

# Modicon M241 Logic Controller

## 编程指南

04/2014



---

本文档中提供的信息包含有关此处所涉及产品之性能的一般说明和 / 或技术特性。本文档并非用于（也不代替）确定这些产品对于特定用户应用场合的适用性或可靠性。任何此类用户或集成者都有责任就相关特定应用场合或使用方面对产品执行适当且完整的风险分析、评估和测试。**Schneider Electric** 或其任何附属机构或子公司对于误用此处包含的信息而产生的后果概不负责。如果您有关于改进或更正此出版物的任何建议，或者从中发现错误，请通知我们。

未经 **Schneider Electric** 明确书面许可，不得以任何形式、通过任何电子或机械手段（包括影印）复制本文档的任何部分。

在安装和使用本产品时，必须遵守国家、地区和当地的所有相关的安全法规。出于安全方面的考虑和为了帮助确保符合归档的系统数据，只有制造商才能对各个组件进行维修。

当设备用于具有技术安全要求的应用场合时，必须遵守有关的使用说明。

未能使用 **Schneider Electric** 软件或认可的软件配合我们的硬件，则可能导致人身伤害、损害或不正确的操作结果。

不遵守此信息可能导致人身伤害或设备损坏。

© 2014 Schneider Electric。保留所有权利。



安全信息 . . . . .	7
关于本书 . . . . .	9
<b>章 1 关于 Modicon M241 Logic Controller . . . . .</b>	<b>13</b>
M241 Logic Controller 描述 . . . . .	13
<b>章 2 如何配置控制器 . . . . .</b>	<b>19</b>
如何配置控制器 . . . . .	19
<b>章 3 库 . . . . .</b>	<b>23</b>
库 . . . . .	23
<b>章 4 支持的标准数据类型 . . . . .</b>	<b>25</b>
支持的标准数据类型 . . . . .	25
<b>章 5 存储器映射 . . . . .</b>	<b>27</b>
控制器存储器结构 . . . . .	28
RAM 存储器结构 . . . . .	29
闪存结构 . . . . .	31
重新定位表 . . . . .	34
<b>章 6 任务 . . . . .</b>	<b>37</b>
最大任务数 . . . . .	38
任务配置屏幕 . . . . .	39
任务类型 . . . . .	41
系统和任务看门狗 . . . . .	44
任务优先级 . . . . .	45
缺省任务配置 . . . . .	48
<b>章 7 控制器状态和行为 . . . . .</b>	<b>49</b>
7.1 控制器状态图 . . . . .	50
控制器状态图 . . . . .	51
7.2 控制器状态介绍 . . . . .	55
控制器状态介绍 . . . . .	55
7.3 状态转换和系统事件 . . . . .	59
控制器状态和输出行为 . . . . .	60
通过命令进行状态转换 . . . . .	63
错误检测、类型和管理 . . . . .	69
剩余变量 . . . . .	70

<b>章 8</b>	<b>控制器设备编辑器</b>	<b>71</b>
	控制器参数	72
	控制器选择	74
	PLC 设置	75
	服务	77
<b>章 9</b>	<b>内嵌输入和输出配置</b>	<b>79</b>
	嵌入式 I/O 配置	79
<b>章 10</b>	<b>专用功能配置</b>	<b>85</b>
	专用配置概述	86
	计数功能	88
	脉冲发生器内嵌功能	90
<b>章 11</b>	<b>卡盒配置</b>	<b>93</b>
	TMC4 卡盒配置	93
<b>章 12</b>	<b>扩展模块配置</b>	<b>95</b>
	I/O 配置一般做法	96
	I/O 总线配置	97
	TM4 扩展模块配置	98
	TM3/TM2 扩展模块配置	99
<b>章 13</b>	<b>以太网配置</b>	<b>101</b>
13.1	以太网服务	102
	以太网服务	103
	IP 地址配置	105
	Modbus TCP 服务器 / 客户端	110
	Web 服务器	112
	FTP 服务器	125
	SNMP	127
13.2	防火墙配置	128
	简介	129
	动态更改过程	131
	防火墙行为	132
	脚本文件语法	134
13.3	以太网可选设备	138
	以太网管理器	139
	Ethernet/IP 设备	140
	Modbus TCP 从站设备	159

<b>章 14</b>	<b>串行线路配置</b> . . . . .	<b>163</b>
	串行线路配置 . . . . .	164
	SoMachine 网络管理器 . . . . .	166
	Modbus 管理器 . . . . .	167
	ASCII 管理器 . . . . .	171
	Modbus IOScanner . . . . .	173
	在 Modbus IOScanner 上添加设备 . . . . .	174
	将调制解调器添加到管理器 . . . . .	179
<b>章 15</b>	<b>CANopen 配置</b> . . . . .	<b>181</b>
	CANopen 接口配置 . . . . .	181
<b>章 16</b>	<b>后配置</b> . . . . .	<b>185</b>
	后配置表现 . . . . .	186
	后配置文件管理 . . . . .	187
	后配置示例 . . . . .	189
<b>章 17</b>	<b>将 Modicon M241 Logic Controller 连接到 PC</b> . . . . .	<b>191</b>
	将控制器连接到 PC . . . . .	191
<b>章 18</b>	<b>SD 卡</b> . . . . .	<b>195</b>
	更新 Modicon M241 Logic Controller 固件 . . . . .	196
	使用 SD 卡传输文件 . . . . .	198
<b>附录</b>	. . . . .	<b>205</b>
<b>附录 A</b>	<b>用于在用户程序中获取 / 设置串行线路配置的功能</b> . . . . .	<b>207</b>
	GetSerialConf: 获取串行线路配置 . . . . .	208
	SetSerialConf: 更改串行线路配置 . . . . .	209
	SERIAL_CONF: 串行线路配置数据类型的结构 . . . . .	211
<b>附录 B</b>	<b>控制器性能</b> . . . . .	<b>213</b>
	处理性能 . . . . .	213
<b>术语表</b>	. . . . .	<b>215</b>
<b>索引</b>	. . . . .	<b>223</b>





## 重要信息

### 声明

在尝试安装、操作或维护设备之前，请仔细阅读下述说明并通过查看来熟悉设备。下述特别信息可能会在本文其他地方或设备上出现，提示用户潜在的危险，或者提醒注意有关阐明或简化某一过程的信息。



在“危险”标签上添加此符号表示存在触电危险，如果不遵守使用说明，会导致人身伤害。



这是提醒注意安全的符号。提醒用户可能存在人身伤害的危险。请遵守所有带此符号的安全注意事项，以避免可能的人身伤害甚至死亡。

## 危险

“危险”表示极可能存在危险，如果不遵守说明，可导致严重的人身伤害甚至死亡。

## 警告

“警告”表示可能存在危险，如果不遵守说明，可导致严重的人身伤害甚至死亡，或设备损坏。

## 注意

“注意”表示可能存在危险，如果不遵守说明，可导致严重的人身伤害或设备损坏。

## 注意

“注意”用于表示与人身伤害无关的危害。

---

## 请注意

电气设备的安装、操作、维修和维护工作仅限于合格人员执行。Schneider Electric 不承担由于使用本资料所引起的任何后果。

专业人员是指掌握与电气设备的制造和操作及其安装相关的技能和知识的人员，他们经过安全培训能够发现和避免相关的危险。

# 关于本书



## 概览

### 文档范围

本文档旨在帮助您通过 SoMachine 软件对 Modicon M241 Logic Controller 进行编程和操作。

**注意：**在安装、操作或维护前，请阅读并了解本文档和所有（参见第 9 页）相关文档 Modicon M241 Logic Controller。

Modicon M241 Logic Controller 用户应当阅读整个文档，以了解所有功能。

### 有效性说明

本文档已随 SoMachine V4.1 的发布进行了更新。

### 相关的文件

文件名称	参考编号
SoMachine 编程指南	EIO0000000067（英语）； EIO0000000069（法语）； EIO0000000068（德语）； EIO0000000071（西班牙语）； EIO0000000070（意大利语）； EIO0000000072（简体中文）
Modicon M241 Logic Controller 硬件指南	EIO0000001456（英语）； EIO0000001457（法语）； EIO0000001458（德语）； EIO0000001459（西班牙语）； EIO0000001460（意大利语）； EIO0000001461（简体中文）
Modicon TM2 扩展模块配置编程指南	EIO0000000396（英语）； EIO0000000397（法语）； EIO0000000398（德语）； EIO0000000399（西班牙语）； EIO0000000400（意大利语）； EIO0000000401（简体中文）

文件名称	参考编号
Modicon TM3 扩展模块配置编程指南	EIO0000001402 (英语); EIO0000001403 (法语); EIO0000001404 (德语); EIO0000001405 (西班牙语); EIO0000001406 (意大利语); EIO0000001407 (简体中文)
Modicon TM4 扩展模块编程指南	EIO0000001802 (英语); EIO0000001803 (法语); EIO0000001804 (德语); EIO0000001805 (西班牙语); EIO0000001806 (意大利语); EIO0000001807 (简体中文)
Modicon TMC4 卡盒编程指南	EIO0000001790 (英语); EIO0000001791 (法语); EIO0000001792 (德语); EIO0000001793 (西班牙语); EIO0000001794 (意大利语); EIO0000001795 (简体中文)
Modicon M241 Logic Controller PLCSystem 库指南	EIO0000001438 (英语); EIO0000001439 (法语); EIO0000001440 (德语); EIO0000001441 (西班牙语); EIO0000001442 (意大利语); EIO0000001443 (简体中文)
Modicon M241 Logic Controller HSC 库指南	EIO0000001444 (英语); EIO0000001445 (法语); EIO0000001446 (德语); EIO0000001447 (西班牙语); EIO0000001448 (意大利语); EIO0000001449 (简体中文)
Modicon M241 Logic Controller PTO/PWM 库指南	EIO0000001450 (英语); EIO0000001451 (法语); EIO0000001452 (德语); EIO0000001453 (西班牙语); EIO0000001454 (意大利语); EIO0000001455 (简体中文)

您可以从我们的网站下载这些技术出版物和其它技术信息，网址是：[www.schneider-electric.com](http://www.schneider-electric.com)。

 **警告**

**失去控制**

- 任何控制方案的设计者都必须考虑到控制路径可能出现故障的情况，并为某些关键控制功能提供一种方法，使其在出现路径故障时，以及出现路径故障后恢复至安全状态。紧急停止和越程停止、断电和重启都属于关键控制功能。
- 对于关键控制功能，必须提供单独或冗余的控制路径。
- 系统控制路径可包括通讯链路。必须对暗含的无法预料的传输延迟或链接失效问题加以考虑。
- 遵守所有事故预防规定和当地的安全指南。<sup>1</sup>
- 为了保证正确运行，在投入使用前，必须对设备的每次执行情况分别进行全面测试。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

<sup>1</sup> 有关详细信息，请参阅 NEMA ICS 1.1（最新版）中的“安全指导原则 - 固态控制器的应用、安装和维护”以及 NEMA ICS 7.1（最新版）中的“结构安全标准及可调速驱动系统的选择、安装与操作指南”或您特定地区的类似规定。

 **警告**

**意外的设备操作**

- 仅使用 Schneider Electric 认可的可与本设备配合使用的软件。
- 每次更改物理硬件配置后，请更新应用程序。

**不遵循上述说明可能导致人员伤亡或设备损坏。**



---

# 章 1

## 关于 Modicon M241 Logic Controller

---

### M241 Logic Controller 描述

#### 概述

M241 Logic Controller 具有多种强大的功能，可在广泛的应用程序中使用。

软件配置、编程和试运行通过 SoMachine 软件完成，该软件在 SoMachine 编程指南和 M241 Logic Controller 编程指南 ( 参见第 页 ) 中进行了介绍。

#### 编程语言

M241 Logic Controller 可通过 SoMachine 软件对其进行配置和编程，该软件支持以下 IEC 61131-3 编程语言：

- IL: 指令列表
- ST: 结构化文本
- FBD: 功能块图
- SFC: 顺序功能图
- LD: 梯形图

SoMachine 软件也可用于使用 CFC ( 连续功能图 ) 语言对这些控制器进行编程。

#### 电源

M241 Logic Controller 的电源为 24 Vdc 或 100 到 240 Vac。

#### 实时时钟

M241 Logic Controller 包括一个实时时钟 (RTC) 系统。

#### 运行 / 停止

M241 Logic Controller 可以通过以下方式在外部进行操作：

- 硬件运行 / 停止开关
- 通过软件配置中定义的专用数字量输入进行运行 / 停止操作。有关详细信息，请参阅数字量输入的配置 ( 参见第 80 页 )。
- SoMachine 软件命令

## 存储器

下表描述了不同类型的存储器：

存储器类型	大小	作用
RAM	64 MB, 其中 8 MB 可供应用程序使用	执行应用程序。
闪存	128 MB	在断电时保存程序和数据。

## 嵌入式输入 / 输出

根据控制器参考号，提供以下嵌入式 I/O 类型：

- 常规输入
- 与计数器关联的快速输入
- 常规漏极 / 源极晶体管输出
- 与脉冲发生器关联的快速漏极 / 源极晶体管输出
- 继电器输出

## 可移动存储

M241 Logic Controller 包含嵌入式 SD 卡插槽。

SD 卡的主要用途是：

- 使用新应用程序初始化控制器
- 更新控制器固件
- 将后配置文件应用于控制器
- 应用配方
- 接收数据记录文件

## 嵌入式通讯功能

根据控制器参考号，以下类型的通讯端口可用：

- CANopen 主站
- 以太网
- USB Mini-B
- 串行线路 1
- 串行线路 2

## M241 Logic Controller

参考号	数字量输入	数字量输出	通讯端口	端子类型	电源
TM241C24R	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	6 路 2A 继电器输出 4 路源极快速输出 (脉冲发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端子块	100...240 Vac
TM241CE24R	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	6 路 2A 继电器输出 4 路源极快速输出 (脉冲发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端子块	100...240 Vac
TM241CEC24R	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	6 路 2A 继电器输出 4 路源极快速输出 (脉冲发生器) (3)	2 个串行线路端口 1 个以太网端口 1 个 CANopen 主站 端口 1 个 USB 编程端口	可插拔螺钉端子块	100...240 Vac
TM241C24T	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	源极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端子块	24 Vdc
TM241CE24T	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	源极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端子块	24 Vdc
TM241CEC24T	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	源极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口 1 个 CANopen 主站 端口	可插拔螺钉端子块	24 Vdc
TM241C24U	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	漏极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端子块	24 Vdc
TM241CE24U	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	漏极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端子块	24 Vdc

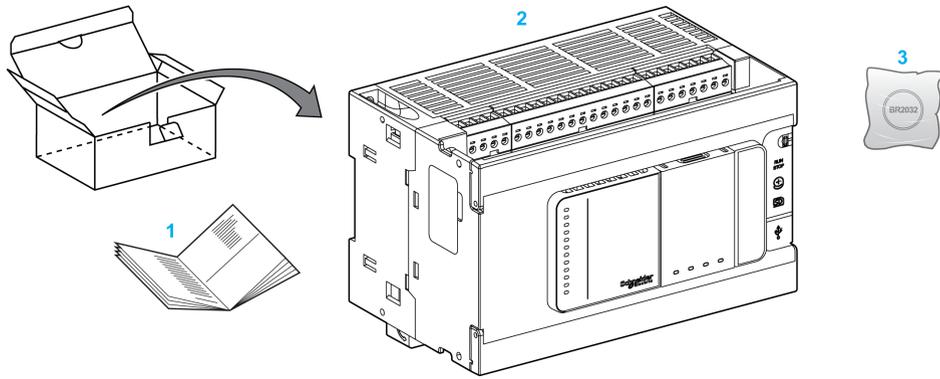
(1) 常规输入的最大频率为 5 kHz。  
(2) 快速输入可用作针对计数或事件功能的常规输入或快速输入。  
(3) 快速晶体管输出可用作常规晶体管输出，也可以用作脉冲发生器 (FG/PTO/PWM) 或计数 (HSC) 功能的快速输出。

参考号	数字量输入	数字量输出	通讯端口	端子类型	电源
TM241CEC24U	6 路常规输入 (1) 8 路快速输入 (计数器) (2)	漏极输出 6 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口 1 个 CANopen 主站 端口	可插拔螺钉端 子块	24 Vdc
TM241C40R	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	12 路 2A 继电器输出 4 路源极快速输出 (脉冲发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端 子块	100...240 Vac
TM241CE40R	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	12 路 2A 继电器输出 4 路源极快速输出 (脉冲发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端 子块	100...240 Vac
TM241C40T	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	源极输出 12 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端 子块	24 Vdc
TM241CE40T	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	源极输出 12 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端 子块	24 Vdc
TM241C40U	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	漏极输出 12 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口	可插拔螺钉端 子块	24 Vdc
TM241CE40U	16 路常规输入 (1) 8 路快速输入 (计数器) (2)	漏极输出 12 路常规晶体管输出 4 路快速输出 (脉冲 发生器) (3)	2 个串行线路端口 1 个 USB 编程端口 1 个以太网端口	可插拔螺钉端 子块	24 Vdc

(1) 常规输入的最大频率为 5 kHz。  
(2) 快速输入可用作针对计数或事件功能的常规输入或快速输入。  
(3) 快速晶体管输出可用作常规晶体管输出，也可以用作脉冲发生器 (FG/PTO/PWM) 或计数 (HSC) 功能的快速输出。

## 交付内容

下图显示了 M241 Logic Controller 的交付内容：



- 1 M241 Logic Controller 说明书
- 2 M241 Logic Controller
- 3 锂氟化碳电池，类型 Panasonic BR2032。



---

# 章 2

## 如何配置控制器

---

### 如何配置控制器

#### 简介

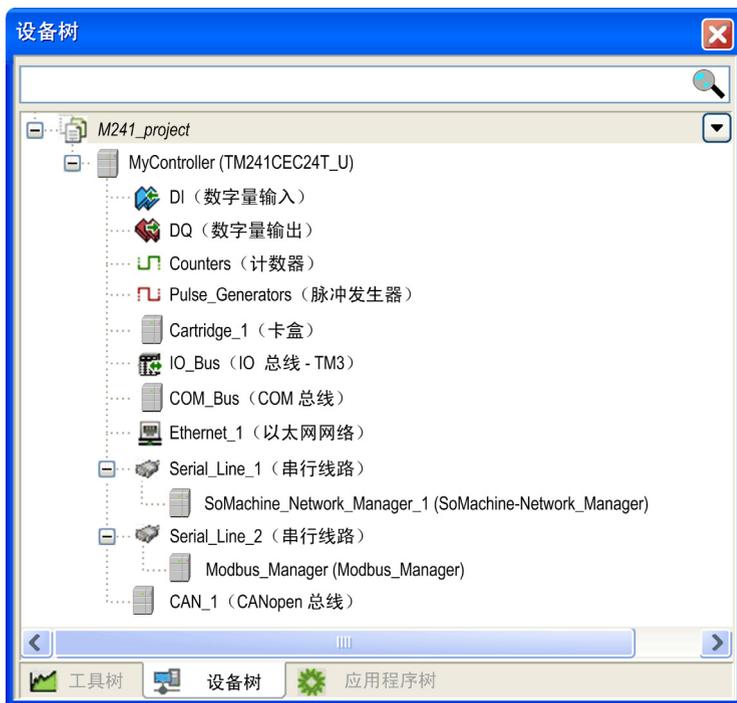
首先，在 SoMachine 软件中创建一个新项目或打开现有项目。

有关如何执行以下操作的信息，请参阅 *SoMachine 编程指南*：

- 将控制器添加到项目
- 将扩展模块添加到控制器
- 更换现有控制器
- 将控制器转换为不同的兼容设备

## 设备树

**设备树**显示当前硬件配置的结构化视图。当您添加控制器到项目时，会将许多节点自动添加到**设备树**中，具体取决于控制器提供的功能。



项	用于配置 ...
DI	Logic Controller 的嵌入式数字量输入
DQ	Logic Controller 的嵌入式数字量输出
计数器	嵌入式计数功能 (HSC)
Pulse_Generators	嵌入式脉冲发生器功能 (PTO/PWM/FG)
Cartridge_x	插入到 Logic Controller 的卡盒
IO_Bus	连接到 Logic Controller 的扩展模块
COM_Bus	Logic Controller 的通讯总线
Ethernet_x	嵌入式以太网、串行线路或 CANopen 通讯接口
Serial_Line_x	<b>注意：</b> 以太网和 CANopen 仅在某些参考号上可用。
CAN_x	

### 应用程序树

**应用程序树**可用于管理项目特定的应用程序以及全局应用程序、POU 和任务。

### 工具树

**工具树**可用于配置项目的 HMI 部分及对库进行管理。



# 章 3

## 库

### 库

#### 简介

库提供可用于开发您的项目的功能、功能块、数据类型和全局变量。

SoMachine 的**库管理器**提供项目所涉及的库的相关信息，并可帮助您安装新库。有关**库管理器**的详细信息，请参阅 SoMachine 编程指南。

#### Modicon M241 Logic Controller

为应用程序选择 Modicon M241 Logic Controller 后，SoMachine 会自动加载以下库：

库名称	描述
IoStandard	<b>CmpIoMgr</b> 配置类型、 <b>ConfigAccess</b> 、参数和帮助功能：管理应用程序中的 I/O。
Standard	包含为符合 IEC61131-3 所需的功能和功能块，作为 IEC 编程系统的标准 POU。将标准 POU 链接到项目 (standard.library)。
Util	模拟量监视器、BCD 转换、位 / 字节功能、控制器数据类型、功能操作器、数学函数、信号。
PLCCommunication ( 参见 SoMachine, Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide)	<b>SysMem, Standard</b> 。这些功能有助于实现特定设备之间的通讯。大多数功能专用于 Modbus 交换。相对于调用功能的应用任务而言，通讯功能的处理是异步的。
M241 PLCSystem ( 参见 Modicon M241 Logic Controller, 系统功能和变量, PLCSystem 库指南)	包含功能和变量，用于获取信息和向控制器系统发送命令。
M241 HSC ( 参见 Modicon M241 Logic Controller, 高速计数, HSC 库指南)	包含功能块和变量，用于获取信息和向 Modicon M241 Logic Controller 的快速输入/输出发送命令。这些功能块可以在 Modicon M241 Logic Controller 的快速输入 / 输出上执行 HSC（高速计数）。
M241 PTO/PWM ( 参见 Modicon M241 Logic Controller, PTO/PWM, 库指南)	包含功能块和变量，用于获取信息和向 Modicon M241 Logic Controller 的快速输入 / 输出发送命令。这些功能块可用于在 Modicon M241 Logic Controller 的快速输出上实现 PTO（脉冲串输出）和 PWM（脉冲宽度调制）功能。
重新定位表 ( 参见第 34 页)	可以将非连续数据重新分组到寄存器的连续表中，从而组织数据以优化 Modbus 客户端与控制器之间的交换。



---

## 章 4

### 支持的标准数据类型

---

#### 支持的标准数据类型

##### 支持的标准数据类型

控制器支持以下 IEC 数据类型：

数据类型	下限	上限	信息内容
BOOL	False	True	1 位
BYTE	0	255	8 位
WORD	0	65,535	16 位
DWORD	0	4,294,967,295	32 位
LWORD	0	$2^{64}-1$	64 位
SINT	-128	127	8 位
USINT	0	255	8 位
INT	-32,768	32,767	16 位
UINT	0	65,535	16 位
DINT	-2,147,483,648	2,147,483,647	32 位
UDINT	0	4,294,967,295	32 位
LINT	$-2^{63}$	$2^{63}-1$	64 位
ULINT	0	$2^{64}-1$	64 位
REAL	1.175494351e-38	3.402823466e+38	32 位
STRING	1 个字符	255 个字符	1 个字符 = 1 个字节
WSTRING	1 个字符	255 个字符	1 个字符 = 1 个字
TIME	-	-	32 位

有关 ARRAY、LTIME、DATE、TIME、DATE\_AND\_TIME 和 TIME\_OF\_DAY 的有关详细信息，，请参阅 SoMachine 编程指南。



---

# 章 5

## 存储器映射

---

### 简介

本章介绍 Modicon M241 Logic Controller 中不同存储器区域的存储器映射和大小。这些存储器区域用于存储用户程序逻辑、数据和编程库。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
控制器存储器结构	28
RAM 存储器结构	29
闪存结构	31
重新定位表	34

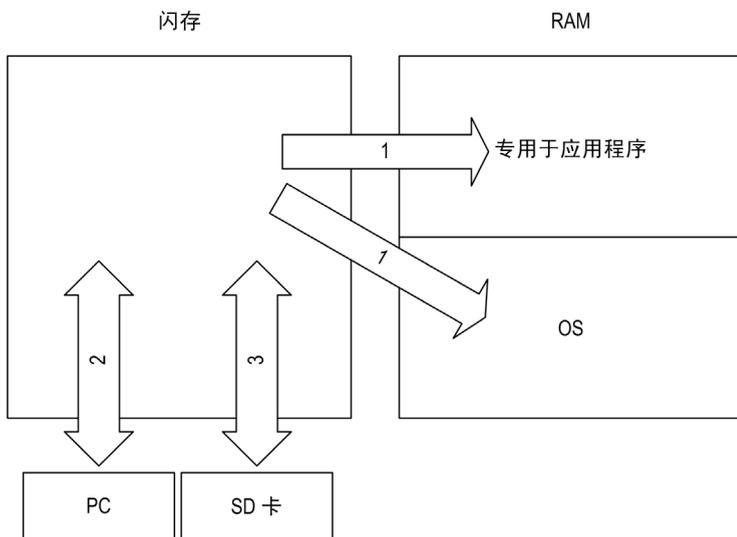
## 控制器存储器结构

### 简介

控制器存储器由 2 种类型的物理存储器组成：

- 闪存 (参见第 37 页) 中包含文件 (应用程序、配置文件等)。
- RAM (Random Access Memory) (参见第 29 页) 用于执行应用程序。

### 存储器中的文件传输



项	控制器状态	文件传输事件	连接	描述
1	-	在电源接通和重新启动时自动启动	内部	从闪存将文件传输到 RAM。RAM 的内容被覆盖。
2	除 INVALID_OS 之外的所有状态 <sup>(1)</sup>	由用户启动	以太网或 USB 编程端口	文件可通过以下途径传输： <ul style="list-style-type: none"> <li>● Web 服务器 (参见第 112 页)</li> <li>● FTP 服务器 (参见第 125 页)</li> <li>● SoMachine</li> </ul>
3	所有状态	连接 SD 卡时由脚本 (数据传输) 或电源重置 (克隆) 自动启动	SD 卡	使用 SD 卡上载 / 下载

**(1)** 如果控制器处于 INVALID\_OS 状态，则唯一可以访问的闪存为 SD 卡，且只能用于进行固件升级。

**注意：**无论控制器处于何种状态，都可以读取、写入或擦除闪存中的所有文件。修改闪存中的文件不会影响正在运行的应用程序。对闪存中文件的任何更改都将在下次重新启动时生效。

## RAM 存储器结构

### 简介

本节介绍 Modicon M241 Logic Controller 不同区域的 RAM (Random Access Memory) 大小。

### 存储器映射

RAM 大小为 64 MB。

RAM 由以下 2 个区域组成：

- 专用应用程序存储器
- 操作系统存储器

下表介绍专用应用程序存储器：

区域	元素	大小
系统区域 192 KB	系统区域可映射的地址 %MW0...%MW5999	125 KB
	系统和诊断变量 (%MW60000...%MW60199) 只能通过 Modbus 请求访问此存储器。 这些请求必须是只读请求。	
	动态存储器区域：读取重新定位表 (参见第 34 页) (%MW60200...%MW61999) 只能通过 Modbus 请求访问此存储器。 这些请求可以是读取或写入请求。但是，如果已经在重新定位表中声明此存储器，则必须是只读请求。	
	系统和诊断变量 (%MW62000...%MW62199) 只能通过 Modbus 请求访问此存储器。 这些请求可以是读取或写入请求。	
	动态存储器区域：写入重新定位表 (参见第 34 页) (%MW62200...%MW63999) 只能通过 Modbus 请求访问此存储器。 这些请求可以是读取或写入请求。但是，如果已经在重新定位表中声明此存储器，则必须使用只写请求。	
	保留	3 KB
	保留和持久性数据 (参见第 31 页)	64 KB
用户区域 8 MB	符号	动态分配
	变量	
	应用程序	
	库	

## 系统和诊断变量

变量	描述
PLC_R	控制器只读系统变量的结构。
PLC_W	控制器读 / 写系统变量的结构。
ETH_R	以太网只读系统变量的结构。
ETH_W	以太网读 / 写系统变量的结构。
PROFIBUS_R	PROFIBUS DP 只读系统变量的结构。
SERIAL_R	串行线路只读系统变量的结构。
SERIAL_W	串行线路读 / 写系统变量的结构。
TM3_MODULE_R	TM3 模块只读系统变量的结构。

有关系统和诊断变量的详细信息，请参阅 *M241 PLCSystem 库指南*。

## 存储器寻址

下表描述了地址大小 Double Word (%MD)、Word (%MW)、Byte (%MB) 和 Bit (%MX) 的存储器寻址：

双字	字	字节	位		
%MD0	%MW0	%MB0	%MX0.7	...	%MX0.0
		%MB1	%MX1.7	...	%MX1.0
	%MW1	%MB2	%MX2.7	...	%MX2.0
		%MB3	%MX3.7	...	%MX3.0
%MD1	%MW2	%MB4	%MX4.7	...	%MX4.0
		%MB5	%MX5.7	...	%MX5.0
	%MW3	%MB6	%MX6.7	...	%MX6.0
		%MB7	%MX7.7	...	%MX7.0
%MD2	%MW4	%MB8	%MX8.7	...	%MX8.0
		...	...	...	...
	...	...	...	...	...

