户手册](#)。

优化通信

与任何可编程控制器一样，可使用多种方法来增强性能和系统通信。

“连接大小”

提高“连接大小”可以增加每个数据包的读取/写入请求数量，从而提高吞吐量。尽管它也会增加 CPU 载荷和响应周转时间，但是可显著提高性能。有关详细信息，请参阅[通信参数](#)。

多请求数据包

Omron NJ Ethernet 驱动程序旨在通过在单个事务中包括多个请求来对读取和写入进行优化。与单个标记事务处理相比，这可显著提高性能。唯一的限制是需要调整单个事务处理的数据字节数。

● 由于读取和写入请求可指定 ASCII 格式的变量地址，因此，用户应使变量名称包含的字符数保持最少。变量名称包含的字符数越少，适合单个事务的标记就越多，并且用于处理所有标记的事务越少。

对数组元素进行分块

要优化基本数组元素的读取，请在单个请求中以块的形式读取数组，而不是单独读取。每个块中读取的元素越多，性能越好。由于事务处理费用较高且处理会消耗大多数时间，因此，在尽可能多地扫描所需标记的同时要尽可能少地进行事务处理。此为数组元素块的实质。

规定块大小为元素计数。块大小为 120 个元素，意味着在一个请求中最多能够读取 120 个数组元素。最大块大小为 3840 个元素，意味着在一个请求中最多能够读取 3840 个数组元素。

正如[通信参数](#)中所述，块大小是可调整的，且应该根据当前项目进行选择。例如，如果数组元素 0 到 26 和元素 3839 是要读取的标记，那么，使用 3840 的块大小不仅过剩，而且影响驱动程序的性能。这是因为：尽管只需要 28 个元素，但是会针对每个请求读取 0 和 3839 之间的所有元素。在这种情况下，块大小设为 30 更合适。在一个请求中读取元素 0 到 26，而在下一个请求中读取元素 3839。

字符串

建议将字符串变量长度定义为最小值，从而使其更好地发挥作用。在 Sysmac Studio 中，默认情况下，字符串变量的长度定义为 256 个字符。读取字符串长度较大的这些字符串变量时，需要额外设备通信，而且可能会影响性能。

通道属性 - 常规

此服务器支持同时使用多个通信驱动程序。服务器项目中使用的各个协议或驱动程序称为通道。服务器项目可以由具有相同通信驱动程序或具有唯一通信驱动程序的多个通道组成。通道充当 OPC 链路的基础构建块。此组用于指定常规通道属性，如标识属性和操作模式。

属性组	<input type="checkbox"/> 标识 <input checked="" type="checkbox"/> 名称 <input type="checkbox"/> 说明 <input type="checkbox"/> 驱动程序 <input checked="" type="checkbox"/> 诊断 <input type="checkbox"/> 诊断数据捕获	
常规	名称	通道 1
写优化	说明	
高级	驱动程序	Simulator
持久存储	诊断	
	诊断数据捕获	禁用

标识

“名称”：此通道的用户定义标识。在每个服务器项目中，每个通道名称都必须是唯一的。尽管名称最多可包含 256 个字符，但在浏览 OPC 服务器的标记空间时，一些客户端应用程序的显示窗口可能不够大。通道名称是 OPC 浏览器信息的一部分。该属性是创建通道所必需的。

● 有关保留字符的信息，请参阅服务器帮助中的“如何正确命名通道、设备、标记和标记组”。

“说明”：有关此通道的用户定义信息。

● 这些属性 (包括 Description) 当中有很多具有关联的系统标记。

“驱动程序”(Driver)：为该通道选择的协议/驱动程序。该属性指定在通道创建期间选择的设备驱动程序。它在通道属性中为禁用设置。该属性是创建通道所必需的。

● **注意:** 服务器全天在线运行时,可以随时更改这些属性。其中包括更改通道名称以防止客户端向服务器注册数据。如果客户端在通道名称更改之前已从服务器中获取了项,那么这些项不会受到任何影响。如果客户端应用程序在通道名称更改之后发布项,并尝试通过原来的通道名称重新获取项,则该项将不被接受。考虑到这一点,一旦开发完成大型客户端应用程序,就不应对属性进行任何更改。利用“用户管理器”可防止操作员更改属性并限制对服务器功能的访问权限。

诊断

“**诊断数据捕获**”: 启用此选项后,通道的诊断信息即可用于 OPC 应用程序。由于服务器的诊断功能所需的开销处理量最少,因此建议在需要时使用这些功能,而在不需要时禁用这些功能。默认设置为禁用状态。

● **注意:** 如果驱动程序不支持诊断,则该属性不可用。

● 有关详细信息,请参阅服务器帮助中的“通信诊断”。

通道属性 - 以太网通信

以太网通信可用于与设备进行通信。

属性组	以太网设置	
常规	网络适配器	默认值
以太网通信		
写优化		
高级		
通信序列化		

以太网设置

“**网络适配器**”(Network Adapter): 指定要绑定的网络适配器。如果留空或选择“默认”,则操作系统将选择默认适配器。

通道属性 - 写入优化

与任何服务器一样,将数据写入设备可能是应用程序应具备的最重要的功能。服务器旨在确保从客户端应用程序写入的数据能够准时发送到设备。为了达到此目标,服务器提供了可用来满足特定需求以提高应用程序响应能力的优化属性。

属性组	写优化	
常规	优化方法	仅写入所有标记的最新值
写优化	占空比	10
高级		
持久存储		

写入优化

“**优化方法**”: 控制如何将写入数据传递至底层通信驱动程序。选项包括:

- “**写入所有标记的所有值**”: 此选项可强制服务器尝试将每个值均写入控制器。在此模式下,服务器将持续收集写入请求并将它们添加到服务器的内部写入队列。服务器将对写入队列进行处理并尝试通过将数据尽快写入设备来将其清空。此模式可确保从客户端应用程序写入的所有数据均可发送至目标设备。如果写入操作顺序或写入项的内容必须且仅能显示于目标设备上,则应选择此模式。
- “**写入非布尔标记的最新值**”: 由于将数据实际发送至设备需要一段时间,因此对同一个值的多次连续写入会存留于写入队列中。如果服务器要更新已位于写入队列中的某个写入值,则需要大大减少写入操作才能获得相同的最终输出值。这样一来,便不会再有额外的写入数据存留于服务器队列中。几乎就在用户停止移动滑动开关时,设备中的值达到其正确值。根据此模式的规定,任何非布尔值都会在服务器的内部写入队列中更新,并在下一个可能的时机发送至设备。这可以大大提高应用性能。

- **注意**：该选项不会尝试优化布尔值的写入。它允许用户在不影响布尔运算的情况下优化 HMI 数据的操作，例如瞬时型按钮等。
- **“写入所有标记的最新值”**：该选项采用的是第二优化模式背后的理论并将其应用至所有标记。如果应用程序只需向设备发送最新值，则该选项尤为适用。此模式会通过当前写入队列中的标记发送前对其进行更新来优化所有的写入操作。此为默认模式。

“占空比”(Duty Cycle)：用于控制写操作与读操作的比率。该比率始终基于每一到十次写入操作对应一次读取操作。占空比的默认设置为 10，这意味着每次读取操作对应十次写入操作。即使在应用程序执行大量的连续写入操作时，也必须确保足够的读取数据处理时间。如果将占空比设置为 1，则每次读取操作对应一次写入操作。如果未执行任何写入操作，则会连续处理读取操作。相对于更加均衡的读写数据流而言，该特点使得应用程序的优化可通过连续的写入操作来实现。

- **注意**：建议在将应用程序投入生产环境前使其与写入优化增强功能相兼容。

通道属性 - 高级

此组用于指定高级通道属性。并非所有驱动程序都支持所有属性，因此不会针对不支持的设备显示“高级”组。

属性组	<input type="checkbox"/> 非规范浮点数处理	
常规	浮点值	替换为零
以太网通信	<input type="checkbox"/> 设备间延迟	
写优化	设备间延迟 (毫秒)	0
高级		
通信序列化		

“非规范浮点数处理”：非规范值定义为无穷大、非数字 (NaN) 或不正规编号。默认值为“替换为零”。具有原生浮点数处理功能的驱动程序可能会默认设置为“未修改”。通过非规范浮点数处理，用户可以指定驱动程序处理非规范 IEEE-754 浮点数据的方式。选项说明如下：

- **“替换为零”**：此选项允许驱动程序在将非规范 IEEE-754 浮点值传输到客户端之前，将其替换为零。
- **“未修改”**：此选项允许驱动程序向客户端传输 IEEE-754 不正规、规范、非数字和无穷大值，而不进行任何转换或更改。

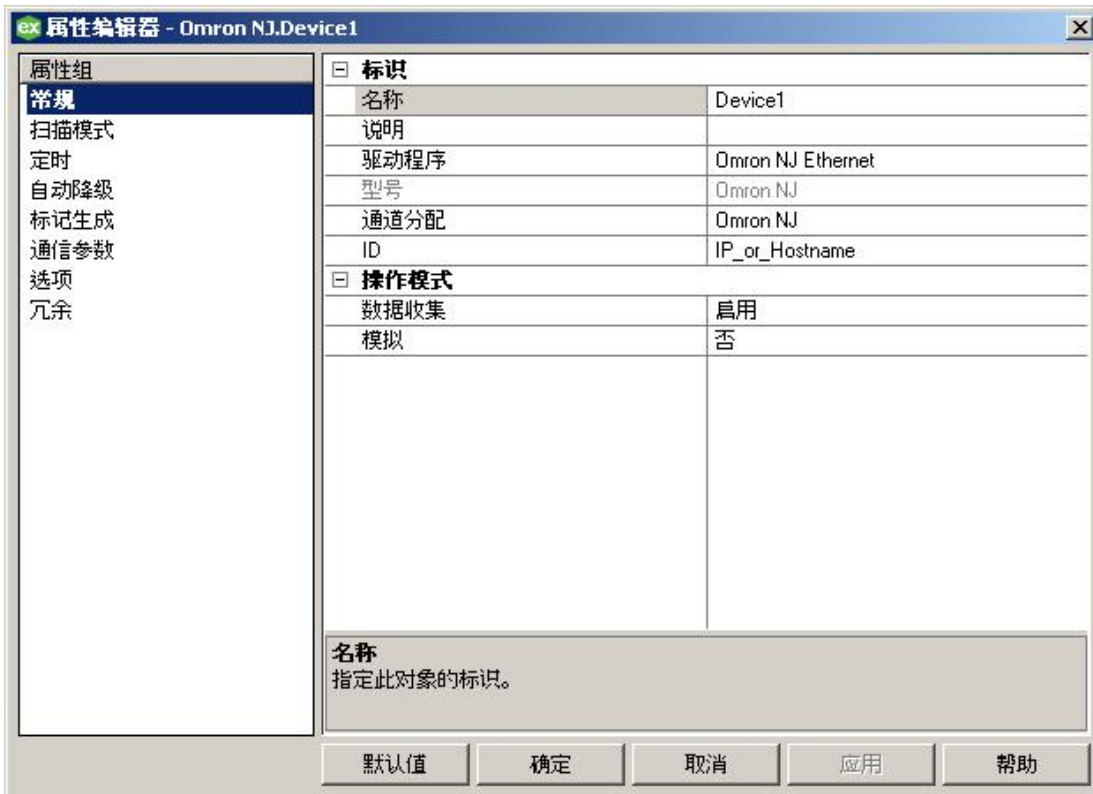
- **注意**：如果驱动程序不支持浮点值或仅支持所显示的选项，则此属性不可用。根据通道的浮点规范化设置，将仅对实时驱动程序标记 (如值和数组) 进行浮点规范化。例如，此设置不会影响 EFM 数据。

- 有关浮点值的详细信息，请参阅服务器帮助中的“如何使用非规范化浮点值”。

“设备间延迟”：指定在接收到同一通道上的当前设备发出的数据后，通信通道向下一设备发送新请求前等待的时间。设置为零 (0) 将禁用延迟。

- **注意**：此属性并不适用于所有驱动程序、型号和相关设置。

设备属性 - 常规



标识

“名称”: 此设备的用户定义标识。

“说明”: 有关此设备的用户定义信息。

“通道分配”: 该设备当前所属通道的用户定义名称。

驱动程序: 为该设备选择的协议驱动程序。

“型号”: 设备的特定版本。有关支持 *FINS* 通信服务的型号列表, 请参阅制造商网站。

ID: ID 用于指定到目标 NJ CPU 单元的路径。

- 将本地 NJ CPU 单元的设备 ID 指定为本地 CPU 单元的 IP 或主机名。该设备 ID 必须是点分四段形式的有效 IP 地址或主机名。例如, "192.168.1.100" 或 "NJ001"。
- 将远程 NJ CPU 单元的设备 ID 指定为本地 CPU 或 Ethernet/IP 单元的 IP 或主机名, 外加指向远程 CPU 单元的 CIP 连接路径 (也称为路由路径)。例如, "192.168.1.100\1#10\2\10.10.110.2"。

