

DIGSI

CFC

使用手册

前言, 目录

产品说明

1

开始工作

2

应用实例

3

CFC 功能块

4

文献, 词汇表

安全使用说明

在使用DIGSI CFC产品的过程中，会出现各种具体的操作状况，需要用户采取更多具体的应对措施，因此本手册不可能将全部的安全措施一一列出。但是本手册中列出的信息都是用户应该遵照执行的，只有这样才能确保人身安全，也才能避免不必要的财产损失。本手册中的安全说明用三角形的警告符号表示，根据危险程度的不同，分为下面几种：



警告

警告的意思是说如果用户没有采取相应的预防措施，可能导致人员死亡或者重大财产损失。

谨慎

谨慎的意思是说如果用户没有采取相应的预防措施，可能导致轻微伤害或者少量财产损失。



合格人员

只有能力合格的人员才允许对本手册介绍的设备（功能块或者装置）进行操作。从遵照本手册安全命令的角度讲，合格人员指的是能够按照安全标准对本手册说明的产品装置或者系统进行调试、安装使用、产品接地、鉴别或者线路连接的人员。

使用说明

本手册中的设备（装置、功能块）只允许在本手册的目录和技术说明列出的情况中使用。配套产品只能使用西门子公司推荐或者批准的产品

要安全正确地使用本手册中列出的产品，需要正确地运输、存储、安装、操作和维护这些产品。本手册中的产品在使用过程中存在危险电压，如果不能正确进行操作，可能导致严重的人身伤害或者重大的财产损失。

- 在将装置与其他组件进行连接之前，一定要与保护性接地端子进行连接。
- 装置与电源连接的所有线路都存在危险电压。
- 在将装置与电源切断以后还可能存在危险电压，例如装置中的电容器还可能处于充电状态。
- 带有变流器的装置不能在敞开的状态下进行操作。

使用装置过程中，不能超出本手册或者有关操作指南规定的操作限度。在进行产品测试或者调试的过程中也应该遵守本手册规定的操作限度。

免责声明

我们已经对本手册的内容进行了检查，并且已经做了最大的努力保证手册对于硬件和软件的说明是准确的。其中的偏差不能完全避免，因此如果本手册中存在与实际情况不符合或者存在缺陷或者疏忽的情况，本公司不承担任何相关责任。我们将定期对本手册中的有关内容进行检查，并且对有关内容在后续版本中进行必要的纠正或者补充。如果用户能够在使用本手册的过程中提出建议或者改进方案，我们将表示十分的感谢。

如有更改，恕不另行通知
1.00.01

版权

本手册受西门子公司2005版权的保护，如果事先没有西门子公司书面授权，任何个人或机构都不允许对本手册进行分发或者复制，也不允许将手册中的有关内容透露给第三方。如果发生违反版权的行为，当事人应该承担相应的赔偿责任。本手册中的全部版权，包括商标注册权利和实用新型以及设计注册权利，一律归西门子公司所有。

注册商标

DIGSI®是SIEMENS AG产品的注册商标。本手册中的其他标志也可能是西门子的注册商标，如果第三方出于自身的目的进行使用，将侵犯西门子的版权。

前言

本手册的用途

本手册提供下面的有关内容：

- 如何使用DIGSI CFC产品的说明
- 产品开发实例
- 如何使用DIGSI CFC功能块的详细说明

目标读者

本手册是专门为那些将要调试、编程或者使用DIGSI的有关人员准备的。
本手册的读者需要掌握关于DIGSI的基本知识。

手册的有效范围

本手册适用于V4.5版以及更高版本的DIGSI产品

产品开发标准

DIGSI产品是按照ISO 9001标准进行开发的。

更多支持

如果用户对DIGSI产品存在任何疑问，可以与产品销售商联系，也可以使用下面的热线：

热线

- ☐ 热线电话: 01 80 - 5 24 70 00
- ☐ 热线传真: 01 80 - 5 24 24 71
- ☐ 热线Email: ptd.support@siemens.com

培训课程

关于培训课程的更多信息，请联系当地培训中心。

SIEMENS AG
POWER TRANSMISSION & DISTRIBUTION
PTD SE CS
Humboldtstr. 59
90459 Nürnberg (纽伦堡)
电话: 09 11/4 33-70 05
传真: 09 11/4 33-79 29
Email: www.ptd-training.de



注意：

本手册中的图片是在不同的操作系统之下采集的，因此用户系统上面的显示画面可能和本手册中的图片不同。

但是显示的画面中的内容不会因为上面的原因而受到影响。

目录

1	产品说明	1
2	开始工作	3
2.1	SIPROTEC 装置编程	4
2.2	优先级	5
2.3	编程指导	8
2.3.1	CFC 标准法则	8
2.3.2	分离和加入信息项	8
2.3.3	最大许用功能块数目	9
2.4	为 CFC 选择信息项	10
2.5	快速 PLC 处理实例	13
2.5.1	创建一个新 CFC 功能图	14
2.5.2	指定优先级	18
2.5.3	设定功能块	19
2.5.4	连接输入信号	22
2.5.5	连接输出信号	24
2.5.6	编译 CFC 功能图	27
2.6	慢速 PLC 处理实例	28
2.6.1	插入新信息	29
2.6.2	将信息配置到指示灯	31
2.6.3	连接输入信号	32
2.6.4	设定功能块	33
2.6.5	连接输出信号	34

2.7	测量值处理实例	35
2.7.1	连接输入信号	36
2.7.2	连接功能块	37
2.7.3	连接输出信号	38
2.8	联锁实例	39
2.8.1	连接输入信号	40
2.8.2	增加输入信号的数目	41
2.8.3	连接功能块	42
2.8.4	连接输出信号	43
3	应用实例	44
3.1	定值组切换功能	45
3.2	闪烁的指示灯	57
3.3	反向联锁	62
3.3.1	短路现象发生后应该出现的联锁状态	62
3.3.2	通过分散布线进行反向联锁	63
3.3.3	通过 IEC 61850 进行反向联锁	64
3.3.4	图：通过 CFC 功能图进行的反向联锁	65
3.4	操作计数	65
4	CFC 功能块	66
4.1	数据类型	67
4.2	算术运算	69
4.2.1	ABSVALUE（绝对值）	70
4.2.2	ADD（求和）	71
4.2.3	DIV（除法）	72
4.2.4	MUL（乘法）	73
4.2.5	SQUARE_ROOT（平方根）	74
4.2.6	SUB（减法）	75

4.3	基本逻辑运算	76
4.3.1	AND (“与” 门)	77
4.3.2	CONNECT (连接)	78
4.3.3	DYN_OR (动态 “或”)	79
4.3.4	NAND (“与非” 门)	81
4.3.5	NEG (“非” 门)	82
4.3.6	NOR (“或非” 门)	83
4.3.7	OR (“或” 门)	84
4.3.8	RISE_DETECT (上升沿检测)	85
4.3.9	X_OR (“异或” 门)	86
4.4	信息状态	87
4.4.1	CV_GET_STATUS	90
4.4.2	DI_GET_STATUS	91
4.4.3	DI_SET_STATUS	92
4.4.4	MV_GET_STATUS	94
4.4.5	MV_SET_STATUS	95
4.4.6	SI_GET_STATUS	97
4.4.7	SI_SET_STATUS	98
4.4.8	ST_AND	100
4.4.9	ST_NOT	102
4.4.10	ST_OR	103
4.5	存储器	105
4.5.1	D_FF	106
4.5.2	D_FF_MEMO	108
4.5.3	RS_FF	110
4.5.4	RS_FF_MEMO	112
4.5.5	SR_FF	114

4.5.6	SR_FF_MEMO	116
4.5.7	MEMORY	118
4.6	控制命令	119
4.6.1	BOOL_TO_CO	120
4.6.2	BOOL_TO_IC	124
4.6.3	BOOL_TO_IE.....	128
4.6.4	CMD_CANCEL	132
4.6.5	CMD_CHAIN	134
4.6.6	CMD_INF	139
4.6.7	LOOP	141
4.7	类型转换器.....	143
4.7.1	BOOL_TO_DI	144
4.7.2	BUILD_DI.....	146
4.7.3	DI_TO_BOOL	148
4.7.4	DINT_TO_REAL	152
4.7.5	DIST_DECODE	153
4.7.6	DM_DECODE	155
4.7.7	REAL_TO_DINT	157
4.7.8	REAL_TO_INT	158
4.7.9	REAL_TO_UINT	159
4.7.10	INT_TO_REAL.....	161
4.7.11	UINT_TO_REAL	162
4.8	数据比较	163
4.8.1	COMPARE	164
4.8.2	LIVE_ZERO	167
4.8.3	LOWER_SETPOINT.....	169
4.8.4	UPPER_SETPOINT	171
4.8.5	ZERO_POINT	172
4.9	计量值.....	174

4.9.1	COUNTER	174
4.10	计时器和时钟	176
4.10.1	ALARM	177
4.10.2	BLINK	179
4.10.3	LONG_TIMER.....	181
4.10.4	TIMER.....	183
4.10.5	TIMER_SHORT	188

文献

词汇表

产品说明

DIGSI CFC

DIGSI CFC（DIGSI连续功能图）是DIGSI4中的一个组件，可以用于为SIPROTEC装置编程。

这个产品中的图形界面可以将用户的信息项与程序相连，以实现如联锁和顺序操作等功能。另外，用户也可以对测量值进行编辑，并且生成相应的信息。

信息

经常使用的信息包括下列内容：

- 开关装置的位置和其它过程元素
- 测量值
- 间隔层和装置所处状态的二进制信息
- 保护性信息
- 一般信息
- 中断信息

用户可以在 DIGSI4配置矩阵中将所需信息分给到DIGSI CFC。

信息互连

信息条可以通过下面的应用块的方式在DIGSI CFC中相互连接起来

- 算术运算
- 基本逻辑运算
- 信息状态
- 存储器
- 控制命令
- 类型转换器
- 数据比较
- 计量值
- 计时器和时钟

用DIGSI CFC进行工作

在用DIGSI CFC进行工作的时候,用户是通过CFC功能图的方式进行编程工作的。

在CFC功能图中,用户通过CFC功能块的方式将DIGSI 4配置矩阵中的信息相互连接起来。

完成的CFC功能图通过DIGSI CFC进行编译,成为可以运行的程序。

开始工作

概述

在这一章中，用户可以熟悉DIGSI CFC功能图的基本结构，并且学会如何使用它。



注意：

下面将要学习的使用指南是相互关联的，每一个使用指南都包括使用DIGSI CFC的重要信息。用户应该按照顺序一步一步地进行学习，并且注意在每个实例中的总体信息。



注意：

本章中的CFC功能功能块是通过**块宽度：宽的**的选择项打开的，其步骤如下：

- 在CFC编辑器中，点击**选项**→**自定义**→**块/边框宽度**。
- 在**块宽度**的选择项下面点击**宽的**的选择项，然后通过**确定**键予以确认。

目录

2.1	SIPROTEC装置编程	4
2.2	优先级	5
2.3	编程指导	8
2.4	为CFC选择信息项	10
2.5	快速PLC处理实例	13
2.6	慢速PLC处理实例	28
2.7	测量值处理实例	35
2.8	联锁实例	39

2.1 SIPROTEC装置编程

用户可以用DIGSI CFC自行设计在SIPROTEC装置中的自动化功能。

选择信息项

在DIGSI的配置矩阵中选择在设计特定功能过程中对所需要的信息项。

DIGSI CFC

在用DIGSI CFC进行工作的时候，用户是通过CFC功能图的方式进行编程工作的。

优先级

根据需要的处理优先级设置每一个CFC功能图的优先级（例如保护功能）以及每个功能图的处理时间（例如周期）

CFC功能图

在CFC功能图中，应该通过CFC功能块的方式将DIGSI配置矩阵中的信息相互连接起来。

参数设置

在完成CFC功能图的编译工作，并且将DIGSI CFC终止以后，可以将程序与参数设置一起下载到SIPROTEC装置中。



注意：

随后的操作通常是基于菜单命令的方式进行说明的。

许多菜单命令可以通过点击工具栏上面的图标的方式进行启动。

应用**查看** → **工具栏**选项可以将工具栏进行隐藏或者显示。

另外许多功能还可以通过快捷键的方式进行使用。关于快捷键的使用，可以参考菜单命令后面的说明。

用户也可以在状态栏中发现许多有用的信息，可以通过**查看** → **状态栏**的方式将状态栏进行隐藏或者显示：

- 关于当前正在运行的命令的有关信息。
 - 可锁定键的状态。
 - 在CFC功能图中当前功能图的数量。
 - 当前正在运行的优先级（例如**PLC_BEARB [快速PLC]**）。
-

2.2 优先级

通过DIGSI CFC设计的每一个功能，都应该赋予一个特定的优先级。不同的优先级处理的优先顺序和执行时间都存在不同：

- 快速PLC处理(**PLC_BEARB / PLC**)。
- 慢速PLC处理(**PLC1_BEARB / PLC1**)。
- 测量值处理(**MW_BEARB/ MEASURE**)。
- 联锁(**SFS_BEARB / INTERLOCK**)

处理优先权

处理优先权也规定了可以指定优先级的功能的类型。

表2-1 优先级，处理优先权以及DIGSI CFC的指定功能

优先级	处理优先权	指定功能
快速PLC处理 (PLC_BEARB / PLC)	本优先级中的功能是通过事件控制的方式进行处理，优先级最高。 对于每一个逻辑输入信号的变化，都会立刻进行处理。 另外，将定期对全部的功能表进行处理。（每秒钟处理一次） 处于该优先级的某种功能处理可以中断位于慢速PLC(PLC1_BEARB / PLC1)等级的某种功能的处理。	保护功能，保护功能的闭锁。 注意： 在该优先级中，用户进行连接的功能块数量相对于慢速PLC(PLC1_BEARB / PLC1)要少。用户应该遵守SIPROTEC装置的用户手册中的有关规定。
慢速PLC处理 (PLC1_BEARB / PLC1)	本优先级中的功能是通过事件控制的方式进行处理，优先级低于快速PLC处理(PLC_BEARB / PLC)。 对于每一个逻辑输入信号的变化，都会立刻进行处理。 另外，将定期对全部的功能表进行处理。（每秒钟处理一次） 但是处于该优先级的某种功能处理可能被位于快速PLC处理(PLC_BEARB / PLC)等级的某种功能所中断。	事件控制的功能： 例如带有计时器和时钟功能的应用，功能键操作。 注意： 该优先级最好应用在不是保护性功能的逻辑功能操作中。
测量值处理 (MW_BEARB/ MEASURE)	该优先级的功能是周期性地在后台进行处理的。 注意： 具有该优先权功能不是通过事件控制的。	处理测量值：例如计算的功率因数(ANSI55)，计算的视在功率(ANSI32)。
连锁 (SFS_BEARB / INTERLOCK)	该优先级的功能是通过控制命令进行启动的，并且也是周期性地在后台进行处理的。 对于保护功能做出反应的情况，这样的优先权级发生的频率比较低。	连锁： 例如连锁

**注意:**

某些CFC功能块（例如计时器）只能使用在某些特定的优先级中。详情可参见本手册第4章的相关内容

**注意:**

几个CFC功能图可以赋予同一个优先级。
SIPROTEC装置中的系统固件通常只有在处理完同一个优先级内的一个功能图之后才会去处理另外一个功能图。

**注意:**

当采用循环测量方式时，注意在测量值处理（MW_BEARB）和联锁时（SFS_BEARB）：为了可靠地进行检测，输入信号的最小激活时间必须大于或者至少等于分别测量情况下的周期。仅改变一个输入信号不能触发处理功能图过程。

**注意:**

在周期性触发时，处理同级功能图的顺序是随机的而且不可预测。

2.3 编程指导

当采用DIGSI CFC编程时，为了避免在编程过程中出现问题必须查阅并遵守一些指导原则。

2.3.1 CFC标准法则

当采用DIGSI CFC时要遵守下列八个连续的步骤：

1. 将信号分配给CFC
2. 保存配置矩阵
3. 插入CFC功能图
4. 指定优先级
5. 绘制功能图
6. 检查运行顺序
7. 编译功能图
8. 保存参数设置

2.3.2 分离和加入信息项

说明

通过使用DIGSI设置，可以把信息项分成几个信号。如果把这些项合并到一个CFC功能图或CFC功能块中，在编程过程中可能出现故障（例如，CFC可能在没有初始化的情况下就起动）。



注意

避免把一个信息项分成几个信号并在以后把它们合并到一个CFC功能图或CFC块。

解决方案

如果一定要把一个信息项分成几个信号，并要把这些信号合并到CFC功能图或CFC块中，必须把这些信号转换成适于CFC功能图中的CFC块的形式。

2.3.3 最大许用功能块数目

说明

在SIPROTEC装置的各自不同优先级的CFC功能图中，最大许用功能块数目取决于装置的计算能力，并由编译器进行监控。要想了解所用装置的最大许用功能块数量，可查阅选用的SIPROTEC装置手册手册中的技术数据。



注意：

MEMORY, RS_FF_MEMO, SR_FF_MEMO, D_FF_MEMO 和 **COUNTER** 的最大许用功能块数目取决于所提供的非易失性内存，并由CFC编译器进行监控。

在CFC编译功能图时要检查最大许用功能块数目。当出错时，将有一致性错误信号显示。功能块的过量使用会在编译记录中记录。



注意：

以下内容适用于低于V4.5版的SIPROTEC装置：

CMD_CHAIN 功能块的最大许用功能块数在快速PLC处理（优先级为**PLC1_BEARB**）和慢速PLC处理（优先级为**PLC1_BEARB**）中是**20**块。



注意：

对于**TIMER** 和 **TIMER_SHORT**功能块，最大的许用数目是由系统可用计时器所限制的，并被CFC编译器监控。

在编译CFC功能图时要检查最大许用块数目。当出错时会提示有一致性错误，编译记录中提示超出可用资源。

2.4 为CFC选择信息项

选择信息项

采用如下步骤来设定在CFC中想选择的信息：

- 在DIGSI中打开配置矩阵：
- 在导航窗口中选择**设置**。
- 在数据装口中双击**Masking I/O** (配置矩阵)。
- 在所打开的配置矩阵中搜索所想要的信息或增加所需要的信息。在这个过程中，请注意**显示文本**，**长文本**和**类型**中的信息。



注意：

注意配置矩阵的工具栏中的过滤器的设置（例如**测量值**和**计量值**在**无过滤**条件下使用）。

- 为了把信息用作输入信号，显示在CFC的左边界左边界，要在**信息目标**中配置CFC信息。
- 右键点击配置矩阵中的**C**列中的对应单元格，并在所打开的关联菜单中选择**X**（已配置）。

Settings - Masking I/O (Configuration Matrix) - test / 北方区 / 75J632 V4.6/75J632														
	信息				信息来源				信息目标					
	编号	显示文本	L	类型	BI	s	C	测量值	测量值窗口	s	X	C	计量值窗口	D
													C	D
Device, General														
EN100-Modul 1														
P.System Data 1														
Osc. Fault Rec.														
P.System Data 2														
50/51 Overcur.														
Measuram. Superv														
Cntrl Authority														
Control Device														
Process Data														
Measurement										*	*			*
Demand meter											*			
Min/Max meter														
Set Points(MV)		I Admd>		LV					Set Points(MV)		X			
		I Bdmd>		LV					Set Points(MV)		X			
		I Cdmd>		LV					Set Points(MV)		X			
		I Idmd>		LV					Set Points(MV)		X			
		I Pdmd>		LV					Set Points(MV)		X			
		I Qdmd>		LV					Set Points(MV)		X			
		I Sdmd>		LV					Set Points(MV)		X			
	37-1		LV					Set Points(MV)		X				
	IPFK		LV					Set Points(MV)		X				
Energy							*			*	*			
Statistics										*				
SetPoint(Stat)														
Thresh.-Switch														

图2-1 把一个信息设置为CFC中输入信号



注意：

信息配置为右边界的CFC输出信号时，当与CFC功能图的相关信号的值发生改变时，仅在每个装置记录中记录。



注意：

瞬态信息不适于CFC中逻辑运算的输入信号。

瞬态信息在事件型优先级**PLC_BEARB** [快速PLC]和**PLC1_BEARB** [慢速PLC]下可用于触发CFC功能图处理。这样一旦有信号发生，CFC功能图格就会被处理。

2.5 快速PLC处理实例

快速PLC处理（优先级为**PLC_BEARB**）具有最高优先级的。快速PLC由于输入信号改变而执行，这种过程是事件控制型的。每秒发生一次循环过程。



注意：

在下面的实例中给出了订货编号为 **7SJ63655ER633HH3**型SIPROTEC装置的功能介绍。为了更好地理解这个实例，可以从DIGSI装置列表中选择一种类似的装置。

在SIPROTEC装置的功能范围中必须激活功能**67**，**67-TOC** 和**67N**，**67N-TOC**。

任务 由于输入电压线路出错导致方向测量功能闭锁。

输入信号

- 组：**Measur. Superv**（测量，监视）
- 显示文本：**VT Fuse Failure**（VT断线）

输出信号

- 组：**67 Direct. O/C**（67 方向-过流）
- 显示文本：**>BLK 67N/67NTOC**

CFC功能块 **CONNECT (连接)**

操作过程

- 在DIGSI配置矩阵中，配置CFC输入信号和输出信号，详见第2.4节
- 用**BLK DIR FOR MCB TRIP**来创建一个新的CFC功能图。
- 指定优先级为**PLC_BEARB [Fast PLC]**。
- 添加新功能块**CONNECT**。
- 连接输入信号。
- 连接输出信号。
- 编译CFC功能图。

2.5.1 创建一个新CFC功能图

用**BLKDIR FORMCB TRIP**创建一个新功能图的步骤如下：

打开CFC文件夹：

- 在导航窗口中选择**设置**菜单。
- 在数据窗口中双击**CFC**选项。

所有功能图将在数据窗口中显示。



注意：

显示在右边界的DIGSI CFC输出信号不能再次和右边界相连接，从而确保系统的一致性（详见2.5.5节）。

- 点击**插入**→**CFC功能图**菜单。

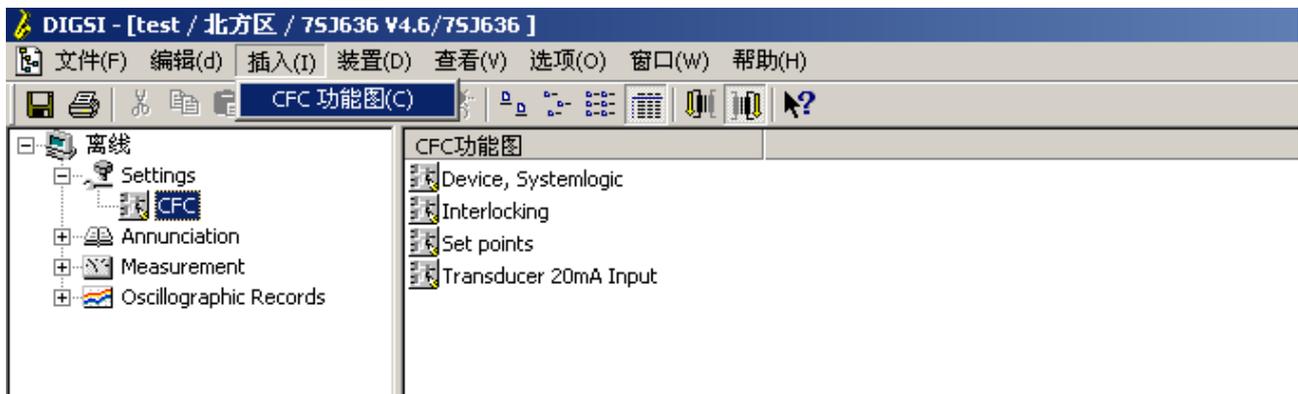


图2-3 插入一个CFC功能图。

- 对新CFC功能图重命名，方法如下：
 - 右键点击自动指定的CFC功能图名称，并在所显示的关联菜单中选择**对象属性...**。
 - 在所显示的**CFC功能图属性**对话框中的**名称**中输入新指定的名称**BLK DIR FOR MCB TRIP**。



图2-4 重新命名CFC功能图。

**注意：**

输入的**作者**和**注释**的内容可用于文档记录：
例如可以在**注释**中输入电子文档修改的历史记录。

- 接受新输入的名称，并点击**确定**，进行确认。

- 打开一个CFC功能图的步骤如下：
- 右键点击CFC功能图名，并在所显示的关联菜单中选择**打开对象**。

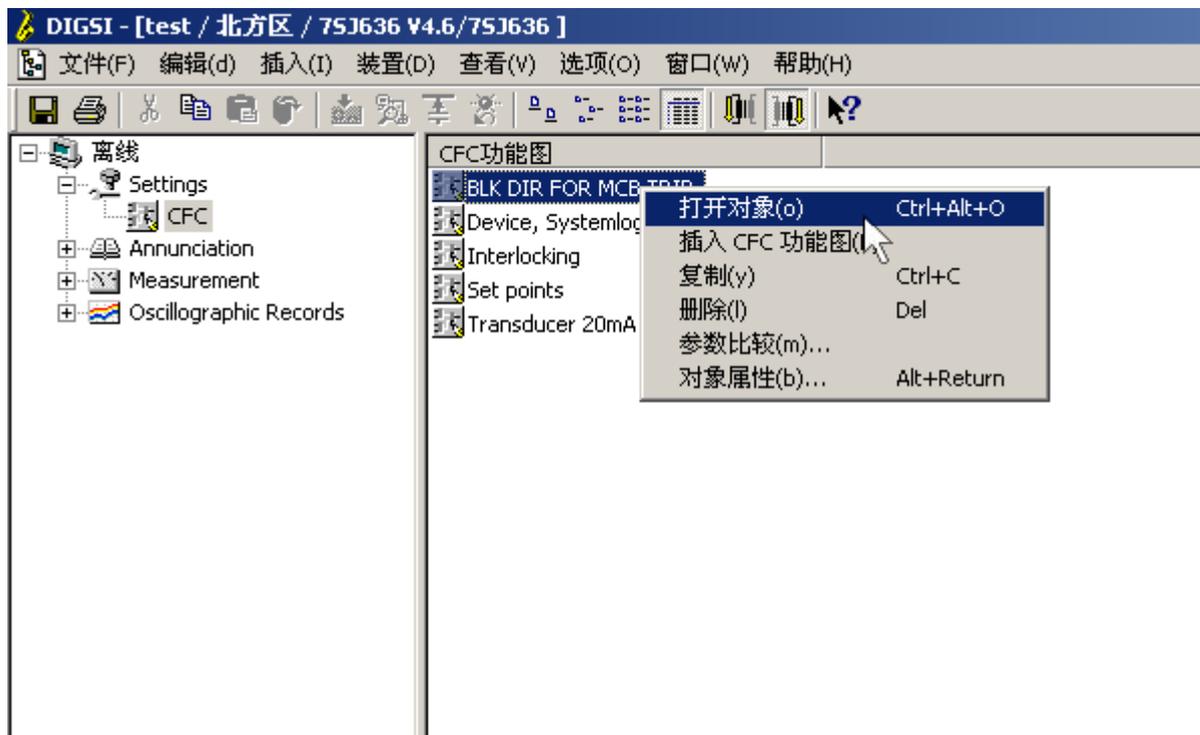


图2—5 打开一个CFC功能图。

- 如果看到概观视图（包括六个表格），切换为表格视图。点击工具栏所对应的图标，如下图所示。

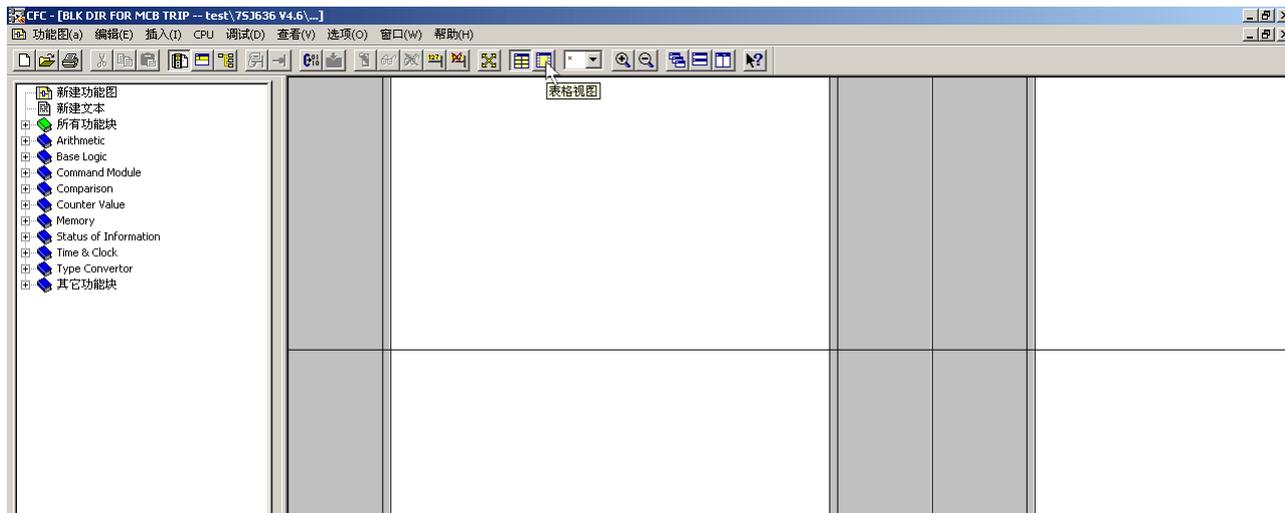


图2—6 将视图切换为表格视图。



注意:

为了更加清楚地查看，可以把一个完整的CFC功能图划分为几个局部功能图（每一个功能图可以包括六个表格）。

可以通过边界直接把局部功能图连接起来。

关于局部图的详细信息可以参见DIGSI CFC的有关在线帮助文档。

2.5.2 指定优先级

指定优先级**PLC_BEARB**：

- 在打开的CFC 功能图中点击**编辑** → **运行序列....**，就会显示**运行时编辑器**窗口。
- 在左侧导航窗格中选择**PLC_BEARB (快速 PLC)**并点击**编辑**→ **插入点位置**。

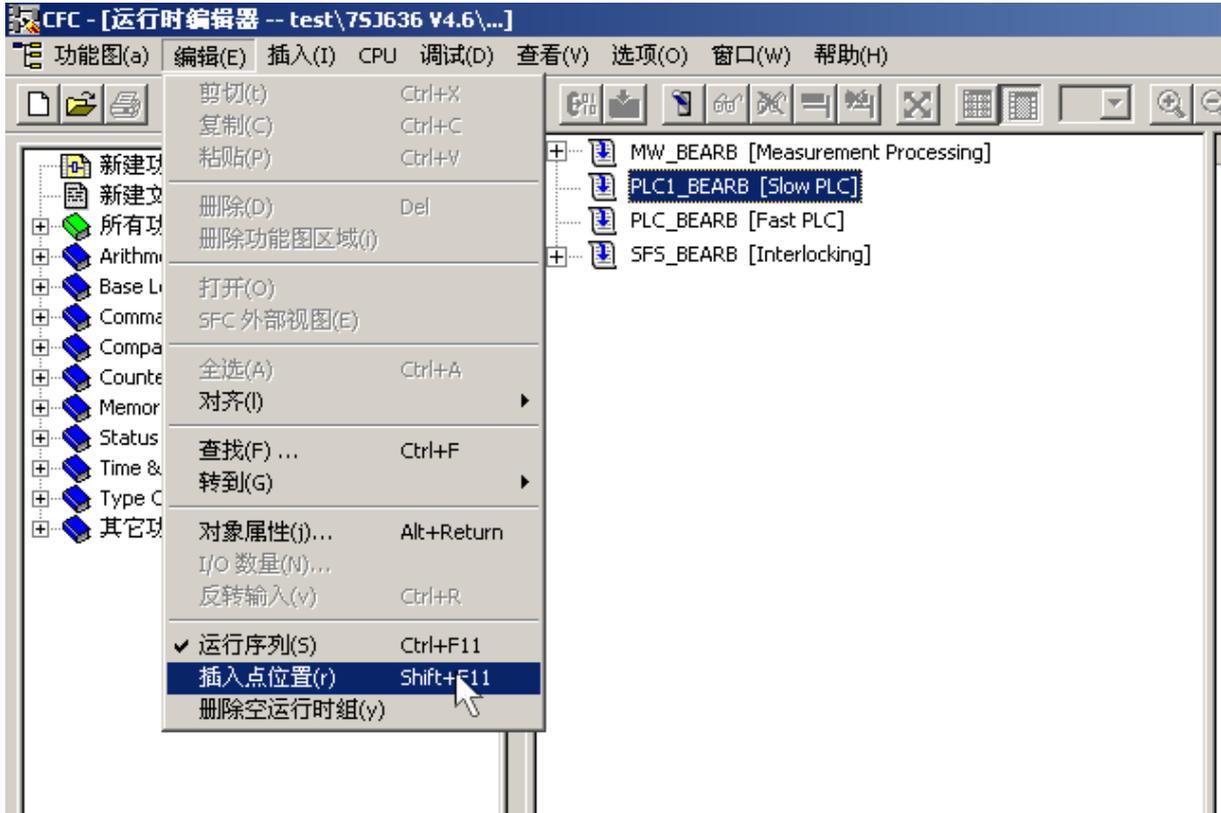


图2—7指定优先级。

- 在确认修改优先级的对话框中点OK确认。

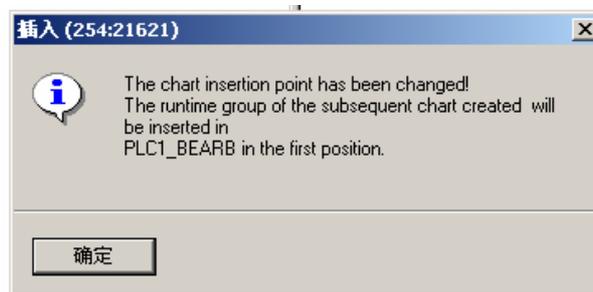


图2—8修改优先级的对话框。

- 在**运行时编辑器**菜单中点击**编辑**→**运行序列...**，CFC功能图就会重新显示。

2.5.3 设定功能块

在CFC功能图中设定**CONNECT**功能块步骤如下：

- 单击右侧目录中的**功能块**的标签页



注意：

关于目录中每个标签中的详细信息可以参见DIGSI CFC的在线帮助。

- 用鼠标左键按下**CONNECT** 块，并保持按下状态，把功能块拖到CFC功能图中。

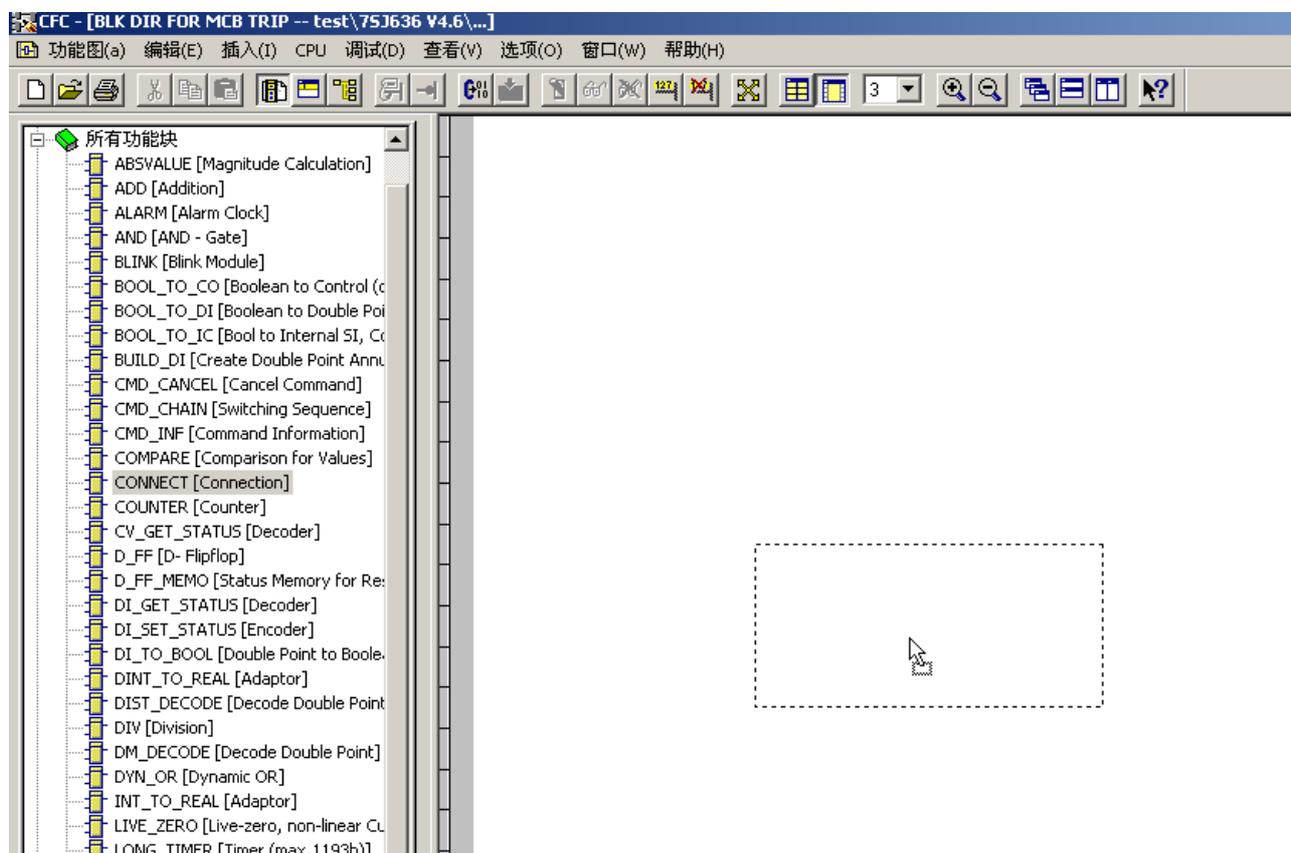


图2—9 通过鼠标的拖放功能来定位一个CFC功能块。

- 释放鼠标的按键， 块就会在表中显示出来。



注意：

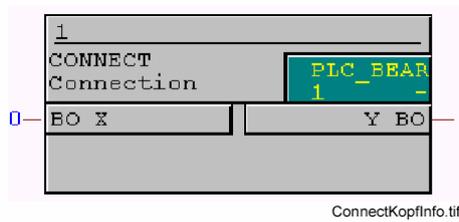


图2—10 功能块标题信息。

放置后的功能块的标题信息包含：

- 功能块名称（例如“1”）。
- 功能块类型（例如“连接”）。
- 功能块优先级（例如“PLC_BEARB”）和优先级下面的运行顺序号（例如“1”）。



注意：

- 一个CFC功能图内的的所有功能块必须具有相同的优先级。



注意：

如果基于用途对CFC功能图的块进行命名，CFC功能图可以更加容易理解：

改变块名称：

- 右键点击功能块，并在所显示的关联菜单中选择**对象属性...**
- 点击**常规**标签，在**名称**的输入框中输入新的名称。
- 点击**确定**进行确认。

