Omron FINS 串行驱动程序

© 2025 PTC Inc. 保留所有权利。

目录

Omron FINS 串行驱动程序 **1**
目录 **2**
Omron FINS 串行驱动程序 4
概述 5
设置 5
通道属性 - 常规 6
标记计数 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 9
设备属性 - 常规 10
操作模式 11
标记计数 12
设备属性 - 扫描模式 12
设备属性 - 定时 12
设备属性 - 自动降级 13
设备属性 - 运行模式设置 14
设备属性 - 通信参数 14
设备属性 - 冗余 15
FINS 网络 16
数据类型说明 **19**
地址说明 **20**
C200H 寻址 20
C500 寻址 25
C1000H 寻址 28
C2000H 寻址 32
CV500 寻址 36
CV1000 寻址 40
CV2000 寻址 46
CVM1-CPU01 寻址 51
CVM1-CPU11 寻址 55
CVM1-CPU21 寻址 59
CS1 寻址 65
CJ1 寻址 71
CJ2 寻址 76

事件日志消息 **81**
设备响应错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 81
设备响应本地节点错误。| 主错误 = <代码>，子错误 = <代码>。 81
设备响应目的节点错误。| 主错误 = <代码>，子错误 = <代码>。 81
设备响应通信错误。| 主错误 = <代码>，子错误 = <代码>。 82
设备无法处理命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 82
设备响应路由表错误。| 主错误 = <代码>，子错误 = <代码>。 82
设备响应命令格式错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 82
设备响应命令参数错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 83
设备响应不可读取。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 83
设备响应单元错误。| 主错误 = <代码>，子错误 = <代码>。 83
设备无法接受命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 84
访问权限被拒绝。| 标记地址 = '<地址>'，主错误 = <代码>，子错误 = <代码>。 84
设备响应不可写入。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 84
无法写入标记。设备处于“运行模式”。| 标记地址 = '<地址>'。 85
设备响应致命 CPU 单元错误。| 终止代码 = <代码>。 85
设备响应非致命 CPU 单元错误。| 终止代码 = <代码>。 85
写入标记后无法将设备设置回“运行模式”。| 标记地址 = '<地址>'。 85
错误掩码定义 85
主错误代码和子错误代码 85
索引 **93**

Omron FINS 串行驱动程序

目录

概述
什么是 Omron FINS 串行驱动程序？

设置
如何配置使用此驱动程序的设备？

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 Omron FINS Serial 设备上的数据位置进行寻址？

事件日志消息
Omron FINS 串行驱动程序 会产生哪些消息？

版本 1.060
© 2025 PTC Inc. 保留所有权利。

概述

Omron FINS 串行驱动程序 提供将 Omron FINS 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。

设置

此驱动程序通过“Sysmac Way：主机链路界面”支持 FINS 协议。

有关支持“Sysmac Way：主机链路界面”的当前型号列表 (参考：Omron CX-Server Runtime - 用户手册)，请参阅制造商的网站。

通信协议

Omron FINS

支持 通信参数

波特率：1200、2400、4800、9600、19200、38400、57600、115200 bps
奇偶校验：偶、奇或无
数据位：7 或 8
停止位：1 或 2

通道和设备限制

此驱动程序支持的最大通道数量为 100。此驱动程序所支持设备的最大数量为每通道 32 个。

以太网封装

此驱动程序支持“以太网封装”，允许驱动程序使用终端服务器与连接到以太网的串行设备进行通信。可以通过通道属性进行设置。

流量控制

使用 RS232/RS485 转换器时，所需的流量控制类型取决于转换器的需求。一些转换器不需要任何流量控制，而其他转换器则需要 RTS 流量。要确定转换器的流量要求，请参阅其文档。建议使用提供自动流量控制功能的 RS485 转换器。

注意：

- 在使用制造商提供的通信电缆时，有时需要在“通道属性”中选取流量控制设置 RTS 或“始终为 RTS”。
- 在未规定正确流量控制的平台上运行时，用户可能需要在服务器通信设置中设置流量控制。

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。

标识

"名称"：指定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"：指定此通道的用户定义信息。

在这些属性中，有很多属性 (包括"说明") 具有关联的系统标记。

"驱动程序"：为该通道指定的协议/驱动程序。指定在创建通道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。

请知悉：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。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。采用适当的用户角色和权限管理来防止操作员更改属性或访问服务器功能。

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。

请知悉：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"。

诊断

"诊断数据捕获"：启用此选项后，将允许使用统计信息标记，用以向客户端应用程序提供有关通道操作的反馈。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。

请知悉：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"统计信息标记"。

标记计数

“静态标记”: 提供此级别 (设备或通道) 上已定义静态标记的总数。此信息有助于排除故障和平衡负载。

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”，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。采用适当的用户角色和权限管理来防止操作员更改属性或访问服务器功能。
- 用户必须定义将使用的具体通信参数。根据驱动程序的不同，通道可能会共享相同的通信参数，也可能不会。只能为虚拟网络配置一个共享串行连接*(请参阅通道属性-串行通信)* 。

连接类型

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

1. **“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。
2. **“COM 端口”:** 选择“COM 端口”可显示和配置“串行端口设置”部分。
3. **“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。
4. **“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 9991 至 16。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对流量进行控制。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”:** 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”:** 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”:** 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。

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

服务器必须确保从客户端应用程序写入的数据能够准时发送到设备。为此，服务器提供了优化属性，用以满足特定需求或提高应用程序响应能力。

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"(Write All Values for All Tags)**：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。此模式会通过在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。

"非规范浮点数处理"：非规范值定义为无穷大、非数字(NaN) 或非规范数。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 非规范、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，则将禁用此属性。根据通道的浮点规范设置，将仅对实时驱动程序标记 (如值和数组) 进行浮点规范。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 指定了唯一标识目标设备的三层网络地址。ID 的格式为 *UU.AAA.NNN*，其中：

- **UU**：用于 PC 接口的主机链路单元的单元编号 (十进制 0 到 31)。
- **AAA**：FINS 目标网络地址 (十进制 0 到 127)。
- **NNN**：FINS 目标节点编号 (十进制 0 到 254)。

有关详细信息，请参阅 ***FINS 网络***。

另请参阅：*操作模式*。

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。

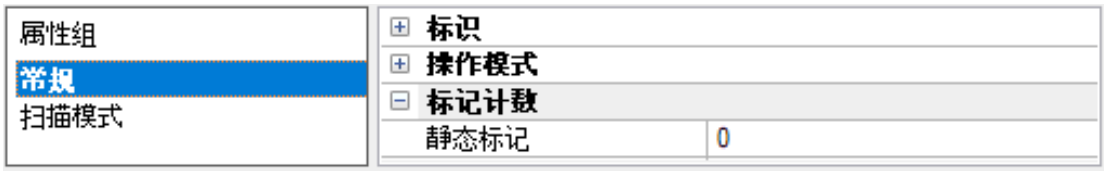
"模拟"：使设备进入或退出模拟模式。在此模式下，驱动程序不会尝试与物理设备进行通信，但服务器将继续返回有效的 OPC 数据。模拟停止与设备的物理通信，但允许 OPC 数据作为有效数据返回到 OPC 客户端。在"模拟模式"下，服务器将所有设备数据处理为反射型：无论向模拟设备写入什么内容，都会读取回来，而且会单独处理每个 OPC 项。项的内存映射取决于组更新速率。如果服务器移除了项 (如服务器重新初始化时)，则不保存数据。默认值为"否"。

请知悉：

1. 只有当客户端断开连接并重新连接后，才会应用更新。
2. "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。
3. 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。
4. 模拟设备时，客户端的更新速度可能不会低于 1 秒。

"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。

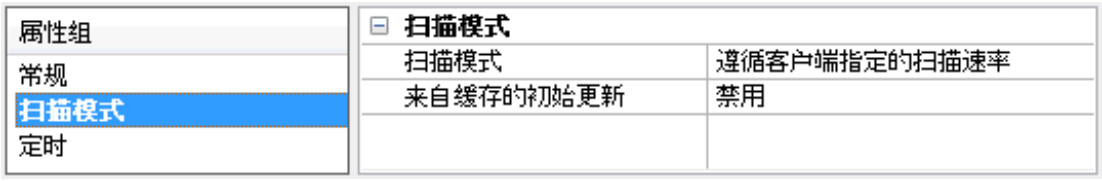
标记计数



"静态标记"：提供此级别 (设备或通道) 上已定义静态标记的总数。此信息有助于排除故障和平衡负载。

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。



"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。OPC 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。

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。

请知悉: 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。

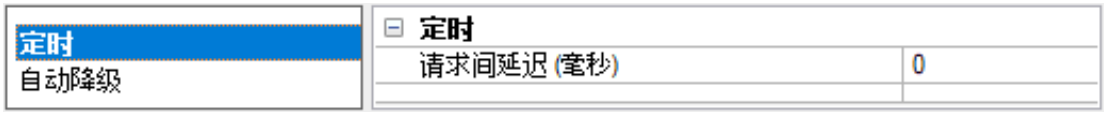
"请求超时": 指定一个所有驱动程序使用的间隔来决定驱动程序等待目标设备完成响应的时间。有效范围是 50 到 9999999 毫秒 (167 分钟)。默认值通常是 1000 毫秒，但可能会因驱动程序而异。大多数串行驱动程序的默认超时是基于 9600 波特或更高的波特率来确定的。当以较低的波特率使用驱动程序时，请增加超时，以补偿获取数据所需增加的时间。

"超时前的尝试次数": 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": 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 到 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。

