
目录

未找到目录项。

第一章 显示和存盘


MCGS 软件提供了丰富的报表和曲线功能，用以实现实时报表、历史报表、实时曲线、历史曲线等功能，本章主要学习这些功能的组态方法，其中讲到了模拟设备的应用，实时报表、实时曲线显示，如何实现数据的存盘，存盘数据如何生成历史报表和历史曲线，最后还会实际操作人机界面中存盘数据的 U 盘导出功能。

第1节 数据准备

模拟设备是 MCGS 软件根据设置的参数产生一组模拟曲线的数据，以供用户调试工程使用。可以产生标准的正弦波，方波，三角波，锯齿波信号，且其幅值和周期都可以任意设置。我们首先添加模拟设备，产生一组正弦信号。

1. 添加驱动



在工作台中激活设备窗口，鼠标双击 **设备窗口** 进入设备组态画面，点击工具条中的  打开“设备工具箱”。

打开“设备工具箱”上的“设备管理”按钮，在通用设备分类下，找到“模拟设备”，添加到选定设备中。

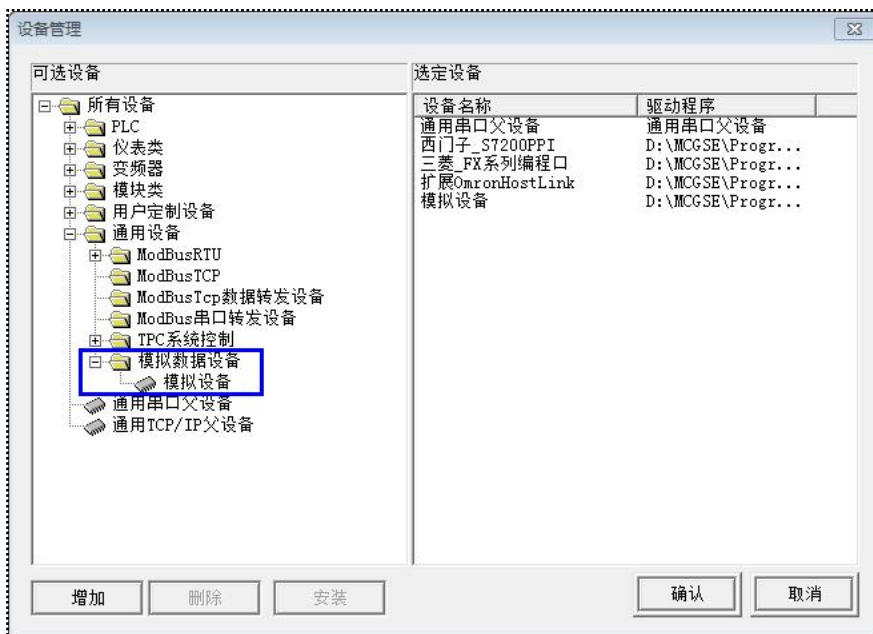


图 1-1-1

在“设备工具箱”中的双击“模拟设备”，驱动即被添加到设备窗口中。驱动添加完成后显示效果如下图所示。

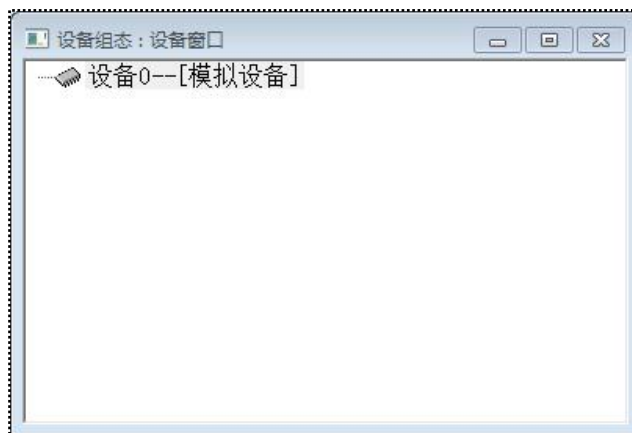


图 1-1-2

2. 连接变量

双击“设备 0--[模拟设备]”打开“设备编辑窗口”。点击 **快速连接变量** 按钮，打开“快速连接”对话框。将通道个数修改为 2 个，确认。



图 1-1-3 快速连接

驱动通道 0 和通道 1 分别连接了两个变量 Data00、Data01。

索引	连接变量	通道名称	通道处理
0000	Data00	通道0	
0001	Data01	通道1	
0002		通道2	
0003		通道3	
0004		通道4	

图 1-1-4

点击“设置设备内部属性”，将通道 1 的正弦信号周期修改为 20 秒。

设备属性名	设备属性值
[内部属性]	设置设备内部属性 ...

图 1-1-5

通道	曲线类型	数据类型	最大值	最小值	周期(秒)
1	0 - 正弦	1 - 浮点	1000	0	10
2	0 - 正弦	1 - 浮点	1000	0	20
3	0 - 正弦	1 - 浮点	1000	0	10

图 1-1-6


点击  按钮，软件出现“添加数据对象”的提示，点击“全部添加”，即可将 Data00、Data01 两个变量在 MCGS 组态工程中进行定义。



图 1-1-7

至此，数据准备部分已经完成。点击“设备属性设置”的“启动设备调试”，可以看到模拟设备按照正弦信号产生的调试数据。

索引	连接变量	通道名称	通道处理	调试数据
0000	Data00	通道0		247.3
0001	Data01	通道1		630.8
0002		通道2		247.3
0003		通道3		247.3



图 1-1-8

第2节 实时报表

实时报表是对变量瞬时值的反映，通常用于将当前时间的数据变量按一定报表格式显示和打印出来。实时报表可以通过 MCGS 系统的自由表格构件组态实现。这一节，我们要通过自由表格构件实现一个简单的实时报表的功能。

1. 添加自由表格

(1)进入用户窗口，点击“新建窗口”，修改窗口名字为“报表”。

(2)双击打开“报表”用户窗口，点击  打开“工具箱”，选取自由表格构件图标 ，在桌面适当位置，绘制一个表格。

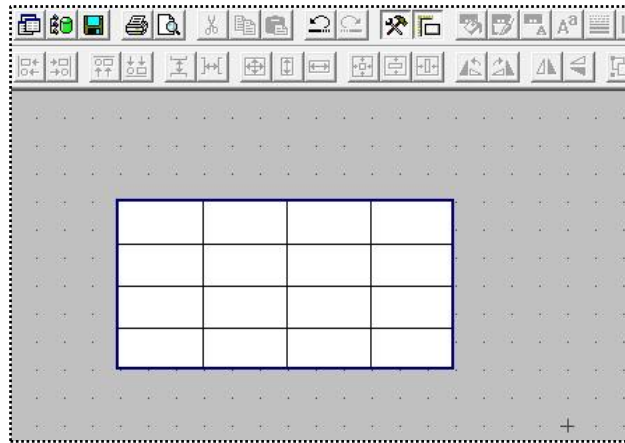
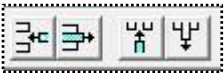


图 1-2-1

(3) 双击表格进入编辑状态，通过表格编辑条 ，将表格调整为 2 行 2 列。

(4) 自由表格改变单元格大小的方法和微软 EXCEL 表格的编辑方式相同，即在编辑状态，把鼠标指针移到 A 与 B 或 1 与 2 之间，当鼠标指针呈分隔线形状时，拖动鼠标至所需大小即可。

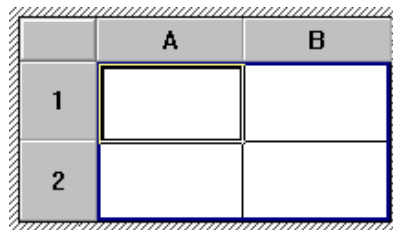


图 1-2-2

2. 变量链接

(1) 双击表格进入编辑状态。在 A 列的两个单元格中分别输入：变量 1、变量 2。



图 1-2-3

(2) 在 B 列中，选中变量 1 对应的单元格，单击右键。从弹出的下拉菜单中选取“连接”项。如下图 2-1-4 所示。自由表格进入连接模式，表格的行号和列号后面加“*”，如图 2-1-5 所示。

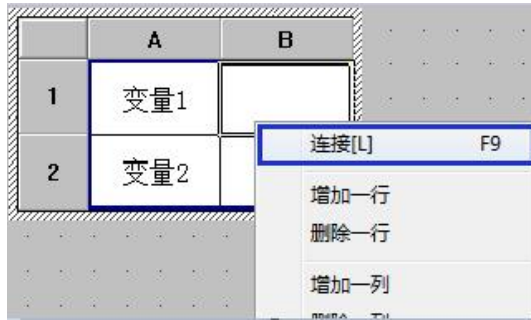


图 1-2-4

连接	A*	B*
1*		
2*		

图 1-2-5

(3)在 B1 单元格点击右键，弹出的变量选择对话框选择 Data00 变量。同样，在 B2 单元格选择 Data01 变量。

连接	A*	B*
1*		Data00
2*		Data01

图 1-2-6

(4)至此，自由表格的链接结束。我们已经完成了一个简单的实时报表的组态。模拟运行程序，可以看到实时报表实时显示 Data00 和 Data01 的当前值。效果图如下所示：

