Modicon M258 Logic Controller

高速计数

M258 Expert I/O 库指南

03/2018

EIO0000000577.07

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。任何此类用户或设备集成商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。

本手册可用于法律所界定的个人以及非商业用途。 在未获得施耐德电气书面授权的情况下，不得翻印传播本手册全部或部分相关内容、亦不可建立任何有关本手册或其内容的超文本链接。施耐德电气不对个人和非商业机构进行非独占许可以外的授权或许可。 请遵照本手册或其内容原义并自负风险。与此有关的所有其他权利均由施耐德电气保留。

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，只允许制造商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。

未能使用施耐德电气软件或认可的软件配合我们的硬件，则可能导致人身伤害、设备损坏或不正确的运行结果。

不遵守此信息可能导致人身伤害或设备损坏。

© 2018 Schneider Electric。 保留所有权利。

目录

安全信息 7
关于本书 9
第I部分 高速计数器和编码器概述 11
第1章 简介 13
专用 I/O 概述 14
添加专用功能 17
嵌入式专用 I/O 映射 19
第2章 高速计数器概述 21
Simple 类型概述 22
Main 类型概述 23
选择 HSC 24
第3章 编码器概述 27
标准编码器概述 27
第II部分 一次性模式 29
第4章 一次性模式原理 31
One-shot 模式原理介绍 31
第5章 使用 Simple 类型的一次性 33
纲要图 34
一次性模式下的 simple 类型的配置 35
Simple 类型编程 36
调整参数 38
第6章 使用 Main 类型的一次性 39
纲要图 40
一次性模式下的 Main 类型的配置 41
Main 类型编程 42
调整参数 45
第III部分 模数回路模式 47
第7章 模数回路原理 49
Modulo-loop 模式原理介绍 49
第8章 使用 Simple 类型的模数回路 53
纲要图 54
模数回路模式下的 Simple 类型的配置 55
Simple 类型编程 56
调整参数 58

第9章 使用 Main 类型的模数回路 59
纲要图 60
模数回路模式下的 Main 类型配置 61
Main 类型编程 63
调整参数 66
第IV部分 自由大型模式 67
第10章 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
第11章 使用 Main 类型的自由大型 75
纲要图 76
自由大型模式下 Main 类型的配置 77
Main 类型编程 78
调整参数 81
第V部分 事件计数模式 83
第12章 事件计数原理 85
事件计数模式原理介绍 85
第13章 使用 Main 类型的事件计数 87
纲要图 88
事件计数模式下 Main 类型的配置 89
Main 类型编程 90
调整参数 93
第VI部分 频率计类型 95
第14章 频率计原理 97
描述 97
第15章 频率计类型 99
纲要图 100
配置 101
编程 102
第VII部分 周期计类型 105
第16章 周期计类型原理 107
描述 107
第17章 周期计类型 109
纲要图 110
配置 111
编程 112
调整参数 115

第VIII部分 编码器 117
第18章 使用编码器的递增模式 119
递增模式原理介绍 120
纲要图 124
专用 I/O 模块上的标准编码器配置 125
标准编码器编程 128
调整参数 131
第IX部分 可选功能 133
第19章 比较功能 135
19.1 使用 Main 类型的比较 136
使用 Main 类型或编码器的比较原理 137
对使用 Main 类型或编码器的比较进行配置 140
外部事件配置 141
第20章 捕捉功能 143
20.1 使用 Main 类型的捕捉 144
使用 Main 类型的捕捉原理 145
对使用 Main 类型的捕捉进行配置 146
20.2 使用编码器捕捉 147
使用编码器捕捉 148
编码器上的捕捉配置 150
第21章 预设和启用功能 151
预设功能 152
自由大型或周期计预设条件 154
启用：准许计数操作 156
附录 157
附录 A 一般信息 159
专用特性 160
有关管理和运动功能块管理的一般信息 161
附录 B 数据类型 163
IMMEDIATE_ERR_TYPE：GetImmediateValue 功能块的错误变量类型 164
EXPERT_ERR_TYPE：EXPERT 功能块的错误变量类型 165
EXPERT_PARAMETER_TYPE：在EXPERT功能块上执行获取或设置的参数类型 166
EXPERT_REF：EXPERT 参考值 167
EXPERT_TIMEBASE_TYPE：HSC 时基变量的类型 168

附录 C 功能块 169
EXPERTGetImmediateValue：读取 HSC 或编码器功能的计数器值 . . 170
EXPERTGetCapturedValue:返回捕捉寄存器的内容 172
EXPERTGetDiag：提供主 EXPERT I/O 功能相关检出错误的详细信息 174
EXPERTGetParam：返回主 EXPERT I/O 功能的参数 176
EXPERTSetParam：调节 HSC 的参数 178
Encoder_M258：编码器功能块 180
HSCMain: HSC Main 功能块 183
HSCSimple_M258：控制 M258 的 Simple 类型计数器 187
附录 D 功能和功能块表示形式 189
功能与功能块的区别 190
如何通过 IL 语言使用功能或功能块 191
如何通过 ST 语言使用功能或功能块 195
术语表 199
索引 203

安全信息

重要信息

声明

在试图安装、操作、维修或维护设备之前，请仔细阅读下述说明并通过查看来熟悉设备。下述特定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”或“警告”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

危险表示若不加以避免，将**会导致**严重人身伤害甚至死亡的危险情况。

⚠ 警告

警告表示若不加以避免，可能**会导致**严重人身伤害甚至死亡的危险情况。

⚠ 小心

小心表示若不加以避免，可能**会导致**轻微或中度人身伤害的危险情况。

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有资质的人员执行。施耐德电气不承担由于使用本资料所引起的任何后果。

有资质的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 M258 Controller 中提供的 High Speed Counter (HSC) 功能和变量。

本文档介绍 M258 HSC 库的功能和变量。

要使用本手册，您必须：

- 对 M258 有着透彻理解，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)

只有 DM72F• 专用模块可与 M258 HSC 库配合使用。

有效性说明

本文档已随 SoMachine V4.3 TM3TI4D 附加程序的发布进行了更新。

相关的文件

文件名称	参考编号
Modicon M258 Logic Controller - 编程指南	*EIO0000000402 (Eng)*、*EIO0000000403 (Fre)*、*EIO0000000404 (Ger)*、*EIO0000000405 (Spa)*、*EIO0000000406 (Ita)*、*EIO0000000407 (Chs)*
Modicon M258 Logic Controller 硬件指南	*EIO0000000432 (Eng)*、*EIO0000000433 (Fre)*、*EIO0000000434 (Ger)*、*EIO0000000435 (Spa)*、*EIO0000000436 (Ita)*、*EIO0000000437 (Chs)*

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：https://www.schneider-electric.com/en/download

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。

第I部分
高速计数器和编码器概述

概述

本部分概述不同的 HSC 类型和编码器类型，并介绍其可用模式、功能及性能。

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	简介	13
2	高速计数器概述	21
3	编码器概述	27

第1章
简介

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
添加专用功能	17
嵌入式专用 I/O 映射	19

专用 I/O 概述

简介

控制器本体提供：

- 2 个嵌入式专用 I/O 模块（DM72F0 和 DM72F1），该模块具有：
  - 5 路快速输入
  - 2 路常规输入
  - 2 路快速输出
- 1 个控制器配电模块 (CPDM)

每个嵌入式专用 I/O 模块 (DM72F•) 均支持专用功能 (参见第 *17* 页)。

嵌入式专用 I/O 配置

要配置专用 I/O，请双击**设备树**中的**专用**节点。

下图提供配置选项卡屏幕截图：



下表提供不同参数的功能：

参数	功能
Run/Stop Input	定义一路要作为运行/停止输入 (参见第 *15* 页)的输入。
Alarm Output	定义一路要作为警报输出 (参见第 *15* 页)的输出。
Rearming Output Mode	定义重置输出模式 (参见第 *16* 页)。

运行/停止输入

下表提供不同状态：

输入状态	结果
状态 0	停止控制器并忽略外部运行命令。
上升沿	从“已停止”状态，触发启动“运行”状态下的应用程序。
状态 1	应用程序可由以下各项控制： ● SoMachine（运行/停止） ● 应用程序（控制器命令） ● 网络命令

注意： 即使没有在 PLC 设置选项卡 *(参见 Modicon M258 Logic Controller, 编程指南)*中选中**停止时更新 I/O** 选项，也会管理运行/停止输入。
分配到已配置专用功能的输入无法配置为运行/停止。

有关控制器状态和状态转变的详细信息，请参阅控制器状态图 *(参见 Modicon M258 Logic Controller, 编程指南)*。

⚠ 警告

机器或过程意外启动

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

不遵循上述说明可能导致人员伤亡或设备损坏。

警报输出

当控制器处于“运行”状态且应用程序未在断点处停止时，此输出设置为逻辑 1。

分配用于配置专用功能的输出无法配置为警报输出。

注意：
在以下情况下将警报输出设置为 0：
- 当任务在断点处停止时，警报输出会发出控制器已经停止执行应用程序的信号。
- 在专用 I/O 上检测到错误（电源中断、短路检测）。

重置输出模式

DM72F• 模块的快速输出采用的是推/挽技术。在检测到错误（短路或过热）时，输出会进入三态，其状态将由状态位（DM72F• 通道 IB1.0）和 PLC_R.i_wLocalIOStatus 显示。

行为可以分为以下两种：

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

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

⚠ 警告
机器意外启动 如果不想对机器或过程执行输出的自动重置，请禁用此功能。 **不遵循上述说明可能导致人员伤亡或设备损坏。**

添加专用功能

简介

每个 DM72F• 专用模块都可支持专用功能。专用功能可以定义为简单或复杂功能。每个模块只能配置一种类型：

- 简单功能：
  - 高速计数器 (Simple)
  - Event_Latch I/O
- 复杂功能：
  - 高速计数器 (Main)
  - 编码器
  - 频率发生器 (FreqGen)
  - 脉冲宽度调制 (PWM)

如果某个 I/O 未被专用功能使用，则可作为常规 I/O 使用。

注意：

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

有关详细信息，请参阅嵌入式专用 I/O 配置 (参见第 *14* 页)。

添加专用功能

要将专用功能（Event_Latch、HSC、PWM 或频率发生器）添加到您的控制器，请在**硬件目录**中选择您需要的专用功能，然后将其拖到**设备树**，并放在突出显示的某个节点上。

有关将设备添加到项目的更多信息，请参阅：

• 使用拖放方法 *(参见 SoMachine, 编程指南)*

• 使用上下文菜单或加号按钮 *(参见 SoMachine, 编程指南)*

要添加编码器功能，请在**硬件目录**中选择**标准编码器**，然后将其拖到**设备树**，并放在突出显示的某个节点上。

可以添加下列专用功能：

功能	描述	请参阅...
Event_Latch	借助 Event_Latch 功能，可以将嵌入式专用输入配置为事件或锁存。	Event_Latch 配置 *(参见 Modicon M258 Logic Controller, 编程指南)*
HSC	HSC 功能可以对来自与专用快速输入连接的传感器、编码器、开关等的脉冲进行快速计数。	M258 HSC 库 (参见第 页)

功能	描述	请参阅...
PWM 频率发生器	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。	M258 PWM 库 *(参见 Modicon M258 Logic Controller, 脉冲宽度调制, M258 Expert I/O 库指南)*
编码器	此功能的目标是连接解码器以获取位置。 此功能可安装到嵌入式专用 I/O 接口上，且仅支持递增编码器。您可以配置线性轴或旋转轴。	M258 HSC 库 (参见第 页)

专用功能分配

专用功能的分配以接口为标准（各列互不包含）：

I/F 接口	专用功能				
	简单功能： ● 快速 I/O：事件或锁存 ● HSC Simple	HSC_Main	编码器	PWM	频率发生器
DM72F0	最多 4 个	1	1	1	1
DM72F1	最多 4 个	1	1	1	1

有关详细信息，请参阅专用 I/O 映射 (参见第 *19* 页)。

常规 I/O 中的专用功能 I/O

常规 I/O 中的专用功能 I/O:

- 即使是在专用功能中配置的输入，也可通过存储器变量标准进行读取。
- 如果输入已经配置为运行/停止，则不能在专用功能中对其进行配置。
- 如果输出已经配置为“警报”，则不能在专用功能中对其进行配置。
- %Q 对反射输出没有任何影响。
- 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 都可作为快速或常规 I/O 使用。

在专用功能（锁存、HSC…）中使用输入时，积分器过滤器将被替换为抗跳动过滤器 *(参见 Modicon M258, Logic Controller, 硬件指南)*。过滤器值将在专用功能屏幕中进行配置。

嵌入式专用 I/O 映射

用于 DM72F•上专用功能的 I/O 映射

专用功能的嵌入式专用 I/O 映射：

		I0	I1	I2	I3	I4	I5	Q0	Q1
Event_Latch 0/4	输入	M							
Event_Latch 1/5	输入		M						
Event_Latch 2/6	输入			M					
Event_Latch 3/7	输入				M				
HSC Simple 0/4	输入 A	M							
HSC Simple 1/5	输入 A		M						
HSC Simple 2/6	输入 A			M					
HSC Simple 3/7	输入 A				M				
HSC Main 0/1	输入 A	M							
	输入 B		C						
	SYNC			C					
	CAP				C				
	EN					C			
	REF						C		
	输出							C	C
PWM 0/1	输出							M	
	SYNC			C					
	EN					C			
频率发生器 0/1	输出							M	
	SYNC			C					
	EN					C			
标准编码器	输入 A	M							
	输入 B		M						
	SYNC			C					
	CAP				C				
	EN					C			
	REF						C		
	输出							C	C

M 强制
C 取决于配置

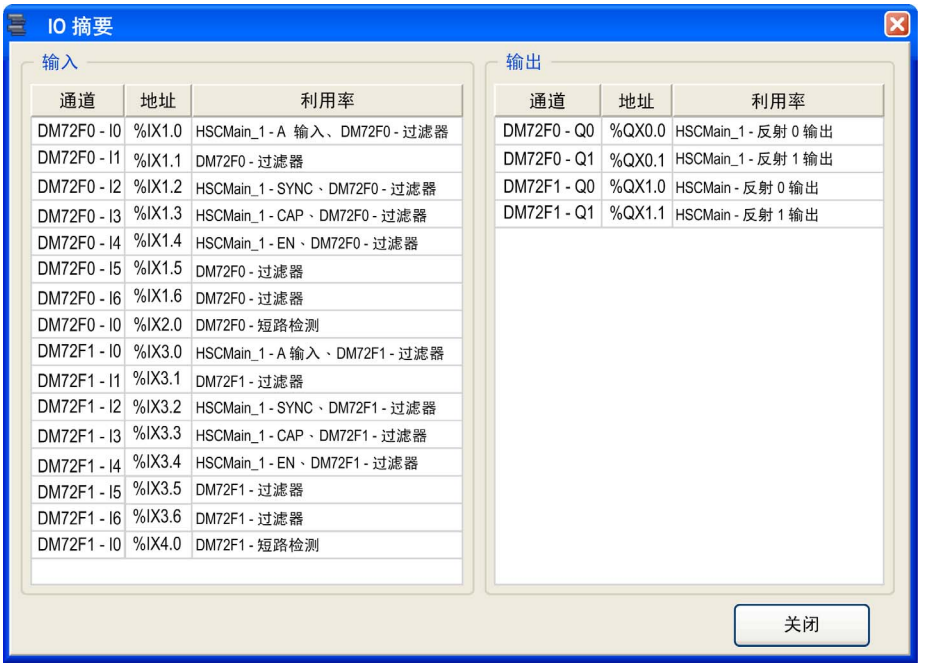
IO 摘要

IO 摘要窗口显示 DM72F• I/O 和专用功能使用的 I/O。

可通过 **DM72F•** 节点访问**IO 摘要**：

步骤	操作
1	在**设备树**选项卡中，展开**专用**节点。
2	双击 **DM72F•**，然后在上下文菜单中选择 **IO 摘要**。

IO 摘要的示例：



第2章
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
Simple 类型概述	22
Main 类型概述	23
选择 HSC	24

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，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。

