

目录

Modbus Serial 驱动程序	2
目录	3
Modbus Serial 驱动程序	5
概述	5
设置	6
通道配置 API 命令	6
通道属性 - 常规	7
通道属性 - 串行通信	7
通道属性 - 写入优化	9
通道属性 - 高级	10
通道属性 - 通信序列化	10
设备配置 API 命令	11
设备属性 - 常规	12
操作模式	13
设备属性 - 扫描模式	14
设备属性 - 以太网封装	14
设备属性 - 定时	15
设备属性 - 自动降级	15
设备属性 - 标记生成	16
设备属性 - 设置	17
设备属性 - 块大小	20
设备属性 - 变量导入设置	20
设备属性 - 框架	21
设备属性 - 错误处理	21
设备属性 - 冗余	22
自动标记数据库生成	23
数据类型说明	24
地址说明	25
Modbus 寻址	25
Magnetek GPD 515 驱动器寻址	28
Elliott 流量计算机寻址	28
Daniels S500 流量计算机寻址	29
Dynamic Fluid Meter 寻址	29
Omni 流量计算机寻址	30
Omni 自定义数据包	33
Omni 原始数据存档	35
Omni 文本报告	39
Omni 文本存档	41
功能代码说明	42
统计信息项	42
事件日志消息	44

块范围中的地址错误。 地址范围 = <起始> 到 <结束>。	44
不良数组。 数组范围 = <开始> 至 <结束>。	44
块地址响应异常代码。 地址范围 = <起始> 到 <结束>, 异常代码 = <代码>。	44
无法写入地址, 设备响应异常代码。 地址 = '<地址>', 异常代码 = <代码>。	45
无法从地址读取, 设备响应异常代码。 地址 = '<地址>', 异常代码 = <代码>。	45
标记导入由于内存资源不足而失败。	45
标记导入期间发生文件异常。	45
解析导入文件中的记录时出错。 记录编号 = <数字>, 字段 = <名称>。	45
针对导入文件中的记录截断了说明。 记录编号 = <编号>。	46
导入的标记名称无效, 且已经更改。 标记名称 = '<标记>', 更改后的标记名称 = '<标记>'。	46
由于不支持数据类型, 因此无法导入标记。 标记名称 = '<标记>', 不支持的数据类型 = '<类型>'。	46
由于内存分配问题, 无法读取 Omni 文本缓冲区。	46
指定日期范围内没有可用的 Omni 文本归档数据。	46
向 Omni 文本报告的写入遭到截断。 报告编号 = <数字>。	47
由于数据包编号限制, 无法读取 Omni 文本报告。 报告编号 = <数字>。	47
写入失败。已超出最大路径长度。 标记地址 = '<地址>', 最大长度 = <数字>。	47
将 Omni 文本数据写入文件时出错。 标记地址 = '<地址>', 原因 = '<原因>'。	47
Omni 文本输出文件无法打开。 标记地址 = '<地址>', 原因 = '<原因>'。	47
无法写入地址。响应中存在意外字符。 标记地址 = '<地址>'。	48
无法从地址读取。响应中存在意外字符。 标记地址 = '<地址>'。	48
无法读取块地址。响应中存在意外字符。 地址范围 = <起始> 到 <结束>。	48
正在从文件导入标记数据库。 文件名 = '<名称>'。	48
错误掩码定义	48
Modbus 异常代码	49
索引	50

Modbus Serial 驱动程序

帮助版本 [1.078](#)

目录

[概述](#)

什么是 Modbus Serial 驱动程序？

[设置](#)

如何配置使用此驱动程序的通道和设备？

[自动标记数据库生成](#)

如何配置 Modbus Serial 驱动程序的标记？

[数据类型说明](#)

此驱动程序支持哪些数据类型？

[地址说明](#)

如何对 Modbus 设备上的数据位置进行寻址？

[事件日志消息](#)

Modbus Serial 驱动程序会产生哪些消息？

概述

Modbus Serial 驱动程序 提供将 Modbus 串行设备连接到 OPC 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。其适用于支持 Modbus RTU 协议的串行设备。已开发的 Modbus Serial 驱动程序 可支持各种 Modbus RTU 兼容设备。

设置

通信协议

Modbus RTU 协议

支持的通信参数

波特率：1200、2400、9600 和 19200

奇偶校验：奇、偶和无

数据位：8

停止位：1 和 2

支持的设备

- Modbus 兼容设备
- Elliott 流量计算机
- Magnetek GPD 515 驱动器
- Omni Flow Computer
- Daniel S500 流量计算机
- 动态流量计 (DFM) SFC3
- TSXCUSBMBP USB 适配器

支持的最大数量

此驱动程序支持的最大通道数量为 2048。

所支持的最大设备数量为每通道 255 个。

● **注意：**并非每个设备都支持列出的所有配置。

通道配置 API 命令

以下命令使用“配置 API”服务来定义通道。

常规属性

`common.ALLTYPES_NAME` * 必需的参数。

● **注意：**更改此属性将导致 API 端点 URL 发生改变。

`common.ALLTYPES_DESCRIPTION`

`servermain.MULTIPLE_TYPES_DEVICE_DRIVER` * 必需的参数

`servermain.CHANNEL_DIAGNOSTICS_CAPTURE`

以太网通信属性

`servermain.CHANNEL_ETHERNET_COMMUNICATIONS_NETWORK_ADAPTER_STRING`

高级属性

`servermain.CHANNEL_NON_NORMALIZED_FLOATING_POINT_HANDLING` * 必需的参数

写入优化

`servermain.CHANNEL_WRITE_OPTIMIZATIONS_METHOD`

`servermain.CHANNEL_WRITE_OPTIMIZATIONS_DUTY_CYCLE`

● **另请参阅：**服务器帮助系统“配置 API 服务”部分。

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 - 串行通信

串行通信属性可用于串行驱动程序，且随驱动程序、连接类型以及所选选项的不同而变化。以下是可能具有的属性的超集。

单击跳转至下列其中一个部分：[“连接类型”](#)、[“串行端口设置”](#)或[“以太网设置”](#)以及[“操作行为”](#)。

● **注意**: 服务器全天在线运行时，可以随时更改这些属性。由于对这些属性进行更改后可能会暂时中断通信，因此可通过“用户管理器”来限制对服务器功能的访问权限。

属性组		
常规		
串行通信		
写优化		
高级		
通信序列化		
链接设置		
	连接类型	
	物理媒体	COM 端口
	已共享	否
	串行端口设置	
	COM ID	2
	波特率	19200
	数据位	8
	奇偶性	无
	停止位	1
	流量控制	无
	操作行为	
	报告通信错误	启用

连接类型

“物理媒体”：选择用于数据通信的硬件设备的类型。选项包括“COM 端口”、“无”、“调制解调器”和“以太网封装”。默认选项为 COM 端口。

- “无”：选择“无”表示没有物理连接，此时将显示**“无通信的操作”**部分。
- **“COM 端口”**：选择“COM 端口”可显示和配置**“串行端口设置”**部分。
- **“调制解调器”**：当用电话线进行通信时，选择“调制解调器”，并在**“调制解调器设置”**部分中对该选项进行配置。
- **“以太网封装”**：选择是否将“以太网封装”用于通信，此时将显示**“以太网设置”**部分。
- **“共享”**：验证是否已将连接正确标识为与其他通道共享当前配置。为只读属性。

串行端口设置

“COM ID”：指定在与分配给通道的设备进行通信时要使用的通信 ID。有效范围为 1 至 999。默认值为 1。

“波特率”：指定用于配置选定通信端口的波特率。

“数据位”：指定每个数据字的数据位数。选项包括 5、6、7 或 8。

“奇偶性”：指定数据的奇偶类型。选项包括“奇”、“偶”或“无”。

“停止位”：指定每个数据字的停止位数。选项包括 1 或 2。

“流量控制”：选择 RTS 和 DTR 控制线的使用方式。在与一些串行设备进行通信时需要流量进行控制。选项包括：

- **“无”**：此选项不会切换或添加控制线。
- **“DTR”**：当通信端口打开并保持开启状态时，此选项将添加 DTR 线路。
- **“RTS”**：此选项指定，如果字节适用于传输，则 RTS 线路为高电平。在发送所有缓冲字节后，RTS 线路变为低电平。这通常用于 RS232/RS485 转换器硬件。
- **“RTS, DTR”**：此选项是 DTR 和 RTS 的组合选项。
- **“始终 RTS”**：当通信端口打开并保持开启状态时，此选项将添加 RTS 线路。
- **“RTS 手动”**：此选项将基于为“RTS 线路控制”输入的定时属性添加 RTS 线路。该选项仅在驱动程序支持手动 RTS 线路控制 (或属性共享且至少有一个通道属于提供此类支持的驱动程序) 时可用。“RTS 手动”添加**“RTS 线路控制”**属性时具有如下选项：
 - **“上升”**：该属性用于指定在数据传输前 RTS 线路上升为高电平所需的时间量。有效范围为 0 至 9999 毫秒。默认值为 10 毫秒。
 - **“下降”**：该属性用于指定在数据传输后 RTS 线路保持高电平的时间量。有效范围为 0 至 9999 毫秒。默认值为 10 毫秒。
 - **“轮询延迟”**：该属性用于指定通信轮询的延迟时间量。有效范围为 0 到 9999。默认值为 10 毫秒。

提示：在使用双线 RS-485 时，通信线路上可能会出现“回波”。由于此类通信不支持回波抑制，因此建议禁用回波或使用 RS-485 转换器。

操作行为

- **“报告通信错误”：**启用或禁用报告低级通信错误。启用时，如果出现低级错误，则会将其发布到“事件日志”。禁用时，即使正常请求失败，也不会发布这些相同的错误。默认设置为“启用”。
- **“关闭空闲连接”：**当通道上的客户端不再引用任何标记时，选择关闭通道连接。默认设置为“启用”。
- **“关闭前空闲时间”：**指定在移除所有标记后服务器在关闭 COM 端口前所等待的时间。默认值为 15 秒。

以太网设置

注意：不是所有的串行驱动程序都支持以太网封装。若此组未出现，则无法支持相关功能。

如果要同与以太网终端服务器相连的串行设备进行通信，则可通过“以太网封装”来实现。终端服务器本质上是将以太网上的 TCP/IP 消息转换为串行数据的虚拟串行端口。消息转换完毕后，用户可将支持串行通信的标准设备连接到终端服务器。必须对终端服务器的串行端口进行正确配置，以满足所连串行设备的要求。有关详细信息，请参阅服务器帮助中的“如何使用以太网封装”。

- **“网络适配器”：**用于指示此通道中以太网设备绑定的网络适配器。选择要绑定的网络适配器，或者允许操作系统选择默认项。
 某些特定的驱动程序可能会显示其他“以太网封装”属性。有关详细信息，请参阅“通道属性 - 以太网封装”。

调制解调器设置

- **“调制解调器”：**指定用于通信的已安装调制解调器。
- **“连接超时”：**指定读取或写入失败前建立连接所等待的时间。默认值为 60 秒。
- **“调制解调器属性”：**配置调制解调器硬件。单击该选项后，将打开供应商特定的调制解调器属性。
- **“自动拨号”：**启用自动拨打电话簿中的条目。默认设置为“禁用”。有关详细信息，请参阅服务器帮助中的“调制解调器自动拨号”。
- **“报告通信错误”：**启用或禁用报告低级通信错误。启用时，如果出现低级错误，则会将其发布到“事件日志”。禁用时，即使正常请求失败，也不会发布这些相同的错误。默认设置为“启用”。
- **“关闭空闲连接”：**当通道上的客户端不再引用任何标记时，选择关闭调制解调器连接。默认设置为“启用”。
- **“关闭前空闲时间”：**指定在移除所有标记后服务器在关闭调制解调器连接前所等待的时间。默认值为 15 秒。

无通信的操作

- **“读取处理”：**选择要在请求显式设备读取时执行的操作。选项包括“忽略”和“失败”。“忽略”不执行任何操作；“失败”会为客户端提供一条指示失败的更新信息。默认设置为“忽略”。

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。

通道属性 - 通信序列化

服务器的多线程架构使通道能够与设备并行通信。尽管这十分高效, 但在存在物理网络限制 (如以太网无线电) 的情况下, 通信可能会进行序列化。通信序列化将限制在虚拟网络中每次仅使用一个通道进行通信。

术语“虚拟网络”是指使用同一管线进行通信的通道和相关设备的集合。例如, 以太网无线电管线是主无线电。使用同一主无线电的所有通道均与同一虚拟网络相关联。通道能够以“循环”方式轮流进行通信。默认情

况下，通道在向另一通道传递通信前，可处理一个事务。一个事务中可包括一个或多个标记。如果控制通道包含的设备未响应请求，则在事务超时之前，通道无法释放控制权。这会导致虚拟网络中其他通道的数据更新延迟。