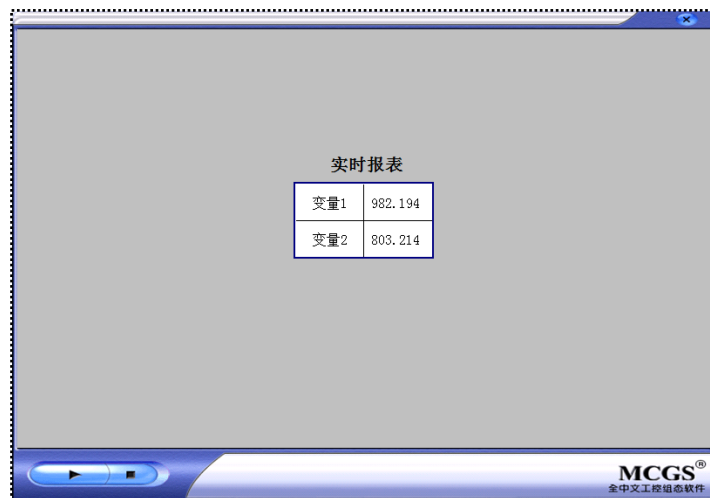


图 1-2-7

第3节 历史报表

历史报表通常用于从历史数据库中提取数据记录，并以一定的格式显示历史数据。实现历史报表有两三种方式：

- 第一种是用动画构件中的“历史表格”构件；
- 第二种是用动画构件中的“存盘数据浏览”构件。

本节，我们以使用率更高的“历史表格”构件为例，来讲解如何实现历史报表功能。

1.3.1 数据存盘

MCGS 把数据的存盘处理作为数据对象的一个属性，封装在变量的属性中。在 MCGS 嵌入版中，需要用组对象类型的变量进行存盘。组对象类型是 MCGS 引入的一种特殊类型的数据对象，类似于一般编程语言中的数组和结构体，用于把相关的多个数据对象集合在一起，作为一个整体来定义和处理。

我们要实现 Data00 和 Data01 两个数据对象的定时存盘功能。具体操作步骤如下：

(1) 点击 **实时数据库** 进入实时数据库，点击右侧的 **新增对象** 按钮，在实时数据库中新建一个数据对象。双击打开数据对象的属性设置页，对象名称修改为“数据组”，对象类型修改为“组对象”。



图 1-3-1

(2)打开数据对象属性设置页的“组对象成员”子页，将需要存盘的 Data00、Data01 两个数据对象添加到组对象成员中。



图 1-3-2

(3)打开数据对象属性设置页的“存盘属性”子页，设置存盘为定时存盘，存盘周期为 10 秒。点击“确认”按钮。



图 1-3-3

至此，Data00 和 Data01 两个数据对象的定时存盘功能我们就完成了。在实时数据库中，在存盘列我们可以看到“数据组”组对象的存盘信息。

名字	类型	注释	报警	存盘
数据组	组对象			存盘周期: 2秒
InputUser2	字符型	系统内建...		
InputUser1	字符型	系统内建...		
InputSTime	字符型	系统内建...		


图 1-3-4

1.3.2 历史报表

MCGS 软件的历史表格构件主要功能是连接存盘数据源,运行时动态的显示存盘数据源中的存盘记录的值。可以多页显示和单页显示,并可以实现按照时间条件或是数值条件进行查询的功能。本节,我们实现用历史表格构件实现多页显示“数据组”组对象的存盘数据。

1. 添加历史曲线

(1)进入“报表”用户窗口。

(2)在“工具箱”中选取历史表格构件图标,在桌面适当位置,绘制一个表格。将表格调整为 10 行 3 列。

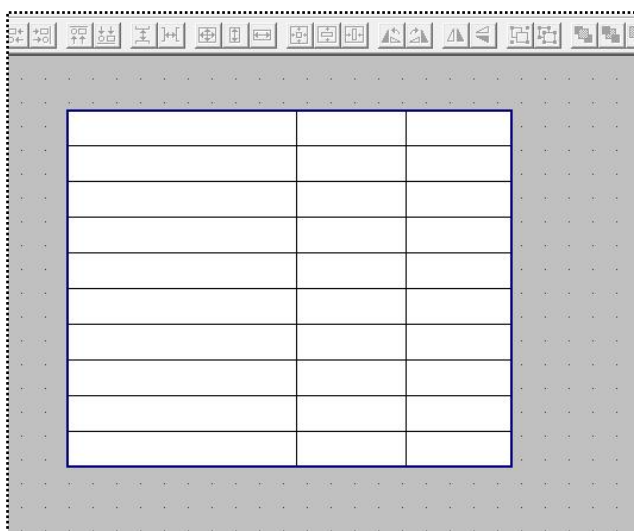


图 1-3-5

(3)双击表格进入编辑状态。在第一列的三个单元格中分别输入:存盘时间、变量 1、变量 2。

	C1	C2	C3
R1	存盘时间	变量1	变量2
R2			
R3			
R4			
R5			
R6			
R7			
R8			
R9			
R10			

图 1-3-6

2. 连接数据源

(1)选中 C1R2 至 C3R10 所有单元格，在选中的单元格区域单击右键。从弹出的下拉菜单中选取“连接”项。如下图 1-3-7 所示。历史表格进入连接模式，表格的行号和列号后面加“*”，如图 1-3-8 所示。



图 1-3-7

连接	C1*	C2*	C3*
R1*			
R2*			
R3*			
R4*			
R5*			
R6*			
R7*			
R8*			
R9*			
R10*			

图 1-3-8

(2)点击菜单栏中的“表格”菜单，选择“合并表元”项，所选单元格区域会出现反斜杠。

连接	C1*	C2*	C3*
R1*			
R2*			
R3*			
R4*			
R5*			
R6*			
R7*			
R8*			
R9*			
R10*			

图 1-3-9

(3)双击反斜杠区域，弹出数据库连接设置对话框，具体设置如下：

- 基本属性页中，连接方式选取：在指定的表格单元内，显示满足条件的数据记录；按照从上到下的方式填充数据行；显示多页记录。

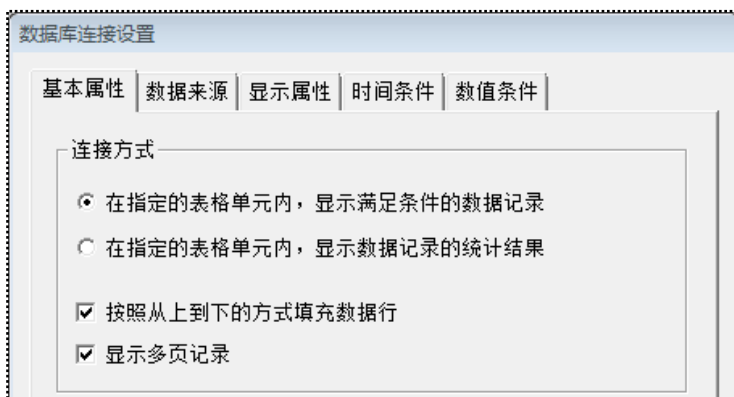


图 1-3-10

- 数据来源页中，选取组对象对应的存盘数据；组对象名为：数据组。



图 1-3-11

- 显示属性页中，点击“复位”按钮。



图 1-3-12

- 时间条件页中，排序列名: MCGS_TIME; 降序; 时间列名: MCGS_TIME; 所有存盘数据。

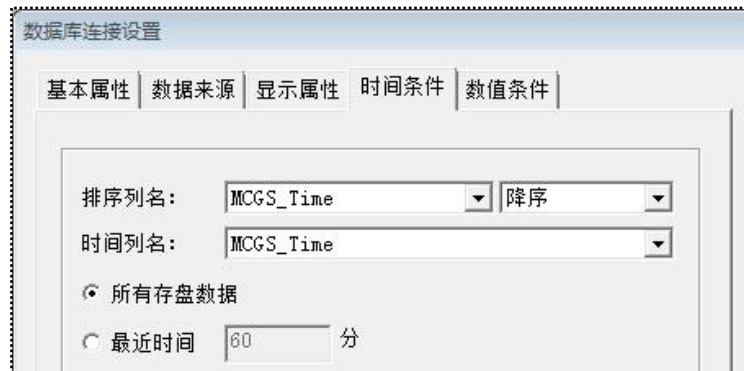


图 1-3-13

(4)至此，历史曲线的功能组态已经完成。最后，我们需要在“报表”窗口的循环脚本中，调用窗口的 Refresh 方法，以使历史表格构件定时装载数据，实现自动刷新。



图 1-3-14

模拟运行程序，可以看到历史报表显示数据组对象的存盘数据。实时报表和历史报表的对比显示效果图下所示：




图 1-3-15

第4节 实时曲线

实时曲线构件是从实时数据库中读取数据，用曲线显示一个或多个数据对象数值的动画图形，实时记录数据对象值的变化情况。本节，我们学习用实时曲线构件显示 Data00 和 Data01 两个数据对象的趋势图。

