

Modicon M251 Logic Controller

系统功能和变量 PLCSystem 库指南

04/2014



EIO0000001473.00

www.schneider-electric.com

Schneider
Electric

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	5
关于本书	7
章 1 M251 系统变量	9
1.1 系统变量：定义和使用	10
了解系统变量	11
使用系统变量	13
1.2 PLC_R 和 PLC_W 结构	14
PLC_R：控制器只读系统变量	15
PLC_W：控制器读 / 写系统变量	19
1.3 SERIAL_R 和 SERIAL_W 结构	20
SERIAL_R[0...1]：串行线路只读系统变量	21
SERIAL_W[0...1]：串行线路读 / 写系统变量	22
1.4 ETH_R 和 ETH_W 结构	23
ETH_R：以太网端口只读系统变量	24
ETH_W：以太网端口读 / 写系统变量	29
1.5 TM3_MODULE_R 结构	30
TM3_MODULE_R[0...13]：TM3 模块只读系统变量	30
1.6 PROFIBUS_R 结构	31
PROFIBUS_R：PROFIBUS 只读系统变量	31
章 2 M251 系统功能	33
2.1 M251 读取功能	34
GetRtc：获取当前 RTC 时间	35
IsFirstMastColdCycle：指示循环是否为第一个 MAST 冷启动循环	36
IsFirstMastCycle：指示循环是否为第一个 MAST 循环	37
IsFirstMastWarmCycle：指示循环是否为第一个 MAST 热启动循环	39
2.2 M251 写入功能	40
SetRTCDrift：向 RTC 设置补偿值	40
2.3 M251 用户功能	42
DataFileCopy：复制文件命令	43
ExecuteScript：脚本命令	46
2.4 TM3 读取功能	48
TM3_GetModuleBusStatus：获取 TM3 模块总线状态	49
TM3_GetModuleInternalStatus：获取 TM3 模块内部状态	50

章 3	M251 PLCSystem 库数据类型	53
3.1	PLC_R/W 系统变量数据类型	54
	PLC_R_APPLICATION_ERROR: 检测到的应用程序错误状态代码.	55
	PLC_R_BOOT_PROJECT_STATUS: 引导项目状态代码	56
	PLC_R_IO_STATUS: I/O 状态代码	57
	PLC_R_SDCARD_STATUS: SD 卡插槽状态代码	58
	PLC_R_STATUS: 控制器状态代码	59
	PLC_R_STOP_CAUSE: 从 RUN 状态向其他状态转换的原因代码 .	60
	PLC_R_TERMINAL_PORT_STATUS: 编程端口连接状态代码 . .	61
	PLC_R_TM3_BUS_STATE: TM3 总线状态代码.	62
	PLC_W_COMMAND: 控制命令代码	63
3.2	DataFileCopy 系统变量数据类型	64
	DataFileCopyError: 检测到的错误代码	65
	DataFileCopyLocation: 位置代码	66
3.3	ExecScript 系统变量数据类型	67
	ExecuteScriptError: 检测到的错误代码	67
3.4	ETH_R/W 系统变量数据类型	68
	ETH_R_FRAME_PROTOCOL: 帧传输协议代码.	69
	ETH_R_IP_MODE: IP 地址源代码.	70
	ETH_R_PORT_DUPLEX_STATUS: 传输模式代码	71
	ETH_R_PORT_IP_STATUS: 以太网 TCP/IP 端口状态代码. . . .	72
	ETH_R_PORT_LINK_STATUS: 通讯链路状态代码	73
	ETH_R_PORT_SPEED: 以太网端口的通讯速度代码.	74
	ETH_R_RUN_IDLE: Ethernet/IP 运行和空闲状态代码	75
3.5	TM3_MODULE_R 系统变量数据类型	76
	TM3_ERR_CODE: 检测到 TM3 扩展模块错误代码	77
	TM3_MODULE_R_ARRAY_TYPE: TM3 扩展模块读取数组类型. .	78
	TM3_MODULE_STATE: TM3 扩展模块状态代码	79
3.6	系统功能数据类型	80
	RTCSETDRIFT_ERROR: 检测到的 SetRTCDrift 功能错误代码 .	80
附录		81
附录 A	功能和功能块表示形式	83
	功能与功能块的区别	84
	如何通过 IL 语言使用功能或功能块	85
	如何通过 ST 语言使用功能或功能块	88
术语表		91
索引		97



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。



概览

文档范围

本文档向您介绍 Modicon M251 Logic Controller 中提供的系统功能和变量。M251 PLCSystem 库包含的功能和变量可用于从控制器系统获取信息及向控制器系统发送命令。

本文档描述 M251 PLCSystem 库的数据类型功能和变量。

需要了解以下知识：

- 有关 M251 Logic Controller 的功能、结构和配置的基本信息
- FBD、LD、ST、IL 或 CFC 语言的编程
- 系统变量（全局变量）

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 M251 Logic Controller 硬件指南	EIO0000001486（英语）； EIO0000001487（法语）； EIO0000001488（德语）； EIO0000001489（西班牙语）； EIO0000001490（意大利语）； EIO0000001491（简体中文）
Modicon M251 Logic Controller 编程指南	EIO0000001462（英语）； EIO0000001463（法语）； EIO0000001464（德语）； EIO0000001465（西班牙语）； EIO0000001466（意大利语）； EIO0000001467（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。

关于产品的资讯

 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。¹
- 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。

¹ 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

M251 系统变量

概述

本章内容：

- 提供系统变量的简介 (参见第 10 页)
- 描述 (参见第 15 页)M251 PLC 系统库中包含的系统变量

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
1.1	系统变量：定义和使用	10
1.2	PLC_R 和 PLC_W 结构	14
1.3	SERIAL_R 和 SERIAL_W 结构	20
1.4	ETH_R 和 ETH_W 结构	23
1.5	TM3_MODULE_R 结构	30
1.6	PROFIBUS_R 结构	31

节 1.1

系统变量：定义和使用

概述

本节定义系统变量以及如何在 Modicon M251 Logic Controller 中实现这些变量。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
了解系统变量	11
使用系统变量	13

了解系统变量

简介

本节介绍实现系统变量的方式。系统变量：

- 使您可以访问一般系统信息、执行系统诊断以及通过命令进行简单操作。
- 是符合 IEC 61131-3 定义和命名约定的结构化变量。您可以通过 IEC 符号名称 PLC_GVL 访问系统变量。其中一些 PLC_GVL 变量是只读变量（例如 PLC_R），另有一些变量是读 / 写变量（例如 PLC_W）。
- 会自动声明为全局变量。这些变量的作用范围是整个系统，它们可以在任何任务中通过任何程序组织单元 (POU) 进行访问。

命名约定

系统变量的标识方法如下：

- 表示系统变量类别的结构名称。例如，PLC_R 表示用于控制器诊断的只读变量的结构名称。
- 一组标识变量用途的组件名称。例如，i_wVendorID 表示控制器的供应商 ID。

访问系统变量时，您可以键入变量的结构名称，然后再输入组件名称。

下面是一个实现系统变量的示例：

```
VAR myCtr_Serial :DWORD; myCtr_ID :DWORD; myCtr_FramesRx :UDINT; END_VAR
myCtr_Serial := PLC_R.i_dwSerialNumber; myCtr_ID := PLC_R.i_wVendorID;
myCtr_FramesRx := SERIAL_R[0].i_udiFramesReceivedOK;
```

注意：上面示例中系统变量的完全限定名称为 PLC_GVL.PLC_R.i_wVendorID。在使用**输入助手**声明变量时，PLC_GVL 是隐式的，但是也可以进行完整输入。优秀编程实践通常会在在声明中使用合格的完整名称。

系统变量位置

在对控制器进行编程时，可定义两种要使用的系统变量：

- 定位变量
- 非定位变量

定位变量：

- 在静态 %MW 区域中具有固定位置：%MW60000 到 %MW60199，用于只读系统变量。
- 可在 RUNNING 和 STOPPED 状态下，通过 Modbus TCP、Modbus 串行和 EtherNet/IP 请求访问这些变量。
- 根据前面介绍的 structure_name.component_name 约定在 SoMachine 程序中使用。可以直接访问从 0 到 59999 的 %MW 地址。大于此地址的地址将被 SoMachine 视为超出范围，只能通过 structure_name.component_name 约定进行访问。

非定位变量：

- 在 %MW 区域中没有实际位置。
- 不能通过任何现场总线或网络请求访问这些变量，除非在重新定位表中找到它们，也只有这样才能在 RUNNING 和 STOPPED 状态中访问它们。重新定位表使用以下动态 %MW 区域：

- %MW60200 到 %MW61999，用于只读变量
- %MW62200 到 %MW63999，用于读 / 写变量
- 根据前面介绍的 `structure_name.component_name` 约定在 **SoMachine** 程序中使用。可以直接访问从 0 到 59999 的 %MW 地址。大于此地址的地址将被 **SoMachine** 视为超出范围，只能通过 `structure_name.component_name` 约定进行访问。

使用系统变量

简介

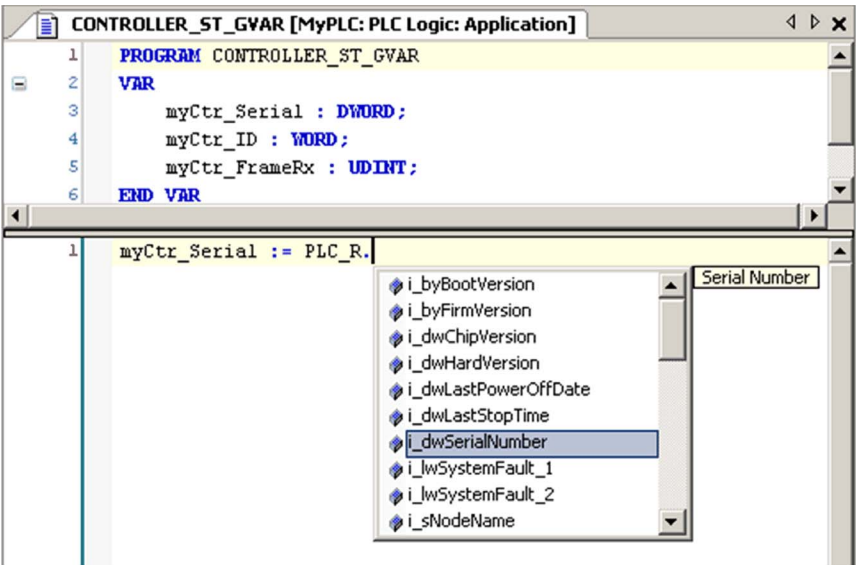
本节介绍在 SoMachine 中编程和使用系统变量所需的步骤。

系统变量是全局范围的，您可以在应用程序的所有程序组织单元 (POU) 中使用它们。

系统变量无需在全局变量列表 (GVL) 中声明。它们会自动从控制器系统库声明。

在 POU 中使用系统变量

SoMachine 具有自动完成功能。在 **POU** 中，首先输入系统变量的结构名称 (PLC_R、PLC_W 等)，然后在其后添加一个句点。系统变量显示在**输入助手**中。您可以选择所需的变量或手动输入完整名称。