存储器范围重叠的示例：

%MD0 包含 %MB0 (...) %MB3，%MW0 包含 %MB0 和 %MB1，%MW1 包含 %MB2 和 %MB3。

## 闪存结构

### 简介

闪存中包含控制器使用的文件系统。

### 文件类型

Modicon M241 Logic Controller 管理以下文件类型：

类型	描述
启动应用程序	此文件位于闪存中，包含可执行应用程序的已编译二进制代码。每次重新启动控制器时，都会从启动应用程序中提取可执行应用程序并将其复制到控制器 RAM <sup>(1)</sup> 。
应用程序源	源文件，如果不在 PC 上，则可从闪存上载至 PC <sup>(2)</sup> 。
后配置	包含以太网、串行线路和防火墙参数的文件。 每次重新启动时，该文件中指定的参数都会覆盖可执行应用程序中的参数。
数据记录	控制器按照用户应用程序的指定在其中记录事件的文件。
HTML 页面	Web 服务器显示的 HTML 页面，用于控制器中嵌入的网站。
操作系统 (OS)	可写入闪存的控制器固件。固件文件会在下次重新启动控制器时应用。
保留变量	剩余变量
保留持久性变量	
<p><b>(1)</b> 在 SoMachine 中，根据应用程序属性，启动应用程序的创建是可选项。缺省选项是在下载时创建启动应用程序。当您将应用程序从 SoMachine 下载到控制器时，只是将二进制可执行应用程序直接传输到 RAM。</p> <p><b>(2)</b> SoMachine 不支持将可执行应用程序或启动应用程序上载到 PC 进行修改。程序修改必须对应用程序源进行。下载应用程序后，可选择将源文件存储到闪存。</p>	

## 文件结构

下表显示了闪存的文件结构：

磁盘	目录	文件	内容	上载 / 下载的数据类型
/sys	OS	M241FW1v_XX.YY <sup>(1)</sup>	核心固件 1	固件
		M241FW2v_XX.YY <sup>(1)</sup>	核心固件 2	
		Version.ini	固件版本的控制文件	
	OS/FWM	xxxxx.bin	TM4 模块的固件	-
	Web	Index.htm	Web 服务器支持的 HTML 页面，用于控制器中嵌入的网站。	网站
		Conf.htm		-
		...		-
/usr	App	Application.app	引导应用程序	应用程序
		Application.crc		-
		Application.map		-
		Archive.prj <sup>(2)</sup>	应用程序源	-
	App/MFW	DeviceID_X.fw <sup>(2)</sup>	扩展模块固件	固件
	Cfg	Machine.cfg <sup>(2)</sup>	后配置文件 ( 参见第 185 页 )	配置
		CodesysLateConf.cfg <sup>(2)</sup>	<ul style="list-style-type: none"> <li>● 要启动的应用程序的名称</li> <li>● 路由表 ( 主 / 子网 )</li> </ul>	配置

(1): v\_XX.YY 表示版本  
(2): 如果有

磁盘	目录	文件	内容	上载 / 下载的数据类型
/usr	日志	UserDefinedLogName_1.log	使用数据记录功能 ( 参见 SoMachine, 数据记录功能, DataLogging 库指南 ) 创建的所有 *.log 文件。必须指定创建的文件总数, 以及每个日志文件的名称和内容。	日志文件
		...	–	–
		UserDefinedLogName_n.log	–	–
	Rcp		配方的主目录	–
	Syslog	Crash.log <sup>(2)</sup>	此文件包含检测到的系统错误的记录。供 Schneider Electric 技术支持使用。	日志文件
		System.log <sup>(2)</sup>	此文件包含通过查看控制器设备编辑器 ( 参见第 72 页 ) 的日志选项卡同时在联机的 SoMachine 中可见的系统事件数据。	–
		*.log	此文件包含固件系统事件的记录。供 Schneider Electric 技术支持使用。	–
	/data	–	–	保留和持久性数据
	/sd0	–	–	SD 卡。可插拔
		–	用户文件	–
(1): v_XX.YY 表示版本				
(2): 如果有				

**注意:** 使用 sysFile、sysDir 和 CAAFile 库访问 /sd0 和 /usr。有关这些库的功能块的有关详细信息, 请参阅 SoMachine 在线帮助中的“CoDeSys 库”主题。

### 备份数据记录文件

数据记录文件可能会变得很庞大, 达到超过文件系统中可用空间的程度。因此, 您应该采用某种方法将日志数据定期存档到 SD 卡上。您可以将日志数据拆分为多个文件, 如 LogMonth1, LogMonth2, 并使用 **ExecuteScript command** ( 参见 *Modicon M241 Logic Controller, 系统功能和变量, PLCSystem 库指南* ) 将第一个文件复制到 SD 卡。然后, 可以在第二个文件累积数据时将第一个文件从内部文件系统删除。如果您任由数据记录文件变大并超过文件大小限制, 您可能会丢失数据。

## 注意

### 丢失数据

定期将 \*.log 文件备份到 SD 卡, 以避免控制器文件系统中的可用空间饱和。

不遵循上述说明可能导致设备损坏。

## 重新定位表

### 简介

借助**重新定位表**，用户可以将非连续数据重新分组到定位寄存器的连续表中（可通过 Modbus 进行访问），从而组织数据以优化控制器与其他设备之间的通讯。

**注意：**重新定位表被视为一个对象。一个控制器只能添加一个重新定位表对象。

### 重新定位表描述

下表介绍**重新定位表**组织方式：

寄存器	描述
60200...61999	动态存储器区域：读取重新定位表
62200...63999	动态存储器区域：写入重新定位表

有关详细信息，请参阅 *M241 PLCSystem 库指南*。

### 添加重新定位表

下表描述了如何将**重新定位表**添加到项目中：

步骤	操作
1	在 <b>应用程序树</b> 选项卡中选择 <b>应用程序</b> 节点。
2	单击  。
3	单击 <b>添加其他对象 → 重新定位表 ...</b> 。 <b>结果：</b> 将显示 <b>添加重新定位表</b> 窗口。
4	单击 <b>添加</b> 。 <b>结果：</b> 创建并初始化新的重新定位表。 <b>注意：</b> 由于对控制器而言，重新定位表是唯一的，因此其名称为 <b>重新定位表</b> ，不能更改此名称。

## 重新定位表编辑器

借助重新定位表编辑器，可以对重新定位表中的变量进行组织。

要访问重新定位表编辑器，请双击工具树选项卡中的**重新定位表**节点：



下图描述了重新定位表编辑器：

Relocation Table [MyController\_1:PLC Logic: 应用程序]
◀ ▶ ✕

**读取:**

+
↓
↑
✕
📄
📁
✕

ID	变量	地址	长度	有效性
1	PLC_GVL.PLC_R.i_dwSerialNumber	%MW60200	2	True
2	PLC_GVL.PLC_R.i_sNodeName	%MW60202	16	True
3	PLC_GVL.PLC_R.i_sProductRef	%MW60218	16	True
4	GVL.DIG_IO_LOOPS_STS	%MW60234	1	True

**写入:**

+
↓
↑
✕
📄
📁
✕

ID	变量	地址	长度	有效性
1	PLC_GVL.PLC_W.q_wResetCounterEvent	%MW62200	1	True
2	PLC_GVL.ETH_W.q_wResetCounter	%MW62201	1	True
3	GVL.AckDigLoopFlt	%MW62202	1	True
4	GVL.TempLoop1SetPoint	%MW62203	2	True

图标	元素	描述
	新项目	向系统变量列表中添加元素。
	下移	将列表中选定的元素向下移动。
	上移	将列表中选定的元素向上移动。
	删除项目	删除列表中选定的元素。
	复制	复制列表中选定的元素。
	粘贴	粘贴复制的元素。
	擦除空项目	删除列表中“变量”列为空的所有元素。
-	ID	自动递增的整数（不可编辑）。
-	变量	变量的名称或完整路径（可编辑）。
-	地址	存储变量的系统区域地址（不可编辑）。
-	长度	变量长度（以字为单位）。
-	有效性	指明输入的变量是否有效（不可编辑）。

**注意：**如果在修改程序后未定义某个变量，则单元格的内容显示为红色，相关的**有效性**单元格为 False，**地址**设置为 -1。

---

# 章 6

## 任务

---

### 简介

应用程序树中的**任务配置**节点用于定义一个或多个任务，以控制应用程序的执行。

可用的任务类型有：

- 循环
- 自由运行
- 事件
- 外部事件

本章先介绍这些任务类型，然后提供有关最大任务数、缺省任务配置以及任务优先级的信息。此外，本章还介绍系统和任务看门狗功能，并说明这些功能与任务执行之间的关系。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
最大任务数	38
任务配置屏幕	39
任务类型	41
系统和任务看门狗	44
任务优先级	45
缺省任务配置	48

## 最大任务数

### 最大任务数

可为 Modicon M241 Logic Controller 定义的最大任务数为：

- 总任务数 = 20
- 循环任务数 = 4；未定义自由运行任务时。  
循环任务数 = 3；已定义自由运行任务时。
- 自由运行任务数 = 1
- 事件任务数 = 8
- 外部事件任务数 = 8

### 自由运行任务特别注意事项

自由运行任务 (参见第 42 页) 没有固定持续时间。在自由运行模式中，每次任务扫描都在上次扫描完成时以及系统处理一段时间后 (自由运行任务总持续时间的 30%) 开始。如果由于其他任务中断而使系统处理周期减少到不足 15% 且时间超过 3 秒，则会检测到系统错误。有关详细信息，请参阅系统看门狗 (参见第 44 页)。

**注意：**在有高优先级且耗时的任务运行时，不应在多任务应用程序中使用自由运行任务。这样做可能会触发任务看门狗超时。您不应将 CANopen 分配给自由运行任务，而应将 CANopen 分给给循环任务。

## 任务配置屏幕

### 屏幕描述

可通过以下屏幕配置任务。在**应用程序树**中双击要配置的任务可访问此屏幕。

每个配置任务都有自己的参数，这些参数与其他任务无关。

**配置**窗口由 4 个部分组成：

The screenshot shows a configuration window titled "MAST" with a close button. The window is divided into four main sections:

- 配置 (Configuration):** A tab at the top left.
- 优先级 (Priority):** A text field labeled "优先级 (0..31):" with the value "1".
- 类型 (Type):** A dropdown menu set to "循环" (Loop) and a text field labeled "间隔 (如 t#200ms):" with the value "t#20ms".
- 看门狗 (Watchdog):** A section with a checked "启用" (Enable) checkbox, a text field labeled "时间 (如 t#200ms):" with the value "100" and a unit dropdown set to "毫秒" (ms), and a text field labeled "灵敏度:" with the value "1".
- Call Management:** A toolbar with icons for "Add Call", "Remove Call", "Change Call", "上移" (Up), "下移" (Down), and "打开 POU" (Open POU).
- Table:** A table with two columns: "POU" and "注释" (Comment). The table is currently empty.

下表介绍配置屏幕的字段：

字段名称	定义
优先级	<p>使用 0 到 31 这些数字配置每个任务的优先级（0 表示最高优先级，31 表示最低优先级）。一次只能运行一个任务。优先级确定何时运行任务：</p> <ul style="list-style-type: none"> <li>● 优先级高的任务先于优先级低的任务执行</li> <li>● 具有相同优先级的任务将轮流运行（2 毫秒时间片）</li> </ul> <p><b>注意：</b>请勿分配具有相同优先级的任务。如果还存在其他任务试图先于具有相同优先级的任务执行，则结果可能不确定且不可预知。有关重要的安全信息，请参阅任务优先级（参见第 45 页）。</p>
类型	<p>下列任务类型可用：</p> <ul style="list-style-type: none"> <li>● 循环（参见第 41 页）</li> <li>● 事件（参见第 42 页）</li> <li>● 外部（参见第 43 页）</li> <li>● 自由运行（参见第 42 页）</li> </ul>
看门狗	<p>要配置看门狗（参见第 44 页），请定义以下 2 个参数：</p> <ul style="list-style-type: none"> <li>● <b>时间：</b>输入看门狗执行前的超时。</li> <li>● <b>灵敏度：</b>定义控制器停止程序执行并进入 HALT 状态前的看门狗定时器到期数。</li> </ul>
POU	<p>由任务控制的 POU（参见 <i>SoMachine, 编程指南</i>）（程序组织单位）列表在任务配置窗口中定义：</p> <ul style="list-style-type: none"> <li>● 要添加链接到任务的 POU，请使用命令 <b>Add Call</b> 并在 <b>输入助手</b> 编辑器中选择 POU。</li> <li>● 要从列表中删除 POU，请使用命令 <b>Remove Call</b>。</li> <li>● 要将列表的当前所选 POU 替换为其他项，请使用命令 <b>Change Call</b>。</li> <li>● 按列表中所显示的顺序执行 POU。要移动列表中的 POU，请选择 POU 然后使用命令 <b>上移</b> 或 <b>下移</b>。</li> </ul> <p><b>注意：</b>可创建所需数量的 POU。如果应用程序使用多个小型 POU（而不是一个大型 POU），则会延长在线模式下的变量刷新时间。</p>

## 任务类型

### 简介

以下部分介绍可用于您的程序的各种任务类型，并介绍了任务类型的特性。

### 循环任务

使用循环任务“配置”子选项卡的“类型”部分中的“间隔”设置向该任务分配固定循环时间。每个循环任务类型的执行方式如下：



- 1. 读取输入：**将物理输入状态写入到 %I 输入存储器变量，并执行其他系统操作。
- 2. 任务处理：**处理任务中定义的用户代码（POU 等）。在此操作期间，%Q 输出存储器变量会根据应用程序指令进行更新，但尚未写入到物理输出。
- 3. 写入输出：**%Q 输出存储器变量使用已定义的任何输出强制进行修改；但物理输出的写入取决于输出类型和所用指令。  
有关定义总线循环任务的有关详细信息，请参阅 SoMachine 编程指南和 Modicon M241 Logic Controller 设置（参见第 75 页）。  
有关 I/O 行为的有关详细信息，请参阅控制器状态的详细描述（参见第 55 页）。
- 4. 剩余间隔时间：**控制器固件执行系统处理和任何其他较低优先级的任务。

**注意：**如果为某个循环任务定义的周期过短，则该任务会在写入输出后立即重复，而不会执行其他较低优先级的任务或任何系统处理。这将会影响所有任务的执行并导致控制器超过系统看门狗限制，从而导致系统看门狗例外。

**注意：**将任务循环时间设为小于 3 毫秒的值时，在试运行期间，应首先通过“任务监控”屏幕监控实际的任务持续时间，以确保它一直低于配置的任务循环时间。如果超出配置的时间，则可能无法遵守任务循环，从而导致任务循环看门狗超时，控制器转换到 HALT 状态。为避免此条件到达特定的程度，将任务循环时间设为小于 3 毫秒的值时，对于任意给定的循环强加 +1 毫秒的实时限制，这样，计算的循环时间会稍微超出配置的循环时间。

**注意：**使用 **GetCurrentTaskCycle** 和 **SetCurrentTaskCycle** 功能通过应用程序获取和设置循环任务的间隔。（有关详细信息，请参阅 Toolbox Advance 库指南。）

## 自由运行任务

自由运行任务没有固定持续时间。在自由运行模式下，每个任务扫描都在前一个扫描完成时以及短时间系统处理后开始。每个自由运行任务类型的执行方式如下：



- 1. 读取输入：**将物理输入状态写入到 %I 输入存储器变量，并执行其他系统操作。
- 2. 任务处理：**处理任务中定义的用户代码（POU 等）。在此操作期间，%Q 输出存储器变量会根据应用程序指令进行更新，但尚未写入到物理输出。
- 3. 写入输出：**%Q 输出存储器变量使用已定义的任何输出强制进行修改；但物理输出的写入取决于输出类型和所用指令。  
有关定义总线循环任务的有关详细信息，，请参阅 SoMachine 编程指南和 Modicon M241 Logic Controller 设置 (参见第 75 页)。  
有关 I/O 行为的有关详细信息，，请参阅控制器状态的详细描述 (参见第 55 页)。
- 4. 系统处理：**控制器固件执行系统处理和任何其他较低优先级的任务（例如：HTTP 管理、以太网管理、参数管理）。

## 事件任务

此类型的任务由事件驱动，并由程序变量启动。除非有更高优先级的任务先于事件任务执行，否则事件任务在与触发事件关联的布尔变量的上升沿启动。在此情况下，事件任务会根据任务优先级分配的指示启动。

例如，如果您已经定义一个称为 my\_Var 的变量并且要将其分配给某个事件，请执行以下步骤：

步骤	操作
1	双击应用程序树中的任务。
2	从配置选项卡的类型列表中选择事件。
3	单击事件字段右侧的输入助手按钮  。 结果：显示输入助手窗口。
4	在输入助手对话框的树中导航，查找并分配 my_Var 变量。

**注意：**当以过高的频率触发事件任务时，控制器将进入 HALT 状态（异常）。事件的最大可接受速率是每毫秒 5 个事件。在这种情况下，将在应用程序日志页中记录“ISR Count Exceeded”（超过 ISR 计数）消息。

## 外部事件任务

此类型的任务由事件驱动，并通过检测到硬件或硬件相关的功能事件而启动。除非有更高优先级的任务先于外部事件任务执行，否则它会在事件发生时启动。在此情况下，外部事件任务会根据任务优先级分配的指示启动。

例如，外部事件任务可能与 HSC 停止事件关联。要将 **BLOCK0\_HSCSTOP** 事件与某个外部事件任务关联，请从配置选项卡的外部事件下拉列表中选择事件任务。

最多有 4 个类型的事件可与外部事件任务关联，具体取决于控制器：

- 高级输入的上升沿 (DI8...DI11)
- HSC 阈值
- HSC 停止
- CAN 同步

**注意：** CAN 同步是特定事件对象，取决于 **CANopen 管理器**配置。

## 系统和任务看门狗

### 简介

Modicon M241 Logic Controller 实现两种类型的看门狗功能：

- **系统看门狗：** 这些看门狗在控制器固件中定义并受其管理。用户无法配置这些看门狗。
- **任务看门狗：** 可为每个任务定义的可选看门狗。这些看门狗由您的应用程序进行管理并可在 SoMachine 中进行配置。

### 系统看门狗

为 Modicon M241 Logic Controller 定义了三个系统看门狗。它们由控制器固件进行管理，因此在 SoMachine 在线帮助中有时也称为硬件看门狗。当其中一个系统看门狗超过其阈值条件时，会检测到错误。

3 个系统看门狗的阈值条件定义如下：

- 如果所有任务需要 85% 以上的处理器资源，且时间超过 3 秒，将检测到系统错误。控制器进入 HALT 状态。
- 如果优先级在 0 到 24 之间的任务的总执行时间达到 100% 的处理器资源，且时间超过 1 秒，将检测到应用程序错误。控制器会以自动重新启动进入 EMPTY 状态进行响应。
- 如果在 10 秒的间隔过程中没有执行优先级最低的系统任务，将检测到系统错误。控制器会以自动重新启动进入 EMPTY 状态进行响应。

**注意：** 用户无法配置系统看门狗。

### 任务看门狗

SoMachine 允许您为应用程序中定义的每个任务配置可选任务看门狗。（在 SoMachine 在线帮助中，任务看门狗有时也称为软件看门狗或控制定时器）。当您定义的任务看门狗之一达到其阈值条件时，将检测到应用程序错误并且控制器会进入 HALT 状态。

在定义任务看门狗时，可使用以下选项：

- **时间：** 这定义允许的最长任务执行时间。当任务所用时间超过此值时，控制器会报告任务看门狗异常。
- **灵敏度：** “灵敏度” 字段用于定义在控制器检测到应用程序错误之前必须发生的任务看门狗异常次数。

要访问任务看门狗的配置，请双击[应用程序树](#)中的任务。

**注意：** 有关看门狗的详细信息，请参阅 SoMachine 编程指南。

## 任务优先级

### 任务优先级配置

可以将各个任务的优先级配置为 0 到 31（0 表示最高优先级，31 表示最低优先级）。每个任务的优先级必须唯一。如果向多个任务分配相同的优先级，则这些任务的执行会无法确定且不可预测，这可能会导致意外后果。



#### 意外的设备操作

请勿将相同的优先级分配给不同的任务。

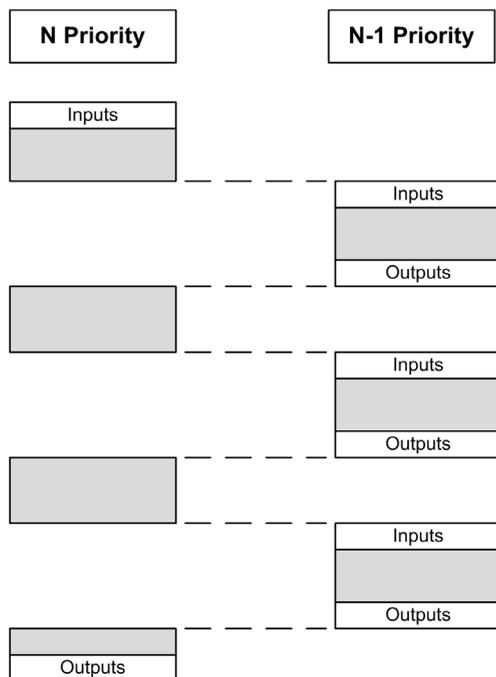
**不遵循上述说明可能导致人员伤亡或设备损坏。**

### 任务优先级建议

- 优先级 0 到 24：控制器任务。将这些优先级分配给具有高可用性要求的任务。
- 优先级 25 到 31：后台任务。将这些优先级分配给具有低可用性要求的任务。

## 嵌入式 I/O 的任务优先级

某个任务循环开始后，它可以中断任何优先级较低的任务（任务抢占）。优先级较高的任务循环完成后，被中断的任务将恢复。



**注意：**如果在不同任务中使用相同输入，则输入映像可能会在较低优先级任务的任务循环期间发生更改。

**注意：**为了提高在多任务执行期间正确输出行为的可能性，在不同任务中使用同一字节中的输出时，将显示一条警告消息。



**警告**

### 意外的设备操作

映射您的输入，从而使任务不会意外更改输入映像。

不遵循上述说明可能导致人员伤亡或设备损坏。

### TM2/TM3 模块和 CANopen I/O 的任务优先级

您可以选择驱动 TM3 和 CANopen 物理交换的任务。在 **PLC 设置**中，选择**总线循环任务**以定义用于交换的任务。缺省情况下，该任务设置为 **MAST**。如果此定义是在控制器级别，则可能会被 I/O 总线配置 (参见第 97 页) 所覆盖。在读取和写入阶段，同时刷新所有物理 I/O。在物理交换阶段，TM3/TM2 和 CANopen 数据被复制到一个虚拟 I/O 映像，如下图所示：



在任务循环开始时从 I/O 映像表读取输入。在任务结束时将输出写入 I/O 映像表。

**注意：**事件任务无法驱动 TM3/TM2 总线循环。

## 缺省任务配置

### 缺省任务配置

可以在“自由运行”或“循环”模式下配置 MAST 任务。缺省情况下，MAST 任务在循环模式下自动创建。其预设优先级为中 (15)，预设时间间隔为 20 毫秒，任务看门狗服务的激活时间为 100 毫秒，灵敏度为 1。有关优先级设置的详细信息，请参阅任务优先级 (参见第 45 页)。有关看门狗的详细信息，请参阅任务看门狗 (参见第 44 页)。

设计高效应用程序对于实现最多任务的系统至关重要。在此类应用程序中，可能难以使资源利用率始终低于系统看门狗阈值。如果只是重新分配优先级不足以保持低于阈值，则当 **SysTaskWaitSleep** 功能添加到一些较低优先级任务时，可以使这些任务使用较少的系统资源。有关此功能的详细信息，请参阅可选的系统 **SysTask** 库 /**SysLibs** 类别的库。

**注意：** 请勿删除或更改 MAST 任务的名称。否则，**SoMachine** 会在您尝试生成应用程序时检测到错误，因而您无法将其下载到控制器。

---

# 章 7

## 控制器状态和行为

---

### 简介

本章提供与控制器状态、状态转换和对系统事件的响应有关的信息。本章开头详细介绍了控制器状态图并描述了各种状态。接着定义了输出状态与控制器状态的关系，然后解释导致状态转换的命令和事件。结尾处介绍了与剩余变量有关的信息以及 **SoMachine** 任务编程选项对系统行为的影响。

### 本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
7.1	控制器状态图	50
7.2	控制器状态介绍	55
7.3	状态转换和系统事件	59

## 节 7.1

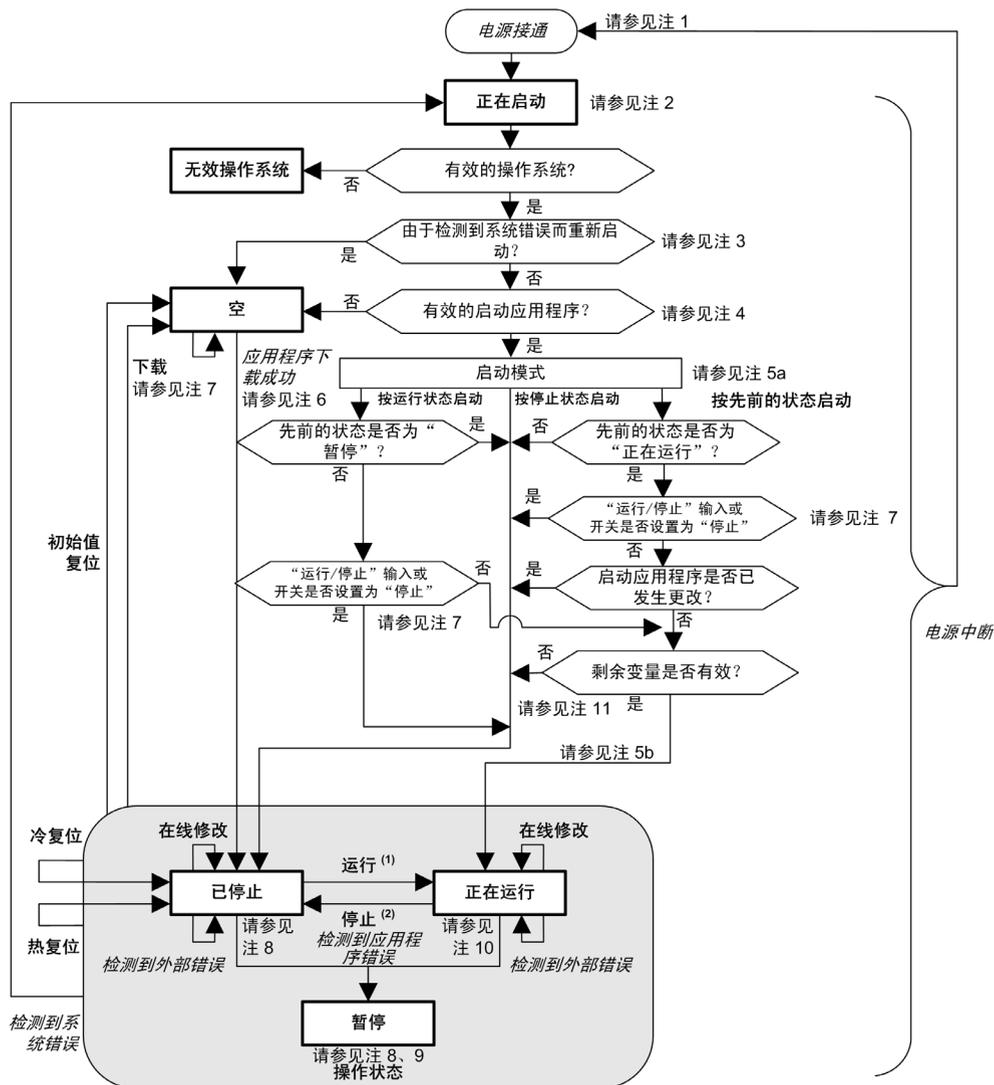
### 控制器状态图

---

## 控制器状态图

### 控制器状态图

下图描述了控制器的操作模式：



图例:

- 控制器状态以**粗体**表示
- 用户和应用程序命令以**粗体**表示
- 系统事件以*斜体*表示
- 决策、决策结果和一般信息用正常文本表示

(1) 有关 STOPPED 到 RUNNING 状态转换的详细信息, 请参阅运行命令 (参见第 63 页)。

(2) 有关 RUNNING 到 STOPPED 状态转换的详细信息, 请参阅停止命令 (参见第 63 页)。

### 注 1

电源重置 (先中断电源, 然后接通电源) 会删除所有输出强制设置。有关详细信息, 请参阅控制器状态和输出行为 (参见第 60 页)。

### 注 2

输出会采用其初始化状态。

### 注 3

在某些情况下, 当检测到系统错误时, 将导致控制器自动重新启动进入 **EMPTY** 状态, 如同闪存中不存在启动应用程序一样。但是, 并未从闪存中删除启动应用程序。在这种情况下, **ERR LED** (红色) 将有规律地闪烁。

### 注 4

验证有效的启动应用后, 会出现以下事件:

- 应用程序加载到 **RAM** 中。
- 应用后配置 (参见第 185 页) 文件设置 (如果有)。

在加载启动应用程序期间, 执行检查环境测试以确保剩余变量有效。如果检查环境测试无效, 则启动应用程序会加载, 但是控制器将进入 **STOPPED** 状态 (参见第 66 页)。

### 注 5a

**启动模式**在**控制器设备编辑器** (参见第 75 页) 的 **PLC 设置**选项卡中设置。

### 注 5b

如果发生电源中断, 控制器至少将继续保持 **RUNNING** 状态 4 毫秒后才会关闭。如果已经配置, 且从与控制器相同的电源向运行 / 停止输入供电, 则会立即检测到此输入掉电, 控制器的行为如同收到停止命令一样。因此, 当**启动模式**设置为**以上一个状态启动**时, 如果从同一个电源向控制器和运行 / 停止输入供电, 则在正常情况下, 控制器在电源中断后会重新启动进入 **STOPPED** 状态。

**注 6**

在成功的应用程序下载过程中，会发生以下事件：

- 应用程序直接加载到 RAM 中。
- 缺省情况下，创建引导应用程序，并将其保存到闪存中。

**注 7**

下载应用程序后的缺省行为是使控制器进入 STOPPED 状态，与运行 / 停止输入设置、运行 / 停止开关位置或下载前的上一个控制器状态无关。

然而，此时需要考虑两个方面：

**在线修改：** 假设运行/停止输入已配置并设置为“运行”或运行/停止开关设置为“运行”，那么在控制器处于 RUNNING 状态期间进行的在线修改（部分下载）会在成功后使控制器返回 RUNNING 状态。在使用**登录并看在线变化**选项之前，请在虚拟或非生产环境中测试对应用程序进行的修改，确认控制器和连接的设备是否具备 RUNNING 状态下的预期条件。

 **警告****意外的设备操作**

请务必先验证对“运行”状态下的应用程序进行的在线修改是否按预期方式运行，然后再将这些修改下载到控制器。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

**注意：** 对应用程序的在线修改不会自动写入启动应用程序，并会在下一次重新启动时由现有启动应用程序覆盖。如果您希望修改在重新启动之后仍然存在，请选择在线菜单中的**创建启动应用**来手动更新启动应用程序（控制器必须处于 STOPPED 状态才能实现此操作）。

**多重下载：** SoMachine 具有一项功能，使您可以将完整应用程序下载到网络或现场总线上的多个目标。选择**多重下载 ...** 命令时的缺省选项之一是**下载或在线修改之后启动全部应用**选项，假设下载目标各自的运行 / 停止输入发出进入 **RUNNING** 状态的命令，则该选项会以 **RUNNING** 状态重新启动所有这些目标，无论这些目标在启动多重下载之前的上一个控制器状态为何种状态。如果您不希望所有目标控制器都以 **RUNNING** 状态重新启动，请取消选择此选项。此外，在使用**多重下载**选项之前，请在虚拟或非生产环境中测试对应用程序进行的修改，确认目标控制器和连接的设备是否具备 **RUNNING** 状态下的预期条件。



## 警告

### 意外的设备操作

请务必先验证应用程序是否对于所有目标控制器和设备都按预期方式运行，然后再于选择了“**下装或在线修改之后启动全部应用**”选项的情况下发出“**多重下载 ..**”命令。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

**注意：** 与正常下载不同，在多重下载过程中， SoMachine 不提供用于创建启动应用程序的选项。您可以通过选择各个目标控制器上的**在线**菜单中的**创建启动应用**，随时手动创建启动应用程序（控制器必须处于 **STOPPED** 状态才能执行此操作）。

### 注 8

SoMachine 软件平台提供了许多功能强大的选项，用于在控制器处于 **STOPPED** 或 **HALT** 状态期间管理任务执行和输出条件。有关详细信息，请参阅控制器状态描述（参见第 55 页）。

### 注 9

要退出 **HALT** 状态，需要发出一个复位命令（热复位、冷复位、初始值复位）、下载应用程序或重置电源。

如果发生不可恢复事件（硬件看门狗或内部错误），则会强制进行电源重置。

### 注 10

**RUNNING** 状态有 2 种异常情况：

- **RUNNING** 但检测到外部错误：这种异常情况由 I/O LED 指示，此时显示为红色常亮。可以通过清除外部错误退出此状态（可能需要更改应用程序配置）。不需要任何控制器命令。
- **带断点 RUNNING**：这种异常情况由 I/O LED 指示，此时显示为一次闪烁。有关详细信息，请参阅控制器状态描述（参见第 55 页）。

### 注 11

启动应用程序可以不同于加载的应用程序。当通过 **SD** 卡、**FTP** 或文件传输下载启动应用程序时，或者当不创建启动应用程序而执行在线修改时，可能会发生这种情况。

## 节 7.2

### 控制器状态介绍

#### 控制器状态介绍

##### 简介

本节详细介绍控制器状态。

### 警告

#### 意外的设备操作

- 在没有通过命令进行状态修改、配置控制器选项、上载程序或修改控制器及其所连接设备的物理配置之前，切勿认为控制器已经处于某种控制器状态下了。
- 在执行以上任何操作之前，请考虑对所有所连接设备的影响。
- 在对某个控制器进行操作之前，请务必主动确认控制器的状态，方法是查看其 LED、确认运行 / 停止输入的情况、验证是否存在输出强制，以及通过 SoMachine 查看控制器状态信息。<sup>(1)</sup>

不遵循上述说明可能导致人员伤亡或设备损坏。

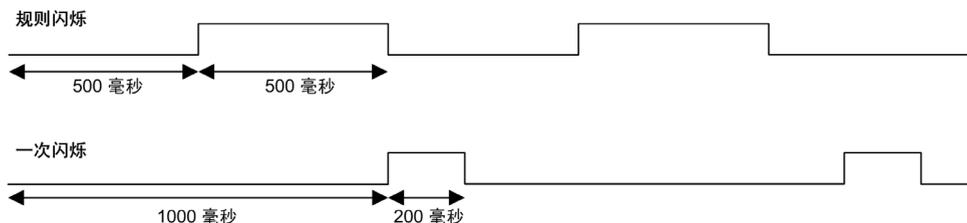
<sup>(1)</sup> 可以在 M241 PLCSystem library ( 参见 *Modicon M241 Logic Controller, 系统功能和变量, PLCSystem 库指南*) (M241 PLCSystem 库) 的 PLC\_R.i\_wStatus 系统变量中读取控制器状态。

## 控制器状态表

下表描述了控制器的状态：

控制器状态	描述	LED		
		RUN (绿色)	ERR (红色)	I/O (红色)
BOOTING (正在启动)	控制器可执行引导固件及其自身的内部自检。随后它将检查固件和应用程序的校验和。	亮起	灭	灭
		灭	亮起	亮起
		灭	亮起	灭
INVALID_OS	闪存中不存在有效固件文件。控制器不执行应用程序。请参阅固件升级部分以恢复正确的状态。	灭	规律闪烁	灭
EMPTY (空)	控制器无应用程序。	灭	一次闪烁	灭
在检测到系统错误后状态为 EMPTY	此状态与正常 EMPTY 状态相同。但是应用程序存在，并且是有意未加载。下一次重新启动（电源重置）后，或者下载新应用程序后，便会恢复正确状态。	灭	快速闪烁	灭
RUNNING (正在运行)	控制器正在执行有效应用程序。	亮起	灭	灭
RUNNING with breakpoint (断点运行)	此状态与 RUNNING 状态相同，只不过存在以下例外情况： <ul style="list-style-type: none"> <li>● 程序的任务处理部分在清除断点之前不会恢复。</li> <li>● LED 指示不同。</li> <li>● 有关断点管理的更多信息，请参阅 SoMachine 编程指南。</li> </ul>	一次闪烁	灭	灭
RUNNING (正在运行)，并检测到外部错误	配置、TM3、SD 卡或检测到的其他 I/O 错误。当 I/O LED 亮起时，可以在 PLC_R.i_lwSystemFault_1 和 PLC_R.i_lwSystemFault_2 中找到有关检测到的错误的详细信息。这些变量报告的所有检测到的错误情况都将导致 I/O LED 亮起。	亮起	灭	亮起
STOPPED (已停止)	控制器中的一个有效应用程序停止。有关此状态下输出和现场总线的行为的说明，请参见 STOPPED state (参见第 57 页) 的详细信息。	规律闪烁	灭	灭
STOPPED (已停止)，并检测到外部错误	配置、TM3、SD 卡或检测到的其他 I/O 错误。	规律闪烁	灭	亮起
HALT (暂停)	控制器停止执行应用程序，因为它检测到应用程序错误	规律闪烁	亮	—
引导应用程序未保存	控制器的存储器中的应用程序与闪存中的应用程序有所不同。在下次电源复位时，闪存中的应用程序将更改应用程序。	点亮或规律闪烁	一次闪烁	灭

此图显示规律闪烁与闪烁一次之间的区别：



## STOPPED 状态的详细信息

以下说明适用于 STOPPED 状态：

- 配置为运行 / 停止输入的输入保持正常运行。
- 配置为警报输出的输出保持正常运行，其值转为 0。
- 以太网、串行（Modbus、ASCII 等）和 USB 通讯服务保持正常运行，由这些服务写入的命令可以继续影响应用程序、控制器状态和存储器变量。
- 所有输出最初采用其配置的缺省状态（**保持当前值**或**将所有输出设置为缺省值**）或输出强制（如果使用）指示的状态。对于 PTO 功能使用的输出，将忽略缺省值，以便不生成额外脉冲。输出的后续状态取决于**停止时更新 IO**设置的值以及从远程设备收到的命令。