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

专用 I/O 接口上允许的最大频率是 100 kHz。

有关跳动过滤器的更多信息，请参阅专用功能 (参见第 *160* 页)。

Main 类型概述

概述

Main 类型计数器最多可使用 6 个快速输入和 2 个反射输出。

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。可与一个编码器配合使用。

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。

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)。

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

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硬件输入
- 最多 4 个阈值
- 最多 4 个事件（每个阈值占 1 个）可与外部任务进行关联
- 最多 2 个反射输出

性能

专用 I/O 接口上允许的最大频率是 100 kHz。

选择 HSC

HSC 阵列

下表概述了所有可用 HSC 及其在所请求模式下的各种规格：

模式	功能	Simple 类型	Main 类型
一次性	计数模式	减计数	减计数
	通过 HSC 物理输入启用	否	是
	通过 HSC 物理输入同步/复位	否	是
	比较功能	否	是，4 个阈值，2 个输出和事件
	捕捉功能	否	是，1 个捕捉寄存器
	配置调节	-	停止事件
模数回路	计数模式	减计数	单相 加/减计数 脉冲/方向 积分
	通过 HSC 物理输入启用	否	是
	通过 HSC 物理输入同步/复位	否	是
	比较功能	否	是，4 个阈值，2 个输出和事件
	捕捉功能	否	是，1 个捕捉寄存器
	配置调节	-	-
自由大型	计数模式	-	加/减计数 脉冲/方向 积分
	通过 HSC 物理输入启用	-	是
	通过 HSC 物理输入同步/复位	-	是
	比较功能	-	是，4 个阈值，2 个输出和事件
	捕捉功能	-	是，1 个捕捉寄存器
	配置调节	-	限制管理
事件	计数模式	-	给定时间段内的脉冲计数
	通过 HSC 物理输入启用	-	是
	通过 HSC 物理输入同步/复位	-	是
	比较功能	-	否
	捕捉功能	-	否
	配置调节	-	时基

模式	功能	Simple 类型	Main 类型
频率计	计数模式	-	时基脉冲计数
	通过 HSC 物理输入启用	-	是
	通过 HSC 物理输入同步/复位	-	是
	比较功能	-	否
	捕捉功能	-	否
	配置调节	-	-
周期计	计数模式	-	时基脉冲计数
	通过 HSC 物理输入启用	-	是
	通过 HSC 物理输入同步/复位	-	是
	比较功能	-	否
	捕捉功能	-	否
	配置调节	-	精度 超时

第3章
编码器概述

标准编码器概述

概述

此功能的目标是连接解码器以获取位置。

此功能可以安装在嵌入式**专用 I/O** 模块（DM72F0 和 DM72F1） 上。

标准编码器模式

在**专用 I/O** 接口上实现时，编码器仅支持递增模式。

在**编码器**接口上，电源被监控。检测到的电源错误会在电源恢复时自动清除。

可选功能

可选功能的配置取决于选择的模式：

- 硬件输入，以对计数器进行操作（启用、预设）或捕捉当前计数值
- 最多 4 个阈值
- 最多 4 个事件（每个阈值触发 1 次）可与外部任务进行关联
- 最多 2 个反射输出

性能

专用 I/O 模块上的最大允许频率为 100 kHz。

第II部分
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
4	**一次性**模式原理	31
5	使用 **Simple** 类型的**一次性**	33
6	使用 **Main** 类型的**一次性**	39

第4章
一次性模式原理

One-shot 模式原理介绍

概述

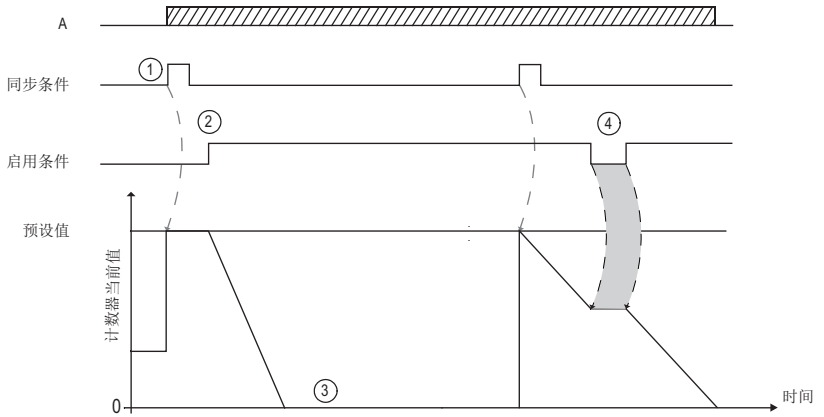
计数器由同步跳变沿激活，同时加载预设值。

启用计数时，每个应用到输入的脉冲都递减当前值。在当前值达到 0 时计数器停止。

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 *156* 页)和预设 (参见第 *152* 页)功能中对这些条件进行了介绍。

第5章
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？

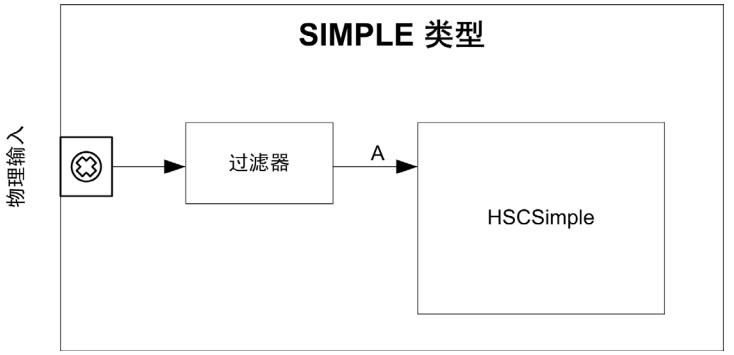
本章包含了以下主题：

主题	页
纲要图	34
一次性模式下的 **simple** 类型的配置	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** 计数类型仅加计数。

一次性模式下的 simple 类型的配置

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**专用 → DM72F• → HSCSimple•**。
2	选择 **HSC Simple 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**一次性**。
4	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
5	输入**范围 → 预设**参数的值。

注意： **IO 摘要**窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Simple 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒	200 kHz
	0.004 毫秒（缺省值）	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

Simple 类型编程

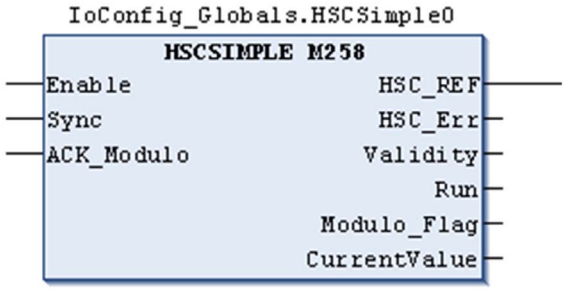
概述

Simple 类型始终由 HSCSimple 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 → M258 → M258 专用 IO → HSC → HSCSimple_M258**，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 逻辑 → 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	注释
EN_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

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF （参见第 *167* 页）	HSC 的参考。 与EXPERT_REF_IN管理**功能块的** 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 *174* 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。 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。
CurrentValue	DWORD	计数器的当前计数值。
Modulo_Flag	BOOL	不相关

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam（参见第 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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

第6章
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？

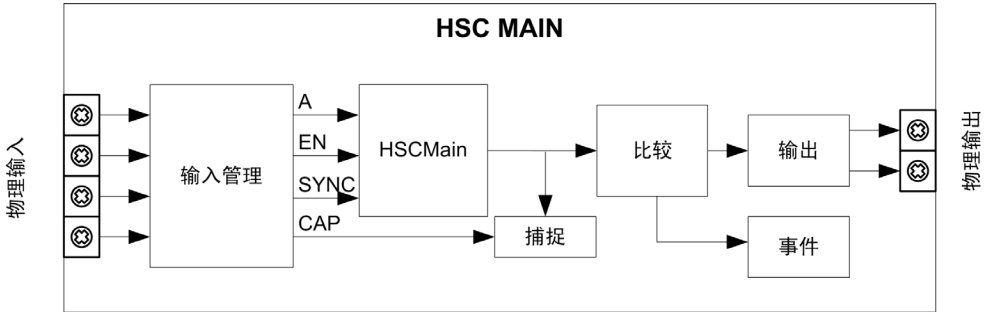
本章包含了以下主题：

主题	页
纲要图	40
一次性模式下的 **Main** 类型的配置	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 *135* 页)
- 捕捉功能 (参见第 *143* 页)
- 预设功能 (参见第 *152* 页)
- 启用功能 (参见第 *156* 页)

一次性模式下的 Main 类型的配置

配置过程

按照以下过程配置**一次性**模式下的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**一次性**。
4	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
5	输入**范围 → 预设**参数的值。
6	或者，从下拉菜单中选择 **SYNC**、**EN** 和 **CAP** 辅助输入的值，以使用物理输入启用预设功能 (参见第 *152* 页)、启用功能 (参见第 *156* 页)和捕捉功能 (参见第 *144* 页)。
7	或者，选择**阈值数**。这样可以启用比较功能并可以配置反射输出 (参见第 *135* 页)。
8	将**停止事件**设置为“是”时，必须使用外部事件（BLOCK0_HSCSTOP 或 BLOCK1_HSCSTOP）触发外部任务 (参见第 *141* 页)。

注意： IO 摘要窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Main 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒（缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

Main 类型编程

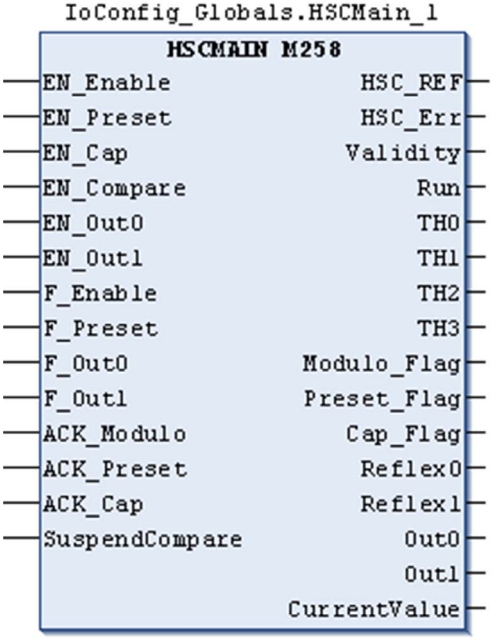
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**， 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 逻辑 → 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 *156* 页)启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *152* 页)同步和启动计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入 (参见第 *144* 页)。
EN_Compare	BOOL	TRUE = 启用比较器操作 (参见第 *135* 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 Output0 回显 Reflex0 值（如果已配置）。
EN_Out1	BOOL	TRUE = 启用物理输出 Output1 回显 Reflex1 值（如果已配置）。
F_Enable	BOOL	强制实施启用条件 (参见第 *156* 页)。优先于 EN_Enable 输入。
F_Preset	BOOL	强制实施预设条件 (参见第 *152* 页)。优先于 EN_Preset 输入。
F_Out0	BOOL	TRUE = 强制 Output0 为 1（如果 Reflex0 已配置在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中。优先于 EN_Out0。
F_Out1	BOOL	TRUE = 强制 Output1 为 1（如果 Reflex1 已配置在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中）。优先于 EN_Out1。
ACK_Modulo	BOOL	在上升沿，复位模数标志。
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。 ● 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 SuspendCompare 时，EN_Compare、EN_Out0、EN_Out1、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 *167* 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 功能块获得有关此检测到的错误的详细信息。 （参见第 *174* 页）
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。 当 CurrentValue 达到 0 时切换为 0。要重新启动计数器，则需要 Sync 存在上升沿。
TH0	BOOL	当 CurrentValue > 阈值 0 时，设为 1 (参见第 *135* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *135* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *135* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *135* 页)。
Modulo_Flag	BOOL	计数器达到 0 时设置为 1。
Preset_Flag	BOOL	在计数器预设 (参见第 *152* 页)时，设为 1。
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器 (参见第 *144* 页)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 (参见第 *138* 页)。
Reflex1	BOOL	Reflex1 的状态 (参见第 *138* 页)。
Out0	BOOL	物理输出 Output0 的状态设置为 1（如果已在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中配置 Reflex0，否则在未配置时设置为 FALSE）。
Out1	BOOL	物理输出 Output1 的状态设置为 1（如果已在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中配置 Reflex1，否则在未配置时设置为 FALSE）。
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 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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_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

第III部分
模数回路模式

概述

本部分介绍在**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
7	**模数回路**原理	49
8	使用 **Simple** 类型的**模数回路**	53
9	使用 **Main** 类型的**模数回路**	59

第7章
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 *152* 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 *156* 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。

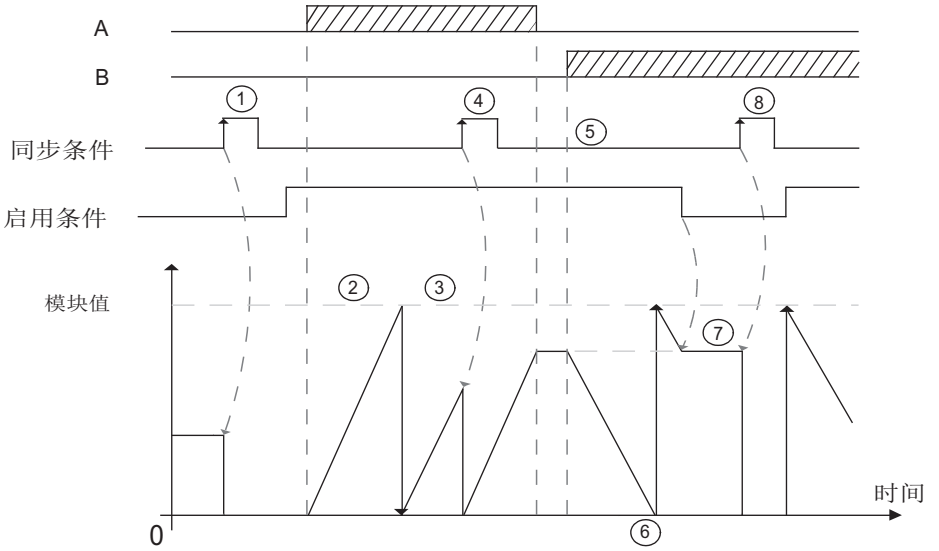
递减方向： 计数器在达到 0 之前一直递减。在下一个脉冲，计数器设置为模数值，模数标志设置为 1 并继续计数。

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	默认模式 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

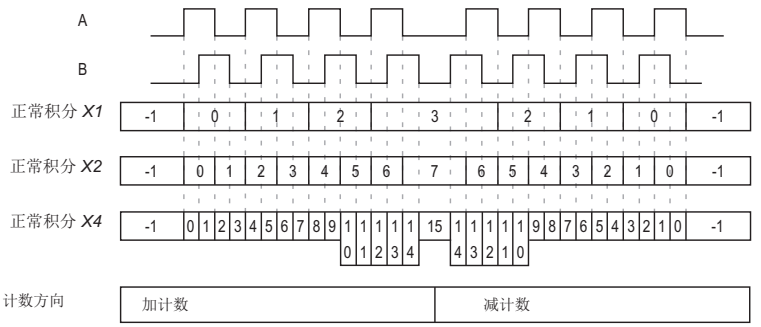


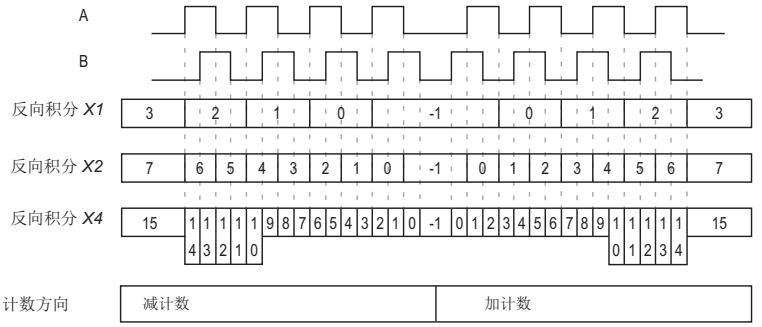
阶段	操作
1	在同步条件的上升沿，当前值复位为 0 并激活计数器。
2	当启用条件 = 1 时，A 上的每个脉冲会递增计数器值。
3	计数器达到模数值减 1 时，计数器在下一个脉冲返回 0，并继续计数。Modulo_Flag 设置为 1。
4	在同步条件的上升沿，当前计数器值复位为 0。
5	当启用条件 = 1 时，B 上的每个脉冲会递减计数器。
6	计数器达到 0 时，计数器在下一个脉冲返回模数值减 1，并继续计数。
7	启用条件 = 0 时，忽略输入上的脉冲。
8	在同步条件的上升沿，当前计数器值复位为 0。

