Modicon M241 Logic Controller

高速计数
HSC 库指南

04/2014

EIO0000001449.02

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。

目录

安全信息 7
关于本书 9
部分 I 简介 11
章 1 高速计数器概述 13
专用 I/O 概述 14
嵌入式专用 I/O 映射 16
章 2 高速计数器类型 21
选择计数器 22
Simple 类型概述 25
Main 类型概述 26
频率计类型概述 27
周期计类型概述 28
部分 II 一次性模式 29
章 3 一次性模式原理 31
一次性模式原理介绍 31
章 4 使用 Simple 类型的一次性 33
纲要图 34
一次性模式下的 Simple 类型配置 35
Simple 类型编程 36
调整参数 38
章 5 使用 Main 类型的一次性 39
纲要图 40
一次性模式下的 Main 类型单相配置 41
Main 类型编程 42
调整参数 45
部分 III 模数回路模式 47
章 6 模数回路原理 49
Modulo-loop 模式原理介绍 49
章 7 使用 Simple 类型的模数回路 53
纲要图 54
模数回路模式下的 Simple 类型配置 55
Simple 类型编程 56
调整参数 58

章 8 使用 Main 类型的模数回路 59
纲要图 60
模数回路模式下的 Main 类型单相配置 61
模数回路模式下的 Main 类型双相配置 62
Main 类型编程. 63
调整参数 66
部分 IV 自由大型模式 67
章 9 自由大型模式原理 69
自由大型模式原理介绍 70
限制管理 73
章 10 使用 Main 类型的自由大型 75
纲要图 76
自由大型模式下的 Main 类型双相配置 77
Main 类型编程. 78
调整参数 81
部分 V 事件计数 模式 83
章 11 事件计数原理 85
事件计数模式原理介绍 85
章 12 使用 Main 类型的事件计数 87
纲要图 88
事件计数模式下的 Main 类型单相配置 89
Main 类型编程. 90
调整参数 93
部分 VI 频率计类型 95
章 13 频率计原理 97
描述 97
章 14 频率计类型 99
纲要图 100
频率计类型配置 101
编程 102
部分 VII 周期计类型 105
章 15 周期计类型原理 107
描述 107

章 16 周期计类型 109
纲要图 110
沿对沿模式下的频率计类型配置 111
沿对反向沿模式下的周期计类型配置 112
编程 113
调整参数 116
部分 VIII 可选功能 117
章 17 比较功能 119
使用 **Main** 类型的比较原理 120
对使用 **Main** 类型的比较进行配置 124
外部事件配置 125
章 18 捕捉功能 127
使用 **Main** 类型的捕捉原理 128
对使用 **Main** 类型的捕捉进行配置 129
章 19 预设和启用功能 131
预设功能 132
自由大型或周期计预设条件 134
启用功能 136
附录 137
附录 A 一般信息 139
专用特性 140
有关管理和运动功能块管理的一般信息 141
附录 B 数据类型 143
EXPERT_ERR_TYPE: Type for Error Variable of EXPERT Function Block 144
EXPERT_FREQMETER_TIMEBASE_TYPE：频率计时基变量类型 145
EXPERT_HSCMAIN_TIMEBASE_TYPE：HSC Main 时基变量的类型 146
EXPERT_IMMEDIATE_ERR_TYPE：GetImmediateValue 功能块的错误变量类型 147
EXPERT_PARAMETER_TYPE：要在 EXPERT 上获取或设置的参数类型 148
EXPERT_PERIODMETER_RESOLUTION_TYPE：周期计时基变量类型 149
EXPERT_REF：EXPERT 参考值 150

附录 C 功能块 151
EXPERTGetCapturedValue：返回捕捉寄存器的内容 152
EXPERTGetDiag：提供在主体专用 I/O 功能 上检测到的错误的详细信息 154
EXPERTGetImmediateValue：读取 HSC 的计数器值. 156
EXPERTGetParam：返回主体专用 I/O 功能 的参数 158
EXPERTSetParam：调整 HSC 的参数 160
HSCMain_M241：HSC Main 功能块 162
HSCSimple_M241：HSC Simple 功能块 166
附录 D 功能和功能块表示形式 169
功能与功能块的区别 170
如何通过 IL 语言使用功能或功能块. 171
如何通过 ST 语言使用功能或功能块 175
术语表 179
索引 181

安全信息

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

本文档向您介绍 M241 Logic Controller 中提供的高速计数器 (HSC) 功能和变量。

本文档介绍 M241 HSC 库的功能和变量。

要使用本手册，您必须：

- 对 M241 有着全面了解，包括其设计、功能和在控制系统中的实现。
- 熟练使用下列 IEC 61131-3 PLC 编程语言：
  - 功能块图 (FBD)
  - 梯形图 (LD)
  - 结构化文本 (ST)
  - 指令列表 (IL)
  - 顺序功能图 (SFC)

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 编程指南	EIO0000001432 （英语）； EIO0000001433 （法语）； EIO0000001434 （德语）； EIO0000001435 （西班牙语）； EIO0000001436 （意大利语）； EIO0000001437 （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。这些关键控制功能包括紧急停止、越程停止、断电重启以及类似的安全措施。
- 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路失效问题加以考虑。
- 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。

部分 I
简介

概述

本部分提供不同功能的概述、可用模式、功能和性能。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
1	高速计数器概述	13
2	高速计数器类型	21

章 1
高速计数器概述

概述

本章主要介绍高速计数器的各种类型。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
专用 I/O 概述	14
嵌入式专用 I/O 映射	16

专用 I/O 概述

简介

M241 逻辑控制器提供：

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

M241 逻辑控制器支持以下专用功能：

功能		描述
计数器	HSC Simple	HSC 功能可以对与快速输入相连的传感器、开关等中的脉冲执行快速计数。 有关 HSC 功能的详细信息，请参阅高速计数器类型 (参见第 *21* 页)。
	HSC Main 单相	
	HSC Main 双相	
	频率计	
	周期计	
脉冲发生器	PTO *(参见 Modicon M241 Logic Controller, PTOPWM, 库指南)*	在开环模式下， PTO 功能生成脉冲串输出以控制线性单轴步进器或伺服驱动器。
	PWM *(参见 Modicon M241 Logic Controller, PTOPWM, 库指南)*	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 Modicon M241 Logic Controller, PTOPWM, 库指南)*	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：

- 当输入作为 “ 运行 / 停止 ” 输入使用时，不能由专用功能使用它。
- 当输出作为 “ 警报 ” 输出使用时，不能由专用功能使用它。

有关详细信息，请参阅内嵌功能配置 *(参见 Modicon M241 Logic Controller, 编程指南)*。

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	双击**设备树**中的**计数器**或 **Pulse_Generators** 节点。 **结果：计数器**或 **Pulse_Generators** 功能窗口随即显示：
2	双击**值**并选择要分配的功能类型。 **结果：**专用功能的参数随即显示。

常规 I/O 中的专用功能 I/O

常规 I/O 中的专用功能 I/O：

- 可以通过标准存储器变量读取输入，即使配置为专用功能。
- 如果输入已经配置为“运行 / 停止”输入，则无法将其配置为专用功能。
- 如果输出已经配置为警报，则无法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。
- 未被专用功能使用的所有 I/O 都可作为任何其他 I/O 使用。

在专用功能（锁存、HSC 等）中使用输入时，积分器过滤器替换为抗跳动过滤器。在专用配置屏幕中配置过滤器值。

嵌入式专用 I/O 映射

M241 上专用功能的输入映射

M241 专用输入		快速输入								常规输入					
		I0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13
事件锁存 0	输入	M													
事件锁存 1	输入		M												
事件锁存 2	输入			M											
事件锁存 3	输入				M										
事件锁存 4	输入					M									
事件锁存 5	输入						M								
事件锁存 6	输入							M							
事件锁存 7	输入								M						
HSC Simple 0	输入	M													
HSC Simple 1	输入		M												
HSC Simple 2	输入			M											
HSC Simple 3	输入				M										
HSC Simple 4	输入					M									
HSC Simple 5	输入						M								
HSC Simple 6	输入							M							
HSC Simple 7	输入								M						
HSC Main 0	输入 A	M													
	输入 B/EN		C												
	SYNC			C											
	CAP				C							C			
	Reflex0														
	Reflex1														

M 强制
C 取决于配置

M241 专用输入		快速输入								常规输入					
		I0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13
HSC Main 1	输入 A					M									
	输入 B/EN						C								
	SYNC							C							
	CAP								C						C
	Reflex0														
	Reflex1														
频率计 0/ 周期计 0	信号	M													
	EN		C												
频率计 1/ 周期计 1	信号					M									
	EN						C								
PWM 0/ 频率发生器 0	输出														
	EN				C							C			
	SYNC										C				
PWM 1/ 频率发生器 1	输出														
	EN								C						C
	SYNC													C	
PTO0	REF （起点）									C					
	INDEX （接近）										C				
	PROBE				C							C			
	输出 A/CW/ 脉冲														
	输出 B/CCW/ 方向														
PTO1	REF （起点）												C		
	INDEX （接近）													C	
	PROBE							C							C
	输出 A/CW/ 脉冲														
	输出 B/CCW/ 方向														

M 强制
C 取决于配置

M241 上专用功能的输出映射

M241 专用输出		快速输出				常规输出					
		Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9
HSC Main 0	输入 A										
	输入 B/EN										
	SYNC										
	CAP										
	反射 0	C				C					
	反射 1		C				C				
HSC Main 1	输入 A										
	输入 B/EN										
	SYNC										
	CAP										
	反射 0			C				C			
	反射 1				C				C		
频率计 0/ 周期计 0	信号										
	EN										
频率计 1/ 周期计 1	信号										
	EN										
PWM 0/ 频率发生器 0	输出	M				C					
	EN										
	SYNC										
PWM 1/ 频率发生器 1	输出			M				C			
	EN										
	SYNC										
PTO0	REF （起点）										
	INDEX （接近）										
	PROBE										
	输出 A/CW/ 脉冲	M				C					
	输出 B/CCW/ 方向		M				C				
M 强制 **C** 取决于配置											

M241 专用输出		快速输出				常规输出					
		Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9
PTO 1	REF （起点）										
	INDEX （接近）										
	PROBE										
	输出 A/CW/ 脉冲			M				C			
	输出 B/CCW/ 方向				M				C		
M 强制 **C** 取决于配置											

I/O 摘要

IO 摘要窗口显示专用功能使用的 I/O。

要显示 **IO 摘要**窗口，请执行以下步骤：

步骤	操作
1	在**设备树**选项卡中，右键单击 **MyController** 节点并选择 **IO 摘要**。

“IO 摘要 ” 窗口示例：

章 2
高速计数器类型

概述

本章概述了各种类型的高速计数器。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
选择计数器	22
Simple 类型概述	25
Main 类型概述	26
频率计类型概述	27
周期计类型概述	28

选择计数器

概述

开始 HSC 配置时，请根据您正在使用的传感器类型和应用需求选择计数器。

在**计数器**编辑器中，从提供以下 5 种类型计数器的列表中选择**计数功能**（有关详细信息，请参阅计数功能 *(参见 Modicon M241 Logic Controller, 编程指南)*）：

- **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。

对于每个计数器功能块，**实例名称**由 SoMachine 自动提供。**实例名称**可编辑，您可以使用它来定义实例名称。但是，不管**实例名称**由软件还是用户定义，请使用与**计数器**编辑器中的定义相同的实例名称，作为对处理计数器的功能块的输入。

类型和模式矩阵

下表介绍了不同的可用类型和模式：

类型 模式	HSC Simple	HSC Main 单相	HSC Main 双相	频率计	周期计
一次性	X	X	–	–	–
模数回路	X	X	X	–	–
事件计数	–	X	–	–	–
自由大型	–	–	X	–	–
沿对沿	–	–	–	–	X
沿对反向沿	–	–	–	–	X

HSCSimple

下表概述了根据请求的模式在 **HSC Simple** 类型中可用的规格：

功能	功能	
	一次性模式	**模数回路模式**
计数模式	减计数	加计数
通过 HSC 物理输入启用	否	否
通过 HSC 物理输入同步 / 预设	否	无 / 有
比较功能	否	否

功能	功能	
	一次性模式	模数回路模式
捕捉功能	否	否
配置调节	–	–

HSC Main 单相

下表概述了根据请求的模式在 **HSC Main 单相**类型中可用的规格：

功能	功能		
	一次性模式	模数回路模式	事件计数模式
计数模式	减计数	加计数	在给定的时基期间进行脉冲计数
通过 HSC 物理输入启用	是	是	否
通过 HSC 物理输入同步 / 预设	是	是	是
比较功能	是，4 个阈值，2 个输出和 4 个事件	是，4 个阈值，2 个输出和 4 个事件	否
捕捉功能	是，1 个捕捉寄存器	是，1 个捕捉寄存器	否
配置调节	停止事件	–	时基

HSC Main 双相

下表概述了根据请求的模式在 **HSC Main 双相**类型中可用的规格：

功能	功能	
	模数回路模式	自由大型模式
计数模式	加 / 减计数 脉冲 / 方向 积分	加 / 减计数 脉冲 / 方向 积分
通过 HSC 物理输入启用	否	否
通过 HSC 物理输入同步 / 预设	是	是
比较功能	是，4 个阈值，2 个输出和 4 个事件	是，4 个阈值，2 个输出和 4 个事件
捕捉功能	是，1 个捕捉寄存器	是，1 个捕捉寄存器
配置调节	–	限制管理

频率计

下表概述了在**频率计**类型中可用的规格：

功能	功能
计数模式	脉冲频率 （以 Hz 为单位）每个时基值 （10、 100 或 1000 毫秒）更新一次。
通过 HSC 物理输入启用	是
通过 HSC 物理输入同步 / 预设	否
比较功能	否
捕捉功能	否
配置调节	–

周期计

下表概述了根据请求的模式在**周期计**类型中可用的规格：

功能	功能	
	沿对沿模式	沿对反向沿模式
计数模式	使用可配置的精度 （1、 100 或 1000 微秒）对持续时间进行计数。	使用可配置的精度 （1、 100 或 1000 微秒）对持续时间进行计数。
通过 HSC 物理输入启用	是	是
通过 HSC 物理输入同步 / 预设	否	无 / 有
比较功能	否	否
捕捉功能	否	否
配置调节	精度 超时	精度 超时

Simple 类型概述

概述

Simple 类型是单个输入计数器。

计数器上的所有操作 （启用、同步）以及所有触发的动作 （达到计数值时）都在任务环境中执行。

使用 **Simple** 类型，无法触发事件或反射输出。

Simple 类型模式

Simple 类型支持 2 个可配置的计数模式 （仅限于单相脉冲）：

One-shot (参见第 *33* 页)：在此模式下，对于应用到 A 输入的每个脉冲，计数器当前值寄存器都会 （从用户定义的值）递减，直到计数器达到 0。

Modulo-loop (参见第 *53* 页)：在此模式下，计数器从 0 计数到用户定义的模数值，然后返回到 0 并重新启动计数，周而复始，反复执行。

性能

如果跳动过滤器值为 0.005 毫秒 （缺省配置值），则**专用 I/O** 接口上允许的最大频率是 100 kHz。如果跳动过滤器值为 0.002 毫秒，则最大频率为 200 kHz。

Main 类型概述

概述

Main 类型的计数器最多可使用 4 路快速输入和 2 路反射输出。M241 Logic Controller 最多可采用 2 个 **Main** 类型的高速计数器。

Main 类型模式

Main 类型支持单相 （1 个输入）或双相 （2 个输入）脉冲上的以下计数模式：

一次性 (参见第 *39* 页)：在此模式下， A 输入上每应用一次脉冲，计数器当前值寄存器就会 （从用户定义的值）递减一次，直到计数器达到 0。

模数回路 (参见第 *59* 页)：在此模式下，计数器从 0 计数到用户定义的模数值，然后返回到 0 并重新启动计数，周而复始，反复执行。反过来，计数器从模数值减计数到 0，然后预设为模数值，再重新启动计数。

自由大型 (参见第 *75* 页)：在此模式下，计数器的行为类似于大范围加和减计数器。

事件计数 (参见第 *87* 页)：在此模式下，计数器对在用户配置的时基期间接收的事件数进行累计。

可选功能

可根据选择的模式配置可选功能：

- 用于操作计数器 （启用、预设）或捕捉当前计数值的的硬件输入
- 最多 4 个阈值
- 最多 4 个事件 （阈值占 1 个）可与外部任务进行关联
- 最多 2 个反射输出

性能

如果跳动过滤器值为 0.005 毫秒 （缺省配置值），则**专用 I/O** 接口上允许的最大频率是 100 kHz。如果跳动过滤器值为 0.002 毫秒，则最大频率为 200 kHz。