### 选择了“停止时更新 IO”时的任务和 I/O 行为

如果选择了**停止时更新 IO**设置：

- 读取输入操作继续正常执行。会读取物理输入，然后将其写入到 %I 输入存储器变量。
- 不执行任务处理操作。
- 写入输出操作继续执行。%Q 输出存储器变量会进行更新以反映**保持当前值**配置或**设置所有输出为默认值**配置，接着针对任何输出强制进行调整，然后写入到物理输出。

**注意：**专用功能停止运行。例如，计数器将停止。

**注意：**- 如果选择了**保持当前值**配置：

**注意：**PTO、PWM、频率发生器输出以及 HSC 反射输出都将设置为 0。

**注意：**- 如果选择了**将所有输出设置为缺省值**配置：

**注意：**PTO 输出设置为 0。

**注意：**PWM、频率发生器输出以及 HSC 反射输出均设置为配置的缺省值。

### 选择了“停止时更新 IO”时的 CAN 行为

以下情况适用于选中**停止时更新 IO** 设置时的 CAN 总线：

- CAN 总线保持全面正常运行。CAN 总线上的设备继续监视是否存在可用的 CAN 主站。
- TPDO 和 RPDO 继续进行交换。
- 可选的 SDO（如果已配置）继续进行交换。
- 心跳和节点防护功能（如果已配置）继续运行。
- 如果**停止时的输出动作**字段设置为**保持当前值**，则 TPDO 继续发送上一个实际值。
- 如果**停止时的输出动作**字段为**设置所有输出为默认值**，则上一个实际值会更新为默认值，后续 TPDO 会发送这些默认值。

### 未选择“停止时更新 IO”时的任务和 I/O 行为

如果未选择**停止时更新 IO** 设置，控制器会将 I/O 设置为**保持当前值**或**设置所有输出为默认值**条件（根据使用的输出强制进行调整）。在此之后，会出现以下情况：

- 读取输入操作停止。%I 输入存储器变量冻结为其上一个值。
- 不执行任务处理操作。
- 写入输出操作停止。%Q 输出存储器变量可以通过以太网、串行和 USB 连接进行更新。然而，物理输出不受影响，保持配置选项指定的状态。

**注意：**专用功能停止运行。例如，计数器将停止。

**注意：**- 如果选择了**保持当前值**配置：

**注意：**PTO、PWM、频率发生器输出以及 HSC 反射输出都将设置为 0。

**注意：**- 如果选择了**将所有输出设置为缺省值**配置：

**注意：**PTO 输出设置为 0。

**注意：**PWM、频率发生器输出以及 HSC 反射输出均设置为配置的缺省值。

### 未选择“停止时更新 IO”时的 CAN 行为

以下情况适用于未选中**停止时更新 IO** 设置时的 CAN 总线：

- CAN 主站停止通讯。CAN 总线上的设备采用其配置的故障预置状态。
- TPDO 和 RPDO 交换停止。
- 可选的 SDO（如果已配置）交换停止。
- 心跳和节点防护功能（如果已配置）停止。
- 在停止 CAN 主站之前，根据情况将当前或默认值写入 TPDO 并发送一次。

## 节 7.3

### 状态转换和系统事件

#### 概述

本节开头解释了控制器可能存在的输出状态。然后介绍了用于在控制器状态之间进行转换的系统命令，以及也可以影响这些状态的系统事件。最后解释了剩余变量，以及在状态转换过程中保留不同变量和数据类型的情况。

#### 本节包含了哪些内容？

本节包含了以下主题：

主题	页
控制器状态和输出行为	60
通过命令进行状态转换	63
错误检测、类型和管理	69
剩余变量	70

## 控制器状态和输出行为

### 简介

Modicon M241 Logic Controller 将输出行为定义为已更加灵活方式响应命令和系统事件。在讨论影响控制器状态的命令和事件之前，有必要先了解此行为。例如，典型控制器仅为停止状态下的输出行为定义 2 个选项：故障预置为缺省值或保持当前值。

可能应用这两个选项的输出行为和控制器状态包括：

- 由**应用程序**管理
- 保持**当前值**
- 设置所有**输出为缺省值**
- 硬件**初始化值**
- 软件**初始化值**
- **输出强制**

### 由应用程序管理

输出一般情况下由应用程序进行管理。这适用于 **RUNNING** 状态和 **STOPPED** 但检测到外部错误状态。

### 保持当前值

通过在**控制器编辑器**的 **PLC 设置**子选项卡的处于“**停止**”状态的输出的行为下拉菜单中选择**保持当前值**，来选择此选项。要访问控制器编辑器，请在设备树中右键单击控制器，然后选择**编辑对象**。

此输出行为适用于 **STOPPED** 控制器状态。还适用于处于 **HALT** 控制器状态的 **CAN** 总线。输出被设置并保持为其当前状态，尽管输出行为的详细信息会因为处于“**停止**”状态时更新 I/O 选项的设置以及通过已配置现场总线命令的操作而发生很大变化。有关这些变化的详细信息，请参阅控制器状态描述（参见第 55 页）。

### 将所有输出设置为缺省值

通过在**控制器编辑器**的 **PLC 设置**子选项卡的处于“**停止**”状态的输出的行为下拉菜单中选择**将所有输出设置为缺省值**，来选择此选项。要访问**控制器编辑器**，请在设备树中右键单击控制器，然后选择**编辑对象**。

当应用程序从 **RUN** 状态进入 **STOPPED** 状态时，或者如果应用程序从 **RUN** 状态进入 **HALT** 状态，会采用此输出行为。还适用于处于 **HALT** 控制器状态的 **CAN** 总线。输出被设置并保持为其当前状态，尽管输出行为的详细信息会因为处于“**停止**”状态时更新 I/O 选项的设置以及通过已配置现场总线的命令操作而发生很大变化。有关这些变化的详细信息，请参阅控制器状态描述（参见第 55 页）。

PTO 专用功能驱动的输出将不应用缺省值。

## 硬件初始化值

此输出状态适用于 **BOOTING**、**EMPTY**（没有启动应用程序时的电源重置之后或检测到系统错误之后）和 **INVALID\_OS** 状态。

在初始化状态下，模拟量、晶体管和继电器输出采用以下值：

- 对于模拟量输出：**Z**（高阻抗）
- 对于快速晶体管输出：**Z**（高阻抗）
- 对于常规晶体管输出：**0 Vdc**
- 对于继电器输出：**打开**

## 软件初始化值

当下载时或复位应用程序时，此输出状态适用。它适用于下载结束或者热复位或冷复位结束时。

软件**初始化值**为输出映像（%I、%Q 或在 %I 或 %Q 上映射的变量）的初始化值。

缺省情况下，它们设置为 **0**，但是可以映射 **GVL** 中的 I/O 并向输出分配一个不同于 **0** 的值。

## 输出强制

控制器允许您将所选输出的状态强制为定义值，以便于系统测试、试运行和维护。

仅当控制器连接到 **SoMachine** 时，才能强制输出的值。

为此，请使用**调试**菜单中的**强制值**命令。

输出强制将覆盖对输出执行的所有其他命令，无论正在执行何种任务编程。

如果在定义了输出强制后退出 **SoMachine**，将向您显示保留输出强制设置的选项。选择此选项后，输出强制会继续控制选定输出的状态，直至您下载应用程序或使用某个复位命令。

如果您的控制器支持，当选中**处于“停止”状态时更新 I/O** 选项时（缺省状态），即使 **Logic Controller** 处于 **STOP** 状态，强制的输出也会保持强制值。

### 输出强制的注意事项

您希望强制的输出必须包含在当前正由控制器执行的任务中。在未执行的任务或者自身执行因优先级或事件而延迟的任务中强制输出将不会对输出产生任何作用。但是，执行已延迟的任务后，强制届时将立即生效。

根据任务执行情况，强制可能会以不明显的方式影响您的应用程序。例如，某一事件任务可能会打开一个输出。之后，您可以尝试关闭该输出，但不要同时触发该事件。这样便会明显忽略强制效果。而且，在这之后，该事件可能会在强制生效之时触发该任务。

无法强制 PTO、PWM 和 HSC 专用功能驱动的输出。



#### 意外的设备操作

- 您必须全面了解强制会对与执行中的任务相关的输出产生怎样的影响。
- 请勿尝试强制包含在您不确定是否会及时执行的任务中的 I/O，除非您打算让强制在下次执行该任务时生效（无论何时）。
- 如果您强制某个输出，但是对物理输出没有产生明显效果，请勿在没有撤销强制的情况下退出 SoMachine。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

## 通过命令进行状态转换

### 运行命令

作用: 命令转换到 RUNNING 控制器状态。

启动条件: BOOTING or STOPPED 状态。

发出运行命令的方法:

- “运行/停止”输入: 如果已配置, 向“运行/停止”输入发出上升沿命令(假设“运行/停止”开关位于“运行”位置)。将“运行/停止”设置为 1, 以使所有后续选项生效。  
有关详细信息, 请参阅运行/停止输入。
- SoMachine 在线菜单: 选择**启动**命令。
- 来自 Web 服务器的“运行”命令
- 使用 M241 PLCSystem 库的 PLC\_W.q\_wPLCControl and PLC\_W.q\_uiOpenPLCControl 系统变量, 通过 Modbus 请求进行外部调用。
- **登录并看在线变化**选项: 在控制器处于 RUNNING 状态期间启动的在线修改(部分下载)会在成功后使控制器返回 RUNNING 状态。
- **多重下载**命令: 如果选择了**下载或在线修改之后启动全部应用**选项, 则将控制器设置为 RUNNING 状态, 无论目标控制器最初是处于 RUNNING、STOPPED、HALT 还是 EMPTY 状态。
- 控制器会在某些条件下自动重新启动为 RUNNING 状态。

有关更多详细信息, 请参阅控制器状态图。

### 停止命令

作用: 命令转换到 STOPPED 控制器状态。

启动条件: BOOTING、EMPTY 或 RUNNING 状态。

发出停止命令的方法:

- 运行/停止输入: 如果已配置, 命令运行/停止输入的值为 0。有关详细信息, 请参阅运行/停止输入。
- SoMachine 在线菜单: 选择**停止**命令。
- 来自 Web 服务器的“停止”命令
- 通过应用程序进行内部调用, 或使用 M241 PLCSystem 库的 PLC\_W.q\_wPLCControl 和 PLC\_W.q\_uiOpenPLCControl 系统变量, 通过 Modbus 请求进行外部调用。
- **登录并看在线变化**选项: 在控制器处于 STOPPED 状态期间启动的在线修改(部分下载)会在成功后使控制器返回 STOPPED 状态。
- **下载**命令: 将控制器隐式设置为 STOPPED 状态。
- **多重下载**命令: 如果未选择**下载或在线修改之后启动全部应用**选项, 则将控制器设置为 STOPPED 状态, 无论目标控制器最初是处于 RUNNING、STOPPED、HALT 还是 EMPTY 状态。

- 借助脚本重新启动: SD 卡上的文件传输脚本可以发出“重新启动”作为其最终命令。控制器会重新启动为 STOPPED 状态,前提是启动序列的其他条件允许发生这种情况。有关详细信息,请参阅重新启动(参见第 66 页)。
- 控制器会在某些条件下自动重新启动为 STOPPED 状态。

有关更多详细信息,请参阅控制器状态图。

## 热复位

作用: 将所有变量(除了剩余变量)复位为其默认值。将控制器置于 STOPPED 状态。

启动条件: RUNNING、STOPPED 或 HALT 状态。

发出热复位命令的方法:

- SoMachine 在线菜单: 选择**热复位**命令。
- 通过应用程序进行内部调用,或使用 M241 PLCSystem 库的 PLC\_W.q\_wPLCControl 和 PLC\_W.q\_uiOpenPLCControl 系统变量,通过 Modbus 请求进行外部调用。

热复位命令的效果作用:

1. 应用程序停止。
2. 擦除强制。
3. 复位针对检测到的错误的诊断指示。
4. 保持保留变量的值。
5. 保持保留 - 持久性变量的值。
6. 所有非定位和非剩余变量都复位为其初始化值。
7. 保持前 1000 个 %MW 寄存器的值。
8. %MW1000 至 %MW59999 寄存器的值复位为 0。
9. 所有现场总线通讯都停止,然后在完成复位后重新启动。
10. 所有 I/O 都暂时复位为其初始化值,然后复位为用户配置的默认值。
11. 在以下情况中,将读取后配置(参见第 185 页)文件:

有关变量的详细信息,请参阅剩余变量(参见第 70 页)。

## 冷复位

**作用:** 将所有变量（除了保留 - 持久性类型的剩余变量）都复位为其初始化值。将控制器置于 STOPPED 状态。

**启动条件:** RUNNING、STOPPED 或 HALT 状态。

**发出冷复位命令的方法:**

- SoMachine 在线菜单: 选择**冷复位**命令。
- 通过应用程序进行内部调用, 或使用 M241 PLCSystem 库的 PLC\_W.q\_wPLCCControl 和 PLC\_W.q\_uiOpenPLCCControl 系统变量, 通过 Modbus 请求进行外部调用。

**冷复位命令的作用:**

1. 应用程序停止。
2. 擦除强制。
3. 复位针对检测到的错误的诊断指示。
4. 保留变量的值复位为其初始化值。
5. 保持保留 - 持久性变量的值。
6. 所有非定位和非剩余变量都复位为其初始化值。
7. 保持前 1000 个 %MW 寄存器的值。
8. %MW1000 至 %MW59999 寄存器的值复位为 0。
9. 所有现场总线通讯都停止, 然后在完成复位后重新启动。
10. 所有 I/O 都暂时复位为其初始化值, 然后复位为用户配置的默认值。
11. 在以下情况中, 将读取 (参见第 185 页) 后配置文件:  
有关变量的详细信息, 请参阅剩余变量 (参见第 70 页)。

## 初始值复位

**作用:** 将所有变量（包括剩余变量）都复位为其初始化值。擦除控制器上的所有用户文件。将控制器置于 EMPTY 状态。

**启动条件:** RUNNING、STOPPED 或 HALT 状态。

**发出初始值复位命令的方法:**

- SoMachine 在线菜单: 选择**初始值复位**命令。

**初始值复位命令的作用:**

1. 应用程序停止。
2. 擦除强制。
3. 擦除所有用户文件（启动应用、数据记录、后配置）。
4. 复位针对检测到的错误的诊断指示。
5. 复位保留变量的值。
6. 复位保留 - 持久性变量的值。
7. 复位所有非定位和非剩余变量。
8. 前 1000 个 %MW 寄存器的值复位为 0。
9. %MW1000 至 %MW59999 寄存器的值复位为 0。
10. 所有现场总线通讯都停止。

11. 将嵌入式专用 I/O 复位为用户以前配置的缺省值。

12. 所有其他 I/O 都复位为其初始化值。

有关变量的详细信息，请参阅剩余变量（参见第 70 页）。

## 重新启动

作用：命令控制器重新启动。

启动条件：任何状态。

发出重新启动命令的方法：

- 电源重置
- 借助脚本重新启动

重新启动的作用：

1. 控制器的状态取决于多种条件：

a. 在以下条件下，控制器将处于 **RUNNING** 状态：

通过电源重置或借助脚本重新启动引发重新启动，并且：

- 启动模式设置为在“运行”状态下启动，未配置“运行/停止”输入，控制器在电源重置前未处于 HALT 状态，并且剩余变量有效。
- 启动模式设置为在“运行”状态下启动，已配置“运行/停止”输入并将其设置为“运行”，控制器在电源重置前未处于 HALT 状态，并且剩余变量有效。
- 启动模式设置为以上一个状态启动，控制器在电源重置前的状态为 **RUNNING**，“运行/停止”输入设置为未配置，启动应用程序未发生更改，并且剩余变量有效。
- 启动模式设置为以上一个状态启动，控制器在电源重置前的状态为 **RUNNING**，并且“运行/停止”输入已配置并设置为“运行”。

b. 在以下条件下，控制器将处于 **STOPPED** 状态：

通过电源重置或借助脚本重新启动引发重新启动，并且：

- 启动模式设置为在“停止”状态下启动。
- 启动模式设置为以上一个状态启动，并且控制器在电源重置前的状态不是 **RUNNING**。
- 启动模式设置为以上一个状态启动，控制器在电源重置前的状态为 **RUNNING**，“运行/停止”输入设置为未配置，并且启动应用程序已发生更改。
- 启动模式设置为以上一个状态启动，控制器在电源重置前的状态为 **RUNNING**，“运行/停止”输入设置为未配置，启动应用程序未发生更改，并且剩余变量无效。
- 启动模式设置为以上一个状态启动，控制器在电源重置前的状态为 **RUNNING**，并且“运行/停止”输入已配置并设置为“停止”。
- 启动模式设置为在“运行”状态下启动，并且控制器在电源重置前的状态是 **HALT**。
- 启动模式设置为在“运行”状态下启动，控制器在电源重置前的状态不是 **HALT**，并且“运行/停止”输入已配置并设置为“停止”。

c. 在以下条件下，控制器将处于 **EMPTY** 状态：

- 不存在启动应用程序或启动应用程序无效，或
- 由检测到的特定系统错误引发重新启动。

d. 如果不存在有效固件，则控制器状态将为 **INVALID\_OS**。

2. 如果成功加载启动应用，则会保持强制。如果未成功加载，则会擦除强制。
3. 复位针对检测到的错误的诊断指示。
4. 如果保存的环境有效，则恢复保留变量的值。
5. 如果保存的环境有效，则恢复保留 - 持久性变量的值。
6. 所有非定位和非剩余变量都复位为其初始化值。
7. 如果保存的环境有效，则恢复前 1000 个 %MW 寄存器的值。
8. %MW1000 至 %MW59999 寄存器的值复位为 0。
9. 所有现场总线通讯都会停止，然后在成功加载了启动应用之后重新启动。
10. 如果控制器在重新启动之后采用 STOPPED 状态，则所有 I/O 都会复位其初始化值，然后复位为用户配置的缺省值。
11. 在以下情况中，将读取 ( 参见第 185 页 ) 后配置文件：

有关变量的详细信息，请参阅剩余变量 ( 参见第 70 页 )。

**注意：**如果应用程序和剩余变量与启动应用中的定义相同，则检查环境测试会将环境视作有效。

**注意：**如果从与控制器相同的电源向运行 / 停止输入供电，则会立即检测到此输入掉电，控制器的行为如同收到停止命令一样。因此，当启动模式设置为以上一个状态启动时，如果从同一个电源向控制器和“运行 / 停止”输入供电，则在正常情况下，控制器在电源中断后会重新启动进入 STOPPED 状态。

**注意：**如果您在控制器处于 RUNNING 或 STOPPED 状态期间对应用程序进行在线修改，但是未手动更新您的启动应用程序，则控制器会在下一次重新启动时检测到环境中存在差异，剩余变量会按照冷复位命令进行复位，并且控制器会进入 STOPPED 状态。

## 下载应用程序

作用: 将可执行的应用程序加载到 RAM 存储器中。也可以在闪存中创建启动应用。

启动条件: RUNNING、STOPPED、HALT 和 EMPTY 状态。

发出下载应用程序命令的方法:

- **SoMachine:**

存在 2 个选项用于下载完整应用程序:

- “下载”命令。
- “多重下载”命令。

有关应用程序下载命令的重要信息, 请参阅控制器状态图。

- **FTP:** 使用 FTP 将启动应用加载到闪存中。更新的文件会在下一次重新启动时应用。
- **SD 卡:** 使用控制器 SD 卡插槽中的 SD 卡加载启动应用程序。更新的文件会在下一次重新启动时应用。有关详细信息, 请参阅使用 SD 卡传输文件。

SoMachine 下载命令的作用:

1. 停止然后现有应用程序, 然后将其擦除。
2. 如果有效, 则会加载新应用程序, 并且控制器采用 STOPPED 状态。
3. 擦除强制。
4. 复位针对检测到的错误的诊断指示。
5. 保留变量的值复位为其初始化值。
6. 保持任何现有保留 - 持久性变量的值。
7. 所有非定位和非剩余变量都复位为其初始化值。
8. 保持前 1000 个 %MW 寄存器的值。
9. %MW1000 至 %MW59999 寄存器的值复位为 0。
10. 所有现场总线通讯都会停止, 然后在下载完成后启动新应用程序的所有已配置现场总线。
11. 嵌入式专用 I/O 都复位为用户以前配置的缺省值, 然后在下载完成后设置为用户配置的新缺省值。
12. 所有其他 I/O 都复位为其初始化值, 然后在下载完成后将其设置为用户配置的新默认值。
13. 在以下情况中, 将读取 (参见第 185 页) 后配置文件:

有关变量的详细信息, 请参阅剩余变量 (参见第 70 页)。

FTP 或 SD 卡下载命令的作用:

下一次重新启动后才能看到这些作用的效果。下一次重新启动时, 效果与无效环境下的重新启动相同。请参阅重新启动 (参见第 66 页)。

## 错误检测、类型和管理

### 管理检测到的错误

控制器管理 3 种检测到的错误：

- 外部检测到的错误
- 应用程序检测到的错误
- 系统检测到的错误

下表描述了可以检测到的错误类型：

检测到的错误类型	描述	生成的控制器状态
检测到外部错误	<p>外部错误可由处于 <b>RUNNING</b> 或 <b>STOPPED</b> 状态时的系统检测到，但不会影响持续的控制器状态。在以下情况下会检测到外部错误：</p> <ul style="list-style-type: none"> <li>● 连接的设备向控制器报告检测到的错误。</li> <li>● 控制器检测到外部设备出现错误（无论是否报告检测到错误），例如当外部设备正在通讯，但未针对用于控制器而正确配置时。</li> <li>● 控制器检测到输出状态存在错误。</li> <li>● 控制器检测到与设备的通讯中断。</li> <li>● 控制器针对不存在或未检测到的扩展模块而配置。</li> <li>● 闪存中的启动应用程序与 <b>RAM</b> 中的不相同。</li> </ul> <p>示例：</p> <ul style="list-style-type: none"> <li>● 输出短路</li> <li>● 缺少扩展模块</li> <li>● 通讯中断</li> <li>● 等。</li> </ul>	<b>RUNNING</b> 但检测到外部错误 或 <b>STOPPED</b> 并检测到外部错误
检测到应用程序错误	<p>遇到错误的编程或超过任务看门狗阈值时，会检测到应用程序错误。</p> <p>示例：</p> <ul style="list-style-type: none"> <li>● 任务（软件）看门狗例外</li> <li>● 执行未知功能</li> <li>● 等。</li> </ul>	<b>HALT</b>
检测到系统错误	<p>当控制器在运行时期间进入无法管理的条件时，会检测到系统错误。大多数此类条件由固件或硬件例外引起，但在有时可能是由于编程不正确而导致检测到系统错误，例如尝试在运行时写入保留的存储器时。</p> <p>示例：</p> <ul style="list-style-type: none"> <li>● 系统（硬件）看门狗溢出</li> <li>● 超过定义的数组大小</li> <li>● 等。</li> </ul> <p><b>注意：</b>有些检测到的系统错误可由运行时进行管理，因此可以像检测到的应用程序错误一样处理这些错误。</p>	<b>BOOTING</b> → <b>EMPTY</b>

**注意：**有关诊断的详细信息，请参阅 M241 PLCSystem 库指南。

## 剩余变量

### 概述

剩余变量在断电、重新启动、复位和应用程序下载时保留变量值。剩余变量有多种类型，分别声明为“保留”、“持久”或者组合为“保留持久性”。

**注意：** 对于此控制器，声明为持久变量的变量行为与声明为保留持久性变量的变量相同。

下表描述了各种情况下剩余变量的行为：

操作	VAR	VAR RETAIN	VAR GLOBAL PERSISTENT RETAIN
对应用程序进行在线修改	X	X	X
停止	X	X	X
电源重置	-	X	X
热复位	-	X	X
冷复位	-	-	X
初始值复位	-	-	-
应用程序下载	-	-	X
<b>X</b> 保持值 - 重新初始化值			

**注意：** 前 1000 个 %MW 会自动持久性保留，前提是没有变量与它们关联。它们的值在重新启动 / 热复位 / 冷复位后得到保留。其他 %MW 作为 VAR 进行管理。

例如，如果程序中具有：

```
VAR myVariable AT %MW0 : WORD; END_VAR
```

%MW0 的行为方式类似于 myVariable（既不是保留，也不是持久）。

### 添加保留持久性变量

在 **PersistentVars** 窗口中声明保留持久性 (**VAR GLOBAL PERSISTENT RETAIN**) 符号：

步骤	操作
1	在 <b>应用程序树</b> 中选择 <b>应用程序</b> 节点。
2	单击  。
3	选择 <b>添加其他对象</b> → <b>持久性变量</b>
4	单击 <b>添加</b> 。 <b>结果：</b> 将显示 <b>PersistentVars</b> 窗口。

---

# 章 8

## 控制器设备编辑器

---

### 简介

本章介绍如何配置控制器。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
控制器参数	72
控制器选择	74
PLC 设置	75
服务	77

## 控制器参数

### 控制器参数

要打开设备编辑器，请双击**设备树**中的 **MyController**：



## 选项卡说明

选项卡	描述	限制
应用程序	显示控制器上当前正在运行的应用程序，并可从控制器中删除应用程序。	仅在线模式
控制器选择 (参见第 74 页)	<p>管理从 PC 到控制器的连接。</p> <ul style="list-style-type: none"> <li>让您找到网络中的控制器。</li> <li>显示可用控制器列表，以便您可以连接到所选控制器并管理控制器中的应用程序。</li> <li>让您从设备编辑器中以物理方式识别控制器。</li> <li>让您更改控制器的通讯设置。</li> </ul> <p>基于通讯设置，通过 NetManage 或通过“活动路径”检测控制器列表。要访问通讯设置，请单击菜单栏中的项目 → 项目设置 ...。有关详细信息，请参阅 SoMachine 编程指南 (通讯设置)。</p>	仅在线模式
文件 (参见第 31 页)	PC 与控制器之间的文件管理。 通过此选项卡一次只能看到一个 Logic Controller 磁盘。插入 SD 卡时，此文件将显示 SD 卡的内容。或者，此选项卡将显示控制器的内部闪存的 /usr 目录的内容。	仅在线模式
日志	查看控制器日志文件。	仅在线模式
PLC 设置 (参见第 75 页)	<p>以下项目的配置：</p> <ul style="list-style-type: none"> <li>应用程序名称</li> <li>处于停止模式时的 I/O 行为</li> <li>总线循环选项</li> </ul>	—
服务 (参见第 77 页)	用于配置控制器的在线服务 (RTC、设备标识)。	仅在线模式
I/O 映射	I/O 设备的输入和输出通道在项目 (应用程序) 变量上的映射。	—
任务显示	显示 I/O 及其分配给任务情况的列表。	仅在编译后
用户和组	为支持在线用户管理的设备提供用户和组选项卡。它可以设置用户和访问权限组并向他们分配访问权限，以便控制在线模式下对 SoMachine 项目和设备的访问。 有关详细信息，请参阅 SoMachine 编程指南。	—
状态	未提供任何信息。	—
信息	显示与设备有关的一般信息 (名称、描述、供应商、版本、图像)。	—

## 控制器选择

### 简介

使用此选项卡可管理从 PC 到控制器的连接：

- 让您找到网络中的控制器。
- 显示可用控制器列表，以便您可以连接到所选控制器并管理控制器中的应用程序。
- 让您从设备编辑器中以物理方式识别控制器。
- 让您更改控制器的通讯设置。

### 处理通讯设置

使用**处理通讯设置**窗口可以更改以太网通讯设置。为此，请单击**控制器选择**选项卡。此时将显示网络中可用的控制器列表。选择并右键单击所需的行，然后在上下文菜单中单击**处理通讯设置 ...**。

在**处理通讯设置**窗口中，可通过以下 2 种方式配置以太网设置：

- 不使用**永久保存设置**选项：  
配置通讯参数并单击**确定**。这些设置将立即考虑，在控制器复位时不予保留。对于接下来的复位，将考虑应用程序中配置的通讯参数。
- 使用**永久保存设置**选项：  
您也可以在单击**确定**之前选中**永久保存设置**选项。选中此选项后，在复位时将始终考虑在此处配置的以太网参数，而不是 SoMachine 应用程序中配置的以太网参数。

有关设备编辑器的**控制器选择**视图的详细信息，请参阅 SoMachine 编程指南。

## PLC 设置

### 概述

下图所示为 **PLC 设置** 选项卡：

元素	描述	
用于 I/O 处理的应用程序	缺省情况下设置为“应用程序”，因为控制器中只有一个应用程序。	
PLC 设置	处于“停止”状态时更新 IO	如果已激活此选项（缺省），在控制器停止时，输入和输出通道的值也会更新。
	处于“停止”状态的输出的行为	从选择列表中选择下列选项之一，以配置在控制器停止时应如何处理输出通道的值： <ul style="list-style-type: none"> <li>● 保持当前值</li> <li>● 设置所有输出为默认值</li> </ul>
	更新所有设备中的所有变量	如果已激活此选项，那么对于使用当前控制器配置的所有设备，所有 I/O 变量在总线循环任务的每次循环中都会更新。这对应于 <b>一直更新变量</b> 选项（可以在 I/O 映射对话框中针对每个设备单独设置该选项）。
总线循环选项	总线循环任务	此配置设置是在应用程序设备树中使用的 <b>所有总线循环任务</b> 参数的父级。具有循环调用的某些设备（如 <b>CANopen 管理器</b> ）可以附加到特定任务。在设备中，将此设置设置为 <b>使用父总线循环设置</b> 时，会使用为控制器配置的设置。此选择列表显示活动应用程序中当前定义的所有任务。缺省设置为 <b>MAST</b> 任务。 <b>注意：</b> < 未指定 > 意味着任务处于“最慢的循环任务”模式下。

元素		描述
附加设置	生成 IO 映射的强制变量	未使用。
	启用设备诊断	未使用。
启动模式选项	启动模式	<p>此选项定义打开电源时的启动模式。有关详细信息，请参阅状态行为图 (参见第 57 页)。</p> <p>使用此选项选择以下启动模式之一：</p> <ul style="list-style-type: none"> <li>● 以上一个状态启动</li> <li>● 在“停止”状态下启动</li> <li>● 在“运行”状态下启动</li> </ul>

## 服务

### “服务”选项卡

服务选项卡分为 3 个部分：

- RTC 配置
- 设备标识
- 后配置

下图所示为服务选项卡：

**注意：**要获取控制器信息，必须先连接到控制器。

元素		描述
RTC 配置	PLC 时间	显示从控制器读取的日期 / 时间。此只读字段最初为空。要读取并显示控制器上保存的日期 / 时间，请单击 <b>读取</b> 按钮。
	当地时间	用于定义通过单击 <b>写入</b> 按钮发送到控制器的日期和时间。命令成功后，将出现一个消息框通知您。使用当前 PC 设置初始化“当地时间”字段。
	与当地的日期 / 时间同步	用于直接发送当前的 PC 设置。命令成功后，将出现一个消息框通知您。
设备标识		连接后，将显示选定控制器的固件版本、引导版本和协处理器版本。
后配置		用于显示被后配置 ( 参见第 185 页 ) 覆盖的应用程序参数。



---

## 章 9

### 内嵌输入和输出配置

---

#### 嵌入式 I/O 配置

##### 概述

使用嵌入式 I/O 功能可配置控制器输入和输出。

M241 Logic Controller 提供:

I/O 类型	24 I/O 参考号	40 I/O 参考号
	TM241•24•	TM241•40•
快速输入	8	8
常规输入	6	16
快速输出	4	4
常规输出	6	12

##### 访问 I/O 配置窗口

按照以下步骤访问 I/O 配置窗口:

步骤	描述
1	双击设备树中的 <b>DI</b> (数字量输入) 或 <b>DQ</b> (数字量输出)。请参阅设备树 (参见第 20 页)。
2	选择 <b>I/O 配置</b> 选项卡。

### 数字量输入的配置

下图显示了数字量输入的 I/O 配置选项卡：

I/O 映射		I/O 配置				
参数	类型	值	缺省值	单位	描述	
输入参数						
I0					已经	
过滤器	WORD 枚举	无	无	毫秒	过滤	
锁存	BYTE 枚举	否	否	毫秒	锁存	
事件	BYTE 枚举	否	否		事件	
I1					已经	
过滤器	WORD 枚举	无	无	毫秒	过滤	
锁存	BYTE 枚举	否	否	毫秒	锁存	
事件	BYTE 枚举	否	否		事件	
I2						
过滤器	WORD 枚举	无	无	毫秒	过滤	
锁存	BYTE 枚举	否	否	毫秒	锁存	
事件	BYTE 枚举	否	否		事件	

**注意：**有关 I/O 映射选项卡的详细信息，请参阅 SoMachine 编程指南。

## 数字量输入配置参数

您可以为每个数字量输入配置以下参数：

参数	值	描述	约束条件
过滤器	无 1 毫秒 4 毫秒（缺省） 12 毫秒	减少控制器输入上的噪声影响。	禁用 <b>锁存</b> 和 <b>事件</b> 时可用。 在其他情况下，禁用此参数并且其值设为 <b>无</b> 。
锁存	否 * 是	允许捕捉和记录幅度宽度短于控制器扫描时间的传入脉冲。	此参数仅可用于快速输入 I0 到 I7。 以下情况下可用： 禁用 <b>事件</b> ，同时禁用 <b>过滤器</b> 。 仅在 <b>MAST</b> 任务中使用锁存输入。
事件	否 * 上升沿 下降沿 上升和下降沿	事件检测	此参数仅可用于快速输入 I0 到 I7。 以下情况下可用： 禁用 <b>事件</b> ，同时禁用 <b>过滤器</b> 。
跳动	0.000 毫秒 0.001 毫秒 0.002 毫秒 * 0.005 毫秒 0.010 毫秒 0.05 毫秒 0.1 毫秒 0.5 毫秒 1 毫秒 5 毫秒	减少控制器输入上的跳动影响。	启用 <b>锁存</b> 或启用 <b>事件</b> 时可用。 在其他情况下，禁用此参数并且其值设为 <b>0.002</b> 。
运行 / 停止输入	无 I0 ... I7	“运行/停止”输入可用于运行或停止控制器中的程序	选择一个输入用作“运行 / 停止输入”。
* 参数缺省值			