请知悉: 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。

“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。

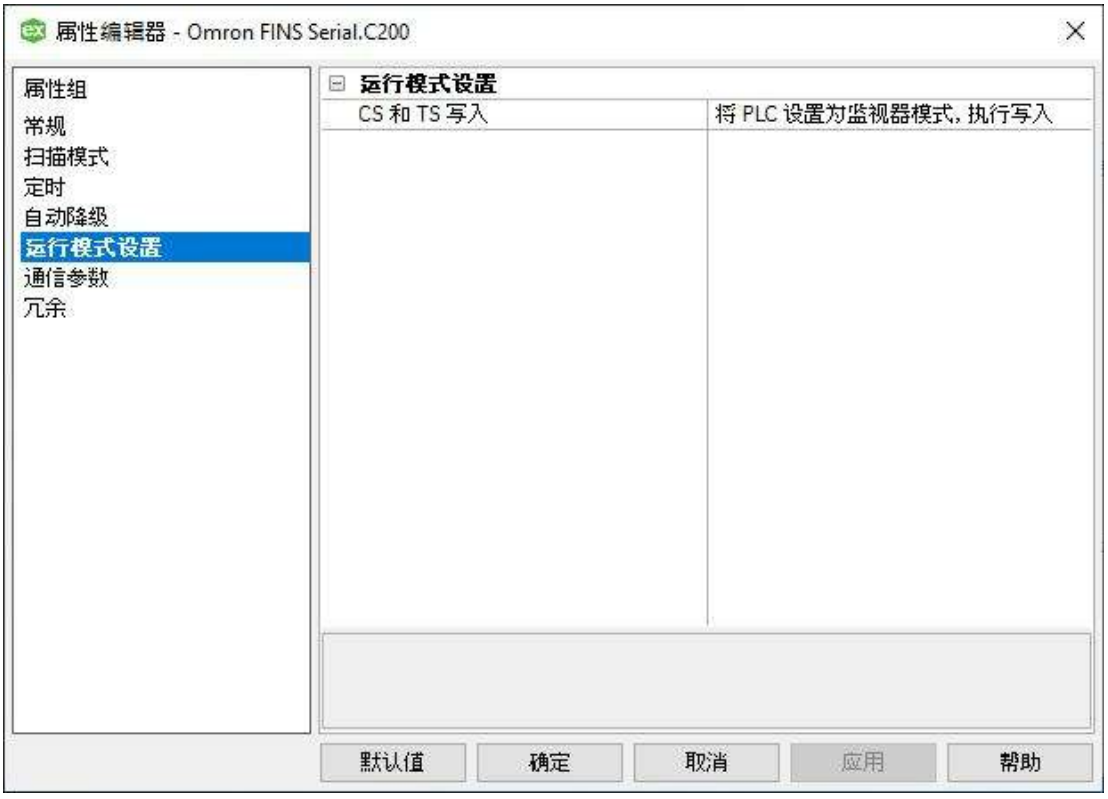
“降级超时”：指定在对设备取消扫描之前，请求超时和重试的连续周期数。有效范围是 1 到 30 次连续失败。默认值为 3。

“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 - 运行模式设置

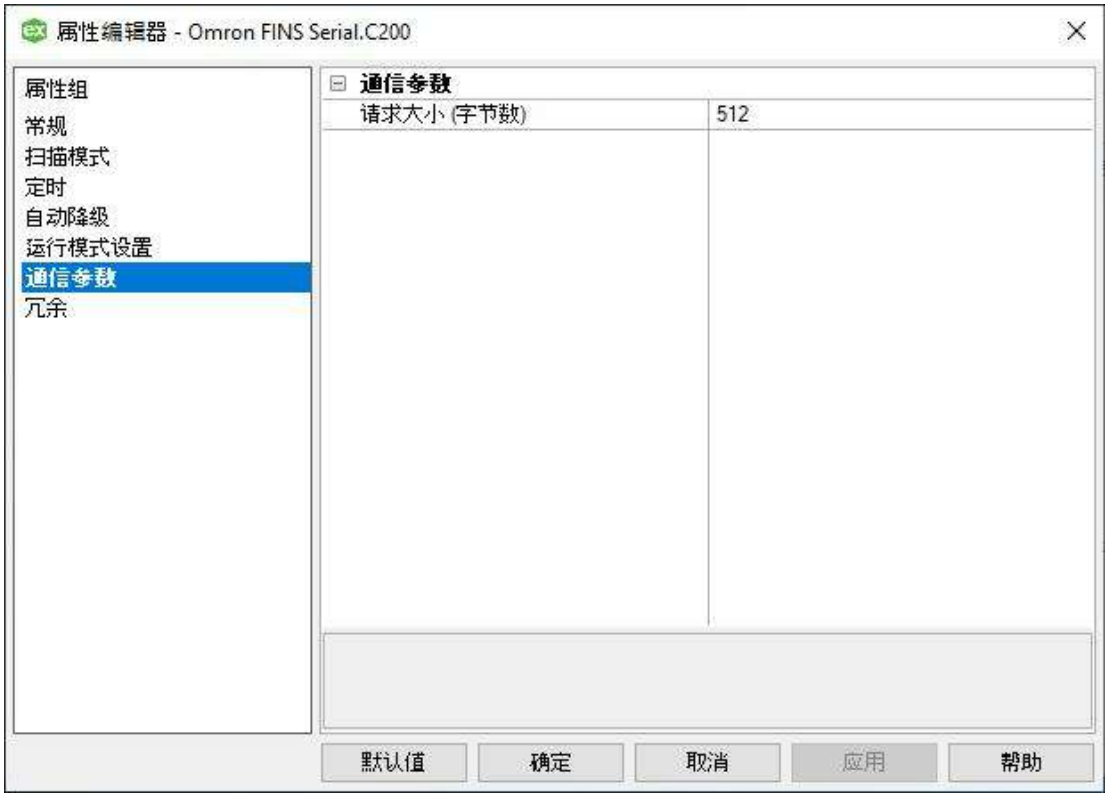
此组指定了当设备在“运行模式”下对“计时器状态”和“计数器状态”进行写入时驱动程序的行为。



“CS 和 TS 写入”：选择设备在“运行”模式下执行写入的方式。默认设置为“写入失败，日志消息”。

- **“写入失败，日志消息”**：在写入命令失败时将消息发布到事件日志。
- **“将 PLC 设置为监视器模式，执行写入”**：在执行写入之前将 PLC 更改为监视器模式。
- **“将 PLC 设置为监视器模式，写入、重置以运行”**：在执行写入之前将 PLC 更改为监视器模式。完成后，PLC 将重置为“运行模式”。

设备属性 - 通信参数



"请求大小"：指定从设备上一次可以请求的字节数。要优化驱动程序的性能，请将"请求大小"设置为以下设置之一：32、64、128、256 或 512 字节。默认设置为 512 字节。

注意：由于此驱动程序使用 ASCII 协议，因此，每个字、短整型和 BCD 可传输四个字节。每个双字型、长整型、LBCD 和浮点型可传输八个字节。

设备属性 - 冗余

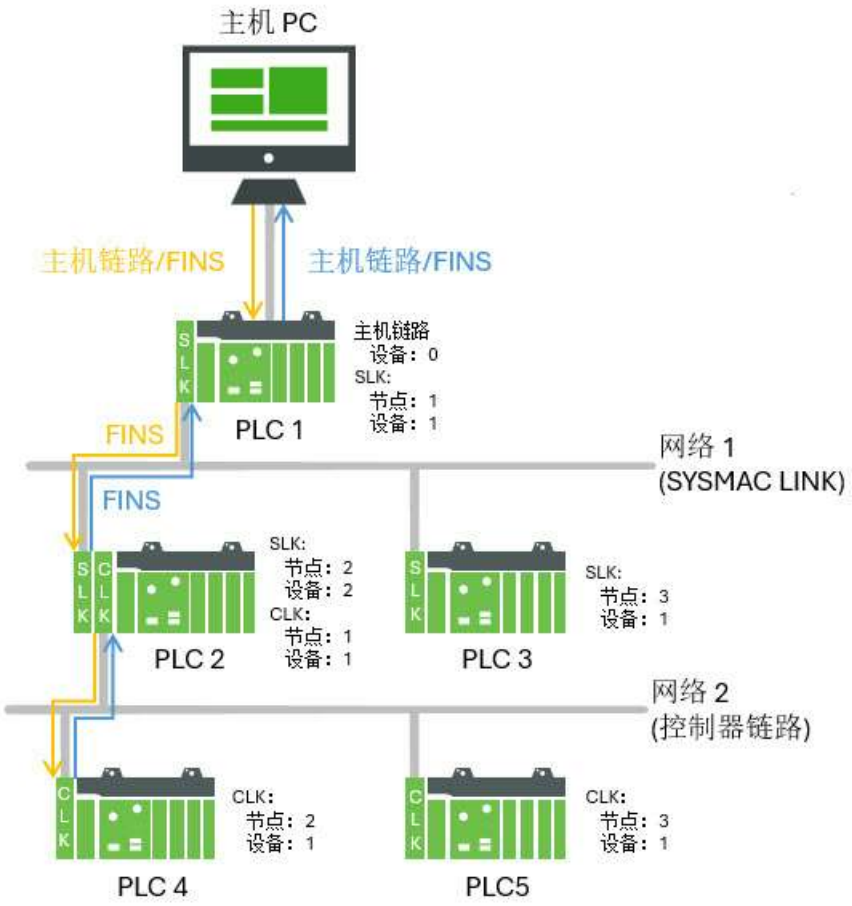
Media-Level Redundancy 插件提供冗余。

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。

FINS 网络

该 FINS 通信服务由 Omron 开发，为各种网络上的 PLC 和计算机提供了一种一致的通信方式。兼容的网络类型包括以太网、主机链路、控制器链路、SYSMAC LINK、SYSMAC WAY 和 Toolbus。FINS 允许最多在三个网络级别的节点之间进行通信。PC 和 PLC 之间通过主机链路的直接链接不算作网络层。

下表显示了由互联 SYSMAC 链路和控制器链路网络组成的 FINS 网络，作为以下讨论的一个例子。FINS 允许在布线图中的任意设备对之间进行通信。PLC 1 用作网络 1 的主机计算机主机链路接口。PLC 2 用作网络 1 和 2 之间的网关。主机 PC 通过 PLC 1 和 2 向 PLC 4 发送数据请求命令 (以红色显示)。响应以蓝色显示。原则上，PLC 4 或 5 可充当主机 PC 也可访问的又一个网络层网关。



FINS 消息

FINS 消息包含两个部分：标题和数据部分。标题中包含源和目标信息等。数据部分包含命令代码和可选的命令参数。标题中包含的六个源和目标参数如下：

- **DNA:** 目标网络地址。
- **DA1:** 目标节点编号。
- **DA2:** 目标模块地址。
- **SNA:** 源网络地址。
- **SA1:** 源节点编号。
- **SA2:** 源模块地址。

此驱动程序将 DA2 和 SA2 设置为零，表示通信发生于主机计算机和目标节点的 CPU 模块之间。此驱动程序还将 SNA 和 SA1 设置为零，表示通信通过接口设备的主机链路端口。将消息转发给其他 PLC 时，接口设备 (PLC 1) 将根据需要重置 SNA 和 SA1。