频率计类型概述

概述

频率计类型是一种计数器，最多使用 2 路快速输入。 M241 Logic Controller 最多可以具有 2 个**频率计**类型的高速计数器。

频率计类型模式

频率计类型支持以下计数模式：

频率计 (参见第 *99* 页)：在此模式下，计数器测量事件的频率。频率是每秒的事件数 (Hz)。

性能

专用 I/O 接口上允许的最大频率是 200 kHz。

周期计类型概述

概述

周期计类型是一种计数器，最多使用 2 路快速输入。 M241 Logic Controller 最多可以具有 2 个**周期计**类型的高速计数器。

周期计类型模式

周期计类型支持以下计数模式：

周期计 (参见第 *109* 页)：**周期计**模式可用来：

- 确定事件的持续时间
- 确定两个事件之间的时间间隔
- 设置并测量过程的执行时间

性能

专用 I/O 接口上允许的最大频率是 200 kHz。

部分 II
一次性模式

概述

本部分介绍在**一次性**模式下如何使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
3	**一次性**模式原理	31
4	使用 **Simple** 类型的**一次性**	33
5	使用 **Main** 类型的**一次性**	39

章 3
一次性模式原理

一次性模式原理介绍

概述

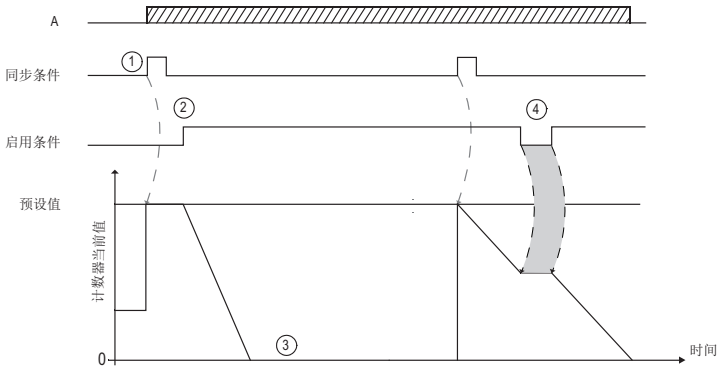
计数器由同步跳变沿激活，同时加载预设值。

启用计数时，每个应用到输入的脉冲都递减当前值。计数器在其当前值达到 0 时停止。

即使新的脉冲应用到输入，计数器值仍然保持为 0。

要再次激活计数器，需要新的同步。

原理图



下表介绍上图的各个阶段：

阶段	操作
1	在同步条件的上升沿上，预设值被加载到计数器 （不考虑当前值），并且计数器被激活。
2	当启用条件 = 1 时，当前的计数器值依据输入 A 上的各个脉冲递减，直至达到 0。
3	计数器等待同步条件的下一个上升沿出现。 **注：**此时，输入 A 上的脉冲对计数器不起作用。
4	当启用条件 = 0 时，计数器忽略来自输入 A 的脉冲，并保留其当前值，直到启用条件再次 = 1。在启用输入的上升沿，计数器从保留的值开始恢复输入 A 的脉冲的计数。

注意：启用和同步条件取决于配置。启用 (参见第 *136* 页) 和预设 (参见第 *132* 页) 功能中对这些条件进行了介绍。

章 4
使用 Simple 类型的一次性

概述

本章介绍如何在使用 **Simple** 类型的**一次性**模式下实现高速计数器。

本章包含了哪些内容？

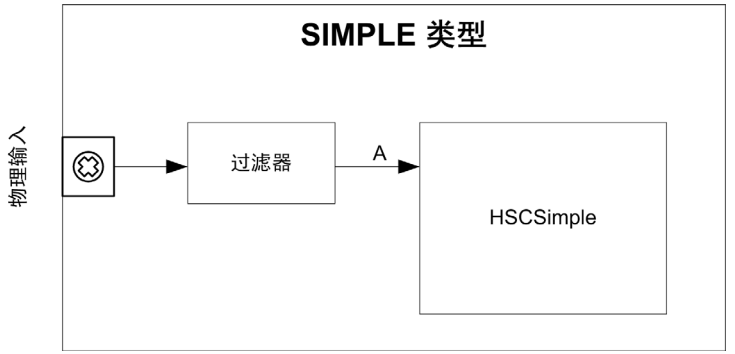
本章包含了以下主题：

主题	页
纲要图	34
一次性模式下的 Simple 类型配置	35
Simple 类型编程	36
调整参数	38

纲要图

纲要图

下图概述了**一次性**模式下的 **Simple** 类型：



A 是 `High Speed Counter` 的计数输入。**一次性**模式的 **Simple** 计数类型仅减计数。

一次性模式下的 Simple 类型配置

过程

按照以下过程配置**一次性**模式下的 **Simple** 类型：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Simple**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**一次性**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	输入**范围** →**预设**参数的值以设置计数初始值。

Simple 类型编程

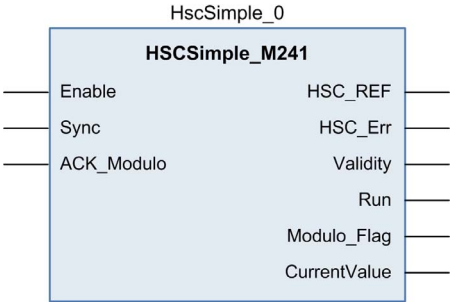
概述

Simple 类型始终由 HSCSimple (参见第 *166* 页) 功能块进行管理。

注意：如果使用 HSCSimple 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCSimple 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**HSC** →**HSCSimple_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Simple** 类型实例名称 （在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量** →**<MyController>** →**PLC** → **IoConfig_Globals**。



I/O 变量用法

下表介绍了功能块的不同引脚在**一次性**模式中的使用方法。

下表对输入变量进行了描述：

输入	类型	注释
Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
Sync	BOOL	在上升沿预设和启动计数器
ACK_Modulo	BOOL	未使用

下表对输出变量进行了描述：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 用作**管理**功能块的输入。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	计数器正在运行时，设为 1。 当 CurrentValue 达到 0 时切换为 0。需要同步才能重新启动计数器。
Modulo_Flag	BOOL	不相关
CurrentValue	DWORD	计数器的当前计数值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可以读取或修改这些参数：

参数	描述
EXPERT_PRESET	获取或设置 HSC 的预设值

章 5
使用 Main 类型的一次性

概述

本章介绍如何在使用 **Main** 类型的**一次性**模式下实现高速计数器。

本章包含了哪些内容？

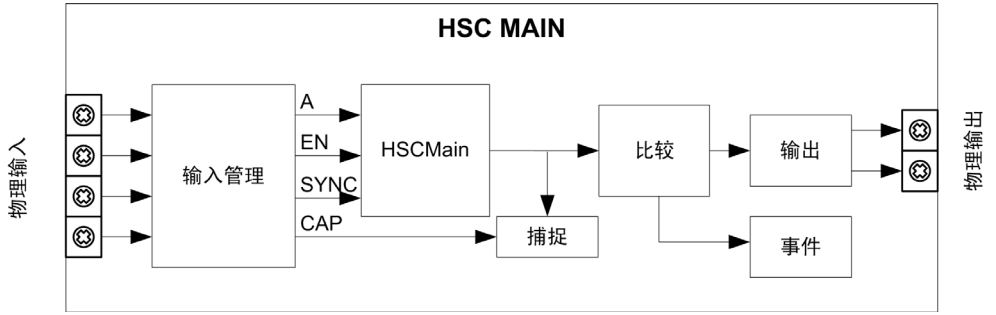
本章包含了以下主题：

主题	页
纲要图	40
一次性模式下的 Main 类型单相配置	41
Main 类型编程	42
调整参数	45

纲要图

纲要图

下图概述了**一次性**模式下的 **Main** 类型：



A 是计数器的计数输入。

EN 是计数器的启用输入。

CAP 是计数器的捕捉输入。

SYNC 是计数器的同步输入。

可选功能

除了**一次性**模式， **Main** 类型还可以提供以下功能：

- 比较功能 (参见第 *119* 页)
- 捕捉功能 (参见第 *127* 页)
- 预设功能 (参见第 *132* 页)
- 启用功能 (参见第 *136* 页)

一次性模式下的 Main 类型单相配置

过程

按照以下过程配置**一次性**模式下的 **Main** 类型单相：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Main 单相**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**一次性**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	输入**范围** →**预设**参数的值以设置计数初始值。
7	还可以选择启用以下功能： ● 比较功能 (参见第 *119* 页) ● 捕捉功能 (参见第 *127* 页) ● 预设功能 (参见第 *132* 页) ● 启用功能 (参见第 *136* 页)
8	还可以选择将**事件** →**停止事件**参数的值设置为**是**，以启用外部事件功能 (参见第 *125* 页)。

Main 类型编程

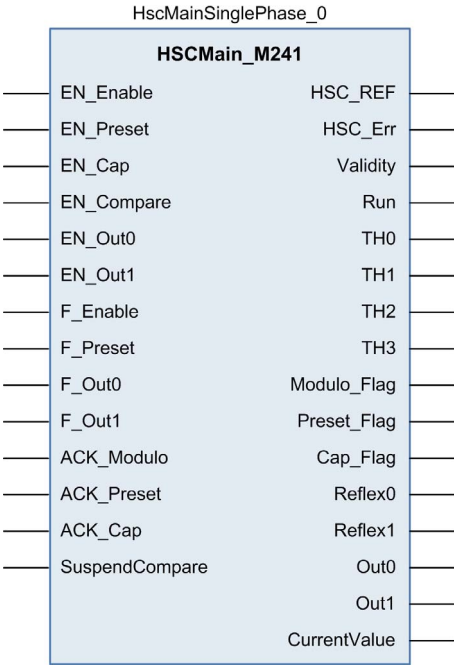
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**HSC** →**HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称（在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量** →**<MyController>** →**PLC** → **IoConfig_Globals**。



I/O 变量用法

下表介绍了功能块的不同引脚在**一次性**模式中的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见第 *136* 页) 启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 或 **REF** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *132* 页) 预设计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入。
EN_Compare	BOOL	TRUE = 启用比较器操作 (参见第 *119* 页) （使用阈值 0、 1、 2、 3）： ● 基本比较 （TH0、 TH1、 TH2、 TH3 输出位） ● 反射 （Reflex0、 Reflex1 输出位） ● 事件 （在超出阈值时触发外部任务）
EN_Out0	BOOL	TRUE = 启用物理输出 Out_R0 回显 Reflex0 值 （如果已配置）。
EN_Out1	BOOL	TRUE = 启用物理输出 Out_R1 回显 Reflex1 值 （如果已配置）。
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿预设和启动计数器。
F_Out0	BOOL	TRUE = 强制 Out_R0 为 1 （如果已配置 Reflex0）。
F_Out1	BOOL	TRUE = 强制 Out_R1 为 1 （如果已配置 Reflex1）。
ACK_Preset	BOOL	在上升沿，复位 Preset_Flag。
ACK_Cap	BOOL	在上升沿，复位 Cap_Flag。
SuspendCompare	BOOL	TRUE = 比较结果已挂起： ● TH0、 TH1、 TH2、 TH3、 Reflex0、 Reflex1、 Out0、 Out1 块输出位保持其上一个值。 ● 硬件输出 0、 1 保持各自的上一个值。 ● 事件被掩蔽。 **注意：**在设置 SuspendCompare 时， EN_Compare、 EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 用作**管理**功能块的输入。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	TRUE = 计数器正在运行。 CurrentValue 达到 0 时 Run 位切换为 0。
TH0	BOOL	当 CurrentValue > 阈值 0 时，设为 1 (参见第 *119* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *119* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *119* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *119* 页)。
Preset_Flag	BOOL	在计数器预设 (参见第 *132* 页) 时，设为 1。
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器后，设为 1。 在进行新的捕捉之前，必须先复位此标志。
Reflex0	BOOL	Reflex0 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Reflex1	BOOL	Reflex1 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Out0	BOOL	物理输出 Out_R0 的状态 （如果已配置 Reflex0）。
Out1	BOOL	物理输出 Out_R1 的状态 （如果已配置 Reflex1）。
CurrentValue	DINT	计数器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可以读取或修改这些参数：

参数	描述
EXPERT_PRESET	获取或设置 HSC 的预设值
EXPERT_THRESHOLD0	获取或设置 HSC 的阈值 0 值
EXPERT_THRESHOLD1	获取或设置 HSC 的阈值 1 值
EXPERT_THRESHOLD2	获取或设置 HSC 的阈值 2 值
EXPERT_THRESHOLD3	获取或设置 HSC 的阈值 3 值
EXPERT_REFLEX0	获取或设置专用功能的输出 0 反射模式
EXPERT_REFLEX1	获取或设置专用功能的输出 0 反射模式

部分 III
模数回路模式

概述

本章介绍在**模数回路**模式下如何使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
6	**模数回路**原理	49
7	使用 **Simple** 类型的**模数回路**	53
8	使用 **Main** 类型的**模数回路**	59

章 6
模数回路原理

Modulo-loop 模式原理介绍

概述

模数回路模式可用于对一系列移动对象执行的重复操作，如包装和贴标签等应用。

原理

在 同步条件 (参见第 *132* 页) 的上升沿，计数器被激活，并且当前值复位为 0。

启用 (参见第 *136* 页) 计数后：

递增方向： 计数器在达到模数值之前一直递增。在下一个脉冲，计数器复位为 0，模数标志设置为 1 并继续计数。

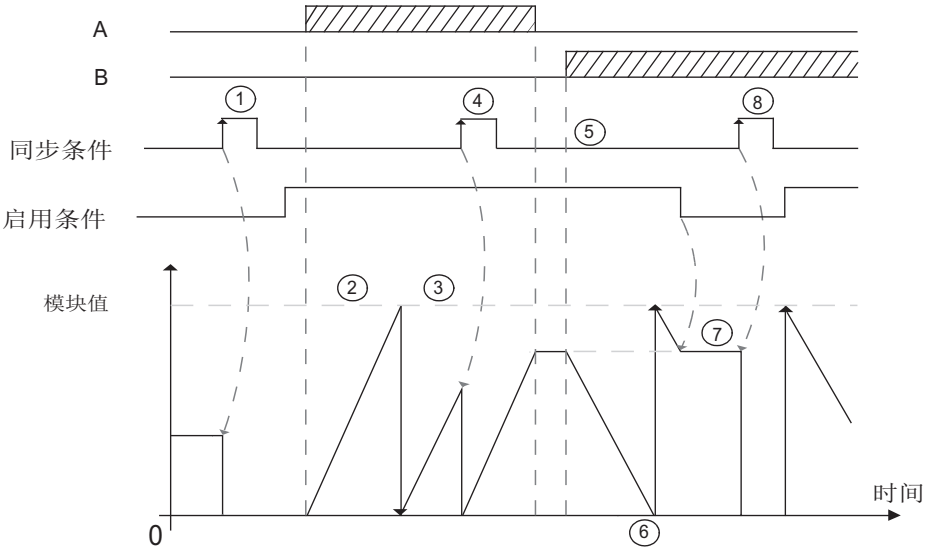
递减方向： 计数器在达到 0 之前一直递减。在下一个脉冲，计数器设置为模数值，模数标志设置为 1 并继续计数。

输入模式

下表介绍可用的 8 种输入模式：

输入模式	注释
A = 加计数， B = 减计数	缺省模式 计数器在 A 上递增，在 B 上递减。
A = 脉冲， B = 方向	如果 A 上存在上升沿且 B 为 TRUE，则计数器递减。 如果 A 上存在上升沿且 B 为 FALSE，则计数器递增。
正常积分 X1	物理编码器始终提供 2 个信号 90° 移位，首次实现计数器可以计数脉冲并检测方向： ● X1：通过编码器循环进行 1 次计数 ● X2：通过编码器循环进行 2 次计数 ● X4：通过编码器循环进行 4 次计数
正常积分 X2	
正常积分 X4	
反向积分 X1	
反向积分 X2	
反向积分 X4	