**注意：** 如果参数不可用，则选择为灰色，处于非活动状态。

## 运行 / 停止输入

下表提供不同状态：

输入状态	结果
状态 0	停止控制器并忽略外部运行命令。
上升沿	从“已停止”状态开始以“运行”状态启动应用程序，前提是不与“运行 / 停止”开关位置发生冲突。
状态 1	应用程序可由以下各项控制： <ul style="list-style-type: none"> <li>● SoMachine（运行 / 停止）</li> <li>● 硬件运行 / 停止开关</li> <li>● 应用程序（控制器命令）</li> <li>● 网络命令（运行 / 停止命令）</li> </ul> 可通过 Web 服务器命令使用运行 / 停止命令。

**注意：**即使没有在控制器设备编辑器（PLC 设置选项卡）（参见第 75 页）中选中处于“停止”状态时更新 I/O 选项，也会管理“运行 / 停止”输入。

**注意：**分配给已配置的专用功能的输入无法配置为“运行 / 停止”输入。

有关控制器状态和状态转变的详细信息，请参阅控制器状态图（参见第 50 页）。

### ⚠ 警告

#### 机器或过程意外启动

- 在对运行 / 停止输入加电之前，请验证机器或过程环境的安全状态。
- 使用运行 / 停止输入可帮助防止从远程位置意外启动。

不遵循上述说明可能导致人员伤亡或设备损坏。

## 数字量输出的配置

下图显示了数字量输出的 I/O 配置选项卡：

参数	类型	值	缺省值	单位	描述
常规参数					
报警输出	WORD 枚举	无	无		
重置输出模式	BYTE 枚举	自动	自动		
同步					
使本地输出的抖动达到最小	BYTE 枚举	否	否		启用

**注意：**有关 I/O 映射选项卡的详细信息，请参阅 SoMachine 编程指南。

## 数字量输出配置参数

下表显示了不同参数的功能：

参数	功能
<b>常规参数</b>	
Alarm Output	选择要用作警报输出 (参见第 83 页) 的输出。
Rearming Output Mode	选择重置输出模式 (参见第 83 页)。
<b>同步</b>	
使本地输出的抖动达到最小	选择此选项可使本地输出的抖动达到最小 (参见第 84 页)。

**注意：** 如果参数不可用，则选择为灰色，处于非活动状态。

## 警报输出

当控制器处于“运行”状态且应用程序未在断点处停止时，此输出设置为逻辑 1。

分配给已配置的专用功能的输出无法配置为警报输出。

**注意：** 当任务在断点处停止时将警报输出设置为 0，该警报输出表示控制器已停止执行应用程序。

## 重置输出模式

的快速输出使用推 / 挽技术。在检测到错误（短路或过热）时，输出会进入三态，其状态将由状态位和 PLC\_R.i\_wLocalIOStatus 表示。

行为可以分为以下两种：

- **自动重置：** 检测到的错误得到纠正后，输出会根据分配给它的当前值再次进行设置，诊断值也将复位。
- **手动重置：** 检测到错误后，状态将被记住，输出也会被强制变为三态，直到用户手动清除此状态（请参阅“I/O 映射通道”）为止。

如果出现短路或电流过载，则公共输出组会自动进入热保护模式（所有输出都设置为 0），随后会定期重置（每秒），以测试连接状态。但是，您必须了解这种重置对所控制的机器或过程的影响。

### 警告

#### 机器意外启动

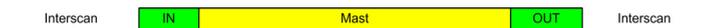
如果不想对机器或过程执行输出的自动重置，请禁用此功能。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

### 使本地输出的抖动达到最小

通过此选项，可以按照可预测的时间间隔读取或设置嵌入式 I/O，不管任务持续时间如何。通过将物理输出的写入延迟到下一个总线循环任务的读取输入操作开始时，可将输出上的抖动减到最小。任务的结束时间通常比开始时间更难预测。

输入 / 输出阶段的正常安排是：



在选择**使本地输出的抖动达到最小**选项时，IN 和 OUT 阶段的安排将变成：



---

# 章 10

## 专用功能配置

---

### 概述

本章介绍了 M241 的专用功能。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
专用配置概述	86
计数功能	88
脉冲发生器内嵌功能	90

## 专用配置概述

### 简介

M241 Logic Controller 中可用的快速输入和快速输出可以连接到专用功能。

M241 Logic Controller 支持下列专用功能：

功能		描述
计数器	HSC Simple	HSC 功能可以通过连接到快速输入的传感器、开关等等来执行脉冲的快速计数。 《HSC 库指南 ( 参见 <i>Modicon M241 Logic Controller, 高速计数, HSC 库指南</i> )》中描述了 HSC 功能
	HSC 主单相	
	HSC 主双相	
	频率计	
	周期计	
脉冲发生器	PTO ( 参见 <i>Modicon M241 Logic Controller, PTOPWM, 库指南</i> )	PTO 功能提供了 2 个脉冲输出通道，以在开放回路模式下控制独立线性单轴步进器或伺服驱动器。
	PWM ( 参见 <i>Modicon M241 Logic Controller, PTOPWM, 库指南</i> )	PWM 功能可以在专用输出通道上生成具有可变占空比的方波信号。
	频率发生器 ( 参见 <i>Modicon M241 Logic Controller, PTOPWM, 库指南</i> )	频率发生器功能可以在专用输出通道上生成具有固定占空比 (50%) 的方波信号。

### 注意：

- 如果某个常规输入作为“运行 / 停止”使用，则不能为专用功能使用。
- 如果某个常规输出作为“报警”使用，则不能为专用功能使用。

有关更多详细信息，请参阅内嵌功能配置 ( 参见第 85 页 )。

## 配置专用功能

要配置专用功能，请执行以下步骤：

步骤	描述
1	<p>在<b>设备树</b>中双击<b>计数器</b>或<b>脉冲发生器</b>节点。  <b>结果：</b>将显示<b>计数器</b>或<b>脉冲发生器</b>功能窗口：</p> 
2	<p>双击<b>值</b>，然后选择要分配的功能类型。  <b>结果：</b>将显示专用功能的参数。</p>

## 常规 I/O 中的专用功能 I/O

常规 I/O 中的专用功能 I/O：

- 即使配置为专用功能，也可以通过标准存储器变量来读取输入。
- 如果输入已经配置为“运行 / 停止”输入，则不能将此输入配置为专用功能。
- 如果输出已经配置为“警报”，则不能在专用功能中对其进行配置。
- 短路管理仍然适用于所有输出。输出的状态可用。
- 专用功能未使用的所有 I/O 都可以用作任何其他常规 I/O。

在专用功能（锁存、HSC..）中使用输入时，积分器过滤器将被替换为抗跳动过滤器。过滤器值是在配置屏幕中进行配置。

## 计数功能

### 概述

计数功能可以对来自与专用快速输入相连的传感器、编码器、开关等的脉冲执行快速计数。

有以下 2 个类型的内嵌计数功能：

- **Simple** 类型：单个输入的计数器。
- **Main** 类型：最多可使用 6 路快速输入和 2 路反射输出的计数器。

根据内嵌计数功能的不同，可以在 SoMachine 中配置 5 种类型的计数器：

- **HSC Simple**
- **HSC Main 单相**
- **HSC Main 双相**
- **频率计**
- **周期计**

频率计类型和周期计类型基于 **HSC Main** 类型。

### 访问计数功能配置窗口

按照以下步骤访问内嵌计数功能配置窗口：

步骤	描述
1	<p>双击设备树中的计数器。 “计数功能”窗口随即显示：</p> 
2	<p>双击值并选择要分配的计数功能类型。</p>

## 计数功能 配置窗口

下图显示 HSC 配置窗口的示例：



下表介绍计数器配置窗口的各个区域：

编号	操作
1	功能的实例名称及当前配置的计数功能类型。
2	单击 + 配置计数功能的新实例。
3	双击值列将显示可用的计数器功能类型列表。
4	双击实例名称值可编辑功能的实例名称。 SoMachine 会自动提供实例名称。可以编辑实例名称参数，并且此参数可用于定义实例名称。但是，无论实例名称是软件定义还是用户定义，都请使用相同实例名称作为处理计数器的功能块的输入，如计数器编辑器中所定义。
5	单击每个参数旁边的加号以访问其设置，对该参数进行配置。 可用的参数取决于使用的模式。

有关配置参数的详细信息，请参阅 M241 HSC 库指南。

## 脉冲发生器内嵌功能

### 概述

M241 提供的脉冲发生内嵌功能包括：

**PTO** PTO（脉冲串输出）实现的数字技术可以精确定位电机驱动器的开路控制。

**PWM** PWM（脉冲宽度调制）功能可以在专用输出通道上生成具有可调整占空比和频率的可编程方波信号。

**FG** FG（频率发生器）功能可以在专用输出通道上生成具有固定占空比 (50%) 的方波信号。

### 访问“脉冲发生器”配置窗口

按照以下步骤，访问“脉冲发生器”配置窗口：

步骤	描述												
1	<p>在<b>设备树</b>上双击<b>脉冲发生器</b>。</p> <p>将显示“脉冲发生功能”窗口。</p>  <table border="1"> <thead> <tr> <th>参数</th> <th>类型</th> <th>值</th> <th>缺省值</th> <th>单位</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>... 脉冲发生功能</td> <td>WORD 枚举</td> <td>无</td> <td>无</td> <td></td> <td>选择脉冲发生应用</td> </tr> </tbody> </table>	参数	类型	值	缺省值	单位	描述	... 脉冲发生功能	WORD 枚举	无	无		选择脉冲发生应用
参数	类型	值	缺省值	单位	描述								
... 脉冲发生功能	WORD 枚举	无	无		选择脉冲发生应用								
2	<p>双击<b>值</b>，然后选择要分配的脉冲发生器功能类型。</p>												

## 脉冲发生器配置窗口

下图显示了用于配置 PTO、PWM 或 FG 功能的**脉冲发生器**配置窗口的示例：

参数	类型	值	缺省值	单位	描述
脉冲发生功能	WORD 枚举	PWM	无		选择脉冲发生应用
常规					
实例名称	STRING	'PWM_0'	"		选择 PWM 的实例名称
输出位置	SINT 枚举	Q0	Q0		选择用于 A 的 PLC 输出
控制输入					
SYNC 输入					
位置	SINT 枚举	已禁用	已禁用		选择用于预设功能的 PLC 输入
EN 输入					
位置	SINT 枚举	已禁用	已禁用		选择用于启用的 PLC 输入

下表介绍**脉冲发生器**配置窗口的各个区域：

编号	操作
1	功能的实例名称以及当前配置的脉冲发生器功能类型。
2	单击 + 可配置脉冲发生器功能的新实例。
3	双击 <b>值</b> 列可显示可用的脉冲发生器功能类型的列表。
4	双击 <b>实例名称</b> 值可编辑功能的实例名称。 SoMachine 会自动提供 <b>实例名称</b> 。可以编辑 <b>实例名称</b> 参数，并且此参数可用于定义实例名称。但是，无论 <b>实例名称</b> 是软件定义还是用户定义，都请使用相同实例名称作为处理计数器的功能块的输入，如 <b>计数器</b> 编辑器中所定义。
5	单击每个参数旁边的加号来配置参数，以访问其设置。 可用参数取决于使用的脉冲发生器类型。

有关配置参数的详细信息，请参阅《M241 PTO/PWM/FG 库指南》。



---

# 章 11

## 卡盒配置

---

### TMC4 卡盒配置

#### 简介

Modicon M241 Logic Controller 支持以下卡盒：

- TMC4 标准卡盒
- TMC4 应用程序卡盒

有关 TMC4 卡盒配置的更多信息，请参阅《TMC4 卡盒编程指南 (参见 *Modicon TMC4, 卡盒, 编程指南*)》。

 <b>警告</b>
<b>意外的设备操作</b> <ul style="list-style-type: none"><li>● 仅使用 Schneider Electric 认可的可与本设备配合使用的软件。</li><li>● 每次更改物理硬件配置后，请更新应用程序。</li></ul> <b>不遵循上述说明可能导致人员伤亡或设备损坏。</b>



#### 添加 TMC4 卡盒

要将卡盒添加到控制器，请在**硬件目录**中选择卡盒，将其拖到**设备树**，然后放在某个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 (参见 *SoMachine, 编程指南*)
- 使用上下文菜单或加号按钮 (参见 *SoMachine, 编程指南*)



---

# 章 12

## 扩展模块配置

---

### 概述

本章介绍如何为 Modicon M241 Logic Controller 配置 TM4、TM3 和 TM2 扩展模块。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
I/O 配置一般做法	96
I/O 总线配置	97
TM4 扩展模块配置	98
TM3/TM2 扩展模块配置	99

## I/O 配置一般做法

### 匹配硬件和软件配置

可在控制器中嵌入的 I/O 独立于采用 I/O 扩展的形式添加的 I/O。程序中的逻辑 I/O 配置应与安装的物理 I/O 配置匹配，这十分重要。如果对 I/O 扩展总线添加或删除任何物理 I/O，或根据控制器参考号，对控制器进行添加或删除操作（以卡盒的形式），则必须更新应用程序配置。这也适用于安装中包含的任何现场总线设备。否则，I/O 扩展将无法再正常工作，而控制器中可能存在的嵌入式 I/O 会继续运行。

### 警告

#### 意外的设备操作

每次添加或删除任何类型的 I/O 扩展，或添加或删除现场总线上的任何设备时，都需更新程序配置。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

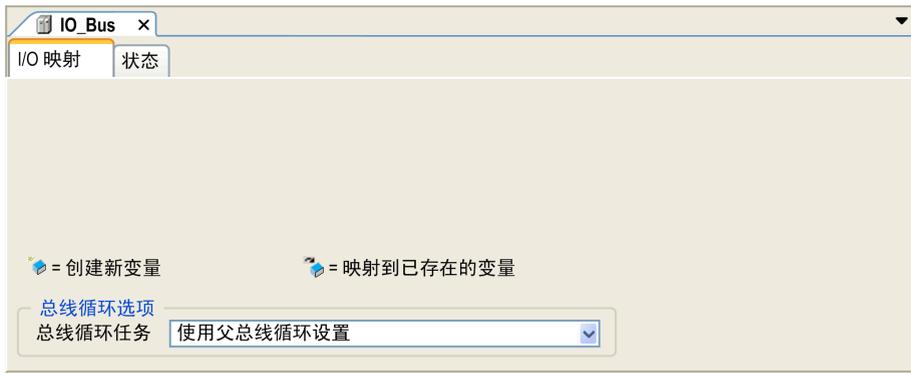
## I/O 总线配置

### 概述

您可以通过 I/O 总线配置来选择驱动 TM3 和 CANopen 物理交换的任务。这样还可以覆盖在 **PLC 设置** (参见第 75 页) 中定义的配置。

### 配置 I/O 总线

按照以下步骤配置 I/O 总线：

步骤	描述
1	<p>双击<b>设备树</b>中的 <b>IO_Bus</b>。</p> <p><b>结果：</b> <b>IO_Bus</b> 编辑器选项卡随即显示：</p> 
2	<p>从列表中将<b>总线循环任务</b>设置为以下值之一：</p> <ul style="list-style-type: none"> <li>● <b>使用父总线循环设置</b> (缺省) 按照 <b>PLC 设置</b>中的定义设置总线交换任务。</li> <li>● <b>MAST</b> 设置主任务进行总线交换，不管 <b>PLC 设置</b>中定义的任务如何。</li> </ul>

## TM4 扩展模块配置

### 简介

Modicon M241 Logic Controller 支持 TM4 通讯扩展模块。

有关 TM4 扩展模块配置的详细信息，请参阅 TM4 扩展模块配置编程指南。

### 警告

#### 意外的设备操作

- 仅使用 Schneider Electric 认可的可与本设备配合使用的软件。
- 每次更改物理硬件配置后，请更新应用程序。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

### 添加扩展模块

要将扩展模块添加到控制器，请在**硬件目录**中选择扩展模块，将其拖动到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的详细信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南* )
- 使用上下文菜单和加号按钮 ( 参见 *SoMachine, 编程指南* )

## TM3/TM2 扩展模块配置

### 简介

Modicon M241 Logic Controller 支持以下扩展模块：

- TM3 扩展模块
  - 数字量 I/O 模块
  - 专用模块
  - 发射器和接收器模块
- TM2 扩展模块
  - 数字量 I/O 模块
  - 模拟量 I/O 模块
  - 专用模块
  - 通讯模块

有关 TM3 和 TM2 扩展模块配置的详细信息，请分别参阅 TM3 扩展模块配置编程指南和 TM2 扩展模块配置编程指南。

### 警告

#### 意外的设备操作

- 仅使用 Schneider Electric 认可的可与本设备配合使用的软件。
- 每次更改物理硬件配置后，请更新应用程序。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

### 添加扩展模块

要将扩展模块添加到控制器，请在**硬件目录**中选择扩展模块，将其拖动到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的详细信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南* )
- 使用上下文菜单和加号按钮 ( 参见 *SoMachine, 编程指南* )



---

# 章 13

## 以太网配置

---

### 简介

本章介绍如何配置 Modicon M241 Logic Controller 的以太网网络接口。

### 本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
13.1	以太网服务	102
13.2	防火墙配置	128
13.3	以太网可选设备	138

# 节 13.1

## 以太网服务

---

### 本节包含了哪些内容？

本节包含了以下主题：

主题	页
以太网服务	103
IP 地址配置	105
Modbus TCP 服务器 / 客户端	110
Web 服务器	112
FTP 服务器	125
SNMP	127

## 以太网服务

### 以太网服务

控制器支持以下服务：

- Modbus TCP 服务器 (参见第 110 页)
- Modbus TCP 客户端 (参见第 110 页)
- Web 服务器 (参见第 112 页)
- FTP 服务器 (参见第 125 页)
- SNMP (参见第 127 页)
- EthernetIP 设备 (参见第 140 页)
- Modbus TCP 从站设备 (参见第 159 页)
- IEC VAR ACCESS (参见第 104 页)

### 以太网协议

控制器支持以下协议：

- IP (Internet Protocol)
- UDP (User Datagram Protocol)
- TCP (Transmission Control Protocol)
- ARP (Address Resolution Protocol)
- ICMP (Internet Control Messaging Protocol)
- IGMP (Internet Group Management Protocol)

### TCP 服务器连接

下表显示了 TCP 服务器的最大连接数：

连接类型	服务器最大连接数
Modbus 服务器	8
EthernetIP 设备	16
FTP 服务器	4
Web 服务器	10

每个基于 TCP 的服务器都管理其各自的一组连接。

如果客户端尝试打开的连接超过了池大小，则控制器会关闭最先打开的连接。

如果客户端尝试打开新连接时所有连接都忙（正在进行数据交换），新连接将被拒绝。

只要控制器继续处于工作状态 (RUN, STOP, HALT)，所有服务器连接都将保持打开。

退出或进入工作状态 (RUN, STOP, HALT) 时，所有服务器连接都将关闭，但断电情况除外（因为控制器来不及关闭连接）。

### 可用服务

借助以太网通讯，控制器可以支持 **IEC VAR ACCESS** 服务。借助 **IEC VAR ACCESS** 服务，可以实现控制器和 HMI 之间的数据交换。

控制器也支持**网络变量**服务。通过**网络变量**服务，可以实现控制器之间的数据交换。

**注意：**有关详细信息，，请参阅 SoMachine 编程指南。

## IP 地址配置

### 简介

可通过以下各种方法分配控制器的 IP 地址：

- 由 DHCP 服务器分配地址
- 由 BOOTP 服务器分配地址
- 固定 IP 地址
- 后配置文件 ( 参见第 185 页 )。如果存在后配置文件，则该分配方法优先于其他方法。

可通过以下方法动态更改 IP 地址：

- 通过 SoMachine 中的控制器选择 ( 参见 *SoMachine, 编程指南* ) 选项卡。

**注意：** 如果尝试的寻址方法不成功，控制器将使用衍生自 MAC 地址的缺省 IP 地址 ( 参见第 108 页 ) 启动。

认真管理 IP 地址，因为网络上的每个设备都需要唯一的地址。使多个设备具有相同的 IP 地址会导致网络和相关设备的意外操作。

### 警告

#### 意外的设备操作

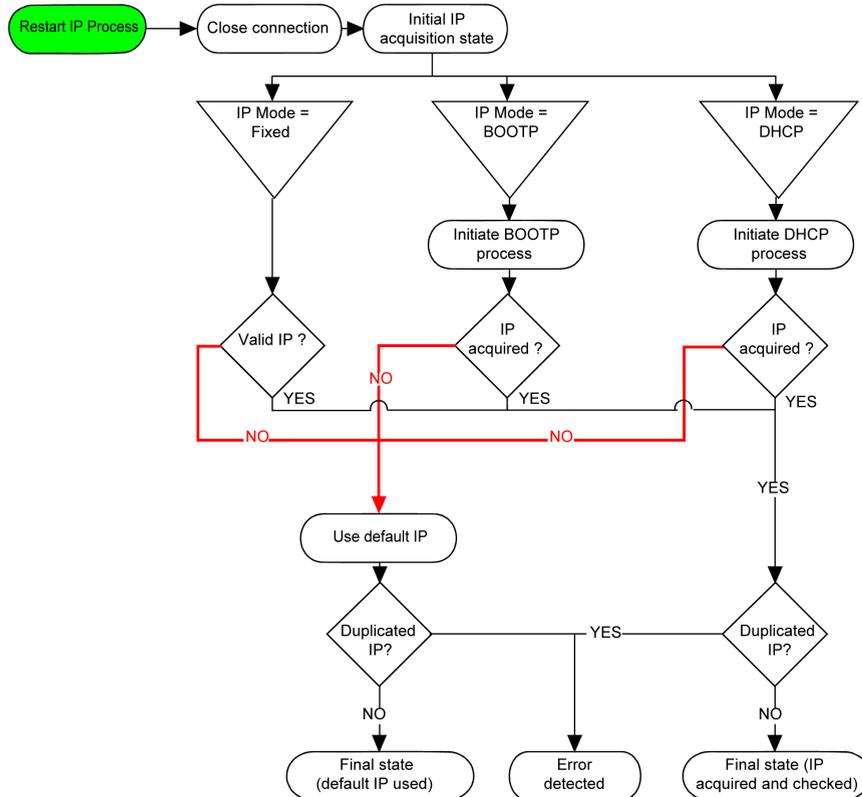
- 确认在网络或远程链路上仅配置了一个主控制器。
- 确认所有设备均有唯一的地址。
- 从系统管理员处获取 IP 地址。
- 在将系统投入使用之前请确认设备的 IP 地址是唯一的。
- 请不要将同一个 IP 地址分配给网络上的任何其他设备。
- 在克隆包括以太网通讯的任何应用程序后将 IP 地址更新为唯一的地址。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

**注意：** 确认系统管理员对网络和子网上所有已分配的 IP 地址留有记录，并向系统管理员通知已执行的所有配置更改。

## 地址管理

下图显示了控制器的不同地址系统的类型：



**注意：**如果编程为使用 DHCP 或 BOOTP 寻址方法的设备无法联系其相应的服务器，则控制器使用缺省 IP 地址。但是，它将不断地重复发出请求。

在以下情况中，IP 进程会自动重新启动：

- 控制器重新启动
- 以太网电缆重新连接
- 下载应用程序（如果 IP 参数发生更改）
- 在前一次寻址尝试不成功时，检测到 DHCP 或 BOOTP 服务器。

## 以太网配置

在**设备树**中，双击 **Ethernet\_x:**

The screenshot shows the '以太网' (Ethernet) configuration window. It is divided into two main sections: '配置参数' (Configuration Parameters) and '安全参数' (Security Parameters).

**配置参数 (Configuration Parameters):**

- 接口名称 (Interface Name): ether\_0
- 网络名称 (Network Name): my\_Device
- IP 地址分配方式 (IP Address Assignment Method):
  - DHCP 分配的 IP 地址 (DHCP assigned IP address)
  - BOOTP 分配的 IP 地址 (BOOTP assigned IP address)
  - 固定 IP 地址 (Fixed IP address)
- IP 地址 (IP Address): 0.0.0.0
- 子网掩码 (Subnet Mask): 0.0.0.0
- 网关地址 (Gateway Address): 0.0.0.0
- 以太网协议 (Ethernet Protocol): 以太网 2 (Ethernet 2)
- 传输速率 (Transmission Rate): 自动 (Automatic)

**安全参数 (Security Parameters):**

- SoMachine 协议处于活动状态 (SoMachine protocol is active)
- Modbus 服务器处于活动状态 (Modbus server is active)
- Web 服务器处于活动状态 (Web server is active)
- FTP 服务器处于活动状态 (FTP server is active)
- 发现协议处于活动状态 (Discovery protocol is active)
- SNMP 协议处于活动状态 (SNMP protocol is active)

配置的参数如下所述：

配置参数	描述
接口名称	网络链路的名称。
网络名称	作为设备名，用于通过 DHCP 检索 IP 地址，最多包含 16 个字符。
DHCP 分配的 IP 地址	IP 地址通过 DHCP 获得。
BOOTP 分配的 IP 地址	IP 地址通过 BOOTP 获得。
固定 IP 地址	IP 地址、子网掩码和网关地址均由用户定义。
以太网协议	使用的协议类型（Ethernet2 或 IEEE 802.3） <b>注意：</b> 如果更改以太网协议，则需要进行电源重置才会被控制器识别。
传输速率	总线上的传输速率和方向会自动配置。

## 缺省 IP 地址

缺省 IP 地址为 10.10.x.x。

缺省 IP 地址的最后 2 个字段由端口 MAC 地址最后 2 个十六进制字节对应的十进制数字组成。

端口的 MAC 地址可从放在控制器前侧的标签上获得。

缺省子网掩码为缺省 A 类子网掩码 255.0.0.0。

**注意：**MAC 地址始终以十六进制格式编写，IP 地址则采用十进制格式。请将 MAC 地址转换成十进制格式。

示例：如果 MAC 地址为 00.80.F4.01.80.F2，则缺省 IP 地址为 10.10.128.242。

**注意：**要在下载项目后考虑新 IP 地址，请通过电源重置重新启动控制器。

## 地址类别

IP 地址与以下两项相关联：

- 设备（即主机）
- 该设备所连接的网络

IP 地址始终采用 4 字节进行编码。

这些字节在网络地址和设备地址之间的分配可能会发生变化。具体分配方法由地址类别定义。

下表定义了不同的 IP 地址类别：

地址类别	字节 1				字节 2	字节 3	字节 4
A 类	0	网络 ID			主机 ID		
B 类	1	0	网络 ID			主机 ID	
C 类	1	1	0	网络 ID			主机 ID
D 类	1	1	1	0	多播地址		
E 类	1	1	1	1	0	保留地址，供以后使用	

## 子网掩码

子网掩码的作用是在同一个网络地址下为多个物理网络编址。掩码用于划分子网地址和主机 ID 的设备地址。

获取子网地址的方法是：保留 IP 地址中与包含 1 的掩码的位置相对应的位，然后用 0 替换其他位。

反之，获取主机设备子网地址的方法是：保留 IP 地址中与包含 0 的掩码的位置相对应的位，然后用 1 替换其他位。

子网地址的示例：

IP 地址	192 (11000000)	1 (00000001)	17 (00010001)	11 (00001011)
子网掩码	255 (11111111)	255 (11111111)	240 (11110000)	0 (00000000)
子网地址	192 (11000000)	1 (00000001)	16 (00010000)	0 (00000000)

**注意：** 如果没有网关，设备不在其子网中进行通讯。

## 网关

网关可将消息路由到不在当前网络中的设备。

如果没有网关，则网关地址为 0.0.0.0。

## 安全参数

安全参数	描述
<b>SoMachine 协议处于活动状态</b>	使用它可在以太网接口上停用 SoMachine 协议。停用后，将拒绝来自每台设备的每一个 SoMachine 请求，包括来自 UDP 或 TCP 连接的请求。这意味着在以太网上将不会与装有 SoMachine 的 PC、希望与此控制器交换变量的 HMI 目标、OPC 服务器或 Controller Assistant 建立连接。
<b>Modbus 服务器处于活动状态</b>	使用它可以停用 Logic Controller 的 Modbus 服务器。这意味着将忽略发送到 Logic Controller 的每一个 Modbus 请求。
<b>Web 服务器处于活动状态</b>	使用它可以停用 Logic Controller 的 Web 服务器。这意味着将忽略发送到 Logic Controller 的每一个 HTTP 请求。
<b>FTP 服务器处于活动状态</b>	使用它可以停用 Logic Controller 的 FTP 服务器。这意味着将忽略每一个 FTP 请求。
<b>Discovery 协议处于活动状态</b>	使用它可以停用 Discovery 协议。这意味着将忽略每一个 Discovery 请求。
<b>SNMP 协议处于活动状态</b>	使用它可以停用 Logic Controller 的 SNMP 服务器。这意味着将忽略每一个 SNMP 请求。

## Modbus TCP 服务器 / 客户端

### 简介

与 Modbus 串行链路不同，Modbus TCP/IP 不以层次结构为基础，而是基于客户端 / 服务器模型。

Modicon M241 Logic Controller 既可以执行客户端服务，也可以执行服务器服务，因此，它既可以启动与其他控制器和 I/O 设备的通讯，也能响应来自其他控制器、SCADA、HMI 以及其他设备的请求。

无需任何配置，控制器的嵌入式以太网端口即可支持 Modbus 服务器。

Modbus 客户端 / 服务器包含在固件中，不需用户进行任何编程操作。通过此功能，可以在 RUNNING、STOPPED 和 EMPTY 状态下对其进行访问。

### Modbus TCP 客户端

Modbus TCP 客户端无需任何配置，即可支持 PLCCommunication 库中的以下功能块：

- ADDM
- READ\_VAR
- SEND\_RECV\_MSG
- SINGLE\_WRITE
- WRITE\_READ\_VAR
- WRITE\_VAR

有关详细信息，请参阅功能块描述（参见 *SoMachine, Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide*）。

### Modbus TCP 服务器

Modbus 服务器支持以下 Modbus 请求：

功能代码 十进制（十六进制）	子功能 十进制（十六进制）	功能
1 (1h)	–	读取数字量输出 (%Q)
2 (2h)	–	读取数字输入 (%I)
3 (3h)	–	读取保持寄存器 (%MW)
6 (6h)	–	写入单个寄存器 (%MW)
8 (8h)	–	诊断
15 (Fh)	–	写入多个数字量输出 (%Q)
16 (10h)	–	写入多个寄存器 (%MW)
23 (17h)	–	读取 / 写入多个寄存器 (%MW)
43 (2Bh)	14 (Eh)	读取设备标识

## 诊断请求

下表包含数据选择代码列表：

数据选择代码	描述
0x00	保留
0x01	基本网络诊断
0x02	以太网端口诊断
0x03	Modbus TCP/ 端口 502 诊断
0x04	Modbus TCP/ 端口 502 连接表
0x05 - 0x7E	为其他公共代码保留
0x7F	数据结构偏移

## Web 服务器

### 简介

作为一种标准设备，控制器提供了带有预定义出厂内置网站的嵌入式 Web 服务器。通过这些网页不仅可以安装和控制模块，还可以诊断和监控应用程序。使用 Web 浏览器即可浏览这些页面。无需进行任何配置或编程。

使用下列 Web 浏览器可以访问 Web 服务器：

- Google Chrome (30.0 版或更高版本)
- Mozilla Firefox (1.5 版或更高版本)

Web 服务器的 TCP 连接 (参见第 103 页) 限制为至多 10 个。

**注意：** 可通过取消选中“以太网配置”选项卡 (参见第 107 页) 中的 **Web 服务器处于活动状态** 参数禁用 Web 服务器。

Web 服务器是用于读写数据的工具，它控制着控制器的状态，对应用程序中的所有数据具有完全访问权限。但是，如果担心这些功能会带来安全问题，至少必须为 Web 服务器指定一个安全密码，或者禁用 Web 服务器以防未经授权访问应用程序。一旦启用 Web 服务器，也就启用了这些功能。

通过 Web 服务器，可以远程监视控制器及其应用程序，执行各种维护活动，包括修改数据和配置参数，以及更改控制器的状态。执行远程控制前，请务必小心处理，以确保机器和过程的直接物理环境所处的状态不会给人员或财产带来安全风险。

### 警告

#### 意外的设备操作

- 如果您的特定控制器可行，请为应用程序配置和安装 RUN/STOP 输入，以便无论发送给控制器什么样的远程命令，均可以通过对控制器的启动或停止保持对控制器的本地控制。
- 为 Web 服务器定义一个安全密码，并且不允许未经授权人员或其他不符合资格的人员使用此功能。
- 确保在从远程位置操作控制器时，本地有胜任且符合资格的观察者在场。
- 试图调整数据、停止正在运行的应用程序或远程启动控制器之前，必须对它正在控制的应用程序和机器 / 进程完全了解。
- 采取必要的预防措施，以确保您是按清晰的识别文档在控制器应用程序内和远程连接上操作目标设备。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

**注意：** Web 服务器只能由经授权而且符合资格的人员使用。合格人员是指这样的人员：他们掌握了与机器的构造和运行以及应用程序及其安装控制的流程相关的知识和技能，并且受到过可识别并避免可能发生的危险的安全培训。对于使用此功能所引发的任何后果，Schneider Electric 概不负责。

