

APROL R 3.2 - 组态

B2 项目组态



贝加莱(B&R)公司对本手册的产品/文本信息不做任何承诺,说明。此外,公司对(B+R Essen)对和这些产品相连,来自这些产品的设备,产品本身的性能和使用不负直接或间接的责任。贝加莱(B&R)保留对该手册内容修改的权力,并且事先不做任何告知。

贝加莱(B&R)保留所有包括翻译,重印和整个文档(或部分章节)复制权限。没有贝加莱(B&R)书面授权,该手册的任何部分都不能以任何形式任何方式,包括电子或机械,照相,录制等方式复印或发布,甚至以教学为目的的信息储存和摘取。

B2 项目组态

(版本 1.60/2 月 1 日) 2007)本文档属于 **APROL R 3.2**

© 2007 Bernecker + Rainer Industrie-Elektronik Ges.m.b.H. 贝加莱工业自动化

作者: Gerd Langhammer, Karl Schulte, Mario Sagrillo

*项目组态*手册包含下列手册

- B1 组态环境**
- B2 项目组态**
- B3 CAE 库组态**
- B4 项目和库文档**
- B5 下载与调试**

该手册集订货号:

MAAPENGI30-CHI

贝加莱工业自动化

A-5142 Eggelsberg 120

Tel.: +43 / (0)7748 / 6586-0

Fax: +43 / (0)7748 / 6586-26

E-mail: office@br-automation.com

主页: www.br-automation.com

www.aprol.com

目录

B2 项目组态 V1.60

B2-1	过程项目元件基础	1-1
B2-1.1	创建新项目元件	1-1
B2-1.2	保存和结束设置	1-2
B2-1.3	创建和编辑注释页	1-4
B2-1.4	项目元件的快捷菜单	1-5
B2-1.5	版本控制	1-6
1.5.1	激活	1-6
1.5.2	编译	1-7
1.5.3	激活和编译	1-7
1.5.4	确认	1-7
1.5.5	恢复	1-7
1.5.6	删除	1-8
1.5.7	特性	1-8
B2-1.6	另存为，复制和剪切	1-8
B2-1.7	重新编译一个CAE项目	1-9
1.7.1	要求列表概述	1-12
1.7.2	远程编译	1-14
B2-1.8	CAE数据库的优化	1-15
B2-1.9	保存CAE数据库	1-16
B2-1.10	全局控制变量内存重组	1-18
B2-2	硬件配置	2-1
B2-2.1	硬件配置概要信息	2-1
2.1.1	控制器任务等级	2-3
B2-2.2	控制器	2-5
B2-2.3	2005 系统控制器	2-6
2.3.1	2005 控制器系统增加模块（主站/主架）	2-7
2.3.2	通过扩展主站（2005）配置扩展	2-10
2.3.3	配置控制CPU	2-12
B2-2.4	配置CPU属性	2-31
2.4.1	配置SG3 对象的Sysconf	2-31

2.4.2	配置SG4 对象的Sysconf	2-43
2.4.3	Can2DD任务, 182527	2-47
B2-2.5	模块和卡的配置	2-57
2.5.1	远程I/O主站配置	2-57
2.5.2	I/O卡配置	2-58
B2-2.6	配置/添加远程I/O站	2-61
B2-2.7	PCS计算机配置	2-62
2.7.1	控制计算机任务配置	2-65
2.7.2	控制系统模块配置	2-66
2.7.3	配置登录用的外部硬件	2-74
2.7.4	操作员自动登录	2-75
B2-2.8	配置过程控制系统网关	2-76
2.8.1	控制计算机中全局变量的网关设置	2-77
2.8.2	APROL外部系统接口	2-81
2.8.3	连接到APROL E	2-81
 B2-3	 参照IEC 61131 的逻辑设计	 3-1
B2-3.1	功能表概述	3-1
3.1.1	功能表和功能模块的物理视图	3-2
3.1.2	创建功能图表	3-6
3.1.3	功能图表的结构和属性	3-7
3.1.4	功能图表中的功能模块	3-10
3.1.5	报警组变量	3-12
B2-3.2	编辑功能图表	3-14
3.2.1	分配I/O条	3-16
3.2.2	添加和放置功能模块	3-33
3.2.3	功能模块的逻辑连接	3-44
3.2.4	显示驱动功能模块的附加属性	3-47
3.2.5	功能图表导航	3-51
3.2.6	功能图表注释	3-52
B2-3.3	为功能图表生成任务	3-53
B2-3.4	调试功能图表	3-56
B2-3.5	打印功能图表	3-58
 B2-4	 过程图	 4-1
B2-4.1	动态过程图基本信息	4-1
B2-4.2	创建过程图	4-1
4.2.1	插入动态图片	4-5

4.2.2	过程图里网格的使用	4-7
4.2.3	在过程图中插入操作选项	4-7
4.2.4	过程图属性	4-8
B2-4.3	在过程图里使用控件	4-11
4.3.1	创建控件	4-12
B2-4.4	使用窗口小部件概述	4-14
B2-4.5	DisplayEditor概述	4-15
4.5.1	保存设计者特别设置	4-15
4.5.2	创建图片对象特点	4-17
4.5.3	使用网格特点	4-18
4.5.4	层技术	4-18
B2-4.6	DisplayEditor功能	4-21
4.6.1	“文件”菜单	4-21
4.6.2	“插入”菜单	4-22
4.6.3	“工具”菜单	4-23
4.6.4	“查看”菜单	4-25
4.6.5	“其他”菜单	4-26
4.6.6	“帮助”菜单	4-27
B2-4.7	创建对象注意事项	4-29
B2-4.8	鼠标中键的使用	4-30
B2-4.9	编辑已有对象	4-30
4.9.1	图像对象分组	4-31
4.9.2	使用对象列表	4-32
4.9.3	动态化概述	4-34
B2-5	参数管理	5-1
B2-5.1	参数管理概述	5-1
B2-5.2	参数管理的许可条件	5-2
B2-5.3	系统部件和单元的示例	5-3
B2-5.4	配置系统组件	5-5
B2-5.5	创建参数模板	5-6
5.5.1	参数应用	5-8
5.5.2	创建参数集	5-9
B2-5.6	在功能图表中使用参数	5-10
B2-6	附录	6-1
B2-6.1	版本历史	6-1

***APROL* 文档**

B2 项目组态 V1.60

属于手册设置
工程

B1-1 简介

B1-2 安全登录

B1-3 CaeManager 基础

B1-4 设计者管理

B1-5 认证系统

B1-6 项目计划

B1-7 一个 *APROL* 项目

B2-1 过程项目元素基础

B2-2 硬件配置

B2-3 符合 IEC 61131 的逻辑设计

B2-4 过程图

B2-5 参数管理

B3-1 CAE 库

B3-2 库模块

B3-3 图形宏

B4-1 建立文档

B4-2 过程控制项目文档

B5-1 项目下载

B5-2 系统管理 t

B5-3 PLC 维护

B5-4 功能图调试

B5-5 自我监视系统

B5-6 APROL 系统信息

目标组，规定，格式

这本手册<<**B2 系项目组态**>> 适用于负责安装 Linux 操作系统和**APROL**系统软件，及**APROL** 系统配置的系统管理员和项目工程师。

手册中 “**B1 工程环境**” 会用到这个：

在该手册中使用了下面的格式：

按钮	[Esc] - button
主菜单	„Modul-Group/open“
目录名	HOME/ENGINE/HTML/001

在该手册中下面的图标代表的含义如下：

-  列表
- 

提示或建议

- 

注意事项

- 

警告

B2-1 过程项目元件基础

B2-1.1 创建新项目元件



关于对象组的信息，即历史版本，可以在此手册的附录中查找。

工程中项目元件在 **B1 工程设计环境** 手册的第一部分的表中列出（**CaeManager 功能概要**）。在这一章节中，功能图作为项目元件的实例阐述。如果其他未包含在功能图的项目元件附加信息必须进行研究，这些信息将在相应的章节中阐述。

在已保存的功能图的目录下，插入一个新的功能图，请使用“**插入 /功能图**”。你必须使用“功能图”菜单项来建立一对输入连接功能图。

图1：新功能图菜单项

在“名称”区域内，输入功能图的名称，用来存放在 **CaeManager** 中。在“描述”区域内，列出与名称类似的功能图的简短介绍。在相应的格中，项目元件类型将显示“功能图”。

在“实例”区域内，输入以字母打头后跟数字的唯一的字符，以区别与系统中的其他实例。

例如：SF01, SF02 或 DF1。

创建功能图部分包含了**物理视图/任务**和“本地任务变量编辑”检验栏的信息。

在这个窗口的确定输入可以在以后改变。需要改变时，标记到被保存的该项目元件，然后打开这个简捷菜单。

在你输入了所有需要的项后，确定后点击[OK]。项目元件现在就被创建并将被显示在 **CaeManage** 的一个输入。不过，此时没有功能输入项在这个元件中，版本开始是 1.0。

B2-1.2 保存和结束设置

在你打开并**作出改变**后，每个项目元件将被完整的保存于一个新的版本。如果你放弃保存新的改变的要求，它将被放弃和删除。有两个可选项供你保存你的工作并结束你的窗口：

- ✓ 直接点击 **CaeManager** 的导航栏区域。
- ✓ 选择“文件/保存”。此时项目元件将被保存，但没有关闭。关闭项目元件，只需要点击导航栏区域。

以前的版本（在调出当前功能之前的当前版本）将无变化的保存。

 选择“**文件/关闭**”来结束所选择的工程！

Eintrag speichern:

Name:

Beschreibung:

Erzeugt:

Benutzer:

Version:

Modifiziert:

Benutzer:

Gespeichert wird:

Version:

Modifiziert:

Benutzer:

Kommentar:

图 2: 保存前的菜单项窗口

在保存前，在左边显示的“**保存输入**”菜单项将提供相关信息，并允许你进行输入。

在此输入的关于版本的注释，不是项目元件的描述。

在你做了大量修改以后，你能改变版本号，保存你的工作在新的发布下。你如果选择**[新发布]**按键，版本的第一个数字号将被加一。

此时你能点击**[OK]**保存你的工作。

B2-1.3 创建和编辑注释页

注释页可以让设计者建立关于独立项目元件版本的具体的文档。请确信关于项目元件的描述或者版本的注释没有错误。注释页将保存在相关的项目元件的目录下。注释页应该包括记录和描述以提供关于项目元件版本状态的情况。

注释页的维护与它保存的工程一样。一个具体描述的调出和编辑注释页在以下的部分提供：

一个功能图的注释页的描述举例如下：

打开项目元件之后，选择工具条上选择盒或者“视图/注释”可以调出相关的注释页。

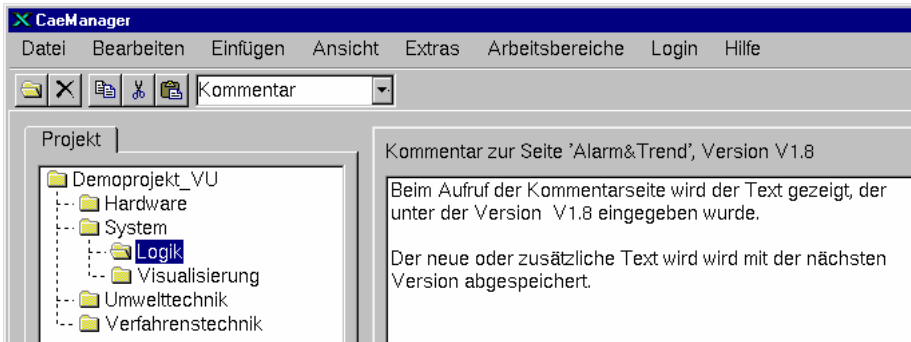


图 3： 编辑注释页

作为规则，你能通过选择块来切换注释页和项目元件。

如果项目元件再次打开，注释页显示的注释是保存的以前版本的。新的。附加的或者改变后的文本作为新的版本保存，因此可以供以后所调用。

 **每个版本都有自己的注释页！**

保存在保存和结束设置章节里进行了描述。如果注释页存在，会显示  图标。

B2-1.4 项目元件的快捷菜单

Ausschneiden
Kopieren
Einfügen
Löschen
Umbenennen
Akt./Komp. (letzte Vers.)
Kompilieren
Generieren
Eigenschaften
Versionen

如果你选择后，但却没有打开项目元件，你能打开如左图所示的简捷菜单。

请注意这些菜单项的执行依据于项目设计组中设计者的权限划分。下面部分将包括一个简短的单个菜单项的描述。

图 4： 项目元件的简捷菜单

下表列出了单个菜单选项：

菜单选项	菜单选项的描述
"剪切" "复制" "粘贴"	这些菜单选项的描述在 保存，复制和剪切 章节。
"删除" "重命名"	这些菜单选项是自我说明的。
"激活/编译（最后版本）"	选择这个菜单项将激活最后的版本并立即编译。
"编译"	编译已经激活的版本。
"生成"	生成全部的下载项目。这个可运用与现场控制器，也可以作用于控制计算机。
"特性"	这个选项允许你改变项目元件的描述。
"版本"	这个菜单项的描述在 版本控制 章节中。

B2-1.5 版本控制

直至现在我们一再的解释，当保存时，每个项目元件（只要发生改变）被完整的保存于新版本以及**新的**版本号。每个版本都可以用于过程控制系统工程，只要它能产生功能。你能使用简捷菜单访问一个项目元件的个体版本列表（看项目元件简捷菜单部分）。

下图显示了在 CaeManager 中的版本列表。

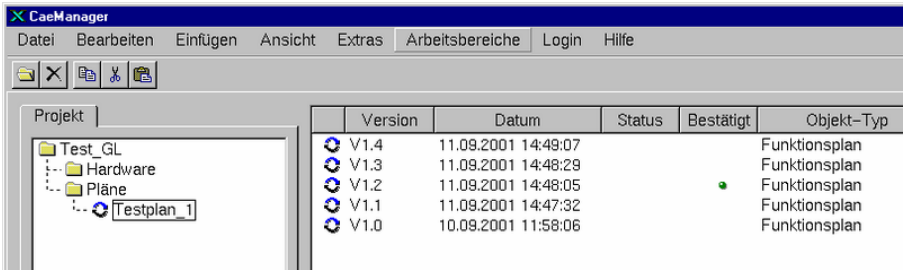
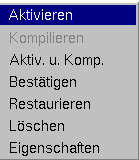


图 5: 已保存的版本



在打开简捷菜单后，每个版本的菜单项将被显示（看左图）。为了能够选择这些菜单项，项目元件的配置必须没有错误。

图 6: 简捷菜单的版本

请注意这些菜单项的执行依据于项目设计组中设计者的权限划分。下面部分将包括一个简短的单个菜单项的描述。

1.5.1 激活

标有记号的所选择的版本是最后的编译的。只有一个版本能被激活。如果另一个版本已经被激活，此版本将被解除激活。在激活之后，一个红点将显示在输入列的**状态**中。

1.5.2 编译

选择和激活的版本将被编译，生成以后的工程。程序码将被产生。如果在编译时有错误产生，错误信息将被显示，一个红点也将标记于状态。成功的编译后将使输入列显示“绿色”的状态。

作为惯例，正在配置的项目元件也需要编译。这是因为：

在配置之中，你将产生输入点，这将被以后的项目元件所使用。在编译之后，你所做的配置将通过数据库被其他项目元件使用。

1.5.3 激活和编译

在选择这个菜单项后，激活和编译将被在一个步骤中执行。

1.5.4 确认

版本确认是一种检查。在确认下的版本会出现一个绿色点显示在输入列。在指定的版本号之后，系统将删除以前的低版本（环形数据缓冲管理）。当创建一个项目时，你能指定系统检查应该删除的版本号。更多的信息，查看 [创建新项目](#) 章节（在 *B1 工程师环境手册*）。

1.5.5 恢复

如果不能纠正的错误堆积起来，在后续的项目元件开发中，你能继续配置工程通过使用一个过去的发布的低版本。老的操作版本（可以被标记“确认”）这样就被恢复了。

选择“**恢复**”后，此版本能自动保存为新的高版本（版本超前于所有的显示版本）。如果你想再次开始，你能选择一个新的发布。

也有一种恢复的简捷方法。打开操作版本，通过（双击）选择，然后编辑。保存之后也将在最高列表位置显示。

1.5.6 删除

如果你选择“删除”并且确认后，此版本的项目元件将从项目中删除。



在删除后，此项目元件的版本被放置于垃圾箱中，在垃圾箱中删除后将被完全删除，或者也可以将其从垃圾箱中恢复回项目中。

1.5.7 特性

在选择这个菜单项后，你能改变项目元件的描述。版本将会在“版本特性”菜单选项窗口显示信息（早期的描述）。

B2-1.6 另存为，复制和剪切

项目元件在打开之后可以保存以一个新的名字。操作时，请选择“文件/保存于”，接着开始编辑相应的条目。



文件被保存于原始的目录。只有最终的版本会被保存！实例保持打开的状态，此实例肯定会被改变！

“保存”功能还有其他的优势。允许复制所有的版本到其他的目录，甚至复制到其他的项目。当复制入其他的项目，需要有相应的项目读取权利。

做此操作时，选择“复制”从简捷菜单中。选择你要复制的文件的目录。在此，从另一个简捷菜单输入“粘贴”命令。所有的项目元件的版本就被复制到这个路径。为了避免错误，已在的参考将被删除，实例将被关闭。实例（已被关闭）打开之后可以输入在“特性”栏中。

“剪切”功能仿佛“复制”，参考不会被删除，实例也不会被关闭。在此，项目元件在原来的目录被删除，而被插入到所选的目录中。

B2-1.7 重新编译一个CAE项目

APROL 项目设计可以利用重新编译功能方便地完成一个完整的 CAE 项目或编译一个完整的硬件，所以要事先将其下载到目标系统。

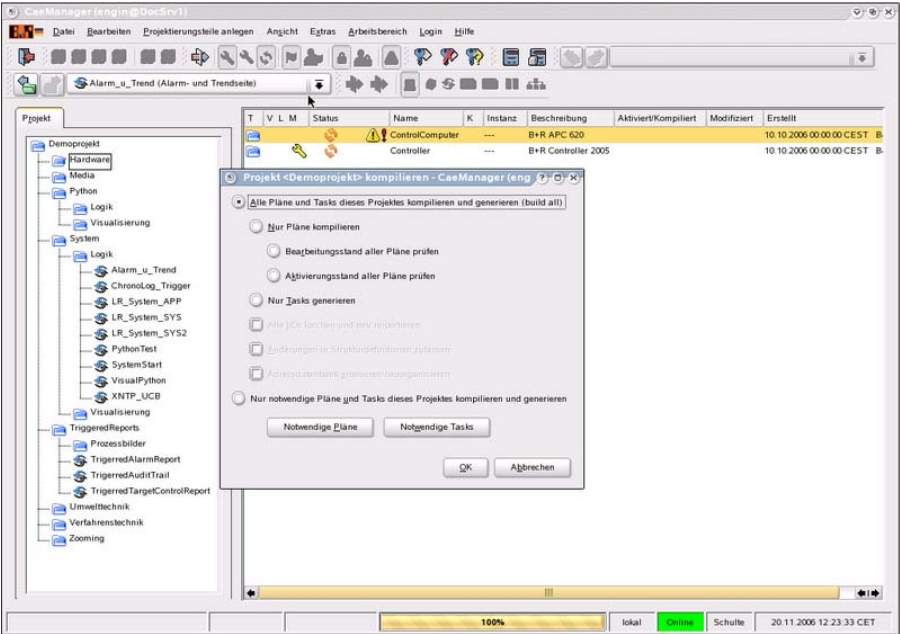


图 7： 重新编译一个 CAE 项目

调用 **CaeManager** 中的菜单选项“编辑/重编译”：

通过单选按钮进行设置：

选项	说明：
编译并生成项目图表和任务	编译所有 已被激活 的图表，生成所有 现在的任务
仅编译图表	编译所有项目中被 激活的功能表
检查所有图表的工作进度	检查所有功能表，在扩展项目前发生错误，并提供给生成过程。 在选择菜单项“检查所有图表的工作状态”之后，所有包含状态工作版本（已修改）的逻辑计划，将通过系统检查。

选项	说明:
检查所有图表的激活状态	检查所有功能表，在扩展项目前发生错误，并提供给生成过程。 选择菜单项“检查所有逻辑图的激活状态”，即检查所有激活逻辑图。
仅生成任务	生成 所有任务 。
删除所有 I/O,并重新导入	此时，不支持此功能
允许定义结构时作适当改变	选项仅用于控制器： 如果改变 PVS 的数据类型和结构，仅生成控制器任务。
更新/重组地址数据库	此时，不支持此功能

通过下列功能进行**选择性重新编译**:

选项	说明:
仅编译并生成项目所需的图表和任务	生成，编译必需的功能图标和任务，在内部系统列表即要求列表中管理项目设计步骤。 通过此选项编译列表中所有的逻辑图。最后，在任务列表中生成所有的任务。
[所需图表]-按钮	在 CAE 中修改的功能图表必需编辑，列于要求列表中管理。 在 <u>章节要求列表的概述</u> 中列出工程设计步骤。 一旦在要求列表中输入一个功能图标，按钮即变。 在确认了[所需图表]之后，对话框中选择所有必要翻译的功能图表。 在对话框中重新选择单个功能图表。
[所需任务]-按钮	在 CAE 中修改之后，有必要生成一个任务管理要求列表。 在 <u>章节要求列表的概述</u> 中列出工程设计步骤。 一旦在要求列表中输入一个功能图标，按钮即变。 在确认了[所需任务]之后，对话框中选择所有必要的任务。 在对话框中重新选择单个功能图表。



或者，可在硬件（控制器或控制计算机）的快捷菜单中“编译/生成所有”中调用对话框。
在这个情况下，所选硬件的对话框中的选项是有限的。



当重新编译的程序中断，警告框“数据库处于不完整状态”弹出。
然后重新选择编译。

下面的图作为一个实例，所选择的功能图表将被编译。选择带“绿勾”的功能图表。

在表的左栏通过鼠标重新选择功能图标。

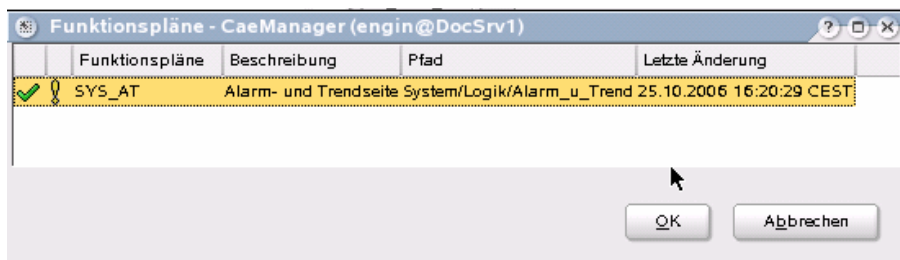


图 8：选择需要编译的功能图表框



通常，项目文档的索引是自动创建的。基于文档的大小可能要长的时间。

你可以在“项目选项”标签的“项目属性”中取消自动索引。



在每个项目中，在 **CaeManager** 中要求列表的所有的输入都被标有重新下载的图标。

激活后功能图表被标记，修改的功能图表编译后，生成的硬件也被标记。



不同版本的相应的项目元素中存储标记和 CAE 消息，在整个项目中显示

如果没有按照 *CaeManager* 的指导（如，不执行编译和生成），当通过 *DownloadManager* 下载，会再次提供此类信息。

1.7.1 要求列表概述

下面列表讲述了工程设计的步骤，在要求列表中输入（功能图表/任务即生成）。



为了提高工程设计的效率 and 安全性，必须保持升级的功能。
因此，必须经常关注主页 www.br-automation.com 上的文档状态。

什么时候在要求列表“功能图表”中输入？

A) 在 CAE 项目的更换：

- 激活一个功能图表

- 编译一个功能图表

如果一个输出芯的物理视图改变，所有含有输出芯的功能图表都被输入到列表中。
要编译的计划首先添加到列表，当中断发生，则选择的重新编译列表已经完成。

- 重新编译所有图表

首先，将图表输入到列表中，中断后，则选择的重新编译列表已经完成。

- 升级 **APROL** 系统软件/备份 CAE 数据库/不一致的数据库 *pcpdb*

所有图表输入到列表中，选择的重新编译列表已经全部完成。

B) 在 CAE 库中结构化的更换一个项目部分：

概述：

- 更改输入和输出芯（除外：描述）- 模块的版面更换（宽和高）
- 更换标记“有用的 CC/控制器

功能/功能模块：

- 本地变量的更换

功能模块：

激活或取消“Init”或“Exit”代码。改变“Remanent”标志。

超级宏：

每个改变。

什么时候在要求列表“任务”中输入？

A) 在 CAE 项目的更换：

- 生成一个任务

首先，将所有将要生成的任务输入到列表中，中断后，则选择的重新编译列表已经完成。

- 编译一个功能图表

当一个任务不再是功能图表的一部分，则输入列表

- 重新编译所有图表

首先，将图表输入到列表中，中断后，则选择的重新编译列表已经完成。

- 重新编译来自硬件的所有任务

首先，将所有任务输入到列表中，中断后，则选择的重新编译列表已经完成。

- 升级 **APROL** 系统软件/备份 CAE 数据库

所有任务输入到列表中，选择的重新编译列表已经全部完成。

- 概述

如果一个功能图表被删除，取消或者重命名，是计划的一部分的任务都被输入到列表中。

如果重新组织的任务从文件系统中删除，则被输入到列表中。

全局 PV 的失效偏移量：

（从控制器中）

- 间断性检测到所有偏移的 PV
- 所有第二个 PV 会占有一个偏移量，从控制器地址数据库删除（文件夹 *dbk*）
- PV 中的任务输入到要求列表中
- 当生成任务时，这些 PV 得到新的偏移量

什么时候在要求列表“任务”中输入？

目录 /home/<Engineering-System>/ENGINE/PROJECTS/<Projektname>.pgp/pgm/<Ortswelt Controller>/<CPU>/丢失

预先生成物理视图然后做检查，只有单个任务或者最新编译的[所需任务]应该被执行。如果不存在，所有物理视图的任务都输入到列表中，输出警告框。

B) 在 CAE 库中结构化的更换一个项目部分：

1. 功能模块：

- 更换 Init，或 Exit 程序代码（如果存在）

2. 功能/功能模块：

- 程序代码的标题，内容声明有所改动

3. UCB 模块：

- 标记“实例-全球锁定”或者“冗余从站的执行”

4. 图片模块：

- 更换制定图片内容

- 更换 Python 代码

1.7.2 远程编译

通过远程编译，可以加快扩展的 CAE 项目的进度。配置此项服务，请参考手册“A2 使用入门”。

在成功配置远程编译以后，编译时，你可以在 CaeManager 管理器中分配任务。

在弹出对话框中，点击按钮 **[远程编译]**，可以了解有关操作系统信息。下图是在 iccream 监视器(icemon)中编译器的概图。icemon 中带有菜单帮助。

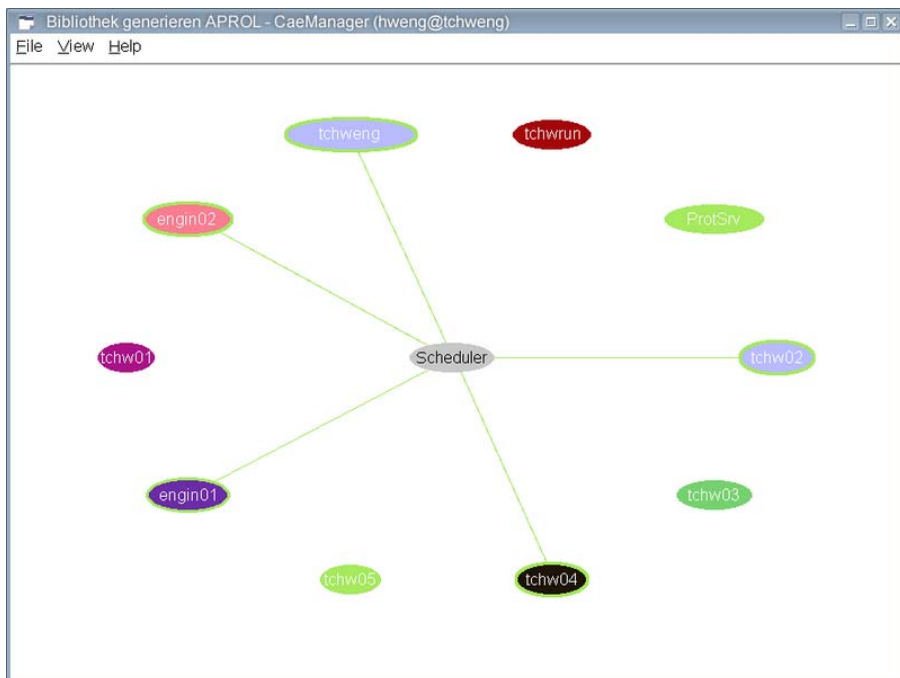


图 9: Icecream 监视器中显示编译组

B2-1.8 CAE数据库的优化

CAE 数据库是工程组态的核心文件。它位于工程设计系统目录..ENGINE/,包括 caedb.link 和 caedb.tree。这个数据库储存了所有的在工程设计系统 CaeManager 创建的,选择的项目。

在项目配置中如果功能图的数据被删除,删除后的实例将不能够在项目中使用,但仍可以保留在数据库中。这个实例未必会增加 caedb 的容量。

CaeManager 的“其他”菜单中包含了“优化数据库”的菜单选项。



这个菜单项只会对当前打开的项目起作用。

如果选择“**优化数据库**”后，系统会确定所有保存在项目中的实例都被使用。如果确定有些实例未被使用，这些实例将被删除。因为这个过程在大的项目中要花费一些时间，所以有进程显示。



在项目完全将不必要的占用空间释放出来之前，这个功能应该不断地使用，以尽可能得保证数据库的最小（数据库，储存于磁盘/CD 等）。并不必要经常使用这个功能。

B2-1.9 保存CAE数据库

APROL系统可以通过网络，多个设计者同时读取数据库进行开发。可以多个设计者进行工程组态。在这种情况下，在**运行时**，使用常规的操作系统命令保存数据库是不可能的。

APROL 系统在 **CaeManager** 中，在“**其他**”菜单下提供“**保存数据库**”选项就是因为这个原因。这样你能在运行时在 **CaeManager** 下保存本地副本。以下条件必须满足：

- ✓ 设计者必须有数据库管理员权限。
- ✓ 你是通过 **CaeManager** 唯一读取数据库的人。
- ✓ 当数据库保存时，你没有打开其他任何的项目元件。



CAE 数据库保存了两个文件：*caedb.link* 和 *caedb.tree*。两个文件要**经常**保存。

只当点击“**保存数据库**”下复制时，将被保存在工程设计系统的 **HOME** 目录下。点击[BROWSE]键，你能容易在其他目录或者另外的存储介质里保存。

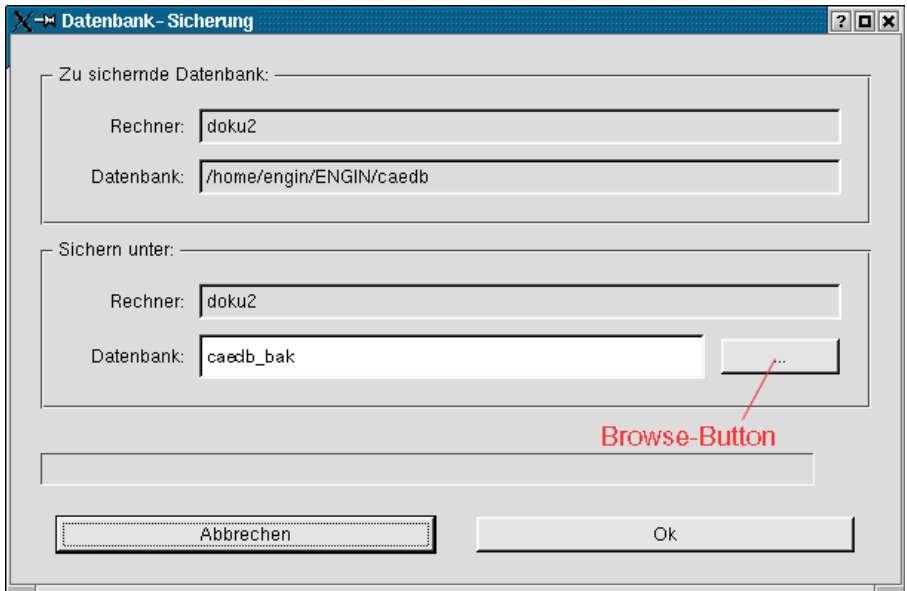


图 10: 保存数据库菜单



保存数据库时，容量也将被优化。



这样保存数据库不能代替正常的文件系统备份。后者的保存必须在相应的电脑，Linux 系统，使用 ROOT 登陆，在 INIT 1 或 2 级下（事先停止所有 APROL 的模块）。

B2-1.10 全局控制变量内存重组

每个现场控制器的 CPU 都有 32KB 内存供全局变量。这个内存可以被工程设计系统在目录

```
...ENGINE/PROJECTS/<Project-name>.pgp/dbk
```

下重新划分为每个 CPU 的数据库类型。在项目中的每个控制器都要分开管理

全局控制变量在功能图中创建。这些全局变量和功能图数据和数值一起改变任务。当数值改变时，同变量也始终被存取，不管是哪一边开始的变化。



因为可以在很多事例中使用，变量能变的非常复杂。

当一个控制器任务产生时，系统将创建连接数据库的全局变量的导入。当全局变量被验证时，数据库检查这个变量是否存在导入和类型是否匹配。如果有一个条件不满足，将创建一个新的导入。如果存在的导入和变量类型匹配，装载变量的信息（变量地址）被输入建立的编码。



下载的时候，全局内存的内容将被传到现场控制器内。

如果在项目配置中连接数据类型改变，在当前的几个任务中，或者引脚被删除，增加，或者在功能块中被改变，那么在现场控制器的全局内存的重组是很有必要的。因为当产生一个新任务时，新的全局变量的导入将在数据库中产生。老的导入将保留因为系统不能检测到这些导入是否在其他的任务里使用。

非常复杂的改变，以及从它们产生的任务将导致有效的全局内存被重写。如果发生这种情况，下载就不能再执行了。

以下步骤必须被采用于重组全局变量内存和删除不再使用的导入。



*此工作只应由熟悉下面任务的，并且非常可靠的人来执行！
不要只为了一个现场控制器尝试此进程！*

- ✓ 从相关项目中删除 dbk 目录下的内容。
- ✓ 重编译全部的项目。在此步骤下，全局数据库(dbk 目录)将被重建。
- ✓ 停止相关项目的运行服务器。
- ✓ 停止相关的现场控制器。如果工程设计系统的任务被负载在现场控制器的 USER ROM 内，它们将被删除。
- ✓ 对所有的现场控制器一次全部的初始化和冷启动。删除所有在 RAM 中的任务。
注意！在 RAM 中的任务也会被删除。它们必须被重新加载！
- ✓ 将所有的任务从工程设计系统下载到每个现场控制器。这将传输全局变量内存里的内容。
- ✓ 全面下载运行服务器系统。
- ✓ 如有必要，全面下载操作员系统。
- ✓ 下载后启动运行新的服务器系统。

B2-2 硬件配置

B2-2.1 硬件配置概要信息

在硬件配置的过程中，B&R 的部件和现场控制器（PLCs），控制器计算机（运行服务器）和工厂操作员系统的操作终端将被组合和配置。你首先应该完成工程的这个部分。当然，也能被延伸到其他的时间里。不过，你至少需要一个 PLC，一个控制计算机（运行服务器）和一个操作计算机（操作员系统），来测试过程控制系统的第一部分。在配置了需要的硬件部件以后，你只可能下载你所建立的文件（程序功能图类型的任务），文件从工程设计系统下载到过程控制系统的部件。

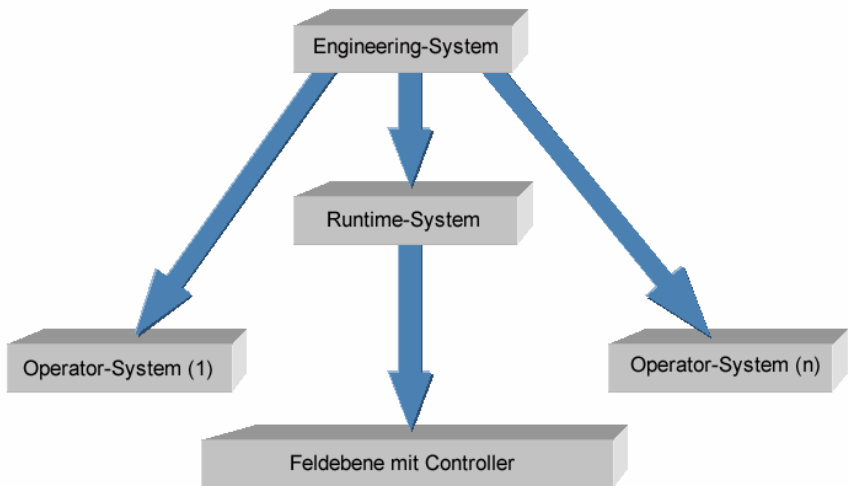


图 11: 下载中的工程数据流

在此，我想能引起你注意到这个情况：完整的配置和**APROL**过程控制系统程序只能在集成了 **B&R** 控制部件组成的控制系统应用。**APROL**系统能连接到其他的系统，而且能配置数据交换，但是你不能通过工程设计系统进行系统编程。

以下部件可以作为现场控制器（粗略划分）：

控制器	说明
主站	在 B&R 控制器的主站中，可以使用 2010，2005 和 2003 系列 CPU。在为 I/O 模块的一般配置中。主站能使用远程主站来进一步扩展或者采用扩展主站或加入远程 I/O 点扩展。 没有 I/O 模块，他只作为主站。
I/O 从站	分散的 I/O 站。它有 I/O 模块，和一个远程从站。远程从站与主站的连接通过远程 I/O 总线。具体信息，特别是可用的距离，可以在 B&R 手册中查找。
扩展	扩展可以和上面一样增加 I/O 模块和主站。每个扩展可以装备一个扩展从站（连接到扩展主站）和 I/O 模块。

 请通过 **B&R** 手册检查最终模块配置。

硬件配置也包括在过程控制系统中的计算机配置（这部分的第一段描述过）。在此不会制作详细的计算机信息列表，因为这将导致无休止的列表。将计算机集成进过程控制系统或者包括一个项目元件，使用“**插入/控制计算机**”菜单项。

 所有的计算机（控制计算机[运行服务器]，操作员站，网关服务器）都统称为控制计算机。

计算机目录“控制系统目录 4 “提高给已打开的项目元件（配置元件和硬件给计算机）选择。请选择计算机类型以配置这个目录。注意计算机需要软件狗和相应的许可在运行时。

2.1.1 控制器任务等级

稍后，控制器程序从大多数功能图中产生。一个用户程序可以表示单个功能图或者几个功能图。这些用户程序由控制器操作系统管理。这个操作系统是 B&R 多任务操作系统，所以可以几个用户程序（任务期限从此时开始）在一个 CPU（可配置多个处理器）上同时运行。处理器能在单个任务之间切换。一个任务就是在控制器上的最小的程序单元。

四种循环任务等级被控制器 CPU 划分。它们相应的循环时间被设置从 10MS 到 5S（步长）。任务被划分到相应的任务等级，按照规定的循环时间连续执行。你不必改变最高速的任务等级的循环时间。下图的两个任务等级显示如何被执行的单个任务等级。

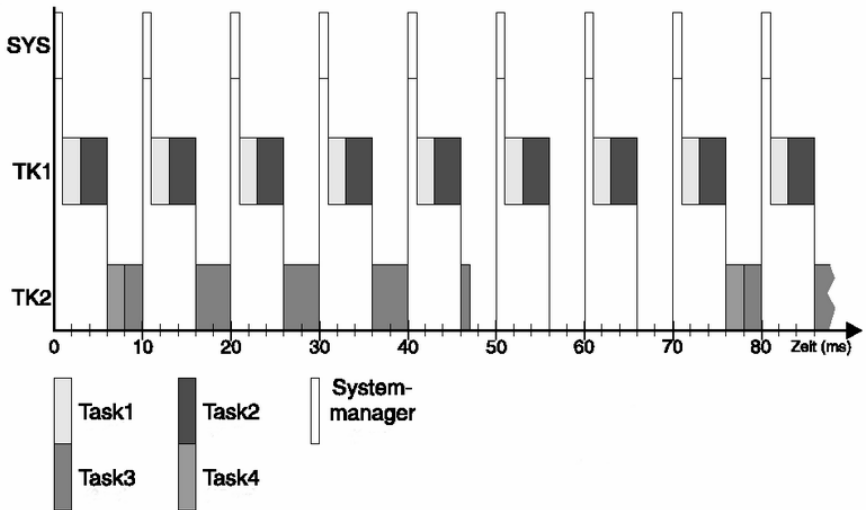


图 12: 在控制器中任务等级顺序图

两个任务等级（循环时间：TC1 = 10 ms, TC2 = 70 ms）。任务 1 和任务 2 为等级 TC1，任务 4 和任务 5 为等级 TC2。任务 1 和任务 2 每 10ms 执行一次，其他的任务保持在 70ms 执行一次在等级 TC2。你能看到近似 45ms 以后，任务 3 和任务 4 就被完成。任务 3 和任务 4 将在 70ms 后再次执行。任务 1 和任务 2 每 10ms 执行一次。所以，这样可以确保任务在确定的时间段内执行。



如果在任务中交换数据，输出数据只能被已经执行的任务采用。输入的数据也只能被其他正在运行的任务接受。

每个任务（任务管理器中有一起信息）都可以使用内存和计算机时间在控制器中，因此任务的数量应该保持在最小。不要为了每一个新的功能开启一个独立的任务。功能相近应被编译到一个任务被一起执行。

功能图，任务等级，最后和 CPU 都被重点显示在下图。一个 CPU 关联一个控制器。从下图你可以更好理解在控制计算机中 运行任务的情况。

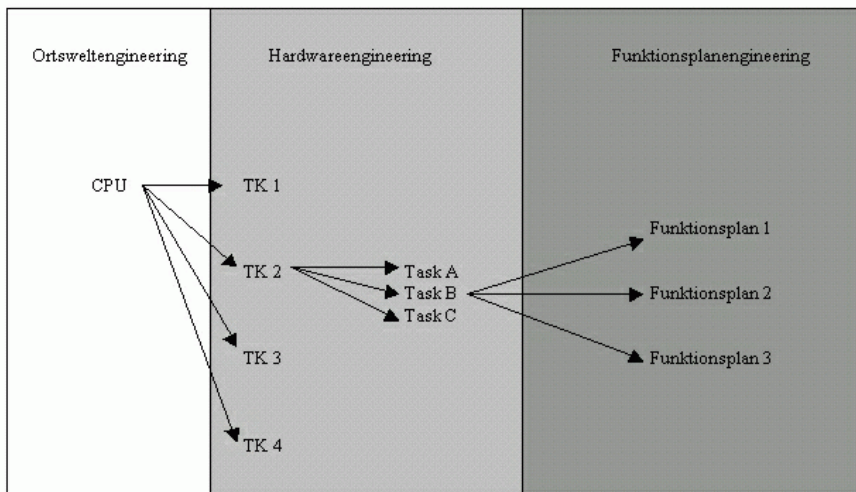


图 13: 功能图，任务，任务等级和 CPU 之间的关系

各个任务等级的任务被控制器 CPU 循环的执行。在编译功能图时候任务将被自己创建。你能将任务关联到选择的物理视图/关联任务。

下表在各个 CPU 中可能运行的任务数量。

CPU 类型	最大任务数量
Controller 2005 with CP260	150
Controller 2010 with IF260	70
Controller 2010 with CP200	200

CPU 类型	最大任务数量
Controller 2010 with CP210	200

不过，此手册的涉及主题太广泛了。其他的关于任务等级的信息可以查阅“系统硬件参考手册”，B&R 模块号：**MASYS2SYS-E**

B2-2.2 控制器

2000 系统的现场控制器被设计成满足某种特殊的需要，因此组件和模块的配置也几乎是独特的。在所给的例子中，不可能显示所有的变化。例子局限在开发一个扩展一个远程 I/O 站。更深的扩展需要或者从这个例子中延伸出的“更小”的配置。

用“插入/控制器”菜单项，控制器被插入到已打开的项目中，也可以插入到相对应的目录下。

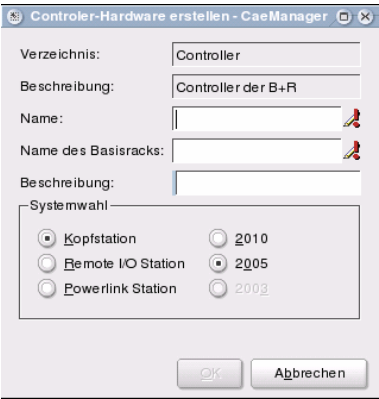


图 14：配置控制器从相应的导引菜单中

下表提供了可供单个控制器使用的应用可能性。当然，一个远程 I/O 站是需要主站的。

	2010	2005	2003
主站	X	X	X (**)
I/O 从站	X	X	X



()** 如果你想使用 2003 系列控制器作为主站，请联系奥地利 B&R 公司的销售代表。

下表显示远程 I/O 站能连接到的各种控制器。

主站	I/O 从站
2010	2010, 2005, 2003
2005	2010, 2005, 2003

B2-2.3 2005 系统控制器

此例是关于 2005 系统控制器带一个扩展和一个远程站。2005 系统控制主站和一个名叫 Basis_2005 主架被包含在这个项目中。第一次打开后，你必须配置这个站用需要的底板，电源，CPU 等等。

