

目 录

序言	1
第一章 项目管理	1-1
一. CBF 登陆界面	1-2
1. 新建一个组态文件	1-3
2. 打开一个项目文件	1-3
3. 引入一个组态文件	1-4
4. 退出 CBF 界面	1-4
二. 组态文件管理界面	1-4
1. 组态文件管理功能	1-5
2. 组态功能	1-5
3. 调试功能	1-5
4. 其它功能	1-5
第二章 建立项目树	2-1
一. 项目树组态界面	2-1
1. 项目树组态界面	2-1
2. 中/英文菜单对照表	2-1
二. 项目树的创建过程及对象介绍	2-2
(一). 建立 CONF 级	2-3
1. 作用	2-3
2. 步骤	2-3
(二). 建立资源或对象级	2-4
1. 建立分支——过程站	2-4
2. 建立分支——操作员站	2-5
3. 建立分支——公共画面库	2-7
4. 建立分支——Operate IT 组态服务器	2-7
5. 建立分支——网关站	2-
(三). 建立过程站下面的程序级	2-10
1. 任务级 (Task)	2-10
2. 程序列表 (PL)	2-12
3. 顺序功能块图 (SFC)	2-13
4. 功能块图 (FBD)	2-14
(四). 建立操作员站下层分支	2-15

三. 检查	2-16
1. 检查	2-17
2. 出错表中的“错误”分类	2-17
3. 错误信息的格式	2-1
练习	2-1
 第三章 功能块图(FBD)组态介绍	3-1
一. 功能块图(FBD)组态界面	3-1
1. 结构	3-1
2. 中/英文菜单对照表	3-2
二. 数据类型与线型关系对照表	3-3
三. 功能块图(FBD)组态步骤	3-4
四. 编辑功能块图	3-5
1. 定义用户菜单	3-5
2. 改变“功能”的输入端脚	3-5
3. 块端脚“取反”	3-6
4. 改变“功能”端脚的数据类型	3-6
5. 改变块的处理顺序	3-6
练习	3-7
 第四章 顺序功能块图(SFC)组态介绍	4-1
一. 用户界面结构	4-1
二. 中/英文菜单对照表	4-2
三. 基本规则	4-3
四. SFC 元素	4-4
1. 初始步	4-4
2. 步	4-4
3. 跳步条件	4-4
4. 垂直线	4-4
5. “或分支”延伸线	4-5
6. “或分支”的分支开始端	4-5
7. “或分支”的分支开始添加端	4-5
. “或分支”的分支结束添加端	4-5
. “或分支”的分支结束端	4-5
10. “与分支”延伸线	4-5
11. “与分支”的分支开始端	4-6
12. “与分支”的分支开始添加端	4-6

13. “与分支”的分支结束添加端	4-6
14. “与分支”的分支 结束端.....	4-6
15. 跳转	4-6
五. SFC 的系统变量	4-6
1. 步的系统变量.....	4-6
2. 跳步条件的系统变量.....	4-7
六. 编辑 SFC 程序.....	4-7
1. 插入、删除列.....	4-7
2. 插入、删除行.....	4-7
3. 删除元素.....	4-7
4. 组态元素参数表.....	4-7
5. 视窗组态.....	4-
6. SFC 元素画面访问功能定义.....	4-10
7. 组态 SFC 程序公共参数	4-11
练习	4-13
第五章 硬件结构组态.....	5-1
一. 硬件结构组态界面.....	5-1
二. 中/英文菜单对照表.....	5-2
三. 操作员站硬件结构组态.....	5-2
1. 选择操作员站资源	5-2
2. 放置操作员站号.....	5-3
3. 组态操作员站外围设备	5-4
四. AC 00F 硬件结构组态.....	5-4
1. 放置过程级资源.....	5-4
2. 放置 AC 00F 站名	5-4
3. 放置模件.....	5-5
4. 放置 Profibus master.....	5-6
5. 放置 Profibus 从设备.....	5-6
6. 放置 Profibus 从设备的子模件	5-7
7. 组态过程级参数.....	5-7
. 组态 Profibus master 参数表.....	5-
. 组态 Profibus slave 参数表.....	5-11
10. I/O editor	5-12
五. 网络与通讯组态设置	5-13
练习	5-15

第六章 调试	6-1
一. 用户界面结构	6-1
1. 启动调试功能	6-1
2. 中/英文菜单对照表	6-1
3. 资源状态指示说明	6-2
二. 系统时间、当地时间、时区的设置	6-4
1. 名词解释	6-4
2. 设置时区	6-4
3. 时钟同步	6-4
三. 程序下装	6-5
1. 下装整个站资源	6-5
2. 下装变量	6-5
3. 下装信息组态	6-5
4. 下装所选中的对象	6-5
5. 下装改变的对象	6-5
四. 调试功能	6-5
(一). 过程级模件状态指示	6-6
1. 现场总线模件的状态	6-6
2. 主模件、子模件的状态显示	6-6
3. 模件产品信息报告	6-6
4. 模件的安全值设置	6-7
(二). 调试 FBD 图	6-7
1. 修改功能块参数表中的参数值	6-7
2. 定义数值视窗与趋势视窗	6-
(三). 调试 SFC 程序	6-
(四). 过程级的调试	6-
练习	6-10
第七章 标签表	7-1
一. 调出标签表	7-1
二. 标签表的结构	7-1
三. 中/英文菜单对照表	7-3
第八章 变量表	-1
一. 调出变量表	-1
二. 变量表的结构	-1

三. 中/英文菜单对照表.....	-2
 第九章 操作员站组态介绍	-1
一. 画面访问组态介绍.....	-1
1. 调“画面访问”	-1
2. 组态画面访问.....	-2
二. 组态总貌画面	-2
三. 组画面组态介绍	-3
1. 新建一个组画面	-3
2. 组态组画面	-4
3. 编辑.....	-5
四.组态趋势画面	-5
1. 新建一幅趋势画面	-5
2. 趋势画面组态介绍	-6
五. 顺序功能块图组态介绍.....	-
1. 新建一个顺序功能块图画面	-
2. SFC 画面组态介绍	-
六. 用户图形画面组态介绍.....	-10
(一). 用户图形编辑界面介绍	-10
1. 用户图形编辑界面	-10
2. 中/英文菜单对照表	-11
(二). 绘制静态图形.....	-13
1. 绘制线条.....	-13
2. 绘制矩形.....	-14
3. 绘制多线段	-15
4. 绘制多边形	-15
5. 绘制椭圆形	-15
6. 写入文字.....	-16
7. 调入位图.....	-17
. 调用宏符号	-1
(三). 绘制动态图形.....	-1
1. 绘制动态棒	-1
2. 绘制选择区	-22
3. 绘制填充区	-23
4. 绘制按键、按键簇、射频按键簇.....	-23
5. 组态趋势视窗.....	-27
练习	-30

附录 I. 功能和功能块	10-1
--------------------	------

序言

序言

当把软件 CBF 安装在计算机上，这台计算机我们就称作工程师站。此软件不仅可用于自动化过程级控制的组态，还可以完成操作员级画面、记录与归档画面的组态；同时还是一种高性能的过程调试工具。使用 CBF 软件组态过程级或操作员级的同时，也随之自动建立了一个系统范围内统一的数据库，支持同一个组态项目下所有资源的调用；此软件具有丰富的功能模块库，用于实现不同的控制、通讯及运算等需求，此外，还支持用户自定义功能块，方便用户组态；组态程序采用项目树结构，生成方便，结构清晰，方便用户对组态程序的调用及修改；程序组态语言方式多元化，并遵从 IEC61131-3 标准；特有的检查功能，对组态语言是否满足 IEC 61131-3 标准自动检查；CBF 软件满足离线及在线组态，是一种应用便捷的组态工具。

CBF 软件特点如下：

- * 唯一的组态软件工具 CBF 可同时完成过程级与操作员站的组态。
- * 其组态语言兼容 IEC 61131-3 标准，支持下列编程语言：
 1. 功能块图语言 FBD
 2. 指令表 IL
 3. 梯形图 LD
 4. 顺序功能块图 SFC
- * 丰富的功能块库中存有 200 多个经过实践验证的功能与功能块，可以满足用户控制、运算、通讯、归档与监视等多种组态需求。
- * 用户可组态大量的自定义功能块(≥200 符号)，用于满足用户特定需求。
- * 采用项目树结构显示用户组态程序，使用项目管理功能管理用户组态。
- * 简洁的交叉参考功能，方便用户对变量、标签的查询及相关程序的调用。
- * 软件特有的检查功能，极大的方便了用户对组态程序的错误识别与更正。
- * 允许项目程序全部及部分的导出与导入。
- * 允许设置口令字，保护组态程序。
- * 允许将程序备份到操作员站。

第一章 项目管理

项目管理

组态工具 **CBF** 采用项目分类管理方法对包括用户组态程序在内的文件及其它文件进行分类管理，即成功安装 **CBF** 软件后，在计算机的指定分区下自动生成一个 **Freelance** 文件夹，在此路径下，又自动建立了以下文件夹，用于不同类型文件的存放。

序号	文件夹缩写名称	存放文件类型
1	bitmaps	位图文件，供用户图形编辑使用
2	bpr	操作记录
3	curv	趋势数据
4	ddesrv	DDE 服务器文件
5	dmsapi	应用接口文件
6	doc	项目文档管理文件
7	exe	执行文件
8	export	整个项目组态的引出文件或部分引出文件
9	gsd	Profibus 文件
10	macro	宏文件
11	opcsrv	OPC 服务器文件
12	proj	项目树文件
13	sap	扰动过程记录文件
14	setctrl	安装控制文件
15	setups	安装记录文件
16	sfp	信号顺序记录文件
17	wave	声音文件

在 **CBF** 登录界面，用户可以新建一个组态文件，打开一个已有的组态文件，存贮组态文件，重新命名或删除组态文件，组态文件的引入等功能。

一、CBF 登录界面

登录 Windows 2000 界面，从 Start → Programs → Freelance2000 v6.2 → Control Builder F → Continue，进入 CBF 登录界面，如图 1-1 所示。

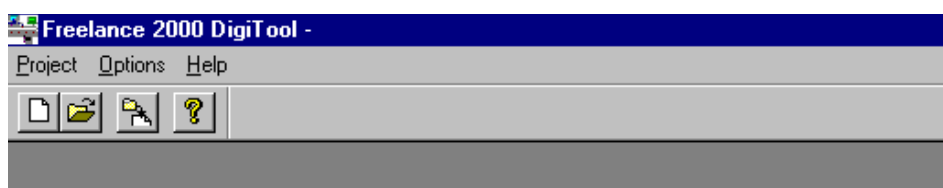


图 1-1 CBF 组态界面

在 CBF 登录界面，包括：标题行、菜单行、工具棒、状态行。从菜单行或工具棒中可完成如下工作：

1、新建一个组态文件

用鼠标点击菜单行中的 Project → new，弹出新建一个项目窗口，如图 1-2 所示。

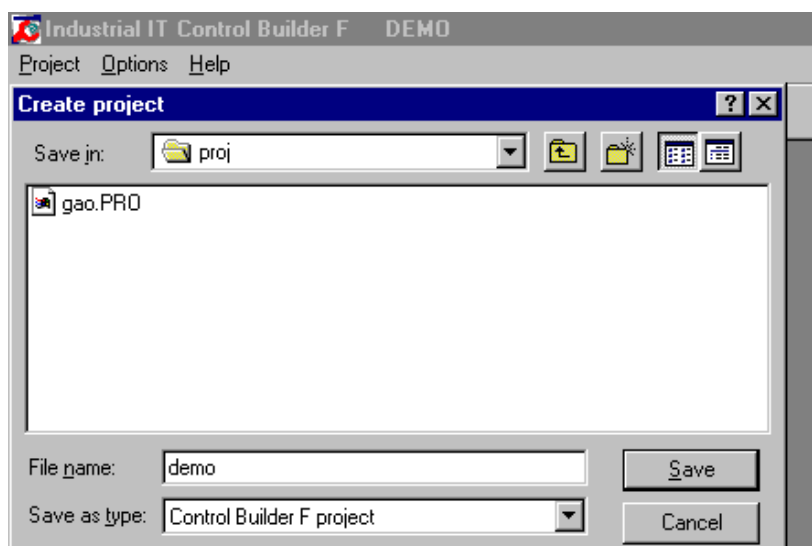


图 1-2 新建一个项目

项目管理

组态文件名(File name)最多允许写入 256 个字符，项目文件的存放路径为 可随意定义；鼠标击 **save** 键后，弹出组态文件标题页，如图 1—3 所示。包括：

Configuration: Project Header

Project name: demo62

Project manager: carol

Project no: sph140

Project orderer: sph140

Project order no: 22345

Project password:

Project size: OK Version: 2003/10/20 10:18:45

Project comment: qing dao glass project

图 1—3 组态文件标题页

- 项目名称 max.12 个字符
- 项目经理 max.16 个字符
- 项目号 max. 6 个字符
- 用户名称 max. 27 个字符
- 定单号 max.12 个字符
- 口令字 max. 8 个字符，min. 4 个字符，用于锁定该组态文件。
- 项目注释 max. 34 个字符

2、打开一个项目文件

使用鼠标点击菜单行 **Project** → **Open**, Windows2000 环境习惯会显示上一次进入的省缺路径。

☞: 在打开一个组态文件的同时, 会产生一个后缀为 (.log) 和 (.bak) 文件。(.log) 文件是一个临时性文件, 用于防止某种变故下组态数据丢失; (.bak) 文件是一个组态文件的压缩存贮文件, 当用户退出 CBF 并存贮了组态文件后, (.bak) 文件自动被 (.pro) 文件取代。因此, 我们建议计算机硬盘的未用空间至少是所建组态文件实际占用空间的 5 倍, 方可以保证 (.log) 文件的正常建立和组态程序的正确性检查。

3、引入一个组态文件

使用鼠标点击菜单行中的 **项目** → **导入**, 可以把一个 (.CSV) 文件引入使用。

☞: 从 V3 开始, (.CSV) 文件为 Unicode 码文件, 只有此码文件方可引入。如果是 V2 或 V1 的 (.CSV) 文件, 是 ASCII 字符的文件, 需加以转换后方可引入。

4、退出 CBF 界面

☞ → **Project** → **Exit**, 退出 CBF 界面。

二、组态文件管理界面

新建一个组态文件或打开一个组态文件后, 进入 CBF 管理界面, 如图 1-4 所示。

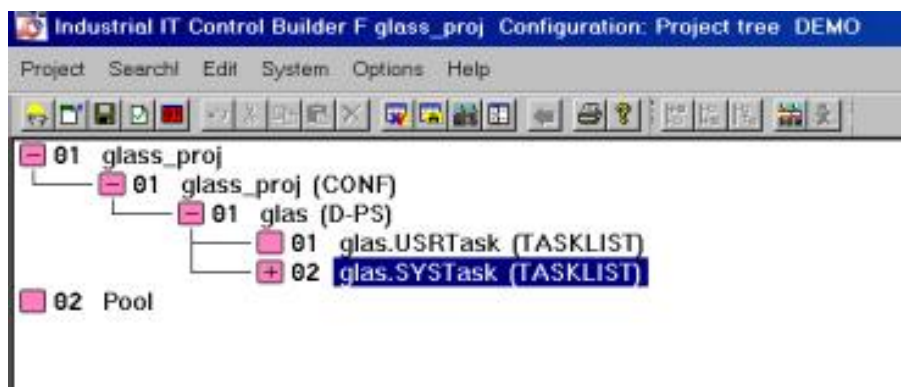


图 1-4 组态文件管理界面

项目管理

从菜单行或工具棒中可完成如下工作：

1、组态文件管理功能

在菜单行 **Project** 中包括了如下组态文件管理功能：

- * **Export / Export backup**: 把项目引出存为一个 (.csv) 文件 / 引出作为备份 (可将引出输出/输出备份文件存到其它 PC 上)。
- * **Close** (关闭) : 关闭当前的组态文件。
- * **Save** (存储) : 在同一个路径下保存当前组态文件。
- * **Save as** (另存为) : 允许在不同路径下保存当前组态文件。
- * **Delete** (删除) : 删除当前组态文件。
- * **Header** (要点信息) : 允许重新编辑当前组态文件标题页。
- * **Comment** (注解) : 允许重新编辑组态文件注释文字
- * **On _ line** (在线) : 在线组态
- * **Off _ line** (离线) : 离线组态
- * **Exit** (退出) : 退出 CBF 管理界面

2、组态功能

鼠标点击菜单行的组态项，进入组态文件项目树组态界面。

3、调试功能

鼠标点击菜单行的调试项，进入组态文件调试界面。

4、其他功能

可从菜单行的 选项完成其他辅助功能，如安装了 **Digilock** 软件，可从此选项下用不同的用户身份登陆组态界面。

第二章 建立项目树

建立项目树

在组态方式下，我们可以创建一个以项目树结构体现用户组态的程序文件。项目树结构的优点包括：各层次生成灵活，结构清晰，各个资源的作用一目了然，便于用户了解组态构造；程序层的添加、删除方便等等。

一、项目树组态界面

1、项目树组态界面

如图 2_1 所示，项目树的组态界面由标题行、菜单行、工具棒、用户组态图形区、状态行四部分组成。

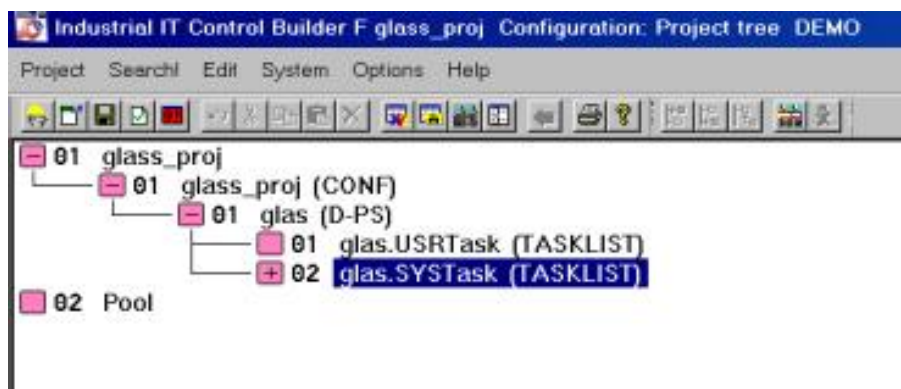


图 2—1 项目树组态界面

2、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Project Menu	项目菜单	Options Menu	选项菜单
Save	在原路径保存组态	Hardcopy	硬拷屏
Documentation	文档管理	Long state line	以长状态行格式显示
Check / Check all	检查	Color setting	调整颜色设置表
Show error list	调出上一次错误表		
Header	调出标题页		
Comment	组态注释文字		

建立项目树

英文菜单	中文说明	英文菜单	中文说明
Commissioning	进入调试界面	Edit Menu	编辑菜单
Exit	退出当前界面	Undo	返回操作前状态
		Program	进入光标所指定界面
System Menu	系统菜单	Insert above	放在光标同层的上方
Variable list	进入变量表	Insert below	放在光标同层的下方
Tag list	进入标签表	Insert next level	放在光标的下一层
Structure data types	定义新的数据类型	Expand	打开分支
Global message processing	定义整个项目树的故障、开关报警信息确认等级	Full expand	将一个分支全部打开
Local message processing	定义每个操作员站报警信息表的格式	Compress	收拢一个分支
Hardware structure	进入硬件结构编辑界面	Export block	导出某一部分组态
Display access	定义 TAG 的画面访问	Import block	导入某一部分组态
Show global variables	显示系统所有的变量		
Show exported Variables	显示引出的变量		
Show all objects	显示所有的组态对象		
Show select objects	显示光标指定的组态对象		

二、项目树的创建过程及对象介绍

用  击菜单行的 Configuration，可进入到项目树的组态界面中。整个项目树的结构最初或最终可显示为：

Project name  项目程序组态级
Pool  库级

项目树的建立步骤如下：

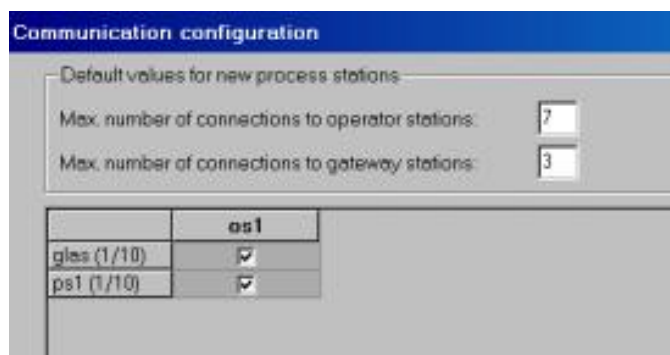
(一)、建立 Configuration (CONF) 级

1、作用

整个组态项目下的资源(包括过程站、操作员站及其它站)及对象的数据库由其统一管理，数据传送由其负责，时区设置由其完成。

2、步骤

光标选中 project name， 编辑， Header，弹出 CONF 选项对话框，选中 Edit → OK，弹出表 2—2 所示 CONF 级组态标题页。在标题页中填写：



The image shows a 'Communication configuration' dialog box. It has a title bar 'Communication configuration'. Below it is a section 'Default values for new process stations' with two input fields: 'Max. number of connections to operator stations' with value '7' and 'Max. number of connections to gateway stations' with value '3'. Below this is a table with two columns. The first column has labels 'gs1 (1/10)' and 'ps1 (1/10)'. The second column has a header 'os1' and two checkboxes, both of which are checked.

	os1
gs1 (1/10)	☒
ps1 (1/10)	☒

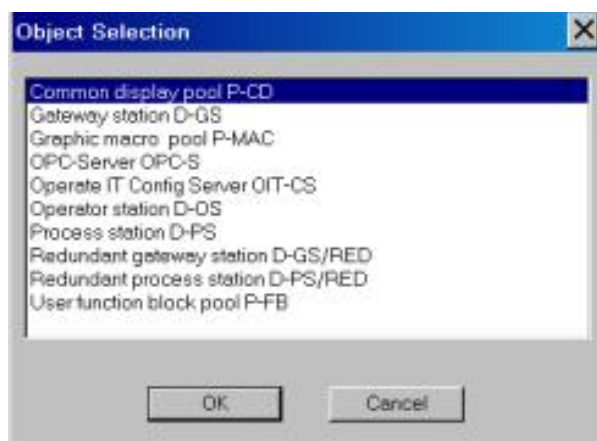
表 2—2 CONF 组态对话框

- * Name: max.12/16 个字符
- * 设置时区: 设置整个组态项目的时区。用按键时区设置时区。
- * 组态 OS 站与 GS 站的数量: 一个项目下允许 OS 与 GS 的数量之和最大为 10 个，当前显示值为默认值。

建立项目树

(二)、建立资源或对象级

项目树 CONF 级下允许建立的分支包括如下表所示的站资源及对象：



Common display pool P-CD ——— 公共画面库

Gateway station D-GS ——— 网关站

Graphic macro pool P-MAC ——— 公共图形库

OPC-Server OPC-S ——— OPC 服务器

Operate IT Config Server OIT-CS ——— Operate IT 组态服务器

Operator station D-OS ——— 操作员站

Process station D-PS ——— 过程站

Redundant gateway station D-GS/RED ——— 冗余配置的网关站

Redundant process station D-PS/RED ——— 冗余配置的过程站

User function block pool P-FB ——— 用户自定义功能块库

1、建立分支——过程站 (D-PS)

对过程设备的控制、运算(加、减、乘、除及逻辑运算)、监视、趋势采集的组态都在过程站下面组态完成。

首先用光标选中 CONF 级， → Edit → Insert next level，弹出资源选项窗口。从窗口中选中过程站 (D-PS) 或冗余的过程站 (D-PS/RED)，按 OK 键，弹出如表 2-4 所示过程站组态标题页。

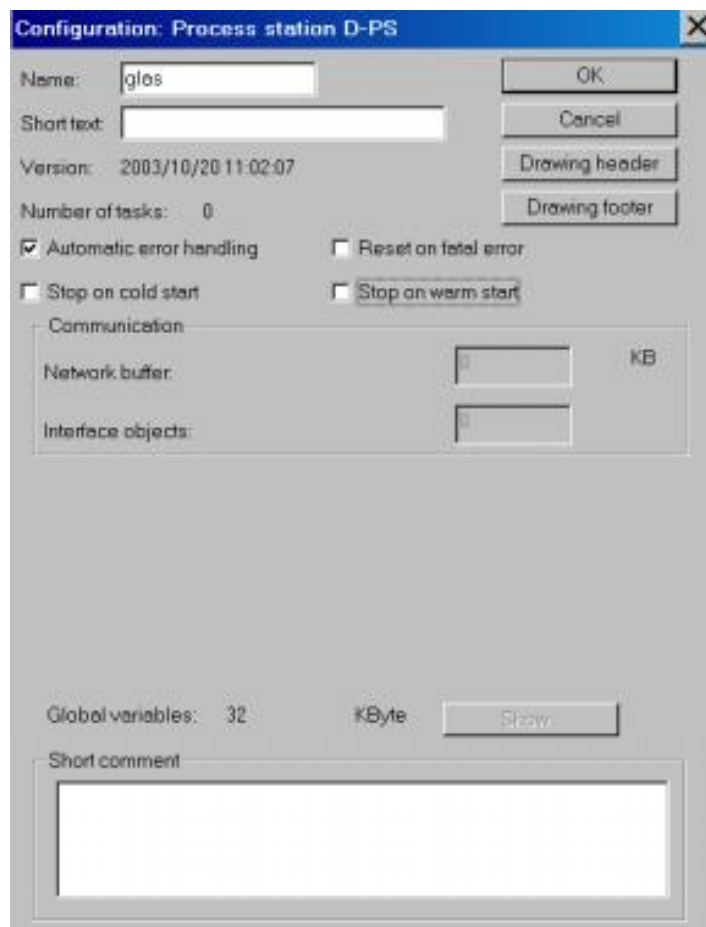


表 2—4 过程站组态标题页

填写内容包括：

- * Name 过程站的名称，max.4 个字符。

2、建立分支——操作员站 (D-OS)

操作员站上显示的工厂流程画面、趋势画面、记录等的组态，均在操作员资源 (D-OS) 下面组态完成。

首先用光标选中 CONF 级， Edit  Insert next level，弹出资源选项窗口。从窗

建立项目树

口中选中操作员站 (D-OS) → OK 后，弹出操作员站组态标题页，如表 2—5 所示。

标题页中的组态内容包括：

- * 名称： 操作员站的名字，max .4 个字符。
- * D-OS 密码： 此处设置的口令字用于阻止操作人员退出操作界面；即当用户退出 DigiVis 界面时需有此口令字方可退出。如不在此设置口令字，用户可随时退出 DigiVis 界面。
- * Diag 密码： 此处设置口令字，用于访问操作员站中系统画面的诊断界面，如不在此设置口令字，用户无权访问系统画面的诊断界面。
- * 扫描周期： 用于定义操作员站上信息表、系统画面和面板的刷新时间。
- * 闪烁频率： 定义画面中闪烁颜色的闪烁时间。

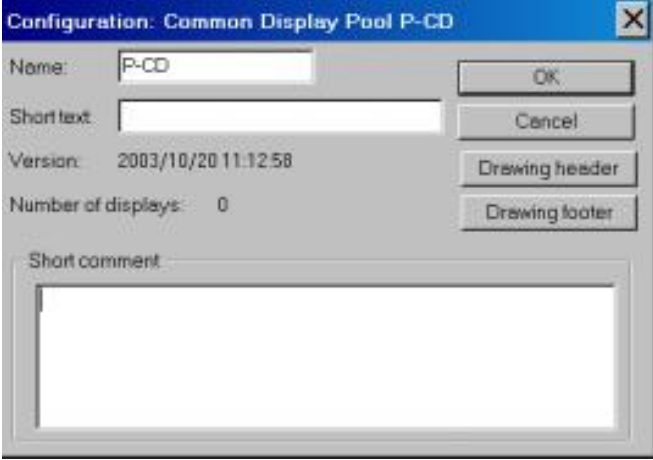
表 2—5 操作员站组态标题页

3、建立分支——公共画面库 (Common display pool P-CD)

在公共画面库中组态的画面，允许被同一个项目树下的所有操作员站调用，即随着各个操作员站的下装，公共画面库中的画面会自动下装到操作员站中。即同一个项目树下的所有操作员站的共有画面只需在公共画面库中组态一次，所有的操作员站就都可以访问这些画面；但趋势画面、记录和总貌画面必须在各个操作员站资源下单独组态。

首先用光标选中 CONF 级， Edit  Insert next level，弹出资源选项窗口。从窗口中选中公共画面库 (Common display pool P-CD) → OK 后，弹出公共画面库组态标题页，如表 2—6 所示。标题页中组态内容包括：

* 名称：公共画面库的名称，max .4 个字符。



The image shows a Windows-style dialog box titled "Configuration: Common Display Pool P-CD". It contains several input fields and buttons. The "Name:" field is filled with "P-CD". The "Short text:" field is empty. The "Version:" field shows "2003/10/20 11:12:58". The "Number of displays:" field shows "0". There are four buttons on the right: "OK", "Cancel", "Drawing header", and "Drawing footer". At the bottom, there is a "Short comment:" label above a large empty text area.

表 2—6 公共画面库标题页

4、建立分支——Operate IT Config Server OIT-CS

Operate IT 可以作为 Freelance 2000 系统的操作员站使用，为了将 Freelance 2000 组态的数据传送给 Operate IT，需要在项目树下建立 Operate IT 组态服务器分支。

建立项目树

表 2—7 Operate IT 组态服务器标题页

首先用光标选中 CONF 级， 编辑 插入下一级，弹出资源选项窗口。从窗口中选中 Operate IT 组态服务器 (Operate IT Config Server) → OK 后，弹出 Operate IT 组态服务器组态标题页，如表 2—7 所示。

标题页中组态内容包括：

- * 名称： Operate IT 组态服务器的名称，max .4 个字符。
- * 计算机名称： 同一个网络中安装有 Operate IT 数据库的 PC 机名或它的 IP 地址
- * 项目名称： 将 Freelance 2000 系统集成在 Operate IT 中的项目名称
- * 语言： 选择 Operate IT 组态语言

5、建立分支——网网站 (Gateway station)

Freelance 2000 系统可以提供与其它数据源进行数据交换的途径。例如：与 DDE Server、OPC Server 或者 Maestro UX 的操作系统进行数据交换。

首先用光标选中 CONF 级， → Edit → Insert next level，弹出资源选项窗口。从窗口中选中网网站 (Gateway station D-GS) → OK 后，弹出网网站组态标题页，如表 2—8 所示。

Configuration: Gateway station D-GS

Name: gs

Shorttext:

Version: 2003/10/20 11:20:06

Type: DCP-Gateway

Ext. time server

☐ Enable

IP address 1:

IP address 2:

IP address 3:

☐ Reset on fatal error

Short comment:

OK

Cancel

Drawing header

Drawing footer

Access rights ..