注意： 启用和同步条件取决于配置。启用 (参见第 *156* 页)和预设 (参见第 *152* 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：





第8章
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？

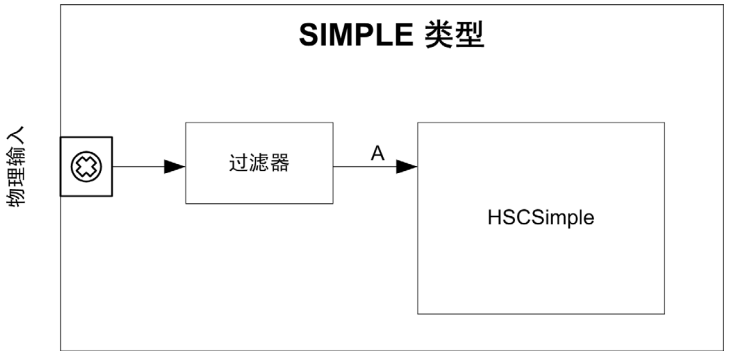
本章包含了以下主题：

主题	页
纲要图	54
模数回路模式下的 Simple 类型的配置	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：



A 是 High Speed Counter 的计数输入。

Simple 类型只能进行加计数。**模数回路**模式的 **Simple** 计数类型仅减计数。**一次性**模式的 **Simple** 计数类型仅加计数。

模数回路模式下的 Simple 类型的配置

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**专用 → DM72F• → HSCSimple•**。
2	选择 **HSC Simple 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**模数回路**。
4	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
5	输入**范围 → 模数**参数的值。

注意： **IO 摘要**窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Simple 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒	200 kHz
	0.004 毫秒 （缺省值）	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

Simple 类型编程

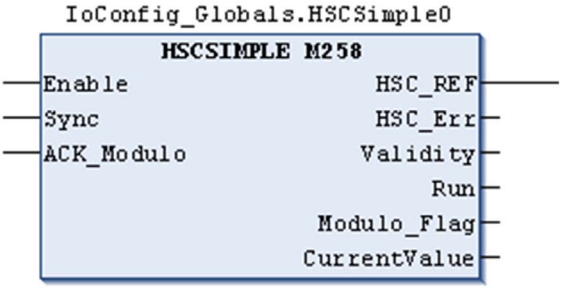
概述

Simple 类型始终由 HSCSimple 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 → M258 → M258 专用 IO → HSC → HSCSimple_M258**，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 逻辑 → 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	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，复位 Modulo_Flag。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF （参见第 *167* 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 *174* 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 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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

第9章
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？

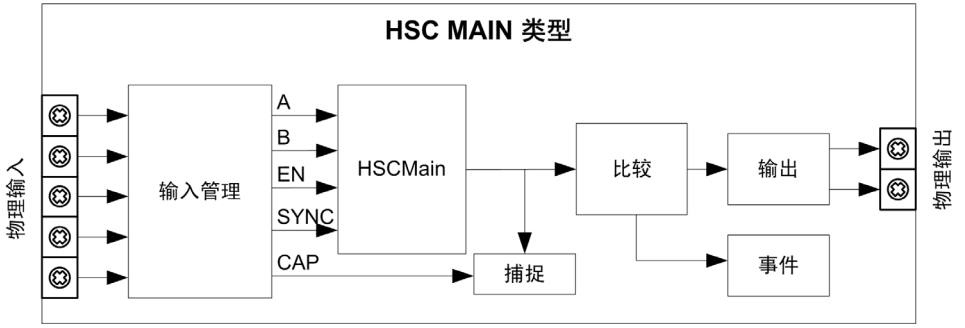
本章包含了以下主题：

主题	页
纲要图	60
模数回路模式下的 Main 类型配置	61
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 *135* 页)
- 捕捉功能 (参见第 *143* 页)
- 启用功能 (参见第 *156* 页)

注意： 预设值为 0，无法修改。

模数回路模式下的 Main 类型配置

配置过程

按照以下过程配置**模数回路**模式下的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**模数回路**。
4	选择**常规 → 输入模式**参数的值。
5	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。 如有必要，选择**计数输入 → B 输入 → 跳动过滤器**参数的值。
6	输入**范围 → 模数**参数的值。
7	或者，可以选择启用以下功能： ● 捕捉功能 (参见第 *143* 页) ● 比较功能 (参见第 *135* 页) ● 启用功能 (参见第 *156* 页) ● 预设功能 (参见第 *152* 页)

注意： **IO 摘要**窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Main 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A, B	0.002 毫秒 （缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

输入参数

输入模式参数的值确定计数行为：

参数值	描述
单相	计数器在 A 上递增。
A=向上，B=向下	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 = TRUE，则计数器递减。 ● 如果 A 上存在上升沿且 B = 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 ● 通过编码器循环进行 1 次计数
正常积分 X2	● 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 ● 通过编码器循环进行 2 次计数
正常积分 X4	● 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 ● 通过编码器循环进行 4 次计数
反向积分 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 ● 通过编码器循环进行 1 次计数
反向积分 X2	● 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 ● 通过编码器循环进行 2 次计数
反向积分 X4	● 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 ● 通过编码器循环进行 4 次计数

Main 类型编程

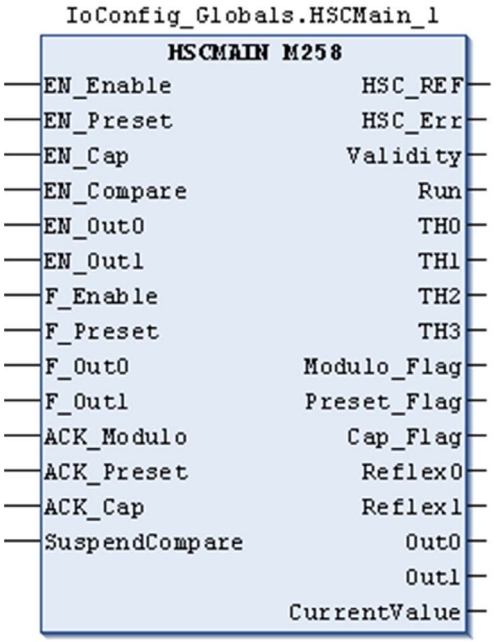
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**，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 逻辑 → 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 *156* 页)启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *152* 页)同步和启动计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入 (参见第 *144* 页)。
EN_Compare	BOOL	TRUE = 启用比较功能 (参见第 *135* 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 Output0 回显 Reflex0 值（如果已配置）。
EN_Out1	BOOL	TRUE = 启用物理输出 Output1 回显 Reflex1 值（如果已配置）。
F_Enable	BOOL	强制实施启用条件 (参见第 *156* 页)。优先于 EN_Enable 输入。
F_Preset	BOOL	强制实施预设条件 (参见第 *152* 页)。优先于 EN_Preset 输入。
F_Out0	BOOL	TRUE = 强制 Output0 为 1（如果 Reflex0 已配置在 SoMachine HSC 内嵌功能中）。优先于 EN_Out0。
F_Out1	BOOL	TRUE = 强制 Output1 为 1（如果 Reflex1 已配置在 SoMachine HSC 内嵌功能中）。优先于 EN_Out1。
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。 ● 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 SuspendCompare 时，EN_Compare、EN_Out0、EN_Out1、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 *167* 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 *174* 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。
TH0	BOOL	当 CurrentValue > 阈值 0 时，设为 1 (参见第 *135* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *135* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *135* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *135* 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 *152* 页)时，设为 1。
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器 (参见第 *145* 页)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 (参见第 *136* 页)。
Reflex1	BOOL	Reflex1 的状态 (参见第 *136* 页)。
Out0	BOOL	物理输出 Output0 的状态设置为 1（如果已在 SoMachine HSC 内嵌功能中配置 Reflex1，否则在未配置时设置为 FALSE）。
Out1	BOOL	物理输出 Output1 的状态设置为 1（如果已在 SoMachine HSC 内嵌功能中配置 Reflex1，否则在未配置时设置为 FALSE）。
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 *176* 页）或 EXPERTSetParam（参见第 *176* 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 M258 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 *166* 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_MODUL0	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

第IV部分
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
10	自由大型模式原理	69
11	使用 Main 类型的自由大型	75

第10章
自由大型模式原理

概述

本章介绍**自由大型**模式原理。

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 *156* 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 *154* 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 *143* 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。

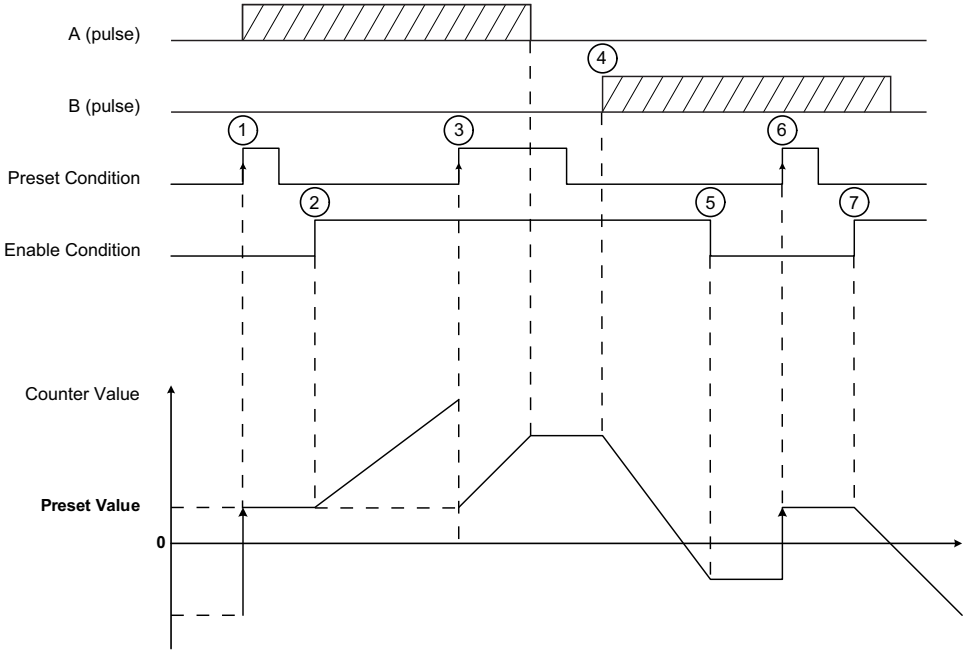
输入模式

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	默认模式 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

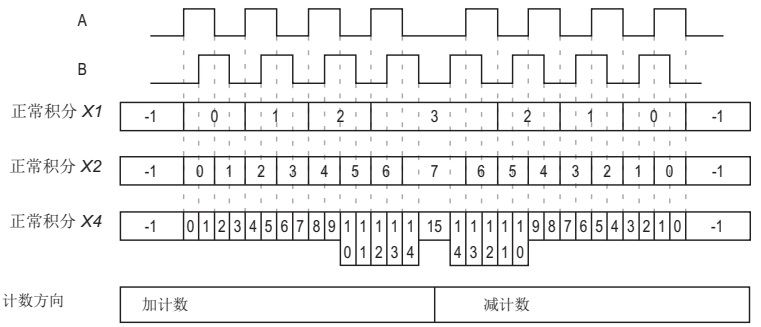
下图显示 **A = 加计数，B = 减计数**模式：

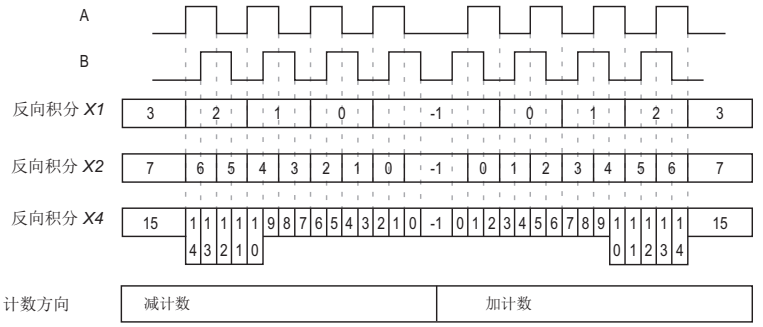


阶段	操作
1	在预设条件的上升沿上，当前值设为预设值，并激活计数器。
2	当启用条件 = 1 时，A 上的每个脉冲会递增计数器值。
3	在预设条件的上升沿上，当前值设为预设值。
4	当启用条件 = 1 时，B 上的每个脉冲会递减计数器值。
5	当启用条件 = 0，忽略 A 或 B 上的脉冲。
6	在预设条件的上升沿上，当前值设为预设值。
7	当启用条件 = 1 时，B 上的脉冲会递减计数器值。

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

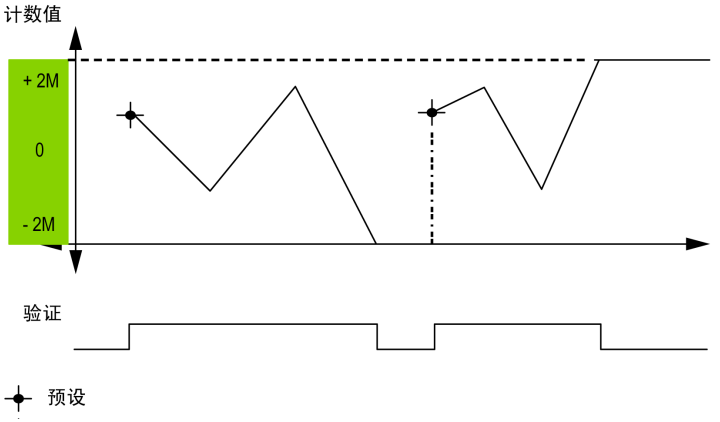
概述

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

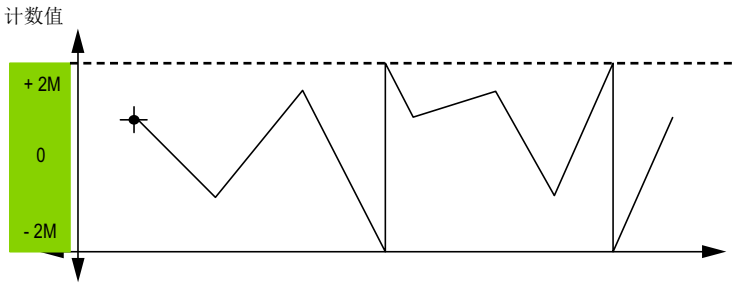
在计数器溢出或下溢的情况下：当前计数器值保持为限制值，Validity 位变为 0，Error 位指示此检测到的错误，直到再次预设计数器。



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 （编码器为Overflow_Flag）输出设置为 1。



第11章
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？

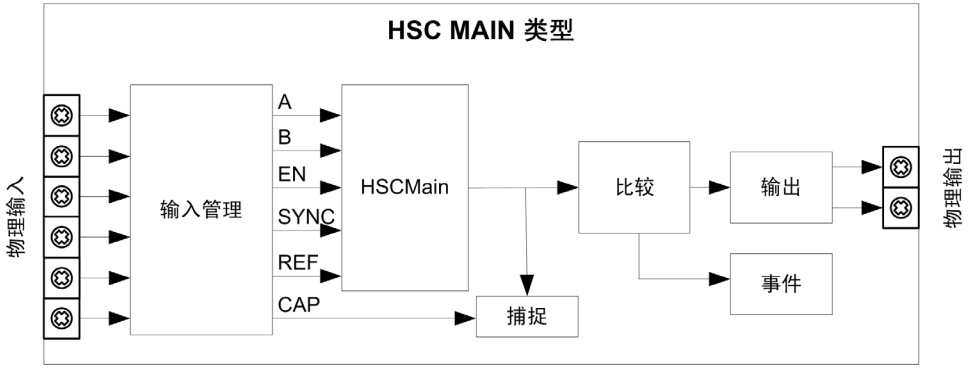
本章包含了以下主题：

主题	页
纲要图	76
自由大型模式下 Main 类型的配置	77
Main 类型编程	78
调整参数	81

纲要图

纲要图

下图简要介绍**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 *135* 页)
- 捕捉功能 (参见第 *143* 页)
- 预设功能 (参见第 *152* 页)
- 启用功能 (参见第 *156* 页)