注意：在上面的示例中，键入结构名称 PLC_R. 后，SoMachine 会提供一个包含可能组件名称 / 变量的弹出菜单。

示例

下面的示例介绍某些系统变量的使用方法：

```
VAR myCtr_Serial :DWORD; myCtr_ID :WORD; myCtr_FramesRx :UDINT; END_VAR  
myCtr_Serial := PLC_R.i_dwSerialNumber; myCtr_ID := PLC_R.i_wVendorID;  
myCtr_FramesRx := SERIAL_R[0].i_udiFramesReceivedOK;
```

节 1.2

PLC_R 和 PLC_W 结构

概述

本节列出 PLC_R 和 PLC_W 结构中包括的不同系统变量并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
PLC_R: 控制器只读系统变量	15
PLC_W: 控制器读 / 写系统变量	19

PLC_R: 控制器只读系统变量

变量结构

下表描述了 PLC_R 系统变量（PLC_R_STRUCT 类型）的参数：

Modbus 地址 ⁽¹⁾	变量名称	类型	注释
60000	i_wVendorID	WORD	控制器供应商 ID。 101A（十六进制）= Schneider Electric
60001	i_wProductID	WORD	控制器参考 ID。 注意： 供应商 ID 和参考 ID 是通讯设置视图中显示的 controllers 的目标 ID 的组成部分（目标 ID = 十六进制的 101A XXXX）。
60002	i_dwSerialNumber	DWORD	控制器序列号
60004	i_byFirmVersion	ARRAY[0..3] OF BYTE	控制器固件版本 [aa.bb.cc.dd]: ● i_byFirmVersion[0]= aa ● ... ● i_byFirmVersion[3]= dd
60006	i_byBootVersion	ARRAY[0..3] OF BYTE	控制器引导版本 [aa.bb.cc.dd]: ● i_byBootVersion[0]= aa ● ... ● i_byBootVersion[3]= dd
60008	i_dwHardVersion	DWORD	控制器硬件版本。
60010	i_dwChipVersion	DWORD	控制器协处理器版本。
60012	i_wStatus	PLC_R_STATUS (参见第 59 页)	控制器的状态。
60013	i_wBootProjectStatus	PLC_R_BOOT_PROJECT_STATUS (参见第 56 页)	返回有关闪存中存储的引导应用程序的信息。
60014	i_wLastStopCause	PLC_R_STOP_CAUSE (参见第 60 页)	上次从“运行”转换为其他状态的原因。
60015	i_wLastApplicationError	PLC_R_APPLICATION_ERROR (参见第 55 页)	上一次控制器异常的原因。

Modbus 地址 ⁽¹⁾	变量名称	类型	注释
60016	i_lwSystemFault_1	LWORD	位域 FFFF FFFF FFFF FFFF（十六进制）表示未检测到错误。 某个位处于低电平表示检测到错误： ● 位 0：保留 ● 位 1 = 检测到 TM3 错误 ● 位 2 = 检测到以太网 IF1 错误 ● 位 3 = 检测到以太网 IF2 错误 ● 位 4 = 检测到串行 1 过流错误 ● 位 5：保留 ● 位 6 = 检测到 CAN 1 错误 ● 位 7：保留 ● 位 8：保留 ● 位 9 = 检测到 TM4 错误 ● 位 10 = 检测到 SD 卡错误 ● 位 11 = 检测到防火墙错误
60020	i_lwSystemFault_2	LWORD	所有位均保留。
60024	i_wIOStatus1	PLC_R_IO_STATUS (参见第 57 页)	保留
60025	i_wIOStatus2	PLC_R_IO_STATUS (参见第 57 页)	TM3 I/O 状态。
60026	i_wClockBatterystatus	WORD	RTC 的电池状态： ● 0 = 需要更换电池 ● 100 = 电池已充满电 其他值 (1...99) 表示充电的百分比。例如，如果值为 75，则表示电池充满 75%。
60028	i_dwAppliSignature1	DWORD	4 个 DWORD 签名（总共 16 个字节）的第 1 个 DWORD。应用程序签名由软件在编译过程中生成。
60030	i_dwAppliSignature2	DWORD	4 个 DWORD 签名（总共 16 个字节）的第 2 个 DWORD。应用程序签名由软件在编译过程中生成。

Modbus 地址 (1)	变量名称	类型	注释
60032	i_dwAppliSignature3	DWORD	4 个 DWORD 签名 (总共 16 个字节) 的第 3 个 DWORD。应用程序签名由软件在编译过程中生成。
60034	i_dwAppliSignature4	DWORD	4 个 DWORD 签名 (总共 16 个字节) 的第 4 个 DWORD。应用程序签名由软件在编译过程中生成。
(1) 无法通过应用程序进行访问。			

无	i_sVendorName	STRING (31)	供应商的名称: “Schneider Electric”。
无	i_sProductRef	STRING (31)	控制器的参考号。
无	i_sNodeName	STRING (99)	SoMachine 网络上的节点名称。
无	i_dwLastStopTime	DWORD	上次检测到“停止”的时间 (以秒为单位, 从 1970 年 1 月 1 日 00:00 开始计起)。
无	i_dwLastPowerOffDate	DWORD	上次检测到电源关闭的日期和时间 (以秒为单位, 从 1970 年 1 月 1 日 00:00 开始计起)。 注意: 使用功能 SysTimeRtcConvertUtcToDate 将此值转换为日期和时间。有关时间和日期转换的详细信息, 请参阅系统库指南 (参见 <i>SoMachine</i> , <i>获取和设置实时时钟</i> , <i>SysTime</i> 库指南)。
无	i_uiEventsCounter	UINT	保留
无	i_wTerminalPortStatus	PLC_R_TERMINAL_PORT_STATUS (参见第 61 页)	USB 编程端口 (USB Mini-B) 的状态。
无	i_wSdCardStatus	PLC_R_SDCARD_STATUS (参见第 58 页)	SD 卡的状态。
无	i_wUsrFreeFileHdl	WORD	可用的文件句柄数。 文件句柄是系统在您打开文件时分配的资源。
无	i_udiUsrFsTotalBytes	UDINT	用户文件系统总存储器大小 (以字节为单位)。 这是用于目录 “/usr/” 的闪存大小。
无	i_udiUsrFsFreeBytes	UDINT	用户文件系统可用存储器大小 (以字节为单位)。

无	i_uiTM3BusState	PLC_R_TM3_BUS_STATE (参见第 62 页)	TM3 总线状态。 i_uiTM3BusState 可以具有以下值： ● 1: TM3_CONF_ERROR 物理配置和 SoMachine 配置之间配置不匹配。 ● 3: TM3_OK 物理配置与 SoMachine 配置相匹配。 ● 4: TM3_POWER_SUPPLY_ERROR TM3 总线未通电（例如，当逻辑控制器由 USB 供电时）。
无	i_ExpertIO_RunStop_Input	BYTE	保留
无	i_x10msClk	BOOL	TimeBase 位为 10 毫秒。 此变量以周期 = 10 毫秒切换开 / 关状态。该值在逻辑控制器处于“停止”和“运行”状态时进行切换。
无	i_x100msClk	BOOL	TimeBase 位为 100 毫秒。 此变量以周期 = 100 毫秒切换开 / 关状态。该值在逻辑控制器处于“停止”和“运行”状态时进行切换。
无	i_x1sClk	BOOL	TimeBase 位为 1 秒。 此变量以周期 = 1 秒切换开 / 关状态。该值在逻辑控制器处于“停止”和“运行”状态时进行切换。

注意：无表示没有用于此系统变量的预定义 Modbus 地址映射。

PLC_W: 控制器读 / 写系统变量

变量结构

下表对 PLC_W 系统变量（PLC_W_STRUCT 类型）的参数进行了描述：

%MW	变量名称	类型	注释
无	q_wResetCounterEvent	WORD	从 0 转换为 1 后，将复位事件计数器 (PLC_R.i_uiEventsCounter)。若要再次复位该计数器，需要先将此寄存器写为 0，之后才能再次进行从 0 到 1 的转换。
无	q_uiOpenPLCControl	UINT	当值从 0 转到 6699 后，将执行以前在下面的 PLC_W.q_wPLCControl 中写入的命令。
无	q_wPLCControl	PLC_W_COMMAND (参见第 63 页)	当系统变量 PLC_R.q_uiOpenPLCControl 值从 0 转到 6699 后，将执行控制器运行 / 停止命令。

注意：无表示没有用于此系统变量的预定义 %MW 映射。

节 1.3

SERIAL_R 和 SERIAL_W 结构

概述

本节列出 SERIAL_R 和 SERIAL_W 结构中包含的各种系统变量并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
SERIAL_R[0...1]：串行线路只读系统变量	21
SERIAL_W[0...1]：串行线路读 / 写系统变量	22

SERIAL_R[0...1]: 串行线路只读系统变量

简介

SERIAL_R 是 2 SERIAL_R_STRUCT 类型的数组。该数组的每个元素都返回对应串行线路的诊断系统变量。

对于 M251 Logic Controller:

- Serial_R[0] 指串行线路
- Serial_R[1] 保留

变量结构

下表对 SERIAL_R[0...1] 系统变量的参数进行了描述:

%MW	变量名称	类型	注释
串行线路			
无	i_udiFramesTransmittedOK	UDINT	已成功传输的帧数。
无	i_udiFramesReceivedOK	UDINT	没有检测到任何错误的已接收帧数。
无	i_udiRX_MessagesError	UDINT	检测到错误（校验和、校验位）的已接收帧数。
特定于 Modbus			
无	i_uiSlaveExceptionCount	UINT	逻辑控制器返回的 Modbus 异常响应数。
无	i_udiSlaveMsgCount	UINT	从主站接收并发送到逻辑控制器的消息数。
无	i_uiSlaveNoRespCount	UINT	逻辑控制器接收的 Modbus 广播请求数。
无	i_uiSlaveNakCount	UINT	未使用
无	i_uiSlaveBusyCount	UINT	未使用
无	i_uiCharOverrunCount	UINT	字符溢出数。

注意: 无表示没有用于此系统变量的预定义 %MW 映射。

注意:

SERIAL_R 计数器在以下情况时复位:

- 下载。
- 控制器复位。
- SERIAL_W[x].q_wResetCounter 命令。
- 通过 Modbus 请求功能代码数字 8 发出复位命令。

SERIAL_W[0...1]: 串行线路读 / 写系统变量

简介

SERIAL_W 是 2 SERIAL_W_STRUCT 类型的数组。该数组的各元素将强制相应串行线路的 SERIAL_R 系统变量复位。

对于 M251 Logic Controller:

- Serial_W[0] 指串行线路
- Serial_W[1] 保留

变量结构

下表对 SERIAL_W[0...1] 系统变量的参数进行了描述:

%MW	变量名称	类型	注释
无	q_wResetCounter	WORD	从 0 转换为 1 后, 将复位所有 SERIAL_R[0...1] 计数器。 若要再次复位这些计数器, 需要先将此寄存器写为 0, 然后才能再次进行从 0 到 1 的转换。

注意: 无表示没有用于此系统变量的预定义 %MW 映射。

节 1.4

ETH_R 和 ETH_W 结构

概述

本节列出 ETH_R 和 ETH_W 结构中包括的不同系统变量并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
ETH_R：以太网端口只读系统变量	24
ETH_W：以太网端口读 / 写系统变量	29

ETH_R: 以太网端口只读系统变量

变量结构

下表描述了 ETH_R 系统变量（ETH_R_STRUCT 类型）的参数：

%MW	变量名称	类型	注释
60050	i_byIPAddress	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_1 接口的 IP 地址 [aaa.bbb.ccc.ddd]: ● i_byIPAddress[0]= aaa ● ... ● i_byIPAddress[3]= ddd
60052	i_bySubNetMask	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_1 接口的子网掩码 [aaa.bbb.ccc.ddd]: ● i_bySub-netMask[0]= aaa ● ... ● i_bySub-netMask[3]= ddd
60054	i_byGateway	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_1 接口的网关地址 [aaa.bbb.ccc.ddd]: ● i_byGateway[0]= aaa ● ... ● i_byGateway[3]= ddd
60056	i_byMACAddress	ARRAY [0..5] OF BYTE	Ethernet 或 Ethernet_1 接口的 MAC 地址 [aa.bb.cc.dd.ee.ff]: ● i_byMACAddress[0]= aa ● ... ● i_byMACAddress[5]= ff
60059	i_sDeviceName	STRING(15)	用于获取服务器 IP 地址的名称。
无	i_wIpMode	ETH_R_IP_MODE (参见第 70 页)	用于获取 IP 地址的方法。
无	i_byFDRServerIPAddress	ARRAY [0..3] OF BYTE	DHCP 或 BootP 服务器的 IP 地址 [aaa.bbb.ccc.ddd]: ● i_byFDRServerIPAddress[0]= aaa ● ... ● i_byFDRServerIPAddress[3]= ddd 如果使用存储的 IP 或默认 IP, 则等于 0.0.0.0。
无	i_udiOpenTcpConnections	UDINT	打开的 TCP 连接数。

%MW	变量名称	类型	注释
无	i_udiFramesTransmittedOK	UDINT	已成功传输的帧数。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_udiFramedReceivedOK	UDINT	已成功接收的帧数。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_udiTransmitBufferErrors	UDINT	已传输但检测到错误的帧的数量。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_udiReceiveBufferErrors	UDINT	已接收但检测到错误的帧的数量。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_wFrameSendingProtocol	ETH_R_FRAME_PROTOCOL (参见第 69 页)	为帧发送配置的以太网协议 (IEEE 802.3 或 Ethernet II)。
无	i_wPortALinkStatus	ETH_R_PORT_LINK_STATUS (参见第 73 页)	以太网端口链路 (0 = 无链路, 1 = 链路连接到其他以太网设备)。
无	i_wPortASpeed	ETH_R_PORT_SPEED (参见第 74 页)	以太网端口网速 (10Mb/s 或 100Mb/s)。
无	i_wPortADuplexStatus	ETH_R_PORT_DUPLEX_STATUS (参见第 71 页)	以太网端口双工状态 (0 = 半双工, 1 = 全双工)。
无	i_udiPortACollisions	UDINT	遭遇一个或多个冲突但随后成功传输的帧的数量。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
特定于 Modbus TCP/IP			
无	i_udiModbusMessageTransmitted	UDINT	已传输的 Modbus 消息数。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_udiModbusMessageReceived	UDINT	已接收的 Modbus 消息数。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。

%MW	变量名称	类型	注释
无	i_udiModbusErrorMessage	UDINT	传输和接收的已检测到的 Modbus 错误消息。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。

特定于 EtherNet/IP

无	i_udiETHIP_IOMessagingTransmitted	UDINT	已传输的 EtherNet/IP 1 类帧的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_udiETHIP_IOMessagingReceived	UDINT	已接收的 EtherNet/IP 1 类帧的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_udiUCMM_Request	UDINT	已接收的 EtherNet/IP 未连接消息的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_udiUCMM_Error	UDINT	已接收的 EtherNet/IP 无效未连接消息的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_udiClass3_Request	UDINT	已接收的 EtherNet/IP 3 类请求的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_udiClass3_Error	UDINT	已接收的 EtherNet/IP 无效 3 类请求的数量。 在电源接通或使用复位命令 <code>ETH_W.q_wResetCounter</code> 后复位。
无	i_uiAssemblyInstanceInput	UINT	输入组件实例编号。有关详细信息，请参阅相应的控制器编程指南。

无	i_uiAssemblyInstanceInputSize	UINT	输入组件实例大小。有关详细信息，请参阅相应的控制器编程指南。
无	i_uiAssemblyInstanceOutput	UINT	输出组件实例编号。有关详细信息，请参阅相应的控制器编程指南。
无	i_uiAssemblyInstanceOutputSize	UINT	输出组件实例大小。有关详细信息，请参阅相应的控制器编程指南。
无	i_uiETHIP_ConnectionTimeouts	UINT	连接超时次数。在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_ucEipRunIdle	ETH_R_RUN_IDLE (参见第 75 页)	EtherNet/IP 1 类连接的运行 (值 =1) / 空闲 (值 =0) 标志。
无	i_byMasterIpTimeouts	BYTE	以太网 Modbus TCP 主站超时事件计数器。 在电源接通或使用复位命令 ETH_W.q_wResetCounter 后复位。
无	i_byMasterIpLost	BYTE	以太网 Modbus TCP 主站链路状态: 0 = 链路正常, 1 = 链路丢失。
无	i_wPortAIpStatus	ETH_R_PORT_IP_STATUS (参见第 72 页)	以太网 TCP/IP 端口栈状态。
无	i_byIPAddress_If2	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_2 接口的 IP 地址 [aaa.bbb.ccc.ddd]: ● i_byIpAddress[0]= aaa ● ... ● i_byIpAddress[3]= ddd
无	i_bySubNetMask_If2	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_2 接口的子网掩码 [aaa.bbb.ccc.ddd]: ● i_bySub-netMask[0]= aaa ● ... ● i_bySub-netMask[3]= ddd

无	i_byGateway_If2	ARRAY [0..3] OF BYTE	Ethernet 或 Ethernet_2 接口的网关地址 [aaa.bbb.ccc.ddd]: ● i_byGateway[0]= aaa ● ... ● i_byGateway[3]= ddd
无	i_byMACAddress_If2	ARRAY [0..5] OF BYTE	Ethernet 或 Ethernet_2 接口的 MAC 地址 [aa.bb.cc.dd.ee.ff]: ● i_byMACAddress[0]= aa ● ... ● i_byMACAddress[5]= ff
无	i_sDeviceName_If2	STRING(15)	用于获取服务器 IP 地址的名称。
无	i_wIpMode_If2	ETH_R_IP_MODE (参见第 70 页)	用于获取 IP 地址的方法。
无	i_wPortALinkStatus_If2	ETH_R_PORT_LINK_STATUS (参见第 73 页)	以太网端口链路 (0 = 无链路, 1 = 链路连接到其他以太网设备)。
无	i_wPortASpeed_If2	ETH_R_PORT_SPEED (参见第 74 页)	以太网端口网速 (10Mb/s 或 100Mb/s)。
无	i_wPortADuplexStatus_If2	ETH_R_PORT_DUPLEX_STATUS (参见第 71 页)	以太网端口双工状态 (0= 半双工, 1= 全双工)。
无	i_wPortAIpStatus_If2	ETH_R_PORT_IP_STATUS (参见第 72 页)	以太网 TCP/IP 端口栈状态。

注意：无表示没有用于此系统变量的预定义 %MW 映射。

ETH_W: 以太网端口读 / 写系统变量

变量结构

下表对 ETH_W 系统变量（ETH_W_STRUCT 类型）的参数进行了描述：

%MW	变量名称	类型	注释
无	q_wResetCounter	WORD	从 0 转换为 1 后，将复位所有 ETH_R 计数器。 若要再次复位，需要先将此寄存器写为 0，之后才能再次进行从 0 到 1 的转换。