完成以后，你可以使用主架的名字保存控制器扩展（也可以配置其他所有需要的模块）你数据库中。



相对应远程的 I/O 站分别输入到项目中。所有你必须分别处理。

下图显示所有的配置到主架的扩展，以及关于其他扩展的选择。

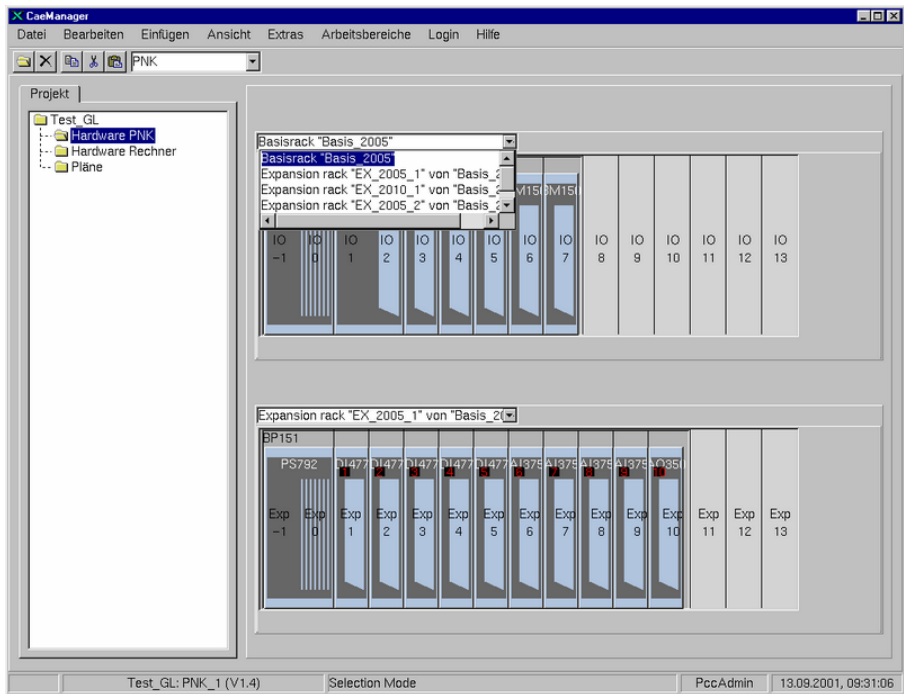


图 15: 2005 控制器扩展的完全配置

2.3.1 2005 控制器系统增加模块（主站/主架）

做此步骤，请将 B&R 关于构造和配置的建议粘住。这个行动可以在运行时进行内部检查，以为了保持错误源在最少。**配置控制器时**，你需要选择你要放置组件的位置，导出相应的简捷菜单。

简捷菜单根据元件的不同状态提供不同菜单选项。在下图中，底板模块和一些附加的模块已经加入。一个未关联点已经被选择在打开的菜单。



只有 2005 系统控制器能使用底板。

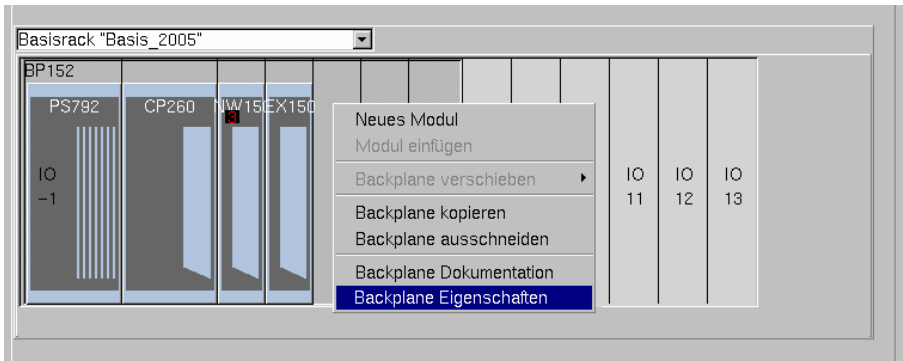


图 16： 底板和新模块的简捷菜单

下表对简捷菜单的选项做了简单的描述。

菜单选项	菜单选项的描述
"新模块"	填加一个新模块。选择此项，模块被放置在一个另外所选的窗口里。
"插入模块"	填加一个模块到已经打开的模块里。
"推卸底板"	卸掉已经填加的底板。此选项对 2005 系统没有作用。
"复制底板"	底板和底板上的所有模块都能被复制到一个同一型号的空架上。
"剪切底板"	此选项近似于复制底板。不过，源底板要被删除。
"底板文档"	显示底板信息，用文档（HTML 格式）。
"底板特性"	底板特性以一个分割的窗口显示。

 “新模块”窗口，带红叉的模块是连接到特定模块的（下图中所示）。

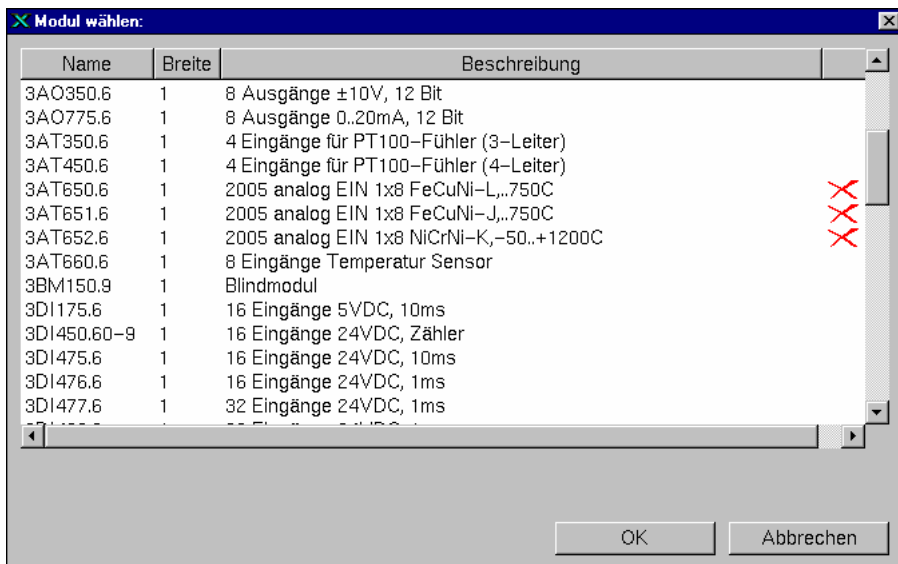


图 17: 所选模块窗口

当配置控制器时，模块被正式取消和在将来被其他所替代，要标记**红叉**，在“描述”栏里。如果模块被选择，下面的信息将被显示：

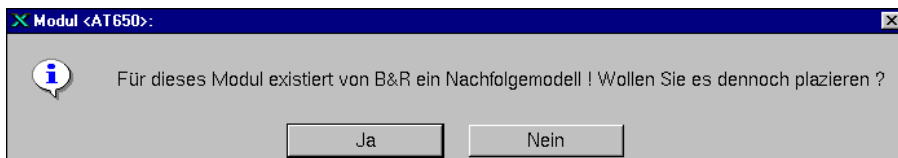
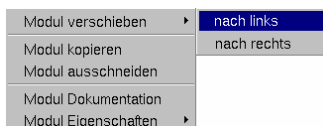


图 18: 当所选模块被放弃时被列出的信息



当新增一个模块，你能用鼠标打开象左边的简捷菜单。菜单选项是可以自我说明，在下面也列出了简单的解释。

图 19: 模块的简捷菜单

菜单选项	菜单选项的描述
"拆卸模块"	新增模块后，可以在底板上卸掉附加的槽。 连接上的模块也一同被卸掉。
"复制模块"	复件的能被插入到不同的位置。
"剪切模块"	原件被删除。 复件的能被插入到不同的位置。
"模块文档"	模块的在线文档，可以使用配置的浏览器显示。
"模块特性"	这里，你能选择附加的模块的特性（如果可以）。否则特性将被立即显示。 附加模块有： I/O 模块=可以配置输入和输出。 CPU=可以配置中央处理器。 Expansion master =可以配置扩展站。 Bus260[1]=>blank= 可以插入附加的模块。等

在控制器被装配和配置后（确定能连接到 CPU 的物理视图），它必须被激活和编译为了能够单独的功能图元件，例如： 物理视图（任务），硬件 I/O 点，交叉通讯等等。

2.3.2 通过扩展主站（2005）配置扩展

扩展被作为控制器主架的一部分从它的功能和电气连接来看。
具体细节可以从 B&R 手册中关于此控制器部分查找。

为了显示连接在主架上的扩展，在 **CaeManager** 的配置区里，必须安装使用扩展主模块（在主架上）。这个例子中，2005 系统配备了**扩展主站**模块，但也可以用独立的模块。

下图显示了安装扩展的菜单。显示了应当连接到主站的扩展。

Expansion-Master Konfiguration:

Modul:

CP152.9

Steckplatz:

Rack 'Basis_2005', ID -1

Ortswelt:

/PCC/SZ01/R001/N01/RPSA/CPU0

Funktionswelt:

Nr.	Name	Typ	Basis	benutzt
1	EX_2010	2010	30	x
2	EX_1	2005		x

Name des Ri

EX_2010

☒ 2010

benutzt: ☒

☐ 2005

benutzt: ☐

☐ 2003

benutzt: ☐

Basisadress

30

<= Anfügen

▲

Liste umstellen

▼

OK

Abbrechen

图 20: 扩展的配置菜单

在这个例子中，一个 2010 系统扩展，可以连接到 2005 系统主站，也可以配置另外的 2005 控制器扩展。

 最多四个的相同系列产品扩展能连接到 2005 系统扩展主站。如果使用 2010 扩展，存在一些限制。

已制成的个体的导入在下表显示。

输入/菜单	解释:
模块 插槽	主系统的系统信息。
物理视图	选择扩展主站的物理视图。
底板模块名	每个扩展都有自己的底板。输入唯一的名字。
单选按钮	输入你想添加的 B&R 扩展主站产品。
“使用” 检验栏	如果你选择检验栏，主系统只使用扩展。

输入/菜单	解释:
基本地址	输入基地址。 只能输入 B&R 2010 相关系列。
添加	添加扩展到列表。
重新整理列表	重新整理列表，当按此按钮。
删除	在图中不显示。 你能删除一个扩展从列表中标记的导入，“删除”它从简捷菜单。

完成配置后显示**[OK]**（可以在任何时间扩展），系统将交互式报告成功的创建组架和连接到主系统（必须确定）。

这样，在配置区域内的所有的扩展和主站都将可用并且可以装备模块。有两个架显示。你能从架子的窗口来切换扩展。相应的图显示在 2005 控制器系统 章节里。

2.3.3 配置控制CPU

现场控制器 CPU 必须根据项目工程师的需要配置。总体来说，配置根据下面项目中的元件来定。

- ✓ 在你的项目中使用的 B&R 库。
- ✓ 控制计算机和控制器之间的通讯类型。
- ✓ 你是否使用超过 2 组的 2000 系列控制器在你交叉通讯中。
- ✓ 你是否正使用特定的数据模块。
- ✓ 你是否在使用网关连接器连接控制器到另一个设备上。
- ✓ 等等。

在这种配置，你将可以支持网关编辑器在最多的部分。

在你将在 CPU 安放在底板上后，你能从简捷菜单“**模块特性/CPU**”打开配置窗口。

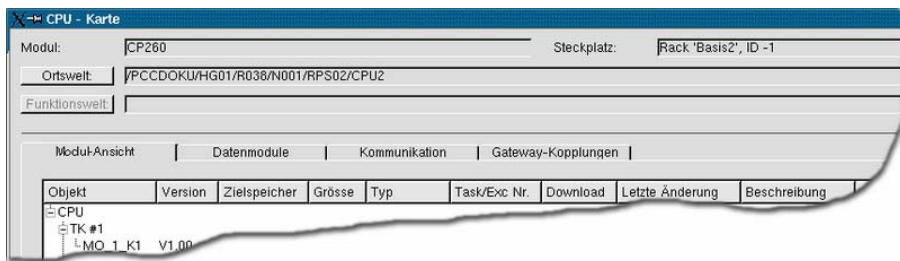


图 21: CPU 的网关编辑器

首先，你可以通过**[物理视图]**按钮定义 CPU 的物理视图。不输入物理视图，你将不能激活和编译控制器。

在物理视图树里，控制器 CPU 的物理视图必须在控制计算机物理视图下面的第一级。

你能选择四个制表符的一个来获得配置 CPU 的不同部分。你做的这些制表符的输入通过点击**[确定]**按钮接受。不过，必须保存控制器以为了永久的储存输入在工程设计系统数据库里。

网关编辑器依据惯例设计，将提供给你最佳的支持在配置单独的模块和网关通讯连接的时候。在大多数部分，配置可以标准化。可连接的数目可以增加，附加也能很容易的添加。

有一个工具条在每个制表符下面（除了模块视图）。下面提供了有关描述。

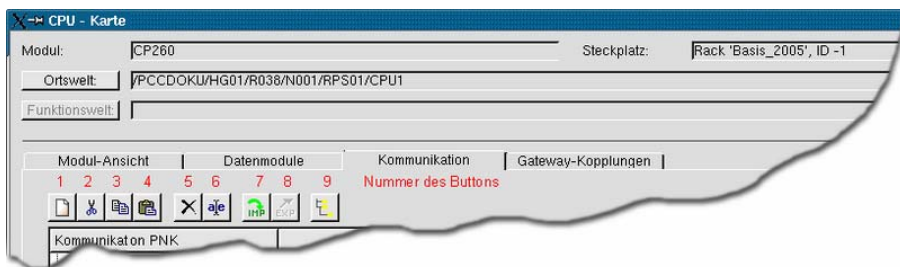


图 22: 配置控制器 CPU 的工具条

显示的按钮都是激活的状态，可以被选择使用。图标是依据数字从左到右来描述的。

No.	描述
1	创建一个新的导入从树结构中。
2	从树结构中剪切一个导入。这些导入可以粘贴到树结构的不同位置。在大多数的案例中，剪切的导入将粘贴到这种类型的最高位置。
3	复制导入在树结构中。
4	粘贴已经复制的导入。在大多数的案例中，复制的导入将粘贴到这种类型的最高位置。
5	从树结构中删除一个导入。
6	在树结构中重命名导入。
7	输入导入从一个文本文件（最好是用文本编辑器创建的）。请确定正确的格式在导入文件的时候。 注意： 只有导入的元件可以被导出。导出的文件由它的结构首先决定。
8	导出。导出一个导入在文本格式下。
9	切换视图。使用或不使用树结构。

所需要的 BR 模块在运行时会自动的在你的配置中产生。当你创建的功能图中生成相关 CPU 时，他们也会被创建同时可以被下载。所有你创建的配置会自动进入 CPU 的配置里。

2.3.3.1 控制器任务和系统模块配置

当选择“模块视图”栏在 CPU 的模块窗口，你能定义任务，配置 SYSCONF，并将系统模块下载到 CPU。

Modul-Ansicht Datenmodule Kommunikation Gateway-Kopplungen									
Objekt	Version	Zielspeicher	Grösse	Typ	Task/Exc. Nr.	Download	Letzte Änderung	Beschreibung	
[-] CPU									
[-] TK #1									
[-] MO_1_K1	V1.00	RAM	-	Apröl	1	Ja	-	Motorsteuerung	
[-] TK #2									
[-] KL1_K2	V1.00	RAM	-	Apröl	2	Ja	-	Klappensteuerung	
[-] VE_1_K2	V1.00	RAM	-	Apröl	1	Ja	-	Ventilsteuerung	
[-] Systemmodule									
[-] \$\$sysconf	V2.22	USERROM	-	Systemkonfig.		Ja	-		
[-] INAIQDrv	V1.12.1	USERROM	42156	System		Ja	05.10.2001 00:00:00		
[-] brsystem	V1.10.4	USERROM	12060	Bibliothek		Ja	06.12.2001 16:23:16		
[-] burtrap	V1.37.2	USERROM	12164	System		Ja	18.10.2001 01:37:01		
[-] commserv	V2.40	USERROM	27188	System		Ja	20.09.2000 10:30:00		
[-] gcloader	V1.31	USERROM	5816	System		Ja	06.11.2000 23:00:00		
[-] ina2000c	V2.22.4	USERROM	13548	System		Ja	22.05.2000 14:13:05		
[-] inacInt	V1.12.1	USERROM	16936	Bibliothek		Ja	05.10.2001 00:00:00		
[-] rio_lib	V5.12	USERROM	51832	Bibliothek		Ja	27.04.2001 00:00:00		
[-] rtk58k	V1.01.1	USERROM	12540	System		Ja	08.01.2001 13:20:41		
[-] sys_lib	V1.37.2	USERROM	25020	Bibliothek		Ja	28.01.2002 09:39:45		

图 23: CPU 模块- 模块视图和系统模块

在工程设计的最后阶段，你将需要用**物理视图**关联功能图和功能模块。大多数这些功能图和功能块在控制器的 CPU 里运行。单个**任务**以四个任务等级的一种在 CPU 中运行（看 [控制器任务等级](#) 章节）。这里，你能为后面工程设计的功能图定义物理视图。



功能图的终端就是物理视图，它运行在控制器中以一种任务等级。因此，没有物理视图的功能图无法被编译。

如果你选择了 CPU 导入或者有存在任务（例如，TC#3）等级的导入，你可以使用鼠标和简捷菜单添加一个任务（“**创建任务**”）。你能导入打开的窗口。

下表是定义任务的菜单选项的描述。

菜单选项	菜单选项的描述
名称	输入任务的名称。名字对于 CPU 必须独一无二并且应尽可能短。名字定义于关联功能图的物理视图。
资源	当选择了下拉菜单，你能定义执行任务的任务等级。 注意 1: 任务等级的循环时间在 CPU 的 SYSCONF 里面定义。（看 B&R CPU Sysconf 配置 章节）。 注意 2: 相关信息：高速任务执行时，中断任务，和例外任务被使用的情况。请参考相应的控制器的手册。

菜单选项	菜单选项的描述
ID 号	输入（1—255）定义了各种任务等级的任务开始的位置。如果“0”被设以后，系统能选择开始位置。
描述	这个输入是可选的。总体来说，添加描述是需要的。描述使系统更加清楚，特别是以后还要扩展的情况。

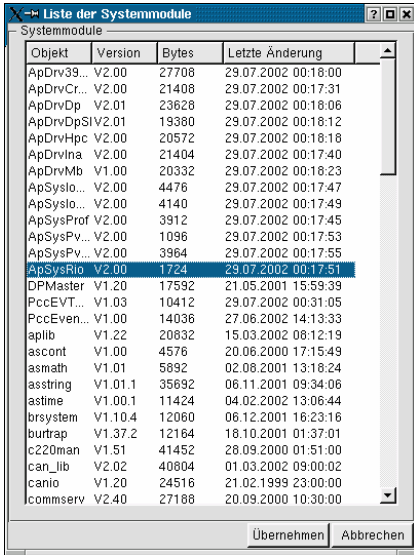
在任务输入后，仍然可以进行配置。你能通过简捷菜单选择任务删除或者关联到不同任务等级。下表中有除了以前已经描述过的菜单选项的“特性”菜单选项。

菜单选项	菜单选项的描述
版本	这里关联版本的编号。
对象内容	选择任务保存的内存区域。
下载	选择 是 或 否 确定任务是否从工程设计系统下载到 CPU 中。
	标记不支持的默认任务。

 生成下载任务。任务从功能图创建以后只有在相应的功能图建立后才能生成。

系统模块

模块视图窗口栏为 CPU 模块提供系统模块，能自动为将来下载可用。这些模块根据项目中所使用的库放到一起。如果你需要附加模块（在大部分例子中），你也可以为它们准备下载。通过简捷菜单“系统模块”来选择你所需要的选项。



Objekt	Version	Bytes	Letzte Änderung
ApDrv39...	V2.00	27708	29.07.2002 00:18:00
ApDrvCr...	V2.00	21408	29.07.2002 00:17:31
ApDrvDp	V2.01	23628	29.07.2002 00:18:06
ApDrvDpSI	V2.01	19380	29.07.2002 00:18:12
ApDrvHpc	V2.00	20572	29.07.2002 00:18:18
ApDrvlna	V2.00	21404	29.07.2002 00:17:40
ApDrvMb	V1.00	20332	29.07.2002 00:18:23
ApSyslo...	V2.00	4476	29.07.2002 00:17:47
ApSyslo...	V2.00	4140	29.07.2002 00:17:49
ApSysProf	V2.00	3912	29.07.2002 00:17:45
ApSysPv...	V2.00	1096	29.07.2002 00:17:53
ApSysPv...	V2.00	3964	29.07.2002 00:17:55
ApSysRio	V2.00	1724	29.07.2002 00:17:51
DPMaster	V1.20	17592	21.05.2001 15:59:39
PccEVT...	V1.03	10412	29.07.2002 00:31:05
PccEven...	V1.00	14036	27.06.2002 14:13:33
aplib	V1.22	20832	15.03.2002 08:12:19
ascont	V1.00	4576	20.06.2000 17:15:49
asmath	V1.01	5892	02.08.2001 13:18:24
asstring	V1.01.1	35892	06.11.2001 09:34:06
astime	V1.00.1	11424	04.02.2002 13:06:44
brsystem	V1.10.4	12060	06.12.2001 16:23:16
burtrap	V1.37.2	12164	18.10.2001 01:37:01
c220man	V1.51	41452	28.09.2000 01:51:00
can_lib	V2.02	40804	01.03.2002 09:00:02
canio	V1.20	24516	21.02.1999 23:00:00
icommsserv	V2.40	27188	20.09.2000 10:30:00

图 24： 系统模块名录窗口

在模块视图里的单个模块的简单菜单，可以从列表中删除模块，定义模块内存，以及决定是否下载该模块。

改变 BR 模块任务等级：

控制器（BR 模块）的驱动提供给一个特定为 B&R 等级的任务。在项目设计中，让你从 BR 模块中改变任务等级可能变的很有必要。

工具 *brpatch* 提供了这个可能性。程序在目录 `/opt/aprol/br/bin` 中。

```
xterm
engin@doku2:/opt/aprol/br/bin>
engin@doku2:/opt/aprol/br/bin> ./brpatch -help
brpatch (B+R Process Patch Utility, Vers. 1.1.0)
APROL, R 2.01 (R), Copyright(c) 1995-2003 B+R Prozessautomation GmbH, Marl
usage: brpatch options

Options                                Purpose

-file BRFILE                          : sets current module to BRFILE
-showInfo                             : shows info about current module
-setTc TC                             : sets module's task class to TC
-userName NAME                        : sets module's modification user to NAME
possible values for TC                :
    1                                 : set to task class 1
    2                                 : set to task class 2
    3                                 : set to task class 3
    4                                 : set to task class 4
NOTE!!!                               : changing TC is only allowed for user tasks

-self                                 : modify application name to origName_<self-option>
-languageCode <AreaCode>             : set language to AreaCode
-ver                                  : print short version information, then exit
                                     <Modul-Vers>,<Prg-Major-VersNr>,<Prg-Minor-VersNr.>
-version                              : print long version information, then exit
-info                                 : print extended version information, then exit
-showlog                             : Start LogViewer with app. log-files
-help, -h, -?                        : show this help, then exit
```

图 25: 调用l brpatch 用-h 选择

选项	描述
-file BRFile	路径和要修补的文件。
-userName UNAME	写 UNAME 到要修补的文件。 否则， brpatch 将被输入
-showInfo	显示所有 BRFile 信息。
-setTc TC	写新任务到 br 文件调整改变的数据和改变的名字（已用），自动调用 -showInfo。 依据 CPU 的类型，你可以选择 1-4 或者 1-8 任务等级。



当用-setTc 时，只有用户任务能被改变，任务等级只有 1-4 可以使用。

系统模块\$\$sysconf

Sysconf (CPU 系统配置)是个特殊的文本，将在 B&R CPU 的Sysconf 配置章节里分开处理。 这里只介绍\$\$sysconf的简捷菜单。

总体上讲，当配置 CPU 的时候一个 *Sysconf* 可以被下载（存储在 caedb 数据库在工程设计系统）。下面是简捷菜单可用的选项：

菜单选项	菜单选项的描述
编辑	在选择了这个选项之后，你能根据系统需要来设置系统配置参数。 例如： 改变任务的循环时间。
导入	如果你在工程设计系统中保存了特定 CPU 的 <i>Sysconf</i> ，你就可以导入它（覆盖以前提供的）。
导出	在定制 <i>Sysconf</i> 后，你可以使用其它的控制器导出它。
特性	在“特性”中，能定义 <i>Sysconf</i> 应该被传输到的目标内存位置，或是否执行全部下载。

2.3.3.2 配置控制器数据模块

在数据模块格中，你能创建自己的数据模块（例如标准驱动的配置）。如果你使用 ASCII 码创建数据模块，它作为 BR 模块下载是不行的。

如果你已经为相关的 CPU 产生了相关的任务，BR 的数据模块也会从 ASCII 码数据模块自动产生，并为下载作好准备。在产生任务后，数据模块在“模块视图”中可以为 BR 模块下载使用。

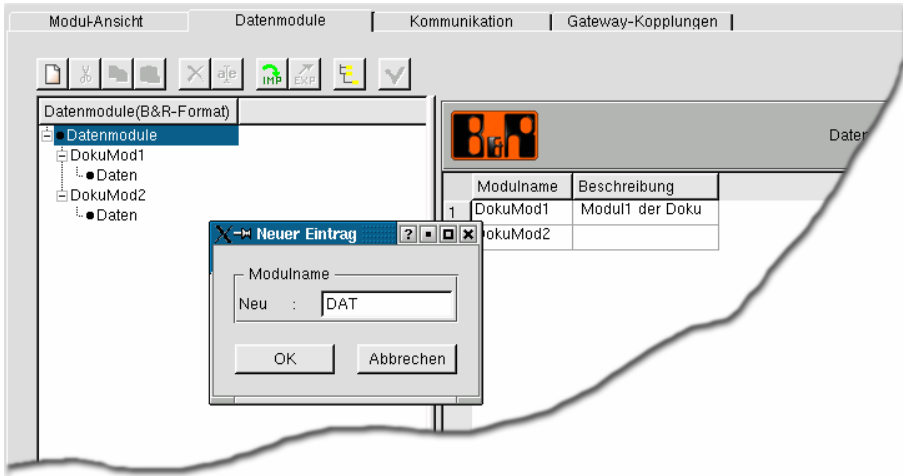


图 26: 插入新的数据模块

你能使用简捷的菜单（鼠标放在“数据模块”输入）或相应的按钮插入新的数据模块。如果一个语法错误产生当关联到名字时候，输入模块名字区域将用红色的边框显示。在下面的例子中，模块已被输入，你能加入一个简短的描述在每个模块名字相应的窗口。

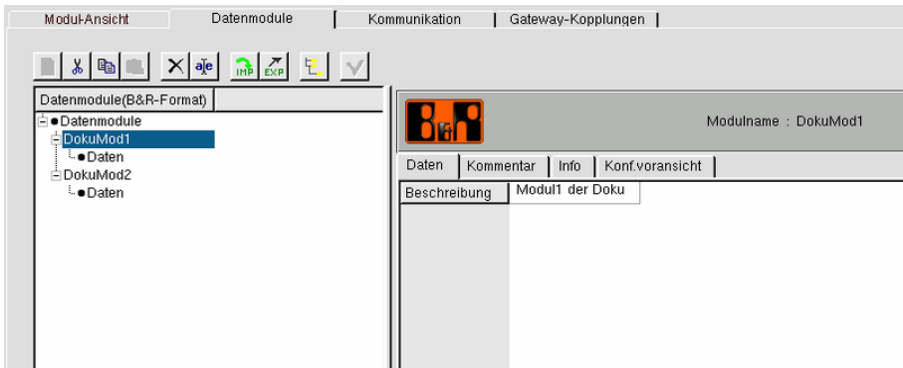


图 27: 编辑数据模块

如果在左边窗口选择数据模块，你能使用简捷菜单或显示按钮如剪切，复制删除和改变名字。你也能从工程设计系统导入或导出数据模块。

当复制数据模块时，将执行语法检查，你将会收到需要改变哪个名字的的建议（输入区域用红色边框）。

右边窗口也有各种可供保存的信息的窗格。

窗格	菜单选项的描述
数据	被选择的包含在内的数据模块的描述，能被改变。
注释	在这个窗格，你能保存注释（描述）关于你的数据模块。
信息	在此不使用。
配置 预览	非编辑的数据模块码视图。

你在左边窗口选择“数据”后，在右边窗口可以编辑数据模块。

2.3.3.3 配置控制器通讯



目前有两种通讯种类在“通讯”窗格里可供使用：CAN 耦合，控制器之间的耦合。

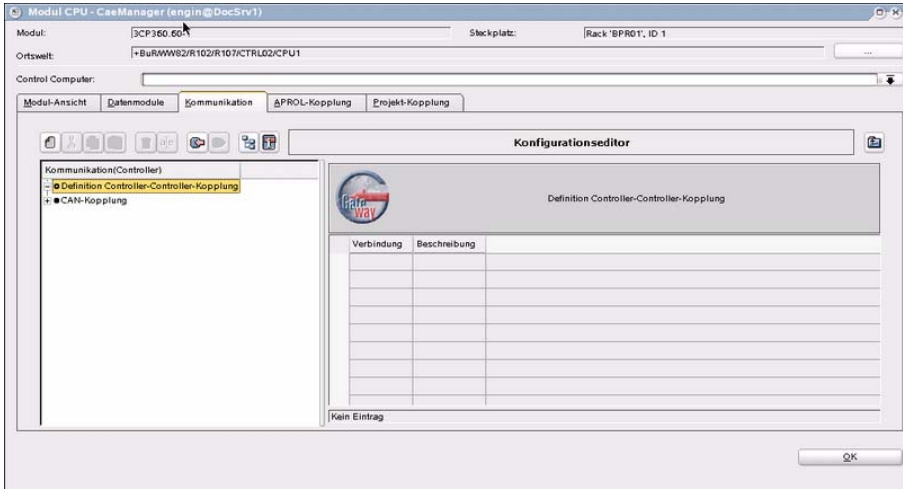


图 28: 控制器通讯

CAN 连接

在 CAN 连接输入下，你能配置几个这种类型的连接。做此时，相应的硬件配置必须连接到这个控制器。



Beschreibung folgt!

Dieses Kapitel ist z. Zt. in Überarbeitung!
Informieren Sie sich bitte in regelmäßigen
Abständen auf unserer Internet-Seite
www.br-automation.com im Bereich
Produktbezogene Downloads über die
aktuelle **APROL** Dokumentation.

定义控制器-控制器的内部连接

在这个定义下，你能准备自动产生直接连接为你目前配置的控制器和搭档控

制器之间的内部连接。两个控制器之间的内部连接要在激活，编译和控制器并发任务产生（包括下载部分）以后建立。



如果你准备好一个控制器之间的内部连接在功能图中。但没有定义控制器之间的内部连接，数据交换将只能通过控制计算机。

以下条件必须满足：

- ✓ 你的工程不只有一个控制器
- ✓ 控制器允许这种通讯水平的连接类型。它们可以通过总线连接或路由读取。
- ✓ 多个连接（软件）被允许。注意：如果连续控制器连接被使用，只有基于软件的连接可用。
- ✓ 在功能图里建立连接。看[自动控制器之间连接的准备](#)章节。

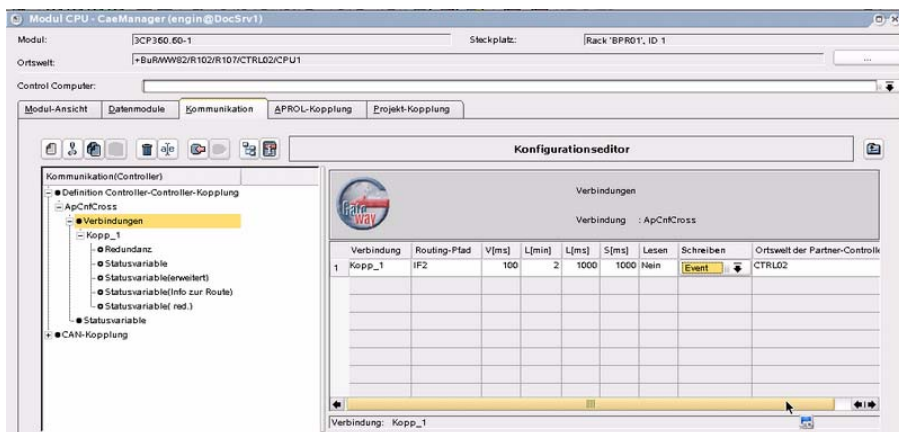


图 29： 控制器之间的内部连接显示例子

在选择定义控制器联接之后，你能定义一个控制器连接在左边窗口，或者右边窗口用简捷菜单（“NEW”菜单选项）。ApCnfCross,只允许在这里输入，它会自动创建。两个附加的输入（连接和变量状态）由ApCnfCross提供。

变量状态输入包括的变量由系统自动产生。

“状态变量”： 全局状态/当前错误

状态变量属于 *STRING* 类型，可采用以下值：

值	描述
"IDLE"	驱动器没有配置
"<ConnName>: 没有连接"	未联接 <ConnName>
"Read error:<INA-errno>"	通信中读操作错误
"Write error:<INA-errno"	通信中写操作错误
"<ConnName>:<PVName>"	<ConnName> 未连接<PVName>
"运行"	驱动器正在运行

如果多处发生错误，则输出第一个错误。

在选择连接之后，你可以通过 window 左边或者右边的快捷菜单（"NEW"菜单项）来定义一个控制器联接”。可以取一个独特的名字来代替连接名字。

一个连接需要定义下表中的输入：

输入	描述
连接	连接名
路由路径	指定搭档控制器硬件的位置 例如： SL0.SS1.IF1.0a/CP 搭档控制器的节点号是 0A（十六进制 10）， 能被连接通过槽 0(SL0)， 子槽 1(SS1)在 CPU 上被配置。 搭档控制器可以使用路由参数 ICP（例如从 IF101 t 到 CP200）。
D [ms] 延迟 [ms]	关系到事件驱动书写。 这涉及到事件从另一个 PV 传输过来等待的延迟。所以所有的时间能被“立即”发送。 值大于 500 会降低网络的负载，但这个要依据现实中的应用。 这个值不能低于变量改变的最小时间。
定义 RT [min] 重复时间[min]	在时间驱动变量能被发送前的累计时间。
R [ms] 读 [ms]	作用与循环读取的输入。 读取变量的值每次循环读取的间隔时间[MS]。
W [ms] 写 [ms]	作用与循环写入的输入。 读取变量的值每次循环写入的间隔时间[MS]。
读	CPU 被配置成循环读取（接受）数据从搭档控制器。 可以禁止读。 定义应用于所有这个连接的 PV。

输入	描述
写	CPU 被配置成循环读取（接受）数据从搭档控制器。写可以被禁止。定义应用于所有这个连接的 PV。
控制器的物理视图	控制器的物理视图 重要信息：只有在 CPU 前输入物理视图才在视图结构可被使用。 例如：PCCDOKU/HG01/R038/N001/RPS01/CPU.

在系统中创建 **状态变量**，**状态变量（扩展）**，**状态变量（info-路由）**和**状态变量（红色）**

“状态变量”： 连接的状态

状态变量是 *INT* 类型时，可采用以下值：

值	意义
0	"IDLE - 正在连接"
1	"INIT - 初始化访问 PVs"
2	"RUNNING – 已连接上"
4	"DISCONNECTED - 未连接上 / 尝试连接"

“状态变量（扩展）”：

状态变量是 *STRING* 类型时，可采用以下值：

值	意义
"IDLE"	正在连接
"INIT"	初始化访问 PV
"RUNNING"	已连接
"PV_ERROR: <PV-Name>"	不能访问 PV
"DISCONNECTED"	未连接上 / 尝试连接

“状态变量（info-路由）”： 路由所有连接

状态变量是 *STRING* 类型时，可采用以下值：

值	意义
----	无连接
<route>	路由所有连接

“状态变量（红色）”：冗余连接的状态

状态变量是 *INT* 类型时，可采用以下值：

值	意义
0	"IDLE – 正在连接"
2	"RUNNING – 已连接上"
4	"DISCONNECTED - 未连接上 / 尝试连接"

关于交叉通讯和控制器与控制器之间的连接的附加信息：

- ✓ 必须为搭档控制器在这里定义一个分开的连接
- ✓ 一个连接**不能**被定义成接收（读）搭档控制器，如果此控制器被功能图设置成只发送（写）。
- ✓ 一个连接**不能**被定义成发送（写）到搭档控制器，如果此控制器被功能图设置成只接收（读）。
- ✓ 两个连接也能为配置的计算机定义。在这个例子中，数据可以双向传输依据功能图中。这样，控制器与控制器之间的连接**不允许**在**搭档计算机**里定义。
- ✓ 如果数据在两个控制器中双向交换数据依据功能图。这时，连接能被定义如下：**连接定义为“写”，“事件驱动”，在每个控制器里。**
此过程只有在数据只被事件驱动交换才有效。

CaeManager 提供给你一个测试的可能：为完全安装和控制器与控制器之间的连接的配置。这个测试只能显示你配置时候的结果在你完成前面的步骤以后：

- ✓ 你已经用连接器创建功能图之间的控制器与控制器之间的连接。
- ✓ 功能图已经被成功的激活和编译。
- ✓ 功能图的任务已经产生。



你已经配置了所有的相应的控制器（CPU）。

在这些步骤完成以后，你能用“其他/静态/交叉通讯”打开自动生成的 HTML 页面（见下图），页面包含了结果。控制器名字的输入也从物理视图中带来。

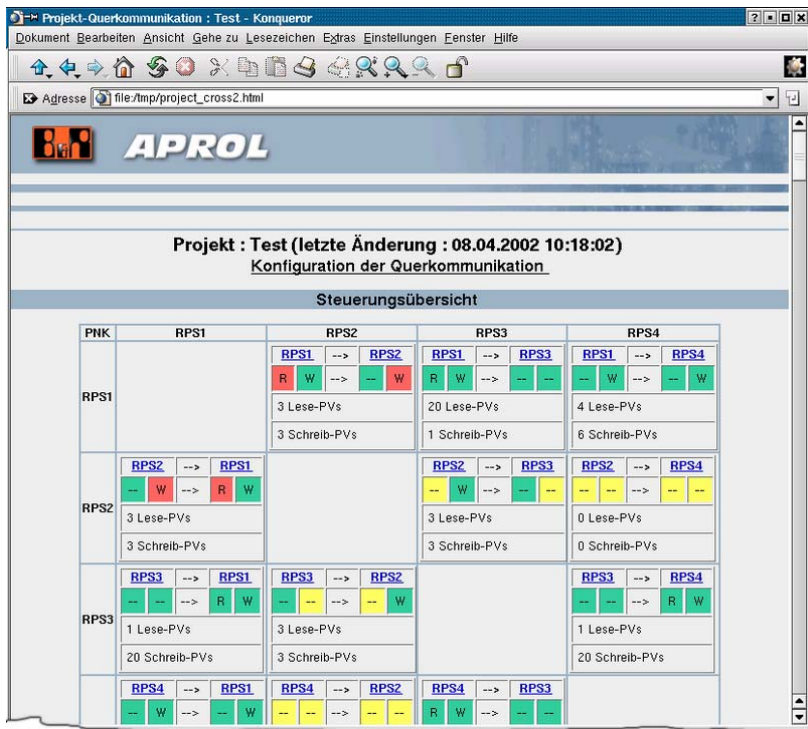


图 30: 检测控制器与控制器之间的已经创建的连接

红色输入显示这是一个错误：：在这个例子中，控制器 1 从控制器 2 读(R = 读)，控制器 2 向控制器 1 写(W = 写)。

输入是绿的说明这个连接配置成功。

黄色说明这个方向没有配置。

2.3.3.4 配置控制器网关连接

在这个窗格里，你能配置一个你现在编辑的 CPU 到一个“外部”的控制器的连接。如果这个连接在你系统运行时候建立起来，数据能从“外部”控制器接受。你能在 **APROL** 过程控制系统里使用，也能发送数据到这个控制器。自由装置的连接是例外的。



网关许可 (G) 需要这些连接。例外的有：由装置连自接和 INA 交叉连接。

控制器通过下列通讯类型交互。相应的硬件以及驱动必须提供：

连接类型	描述
自由装置连接	这种连接类型是例外的。你有选择的权利去创建全局变量给功能图。不过，这些变量应该只能在关联 CPU 物理视图的功能图里使用。
INA 交叉连接	这个通讯类型对比于控制器与控制器的内部联系在通讯窗格下。在这个例子里，一个自动配置将不会完成。你必须定义你想发送或者接受每个个体数字点（变量）。
3964R 连接	使用 3964R 驱动的连接。 需要网关许可！
DP 连接	需要网关许可！
DP 从站连接	需要网关许可！
FMS 连接	需要网关许可！
Modbus 连接	需要网关许可！
HPC 连接	需要网关许可！

当然，你希望使用，你就得配置这些连接。你能保存在注释输入里显示的每个配置的具体描述。如果你选择的配置类型被许可，你能看到在配置窗格里的配置结果。

自由装置连接

这个连接非常相似于网关连接，也可以由一个网关计算机产生。（在控制计算机的网关全局变量里看连接的建立）。

在许多例子里，全局 PV 将必须使用在这个控制器的许多的功能图里。PV 值在这个功能图里决定和计算。这个值在更进一步的过程中的其他功能图中也需要。在这个例子里，你能用连接器工作。在这里例子里，一种更高级的可能性，可通过网关创造全局变量，并且使用这些。优点是 PV 在中心位置进行定义和描述。

网关的 PV 只能在此控制器的物理视图相关联的功能图里接受。I/O 数据指向相应的真实控制器。默认的，功能图的输出栏里能创建输出。只在输出定义里允许这些全局变量。

INA 交叉连接

这种连接类型在配置控制器直接与 B&R2000 系列搭档控制器通讯时使用。在这个例子，搭档控制器不能通过 **APROL** 编程，代替的是“外部”编程。这种连接近似于控制器与控制器之间的内部连接，不过每个单独变量值改变必须在这里定义。

模块需要需要控制器配置成自动下载准备。为这种通讯类型的模块必须在搭档控制器里安装。哪一个模块需要在搭档控制器里在使用这种通讯类型里，可以在 **APROL** 系统手册 控制器的交叉连接 章节里查找。在搭档控制器里，PV 不能被配置。

一个连接通过外部程序搭档控制器建立，使用名字和连接参数和输入变量的数据一样应该被传输。

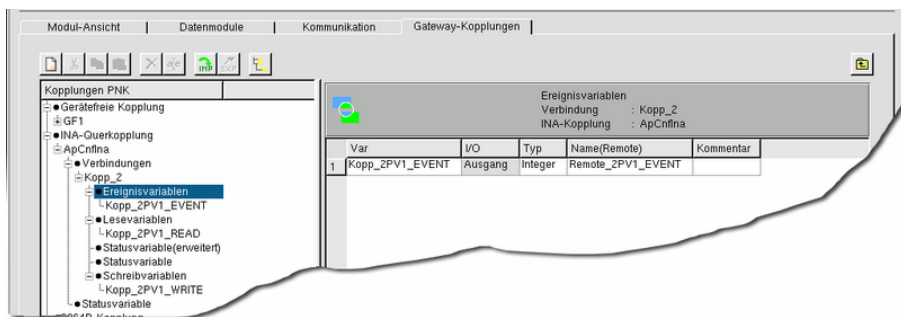


图 31: INA 交叉连接的视图

下列必须在建立连接时输入：

输入	描述
Conn	连接名
路由路径	指定搭档控制器硬件的位置 例如： SL0.SS1.IF1.0a/CP 搭档控制器的节点号是 0A（十六进制 10），能被连接通过槽 0(SL0)，子槽 1(SS1)在 CPU 上被配置。 搭档控制器可以使用路由参数 ICP（例如从 IF101 t 到 CP200）。
D [ms] 延迟 [ms]	关系到事件驱动书写。 指定的延迟传输时间后[ms]， 多个 PV 的值才写入这个值。 这涉及到事件从另一个 PV 传输过来等待的延迟。所以所有的时间能被“立即”发送。 值大于 500 会降低网络的负载，但这个要依据现实中的应用。 这个值不能低于变量改变的最小时间。
定义 RT [min] 重复时间[min]	在时间驱动变量能被发送前的累计时间。
R [ms] 读 [ms]	作用与循环读取的输入。 读取变量的值每次循环读取的间隔时间 [MS]。
W [ms] 写 [ms]	作用与循环写入的输入。 读取变量的值每次循环写入的间隔时间 [MS]。

变量在三组中输入（看下表）。 输入每个单个变量的名字在搭档控制器（*远*程）和在 **APROL** 系统里的名字。 变量类型必须被选择。 变量方向（*输出和输入类型*）决定于你在哪个组里输入。