主机链路通信

此驱动程序能够通过主机链路端口与任何 FINS 兼容设备进行通信。如果该设备是目标节点，则可处理给定 FINS 命令，否则，会将消息中继到另一个设备。在上图中，需要将数据请求从主机 PC 发送到 PLC 4。该驱动程序将构建一条 FINS 数据请求消息，并在标题中设置适当的源和目标参数，将该消息封装在主机链路封套元素中，然后将其发送到 PLC 1。接下来，PLC 1 检查 FINS 消息标题，并确定网络 2 (PLC 4) 上的节点 2 是否为预期目标。通过参考其路由表，PLC 1 可以确定消息必须发送到网关设备 PLC 2，以便到达网络 2 上的目标节点。由于 PLC 2 不在主机链路网络上，因此将移除主机链路封套元素。PLC 2 转而将消息发送至 PLC 4，PLC 4 会通过 PLC 2 和 1 将其回复发送回主机 PC。PLC 1 将 FINS 响应封装在主机链路封套元素中，供该驱动程序识别和处理。

在本例中，服务器必须如下配置：

1. 首先，创建一个使用此驱动程序的通道。然后，创建代表目标节点 (PLC 4) 的设备。
2. 下一步，设置设备 ID。接口设备的主机链路单元编号为 0，FINS 目标网络地址 (DNA) 为 2，FINS 目标节点编号 (DA1) 为 2。因此，此示例中的设备 ID 是 0.2.2。

   注意：如之前所述，SNA、SA1、SA2 和 DA2 将由驱动程序自动设置为零。

3. 必须为 PLC 1、2、3 和 5 创建类似设备对象。

路由表

对于多级网络，必须将附加信息编程到 PLC 中，以便它们可以向系统中的其他节点发送消息。此操作通过 FINS 路由表完成。FINS 路由表分为两种类型：本地和远程。本地路由表将网络号与在 PLC 机架中安装的通信模块或特殊输入/输出单元 (SIOU) 之一相关联。远程路由表指示如何到达下面的网络层级之一。此示例中的路由表如下所示：

PLC 1 (本地)

网络编号	单位
1	1

由于 PLC 1 仅连接到 SYSMAC 链路网络，因此它在本地路由表中只有一个条目。

PLC 1 (远程)

远程网络编号	继电器网络	继电器节点
2	1	2

网络 2 是 PLC 1 的远程网络。要将消息发送到网络 2 上的节点，PLC 1 必须将该消息发送到其中一个本地网络上的中继节点 (或网关)。此本地网络称为继电器网络，且必须在此示例中的网络 1。网关 (PLC 2) 中的 SYSMAC 链路模块采用节点编号 2。因此，PLC 1 继电器节点为 2。

PLC 2 (本地)

网络编号	单位
1	2
2	1

PLC 2 有两个本地网络。SYSMACK 链路和控制器链路网络编号将分别分配为 1 和 2。SYSMACK 链路和控制器链路模块将分别分得单元编号 2 和 1。所有这些模块的网络节点编号均由用户配置，且在其关联网络中必须具有唯一性。在本例中，PLC 2 中的两个通信模块均可称为节点 10，因为它们位于不同的网络上。

PLC 3 (本地)

网络编号	单位
1	1

PLC 3 的本地路由表仅有一个条目，因为该设备仅连接到网络 1。

PLC 3 (远程)

远程网络编号	继电器网络	继电器节点
2	1	2

网络 2 是 PLC 3 的远程网络。与 PLC 1 类似，继电器网络为 1，继电器节点为 2 (网关设备 PLC 2 的 SLK 模块)。

路由表 PLC 4 和 5 必须如下所示。路由表 PLC 4 和 5 必须如下所示。必须为这些模块分配网络 2 中唯一的节点编号。

PLC 4 和 5 (本地)

网络编号	单位
2	1

PLC 4 和 5 (远程)

远程网络编号	继电器网络	继电器节点
1	2	1

有关详细信息，请参阅 Omron 文档。

数据类型说明

数据类型	说明
布尔型	单个位
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
字	无符号 16 位值 位 0 是低位 位 15 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
双字	无符号 32 位值 位 0 是低位 位 31 是高位
浮点型	32 位实数
BCD	两字节压缩 BCD 值的范围是 0 至 9999。对于超出此范围的值，未定义行为。
LBCD	四字节压缩 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。 支持多达 256 个字符的字符串长度，并可选择由高到低的字节顺序、由低到高的字节顺序、仅高位字节和仅低位字节。

地址说明

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

C200H 寻址
C500 寻址
C1000H 寻址
C2000H 寻址
CV500 寻址
CV1000 寻址
CV2000 寻址
CVM1 CPU01 寻址
CVM1 CPU11 寻址
CVM1 CPU21 寻址
CS1 寻址
CJ1 寻址
CJ2 寻址

C200H 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	AR00-AR27 AR00-AR26 ARxx.00-ARxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
辅助继电器为字符串型 (字节排序由高到低)	AR00.056H-AR27.002H .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (字节排序由低到高)	AR00.056L-AR27.002L .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (仅使用每个字的高位字节)	AR00.028D-AR27.001D .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (仅使用每个字的低位字节)	AR00.028E-AR27.001E .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
数据内存	DM0000-DM6655 DM0000-DM6654 DMxxxx.00-DMxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	DM0000.256H-DM6655.002H	**字符串**	读/写

设备类型	范围	数据类型	访问
	.l 是字符串长度，范围介于 2 到 256 个字符之间		
数据内存为字符串型 (字节排序由低到高)	DM0000.256L-DM6655.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	DM0000.128D-DM6655.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	DM0000.128E-DM6655.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组)	EM0000-EM6143 EM0000-EM6142 EMxxxx.00-EMxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	EM0000.256H-EM6143.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	EM0000.256L-EM6143.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	EM0000.128D-EM6143.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	EM0000.128E-EM6143.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存	EM00:0000-EM07:6143 EM00:0000-EM07:6142 EMx:x.00-EMxx:xxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存为字符串型 (字节排序由高到低)	EM00:0000.256H-EM07:6143.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (字节排序由低到高)	EM00:0000.256L-EM07:6143.002L	**字符串**	读/写

设备类型	范围	数据类型	访问
	.l 是字符串长度，范围介于 2 到 256 个字符之间		
扩展数据内存为字符串型 (仅使用每个字的高位字节)	EM00:0000.128D-EM07:6143.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	EM00:0000.128E-EM07:6143.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	HR00-HR99 HR00-HR98 HRxx.00-HRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
保持继电器为字符串型 (字节排序由高到低)	HR00.200H-HR99.002H .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (字节排序由低到高)	HR00.200L-HR99.002L .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的高位字节)	HR00.100D-HR99.001D .l 是字符串长度，范围介于 1 到 100 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的低位字节)	HR00.100E-HR99.001E .l 是字符串长度，范围介于 1 到 100 个字符之间	**字符串**	读/写
内部继电器	IR000-IR511 IR000-IR510 IRxxx.00-IRxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
内部继电器为字符串型 (字节排序由高到低)	IR000.256H-IR511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (字节排序由低到高)	IR000.256L-IR511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的高位字节)	IR000.128D-IR511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的低位字节)	IR000.128E-IR511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
链路继电器	LR00-LR63 LR00-LR62	**字**、短整型、BCD	读/写

设备类型	范围	数据类型	访问
	LRxx.00-LRxx.15	长整型、双字型、LBCD、浮点型 布尔型	
链路继电器为字符串型 (字节排序由高到低)	LR00.128H-LR63.002H .l 是字符串长度，范围介于 2 到 128 个字符之间	字符串	读/写
链路继电器为字符串型 (字节排序由低到高)	LR00.128L-LR63.002L .l 是字符串长度，范围介于 2 到 128 个字符之间	字符串	读/写
链路继电器为字符串型 (仅使用每个字的高位字节)	LR00.064D-LR63.001D .l 是字符串长度，范围介于 1 到 64 个字符之间	字符串	读/写
链路继电器为字符串型 (仅使用每个字的低位字节)	LR00.064E-LR63.001E .l 是字符串长度，范围介于 1 到 64 个字符之间	字符串	读/写
暂存继电器	TR TR0-TR7	字、短整型 布尔型	读/写
计时器/计数器	TC000-TC511	**BCD**、字、短整型	读/写
计时器/计数器为字符串型 (字节排序由高到低)	TC000.256H-TC511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
计时器/计数器为字符串型 (字节排序由低到高)	TC000.256L-TC511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
计时器/计数器为字符串型 (仅使用每个字的高位字节)	TC000.128D-TC511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器/计数器为字符串型 (仅使用每个字的低位字节)	TC000.128E-TC511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器/计数器状态	TS000-TS511	布尔型	读/写

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

C200H 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 DM1000 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：DM1000.100H
2. 对于从 DM1100 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：DM1100.078L
3. 对于从 DM2000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：DM2000.055D
4. 对于从 DM2200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：DM2200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

DMxxxx [行数] [列数]
DMxxxx [列数] - (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 DM0 和 DM1 在字 DM1 处重叠。因此，写入 DM0 还会修改保存在 DM1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 DM0、DM2、DM4 等字可以防止“字”重叠。

C500 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
数据内存	DM000-DM511 DM000-DM510 DMxxx.00-DMxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	DM000.256H-DM511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	DM000.256L-DM511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	DM000.128D-DM511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	DM000.128E-DM511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	HR00-HR31 HR00-HR30 HRxx.00-HRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
保持继电器为字符串型 (字节排序由高到低)	HR00.064H-HR31.002H .l 是字符串长度，范围介于 2 到 64 个字符之间	**字符串**	读/写
保持继电器为字符串型 (字节排序由低到高)	HR00.064L-HR31.002L .l 是字符串长度，范围介于 2 到 64 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字	HR00.032D-HR31.001D	**字符串**	读/写