加减计数原理图

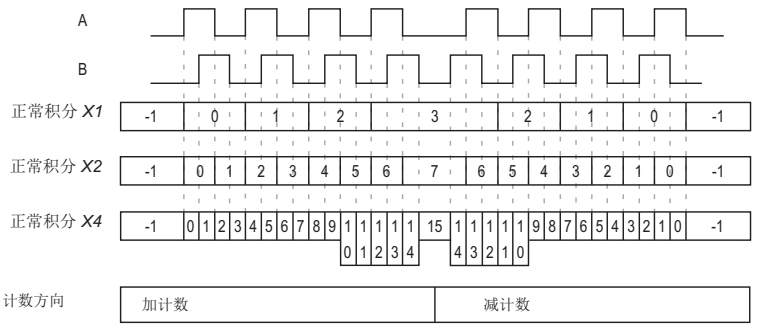


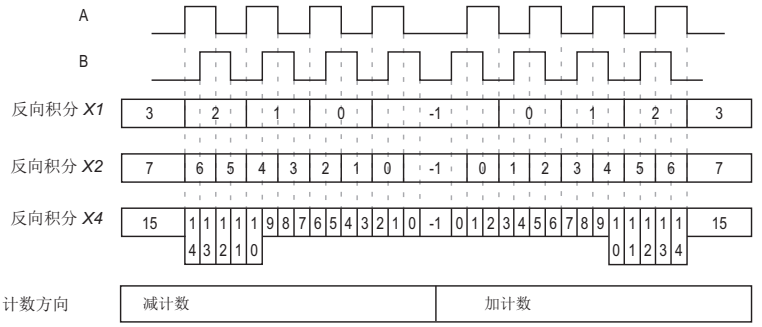
阶段	操作
1	在同步条件的上升沿，当前值复位为 0 并激活计数器。
2	当启用条件 = 1 时， A 上的每个脉冲会递增计数器值。
3	计数器达到模数值减 1 时，计数器在下一个脉冲返回 0，并继续计数。 Modulo_Flag 设置为 1。
4	在同步条件的上升沿，当前计数器值复位为 0。
5	当启用条件 = 1 时， B 上的每个脉冲会递减计数器。
6	计数器达到 0 时，计数器在下一个脉冲返回模数值减 1，并继续计数。
7	启用条件 = 0 时，忽略输入上的脉冲。
8	在同步条件的上升沿，当前计数器值复位为 0。

注意：启用和同步条件取决于配置。启用 (参见第 *136* 页) 和预设 (参见第 *132* 页) 功能中对这些条件进行了介绍。

积分原理图

编码器信号根据选定的输入模式进行计数，如下所示：





章 7
使用 Simple 类型的模数回路

概述

本章介绍如何在使用 **Simple** 类型的**模数回路**模式下实现高速计数器。

本章包含了哪些内容？

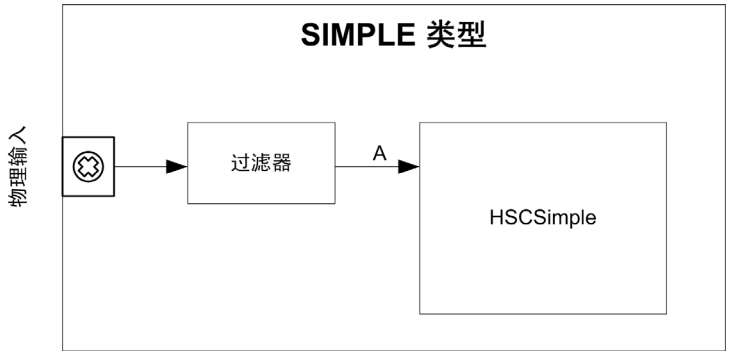
本章包含了以下主题：

主题	页
纲要图	54
模数回路模式下的 Simple 类型配置	55
Simple 类型编程	56
调整参数	58

纲要图

纲要图

下图概述了**模数回路**模式下的 **Simple** 类型：



模数回路模式的 **Simple** 计数类型仅加计数。

模数回路模式下的 Simple 类型配置

过程

按照以下过程配置**模数回路**模式下的 **Simple** 类型：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Simple**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**模数回路**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	输入**范围** →**模数**参数的值以设置计数模数值。

Simple 类型编程

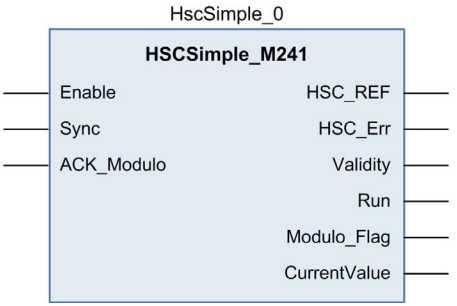
概述

Simple 类型始终由 HSCSimple (参见第 *166* 页) 功能块进行管理。

注意：如果使用 HSCSimple 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCSimple 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**HSC** →**HSCSimple_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Simple** 类型实例名称 （在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量** →**<MyController>** →**PLC** → **IoConfig_Globals**。



I/O 变量用法

下表介绍了功能块的不同引脚在**模数回路**模式中的使用方法。

下表对输入变量进行了描述：

输入	类型	注释
Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
Sync	BOOL	在上升沿复位和启动计数器。
ACK_Modulo	BOOL	在上升沿，复位 Modulo_Flag。

下表对输出变量进行了描述：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 用作**管理**功能块的输入。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	不相关
Modulo_Flag	BOOL	计数器转过模数时，设为 1。
CurrentValue	DWORD	计数器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可以读取或修改这些参数：

参数	描述
EXPERT_MODULO	获取或设置 HSC 的模数值

章 8
使用 Main 类型的模数回路

概述

本章介绍如何在使用 **Main** 类型的**模数回路**模式下实现高速计数器。

本章包含了哪些内容？

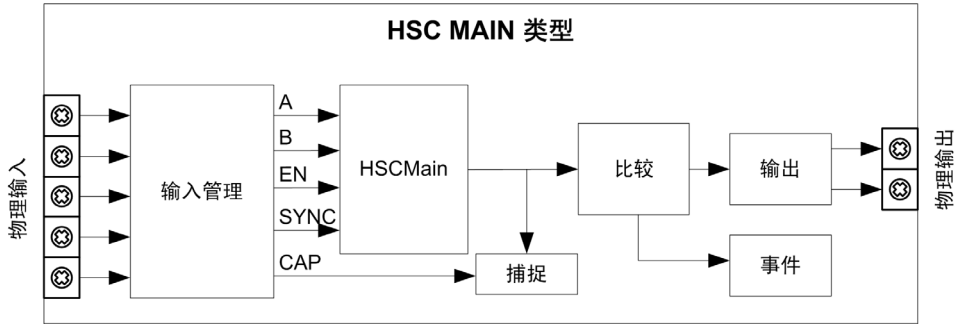
本章包含了以下主题：

主题	页
纲要图	60
模数回路模式下的 Main 类型单相配置	61
模数回路模式下的 Main 类型双相配置	62
Main 类型编程	63
调整参数	66

纲要图

纲要图

下图概述了**模数回路**模式下的 **Main** 类型：



A 和 B 是计数器的计数输入。

EN 是计数器的启用输入。

CAP 是计数器的捕捉输入。

SYNC 是计数器的同步输入。

可选功能

除了**模数回路**模式， **Main** 类型还可以提供以下功能：

- 比较功能 (参见第 *119* 页)
- 捕捉功能 (参见第 *127* 页)
- 预设功能 (参见第 *132* 页)
- 启用功能 (参见第 *136* 页)

模数回路模式下的 Main 类型单相配置

过程

按照以下过程配置**模数回路**模式下的 **Main** 类型单相：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Main 单相**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**模数回路**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	输入**范围** →**模数**参数的值以设置计数模数值。
7	还可以选择启用以下功能： ● 比较功能 (参见第 *119* 页) ● 捕捉功能 (参见第 *127* 页) ● 预设功能 (参见第 *132* 页) ● 启用功能 (参见第 *136* 页)

模数回路模式下的 Main 类型双相配置

过程

按照以下过程配置**模数回路**模式下的 **Main** 类型双相：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Main 双相**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**模数回路**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	设置**计数输入** →**B 输入** →**跳动过滤器**参数的值。
7	输入**范围** →**模数**参数的值以设置计数模数值。
8	还可以选择启用以下功能： ● 比较功能 (参见第 *119* 页) ● 捕捉功能 (参见第 *127* 页) ● 预设功能 (参见第 *132* 页)

Main 类型编程

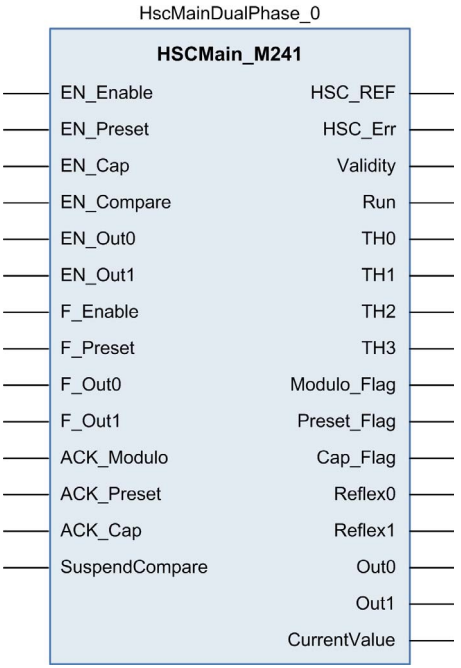
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →HSC →HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称 （在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量 →<MyController> →PLC → IoConfig_Globals**。



I/O 变量用法

下表介绍了功能块的不同引脚在**模数回路**模式中的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见第 *136* 页) 启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 或 **REF** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *132* 页) 预设计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入。
EN_Compare	BOOL	TRUE = 使用阈值 0、 1、 2、 3 启用比较器操作 (参见第 *119* 页)： ● 基本比较 （TH0、 TH1、 TH2、 TH3 输出位） ● 反射 （Reflex0、 Reflex1 输出位） ● 事件 （在超出阈值时触发外部任务）
EN_Out0	BOOL	TRUE = 启用物理输出 Out_R0 回显 Reflex0 值 （如果已配置）。
EN_Out1	BOOL	TRUE = 启用物理输出 Out_R1 回显 Reflex1 值 （如果已配置）。
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿复位和启动计数器。
F_Out0	BOOL	TRUE = 强制 Out_R0 为 1 （如果 Reflex0 已配置）。
F_Out1	BOOL	TRUE = 强制 Out_R1 为 1 （如果 Reflex1 已配置）。
ACK_Modulo	BOOL	在上升沿，复位 Modulo_Flag。
ACK_Preset	BOOL	在上升沿，复位 Preset_Flag。
ACK_Cap	BOOL	在上升沿，复位 Cap_Flag。
SuspendCompare	BOOL	TRUE = 比较结果已挂起： ● TH0、 TH1、 TH2、 TH3、 Reflex0、 Reflex1、 Out0、 Out1 块输出位保持其上一个值。 ● 硬件输出 0、 1 保持各自的上一个值。 ● 事件被掩蔽。 **注意：**在设置 SuspendCompare 时， EN_Compare、 EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 用作**管理**功能块的输入。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	TRUE = 计数器正在运行。 CurrentValue 达到 0 时 Run 位切换为 0。 需要同步才能重新启动计数器。
TH0	BOOL	当 CurrentValue > 阈值 0 时，设为 1 (参见第 *119* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *119* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *119* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *119* 页)。
Modulo_Flag	BOOL	计数器回转到模数或 0 时，设为 1。
Preset_Flag	BOOL	在计数器预设 (参见第 *132* 页) 时，设为 1。
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器后，设为 1。 在进行新的捕捉之前，必须先复位此标志。
Reflex0	BOOL	Reflex0 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Reflex1	BOOL	Reflex1 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Out0	BOOL	物理输出 Out_R0 的状态 （如果已配置 Reflex0）。
Out1	BOOL	物理输出 Out_R1 的状态 （如果已配置 Reflex1）。
CurrentValue	DINT	计数器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *158* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出了来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可对这些参数进行修改：

参数	描述
EXPERT_MODULO	获取或设置 HSC 的模数值
EXPERT_THRESHOLD0	获取或设置 HSC 的阈值 0 值
EXPERT_THRESHOLD1	获取或设置 HSC 的阈值 1 值
EXPERT_THRESHOLD2	获取或设置 HSC 的阈值 2 值
EXPERT_THRESHOLD3	获取或设置 HSC 的阈值 3 值
EXPERT_REFLEX0	获取或设置专用功能的输出 0 反射模式
EXPERT_REFLEX1	获取或设置专用功能的输出 1 反射模式

部分 IV
自由大型模式

概述

本部分介绍在**自由大型**模式下如何使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
9	自由大型模式原理	69
10	使用 Main 类型的自由大型	75

章 9
自由大型模式原理

概述

本章介绍**自由大型** 模式的原理。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
自由大型模式原理介绍	70
限制管理	73

自由大型模式原理介绍

概述

对于必须了解每个部件传入位置的轴监控或轴贴标情况，可以使用**自由大型**模式。

原理

在**自由大型**模式中，模块在行为上类似于标准的加减计数器。

计数启用 (参见第 *136* 页) 后，计数器计数方式如下：
递增方向： 计数器递增。
递减方向： 计数器递减。

计数器通过预设跳变沿 (参见第 *134* 页) 激活，该跳变沿将加载预设值。

当前计数器通过捕捉 (参见第 *127* 页) 功能存储在捕捉寄存器。

如果计数器达到计数限制，则计数器会根据限制管理 (参见第 *73* 页) 配置做出反应。

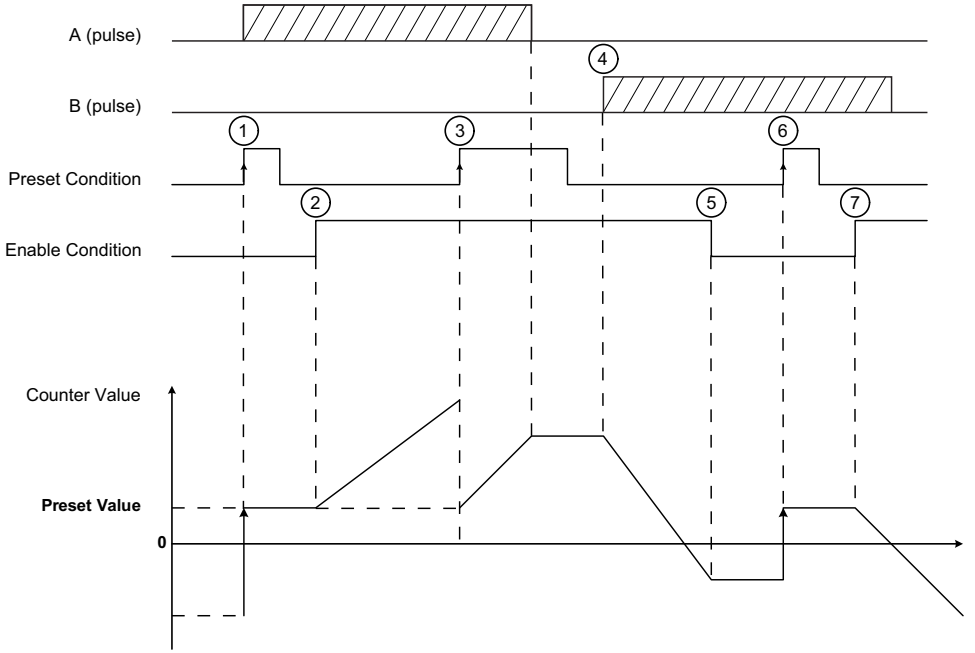
输入模式

下表介绍可用的 8 种输入模式：

输入模式	注释
A = 加计数， B = 减计数	缺省模式 计数器在 A 上递增，在 B 上递减。
A = 脉冲， B = 方向	如果 A 上存在上升沿且 B 为 TRUE，则计数器递减。 如果 A 上存在上升沿且 B 为 FALSE，则计数器递增。
正常积分 X1	物理编码器始终提供 2 个信号 90° 移位，首次实现计数器可以计数脉冲并检测方向： ● X1：通过编码器循环进行 1 次计数 ● X2：通过编码器循环进行 2 次计数 ● X4：通过编码器循环进行 4 次计数
正常积分 X2	
正常积分 X4	
反向积分 X1	
反向积分 X2	
反向积分 X4	