**注意:**

在CFC功能图中，相互连接的块是按照一个顺序进行处理的。处理顺序为功能块所显示的序号。序号应与逻辑顺序相一致。

如果在一个CFC功能图配置几个功能块，或在以后插入一个新块，必须检查块的优先级或运行顺序，有时可能要进行调整。

改变功能块的优先级，或者改变功能块在运行队列中的顺序号的步骤如下：

- 右键点击功能块，并在所显示的关联菜单中选择**转到插入点**，就会显示**运行时编辑器**窗口。在窗口中会显示出当前的功能块。
- 从优先级或运行顺序队列中删除功能块的方法是：右键点击功能块并在所显示的关联菜单中选择**剪切**，功能块就会隐藏。
- 把功能块插入到优先级或运行顺序队列中的一个新位置的方法为：右键点击新位置并在所显示的关联菜单中选择**粘贴**，功能块就会插入到一个新位置。
- 在**编辑** → **运行序列...**中点击**运行时编辑器**，CFC功能图就会重新显示。

**注意:**

一个配置好的或相互连接的块可在一个CFC功能图内移动，其方法如下：

- 左键点击功能块的标题，按住鼠标并把块拖动到新位置。
- 放开左键，块和它的连接信息就会在新位置显示。

当改变一个块的位置时，该块的运行顺序不会自动发生改变。

2.5.4 连接输入信号

把**Measur. Superv Fuse Failure**输入信号和**CONNECT**功能块相互关联的步骤如下：

- 右键点击**CONNECT** 块的输入连接**BO X**并在所显示的关联菜单中选择**连接到地址...**。

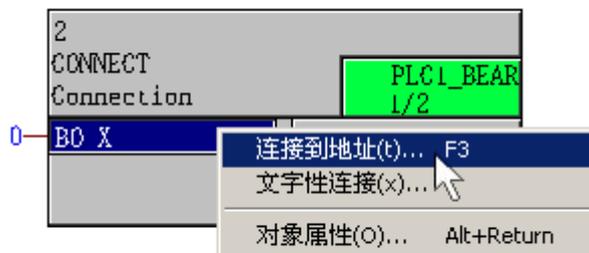


图2-11 连接输入信号。

- 在所显示的**选择左边界**对话框中选择**Measur. Superv Fuse Failure**。注意组，**显示文本** 和 **类型**三个列，这三列信息也可以在配置矩阵中找到。



注意：

选择左边界仅用于显示下面的信息：

- 在DIGSI的配置矩阵中配置给信息目标中的**CFC**的信息
- 并且可以和功能块的连接类型匹配的信息。

- 点击**确定**，确认选择。



图2—12 选择左边界对话框

在CFC功能图中就会显示新的连接。输出信号进入功能块的左边界，并用一根线与功能块相连。

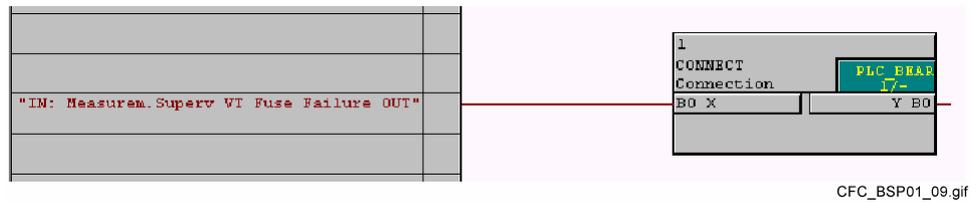


图2—13 快速PLC处理连接输入信号

2.5.5 连接输出信号

把**67 Direct. O/C >BLK 67/67-TOC** 和 **67 Direct.O/C >BLK 67N/67NTOC** 输出信号与**CONNECT**功能块相连接的步骤如下：

- 把**67 Direct. O/C >BLK 67/67-TOC**信息与功能块相连接，其方法为：
 - 右击**CONNECT**的输出Y并在所显示的关联菜单中选择**连接到地址...**。

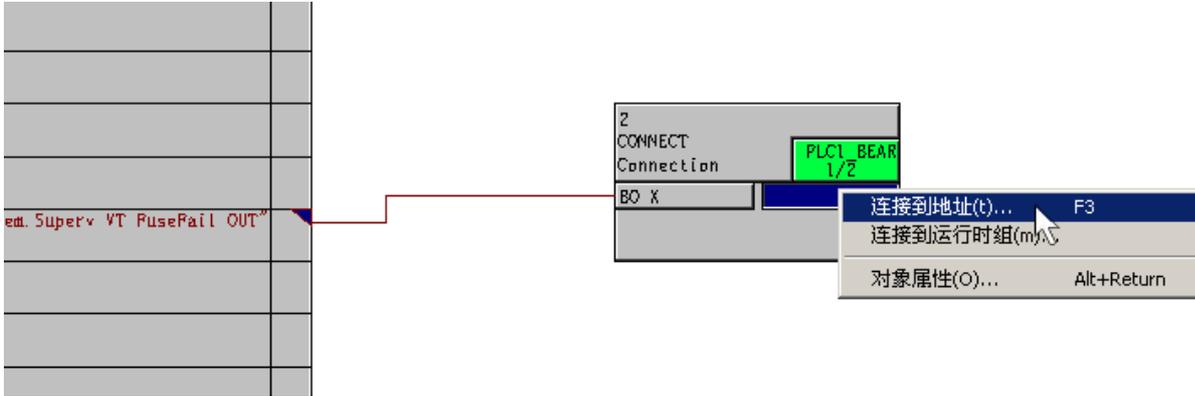


图2-14连接输出信号。



注意：

在任何情况下都不要使用**连接到运行时组...功能**。
DIGSI CFC不支持跨功能图连接。



注意：

为了确保系统的一致性，右边界的每一输出信号只能与一个功能块相连接。

- 在所显示的**选择右边界**对话框中选择**67 Direct. O/C >BLK 67/67-TOC**。注意**组**，**显示文本** 和 **类型**三个列中的信息，这三列信息也可以在配置矩阵中找到。

**注意：**

选择右边界 对话框仅用于显示以下信息：

- 在DIGSI配置矩阵中配置给信息来源中的CFC的信息
- 并且可以与功能块的类型相互匹配的信息。

- 点击OK，确认选择。



图2-15 选择右边界对话框。

在CFC功能图中就会显示新的连接。输出信号在右边界中输出，并用一根直线与功能块相连。

- 把**67 Direct. O/C >BLK67N/67NTOC**与功能块连接的步骤如下：
 - 右击**CONNECT**块的**Bo Y**输出并在所示的关联菜单中选择**连接到地址...**
 - 在所显示的对话框**选择右边界**选择**Direct.O/C >BLK 67N/67NTOC**。注意**组**，**显示文本**和**类型**三列。
 - 选择**确定**，进行确认。

在CFC功能图中就会显示新的联接。在右边界中输入信号，并用一根线与块相连接。

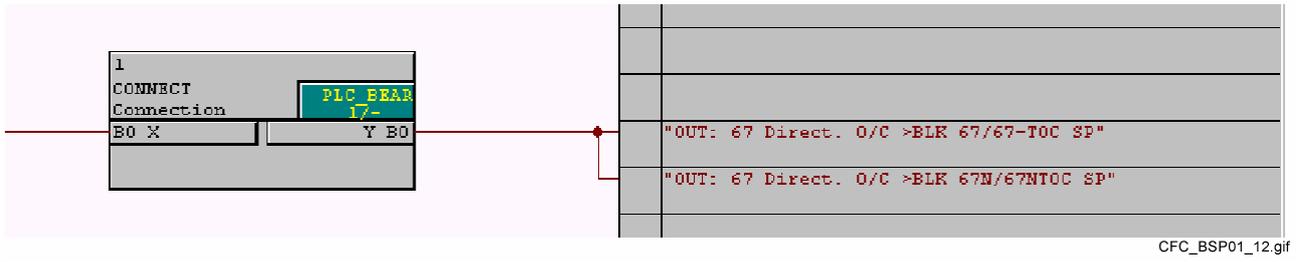


图2-16 连接输出信号：快速PLC

2.5.6 编译CFC功能图

为了使用创建的CFC功能图并在SIPROTEC装置中使用，必须对功能图进行编译，步骤如下：

- 左击功能图 → 编译 → 功能图作为程序....，所有功能图就会被编译。

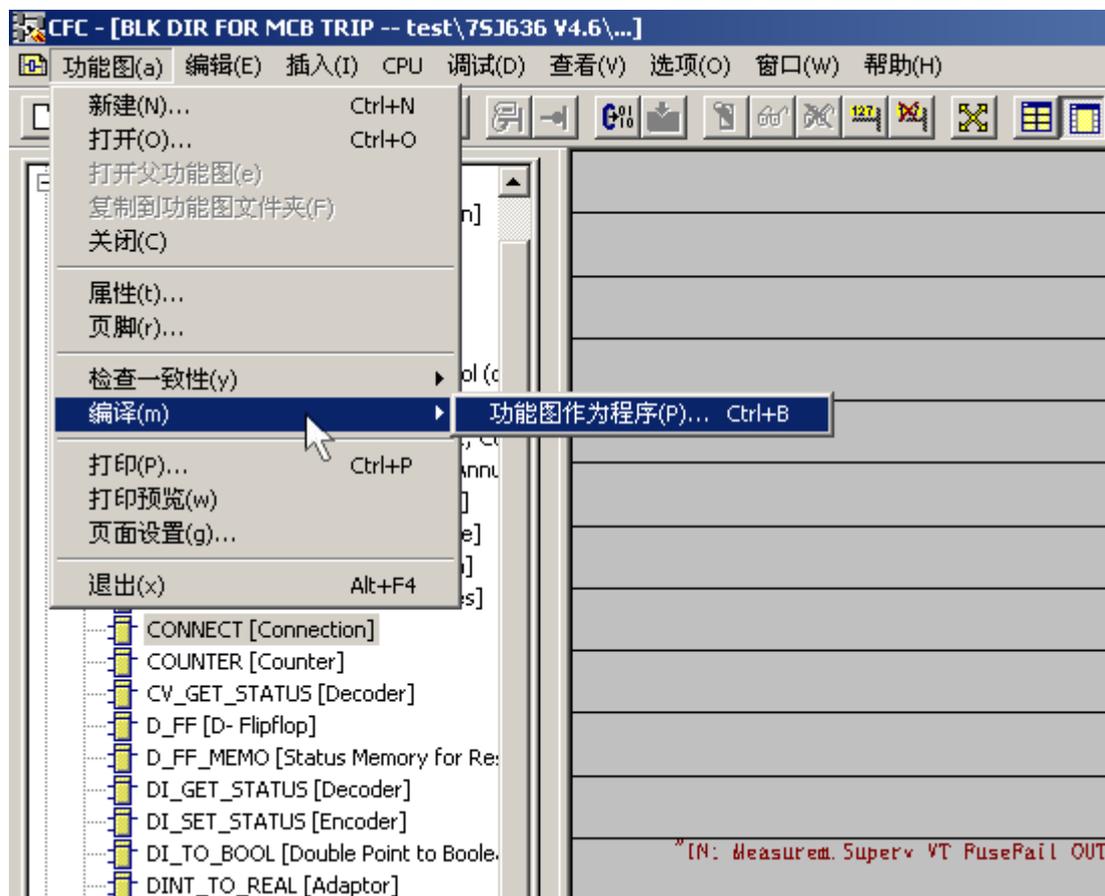


图2—17 编译CFC功能图

- 点击**确定**来确定所显示的信息。



图2—18 编译消息对话框

为了使用编译好的功能，必须在 DIGSI中保存参数并在SIPROTEC装置中重新载入。

2.6 慢速PLC处理实例

慢速PLC处理（优先级为**PLC1_BEARB**）具有较低的优先级。慢速PLC由于信号改变而执行。每秒也循环一次。



注意：

在下面的实例中选订货编号为（MLFB）**7SJ63655ER633HH3**型SIPROTEC装置。为了理解这个实例，必须从DIGSI装置列表中选择一种类似的装置。

任务

在SIPROTEC装置上用 一个指示灯来显示本地控制模式。

输入信号

- 组：**Cntrl Authority**（控制权限）
- 显示文本：**Cntrl Auth**

输出信号

- 组：**Cntrl Authority**
- 显示文本：**LocalCntrl**

在DIGSI配置矩阵中没有这个信息

CFC功能块

- DI_TO_BOOL**(双信号到布尔型)

操作过程

- 在DIGSI配置矩阵中插入**Cntrl Authority LocalCntrl** 信息。
- 在DIGSI配置矩阵中对一个指示灯配置**Cntrl Authority LocalCntrl**信息。
- 在DIGSI配置矩阵中对CFC配置输入信号和输出信号（详见第2.4节）。
- 创建一个新CFC功能图并起名为**DEVICE PANEL CTRL MODE**（详见第2.5.1节）
- 指定**PLC1_BEARB**的优先级（详见2.5.2）。
- 设置功能块**DI_TO_BOOL**（详见2.5.3）。
- 连接输入信号。（详见2.5.4）。
- 为了评估**Cntrl Authority Cntrl Auth**的双信号情况，配置**DI_TO_BOOL**的阈值**IS_OFF** 和**IS_ON**。
- 连接输出信号（详见2.5.5）。
- 编译CFC功能图（详见2.5.6）。

2.6.1 插入新信息

在DIGSI配置矩阵中插入新信息**Cntrl Authority LocalCntrl**的步骤如下：

- 在DIGSI中打开配置矩阵
 - 在导航窗口中选择**设置**
 - 在数据窗口中双击**Masking I/O**（配置矩阵）
- 显示**Cntrl Authority**
- 点击**插入** → **信息...**，就会显示**信息目录**对话框。



图2-19 插入新信息

- 在信息目录对话框中，点击**Annunciations\Single Point (SP)\ON/OFF (SP)** 信息类型，按住鼠标，并把信息类型拖动到**Cntrl Authority**上。

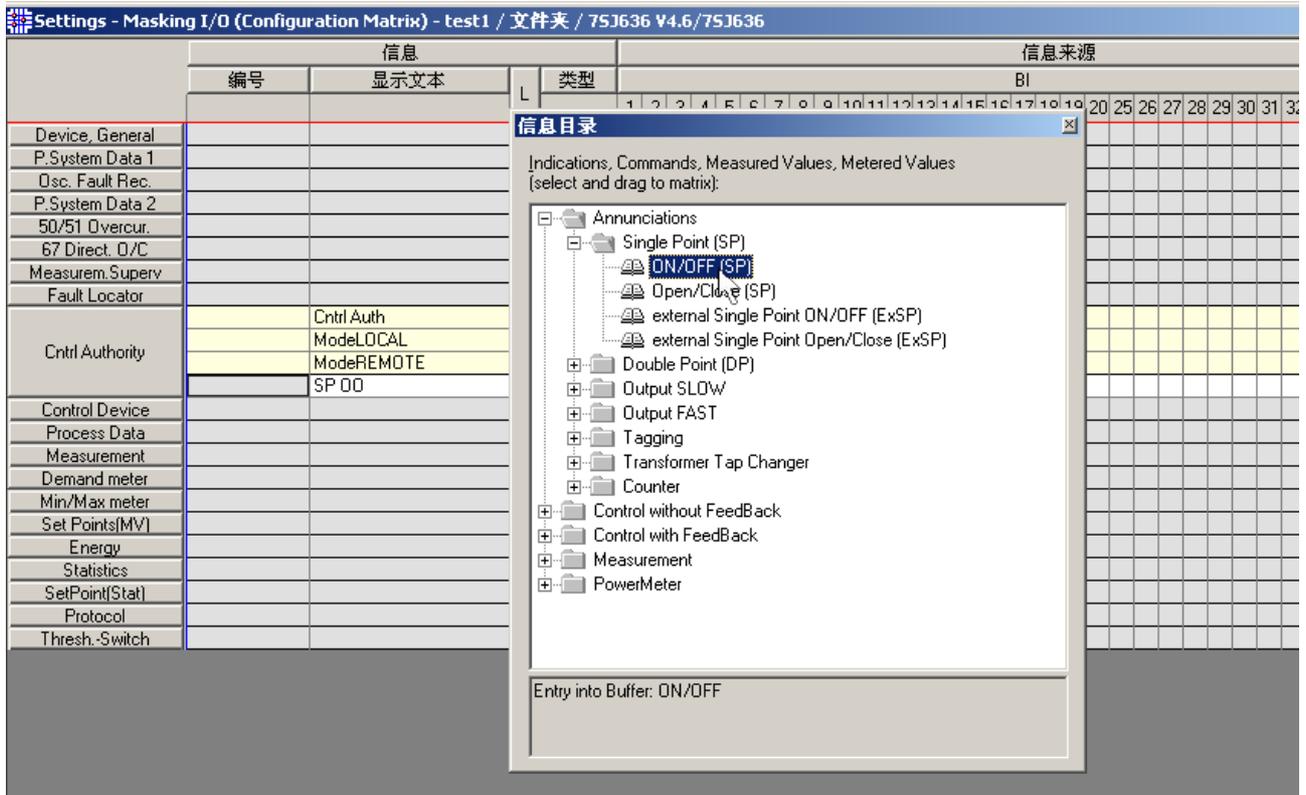


图2-20 选择信息类型

- 为了重新命名插入的信息，双击在**显示文本**栏中的名称，并输入 **LocalCntrl**。

Fault Locator				
Cntrl Authority	Cntrl Auth			DP
	ModeLOCAL			DP
	ModeREMOTE			IntSP
Control Device	LocalCntrl			SP

Bs_4_006a.gif

图2-21 重新命名信息

2.6.2 将信息配置到指示灯

在DIGSI配置矩阵中设定指示灯的**Cntrl Authority LocalCntrl**信息。

- 在DIGSI中打开配置矩阵，方法如下：
 - 在导航窗格中选择**设置**。
 - 在数据窗格中双击**Masking I/O** (配置矩阵)功能。
- 显示组**Cntrl Authority**并选择**LocalCntrl**信息。
- 在配置矩阵中的**信息目标**中的**指示灯**项中第7格单元格点击右键并在所显示的关联菜单中选择**U (不保持)**。

	信息				信息目标																				
	编号	显示文本	L	类型	LEDs														缓冲器			s	C	D	CM
					1	2	3	4	5	6	7	8	9	10	11	12	13	14	D	s	T	C	D	CM	
Device, General												*							*			*	*		
P.System Data 1																			*			*	*		
Osc. Fault Rec.																			*			*	*		
P.System Data 2						*													*	*	*	*	*		
50/51 Overcur.						*	*	*	*										*	*	*	*	*		
67 Direct. O/C						*	*	*	*										*	*	*	*	*		
Measurem.Superv										*									*		*	*	*		
Fault Locator																			*	*	*	*	*		
Cntrl Authority		Cntrl Auth		DP															00			×			
		ModeLOCAL		DP															00			×			
		ModeREMOTE		IntSP															00			×			
		LocalCntrl		SP															00						
Control Device																		*			*	*	*	*	*
Process Data																		*			*	*	*	*	*
Measurement																		*			*	*	*	*	*
Demand meter																		*			*	*	*	*	*
Min/Max meter																		*			*	*	*	*	*
Set Points(MV)																		*			*	*	*	*	*
Energy																		*			*	*	*	*	*
Statistics																		*			*	*	*	*	*
SetPoint(Stat)																		*			*	*	*	*	*
Protocol																		*			*	*	*	*	*
Thresh.-Switch																		*			*	*	*	*	*

图2-22 配置指示灯信息

2.6.3 连接输入信号

把输入信号**Cntrl Authority Cntrl Auth**连接到**DI_TO_BOOL**功能块，其过程与第2.5.4节描述的内容类似。

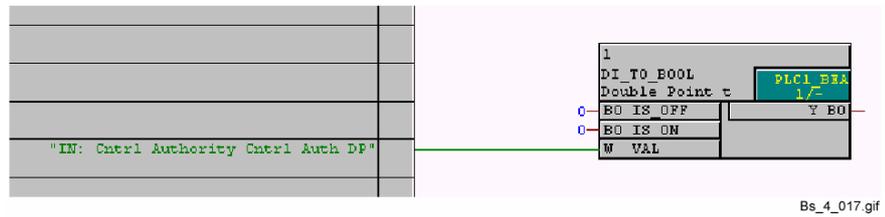


图2—23 慢速PLC处理连接输入信号的实例

2.6.4 设定功能块

设置 **Cntrl Authority Cntrl Auth DP** 的阈值 **IS_OFF** 和 **IS_ON** 参数的步骤如下：

- 右击功能块并在所显示的关联菜单中选择**对象属性...**。
- 点击**I/Os**标签并在栏Value中的**IS_OFF**中输入1。

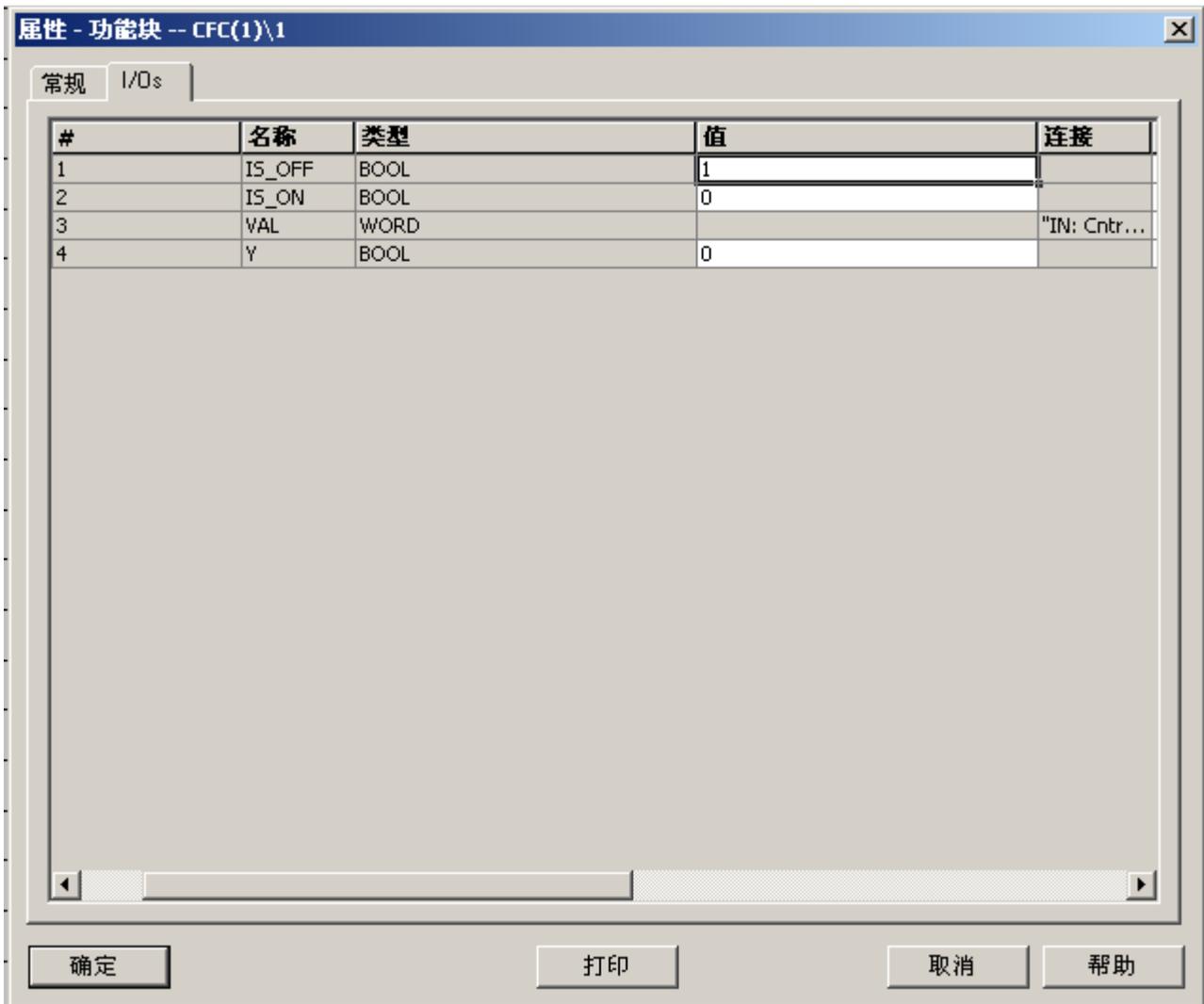


图2—24确定功能块的参数

- 点击**确定**，进行确认。

2.6.5 连接输出信号

把输出信号**Cntrl Authority LocalCntrl**与**DI_TO_BOOL**功能块相连接的过程与2.5.5所述内容一致。

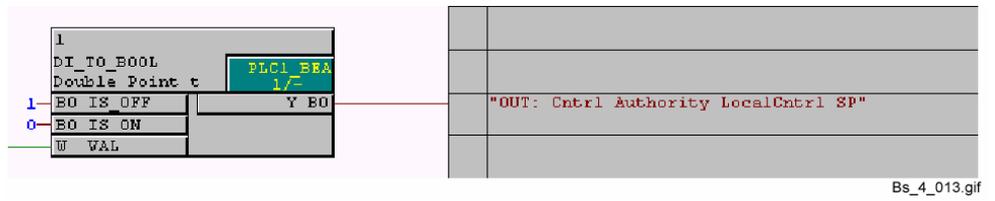


图2—25 慢速PLC处理连接输出信号实例

2.7 测量值处理实例

测量值处理（优先级为**MW_BEARB**）的优先级为中等。处理过程是周期性的。



注意:

下面的实例中选用了订货编号为（MLFB） **77SJ63655ER633HH3** 型 SIPROTEC装置。为了理解这个实例，必须从DIGSI装置目录中选择一种类似的装置。

任务

传感器的输入信号仅仅用于提供线性范围的测量电压。

如果测量电压位于线形范围以外，或者低于SIPROTEC装置中设定的范围时，就输出信息。

输入信号

组: **Measurement**（测量）

显示文本: **Td1=**

组: **Set Points (MV)**（整定值）

显示文本: **Press<**

输出信号

组: **Measurement**（测量）

显示文本: **Superv. Pressure**

组: **Set Points (MV)**（整定值）

显示文本: **SP. Pressure<**

CFC功能块

LIVE_ZERO (零漂)

LOWER_SETPOINT (下限)

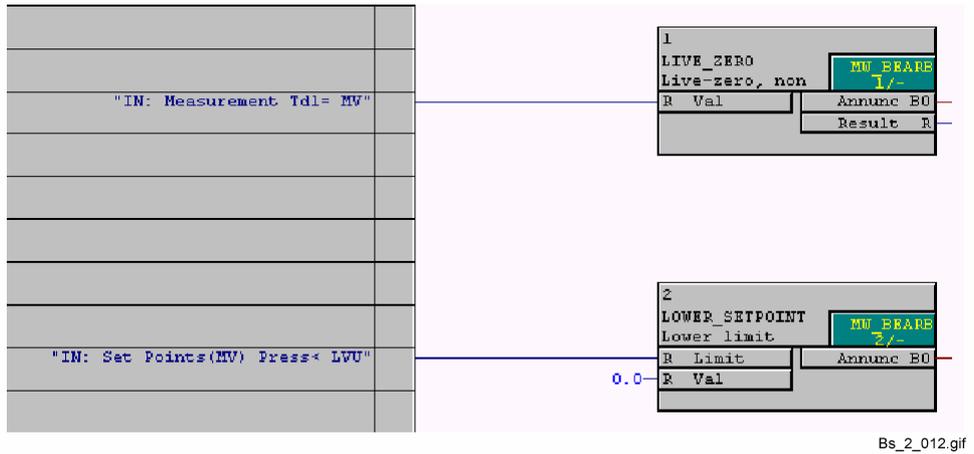
操作过程

- 在DIGSI配置矩阵中，对CFC配置输入信号和输出信号，详见第2.4节。
- 创建一个新的CFC功能图并命名为**MES VAL PROCESS ING**（详见第2.5.1节）。
- 指定**MW_BEARB**的优先级（详见第2.5.2节）。
- 设定**LIVE_ZERO** 和 **LOWER_SETPOINT**功能块（详见第2.5.3节）。
- 连接输入信号（详见2.5.4节）。
- 为测量值 **measured values Td1=** 设置LiveZero的阈值 **DetecKnee** , **DispKnee**。
- 连接**LIVE_ZERO** 和 **LOWER_SETPOINT**功能块。

- 连接输出信号（详见2.5.5）。
- 编译CFC功能图（详见2.5.6）。

2.7.1 连接输入信号

把输入信号 **Measurement Td1=** ， **Set Points Press<**与 **LIVE_ZERO** 和 **LOWER_SETPOINT**功能块连接起来的过程与2.5.4节的内容相似。



Bs_2_012.gif

图2—26 测量值处理连接输入信号的示例

2.7.2 连接功能块

把LIVE_ZERO 和LOWER_SETPOINT功能块相互连接的过程如下：

- 点击LIVE_ZERO的Result端口，保持鼠标为按下状态并把光标拖动到LOWER_SETPOINT功能块的Val端口上。

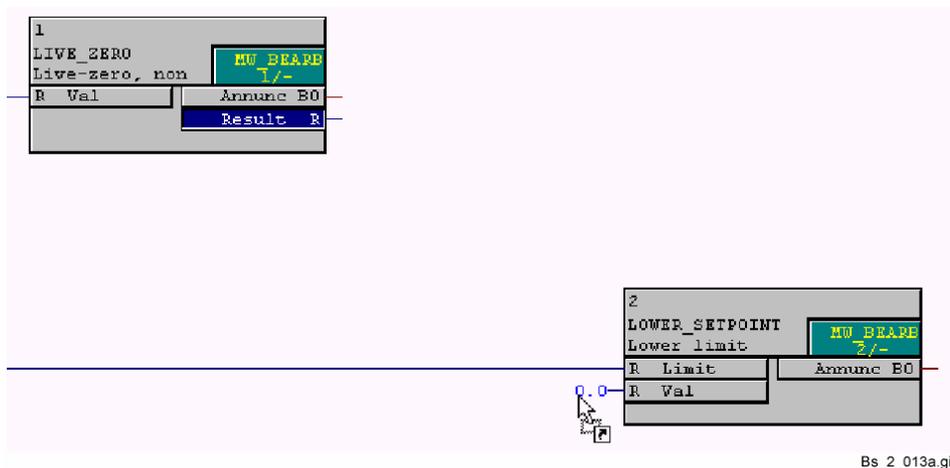


图2-27 测量值处理中功能块连接的示例

- 释放鼠标，在两个I/Os之间就显示了一个连接。

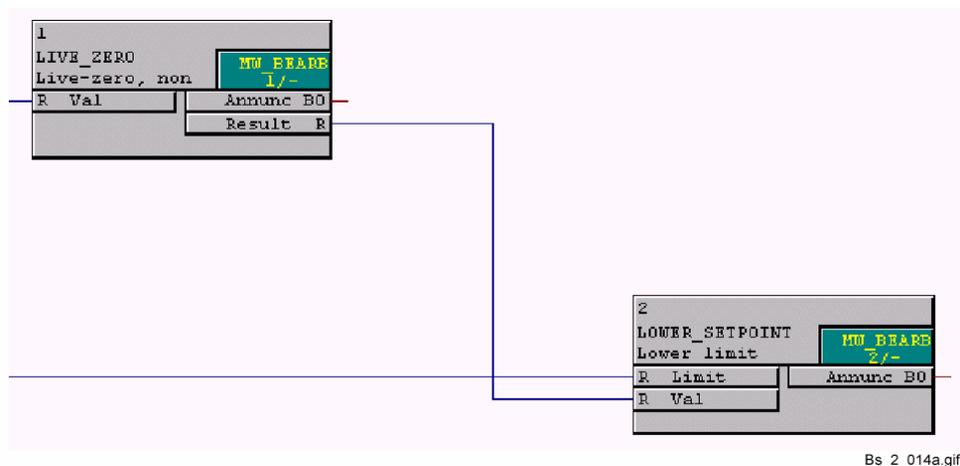


图2-28 已连接的功能块



注意：

只有当两个I/O的数据类型一致时，它们才能连接起来。

2.8 联锁实例

在发出命令时，需要用到联锁（优先级为**SFS_BEARB**）。此外，周期性处理是在后台进行的。



注意：

在下面的实例中选用订货编号为（MLFB） **7SJ63655ER633HH3** 型 SIPROTEC装置。为了理解这个实例，必须从DIGSI装置列表中选择一种类似的装置。

当创建SIPROTEC装置时，默认CFC功能图也被同时插入了。CFC的联锁已经使用了下面实例中的输出信号。为了完成这个实例，必须首先删除CFC功能图联锁。

任务

检查接地开关的开和关的联锁条件，并产生信号。

输入信号

组：Control Device（控制设备）

显示文本：52Breaker

长文本：52 Breaker

组：Control Device（控制设备）

显示文本：Disc.Swit.

长文本：Disconnect Switch

组：Control Device（控制设备）

显示文本：GndSwit.