设备类型	范围	数据类型	访问
的高位字节)	.l 是字符串长度，范围介于 1 到 32 个字符之间		
保持继电器为字符串型 (仅使用每个字的低位字节)	HR00.032E-HR31.001E .l 是字符串长度，范围介于 1 到 32 个字符之间	**字符串**	读/写
内部继电器	IR00-IR63 IR00-IR62 IRxx.00-IRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
内部继电器为字符串型 (字节排序由高到低)	IR00.128H-IR63.002H .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
内部继电器为字符串型 (字节排序由低到高)	IR00.128L-IR63.002L .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的高位字节)	IR00.064D-IR63.001D .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的低位字节)	IR00.064E-IR63.001E .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
链路继电器	LR00-LR31 LR00-LR30 LRxx.00-LRxx.15	**字**、短整型、 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
链路继电器为字符串型 (字节排序由高到低)	LR00.064H-LR31.002H .l 是字符串长度，范围介于 2 到 64 个字符之间	**字符串**	读/写
链路继电器为字符串型 (字节排序由低到高)	LR00.064L-LR31.002L .l 是字符串长度，范围介于 2 到 64 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的高位字节)	LR00.032D-LR31.001D .l 是字符串长度，范围介于 1 到 32 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的低位字节)	LR00.032E-LR31.001E .l 是字符串长度，范围介于 1 到 32 个字符之间	**字符串**	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器/计数器	TC000-TC127	**BCD**、字、短整型	读/写
计时器/计数器为字符串型 (字节排序由高到低)	TC000.256H-TC127.002H .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
计时器/计数器为字符串型 (字节排序由低到高)	TC000.256L-TC127.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器/计数器为字符串型 (仅使用每个字的高位字节)	TC000.128D-TC127.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器/计数器为字符串型 (仅使用每个字的低位字节)	TC000.128E-TC127.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器/计数器状态	TS000-TS127	**布尔型**	读/写

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

C500 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 DM100 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：DM100.100H
2. 对于从 DM110 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：DM110.078L
3. 对于从 DM200 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：DM200.055D
4. 对于从 DM220 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：DM220.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。使用 ata 内存位置举例说明。

DMxxxx [行数] [列数]
DMxxxx [列数] - (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 DM0 和 DM1 在字 DM1 处重叠。因此，写入 DM0 还会修改保存在 DM1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 DM0、DM2、DM4 等字可以防止“字”重叠。

C1000H 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	AR00-AR27 AR00-AR26 ARxx.00-ARxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
辅助继电器为字符串型 (字节排序由高到低)	AR00.056H-AR27.002H .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
辅助继电器为字符串型 (字节排序由低到高)	AR00.056L-AR27.002L .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (仅使用每个字的高位字节)	AR00.028D-AR27.001D .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (仅使用每个字的低位字节)	AR00.028E-AR27.001E .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
数据内存	DM0000-DM4095 DM0000-DM4094 DMxxxx.00-DMxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	DM0000.256H-DM4095.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	DM0000.256L-DM4095.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	DM0000.128D-DM4095.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	DM0000.128E-DM4095.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	HR00-HR99 HR00-HR98 HRxx.00-HRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
保持继电器为字符串型 (字节排序由高到低)	HR00.200H-HR99.002H .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (字节排序由低到高)	HR00.200L-HR99.002L .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的高位字节)	HR00.100D-HR99.001D .l 是字符串长度，范围介于 1 到 100 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的低位字节)	HR00.100E-HR99.001E .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 1 到 100 个字符之间		
内部继电器	IR000-IR255 IR000-IR254 IRxxx.00-IRxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
内部继电器为字符串型 (字节排序由高到低)	IR000.256H-IR255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (字节排序由低到高)	IR000.256L-IR255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的高位字节)	IR000.128D-IR255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的低位字节)	IR000.128E-IR255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
链路继电器	LR00-LR63 LR00-LR62 LRxx.00-LRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
链路继电器为字符串型 (字节排序由高到低)	LR00.128H-LR63.002H .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
链路继电器为字符串型 (字节排序由低到高)	LR00.128L-LR63.002L .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的高位字节)	LR00.064D-LR63.001D .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的低位字节)	LR00.064E-LR63.001E .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器/计数器	TC000-TC511	**BCD**、字、短整型	读/写
计时器/计数器为字符串型 (字节排序由高到低)	TC000.256H-TC511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器/计数器为字符串型 (字节排序由低到高)	TC000.256L-TC511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
计时器/计数器为字符串型 (仅使用每个字的高位字节)	TC000.128D-TC511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器/计数器为字符串型 (仅使用每个字的低位字节)	TC000.128E-TC511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器/计数器状态	TS000-TS511	布尔型	读/写

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

C1000H 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串的长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 DM1000 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：DM1000.100H
2. 对于从 DM1100 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：DM1100.078L
3. 对于从 DM2000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：DM2000.055D
4. 对于从 DM2200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：DM2200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

DMxxxx [行数] [列数]
DMxxxx [列数] - (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 DM0 和 DM1 在字 DM1 处重叠。因此，写入 DM0 还会修改保存在 DM1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 DM0、DM2、DM4 等字可以防止“字”重叠。

C2000H 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	AR00-AR27 AR00-AR26 ARxx.00-ARxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
辅助继电器为字符串型 (字节排序由高到低)	AR00.056H-AR27.002H .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (字节排序由低到高)	AR00.056L-AR27.002L .l 是字符串长度，范围介于 2 到 56 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
辅助继电器为字符串型 (仅使用每个字的高位字节)	AR00.028D-AR27.001D .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
辅助继电器为字符串型 (仅使用每个字的低位字节)	AR00.028E-AR27.001E .l 是字符串长度，范围介于 1 到 28 个字符之间	**字符串**	读/写
数据内存	DM0000-DM6655 DM0000-DM6654 DMxxxx.00-DMxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	DM0000.256H-DM6655.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	DM0000.256L-DM6655.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	DM0000.128D-DM6655.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	DM0000.128E-DM6655.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	HR00-HR99 HR00-HR98 HRxx.00-HRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
保持继电器为字符串型 (字节排序由高到低)	HR00.200H-HR99.002H .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (字节排序由低到高)	HR00.200L-HR99.002L .l 是字符串长度，范围介于 2 到 200 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的高位字节)	HR00.100D-HR99.001D .l 是字符串长度，范围介于 1 到 100 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的低位字节)	HR00.100E-HR99.001E .l 是字符串长度，范围介于 1 到 100 个字符之间	**字符串**	读/写
内部继电器	IR000-IR255 IR000-IR254	**字**、短整型、BCD 长整型、双字型、LBCD、	读/写

设备类型	范围	数据类型	访问
	IRxxx.00-IRxxx.15	浮点型 **布尔型**	
内部继电器为字符串型 (字节排序由高到低)	IR000.256H-IR255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (字节排序由低到高)	IR000.256L-IR255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的高位字节)	IR000.128D-IR255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
内部继电器为字符串型 (仅使用每个字的低位字节)	IR000.128E-IR255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
链路继电器	LR00-LR63 LR00-LR62 LRxx.00-LRxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
链路继电器为字符串型 (字节排序由高到低)	LR00.128H-LR63.002H .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
链路继电器为字符串型 (字节排序由低到高)	LR00.128L-LR63.002L .l 是字符串长度，范围介于 2 到 128 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的高位字节)	LR00.064D-LR63.001D .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
链路继电器为字符串型 (仅使用每个字的低位字节)	LR00.064E-LR63.001E .l 是字符串长度，范围介于 1 到 64 个字符之间	**字符串**	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器/计数器	TC000-TC511	**BCD**、字、短整型	读/写
计时器/计数器为字符串型 (字节排序由高到低)	TC000.256H-TC511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器/计数器为字符串型 (字节排序由低到高)	TC000.256L-TC511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器/计数器为字符串型 (仅使用每个字的高位字节)	TC000.128D-TC511.001D .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 1 到 128 个字符之间		
计时器/计数器为字符串型 (仅使用每个字的低位字节)	TC000.128E-TC511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器/计数器状态	TS000-TS511	**布尔型**	读/写

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

C2000H 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 DM1000 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：DM1000.100H
2. 对于从 DM1100 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：DM1100.078L
3. 对于从 DM2000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：DM2000.055D
4. 对于从 DM2200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：DM2200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

DMxxxx [行数] [列数]
DMxxxx [列数] - (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 DM0 和 DM1 在字 DM1 处重叠。因此，写入 DM0 还会修改保存在 DM1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 DM0、DM2、DM4 等字可以防止“字”重叠。

CV500 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
操作标志	AC0000-AC1023	**布尔型**	只读
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读

设备类型	范围	数据类型	访问
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C000-C511	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C000.256H-C511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C000.256L-C511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C000.128D-C511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C000.128E-C511.001E	**字符串**	读/写

设备类型	范围	数据类型	访问
	.l 是字符串长度，范围介于 1 到 128 个字符之间		
计数器状态	CS000-CS511	布尔型	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
数据内存	D0000-D8191 D0000-D8190 Dxxxx.00-Dxxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
数据内存为字符串型 (字节排序由高到低)	D0000.256H-D8191.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
数据内存为字符串型 (字节排序由低到高)	D0000.256L-D8191.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D0000.128D-D8191.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D0000.128E-D8191.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
数据寄存器	DR0-DR2	字、短整型、BCD*	读/写
索引寄存器	IR0-IR2	字、短整型、BCD*	读/写
步进计时器	ST000-ST511	字、短整型、BCD*	读/写
步进计时器状态	STS000-STS511	布尔型	读/写
暂存继电器	TR TR0-TR7	字、短整型 布尔型	读/写

设备类型	范围	数据类型	访问
计时器	T000-T511	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T000.256H-T511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T000.256L-T511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T000.128D-T511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T000.128E-T511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS000-TS511	**布尔型**	读/写
转换标志	TN000-TN511	**布尔型**	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CV500 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D1000 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：D1000.100H
2. 对于从 D1100 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：D1100.078L
3. 对于从 D2000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D2000.055D
4. 对于从 D2200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D2200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”、“步进计时器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CV1000 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
操作标志	AC0000-AC2047	**布尔型**	只读

设备类型	范围	数据类型	访问
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型 布尔型	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串 字符串	读/写 只读
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串 字符串	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串 字符串	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串 字符串	读/写 只读
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写

设备类型	范围	数据类型	访问
计数器	C0000-C1023	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C0000.256H-C1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C0000.256L-C1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C0000.128D-C1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C0000.128E-C1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS0000-CS1023	**布尔型**	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存	D00000-D24575 D00000-D24574 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D24575.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D24575.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的	D00000.128D-	**字符串**	读/写

设备类型	范围	数据类型	访问
高位字节)	D24575.001D .l 是字符串长度，范围介于 1 到 128 个字符之间		
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D24575.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR0-DR2	**字**、短整型、BCD*	读/写
扩展数据内存 (现行存储器组)	E00000-E32765 E00000-E32764 Exxxxx.00-Exxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32765.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128D-E32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存	E00:00000-E07:32765 E00:00000-E07:32764 Ex:x.00-Exx:xxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H-E07:32765.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L-E07:32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E07:32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E07:32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
索引寄存器	IR0-IR2	**字**、短整型、BCD*	读/写
步进计时器	ST0000-ST1023	**字**、短整型、BCD*	读/写
步进计时器状态	STS0000-STS1023	**布尔型**	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器	T0000-T1023	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T0000.256H-T1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T0000.256L-T1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T0000.128D-T1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T0000.128E-T1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS0000-TS1023	**布尔型**	读/写
转换标志	TN0000-TN1023	**布尔型**	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型