加减计数原理图

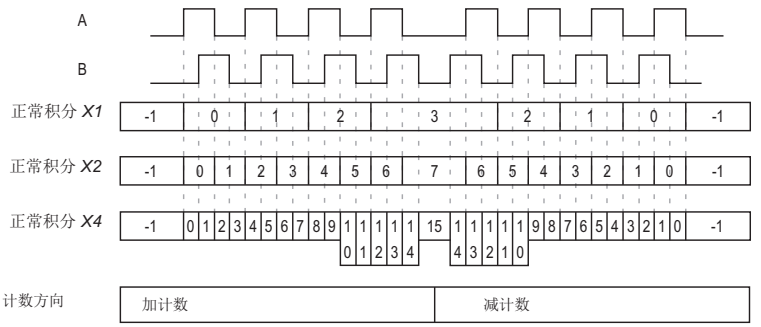
下图显示 **A = 加计数， B = 减计数**模式：

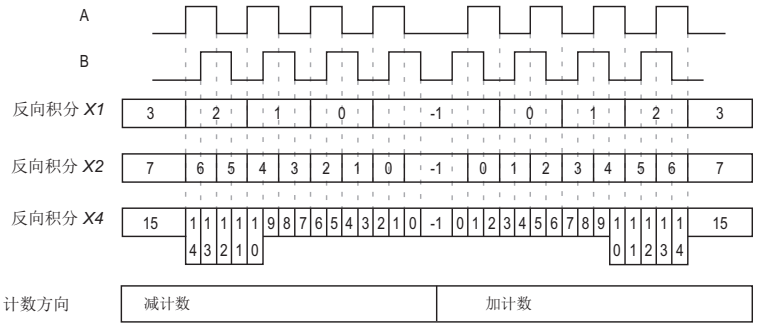


阶段	操作
1	在预设条件的上升沿上，当前值设为预设值，并激活计数器。
2	当启用条件 = 1 时， A 上的每个脉冲会递增计数器值。
3	在预设条件的上升沿上，当前值设为预设值。
4	当启用条件 = 1 时， B 上的每个脉冲会递减计数器值。
5	当启用条件 = 0，忽略 A 或 B 上的脉冲。
6	在预设条件的上升沿上，当前值设为预设值。
7	当启用条件 = 1 时， B 上的脉冲会递减计数器值。

积分原理图

编码器信号根据选定的输入模式进行计数，如下所示：





限制管理

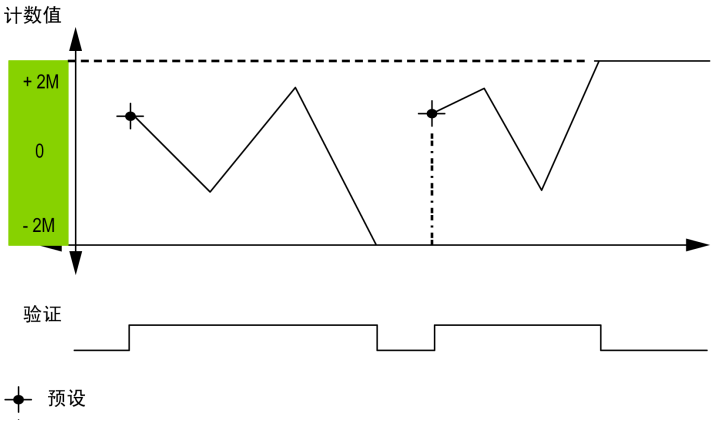
概述

达到计数器限制时，计数器根据配置可以执行 2 种行为：

- 锁定限制
- 回转

锁定限制

在计数器溢出或下溢的情况下：当前计数器值保持为限制值， Validity 位变为 0， `Error` 位指示此检测到的错误，直到再次预设计数器。



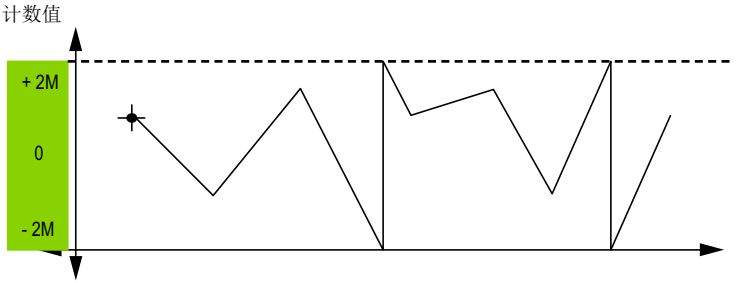
2M 值如下计算：

- +2M = 2 (exp 31) -1
- -2M = -2 (exp 31)

回转

在计数器溢出或下溢的情况下，当前计数器值会自动变为相反限制值。

`Modulo_Flag` 输出设置为 1。



章 10
使用 Main 类型的自由大型

概述

本章介绍如何在使用 **Main** 类型的**自由大型**模式下实现高速计数器。

本章包含了哪些内容？

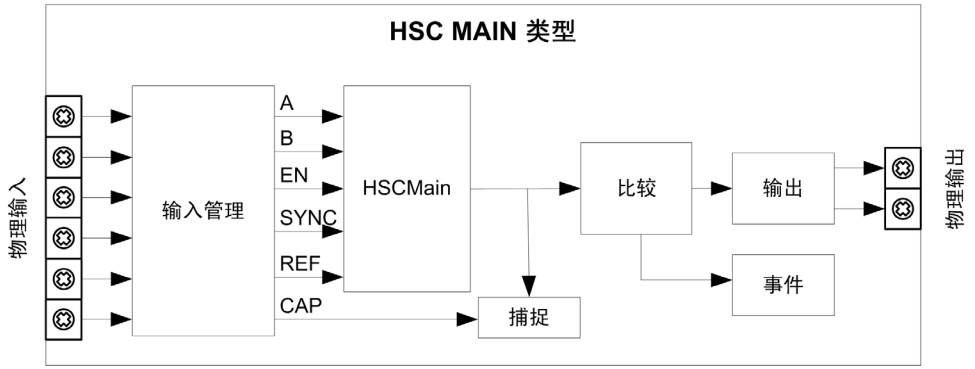
本章包含了以下主题：

主题	页
纲要图	76
自由大型模式下的 Main 类型双相配置	77
Main 类型编程	78
调整参数	81

纲要图

纲要图

下图概述了**自由大型**模式下的 **Main** 类型：



A 和 B 是计数器的计数输入。

EN 是计数器的启用输入。

REF 是计数器的参考输入。

CAP 是计数器的捕捉输入。

SYNC 是计数器的同步输入。

可选功能

除了**自由大型**模式， **Main** 类型还可以提供以下功能：

- 比较功能 (参见第 *119* 页)
- 捕捉功能 (参见第 *127* 页)
- 预设功能 (参见第 *132* 页)
- 启用功能 (参见第 *136* 页)

自由大型模式下的 Main 类型双相配置

过程

按照以下过程配置**自由大型**模式下的 **Main** 类型双相：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Main 双相**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**自由大型**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	设置**计数输入** →**B 输入** →**跳动过滤器**参数的值。
7	输入**范围** →**预设**参数的值以设置计数初始值。
8	输入**范围** →**限制**的值进行限制管理。
9	还可以选择启用以下功能： ● 比较功能 (参见第 *119* 页) ● 捕捉功能 (参见第 *127* 页) ● 预设功能 (参见第 *132* 页)

Main 类型编程

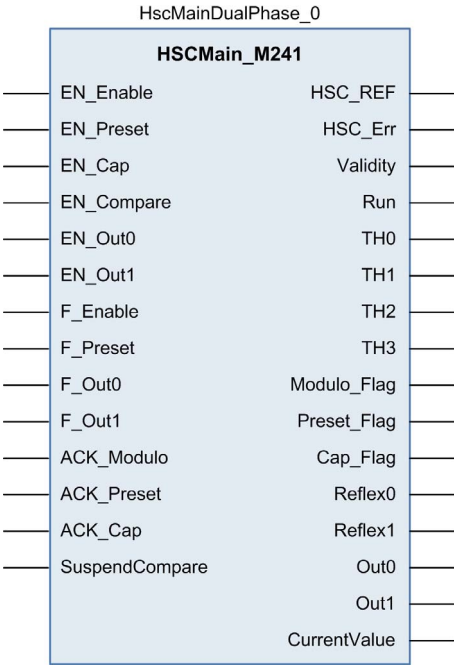
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**HSC** →**HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称（在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量** →**<MyController>** →**PLC** → **IoConfig_Globals**。



I/O 变量用法

下表介绍了功能块的不同引脚在**自由大型**模式中的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	配置 **EN** 输入后：如果为 TRUE，则准许通过启用输入 (参见第 *136* 页) 启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 或 **REF** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *132* 页) 预设计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入。
EN_Compare	BOOL	TRUE = 启用比较器操作 (参见第 *119* 页) （使用阈值 0、 1、 2、 3）： ● 基本比较 （TH0、 TH1、 TH2、 TH3 输出位） ● 反射 （Reflex0、 Reflex1 输出位） ● 事件 （在超出阈值时触发外部任务）
EN_Out0	BOOL	TRUE = 启用物理输出 Out_R0 回显 Reflex0 值 （如果已配置）。
EN_Out1	BOOL	TRUE = 启用物理输出 Out_R1 回显 Reflex1 值 （如果已配置）。
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿预设和启动计数器。
F_Out0	BOOL	TRUE = 强制 Out_R0 为 1 （如果已配置 Reflex0）。
F_Out1	BOOL	TRUE = 强制 Out_R1 为 1 （如果已配置 Reflex1）。
ACK_Modulo	BOOL	在上升沿，复位 Modulo_Flag。
ACK_Preset	BOOL	在上升沿，复位 Preset_Flag。
ACK_Cap	BOOL	在上升沿，复位 Cap_Flag。
SuspendCompare	BOOL	TRUE = 比较结果已挂起： ● TH0、 TH1、 TH2、 TH3、 Reflex0、 Reflex1、 Out0、 Out1 块输出位保持其上一个值。 ● 硬件输出 0、 1 保持各自的上一个值。 ● 事件被掩蔽。 **注意：**在设置 SuspendCompare 时， EN_Compare、 EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 用作**管理**功能块的输入。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	不相关
TH0	BOOL	当 CurrentValue > 阈值 0 时，设为 1 (参见第 *119* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *119* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *119* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *119* 页)。
Modulo_Flag	BOOL	计数器回转到其限制时，设为 1。
Preset_Flag	BOOL	在计数器预设 (参见第 *132* 页) 时，设为 1
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器后，设为 1。 在进行新的捕捉之前，必须先复位此标志。
Reflex0	BOOL	Reflex0 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Reflex1	BOOL	Reflex1 的状态。 只有在设置了 EN_Compare 后才处于活动状态。
Out0	BOOL	物理输出 Out_R0 的状态 （如果已配置 Reflex0）。
Out1	BOOL	物理输出 Out_R1 的状态 （如果已配置 Reflex1）。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可以读取或修改这些参数：

参数	描述
EXPERT_PRESET	获取或设置 HSC 的预设值
EXPERT_THRESHOLD0	获取或设置 HSC 的阈值 0 值
EXPERT_THRESHOLD1	获取或设置 HSC 的阈值 1 值
EXPERT_THRESHOLD2	获取或设置 HSC 的阈值 2 值
EXPERT_THRESHOLD3	获取或设置 HSC 的阈值 3 值
EXPERT_REFLEX0	获取或设置专用功能的输出 0 反射模式
EXPERT_REFLEX1	获取或设置专用功能的输出 0 反射模式

部分 V
事件计数 模式

概述

本部分介绍在**事件计数** 模式下如何使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
11	**事件计数**原理	85
12	使用 **Main** 类型的**事件计数**	87

章 11
事件计数原理

事件计数模式原理介绍

概述

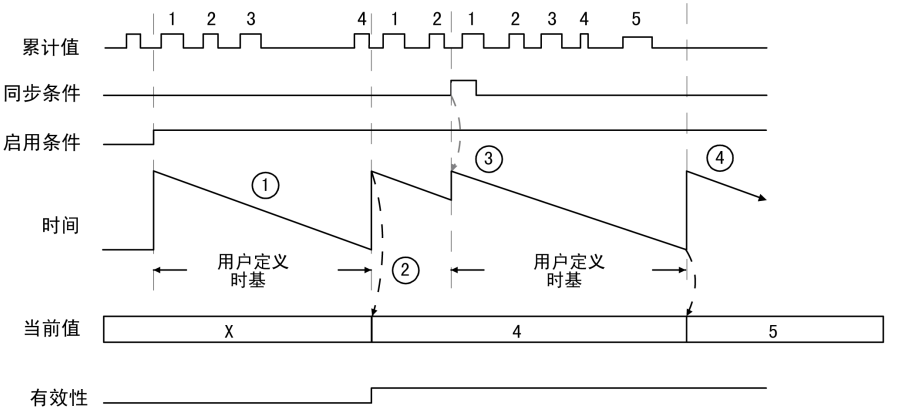
事件计数模式用于计算给定时间段内一系列事件的数量。

规则

计数器评估在预定义时间段内应用到输入的脉冲数。在每个时间段结束时用收到的事件数更新计数寄存器。

在该时间段内可以使用同步。这样将为新的预定义时间段重新启动事件计数。计数在同步条件 (参见第 *132* 页)。

原理图



阶段	操作
1	启用条件 = 1 时，计数器对预定义时间段内物理输入上的事件 （脉冲）数进行累计。 如果 Validity = 0，则当前值不相关。
2	第一个时间段过去后，计数器值设置为该时间段内算出的事件数，并且 Validity 设置为 1。 计数针对新时间段重新启动。
3	在同步条件的上升沿： ● 累计值复位为 0 ● 当前值不更新 ● 计数针对新时间段重新启动
4	该时间段过去后，计数器值设置为在该时间段内算出的事件数。 计数针对新时间段重新启动。

注意：

在 **Main** 类型上，当启用条件：

- 设置为 0：中止当前计数，并且**当前值**保持为前一个有效值。
- 设置为 1：累计值复位为 0，**当前值**保持不变，计数为新时间段重新启动。

章 12
使用 Main 类型的事件计数

概述

本章介绍如何在使用 **Main** 类型的**事件计数**模式下实现高速计数器。

本章包含了哪些内容？

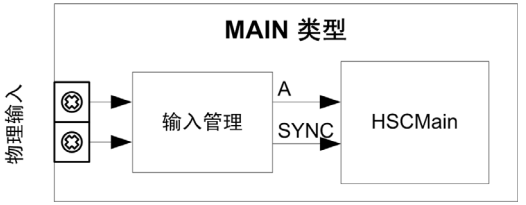
本章包含了以下主题：

主题	页
纲要图	88
事件计数模式下的 Main 类型单相配置	89
Main 类型编程	90
调整参数	93

纲要图

纲要图

下图概述了**事件计数**模式下的 **Main** 类型：



A 是计数器的计数输入。

SYNC 是计数器的同步输入。

可选功能

除了**事件计数**模式， **Main** 类型还可以提供预设功能 (参见第 *132* 页)。

事件计数模式下的 Main 类型单相配置

过程

按照以下过程配置**事件计数**模式下的 **Main** 类型单相：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为 **HSC Main 单相**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**计数模式**参数的值设置为**事件计数**。
5	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	设置**范围** →**时基**参数的值以确定对事件数进行计数的周期。 选择更新循环时间的测量值： ● 0.1 秒 ● 1 秒 （缺省值） ● 10 秒 ● 60 秒
7	还可以设置**控制输入** →**SYNC 输入** →**位置**参数的值以启用预设功能 (参见第 *132* 页)。

Main 类型编程

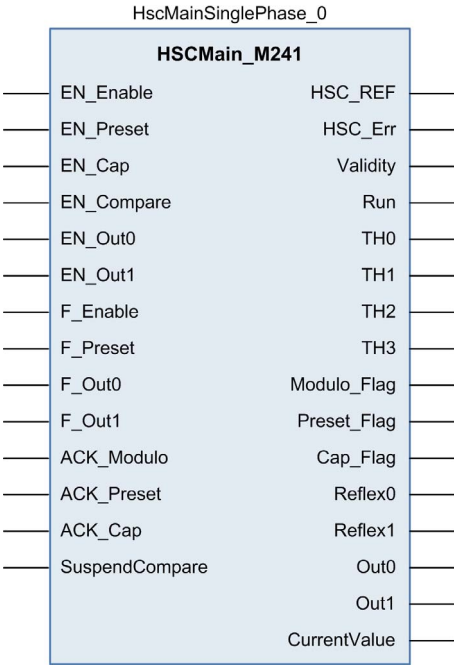
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**HSC** →**HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称（在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量** →**<MyController>** →**PLC** → **IoConfig_Globals**。



I/O 变量用法

下面的表格介绍功能块的不同引脚在**事件**模式下的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	未使用
EN_Preset	BOOL	当配置 **SYNC** 或 **REF** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *132* 页) 预设计数器。
EN_Cap	BOOL	未使用
EN_Compare	BOOL	未使用
EN_Out0	BOOL	未使用
EN_Out1	BOOL	未使用
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿，重新启动与时基相对的内部定时器。
F_Out0	BOOL	未使用
F_Out1	BOOL	未使用
ACK_Modulo	BOOL	未使用
ACK_Preset	BOOL	在上升沿，复位 Preset_Flag。
ACK_Cap	BOOL	未使用
SuspendCompare	BOOL	未使用 **注意：**在设置 SuspendCompare 时， EN_Compare、EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 与**管理**功能块的 EXPERT_REF_IN 输入引脚配合使用。
HSC_Err	BOOL	TRUE = 表示检测到一个错误。 EXPERTGetDiag (参见第 *154* 页) 功能块可用于获取有关此检测到的错误的更多信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	不相关
TH0	BOOL	不相关
TH1	BOOL	不相关
TH2	BOOL	不相关
TH3	BOOL	不相关
Modulo_Flag	BOOL	不相关
Preset_Flag	BOOL	在计数器预设 (参见第 *132* 页) 时，设为 1
Cap_Flag	BOOL	不相关
Reflex0	BOOL	不相关
Reflex1	BOOL	不相关
Out0	BOOL	不相关
Out1	BOOL	不相关
CurrentValue	DINT	计数器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可以读取或修改这些参数：