属性组	☐ 通道级别设置	
常规	虚拟网络	无
以太网通信	每周期的事务数	1
写优化	☐ 全局设置	
高级	网络模式	负载已平衡
通信序列化		

通道级别设置

“虚拟网络”: 此属性可指定通道的通信序列化模式。选项包括“无”和“网络 1 - 网络 500”。默认值为“无”。选项说明如下:

- **“无”**: 此选项禁用通道的通信序列化。
- **“网络 1 - 网络 500”**: 此选项可指定分配通道的虚拟网络。

“每周期的事务数” 此属性可指定通道中可能发生的单一分块/非分块读/写事务的数量。当通道可以进行通信时，将尝试该事务数。有效范围为 1 到 99。默认值为 1。

全局设置

- **“网络模式”**: 此属性用于控制委派通道通信的方式。在**“负载平衡”**模式下，每个通道可以逐一轮流进行通信。在**“优先级”**模式下，通道可以根据以下规则 (优先级由高到低) 进行通信:
 - 具有待处理写入操作的通道具有最高优先级。
 - 具有待处理显式读取操作 (通过内部插件或外部客户端接口) 的通道的优先级基于读取的优先级。
 - 扫描读取和其他定期事件 (特定于驱动程序)。

默认设置为“负载平衡”，这并影响所有虚拟网络和通道。

● 依赖于主动响应的设备不应置于虚拟网络中。在必须进行通信序列化的情况下，建议启用“自动降级”。

由于驱动程序的数据读取和写入方式的差异 (如单一、分块或非分块事务)，可能需要调整应用程序的“每周期的事务数”属性。执行此操作时，请考虑以下因素:

- 必须从每个通道读取多少标记?
- 数据写入各个通道的频率如何?
- 通道使用串行驱动程序还是以太网驱动程序?
- 驱动程序是读取单独请求中的标记还是读取块中的多个标记?
- 设备的定时属性 (如请求超时和 x 次连续超时后失败) 是否针对虚拟网络通信媒介进行了优化?

设备配置 API 命令

以下命令使用“配置 API”服务来定义通道。

常规属性

common.ALLTYPES_NAME

common.ALLTYPES_DESCRIPTION

servermain.DEVICE_CHANNEL_ASSIGNMENT

servermain.MULTIPLE_TYPES_DEVICE_DRIVER

```
servermain.DEVICE_MODEL  
servermain.DEVICE_ID_STRING  
servermain.DEVICE_DATA_COLLECTION  
servermain.DEVICE_SIMULATED
```

扫描模式

```
servermain.DEVICE_SCAN_MODE * 必需的参数  
servermain.DEVICE_SCAN_MODE_RATE_MS  
servermain.DEVICE_SCAN_MODE_RATE_MS  
servermain.DEVICE_SCAN_MODE_PROVIDE_INITIAL_UPDATES_FROM_CACHE
```

自动降级

```
servermain.DEVICE_AUTO_DEMOTION_ENABLE_ON_COMMUNICATIONS_FAILURES  
servermain.DEVICE_AUTO_DEMOTION_DEMOTE_AFTER_SUCESSIVE_TIMEOUTS  
servermain.DEVICE_AUTO_DEMOTION_PERIOD_MS  
servermain.DEVICE_AUTO_DEMOTION_DISCARD_WRITES
```

标记生成

```
servermain.DEVICE_TAG_GENERATION_ON_STARTUP * 必需的参数  
servermain.DEVICE_TAG_GENERATION_DUPLICATE_HANDLING * 必需的参数  
servermain.DEVICE_TAG_GENERATION_GROUP  
servermain.DEVICE_TAG_GENERATION_ALLOW_SUB_GROUPS
```

● **提示:** 要调用“自动生成标记”，请向设备上的 **TagGeneration** 服务端点发送正文为空的 PUT。
● **另请参阅:** 有关详细信息，请参阅“服务”帮助。

定时

```
servermain.DEVICE_CONNECTION_TIMEOUT_SECONDS  
servermain.DEVICE_REQUEST_TIMEOUT_MILLISECONDS  
servermain.DEVICE_RETRY_ATTEMPTS  
servermain.DEVICE_INTER_REQUEST_DELAY_MILLISECONDS
```

● **另请参阅:** 服务器帮助系统“配置 API 服务”部分。

设备属性 - 常规

一个设备代表通信通道上的单一目标。如果驱动程序支持多个控制器，则用户必须为每个控制器输入一个设备 ID。

Property Groups	<input type="checkbox"/> Identification	
General	Name	
Scan Mode	Description	
	Channel Assignment	
	Driver	
	Model	
	ID Format	Decimal
	ID	2

标识

名称: 此属性用于指定设备的名称。此为用户定义的逻辑名称，最长可达 256 个字符，并且可以用于多个通道。

● **注意:** 尽管描述性名称通常是不错的选择，但浏览 OPC 服务器的标记空间时，一些 OPC 客户端应用程序的显示窗口可能不够大。设备名称和通道名称也成为浏览树信息的一部分。OPC 客户端中，通道名称和设备名称的组合将显示为“通道名称.设备名称”。

● **有关详细信息，请参阅服务器帮助中的“如何为通道、设备、标记和标记组正确命名”。**

“说明”: 有关此设备的用户定义信息。

● 在这些属性中，有很多属性 (包括“说明”) 具有关联的系统标记。

“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: 此属性指定与此 ID 关联的特定设备类型。下拉菜单中的内容取决于正在使用的通信驱动程序类型。驱动程序不支持的型号将被禁用。如果通信驱动程序支持多个设备型号，则只有当设备未与任何客户端应用程序连接时，才能改变型号的选择。

● **注意:** 如果通信驱动程序支持多种型号，则用户应将型号选择与物理设备进行匹配。如果下拉列表菜单中未显示该设备，则选择与目标设备最相近的型号。一些驱动程序支持名为“开放式”的型号选择，该选择使用户无需了解目标设备的具体信息即可进行通信。有关详细信息，请参阅驱动程序帮助文档。

ID: 此属性用于指定设备中特定于驱动程序的工作站或节点。输入 ID 类型取决于正在使用的通信驱动程序。对于许多通信驱动程序而言，ID 是一个数值。支持数字 ID 的驱动程序使用户能够输入格式可更改的数值，以适应应用需要或所选通信驱动程序特点。默认情况下，该格式由驱动程序设置。选项包括十进制、八进制和十六进制。

● **注意:** 如果驱动程序基于以太网，或者支持非常规工作站或节点名称，则可使用设备的 TCP/IP 地址作为设备 ID。TCP/IP 地址包含四个由句点分隔的值，每个值的范围在 0 至 255 之间。某些设备 ID 基于字符串。根据不同驱动程序，也可以在 ID 字段中配置其他属性。有关详细信息，请参阅驱动程序的帮助文档。

操作模式

Property Groups	<input type="checkbox"/> Identification	
General	<input type="checkbox"/> Operating Mode	
Scan Mode	Data Collection	Enable
	Simulated	No

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	☐ 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 - 以太网封装

“以太网封装”旨在为通过以太网与终端服务器相连的串行设备提供通信。终端服务器实质上是虚拟串行端口。终端服务器会将以太网上的 TCP/IP 消息转换为串行数据。消息转换为串行形式后，用户可将支持串行通信的标准设备连接到终端服务器。

● 有关详细信息，请参阅服务器帮助中的“如何使用以太网封装”。

● “以太网封装”对于驱动程序来说是透明的；配置其余属性时，应与直接连接本地串行端口上的设备一样。

属性组	☐ 以太网设置	
常规	IP 地址	255.2.255.245
扫描模式	端口	2101
以太网封装	协议	TCP/IP
定时		

“IP 地址”(IP Address)：此属性用于输入与设备连接的终端服务器的四字段 IP 地址。IP 指定为 YYY.YYY.YYY.YYY。YYY 指定 IP 地址；每个 YYY 字节应在 0 至 255 的范围内。每个串行设备都可以有其自己的 IP 地址；但是，如果多个设备与单个终端服务器进行多点通信时，则这些设备可能使用相同的 IP 地址。

“端口”(Port)：在连接远程终端服务器时，可使用此属性配置以太网端口。

“协议”(Protocol): 此属性用于选择 TCP/IP 或 UDP 通信。该选择取决于正在使用的终端服务器的性质。默认协议选项为 TCP/IP。有关可用协议的详细信息，请参阅终端服务器的帮助文档。

● **注意**

1. 服务器全天在线运行时，可以随时更改这些属性。使用“用户管理器”可限制对服务器功能的访问权限，并防止操作员更改属性。
2. 有效的 IP 地址范围大于 (>) 0.0.0.0 且小于 (<) 255.255.255.255。

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。

Property Groups	Communication Timeouts	
General	Connect Timeout (s)	3
Scan Mode	Connect Attempts	3
Ethernet Encapsulation	Request Timeout (ms)	1000
Timing	Attempts Before Timeout	3
Auto-Demotion	Timing	
Tag Generation	Inter-Request Delay (ms)	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。

“连接尝试次数”: 此属性 (主要针对基于以太网封装的驱动程序) 限制驱动程序与目标设备之间尝试连接的次数。如果达到此限制，则连接请求失败。“连接超时”属性指定连接尝试的时间间隔。有效的尝试次数范围为 1 到 10。默认值为 3 次尝试。如果驱动程序不支持此设置，则此设置将被禁用。

“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 - 设置

Property Groups		
General		
Scan Mode		
Timing		
Auto-Demotion		
Tag Generation		
Settings		
Block Sizes		
Variable Import Settings		
Framing		
Error Handling		
Redundancy		
	Data Access	
	Zero-Based Addressing	Enable
	Zero-Based Bit Addressing	Enable
	Holding Register Bit Writes	Disable
	Modbus Function 06	Enable
	Modbus Function 05	Enable
	Data Encoding	
	Modbus Byte Order	Enable
	First Word Low	Enable
	First DWord Low	Enable
	Modicon Bit Order	Disable
	Treat Longs as Decimals	Disable

数据访问

“从零开始的寻址”: 如果设备的地址编号约定从 1 开始而不是零, 可在定义设备参数时指定该值。默认情况下, 当将帧构建为与 Modbus 设备通信时, 用户输入的地址将减一。如果设备不遵守此约定, 请选择禁用。默认行为遵守 Modicon PLC 的约定。

“从零开始的位寻址”: 在寄存器内, 允许字内位的内存类型可作为布尔型引用。寻址符号为 `<address>.<位>`, 其中, `<位>` 表示字内的位数。此选项提供了两种对给定字内的位进行寻址的方法: 从零开始或从 1 开始。从零开始意味着第一位从 0 开始 (范围 = 0-15); 从 1 开始意味着第一位从 1 开始 (范围 = 1-16)。

“保持寄存器位掩码”: 写入保存寄存器中的位位置时, 驱动程序应仅修改正在使用的位。某些设备支持一个特殊命令, 来操纵寄存器中的单个位 (功能代码十六进制 `0x16` 或十进制 `22`)。如果设备不支持此功能, 则驱动程序必须执行“读取/修改/写入”操作, 以确保只更改单个位。启用后, 驱动程序将使用功能代码 `0x16`, 与单个寄存器写入操作的设置无关。禁用后, 驱动程序将根据是否对单个寄存器写入操作选择了 Modbus 功能代码 `06` 来使用功能代码 `0x06` 或 `0x10`。默认设置为禁用状态。

● **注意**: 如果禁用 Modbus 字节顺序, 则命令中发送的掩码的字节顺序将为 Intel 字节顺序。

“Modbus 功能代码 06”: 此驱动程序支持 Modbus 协议功能, 以将保持寄存器数据写入到目标设备。在大多数情况下, 驱动程序可基于正在写入的寄存器数在功能代码 `06` 和 `16` 之间进行切换。当写入单个 16 位寄存器时, 驱动程序通常使用 Modbus 功能代码 `06`。当将 32 位值写入两个寄存器时, 驱动程序将使用 Modbus 功能代码 `16`。对于标准 Modicon PLC, 使用这些功能中的任何一个都不是问题。但是, 使用 Modbus 协议的第三方设备很多, 并且许多第三方设备仅支持 Modbus 功能代码 `16` 来写入保持寄存器。默认情况下将启用此选项, 允许驱动程序根据需要在 `06` 和 `16` 之间切换。如果设备要求所有写入都只使用 Modbus 功能代码 `16`, 则请禁用此选择。

● **注意**: 对于字内位写入, “保存寄存器位掩码”属性优先于此选项。如果启用了“保存寄存器位掩码”, 则将使用功能代码 `0x16`, 而无需考虑此属性。如果未启用, 则功能代码 `0x06` 或 `0x10` 将用于位内字写入。