自由大型模式下 Main 类型的配置

配置过程

按照以下过程配置**自由大型**模式下的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**自由大型**。
4	选择**常规 → 输入模式**参数的值。
5	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。 如有必要，选择**计数输入 → B 输入 → 跳动过滤器**参数的值。
6	输入**范围 → 预设**参数的值。 **限制**定义计数器在达到限制时的行为。
7	或者，可以选择启用以下功能： ● 捕捉功能 (参见第 *143* 页) ● 比较功能 (参见第 *135* 页) ● 启用功能 (参见第 *156* 页) ● 预设功能 (参见第 *152* 页)

注意： **IO 摘要**窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Main 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A, B	0.002 毫秒 （缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

Main 类型编程

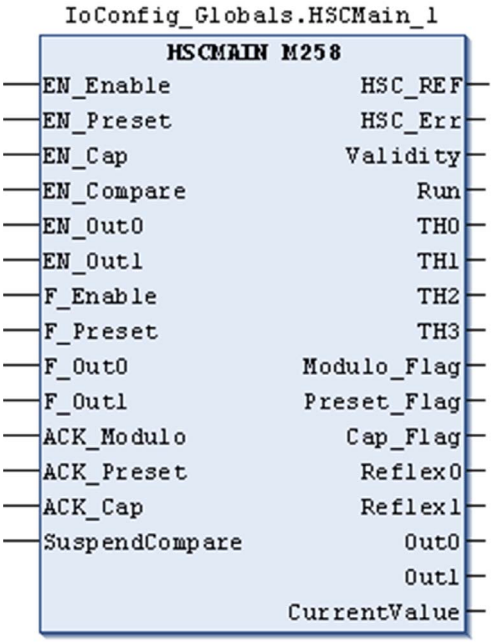
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**，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 逻辑 → 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	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见第 *156* 页) 启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *152* 页) 同步和启动计数器。
EN_Cap	BOOL	当配置 **CAP** 输入时：如果为 TRUE，则启用 Capture 输入 (参见第 *144* 页)。
EN_Compare	BOOL	TRUE = 启用比较器操作 (参见第 *135* 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 Output0 回显 Reflex0 值 (如果已配置)。
EN_Out1	BOOL	TRUE = 启用物理输出 Output1 回显 Reflex1 值 (如果已配置)。
F_Enable	BOOL	强制实施启用条件 (参见第 *156* 页)。优先于 EN_Enable 输入。
F_Preset	BOOL	强制实施预设条件 (参见第 *152* 页)。优先于 EN_Preset 输入。
F_Out0	BOOL	TRUE = 强制 Output0 为 1 (如果 Reflex0 已配置在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中。优先于 EN_Out0。
F_Out1	BOOL	TRUE = 强制 Output1 为 1 (如果 Reflex1 已配置在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中)。优先于 EN_Out1。
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。 ● 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 SuspendCompare 时，EN_Compare、EN_Out0、EN_Out1、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 *167* 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 *174* 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 *135* 页)。
TH1	BOOL	当 CurrentValue > 阈值 1 时，设为 1 (参见第 *135* 页)。
TH2	BOOL	当 CurrentValue > 阈值 2 时，设为 1 (参见第 *135* 页)。
TH3	BOOL	当 CurrentValue > 阈值 3 时，设为 1 (参见第 *135* 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 *152* 页)时，设为 1
Cap_Flag	BOOL	有新的捕捉值存储到捕捉寄存器 (参见第 *144* 页)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。 (参见第 *136* 页)
Reflex1	BOOL	Reflex1 的状态。 (参见第 *136* 页)
Out0	BOOL	物理输出 Output0 的状态（如果已在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中配置 Reflex0，否则在未配置时设置为 FALSE）。
Out1	BOOL	物理输出 Output1 的状态（如果已在 HSC 内嵌功能 *(参见 Modicon M238 Logic Controller, 高速计数, M238 HSC 库指南)*中配置 Reflex1，否则在未配置时设置为 FALSE）。
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 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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_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

第V部分
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
12	**事件计数**原理	85
13	使用 **Main** 类型的**事件计数**	87

第12章
事件计数原理

事件计数模式原理介绍

概述

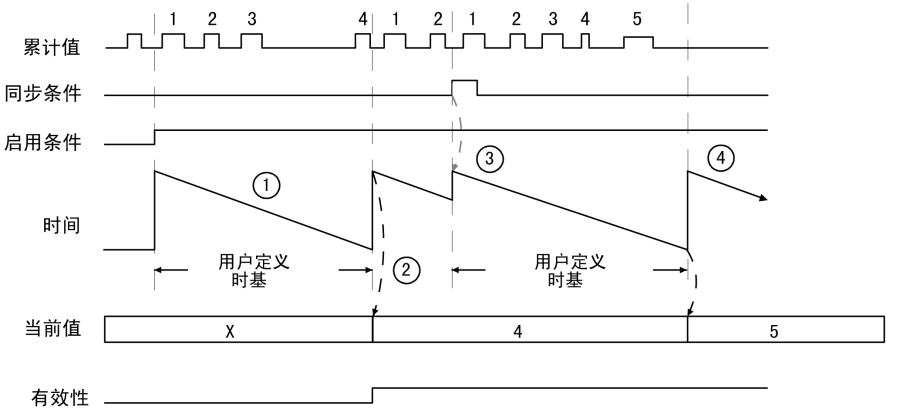
事件计数模式用于计算给定时间段内一系列事件的数量。

原理

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 *152* 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。

阶段	操作
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 CurrentValue 保持为前一个有效值。
- 设置为 1：累计值复位为 0，CurrentValue 保持不变，计数为新时间段重新启动。

第13章
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？

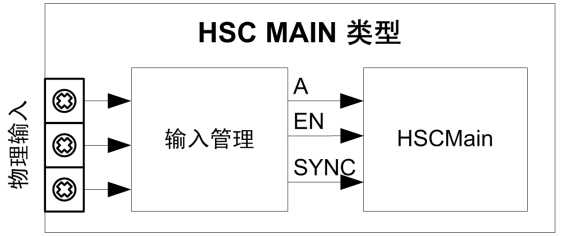
本章包含了以下主题：

主题	页
纲要图	88
事件计数模式下 **Main** 类型的配置	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。

EN 是计数器的启用输入。

SYNC 是计数器的同步输入。

可选功能

除了**事件计数**模式，**Main** 类型还可以提供以下功能：

- 预设功能 (参见第 *152* 页)
- 启用功能 (参见第 *156* 页)

事件计数模式下 Main 类型的配置

配置过程

按照以下过程来配置**事件计数**模式下的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**事件**。
4	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
5	从**范围 → 时基**设置时基值。
6	或者，从下拉菜单中选择 **SYNC** 辅助输入的值，以使用物理输入启用预设功能 (参见第 *152* 页)。

注意： IO 摘要窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Main 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒（缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

Main 类型编程

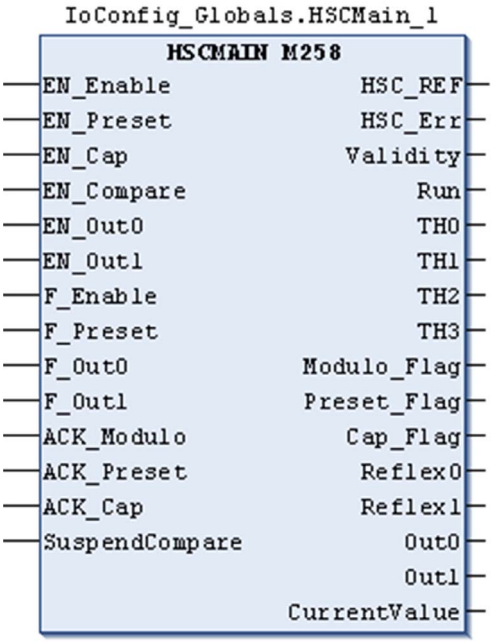
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**，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 逻辑 → 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	当配置 **EN** 输入时：如果为 TRUE，则准许通过 Enable 输入 (参见第 *156* 页)启用计数器。
EN_Preset	BOOL	当配置 **SYNC** 输入时：如果为 TRUE，则准许通过 Sync 输入 (参见第 *152* 页)同步和启动计数器。
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	强制实施启用条件 (参见第 *156* 页)。优先于 EN_Enable 输入。
F_Preset	BOOL	强制实施预设条件 (参见第 *152* 页)。优先于 EN_Preset 输入。
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_Reflex0、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 *167* 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。 EXPERTGetDiag（参见第 *174* 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

输出	类型	注释
Modulo_Flag	BOOL	不相关
Preset_Flag	BOOL	在计数器预设 (参见第 *152* 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 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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	类型	描述
EXPERT_TIMEBASE	EXPERT_TIMEBASE_TYPE 如需详细资讯，请参阅适用于 HSC 的类型 (参见第 *168* 页)。	获取或设置 HSC 的时基值。

第VI部分
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
14	频率计原理	97
15	频率计类型	99

第14章
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）每 10 毫秒更新一次。

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%。

第15章
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？

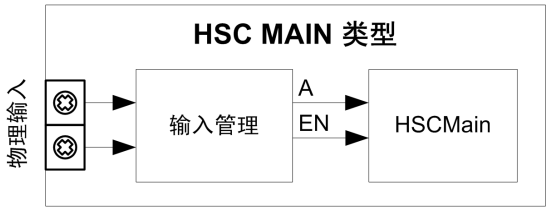
本章包含了以下主题：

主题	页
纲要图	100
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 *156* 页)

配置

配置过程

按照以下过程来配置**频率计**模式下的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**频率计**。
4	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
5	从**范围 → 时基**设置时基值。
6	或者，从下拉菜单中选择 **EN** 辅助输入的值，以使用物理输入启用启用功能 (参见第 *156* 页)。

注意： IO 摘要窗口显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

频率计类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒（缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

编程

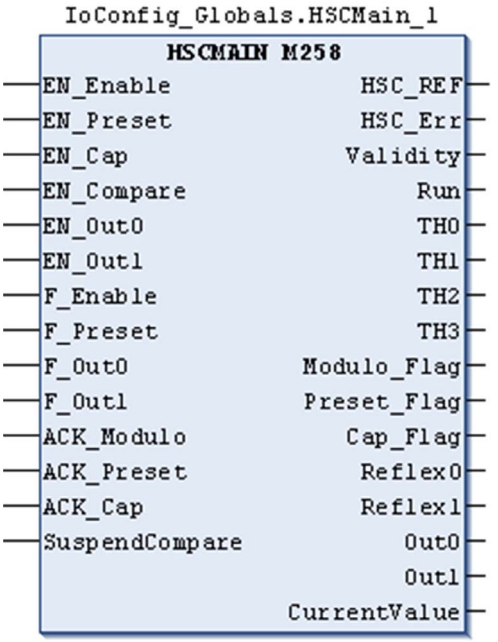
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**， 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 逻辑 → 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 *156* 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	强制实施启用条件 (参见第 *156* 页)。
F_Preset	BOOL	强制实施预设条件 (参见第 *152* 页)
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_Reflex0、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 *167* 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 *174* 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

输出	类型	注释
Preset_Flag	BOOL	在计数器预设 (参见第 *152* 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。

第VII部分
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
16	周期计类型原理	107
17	**周期计**类型	109

第16章
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

可以通过两种方式使用**周期计**：

- 沿对反向沿：允许测量事件的持续时间。
- 沿对沿：允许测量两个事件之间的时间间隔。

可在配置屏幕中指定超时值。如果超过了超时值，则停止测量。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（0.1 µs、1 µs、100 µs、1000 µs）定义的单位表示。

例如，如果当前值 = 100，则**精度**参数为：
0.0001 (0.1 µs) 测量 = 0.01 ms
0.001 (1 µs) 测量 = 0.1 ms
0.1 (100 µs) 测量 = 10 ms
1 (1000 µs) 测量 = 100 ms

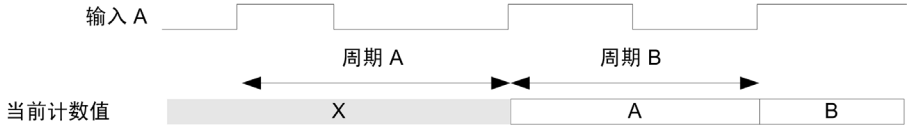
沿对反向沿模式

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。



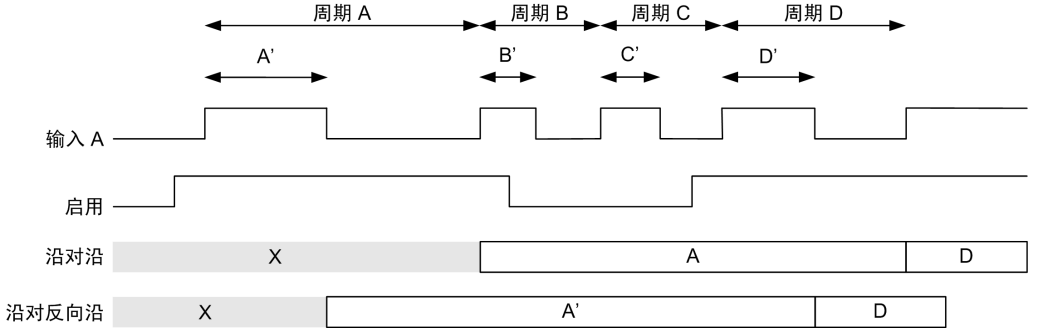
沿对沿模式

启用条件 = 1 时，在 A 输入的两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

下面的趋势图显示了在启用条件中断时计数寄存器的行为：



操作限制

此模块每 5 毫秒内最多执行一次测量。

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。

第17章
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？

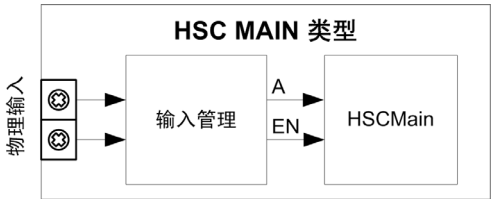
本章包含了以下主题：

主题	页
纲要图	110
配置	111
编程	112
调整参数	115

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 *156* 页)

配置

配置过程

按照以下过程来配置**周期计**模式中的 **Main** 类型：

步骤	操作
1	双击**专用 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	将**常规 → 计数模式**参数的值设置为**周期计**。
4	从**常规 → 周期计模式**中设置周期计模式。
5	选择**计数输入 → A 输入 → 跳动过滤器**参数的值。
6	或者，从下拉列表中选择 **EN** 辅助输入的值，以使用物理输入启用启用功能 (参见第 *156* 页)。
7	从**范围 → 精度**中设置精度值。
8	从**范围 → 超时**中设置超时值。

注意： IO 摘要窗口会显示 **DM72F•** I/O 映射。可以查看专用功能使用的 I/O。

可编程过滤器

Main 类型输入的过滤值确定计数器最大频率，如下表所示：

输入	过滤器值	最大计数器频率
A	0.002 毫秒（缺省值）	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