参数	描述
EXPERT_TIMEBASE	获取或设置 HSC 的时基值

部分 VI
频率计类型

概述

本部分介绍如何在**频率计**类型下使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
13	频率计原理	97
14	频率计类型	99

章 13
频率计原理

描述

概述

频率计类型用于测量事件频率 （以 Hz 为单位）。

频率计类型计算 1 秒时间间隔内的脉冲数。值（以 Hz 为单位）在每个时基值（10、100 或 1000 毫秒）处更新一次。

频率出现波动时，值恢复时间为 1 秒，值精度为 1 Hz。

操作限制

模块在 A 输入上可测量的最大频率为 200 kHz。超出 200 kHz 时，计数寄存器值可能会递减，直到值达到 0。

200 kHz 时的最大占空比为 60%。

章 14
频率计类型

概述

本章介绍如何在**频率计**类型下实现高速计数器。

本章包含了哪些内容？

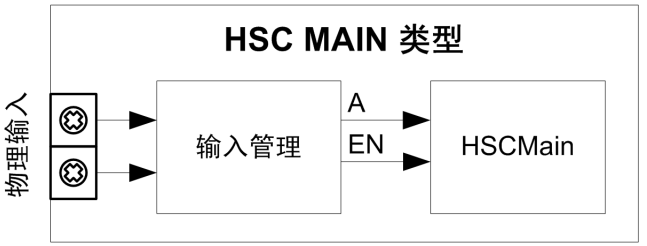
本章包含了以下主题：

主题	页
纲要图	100
频率计类型配置	101
编程	102

纲要图

纲要图

下图概述了**频率计**类型下的 **Main** 类型：



A 是计数器的计数输入。

EN 是计数器的启用输入。

可选功能

除了**频率计**类型， **Main** 类型还可以提供以下功能：

- 启用功能 (参见第 *136* 页)

频率计类型配置

过程

按照以下过程来配置**频率计**类型：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为**频率计**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	设置 **计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
5	设置**范围** →**时基**参数的值以确定对事件数进行计数的周期。 选择更新循环时间的测量值： ● 10 毫秒 ● 100 毫秒 ● 1000 毫秒 （缺省值）
6	还可以设置**控制输入** →**EN 输入** →**位置**参数的值以启用启用功能 (参见第 *136* 页)。

编程

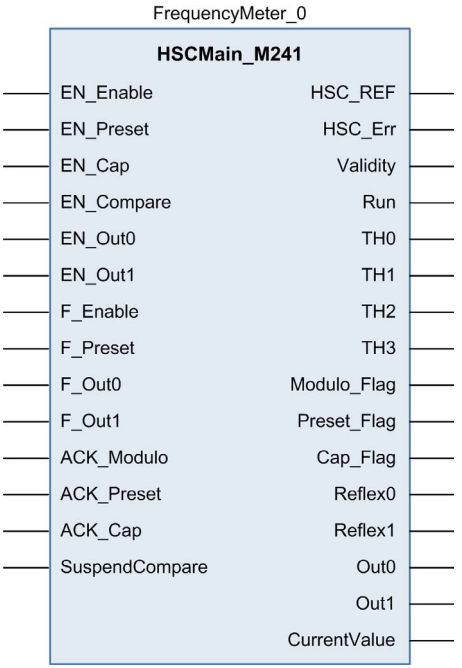
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →HSC →HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称（在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量 →<MyController> →PLC → IoConfig_Globals**。



I/O 变量用法

下面的表格介绍功能块的不同引脚在**频率计**类型下的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见 第 *136* 页) 启用计数器。
EN_Preset	BOOL	未使用
EN_Cap	BOOL	未使用
EN_Compare	BOOL	未使用
EN_Out0	BOOL	未使用
EN_Out1	BOOL	未使用
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿，重新启动与时基相对的内部定时器。
F_Out0	BOOL	未使用
F_Out1	BOOL	未使用
ACK_Modulo	BOOL	未使用
ACK_Preset	BOOL	在上升沿，复位 。
ACK_Cap	BOOL	未使用
SuspendCompare	BOOL	未使用 **注意：**在设置 SuspendCompare 时， EN_Compare、 EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 与**管理**功能块的 EXPERT_REF_IN 输入引脚配合使用。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	不相关
TH0	BOOL	不相关
TH1	BOOL	不相关
TH2	BOOL	不相关
TH3	BOOL	不相关
Modulo_Flag	BOOL	不相关
Preset_Flag	BOOL	在计数器预设 (参见第 *132* 页) 时，设为 1
Cap_Flag	BOOL	不相关
Reflex0	BOOL	不相关
Reflex1	BOOL	不相关
Out0	BOOL	不相关
Out1	BOOL	不相关
CurrentValue	DINT	计数器的当前计数器值。

部分 VII
周期计类型

概述

本部分介绍如何在**周期计**类型下使用 HSC。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
15	周期计类型原理	107
16	**周期计**类型	109

章 15
周期计类型原理

描述

概述

周期计类型可用来：
- 确定事件的持续时间
- 确定两个事件之间的时间间隔
- 设置并测量过程的执行时间

可以通过 2 种方式使用**周期计**：
- 沿对反向沿：用于测量事件的持续时间。
- 沿对沿：用于测量 2 个事件之间的时间间隔。

可在配置屏幕中指定超时值。

使用此功能可以停止超过此超时时间的测量。这种情况下，计数寄存器在下一次完整的测量之前一直无效。

测量用精度参数 （1 µs、 100 µs、 1000 µs）定义的单位表示。

例如，如果 `CurrentValue` = 100，则**精度**参数为：
0.001 (1 µs) 测量 = 0.1 毫秒
0.1 (100 µs) 测量 = 10 毫秒
1 (1000 µs) 测量 = 100 毫秒

沿对反向沿模式

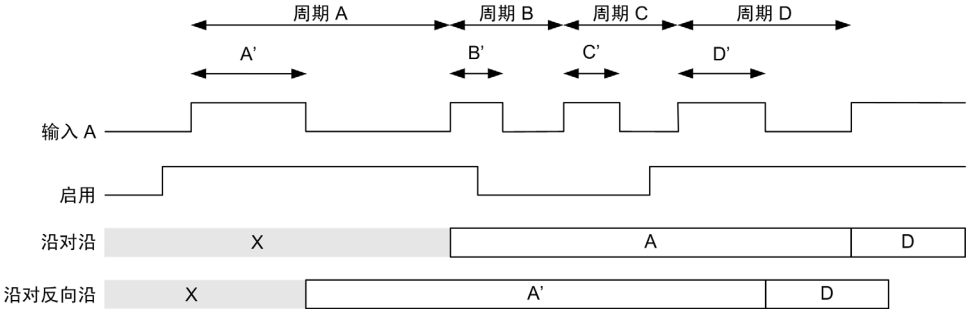
启用条件 = 1 时，在 A 输入的上升沿和下降沿之间进行测量。一检测到下降沿，即更新计数寄存器。

沿对沿模式

启用条件 = 1 时，在 A 输入的 2 个上升沿之间进行测量。一检测到第二个上升沿，即更新计数寄存器。

启用条件中断行为

下面的趋势图描述启用条件中断时计数寄存器的行为：



操作限制

此模块每 5 毫秒内最多执行 1 次测量。

即使配置中定义的单位为 1 µs，可以测量到的最短脉冲也仍为 100 µs。

可以测量的最大持续时间为 1,073,741,823 个单位。

章 16
周期计类型

概述

本章介绍如何在**周期计**类型中实现高速计数器。

本章包含了哪些内容？

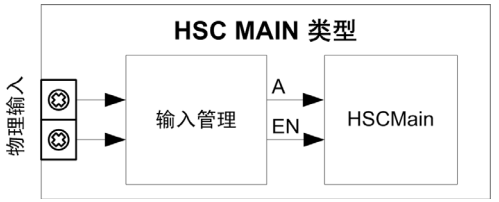
本章包含了以下主题：

主题	页
纲要图	110
沿对沿模式下的频率计类型配置	111
沿对反向沿模式下的周期计类型配置	112
编程	113
调整参数	116

纲要图

纲要图

下图概述了**周期计**类型中的 **Main 类型**：



A 是计数器的计数输入。

EN 是计数器的启用输入。

可选功能

除了**周期计**类型， **Main** 类型还可以提供以下功能：

- 启用功能 (参见第 *136* 页)

沿对沿模式下的频率计类型配置

过程

按照以下过程配置**沿对沿**模式下的**周期计**类型：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果：计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为**周期计**。 **结果：**根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意：实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**周期计模式**参数的值设置为**沿对沿**。
5	设置**计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	设置**范围** →**精度**参数的值。 选择测量单位： ● 1 微秒 （缺省值） ● 100 微秒 ● 1000 微秒
7	输入**范围** →**超时**参数的值以设置测量的周期不得超过的时间值。
8	还可以设置**控制输入** →**EN 输入** →**位置**参数的值以启用启用功能 (参见第 *136* 页)。

沿对反向沿模式下的周期计类型配置

过程

按照以下过程配置**沿对反向沿**模式下的**周期计**类型：

步骤	操作
1	双击 **MyController** →**计数器**。 **结果**：**计数器**编辑器选项卡随即打开以进行 HSC 配置。
2	在**计数器**编辑器选项卡中，将**计数功能**参数的值设置为**周期计**。 **结果**：根据选择的计数器功能，将在**计数器**编辑器选项卡中显示配置参数。
3	如有必要，输入**常规** →**实例名称**参数的值。 **注意**：**实例名称**由软件自动提供，可以为计数器功能块原样使用。
4	将**常规** →**周期计模式**参数的值设置为**沿对反向沿**。
5	设置**计数输入** →**A 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
6	设置**范围** →**精度**参数的值。 选择测量单位： ● 1 微秒 （缺省值） ● 100 微秒 ● 1000 微秒
7	输入**范围** →**超时**参数的值以设置测量的周期不得超过的时间值。
8	还可以设置**控制输入** →**EN 输入** →**位置**参数的值以启用启用功能 (参见第 *136* 页)。

编程

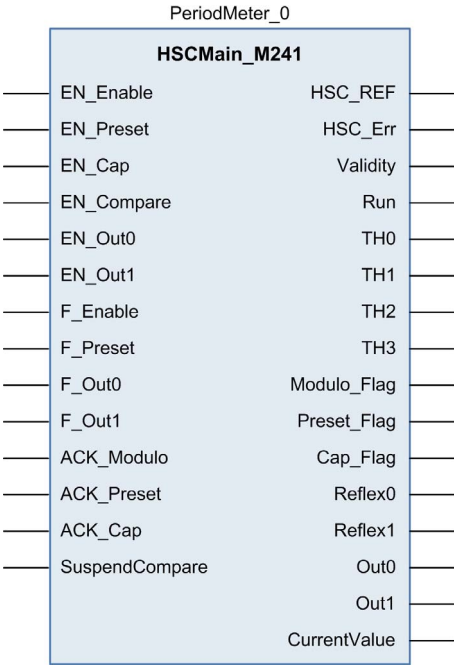
概述

Main 类型始终由 HSCMain 功能块进行管理。

注意：如果使用 HSCMain 功能块管理其他 HSC 类型，则在编译时会显示检测到的错误代码。

添加 HSCMain 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →HSC →HSCMain_M241**，然后将该项目拖放至 **POU** 窗口中。
2	键入 **Main** 类型实例名称（在配置中定义），或通过单击以下内容选择功能块实例： 使用输入助手，可按照以下路径选择 HSC 实例：**全局变量 →<MyController> →PLC → IoConfig_Globals**。



I/O 变量用法

下面的表格介绍功能块的不同引脚在**周期计**类型下的使用方法。

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见 第 *136* 页) 启用计数器。
EN_Preset	BOOL	未使用
EN_Cap	BOOL	未使用
EN_Compare	BOOL	未使用
EN_Out0	BOOL	未使用
EN_Out1	BOOL	未使用
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	未使用
F_Out0	BOOL	未使用
F_Out1	BOOL	未使用
ACK_Modulo	BOOL	未使用
ACK_Preset	BOOL	未使用
ACK_Cap	BOOL	未使用
SuspendCompare	BOOL	未使用 **注意：**在设置 SuspendCompare 时， EN_Compare、 EN_ReflexO、 EN_Reflex1、 F_Out0、 F_Out1 保持运行。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。 与**管理**功能块的 EXPERT_REF_IN 输入引脚配合使用。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。 如果超过了超时值，则 Validity = FALSE。
Run	BOOL	TRUE = 计数器正在运行。
TH0	BOOL	不相关
TH1	BOOL	不相关
TH2	BOOL	不相关
TH3	BOOL	不相关
Modulo_Flag	BOOL	不相关
Preset_Flag	BOOL	不相关
Cap_Flag	BOOL	在进行新的捕捉之前，必须先复位此标志。
Reflex0	BOOL	不相关
Reflex1	BOOL	不相关
Out0	BOOL	不相关
Out1	BOOL	不相关
CurrentValue	DINT	计数器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam (参见第 *158* 页) 或 EXPERTSetParam (参见第 *160* 页) 功能块进行读取或修改。

注意：通过程序设置的参数将覆盖在 HSC 配置窗口中配置的参数值。初始配置参数会在冷启动或热启动 *(参见 Modicon M241 Logic Controller, 编程指南)* 后恢复。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE (参见第 *148* 页) 的参数，在程序运行过程中可对这些参数进行读取或修改：

参数	描述
EXPERT_TIMEBASE	获取或设置 HSC 的精度值

部分 VIII
可选功能

概述

本部分提供有关 HSC 可选功能的信息。

本部分包含了哪些内容？

本部分包括以下各章：

章	章节标题	页
17	比较功能	119
18	捕捉功能	127
19	预设和启用功能	131

章 17
比较功能

概述

本章介绍有关 HSC 的比较功能的信息。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
使用 **Main** 类型的比较原理	120
对使用 **Main** 类型的比较进行配置	124
外部事件配置	125

使用 Main 类型的比较原理

概述

使用 **Main** 类型的比较块用于管理以下模式中的阈值、反射输出和事件：

- One-shot (参见第 *33* 页)
- Modulo-loop (参见第 *47* 页)
- Free-Large (参见第 *67* 页)

激活至少一个阈值后，在配置屏幕 (参见第 *124* 页) 中对比较进行配置。

比较可以用来触发：

- 对关于阈值的操作进行编程 (参见第 *121* 页)
- 与外部任务相关联的阈值事件 (参见第 *121* 页)
- 反射输出 (参见第 *121* 页)

比较原理

Main 类型或可最多管理 4 个阈值。

相对于当前的计数值而言，阈值是一个配置值。阈值用来定义最多 5 个区域，或对越过的值作出反应。

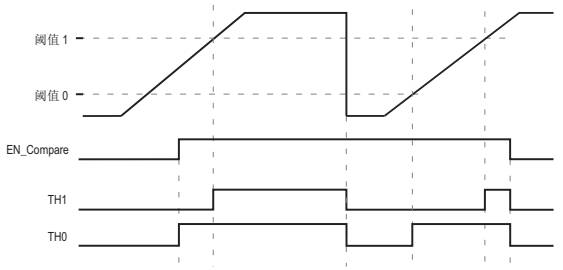
阈值通过配置来定义，也可通过使用 EXPERTSetParam (参见第 *160* 页) 功能块在应用程序中进行调整。

如果配置了阈值 x（x= 1、 2、 3），并且已启用比较 (`EN_Compare` = 1)，则 `HSCMain_M241` 功能块的输出引脚 THx 发生以下情况：

