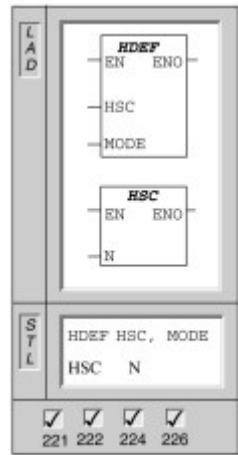


高速计数器指令

高速计数器定义，高速计数器



定义高速计数器指令为指定的高速计数器分配一种工作模式，见表 9-5。

高速计数器指令 (HSC) 执行时，根据 HSC 特殊存储器位的状态，设置和控制高速计数器的工作模式。参数 N 指定了高速计数器号。

CPU 221 和 CPU 222 不支持 HSC1 和 HSC2。

每个高速计数器只能用 1 个 HDEF。

使 ENO=0 的 HDEF 出错条件：
SM4.3 (运行时间)，0003 (输入冲突)，0004 (中断中的非法指令)，000A (HSC 重定义)

使 ENO=0 的 HSC 出错条件：
SM4.3 (运行时间) 0001 (在 HDEF 前使用 HSC/HDEF)，0005 (同时操作 HSC/PLS)

输入/输出	操 作 数	数据类型
HSC	常数	BYTE
MODE	常数	BYTE
N	常数	WORD

理解高速计数器指令

高速计数器累计 CPU 扫描速率不能控制的高速事件，可以配置最多 12 种不同的操作模式，这些操作模式在表 9-5 中列出。高速计数器的最高计数频率有赖于 CPU 的型号，有关 CPU 的详细信息参阅附录 A。

每个计数器对它所支持的时钟，方向控制，复位和启动都有专用的输入。对于两相计数器，两个时钟可以同时以最大速率工作。对正交模式，可以选择以单倍 (1X) 或 4 倍 (4X) 最大计数速率工作。HSC1 和 HSC2 互相完全独立，并且不影响其它的高速功能。所有高速计数器可同时以最高速率工作而互不干扰。

使用高速计数器

一般来说，高速计数器被用作驱动鼓形计时器设备，该设备有一个安装了增量轴式编码器的轴以恒定的速度转动。轴式编码器每圈提供一个确定的计数值和一个复位脉冲。来自轴式编码器的时钟和复位脉冲做为高速计数器的输入。高速计数器装入一组预置值中的第一个值，当前计数值小于当前预置值时，希望的输出有效。计数器设置成在当前值等于预置值和有复位时产生中断。

随着每次当前计数值等于预置值的中断事件的出现，一个新的预置值被装入，并重新设置下一个输出状态。当出现复位中断事件时，设置第一个预置值和第一个输出状态，这个循环又重新开始。

由于中断事件产生的速率远低于高速计数器的计数速率，用高速计数器可实现精确控制，而与 PLC 整个扫描周期的关系不大。采用中断的方法允许在简单的状态控制中用独立的中断程序装入一个新的预置值，这样使得程序简单直接，并容易读懂。当然，也可以在一个中断程序中处理所有的中断事件。若要更详细地了解中断，请参阅 9.15 节。

理解高速计数器的详细时序

下面的时序图 (图 9-10 到图 9-16) 按模式给出了每个计数器是如何工作的。复位和启动输入的操作用独立的时序图表示，并且对所有用到复位和启动输入的种类都给出了时序图。在复位和启动输入图中，复位和启动都编程为高电平有效。