## Web 服务器访问

在控制器中启用了用户权限后，通过用户权限控制对 Web 服务器的访问。有关详细信息，请参阅 **用户和组** 选项卡描述（参见第 73 页）。

如果未在控制器中启用用户权限，系统会提示您输入 FTP/Web 服务器的唯一用户名和密码。缺省用户名为 **USER**，缺省密码也是 **USER**。

**注意：** 您无法修改缺省用户名和密码。为保证 FTP/Web 服务器正常运行，您必须通过 **用户和组** 进行操作。

### 警告

#### 未经授权的数据访问

- 使用用户权限保护 FTP/Web 服务器的访问。
- 如果不启用用户权限，则禁用 FTP/Web 服务器以防止意外或未授权的访问应用程序中的数据。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

要更改密码，请转到设备编辑器的 **用户和组** 选项卡。有关详细信息，请参阅 **SoMachine** 编程指南。

**注意：** 如果要访问已启用户访问权限并且您没有其密码的控制器，唯一的方法就是执行更新固件操作。更新控制器固件时，只能通过 **SD 卡** 或 **USB 密钥**（视您特定控制器的支持情况而定）清除用户权限。此外，您可以通过运行脚本清除控制器中的用户权限（有关更多信息，请参阅 **SoMachine** 编程指南）。这样可以从控制器存储器中有效删除现有应用程序，并恢复访问控制器的能力。

### 访问主页

要访问网站主页，请在导航器中键入控制器的 IP 地址。

下图显示了 Web 服务器站点的登录页面：



下图显示了登录后的 Web 服务器站点主页：



**注意：** Schneider Electric 在开发和实现控制系统时遵循并向其客户建议行业最佳实践。此建议包括“深度防护”方法，以保护工业控制系统的安全。此方法将控制器置于一个或多个防火墙后面，以仅限授权的人员和协议访问。

## 警告

### 未经授权访问及其导致的未经授权的机器操作

- 评估环境或机器是否已连接到关键基础结构，如果已连接，请在将自动化系统连接到任何网络之前，基于深度防护采取适当的预防措施。
- 将连接到网络的设备数限制为所需的最小数量。
- 将工业网络与公司内部的其他网络隔离。
- 使用防火墙、VPN 或其他经证实的安全措施，防止意外访问任何网络。
- 监控系统内的活动。
- 防止未经授权方或未经身份验证的操作直接访问或直接链接主体设备。
- 准备恢复计划，包括系统和过程信息的备份。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

## 监视：IO 查看器子菜单

使用 **IO 查看器** 可以显示和修改当前 I/O 值：

TM241CE40T\_U

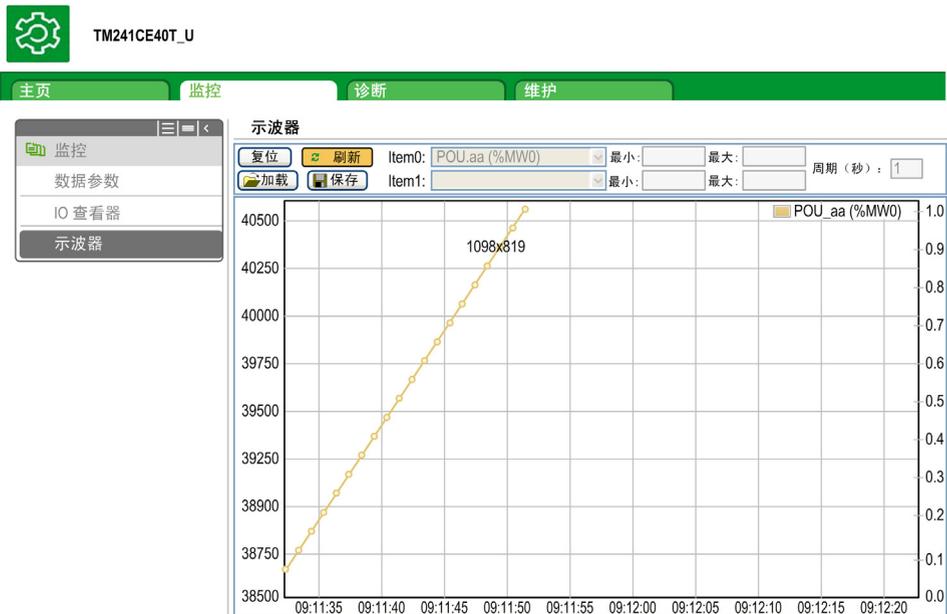
IO 查看器

映射	地址	类型	格式	值
ixDI_I0	%IX0.0	BOOL	布尔值	false
ixDI_I1	%IX0.1	BOOL	布尔值	false
ixDI_I2	%IX0.2	BOOL	布尔值	false
ixDI_I3	%IX0.3	BOOL	布尔值	false
ixDI_I4	%IX0.4	BOOL	布尔值	false
ixDI_I5	%IX0.5	BOOL	布尔值	false
ixDI_I6	%IX0.6	BOOL	布尔值	false
ixDI_I7	%IX0.7	BOOL	布尔值	false
ixDI_I8	%IX1.0	BOOL	布尔值	false
ixDI_I9	%IX1.1	BOOL	布尔值	false
ixDI_I10	%IX1.2	BOOL	布尔值	false
ixDI_I11	%IX1.3	BOOL	布尔值	false
ixDI_I12	%IX1.4	BOOL	布尔值	false
ixDI_I13	%IX1.5	BOOL	布尔值	false
ixDI_I14	%IX1.6	BOOL	布尔值	false
ixDI_I15	%IX1.7	BOOL	布尔值	false
ixDI_I16	%IX2.0	BOOL	布尔值	false
ixDI_I17	%IX2.1	BOOL	布尔值	false
ixDI_I18	%IX2.2	BOOL	布尔值	false
ixDI_I19	%IX2.3	BOOL	布尔值	false

元素	描述
刷新	启用 I/O 刷新： <ul style="list-style-type: none"> <li>● 灰色按钮：刷新已禁用</li> <li>● 橙色按钮：刷新已启用</li> </ul>
1000 毫秒	I/O 刷新周期（毫秒）
<<	转到上一个 I/O 列表页面
>>	转到下一个 I/O 列表页面

## 监视：示波器子菜单

示波器页面可以采用记录器时间图表的形式最多显示 2 个变量：



元素	描述
复位	擦除记忆
刷新	开始 / 停止刷新
加载	加载 Item0 和 Item1 的参数配置
保存	在控制器中保存 Item0 和 Item1 的参数配置
Item0	要显示的变量
Item1	要显示的变量
最小值	变量轴的最小值
最大值	变量轴的最大值
周期 (秒)	页面刷新周期 (秒)

**监视：数据参数****在 Web 服务器中监视变量**

要在 Web 服务器中监控变量，应将 **Web 数据配置** 对象添加到您的项目。在此对象中，您可以选择要监控的所有变量。

下表介绍了如何添加 **Web 数据配置** 对象：

步骤	操作
1	右键单击 <b>应用程序树</b> 选项卡中的 <b>应用程序</b> 节点。
2	单击 <b>添加对象</b> → <b>Web 数据配置 ...</b> 。 <b>结果：</b> 将会显示添加 <b>Web 数据配置</b> 窗口。
3	单击 <b>添加</b> 。 <b>结果：</b> 创建 <b>Web 数据配置</b> 对象并打开 <b>Web 数据配置</b> 编辑器。 <b>注意：</b> 由于 <b>Web 数据配置</b> 对象对于一个控制器来说是唯一的，因此它的名称不能更改。

**Web 数据配置编辑器**

单击 **刷新** 按钮可以选择变量，此操作将显示应用程序中定义的所有变量。



在 Web 服务器中选择要监视的变量：

WebDataConfiguration [MyController: PLC Logic: 应用程序] ×

刷新

符号	类型	注释
<input checked="" type="checkbox"/> ioConfig_Globals_Mapping		
<input checked="" type="checkbox"/> ixDI_I0 (%IX0.0)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I1 (%IX0.1)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I2 (%IX0.2)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I3 (%IX0.3)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I4 (%IX0.4)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I5 (%IX0.5)	Bool	DI: 快速输入, 漏极/源极
<input checked="" type="checkbox"/> ixDI_I6 (%IX0.6)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I7 (%IX0.7)	Bool	DI: 快速输入, 漏极/源极
<input type="checkbox"/> ixDI_I8 (%IX1.0)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I9 (%IX1.1)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I10 (%IX1.2)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I11 (%IX1.3)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I12 (%IX1.4)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I13 (%IX1.5)	Bool	DI: 常规输入, 漏极/源极
<input type="checkbox"/> ixDI_I0_1 (%IX2.0)	Bool	DI: 检测到短路 (如果为 True)
<input type="checkbox"/> qxDQ_Q0 (%QX0.0)	Bool	DQ: 快速输出, 推/拉
<input type="checkbox"/> qxDQ_Q1 (%QX0.1)	Bool	DQ: 快速输出, 推/拉
<input type="checkbox"/> qxDQ_Q2 (%QX0.2)	Bool	DQ: 快速输出, 推/拉
<input checked="" type="checkbox"/> qxDQ_Q3 (%QX0.3)	Bool	DQ: 快速输出, 推/拉
<input type="checkbox"/> qxDQ_Q4 (%QX0.4)	Bool	DQ: 常规输出
<input type="checkbox"/> qxDQ_Q5 (%QX0.5)	Bool	DQ: 常规输出
<input type="checkbox"/> qxDQ_Q6 (%QX0.6)	Bool	DQ: 常规输出
<input type="checkbox"/> qxDQ_Q7 (%QX0.7)	Bool	DQ: 常规输出
<input type="checkbox"/> qxDQ_Q8 (%QX1.0)	Bool	DQ: 常规输出
<input checked="" type="checkbox"/> qxDQ_Q9 (%QX1.1)	Bool	DQ: 常规输出
<input type="checkbox"/> qxDQ_Q0_1 (%QX2.0)	Bool	DQ: 重置命令 (在上升沿)
<input type="checkbox"/> qxModule_2_Q0 (%QX4.0)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q1 (%QX4.1)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q2 (%QX4.2)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q3 (%QX4.3)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q4 (%QX4.4)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q5 (%QX4.5)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q6 (%QX4.6)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q7 (%QX4.7)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q8 (%QX5.0)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q9 (%QX5.1)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q10 (%QX5.2)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q11 (%QX5.3)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q12 (%QX5.4)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q13 (%QX5.5)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q14 (%QX5.6)	Bool	Module_2:
<input type="checkbox"/> qxModule_2_Q15 (%QX5.7)	Bool	Module_2:
<input checked="" type="checkbox"/> GVL		
<input checked="" type="checkbox"/> 计数	Int	

**注意：** 变量选择只能在离线模式下完成。

## 监视：数据参数子菜单

**数据参数**页面可让您创建和监视变量的一些列表。您可以创建变量的多个列表（最多 10 个列表），每个列表包含控制器应用程序的多个变量（每个列表最多 20 个变量）。

每个列表有一个名称和一个刷新周期。这些列表保存在控制器的闪存中，因此可以从访问此控制器的任何 Web 客户端应用程序中访问（加载、修改、保存）创建的列表。

通过**数据参数**可以显示和修改变量值：



元素	描述
加载	将已保存的列表从控制器内部闪存加载到 Web 服务器页面
保存	在控制器中保存选择的列表描述（/usr/web 目录）
添加	添加列表描述或变量
删除	删除列表描述或变量
刷新周期	列表描述中包含的变量的刷新周期（毫秒）
刷新	启用 I/O 刷新： <ul style="list-style-type: none"> <li>● 灰色按钮：刷新已禁用</li> <li>● 橙色按钮：刷新已启用</li> </ul>

**注意：**IEC 对象（%IW、%M 等）不可直接访问。要访问 IEC 对象，必须首先在定位寄存器中将它们的内容进行分组（请参阅重新定位表（参见第 34 页））。

## 诊断：以太网子菜单

下图显示了远程 Ping 服务：

The screenshot shows the web interface for the TM241CE40T\_U device. The top navigation bar includes '主页' (Home), '监控' (Monitoring), '诊断' (Diagnosis), and '维护' (Maintenance). The left sidebar menu is expanded to show '以太网' (Ethernet) selected. The main content area is titled '以太网' and contains the following sections:

**远程 Ping 服务**

输入用于从控制器进行 ping 的 IP 地址：

**统计信息**

Ethernet_1		Modbus 统计信息	
MAC 地址	0.80.F4.A.9.7D	传输的消息正常	0
IP 地址	85.15.1.51	接收的消息正常	0
子网掩码	255.0.0.0	错误消息	0
网关地址	0.0.0.0	IpMaster 连接状态	未连接 (1)
状态	链路打开 (1)	IpMaster 超时事件计数器	0
以太网统计信息			
已打开的上层连接	3		
成功传输的帧	552		
成功接收的帧	1288		
传输的缓冲不正常	0		
接收的缓冲不正常	0		
以太网 IP 统计信息			
传输的 IO 消息	0		

## 维护选项卡

通过“维护”页面可以访问控制器闪存(参见第 37 页)的 /usr 和 /sys 文件夹:

/usr 的索引:



/sys 的索引:



## 维护：后配置子菜单

通过**后配置**页面可以更新控制器上保存的后配置文件 ( 参见第 185 页 ):



步骤	操作
1	单击 <b>加载</b> 。
2	修改参数 ( 参见第 188 页 )。
3	单击 <b>保存</b> 。 <b>注意：</b> 下一次后配置文件读取 ( 参见第 186 页 ) 时将考虑这些新参数。

**维护：EIP 配置文件子菜单**

只有在控制器上配置了以太网 IP 服务时才会出现文件树。

/usr 的索引：



文件	描述
My Machine Controller.gz	GZIP 文件
My Machine Controller.ico	图标文件
My Machine Controller.eds	电子数据表文件

## FTP 服务器

### 简介

安装在连接到控制器（通过以太网端口）但未安装 SoMachine 的计算机上的任何 FTP 客户端，均可用来在控制器的数据存储区域传输和接收文件。

**注意：** Schneider Electric 在开发和实现控制系统时遵循并向其客户建议行业最佳实践。此建议包括“深度防护”方法，以保护工业控制系统的安全。此方法将控制器置于一个或多个防火墙后面，以仅限授权的人员和协议访问。

### 警告

#### 未经授权访问及其导致的未经授权的机器操作

- 评估环境或机器是否已连接到关键基础结构，如果已连接，请在将自动化系统连接到任何网络之前，基于深度防护采取适当的预防措施。
- 将连接到网络的设备数限制为所需的最小数量。
- 将工业网络与公司内部的其他网络隔离。
- 使用防火墙、VPN 或其他经证实的安全措施，防止意外访问任何网络。
- 监控系统内的活动。
- 防止未经授权方或未经身份验证的操作直接访问或直接链接主体设备。
- 准备恢复计划，包括系统和过程信息的备份。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

**注意：** 在当前登录的目标设备的在线用户管理中使用安全相关命令 ( 参见 *SoMachine, 编程指南* ), 这些命令提供了用于添加、编辑和删除用户的方法。

即使控制器为空（未启用用户应用程序和用户权限），也可使用 FTP 服务器。

## FTP 访问

如果在控制器中启用了用户权限，则通过用户权限控制对 FTP 服务器的访问。有关详细信息，请参阅**用户和组**选项卡描述（参见第 73 页）。

如果未在控制器中启用用户权限，系统会提示您输入 FTP/Web 服务器的唯一用户名和密码。缺省用户名为 **USER**，缺省密码也是 **USER**。

**注意：**您无法修改缺省用户名和密码。为保证 FTP/Web 服务器正常运行，您必须通过**用户和组**进行操作。

### 警告

#### 未经授权的数据访问

- 使用用户权限保护 FTP/Web 服务器的访问。
- 如果不启用用户权限，则禁用 FTP/Web 服务器以防止意外或未授权的访问应用程序中的数据。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

要更改密码，请转到设备编辑器的**用户和组**选项卡。有关详细信息，请参阅 SoMachine 编程指南。

**注意：**如果要访问已启用户访问权限并且您没有其密码的控制器，唯一的方法就是执行更新固件操作。更新控制器固件时，只能通过 SD 卡或 USB 密钥（视您特定控制器的支持情况而定）清除用户权限。此外，您可以通过运行脚本清除控制器中的用户权限（有关更多信息，请参阅 SoMachine 编程指南）。这样可以从控制器存储器中有效删除现有应用程序，并恢复访问控制器的能力。

## 文件访问

请参阅文件结构（参见第 31 页）。

## SNMP

### 简介

SNMP 协议 (Simple Network Management Protocol) 用于提供管理网络所需的数据和服务。

数据存储在 MIB (管理信息库) 中。SNMP 协议用于读取或写入 MIB 数据。执行以太网 SNMP 服务是最基本要求, 因为它只处理必要的对象。

M241 控制器支持标准 MIB-2 对象。

### SNMP 对象处理

对象	描述	访问	缺省值
sysDescr	设备的文本描述	读取	SCHNEIDER M241-51 Fast Ethernet TCP/IP
sysName	节点管理名称	读 / 写	控制器参考号

写入的值通过 SNMP 客户端工具软件保存到控制器中。用于此操作的 Schneider Electric 软件是 ConneXview。ConneXview 不随控制器提供。有关详细信息, 请参阅 [www.schneider-electric.com](http://www.schneider-electric.com)。

这些字符串的大小限制为 50 个字符。

## 节 13.2

### 防火墙配置

---

#### 简介

本节介绍如何配置 Modicon M241 Logic Controller 的防火墙。

#### 本节包含了哪些内容?

本节包含了以下主题:

主题	页
简介	129
动态更改过程	131
防火墙行为	132
脚本文件语法	134

## 简介

### 防火墙介绍

一般情况下，防火墙通过拦截未授权的访问和允许授权的访问帮助来保护网络安全区周围。防火墙指的是一台设备或一套设备，基于一套规则和其他标准将其配置为允许、拒绝、加密或代理不同安全区之间的流量。

流程控制设备和高速生产机器要求快速的数据吞吐量，并且经常不能容忍控制网络中进攻性安全策略所引入的延迟。因此，通过网络周围提供保护，防火墙在安全策略中扮演了重要的角色。防火墙是整个系统级别策略的重要组成部分。

**注意：** Schneider Electric 在开发和实现控制系统时遵循并向其客户建议行业最佳实践。此建议包括“深度防护”方法，以保护工业控制系统的安全。此方法将控制器置于一个或多个防火墙后面，以仅限授权的人员和协议访问。

### 警告

#### 未经授权访问及其导致的未经授权的机器操作

- 评估环境或机器是否已连接到关键基础结构，如果已连接，请在将自动化系统连接到任何网络之前，基于深度防护采取适当的预防措施。
- 将连接到网络的设备数限制为所需的最小数量。
- 将工业网络与公司内部的其他网络隔离。
- 使用防火墙、VPN 或其他经证实的安全措施，防止意外访问任何网络。
- 监控系统内的活动。
- 防止未经授权方或未经身份验证的操作直接访问或直接链接主体设备。
- 准备恢复计划，包括系统和过程信息的备份。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

### 防火墙配置

可通过以下 3 种方式管理控制器防火墙配置：

- 静态配置，
- 动态更改，
- 应用程序设置。

在静态配置中使用脚本文件，以及使用它进行动态更改。

### 静态配置

在控制器启动时加载静态配置。

可通过管理位于控制器中的缺省脚本文件来静态配置控制器防火墙。此文件的路径是 `/Usr/Cfg/FirewallDefault.cmd`。

### 动态更改

在控制器启动后，可通过使用脚本文件来更改控制器防火墙配置。

可通过以下 2 种方法来加载这些动态更改：

- 使用物理 SD 卡 ( 参见第 131 页 )，
- 使用应用程序中的功能块 ( 参见第 131 页 )。

### 应用程序设置

请参阅以太网配置 ( 参见第 107 页 )。

## 动态更改过程

### 使用 SD 卡

下表描述了从 SD 卡执行脚本文件的过程：

步骤	操作
1	创建有效的脚本文件 ( 参见第 134 页 )。 例如，将脚本文件命名为 <i>FirewallMaintenance.cmd</i> 。
2	加载 SD 卡上的脚本文件。 例如，加载 <i>Usr/cfg</i> 文件夹中的脚本文件。
3	在文件 <i>Sys/Cmd/Script.cmd</i> 添加一行代码，包含命令 <code>Firewall_install "pathname/FileName"</code> 。 例如，代码行是 <code>Firewall_install "/sd0/Usr/cfg/FirewallMaintenance.cmd"</code>
4	在控制器上插入 SD 卡。

### 使用应用程序中的功能块

下表描述了从应用程序中执行脚本文件的过程：

步骤	操作
1	创建有效的脚本文件 ( 参见第 134 页 )。 例如，将脚本文件命名为 <i>FirewallMaintenance.cmd</i> 。
2	在控制器存储器中加载脚本文件。 例如，使用 FTP 加载 <i>Usr/Syslog</i> 文件夹中的脚本文件。
3	使用 <code>ExecuteScript</code> ( 参见 <i>Modicon M241 Logic Controller, 系统功能和变量, PLCSystem 库指南</i> ) 功能块。 例如，[SCmd] 输入为 <code>'Firewall_install "/usr/Syslog/FirewallMaintenance.cmd"'</code>

## 防火墙行为

### 简介

防火墙配置取决于在控制器上执行的操作和初始配置状态。有以下 5 种可能的初始状态：

- 在控制器中没有缺省脚本文件。
- 存在正确脚本文件。
- 存在不正确的脚本文件。
- 不存在缺省脚本文件，并且应用程序已配置防火墙。
- 已执行动态脚本文件配置。

### 无缺省脚本文件

如果 ...	则 ...
启动控制器	不配置防火墙。不激活任何保护。
执行动态脚本文件	根据动态脚本文件配置防火墙。
执行不正确的动态脚本文件	不配置防火墙。不激活任何保护。
下载应用程序	根据应用程序设置配置防火墙。

### 存在缺省脚本文件

如果 ...	则 ...
启动控制器	根据缺省脚本文件配置防火墙。
执行动态脚本文件	删除缺省脚本文件的整个配置。 根据动态脚本文件配置防火墙。
执行不正确的动态脚本文件	根据缺省脚本文件配置防火墙。 不考虑动态脚本文件。
下载应用程序	忽略应用程序的整个配置。 根据缺省脚本文件配置防火墙。

### 存在不正确的缺省脚本文件

如果 ...	则 ...
启动控制器	不配置防火墙。不激活任何保护
执行动态脚本文件	根据动态脚本文件配置防火墙。
执行不正确的动态脚本文件	不配置防火墙。不激活任何保护。
下载应用程序	根据应用程序设置配置防火墙。

### 无缺省脚本文件的应用程序设置

如果 ...	则 ...
启动控制器	根据应用程序设置配置防火墙。
执行动态脚本文件	删除应用程序设置的整个配置。 根据动态脚本文件配置防火墙。
执行不正确的动态脚本文件	根据应用程序设置配置防火墙。不考虑动态脚本文件。
下载应用程序	删除上一个应用程序的整个配置。 根据新应用程序设置配置防火墙。

### 执行已执行的动态脚本文件

如果 ...	则 ...
启动控制器	根据动态脚本文件配置来配置防火墙（请参见注释）。
执行动态脚本文件	删除上一个动态脚本文件的整个配置。 根据新动态脚本文件配置防火墙。
执行不正确的动态脚本文件	根据上一个动态脚本文件配置来配置防火墙。不考虑不正确的动态脚本文件。
下载应用程序	忽略应用程序的整个配置 根据动态脚本文件配置防火墙。
<b>注意：</b> 如果将包含网络安全脚本的 SD 卡插入到控制器中，将阻止启动。首先拔出 SD 卡才能正确启动控制器。	

## 脚本文件语法

### 概述

本节介绍如何编写脚本文件（缺省脚本文件或动态脚本文件），以便在启动控制器期间或用户触发的特定命令期间正确执行脚本文件。

### 一般编写准则

脚本中的每一行命令都以 ";" 结束。

如果某行以 ";" 开头，表明该行是注释行。

脚本文件中的最大行数为 50。

语法不区分大小写。

如果在脚本文件中不遵守该语法，则根本不会执行该脚本文件。这表示防火墙配置将保持上一个状态。

**注意：**如果未执行脚本文件，则会将该问题记录到日志文件中。此日志文件在控制器中的位置是 `/usr/Syslog/FWLog.txt`。

### 防火墙一般命令

命令	描述
FireWall enable	阻止来自 <b>Ethernet</b> 接口的所有帧。如果未进一步授权任何 IP 地址，将无法在 <b>Ethernet</b> 接口上进行任何通讯。 <b>注意：</b> 缺省情况下，在启用防火墙时，将拒绝所有帧。
FireWall Disable	允许所有 IP 地址在所有 <b>Ethernet</b> 接口上访问控制器。
FireWall Eth1 Default Enable	控制器接受所有帧。
FireWall Eth1 Default Reject	控制器拒绝所有帧。 <b>注意：</b> 缺省情况下，如果不存在此行，将相当于命令 <code>FireWall Eth1 Default Reject</code> 。
<b>注意：</b> 在脚本文件中编写的行数不得超过 50。	

## 防火墙特定命令

命令	范围	描述
Firewall Eth1 Allow IP ●●●●●	● = 0...255	在所有端口号和端口类型上允许来自所述 IP 地址的所有帧。
Firewall Eth1 Reject IP ●●●●●	● = 0...255	在所有端口号和端口类型上拒绝来自所述 IP 地址的所有帧。
Firewall Eth1 Allow IPs ●●●●● to ●●●●●	● = 0...255	所有端口号和端口类型都允许来自所述范围中的 IP 地址的所有帧。
Firewall Eth1 Reject IPs ●●●●● to ●●●●●	● = 0...255	所有端口号和端口类型都拒绝来自所述范围中的 IP 地址的所有帧。
Firewall Eth1 Allow port_type port Y	Y = (目标端口号 (参见第 137 页))	允许带有所述目标端口号的所有帧。
Firewall Eth1 Reject port_type port Y	Y = (目标端口号 (参见第 137 页))	允许带有所述目标端口号的所有帧。
Firewall Eth1 Allow port_type ports Y1 to Y2	Y = (目标端口号 (参见第 137 页))	允许带有所述范围中的目标端口号的所有帧。
Firewall Eth1 Reject port_type ports Y1 to Y2	Y = (目标端口号 (参见第 137 页))	拒绝带有所述范围中的目标端口号的所有帧。
Firewall Eth1 Allow IP ●●●●● on port_type port Y	● = 0...255 Y = (目标端口号 (参见第 137 页))	允许来自所述 IP 地址并带有所述目标端口号的所有帧。
Firewall Eth1 Reject IP ●●●●● on port_type port Y	● = 0...255 Y = (目标端口号 (参见第 137 页))	拒绝来自所述 IP 地址并带有所述目标端口号的所有帧。
Firewall Eth1 Allow IP ●●●●● on port_type ports Y1 to Y2	● = 0...255 Y = (目标端口号 (参见第 137 页))	允许来自所述 IP 地址并带有所述范围中的目标端口号的所有帧。
Firewall Eth1 Reject IP ●●●●● on port_type ports Y1 to Y2	● = 0...255 Y = (目标端口号 (参见第 137 页))	拒绝来自所述 IP 地址并带有所述范围中的目标端口号的所有帧。

命令	范围	描述
Firewall Eth1 Allow IPs ●1.●1.●1.●1 to ●2.●2.●2.●2 on port_type port Y	● = 0...255 Y = (目标端口号 (参见第 137 页))	拒绝来自所述范围中的 IP 地址并带有所述目标端口号的所有帧。
Firewall Eth1 Reject IPs ●1.●1.●1.●1 to ●2.●2.●2.●2 on port_type port Y	● = 0...255 Y = (目标端口号 (参见第 137 页))	拒绝来自所述范围中的 IP 地址并带有所述目标端口号的所有帧。
Firewall Eth1 Allow IPs ●1.●1.●1.●1 to ●2.●2.●2.●2 on port_type ports Y1 to Y2	● = 0...255 Y = (目标端口号 (参见第 137 页))	允许来自所述范围中的 IP 地址并带有所述范围中的目标端口号的所有帧。
Firewall Eth1 Reject IPs ●1.●1.●1.●1 to ●2.●2.●2.●2 on port_type ports Y1 to Y2	● = 0...255 Y = (目标端口号 (参见第 137 页))	拒绝来自所述范围中的 IP 地址并带有所述范围中的目标端口号的所有帧。
Firewall Eth1 Allow MAC●:●:●:●:●:●:●	● = 0...F	允许来自所述 MAC 地址 ●●:●●:●●:●●:●●:●● 的所有帧。
Firewall Eth1 Reject MAC●:●:●:●:●:●:●	● = 0...F	拒绝带有所述 MAC 地址 ●●:●●:●●:●●:●●:●● 的所有帧。

### 脚本文件示例

```

; Enable firewall on Ethernet 1. All frames are rejected;
FireWall Eth1 Enable;

; Block all Modbus Requests on all IP address
Firewall Eth1 Reject tcp port 502;

; Allow FTP active connection for IP address 85.16.0.17
Firewall Eth1 Allow IP 85.16.0.17 on tcp port 20 to 21;

```

## 使用的端口列表

协议	目标端口号
SoMachine	UDP 1740、1741、1742、1743 TCP 1105
FTP	TCP 21、20
HTTP	TCP 80
Modbus	TCP 502
Discovery	UDP 27126、27127
SNMP	UDP 161、162
NVL	UDP 缺省值：1202
Ethernet/IP	UDP 2222 TCP 44818

## 节 13.3

### 以太网可选设备

---

#### 本节包含了哪些内容？

本节包含了以下主题：

主题	页
以太网管理器	139
Ethernet/IP 设备	140
Modbus TCP 从站设备	159

## 以太网管理器

### 添加以太网管理器

控制器支持以下以太网管理器：

- EthernetIP（用于 CIP 设备）
- ModbusTCP Slave Device

要向控制器中添加以太网管理器，请在**硬件目录**中选择：

- 对于 EthernetIP：**EthernetIP**
- 对于 ModbusTCP：**ModbusTCP 从站设备**

将其拖到**设备树**并将其放到一个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法（参见 *SoMachine, 编程指南*）
- 使用上下文菜单或加号按钮（参见 *SoMachine, 编程指南*）

## Ethernet/IP 设备

### 简介

本节介绍 Ethernet/IP 设备 (CIP) 与控制器的配置。

有关 Ethernet/IP (CIP) 的详细信息，请参阅 [www.odva.org](http://www.odva.org) website。

### 添加 Ethernet/IP 设备

请参阅添加以太网管理器 ( 参见第 139 页 )。

### Ethernet/IP 设备配置

要配置 Ethernet/IP 设备参数，请双击**设备树**中的**以太网** → **EthernetIP**。

屏幕将显示以下对话框：



**EthernetIP** 配置参数定义如下：

- **实例**  
区对应的编号。
- **大小**：  
区通道数。  
每个通道的内存大小为 2 个字节，用于存储 %IWx 或 %QWx 对象的值，其中 x 是通道号。  
例如，如果**输出区**的大小为 20，这表示有 20 个输入通道 (IW0...IW19) 用于寻址 %IWy...%IW(y+20-1)，其中 y 是该输出区的第一个可用通道。

元素		允许的控制范围	SoMachine 缺省值
输出区	实例	150...189	150
	大小	2...40	20
输入区	实例	100...149	100
	大小	2...40	20

## EDS 文件生成

根据上述参数，在下载应用程序时或启动（如果存在引导应用程序）时，会在控制器的“/usr/Eip”目录中自动生成 EDS 文件。

**注意：**如果以太网网络在控制器上正常工作（电缆已连接且获取了 IP 地址），则会生成 EDS 文件。

## EthernetIP Slave I/O Mapping 选项卡

可以在 **EthernetIP Slave I/O Mapping** 选项卡中定义和命名变量。此选项卡还提供其他信息，例如拓扑寻址。

EthernetIP							
EthernetIP Slave I/O Mapping							
信息							
通道							
变量	映射	通道	地址	类型	缺省值	单位	描述
输入							输入
		IW0	%IW9	WORD			
		位 0	%IX18.0	BOOL	FALSE		
		位 1	%IX18.1	BOOL	FALSE		
		位 2	%IX18.2	BOOL	FALSE		
		位 3	%IX18.3	BOOL	FALSE		
		位 4	%IX18.4	BOOL	FALSE		
		位 5	%IX18.5	BOOL	FALSE		
		位 6	%IX18.6	BOOL	FALSE		
		位 7	%IX18.7	BOOL	FALSE		
		位 8	%IX19.0	BOOL	FALSE		
		位 9	%IX19.1	BOOL	FALSE		
		位 10	%IX19.2	BOOL	FALSE		
		位 11	%IX19.3	BOOL	FALSE		
		位 12	%IX19.4	BOOL	FALSE		
		位 13	%IX19.5	BOOL	FALSE		
		位 14	%IX19.6	BOOL	FALSE		
		位 15	%IX19.7	BOOL	FALSE		
		IW1	%IW10	WORD			
输出							输出
		QW0	%QW3	WORD			
		QW1	%QW4	WORD			
		QW2	%QW5	WORD			
		QW3	%QW6	WORD			
		QW4	%QW7	WORD			

下表描述了 Ethernet/IP 从站 I/O 映射配置：

通道		类型	缺省值	描述
输入	IW0	WORD	-	控制器输出的命令字 (%QW)
	IWxxx			
输出	QW0	WORD	-	控制器输入的状态 (%IW)
	QWxxx			

字数取决于在 Ethernet/IP 设备配置 ( 参见第 140 页 ) 中配置的大小参数。

输出表示来自主控制器的 OUTPUT (= 控制器的 %IW)。

输入表示来自主控制器的 INPUT (= 控制器的 %QW)。

### Ethernet/IP 上的连接

要访问从站，必须打开连接 ( Ethernet/IP 协议级别所使用的全局名称 )，该连接可包含若干发送请求的会话。

一个显式连接使用一个会话 ( 会话为 TCP 或 UDP 连接 )。

一个 I/O 连接使用 2 个会话。

下表显示了 Ethernet/IP 连接限制：

特性	描述
最大显式连接数	8 ( 3 类 )
最大 I/O 连接数	1 ( 1 类 )
最大连接数	8
最大会话数	16
最大同时请求数	32

## 配置文件

控制器支持以下对象：

对象类	类 ID	类别	实例数	对接口行为的影响
标识对象 (参见第 143 页)	01h	1	1	支持复位服务
消息路由器对象 (参见第 146 页)	02h	1	1	显式消息连接
组件对象 (参见第 149 页)	04h	2	2	定义 I/O 数据格式
连接管理器对象 (参见第 151 页)	06h		1	-
文件对象 (参见第 153 页)	37h		2	可用于交换 EDS 文件
Modbus 对象 (参见第 155 页)	44h		1	-
TCP/IP 接口对象 (参见第 156 页)	F5h	1	1	TCP/IP 配置
以太网链接对象 (参见第 158 页)	F6h	1	1	计数器和状态信息