±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CV1000 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”、“步进计时器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CV2000 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
操作标志	AC0000-AC2047	**布尔型**	只读
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C1023	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C0000.256H-C1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C0000.256L-C1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C0000.128D-C1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C0000.128E-C1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS0000-CS1023	**布尔型**	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存	D00000-D24575 D00000-D24574 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写

设备类型	范围	数据类型	访问
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D24575.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D24575.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D24575.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D24575.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR0-DR2	**字**、短整型、BCD*	读/写
扩展数据内存 (现行存储器组)	E00000-E32765 E00000-E32764 Exxxxx.00-Exxxxx.15	**字**、短整型、CD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32765.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128D-E32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存	E00:00000-E07:32765 E00:00000-E07:32764 Ex:x.00-Exx:xxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H-E07:32765.002H .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L-E07:32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E07:32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E07:32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
索引寄存器	IR0-IR2	**字**、短整型、BCD*	读/写
步进计时器	ST0000-ST1023	**字**、短整型、BCD*	读/写
步进计时器状态	STS0000-STS1023	**布尔型**	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器	T0000-T1023	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T0000.256H-T1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T0000.256L-T1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T0000.128D-T1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T0000.128E-T1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS0000-TS1023	**布尔型**	读/写
转换标志	TN0000-TN1023	**布尔型**	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CV2000 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”、“步进计时器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CVM1-CPU01 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。
*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写

设备类型	范围	数据类型	访问
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C000-C511	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C000.256H-C511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C000.256L-C511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C000.128D-C511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C000.128E-C511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS000-CS511	**布尔型**	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存	D0000-D8191 D0000-D8190 Dxxxx.00-Dxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D0000.256H-D8191.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D0000.256L-D8191.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D0000.128D-D8191.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D0000.128E-D8191.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR0-DR2	**字**、短整型、BCD*	读/写
索引寄存器	IR0-IR2	**字**、短整型、BCD*	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器	T000-T511	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T000.256H-T511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T000.256L-T511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T000.128D-T511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T000.128E-T511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS000-TS511	**布尔型**	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CVM1-CPU01 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D1000 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：D1000.100H
2. 对于从 D1100 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：D1100.078L
3. 对于从 D2000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D2000.055D

4. 对于从 D2200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D2200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CVM1-CPU11 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E	**字符串** **字符串**	读/写 只读

设备类型	范围	数据类型	访问
	.l 是字符串长度，范围介于 1 到 128 个字符之间		
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C1023	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C0000.256H-C1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C0000.256L-C1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C0000.128D-C1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C0000.128E-C1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS0000-CS1023	**布尔型**	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存	D00000-D24575 D00000-D24574 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D24575.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D24575.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D24575.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D24575.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR0-DR2	**字**、短整型、BCD*	读/写
索引寄存器	IR0-IR2	**字**、短整型、BCD*	读/写
暂存继电器	TR TR0-TR7	**字**、短整型 **布尔型**	读/写
计时器	T0000-T1023	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T0000.256H-T1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T0000.256L-T1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T0000.128D-T1023.001D .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 1 到 128 个字符之间		
计时器为字符串型 (仅使用每个字的低位字节)	T0000.128E-T1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS0000-TS1023	**布尔型**	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CVM1-CPU11 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CVM1-CPU21 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A255 A000-A254 A256-A511 A256-A510 A000.00-A000.15-A255.00-A255.15 A256.00-A256.15-A511.00-A511.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	读/写 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A255.002H A256.256H-A511.002H .l 是字符串长度，范围介	**字符串** **字符串**	读/写 只读

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A255.002L A256.256L-A511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A255.001D A256.128D-A511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A255.001E A256.128E-A511.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	读/写 只读
CIO	CIO0000-CIO2555 CIO0000-CIO2554 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO2555.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO2555.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO2555.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO2555.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C1023	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C0000.256H-C1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C0000.256L-C1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C0000.128D-C1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
计数器为字符串型 (仅使用每个字的低位字节)	C0000.128E-C1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS0000-CS1023	**布尔型**	读/写
CPU 总线链路	G000-G255 G000-G254 Gxxx.00-Gxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CPU 总线链路为字符串型 (字节排序由高到低)	G000.256H-G255.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (字节排序由低到高)	G000.256L-G255.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的高位字节)	G000.128D-G255.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CPU 总线链路为字符串型 (仅使用每个字的低位字节)	G000.128E-G255.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存	D00000-D24575 D00000-D24574 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D24575.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D24575.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D24575.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D24575.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR0-DR2	**字**、短整型、BCD*	读/写
扩展数据内存 (现行存储器组)	E00000-E32765 E00000-E32764	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写

设备类型	范围	数据类型	访问
	Exxxxx.00-Exxxxx.15	布尔型	
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32765.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128D-E32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存	E00:00000-E07:32765 E00:00000-E07:32764 Ex:x.00-Exx:xxxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H-E07:32765.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L-E07:32765.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E07:32765.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E07:32765.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
索引寄存器	IR0-IR2	字、短整型、BCD*	读/写
暂存继电器	TR TR0-TR7	字、短整型 布尔型	读/写
计时器	T0000-T1023	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T0000.256H-T1023.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写

设备类型	范围	数据类型	访问
计时器为字符串型 (字节排序由低到高)	T0000.256L-T1023.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T0000.128D-T1023.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T0000.128E-T1023.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
计时器状态	TS0000-TS1023	布尔型	读/写

*不支持数组。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例
IR0D @ Short = -50
IR0D @ Word = 8050 (MSB 设置)
IR0 @ BCD = 8050

IR0D @ Short = 50
IR0D @ Word = 50
IR0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CCM1-CPU21 模型支持以 ASCII 字符串的形式读取和写入多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”、“数据寄存器”、“索引寄存器”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

修改 32 位值 (双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠。因此，写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型，以避免发生重叠。例如，在使用双字型时，使用 D0、D2、D4 等字可以防止“字”重叠。

CS1 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A447 A000-A446 A448-A959 A448-A958 A000.00-A000.15-A447.00-A447.15 A448.00-A448.15-A959.00-A959.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	只读 读/写 只读 读/写
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A447.002H A448.256H-A959.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A447.002L A448.256L-A959.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A447.001D A448.128D-A959.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A447.001E A448.128E-A959.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	只读 读/写
CIO	CIO0000-CIO6143 CIO0000-CIO6142 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO6143.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO6143.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO6143.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO6143.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C4095	**BCD**、字、短整型	读/写
计数器为字符串型 (字节排序由高到低)	C0000.256H-C4095.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (字节排序由低到高)	C0000.256L-C4095.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的高位字节)	C0000.128D-C4095.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器为字符串型 (仅使用每个字的低位字节)	C0000.128E-C4095.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器状态	CS0000-CS4095	**布尔型**	读/写*
数据内存	D00000-D32767 D00000-D32766 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR00-DR15 DR00-DR14	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写*
扩展数据内存 (现行存储器组)	E00000-E32767 E00000-E32766	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写

设备类型	范围	数据类型	访问
	Exxxxx.00-Exxxxx.15	布尔型	
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128E-E32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存	E00:00000-E12:32767 E00:00000-E12:32766 Ex:x.00-Exx:xxxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H - E12:32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L - E12:32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	字符串	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E12:32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E12:32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	字符串	读/写
保持继电器	H0000-H1535 H0000-H1534 Hxxxx.00-Hxxxx.15	字、短整型、BCD 长整型、双字型、LBCD、浮点型 布尔型	读/写
保持继电器为字符串型 (字节排序由高到低)	H0000.256H-H1535.002H .l 是字符串长度，范围介	字符串	读/写

设备类型	范围	数据类型	访问
	于 2 到 256 个字符之间		
保持继电器为字符串型 (字节排序由低到高)	H0000.256L-H1535.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的高位字节)	H0000.128D-H1535.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器为字符串型 (仅使用每个字的低位字节)	H0000.128E-H1535.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
索引寄存器	IR00-IR15	**双字型**、长整型、LBCD、浮点型	读/写*
任务标志	TK00-TK31	**布尔型**	只读
计时器	T0000-T4095	**BCD**、字、短整型	读/写
计时器为字符串型 (字节排序由高到低)	T0000.256H-T4095.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (字节排序由低到高)	T0000.256L-T4095.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的高位字节)	T0000.128D-T4095.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器为字符串型 (仅使用每个字的低位字节)	T0000.128E-T4095.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计时器状态	TS0000-TS4095	**布尔型**	读/写*
工作继电器	W000-W511 W000-W510 Wxxx.00-Wxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
工作继电器为字符串型 (字节排序由高到低)	W000.256H-W511.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
工作继电器为字符串型 (字节排序由低到高)	W000.256L-W511.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
工作继电器为字符串型 (仅使用每个字的高位字节)	W000.128D-W511.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
工作继电器为字符串型 (仅使用每个字的低位字节)	W000.128E-W511.001E .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 1 到 128 个字符之间		

*修改 32 位值 (如双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠：写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型以防止重叠。例如，为了防止“字”重叠，用户应针对“双字型”使用 D0、D2、D4 等字。此情况的例外为 IR 标记，它是本机 32 位值，对于 CS1 系列的 PLC，采用 MSB 到 LSB 字节排序。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例

D0D @ Short = -50
D0D @ Word = 8050 (MSB 设置)
D0 @ BCD = 8050

D0D @ Short = 50
D0D @ Word = 50
D0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CS1 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

写入 DR 和 IR 寄存器

仅当设备处于“编程模式”时，才可以写入 DR 和 IR 寄存器。如果设备处于“运行模式”，并且尝试写入其中一个寄存器，则设备中的值不会更改。因为写入成功，所以不会返回任何错误消息。设备的值不会更改，因为设备处于“运行模式”。

CJ1 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A447 A000-A446 A448-A959 A448-A958 A000.00-A000.15-A447.00-A447.15 A448.00-A448.15-A959.00-A959.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型**	只读 读/写 只读 读/写
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A447.002H A448.256H-A959.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A447.002L A448.256L-A959.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A447.001D A448.128D-A959.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	只读 读/写
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A447.001E A448.128E-A959.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串**	只读 读/写
CIO	CIO0000-CIO6143 CIO0000-CIO6142 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO6143.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO6143.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO6143.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写