- 在计数器值 >= 阈值 x 时设置
- 在计数器值 < 阈值 x 时复位

注意： 如果 `HSCMain_M241` 功能块上的 `EN_Compare` 设为 0，则禁用比较功能，包括由阈值事件和反射输出触发的外部任务。

2 个阈值示例：



阈值行为

对于时间常数较低的应用程序来说，使用任务环境中（功能块的 TH0 到 TH2 输出引脚）的可用阈值比较状态是一个理想选择。

例如，可以用它监控水箱中的液体水位。

配置事件

配置越过阈值时的事件可触发外部任务 (参见第 *125* 页)。可以选择在向下、向上或向下 / 向上越过配置的阈值时触发事件。

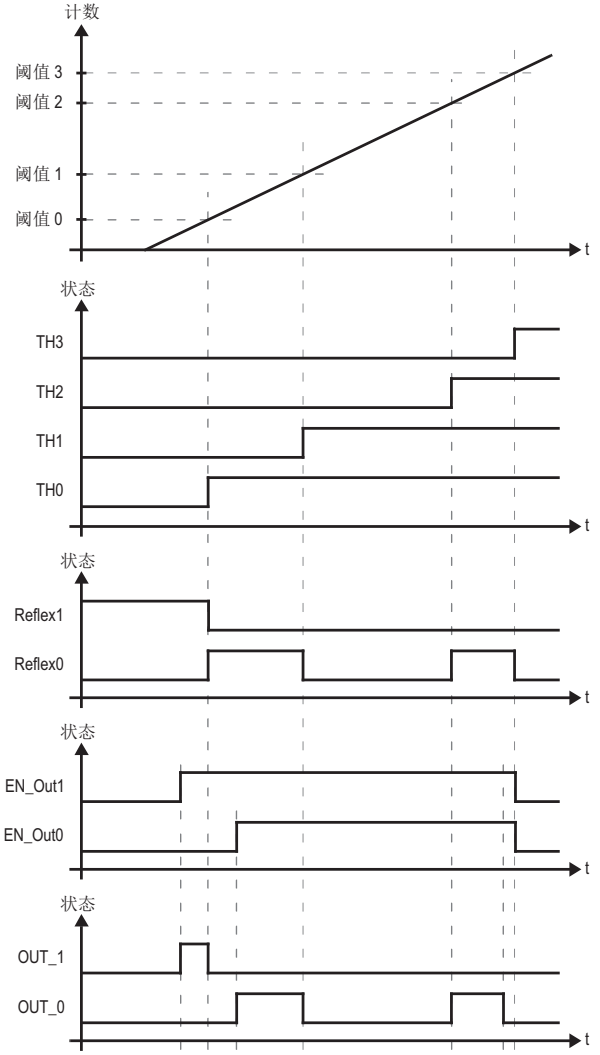
反射输出行为

配置反射输出可触发物理反射输出。

这些输出在任务环境中不受控制，因此可以将反应时间降至最短。这为需要快速执行的操作提供了便利。

高速计数器使用的输出只能通过功能块访问。不能在应用程序中直接对其进行读写。

由阈值触发的反射输出的示例：



注意：反射输出的状态取决于配置。

更改阈值

在阈值比较正在进行时，请谨慎操作，避免输出或突发性事件任务执行导致的意外结果。如果禁用比较功能，则可以随意修改阈值。但是，如果启用了比较功能，请在修改阈值时至少暂停阈值比较功能。

⚠ 警告

意外的设备操作

- 如果 EN_Compare = 1，则在更改阈值时，必须使用 SuspendCompare 输入。
- 在重新激活阈值比较功能之前，确认 TH0 小于 TH1， TH1 小于 TH2， TH2 小于 TH3。

不遵循上述说明可能导致人员伤亡或设备损坏。

当 EN_Compare = 1 时，比较处于活动状态，需要遵循以下过程：

步骤	操作
1	将 SuspendCompare 设为 1。 在当前值下，比较被冻结： ● 块的 TH0、 TH1、 Reflex0、 Reflex1、 Out0、 Out1 输出位保持其上一个值。 ● 物理输出 0、 1 保持各自上一个值 ● 事件被隐藏 **注意：**在设置 SupendCompare 的同时， EN_Compare、 EN_Out0、 EN_Out1、 F_Out0、 F_Out1 保持运行。
2	根据需要，使用 EXPERTSetParam (参见第 *158* 页) 功能块修改阈值。 **注意：**根据下面的规则，配置阈值：TH0 < TH1 < TH2 < TH3。
3	将 SuspendCompare 设为 0。 应用新的阈值，并恢复比较。

对使用 Main 类型的比较进行配置

配置过程

请按照以下步骤对使用 **Main** 类型的比较功能进行配置：

步骤	操作
1	在**设备树**中，双击 **MyController** →**计数器**。
2	将**计数功能**参数的值设置为 **HSC Main 单相**或 **HSC Main 双相**。
3	在**阈值数**参数中，选择要使用的阈值数。
4	选择阈值的值。 **注意：**根据以下规则配置阈值：TH0 < TH1 < TH2 < TH3
5	也可以选择提供事件条件 (参见第 *125* 页)。
6	也可以选择配置**反射输出**行为 (参见第 *120* 页)。

外部事件配置

过程

下面的过程介绍如何配置外部事件 *(参见 Modicon M241 Logic Controller, 编程指南)* 以激活任务:

步骤	操作
1	在**应用程序树**选项卡中，添加任务。
2	双击要关联到外部事件的任务节点。
3	在**类型**下拉菜单中，选择**外部**。
4	在**外部事件**下拉菜单中，选择要关联到任务的事件 （请参见以下列表）。

外部事件

下表介绍可能会关联任务的外部事件：

事件名称	描述
I0	输入 I0 设置为 1 时，激活任务。
I1	输入 I1 设置为 1 时，激活任务。
I2	输入 I2 设置为 1 时，激活任务。
I3	输入 I3 设置为 1 时，激活任务。
I4	输入 I4 设置为 1 时，激活任务。
I5	输入 I5 设置为 1 时，激活任务。
I6	输入 I6 设置为 1 时，激活任务。
I7	输入 I7 设置为 1 时，激活任务。
HSC0_TH0	HSC0 的阈值 TH0 设置为 1 时，激活任务。
HSC0_TH1	HSC0 的阈值 TH1 设置为 1 时，激活任务。
HSC0_TH2	HSC0 的阈值 TH2 设置为 1 时，激活任务。
HSC0_TH3	HSC0 的阈值 TH3 设置为 1 时，激活任务。
HSC0_Stop	HSC0 设置为 0 时，激活任务。
HSC1_TH0	HSC1 的阈值 TH0 设置为 1 时，激活任务。
HSC1_TH1	HSC1 的阈值 TH1 设置为 1 时，激活任务。
HSC1_TH1	HSC1 的阈值 TH1 设置为 1 时，激活任务。
HSC1_TH3	HSC1 的阈值 TH3 设置为 1 时，激活任务。
HSC1_Stop	HSC1 设置为 0 时，激活任务。

章 18
捕捉功能

概述

本章提供有关 HSC 捕捉功能的信息。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
使用 **Main** 类型的捕捉原理	128
对使用 **Main** 类型的捕捉进行配置	129

使用 Main 类型的捕捉原理

概述

捕捉功能能够根据外部输入信号存储当前的计数器值。

捕捉功能在使用以下模式的 **Main** 类型中可用：

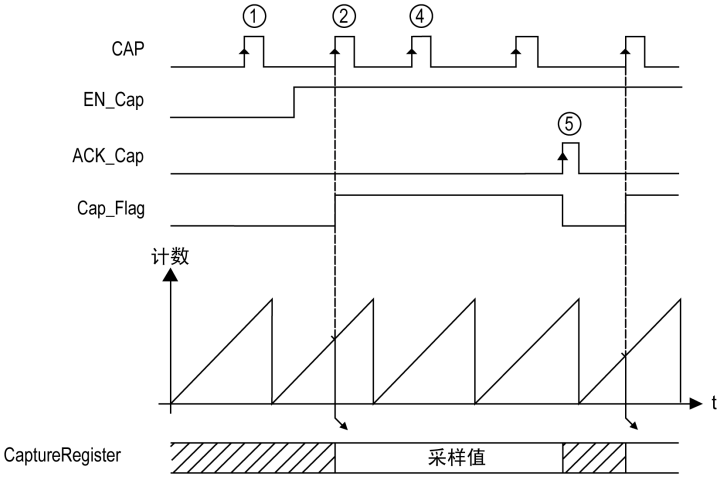
- One-shot (参见第 *39* 页)
- Modulo-loop (参见第 *59* 页)
- Free-large (参见第 *75* 页)

使用此功能需要：

- 配置可选的捕捉输入：**CAP**
- 使用 EXPERTGetCapturedValue （参见第 *152* 页) 功能块检索在应用程序中捕捉到的值。

捕捉原理

下图显示了**模数回路**模式中的捕捉工作原理：



阶段	操作
1	当 EN_Cap = 0 时，无法执行此功能。
2	当 EN_Cap = 1 时， **CAP** 上的跳变沿会捕捉当前的计数器值并将其放入捕捉寄存器，然后触发 Cap_Flag 的上升沿。
3	使用 EXPERTGetCapturedValue (参见第 *152* 页) 获取存储的值。
4	当 Cap_Flag = 1 时，忽略物理输入 CAP 上任何新的跳变沿。
5	HSCMain_M241 (参见第 *162* 页) 功能块输入 ACK_Cap 的上升沿触发下降沿 Cap_Flag 输出。准许新的捕捉。

对使用 Main 类型的捕捉进行配置

配置过程

请按照以下过程对使用 **Main** 类型的捕捉功能进行配置：

步骤	操作
1	在**设备树**中，双击 **MyController** →**计数器**。
2	将**计数功能**参数的值设置为 **HSC Main 单相**或 **HSC Main 双相**。
3	将**捕捉** →**CAP 输入** →**位置**参数的值选择为 **I3** 或 **I10**。
4	选择**捕捉** →**CAP 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。
5	选择**捕捉** →**模式**参数的触发模式： ● 预设 (参见第 *132* 页)（缺省值） ● **CAP 上升沿** ● **CAP 下降沿** ● **CAP 上升和下降沿**

章 19
预设和启用功能

概述

本章提供有关 HSC 的同步和启用功能的信息。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
预设功能	132
自由大型或周期计预设条件	134
启用功能	136

预设功能

概述

预设功能用来设置 / 复位计数器操作。

预设功能准许在下列计数模式中同步和启动计数功能：

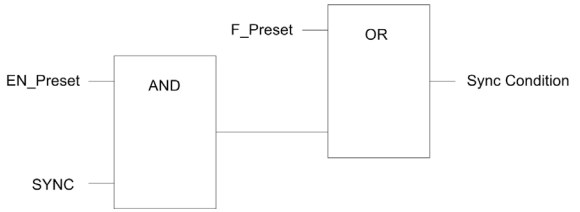
- **一次性**计数器：预设和启动计数器
- **模数回路**计数器：复位和启动计数器
- **事件计数**：在开始时重新启动内部时基

注意：Simple HSC 类型的同步条件对应于功能块输入 `Sync`。

描述

此功能用来根据可选 SYNC 物理输入和功能块输入 `F_Preset` 和 `EN_Preset` 的状态和配置，对计数器进行同步。

下图说明了 HSC 的同步条件：



EN_Preset HSC 功能块的输入
F_Preset HSC 功能块的输入
SYNC 物理输入 SYNC

当达到同步条件时，功能块输出 `Preset_Flag` 设为 1。

同步条件在上升沿上运行。

配置

以下过程描述如何配置预设功能：

步骤	操作
1	在**设备树**中，双击 **MyController** →**计数器**。
2	将**计数功能**参数的值设置为 **HSC 主单相**或 **HSC 主双相**。
3	选择**控制输入** →**SYNC 输入** →**位置**参数的值。
4	选择**控制输入** →**SYNC 输入** →**跳动过滤器**参数的值。
5	选择**控制输入** →**沿检测**参数的值。

转换类型

SYNC 物理输入的转换类型由 **沿检测**参数确定。

有 3 种可用转换，通过配置定义：

- SYNC 输入的上升沿
- SYNC 输入的下降沿
- SYNC 输入的上升沿和下降沿

有 3 种可用转换，通过配置定义：

值	描述
Sync 上升沿	在上升沿完成检测。
Sync 下降沿	在下降沿完成检测。
Sync 上升和下降沿	在上升沿和下降沿完成检测。

自由大型或周期计预设条件

概述

在**自由大型**模式中，使用 2 个输入创建预设条件：

- SYNC
- REF

可以使用 个预设条件：

- 在输入 SYNC 的跳变沿 （上升沿）
- 在输入 REF 的上升沿
- 如果输入 REF 为活动高信号，则在输入 SYNC 的上升沿
- 在 REF 输入信号上升沿之后的第一个 SYNC 脉冲
- 在 REF 输入信号下降沿之后的第一个 SYNC 脉冲

在输入 SYNC 的跳变沿 （上升沿）

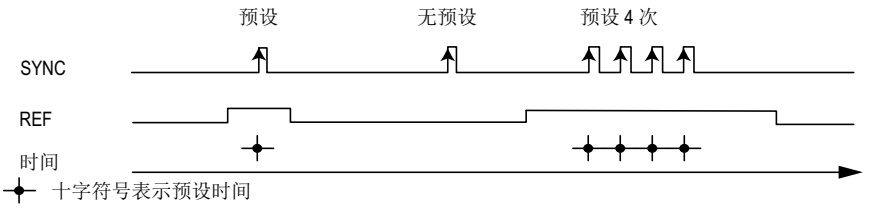
计数器在编码器参考点上进行同步。

在输入 REF 的上升沿

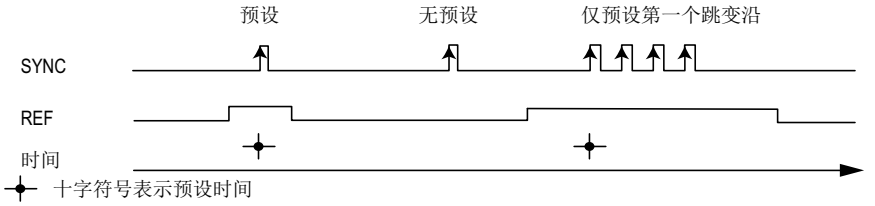
计数器在机械位置进行同步。

如果输入 REF 为活动高信号，则在输入 SYNC 的上升沿

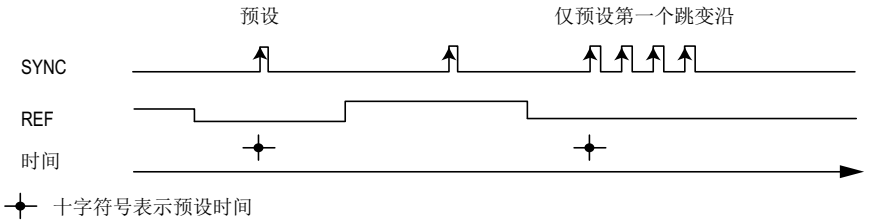
当 **REF** 信号为 TRUE 时，计数器在编码器参考点上进行同步，如下所示：



在 REF 输入信号上升沿之后的第一个 SYNC 脉冲：



在 REF 输入信号下降沿之后的第一个 SYNC 脉冲：



启用功能

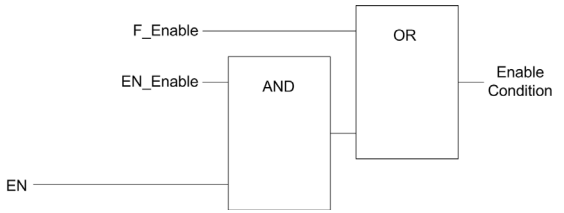
概述

启用功能用于准许计数操作。

描述

此功能用于准许根据可选 EN 物理输入以及功能块输入 F_Enable 和 EN_Enable 的状态对当前计数器值进行更改。

下图显示了启用条件：



EN_Enable HSC 功能块的输入
F_Enable HSC 功能块的输入
EN 物理输入 Enable

如果没启用此功能，计数脉冲将被忽略。

注意：Simple 类型的启用条件与功能块输入 Enable 一致。

配置

以下过程介绍如何配置启用功能：

步骤	操作
1	在**设备树**中，双击 **MyController** →**计数器**。
2	选择**计数器**选项卡。
3	将**控制输入** →**EN 输入** →**位置**参数的值设置为 **I1**。
4	选择**控制输入** →**EN 输入** →**跳动过滤器**参数的值以减少输入上的跳动影响。 过滤值确定计数器最大频率，如跳动过滤器表 (参见第 *140* 页) 所示。

附录

概述

本附录摘录了部分编程指南，以方便用户获得对库文档的技术理解。

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	一般信息	139
B	数据类型	143
C	功能块	151
D	功能和功能块表示形式	169

附录 A
一般信息

概述

本章介绍的信息通用于 FG/PTO/PWM 和 HSC 功能。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
专用特性	140
有关管理和运动功能块管理的一般信息	141