变量	描述
事件变量	变量的值只在值改变以后发送给搭档控制器。
读取变量	变量的值由搭档的控制器循环读取。
写入变量	变量的值由搭档的控制器循环写入。

B2-2.4 配置CPU属性

所有的中央处理器（CPU）在 B&R 控制器里有一个自己的系统配置 (Sysconf)，使用在配置控制器的操作系统以适合于应用的需要。一个 Sysconf 能被使用于一般的应用于每个 CPU 的数据库。你能打开 CPU 模块的配置窗口菜单，通过双击 Sysconfzai 模块视图窗格（看控制器任务和系统模块的配置章节），你能改变标准的输入在你需要的单个输入窗口的位置。



在模块视图里，到系统模块，标记 Sysconf，使用简捷菜单去打开单个的编辑选项。

项目创建， Rel. < 1.0	当编辑项目创建的时候 Rel. < 1.0（放置 CPU）。不能使用 CaeLoader 装载 Sysconf。Sysconf 的数据不能被装载在这些项目里。 项目里的 Sysconf： 首先，上载项目 Sysconf 从控制器到一个工程设计系统的路径通过使用相关的程序（例如：InaConnect 或者 ControllerManager）。从这个路径，导入 Sysconf 进入项目[Sysconf imp.]。你能编辑 Sysconf。这个 Sysconf 能使用 CaeLoader 装载到控制器里，成为项目的一部分。
项目创建， Rel. 1.0 或更高	装载你需要编辑的控制器的 Sysconf，通过选择[配置]按钮在 CaeLoader 中应用。 编辑控制器中已存在的 Sysconf: 导入按照上面描述的 Sysconf（Rel. < 1.0）。保持你已经编辑的项目的 Sysconf，在你作出改变以后使用 CaeLoader 装载到控制器。

2.4.1 配置SG3 对象的Sysconf

在你开始编辑 Sysconf 以后，编辑 Sysconf 的程序将被装载，菜单选项的窗口也被打开。当窗口打开以后，下面的一些基本信息将被显示：CPU 的类型，操作系统版本，以前定义的 CPU 名。能被使用的名字（最大 8 个字符）。

单个配置选项能在几个窗格里选择。箭头记号可以在菜单选项窗口的许多窗格里显示。

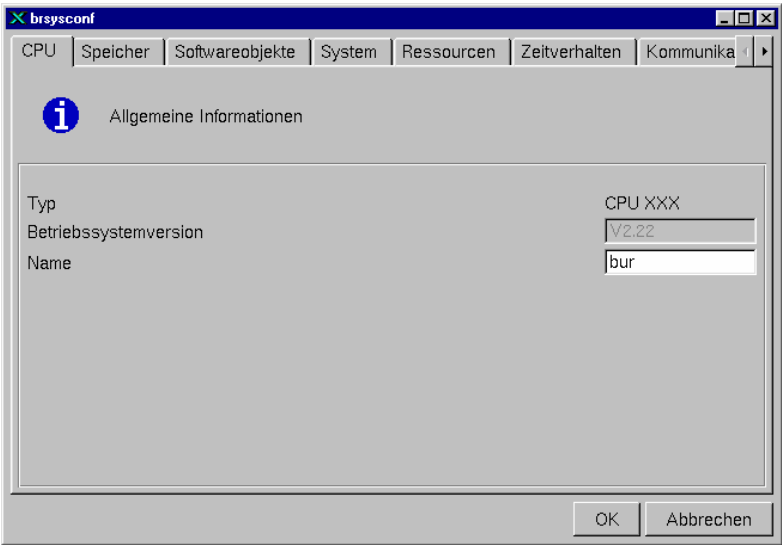


图 32: Sysconf 开始菜单选项窗口

如果你在“设计者”窗口没有激活专家模式（“其他选项”菜单选项），但是你能改变Sysconf,相应的一些改变将不能生效。

内存窗格:

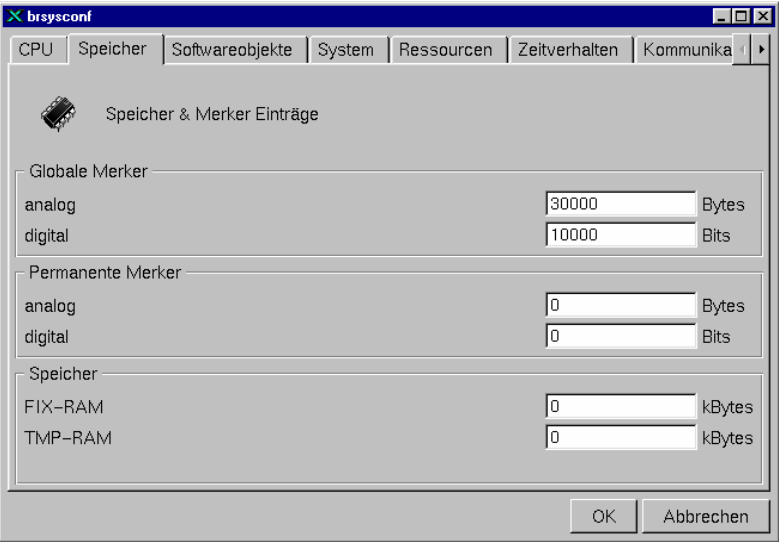


图 33: 内存划分

菜单选项	解释:
全局模拟变量	决定全局模拟数据区域。这个数据区域是在双口 RAM 部分区域。在这个区域所有的全局变量不是这种类型 (=BOOL)，是可以保存的。如果安装任务时，出现 8012 的错误说明这个区域太小。数据区域的尺寸在重新安装任务之后必须增长。目前的尺寸只应该在例外事例中改变。
全局数字变量	决定全局数字数据区域。在这个数据区域（双口 RAM），所有的全局变量的“bit”型是可以保存的。如果安装任务时，出现 8013 的错误说明这个区域太小。数据区域的尺寸在重新安装任务之后必须增长。目前的尺寸只应该在例外事例中改变。
永久模拟变量	决定非 volatile 数据区域“byte”类型的大小。到这时，使用 CaeManager 不能在这个区域创建数据。
永久数字变量	决定非 volatile 数据区域“byte”类型的大小。到这时，使用 CaeManager 不能在这个区域创建数据。
FIX RAM 内存	决定了 RAM 的大小。只有输入 0 或者多重 16 可被接受。内存区域象 User ROM 一样动作，在全部初始化后。模块在此保存后，将不被删除。
TMP-RAM 内存	决定了临时内存的大小。只有输入 0 或者多重 4 可被接受。

软件对象窗格:

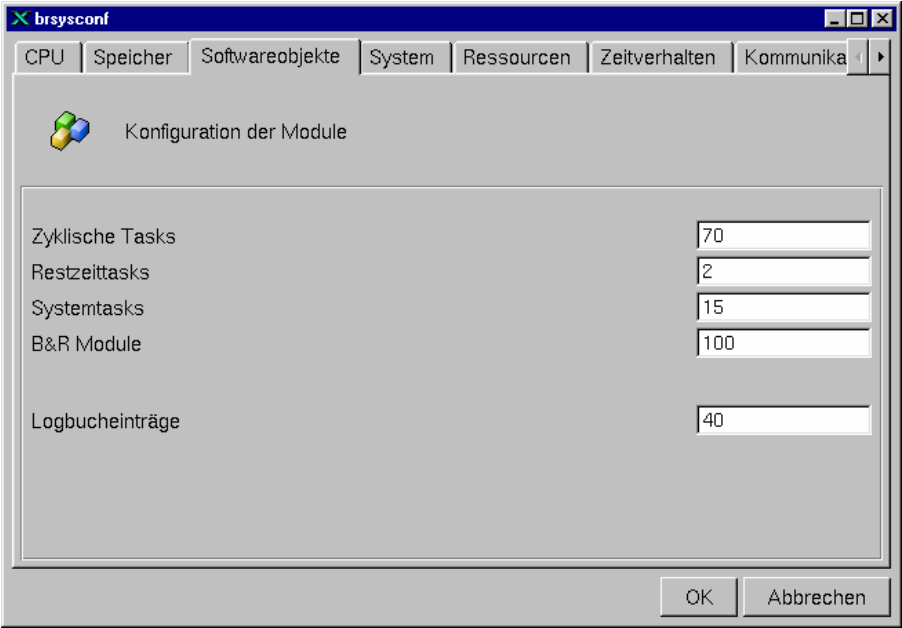


图 34: 配置任务编号

菜单选项	解释:
循环任务	输入你能安装在控制器上的循环任务的最大数目（循环，定时器，中断和例外任务）。 如果你安装了更多的任务，安装操作将会被错误信息中断。
空闲时间任务	输入你能安装在控制器上的非循环任务的最大数目（空闲时间任务）。 如果你尝试于安装另外的任务，安装程序将会中止，错误信息将会显示。
系统任务	输入你能安装在控制器上的非循环系统对象和模块的最大数目。
B&R 模块	输入你能安装在控制器所有 B&R 对象的最大数目（软件对象，除了操作站，在功能库里运行时对象等等）。 你能使用下面公式计算最小值： 数量= 循环任务数+ 空闲时间任务数 + 系统任务数 + 10
日志输入	输入日志输入的最大数目。如果你输入 20 值在这里，最后 20 个错误将会在任何时刻记录在数据记录中。 旧的输入会被覆盖 错误数据输入数目应该在 16 到 500。

系统窗格:

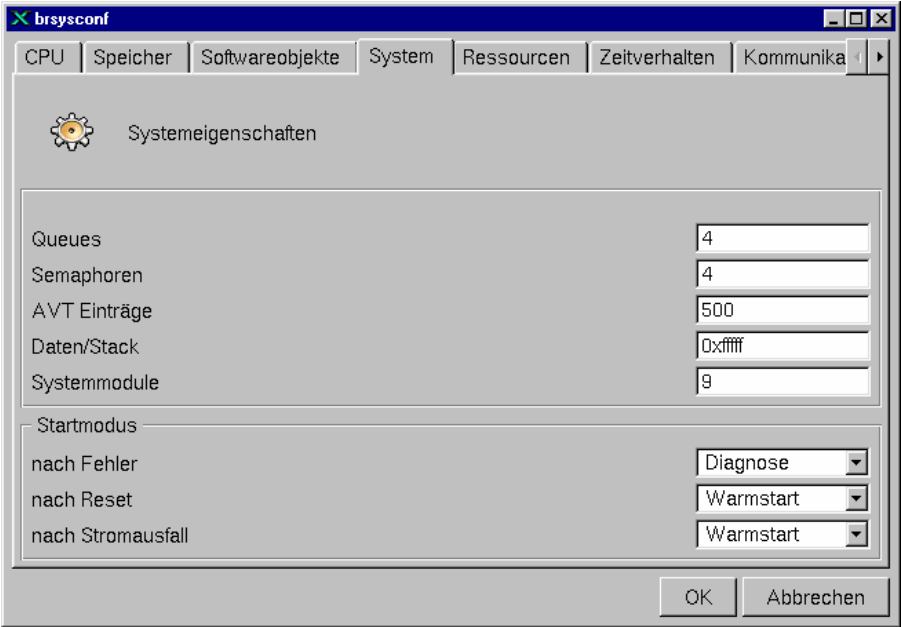


图 35: 系统设定

菜单选项	解释:
队列	输入队列数目 (最小值: 4). 它们使用于任务之间的通讯。
信号	输入操作系统的信号数目 (最小值: 4). 这些变量将用于普通内存区域的信号读取。
ADT 输入	输入到 ADT (地址分配表)中的可能数目。
数据 / 堆栈	输入系统 RAM 用于储存操作系统数据部分的大小, 和全局数据以及空闲时间堆栈的应用一样。
系统模块	输入系统模块在架子上的最大数目。 这里包括远程 I/O 主站, MP100, IF050, 等等。
错误后的开始模式	在严重的启动错误事件中, 控制器将开始在默认的诊断模式。下次诊断模式的启动, 控制器只能开始于冷启动。 这意味着冷启动被初始化到应用运行之中。 为了避免这种情况, 你能设置严重错误后的启动模式 (诊断模式或者冷启动)。 “冷启动 “设置在一次错误以后将触发冷启动, 应用将自动开始。

菜单选项	解释:
测试后的开始模式	这里你可以选择热启动或者冷启动开始
断电后的开始模式	这里你可以选择热启动或者冷启动开始

资源格:

在这个菜单里设置任务等级。 高速任务，比如中断任务和例外任务级别，在这里**APROL**不支持。

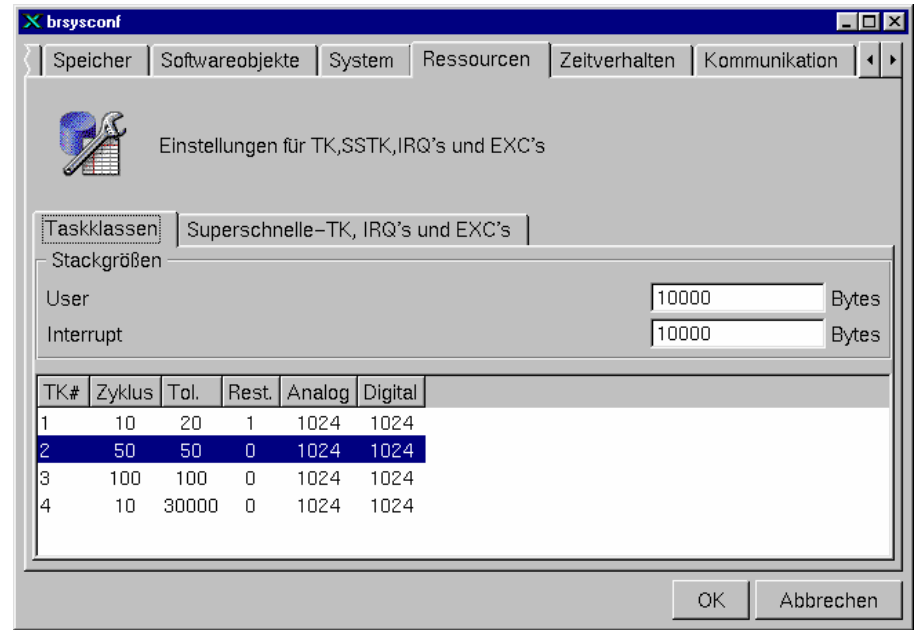


图 36: 配置资源

用户”和“中断”堆栈大小将影响到所有 4 个任务的登记！

菜单选项	解释:
用户	输入循环任务等级用户堆栈的大小。用户堆栈在系统的 RAM 中使用一块数据存储区域。用户堆栈的大小依据你在任务等级里调用的功能图的需要。如果用户堆栈大于输入的尺寸，操作系统能识别堆栈过载，CPU 将开始在错误的模式下。

菜单选项	解释:
中断	输入循环任务等级中断堆栈的大小。 用户堆栈在系统的 RAM 中使用一块数据存储区域。 中断堆栈的大小依据你在任务等级里运行的功能图的需要。 如果用户堆栈大于输入的尺寸, 操作系统能识别堆栈过载, CPU 将开始在错误的模式下。

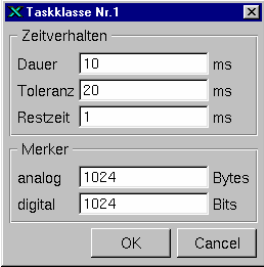


图 37: 任务等级的配置

左边的图显示设置必须为每个任务等级改变（双击在所选的任务等级）。 内部变量的区域通过 **APROL** 产生, 所以你不必在这里设置。

中断任务等级和例外任务等级信息:

中断任务等级中的任务不是时间控制; 相反, 当某个特定任务发生时, 它们才执行。 中断只能被特殊的硬件产生。 相反, 当某个特定任务发生时, 它们才执行。 中断只能被特殊的硬件产生。 一个中断模块, 例如, 数字输入模块 DI400。 8 位数字输入能使用在 CSI 输入时间 (CSI= “状态变化中断”) 卡上。 如果 CSI 输入状态改变, 在 CPU 上中断被触发。

例外任务等级中的任务不是时间控制; 相反, 当某个特定任务发生时, 它们才执行。 例外是一种信号, 来自于系统显示某种例外状态。 当系统显示重要错误时, 它们能触发中断。 例外任务评价这种信号, 并采取必要措施纠正错误。 例外任务等级有最高的优先级。 当一个例外发生时, 将伴随着例外任务中断, 以正常和中断任务。

时间格:

在这个菜单, 你能定义控制器优先开始任务的延迟。

延迟	解释:
冷启动后	冷启动后任务开始前的系统等待时间长度。
热启动后	热启动后任务开始前的系统等待时间长度。

延迟	解释:
电源关闭信息前的延迟	电源关闭就象热启动。总之应用程序会自动开始（再次切换以后）。在一个有 I/O 总线扩展或远程 I/O 总线，在网络组件的不同负载中可以导致电源切断一些连接到主 CPU 的系统组件。 如果主 CPU 检测这些系统组件的失误，在电源断电以前，一个重要错误就会报告。 在这个例子中，对象系统开始于错误模式，当电源重连接和激活了服务模式。这意味控制器是不激活的，必须手动地开始（意味着热启动）。 为了避免在电源失效以后，控制器的外围设备在服务模式，你能输入一个延迟时间给触发一个严重错误。这意味着如果一个严重错误被触发，系统等待到延迟时间来决定是否因为电源失效产生错误。如果电源供给主 CPU 在延迟时间之中失效，严重错误不会触发，控制器被热启动，在切换之后，控制器自动启动。

容许的系统循环时间违例	当对应于循环资源的循环时间，操作系统的系统循环时间被监控（10ms 循环）。为了避免系统循环时间违例被立即报告在一个短暂的时间瓶颈内（例如定时器中断对象在整个系统时间里使用），你能设置一个最大的系统容忍时间违例以 10MS 每步。如果时间超过，操作系统触发例外“系统循环时间违例”（例外号：162）。
-------------	---

通讯格:



Kommunikation

Turbo-Modus für Kommunikator

ja

Kommunikationskanäle

7

Parallele Verbindungen

3

Gerätetreiber

15

PV-Tabellen

77

Logische Variablenlisten

10

Physikalische Variablenlisten

10

Force-Aufträge

10

OK

Abbrechen

图 38：配置通讯数据

菜单选项	解释:
通讯器的 Turbo 模式	V2.20 或者更高。 当使能时, INA 服务也可以手动在空闲时间。 那意味着更快地任务转换, 更快的在线通讯, 等等。
通讯通道	输入同时使用的最大交互界面数目。 这个值只相关于不使用的标准的网络配置模块。
平行连接	
设备驱动	输入在这个控制器上最大数目的驱动模块。 如果你试图安装更多的驱动模块, 将报错。
PV 表	输入变量表的最大数目。 <i>APROL</i> 不使用。
逻辑变量列表	<i>APROL</i> 不使用。
物理变量列表	<i>APROL</i> 不使用。
强制任务	<i>APROL</i> 不使用。

交互格式:

在线界面和控制器中的调制解调器能在这个窗口中定义。 如果新的界面没有定义, 两个窗口和一个 CAN 的界面被事先设定。 这个默认的设置无效如果系统配置里有一个或多个界面被重复设置。 高于 2.00 的版本, 可以有多个界面能被在线配置。

使用[新界面]或[新调制解调器]按钮, 你能决定你想安装或配置什么。 实际中的配置在一个新的窗口里产生。 你双击在新的输入里可弹出新的窗口。 下图显示了两个配置窗口。

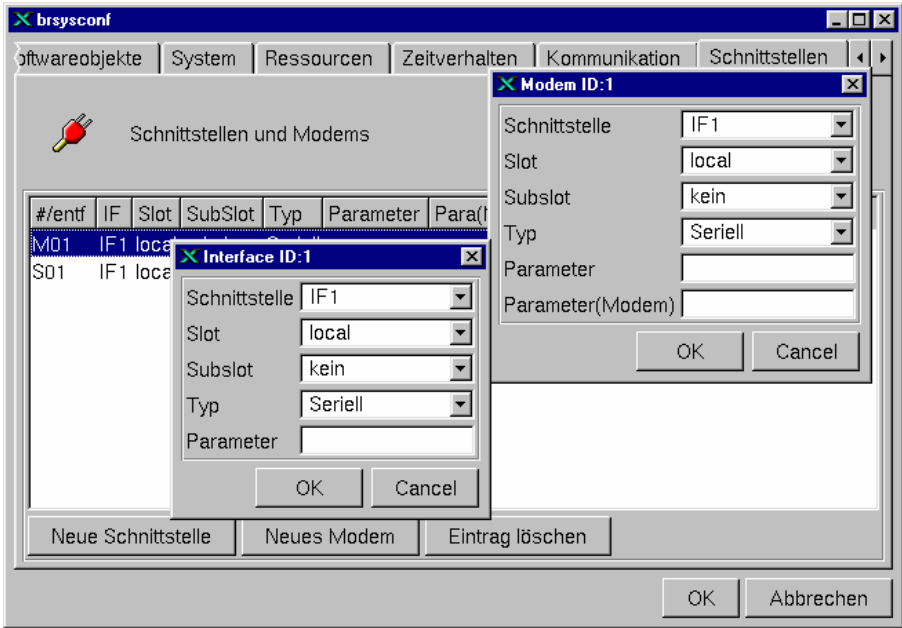


图 39: 配置界面

菜单选项	解释:
界面	定义了在线界面的序号 (IF1 - IF5)。
插槽	在此，你能指定槽给特定的模块（在线界面）。 本地: 处理器模块上的界面。 SL 1 至 SL 13: 设置 SL1 至 SL13 确定的界面模块地址(例如, "SL1" = 模块地址 s 1)。
子槽	如果模块界面插入一个插入的槽（子槽）中，子槽的序号（ SS1 , SS2 或 SS3 ）必须被指定。这个设置只对 CP260 和 IF260 模块必要，因为目前只有这些模块有插入槽。其他所有的模块此设置都没有用。
类型	定义界面的类型 (串口, , CAN, PP, PB 或 以太网)。高于 2.00 的版本的操作系统， 串口, , CAN,和 PROFIBUS 都可以使用在在线交互界面上。
参数	
参数（调制解调器）	

RIO 窗格:

在这个格子，你能配置操作系统为远程 I/O 连接控制器。一个远程 I/O 主站是预先配置。实际的配置是在一个新的窗口（双击输入）。

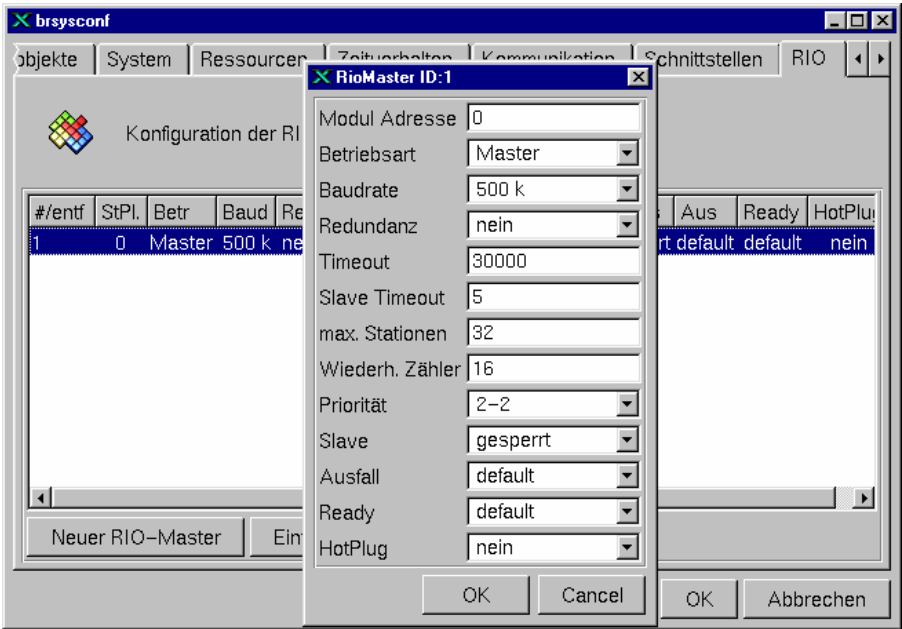


图 40: 为远程 I/O 配置 OS

菜单选项	解释:
模块地址	
操作模式	
波特率	
冗余	
超时	
从站超时	
站点最大序号	
计数器	
优先级	
从站	

菜单选项	解释:
失效	
准备	
热插拔	

CAN 窗格:

给控制器的操作系统配置 CAN 总线。主站的实际配置和从站的实际配置在一个新的窗口里实现（双击在输入处）。

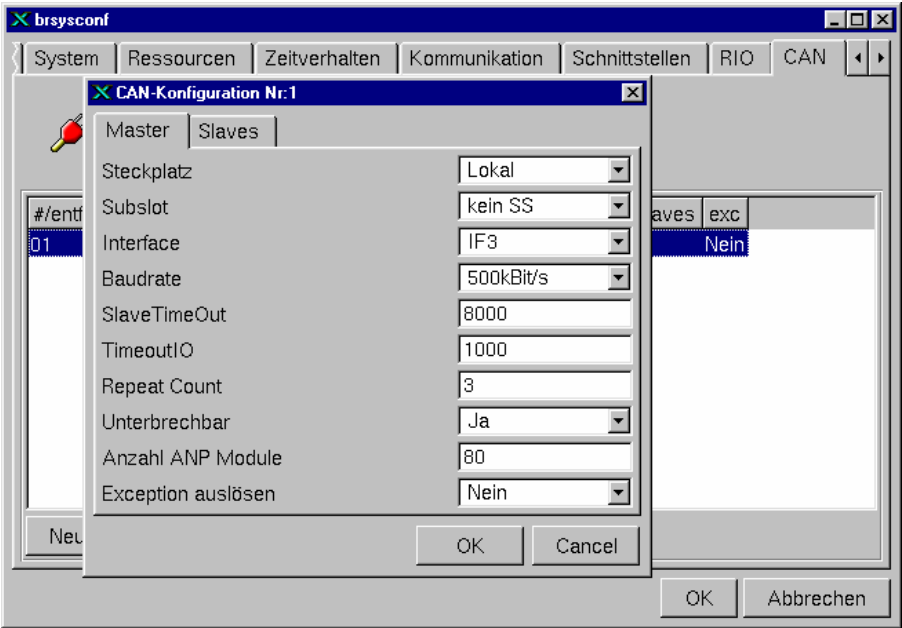


图 41: 配置 CAN 总线

菜单选项	解释:
插槽	
子槽	
界面	
波特率	
从站超时	

菜单选项	解释:
从站超时	
I/O 超时	
重新计数	
中断性能	
增加的模块序号	
例外触发	

Profibus 窗格:

在此窗格无需改变。这个菜单只供 B&R 服务工程师使用。

2.4.2 配置SG4 对象的Sysconf

请注意:

这个部分（配置 SG4 对象的 Sysconf）已经在 B&R 文档“AR SYSCONF XML 配置选择”中介绍，版本 1.0.0,发布：2000 年 12 月。

2.4.2.1 其他

在 SYSCONF XML 部分，Automation Runtime 的配置显示在树形结构里。用 ROOT 开始，有多级并且每级都有多个分支。每个分支必须一个关键的名字。一般来说，实际的数据片是单个输入，每一个数据片有一个名字，数据类型和数据本身。

下表是可用的数据类型：

数据类型	描述
ui4	无符号整型 4 字节 (32-bit)
字符串	空格结束的 ASCII 字符串
布尔	布尔值 (0 或 1)
十六进制	十六进制字节区域 (例如. 12ABCDEF34)
字符	

ROOT:

不同的 AR 配置区域输入在 ROOT 输入 XML 部分 (下列区域是目前定义的):

- ✓ DEVICES
- ✓ INADEVICES
- ✓ CANIOMASTER
- ✓ ETHERNET
- ✓ TASKS
- ✓ IDLETIME

这些名字是固定的关键名字。但不是所有的区域都被使用。不过，一些区域是强制的，因为 AR 需要一些区域启动（寻找可以储存的模块安装）或者 INA2000 通讯。

2.4.2.2 系统表



Beschreibung folgt!

Dieses Kapitel ist z. Zt. in Überarbeitung!
Informieren Sie sich bitte in regelmäßigen
Abständen auf unserer Internet-Seite
www.br-automation.com im Bereich
Produktbezogene Downloads über die
aktuelle **APROL** Dokumentation.

2.4.2.3 启动属性



Beschreibung folgt!

Dieses Kapitel ist z. Zt. in Überarbeitung!
Informieren Sie sich bitte in regelmäßigen
Abständen auf unserer Internet-Seite
www.br-automation.com im Bereich
Produktbezogene Downloads über die
aktuelle **APROL** Dokumentation.

2.4.2.4 全局PV



Beschreibung folgt!

Dieses Kapitel ist z. Zt. in Überarbeitung!
Informieren Sie sich bitte in regelmäßigen
Abständen auf unserer Internet-Seite
www.br-automation.com im Bereich
Produktbezogene Downloads über die
aktuelle **APROL** Dokumentation.

2.4.2.5 循环系统

循环系统的参数配置在 CYCLICSYSTEM 配置区域里。系统时间刻度决定了最小可能循环任务时间。（注意：循环时间和循环任务的兼容时间不在目前这里配置—但应该在晚些时间在这里做。）**所有的时间设定（也在 TC 循环时间）必须是整个系统多重任务在一个系统的时间刻度。**

在 IDLETIME 配置子区域进行任务等级的空闲时间配置。例如，所有型号的 INA 通讯都需要这个。设置 IdleTime 值为 0 可以解除激活任务等级的空闲时间。

任务等级空闲时间的作用 在循环时间结束后，所有任务等级的循环工作时间大于或等于循环时间。

数据名称	数据类型	描述
系统时间刻度	ui4	AR 系统时间刻度在微秒级（默认：1000 us）
关键名称		
空闲时间		固定名称
数据名称	数据类型	描述
循环时间	ui4	任务等级的循环时间在微秒级（默认：10000 us）。
空闲时间	ui4	任务等级的空闲时间在微秒级（默认：1000 us）。

此区域可**任选**。如果不可选或者这部分不能选。那么使用默认值。

2.4.2.6 任务

在任务的配置区域，所有 AR 任务的参数被设置。在 AR 基本系统任务也被装载。这样的话，是可能的，例如，配置关联任务的优先极到客户当前客户项目需要。

每个“任务”的输入必须有相应的任务名称作为线索，任务特性在数据输入。

关键名称		描述
任务名称		在 Automation Runtime 中的任务名称
数据名称	数据类型	描述
优先级	ui4	在 Automation Runtime 中的优先级(0-255)

这个区域是**强制的**，所有的 AR 任务必须输入。如果这个区域不可用，那么所有的 AR 任务都开始于最小的优先级，AR 的功能也无法保证。

在下列任务中，优先级也需要调整到适应项目的需要：

- **tc#1, tc#2, tc#3, etc.**

在任务等级 1, 2, 3 等，优先级可能需要根据循环时间和运行时间改变。



1. 任务等级的优先级经常不得不低于“IO 调度程序”。

2. 因为 AR105 配置一个 IPC5000 Celeron 和 INTEL 以太网控制器，, VxWorks 的"tNetTask" 和 "tFtpdTask" 任务最大运行时间超过 100msec。所以，只有任务等级有很高的优先级的容忍时间应该用于这些任务 (例如. tc#4 以 10 msec 循环时间,但至少还有 500 msec 的兼容时间)。

2.4.3 Can2DD任务, 182527

CAN 通讯

- DdItShRx, DdItShTx

AR010 共享通讯

- FBSF_EventHandler, SerFrmReceive, SerFrmTransmit

串口通讯

- FBEthReceive, FBEthSend

通过 INA2000 的以太网通讯
- canio

CANIO 处理
- arou, syssserv, cs_brmod, dbtracer, applader, InaEvChk

AR 内部 INA2000 通讯任务的通讯 n
- tNetTask, tFtpdTask

VxWorks 内部任务优先级不能被改变



重要: 因为 AR105 配置一个 IPC5000 Celeron 和 INTEL 以太网控制器, VxWorks 的 "tNetTask" and "tFtpdTask" 任务 最大运行时间超过 100 msec. 所以, 只有任务等级有很高的优先级的容忍时间应该用于这些任务 (例如. tc#4 以 10 msec 循环时间,但至少还有 500 msec 的兼容时间)。

2.4.3.1 设备

Powerlink 模块的例子:

这里, 一个 IF786 PL 模块应用于 CP382 SL0,SS3。
根据所使用的插槽输入**物理设备名称**和**设备参数**。

设备				
	SS3.IF2			
		物理设备名称	string	PCI.3.IF786.IF2
		驱动器名称	string	DdVxIoTimerPL
		驱动器类型	ui4	11
		开始模式	ui4	1
		参数设置	string	/CYCLE=1000 /PRESCALE=1 /SL=0 /SS=3 /IF=2

关键名称		描述
<任何设备名称>		任何长度在 32 字符内的 ASCII 字符串 (以 \0 结束)
数据名称	数据类型	描述
驱动器名称	string	设备驱动名称
驱动器类型	ui4	设备驱动名称 DM_DDTYPE_INVALID (0) DM_DDTYPE_SERIAL_FRAME (1) DM_DDTYPE_CAN (3) DM_DDTYPE_FILE (5) DM_DDTYPE_NON_VOLATILE (6) DM_DDTYPE_ISAIO (9) DM_DDTYPE_CPUHW (10) DM_DDTYPE_IOTIM (11)
设备参数	string	驱动类型或驱动依据的参数 DM_DDTYPE_INVALID (0) DM_DDTYPE_SERIAL_FRAME (1) DM_DDTYPE_CAN (3) DM_DDTYPE_FILE (5) DM_DDTYPE_NON_VOLATILE (6) DM_DDTYPE_ISAIO (9) DM_DDTYPE_CPUHW (10) DM_DDTYPE_IOTIM (11)
设备依赖	string	另一个设备的依赖性(设备层叠); 依赖于这个设备的设备名称
开始模式	ui4	设备开始模式 (0=disabled, 1=enabled)
物理设备名称	string	物理设备名称

这个区域是强制的。例如，所有的给串口通讯或 CA 通讯的通讯设备必须输入。

“设备参数” 值根据设置类型而改变。

设备参数 DM_DDTYPE_SERIAL_FRAME (1)

标签	值/值域	默认值	例子	描述
/IO 或 /BA 或 /NAME	0-0xffff 0-0xffffffff <Host-OS-Device- Name>	0 0 '\0'	/IO=0x3F8 /BA=0xD000 /Name=COM2	IO 口地址 给 UARTS 基地址给 UARTS 设备地址在主机 OS
/IR	0-255	-1	/IR=4	中断数
/CLK	0-0xffffffff	0	/CLK=1843200	UART 时钟频率[Hz]
/BD	300-115200	57600	/BD=57600	波特率
/PA	N E O H L	N	/PA=N	奇偶校验
/DB	7 8	8	/DB=8	数据位
/SB	1 2	1	/SB=1	停止位
/PHY	RS232 RS422 RS4 85	RS232	/PHY=RS422	物理特性
/RIT	1-65535	4	/RIT=4	接收空闲时间
/TIT	1-65535	0	/TIT=5	传输空闲时间
/RREQ	0-255	8	/RREQ=2	在设备读取需要的最大 数字
/WREQ	0-255	1	/WREQ=1	设备写入需要的最大 数字
/CCHAR R1	Code of every single ASCII character	No character s	/CCHAR1=0x0 a	控制字符 1 (框架和 字符)
/CCHAR R2	-	-	-	控制字符 2
/CCHAR R3	-	-	-	控制字符 3
/CCHAR R4	-	-	-	控制字符 4
/FCCT S	0 1 (FALSE/TRUE)	0	/FCCTS=1	CTS 输出流控制 (输出 方向的硬件握手)
/RTSTX	0 1 (FALSE/TRUE)	0	/RTSTX=1	TX 框架-信号的框架

设备参数 DM_DDTYPE_CAN (3)

AR10x: DdVxCnLs172 - LS172 设备驱动参数和

AR010: DdItCnLs172 - LS172 设备驱动参数

标签	值/值域	默认值	需求	例子	描述
/BA	0-0xffff	0	Yes	/BA=0x250	I/O 地址
/IR	0-255	-1	Yes	/IR=5	中断数
/IF	1-2		Yes	/IF=1	附加设备信息
/BD	10000 20000 50000 100000 125000 250000 500000 1000000	250000	No	/BD=250000	波特率[kBit/s]
/BT	0-1	0	No	/BT=0	总线类型 (0=PCI,1=ISA)
/EX	0-1		Yes	/EX=0	扩展 CAN 校验 0=11Bit 1=29Bit
/HWM	0-0xffff	1	No	/FS=1	堆栈大小 LS172 (H8)
/RT	0-0xffff	10	No	/RT=10	接受超时 [ms] LS172 (H8)

AR10x: DdVxCnI82527 - Onboard 82527 设备驱动参数

标签	值/值域	默认值	需求	例子	描述
/BA	0-0xffff	0	Yes	/BA=0x250	I/O 地址
/IR	0-255	-1	Yes	/IR=5	中断数
/BD	10000 20000 50000 100000 125000 250000 500000 1000000	500000	No	/BD=250000	波特率[kBit/s]
/EX	0-1		Yes	/EX=0	扩展 CAN 校验 0=11Bit 1=29Bit

标签	值/值域	默认值	需求	例子	描述
/FS	0-0xffff	64	No	/FS=64	堆栈大小(传输/接收)

设备参数 DM_DDTYPE_FILE (5)

标签	值/值域	默认值	例子	描述
/DEVICE	<DeviceBaseDirectory>	-	/DEVICE=C:\Data	主机文件系统中文件的基目录

设备参数 DM_DDTYPE_NON_VOLATIL (6)

设备参数根据所使用的设备驱动而变化：

- ✓ 内存设备型号
- ✓ 基本文件类型

设备参数 DM_DDTYPE_NON_VOLATILE (6) 内存设备型号

标签	值/值域	默认值	例子	描述
/BA	0-0xffffffff	N/A	/BA=0x100000	内存基地址
/LEN	0-0xffffffff	Depends on HW	/LEN=0x1000	内存长度以字节形式
/DIRECT	0 1	0	/DIRECT=1	是否支持直接读取？

参数/BA 和/DIRECT 可以被 PnP 覆盖。 只有/LEN 参数使用（如果正确）。

设备参数 DM_DDTYPE_NON_VOLATILE (6)文件基本型号

标签	值/值域	默认值	例子	描述
/DEVICE	<DeviceName>	-	/DEVICE=RPSHD	文件设备的名称 (DM_DDTYPE_FILE)
/PATH	<PathName>	.\	/PATH=USERR OM	文件设备的相关路径

例如：
如果文件设备 RPSHD 配置在"C:\RPSHD"目录下，那么文件索引
"PATH=USERROM"在"C:\RPSHD\USERROM"目录下。

设备参数 DM_DDTYPE_ISAIO (9)

标签	值/值域	默认值	例子	描述
/NR	0-7	N/A	/NR=1	给卡片设置的卡片名

ISAIO 输入设备自动产生(Plug&Play)。ISAIO 输入在 XML 下不被允许。

设备参数 DM_DDTYPE_CPUHW (10)

无！

设备参数 DM_DDTYPE_IOTIM (11)

准备中！

物理设备名称：

参数"PhysDeviceName"被关联到“动态”硬件数据，使用 Plug&Play 记录在 Plug&Play 注册里，因为实际的设备名称只有一个逻辑名称自由关联到一个使用者（或者 Automation Studio）。

注意： PnP 记录设备参数只能被记录在对象里。一般来说，有一个基地址（/BA or /IO)和一个中断数字(/IR)。

总线设备

"PhysDeviceName"描述硬件设备路径在下表：

```
<BusType>.<Slot>.<DeviceName>{.<BusType>.<Slot>.<DeviceName>}
{.<DevicePart>}
    <BusType>=ISA, PCI, 2005BUS, LIB, ...
    <DeviceName> = LS191, LS172, IF772, IF060, IF672, ...
    <Slot> = 1, 2, 3, ...
    <DevicePart> = IF1, IF2, IF3, ..., SRAM, ...
```

例子：

PCI.4.LS172.IF1
PCI.1.IF772.1.IF2
PCI.4.LS172.SRAM

ISA.2.LS172.IF1
2005BUS.5.IF060.LIB.1.IF613.IF3

 **注意 (< ARV3.0):**
槽 no.1 被永久设置为 LS172 和 IF772，槽 no.2 被永久设置为 LS191 因为目前的 PCI 设备(在 May 2001)还没有槽切换。

本地 (CPU) 设备

<CPU-Name>.<DevicePart>

例子:

CP360.SRAM

2.4.3.2 以太网

关键名称		描述
xxx		对象名称 (vxTarget)
数据名称	数据类型	描述
开始模式	ui4	激活配置 (开始模式=1)
参数参数	string	选择网络波特率作固定设置 这里缩短时间需要在 V2.65 的开始程序。 有效输入: Entry not available Automatic baud rate recognition /BD=10 Baud rate permanently set to 10 Mbit/s /BD=100 Baud rate permanently set to 100 Mbit/s
界面名称	string	界面名称 (硬件依靠: ene0 or eeE0)
互联网地址	string	IP 地址使用字符串 (例如. 10.43.70.203)
广播地址	string	广播地址使用字符串(例如. 10.255.255.255)
子网掩码	ui4	子网掩码- 十进制: 4278190080 -> 0xFF000000 = "255.0.0.0" 4294901760 -> 0xFFFF0000 = "255.255.0.0" 4294967040 -> 0xFFFFF000 = "255.255.255.0"
默认网关	string	IP 地址给默认网关使用字符串 (例如. 10.43.70.1)。

此区域**可任选**。如果要使用以太网并且这个 CPU 支持的话，仅仅需要一个相应的入口。

如果这个入口暂时不需要，可以通过设置”StartMode”为 0 不激活它。

2.4.3.3 INA设备

关键名称		描述
[SL<z>][.][SS<y>][.][IF<x>]		INA 设备名称依据 INA 设备命名规则。
数据名称	数据类型	描述
Fb 驱动名称	string	INA 设备驱动名称（例如： FbSerial, FbEthUdp, FbCan）
设备	string	优先设备名称(例如: COM1, CAN1, any DeviceName)
Ina 参数	string	INA 设备特殊参数
Ina 节点	ui4	INA 节点号 注意： 目前只有 CAN 和以太网有意义，但是如果节点切换存在，节点号可以使用 Plug&Play(CAN LS172, CP360 以太)读取 。登录进注册表。→如果一个值已经存在在 SYSCONF，那么值被使用 PnP 覆盖。如果节点切换不可用（CAN IPC2001 主板附带 IPC5000 赛扬 Ethernet 主板附带），那么 INA 节点号从 Ina 节点输入 SYSCONF XML 部分是可用的。
开始模式	ui4	开始模式使用 INA 设备 (0=disabled, 1=enabled)

这个区域是**强制的**因为其他的 INA2000 通讯是不行的，配置是无效的（没有意义）的。所有的 INA2000 设备必须输入。

“Ina 参数”

串口：无！

CAN:

标签	描述
/NCNT	CAN 子网节点数
/BID	CAN 基本 ID
/MC	信息计数，最大数字 CAN 包数据发送每 10MS 一次（减少中断负载）
/CRT	CAN 接收超时

以太网/UDP：无！

2.4.3.4 INA2000



Beschreibung folgt!

Dieses Kapitel ist z. Zt. in Überarbeitung!
 Informieren Sie sich bitte in regelmäßigen
 Abständen auf unserer Internet-Seite
www.br-automation.com im Bereich
Produktbezogene Downloads über die
 aktuelle **APROL** Dokumentation.