● 有关连接路径语法的信息, 请参阅 [连接路径规范](#)。

操作模式

“数据收集”: 此属性控制设备的活动状态。尽管默认情况下会启用设备通信, 但可使用此属性禁用物理设备。设备处于禁用状态时, 不会尝试进行通信。从客户端的角度来看, 数据将标记为无效, 且不接受写入操作。通过此属性或设备系统标记可随时更改此属性。

“模拟”: 此选项可将设备置于模拟模式。在此模式下, 驱动程序不会尝试与物理设备进行通信, 但服务器将继续返回有效的 OPC 数据。模拟停止与设备的物理通信, 但允许 OPC 数据作为有效数据返回到 OPC 客户端。在“模拟模式”下, 服务器将所有设备数据处理为反射型: 无论向模拟设备写入什么内容, 都会读取回来, 而且会单独处理每个 OPC 项。项的内存映射取决于组更新速率。如果服务器移除了项 (如服务器重新初始化时), 则不保存数据。默认值为“否”。

● **注意:**

1. “系统”标记 (`_Simulated`) 为只读且无法写入，从而达到运行时保护的目的。“系统”标记允许从客户端监控此属性。
2. 在“模拟”模式下，项的内存映射取决于客户端更新速率 (OPC 客户端的“组更新速率”或本机和 DDE 接口的扫描速率)。这意味着，参考相同项、而采用不同更新速率的两个客户端会返回不同的数据。

●“模拟模式”仅用于测试和模拟目的。该模式永远不能用于生产环境。

设备属性 - 扫描模式

“扫描模式”为需要设备通信的标记指定订阅客户端请求的扫描速率。同步和异步设备的读取和写入会尽快处理；不受“扫描模式”属性的影响。

属性组	<input type="checkbox"/> 扫描模式	
常规	扫描模式	遵循客户端指定的扫描速率
扫描模式	来自缓存的初始更新	禁用
定时		

“扫描模式”：为发送到订阅客户端的更新指定在设备中扫描标记的方式。选项说明如下：

- “遵循客户端指定的扫描速率”：此模式可使用客户端请求的扫描速率。
- “不超过扫描速率请求数据”：此模式可将该数值集指定为最大扫描速率。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
● 注意：当服务器有活动的客户端和设备项且扫描速率值有所提高时，更改会立即生效。当扫描速率值减小时，只有所有客户端应用程序都断开连接，更改才会生效。
- “以扫描速率请求所有数据”：此模式将以订阅客户端的指定速率强制扫描标记。有效范围为 10 至 99999990 毫秒。默认值为 1000 毫秒。
- “不扫描，仅按需求轮询”：此模式不会定期轮询属于设备的标签，也不会在一个项变为活动状态后为获得项的初始值而执行读取操作。客户端负责轮询以便更新，方法为写入 `_DemandPoll` 标记或为各项发出显式设备读取。有关详细信息，请参阅服务器帮助中的“设备需求轮询”。
- “遵循标签指定的扫描速率”：此模式将以静态配置标记属性中指定的速率强制扫描静态标记。以客户端指定的扫描速率扫描动态标记。

“来自缓存的初始更新”：启用后，此选项允许服务器为存储 (缓存) 数据的新激活标签参考提供第一批更新。只有新项参考共用相同的地址、扫描速率、数据类型、客户端访问和缩放属性时，才能提供缓存更新。设备读取仅用于第一个客户端参考的初始更新。默认设置为禁用；只要客户端激活标记参考，服务器就会尝试从设备读取初始值。

设备属性 - 定时

设备的“定时”属性允许调整驱动程序对错误条件的响应，以满足应用程序的需要。在很多情况下，需要更改环境的此类属性，以便获得最佳性能。由电气原因产生的噪音、调制解调器延迟以及较差的物理连接等因素都会影响通信驱动程序遇到的错误数或超时次数。“定时”属性特定于每个配置的设备。

属性组	<input type="checkbox"/> 通信超时	
常规	连接超时 (秒)	3
扫描模式	请求超时 (毫秒)	1000
定时	重试次数	3
自动降级	<input type="checkbox"/> 定时	
冗余	请求间延迟 (毫秒)	0

通信超时

“连接超时”(Connect Timeout)：此属性 (主要由基于驱动程序的以太网使用) 控制建立远程设备套接字连接所需的时间长度。设备的连接时间通常比针对同一设备的正常通信请求所花费时间更长。有效范围为 1 到 30 秒。默认值通常为 3 秒钟，但可能会因驱动程序的具体性质而异。如果驱动程序不支持此设置，则此设置将被禁用。

● 注意：鉴于 UDP 连接的性质，当通过 UDP 进行通信时，连接超时设置不适用。

“请求超时”(Request Timeout): 此属性可指定一个所有驱动程序使用的间隔来决定驱动程序等待目标设备完成响应的的时间。有效范围是 50 至 9,999,999 毫秒 (167.6667 分钟)。默认值通常是 1000 毫秒, 但可能会因驱动程序而异。大多数串行驱动程序的默认超时是基于 9600 波特或更高的波特率来确定的。当以较低的波特率使用驱动程序时, 请增加超时, 以补偿获取数据所需增加的时间。

“超时前的尝试次数”: 此属性用于指定在认定请求失败以及设备出错之前, 驱动程序发出通信请求的次数。有效范围为 1 到 10。默认值通常是 3, 但可能会因驱动程序的具体性质而异。为应用程序配置的尝试次数很大程度上取决于通信环境。此属性适用于连接尝试和请求尝试。

定时

“请求间延迟”(Inter-Request Delay): 此属性指定驱动程序在将下一个请求发送到目标设备之前等待的时间。它会覆盖设备关联标记的一般轮询频率, 以及一次性读取和写入次数。在处理周转时间慢的设备时, 以及担心网络负载问题时, 这种延迟很有用。为设备配置延迟会影响与通道上所有其他设备的通信。建议用户尽可能将所有需要请求间延迟的设备隔离至单独的通道。其他通信属性 (如通信序列化) 可以延长此延迟。有效范围是 0 至 300,000 毫秒; 但是, 某些驱动程序可能因某项特别设计的功能而限制最大值。默认值为 0, 它表示对目标设备的请求之间没有延迟。

● **注意:** 不是所有的驱动程序都支持“请求间延迟”。如果不可用, 则此设置不会出现。

设备属性 - 自动降级

自动降级属性可以在设备未响应的情况下使设备暂时处于关闭扫描状态。通过将特定时间段内无响应的设备脱机, 驱动程序可以继续优化与同一通道上其他设备的通信。该时间段结束后, 驱动程序将重新尝试与无响应设备进行通信。如果设备响应, 则该设备会进入开启扫描状态; 否则, 设备将再次开始其关闭扫描时间段。

属性组	☐ 自动降级	
常规	故障时降级	启用
扫描模式	降级超时	3
定时	降级期间 (毫秒)	10000
自动降级	降级时放弃请求	禁用
标记生成		

“故障时降级”: 启用后, 将自动对设备取消扫描, 直到该设备再次响应。

● **提示:** 使用 _AutoDemoted 系统标记来监视设备的降级状态, 确定何时对设备取消扫描。

“降级超时”: 指定在对设备取消扫描之前, 请求超时和重试的连续周期数。有效范围是 1 到 30 次连续失败。默认值为 3。

“降级期间”: 指示当达到超时值时, 对设备取消扫描多长时间。在此期间, 读取请求不会被发送到设备, 与读取请求关联的所有数据都被设置为不良质量。当此期间到期时, 驱动程序将对设备进行扫描, 并允许进行通信尝试。有效范围为 100 至 3600000 毫秒。默认值为 10000 毫秒。

“降级时放弃请求”: 选择是否在取消扫描期间尝试写入请求。如果禁用, 则无论是否处于降级期间都始终发送写入请求。如果启用, 则放弃写入; 服务器自动将接收自客户端的写入请求视为失败, 且不会在事件日志中记录消息。

设备属性 - 标记生成

自动标记数据库生成功能使设置应用程序成为一项即插即用操作。选择可以配置为自动构建标记列表的通信驱动程序 (标记与特定于设备的数据相对应)。可以从客户端浏览这些自动生成的标记 (这取决于支持驱动程序的性质)。

● **并非所有设备和驱动程序都支持全自动标记数据库生成, 也并非所有都支持相同的数据类型。有关具体信息, 请参阅各驱动程序的数据类型说明或支持的数据类型列表。**

如果目标设备支持其自身的本地标记数据库, 则驱动程序会读取设备的标记信息, 并使用该数据来在服务器中生成标记。如果该设备本身不支持已命名的标记, 则驱动程序会根据特定于驱动程序的信息来创建标记列表。这两个条件的示例如下:

1. 如果数据采集系统支持其自身的本地标记数据库，则通信驱动程序将使用在设备中发现的标记名称来构建服务器的标记。
2. 如果以太网 I/O 系统支持其自身可用 I/O 模块类型的检测，则通信驱动程序会基于插入以太网 I/O 机架的 I/O 模块类型在服务器中自动生成标记。

● **注意:** 自动标记数据库生成的操作模式可进行完全配置。有关详细信息，请参阅下方的属性说明。

属性组	<input type="checkbox"/> 标记生成	
常规	设备启动时	启动时不生成
扫描模式	对于重复标记	创建时删除
定时	父组	
自动降级	允许自动生成的子组	启用
标记生成	创建	创建标记
冗余		

“属性更改时”: 如果设备支持在特定属性更改时自动生成标记，系统会显示“属性更改时”选项。默认情况下，该选项设置为“是”，但可以将其设置为“否”来控制何时生成标记。在此情况下，必须手动执行“创建标记”操作以执行标记生成。

“设备启动时”: 此属性指定何时自动生成 OPC 标记。选项说明如下：

- **“启动时不生成”:** 此选项可防止驱动程序向服务器的标记空间添加任何 OPC 标记。这是默认设置。
- **“始终在启动时生成”:** 此选项可使驱动程序评估设备，以便获得标记信息。每次启动服务器时，它还会向服务器的标记空间添加标记。
- **“首次启动时生成”:** 此选项可使驱动程序在首次运行项目时评估目标设备，以便获得标记信息。它还可以根据需要对服务器标记空间添加任何 OPC 标记。

● **注意:** 如果选择自动生成 OPC 标记的选项，添加到服务器标记空间的任何标记都必须随项目保存。用户可以在“工具”|“选项”菜单中将项目配置为自动保存。

“对于重复标记”: 启用自动标记数据库生成后，服务器需要了解如何处理先前已添加的标记，或在初始创建通信驱动程序后添加或修改的标记。此设置可控制服务器处理自动生成的以及当前存在于项目中的 OPC 标记的方式。它还可以防止自动生成的标记在服务器中累积。

例如，如果用户更改机架中的 I/O 模块，并且服务器配置为“始终在启动时生成”，则每当通信驱动程序检测到新的 I/O 模块时，新标记就会添加到服务器。如果未移除旧标记，则许多未使用的标记可能会在服务器的标记空间中累积。选项包括：

- **“创建时删除”:** 此选项可在添加任何新标记之前，将先前添加到标记空间的任何标记删除。这是默认设置。
- **“根据需要覆盖”:** 此选项可以指示服务器仅移除通信驱动程序要用新标记替换掉的标记。所有未被覆盖的标记仍将保留在服务器的标记空间中。
- **“不覆盖”:** 此选项可以防止服务器移除任何之前生成的标记或服务器中已存在的标记。通信驱动程序只能添加全新的标记。
- **“不覆盖，记录错误”:** 此选项与前一选项有相同效果，并且在发生标记覆盖时，也会将错误消息发布到服务器的事件日志。

● **注意:** 删除 OPC 标记会影响通信驱动程序已自动生成的标记以及使用匹配已生成标记的名称添加的任何标记。如果标记所使用的名称可能与驱动程序自动生成的标记相匹配，则用户应避免将此类标记添加到服务器。

“父组”: 此属性通过指定将要用于自动生成标记的组，来防止自动生成的标记与已手动输入的标记发生混淆。组名称最多可包含 256 个字符。此父组具有一个根分支，可将所有自动生成的标记添加到其中。