编程

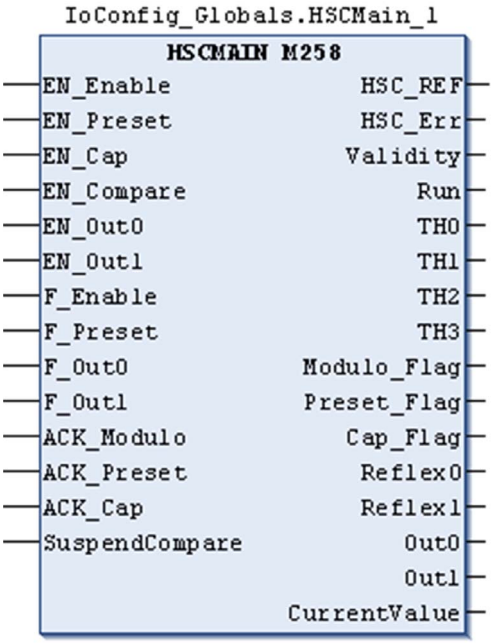
概述

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 → M258 → M258 专用 IO → HSC → HSCMain_M258**，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 逻辑 → 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 *156* 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	强制实施启用条件 (参见第 *156* 页)。
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_Reflex0、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 *167* 页）	HSC 的参考。 与EXPERT_REF_IN管理**功能块的** 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 *174* 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

输出	类型	注释
Modulo_Flag	BOOL	不相关
Preset_Flag	BOOL	不相关
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 *176* 页）或 EXPERTSetParam（参见第 *178* 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 M258 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 *166* 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。
EXPERT_PERIODMETER_ RESOLUTION_TYPE	动态读取或修改时基。 如需详细信息，请参阅适用于周期计的类型 *(参见 Modicon M241 Logic Controller, 高速计数, HSC 库指南)*。

第VIII部分
编码器

第18章
使用编码器的递增模式

概述

本章介绍如何在**递增**模式中使用**编码器**。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
递增模式原理介绍	120
纲要图	124
专用 I/O 模块上的**标准编码器**配置	125
标准编码器编程	128
调整参数	131

递增模式原理介绍

简介

使用**递增**模式连接递增编码器。

原理

递增模式在行为上类似于标准加减计数器。

如果启用条件 (参见第 *156* 页)为 FALSE，则计数器会忽略应用到计数输入 A/B 的脉冲。

在**递增**模式中，计数器必须至少复位一次，才能正常运行。当前计数器值在每一次预设条件 (参见第 *152* 页)发生时与预设值一起加载。

通过配置捕捉条件 (参见第 *147* 页)，可以将当前的计数器存储在捕捉寄存器中。

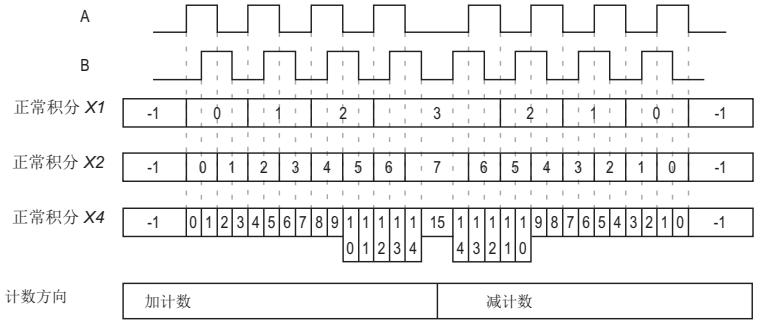
轴类型

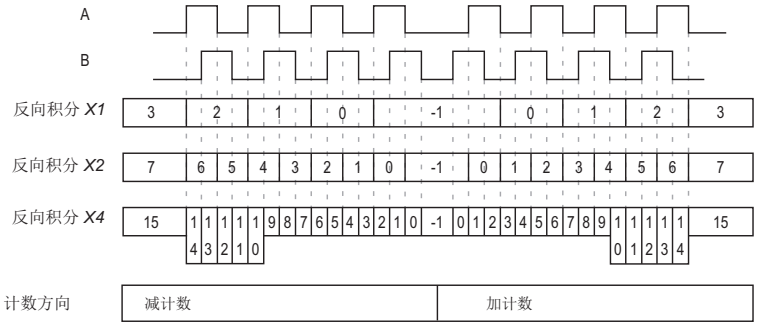
下表介绍可用的 2 种轴类型：

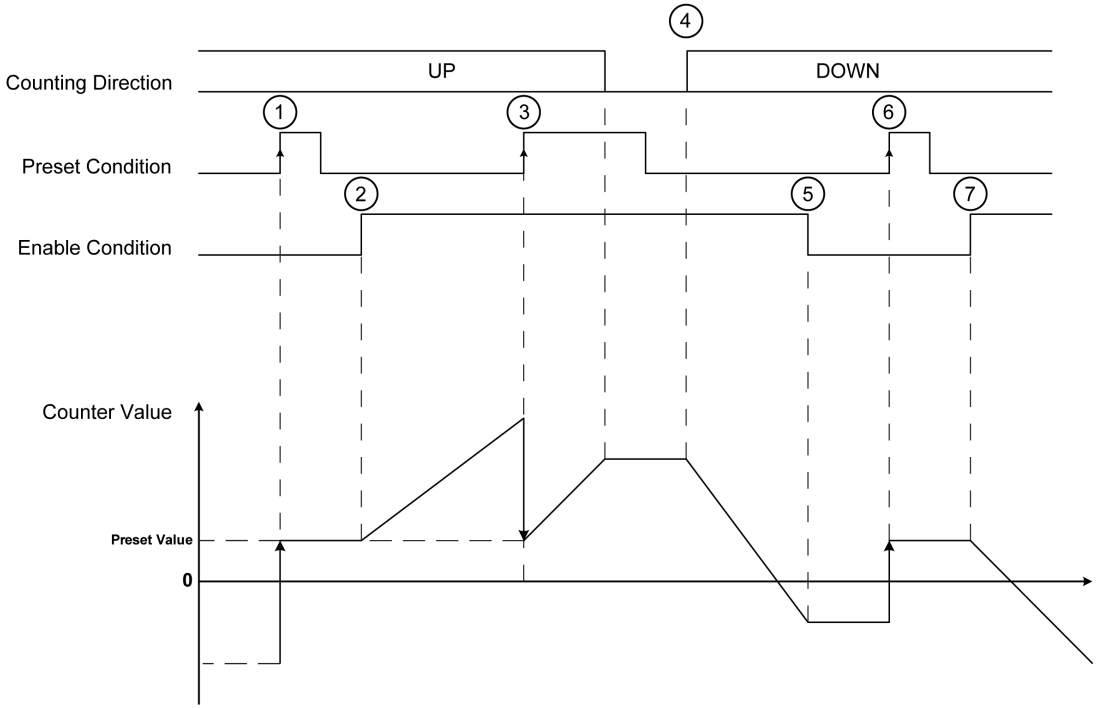
轴类型	注释
线性	该模式可以作为有限计数器使用。
旋转	该模式可以作为无限计数器使用。

原理图

递增模式中的输入模式通常是积分模式：







阶段	操作
1	在预设条件的上升沿上，当前值设为预设值，且计数器已激活。
2	当启用条件 = 1，计数方向为加时，计数器启动并递增。
3	预设条件上的上升沿加载**预设**值。
4	当启用条件 = 1，计数方向为减时，计数器启动并递减。
5	当启用条件 = 0 时，计数器忽略应用到计数输入 A/B 的脉冲。
6	预设条件上的上升沿加载预设值。
7	当启用条件 = 1，计数方向为减时，计数器启动并递减。

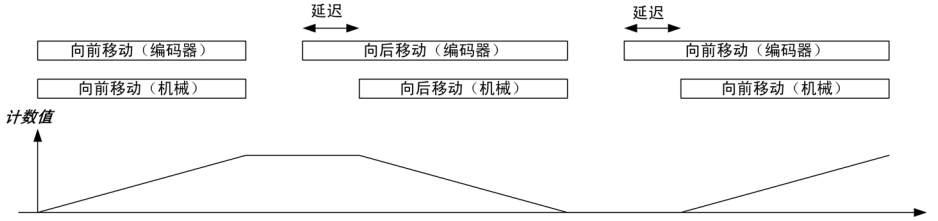
注意： 启用和预设条件取决于配置。这些在启用 (参见第 *156* 页)和预设 (参见第 *154* 页)功能中另有介绍。

延迟

如果旋转被反转，则计数器会应用一个滞后值。延迟值定义在旋转反转期间计数器没有确认的点数。

这样做的目的是考虑编码器/电机轴和机械轴（例如测量垫位置的编码器）之间的延迟。

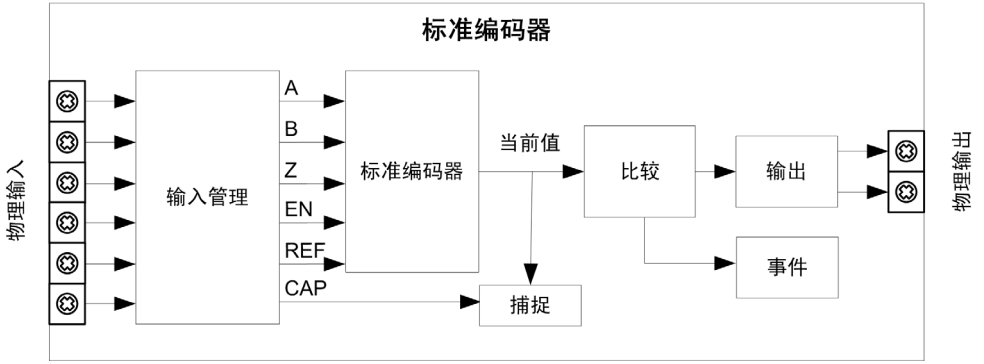
下图对此行为进行了描述：



纲要图

纲要图

下图简要介绍**递增**模式下的**标准编码器**：



A 和 B 是编码器的计数输入。

EN 是编码器的启用输入。

Z 和 REF 是编码器的参考输入。

CAP 是编码器的捕捉输入。

可选功能

除了**递增**模式，**标准编码器**还可以提供以下功能：

- 比较 (参见第 *135* 页)
- 捕捉 (参见第 *147* 页)
- 使用物理输入启用 (参见第 *156* 页)
- 预设物理输入 (参见第 *152* 页)

专用 I/O 模块上的标准编码器配置

配置窗口

按照以下过程来配置**递增**模式下的**标准编码器**类型：

步骤	操作
1	在**设备树**中，双击 **MyController → 专用 → DM72Fx → Standard_Encoder**。 **结果**：配置窗口随即显示。
2	选择**标准编码器配置**选项卡。

I/O 配置参数

下表描述了**递增**模式下的**标准编码器配置**选项卡的属性：

参数			值	缺省值	描述
常规	**输入模式**		正常积分 X1 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	正常积分 X1	选择周期测量间隔。
计数输入	**A 输入**	**跳动过滤器**	0.002 0.004 0.012 0.04 0.12 0.4 1.2 4	0.002	设置用来减小对 A 输入的跳动影响的过滤值（以毫秒为单位）。
范围	**预设**		0	0	设置计数初始值。
	限制		锁定限制 模数	锁定限制	限制管理。
	延迟修正		0	0	指定延迟修正值。

参数			值	缺省值	描述
控制输入	**Z 输入**	**位置**	已禁用 I1.2	已禁用	选择用于预设计数功能的 Z 输入。
		跳动过滤器	0.002 0.004 0.012 0.04 0.12 0.4 1.2 4	0.002	设置用来减小对 Z 输入的跳动影响的过滤值（以毫秒为单位）。
	EN 输入	**位置**	已禁用 I1.4	已禁用	选择用于启用计数功能的 PLC 输入。
	REF 输入	**位置**	已禁用 I1.5	已禁用	选择用于预设计数功能的 REF 输入。
	沿检测		无 Z 上升沿 Z 下降 Z 上升和下降	无	选择使用 SYNC 和 REF 输入预设计数功能的条件。
捕捉	**CAP 输入**	**位置**	已禁用 I1.3	已禁用	选择用于捕捉当前计数值的 PLC 输入。
	CAP0 模式		无 Z 上升沿	无	选择对计数当前值执行第一个捕捉的条件。
	CAP1 模式		无 Z 上升沿	无	选择对计数当前值执行第二个捕捉的条件。
比较	**阈值**	**阈值数**	0 1 2 3 4	0	选择阈值数 (0...4)。
比例缩放	**轴类型**		线性 旋转	线性	选择轴类型。
	增量		2048	2048	每转增量
	单位		360	360	每转单位

⚠ 警告

编码器值不准确

遵守模数计算规则（(模数 x 增量)/单位 = 整数）以避免出错。

不遵循上述说明可能导致人员伤亡或设备损坏。

可编程过滤器

标准编码器输入的过滤值对计数器最大频率的确定如下表所示：

输入	过滤器值	最大计数器频率
A，B	0.002 毫秒	200 kHz
	0.004 毫秒	100 kHz
	0.012 毫秒	40 kHz
	0.04 毫秒	10 kHz
	0.12 毫秒	4 kHz
	0.4 毫秒	1 kHz
	1.2 毫秒	400 Hz
	4 毫秒	100 Hz

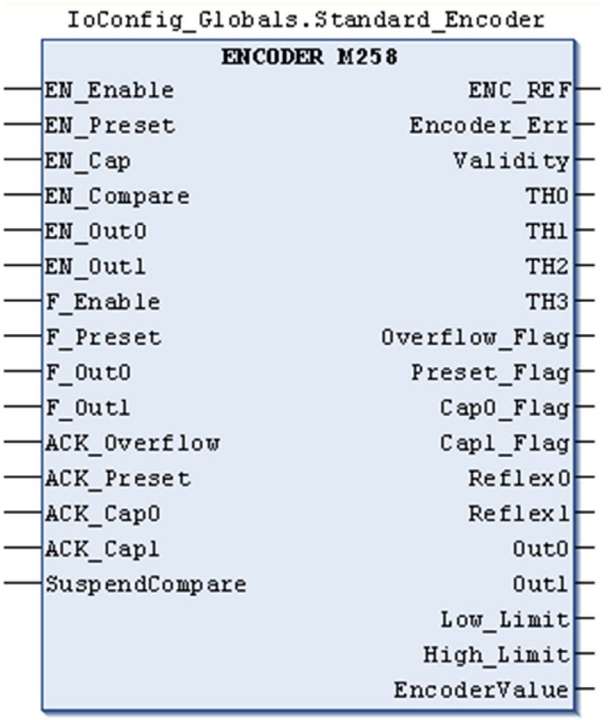
标准编码器编程

概述

标准编码器始终由 Encoder_M258 (参见第 *180* 页) 功能块进行管理。

添加标准编码器功能块

步骤	操作
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 → M258 → M258 专用 IO → 编码器 → ENCODER_M258**，然后将该项目拖放到 **POU** 窗口上。
2	键入 Encoder_M258 实例名称或通过单击以下内容来选择功能块实例： ??? ... 使用输入助手，可以在以下路径选择 Encoder_M258 实例：**全局变量 → PLC → IoConfig_Globals**。



I/O 变量用法

下表介绍了输入变量：

输入	类型	注释
EN_Enable	BOOL	配置 **EN** 输入后，准许编码器通过输入启用 (参见第 *156* 页)。
EN_Preset	BOOL	配置 **Z** 或 **REF** 输入后，准许计数器通过输入预设 (参见第 *152* 页)。
EN_Cap	BOOL	配置至少一个 **CAP** 输入后，准许通过输入实现捕捉功能 (参见第 *147* 页)。
EN_Compare	BOOL	TRUE = 使用阈值 0、1、2、3 (参见第 *135* 页) 启用比较器操作： ● 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 = 准许物理输出 Output0 回显 Reflex0 值。
EN_Out1	BOOL	TRUE = 准许物理输出 Output1 回显 Reflex1 值。
F_Enable	BOOL	强制实施启用条件 (参见第 *156* 页)。
F_Preset	BOOL	强制实施预设条件。
F_Out0	BOOL	TRUE = 强制物理输出 Output0 为 1 (如果已配置 Reflex0) (参见第 *136* 页)。
F_Out1	BOOL	TRUE = 强制物理输出 Output1 为 1 (如果已配置 Reflex1) (参见第 *136* 页)。
ACK_Overflow	BOOL	在上升沿上，复位 Overflow_Flag。
ACK_Preset	BOOL	在上升沿上，复位 Preset_Flag (参见第 *152* 页)。
ACK_Cap0	BOOL	在上升沿上，复位 Cap0_Flag (参见第 *147* 页)。
ACK_Cap1	BOOL	在上升沿上，复位 Cap1_Flag (参见第 *147* 页)。
SuspendCompare	BOOL	TRUE = 比较器操作结果被冻结 (参见第 *135* 页)： ● TH0、TH1、TH2、TH3、Reflex0、Reflex1 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。 在设置 SupendCompare 的同时，保持 EN_Compare、EN_Reflex0、EN_Reflex1、F_Out0、F_Out1 运行。