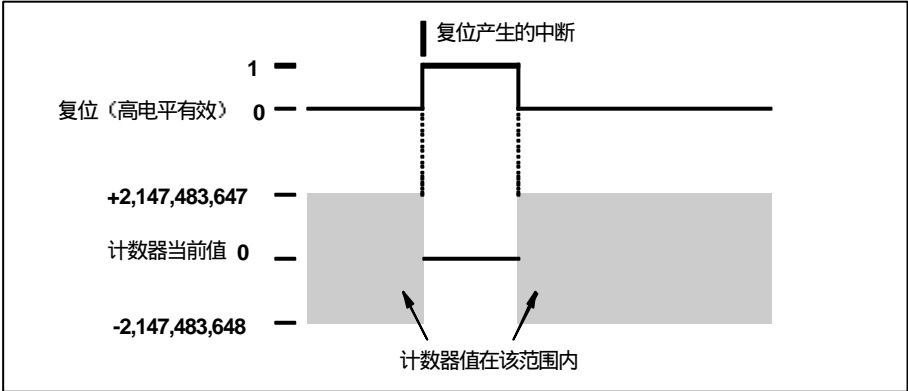


图 9-

10 有复位无启动的操作举例

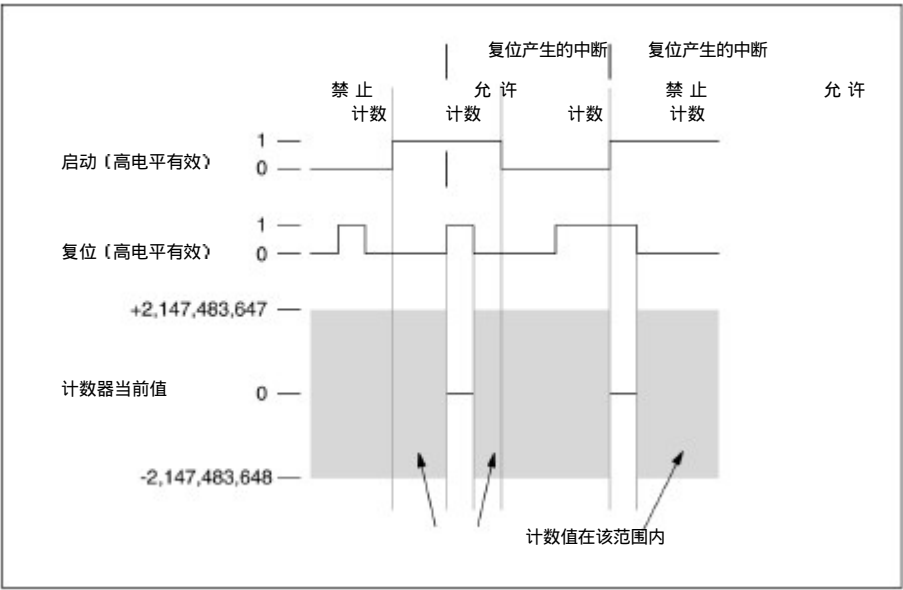


图 9-11 有复位和启动的操作举例

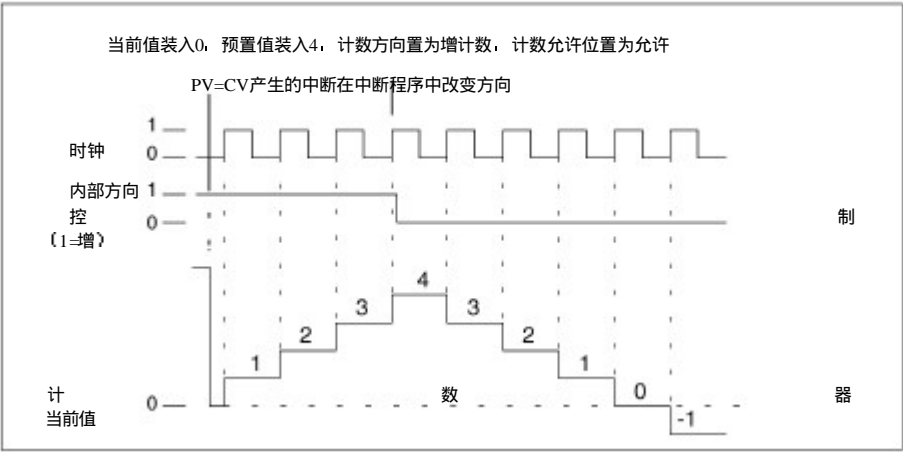


图 9-12 模式 0、1 或 2 的操作举例

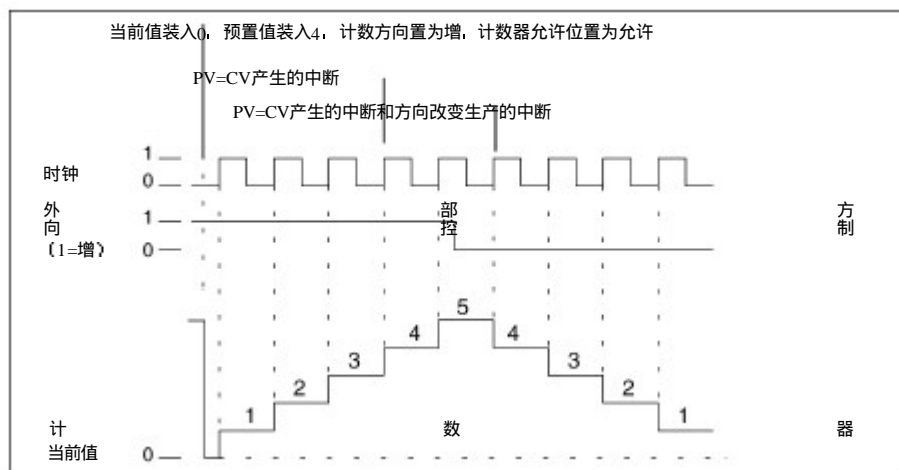


图 9-13 模式 3，4 或 5 的操作举例

当采用计数模式 6，7 或 8 时，若增时钟和减时钟的上升沿出现彼此相差不到 0.3 ms，高速计数器会认为这些事件是同时发生的。如果出现这种情况，当前值不会发生变化，也不会有计数方向变化的指示。当增时钟和减时钟的上升沿距离大于这个时间段 (0.3 ms) 时，高速计数器可以分别捕获到每一个独立事件。在任一情况下，都不会有错误产生，计数器会保持正确的计数值。请见图 9-14, 图 9-15 和图 9-16。