2.4.3.5 CAN IO主站

关键名称		描述
设备		设备名称。这个设备必须输入 "DEVICES" 区域。

关键名称		描述
数据名称	数据类型	描述
总线	ui4	CANIO 总线数： 这个号码只能在 Automation Studio 中看到（在“特性”下）

此区域**可任选**。如果 CANIO 被使用只需要相应的输入。被使用作为在 AS 显示的总线号和设备之间的连接。如果一个适当的总线没有发现或者 "CANIOMASTER"区域不可用，那么从 CANIOD 的“CAN”装置的使用作为默认。

B2-2.5 模块和卡的配置

几乎所有的模块和卡件都是安装在基本控制中的，扩展和远程 I/O 站将在项目中得到进一步的配置，以便于进一步的工作（例如，创建功能图表）。有经过配置、激活和编译以后，在功能图表中的 I/O 点才可以被选择，并被用以连接输入输出。

这些配置将分成以下的几个部分说明。

2.5.1 远程I/O主站配置

远程 I/O 主站用来将远程 I/O 站连接到主控制器底板。对远程 I/O 主站的**配置**仅仅在项目中相关的**远程 I/O 站有效的**条件下才可以实施（请参见[配置/添加远程 I/O 站](#)）。



在配置远程 I/O 主站时，远程 I/O 站不需要添加所有已连接的模块。安装远程从站就足够了。

2.5.2 I/O卡配置

B&R 有大量可以被安装在控制器上的 I/O 卡件。因此这里不会对所有的 I/O 卡件加以说明。一个详细的说明将仅在各对应型号卡件的条目中提供。

 可以导入预配置。详见[系统手册的导入 I/O 卡件数据文件章节](#)。

下面是一些典型模块配置的图片，以便于理解。

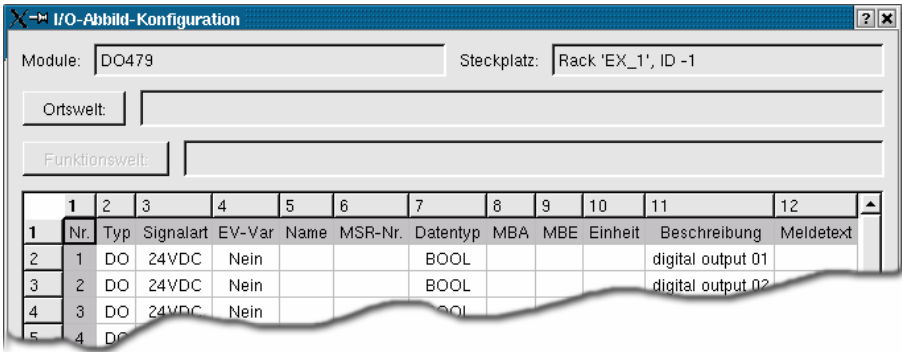


图 42: 24V 直流数字输出 (DO)

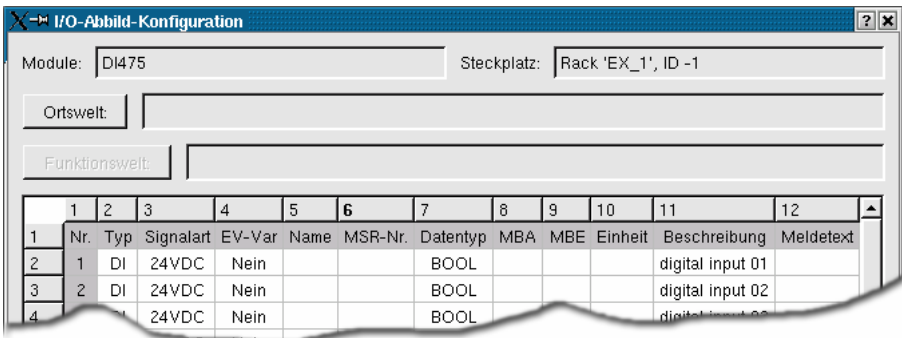


图 43: 24V 直流数字输入 (DI)

I/O-Abbild-Konfiguration

Module: AM050

Steckplatz: Rack 'EX_1', ID -1

Ortswelt:

Funktionswelt:

	1	2	3	4	5	6	7	8	9	10	11	12
1	Nr.	Typ	Signalart	EV-Var	Name	MSR-Nr.	Datentyp	MBA	MBE	Einheit	Beschreibung	Meldetext
2	1	AI	0V,+10V	Nein			REAL				analog input 01	
3	2	AI	-10V,+10V	Nein			REAL				analog input 02	
4	3	AI	-10V,+10V	Nein			REAL				analog input 03	
5	4	AI	-10V,+10V	Nein			REAL				analog input 04	
6	5	AO	0V,+10V	Nein			REAL				analog output 01	
7	6	AO	0V,+10V	Nein			REAL				analog output 02	
8	7	AO	0V,+10V	Nein			REAL					

图 44: 模拟量混合模块 (AI 或 AO)

可以看到在各种不同模块和板卡中某些选项都会出现。 根据模块的类型，可以在 *信号类型* 输入框中选择对用得类型。

名称	描述
序号	通道号。可以在 <i>B&R 用户手册</i> 中找到对应管脚的说明(模块编号：MASYS2SYS-E)。
类型	信号形式： DI=数字输入， DO=数字输出， AI=模拟输入， AO=模拟输出
信号类型	在这个选项中配置输入输出信号的类型（根据不同模块）。 可以是一个常数（例如， 24V 直流或继电器触点）或一个模拟信号（例如： 0V 到 +10V 或 -10V 到+10V）。
EV 变量	对应到该输入的变量，同时又是一个事件变量时，可选择。 详细信息可以在 <i>B&R 连接驱动</i> 手册中找到（在 <i>事件驱动</i> 一章）。 这部分有一个简短的介绍作为附录。
名称	输入输出的名字。 过程变量名。 这个名字必须唯一而且符合 C 语言变量名。
MSR 编号	输入测量点名称。 可选
数据类型	由系统设定不能被更改。
MRB / MRE	测量范围开始/测量范围结束。 在 PCS 上测量范围（即显示量程）的配置在此处设置。 当使用模拟量数据时，该选项必须设置。 <i>最终的显示量程由功能图表决定！ 进一步的信息可以在<i>使用 PIC 量程功能模块</i>章节中看到。</i>
单位	明显的名称。 可选
描述	该选项十分重要！

名称	描述
消息	可选。用于报警和消息系统。



在**事件变量选项**中请特别注意其中的一个**改变**！
所有使用该卡上对应事件变量选项的功能图表必须重新编译。所有相关的任务需要重生成。



控制计算机上的事件驱动的数据过程：
使用各种驱动将控制器上的数据传输到控制计算机的 Iosys 取代了使用一个表决进程。控制计算机上的 Iosys 进一步传递数据。它仅仅在数据发生改变后传输数据，并更新相应的时间标签。

事件变量类型：
控制器监控这些变量以供改变，他们并不是通过**表决进程**传输到控制计算机的。当一个值发生变化时，该过程变量就通过相应得驱动发送到控制计算机的 Iosys 上，并包含一个值改变时的时间标签。过程变量又进入“未改变状态”。



下面的公式为了方便和显示（MBA，MBE）。请注意毫安值和伏特值都被转化成数字量。

从控制器到控制计算机的模拟量值

描述	描述
RVB	升序排列。卡件上的最小值。
RVE	降序排列。卡件上的最大值。
RVAL	卡件上的到达值。
系统中的 PV 值	$(MRE - MRB) * \frac{RVAL - RVB}{RVE - RVB} + MRB$

从控制计算机到控制器的模拟量值

描述	描述
RVB	升序排列。卡件上的最小值。
RVE	降序排列。卡件上的最大值。
RVAL	卡件上的到达值。
AVAL	PCS 值。
发到现场 PV 值	$(AVAL - MRB) * \frac{RVE - RVB}{MRE - MRB} + RVB$



对于数字量输入，只有 MBA=0 和 MBE=1 是被允许的。

B2-2.6 配置/添加远程I/O站

下面这个简短的介绍包括远程 I/O 站和主站本身的使用。

使用:

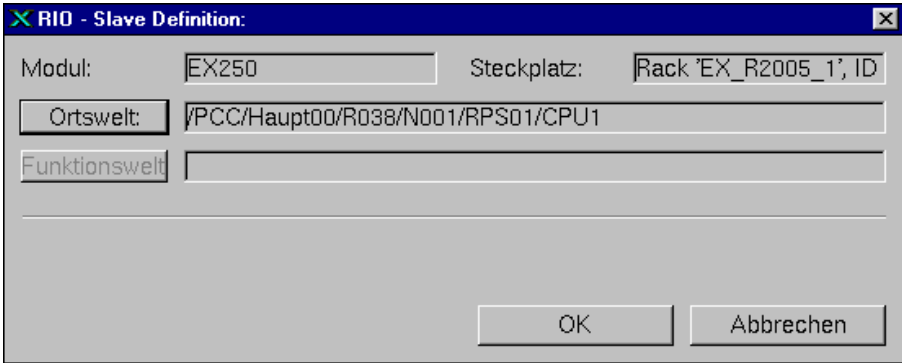
当 I/O 到端子的走线要求短或者需要连接的设备（例如伺服驱动器，电机）离中心控制器的距离比较远的条件下，可以使用 I/O 站。在这种条件下，可以把远程 I/O 站布置在比较靠近现场设备。所有得 I/O 可以直接连到 I/O 站的模块上。

通过远程 I/O 总线，可以连接很多个站。两个站之间总线电缆最大距离 1200m。在一个标准配置中包括一个底板、一个电源模块、一个**远程 I/O 从站**和需要的 I/O 模块。这不是一个智能站。主控制器也是相同的方法配置。只有在主站配置了远程 I/O 主站模块，并经过配置，激活和编译后，远程 I/O 站才可以连接到主站上。详见[远程主站模块的配置](#)。

.

配置:

如下图所示, 你只需要在“**模块属性/远程 I/O 从站**”这个路径下, 输入物理视图(主站的 CPU)。



The image shows a software dialog box titled "RIO - Slave Definition". It contains several input fields: "Modul:" with the value "EX250", "Steckplatz:" with the value "Rack 'EX_R2005_1', ID", "Ortswelt:" with the value "/PCC/Haupt00/R038/N001/RPS01/CPU1", and "Funktionswelt:" which is currently empty. At the bottom right, there are two buttons labeled "OK" and "Abbrechen".

图 45: 在远程 I/O 站上配置远程 I/O 从站

组建的安装:

在主站上的远程 I/O 主站模块和远程 I/O 从站之间的通讯是通过总线实现的。模块和卡件的 I/O 信号数据是通过总线交换的。实际使用这些站、模块和卡件需要, 也必须首先经过配置。请见 [I/O 卡件的配置](#) 章节中相关的配置。

B2-2.7 PCS 计算机配置

APROL 一个过程控制系统(除了连接的控制器)可以包括一台计算机或者一组计算机, 在 **B1 工程环境** 手册的 [APROL 过程控制系统](#) 章节中有一个简短的说明。在 PCS 中的每个计算机都是一个项目组件, 都需要通过 **CAEmanager** 配置, 只有这样相关的组件才能在 runtime 中启动。下载这些配置时, 相关得系统需要安装在计算机上, 并且到工程师站计算机的连接必须有效。

首先介绍“aprol 系统合 1”的配置。有了这个计算机，两个 runtime 和操作员系统都可以包含在 PCS 的 runtime 系统中。此外你可以非常容易的得到一个 runtime 计算机的配置，控制计算机和操作员站也包含在这个配置中。

控制计算机永远是 runtime 实际运行的那个计算机。所有的 PCS 计算机都属于“控制计算机”一类。需要创建一个项目组件“控制计算机”请参见[创建一个新组件](#)一章。

创建这个项目组件并打开后，Caemanager 会显示所有用于输入域设置的选项，按钮和分页。冗余的配置在另一个独立的区域显示。

这一部分包括计算机的全局选项，包括网络标签的描述。根据选择不同的计算机类型，相关的选项和分页标签也会不同。第一步是配置控制计算机。

Orts-/Funktionswelt

Zuweisen

Rechnerkatalog

Leitsystemkatalog 4

Zielsystem

runtime

Rechnertyp

Server Standard Industrial ROE

IOSYS-Port

1

Rechnerinfo

Runtime mit integriertem Operator und

Kommunikation

lappygerd:1

Bestellnummer

AP:SV4-STDI-ROE-0/1

Netzwerk

Netzwerk: Slave

Leitsystemmodule

Gateway

Fremdhardware

Rechnername:

lappygerd

☒ Als Default-Leitrechner verwenden

Bedienbus

Netzwerkadresse:

lappygerd

Netzwerkadresse (red.):

Prozessbus

Netzwerkadresse:

Netzwerkadresse (red.):

Leitrechner-Task

图 46: 计算机配置主页面

控制计算机配置的相关选项，可以在下面这个表格中找到。

输入/菜单	解释:
计算机目录	选择实际的计算机目录。控制计算机目录 4 只有在 2.0 版后才有效。2.0.
计算机类型	选在计算机的类型。例如包含 runtime 系统和操作员系统的类型。控制计算机模块的编译根据这个控制计算机的类型进行。 每次改变计算机类型，控制计算机模块将被改写，原有设置会被复位。
物理/功能视图	定义物理视图。每个不同的计算机在物理视图中必须有不同的项目。
目标系统	在这里输入 runtime 系统的 Linux 登录信息，这个是之前用 aprol 配置脚本安装的。可以以“runtime”这个账户登录。该选项中不允许输入特殊字符。 当一台计算机中运行多个 runtime 系统时，各个不同的系统需要有不同的名字。
IOSYS 端口	如果创建的是一个标准系统（控制计算机）默认的端口是 0。如果在同一个计算机上有多个 runtime 运行，则不同的 runtime 需要有不同端口。
通讯	用以替代实际运行 losys 的 runtime 系统的计算机（右键点击选择）。右键点击后，选择菜单以下拉菜单方式显示。该选项由计算机名和 losys 端口组成。
计算机名	在 Linux 操作系统下，每台计算机在安装操作系统时都有一个唯一的名字。在这里输入该计算机名。可以通过终端输入“uname-r”“GetEnvi-host”这两个命令得到当前计算机的计算机名。
操作总线网络地址	输入由上位计算机共享用得通讯地址，即时只有一台上位机也同样需要填写该项。你所配置的计算机的网络地址（IP 地址）必须被输入。如果在系统中服务器已经得到命名，在这里也可以填写对应的计算机名。 （详见计算机的命名）
过程总线网络地址	仅当使用 Profibus 这种通讯介质时，要输入控制器通讯。安装 Profibus 卡件所在站的地址必须输入。
默认控制计算机	控制计算机的这个选项必须激活。在一个项目中只允许一个控制计算机被激活。在例子中显示的，所配置的系统（包含操作员站和 runtime 系统）也是控制计算机。



在 CaeManager 里对于每个计算机只能创建一个有效的项目组件。除了冗余计算机和热挂机以外。对于冗余计算机两台计算机包含在一个硬件页里面，但是分成两个独立的分页，对于主从站的配置会有两个不同的计算机名。

热备：

runtime 系统的运行站（在 **APROL** 表示为热站）必须有工程师站、runtime、操作员站的授权，并且要有冗余扩展。挂起计算机必须有 runtime 和操作员授权。两个计算机分别进各自不同硬件分页。

一个控制计算机（runtime 系统）就包含了一个操作员站，该操作员站的配置在第一部分。下面的表格仅仅包含了与配置一个远程操作员站（操作员系统）相关的，一些需要注意的选项。

当配置一个操作员站和网网站时，与全局输入不同。

输入/菜单	解释：
计算机类型	选择计算机的类型。选择 <i>工业操作员终端</i> 。设定网关服务器，选择 <i>网关</i> 。
目标系统	在这里输入 runtime 系统的 Linux 登录信息，这个是之前用 <i>aprol</i> 配置脚本安装的。操作员站可以以“ <i>operate01</i> ”账户登陆，网网站可以以“ <i>gateway1</i> ”账户登陆。
通讯	用以替代实际运行 <i>losys</i> 的 runtime 系统的计算机（右键点击选择）。它包括 runtime 系统的计算机名和 <i>IOSYS</i> 的端口。
计算机名	操作员站或者网网站的计算机名。
操作总线网络地址	你所配置的计算机的网络地址（IP 地址）必须被输入。当该计算机名已经在系统中定义（域名解析），可以以该计算机名代替。

在下面的部分主要与控制计算机任务相关，还会介绍由计算机类型决定，相关的几个分页。

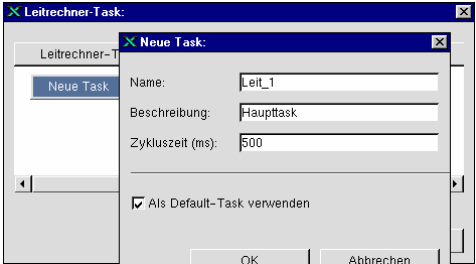
当创建一个独立的配置，例如备站和扩展连接时，请与 **B&R** 服务部联系。

2.7.1 控制计算机任务配置

点击“**控制计算机任务**”按钮，打开控制计算机定义窗口。在 runtime 中，通过在功能图表中物理视图的*控制/系统默认*的定义，将这些功能图表加入导控制计算机任务中。这些功能图表也可以是控制系统功能图表。

如果您进一步了解，控制计算机的功能图表的创建，你会知道这些任务主要与图形显示和 PCS 上的操作有关。从另一个角度来讲，你可以说这些任务建立了控制器之间的通讯，被控制计算机用来执行一些基于计算机的逻辑。这些任务即 **HMI**（人机接口）接口。

控制计算机任务不允许执行控制逻辑，因为这些逻辑需要根据实时的反馈信号才能进行，而 **runtime** 的操作系统并不支持采集这样的实时信号。因此这些控制逻辑必须在控制器上运行。我们建议的控制计算机任务是 **500ms** 的时间循环级别。必须注意的是这个循环时间并不是很严格的。。



左图显示了用快捷菜单打开的“控制计算机任务”窗口，里面有相关配置的各个输入项。。

图 47: 控制计算机任务配置

你可以（也应该）定义多个任务，然后将各控制计算机对应的功能图表对应到任务。将所定义的任务对应到各个控制系统，在控制系统模块中完成。

必须有一个且只有一个控制计算机任务定义为“默认任务”（在“作为默认任务使用”选项后打点）。所有物理视图是“控制系统默认”的功能块将保存再这个任务中（并将在 **runtime** 中运行）。

 下载的任务将被生成。当任务由相关功能图表生成后，任务只有创建相关功能图表后才能被生成。

2.7.2 控制系统模块配置

在“控制系统模块”分页显示了所选择计算机类型的所有 **APROL** 系统程序和应用程序（也被称为控制系统模块）。菜单被分为两个部分（如下图所示）。你可以拖动右侧的滑块，来显示其他所有的选项。

 如果你在配置后，改变计算机类型，所有的配置将会被删除并被新类型的标准配置所取代。

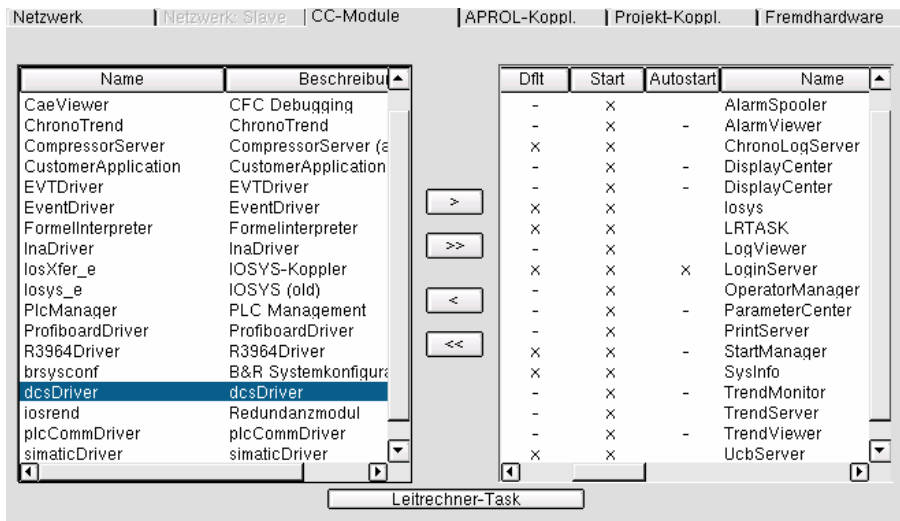


图 48: 控制系统模块配置

在右侧窗口显示的是已包含在 Aprot 系统中的模块列表。当其中的某个模块不在需要时，必须将改模块移到左侧的窗口。左侧窗口中的模块不在系统中使用，但是将这些模块移到右侧后，就可以被系统所使用。

下面的是在 runtime 系统所有的模块列表（代码或应用程序）。根据所属的不同系统，它们被标上不同的标示（OS = 操作员系统，RS = runtime 系统）。有些模块需要在操作员系统和 runtime 系统都存在。在 OS 和 RS 列中标示 X 的模块，都必须同时在 runtime 系统和操作员系统中有效。当你将 OS 列标有 X 的所有模块都删除后，你就将操作员站删除了。

你可以使用 OS 和 RS 两列的标定，来配置不同计算机上的 runtime 系统和操作员系统。

名称	OS	RS	描述
AlarmServer		X	采集所有的报警和消息（事件驱动）并保存。在显示中心显示和提醒报警。在报警浏览器中显示历史事件。将报警和消息发送到报警池中。
AlarmSpooler		X	将报警和消息收集并转化为打印文件，并将其发送到打印服务器。
AlarmViewer	X		显示所要求的历史报警和消息。
CaeViewer	X	X	用于功能图表的调试。 CaeViewer 是操作员站的一个选项，但是在试运行中十分有用。

名称	OS	RS	描述
CronoLog Server		X	用于压缩测量值（压缩数据），同时给操作跟踪器（ Audit Trail log ）提供数据。
客户端应用		X	在这里集中的配置一个或多个外部程序，可以在系统内对其加以调用。也可以定义一些额外的实例用以调用这些外部程序。
显示中心	X		操作员站的主要进程。显示动态过程图和相应的操作元件。包括相应的报警。 <i>必须定义启动画面!</i>
losys		X	启动通过版本 2.0. 过程数据库管理 用以保存任何时间下所有的过程变量。
操作管理	X	X	用于给各个不同的操作员账户配置不同操作 PCS 的权限（利用操作员配置文件）。操作员配置文件是在工程师环境下建立的。
登陆服务器	X	X	控制登录到不同界面，以使对应权限的操作员在 PCS 上做相应的操作。
打印服务器		X	用已配置的打印机，打印报警池提供的报警和消息文件。
控制器管理	X	X	控制器管理器用于工程师站和 runtime 站与控制器通讯。 这是可选的，但十分有用，推荐添加。
启动管理	X	X	提供 PCS 上存在的所有程序的详细信息，并可以用他开始停止各个程序。
系统信息	X	X	系统自我监视。提供系统变量值。SysInfo 收集 losys 监控的相关程序的信息和来自 Linux 操作系统相关的信息。
趋势管理	X		图形化显示选择变量的当前和历史趋势数据。这些数据也从 losys 的趋势数据库收集的。
趋势服务器		X	收集并存储所有趋势数据库中趋势数据。
趋势视窗	X		图形化显示趋势数据库提供的历史趋势。因为数据都是立即写入数据库的，因此也可以认为是一种在线显示。
UCB 服务器		X	使用 UCBs 脚本以用户定义的方式运行事件驱动。
Aggregate Server		X	用于记录和压缩来自 PCS 的过程变量服务器。通过不同进程，您可以监测各个过程变量，并在预设的间隔后，将数据存储在对应的数据库中。 这个服务器使用另外一个小的数据库配置，这个数据库是 AggregateConf 程序创建，并通过 AggregateData 输出数据。这个两个程序始终在系统中，但不会列出。 该部分已经被取消，在新项目中不能被使用。
BrProfiDriver		X	B&R Profibus 通讯卡的 Profibus 驱动。

名称	OS	RS	描述
Compressor Server		X	一个附加的服务器用于压缩和存储来自 PCS 的过程数据。通过不同进程，您可以监测各个过程变量，并在预设的间隔后，将数据存储在对应的压缩数据库中。该服务器使用标准库下压缩数据库路径下的功能块配置。 该部分已经被取消，在新项目中不能被使用。
Formula interpreter		X	一个过时的用于创建控制计算机的选项。不能通过版本 2.0 使用。
InaDriver		X	一个用于和控制器通讯的驱动，根据不同的通讯卡可以支持多种通讯介质，例如以太网、Profibus 或串口。
LRTASK		X	控制计算机任务附加实例，用于执行功能图表中各种函数。 在配置控制计算机任务一章中定义的任务，必须分配在这里！
Parameter Center	X		用于下载系统各个部分控制器中的各组参数（过程变量的预设值）。例如在需要改变产品时，只需要简单的点击一个按钮，启动参数中心相应服务，就可以实现。
Event Driver		X	用于侦测控制器上一个过程变量事件的时间标签。通常情况下，这个时间标签是在这些过程变量抵达 runtime 系统，控制器被生成时，设置的。 该驱动用来对选择的一定数量的事件抵达控制器时，分配时间标签。
Profiboard Driver		X	PROFIBoard Profibus 通讯卡件的 Profibus 驱动。

下面的列表是，当你为*控制计算机*配置*网关*时，根据你的需要所要配置的控制系统模块。

名称	描述
自定义应用程序	在这里你可以定义一个或多个外部程序（驱动），并在系统中启动他们。也可以给外部程序定义附加的实例名。
EVTDriver	该驱动用于建立一个控制计算机到 B&R 2000 系统控制器，事件控制的连接。该驱动使用了 B&R DCS2000 系统中 EVT.DRV 的概念。
losXfer_e	用于建立 <i>APROL</i> 系统和原有系统（例如 <i>APROL</i> E 系统）的数据连接。通过这个连接，原有系统（ <i>losys_e</i> ）中的数据也可以在新系统（ <i>losys</i> / runtime 系统）中访问，方便了系统的升级扩展。 通过这种方式，所有的 <i>APROL</i> 系统的数据，都可以被新系统的 <i>losys</i> 使用（不需要对所连接的控制器的修改）。
losys	启动通过版本 2.0. 过程数据库管理。用以保存任何时间下所有的过程变量。在 <i>当前版本的APROL</i> 系统中，不允许在网关服务器上启动这个服务。这里是一个预留。
losys_e	参见 <i>losys</i> 部分。在 <i>APROLE</i> 版本的演示中。
登陆服务器	控制登录到不同界面，以使对对应权限的操作员在 PCS 上做相应的操作。
系统信息	系统自我监视。提供系统变量值。Sysinfo 收集 <i>losys</i> 监控的相关程序的信息和来自 Linux 操作系统相关的信息。
Dcs 驱动器	用于连接 B&R DCS2000 系统的驱动，通过 Arcnet 方式连接。
losrend	控制计算机冗余用程序。

对于任何一个选择的模块（程序），在窗口的右侧有 *Dflt*, *启动* 和 *自启动* 这三列。

名称	描述
Dflt	当在这里标一个 X ，则表示该程序有效，并且在启动 <i>APROL</i> 时，该程序也会自动启动。
开始	如果只在这里设 X 而没有在 Autostart 设 X ，则程序将在 <i>APROL</i> 启动时启动。否则，程序将不会启动。 操作员管理器（ OperatorManager ）比较特别。必须在 Start 中设 X ，否则将不能通过 KDE 启动。
自动开始	在这里标 X ，会使程序在登录系统时自动启动。除此外该功能不是必须的（这些程序也可以通过 KDE 运行）。

你可以点击模块（程序）这一行，在里面的快捷菜单中选择**开始**或**自动开始**将其打开或关闭。

对于**公式解释器**，**控制计算机任务**和**客户自定义程序**，你可以使用快捷菜单生成附加实例。这个意思就是一些公式解释器，客户自定义程序和控制计算机任务，可以在配置设置后，启动。

你也可以在需要时配置这些在控制系统（runtime 系统，操作员系统和网网站）中需要的模块（在快捷菜单或双击这个模块）。根据需要，配置中的所有选项都可以修改（双击对应选项）。这些选项可以在配置中选择，或者与参数对应，这些选项在菜单窗口都有一个简短的描述解释。

标准条件下，单独一个 **APROL** 系统模块的相关选项都是预设的。但是存在一些例外。



例外1:

在显示中心，启动画面、控制计算机任务、公式解释器中公式的文件名，必须设置。



例外2:

*各个系统计算机 **SysInfo** 程序的实例名（通常就是本事）必须不同（例如 computer_01 上的程序叫 01，则 computer_02 上的程序应该叫 02）。*

下图显示了**显示中心**的配置。所使用的选项（X 表示使用）将在程序启动后被激活。

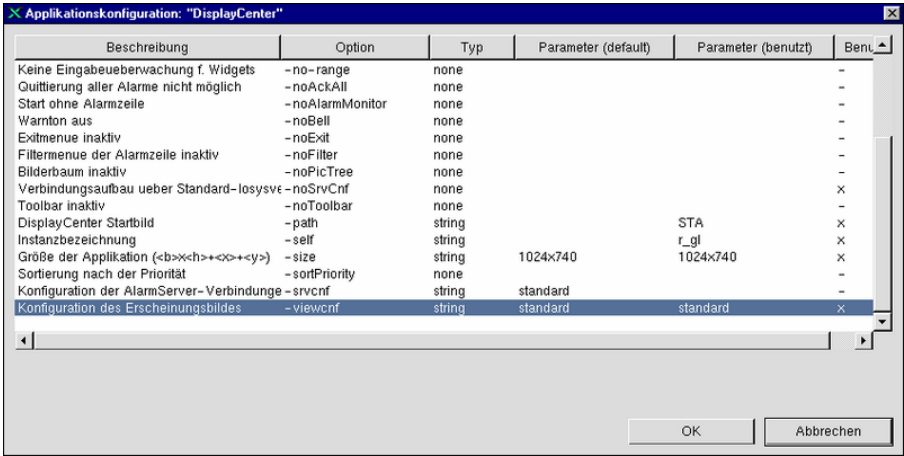


图 49: 配置显示中心

APROL应用程序中支持的几种实例：

对于一些**APROL**应用程序（可见下表），可以设置一些不同的配置实例名。

在这个方式，就可以启动**不同的配置**，针对管理员和系统操作员“显示中心”就能提供一个各自理想的操作环境。下表显示了各个进程所能支持的最大实例数。

前端程序	最大实例数
AlarmViewer	Max. 5
DisplayCenter	Max. 5
LogViewer	Max. 5
OperatorManager	Max. 5
ParameterCenter	Max. 1
ControllerManager	Max. 5
StartManager	Max. 5
TrendMonitor	Max. 5
TrendViewer	Max. 1

下面的例子将会说明如何设置第二个**显示中心**：

在控制计算机的 CC 模块（控制计算机模块）中选择“显示中心”条目。在对应的快捷菜单中选择“添加实例”，于是另一个显示中心实例就被添加了。可以对新显示中心进行相应的配置了。



进行这个操作时，请注意一下对应的实例名(Option "-self")。

这里我们将再注意一下，一些必要附加步骤。

如果需要在 KDE 菜单中也显示这些实例，则需要手动编辑 *desktop* 文件（使用菜单编辑器）。

首先在 KDE 菜单中打开菜单编辑器。在浏览器树形菜单里，选择"**APROL** / Operator KDE Menus / Customer Area / **APROL** / DisplayCenter"。

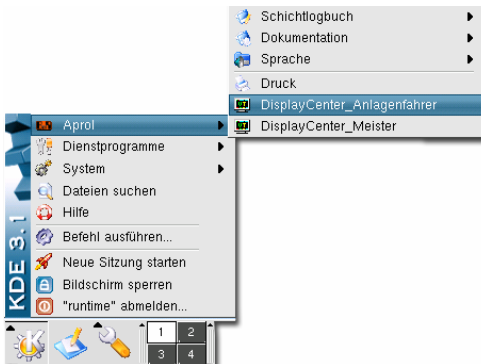


严禁修改 B&R 提供的模版 (BuR_Basic, BuR_Standard, BuR_Professional)。

例如：在"Command"旁边的对话框中输入 **DisplayCenter.start -start <selfID>**。在名称栏输入"**DisplayCenter_SystemOperator**"。

然后创建另外一个“显示中心”（"**DisplayCenter**"）条目。在名称栏，可以输入例如"**DisplayCenter_Supervisor**"这样的名称以示区别。在命令栏，输入 **DisplayCenter.start -start <selfID>**（根据第二个实例的实际的 selfID 号）。

然后下载操作员系统（KDE 菜单"**Customer** / **APROL**"路径下）。



这张图显示如何使用 KDE 菜单选择。

图 50: 在 KDE 中打开不同的实例

2.7.3 配置登录用的外部硬件

当你选择外部硬件组件时，你需要决定允许可以使用那些硬件登陆操作员系统（runtime 系统内嵌操作员系统）。当你登录一个没有操作员系统的 runtime 系统，只需要做 Linux 登录和相应的密码。

通过使用操作员管理器，可以分别对各个操作员配置不同的登陆权限。只能在这里进行操作。

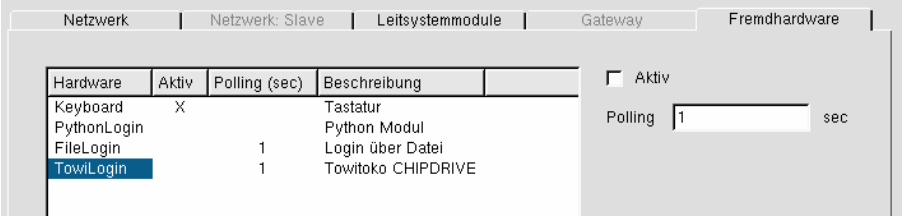


图51: 外部硬件配置

可以使用的选项有 *Keyboard*, *PythonLogin*, *FileLogin* 虚拟设备 and *Towitoko* 芯片驱动（芯片读卡器）。在下载后，计算机将按照你的配置激活组件。在 runtime 系统中，登录服务器（**LoginServer**）将循环的询问这些激活的组件，以确认登陆。

*PythonLogin 在一个独立的手册中加以说明。在使用 PythonLogin 前，需要熟悉 **System API** 手册（Python 的功能扩展章节）。*

在快捷菜单中，你可以激活所选设备，然后可以在“属性”（"Properties"如果有的话，见下表）栏中进一步设置。

FileLogin 和 TowiLogin 的属性

硬件	登录方式	描述
FileLogin	文件名	FileLogin 用的文件名和文件位置。
FileLogin	登录命令	默认的登陆命令。如果要登出，则删除此行。
FileLogin	登出命令	默认的登出命令。如果要登录，则删除此行。
TowiLogin	COM 端口号	COM 的串口号（1-4）

硬件	登录方式	描述
TowiLogin	卡件类型	默认是 <i>APROL</i> 。这个卡件可以在 B&R 订购。 <i>其他雅类型的卡件同样有效！</i>

对于 **FileLogin** 和 **the TowiLogin**，必须在登录服务器上，设置轮询时间。在所选的硬件前的单选器上选择 **Active** 后也有效。

FileLogin 和 TowiLogin 的说明：

FileLogin 通过创建一个文件，来进行登陆登出的动作。登陆服务器一旦找到这个登陆文件（取决于轮询时间），就会进行相应的登入操作，并立即将该文件删除。使用**操作员管理器（OperatorManager）**做进一步配置。无论如何，名字必须被定义（必须在文件的<id>项下输入）。文件结构创建并保存在"**LOGIN <id> | LOGOUT <id>**"这个位址。

卡件的关键代码，通过操作员管理器分配到 TowiLogin。

2.7.4 操作员自动登录

操作员可以迅速的登陆到操作员系统或 runtime 系统。关于登陆的一些补充说明请见 *登陆用途的外部硬件的配置* 章节。

自动登录设置包含在控制系统模块中（登陆服务器中–defUser 的选项设置）。默认登陆名必须输入（如下图所示）。



*操作员系统永远集成了一个控制计算机（runtime 系统）。
对于各个包含操作员系统的计算机，自动登陆选项必需重新输入!*

登录的意义：

操作员 linux 登陆名和密码，登陆相应的 **APROL** 系统（runtime 系统或操作员系统）。如果登陆者希望启动一个程序来操作或监控过控系统，出于安全的理由，改操作员必须重新登录。

如果 *自动登陆启动*，则这里输入的操作员将会在 Linux 启动后，立即登入到相应的 **APROL** 系统，并能够对相应过程进行操作和监控。



*出于对 PCS 系统安全的考虑，这里一般输入 **guest** 账户!*

使用操作员管理器，将配置文件指向 **guest** 账户（相应的访问权限受限，提高系统的安全性）。

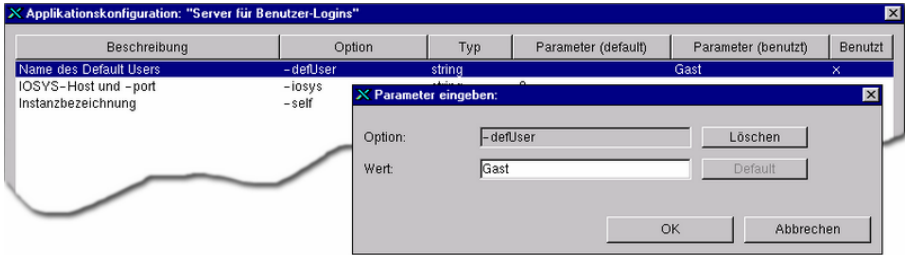


图52: 操作员自动登录设置

B2-2.8 配置过程控制系统网关

网关可以在配置计算机硬件时创建。以下条件必须满足：
对于一台已经安装网网关的计算机，你需要 **G 授权**。

控制计算机中的网关，通过 **APROL** 可以有以下功能：

- ✔ 给功能图表定义全局变量。
没有 **G 授权** 也同样可以定义全局变量。
- ✔ 通过网关计算机，其他供应商的系统就可以连到 **APROL** 系统中。
- ✔ **APROL** 系统可以通过网关与 **APROL E** 系统交换数据。

集成在系统中的网关编辑器可以指导你的配置。

2.8.1 控制计算机中全局变量的网关设置

在很多条件下，全局变量都会在很多功能图表中被使用。变量值在这个功能图里决定和计算。这个值在更进一步的过程中的其他功能图中也需要。在这个例子里，你能用连接器工作。在这种条件下，你需要使用到连接变量或者更高级的选项创建全局变量（在该进程使用网关时）。



这种设置不需要网网站授权！

在任何条件下，必须在项目组件中执行以下配置步骤（在硬件配置第一页）：

- ✓ 网网站（*网关服务器*）必须有自己的物理视图。
- ✓ 控制计算机（包含 *runtime* 系统的计算机）的计算机系统功能需要设置为 *runtime* 系统。
- ✓ *计算机名*和操作总线的*网络地址*，都必须被 *Runtime* 系统的控制计算机所接受。
- ✓ 计算机类型必须为 *网关*。
- ✓ 过程变量只有是*控制系统*物理视图才会被功能图表接受。

关于以上内容的详细描述，请见 *PCS 计算机配置*一章。

相应的配置在**网网站**标签分页下。在选项 *Device-free connection* 下使用 *New*（快捷菜单）创建任务。在这个名字下变量将在随后的定义中被创建。变量在随后会被创建。

下图显示了在**网网站**分页下的菜单，一个已经被创建的“任务”*GW1*和*GW2*“任务”的创建。当输入名字默认的 *X* 将被替代。在这种条件下，*App-Name* 和 *Cnf-file* 选项将不会改变。添加一定的描述往往是很有用的。

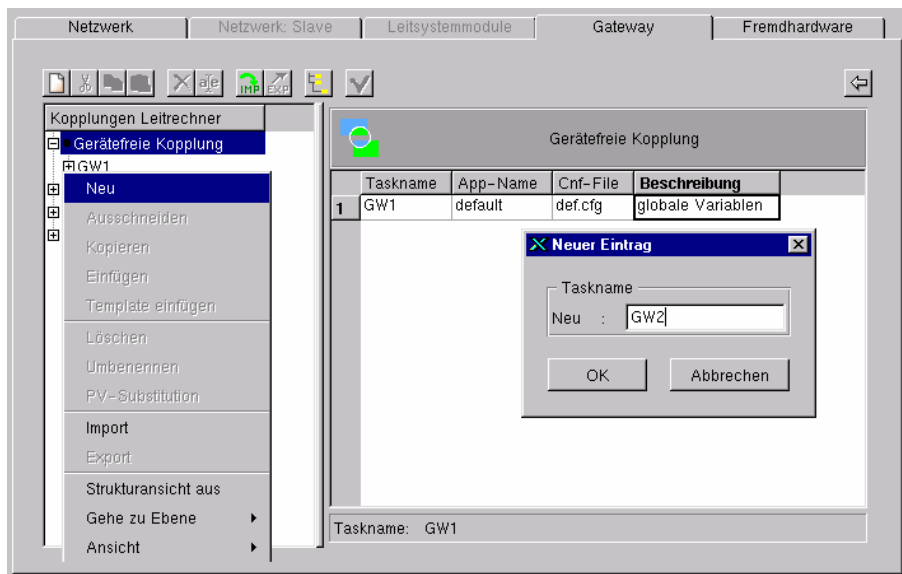


图53: 全局变量表

点击窗口左侧（导航栏）的任务前方的+过程变量将被展开。新变量的添加，也同样使用快捷菜单。默认名同样会被覆盖。

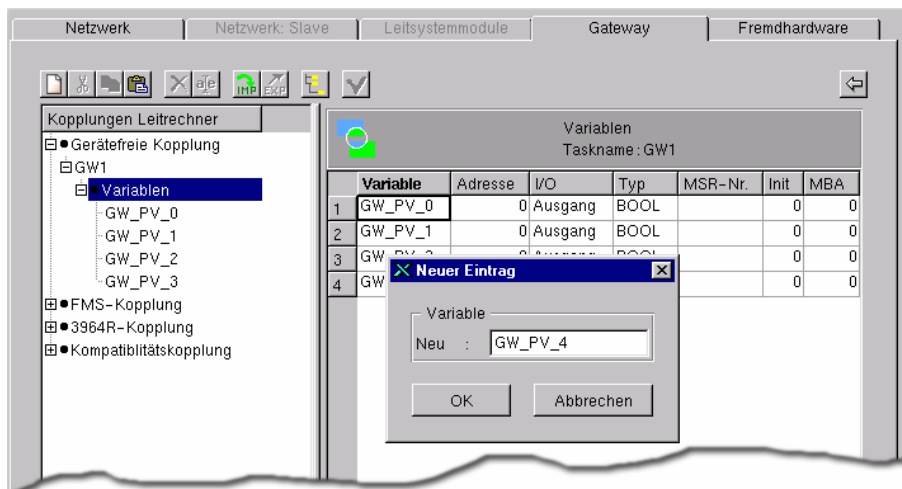


图54: 创建全局变量

下表提供了创建变量所需选项的相关描述。

选项	解释:
地址	在这种条件下，不需要该选项。
I/O	输出是定义全局变量的唯一选项。这里定义控制器上的实际 I/O。默认的，功能图的输出栏里能创建输出。
类型	变量类型。需要时选择。
MSR-Nr	输入测点名（可选）。
初始值	变量的默认值。该值必须在 MRB/MRE 中所定义的范围中。
MRB	测量范围起点，该值必须在所定义的类型范围之内。
MRE	测量范围终点，该值必须在所定义的类型范围之内。
单位	注释。可选。
描述	输入描述，必选选项。
报警文本	保存报警文字（可选选项）

2.8.1.1 显示/复制全局变量

如果任何可获得的数据，其入口可以经过改变并以不同方式显示。

选择一个任务(单个显示)

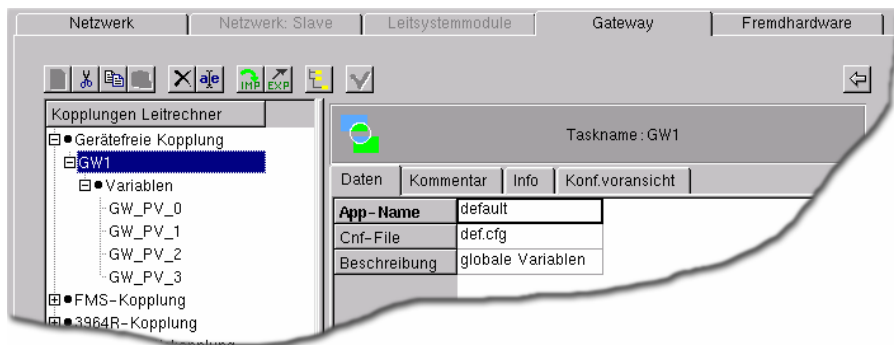


图55: 单个显示任务

显示的类型提供添加你自己的注释的选项。你可以在注释表中输入一个详细的描述，表明使用这些过程变量的原因。

一个全局变量 (单个显示)

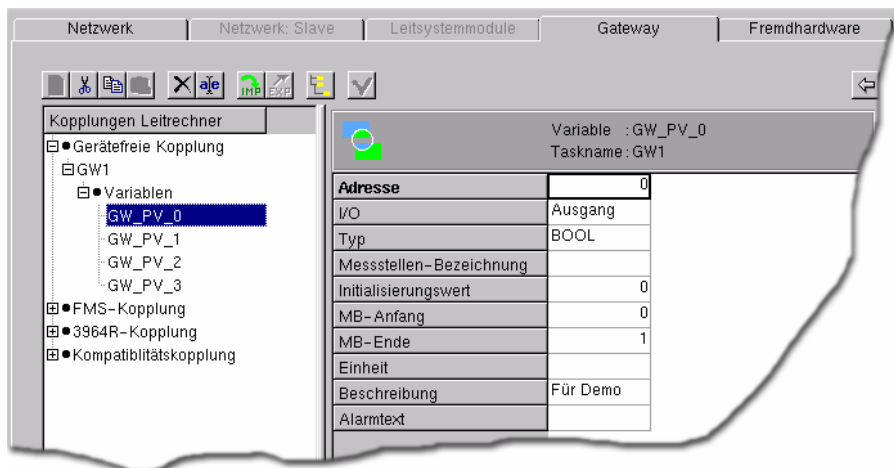


图56: 单个显示全局变量

在上述显示界面, 通过在导航区域选择变量, 可以用快捷菜单在配置区域拷贝和粘贴变量。系统将要求你改变变量名字。

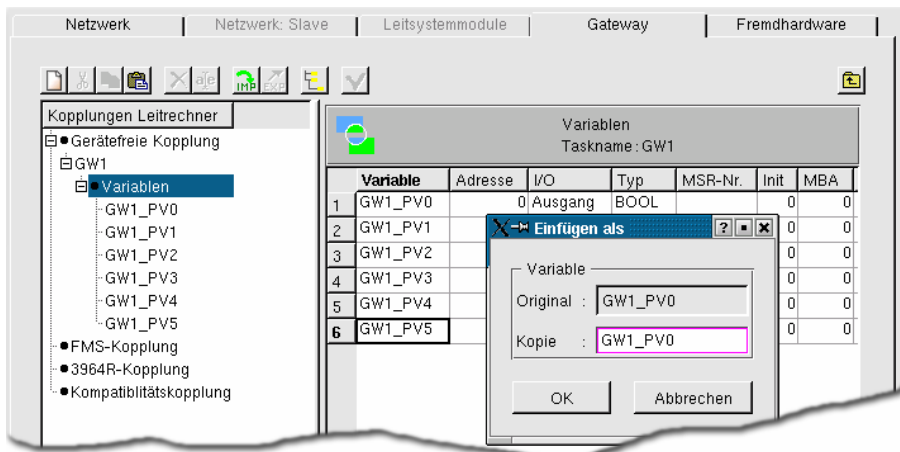


图57: 复制变量

一个完整的任务也可以通过相似方式拷贝。为了做到这一点，所有变量具有唯一的前缀，如例子中显示的一样。当拷贝一个任务时，必须给新任务一个名字，变量的拷贝也一样。系统会自动要求你做这些改变。