操作步骤如下：

(1)进入用户窗口，点击“新建窗口”，修改窗口名字为“曲线”。

(2)双击打开“曲线”用户窗口，在“工具箱”中选取实时曲线构件图标, 在桌面适当位置，添加实时曲线。

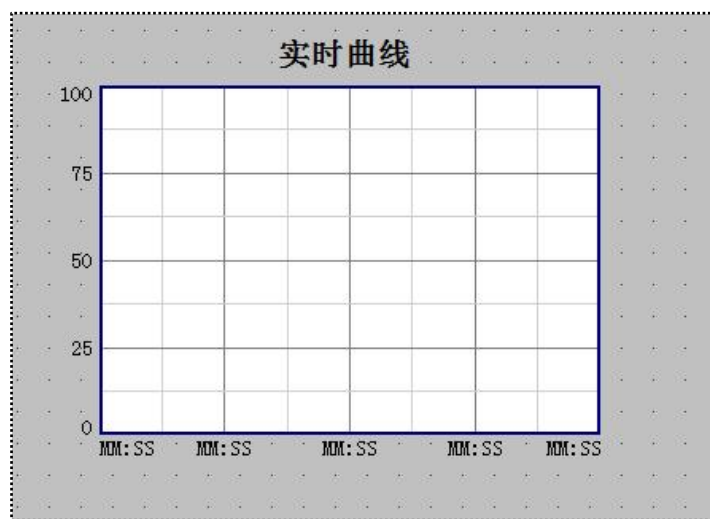


图 1-4-1

(3)双击实时曲线，打开实时曲线构件属性设置对话框，具体设置如下：

- 在基本属性页中，默认设置不需要修改。
- 在标注属性页中，时间单位设为秒钟，最大值修改为 1000，其它不变。



图 1-4-2

- 在画笔属性页中，设置：
 - ◇ 曲线 1 对应的表达式设为 Data00，颜色为蓝色；
 - ◇ 曲线 2 对应的表达式设为 Data01，颜色为红色。



图 1-4-3

(4)实时曲线设置完成。模拟运行可以看到周期为 10 秒的 Data00 的正弦曲线和周期为 20 秒的 Data01 的正弦曲线。效果图如下：

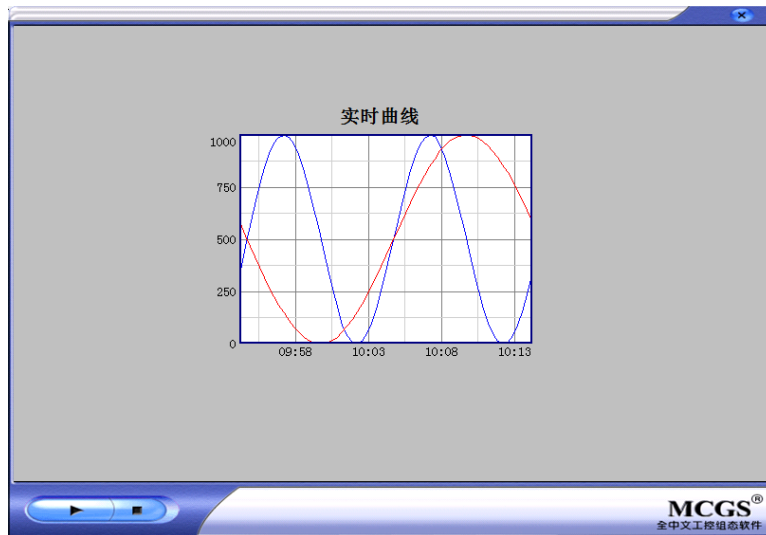



图 1-4-4

第5节 历史曲线

MCGS 软件中历史曲线是将存盘数据从数据库中读出，以时间为 X 轴，数据值为 Y 轴进行曲线绘制。工艺上，历史曲线主要用于事后查看数据的状态变化趋势和总结规律。本节，我们要用历史曲线构件将数据组中的 Data00 和 Data01 两个数据对象的存盘数据的趋势图进行显示。

1. 添加历史曲线

(1) 进入“曲线”用户窗口。

(2) 在“工具箱”中选取历史曲线构件图标，在桌面适当位置绘制一个历史曲线构件。

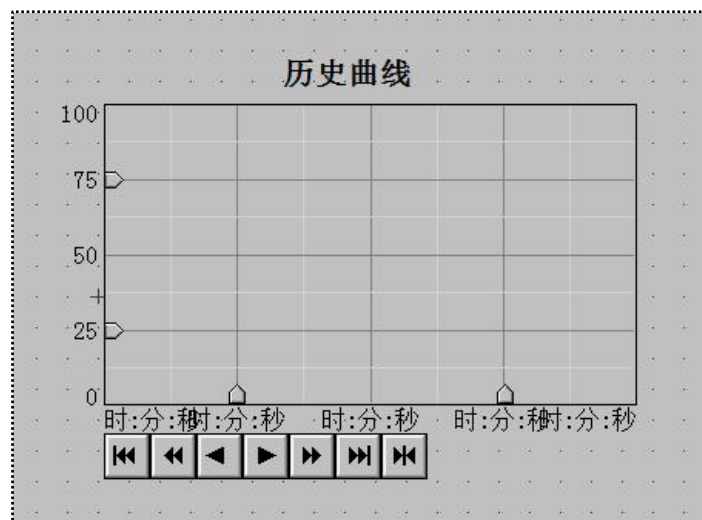


图 1-5-1

(3)在历史曲线矩形框的下方有一排按钮，按钮上有前进、后退、快进、快退、到最后、到最前以及曲线设置和时间设置按钮。这些按钮是历史曲线操作的缺省按钮。

2. 添连接数据源

(1)双击历史曲线构件，打开历史曲线构件属性设置对话框，进行如下设置：

- 在基本属性页中，将背景颜色设为白色；
- 在存盘数据页中，历史存盘数据来源选择组对象对应的存盘数据，并在下拉菜单中选择数据组；



图 1-5-2

- 在标注设置页中，坐标长度修改为 10，时间单位修改为分；

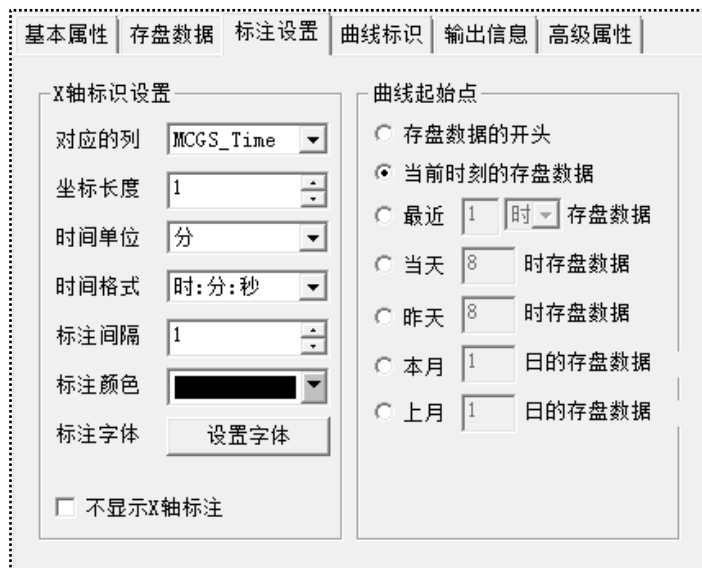


图 1-5-3

■ 在曲线标识页中：

◇ 选中曲线 1，曲线内容设为 Data00，曲线颜色设为蓝色，最大值设为 1000，实时刷新修改为 Data00；其它不变；

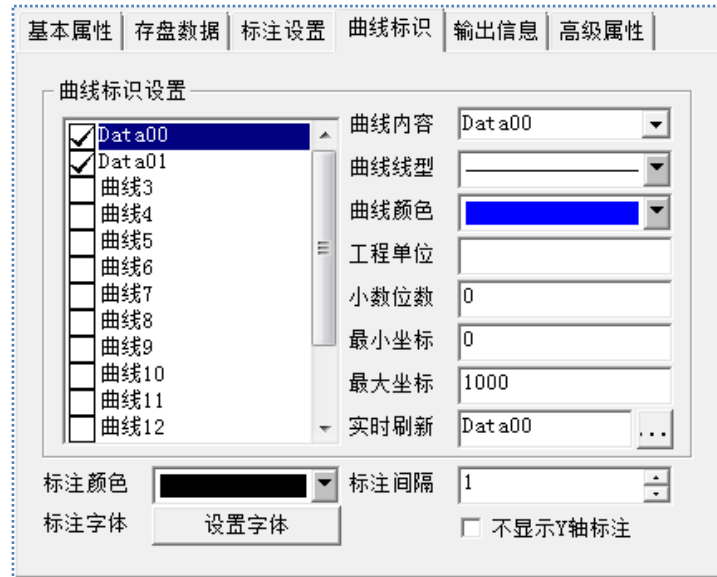


图 1-5-4

◇ 同样，选中曲线 2，曲线内容设为 Data01，曲线颜色设为红色，最大值设为 1000，实时刷新修改为 Data01；其它不变。

■ 在高级属性页中，勾选上运行时自动项。

(2)至此，我们就将历史曲线的组态工作完成了。模拟运行程序，可以看到历史报表显示数据组对象的存盘数据。实时曲线和历史曲线的对比显示效果图如下所示：

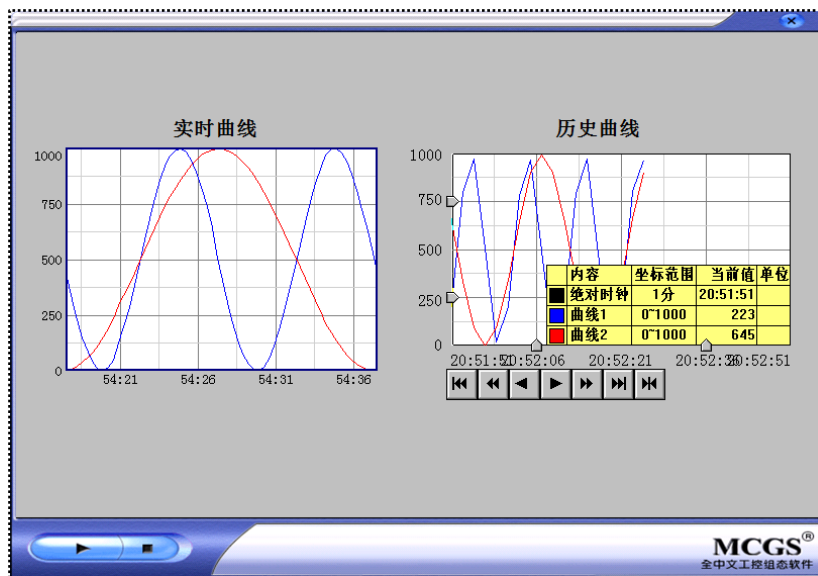


图 1-5-5

其中，历史曲线有几个功能需要说明一下：

游标：游标是 X 轴和 Y 轴坐标线上的四个小图标。通过这个小图标，可以进行曲线的放大和缩小以及平移操作。

光标：光标是曲线区域中的一根线，随着鼠标移动，在信息显示窗口中，显示了光标当前指向的时间，以及此刻这些趋势点的值。

信息显示窗口：运行时显示光标所在位置的时间点和各条曲线的值，如下图所示。

	内容	坐标范围	当前值	单位
■	绝对时钟	59秒	21:01:02	
■	曲线1	0~1000	485	
■	曲线2	0~1000	492	

图 1-5-6

第6节 U 盘导出

MCGS 软件提供了将存储在人机界面存储介质中的历史存盘数据导出到 U 盘的功能，并且将数据转换成 CSV 格式，导出的数据表格在电脑上可以直接用 Excel 打开查看编辑。本节，我们将结合实际操作学习这一功能。