“Modbus 函数 05”: 此驱动程序支持 Modbus 协议功能, 可将输出线圈数据写入目标设备。在大多数情况下, 驱动程序可基于正在写入的线圈数在这两个功能之间进行切换。当写入单个线圈时, 驱动程序将使用 Modbus 功能代码 `05`。当写入线圈数组时, 驱动程序将使用 Modbus 功能代码 `15`。对于标准 Modicon PLC, 使用这些功能都不是问题。但是, 使用 Modbus 协议的第三方设备很多, 并且许多第三方设备无论输出线圈数为多少都仅支持使用 Modbus 功能代码 `15` 写入输出线圈。默认情况下将启用此选项, 允许驱动程序根据需要在 `05` 和 `15` 之间切换。如果设备要求所有写入都只使用 Modbus 功能代码 `15`, 则请禁用此选择。

数据编码

“Modbus 字节顺序”: 设置每个寄存器/16 位值的数据编码。使用此选项, 可将字节顺序从默认的 Modbus 字节排序更改为 Intel 字节排序。默认设置为启用状态, 这是 Modbus 兼容设备的常用设置。如果设备使用 Intel 字节排序, 请禁用此属性以读取 Intel 格式的数据。

● **注意**: 此设置不适用于 Omni 型号。它始终使用 Modbus 字节顺序。

“首字排序为低”: 设置 32 位值的数据编码和 64 位值的双字。Modbus 设备中的两个连续寄存器地址用于 32 位数据类型。驱动程序可根据此选项读取第一个字作为 32 位值的低位字或高位字读取。默认设置为启用状态, 即首字排序为低, 以遵守 Modicon Modsoft 编程软件的约定。

● **注意:** 此设置不适用于 Omni 型号。它始终使用 Modbus 字节顺序。

“首双字排序为低”: 设置 64 位值的数据编码。Modbus 设备中的四个连续寄存器地址用于 64 位数据类型。驱动程序可将第一个双字作为 64 位值的低位双字或高位双字读取。默认设置为启用状态，即首双字排序为低，以遵守 32 位数据类型的默认约定。

● **注意:** 此设置不适用于 Omni 型号。它始终使用 Modbus 字节顺序。

“Modicon 位顺序”: 启用后，驱动程序将反转读取和写入至寄存器的位顺序，以遵守 Modicon Modsoft 编程软件的约定。例如，启用此选项后，写入地址 40001.0/1 会影响设备的位 15/16。默认情况下，此选项处于禁用状态 (已禁用)。

对于以下示例，第 1 位至第 16 位表示 0-15 位或 1-16 位，具体取决于驱动程序在寄存器中使用从零开始还是从 1 开始的位寻址。

MSB = 最高有效位

LSB = 最低有效位

启用的 Modicon 位顺序

MSB								LSB							
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

禁用的 Modicon 位顺序

MSB								LSB							
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

“将长整型作为十进制处理”: 启用后，驱动程序会将双精度无符号长整型和双字型数据类型编码和解码为介于 0 到 99999999 之间的值。此格式指定了每个字表示介于 0 到 9999 之间的值。超过指定范围的值读取不受限制，但行为未定义。所有读取值都将使用以下公式进行解码： $[读取值] = HighWord * 10000 + LowWord$ 。大于 99999999 的写入值将限制为最大值。所有写入值都使用以下公式进行编码： $原始数据 = [写入值] / 10000 + [写入值] \% 10000$ 。

● **有关设置的提示**

数据类型	Modbus 字节顺序	首字排序为低	首双字排序为低
字、短整型、BCD	适用	不适用	不适用
浮点型、双字型、长整型、LBCD	适用	适用	不适用
双精度	适用	适用	适用

如果需要，请使用以下信息和设备的文档来确定数据编码选项的正确设置。

默认设置对于大多数 Modbus 设备都是可接受的。

数据编码选项	数据编码	
“Modbus 字节顺序”	高字节 (15..8)	低字节 (7..0)
“Modbus 字节顺序”	低字节 (7..0)	高字节 (15..8)
“首字排序为低”	高位字 (31..16) 64 位数据类型中双字的高位字 (63..48)	低位字 (15..0) 64 位数据类型中双字的低位字 (47..32)
“首字排序为低”	低位字 (15..0) 64 位数据类型中双字的低位字 (47..32)	高位字 (31..16) 64 位数据类型中双字的高位字 (63..48)
“首双字排序为低”	高位双字 (63..32)	低位双字 (31..0)
“首双字排序为低”	低位双字 (31..0)	高位双字 (63..32)

设备属性 - 块大小

Property Groups	<input type="checkbox"/> Coils	
General	Output Coils	32
Scan Mode	Input Coils	32
Timing	<input type="checkbox"/> Registers	
Auto-Demotion	Internal Registers	32
Tag Generation	Holding Registers	32
Settings	<input type="checkbox"/> Block Sizes	
Block Sizes	Block Read Strings	Enable
Variable Import Settings		
Framing		
Error Handling		
Redundancy		

“线圈”

“输出线圈”: 指定输出块大小, 以位为单位。可一次性读取线圈 8 到 2000 个点 (位)。较大的块意味着在单个请求中可从设备中读取更多的点。可以减小块大小以从设备内的非连续位置读取数据。默认设置为 32。

“输入线圈”: 指定输入块大小, 以位为单位。可一次性读取线圈 8 到 2000 个点 (位)。较大的块意味着在单个请求中可从设备中读取更多的点。可以减小块大小以从设备内的非连续位置读取数据。默认设置为 32。

寄存器

“内部寄存器”: 指定内部寄存器块大小, 以位为单位。可一次性读取 1 至 125 个 16 位标准 Modbus 寄存器。较大的块意味着在单个请求中可从设备中读取更多的寄存器值。可以减小块大小以从设备内的非连续位置读取数据。默认设置为 32。

“保持寄存器”: 指定保持寄存器块大小, 以位为单位。可一次性读取 1 至 125 个 16 位标准 Modbus 寄存器。较大的块意味着在单个请求中可从设备中读取更多的寄存器值。可以减小块大小以从设备内的非连续位置读取数据。默认设置为 32。

警告: 如果寄存器块大小值设置为大于 120, 并且将 32 位或 64 位数据类型用于任何标记, 则会出现“块中的地址错误”错误。为了防止此错误, 请将块大小值减小到 120。

块大小

“块读取字符串”: 启用通常单独读取的字符串标记的组/块读取。字符串标记根据所选块大小分组在一起。只能对 Modbus 型号字符串标记执行块读取。默认设置为禁用状态。

设备属性 - 变量导入设置

“变量导入设置”参数指定要用于“自动标记数据库生成”的变量导入文件的位置。

有关 Modbus 驱动程序的 CSV 文件的详细信息, 请参阅[创建 Kepware Modbus 驱动程序的 CSV 文件](#)。

Property Groups	<input type="checkbox"/> Variable Import Settings	
General	Variable Import File	*.bt
Scan Mode	Include Descriptions	Enable
Timing		
Auto-Demotion		
Tag Generation		
Settings		
Block Sizes		
Variable Import Settings		
Framing		
Error Handling		
Redundancy		

“变量导入文件”: 此参数用于浏览用于“自动标记数据库生成”的变量导入文件的确切位置。

“包括说明”: 启用后, 如果文件中存在, 则使用导入的标记说明。

有关配置自动标记数据库生成功能 (以及如何创建变量导入文件) 的详细信息, 请参阅[自动标记数据库生成](#)。

设备属性 - 框架

对于 32 位数据类型的数组, 行数乘以列数再乘以 2 所得结果不能超过块大小。

Property Groups	<input type="checkbox"/> Framing	
Ethernet Encapsulation	Modbus TCP Framing	Disable
Settings	Leading Bytes	0
Block Sizes	Trailing Bytes	0
Variable Import Settings		
Framing		
Error Handling		
Redundancy		

框架

“Modbus TCP 组帧”: 如果驱动程序应当使用带 MBAP 头的 Modbus TCP 帧, 请选择**“启用”**。默认设置为禁用状态。

提示: 在与本机 Modbus TCP 设备通信时, 应启用此设置。

“前导字节”: 指定要附加到 Modbus 响应开头的字节数。值范围介于 0 到 8 之间。

“尾随字节”: 指定要附加到 Modbus 响应结尾的字节数。值范围介于 0 到 8 之间。

设备属性 - 错误处理

错误处理参数确定如何处理来自设备的错误。

Property Groups	<input checked="" type="checkbox"/> Error Handling	
General	Deactivate Tags on Illegal Address	Enable
Scan Mode	Reject Repeated Messages	Disable
Timing		
Auto-Demotion		
Tag Generation		
Settings		
Block Sizes		
Variable Import Settings		
Framing		
Error Handling		
Redundancy		

“取消激活非法地址上的标记”: 启用后, 驱动程序会在设备读取某个数据块的响应过程中返回 Modbus 异常代码 2 (非法地址) 或 3 (点数之类的非法数据) 时, 停止轮询该数据块。当禁用时, 驱动程序继续轮询该数据块。默认设置为启用状态。

“拒绝重复消息”: 启用后, 驱动程序会将重复消息解释为无效响应, 并重试该请求。默认设置为启用状态。当禁用时, 驱动程序需要重复的消息。

● **注意**: 某些消息继电器设备会将 Modbus 请求传回到驱动程序。

设备属性 - 冗余

属性组	<input checked="" type="checkbox"/> 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。

自动标记数据库生成

Modbus Serial 驱动程序 可使用“自动标记数据库生成”，从而能够自动创建可访问设备梯形图程序所用数据点的标记。尽管某些情况下可以在设备中查询构建标记数据库所需的信息，但此驱动程序必须改用“**变量导入文件**”。变量导入文件可以使用 Concept 和 ProWORX 设备编程应用程序生成。

创建变量导入文件

导入文件必须采用以分号分隔的 .TXT 格式，这是诸多设备编程应用程序的默认导出文件格式。

● 有关创建变量导入文件的具体信息，请参阅技术说明“为 Modbus 驱动程序创建 CSV 文件”。

除了所有支持“自动标记数据库生成”的驱动程序通用的基本设置外，此驱动程序还需要其他设置。专用设置包括变量导入文件的名称和位置，此信息可以在“设备向导”的“变量导入设置”步骤中指定，也可以稍后通过选择“**设备属性**”|“**变量导入设置**”进行指定。

● 有关详细信息，请参阅[变量导入设置](#)。

服务器配置

可自定义“自动标记数据库生成”以满足应用程序的需求。主要控制选项可以在“设备向导”的“数据库创建”步骤中进行设置，也可以稍后通过选择“**设备属性**”|“**标记生成**”进行设置。

操作

标记生成可在服务器项目启动时自动启动，也可在其他时间手动启动，具体取决于配置。“事件日志”将显示标记生成过程的开始时间、处理变量导入文件时发生的任何错误以及过程的完成时间。

数据类型说明

数据类型	说明
布尔型	单个位
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
双字型	无符号 32 位值 位 0 是低位 位 31 是高位
长整型	有符号 32 位值 位 0 是低位 位 30 是高位 位 31 是符号位
BCD	两个字节封装的 BCD 值的范围是 0 至 9999。对于超出此范围的值，未定义行为。
LBCD	四个字节封装的 BCD 值的范围是 0 至 99999999。对于超出此范围的值，未定义行为。
字符串	空终止 ASCII 字符串 支持 Modbus 型号，包括由高到低、由低到高的字节排序、8 字节和 16 字节 OMNI 流量计算机字符串数据。
双精度*	64 位浮点值 驱动程序将四个连续寄存器解释为双精度值，方法将最后两个寄存器作为高位双字，将前两个寄存器作为低位双字。
双精度示例	如果将寄存器 40001 指定为双精度数，则寄存器 40001 的位 0 将是 64 位数据类型的位 0，寄存器 40004 的位 15 将是 64 位数据类型的位 63。
浮点型*	32 位浮点值 驱动程序将两个连续寄存器解释为单精度值，方法是将最后一个寄存器作为高位字，将第一个寄存器作为低位字。
浮点型示例	若将寄存器 40001 指定为浮点型，则寄存器 40001 的位 0 将是 32 位数据类型的位 0，寄存器 40002 的位 15 将是 32 位数据类型的位 31。

*说明假定为默认，按首双字排序为低处理 64 位数据类型，按首字排序为低处理 32 位数据类型。

地址说明

地址规范因所使用的型号而异。从以下列表选择一个链接，以获取相关型号的具体地址信息。

[Modbus 寻址](#)

[Magnetek GPD 515 驱动器寻址](#)

[Elliott 流量计算机寻址](#)

[Daniels S500 流量计算机寻址](#)

[动态流量计寻址](#)

[Omni 流量计算机寻址](#)

[统计信息](#)

● 另请参阅：[功能代码说明](#)