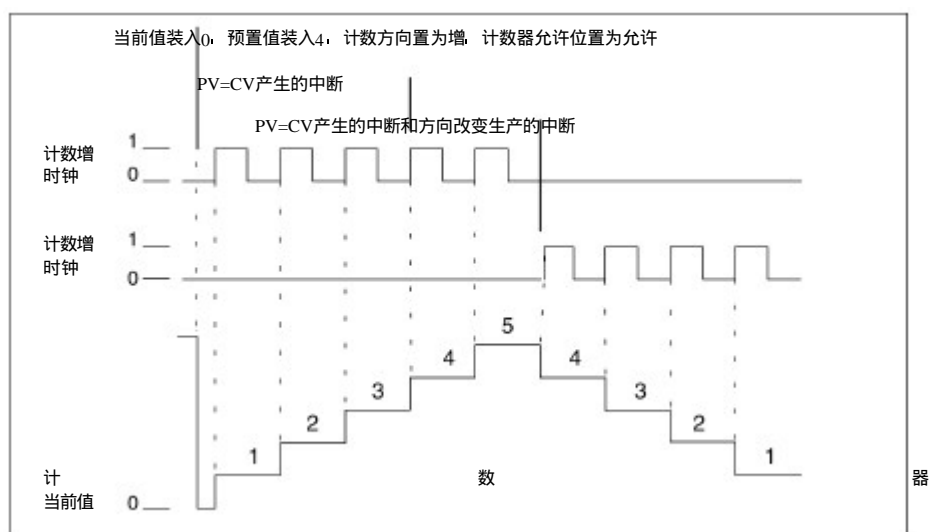


图 9-14 模式 7，8 或 9 的操作举例

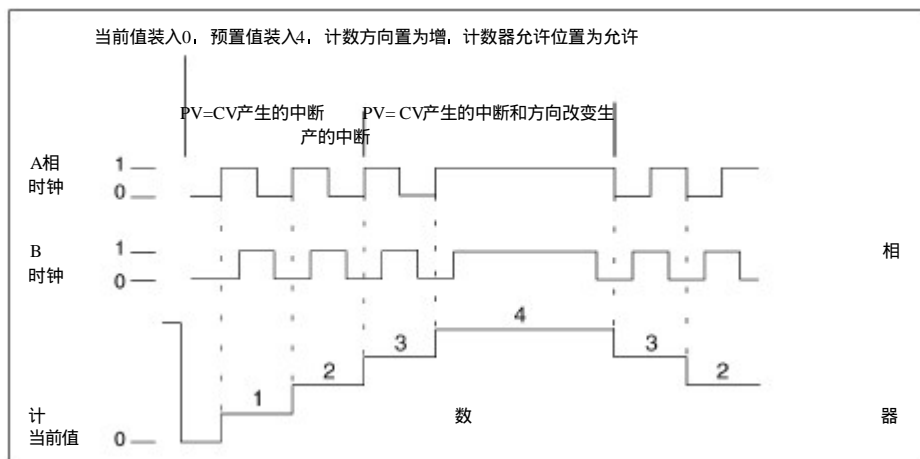


图 9-15 模式 9，10 或 11 (正交 1X 模式) 的操作举例

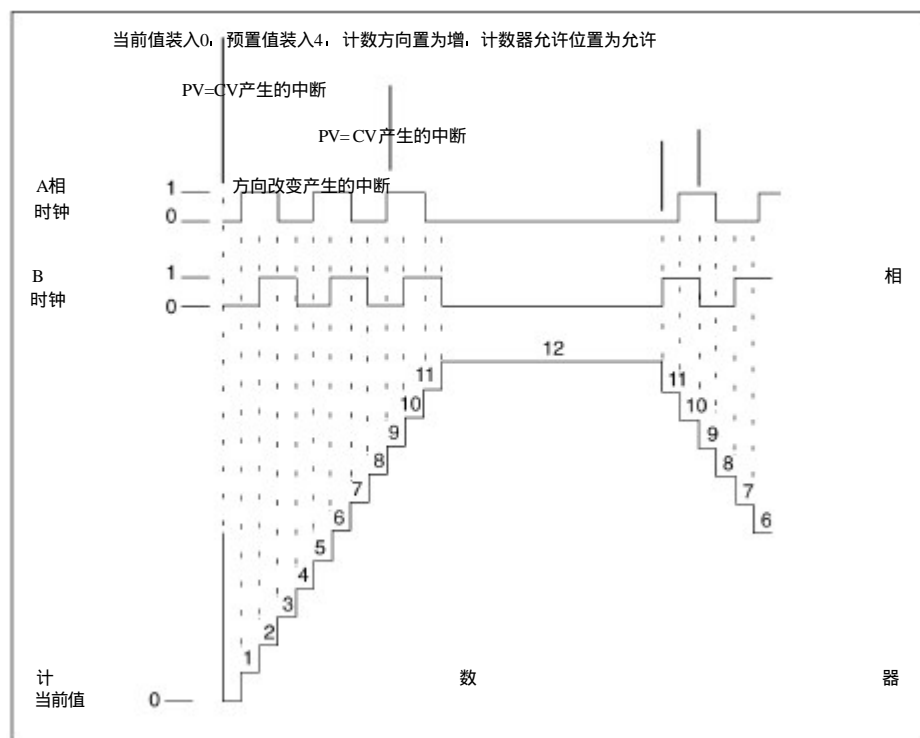


图 9-16 模式 9，10 或 11 (正交 4X 模式) 的操作举例

高速计数器输入线的连接

表 9-3 给出了高速计数器的时钟、方向控制、复位和启动所使用的输入。这些输入功能描述见表 9-5 到 9-10。

表 9-3 高速计数器的指定输入

高速计数器	使用的输入
HSC0	I0.0, I0.1, 0.2
HSC1	I0.6, I0.7, I1.0, I1.1
HSC2	I1.2, I1.3, I1.4, I1.5
HSC3	I0.1
HSC4	I0.3, I0.4, I0.5
HSC5	I0.4