下表介绍输出变量：

输出	类型	注释
ENC_REF	EXPERT_REF （参见第 *167* 页）	标准编码器的参考。 与**管理**功能块的 EXPERT_REF_IN 输入配合使用
Encoder_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag（参见第 *174* 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。 第一次预设之后为 TRUE
TH0	BOOL	当当前值 > 阈值 0 (如果已配置) (参见第 *136* 页)时，设为 1。

输出	类型	注释
TH1	BOOL	当当前值 > 阈值 1（如果已配置）(参见第 *136* 页)时，设为 1。
TH2	BOOL	当当前值 > 阈值 2（如果已配置）(参见第 *136* 页)时，设为 1。
TH3	BOOL	当当前值 > 阈值 3（如果已配置）(参见第 *136* 页)时，设为 1。
Overflow_Flag	BOOL	当编码器回转到其限制时，设为 1。
Preset_Flag	BOOL	在编码器预设 (参见第 *154* 页)之后，设为 1。
Cap0_Flag	BOOL	有新的捕捉值存储到捕捉寄存器 (参见第 *147* 页)后，设为 1。必须先复位此标志才能开始新的捕捉。
Cap1_Flag	BOOL	有新的捕捉值存储到捕捉寄存器 (参见第 *147* 页)后，设为 1。必须先复位此标志才能开始新的捕捉。
Reflex0	BOOL	Reflex0 的状态 (参见第 *136* 页)。
Reflex1	BOOL	Reflex1 的状态 (参见第 *136* 页)。
Out0	BOOL	Output0 的状态 (参见第 *136* 页)。
Out1	BOOL	Output1 的状态 (参见第 *136* 页)。
Low_Limit	BOOL	当编码器超过 -2.147.483.648 时，设为 1。 (参见第 *73* 页)预设编码器后，复位为 0。
High_Limit	BOOL	当编码器超过 +2,147,483,647 时，设为 1。 (参见第 *73* 页)预设编码器后，复位为 0
EncoderValue	DINT	编码器的当前值。

调整参数

概述

下表中介绍的参数列表可使用 EXPERTGetParam（参见第 *176* 页）或 EXPERTSetParam（参见第 *178* 页）功能块进行读取或修改。

可调整参数

下表列出的是来自 EXPERT_PARAMETER_TYPE（参见第 *166* 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_PRESET	获取或设置编码器的预设值
EXPERT_THRESHOLD0	获取或设置编码器的阈值 0 值
EXPERT_THRESHOLD1	获取或设置编码器的阈值 1 值
EXPERT_THRESHOLD2	获取或设置编码器的阈值 2 值
EXPERT_THRESHOLD3	获取或设置编码器的阈值 3 值
EXPERT_OFFSET	获取或设置**旋转**轴模式下的编码器的偏移
EXPERT_MODULO	获取或设置**旋转**轴模式下的编码器模数
EXPERT_SLACK	获取或设置编码器延迟
EXPERT_SCALING	获取或设置编码器比例调整参数 比例调整参数（功能块的 ParamValue）有 2 个子参数： **高有效 INT** = 增量 **低有效 INT** = 单位
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

第IX部分
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
19	比较功能	135
20	捕捉功能	143
21	预设和启用功能	151

第19章
比较功能

第19.1节
使用 Main 类型的比较

概述

本节提供有关使用 **Main** 类型或编码器的比较功能的信息。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
使用 **Main** 类型或编码器的比较原理	137
对使用 **Main** 类型或编码器的比较进行配置	140
外部事件配置	141

使用 Main 类型或编码器的比较原理

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)

使用编码器的比较块用于管理阈值、反射输出和事件。

激活至少一个阈值后，在配置屏幕 (参见第 *140* 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 *138* 页)
- 与外部任务相关联的阈值事件 (参见第 *138* 页)
- 反射输出 (参见第 *138* 页)

比较原理

Main 类型或编码器能够管理最多 到 4 个阈值。

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

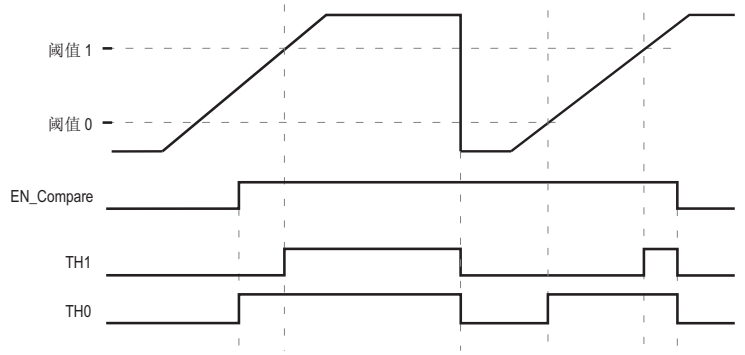
阈值在配置窗口中定义，也可通过使用 EXPERTSetParam (参见第 *178* 页) 功能块在应用程序中进行调整。

如果配置了阈值 x (x= 0、1、2、3) ，并且已启用比较 (EN_Compare = 1)，则 HSCMain_M258 (Encoder_M258) 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_M258 (Encoder_M258) 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。

以下两个阈值的示例显示了 HSCMain_M258（Encoder_M258）功能块中的比较：



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 THO 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。

配置事件触发

配置越过阈值时的事件可触发外部任务 (参见第 *141* 页)。可以选择在越过配置的阈值时触发事件，如下所述：

- **向上交叉**。阈值变成高于阈值时，触发事件。
- **向下交叉**。阈值变成低于阈值时，触发事件。
- **向上和向下交叉**。阈值变成高于阈值时以及阈值变成低于阈值时，触发事件。

反射输出行为

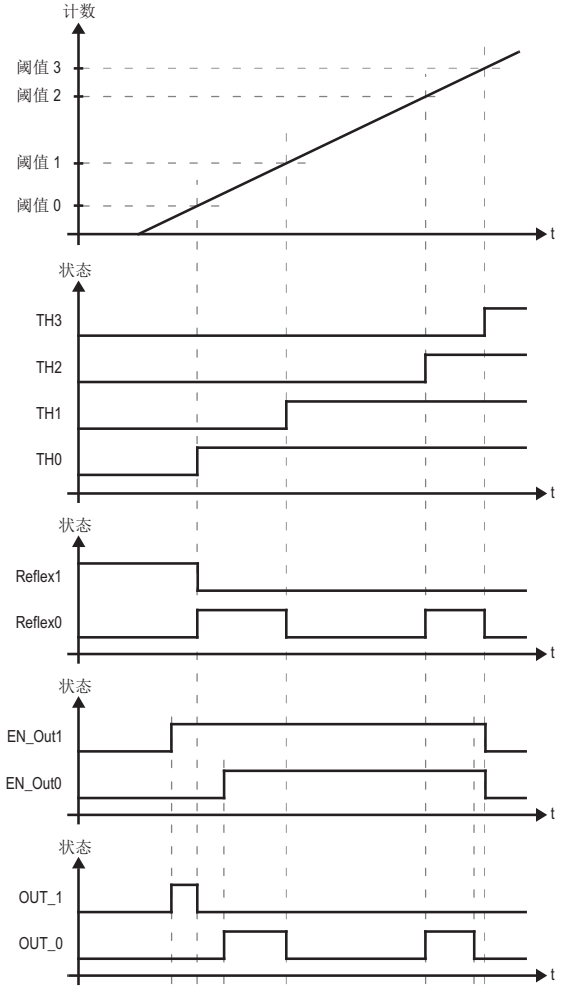
配置反射输出可触发物理反射输出。

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。

高速计数器或编码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。

性能与所使用的输出的类型（快速或常规）直接相关。有关详细信息，请参阅嵌入式专用 I/O 分配。

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。

对使用 Main 类型或编码器的比较进行配置

配置过程

请按照以下步骤对使用 Main 类型或编码器的比较功能进行配置：

步骤	操作
1	在**设备树**中，双击**专用 → DM72F• → HSCMain** 或 **Standard_Encoder** 节点。
2	选择 **HSC Main 配置**或**标准编码器配置**选项卡。
3	在**阈值数**参数中，选择要使用的阈值数。
4	设置每个阈值的值。 **注意：** 根据以下规则配置阈值：TH0 < TH1 < TH2 < TH3
5	也可以选择配置**反射输出**行为 (参见第 *137* 页)。 **结果**：所选组中的外部事件（HSCx_TH0、HSCx_TH1、HSCx_TH2、HSCx_TH3、HSCx_STOP）出现在**阈值 x 外部事件**下方。

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 M258 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
BLOCK0_I0	功能块 **DM72F0** 的输入 I0 设置为 1 时，激活任务。
BLOCK0_I1	功能块 **DM72F0** 的输入 I1 设置为 1 时，激活任务。
BLOCK0_I2	功能块 **DM72F0** 的输入 I2 设置为 1 时，激活任务。
BLOCK0_I3	功能块 **DM72F0** 的输入 I3 设置为 1 时，激活任务。
BLOCK1_I0I4	功能块 **DM72F1** 的输入 I0 设置为 1 时，激活任务。
BLOCK1_I1I5	功能块 **DM72F1** 的输入 I1 设置为 1 时，激活任务。
BLOCK1_I2I6	功能块 **DM72F1** 的输入 I2 设置为 1 时，激活任务。
BLOCK1_I3I7	功能块 **DM72F1** 的输入 I3 设置为 1 时，激活任务。
BLOCK0_TH0	HSC 或功能块 **DM72F0** 的编码器的阈值 TH0 设置为 1 时，激活任务。
BLOCK0_TH1	HSC 或功能块 **DM72F0** 的编码器的阈值 TH1 设置为 1 时，激活任务。
BLOCK0_TH2	HSC 或功能块 **DM72F0** 的编码器的阈值 TH2 设置为 1 时，激活任务。
BLOCK0_TH3	HSC 或功能块 **DM72F0** 的编码器的阈值 TH3 设置为 1 时，激活任务。
BLOCK1_TH0	HSC 或功能块 **DM72F1** 的编码器的阈值 TH0 设置为 1 时，激活任务。
BLOCK1_TH1	HSC 或功能块 **DM72F1** 的编码器的阈值 TH1 设置为 1 时，激活任务。
BLOCK2_TH2	HSC 或功能块 **DM72F2** 的编码器的阈值 TH1 设置为 1 时，激活任务。
BLOCK1_TH3	HSC 或功能块 **DM72F1** 的编码器的阈值 TH3 设置为 1 时，激活任务。
ENCODER_TH0	**编码器**接口的某个编码器的阈值 TH0 设置为 1 时，激活任务。
ENCODER_TH1	**编码器**接口的某个编码器的阈值 TH1 设置为 1 时，激活任务。
ENCODER_TH2	**编码器**接口的某个编码器的阈值 TH2 设置为 1 时，激活任务。
ENCODER_TH3	**编码器**接口的某个编码器的阈值 TH3 设置为 1 时，激活任务。
BLOCK0_HSCSTOP	在**一次性**模式下，与功能块 **DM72F0** 相关的 HSC 的值达到 0 时，激活任务。

事件名称	描述
BLOCK1_HSCSTOP	在**一次性**模式下，与功能块 **DM72F1** 相关的 HSC 的值达到 0 时，激活任务。
CAN0_SYNC	在 CANopen 管理器发送 Sync 消息（在 CANopen 管理器配置窗口中选中**启用同步生成**时启用）时，激活任务。

第20章
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？

本章包含了以下部分：

节	主题	页
20.1	使用 **Main** 类型的捕捉	144
20.2	使用编码器捕捉	147

第20.1节
使用 Main 类型的捕捉

概述

本节提供有关使用 **Main** 类型的捕捉功能的信息。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
使用 **Main** 类型的捕捉原理	145
对使用 **Main** 类型的捕捉进行配置	146

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：

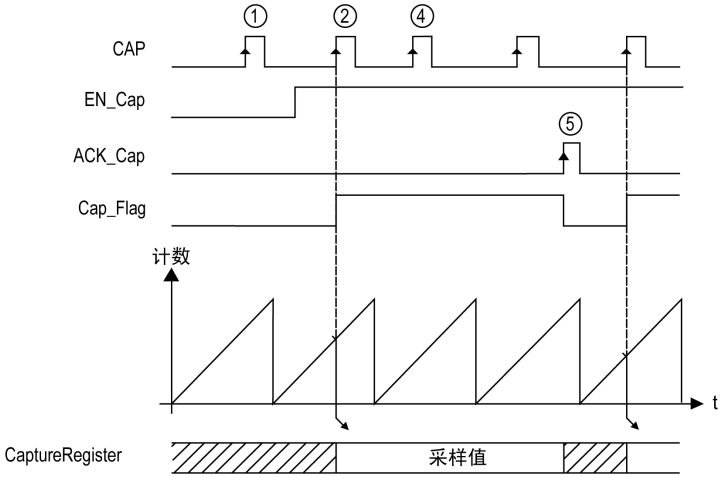
- One-shot (参见第 *39* 页)
- Modulo-loop (参见第 *59* 页)
- Free-large (参见第 *75* 页)

使用此功能需要：

- 配置可选的捕捉输入：**CAP**
- 使用 EXPERTGetCapturedValue （参见第 *172* 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 *172* 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_M258（参见第 *183* 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**专用 → DM72F• → HSCMain** 或 **Standard_Encoder** 节点。
2	选择 **HSC Main 配置**或**标准编码器配置**选项卡。
3	选择**捕捉 → CAP 输入 → 位置**参数的值。
4	选择**捕捉 → CAP 输入 → 跳动过滤器**参数的值。
5	定义**捕捉 → 模式**参数的触发模式。

第20.2节
使用编码器捕捉

概述

本节提供有关**标准编码器**或**运动编码器**的捕捉功能的信息。

本节包含了哪些内容？

本节包含了以下主题：

主题	页
使用编码器捕捉	148
编码器上的捕捉配置	150

使用编码器捕捉

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。

每个编码器有 2 个捕捉寄存器（CAP0 和 CAP1）。这些寄存器可以通过 2 种方式使用：

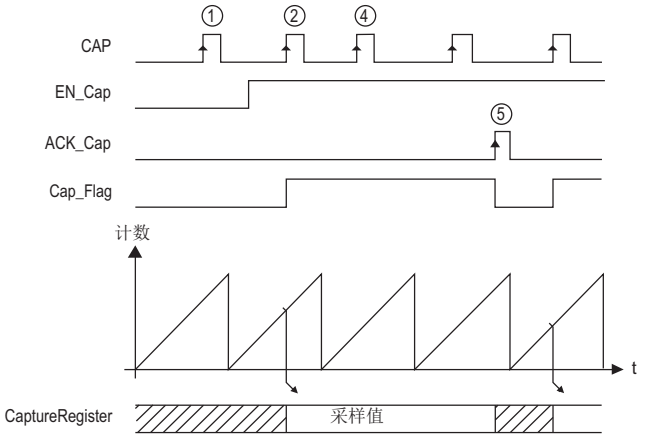
- 最多 2 个位置捕捉
- 1 个距离捕捉

使用此功能，需要：

- 配置可选的捕捉输入：**CAP**
- 使用 EXPERTGetCapturedValue（参见第 *172* 页）功能块来检索在应用程序中捕捉到的值。

捕捉原理

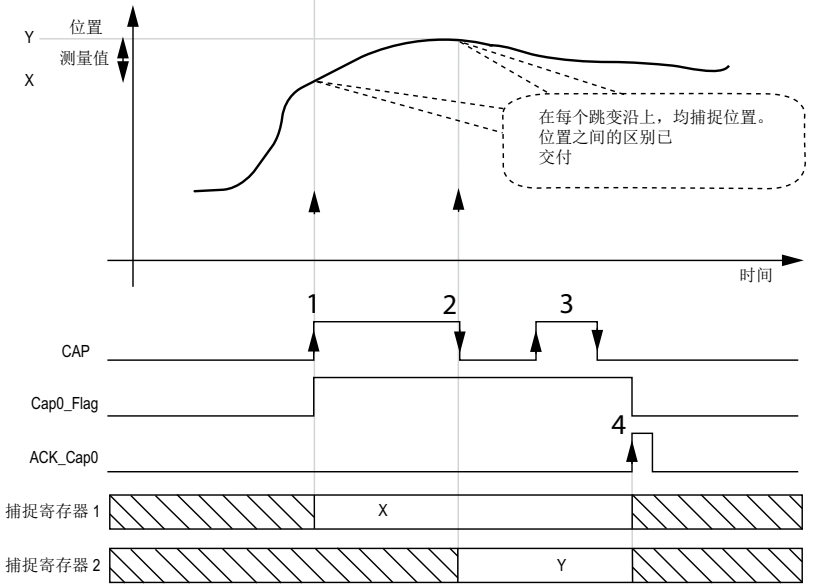
下图说明了位置捕捉的工作方式（仅显示了一个寄存器）：



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 *172* 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	Encoder（参见第 *183* 页）功能块输入 ACK_Cap 的上升沿会触发下降沿 Cap_Flag 输出。 准许新的捕捉。