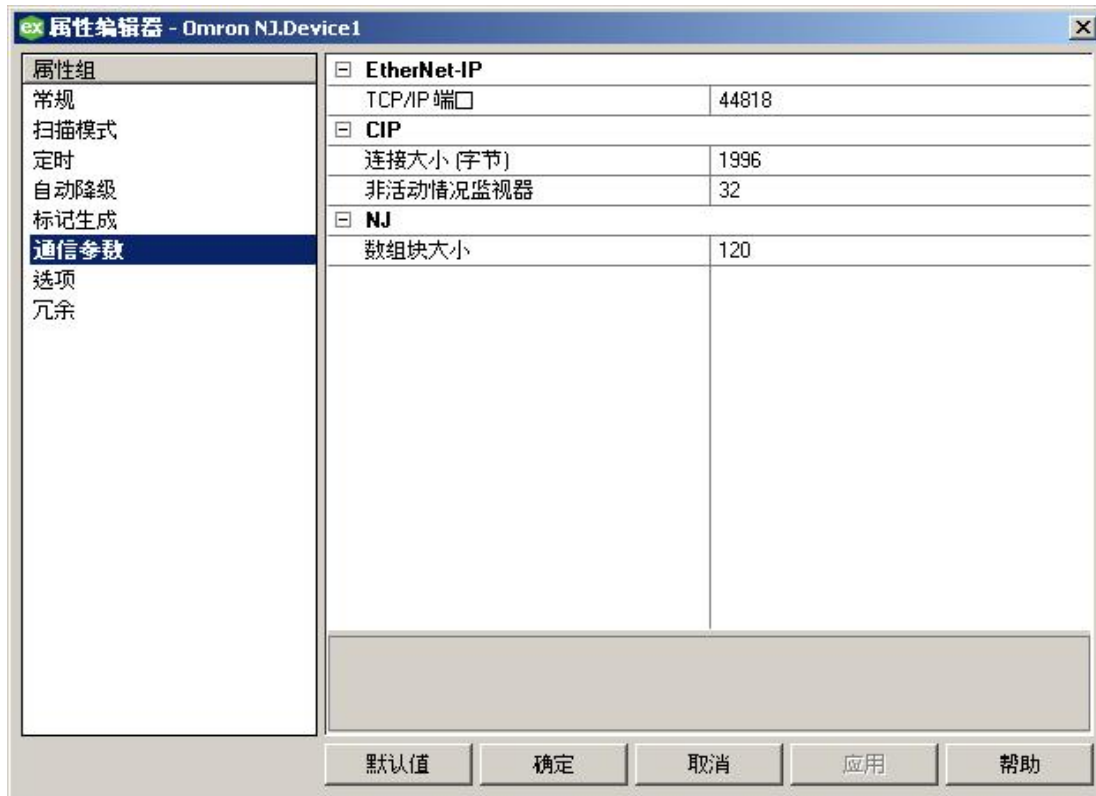
“允许自动生成的子组”: 此属性用于控制服务器是否为自动生成的标记自动创建子组。这是默认设置。如果禁用，则服务器会在没有任何分组的简单列表中生成设备标记。在服务器项目中，生成的标记使用地址值命名。例如，生成过程中不会保留标记名称。

● **注意:** 如果在服务器生成标记的过程中, 分配给标记的名称与现有标记的名称相同, 则系统会自动递增到下一个最高数字, 以免标记名称发生重复。例如, 如果生成过程中创建了名为 "AI22" 的标记且该名称已存在, 则会将标记创建为 "AI23"。

“创建”: 开始创建自动生成的 OPC 标记。如果已修改设备的配置, 则**“创建标记”**可强制驱动程序重新评估设备以发现可能的标记更改。由于该选项可以通过系统标记进行访问, 这使得客户端应用程序能够启动标记数据库创建。

● **注意:** 当“配置”对项目进行离线编辑时, 会禁用**“创建标记”**。

设备属性 - 通信参数



EtherNet-IP

“TCP/IP 端口”(TCP/IP Port): 指定对设备进行配置后, 该设备所使用的 TCP/IP 端口号。默认设置为 44818。

CIP

“连接大小”(Connection Size): 指定 CIP 连接上可用于数据请求和响应的字节数。有效范围为 500 到 1996 字节。默认设置为 1996 字节。

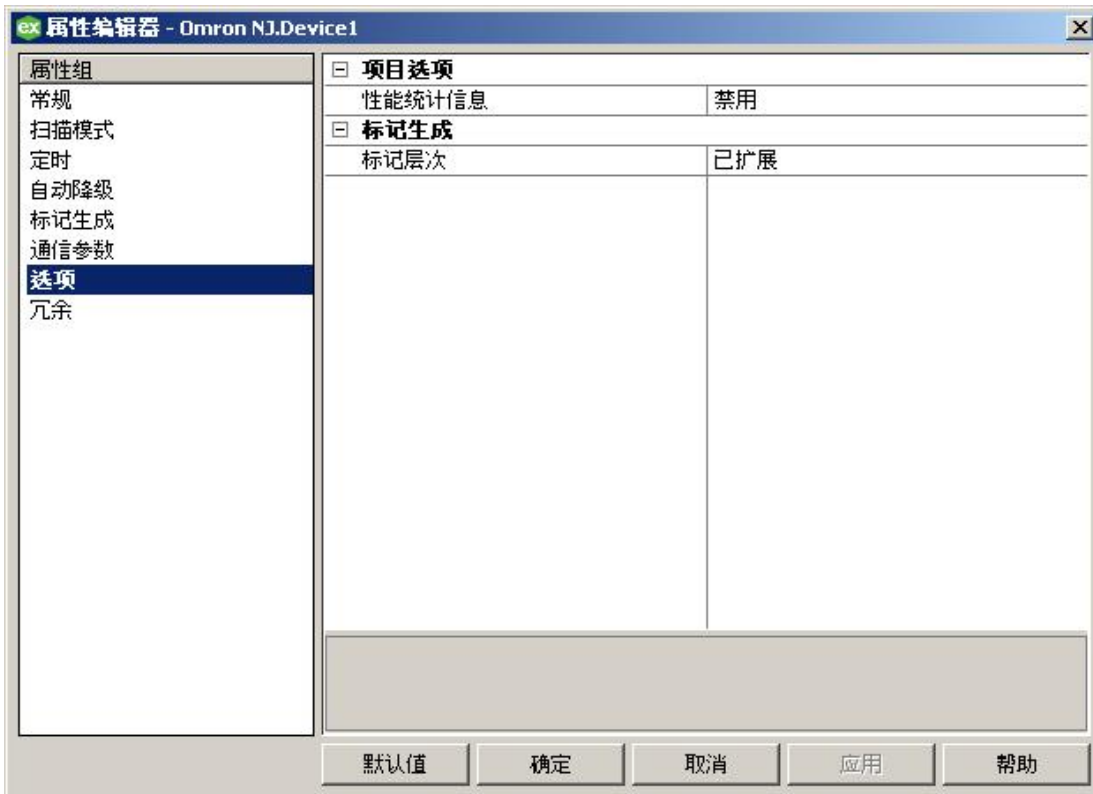
● **提示:** “连接大小”值可通过系统标记 `_CIPConnectionSizeRequested` 来请求。

“非活动情况监视器”(Inactivity Watchdog): 指定连接在由控制器关闭之前保持空闲状态 (不进行读/写事务处理) 的时长。通常情况下, 监视器值越大, 控制器释放连接资源所需的时间越长。默认设置为 32 字节。

NJ

“数组块大小”(Array Block Size): 指定在单个事务处理中要读取的数组元素的最大数量。值范围介于 30 到 3840 个元素之间。默认设置为 120 个元素。

设备属性 - 选项



项目选项

“性能统计信息”: 选择此选项可收集通信统计数据，以分析驱动程序的性能。启用时，驱动程序可跟踪客户端-服务器标记更新的数量和类型。重新启动服务器应用程序后，结果会显示在“事件日志”中。为了能够正常操作，默认设置为处于禁用状态。

● **注意：**

1. 在对项目进行配置以获得最佳性能后，禁用“性能统计信息”以获得最佳性能。
2. 统计信息在关闭时被发送到“事件日志”，因此，必须重新启动服务器才能查看结果。

● 有关性能和诊断的详细信息，请参阅[性能统计信息和调优](#)。

“标记生成”

“标记层次”: 选择标记层次的显示方式。默认设置为“已扩展”。

- **“已压缩”**在此模式下，由自动标记生成创建的服务器标记遵循符合标记地址的组/标记层次。会为前面带有句点的每个段创建组。
- **“已扩展”**在“已扩展”模式下，会为前面带有句点的每个段创建标记组（和“已压缩”模式一样），但还会为数组标记创建组。这是默认设置。

● 有关如何创建组的详细信息，请参阅[标记层次](#)。

● **提示：**要使用此功能，请启用“允许子组”。

设备属性 - 冗余

属性组	☐ 冗余	
常规	次级路径	...
扫描模式	操作模式	故障切换
定时	监视器项目	
自动降级	监视器间隔 (秒)	300
冗余	尽快返回至主要设备	是

Media-Level Redundancy 插件提供冗余。

● 有关详细信息，请参阅网站、向销售代表咨询或查阅用户手册。

优化通信

与任何可编程控制器一样，可使用多种方法来增强性能和系统通信。

“连接大小”

提高“连接大小”可以增加每个数据包的读取/写入请求数量，从而提高吞吐量。尽管它也会增加 CPU 载荷和响应周转时间，但是可显著提高性能。有关详细信息，请参阅[通信参数](#)。

多请求数据包

Omron NJ Ethernet 驱动程序旨在通过在单个事务中包括多个请求来对读取和写入进行优化。与单个标记事务处理相比，这可显著提高性能。唯一的限制是需要调整单个事务处理的数据字节数。

● 由于读取和写入请求可指定 ASCII 格式的变量地址，因此，用户应使变量名称包含的字符数保持最少。变量名称包含的字符数越少，适合单个事务的标记就越多，并且用于处理所有标记的事务越少。

对数组元素进行分块

要优化基本数组元素的读取，请在单个请求中以块的形式读取数组，而不是单独读取。每个块中读取的元素越多，性能越好。由于事务处理费用较高且处理会消耗大多数时间，因此，在尽可能多地扫描所需标记的同时要尽可能少地进行事务处理。此为数组元素块的实质。

规定块大小为元素计数。块大小为 120 个元素，意味着在一个请求中最多能够读取 120 个数组元素。最大块大小为 3840 个元素，意味着在一个请求中最多能够读取 3840 个数组元素。

正如[通信参数](#)中所述，块大小是可调整的，且应该根据当前项目进行选择。例如，如果数组元素 0 到 26 和元素 3839 是要读取的标记，那么，使用 3840 的块大小不仅过剩，而且影响驱动程序的性能。这是因为：尽管只需要 28 个元素，但是会针对每个请求读取 0 和 3839 之间的所有元素。在这种情况下，块大小设为 30 更合适。在一个请求中读取元素 0 到 26，而在下一个请求中读取元素 3839。

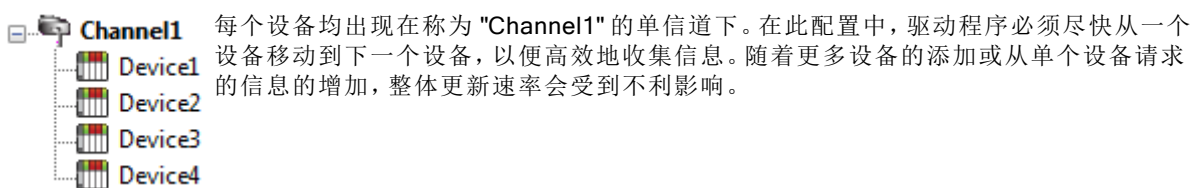
字符串

建议将字符串变量长度定义为最小值，从而使其更好地发挥作用。在 Sysmac Studio 中，默认情况下，字符串变量的长度定义为 256 个字符。读取字符串长度较大的这些字符串变量时，需要额外设备通信，而且可能会影响性能。

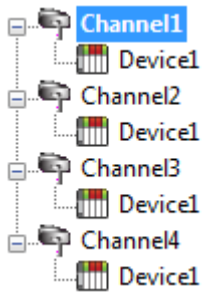
优化应用程序

Omron NJ Ethernet 驱动程序旨在提供最佳性能，使得其对系统的整体性能影响最小。即使 Omron NJ Ethernet 驱动程序速度很快，也可以利用一系列指南来优化应用程序，并获得最佳性能。