## 标识对象

下表描述了标识对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	01h	标识对象的实现修订号
2	获取	最大实例数	UINT	01h	最大实例数
3	获取	实例数	UINT	01h	对象实例数
4	获取	可选实例属性列表	UINT、 UINT []	00h	前 2 个字节包含可选实例属性数。随后的每个字节对表示其他可选实例属性数。
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	07h	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

下表对实例服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
05h	复位 <sup>(1)</sup>	初始化 Ethernet/IP 组件（控制器重新启动）
0Eh	获取单个属性	返回指定属性的值

(1) 复位服务描述：

当标识对象收到复位请求时，它将执行以下操作：

- 判断自己是否可以提供所请求的复位类型
- 对请求做出响应
- 尝试执行请求的复位类型

复位公共服务有一个特定参数：复位类型 (USINT)，它具有以下值：

值	复位类型
0	尽可能相似地模拟重新通电。仿真重新启动命令。 <b>注意：</b> 如果省略此参数，则该值将成为缺省值。
1	尽可能相似地模拟拆卸和重新接通控制器电源，然后将 I/O 恢复为初始值。
2	尽可能返回到初始配置（通讯链路参数除外），越接近越好，然后尽可能相似地模拟重新通电。要保存的通讯链路参数由各个网络类型定义。有关详细信息，请参阅网络特定链接对象的复位服务。仿真初始值复位命令。
3...99	保留
100...199	供应商特定
200...255	保留

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	供应商 ID	UINT	243h	Schneider 自动化 ID
2	获取	设备类型	UINT	0Eh	PLC
3	获取	产品代码	UINT	-	控制器产品代码
4	获取	修订号	USINT、 USINT 的结构	-	控制器的产品修订号 <sup>(1)</sup> 相当于控制器版本的 2 个低字节
5	获取	状态	WORD <sup>(1)</sup>	-	请参阅下表中的定义
6	获取	序列号	UDINT	-	控制器的序列号 XX + MAC 地址上的 3 LSB
7	获取	产品名	USINT、 STRING 的结 构	-	-

(1)WORD 中的映射:

- MSB: 次修订号 (第二个 USINT)
- LSB: 主修订号 (第一个 USINT)

示例: 0205h 表示修订号 V5.2。

状态描述 (属性 5):

位	名称	描述
0	已被拥有	未使用
1	保留	-
2	已配置	TRUE 表示设备应用程序已进行重新配置。
3	保留	-
4...7	扩展设备状态	<ul style="list-style-type: none"> <li>● 0: 自检或未知</li> <li>● 1: 正在进行固件更新</li> <li>● 2: 至少检测到一个无效 I/O 连接错误</li> <li>● 3: 未建立任何 I/O 连接</li> <li>● 4: 非易失性配置无效</li> <li>● 5: 检测到不可恢复错误</li> <li>● 6: 至少有一个 I/O 连接处于运行状态</li> <li>● 7: 至少已建立一个 I/O 连接, 所有连接都处于空闲模式</li> <li>● 8: 保留</li> <li>● 9...15: 未使用</li> </ul>
8	轻微可恢复故障	TRUE 指示设备检测到认为是可恢复的错误。 此类型事件不会引起设备状态的改变。
9	轻微不可恢复故障	TRUE 指示设备检测到认为是不可恢复的错误。 此类型事件不会引起设备状态的改变。
10	重大可恢复故障	TRUE 表示设备检测到错误, 该错误需要设备报告例外并进入 HALT 状态。
11	重大不可恢复故障	TRUE 表示设备检测到错误, 该错误需要设备报告例外并进入 HALT 状态。
12...15	保留	-

## 消息路由器对象

下表描述了消息路由器对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	01h	消息路由器对象的实现修订号
2	获取	最大实例数	UINT	01h	最大实例数
3	获取	实例数	UINT	01h	对象实例数
4	获取	可选实例属性列表	UINT、 UINT [] 的结构	20	前 2 个字节包含可选实例属性数。随后的每个字节对表示其他可选实例属性数（从 100 到 119）。
5	获取	可选服务列表	UINT	00h	所有已实现可选服务属性数和列表（0：不支持任何可选服务）
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	119	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

下表对实例服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	已实现对象列表	UINT、 UINT [] 的 结构	-	已实现对象列表。前 2 个字节包含已实现对象数。随后的每两个字节都表示其他已实现类数。 此列表包含以下对象： <ul style="list-style-type: none"> <li>● 标识</li> <li>● 消息路由器</li> <li>● 组件</li> <li>● 连接管理器</li> <li>● 参数</li> <li>● 文件对象</li> <li>● Modbus</li> <li>● 端口</li> <li>● TCP/IP</li> <li>● 以太网链路</li> </ul>
2	获取	可用数	UINT	20h	支持的最大并发 CIP（1 类或 3 类）连接数
100	获取	在最后一秒内接收的 1 类传入数据包总数	UINT	-	在最后一秒内所有隐式（1 类）连接接收的传入数据包总数
101	获取	在最后一秒内发送的 1 类传出数据包总数	UINT	-	在最后一秒内所有隐式（1 类）连接发送的传出数据包总数
102	获取	在最后一秒内接收的 3 类传入数据包总数	UINT	-	在最后一秒内所有显式（3 类）连接接收的传入数据包总数
103	获取	在最后一秒内发送的 3 类传出数据包总数	UDINT	-	在最后一秒内所有显式（3 类）连接发送的传出数据包总数
104	获取	在最后一秒内接收的未连接传入数据包总数	UINT	-	在最后一秒内接收的未连接传入数据包总数
105	获取	在最后一秒内发送的未连接传出数据包总数	UINT	-	在最后一秒内发送的未连接传出数据包总数
106	获取	在最后一秒内接收的 Ethernet/IP 传入数据包总数	UINT	-	在最后一秒内接收的未连接 1 类或 3 类数据包总数
107	获取	在最后一秒内发送的 Ethernet/IP 传出数据包总数	UINT	-	在最后一秒内接收的未连接 1 类或 3 类数据包总数

属性 ID	访问	名称	数据类型	值	描述
108	获取	接收的 1 类传入数据包总计	UINT	-	所有隐式（1 类）连接接收的传入数据包总数。
109	获取	发送的 1 类传出数据包总数	UINT	-	所有隐式（1 类）连接发送的传出数据包总数
110	获取	接收的 3 类传入数据包总数	UINT	-	所有显式（3 类）连接接收的传入数据包总数。此数量包括在检测到错误时返回的数据包（列在接下来的两行中）。
111	获取	包含无效参数值的 3 类传入数据包的总数	UINT	-	目标为不受支持的服务 / 类 / 实例 / 属性 / 成员的 3 类传入数据包的总数
112	获取	包含无效格式的 3 类传入数据包总数	UINT	-	包含无效格式的 3 类传入数据包的总数
113	获取	发送的 3 类传出数据包总数	UINT	-	所有显式（3 类）连接发送的数据包总数
114	获取	接收的未连接传入数据包总数	UINT	-	未连接的传入数据包的总数。此数量包括在检测到错误时返回的数据包（列在接下来的两行中）。
115	获取	包含无效参数值的未连接传入数据包总数	UINT	-	目标为不受支持的服务 / 类 / 实例 / 属性 / 成员的未连接传入数据包的总数
116	获取	包含无效格式的未连接传入数据包总数	UINT	-	包含无效格式的未连接传入数据包的总数
117	获取	发送的未连接传出数据包总数	UINT	-	发送的所有未连接数据包的总数
118	获取	传入 Ethernet/IP 数据包总数	UINT	-	接收的未连接类 1 或类 3 数据包总数
119	获取	传出 Ethernet/IP 数据包总数	UINT	-	发送的未连接类 1 或类 3 数据包总数

## 组件对象

下表描述了组件对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	2	组件对象的实现修订号
2	获取	最大实例数	UINT	189	最大实例数
3	获取	实例数	UINT	2	对象实例数
4	获取	可选实例属性列表	结构： UINT UINT []	1 4	前 2 个字节包含可选实例属性数。随后的每个字节对表示其他可选实例属性数。
5	获取	可选服务列表	UINT	00h	所有已实现可选服务属性数和列表（0：不支持任何可选服务）
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	04h	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
0Eh	获取单个属性	返回指定属性的值

下表对实例服务进行了描述：

服务代码	名称	描述
10h	获取单个属性	修改指定属性的值
0Eh	获取单个属性	返回指定属性的值
18h	获取成员	读取集合对象实例的成员
19h	设置成员	修改集合对象实例的成员

## 支持的实例

输出表示来自主控制器的 OUTPUT（= 控制器的 %IW）。

输入表示来自主控制器的 INPUT（= 控制器的 %QW）。

该控制器支持 2 个组件：

名称	实例	数据大小
本地控制器输出 (%IW)	可配置：必须介于 100 和 149 之间	2...40 个字
本地控制器输入 (%QW)	可配置：必须介于 150 和 189 之间	2...40 个字

**注意：** 组件对象将多个对象的属性绑定在一起，因此通过单个连接即可在各对象之间进行信息交换。组件对象是静态对象。

**注意：** 使用中的组件可通过访问网络配置工具 (RSNetWorx) 中的参数进行修改。控制器需要重新接通电源才能注册新的组件赋值。

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	成员对象列表数	UINT	2...40	控制器的成员数始终为 1
2	获取	成员列表	结构数组	-	1 个结构的数组，每个结构表示一个成员
3	Get/Set	实例数据	字节数组	-	只有本地控制器输出可使用数据集服务
4	获取	实例数据大小	UINT	4...80	数据大小（以字节为单位）

成员列表内容：

名称	数据类型	值	复位类型
成员数据大小	UINT	4...40	成员数据大小（以位为单位）
成员路径大小	UINT	6	EPATH 大小（请参见下表）
成员路径	EPATH	-	成员的 EPATH

EPATH 是：

字	值	语义
0	2004h	4 类
1	24xxh	实例 xx，其中 xx 指代实例值（示例：2464h = 实例 100）。
2	30h	属性 3

## 连接管理器对象

下表描述了组件对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	2	连接管理器对象的实现修订号
2	获取	最大实例数	UINT	189	最大实例数
3	获取	实例数	UINT	2	对象实例数
4	获取	可选实例属性列表	结构： UINT UINT []	-	可选属性数和列表。第一个字包含要跟随的属性数，接下来的每个字包含其他属性代码。 跟随的可选属性包括： <ul style="list-style-type: none"> <li>● 传入 connection open 请求总数</li> <li>● 因意外格式的 Forward Open 而遭到拒绝的请求数</li> <li>● 因资源不足而遭到拒绝的请求数</li> <li>● 因 Forward Open 发送的参数值而遭到拒绝的请求数</li> <li>● 接收的 Forward Close 请求数</li> <li>● 格式无效的 Forward Close 请求数</li> <li>● 无法与活动连接匹配的 Forward Close 请求数</li> <li>● 因另一端停止生成或网络断开连接而超时的连接数</li> </ul>
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	08h	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

下表对实例服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有实例属性的值
0Eh	获取单个属性	返回指定属性的值
4Eh	Forward Close	关闭现有连接
52h	未连接发送	发送多跳未连接请求
54h	Forward Open	打开新连接

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	Open 请求	UINT	-	收到的 Forward Open 服务请求数
2	获取	Open 格式拒绝	UINT	-	因格式无效而遭到拒绝的 Forward Open 服务请求数
3	获取	Open 资源拒绝	字节数组	-	因缺乏资源而遭到拒绝的 Forward Open 服务请求数
4	获取	Open 其他拒绝	UINT	-	因格式无效或缺乏资源之外的原因而遭到拒绝的 Forward Open 服务请求数
5	获取	Close 请求	UINT	-	收到的 Forward Close 服务请求数
6	获取	关闭格式请求	UINT	-	因格式无效而遭到拒绝的 Forward Close 服务请求数
7	获取	Close 其他请求	UINT	-	因格式无效之外的原因而遭到拒绝的 Forward Close 服务请求数
8	获取	连接超时	UINT	-	由此连接管理器控制的连接中已发生的总连接超时数

## 文件对象

下表描述了文件对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	1	文件对象的实现修订号
2	获取	最大实例数	UINT	C9h	最大实例数
3	获取	实例数	UINT	2	对象实例数
6	获取	最大类属性	UINT	20h	最大类属性值
7	获取	最大实例属性	UINT	0Bh	最大实例属性值
32	获取	实例列表	-	-	返回有关所有已配置实例的信息，包括实例数、实例名称和实例文件名

下表对类服务进行了描述：

服务代码	名称	描述
0Eh	获取单个属性	返回指定属性的值

## 实例代码

下表对实例服务进行了描述：

服务代码	名称	描述
0Eh	获取单个属性	返回指定实例属性的值
4Bh	启动上载	开始上载过程。请求中包含了客户端在上载时可接受的“最大文件大小”。响应中包含了实际“文件大小”，该大小决不能大于“最大文件大小”和“传输大小”（每个“上载传输”请求传输的字节数）。
4Fh	上载传输	上载其他文件数据段。请求中包含“传输数量”，该数量将随着后续传输的逐个增加而递增。响应中包含匹配的“传输数量”、“传输类型”、“文件数据”以及针对上次传输的“校验和”字。“传输类型”将表明这是首个、中间还是最后一个数据包，是否是唯一的数据包，或者是否应中止传输。

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	状态	USINT	-	<p>下列情况之一：</p> <ul style="list-style-type: none"> <li>● 0: 不存在</li> <li>● 1: 文件为空 - 在从远程客户端下载之前，文件中不应有任何内容。设置后，“文件名”、“修订号”、“校验和”和“文件保存标志”值没有任何意义，并且“文件大小”为 0。</li> <li>● 2: 已加载文件 - 应用程序已预加载文件内容（文件大小 &gt; 0），或者已将文件数据下载并存储到非易失性存储器区域</li> <li>● 3: 已启动上载</li> <li>● 4: 已启动下载</li> <li>● 5: 正在进行上载</li> <li>● 6: 正在进行下载</li> <li>● 7: 正在存储到非易失性存储器区域</li> </ul>
2	获取	实例名称	STRING	-	分配给文件对象实例的唯一名称。对于 0xC8 实例，它是“EDS 和图标文件”。对于 0xC9 实例，它是“相关 EDS 和图标文件”。
3	获取	实例格式修订号	UINT	-	应用程序为该实例分配的修订号，用于区分不同的文件格式。
4	获取	文件名	STRING	-	用于文件存储的唯一名称
5	获取	文件修订号	USINT	主次	“文件修订号”会随着文件内容的更改而更新。
6	获取	文件大小	UDINT	-	文件大小（以字节为单位）
7	获取	文件校验和	UINT	-	所有字节的 16 位总和的 2 的补码
8	获取	调用方法	USINT	-	<p>定义下载文件后应采取的操作。可能的选项包括：</p> <ul style="list-style-type: none"> <li>● 0: 无操作</li> <li>● 2: 电源重置等。</li> </ul>
9	获取	文件保存参数	BYTE	-	如果设置了位 1，则应在下载完成后将文件显式保存到非易失性存储器中。
10	获取	文件类型	USINT	-	<ul style="list-style-type: none"> <li>● 0: 允许读 / 写访问</li> <li>● 1: 只读访问</li> </ul>
11	获取	文件编码格式	UINT	-	<ul style="list-style-type: none"> <li>● 0: 无编码</li> <li>● 1: 使用 ZLIB 编码</li> </ul>

## Modbus 对象

Modbus 对象提供访问 Modbus 表数据的其他方法。单个显式请求将读取或写入 1 个或更多的连续寄存器。附加传递服务允许用户指定实际的 Modbus 消息数据。

下表描述了 Modbus 对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	1	Modbus 对象的实现修订号

下表对类服务进行了描述：

服务代码	名称	描述
0Eh	获取单个属性	返回指定属性的值

### 实例代码

仅支持实例 1。

下表对实例服务进行了描述：

服务代码	名称	描述
4Bh	读取数字输入	返回一个或多个连续数字输入寄存器的值
4Ch	读取线圈	返回一个或多个连续线圈的值
4Eh	读取保持寄存器	返回一个或多个连续保持寄存器的值
4Fh	写入线圈	更新一个或多个连续线圈的值
50h	写入保持寄存器	更新一个或多个连续保持寄存器的值

**注意：**读取寄存器服务需要 4 字节的数据：第一个字包含起始寄存器地址，第二个字包含要读取的寄存器的数量。写入服务请求同样需要 4 字节数据，后跟实际数据。

**注意：**Modbus 传递服务表示特定的 Modbus 功能。转换功能不会对请求或响应数据执行任何 Indian 转换。请求和响应都包含 1 字节的 Modbus 功能代码，后跟 Modbus 消息数据，包括子功能代码（如果存在）。

**TCP/IP 接口对象**

此对象维护以太网 802.3 通讯接口的特定与链路的计数器和状态信息。

下表描述了 TCP/IP 接口对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	1	TCP/IP 接口对象的实现修订号
2	获取	最大实例数	UINT	1	最大实例数
3	获取	实例数	UINT	1	对象实例数
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	06h	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

**实例代码**

仅支持实例 1。

下表对实例服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有实例属性的值
0Eh	获取单个属性	返回指定实例属性的值

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	状态	DWORD	位级别	<ul style="list-style-type: none"> <li>● 0: 尚未配置接口配置属性。</li> <li>● 1: 接口配置包含有效配置。</li> <li>● 2...15: 保留以供将来使用。</li> </ul>
2	获取	配置能力	DWORD	位级别	<ul style="list-style-type: none"> <li>● 0: BOOTP 客户端</li> <li>● 1: DNS 客户端</li> <li>● 2: DHCP 客户端</li> <li>● 3: 支持 DHCP-DNS</li> <li>● 4: 接口配置设置表</li> </ul> 所有其他位均保留，且设置为 0。
3	获取	配置	DWORD	位级别	<ul style="list-style-type: none"> <li>● 0: 接口配置有效。</li> <li>● 1: 使用 BOOTP 获取接口配置。</li> <li>● 2: 使用 DHCP 获取接口配置。</li> <li>● 3: 保留</li> <li>● 4: DNS 启用</li> </ul> 所有其他位均保留，且设置为 0。
4	获取	物理链路	UINT	路径大小	元素路径中 16 位字节的数量
			填充的 EPATH	路径	识别物理链接对象的逻辑段。该路径被限制到一个逻辑类段和一个逻辑实例段。最大大小为 12 字节。
5	获取	接口配置	UDINT	IP 地址	-
			UDINT	网络掩码	-
			UDINT	网关地址	-
			UDINT	主要名称	-
			UDINT	辅助名称	0: 尚未配置任何辅助名称服务器地址。否则，会将名称服务器地址设置为有效的 A、B 或 C 类地址。
			STRING	缺省域名	ASCII 字符。最大长度为 48 个字符。填充为偶数个字符（填充字符本身不计入长度）。0: 未配置任何域名
6	获取	主机名	STRING	-	ASCII 字符。最大长度为 64 个字符。应填充为偶数个字符（填充字符本身不计入长度）。0: 未配置任何主机名

## 以太网链接对象

此对象提供配置 TCP/IP 网络接口设置的机制。

下表描述了以太网链接对象的类属性：

属性 ID	访问	名称	数据类型	值	详细信息
1	获取	修订号	UINT	2	以太网链接对象的实现修订号
2	获取	最大实例数	UINT	1	最大实例数
3	获取	实例数	UINT	1	对象实例数
6	获取	最大类属性	UINT	07h	最大类属性值
7	获取	最大实例属性	UINT	03h	最大实例属性值

下表对类服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有类属性的值
0Eh	获取单个属性	返回指定属性的值

### 实例代码

仅支持实例 1。

下表对实例服务进行了描述：

服务代码	名称	描述
01h	获取全部属性	返回所有实例属性的值
10h	设置单个属性	修改指定属性的值
0Eh	获取单个属性	返回指定实例属性的值

下表对实例属性进行了描述：

属性 ID	访问	名称	数据类型	值	描述
1	获取	接口速度	UDINT	-	速度 (Mbps) (10 或 100)
2	获取	接口标志	DWORD	位级别	<ul style="list-style-type: none"> <li>● 0: 链接状态</li> <li>● 1: 半双工 / 全双工</li> <li>● 2...4: 协商状态</li> <li>● 5: 手动设置 / 需要复位</li> <li>● 6: 检测到本地硬件错误</li> </ul> 所有其他位均保留，且设置为 0。
3	获取	物理地址	6 USINT 数组	-	此数组包含产品 MAC 地址。 格式 :XX-XX-XX-XX-XX-XX

## Modbus TCP 从站设备

### 概述

本节介绍 Modbus TCP 从站设备到控制器的配置。

Modbus TCP 从站设备在控制器上创建一个特定 I/O 区域，可通过 Modbus/TCP 协议进行访问。当外部 I/O 扫描器（主站）需要访问控制器的 %IW 和 %QW 对象时使用它。使用 Modbus TCP 从站设备的主要优点是收集控制器对象并可通过单个 Modbus 请求访问这些对象。

Modbus 从站设备向控制器添加另一个 Modbus 服务器功能。此服务器可由 Modbus 客户端应用程序通过使用配置的 Unit\_ID（非 255）进行访问。控制器的常规 Modbus 服务器不需要进行任何配置，可通过 Unit\_ID = 255 进行寻址。

从控制器查看输入 / 输出：由主站写入输入，并从主站读取输出。

TCP 从站设备还有能力定义特权 Modbus 客户端应用程序，它的连接绝不会被强制关闭（当需要 8 个以上的连接时，可能会关闭常规 Modbus 连接）。

与特权连接关联的超时持续时间可让您检查控制器是否受特权主站的轮询。如果在超时持续时间内未收到 Modbus 请求，则将诊断信息 i\_byMasterIpLost 设置为 1 (TRUE)。有关详细信息，请参阅以太网端口只读系统变量（参见 *Modicon M241 Logic Controller, 系统功能和变量, PLCSystem 库指南*）。

有关 Modbus TCP 的详细信息，请参阅 [www.modbus.org](http://www.modbus.org) 网站。

### 添加 Modbus TCP 从站设备

请参阅添加以太网管理器（参见第 139 页）。

## Modbus TCP 配置

要配置 Modbus TCP 从站设备，请双击**设备树**中的 **Modbus →Modbus TCP**。

显示以下对话框：

The screenshot shows a configuration window for Modbus TCP. It has three tabs: 'ModbusTCP', 'Modbus TCP 从设备 I/O 映像', and '信息'. The 'ModbusTCP' tab is selected. Below the tabs, there is a section titled '配置的参数' (Configuration Parameters). The parameters are as follows:

- IP 主站地址: 0 . 0 . 0 . 0
- 超时:  2000
- 从站端口: 502
- 单元 ID: (empty field)
- 保持寄存器 (%IW): 10
- 输入寄存器 (%QW): 10

元素	描述
<b>IP 主站地址</b>	Modbus 主站的 IP 地址 此地址上的连接不会关闭。
<b>超时</b>	超时（以毫秒为单位，步长 500 毫秒） <b>注意：</b> 超时适用于 IP 主站地址，该地址为 0.0.0.0 时除外。
<b>从站端口</b>	Modbus 通讯端口 (502)
<b>单元 ID</b>	将请求发送到 Modbus TCP 从站设备 (1...247)，而不是将请求发送到常规 Modbus 服务器 (255)。
<b>保持寄存器 (%IW)</b>	输入区中的寄存器数量 (2...40)（每个寄存器的大小是 2 字节）
<b>输入寄存器 (%QW)</b>	输出区中的寄存器数量 (2...40)（每个寄存器的大小是 2 字节）

## Modbus TCP 从站设备 I/O 映射选项卡

I/O 从主站视角映射到 Modbus 寄存器，如下所示：

- %IW<sub>s</sub> 从寄存器 0 映射到 n-1 且可读 / 写（n = 保持寄存器数量，一个保持寄存器的大小为 2 字节）。
- %QW<sub>s</sub> 从寄存器 n 映射到 n+m-1 且只读（m = 输入寄存器数量，一个输入寄存器的大小为 2 字节）。

一旦配置 Modbus TCP 从站设备，发送到其 Unit\_ID（Modbus 地址）的 Modbus 命令便会访问控制器的 %IW 和 %QW 对象，而不是常规 Modbus 字（当 Unit\_ID 为 255 时访问）。该命令帮助 Modbus TCP I/O 扫描器应用程序进行读 / 写操作。

Modbus TCP 从站设备响应 Modbus 命令的子集，其目的是与外部 I/O 扫描器交换数据。Modbus TCP 从站设备支持以下 4 个 Modbus 命令：

功能代码十进制 (十六进制)	功能	注释
3 (3 h)	读取保持寄存器	允许主站 I/O 扫描器读取设备的 %IW 和 %QW
6 (6 h)	写入单个寄存器	允许主站 I/O 扫描器写入设备的 %IW
16 (10 h)	写入多个寄存器	允许主站 I/O 扫描器写入设备的 %IW
23 (17 h)	读取 / 写入多个寄存器	允许主站 I/O 扫描器读取设备的 %IW 和 %QW，并写入设备的 %IW
其他	不支持	—

**注意：** 尝试访问 n+m-1 以上的寄存器的 Modbus 请求通过“02 - 非法数据地址”例外代码进行应答。

要将 I/O 链接到变量，请选择 **Modbus TCP Slave Device I/O Mapping** 选项卡：

Modbus TCP		Modbus TCP 从站设备 I/O 映射	信息				
通道							
变量	映射	通道	地址	类型	缺省值	单位	描述
[-] [+] [↔]		输入	%IW5	ARRAY [0...9] OF WORD			Modbus 保持寄存器
[+] [↔]	iwModbusTCT_Sla...	Inputs[0]	%IW5	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[1]	%IW6	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[2]	%IW7	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[3]	%IW8	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[4]	%IW9	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[5]	%IW10	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[6]	%IW11	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[7]	%IW12	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[8]	%IW13	WORD			
[+] [↔]	iwModbusTCT_Sla...	Inputs[9]	%IW14	WORD			
[-] [+] [↔]		输出	%QW2	ARRAY [0...9] OF WORD			Modbus 输入寄存器
[+] [↔]	qwModbusTCP_Sl...	Outputs[0]	%QW2	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[1]	%QW3	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[2]	%QW4	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[3]	%QW5	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[4]	%QW6	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[5]	%QW7	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[6]	%QW8	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[7]	%QW9	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[8]	%QW10	WORD			
[+] [↔]	qwModbusTCP_Sl...	Outputs[9]	%QW11	WORD			

通道		类型	描述
输入	IW0	WORD	保持寄存器 0
	...	...	...
	IWx	WORD	保持寄存器 x
输出	IW0	WORD	输入寄存器 0
	...	...	...
	IWy	WORD	输入寄存器 y

字数取决于 ModbusTCP 选项卡的**保持寄存器 (%IW)** 和**输入寄存器 (%QW)** 参数。

**注意：** 输出表示来自主控制器的 OUTPUT (= 控制器的 %IW)。

**注意：** 输入表示来自主控制器的 INPUT (= 控制器的 %QW)。

---

# 章 14

## 串行线路配置

---

### 简介

本章介绍如何配置 Modicon M241 Logic Controller 的串行线路通讯。

Modicon M241 Logic Controller 有 2 个串行线路端口。这些端口配置为在使用新控制器时或在控制器固件更新后使用以下协议：

- 串行线路 1: SoMachine 网络管理器。
- 串行线路 2: Modbus 管理器。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
串行线路配置	164
SoMachine 网络管理器	166
Modbus 管理器	167
ASCII 管理器	171
Modbus IOScanner	173
在 Modbus IOScanner 上添加设备	174
将调制解调器添加到管理器	179

## 串行线路配置

### 简介

通过串行线路配置窗口可以配置串行线路的物理参数（波特率、奇偶校验等）。

### 串行线路配置

要配置串行线路，请双击**设备树**中的**串行线路**。

**配置**窗口随即显示，如下所示：

对于连接到端口的每个串行设备，以下参数必须相同。

元素	描述
波特率	传输速度（位 / 秒）
奇偶校验	用于错误检测
数据位	用于传输数据的位数
停止位	停止位的数目
物理介质	指定要使用的介质： <ul style="list-style-type: none"> <li>● RS485（是否使用极化电阻器）</li> <li>● RS232（仅在串行线路 1 上可用）</li> </ul>
极化电阻器	控制器中集成了极化电阻器。通过此参数可将它们打开或关闭。

SoMachine 协议与其他协议（如 Modbus 串行线路）不兼容。如果将新控制器连接到某个配置了 Modbus 的活动串行线路，或更新连接到该串行线路的控制器的固件，则可能会导致该串行线路上的其他设备停止通讯。在下载针对预期协议正确配置了相关端口的有效应用程序之前，请确保控制器未连接到活动 Modbus 串行线路网络。

## 注意

### 串行线路中断

在将控制器物理连接到正常运行的 Modbus 串行线路网络之前，请确保应用程序针对 Modbus 正确配置了串行线路端口。

**不遵循上述说明可能导致设备损坏。**

下表指出了管理器的最大波特率值：

管理器	最大波特率（位 / 秒）
SoMachine 网络管理器	115200
Modbus 管理器	38400
ASCII 管理器	
Modbus IOScanner	

## SoMachine 网络管理器

### 简介

使用 SoMachine 网络管理器，可通过 SoMachine 软件协议与 XBTGT/XBTGK 高级面板交换变量，或者使用串行线路进行 SoMachine 编程。

### 添加管理器

要将 SoMachine 网络管理器添加到控制器，请在**硬件目录**中选择 **SoMachine – 网络管理器**，将其拖到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 (参见 *SoMachine, 编程指南*)
- 使用上下文菜单或加号按钮 (参见 *SoMachine, 编程指南*)

### 配置管理器

SoMachine 网络管理器不需要进行配置。

### 添加调制解调器

要向 SoMachine 网络管理器添加调制解调器，请参阅将调制解调器添加到管理器 (参见第 179 页)。

## Modbus 管理器

### 简介

Modbus 管理器用于主站或从站模式下的 Modbus RTU 或 ASCII 协议。

### 添加管理器

要将 Modbus 管理器添加到控制器，请在**硬件目录**中选择 **Modbus 管理器**，将其拖到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 (参见 *SoMachine, 编程指南*)
- 使用上下文菜单或加号按钮 (参见 *SoMachine, 编程指南*)

### Modbus 管理器配置

要配置控制器的 Modbus 管理器，请双击**设备树**中的 **Modbus 管理器**。

Modbus 管理器配置窗口随即显示，如下所示：

如下表所述设置参数：

元素	描述
传输模式	指定要使用的传输模式： <ul style="list-style-type: none"> <li>● RTU：使用二进制编码和 CRC 错误校验（8 个数据位）</li> <li>● ASCII：消息采用 ASCII 格式，LRC 错误校验（7 个数据位）</li> </ul> 对于链路上的每个 Modbus 设备，此参数设置必须完全相同。
寻址	指定设备类型： <ul style="list-style-type: none"> <li>● 主站</li> <li>● 从站</li> </ul>

元素	描述
地址	设备的 Modbus 地址（选择从站时）。
帧间时间（毫秒）	避免总线冲突的时间。 对于链路上的每个 Modbus 设备，此参数设置必须完全相同。
串行线路设置	串行线路配置窗口中指定的参数。

### Modbus 主站

当控制器配置为 Modbus 主站时，PLCCommunication 库中支持以下功能块：

- ADDM
- READ\_VAR
- SEND\_RECV\_MSG
- SINGLE\_WRITE
- WRITE\_READ\_VAR
- WRITE\_VAR

有关详细信息，请参阅 PLCCommunication ( 参见 *SoMachine, Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide*) 库的 PLCCommunication。

### Modbus 从站

当控制器配置为 Modbus 从站时，支持以下 Modbus 请求：

功能代码 十进制（十六进制）	子功能 十进制（十六进制）	功能
1 (1 hex)	—	读取数字量输出 (%Q)
2 (2 hex)	—	读取数字输入 (%I)
3 (3 hex)	—	读取多个寄存器 (%MW)
6 (6 hex)	—	写入单个寄存器 (%MW)
8（十六进制 8）	—	诊断
15（十六进制 F）	—	写入多个数字量输出 (%Q)
16 (10 hex)	—	写入多个寄存器 (%MW)
23 (17 hex)	—	读取 / 写入多个寄存器 (%MW)
43（十六进制 2B）	14（十六进制 E）	读取设备标识

下表包含诊断 Modbus 请求 08 支持的子功能代码：

子功能代码		功能
十进制	十六进制	
10	0A	清除计数器和诊断寄存器
11	0B	返回总线消息计数
12	0C	返回总线通讯错误计数
13	0D	返回总线异常错误计数
14	0E	返回从站消息计数
15	0F	返回从站无响应计数
16	10	返回从站 NAK 计数
17	11	返回从站忙计数
18	12	返回总线字符溢出计数

下表列出了可以使用读取设备标识请求（基本标识级别）读取的对象：

对象 ID	对象名称	类型	值
00（十六进制）	供应商代码	ASCII 字符串	Schneider Electric
01（十六进制）	产品代码	ASCII 字符串	控制器参考号 例如：TM241CE24T
02（十六进制）	主 / 次修订号	ASCII 字符串	aa.bb.cc.dd（与设备描述符相同）

下节描述控制器 Modbus 存储器映射与 HMI Modbus 映射之间的差异。如果不对应用程序进行编程以识别映射中的这些差异，则控制器和 HMI 将不会正确通讯。因此，可能会将不正确的值写入到负责输出操作的存储器区域。