长文本：Ground Switch

组：Process Data（控制设备）

显示文本：>DoorClose

长文本：>Door closed

输出信号

组：Control Device（控制设备）

显示文本：GndSw. Cl.

长文本：Interlocking: Ground switch Close

组：Control Device（控制设备）

显示文本：GndSw. Open

长文本：Interlocking: Ground switch open

CFC功能块

DM_DECODE (双精度解码信号)

X_OR (“异或”门)

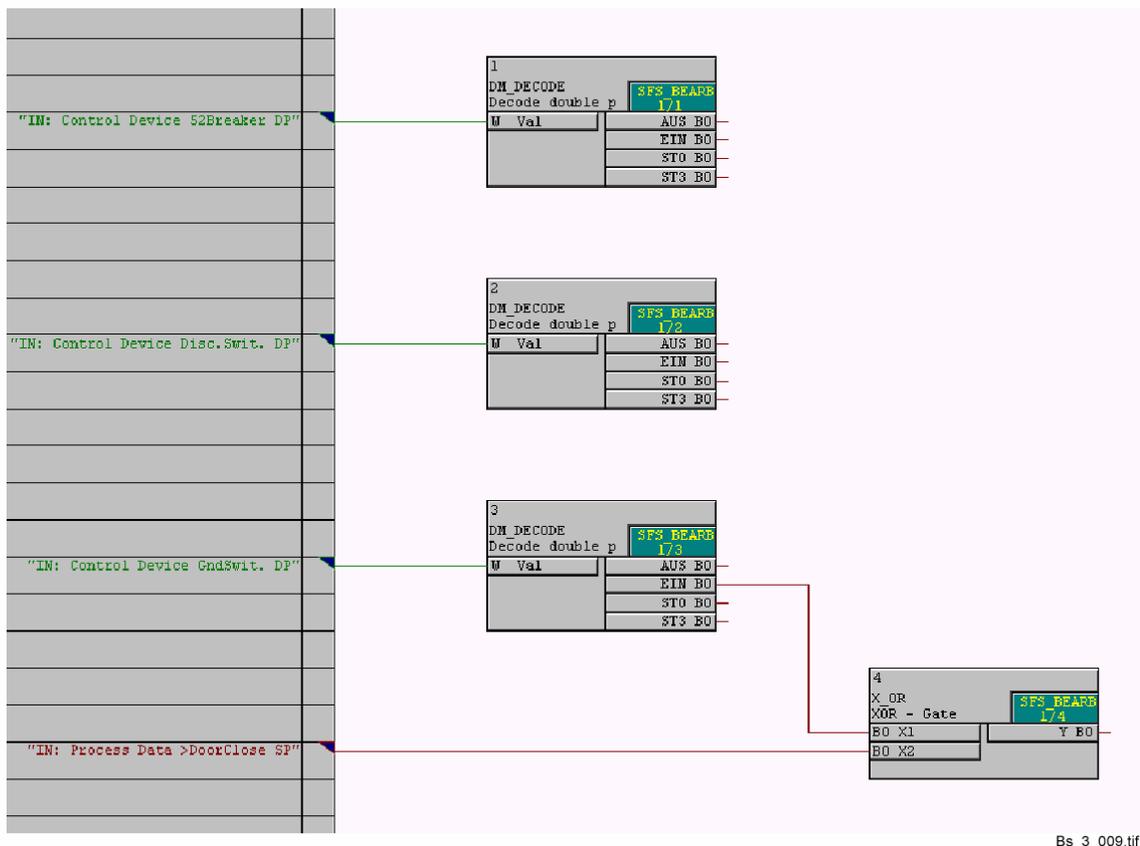
AND (“与”门)

操作过程

- 在DIGSI配置矩阵中，为CFC配置输入和输出信号（详见第2.4节）。
- 创建一个新的CFC功能图并命名为**INTERL GND SWITCH**（详见2.5.1）。
- 指定**SFS_BEARB**优先级（详见第2.5.2节）。
- 设置**DM_DECODE**, **X_OR** 和**AND**功能块（详见第2.5.3节）。
- 连接输入信号（详见第2.5.4节）。
- 把**AND**功能块的输入数目增加到4个。
- 连接**DM_DECODE**, **X_OR** 和**AND**功能块（详见2.7.2节）。
- 连接输出信号（详见2.5.5节）
- 编译CFC功能图（详见2.5.6）

2.8.1 连接输入信号

将输入信号**Control Device 52Breaker DP**， **Control Device Dis.Swit. DP**， **Control Device GndSwit.**， **Process indications >DoorClose SP** 与功能块**DM_DECODE**, **X_OR**， **AND**连接过程与第2.5.4节所述内容相似。



Bs_3_009.tif

图2-30 联锁处理中连接输入信号的示例

2.8.2 增加输入信号的数目

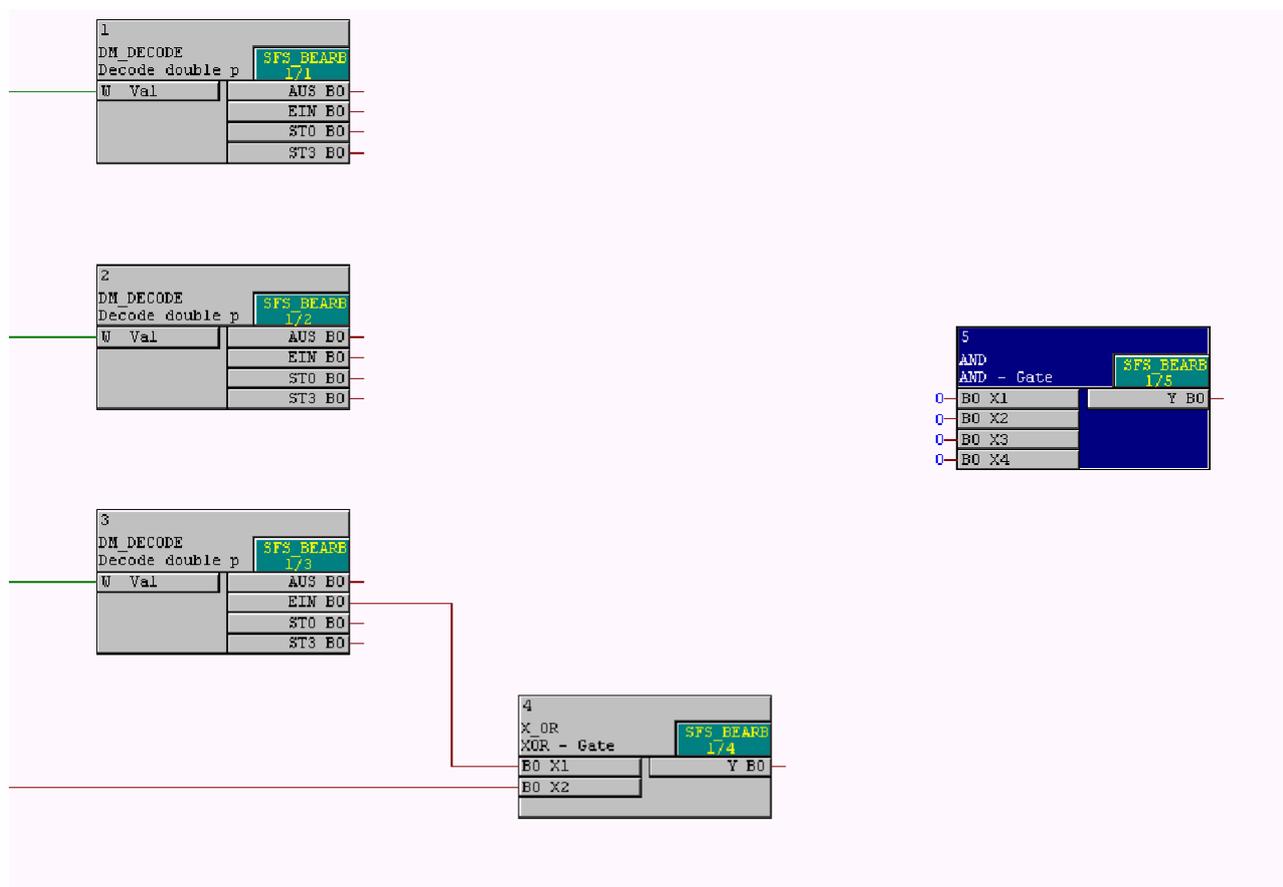
把AND功能块的输入信号数增加为4个的步骤为:

- 右键点击功能块，并在所示的关联菜单中选择I/O数量...
- 在窗口I/O数量中的Number选项内输入4，点击OK确认。



图2-31 I/O数量对话框

更改后，功能块将显示四个输入端。

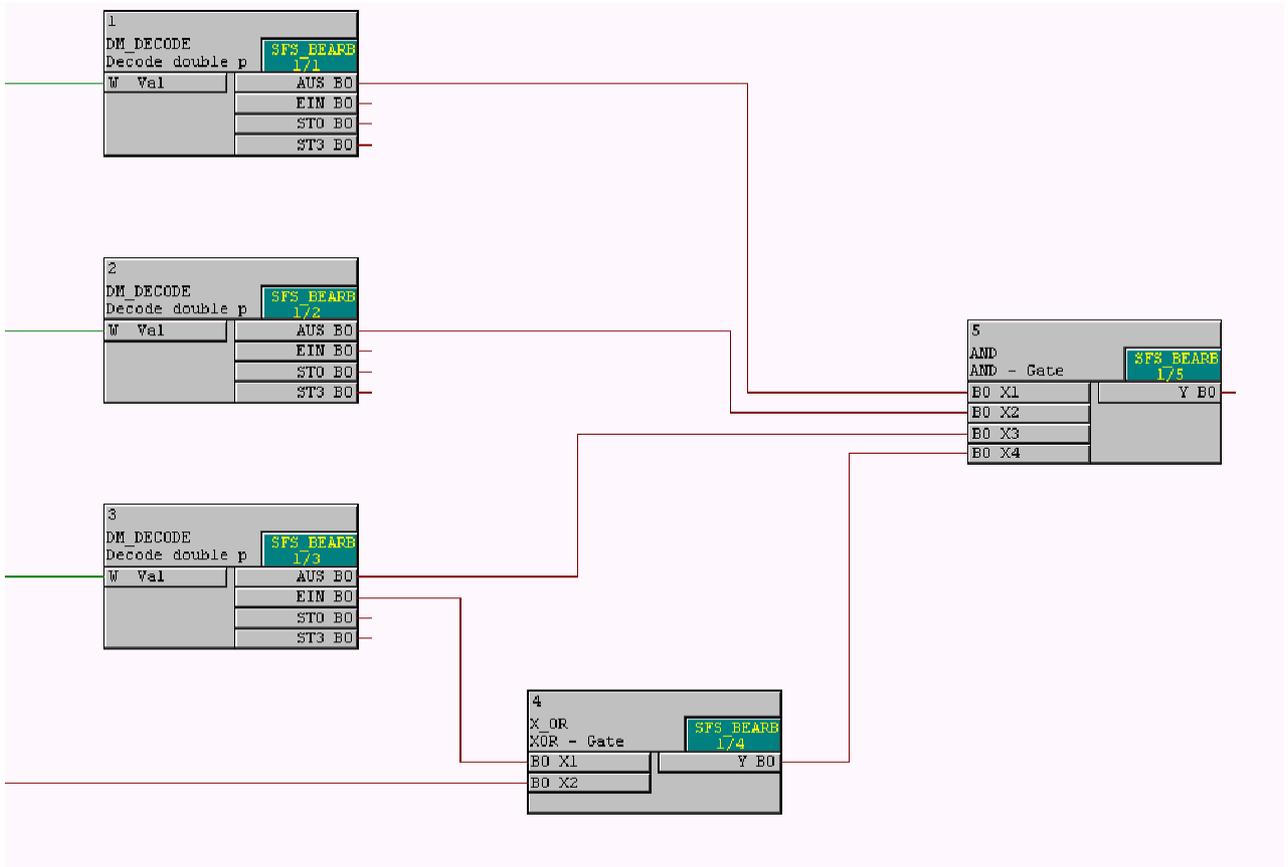


Bs_3_011.tif

图2-32 增加AND功能块的输入信号数目

2.8.3 连接功能块

把功能块DM_DECODE, X_OR 和AND连接起来的过程与2.7.2的内容相一致。



Bs_3_012.tif

图2-33 联锁处理中连接的功能块示例

2.8.4 连接输出信号

Control Device GndSw Cl.和Control Device GndSw Open与AND功能块连接过程如第2.5.5所示。

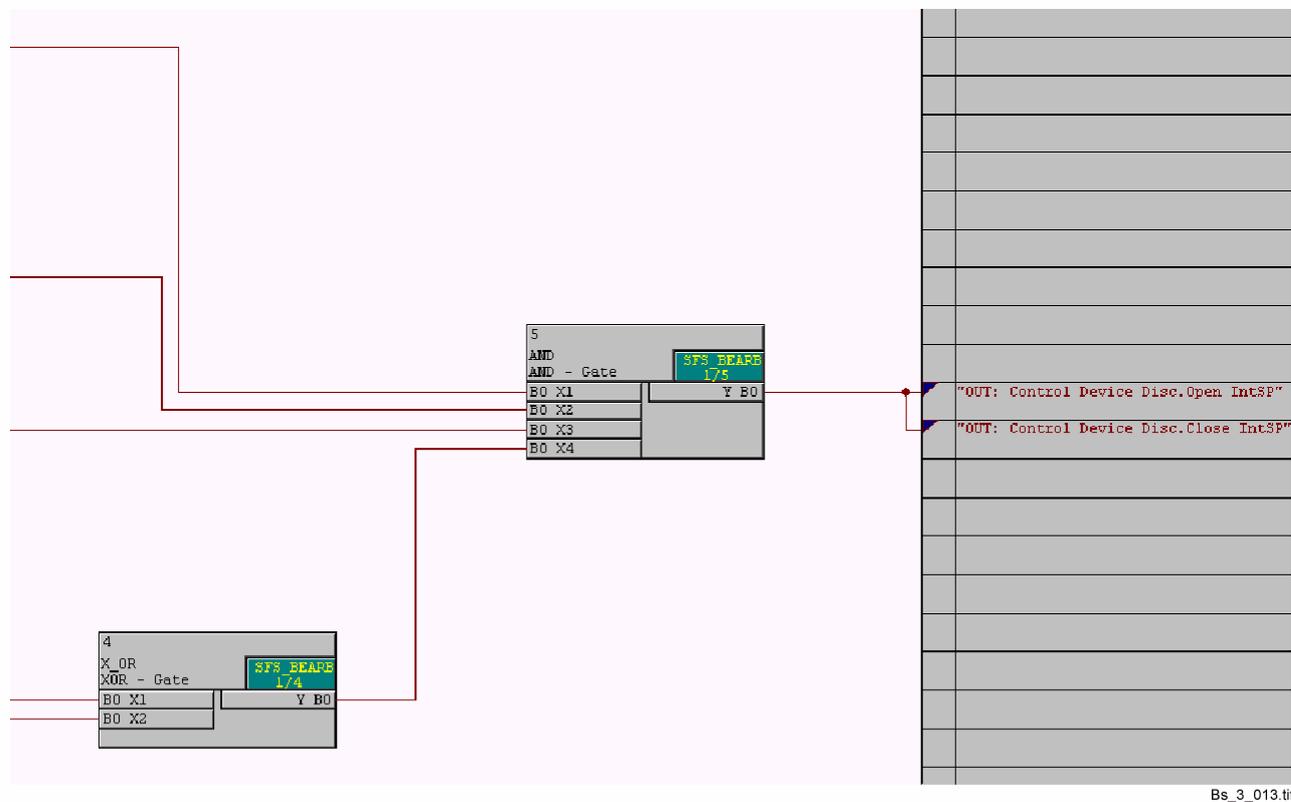


图2-34 联锁处理中连接输出信号示例

应用实例

概述

本章的主要内容是来自实际应用的一些典型方案。



注意：

本章的内容是假定读者已经对DIGSI V4或者DIGSI CFC的操作比较熟悉为前提的。

关于DIGSI V4或者DIGSI CFC操作的基本内容，读者可以在本手册的第2章中进行了解。



注意：

本章中的CFC功能块是以“宽的”模式显示的，具体步骤如下：

- 在CFC编辑器中，点击**选项** → **自定义** → **块/边框宽度**。
- 在**块宽度**的选择项下点击**宽的**选择项，然后点击**确定键**，进行确认。

目录

3.1 定值组切换功能	45
3.2 闪烁的指示灯	57
3.3 反向联锁	62
3.4 操作计数	65

3.1 定值组切换功能

任务

下面是我们将要完成的任务：

- ❑ 将一个电机连接到开关量输出 **BO1**。电机通过功能键 **F1** 直接启动。
- ❑ 当所有相电流均低于额定值5%以下，且在电机启动后的10秒内时，**B**组定值将被投入。
- ❑ 回到正常情况后，**A**组定值将被投入。

通过F1启动电机

将电机通过功能键**F1**启动，具体步骤如下：

- 在DIGSI的配置矩阵中生成一个组，名称为**Start Motor**。
- 在名称为**Start Motor**的组插入一个新的类型为**Tagging Internal On/Off**的信息，这条信息在插入的时候保留系统提供的默认名称。
- 将这条信息的名称重新命名为**F1 Start**。
- 将**F1 Start**的信息配置给功能键**1**。
- 将**F1 Start**的信息配置给开关量输出**BO1**。信息输出是在**不保持**的状态下完成的。

启用定值组切换功能

启用定值组切换功能，需要完成下面的工作：

- 双击**设置**下的**设备配置**打开**功能范围**将定值组切换功能设置为**启用**

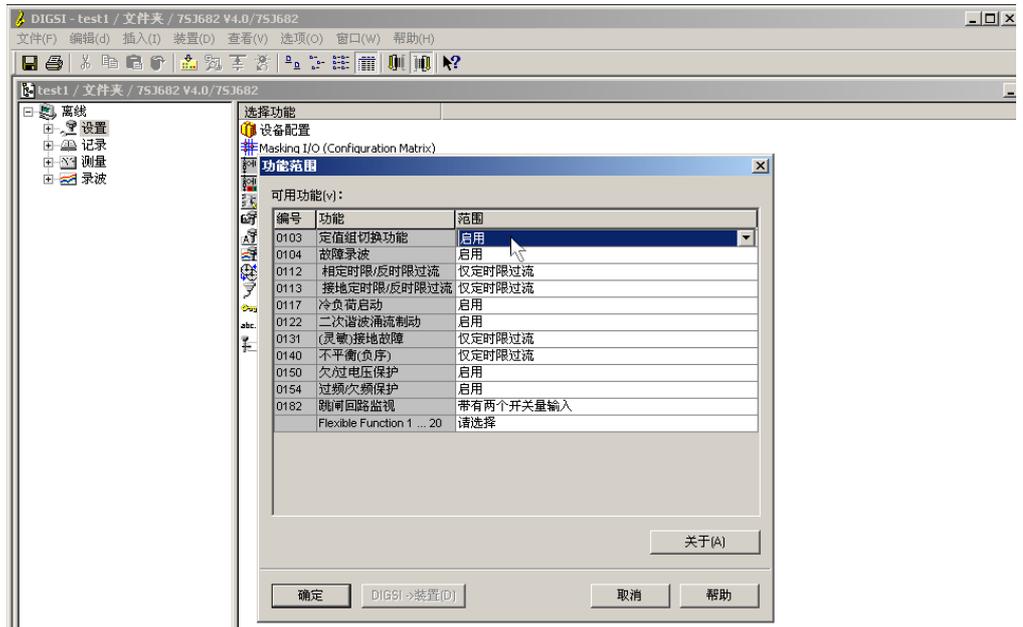


图3-1 启用定值组切换功能

动态切换定值组

为了能够通过CFC逻辑运算动态地切换定值组，应该通过开关量输入的方式激活定值组切换选项。

- 在定值组切换对象中，将数值的类型定义为开关量输入。

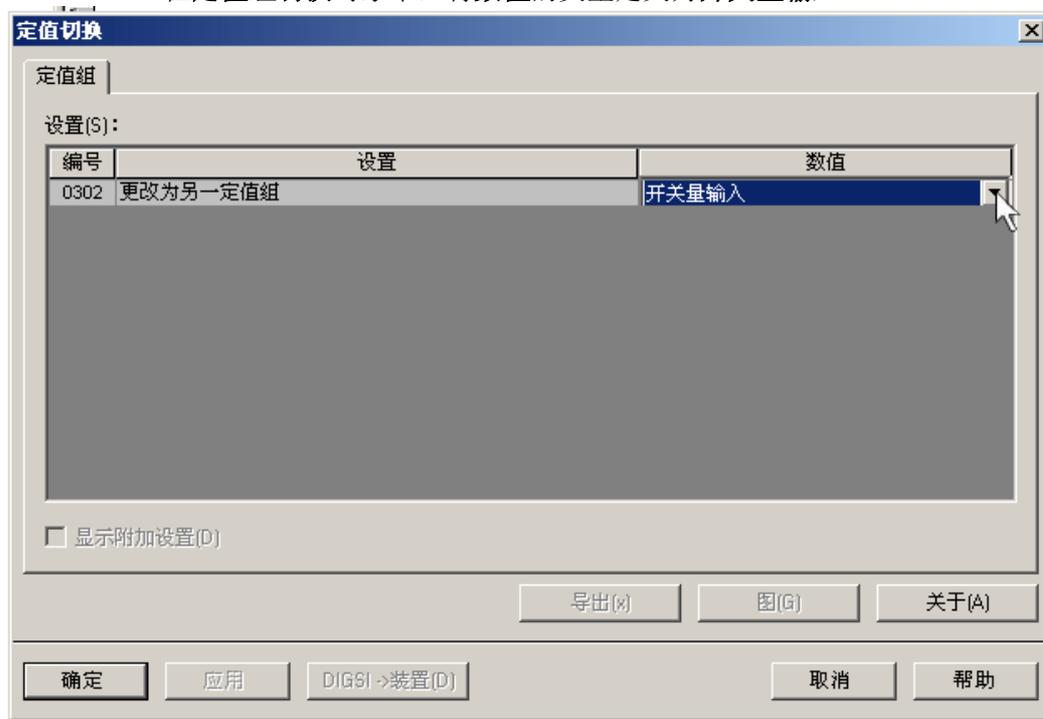


图3-2 动态切换定值组

切换定值组功能

当切换定值组功能被启用之后，将有一个名字为**定值切换**的新设置组显示在DIGSI配置矩阵中。

信息	编号	显示文本	L	类型	信息来源											信息目标																						
					BI											开关量输出							LEDs				缓冲器		C		D		CM					
					1	2	3	4	5	6	7	8	9	10	11	1	2	3	4	5	6	7	1	2	3	4	5	6	7	Q	s	T	C	D	CM			
设备																*																				*		
电力系统数据1																*																						
故障录波																																						
定值切换	00007	>定值组选择位0		SP																																		
	00008	>定值组选择位1		SP																																		
		A组定值		IntSP																								00										
		B组定值		IntSP																								00										
		C组定值		IntSP																								00										
	D组定值		IntSP																								00											
电力系统数据2					*	*									*			*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*			
过电流																																						
冷负荷启动																																						
过电压/欠电压																																					*	
非对称负荷(顺序)																												*	*	*	*	*	*	*	*	*		
频率保护															*												*	*	*	*	*	*	*	*	*	*		

图3—3 在DIGSI配置矩阵中设置切换定值组功能

指定有效的定值组

有效定值组的设置（如图3—4中的A、B、C、D所示）在组**定值切换**中进行。设置方式为通过两条信息完成，分别是**>定值组选择位1**（图3-4中的P1）以及**>定值组选择位2**（图3-4中的P2）：

当**>定值组选择位1**为二进制数字**1**，并且**>定值组选择位2**为二进制数字**0**的时候，B组定值就为有效组。

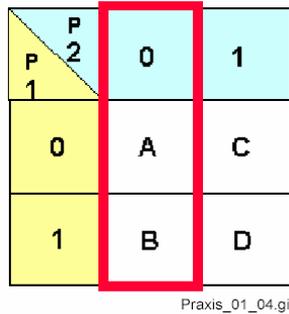


图3—4 有效的定值组

切换有效定值组

为了能够在A组定值和B组定值之间切换，只需要通过一个CFC程序将二进制的信息>定值组选择位1（图3-4中的P1）进行定义即可：

- 将>定值组选择位1作为输入信息配置给CFC



图3—5 将>定值组选择位1在信息目标中配置给CFC

切换定值组所需的的不同优先级

□ 当所有的相电流低于额定值的5%，或者在电机启动后的10秒钟之内，CFC程序将切换到B组定值。

□ 相电流是以CFC优先级MW_BEARB（测量值处理）进行监测的。

为了在10秒后切换定值组，需要使用计时器。但是这种计时器（通用计时器）不能在CFC优先级MW_BEARB（测量值处理）中工作。为了计时器正常工作，应该使用CFC优先级PLC1_BEARB（慢速PLC处理）。

因为需要使用不同的优先级，我们将CFC程序分割成两个CFC功能图，分别为：

- CFC功能图ParaUmsch:
优先级MW_BEARB（测量值处理）
其功能是监视相电流
- CFC功能图ParaUmsch1:
优先级PLC1_BEARB（慢速PLC）
其功能是监视启动时间和切换定值组。

不同优先级之间的通信

不同的优先级，比如MW_BEARB 和PLC1_BEARB之间可以通过信息项进行相互通信，但是前提是Tagging Internal On/Off应该满足下面的条件：
 在一个优先级，例如CFC功能图ParaUmsch中的MW_BEARB 等级中的一条信息被赋予一个数值，然后在另外一个优先级，例如CFC功能图ParaUmsch1中的PLC1_BEARB等级中对该条信息进行评估。



注意：

同一功能图中不能出现不同优先级的功能块，应该为不同的优先级的功能块使用不同的CFC功能图。

在一个CFC功能图内部，Tagging Internal On/Off类型的信息既可以用在功能图的左边界又可以用在右边界。

使用信息进行通信

为了能够使用信息在优先级MW_BEARB 和 PLC1_BEARB之间进行通信，应该完成下面的工作：

- 在名称为Grp Chge OPTION 的组中插入一个新的类型为Tagging Internal On/Off的信息，这条信息在插入的时候保留系统提供的默认名称。
- 将这条信息的名称重新命名为I<5%。
- 将I<5%的信息在信息来源和信息目标中配置给CFC。

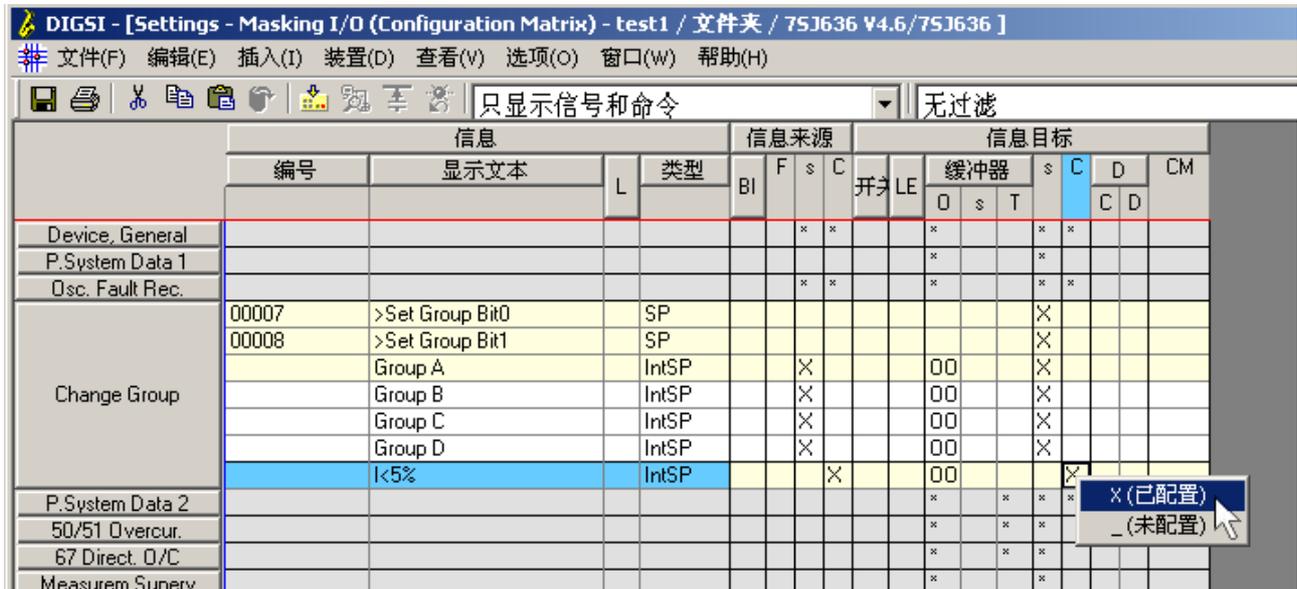


图3—6 将信息在信息来源和信息目标中配置给CFC

通过CFC程序监视相电流

为了能够通过CFC程序监视相电流，应该完成下面的工作：

- 将相电流在信息目标中配置给CFC。

	信息			信息来源			信息目标				
	编号	显示文本	L 类型	BI	s	C	测量值	测量值窗口	s C	计量值窗口	D
Device, General											
P.System Data 1											
Osc. Fault Rec.											
Change Group											
P.System Data 2											
50/51 Overcur.											
67 Direct. O/C											
Measuram.Superv											
Fault Locator											
Cntrl.Authority											
Control Device											
Process Data											
Measurement	00601	Ia =	MV						x	x	
	00602	Ib =	MV						x	x	x
	00603	Ic =	MV						x	x	
	00604	In =	MV						x	x	
	00605	I1 =	MV						x	x	
	00606	I2 =	MV						x	x	
	00831	Ilo =	MV						x	x	
	00621	Va =	MV						x	x	
	00622	Vb =	MV						x	x	
	00623	Vc =	MV						x	x	
	00624	Va-b=	MV						x	x	
	00625	Vb-c=	MV						x	x	x
	00626	Vc-a=	MV						x	x	
	00627	VN =	MV						x	x	
	00629	V1 =	MV						x	x	
	00630	V2 =	MV						x	x	
	00641	P =	MV						x	x	
00642	Q =	MV						x	x		
00645	S =	MV						x	x		
00644	Freq=	MV						x	x		
00832	Vo =	MV						x	x		

图3—7 将相电流在信息目标中配置给CFC

- 在CFC中创建一个新的名称为**ParaUmsch**的功能图，然后打开这个新功能图。
- 使用**编辑** → **运行序列**菜单将功能图的优先级设置为**MW_BEARB**。以后每一个新功能块都要插入这个优先级**MW_BEARB**之中。
- 对于相电流的监视是通过**LOWER_SETPOINT**（下限）功能块功能而进行的。所有的三个监视功能通过**AND**（“与”门）功能块进行连接，从而生成**I<5%**的信息，可以用于和下一个优先级**PLC1_BEARB**进行通信。
- 设置CFC功能块。



注意：

当插入CFC功能块时，应该按照系统提供的默认运行顺序进行运行。如果有必要，可以通过菜单**编辑** → **运行序列**对默认的运行顺序进行更改。

- 使用名称为**对象属性**的关联菜单将**LOWER_SETPOINT**功能块的参数设置为**5.0%**的限度。

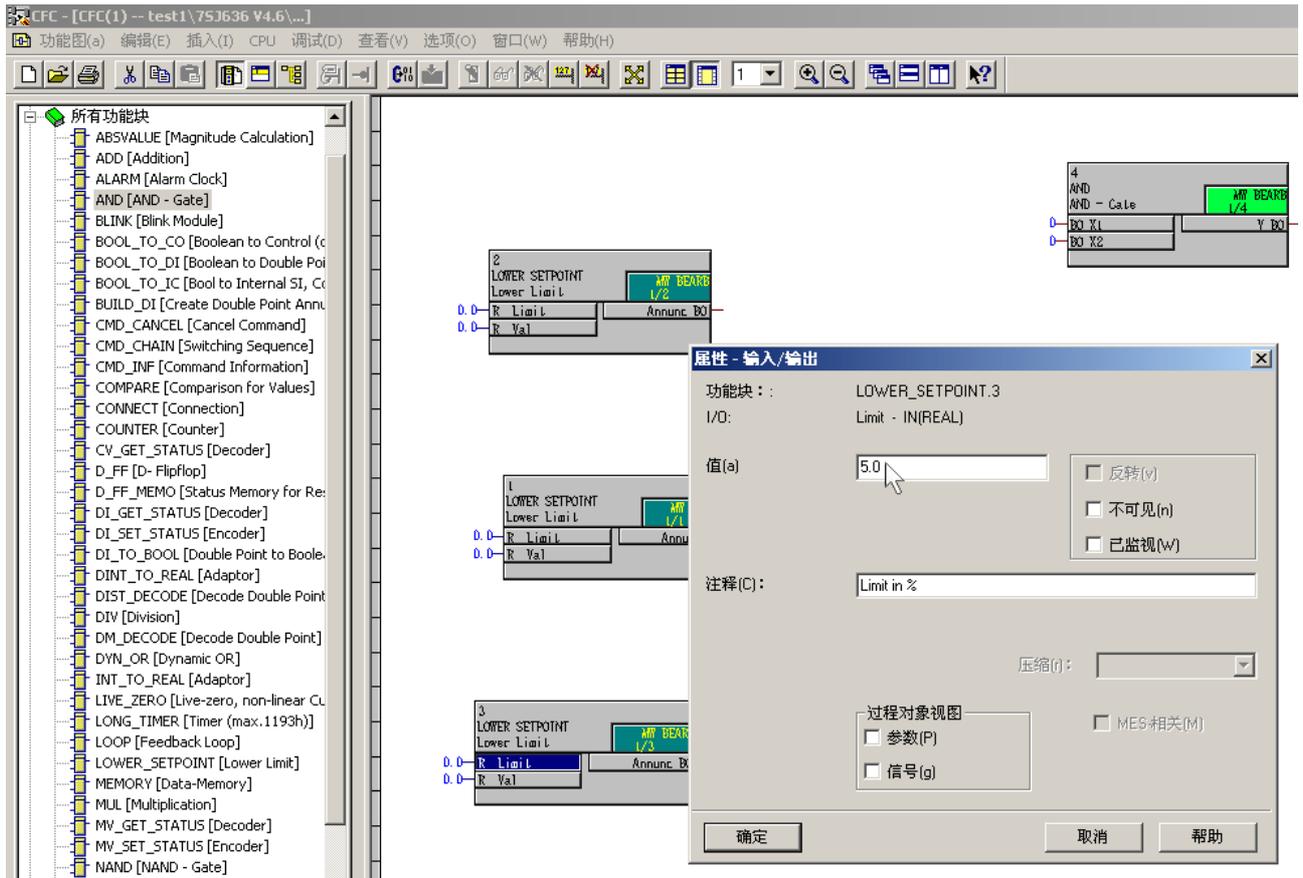


图3—8 设置极限值5.0%

- 右击**AND**功能块，在关联菜单中选择**I/O数量**将**AND**功能块中的**I/O**数量增加到3个。
- 将功能块相互连接，并且和功能图边界上的信号进行连接。在上述过程中，将**AND**功能块的输出数据和功能图右边界上面名为**I<5%**的信息进行连接。