表 2—8 网关站组态标题页

填写内容包括:

名称	网关站的名称, Max. 4 个字符
类型	网关站的类型

建立项目树

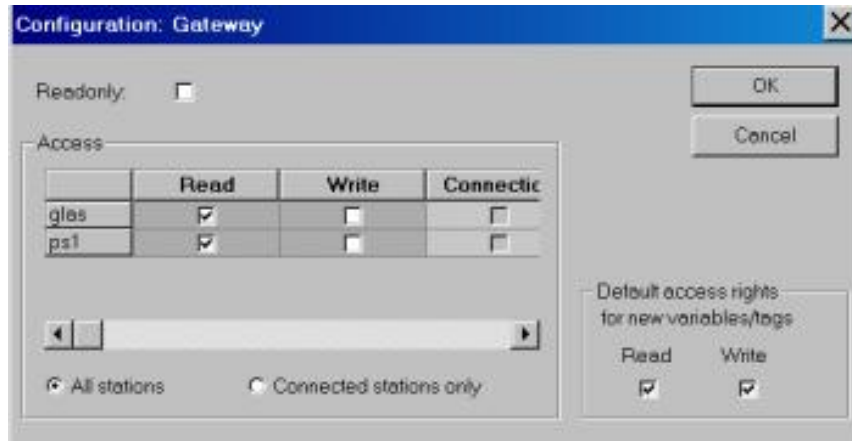


表 2—9

需要定义网关站的访问权限，按下 **Access right** 按键，弹出表 2—9。

Read ☐ 表示允许第三方将数据写入过程站，反之则不能写入。

Access 定义网关站访问每一个资源站的权限。

☐ 表示网关站无权对这个资源站写入或读出数据

☒ 表示允许网关站从这个资源站读出数据或写数据到资源站

All stations ☒ 表示网关站与所有的资源站通讯

Connected stations only ☐ 表示网关站仅与“**Connection** ☒ ”的资源站通讯

(三)、建立过程站 (D—PS) 下面的程序级

过程站 (D—PS) 下的程序级需要分层建立，首先需要建立任务级，在任务级下面我们再建立程序列表级 (Program lists) 与顺序功能图级 (SFC program)，在程序列表级与顺序功能图级的下层是最终的组态控制程序与计算程序，我们可以使用功能块图 (FBD) 语言、指令表 (IL) 语言、梯形图 (LD) 语言完成控制与运算程序的组态。

1、任务级 (TAS)

随着过程站的建立，在过程站的下层会自动生成 用户 任务和系统 任务，其中系统任务自动生成而不需要用户单独组态；用户对过程级的组态放置在用户任务下完成；每一个

过程站的用户任务下面最多可组态 8 个任务 (Task) 和 1 个省缺任务 (Default task); 任务下又分为程序列表 (PL) 和 SFC 程序。其中任务 (Task) 为循环处理任务, 优先级可以设置为 51~99, 99 的优先级最高; 省缺任务(Default task)为 PLC 方式, 占有最低的优先级 50。



图 2—9 任务级组态标题页

标题页填写内容包括:

名称:任务名称, Max.12 个字符

等间距: 表示任务的执行方式是按照规定循环时间做严格的等间隔方式循环

建立项目树

最优化加载：在正常情况下，此方式与 等间距方式无任何区别，当因系统中的某些原因导致任务延时执行时，等间距仍会按原时间间隔方式执行，而造成 CPU 高负荷工作；最优化加载会适当延长任务循环时间，缓和 CPU 的负荷。

Cycle time: 任务循环处理时间。最少允许设置为 **5ms**，最多允许设置为 **24 天**，必须以 **5ms** 的整数倍写入。
输入格式为时间格式，为：**T#x h x m x s**。

PLC mode: 是 **Default Task** 工作方式，占有最低的优先级 **50**，可被任何其它任务随时中断。

Automatic startup: 当下装或启动过程站资源后，任务会自动启动。

Overload message: 如果任务间隔时间过短，通过此处组态会在操作员站上产生一个任务过载信息；优先级可定义为 **1—3**，信息文字最多 **8** 个字符。

Process image: **4kB** 的倍数，最多 **32kB**

2、程序列表 (PL)

程序列表 (PL) 附属在任务 Task 的下层，标题页如表 2—10 所示，填写内容包括：

Name: 程序列表名称，**max.12** 个字符。

Processing [V] on: PL 的下级程序允许执行
[] on: PL 的下级程序不允许执行

Short comment: **max.159** 个字符。

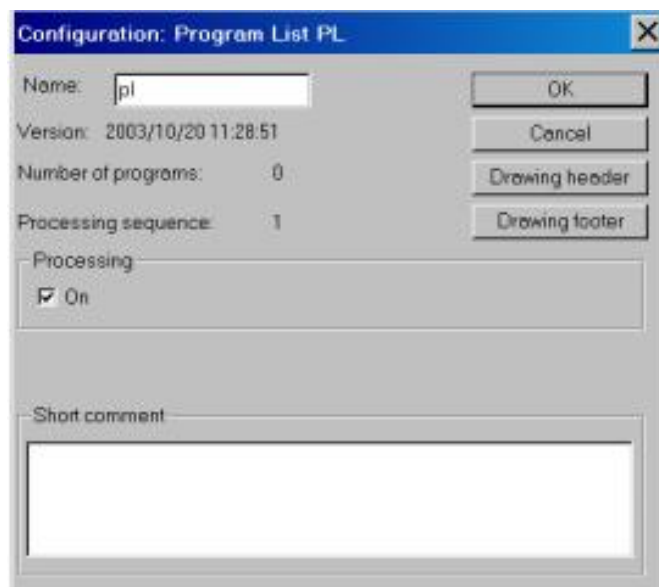


图 2—10 程序列表(PL) 组态标题页

3、顺序功能块图 (SFC)

顺序功能块图 (SFC) 附属在任务 Task 下层，与程序列表 (PL) 同级，组态对话框如表 2—11 所示，填写内容包括：

建立项目树

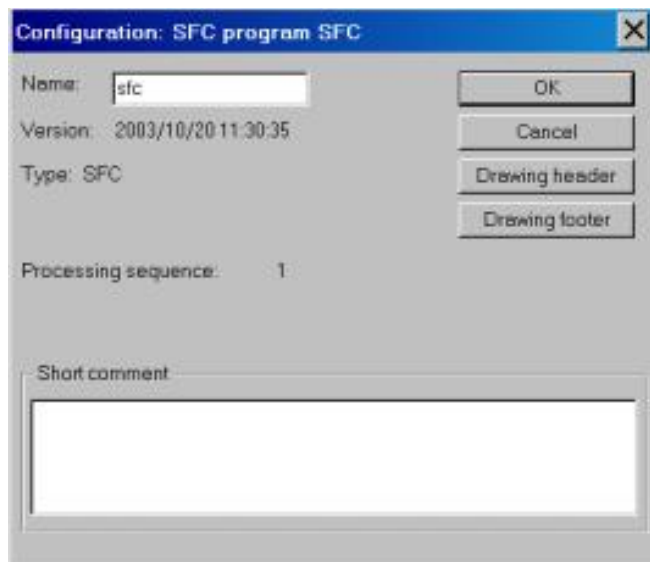


图 2—11 组态顺序功能块图(SFC)

Name: 顺序功能块图名称，max.12 个字符

Short comment: 文本注释，max.159 个字符

4、功能块图 (FBD)

功能块图 (FBD) 附属在程序列表 (PL) 或顺序功能图 (SFC) 的下层，组态标题页如图表 2—12 所示，填写内容包括：

Name: 功能块图名称，max.12 个字符。

Short comment: 文本说明，max.159 个字符。

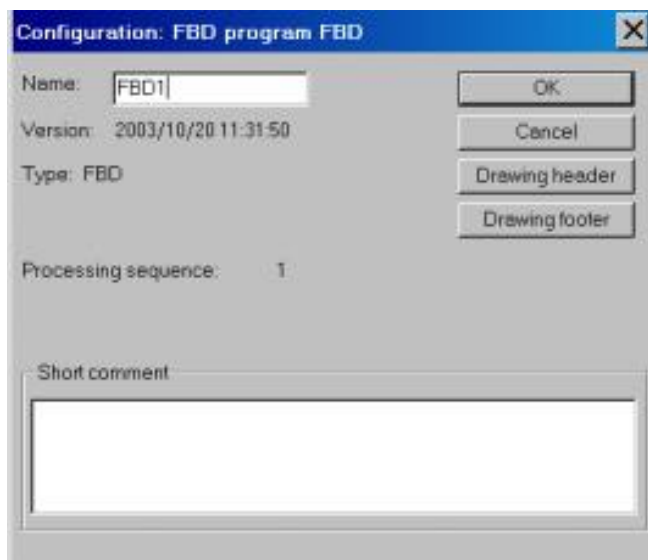


图 2—12 功能块图(FBD) 组态对话框

(四)、建立操作员站 (D—OS) 下层分支

光标选中项目树中的操作员站 (D—OS), 用  → Edit → Insert next level, 调出如图 2—13 所示对话框, 从中选出需要在操作员站下组态的画面种类即可。

操作员站中显示的画面种类包括:

- Disturbance course log (DCL_D—OS) —— 干扰过程记录
- Graphic display (FGR) —— 用户图形画面
- Group display (GRP) —— 组画面

- Operation log (OPL) —— 操作记录
- Overview display (OVW) —— 总貌画面
- SFC display (SFCP) —— 顺序功能图画面
- Signal sequence log 1 (SSL1) —— 信号顺序记录 1
- Signal sequence log N (SSLN) —— 信号顺序记录 N
- Trend display (TR_D—OS) —— 趋势画面

建立项目树

- Time scheduler display (TD_D-OS) —— 时间程序画面
- Structure node (STRUCT) —— 操作员站下层的结构元素；用于将画面归组分类，结构清晰，便于用户调用。

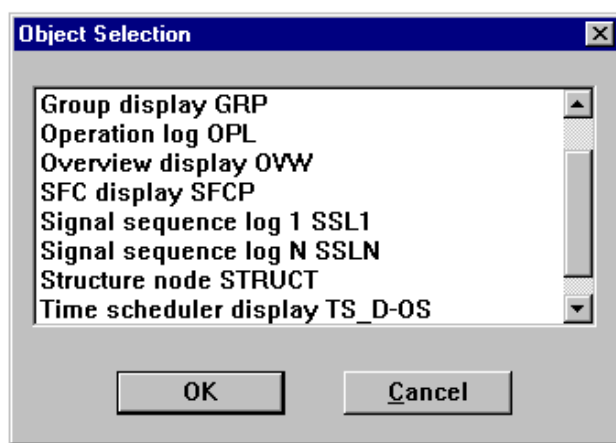


图 2—13 操作员站画面种类选项窗口

三、检查

项目树中的各个资源对象是由连接小方块 分层相连构成了项目树，组态好的对象可以

经过运行检查(Check)指令或对象全部检查(Check all)指令发现组态是否存在错误，用出错表说明的同时，还通过连接小方块 I 的颜色显示检查结果。其中：

粉色  ： 表示项目对象的内容被修改或检查后有错误。

→  粉色有箭头，检查后发现与以前的组态内容相比，删掉了一些内容。

绿色  ： 表示对象经过检查。

红色  ： 表示对象经过检查正确，但是此对象与过程源有连带关系，在下装此对象之前，用户必须先下装过程源。

黄色: 表示对象经过正确性检查, 但是此对象与其它任务有连带关系, 下装此对象之前, 用户必须先下装相关任务。

注: 以上颜色为缺省颜色设置, 如果想修改连接小方块 颜色, 用→ Options → color setting , 调出颜色设置对话框修改 (建议用户不要修改此项设置)。

1、检查

新完成的组态、出错后更正过的组态都需要经过检查; 检查的方式包括检查 (Check) 与全部检查 (Check all), 只有经过检查后没有错误的资源与对象, 才允许进入调试方式。

- 检查 (Check)

 选中对象 → 组态方式 → Project → Check

检查 (Check) 仅仅是对选中对象的错误点做局部检查, 并将出错结果列举在出错表中。

- 全部检查 (Check all)

 选中对象 → 组态方式 → Project → Check all

检查 (Check all) 是对选中对象的全部组态做检查, 将出错结果列举在出错表中。一般情况下需要等待几分钟。

如果出错表中的错误较多时, 需要从较下层的对象开始一一检查并更正错误; 为了方便查找错误, 我们用鼠标左键双击出错表中的某个错误后, 关闭出错表, 当前的错误就显示在屏幕的中央, 并用淡蓝色表示。

2、出错表中的“错误”分类

出错表中的“错误”分为错误 (Error)、警告 (Warning)、提示 (Hint) 三类。

- 错误 (Error)

表示组态过程中有“语法”错误, 这种错误不允许项目进入调试方式下装组态, 是必须要更正的错误。

建立项目树

• 警告 (Warning)

表示某个对象的组态存在不连贯性或不一致性，但是不影响项目的调试与下装。是否更正警告 (Warning) 信息，完全取决于组态人员的意愿。

• 提示 (Hint)

仅仅是对用户的提示信息，丝毫不影响项目的调试与下装；可以从出错表中屏蔽掉。

3、错误信息的格式

错误信息的格式为：

< 出错号 > < 出错级别 > < 出错说明 >

如：[xxxx.2.xxxxx] Error : < 出错说明 >

[xxxx.3.xxxxx] Warning : < 出错说明 >

[xxxx.4.xxxxx] Hint : < 出错说明 >

如图 2—14 所示：

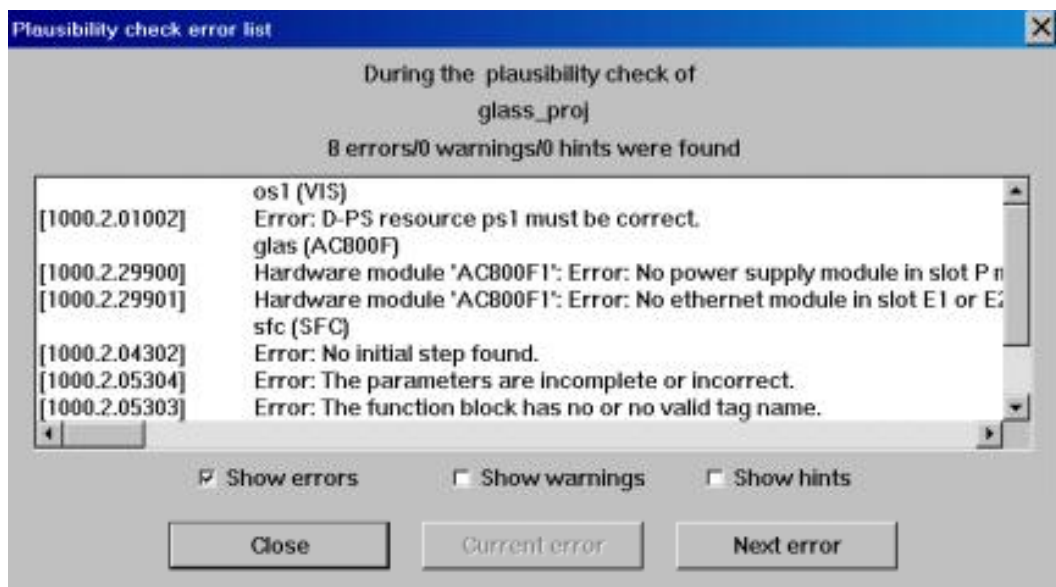


图 2—14 出错表

练习

- 1、建立一个新的项目文件，项目名称 (Name): Basic1
- 2、在项目树下建一个过程站和一个操作员站，名称分别为 (Name): PS_1、OS_1
- 3、在过程站下建两个任务 (Task)
 - 1)、Task1 参数设置包括：

名称: Task_1	☐ 间隔方式	☐ 最优化加载
循环时间: T#500ms	☒ 自动启动	
优先级: 51		
 - * Task2 参数设置包括：

名称: Task_2	☐ 间隔方式	☐ 最优化加载
循环时间: T#500ms	☒ 自动启动	
优先级: 51		
- 4、在 Task_1 下建一个程序列表 PL：

名称: PL_1	☒ ON
----------	--
- 5、在 PL_1 下建 4 个 FBD 图

名称: FBD_1、FBD_2、FBD_3、FBD_4

- 6、在 Task_2 下建一个程序列表 PL 和一个顺序功能块图 SFC

名称: PL_2、SFC_1

- 7、在 PL_2 下建 4 个 FBD 图

名称: FBD_5、FBD_6、FBD_7、FBD_8

- 8、在操作员站 (D-OS) 下建立一幅总貌画面，2 幅组画面，3 幅用户图形画面，1 幅 SFC 画面和 2 幅趋势画面。

第三章 功能块图（FBD）组态介绍

功能块组态介绍

一、功能块图（FBD）组态界面

功能块图（FBD）是一种图形式的编程语言，遵从 IEC61131-3 标准。通过功能块图特有的 CAD 功能，用户仅需要按照设计要求，将需要的功能及功能块从功能块库中调出，参照设计要求与系统规范填写参数表，用软线将功能与功能块与现场过程变量连接后，即完成用户组态；组态过程简便，用户无需具有计算机编程知识。

1、结构

功能块图（FBD）组态界面如图 3-1 所示，包括菜单行、工具棒、图形区、状态行四部分。图形区共有十页，页与页间用虚线分割，我们可以通过拖动下面或右边的滚动条改变图形区页面；图形区的网格允许用户打开或关闭；用户组态由功能块、功能、变量及连线组成，我们习惯将用户组态放在当前页中，一幅 FBD 图中建议放置一个相关组态程序。

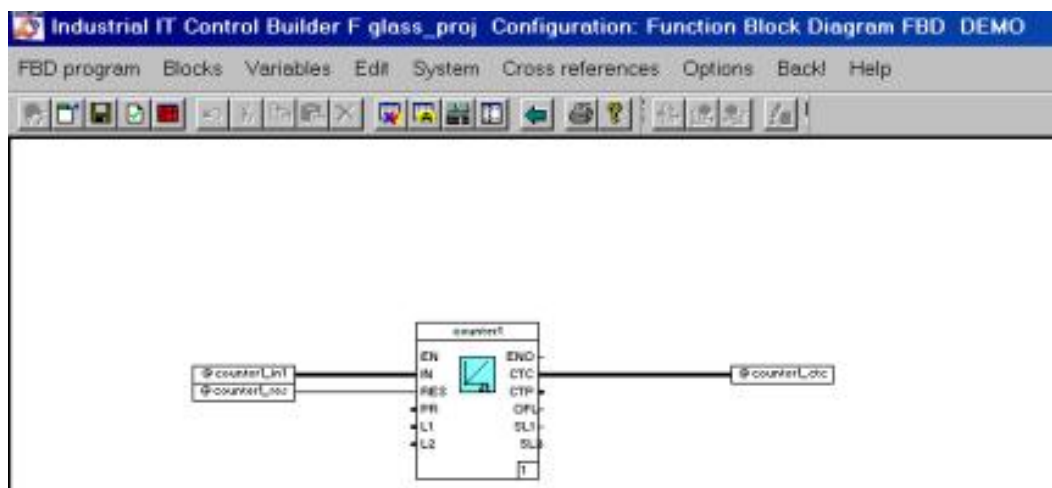


图 3-1 FBD 图组态界面

2、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
FBD Program	FBD 程序菜单	System	系统菜单
Save	储存	Variable list	调变量表
Check	程序检查	Tag list	调标签表
Header	调出对象标题页	Hardware structure	调系统硬件结构界面
Comment	编辑说明文字	Structure data types	增添新的数据类型
Exit	退出 FBD 图组态界面		
Analog	模拟量功能块集合	Undo	返回操作前状态
Binary	数字量功能块集合	Change data type	改变数据类型
Constant	常数功能块集合	Parameters	组态指定功能块参数
Converter	数据转换功能块集合	Processing sequence	改变功能块的处理顺序
Acquisition	变量采集功能块集合	Change number of input	改变功能输入端脚的数 量
Controller	PID 控制功能块集合	Select variable	调出变量表(等同于 F2 键)
Standard	标准功能块集合	Export block	引出部分组态程序
Open-loop Controller	阀或电机类控制功能 块集合	Import block	引入部分组态程序
Modbus Master	Modbus 通讯功能块 (主方)集合	Cross reference	交叉参考菜单
Modbus Slave	Modbus 通讯功能块 (从方)集合	Cross reference	启动交叉参考功能(等同 于 F5 键)
Monitoring	监视功能块集合		
System Function	系统功能块集合	Options	选项菜单
TCP/IP Send and Receive	TCP/IP 发送接收功 能块集合	Hardcopy	打印当前屏幕显示的组 态程序

功能块组态介绍

Blocks	功能块库菜单	Edit	编辑菜单
DigiBatch	DigiBatch 接口功能块	Version	显示组态版本信息
Remote control	远程控制功能块	Raster on	打开/关闭图形区网格
User function block	用户自定义功能块	Define user menu	定义用户菜单
User menu	用户菜单	Colors	设置 FBD 图中的功能块、功能、线条等的显示颜色

二、数据类型与线型关系对照表

在 Control Builder 组态软件中，包括以下数据类型：REAL、DINT、INT、UDINT、UINT、DWORD、WORD、BYTE、BOOL、DT、TIME，这些数据依据其分类不同，在功能块上用不同的颜色和粗细线型加以区分。

数据类型	位数	数值范围	说明	显示颜色	线型	输入格式
REAL	32	-1.175494351E-38 3.402823466E38	浮点值	黑色	粗	0.0、3.2 3.14E-6
DINT	32	-2147483648... ... +2147483647	带符号的双整数值	草绿色	粗	+23456
INT	16	-32768... ... +32767	带符号的整数值	淡绿	粗	3, -3
UDINT	32	0...4294967295	不带符号双整数值	褐色	粗	4566, 1234567
UINT	16	0...65535	不带符号的整数值	深兰色	粗	66
DWORD	32	0... $2^{32}-1$	双 word	紫红色	粗	0, 655
WORD	16	0~ $2^{16}-1$	word	兰色	粗	2, 554

数据类型	位数	数值范围	说明	显示颜色	线型	输入格式
BYTE	8	0~255 0~2 ⁸ -1	二进制	灰色	粗	0,55 2#00000011
BOOL	8	0 or 1 FALSE or TRUE	布尔值	黑色	细	0, 1 TRUE or FALSE
DT	32		日期+时间	深黄色	粗	DT#1994-02- 14-10:00:00.00
TIME	32		时间	淡黄	粗	T#3m22s
错误状态				红色	细	
选中对象				浅兰色		
未选中对象				黑色	细	

三、功能块图 (FBD) 组态步骤

- 如果某个功能或功能块的放置位置需要改变，首先选中它，然后按住鼠标左键拖拽功能或

功能块到指定位置后，释放鼠标左键。

- 双击某功能块或选中某功能块，用  → Edit → Parameters 调出功能块参数表，按照需要填写相关参数。

- 信号-read 对应输入变量，信号-write 对应输出变量。

第一次写入的输入、输出变量必须指定数据类型，如果变量指定的数据类型与功能块要求的数据类型不一致，软连线的颜色显示为红色；必须进入变量表，修改数据类型。

- 按住 ctrl 键的同时拖动鼠标左键绘制连接软线。
- 如果出错表中显示有错误 (Error)，必须更正错误。只有经过检查 (Check) 后没有错误的组态，允许储存后退出。

所有的输入、输出变量和标签，一经写入会自动存放在公共数据库的变量表 (Variable list) 和标签表 (Tag list) 中，以后相同变量或标签的调用可不必重新写入，可用 F2

功能块组态介绍

从变量表或标签表中调出即可。

四、编辑功能块图

1、定义用户菜单

我们可以将组态过程中经常用到的功能与功能块放到用户菜单中，方便用户调用。

☞ Options→Define user menu

→ 将右边窗口中列举的功能或功能块根据需要选中

→ 按下 **Add** 键或双击

选中的功能与功能块添加到左边的窗口中，我们可以从菜单项块下的用户菜单中调出使用；如果想删除用户菜单中的某个功能块，我们可以

☞ Options →Define user menu

→ 将左边窗口中列举的功能或功能块根据需要选中

→ 按下 删除键

我们还可以将用户菜单中的功能与功能块归类放置。

☞ Options →Define user menu

→ 将左边窗口中列举的功能或功能块根据归类需要选中

→ 将功能块放置到已有的同类功能块的中间位置

→ 单击鼠标

2、改变“功能”的输入端脚

小功能块中的“ADD”、“OR”、“SUB”等的输入端脚可以根据需要改变，端脚



数量可以选择从 2 到 10 。

☞ 选中功能 → Edit → Change number of input

- 通过鼠标移动改变功能的输入端脚
- 按 ESC 键或右键结束

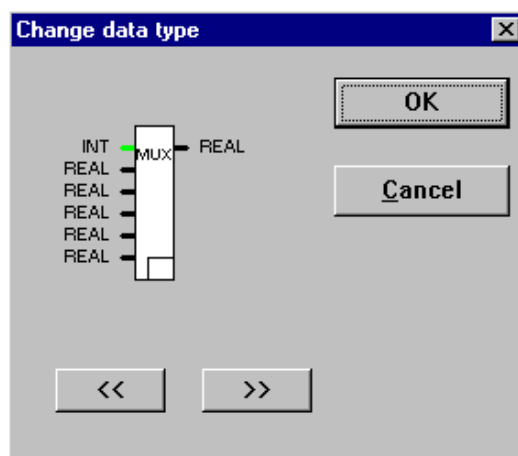
3、块端脚“取反”

我们可以使用如下方法在功能块上直接“取反”，即按住 **Ctrl** 键的同时，用鼠标左键在功能块的端脚上按下一个点；仅局限于 **Bool** 数据类型的端脚。



4、改变“功能”端脚的数据类型

- 选中块 → Edit → Change data type
- 用 » 或 « 改变端脚的数据类型



5、改变块的处理顺序

- 选中块 → Edit → Processing sequence → 写入顺序号 → Enter
- 或 按住 Ctrl 键 → 选中块的顺序号 → 写入新的顺序号 → Enter

练习

1、参照图 2.1，在 FBD_1 图中，使用模拟量输入计数器功能块 模拟输入累计 Counter with Analog Input (CT—ANA) 和 单向阶跃 Monoflop (MONO—F) 完成功能块图组态，通过组态设置，如果计数器的输出超出了功能块所设定的报警限值后，CT—ANA 功能块被复位，此功能块也可以由外部输入信号 Count1—RES 触发复位。

MONO—F 的作用是单向阶跃，脉冲的间隔的宽度可设定，触发方式为上升沿和下降沿触发。

• 模拟量输入计数器 (CT—ANA)

名称: Count1

短注释: Counter1

量程 start / end: 0.0 / 200.0

单位: %

时基: Minute

信息

类型: H 数值: 100.0

Access ☐

优先级: 3

信息文本: HIGH

• 单向阶跃 (MONO—F)

名称: Mono1

短注释: Monoflop

脉冲延续: T#10s

量程结束: T#60s

2、参照图 3.1，在 FBD_2 图中使用 PID 运算功能块 C—CS 组态完成一个比例积分运算。

为了构成仿真控制，在控制器的输出端增加一个延时功能块，将输出延时后引入为控制器的输入值 (PV)。

PID 控制为外设定方式，使用选择开关 SEL 为控制器提供两路外设定值选择，由功能块 MONO—F 的输出变量作为选择信号。(开关 SEL 有三路输入，第一路输入为 Bool 值，用于选择输入端哪路有效，当其为 logic—0 时，中间的一路允许输出，当其为 logic—1 时，最下面的一路允许输出)

- Continuous standard controller (C—CS) :

Parameters: Continuous controller, universal C_CU (1/3)

General data

Name: CONTROL Short text: 水位控制 Processing: ☒

Long text: Sequence: 4

Range start: 0.0 Range end: 10.0 Dimension: m

Internal set point: 0.0 ☒ Access

Output: 0.0 ☒ Access

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1	H	9.0	☐	0.3	3	-	H
2			☐	0.3	-	-	
3			☐	0.3	-	-	
4			☐	0.3	-	-	

OK Cancel Save Reset Check Help << >>

功能块组态介绍

Parameters: Continuous controller, universal C_CU (3/3)

Gradient function (time per range)

Set point rising: T#0s
 falling: T#0s

☒ Setpoint gradient active in manual

Output rising: T#0s
 falling: T#0s

☒ Output gradient active in manual

Constants

Set point limit high: 10.0
 Set point limit low: 0.0
 Output limit high: 100.0
 Output limit low: 0.0
 Dead band for c.-error: 0.0

Effectiveness of output limits

☐ active in Man and Auto
☒ active in Auto
☐ not active

Set point tracking

☒ SPI tracks actual set point
☐ SPI as adjusted

OK Cancel Save Help << >>

Parameters: Continuous controller, universal C_CU (2/3)

PID parameters

☒ P branch
☒ I branch
☒ Inverse char.
☒ Valve char. rising
☐ CP-var. peakless
 Operating point: 0.0

D-action
☐ on
☒ positive
☐ negative
☐ off

D - action of
☐ PV ☒ CE
 P - action of
☐ PV ☒ CE
 Disturbance to:
☒ PV ☐ Out ☐ CE
☐ Monit. disturbance

CP: 2.0
 TR: T#15s
 TD: T#0s
 CD: 1.0
 Tsync: T#30s

Operating mode inhibition

☐ SP-Intern ☐ Out-Man
☐ SP-Extern ☐ Out-Auto

Operating modes

☒ Intern ☐ Extern
☒ Man ☐ Auto

Sync. PI - behav. to P - behav.