如表 9-4 所示，高速计数器和边沿中断的输入点分配存在一些重叠。同一个输入不能用于两个不同的功能，但是，不使用高速计数的输入端可以作它用。例如，如果 HSC0 工作于模式 2，它使用 I0.0 和 I0.2，于是，I0.1 可以用于 HSC3 的边沿中断。

如果 HSC0 的模式不使用输入 I0.1，那么该输入端可以用作 HSC3 或边沿中断。同样地，如果在选择的 HSC0 模式中使用 I0.2，该输入端可以作边沿中断；如果在选择的 HSC4 模式中使用 I0.4，该输入端可以用为 HSC5 所用。注意：HSC0 的所有模式都使用 I0.0，HSC4 的所有模式都使用 I0.3。所以，当使用这些计数器时这些点不能作它用。

表 9-4 高速计数器和边沿中断的输入点分配

		输入点 (I)													
高速计数器		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0	1.1	1.2	1.3	1.4	1.5
HSC0		x	x	x											
HSC1								x	x	x	x				
HSC2												x	x	x	x
HSC3			x												
HSC4					x	x	x								
HSC5						x									
边沿中断		x	x	x	x										

表 9-5 HSC0 操作模式 (CPU 221、CPU 222、CPU 224 和 CPU 226)

HSC0					
模式	描 述	I0.0	I0.1	I0.2	
0	带内部方向控制的单相增/减计数器 SM37.3 = 0，减计数	时钟			
1	SM37.3 = 1，增计数			复位	
3	带外部方向控制的单相增/减计数器 I0.1 = 0，减计数	时钟	方向		
4	I0.1 = 1，增计数			复位	
6	带增减计数时钟输入的双相计数器	时钟	时钟		
7		(增)	(减)	复位	
9	A/B 相正交计数器 A 相超前 B 相 90 度，顺时针转动	时钟	时钟		
10	B 相超前 A 相 90 度，逆时针转动	A 相	B 相	复位	

表 9-6 HSC1 操作模式(CPU 224 和 CPU 226)

HSC1					
模式	描 述	I0.6	I0.7	I1.0	I1.1
0	带内部方向控制的单相增/减计数器	时钟		复位	启动
1	SM47.3 = 0，减计数				
2	SM47.3 = 1，增计数				
3	带外部方向控制的单相增/减计数器	时钟	方向	复位	启动
4	I0.7 = 0，减计数				
5	I0.7 = 1，增计数				
6	带增减计数时钟输入的双相计数器	时钟 (增)	时钟 (减)	复位	启动
7					
8					
9	A/B 相正交计数器	时钟	时钟	复位	启动
10	A 相超前 B 相 90 度，顺时针转动				
11	B 相超前 A 相 90 度，逆时针转动				

表 9-7 HSC2 操作模式(CPU 224 和 CPU 226)

HSC2					
模式	描 述	I1.2	I1.3	I1.4	I1.5
0	带内部方向控制的单相增/减计数器	时钟		复位	启动
1	SM57.3 = 0，减计数				
2	SM57.3 = 1，增计数				
3	带外部方向控制的单相增/减计数器	时钟	方向	复位	启动
4	I1.3 = 0，减计数				
5	I1.3 = 1，增计数				
6	带增减计数时钟输入的双相计数器	时钟 (增)	时钟 (减)	复位	启动
7					
8					
9	A/B 相正交计数器	时钟	时钟	复位	启动
10	A 相超前 B 相 90 度，顺时针转动				
11	B 相超前 A 相 90 度，逆时针转动				

表 9-8 HSC3 操作模式(CPU 221、CPU 222、CPU 224 和 CPU 226)

HSC3				
模式	描 述	I0.1		
0	带内部方向控制的单相增/减计数器 SM137.3 = 0，减计数 SM137.3 = 1，增计数	时钟		

表 9-9 HSC4 操作模式 (CPU 221、CPU 222、CPU 224 和 CPU 226)

HSC4					
模式	描 述	I0.3	I0.4	I0.5	
0	带内部方向控制的单相增/减计数器 SM147.3 = 0，减计数 SM147.3 = 1，增计数	时钟			
1				复位	
3	带外部方向控制的单相增/减计数器 I0.4 = 0，减计数 I0.4 = 1，增计数	时钟	方向		
4				复位	
6	带增减计数时钟输入的双相计数器	时钟 (Up)	时钟 (Dn)		
7				复位	
9	A/B 相正交计数器 A 相超前 B 相 90 度，顺时针转动 B 相超前 A 相 90 度，逆时针转动	时钟 A 相	时钟 B 相		
10				复位	

表 9-10 HSC5 操作模式 (CPU 221、CPU 222、CPU 224 和 CPU 226)

HSC5					
模式	描 述	I0.4			
0	带内部方向控制的单相增/减计数器 SM157.3 = 0，减计数 SM157.3 = 1，增计数	时钟			

访问高速计数器 (HC)

存取高速计数器的计数值，必须指明高速计数器的地址，并采用 HC 类型和计数器号 (例如 HCS0)。高速计数器的当前值是只读的，并且只能用双字 (32 位) 来寻址，如图 9-17 所示。