Modbus 寻址

动态定义标记的默认数据类型以**粗体**显示。功能代码均以十进制显示。

● 有关详细信息，请参阅[“功能代码说明”](#)。

5 位寻址与 6 位寻址

在 Modbus 寻址中，地址的第一位指定了主表格。其余位表示设备的数据项。数据项的最大值是一个两字节的无符号整型值 (65,535)。驱动程序内部需要六位数字来表示整个地址表格和项。请务必注意，许多 Modbus 设备可能不支持数据项的整个范围。为避免在输入此类设备的地址时出现混乱，此驱动程序将根据地址字段中输入的内容来“填充”地址 (添加数字)。如果主表格类型后跟最多 4 位数字 (示例：4 x、4xx、4xxx 或 4xxxx)，则地址仍可保留或使用额外的零填充到五 (5) 位数字。如果主表格类型后跟五 (5) 位数字 (示例：4xxxxx)，则该地址不变。在内部，输入为 41、401、4001、40001 或 400001 的地址是指定主表格类型 4 和数据项 1 的地址的所有对等表示。

主表格	说明
0	输出线圈
1	输入线圈
3	内部寄存器
4	保持寄存器

以十进制格式表示的 Modbus 寻址

地址	范围	数据类型	访问*	功能代码
输出线圈	000001-065536	布尔型	读/写	01、05、15
输入线圈	100001-165536	布尔型	只读	02
内部寄存器	300001-365536 300001-365535 300001-365533 3xxxxx.0/1-3xxxxx.15/16****	字、短整型、BCD 浮点型、双字型、长整型、LBCD 双精度 布尔型	只读	04
内部寄存器为字符串型 (字节排序由高到低)	300001.2H-365536.240H .位是字符串长度，范围 2 到 240 个字节。	字符串**	只读	04
内部寄存器为字符串型 (字节排序由低到高)	300001.2L-365536.240L	字符串**	只读	04

地址	范围	数据类型	访问*	功能代码
	.位是字符串长度, 范围 2 到 240 个字节。			
保持寄存器	400001-465536 400001-465535 400001-465533 4xxxx.0/1-4xxxx.15/16***	字、短整型、BCD 浮点型、双字型、长整型、LBCD 双精度 布尔型	读/写	03、06、16 03、06、16、22
字符串型保存寄存器 (字节排序由高到低)	400001.2H-465536.240H .位是字符串长度, 范围 2 到 240 个字符。	字符串**	读/写	03、16
字符串型保存寄存器 (字节排序由低到高)	400001.2L-465536.240L .位是字符串长度, 范围 2 到 240 个字符。	字符串**	读/写	03、16

* 必须对地址附加前缀 "W" (例如: W40001) 才能将读/写地址设置为只写。这可防止驱动程序读取指定地址的寄存器。客户端读取“只写”标记的所有尝试都会导致获得指定地址的上次成功写入值。如果没有任何成功写入, 则客户端将接收到数值/字符串值 0/NULL 作为初始值。

警告: 将“只写”标记的客户端访问权限设置为“只读”将导致无法写入这些标记, 且客户端将始终接收到数值/字符串值 0/NULL。

**有关详细信息, 请参阅“字符串支持”。

***有关详细信息, 请参阅“设置”中的“从零开始的位寻址”。

十六进制 Modbus 寻址

地址	范围	数据类型	访问	功能代码
输出线圈	H000001-H0FFFF	布尔型	读/写	01、05、15
输入线圈	H100001-H1FFFF	布尔型	只读	02
内部寄存器	H300001-H310000 H300001-H3FFFF H300001-H3FFFD H3xxxx.0/1-H3xxxx.F/10*	字、短整型、BCD 浮点型、双字型、长整型、LBCD 双精度 布尔型	只读	04
内部寄存器为字符串型 (字节排序由高到低)	H300001.2H-H3FFFF.240H. .位是字符串长度, 范围 2 到 240 个字符。	字符串**	只读	04
内部寄存器为字符串型 (字节排序由低到高)	H300001.2L-H3FFFF.240L. .位是字符串长度, 范围 2 到 240 个字符。	字符串**	只读	04
保持寄存器	H400001-H410000 H400001-H4FFFF H400001-H4FFFD H4xxxx.0/1-H4xxxx.F/10*	字、短整型、BCD、浮点型、双字型、长整型、LBCD、双精度、布尔型	读/写	03、06、16 03、06、16、22
字符串型保存寄存	H400001.2H-H4FFFF.240H.	字符串**	读	03、

地址	范围	数据类型	访问	功能代码
器 (字节排序由高到低)	位是字符串长度, 范围 2 到 240 个字符。		/写	16
字符串型保存寄存器 (字节排序由低到高)	H400001.2L-H4FFFF.240L. 位是字符串长度, 范围 2 到 240 个字符。	字符串**	读/写	03、16

● *有关详细信息, 请参阅“[设置](#)”中的“从零开始的位寻址”。

● **有关详细信息, 请参阅“[字符串支持](#)”。

字符串支持

Modbus 型号支持以 ASCII 字符串的形式读写保持寄存器内存。当将保持寄存器用于字符串数据时, 每个寄存器包含 ASCII 数据的两个字节。定义字符串时, 可以选择指定寄存器中 ASCII 数据的顺序。字符串的长度可以为 2 到 240 个字节, 并代替位数输入。长度必须输入为偶数。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

字符串示例

1. 对于从 40200 开始、长度为 100 字节并采用由高到低字节排序的字符串, 要对其进行寻址, 请输入: "40200.100H"。
2. 对于从 40500 开始、长度为 78 字节并采用由低到高字节排序的字符串, 要对其进行寻址, 请输入: "40500.78L"。

● **注意:** 字符串长度可能受设备允许的写入请求字符数上限的限制。如果在使用字符串标记时, “服务器事件”窗口显示错误消息“无法写入设备 <address> 上的地址 <device>: 设备响应异常代码 3”, 则意味着设备不接受字符串的长度。如果可能, 请缩短字符串。

常规地址示例

1. 对第 255 输出线圈的寻址, 将以十进制“0255”或十六进制“H0FF”进行。
2. 一些文档通过功能代码和位置引用 Modbus 地址。例如, 功能代码 3; 位置 2000 将以“42000”或“H47D0”进行寻址。居前的“4”表示保持寄存器或功能代码 3。
3. 一些文档通过功能代码和位置引用 Modbus 地址。例如, 设置功能代码 5 位置 100 将以“0100”或“H064”进行寻址。居前“0”表示输出线圈或功能代码 5。将 1 或 0 写入此地址将会设置或重置线圈。

数组支持

除了布尔型和字符串之外, 所有数据类型的内部和保持寄存器位置都支持数组。对于输入和输出线圈 (布尔数据类型), 同样支持数组。有关数组寻址的方法有两种。以下示例使用保留寄存器位置:

4xxxx [行数][列数]

4xxxx [列数] 此方法假定行数等于 1。

对于数组, 行数乘以列数所得结果不能超过已分配给设备的寄存器/线圈类型的块大小。对于 32 位数据类型的寄存器数组, 行数乘以列数再乘以 2 所得结果不能超过块大小。

封装线圈地址类型

封装线圈地址类型允许访问多个连续的线圈作为模拟值。此功能只有在轮询模式下才适用于输入线圈和输出线圈。唯一有效的数据类型是“字”。语法为:

输出线圈: 0xxxxx#nn 字读/写

输入线圈: 1xxxxx#nn 字只读

其中, xxxx 是第一个线圈的地址 (允许十进制和十六进制值), nn 是封装为模拟值的线圈数 (1-16, 仅限十进制)。

位顺序使开始地址为模拟值的 LSB (最低有效位)。

Magnetek GPD 515 驱动器寻址

此表提供了 Magnetek GPD 515 驱动器中可用数据的常规范围。有关如何使用 Modbus RTU 寻址访问特定驱动器参数的详细信息，请参阅 **Magnetek Modbus RTU 技术手册** (部件号 TM4025)。在所有情况下，字母 H (用于表示十六进制寻址) 均应在所需地址之前。动态定义标记的默认数据类型以**粗体**显示 (若适用)。

Magnetek GPD 515 寻址十六进制格式

地址	范围	数据类型	访问
命令寄存器位级访问	H40001-H4000F H4xxxx.0/1-H4xxxx.F/10*	字、短整型、布尔型	读/写
监视器寄存器位级访问	H40010-H4001A H4xxxx.0/1-H4xxxx.F/10*	字、短整型、布尔型	只读
驱动程序参数寄存器 (仅限监视器) 位级访问	H40020-H40097 H4xxxx.0/1-H4xxxx.F/10*	字、短整型、布尔型	只读
驱动程序参数寄存器位级访问	H40100-H4050D H4xxxx.0/1-H4xxxx.F/10*	字、短整型、布尔型	读/写
特殊寄存器	H4FFDD ACCEPT H4FFFD ENTER	字、短整型	只写

● 有关详细信息，请参阅“[设置](#)”中的“从零开始的位寻址”(Zero-Based Bit Addressing)。

示例

要访问驱动程序的操作状态，地址 02BH，请输入以下地址：H4002B。

● **注意：**将 Magnetek 设备添加到 OPC 服务器项目时，用户必须确保禁用设置“从零开始的寻址”。如果未正确设置此参数，Modbus RTU 驱动程序会将所有 Magnetek 地址偏移 1。

数组支持

除了布尔型之外，所有数据类型的支持寄存器位置都支持数组。数组寻址方法有两种。以下示例使用保留寄存器位置：

4xxxx [行数] [列数]

4xxxx [列数] 此方法假定行数等于 1。

行数乘以列数所得结果不能超过已分配给设备的寄存器类型的块大小。

Elliott 流量计算机寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

地址	范围	数据类型	访问
输出线圈	000001-065536	布尔型	读/写
输入线圈	100001-165536	布尔型	只读
内部寄存器	300001-365536 300001-365535 3xxxxx.0/1-3xxxxx.15/16*	字、短整型、BCD 浮点型、双字型、长整型、LBCD 布尔型	只读
保持寄存器	400001-465536 400001-465535 4xxxxx.0/1-4xxxxx.15/16*	字、短整型、BCD** 浮点型、双字型、长整型、LBCD 布尔型	读/写

● 有关详细信息，请参阅“[设置](#)”中的“从零开始的位寻址”(Zero-Based Bit Addressing)。

**地址范围 405001 到 405315 和 407001 到 407315 是 32 位寄存器。405001 到 405315 范围内的地址使用默认的长整型数据类型。

407001 到 407315 范围内的地址使用默认的浮点型数据类型。由于这些地址寄存器为 32 位，所以只允许使用浮点型、双字型、长整型或 LBCD 数据类型。不允许使用数组。

数组支持

除了布尔型之外，所有数据类型的内部和保持寄存器位置都支持数组。数组寻址方法有两种。以下示例使用保留寄存器位置：

4xxxx [行数] [列数]

4xxxx [列数] 此方法假定“行数”等于 1。

行数乘以列数所得结果不能超过已分配给设备的寄存器类型的块大小。对于 32 位数据类型的数组，行数乘以列数再乘以 2 所得结果不能超过块大小。

Daniels S500 流量计算机寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。功能代码均以十进制显示。

有关详细信息，请参阅[功能代码说明](#)。

地址	十六进制范围	十进制范围	数据类型	功能代码	访问
总计	000-0FF	4096-4351	双精度	03	只读
计算/测量的变量	100-24F	4352-4687	浮点型	03、16	读/写
计算常量	250-28F	4688-4751	浮点型	03、16	读/写
小键盘默认值	290-2AF	4752-4783	浮点型	03、16	读/写
警报和缩放常量	2B0-5FF	4784-5631	浮点型	03、16	读/写
状态/控制	700-7FF	5888-6143	布尔型	01、05	读/写
警报	800-FFF	6144-8191	布尔型	02	只读

Dynamic Fluid Meter 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

Dynamic Fluid Meter 寻址十进制格式

地址	范围	数据类型	访问
保持寄存器 (16 位)	400000-407000 400000-406999	字、短整型、BCD 浮点型、双字型、长整型、LBCD	读/写
	408001-465535 408001-465534	字、短整型、BCD 浮点型、双字型、长整型、LBCD	
	4xxxxx.0/1-4xxxxx.15/16*	布尔型	
保持寄存器 (32 位)	407001-408000	浮点型	读/写
保持寄存器为字符串型 (由高到低字节排序)	400000.2H-407000.240H 408001.2H-465535.240H .Bit 是字符串长度，范围介于 2 到 240 个字符之间。	字符串	读/写
保持寄存器为字符串型 (字节排序由低到高)	400000.2L-407000.240L 408001.2L-465535.240L .Bit 是字符串长度，范围介于 2 到 240 个字符之间。	字符串	读/写