注意：无表示没有用于此系统变量的预定义 %MW 映射。

节 1.5

TM3_MODULE_R 结构

TM3_MODULE_R[0...13]：TM3 模块只读系统变量

简介

TM3_MODULE_R 是包含 14 个 TM3_MODULE_R_STRUCT 类型的数组。该数组的每个元素都返回对应 TM 扩展模块的诊断系统变量。

对于 Modicon M251 Logic Controller:

- TM3_MODULE_R[0] 指 TM3 扩展模块 0
- ...
- TM3_MODULE_R[13] 指 TM3 扩展模块 13

变量结构

下表描述了 TM3_MODULE_R[0...13] 系统变量的参数：

%MW	变量名称	类型	注释
无	i_wProductID	WORD	TM3 扩展模块 ID。
无	i_wModuleState	TM3_MODULE_STATE (参见第 79 页)	描述 TM3 模块的状态。

注意：无表示没有用于此系统变量的预定义 %MW 映射。

节 1.6

PROFIBUS_R 结构

PROFIBUS_R: PROFIBUS 只读系统变量

变量结构

下表描述了 PROFIBUS_R 系统变量（PROFIBUS_R_STRUCT 类型）的参数：

%MW	变量名称	类型	注释
无	i_wPNOIdentifier	WORD	从站标识代码。
无	i_wBusAdr	UINT	PROFIBUS 从站地址。
无	i_CommState	UDINT	表示 PROFIBUS 模块状态的值： ● 0x00: 未知 ● 0x01: 未配置 ● 0x02: 停止 ● 0x03: 空闲 ● 0x04: 操作
无	i_CommError	UDINT	通讯错误代码。
无	i_ErrorCount	UDINT	通讯错误计数器。

注意：“无”表示没有用于此系统变量的预定义 %MW 映射。

章 2

M251 系统功能

概述

本章介绍 M251 PLCSystem 库中包含的系统功能。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
2.1	M251 读取功能	34
2.2	M251 写入功能	40
2.3	M251 用户功能	42
2.4	TM3 读取功能	48

节 2.1

M251 读取功能

概述

本节描述 M251 PLCSystem 库中包含的读取功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
GetRtc: 获取当前 RTC 时间	35
IsFirstMastColdCycle: 指示循环是否为第一个 MAST 冷启动循环	36
IsFirstMastCycle: 指示循环是否为第一个 MAST 循环	37
IsFirstMastWarmCycle: 指示循环是否为第一个 MAST 热启动循环	39

GetRtc: 获取当前 RTC 时间

功能描述

此功能以 UNIX 格式返回当前的 RTC 时间（精确到秒）。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表描述了 I/O 变量：

输出	类型	注释
GetRtc	DINT	UNIX 格式的当前 RTC 时间（精确到秒）。

示例

以下示例介绍如何获取当前 RTC 时间。

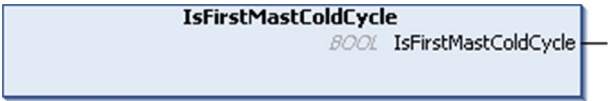
```
VAR
    MyRTC : DINT := 0;
END_VAR
MyRTC := GetRtc();
```

IsFirstMastColdCycle: 指示循环是否为第一个 **MAST** 冷启动循环

功能描述

此功能在冷启动之后的第一个 **MAST** 循环期间（下载或冷复位后的第一个循环）返回 **TRUE**。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表对输出变量进行了描述：

输出	类型	注释
IsFirstMastColdCycle	BOOL	冷启动之后的第一个 MAST 任务循环期间为 TRUE 。

示例

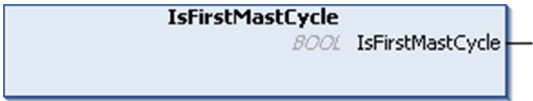
请参阅 IsFirstMastCycle (参见第 37 页) 功能。

IsFirstMastCycle: 指示循环是否为第一个 MAST 循环

功能描述

此功能在启动后的第一个 MAST 循环期间返回 TRUE。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

输出	类型	注释
IsFirstMastCycle	BOOL	启动之后的第一个 MAST 任务循环期间为 TRUE。

示例

此示例介绍三个一起使用的功能 IsFirstMastCycle、IsFirstMastColdCycle 和 IsFirstMastWarmCycle。

请在 MAST 任务中使用此示例。否则，可能会运行多次或一次也不运行（在第一个 MAST 任务循环期间可能会多次调用或不调用某个附加任务）：

```
VAR MyIsFirstMastCycle :BOOL; MyIsFirstMastWarmCycle :BOOL;
MyIsFirstMastColdCycle :BOOL; END_VAR

MyIsFirstMastWarmCycle := IsFirstMastWarmCycle(); MyIsFirstMastColdCycle
:= IsFirstMastColdCycle(); MyIsFirstMastCycle := IsFirstMastCycle();

IF (MyIsFirstMastWarmCycle) THEN
(* 这是热启动后的第一个 Mast 循环：除了保留变量，所有变量都设置为其初始化值 *)
(*=> 初始化所需的变量，以便应用程序在这种情况下按预期运行 *)
END_IF;

IF (MyIsFirstMastColdCycle) THEN
(* 这是冷启动后的第一个 Mast 循环：所有变量都设置为其初始化值，包括保留变量 *)
(*=> 初始化所需的变量，以便应用程序在这种情况下按预期运行 *)
```

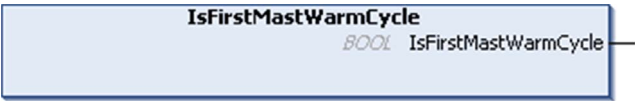
```
END_IF;  
IF (MyIsFirstMastCycle) THEN  
(* 这是启动后的第一个 Mast 循环，这里的启动是指热启动或冷启动，以及停止 / 运行命令 *)  
(*=> 初始化所需的变量，以便应用程序在这种情况下按预期运行 *)  
END_IF;
```

IsFirstMastWarmCycle: 指示循环是否为第一个 **MAST** 热启动循环

功能描述

此功能在热启动后的第一个 **MAST** 循环期间返回 **TRUE**。

图形表示形式



IL 和 ST 表示形式

若要查看 **IL** 或 **ST** 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表对输出变量进行了描述：

输出	类型	注释
IsFirstMastWarmCycle	BOOL	热启动之后的第一个主任务循环期间为 TRUE 。

示例

请参阅功能 **IsFirstMastCycle** (参见第 37 页)。

节 2.2

M251 写入功能

SetRTCDrift: 向 RTC 设置补偿值

功能描述

此功能可加快或减慢 RTC 的频率，以控制 RTC 补偿的应用，具体取决于操作环境（温度等）。每周以秒为单位提供补偿值。它可以是正值（加快）或负值（减慢）。

注意：SetRTCDrift 功能必须仅调用一次。每一次新调用都会将补偿值替换为一个新值。当 RTC 由主电源或电池供电时，该值保存在逻辑控制器硬件中。如果同时断开电池和电源的供电，将不提供 RTC 补偿值。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量描述

下表介绍输入参数：

输入	类型	注释
RtcDrift	SINT (-29..29)	每周按秒修正 (-29 ... +29)。

注意：参数 Day、Hour 和 Minute 仅用于确保向后兼容。

注意：如果为 RtcDrift 输入的值超过限制值，则逻辑控制器固件将该值设置为其最大值。

下表描述了输出变量：

输出	类型	注释
SetRTCDrift	RTCSETDRIFT_ERROR (参见第 80 页)	如果命令正常运行，则返回 RTC_OK (十六进制的 00)，否则返回检测到的错误的 ID 代码。

示例

在此示例中，仅在第一个 **MAST** 任务循环期间调用此功能一次。它每周加快 **RTC 4 秒**（一个月 **18 秒**）。

```
VAR MyRTCDrift :SINT (-29..29) := 0; MyDay :DAY_OF_WEEK; MyHour :HOUR;
MyMinute :MINUTE; END_VAR

IF IsFirstMastCycle() THEN
    MyRTCDrift := 4;
    MyDay := 0;
    MyHour := 0;
    MyMinute := 0;
    SetRTCDrift(MyRTCDrift, MyDay, MyHour, MyMinute);
END_IF
```

节 2.3

M251 用户功能

概述

本节介绍 M251 PLCSystem 库中包含的 DataFileCopy 和 ExecuteScript 功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
DataFileCopy: 复制文件命令	43
ExecuteScript: 脚本命令	46

DataFileCopy: 复制文件命令

功能描述

此功能可将存储器数据复制到文件，或将文件复制到存储器。该文件位于内部文件系统或外部文件系统（SD 卡）。

DataFileCopy 功能块可以：

- 从具有一定格式的文件中读取数据，或
- 将存储器缓冲区中的数据复制到一定格式的文件中。有关详细信息，请参阅 Flash Memory Organization (参见 *Modicon M251 Logic Controller, Programming Guide*)。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表介绍输入变量：

输入	类型	注释
xExecute	BOOL	在上升沿上，启动功能块的执行。 在下降沿上，则在其执行终结时，复位功能块的输出。
sFileName	STRING	不含扩展名的文件名（自动添加扩展名 <i>.DTA</i> ）。只使用 <i>a...z</i> 、 <i>A...Z</i> 、 <i>0...9</i> 字母数字字符。
xRead	BOOL	TRUE ：从文件复制到存储器。 FALSE ：从存储器复制到文件。
xSecure	BOOL	TRUE ：MAC 地址始终存储在文件中。只有具有相同 MAC 地址的控制器才能从该文件中读取。 FALSE ：采用相同类型存储器的其他控制器可以从该文件中读取数据。
iLocation	INT	0 ：文件位置是内部文件系统 <i>中的 /usr/Dta</i> 。 1 ：文件位置是外部文件系统（SD 卡） <i>中的 /usr/Dta</i> 。