设备类型	范围	数据类型	访问
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO6143.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C4095	**BCD**、字、短整型	读/写
计数器状态	CS0000-CS4095	**布尔型**	读/写*
数据内存	D00000-D32767 D00000-D32766 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR00-DR15 DR00-DR14	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写*
扩展数据内存 (现行存储器组)	E00000-E32767 E00000-E32766 Exxxxx.00-Exxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128D-E32767.001D .l 是字符串长度，范围介	**字符串**	读/写

设备类型	范围	数据类型	访问
	于 1 到 128 个字符之间		
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存	E00:00000-E12:32767 E00:00000-E12:32766 Ex:x.00-Exx:xxxxx.15	**字**、短整型、BCD **布尔型**	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H-E12:32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L-E12:32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E12:32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E12:32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	H0000-H1535 H0000-H1534 Hxxxx.00-Hxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
索引寄存器	IR00-IR15	**双字型**、长整型、LBCD、浮点型	读/写*
任务标志	TK00-TK31	**布尔型**	只读
计时器	T0000-T4095	**BCD**、字、短整型	读/写
计时器状态	TS0000-TS4095	**布尔型**	读/写*
工作继电器	W000-W511 W000-W510 Wxxx.00-Wxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写

*修改 32 位值 (如双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠：写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型以防止重叠。例如，为了防止“字”重叠，用户应针对“双字型”使用 D0、D2、D4 等字。此情况的例外为 IR 标记，它是本机 32 位值，对于 CJ1 系列的 PLC，采用 MSB 到 LSB 字节排序。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例

D0D @ Short = -50
D0D @ Word = 8050 (MSB 设置)
D0 @ BCD = 8050

D0D @ Short = 50
D0D @ Word = 50
D0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CJ1 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L

3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

写入 DR 和 IR 寄存器

仅当设备处于“编程模式”时，才可以写入 DR 和 IR 寄存器。如果设备处于“运行模式”，并且尝试写入其中一个寄存器，则设备中的值不会更改。因为写入成功，所以不会返回任何错误消息。设备的值不会更改，因为设备处于“运行模式”。

CJ2 寻址

动态定义标记的默认数据类型以**粗体**显示 (若适用)。

*有关注意事项和限制的详细信息，请参阅 **BCD 支持**、**字符串支持**以及**数组支持**。*

设备类型	范围	数据类型	访问
辅助继电器	A000-A447 A000-A446 A448-A1471 A448-A1470 A10000-A11535 A10000-A11534 A000.00-A000.15-A447.00-A447.15 A448.00-A448.15-A1471.00-A1471.15 A10000.00-A10000.15-A11535.00-A11535.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型** **布尔型** **布尔型**	只读 读/写 只读 只读 读/写 只读
辅助继电器为字符串型 (字节排序由高到低)	A000.256H-A447.002H A448.256H-A1471.002H A10000.256H-A11535.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串** **字符串**	只读 读/写 只读
辅助继电器为字符串型 (字节排序由低到高)	A000.256L-A447.002L A448.256L-A1471.002L A10000.256L-A11535.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串** **字符串** **字符串**	只读 读/写 只读
辅助继电器为字符串型 (仅使用每个字的高位字节)	A000.128D-A447.001D A448.128D-A1471.001D A10000.128D-A11535.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串** **字符串**	只读 读/写 只读
辅助继电器为字符串型 (仅使用每个字的低位字节)	A000.128E-A447.001E A448.128E-A1471.001E A10000.128E-A11535.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串** **字符串** **字符串**	只读 读/写 只读
CIO	CIO0000-CIO6143 CIO0000-CIO6142 CIOxxxx.00-CIOxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写

设备类型	范围	数据类型	访问
CIO 内存为字符串型 (字节排序由高到低)	CIO0000.256H-CIO6143.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (字节排序由低到高)	CIO0000.256L-CIO6143.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的高位字节)	CIO0000.128D-CIO6143.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
CIO 内存为字符串型 (仅使用每个字的低位字节)	CIO0000.128E-CIO6143.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
计数器	C0000-C4095	**BCD**、字、短整型	读/写
计数器状态	CS0000-CS4095	**布尔型**	读/写*
数据内存	D00000-D32767 D00000-D32766 Dxxxxx.00-Dxxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
数据内存为字符串型 (字节排序由高到低)	D00000.256H-D32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (字节排序由低到高)	D00000.256L-D32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的高位字节)	D00000.128D-D32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据内存为字符串型 (仅使用每个字的低位字节)	D00000.128E-D32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
数据寄存器	DR00-DR15 DR00-DR14	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写*
扩展数据内存 (现行存储器组)	E00000-E32767 E00000-E32766 Exxxxx.00-Exxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型	读/写

设备类型	范围	数据类型	访问
		布尔型	
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由高到低)	E00000.256H-E32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (字节排序由低到高)	E00000.256L-E32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的高位字节)	E00000.128D-E32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存 (现行存储器组) 为字符串型 (仅使用每个字的低位字节)	E00000.128E-E32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存	E00:00000-E24:32767 E00:00000-E24:32766 Ex:x.00-Exx:xxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
扩展数据内存为字符串型 (字节排序由高到低)	E00:00000.256H-E24:32767.002H .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (字节排序由低到高)	E00:00000.256L-E24:32767.002L .l 是字符串长度，范围介于 2 到 256 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的高位字节)	E00:00000.128D-E24:32767.001D .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
扩展数据内存为字符串型 (仅使用每个字的低位字节)	E00:00000.128E-E24:32767.001E .l 是字符串长度，范围介于 1 到 128 个字符之间	**字符串**	读/写
保持继电器	H0000-H1535 H0000-H1534 Hxxxx.00-Hxxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写
索引寄存器	IR00-IR15	**双字型**、长整型、LBCD、浮点型	读/写*
任务标志	TK00-TK127	**布尔型**	只读

设备类型	范围	数据类型	访问
计时器	T0000-T4095	**BCD**、字、短整型	读/写
计时器状态	TS0000-TS4095	**布尔型**	读/写*
工作继电器	W000-W511 W000-W510 Wxxx.00-Wxxx.15	**字**、短整型、BCD 长整型、双字型、LBCD、浮点型 **布尔型**	读/写

*修改 32 位值 (如双字型、长整型、LBCD 和浮点型) 时请小心。允许这些数据类型的每个地址在设备内的字偏移处开始。因此，双字型 D0 和 D1 在字 D1 处重叠：写入 D0 还会修改保存在 D1 中的值。建议使用这些数据类型以防止重叠。例如，为了防止“字”重叠，用户应针对“双字型”使用 D0、D2、D4 等字。此情况的例外为 IR 标记，它是本机 32 位值，对于 CJ1 系列的 PLC，采用 MSB 到 LSB 字节排序。

BCD 支持

16 位和 32 位数据可以作为有符号和无符号 BCD 值引用。

通过将 "D" 附加到任何“字”或“双字型”标记地址或使用 BCD 和 LBCD 数据类型实现对无符号 BCD 的支持。值范围如下：

支持无符号 BCD 的字
0 - 9999

支持无符号 BCD 的双字型
0 - 99999999

通过将 "D" 附加到任何“短整型”或“长整型”标记地址实现对有符号 BCD 的支持。“短整型”/“长整型”的最高有效位对应负值时设置为 1，对应正值时设置为 0。值范围如下：

支持有符号 BCD 的短整型
±7999

支持有符号 BCD 的长整型
±79999999

示例

D0D @ Short = -50
D0D @ Word = 8050 (MSB 设置)
D0 @ BCD = 8050

D0D @ Short = 50
D0D @ Word = 50
D0 @ BCD = 50

注意：布尔型、浮点型、字符串数据类型不支持将 "D" 附加到标记地址。此外，不能将 "D" 附加到 BCD 或 LBCD 标记，或者默认数据类型为 BCD 或 LBCD 的标记 (如“计数器”和“计时器”标记)。

字符串支持

CJ1 型号支持以 ASCII 字符串的形式读写多种设备类型。当将数据内存用于字符串数据时，每个寄存器包含 ASCII 数据的两个字节 (两个字符)。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 256 个字符，并代替位数输入。字符串长度不得超过已分配给设备的请求大小。长度必须输入为

偶数。范围同样不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此为每个字符传输 2 个字节。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "H" 或 "L" 附加到地址来指定字节顺序。

如果每个寄存器使用一字节 ASCII 数据，则字符串长度可以为 1 到 128 个字符，并代替位数输入。4 倍字符串长度不能超过已分配给设备的请求大小。字符串跨越的寄存器范围不得超过设备类型的范围。通过将 "D" 或 "E" 附加到地址来指定要在寄存器内使用的字节。有关详细信息，请参阅设置。

示例

1. 对于从 D01000 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：D01000.100H
2. 对于从 D01100 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：D01100.078L
3. 对于从 D02000 开始、长度为 55 字节并采用仅高位字节的字符串，要对其进行寻址，请输入：D02000.055D
4. 对于从 D02200 开始、长度为 37 字节并采用仅低位字节的字符串，要对其进行寻址，请输入：D02200.037E

数组支持

除了“布尔型”和“字符串”之外，所有数据类型都支持数组。有关数组寻址的方法有两种。以下示例为根据数据内存位置。

Dxxxx [行数] [列数]
DMxxxx [列数] (此方法假定“行数”等于 1)

行数、列数与以字节为单位的数据大小的乘积不能超过已分配给设备的请求大小。由于此驱动程序使用 ASCII 协议，因此每个字、短整型和 BCD 对应 4 个字节，而每个双字型、长整型、LBCD 和浮点型对应 8 个字节。例如，10 X 10 字数组的数组大小为 400 字节，需要的请求大小为 512。下一个较小的请求大小选择 256 过小。有关详细信息，请参阅设置。

写入 DR 和 IR 寄存器

仅当设备处于“编程模式”时，才可以写入 DR 和 IR 寄存器。如果设备处于“运行模式”，并且尝试写入其中一个寄存器，则设备中的值不会更改。因为写入成功，所以不会返回任何错误消息。设备的值不会更改，因为设备处于“运行模式”。

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。。关于如何筛选和排序“事件日志”详细信息视图，请参阅 OPC 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。

提示：来自数据源 (如第三方软件，包括数据库) 的消息通过事件日志显示。故障排除步骤应包括在网上和供应商文档中研究这些消息。

设备响应错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应本地节点错误。| 主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应目的节点错误。| 主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应通信错误。| 主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备无法处理命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应路由表错误。| 主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应命令格式错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：

警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应命令参数错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应不可读取。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：
主错误和子错误代码