专用特性

跳动过滤器

下表显示的是由过滤值确定的最大计数器频率，这些过滤值用于减少对输入的跳动影响：

输入	跳动过滤器值 （毫秒）	最大计数器频率
A B EN CAP	0.000	200 kHz
	0.001	200 kHz
	0.002	200 kHz
	0.005	100 kHz
	0.010	50 kHz
	0.05	25 kHz
	0.1	5 kHz
	0.5	1 kHz
	1	500 Hz
	5	100 Hz
A 是计数器的计数输入。 B 是双相位计数器的计数输入。 EN 是计数器的启用输入。 CAP 是计数器的捕捉输入。		

专用输出

Frequency Generator, 脉冲串输出、脉冲宽度调制和高速计数器使用的输出只能通过功能块进行访问。不能在应用程序内直接对其进行读写。

⚠ 警告

意外的设备操作

- 请勿在多个任务中使用某个功能块的同一实例。
- 请勿在功能块处于活动状态 （正在执行）时修改功能块参考 (••_REF_IN)。

不遵循上述说明可能导致人员伤亡或设备损坏。

有关管理和运动功能块管理的一般信息

输入变量的管理

该功能块在 Execute 输入的上升沿上启动。

此时无需对输入变量进行任何进一步的修改。

按照 IEC 61131-3 标准，如果功能块有任何变量输入缺失 （即断开或未连接），则使用上一次调用功能块实例的值。在此情况下，第一次调用时将应用初始配置值。因此，功能块最好始终带有特定于其输入的已知值，这样有助于消除调试程序的麻烦。对于 HSC 和 PTO 功能块，最好只使用一次实例，且该实例最好位于主任务中。

输出变量的管理

Done、 InVelocity 或 InFrequency 输出与 Busy、 CommandAborted 和 Error 输出互斥：在一个功能块上，只能有一个输出为 TRUE。如果 Execute 输入为 TRUE，则其中一个输出为 TRUE。

在 Execute 输入的上升沿，会设置 Busy 输出。在功能块执行过程中，此 Busy 输出保持已设置状态，并在某一其他输出 （Done、 InVelocity、 InFrequency、 CommandAborted 和 Error）的上升沿复位。

当功能块成功执行完毕时，会设置 Done、 InVelocity 或 InFrequency 输出。

当功能块的执行被另一功能块中断时，则改为设置 CommandAborted 输出。

当功能块的执行由于检测到错误而结束时，则会设置 Error 输出，并通过 ErrId 输出给出检测到的错误编号。

Done、 InVelocity、 InFrequency、 Error、 ErrID 和 CommandAborted 输出在 Execute 的下降沿复位。如果 Execute 输入在执行完成之前复位，则在执行结束时，这些输出的设置状态将持续一个任务循环。

当功能块的某个实例在完成之前收到新的 Execute 时，则对于以前的操作，功能块不返回任何反馈，例如 Done。

处理检测到的错误

所有功能块都有 2 个输出，可以报告在执行功能块期间检测到的错误。

- 检测到错误时， Error = TRUE。
- ErrID 在 Error = TRUE 时返回检测到的错误 ID。

附录 B
数据类型

概述

本章介绍 HSC 库的数据类型。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
EXPERT_ERR_TYPE: Type for Error Variable of EXPERT Function Block	144
EXPERT_FREQMETER_TIMEBASE_TYPE：频率计时基变量类型	145
EXPERT_HSCMAIN_TIMEBASE_TYPE：HSC Main 时基变量的类型	146
EXPERT_IMMEDIATE_ERR_TYPE：GetImmediateValue 功能块的错误变量类型	147
EXPERT_PARAMETER_TYPE：要在 EXPERT 上获取或设置的参数类型	148
EXPERT_PERIODMETER_RESOLUTION_TYPE：周期计时基变量类型	149
EXPERT_REF：EXPERT 参考值	150

EXPERT_ERR_TYPE: Type for Error Variable of EXPERT Function Block

枚举类型描述

枚举数据类型 ENUM 包含检测到的错误的各种类型，这些类型具有下列值：

枚举器	值	描述
EXPERT_NO_ERROR	00 hex	未检测到错误。
EXPERT_UNKNOWN	01 hex	参考 EXPERT 不正确或未配置。
EXPERT_UNKNOWN_PARAMETER	02 hex	参数引用不正确。 请参阅 PARAMETER_TYPE 一节，了解有效参数 (参见第 *148* 页)。
EXPERT_INVALID_PARAMETER	03 hex	参数值不正确。 例如， Preset Value <TH1 或 <TH0。
EXPERT_COM_ERROR	04 hex	检测到 EXPERT 模块存在通讯错误。
EXPERT_CAPTURE_NOT_CONFIGURED	05 hex	未配置捕捉。 无法获取捕捉的值。

EXPERT_FREQMETER_TIMEBASE_TYPE：频率计时基变量类型

枚举类型描述

枚举数据类型 ENUM 包含可用于专用功能块的各种时基值：

名称	值
EXPERT_FREQMETER_10ms	10
EXPERT_FREQMETER_100ms	100
EXPERT_FREQMETER_1000ms	1000

EXPERT_HSCMAIN_TIMEBASE_TYPE：HSC Main 时基变量的类型

枚举类型描述

枚举数据类型 ENUM 包含可用于 EXPERT Main 功能块的各种时基值：

名称	值
EXPERT_HSCMAIN_100ms	00 hex
EXPERT_HSCMAIN_1s	01 hex
EXPERT_HSCMAIN_10s	02 hex
EXPERT_HSCMAIN_60s	03 hex

EXPERT_IMMEDIATE_ERR_TYPE：GetImmediateValue 功能块的错误变量类型

枚举类型描述

枚举数据类型 ENUM 包含不同类型的检测到的错误，具有下列值：

枚举器	值	描述
EXPERT_IMMEDIATE_FUNC_NO_ERROR	00 hex	未检测到错误
EXPERT_IMMEDIATE_FUNC_UNKNOWN	01 hex	IMMEDIATE 功能的参考不正确或未配置
EXPERT_IMMEDIATE_FUNC_UNKNOWN_PARAMETER	02 hex	参数参考不正确

EXPERT_PARAMETER_TYPE：要在 EXPERT 上获取或设置的参数类型

枚举类型描述

枚举数据类型 ENUM 包含下列值：

枚举器	值	描述
EXPERT_PRESET	00 hex	获取或设置 EXPERT 功能的预设值。
EXPERT_MODULO	01 hex	获取或设置 EXPERT 功能的模数值。
EXPERT_TIMEBASE	03 hex	获取或设置 EXPERT 功能的时基值 (参见 第 *146* 页)。
EXPERT_THRESHOLD0	06 hex	获取或设置 EXPERT 功能的阈值 0 值。
EXPERT_THRESHOLD1	07 hex	获取或设置 EXPERT 功能的阈值 1 值。
EXPERT_THRESHOLD2	08 hex	获取或设置 EXPERT 功能的阈值 2 值。
EXPERT_THRESHOLD3	09 hex	获取或设置 EXPERT 功能的阈值 3 值。
EXPERT_REFLEX0	0A（十六进制）	获取或设置 EXPERT 功能的输出 0 反射模式
EXPERT_REFLEX1	0B（十六进制）	获取或设置 EXPERT 功能的输出 1 反射模式

EXPERT_PERIODMETER_RESOLUTION_TYPE：周期计时基变量类型

枚举类型描述

枚举数据类型 ENUM 包含可用于专用功能块的各种时基值：

名称	值
EXPERT_PERIODMETER_1µs	00 hex
EXPERT_PERIODMETER_100µs	01 hex
EXPERT_PERIODMETER_1000µs	02 hex

EXPERT_REF：EXPERT 参考值

数据类型描述

EXPERT_REF 是一个字节，用于识别与管理功能块关联的 EXPERT 功能。

附录 C
功能块

概述

本章介绍 HSC 库的功能和功能块。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
EXPERTGetCapturedValue：返回捕捉寄存器的内容	152
EXPERTGetDiag：提供在主体专用 I/O 功能 上检测到的错误的详细信息	154
EXPERTGetImmediateValue：读取 HSC 的计数器值	156
EXPERTGetParam：返回主体专用 I/O 功能 的参数	158
EXPERTSetParam：调整 HSC 的参数	160
HSCMain_M241：HSC Main 功能块	162
HSCSimple_M241：HSC Simple 功能块	166

EXPERTGetCapturedValue：返回捕捉寄存器的内容

功能描述

此管理功能块返回捕捉寄存器的内容。

图形表示形式

IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表对输入变量进行了描述：

输入	类型	注释
EXPERT_REF_IN	EXPERT_REF (参见第 *150* 页)	专用功能块的参考。 在功能块执行期间，不得进行更改。
Execute	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。
CaptureNumber	BYTE	捕捉寄存器的索引： ● 对于 HSCMain：始终为 0 ● 对于 Cap0：0 ● 对于 Cap1：1

下表对输出变量进行了描述：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF (参见第 *150* 页)	专用功能块的参考。
Done	BOOL	TRUE = 表示 CaptureValue 有效。 功能块执行结束。
Busy	BOOL	TRUE = 表示功能块执行正在进行中。
Error	BOOL	TRUE = 表示检测到一个错误。 功能块执行结束。
ErrID	EXPERT_ERR_TYPE (参见第 *144* 页)	当 Error 为 TRUE 时：检测到的错误的类型。
CaptureValue	DINT	如果 Done 为 TRUE：捕捉寄存器值有效。

注意：如果检测到错误，则变量采用捕捉的最后一个值。

注意：有关 Done、 Busy 和 Execution 引脚的详细信息，请参阅功能块管理的一般信息 (参见第 *141* 页)。

添加 EXPERTGetCapturedValue 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →管理 →EXPERTGetCapturedValue**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

EXPERTGetDiag：提供在主体专用 I/O 功能 上检测到的错误的详细信息

功能描述

此管理功能块返回检测到的 HSC 错误的详细信息。

图形表示形式

IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表对输入变量进行了描述：

输入	类型	注释
EXPERT_REF_IN	EXPERT_REF（参见第 *150* 页）	专用功能块的参考。 在功能块执行期间，不得进行更改。
Execute	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。

下表对输出变量进行了描述：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF（参见第 *150* 页）	专用功能块的参考。
Done	BOOL	TRUE = 表示 HSCDiag 有效。 功能块执行结束。
Busy	BOOL	TRUE = 表示功能块执行正在进行中。
Error	BOOL	TRUE = 表示检测到一个错误。 功能块执行结束。
ErrID	EXPERT_ERR_TYPE（参见第 *144* 页）	当 Error 为 TRUE 时：检测到的错误的类型。
EXPERTDiag	DWORD	当 Done 为 TRUE 时：诊断值有效，请参阅下表。

注意：有关 Done、 Busy 和 Execution 引脚的详细信息，请参阅功能块管理的一般信息 (参见第 *141* 页)。

下表指示诊断值：

位	本体 （HSCMain 或 HSCSimple）	描述
0	–	检测到输入错误
1	–	检测到输出错误
2	–	–
3	–	–
4	–	自检时检测到内部错误
5	–	–
6	–	检测到总线错误
7	检测到配置错误	检测到配置错误
8	–	–
9	–	配置初始化
10	–	传感器电源电压低
11	–	执行器电源电压低
12	–	检测到 Output0 短路或过流
13	–	检测到 Output1 短路或过流
14	–	–
15	–	–

添加 EXPERTGetDiag 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →管理 →EXPERTGetDiag**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

EXPERTGetImmediateValue：读取 HSC 的计数器值

功能描述

此管理功能允许绕过控制器循环读取 HSC 的计数器值。

图形表示形式

IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表介绍输入变量：

输入	类型	注释
EXPERT_REF_IN	EXPERT_REF（参见第 *150* 页）	参考专用功能块。
Execute	BOOL	在上升沿启动功能块的执行。在下降沿上，则在其执行终结时，复位功能块的输出。

下表介绍输出变量：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF（参见第 *150* 页）	参考专用功能块。
Done	BOOL	TRUE = 表示 ExpertDiag 有效。 功能块执行结束。
Error	BOOL	TRUE = 表示检测到一个错误。
ErrID	IMMEDIATE_FUNC_ERR_TYPE（参见第 *147* 页）	当 Error 为 TRUE 时：检测到的错误的类型。
ImmediateValue	DINT	包含计数器值。

添加 EXPERTGetImmediateValue 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →管理 →EXPERTGetImmediateValue**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

EXPERTGetParam：返回主体专用 I/O 功能 的参数

功能描述

此管理功能块返回 HSC 的参数值。

图形表示形式

IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表对输入变量进行了描述：

输入	类型	注释
EXPERT_REF_IN	EXPERT_REF（参见第 *150* 页）	专用功能块的参考。 在功能块执行期间，不得进行更改。
Execute	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。
Param	EXPERT_PARAMETER_TYPE（参见第 *148* 页）	要读取的参数。

下表对输出变量进行了描述：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF (参见第 *150* 页)	专用功能块的参考。
Done	BOOL	TRUE = 表示 ParamValue 有效。 功能块执行结束。
Busy	BOOL	TRUE = 表示功能块执行正在进行中。
Error	BOOL	TRUE = 表示检测到一个错误。 功能块执行结束。
ErrID	EXPERT_ERR_TYPE (参见第 *144* 页)	当 Error 为 TRUE 时：检测到的错误的类型。
ParamValue	DINT	已读取的参数值。

注意：有关 Done、 Busy 和 Execution 引脚的详细信息，请参阅功能块管理的一般信息 (参见第 *141* 页)。

添加 EXPERTGetParam 功能块

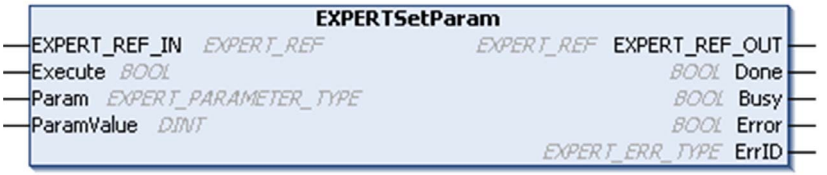
步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 →M241 →M241 HSC →管理 →EXPERTGetParam**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

EXPERTSetParam：调整 HSC 的参数

功能描述

此管理功能块修改 HSC 的参数值。

图形表示形式



IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表对输入变量进行了描述：

输入	类型	注释
EXPERT_REF_IN	EXPERT_REF（参见第 *150* 页）	专用功能块的参考。 在功能块执行期间，不得进行更改。
Execute	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。
Param	EXPERT_PARAMETER_TYPE（参见第 *148* 页）	要读取的参数。
ParamValue	DINT	要写入的参数值。

下表对输出变量进行了描述：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF (参见第 *150* 页)	专用功能块的参考。
Done	BOOL	TRUE = 表示参数已成功写入。 功能块执行结束。
Busy	BOOL	TRUE = 表示功能块执行正在进行中。
Error	BOOL	TRUE = 表示检测到一个错误。 功能块执行结束。
ErrID	EXPERT_ERR_TYPE (参见第 *144* 页)	当 Error 为 TRUE 时：检测到的错误的类型。

注意：有关 Done、 Busy 和 Execution 引脚的详细信息，请参阅功能块管理的一般信息 (参见第 *141* 页)。

添加 EXPERTSetParam 功能块

步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器** →**M241** →**M241 HSC** →**管理** →**EXPERTSetParam**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