输入	类型	注释
uiSize	UINT	表示以字节为单位的大小。最大为 65534 字节。 仅使用符合 IEC 6113-1（变量、数组、结构）规范的变量地址，例如： Variable : int; uiSize := SIZEOF (Variable);
dwAdd	DWORD	表示存储器中的地址。 仅使用符合 IEC 6113-1（变量、数组、结构）规范的变量地址，例如： Variable : int; dwAdd := ADR (Variable);

警告

意外的设备操作

在将文件复制到存储器之前，确认存储器位置的大小正确，文件的类型正确。

不遵循上述说明可能导致人员伤亡或设备损坏。

下表介绍输出变量：

输出	类型	注释
xDone	BOOL	TRUE = 表示操作已成功完成。
xBusy	BOOL	TRUE = 表示功能块正在运行。
xError	BOOL	TRUE = 表示检测到错误，功能块中止操作。
eError	DataFileCopyError (参见第 65 页)	表示检测到的数据文件复制错误的类型。

注意： 如果您在文件写入的区域内写入到存储器变量，将生成检测到的 CRC 错误。

示例

此示例介绍如何复制文件命令：

```
VAR
LocalArray : ARRAY [0..29] OF BYTE;
myFileName: STRING := 'exportfile';
EXEC_FLAG: BOOL;
DataFileCopy: DataFileCopy;
END_VAR
DataFileCopy(
  xExecute:= EXEC_FLAG,
  sFileName:= myFileName,
  xRead:= FALSE,
  xSecure:= FALSE,
  iLocation:= DFCL_INTERNAL,
  dwSize:= SIZEOF(LocalArray),
  dwAdd:= ADR(LocalArray),
  xDone=> ,
  xBusy=> ,
  xError=> ,
  eError=> );
```

ExecuteScript: 脚本命令

功能描述

此功能可以运行以下 SD 卡脚本命令：

- 下载
- 上载
- SetNodeName
- 删除
- 重新启动

使用与 USB 脚本相同的语法运行这些命令（区分大小写）。请参阅使用 SD 卡大容量存储生成脚本和文件 (参见 *Modicon M251 Logic Controller, Programming Guide*)。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表介绍输入变量：

输入	类型	注释
xExecute	BOOL	在 rising edge，开始功能块的执行。 在 falling edge，在功能块的执行终止时复位功能块的输出。
sCmd	STRING	SD 卡脚本命令语法。 不允许并行命令执行：如果正在从其他功能块或从 SD 卡脚本执行某一命令，则功能块会将命令排入队列，不会立即执行该命令。 注意： 在拔出 SD 卡之前，从 SD 卡执行的 SD 卡脚本会一直被视为正在执行。

下表介绍输出变量：

输出	类型	注释
xDone	BOOL	TRUE = 表示操作已成功完成。
xBusy	BOOL	TRUE = 功能块正在运行。
xError	BOOL	TRUE = 表示检测到错误，功能块中止操作。
eError	ExecuteScriptError (参见第 67 页)	表示检测到的执行脚本错误的类型。

示例

此示例介绍了如何执行脚本命令：

```
VAR
EXEC_FLAG: BOOL;
ExecuteScript: ExecuteScript;
END_VAR
ExecuteScript(
  xExecute:= EXEC_FLAG,
  sCmd:= 'Upload "/usr/Syslog/*"',
  xDone=> ,
  xBusy=> ,
  xError=> ,
  eError=> );
```

节 2.4

TM3 读取功能

概述

本节描述 M251 PLCSystem 库中包含的 TM3 读取功能。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
TM3_GetModuleBusStatus: 获取 TM3 模块总线状态	49
TM3_GetModuleInternalStatus: 获取 TM3 模块内部状态	50

TM3_GetModuleBusStatus: 获取 TM3 模块总线状态

功能描述

此功能返回模块的总线状态。以输入参数的形式提供模块的索引。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

下表介绍了输入变量：

输入	类型	注释
ModuleIndex	BYTE	模块的索引（0 表示第一个扩展，1 表示第二个，以此类推）。

下表介绍了输出变量：

输出	类型	注释
TM3_GetModuleBusStatus	TM3_ERR_CODE (参见第 77 页)	如果命令正常运行，则返回 TM3_OK（十六进制的 00），否则返回检测到的错误的 ID 代码。

TM3_GetModuleInternalStatus: 获取 TM3 模块内部状态

功能描述

此功能使用模块 ModuleIndex 的状态表填充 pStatusBuffer。

图形表示形式



IL 和 ST 表示形式

若要查看 IL 或 ST 语言的一般表示形式，请参阅 *功能和功能块表示形式* (参见第 83 页) 一章。

I/O 变量介绍

警告

意外的设备操作

确保已分配 pStatusBuffer。

不遵循上述说明可能导致人员伤亡或设备损坏。

下表介绍输入变量：

输入	类型	注释
ModuleIndex	BYTE	模块的索引（0 表示第一个扩展，1 表示第二个，以此类推）。
StatusOffset	BYTE	要在状态表中读取的第一个状态的偏移。
StatusSize	BYTE	要在状态表中读取的字节数。
pStatusBuffer	POINTER TO BYTE	包含读取状态表的缓冲区。

下表介绍了输出变量：

输出	类型	注释
TM3_GetModuleInternalStatus	TM3_ERR_CODE (参见第 77 页)	如果命令正常运行，则返回 TM3_OK（十六进制的 00），否则返回检测到的错误的 ID 代码。

示例

以下示例介绍如何获取模块内部状态：

```
VAR  
AMM3HT_Channel1_Input_Status: BYTE;  
END_VAR  
TM3_GetModuleInternalStatus(0, 1, 1, ADR(AMM3HT_Channel1_Input_Status));
```

章 3

M251 PLCSystem 库数据类型

概述

本章介绍 M251 PLCSystem 库的数据类型。

可用的数据类型有 2 种：

- 系统变量数据类型，由 M251 PLCSystem 库的系统变量 (参见第 9 页) (PLC_R、PLC_W 等) 使用。
- 系统功能数据类型，由 M251 PLCSystem 库的读 / 写 系统功能 (参见第 33 页) 使用。

本章包含了哪些内容？

本章包含了以下部分：

节	主题	页
3.1	PLC_R/W 系统变量数据类型	54
3.2	DataFileCopy 系统变量数据类型	64
3.3	ExecScript 系统变量数据类型	67
3.4	ETH_R/W 系统变量数据类型	68
3.5	TM3_MODULE_R 系统变量数据类型	76
3.6	系统功能数据类型	80

节 3.1

PLC_R/W 系统变量数据类型

概述

本节列出 PLC_R 和 PLC_W 结构中包括的系统变量数据类型并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
PLC_R_APPLICATION_ERROR: 检测到的应用程序错误状态代码	55
PLC_R_BOOT_PROJECT_STATUS: 引导项目状态代码	56
PLC_R_IO_STATUS: I/O 状态代码	57
PLC_R_SDCARD_STATUS: SD 卡插槽状态代码	58
PLC_R_STATUS: 控制器状态代码	59
PLC_R_STOP_CAUSE: 从 RUN 状态向其他状态转换的原因代码	60
PLC_R_TERMINAL_PORT_STATUS: 编程端口连接状态代码	61
PLC_R_TM3_BUS_STATE: TM3 总线状态代码	62
PLC_W_COMMAND: 控制命令代码	63

PLC_R_APPLICATION_ERROR: 检测到的应用程序错误状态代码

枚举类型介绍

PLC_R_APPLICATION_ERROR 枚举数据类型包含以下值:

枚举器	值	注释
PLC_R_APP_ERR_UNKNOWN	FFFF (十六进制)	检测到未定义的错误。
PLC_R_APP_ERR_NOEXCEPTION	0000 (十六进制)	未检测到错误。
PLC_R_APP_ERR_WATCHDOG	0010 (十六进制)	任务的应用程序看门狗已过期。
PLC_R_APP_ERR_HARDWAREWATCHDOG	0011 (十六进制)	硬件看门狗已过期。
PLC_R_APP_ERR_IO_CONFIG_ERROR	0012 (十六进制)	检测到不正确的 I/O 配置参数。
PLC_R_APP_ERR_UNRESOLVED_EXTREFS	0018 (十六进制)	检测到未定义的功能。
PLC_R_APP_ERR_IEC_TASK_CONFIG_ERROR	0025 (十六进制)	检测到不正确的任务配置参数。
PLC_R_APP_ERR_ILLEGAL_INSTRUCTION	0050 (十六进制)	检测到未定义的指令。
PLC_R_APP_ERR_ACCESS_VIOLATION	0051 (十六进制)	试图访问保留存储器区域。
PLC_R_APP_ERR_DIVIDE_BY_ZERO	0102 (十六进制)	检测到整数除 0。
PLC_R_APP_ERR_PROCESSORLOAD_WATCHDOG	0105 (十六进制)	处理器由于应用程序任务而过载。
PLC_R_APP_ERR_DIVIDE_REAL_BY_ZERO	0152 (十六进制)	检测到实数除 0。
PLC_R_APP_ERR_EXPIO_EVENTS_COUNT_EXCEEDED	4E20 (十六进制)	检测到专用 I/O 上有太多的事件。 减少事件任务。
PLC_R_APP_ERR_APPLICATION_VERSION_MISMATCH	4E21 (十六进制)	检测到应用程序版本不匹配。

PLC_R_BOOT_PROJECT_STATUS: 引导项目状态代码

枚举类型介绍

PLC_R_BOOT_PROJECT_STATUS 枚举数据类型包含以下值:

枚举器	值	注释
PLC_R_NO_BOOT_PROJECT	0000 （十六进制）	闪存中不存在引导项目。
PLC_R_BOOT_PROJECT_CREATION_IN_PROGRESS	0001 （十六进制）	正在创建引导项目。
PLC_R_DIFFERENT_BOOT_PROJECT	0002 （十六进制）	闪存中的引导项目与 RAM 中加载的项目不同。
PLC_R_VALID_BOOT_PROJECT	FFFF （十六进制）	闪存中的引导项目与 RAM 中加载的项目相同。