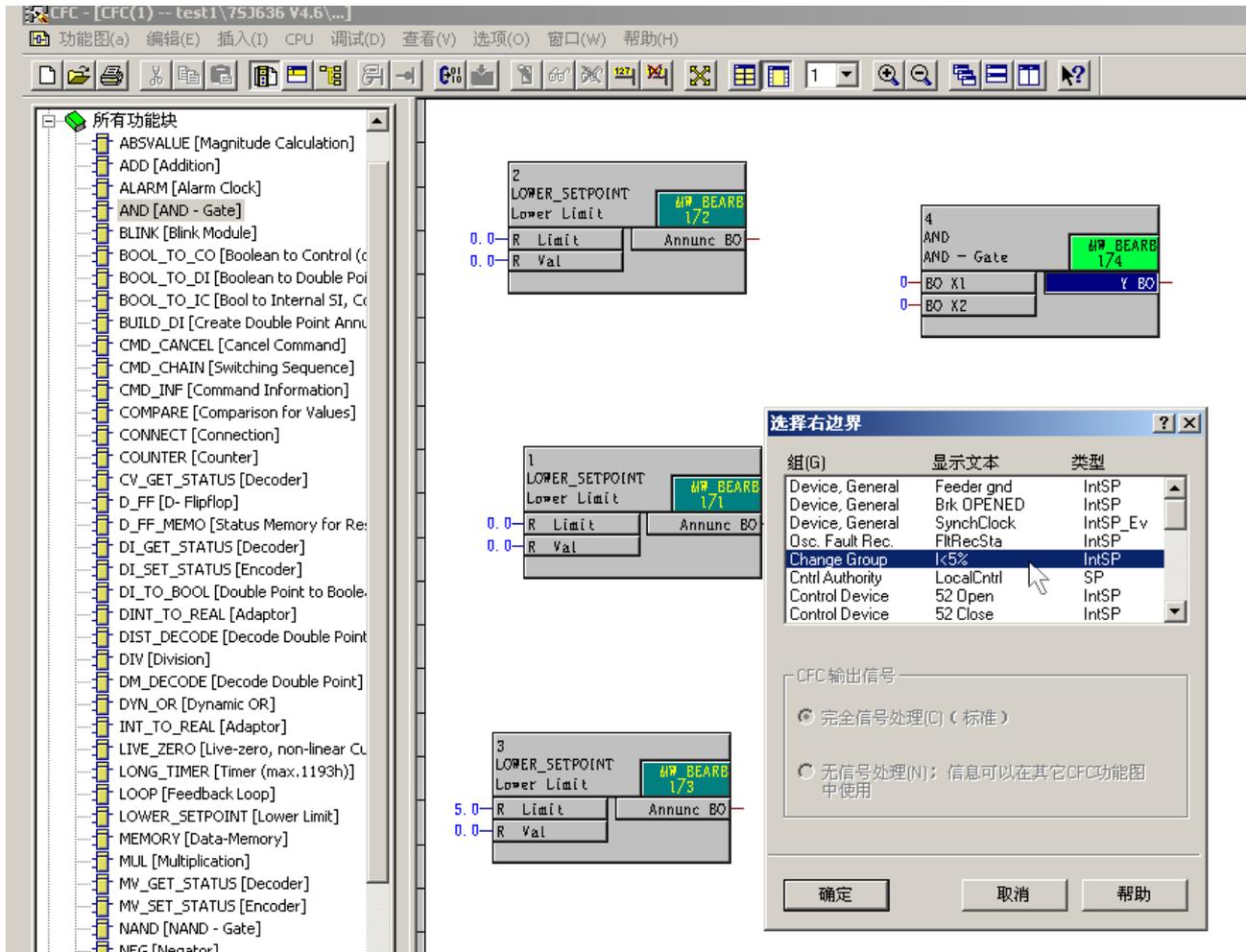


图3—9 连接功能块与功能图边界

编译CFC功能图

用户在生成CFC功能图之后，可以通过菜单**功能图** → **编译** → **功能图**作为程序对CFC功能图进行编译。

**注意：**

在上面的例子中，使用了**TIMER**（通用计时器）来监视电机的启动时间。根据SIPROTEC装置的不同版本，用户在监视的时候也可以使用**TIMER_SHORT**功能块（简易计时器）或者**LONG_TIMER**功能块（计时器（最长计时时间为1193小时））。

LONG_TIMER功能块（计时器（最长计时时间为1193小时））也可以在CFC优先级**MW_BEARB**（测量值处理）上进行工作。在使用这个功能块的时候，用户可以在**MW_BEARB**（测量值处理）的优先级上整体完成定值组切换的工作，没有必要将定值组切换工作分至不同优先级。。

- 在**TIMER**功能块中，使用关联菜单中的**对象属性**将**T1x1ms**的参数设置为**10000 ms**。

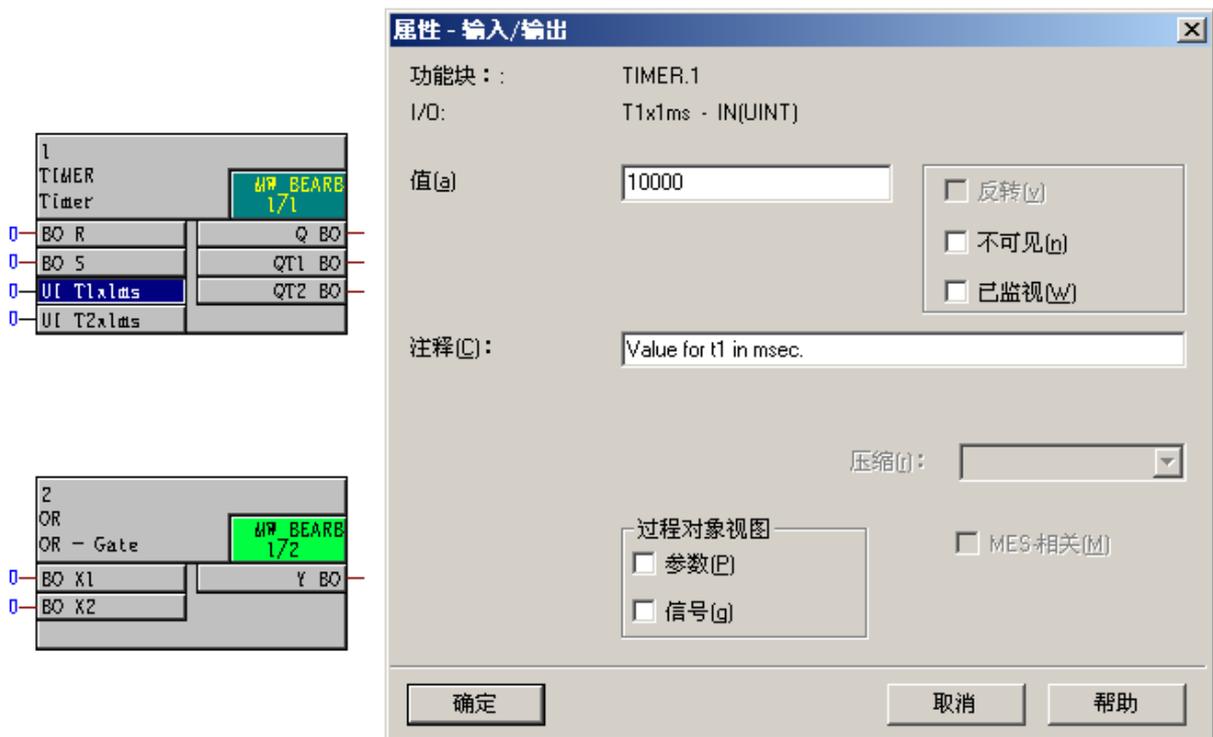


图3—11 设置Timer功能块属性

- 将功能块相互连接，并且和功能图边界上的操作对象进行连接。在上述过程之中，将OR功能块的输入端和功能图左边界上面的名称为I<5%的信息进行连接。同时要将OR功能块的输出端和功能图右边界上面的名称为>Set Group Bit1 (>定值组选择位1) 的信息进行连接。

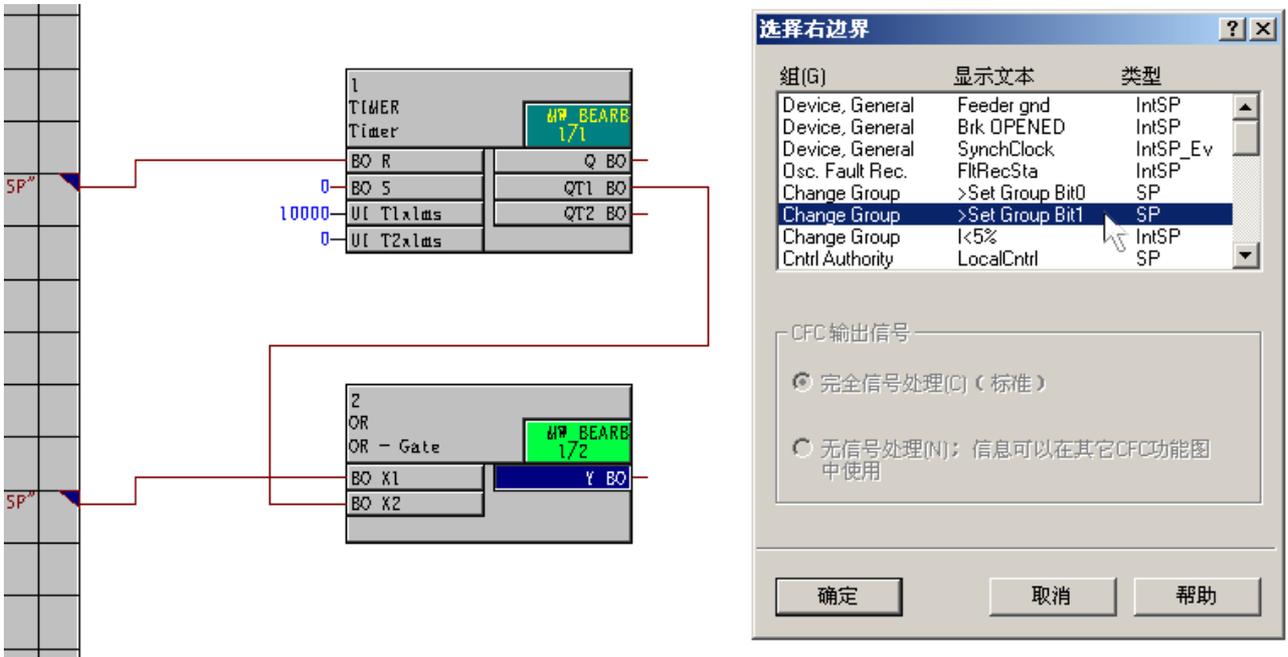


图3—12 在功能块和边界之间创建

编译CFC功能图

用户在生成CFC功能图之后，可以通过菜单功能图 → 编译 → 功能图作为程序对CFC功能图进行编译。

3.2 闪烁的指示灯

在实际的生产过程中，我们经常使用闪烁的指示灯来强调某些特定的状况，例如临界安全状态等。在SIPROTEC4装置中，用户可以使用CFC程序来模拟实际的闪光机制。

注意：

在V4.5版的SIPROTEC装置中可以使用BLINK功能块，该功能块在所有的优先级中都可以使用。

任务

下面是我们将要执行的任务：

- 在许多情况下，一个短时间脉冲，例如按下F1功能键，可以用来转换到一个临界的安全状态。这个状态可以通过一个闪烁的指示灯显示。
- 下一次短时间脉冲，例如按下F2功能键，可以用来结束一个临界的安全状态，同时关闭指示灯。

通过CFC程序 设置功能键

为了通过CFC程序进行功能键F1和F2的设置，需要完成下面的工作：

- 在DIGSI的配置矩阵中，生成一个名称为**LED Flash**的组。
- 为每一个功能键在**LED Flash**设置组中插入一个类型为**Tagging Internal On/Off**的新信息。信息在插入的时候保留系统提供的默认名称。
- 将上述的新信息的默认名称改为**F1 Flash**和**F2 NoFlash**。
- 将功能键**1**配置给信息**F1 Flash**。
- 将功能键**2**配置给信息**F2 NoFlash**。
- 将功能键**F1 Flash**和**F2 NoFlash**在信息目标中配置给CFC。

The screenshot shows the DIGSI configuration matrix for 'Masking I/O (Configuration Matrix)'. The table below represents the data visible in the image.

信息	编号	显示文本	L	类型	信息来源				信息目标									
					BI	F	s	C	开关	LE	缓冲器			s	C	D	CM	
											O	s	T			C	D	
LED FLash		F1 Flash		IntSP		1					00					X		
		F2 No Flas		IntSP		2					00					X		
start driver		F1 start		IntSP							00					X		
Device, General							x	x			x				x	x		
P.System Data 1											x				x			

图3—13 将功能键分配至CFC

通过CFC程序控制指示灯

为了通过CFC程序控制指示灯，需要完成以下的工作：

- 在名称为**LED Flash**的组中插入一条**Tagging Internal On/Off**类型的新信息。这条信息在插入的时候采用系统提供的默认名称。
- 将这条信息的名称更改为**LED Flash**。
- 将名称为**LED Flash**的信息在信息来源中配置给CFC。
- 将指示灯1配置给名称为**LED Flash**的信息。输出为**不保持**的状态。



图3—14 通过CFC程序准备控制指示灯

闪烁节奏

闪烁的快慢可以通过**LED is off**($t_{\text{off}}=250 \text{ ms}$)和**LED is on**($t_{\text{on}}=50 \text{ ms}$)的间隔变化来进行表示。

闪烁

为了在CFC程序中模拟闪烁的指示灯的闪烁情况，需要完成以下工作：

- 在CFC中生成一个名称为**LED Flash**的新功能图，然后将这个功能图打开。
- 通过菜单**编辑** → **运行序列**将**PLC1_BEARB**优先级设置为功能图的标准优先级。以后插入的新信息的优先级都是**PLC1_BEARB**。



注意：

TIMER（通用计时器）功能块只在快速PLC处理（**PLC_BEARB**）的优先级和慢速PLC处理（**PLC1_BEARB**）的优先级上进行工作。

- 闪烁操作是通过两个**TIMER**（通用计时器）功能块来完成的：

第一个计时器通过信号**F1 Flash**启动，启动后内部计时器运行**250ms** (t_{Off})，在此期间将输出信号**QT1**设为**1**。

第一个计时器的输出信号**QT1**可以作为第二个计时器的启动的信号。但内部计时器**T2**是在启动信号再次发生时才能启动。也就是说只有 **250ms** (t_{Off})结束之后才开始启动。**T2**在启动之后的运行时间为**50ms** (t_{On})，然后把相应的输出信号**QT2**设为**1**。这个信号接着被传送到**指示灯 (LED Flash)**，使这个指示灯闪烁。在计时器的时间结束之后，指示灯的闪烁也就结束了。

上面的一系列操作应该重复进行，这样才能够保证闪烁的指示灯不断进行闪光。因此第**2**个计时器的输出数据**QT2**应该通过**OR**（“或”门）传送到第**1**个计时器功能块的启动输入位置。这个反馈操作是允许的，如果在编译**CFC**功能图的过程中出现相应的出错警告，可以不予理睬。

如果需要关闭指示灯，可以将两个计时器的重置输入信号连接到信号**F2 NOFlash**。

- 设置**CFC**功能块



注意：

在插入**CFC**功能块的过程中，应该遵守系统提供的默认运行顺序。**OR**功能块的运行顺序号应该大于第**2**个**TIMER**功能块的运行顺序号。如果有必要，可以通过菜单**编辑** → **运行序列**对功能块的运行顺序进行改变。



注意：

在上面的例子中，使用了**TIMER**（通用计时器）功能块来进行闪烁功能的模拟。根据**SIPROTEC**装置的不同版本，用户在进行模拟的时候也可以使用**TIMER_SHORT**（简易计时器）或者**LONG_TIMER**功能块（计时器（最长计时时间为**1193**小时））。

LONG_TIMER功能块功能块（计时器（最长计时时间为**1193**小时））也可以在所有的**CFC**优先级上进行工作。

- 根据闪光节奏的不同要求，用户可以通过关联菜单**对象属性**对**TIMER**功能块的有关参数进行设置：
 计时器1的**T1x1ms**数值为**250 ms**，
 计时器2的**T2x1ms**数值为**50 ms**。



图3—15 Timer参数设置

- 将功能块相互连接起来，并且和功能图边界进行连接（参见3-16）
 将第1个**TIMER**功能块的设置输入信号**S**与功能图左边界上的**F1 Flash**的信息进行连接。
 将两个**TIMER**功能块的重置输入信号**R**与功能图左边界上的**F2 NoFlash**的信息进行连接。
 将第2个计时器功能块的设置输出信号**QT2**与功能图的右边界上的**LED Flash**的信息进行连接。

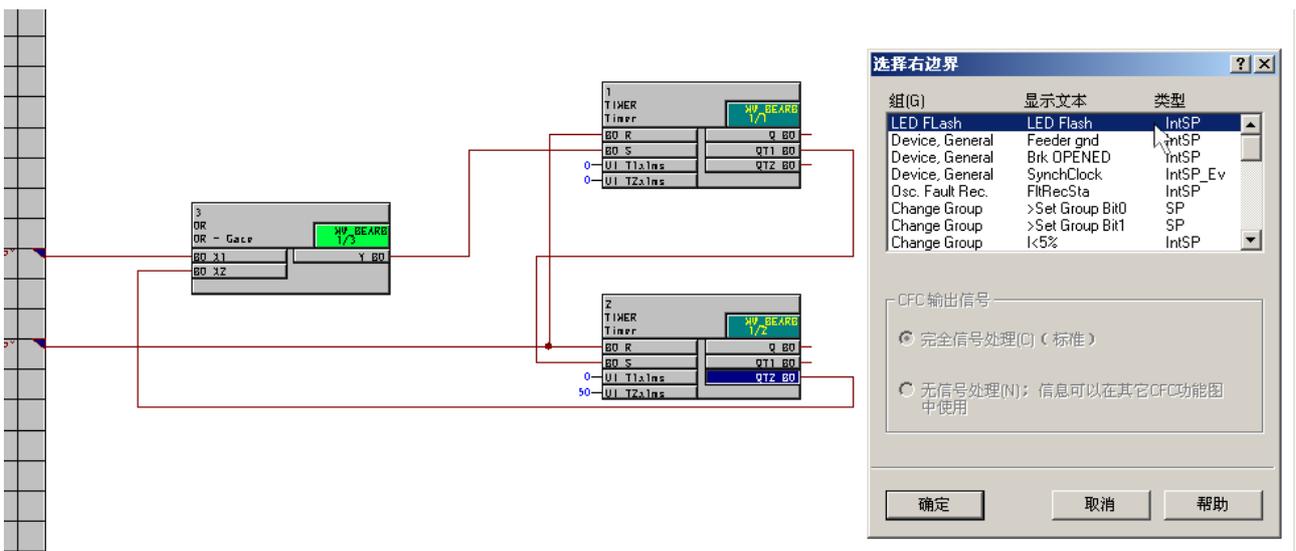


图3—16 在功能块和功能图边界之间生成连接

编译CFC功能图

用户在完成功能图的制作之后，可以通过菜单**功能图** → **编译** → **功能图作为程序**的方式对CFC功能图进行总体的编译工作。

**注意：**

在编译CFC功能图的过程中，由于在CFC功能图内部存在反馈将出现警告提示，但是在这种情况下的反馈是允许的，对于出现的警告提示可以不予理睬。

连续信号作为闪烁指示灯的启动信号

如果没有采用短脉冲（例如功能键**F1**），而是采用连续信号（例如钥匙开关）作为闪烁指示灯的启动信号，这种连续信号必须经过转换。

连续信号可以通过附加的上行计时器功能块转换为相应**F1 Flash**和**F2NoFlash**信号。

将需要转换的信号传送到上述附加计时器的启动输入的位置。**T1x1ms**和**T2x1ms**分别为10ms。

不管信号原来的持续时间是多久，经过转换之后，输入的启动信号转换成位于输出位置**QT1**的持续长度为10ms的短脉冲，而输出的启动信号转换成位于输出位置**QT2**的持续长度为10ms的短脉冲。因此，输出量**QT1**（对应于信号**F1 FALSH**）与第1个计时器的设置输入信号**S**连接起来，而输出量**QT2**（对应于信号**F2 NoFlash**）与两个计时器的重新设置输入信号**R**连接起来。

3.3 反向联锁

通过反向联锁，用户可以建立起带有过流保护功能的经济型母线保护系统。

在这种联锁保护中，我们通过一条馈线向母线馈电，其它馈线与系统的负载连接。典型用途是应用在具有高压或者中压的电力输送网络中。而在超高压的网络中，这种反向联锁的原理很少使用，一般采用单独的母线保护系统。

原理

这种保护系统的工作原理十分简单。连接负载的馈线上的具有过流保护装置的正常启动I>段位可以通过一个开关量输入信号的方式将输入馈线上的 I>>段闭锁。将输入馈线的高电流电位的延迟时间进行固定，从而可以通过馈线启动电位>I保证安全闭锁(50-100 ms)。输入馈线上的启动电位I>>应该明显高于与负载连接的馈线上的启动电流I>段，从而可以保证启动I>进行正确寻址。在联锁处理中，如果有误操作发生，可以导致母线自动关闭。因此，从系统保护的角度来看，该系统具有很高的安全性能。

3.3.1 短路现象发生后应该出现的联锁状态

馈线上的外部短路

外部短路可以导致馈线内的过流保护I>段启动。将这种启动配置给一个触点，从而可以通过指定给功能块的开关量数据的方式将输入馈线上的I>>闭锁。这样I>>即使在延迟时间已经结束的情况下也不能跳闸。用户可以通过短路馈线上的过流保护装置将短路现象消除。在I>降低之后，因为短路现象已经消除，闭锁状态将复位。

母线上面的短路现象

输入馈线的电压设置应该确保在母线发生短路的时候能够启动。但是母线短路不会导致其它馈线上的I>电位的启动。在这个过程中，只有I>>能够启动。在预先设置的延迟时间结束以后，将发出一个TRIP（跳闸）命令，从而使馈电电路上面的断路器跳闸。这样可以消除母线上出现的短路现象。

3.3.2 通过分散布线进行反向联锁

从理论上讲，存在两种通过分散布线进行反向联锁的方法。

方案1

通过将电压和开关量输入的连接，实现对开关量输入数据的闭锁功能。馈线上I>段启动的触点与开关量输入I>>并联，也就是逻辑“或”的关系。对一个或多个触点的访问会导致开关量输入也就是I>>段闭锁。

这个方法存在一个缺点：如果馈线上的过流保护装置发生临时性故障，而馈线发生短路，闭锁将不能发生，从而导致母线产生不正确的开关现象。第二种方案可以解决这一问题。

方案2

使用常闭触点将I>段启动串连形成静态的电流环路。在正常操作条件下，将电压与开关量的输入信号进行连接。然后将这个开关量的输入信号分配给Pickup I>>信号。

如果装置只有闭锁I>>信号，这种情况下应该将系统设置为“无电压有效”的状态，从逻辑的角度讲，这种方法是一种Inverted Pickup（反向启动）的方法。

只要没有I>段启动，I>>段就是有效的，如果母线发生故障，电路将跳闸。相反，如果I>段在馈线中启动，触点打开，环路将断开。这时候将没有电压与开关量的输入信号进行连接。这种状态将导致I>>段闭锁，这种现象为无电压有效。

这种方法的优点就是可以允许装置发生一定程度的故障。在这种情况下，环路将中断，I>>段将自动闭锁。这样可以避免在系统发生短路的情况下带来不必要的操作情况。

3.3.3 通过IEC 61850进行反向联锁

出于反向联锁的安全考虑，对于通过IEC 61850进行分散布线时，也应该采取上述**solution 2**中采取的方法。

要求

需要满足以下条件：

- 反向联锁是通过**IEC 61850 GOOSE service**完成的。
- 馈线是过流保护可以提供闭锁**I>>**段的功能。
- 其它O/C的**I>**启动状态是通过**IEC 61850**进行输送的。
- GOOSE传送特性应作如下设置，第1次重复时间为**4 ms**，最后一次重复时间为**1 ms**。

实施

建议按照下面的方法进行具体实施：

来自馈线O/C装置上的启动信号**I>**，并发包括伴随量属性及时间标志等作为GOOSE信息传递，并具有以上传送特性。

这些GOOSE信息都是被馈电装置所定制。启动信号和状态信息**valid**将被传送到内部的CFC逻辑单元中。

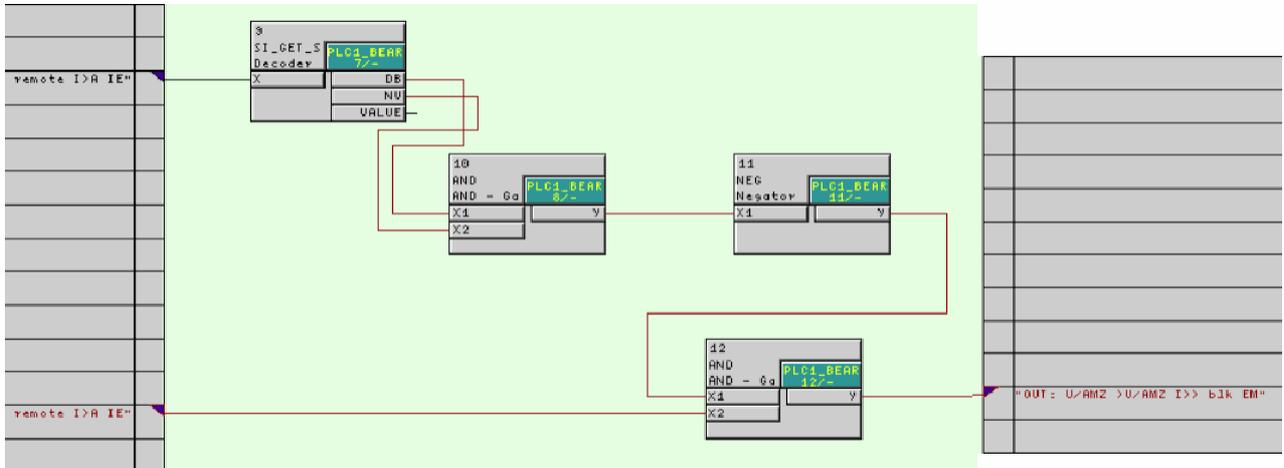
通过使用一个CFC功能图，可以模拟前面列出的传统方法（**solution 2**：静态电流环路）：

- 每一个 **I>**段启动都将导致**I>>**段闭锁。
- 如果馈线的O/C启动信号发生故障，相应的并发启动信号将在馈线装置的超时时间（最后重复时间的两倍）结束之后设置为**invalid**。这种状态的改变将导致**I>>**段的闭锁。

使用这种逻辑方式，可以对通过**solution 2**分散布线进行模拟：装置启动或者装置故障将导致**I>>**段闭锁，也可以避免馈线过流保护装置的不必要操作。

3.3.4 图：通过CFC功能图实现反向联锁

通过下面的CFC功能图，用户可以对SIPROTEC装置中的反向联锁逻辑进行模拟。

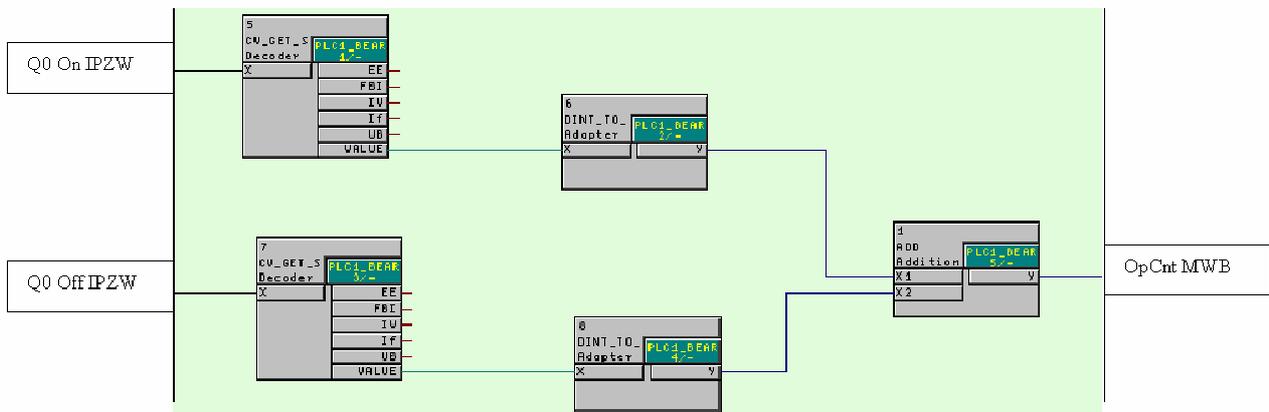


RueckwVerrieg_Beispiel_Plan.tif

图3-17 反向联锁应用实例，CFC功能图

3.4 操作计数

通过下面的CFC功能图，用户可以对SIPROTEC装置中可操作的控制装置的操作次数进行计算。



OpCnt_Beispiel_Plan.tif

图3-18 操作计数的应用实例，CFC功能图

功能原理

Q0 On PMV 和 **Q0 Off PMV** 是脉冲计量值，这些数值的来源（通过CFC配置）分别是双信号**Q0 ON/OFF**的高位和低位。

测量值是通过**CV_GET_STATUS**功能块与状态信息相互分开的。

通过类型转换器**DINT_TO_REAL**，测量值就可以通过逻辑功能块**ADD**进行加法运算。

ADD功能块将**ON/OFF**切换的总量进行求和，然后将数值输送到运算计数器**OpCnt MVU**。

CFC功能块

概述

本章的主要内容是对DIGSI CFC中的功能块进行详细说明。



注意：

SIPROTEC的版本决定了在装置中可用哪些功能块，从而也决定了DIGSI CFC会出现哪些功能块。



注意：

本章中的CFC功能块是通过**块宽度：宽的**的选择项进行显示的：

- 在CFC编辑器中依次点击**选项** → **自定义** → **块/边框宽度**。
- 在显示窗口中**块宽度**选择项下面点击**宽的**，然后点击**确定**键进行确认。

目录

4.1 数据类型	67
4.2 算术运算	69
4.3 基本逻辑运算	76
4.4 信息状态	87
4.5 存储器	105
4.6 控制命令	119
4.7 类型转换器	143
4.8 数据比较	163
4.9 计量值	174
4.10 计时器和时钟	176

4.1 数据类型

下面是DIGSI CFC功能图中可以使用的数据类型：

表4-1 DIGSI CFC功能图中可以使用的数据类型

数据类型	在DIGSI中的显示方式	意义	取值范围
BOOL	BO	逻辑值（二进制数据）	0（假），1（真）
DINT	DI	带符号整数（32位）	-2147483632到2147483632（在DIGSI中是-2147483648到2147483647） 注意： DIGSI中的取值范围是由作为状态信息的附加值所产生的。
INT	I	地址（例如命令地址）	取值范围是通过DIGSI进行定义和输入的。
REAL	R	浮点数据	$-3.402823466e^{+38}$ 到 $3.402823466e^{+38}$
STRUCT	ST	数据结构	各种各样 （包含的信息可以通过信息状态功能块进行处理）
UNIT	UI	不带符号整数（16位）	0到65535
WORD	W	位字段（例如双信号信号）	16#0000到16#FFFF

**数据类型DINT
的状态信息**

对于**DINT**数据类型来说，在DIGSI CFC中定义了一些作为状态信息的附加数据，可以作为有效数据范围的补充：

表4-2 在DIGSI CFC中应用于数据类型DINT的状态信息

DINT数据	状态	意义
-2147483647	OVERFLOW-NEG	数据范围的负尖峰
2147483646	NOT-DEFINED	未定义值
2147483644	LIVE-ZERO	无效，例如因为电路断路
2147483645	NOT-CALCULATED	没有执行计算
2147483646	INVALID	无效值
2147483647	OVERFLOW-NEG	数值范围的正尖峰

**注意：**

在DIGSI中，**DINT**数据的数值范围是没有限制的。当在DIGSI CFC中使用信息的时候，应该保证没有设置这种数据的数值范围。

REAL型数据的状态信息

对于**REAL**类型的数值，可以采用与**DINT**类型数据相同的状态信息。但是状态值超出了有效值范围，因此无法用数字表示。

4.2 算术运算

用户可以对**REAL**类型的测量值进行处理，并且可以通过运算功能块对这些数据进行计算。

以下为可用的算术运算功能块：

- **ABSVALUE** (绝对值)
- **ADD** (求和运算)
- **DIV** (除法运算)
- **MUL** (乘法运算)
- **SQUARE_ROOT**(平方根运算)
- **SUB** (减法运算)

4.2.1 ABSVALUE (绝对值)

功能

在输出位置**AbsVal**，**Absolute Value**功能块可以求出输入**Val**的绝对值。

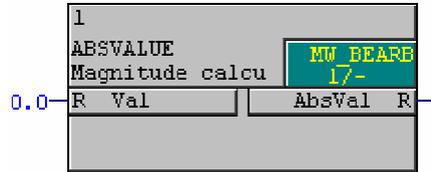


图4-1 ABSVALUE功能块



注意：

Absolute Value功能块只能在优先级“测量值处理”（优先级为**MW_BEARB**）中使用。

I/O分配

ABSVALUE功能块具有以下I/O分配：

表4-3 ABSVALUE功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	Val	REAL	测量值 %值	0.0
输出	AbsVal	REAL	测量值 %绝对值	0.0

4.2.2 ADD（求和）

功能

通过**ADDITION**功能块，用户可以将两个数值**X1**和**X2**（例如测量值）求和，求和的结果在**Y**的位置输出。

在关联菜单中用户可以将输入的量最多增加到**120**个：

- 用鼠标的右击**ADD**功能块，然后在关联菜单中选择**I/O数量**。
- 输入数值后，点击**确定**键进行确认。

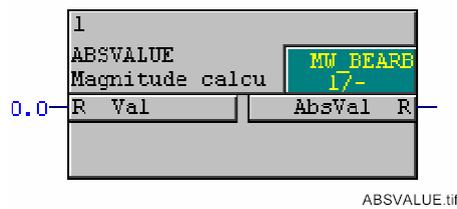


图4-2 ADD功能块

I/O分配

ADD功能块具有以下的I/O分配：

表4-4 **ADD**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	REAL	加数1	0.0
	X2	REAL	加数2	0.0
输出	Y	REAL	求和运算结果 ($Y=X1+X2$)	0.0

4.2.3 DIV (除法)

功能

通过**Division**功能块，用户可以用**X2**（例如测量值）去除以 **X1**。除法运算的结果在**Y**的位置进行输出。

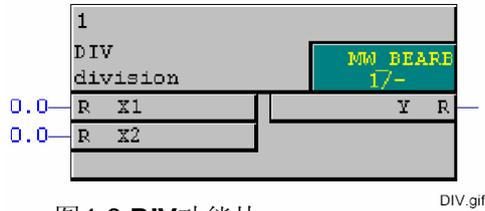


图4-3 DIV功能块

I/O分配

DIV功能块具有以下I/O分配：

表4-5 **DIV**功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	X1	REAL	被除数	0.0
	X2	REAL	除数	0.0
输出	Y	REAL	除法运算结果 ($Y=X1/X2$)	0.0

4.2.4 MUL（乘法）

功能

通过**Multiplication**功能块，用户可以将输入**X1**和**X2**进行乘法运算。乘法运算的结果在**Y**的位置进行输出。

在关联菜单中，用户可以将输入的数量最多增加到120个：

- 用鼠标的右键点击**MUL**功能块，然后在关联菜单中选择**I/O数量**。
- 输入数值后点击**确定**键进行确认。

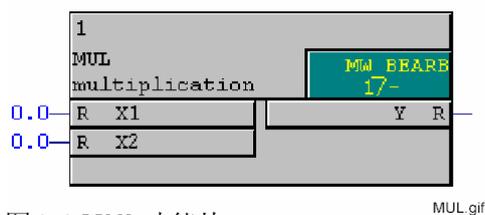


图4-4 MUL功能块

I/O分配

MUL功能块具有以下I/O分配：

表4-6 MUL功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	REAL	乘数1	0.0
	X2	REAL	乘数2	0.0
输出	Y	REAL	乘法运算结果 ($Y=X1*X2$)	0.0

4.2.5 SQUARE_ROOT （平方根）

功能

通过使用**Square Root Extractor**功能块，用户可以计算某个数**X**的平方根。如果**X**的值小于零，错误输出**ERR**将被置1，并且在**Y**的位置输出数值0.0。