HSCMain_M241：HSC Main 功能块

功能描述

此功能块通过下列功能控制 **Main** 类型计数器：

- 加 / 减计数
- 频率计
- 阈值
- 事件

使用 **Main** 计数器时，必须使用 HSC Main 功能块。

功能块实例名称必须与配置定义的名称相匹配。此功能块管理的硬件相关信息与 MAST 任务循环同步。

⚠ 警告

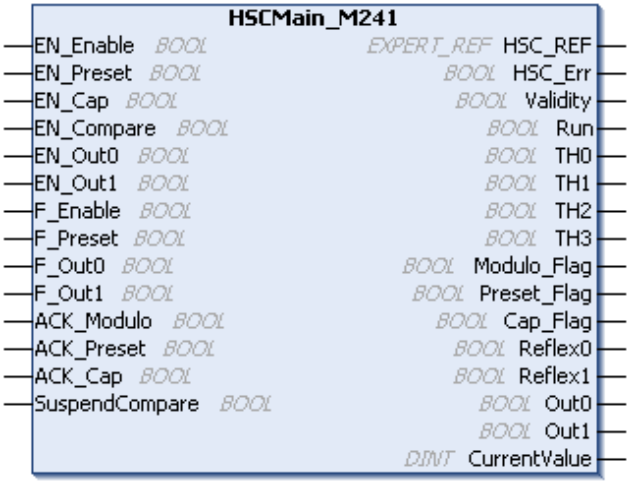
意外的输出值

- 只能在 MAST 任务中使用该功能块实例。
- 请勿在不同的任务中使用同一个功能块实例。

不遵循上述说明可能导致人员伤亡或设备损坏。

注意：SoMachine 允许对 FB 的逻辑输出值进行强制，但是如果此功能处于活动状态 （正在执行），则不会影响硬件的相关输出。

图形表示形式



IL 和 ST 表示形式

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 *169* 页）。

I/O 变量描述

下表介绍输入变量：

输入	类型	描述
EN_Enable	BOOL	TRUE = 启用计数器。
EN_Preset	BOOL	TRUE = 准许通过同步输入进行计数器同步并启动。
EN_Cap	BOOL	TRUE = 启用捕捉输入 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。
EN_Compare	BOOL	TRUE = 启用比较器操作 （使用阈值 0、 1、 2、 3）： ● 基本比较 （TH0、 TH1、 TH2、 TH3 输出位） ● 反射 （Reflex0、 Reflex1 输出位） ● 事件 （在超出阈值时触发外部任务）
EN_Out0	BOOL	TRUE = 启用 Output0 回显 Reflex0 值 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。
EN_Out1	BOOL	TRUE = 启用 Output1 回显 Reflex1 值 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。
F_Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
F_Preset	BOOL	在上升沿时，准许在下列计数模式中同步和启动计数功能： **一次性计数器：** 预设和启动计数器 **模数回路计数器：** 复位和启动计数器 **自由大型计数器：** 预设和启动计数器 **事件计数器：** 在开始时重启内部时基 **频率计：** 在开始时重启内部时基
F_Out0	BOOL	TRUE = 强制 Output0 为 1（如果已在**一次性**、**模数回路**、**自由大型**模式下进行配置）。
F_Out1	BOOL	TRUE = 强制 Output1 为 TRUE （如果已在**一次性**、**模数回路**、**自由大型**模式下进行配置）。
ACK_Modulo	BOOL	在上升沿，复位 Modulo_Flag （**模数回路**和**自由大型**模式）。
ACK_Preset	BOOL	在上升沿，复位 Preset_Flag。
ACK_Cap	BOOL	在上升沿，复位 Cap_Flag （**一次性**、**模数回路**、**自由大型**模式）。
SuspendCompare	BOOL	TRUE = 比较结果已挂起： ● TH0、 TH1、 TH2、 TH3、 Reflex0、 Reflex1、 Out0、 Out1 块输出位保持其上一个值。 ● 物理输出 Output0 和 Output1 保持各自的上一个值。 ● 事件被掩蔽。 **注意：**在设置 SupendCompare 的同时， EN_Compare、 EN_Out0、 EN_Out1、 F_Out0、 F_Out1 保持运行。

下表对输出变量进行了描述：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。 在**周期计类型**下，如果超过了超时值，则 Validity = FALSE。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 HSCGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Run	BOOL	TRUE = 计数器正在运行。 在一次性模式下， CurrentValue 达到 0 时 Run 位切换为 0。
TH0	BOOL	TRUE = 当前计数器值 > **阈值** 0 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。 只有在设置了 EN_Compare 后才处于活动状态。
TH1	BOOL	TRUE = 当前计数器值 > **阈值** 1 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。 只有在设置了 EN_Compare 后才处于活动状态。
TH2	BOOL	TRUE = 当前计数器值 > **阈值** 2 （如果已在**一次性**、**模数回路**、**自由大型**模式下进行配置）。 只有在设置了 EN_Compare 后才处于活动状态。
TH3	BOOL	TRUE = 当前计数器值 > **阈值** 3 （如果已在**一次性**、**模数回路**、**自由大型**模式下进行配置）。 只有在设置了 EN_Compare 后才处于活动状态。
Modulo_Flag	BOOL	通过以下回转设置为 TRUE： ● **模数回路计数器**：当计数器回转到模数或 0 时 ● **自由大型**计数器：当计数器回转其限制时
Preset_Flag	BOOL	通过以下同步设置为 TRUE： ● **一次性**计数器：预设和启动计数器时 ● **模数回路**计数器：当计数器复位时 ● **自由大型**计数器：当预设计数器时 ● **事件**计数器：重新启动与时基对应的内部定时器时 ● **频率计**：重新启动与时基对应的内部定时器时
Cap_Flag	BOOL	TRUE = 表示已在捕捉寄存器中锁存了一个值。 在进行新的捕捉之前，必须先复位此标志。
Reflex0	BOOL	Reflex0 的状态 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。 只有在设置了 EN_Compare 后才处于活动状态。
Reflex1	BOOL	Reflex1 的状态 （如果已在**一次性**、**模数回路**、**自由大型**模式中进行了配置）。 只有在设置了 EN_Compare 后才处于活动状态。

输出	类型	注释
Out0	BOOL	指示 Output0 的状态。
Out1	BOOL	指示 Output1 的状态。
CurrentValue	DINT	计数器的当前计数器值。

HSCSimple_M241：HSC Simple 功能块

功能描述

此功能块通过下列简化的功能控制 **Simple** 类型计数器：

- 单向计数
- 无阈值

使用 **Simple** 计数器类型时， `HSCSimple` 功能块是必需的。

功能块实例名称必须与配置定义的名称相匹配。此功能块管理的硬件相关信息与 MAST 任务循环同步。

⚠ 警告

意外的输出值

- 只能在 MAST 任务中使用该功能块实例。
- 请勿在不同的任务中使用同一个功能块实例。

不遵循上述说明可能导致人员伤亡或设备损坏。

注意：SoMachine 允许对 FB 的逻辑输出值进行强制，但是如果此功能处于活动状态 （正在执行），则不会影响硬件的相关输出。

图形表示形式

IL 和 ST 表示形式

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 *169* 页)。

I/O 变量描述

下表介绍输入变量：

输入	类型	注释
Enable	BOOL	TRUE = 准许对当前计数器值进行更改。
Sync	BOOL	在上升沿预设和启动计数器。
ACK_Modulo	BOOL	在上升沿上，复位模数标志 Modulo_Flag （在模数回路模式中）。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF (参见第 *150* 页)	HSC 的参考。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag (参见第 *154* 页) 功能块获得有关此检测到的错误的详细信息。
Validity	BOOL	TRUE = 表示功能块上的输出值有效。
Run	BOOL	TRUE = 计数器正在运行。 在一次性模式下，当 CurrentValue 达到 0 时切换为 0。需要 Sync 的上升沿来重新启动计数器。
Modulo_Flag	BOOL	通过模数回路计数器回转来设置为 1 （当计数器回转模数时）。
CurrentValue	DWORD	计数器的当前计数值。

附录 D
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	170
如何通过 IL 语言使用功能或功能块	171
如何通过 ST 语言使用功能或功能块	175

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：

```
PROGRAM MyProgram_ST
VAR
    Timer_ON: TON;  // Function Block Instance
    Timer_RunCd: BOOL;
    Timer_PresetValue: TIME := T#5S;
    Timer_Output: BOOL;
    Timer_ElapsedTime: TIME;
END_VAR
```

```
Timer_ON(
    IN:=Timer_RunCd,
    PT:=Timer_PresetValue,
    Q=>Timer_Output,
    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。

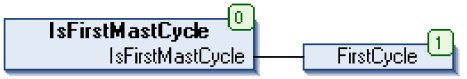
我们以功能 IsFirstMastCycle、功能 SetRTCDrift 和功能块 TON 为例来演示实现的过程。

通过 IL 语言使用功能

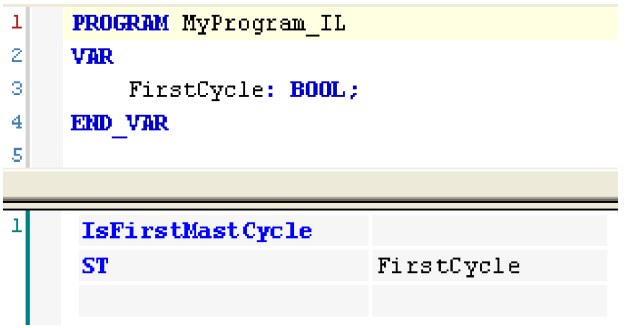
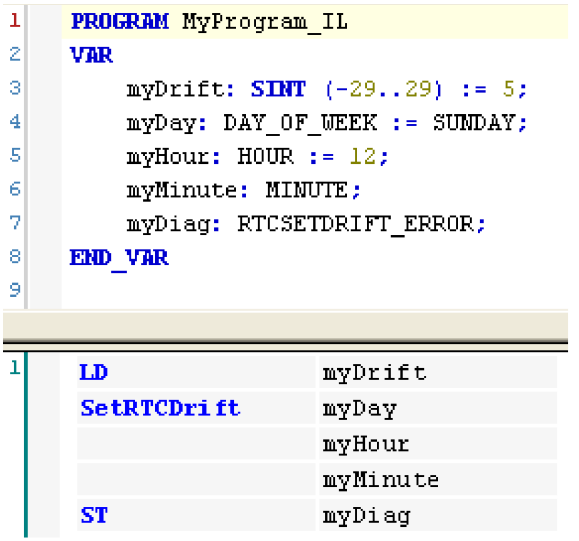
以下过程描述如何用 IL 语言插入一个功能：

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 SoMachine, 编程指南)*。
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	
带输入参数的功能的 IL 示例： SetRTCDrift	

通过 IL 语言使用功能块

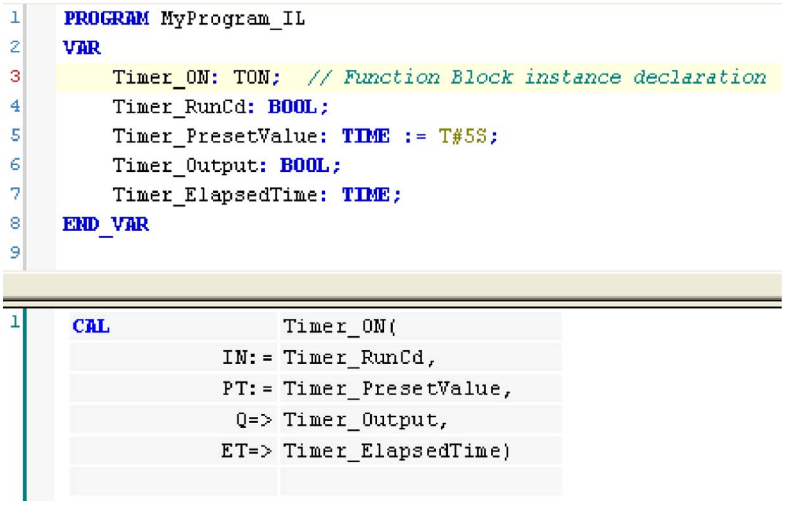
以下过程描述如何用 IL 语言插入一个功能块：

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 SoMachine, 编程指南)*。
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：

功能块	图形表示形式
TON	

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	

```
PROGRAM MyProgram_IL
VAR
    Timer_ON: TON;  // Function Block instance declaration
    Timer_RunCd: BOOL;
    Timer_PresetValue: TIME := T#5S;
    Timer_Output: BOOL;
    Timer_ElapsedTime: TIME;
END_VAR
```

```
CAL             Timer_ON(
            IN:=Timer_RunCd,
            PT:=Timer_PresetValue,
             Q=>Timer_Output,
            ET=>Timer_ElapsedTime)
```

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 SoMachine, 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	`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);`

通过 ST 语言使用功能块

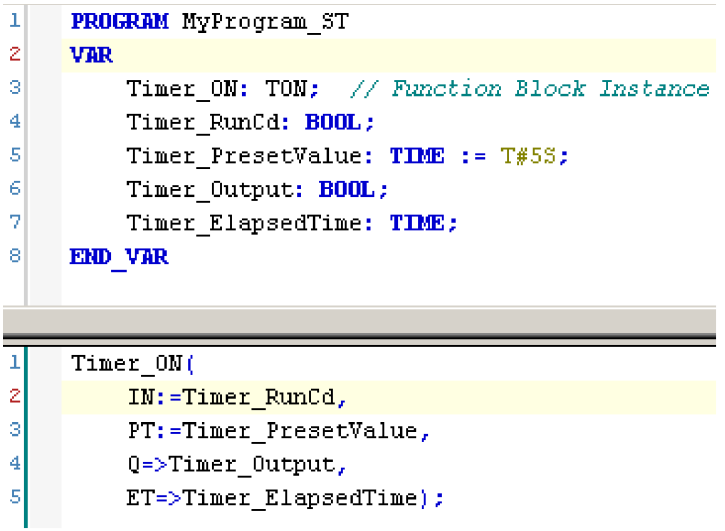
以下过程描述如何用 ST 语言插入一个功能块：

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 SoMachine, 编程指南)*。
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：

功能块	图形表示形式
TON	

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	

```
PROGRAM MyProgram_ST
VAR
    Timer_ON: TON;  // Function Block Instance
    Timer_RunCd: BOOL;
    Timer_PresetValue: TIME := T#5S;
    Timer_Output: BOOL;
    Timer_ElapsedTime: TIME;
END_VAR
```

```
Timer_ON(
    IN:=Timer_RunCd,
    PT:=Timer_PresetValue,
    Q=>Timer_Output,
    ET=>Timer_ElapsedTime);
```

术语

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。

变量

由程序寻址和修改的存储器单元。

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。

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。

节点

通讯网络上的可寻址设备。

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。

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。

HSC

（*高速计数器*）

ID

（*标识符 / 标识*）

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。

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）。

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。

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。

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

索引

Busy
状态变量的管理 , *141*
CommandAborted
状态变量的管理 , *141*
Done
状态变量的管理 , *141*
ErrID
处理检测到的错误 , *141*
状态变量的管理 , *141*
Error
处理检测到的错误 , *141*
状态变量的管理 , *141*
Execute
状态变量的管理 , *141*
EXPERT_ERR_TYPE, *144*
EXPERT_FREQMETER_TIMEBASE_TYPE
数据类型 , *145*
EXPERT_HSCMAIN_TIMEBASE_TYPE
数据类型 , *146*
EXPERT_IMMEDIATE_ERR_TYPE, *147*
EXPERT_PARAMETER_TYPE, *148*
EXPERT_PERIODMETER_RESOLUTION_TYPE
数据类型 , *149*
EXPERTGetCapturedValue, *152*
EXPERTGetDiag, *154*
EXPERTGetImmediateValue, *156*
EXPERTGetParam, *158*
EXPERTSetParam, *160*
HSC_REF, *150*
HSCMain
功能块 , *162*
捕捉 , *128*
比较 , *120*
HSCSimple
功能块 , *166*
专用特性 , *140*
事件计数
嵌入 HSC 的 HSC 模式 , *85*
功能
功能与功能块的区别 , *170*
启用 , *136*
如何通过 IL 语言使用功能或功能块 , *171*
如何通过 ST 语言使用功能或功能块 , *175*
功能块
EXPERTGetCapturedValue, *152*
EXPERTGetDiag, *154*
EXPERTGetImmediateValue, *156*
EXPERTGetParam, *158*
EXPERTSetParam, *160*
HSCMain, *162*
HSCSimple, *166*
启用
功能 , *136*
周期计
参数 , *116*
描述 , *107*
纲要 , *110*
编程 , *113*
处理检测到的错误
ErrID, *141*
Error, *141*
嵌入 HSC 的 HSC 模式
事件计数 , *85*
嵌入式 HSC 的 HSC 模式
模数回路 , *49*
自由大型 , *70*
捕捉
HSCMain, *128*
数据类型
EXPERT_ERR_TYPE, *144*
EXPERT_FREQMETER_TIMEBASE_TYPE, *145*
EXPERT_HSCMAIN_TIMEBASE_TYPE, *146*
EXPERT_IMMEDIATE_ERR_TYPE, *147*
EXPERT_PARAMETER_TYPE, *148*
EXPERT_PERIODMETER_RESOLUTION_TYPE, *149*
HSC_REF, *150*

模数回路
    嵌入式 HSC 的 HSC 模式 , *49*
比较
    HSCMain, *120*
状态变量的管理
    Busy, *141*
    CommandAborted, *141*
    Done, *141*
    ErrID, *141*
    Error, *141*
    Execute, *141*
自由大型
    嵌入式 HSC 的 HSC 模式 , *70*
计数器功能块
    HSCMain, *162*
    HSCSimple, *166*
诊断功能
    EXPERTGetDiag, *154*
调整功能
    EXPERTGetParam, *158*
    EXPERTSetParam, *160*
频率计
    描述 , *97*
    纲要 , *100*
    编程 , *102*