PLC_R_IO_STATUS: I/O 状态代码

枚举类型描述

PLC_R_IO_STATUS 枚举数据类型包含以下值：

枚举器	值	注释
PLC_R_IO_OK	FFFF（十六进制）	输入 / 输出运行正常。
PLC_R_IO_NO_INIT	0001（十六进制）	输入 / 输出未初始化。
PLC_R_IO_CONF_FAULT	0002（十六进制）	检测到不正确的 I/O 配置参数。
PLC_R_IO_SHORTCUT_FAULT	0003（十六进制）	检测到输入 / 输出短路。
PLC_R_IO_POWER_SUPPLY_FAULT	0004（十六进制）	检测到输入 / 输出电源错误。

PLC_R_SDCARD_STATUS: SD 卡插槽状态代码

枚举类型介绍

PLC_R_SDCARD_STATUS 枚举数据类型包含以下值：

枚举器	值	注释
NO_SDCARD	0000 （十六进制）	在插槽中未检测到 SD 卡，或未连接该插槽。
SDCARD_READONLY	0001 （十六进制）	SD 卡处于只读模式。
SDCARD_READWRITE	0002 （十六进制）	SD 卡处于读 / 写模式。
SDCARD_ERROR	0003 （十六进制）	在 SD 卡中检测到错误。

PLC_R_STATUS: 控制器状态代码

枚举类型介绍

PLC_R_STATUS 枚举数据类型包含以下值：

枚举器	值	注释
PLC_R_EMPTY	0000 （十六进制）	控制器不包含应用程序。
PLC_R_STOPPED	0001 （十六进制）	控制器已停止。
PLC_R_RUNNING	0002 （十六进制）	控制器正在运行。
PLC_R_HALT	0004 （十六进制）	控制器处于“暂停”状态。（请参见控制器 <i>编程指南</i> 中的“控制器状态图”）。
PLC_R_BREAKPOINT	0008 （十六进制）	控制器已在断点处暂停。

PLC_R_STOP_CAUSE: 从 RUN 状态向其他状态转换的原因代码

枚举类型介绍

PLC_R_STOP_CAUSE 枚举数据类型包含下列值:

枚举器	值	注释
PLC_R_STOP_REASON_UNKNOWN	00 (十六进制)	未定义初始值或停止原因。
PLC_R_STOP_REASON_HW_WATCHDOG	01 (十六进制)	在硬件看门狗超时后停止。
PLC_R_STOP_REASON_RESET	02 (十六进制)	复位后停止。
PLC_R_STOP_REASON_EXCEPTION	03 (十六进制)	例外后停止。
PLC_R_STOP_REASON_USER	04 (十六进制)	用户请求后停止。
PLC_R_STOP_REASON_IECPROGRAM	05 (十六进制)	程序命令请求 (例如: 带参数的控制命令 PLC_W.q_wPLCControl:=PLC_W_COMMAND.PLC_W_STOP;) 后停止。
PLC_R_STOP_REASON_DELETE	06 (十六进制)	删除应用程序命令后停止。
PLC_R_STOP_REASON_DEBUGGING	07 (十六进制)	进入调试模式后停止。
PLC_R_STOP_FROM_NETWORK_REQUEST	0A (十六进制)	从网络、SD 卡或 PLC_W command. 进行请求后停止。
PLC_R_STOP_FROM_INPUT	0B (十六进制)	控制器输入要求停止。
PLC_R_STOP_FROM_RUN_STOP_SWITCH	0C (十六进制)	控制器开关要求停止。
PLC_R_STOP_REASON_RETAIN_MISMATCH	0D (十六进制)	重新启动过程中检查环境测试不成功 后停止。
PLC_R_STOP_REASON_BOOT_APPLI_MISMATCH	0E (十六进制)	重新启动之前比较引导应用程序和已在存储器中的应用程序不成功 后停止。
PLC_R_STOP_REASON_POWERFAIL	0F (十六进制)	电源中断后停止。

有关控制器停止原因的详细信息, 请参阅控制器状态描述 (参见 *Modicon M251 Logic Controller, Programming Guide*)。

PLC_R_TERMINAL_PORT_STATUS：编程端口连接状态代码

枚举类型介绍

PLC_R_TERMINAL_PORT_STATUS 枚举数据类型包含以下值：

枚举器	值	注释
TERMINAL_NOT_CONNECTED	00（十六进制）	无 PC 连接到编程端口。
TERMINAL_CONNECTION_IN_PROGRESS	01（十六进制）	连接正在进行。
TERMINAL_CONNECTED	02（十六进制）	PC 已连接到编程端口。
TERMINAL_ERROR	0F（十六进制）	在连接过程中检测到错误。

PLC_R_TM3_BUS_STATE: TM3 总线状态代码

枚举类型介绍

PLC_R_TM3_BUS_STATE 枚举数据类型包含以下值:

枚举器	值	注释
TM3_CONF_ERROR	01 (十六进制)	由于物理配置和 SoMachine 中的配置不匹配而检测到错误。
TM3_OK	03 (十六进制)	检测到物理配置和 SoMachine 中的配置正确匹配。
TM3_POWER_SUPPLY_ERROR	04 (十六进制)	检测到电源错误。

PLC_W_COMMAND：控制命令代码

枚举类型介绍

PLC_W_COMMAND 枚举数据类型包含以下值：

枚举器	值	注释
PLC_W_STOP	0001 （十六进制）	用于停止控制器的命令。
PLC_W_RUN	0002 （十六进制）	用于运行控制器的命令。
PLC_W_RESET_COLD	0004 （十六进制）	用于启动控制器冷复位的命令。
PLC_W_RESET_WARM	0008 （十六进制）	用于启动控制器热复位的命令。

节 3.2

DataFileCopy 系统变量数据类型

概述

本节列出 DataFileCopy 结构中包含的系统变量数据类型并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
DataFileCopyError: 检测到的错误代码	65
DataFileCopyLocation: 位置代码	66

DataFileCopyError: 检测到的错误代码

枚举类型描述

DataFileCopyError 枚举数据类型包含下列值:

枚举器	值	描述
ERR_NO_ERR	00 (十六进制)	未检测到错误。
ERR_FILE_NOT_FOUND	01 (十六进制)	文件不存在。
ERR_FILE_ACCESS_REFUSED	02 (十六进制)	无法打开文件。
ERR_INCORRECT_SIZE	03 (十六进制)	请求大小与从文件中读取的大小不一致。
ERR_CRC_ERR	04 (十六进制)	CRC 不正确, 文件大概已损坏。
ERR_INCORRECT_MAC	05 (十六进制)	试图读取文件的控制器的 MAC 地址与文件中包含的 MAC 地址不一致。

DataFileCopyLocation: 位置代码

枚举类型介绍

DataFileCopyLocation 枚举数据类型包含下列值:

枚举器	值	描述
DFCL_INTERNAL	00 (十六进制)	带有 DTA 扩展名的数据文件位于 <i>/usr/Dta</i> 目录中。
DFCL_EXTERNAL	01 (十六进制)	带有 DTA 扩展名的数据文件位于 <i>/sd0/usr/Dta</i> 目录中。
DFCL_TBD	02 (十六进制)	未使用。

节 3.3

ExecScript 系统变量数据类型

ExecuteScriptError: 检测到的错误代码

枚举类型描述

ExecuteScriptError 枚举数据类型包含下列值:

枚举器	值	描述
CMD_OK	00 (十六进制)	未检测到错误。
ERR_CMD_UNKNOWN	01 (十六进制)	不识别该命令。
ERR_SD_CARD_MISSING	02 (十六进制)	不存在 SD 卡。
ERR_SEE_FWLOG	03 (十六进制)	在命令执行过程中检测到错误, 参见 FwLog.txt。有关详细信息, 请参阅文件类型 (参见 <i>Modicon M251 Logic Controller, Programming Guide</i>)。
ERR_ONLY_ONE_COMMAND_ALLOWED	04 (十六进制)	试图同时执行多个脚本。
CMD_BEING_EXECUTED	05 (十六进制)	某一脚本已经在执行。

节 3.4

ETH_R/W 系统变量数据类型

概述

本节列出 ETH_R 和 ETH_W 结构中包括的系统变量数据类型并加以说明。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
ETH_R_FRAME_PROTOCOL: 帧传输协议代码	69
ETH_R_IP_MODE: IP 地址源代码	70
ETH_R_PORT_DUPLEX_STATUS: 传输模式代码	71
ETH_R_PORT_IP_STATUS: 以太网 TCP/IP 端口状态代码	72
ETH_R_PORT_LINK_STATUS: 通讯链路状态代码	73
ETH_R_PORT_SPEED: 以太网端口的通讯速度代码	74
ETH_R_RUN_IDLE: Ethernet/IP 运行和空闲状态代码	75

ETH_R_FRAME_PROTOCOL：帧传输协议代码

枚举类型介绍

ETH_R_FRAME_PROTOCOL 枚举数据类型包含以下值：

枚举器	值	注释
ETH_R_802_3	00（十六进制）	用于帧传输的协议为 IEEE 802.3。
ETH_R_ETHERNET_II	01（十六进制）	用于帧传输的协议为 Ethernet II。

ETH_R_IP_MODE: IP 地址源代码

枚举类型介绍

ETH_R_IP_MODE 枚举数据类型包含以下值：

枚举器	值	注释
ETH_R_STORED	00（十六进制）	使用存储的 IP 地址。
ETH_R_BOOTP	01（十六进制）	使用引导程序协议获取 IP 地址。
ETH_R_DHCP	02（十六进制）	使用 DHCP 协议获取 IP 地址。
ETH_DEFAULT_IP	FF（十六进制）	使用缺省 IP 地址。

ETH_R_PORT_DUPLEX_STATUS: 传输模式代码

枚举类型介绍

ETH_R_PORT_DUPLEX_STATUS 枚举数据类型包含以下值：

枚举器	值	注释
ETH_R_PORT_HALF_DUPLEX	00（十六进制）	使用半双工传输模式。
ETH_R_FULL_DUPLEX	01（十六进制）	使用全双工传输模式。
ETH_R_PORT_NA_DUPLEX	03（十六进制）	不使用双工传输模式。