设备响应单元错误。| 主错误 = <代码>，子错误 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：

有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：

主错误和子错误代码

设备无法接受命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：

警告

可能的原因：

错误代码指示可能的原因。

可能的解决方案：

有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：

主错误和子错误代码

访问权限被拒绝。| 标记地址 = '<地址>'，主错误 = <代码>，子错误 = <代码>。

错误类型：

警告

可能的原因：

错误代码指示可能的原因。

可能的解决方案：

有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：

主错误和子错误代码

设备响应不可写入。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。

错误类型：

警告

可能的原因：

错误代码指示可能的原因。

可能的解决方案：

有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

也可以看看：

主错误和子错误代码

无法写入标记。设备处于“运行模式”。| 标记地址 = '<地址>'。

错误类型：
警告

设备响应致命 CPU 单元错误。| 终止代码 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

设备响应非致命 CPU 单元错误。| 终止代码 = <代码>。

错误类型：
警告

可能的原因：
错误代码指示可能的原因。

可能的解决方案：
有关可行解决方案与更正措施的说明，请参阅“主错误和子错误代码”。

写入标记后无法将设备设置回“运行模式”。| 标记地址 = '<地址>'。

错误类型：
警告

错误掩码定义

B = 检测到硬件断点
F = 框架错误
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

主错误代码和子错误代码

主代码	子代码	检查点	可能的根源	修正
00：正常完成	00：正常完成			
	01：服务已取消		服务已取消。	检查第三个节点中目标区域的容量。
		数据链路状况	服务已取消。	检查数据链路的状况。

主代码	子代码	检查点	可能的根源	修正
01：本地节点错误	01：本地节点不在网络中	本地节点的网络状况	本地节点未加入网络。	将节点连接到网络。
	02：令牌超时	最大节点地址	令牌未到达。	将本地节点设置在最大节点地址内。
	03：重试失败		在指定的重试次数内无法发送。	在节点之间执行通信测试，如果测试失败，重新检查系统环境。
	04：发送帧过多	已启用的发送帧数	无法发送，因为超出了最大事件帧数。	检查网络上的事件执行，并减少每个循环的事件数目。增加事件帧的最大数目。
	05：节点地址范围错误	节点地址	发生节点地址设置错误。	检查旋转开关的设置，确保地址在范围内，并且每个地址在同一网络中只设置一次。
	06：节点地址重复	节点地址	相同的节点地址已在同一网络中设置了两次。	更改其中一个具有相同地址的节点的地址。
02：目标节点错误	01：目标节点不在网络中	单元上的 INS 指示器	目标节点不在网络中。	将目标节点添加到网络。
	02：单元缺失	指令控制数据	具有指定单元地址的单元不存在。	检查目标单元地址。
	03：第三个节点缺失	指令控制数据	第三个节点不存在。	检查第三个节点的单元地址。检查 CMND (490) 的发送数据中第三个节点的节点地址。
		命令数据	已指定广播。	为第三个节点仅指定一个节点。
	04：目标节点忙碌		目标节点忙碌。	增加重试次数或检查系统，使目标节点不会接收如此多的消息。
	05：响应超时		消息已被噪声破坏。	增加重试次数或测试节点之间的通信，以查看是否存在过多噪声。
		指令控制数据	响应监控时间过短。	增加响应监控时间的长度。
		错误历史记录	发送/接收帧被放弃。	根据错误历史记录，采取适当措施。
03：控制器错误	01：通信控制器错误	单元/电路板指示器	通信控制器中发生错误。	根据相关单元/电路板的操作手册，采取适当措施。
	02：CPU 单元错误	目标节点处的 CPU 单元指示器	目标 CPU 单元中出现 CPU 错误。	按照相应的操作手册清除 CPU 单元中的错误。
	03：控制器错误	电路板指示器	由于电路板中出现错误，因此未返回响应。	检查网络通信状况并重新启动电路板。如果问题仍然存在，请更换电路板。
	04：单元编号错误	单元编号	单元编号设置不正确。	正确设置旋转开关，确保单元编号在相应范围内，并且每个编号只使用一次。

主代码	子代码	检查点	可能的根源	修正
04：服务不受支持	01：未定义命令	命令代码	单元/电路板不支持指定的命令代码。	检查命令代码。
	02：不受型号/版本支持	单元型号和版本	无法执行命令，因为型号或版本不正确。	检查型号编号和版本。
05：路由表错误	01：目标地址设置错误	路由表	未在路由表中设置目标网络或节点地址。	将目标网络和节点注册到路由表中。
	02：没有路由表	路由表	无法进行中继，因为没有路由表。	在源节点、标识节点和继电器节点中设置路由表。
	03：路由表错误	路由表	路由表中存在错误。	正确设置路由表。
	04：继电器过多	网络配置	尝试发送到超过 3 个网络以外的网络	重新构建网络或更改路由表，以在 3 个或更少的网络范围内发送命令。
10：命令格式错误	01：命令过长	命令数据	命令超出最大允许长度。	检查命令格式并修正命令数据。
	02：命令过短	命令数据	命令低于最小允许长度。	检查命令格式并修正命令数据。
	03：元素/数据不匹配	命令数据	指定的元素数量与写入数据项的数量不同。	检查元素数量，并为每个元素设置数据。
	04：命令格式错误	命令数据	使用了错误格式。	检查命令格式并修正命令数据。
	05：标题错误	路由表	本地节点中的继电器表或继电器节点中的本地网络表不正确。	正确设置路由表。
11：参数错误	01：区域分类缺失	命令数据中的内存区域代码	内存区域中指定的字不存在，或没有 EM 区域。	检查命令中的内存区域和参数代码，并修正命令数据。
	02：访问大小错误	命令数据中的访问大小规范	访问大小规范不正确，或指定了奇数字地址。	检查内存区域和访问大小，并修正访问大小。
	03：地址范围错误	命令数据中的起始地址	命令进程中的起始地址超出可访问区域。	检查正在处理的区域，并设置正确的范围。
	04：超出地址范围	命令数据中的起始地址和元素数量	命令进程中的结束地址超出可访问区域。	检查正在处理的区域，并设置正确的范围。
		数据链路表	总字数超出限制。	修正数据链路表。
	06：程序缺失	命令数据中的程序编号	未指定十六进制 FFFF。	指定十六进制 FFFF。
	09：关系错误	命令数据	命令数据中元素的大小关系不正确。	检查命令数据，并修正元素之间的关系。

主代码	子代码	检查点	可能的根源	修正
		数据链路表	将未设置在常用链接参数中的节点设置为刷新参数。	修正数据链路表。
	0A：数据访问重复	CPU 单元中的 I/O 访问	在数据跟踪期间指定了差异监控，或在差异监控期间指定了数据跟踪。	取消当前进程或等待其结束，然后再执行命令。
		数据链路表	多次指定了相同的节点地址。	修正数据链路表。
	0B：响应过长	命令数据中的元素数量	响应格式超出最大允许长度。	检查命令格式并修正元素数量。
	0C：参数错误	命令数据中的参数	其中一个参数设置存在错误。	检查命令数据并修正参数。
		数据链路表文件	文件中存在错误。	检查文件的内容。
20：无法读取	02：受保护		程序区域受到保护。	从编程设备解除保护，然后执行命令。
	03：表缺失	表	表尚未注册。	注册表。
			表中存在错误。	修正表。
	04：数据缺失		搜索数据不存在。	
	05：程序缺失	命令数据中的程序编号	指定了不存在的程序编号。	检查程序编号，并指定有效的编号。
	06：文件缺失	文件名和文件设备	文件在指定的文件设备中不存在。	检查路径和文件名，并对它们进行更正。
	07：数据不匹配	正在比较的内存内容	正在比较的某一数据不相同。	检查内存内容，并使用正确的数据。
			文件读取操作失败。	检查文件的内容。
21：无法写入	01：只读		指定的区域是只读的。	如果该区域是使用开关设置进行保护的，请解除保护，然后执行命令。如果区域为永久只读，则无法执行该命令。
	02：受保护，无法写入数据链路表		程序区域受到保护。	从编程设备解除保护，然后执行命令。
		PLC 设置	由于已指定自动数据链路表生成，因此无法进行写入。	更改 PLC 设置，以便可以手动写入数据链路表。
	03：无法注册	文件设备中的文件数	无法创建文件，因为已超出限制。	删除任何不需要的文件，或创建更多的文件内存。
		打开的文件数	打开的最大文件数已达系统限制。	关闭一个或多个文件，然后执行命令。
	05：程序缺失	命令数据中的程序编号	指定了不存在的程序编号。	检查程序编号，并指定有效的编号。
	06：文件缺失	文件名	文件在指定的文件设备中不存在。	修正文件名，然后执行命令。

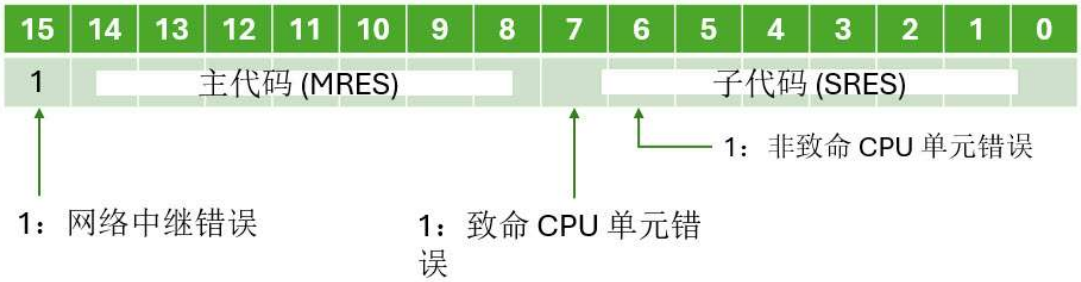
主代码	子代码	检查点	可能的根源	修正
	07：文件名已经存在	文件名	在指定的文件设备中，具有相同名称的文件已经存在。	更改正在写入的文件的名称，然后执行命令。
	08：无法更改	正在更改的内存内容	无法进行更改，因为这样做会产生问题。	
22：在当前模式下不可执行	01：在执行期间无法进行		模式不正确。	检查模式。
		数据链路状况	数据链路正在运行。	检查数据链路的状态。
	02：运行时无法实现		模式不正确。	检查模式。
		数据链路状况	数据链路处于活动状态。	检查数据链路的状态。
	03：PLC 模式错误		PLC 处于“编程”模式。	检查 PLC 和计算机的模式。
	04：PLC 模式错误		PLC 处于“调试”模式。	检查 PLC 和计算机的模式。
	05：PLC 模式错误		PLC 处于“监视”模式。	检查 PLC 和计算机的模式。
	06：PLC 模式错误		PLC 处于“运行”模式。	检查 PLC 和计算机的模式。
	07：指定的节点不是轮询节点		指定的节点不是轮询节点。	检查节点是否用作网络的轮询节点。
	08：无法执行步骤		模式不正确。	检查步骤状况。
23：没有这样的设备	01：文件设备缺失	单元配置	指定的内存并不作为文件设备而存在。	安装内存，或将 EM 格式设置为文件内存。
	02：内存缺失		没有文件内存。	检查文件内存，查看其是否已安装。
	03：时钟缺失		没有时钟。	检查型号。
24：无法启动/停止	01：表缺失	数据链路表	数据链路表尚未注册或其中包含错误。	设置数据链路表。
25：单元错误	02：内存错误	正在处理的内存内容	内存中的内容包含错误。	将正确的内容传输到内存。
	03：I/O 设置错误	I/O 单元配置	注册的 I/O 表与实际的 I/O 配置不一致。	修正 I/O 表或 I/O 配置。
	04：I/O 点过多	已注册 I/O 表中的 I/O 数量	注册的 I/O 点和远程 I/O 点过多。	更改注册的 I/O 表，使其在限制范围内。
	05：CPU 总线错误	CPU 总线	CPU 和 CPU 总线单元之间的数据传输出现错误。	检查单元、电路板和缆，确保它们已正确连接，然后执行“清除错误”命令。
	06：I/O 重复	PLC 设置中的机架编号、单元编号和	多次设置了相同的编号/地址。	检查 PLC 设置并更正编号/地址，以便确保每个编号/地址仅