## 警告

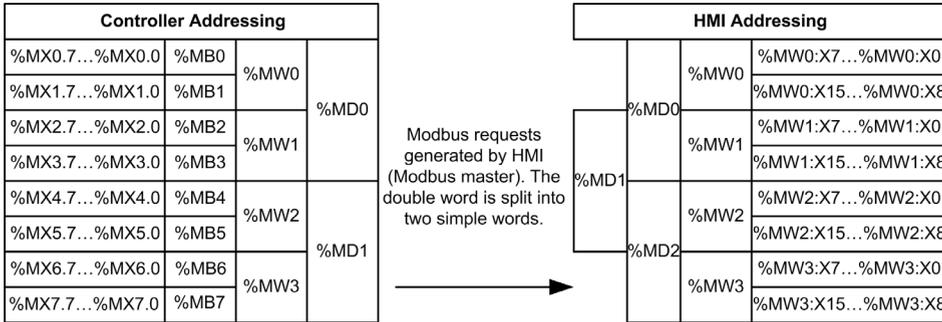
### 意外的设备操作

请将应用程序编程为可在控制器使用的 Modbus 存储器映射与任意附加 HMI 设备使用的 Modbus 存储器映射之间进行转换。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

当控制器和 Magelis HMI 通过 Modbus 进行连接时（HMI 是 Modbus 请求的主站），数据交换使用简单字请求。

使用双字时，HMI 存储器的简单字上存在重叠，但是控制器存储器中不存在这种重叠（请参见下图）。为了使 HMI 存储器区域与控制器存储器区域相匹配，HMI 存储器的双字与控制器的双字之比必须为 2。



下面提供了针对双字的存储器匹配的示例：

- HMI 的 %MD2 存储器区域对应于控制器的 %MD1 存储器区域，因为 Modbus 请求使用相同的简单字。
- HMI 的 %MD20 存储器区域对应于控制器的 %MD10 存储器区域，因为 Modbus 请求使用相同的简单字。

下面给出了针对位的存储器匹配的示例：

- HMI 的 %MW0:X9 存储器区域对应于控制器的 %MX1.1 存储器区域，因为控制器存储器中的简单字分为 2 个不同的字节。

### 添加调制解调器

要向 Modbus 管理器添加调制解调器，请参阅将调制解调器添加到管理器（参见第 179 页）。

---

## ASCII 管理器

### 简介

ASCII 管理器用于通过简单设备传输和 / 或接收数据。

### 添加管理器

要将 ASCII 管理器添加到控制器，请在**硬件目录**中选择 **ASCII 管理器**，将其拖到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南* )
- 使用上下文菜单或加号按钮 ( 参见 *SoMachine, 编程指南* )

## ASCII 管理器配置

要配置控制器的 ASCII 管理器，请双击**设备树**中的 **ASCII 管理器**。

ASCII 管理器配置窗口随即显示，如下所示：

如下表所述设置参数：

参数	描述
起始字符	如果为 0，则帧中不使用起始字符。否则，将在 <b>接收模式</b> 下使用相应的 ASCII 字符检测帧的开头。在 <b>发送模式</b> 下，此字符将添加到帧的开头。
第一个结束字符	如果为 0，则帧中不使用第一个结束字符。否则，将在 <b>接收模式</b> 下使用相应的 ASCII 字符检测帧的结尾。在 <b>发送模式</b> 下，此字符将添加到帧的结尾。
第二个结束字符	如果为 0，则帧中不使用第二个结束字符。否则，将在 <b>接收模式</b> 下使用相应的 ASCII 字符检测帧的结尾。在 <b>发送模式</b> 下，此字符将添加到帧的结尾。
收到的帧长度	如果为 0，则不使用此参数。此参数使系统可以在控制器收到指定的字符数后推断接收的帧结尾。 <b>注：</b> 此参数不能与 <b>帧收到超时（毫秒）</b> 同时使用。
帧收到超时（毫秒）	如果为 0，则不使用此参数。此参数使系统经过指定的无收发时间（毫秒）后，在接收时关闭帧结尾。
串行线路设置	串行线路配置窗口（参见第 164 页）中指定的参数。

**注意：**如果使用多个帧终止条件，则第一个为 TRUE 的条件会终止交换。

## 添加调制解调器

要向 ASCII 管理器添加调制解调器，请参阅将调制解调器添加到管理器（参见第 179 页）。

## Modbus IScanner

### 简介

Modbus IScanner 用于简化与 Modbus 从站设备的交换。

### 添加 Modbus IScanner

要将 Modbus IScanner 添加到串行线路，请在**硬件目录**中选择 **Modbus\_IOScanner**，将其拖动到**设备树**，然后将其放到一个突出显示的节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南* )
- 使用上下文菜单或加号按钮 ( 参见 *SoMachine, 编程指南* )

### Modbus IScanner 配置

要在串行线路上配置 Modbus IScanner，请双击**设备树**中的 **Modbus IScanner**。

配置窗口随即显示，如下所示：

如下表所述设置参数：

元素	描述
传输模式	指定要使用的传输模式： <ul style="list-style-type: none"> <li>• RTU：使用二进制编码和 CRC 错误校验（8 个数据位）</li> <li>• ASCII：消息采用 ASCII 格式，LRC 错误校验（7 个数据位）</li> </ul> 对于网络上的每个 Modbus 设备，此参数设置必须完全相同。
响应超时（毫秒）	交换中使用的超时。
帧间时间（毫秒）	延迟可减少总线上的数据冲突。 对于网络上的每个 Modbus 设备，此参数设置必须完全相同。

## 在 Modbus IOScanner 上添加设备

### 简介

本节介绍如何在 Modbus IOScanner 上添加设备。

### 在 Modbus IOScanner 上添加设备

要在 Modbus IOScanner 上添加设备，请在**硬件目录**中选择**一般 Modbus 从站**，将其拖动到**设备树**，然后将其放到**设备树**的 **Modbus\_IOScanner** 节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南*)
- 使用上下文菜单或加号按钮 ( 参见 *SoMachine, 编程指南*)

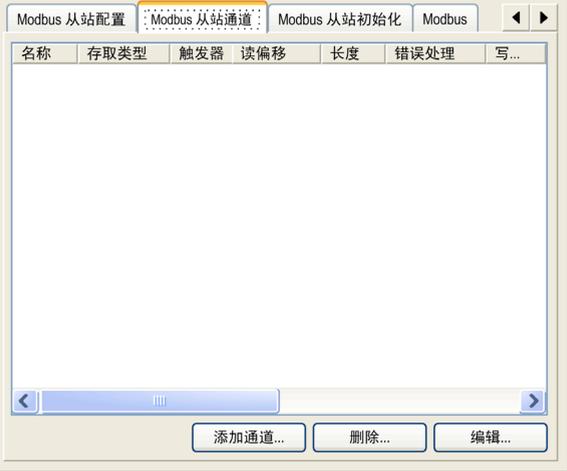
**注意：** %IWx 选项卡的 %QWx 和 %QWx 中自动创建用于交换的变量。

### 配置 Modbus IOScanner 上添加的设备

要配置 Modbus IOScanner 上添加的设备，请执行以下步骤：

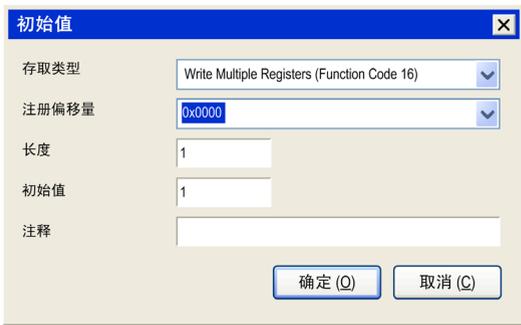
步骤	操作
1	<p>在<b>设备树</b>中，双击<b>一般 Modbus 从站</b>。</p> <p><b>结果：</b>将显示配置窗口。</p> 
2	为您的设备输入 <b>从站地址</b> 值（从 1 到 247 中选择一个值）。
3	为 <b>响应超时</b> （毫秒）选择一个值。

要配置 **Modbus 通道**，请执行以下步骤：

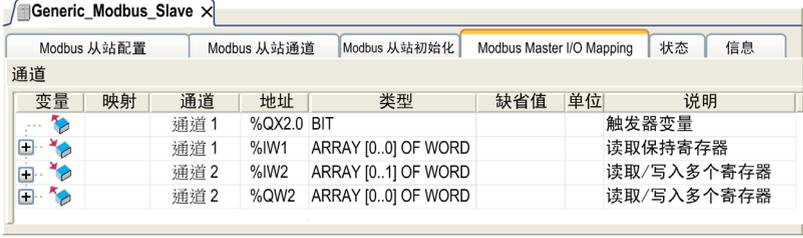
步骤	操作
1	<p>单击 <b>Modbus 从站通道</b> 选项卡：</p> 
2	<p>单击 <b>添加通道</b> 按钮：</p> 

步骤	操作
3	<p>配置交换：</p> <p>在<b>通道</b>字段中，可以添加下列值：</p> <ul style="list-style-type: none"> <li>● <b>通道</b>：输入通道的名称。</li> <li>● <b>访问类型</b>：选择交换类型：读取、写入或读 / 写多个寄存器（即 %MW）（参见第 178 页）。</li> <li>● <b>触发器</b>：选择交换的触发器。该触发器可以使用在<b>循环时间 (ms)</b> 字段中定义的周期进行<b>循环</b>，也可以通过某个布尔变量（随后会在 <b>Modbus Master I/O Mapping</b> 选项卡上创建此布尔变量）的<b>上升沿</b>来启动。</li> <li>● <b>注释</b>：添加有关此通道的注释。</li> </ul> <p>在字段<b>读寄存器</b>（如果通道是“读取”通道或“读 / 写”通道）中，可以配置要在 <b>Modbus</b> 从站上读取的 %MW。它们将映射在 %IW 上（参见“<b>Modbus Master I/O Mapping</b>”选项卡）：</p> <ul style="list-style-type: none"> <li>● <b>偏移</b>：要读取的 %MW 的偏移。0 表示将读取的第一个对象会是 %MW0。</li> <li>● <b>长度</b>：要读取的 %MW 数。例如，如果“偏移”= 2 且“长度”= 3，则通道将读取 %MW2， %MW3 和 %MW4。</li> <li>● <b>错误处理</b>：选择通讯中断时相关 %IW 的行为。</li> </ul> <p>在字段<b>写寄存器</b>（如果通道是“写入”通道或“读 / 写”通道）中，可以配置要写入到 <b>Modbus</b> 从站的 %MW。它们将映射在 %QW 上（参见“<b>Modbus Master I/O Mapping</b>”选项卡）：</p> <ul style="list-style-type: none"> <li>● <b>偏移</b>：要写入的 %MW 的偏移。0 表示将写入的第一个对象会是 %MW0。</li> <li>● <b>长度</b>：要写入的 %MW 数。例如，如果“偏移”= 2 且“长度”= 3，则通道将写入 %MW2， %MW3 和 %MW4。</li> </ul>
4	<p>单击<b>删除</b>按钮删除通道。</p> <p>单击<b>编辑</b>按钮更改通道的参数。</p>
5	<p>单击<b>确定</b>确认此通道的配置。</p>

要配置 **Modbus 初始值**，请按照以下步骤操作：

步骤	操作
1	<p>单击 <b>Modbus 从站初始化</b>选项卡：</p> 
2	<p>单击<b>新建</b>创建新的初始值：</p>  <p><b>初始值</b>窗口包含以下参数：</p> <ul style="list-style-type: none"> <li>● <b>访问类型</b>：选择交换类型：读取、写入或读 / 写多个寄存器（即 <math>\%MW</math>）（参见第 178 页）。</li> <li>● <b>寄存器偏移</b>：要初始化的寄存器的寄存器编号。</li> <li>● <b>长度</b>：要读取的 <math>\%MW</math> 数。例如，如果“偏移”= 2 且“长度”= 3，则通道将读取 <math>\%MW2</math>，<math>\%MW3</math> 和 <math>\%MW4</math>。</li> <li>● <b>初始值</b>：用于初始化寄存器的值。</li> <li>● <b>注释</b>：添加有关此通道的注释。</li> </ul>
3	<p>单击<b>上移</b>更改值在列表中的位置。 单击<b>删除</b>删除列表中的值。 单击<b>编辑</b>更改值的参数。</p>
4	<p>单击<b>确定</b>创建新的<b>初始值</b>。</p>

要配置 **Modbus Master I/O Mapping**，请按照以下步骤操作：

步骤	操作
1	<p>单击 <b>Modbus Master I/O Mapping</b> 选项卡：</p> 
2	<p>双击<b>变量</b>列的单元格，以打开文本字段。 输入变量的名称或单击浏览按钮 [...]，然后使用<b>输入助手</b>选择变量。</p>
3	<p>有关有关详细信息， I/O 映射的， 请参阅 SoMachine 编程指南。</p>

### 访问类型

下表描述了可用的各种访问类型：

功能	功能代码	可用性
Read Coils	1	<b>Modbus 通道</b>
Read Discrete Inputs	2	<b>Modbus 通道</b>
Read Holding Registers (通道配置的缺省设置)	3	<b>Modbus 通道</b>
Read Input Registers	4	<b>Modbus 通道</b>
Write Single Coil	5	<b>Modbus 通道</b> 初始化值
Write Single Register	6	<b>Modbus 通道</b> 初始化值
Write Multiple Coils	15	<b>Modbus 通道</b> 初始化值
Write Multiple Registers (从站初始化的缺省设置)	16	<b>Modbus 通道</b> 初始化值
Read/Write Multiple Registers	23	<b>Modbus 通道</b>

## 将调制解调器添加到管理器

### 简介

调制解调器可添加到以下管理器中：

- ASCII 管理器
- Modbus 管理器
- SoMachine 网络管理器

**注意：**如果您需要使用 SoMachine 网络管理器连接调制解调器，请使用调制解调器 TDW-33（它执行 AT 和 A1 命令）。

### 将调制解调器添加到管理器

要将调制解调器添加到控制器，请在**硬件目录**中选择所需的调制解调器，将其拖到**设备树**，然后将其放到管理器节点上。

有关将设备添加到项目的更多信息，请参阅：

- 使用拖放方法 ( 参见 *SoMachine, 编程指南* )
- 使用上下文菜单或加号按钮 ( 参见 *SoMachine, 编程指南* )

有关详细信息，请参阅调制解调器库 ( 参见 *SoMachine, 调制解调器功能, 调制解调器库指南* )。



# 章 15

## CANopen 配置

### CANopen 接口配置

#### CAN 总线配置

要配置控制器的 CAN 总线，请执行以下操作：

步骤	操作
1	在设备树中，双击 <b>CAN_1</b> 。
2	<p>配置波特率（缺省情况：250000 位 / 秒）：</p>  <p><b>注意：</b>使用 <b>在线总线访问</b> 选项，可以通过状态屏幕阻止 SDO、DTM 和 NMT 发送。</p>

## CANopen 管理器创建和配置

如果 **CAN** 节点下尚未显示 **CANopen 管理器**，请执行以下步骤以创建并配置它：

步骤	操作
1	<p>在<b>硬件目录</b>中选择 <b>CANopen 性能</b>，将其拖到<b>设备树</b>，然后将其放到一个突出显示的节点上。有关将设备添加到项目的详细信息，请参阅：</p> <ul style="list-style-type: none"> <li>● 使用拖放方法 (参见 <i>SoMachine, 编程指南</i>)</li> <li>● 使用上下文菜单或加号按钮 (参见 <i>SoMachine, 编程指南</i>)</li> </ul>
2	<p>双击 <b>CANopen_Performance</b>。  <b>结果：</b>显示 <b>CANopen 管理器</b> 配置窗口：</p> 

**注意：**如果选中**使能同步生产**，则会将 **CAN\_x\_Sync** 任务添加到**应用程序树**选项卡中的**应用程序**→**任务配置**节点。

**注意：**请勿删除或更改 **CAN\_x\_Sync** 任务的**类型**或**外部事件**属性。如果这样做，SoMachine 会在您尝试生成应用程序时检测到错误，并且您将无法将其下载到控制器。

**注意：**如果在 **CANopen\_Performance** 选项卡的 **CANopen 管理器**子选项卡中取消选中**使能同步生产**选项，则会自动从程序中删除 **CAN0\_Sync** 任务。

## 添加 CANopen 设备

有关添加通讯管理器和将从站设备添加到通讯管理器的详细信息，请参阅 SoMachine 编程指南。

## CANopen 操作限制

Modicon M241 Logic Controller CANopen 主站有如下操作限制：

从站设备的最大数	63
已接收 PDO (RPDO) 的最大数	252
已传输 PDO (TPDO) 的最大数	252

### 警告

#### 意外的设备操作

- 连接到控制器的 CANopen 从站设备不能超过 63 个。
- 对应用程序进行编程，以使用 252 个或更少的传输 PDO (TPDO)。
- 对应用程序进行编程，以使用 252 个或更少的接收 PDO (RPDO)。

不遵循上述说明可能导致人员伤亡或设备损坏。



---

# 章 16

## 后配置

---

### 简介

本章介绍如何生成和配置 Modicon M241 Logic Controller 的后配置文件。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
后配置表现	186
后配置文件管理	187
后配置示例	189

## 后配置表现

### 简介

使用后配置选项可以在不更改应用程序的情况下修改应用程序的某些参数。后配置参数在一个存储在控制器中的名为 **Machine.cfg** 的文件中定义。

缺省情况下，所有参数都在应用程序中设置。后配置文件中定义的参数可用于代替应用程序中定义的相应参数。并非所有参数都必须在后配置文件中指定（例如：一个参数可以在不更改网关地址的情况下更改 IP 地址）。

### 参数

使用后配置文件可以更改网络参数。

以太网参数：

- IP 地址
- 子网掩码
- 网关地址
- 传输速率
- IP 配置模式
- 设备名称
- IP 主站地址 ( 参见第 159 页 )

串行线路参数，对于应用程序中的各个串行线路（嵌入式端口或 PCI 模块）：

- 波特率
- 奇偶校验
- 数据位
- 停止位

应用程序 (TM4 module) 中每个 Profibus 的 Profibus 参数：

- 工作站地址
- 波特率

**注意：** 如果后配置文件会影响其他设备通过通讯端口使用的参数，则参数更新不会在其他设备中进行。

**注意：** 例如，如果 HMI 使用的 IP 地址在带后配置文件的配置中更新，则该 HMI 仍将使用以前的地址。您必须独立更新 HMI 使用的地址。

### 操作模式

在以下情况中，将读取后配置文件：

- 在热复位命令 ( 参见第 64 页 ) 后
- 在冷复位命令 ( 参见第 65 页 ) 后
- 在重新启动 ( 参见第 66 页 ) 后
- 在下载应用程序 ( 参见第 68 页 ) 后

有关控制器状态和转换的详细信息，请参阅控制器状态和行为 ( 参见第 49 页 )。

## 后配置文件管理

### 简介

**Machine.cfg** 文件位于目录 `/usr/cfg` 下。

每个参数都通过变量类型、变量 ID 和值进行指定。格式为：

```
id[moduleType].param[paramId].paramField=value
```

其中：

- `moduleType` 是数值，如 **111**。
- `paramId` 是指定要修改的参数的数值，如 **10000**。
- `paramField` 是字符串值，必须与 `paramId` 结合使用来指定串行线路参数，例如 **“Bauds”**。
- `value` 是分配给参数的值。其类型取决于参数的数据类型。

后配置文件中，每个参数都使用 **3** 行来进行定义：

- 第一行描述此参数的内部“路径”。
- 第二行是综合描述参数的注释。
- 第三行是参数（如上所述）的定义及其值。

### 后配置文件的生成

后配置文件 (**Machine.cfg**) 由 **SoMachine** 生成。

要生成此文件，请执行以下步骤：

步骤	操作
1	在菜单栏中，选择 <b>生成</b> → <b>生成后配置</b> <b>结果：</b> 显示一个资源管理器窗口。
2	选择后配置文件的目標文件夹。
3	单击 <b>确定</b> 。

**注意：**使用 **SoMachine** 创建后配置文件时，它会读取应用程序中当前已分配的每个参数的值，然后使用这些值写入新文件。此自动生成的文件会为可以通过后配置指定的每个参数都显式分配值。生成后配置文件后，请查看该文件并删除您希望仍由应用程序控制的所有参数赋值。仅保留您希望通过后配置功能更改、使应用程序保持轻便所需的参数赋值。

### 后配置文件的传输

创建和修改后配置文件后，请将该文件传输到控制器的 `/usr/cfg` 目录。除非 **Machine.cfg** 文件位于此目录中，否则控制器不会读取该文件。

可通过下列方法传输后配置文件：

- SD 卡（带有正确的脚本）
- 通过 FTP 服务器（参见第 [125](#) 页）下载
- 使用 **SoMachine** 控制器设备编辑器（参见第 [72](#) 页）下载

### 修改后配置文件

如果后配置文件位于 PC 中，请使用文本编辑器对其进行修改。

**注意：**请勿更改文本文件编码。缺省编码为 ANSI。

要直接在控制器中修改后配置文件，请使用 Web 服务器 (参见第 112 页) 的**设置**菜单。

### 删除后配置文件

可通过下列方法删除后配置文件：

- SD 卡（带有删除脚本）
- 通过 FTP 服务器 (参见第 125 页)
- 在线使用 SoMachine 控制器设备编辑器 (参见第 72 页) 的**文件**选项卡

有关设备编辑器的**文件**选项卡的有关详细信息，，请参阅 SoMachine 编程指南。

**注意：**

在以下情况下，将使用在应用程序中定义的参数而不是在后配置文件中定义的相应参数：

- 在热复位命令 (参见第 64 页) 后
- 在冷复位命令 (参见第 65 页) 后
- 在重新启动 (参见第 66 页) 后
- 在下载应用程序 (参见第 68 页) 后

## 后配置示例

### 后配置文件示例

```
# TM241CEC24T / Ethernet_1 / IPAddress
# Ethernet IP address
id[45000].pos[7].id[111].param[0] = [172, 30, 3, 99]

# TM241CEC24T / Ethernet_1 / SubnetMask
# Ethernet IP mask
id[45000].pos[7].id[111].param[1] = [255, 255, 0, 0]

# TM241CEC24T / Ethernet_1 / GatewayAddress
# Ethernet IP gateway address
id[45000].pos[7].id[111].param[2] = [0, 0, 0, 0]

# TM241CEC24T / Ethernet_1 / IPConfigMode
# IP configuration mode: 0:FIXED 1:BOOTP 2:DHCP
id[45000].pos[7].id[111].param[4] = 0

# TM241CEC24T / Ethernet_1 / DeviceName
# Name of the device on the Ethernet network
id[45000].pos[7].id[111].param[5] = 'my_Device'

# TM241CEC24T / Serial_Line_1 / Serial Line Configuration / Baudrate
# Serial Line Baud Rate in bit/s
id[45000].pos[8].id[40101].param[10000].Bauds = 115200

# TM241CEC24T / Serial_Line_1 / Serial Line Configuration / Parity
# Serial Line Parity (0=None, 1=Odd, 2=Even)
id[45000].pos[8].id[40101].param[10000].Parity = 0

# TM241CEC24T / Serial_Line_1 / Serial Line Configuration / DataBits
# Serial Line Data bits (7 or 8)
id[45000].pos[8].id[40101].param[10000].DataFormat = 8
```

```
# TM241CEC24T / Serial_Line_1 / Serial Line Configuration / StopBits
# Serial Line Stop bits (1 or 2)
id[45000].pos[8].id[40101].param[10000].StopBit = 1

# TM241CEC24T / Serial_Line_2 / Serial Line Configuration / Baudrate
# Serial Line Baud Rate in bit/s
id[45000].pos[9].id[40102].param[10000].Bauds = 19200

# TM241CEC24T / Serial_Line_2 / Serial Line Configuration / Parity
# Serial Line Parity (0=None, 1=Odd, 2=Even)
id[45000].pos[9].id[40102].param[10000].Parity = 2

# TM241CEC24T / Serial_Line_2 / Serial Line Configuration / DataBits
# Serial Line Data bits (7 or 8)
id[45000].pos[9].id[40102].param[10000].DataFormat = 8

# TM241CEC24T / Serial_Line_2 / Serial Line Configuration / StopBits
# Serial Line Stop bits (1 or 2)
id[45000].pos[9].id[40102].param[10000].StopBit = 1
```

---

# 章 17

## 将 Modicon M241 Logic Controller 连接到 PC

---

### 将控制器连接到 PC

#### 概述

要传输、运行和监视应用程序，请使用 USB 电缆或以太网连接（针对支持以太网端口的参考号）将控制器连接到已安装 SoMachine 的计算机。

<h3>注意</h3>
<p><b>设备无法操作</b></p> <p>务必先将通讯电缆连接到 PC 之后再连接到控制器。</p> <p>不遵循上述说明可能导致设备损坏。</p>

#### USB 供电的下载

M241 Logic Controller 可以通过 USB Mini-B 端口供电，以便执行有限的操作。二极管机制可避免 Logic Controller 同时通过 USB 和正常电源供电，或在 USB 端口上供电。

如果仅通过 USB 供电，则 Logic Controller 将执行固件和引导项目（如有），且 I/O 板在引导过程中不会通电（持续时间与正常引导相同）。控制器由 USB 供电时，USB 供电的下载会使用某些固件或某些应用程序和参数来初始化内部闪存。连接到控制器的首选工具为**控制器助手**。

控制器包装可让您经由包装的最小开口轻松使用 USB Mini-B 端口。可以使用 USB 电缆将控制器连接到 PC。长电缆不适合进行 USB 供电的下载。

<h3> 警告</h3>
<p><b>电量不足，无法进行 USB 下载</b></p> <p>请勿使用长度超出 3 米（9.8 英尺）的电缆进行 USB 供电的下载。</p> <p>不遵循上述说明可能导致人员伤亡或设备损坏。</p>

**注意：**它不支持在安装的控制器上使用 USB 供电的下载。根据已安装控制器的物理配置中的 I/O 扩展模块数，PC USB 端口可能没有完成下载的充足电源。

## USB Mini-B 端口连接

**TCSXCNAMUM3P:** 此 USB 电缆适用于持续时间较短的连接，如快速更新或检索数据值。

**BMXXCAUSBH018:** 接地并屏蔽后，此 USB 电缆适用于持续时间较长的连接。

**注意:** 每次只能将 1 台控制器或与 SoMachine 关联的任何其他设备及其组件连接到 PC。

USB Mini-B 端口是编程端口，可以用于通过 SoMachine 软件将 PC 与 USB 主机端口连接。使用典型的 USB 电缆时，此连接适用于程序的快速更新或持续时间较短的连接，以执行维护和检查数据值。如果不使用经过特殊调整的电缆以将电磁干扰的影响降到最低，则此连接不适合长期连接（如试运行或监控）。

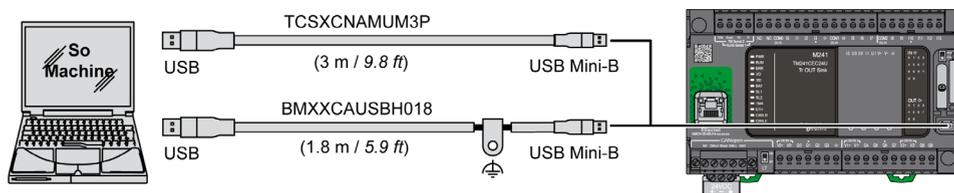
### 警告

#### 设备无法操作或意外的设备操作

- 必须使用 USB 屏蔽电缆（如 BMX XCAUSBH0），以稳固连接至系统的功能性接地 (FE) 进行长期连接。
- 不要使用 USB 连接同时连接多个控制器。

不遵循上述说明可能导致人员伤亡或设备损坏。

首先应将通讯电缆连接到 PC，以最大程度减少影响控制器的静电释放可能性。

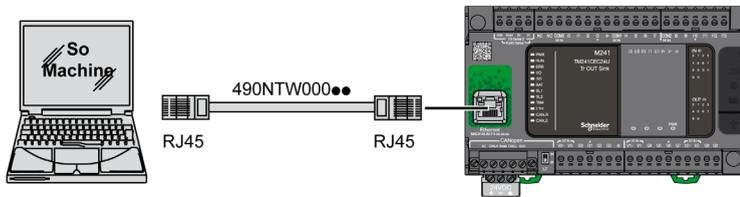


要将 USB 电缆连接到控制器，请执行以下操作：

步骤	操作
1	<p><b>1a</b> 如果使用电缆 BMXXCAUSBH018 或其他具有接地屏蔽连接的电缆建立长期连接，请确保在将电缆连接到控制器和 PC 之前，将屏蔽连接器牢固地连接到系统的功能性接地 (FE) 或保护性接地 (PE)。</p> <p><b>1b</b> 如果使用电缆 TCSXCNAMUM3P 或其他非接地 USB 电缆建立短期连接，请继续执行步骤 2。</p>
2	将 USB 电缆连接到计算机。
3	打开带铰链的入口盖。
4	将 USB 电缆的 Mini 连接器连接到控制器 USB 连接器。

### 以太网端口连接

也可以使用以太网电缆将控制器连接到 PC。



要将控制器连接到 PC，请执行下列操作：

步骤	操作
1	将以太网电缆连接到 PC。
2	将以太网电缆连接到控制器上的以太网端口。



---

# 章 18

## SD 卡

---

### 简介

本章介绍如何使用 SD 卡将固件、应用程序传输到 Modicon M241 Logic Controller。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
更新 Modicon M241 Logic Controller 固件	196
使用 SD 卡传输文件	198

## 更新 Modicon M241 Logic Controller 固件

### 简介

Modicon M241 Logic Controller 的固件更新可从 <http://www.schneider-electric.com website> 网站上获取（.zip 格式）。

可通过以下方法更新固件：

- 使用包含兼容脚本文件的 SD 卡
- 使用**控制器助手**

执行固件更改会删除设备中现有的应用程序，包括闪存中的引导应用程序。

### 注意

#### 应用程序数据丢失

- 在尝试固件更改之前需备份应用程序，将其备份到 PC 的硬盘。
- 固件更改成功后，恢复设备的应用程序。

**不遵循上述说明可能导致设备损坏。**

如果在应用程序传输或固件更改的过程中出现断电或通讯中断，那么您的设备可能无法正常工作。如果出现断电或通讯中断，请再次尝试传输。

### 注意

#### 设备无法操作

- 传输一旦开始，不要中断应用程序的传输或固件更改。
- 在传输成功完成之前不要将设备投入使用。

**不遵循上述说明可能导致设备损坏。**

当使用新控制器或更新控制器固件时，控制器的串行线路端口在缺省情况下会针对 SoMachine 协议进行配置。SoMachine 协议与其他协议（如 Modbus 串行线路）不兼容。如果将新控制器连接到配置了 Modbus 的活动串行线路，或更新连接到这些线路的控制器的固件，则可能会导致串行线路上的设备停止通讯。在首次下载针对预期协议正确配置了相关端口的有效应用程序之前，请确保控制器未连接到活动 Modbus 串行线路网络。

### 注意

#### 串行线路中断

在将控制器物理连接到正常运行的 Modbus 串行线路网络之前，请确保应用程序针对 Modbus 正确配置了串行线路端口。

**不遵循上述说明可能导致设备损坏。**

## 通过 SD 卡更新固件

遵循以下步骤以通过 SD 卡更新固件：

步骤	操作
1	将 .zip 文件解压缩到 SD 卡的根目录。 <b>注意：</b> SD 卡的文件夹 \sys\cmd\ 包含下载脚本文件。
2	拔下控制器的电源。
3	将 SD 卡插入控制器。
4	恢复控制器的电力。 <b>注意：</b> 在操作期间，SD LED（绿色）将闪烁。
5	等到下载结束（SD LED 呈常绿状态）： <ul style="list-style-type: none"> <li>● 如果检测到错误，SD LED 将关闭，ERR（红色）和 I/O（红色）LED 开始闪烁。</li> <li>● 如果下载成功结束，ERR LED（红色）将规律闪烁，并且 SD LED（绿色）稳定点亮。</li> </ul>
6	从控制器中移除 SD 卡。 <b>结果：</b> 如果下载成功结束，控制器将使用新固件自动重新启动。

## 通过控制器助手更新固件

启动 **SoMachine Central**，然后单击 **维护** → **控制器助手** 以打开 **控制器助手** 窗口。

要执行控制器的完整固件更新而不替换引导应用程序和数据，请执行以下步骤：

步骤	操作
1	在 <b>主页</b> 对话框中，单击 <b>读取自 ....</b> 控制器按钮。 <b>结果：</b> 将打开 <b>控制器选择</b> 对话框。
2	选择必需的连接类型和控制器，然后单击 <b>读取</b> 按钮。 <b>结果：</b> 映像从控制器传输到计算机。 在成功完成此操作后，会自动将您重定向到 <b>主页</b> 对话框。
3	单击 <b>新建 / 处理 ...</b> 按钮，然后单击 <b>更新固件 ...</b> 。 <b>结果：</b> 将打开用于更新固件的对话框。
4	执行个别步骤以更新当前映像中的固件（更改仅在您的计算机上的映像中受影响）。 在最后一步，您可以决定是否要为控制器读取的映像创建一个备份副本。 <b>结果：</b> 在更新固件后，您将自动返回到 <b>主页</b> 对话框。
5	在 <b>主页</b> 对话框中，单击 <b>Write on....</b> 控制器按钮。 <b>结果：</b> 将打开 <b>控制器选择</b> 对话框。
6	选择必需的连接类型和控制器，然后单击 <b>写入</b> 按钮。 <b>结果：</b> 映像从您的计算机传输到控制器。 传输完成后，您将自动返回到 <b>主页</b> 对话框。