2.8.2 APROL外部系统接口

APROL 是一个开放系统，容许从外部取数据和发送数据到外部。要做到这一点，必须满足两个条件:

- ✓ 安装一个单独的计算机作为网关 (*网关服务器*)。
- ✓ 仅仅在网关表格中显示连接模式。

2.8.3 连接到APROL E

APROL 允许你从 **APROL E** 取得数据或发送数据给 **APROL E**。在这种情况下，这种连接是通过网关中 IOSYS 耦合器 **IosXfer_e** 来实现的

下表是描述连接的类型:

- ✓ 网关运行在**APROL** 运行时计算机。
- ✓ 提供一个**APROL** 运行系统。
- ✓ 通过网络连接到一个 **APROL E** 计算机。
- ✓ 通过两个网络连接到一个 **APROL E** 计算机。
- ✓ 通过网络连接到两个冗余 **APROL E** 计算机系统，每个计算机含两个以太网口。

当连接 **APROL** 到 **APROL E** 时，请按下列步骤进行:

步骤	描述
1	创建一个配置文件，通过 <code>IosXfer_e</code> 用来接受数据。
2	配置网关服务器。
3	配置 IOSYS 耦合器 <code>IosXfer_e</code> 。
4	在网关服务器上配置自由设备连接。
5	总结了第一步和第四步兼容性连接的创建。

2.8.3.1 连接到APROL E的文件配置

这个配置文件被用来决定从 **APROL E** 系统的变量。这个配置文件被用来决定哪些从 **APEOL E** 系统来的过程变量将被接受，那些名字被用来存取从 **APROL** 来的过程变量。这个配置文件必须被保存在工程师环境的以下目录:

`.../ENGIN/cnf/IosXfer_e`。这个配置文件可以自定义文件名，但是我们推荐使用 *standard.cfg* 以适合某些标准。 也可以用 VI 创建这个文件。

这个文件具有以下结构:

<在 **APROL E** 中的名字> <交换方向 > <在 **APROL** 中名字>

数据交换方向可以有以下几种选择:

方向	描述
>	数据仅仅从 APROL E 流向 APROL 。
<	数据仅仅从 APROL 流向 APROL E。
=	数据双向流动。
#	注释行。

例子文件 *standard.cfg*

```
# Example configuration
# APROL_E-PV                                Direction      APROL-PV

Sine                                >   IosXfer_Sine
Tan                                <   IosXfer_Tan
Printing                           =   Iosxfer_Print
# The value of print is transferred in both directions.
```

2.8.3.2 配置网关服务器

如果在 **RUNTIME** 计算机上安装网关服务器，必须在硬件页的网络表格上创建下列入口。

- ✔ 网关站（*网关服务器*）必须有自己的物理视图。
- ✔ 控制计算机（包含 *runtime* 系统的计算机）的计算机系统功能需要设置为 *runtime* 系统。
- ✔ *计算机名*和操作总线的*网络地址*，都必须被 **Runtime** 系统的 控制计算机所接受。
- ✔ 计算机类型必须为*网关*。

关于以上内容的详细描述，请见 *PCS 计算机配置*一章。

仅仅 **IosXfer_e**, **LoginServer** 和 **SysInfo** 等过程控制系统模块能够启动。

2.8.3.3 配置 IOSYS 耦合

下列表格描述如何配置 IOSYS 耦合。

选项名称	描述
-file	IOSYS 耦合的配置文件名称。比如 standard.cfg
-iosys	RUNTIME 计算机上的网络地址和通讯端口 (地址: 端口)
-iosys_gw1	RAPROL E-1 计算机上 losys_e 网络地址和端口 (地址: 端口)
-iosys_gw1_b	APROL E-1 计算机上 losys_e1 的替补网络地址和通讯端口 (地址: 端口), 计算机有两个以太网口时, 做总线冗余使用。
-iosys_gw2	APROL E-2 计算机上 losys_e1 的替补网络地址和通讯端口 (地址: 端口), 计算机有两个以太网口时, 计算机具有冗余架构。
-iosys_gw2_b	APROL E-2 计算机上 losys_e1 的替补网络地址和通讯端口 (地址: 端口), 计算机有两个以太网口。



数据通常在两个网之一交换。然而,当一路连接失败2小时15分钟后路径自动切换到另一替代路径上继续进行数据备份。个时间依操作系统和而定不能减少。

2.8.3.4 在网关服务器上配置自由设备连接

被**APROL**接受的过程变量经过 IosXfer_e 在**APROL**系统还没有被组织的起来。仅仅存在 IosXfer_e 配置文件 (参阅连接 APROL 的配置文件 (Configuration file for linking to APROL))

过程变量必须经过自由设备连接进入 APROL 系统, 并且在**APROL**系统中得以重新组织。在 standard.cfg 配置文件中, 你已经在 **APROL E** 和 **APROL**建立起变量索引。在这个配置文件中, 名字被**APROL**用来为网关服务器创建自由设备的连接。额外信息请参阅在控制计算机中全局变量的网关。

2.8.3.5 通过兼容性连接进行配置

在 **APROL** 里，一个步骤来创建 *standard.cfg* 文件和设备无关的连接是可能的。一个兼容性连接在 *Gateway* 属性页上创建。

下列表格提供必要信息。

描述	说明
变量名	在 APROL 中的过程变量名
传送方向	输入数据交换的方向。
I/O	没有使用
类型	定义数据类型。这里类型必须和远端数据类型匹配。
远端过程变量	在 APROL E 中的过程变量名

方向	说明
读	数据仅仅从 APROL E 流向 APROL 。
写	数据仅仅从 APROL 流向 APROL E。
同步	数据双向流动。

完成你的工作后，必须激活并编译网关。一旦你生成了控制计算机任务，系统自动创建 *standard.cfg* 配置文件。

standard.cfg 文件将从下列目录下载到 *RUNTIME*:

**...ENGIN/PROJECTS/<Name Projekt>.pgp/pgm/
pgm/<Gateway physical view>/APROL/cnf/IosXfer_e/standard**

当下载时，这个和 *IOSYS* 通讯的配置文件将被下载到 *RUNTIME*。

B2-3 参照IEC 61131 的逻辑设计

B2-3.1 功能表概述

功能图是组成 PCS 运行功能的程序的图形描述。在 **APROL** 运行系统中的功能图能够真实的技术控制任务（操作和监控）和真实系统控制任务（控制器逻辑），以及两种任务参照一定规则复合的形式。正如下面显示，功能图是用面向逻辑的编程，图形功能块语言（IEC-1131-3）。根据功能图创建的超级宏也完全支持。

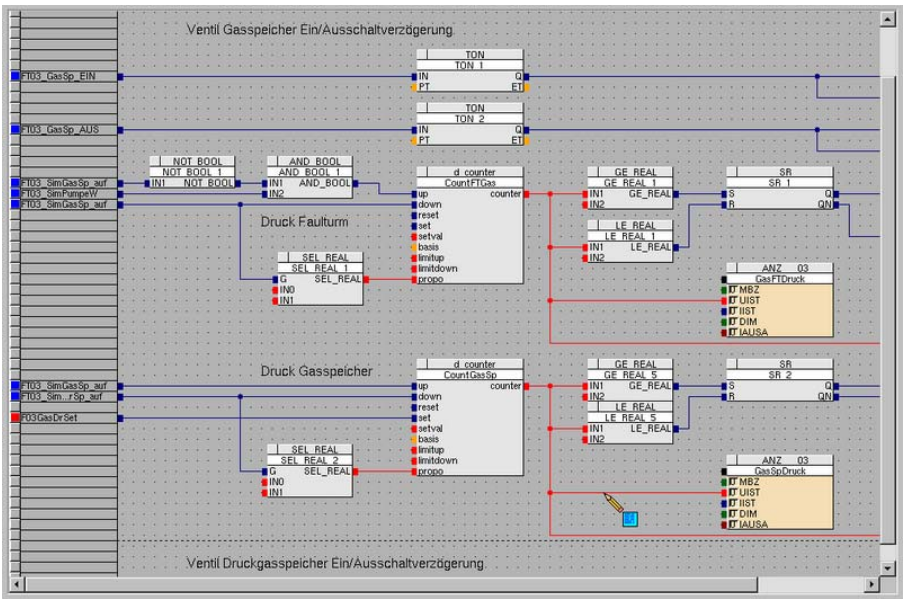


图 58: 功能图表的部分视角

根据一个或多个功能图创建的任务，要么运行在运行计算机，或者某个控制器上。后续章节将描述怎样逻辑地连接属于一个任务的功能图。这个从功能图到任务的连接对一定程度的清晰和组织是必要的。

当创建一个功能图时，一个重要方面必须加以考虑：
被分配给不同物理视图的任务之间需要交换数据。这些数据应该仅仅通过各自的功能图的连接交换。这个过程将在后续章节里描述：

除了逻辑图创建外，下面市给功能图中运行（RUNTIME）系统提供的随后配置：

- ✔ 用报警功能块配置报警和消息系统。参考 功能图中的报警功能块。
- ✔ 配置趋势系统和将模拟量以曲线形式输出。参考 功能图中的趋势功能块。
- ✔ 配置数据压缩。参考 功能图中的压缩功能块。

3.1.1 功能表和功能模块的物理视图

某些过程控制的技术功能应该连接到一个物理视图。这意味着过程控制系统功能块能够仅仅在一个过程控制系统物理视图中执行。逻辑功能块通常放置在控制器物理视图上，在这个物理视图上，他们被处理和连接到你选择的逻辑图上。因为这个原因，在创建一个功能图前，这个用户必须决定运行时功能图应该在哪儿执行。因此，必须给每个功能图选择物理视图和任务。

然而也许对预计算和量程缩放逻辑函数在控制系统层而不是控制器中很有用。功能图的结构（面向过程对象）无须在**APROL**中分开。这容许在一个功能图里完成一个完整程序和马达的配置。控制系统层（控制计算机）功能块能被放在一个以控制器物理视图的功能图内，反之亦然。然而，在连接功能块和物理视图方面，有些规则必须遵循。

在一个赋予控制器物理视图的功能图内的功能块后来用在控制计算机里处理，就必须分配一个目标系统的计算机任务。通过选择快捷菜单(“属性”)来完成各个功能块的分配。这意味着这些“控制”功能块必须被分别选择一个的物理视图，除非他们应该在这个功能图的物理视图上运行。这些控制功能块在功能块内被用红色方块标记。



这个选项仅仅在例外情况下被采用，这个例外就是超级宏。维持一个明显的分开机制将使你的系统更清晰和有组织。

当创建纯过程控制任务比如趋势，报警或视图配置时，控制计算机任务作为物理视图被分配给这个图。在这种情况下，所有被放置在功能图内的功能块被控制计算机处理。当有一个额外的逻辑图被要求时，仅仅逻辑功能块应该被放置在提供给处理控制计算机上。

这里**超级宏**是一个特例，请参阅 功能块的可能物理视图。



基于对整个操作的安全性原因, 我们不建议在控制计算机上运行与安全有关的功能。这个计算机（甚至在一个冗余系统）没有一个控制器能够提供的可操作的安全性。

3.1.1.1 联合功能图形成任务

当合并功能图(在工程环境里创建函数或者过程控制逻辑)以形成一个任务时，请参考下列过程：



用同一物理视图创建空功能图。

控制系统(控制计算机/运行计算机)： CPU 和任务标识

控制器： 控制器，任务等级和任务标识。



当你编程时，你将决定哪些数据将被在功能图之间交换。从功能图 X 的数据，你需要在功能图 N 中使用，将从功能图 X 的输出栏作为**连接变量**（Kn_X）引出，同时该连接变量连接功能图中与之相关的功能块的输出端。无论何时这个信息（Kn_X）在另一个功能图中放置，它应该作为连接变量放置在功能图的输入栏，并与相关的功能块输入引脚相连。

当创建这个任务时，通过这个过程，所有的功能图将在后面被复合组成一个任务。

另外信息请参阅 创建一个功能图和分配 I/O 栏。

3.1.1.2 不同任务间的数据交换

控制系统技术的个别过程要求在个别任务中进行数据交换。

在一个功能图中产生的数据和任务（在功能块的输出）也会要求一个不同任务对应，将被功能块制定一个连接变量，这个连接变量在功能图的输出栏。当这个数据被要求在不同任务（功能图）时，这个连接变量将必须被放置在相应功能图的输入栏和随后进一步在功能图内连接。

数据从一个任务传送到另一个任务的方法必须考虑到这个过程。如需额外信息，请参阅控制器任务级别。

3.1.1.3 功能图表可能的物理视图

下列物理视图可以分配给一个功能图：

- ✓ 一个控制系统（控制计算机）和指定好循环时间任务的物理视图。
- ✓ 一个具有任务级别的控制器和一个相应任务的物理视图。

具有下列选项的功能图，有控制器物理视图并且包含超级宏：

在超级宏里面的函数和功能块有控制系统缺省物理视图，能够被指定给一个控制计算机任务（功能图“属性”菜单）。**一个要求是你已经创建几个控制计算机任务。**

这使得控制计算机部分分成几个任务，运行在各个控制器上。

如果在功能图的属性菜单里，没有选择一个控制计算机任务，这些函数和功能块在缺省任务中被处理。

3.1.1.4 功能模块可能的物理视图

功能块在功能图中放置和互联。如*功能图和功能块的物理视图*所描述的, 功能块可以拥有自己的物理视图 (不同于功能图物理视图)。因此, 下列功能块能被放置在一个功能图内。

- ✓ 功能块的功能能够在—个控制器里执行, 也可以在—个控制系统 (控制计算机里执行)。
你能够用功能块的*属性 (Properties)* 设置其物理视图。
如果功能块的物理视图没有被指定, 其将拥有功能图的物理视图。
- ✓ 仅仅能够运行在控制系统 (控制计算机) 上的功能块。
- ✓ 仅仅能够运行在—个控制器的功能块。
- ✓ 超级宏。这里**超级宏**是—个特例, 超级宏由功能块组成。它们的物理视图要么设置成控制系统-缺省 (*CONTROLSYSTEM-DEFAULT*) (仅仅在控制计算机上执行) 或者 (页面缺省) *PAGE-DEFAULT* (功能图缺省)。功能块的物理视图能够在切换到超级宏的视图时定义。
因此, 你将知道当用超级宏时, 超级宏具有和功能图—样的物理视图。

如果功能图中的功能块具有不同的物理视图, 它们互联时应遵循某些严格的应用规则! 关于这些规则, 请参阅以下章节列出的信息:

- ✓ 从 I/O 栏到功能块的连接是容许的
- ✓ 功能图中的功能块之间互联是容许的

3.1.2 创建功能图表

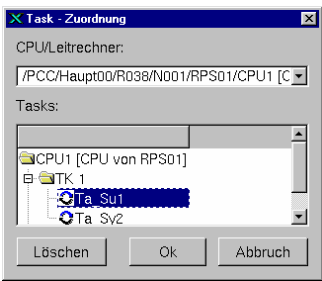
一个功能图的基本创建方法请参阅[创建一个新项目](#)部分。这一节将阐述功能图的特殊输入。

选项	解释:
功能图类型	一个功能图总是使用功能图类型。
例子	除非你指定一个实例名（字母和数字的组合）， 否则功能图不能被成功创建。
物理视图/任务	当生成这个入口时，你应该决定这个功能图在运行时在哪儿执行和使用（由其任务决定）。为了使能这个入口，与其相关的任务运行的硬件必须完全事先配置好，激活并编译才能使用。 请注意下列事实： 除非指定了物理描述，否则功能图不能被编译！
检查框	任务局部变量的调试功能使能将在 功能图的结构和属性 中描述。



在选择功能图之前，必须先创建或者定义控制计算机的任务，控制器的CPU。
更多信息请参考章节:[现场控制器的任务和系统模块的配置](#)以及[现场控制计算机的配置](#)。

物理视图和相应得任务通过[\[浏览\]](#)按钮定义。然后打开"任务分配窗口" 显示相应的选项, 如下图所示。



首先[\[下拉框\]](#) 按钮选择控制计算机以及 CPU。下面一个窗口选择控制计算机的任务。对于现场控制器来说先选择任务等级。 **[OK]**按钮激活选择项。 这些设置可以在功能图的 "属性"框中修改。

图59: 任务分配



利用功能图创建程序部分在分配任务中构成自带容器的模块!

3.1.3 功能图表的结构和属性

一个功能图由一个 DIN A4 组成，它可以被扩展到最大 10x10 页。这页的左边列包含输入栏。右边栏包含输出栏。每个到下一页的切边被点虚线加亮。这个边线不能被功能块覆盖或逻辑操作取消。



我们不建议创建一个最大尺寸为 10x10 页的功能图，因为这能减少逻辑的清晰性。



最好拆分逻辑图的相应部分为 2 个或多个功能图的逻辑互联。连接变量是用来连接各个功能图。

在运行中，每个功能图是一个程序单元，它们被连接变量连接起来。这些功能图组成一个任务。

在功能图中，输入栏总是位于左边，输出栏位于右边。功能块应该放置在功能图内，通过这种方式，数据从左边流向右边。

功能图的尺寸和在“功能图属性”菜单窗口中定义的其他属性。这个菜单窗口也能在功能图的空余部分通过快捷菜单“**属性**”这一项打开。

选项	解释:
类型	初始化 (Init), 循环 (Cyclic) 和退出 (Exit) 功能能够单选分配给一个任务。 更多的详细解释在后续表格描述。
任务局部变量的调试使能	任何能够被编译的功能图, 被下载到控制器并启动, 能在 CaeManager 中以调试模式启动功能检查, I/O 栏上和所有功能块管脚的所有输入/输出改变的值被动态刷新。 <i>如果检查框额外被激活, 任务局部变量的动态改变也能被显示出来。而这个方法导致通讯负载的增加, 因此仅被用于测试目的。整个系统被测试后证明没有任何错误, 这个配置能够被取消, 功能图必须被再次编译, 再次生成系统并下载到控制器。</i>
尺寸	在这节, 功能图能按照选定的方向放大或缩小。减少功能图尺寸仅仅在被减少的图上没有功能块时能够进行。



任务-局部变量存在于两个非保持功能块连接在一起放置于功能图内, 并且这个逻辑连接不包含 I/O 栏上的裸露数据点。

类型	说明
初始化	任务 (被当前功能图代表的) 的这部分初始化时 (启动一个控制器) 执行一次。涉及到控制计算机, 这部分在任务初始化时执行一次。
循环	任务的这部分被循环执行。控制器和计算机中是一样的。
退出	控制器: 当任务被删除时 (还未被停止), 任务的这个部分被执行。一个任务在下载控制器时被删除。 SG4 目标除外: 任务这部分在任务以“拷贝模式”下载时不被执行。 <i>这部分在关闭控制器电源时不被执行!</i> 控制计算机: 任务这部分被执行甚至在正确关闭控制计算机时, 或者当通过标准 KILL 停止任务时。这也在下载控制计算机时发生。 <i>当突然停电时不被执行 (突然掉电导致控制计算机应有功能丧失) !</i>

3.1.4 功能图表中的功能模块

在各个函数库中，下列类型功能块可供选择（用各自颜色标记）：

模块类型	颜色	描述
函数	灰色	用 ANSI-C 设计逻辑，在一个循环执行（其返回值名字和函数名一样）。有且仅有一个返回值（输出）。
功能模块	灰色	用 ANSI-C 设计，有多个返回值（输出）。中的局部变量值可以保持。
图片模块	黄色	逻辑和过程图间的接口。图片模块位于图片宏的后台。
控制系统模块	淡蓝色	贝加莱提供的配置功能块。控制系统功能：比如报警和消息系统，趋势系统和数据压缩。
超级宏	深蓝色	宏（类似于一个功能图，只不过在库里创建）可以如同一个功能块一样放在功能图中。从功能块伸出的 I/O 栏上的连接就是宏的“管脚”，能够在周围的逻辑图中互联。
UCB 模块	红色	非标准类型块。大多数情况下，这些块是用 SHELL 命令编写的。

3.1.4.1 功能模块-管脚分配和视图

所以功能块左边的管脚为输入，右边的为输出。管脚可用的数据类型参照 IEC 标准。



在 CAEManager 通过菜单项“**Help/IEC type info**”查到 IEC 类型值范围和数据类型依赖于物理视图(控制计算机 / 控制器)。
如需自动转换物理视图，**APROL** 驱动器会转换数据类型。 请注意在值范围和精确度方面有限制。

平台和不同数据类型大小在转换时,会有丢失,标记有"三角形的黄色警告"。

下表描述了 IEC 数据类型值的范围

名称	描述	值范围
BOOL	逻辑值(False/True)	0 1
BYTE	顺序位 8	0x0 to 0xff
WORD	顺序位 16	0x0 to 0xffff
DWORD	顺序位 32	0x0 to 0xffffffff
LWORD	顺序位 64	0x0 to 0xffffffffffffff
SINT	短整型	-128 to 127
INT	整型	-32768 to 32767
DINT	双整型	-2147483648 to 2147483647
LINT	长整型	-9223372036854775808 to 9223372036854775807
USINT	无符号短整型	0 to 255
UINT	无符号整型	0 to 65535
ULINT	无符号长整型	0 to 18446744073709551615
UDINT	无符号双整型	0 to 4294967295
REAL	实型	-3.402823466385289 e+38 to 3.402823466385289 e+38
LREAL	长实型 特征: IEC 数据类型 LREAL 与控制器类型 float 上的控制计算机不同 (double)。	-1.797693134862316 e +308 to 1.797693134862316 e +308
STRING	字符变量	0 to 64 characters
TIME	Duration	T#-24d20h31m23s648ms to T#24d20h31m23s647ms
DATE	Date (only)	D#1970-01-01 to D#2038-01-19
TOD	Time of day (only)	TOD#00:00:00 to TOD#23:59:59.999
DT	日期和时间 特征: IEC 数据类型 LREAL 与控制器类型 float 上的控制计算机不同 (double)。	DT#1970-01-01-01:00:00.0 to DT#2038-01-19-04:14:07.0

针对计划设计的所有功能块的某个特定管脚，你有 5 个不同视角（通过“**视图/功能模块**”）。

视图	描述
名称	显示输入输出的名字。
数据类型	显示输入输出管脚的类型。
默认值	所有未使用的输入管脚必须设置初始值。
单位	管脚单位。
描述	管脚的简洁描述。

在 **CaeManager** 菜单区，你会发现左边所视的快速按钮，通过它们有利于快速切换各个视图。



图 61: 模块视图工具条

3.1.5 报警组变量

报警和消息在功能图中通过报警功能块(参考 功能图中的报警功能块配置)。在大多数情况下，你的过程控制系统需要下列信息：

- ✓ 所有等待确认报警的数量，或在一个组中正在等待处理的报警数量。
- ✓ 所有等待确认报警的数量，和要求确认或一个报警组中这种报警的数量。
- ✓ 所有被锁定的报警数量或在一个组中被锁定的报警数量。

被提供描述这个信息的过程变量位于控制系统物理视图的功能图的输入栏上。你能更进一步处理这些变量，这些变量是运行期间报警服务器储存信息的地方。

在当前版本，这些过程变量必须在输入栏可获得前，通过网关用自由设备连接来创建。

变量必须符合下列句法：

以及

Addition1 = 报警服务器实例名称(= Option -self)

Addition2 = 报警组名称

名称	类型	描述/值
AL_<Addition1>_total	I N T	在 PCS 中，所有等待处理报警的数量。
AL_< Addition1>_ack	I N T	在 PCS 中，所有等待处理和要求确认报警的数量。
AL_< Addition1>_locked	I N T	在 PCS 中，所有被锁定的报警的数量。
AL_< Addition1>_<Addition2>_total	I N T	在 PCS 中，仅仅针对报警组。
AL_< Addition1>_< Addition2>_ack	I N T	在 PCS 中，仅仅针对报警组。
AL_< Addition1>_< Addition2>_locked	I N T	在 PCS 中，仅仅针对报警组。



请关注以下报警组变量:

当 CC 模块/报警服务器实例描述被改变,在 APROL connection/Device-free connection/Alarm variables/variables 中改变受到影响变量名。

例如:

AL_01_Fackel_ack

AL_01_Fackel_locked

AL_01_Fackel_total

B2-3.2 编辑功能图表

在功能块图的实际工作包括插入功能块和把它们连接起来创建一个所要求的控制逻辑。 逻辑电路必须连接到 I/O 条。 I/O 边是功能图，过程控制系统的其余部分接口。

创建功能块图后，功能块图必须被激活，编译和最后确认（只要这是正在被使用的版本，就不会被自动删除）。 这些步骤参考 工作版本 中的"激活" 和"编译"。



在功能图被传送前,在功能图里(控制计算机和/或控制器)物理视图所指的硬件必须先被激活并且被编译。

在 **CaeManager** 的视图上（菜单条上），提供了一些菜单包含选项和按钮。 这些菜单是这节的主要焦点。 然而，为进一步澄清，单个按钮将再次在它们各自节内阐述。



"View"按钮启动下列编辑一个功能图的菜单和子菜单。 激活后的菜单项用一个检查标记标明。 子菜单（用箭头标记）分支到额外菜单项。

菜单项/子菜单	描述
功能图	菜单选项。 如果功能块图被选择，在功能图的配置部分显示可编辑。
注释	菜单选项。 如果解释被选择, 在解释页的配置部分，功能图的实际版本显示为可编辑。
导航	菜单选项。 如果这个菜单项被激活，鼠标焦点放在这个功能图上点击一下，你能使用 [Strg] 键很快调用大平面导航" FUP-Navigator "。 更进一步的导航信息能够在 <u>功能图中的导航</u> 章节找到。
全屏	菜单选项。 这个菜单项关闭 CaeManager 的导航部分, 显示功能图为一个 DIN A4 大小。
模块	子菜单。 描述在" 功能块子菜单 "表里。
I/O-栏	子菜单。 描述在" I/O 功能块子菜单 "表里。

子菜单 功能模块	描述
名称	菜单选项。 激活该选项，显示所有功能块的管脚名字。
类型	菜单选项。 激活该选项，显示所有功能块的管脚数据类型。
常数	菜单选项。 激活该选项，显示所有功能块输入的预设常数值。

子菜单 I/O-栏	描述
名称	菜单选项。 激活该菜单项后，I/O 栏上显示每个 I/O 的名字。
描述	菜单选项。 激活该菜单项后，I/O 栏上显示每个 I/O 的描述。

快速选择编辑模式、某个菜单项和一些上面提起的菜单，这些模式选择出现在按钮和菜单区域。下面图形显示功能图的按钮。后面的描述对应从左到右的图标

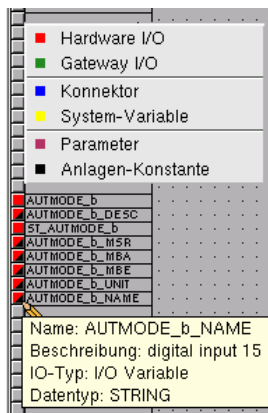


图 62: 功能图按钮

- ✔ **[导航]**按钮。这个按钮激活导航"FUP-navigator"。鼠标光标位于功能图内后，点击**[Ctrl]**键。
- ✔ **[调试-模式]**按钮。启动当前打开的功能图的调试模式。这种情况下, 功能图所包含的任务必须位于目标系统上, 并且已经下载并启动。
- ✔ **[功能图-属性]**按钮。选择功能图的功能图属性。
- ✔ **[编辑]**按钮。按下这个按钮，编辑功能图内的功能块和逻辑线的"缺省（Default）"设置。 **这个按钮被选择来移动功能块和逻辑线。**
- ✔ **[逻辑线]**按钮。仅仅针对功能块逻辑互联。
- ✔ **[注释]**按钮。用注释说明标注功能图。
- ✔ **[名称]**按钮。显示管脚名字。
- ✔ **[类型]**按钮。显示管脚类型。

- ✓ [内容]按钮。显示管脚的预设值。
- ✓ [显示扩展/标准工具提示]按钮。

3.2.1 分配I/O条



I/O 条构成了与控制器硬件 I/Os, 其它功能图表, 控制器, Linux 系统, 和通过参数中心能下载的参数等之间的界面。硬件输入, 网关 I/O, 连接变量等在输入框里能很方便输入。

在输入或输出区右击可以显示分配选项, (如左图显示)。

图 63: 选择 I/O 时的输入选项

3.2.1.1 图表输入条的I/O扩展信息

扩展信息也能存储在**输入区域**的图表除了某些值外 (硬件点, 网关节点 连接变量等)。

下列扩展信息可以被存储:

- ✓ 过程变量的描述
- ✓ 过程变量的状态

- ✓ 计算点数
- ✓ 计算开始/结束范围
- ✓ 用来记录计算值得单元
- ✓ 过程变量的名字

I/O 信息在输入区域创建管脚时通过选择对话框来保存。

也可以在存储数据范围的同时存储输入区域的描述。只要各自的输入条选择"使用扩展 I/O 信息"。相应的信息选择在"checkmark symbol"。

实际的过程变量存储在输入表，在选择列表选择 "值" 这项。

Hardware I/Os:

Suchfilter:

Name: F03FackelFI MSR-Nr: * EV-Var: * Datentyp: * Rückwandmodul: * Kartentyp: *

Beschreibung: * I/O-Typ: Eingang PNK: * Steckplatz: * Divergenz: 6 minimal maximal

Hardware I/Os:

Name	I/O	Event-Variable	Datentyp	Einheit	MSR-Nr	Beschreibung
F03FackelFI	Eingang Nein		BOOL			Faulturm Fackel Flammwächter
F03FackelEin	Eingang Nein		BOOL			Faulturm Fackel einschalten
F03KUEHL	Eingang Nein		REAL	%		Störgröße Kühlen Faulturm

Erweiterte I/O-Informationen verwenden

Platzierung	Information
1	✓ Wert
2	✓ Beschreibung
3	✓ Status
4	✓ MSR-Nr

OK Abbrechen

图 64: 硬件 I/O – 储存扩展 I/O 信息



使用这些按钮可以修改选择列表的顺序。

从 "使用扩展 I/O 信息" 列表里选择的所有信息通过选择对话框的[OK]按钮来确认。

功能图表里信息的顺序与选择列表是一致的。

存储在输入区代表扩展信息的管脚 通过 "黑三角" 来指示并以相互关联的组来存储。

"没有足够的空间来存储选择的信息" 出现时表示没有足够的空间保存这些信息。 如果出现这种情况, 在输入区可以移动一些管脚到其它地方来作调整。

实际的过程变量存储在输入表, 在选择列表选择 "值" 这项。

图表输入和输出的工具提示提供了进一步处理功能图表的重要信息。

3.2.1.2 在栏上分配I/O

你必须熟悉栏目上不同的选项显示 I/O 分配状态。 **输出栏只分配给模块输出**, 连接器用于传输信息, 网关 I/O's 是将数据传输到其它厂商的硬件。

分配功能的每个选项在下表中有详细的描述。 请注意每个菜单都包含的信息比图示的要多。 当滚动条出现时, 窗口中会显示更多的信息。

下表简述了输入和输出栏。

条目	输入	输出	描述
硬件	X	X	与硬件相连(i.e. 字段)。
网关 I/O	X	X	与外部硬件相连 (i.e.到字段) 或者工程设计的全局变量。
连接器	X	X	与其他功能图连接。
系统变量	X		系统变量从计算机系统传送到过程控制系统。
参数	X		工程设计参数中心的参数传递。
植入常数	X		从设备中接收常数。



如果每个项目元件被激活或者翻译，只显示条目。

3.2.1.3 连接I/O栏到功能模块

图表的 I/O 栏中的元素用户访问信息并传送图表信息。然而,并不是每一个输入/输出栏的元素与每个功能模块相连。请注意由于功能模块分配给不同的物理视图经过允许才能建立连接(物理视图的传输)。下表显示经过允许的连接。

输入栏实例 1：

功能图表上有控制计算机的物理视图,并显示任务[1]。所有可能的输出栏输入到左栏。功能模块的物理视图位于第一行。



此实例不符合要求，因此不能效仿。

通过下例实现功能图中不同的物理视图(比如， 控制器_1/ 任务等级 2/ 任务 [1])。

输入条/功能模块	默认图	图表任务[n]	控制器-1 TC1 任务[1]	控制器-1 TC1 任务[2]	控制器-1 TC2 任务[3]	控制器-2 TC1 任务[4]	控制器-2 TC1 任务[5]	控制器-2 TC2 任务[6]
HW-In controller-1	Yes	Yes	Yes	Yes	Yes			
HW-Out controller-1	Yes	Yes	Yes	Yes	Yes			
Gateway-In (controller-1)	Yes	Yes	Yes	Yes	Yes			
HW-In controller-2	Yes	Yes				Yes	Yes	Yes
HW-Out controller-2	Yes	Yes				Yes	Yes	Yes
Gateway-In (controller-2)	Yes	Yes				Yes	Yes	Yes
GateWay-In (control computer)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Connector	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

System var.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Parameter	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Parameter_K	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

解释:

TCn 是现场控制器 4 个任务等级中的其中之一。

Gateway-In (controller-n) 是网关-输入变量。通过 controller-n 访问网关。

Gateway-In (control computer) 是网关-输入变量。通过控制计算机访问网关。

输出栏实例 2：

功能图上有控制计算机的物理视图, 并显示任务[1]。所有可能的输出栏输入到左栏。功能模块的物理视图位于第一行。

 此实例不符合要求，因此不能效仿。

通过下列实现功能图中不同的物理视图(比如， 控制器_1/ 任务等级 2/ 任务[1])。

输出条/功能模块	默认图	图表任务[n]	控制器-1 TC1 任务[1]	控制器-1 TC1 任务[2]	控制器-1 TC2 任务[3]	控制器-2 TC1 任务[4]	控制器-2 TC1 任务[5]	控制器-2 TC2 任务[6]
HW-Out controller-1	Yes	Yes	Yes	Yes	Yes			
Gateway-Out (controller-1)	Yes	Yes	Yes	Yes	Yes			
HW-Out controller-2	Yes	Yes				Yes	Yes	Yes
Gateway-Out (controller-2)	Yes	Yes				Yes	Yes	Yes
GateWay-Out (control computer)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Connector	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

解释:

TCn 是现场控制器 4 个任务等级中的其中之一。

Gateway-Out (controller-n) 是网关-输出变量。通过 controller-n 访问网关。

Gateway-Out(control computer) 是网关-输出变量。通过控制计算机访问网关。

3.2.1.4 为控制器之间的连接作准备



关于过程总线冗余的相关信息请参见 **APROL** 文档“F1 驱动器的连接”!中过程总线冗余一章!

一个复杂的过程控制系统几乎是由多个现场控制器组成。控制器之间的数据交换是经常性的操作。数据交换是将数据从现场控制器上传输到控制计算机上(一个控制系统任务),现场控制器并从控制计算机上接收数据。

当一个控制系统中存在多个现场控制器,他们通常连接到同一个通信系统中。因此,一般是通过最短的路由来交换数据,不用通过控制计算机任务绕道。

之前讲述的是现场控制器之间的连接或者 **APROL** 中现场控制器的横向通信。



通过路由到其它的现场控制器访问一个现场控制器实现现场控制器之间的连接。

通过连接器交换数据实现现场控制器-1 到现场控制器-2 的交叉通信。通过现场控制器的 CPU 上通信标签建立潜在的横向通信。



然而,有选择的控制器连接并不以为这你应该随便的在控制器之间交换数据。
你应该根据功能分配功能图(以及现场控制器的逻辑电路)!

在准备控制器之间建立连接的阶段,你应该考虑到一下问题:



在连接中尽量少用连接器,增加连接器的数量以及总线负载。

- ✓ 现场控制器必须可以横向通信。Profibus 连接一个现场控制器，而第二个现场控制器由 Ethernet 连接是不能实现的。
- ✓ 两个现场控制器之间所有的横向通信除了必要的连接到控制计算机外，都需要一个或者两个连接（关于软件）。通信层必须尽可能多的允许连接的数量！

3.2.1.5 访问硬件I/O

通过选择 *硬件*, 你可以访问现场控制器硬件上的输入栏和输出栏。

关于此类信息，请参见章节 [通过图表上的输入栏访问扩展的I/O 信息](#)。

如果相应的部件安装到现场控制器并且 I/O 卡配置正确，选择才有效。如果此类步骤都没有执行, 显示的菜单就没有输入项。

如果卡配置时带输出信号, 则这些信号会在输入输出栏中显示。

输入栏中也提供输出，原因如下：

此选项可以立即反馈硬件输出的触发逻辑，并经另外的逻辑检查功能图表的输出配置(通过触发逻辑)。不用反馈另外的连接器的检查信息！

Hardware I/Os:

Suchfilter:

Name: F03FackelFI

MSR-Nr: *

Beschreibung: *

EV-Var: *

I/O-Typ: Eingang

Datentyp: *

PNK: *

Rückwandmodul: *

Steckplatz: *

Kartentyp: *

Divergenz: 6

minimal

maximal

Hardware I/Os:

Name	I/O	Event-Variable	Datentyp	Einheit	MSR-Nr	Beschreibung
F03FackelFI	Eingang Nein		BOOL			Faulturm Fackel Flammwächter
F03FackelEin	Eingang Nein		BOOL			Faulturm Fackel einschalten
F03KUEHL	Eingang Nein		REAL	%		Störgröße Kühlen Faulturm

Erweiterte I/O-Informationen verwenden

	Platzierung	Information
1	✓	Wert
2	✓	Beschreibung
3	✓	Status
4	✓	MSR-Nr

OK

Abbrechen

图 65: 输入硬件 I/O 菜单

 激活卡上的"I/O 状态" 类型，模拟量输入 4 - 40 mA。在这个情况下，一个过程变量命名 "ST_<name>" 必须在输入栏中创建。当传感器的电流小于 4 mA 时，过程变量发出"接线断路"的信号!

项目中所有的硬件输出都在菜单中显示。过定义相应的过滤器来限制数据量。



输出栏中只能一次输入一个硬件输出。因此，当在输出栏中选择硬件不会显示已有的输出!

当配置 I/Os 时优化搜索：

通常，大型的项目中会有多个现场控制器以及更多的 IO 点数，容易造成名字的混淆(硬件 I/O, 系统参数，系统常数，网关 I/O)。

为了提高在工程设计系统中的工作效率，**APROL** 为您提供了一个智能的算法(**Levenshtein distance**)，用于在功能图上选择 I/O。

与通过 wildcards (*, ?, etc.)执行搜索不同的是，如果输入的是不确定或者不正确的信息，同样也会得到正确的搜索结果。多个设置可能会流失信息。



请注意选择对话框中会有优化以及简单的用户指导!

不同于已有的方法, 只要在变量列表中轻轻点击便可执行。推荐选择"模糊查询"模式。

通过文本菜单(column depedent) 直接传输搜索字段中的搜索术语。

Name	I/O	Event-Variable	IEC-Typ	Einheit	MSR-Nr	Beschreibung
Pump_82102_COn_ST	Eingang	Nein	BOOL	---	P82102_ST	Statusmeldung Entleerungspumpe Fauturm ein
TRC_17121_Ext_ST	Eingang	Nein	BOOL	---	TRC17121F_ST	Statusmeldung Temperaturregelung Reaktor 1 extern
TRC_17121_Fgb_ST	Eingang	Nein	BOOL	---	TRC17121_ST	Statusmeldung Temperaturregelung Reaktor 1 freigeben
Vent_17101_CClose_ST	Eingang	Nein	BOOL	---	ZO17101SL_ST	Statusmeldung Zulaufventil Grundstoff D Reaktor 1 schließen
Vent_27102_COpen_ST	Eingang	Nein	BOOL	---	ZO27102SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_27103_COpen_ST	Eingang	Nein	BOOL	---	ZO27103SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_27104_COpen_ST	Eingang	Nein	BOOL	---	ZO27104SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_27105_COpen_ST	Eingang	Nein	BOOL	---	ZO27105SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_27106_COpen_ST	Eingang	Nein	BOOL	---	ZO27106SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_53101_COpen_ST	Eingang	Nein	BOOL	---	ZO53101SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen
Vent_72103_COpen_ST	Eingang	Nein	BOOL	---	ZO72103SH_ST	Statusmeldung Zulaufventil H2S Reaktor 2 öffnen

图 66: 选择对话框中的模糊查询

选择 菜单项 **搜索... 相似** 下的**"Name", "Description", 或"MSR-No."**传输到搜索字段, 同时其它字段设置成**"***", 并根据 Levenshtein 算法进行模糊搜索。



模糊搜索仅在**"名字", "描述", 和"MSR-No."**中有效。

过滤输入字段用于 模糊搜索, 其背景为黄色。

选择**"模糊搜索"** 可以在三个属性中转换。选择**"取消"**关闭模糊搜索。

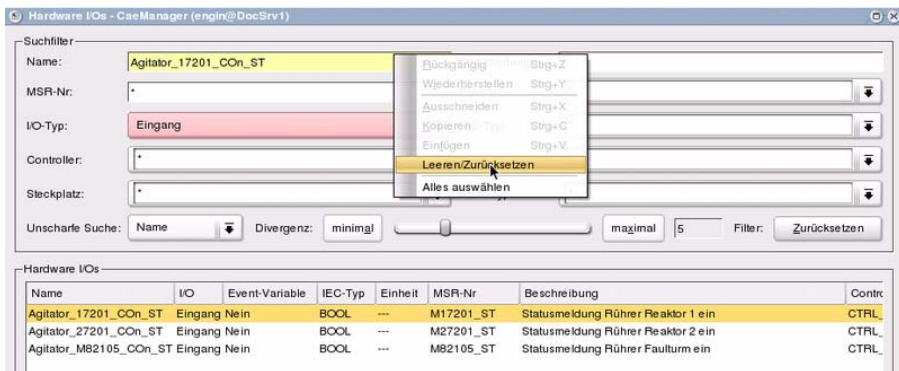


图 67: 为搜索过滤标上不同的颜色

在选择列表中，也可以激活一个"精确"搜索, 通过相应的过滤器。选择此菜单项，列表中的输入项被传递到相应的搜索字段，并开始搜索。所有的过滤输入字段, 有值但是不等于"*", 可以用于精确搜索, 其背景是粉色。

字段 "Name", "Description" and "MSR No." 直接被作为过滤设置采用。其它的输入字段要在按了[返回] 按钮后，作为过滤设置采用（如，现场控制器，插槽等）。

 经菜单项 "Empty/Reset" 来重新设置编辑过的过滤字段。

点击[重置]按钮后，所有过滤器被重新设置成"*"。但是关于模糊搜索的过滤并为改变。

3.2.1.6 访问网关I/O

在输入/输出框选择网关I/O 可以接受/发送数据去/来自外部系统。

可参见 在输入条获取扩展 I/O 信息。

APROL中需要有一个网关服务器并且/或已经配置好网关的控制器。如果一个网关连接建立了则变量可以以全局变量使用。



网关变量**输出**在功能图表的输出框只能使用一次。如果一个网关变量在输出框已经使用过则在输出框的网关变量选择里选择不到该变量。

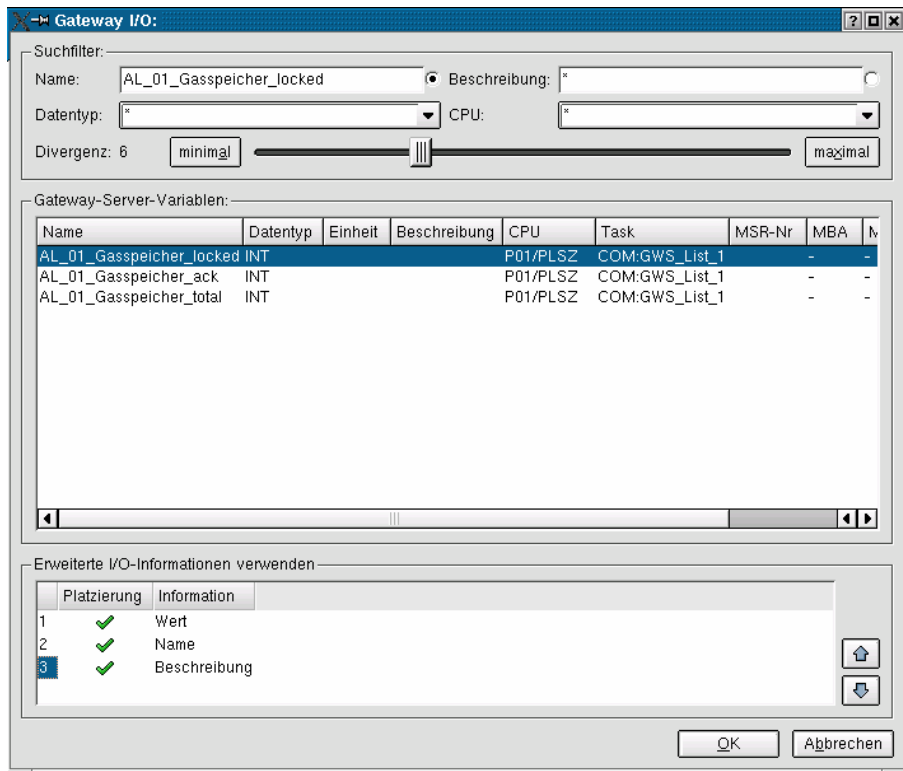


图 68: 采用网关变量菜单

接受和限制数据量的过滤机制参见[获取硬件输入输出变量](#)。

在这一点上，要注意网关变量既可以在控制器上也可以在控制计算机里（网关服务器）。

在输入框选择变量时**输出**和**输入**类型变量都可以选择。但在输出框选择变量时，只有输出类型的变量能被选择。

原因如下：

配置网关的时候 信号流的方向 (*OUT*, *IN*) 控制器上的 I/O 卡是一样的。**输入**

类型变量只是发送数据到 **APROL**。输出类型的变量则发送数据到外部系统，但同样可写回到项目部件，如硬件输入输出点。

3.2.1.7 连接变量的输入和获取

连接变量是用来建立在同一个 CPU 处理的不同功能图表的连接。

相关参见 在输入区域获取扩展 I/O 信息。

连接变量菜单是用来选择和获取在输入框里已存在的连接变量。此菜单项只用来在输出框定义新的连接变量。



在输出表里连接变量只能定义一次!