格式：HC [高速计数器号] HC2

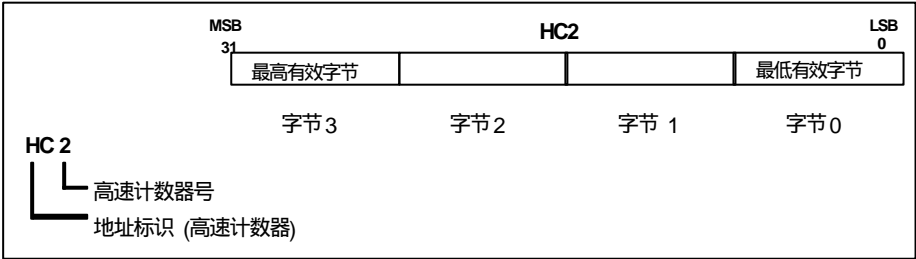


图 9-

17 存取高速计数器的当前值

对高速计数器差异的理解

所有计数器在相同的工作模式下有相同的功能。如表 9-5 所示共有 4 种基本的计数模式。你可使用下列类型：无复位或启动输入，有复位无启动输入，或同时有复位和启动输入。
当激活复位输入，就清除当前计数值并保持到复位无效。当激活启动输入，就允许计数器计数。当启动输入无效时，计数器的当前值保持不变，时钟事件被忽略。如果在启动输入保持无效时，复位有效，则复位被忽略，当前值不变。如果在复位保持有效时，启动变为有效，则计数器的当前值被清除。

使用高速计数器前，必须选定一种工作模式，你可以用 HDEF 指令 (定义高速计数器) 做到这件事。HDEF 给出了高速计数器 (HSCx) 和计数模式之间的联系。对每个高速计数器只能使用一条 HDEF 指令。可利用初次扫描存储器位 SM0.1 (此位仅在第一次扫描周期时接通，然后断开) 调用一个包含 HDEF 指令的子程序来定义高速计数器。

选择有效状态和 1x/4x 模式

四个高速计数器有 3 个控制位，用来设置复位与启动输入的有效状态，以及选择 1x 或 4x 计数方式 (只能是正交计数器)。这些位在每个计数器的控制字节中，只有在执行 HDEF 指令时才有用。这些位的定义见表 9-11。

在执行 HDEF 指令前，必须把这些控制位设定到希望的状态。否则，计数器对计数模式的选择取缺省设置。缺省的设置为：复位和启动输入高电平有效，正交计数速率是 4x (4 倍输入时钟频率)。一旦 HDEF 指令被执行，你就不能再更改计数器的设置，除非先进入 STOP 模式。

表 9-11 复位、启动和 1x/4x 控制位的有效电平

HSC0	HSC1	HSC2	HSC4	描述 (仅当 HDEF 执行时使用)
SM37.0	SM47.0	SM57.0	SM147.0	复位有效电平控制位： 0 = 复位高电平有效；1 = 复位低电平有效
--	SM47.1	SM57.1	--	启动有效电平控制位： 0 = 启动高电平有效；1 = 启动低电平有效
SM37.2	SM47.2	SM57.2	SM147.2	正交计数器计数速率选择： 0 = 4X 计数率；1 = 1X 计数率

控制字节

只有定义了计数器和计数器模式，才能对计数器的动态参数进行编程。每个高速计数器都有一个控制字节，包括下列几项：允许或禁止计数，计数方向控制 (只能是模式 0,1,2) 或对所有其它模式的初始化计数方向，要装入的计数器当前值和要装入的预置值。执行 HSC 指令时，要检验控制字节和有关的当前值及预置值。表 9-12 对这些控制位逐一做了说明。

表 9-12 HSC0、HSC1 和 HSC2 的控制位

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	描 述
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	计数方向控制位： 0=减计数；1=增计数
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	向HSC中写入计数方向： 0=不更新；1=更新计数方向
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	向HSC中写入预置值： 0=不更新；1=更新预置值
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	向HSC中写入新的当前值： 0=不更新；1=更新当前值
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	HSC 允许：0 = 禁止 HSC；1 = 允许 HSC

设定当前值和预置值

每个高速计数器都有一个 32 位的当前值和一个 32 位的预置值。当前值和预置值都是符号整数。为了向高速计数器装入新的当前值和预置值，必须先设置控制字节，并把当前值和 / 或预置值存入特殊存储器字节中，然后必须执行 HSC 指令，从而将新的值送给高速计数器。表 9-13 对保存新的当前值和预置值的特殊存储器字节作了说明。

除了控制字节和新的预置值与当前值保存字节外，每个高速计数器的当前值可利用数据类型 HC (高速计数器当前值)后跟计数器号 (0, 1, 2, 3, 4 或 5) 的格式读出。因此，可用读操作直接访问当前值，但写操作只能用上述的 HSC 指令来实现。

表 9-13 HSC0、HSC1、HSC2、HSC3、HSC4 和 HSC5 的当前值和预置值

要装入的值	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
新当前值	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
新预置值	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162