1.6.1 U 盘导出函数

U 盘导出功能需要用到 MCGS 中的系统函数!ExportHisDataToCSV()，在帮助中索引该函数，可以看到函数功能及参数说明。我们需要注意函数的各参数意义和类型。

!ExportHisDataToCSV(文件名,组对象名,字段名,开始时间,结束时间,最大记录数,导出模式,导出参数,进度指示数据对象名,取消控制数据对象名)


函数意义：

- 1.导出指定组对象存盘数据，导出的条件包括：开始时间-结束时间，最大导出总条数；
- 2.导出字段列表，组对象为MCGS_ALARM则表示导出报警历史数据；
- 3.可指定导出文件名和路径，主要是U盘导出，可以追加方式也可覆盖方式导出；
- 4.可显示导出过程进度情况；
- 5.可取消和中止长时间的导出过程；
- 6.导出格式为csv文件，导出的第一行为字段名字。

图 1-6-1

1.6.2 U 盘导出函数

操作步骤如下：

(1)在“报表”窗口中，在报表的下方新建一个按钮构件，并将按钮的文本修改为“U 盘导出”。

(2)双击按钮打开按钮构件属性设置，在脚本程序的抬起脚本里输入 U 盘导出函数“!ExportHisDataToCSV("\usbharddisk\数据.csv","数据组","", "2014/1/1 0:0:0","2015/1/1 0:0:0",20000,1,"",进度,控制)”。

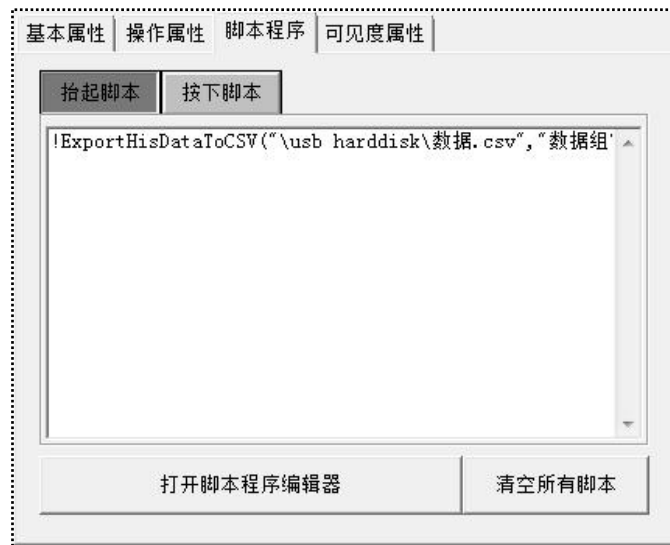


图 1-6-2

(3)点击按钮属性页的确认按钮，此时，会弹出未知对象的组态提示，点击“是”，定义“进度”和“控制”两个数据对象，变量类型均为开关型。




图 1-6-3

(4)函数意义:

!ExportHisDataToCSV("\usbharddisk\ 数据.csv"," 数据组 ","","2014/1/1 0:0:0","2015/1/1 0:0:0",20000,1,"",进度,控制)

- ◇ 表示将数据组中的存盘数据导出到 TPC 的 U 盘目录\usbharddisk\下
- ◇ 目标文件是数据名为“数据.csv”
- ◇ 导出的字段名为空，表示导出所有字段的数据
- ◇ 导出数据的开始时间为 2014/1/1 0:0:0，结束时间为 2015/1/1 0:0:0
- ◇ 最大导出条数为 20000 条
- ◇ 导出方式是覆盖方式

(5)模拟运行，点击（下载配置）按钮，弹出下载配置对话框。将程序下载到屏中，点击 U 盘导出按钮。执行结果如下图：

历史报表

存盘时间	变量1	变量2
2014-01-09 10:37:50	22.668	97.2365
2014-01-09 10:37:48	483.055	0.071531
2014-01-09 10:37:46	977.528	97.4518
2014-01-09 10:37:44	797.184	343.568
2014-01-09 10:37:42	218.155	647.485
2014-01-09 10:37:40	27.1686	906.993
2014-01-09 10:37:38	492.791	999.987
2014-01-09 10:37:36	969.694	909.703
2014-01-09 10:37:34	789.296	651.815

实时报表

变量1	230.472
变量2	359.57

U盘导出



文件	开始	插入	页面布局	公式	数据	审阅	视图
J1415							
	A	B	C	D	E		
393	2014/1/9 9:45	9	483.369	0.068899			
394	2014/1/9 9:45	6	22.668	97.2365			
395	2014/1/9 10:37	778	789.296	651.815			
396	2014/1/9 10:37	775	969.694	909.703			
397	2014/1/9 10:37	771	492.791	999.987			
398	2014/1/9 10:37	768	27.1686	906.993			
399	2014/1/9 10:37	765	218.155	647.485			
400	2014/1/9 10:37	777	797.184	343.568			
401	2014/1/9 10:37	774	977.528	97.4518			
402	2014/1/9 10:37	771	483.055	0.071531			
403	2014/1/9 10:37	768	22.668	97.2365			
404	2014/1/9 10:37	780	202.285	343.223			
405	2014/1/9 10:37	777	808.68	663.294			
406	2014/1/9 10:37	774	973.046	906.801			
407	2014/1/9 10:37	786	502.844	999.998			
408	2014/1/9 10:38	783	18.566	898.43			
409							

第二章 权限和安全

MCGS 嵌入版组态软件提供了一套完善的安全机制,用户能够自由组态进入和退出系统的操作权限,只允许有操作权限的操作员才能对某些功能进行操作。MCGS 嵌入版还提供了工程密码功能,来保护使用 MCGS 嵌入版组态软件开发所得的成果,开发者可利用这些功能保护自己的合法权益。

本章,我们将学习用户权限的设置、用户窗口的进入权限、工程密码、组态工程中的运行期限功能以及工程下载保护等内容。

第1节 用户权限

MCGS 嵌入版系统的操作权限机制和 Windows NT 类似,采用用户组和用户概念来进行操作权限的控制。在 MCGS 嵌入版中可以定义多个用户组,每个用户组中可以包含多个用户,同一个用户可以隶属于多个用户组。

操作权限的分配是以用户组为单位来进行的,即某种功能的操作哪些用户组有权限,而某个用户能否对这个功能进行操作取决于该用户所在的用户组是否具备对应的操作权限。如此,能方便地建立多层次的安全机制。

3. 定义用户组 and 用户


新建 MCGS 组态工程,选取“工具”菜单中的“用户权限管理”菜单项,弹出如下用户管理窗口:

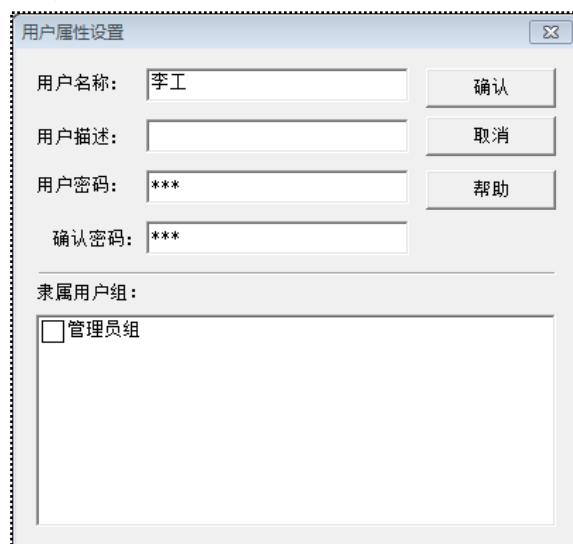


图 2-1-1

可以看到,用户管理器里固定有一个名为“管理员组”的用户组和一个名为

“负责人”的用户，它们的名称不能修改。管理员组中的用户有权利在运行时管理所有的权限分配工作，管理员组的这些特性是由 MCGS 嵌入版系统决定的，其它所有用户组都没有这些权利。

点击  按钮，新增一个用户名为“李工”的用户，密码为“111”。



用户属性设置

用户名称: 李工

用户描述:


用户密码: ***

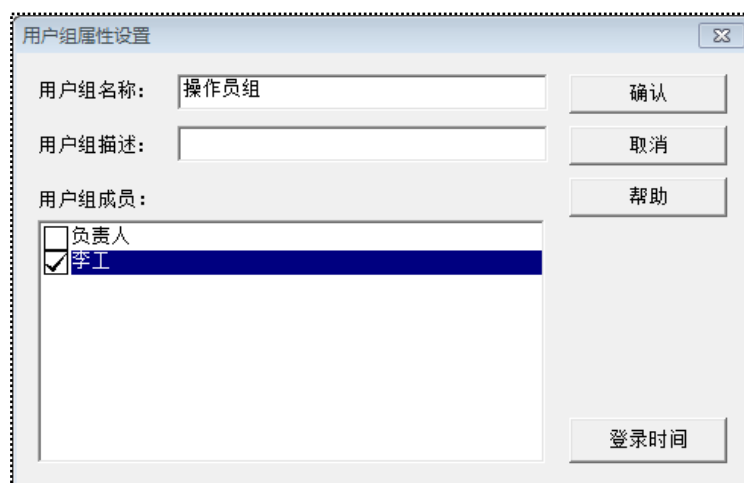
确认密码: ***

隶属用户组:

管理员组

图 2-1-2

选择管理员组，点击  按钮，新增一个“操作员组”，用户组成员选择“李工”。



用户组属性设置

用户组名称: 操作员组

用户组描述:

用户组成员:




负责人

李工

图 2-1-3

4. 系统权限设置

为了更好的保证工程运行的安全、稳定可靠，防止与工程系统无关的人员进入或退出工程系统，MCGS 嵌入版系统提供了对工程运行时进入和退出工程的权限管理。




 点击 **主控窗口** 进入主控窗口，选择 **主控窗口** 图标，再点击 **系统属性** 按钮打开主控窗口属性设置，如下窗口：

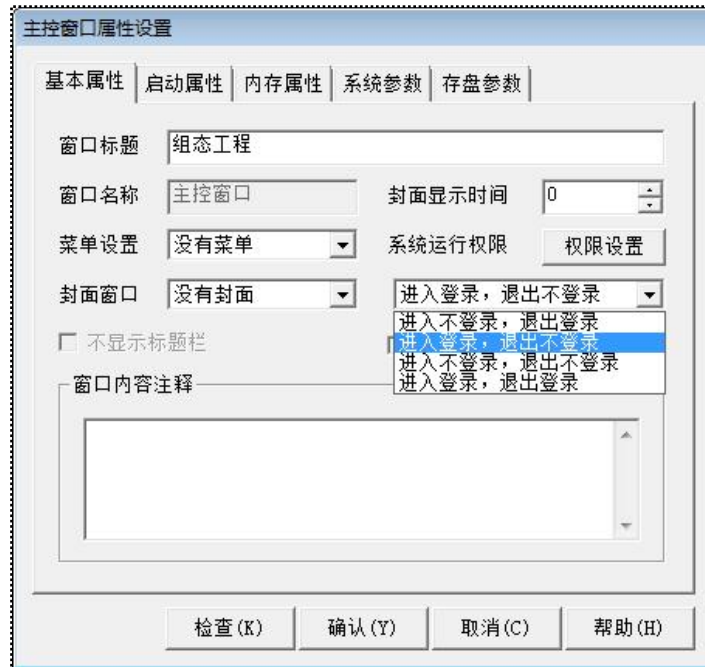


图 2-1-4

在基本属性页中，设置为“进入登录，退出不登录”，确认。


点击工具栏的下载配置按钮 ，弹出下载配置对话框，选择模拟运行，点击工程下载，待提示下载成功后启动工程，如下图所示，可以看到启动时自动弹出用户登录对话框，需要正确登录用户后才能进入运行窗口。



图 2-1-5

如果登录的用户信息不正确，会提示“用户未授权，退出系统”，而不能进入工程界面。

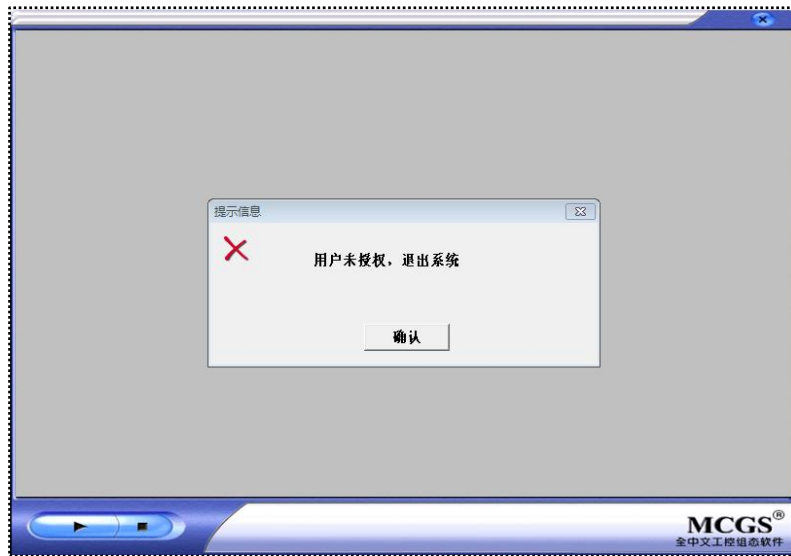


图 2-1-6


第2节 窗口权限


工艺上，组态工程中的一些厂家参数或是生产参数等重要的用户窗口不允许现场操作工随意进入操作，进入窗口时需要登录级别更高的用户。本节我们就实现用户窗口的权限。有两种方法可以实现这一功能：利用按钮构件权限和利用用户登录函数，下面我们分别讲解。

2.2.1 利用按钮构件权限

具体操作步骤如下：

(1)新建两个用户窗口，修改窗口名为“主窗口”、“参数设置”。

(2)双击打开“主窗口”用户窗口，新建一个按钮构件，并将按钮的文本修改为“参数设置”。

(3)双击“参数设置”按钮打开按钮属性设置，点击打开用户权限设置对话框，选择“管理员组”，表示只有管理员组才有权限才操作此按钮。其他用户登录时按钮无效。

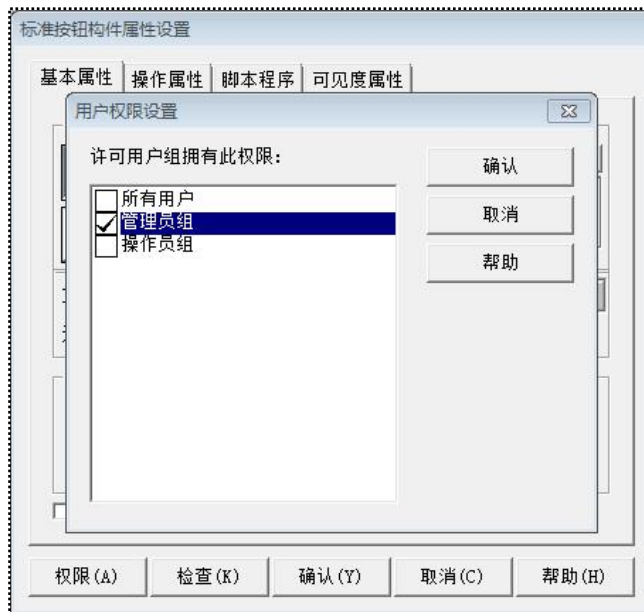


图 2-2-1

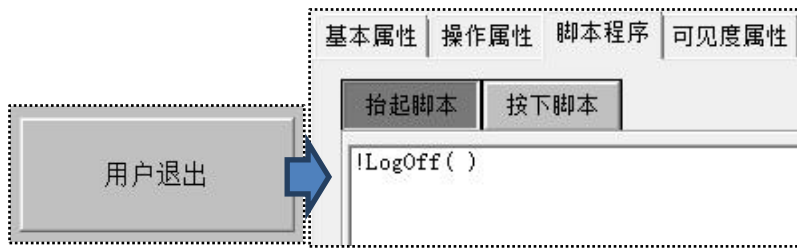
(4)操作属性页，选择打开用户窗口，窗口选择参数设置窗口。



图 2-2-2

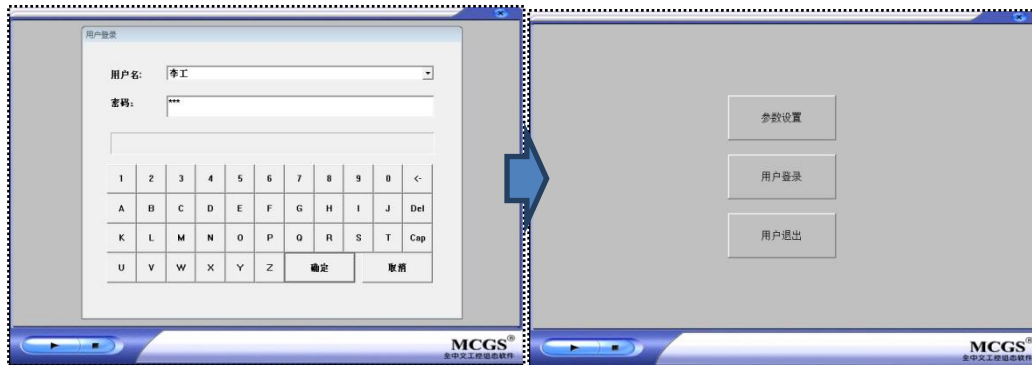
(5)如此，便完成了打开参数设置窗口的按钮权限设置。配合这一功能，我们还需要做两个按钮，用脚本函数!LogOn()、!LogOff()来实现用户的登录和退出。