距离捕捉的原理

使用编码器时，距离捕捉可以获取 CAP 输入每个跳变沿间的差异，如下图所示：



阶段	操作
1	Cap 的上升沿捕捉当前计数器值并将其存入第一个捕捉寄存器。
2	Cap 的下降沿捕捉当前计数器值并将其存入第二个捕捉寄存器，然后触发 Cap0_Flag 的上升沿。
3	使用 EXPERTGetCapturedValue（参见第 *172* 页）获取存储的值。 EXPERTGetCapturedValue（参见第 *172* 页）功能块可以获取： ● 上升沿上的位置 ● 下降沿上的位置 ● 距离值
4	当 Cap0_Flag = 1 时，忽略物理输入 CAP 上任何新的跳变沿。
5	编码器（参见第 *183* 页）功能块输入 ACK_Cap 的上升沿会触发下降沿 Cap_Flag 输出。 准许新的捕捉。

注意： 对于旋转轴，距离始终为正值，即使下降沿上的位置小于上升沿上的位置。

编码器上的捕捉配置

专用模块上的配置过程

按照以下过程配置**编码器**的捕捉功能：

步骤	操作
1	在**设备树**中，双击**专用** → **DM72Fx** → **Standard_Encoder**
2	选择配置选项卡。
3	选择**捕捉** → **CAP 输入** → **位置**参数的值。
4	选择**捕捉** → **CAP 输入** → **跳动过滤器**参数的值。
5	如有必要，选择 **CAP0 模式**和 **CAP1 模式**参数。
6	如有必要，启用**捕捉距离**参数。

第21章
预设和启用功能

概述

本章提供有关 HSC的同步预设和启用功能的信息。

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	152
自由大型或周期计预设条件	154
启用：准许计数操作	156

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：

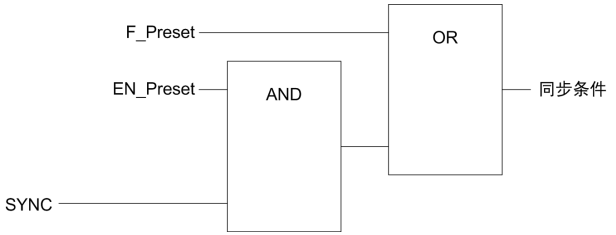
- **一次性**计数器：预设和启动计数器
- **模数回路**计数器：复位和启动计数器
- **事件计数**：在开始时重新启动内部时基

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。

以下任一事件都会触发对同步条件的捕捉：

- F_Preset 输入的上升沿
- SYNC 物理输入的上升沿和/或下降沿（如果已配置 SYNC 输入，并且 EN_Preset 输入为 TRUE）。

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 → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
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**模式下，利用两个物理输入来创建预设条件：

- SYNC （编码器对应的是 Z）
- REF

可用的预设条件有七个：

- 在输入 SYNC 的跳变沿（上升沿、下降沿或上升/下降沿）
- 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、下降沿或上升/下降沿）

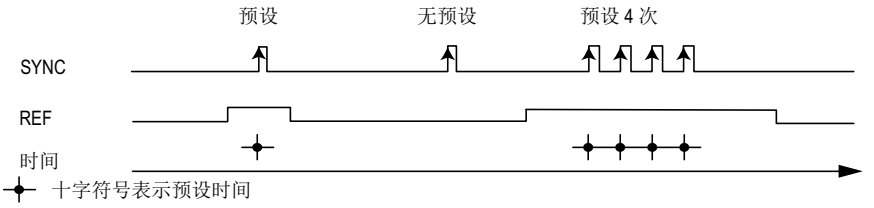
计数器在编码器参考点上进行同步。

在输入 REF 的上升沿

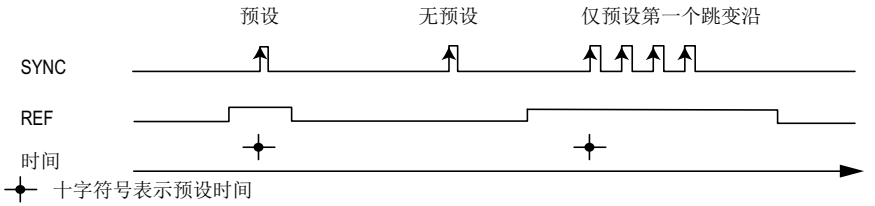
计数器在机械位置进行同步。

如果输入 REF 为活动高信号，则在输入 SYNC 的上升沿

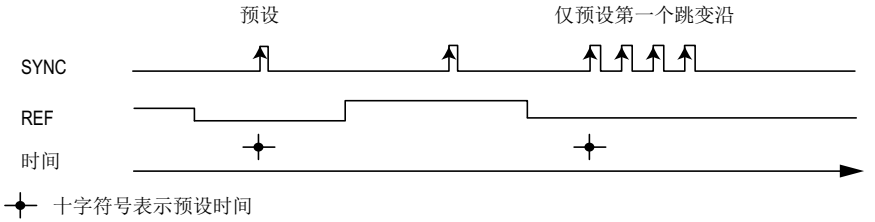
当 **REF** 信号为 TRUE 时，计数器在编码器参考点上进行同步，如下所示：



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：准许计数操作

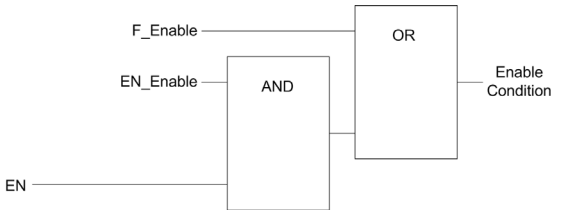
概述

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。

下图介绍了启用条件：



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 相对应。

配置

以下过程介绍如何配置启用功能：

步骤	动作
1	在**设备树**中，双击 **MyController → DM72F• → HSCMain**。
2	选择 **HSC Main 配置**选项卡。
3	选择**控制输入 → EN 输入 → 位置**参数的值。
4	选择**控制输入 → EN 输入 → 跳动过滤器**参数的值。

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	159
B	数据类型	163
C	功能块	169
D	功能和功能块表示形式	189

附录 A
一般信息

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	160
有关管理和运动功能块管理的一般信息	161

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 专用	最大计数器频率 常规
A B	0.000	200 kHz	1 kHz
	0.001	200 kHz	1 kHz
	0.002	200 kHz	1 kHz
	0.005	100 kHz	1 kHz
	0.010	50 kHz	1 kHz
	0.05	25 kHz	1 kHz
	0.1	5 kHz	1 kHz
	0.5	1 kHz	1 kHz
	1	500 Hz	500 Hz
	5	100 Hz	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。			

专用输出

高速专用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。

⚠ 警告

意外的设备操作

- 请勿在不同的程序任务中使用同一个功能块实例。
- 请勿在功能块正在执行时修改或更改功能块参考 (AXIS)。

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
IMMEDIATE_ERR_TYPE：GetImmediateValue 功能块的错误变量类型	164
EXPERT_ERR_TYPE：EXPERT 功能块的错误变量类型	165
EXPERT_PARAMETER_TYPE：在EXPERT功能块上执行获取或设置的参数类型	166
EXPERT_REF：EXPERT 参考值	167
EXPERT_TIMEBASE_TYPE：HSC 时基变量的类型	168

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
IMMEDIATE_NO_ERROR	0	未检测到错误
IMMEDIATE_UNKNOWN	1	IMMEDIATE 功能的参考不正确或未配置
IMMEDIATE_UNKNOWN_PARAMETER	2	参数参考不正确

EXPERT_ERR_TYPE：EXPERT 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 *166* 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_PARAMETER_TYPE：在EXPERT功能块上执行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	获取或设置专用功能的预设（编码器的偏移）值。
EXPERT_MODULO	01 hex	获取或设置专用功能的模数值。
EXPERT_OFFSET	02 hex	获取或设置专用功能的偏移。
EXPERT_TIMEBASE	03 hex	获取或设置专用功能的时基值 (参见第 *168* 页)。
EXPERT_SLACK	04 hex	获取或设置专用功能（仅限编码器）的延迟值。
EXPERT_SCALING	05 hex	获取或设置比例调整参数。 比例调整参数（FB 的 ParamValue）有 2 个子参数： **高有效 INT** = 增量 **低有效 INT** = 单位
EXPERT_THRESHOLD0	06 hex	获取或设置专用功能的阈值 0 值。
EXPERT_THRESHOLD1	07 hex	获取或设置专用功能的阈值 1 值。
EXPERT_THRESHOLD2	08 hex	获取或设置专用功能的阈值 2 值。
EXPERT_THRESHOLD3	09 hex	获取或设置专用功能的阈值 3 值。
EXPERT_REFLEX0	0A（十六进制）	获取或设置专用功能的输出 0 反射模式
EXPERT_REFLEX1	0B（十六进制）	获取或设置专用功能的输出 1 反射模式

EXPERT_REF：EXPERT 参考值

数据类型描述

EXPERT_REF是用于标识与管理功能块相关的 EXPERT 功能的字节。

EXPERT_TIMEBASE_TYPE：HSC 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 功能块的各种时基值：

名称	值
EXPERT_100ms	00 hex
EXPERT_1s	01 hex
EXPERT_10s	02 hex
EXPERT_60s	03 hex

附录 C
功能块

概述

本章介绍 HSC 的功能和功能块以及专用 I/O 库的编码器部分。

本章包含了哪些内容？

本章包含了以下主题：

主题	页
EXPERTGetImmediateValue：读取 HSC 或编码器功能的计数器值	170
EXPERTGetCapturedValue:返回捕捉寄存器的内容	172
EXPERTGetDiag：提供主 EXPERT I/O 功能相关检出错误的详细信息	174
EXPERTGetParam：返回主 EXPERT I/O 功能的参数	176
EXPERTSetParam：调节 HSC 的参数	178
Encoder_M258：编码器功能块	180
HSCMain: HSC Main 功能块	183
HSCSimple_M258：控制 M258 的 Simple 类型计数器	187

EXPERTGetImmediateValue：读取 HSC 或编码器功能的计数器值

功能描述

此管理功能允许绕过控制器循环读取 HSC 或编码器的计数器值。

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 *189* 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 *167* 页）	专用功能块的参考。
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 *167* 页）	专用功能块的参考。
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 *164* 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 → M258 → M258 专用 IO → 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。

EXPERTGetCapturedValue:返回捕捉寄存器的内容

功能块描述

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 *189* 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 *167* 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 ● 对于编码器：0：Cap0，1：Cap1，或 2：距离

下表介绍输出变量：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF （参见第 *167* 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。
Error	BOOL	TRUE = 表示检测到一个错误。 功能块执行结束。

输出	类型	注释
ErrID	EXPERT_ERR_TYPE （参见第 *165* 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 *161* 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 → M258 → M258 专用 IO → 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：提供主 EXPERT I/O 功能相关检出错误的详细信息

功能块描述

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 *189* 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 *167* 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 *167* 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。
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 *165* 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 *161* 页)。

下表指示诊断值：

位	HSC	标准编码器
0	在物理输入上检测到的错误	
1	–	在物理输出上检测到的错误
2	–	–
3	–	–
4	–	编码器配电反馈
5(1)	–	在绝对 SSI 编码器帧的传输上检测到错误
6(1)	–	表示检测到绝对 SSI 编码器帧的校验错误
7	检测到无效配置	
8	检测到无效调整参数	
9	–	编码器配置正在进行
10	–	–
11(1)	–	绝对 SSI 编码器状态位 0。请参阅您的编码器用户指南。
12(1)	–	绝对 SSI 编码器状态位 1。请参阅您的编码器用户指南。
13(1)	–	绝对 SSI 编码器状态位 2。请参阅您的编码器用户指南。
14(1)	–	绝对 SSI 编码器状态位 3。请参阅您的编码器用户指南。
15(1)	–	–
(1) 检测到 SSI 错误时，将启用条件 (参见第 *156* 页)设为 0 以确认错误条件。		

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 → M258 → M258 专用 IO → 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。

EXPERTGetParam：返回主 EXPERT I/O 功能的参数

功能块描述

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 *189* 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 *167* 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 *166* 页）	要读取的参数。

下表介绍输出变量：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF （参见第 *167* 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。
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 *165* 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 *161* 页)。

添加 EXPERTGetParam 功能块

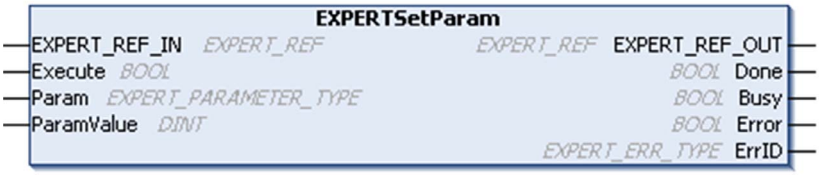
步骤	描述
1	在**软件目录**中选择**库**选项卡，然后单击**库**。 在列表中选择**控制器 → M258 → M258 专用 IO → 管理 → EXPERTGetParam**，然后将该项目拖放至 **POU** 窗口中。
2	将 **EXPERT_REF_IN** 输入链接到 HSC 的 **HSC_REF** 输出。

EXPERTSetParam：调节 HSC 的参数

功能块描述

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 *189* 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 *167* 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 *166* 页）	要读取的参数。
ParamValue	DINT	要写入的参数值。

下表介绍输出变量：

输出	类型	注释
EXPERT_REF_OUT	EXPERT_REF （参见第 *167* 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。
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 *165* 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 *161* 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 → M258 → M258 专用 IO → 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。