ETH_R_PORT_IP_STATUS: 以太网 TCP/IP 端口状态代码

枚举类型介绍

ETH_R_PORT_IP_STATUS 枚举数据类型包含以下值:

枚举器	值	注释
WAIT_FOR_PARAMS	00 (十六进制)	等待参数。
WAIT_FOR_CONF	01 (十六进制)	正在等待配置。
DATA_EXCHANGE	02 (十六进制)	数据交换准备就绪。
ETH_ERROR	03 (十六进制)	检测到以太网 TCP/IP 端口错误 (电缆断开连接、无效配置等)。
DUPLICATE_IP	04 (十六进制)	IP 地址已被其他设备使用。

ETH_R_PORT_LINK_STATUS: 通讯链路状态代码

枚举类型介绍

ETH_R_PORT_LINK_STATUS 枚举数据类型包含以下值:

枚举器	值	注释
ETH_R_LINK_DOWN	00 (十六进制)	通讯链路不可用于其他设备。
ETH_R_LINK_UP	01 (十六进制)	通讯链路可用于其他设备。

ETH_R_PORT_SPEED: 以太网端口的通讯速度代码

枚举类型介绍

ETH_R_PORT_SPEED 枚举数据类型包含以下值：

枚举器	值	注释
ETH_R_SPEED_NA	0（十进制）	网络速度为每秒 0 兆位。
ETH_R_SPEED_10_MB	10（十进制）	网络速度为每秒 10 兆位。
ETH_R_100_MB	100（十进制）	网络速度为每秒 100 兆位。

ETH_R_RUN_IDLE: Ethernet/IP 运行和空闲状态代码

枚举类型介绍

ETH_R_RUN_IDLE 枚举数据类型包含以下值：

枚举器	值	注释
IDLE	00 （十六进制）	EtherNet/IP 连接空闲。
RUN	01 （十六进制）	EtherNet/IP 连接正在运行。

节 3.5

TM3_MODULE_R 系统变量数据类型

概述

本节列出并描述 TM3_MODULE_R 结构中包含的系统变量数据类型。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
TM3_ERR_CODE: 检测到 TM3 扩展模块错误代码	77
TM3_MODULE_R_ARRAY_TYPE: TM3 扩展模块读取数组类型	78
TM3_MODULE_STATE: TM3 扩展模块状态代码	79

TM3_ERR_CODE: 检测到 TM3 扩展模块错误代码

枚举类型介绍

TM3_ERR_CODE 枚举数据类型包含以下值：

枚举器	值	注释
TM3_NO_ERR	00（十六进制）	上一次与扩展模块的总线交换成功。
TM3_ERR_FAILED	01（十六进制）	由于上一次与扩展模块的总线交换不成功而检测到错误。
TM3_ERR_PARAMETER	02（十六进制）	在上一次与模块的总线交换中检测到参数错误。
TM3_ERR_COK	03（十六进制）	在一个 TM3 扩展模块上检测到临时或永久硬件错误。
TM3_ERR_BUS	04（十六进制）	在上一次与扩展模块的总线交换中检测到总线错误。

TM3_MODULE_R_ARRAY_TYPE: TM3 扩展模块读取数组类型

描述

TM3_MODULE_R_ARRAY_TYPE 是包含 0...13 个 TM3_MODULE_R_STRUCT 的数组。

TM3_MODULE_STATE: TM3 扩展模块状态代码

枚举类型介绍

TM3_MODULE_STATE 枚举数据类型包含以下值:

枚举器	值	注释
TM3_EMPTY	00 (十六进制)	无模块。
TM3_CONF_ERROR	01 (十六进制)	物理扩展模块与 SoMachine 中配置的模块不匹配。
TM3_BUS_ERROR	02 (十六进制)	在上一次与模块的交换中检测到总线错误。
TM3_OK	03 (十六进制)	上一次与此模块的总线交换成功。

节 3.6

系统功能数据类型

RTCSETDRIFT_ERROR: 检测到的 SetRTCDrift 功能错误代码

枚举类型介绍

RTCSETDRIFT_ERROR 枚举数据类型包含以下值:

枚举器	值	注释
RTC_OK	00 (十六进制)	RTC 漂移配置正确。
RTC_BAD_DAY	01 (十六进制)	未使用。
RTC_BAD_HOUR	02 (十六进制)	未使用。
RTC_BAD_MINUTE	03 (十六进制)	未使用。
RTC_BAD_DRIFT	04 (十六进制)	RTC 漂移参数超出范围。
RTC_INTERNAL_ERROR	05 (十六进制)	由于内部检测到错误而拒绝 RTC 漂移设置。

附录



附录 A

功能和功能块表示形式

概述

每个功能可以使用以下语言表示：

- IL：指令列表
- ST：结构化文本
- LD：梯形图
- FBD：功能块图
- CFC：连续功能图

本章提供功能和功能块表现形式示例，并解释如何将它们用于 IL 和 ST 语言。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
功能与功能块的区别	84
如何通过 IL 语言使用功能或功能块	85
如何通过 ST 语言使用功能或功能块	88

功能与功能块的区别

功能

功能：

- 是返回一个直接结果的 POU（程序组织单元）。
- 通过其名称（而不是通过实例）直接调用。
- 从一次调用到另一次调用不会保持原有状态。
- 可以用作其他表达式中的操作数。

示例：布尔操作符 (AND)、计算、转换 (BYTE_TO_INT)

功能块

功能块：

- 是返回一个或多个输出的 POU（程序组织单元）。
- 需要通过实例（具有专用名称和变量的功能块副本）进行调用。
- 从功能块或程序的一次调用到另一次调用，每个实例都具有持续状态（输出和内部变量）。

示例：定时器、计数器

在下面的示例中，Timer_ON 是功能块 TON 的一个实例：

```
1  PROGRAM MyProgram_ST
2  VAR
3      Timer_ON: TON; // Function Block Instance
4      Timer_RunCd: BOOL;
5      Timer_PresetValue: TIME := T#5S;
6      Timer_Output: BOOL;
7      Timer_ElapsedTime: TIME;
8  END_VAR
```

```
1  Timer_ON(
2      IN:=Timer_RunCd,
3      PT:=Timer_PresetValue,
4      Q=>Timer_Output,
5      ET=>Timer_ElapsedTime);
```

如何通过 IL 语言使用功能或功能块

一般信息

本部分介绍如何使用 IL 语言实现功能和功能块。

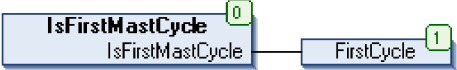
我们以功能 IsFirstMastCycle、功能 SetRTCDrift 和功能块 TON 为例来演示实现的过程。

通过 IL 语言使用功能

以下过程描述如何用 IL 语言插入一个功能：

步骤	操作
1	通过指令列表语言打开 POU 或创建新 POU。 注意： 此处未详细介绍创建 POU 的步骤。有关详细信息，请参阅添加和调用 POU (参见 <i>SoMachine, 编程指南</i>)。
2	创建功能所需的变量。
3	如果功能具有 1 个或多个输入，则使用 LD 指令开始加载第一个输入。
4	在下面插入新行，并执行以下操作： - 在操作符列（左侧字段）中键入功能的名称，或 - 使用输入助手选择功能（在上下文菜单中选择插入运算块）。
5	如果功能具有多个输入，则在使用输入助手时，会在右侧字段中使用 ??? 自动创建必需的行数。使用与输入顺序对应的适当值或变量来替换 ???。
6	插入新行，以便将功能的结果存储到适当的变量中：在操作符列（左侧字段）中键入 ST 指令，并在右侧字段中键入变量名。

要阐释该过程，请考虑下面以图形方式表示的功能 IsFirstMastCycle（不带输入参数）和功能 SetRTCDrift（带输入参数）：

功能	图形表示形式
不带输入参数： IsFirstMastCycle	
带输入参数： SetRTCDrift	

在 IL 语言中，功能名称直接用在操作符列中：

功能	SoMachine POU IL 编辑器中的表示形式
不带输入参数的功能的 IL 示例： IsFirstMastCycle	1PROGRAM MyProgram_IL 2VAR 3FirstCycle: BOOL; 4END_VAR 5 1IsFirstMastCycle STFirstCycle
带输入参数的功能的 IL 示例： SetRTCDrift	1PROGRAM MyProgram_IL 2VAR 3myDrift: SINT (-29..29) := 5; 4myDay: DAY_OF_WEEK := SUNDAY; 5myHour: HOUR := 12; 6myMinute: MINUTE; 7myDiag: RTCSETDRIFT_ERROR; 8END_VAR 9 1LDmyDrift SetRTCDriftmyDay myHour myMinute STmyDiag

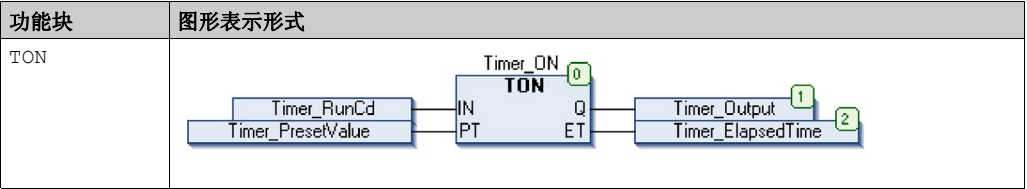
通过 IL 语言使用功能块

以下过程描述如何用 IL 语言插入一个功能块：

步骤	操作
1	通过指令列表语言打开 POU 或创建新 POU。 注意： 此处未详细介绍创建 POU 的步骤。有关详细信息，请参阅添加和调用 POU (参见 <i>SoMachine, 编程指南</i>)。

步骤	操作
2	创建功能块所需的变量（包括实例名称）。
3	使用 CAL 指令调用功能块： - 使用输入助手选择 FB（右键单击并在上下文菜单中选择插入运算块）。 - 会自动创建 CAL 指令和必要的 I/O。 每个参数 (I/O) 都是一条指令： - 输入的值通过 “:=” 进行设置。 - 输出的值通过 “=>” 进行设置。
4	在 CAL 右侧字段中，使用实例名称替换 ???。
5	使用适当的变量或立即值替换其他 ???。