如果连接变量使用来连接不同任务等级的任务时则需要考虑任务的运行时间。

但数据传输从慢的任务向快的任务时，要确保 "传输数据" 就是慢任务完成后的值。快速任务只是暂时从上一循环的慢任务里获取数据。

反之，慢任务要直到任务开始才能接受到数据。这就意味着会有一些数据丢失。



连接变量被使用后，可以通过查看属性来查找此变量是在那哪里被使用。

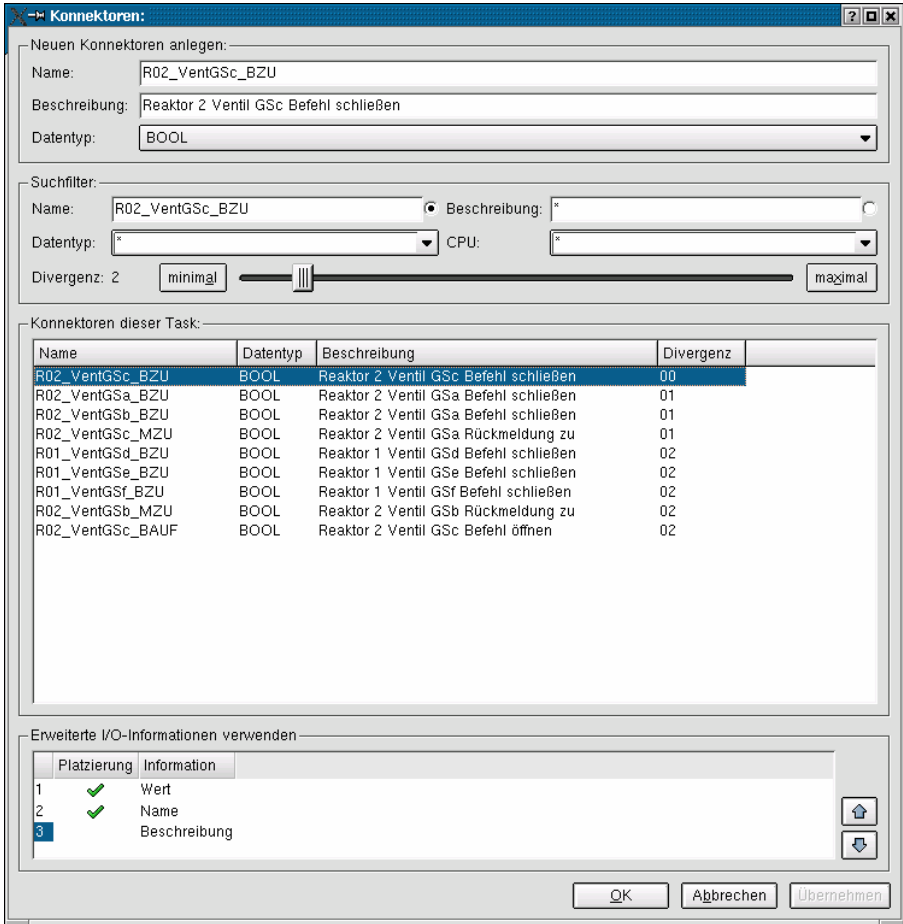


图 69: 创建和获取连接变量的菜单

3.2.1.8 输入系统变量

过程控制系统的程序运行在每一个运行的计算机里(操作员站 1 到 n, 运行服务器和网关服务器)。运行在计算机里的程序取决于被赋予的控制系统模块。它主要是用来监控程序和过程图里的特定数据。

相关参见 [在输入区域获取扩展 I/O 信息](#)。

系统变量用来从控制系统监控功能图表。调用这个菜单时，所有系统变量的格式和描述都显示在列表里以便选择。当然，每个站的所有硬件都必须被配置，激活和编译。

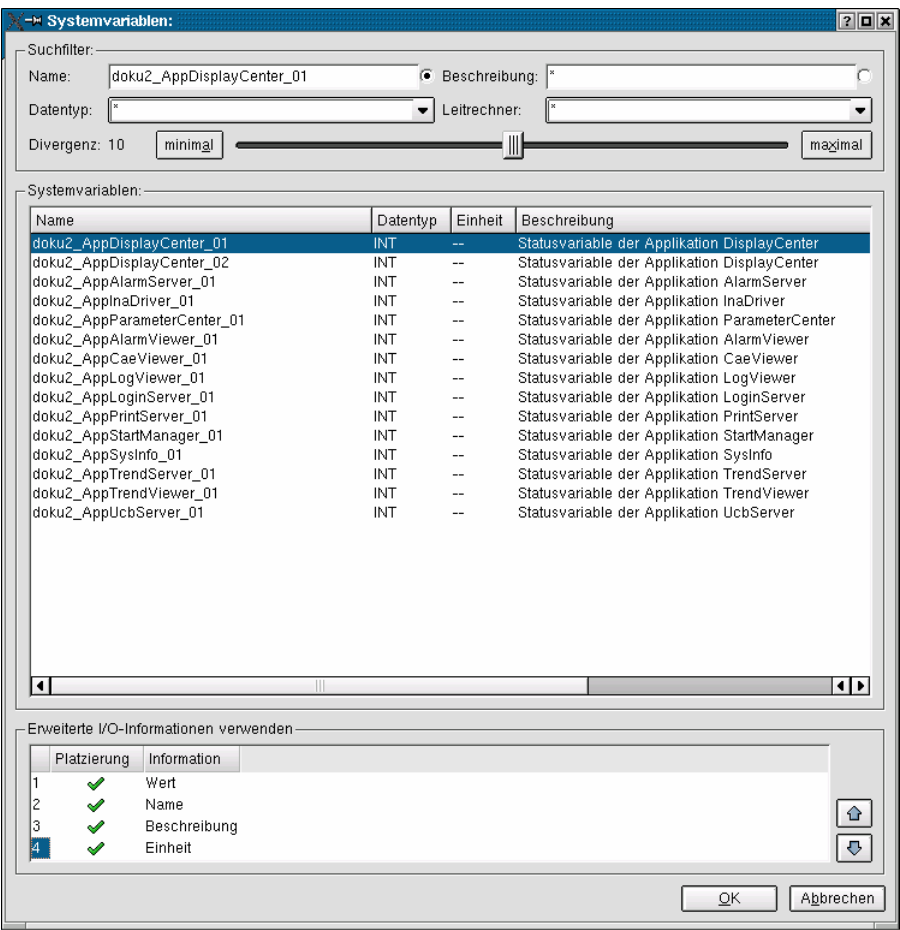


图 70: 接收系统变量菜单

有关系统变量的详细描述参见系统手册中系统变量一节。下面是一些例子。变量 doku2_AppTendServer_01 是 INT 类型，是 TrendServer 里面的一个状态变量。下面是相应值代表的意义：

返回值	说明
0	错误! 变量没有被提供，应用程序的状态不能获取。 原因： 程序尚未启动。 或者 losys ， SysInfo 或程序未启动。
1	应用已启动并运行。
2	功能未定义。
3	功能未定义。

变量 **doku2_IdleStatePercent** 类型是 **REAL** 用来显示处理器负荷。

APROL 系统保护信息，**加密狗**，可以在这里获取。 这些是系统变量：**doku2_DongleDemoTime** and **doku2_DongleState**。 这个用来监测使用演示版 **APROL** 创建的项目的剩余时间。 如果有加密狗则相应系统变量的值为 1。

3.2.1.9 输入参数和系统常数

具有参数和参数集的部件是通过配置设备来配置的。 详情参见[创建参数模板](#)。

相关参见[在输入区域获取扩展 I/O 信息](#)。

系统运行时设备参数集的参数是从控制计算机到特定的地方, 具体取决于运用场合。 参数值可能运用在控制器或在运行系统做进一步处理。

因此，参数可以使用在具有物理视图的控制其和控制系统的图表里然后做进一步处理。 在图表里的连接就确定了参数下载时此值输入的确切地方。

常量同样在部件里定义。 然后这些常量就可以使用了。

参数和系统常量一样, 只能用在输入区。

下图显示的是在输入区插入参数的一个实例的菜单窗口。

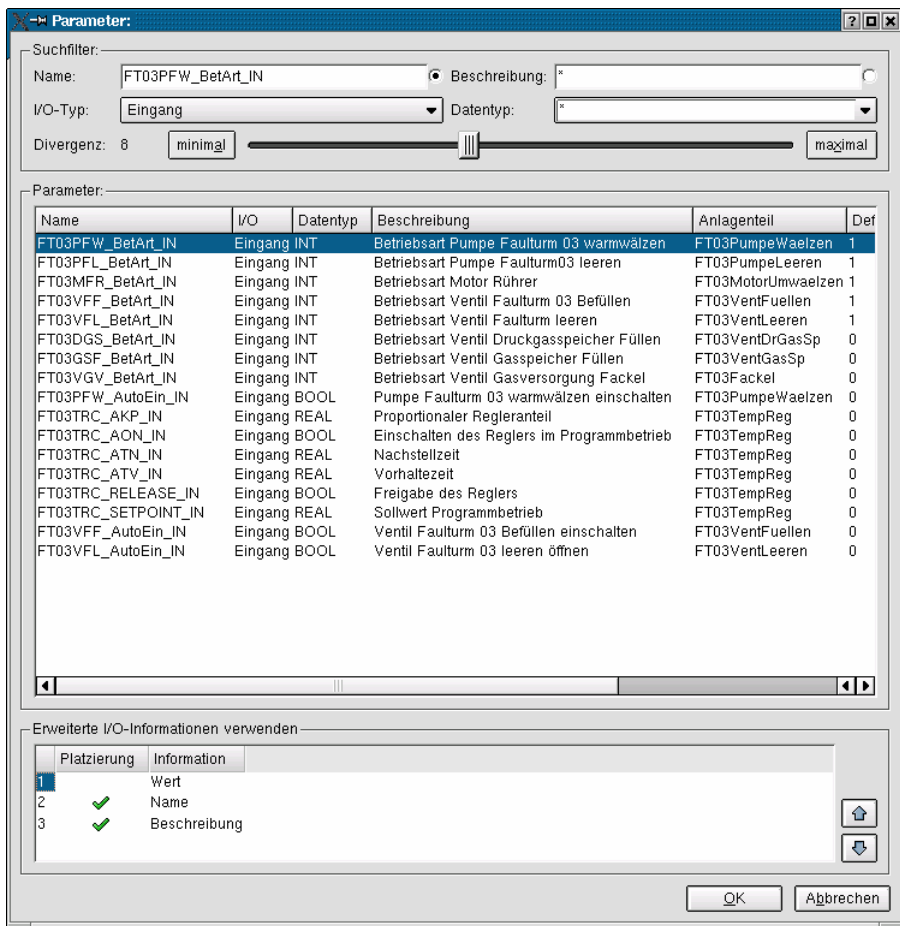


图 71: 获取参数菜单

3.2.1.10 察看属性/从图表里删除I/O

功能图表里的输入输出点被占用后，可以通过快捷菜单来删除。通过 "属性" 可以查看相关信息。下图显示了一些属性。

Eigenschaften (Gateway-I/O)

Name: GW1_PV0

Beschreibung: Demo-PV

Datentyp: BOOL

CPU: R001/PLS03

Task: #GW1

Wird in folgenden Plänen benutzt :

Plan	I/O	CPU	Taskklasse	Task
Hardware/Rechner/GateWay	Ausgang	R001/PLS03		#GW1
Leitsystem/Pläne_S/TestGateway	Eingang	R001/PLS01 1		GLH_1
Leitsystem/Pläne_S/Werte	Ausgang	R001/PLS01 1		GLH_1
Leitsystem/Pläne_S/Werte	Eingang	R001/PLS01 1		GLH_1

Ok Abbrechen

图 72: 功能图表输入点的属性

显示的信息取决于图表框里的类型 (硬件 I/O, 连接变量等)。下面就是实例里的信息列表:

- ✓ 变量被使用的图表,
- ✓ 是否可以插入到输入或输出区,
- ✓ 所赋给的物理视图(CPU),
- ✓ 任务级别和任务。

3.2.2 添加和放置功能模块

功能块是从库里面拷贝并放置到功能图表里。在项目库里应该已经有需要添加的功能块。

在库这一栏，在 **CaeManager** 的导航区域里可以选择功能块组或单个的功能块。提示工具显示了需要的库，组，或者模块的简要信息。如果需要了解此功能块的相关信息，可以通过 "Help" 来查看相应得文档。

在拷贝功能块后，移动鼠标到功能块要添加的地方。



请注意用快捷菜单插入时，最后一个模块不能插入。插入模块快捷菜单会列出已经复制的模块列表

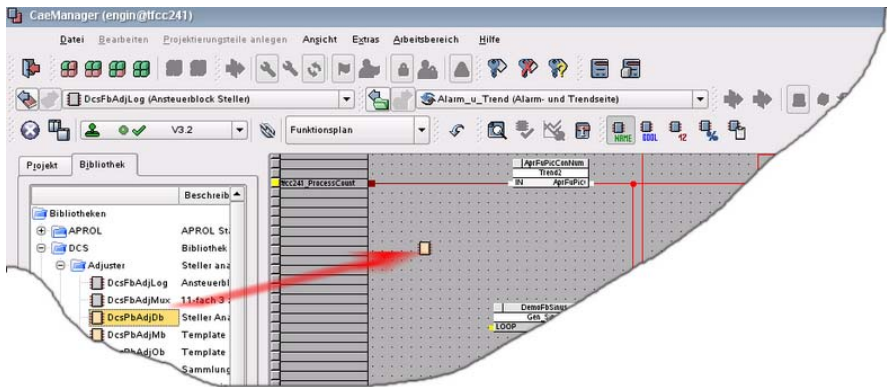


图73: 使用 drag & drop 插入模块



功能块所在的位置应确保直接并不阻挡逻辑的流入和流出。这样功能图表的可读性比较好。

注意功能块之间在以后还可能再创建!

连接只能放置在显示的网格系统中!



注意删除功能块的时候相应的连接线也会被删除!

选择功能块的快捷菜单有两个选项。不同的选择类型具有不同的功能。下面只描述 "属性" 的功能，其它功能如字面意思。

选择的类型：

- ✓ 用鼠标选择这个块，鼠标右键确认。快捷键显示选项功能。
- ✓ 有鼠标左键点击功能块，功能块就会高亮。此时右击，快捷菜单提供更多的功能选择。

功能块在一个功能图表可能会使用多次，因此每个功能块需要一个唯一的名字。在功能图表里放置一个功能块后，在原有实例名的基础上自动添加了下划线和一个数字。如果此功能块再次出现，数字就自动增 1。



插入功能块时，并不是拷贝要插入的功能块，而只是在数据库里创建这个功能块的参考并配置个别参数(实例名, I/O 常量等)。当使用面向对象语言编程时，就意味着：**选择的类的一个实例被创建和配置。**

重要的是要知道只是创建了一个实例而不是拷贝。**如果类修改了，相应的实例也会修改。**

在库里修改一个功能块会影响到项目里所有的此功能块。在库里修改功能块时，可以不改变原始的功能块，而是修改它的拷贝的名字。

更多的信息参见 *CAE 库工程设计* 文档。

3.2.2.1 通过“属性”配置功能模块

所有功能块，即使是超级宏，也可以通过快捷菜单的 "属性" 来配置，示例如下图。

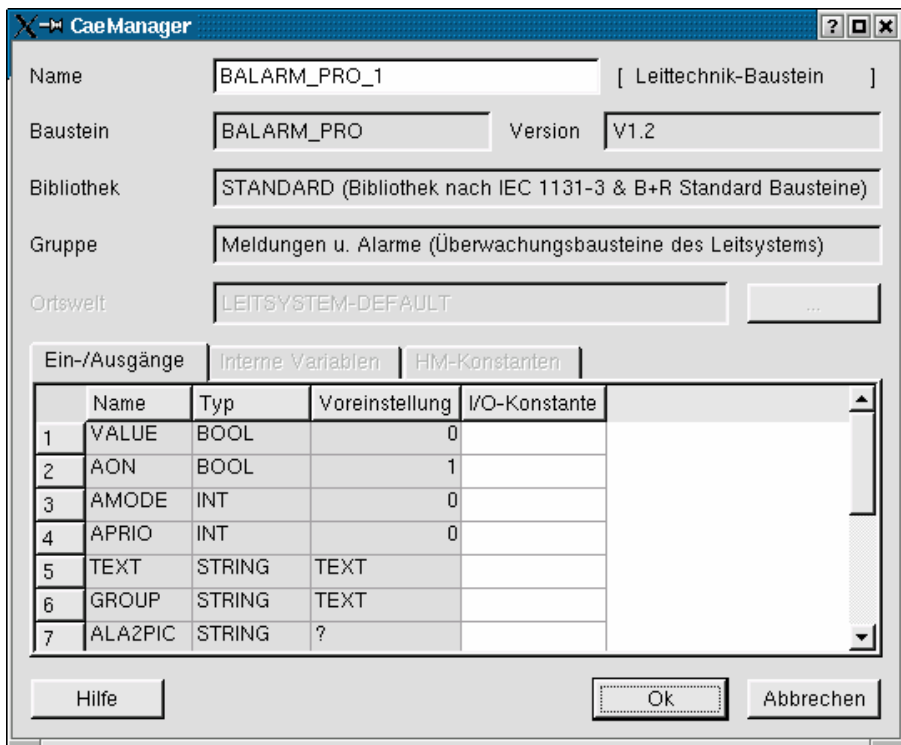


图 74: 配置功能模块菜单

I/O 常量使用 "输入" 栏输入用 **[返回]** 来确认。功能块的值通过 **[确定]** 确认。



注意字符串不能超过 64 个字符!

对此类型配置:

- ✓ 明确定义功能块的物理视图。
- ✓ 编辑功能块的常量输入。原则上，这些 I/O 常量通过"输入" 栏来输入。
- ✓ 功能块的实例名会自动修改。名字在功能图表里必须唯一并符合 ANSI-C 语法。



"内部变量" 和 "超级宏常量" 是用来配置超级宏的。

某些时候多个功能块 (如上面列表显示) 必须在同一个物理视图执行而不仅仅是同一个功能图表。这样, 在菜单里要重新定义物理视图。功能块的左上角就有一个红色标记来识别。这些功能块只在单个的物理视图里执行 (如控制系统功能块)。一旦物理视图设置后功能块就被放置在上面了。



任何没有被使用的输入也必须在库里设置。如果输入没有连接也必须输入一个有意义的常量!

那些需要常量赋值的功能块的管脚需要有相关的管脚描述信息。这些描述可以通过菜单里"帮助" 项显示出来。

3.2.2.2 功能图表里的报警功能模块

此功能块是用来在 **DisplayCenter (AlarmMonitor)** 或者 **AlarmViewer** 存储和显示报警信息。请根据功能块描述确认那些值需要填写或那些管脚需要连接。



注意大多数管脚都是常量, 在运行时是不能改变的。报警和信息系统的数据库启动时配置的就是这些管脚的值。



多路报警功能块的 **VALUE** 管脚(激发报警)不能并行连接! 必要的话, 需要用一系列的 **PIN_CON_BOOL** 或 **PIN_CON_NUM** 功能块连接到每一个变量的输入。此功能块在 **STANDARD** 库/控制系统组。

这里只涉及定义操作员架构和提高报警识别的输入和输出。现在描述的是最新的 **BALARM_EXT**。此功能块的物理视图是控制系统。

下面有此功能块的三种视图: 管脚名, 管脚类型和最重要的管脚的赋值。

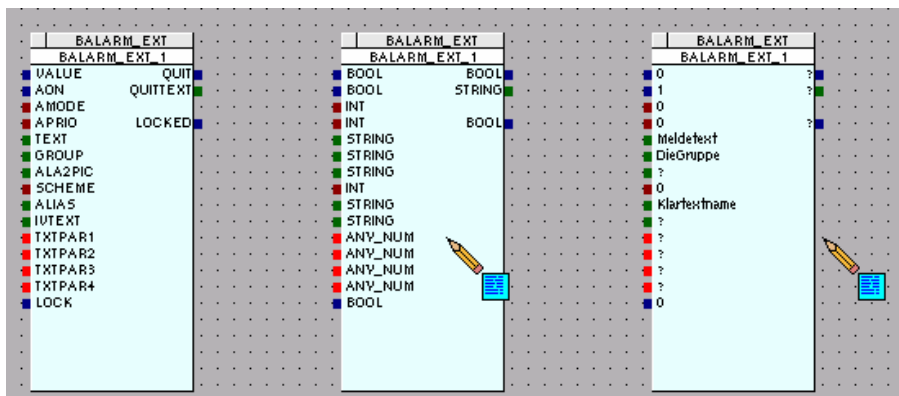


图 75: BALARM_EXT 功能模块的三种视图

输入	说明
VALUE	报警通过此值触发。
AMODE	定义报警类型。定义数据类型。 参见下表！
TEXT	显示在 AlarmMonitor 里的信息文本。
GROUP	组名字。 通过合适的授权管理这些报警。
ALIAS	显示在 AlarmMonitor 里的报警描述。 此名是与变量结合在一起赋给系统的，只有有经验的系统管理员才能做。
LOCK	用来锁定报警的信号(1 = lock)，例如不是来自 AlarmMonitor 而是来自过程图。
IVTEXT	任何时候出现一个报警，可以让操作员去选择下一步操作的文本(HTML 格式)。 可以存储这些页(对每一个重要报警有一个独立的页) 到工程师系统的本地目录并输入一个名字。 \$HOME/ENGIN/WEB/<CountryCode>/PROJECTS/<ProjectName>/IVTEXTS



锁定输入不使用报警组权限来检查。设计者有责任位置个输入添加操作权限! 设计者同时要确保次输入的信号发生也同样有相应的权限保护!

输出	说明
锁定	如果 LOCK = 1 或者警报监控器的警报被锁定，那么 Signal = 1
激活	Signal = 1 if alarm pending and 0 if alarm no longer pending. Independent of LOCK or acknowledgment.

下面的表显示 AMODE 和 SCHEME 组合的所有可能性以及相应的解释。

AMODE	0 SCHEME	1 SCHEME	2 SCHEME	3 SCHEME
0	信息	在 alarm monitor 中有报警显示	打印信息	在 alarm monitor 中有报警显示同时打印信息
1	信息	在 alarm monitor 中有报警显示	打印报警	在 alarm monitor 中有报警显示同时打印信息
2	在 alarm monitor 中有报警显示同时需要确认	在 alarm monitor 中有报警显示同时需要确认	在 alarm monitor 中有报警显示同时需要确认	
3	在 alarm monitor 中有报警显示同时需要文本确认	在 alarm monitor 中有报警显示同时需要文本确认	在 alarm monitor 中有报警显示，打印报警，同时需要文本确认	在 alarm monitor 中有报警显示，打印报警，同时需要文本确认

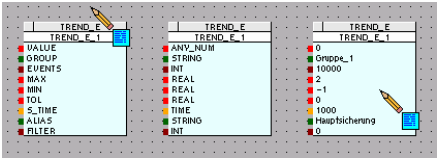
3.2.2.3 功能图表里的趋势功能模块

此功能块用来在 **TrendViewer** 和 **TrendMonitor** 里存储和显示趋势。请根据功能块描述确认那些值需要填写或那些管脚需要连接。此功能块的物理视图是控制系统。



注意大多数管脚都是常量，在运行时是不能改变的。报警和信息系统的数据库启动时配置的就是这些管脚的值。

此处文档只涉及基本项目设计中需要用到的输入。现在使用最新的功能块 TRENDE。



下面有此功能块的三种视图：管脚名，管脚类型和最重要的管脚的赋值。

图 76: TRENDE 功能模块的三种视图

输入	说明
值	使用此输入来记录值。
组	定义组。组在后面用来作为一个选择的标准。
最大值	在配置趋势显示时能接受的最大量程值。在配置是可以修改。 所有值都被记录。
最小值	在配置趋势显示时能接受的最小量程值。在配置是可以修改。 所有值都被记录。
事件	定义在数据库里记录的数量。定义多少个过程变量应该被存储，直到第一个值被覆盖 (环内存)。
别名	输入简洁的文本名字来提高可读性，也可以使用过程变量的名字。

3.2.2.4 趋势显示配置 (runtime)

趋势是在操作员系统使用 **TrendViewer** 来显示的。 在一个显示画面上最多可显示 8 个变量的趋势。 有合适授权的操作员可以组合这些趋势。 但是，最初是在工程师系统里来组合这些趋势。

趋势以组结合在一起，它们的结合描述如下。 这些组可以以任意方式组合。 意味着组是独立于趋势功能块的(功能图表里的趋势功能块)。 在组里单个的趋势也可使用。

在启动 **TrendViewerConfig** 后，首先点击 [Project] 按钮，然后是[Data] 按钮。 配置就像下面显示的菜单一样执行。

按钮/输入	说明
组	在"组"里，选择事先在趋势功能块里定义的组。
趋势	在一个组里的所有过程变量都显示出来以便选择。
S	在趋势功能块里的值的下限被此值覆盖。
E	在趋势功能块里的值的上限被此值覆盖。
单位	可以输入值的单位。
载入	在配置完成后可以保存在这里。在运行时可以下载。

 显示趋势时的线型，线宽，颜色等，可参见[过程交互控制手册\(趋势系统\)](#)。

3.2.2.5 功能图表里的超级宏

超级宏是一个和功能图表相似包含复杂的逻辑关系的功能块。超级宏通过输入来控制。超级宏的结果传输到输出。

 超级宏的多层嵌套功能从**APROL**发行版本 2.0 开始。

快捷菜单的 "查看" 用来显示功能图表包含的超级宏的逻辑流。基于这个超级宏的所有属性都可以修改。包含在这个超级宏里的超级宏也可以以同样方式显示。

×

CaeManager

✕

Name

MAKRO_2_1

[Hypermakro-Baustein]

Baustein

MAKRO_2

Version V1.6

Bibliothek

TEST_DOKU

Gruppe

HYPERMAKRO (Test von Makro in Makro)

Ortswelt

...

LR-Ortswelt

...

Ein-/Ausgänge

Interne Variablen

HM-Konstanten

	Name	Typ	Voreinstellung	Konstante
1	STA	BOOL		

Ein-/Ausgänge

Interne Variablen

HM-Konstanten

	Name	Wert
1	\$EIN	

Ein-/Ausgänge

Interne Variablen

HM-Konstanten

	Name	Typ	Voreinstellung	Konstante
1	E_A	BOOL		0

左图显示了所有选择的选项。下表是描述。只解释菜单栏部分。

图 78: 超级宏属性

窗格	说明
输入	输入不能嵌套而且要赋予有意义的常量。
内部变量	这些变量从 APROL 发行版 2.0 以后不再使用。在旧版本里是超级宏的辅助变量。
HM 内容	超级宏的大部分输入是赋予不同的常量,主要取决以应用。超级宏提供了存储这些常量的选项。为了避免超级宏有更多的管脚,这些超级宏常量之显示在此栏。

超级宏创建后可以在不同的功能图表里重复使用。然而，超级宏也可能包含具有操作权限的显示驱动功能块，也必须调整它们到相应的条件。不同的权限可以赋给特定的操作动作。

在这种情况下，通过使用快捷菜单项“**操作者权限**”来修改超级宏里的权限。此权限可以决定项目里那些改变是可用的。

B2 项目组态

3-42

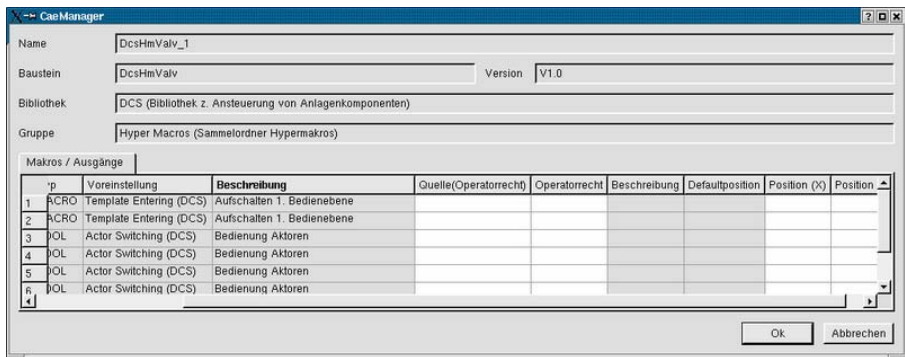


图 79: 修改超级宏显示驱动功能模块的权利和位置

所有可用或仍在创建中的条目描述如下：

输入	说明
图片模块	显示驱动功能块在超级宏里的名字。
名称	可以修改的管脚名。
类型	管脚类型
默认设置	默认的设置。但是，可能已经事先设定了改变。 A. 只是采用功能块的默认值。 B. 创建超级宏时用来覆盖" A "。 同样有超级宏库的权限。
说明	默认情况下的描述。
资源	如果想更改权限，用此项来选择源的权限。
操作员权利	用来选择预先设置了源的权限。 在此区域双击即可选择。
说明	新选择权限的说明。
默认位置	此条只显示输出是否启动一个显示驱动功能块。因此，如果输出类型是宏时只有一个条目。 因为这是默认的，"Default setting"中的描述 a) 和 b) 同样设用于这里。 唯一的不同在于此条的位置。
位置 (X)	输入新 X-轴。
位置 (Y)	输入新 Y-轴。



有关位置的重要信息:

所有位置的输入使用与 DisplayEditor 一致的全局变量。X-方向为 0-32000, Y 为方向 0-24000。

假如没有默认的位置或新的位置还没有被输入时:
启动的图片宏的左上角会显示鼠标指针。

已经或将要输入的是一个绝对位置。在过程图里**左上角就是零点**。

3.2.3 功能模块的逻辑连接

连接放置在功能图表里的功能块就是在功能图表里组态。

 左边显示的**[连接线]**按钮用来画连接线(如 功能块之间的逻辑连接)。

用鼠标左键点击很容易选择要连接的点。系统会自动选择这些点最优化的连接。直接在功能图表里通过点击来确认连接路径也行。特别是在连接路径比较长的时候比较有用。在连线上可以插入新点作为输入分支到其它功能块。



输出类型只用和输入一致时才能连接在一起! 如果要连接**不同的类型**, 必须在连接中插入**转换功能块**。

在功能图表里选择管脚的不同视图是非常有用的。

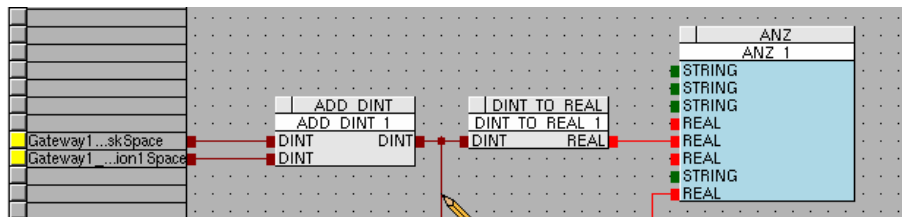


图 80: 通过转换器连接

上面是两个 DINT 变量的加法。加法的结果要在画面上显示。然而, 超级宏显示(ANZ)是 REAL 变量。因此, 必须插入 DINT_TO_REAL 类型转换功能块。



插入转换功能块时，要考虑到变量范围和转换规则(C-syntax)，这样才不会丢失信息！

在很大的功能图表里尽管连接存在但要找到使劲连接点并不是很容易。要快速知道在连接线上的输入和输出点，可以通过快捷菜单项 **"属性"**。首先点击选择连接线,然后通过右击选择属性项。选择源或目标连接线时候，就可以直接到达功能图表里该功能块。

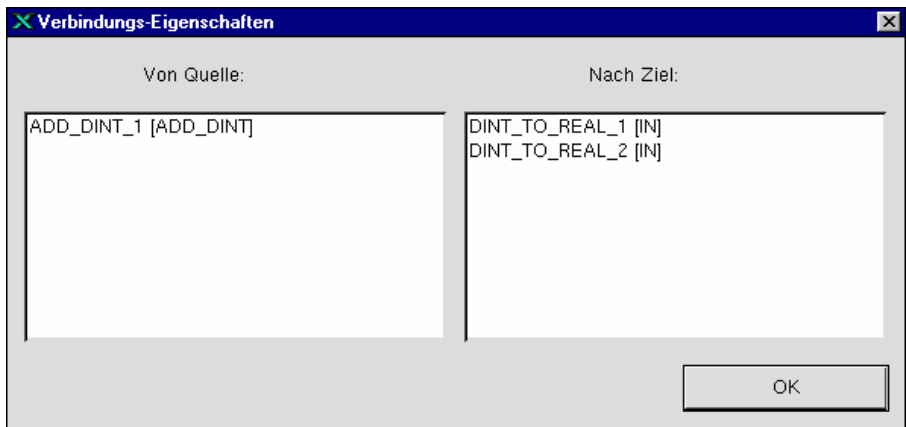


图 81: 线的连接属性

也可以通过快捷菜单项**删除**和**移动**连接线。



如果在功能图表里面需要移动功能块到不同的位置, 连接线会由 **Autorouter** 重新建立。在选择不用的坐标后也同理。你会得到一个新位置。

在功能图里也可很方便移动连接的输入或输出点。把鼠标移动到此表的管脚上 (不是功能块的连接点)。点击移动。然后选择一个新的可用的位置。再通过鼠标点击就可以完成。自动路由就会重新优化功能块和这个管脚之间的连线。

3.2.3.1 功能表中功能模块之间的连接

与 I/O 条与功能块的连接类似下表列出了功能块之间允许的连接。可能的物理视图列在轴上。 请注意： *最坏的情况*已列出，但这种组合不应该出现在功能图表里。

来自不同控制器的功能块应该通过连接变量来连接。

Function block / Function block	Chart default	controller-1 TC1 Task[1]	controller-1 TC1 Task[2]	controller-2 TC1 Task[4]	controller-2 TC1 Task[5]	controller-2 TC2 Task[6]	CC Task[1]	CC Task[2]
Chart default	Yes						Yes	Yes
controller-1 TC1 Task[1]		Yes	Yes				Yes	Yes
controller-1 TC1 Task[2]		Yes	Yes				Yes	Yes
controller-2 TC1 Task[4]				Yes	Yes	Yes	Yes	Yes
controller-2 TC1 Task[5]				Yes	Yes	Yes	Yes	Yes
controller-2 TC2 Task[6]				Yes	Yes	Yes	Yes	Yes
CC Task[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CC Task[2]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

3.2.4 显示驱动功能模块的附加属性

显示驱动功能块可以描述为：通过 **DisplayEditor** 创建的代表动态图片宏的图片模块。这类图片宏以动态化为目的。必须通过显示驱动功能块才能在功能图表里连接在一起。显示驱动功能块通过它的管脚来连接。

通过随后的显示驱动功能块的选择可以再过程图里插入一个或多个图片宏。在这一步，显示驱动功能块显示带名字的位置和功能图表的信息。因此当过程图动态化时必须记录功能图表里显示驱动功能块的功能。

唯一的名字和实例名就用来分配给上面提到的定位。在功能图表里是唯一的并通过快捷菜单的“属性”来显示。

上面提到的图片宏可能是简单的数字显示，液位显示或复杂的操作模版。

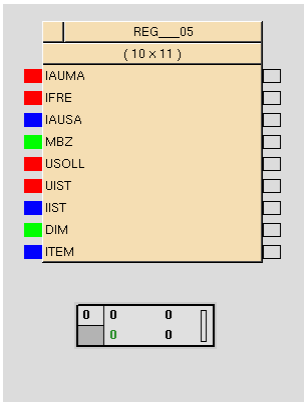


图 82: 具有显示驱动功能模块的图片宏

左边显示的是具有动态图片宏的显示驱动功能块。这个图片宏在运行时根据功能图表的输入显示相应的值。

此过程与操作模版相似。在功能图表里首先要创建逻辑操作或在运行时通过模版来控制相应单元的超级宏。

动态的显示驱动功能块可以被看作是没有控制逻辑的操作或显示单元。控制逻辑是在功能图表或超级宏里完成的。一般来说，显示驱动功能块是在控制计算机里运行的。

*在功能图表里插入 **超级宏**时，同时也包含了显示驱动功能块。在这种情况下，它们跟显示或控制命令已经连接在一起。*

一个显示驱动功能块也代表一个简单的控制器(参见 [在过程图片里插入操作选项](#))。在过程图里可以通过这个控制启动操作模版。

3.2.4.1 配置显示驱动功能模块

在使用显示驱动功能块时，通过快捷菜单可以重置默认值。通过**操作者权限**重新设置权限和**动态图片宏**显示的位置。

属性

对此菜单项，重要的是要知道显示驱动功能块，预设的文本是用来代替随后在动态图片宏里输入的名字，或赋给输入的常量。输入窗的功能类似输入超级宏属性的输入窗(参见 [功能图表里的超级宏](#))在具有相应的显示功能块的图片下是可见的。这样我们不需要更多地解释。

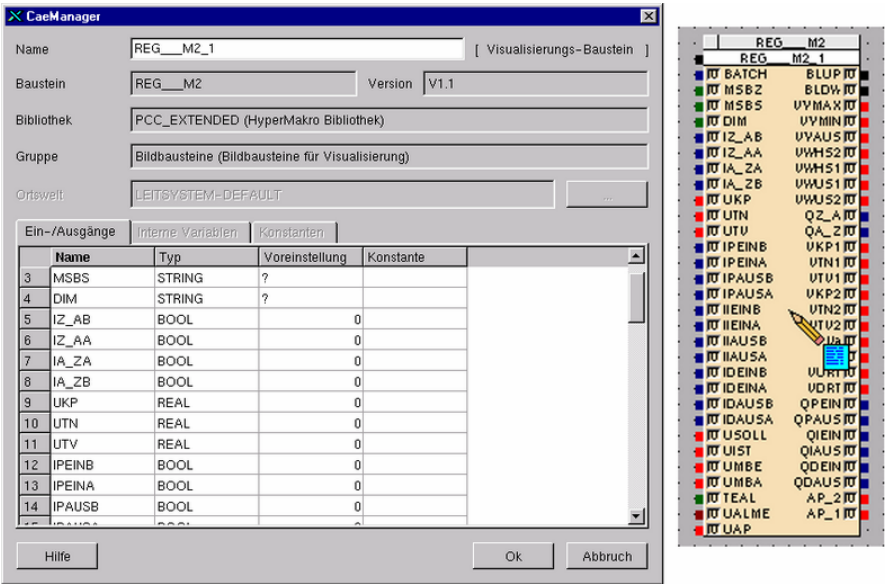


图 83: 显示驱动功能模块的属性

操作权限

此处重要的是显示驱动功能块的操作权限动态图片宏的位置是可以重新设置的。 请注意： 在这个菜单的操作权限和超级宏菜单里的操作权限很相似。 我们不需要做更多的解释。

下图显示的是更复杂的功能块。 此功能包含几个控制面板。 当鼠标碰触其中的一个面板时图表就会显示相应的光标的变化。 点击其中的一个面板会触发一个脉冲给输出。 相应的权限赋给这个动作，因为可以用来操作一个阀。

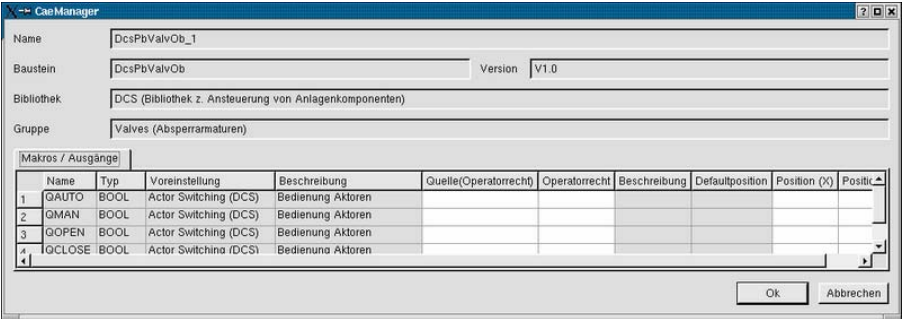


图 84: 显示驱动功能模块的操作菜单

如果一个显示驱动功能块启动另一个显示驱动功能块 (动态图片宏)，可以通过超级宏的管脚类型改变位置。 更过信息参见 功能图表里的超级宏。

3.2.4.2 使用PIC_SCALE功能模块

通过一个实例来描述 *PIC_SCALE* 功能块的使用。 有相当的显示驱动功能块由多个显示输入值从而组成如棒状条或位置标记。

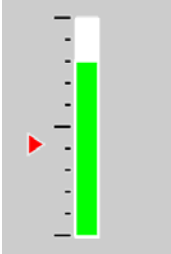


图 85: 动态图片宏

如果 `PIC_SCALE` 功能块不添加的话，则上面的元素的量程从 0 到 32767 使用 `PIC_SCALE` 功能块就可以使图片宏里的显示驱动功能块的输入显示更低量程的值。因此上面提到的每个功能块的输入都要连接一个 `PIC_SCALE` 功能块。



在连接 `PIC_SCALE` 功能块时，必须直接连接到显示驱动功能块的输入！

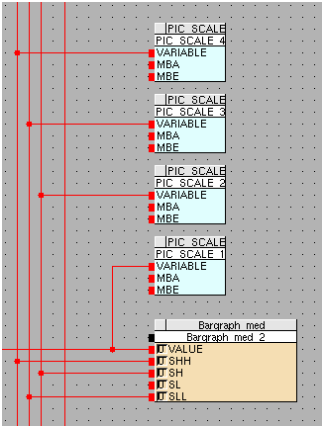


图 86: 带 `PIC_SCALE` 功能模块的图片模块

左边的功能图表部分显示了相互的连接。

棒状条(值)和几个指示箭头(S*) 将由图片宏里赋给输入的显示驱动功能块来控制。

MRB 和 MRE 都是常量并需要赋值。在运行时是不能修改这些值的。

常数	说明
MRB	定义动态化的零位置。零位置对棒状条来书就是"无棒状条"。对指示箭头来说，这个位置相当于 "图片宏绘制的起始位置"。

常数	说明
MRE	定义动态化的终点位置。此位置对棒状条来说 "填满"。对指示箭头来说就是事先设置的动态化的位置。

系统会处理中间的标定(线性化)。



"窗口显示" 也可以使用此方法来配置。MRB 和 MRE 的值就是希望显示的那部分窗口。

3.2.5 功能图表导航

在 CaeManager 里打开一个功能图表，标准的显示大小为 A4 纸张的 90%。"水平" 滚动条用来浏览水平方向其余部分。"垂直" 滚动条用来在垂直方向浏览。A 在激活"视图 / 全屏"可以显示 A4 大小的画面。

在所有页中导航

在多页的情况下有两个选项用于浏览：

选项 1

按下[Ctrl]键同时使用鼠标可以移动功能图表到相应位置。通过滚动条来查看边界部分。

选项 2

通过 "视图/ 导航" 或点击相应的按钮来打开功能图表的导航。在鼠标选择功能图表后，可以通过按下[ctrl]键来启动导航。此时就可以显示这个功能图表的示意图。通过 "全屏" 可以显示完整的图表。