☐ Oper.-point tracks integr. part
☒ Oper.-point as adjusted

Sync. Man to Auto

☐ without
☐ adj. to old set point
☒ adj. to current set point

OK Cancel Save Reset Check Help << >>

- **Delay Function (DELA)**

名称: Delay1 短注释: Simulation

延时时间: T#10s

3、参照图 4.1，在 FBD_3 中用模拟量监视功能块组态监视来自模拟输入模块 DAI01 通道 0 的 4---20 mA 输入值；用数字量监视功能块组监视一个数字量信号。

来自模拟量输入子模块 **DAI01** 的现场模拟量信号类型为 **UNIT** 类型，而模拟量监视功能块所接收的数据类型为 **REAL** 类型，需要使用功能块 **Analog input transformation** 将信号转换后方可使用。

- **Analog input transformation (AI_TR)**

Name: AI_TR1 Short text: Analog

Long text: UNIT transfer to REAL

Default : 量程的起点: prio: 3 Message text : LOW

量程的终点: prio: 3 Message text : HIGH

功能块组态介绍

• Analog monitoring (M_ANA)

Parameters: Analog monitoring M_ANA

General data

Name: **ANALOG** Short text: 反应罐温度 Processing: ☒

Long text: Sequence: 5

Scale start: 0.0 Scale end: 100.0 Dimension: BC

Messages

☐ Reset messages on DIS=1

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1	H	95.0	☐	3.0	3	-	温度 H
2	HH	90.0	☐	3.0	3	-	温度 HH
3	L	10.0	☐	3.0	3	-	温度 L
4	LL	5.0	☐	3.0	3	-	温度 LL

OK Cancel Save Reset Check Help << >>

Binary monitoring (M_BIN)

Parameters: Binary monitoring M_BIN

General data

Name: **DIGITAL** Short text: 容温监视 Processing: ☒

Long text: Sequence: 6

Display text for

Message state 1: 温度高

Message state 0: 正常

Message

☒ Message on I-signal

Mess. priority: 3 Hint: - Message text: 容温高

OK Cancel Save Reset Check Help << >>

第四章 顺序功能块图（SFC）组态介绍

SFC 组态介绍

顺序功能块图 (SFC) 组态方法遵循 IEC 1131—3 标准，是按照事件的逻辑顺序安排其控制子程序结构的；SFC 的程序组态图是由元素：步 (Step)、跳步条件(Transition)、跳转(Jump)、连线(Line)等组成的；元素中的组态子程序可以选择由功能块图（FBD）、梯形图（LD）、指令表（IL）的一种或多种方式完成；通过 SFC 组态，可以将复杂的逻辑控制程序由其简洁的元素框图表现出来，结构清晰，组态方式简便。

一、用户界面结构

如图 4—1 所示，顺序功能块图 (SFC) 组态界面包括：标题行、菜单行、工具棒、

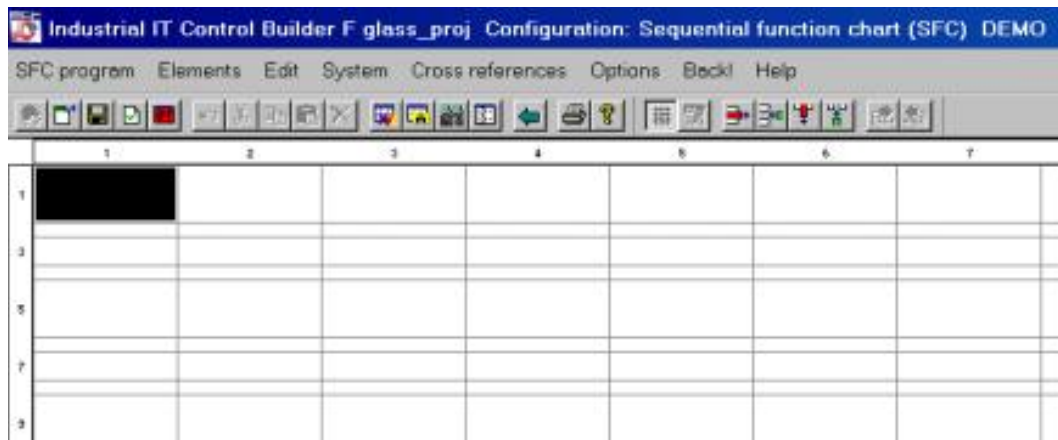


图 4—1 SFC 组态界面

SFC 组态介绍

图形区、状态行；图形区是由大宽行、中宽行和小细行组成，共有 512 行 16 列，用于放置 SFC 的元素；其中大宽行用于放置步，中宽行用于放置条件，小细行用于放置元素“连线”（包括垂直连线和分支连线）。

2、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
SFC Program	SFC 程序菜单	System	系统菜单
Save	储存	Variable list	调变量表
Check	程序检查	Tag list	调标签表
Header	调出对象标题页	Hardware structure	调系统硬件结构界面
Comment	编辑说明文字	Structure data types	增添新的数据类型
Exit	退出 SFC 组态界面		

Elements	元素菜单	Edit	编辑菜单
Initial step	初始步	Undo	返回操作前状态
Step	步元	Column insert	列插入
Jump	跳转	Column delete	列删除
Transition	跳步条件	Row insert	行插入
Vertical line	垂直连线	Row delete	行删除
Horizontal seq. Selection line	“或分支”延伸线	Delete element	删除元素
Seq. Selection divergence start	“或分支”的分支开始端	Parameters of element	组态元素中的子程序
Seq. Selection divergence add	“或分支”的分支开始添 加端	Edit program	编辑元素中的子程序
Seq. Selection convergence add	“或分支”的分支结束添 加端	Options	选项菜单
Seq. Selection convergence end	“或分支”的分支结束端	Version	显示子程序的组态信息

SFC 组态介绍

Elements	元素菜单	Edit	编辑菜单
Horizontal simultaneous seq. Line	“与分支”延伸线	Hardcopy	打印当前屏幕显示内容
Simultaneous divergence start	“与分支”的分支开始端	Grid	打开或关闭图形区网格
Simultaneous divergence add	“与分支”的分支开始添加端	Scale	打开或关闭行号、列号显示
Simultaneous convergence end	“与分支”的分支结束端		
Simultaneous convergence add	“与分支”的分支结束添加端		
Define criteria window	定义元素视窗功能(便于在操作员站上观察)		
Define display selection	定义元素画面访问		
Parameters of SFC	组态 SFC 程序的参数		

三、基本规则

任一顺序控制 SFC 程序总是由一个初始步开始的。

- _ 每一步 (step) 的后面应跟从一个条件。
- _ 一个 SFC 程序的结束元素必须是条件。

在 SFC 程序中有两种分支存在, 即“或分支”和“与分支”。“或分支”是指在同一时刻不能满足分支下的所有跳步条件, 只能允许分支下的某一路被执行; “与分支”是指当跳步条件满足时, 允许跳步条件下的所有分支被同时处理。

SFC 程序中元素步的执行, 取决于跳步条件是否得到满足。如果某一步处于执行状态, 而其下方的跳步条件不能满足时, SFC 只能循环执行当前步, 直至其下面的跳步条件得到满足时, 才允许执行下一步; 此时 SFC 程序中其它步都处于不执行状态, 即这些步中

SFC 组态介绍

的组态子程序不再被执行。因此，过程点的监视、信息报警、设备的控制是不能直接在 SFC 程序中组态完成的，这些组态程序应放在程序列表(PL)下面由任务作循环处理。

四、SFC 元素

1. 初始步 (Initial step)

一个 SFC 程序中只允许包括一个初始步，它是一个 SFC 程序的开始；初始步的外观如下图所示。



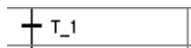
2. 步 (Step)

一个 SFC 程序中允许存在多个步元素，步的外观如下图所示。每一步(包括初始步)中的子程序组态可用 FBD、IL 或 LD 完成，子程序数量最多允许 8 个。



3. 跳步条件 (Transition)

用于判断上一步的完成和决定下一步的开始。每一个跳步条件中仅仅允许组态一个子程序。外观如下图所示。



4. 垂直线 (Vertical line)

放置在细行中，起元素间的连接作用。



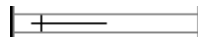
5. “或分支”延伸线 (Hori ontal selection line)

用于 “或分支”分支间的连接，只能放置在细行中。



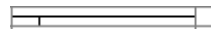
6. “或分支”的分支开始端 (Seq. Selection divergence start)

“或分支” 的分支发出起点，只能放置在细行中。



7. “或分支”的分支开始添加端 (Seq. Selection divergence add)

“或分支” 的开始分支添加端，只能放置在细行中。



8. “或分支”的分支结束添加端 (Seq. Selection convergence add)

“或分支” 的分支结束添加端，只能放置在细行中。



9. “或分支”的分支结束端 (Seq. Selection convergence end)

“或分支” 的分支结束端，只能放置在细行中。



10. “与分支”延伸线 (Hori ontal simultaneous seq. line)

用于 “与分支”分支间的连接，只能放置在细行中。



SFC 组态介绍

11. “与分支”的分支开始端 (Simultaneous seq. divergence start)

“与分支”的发出点，只能放置在细行中。



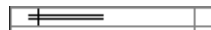
12. “与分支”的分支开始添加端 (Simultaneous seq. divergence add)

“与分支”的发出添加端，只能放置在细行中。



13. “与分支”的分支结束端 (Simultaneous seq. convergence end)

“与分支”的分支结束端，只能放置在细行中。



14. “与分支”的分支结束添加端 (Simultaneous seq. convergence add)

“与分支”的结束添加端，只能放置在细行中。



15. 跳转 (ump)

用于程序的跳转，图标如下图所示；应放在大宽行中，名字为要跳转步 (step) 的名字，同时在目的步上附加一个跳转符号表示。



五、SFC 的系统变量

1、步的系统变量 —— .X

SFC 程序中的每一步都包括系统变量 **.X**，**.X** 是一个只读变量 (read)，用于指示步的当前状态——激活或非激活。

.X=1 或 TRUE	表示此步处于激活状态
.X=0 或 FALSE	表示此步未激活或已激活过

2、跳步条件的系统变量 —— **.RESULT**

SFC 程序中的每一个跳步条件都包括系统变量 **.RESULT**，并且每一个跳步条件的输出变量必须包括 **.RESULT**。

.RESULT=1 或 TRUE	表示条件满足，允许执行下一步
.RESULT=0 或 FALSE	表示转换条件不满足，只允许执行当前步

六、编辑 SFC 程序

1、插入、删除列 (Column insert / delete)

相当于列号的右移或左移，但是列的总数不变。

2、插入、删除行 (Row insert / delete)

相当于行号的下移或上移，但是行的总数不变。

3、删除元素 (Delete element)

 选中元素 → Edit → Delete element，将选中元素从 SFC 程序中删除。

4、组态元素参数表 (Parameters of element)

1)、组态步 (Step) 参数表

步 (Step) 参数表如图 4-2 所示：

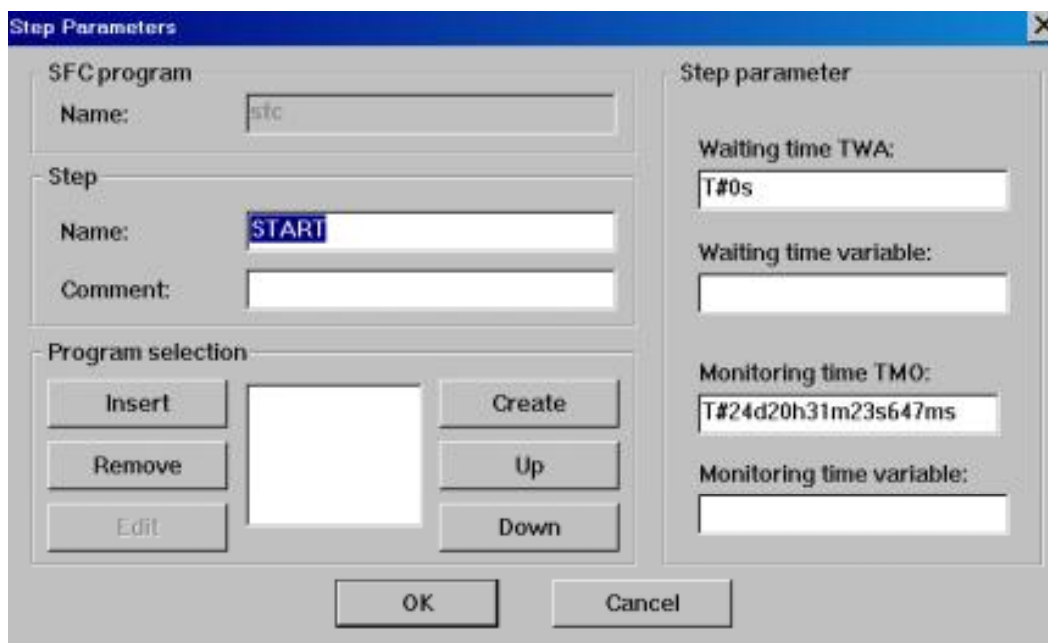


图 4—2 元素步 (Step) 参数表

Step

Name: 步的名称，最多允许 8 个字符

Comment: 步的文本说明

Program selection

Insert 将移走的程序从 Pool 中移回

Move 将选中的程序移到 Pool 中

Edit 进入步的子程序组态界面

Create 生成步的新的组态子程序，子程序的组态方式可以选用 FBD 图、IL 表、LD 图，每一步中最多允许组态 8 个子程序。

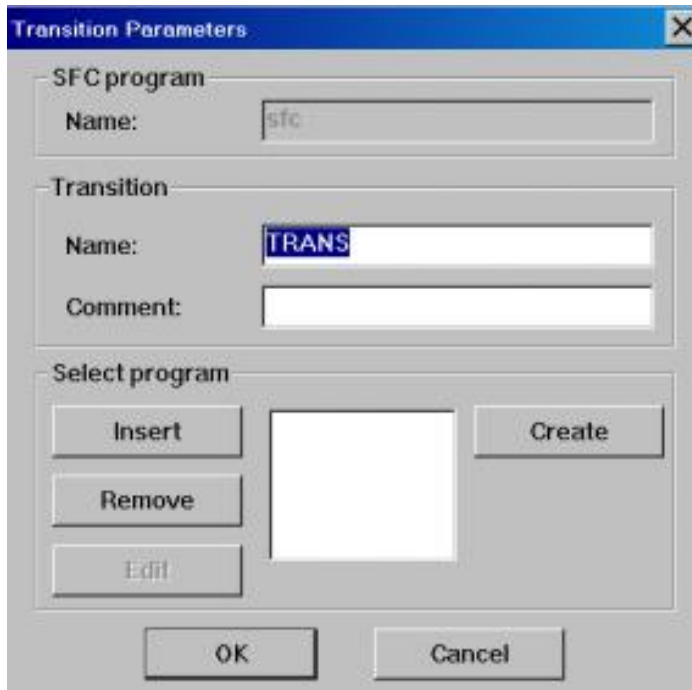
向上 / 向下 可以改变子程序的处理顺序，上面的子程序先处理。

Waiting time TWA : 以时间格式写入，是当前步的最小等待时间

Monitoring time TMO : 以时间格式写入，是当前步的最大允许等待时间，允许保留默认值

2)、组态跳步条件 (Transition) 参数表

跳步条件 (Transition) 的参数表如图 4—3 所示:



The image shows a 'Transition Parameters' dialog box. It has three main sections: 'SFC program' with a 'Name' field containing 'sfc'; 'Transition' with a 'Name' field containing 'TRANS' and an empty 'Comment' field; and 'Select program' which includes 'Insert', 'Remove', and 'Edit' buttons on the left, a central empty box, and a 'Create' button on the right. At the bottom are 'OK' and 'Cancel' buttons.

图 4—3 跳步条件 (Transition) 参数表

Transition

Name : 跳步的名字，最多允许 8 个字符

Comment: 跳步的文本说明

Select program

Insert 将移走的程序从 Pool 中移回

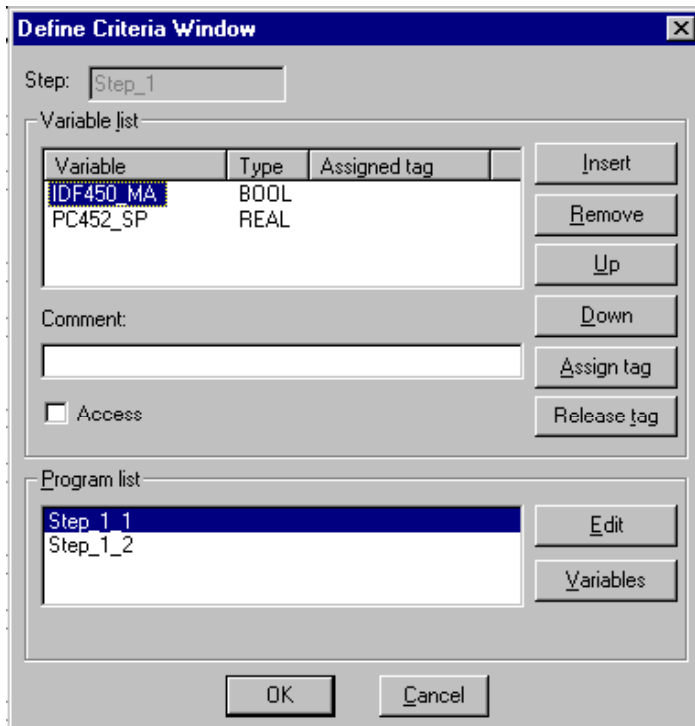
Remove 将选中的程序移到 Pool 中

Create 生成元素新的子程序，子程序的组态方式可以选用 FBD 图、IL 表、LD 图，每一个跳步条件中仅允许组态 1 个子程序

5、视窗组态 (Define criteria window)

SFC 组态介绍

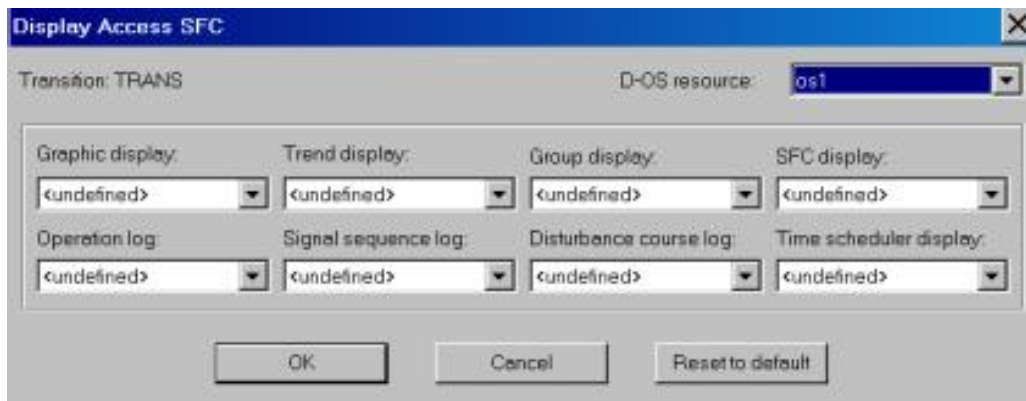
此选项用于定义 SFC 程序中每一个元素的视窗，视窗具有在 DigiVis 上监视已定义元素步或跳步条件中变量值变化的功能，每一个元素中最多允许定义 20 个变量，一个变量可以组态对应一个标签 Tag，即对应操作员站上的一个面板。



6、SFC 元素画面访问功能定义 (Define display access)

我们可以采用定义 SFC 程序中某一个元素的画面访问功能，使用户方便的在 DigiVis 上调出 SFC 程序中这些元素所对应的相关画面。

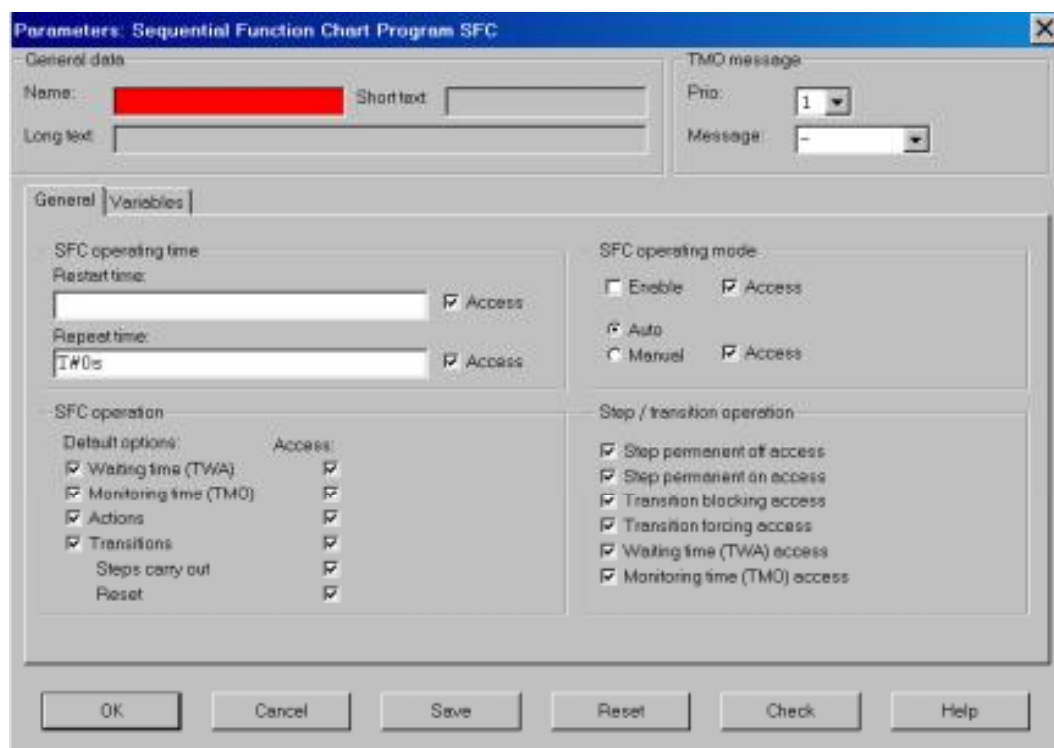
☞ → Edit → Define display access



7、组态 SFC 程序公共参数 (Parameters of SFC)

SFC 程序的组态还必须包括对 SFC 程序公共参数的定义

→ Edit → SFC 参数编辑弹出下图，组态选项包括：



General data

Name: 是这个 SFC 程序在项目数据库中特定标签名，max.11 个字节

SFC 组态介绍

Short text: SFC 程序的短文本说明，max.12 个字节

Long text: SFC 程序的长文本说明，max.30 个字节

TMO message

Prio: 设置超出 TMO 时间时产生报警信息的优先级

Message: 报警信息文字

SFC operating time

Restart time: 输入格式：年一月一日一天一小时：分：秒：毫秒

Repeat time: 是此 SFC 程序重新启动的间隔时间

SFC Operating mode

Enable ☐ 允许此 SFC 程序在过程站中执行

☐ 不允许此 SFC 程序在过程站中执行

Acess ☐ 允许在操作员站上由操作员决定 SFC 程序能否执行

Auto ☒ 允许 SFC 程序在操作员站上自动执行。

Manu ☒ 允许 SFC 程序在操作员站上由操作人员手动控制。

Enable ☐ 允许在操作员站上由操作员决定 SFC 程序的手、自动工作方式。

SFC operation: 通过对选项的选择，决定 SFC 程序的执行方式。

练习 1、 参照下图，在 FBD_5 中选用单方向驱动功能块 **IDF_1** 完成电机控制组态图。

• **Individual drive function (IDF_1)**

Parameters: IDF for unidirectional units IDF_1 (1/2)

General data

Name: **IDF1** Short text: 电动门控制 Processing: ☒ Sequence: 1

Long text:

Status texts

Status text 1: 开

Status text 0: 关

Run time monitoring

Max. run time: T#5s

☒ Monitoring ON

Reaction after trouble

☒ MANUAL and OFF

☐ previous values

Feedback

☒ Feedback variable available

☒ Monitoring end position ON

OK Cancel Save Reset Check Help << >>

2、 参照下图、在 FBD_6 中组态一个比例积分控制回路。

SFC 组态介绍

- Level controller function block type ____ “Universal”

Parameters: Continuous controller, universal C_CU (1/3)

General data

Name: Short text: Processing: ☒ Sequence:

Long text:

Range start: Range end: Dimension:

Internal set point: ☒ Access

Output: ☒ Access

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1	H	85.0	☐	0.3	3	-	液位H
2	L	35.	☐	0.3	3	-	液位L
3			☐	0.3	-	-	
4			☐	0.3	-	-	

OK Cancel Save Reset Check Help <>

Parameters: Continuous controller, universal C_CU (2/3)

PID parameters

☒ P branch ☐ D-action: ☐ on ☐ P-action of: ☐ PV ☒ CE CP:

☒ I branch ☐ positive ☐ P-action of: ☐ PV ☒ CE TR:

☒ Inverse char. ☐ negative ☐ Disturbance to: ☐ PV ☐ Out ☐ CE TD:

☒ Valve char. rising ☐ off ☐ Monit. disturbance CD:

☐ CP-var. peakless Operating point: Tsync:

Operating mode inhibition

☐ SP-Intern ☐ Out-Man

☐ SP-Extern ☐ Out-Auto

Sync. PI-behav. to P-behav.

☐ Oper.-point tracks integr. part

☒ Oper.-point as adjusted

Operating modes

☒ Intern ☐ Extern

☒ Man ☐ Auto

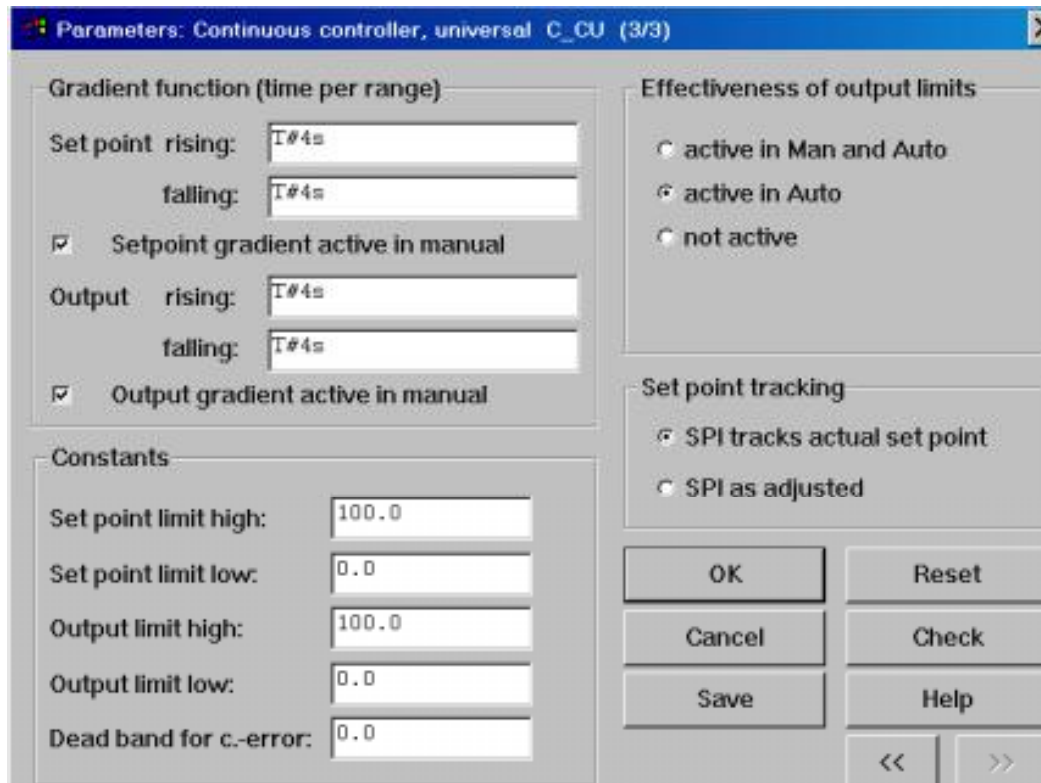
Sync. Man to Auto

☐ without

☐ adj. to old set point

☒ adj. to current set point

OK Cancel Save Reset Check Help <>



- 延时功能块的延时时间为 T 20s

3、参照下图、在 FBD_7 中组态一个比例积分控制回路。

- Pressure controller function block type ____ “Universal”

SFC 组态介绍

Parameters: Continuous controller, universal C_CU (1/3)

General data

Name: **PC450** Short text: Processing: ☒

Long text: Sequence:

Range start: **0.0** Range end: **100.0** Dimension:

Internal set point: ☒ Access

Output: ☒ Access

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1	H	85.0	☐	0.3	3	-	H
2	L	35.	☐	0.3	3	-	L
3			☐	0.3	-	-	
4			☐	0.3	-	-	

OK Cancel Save Reset Check Help << >>

Parameters: Continuous controller, universal C_CU (2/3)

PID parameters

☒ P branch	D-action ☐ on	D - action of ☐ PV ☒ CE	CP: 1.5
☒ I branch	☐ positive	P - action of ☐ PV ☒ CE	TR: T#15s
☒ Inverse char.	☐ negative	Disturbance to : ☒ PV ☐ Out ☐ CE	TD: T#0s
☒ Valve char. rising	☒ off	☐ Monit. disturbance	CD: 1.0
☐ CP- var. peakless			Tsync: T#30s
Operating point: 0.0			

Operating mode inhibition

☐ SP-Intern	☐ Out-Man
☐ SP-Extern	☐ Out-Auto

Operating modes

☒ Intern	☐ Extern
☒ Man	☐ Auto

Sync. PI - behav. to P - behav.