主代码	子代码	检查点	可能的根源	修正
		I/O 地址		使用一次。
	07：I/O 总线错误	I/O 总线	CPU 和 I/O 单元之间的数据传输出现错误。	检查单元、电路板和缆，确保它们已正确连接，然后执行“清除错误”命令。
	09：SYSMAC BUS/2 错误	SYSMAC BUS/2 传输路径	SYSMAC BUS/2 线上的数据传输出现错误。	检查单元、电路板和缆，确保它们已正确连接，然后执行“清除错误”命令。
	0A：CPU 总线单元错误	CPU 总线单元传输路径	CPU 总线单元的数据传输出现错误。	检查单元、电路板和缆，确保它们已正确连接，然后执行“清除错误”命令。
	0D：SYSMAC BUS 编号重复	字设置	多次分配了相同的字。	检查 I/O 表并修正分配。
	0F：内存错误	正在处理的内存状况	内部内存、内存卡或 EM 文件内存中发生内存错误。	对于内部内存，写入正确的数据，然后执行命令。对于内存卡或 EM 文件内存，文件数据已被破坏。执行“文件内存格式”命令。如果问题仍然存在，请更换内存。
	10：SYSMAC BUS 终结器		未设置终结器。	正确设置终结器。
26：命令错误	01：无保护	程序区域的命令保护	指定的区域不受保护。	之前曾尝试清除现在无保护区域的保护，即：没有理由清除保护。
	02：密码不正确		指定的密码不正确。	指定正确密码。
	04：受保护		指定的区域受保护。	从编程设备清除保护，然后执行命令。
		正在执行的命令数	接收命令的节点已经在处理 5 个命令。	等待当前进程结束或强制结束当前进程，然后执行命令。
	05：服务已经在执行		正在执行服务。	等待服务结束或强制结束服务，然后执行命令。
	06：服务已停止		服务并未正在执行。	如有必要，请启动服务。
	07：没有执行权限	单元/电路板上的 LNK 指示器	尚未获得执行服务的权限。	本地节点不在数据链路中。从加入数据链路的节点执行命令。
			由于发生缓冲器错误，响应未返回。	重新启动电路板。如果问题仍然存在，请更换电路板。
	08：设置未完成	执行前所需的设置	尚未完成执行服务前所需的设置。	执行所需的设置。
	09：未设置所需项	命令数据	尚未在命令数据中设置所需的元素。	检查命令格式，并在命令数据中设置所需的元素。
	0A：已定义编号	程序区域中程序的操作编号和转变编	指定的操作/转变编号已注册到上一个	检查操作/转变编号，将其改为未使用的编号，然后执行命

主代码	子代码	检查点	可能的根源	修正
		号	程序中。	令。
	0B：错误未清除	正在清除的错误的原因	尚未移除错误的原因。	移除错误的原因，然后执行“清除错误”。
30：访问权限错误	01：没有访问权限		访问权限由另一个设备保持。(正在从另一个节点执行在线编辑，或者已由另一个节点执行“访问权限获取”或“访问权限强制获取”。)	等待至访问权限被释放，然后执行命令。可以执行“访问权限获取”或“访问权限强制获取”以获得访问权限，但这可能会对由先前保持访问权限的节点所执行的处理产生不利影响。
40：取消	01：服务已取消		服务已被“取消”(中止) 命令取消。	

标志

在同时可能处于“开”值的结束代码字 (位 6、7 和 15) 中有特定标志。下图显示了结束代码字的内容。



如果位 6 或 7 为 ON，在目标 CPU 单元中出现错误。如果发生这种情况，请参阅操作手册以找到出现错误的 CPU 单元，然后消除错误原因。

如果位 15 处于“开”值，则在网络继电器操作期间发生了错误。结束代码中包含一个附加的两字节 (一个字) 网络继电器错误代码，可用于确定继电器错误的位置。下图显示了网络继电器错误的响应数据。该信息可以用于确定发生错误的节点，以便采取适当的措施。

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D+1	1	主代码 (MRES)								子代码 (SRES)						
D+2	1	错误网络地址								错误节点地址						

1：非致命 CPU 单元错误

1：网络中继错误

1: 致命 CPU 单元错误

错误网络地址：00 到 7F (0 到 127)
错误节点地址

以太网:	01 到 7E (1 – 128)
SYSMAC NET：	01 到 7E (1 – 126)
Controller Link：	01 到 20 (1 – 32)
SYSMAC LINK：	01 到 3E (1 – 62)

索引

“

“标识” 6
“无通信的操作” 8

B

BCD 19

C

C1000H 寻址 28
C2000H 寻址 32
C200H 寻址 20
C500 寻址 25
CJ1 寻址 71
CJ2 寻址 76
COM ID 7
COM 端口 7
CS 和 TS 写入 14
CS1 寻址 65
CV1000 寻址 40
CV2000 寻址 46
CV500 寻址 36
CVM1-CPU11 寻址 55
CVM1-CPU21 寻址 59

D

DTR 8

F

FINS 通信服务 16
FINS 网络 16

FINS 消息 16
FINS 协议 5

I

I/O 错误 85
ID 11

L

LBCD 19

R

RS-485 8
RTS 8
RX 缓冲区 85

S

Sysmac Way 5

T

TX 缓冲区 85

V

VM1 CPU01 寻址 51

报

报告通信错误 8

本

本地 17

标

标记计数 7, 12
标识 10
标志 91

波

波特率 7

不

不扫描，仅按需求轮询 12

布

布尔型 19

操

操作模式 11
操作行为 8

超

超时前的尝试次数 13

串

串行端口设置 7
串行通信 7

错

错误掩码定义 85

地

地址说明 20

调

调制解调器 7-8
调制解调器设置 8

定

定时 12

读

读取处理 9

短

短整型 19

访

访问权限被拒绝。| 标记地址 = '<地址>'，主错误 = <代码>，子错误 = <代码>。 84

非

非规范浮点数处理 10

浮

浮点型 19

概

概述 5

共

共享 7

故

故障时降级 14

关

关闭空闲连接 8
关闭前空闲时间 8

将

将 PLC 设置为监视器模式，写入、重置以运行 14
将 PLC 设置为监视器模式，执行写入 14

降

降级超时 14
降级期间 14
降级时放弃请求 14

框

框架 85

来

来自缓存的初始更新 12

连

连接超时 8, 13
连接类型 7

流

流量控制 5, 8

路

路由表 17

轮

轮询延迟 8

模

模拟 11

奇

奇偶校验 8, 85

请

请求超时 13
请求大小 15

驱

驱动程序 11

冗

冗余 15

扫

扫描模式 12

上

上升 8

设

设备间延迟 10
设备属性 - 定时 12
设备属性 - 冗余 15
设备属性 - 自动降级 13
设备无法处理命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 82
设备无法接受命令。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 84
设备响应本地节点错误。| 主错误 = <代码>，子错误 = <代码>。 81
设备响应不可读取。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 83
设备响应不可写入。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 84
设备响应错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 81
设备响应单元错误。| 主错误 = <代码>，子错误 = <代码>。 83
设备响应非致命 CPU 单元错误。| 终止代码 = <代码>。 85
设备响应路由表错误。| 主错误 = <代码>，子错误 = <代码>。 82
设备响应命令参数错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 83
设备响应命令格式错误。| 标记地址 = '<地址>'，数据大小 = <字节数>，主错误 = <代码>，子错误 = <代码>。 82
设备响应目的节点错误。| 主错误 = <代码>，子错误 = <代码>。 81
设备响应通信错误。| 主错误 = <代码>，子错误 = <代码>。 82
设备响应致命 CPU 单元错误。| 终止代码 = <代码>。 85
设置 5

事

事件日志消息 81

数

数据类型说明 19

数据收集 11
数据位 7

双

双字 19

替

替换为零 10

停

停止位 8

通

通道分配 11
通道属性 - 常规 6
通道属性 - 串行通信 7
通道属性 - 高级 9
通道属性 - 写入优化 9
通信参数 5, 14
通信超时 12
通信协议 5

网

网络 5

未

未修改 10

无

无 7

无法写入标记。设备处于“运行模式”。| 标记地址 = '<地址>'。 85
无符号 19

物

物理介质 7

下

下降 8

写

写入标记后无法将设备设置回“运行模式”。| 标记地址 = '<地址>'。 85
写入非布尔标记的最新值 9
写入失败，日志消息 14
写入所有标记的所有值 9
写入所有标记的最新值 9

型

型号 11

溢

溢出 85

硬

硬件断点 85

优

优化方法 9

有

有符号 19

远

远程 17

运

运行模式设置 14

占

占空比 9

长

长整型 19

诊

诊断 6

主

主错误代码和子错误代码 85
主机链路界面 5
主机链路通信 17

自

自动拨号 8
自动降级 13

字

字 19
字符串 19

遵

遵循标签指定的扫描速率 12