状态字节

每个高速计数器都有一个状态字节，其中某些位指出了当前计数方向，当前值是否等于预置值，当前值是否大于预置值。表 9-14 对每个高速计数器的状态位作了定义。

表 9-14 HSC0、HSC1、HSC2、HSC3、HSC4 和 HSC5 的状态位

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	描 述
SM36.0	SM46.0	SM56.0	SM136.0	SM146.0	SM156.0	不用
SM36.1	SM46.1	SM56.1	SM136.1	SM146.1	SM156.1	不用
SM36.2	SM46.2	SM56.2	SM136.2	SM146.2	SM156.2	不用
SM36.3	SM46.3	SM56.3	SM136.3	SM146.3	SM156.3	不用
SM36.4	SM46.4	SM56.4	SM136.4	SM146.4	SM156.4	不用
SM36.5	SM46.5	SM56.5	SM136.5	SM146.5	SM156.5	当前计数方向状态位： 0 = 减计数 1 = 增计数
SM36.6	SM46.6	SM56.6	SM136.6	SM146.6	SM156.6	当前值等于预置值状态位： 0 = 不等； 1 = 相等
SM36.7	SM46.7	SM56.7	SM136.7	SM146.7	SM156.7	当前值大于预置值状态位： 0 = 小于等于； 1 = 大于

注:

只有执行高速计数器的中断程序时，状态位才有效。监视高速计数器的状态的目的是使外部事件可产生中断,以完成重要的操作。

HSC 中断

所有高速计数器支持中断条件：当前值等于预置时产生中断。使用外部复位输入的计数器模式支持外部复位有效时产生的中断。除模式 0、1 和 2 外，所有的计数器模式支持计数方向改变的中断，每个中断条件可分别地被允许或禁止。为全面了解中断的用法，请参看 9.15 节。

注:

当使用外部复位中断时，不要写入一个新当前值或者在与那个事件有关的中断程序内禁止然后再允许高速计数器,否则,会造成一个致命错误。

为帮助你理解高速计数器的操作，提供了如下的初始化和操作顺序的描述，并以 HSC1 作为这些描述的计数器模型，即以 HSC1 为例。初始化描述假定 S7-200 已置成 RUN 模式，由于这个原因，初次扫描存储器位为真 (SM0.1=1)。如果不是这种情况，记住在进入 RUN 模式后，对每个高速计数器的 HDEF 指令只能执行一次。对一个高速计数器执行第二个 HDEF 指令会引起运行错误，而且不能改变第一次执行 HDEF 指令时对计数器的设置。

初始化模式 0、1、或 2

HSC1 为内部方向控制的单相增 / 减计数器 (模式 0、1 或 2)，初始化步骤如下：

- 1. 用初次扫描存储器位 (SM0.1=1) 调用执行初始化操作的子程序。由于采用了这样的子程序调用，后续扫描不会再调用这个子程序，从而减少了扫描时间，也提供了一个结构优化的程序。
- 2. 初始化子程序中，根据所希望的控制操作对 SMB47 置数。例如：
SMB47 = 16#F8 产生如下的结果：
允许写入新计数方向为增
写入新的预置值为增
置计数方向为增
置启动和复位输入为高电平有效
- 3. 执行 HDEF 指令时，HSC 输入置 1，MODE 输入置 0 (无外部复位或启动) 或置 1 (有外部复位和无启动) 或置 2 (有外部复位和启动)。
- 4. 用所希望的当前值装入 SMD48 (双字) 中，若装入 0，则清除 SMD48。
- 5. 用所希望的预置值装入 SMD52 (双字) 中。
- 6. 为了捕获当前值 (CV) 等于预置值 (PV) 中断事件，编写中断子程序，并指定 CV=PV 中断事件(事件号13)调用该中断子程序。参看本章 9.15 节，以了解中断处理的细节。
- 7. 为了捕获外部复位事件，编写中断子程序，并指定外部复位中断事件 (事件号 15) 调用该中断子程序。
- 8. 执行全局中断允许指令 (ENI) 来允许 HSC 中断。
- 9. 执行 HSC 指令，使 S7-200 对 HSC1 编程。
- 10. 退出子程序。

初始化模式 3、4 或 5

HSC1 为外部方向控制的单相增 / 减计数器 (模式 3、4 或 5)，初始化步骤如下：

1. 用初次扫描存储器位 (SM0.1=1) 调用执行初始化操作的子程序。由于采用了这样的子程序调用，后续扫描不会再调用这个子程序，从而减少了扫描时间，也提供了一个结构优化的程序。
2. 初始化子程序中，根据所希望的控制操作对 SMB47 置数。例如：
SMB47 = 16#F8 产生如下结果：
允许写入新的当前值
写入新的预置值
置 HSC 的初始计数方向为增
置启动和复位输入为高电平有效
3. 执行 HDEF 指令时，HSC 输入置 1，MODE 输入置 3 (无外部复位或启动) 或置 4 (有外部复位和无启动) 或置 5 (有外部复位和启动)。
4. 用所希望的当前值装入 SMD48 (双字) 中，若装入 0，则清除 SMD48。
5. 用所希望的预置值装入 SMD52 (双字) 中。
6. 为了捕获当前值 (CV) 等于预置值 (PV) 中断事件，编写中断子程序，并指定 CV= PV 中断事件 (事件号 13) 调用该中断子程序。参看本章 9.15 节，以了解中断处理的细节。
7. 为了捕获计数方向改变中断事件，编写中断子程序，并指定计数方向改变中断事件 (事件号 14) 调用该中断子程序。
8. 为了捕获外部复位中断事件，编写中断子程序，并指定外部复位中断事件 (事件号 15) 调用该中断子程序。
9. 执行全局中断允许指令 (ENI) 来允许 HSC 中断。
10. 执行 HSC 指令，使 S7-200 对 HSC1 编程。
11. 退出子程序。