服务器将诸如 Omron NJ Ethernet 等通信协议称为信道。应用程序中定义的每个信道都表示服务器中一个单独的执行路径。一旦定义了通道，则必须在该通道下定义一系列设备。每一个此类设备都代表一个可从中收集数据的 Omron NJ CPU。虽然这种定义应用程序的方法提供了高水平的性能，但它不能充分利用 Omron NJ Ethernet 驱动程序或网络。下面显示了使用单个通道配置时应用程序所呈现效果的示例。



如果 Omron NJ Ethernet 驱动程序仅可以定义一个信道，则项目所需的所有设备必须在该信道下方创建；但是，驱动程序最多可以定义 256 个信道。使用多个通道，可通过同时向网络发出多个请求来分发数据集合工作载荷。下面显示了使用多个通道来提高性能时相同应用程序所呈现效果的示例。



当前，每个设备已在其自身的通道下定义。在这个新配置中，单个执行路径专用于从每个设备收集数据。如果应用程序的设备数小于等于 256 个，则可在其显示方式进行精确优化。

即使应用程序设备数大于 256 个，也可改善性能。虽然设备数小于等于 256 个时可能是理想情况，但附加信道仍会对应用程序有益。尽管在全部信道上分散设备载荷会使服务器再次从一个设备移动到另一个设备，但是，它现在可以用极少的设备在单个信道上进行处理。

性能统计信息和调优

“性能统计信息”功能提供了与 Omron NJ Ethernet 应用程序性能有关的基准和统计信息。由于其为处理的附加层，因此会影响服务器性能。在这种情况下，其默认处于禁用状态。要使用“性能统计信息”功能，请访问“设备属性”，然后展开“选项”组。对于“性能统计信息”，请选择“启用”。

性能类型 统计信息

性能统计信息提供了涵盖三个范围 (设备、信道和驱动程序) 的有意义的数值结果。类型说明如下：

- **设备：**这些统计信息提供了特定设备上的数据访问性能。
- **信道：**这些统计信息提供了启用性能统计信息的给定信道下所有设备的平均数据访问性能。
- **驱动程序：**这些统计信息提供了使用 Omron NJ Ethernet 驱动程序 且已启用性能统计信息的所有设备的平均数据访问性能。

选择统计信息类型

所需的统计信息类型取决于应用程序。通常情况下，驱动程序统计信息提供了应用程序性能的真实度量，而信道和设备统计信息在调优应用程序时最为相关。例如，将设备 A 中的 10 个特定标记移动到设备 B 中可能会提高设备 A 的性能。将设备 A 从信道 1 移动至信道 2 可能会提高信道 1 的性能。这些都是使用设备和信道统计信息的情况的很好的示例。

找到统计信息

服务器统计信息在关闭时输出到服务器“事件日志”。要查看结果，请关闭服务器并重新启动。

服务器统计信息和性能统计信息之间的差异

性能统计信息提供已执行的读取类型组合 (例如设备读取与缓存读取)，而服务器统计信息则提供一般读取计数值。

调优应用程序以提高性能

要提高设备和信道统计信息结果，请将变量名称长度保持最小并尽可能多地使用变量数组。有关详细信息，请参阅[优化通信](#)。

有关提高驱动程序统计信息结果的信息，请参阅下述说明。有关详细信息，请参阅[优化应用程序](#)。

1. 设备应分布在不同的信道上。除非有必要，否则不应将多个设备放在一个信道上。
2. 载荷应均匀分布在设备上。除非有必要，否则单个设备不应过载。
3. 不同设备之间不应引用相同的变量标记。

数据类型说明

数据类型	说明
布尔型	单个位
字节	无符号 8 位值
字符	有符号 8 位值
字	无符号 16 位值 位 0 是低位 位 15 是高位
短整型	有符号 16 位值 位 0 是低位 位 14 是高位 位 15 是符号位
长整型	有符号 32 位值
双字型	无符号 32 位值
浮点型	32 位 IEEE 浮点
双精度	64 位 IEEE 浮点
日期	64 位日期和时间值
字符串	空终止 Unicode 字符串
默认值	*

*当创建静态标记时，如果将数据类型指定为“默认”，则驱动程序会向控制器查询标记的数据类型，并将参考该静态标记的项的规范数据类型设置为查询结果。当创建动态标记时，如果未指定数据类型，则驱动程序会向控制器查询标记的数据类型，并将该动态标记的规范数据类型设置为查询结果。

地址说明

Omron NJ Ethernet 驱动程序使用称为“变量”的标记或基于符号的寻址结构。这些标记与常规 PLC 数据项的区别在于，标记名称本身是地址，而不是文件或寄存器编号。用户可以访问控制器的基本数据类型。尽管某些系统定义的类型为结构，但归根结底，它们基于这些基本数据类型。因此，结构的所有基本成员均可供访问。

Omron 数据类型	说明	数据类型	范围
BOOL	单个位值	布尔型	0、1
SINT	有符号 8 位值	字符	-128 到 127
无符号短整型	无符号 8 位值	字节	0 到 255
BYTE	位字符串 (8 位)	字节	0 到 255
INT	有符号 16 位值	短整型	-32768 到 32767
UINT	无符号 16 位值	字	0 到 65535
WORD	位字符串 (16 位)	字	0 到 65535
DINT	有符号 32 位值	长整型	-2147483648 到 2147483647
UDINT	无符号 32 位值	双字型	0 到 4294967295
DWORD	位字符串 (32 位)	双字型	0 到 4294967295
LINT	有符号 64 位值	双精度	-9223372036854775808 到 9223372036854775807
无符号长整型	无符号 64 位值	双精度	0 到 18446744073709551615
REAL	32 位 IEEE 浮点	浮点型	-3.402823e+38 到 -1.175495e-38 0 1.175495e-38 到 3.402823e+38
LREAL	64 位 IEEE 浮点	双精度	-1.79769313486231E+308 到 -2.22507385850721e-308 0 2.22507385850721E-308 到 1.79769313486231e+308
DATE AND TIME	无符号 64 位值	日期	日期/时间变量格式为: YYYY-MM-DDTHH:MM:SS.MS。支持的范围为 1970-01-01T00:00:00.000 到 2106-02-06T23:59:59.999。
STRING	字符串	字符串	字符串长度范围为 1 到 1985 个字符。这分别等同于 Sysmac Studio 中定义为 STRING[2] 和 STRING[1986] 的变量。额外的字符作为空终止符。

Omron 数据类型	说明	数据类型	范围
枚举	有符号 32 位值	长整型	-2147483648 到 2147483647*

*枚举的有效值实际上是指定范围内的值的子集。值的子集由 Omron NJ 设备中的枚举配置决定。

客户端/服务器 标记地址规则

变量名称与客户端/服务器标记地址对应。变量名称和客户端/服务器标记地址均遵循 IEC 61131-3 标识符规则。规则说明如下：

- 只能包含字母数字字符和下划线
- 每段最多包含 127 个字符
- 字符不区分大小写
- 忽略空格

客户端/服务器 标记名称规则

服务器中的标记名称分配与地址分配不同，因为名称不能以下划线开头。有关语法和示例，请参阅[地址格式](#)。

重要事项：如果标记地址过大，超出了协议限制的 511 个字节，则将由于“地址超出范围”错误而验证失败。如果发生这种情况，请减少标记地址中的字符数，直至通过验证。

数据类型 强制

Omron NJ Ethernet 驱动程序 可以将控制器中的某些 Omron 数据类型强制转换为多种服务器数据类型。例如，可以使用“字节”服务器数据类型在控制器中创建“短整型”变量的标记。有关驱动程序支持的所有 Omron 数据类型的受支持数据类型强制列表，请参见下表。

Omron 数据类型	数据类型
BOOL	布尔型
短整型、无符号短整型或字节	字符或字节
整型、无符号整型或字	短整型或字
双整型、无符号双整型、双字型或枚举	长整型或双字型
长整型或无符号长整型	双精度
REAL	浮点数
LREAL	双精度
日期和时间	日期
STRING	字符串

地址格式

进行变量标记寻址有多种方法：可以在服务器中静态寻址，也可以从客户端动态寻址。标记的格式取决于其类型及用途。变量类型说明如下：

- **数组元素：**可使用下列语法在控制器中定义变量：`ARRAY[x1..x2, y1..y2, z1..z2] OF TYPE`，其中 `TYPE` 是[地址说明](#)中列出的 Omron 数据类型之一。要访问单个元素，请指定 `x`、`y` 和 `z` 偏移。驱动程序块会在最后一个维度上读取请求。例如，使用 `"MyArray [1,0]"` 和 `"MyArray [1,4]"` 等变量，驱动程序可针对从 `"MyArray [1,0]"` 开始的五个元素执行单个请求。有关详细信息，请参阅[通信参数](#)和[优化通信](#)。
- **数组：**可使用下列语法在控制器中定义变量：`ARRAY[x1..x2, y1..y2, z1..z2] OF TYPE`，其中 `TYPE` 是[地址说明](#)中列出的 Omron 数据类型之一。要访问单个客户端项目中的多个元素，请使用数组类型语法。与数组元素一样，驱动程序会执行单个请求来读取和写入多个数组元素。与数组的不同之处在于数组中的所有项目均通过原子操作提供给客户端。不支持字符串数组。

● **注意:** 并非所有客户端都支持数组类型。有关支持信息, 请参阅客户端应用程序。

- **基本:** 使用基本类型定义的变量, 并且在定义时没有使用任何数组语法。
- **字符串:** 使用字符串基本类型定义的变量。

● **提示:** 可将 Sysmac Studio 中的所有“符号变量标记”名称复制并粘贴到服务器的标记地址字段, 并可供使用。

数组元素

必须指定至少一个维度 (但不能多于三个)。

语法	示例
<变量标记名称> [1 维]	tag_1 [5]
<变量标记名称> [1 维, 2 维]	tag_1 [2, 3]
<变量标记名称> [1 维, 2 维, 3 维]	tag_1 [2, 58, 547]

示例

```
MyBooleanArray[31]
MyBooleanArray3D[2,2,7]
MySintArray[1]
MyLrealArray[65535]
MySintArray2D[1,2]
MyLrealArray2D[2,500]
MySintArray3D[2,3,9]
MyLrealArray3D[2,10,10]
```

数组

使用此格式, 可在单个事务处理中读取和写入变量数组的多个元素。客户端必须支持数组类型 (如 "VT_ARRAY")。客户端数据按照行数乘列数的格式进行组织, 以实现一个维度 (1 行, y 列) 或两个维度 (x 行, y 列)。仅一维、二维和三维变量数组支持此格式。与数组元素一样, 必须指定至少一个维度 (但不能多于三个)。

● **注意:** 此驱动程序支持的所有 Omron 数据类型均支持除字符串、日期和时间之外的所有数组格式。

重要事项: 数组不得跨越多个变量数组维度。如果在二维或三维变量数组上创建数组, 则该数组的大小 (即行数乘列数) 不得超过最后一个维度的边界。例如, 假定变量数组 "MySintArray3D" 定义为“短整型”数组 [0..2,0..3,0..9], 则数组标记 MySintArray3D[0,0,0]{10} 有效, 因为它所引用的元素 [0,0,0..9] 均位于最后一个维度的边界内。但是, MySintArray3D[0,0,0]{11} 无效, 因为它要引用的元素 [0,0,0..9] 和 [0,1,0] 的一个元素超过了最后一个维度的边界。

语法	示例
<变量标记名称> [1 维偏移] {列数}	tag_1 [5]{8}
<变量标记名称> [1 维偏移, 2 维偏移] {列数}	tag_2 [0, 5]{8}
<变量标记名称> [1 维偏移, 2 维偏移, 3 维偏移] {列数}	tag_3 [1,0, 5]{8}
<变量标记名称> [1 维偏移] {行数}{列数}	tag_4 [5]{2}{4}
<变量标记名称> [1 维偏移, 2 维偏移] {行数}{列数}	tag_1 [0,5]{2}{4}
<变量标记名称> [1 维偏移, 2 维偏移, 3 维偏移] {行数}{列数}	tag_1 [1,0,5]{2}{4}

● **注意:** 要读取和/或写入的元素数等于行数乘以列数。如果未指定任何行, 则行数默认设置为 1。必须为至少一个数组元素寻址。行数 x 列数必须介于 1 至 65535 之间。

示例

```
MyBooleanArray[0]{32}
MyBooleanArray3D[2,2,7]{1}
MySintArray[1]{5}
MyLrealArray[65535]{1}
MySintArray2D[1,2]{10}
MyLrealArray2D[2,500]{14}{20}
```

```
MySintArray3D[2,3,9]{10}
MyLrealArray3D[2,10,10]{14}{20}
```