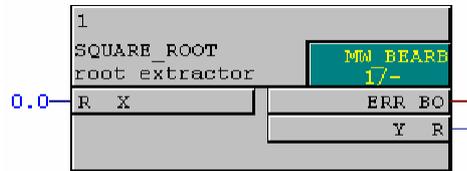


图4-5 SQUARE_ROOT功能块

I/O分配

SQUARE_ROOT功能块具有以下的I/O分配：

表4-7 SQUARE_ROOT功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	REAL	求平方根的数值	0.0
输出	ERR	BOOL	错误输出（如果求平方根的数值小于零，错误输出将被设为1。）	0.0
	Y	REAL	求平方根运算的结果（ $Y=SQR(X)$ ）	0.0

4.2.6 SUB (减法)

功能

通过**Subtraction**功能块，用户可以输入**X1**（例如测量值）中减掉输入**X2**。减法运算的结果在**Y**的位置进行输出。

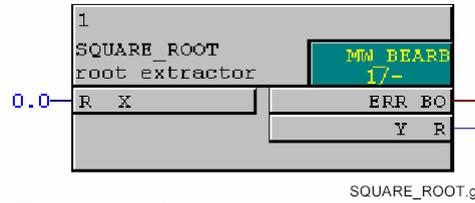


图4-6 SUB功能块

I/O分配

SUB功能块具有以下I/O分配:

表4-8 SUB功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	REAL	被减数	0.0
	X2	REAL	减数	0.0
输出	Y	REAL	减法运算结果 ($Y=X1-X2$)	0.0

4.3 基本逻辑运算

通过基本逻辑运算功能块，用户可以对逻辑信号（布尔值）进行连接或者处理。

以下为可用的基本逻辑运算功能块：

- **AND** (“与” 门)
- **CONNECT** (连接)
- **DYN_OR** (动态 “或” 门)
- **NAND** (“与非” 门)
- **NEG** (“非” 组件)
- **NOR** (“或非” 门)
- **OR** (“或” 门)
- **RISE_DETECTOR** (上升沿检测器)
- **X_OR** (“异或” 门)

4.3.1 AND (“与” 门)

功能

AND Gate功能块可以通过逻辑运算符“AND”将所有的输入信号连接起来，并且生成相应的输出信号。

在关联菜单中用户可以将进行逻辑运算输入的数量最多增加到120个：

- 用鼠标的右键点击**AND**功能块，然后在关联菜单中选择**I/O数量**。
- 输入数值后点击**确定**键进行确认。

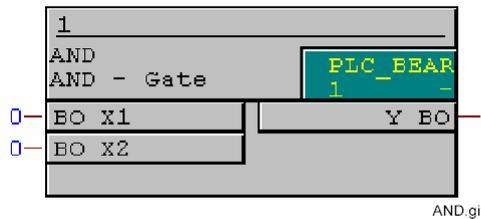


图4-7 AND功能块



注意：

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

AND功能块具有以下的I/O分配：

表4-9 AND功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
	X2	BOOL	输入	0
输出	Y	BOOL	输出 (Y=X1ANDX2)	0

表4-10 AND功能块的真值表

X1输入	X2输入	X3输出
0	0	0
0	1	0
1	0	0
1	1	1

4.3.2 CONNECT (连接)

功能

通过**CONNECT**功能块，用户可以直接将位于**CFC**功能图边框的信号直接连接起来，也就是将**CFC**功能图的输入信息与输出信息进行连接。如果没有这个功能块，将不可能。

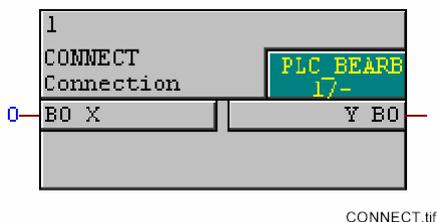


图4-8 **CONNECT**功能块



注意：

CONNECT功能块只能分别在快速PLC处理（**PLC_BEARB**）、慢速PLC处理（**PLC1_BEARB**）、和联锁（**SFS_BEARB**）的优先级上进行工作。



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到**CFC**功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

CONNECT功能块具有以下I/O分配：

表4-11 **CONNECT**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	BOOL	输入（输入到左边界）	0
输出	Y	BOOL	输出（输入到右边界）	0

4.3.3 DYN_OR （动态“或”门）

功能

Dynamic OR Gate功能块可以用来将信息连接起来。

与逻辑“或”门不同，动态“或”门将每一个输入信号的变化在输出位置显示出来。

当装置开始启动的时候，双信号**intermediate position (00)**将被输出。

如果在功能图运行的过程当中没有信号连接到输入，双信号**OFF (01)**将被输出。

如果在功能图运行的过程当中某个信号是有效的，双信号**ON (10)**将被输出。

如果有一个已经活动的信号在输入位置被检测出来之后在另外一个输入位置又有一个附加的输入信号被检测出来，输出信号将设置在**intermediate position (11)**状态的功能图中。在下一个功能图运行时，信号的变化作为一个双信号**ON (10)**而输出。

动态“或”门具有5个输入端。如果用户需要使用超过5个以上的输入信息，可以将几个动态“或”门进行串联，为此，每个动态“或”门的双信号必须通过**DM_DECODE**功能块在前一个动态“或”门的输出位置进行解码，并且这个动态“或”门的输出信号**ON (10)**必须和下一个动态“或”门的输入端相互连接。

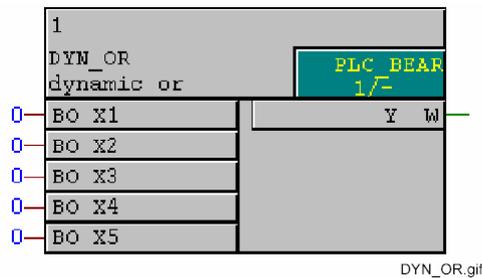


图4-9 DYN_OR功能块



注意：

DYN_OR 功能块只能分别在慢速PLC处理（**PLC1_BEARB**）、测量值处理（**MW_BEARB**）、和联锁（**SFS_BEARB**）的优先级上进行工作。



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O 分配

DYN_OR 功能块具有以下I/O分配

表4-12 **DYN_OR** 功能块具有的I/O分配

	数据名称	数据类型	注释	默认设置
输入	X1	BOOL	输入信号	0
	X2	BOOL	输入信号	0
	X3	BOOL	输入信号	0
	X4	BOOL	输入信号	0
	X5	BOOL	输入信号	0
输出	Y	WORD	双信号输出信号	16#0000

4.3.4 NAND (“与非”门)

功能

NAND Gate功能块可以通过逻辑运算符“**NAND**”将所有的输入信号连接起来，并且生成相应的输出信号。

在关联菜单中用户可以将进行逻辑运算的输入数量最多增加到120个：

- 用鼠标的右键点击**NAND**功能块，然后在关联菜单中选择**I/O数量**。
- 输入数值后点击**确定**键进行确认。

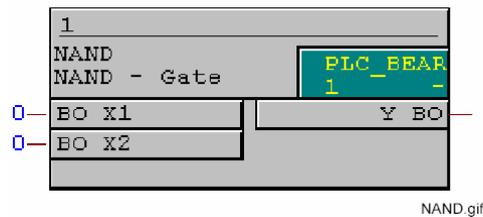


图4-10 NAND功能块



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

NAND功能块具有以下I/O分配：

表4-13 NAND功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
	X2	BOOL	输入	0
输出	Y	BOOL	输出（ $Y=NEG(X1 \wedge X2)$ ）	0

表4-14 NAND功能块的真值表

X1输入	X2输入	Y输出
0	0	1
0	1	1
1	0	1
1	1	0

4.3.5 NEG（“非”门）

功能

Negator功能块可以转换所有的输入信号，并且生成相应的输出信息。

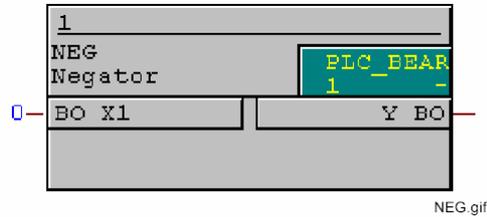


图4-11 NEG功能块



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

NEG功能块具有下面的I/O分配情况：

表4-15 NEG功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
输出	Y	BOOL	输出（Y=NEG（X1））	1

表4-16 NEG功能块的真值表

X1输入	Y输出
0	1
1	0

4.3.6 NOR (“或非”门)

功能

NOR Gate功能块可以通过逻辑运算符“**NOR**”将所有的输入信号连接起来，并且生成相应的输出信号。

在关联菜单中用户可以将进行逻辑运算的输入数量最多增加到120个：

- 用鼠标的右键点击**NOR Gate**功能块，然后在关联菜单中选择**I/O数量**。
- 输入数值后点击**确定**键进行确认。

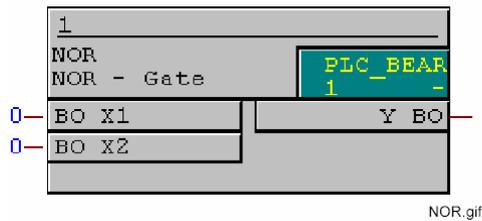


图4-12 NOR功能块



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界**的快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

NOR功能块具有以下的I/O分配：

表4-17 NOR功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
	X2	BOOL	输入	0
输出	Y	BOOL	输出（ $Y = \text{NEG}(X1 \text{ OR } X2)$ ）	0

表4-18 NOR功能块的真值表

X1输入	X2输入	Y输出
0	0	1
0	1	0
1	0	0
1	1	0

4.3.7 OR (“或”门)

功能

OR Gate功能块可以通过逻辑运算符“OR”将所有的输入信号连接起来，并且生成相应的输出信号。

在关联菜单中用户可以将进行逻辑运算的输入数量最多增加到120个：

- 用鼠标的右键点击**OR Gate**功能块，然后在关联菜单中选择**I/O数量**。
- 输入将要进行逻辑运算的数值，然后点击**确定**键进行确认。

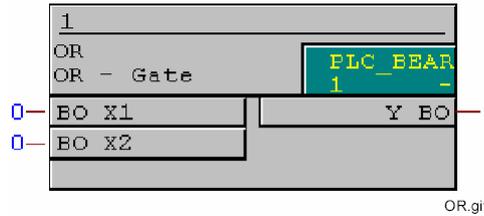


图4-13 OR功能块



注意：

如果在测量值处理（MW_BEARB）和联锁（SFS_BEARB）优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

OR功能块具有下面的I/O分配情况：

表4-19 OR功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
	X2	BOOL	输入	0
输出	Y	BOOL	输出 (Y= X1 OR X2)	0

表4-20 OR功能块的真值表

X1输入	X2输入	Y输出
0	0	0
0	1	1
1	0	1
1	1	1

4.3.8 RISE_DETECT （上升沿检测）

功能

RISE_DETECT功能块可以根据输入位置D的信号在CFC功能图运行的过程中发生的数值上升和下降的变化而相应地分别显示**RISING**输出与**FALLING**输出。

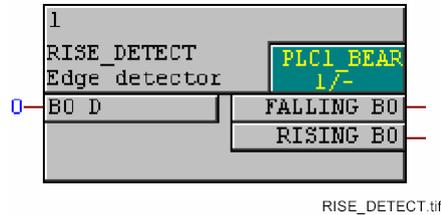


图4-14 RISE_DETECT功能块



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

RISE_DETECT功能块具有下面的I/O分配情况：

表4-21 RISE_DETECT功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	D	BOOL	上升检测器输入	0
输出	FALLING	BOOL	检测到的下降变化	0
	RISING	BOOL	检测到的上升变化	0

4.3.9 X_OR (“异或”门)

功能

X_OR功能块可以通过逻辑运算符“**X_OR**”将所有的输入信号连接起来，并且生成相应的输出信号。

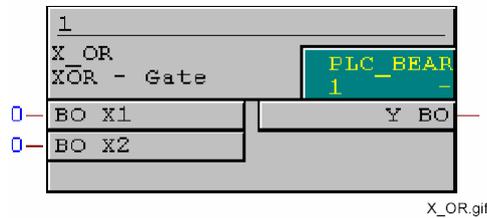


图4-15 X_OR功能块



注意：

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

X_OR功能块具有以下的I/O分配情况：

表4-22 X_OR功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	X1	BOOL	输入	0
	X2	BOOL	输入	0
输出	Y	BOOL	输出 (Y= X1 XOR X2)	0

表4-23 X_OR功能块的真值表

X1输入	X2输入	Y输出
0	0	0
0	1	1
1	0	1
1	1	0

4.4 信息状态

使用信息状态功能块，用户可以对信号状态、测量值和SIPROTEC装置中的计量值进行读取、设置和处理。

以下是用户可以使用的信息状态：

表4-24 信息状态概观

CFC输入/输出（只有英文格式）	CFC长文信息（因为语言不同而不同）	DIGSI信号记录（因为语言不同而不同）	IEC61850（只有英文格式）	装置中的值	意义
信号					
NV	没有更新	NA	OldData	0x0040h	数据不是最新的，例如系统启动、通信故障之后的数据，没有经过配置（或者配置给不存在或者有缺陷的功能块）的数据
DB	抖动打开	FS	Oscillatory	0x0002h	抖动抑制有效（开关量输入闭锁）
测量值					
NV	没有更新	NA	OldData	0x0040h	数据不是最新的，例如系统启动、通信故障之后的数据，没有经过配置（或者配置给不存在或者有缺陷的功能块）的数值
OF	溢出	UL	Overflow	0x0001h	数值因为溢出而不正确
UG	无效	UG		0x0008h	数值无效，例如因为测量值的处理发生故障（过载）

表4-24 信息状态概观

CFC输入/输出（只有英文格式）	CFC长文 信息（因为语言不同而不同）	DIGSI信号记录（因为语言不同而不同）	IEC61850（只有英文格式）	装置中的信息	意义
测量值					
NV	没有更新	NA	OldData	0x0040h	数据不是最新的，例如系统启动、通信故障之后的数据，没有经过配置（或者配置给不存在或者有缺陷的功能块）的数值
IF	因为重新设置导致信息不准确		substituted	0x0020h	执行首次启动。测量值设置为零
FBI	溢出		overflow	0x0100h	最后一次读取导致测量值溢出
IV	无效		invalid	0x0200h	测量值无效，例如在启动过程中执行了一次重新设置，测量值可能丢失
EE	外部错误	EE	failure	0x0400h	外部错误输入的影响
UB	触发器还原位			0x0800h	每次还原之后，信息状态发生改变

下面是可以使用的信息状态功能块：

- **CV_GET_STATUS** (解码器)
- **DI_GET_STATUS** (解码器)
- **DI_SET_STATUS** (编码器)
- **MV_GET_STATUS** (解码器)
- **MV_SET_STATUS** (编码器)
- **SI_GET_STATUS** (解码器)
- **SI_SET_STATUS** (编码器)
- **ST_AND** (“与”门)， 带有信息状态
- **ST_NOT** (“非”组件)， 带有信息状态
- **ST_OR** (“或”门)， 带有信息状态

4.4.1 CV_GET_STATUS

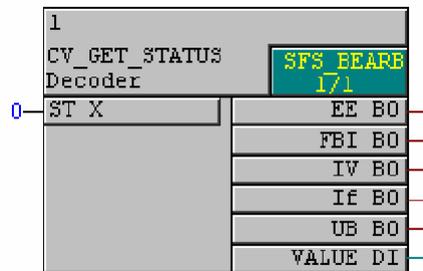
功能

CV_GET_STATUS功能块的功能是对计量值的状态进行解码。

这个功能块和CFC功能图的左边界进行连接,用于读取已配置的的计量值及其状态。

在此,位于输入端**X**的计量值的结构将分割成计量值的数值**VALUE**和以下状态信息:

- EE** (外部错误)
- FBI** (溢出)
- IV** (无效)
- IF** (复位后出错)
- UB** (触发器还原位)



CV_GET_STATUS.tif

图4-16 **CV_GET_STATUS**功能块

I/O分配

CV_GET_STATUS功能块具有以下的I/O分配:

表4-25 **CV_GET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	CVST	带有状态的计量值	(0)
输出	EE	BOOL	外部错误	0
	FBI	BOOL	数据溢出	0
	IV	BOOL	无效	0
	IF	BOOL	因复位导致的错误	0
	UB	BOOL	触发器还原位	0
	VALUE	DINT	计量值	0

4.4.2 DI_GET_STATUS

功能

DI_GET_STATUS功能块的功能是对双信号的状态进行解码。

这个功能块和CFC功能图的左边界进行连接，用于处理双信号及其状态。

在此，位于输入端**X**的双信号的结构将被分为双信号的数值**VALUE**和双信号的状态信息：

- DB** (抖动抑制)
- NV** (没有更新)



图4-17 **DI_GET_STATUS**功能块

I/O分配

DI_GET_STATUS功能块具有以下的I/O分配：

表4-26 **DI_GET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	DIST	带有状态的双信号	(0)
输出	DB	BOOL	抖动开	0
	NV	BOOL	没有更新	0
	VALUE	WORD	双信号	16#0000

4.4.3 DI_SET_STATUS

功能

DI_SET_STATUS功能块的功能是生成一个带有状态信息的双信号。

这个功能块和CFC功能图的右边界进行相互连接,可以设置一个带有状态信息的双信号。

在此,双信号输出**Y**是根据双信号**VALUE**和带有以下状态确定的:

- DB** (抖动抑制)
- NV** (没有更新)

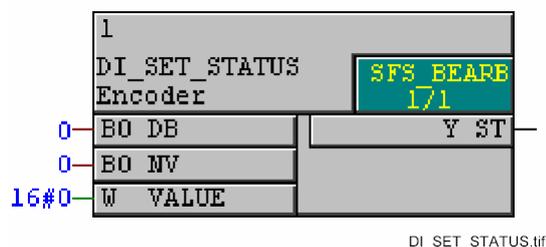


图4-18 **DI_SET_STATUS**功能块

I/O分配

DI_SET_STATUS功能块具有以下的I/O分配情况:

表4-27 **DI_SET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	DB	BOOL	抖动开	0
	NV	BOOL	没有更新	0
	VALUE	WORD	双信号	16#0000
输出	Y	DIST	带有状态的双信号	(0)



注意:

VALUE输入: 在这个过程中传递的只是信息值, 不考虑信息的状态。

**注意:**

输出Y: 如果输出与CFC功能图的右边界连接, 也就是说信息需要在装置中进行进一步的处理, 以下几点将用于装置中的处理过程:

Status NV = TRUE

产生一个闭锁, 也就是说如果输出信息的设置为**not current**, 那么其它值和状态的改变将被忽略。

Status NV

对于输入或者输出过程中的**not updated** (没有更新) 的状态变化, 只将其记录在自发生成的信号中, 在操作记录中不进行记录。

Status DB

对于输入或者输出过程中的**抖动抑制**的状态变化, 只应用在后面的处理过程的检测中。**抖动抑制**不会因此而激活。

所有变化将在一张图中进行处理 (例如转寄到解码器**DI_SET_STATUS**)

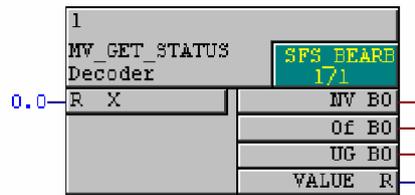
4.4.4 MV_GET_STATUS

功能

MV_GET_STATUS功能块的功能是对测量值的状态信息进行解码。

测量值VALUE将被各自的状态信息替换，例如NV、OF、UG等状态信息，然后在输出位置Y输出。。

- NV** (没有更新)
- OF** (溢出)
- UG** (无效)



MV_GET_STATUS.tif

图4-19 **MV_GET_STATUS**功能块

I/O分配

MV_GET_STATUS功能块具有以下的I/O分配:

表4-28 **MV_GET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	REAL	带有状态的测量值	0.0
输出	NV	BOOL	没有更新	0
	OF	BOOL	溢出	0
	UG	BOOL	无效	0
	VALUE	REAL	测量值	0.0

4.4.5 MV_SET_STATUS

功能

MV_SET_STATUS功能块的功能是设置测量值的状态。

位于输出Y的双信号的结构是根据测量值的数值VALUE以及下面的状态而决定的：

- NV** (没有更新)
- OF** (溢出)
- UG** (无效)

状态信息替换的先后优先顺序为：**UG**、**NV**、**OF**

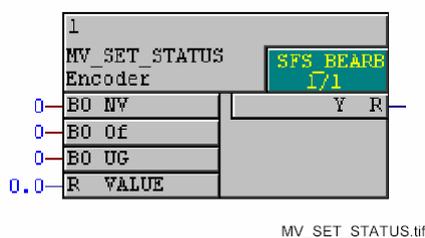


图4-20 **MV_SET_STATUS**功能块

I/O分配

MV_SET_STATUS功能块具有以下的I/O分配：

表4-29 **MV_SET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	NV	BOOL	没有更新	0
	OF	BOOL	溢出	0
	UG	BOOL	无效	0
	VALUE	REAL	测量值	0.0
输出	Y	REAL	带有状态的测量值	(0)



注意：

VALUE输入：在这个过程中传递的只是信息值，不考虑信息的状态。



注意：

输出信息**Y**：如果输出信息与**CFC**功能图的右边界连接，也就是说信息需要在装置中进行进一步的处理，则下面的说明适用于在装置中进行的处理过程：

Status NV = TRUE

产生一个闭锁，也就是说如果输出信息的设置为**not current**，那么其它值和状态的改变将被忽略。

Status NV

对于输入或者输出过程中的**not updated**（没有更新）的状态变化，只将其记录在自发生成的信号中，在操作记录中不进行记录。

Status DB

对于输入或者输出过程中的**chatter block**的状态变化，只应用在后面的处理过程的检测中。**chatter block**不会因此而启动。

所有变化将在一张功能图中进行处理（例如转寄到解码器**DI_SET_STATUS**进行处理）

4.4.6 SI_GET_STATUS

功能

SI_GET_STATUS功能块的功能是对一个单信号的状态信息进行解码。

在这个过程中，单信号的结构在输入位置**X**分割成单信号的数值**VALUE**和带有的状态信息：

DB (抖动抑制)

NV (没有更新)



SI_GET_STATUS.tif

图4-21 **SI_GET_STATUS**功能块



注意：

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

SI_GET_STATUS功能块具有以下I/O分配：

表4-30 **SI_GET_STATUS**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入数据	X	SIST	带有状态的单信号	(0)
输出数据	DB	BOOL	抖动抑制打开	0
	NV	BOOL	没有更新	0
	VALUE	WORD	单信号	16#0000

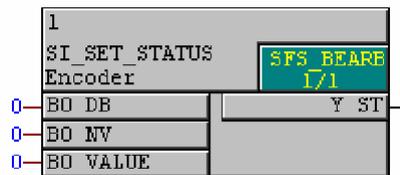
4.4.7 SI_SET_STATUS

功能

SI_SET_STATUS功能块的功能是生成一个带有状态信息的单信号。

在此，位于输出位置Y的单信号的结构是根据测量值VALUE以及下面的状态而决定的：

- DB** (抖动抑制)
- NV** (没有更新)



SI_SET_STATUS.tif

图4-22 SI_SET_STATUS功能块

I/O分配

SI_SET_STATUS功能块具有以下的I/O分配：

表4-31 SI_SET_STATUS功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	DB	BOOL	抖动抑制开	0
	NV	BOOL	没有更新	0
	VALUE	BOOL	单信号	0
输出	Y	SIST	带有状态的单信号	(0)



注意：

请注意以下关于**VALUE**输入的说明：

在这个过程中传递的只是信息值，不考虑信息的状态。

**注意:**

请注意以下关于输出信息 Y 的说明:

如果输出信息与CFC功能图的右边界连接,也就是说信息需要在装置中进行进一步的处理,以下几点将用于装置中的处理过程:

Status NV = TRUE

产生一个闭锁,也就是说如果输出信息的设置为**not current**,那么其它值和状态的改变将被忽略。

Status NV

对于输入或者输出过程中的**not updated**(没有更新)的状态变化,只将其记录在自发生成的信号中,在操作记录中不进行记录。

Status DB

对于输入或者输出过程中的**chatter block**的状态变化,只应用在后面的处理过程的检测中。**chatter block**不会因此而启动。

所有变化将在一张功能图中进行处理(例如转寄到解码器**DI_SET_STATUS**进行处理)。

4.4.8 ST_AND

功能

ST_AND功能块的功能是通过逻辑操作**AND**将两个单信号连接起来，并且生成相应的输出信号。

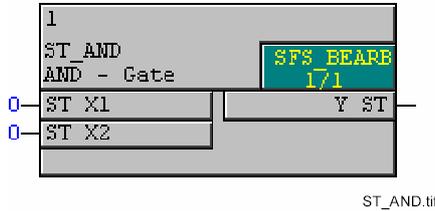


图4-23 **ST_AND**功能块



注意：

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

ST_AND功能块具有以下I/O分配：

表4-32 **ST_AND**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X1	SIST	带有状态信息的输入	(0)
	X2	SIST	带有状态信息的输入	(0)
输出	Y	SIST	带有状态信息的输出	(0)

信息和状态

如果所有信号都被连接至输入端，输出将被设为**INCOMING**。输出信号的状态是所有输入状态的“或”结果。



注意：

如果至少一个输入为**OUTGOING**并且信息状态为**Current(=0)**，那么输出的状态将被设为**Current(=0)**。

状态值

ST_AND块只考虑以下状态值：

- NV（没有更新）
- DB（抖动抑制打开）

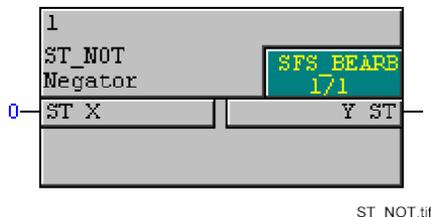
表4-33 ST_AND功能块的真值表

X1输入的状态	X1输入的数值	X2输入的状态	X2输入的数值	Y输出的状态	Y输出的数值
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	0	1	0	1
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	1	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	0	1	0	1
1	1	1	0	1	0
1	1	1	1	1	1

4.4.9 ST_NOT

功能

ST_NOT功能块的功能是开关带有状态信息的单信号，并且生成相应的输出信息。

图4-24 **ST_NOT**功能块**注意:**

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界**的快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

ST_NOT功能块具有以下I/O分配：

表4-34 **ST_NOT**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	SIST	带有状态信息的输入	(0)
输出	Y	SIST	带有状态信息的输出	(0)

信息和状态

ST_NOT 功能块反转输入信号并且用来自以下状态形成输出信号。

状态值

ST_NOT块只考虑以下状态值：

- NV（没有更新）
- DB（抖动抑制打开）

表4-35 **ST_NOT**功能块的真值表

X1输入的状态	X1输入的数值	Y输出的状态	Y输出的数值
0	0	1	1
0	1	1	0
1	0	0	1
1	1	0	0

4.4.10 ST_OR

功能

ST_OR功能块的功能是通过逻辑运算“或”将两个单信号结合起来，并且生成相应的输出信号。

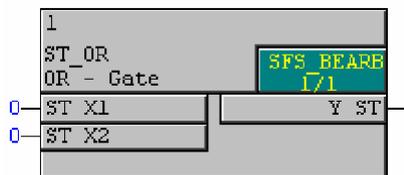


图4-25 ST_OR功能块

ST_OR.tif



注意：

如果在测量值处理（MW_BEARB）和联锁（SFS_BEARB）优先级中连接到CFC功能图左边界快速信号，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

ST_OR功能块具有下面的I/O分配情况：

表4-36 ST_OR功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	X1	SIST	带有状态信息的输入	(0)
	X2	SIST	带有状态信息的输入	(0)
输出数据	Y	SIST	带有状态信息的输出	(0)

信息和状态

当第一个信号被连接至功能块，输出将被设为INCOMING。如果最后一个连接至功能块是OUTGOING，那么输出信号将被设为OUTGOING。其状态是所有输入状态的“或”结果。



注意：

如果至少一个输入为OUTGOING并且信息状态为Current(=0)，那么输出的状态将被设为Current(=0)。

状态值

ST_OR块只考虑以下状态值：

- NV（没有更新）
- DB（抖动抑制打开）

表4-37 ST_OR功能块的真值表

X1输入的状态	X1输入的数值	X2输入的状态	X2输入的数值	Y输出的状态	Y输出的数值
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	1	1
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	1	1

4.5 存储器

通过存储器功能块, 用户可以在CFC功能图没有运行的时候将装置的逻辑状态存储下来。

以下为可用存储器功能块:

- D_FF** (D触发器)
- D_FF_MEMO** (用于重新启动的状态存储器)
- RS_FF** (RS触发器)
- RS_FF_MEMO** (用于重新启动的状态存储器)
- SR_FF** (SR触发器)
- SR_FF_MEMO** (用于重新启动的状态存储器)
- MEMORY** (数据内存)

4.5.1 D_FF (D触发器)

功能

使用**D Flipflop**功能块，可以在**CLK**输入端出现上升沿输入时将输入端**D**的信号传送到输出端**Q**。输出端**Q**的值将保持不变，直到检测到在**CLK**输入端的下一个上升脉冲。



注意:

位于输出端**Q**的信号在第一次启动的时候可以设置为成一个标准信号，然后在每一次重新运行之前进行保存，运行之后可以恢复原先的状态。为了进行上述操作，用户应该使用**D_FF_MEMO**功能块。

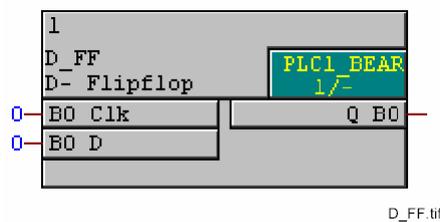


图4-26 D_FF功能块



注意:

D Flipflop功能块只能在优先级为快速PLC处理(**PLC_BEARB**)、慢速PLC处理(**PLC1_BEARB**)、和联锁(**SFS_BEARB**)的等级上面工作。



注意:

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

D_FF功能块具有以下的I/O分配:

表4-38 D_FF功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Clk	BOOL	时钟	0
	D	BOOL	数据	0
输出	Q	BOOL	输出	0

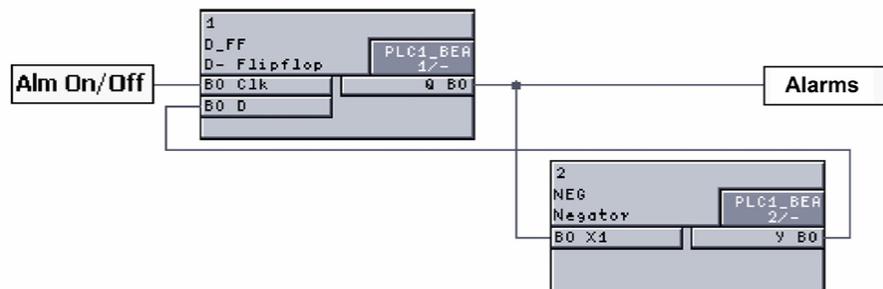
表4-39 D_FF功能块的真值表

Clk输入	D输入	Q输出
0或者1	0	Q_{n-1}
从0变化为1	0	0
0或者1	1	Q_{n-1}
从0变化为1	1	1

应用实例

通过下面的CFC功能图，用户可以通过使用一个功能键来切换hornon和off:

- 在配置矩阵中插入一个内部单信号**horn on/off**，为这个单信号配置一个功能键（信息来源）并连接到CFC功能图（信息目标）。
- 一个新的名为**horn**的输出信号将在CFC功能图中形成，通过一个开关量输出信号与**horn**相连。



D_FF_Bispiel.tif

图4-27 D_FF功能块的应用实例，CFC功能图部分

4.5.2 D_FF_MEMO（带有状态内存的D触发器）

功能

D Flipflop with State Memory（D触发器带有状态内存） 功能块的运行方式与**D_FF**功能块的运行方式类似：

带有状态内存**D**输入端的信号在**CLK**输入端上升沿对传递给输出端**Q**。在检测到下一个**CLK**输入端上升沿之前，位于输出端**Q**的数值保持不变。

另外，在默认方式下，位于输出位置**Q**的信号在系统第一次启动的时候被赋予了**INIT**数值，以后在系统每次重新启动之前复位到初始状态。

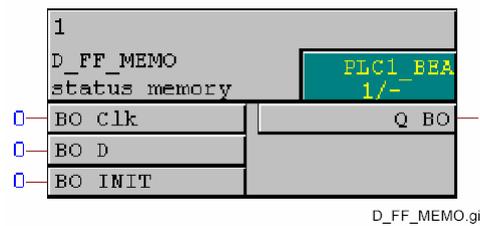


图4-28 D_FF_MEMO功能块



注意：

可以使用的**MEMORY**功能块**RS_FF_MEMO**功能块、**SR_FF_MEMO**功能块、**D_FF_MEMO**功能块、**COUNTER**功能块的最大数量取决于系统可以使用的非易失存储器的容量，并且这个最大数量将通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，应该遵守该装置用户手册中的技术数据。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。



注意：

如果在测量值处理（**MW_BEARB**）和连锁（**SFS_BEARB**）优先级中连接到**CFC**功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

D_FF_MEMO功能块具有以下的I/O分配:

表4-40 **D_FF_MEMO**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Clk	BOOL	时钟	0
	D	BOOL	数据	0
	INT	BOOL	数据默认设置 注意: 默认设置只能应用于 系统的首次启动	0
输出	Q	BOOL	输出	首次启动: Q=INT 再次启动: 当装置关闭的时候 Q=Q

表4-41 **D_FF_MEMO**功能块的真值表

Clk输入	D输入	Q输出
0或者1	0	Q_{n-1}
从0变化为1	0	0
0或者1	1	Q_{n-1}
从0变化为1	1	1

4.5.3 RS_FF (RS触发器)

功能

当使用**RS_FF**功能块的时候，如果输入端**S**输入数值1，就可以将输出端**Q**的数值设置为1。这个数值在**R**接收数值1之前保持不变。

在这个功能块中，输入数据**R**占主导地位，也就是说当数值1在**S**和**R**中同时有效的时候，输出**Q**也要进行重新设置。

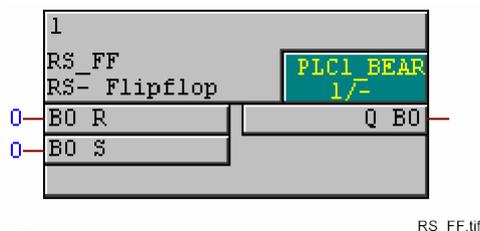


图4-29 RS_FF功能块



注意:

RS_FF功能块只能在优先级为快速PLC处理(**PLC_BEARB**)、慢速PLC处理(**PLC1_BEARB**)、和联锁(**SFS_BEARB**)的等级上面进行工作。



注意:

如果在测量值处理(**MW_BEARB**)和联锁(**SFS_BEARB**)优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

RS_FF功能块具有以下I/O分配:

表4-42 **RS_FF**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	R	BOOL	复位	0
	S	BOOL	设置	0
输出	Q	BOOL	输出	0

表4-43 RS_FF功能块的真值表

R输入	S输入	Q _n 输出
0	0	Q _{n-1}
0	1	1
1	0	0
1	1	0

4.5.4 RS_FF_MEMO （带有状态内存的RS触发器）

功能

RS Flipflop with State Memory（RS触发器带有状态内存） 功能块的运行方式与**RS_FF**功能块的运行方式类似：

如果**S**输入端 值为1，就可以将输出端**Q**的数值设置为1。这个数值在**R**接收 数值1之前保持不变。

在这个功能块中，输入**R**占主导地位，也就是说当值1在**S**和**R**中同时有效的时候，输出**Q**也要进行重新设置。

另外，在默认方式下，位于输出端**Q**的信号在系统第一次启动的时候被赋予了**INIT**数值，以后在系统每次重新启动之前复位到初始状态。

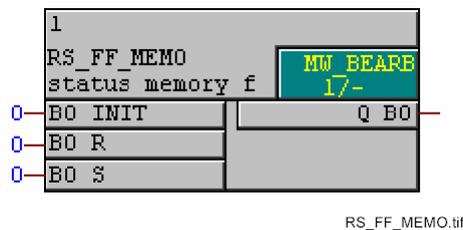


图4-30 RS_FF_MEMO功能块



注意：

可以使用的**MEMORY**功能块**RS_FF_MEMO**功能块、**SR_FF_MEMO**功能块、**D_FF_MEMO**功能块、**COUNTER**功能块的最大数量取决于系统可以使用的非易失存储器的容量，并且这个最大数量将通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，应该遵守该装置用户手册中的技术数据。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到**CFC**功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