初始化模式 6、7 或 8

HSC1 为具有增 / 减两种时钟的双相增 / 减计数器 (模式 6、7 或 8)，初始化步骤如下：

1. 用初次扫描存储器位 (SM0.1=1) 调用执行初始化操作的子程序。由于使用了这样的子程序调用，后续的扫描不会再调用这个子程序，从而减少了扫描时间，也提供了一个结构优化的程序。
2. 初始化子程序中，根据所希望的控制操作对 SMB47 置数。例如：
SMB47 = 16#F8 产生如下结果：
允许写入新的当前值
写入新的预置值
置 HSC 的初始计数方向为增
置启动和复位输入为高电平有效
3. 执行 HDEF 指令时，HSC 输入置 1，MODE 输入置 6 (无外部复位或启动) 或置 7 (有外部复位和无启动) 或置 8 (有外部复位和启动)。
4. 用所希望的当前值装入 SMD48 (双字) 中，若装入 0，则清除 SMD48。
5. 用所希望的预置值装入 SMD52 (双字) 中。
6. 为了捕获当前值 (CV) 等于预置值 (PV) 中断事件，编写中断子程序，并指定 CV= PV 中断事件 (事件号 13) 调用该中断子程序。参看本章 9.15 节，以了解中断处理的细节。
7. 为了捕获方向改变中断事件，编写中断子程序，并指定计数方向改变中断事件 (事件号 14) 调用该中断子程序。
8. 为了捕获外部复位中断事件，编写中断子程序，并指定外部复位中断事件 (事件号 15) 调用该中断子程序。
9. 执行全局中断允许指令 (ENI) 来允许 HSC1 中断。
10. 执行 HSC 指令，使 S7-200 对 HSC1 编程。

11. 退出子程序。

初始化模式 9、10 或 11

HSC1 为 A/B 相正交计数器 (模式 9、10 或 11)，初始化步骤如下：

1. 用初次扫描存储器位 (SM0.1=1) 调用执行初始化操作的子程序。由于采用了这样的子程序调用，后续的扫描不会再调用这个子程序，从而减少了扫描时间，也提供了一个结构优化的程序。
2. 初始化子程序中，根据所希望的控制操作对 SMB47 置数。

例 如 (1X 计 数 方 式) :
SMB47 = 16#FC 产 生 如 下 的 结 果 :
允 许 新 的 计 数 方 式 的 预 置 值 增
写 入 新 的 初 始 计 数 方 向 为 增
置 HSC 的 初 始 计 数 方 向 为 增
置启动和复位输入为高电平有效

例 如 (4X 计 数 方 式) :
SMB47 = 16#F8 产 生 如 下 的 结 果 :
允 许 新 的 计 数 方 式 的 预 置 值 增
写 入 新 的 初 始 计 数 方 向 为 增
置 HSC 的 初 始 计 数 方 向 为 增
置启动和复位输入为高电平有效

3. 执行 HDEF 指令时，HSC 输入置 1，MODE 输入置 9 (无外部复位或启动) 或置 10 (有外部复位和无启动) 或置 11 (有外部复位和启动)。
4. 用所希望的当前值装入 SMD48 (双字) 中，若装入 0，则清除 SMD48。
5. 用所希望的预置值装入 SMD52 (双字) 中。
6. 为了捕获当前值 (CV) 等于预置值 (PV) 中断事件，编写中断子程序，并指定 CV= PV 中断事件 (事件号 13) 调用该子程序。参见本章 9.15 节，以了解中断处理的细节。
7. 为了捕获计数方向改变中断事件，编写中断子程序，并指定计数方向改变中断事件 (事件号 14) 调用该中断子程序。
8. 为了捕获外部复位中断事件，编写中断子程序，并指定外部复位中断事件 (事件号 15) 调用该中断子程序。
9. 执行全局中断允许指令 (ENI) 来允许 HSC1 中断。
10. 执行 HSC 指令，使 S7-200 对 HSC1 编程。
11. 退出子程序。

改变模式 0、1 或 2 的计数方向

对具有内部方向控制(模式 0、1 或 2)的单相计数器 HSC1，改变其计数方向的步骤如下：

1. 向 SMB47 写入所需的计数方向:
- SMB47 = 16#90

置 HSC 计数方向为减

允许计数
- SMB47 = 16#98

置 HSC 计数方向为增

允许计数
2. 执行HSC指令，使 S7-200 对 HSC1 编程。

写入新的当前值(任何模式下)

以下步骤描述了如何改变 HSC1 的当前值(任何模式下)：

在改变当前值时，迫使计数器处于非工作状态，此时计数器不再计数，也不产生中断。

1. 向 SMB47 写入新的当前值的控制位：
- SMB47 = 16#C0

写入新的当前值

允许计数
2. 向 SMD48(双字) 写入所希望的当前值(若写入 0，则清除)。
3. 执行 HSC 指令，使 S7-200 对 HSC1 编程。

写入新的预置值(任何模式下)

以下步骤描述了如何改变 HSC1 的预置值(任何模式下)：

1. 向 SMB47 写入允许写入新的预置值的控制位：
- SMB47 = 16#A0

写入新的预置值

允许计数
2. 向 SMD52(双字) 写入所希望的预置值。
3. 执行 HSC 指令，使 S7-200 对 HSC1 编程。

禁止 HSC (任何模式下)