基本

语法	示例
<变量标记名称>	tag_1

示例

```
MyBool
MyByte
MyInt
MyWord
MyReal
```

字符串

要读取和/或写入的字符数等于字符串长度，并且必须至少为 2。尽管 Sysmac Studio 允许将变量定义为 STRING [1]，但无法进行读取和写入，这是因为会为空终止符保留一个字符。因此，建议将字符串变量的长度定义为 2 到 1986。要将此空终止符考虑在内，驱动程序中字符串长度的有效范围为 1 到 1985。字符串支持以 UTF-8 编码的所有字符。一个 UTF-8 字符等于 1 到 4 个字节。

● **注意：**如果 256 字节字符串包含需要多个字节的字符，则以 UTF-8 编码时，它代表少于 256 个字符。

语法	示例
<变量标记名称> / <字符串长度>	tag_1 / 255

示例

```
MyString256/255
MyString1986/1985
MyString1986/100
MyStruct[23].Banners[4].Output/10
MyStringArray3D[2,10,10]/255
```

标记范围

● **注意：**仅可在定义本地变量的 POU (程序、功能或功能块) 中读取和写入本地变量。

全局标记

全局标记是在控制器中具有全局范围的“变量标记”。任何程序或任务都可以访问“全局标记”；但是，“全局标记”的引用方式数量取决于其“变量数据类型”和使用的地址格式。

结构标记寻址

结构标记是带有一个或多个成员标记的标记，属性可以是基本型或结构型。

<结构名称> . <基本型标记>

这意味着将按照以下语法对子结构进行寻址：

<结构名称> . <子结构名称> . <基本型标记>

将按照以下语法对结构数组进行寻址：

<结构数组名称> [1 维, 2 维, 3 维] . <基本型标记>

同样，这意味着将按照以下语法对子结构数组进行寻址：

<结构名称> . <子结构数组名称> [1 维, 2 维, 3 维] . <基本型标记>

● **注意：**上述示例仅列举出多种涉及结构的寻址可能性中的几种。所示示例仅用于介绍结构寻址。

● 有关详细信息，请参阅 *Omron NJ 文档*。

预定义术语标记

下表中显示的标记可用于从 PLC 获取常规处理器信息。

标记名称	数据类型	说明
#DEVICETYPE	字	与 PLC 的 EDS 文件中指定的 "ProdType" 属性对应的整数数值。
#REVISION	字符串	固件修订显示为 <主要>.<次要>。
#PRODUCTNAME	字符串	与 PLC 的 EDS 文件中指定的 "ProdName" 属性对应的处理器名称。
#PRODUCTCODE	字	与 PLC 的 EDS 文件中指定的 "ProdCode" 属性对应的整数数值。
#VENDORID	字	与 PLC 的 EDS 文件中指定的 "VendCode" 属性对应的整数数值。

自动生成标记数据库

Omron NJ Ethernet 驱动程序 可配置为在服务器内自动生成标记列表，并使这些标记与 Omron SYSMAC NJ 系列控制器程序中使用的全局变量相对应，并作为输入、输出或仅发布变量发布到网络。

要通过设备生成标记：

1. 在配置中，选择要为其生成标记的设备。
2. 右键单击并选择“属性...”以打开设备属性。
3. 找到并展开“标记生成”部分。
4. 找到“创建”属性。值是蓝色文本，内容为“创建标记”。
5. 单击“创建标记”文本以开始创建标记数据库。
6. 单击“关闭”按钮，退出对话框。
7. 查看“事件日志”中的成功生成确认消息。

● 关于自定义设置的详细信息，请参阅服务器帮助文件。

● 注意：

1. 建议在标记数据库创建过程中，停止与 Omron NJ 设备的所有通信。
2. 为进行枚举而生成的变量标记的数据类型为长整型。

● 另请参阅：[地址格式](#)和[地址说明](#)。

标记层次

自动标记生成所创建的标记可以遵循两个层次之一：“已扩展”(Expanded) 或“已压缩”(Condensed)。要启用此功能，请确保设备属性中的“允许子组”(Allow Sub Groups) 已启用。默认设置为“已扩展”。

已扩展模式

在“已扩展”(Expanded) 模式下，会为每个段 (前面带有句点) 创建标记组 (如在“已压缩”模式下一样)，但还会以逻辑分组的形式创建这些标记组。创建的组包括以下内容：

- 结构和子结构
- 联合
- 数组

基本全局标记 (或非结构、非联合和非数组标记) 位于设备级别。以父组中各自子组的形式提供每个结构、联合和数组标记。

结构、联合或数组子组的名称同样提供对结构、联合和数组的说明。例如，在控制器中定义的数组 tag1[1,6] 具有一个子组名称 "tag1[x,y]"，其中 x 表示存在 1 维，y 表示存在 2 维。数组子组内的标记都是该数组的元素。结构子组中的标记是结构成员本身。如果结构中包含数组，则也会在结构组中创建数组子组。联合子组内的标记是联合成员本身。如果联合中包含数组，则也会在联合组中创建数组子组。

数组标记组

为包含数组元素的每个数组创建组。组名称带有符号：<数组名称>[x,y,z]，其中：

- [X,y,z]x,y,z] 是 3 维数组
- [x,y] 是 2 维数组
- [x] 是 1 维数组

数组标记带有符号：<标记元素>[XXXXXX,YYYYY,ZZZZZ]。例如，元素 tag1[12,2,987] 具有标记名称 "tag1[12,2,987]"。

已压缩模式

在“已压缩”(Condensed) 模式下，自动标记生成所创建的服务器标记遵循的组/标记层次与标记的地址一致。会为每个段 (前面带有句点) 创建组。创建的组包括以下内容：

- 结构和子结构
- 联合

● **注意:** 不会为数组创建组。

事件日志消息

以下信息涉及发布到主要用户界面中“事件日志”窗格的消息。请参阅有关筛选和排序“事件日志”详细信息视图的服务器帮助。服务器帮助包含许多常见的消息，因此也应对其进行搜索。通常，其中会尽可能提供消息的类型(信息、警告)和故障排除信息。

尝试写入标记时出现内部错误。意外的数据类型。| 标记地址 = '<地址>', 数据类型 = '<类型>', DTRV = <代码>。

错误类型:

错误

从设备上载控制器项目时出现以下错误。正在改为使用符号协议。

错误类型:

错误

可能的原因:

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案:

解决方案取决于返回的错误代码。

● **也可以看看:**

CIP 错误代码

发生未知错误。

错误类型:

错误

内存资源不足。

错误类型:

错误

在同步时检测到无效或损坏的控制器项目。将立即重试同步。

错误类型:

错误

同步时检测到项目下载。稍后将重试同步。

错误类型:

错误

上传项目信息期间发生封装错误。| 封装错误 = <代码>。

错误类型:

错误

可能的原因:

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的封装部分返回了错误。

可能的解决方案:

驱动程序尝试从此错误中恢复。

● **注意:**

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

上传项目信息期间发生错误。| CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

错误

可能的原因：

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

● 也可以看看：

CIP 错误代码

上传项目信息期间发生帧错误。

错误类型：

错误

可能的原因：

1. 数据包由于 PC 和设备之间的连接或连接断开而未对齐。
2. 设备电缆连接不良，导致噪声。

可能的解决方案：

1. 将设备置于不繁忙的网络上。
2. 请增加“请求超时”和/或“重试次数”。

上传项目信息期间 CIP 连接超时

错误类型：

错误

数据库错误。上传项目信息期间 CIP 连接超时。

错误类型：

错误

数据库错误。上载项目信息时发生错误。| CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

错误

可能的原因：

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

● 也可以看看：

CIP 错误代码

数据库错误。注册会话请求期间发生封装错误。| 封装错误 = <代码>。

错误类型：

错误

数据库错误。注册会话请求期间发生帧错误。

错误类型：

错误

可能的原因：

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的封装部分返回了错误。

可能的解决方案：

驱动程序尝试从此错误中恢复。

● 注意：

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

● 也可以看看：

封装错误代码

数据库项目。发生内部错误。

错误类型：

错误

数据库错误。转发打开请求时发生封装错误。| 封装错误 = <代码>。

错误类型：

错误

可能的原因：

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的封装部分返回了错误。

可能的解决方案：

驱动程序尝试从此错误中恢复。

● 注意：

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

● 也可以看看：

封装错误代码

数据库错误。fwd. 打开请求再无可用连接。

错误类型：

错误

可能的原因：

Omron 设备支持的连接数存在限制。已超出连接限制。

可能的解决方案：

请减少服务器与设备的连接数，然后重试。

数据库错误。在转发开放请求期间发生错误。| CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

错误

可能的原因：

Omron 设备支持的连接数存在限制。已超出连接限制。

可能的解决方案：

请减少服务器与设备的连接数，然后重试。

数据库错误。在转发开放请求期间发生帧错误。

错误类型：

错误

可能的原因：

1. 数据包由于 PC 和设备之间的连接和/或连接断开而未对齐。
2. 设备电缆连接不良，导致噪声。

可能的解决方案：

1. 将设备置于不繁忙的网络上。
2. 请增加“请求超时”和/或“重试次数”。

数据库错误。上载项目信息时发生封装错误。|封装错误 = <代码>。

错误类型：

错误

可能的原因：

在自动标记生成请求期间，设备在 Ethernet/IP 数据包的封装部分返回了错误。

可能的解决方案：

驱动程序尝试从此错误中恢复。

● 注意：

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

● 也可以看看：

封装错误代码

数据库错误。上载项目信息时发生帧错误。

错误类型：

错误

可能的原因：

1. 数据包由于 PC 和设备之间的连接和/或连接断开而未对齐。
2. 设备电缆连接不良，导致噪声。

可能的解决方案：

1. 将设备置于不繁忙的网络上。
2. 请增加“请求超时”和/或“重试次数”。

从设备接收的帧有误。

错误类型：

警告

可能的原因：

1. 数据包由于 PC 和设备之间的连接和/或连接断开而未对齐。
2. 设备电缆连接不良，导致噪声。

可能的解决方案：

1. 将设备置于不繁忙的网络上。
2. 请增加请求超时和/或尝试次数。

请求设备期间发生错误。| CIP 错误 = <代码>, 扩展错误 = <代码>。

错误类型：

警告

可能的原因：

在请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。请求中的所有读取和写入失败。

可能的解决方案：

解决方案取决于返回的错误代码。

● 也可以看看：

CIP 错误代码

对标记的写入请求由于帧错误而失败。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

1. 由于请求服务代码不正确，导致重试次数过多，对指定标记的写入请求失败。
2. 由于接收到的字节数多于或少于预期，导致重试次数过多，对指定标记的写入请求失败。

可能的解决方案：

如果此错误频繁发生，电缆或设备可能出现了问题。请增加重试次数以为驱动程序从此错误中恢复提供更多机会。

对标记的读取请求由于帧错误而失败。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

1. 请求服务代码不正确。
2. 接收到的字节数多于或少于预期。
3. 电缆或设备可能出现了问题。

可能的解决方案：

请增加“重试次数”以为驱动程序从此错误中恢复提供更多机会。

块读取请求由于帧错误而失败。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。

错误类型:

警告

可能的原因:

1. 由于请求服务代码不正确, 指定的标记地址和计数失败。
2. 由于接收到的字节数多于或少于预期, 指定的标记地址和计数失败。

可能的解决方案:

如果此错误频繁发生, 电缆或设备本身可能出现了问题。请增加请求尝试次数以为驱动程序从此错误中恢复提供更多机会。为响应此错误, 块的元素将被取消激活, 并且不会再次对其进行处理。

块读取请求由于帧错误而失败。| 块大小 = <数字> (字节), 标记名称 = '<标记>'。

错误类型:

警告

可能的原因:

1. 由于请求服务代码不正确, 指定的标记地址和计数失败。
2. 由于接收到的字节数多于或少于预期, 指定的标记地址和计数失败。

可能的解决方案:

如果此错误频繁发生, 电缆或设备本身可能出现了问题。请增加请求尝试次数以为驱动程序从此错误中恢复提供更多机会。为响应此错误, 块的元素将被取消激活, 并且不会再次对其进行处理。

无法写入标记。| 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>'。

错误类型:

警告

可能的原因:

1. 设备与主机 PC 之间的以太网连接断开。
2. 以太网连接的通信参数不正确。
3. 可能为指定设备分配了不正确的 IP 地址。

可能的解决方案:

1. 验证 PC 和设备之间的电缆连接。
2. 验证是否已为指定设备指定正确端口。
3. 验证分配给指定设备的 IP 地址是否与实际设备的 IP 地址相符。

无法读取标记。| 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>'。

错误类型:

警告

可能的原因：

在对指定标记的读取请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

也可以看看：

CIP 错误代码

无法读取设备上的块。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'，CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

警告

可能的原因：

在对指定标记的读取请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

也可以看看：

CIP 错误代码

无法读取设备上的块。| 块大小 = <数字> (字节)，标记名称 = '<标记>'，CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

警告

可能的原因：

在对指定标记的读取请求期间，设备在 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

也可以看看：

CIP 错误代码

无法写入标记。控制器标记数据类型未知。| 标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于不支持变量数据类型，对指定标记的读取请求失败。

可能的解决方案：

移除对此变量的引用。为响应此错误，标记将被取消激活，并且不会再次对其进行处理。

无法读取标记。控制器标记数据类型未知。标记已取消激活。| 标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于不支持变量数据类型，对指定标记的读取请求失败。

可能的解决方案：

移除对此变量的引用。为响应此错误，标记将被取消激活，并且不会再次对其进行处理。

无法读取设备上的块。控制器标记数据类型未知。标记已取消激活。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于不支持变量数据类型，对指定标记的读取请求失败。

可能的解决方案：

移除对此变量的引用。为响应此错误，块将被取消激活，并且不会再次对其进行处理。

无法写入标记。不支持数据类型。| 标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于不支持客户端标记数据类型，对指定标记的写入请求失败。

可能的解决方案：

将标记数据类型更改为支持的类型，然后再试一次。

无法读取标记。不支持数据类型。标记已取消激活。| 标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于不支持客户端标记数据类型，对指定标记的读取请求失败。

可能的解决方案：

将标记数据类型更改为支持的类型。为响应此错误，标记将被取消激活，并且不会再次对其进行处理。

无法读取设备上的块。不支持数据类型。块已取消激活。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'，数据类型 = '<类型>'。

错误类型：

警告

可能的原因：

由于不支持客户端标记数据类型，对指定标记的指定标记地址和计数的读取请求失败。

可能的解决方案：

将此块中的标记数据类型更改为支持的类型。为响应此错误，块的元素已取消激活，并且不会再次被处理。

无法写入标记。此标记数据类型非法。| 标记地址 = '<地址>'，数据类型 = <类型>。

错误类型：

警告

可能的原因：

由于给定变量的客户端标记数据类型是非法的，对指定标记的写入请求失败。

可能的解决方案：

将标记数据类型更改为支持的类型。

也可以看看：

数据类型强制

无法读取标记。此标记数据类型非法。| 标记地址 = '<地址>'，数据类型 = '<类型>'。

错误类型：

警告

可能的原因：

由于给定变量的客户端标记数据类型是非法的，对指定标记的读取请求失败。

可能的解决方案：

将标记数据类型更改为支持的类型。

也可以看看：

数据类型强制

无法读取设备上的块。此块的数据类型是非法的。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'，数据类型 = '<类型>'。

错误类型：

警告

可能的原因：

由于给定变量的客户端标记数据类型是非法的，对指定标记的读取请求失败。

可能的解决方案：

将标记数据类型更改为支持的类型。

也可以看看：

数据类型强制

无法写入标记。标记不支持多元素数组。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

由于驱动程序不支持对给定变量进行多元素数组访问，对指定标记的写入请求失败。

可能的解决方案：

将标记数据类型或地址更改为支持的类型。

无法读取标记。标记不支持多元素数组。标记已取消激活。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

由于驱动程序不支持对给定变量进行多元素数组访问，对指定标记的读取请求失败。

可能的解决方案：

将标记数据类型或地址更改为支持的类型。为响应此错误，标记将被取消激活，并且不会再次对其进行处理。

无法读取块。块不支持多元素数组。块已取消激活。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

由于驱动程序不支持对给定变量进行多元素数组访问，对指定标记的指定标记地址和计数的读取请求失败。

可能的解决方案：

将此块中的标记数据类型或地址更改为支持的类型或地址。为响应此错误，块的元素将被取消激活，并且不会再次对其进行处理。

无法写入标记。原生标记大小不匹配。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取标记。原生标记大小不匹配。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取设备上的块。原生标记大小不匹配。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'。

错误类型：

警告

无法读取设备上的块。原生标记大小不匹配。| 块大小 = <数字> (字节)，标记名称 = '<标记>'。

错误类型：

警告

无法写入标记。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

1. 设备与主机 PC 之间的以太网连接断开。
2. 以太网连接的通信参数不正确。
3. 可能为指定设备分配了不正确的 IP 地址。

可能的解决方案：

1. 验证 PC 和设备之间的电缆连接
2. 验证是否已为指定设备指定正确端口。
3. 验证分配给指定设备的 IP 地址是否与实际设备的 IP 地址相符。

无法读取标记。标记已取消激活。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取块。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。

错误类型：

警告

无法读取设备上的块。块已取消激活。| 块大小 = <数字> (字节), 标记名称 = '<标记>'。

错误类型：

警告

无法读取标记。内存无效。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取标记。此标记数据类型非法。| 标记地址 = '<地址>', 数据类型 = '<类型>'。

错误类型：

警告

可能的原因：

由于给定变量的客户端标记数据类型是非法的, 对指定标记的读取请求失败。

可能的解决方案：

将标记的数据类型更改为支持的类型。

● 也可以看看：

数据类型强制

无法读取设备上的块。内存无效。标记已取消激活。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取设备上的块。内存无效。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。

错误类型：

警告

无法写入地址。内存无效。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取设备上的块。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。

错误类型：

警告

在读取标记时，设备所返回的数据超过预期。请验证地址，包括元素偏移和该偏移中的所有维度。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

标记引用了数组变量，但在元素偏移中并未指定所有维度，或未指定任何元素偏移。例如，将 "MyArray" 定义为 ARRAY[0..2,0..9] OF INT。创建一个类似 MyArray[0]@Word 数组的标记地址可能会导致出现此错误，因为未指定第二个维度。在此例中，读取该标记将会返回 10 个整数，而不是所预期的 2 个整数。同样，创建一个类似 MyArray@Word 的标记地址也可能导致出现此错误，因为既未指定第一个维度也未指定第二个维度。在此例中，读取该标记将返回 30 个整数，而不是所预期的 1 个整数。

可能的解决方案：

向标记地址添加完全限定的元素偏移。在以上两例中，正确的标记地址应分别为 MyArray[0,0] 和 MyArray[0,0]。

● **注意：**

此消息仅仅是警告。读取会成功，但效率会有所降低，因为读取响应花费了额外的时间。建议用户修复显示此行为的标记地址。

在读取块时，设备所返回的数据超过预期。请验证地址，包括元素偏移和该偏移中的所有维度。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

标记引用了数组变量，但在元素偏移中并未指定所有维度，或未指定任何元素偏移。例如，将 "MyArray" 定义为 ARRAY[0..2,0..9] OF INT。创建一个类似 MyArray[0]@Word 数组的标记地址可能会导致出现此错误，因为未指定第二个维度。在此例中，读取该标记将会返回 10 个整数，而不是所预期的 2 个整数。同样，创建一个类似 MyArray@Word 的标记地址也可能导致出现此错误，因为既未指定第一个维度也未指定第二个维度。在此例中，读取该标记将返回 30 个整数，而不是所预期的 1 个整数。

可能的解决方案：

向标记地址添加完全限定的元素偏移。在以上两例中，正确的标记地址应分别为 MyArray[0,0] 和 MyArray[0,0]。

● **注意：**

此消息仅仅是警告。读取会成功，但效率会有所降低，因为读取响应花费了额外的时间。建议用户修复显示此行为的标记地址。

无法写入标记。地址超出了当前 CIP 连接大小。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

请求帧中产生的超出协议限制的标记地址大小。

可能的解决方案：

将“CIP 连接大小”增加到可容纳标记请求帧的值。

无法读取设备上的块。地址超出了当前 CIP 连接大小。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

请求帧中产生的超出协议限制的标记地址大小。

可能的解决方案：

将“CIP 连接大小”增加到可容纳标记请求帧的值。

无法读取标记。地址超出了当前 CIP 连接大小。| 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

请求帧中产生的超出协议限制的标记地址大小。

可能的解决方案：

将“CIP 连接大小”增加到可容纳标记请求帧的值。

本设备不支持请求的 CIP 连接大小。正在自动回退到最大大小。| 请求的大小 = <数字> (字节)，最大大小 = <数字> (字节)。

错误类型：

警告

对于 XML 元素，当前值在此模型上不受支持。正在自动设置为新值。| 当前值 = '<值>'，XML 元素 = '{<命名空间><元素>'，模型 = '<模型>'，新值 = '<值>'。

错误类型：

警告

数据库错误。复杂类型的数据类型不受支持。不会将此成员的标记添加到数据库。| 数据类型 = <类型>，复杂类型 = '<类型>'，成员 = '<名称>'。

错误类型：

警告

可能的原因：

复杂类型成员的设备配置使用了不支持的数据类型。

可能的解决方案：

请将复杂类型成员的设备配置修改为使用支持的数据类型。

● 也可以看看：

地址说明

数据库错误。无法解析标记的 CIP 数据类型。不会将标记添加到数据库。| 数据类型 = <类型>，标记名称 = '<标记>'。

错误类型：

警告

可能的原因：

1. 在自动标记数据库生成期间因以太网封装、设备或组帧而发生通信错误。
2. 驱动程序不支持 CIP 数据类型。

可能的解决方案：

1. 更正通信错误，然后重新尝试自动标记数据库生成。
2. 将变量数据类型更改为支持的类型。

数据库错误。标记的地址验证失败。不会将标记添加到数据库。| 标记名称 = '<标记>', 标记地址 = '<地址>'。

错误类型：

警告

可能的原因：

标记地址大小超出协议限制。

可能的解决方案：

请减少标记名称中的字符数。如果错误仍然存在，请减少标记所在的结构或联合 (包括任何嵌套结构或联合) 中的字符数。

无法检索设备的标识。| 封装错误 = <代码>。

错误类型：

警告

可能的原因：

未检索到标识，因为设备在请求期间于 Ethernet/IP 数据包的封装部分返回了错误。

可能的解决方案：

驱动程序尝试从此类错误中恢复。

注意：

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

也可以看看：

封装错误代码

无法检索设备的标识。| CIP 错误 = <代码>，扩展错误 = <代码>。

错误类型：

警告

可能的原因：

未检索到标识，因为设备在请求期间于 Ethernet/IP 数据包的 CIP 部分返回了错误。

可能的解决方案：

解决方案取决于返回的错误代码。

也可以看看：

CIP 错误代码

无法检索设备的标识。接收到的帧有错误。

错误类型：

警告

可能的原因：

1. 数据包由于 PC 和设备之间的连接和/或连接断开而未对齐。
2. 设备电缆连接不良，导致噪声。

3. 接收到的帧大小错误。
4. TNS 不匹配。
5. 从设备返回了无效的响应命令。
6. 设备未启用 Ethernet/IP。

可能的解决方案：

1. 驱动程序不经干预即可从此错误中恢复。电缆、网络或设备本身可能存在问题。
2. 验证与设备进行通信的是否为已启用 Omron Ethernet/IP 的设备。

请求期间发生封装错误。| 封装错误 = <代码>。**错误类型：**

信息化

可能的原因：

在请求期间，设备在 Ethernet/IP 数据包的封装部分返回了错误。请求中的所有读取和写入失败。

可能的解决方案：

驱动程序尝试从此类错误中恢复。

● 注意：

这排除了错误 0x02，该错误是与设备相关的，而不是与驱动程序相关。

● 也可以看看：

封装错误代码

无法为标记分配内存。| 标记地址 = '<地址>'。**错误类型：**

信息化

可能的原因：

无法分配生成进程所需的内存。进程将中止。

可能的解决方案：

关闭任何未使用的应用程序和/或增加虚拟内存量，然后再试一次。

数据库错误。标记已重命名，因为它超出了最大字符长度。| 标记名称 = '<标记>'，最大长度 = <数字>，新标记名称 = '<标记>'。**错误类型：**

信息化

数据库错误。数组标记已重命名，因为它超出了最大字符长度。| 数组标记名称 = '<名称>'，最大长度 = <数字>，新数组标记名称 = '<名称>'。**错误类型：**

信息化

数据库状态: 标记已导入。| 数据类型 = <类型>，导入的标记数 = <数字>。**错误类型：**

信息化

数据库状态: 正在生成 OPC 标记。

错误类型:

信息化

数据库状态: 正在构建标记项目, 请等待。| 标记项目计数 = <数字>。

错误类型:

信息化

数据库状态: 正在检索控制器项目。

错误类型:

信息化

运行时间 = <数字> (秒)

错误类型:

信息化

符号设备读取次数 = <数字>

错误类型:

信息化

符号、数组块设备读取次数 = <数字>

错误类型:

信息化

符号、数组块缓存读取次数 = <数字>

错误类型:

信息化

符号实例非块设备读取次数 = <数字>

错误类型:

信息化

符号实例非块、数组块设备读取次数 = <数字>

错误类型:

信息化

符号实例非块、数组块缓存读取次数 = <数字>

错误类型:

信息化

符号实例块设备读取次数 = <数字>

错误类型:

信息化

符号实例块缓存读取次数 = <数字>

错误类型:

信息化

读取的标记数 = <数字>

错误类型：
信息化

发送的数据包数 = <数字>

错误类型：
信息化

接收的数据包数 = <数字>

错误类型：
信息化

初始化事务数 = <数字>

错误类型：
信息化

读取/写入事务数 = <数字>

错误类型：
信息化

平均每秒发送的数据包数 = <数字>

错误类型：
信息化

平均每秒接收的数据包数 = <数字>

错误类型：
信息化

平均每秒标记读取次数 = <数字>

错误类型：
信息化

平均每个事务的标记数 = <数字>

错误类型：
信息化

-

错误类型：
信息化

%s | 设备统计信息

错误类型：
信息化

平均设备回转时间 = <数字> (毫秒)。

错误类型：
信息化

%s | 通道统计信息

错误类型:

信息化

驱动器统计信息

错误类型:

信息化

详细信息。| IP = '<地址>', 供应商 ID = <供应商>, 设备类型 = <类型>, 产品代码 = <代码>, 修订版本 = <修订版本>, 产品名称 = '<名称>', 产品序列号 = <数字>。

错误类型:

信息化

检索控制器项目出错。

错误类型:

信息化

内部驱动器出错。

错误类型:

信息化

在同步时检测到无效或损坏的控制器项目。请稍后重试。

错误类型:

信息化

在同步时检测到项目下载。请稍后重试。

错误类型:

信息化

内存资源不足。

错误类型:

信息化

发生未知错误。

错误类型:

信息化

错误代码

以下部分定义可能在服务器的事件日志中遇到的错误代码。有关特定错误代码类型的详细信息，请从下表中选择一个链接。

[封装错误代码](#)

[CIP 错误代码](#)

封装错误代码

以下错误代码为十六进制。

状态 (十六进制)	说明
0001	发件人发出无效或不受支持的封装命令。
0002	接收器中没有足够的内存资源来处理该命令。
0003	封装消息数据部分中的信息不正确或不完整。
0004 - 0063	保留
0064	发起者在向目标发送封装消息时使用了无效的会话句柄。
0065	标题长度无效。
0066 - 0068	保留
0069	不支持请求的协议版本。
006A-FFFF	保留

CIP 错误代码

以下错误代码为十六进制。

状态 (十六进制)	说明
01	连接相关服务在连接路径时失败。另请参阅: 0x01 扩展错误代码
02	对象执行已请求服务所需的资源不可用。
03	参数值无效。
04	路径段错误。标记未存在于设备中。
05	路径目标未知。结构成员不存在或数组元素超出范围。
06	部分传输;仅传输了部分预期数据。
07	失去连接。
08	服务不受支持。已请求服务未执行,或未针对此类或实例进行定义。
09	属性值无效。
0A	属性列表错误。
0B	对象已处于服务所请求的模式/状态。
0C	对象无法在当前模式/状态下执行已请求服务。项目更改可能正在进行。另请参阅: 0x0C 扩展错误代码
0D	待创建对象所请求的实例已存在。
0E	接收到修改非可编辑属性的请求。
0F	权限/特权检查失败。
10	设备的当前模式/状态禁止执行所请求的服务。
11	回复数据过大。要在响应缓冲区传输的数据大于分配的响应缓冲区。
12	服务指定了将对原始数据值进行分段的操作。
13	数据不足。服务未提供足够的数据来执行指定操作。
14	不支持属性。
15	数据太多。服务所提供的数据已超出预期。
16	指定的对象未存在于设备中。

状态 (十六进制)	说明
17	当前, 服务的碎片化序列对于此数据并未处于活动状态。
18	此对象的属性数据未在所请求的服务之前保存。
19	此对象的属性数据未保存, 因为尝试期间存在故障。
1A	路由失败; 请求数据包过大。
1B	路由失败; 响应数据包太大。
1C	列表条目数据中属性缺失。
1D	属性值列表无效。
1E	嵌入式服务错误。在多服务包服务中, 一个或多个服务返回错误。
1F	供应商特定错误。请参阅厂商文档。另请参阅: 0x1F 扩展错误代码
20	参数无效。参数不满足 CIP 规范或 Omron 规范的要求。另请参阅: 0x20 扩展错误代码
21	尝试对已写入的一次性写入媒介进行写入操作。
22	收到的回复无效。回复服务代码与请求服务代码不匹配或回复消息长度小于最短预期回复长度。
23	已接收的消息大于接收缓冲区可处理的量。
24	已接收消息的格式在服务器中不受支持。
25	已包含为路径第一段的关键段与目标模块不匹配。
26	与服务请求一起发送的路径, 大小不足以将请求路由至对象或包括过多路由数据。
27	列表中存在意外的属性。
28	请求中指定的成员 ID 在指定的类、实例或属性中不存在。
29	接收到修改非可修改成员请求。
2A	DeviceNet 特定错误。
2B	CIP 到 Modbus 转换器接收到未知的 Modbus 异常代码。
2C	接收到读取非可读属性的请求。
2D	不能删除已请求的对象实例。
2E	对象支持此服务, 但不支持指定的应用程序路径 (例如, 属性)。
2F - CF	由 CIP 保留。
D0 - FF	对象类特定错误。

0x01 扩展错误代码

以下错误代码为十六进制。

扩展状态 (十六进制)	说明
0100	连接正在使用中或“转发开放”请求重复。发起者正在尝试通过已建立的连接与目标相连。
0101 - 0102	由 CIP 保留。
0103	目标应用程序不支持指定的传输类和触发器组合。
0104	由 CIP 保留。
0105	请参阅“CIP 安全规范”。
0106	所有权冲突。无法建立连接, 因为另一个连接以独占方式分配了此连接所需的一些资源。
0107	未找到目标连接。在目标节点中未找到要关闭的连接时, 通常会针对“转发关闭”请求返回此错误代码。
0108	网络连接参数无效。设备不支持连接类型、优先级或固定量/变量。
0109	连接大小无效。目标不支持指定的连接大小。
010A - 010F	由 CIP 保留。
0110	未配置连接目标。
0111	不支持 RPI。如果设备不支持请求 (O->T 或 T->O RPI) 或不支持连接超时倍乘系数生成的

扩展状态 (十六进制)	说明
	超时值, 则将返回此错误代码。
0112	RPI 值不可接受。如果“转发开放”请求的 RPI 值超出目标设备中应用程序的所需范围或目标以不同的间隔生成值, 则将返回此错误代码。
0113	超出连接数。已达到“连接管理器”所支持的最大连接数。
0114	电子密钥逻辑段中指定的供应商 ID 或产品代码与设备的供应商 ID 或产品代码不匹配。
0115	电子密钥逻辑段中指定的设备类型与设备的设备类型不匹配。
0116	电子密钥逻辑段中指定的主要和次要修订版本与设备的修订版本不匹配。
0117	连接路径中指定的已生成或已使用应用程序路径与目标应用程序中有效的已生成或已使用应用程序路径不对应。
0118	为配置数据指定的应用程序路径与配置应用程序不对应, 或与已使用或已生成的应用程序路径不一致。
0119	当前未打开任何非仅监听类型的连接。如果尝试与目标建立仅监听类型的连接, 但该目标尚未建立任何非仅监听连接, 则将返回此错误代码。
011A	已超过目标对象的此实例所支持的最大连接数。
011B	“生产禁止时间”大于 T->ORPI。
011C	不支持传输类型/触发参数中请求的传输类。
011D	不支持传输类型/触发参数中请求的生产触发器。
011E	不支持传输类型/触发参数中请求的方向。
011F	不支持 O->T 固定量/变量标志。
0120	不支持 T->O 固定量/变量标志。
0121	不支持 O->T 优先级代码。
0122	不支持 T->O 优先级代码。
0123	不支持 O->T 连接类型。
0124	不支持 T->O 连接类型。
0125	不支持 O->T 冗余所有者标志。
0126	连接路径参数中的数据段不包含请求的配置应用程序路径可接受的 16 位字数。
0127	在“转发开放”请求中声明且在目标上可用的使用对象大小与 O->T 网络连接参数中声明的连接大小不匹配。
0128	在“转发开放”请求中声明且在目标上可用的生成对象大小与 T->O 网络连接参数中声明的连接大小不匹配。
0129	连接路径中指定的配置应用程序路径与目标应用程序中有效的配置应用程序路径不对应。
012A	连接路径中指定的已使用应用程序路径与目标应用程序中有效的已使用应用程序路径不对应。
012B	连接路径中指定的已生成应用程序路径与目标应用程序中有效的已生成应用程序路径不对应。
012C	发起者尝试连接的配置标记名称不在目标中定义的标记列表中。
012D	发起者尝试连接的使用标记名称不在目标中定义的标记列表中。
012E	发起者尝试连接的生成标记名称不在目标中定义的标记列表中。
012F	连接路径中指定的配置、使用和/或生成应用程序路径的组合彼此不一致。
0130	数据段中的信息与已使用数据的格式不一致。
0131	数据段中的信息与已生成数据的格式不一致。
0132	目标不支持空的“转发开放”请求。
0133	指定的连接超时倍乘系数 (非活动情况监视器) 被保留或生成的超时值对于设备而言过大。
0134 - 0202	由 CIP 保留。
0203	连接超时。

扩展状态 (十六进制)	说明
0204	未连接请求超时。这可能是由于目标节点拥塞或节点未上电或不存在的。
0205	中间节点不支持未连接请求中的连接时间和连接超时组合。
0206	消息对于 <code>unconnected_send</code> 服务而言过大。
0207	接收到未连接确认消息，但未接收到数据响应消息。
0208 - 0300	由 CIP 保留。
0301	连接缓冲区内内存不足。
0302	沿路径的生成者无法为其链路上的连接分配足够带宽。
0303	没有可用的已使用连接 ID 筛选器。
0304	设备无法发送已安排的优先级数据。
0305	发起者设备中的连接安排信息与目标网络中的连接安排信息不一致。
0306	发起者设备中的连接安排信息无法在目标网络中进行验证。
0307 - 0310	由 CIP 保留。
0311	端口段中指定的端口不可用或不存在的。
0312	端口段中指定的链路地址不适用于目标网络类型。
0313 - 0314	由 CIP 保留。
0315	连接路径中的段类型或段值无效。
0316	“转发关闭”服务中的连接路径与正在关闭的连接中的连接路径不匹配。
0317	已安排的网络段不存在或段内的值无效。
0318	至自身的链路地址 (环回地址) 无效。
0319	双机架冗余系统中的次系统无法复制针对主系统的连接请求。
031A	已拒绝模块连接请求，因为部分相应数据已包括在机架连接中。
031B	已拒绝机架连接请求，因为部分相应数据已包括在模块连接中。
031C	发生其他连接相关错误。
031D	冗余连接不匹配。
031E	已达到生成应用程序的使用者配置数。
031F	未为要使用的生成应用程序配置使用者。
0320 - 07FF	供应商特定错误。
0800	模块路径中的网络链路处于离线状态。
0801 - 080F	请参阅“CIP 安全规范”。
0810	目标应用程序不具有请求连接所需的有效数据。
0811	发起者应用程序不具有请求连接所需的有效数据。
0812	由于已安排网络，因此节点地址已更改。
0813	已在位于不同子网上的生成者和使用者之间请求多播连接，且未针对离线子网多播配置生成者。
0814	数据段中的信息指示已生成和/或使用数据的格式无效。
0815 - FCFF	由 CIP 保留。

0x0C 扩展错误代码

以下错误代码为十六进制。

扩展状态 (十六进制)	说明
8010	正在进行下载。
8011	标记内存中存在错误。

● 对于未列出的错误代码，请参阅 *Omron* 文档。

0x1F 扩展错误代码

以下错误代码为十六进制。

扩展状态 (十六进制)	说明
0101	可能是下列错误之一： <ul style="list-style-type: none"> • 变量类型和读取地址的组合大小错误。 • 变量类型规范错误。 • 读取起始地址超出变量区域范围。 • 读取结束地址超出变量区域范围。 • 元素数过多。
0102	可能是下列错误之一： <ul style="list-style-type: none"> • 元素数与写入数据的大小不匹配。 • 变量类型规范错误。 • 写入区域中包含只读区域。
0104	变量类型超出范围。
8001	发生内部错误。
800D	注册标记信息中存在错误。
8014	发生内部错误。
8016	变量未正确注册。