有关详细信息，请参阅[“设置”](#)中的“从零开始的位寻址”(Zero-Based Bit Addressing)。

Dynamic Fluid Meter 寻址十六进制格式

地址	范围	数据类型	访问
保持寄存器 (16 位)	H400000-H401B58 H400000-H401B57 H401F41-H40FFFF H401F41-H40FFFE H4xxxxx.0/1-H4xxxxx.F/10*	字、短整型、BCD 浮点型、双字型、长整型、LBCD 字、短整型、BCD 浮点型、双字型、长整型、LBCD 布尔型	读/写
保持寄存器 (32 位)	H401B59-H401F40	浮点型	读/写
保持寄存器为字符串型 (由高到低字节排序)	H400000.2H-H401B58.240H H401F41.2H-H40FFFF.240H .Bit 是字符串长度, 范围介于 2 到 240 个字符之间。	字符串	读/写
保持寄存器为字符串型 (字节排序由低到高)	H400000.2L-H401B58.240L H401F41.2L-H0FFFF.240L .Bit 是字符串长度, 范围介于 2 到 240 个字符之间。	字符串	读/写

● 有关详细信息, 请参阅“[设置](#)”中的“从零开始的位寻址”(Zero-Based Bit Addressing)。

● **注意:** 该驱动程序要求 Dynamic Fluid Meter 模型的所有的地址都以 "4" 开始。Dynamic Fluid Meter 文档有时可能并未明文规定使用 4。例如, 用户可能会看到对“地址 3001 的单元 ID”的参考。该值在服务器中必须作为 "403001" 处理。

字符串支持

Dynamic Fluid Meter 型号支持以 ASCII 字符串的形式读写保持寄存器内存。当将保持寄存器用于字符串数据时, 每个寄存器包含 ASCII 数据的两个字节。定义字符串时, 可以选择指定寄存器中 ASCII 数据的顺序。字符串的长度可以为 2 到 240 个字节, 并代替位数输入。长度必须输入为偶数。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

字符串示例

1. 对于从 40200 开始、长度为 100 字节并采用由高到低字节排序的字符串, 要对其进行寻址, 请输入: "40200.100H"。
2. 对于从 40500 开始、长度为 78 字节并采用由低到高字节排序的字符串, 要对其进行寻址, 请输入: "40500.78L"。

● **注意:** 字符串长度可能受设备允许的写入请求字符数上限的限制。如果在使用字符串标记时, “服务器事件”窗口显示错误消息“无法写入设备 <设备> 上的地址 <地址>: 设备响应异常代码 3”, 则意味着设备不接受字符串的长度。如果可能, 请尝试缩短字符串。

Omni 流量计算机寻址

动态定义标记的默认数据类型以**粗体**显示。

地址	范围	数据类型	访问
数字 I/O 点	1001-1024	布尔型	读/写
可编程布尔点	1025-1088	布尔型	读/写
仪表运行状态和报警点	1n01-001n59 1n76-1n99 n = 仪表运行数	布尔型	读/写
Micro Motion 报警状态点	1n60-1n75 n = 仪表运行数	布尔型	读/写

地址	范围	数据类型	访问
用户 Scratchpad 布尔点	1501-1599 1601 -1649	布尔型	读/写
用户暂存单点	1650-1699	布尔型	读/写
命令布尔点/变量	1700-1798	布尔型	读/写
仪表站报警及状态点	1801-1899	布尔型	读/写
校准仪警报及状态点	1901-1967	布尔型	读/写
仪表总计翻转标志	2n01-2n37 n = 仪表运行数	布尔型	读/写
杂项仪表站报警及状态	2601-2623	布尔型	读/写
工作站总计翻转标志	2801-2851	布尔型	读/写
工作台总计十进制分辨率	2852-2862 2865-2999	布尔型	读/写

16 位整数数据地址	范围	数据类型	访问
自定义数据包 #1	3001-3040	短整型、字、BCD	读/写
自定义数据包 #2	3041-3056	短整型、字、BCD	读/写
自定义数据包 #3	3057-3096	短整型、字、BCD	读/写
杂项 16 位整数数据	3097-3099 3737-3799 3875-3899	短整型、字、BCD	读/写
仪表运行 16 位整数数据	3n01-3n52 n = 仪表运行数	短整型、字、BCD	读/写
Scratchpad 16 位整数数据	3501-3599	短整型、字、BCD	读/写
用户显示 1	3601-3608	短整型、字、BCD	读/写
用户显示 2	3609-3616	短整型、字、BCD	读/写
用户显示 3	3617-3624	短整型、字、BCD	读/写
用户显示 4	3625-3632	短整型、字、BCD	读/写
用户显示 5	3633-3640	短整型、字、BCD	读/写
用户显示 6	3641-3648	短整型、字、BCD	读/写
用户显示 7	3649-3656	短整型、字、BCD	读/写
用户显示 8	3657-3664	短整型、字、BCD	读/写
访问原始数据存档记录	3701-3736	短整型、字、BCD	读/写
仪表工作站 16 位整数数据	3800-3842	短整型、字、BCD	读/写
仪表 #1 批序列	3843-3848	短整型、字、BCD	读/写
仪表 #2 批序列	3849-3854	短整型、字、BCD	读/写
仪表 #3 批序列	3855-3860	短整型、字、BCD	读/写
仪表 #4 批序列	3861-3866	短整型、字、BCD	读/写
流量计算机时间/日期	3867-3874	短整型、字、BCD	读/写
校准仪 16 位整数数据	3901-3999	短整型、字、BCD	读/写

8 字符 ASCII 字符串数据	范围	数据类型	访问
仪表运行 ASCII 数据	4n01-4n39 n = 仪表运行数	字符串	读/写
Scatch Pad ASCII 数据	4501-4599	字符串	读/写
用户显示定义变量	4601 -4640	字符串	读/写
站辅助输入变量	4707-4710	字符串	读/写
仪表站 ASCII 数据	4801-4851	字符串	读/写

8 字符 ASCII 字符串数据	范围	数据类型	访问
仪表 #1 批次 ID	4852-4863	字符串	读/写
仪表 #2 批次 ID	4864-4875	字符串	读/写
仪表 #3 批次 ID	4876-4887	字符串	读/写
仪表 #4 批次 ID	4888-4899	字符串	读/写
校准仪 ASCII 字符串数据	4901-4942	字符串	读/写

32 位整数数据	范围	数据类型	访问
仪表运行 32 位整数数据	5n01-5n99 n = 仪表运行数	长整型、双字型、LBCD、浮点型	读/写
Scratchpad 32 位整数数据	5501-5599	长整型、双字型、LBCD、浮点型	读/写
工作台 32 位整数数据	5801-5818	长整型、双字型、LBCD、浮点型	读/写
仪表 #1 批次大小	5819-5824	长整型、双字型、LBCD、浮点型	读/写
仪表 #2 批次大小	5825-5830	长整型、双字型、LBCD、浮点型	读/写
仪表 #3 批次大小	5831-5836	长整型、双字型、LBCD、浮点型	读/写
仪表 #4 批次大小	5837-5842	长整型、双字型、LBCD、浮点型	读/写
额外的 32 位仪表运行数据	5843-5899	长整型、双字型、LBCD、浮点型	读/写
校准仪 32 位整数数据	5901-5973	长整型、双字型、LBCD、浮点型	读/写
压缩校准仪 TDVOL/TDFMP 脉冲	5974-5999	长整型、双字型、LBCD、浮点型	读/写

32 位 IEEE 浮点数据	范围	数据类型	访问
保留的数据	6001-7000	浮点型、长整型、双字、LBCD	读/写
数字到模拟输出	7001-7024	浮点型、长整型、双字、LBCD	读/写
用户变量	7025-7088	浮点型、长整型、双字、LBCD	读/写
可编程蓄电池	7089-7099	浮点型、长整型、双字、LBCD	读/写
仪表运行数据	7n01 - 7n99 n = 仪表运行数	浮点型、长整型、双字、LBCD	读/写
Scratchpad 数据	7501-7599	浮点型、长整型、双字、LBCD	读/写
PID 控制数据	7601-7623	浮点型、长整型、双字、LBCD	读/写
杂项仪表运行数据	7624-7699	浮点型、长整型、双字、LBCD	读/写
杂项变量	7701-7799	浮点型、长整型、双字、LBCD	读/写
仪表工作站数据	7801-7899	浮点型、长整型、双字、LBCD	读/写
校准仪数据	7901-7918	浮点型、长整型、双字、LBCD	读/写
校准仪配置数据	7919-7958	浮点型、长整型、双字、LBCD	读/写
前一次核验数据	7959-7966	浮点型、长整型、双字、LBCD	读/写
核验过程中数据遭拒	7967-7990	浮点型、长整型、双字、LBCD	读/写
核验运行数据	7991-8050	浮点型、长整型、双字、LBCD	读/写
核验平均数据	8051-8079	浮点型、长整型、双字、LBCD	读/写
核验运行 - 主仪表数据	8080-8199	浮点型、长整型、双字、LBCD	读/写
核验系列数据	8200-8223	浮点型、长整型、双字、LBCD	读/写
仪表核验数据	8224-8230	浮点型、长整型、双字、LBCD	读/写
大量核验数据	8231-8500	浮点型、长整型、双字、LBCD	读/写
杂项仪表运行 #1	8501-8599	浮点型、长整型、双字、LBCD	读/写
杂项仪表运行 #2	8601-8699	浮点型、长整型、双字、LBCD	读/写
杂项仪表运行 #3	8701-8799	浮点型、长整型、双字、LBCD	读/写
杂项仪表运行 #4	8801-8899	浮点型、长整型、双字、LBCD	读/写
工作站提前批平均数据	8901-8999	浮点型、长整型、双字、LBCD	读/写

16 位整数配置数据	范围	数据类型	访问
仪表运行 #1	13001-13013	短整型、字、BCD	读/写
仪表运行 #2	13014-13026	短整型、字、BCD	读/写
仪表运行 #3	13027-13039	短整型、字、BCD	读/写
仪表运行 #4	13040-13052	短整型、字、BCD	读/写
校准仪配置	13053-13073	短整型、字、BCD	读/写
常规流配置	13074-13084	短整型、字、BCD	读/写
串行端口配置	13085-13128	短整型、字、BCD	读/写
PID 配置	13129-13160	短整型、字、BCD	读/写
PLC 数据	13161-13299	短整型、字、BCD	读/写
对等设置	13300-13499	短整型、字、BCD	读/写
原始数据存档	13500-13999	短整型、字、BCD	读/写

16 字符 ASCII 字符串数据	范围	数据类型	访问
流量计算机配置	14001-14499	字符串	读/写

32 位整数数据	范围	数据类型	访问
流量计算机配置	15001-16999	长整型、双字型、LBCD、浮点型	读/写

32 位 IEEE 浮点数据	范围	数据类型	访问
流量计算机配置	17001-18999	浮点型、长整型、双字、LBCD	读/写

支持的扩展 Omni 类型

[自定义数据包](#)

[原始数据存档](#)

[文本报告](#)

[文本存档](#)

Omni 自定义数据包

Omni 流量计算机允许用户将各种内存范围映射到单个数据结构中，然后通过单个高效读取命令进行读取。这些数据结构被称为“自定义数据包”。

数据包配置

每个自定义数据包最多可以包含 20 组数据点。每个组由其开始索引和数据点数定义。自定义数据包的总大小不得超过 250 个字节。下面列出了用于定义自定义数据包的地址。

自定义数据包 1 (地址 1)

3001 组 1 - 开始索引
 3002 组 1 - 点数
 至
 3039 组 20 - 开始索引
 3040 组 20 - 点数

自定义数据包 2 (地址 201)

3041 组 1 - 开始索引
 3042 组 1 - 点数
 至
 3055 组 20 - 开始索引
 3056 组 20 - 点数

自定义数据包 3 (地址 401)

3057 组 1 - 开始索引
 3058 组 1 - 点数
 至

3095 组 20 - 开始索引
3096 组 20 - 点数

● **注意:** 数据将作为 16 位寄存器从设备返回。数字 I/O 必须映射至 16 位的块。

自定义数据包地址语法

可以创建标记，以访问自定义数据包内给定偏移量处的数据。地址语法如下。默认数据类型以**粗体**显示。

地址	范围	数据类型	访问
Cpn_o	n = 数据包编号 (1-3) o = 字偏移 (0-125)	字、 短整型 、BCD、双字型、长整型、LBCD、浮点型、字符串	只读
Cpn_o.b	n = 数据包编号 (1-3) o = 字偏移 (0-125) b = 位数 (0/1-15/16)*	布尔型	只读

● 有关详细信息，请参阅[“设置”](#)中的“从零开始的位寻址”。

● **注意:**

1. 仅支持 8 个字符的 ASCII 字符串数据。
2. 组配置中包含 16 个字符的 ASCII 字符串数据地址时，会将数据读取为两个 8 字符 ASCII 字符串数据项。

示例

定义要映射到以下内容的“自定义数据包 #1”:

- 16 位数字 I/O (1001-1016)。
- “计量器运行”1 批次数据的十五个 32 位整数 (5101-5115)。
- “模拟输出”的十二个 32 位浮点数 (7001-7012)。
- “计量器运行”的四个 8 字符 ASCII 字符串 (4101-4104)。
- “计量器工作站”的六个 8 字符 ASCII 字符串 (4808-4813)。
- “流量配置”数据的两个 16 字符 ASCII 字符串 (14001-14002)。