☐ Oper.-point tracks integr. part
☒ Oper.-point as adjusted

Sync. Man to Auto

☐ without
☐ adj. to old set point
☒ adj. to current set point

OK
Cancel
Save
Reset
Check
Help
<< >>

Parameters: Continuous controller, universal C_CU (3/3)

Gradient function (time per range)

Set point rising: T#1s

falling: T#1s

☒ Setpoint gradient active in manual

Output rising: T#1s

falling: T#1s

☒ Output gradient active in manual

Constants

Set point limit high: 100.0

Set point limit low: 0.0

Output limit high: 100.0

Output limit low: 0.0

Dead band for c.-error: 0.0

Effectiveness of output limits

☐ active in Man and Auto

☐ active in Auto

☐ not active

Set point tracking

☐ SPI tracks actual set point

☐ SPI as adjusted

确认

取消

存储

复位

检查

帮助

<<

>>

Parameters: Continuous controller, universal C_CU (3/3)

Gradient function (time per range)

Set point rising: T#3s

falling: T#3s

☒ Setpoint gradient active in manual

Output rising: T#3s

falling: T#3s

☒ Output gradient active in manual

Constants

Set point limit high: 100.0

Set point limit low: 0.0

Output limit high: 100.0

Output limit low: 0.0

Dead band for c.-error: 0.0

Effectiveness of output limits

☐ active in Man and Auto

☐ active in Auto

☐ not active

Set point tracking

☐ SPI tracks actual set point

☐ SPI as adjusted

OK

Cancel

Save

Reset

Check

Help

<<

>>

SFC 组态介绍

- 延时功能块的延时时间为 T 20s

4、在 Task_2 的 SFC_1 中组态一个 SFC 程序，通过 SFC 程序组态完成对电机 IDF1 的开关及手自动控制，液位 LCV5201 回路和压力 PC450 回路的调控。子程序可以用 FBD 图完成。

1)、组态 SFC 程序

- **初始步 Start**

参数：

名称：Start 等待时间 Tw：T#20s

子程序的任务：

将电机控制功能块 IDF450 的工作方式置于“**AUTO**”。

- **条件 rans**

参数：

名称：Trans_1

子程序的任务：如果 IDF1 已工作于自动方式，那么允许继续执行 Step_1 和 Step_11。

- **步 tep**

参数：

名称：Step_1 等待时间：T#20s

子程序的任务：

开动电机 IDF1，并将液位控制回路 LCV5201 的液位设定值置于 90 m。

- **步： tep**

参数：

名称: Step_11 等待时间 Tw : T#20s

子程序的任务: 将压力控制回路 PC450 的压力设定值置于 90 mbar 。

- **条件: rans**

参数:

名称: Trans_21

子程序的任务:

如果液位控制器 LCV5201 的输出超过报警高限 L1(85° C), 允许执行 Step_2。

- **条件: rans**

参数:

名称: Trans_22

子程序的任务:

如果压力控制器 PC450 的输出超过报警高限 L1 (85mbr), 允许执行 Step_21。

- **步: tep**

参数:

名称: Step_2 等待时间 Tw : T#20s

子程序的任务:

将电机 IDF1 置于关, 并将液位控制器 LCV5201 的设定值设置为 25 m。

- **步 tep**

参数:

名称: Step_21 名称时间 Tw : T#20s

子程序的任务:

将压力控制器 PC450 的设定值设置为 25 mbr。

SFC 组态介绍

- **条件:** *rans*

参数:

名称: Trans_3

子程序的任务:

如果电机 IDF1 已处于关位置, 并且液位控制器 LCV5201 低于下限 L2 (30° C), 继续执行 Step_3。

- **tep :** *tep*

参数:

名称: Step_3

Waiting time Tw : T#20s

子程序的任务:

将电机 IDF1 的工作方式置于手动方式。

- **条件:** *rans*

参数:

名称: Trans_4

子程序的任务:

结束 SFC 程序。

2)、组态 SFC 程序的公共参数

名称: SFC100

短注释: SFC Program

长描述: Level-Pressure-Variation

SFC 操作模式: AUTO

Enable

3)、为所有的步与条件定义标准视窗。

第五章 硬件结构组态

硬件结构组态

我们在项目树下组态的资源如过程级(D-PS)、操作员站(D-OS)、网关节点(D-GS)等，都包括各自的硬件设备，并且，过程级从种类上可以分为 AC800F、现场控制器和机架式过程站等，每一种过程级中都包括不同的模块，模块的放置位置及参数要求会各不相同；这就需要我们按照实际的配置将设备硬件组态到项目的硬件结构中去，系统硬件结构组态将在操作员站上以系统画面的方式显示出来。

在硬件结构组态中还要完成网络资源 ID 与 IP 地址的设置。

一、硬件结构组态界面

☞ → System → Hardware structure，进入硬件结构组态界面，如图 5-1 所示。

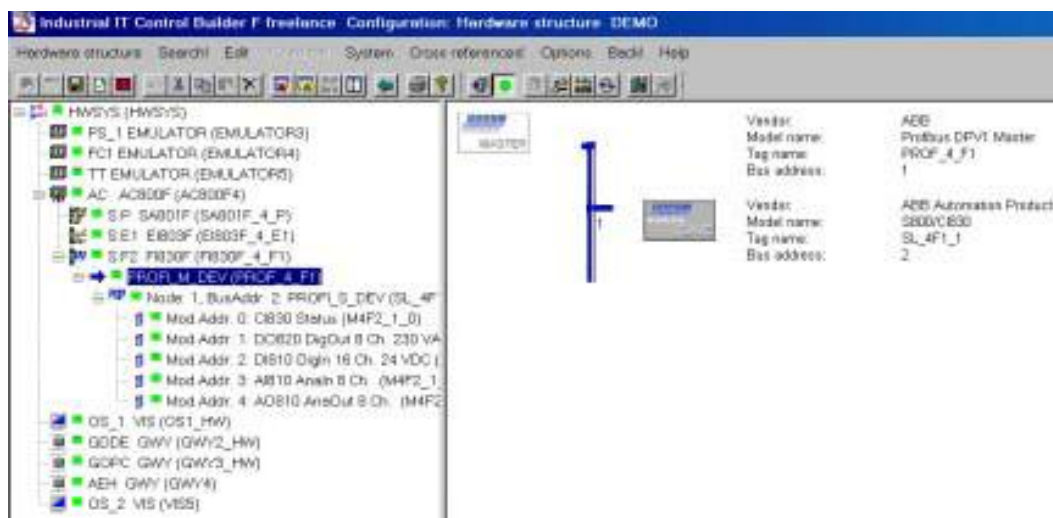


图 5-1 硬件结构组态界面

硬件结构组态

二、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Hardware structure	硬件结构菜单	System	系统菜单
Save	储存	Variable list	调变量表
Check	程序检查	Tag list	调标签表
Header	调出对象标题页	Structure data types	增添新的数据类型
Comment	编辑说明文字		
Network	网络组态	Edit	编辑菜单
Communication configuration	通讯连接组态	Templates	模板组态
Exit	退出 SFC 组态界面	Insert	插入新的对象
		Resource allocation	资源放置
I/O Editor	I/O 通道变量放置表	Parameters	组态对象参数
Cross references	启动交叉参考功能		

三、操作员站硬件结构组态 项目树下组态的操作员站资源(D-OS)应与实际的操作员站——台式计算机——对应，

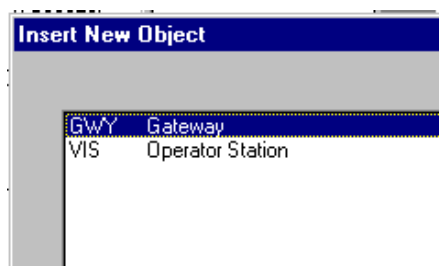
实际的操作员站需要配备显示器、打印机、键盘等，我们就需要将这些硬件设备在硬件结构中做相应的组态设置。

1、选择操作员站资源

用鼠标双击操作员站放置位置，或  → Edit → Insert，弹出选项对话框，根据资源类型选择站类型。

GWY: 表示站类型为网网站

VIS: 表示站类型为操作员站



2、放置操作员站号

我们可以用鼠标双击图 5—2 中台式机——操作员站下方的灰色区域，或选中此灰色区域后 → Edit → Resource allocation，弹出图 5—2 所示的资源站号放置对话框，选择相应站号 OK。

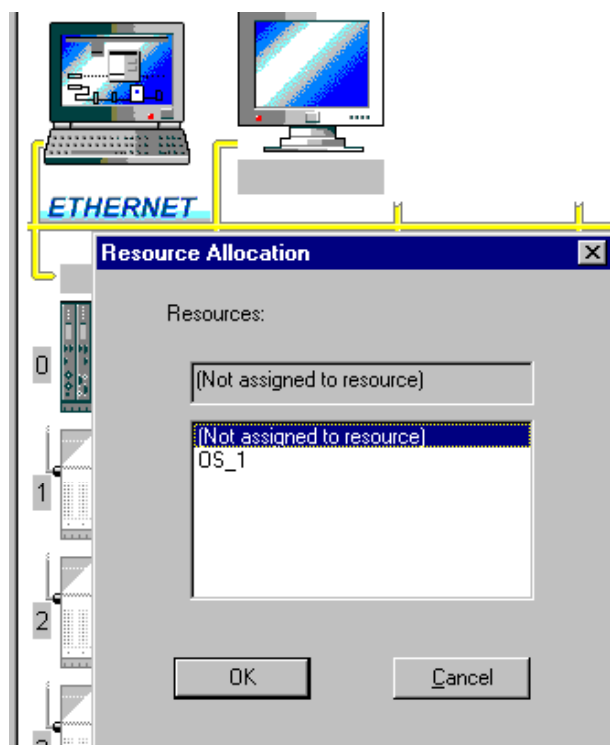


图 5—2 组态操作员站站号

硬件结构组态

3、组态操作员站外围设备

用鼠标双击硬件结构树中操作员站的站名或右键 → Parameters …，弹出图 5—3 对话框，组态操作员站的打印机、键盘、鼠标等元器件。



图 5—3 操作员站外围设备组态

四、AC 00F 硬件结构组态

项目树下的过程级 (D—PS) 资源应与实际的过程控制级相对应；实际安装过程控制级时，需要选择电源模件型号，将模件插入机架的指定槽位中，连接过程 I/O 设备，设置通讯速率等等，这些工作也需要在组态工具的硬件结构中做相应的设置。

1、放置过程级资源

用鼠标选中过程级放置的位置，鼠标双击此区域或用  → Edit → Insert，弹出对话框，其中：

AC 800F：表示站类型是控制器 AC 800F

AC 800FR：表示站类型是冗余控制器 AC 800F

2、放置 AC 00F 站名

鼠标双击图 5-4 中过程级上方的灰色区域，或选中此灰色区域用  → Edit → Resource allocation，弹出图 5-4 所示的资源站号放置对话框，选择相应站号 OK 即可。

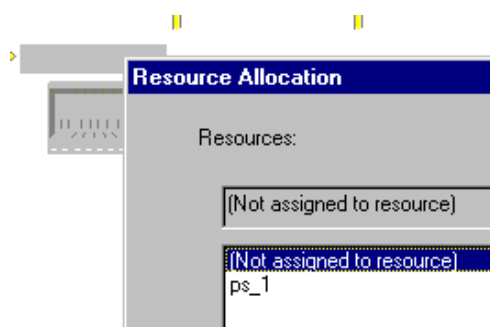


图 5-4 过程级站号放置对话框

3、放置模块

鼠标双击 AC 800F，进入过程级硬件组态界面，双击机架模块放置槽位，弹出图 5-5 所示模块类型选项对话框，按照模块实际放置表，选择相应的模块放置在指定的槽位中。

硬件结构组态

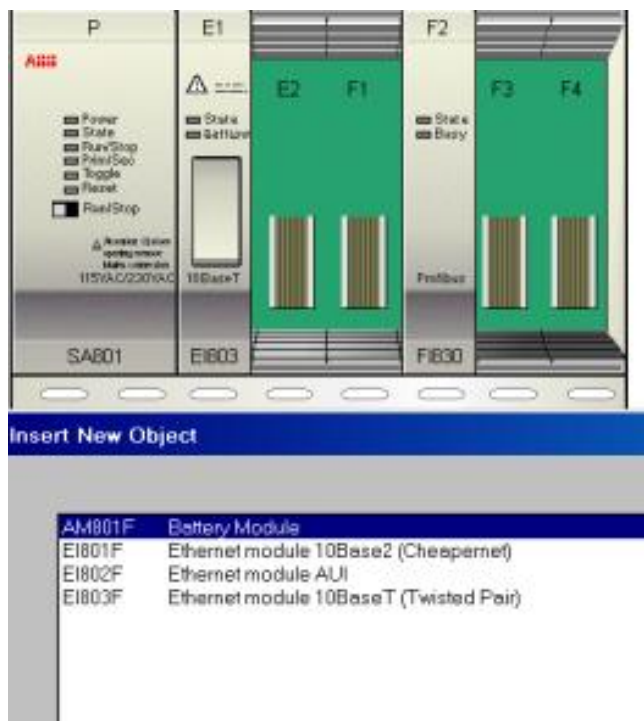


图 5—5 模块类型选项对话框

4、放置 Profibus master

Profibus master 应放置在 AC 800F 中 Profibus 模块 (FI830F) 下面，方法如下：

- ☞ → 选中硬件结构浏览树中的 Profibus 模块 (FI830F) → Edit → Insert
- 从表中选中 Profibus master (PROFI_M_DEV) → OK

或 → 选中 AC 800F 中的 Profibus 模块 (FI830F)

- 选中现场总线电缆双击
- 从表中选中 Profibus master (PROFI_M_DEV) → OK

5、放置 Profibus 从设备

每个 Profibus master 节点下允许最多放置 125 个 Profibus 从设备，放置方法如下：

- ☞ → 从硬件结构浏览树中选中 Profibus master (PROFI_M_DEV) → Edit → insert
- 从表中选中 Profibus 从对象 (PROFI_S_DEV)
- 任意安排总线地址

→ 选中 Profibus 从设备的设备数据库文件(GSD)

→ 选择开始引入

引入的 GSD 文件中不包括从设备的位图图形，如果需要可以从相应的位图文件引入相关的从设备图形。

6、放置 Profibus 从设备的子模块

如果 Profibus 从设备中包括子模块，需要按照子模块的实际位置放置子模块。

- ✎ → 从硬件结构浏览树中选中 Profibus 从设备节点
- Edit → Insert
- 从模块选项表中选中相关子模块 → Insert

7、组态过程级参数

双击硬件结构浏览树中的过程级，弹出图 5—6 所示标题页，组态过程级的相关参数。

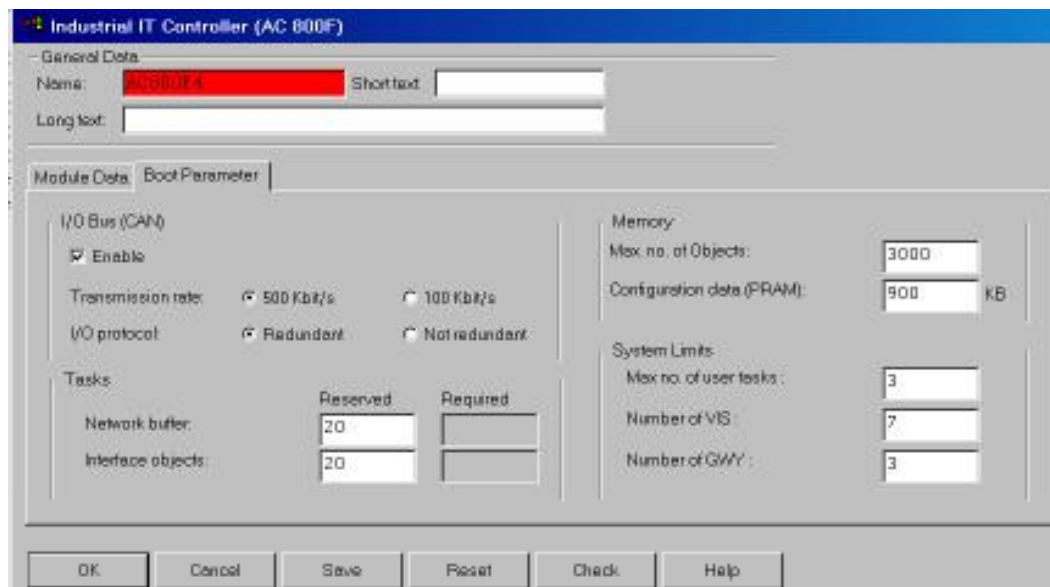


图 5—6 过程级参数组态标题页

通常情况下，默认的参数不用修改。

AC800F 4M 内存，分为以下部分。

- FLASH 内存

硬件结构组态

- 操作系统
- Boot-EPROM 可擦可编程只读存储器

RAM 内存包含以下几项

- 对象目录
- PRAM
- 冗余内存
- 对象数据
- 通讯管理

其中 PRAM 的大小、对象的内存，甚至与冗余内存都可调整。

I/O CAN 总线

传输速率有两种选项。

传输速率	最后一个 I/O 站的距离
500K bit/s	80m
100 K bit/s	400m

I/O 协议

冗余： 默认选项。I/O EPROM 可冗余操作，所有的模块需要统一设置。

非冗余： 6.1 版本不支持冗余

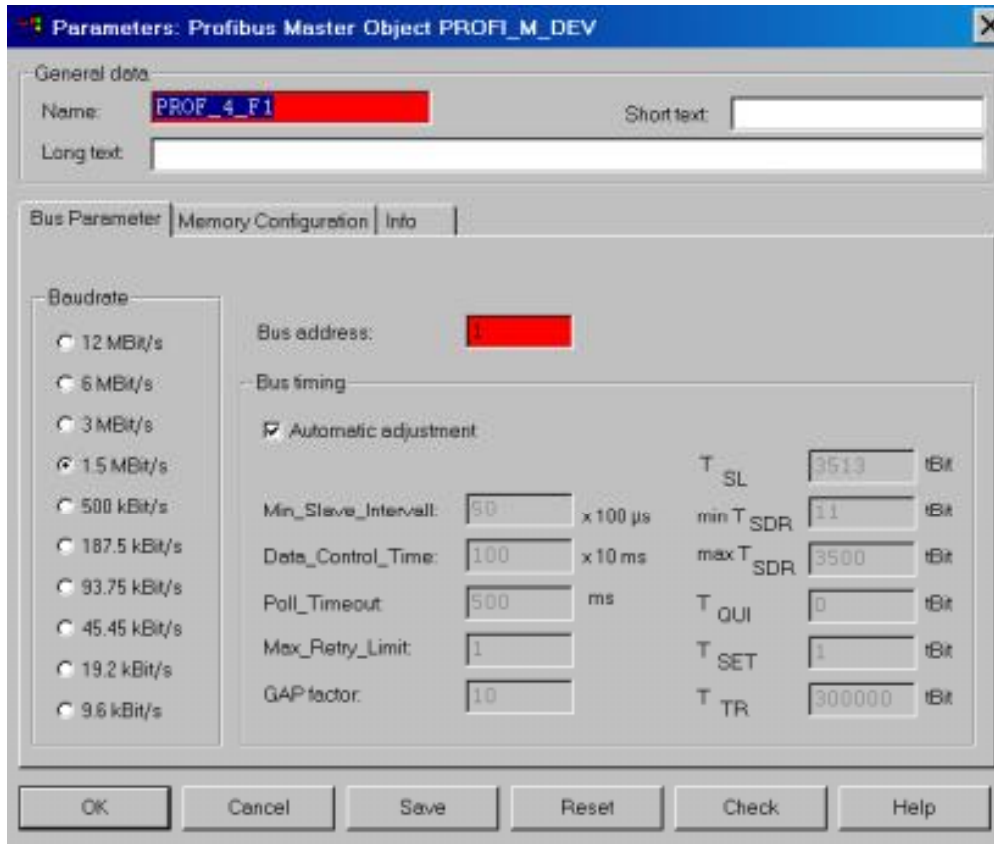
内存

最大对象数量：在过程源中允许下装的最大的对象数。

PRAM: 用于存放组态数据。

、组态 Profibus Master 参数表

双击硬件结构浏览树中下的 Profibus Master ，弹出 Profibus Master 参数表。



.1 Bus Parameter 参数表

- Baudrate 组态通讯速率
- Bus address Profibus master 的总线地址
- Bus timing Profibus master 总线通讯时间，允许依据设备间的连接自动计算总线时间参数或人为组态。

.2 Memory Configuration 参数表

硬件结构组态

Parameters: Profibus Master Object PROFI_M_DEV

General data:

Name: PROF_1_F1 Short text:

Long text:

Bus Parameter Memory Configuration Info

Bus station addresses:

Lowest bus address: 1

Highest bus address: 125

Number of I/O bytes per slave:

Max. no of input bytes: 244

Max. no of output bytes: 244

Diagnostic:

Number of diagnostic buffer: 128

Max. diagnostic length per message: 128

OK Cancel Save Reset Check Help

- Bus station address

最低总线地址 Profibus-DP master 允许控制的最低总线地址

最高总线地址 Profibus-DP master 允许控制的最高总线地址

- Number of I/O bytes per slave

最大输入字节数量 每个子模块允许被处理的最多输入字节

最大输出字节输量 每个子模块允许被处理的最多输出字节

- Diagnostic

诊断缓冲数量 每个子模块允许的最大诊断缓冲器数量

每个诊断信息最大长度 每个子模块允许被处理的最多诊断字节

、组态 Profibus Slave 参数

所有的子模块都包括其相关的参数，我们可以选中子模块，用 **Edit → Parameters** 调出如图 5—7 所示的 Profibus Slave 参数表，或双击子模块调出此参数表。

图 5—7 Profibus Slave 参数表

Definition

Bus address Profibus master 设置的 Profibus Slave 模块的地址，此地址应与 CI830 一致。

No data in clear state 用于定义在“Clear”状态时是否允许 Master 从 Slave 读取数据

- ☒ 从 Slave 中读不到数据
- ☐ 允许从 Slave 中读数据

硬件结构组态

注: **Clear** 此状态下, DPM1 读取 PROFIBUS-DP slaves 的输入信息, 同时保持 **Slave** 以故障安全值模式输出。

s

此表中显示的信息是由设备的“GSD”文件决定的, 仅仅作为显示, 不允许修改和组态。

Info

此表中显示的信息是由设备的“GSD”文件决定的, 仅仅作为显示, 不允许修改和组态。

ser

此表中显示的信息是由设备的“GSD”文件决定的。

DP

此表中显示的信息是由设备的“GSD”文件决定的。

orce s

此表用于过程调试时使用。

强制值: 可以用于定义包括输入与输出类型的所有 **component** 的强制值。

替代值: 仅可以用于定义输入类型的 **component** 的替代值, 即假定与 Profibus 设备循环数据交换故障时的替代值。

10、I/O 编辑

现场过程设备通过信号线连接到 AC 800F 的 Profibus I/O 设备或 I/O 模件的接线端子上, 从版本 4 以后, 每一路输入输出过程信号应与组态中的 I/O 元件 (**component**) 对应, I/O 元件可以直接用于控制过程组态和图形组态。

选中 Profibus I/O 设备或 I/O 模件, 调出 I/O editor 表。

I/O editor 中变量的调用支持用 F2 键调出变量表。

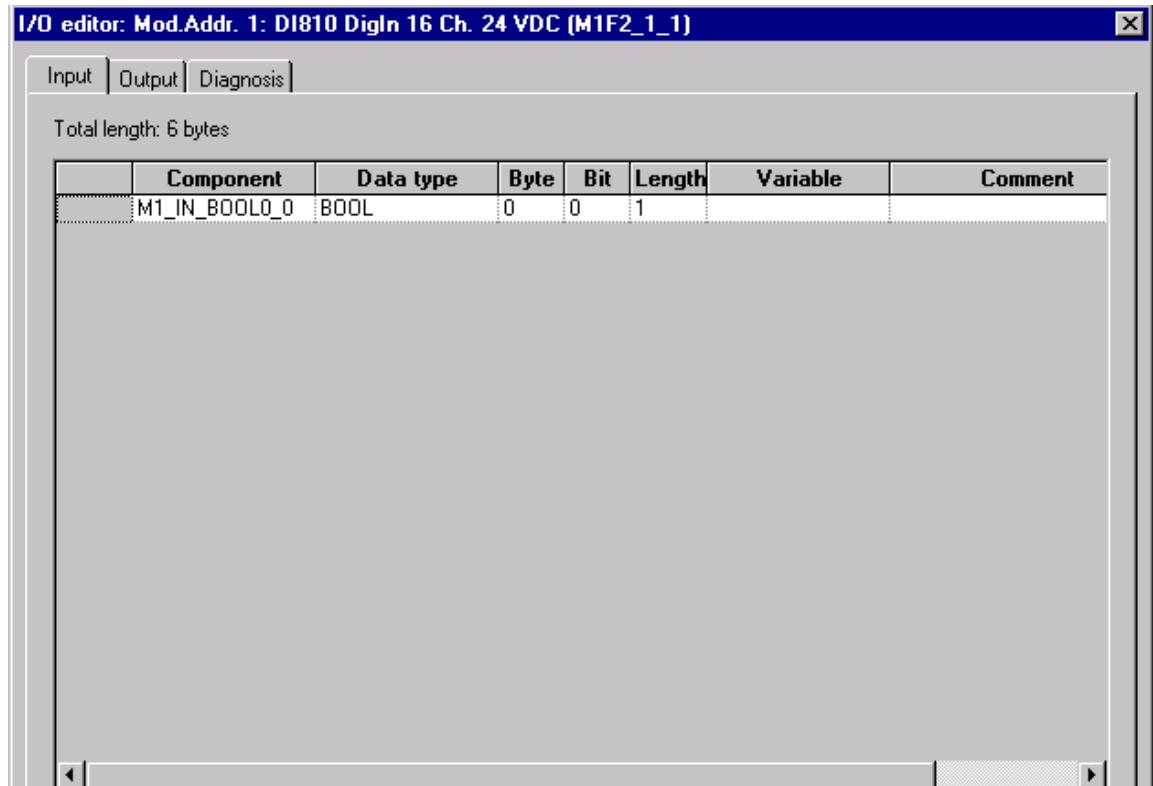


图 5—8 I/O editor 表

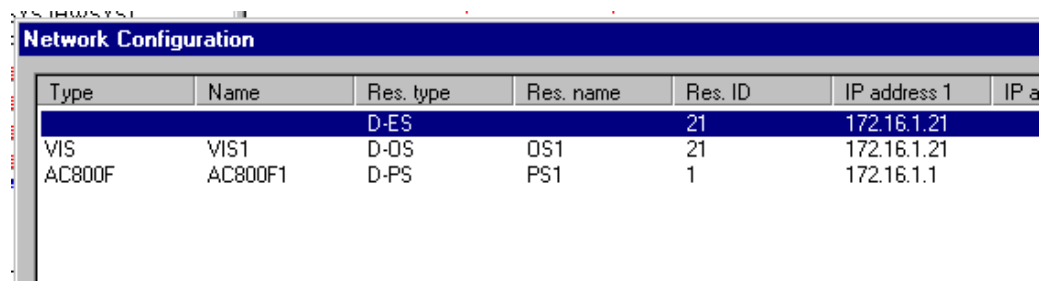
五、网络与通讯组态设置

在一套完整的控制系统中，操作员站、过程级(AC 800F)、工程师站间的数据交换是通过网络完成的；我们所使用网络遵循的协议是：以太网的 TCP/IP 协议；按照此协议的要求，在实际安装过程中，每一个站资源需要赋予 IP 地址，即在实际的站资源间建立网络连接的同时，还需要在硬件结构组态中建立相应的网络软连接。

🏠 → Hardware structure → Network

调出图 5—9 所示网络组态设置对话框，按照实际设置站 ID 与 IP 地址。

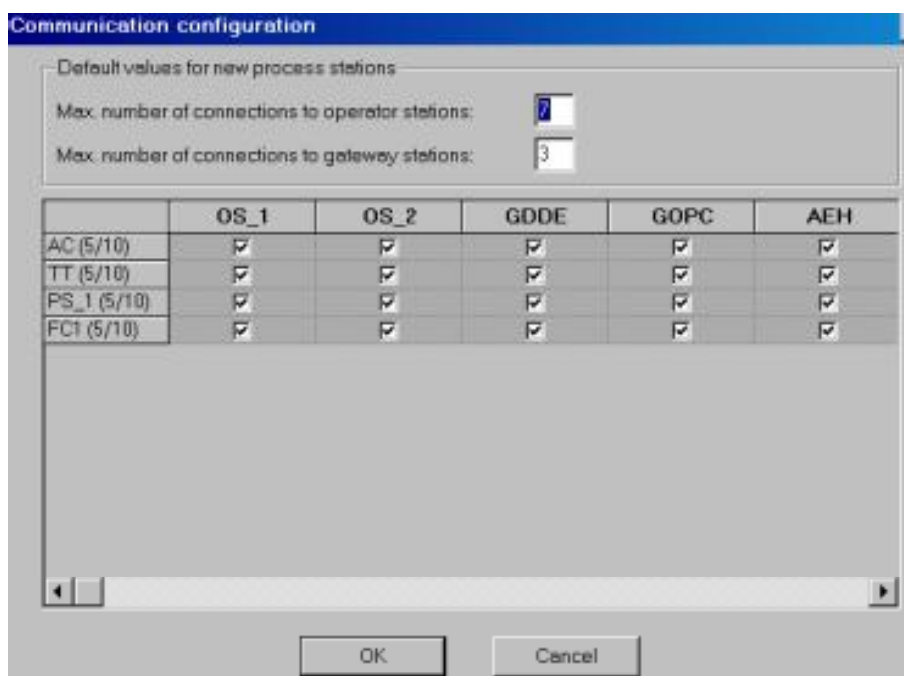
硬件结构组态



Type	Name	Res. type	Res. name	Res. ID	IP address 1	IP a
		D-ES		21	172.16.1.21	
VIS	VIS1	D-OS	OS1	21	172.16.1.21	
AC800F	AC800F1	D-PS	PS1	1	172.16.1.1	

图 5—9 网络组态对话框

网络通讯组态



Default values for new process stations

Max. number of connections to operator stations:

Max. number of connections to gateway stations:

	OS_1	OS_2	GDDE	GOPC	AEH
AC (5/10)	☒	☒	☒	☒	☒
TT (5/10)	☒	☒	☒	☒	☒
PS_1 (5/10)	☒	☒	☒	☒	☒
FC1 (5/10)	☒	☒	☒	☒	☒

OK Cancel

图 5—10 所示资源通讯组态对话框

☰ → Hardware structure → Communication configuration

调出图 5—10 所示资源通讯组态对话框，按照实际站资源间的通讯情况设置。

练习

在硬件结构组态中完成过程站、操作员站、通讯网络、模件及 I/O 通道的组态。

如果过程站选用的是机架式，要求把变量 Ana_Input1 放置在模件 DAI01 的 0 号通道，信号

范围设置为 4--20mA，循环时间设置为：100ms。

如果过程站选用的是 AC 800F，要求把变量 Ana_Input1 放置在 S800 的模拟量输入模件的 0 号通道，信号范围设置为 4--20mA。

第六章 调试

调试

当一个项目的过程级组态已经完成并通过正确性检查后，可从组态方式进入调试方式，在调试方式下完成过程级程序的下装和组态程序的调试工作。

一、用户界面结构

1、启动调试功能

☞ → Project → Commssioning 或 ☞ → 菜单行 → Commissioning 后进入调试界面，如图 6—1 所示。调试界面包括：标题行、菜单行、工具棒、状态行。

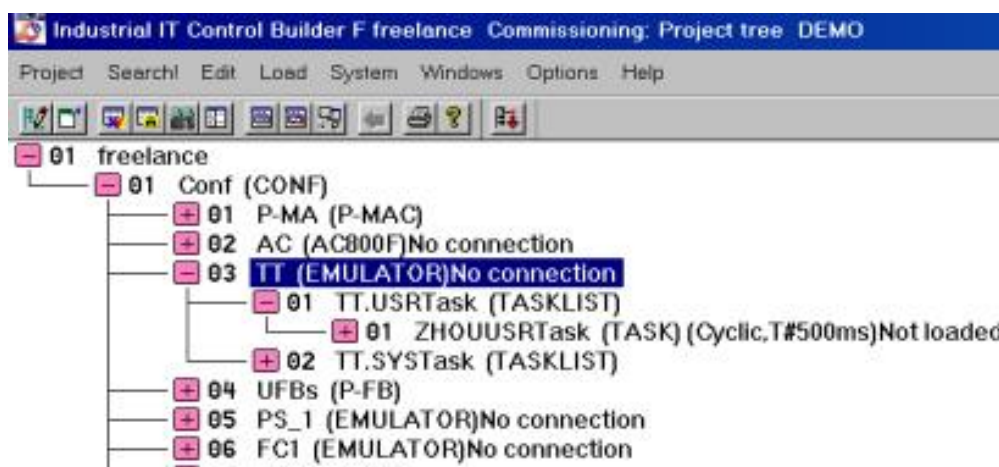


图 6—1 项目树的调试界面

2、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Pro ect Program	项目程序菜单	System	系统菜单
Header	调出对象标题页	Variable list	调变量表
Configuration	进入组态界面	Tag list	调标签表