RS_FF_MEMO功能块具有以下的I/O分配:

表4-44 RS_FF_MEMO功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	INT	BOOL	数据默认设置 注意: 默认设置只能应用于系统的首次启动	0
	R	BOOL	复位	0
	S	BOOL	设置	0
输出	Q	BOOL	输出	首次启动: Q=INT 再次启动: 当装置关闭的时候 Q=Q

表4-45 RS_FF_MEMO功能块的真值表

R输入	S输入	Q _n 输出
0	0	Q _{n-1}
0	1	1
1	0	0
1	1	0

4.5.5 SR_FF (SR触发器)

功能

当使用**SR Flipflop**功能块的时候，如果 **S**输入端值为1 输出端**Q**将被设为1。这个数值在**R**接收数值1之前保持不变。

在这个功能块中，输入端**S**占主导地位，也就是说当数值1在**S**和**R**中同时有效的时候，输出**Q**也要进行重新设置。

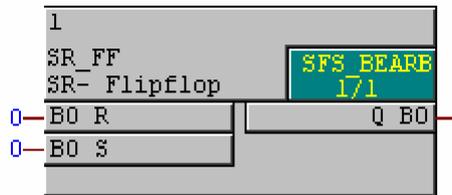


图4-31 SR_FF功能块 SR_FF.tif



注意：

SR Flipflop功能块只能在优先级为快速PLC处理(**PLC_BEARB**)、慢速PLC处理(**PLC1_BEARB**)、和联锁(**SFS_BEARB**)的等级上面进行工作。



注意：

如果在测量值处理 (**MW_BEARB**)和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

SR_FF功能块具有下面的I/O分配：

表4-46 **SR_FF**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	S	BOOL	设置	0
	R	BOOL	复位	0
输出	Q	BOOL	输出	0

表4-47 SR_FF功能块的真值表

R输入	S输入	Q _n 输出
0	0	Q _{n-1}
1	0	1
0	1	0
1	1	1

4.5.6 SR_FF_MEMO （带有状态内存的SR触发器）

功能

SR Flipflop with State Memory （带有状态内存的SR触发器）功能块的运行方式与**SR_FF**功能块的运行方式类似：

如果输入端**S** 值为1， 输出端**Q**值将被设为1。这个数值在**R**接收用户 值1之前保持不变。

在这个功能块中，输入端**S**占主导地位，也就是说当数值1在**S**和**R**中同时有效的时候，输出**Q**也要进行重新设置。

另外，在默认方式下，位于输出位置**Q**的信号在系统第一次启动的时候被赋予了**INIT**数值，以后在系统每次重新启动之前复位复到初始状态。

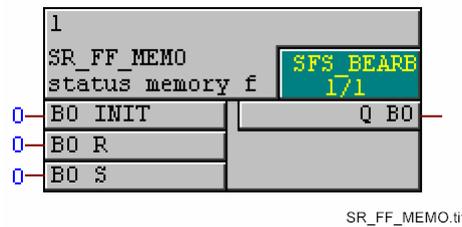


图4-32 SR_FF_MEMO功能块



注意：

可以使用的**MEMORY**功能块**RS_FF_MEMO**功能块、**SR_FF_MEMO**功能块、**D_FF_MEMO**功能块、**COUNTER**功能块的最大数量取决于系统可以使用的非易失存储器的容量，并且通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，应该遵守该装置用户手册中的技术数据。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到**CFC**功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

SR_FF_MEMO功能块具有以下的I/O分配:

表4-48 SR_FF_MEMO功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	INT	BOOL	数据默认设置 注意: 默认设置只能应用于系统的首次启动	0
	S	BOOL	设置	0
	R	BOOL	复位	0
输出	Q	BOOL	输出	首次启动: Q=INT 再次启动: 当装置关闭的时候 Q=Q

表4-49 SR_FF_MEMO功能块的真值表

R输入	S输入	Q _n 输出
0	0	Q _{n-1}
1	0	1
0	1	0
1	1	1

4.5.7 MEMORY（内存）

功能

当Clk检测到上升沿信号时，MEMORY功能块D输入端的信号传递给输出Q。该输出保持不变知道Clk输入端检测到下一上升沿信号。

另外，在默认方式下，位于输出位置Q的信号在系统第一次启动的时候被赋予了INIT数值，以后在系统每次重新启动之前复位复到初始状态。



注意：

可以使用的MEMORY功能块RS_FF_MEMO功能块、SR_FF_MEMO功能块、D_FF_MEMO功能块、COUNTER功能块的最大数量取决于系统可以使用的非易失存储器的容量，并且通过CFC编译器进行监测。用户在使用某种SIPROTEC装置的过程中，应该遵守该装置用户手册中的技术数据。

在编译CFC功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在CFC功能图的编译日志中。



注意：

如果在测量值处理（MW_BEARB）和联锁（SFS_BEARB）优先级中连接到CFC功能图左边界的快速信号，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

MEMORY功能块具有以下I/O分配：

表4-50 MEMORY功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Clk	BOOL	时钟	0
	D	BOOL	数据	0
	INT	BOOL	数据默认设置 注意： 默认设置只能应用于系统的首次启动	0
输出	Q	BOOL	输出	首次启动： Q=INT 再次启动： 当装置关闭的时候 Q=Q

4.6 控制命令

通过控制命令功能块，用户可以启动或者取消控制命令，也可以接收关于开关命令的信息。

下面是用户可用的控制命令功能块：

- BOOL_TO_CO** (布尔逻辑到命令的转换)
- BOOL_TO_IC** (布尔逻辑到内部单信号的转换)
- BOOL_TO_IE** (布尔逻辑到内部单信号的转换)
- CMD_CANCEL** (取消命令)
- CMD_CHAIN** (操作顺序)
- CMD_INF** (命令信息)
- LOOP** (信号反馈)

4.6.1 BOOL_TO_CO (布尔逻辑到命令的转换)

功能 Boolean to Command功能块可以生成一个开关命令

开关命令由输入端TRIG触发，由参数ORIGIN,PROP,VAL和TIME来定义。

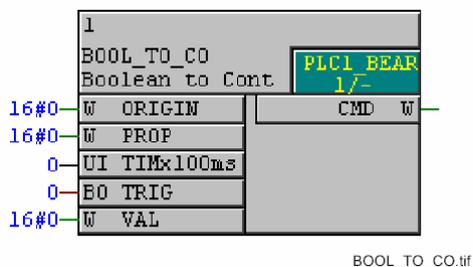


图4-33 BOOL_TO_CO功能块



注意：

BOOL_TO_CO功能块只能在优先级为快速PLC处理（**PLC_BEARB**）和慢速PLC处理（**PLC1_BEARB**）的等级上面进行工作，并且能够用于与右边界之间的直接连接。

I/O分配 **BOOL_TO_CO**功能块具有下面的I/O分配：

表4-51 **BOOL_TO_CO**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	PROP	WORD	命令属性	16#0000
	TRIG	BOOL	触发	0
	ORIGIN	WORD	命令来源	16#0000
	VAL	WORD	开关方向	16#0000
	TIMx100ms	UNIT	输出时间（精度100ms；有效数值范围为从0到32767）	0
输出	CMD	WORD	命令	16#0000

PROP设置

在**PROP**（命令属性）设置过程中可以使用以下数据：

表4-52 **BOOL_TO_CO**功能块的数据，**PROP**设置

值（16进制）	值（10进制）	命令属性
00	0	未解锁
01	1	解锁设置点=实际
02	2	站闭锁的解锁
04	4	间隔层闭锁解锁
08	8	-(忽略)
10	16	-(忽略)
20	32	解锁双命令闭锁
40	64	解锁保护装置闭锁
80	128	当站或间隔层闭锁时解锁

当**PROP** 设 为00（16进制）时，命令将使用在DIGSI配置矩阵中设置的原始值。

**注意：**

如果希望能够同时定义几个命令属性，应该分别命令累加16进制数值。

ORIGIN 设置

通过**ORIGIN**设置，用户可以给已经生成的开关命令分配一个命令特定的命令来源。例如，如果用户将**ORIGIN**设为**01**，则命令在运行过程中将作为本地命令（来自装置自身）。

以下参数可以用于**ORIGIN**命令设置：

表4-53 **BOOL_TO_CO**功能块的数值，**ORIGIN**设置

值	意义
00	自动生成的命令
01	集成操作，装置面板控制（例如来自装置自身）
02	DIGSI、SICAM，本地控制（例如DIGSI远程，DIGSI总线）
03	远程控制中心，远程控制（例如WinCC，远程控制站）

VAL设置

以下参数用于**VAL**（动作方向）的参数设置：

表4-54 **BOOL_TO_CO**功能块的数值，**VAL**设置

值	意义
01	关闭（01）
02	打开（10）

TIMx100ms设置

在**TIMx100ms**（输出时间）参数中，用户可以按照100毫秒的时间增量进行输出时间的设置。

如果将**TIMx100ms**设置为0，那么将使用在DIGSI配置矩阵中的原始设置。

在SIPROTEC装置启动时会使用**TIMx100ms**参数（输出时间）。此参数在运行过程中不能改变。

可以接受的时间范围应该小于3276.8秒，更大的数值将不被系统接受。

IE 输出

IE输出必须在功能图右边界的位置与命令直接连接。

在**IE**输出的位置，这个功能块将显示以下的性能：如果信号在**TRIG**输入端从0改变为1，那么通过**PROP**、**ORIGIN**、**VAL**、以及**TIME**设置生成的命令将转换到**CMD**。

4.6.2 BOOL_TO_IC (布尔逻辑到内部单信号的转换)



注意：

BOOL_TO_IC功能块和**BOOL_TO_IE**功能块的结构和功能都相同。在固件配备比较旧的SIPROTEC4装置中，功能块的名称仍然是**BOOL_TO_IE**。

功能

Boolean to Internal IC功能块可以生成一个作为内部单信号的命令。

通过设置**ORIGIN**、**PROP**、**VAL**、**TIME**的参数将内部的信号定义为一个命令 并由**TRIG**输入端的信号触发的命令。

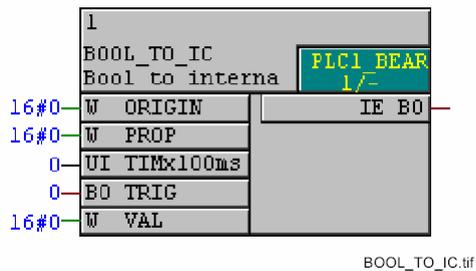


图4-36 **BOOL_TO_IC**功能块



注意：

Boolean to Internal IC功能块只能在优先级为快速PLC处理 (**PLC_BEARB**)、慢速PLC处理 (**PLC1_BEARB**)、以及联锁 (**SFS_BEARB**) 的等级上面工作。

I/O分配

BOOL_TO_IC功能块具有以下I/O分配：

表4-55 **BOOL_TO_IC**功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入	PROP	WORD	命令属性	16#0000
	TRIG	BOOL	跳闸	0
	ORIGIN	WORD	命令来源	16#0000
	VAL	WORD	开关方向	16#0000
	TIMx100ms	UNIT	输出时间（精度100ms；有效数值范围为从0到32767）	0
输出	IE	BOOL	命令（内部单信号）	0

PROP设置

在**PROP**（命令属性）设置过程中可以使用以下值：

表4-56 **BOOL_TO_IC**功能块的数值，**PROP**设置

数值（16进制）	数值（10进制）	命令属性
00	0	没有解锁
01	1	解锁设置点=实际
02	2	解锁站闭锁
04	4	解锁间隔层闭锁
08	8	（没有影响）
10	16	（没有影响）
20	32	解锁双命令功能闭锁
40	64	解锁保护装置闭锁
80	128	当站或间隔层闭锁时解锁

当**PROP**设为00（16进制）的时候，将使用DIGSI配置矩阵中设置的原始的内部单信号命令属性。



注意：

如果用户希望能够同时定义内部单信号的几个命令属性，应该分别累加16进制值。

ORIGIN 设置

通过**ORIGIN**设置，用户可以给已经生成的内部单信号分配一个特定的命令来源。例如，如果用户将**ORIGIN**设为**01**，则命令在运行过程中将作为本地命令（来自装置自身）。

以下值可以作为**ORIGIN**（命令来源）参数设置：

表4-57 **BOOL_TO_IC**功能块的值，**ORIGIN**设置

值	意义
00	自动生成的命令
01	集成操作，装置面板控制 (例如来自装置自身)
02	DIGSI、SICAM，本地控制 (例如DIGSI远程，DIGSI总线)
03	远程控制中心，远程控制 (例如WinCC，远程控制站)

VAL设置

以下参数用**VAL**（动作方向）的参数设置：

表4-58 **BOOL_TO_IC**功能块的值，**VAL**设置

值	意义
01	关闭（01）
02	打开（10）

TIMx100ms设置

在**TIMx100ms**（输出时间）参数中，用户可以按照100毫秒的时间增量进行输出时间的设置。

如果将**TIMx100ms**设置为0则使用在DIGSI配置矩阵中作为内部单信号的原始命令输出时间。

在SIPROTEC装置启动时使用**TIMx100ms**参数（输出时间）会在运行过程中不能改变。

可以接受的时间范围应该小于3276.8秒，更大的数值将不被系统接受。

IE输出

在**IE**输出端，这个功能块将显示下面的性能：

如果在**TRIG**输入端由0变化至1，由**PROP**、**ORIGIN**、**VAL**和**TIME**参数设置的内部单信号命令将被切换到**CMD**。

4.6.3 BOOL_TO_IE

**注意：**

BOOL_TO_IE功能块和**BOOL_TO_IC**功能块的结构和功能都相同。在固件配备比较新的SIPROTEC4装置中，功能块的名称是**BOOL_TO_IE**。

功能

BOOL_TO_IE功能块可以生成一个作为内部单信号的命令。

这个命令可以通过设置**ORIGIN**、**PROP**、**VAL**、**TIME**的参数，将内部单信号定义为一个由**TRIG**输入端的信号触发的命令。

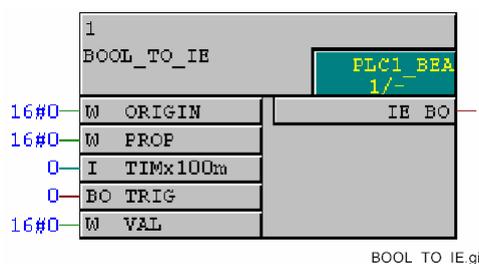


图4-37 **BOOL_TO_IE**功能块

**注意：**

BOOL_TO_IE功能块只能在优先级为快速PLC处理（**PLC_BEARB**）、慢速PLC处理（**PLC1_BEARB**）、以及联锁（**SFS_BEARB**）的等级进行工作。

I/O分配

BOOL_TO_IE功能块具有以下的I/O分配:

表4-59 **BOOL_TO_IE**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	PROP	WORD	命令属性	16#0000
	TRIG	BOOL	跳闸	0
	ORIGIN	WORD	命令来源	16#0000
	VAL	WORD	开关方向	16#0000
	TIMx100ms	UNIT	输出时间（精度100ms；有效数值范围为从0到32767）	0
输出	IE	BOOL	命令（内部单信号）	0

PROP设置

在**PROP**（命令属性）设置过程中可以使用以下值:

表4-60 **BOOL_TO_IE**功能块的值，PROP设置

数值（16进制）	数值（10进制）	命令属性
00	0	没有解锁
01	1	解锁设置点=实际
02	2	解锁站闭锁
04	4	解锁间隔层闭锁
08	8	（没有影响）
10	16	（没有影响）
20	32	解锁双命令功能闭锁
40	64	解锁保护装置闭锁
80	128	当站或间隔层闭锁时解锁

当**PROP**设为为00（16进制）时将使用DIGSI配置矩阵中设置的原始的内部单信号的命令属性。



注意：

如果用户希望能够同时定义多个内部单信号的命令属性，应该分别累加16进制值。

ORIGIN 设置

通过设置**ORIGIN**，用户可以给已经生成的内部单信号分配一个特定的命令来源。例如，如果用户将**ORIGIN**设为01，那么命令在运行过程中将作为本地命令（来自装置自身）。

以下值可以作为**ORIGIN**（命令来源）的参数设置：

表4-61 **BOOL_TO_IE**功能块的值，**ORIGIN**设置

数值	意义
00	自动生成的命令
01	集成操作，装置面板控制（例如来自装置自身）
02	DIGSI、SICAM，当地控制（例如DIGSI远程，DIGSI总线）
03	远程控制中心，远程控制（例如WinCC，远程控制站）

VAL设置

以下参数用于**VAL**（动作方向）的参数设置：

表4-62 **BOOL_TO_IE**功能块的数值，**VAL**设置

数值	意义
01	关闭（01）
02	打开（10）

TIMx100ms设置

在**TIMx100ms**（输出时间）参数中，用户可以按照100毫秒的时间增量进行输出时间的设置。

如果将**TIMx100ms**设置为零则使用DIGSI配置矩阵中设置的作为内部单信号的原始命令输出时间。

（输出时间）在SIPROTEC装置启动时会使用**TIMx100ms**参数，此参数在运行过程中不能改变。

可以接受的时间范围应该小于**3276.8**秒，更大的值将不被系统接受。

IE输出

IE输出必须直接在右边界与命令连接。

在**IE**输出端，这个功能块将显示下面的性能：

如果在**TRIG**输入端由0变化至1，由**PROP**、**ORIGIN**、**VAL**和**TIME**参数设置的内部单信号的命令将被切换到**CMD**。

4.6.4 CMD_CANCEL（取消命令）

功能

通过**Cancel Command**功能块，用户可以取消一个正在运行的命令。

如果 对象地址**DEVICE**被分配为**-1（默认值）**，那么所有命令都会被取消。不管其它操作条件处于什么状态，比如开关权限和联锁状态等等，只要在名称为**TRIG**的输入端输入一个上升沿脉冲，就可以取消命令。

ORIGIN的参数不会影响**CMD_CANCEL**块的运行，这个信息只是一种信号记录的附加信息而已。



图4-38 **CMD_CANCEL**功能块



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界**的快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

CMD_CANCEL功能块具有以下I/O分配：

表4-63 **CMD_CANCEL**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	DEVICE	INT	将要取消的命令。 将输入与功能图的边界连接。将要取消的命令的名称将在边框的位置显示。	-1
	ORIGIN	WORD	命令来源	16#0000
	TRIG	BOOL	启动输入	0

ORIGIN 设置

通过**ORIGIN**的参数设置，用户可以为命令取消功能块分配一个命令来源。例如，如果用户将**01**分配给**ORIGIN**，那么命令在运行过程中将作为本地命令（来自装置自身）。

以下值可以用于**ORIGIN**（命令来源）设置：

表4-64 **CMD_CANCEL**功能块的数值，**ORIGIN**设置

值	意义
00	自动生成的命令
01	集成操作, 装置面板控制(如来自装置自身)
02	DIGSI、SICAM, 本地控制 (例如DIGSI远程, DIGSI总线)
03	远程控制中心, 远程控制 (例如WinCC, 远程控制站)

4.6.5 CMD_CHAIN（操作顺序）

功能

Switching Sequence功能块可以生成一个包括几个单独命令的命令链条。

这个功能块具有以下的基本功能：

- 提交命令
- 等待已经成功执行的命令的反馈信息
- 在功能块输出端返回成功信息，并等待系统进一步指令

在这个功能块中，操作顺序是作为一个指令链条进行设置的。几个**CMD_CHAIN**功能块可以按照一定次序串行执行。功能块链条采用的是菊链机理。在一个指令链条中，某个功能块的执行顺序是由这个功能块在指令链条中的位置所决定的。

在命令链中的最后一条命令成功执行完后，可以通过**LOOP**功能块将整个操作顺序复位。



CMD_CHAIN.tif

图4-39 **CMD_CHAIN**功能块



注意：

CMD_CHAIN功能块只能在优先级为快速PLC处理(**PLC_BEARB**)、慢速PLC处理(**PLC1_BEARB**)、和联锁(**SFS_BEARB**)的等级上面工作。



注意：

如果需要将**CMD_CHAIN**功能块在优先级为慢速PLC处理（**PLC1_BEARB**）的等级上面进行工作，则一定要同时使用**LOOP**功能块，从而可以在命令链条成功执行完最后一个命令后，可以通过**LOOP**功能块将整个操作顺序进行复位。

**注意：**

对于**V4.5**版本以及更低版本SIPROTEC装置，应该满足下面的条件：

CMD_CHAIN功能块的最大使用数量不能超过**20**个，

--不管是在快速PLC处理（**PLC_BEARB**）

--还是在慢速PLC处理（**PLC1_BEARB**）的等级上工作。

I/O分配 **CMD_CHAIN**功能块具有以下的I/O分配：

表4-65 **CMD_CHAIN**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入数据	DCI	BOOL	功能块的菊链输入。 在一个命令链条中，命令之间的相互连接应该通过上一个 CMD_CHAIN 功能块的 DCO 输出来进行。	0
	OBJADR	INT	等待寻址的命令。 输入命令必须和功能图的左边界连接，命令的名称在这个位置输入。	0
	PROP	WORD	命令属性	16#0000
	T	UNIT	输出时间（精度 100ms ；有效数值范围为从 0 到 32767 ）	0
	VAL	WORD	开关方向	16#0000

表4-65 CMD_CHAIN功能块的I/O分配

	名称	数据类型	注释	默认设置
输出数据	ABORT	BOOL	取消 如果一个正在运行的命令被取消 (AB ⁺)，则输出是有效的。	0
	CMD_ERR	BOOL	命令错误 如果命令没有成功执行 (BF ⁻)，则输出是有效的。	0
	DCO	BOOL	功能块的菊链输出。 在一个命令链条中，命令之间的相互连接应该通过下一个CMD_CHAIN功能块的DCI输入来进行。如果开关命令被成功执行，则输出有效。当DCI输入无效时DCO也将输出无效。	0
	ERR	BOOL	组错误 如果ABORT、CMD_ERR、或者FB_ERR有效，或者如果有内部错误发生，则输出是有效的。	0
	FB_ERR	BOOL	反馈错误 如果命令执行以后没有收到反馈 (RM ⁻)，则输出是有效的。	0

PROP设置

在**PROP**（命令属性）设置过程中可以使用下面的数值：

表4-66 CMD_CHAIN功能块的数值，PROP设置

16进制值	10进制值	命令属性
00	0	没有解锁
01	1	解锁设置点=实际
02	2	解锁站闭锁
04	4	解锁间隔层闭锁
08	8	同步检查解锁（SY配置忽略）

表4-66 CMD_CHAIN功能块的值，PROP设置

16进制值	10进制值	命令属性
10	16	带有同步检测（同步状态被强制检测）
20	32	解锁双命令闭锁
40	64	解锁保护装置闭锁
80	128	当站和间隔层闭锁时解锁

当PROP的值为00（16进制）的时候使用在配置矩阵中的原先的命令属性。

**注意：**

如果用户希望能够同时定义几个命令属性，应该分别累加各个命令的16进制值。

VAL设置

以下参数用于VAL（运动方向）的参数设置：

表4-67 CMD_CHAIN功能块的值，VAL设置

值	意义
01	关闭（01）
02	打开（10）

T设置

在T（输出时间）参数中，用户可以按照100毫秒的时间间隔进行输出时间的设置。

如果将T设置为零则使用在DIGSI配置矩阵中原先设置的输出时间。

是在SIPROTEC装置启动时将会使用T参数（输出时间），此参数在运行过程中不能改变。

可以接受的时间范围应该小于3276.8秒，更大的数值将不被系统接受。



注意:

通过在命令链中的第一个功能块DCI输入端的复位信号，可以将一个正在运行的顺序开关取消。这个通过**DCO**输出将信号开关到同一个命令链条中的全部后续功能块。命令链条中的有效功能块通过**DCI=1**以及**DCO=0**的符号进行标记。

在某个开关命令正在执行的过程中，如果某个**DCI**输入信号丢失，将自动生成一个当前命令的取消命令，开关命令将被终止。然后将等待下一步操作：

可以取消的命令:

当前可以放弃的开关命令将立即终止。这条命令的后续开关操作将不予执行。

不能取消的命令:

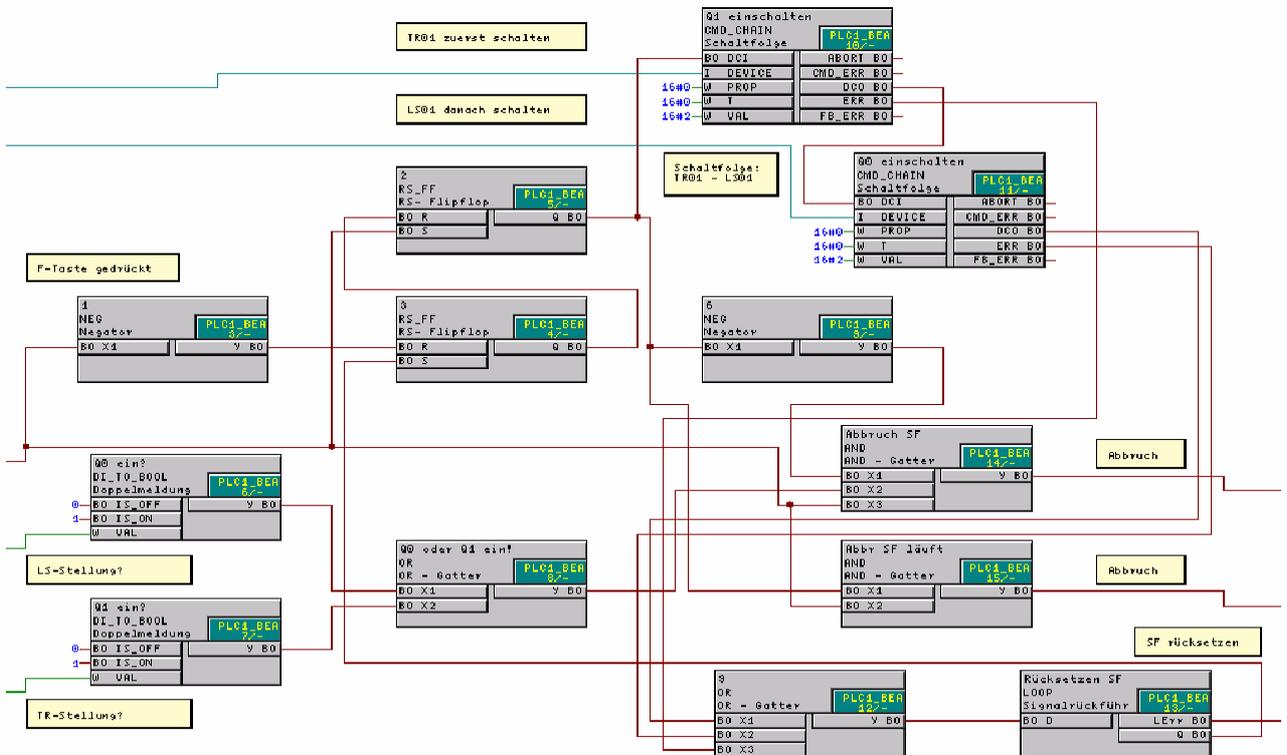
当前不可以取消的开关命令将立即终止。这条命令的后续开关操作将不予执行。

一条命令是否可以取消是由这条命令的运行方式（对象属性）所决定的：

- 不可取消的持久性输出。
- 可以取消的脉冲输出。

应用实例

下面的例子表示两个相邻的操作顺序功能块，其中**TR01**功能块首先运行，然后运行**LS01**功能块：



Schaltfolge.gif

图4-40 操作顺序实例

4.6.6 CMD_INF (命令信息)

功能 **Command Information**功能块可以通过一个初始化的开关命令提供关于命令的有关信息。



注意:

Command Information 功能块主要应用于联锁优先级（优先级为 **SFS_BEARB**），可以检测某开关对象的开关状态。只有在某些特定的条件下，这个功能块才可以在其它优先级上进行工作。联锁条件将不被考虑。仅显示是否附有开关命令。（不管开关命令是否已被执行。）

当用户需要一个已初始化的开关命令的信息时，就需要为每个控制装置连接一个**Command Information**块。

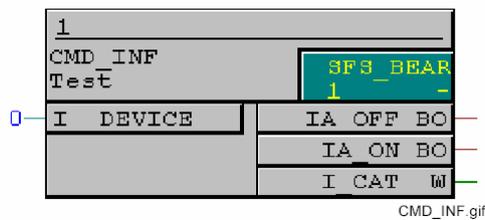


图4-41 CMD_INF块

I/O分配 **CMD_INF**块具有以下的I/O分配:

表4-68 CMD_INF功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	DEVICE	INT	输入信息与功能图的左边界连接。命令对象的名称在此显示。	0
输出	IA_ON	BOOL	当用户发出ON命令时，信息有效	0
	IA_OFF	BOOL	当用户发出OFF命令时，信息有效	0
	I_CAT	WORD	关于命令启动程序的信息	16#0000

I_CAT设置

以下参数用于**I_CAT**的参数设置：

表4-69 **CMD_INF**功能块的值，**I_CAT**设置

数值	意义
00	自动生成的命令
01	集成操作，装置面板控制 (例如来自装置自身)
02	DIGSI、SICAM，本地控制 (例如DIGSI远程，DIGSI总线)
03	远程控制中心，远程控制 (例如WinCC，远程控制站)

4.6.7 LOOP（信号反馈）

功能

signal feedback功能块用于信息的反馈。

在这里，反馈的意思是说某个功能块的输出与同一张CFC功能图中的运行顺序号更小的功能块的输入之间的相互连接。

接收到反馈信号之后，CFC功能图将再次进行处理。在处理时间内到达功能块的其它新的输入信息将被忽略。除了**LOOP**功能块反馈的信号之外，在CFC功能图进行再次处理的过程中的输入信号不会改变。

LOOP功能块可以应用于开关序列（参见**CMD_CHAIN**功能块）：在上一个命令成功执行之后，信号被反馈，从而可以将整个开关序列复位。

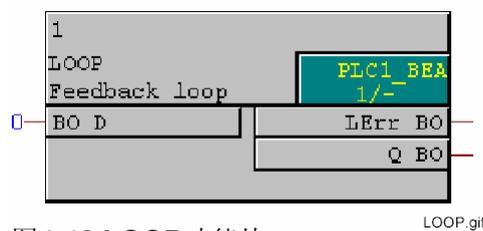


图4-42 **LOOP**功能块



注意：

LOOP功能块只能在下面的优先级上工作；

- 快速PLC处理的优先级（**PLC_BEARB**）
- 慢速PLC处理的优先级（**PLC1_BEARB**）



注意：

如果需要将**CMD_CHAIN**功能块在优先级为慢速PLC处理（**PLC1_BEARB**）的等级上面进行工作，则一定要同时使用**LOOP**功能块，从而可以在命令链条成功执行完最后一个命令后，可以通过**LOOP**功能块将整个操作顺序复位。

I/O分配

LOOP功能块具有以下的I/O分配:

表4-70 **LOOP**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	D	BOOL	位于输入位置D的信号传输到输出位置Q。	0
输出	Q	BOOL	位于输出位置Q的信号传输到输入位置D。	0
	LErr	BOOL	错误输出 如果某个信号在被 LOOP 功能块反馈超过 5 次，输出即为有效	0

**注意:**

为了避免无休止的反馈，信号不能用**LOOP**功能块反馈超过**5**次（输入**D**=输出**Q**）。如果反馈超过了**5**次，将输出错误提示**Lerr**，并且将反馈结束。

4.7 类型转换器

通过类型转换器，用户转换信息项的数据类型。
下面是用户可用的类型转换器：

- ❑ **BOOL_TO_DI** (布尔型到双信号)
- ❑ **BUILD_DI** (生成双信号)
- ❑ **DI_TO_BOOL** (双信号到布尔型)
- ❑ **DINT_TO_REAL** (转换器)
- ❑ **DIST_DECODE** (双信号解码)带有状态信息
- ❑ **DM_DECODE** (双信号解码)
- ❑ **REAL_TO_DINT** (转换器)
- ❑ **REAL_TO_UINT** (转换器)
- ❑ **REAL_TO_INT** (转换器)
- ❑ **INT_TO_REAL** (转换器)
- ❑ **UINT_TO_REAL** (转换器)

4.7.1 BOOL_TO_DI 布尔型到双信号

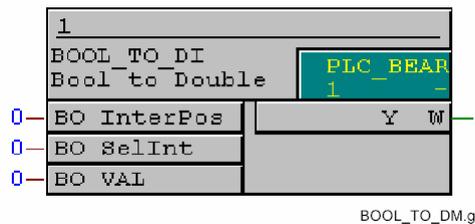


注意:

BOOL_TO_DI功能块属于最初的SIPROTEC4装置中包含的组件，为了能够生成新式的CFC功能图，我们推荐用户采用更加便于使用的**BUILD_DI**功能块。

功能

Boolean to Double-Point Indication功能块可以生成一个双信号。



BOOL_TO_DM.gif

图4-43 **BOOL_TO_DI**功能块



注意:

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界快速信号，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。



注意:

Boolean to Double-Point Indication功能块只能在优先级为快速PLC处理 (**PLC_BEARB**)、慢速PLC处理 (**PLC1_BEARB**)、以及联锁 (**SFS_BEARB**) 的等级上面进行工作。

I/O分配

BOOL_TO_DI功能块具有以下I/O分配:

表4-71 **BOOL_TO_DI**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	SelInt	BOOL	在VAL与InterPos输入之间切换（在开关的端点位置与中间位置之间切换）	0
	VAL	BOOL	值（开关的端点位置）	0
	InterPos	BOOL	中间位置	0
输出	Y	WORD	创建的的双信号	16#0000

生成的双信号

位于输入位置**VAL**与**InterPos**的输入信息将作为**SelInt**功能块的一个函数的形式生成一个名称为**Y**的输出信息。**SelInt**功能块在开关的端点位置与中间位置之间进行切换。

这里采用的缩写形式具有以下含义：

DP的意思是带有不同中间位置的双信号

DP_I的意思是双信号。

表4-72 **BOOL_TO_DI**功能块生成的双信号

SelInt输入	InterPos输入	VAL输入	Y输出	DP状态	DP_I的状态
0	X	0	0001	关闭	关闭
0	X	1	0002	打开	打开
1	0	X	0000	INTERM（中间位置00）	未定义
1	1	X	0003	INTERM（中间位置11）	INTERM

4.7.2 BUILD_DI 生成双信号

功能

Create Double-Point Indication功能块可以通过两个输入信号生成一个双信号。

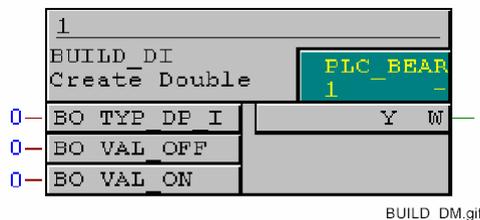


图4-44 BUILD_DI功能块



注意：

Create Double-Point Indication功能块只能在优先级为快速PLC处理（**PLC_BEARB**）、慢速PLC处理（**PLC1_BEARB**）、以及联锁（**SFS_BEARB**）的等级上面工作。



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到CFC功能图左边界的**快速信号**，那么：
当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

BUILD_DI功能块具有下面的I/O分配：

表4-73 BUILD_DI功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	VAL_OFF	BOOL	显示OFF的状态	0
	VAL_ON	BOOL	显示ON的状态	0
	TYP_DP_I	BOOL	显示的双信号的类型	0
输出	Y	WORD	生成的双信号	16#0000

TYP_DP_I设置

在**TYP_DP_I**（生成的双信号的类型）的设置过程中，可以使用下面的数值：

表4-74 **BUILD_DI**功能块在**TYP_DP_I**设置过程中可以使用的数值

数值	意义
0	DP_I（双信号）
1	DP（双信号，位于中间位置）

生成的双信号

位于输入位置**VAL_OFF**与**VAL_ON**的输入信息将作为**TYP_DP_I**功能块的一个函数的形式生成一个名称为**Y**的输出信息。

表4-75 **BUILD_DI**功能块生成的双信号

TYP_DP_I输入	VAL_ON输入	VAL_OFF输入	Y输出	意义
0	0	0	0003	INTERM
0	0	1	0001	关闭
0	1	0	0002	打开
0	1	1	0003	INTERM
1	0	0	0000	INTERM (中间位置00)
1	0	1	0001	关闭
1	1	0	0002	打开
1	1	1	0003	INTERM (中间位置11)