● 对于未列出的错误代码，请参阅 *Omron* 文档。

0x20 扩展错误代码

以下错误代码为十六进制。

扩展状态 (十六进制)	说明
8017	已为不包含元素的变量指定多个元素。
8018	未为数组指定元素或为数组指定的数据超出数组范围。
8022	请求服务数据中指定的数据类型与标记信息不一致。请求服务数据中的 AddInfo 长度不是 0。
8028	值超出范围。

● 对于未列出的错误代码，请参阅 *Omron* 文档。

索引

-

----- 58

%

%s | 设备统计信息 58

%s | 通道统计信息 59

0

0x01 扩展错误代码 61

0x0C 扩展错误代码 63

0x1F 扩展错误代码 64

0x20 扩展错误代码 64

C

CIP 20, 29

CIP 错误代码; 错误代码 60

E

EtherNet-IP 19, 29

I

ID 16, 25

N

NJ 20, 29

T

TCP/IP 端口 20, 29

罍

本设备不支持请求的 CIP 连接大小。正在自动回退到最大大小。| 请求的大小 = <数字> (字节), 最大大小 = <数字> (字节)。 54

椀

标记层次 40

标记地址规则 36

标记名称规则 36

标记生成 18, 27

标识 15, 24

标识请求 8

尪

不扫描, 仅按需求轮询 16, 26

嶲

布尔型 34

宽

尝试写入标记时出现内部错误。意外的数据类型。| 标记地址 = '<地址>', 数据类型 = '<类型>', DTRV = <代码>。 42

败

超时前的尝试次数 17, 27

冽

初始化事务数 = <数字> 58

冻

创建 19, 29

空

从设备接收的帧有误。 46

从设备上载控制器项目时出现以下错误。正在改为使用符号协议。 42

钹

错误代码 60

嗽

地址格式 36

地址说明 35

讌

读取/写入事务数 = <数字> 58

读取的标记数 = <数字> 58

穿

端口 ID 8

瞍

短整型 34

宙

对标记的读取请求由于帧错误而失败。| 标记地址 = '<地址>'。 46

对标记的写入请求由于帧错误而失败。| 标记地址 = '<地址>'。 46

对数组进行分块 22, 32

对于 XML 元素, 当前值在此模型上不受支持。正在自动设置为新值。| 当前值 = '<值>', XML 元素 = '{<命名空间><元素>', 模型 = '<模型>', 新值 = '<值>'。 54

对于重复标记 18, 28

噢

多请求数据包 22, 32

多跃点 9

危

发生未知错误。 42, 59

发送的数据包数 = <数字> 58

雾

非活动情况监视器 20, 29

审

封装错误代码 60

洎

浮点型 34

窳

符号、数组块缓存读取次数 = <数字> 57

符号、数组块设备读取次数 = <数字> 57

符号设备读取次数 = <数字> 57

符号实例非块、数组块缓存读取次数 = <数字> 57

符号实例非块、数组块设备读取次数 = <数字> 57

符号实例非块设备读取次数 = <数字> 57

符号实例块缓存读取次数 = <数字> 57

符号实例块设备读取次数 = <数字> 57

燭

父组 19, 28

襪

覆盖 19, 28

榧

概述 7

孽

故障时降级 17, 27

垚

基本 37

桎

检索控制器项目出错。 59

闭

降级超时 17, 27

降级期间 18, 27

降级时放弃请求 18, 27

捅

接收的数据包数 = <数字> 58

繳

结构标记寻址 38

结构标记寻址; 标记范围 38

困

块读取请求由于帧错误而失败。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。 47

块读取请求由于帧错误而失败。| 块大小 = <数字> (字节), 标记名称 = '<标记>'。 47

竭

来自缓存的初始更新 16, 26

轻

连接超时 17, 26

连接大小 20, 22, 29, 32

连接路径规范 8

钞

链路地址 9

超

路由定时 8

路由路径 9

路由示例 9

稿

模拟 16, 25

模型 15, 25

價

内部驱动器出错。 59

内存资源不足。 42, 59

昧

平均每个事务的标记数 = <数字> 58

平均每秒标记读取次数 = <数字> 58

平均每秒发送的数据包数 = <数字> 58

平均每秒接收的数据包数 = <数字> 58

平均设备回转时间 = <数字> (毫秒)。 58

廚

强制 36

講

请求超时 17, 27

请求间延迟 17, 27

请求期间发生封装错误。| 封装错误 = <代码>。 56

请求设备期间发生错误。| CIP 错误 = <代码>, 扩展错误 = <代码>。 46

駢

驱动程序 15, 25

驱动器统计信息 59

儇

全局标记 38

儻

冗余 21, 30

戔

扫描模式 16, 26

净

删除 19, 28

讖

上传项目信息期间 CIP 连接超时 43

上传项目信息期间发生错误。| CIP 错误 = <代码>, 扩展错误 = <代码>。 43

上传项目信息期间发生封装错误。| 封装错误 = <代码>。 42

上传项目信息期间发生帧错误。 43

譚

设备启动时 18, 28

设备属性 15

设备属性 - 标记生成 18, 27

设置 8

璿

生成 18, 28

Υ

事件日志消息 42

対

属性更改时 18,28

攘

数据库错误。fwd. 打开请求再无可用连接。 44

数据库错误。标记的地址验证失败。不会将标记添加到数据库。| 标记名称 = '<标记>', 标记地址 = '<地址>'。 55

数据库错误。标记已重命名, 因为它超出了最大字符长度。| 标记名称 = '<标记>', 最大长度 = <数字>, 新标记名称 = '<标记>'。 56

数据库错误。复杂类型的数据类型不受支持。不会将此成员的标记添加到数据库。| 数据类型 = <类型>, 复杂类型 = '<类型>', 成员 = '<名称>'。 54

数据库错误。上传项目信息期间 CIP 连接超时。 43

数据库错误。上载项目信息时发生错误。| CIP 错误 = <代码>, 扩展错误 = <代码>。 43

数据库错误。上载项目信息时发生封装错误。| 封装错误 = <代码>。 45

数据库错误。上载项目信息时发生帧错误。 45

数据库错误。数组标记已重命名, 因为它超出了最大字符长度。| 数组标记名称 = '<名称>', 最大长度 = <数字>, 新数组标记名称 = '<名称>'。 56

数据库错误。无法解析标记的 CIP 数据类型。不会将标记添加到数据库。| 数据类型 = <类型>, 标记名称 = '<标记>'。 54

数据库错误。在转发开放请求期间发生错误。| CIP 错误 = <代码>, 扩展错误 = <代码>。 44

数据库错误。在转发开放请求期间发生帧错误。 45

数据库错误。注册会话请求期间发生封装错误。| 封装错误 = <代码>。 44

数据库错误。注册会话请求期间发生帧错误。 44

数据库错误。转发打开请求时发生封装错误。| 封装错误 = <代码>。 44

数据库项目。发生内部错误。 44

数据库状态

标记已导入。| 数据类型 = <类型>, 导入的标记数 = <数字>。 56

正在构建标记项目, 请等待。| 标记项目计数 = <数字>。 57

正在检索控制器项目。 57

正在生成 OPC 标记。 57

数据类型说明 34

数据收集 16,25

数组 36

数组标记组 40

数组块大小 20,29

数组元素 36

印

双精度 34

双字型 34

讠

调节 33

辺

通道分配 15, 25

通信参数 19, 29

通信超时 17, 26-27

通信路由和定时 8

通信协议 8

辱

同步时检测到项目下载。稍后将重试同步。 42

繼

统计信息 33

駁

无法读取标记。| 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。 47

无法读取标记。标记不支持多元素数组。标记已取消激活。| 标记地址 = '<地址>'。 50

无法读取标记。标记已取消激活。| 标记地址 = '<地址>'。 52

无法读取标记。不支持数据类型。标记已取消激活。| 标记地址 = '<地址>', 数据类型 = <类型>。 49

无法读取标记。此标记数据类型非法。| 标记地址 = '<地址>', 数据类型 = '<类型>'。 50, 52

无法读取标记。地址超出了当前 CIP 连接大小。| 标记地址 = '<地址>'。 54

无法读取标记。控制器标记数据类型未知。标记已取消激活。| 标记地址 = '<地址>', 数据类型 = <类型>。
48

无法读取标记。内存无效。| 标记地址 = '<地址>'。 52

无法读取标记。原生标记大小不匹配。| 标记地址 = '<地址>'。 51

无法读取块。块不支持多元素数组。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。
51

无法读取块。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。 52

无法读取设备上的块。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。 48

无法读取设备上的块。| 块大小 = <数字> (字节), 标记名称 = '<标记>', CIP 错误 = <代码>, 扩展错误 = <代码>。 48

无法读取设备上的块。不支持数据类型。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = '<类型>'。 49

无法读取设备上的块。此块的数据类型是非法的。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = '<类型>'。 50

无法读取设备上的块。地址超出了当前 CIP 连接大小。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。 53

无法读取设备上的块。控制器标记数据类型未知。标记已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', 数据类型 = <类型>。 49

无法读取设备上的块。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。 52

无法读取设备上的块。块已取消激活。| 块大小 = <数字> (字节), 标记名称 = '<标记>'。 52

无法读取设备上的块。内存无效。标记已取消激活。| 标记地址 = '<地址>'。 52

无法读取设备上的块。内存无效。块已取消激活。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。 52

无法读取设备上的块。原生标记大小不匹配。| 块大小 = <数字> (元素), 起始标记地址 = '<地址>'。 51

无法读取设备上的块。原生标记大小不匹配。| 块大小 = <数字> (字节), 标记名称 = '<标记>'。 51

无法检索设备的标识。| CIP 错误 = <代码>, 扩展错误 = <代码>。 55

无法检索设备的标识。| 封装错误 = <代码>。 55

无法检索设备的标识。接收到的帧有错误。 55

无法为标记分配内存。| 标记地址 = '<地址>'。 56

无法写入标记。| 标记地址 = '<地址>', CIP 错误 = <代码>, 扩展错误 = <代码>。 47

无法写入标记。| 标记地址 = '<地址>'。 51

无法写入标记。标记不支持多元素数组。| 标记地址 = '<地址>'。 50

无法写入标记。不支持数据类型。| 标记地址 = '<地址>', 数据类型 = <类型>。 49

无法写入标记。此标记数据类型非法。| 标记地址 = '<地址>', 数据类型 = <类型>。 49

无法写入标记。地址超出了当前 CIP 连接大小。| 标记地址 = '<地址>'。 53

无法写入标记。控制器标记数据类型未知。| 标记地址 = '<地址>', 数据类型 = <类型>。 48

无法写入标记。原生标记大小不匹配。| 标记地址 = '<地址>'。 51

无法写入地址。内存无效。| 标记地址 = '<地址>'。 52

警

详细信息。| IP = '<地址>', 供应商 ID = <供应商>, 设备类型 = <类型>, 产品代码 = <代码>, 修订版本 = <修订版本>, 产品名称 = '<名称>', 产品序列号 = <数字>。 59

頭

项目选项 20, 29

恊

性能 20, 30, 33

性能统计信息和调优 33

辩

选项 20, 30

增

已扩展 21, 30

已扩展模式 40

已压缩 21, 30

已压缩模式 40

弹

优化通信 22, 32

优化应用程序 32

颐

预定义术语标记 39

轼

远程 CPU 9

德

允许子组 19, 28

轰

运行时间 = <数字> (秒) 57

嚙

在读取标记时，设备所返回的数据超过预期。请验证地址，包括元素偏移和该偏移中的所有维度。| 标记地址 = '<地址>'。 53

在读取块时，设备所返回的数据超过预期。请验证地址，包括元素偏移和该偏移中的所有维度。| 块大小 = <数字> (元素)，起始标记地址 = '<地址>'。 53

在同步时检测到无效或损坏的控制器项目。将立即重试同步。 42

在同步时检测到无效或损坏的控制器项目。请稍后重试。 59

在同步时检测到项目下载。请稍后重试。 59

锶

长整型 34

輻

转发开放请求 8

膊

自动标记数据库生成;长控制器程序 & 标记名称 40

自动降级 17, 27

媯

字 34

字符 34

字符串 22, 32, 34, 37

字节 34

逕

遵循标签指定的扫描速率 16, 26