调试

英文菜单	中文说明	英文菜单	中文说明
Exit	退出调试界面	Hardware structure	调系统硬件结构界面
		Global message processing	定义操作员站报警信息的确认方式
Edit	编辑菜单	Local message processing	定义本地操作员站报警信息表的格式
Expand	打开光标选定下层分支		
Full Expand	打开光标选定下层的所有分支	Windows	视窗菜单
Compress	收拢分支	Define debug windows	定义调整视窗
		Define value window	调出数值视窗
Load	下装菜单	Define trend window	调出趋势视窗
Whole station	下装整个资源站		
Variable	下装组态变量	Options	选项菜单
Message configuration	下装报警信息组态	Hard copy	打印当前页
Selected objects	下装被选择的对象	Set system time	设置系统时间
Change objects	下装改变的对象	Long state line	状态行以长文本格式显示
Version information ...	调出版本信息		

3、资源状态指示说明

1)、过程级与操作员级的状态说明

进入调试状态有以下要求：

- 以太网卡已经安装在工程师站上

调试

- **TCP/IP 的驱动**
- **以太网线连接到要调试的过程站**
- **站的 IP 地址唯一**
- **组态中站的地址组态正确**
- **EPROM 已经下装到模件**

No connection: 表示过程级 (D-PS) 或操作员站 (D-OS) 尚未与工程师站连接。

No operation system: EPROM 没有下装到模件。

Stop: 表示过程级停止工作。

- **Stop[HW]:** 表示由硬件停止过程级的工作。
- **Stop[SW]:** 表示由软件停止过程级的工作。

Running: 表示过程站运行正常。

Version error , Running: 表示过程站上电运行的软件版本与组态的不同。

2)、任务项下的状态说明:

Stopped: 表示任务停止工作。

Running: 表示任务运行正常。

No loaded: 表示整个过程级没有下装组态程序而导致任务程序没有被下装。

3)、数据库变量状态说明:

CLEAN:	说明此变量正确并已下装过程级。
DIRT :	说明此变量与过程级上的变量版本不同。
CREATE:	此变量尚未下装到过程级。
DELETE:	变量已从数据库中删除，但仍保存在过程级中。

二、系统时间、当地时间、时区的设置

1、名词解释

系统时间: 是过程站中的时间，用于系统内部各资源站间数据传输时的时间标记，即格林威治时间。

当地时间: 是过程级或操作员站所在地的时间。

当地时间 = 系统时间 - 时区

时 区: 是当地时间与系统时间的时间差。

2、设置时区

同一个控制系统中的每一个资源 (D-OS、D-PS) 都应该具有相同的时区，过程级的

时区由项目树组态设置，并通过下装组态时的引导程序完成对过程级资源站时区的设置；工程师站与操作员站的时区设置分别在各自的计算机上完成：

 Start → Settings → Control Panel → Date / Time

3、时钟同步

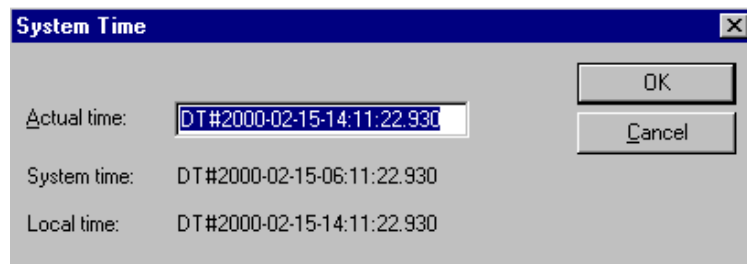
当系统运行时，所有资源站的时钟应统一为一个时间——系统时间，即格林威治时间；主导者为最低站号的过程站，每隔一分钟由主导者同步系统中的其它站资源一次；精确值可达到 3ms，最大差异允许为 5ms。

整个控制系统的系统时间由工程师站设置完成：

调试

☰ 选项 → 设置系统时间

如果不存在夏时制，实际时间即是当地时间。



三、程序下装

在下装项目组态程序时，既可以下装整个过程站或整个操作员站，也可以只下装程序的某一部分。

1. 下装整个站资源

光标选中项目树中要下装的站资源：

☰ Load → Whole station

2. 下装变量

☰ Load → Variables

3. 下装信息组态

☰ Load → Message configuration

4. 下装所选中的对象

☰ Load → Selected objects

5. 下装改变的对象

☰ Load → Changed objects

四、调试功能

(一)、过程级模件状态显示

在组态方式下，我们通过硬件结构组态完成了过程级 (AC800F) 的网络通讯、模件放置与通道变量等的组态，但是组态正确与否并不能在“组态”方式下面直观显示出来，必须进入调试方式下，我们通过模件的背景色，才能判断模件组态放置正确与否。

 System → Hardware structure

1、现场总线模件的状态

2. 主模件、子模件的状态显示

3. 模件产品信息报告

选中模件，进入模件产品信息页，可以获得模件的产品系列号、硬件版本(模件索引号)、软件版本 (I/O EPROM)、运行时间等信息。

现场设备通道的输入/输出强置

在调试时，有些现场信号尚未与现场设备通讯，所需信号可采用强置方式接入组态程序进行调试。允许单通道强置或多通道同时强置。

强置方法：选中要强置的模件，双击进入模件的参数表，选择 **FS**，强置通道即可。

调试

General data

Name: **M4P2_1_1** Short text:

Long text:

Module data | User | DPV1 | PS

I/	Component	Data type	Hold id	Substitute value	Force active	Force value
O	M1_OUT_BYTE2	BYTE			☐	255
O	M1_OUT_BOOL1_0	BOOL			☐	☐
O	M1_OUT_BOOL1_1	BOOL			☐	☐
O	M1_OUT_BOOL1_2	BOOL			☐	☐
O	M1_OUT_BOOL1_3	BOOL			☐	☐
O	M1_OUT_BOOL1_4	BOOL			☐	☐
O	M1_OUT_BOOL1_5	BOOL			☐	☐
O	M1_OUT_BOOL1_6	BOOL			☐	☐
O	M1_OUT_BOOL1_7	BOOL			☐	☐

OK Cancel Save Reset Check Help

4. 模块的安全值设置

当系统出现供电故障或 CPU 启动上电时，输出模块以安全值输出。

(二)、调试 FBD 图

在调试方式下进入 FBD 图，其结构与组态方式无任何区别，只是布尔值(二进制值) 0 或 1 的逻辑状态直接由其线型表示出来；其次鼠标点中信号线后会显示其信号值。

逻辑 1 (TRUE) 用实线表示 _____

逻辑 0 (FALSE) 用虚线表示 -----

1、修改功能块参数表中的参数值

仅允许修改功能块参数表中白色区域的参数值。

- **Write** 对变量赋值。
- **Correct** 修改组态值。

2、定义数值视窗与趋势视窗

组态调试时，我们有时不仅需要看到变量的数值变化，还需要看到变量的趋势；我们可以使用数值视窗功能与趋势视窗功能来达到不同的监视需求。

首先选中被监视变量，双击此变量或用
窗口 → 输入值，弹出 调试窗口 的同时，变量被定义到调整视窗中，如图 6—2 所示。

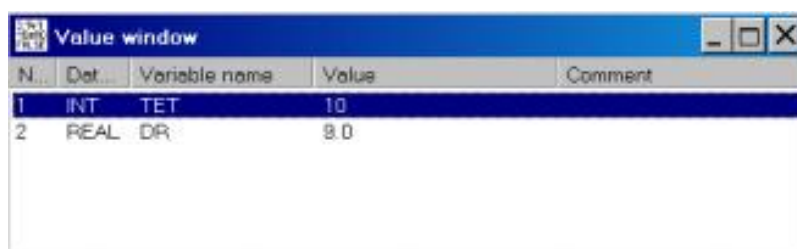


图 6—2 定义调整视窗

在调整视窗中，我们可以选择某个变量在数值视窗中以什么数据类型显示和趋势视窗中变量的显示颜色，也可以用 **UP** 和 **Down** 指令改变变量在视窗中的位置，用 **Delete** 指令删除视窗中的变量等等。

将变量定义到调整视窗后，我们就可以通过调用数值视窗监视变量的数值变化，调用趋势视窗监视变量趋势的变化

数值视窗由鼠标 → 窗口 → 显示数值窗口调出；

趋势视窗由鼠标 → 窗口 → 显示趋势窗口调出。

(三)、调试 SFC 程序

在调试方式下，SFC 程序可以由一些控制命令完成与操作员站 (D—OS) 上类似的操作，如：

- * SFC 程序的手/自动切换
- * 手动执行每一步和每一个跳步条件
- * 修改 重复时间、重启时间和等待时间

调试

- * 步 (step) 的永久执行 (Permanent on) 和永不执行 (Permanent off)
- * 跳步条件的禁止 (Blocked)、强置 (Force)、正常执行 (Normal) 各一次
- * 允许 SFC 程序执行 (Enable) 和不允许 SFC 程序执行 (Disable)

除此之外，SFC 程序中的每一个子程序的组态，都是由 FBD 图组态完成，其子程序的调试可以参照 FBD 图的调试进行。

方法如下：

1. 在调试方式下，进入 SFC 程序界面
2. 选中或激活需要调试的元素步 (Step) 或转换条件 (Transition)
3. 从菜单窗口下选择要调试的组态图，进入元素的编辑界面，按下编辑键后，进入元素的子程序调试界面。

(四)、过程级的调试

- * 全部初始化： 删除所选中的过程级中的操作系统和所有用户程序。
- * 初始化： 仅删除所选中的过程级中的用户程序，而操作系统并不受影响，此时输出模件的通道输出值为 0 电压/电流。
- * 冷启动： 此时系统任务中的冷起动任务执行一次，所有过程的写入值均丢失，输出模件的通道输出值为组态好的安全值。
- * 热启动： 此时项目树下的系统任务中的热起动任务执行一次，输出模件的状态、能块参数值和当前变量都不受影响。
- * 启动： 起动选中的过程级及下属的所有程序。
- * 停止： 停止执行选中的过程级及下属的所有程序。

练习

对所有过程站组态程序进行调试：

1、项目树在组态方式和调试方式下有什么区别？

- _____
- _____
- _____

2、调试 FBD 图程序

调试方法为：

3、功能块中什么样的参数可以修改？

4、调试 SFC 程序。

第七章 标签表 (Tag list)

标签表

标签表是用于存放一个组态项目中所有标签的数据库，标签表中的标签可供用户在一个项目组态中随时调用；数据库中的标签可以在项目组态时自动生成，也可以由用户直接在标签表中填写或从其它项目导入生成。

一、调出标签表

不论是从项目树下还是从 FBD、SFC 及硬件结构组态界面下，用下述方法可调出标签表  → 系统 → 标签列表

二、标签表的结构

标签表的结构如下：

Name	T	A	R	Short text	Long text	Type	name	L	P
AI_TR1	S	-	+	Analog	UNIT transf...	AI_TR	S	@	
Catalog	T	-	?			HWSTR	S	#	
Control1	S	-	+	Controlle...	Controller1...	C_CS	S	@	
Count1	S	-	+	Counter1		CT_ANA	S	@	
DAI01_1_0_1	S	-	?			DAI01	S	@	
DAI01_1_0_2	S	-	?			DAI01	S	#	
DA001_1_0_3	S	-	?			DA001	S	#	
DCP02_1_0_0	S	-	?			DCP02	S	@	
DDI01_1_0_4	S	-	?			DDI01	S	#	
DD001_1_0_5	S	-	?			DD001	S	#	
DLM01_1_0_L	S	-	?	DLM01	LIANJIEMOJI...	DLM01	S	@	
DLM01_1_1_L	S	-	?			DLM01	S	#	
Dely	S	-	+	simulation		DELAY	S	@	
HWSYS	S	-	?			HWSYS	S	#	
IDF450	S	-	+		individual ...	IDF_1	S	@	
Monitor1	S	-	+	Analog	Analog moni...	M_ANA	S	@	
Monitor2	S	-	+	Binary	Binary moni...	M_BIN	S	@	
Mono1	S	-	+	Monoflop		MONO_F	S	@	
PC452	S	-	+	Pressure		C_CU	S	@	
PS1	S	-	?			PS	S	@	

Name: 即标签名，是功能块的名字，Max.12 个字符。

T:	标签类型说明。 S 标准名称 即功能块的名字，SFC 程序名字，模件名字和硬件结构对象的名字等的标识。 F 形式名称 用于自定义功能块名字的标识。 T 模板名称 用于硬件结构中模板名字的标识。
A:	标签所在的厂区，最多可定义 15 个（A……O）厂区。
R:	标签的状态。 + 表示功能块允许处理（即功能块的 Processing ☐V） — 表示功能块不允许处理（即功能块的 Processing ☐） ? 表示功能块的处理与否未确定（即功能块的 Processing ☐三） ☞ 顺序功能块图和 I/O 模件的状态总是用“?”表示。
Short text:	Max.12 个字符
Long text:	Max.30 个字符
Type name:	功能块类型的缩写，如 M—ANA
L:	功能块库的类型 注： S：标准功能块库 U：用户功能块库 E：特殊功能块库（特指 SFC 程序）
P:	# 表示功能块未作 check;

标签表

@ 表示功能块已作 check (此栏在标签表中不允许修改)。

三、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Tag list	标签表菜单	Search	查询菜单
Sort	定义标签在标签表中的排列方式	Define	定义查询方式
Normal view	常规浏览	Activate F3	启动查询功能
Station view	资源站浏览		
Exit	退出标签表		

Edit	编辑菜单	Cross references	启动交叉参考功能
Undo	返回操作前状态		
Insert new tag	在标签表中插入新标签	Options	选项菜单
Edit field	选择标签插入位置	Hardcopy	打印当前页
Delete Field	删除选定区域中的文字	Color	允许选择标签表中颜色的显示方式
Delete unused tags	删除未使用标签		
Export block	标签库的引出文件	Back	返回进入前的界面
Import block	标签库引出文件的引入		
Station access	定义资源站对标签数据的存取权限		
Area	划分标签的厂区		

第八章 变量表 (Variable list)

变量表

变量表是用于存放一个组态项目中所有变量名的数据库，变量表中的变量可供用户随时调用；数据库可以在项目组态时自动生成，也可以由用户直接在变量表中填写或从其它项目导入生成。

一个组态项目下的所有变量都存放在变量表中。

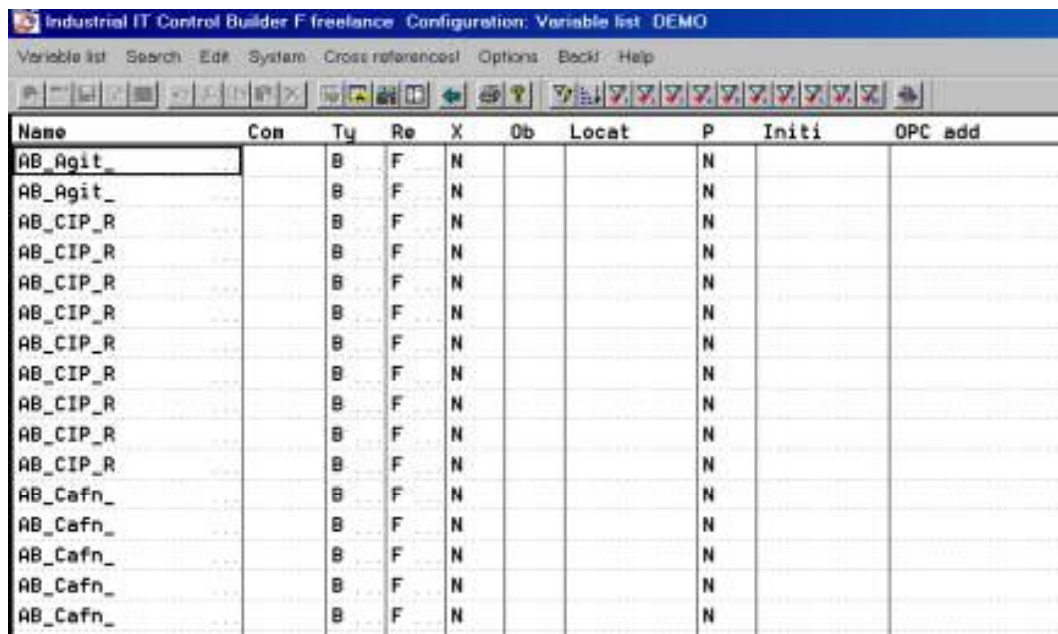
一、调出变量表

不论是从项目树下还是从 FBD 图、SFC 程序或硬件结构组态界面下，用下述方法可调出变量表。

☰ 系统 → 变量列表

二、变量表的结构

变量表的结构如下：



Name	Con	Ty	Re	X	Ob	Locat	P	Initi	OPC add
AB_Agit		B	F	N			N		
AB_Agit		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_CIP_R		B	F	N			N		
AB_Cafn		B	F	N			N		
AB_Cafn		B	F	N			N		
AB_Cafn		B	F	N			N		
AB_Cafn		B	F	N			N		
AB_Cafn		B	F	N			N		

表量表

Name:	变量名，Max.16 个字符
Comment:	变量名注释，Max.34 个字符
TYPE:	变量的数据类型，如 REAL、INT 等。
D-PS:	变量所属的资源站号
X:	Y 表示此变量与其它资源有关 N 表示此变量仅与自身的资源有关
Module type:	变量所属模件的类型，如 DDO01
Slot:	模件所在机架的槽位号；如 1/8 表示机架的 ID 为 1，槽位号为 8，此栏不允许更改。
Cha:	变量所在模件相关通道号，此栏不允许更改
P:	Y 表示变量来自与过程成像（即 process image ☐ V ☐ ） N 表示变量来自 I/O 模件（即 process image ☐ ☐ ）
Initial value:	当过程站冷起动时的默认值

三、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Variable list	变量表菜单	Search	查询菜单
Sort	定义标签在标签表中的排列方式	Define	定义查询方式
Normal view	常规浏览	Activate F3	启动查询功能
Station view	资源站浏览		
Exit	退出变量表		

表量表

Edit	编辑菜单	System	系统菜单
Undo	返回操作前状态	Structured data types	定义新的数据类型
Insert new variable	在变量表中插入新变量	Tag list	调出标签表
Field	选择变量插入位置	Hardware structure	访问硬件结构
Delete Field	删除选定区域中的文字		
Delete unused variable	删除未使用的变量	Cross references	启动交叉参考功能
Export block	形成变量的引出文件		
Import block	引入变量的引出文件	Options	选项菜单
Station access	定义资源站对变量数据的存取权限	Hardcopy	打印当前页
Assign resource to block automatically	自动为引入变量安排资源站	Color	设置变量表颜色
Assign resource to block manually	手动为引入变量安排资源站		
Assign process image block	为引入变量安排过程成像		

第九章 操作员站组态介绍

操作员站组态

一、画面访问组态介绍

画面访问功能是一种方便、快捷的调出画面的手段，它允许操作员在操作员站上由某个特定的标签直接进入与其相关的画面，如：用户图形画面、趋势画面、SFC 画面、组画面、记录等中。

1. 调“画面访问”

如果用光标选中某个操作员站(D-OS)资源，所组态的画面访问只能对应于此操作员的特定画面；如果光标选中的站资源是公共图形库，所组态的画面访问则对所有的操作员站均有效。

☰ → System → Display access

从主菜单下调出画面访问组态对话框如图 9—1 所示。

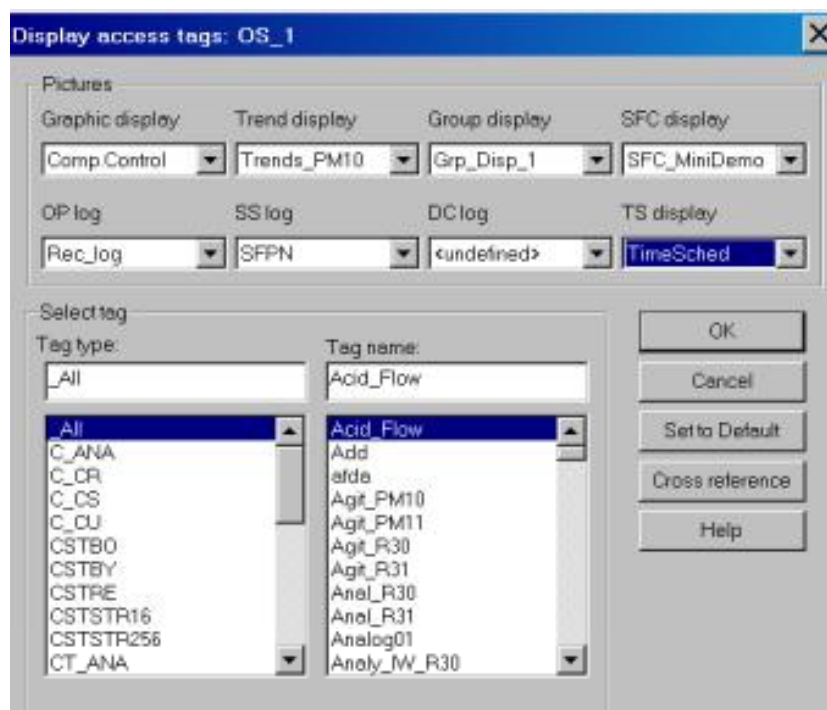


图 9—1 画面访问组态对话框

2. 组态画面访问

首先从图 9—1 中选定标签类型及标签名，其次选定所要访问的画面类型及相关画面名称即可。

二、组态总貌画面

总貌画面只允许创建在某个操作员站的下层，不允许创建在公共画面库的下层。

项目树下主菜单 → Edit → Insert next level → Overview display

在操作员站的下层建立总貌画面的分支，双击此总貌画面，弹出图 9—2 所示总貌画面组态界面。

其中

- * 标题文本：允许写入 Max.48 个字符，是操作员站总貌画面的标题行
- * 行标题：允许写入 Max.77 个字符/每行，是每一行的标题行
- * 段：用于放置画面名和记录名

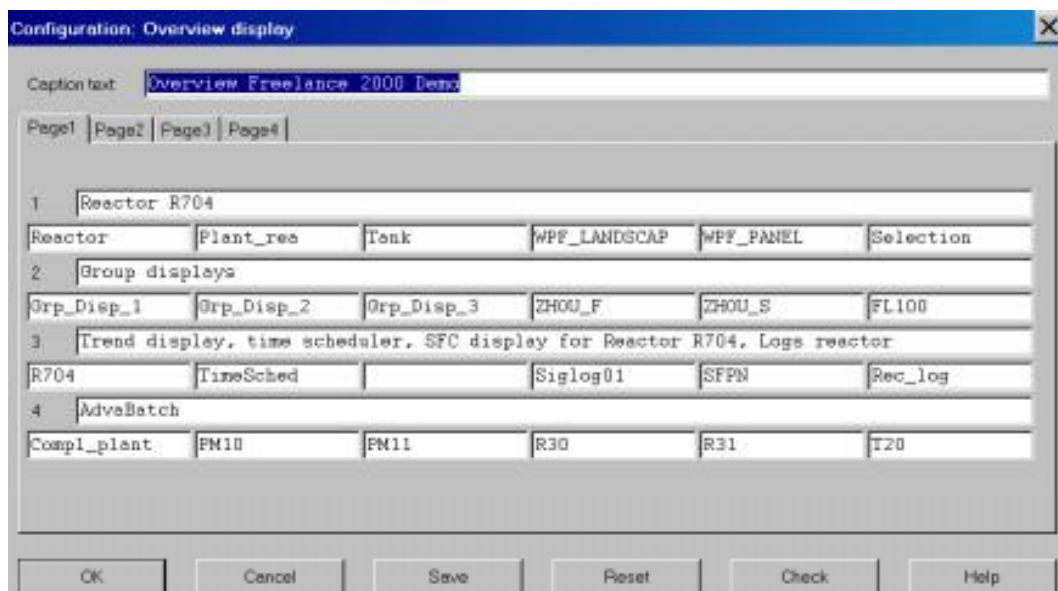


图 9—2 总貌画面组态对话框

用鼠标选中总貌画面中的 Segment，使用 F2 键调出表 9—3 所示对话框，

操作员站组态

从画面列表中选择画面名称 → OK；依次将每幅画面放置在每一个“Segment”中

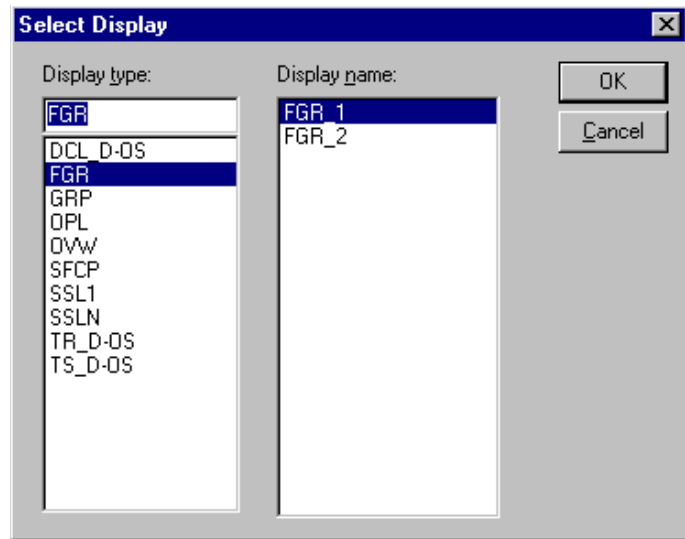


图 9—3 画面选项对话框

三、组画面组态介绍

1. 新建一幅组画面

组画面可以建在某个操作员站的下层，也可以创建在公共图形库的下层，在公共图形库中的组画面可被任何操作员站所调用。

 Edit → Insert next level → Group display

调出组画面组态对话框，如图 9—4 所示。

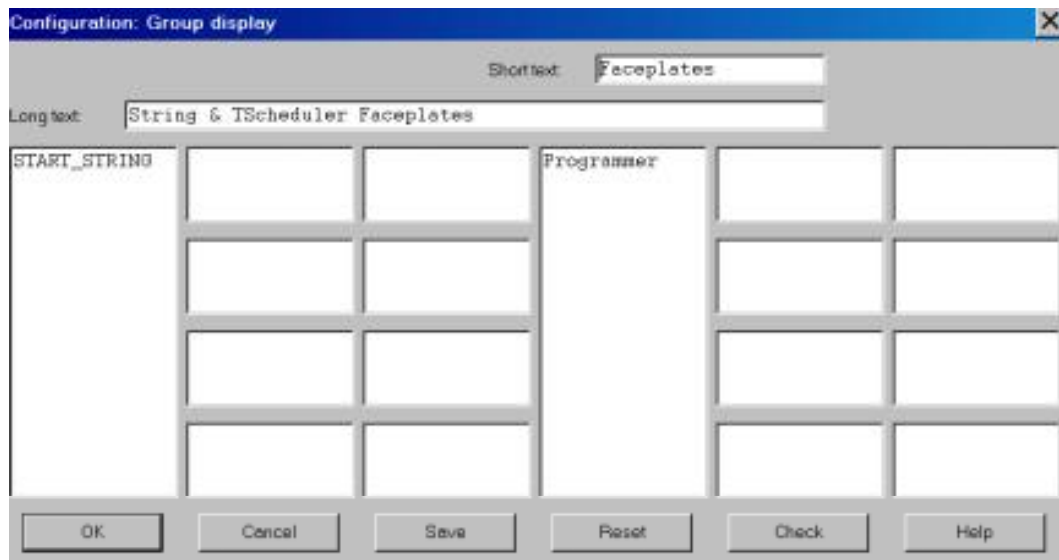


图 9—4 组画面组态对话框

2. 组态组画面

鼠标 → 双击项目树下组画面的名字，进入到组画面组态对话框图 9—4 中。

其中：

- * **Short text:** Max.12 个字符，是组画面在操作员站上的短文本注释
- * **Long text:** Max.30 个字符，是组画面在操作员站上的长文本注释
- * **段:** 放置标签区

组态时

☞ → 选中组画面某一分区 → 使用 F2 键调出图 9—5 所示标签表，选中表中标签
→ OK

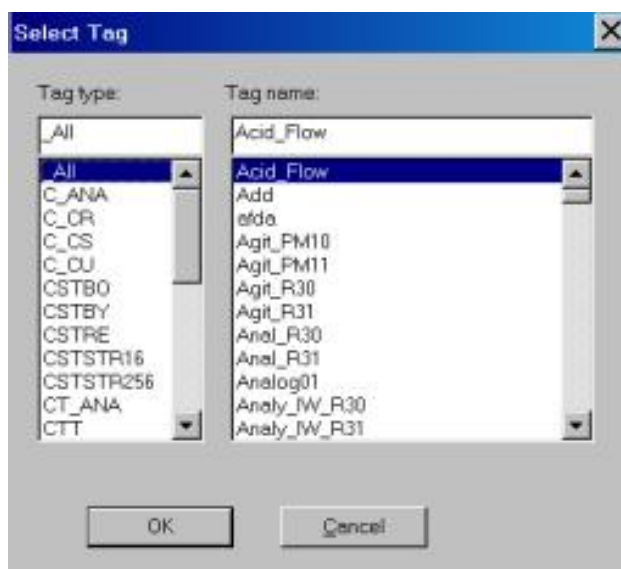


图 9—5 标签选项表

3. 编辑

- * 删除某一标签（面板）：

→ 选中某一分区 → 删除

- * 检查

→ Group display → Check

- * 退出组画面

→ Group display → Exit

四、组态趋势画面

在每一个操作员站上，最多允许组态趋势画面的数量不受限制；每一幅趋势画面中，最多允许组态 6 条趋势曲线；每一幅趋势画面应与过程站中的趋势采集功能块相对应，并且为一一对应关系。