● **注意:** 总共有 222 个字节。自定义数据包配置寄存器将具有以下值:

```
3001 = 1001
3002 = 16
3003 = 5101
3004 = 15
3005 = 7001
3006 = 12
3007 = 4101
3008 = 4
3009 = 4808
3010 = 6
3011 = 14001
3012 = 2
```

用于访问数字 I/O 数据的标记将具有以下地址 (所有 16 个值都包含在字 0 中):

```
CP1_0.0 (自定义数据包 1 的字 0, 位 0 - 映射到 1009)
CP1_0.1 (自定义数据包 1 的字 0, 位 1 - 映射到 1010)
...
CP1_0.6 (自定义数据包 1 的字 0, 位 6 - 映射到 1015)
CP1_0.7 (自定义数据包 1 的字 0, 位 7 - 映射到 1016)
CP1_0.8 (自定义数据包 1 的字 0, 位 8 - 映射到 1001)
CP1_0.9 (自定义数据包 1 的字 0, 位 9 - 映射到 1002)
```

```
...
CP1_0.14 (自定义数据包 1 的字 0, 位 14 - 映射到 1007)
CP1_0.15 (自定义数据包 1 的字 0, 位 15 - 映射到 1008)
```

用于访问“计量器运行”1 批次数据的标记将具有以下地址 (每个 32 位值使用 2 个字):

CP1_1 (自定义数据包 1 的字 1 - 映射到 5101)
 CP1_3 (自定义数据包 1 的字 3 - 映射到 5102)
 ...
 CP1_29 (自定义数据包 1 的字 29 - 映射到 5115)

用于访问“模拟输出”数据的标记将具有以下地址 (每个 32 位值使用 2 个字):

CP1_31 (自定义数据包 1 的字 31 - 映射到 7001)
 CP1_33 (自定义数据包 1 的字 33 - 映射到 7002)
 ...
 CP1_53 (自定义数据包 1 的字 53 - 映射到 7012)

用于访问“计量器运行”8 字符 ASCII 字符串数据的标记将具有以下地址 (每个字符串值使用 4 个字):

CP1_55 (自定义数据包 1 的字 55 - 映射到 4101)
 ...
 CP1_67 (自定义数据包 1 的字 67 - 映射到 4104)

用于访问“计量器工作站”8 字符 ASCII 字符串数据的标记将具有以下地址 (每个字符串值使用 4 个字):

CP1_71 (自定义数据包 1 的字 71 - 映射到 4808)
 ...
 CP1_91 (自定义数据包 1 的字 91 - 映射到 4813)

用于访问“流量配置”16 字符 ASCII 字符串数据的标记将具有以下地址 (每个字符串值使用 4 个字):

CP1_95 (自定义数据包 1 的字 95 - 映射到 14001 字符 1-8)
 CP1_99 (自定义数据包 1 的字 99 - 映射到 14001 字符 9-16)
 CP1_103 (自定义数据包 1 的字 103 - 映射到 14002 字符 1-8)
 CP1_107 (自定义数据包 1 的字 107 - 映射到 14002 字符 9-16)

Omni 原始数据存档

Omni 流量计算机可以配置为将各种内存范围映射到单个数据结构中，然后在触发时将该结构存储在存档中。用户最多可以配置十个存档。“警报”和“审计”数据还有两个附加的固定格式存档。每个存档都是一个循环缓冲区，其中每个新记录将替换最早的记录。

“记录配置”和“检索”

用户可以配置原始数据存档 1 至 10 的记录结构。存档 11 和 12 具有固定格式，分别包含“警报”和“审计”数据。

原始数据存档的完整讨论，请参阅全技术简报 96073。

每条记录最多可以包含十六组数据点。每个组由其开始索引和数据点数定义。下面列出了用于定义存档记录的地址。记录的总大小不得超过 250 个字节。设备将前 6 个字节用于日期和时间戳数据，而将剩下的 244 个字节用于原始数据。每条记录都有其自己的布尔型触发器。当触发器从低电平变为高电平时，存储数据。

在一组开始索引之前，可以更改原始数据存档的组或触发器中的点数，且存档必须中止。必须在设备中设置“允许存档配置标志”。请注意，这可能导致重新初始化设备中的数据存档，包括所有原始数据存档和文本存档。

13920“存档运行”- 0 = 停止，1 = 开始

13921“重新配置存档”- 0 = 不允许配置更改，1 = 允许配置更改

此驱动程序可用于一次读取“原始数据存档”中的一条记录。要读取记录，首先在“请求记录”寄存器中写入所需的记录索引。此值设置后，用户可以读取带有“RA”标记的记录。用户应确保指定的记录索引不超过该存档允许的最大记录数。如果“更新的最后一条记录”值为零，则自上次初始化以来，存档中未保存任何记录。

原始数据存档 1 (地址 701)

13500 组 1 - 开始索引

13501 组 1 - 点数

至

13530 组 16 - 开始索引

13531 组 16 - 点数

13900 触发器布尔型

3701 最大记录数

3702 更新的最后一条记录

3703 请求的记录

原始数据存档 2 (地址 702)

13540 组 1 - 开始索引

13541 组 1 - 点数

至

13570 组 16 - 开始索引

13571 组 16 - 点数

13901 触发器布尔型

3704 最大记录数

3705 更新的最后一条记录

3706 请求的记录

原始数据存档 3 (地址 703)

13580 组 1 - 开始索引

13581 组 1 - 点数

至

13610 组 16 - 开始索引

13611 组 16 - 点数

13902 触发器布尔型

3707 最大记录数

3708 更新的最后一条记录

3709 请求的记录

原始数据存档 4 (地址 704)

13620 组 1 - 开始索引

13621 组 1 - 点数

至

13650 组 16 - 开始索引

13651 组 16 - 点数

13903 触发器布尔型

3710 最大记录数

3711 更新的最后一条记录

3712 请求的记录

原始数据存档 5 (地址 705)

13660 组 1 - 开始索引

13661 组 1 - 点数

至

13690 组 16 - 开始索引

13691 组 16 - 点数

13904 触发器布尔型

3713 最大记录数

3714 更新的最后一条记录

3715 请求的记录

原始数据存档 6 (地址 706)

13700 组 1 - 开始索引

13701 组 1 - 点数

至

13730 组 16 - 开始索引

13731 组 16 - 点数

13905 触发器布尔型

3716 最大记录数

3717 更新的最后一条记录

3718 请求的记录

原始数据存档 7 (地址 707)

13740 组 1 - 开始索引
13741 组 1 - 点数
至
13770 组 16 - 开始索引
13771 组 16 - 点数

13906 触发器布尔型

3719 最大记录数
3720 更新的最后一条记录
3721 请求的记录

原始数据存档 8 (地址 708)

13780 组 1 - 开始索引
13781 组 1 - 点数
至
13810 组 16 - 开始索引
13811 组 16 - 点数

13907 触发器布尔型

3722 最大记录数
3723 更新的最后一条记录
3724 请求的记录

原始数据存档 9 (地址 709)

13820 组 1 - 开始索引
13821 组 1 - 点数
至
13850 组 16 - 开始索引
13851 组 16 - 点数

13908 触发器布尔型

3725 最大记录数
3726 更新的最后一条记录
3727 请求的记录

原始数据存档 10 (地址 710)

13860 组 1 - 开始索引
13861 组 1 - 点数
至
13890 组 16 - 开始索引
13891 组 16 - 点数

13909 触发器布尔型

3728 最大记录数
3729 更新的最后一条记录
3730 请求的记录

原始数据存档 11 警报 (地址 711)

不可配置

3731 最大记录数
3732 更新的最后一条记录
3733 请求的记录

原始数据存档 12 存档 (地址 712)

不可配置

3734 最大记录数
3735 更新的最后一条记录
3736 请求的记录

注意: 数据将作为 16 位寄存器从设备返回。数字 I/O 必须映射至 16 位的块。

原始数据存档地址语法

可以创建标记，以访问原始数据存档记录内给定偏移量处的数据。地址语法如下。默认数据类型以**粗体**显示。

地址	范围	数据类型	访问
RAn_o	n = 存档编号 (1-12) o = 字偏移 (0-125)	字、 短整型 、BCD、双字型、长整型、LBCD、浮点型、字符串	只读
RAn_o.b	n = 存档编号 (1-12) o = 字偏移 (0-125) b = 位数 (0/1-15/16)*	布尔型	只读

● 有关详细信息，请参阅“[设置](#)”中的“从零开始的位寻址”。

● 注意:

1. 仅支持 8 个字符的 ASCII 字符串数据。
2. 组配置中包含 16 个字符的 ASCII 字符串数据地址时，会将数据读取为两个 8 字符 ASCII 字符串数据项。

时间戳格式

每条记录的前 6 个字节包含记录放入存档中的日期和时间。

字节	说明
1	月 (1-12)* 日 (1-31)
2	日 (1-31)* 月 (1-12)
3	年 (0-99)
4	小时 (0-23)
5	分钟 (0-59)
6	秒 (0-59)

*通过寄存器 3842 设置日期格式 (0 = dd/mm/yy, 1 = mm/dd/yy)。

警报/事件日志记录结构 (地址 711)

字段	数据类型	说明
1	3 字节日期	dd/mm/yy 或 mm/dd/yy。
2	3 字节时间	hh/mm/ss。
3	16 位整数	警报或事件的 Modbus 索引编号。
4	1 字节	警报类型。
5	1 字节	0 = 确定, 1 = 警报。
6	IEEE 浮点数	警报或事件发生时传感器变量的值。
7	32 位整数	警报或事件发生时的体积累加器。
8	32 位整数	警报或事件发生时的质量累加器。

警报类型

类型	说明
0	日志事件、声音蜂鸣器和 LCD 显示中由字段 #3 标识的位的任何边缘变化。
1	日志事件、声音蜂鸣器和 LCD 显示中由字段 #3 标识的位的上升沿变化。

2	事件日志中由字段 #3 标识的位的任何边缘变化。无蜂鸣器或 LCD 显示操作。
3	事件日志中由字段 #3 标识的位的上升沿变化。无蜂鸣器或 LCD 显示操作。

审计事件日志记录结构 (地址 712)

字段	数据类型	说明
1	3 字节日期	dd/mm/yy 或 mm/dd/yy。
2	3 字节时间	hh/mm/ss。
3	16 位整数	事件编号, 各事件的增量, 在 65535 处滚动。
4	16 位整数	更改的变量的 Modbus 索引。
5	IEEE 浮点数	更改旧值之前的数字变量值。
6	IEEE 浮点数	更改新值之后的数字变量值。
7	16 字符 ASCII	更改旧值之前的字符串变量值。
8	16 字符 ASCII	更改新值之后的字符串变量值。
9	32 位整数	更改时的体积累加器。
10	32 位整数	更改时的质量累加器。

● **注意:** 当更改的变量类型为字符串时, 字段 5 和 6 将设置为 0.0。当变量类型更改不是字符串时, 字段 7 和 8 将包含空字符。当字段 7 和 8 包含 8 字符字符串时, 将用空值填充其余的 8 个字符。

Omni 文本报告

Omni 流量计算机可以生成多种不同类型的文本报告。每份报告都可以由此驱动程序读取, 并作为字符串值发送到 OPC 客户端。

文本报告类型

从 Omni 流量计算机可以检索到多种报告类型。可以使用 "TR" 标记读取它们。报告类型如下所示。

自定义报告模板

9001 报告模板 - 快照/间隔
 9002 报告模板 - 批处理
 9003 报告模板 - 每天
 9003 报告模板 - 证明

之前的批处理报告

9101 批处理报告 - 最后一个
 9102 批处理报告 - 倒数第二个
 ...
 9108 批处理报告 - 倒数第八个

之前的证明报告

9201 证明报告 - 最后一个
 9202 证明报告 - 倒数第二个
 ...
 9208 证明报告 - 倒数第八个

之前的日常报告

9301 之前的日常报告 - 最后一个
 9302 之前的日常报告 - 倒数第二个
 ...
 9308 之前的日常报告 - 倒数第八个

最后一个快照报告

9401 最后一个本地快照/间隔报告

杂项报告缓冲区

9402 杂项报告缓冲区

文本报告地址语法

地址	范围	数据类型	访问
TRn TRn T (触发读取)	n = 报告地址 (9001-9402)	字符串	读/写

示例

要读取或写入快照报告模板 (地址 9001), 请创建地址为 "TR9001" 的标记。

● **注意:** 由于读取文本报告可能需要几秒钟的时间, 因此 "TR" 标记应在 OPC 客户端中保持非活动状态。或者, 改为使用触发的读数。在驱动程序读取或写入文本报告的同时, 无法读取或写入信道上的其他标记。