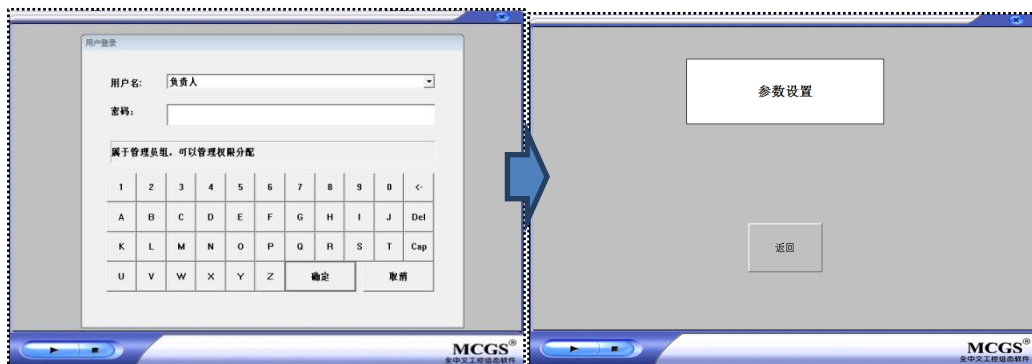


模拟运行程序，可以看到，没有登录用户时或是登录操作员组，点击参数设置按钮时没有任何反应。只有当登录的用户时管理员组时，才可以正常点击按钮打开参数设置用户窗口。

操作员组登录，点击“参数设置”按钮无反应：




管理员组登录，点击“参数设置”按钮进入参数设置窗口：



2.2.2 利用用户登录函数

具体操作步骤如下：

(1)打开“主窗口”用户窗口，新建一个按钮构件，并将按钮的文本修改为“参数设置”。

(2)双击打开按钮属性页，在脚本程序的抬起脚本输入以下脚本：

```

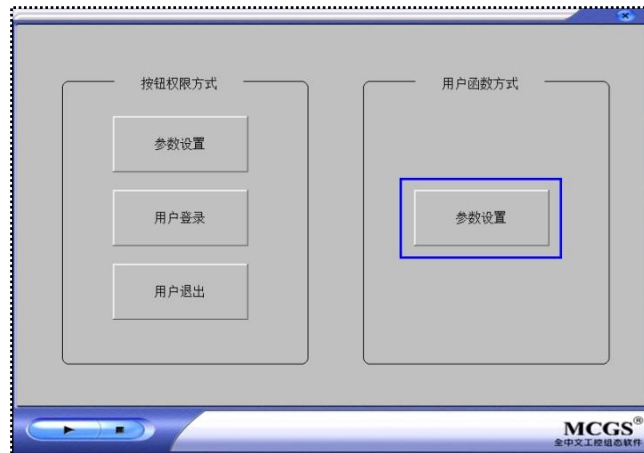
IF !LogOn( )=0 and !StrComp(!GetCurrentGroup( ),"管理员组")=0 THEN
用户窗口.参数设置.Open( )
ENDIF

```

脚本意义：

- ✧ 点击按钮时，弹出用户登录对话框
- ✧ 如果用户登录成功，且当前登录的用户组是管理员组，则打开到参数设置窗口

(3)模拟运行可以看到如下效果：



点击“参数设置”按钮：



以管理员组登录后，自动进入了参数设置窗口：



以上就是实现窗口权限的两种方式，第一种方式只要当前登录的用户权限足够就可以点击按钮进入参数设置窗口；第二种方式则每次点击参数设置按钮都需要登录一次，登录权限足够才可以进入参数设置窗口。

第3节 工程密码

工程密码设置是 MCGS 软件提供的一个保护工程文件的功能。给正在组态或已完成的工程设置密码，可以保护该工程不被其它人打开使用或修改。当使用 MCGS 嵌入版来打开这些工程时，首先弹出输入框要求输入工程密码，如密码不正确则不能打开该工程，从而起到保护劳动成果的作用。

在 MCGS 工作台，选择工具菜单“工具”-“工程安全管理”-“工程密码设置”选项。



图 2-3-1

打开“修改工程密码”对话框，旧密码为空，在新密码、确认新密码输入框内输入“666”，点击确认。



图 2-3-2

完成工程密码设置后，我们可以测试一下 MCGS 的安全管理，首先我们关闭当前工程，重新打开工程，此时弹出一个“输入工程密码”对话框。此时如果输入密码“666”，确认后可以打开工程。

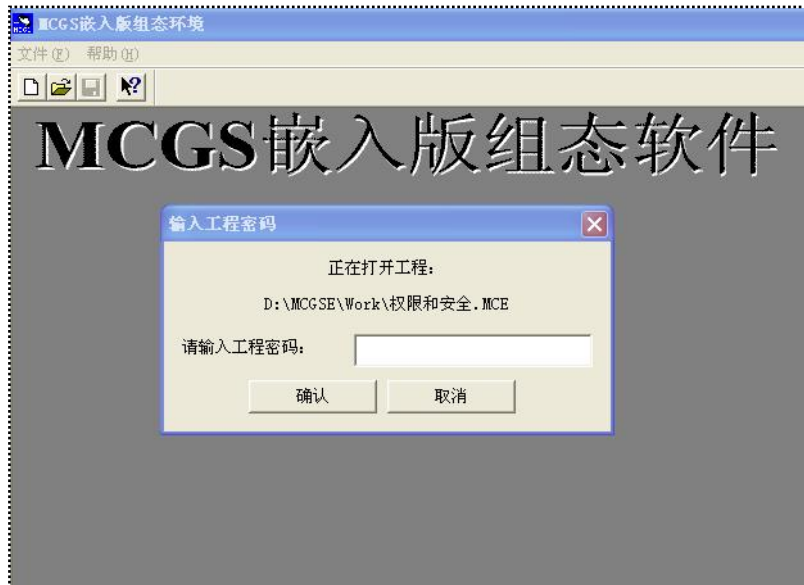


图 2-3-3

第4节 运行期限

因为商务或款项等原因，有时需要在 MCGS 组态工程中实现运行期限的功能，组态工程运行到设定的日期后，系统会自动提示，如不能输入正确密码，则组态工程停止运行。如此来保护 MCGS 组态工程开发者的合法权益。本小节，我们以实例来讲解如何在 MCGS 组态软件通过脚本程序实现运行期限的功能。

2.4.1 窗口和变量准备

1. 添加窗口和构件

- (1)新建组态工程，工程名为“运行期限”。
- (2)新建 2 个用户窗口，窗口名分别修改为“主画面”、“运行期限”。
- (3)打开“主画面”窗口，增加“标签”构件，文本内容输入“主画面”。

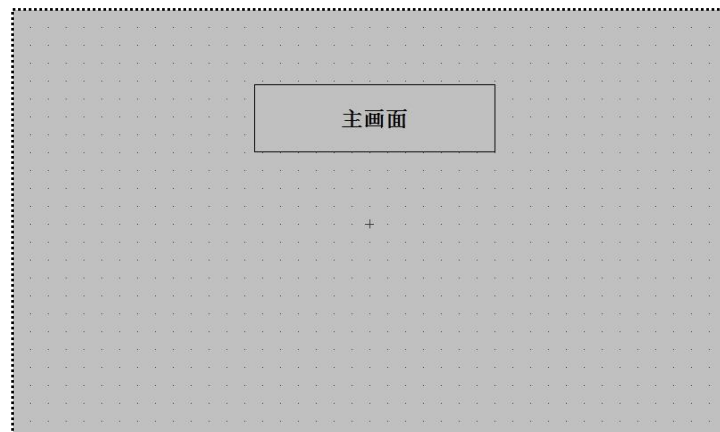


图 2-4-1

(3)打开“运行期限”窗口，增加“标签”、“输入框”、“标准按钮”构件，构件文本内容和布局如下：

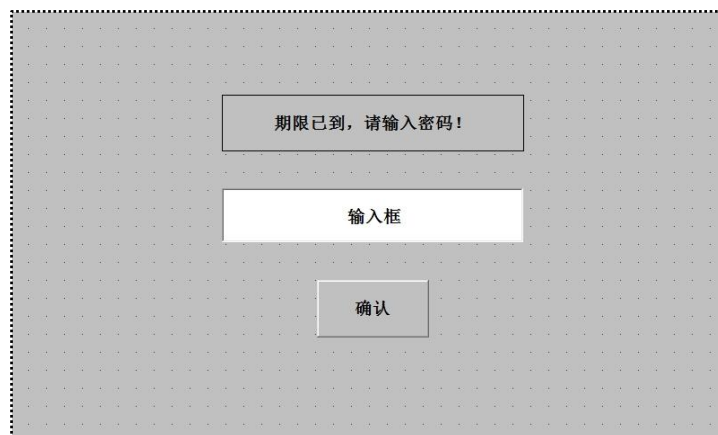


图 2-4-2

2. 创建数据对象

为了实现运行期限功能，我们需要创建以下数据对象：

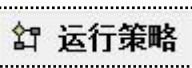
- ◇ 标识开关型变量初始值为 0
- ◇ 密码字符型变量初始值为空

名字	类型	注释
密码	字符型	
标识	开关型	

图 2-4-3


2.4.2 编写功能脚本

1. 期限判断脚本

(1)首先要编写的期限判断的脚本。点击  运行策略 图标，进入运行策略，双击打开“循环策略”。

名字	类型	注释
启动策略	启动策略	当系统启动时运行
退出策略	退出策略	当系统退出前运行
循环策略	循环策略	按照设定的时间循环运行

图 2-4-4

(2)点击工具栏的新增策略行图标 ，新增一个策略行。

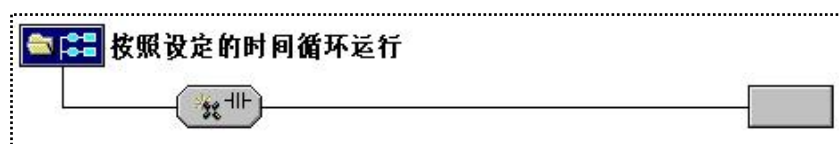




图 2-4-5

(3) 点击工具栏的 ，打开“策略工具箱”。点击“脚本程序”，将鼠标指针移到策略块图标  上，单击鼠标左键，添加脚本程序构件。

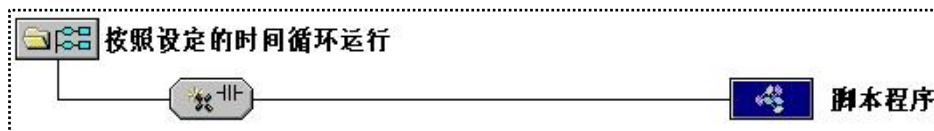


图 2-4-6

(4) 双击“脚本程序”，打开脚本程序编辑器。输入以下脚本：

```
IF !TimeSpanGetSeconds(!TimeGetSpan(!TimeStr2I($Date+" "+$Time),  
!TimeStr2I("2015-1-1 09:00:00")))>0 THEN
```

```
用户窗口.运行期限.Open()
```

```
ENDIF
```