1. 新建一幅趋势画面

趋势画面只能在某一个操作员站的下层建立，而不能放在 Common display tool 下建立。

☰ → Edit → Insert next level → Trend display

2. 趋势画面组态介绍

用鼠标双击项目树下趋势画面的名字，进入趋势画面组态表中，如图 9—6 所示。

* 第一张表 (display)

Name: 是此趋势画面的名字，只允许在项目树下更改

Short text: Max.12 个字符

Long text: Max.30 个字符

放置趋势采集的功能码

Tag name: 是趋势采集功能块的标签名，应该使用 F2 键调出。

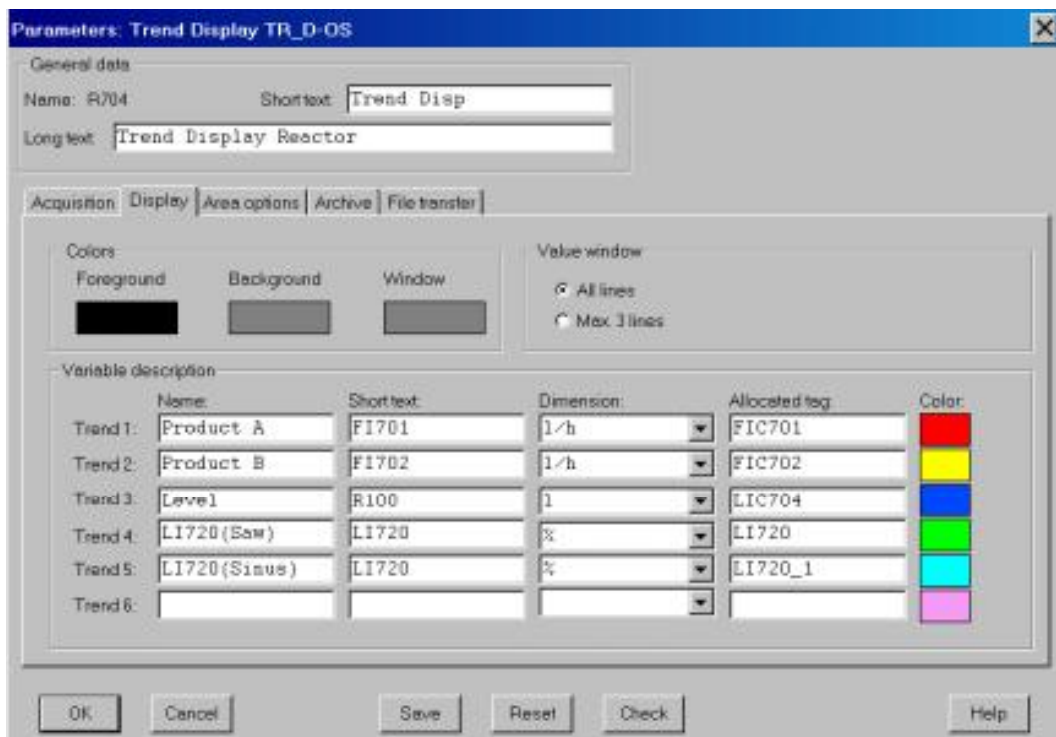


图 9—6 趋势画面组态表

Variable description

Name:	是每一条趋势曲线在操作员站上显示的名字
Short text:	Max.8 个字节，是每一条趋势曲线的短语注释
Dimension:	每一个趋势变量的工程单位
Allocated tag:	用 F2 键调出标签表，选中某个标签后放置在此区域， 操作员有权在 OS 站上双击此趋势名字时调出相关面板
Color:	为每条趋势曲线在 OS 站上的显示颜色，表中显示颜色 为默认设置

*** 第二张表: (Area options)**

Time gaps:	用于定义变量被采集记录的时间间隔
Time axis:	必填区，为趋势曲线的横坐标
Band start~Band end:	带工程单位的起点~终点
% start ~ % end:	0%~100%的坐标轴

*** 第三张表: (Archive)**

Enable archiving ☒ :	趋势变量允许归档
Delete old archive ☒ :	一旦退出 DigiVis，OS 站上的原有归档数据会被删除，趋势归档即可停止；在重新启动 DigiVis 后，趋势归档重新启动，但与原有归档记录有时间差。
Archive duration:	归档持续时间最大允许 T#24855d=68 年
Start up:	用于定义起动 DigiVis 时，是否立刻归档趋势数据
⊙Ready:	当起动 DigiVis 时，OS 站上的变量归档处于准备状态，归档的启动由趋势采集功能块的输入端 SEN 决定。
⊙on:	一旦启动 DigiVis，趋势归档就立即开始。
Archive name:	趋势归档文件的名字

* 第三张表: (File transfer)

- File transfer:** 用于把趋势归档文件从 DigiVis-PC 传送到另一台 PC 机上作备份,此机器必须安装 FTP Sever 才能得到传输数据。归档的文件是 CSV 文件, 需要 Digibrowse 软件来浏览。
- Start export as :** ☒ 当满足组态时间时, 允许循环传送数据
☐ 不允许循环传送数据
 以 DT 时间格式写入如 DT # 1997-07-22-12: 11: 43.018。
- Reexport every:** 以多长时间间隔进行数据传送时间格式写入, 如 T # 24h (即每隔 24 小时传送数据)
- ☒ incremental: 不是全部的归档数据被传送, 仅传送最后的数据
☒ complete: 所有归档数据都被传送
- 事件驱动传输** ☒ 数据传送由事件功能块触发, 用 F2 键调标签名
☐ 无事件功能块控制数据传送
- Export on event**
☒ 每次事件驱动 当事件功能块的输入有变化时, 数据被传送
☒ 每个周期 当事件功能块触发后, 数据循环被传送, 以时间格式写入。
- Acess manually export**
☒ 允许操作人员在操作员站上控制文件传送
☐ 不允许
- Target station:** 写入目的站的 IP 地址
- Patj:** 如果文件拷贝到某个特定的路径, 写明其路径如: C: \ Freelance \ archive FTP 的文件为 C: \ Freelance\archive\XXXX.001
- File:** 导出的文件存在目的站路径下的定义的文件名字下。
- Directory:** 导出的文件存在它目的站的路径下,名称为定义的名称加上日期时间。
- Directory (8.3) :** 导出的文件存在它目的站的路径下,名称为定义的名称加上日期

操作员站组态

时间。名称长度为 8 位，扩展名为 3 位。

FTP timeout: FTP 服务的同步服务，（输入在连接用户断开之前保持不活动的分钟数，默认值是 10，最大限制是 60 分钟，0 表示用户决不断开)为必填区。

User name: 用户名必填，如果可能，用 FTP 服务器检查用户名正确与否。

Password: 保证对侧只有知道密码的用户可以保存传输数据。

五、顺序功能块图组态介绍

1. 新建一幅顺序功能块图（SFC）画面

SFC 画面既可以建在某个操作员站的下层，也可以建在公共图形库的下面，如果建在公共图形库中，允许被所有的操作员站调用。

 → Edit → Inset next level → SFC display

2. SFC 画面组态介绍

鼠标双击项目树下 SFC 画面的名字，可以进入到 SFC 画面组态表 9-7 中。组态表的填写内容包括：

Name: SFC 画面名字，只允许在项目树下更改

Short text: Max.12 个字符

Long text : Max.30 个字符

Assigned SFC program

SFC program name: 必填区，用调出在 PS 站中组态好的 SFC 程序

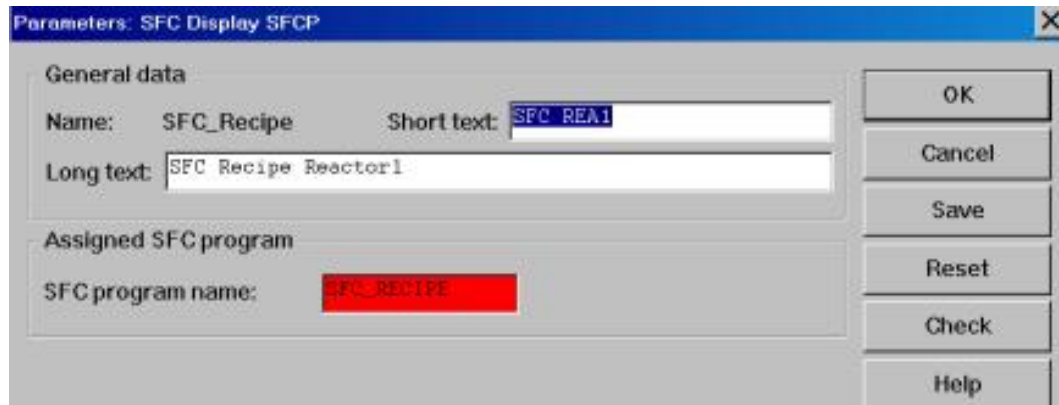


图 9—7 SFC 画面组态表

六、用户图形画面组态介绍

(一)、用户图形编辑界面介绍

1、用户图形编辑界面

如图 9—8 所示，用户图形编辑界面包括标题行、菜单行、工具棒、图形编辑区、工具箱五部分。

操作员站组态

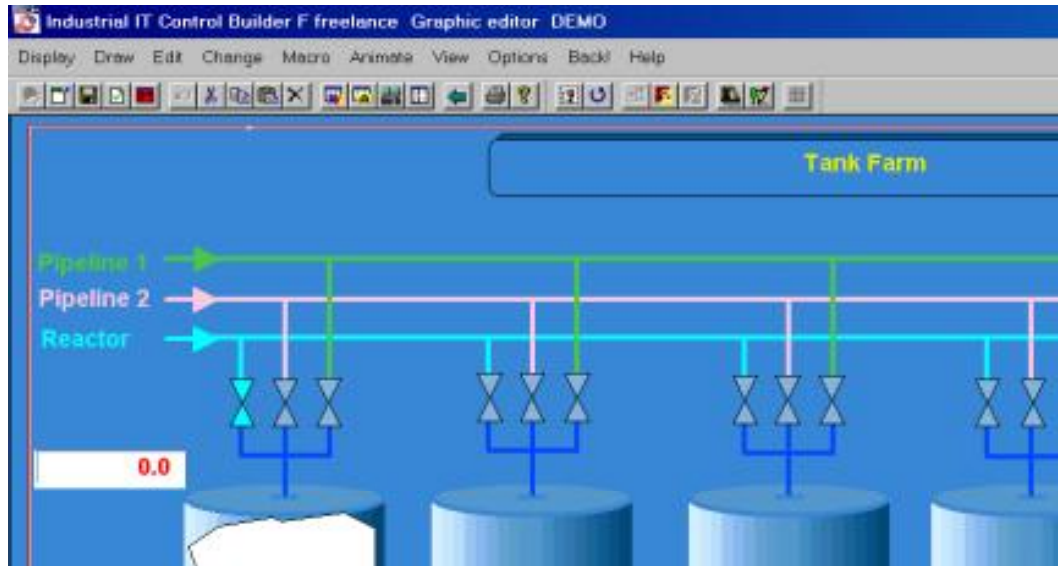


图 9—8 用户图形编辑界面

2、中/英文菜单对照表

英文菜单	中文说明	英文菜单	中文说明
Display	画面菜单	Draw	绘画菜单
Save	储存	Line	线条绘制
Check	正确性检查	Rectangle	矩形绘制
Show error list	显示错误表	Polyline	多边形线条绘制
Header	调出用户画面标题页	Polygon	多边形绘制
Export	将当前画面引出存为一个文件	Ellipse	绘制椭圆形
Import	将图形文件引入为画面图形	Arc	绘制圆弧
Exit	退出用户画面编辑	Circle segment	绘制圆弧区
		Text	写入文字
		Bitmap	调入位图
Edit	编辑菜单	Change	改变菜单

英文菜单	中文说明	英文菜单	中文说明
Return to object	返回到对象编辑参数表	Line attributes	修改线条的属性
Undo	恢复到操作前状态	Area attributes	修改区域的属性
Repeat	重复	Points	点的删除、移动
Select single object	选中单个对象	Text edit	编辑修改文字
Select all	选中所有对象	Text attributes	改变文字属性
Group	将选中对象编成一组	Object name define	定义对象名字
Ungroup	将一组对象分割为单个对象	Background color	改变用户图形编辑界面的背景颜色
Doplicate	同样复制		
Rotate	90° 旋转	Macro	宏菜单
Mirror	成镜像	Library function	调宏库功能对话框
Overlape	覆盖(将一个物体覆盖在另一个物体上面)	Create Macro	将宏符号添加到宏库中
To graphic pool	进入图形库	Edit	编辑宏符号
To display	进入画面编辑	Parameters	修改宏符号设置参数
Switch contents display/pool	画面区与图形库间的切换	Draw	将宏从宏库中调出到画面编辑区
Insert pool in display	将图形库中的图形引入画面		

操作员站组态

英文菜单	中文说明	英文菜单	中文说明
Animate	动态元素菜单	Options	选项菜单
Reedit	重新编辑	Hardcopy	打印当前的画面区
Bar graph	组态动态棒图形	Snap/Grid	调整背景网格密度
Fill area	组态填充区	Toolbox	控制工具箱的开、关及放置位置
Alphanumeric display	组态数字监视区	Threaded cursor	控制光标的有或无
Graphic symbol	组态图形符号		
Self-animated object	组态自动作对象	Back	返回到项目树
Trend window	组态趋势小视窗		
Selection area	组态选择区	Help	帮助菜单
Button	组态按键		
Button field	组态按键区		
Radio button	组态射频按键		

(二)、绘制静态图形

在用户图形编辑中，所有静态图形绘制中需要的静态图素都需要在菜单项的画图 下面调出而绘制完成的，绘制及编辑方法如下：

1、绘制线条 —— Line

 → Draw → Line

然后用鼠标在绘图区中确定线条的起点(+)后，拖动鼠标左键画出线条，点击鼠标左键结束划线；通过线条属性窗口改变线条的颜色和宽度。

 → Change → Attributes line 弹出线条属性编辑窗口 9—9。

属性包括：

可以从 192 种颜色中选择

线型有实线、虚线、点划线等

可以选择宽度：有 6 种宽度

可以左、右或两头都放置箭头

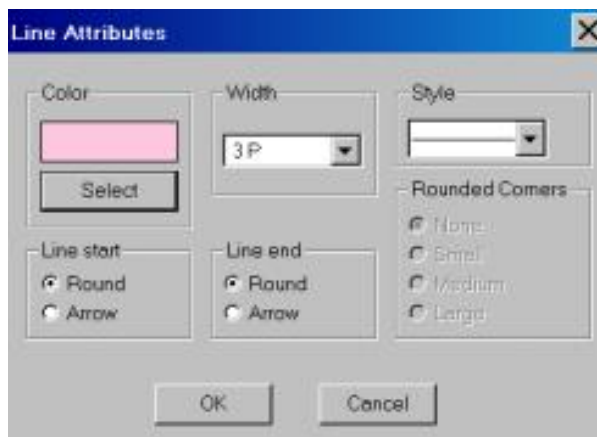


图 9—9 线条属性编辑窗口

2、绘制矩形 —— Rectangle

☞ → Draw → Rectangle

然后用鼠标在绘图区中确定起点(+)后，拖动鼠标左键画出矩形，点击鼠标左键一下结束画图。通过属性编辑窗口改变矩形的属性：如图 9—10 所示，

☞ → Change → Area attributes



图 9—10 矩形属性编辑窗口

其属性包括：

- 边框有 192 种颜色可以选择
- 边框线形有实线、虚线、点划线等
- 线的宽度有 6 种宽度可选择
- 边角可以选择大、小圆弧形或没有圆弧
- 填充区允许前景色、背景色有 192 种颜色可以选择
- 填充区有 15 种类型可以选择

3、绘制多线段 —— Polyline

 → Draw → Polyline

首先用鼠标在绘图区中确定起点，然后拖动鼠标画出一条线段，点击鼠标左键一下作为此线段的结束点和下一条线段的起始点，最后在结束点双击鼠标左键结束多线段绘制。

属性编辑：同 Line

4、绘制多边形 —— Polygon

 → Draw → Polygon

首先用鼠标在绘图区中确定起点，然后拖动鼠标画出一部分区域图形，点击鼠标左键一下作为此区域的结束点和下部分图形的起始点，最后在结束点双击鼠标左键结束多边形的绘制。

属性编辑：同 Rectangle

5、绘制椭圆形 —— Ellipse

 → Draw → Ellipse

用鼠标在绘图区中确定起始点，拖动鼠标左键画出圆或椭圆后点击鼠标左键结束图形绘制。

属性编辑：同 Rectangle

6、写入文字 —— Text

☞ → Draw → Text

随之弹出文字输入对话框，输入需要写入的文字。

通过文字属性编辑窗口如图 9—11 所示，改变文字的属性。

☞ → Change → text attributes

其属性包括：

改变字型大小，共有四种字形可供选择

字体水平或垂直放置

文字区的前景色 (文字颜色)、背景色(文字区的底色)有 192 种颜色可以选择

文字可以是加粗的、斜体的、或下划线的 定位点可以是：起始、

中间、中点、顶部、底部

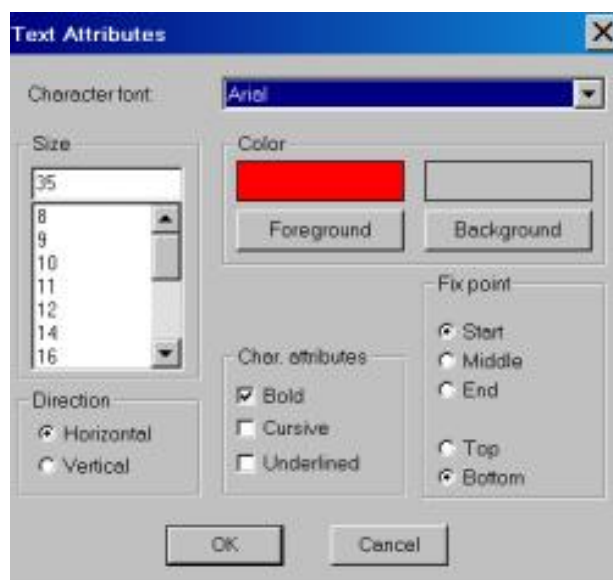


图 9—11 文字属性编辑窗口

操作员站组态

如果需要修改文字的内容，可以调出文字编辑窗口。

☞ → Change → Text edit

7、调入位图 —— Bitmap

☞ → Draw → Bitmap

随之弹出图 9—12 所示的位图引入对话框，从 Freelance \ bitmap 文件夹中或其它路径下找出合适的位图文件。

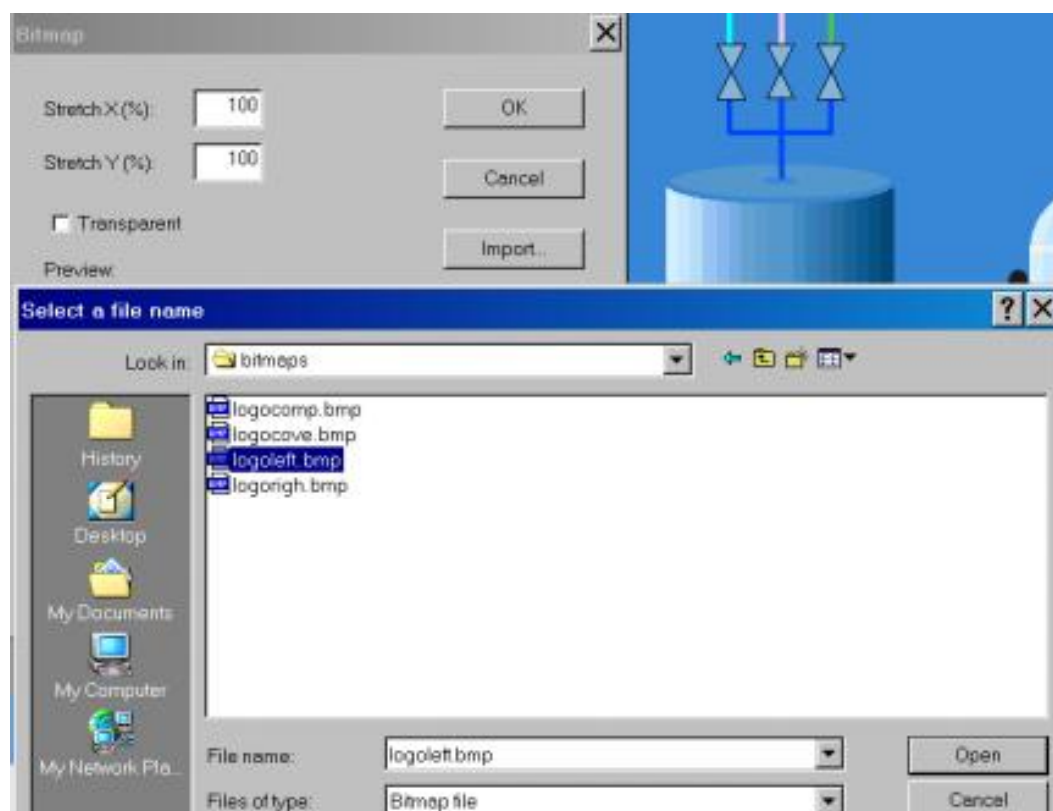


图 9—12 位图引入窗口

、调用宏符号 —— Macro

☰ → Macro → Library function

随之弹出如图 9—13 所示宏符号下装/存储窗口。

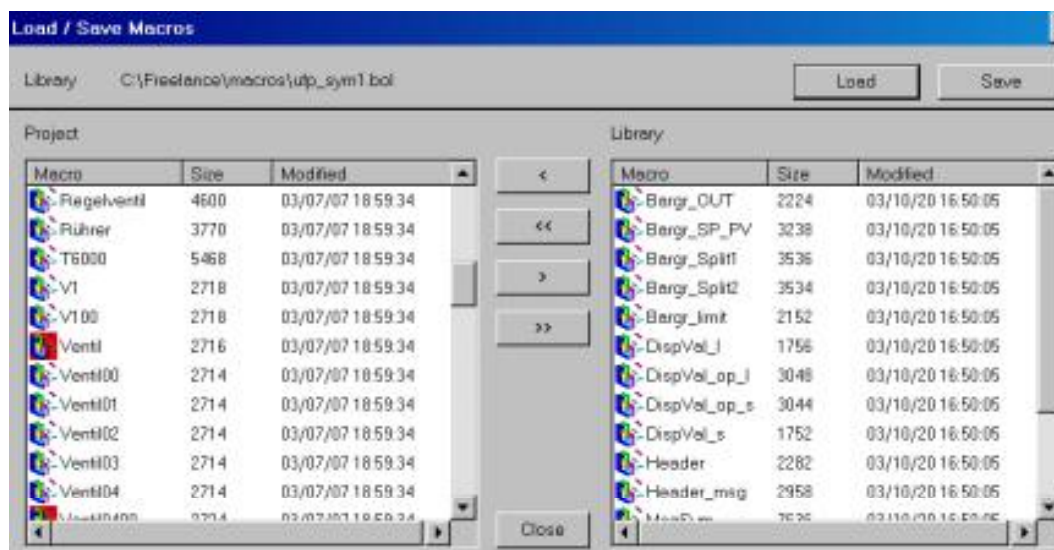


图 9—13 宏符号的下装/存储窗口

Freelance2000 v6.2 中的 Macro(宏符号)由三级构成，即：Macro file、Macro library、Macro pool。只有存放在 Macro Pool 中的宏符号，允许在用户图形组态编辑中使用。所以，用户首先需要把 Macro 文件夹中的 Macro file 下装到 Macro library 中，然后再从 Macro library 中选出所要调用的 Macro 存到 Macro pool 中，只有 Macro pool 中的宏符号才可以被用户最终使用。

* 第一步：下装 Macro file

从图 9—13 中 ☰ → Load 弹出图 9—14 所示窗口

选择 Macro file 文件 → Open，将宏文件下装到 Macro library 中。

操作员站组态

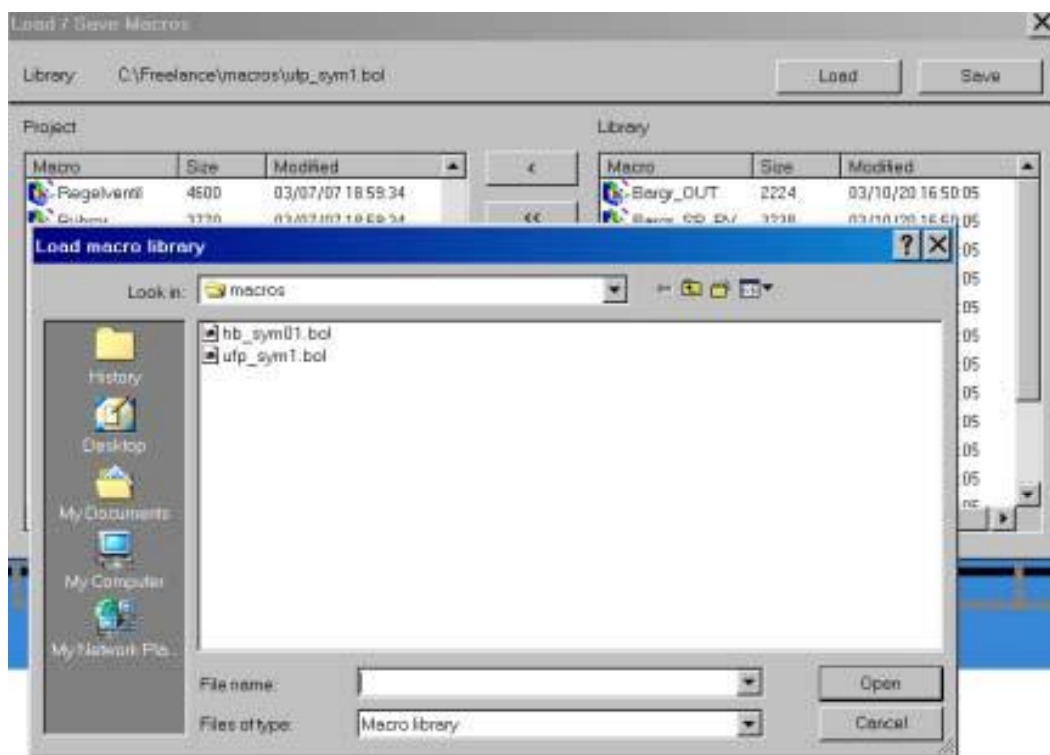


图 9—14 宏文件的下装

* 第二步：从 Macro library 中选择宏符号到 Macro pool 中

从图 9—13 中使用符号 $\leq$ 、 $\ll$ 将右边 Library 中的宏符号下装到项目的 Pool 中。

* 第三步：从宏 菜单下的 Draw 中选择要使用的宏符号调入组态画面。

$\uparrow$ → Macro → Draw

(三)、绘制动态图形

用户图形中的所有动态元素的绘制组态如动态棒、动态数字、按键、填充区等，都在菜单行下的 Animate 中调出组态完成。

1、绘制动态棒—— Bar graph

$\uparrow$ → Animate → Bar graph

首先用鼠标在图形区选中一点，然后按住鼠标左键拖动鼠标画出棒图，棒图画好后释放鼠标，随之弹出图 9—15 所示对话框，按照要求填写各项内容。

1)、Process value 参数表

此表用于设置动态棒跟踪哪个变量，如图 9—15 所示，其中 Display variable 是必填区，应该用 F2 键调出变量置入 Display variable 区域。

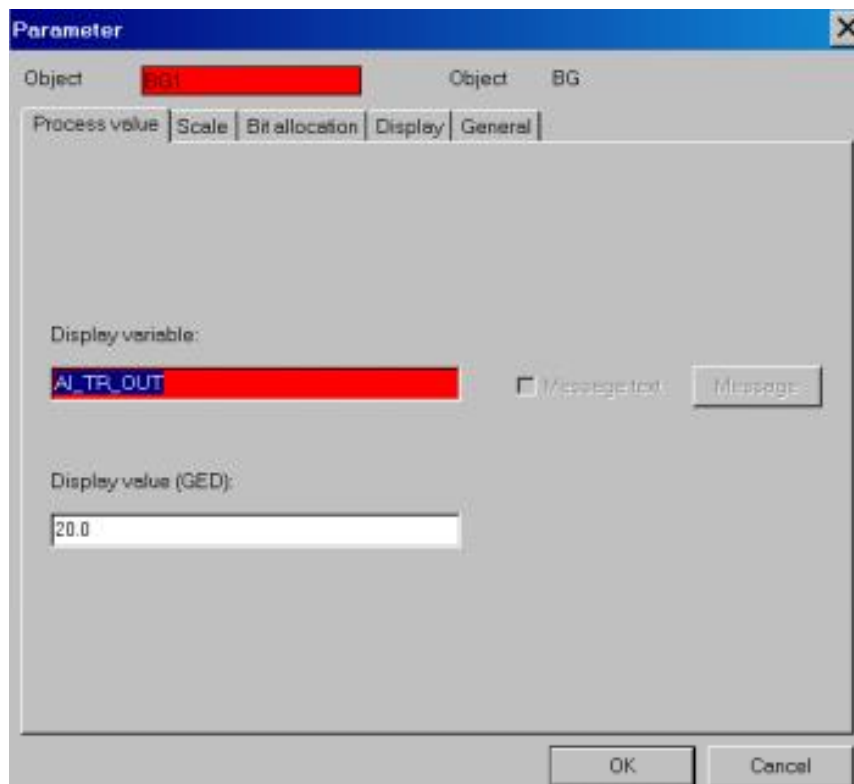


图 9 —15 动态棒组态参数表

2)、Scale 参数表

此表用于设置变量的量程范围及填充方向，如图 9—16 所示。红色区是必填区。

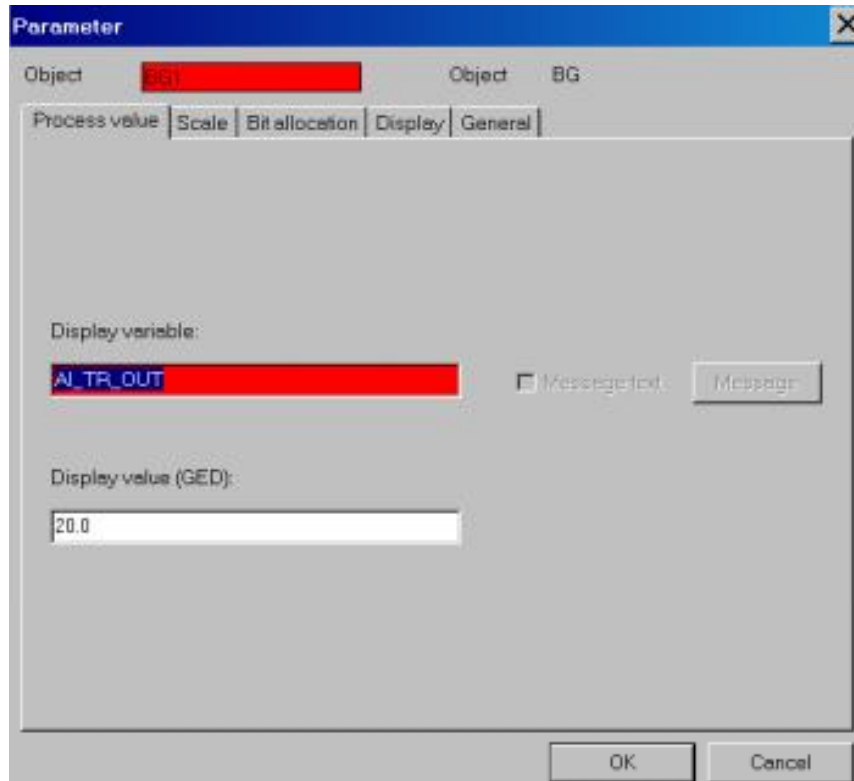


图 9—16 动态棒量程参数表

3)、Bit allocation 表

此表用于定义动态棒的颜色变化种类，应依据变量的报警限设置。用 F2 键调出相关变量置入变量区。

4)、Display 表

此表用于定义动态棒**边框及填充区的颜色**，双击颜色区，可以弹出颜色选项窗口，选择边框及填充区的颜色。

5)、General 表

此表用于定义动态棒在操作员站上的其它连带功能，如双击动态棒调出面板或切换到其它相关画面、或给某一个变量赋值等等。

☚ → Action 弹出图 9—17 所示动作定义选项表。

- * Display allocation: 放置需要切换画面的名称，用 F2 键调出置入。
- * 指定标签: 放置需要调出面板的标签名，用 F2 键调出置入。
- * 写入变量: 为需要赋值的变量名，需要继续组态 Operation 选项中的变量赋值项。