触发的文本报告读取

如上所述, 建议尽量将文本报告标记保持在非活动状态。已将触发的读取功能添加为备用功能, 这样文本报告标记就可以保持活动状态。它还通过辅助触发标记来控制何时实际进行设备读取。

触发的读取可能不会立即开始, 这取决于文本报告标记的更新循环中触发器设置的时间。读取尝试完成后, 驱动程序将清除触发状态。文本报告标记显示上次触发的读取尝试所产生的值以及数据质量。

文本报告读取触发地址语法

地址	范围	数据类型	访问
TRIG_TRn	n = 报告地址 (9001-9402)	布尔型	读/写

示例

要在触发器上读取最后一个批处理报告 (地址 9101), 请创建两个标记: 第一个是地址为 "TR9101 T" 的文本报告标记, 第二个是地址为 "TRIG_TR9101" 的文本报告读取触发标记。

● **注意:** 文本报告标记地址看起来像正常的文本报告地址, 后面跟一个空格以及表示“触发读取”的字母 "T"。为正常触发读取, 此处的 "T" 必须存在于地址中。

要触发读取, 请将触发标记值设置为“真”(非零)。读取尝试完成后, 驱动程序将触发器值设置为“假”(0)。如果读取成功, 则文本报告标记的数据质量良好。如果读取失败, 则文本报告标记的数据质量不佳且值为上次成功读取时的值。

将文本报告数据保存到磁盘

驱动程序可以将文本报告数据保存到磁盘。可以通过文本报告路径标记启用该功能。这些标记用于将文件路径字符串写入驱动程序的内存。每个报告类型都有其自己的路径缓冲区。成功读取文本报告后, 驱动程序会检查相关的路径缓冲区。如果存储在此的路径有效, 则驱动程序会将报告数据作为 ASCII 文本保存在该文件中。如果需要, 将创建文件。文件会在随后的文本报告读取中被覆盖。

路径缓冲区会在服务器启动时初始化为空字符串。除非在相关的路径缓冲区中保存了有效的路径, 否则驱动程序不会将文本报告数据写入文件。路径数据不是永久性的。每次重新启动服务器时, 都必须重写路径字符串。可随时更改路径值, 这样用户便可在每次读取时将数据保存到不同的文件中 (如果需要)。

路径字符串最长可达 255 个字符。

文本报告路径地址语法

地址	范围	数据类型	访问
PATH_TRn	n = 报告地址 (9001-9402)	字符串	读/写

示例

要读取最后一个批处理报告 (地址 9101) 并将结果保存至磁盘, 请创建两个标记: 第一个是地址为 "TR9101" 的文本报告标记, 第二个是地址为 "PATH_TR9101" 的路径标记。

要报告数据保存在名为 "LastBatch.txt" (将在文件夹 "C:\OmniData\BatchReports" 下创建) 的文件中, 请设置客户端以实现首先将 "C:\OmniData\BatchReports\LastBatch.txt" 写入路径标记。完成此操作后, 读取文本报告标记。如果在第一次读取文本报告之前未设置路径, 则驱动程序无法将数据保存到磁盘。

● **注意:** 要禁用此功能, 请将空字符串写入路径标记。

Omni 文本存档

Omni 流量计算机还可以将报告存储在存档中。此驱动程序可以从存档中读取一系列报告，并将其以字符串的形式发送到 OPC 客户端。

读取文本存档

在可以读取文本存档之前，必须在设备上进行两项设置：存档开始日期以及要检索的天数。这些 32 位整数分别位于地址 15128 和 15127。日期格式可以通过位于地址 3842 的值进行指定 (0 = dd/mm/yy、1 = mm/dd/yy)。设置完天数后，紧接着设备就开始准备数据。准备好读取数据时，天数值变为负值。设置完天数后，可以随时读取文档存档。驱动程序会等待值变为负值。

文本存档地址语法

地址	范围	数据类型	访问
TA TA T (触发读取)	不适用	字符串	只读

● **注意：**由于读取文本存档可能需要几分钟的时间，因此 "TA" 标记应在 OPC 客户端中保持非活动状态。或者，改为使用触发的读数。由于在服务器中无法将最大同步读取超时增加到足以读取典型的文档存档请求，因此只能以异步读取的方式读取标记。在读取文本存档的同时，无法读取或写入信道上的其他标记。

如果文本存档读取中途失败，用户应通过将 999 写入天数寄存器 (15127) 对设备的读取缓存进行重置，然后重复正常的文本存档读取过程。否则，驱动程序可能会获取所请求存档范围的第一部分。

触发的文本存档读取

建议尽量将文本存档标记保持在非活动状态。已将触发的读取功能添加为备用功能，这样文本存档标记就可以保持活动状态。它还通过辅助触发标记来控制何时实际进行设备读取。触发器值存储在驱动程序的内存中，并可以通过具有下述地址语法的标记进行读取和设置。

触发的读取可能不会立即开始，这取决于文本存档标记的更新循环中触发器设置的时间。读取尝试完成后，驱动程序将清除触发状态。文本存档标记显示上次触发的读取尝试所产生的值和数据质量。

文本存档读取触发地址语法

地址	范围	数据类型	访问
TRIG_TA	不适用	布尔型	读/写

示例

要在触发器上读取文本存档，请创建两个标记：第一个是地址为 "TA T" 的文本存档标记，第二个是地址为 "TRIG_TA" 的文本存档读取触发标记。用户必须创建开始日期和天数标记。

● **注意：**文本存档标记地址看起来像正常的文本存档地址，后面跟一个空格以及表示“触发读取”的字母 "T"。为正常触发读取，此处的 "T" 必须存在于地址中。

要触发读取，请将触发标记值设置为“真”(非零)。读取尝试完成后，驱动程序将触发器值设置为“假”(0)。如果读取成功，则文本存档标记的数据质量良好。如果读取失败，则文本存档标记的数据质量不佳且值为上次成功读取时的值。

将文本存档数据保存到磁盘

驱动程序可以将文本存档数据保存到磁盘。可以通过文本存档路径标记启用该功能。该标记用于将文件路径字符串写入驱动程序的内存。成功读取文本存档后，驱动程序会检查相关的路径缓冲区。如果存储在此的路径有效，则驱动程序会将文本存档数据作为 ASCII 文本保存在该文件中。如果需要，将创建文件。文件会在随后的文本存档读取中被覆盖。

路径缓冲区会在服务器启动时初始化为空字符串。除非在相关的路径缓冲区中保存了有效的路径，否则驱动程序不会将文本存档数据写入文件。路径数据不是永久性的。每次重启服务器时，用户必须重新写入路径字符串。可随时更改路径值，这样便可在每次读取时将数据保存到不同的文件中 (如果需要)。

路径字符串最长可达 255 个字符。

文本存档路径地址语法

地址	范围	数据类型	访问
PATH_TA	不适用	字符串	读/写

示例

要读取文本存档并将结果保存至磁盘，请创建两个标记：第一个是地址为 "TA" 的文本存档标记，第二个是地址为 "PATH_TA" 的路径标记。用户必须按上述方式创建开始日期和天数标记。

要将文本存档数据保存在名为 "TextArchive.txt" (将在文件夹 "C:\OmniData\ArchiveData" 下创建) 的文件中，请设置客户端以实现首先将 "C:\OmniData\ArchiveData\TextArchive.txt" 写入路径标记。完成此操作后，读取文本存档标记。如果在第一次读取文本存档之前未设置路径，则驱动程序无法将数据保存到磁盘。

● **注意：**要禁用此功能，请将空字符串写入路径标记。

功能代码说明

Modbus 寻址模型

十进制	十六进制	说明
01	0x01	读取线圈状态
02	0x02	读取输入状态
03	0x03	读取保持寄存器
04	0x04	读取内部寄存器
05	0x05	强制单个线圈
06	0x06	预设单个寄存器
15	0x0F	强制多个线圈
16	0x10	预设多个寄存器
22	0x16	已屏蔽的写入寄存器

Daniels S500 流量计算机寻址模型

十进制	十六进制	说明
01	0x01	读取线圈状态
02	0x02	读取输入线圈
03	0x03	读取保持寄存器
05	0x05	强制单个线圈
16	0x10	预设多个寄存器

统计信息项

统计信息项使用通过其他诊断信息收集默认情况下不收集的数据。要使用统计信息项，必须启用“通信诊断”。要启用“通信诊断”，在“项目视图”中右键单击通道，然后单击“属性”|“启用诊断”。或者，双击通道，然后选择“启用诊断”。

通道级统计信息项

通道级统计信息项的语法为 <通道>_Statistics。

● **注意：**通道级别的统计信息是设备级别中相同项的汇总。

项	数据类型	访问	说明
_CommFailures	双字型	读/写	通信失败 (或耗用的重试) 的总次数。
_ErrorResponses	双字型	读/写	收到有效错误响应的总数。
_ExpectedResponses	双字型	读/写	收到预期响应的总数。
_LastResponseTime	字符串	只读	收到最后一次有效响应的的时间。
_LateData	双字型	读/写	驱动程序标记的数据更新时间晚于预期 (基于指定的扫描速率) 的总次数。

项	数据类型	访问	说明
_MsgResent	双字型	读/写	作为重试发送的消息总数。
_MsgSent	双字型	读/写	初始发送的消息总数。
_MsgTotal	双字型	只读	发送的消息总数 (_MsgSent + _MsgResent 二者之和)。
_PercentReturn	浮点型	只读	预期响应 (“已接收”) 与初始发送 (“已发送”) 的比例, 以百分比形式表示。
_PercentValid	浮点型	只读	收到的有效响应总数 (_TotalResponses) 与发送的总请求数 (_MsgTotal) 的比例, 以百分比形式表示。
_Reset	布尔型	读/写	重置所有诊断计数器。写入 “_Reset 标记” 会导致此级别的所有诊断计数器重置。
_RespBadChecksum	双字型	读/写	校验和错误的响应总数。
_RespTimeouts	双字型	读/写	无法接收任何响应类型的消息总数。
_RespTruncated	双字型	读/写	仅接收部分响应的消息总数。
_TotalResponses	双字型	只读	收到有效响应的总数 (_ErrorResponses + _ExpectedResponses)。

统计信息项不会在模拟模式下更新 (请参阅设备常规属性)。

设备级统计信息项

设备级统计信息项的语法为 <通道>.<设备>._Statistics。

项	数据类型	访问	说明
_CommFailures	双字型	读/写	通信失败 (或耗用的重试) 的总次数。
_ErrorResponses	双字型	读/写	收到有效错误响应的总数。
_ExpectedResponses	双字型	读/写	收到预期响应的总数。
_LastResponseTime	字符串	只读	收到最后一次有效响应的时间。
_LateData	双字型	读/写	驱动程序标记的数据更新时间晚于预期 (基于指定的扫描速率) 的总次数。
_MsgResent	双字型	读/写	作为重试发送的消息总数。
_MsgSent	双字型	读/写	初始发送的消息总数。
_MsgTotal	双字型	只读	发送的消息总数 (_MsgSent + _MsgResent 二者之和)。
_PercentReturn	浮点型	只读	预期响应 (“已接收”) 与初始发送 (“已发送”) 的比例, 以百分比形式表示。
_PercentValid	浮点型	只读	收到的有效响应总数 (_TotalResponses) 与发送的总请求数 (_MsgTotal) 的比例, 以百分比形式表示。
_Reset	布尔型	读/写	重置所有诊断计数器。写入 “_Reset 标记” 会导致此级别的所有诊断计数器重置。
_RespBadChecksum	双字型	读/写	校验和错误的响应总数。
_RespTimeouts	双字型	读/写	无法接收任何响应类型的消息总数。
_RespTruncated	双字型	读/写	仅接收部分响应的消息总数。
_TotalResponses	双字型	只读	收到有效响应的总数 (_ErrorResponses + _ExpectedResponses)。

统计信息项不会在模拟模式下更新 (请参阅设备常规属性)。

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。

块范围中的地址错误。| 地址范围 = <起始> 到 <结束>。

错误类型：

错误

可能的原因：

1. 已尝试引用指定设备中不存在的位置。
2. 尝试读取了多于协议所允许数目的寄存器。

可能的解决方案：

1. 验证分配给设备上指定范围内地址的标记，并去除参考无效位置的标记。
2. 将寄存器块大小减小至 125。

也可以看看：

1. 错误处理
2. 块大小

不良数组。| 数组范围 = <开始> 至 <结束>。

错误类型：

错误

可能的原因：

定义了一个地址数组，该数组超越了地址空间末端。

可能的解决方案：

1. 验证设备内存空间的大小，并相应地重新定义数组长度。
2. 验证分配给设备上指定范围内地址的标记，并去除参考无效位置的标记。
3. 将数组大小值减小至 125。