以下步骤描述了如何禁止 HSC1 高速计数器(任何模式下)：

1. 写入 SMB47 以禁止计数：
- SMB47 = 16#00

禁止计数
2. 执行 HSC 指令，以禁止计数。

虽然上面依次给出了如何单独改变计数方向、当前值和预置值，但实际上你可以在同一步中，通过对 SMB47 设置适当的值去改变所有的或其中的任意几个，然后执行 HSC 指令。

高速计数器举例

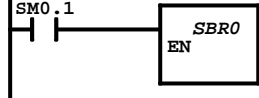
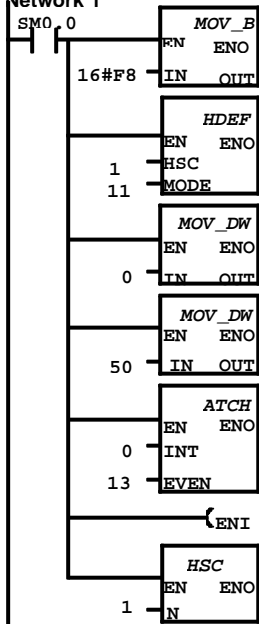
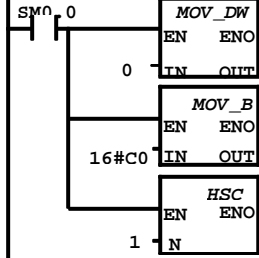
LAD	STL
MAIN OB1	
Network 1	
	Network 1
初次扫描，调用子程序 0	LD SM0.1
主程序结束	CALL 0
SUBROUTINE 0	
Network 1	
	Network 1
允许计数器	LD SM0.0
写入新当前值，写入新预置	MOVB 16#F8,SMB47
设定计数初始方向为增，	HDEF 1,11
设定启动和复位输入的有效电	MOVD 0,SMD48
平为高，设定为 4X 模式	MOVD 50,SMD52
HSC1 设置为带有复位和启动	ATCH 0,13
输入的正交模式	ENI
清除 HSC1 的当前值	HSC 1
设定 HSC1 的预置值为 50	
HSC 1 当前值= 预置值（事	
件 13）连到中断程序 0	
允许全局中断	
编程 HSC1.	
INTERRUPT 0	
Network 1	
	Network 1
清除 HSC1 的当前值	LD SM0.0
写入新当前值并允许计数器	MOVD 0,SMD48
编程 HSC1.	MOVB 16#C0,SMB47
	HSC 1

图 9-18 初始化 HSC1 的举例 (LAD 和 STL)

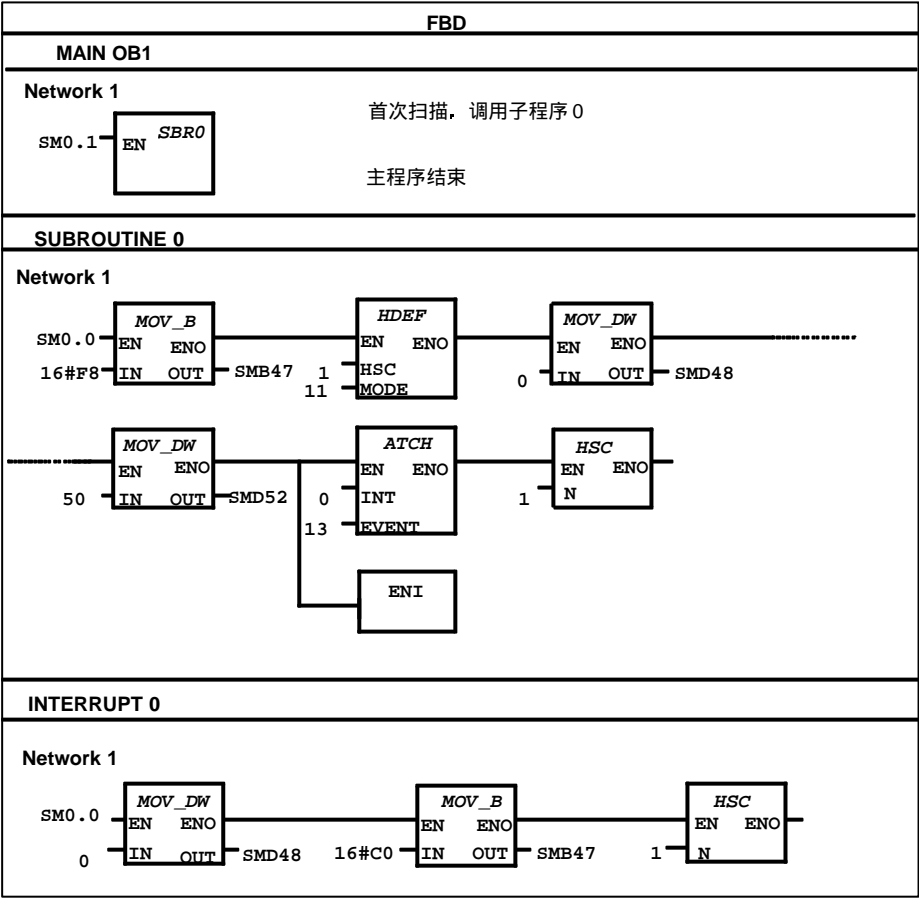


图 9-

19 初始化 HSC1 的举例 (FBD)