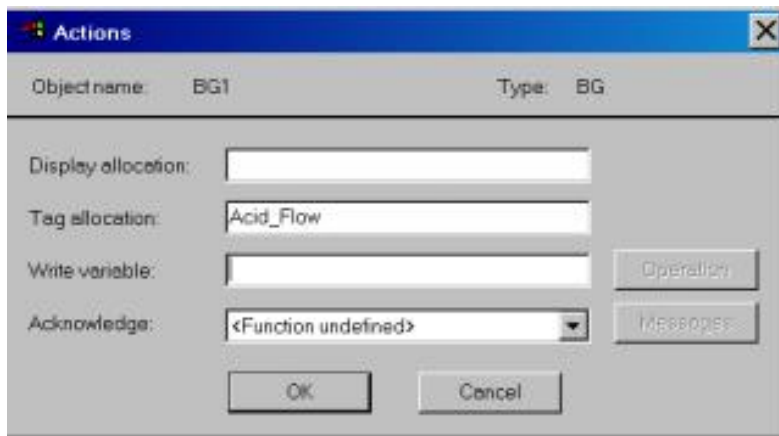


图 9—17 动态棒的动作定义表

2、绘制选择区—— Selection area

☚ → Animate → Selection area

用鼠标在图形区选中一点，按住鼠标左键拖动鼠标画出选择区，随之弹出图 9—18 所示对话框，按照需要填写各项内容。

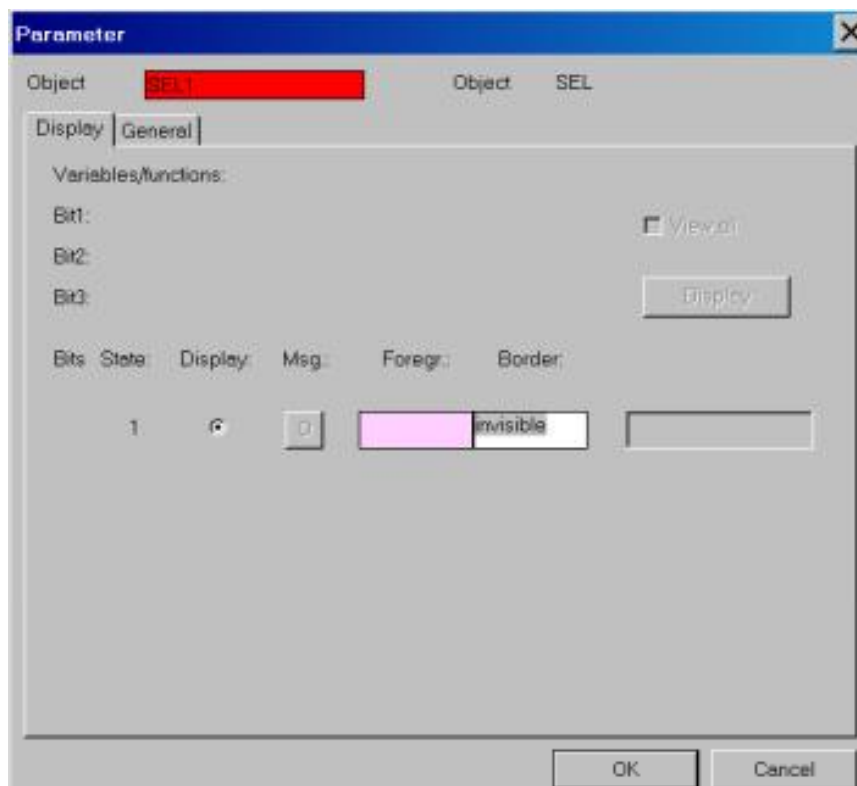


图 9—18 选择区参数组态表

3、绘制填充区——Fill area

☞ → Animate → Fill area

用鼠标在图形区选中一点，然后释放鼠标，随之弹出相应的组态对话框，按照需要(与动态棒的填写类似)填写各项内容。与动态棒组态的区别是：动态棒是先画出图形，后组态参数表；而填充区是先组态参数表，然后通过 **Display** 表中 **Display** 按键返回到绘图区中，使用 **Polygon** 画出多边形

图形后，从菜单 **Edit** 项下 → **Return to object** → **OK**，组态完成。

4、绘制按键、按键簇、射频按键簇——Button ,Button field ,Radio button field

☞ → Animate → Button (或 Button field / Radio button field)

→ 确定按键放置位置，拖动鼠标画出按键或按键簇

→ 组态按钮参数表

操作员站上按钮的作用是：可以调出某一幅画面或面板，为某一个变量赋值，确认某些信息等。

按钮、按钮簇、射频按钮簇 (Button , Button field , Radio button field) 的外观可以选择不同的形状，如：矩形、圆形、3D 等，每个按钮上的文字可以依照不同的需要写入需要的文字、数字或某个符号，颜色可以多种多样，如图 9—19 所示。文字的属性可以通过文字的属性 (Text attributes) 改变其字体及字的大小等。

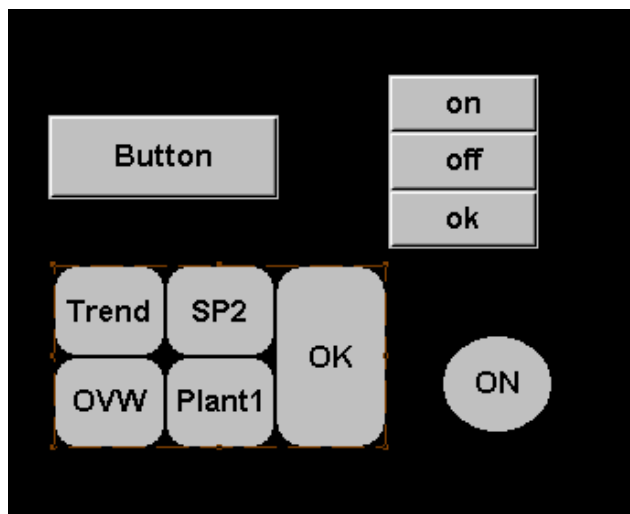


图 9—19 按钮外观图

按钮、按钮簇、射频按钮簇 (Button , Button field , Radio button field) 的组态大致相同，区别仅在于：**按钮 (Button)**

在 DigiVis 中无需对按钮确认，按下按钮即可有效，随之可出现画面、面板或为变量成功赋值。

按钮簇 (Button field)

操作员站组态

按键簇可以由 1 个至 Max.25 个按键和一个确认键组成，按键间是独立组态完成的。与按键的组态一致。在 DigiVis 中，对按键簇的操作包括：按下按键后需点击确认后，才会出现需调出的画面、面板或赋值等。

射频按键簇 (Radio button field)

射频按键簇可以由 1—25 个射频按键组成，它与按键簇的区别在于，射频按键簇中的所有按键只能为同一个过程变量赋值，不能调出画面和面板，也不能为多个变量赋值。射频按键簇中没有确认按键。

组态按键参数表

双击按键，可以调出如图 9—20 所示按键参数表，按照用户要求填写按键名称，选择按键外形，组态按键的动作。

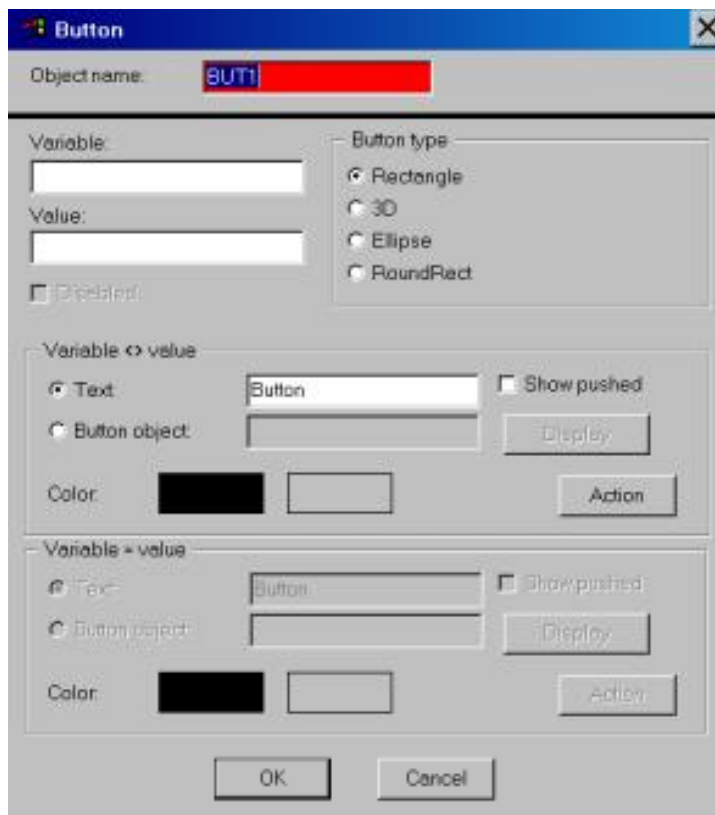


图 9—20 按键参数组态表

其中：

Vaible	用于定义按键以什么方式、文字、颜色显示等
Value	与变量一同决定按键的显示方式、文字、颜色等
Button field	用于定义按键的外形，外观形状包括：矩形、3D、圆形和带圆弧角的矩形。 ☉ Rectangle 矩形 ☉ 3D 具有立体感 ☉ Ellipse 圆形或椭圆形 ☉ RoundRect 带圆弧角的矩形
Text	用于定义按键上显示的文字
Button ob ect	用于定义按键以什么画面取代文字显示，画面由 Display 返回用户画面编辑区定义
Color	用于定义按键上文字显示颜色及底色
Show pushed	☑ 按键以按下方式显示 ☐ 按键以凸出方式显示
Action	用于定义按键的动作，随即弹出动作组态表，如图 9-21 所示。

操作员站组态

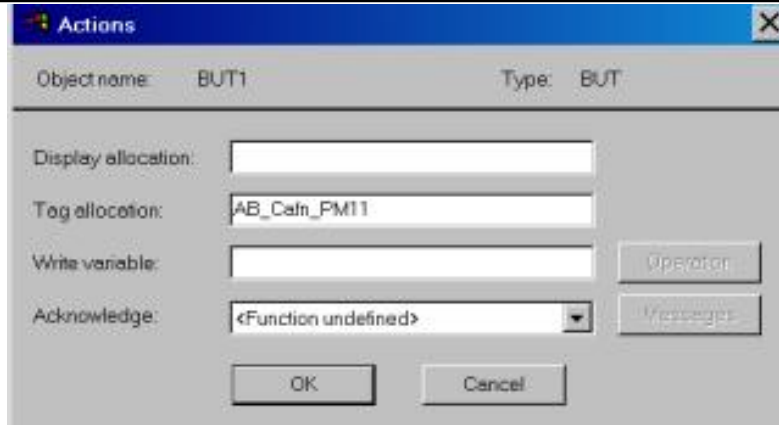


图 9-21 动作组态表

- | | |
|--------------------|---|
| Display allocation | 填写需调出画面的名称，用 F2 键调出画面选项对话框，将需要的画面名选中后 OK |
| Tag allocation | 填写需调出面板的名称，用 F2 键调出标签选项对话框，将需要的标签名选中后 OK |
| Write variable | 填写需赋值变量的名称，用 F2 键调出变量表对话框，将需要赋值的变量名选中后 OK，在按下 Operation 键后赋值表中写入变量值 |

5、Trend window —— 趋势视窗

- Animate → Trend window
- 在用户画面编辑区中画出趋势窗口的大小
- 组态趋势参数表

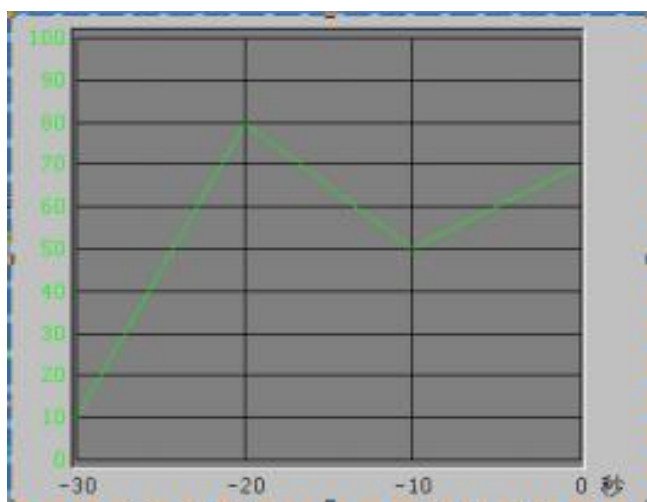


图 9—22 趋势视窗图

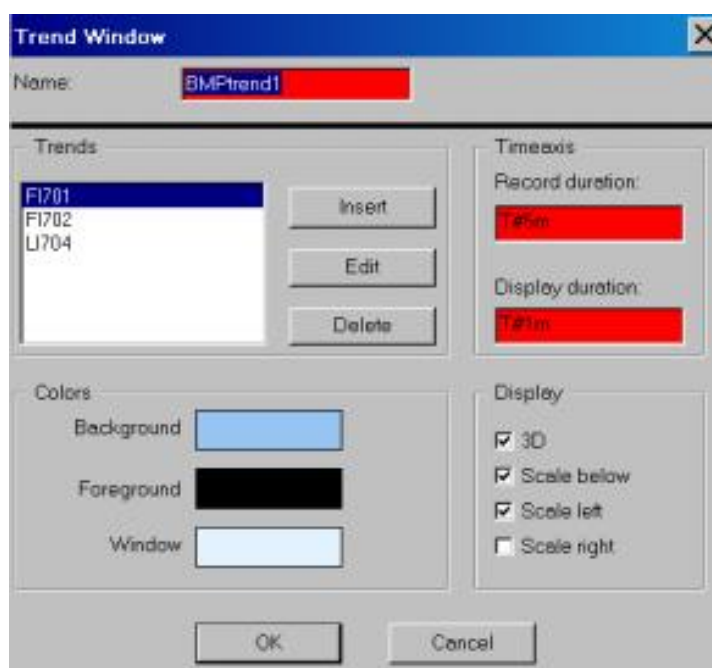


图 9—23 趋势视窗参数组态表

其中：

操作员站组态

Trends	其条目下列出的变量是用户画面趋势视窗中监视的变量
Insert	按下 Insert 键后，随即出现趋势变量编辑表，如图 9—24 所示。
Edit	编辑已存在的趋势变量。
Delete	将条目下列出的某个选中变量删除。
Display	定义用户图形画面中趋势视窗的表现方式

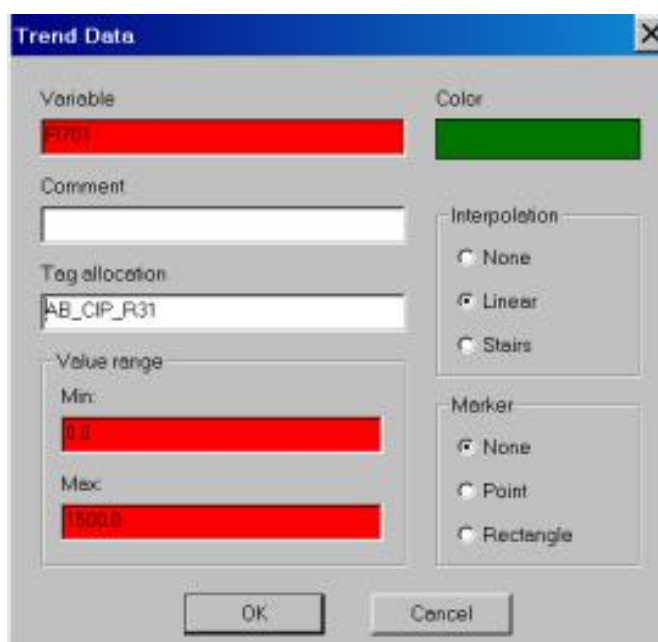


图 9—24 趋势变量编辑表

练习

1、趋势画面组态练习

1)、参照图 10.1 在 FBD_4 中组态一个趋势功能块。

2)、参照图 10.2 在 FBD_8 中组态一个趋势功能块。

3)、在操作员站(D-OS)上组态两幅趋势画面。

为什么需要先在过程站组态趋势采集功能块，操作员站上的趋势画面与它有什么关系？

2、使用标签表中的标签，在操作员站上组态两幅组画面。

组画面的作用是什么？

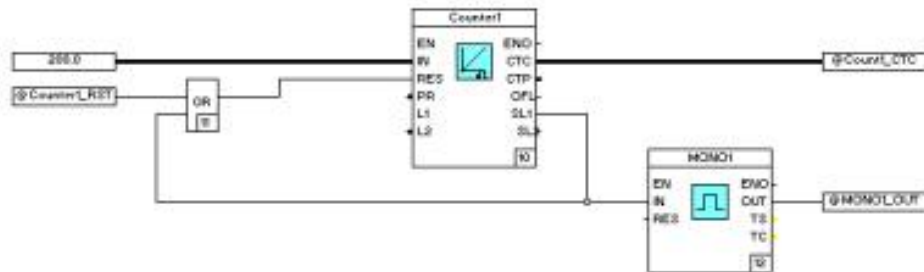
3、在操作员站上组态一幅总貌画面。

每个操作员站上的总貌画面是否有数量限制？多少个？

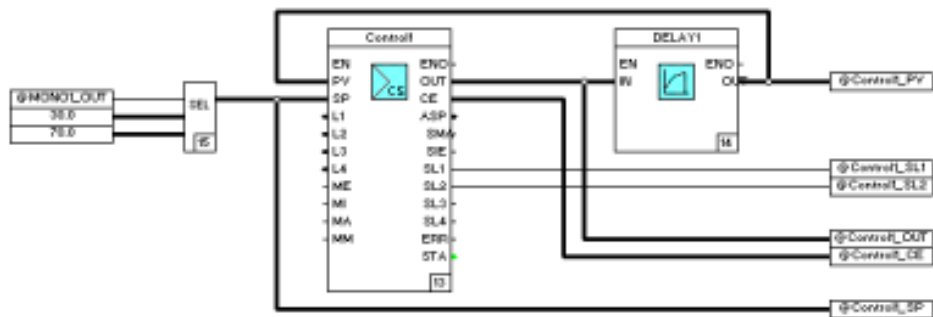
4、在操作员站上组态一幅 SFC 画面。

5、参照图 14.1、14.2、14.3 在操作员站上组态三幅用户图形画面。

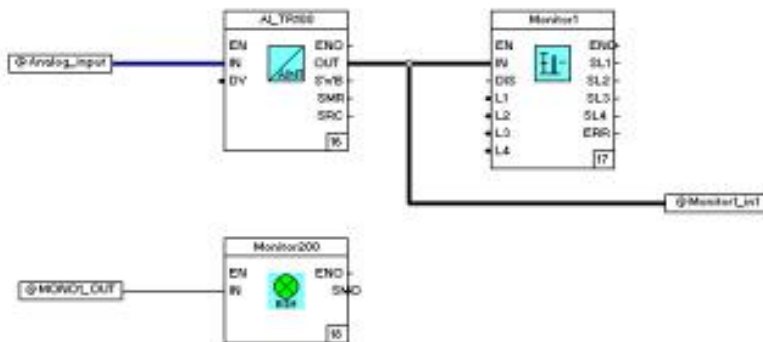
练习 2.1 Counter with analog input, Monoflop



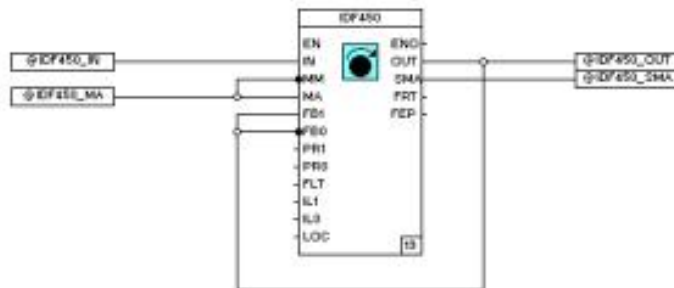
练习 3.1 Continuous standard controller



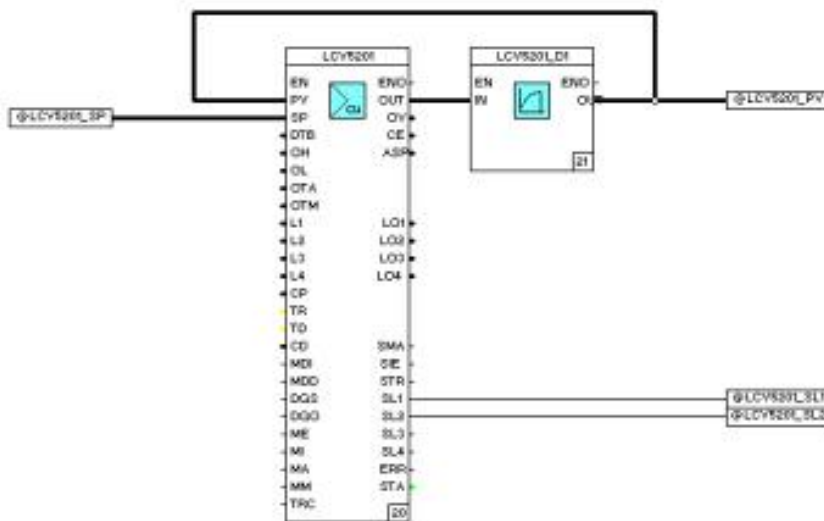
练习 4.1 Analog and digital monitoring

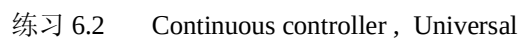


练习 5.1 Open_loop control , IDF_1

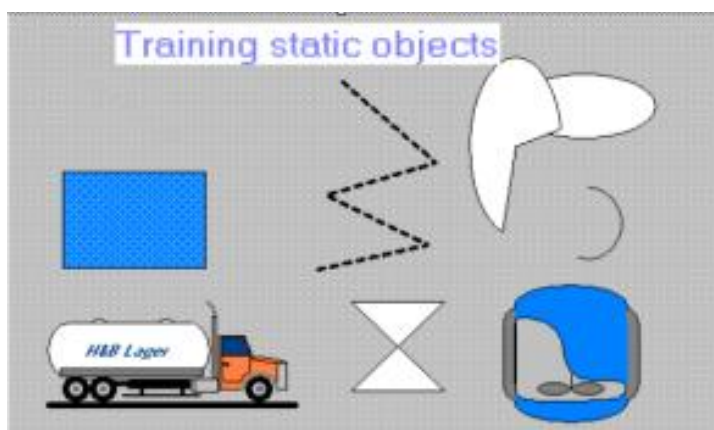


练习 6.1 Continuous controller , Universal





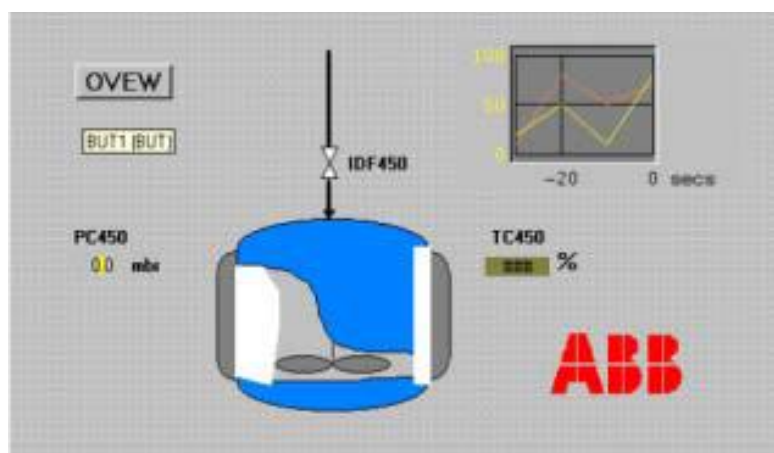
练习 14.1 绘制静态图形元素



练习 14.2 绘制动态元素



练习 14.3 绘制用户画面



附录一 功能和功能块

功能和功能块

现场控制器是集散控制系统的核心部件，由它完成现场设备的闭环控制、顺序逻辑控制和数据采集、计算等任务。现场控制器的控制与计算功能如何实现，用户使用什么手段完成控制要求就显得十分重要。为了集散系统的推广，各个 DCS 厂家都开发了面向用户、面向控制的语言，这些语言往往不需要用户具有计算机软件编程知识，只要用户具有自动化过程控制方面的知识，经过 DCS 厂家的短期培训，就能较熟练的使用和掌握。目前 DCS 系统普遍使用的语言是功能块语言。

功能块语言是把相当于由模拟仪表实现的功能，如：PID 控制、累加器等，编制成一个个子程序，预先固化在存储器中，功能块的整体我们称为功能块库。功能块的控制作用比仪表更强，使用更加方便；用户只需要参照 DCS 制造厂家提供的组态手册，调出需要的功能块，填写相应的参数，并用软线连接起来的方式，我们称为组态。

功能块语言处理子程序的方法有两种类型：一种是把子程序尽量划小，如：加、减、乘、与、或等，我们称为小功能块语言；另一种是把功能块所能完成的功能尽量扩大，如 PID 控制器功能块，它不仅能够完成 PID 运算，还包括报警等功能，我们称为大功能块语言。

Control Builder F 为用户提供的组态语言是功能块语言，在它的功能块库中存有 14 大类 200 多种功能和功能块，这些功能和功能块已能满足如：逻辑运算、开环与闭环回路控制、监视、记录、趋势等功能需求；同时用户还可以组态自定义功能块，满足用户特定的需求。

一、概念介绍

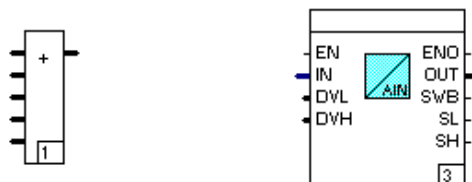
1、功能

在 Control Builder F 中，我们称小功能块语言为功能。功能的作用单一，数据类型单一，不需要填写标签名和块参数。如加、减、乘、与、或、……等。

2、功能块

是指可以完成各种复杂运算、控制、监视、记录、趋势等功能要求的块，允许多种数据类型存在同一个块中，需要填写块参数和标签名。

3、功能与功能块外观说明



功能和功能块

从外观上看，功能与功能块都包括输入端脚与输出端脚，较长的输入与输出端脚我们称为“指令”端脚，较短的输入与输出端脚我们称为“随意”端脚；“指令”端脚是必须

通过软信号连线将变量与其连接的端脚，否则功能块不能正常工作；而“随意”端脚连线与否不会影响功能块的正常工作。

功能与功能块右下方的数字用以指示它在一个组态图中的处理顺序。

4、功能块参数定义

每一个功能块都需要定义它的块参数。在功能块参数表中，红色的区域我们称为“指令”区域，即必须填写文字或数字的区域。白色区域，我们称为“随意”区域，即可以填写或不填写文字与数字，或保持默认值均可的区域。一般看来，功能块的名字、量程的起点与终点为“指令”区域。

块参数包括：

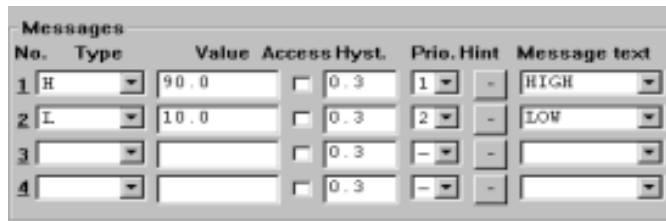
名字(Name)	在同一个项目树组态中，每一个功能块的名字必须是唯一的，这个名字我们称为标签，它会自动添加到标签表中。 允许以任意字母开头，最多允许写入 12 个字符。
短文本注释(Short text)	最多允许写入 12 个字符，没有字符限制。
长文本注释(Long text)	最多允许写入 30 个字符，没有字符限制。
处理顺序(Sequence)	功能块的处理顺序 (1—99) 允许在这里改变，一个功能块顺序的改变，会导致一个组态图中所有功能块顺序的改变
报警信息(Message)	每个块参数中最多允许定义四个报警极限值，包括报警类型(type)、极限值(value)、优先级(prio)、文本说明(message text)等。 每个报警极限值对应一个数字量输出，当违反报警极限值后相应的数字量输出为逻辑 1，反之输出逻辑 0。
Type	定义报警信息类型，如：HH、H、L、LL 等。
Value	定义报警极限值(应在量程范围内)。
Access	✓ 允许操作员在面板上修改报警极限值。
Hyst	每个极限值最大允许有 30% 的滞后。
Prio	定义报警信息优先级（1~4）。

Hint

组态报警提示信息。

Mess text

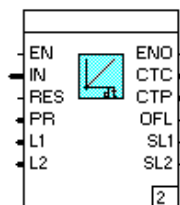
报警信息文本说明，可以使用键盘写入，也可从下拉表中选择填入。



二、功能块介绍

(一). Counter with Analog Input, CT_ANA

· 图标



1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
IN	输入信号端	CTC	输出值
RES	复位信号端	CTP	上一次的输出值
PR	基础值	OFL	进位信号
L1	报警极限值 1	SL1	违反极限值 1 时，输出 Logic-1
L2	报警极限值 2	SL2	违反极限值 2 时，输出 Logic-1