有关固件更新以及通过固件创建新闪存磁盘的更多信息，请参阅《**SoMachine 控制器助手用户指南**》。

## 使用 SD 卡传输文件

### 简介

Modicon M241 Logic Controller 允许使用 SD 卡传输文件。

要使用 SD 卡将文件上载或下载到控制器，请使用下列方法之一：

- 克隆功能（使用空 SD 卡）
- 存储在 SD 卡上的脚本

将 SD 卡插入到控制器的 SD 卡插槽时，固件将搜索并执行包含在 SD 卡中的脚本 (/sys/cmd/Script.cmd)。

**注意：**控制器操作在文件传输期间不会被修改。

通过 **USB 大容量存储**编辑器，可以生成脚本及所有必要文件，并将其复制到 SD 卡上。

**注意：**Modicon M241 Logic Controller 只接受 FAT 或 FAT32 格式的 SD 卡。

### 警告

#### 意外的设备操作

- 在将此设备连接到控制器之前，您必须具有机器或流程的操作知识。
- 确保防护措施已到位，以便任何可能的意外设备操作不会导致人身伤害或设备损坏。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

如果在应用程序传输或固件更改的过程中出现断电或通讯中断，那么您的设备可能无法正常工作。如果出现断电或通讯中断，请再次尝试传输。

### 注意

#### 设备无法操作

- 传输一旦开始，不要中断应用程序的传输或固件更改。
- 在传输成功完成之前不要将设备投入使用。

**不遵循上述说明可能导致设备损坏。**

## 克隆功能

通过克隆功能，可以从一个控制器上载应用程序，以及将其下载到相同的参考控制器。

该功能将克隆该控制器的所有参数（例如应用程序、固件、数据文件、后配置）。请参阅存储器映射（参见第 27 页）。但是，出于安全考虑，它不会在任何目标机器上复制 Web 服务器 /FTP 密码，也不会复制任何用户访问权限。

**注意：**请确保在执行复制操作之前，在源控制器中禁用访问权限。有关“访问权限”的更多详细信息，请参阅 SoMachine 编程指南。

以下过程描述如何将控制器中的现有应用程序上载到 SD 卡：

步骤	操作
1	擦除 SD 卡并按照如下所示设置卡标签： <b>CLONExxx</b> <b>注意：</b> 标签必须以 ' <b>CLONE</b> '（不区分大小写）开始，后跟任何标准字符。
2	拔下控制器的电源。
3	将准备好的 SD 卡插入控制器。
4	恢复控制器的电力。 <b>结果：</b> 克隆操作自动开始。在克隆操作期间，下列 LED 亮起： <b>PWR</b> 、 <b>I/O</b> 和 <b>SD</b> 。 <b>注意：</b> 克隆操作持续 2 或 3 分钟。
5	请等到克隆操作完成（SD LED 关闭）。 <b>结果：</b> 控制器在正常应用模式下启动。
6	从控制器中移除 SD 卡。

以下过程描述如何将 SD 卡中的现有应用程序下载到控制器：

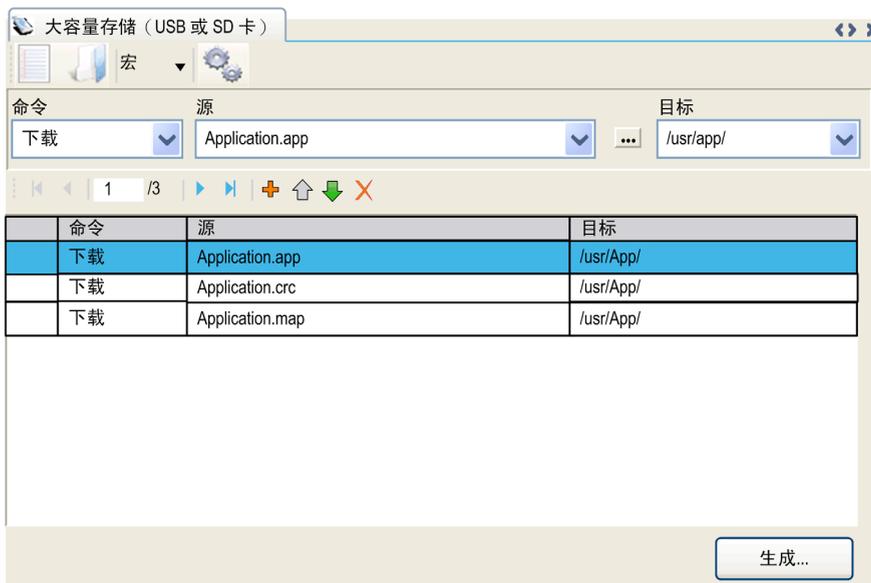
步骤	操作
1	拔下控制器的电源。
2	将 SD 卡插入控制器。
3	恢复控制器的电力。 <b>结果：</b> 克隆操作正在进行。 <b>注意：</b> 在操作期间，SD LED（绿色）将闪烁。
4	等到下载结束（SD LED 呈常绿状态）： <ul style="list-style-type: none"> <li>● 如果检测到错误，SD LED 将关闭，ERR（红色）和 I/O（红色）LED 开始闪烁。</li> <li>● 如果下载成功结束，ERR LED（红色）将规律闪烁，并且 SD LED（绿色）稳定点亮。</li> </ul>
5	删除 SD 卡以重新启动控制器。

**注意：**如果您想要控制目标控制器中已克隆应用程序的访问权限，则需要启用并建立用户访问权限，以及特定控制器的任意 Web 服务器 /FTP 密码。有关“访问权限”的更多详细信息，请参阅 SoMachine 编程指南。

**注意：**若将已克隆的应用程序下载到控制器，首先会从控制器存储器中删除现有的应用程序，不管在目标控制器中启用了哪个用户访问权限。

## 使用大容量存储生成脚本和文件

单击主菜单中的项目 → 大容量存储 (USB 或 SDCard) :



元素	描述
新建	创建新脚本。
打开	打开脚本。
宏	插入宏。 宏是一系列的单个命令。宏有助于执行许多常见操作，例如上载应用程序、下载应用程序等。
生成	在 SD 卡上生成脚本及所有必要文件。
命令	基本指令。
源	PC 或控制器上的源文件路径。
目标	PC 或控制器上的目标目录。
新增	添加脚本命令。
上移 / 下移	更改脚本命令顺序。
删除	删除脚本命令。

命令描述:

命令	描述	源	目标	语法
下载	将 SD 卡中的文件下载到控制器。	选择要下载的文件。	选择控制器目标目录。	'Download "/usr/Cfg/*"'
SetNodeName	设置控制器的节点名称。	新节点名称。	控制器节点名称	'SetNodeName "Name_PLC"'
上载	将控制器目录中包含的文件上载到 SD card。	选择目录。	-	'Upload "/usr/*"'
删除	删除控制器目录中包含的文件。 <b>注意:</b> 删除 "*" 不会删除系统文件。	选择目录并输入特定的文件名 <b>重要提示:</b> 缺省情况下, 所有目录文件均被选定。	-	'Delete "/usr/SysLog/*"'
	从控制器中删除 UserRights。	-	-	'Delete "/usr/*"'
重新启动	重新启动控制器 (仅在脚本结束后可用)。	-	-	'Reboot'

**注意:** 在控制器上激活 UserRights 时, 如果不允许用户读取 / 写入 / 删除文件系统, 则会禁用用于上载 / 下载 / 删除文件的脚本。这包括克隆操作。有关 UserRights 的详细信息, 请参阅 SoMachine 编程指南。

宏描述

宏	描述	目录 / 文件
下载应用程序	将 SD 卡中的应用程序下载到控制器。	/usr/App/*.app
上载应用程序	将控制器中的应用程序上载到 SD 卡。	/usr/App/*.crc /usr/App/*.map
下载源	将 SD 卡中的项目存档下载到控制器。	/usr/App/*.prj
上载源	将控制器中的项目存档上载到 SD 卡。	
下载多个文件	将 SD 卡中的多个文件下载到控制器目录。	由用户定义
上载日志	将控制器中的日志文件上载到 SD 卡。	/usr/Log/*.log

## 传输过程

 **警告****意外的设备操作**

- 在将此设备连接到控制器之前，您必须具有机器或流程的操作知识。
- 确保防护措施已到位，以便任何可能的意外设备操作不会导致人身伤害或设备损坏。

**不遵循上述说明可能导致人员伤亡或设备损坏。**

步骤	操作
1	使用 <b>USB 大容量存储</b> 编辑器创建脚本。
2	单击 <b>生成 ...</b> 并选择 SD 卡根目录。 <b>结果：</b> 脚本和文件传输到 SD 卡上。
3	将 SD 卡插入控制器。 <b>注意：</b> 传输期间，SD LED 呈绿色闪烁。 有关传输期间的 LED 行为的详细信息，请参阅 <b>Updating Firmware by SD Card</b> (参见第 197 页)。
4	从控制器中移除 SD 卡。 <b>注意：</b> 将在下一次重新启动后应用更改。

控制器执行脚本后，会将结果记录到 SD 卡上（文件 `/sys/cmd/Command.log`）。

 **警告****意外的设备操作**

请参阅本文档中的控制器状态和行为图以了解在重置电源后控制器将处于的状态。

**不遵循上述说明可能导致人员伤亡或设备损坏。**





## 概述

本附录列出了在技术上理解 Modicon M241 Logic Controller 编程指南所必需的文档。

## 本附录包含了哪些内容？

本附录包含了以下章节：

章	章节标题	页
A	用于在用户程序中获取 / 设置串行线路配置的功能	207
B	控制器性能	213



---

# 附录 A

## 用于在用户程序中获取 / 设置串行线路配置的功能

---

### 概述

本节介绍用于在程序中获取 / 设置串行线路配置的功能。

要使用这些功能，请添加 **M2xx 通讯库**。

有关添加库的详细信息，请参阅 **SoMachine 编程指南**。

### 本章包含了哪些内容？

本章包含了以下主题：

主题	页
GetSerialConf: 获取串行线路配置	208
SetSerialConf: 更改串行线路配置	209
SERIAL_CONF: 串行线路配置数据类型的结构	211

## GetSerialConf: 获取串行线路配置

### 功能描述

GetSerialConf 为特定串行线路的通讯端口返回配置参数。

### 图形表示形式



### 参数描述

输入	类型	注释
Link	LinkNumber (参见 <i>SoMachine, Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide</i> )	Link 是通讯端口号。
PointerToSerialConf	指向 SERIAL_CONF (参见第 211 页) 的指针	PointerToSerialConf 是配置结构的地址 (SERIAL_CONF 类型的变量), 配置参数存储在该地址中。ADR 标准功能是定义关联指针的必用功能。(请参见下面的示例。)

输出	类型	注释
GetSerialConf	WORD	此功能返回: <ul style="list-style-type: none"> <li>● 0: 返回配置参数</li> <li>● 255: 未返回配置参数, 原因是:                             <ul style="list-style-type: none"> <li>● 该功能不成功</li> <li>● 该功能正在使用中</li> </ul> </li> </ul>

### 示例

请参阅 SetSerialConf (参见第 210 页) 示例。

## SetSerialConf: 更改串行线路配置

### 功能描述

SetSerialConf 用于更改串行线路配置。

### 图形表示形式



**注意：**在执行编程时更改串行线路端口的配置，可能会中断与其他连接的设备正在进行的通讯。



### 警告

#### 意外的配置更改导致失去控制

将程序投入使用前，请验证并测试 SetSerialConf 功能的所有参数。

不遵循上述说明可能导致人员伤亡或设备损坏。

### 参数描述

输入	类型	注释
Link	LinkNumber (参见 <i>SoMachine, Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide</i> )	LinkNumber 是通讯端口号。
PointerToSerialConf	指向 SERIAL_CONF (参见第 211 页) 的指针	PointerToSerialConf 是配置结构的地址 (SERIAL_CONF 类型的变量)，新的配置参数存储在该地址中。ADR 标准功能是定义关联指针的必用功能。(请参见下面的示例。) 如果为 0，请将应用程序缺省配置设置为串行线路。

输出	类型	注释
SetSerialConf	WORD	此功能返回： <ul style="list-style-type: none"> <li>● 0: 新配置已设置</li> <li>● 255: 拒绝新配置，原因是： <ul style="list-style-type: none"> <li>● 该功能正在使用中</li> <li>● 输入参数无效</li> </ul> </li> </ul>

## 示例

```
VAR
    MySerialConf:SERIAL_CONF
    result:WORD;
END_VAR

(* 获取串行线路 1 的当前配置 *)
GetSerialConf(1, ADR(MySerialConf));

(* 更改为 modbus RTU 从站地址 9*)
MySerialConf.Protocol := 0;          (*Modbus RTU/Somachine 协议 (在此例
中, CodesysCompliant 选择该协议) *)
MySerialConf.CodesysCompliant := 0; (*Modbus RTU*)
MySerialConf.address := 9;          (* 将 modbus 地址设置为 9*)

(* 重新配置串行线路 1*)
result := SetSerialConf(1, ADR(MySerialConf));
```

## SERIAL\_CONF: 串行线路配置数据类型的结构

### 结构描述

SERIAL\_CONF 结构中包含与串行线路端口有关的配置信息。它包含以下这些变量：

变量	类型	说明
Bauds	DWORD	波特率
InterframeDelay	WORD	Modbus (RTU、ASCII) 中 2 个帧之间的最短时间 (毫秒)
FrameReceivedTimeout	WORD	在 ASCII 协议中, FrameReceivedTimeout 使系统经过指定的无收发时间 (毫秒) 后, 在接收时关闭帧结尾。如果为 0, 则不使用此参数。
FrameLengthReceived	WORD	在 ASCII 协议中, FrameLengthReceived 使系统在控制器已经接收到指定字符数的情况下关闭帧结尾。如果为 0, 则不使用此参数。
Protocol	BYTE	0: Modbus RTU 或 SoMachine (请参见 CodesysCompliant) 1: Modbus ASCII 2: ASCII
Address	BYTE	Modbus 地址为 0 到 255 (0 用于总站)
Parity	BYTE	0: 无 1: 奇 2: 偶
Rs485	BYTE	0: RS232 1: RS485
ModPol (极化电阻器)	BYTE	0: 否 1: 是
DataFormat	BYTE	7 位或 8 位
StopBit	BYTE	1: 1 个停止位 2: 2 个停止位
CharFrameStart	BYTE	在 ASCII 协议中, 0 表示帧中没有起始字符。否则, 在接收模式下, 将使用相应的 ASCII 字符检测帧的开头。在发送模式下, 此字符将添加到用户帧的开头。
CharFrameEnd1	BYTE	在 ASCII 协议中, 0 表示帧中没有第二个结束字符。否则, 在接收模式下, 将使用相应的 ASCII 字符检测帧的结尾。在发送模式下, 此字符将添加到用户帧的结尾。

变量	类型	说明
CharFrameEnd2	BYTE	在 ASCII 协议中，0 表示帧中没有第二个结束字符。否则，在接收模式下，将使用相应的 ASCII 字符（和 CharFrameEnd1）检测帧的结尾。在发送模式下，此字符将添加到用户帧的结尾。
CodesysCompliant	BYTE	0: Modbus RTU
		1: SoMachine（当 Protocol = 0 时）
CodesysNetType	BYTE	未使用

---

# 附录 B

## 控制器性能

---

### 处理性能

#### 简介

本章提供有关 M241 处理性能的信息。

#### 逻辑处理

下表显示了各个逻辑指令的逻辑处理性能：

IL 指令类型	1000 条指令的持续时间
INT 的加 / 减 / 乘法	42 微秒
DINT 的加 / 减 / 乘法	41 微秒
REAL 的加 / 减 / 乘法	336 微秒
REAL 的除法	678 微秒
BOOLEAN 的运算，例如状态 := 状态 + 值	75 微秒
LD INT + ST INT	64 微秒
LD DINT + ST DINT	49 微秒
LD REAL + ST REAL	50 微秒

#### 通讯和系统处理时间

通讯处理时间因发送 / 接收的请求数而异。

#### 事件的响应时间

下表中显示的是响应时间，即从触发外部任务的输入的信号上升沿到此任务设置的输出跳变沿的时间。事件任务还在设置输出前处理 100 条 IL 指令：

最小值	典型值	最大值
120 微秒	200 微秒	500 微秒





## 任务

一组段和子程序，**MAST** 任务为循环或周期性执行，**FAST** 任务为周期性执行。任务具有优先级，并且链接到控制器的输入和输出。可以根据任务来刷新这些 I/O。一个控制器可以有多个任务。

## 协议

一种用于控制和启用两个计算端点和设备之间的连接、通讯和数据传输的惯例或标准。

## 变量

由程序寻址和修改的存储器单元。

## 后配置

(*后配置*) 使用该选项可以在不更改应用程序的情况下修改应用程序的某些参数。后配置参数由存储在控制器上的某个文件定义。它们可能会使应用程序的配置参数过载。

## 固件

表示构成控制器上操作系统的 **BIOS**、数据参数和编程指令。固件存储在控制器内的非易失性存储器上。

## 字符串

一系列 **ASCII** 字符的变量。

## 字节

一种以 8 位格式进行编码的类型，在十六进制中，范围为 16#00 到 16#FF。

## 应用程序

包括配置数据、符号和文档的程序。

## 应用程序源

用户可读控制器指令、配置数据、**HMI** 指令、符号和其他程序文件的集合。可以将应用程序源保存在 **PC** 上，也可以将应用程序源文件下载到大多数 **Logic Controller**。应用程序源文件用来生成可在 **Logic Controller** 上运行的可执行程序。

## 开放回路

开放回路控制是指可以不使用外部传感器提供位置或速度修正信号的运动控制系统。另请参阅：*闭合回路*。

## 引导应用程序

(*引导应用程序*) 包含应用程序的二进制文件。通常，该应用程序存储在 **PLC** 上，且允许 **PLC** 在用户生成的应用程序中引导。

## 扩展总线

扩展 I/O 模块和控制器之间的电子通讯总线。

## 扫描

该功能包括：

- 读取输入并将这些值放入存储器中
- 每次执行一个应用程序指令，并将结果存储在存储器中
- 使用这些结果来更新输出

## 指令列表语言

以指令列表语言编写的程序，包括由控制器按顺序执行的一系列基于文本的指令。每个指令均包括一个行号、一个指令代码和一个操作数（请参阅 IEC 61131-3）。

## 控制器

自动化工业流程（也称为可编程 Logic Controller 或可编程控制器）。

## 数据日志

控制器在*数据日志*中记录与用户应用程序相关的事件。

## 梯形图语言

控制器程序指令的图形表示方法，其中包括控制器按顺序执行的一系列梯形中的触点、线圈和功能块符号（请参阅 IEC 61131-3）。

## 模拟量输出

在 Logic Controller 内转换数值，并按比例发送电压或电流电平。

## 毫秒

（毫秒）

## 看门狗

看门狗是一种特殊的定时器，用于确保程序不会超过为它们所分配的扫描时间。通常将看门狗定时器设置为比扫描时间较高的值，并在每个扫描周期结束后重置为零。如果看门狗定时器达到预设值（例如，因为程序陷入了死循环），则表明出现了故障且程序将会停止。

## 程序

应用程序的组成部分，其中包括可以在 Logic Controller 的存储器中安装的经过编译的源代码。

## 端子块

（端子块）安装在电子模块中的组件，用于在控制器和现场设备之间提供电气连接。

## 编码器

用于测量长度或角度的设备（线性或旋转编码器）。

## 网络

共享一个公用数据路径和通讯协议的各种互联设备系统。

## 节点

通讯网络上的可寻址设备。

## 设备

包括子组件（如传送带和转盘等）的机器的一部分。

## 连续功能图语言

一种基于功能块图语言的图形编程语言（IEC61131-3 标准的扩展），工作原理与流程图类似。但是，不可以使用网络并对图形元素进行任意定位，允许反馈回路。每个功能块的输入位于左侧，输出位于右侧。可以将功能块输出链接到其他功能块的输入以创建复合表达式。

## 配置

一个系统内硬件组件的布局 and 互连以及硬件和软件的参数，可决定系统的运行特性。

## 闪存

可覆盖的非易失性存储器它存储在一个特殊的可擦除和可重编程的 EEPROM 上。

## ARP

（地址解析协议）用于将 IP 地址映射到 MAC（硬件）地址的以太网 (Ethernet) IP 网络层协议。

## ARRAY

在 Logic Controller 存储器中以表格形式定义的单一类型数据对象的系统排列。语法如下：ARRAY [<dimension>] OF <Type>

示例 1: ARRAY [1..2] OF BOOL 是由两个 BOOL 类型的元素组成的一维表。

示例 2: ARRAY [1..10, 1..20] OF INT 是由 10 x 20 个 INT 类型的元素组成的二维表。

## BCD

（二进制编码的十进制）利用一个 4 位组（nybble/nibble，也称为半字节）表示 0 到 9 之间的十进制数的格式。在此格式中，用于编码十进制数字的四个位具有部分未使用的组合。

例如，数字 2,450 编码为 0010 0100 0101 0000。

## BOOL

（布尔）用于计算的基本数据类型。BOOL 变量可以是以下两个值之一：0 (FALSE) 或 1 (TRUE)。从 WORD 中抽取的位为 BOOL 类型，例如：%MW10.4 是编号为 10 的存储器 WORD 的五分之一位。

## BOOTP

（引导程序协议）可由网络客户端用于从服务器自动获取 IP 地址（可能还包括其他数据）的 UDP 网络协议。客户端使用客户端 MAC 地址向服务器标识自己。服务器会维护预先配置的客户端设备 MAC 地址及关联 IP 地址表，从而向客户端发送其预先配置的 IP 地址。BOOTP 最初用于使无盘主机能够通过网络远程启动。BOOTP 进程分配一个无限租期的 IP 地址。BOOTP 服务利用 UDP 端口 67 和 68。

## CFC

（连续功能图）一种基于功能块图语言的图形编程语言（IEC 61131-3 标准的扩展），工作原理与流程图类似。但是，不可以使用网络并对图形元素进行任意定位，允许反馈回路。每个功能块的输入位于左侧，输出位于右侧。可以将功能块输出链接到其他功能块的输入以创建复合表达式。

## **CIP**

(*公共工业协议*) 在网络应用层实施 CIP 后, 该协议可以与其他基于 CIP 的网络进行无线通讯, 而不考虑协议。例如, 如果在以太网 TCP/IP(Ethernet TCP/IP) 网络的应用层中实施 CIP, 则可以创建 EtherNet/IP 环境。同样, 如果在 CAN 网络的应用层实施 CIP, 则可以创建 DeviceNet 环境。在这种情况下, EtherNet/IP 网络上的设备可以通过 CIP 桥接器或路由器与 DeviceNet 网络上的设备进行通讯。

## **CRC**

(*循环冗余检验*) 用来确定通讯传输的有效性的方法。传输包含构成校验和的位域。发射器根据消息的内容来计算所使用消息的检验和。接收节点后, 按照相同的方式再次计算该字段。如果两次 CRC 计算的值存在任何差异, 则说明传输的消息与收到的消息不同。

## **DHCP**

(*动态主机配置协议*) BOOTP 的高级扩展。DHCP 虽然较为高级, 但是 DHCP 和 BOOTP 可以通用。(DHCP 可以处理 BOOTP 客户端请求。)

## **DINT**

(*双精度整数类型*) 以 32 位格式进行编码的整数。

## **DNS**

(*域名系统*) 为连接至 LAN 或 Internet 的计算机和设备进行命名的系统。

## **DTM**

(*设备类型管理器*) 分为两类:

- 连接到现场设备配置组件的设备 DTMs。
- 连接到软件通讯组件的 CommDTMs。

DTM 提供了用于访问设备参数, 以及配置、操作和诊断设备的统一结构。从用于设置设备参数的简单图形用户界面到用于诊断和维护目的而执行复杂实时计算的高度复杂的应用程序都属于 DTMs。

## **DWORD**

(*双字*) 以 32 位格式进行编码的类型。

## **EDS**

(*电子数据表*) 例如, 包含设备的属性 (如参数和设置) 的现场总线设备描述文件。

## **Ethernet**

用于 LANs 的物理和数据链路层技术, 也称为 IEE 802.3。

## **EtherNet/IP**

(*Ethernet 工业协议*) 用于工业系统中自动化解决方案制造的开放式通讯协议。EtherNet/IP 是在其上层执行公共工业协议的网络家庭成员。支持组织 (ODVA) 规定 EtherNet/IP 是为了实现全球适应性和介质独立性。

**FBD**

(*功能块图*) 控制系统的 IEC 61131-3 标准所支持的五种逻辑或控制语言中的其中一种语言。功能块图是面向图形的编程语言。它可以与一系列网络搭配使用，其中每个网络包含一个框和连接线路的图形结构，该图形结构表示逻辑或算术表达式、功能块的调用、跳转或返回指令。

**FE**

(*功能性接地*) 用于增强或以其他方式允许正常操作电敏感设备的公共接地连接（在北美地区也称为功能性接地）。

与保护性接地（保护性接地）相比，功能性接地连接可用于除防震保护以外的任何其他目的，并且通常可以承载电流。使用功能性接地连接的设备示例包括浪涌抑制器和电磁干扰滤波器、某些天线和测量仪器。

**FG**

(*频率发生器*) 使用可编程频率生成方波信号的功能。

**freewheeling**

当 Logic Controller 处于自由运行模式时，新的任务扫描都在上次扫描完成后开始。*周期扫描模式* 对比度。

**FTP**

(*文件传输协议*) 一种以客户端 - 服务器架构为构建基础的标准网络协议，用于通过基于 TCP/IP 的网络交换和操作文件，不考虑其大小。

**HE10**

用于频率低于 3 MHz 的电子信号的矩形连接器，符合 IEC 60807-2。

**I/O**

(*输入 / 输出*)

**ICMP**

(*因特网控制消息协议*) 报告检测到的错误，并提供与数据报处理有关的信息。

**IEC**

(*国际电工委员会*) 负责为所有电器、电子和相关技术制定和发布国际标准的非盈利性和非政府性的国际标准组织。

**IEC 61131-3**

工业自动化设备的 3 部分标准的第 3 部分。IEC 61131-3 针对控制器编程语言，并定义了两个图形编程语言和两个文本编程语言标准。图形编程语言既是梯形图语言又是功能块图语言。文本编程语言包括结构化文本和指令列表。

**IL**

(*指令列表*) 以某种语言编写的程序，包括由控制器按顺序执行的一系列基于文本的指令。每个指令均包括一个行号、一个指令代码和一个操作数（请参阅 IEC 61131-3）。

**INT**

(*整数*) 以 16 位格式进行编码的整数。

## IP

(*因特网协议*) TCP/IP 协议系列的一部分, 用于跟踪设备的因特网地址、对传出消息进行路由并识别传入消息。

## LD

(*梯形图*) 控制器程序指令的图形表示方法, 其中包括控制器按顺序执行的一系列梯级中的触点、线圈和功能块符号 (请参阅 IEC 61131-3)。

## LINT

(*长整数*) 以 64 位格式进行编码的整数 (INT 的 4 倍或 DINT 的 2 倍)。

## LRC

(*纵向冗余校验*) 用来确定所传输和存储的数据正确性的错误检测方法。

## LWORD

(*长字*) 以 64 位格式进行编码的数据类型。

## MAC 地址

(*介质访问控制地址*) 与特定硬件设备关联的唯一 48 位编号。在生产网卡或设备过程中, 需要为每个网卡或设备编入一个 MAC 地址。

## MAST

通过其编程软件运行的处理器任务。MAST 任务有两个段:

- **IN:** 在 MAST 任务执行之前, 将输入复制到 IN 段。
- **OUT:** 在 MAST 任务执行完后, 将输出复制到 OUT 段。

## MIB

(*管理信息库*) 通过类似 SNMP 的网络管理系统所监控的对象数据库。SNMP 用于监控由设备的 MIBs 所定义的设备。Schneider Electric 已获得了一个专用 MIB: grou peschneider (3833)。

## MSB

(*最高有效位/字节*) 在传统的十六进制或二进制表示法中, 是数字、地址或字段的一部分, 作为最左侧的单值写入。

## NMT

(*网络管理*) 提供网络初始化、检测到的错误控制和设备状态控制服务的 CANopen 协议。

## PDO

(*过程数据对象*) 无需确认的广播消息, 或在基于 CAN 的网络中从生产者设备发送到消费者设备。来自生产者设备的传输 PDO 具有特定标识符, 该标识符与消费者设备的接收 PDO 对应。

## PE

(*保护性接地*) 通过以接地电位保持设备的任何暴露的导电表面以帮助避免触电危险的公共接地连接。为了避免可能出现电压降, 在该导体上不允许电流流过 (在北美地区也称为*保护性接地*, 或在美国国家电气规范中称为设备接地导体。)

**PTO**

(*脉冲串输出*) 采用固定占空比 50-50 在开启和关闭之间振荡以产生方波形式的快速输出。PTO 特别适用于如步进器电机、频率转换器和伺服电机控制等应用。

**PWM**

(*脉冲宽度调制*) 以可调占空比在关闭和开启之间振荡以产生矩形波形式的快速输出 (尽管可以调整它来产生方波)。PTO 非常适合用于模拟或近似模拟量输出, 因为它可以通过其周期调节输出的电压, 使得它在调光或速度控制应用等中非常有用。

**REAL**

一种数据类型, 可以将它定义为以 32 位格式进行编码的浮点数。

**RJ-45**

用于为 Ethernet 定义的网络电缆的 8 针连接器的标准类型。

**RPDO**

(*接收过程数据对象*) 无需确认的广播消息, 或在基于 CAN 的网络中从生产者设备发送到消费者设备。来自生产者设备的传输 PDO 具有特定标识符, 该标识符与消费者设备的接收 PDO 对应。

**RTC**

(*实时时钟*) 由电池供电可连续运转以显示当天时间和日历的时钟, 即使在为延长电池使用寿命而未对控制器通电时也一样。

**run**

使控制器根据程序的逻辑解决方案扫描应用程序、读取物理输入并写入物理输出的命令。

**SDO**

(*服务数据对象*) 在基于 CAN 的网络中, 现场总线主站用于访问 (读 / 写) 网络节点的对象目录的消息。SDO 类型包括服务 SDOs (SSDOs) 和客户端 SDOs (CSDOs)。

**SFC**

(*顺序功能图*) 一种包括具有关联操作的步骤、具有相关联逻辑条件的转换, 以及步骤和转换之间的定向链接的语言。(SFC 标准已在 IEC 848 中定义。符合 IEC 61131-3。)

**SINT**

(*有符号整数*) 15 位值带加号。

**SNMP**

(*简单网络管理协议*) 可以通过轮询设备状态和查看与数据传输相关的信息来远程控制网络的协议。它还可用于远程管理软件和数据库。该协议还允许执行活动的管理任务, 如修改和应用新配置。

**ST**

(*结构化文本*) 一种包括复杂的语句和嵌套指令 (如迭代循环、条件执行或功能) 的语言。ST 符合 IEC 61131-3

**STOP**

使控制器停止运行应用程序的命令。

**TCP**

(*传输控制协议*) 基于连接的传输层协议，可提供同步双向数据传输。TCP 是 TCP/IP 协议套件的一部分。

**TPDO**

(*传输过程数据对象*) 无需确认的广播消息，或在基于 CAN 的网络中从生产者设备发送到消费者设备。来自生产者设备的传输 PDO 具有特定标识符，该标识符与消费者设备的接收 PDO 对应。

**UDINT**

(*无符号双精度整数*) 以 32 位格式进行编码的整数。

**UDP**

(*用户数据报协议*) 用于将数据报 (数据电报) 中的消息传递到 IP 网络上的目标计算机的无连接模式协议 (由 IETF RFC 768 定义)。UDP 协议通常与因特网协议捆绑在一起。UDP/IP 消息不要求获得响应，因此非常适合那些对于丢弃的数据包不需要重新传输 (如流视频和需要实时性能的网络) 的应用。

**UINT**

(*无符号整数*) 以 16 位格式进行编码的整数。

**WORD**

一种以 16 位格式进行编码的类型。



- ASCII 管理器, 172
- FTP 服务器
  - 以太网, 125
- GetSerialConf, 208
- I/O 总线配置, 97
- I/O 配置一般信息
  - 一般做法, 96
- Modbus
  - 协议, 110
- Modbus Ioscanner, 173
- Modbus TCP 客户端 / 服务器
  - 以太网, 110
- Modbus 管理器, 167
- SERIAL\_CONF, 211
- SetSerialConf, 209
- SNMP
  - 以太网, 127
  - 协议, 127
- Web 服务器
  - 以太网, 112
- 下载应用程序, 68
- 串行线路
  - ASCII 管理器, 172
  - Modbus 管理器, 167
- 以太网
  - Ethernet/IP 设备, 140
  - FTP 服务器, 125
  - Modbus TCP 从站设备, 159
  - Modbus TCP 客户端 / 服务器, 110
  - SNMP, 127
  - Web 服务器, 112
  - 服务, 103
- 任务
  - 事件任务, 42
  - 外部事件任务, 43
  - 循环任务, 41
  - 看门狗, 44
  - 类型, 41
  - 自由运行任务, 42
- 停止命令, 63
- 内嵌功能配置
  - 内嵌 HSC 配置, 88
  - 内嵌脉冲发生器配置, 90
  - 嵌入式 I/O 配置, 79
- 冷复位, 65
- 初始值复位, 65
- 剩余变量, 70
- 功能
  - 关键功能, 13
- 协议, 103
  - IP, 105
  - Modbus, 110
  - SNMP, 127
- 后配置, 185, 185
  - IP 地址, 186
  - IP 配置模式, 186
  - 传输速率, 186
  - 停止位, 186
  - 奇偶校验, 186
  - 子网掩码, 186
  - 工作站地址, 186
  - 数据位, 186
  - 文件管理, 187, 187
  - 波特率, 186, 186
  - 演示, 186, 186
  - 示例, 189, 189
  - 网关地址, 186
  - 设备名称, 186
- 外部事件, 43
- 存储器映射, 27
- 库, 23
- 控制器配置
  - PLC 设置, 75
  - 控制器选择, 74
  - 服务, 77
- 热复位, 64
- 状态图, 51
- 硬件初始化值, 61
- 编程语言
  - IL、LD、Grafcet, 13
- 软件初始化值, 61

输出强制, 61

输出行为, 61, 61, 61

运行命令, 63

重新启动, 66