4.7.3 DI_TO_BOOL (双信号到布尔型)

**注意:**

DI_TO_BOOL功能块属于最初的SIPROTEC4装置中包含的组件，为了能够生成新式的CFC功能图，我们推荐用户采用更加便于使用的**DM_DECODE**功能块。

功能

Double-Point Indication to Boolean功能块将对双信号的四种可能的状态中的每一种状态进行检验，并且将检测结果生成一个信号。

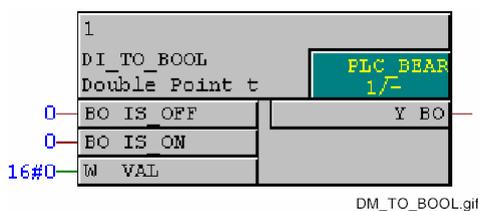


图4-45 DI_TO_BOOL功能块

**注意:**

Double-Point Indication to Boolean功能块只能在优先级为快速PLC处理（**PLC_BEARB**）、慢速PLC处理（**PLC1_BEARB**）、以及联锁（**SFS_BEARB**）的等级上工作。

I/O分配

DI_TO_BOOL功能块具有以下的I/O分配:

表4-76 DI_TO_BOOL功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	IS_OFF	BOOL	查询“关闭”	0
	IS_ON	BOOL	查询“打开”	0
	VAL	WORD	双信号	16#0000
输出	Y	BOOL	结果	0

VAL设置

以下参数用于**VAL**（双信号）的参数设置：

表4-77 **DI_TO_BOOL**功能块的**VAL**参数值

值	DP_I（双信号）的状态	DP（双信号）的状态，位于中间位置
0000	未定义（对应于00）	INTERM（中间位置00），未定义
0001	关闭（对应于01）	关闭（对应于01）
0002	打开（对应于10）	打开（对应于10）
0003	中间（对应于00）	INTERM（中间位置11）

生成的输出信号

IS_ON与**IS_OFF**的输入端的信号可以生成以下作为**VAL**双信号的一个函数的输出Y。

表4-78 **DI_TO_BOOL**功能块生成的输出信号

IS_ON输入	IS_OFF输入	DP_1的VAL输入	DP的VAL输入	Y输出
0	0	未定义	INTERM (中间位置00) 未定义	1
		关闭, 打开 INTERM	关闭 打开 INTERM (中间位置11)	0
0	1	关闭	关闭	1
		打开 INTERM 未定义	打开 INTERM (中间位置00) INTERM (中间位置11) 未定义	0
1	0	打开	打开	1
		关闭 INTERM 未定义	关闭 INTERM (中间位置00) INTERM (中间位置11) 未定义	0
1	1	INTERM 未定义	INTERM (中间位置11)	1
		关闭 打开	关闭 打开 INTERM (中间位置00) 未定义	0

应用实例

使用下面的CFC功能图，用户可以在优先级SFS_BEARB（联锁）上生成下面的联锁状态：关闭circuit breaker，断路器= ON，接地开关= OFF，关闭接地开关，线路断路器= OFF，接地开关= OFF，关闭断路器，线路断路器= OFF，接地开关 = OFF。

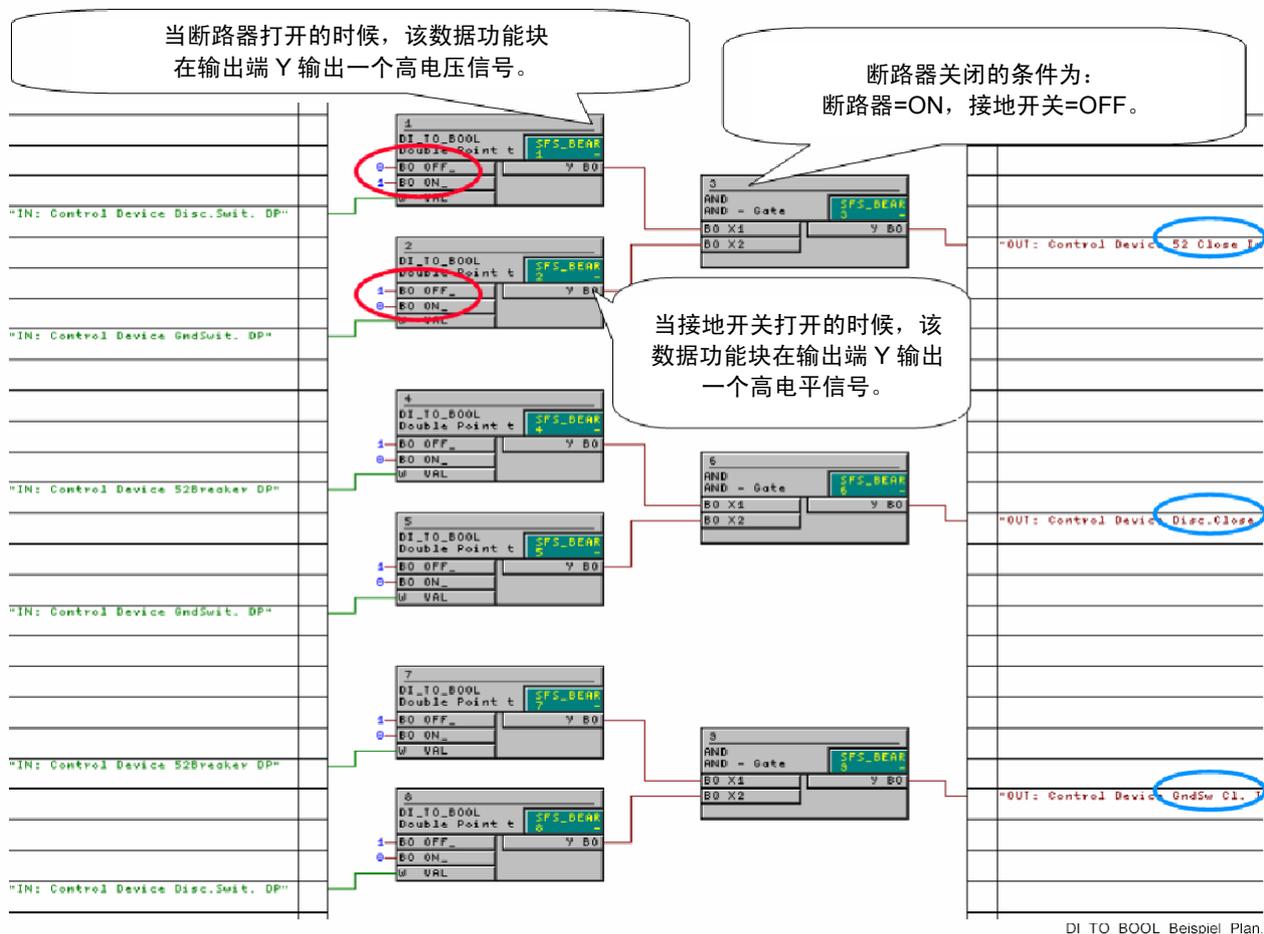


图4-46 DI_TO_BOOL功能块的应用实例

4.7.4 DINT_TO_REAL

功能

DINT_TO_REAL功能块与**REAL_TO_DINT**功能块相反可以将双精度整型数值开关成实型数据。

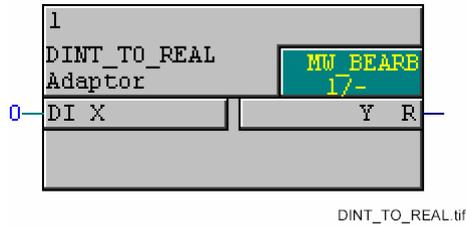


图4-47 **DINT_TO_REAL**功能块

I/O分配

DINT_TO_REAL功能块具有下面的I/O分配：

表4-79 **DINT_TO_REAL**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	DINT	双精度型整型数值的输入规格	0
输出	Y	REAL	实型数值输出规格	0.0



注意：

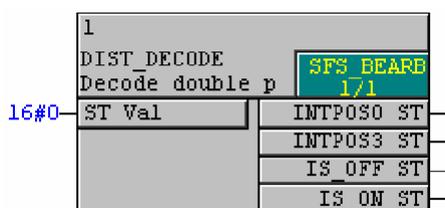
在实型数值中包含的状态信息（参见表4-2）在数值向**DINT**类型转换的过程中保持不变，并且在输出端**Y**输出。

4.7.5 DIST_DECODE

功能

DIST_DECODE功能块可以将一个带有状态信息的双信号转换成四个带有信息的单信号。与**DM_DECODE**功能块的工作方式不同，在**DIST_DECODE**功能块，状态信息传送到输出端之后并没有变化。

IS_ON, **IS_OFF**, **INTPOS0**、以及**INTPOS3**是**DIST_DECODE**功能块可以用的数值，这些数值也可以作为带有状态的输出信息使用。



DIST_DECODE.tif

图4-48 **DIST_DECODE**功能块

I/O分配

DIST_DECODE功能块具有以下I/O分配：

表4-80 **DIST_DECODE**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	VAL	STRUCT	双信号值	0
输出	INTPOS0	SIST	INTERM0（解码双信号），带有状态	0
	INTPOS3	SIST	OFF（解码双信号），带有状态	0
	IS_OFF	SIST	ON（解码双信号），带有状态	0
	IS_ON	SIST	INTERM3（解码双信号），带有状态	0

VAL设置

根据不同的连接情况，**VAL**参数可以包括以下值：

表4-81 **DIST_DECODE**功能块的数值，**VAL**设置

输入值ST值	DP_I输出	DP 输出
0000（中间位置00）	INTPOS3	INTPOS0
0001关闭	IS_OFF	IS_OFF
0002 打开	IS_ON	IS_ON
0003（中间位置11）	INTPOS3	INTPOS3

4.7.6 DM_DECODE

功能

Decode Double-Point Indication功能块可以将一个双信号开关成四个布尔类型的数值。与**DIST_DECODE**功能块的工作方式不同,状态信息将不被传送。

四种可能的取值: **ON**、**OFF**、**INTERM0**、以及**INTERM3**, 可以用来作为该功能块的输出。。

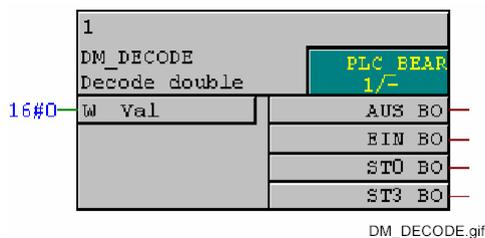


图4-49 DM_DECODE功能块

I/O分配

DM_DECODE功能块具有以下的I/O分配:

表4-82 DM_DECODE功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	VAL	WORD	双信号值	16#0000
输出	ON	BOOL	ON (解码双信号)	0
	OFF	BOOL	OFF (解码双信号)	0
	ST0	BOOL	INTERM0(解码双信号)	0
	ST3	BOOL	INTERM3(解码双信号)	0

VAL设置

根据不同的连接情况，**VAL**参数可以包括以下值：

表4-83 **DM_DECODE**功能块的VAL设置数据

数值	DP_I（双信号）的状态	DP（双信号）的状态，带有不同中间位置	DM_DECODE功能块的输出
0000	未定义	INTERM（中间位置00），未定义	ST0
0001	关闭	关闭	关闭
0002	打开	打开	打开
0003	INTERM	INTERM（中间位置11）	ST3

**注意：**

为了能够处理双信号的状态，用户应该使用**DIST_DECODE**功能块，而不能使用**DM_DECODE**功能块。

4.7.7 REAL_TO_DINT

功能

REAL_TO_DINT功能块可以将实型数值转换为双精度整型数据。

数值采用四舍五入的方法：对于一个包含的.5的数据，将进位到下一个最大的整数。

如果输出数据范围超出上限或者超出下限则输出数据的最大或者最小值并将**ERR**输出设置。

在输入的数据重新符合输出数据**Y**的范围要求之前，错误提示**ERR**将保持不变。

如果在输入端**X**没有有效的实型数据，将在输出端**Y**输出**0**，同时输出错误提示**ERR**。

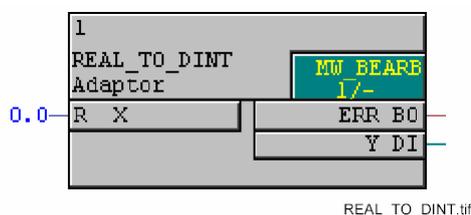


图4-50 REAL_TO_DINT功能块

I/O分配

REAL_TO_DINT功能块具有以下I/O分配：

表4-84 REAL_TO_DINT功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	REAL	实型数据的输入规格	0.0
输出	Y	DINT	双精度整型数值输出规格	0
	ERR	BOOL	输出数据范围的上越界或者下越界	0



注意：

在实型数值中包含的状态信息（参见表4-2）在数值向**DINT**类型转换的过程中保持不变，并且在输出端**Y**输出。

4.7.8 REAL_TO_INT

功能

REAL_TO_INT功能块可以将实型数值转换为整型数据。它允许将整定值（极限值）连接至计时器，这样在现场进行装置操作时，计时器可以被置位。

数值采用四舍五入的方法：对于一个包含的.5的数据，将进位到下一个最大的整数。

如果在输入端**X**没有有效的实型数据，将在输出端**ERR**将被置位，输出端**Y**将按下表进行输出：

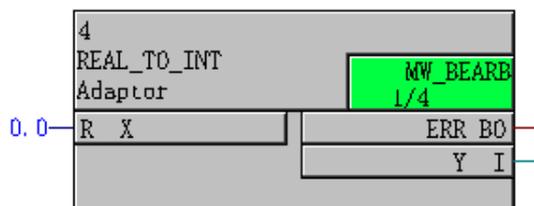


图4-51 REAL_TO_INT功能块

I/O分配

REAL_TO_DINT功能块具有以下的I/O分配：

表4-85 REAL_TO_DINT功能块的I/O分配

X输入端状态	输出Y	ERR输出
OVERFLOW	65535	1
OVERFLOW_NEG	0	1
LIVE_ZERO	Y(n-1) 没有变化	1
NOT_DEFINED	Y(n-1) 没有变化	1
NOT_CALCULATED	Y(n-1) 没有变化	1
INVALID	Y(n-1) 没有变化	1



注意：

REAL_TO_INT功能块属于最初的SIPROTEC4装置中包含的组件，为了能够生成新式的CFC功能图，我们推荐用户采用更加便于使用的**REAL_TO_UINT**功能块。

4.7.9 REAL_TO_UINT

功能

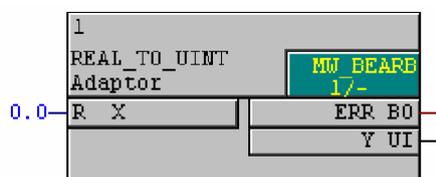
REAL_TO_UINT功能块可以将实型数值转换成无符号整型数值。例如，这个功能块将一个定点值（极限值）与计时器的输入时间相互连接，这样计时器在装置运行过程中也可以通过现场控制装置进行设置。

数据采用四舍五入的方法：对于一个包含的.5的数值，将进位到下一个最大的整数。

如果在输入端**X**没有有效的实型数据，输出**ERR**将被设置，输出数据**Y**将根据下面的功能图中的规定进行进一步的处理：

表4-86 REAL_TO_UINT功能块的输出情况

输入位置X的状态	输出Y	输出ERR
OVERFLOW	65535	1
OVERFLOW_NEG	0	1
LIVE_ZERO	Y _(n-1) 没有变化	1
NOT_DEFINED	Y _(n-1) 没有变化	1
NOT_CALCULATED	Y _(n-1) 没有变化	1
INVALID	Y _(n-1) 没有变化	1



REAL_TO_UINT.tif

图4-52 REAL_TO_UINT功能块

I/O分配

REAL_TO_UINT功能块具有下面的I/O分配:

表4-87 **REAL_TO_UINT**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	REAL	实型数据的输入规格	0.0
输出	Y	UNIT	无符号整型数据输出规格	0
	ERR	BOOL	输出数据范围的上越界或者下越界	0

4.7.10 INT_TO_REAL

功能 **INT_TO_REAL** 功能块可以将双精度整型数值转换为实型值。功能与 **REAL_TO_INT**功能块相反。

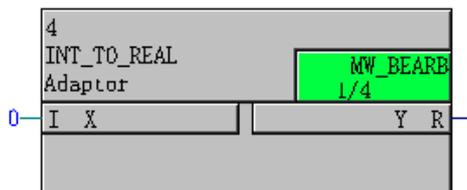


图4-53 INT_TO_REAL功能块

I/O分配 **INT_TO_REAL**功能块具有以下的I/O分配：

表4-88 INT_TO_REAL功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	UINT	无符号整型数据输出规格	0
输出	Y	REAL	实型数据的输入规格	0.0

**注意：**

INT_TO_REAL功能块属于最初的SIPROTEC4装置中包含的组件，为了能够生成新式的CFC功能图，我们推荐用户采用更加便于使用的**UNIT_TO_REAL**功能块。

4.7.11 UINT_TO_REAL

功能

UINT_TO_REAL功能块与**REAL_TO_UINT**相反可以将无符号整型数据转换成实型数据。



图4-54 **UINT_TO_REAL**功能块

I/O分配

UINT_TO_REAL功能块具有以下的I/O分配：

表4-89 **UINT_TO_REAL**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	X	UNIT	无符号整型数据的输入规格	0
输出	Y	REAL	实型数据输出规格	0.0

4.8 数据比较

通过使用比较功能块，用户能够比较，或在一定限度内修正测量值。

下面是用户可以使用的功能块：

- COMPARE** (测量值比较)
- LIVE_ZERO** (零漂监视)
- LOWER_SETPOINT** (下限)
- UPPER_SETPOINT** (上限)
- ZERO_POINT** (清零)

4.8.1 COMPARE （测量值比较）

功能

measured value comparison功能块可以通过使用滞后量HYSVAL将两个实型数据VAL1与VAL2进行比较，比较结果可以为大于，小于或等于。

通过使用数据类型转换器（比如DINT_TO_REAL），用户可以进行更多种类数据的比较，比如可以将计量数据进行比较。比较结果将分别输出到名称为**EQUAL**、**GREATER**、或者**LESS**的输出位置，输出类型为**BOOL**型。



图4-55 COMPARE功能块

I/O分配

COMPARE功能块具有以下的I/O分配：

表4-90 COMPARE功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	HYSVAL	REAL	滞后量数据	0.0
	VAL1	REAL	比较对象1	0.0
	VAL2	REAL	比较对象2	0.0
输出	EQUAL	BOOL	数据相等 (VAL1=VAL2)	0
	GREATER	BOOL	比较对象1大于比较对象2 (VAL1>VAL2)	0
	LESS	BOOL	比较对象1小于比较对象2 (VAL1<VAL2)	0

当HYSVAL=0时的输出形态

当滞后量数据HYSVAL=0时（两个数据的简单比较）时，输出情况如下所示：

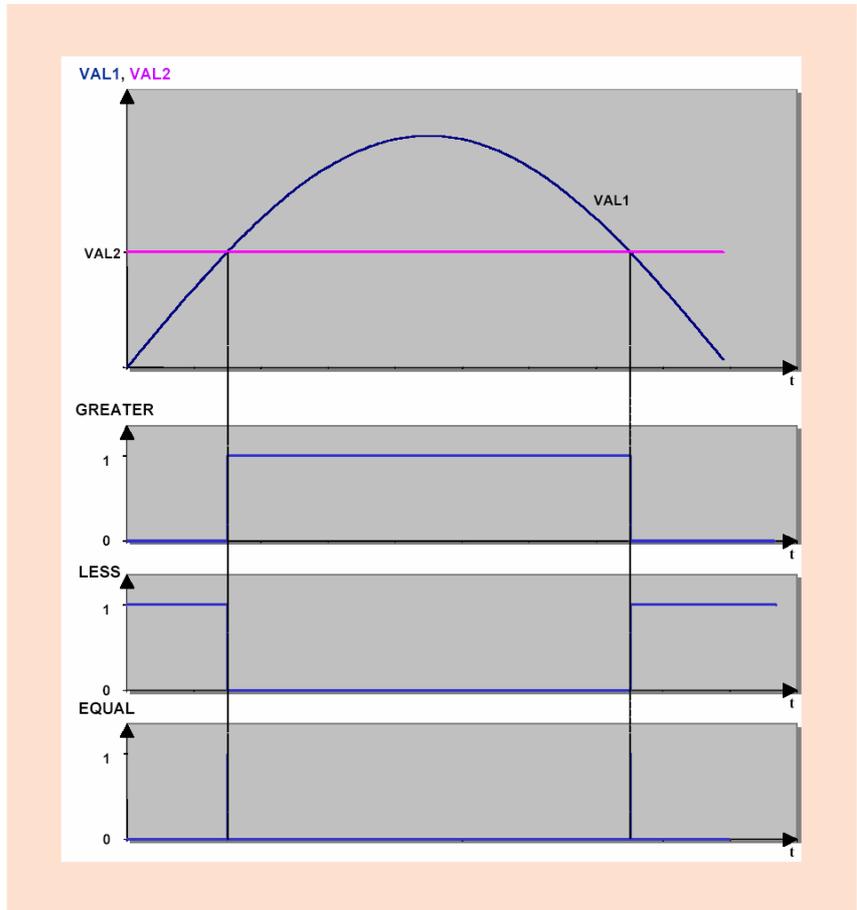


图4-56 滞后量数据HYSVAL=0时（两个数据的简单比较）时的比较图

当HYSVAL>0时的输出情况

当滞后量数据HYSVAL>0时（滞后量数据在零点附近）时，输出情况如下所示：

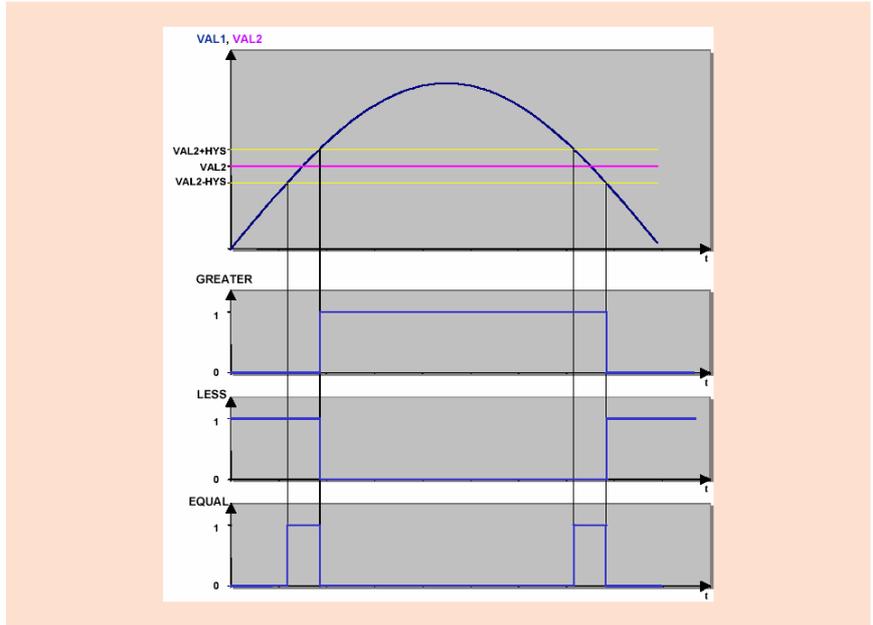


图4-57 滞后量数据HYSVAL>0时（滞后量数据在零点附近）时的比较图

当HYSVAL<0时的输出情况

当滞后量数据HYSVAL<0时（延迟回动点的比较）时，输出情况如下所示：

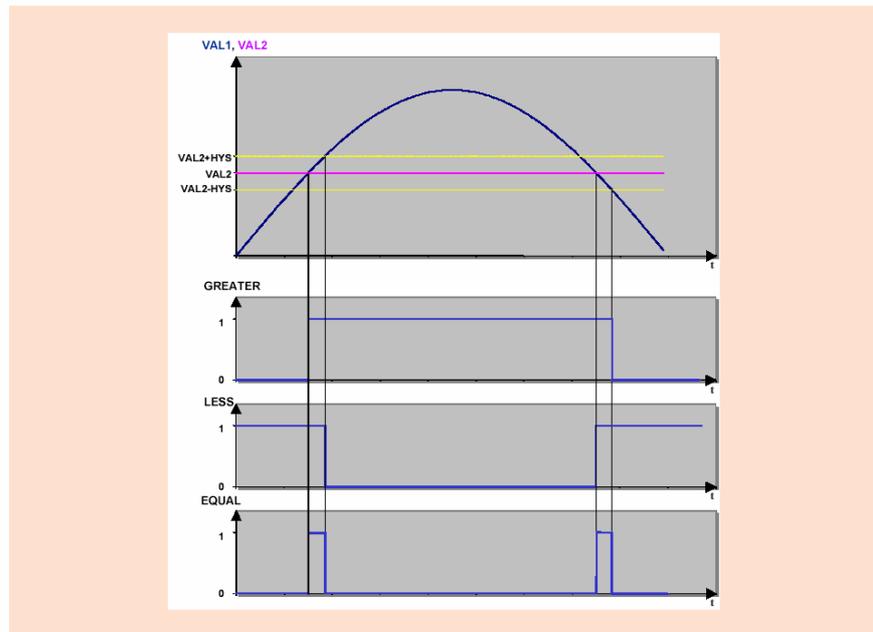


图4-58 滞后量数据HYSVAL<0时（延迟回动点的比较）时的比较图

4.8.2 LIVE_ZERO （零漂监视）

功能

为了检测传感器与变频器间的传感故障或线路故障，测量值将不能在0%与100%之间变化，只能在零漂值（通常20%）——100%间。小于零漂值将被认为是一种故障，**Live Zero**功能块将生成一个错误提示信息。

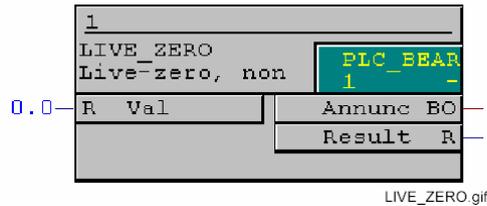


图4-59 LIVE_ZERO功能块



注意：

LIVE_ZERO, **LOWER_SETPOINT**, **UPPER_SETPOINT**功能块只能在测量值处理的优先级（**MW_BEARB**）上进行工作。

I/O分配

LIVE_ZERO功能块具有以下的I/O分配：

表4-91 **LIVE_ZERO**功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Val	REAL	测量值in %	0.0
	LiveZero	REAL	零漂值 (%) ($\geq 0.0\%$)	20.0
	DetecKnee	REAL	检测的拐点 (%) ($\geq 0.0\%$ 且 $< 100.0\%$)	50.0
	DispKnee	REAL	显示的拐点 (%) ($0.0 \dots 200.0\%$)	70.0
输出	Result	REAL	零漂测量值 (%)	0.0
	Annunc	BOOL	信号：零漂监视	0

**注意:**

LiveZero、**DetecKnee**、以及**DispKnee**输入端的设置可以在功能块的关联菜单**对象属性**中进行配置。

**注意:**

为了防止**Live Zero Monitoring**的信息出现抖动，通常会在生成此信息时使用至少5%的复位率为0.95的滞后。

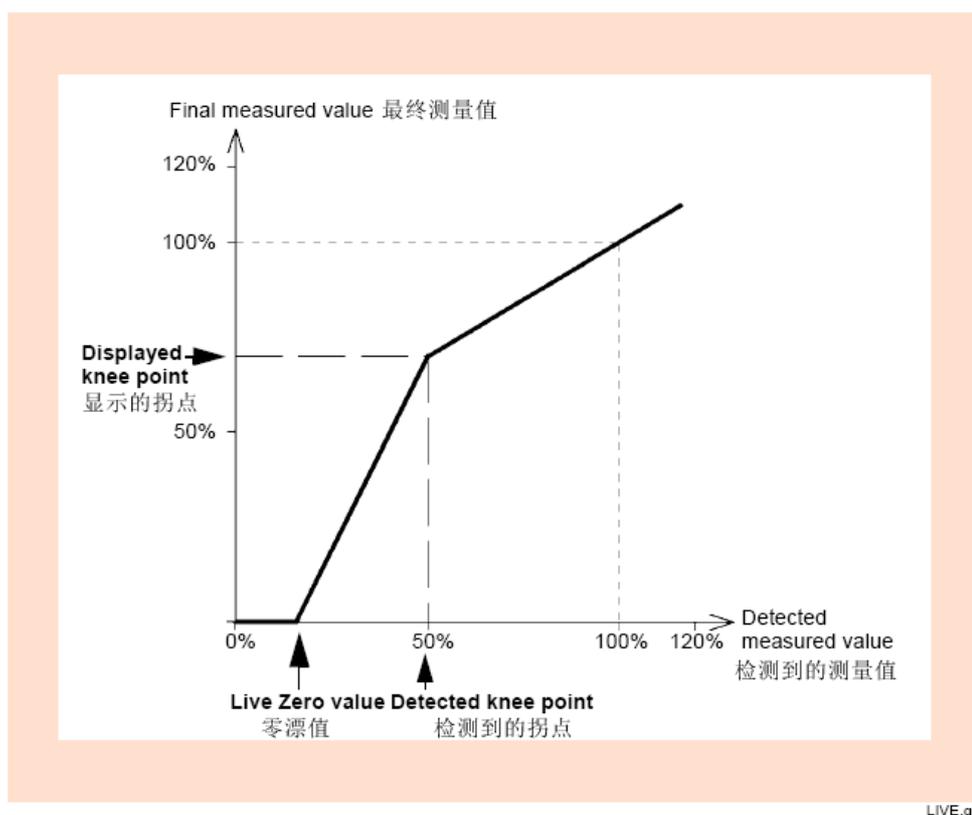


图4-60 零漂监测图

**注意:**

如果用户没有正确配置**LIVE ZERO**功能块的，也没有设置**LiveZero**、**DetecKnee**、以及**DispKnee**的允许输入范围，系统将在**Result**输出**NOT_CALCULATED**的提示信息。这个信息将在SIPROTEC设备的显示界面上以三个点...显示出来。

4.8.3 LOWER_SETPOINT （下限设定点）



注意:

LIVE_ZERO, **LOWER_SETPOINT**, **UPPER_SETPOINT**功能块只能在测量值处理的优先级（**MW_BEARB**）上进行工作。

功能

如果**Val**输入端的值低于**LIMIT**端的值，那么**Annunc**的输出端产生一个信号。

用户可以将下限输入端和另一个功能块的输出端相连，或者直接给它定一个值。

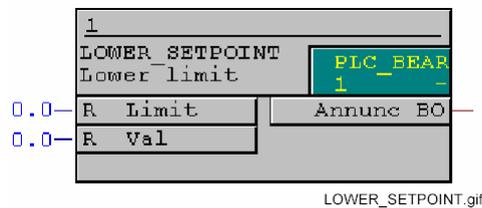


图4-61 LOWER_SETPOINT功能块

I/O分配

LOWER_SETPOINT功能块具有以下的I/O分配:

表4-92 LOWER_SETPOINT功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Limit	REAL	极限值 (%) ($\pm 1.0e^{+38}$)	0.0
	Val	REAL	测量值 (%) ($\pm 1.0e^{+38}$)	0.0
输出	Annunc	BOOL	信号 下限	0



注意:

为了防止信息**Lower Limit**抖动，通常会在生成此信息时使用至少0.5%的复位率为0.95的滞后。



注意:

在**VAL**输入端输入一个标记为**FlagOverflow**的值会导致**Annunc**输出端输出为0。

应用实例

可以使用下面的CFC功能图对三相电流进行监测：
如果三相电流全部小于额定电流的5%，则切换定值组。

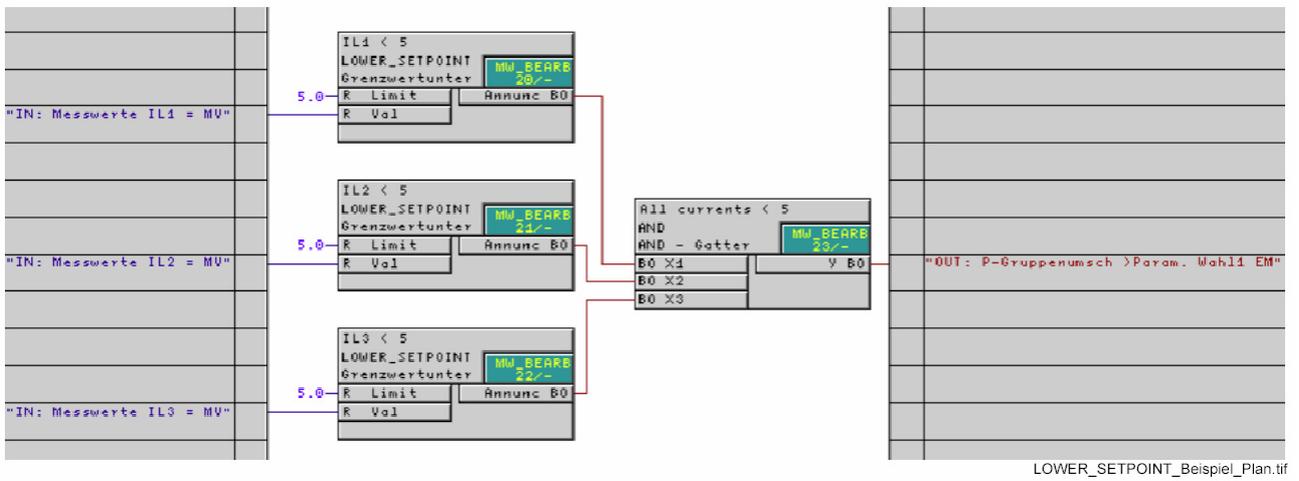


图4-62 LOWER_SETPOINT功能块的应用实例，CFC功能图

4.8.4 UPPER_SETPOINT （上限设定点）

**注意：**

LIVE_ZERO, **LOWER_SETPOINT**, **UPPER_SETPOINT**功能块只能在测量值处理的优先级（**MW_BEARB**）上进行工作。

功能

如果**Val** 输入端的值高于LIMIT端的值，那么**Annunc**的输出端将输出一个信号。

用户可以将上限输入端和其它功能块的输出信号相连，也可以指定一个固定值。

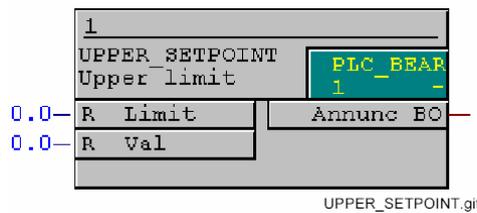


图4-63 UPPER_SETPOINT功能块

I/O分配

UPPER_SETPOINT功能块具有以下的I/O分配：

表4-93 UPPER_SETPOINT功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Limit	REAL	极限值（%） ($\pm 1.0e^{+38}$)	0.0
	Val	REAL	测量值（%） ($\pm 1.0e^{+38}$)	0.0
输出	Annunc	BOOL	信号 下限	0

**注意：**

为了防止信息**Upper Limit**抖动，通常会在生成此信息时使用至少5%的复位率为0.95的滞后。

**注意：**

在VAL输入端输入一个标记为**OVERFLOW**的值会导致**Annunc**输出端输出为1。

4.8.5 ZERO_POINT (零点)



注意:

LIVE_ZERO, LOWER_SETPOINT, UPPER_SETPOINT功能块只能在测量值处理的优先级 (**MW_BEARB**) 上进行工作。

功能

由于测量误差，测量到的零值点可能与准确的零值点存在一个轻微的偏差。为了弥补这个偏差的影响，可以将一个低于可测定值的测量值设为零。上述工作可以通过**Zero Suppression**功能块来完成。