脚本意义：

- ✧ 获取当前系统时间（ $\$Date+ " "+ \$Time$ ）到“2015-1-1 09:00:00”之间的时间差
- ✧ 如果到达设定的日期，那么打开“运行期限”窗口

(5) 双击打开策略行的条件表达式，设置为当“标识”变量值为 0 时才执行这个策略行。也就是，当“标识=1”时便不再执行(4)中期限判断脚本，相当于解除了运行期限的功能。



图 2-4-7

2. 解除期限密码

(1)进入“运行期限”窗口，双击输入框构件打开属性设置页，对应数据对象选择“密码”，勾选上“密码”选项。

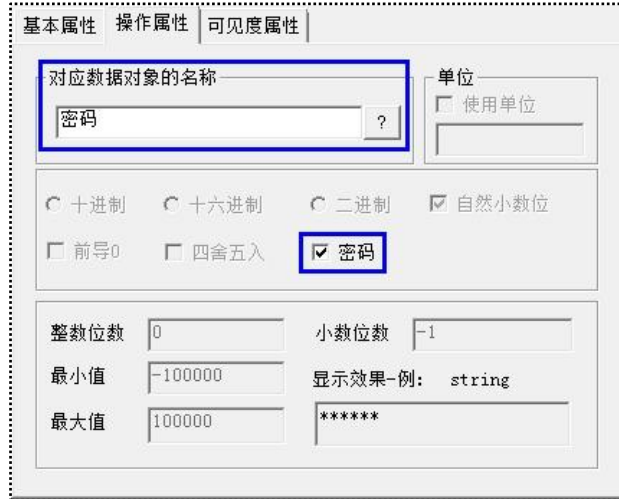


图 2-4-8

(2)双击标准按钮构件，打开脚本程序页，输入以下脚本：

```
IF !StrComp(密码,"ABC")=0 THEN  
用户窗口.主画面.Open()  
标识=1  
标识.SaveDataInitValue()  
!FlushDataInitValueToDisk()  
ENDIF
```

脚本意义：

- ✧ 初始密码为“ABC”
- ✧ 输入的密码和初始密码“ABC”作比较，如正确，则打开到“主画面”窗口，并执行“标识=1”使期限判断脚本不再执行
- ✧ “标识”数据对象的值掉电保持。即使重启，组态程序还是能维持解除运行期限功能的状态

至此，我们就完成了运行期限功能的所有组态工作。我们将设定期限适当调整，模拟运行可以看到如下效果：

启动工程之后，显示主画面：



运行期限到时，自动打开到运行期限已到的提示窗口：




此时，需要 MCGS 组态工程开发者提供密码，否则组态工程将不能正常使用。输入正确的密码后，则可以解除运行期限功能继续使用。

本实例，只讲解了运行期限基础功能的使用。用户还可以在基础上进行修改和增加特殊功能，比如，到达期限时停止组态工程与下位机设备的通讯等，可以举一反三地应用到 MCGS 组态工程中。

第5节 下载安全

开发好的组态工程需要下载到人机界面中使用，MCGS 软件提供了可以将人机界面的工程上传到电脑的功能。用户可以根据需要设置，如果要保护下载的程序不被他人上传使用，则需要注意不要激活工程上传功能。本节我们讲解一下如何才可以实现工程上传。

点击图标打开下载配置对话框，选择联机运行后，支持工程上传的选择激活，默认状态是不支持工程上传。此时，如果我们勾选上支持工程上传，那么下载到人机界面的工程可以通过 MCGS 组态软件上传到电脑中。

打开下载配置后，勾选支持工程上传，点击工程下载。

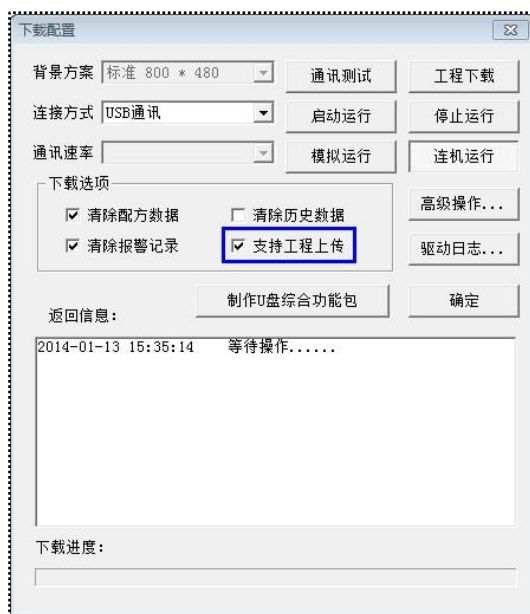


图 2-5-1

下载成功后，打开 MCGS 组态软件工作台，在文件下点击“上传工程”项。打开上传工程对话框，设置工程路径及连接方式，点击“开始上传”，即完成了工程上传的操作。



图 2-5-2

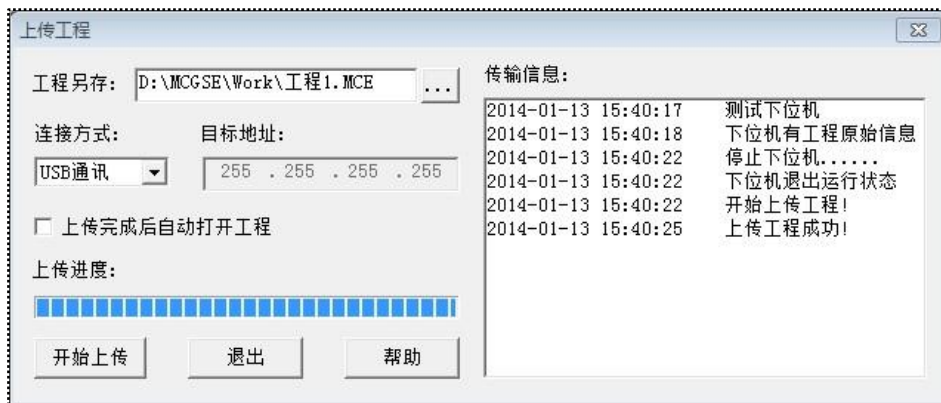


图 2-5-3

第三章 配方组态

MCGS 软件提供了配方组态工具，可以在组态工程中实现配方功能。


配方数据可以存储在 PLC 中，根据工艺选取配方数据，通过驱动采集将数据传送给 HMI 进行显示和使用，这种用法需要在 PLC 编程软件中实现。本章我们介绍配方数据存储存储在 HMI 中的用法。

第1节 配方使用

配方数据存储存储在 HMI 中，使用时选择其中的一条配方，将数据写到 PLC 中进行生产和计算。本节，我们以面包配方为例来讲解配方在组态工程中的应用。

5. 配方组态

(1)新建组态工程，工程名为“配方组态”。

(2)点击“工具”菜单下的“配方组态设计”菜单项，打开配方组态设计对话框。点击工具栏的新建配方图标自动创建一个配方组，将配方组名改名为“面包配方”。

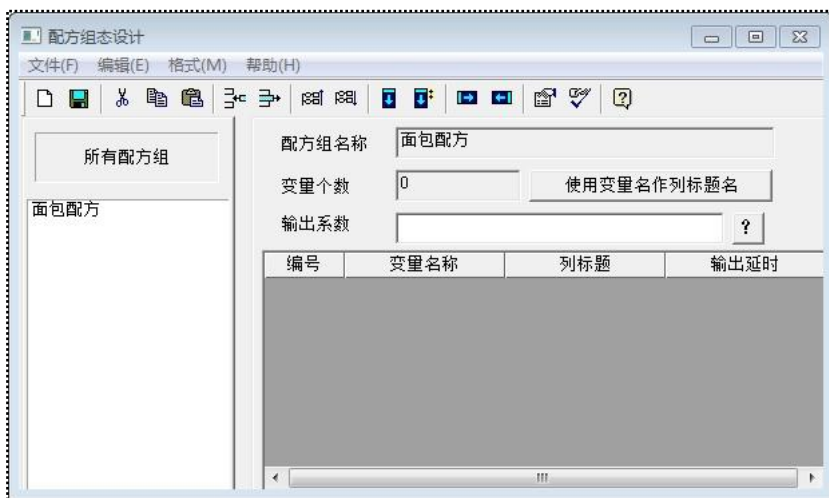



图 3-1-1

(3)面包配方需要面粉、水、糖、盐 4 种配料，我们在实时数据库中建立 4 个数据对象作为配方变量：

名字	类型	注释
 面粉	数值型	
 水	数值型	
 糖	数值型	
 盐	数值型	

图 3-1-2

(4) 点击工具栏的新增一行图标 ，给配方组增加四行，并在变量名称单元格分别输入“面粉”、“水”、“糖”、“盐”。点击“使用变量名作列标题名”。

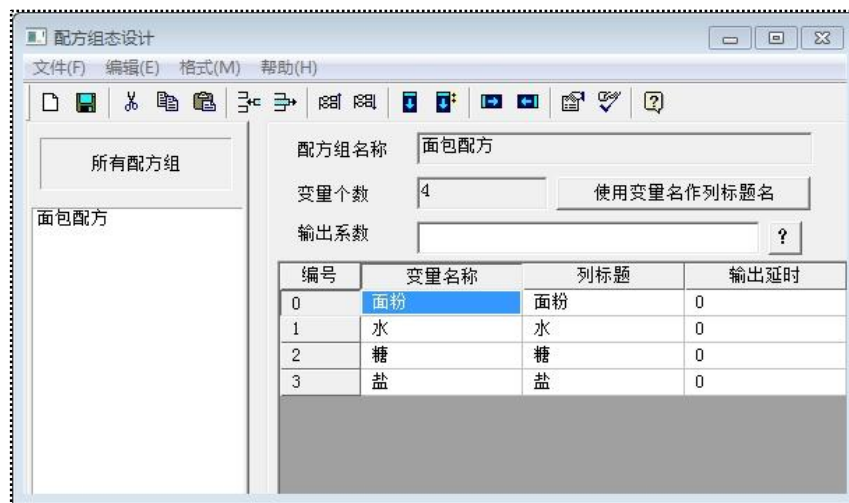


图 3-1-3

(5) 实例中，我们要做以下三种口味的面包，也就是对应三条配方数据，我们拟设定配方数据如下表所示：

	面粉	水	糖	盐
甜面包	80	30	80	10
低糖面包	80	30	30	5
无糖面包	80	30	10	5

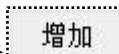
(6) 双击“面包配方”配方组，打开配方修改对话框。点击  按钮增加三条配方，按着(5)中列表所示的数据填写。保存后退出。