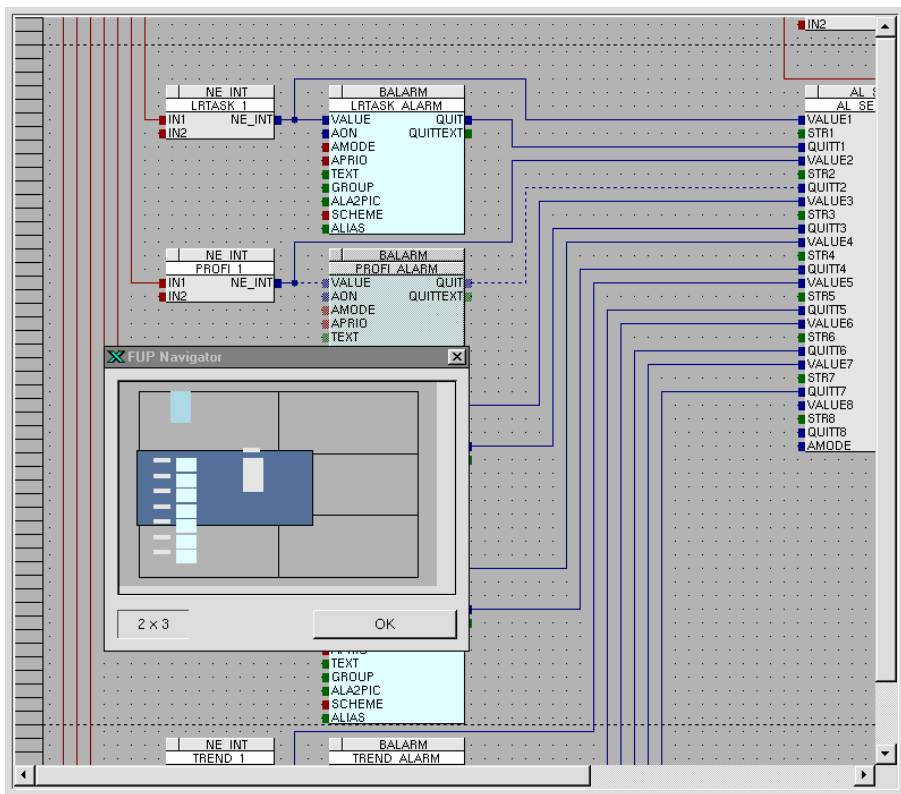


图87: 使用FUP 导航工具导航

使用鼠标左键点击并拖曳灰色区域到相应的位置。然后相应的这部分区域就会显示出来。通过滚动条来查看边界部分。

3.2.6 功能图表注释

除了维护注释页，在功能图表里也需要提供有助于组织和清晰的注释。

APROL 给功能图表提供了几种注释选项。

选择[Commentary]按钮。在功能图表里使用鼠标左键并点击来确定注释添加的地方。一个文本输入框会弹出用来输入文本。输入的文本有三种大小可选

(小, 中和大)。从预先定义的位置起, 文本是左对齐方式, 可以覆盖在线和功能块上面。下图显示了相应的操作和结果。

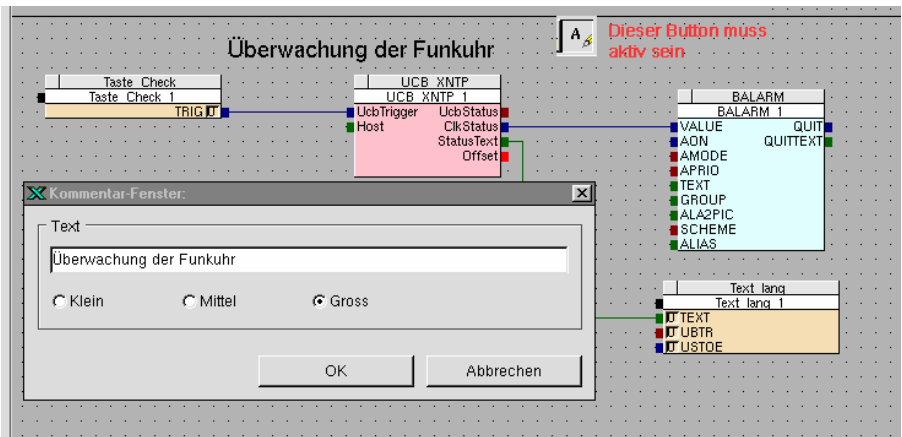


图 88: 在功能图表里添加注释

注释可以按如下步骤删除:

激活[Commentary]按钮并使用鼠标选择需要删除的注释。然后注释就出现在能被删除的注释窗里。

B2-3.3 为功能图表生成任务

功能图表是以图片描述性编程来组成过程控制系统的。功能图表在激活和编译成功后就可以在过程控制系统里执行。



注意系统不检查逻辑错误！设计者必须对要执行的逻辑结构负责。

然而, 控制计算机或控制器要生成一个任务只是激活和编译的话还是不够的。还必须为相应的设备和控制计算机生成任务。

要做到这一点, 从快捷菜单中选择相应的硬件。选择快捷菜单中的 *声明*。生成控制器任务的菜单如下显示。

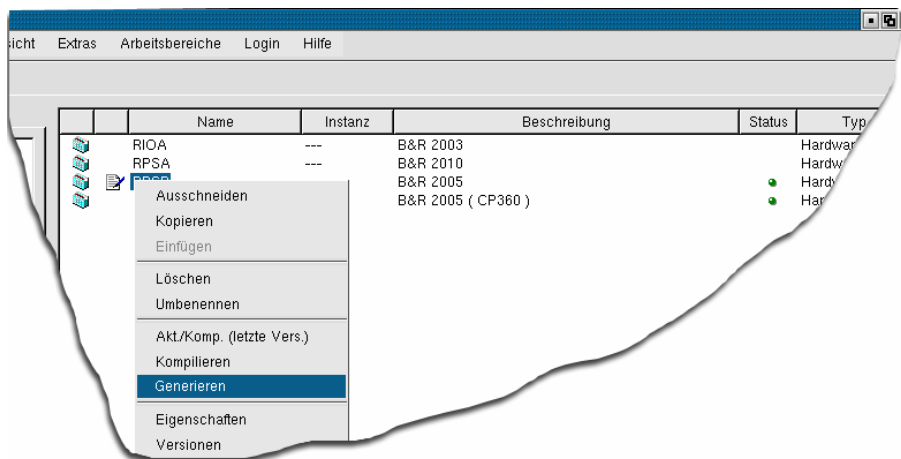


图89: 选择要生成任务的硬件

下面的菜单就会显示出来。系统只知道已经定义的任务。但不知道功能图表所在的那个任务。如果生成一个不包含功能图表的任务，那么就会生成一个空任务。

所需条件： 每个任务包含有功能图表。在这种情况下可以把所有任务移动到右边的窗口并开始生成。每个任务是单独报告并显示可能出现的错误。在点击[OK]后才会生成下一个任务。

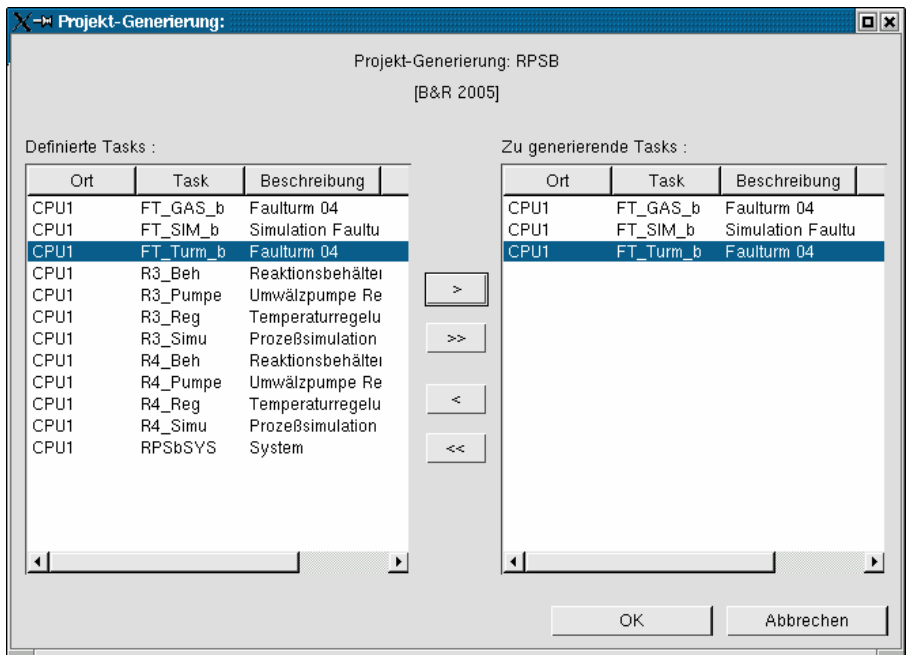


图 90: 选择任务并开始生成

生成需要比较长的时间，可以选择选项只生成那些功能图表已经修改的任务。这样可以节省时间。

下图显示任务生成结束的过程。如果有错误，可以通过滚动条来浏览。

下图显示的生成过程没有错误发生。

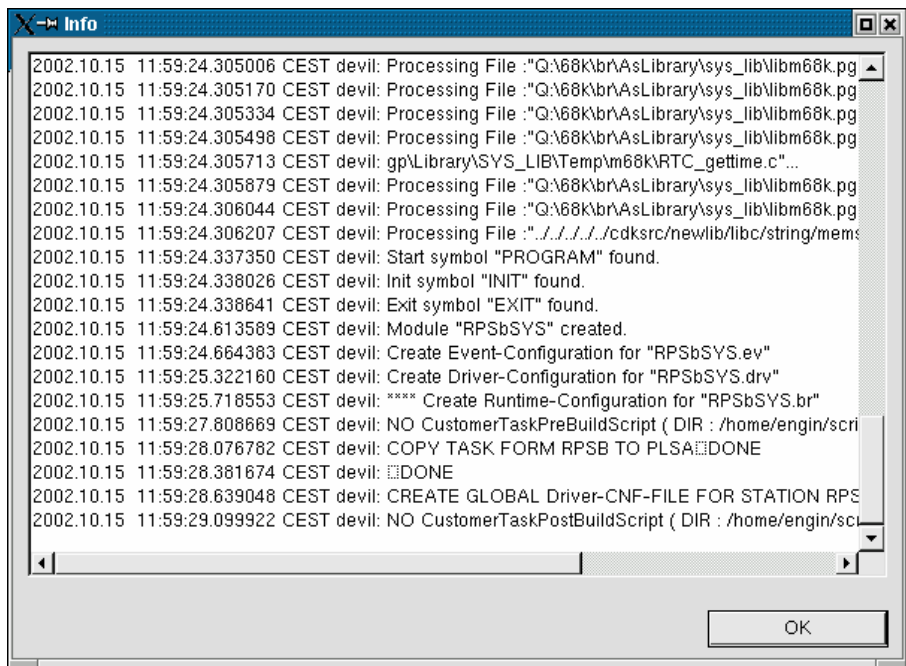


图91: 生成完成

B2-3.4 调试功能图表

功能图表调试前必须满足如下需求：

- ✓ 在调试期间必须启动校验栏。
- ✓ 项目要激活和编译。
- ✓ 功能图表所在任务必须要生成。
- ✓ 任务要下载到控制计算机或控制器并启动。

在功能图表启动后按下[Debug mode]按钮，此时功能图表就会显示动态改变的值。



如果在运行计算机打开 **CaeViewer**，此时也可调试功能图表。但鼠标移动到动态区域，右击选择调试选项。功能图表的动态区域部分就会显示出来。

运行系统从 KDE 菜单 **APROL/Diagnose/CaeViewer** 可以打开 **CaeViewer**。在选择窗口，可以选择项目里的任意功能图表区域双击来调试。



功能图表是以树状结构来排列的。
但是只显示包含功能图表文件夹。

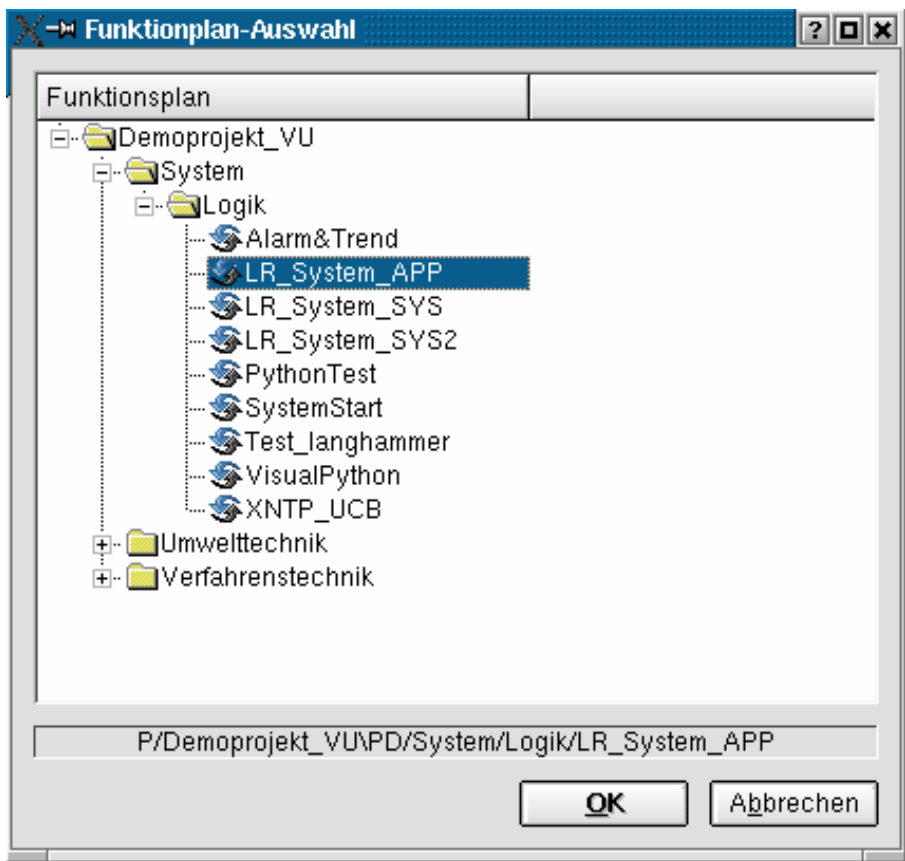


图 92: 在 CaeViewer 中选择要调试的功能图表

B2-3.5 打印功能图表

在 **CaeManager** 使用**文件/打印**菜单选项可以打印功能图表。

打印选项对话框可以作如下设置：



打印预览

此选项显示打印区域的预览。在点击<**OK**>确认后，KDE 应用的 *KghostView* 就启动了。*KghostView* 可以在多页功能图表导览，缩小，放大。

打印对话框也可以直接从 *KghostView* (从菜单或工具条)打开。在选择理想的打印模式后，在"**当前打印系统**"的下拉菜单中要选上 "CUPS Common UNIX Print System"。



添加封面

添加一张封面到打印区。

如果打印之前不要预览，可以通过**打印选项**对话框来配置。*kprinter* 的打印菜单就会显示出来。

在选择理想的打印模式后，在"**当前打印系统**"的下拉菜单中要选上 "CUPS Common UNIX Print System"。

B2-4 过程图

B2-4.1 动态过程图基本信息

过程图就是操作者和过程控制系统之间的人机交互界面(HMI)。随着图片宏插入，每幅图上会显示所有有关运行过程的必须的信息。如果操作员有相应的权限，则可以介入到过程图中作某些设置或修改。

下面的步骤简要说明了如何创建一幅过程图：

步骤	说明
步骤 1	在项目里创建 <i>过程图</i> ： →→导入先前导出的静态图到 DisplayEditor →插入一幅背景图。
步骤 2	在 DisplayEditor 项目中创建一个过程图的静态部分。
步骤 3	粘贴 <i>过程图</i> 到选定的静态图上。

B2-4.2 创建过程图

在 **CaeManager** 里打开菜单项*创建部件/过程图*。输入名字和描述后，在你添加的目录里面就生成了一个空的过程图。

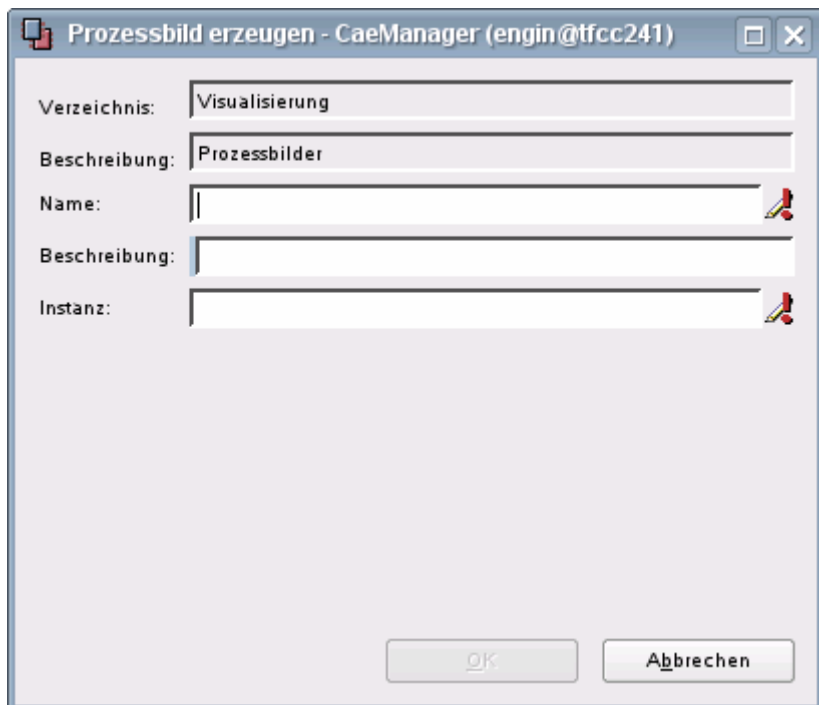


图 93: 创建过程图



APROL 命名规则在为过程图命名时必须遵循。

创建过程图后，你可以在 **DisplayEditor** 中创建其中的静态部分。



静态图应该包含能表现 **P&ID** (管道和仪表图) 的静态部分！

根据 **P&ID** 图和相应的系统功能的划分，整个生产过程可以被明晰的分成几幅能联成一体的过程图。

也可以把一幅过程图分解成几幅子图。运用方便的放大功能，这样可以在一个非常大的系统里面不切换画面就能执行相应的过程控制。

图片的数量没有限制，只要硬盘容量足够就行。

使用 **DisplayEditor** 和提示工具可以快速有效的创建过程图的静态部分。通过 **插入/ 静态图片宏**，可以插入已经创建的图片模块静态部分。你还可以绘制过程图的静态部分，或者使用图片工具合并已经创建的图片。

静态图包含所有的过程图素，不包含动态图素，如罐，线，设备，信号线。

组合背景图片到静态图：

组合一幅图片(如照片或 P&ID)到静态图。

首先打开 **DisplayEditor** 的菜单工具/背景/插入图片。

在选择框里面可以选择在文件系统里面不同图片格式的图片作为要组合的对象。

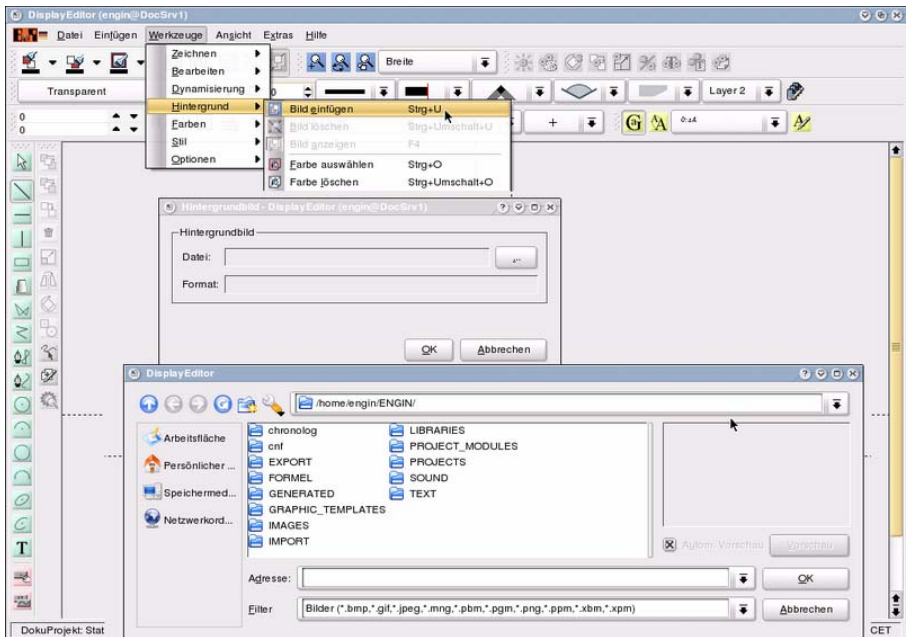


图 94: 插入背景图



建议使用 **PNG** 格式的图片。



不管是哪一种图片格式，能被组合的图片的大小不能超过 1MB！

创建静态图时可指定最终过程图的设计。通过**插入/静态图片宏**，可以把库里面的动态图片宏或事先做好的模版作为静态图的一部份添加到静态图上来。



如果过程图在 **DisplayCenter** 中使用直接图片调用，请注意以下信息！

在静态图里，动态只允许使用在直接图片调用控件的时候。

在 **DisplayEditor** 里面相应配置这个控件的对话框需要填写在 **DisplayCenter** 里调用的图片实例名。



在项目计划阶段根据现有的 P&ID 图可以先建立过程图的清单。建议同时在清单里面输入过程图的实例名这样，可以立即开始图片连接。

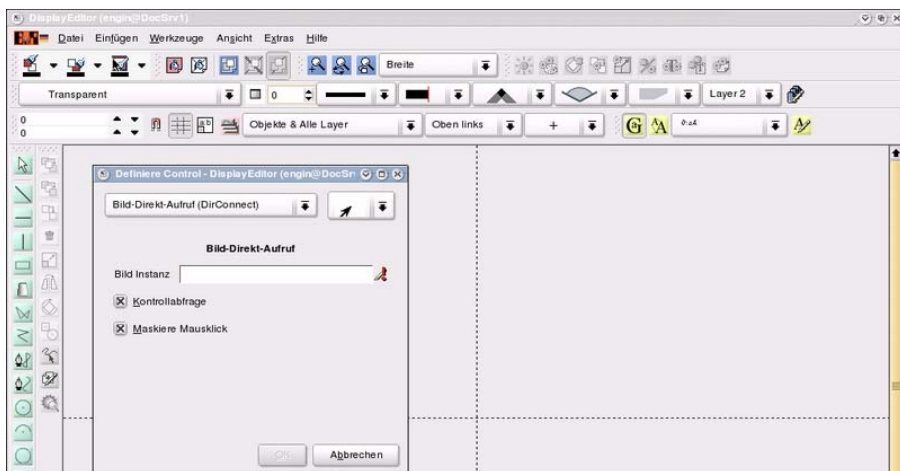


图 95: 创建直接图片调用

4.2.1 插入动态图片

只有在过程图里面插入动态图片宏，在 **DisplayCenter** 里打开过程图时才能与生产过程建立可视化的连接。在过程图中插入动态图片发生在 **CaeManager** 中。

显示驱动模块和相应的图片宏的连接在显示驱动功能块的其它属性一章中讲述。

"显示驱动功能块" 栏列出所有被激活和在当前项目的功能图表里使用的显示驱动功能块。

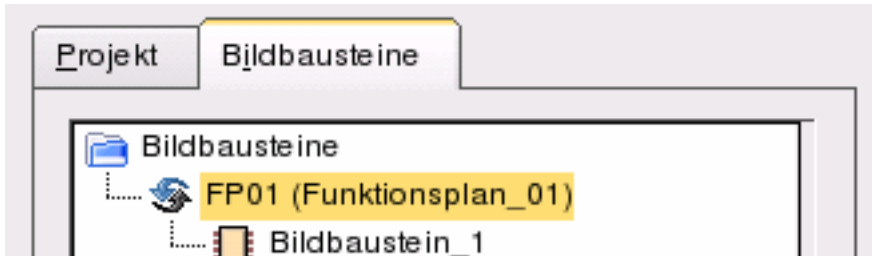


图96: 指定图片功能模块

选择一个功能图表后,包含在里面所有的显示驱动功能块和超级宏都会显示出来。另外,超级宏被选上后,包含在里面的所有显示驱动功能块也会显示出来。这样,项目里面的所有显示驱动功能块都是可以使用的。提示工具显示了例子,名字以及图片模块或者功能图表的描述。

在功能图表里,选择所需的显示驱动功能块(通过快捷菜单的拷贝功能)并把它插入到过程图里。然而,在这种情况下不是插入显示驱动功能块而是插入动态图片宏! 你可以插入相同的模块很多次。



当使用快捷菜单插入模块时,并没有插入最后一个模块,只是通过快捷菜单复制了一个模块。

动态图片宏在插入后可以移动到任意位置。过程图在激活和编译后才能被下载。这样在操作系统的 **DisplayCenter** 中就可以看到那些动态的图片宏的效果。

下图显示了标签 *preview* 的快捷菜单:

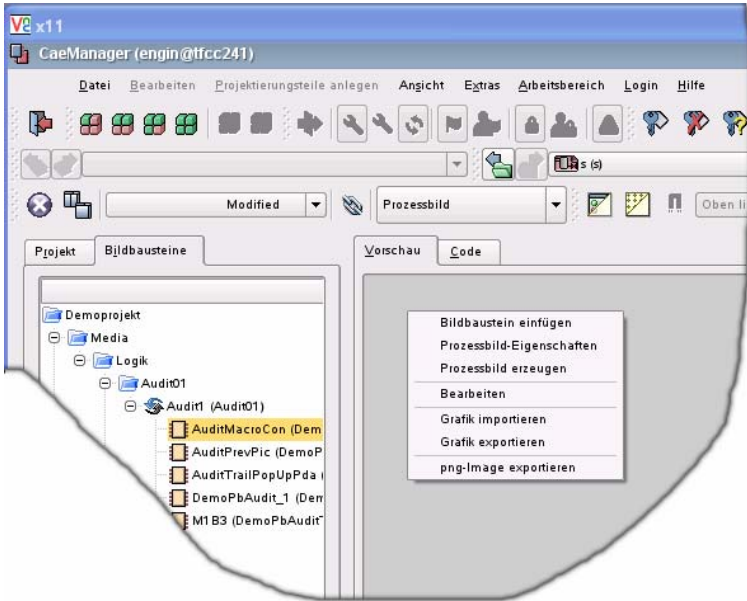


图97: 在CaeManager 中预览过程图

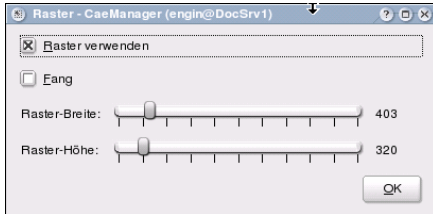
下面功能可通过快捷菜单选择:

步骤	说明
插入显示驱动功能模块	插入最后一个复制的图片模块
过程图表属性	显示名字, 说明, 实例, 创建日期, 过程图片的当前版本。
创建过程图	以 PNG 格式保存过程图。
编辑	启动 DisplayEditor 来创建/编辑过车图的静态部分。
导入过程图	从文件系统导入一个静态部分
导出过程图	导出静态图。默认设置中, 过程图存在以下目录 /home/<EngineeringSystem>/ENGINE/EXPORT/。
导出 png 图	预览图片可以以 PNG 格式保存, 如用于项目文档。

Code 标号用于浏览过程图的 ASCII 代码。只有在 **CaeManager** 菜单项 “**Extras/Options**” 中选择“**Edit old PVE format**”(tab **General**), 才能直接改变代码, 只能由专业的 **APROL** 技术人员来修改。

4.2.2 过程图里网格的使用

在相应的校验栏选择"使用网格"和"捕捉"项就可以使用菜单项"显示/网格"。网格大小可以通过鼠标的移动键来调整或通过键盘的[TAB]和方向键来调整。



建议使用"网格"和"捕捉"来定位图像在过程图的位置。

图 98: 创建过程图时的“网格”窗口



在创建静态图之前应该配置好网格。



使用"显示/全屏"或相应的图表来放大过程图的某些区域。

4.2.3 在过程图中插入操作选项

在过程图里插入操作选项可以参照如下：

- ✓ 直接在过程图里放置操作模板。模板是随时可见的；但是，模板里包含太多的选项的话会使过程图不明晰。
- ✓ 在过程图里使用控件来放置图片宏。点击控制区域可以暂时打开一个操作模板。控件可定位在操作的设备图例的下一层。设备本身可以动态显示其状态。这样可以使图不被过度调用。可以参考 创建控件。

当鼠标移动到控件所在区域时此区域会有一个边框显示("高亮控件")。



在 DisplayCente 的 CC-Modules 选择"-noHighlightControl" 可以是控件区域不高亮。

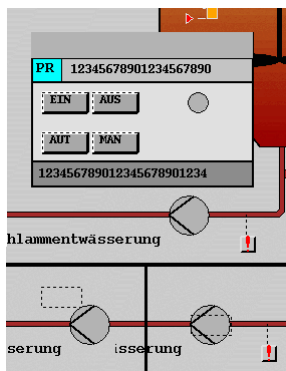


图 99: 过程图利的控件和操作模板

左图显示的就是上面提到的显示选项。

上半部区域显示包含泵的直接放置的操作模版。

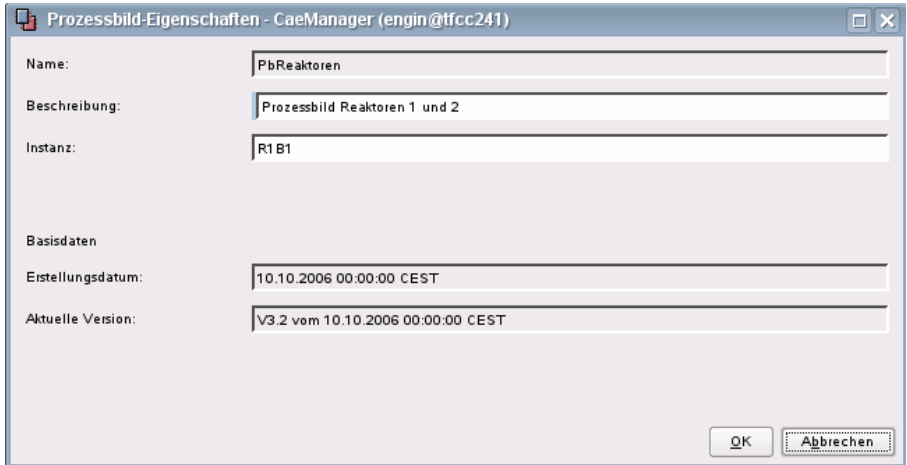
下半部是控件定位在设备的**旁边**或下面。控制区域由虚线围绕。这些虚线是不可见的。

4.2.4 过程图属性

在过程图里的空白区域通过快捷键可以打开相应得菜单项 "**过程图属性**".

在图片宏的快捷菜单"**显示驱动功能块属性**" 可以看到有关动态图片宏的一些信息。鼠标放置在图片宏上面时可以选择该图片宏。

过程图属性



Prozessbild-Eigenschaften - CaeManager (engin@tfcc241)

Name: PbReaktoren

Beschreibung: Prozessbild Reaktoren 1 und 2

Instanz: R1 B1

Basisdaten

Erstellungsdatum: 10.10.2006 00:00:00 CEST

Aktuelle Version: V3.2 vom 10.10.2006 00:00:00 CEST

OK Abbrechen

图 100: 过程图属性



给过程图再重新指定另一个静态图也是可以的。

显示驱动功能模块属性

Bildbaustein-Eigenschaften - CaeManager (engi)

Bibliothek

Baustein:

DcsPbMotDb

Bibliothek:

DCS

Projekt

Funktionsplan:

R02Ruehr

Bausteininstanz:

SYMBOL_1

Hypermakro:

M27201

Skalieren und Rotieren

☒ Proportional Skalieren

X-Skalierung:

1

Y-Skalierung:

1

Winkel:

0

OK

Abbrechen

图 101: 显示驱动功能模块属性s

输入	说明
模块	显示驱动功能块在库里的名字。
库	块所在库的名字。
功能图	显示功能块所在的功能图表
模块实例	模块的实例名，从块的名字自动形成并带有序号。模块的实例名也可修改以提高可读性。
超级宏	如果显示驱动功能块是直接放置在功能图表里， 就没有此项 。 如果是放置在超级宏里， 就显示超级宏的名字 。
比例和旋转	允许修改图片宏在不同轴上的比例。另外，图片宏也可以通过输入角度来逆时针旋转。值输入是做网格设置的。此值可以直接输入。

B2 项目组态

4-10

B2-4.3 在过程图里使用控件

操作模板可以直接放置在过程图里。这种方式下操作模板一直是可见的，大量的操作选项会导致过程图变得比较拥挤和杂乱。

另一种方法就是在过程图里使用控件。

控件是指一个敏感区域，当这个区域被激活(鼠标点击)会导致一个动作。

在过程图里使用控件时需要注意如下：

- ✔ 控件区域在 **DisplayCenter** 里面是不可见的。操作选项通过高亮边框来提示，在 **DisplayCenter** 通过鼠标来实现改变。点击鼠标来改变的内容是在创建控件时定义的。
- ✔ 建议把控件和窗口小部件(如按键，按钮)结合在一起然后产生一个动作。
窗口小部件可以用图像可视化元素事先定义好。
- ✔ 在 **DisplayCenter** 不选择 *"-noHighlightControl"* 选项则当鼠标放置在此区域上会有边框显示 (高亮控件)。
- ✔ 控件区域可以放置在过程图的任意地方。然而，建议把控件放置在设备符号的上面或下面。

在 B&R 的 CAE 库里提供如下控件类型：

控件	说明
直接图片调用	在不同的过程图里放置其它过程图。
发送脉冲	发送一个脉冲到一个数字变量。脉冲的长度和值是可以定义的。
文本窗口输入	打开一个输入窗口输入值并赋给一个过程变量。
滑动条输入	过程变量的值通过一个滑动条来改变。
LED 按钮	数字量的值通过一个 LED 按钮来改变并显示。
开关	数字量的值通过一个开关来改变并显示。
键	定义的键给指定的过程变量赋值。
显示宏	控件激活时可以打开另一个图片宏。
Python 按钮	此按钮激活时执行 Python 代码。

4.3.1 创建控件

如上所述 只有“直接图片调用”才能用在静态图上。

要创建一个控件，在 **DisplayEditor** 选择“工具/绘制/按钮”菜单。



也可以在“工具”条上选择左图所示的图标。

在需要放置**控件**的地方单击鼠标左键。控件是长方形的并可以象绘图工具一样使用。另外，控件可以任意方向拖曳。一旦控件形成所需形状，再次点击鼠标即可。长方形的区域就是随后过程图里的敏感区域。

在控件的终点被确定后，出现如下对话框：

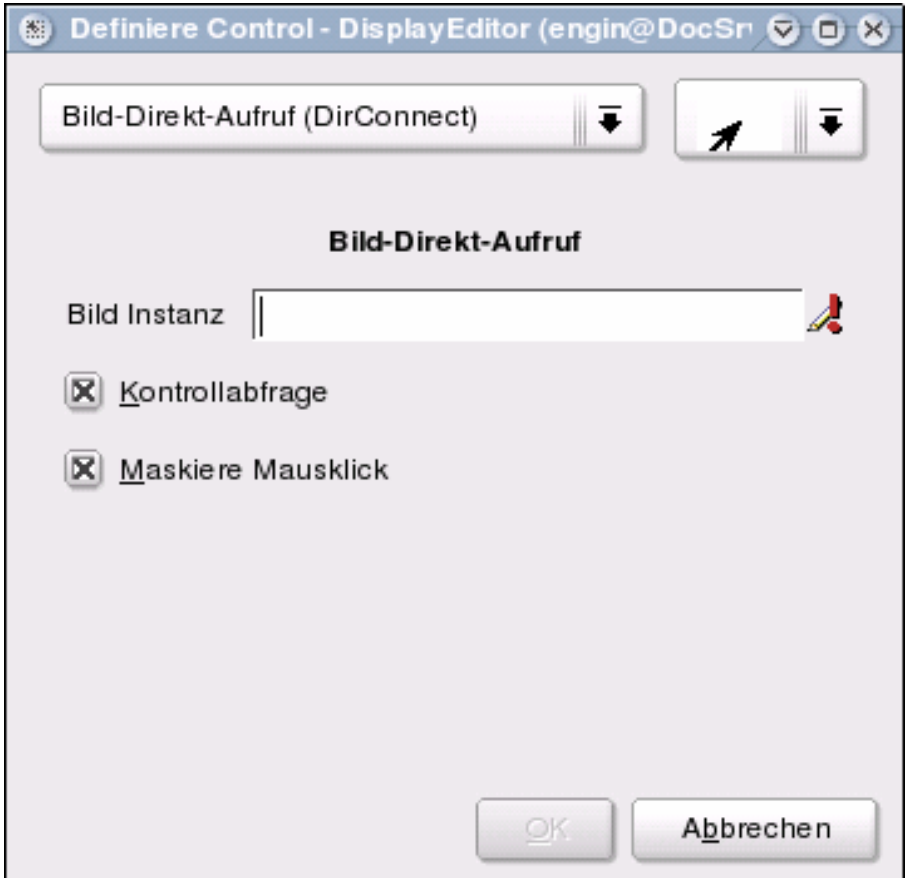


图 102: 定义“直接调用图片”控件

鼠标放置在敏感区域时显示的形状可以通过下拉菜单来选择。

在校验栏选项可以用来执行附加的**确认**(在动作执行前出现)。

如果控件区域有重叠，“**鼠标阴影**”项确保最上面一层控件被激活并被执行。



可以修改已经定义好的控件。

使用“**编辑复杂对象属性**”功能可以修改。

在选定的控件上双击，然后选择[Strg+A]，或菜单“工具/编辑/属性”可以知道此控件的功能。

B2-4.4 使用窗口小部件概述

可以把窗口小控件和控件结合在一起执行一个动作。

创建一个窗口小部件，在 DisplayEditor 菜单"工具/绘制/可视化元素"。

 步骤和定义一个控件类似。可参见[创建控件](#)。

库里面可以使用的窗口小部件的类型如下：

Widget	说明
3 维按钮	显示带文本的三维按钮。
3 维箭头	显示三维箭头。
变量的文本输入	产生一个输入区域，并通过键盘赋值给过程变量。
变量的比例输入	显示一个滑动条用以改变过程变量的值。
变量显示	显示过程变量的值。
文件/URL 显示	返回文件的内容。 如果输入的是 URL，网页浏览器会打开并显示相应页面。
多变量的在线趋势	用来显示变量的在线趋势。

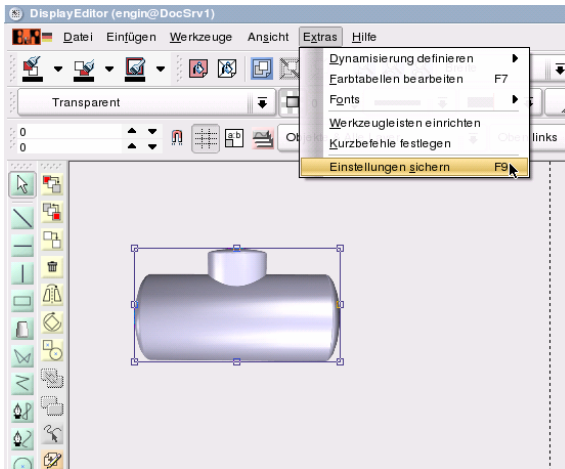
B2-4.5 DisplayEditor概述

4.5.1 保存设计者特别设置

在创建一个对象之前，可以先做一些设置。

例如，定义对象属性(如，填充模式)或设置工作环境(网格，层等)。

在 **DisplayEditor**，可以把如网格，填充模式，每个设计者的规则等保存。



选择"其它/保存设置"来保存设置。下次打开 **DisplayEditor**，就会显示上次保存的设置。

图 103: 保存设计者特定设置

设计者特定的工具条配置和快捷键

另外也可为每个设计者保存对象属性和工作环境，或为设计者配置的特定的工具条和快捷键。

"其它 /设置工具条"可以根据设计者的需要配置工具条，在必须的情况还可设置新的图标。



一旦所有的设置被确认后，下次再打开 DisplayEditor 就会显示这些设定。

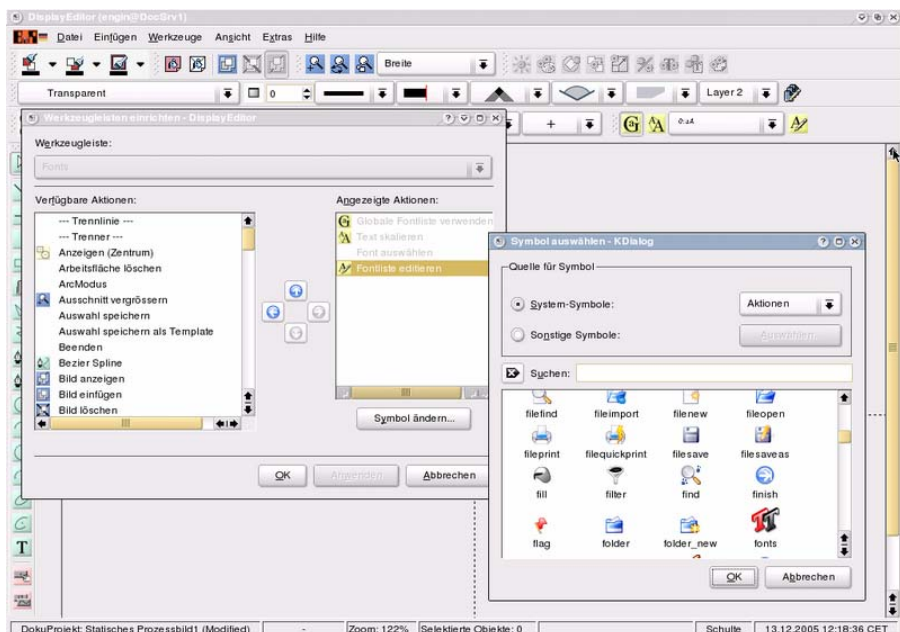


图 104: 设置工具条

使用菜单项 **其它/设置快捷键** 可以象配置工具条一样配置快捷键。在设置好后要使 **用户定义** 按钮使能。快捷键通过键盘输入，用 **[OK]** 来关闭对话框。



工具条和快捷键所作的修改都存储在 `<DesignerLogin>_DisplayEditorui.rc` 的 XML 格式的配置文件里。

这个文件的路径为
`/home/<EngineeringSystem>/ENGINE/cnf/DisplayEditor/user`
目录下可以由有经验的设计者用 XML 编辑器来直接修改。

4.5.2 创建图片对象特点

要创建图片对象，选择"工具/绘制"里面的绘制功能即可绘制。

可以通过菜单项"工具/绘制/图像"插入已经绘制好的图片。



*推荐将图片压缩到最小的分辨率，确保最小的图片尺寸。
图像以理想的分辨率显示。高分辨率图片会使图片下载时间延长。
由于性能因素，推荐使用"PNG"格式。*



不管是哪一种图片格式，能被组合的图片的大小不能超过 1MB！

选择这个菜单项后，在设定的框里插入对应位置和相应大小的图片。在紧接着的预览对话框里，在按下 [...], 在文件系统里已保存的文件可以被选择并最终插入。

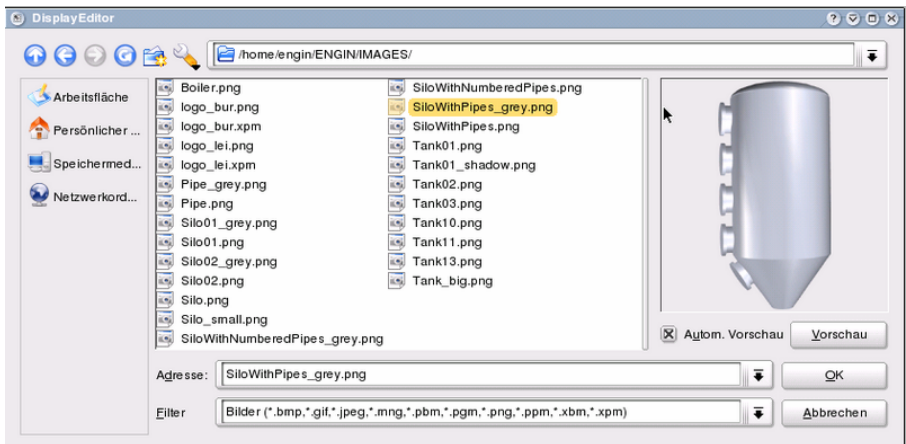


图 105: 插入图片



在 **APROL** 安装后有几幅范例的图片可以使用。
这些图片可以直接使用，图片存储在/opt/aprol/ENGIN/IMAGES。

在创建新的工程设计系统后，这些范例图片会自动保存在目录
/home/<Engineering-System>/ENGIN/IMAGES 里。

4.5.3 使用网格特点

创建静态背景图的时候使用的网格同样可以使用在插入动态图片宏的时候，有助于把动态图片宏插入到特定的位置。



创建静态图片宏的时候，选择 320x320 网格。也可以在把网格分的
更小(160, 80, 40, 等)。

除了设定网格，还可以使用"捕捉"功能来精确放置。

4.5.4 层技术

在 **DisplayEditor** 里创建静态图，图片宏，或图片模版时，在每一层要显示或隐藏时就要充分运用层。确保在创建可视化应用时能更好浏览整个效果图。



在 **DisplayEditor 保存对象时要注意：**

隐藏层里的对象同样能被保存，即使此层不可见!