也可以看看：

1. 错误处理
2. 块大小

块地址响应异常代码。| 地址范围 = <起始> 到 <结束>，异常代码 = <代码>。

错误类型：

警告

可能的原因：

有关异常代码的说明，请参阅“Modbus 异常代码”。

可能的解决方案：

请参阅“Modbus 异常代码”。

无法写入地址，设备响应异常代码。| 地址 = '<地址>'，异常代码 = <代码>。**错误类型：**

警告

可能的原因：

有关异常代码的说明，请参阅“Modbus 异常代码”。

可能的解决方案：

请参阅“Modbus 异常代码”。

无法从地址读取，设备响应异常代码。| 地址 = '<地址>'，异常代码 = <代码>。**错误类型：**

警告

可能的原因：

有关异常代码的说明，请参阅“Modbus 异常代码”。

可能的解决方案：

请参阅“Modbus 异常代码”。

标记导入由于内存资源不足而失败。**错误类型：**

警告

可能的原因：

驱动程序无法分配所需内存来处理变量导入文件。

可能的解决方案：

关闭所有不必要的应用程序，然后重试。

标记导入期间发生文件异常。**错误类型：**

警告

可能的原因：

无法读取变量导入文件。

可能的解决方案：

重新生成变量导入文件。

解析导入文件中的记录时出错。| 记录编号 = <数字>，字段 = <名称>。**错误类型：**

警告

可能的原因：

变量导入文件中的指定字段无法解析，因为它无效或长于预期长度。

可能的解决方案：

编辑变量导入文件以更改违规字段。

针对导入文件中的记录截断了说明。| 记录编号 = <编号>。

错误类型：

警告

可能的原因：

指定记录中提供的标记说明过长。

可能的解决方案：

驱动器会根据需要截断说明。为了防止此错误，请编辑变量导入文件以缩短说明。

导入的标记名称无效，且已经更改。| 标记名称 = '<标记>'，更改后的标记名称 = '<标记>'。

错误类型：

警告

可能的原因：

在变量导入文件中发现的标记名称包含无效字符。

可能的解决方案：

驱动器会根据变量导入文件构造有效的名称。为防止此错误并保持名称的一致性，请更改导出变量的名称。

由于不支持数据类型，因此无法导入标记。| 标记名称 = '<标记>'，不支持的数据类型 = '<类型>'。

错误类型：

警告

可能的原因：

驱动程序不支持变量导入文件中指定的数据类型。

可能的解决方案：

将变量导入文件中指定的数据类型更改为支持的类型。如果变量用于结构，请手动编辑文件以定义结构所需的每个标记或在服务器中手动配置所需标记。

● 也可以看看：

从概念中导出变量

由于内存分配问题，无法读取 Omni 文本缓冲区。

错误类型：

警告

可能的原因：

驱动程序无法分配 Omni 文本记录或文本归档读取操作所需的内存。

可能的解决方案：

关闭所有不必要的应用程序，然后重试。

指定日期范围内没有可用的 Omni 文本归档数据。

错误类型：

警告

可能的原因：

“开始日期”寄存器 (15128) 和“天数”寄存器 (15127) 指定的日期范围内的文本归档中无数据。

可能的解决方案：

这未必是错误。验证在指定的范围内是否有可用数据。

向 Omni 文本报告的写入遭到截断。| 报告编号 = <数字>。

错误类型：

警告

可能的原因：

尝试向文本报告中写入的字节数多于 8192。此限制由协议规定。

可能的解决方案：

不得写入超过 8192 个字节限制的字符串。如果字符串过长，则仅前 8192 个字符会被写入设备。

由于数据包编号限制，无法读取 Omni 文本报告。| 报告编号 = <数字>。

错误类型：

警告

可能的原因：

预期文本报告为 8192 个字节或更少。此限制为协议规定。驱动程序将读取预期文件字符结尾之前的 8192 个字节。

可能的解决方案：

验证设备所用报告模版生成的报告是否为 8192 个字节或更少。

写入失败。已超出最大路径长度。| 标记地址 = '<地址>'，最大长度 = <数字>。

错误类型：

警告

可能的原因：

路径长度被限制为指定字符数。

可能的解决方案：

请使用稍短的路径。

将 Omni 文本数据写入文件时出错。| 标记地址 = '<地址>'，原因 = '<原因>'。

错误类型：

警告

可能的原因：

由于特定原因，驱动程序无法将 Omni 文本数据写入磁盘。

可能的解决方案：

有关特定原因的适当更正措施，请参阅操作系统文档。

Omni 文本输出文件无法打开。| 标记地址 = '<地址>'，原因 = '<原因>'。

错误类型：

警告

可能的原因：

Omni 文本路径标记中指定的文件无法创建或打开。

可能的解决方案：

有关特定原因的适当更正措施，请参阅操作系统文档。最可能的原因为路径无效。

● 也可以看看：

1. Omni 文本报告
2. Omni 文本归档

无法写入地址。响应中存在意外字符。| 标记地址 = '<地址>'。

错误类型：

警告

无法从地址读取。响应中存在意外字符。| 标记地址 = '<地址>'。

错误类型：

警告

无法读取块地址。响应中存在意外字符。| 地址范围 = <起始> 到 <结束>。

错误类型：

警告

正在从文件导入标记数据库。| 文件名 = '<名称>'。

错误类型：

信息化

错误掩码定义

B = 检测到 检测到硬件断点

F = 帧错误

E = I/O 错误

O = 字符缓冲区溢出

R = RX 缓冲区溢出

P = 已接收字节奇偶校验错误

T = TX 缓冲区已满

Modbus 异常代码

以下数据来自 Modbus 应用程序协议规范文档。

代码 (十进制/ 十六进制)	名称	含义
01/0x01	非法功能	在查询中接收的功能代码不是服务器 (或从属设备) 允许的操作。这可能是由于功能代码仅适用于较新的设备, 而未在选定单元中实施。该代码也可以指示服务器 (或从属设备) 处理此类型请求的状态错误; 例如服务器未配置但被要求返回寄存器值。
02/0x02	非法数据地址	在查询中接收的数据地址不是服务器 (或从属设备) 的允许地址。更具体地说, 参考数字和传输长度的组合是无效的。对于具有 100 个寄存器的控制器, 采用偏移为 96 且长度为 4 的请求时会成功。采用偏移为 96 且长度为 5 的请求时将发生异常 02。
03/0x03	非法数据值	包含于查询数据字段中的值不是服务器 (或从属设备) 的允许值。这指示了复杂请求中其余部分结构的故障, 例如隐含长度不正确。具体而言, 它并不意味着提交用于存储在寄存器中的数据项的值超出应用程序预期, 因为 Modbus 协议不知道任何特定寄存器的任何特定值的重要性。
04/0x04	从属设备故障	服务器 (或从属设备) 尝试执行请求的操作时发生不可恢复的错误。
05/0x05	ACKNOWLEDGE	从属设备已接受请求并正在处理, 但需要很长时间才能完成。返回此响应是为了防止主设备发生超时错误。主设备接下来可以发出轮询程序完成消息, 从而确定是否已完成处理。
06/0x06	从属设备忙碌	从属设备正在处理持续时间较长的编程命令。当从属设备空闲时, 主设备稍后应重新发送消息。
07/0x07	负值确认	从属设备不能执行查询中接收到的编程功能。使用功能代码 13 或 14 (十进制) 的编程请求未成功时, 将返回此代码。主设备应请求来自从属设备的诊断或错误信息。
08/0x08	内存奇偶性错误	从属设备尝试读取扩展内存, 但检测到内存中的奇偶性错误。主设备可以重试请求, 但从属设备可能需要服务。
10/0x0A	网关路径不可用	专门与网关结合使用, 指示网关无法分配从输入端口到输出端口的内部通信路径来处理请求。通常, 这意味着网关配置错误或过载。
11/0x0B	网关目标设备响应失败	专门与网关结合使用, 指示没有从目标设备获得响应。通常, 这意味着设备未存在于网络上。

注意: 对于此驱动程序, 术语“从属”和“主动提供”可交替使用。

索引

B

BCD 24

D

Daniels S500 流量计算机寻址 29

Dynamic Fluid Meter 寻址 29

E

Elliott 流量计算机寻址 28

I

I/O 错误 48

ID 13

IP 地址 14

L

LBCD 24

M

Magnetek GPD 515 驱动器寻址 28

Modbus 功能代码 05 18

Modbus 功能代码 06 18

Modbus 寻址 25

Modbus 异常代码 49

Modbus 字节顺序 18

Modicon 位顺序 19

O

Omni 流量计算机寻址 30

Omni 文本报告 39

Omni 文本存档 41

Omni 文本输出文件无法打开。| 标记地址 = '<地址>', 原因 = '<原因>'。 47

Omni 原始数据存档 35

Omni 自定义数据包 33

R

RX 缓冲区溢出 48

T

TX 缓冲区已满 48

勛

包括说明 21

飲

保持寄存器 20

保持寄存器位掩码 18

卸

变量导入设置 20

变量导入文件 21

椀

标记导入期间发生文件异常。 45

标记导入由于内存资源不足而失败。 45

标记生成 16

标识 12-13

沂

波特率 6

庭

不良数组。| 数组范围 = <开始> 至 <结束>。 44

不扫描, 仅按需求轮询 14

焯

布尔型 24

摭

操作模式 13

噉

常规 12

败

超时前的尝试次数 15

冻

创建 17

兜

从零开始的位寻址 18

从零开始的寻址 18

钹

错误处理 21

错误掩码定义 48

宜

导入的标记名称无效, 且已经更改。| 标记名称 = '<标记>', 更改后的标记名称 = '<标记>'。 46

穿

端口 14

瞍

短整型 24

宙

对于重复标记 17

泊

浮点型 24

燭

父组 17

襪

覆盖 17

栢

概述 5

别

功能代码说明 42

孽

故障时降级 16

孤

寄存器;线圈;地址说明 25

宦

将 Omni 文本数据写入文件时出错。| 标记地址 = '<地址>', 原因 = '<原因>'。 47

将长整型作为十进制处理 19

闭

降级超时 16

降级期间 16

降级时放弃请求 16

覃

解析导入文件中的记录时出错。| 记录编号 = <数字>, 字段 = <名称>。 45

拎

拒绝重复消息 22

困

块大小 20

块地址响应异常代码。| 地址范围 = <起始> 到 <结束>, 异常代码 = <代码>。 44

块读取字符串 20

块范围中的地址错误。| 地址范围 = <起始> 到 <结束>。 44

袒

框架 21, 48

揭

来自缓存的初始更新 14

轻

连接尝试次数 15

连接超时 15

厭

名称 13

稿

模拟 13

價

内部寄存器 20

礮

奇偶校验 6, 48

凭

前导 21

講

请求超时 15

请求间延迟 15

駢

驱动程序 13

芻

取消激活非法地址上的标记 22

儷

冗余 22

戔

扫描模式 14

净

删除 17

讓

设备启动时 16

设备属性 - 标记生成 16

设置 6, 17

璿

生成 17

丫

事件日志消息 44

銷

首双字排序为低 19

首字排序为低 18

轄

输出线圈 20

输入线圈 20

对

属性更改时 16

攘

数据编码 18

数据类型说明 24

数据收集 13

数据位 6

印

双精度 24

双字型 24

侑

停止位 6

辺

通道分配 13

通信超时 15

通信协议 6

繼

统计信息项 42

寔

尾随 21

駁

无法从地址读取, 设备响应异常代码。| 地址 = '<地址>', 异常代码 = <代码>。 45

无法从地址读取。响应中存在意外字符。| 标记地址 = '<地址>'。 48

无法读取块地址。响应中存在意外字符。| 地址范围 = <起始> 到 <结束>。 48

无法写入地址, 设备响应异常代码。| 地址 = '<地址>', 异常代码 = <代码>。 45

无法写入地址。响应中存在意外字符。| 标记地址 = '<地址>'。 48

廛

向 Omni 文本报告的写入遭到截断。| 报告编号 = <数字>。 47

勳

协议 15

償

写入失败。已超出最大路径长度。| 标记地址 = '<地址>', 最大长度 = <数字>。 47

坵

型号 13

丿

以太网封装 14

灣

溢出 48

砌

硬件断点 48

璣

由于不支持数据类型, 因此无法导入标记。| 标记名称 = '<标记>', 不支持的数据类型 = '<类型>'。 46

由于内存分配问题, 无法读取 Omni 文本缓冲区。 46

由于数据包编号限制, 无法读取 Omni 文本报告。| 报告编号 = <数字>。 47

德

允许子组 17

錕

长整型 24

鑽

针对导入文件中的记录截断了说明。| 记录编号 = <编号>。 46

櫛

正在从文件导入标记数据库。| 文件名 = '<名称>'。 48

撤

支持的设备 6

摺

指定日期范围内没有可用的 Omni 文本归档数据。 46

膊

自动标记数据库生成 23

自动降级 15

媯

字 24

字符串 24

逕

遵循标签指定的扫描速率 14