要阐释该过程，请考虑下面以图形方式表示的 TON 功能块示例：



在 IL 语言中，功能块名称直接用在操作符列中：

功能块	SoMachine POU IL 编辑器中的表示形式
TON	<pre>1 PROGRAM MyProgram_IL 2 VAR 3 Timer_ON: TON; // Function Block instance declaration 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000</pre>

如何通过 ST 语言使用功能或功能块

一般信息

本部分介绍如何使用 ST 语言实现功能和功能块。
我们以功能 SetRTCDrift 和功能块 TON 为例演示实现的过程。

通过 ST 语言使用功能

以下过程描述如何用 ST 语言插入一个功能：

步骤	操作
1	通过结构化文本语言打开 POU 或创建新 POU。 注意： 此处未详细介绍创建 POU 的步骤。有关详细信息，请参阅添加和调用 POU (参见 <i>SoMachine</i> , 编程指南)。
2	创建功能所需的变量。
3	在 POU ST 编辑器 中，使用功能 ST 语言的常规语法。常规语法为： FunctionResult:= FunctionName (VarInput1, VarInput2,..VarInputx);

要阐释该过程，请考虑下面以图形方式表示的 SetRTCDrift 功能：

功能	图形表示形式
SetRTCDrift	

此功能的 ST 语言如下所示：

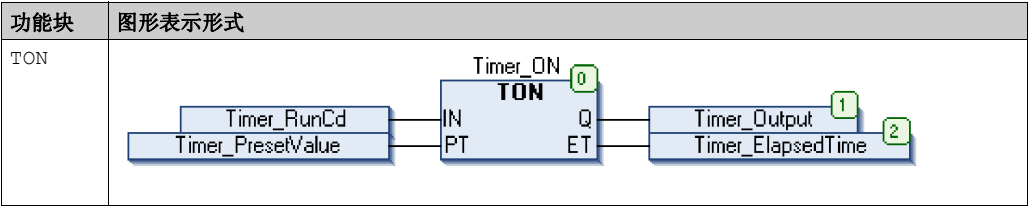
功能	SoMachine POU ST 编辑器中的表示形式
SetRTCDrift	<pre>PROGRAM MyProgram_ST VAR myDrift: SINT(-29..29) := 5; myDay: DAY_OF_WEEK := SUNDAY; myHour: HOUR := 12; myMinute: MINUTE; myRTCAdjust: RTCDRIFT_ERROR; END_VAR myRTCAdjust:= SetRTCDrift(myDrift, myDay, myHour, myMinute);</pre>

通过 ST 语言使用功能块

以下过程描述如何用 ST 语言插入一个功能块：

步骤	操作
1	通过结构化文本语言打开 POU 或创建新 POU。 注意： 此处未详细介绍创建 POU 的步骤。有关添加、声明和调用 POU 的更多信息，请参阅相关文档 (参见 <i>SoMachine, 编程指南</i>)。
2	创建功能块所需的输入和输出变量以及实例： ● 输入变量是功能块所需的输入参数 ● 输出变量接收功能块返回的值
3	在 POU ST 编辑器 中，使用功能块 ST 语言的常规语法。常规语法为： FunctionBlock_InstanceName (Input1:=VarInput1, Input2:=VarInput2, ... Ouput1=>VarOutput1, Ouput2=>VarOutput2, ...) ;

要阐释该过程，请考虑下面以图形方式表示的 TON 功能块示例：



下表显示了采用 ST 语言的功能块调用的示例：

功能块	SoMachine POU ST 编辑器中的表示形式
TON	1 PROGRAM MyProgram_ST 2 VAR 3 Timer_ON: TON; // Function Block Instance 4 Timer_RunCd: BOOL; 5 Timer_PresetValue: TIME := T#5S; 6 Timer_Output: BOOL; 7 Timer_ElapsedTime: TIME; 8 END_VAR 1 Timer_ON(2 IN:=Timer_RunCd, 3 PT:=Timer_PresetValue, 4 Q=>Timer_Output, 5 ET=>Timer_ElapsedTime);



%MW

根据 IEC 标准，%MW 表示存储器字寄存器（例如，存储器字类型的语言对象）。

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。

元件

ARRAY 元件的简称。

功能

拥有一个输入和返回一个直接结果的编程单元。但是，与 FBs 不同，它通过其名称（而不是通过实例）直接调用、不具备从一个调用到下一个调用的持久状态且可以用作其他编程表达式中的操作数。

示例：布尔 (AND) 操作符、计算、转换 (BYTE_TO_INT)

功能块

拥有一个或多个输入并返回一个或多个输出的编程单元。FBs 通过实例（具有专用名称和变量的功能块副本）进行调用，且每个实例在从一个调用到另一个调用会保持原有状态（输出和内部变量）。

示例：定时器、计数器

功能块图

控制系统的标准 IEC 61131-3 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，其中每个网络均包含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。

十六进制

(十六进制)

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。

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。

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。

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）。

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。

系统变量

用于提供控制器数据和诊断信息，并用来向控制器发送命令的变量。

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。

设备

包括子组件（如传送带和转盘等）的机器的一部分。

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。

非定位变量

没有地址的变量（请参阅定位变量）。

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。

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。

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。)

DWORD

(*双字*) 以 32 位格式进行编码的类型。

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。

FB

(*功能块*) 用于整合一组编程指令以执行特定和规范化操作 (如速度控制、间隔控制或计数) 的实用编程机制。功能块可以包含配置数据和一组内部或外部操作参数, 通常是一个或多个数据输入和输出。

GVL

(*全局变量列表*) 用于管理在 Ethernet TCP/IP Modbus 网络中的控制器之间传递的全局变量。

ID

(*标识符 / 标识*)

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。

IEEE 802.3

定义了有线 Ethernet 的物理层以及数据链路层的介质访问控制子层的 IEEE 标准集合。

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）。

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。

PLC

（可编程 Logic Controller）用于自动化制造、工业和其他机电进程的工业计算机。此外，PLCs 与普通计算机不同，因为这些计算机拥有多个输入和输出数组，并且符合冲击、振动、温度和电气干扰的更强大的规范。

POU

（程序组织单元）源代码的变量声明和相应的指令集。POUs 有助于简化软件程序、功能和功能块的模块化重用。经过声明后，POUs 便可相互使用。

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。

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。

UDINT

（*无符号双精度整数*）以 32 位格式进行编码的整数。

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。



DataFileCopy	PLC_R_SDCARD_STATUS
功能, 43	数据类型, 58
DataFileCopyError	PLC_R_STATUS
数据类型, 65	数据类型, 59
DataFileCopyLocation	PLC_R_STOP_CAUSE
数据类型, 66	数据类型, 60
ETH_R	PLC_R_TERMINAL_PORT_STATUS
系统变量, 24	数据类型, 61
ETH_R_FRAME_PROTOCOL	PLC_R_TM3_BUS_STATE
数据类型, 69	数据类型, 62
ETH_R_IP_MODE	PLC_W
数据类型, 70	系统变量, 19
ETH_R_PORT_DUPLEX_STATUS	PLC_W_COMMAND
数据类型, 71	数据类型, 63
ETH_R_PORT_LINK_STATUS	PROFIBUS_R
数据类型, 73	系统变量, 31
ETH_R_PORT_SPEED	RTCSETDRIFT_ERROR
数据类型, 74	数据类型, 80
ETH_W	SERIAL_R
系统变量, 29	系统变量, 21
ExecuteScript	SERIAL_W
功能, 46	系统变量, 22
ExecuteScriptError	SetRTCDrift
数据类型, 67	功能, 40
GetRtc	TM3_ERR_CODE
功能, 35	数据类型, 77
IsFirstMastColdCycle	TM3_GetModuleBusStatus
功能, 36	功能, 49
IsFirstMastCycle	TM3_GetModuleInternalStatus
功能, 37	功能, 50
IsFirstMastWarmCycle	TM3_MODULE_R
功能, 39	系统变量, 30
PLC_R	TM3_MODULE_R_ARRAY_TYPE
系统变量, 15	数据类型, 78
PLC_R_APPLICATION_ERROR	TM3_MODULE_STATE
数据类型, 55	数据类型, 79
PLC_R_BOOT_PROJECT_STATUS	
数据类型, 56	
PLC_R_IO_STATUS	
数据类型, 57	

功能

DataFileCopy, 43
ExecuteScript, 46
GetRtc, 35
IsFirstMastColdCycle, 36
IsFirstMastCycle, 37
IsFirstMastWarmCycle, 39
SetRTCDrift, 40
TM3_GetModuleBusStatus, 49
TM3_GetModuleInternalStatus, 50
功能与功能块的区别, 84
如何通过 IL 语言使用功能或功能块, 85
如何通过 ST 语言使用功能或功能块, 88

数据类型

DataFileCopyError, 65
DataFileCopyLocation, 66
ETH_R_FRAME_PROTOCOL, 69
ETH_R_IP_MODE, 70
ETH_R_PORT_DUPLEX_STATUS, 71
ETH_R_PORT_IP_STATUS, 72
ETH_R_PORT_LINK_STATUS, 73
ETH_R_PORT_SPEED, 74
ETH_R_RUN_IDLE, 75
ExecuteScriptError, 67
PLC_R_APPLICATION_ERROR, 55
PLC_R_BOOT_PROJECT_STATUS, 56
PLC_R_IO_STATUS, 57
PLC_R_SDCARD_STATUS, 58
PLC_R_STATUS, 59
PLC_R_STOP_CAUSE, 60
PLC_R_TERMINAL_PORT_STATUS, 61
PLC_R_TM3_BUS_STATE, 62
PLC_W_COMMAND, 63
RTCSETDRIFT_ERROR, 80
TM3_ERR_CODE, 77
TM3_MODULE_R_ARRAY_TYPE, 78
TM3_MODULE_STATE, 79

系统变量

ETH_R, 24
ETH_W, 29
PLC_R, 15
PLC_W, 19
PROFIBUS_R, 31
SERIAL_R, 21
SERIAL_W, 22
TM3_MODULE_R, 30
使用, 13
定义, 11