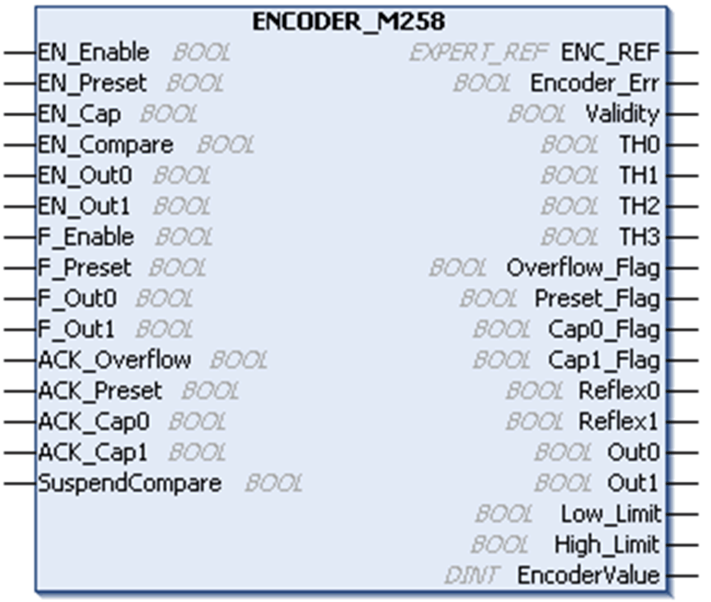
Encoder_M258：编码器功能块

功能说明

此功能块控制编码器类型计数器。

功能块实例名称必须与配置定义的名称相匹配。

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 *189* 页)。

I/O 变量描述

下表介绍输入变量：

输入	类型	注释
EN_Enable	BOOL	TRUE = 准许通过 Enable 输入（如果已配置）启用编码器。
EN_Preset	BOOL	TRUE = 准许通过 Sync 输入（如果已配置）进行编码器同步和启动。
EN_Cap	BOOL	TRUE = 启用 Capture 输入（如果已配置）。
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 来回显 Reflex0 值（如果已在 DM72F 模块上配置）。
EN_Out1	BOOL	TRUE = 启用 Output1 来回显 Reflex1 值（如果已在 DM72F 模块上配置）。
F_Enable	BOOL	强制实施启用条件 (参见第 *156* 页)。
F_Preset	BOOL	强制实施预设条件。
F_Out0	BOOL	TRUE = 强制实施 Output0 为 1（如果已配置了 Reflex0）。
F_Out1	BOOL	TRUE = 强制实施 Output1 为 1（如果已配置了 Reflex1）。
ACK_Overflow	BOOL	在上升沿上，复位 Overflow_Flag。
ACK_Preset	BOOL	在上升沿上，复位 Preset_Flag。
ACK_Cap0	BOOL	在上升沿上，复位 Cap0_Flag。
ACK_Cap1	BOOL	在上升沿上，复位 Cap1_Flag。
SuspendCompare	BOOL	TRUE = 比较器操作结果被冻结： ● TH0、TH1、TH2、TH3、Reflex0、Reflex1 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。 ● 事件被隐藏. 在设置 SupendCompare 的同时，保持 EN_Compare、EN_Reflex0、EN_Reflex1、F_Out0、F_Out1 运行。

下表描述了输出变量：

输出	类型	注释
ENC_REF	EXPERT_REF（参见第 *167* 页）	编码器的参考。
Encoder_Err	BOOL	TRUE = 表示检测到一个错误。 使用 EXPERTGetDiag（参见第 *174* 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。
TH0	BOOL	当 CurrentValue > 阈值 0（如果已配置）时，设为 1。 只有在设置了 EN_Compare 后才处于活动状态。
TH1	BOOL	当 CurrentValue > 阈值 1（如果已配置）时，设为 1。 只有在设置了 EN_Compare 后才处于活动状态。
TH2	BOOL	当 CurrentValue > 阈值 2（如果已配置）时，设为 1。 只有在设置了 EN_Compare 后才处于活动状态。
TH3	BOOL	当 CurrentValue > 阈值 3（如果已配置）时，设为 1。 只有在设置了 EN_Compare 后才处于活动状态。
Overflow_Flag	BOOL	当编码器回转到其限制时，设为 1。
Preset_Flag	BOOL	在编码器预设 (参见第 *154* 页)之后，设为 1。
Cap0_Flag	BOOL	有新的捕捉值存储到 Capture 寄存器后，设置为 1。 必须先复位此标志才能开始新的捕捉。
Cap1_Flag	BOOL	有新的捕捉值存储到 Capture 寄存器后，设置为 1。 必须先复位此标志才能开始新的捕捉。
Reflex0	BOOL	Reflex0 的状态（如果已配置）。 只有在设置了 EN_Compare 后才处于活动状态。
Reflex1	BOOL	Reflex1 的状态（如果已配置）。 只有在设置了 EN_Compare 后才处于活动状态。
Out0	BOOL	指示 Output0 的状态。
Out1	BOOL	指示 Output1 的状态。
Low_Limit	BOOL	仅用于管理锁定限制的递增线性。编码器超过 -2.147.483.648时，设置为 1，编码器预设或复位时，复位为 0。
High_Limit	BOOL	仅用于管理锁定限制的递增线性。编码器超过 +2.147.483.648时，设置为 1，编码器预设或复位时，复位为 0。
EncoderValue	DINT	编码器的当前值。

HSCMain: HSC Main 功能块

功能块描述

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
- 周期计
- 双相

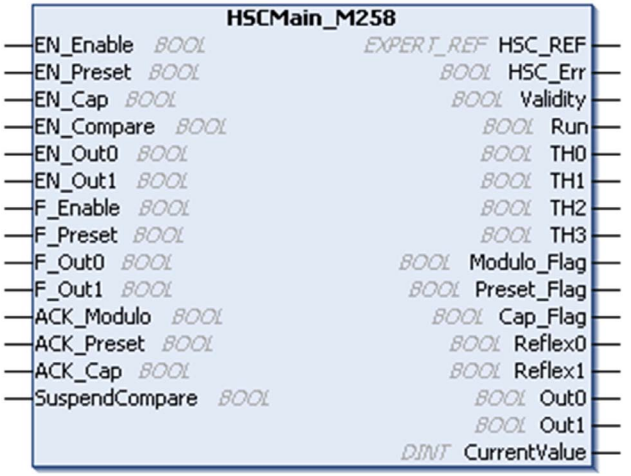
使用 **Main** 计数器时，必须使用 HSC Main 功能块。

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 *189* 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。

输入	类型	描述
F_Preset	BOOL	在上升沿上，准许进行下列计数模式中的计数功能同步和启动： **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。

下表介绍输出变量：

输出	类型	注释
HSC_REF	EXPERT_REF （参见第 *167* 页）	HSC 的参考。
Error	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag（参见第 *174* 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。 在**周期计**类型下，如果超过了超时值，则 Validity = FALSE。
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。

输出	类型	注释
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：当计数器转过其限制时
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：重新启动相对于时基的内部定时器时。 ● **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。
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_M258：控制 M258 的 Simple 类型计数器

功能块描述

此功能块通过下列简化的功能控制 **Simple** 类型计数器：

- 单通道计数
- 无阈值
- 无事件
- 无捕捉
- 无反射

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 *189* 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 *167* 页）	HSC 的参考。
HSC_Err	BOOL	TRUE = 表示检测到错误。 使用 EXPERTGetDiag（参见第 *174* 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，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	190
如何通过 IL 语言使用功能或功能块	191
如何通过 ST 语言使用功能或功能块	195

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。

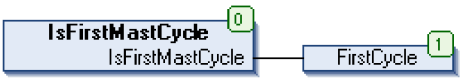
我们以功能 IsFirstMastCycle、功能 SetRTCDrift 和功能块 TON 为例来演示实现的过程。

通过 IL 语言使用功能

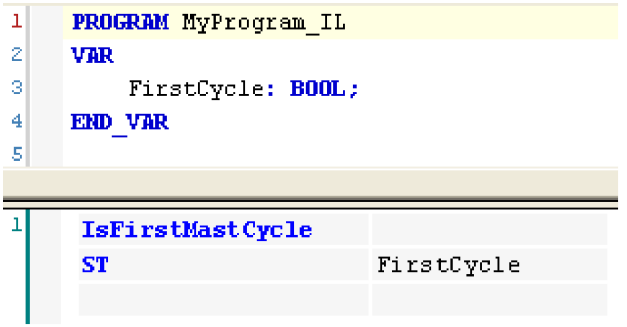
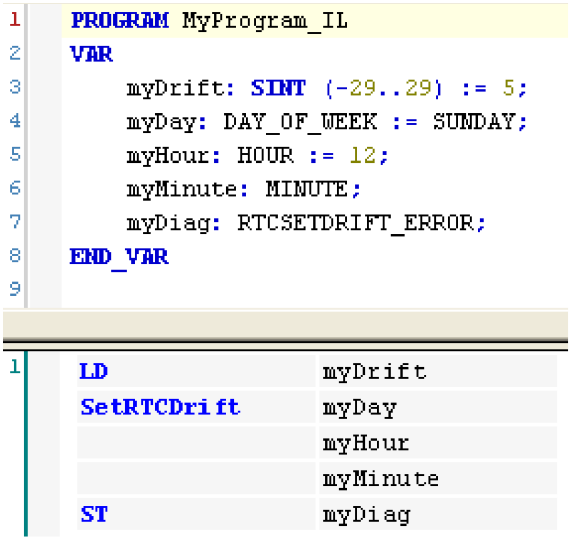
以下过程描述如何用 IL 语言插入一个功能：

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	插入新的行，将功能的结果存储到相应的变量中：在操作符列（左侧字段）中输入 ST 指令， 并在右侧的字段中输入变量名称。

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	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

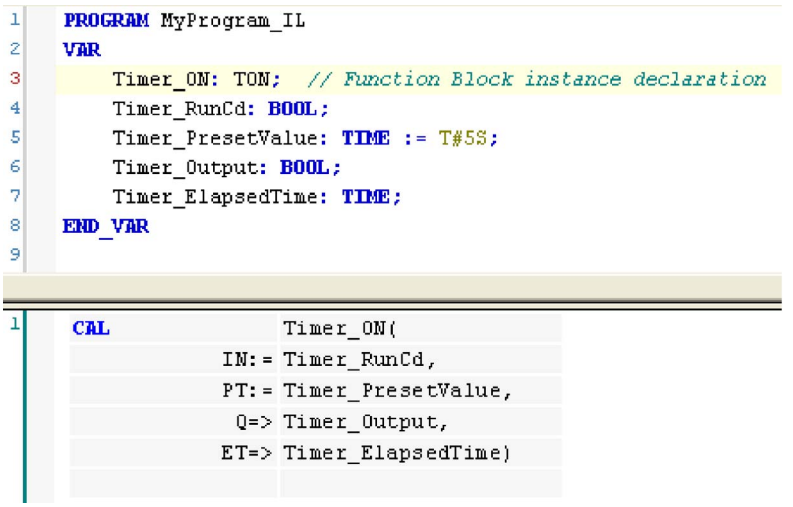
以下过程描述如何用 IL 语言插入一个功能块：

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	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	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

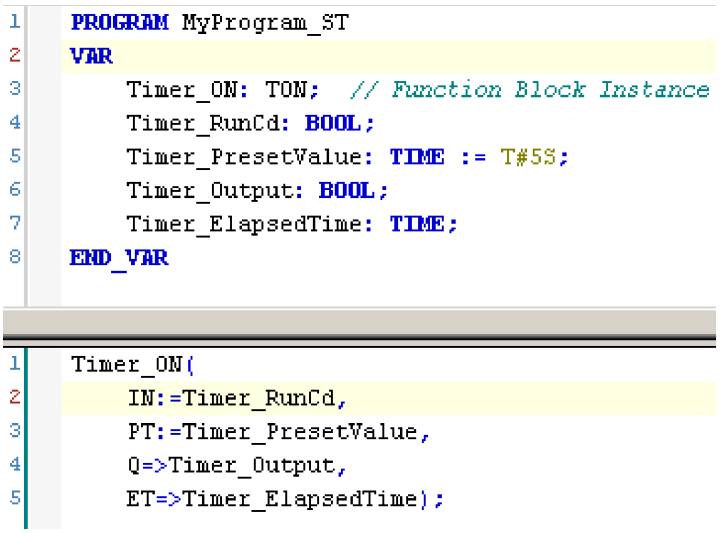
以下过程描述如何用 ST 语言插入一个功能块：

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	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

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。

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*）

反射输出

HSC 的输出是与计数器值进行比较的阈值关联的反射输出，具体取决于 HSC 的配置。反射输出可以在开启或关闭之间切换，具体取决于使用阈值配置的关系。

变量

由程序寻址和修改的存储器单元。

字节

采用 8 位格式编辑的类型，范围从十六进制 00 到十六进制 FF。

应用程序

包括配置数据、符号和文档的程序。

快速 I/O

（*快速输入/输出*）具有某些电子特性（例如，响应时间）的特定 I/O 模块，但对这些通道的处理则直接由控制器完成。

控制器

自动化工业流程（也称为可编程逻辑控制器或可编程控制器）。

控制网络

此网络中包含逻辑控制器、SCADA 系统、PC、HMI、交换机……

支持以下两种拓扑：

- 扁平：此网络中的所有模块和设备都属于同一个子网。
- 2 层：网络分为操作网络和控制器间网络。

这两个网络可以在物理上独立，但通常通过路由设备链接。

程序

应用程序的组成部分，其中包括可以在逻辑控制器的存储器中安装的经过编译的源代码。

编码器

用于测量长度或角度的设备（线性或旋转编码器）。

网络

共享一个公用数据路径和通讯协议的各种互联设备系统。

节点

通讯网络上的可寻址设备。

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。

CPDM

（*控制器配电模块*）控制器到外部 24 Vdc 电源的连接，并且是本地配置的配电起点。

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*）一种对控制器或扩展模块输入上的脉冲进行计数的功能。

I/O

（*输入/输出*）

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，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50-50的固定占空比在开启和关闭之间振荡以产生方波形式的快速输出。PTO 特别适用于如步进电机、频率转换器和伺服电机控制等应用。

PWM

（*脉冲宽度调制*）以可调占空比在关闭和开启之间振荡以产生矩形波形式的快速输出（尽管可以调整它来产生方形波）。

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。

索引

Busy
状态变量的管理, *161*
CommandAborted
状态变量的管理, *161*
Done
状态变量的管理, *161*
ErrID
处理检测到的错误, *161*
状态变量的管理, *161*
Error
处理检测到的错误, *161*
状态变量的管理, *161*
Execute
状态变量的管理, *161*
EXPERT_ERR_TYPE
数据类型, *165*
EXPERT_PARAMETER_TYPE
数据类型, *166*
EXPERT_TIMEBASE_TYPE
数据类型, *168*
EXPERTGetCapturedValue
功能块, *172*
EXPERTGetDiag
功能块, *174*
EXPERTGetImmediateValue
功能, *170*
EXPERTGetParam, *176*
EXPERTSetParam
功能块, *178*
HSC Main 类型配置
一次性模式, *41*
事件模式, *89*
模数回路模式, *61*, *125*
自由大型模式, *77*
HSC Simple 类型配置
一次性模式, *35*
模数回路模式, *55*
HSC_REF
数据类型, *167*
HSCMain
Main 类型 HSC 计数器, *183*
捕捉, *145*
比较, *137*
MC_IMMEDIATE_ERR_TYPE
数据类型, *164*
专用特性, *160*
事件计数
嵌入 HSC 的 HSC 模式, *85*
功能
EXPERTGetImmediateValue, *170*
功能与功能块的区别, *190*
启用, *156*
如何通过 IL 语言使用功能或功能块, *191*
如何通过 ST 语言使用功能或功能块, *195*
功能块
EXPERTGetCapturedValue, *172*
EXPERTGetDiag, *174*
EXPERTGetParam, *176*
EXPERTSetParam, *178*
HSCMain, *183*
编码器, *180*
启用
准许计数操作, *156*
周期计
参数, *115*
描述, *107*
纲要, *110*
编程, *112*
配置, *111*
处理检测到的错误
ErrID, *161*
Error, *161*
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, *145*
编码器, *148*

数据类型
EXPERT_ERR_TYPE, *165*
EXPERT_PARAMETER_TYPE, *166*
EXPERT_TIMEBASE_TYPE, *168*
HSC_REF, *167*
MC_IMMEDIATE_ERR_TYPE, *164*
模数回路
嵌入式 HSC 的 HSC 模式, *49*
比较
HSCMain, *137*
状态变量的管理
Busy, *161*
CommandAborted, *161*
Done, *161*
ErrID, *161*
Error, *161*
Execute, *161*
编码器
功能块, *180*
捕捉, *148*
编码器模式
递增, *120*
自由大型
嵌入式 HSC 的 HSC 模式, *70*
计数器功能块
HSCMain, *183*
诊断功能
EXPERTGetDiag, *174*
调整功能
EXPERTGetParam, *176*
EXPERTSetParam, *178*
递增
编码器模式, *120*
频率计
描述, *97*
纲要, *100*
编程, *102*
配置, *101*