要删除一个对象时必须使相应得层可见。然后此对象才能被删除或移动。

在 **DisplayEditor** 提供 16 个不同层，在任何层都可创建控件和小窗口部件。



层是从小到大排列。

即在"Layer 2"包含的图像显示在"Layer 1"包含的图像的上面!

窗口小部件无论放在哪一层都显示在最上面一层。

对每一层可以添加相应的语言描述等以便区分。

建议预先规划好对象所在的层数。

过程如下:

- ✓ 在 DisplayEditor 的选择相应工具条可以调出"编辑层"的对话框。
- ✓ 在"层"下拉列表里选择层。
- ✓ 用鼠标选择相应层的"描述"。
此处输入描述。
- ✓ 通过[Enter]键来确认, 点击[OK]按钮关闭次对话框。

下次再打开"编辑层"对话框时就会显示实际的描述。

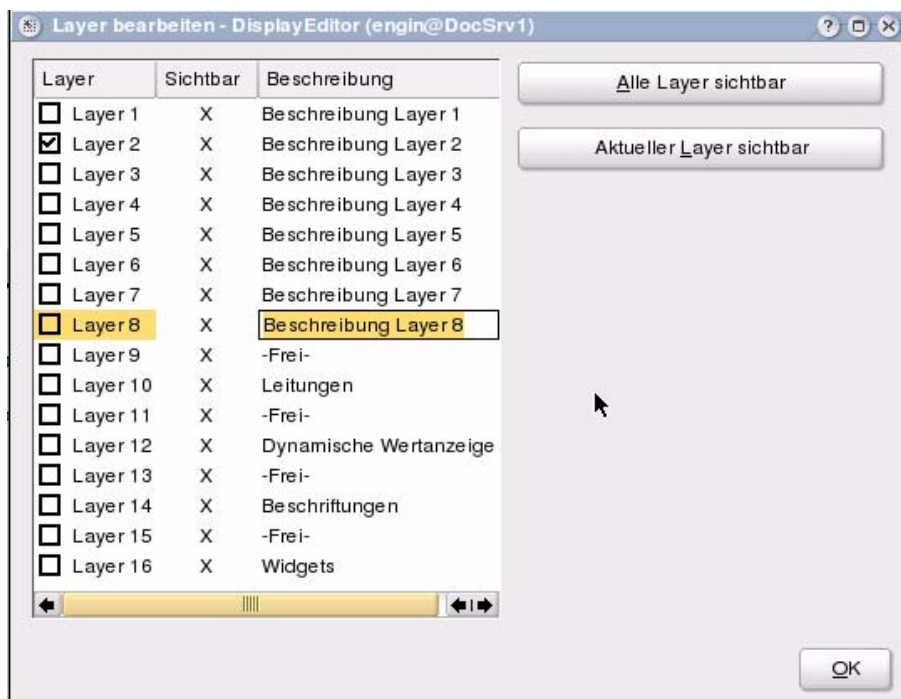


图 106: 给定描述的层

标准的层的描述(层 1 描述, ... 层 16 描述)保存在文件 `layer_<LanguageCode>.list`, 在目录 `/opt/aprol/cnf/DisplayEditor/`。



如果层的描述第一次修改, 会产生一个拷贝(如上), 此修改会保存在一个新的文件 `layer_<LanguageCode>` 里面。

以后的描述文本的修改会保存在本地目录 `~/ENGINE/cnf/Display Editor` 里面。

B2-4.6 DisplayEditor功能

DisplayEditor 支持**工具条**和**工具提示**。

下面不再就图形编辑的一些通用功能作更详细的描述。

下面讨论 **DisplayEditor** 里面其它特点。



***DisplayEditor** 打开后一些事先定义的动作就被选择了。*

4.6.1 “文件” 菜单

文件 菜单包含如下功能：

菜单选项	说明
页面尺寸	在 DisplayCenter 显示多页过程图定义子页。
删除工作空间	注释。
保存选择	保存事先用选择工具和选择对话框选择的图形对象。 在静态图里是不可能的。
保存选择为模板	在静态图或图片宏里的单个图形可以保存为图像模板。默认情况下，图像模版保存在 home/<Engineering system>/ENGINE/GRAPHIC_TEMPLATES 目录下，在创建图片宏时可以有动态模版的方式插入。 可以通过选择对话框选择不同的存放目。
保存	保存背景图或图片宏的修改。
保存并关闭	保存并关闭 DisplayEditor 。
关闭	在显示一个警告后， DisplayEditor 关闭不保存。

4.6.2 “插入” 菜单

插入 菜单包含如下功能:

菜单选项	说明
静态图片宏	通过选择窗口在库里导览所有的图片宏并选择作为部分或整个图片宏。 当调用静态部分时，所有动态的操作都被过滤。
动态图片宏	通过选择窗口在库里导览所有的图片宏并选择作为部分或整个图片宏。 这个不可能在静态图中，因为动态化只能在图片宏里!

下面的图片显示了预览激活的选择对话框:

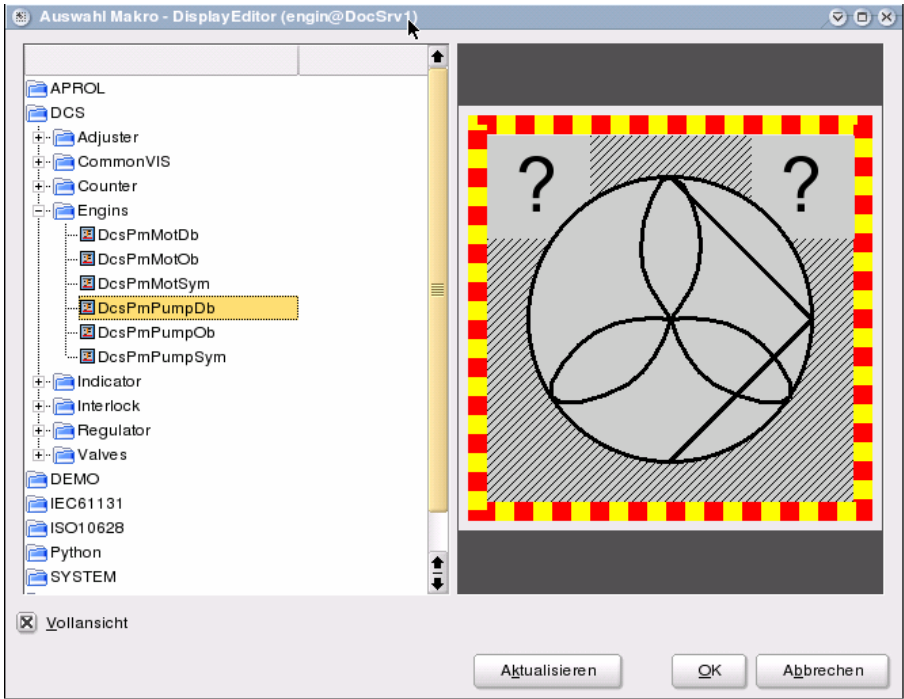


图 107: 选择图片宏的插入图片部分

4.6.3 “工具” 菜单

工具 菜单包含如下功能:

菜单选项	说明
绘制	工具/绘制 菜单条用来创建对象。 可以创建线(线, 水平线, 垂直线), 多边形, 圆, 椭圆, 等等。 按钮 菜单条用来创建控件, 可视化元件 用来创建窗口小部件。
编辑	使用菜单条放置到前面, 放置到底层, 可以移动对象到当前层的前面或后面。 拷贝, 删除, 镜像, 旋转, 显示(中心)的功能如字面意思。 组: 选择几个对象后, 可以把它们成一组。更详细的信息参见图形对象的成组。 拆分: 选择一个组把它解散。更详细的信息参见图形对象的成组。 修改线: 以多边形, 样条曲线, 多段线方式移动线; 添加新的线。 属性: 选中图片的属性可以在“属性”对话框中定义。 “属性”对话框可以通过[编辑目标属性]图标, 双击选中的图片, 或双击目标列表中的组调用 因此, 请注意下面的图片。 更多信息参见文本对象的属性。 动态显示: 选择相应的图形对象后, 显示此图形对象的动态显示列表。 选择图形对象后可直接移动或放大缩小。选择图形对象后可直接移动或放大缩小。 用户指导如字面意思。 用户指导如字面意思。
动态类型	动态化有下述功能: 闪烁, 颜色, 旋转, 移动, 缩放, 比例, 文本, 文本列表, 工具提示。 只有创建图片宏时才可以用这些动态化!
背景	插入图片: 组合一幅背景图到静态图里(如照片或扫描的 P&ID 图)。 删除图片: 删除组合的背景图。 显示图片: 在静态图里显示/不显示背景图(拖曳功能)。 选择颜色: 背景颜色可以通过调色板来设置或运用吸管吸取屏幕上其它对象的颜色。颜色以 RGB 码保存。 删除颜色: 删除背景颜色。

菜单选项	说明
颜色	前景颜色: 为图形物体选择前景颜色。 背景颜色: 为图形物体选择背景颜色。 边框颜色: 为图形物体选择边框颜色。
类型	选择填充模式, 线形, 等。
选项	捕捉到网格: 鼠标移动到网格时被捕捉。对象不在网格里不能编辑。 如此项功能不起作用, 则在编辑时对象可移动到任意位置。 页面限制: 多页过程图的显示限制。 比例: 等比例/不等比例缩放。 控件显示在顶端: 在底层的控件也显示在顶端。 选择模式: 单个对象在所有层/当前层被选择。 图形对象(从当前层/所有层)也可以通过对象列表来选择。 定位模式: 设定此位置哪一个对象被捕捉。在操作和精确移动物体是非常有用。 规则: 设定光标形状。

为每个对象及对象组定义对象属性。

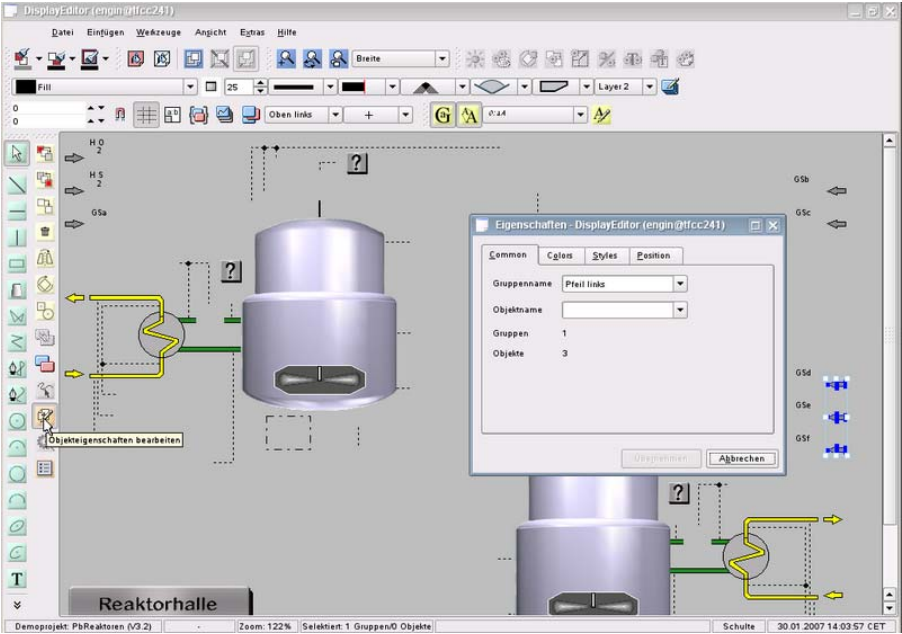


图 108: 编辑对象属性

4.6.4 “查看” 菜单

查看 菜单包含如下功能：

菜单选项	说明
部分放大	放大选择的部分。
拖曳放大	在以前和当前的放大因子变化。
预览	屏幕预览。
放大	设置放大因子。



选择部分对象如下：

按着鼠标左键。拖动鼠标到理想的位置。释放鼠标左键即可。

4.6.5 “其他” 菜单

其他 菜单包含如下功能:

菜单选项	说明
动态化	颜色定义: 动态化列表的名称可以修改，也可以新定义动态化的名称。名称可以重命名，在名称列表里添加或删除。 建议动态化的命名采用自解析的命名方式(如: NonFlammableGas)表示一个状态，不使用颜色。建议放在不同的库里面-特定前缀(如: DCS_NonFlammableGas)。 颜色通过调色板来选择(双击前景或背景)。也可以用吸管来吸取屏幕上其它对象的颜色。颜色定义可以保存。颜色的名称以 RGB 码保存。 通过快捷菜单可以打开填充模式和线形。 选择填充方式，符 DIN ISO128-50 标准。 文本定义: 文本定义可作为系统画面某种事件的信息文本出现在文本列表动态化的文本框里。 这个不可能在静态图中，因为动态化只能在图片宏里!这个不可能在静态图中，因为动态化只能在图片宏里!
编辑颜色条	经常使用的颜色可以组合在一起放在一个单独得调色板里，方便快速使用。 Color 颜色条也可以通过 KcolorEdit 来编辑。可以定义任意颜色的调色板并保存。

菜单选项	说明
字体	使用全局字体列表： 创建静态图和图片宏的时候，可以选择 global 全局或局部字体列表。 不同之处： 全局字体列表的优点在于每个字体都在整个项目里使用。通过修改全局字体列表来修改个别的字体，这样使用到该字体的所有图都修改了而不必去修改每一幅图。 局部字体确保某种字体的特定使用。任何图片和图片宏使用到局部字体时只显示当时创建图片宏或静态图时字体，不管字体是否有改变。 字体大小： 在静态图使用的字体的大小。 默认此选项是激活的。 编辑字体列表： 要编辑一个新的字体，在"字体属性"对话框(双击)选择一个字体。使用 "选择字体" 菜单为新的字体设置参数。在使用升级版本时，有些项目不用选择字体，或在错误命名下不能使用。 如果出现这些问题会有装置帮助你选择一个适合的字体。 当前字体会显示在“字体属性”对话框中。 只有在使用局部字体时： [采用全局字体] 按钮运用全局字体在局部字体列表里。
设置工具条	设计者特定的工具条和快捷键设置 参见 <u>保存设计者特定设置</u> 。
定义快捷键	设计者特定的工具条和快捷键设置 参见 <u>保存设计者特定设置</u> 。
保存设置	参见 <u>保存设计者特定设置</u> 。

4.6.6 “帮助”菜单

帮助 菜单包含如下功能：

菜单选项	说明
信息	显示应用的版本和使用的库。 "
详细信息	显示详细的信息。

4.6.6.1 文本对象的扩展属性

要创建一个文本对象，选择 **工具/绘制/文本** 菜单项。用鼠标选择文本区域。上半部输入框和配置对话框可以用来输入所需的文本。



按下[CTRL]键同时使用鼠标点击可以同时选择多个对象，或直接用鼠标拖动来选择。选择对象的数量会显示在状态条上。

对话框的中间区域用来定义文本字符串的文本区域形状。另外，可以定义文本与垂直边框的距离。

下列属性除了 **字体选择** 外都可用：

断字属性定义文本长度超过文本区域的水平空间时如何显示。如果断字的属性是激活的，在显示时文本会自动换行。否则则显示文本区域内的文字。

如果 **不能修剪**属性被设置，文本字符显示可以超出文本区域边界。文本显示可超出水平区域和垂直区域(多行)。



如果断字属性同时激活，断字优先。

动态边界属性使文本区域根据输入的文本动态调整长度。

设置此属性就自动清除断字属性，反之亦然。

不能翻转属性使文本不能被翻转。



文本区域的大小可以通过“**缩放**”工具来改变。文本框里的文本根据设置的属性排列。

文本区域的文本可以通过“**旋转**”工具来旋转。“**旋转**”动态特性也可以使用。

“**镜像**”工具只是作用于“**边框区域**”。

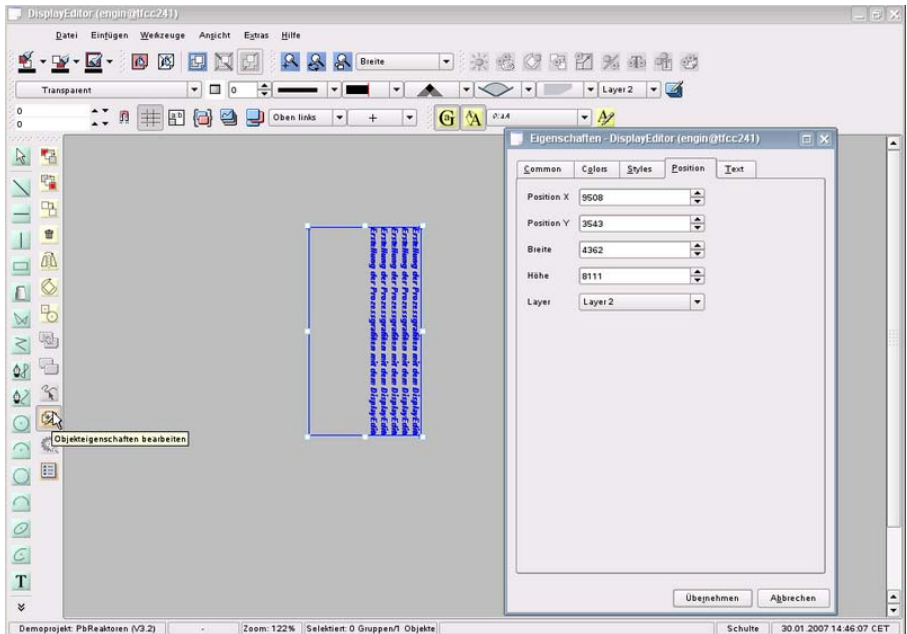


图 109: 创建文本对象时的扩展属性

B2-4.7 创建对象注意事项

创建对象的指导工具是很明显的。

下列事项还是有帮助的:

- ✓ 图形对象是以事先选择的对象属性来显示的(颜色, 线形, 等)。
- ✓ 当绘制闭合的多边形和样条函数, 线的闭合是通过最终双击来完成, 因此最终的绘制点直接连接到起点即可。
- ✓ 创建部分圆或椭圆时, 首先定义圆的形状(或椭圆)。然后鼠标点击圆的开始, 在点击鼠标确定结束点。方向一般是反时针方向。预览会显示出来。
- ✓ 鼠标右键单击可以取消选择的功能。

B2-4.8 鼠标中键的使用



下列绘制功能是通过两个点来绘制。如果起点/中间点被错误放置，可以通过**鼠标中键来重新设定起点/中间点**。

当按下中键时可以通过鼠标重新设定起点/中间点的位置。选择另一个点后绘制过程就结束了。在绘制结束后，中键就不再起作用了。

下列绘制功能是通过多点来绘制。最后的设定点可以通过中键再重新设定。

当按下中键时设定点就被重新设定了。在绘制结束后，中键就不再起作用了。

这些绘制功能也是通过多点来确定的，并通过双击来完成。当**按下鼠标中键设定点被删除，可重新设置定**。相比较上面提到的绘制功能，最终的绘制点被删除，按下鼠标中键时光标所在并不是新的绘制点。

在使用**"旋转"**时**鼠标中键选择对象的旋转点**。旋转点是相对于原始对象来说的。通过单击作键来结速旋转。

在按下鼠标中键，同时按下[Strg.]键，工作区可以在选择窗口下移动。这样提供了另一种可能来移动而不使用滑动。

B2-4.9 编辑已有对象

在编辑前切换到选择模式。然后选择工具或对象列表。

菜单选项	说明
拖曳到上一层/ 拖曳到下一层	通过菜单条 "拖曳到上一层" ，或 "拖曳到下一层" ，可以将对象移动到实际层的上一层或下一层。
复制	如字面意思 注意： 在拷贝一个图形对象后，拷贝功能还是激活。
删除	如字面意思。

菜单选项	说明
镜像	镜像一个对象的过程如下： 首先选择对象。其次选择对象。对象会被封闭边框围绕，边框的角和线的中部都有显示镜像方向的三角。移动鼠标选择理想的三角方向。三角方向垂直与镜像轴，同时显示在那个方向镜像。 点击鼠标确认此步骤。此对象就显示镜像后的状态。
旋转	在"旋转" 菜单条选择后，对象按照默认的中心旋转。旋转点可选在对象的内部或外部。
显示（中心）	点击显示工作空间任何地方显示对象的中心。除中心点外，围绕对象的边框的中间点和角也会显示。
组	选择几个对象后，可以把它们成一组。更详细的信息参见图形对象的成组。
拆分	选择一个组把它解散。 更详细的信息参见图形对象的成组。
修改线	根据多边形，样条曲线，或多段线来创建或修改线。鼠标双击可以抓取并移动线，按下左键释放。双击可以新增线段。
扩展属性	高级属性使用在文本对象，组合图形，控件，窗口小部件。 选择一个对象后，可通过双击,菜单条，工具条来弹出对话框编辑属性。 更多信息参见 文本对象属性。
动态显示	选择相应的图形对象后，显示此图形对象的动态显示列表。

选择图形对象后可直接移动或放大缩小。用户指导如字面意思。

4.9.1 图像对象分组

在创建静态图，图片宏时，可以参见如下步骤把一些图形对象成组。

首先使用选择工具选择要组合的对象。



使用选择工具选择对象时可以通过按下[Ctrl.]键，或用鼠标拖曳选择区域来选择多个对象。



组合可以使用菜单条"工具/编辑/组合", 或通过选择相应得图标。

在输入组的名字后, 就可以对这个对象组做进一步的操作(包括动态化)。



对一个组必须输入组的名字。

现存的组可以通过菜单项"工具/编辑/拆分"来打散。

有时需要在不拆分组的情况下来修改组里单个对象的属性。



使用这个图标, 所有现存的组都可以打开, 并且关闭。

4.9.2 使用对象列表

对象列表通过菜单“工具 / 编辑 / 显示对象列表”来调用, 或者通过工具栏内相应的图标。打开对象列表后, 可在放置在 **DisplayEditor** 的任意位置。

根据分组和布局, 显示每个对象。

由这个元素选择整个组。 要求就是组的状态为关闭。

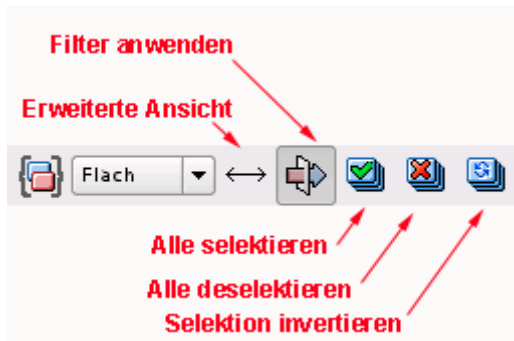


图110: 在对象列表中工作

在显示的对象列表中有两种方法选择对象。

可以在列表中选择或者在工具栏中选择(标记为"勾")。 另外一种方式是在图像区域中拖拉(鼠标左键)框选择。

4.9.3 动态化概述

创建动态图片宏时，每一个对象都可以通过"工具 / 动态化" 菜单动态化。

有关动态化更详细的描述在 ***APROL B3 CAE 库工程设计*** 一章。

动态化有下述功能：

动态类型	说明
闪烁	对象根据过程变量的值以不同的方式闪烁(背景，交替，不同频率，等)。
颜色	对象根据过程变量的值改变颜色。背景和前景色，填充模式，线形都可以改变。
旋转	对象根据过程变量的值围绕一个设定点旋转。
移动	在过程图里移动物体。X 轴和 Y 轴上的移动是相互独立的。
缩放	对象根据过程变量的值改变大小。X 轴和 Y 轴上的缩放是相互独立的。
比例	对象根据过程变量的值改变棒状比例显示。
文本	过程变量的值以文本的方式显示。
文本列表	文本定义成一个列表并根据过程变量值的不同显示不同的文本。
工具提示	可以设定特定的工具提示(特别是与控件相连时)。 要注意在 CC module 中的设置。在 DisplayCenter , " 工具提示类型 " 选项应该设置为" <i>custom</i> "。

下面的图片是闪烁动态化的对话框的一个实例。

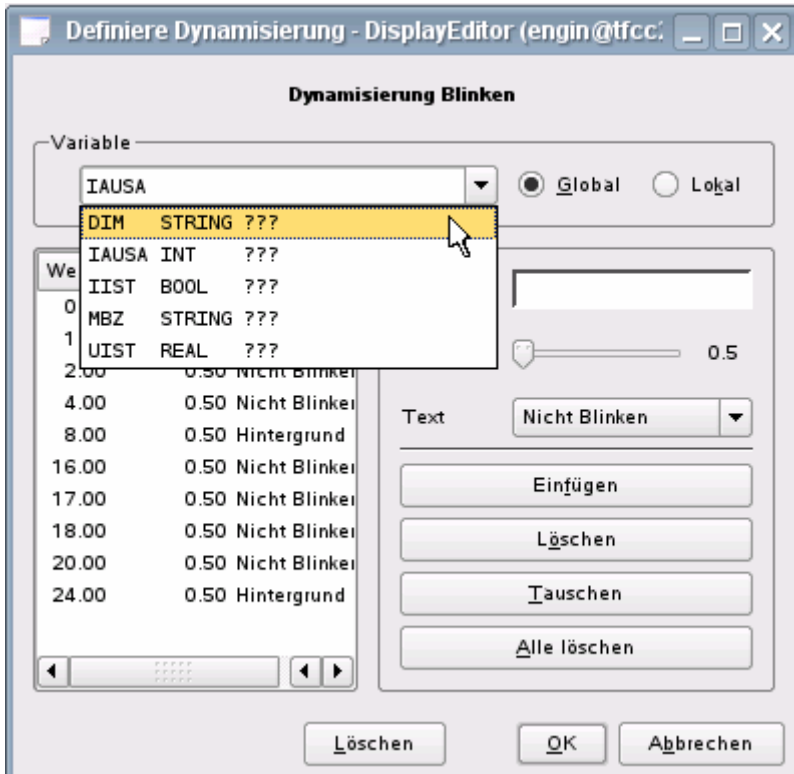


图 112: 闪烁动态化实例

要动态化一个对象，需要连接到一个过程变量。连接是在功能图表里完成，在显示驱动功能块里定义管脚名。

确切说变量可以直接在显示驱动功能块找到。“全局”项意味着是一个全局过程变量。选择局部意味是一个 Python 变量。

 在 *dynamization* 对话框中可以选择所有已存在的管教分配。

在输入变量名后，通过变量值来定义不同的动作。现存的输入项可以交换或删除。

对于闪烁，可以是“不闪烁，交替，开<->关”。

对其它动态化，必须在相应的对话框里设置相应的选项。

B2-5 参数管理

B2-5.1 参数管理概述

在工程设计阶段创建**系统部件**和**参数模版**时,可设置一个优化的参数管理系统即通过**参数中心**(在 Runtime 系统)来操作。

在过程控制系统,必须经常输入或赋一个新值(参数)到运行服务器以便符合当前的产品环境。可以通过提供的操作模版来输入新值。然而,此过程需要操作员或系统管理员高度集中,在某极端情况下可能导致错误。输入数据的时间也是值得考虑的。

为简化和确保此操作,整个控制系统的重要的参数(或者时间变量值)可以组合到一个**参数中心模版**里,参数中心里可以设定新的输入值。你可以说:



参数中心可替代过程控制系统里的部分目标数据。

在**参数模版**里,可以把不同的参数组合到在一起并放置到一个参数集里面。例子:



一系列控制器的不同组的控制参数集。其它控制参数取决于产品需求或产品。



不同的参数集适合于生产不同的产品。

换句话说,当从产品 A 换到产品 B,不必对**每一个部件**的参数都进行修改。只要从参数模版里选择 *product B* 的参数集并通过按钮把它装载到系统里。

需要修改参数的**系统部件**首先要定义好。然后从这些系统部件生成**单元**,**参数模版集**,或一个**参数模版**。对系统的此部分,必要的参数集要放置到运行系统并下载。

在功能图表组态设计中，通过命名来识别的参数可以插入到输入或输出框并通过它们所在的功能图表的逻辑来扫描和连接。下载后值就输入到运行系统。

B2-5.2 参数管理的许可条件

要在参数模版里存储参数集，参数管理系统要使用 **MySQL server** 里的 **MySQL 数据库**。**MySQL 数据库**和服务器需要选择因为在运行时可能需要获取外部数据。可以创建，管理参数集，写入参数到**APROL** PCS。使用参数管理的时候，要考虑到 **MySQL** 应用需要特别的许可条件。

在运行系统进行工程设计时，注意服务器的整个 **MySQL 特定源文件**，*mysql 客户端库和客户端*，和 **GNU 库**一样是要有 **GNU 许可**。在参数管理作为使用到 **MySQL** 因此必须需要一个有效的 **MySQL 许可**。

确保 **APROL 许可**同样适用于 **MySQL 许可**。假如不能确定 **APROL** 系统有有效的 **MySQL 许可**，那么查找是否有能扩展**APROL** 许可。在这种情况下，联系**APROL**销售代表。

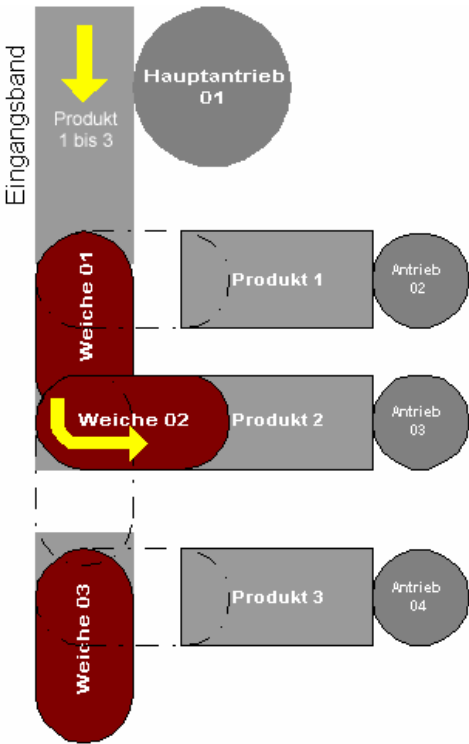
注意每一台运行 **MySQL servers** 的机器都需要 **MySQL 许可**。
运行在一台机器上的 **MySQL servers** 个数是限制的，同时连接到一个服务器上的客户端的个数也是不限的！

因此，对使用到参数管理的控制计算机都需要一个 **MySQL 许可**。这意味着要使用冗余系统时需要两个许可。

有关 **MySQL 许可**的更多信息可参见

<http://dev.mysql.com/doc/mysql/de/Lizenzpolitik.html> 或咨询 **APROL** 销售代表。

B2-5.3 系统部件和单元的示例



假设有三个产品通过传送带来传输。根据产品来确定切换和驱动方式。让产品 1 就到达运送产品 1 的皮带，其它类似。左边的图显示切换中要使用到的所有系统组件。

示例中显示了要操作的单元。产品 2 正在运输。产品_2 参数集从产品 1-3 模版装载到控制器里。

图 113: 系统组件和单元

具有相应功能的下列系统组件可以在过程控制系统中使用：

系统组件 1 (SystemComponent_01):

单位	说明和功能
MainDrive_01	所有 产品都必须通过此通道来传输。对每一种它都应该是激活的。假如没有产品被传输，则为 off。 功能切换： On / Off

系统组件 2 (SystemComponent_02):

单位	说明和功能
Switch_01	必须切换让 P1 能够转到产品 1 的传送带上。 功能切换： 转向 P1/其它产品继续向前。
Drive_02	驱动 P1 传送带。 功能切换： P1 时 On/否则 off

系统组件 3 (SystemComponent_03):

单位	说明和功能
Switch_02	必须切换让 P2 能够转到产品 2 的传送带上。 功能切换： 转向 P2 / P3 继续向前。
Drive_03	驱动 P2 传动带。 功能切换： P2 时 On/否则 off。

系统组件 4 (SystemComponent_04):

单位	说明和功能
Switch_03	必须切换让 P3 能够转到产品 3 的传送带上。 功能切换： 转向 P3。
Drive_04	驱动 P3 传送带。 功能切换： P3 时 On/否则 off

一个系统组件不一定只包含一个单元。 可以有多个单元组成。

多个单元可以组合在一起形成一个系统组件。 上面的示例单元分配给产品 1 到 3。然而，并不是各个系统组件都要包含所有的元素。

单元 1(Unit_01):

在这个示例中，Unit_01 由 SystemComponent_01 到 SystemComponent_04 的所有单元组成。



在工程设计时，详细的注视(描述)对工作非常有帮助。 请使用提供的选项!

下面所指是用**参数模版**取代单元。



参数模版把所有工作单元的参数组合在一起。这些参数就形成参数集。

B2-5.4 配置系统组件

系统组件在 CaeManager(插入/系统组件)可以直接插入。在打开的对话框，需要输入带描述的唯一的名字和实例名。



选择一个短且意思丰富的名字来做实例名(参见[创建一个新的部件](#))。实例名是在插入系统组件参数到**参数模板**时使用

下图是用**实例名 AT02** 填写系统组件 2 的例子。

Konfiguration des Anlagenteils "Anlagenteil_02"

Instanz:

Parameter | Anlagen-Konstanten

	Name	Beschreibung	Typ	Minimum	Maximum	Default
1	Antrieb_02	Ein für Produkt1	BOOL	0	1	0
2	Weiche_01	Ein für Produkt1	BOOL	0	1	0
3			BOOL	0	1	0

图 114: 配置系统组件

使用[New]按钮打开新的输入条；使用[Deleted]按钮删除输入条。

解释:

输入	描述
名称	参数的名字，值可以通过参数集来修改。 一旦系统组件被编译后此参数就可以在功能图表的输入和输出端使用。
说明	参数的文本描述。输入条必须填写！
类型	选择参数类型。
最小值	赋给参数的最小值。
最大值	赋给参数的最大值。
默认	参数的默认值。创建参数集时的预设值。

对每一个系统组件也可以定义**系统组件常量**。这些值在**系统部件常量**栏里输入。参数的值也是以同样的方式来输入。最小值，最大值，默认值是不提供的。只有常量值和类型，描述，和单元能输入。

 在单元被激活和编译后参数和系统部件常量可以在功能图表的输入框获取和连接。

 **系统组件常量的值只能在系统组件里修改! 即使在过程控制系统运行时这些都是常量!**

B2-5.5 创建参数模板

在系统组件里可以组合一些参数这样通过点击按钮就可以同时修改相应的值。更详细的信息参见[参数管理的基本信息](#)。

系统组件在 **CaeManager** (插入/系统组件)插入后，在打开的对话框，需要输入名字，描述和实例名。

 **实例名**用在运行系统查找参数模版(在打开参数中心)。

放置在一起的参数具有相同功能联系。参数集在另一步创建。使用不同的名字，“变量集”被赋值。这些参数集的值可被设置通过“按下按钮”到过程控制系统和控制器。

对上面的示例，意味着：

- ✔ 对过程控制系统需要参数模板。
- ✔ 需要至少 4 个参数集才能覆盖例子里的所有功能：

- 一套设置，切换所有到 off，
- 其中一套(总共 3 套)，切换产品(1 to 3)到正确的传送带。

接下来的三张图片显示了例子里的参数选择，参数集的创建，和其中的一套参数集。

Definition des Parameter-Templates <Produkt1_3>

Selektierte Parameter

Parametersätze

Instanz:

PT1

	Anlagenteil	Name	Beschreibung	Typ	Minimum	Maximum	Default	Fix	
1	AN1	Hauptantrieb_01	Motor Eingangsband	BOOL	0	1	0	Nein	
2	AN2	Antrieb_02	Treibt Band 1	BOOL	0	1	0	Nein	
3	AN2	Weiche_01	Lenkt auf Band für P1	BOOL	0	1	0	Nein	
4	AN3	Antrieb_03	Treibt Band 2	BOOL	0	1	0	Nein	
5	AN3	Weiche_01	Lenkt auf Band für P2	BOOL	0	1	0	Nein	
6	AN4	Antrieb_04	Treibt Band 3	BOOL	0	1	0	Nein	
7	AN4	Weiche_03	Lenkt auf Band für P3	BOOL	0	1	0	Nein	

图 115: Product1_3 例子的参数模板

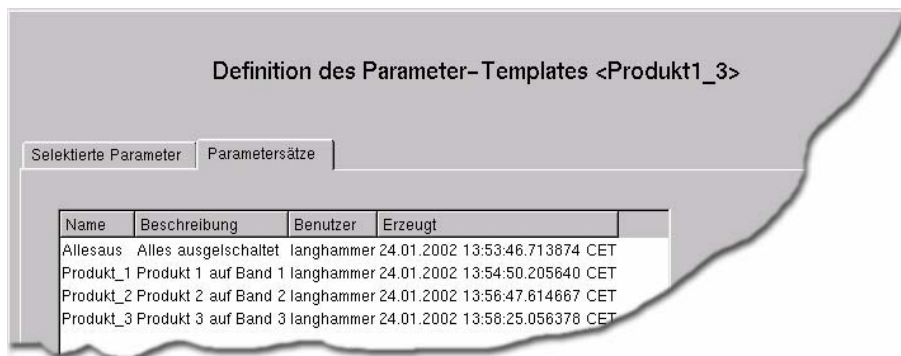


Illustration 116: Product1_3 例子的参数集

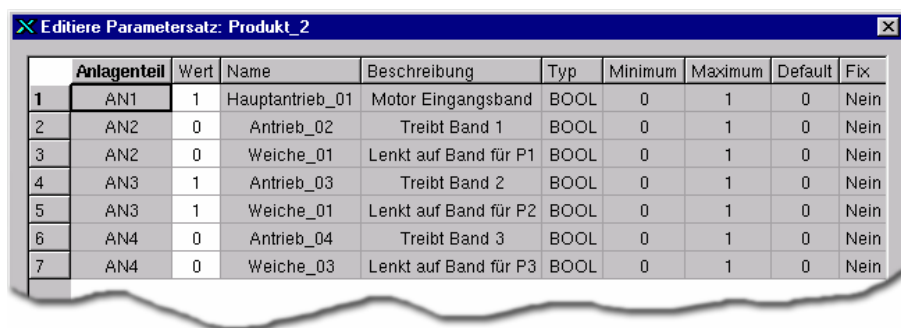


Illustration 117: 具有设定值的参数集

5.5.1 参数应用

在创建参数模版后，在选择参数栏的配置区显示空白的列表。可以在系统组件栏的导览区插入可选择的参数到列表里。

i 系统组件的实例名而不是名字可以被选择。一个参数在参数模版只能添加一次。

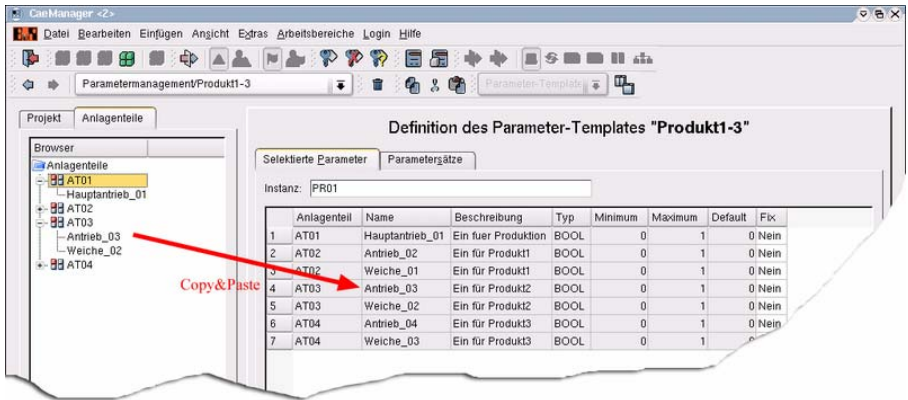


图 118: 在模板中插入参数

在选择参数栏有**固定** (是否) 选项来确定这些参数在运行时是否可以修改。参数条的输入可被移出或删除。



如果参数模版里使用的参数在**系统组件里被删除**并且系统组件重新编译后，参数模版就会跟着改变。系统组件里被删除的参数在参数模版里以**红色高亮显示**！

5.5.2 创建参数集

创建好的参数集在下载到运行系统后是可以使用的。每个参数集都有名字最好要有描述。

可以创建，编辑，或删除单个参数集。通过相应得按钮来完成。

[新建]按钮可以创建一个新的参数集。



参数集的名字和描述应参考它的内容。

参数集可以通过双击来编辑或选择参数后在点击**[打开]**。**[删除]**用来删除选择的参数集。

5.5.2.1 编辑参数集

一旦参数集被打开，参数就可以被设定了。可以在参数集里修改参数的值。

你只能改变这个集里的参数值。值应该在最大值和最小值之间。

参数集里参数的值就是默认值。

输入条 **Fixed = Yes** 意味着此参数可以在参数集里修改，但是在运行的过程控制系统里是不能修改的。

B2-5.6 在功能图表中使用参数

下载时需要在过程控制系统里(在功能图表)插入参数和常量参数。参数和常量在功能图表里可以从输入框连接到“物理视图”。

如果想上载参数集里参数的值，就需要把参数连接到输出框。

参数管理里的参数和常量在功能图表的输入框则是可以下载的。可以下载的参数名字表示如<Name>_IN。

可以从输出框选择要上载的参数。可以上载的参数表示如<Name>_OUT。



<Name>_OUT 应该放置在输出框同时使用上载机制对哪一个值是参数管理的值有合适的显示。

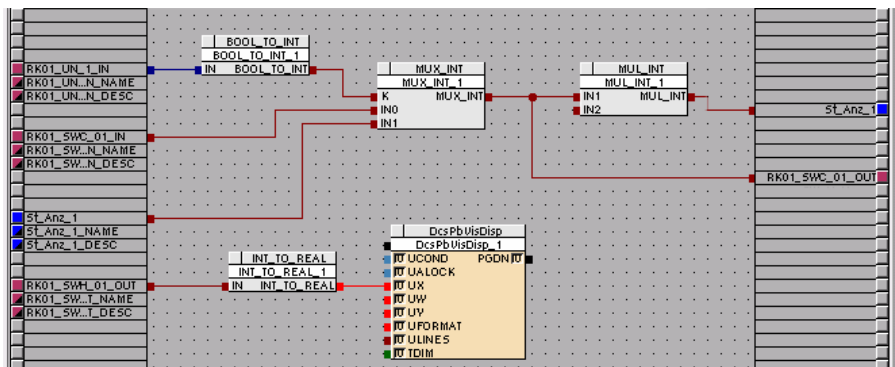


图 119: 功能图表参数连接示例

在示例中，**SWC_01_IN** 的值从参数中心通过多路转换器输出。如果多路转换器的 **K** 值被设定，通过调整 **St_No_1** 的值就可以通过多路转换器输出。假如系统同步良好，在参数管理系统通过上载就可以回收到 **SWC_1_OUT** 的值，此时是一个读的状态。通过 **SWC_01_OUT** 值可以知道是哪个输入的。

B2-6 附录

B2-6.1 版本历史

手册版本	日期	更改记录	作者
1.60	01.02.2007	修改章节“过程图”	Sagrillo Heitmann Schulte
	31.01.2007	对象列表中的过滤项，字体属性，图形对象的对话框“属性”，动态对话框选择。	
	30.01.2007	远程编译章节“重新编译 CAE 项目”	Sagrillo Heitmann
1.55	09.01.2007	报警组变量注释	Sagrillo Heitmann, Teligmann Schulte
	19.12.2006	实施章节“过程图”	
1.54	20.11.2006	- 重新编译项目 - 注释 <i>ApDrvCross</i> 和 <i>ApDrvIna</i> 状态变量 - 注释 IEC 类型	Schulte <i>Feldhaus Teligmann</i>
1.5.3	13.09.06	- 提供图片示例	Schulte
1.52	10.08.06	章节“过程图” - 命名层注释 - 添加图片示例	Schulte
1.51	28.07.2006	章节“过程图” - 移动 <i>图片目标</i>	Schulte
1.50	21.06.06	修改章节“访问硬件 I/O” - 选择对话框改变	Schulte
1.49	30.05.06	修改章节“过程图”	Schulte

手册版本	日期	更改记录	作者
1.48	11.04.2006	章节“过程图” - 分组图片目标 修改章节“功能模块 – 管脚分配 预览”	Schulte
1.47	27.01.2006	“过程图” 章节全部修订	Schulte
1.46	22.11.2005	历史开始	Schulte

File name of this documentation: *APROL_R32_B2_ProjectEngineering_086.pdf*

公司总部

Bernecker + Rainer Industrie-Elektronik Ges.m.b.H.

B&R Strasse 1

5142 Eggelsberg

Austria

Tel.: +43 (0) 77 48 / 65 86 - 0

Fax: +43 (0) 77 48 / 65 86 - 26

info@br-automation.com

www.br-automation.com

在全球55个国家设有144个办事处 - www.br-automation.com/contact



Argentina • Australia • Austria • Belarus • Belgium • Brazil • Bulgaria • Canada • Chile • China • Croatia • Cyprus • Czech Republic
Denmark • Egypt • Emirates • Finland • France • Germany • Greece • Hungary • India • Indonesia • Ireland • Israel • Italy • Japan • Korea
Kyrgyzstan • Malaysia • Mexico • New Zealand • The Netherlands • Norway • Pakistan • Poland • Portugal • Romania • Russia • Singapore
Slovakia • Slovenia • South Africa • Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • USA