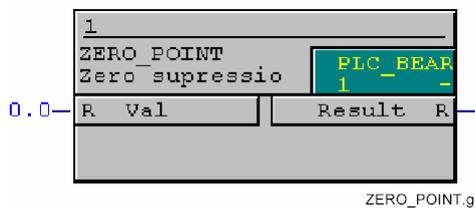


图4-64 ZERO_POINT功能块

I/O分配

ZERO_POINT功能块具有以下的I/O分配:

表4-94 ZERO_POINT功能块的I/O分配

	名称	数据类型	注释	默认设置
输入	Val	REAL	测量值 in %	0.0
	Zero-p oint	REAL	零值点 (%) (0.0—30.0%)	5.0
输出	Result	REAL	已消零测量值 (%)	0.0



注意:

ZeroPoint输入端的设置可以通过功能块的关联菜单**对象属性**中进行配置。

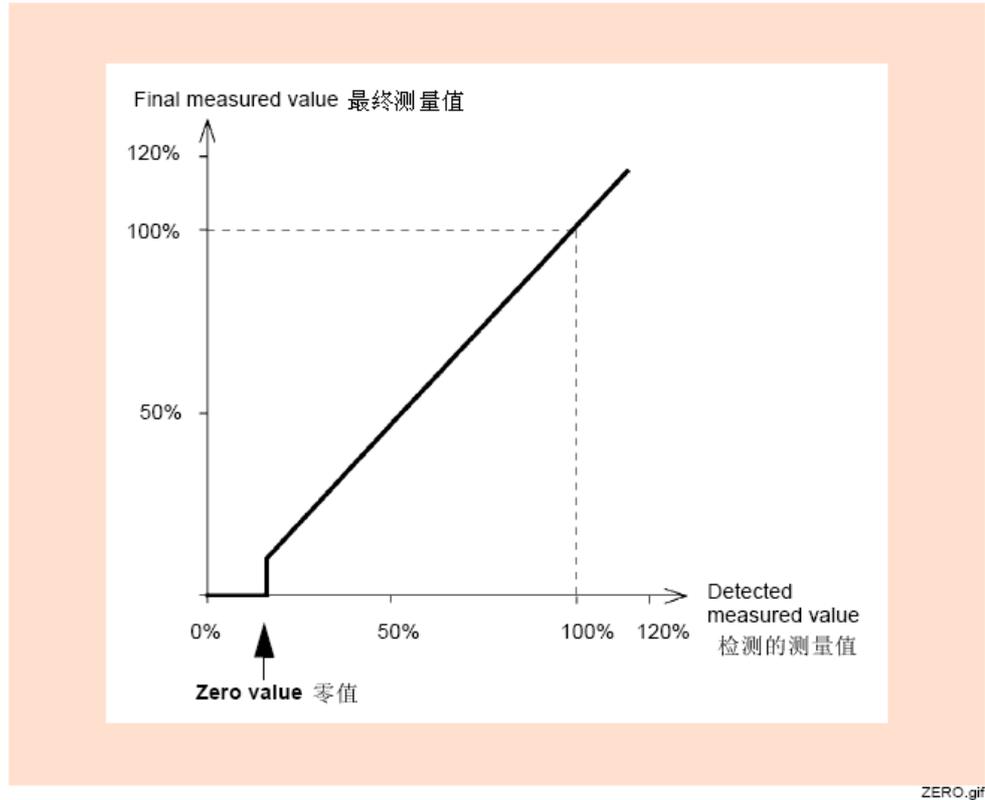


图4-65 ZeroPoint清零

**注意:**

如果用户没有正确配置**ZERO_POINT**功能块，也没有设置**Val**输入端 或 **ZeroPoint**的输入端，那么**Result**将输出**NOT_CALCULATED**，这个信息将在SIPROTEC设备的显示界面上以三个点...显示出来

。

4.9 计量值

用户可以通过计量值功能块完成计数器的设计工作。

下面是用户可以使用的计量数据功能块：

□ COUNTER

4.9.1 COUNTER

功能

COUNTER功能块对每次**CLK**的上升沿信号输入进行计数，并且累加**DELTA**值，放在保存值中。

在系统首次启动的时候，当**BEGIN**输入端中的有效数据计数器功能块的状态**VAL**进行初始化。

在系统重新启动过程中，将使用计数器状态**VAL**值。

如果计数器的状态超出了**END**值规定的范围，计数器将通过**BEGIN**输入中的有效数据进行初始化，并且输出**OVERFLOW**将从0开关到1，表明存在数据溢出现象。溢出信号会一直保持到**CLK**信号再次有效（上升脉冲）或者设备重启为止。

通过**RESET**输入中的上升脉冲，可以将计数器在**BEGIN**输入中设置有效数据。



图4-66 COUNTER功能块

COUNTER.tif



注意：

可以使用的**RS_FF_MEMO**功能块、**SR_FF_MEMO**功能块、**D_FF_MEMO**功能块、**COUNTER**功能块的最大数量取决于系统可以使用的非易失存储器的容量，并且这个最大数量将通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，可以查看设备用户手册中的具体技术参数数据。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。

**注意：**

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

I/O分配

COUNTER功能块具有下面的I/O分配情况：

表4-95 **COUNTER**功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	BEGIN	DINT	计数器初始值	0
	CLK	BOOL	信号是否上升沿跳变	0
	DELTA	DINT	计数器的步长	0
	END	DINT	计数器的最大值	2147483631
	RESET	BOOL	将计数器的数值恢复到初始状态	0
输出数据	VAL	DINT	计数器的当前状态	保存数据
	OVERFLOW	BOOL	计数器溢出，表明已经超出了计数器的最终数据	0

**注意：**

如果在**BEGIN**、**DELTA**、或者**END**输入中存在超出数据范围的有效数据（**DINT**类型数据的范围是从**-2147483631** 到**2147483631**），输出**VAL**将被设置为**NOT_CALCULATED**（**OVERFLOW = 0**）。输入**CLK**和**RESET**将不被计算，内部计量数据也不会改变。

当**BEGIN**、**DELTA**、或者**END**中存在新的有效数据时，在输出位置**VAL**的内部计量数据将被输出。

4.10 计时器和时钟

用户可以通过计时器和时钟功能块对基于时间的功能进行控制。

下面是可以使用的计时器和时钟功能块：

- ALARM** (报警)
- BLINK** (闪烁功能块)
- LONG_TIMER** (长型计时器 (最大计时长度1,193小时))
- TIMER** (通用计时器)
- TIMER_SHORT** (简易计时器)

4.10.1 ALARM

功能

Alarm功能块表明在输入数据中规定的报警时间已经来到，同时将**Q**输出的数值从**0**改变为**1**。

在匹配持续时间（1秒钟）内，**Q**保持不变。

通过使用占位符，用户可以完成周期性启动的时间说明的设计工作。

如果没有指定有效时间（例如**2月31**），将输出一个错误提示**ERR**。在重新指定有效的之前，错误提示**ERR**一直保持。



注意：

不必考虑时间不一致（例如从冬令时到夏令时）。如果时间位于这个时间框架内，将没有双重启动发生，或者只有一次发生。

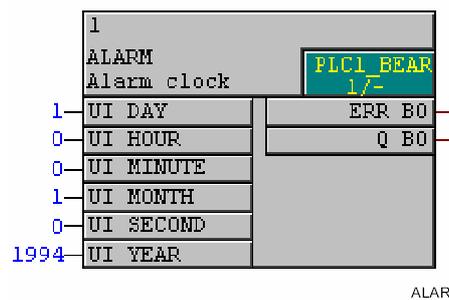


图4-67 **ALARM**功能块

**注意:**

如果在测量值处理 (MW_BEARB) 和联锁 (SFS_BEARB) 优先级中连接到CFC功能图左边界的**快速信号**, 那么:

当信号发生时间小于处理周期时, 布尔量输入不会作为事件触发, 输入将被忽略。

I/O分配

ALARM功能块具有下面的I/O分配情况:

表4-96 **ALARM**功能块的I/O分配情况

	名称	数据类型	注释	占位符	默认设置
输入数据	DAY	UNIT	天	0	1
	HOUR	UNIT	小时	24	0
	MINUTE	UNIT	分钟	60	0
	MONTH	UNIT	月	0	1
	SECOND	UNIT	秒	无	0
	YEAR	UNIT	年	0	1994
输出数据	ERR	BOOL	无效日期		0
	Q	BOOL	报警时间已到(1秒钟)		0

4.10.2 BLINK

功能

BLINK功能块是专门为了将信号编码成光信号（例如**指示灯**）而设计的。为此，这个功能块为每个光相与暗相的时间设置规定了一个输入数据。

当**ENABLE**输入端的值从**0**跳变到**1**，就启动闪光功能块。同时在输出位置**Q**，信号从**0**改变为**1**，根据设置时间的不同，也可能从**1**改变为**0**。**ENABLE**值从**1**改变到**0**，就可以将闪光功能块关闭。



注意：

闪光计时器的精度是**100ms**。按照这种方式，根据闪烁的指示灯的首次启动时间，第**1**个光亮可以比设置数据**THx100ms**多出**99ms**。

最短的循环周期是**100ms**。不管是对光亮**THx100ms**还是光灭**TLx100ms**的周期时间，都不能少于这个时间。如果周期时间小于**100ms**，系统将自动采用**100ms**的时间。

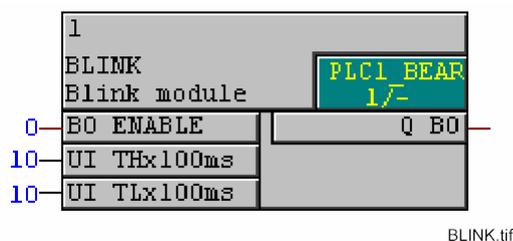


图4-68 **BLINK**功能块

I/O分配

BLINK功能块具有下面的I/O分配情况：



注意：

如果在测量值处理（**MW_BEARB**）和联锁（**SFS_BEARB**）优先级中连接到**CFC**功能图左边界的**快速信号**，那么：

当信号发生时间小于处理周期时，布尔量输入不会作为事件触发，输入将被忽略。

表4-97 **BLINK**功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	ENABLE	BOOL	释放 启动或者关闭闪光功能块	0
	THx100ms	UNIT	光亮时间 (精度为100ms)	10 (=1s)
	TLx100ms	UNIT	光灭时间 (精度为100ms)	10 (=1s)
输出数据	Q	BOOL	运行启动之后, 根据亮/灭在数据1与数据0之间切换	0

4.10.3 LONG_TIMER

功能

用户可以通过**Long Timer**功能块完成按照小时计算的时延工作。

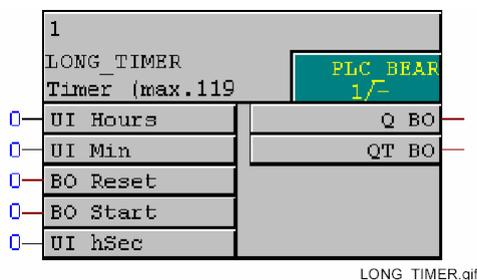


图4-69 LONG_TIMER功能块



注意:

如果在测量值处理 (**MW_BEARB**) 和联锁 (**SFS_BEARB**) 优先级中连接到CFC功能图左边界的**快速信号**, 那么:

当信号发生时间小于处理周期时, 布尔量输入不会作为事件触发, 输入将被忽略。

I/O分配

LONG_TIMER功能块具有下面的I/O分配情况:

表4-98 LONG_TIMER功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	Hours	UNIT	按照小时计算的数值	0
	Min	UNIT	按照分钟计算的数值	0
	Reset	BOOL	复位计时器	0
	HSec	UNIT	按照秒计算的数值 (精度100ms)	0
	Start	BOOL	启动或者重新启动计时器 (位于0-1变迁)	0
输出数据	Q	BOOL	计时器已经超时	0
	QT	BOOL	计时器仍在运行	0

重启长型计时器

Long Timer功能块可以通过下面的方式启动：

在**Start**输入中使用一个新的从0到1的改变，重新配置时间，从而可以将正在运行的计时器取消或者重新启动。

复位输入

当前计时器可以通过在复位输入**Reset**的位置输入一个信号来取消。输出信号**Q**和**QT**将被设置给数据0。



注意：

下面的规定适用于V4.5以下版本的SIPROTEC装置。

最大延迟时间为1193小时。

如果用户在参数**Hours**、**Min**、或者**Sec**中输入的时间多于1193小时，将在SIPROTEC装置中输出一个内部错误信号，并且自动将运行时间设置为0。

4.10.4 TIMER

功能

通过**Universal timer**功能块，用户可以实现下面的计时器功能：

- 通用计时器功能
- 可再触发式单拍计时器功能
- 延迟计时器功能
- 脉冲展宽计时器功能

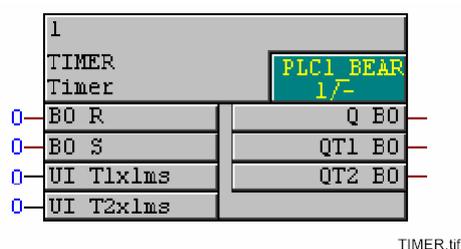


图4-70 **TIMER**功能块



注意：

Universal timer功能块只能在下面的优先级上工作：

- 快速PLC处理的优先级（**PLC_BEARB**）
- 慢速PLC处理的优先级（**PLC1_BEARB**）



注意：

Universal Timer功能块（**TIMER**）可以用**simple timer (TIMER_SHORT)**功能块进行替换。

可以使用的**TIMER**功能块或者**TIMER_SHORT**功能块最大数量取决于系统可以使用的计时器数量，并且这个最大数量将通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，应该遵守该装置手册中的有关规定。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。

I/O分配

TIMER功能块具有下面的I/O分配情况：

表4-99 **TIMER**功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	S	BOOL	设置输入	0
	R	BOOL	复位输入	0
	T1x1ms	UNIT	T1的数值（精度1ms）	0
	T2x1ms	UNIT	T2的数值（精度1ms）	0
输出数据	Q	BOOL	Q输出	0
	QT1	BOOL	T1输出	0
	QT2	BOOL	T2输出	0



注意：

T1x1ms和**T2x1ms**的最小许用时间值和使用的SIPROTEC装置的时间精度有关。如果使用了小于许用精度的时间设置，那么即使计时器收到启动脉冲信号，仍将不运行。详情请遵守该装置手册中的有关规定。

各个计数器的功能和线路连接情况在下面进行说明：

复位输入

复位输入在各个计时器中的操作方式相同（在图中不再进行说明）

当输入 **Reset** 信号，所有运行的计时器停止。输出数据 **QT1** 与 **QT2** 被设置给数据 **0**。位于输入位置 **S** 的信号将直接被映射到输出 **Q**。

标准计时器功能

下面的情况适用于标准计时器：

如果位于输入 **S** 的信号具有有效上升沿，**T1** 启动。如果 **T1** 到期，并且在输入 **S** 存在一个信号，信号将被输出到 **Q**。当信号在 **Q** 被输出，时间 **T2** 可以通过位于输入 **S** 的信号的下沿启动。时间是可以重新触发的。

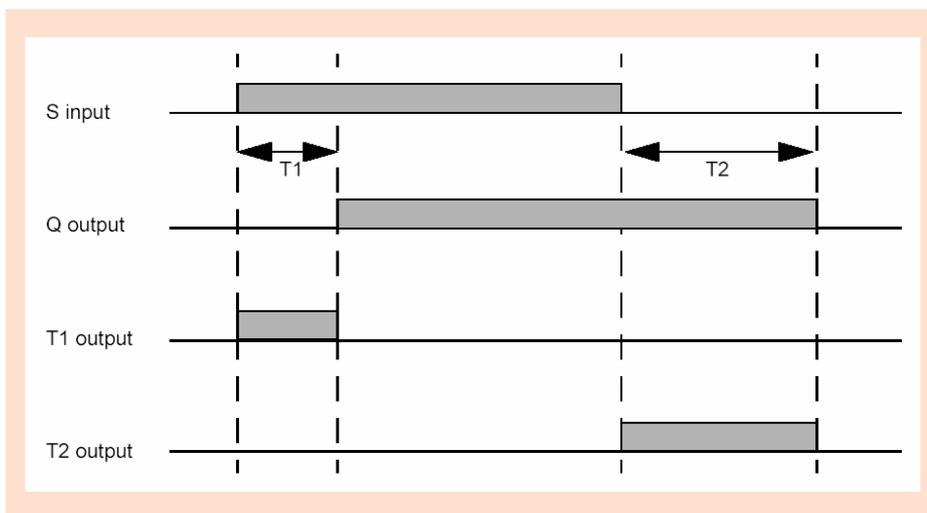


图4-71 计时器功能描述图

可触发式单拍计时器功能 下面的情况适用于可触发式单拍计时器：

T2配置给0。只使用输出**T1**。

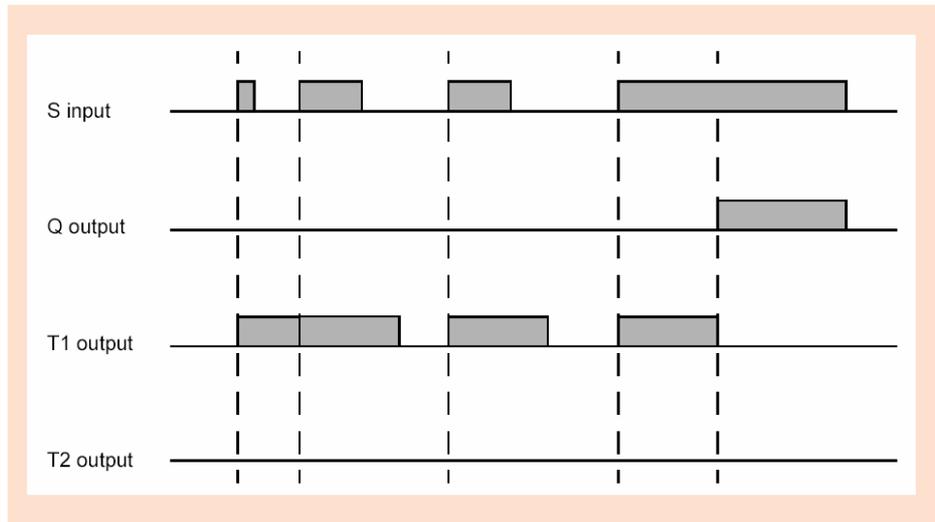


图4-72 可触发式单拍计时器功能描述图

延迟计时器功能 下面的情况适用于延迟计时器：

T2配置给0。只使用输出**Q**。

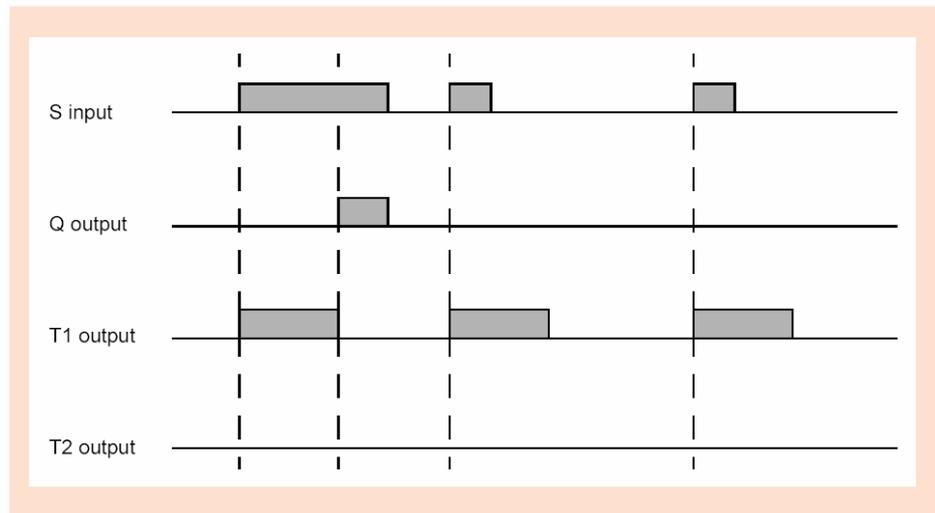


图4-73 延迟计时器功能描述图

脉冲展宽计时器功能

下面的情况适用于脉冲展宽计时器：

T2配置为0。

只使用输出**Q**。

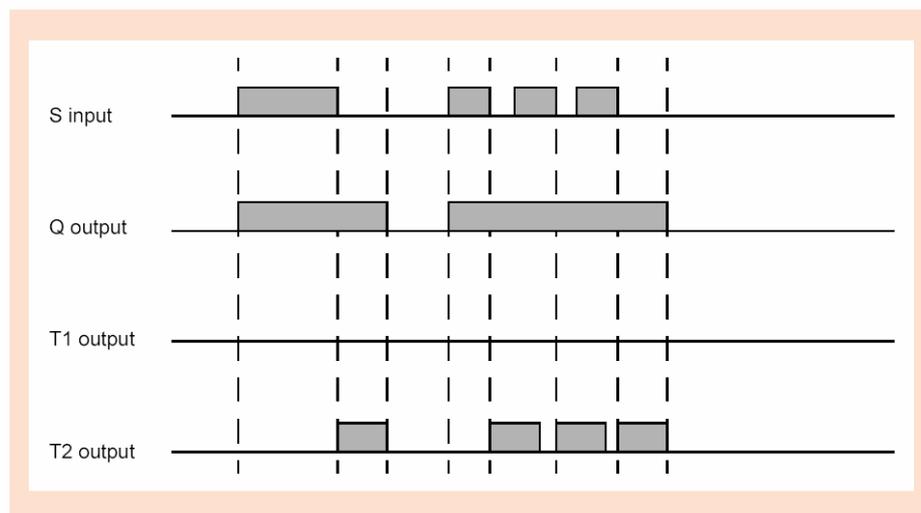


图4-74 脉冲展宽计时器功能描述图

应用实例

通过下面的CFC功能图，用户可以完成一个简单的操作序列：

如果**F1**键被按下，**LED1**将发光5秒钟。

然后**LED2**将发光15秒钟。

- 根据下面的功能图，在配置矩阵中插入所需要的信息：

Settings - Masking I/O (Configuration Matrix) - test1 / 文件夹 / 75J636 V4.6/75J636																
	信息			信息来源				信息目标								
	编号	显示文本	L	类型	BI	F	s	C	开关	LE	缓冲器			C	D	CM
											O	s	T			
LED Flash		F1 Flash		IntSP		1					00			X		
		F2 No Flas		IntSP		2					00			X		
		LED Flash		IntSP				X			00					
start driver										*				*		

图4-75 计时器时钟的应用实例，配置矩阵部分

- 在**PLC1_BEARB**（慢速PLC处理）的优先级上运行下面的CFC功能图：

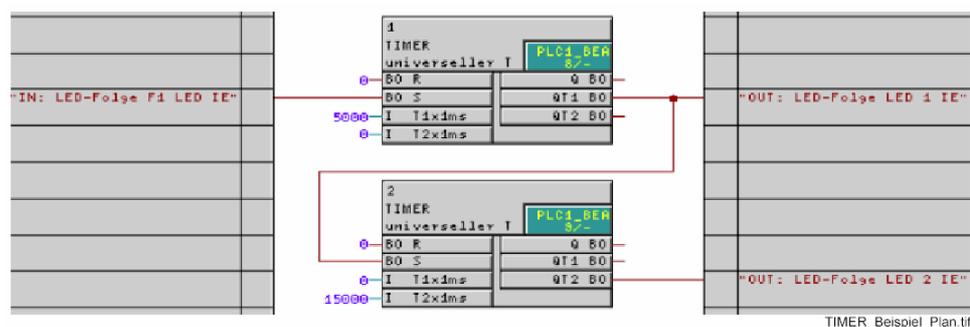


图4-76 计时器时钟的应用实例，CFC功能图部分

4.10.5 TIMER_SHORT

功能

通过**simple timer**功能块，用户可以完成简单的计时功能（例如时间延迟）。在这个功能块中，最多可以设置65535秒的时间，精度为1ms。

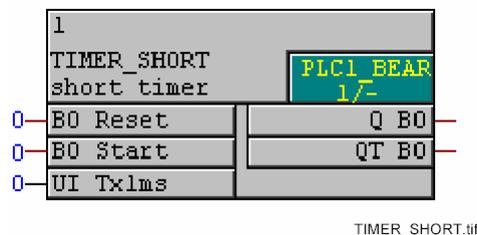


图4-77 TIMER_SHORT功能块



注意：

simple timer功能块只能在下面的优先级上工作：

- 快速PLC处理的优先级（PLC_BEARB）
- 慢速PLC处理的优先级（PLC1_BEARB）



注意：

simple timer(TIMER_SHORT)功能块可以作为**Universal Timer**功能块（**TIMER**）的替代功能块使用。

可以使用的**TIMER**功能块或者**TIMER_SHORT**功能块最大数量取决于系统可以使用的计时器的数量，并且这个最大数量将通过**CFC**编译器进行监测。用户在使用某种**SIPROTEC**装置的过程中，应该遵守该装置手册中的有关规定。

在编译**CFC**功能图的过程中，上述的功能块最大使用数量可以得到检测。如果有错误发生，系统将显示一致性错误的提示信息。对于系统资源过多使用的情况将显示在**CFC**功能图的编译日志中。

I/O分配

TIMER_SHORT功能块具有下面的I/O分配情况：

表4-100 TIMER_SHORT功能块的I/O分配情况

	名称	数据类型	注释	默认设置
输入数据	Reset	BOOL	复位输入	0
	Start	BOOL	设置输入	0
	Tx1ms	UNIT	时间数据（精度1ms）	0
输出数据	Q	BOOL	时间已经结束	0
	QT	BOOL	时间正在运行	0



注意：

T1x1ms和**T2x1ms**的最小许用时间值和使用的SIPROTEC装置的时间精度有关。如果使用了小于许用精度的时间设置，那么即使计时器收到启动脉冲信号，仍将不运行。详情请遵守该装置手册中的有关规定。

触发简易计时器

simple timer功能块可以按照这样的方式触发：

在**Start**输入中使用一个新的从0到1的改变，从而可以将正在运行的计时器取消或者重新启动。

复位输入

当前计时器可以通过在复位输入**Reset**的位置输入一个信号来取消。输出信号**Q**和**QT**将被设置给数据0。

计时器功能：

下图描述了TIME_SHORT的功能原理

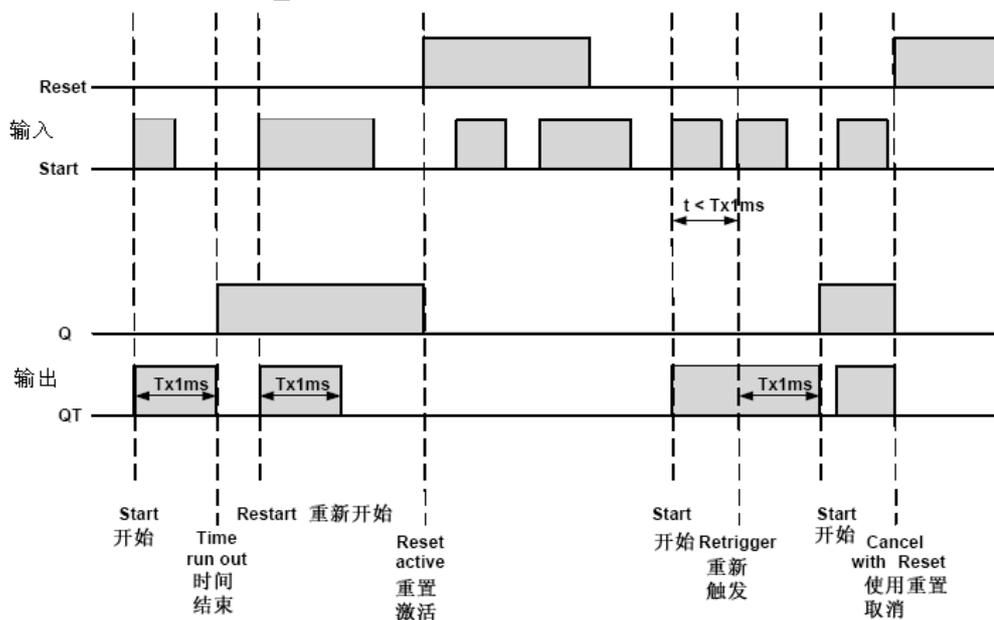


图4-78 TIMER_SHORT原理图

文献

- (1) SIPROTEC, 系统说明
E50417-H1100-C151
- (2) SIPROTEC DIGSI 4, 启动
E50417-G1100-C152

词汇表

间隔层控制器	间隔层控制器是具有控制和监测功能的装置，不具有保护功能。
控制显示画面	用户在按下装置上的控制键之后，在装置上会显示一个比较大的显示画面，这个画面叫做控制显示画面。控制显示画面上面可以显示用户可以在馈线中可以进行开关操作的元件，可以将装置的状态显示出来。控制显示画面可以用来进行开关操作。控制显示画面的定义是组态的一部分。
清理	对象的频繁增加与删除将在CFC产品中产生大量的不能使用的存储空间。通过对对象清理，用户可以将这些空间重新释放。但是对象清理将对VD地址重新命名，因此所有的SIPROTEC4装置必须重新初始化，才能保证正常工作。
组合装置	组合装置指的是带有保护功能与控制显示画面的间隔层装置。
组合矩阵	在继电器间通信组合（IRC组合）中，最多可以有16个兼容的SIPROTEC4装置进行相互通信。在这种通信过程中，哪种装置交换哪种信息是通过组合矩阵规定的。
通信参考CR	通信参考的功能是说明PROFIBUS（现场总线）通信中使用的工作站的类型和版本。
组件视图	除了提供系统的拓扑图之外，SIMATIC管理器还向用户提供了系统组件视图。系统构成图并不将系统的全部层次结构进行说明，而只是提供这个工程中的所有SIPROTEC4装置的组件视图。
容器	容器是表示一种包括更多对象的一个对象。文件夹对象就是一种容器。

装置容器	在组件视图中，所有的SIPROTEC4装置都被分配了一个类型为Device container（装置容器）的对象。这个对象也是一种特殊的DIGSI4管理器。但是因为DIGSI4管理器没有组件视图，这个对象只有和STEP7联合时才是可以看见的。
现场装置	现场装置指的是在工程现场层次上的装置：保护装置、组合装置、以及间隔层控制器等。
文件夹	这种对象类型用于生成一个对象的层次结构。
HV现场说明	HV工程现场说明文件包括ModPara工程的现场情况的详细说明。每一个现场的实际情况都记录在HV现场说明文件中。在HV工程说明文件中，每一个工程现场都分配了一个HV现场说明文件，通过文件名称可以查看这些文件。
IEC地址	在IEC总线中，每一个SIPROTEC4装置都指定了一个唯一的IEC地址。每一IEC总线可以使用的IEC地址为254个。
初始化字符串	一个初始化字符串包括一系列的针对于不同调制解调器的命令。这些命令在调制解调器初始化帧内被传送到调制解调器，可以对调制解调器进行一些特定的设计。
IRC组合	IRC（继电器间通信）可以用于在SIPROTEC4装置中直接交换处理信息。为配置继电器间通信，用户需要使用一个类型为IRC组合的对象。IRC组合的用户以及所需要全部通信参数都将在这个对象中进行定义。在用户之间交换的信息的类型和范围也都将存储在这个对象中。
连接地址	连接地址可以表示一个V3/V2装置的地址。用户可以通过DIGSI 3.x改变这个地址。
列表视图	工程视窗的右边窗格区域显示一些对象的名称和图标，这些对象代表那些在树型界面选择的容器所包含的内容。因为这些对象是按照列表的形式显示的，因此这个区域叫做列表视图。
订货编号（MLFB）	订货编号（MLFB）是 Maschinenlesbare Fabrikatebezeichnung （机械易读产品名称）的简称。这个编号与产品的订购号相同。在产品订购号中，包含SIPROTEC4装置的类型和版本。

调制解调器连接	这种对象类型包括调制解调器的连接双方的有关信息，包括当地调制解调器与远程调制解调器。
调制解调器配置文件	调制解调器配置文件包括配置文件的名称、调制解调器的驱动器，也可能包括几个初始化命令和一个用户地址。用户可以为一个实际的调制解调器生成几个配置文件，为此，用户需要将各种初始化命令或者用户地址与调制解调器的驱动器及其属性连接，并且将这些初始化命令和用户地址保存在不同的文件名称下面。
调制解调器	某个调制解调器连接所需要的配置文件保存在这个对象类型中。
对象	这个词指的是CFC工程结构中的每一个组件。
对象属性	每一个对象都具有一定的属性。这些属性既可以是几个对象共有的属性，也可以是这个对象所特有的属性。
参数设置	参数设置指的是为某个SIPROTEC4装置设置其所有的参数。
电话本	某个调制解调器连接中的用户地址保存在这个对象类型中。
现场总线	PROFIBUS是PROcess Field BUS的缩写，德国过程和现场总线标准，在标准EN 50170第2卷PROFIBUS中进行说明。PROFIBUS定义了位串行现场总线的功能属性、电气属性和机械属性。
现场总线地址	在现场总线网络中，必须为每一个SIPROTEC4装置指定一个唯一的地址。每一个现场总线网络可以使用254个现场总线地址。
工程	从内容的角度讲，工程是实际的电力供应系统的映射。从数据的角度看，工程是由一定数量的对象构成的，这些对象集成在一个层次结构中。从实际物质的角度看，工程包括一系列的保存工程数据的文件和文件夹。
保护装置	保护装置指的是所有具有保护功能的装置。
服务接口	位于装置后面的串行接口，可以连接DIGSI4装置，例如通过调制解调器进行连接。
设置参数	设置参数可以包括所有的对于装置所做的调整。参数配置是通过DIGSI4完成的，在某些情况下也可以直接在装置上进行设置。

SIPROTEC	SIPROTEC 4是根据V4系统制作的装置的注册商标。
SIPROTEC 4装置	这个对象类型代表一个实际的SIPROTEC4装置，包括这个装置设置参数和过程数据。
SIPROTEC 4副本	这种对象类型代表一个SIPROTEC 4 装置类型的对象的副本。这种副本的装置数据可能和发生变化以前的源对象的装置数据不同，但是同一个源对象生成的副本与原来的对象具有相同的VD地址。因此这些副本对象与源对象所指向的实际SIPROTEC4装置相同。SIPROTEC 4 副本类型的对象有很多用途，例如记录当输入设置参数时SIPROTEC4装置的不同运行状态。
系统接口	位于装置后面的串行接口，可以通过IEC或者现场总线与控制系统相互连接。
拓扑图	DIGSI4管理器通常将一个工程以拓扑图的方式进行显示。这个拓扑图可以显示工程的层次结构及其使用的全部对象。
树型视图	工程视窗的左边窗格区域按照文件夹树的形式显示所有工程容器的名称和图标，因此这个区域叫做树型视图。
用户地址	一个专门的用户地址，包括站名、国家代码、区域代码以及特定用户的电话号码。
用户	在一个继电器间通信连接中，最多可以有16个相互兼容的SIPROTEC4装置进行相互通信。每一个参与通信的装置就是一个用户。
V3/V2 装置	这个对象类型代表对V3 或者V2版的装置中的数值的参考。
VD 地址	VD地址是由DIGSI4管理器自动指定的。这种地址在整个工程中的名称是唯一的，因此可以确切无误地表示一个真实的SIPROTEC4装置。由DIGSI4管理器指定的VD地址必须传送到SIPROTEC4装置中，只有这样才能利用DIGSI 4 装置编辑器进行通信。
VFD	VFD（虚拟现场装置）包括全部的通信对象及其属性和状态，可以供通信的一方通过通信设服务使用。

索引

收件人:

Siemens AG

PTD EA D SC22

Postfach 4806

D-90026 Nürnberg

寄件人:

寄件人姓名:

寄件人职位:

寄件人所在公司名称:

部门名称:

街道:

城市:

电话:

传真:

请在贵公司所属的行业上面打勾号:

自动化

采矿

化工

发电

供电系统、电力管理系统

燃气、水、卫生系统

其它行业

建筑、电力、空调系统

重型机械制造、材料处理技术

管道系统

造船、航海

环保技术

交通工程