功能和功能块

2、用途

此功能块将模拟量输入信号值经线性累加后输出。

公式：

$$CTC = CTC_{n-1} + IN \times \frac{TZ}{ZB}$$

其中：

CTC: 计数器的输出值

CTC_{n-1}: 计数器前一次的输出值

IN: 计数器的输入值

TZ: 任务循环时间

ZB: 时间基数

☞ 此功能块必须运行在 **Interval tasks** 的 **Equidistant** 方式下。

3、面板

显示面板包括：功能块名字（Tag Name）、短文本说明、量称的起点与终点、工程单位、当前输出值 CA（CTC 的输出）、上一次的输出值 CP（CTP 的输出值）、基础值 BV（PR 的输入）、极限值 L1、L2。

4、面板操作

可以改变基础值和极限值 L1、L2。如果 **Reset** 有效，则允许操作面板上的 **RESET** 键，计数器复位到基础值后重新开始累加计数。

5、参数表

Scale start:	实际的量程起点，数据类型为 Real 。
Scale end:	实际的量程终点，数据类型为 Real 。
Dimension:	工程单位（可以直接用键盘写入，也可以从下拉表中选择后填入）
Basic value:	基础值即计数器开始计数的基值（基础值应在量程范围内），如果不填，表示从 0 开始计数。

Parameters: Counter with analog input CT_ANA

General data

Name: [redacted] Short text: [] Processing: ☒ Sequence: 3

Long text: []

Scale start: 0.0 Scale end: 1000.0 Dimension: []

Basic value: [] Resetting val.: []

Time base:
☐ second
☐ minute
☒ hour
☐ day

Period duration: []

Cycle count for QFL: 2

Access

☒ Basic value ☐ Reset

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1	[]	[]	☐	3.0	-	-	[]

OK Cancel Save Reset Check Help

复位计数器:

Resetting value

当计数器输出超过此处写入的复位值后，计数器会自动复位到基础值，然后再重新计数开始。

如果此区域不填，复位值没有响应，在 commissioning 状态下，此区域显示 0.0，并且不允许改变。

Period duration

复位时间，即在某一固定时间间隔后，计数器会自动复位；以时间格式写入，例如：T# 2h。

如果此区域不填，复位时间作用无效，在 commissioning 下，此区域值不允许修改。

功能和功能块

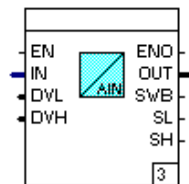
Time base: 输入值的运算时间单位。

Cycle count for OFL 每复位一次，OFL 增加 1，表示循环又开始一次。OFL 为循环次数（有效值为 1~99）。

Access

Basic value ✓ 允许在面板上修改基础值。
 L 不允许在面板上修改基础值。

Reset ✓ 允许操作员使用面板进行复位操作。
 I 不允许操作员使用面板进行复位操作。

(二). Analog Input Transformation, AI_TR

· 图标

1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
IN	输入信号端	SWB	与现场接线断开，端脚输出逻辑 1
DVL	输入信号低于下限量程范围时的输入值	SL	输入信号低于下限量程范围，端脚输出逻辑 1
DVH	输入信号高于上限量程范围时的输入值	SH	输入信号高于上限量程范围，端脚输出逻辑 1

2、用途

用于把来自数据类型为 **UNIT** 的模拟量输入模件的输入信号，转换成数据类型为 **REAL** 类型的输入值；如果模拟量输入信号低于或高于信号范围限制，由输出端脚 **SL** 或 **SH** 输出 **logic-1** 表示超限报警。

3、面板

OUT 是输出信号值，即功能块端脚 **OUT** 的输出值，**DL** 是功能块端脚 **DVL** 的值，**DH** 是功能块端脚 **DVH** 的值。

4、面板操作：无

5、参数表

Parameters: Analog input transformation AI_TR				
General data				
Name:	[Redacted]	Short text:		Processing: ☒
Long text:			Sequence:	1
Measuring range start:	0.0	Tolerance range:	1.0	
Measuring range end:	100.0			
Input range				
☐ 0...20 mA ☒ 4...20 mA				
☐ Hold last value				
Messages				
	Default value	Prio.	Hint	Message text
Too low:		-	-	
Too high:		-	-	
Wire break:		S3		

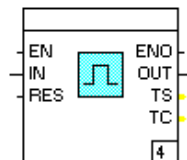
Tolerance rang : 偏离标准范围的最大容许限度，即允许超出测量范围 0...20mA 或 4 ... 20 mA 的百分比范围。
有效值为 0.0%~3.0%，以 **REAL** 数据类型写入，
即 3.0 = 3.0%

功能和功能块

Input range:	输入信号范围，有 0~20 mA 或 4~20 mA 两种选项
Hold last value:	I 故障时以组态值为有效值，即由 Default value 的定义值 ✓ 故障时保持最后值为有效值。
Default value:	定义故障时的有效值。
Too low:	写入值为输入信号低于 0/4 mA 对应的信号范围起点的有效值，以 REAL 数据类型写入。
Too High:	写入值为输入信号高于 20 mA 对应的信号范围终点的有效值，以 REAL 数据类型写入。
Wire break:	断线监视功能，不需要组态，仅在输入信号为 4~20 mA 时有效；如果现场设备与模件的连线出现断路(采集进来的信号 <3.2mA，输出的信号 >3.8mA)，此功能块将产生一个线路断路信号，即在输出端 SWB 产生 logic1 的同时，在 DigiVis 上
生	成一个线路断线的系统报警信息。

(三). Monoflop , MONO_F

· 图标



1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
IN	输入信号端	TS	脉冲间隔
RES	复位信号端	TC	过去的时间
		OUT	输出值

2、用途

产生脉冲输出的功能块，允许组态脉冲时间；允许用 **RESET** 复位脉冲输出信号，即将输出值为 **logic-1** 复位为 **logic-0** 输出，输入信号既可以选正脉冲触发也可以选负脉冲触发。

3、面板

标签名，**Short text**，脉冲时间 **DT**（对应功能块的输出端脚 **TS**），过去的时间 **TC**（对应功能块的输出端脚 **TC**），动态棒图，复位键 **RESET**。

4、面板操作

- 在定义的范围内改变脉冲时间
- 控制复位键 **Reset**

5、参数表

功能和功能块

Parameters: Monoflop MONO_F

General data

Name: [REDACTED] Short text: Processing ☒

Longtext: Sequence:

Pulse

Pulse duration: ☐ Access

Scale end:

Trigger

☒ trigger on + edge

☐ trigger on - edge

Reset

☐ Access

Message

Mess.priority: Hint: Mess.text:

Buttons: OK, Cancel, Save, Reset, Check, Help, <<, >>

Pulse

Pulse duration: 脉冲时间；以时间格式写入，如 T#5m3s。

Access:



允许操作员在面板上改变脉冲时间。



不允许操作员在面板上改变脉冲时间。

Scale end:

画面显示时某一段范围的结束点，以时间格式写入。

Trigger

Trigger on +edge: ☒ 正边缘触发

Trigger on -edge : ☐ 负边缘触发

Reset

Access :



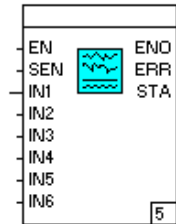
允许操作员在面板上控制 Reset 键



不允许操作员在面板上控制 Reset 键

(四). Trend Acquisition, TREND—趋势归档功能块

· 图标:



1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
SEN	当输入为逻辑 1 时，归档方式为“Ready”方式	ERR	输出逻辑 1 时，表示功能块工作在错误方式
IN 1—6	输入信号端	STA	出错码

2、用途

每个操作员站上最多允许组态 42 幅趋势画面，每幅趋势画面中最多允许组态 6 个变量的趋势曲线，那么每个操作员站上最多允许组态 252 条趋势曲线。

此功能块用于操作员站趋势画面中变量数据的采集，为降低操作员站的负荷，在操作员站上的每一幅趋势画面，应该对应过程站中的一个趋势采集功能块组态。

所有数据类型的变量值均可以直接作为趋势归档功能块的输入值。

对归档功能块中的每一个输入信号来说，它最近的 200 个带有时间标记的值会存储在缓冲器中(即 200×6 输入信号 + 200×1 时间标记 + 200 状态)，当归档有效时，如果缓冲器中的值超出其安全阈值时，所有的采集值将会被传输到长期归档中去；我们可以使用 DigiBrowsers 软件查看长期归档文件。

如果一幅趋势画面的归档方式设置为“ready”方式，那么 DigiVis 中趋势归档的起

功能和功能块

动、停止取决于归档功能块的输入端 SEN 的输入值，DigiVis 的归档设置将不起作用。

趋势采集功能块为 DigiVis 上的短期归档趋势画面提供数据，当 DigiVis 启动后，所有归档变量的采集值从相应的趋势归档功能块中读出，并存放在短期归档数据库中 (即保存在 DigiVis—PC 机的硬盘上)，一秒钟可以归档最近的 30 个记录，当退出 DigiVis 时，趋势画面的短期归档数据不做保存。

DigiVis—PC 机在短期归档时所需占用计算机硬盘的空间由工程师站自动计算出来：

计算公式如下

$$SHD = SR \cdot \frac{Ad}{Tz \cdot IU}$$

SHD: DigiVis—PC 短期归档时所需占用的硬盘空间

SR: 采集变量的字节数 (尺寸)，即 Record size

Ad: 归档间隔

Tz: 项目树中趋势归档功能块所在任务的循环时间

IU: 扫描系数 (Scan—down factor)

3、趋势画面

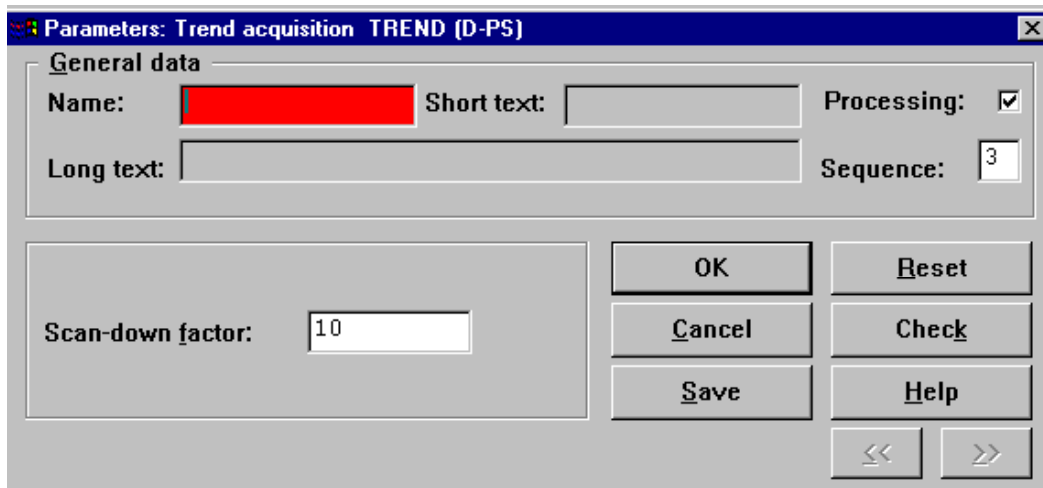
每一个趋势采集功能块对应操作员站上的一幅趋势画面，趋势画面包括数值区、图形区与操作区。

4、趋势画面操作控制

包括：

- 调历史趋势
- 某一个变量趋势曲线的隐藏与显现
- 修改变量的量程范围
- 改变变量的显示颜色与线条形状
- 改变趋势画面的 X 轴的范围
- 调某一个变量的控制面板

5、参数表



Scan-down factor: 扫描系数，有效值为 1～99，用于归档空间的计算。

(五). Continuous Controller

用于 PID 控制运算的连续控制器功能块，能产生一个跟踪设定值的连续输出变量 OUT 控制现场的调节阀。

Freelance 2000 控制系统包括三种类型的 PID 调节功能块，即：C_CS 控制器；C_CU 控制器；C_CR 控制器。

其中：

C_CS 控制器可以完成一个标准的 PID 调节运算，但不允许通过外部输入改变其控制状态。

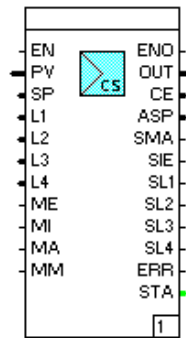
C_CU 控制器可以完成一个复杂的 PID 调节运算，并允许通过外部输入值改变其控制器状态。

C_CR 是比率控制器，比率值允许外设定或内设定。

以 C_CS 为例介绍连续控制器功能块

功能和功能块

· 图标



1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
PV	过程值	OUT	输出值
SP	设定值	CE	偏差值
PR	基础值	ASP	设定值是否为有效状态信号
L1	报警极限值 1	SL1	违反极限值 1 时，输出 Logic-1
L2	报警极限值 2	SL2	违反极限值 2 时，输出 Logic-1
L3	报警极限值 3	SL3	违反极限值 3 时，输出 Logic-1
L4	报警极限值 4	SL4	违反极限值 4 时，输出 Logic-1
ME	此端脚为逻辑 1 时，功能块工作方式为外设定方式	SIE	输出逻辑 0，为内设定控制方式 输出逻辑 1，为外设定控制方式
MI	此端脚为逻辑 1 时，功能块工作方式为内设定方式	SMA	输出逻辑 0，为手动控制方式 输出逻辑 1，为自动控制方式

MM	此端脚为逻辑 1 时，功能块工作方式手动方式	ERR	输出逻辑 1，表示功能块工作在出错方式。
MA	此端脚为逻辑 1 时，功能块工作方式自动方式	STA	出错码

2、用途

所有的控制器都包括一个输入变量 **PV**，**PV** 作为被控制变量，在任何时候都应该尽可能的与设定值 **SP** 接近，我们可以通过调节控制偏差 **CE** ($CE = SP - PV$) 来实现，即控制器通过输出变量 **OUT** 控制设备，尽可能的消除偏差 **CE**。

设定值 **SP** 允许内给定或外给定。可以由功能块参数设置，也可以由功能块的输入端脚 **MI** (设定方式—内部)与 **ME** (设定方式—外部)决定，同时由功能块的输出端脚 **SIE** 响应现在的设定值给定方式。如果 **MI** 与 **ME** 同时为逻辑—1，内部设定优先级高于外部设定优先级，功能块工作于内设定方式。

功能块的工作方式允许为手动或自动工作方式 (**MM/MA**)，可以由功能块参数设置，也可以由功能块的输入端脚 **MM**(工作方式—手动)与 **MA**(工作方式—自动)决定，同时由输出端脚 **SMA** 显示功能块当前的工作方式。如果 **MM** 与 **MA** 同时为逻辑—1，手动优先级高于自动，功能块工作于手动方式。

由下表给出工作方式与设定方式的响应关系表

In put		Output
MM 手动	MA 自动	SMA 手动/自动
0	1	1
1	1	0
1	0	0

In put		Output
MI 内设定	ME 外设定	SIE 内设定/外设定
0	1	1
1	1	0
1	0	0

功能块最多允许设置四个报警极限，极限值可以由功能块参数表设置，也可以由功能块输入端脚 **L1—L4** 接入。当输出值超出设置的报警极限值后，由 **SL1—SL4** 输出逻辑 1 响应。

当功能块出错后，输出端脚 **ERR** 输出逻辑 1，同时由输出端脚 **STA** 给出出错码。

功能和功能块

STA	Meaning
0	没有错误
1	输入错误(输入信号超出信号范围)
3	内部出错(范围 $range \leq 0$)
9	小数点位数过多，执行校正
10	小数点位数过多，校正失败

3、面板

标签名, **Short text**, 量程范围和工程单位, 设定值(对应于功能块的端脚 **SP**)、过程值 **PV** (对应于功能块的端脚 **PV**)、输出值 **OUT** (对应于功能块的端脚 **OUT**) 的数字值指示和棒图显示, 设定值与过程值的棒图显示位于左上方; 输出棒图显示位于右上方。L1 至 L4 为报警极限指示值。

4、面板控制

- 允许改变控制方式: 手动/自动间转换。
- 允许在手动方式改变输出值。
- 使用 INT/EXT 按键, 改变设定值 **SP** 的设定方式。
- 修改 L1—L4 的报警极限值。
- 确认报警信息

5、参数表

C_CS 功能块包括三张组态参数表。

表 1/3:

Parameters: Continuous controller, standard C_CS (1/3)

General data

Name: Short text: Processing: ☒

Long text: Sequence:

Range start: Range end: Dimension:

Internal set point: ☒ Access

Output: ☒ Access

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1			☐	0.3	-	-	
2			☐	0.3	-	-	
3			☐	0.3	-	-	
4			☐	0.3	-	-	

OK Cancel Save Reset Check Help << >>

Range start: 量程的起点（REAL 格式）

Range end: 量程的终点（REAL 格式）

Dimension: 工程单位

Internal set point: 内部设定值

Access: ☒ 允许操作员在操作员站上改变内部设定值
☐ 不允许操作员在操作员站上改变内部设定值

Output: 输出值

Access: ☒ 允许手动方式时在操作员站上改变输出值
☐ 不允许手动方式时在操作员站上改变输出值

表 2/3:

Parameters: Continuous controller, standard C_CS (2/3)

PID parameters

☒ P branch D-action D - action of CP: 0.5

☒ I branch ☐ on ☐ PV ☒ CE TR: T#15s

☒ Inverse char. ☐ positive P - action of TD: T#0s

☐ negative ☐ PV ☒ CE CD: 1.0

☒ off Operating point: 0.0 Tsync: T#30s

Operating mode inhibition

☐ SP-Intern ☐ Out-Man

☐ SP-Extern ☐ Out-Auto

Sync. Man to Auto

☒ without

☐ adj. to old set point

☐ adj. to current set point

Operating modes

☒ Intern ☐ Extern

☒ Man ☐ Auto

OK

Cancel

Save

Reset

Check

Help

<< >>

PID parameters (PID 参数)

- ✓ P branch 比例运算有效
- ✓ I branch 积分运算有效

D - action

- on 微分运算有效
- positive 正微分运算有效
- negative 反微分运算有效
- off 没有微分作用

D - action of

- PV 微分作用于过程值
- CE 微分作用于偏差值

P—action of

PV	比例作用于过程值
CE	比例作用于偏差值

Derivative action

微分运算 = $CD \times D$ ，其中 CD 为微分作用系数，
范围为 $0.0 \leq CD \leq 20.0$ ，以 **Real** 类型值写入

Controller Parameters

CP:	比例修正值，以 Real 数据类型写入， 范围 $0.0 \leq CP \leq 1000.0$
TR:	积分稳定作用时间 TR ，以时间格式写入， 范围 $TR \geq 0ms$
TD:	比率时间（ Rate time ），以时间格式写入， 范围 $TD \geq 0ms$
Tsync: PT1 动作指的是	PT1 动作的稳定时间，时间类型， $Tsync \geq 0ms$ — 无积分运算时从手动切换到自动 — 无积分运算时比例系数 KP 平滑变化 — 依据组态要求，PI 到 P 的转换
Inverse char	✓ 作用于反偏差 CE
Operating point	仅在没有积分运算时需要组态 静态偏差 CE 不为 0 时的输出值的百分数，以 Real 格式写入。
Sync. MAN to AUTO	从手动切换到自动时，为了避免对设定值 SP 产生冲突，有以下三种选择：
Without :	设定值 SP 保持不变
adjust to old set point:	在手动方式时设定值 SP 跟踪 PV 值；在自动方式时设定值按照某一个预定义的斜度变为原 SP 值。

功能和功能块

- **adjust to current set point:** 在手动方式时设定值 SP 跟踪 PV 值；在自动方式时 SP 保持当前的内部设定值。

Operating mode inhibition

SP—Intern	禁止内给定设定方式
SP—Extern	禁止外给定设定方式
Out—Man	禁止手动工作方式
Out—Auto	禁止自动工作方式

Operation Mode

Intern	控制器起动时工作于内设定方式
Extern	控制器起动时工作于外设定方式
Man	控制器起动时工作于手动工作方式
Auto	控制器起动时工作于自动工作方式

表 3/3:

Parameters: Continuous controller, standard C_CS (3/3)

Gradient function (time per range)

Set point rising: T#0s
falling: T#0s

☒ Setpoint gradient active in manual

Output rising: T#0s
falling: T#0s

☒ Output gradient active in manual

Constants

Set point limit high: 100.0
Set point limit low: 0.0
Output limit high: 100.0
Output limit low: 0.0
Dead band for c.-error: 0.0

Effectiveness of output limits

☐ active in Man and Auto
☒ active in Auto
☐ not active

Set point tracking

☒ SPI tracks actual set point
☐ SPI as adjusted

OK Reset
Cancel Check
Save Help
≤≤ ≥≥

Gradient function (time per span)

Set point rising 面板上设定值的指示棒从某一低点到某一高点的上升时间，以时间类型写入

Set point Falling 面板上设定值的指示棒从某一高点到某一低点的下降时间，以时间类型写入

set point gradient active in manual

✓ 面板上设定值指示棒的上升时间与下降时间在手动自动工作方式都有效。

L 面板上设定值指示棒的上升时间与下降时间在手动工作方式时有效。

Output rising

面板上输出值的指示棒图从某一低点到某一高点的上升时间，以时间类型写入

功能和功能块

Output Falling

面板上输出值的指示棒图从某一高点到某一低点的下降时间，以时间类型写入

output gradient active in manual

- ✓ 面板上输出值的上升时间与下降时间设置在手动、自动工作方式都有效。
- ! 面板上输出值的上升时间与下降时间设置在手动工作方式时有效。

Constants:

Set point limit high

设定值 SP 的高限值，以 Real 类型写入

Set point limit low

设定值 SP 的低限值，以 Real 类型写入

Output limit high (OH)

输出值 OUT 的高限值，以 Real 类型写入

Output limit low (OL)

输出值 OUT 的低限值，以 Real 类型写入

Dead band for CE

CE 的不工作区域，即偏差 CE 的死区时间，以 REAL 类型写入。

Effectiveness of output limits

Active in Man and Auto

无论在手动方式还是自动方式，输出变量 OUT 均受到 OH 和 OL（高限与低限）的限制

active in Auto

输出变量 OUT 仅在自动方式时受到 OH 与 OL 的限制

not active

输出变量 OUT 没有限制

Set point tracking

SPI tracks actual set point

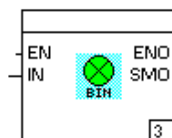
在外设定方式时内设定跟踪外设定

SPI as adjusted

内设定不改变

6. Binary Monitoring, M_BIN

· 图标



1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时，表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
IN	输入信号值	SMO	输出逻辑 1，表示响应监视功能

2、用途

用途是监视来自现场的一个数字量输入值的状态。

3、面板

标签名，Short text，工作状态文本说明，此面板是一个 1/4 小面板。

4、面板操作：无

5、参数表

Parameters: Binary monitoring M_BIN

General data

Name: [Redacted] Short text: [] Processing: ☒ Sequence: 2

Long text: []

Display text for

Message state 1: []

Message state 0: []

Message

☒ Message on 1-signal

Mess. priority: [] Hint: [] Message text: []

OK Cancel Save Reset Check Help

<< >>

Display text for

Message state 1: 输入为逻辑—1 时的状态文字

Message state 0: 输入为逻辑—0 时的状态文字

功能和功能块

Message

Message on 1—signal:

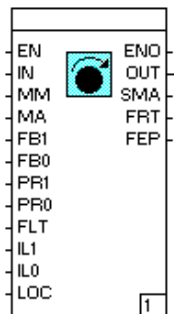
✓ 表示 IN 为 0 时，输出管脚 SMO 为 0，IN 为 1 时，输出管脚 SMO 为 1。

□ 表示 IN 为 0 时，输出管脚 SMO 为 1，IN 为 1 时，输出管脚 SMO 为 0。

Message priority: 信息优先级 1_5，如无信息组态，输入“—”。

Hint: 信息提示文字

Message text: 信息文本说明，可直接输入或从选择窗选中。

7. Individual Drive Function for Unidirectional Units, IDF-1

· 图标

1、输入/输出端脚说明

输入端脚	输入端脚说明	输出端脚	输出端脚说明
EN	当其输入为 Logic_1 时表示此功能块允许执行	ENO	输出值为 Logic_1，表示此功能块处于执行状态
IN	输入信号	OUT	输出指令，用于控制电机或阀的开关
MM	手动工作方式控制信号	SMA	操作方式状态指示：逻辑_1 表示工作在自动方式；逻辑_0 表示工作在手动方式
MA	自动工作方式控制信号	FRT	输出为逻辑_1 时，表示运行时间错误

FB1	结束位置 1 的确认信号	FEP	输出为逻辑_1 时，表示结束位置与控制指令不符，是结束位置错误
FB0	结束位置 0 的确认信号		
PR1	安全干预 1		
PR0	安全干预 0		
FLT	故障		
IL1	禁止输入信号 1		
IL0	禁止输入信号 0		
LOC	就地干预		

2、用途

此功能块是一个输出单一控制指令，控制单一方向电机或电磁阀设备开关的功能块。手动控制方式 **MM** 和自动控制方式 **MA** 均有效；手动控制方式和自动控制方式的优先级取决于 **MM** 和 **MA** 的输入值，当两者均为 **Logic-0** 时，表示可以在手动控制方式和自动控制方式之间选择，如果两者都为 **Logic-1** 时，手动控制方式优先级高于自动控制方式优先级。

控制输出指令的最终输出值取决于输入信号、安全干预、禁止输入信号 **0** 和 **1**、就地干预等信号。当控制命令改变时，同时开始监视结束位置反馈信号是否已到达应有的结束位置，监视时间即运行时间(结束位置到达时间)在面板上有时间与报警指示。反馈信号可以由外部给定(即反馈变量 **FB0** 与 **FB1** 有效)或内部给定。如果运行时间超出给定值，会在操作员站上产生运行时间错误报警。

如果没有控制输出命令而设备离开某种结束状态，就会在操作员站上产生一个错误信号（即当电机运行或电磁阀处于某种位置时，没有控制命令改变它时，它不能进入其它状态）。结束位置的监视不受运行时间的影响。

禁止输入信号 **0** 和 **1**

禁止输入信号 **0** 和 **1** 在手动与自动方式都有效；控制指令的输出由 **IL0** 与 **IL1** 预先控制。

安全干预

如果 **PR0** 与 **PR1** 同时为 **Logic-1** 时，**PR0** 的优先级高于 **PR1**。

就地干预

功能和功能块

如果 LOC 为 Logic-1 输入时，表示就地控制有效，即电机与电磁阀从一个就地的变送器接收控制信号。此时功能块保持它原有的操作方式，它的输出端 OUT 跟踪结束位置输入端(FB0 和 FB1)。

故障

FLT 故障信号会取消控制指令信号，并将功能块的工作方式切换为手动方式，接着由参数表中的“Reaction after trouble”定义其响应信息。

3、面板

标签名、Short text、结束位置的说明文字（如 ON，OFF），激活的结束位置背景色为黄色，未激活状态背景色为灰色；当组态了禁止控制命令(IL1 与 ILO)后，在相应的结束位置就不会有激活状态，其背景色总是灰色。

RT 表示运行时间，M 表示手动方式，A 表示自动方式。

依据信号的变化，在面板上会有如下信息报警：

- ✎ 输入信号故障
- ✎ 运行时间超时
- ✎ 没有控制指令而改变结束位置
- ✎ 安全干预信号
- ✎ 就地干预信号

4、面板操作

改变手动（M）和自动（A）工作方式，在手动方式下，用按键 1 和 0 控制电机的开和关。

5、参数表

(表 1/2) :

The screenshot shows a software dialog box titled "Parameters: IDF for unidirectional units IDF_1 [1/2]". It contains several sections for configuring a function block:

- General data:** Includes fields for "Name:" (a red box), "Short text:", "Long text:", "Processing:" (checked), and "Sequence:" (set to 1).
- Status texts:** Includes dropdown menus for "Status text 1:" and "Status text 0:".
- Run time monitoring:** Includes a "Max. run time:" field (set to T#5s) and a checked checkbox for "Monitoring ON".
- Reaction after trouble:** Includes radio buttons for "MANUAL and OFF" (selected) and "previous values".
- Feedback:** Includes checked checkboxes for "Feedback variable available" and "Monitoring end position ON".

On the right side of the dialog, there are buttons for "OK", "Cancel", "Save", "Reset", "Check", "Help", and navigation arrows "<<" and ">>".

Status texts:

status text 1: 1 结束位置状态说明文字
 status text 0: 0 结束位置状态说明文字

Run time monitoring:

Max. run time : 以时间格式写入，如 T#2ms
 Monitoring on : ☒ 监视运行时间有效
 ☐ 监视运行时间无效

Reaction after trouble:

Manual and off : ☒ 当接收到故障信号时，此功能块改变为手动工作方式，由输出端 OUT 输出控制命令 0

功能和功能块

Previous values : ☒ 当接收到故障信号时，此功能块工作方式不变，并保持故障前的输出值不变。

Feedback :

Feedback variable available: ☒ 允许反馈结束位置 FB0 和 FB1
☐ 不允许反馈结束位置信号 FB0 和 FB1

monitoring end position ON: ☒ 监示结束位置
☐ 不监示结束位置

(表 2/2) :

Type	Prio.	Hint	Message text
Fault:	[Blue square icon]	-	[Empty text box]
Time:	-	-	[Empty text box]
End pos.:	-	-	[Empty text box]
Protect.:	-	-	[Empty text box]
Local:	-	-	[Empty text box]

Manual start mode

☒ OFF
☐ ON
☒ Access

Operating mode

☒ MAN ☐ AUTO
☐ Lock MAN
☐ Lock AUTO

Buttons: OK, Cancel, Save, Reset, Check, Help, <<, >>

Manual start mode:

此选项为功能块在手动时的起动方式或无外部控制输入时的起动方式

OFF: ☒ 控制端脚 OUT 输出 logic—0
ON: ☒ 控制端脚 OUT 输出 logic—1

Access: ☒ 手动方式时允许在操作员站的面板上改变输出控制
☐ 不允许在操作员站的面板上改变输出控制

Operating mode:

功能和功能块

AUTO:	☒	工作在自动方式
MAN:	☒	工作在手动方式
Lock MAN:	☒	不允许转换到手动工作方式
	☐	允许转换到手动工作方式
Lock AUTO:	☒	不允许转换到自动工作方式
	☐	允许转换到自动工作方式