图 3-1-4

6. 窗口组态

(1) 根据初级教程学习的内容，我们在设备窗口创建西门子 200PPI 的驱动，

并添加 VW0、VW2、VW4、VW6 四个地址，作为 PLC 的配方数据地址。

索引	连接变量	通道名称	通道地址
0000		通讯状态	
0001	设备0_读写VWUB000	读写VWUB000	
0002	设备0_读写VWUB002	读写VWUB002	
0003	设备0_读写VWUB004	读写VWUB004	
0004	设备0_读写VWUB006	读写VWUB006	

图 3-1-5

(2)新建用户窗口，窗口名称为“配方使用”。

(3)打开“配方使用”窗口，增加标签和标准按钮构件，文字和布局如下：

配方名

面粉

水

糖

盐

图 3-1-6

- ✧ 配方名对应的标签关联变量“配方名”，定义为字符型变量，显示选择的配方名称
- ✧ 面粉对应的标签关联变量“面粉”，输出值类型：数数量
- ✧ 水对应的标签关联变量“水”，输出值类型：数数量
- ✧ 糖对应的标签关联变量“糖”，输出值类型：数数量
- ✧ 盐对应的标签关联变量“盐”，输出值类型：数数量

(4)双击“通过对话框选择配方”按钮，在脚本程序页输入以下脚本：

```
!RecipeLoadByDialog("面包配方","请选择一个面包配方")
```

```
配方名=!RecipeGetName("面包配方")
```

```
设备0_读写VWUB000=面粉
```

```
设备0_读写VWUB002=水
```

```
设备0_读写VWUB004=糖
```

设备0_读写VWUB006=盐

脚本意义：

- ◇ 运行时可以打开配方选择对话框，选择配方数据装载到配方变量中
- ◇ 将当前配方的配方名获取出来，赋值给“配方名”数据对象中
- ◇ 将当前配方的4个数据对象面粉、水、糖、盐的值分别写到PLC的对应地址中

至此，我们就完成了配方使用的组态工作，可以在组态工程运行时选择甜面包、低糖面包、无糖面包三种配方中选择，并将选择的配方的值赋值到面粉、水、糖、盐4个变量中进行显示和生产。

模拟运行，可以看到面粉、水、糖、盐的数值全部为初始值0。

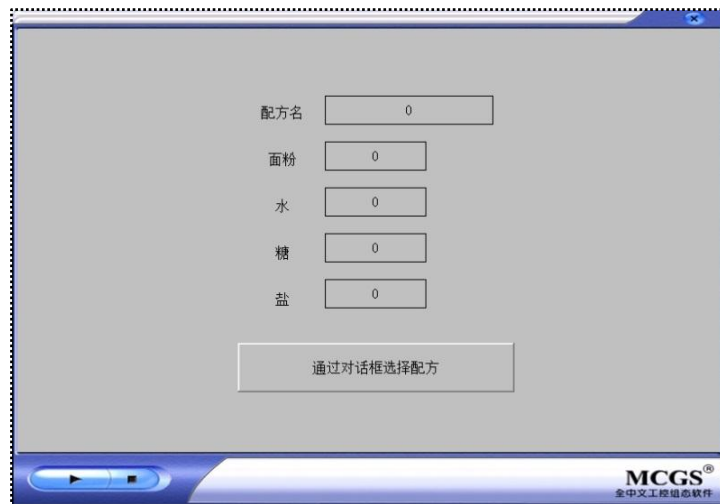


图 3-1-6

点击“通过对话框选择配方”按钮，可以弹出配方选择对话框。在配方组态编辑好的三条配方中进行选择。



图 3-1-7

选择甜面包，点击确认，可以发现变量的数据都变成了选择的甜面包配方中的数据。

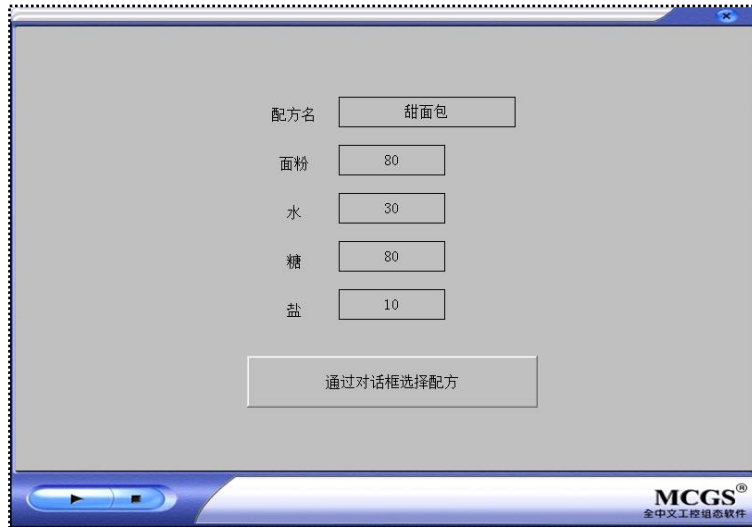


图 3-1-8

第2节 配方修改

使用配方功能时，一般还需要实现在 HMI 中增加、删除、修改配方等操作。本节我们讲解如何在 HMI 实现配方修改的操作。

打开“配方使用”窗口，再增加一个标准按钮构件，名称为“通过对话框编辑配方”，如下图：

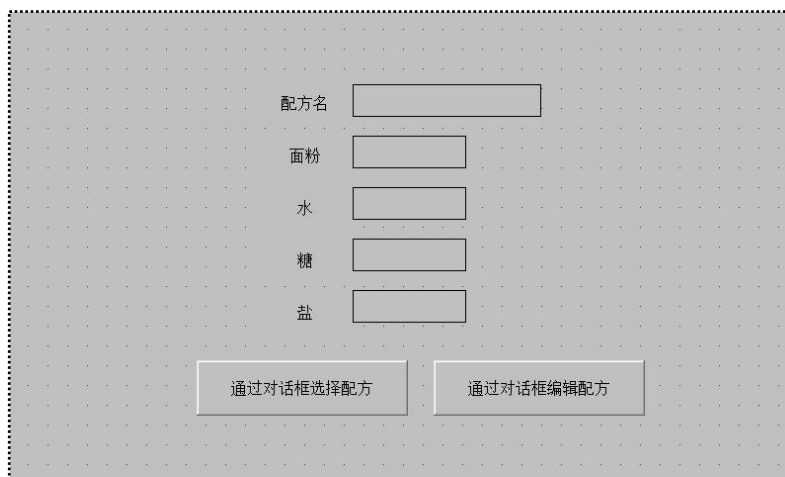


图 3-2-1

双击打开“通过对话框编辑配方”按钮，在脚本程序页输入以下脚本：

```
!RecipeModifyByDialog("面包配方")
```

脚本意义：

✧ 运行时可以打开配方编辑对话框，进行增加、删除、修改、移动、保存

配方等编辑工作

再次模拟运行，选择甜面包配方，可以看到此时的配方数据为：面粉 80、水 30、糖 80、盐 10。

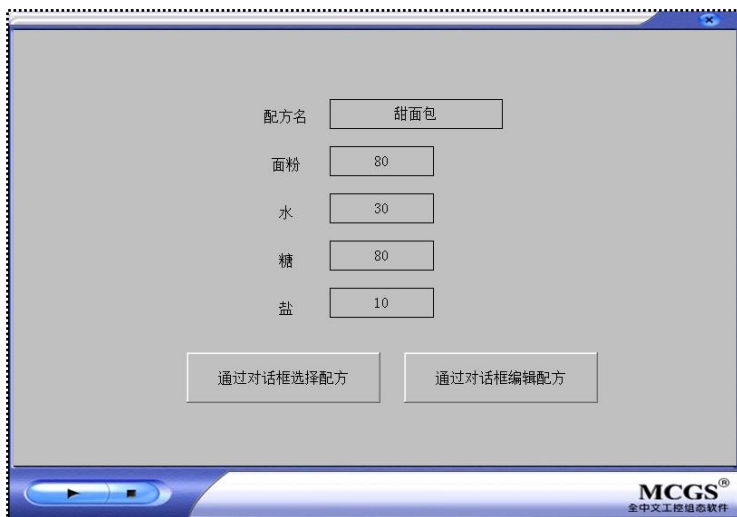


图 3-2-2

点击“通过对话框编辑配方”按钮，打开配方编辑对话框，增加一条“牛角面包”，将甜面包的数据修改为：面粉 80、水 35、糖 60、盐 10。

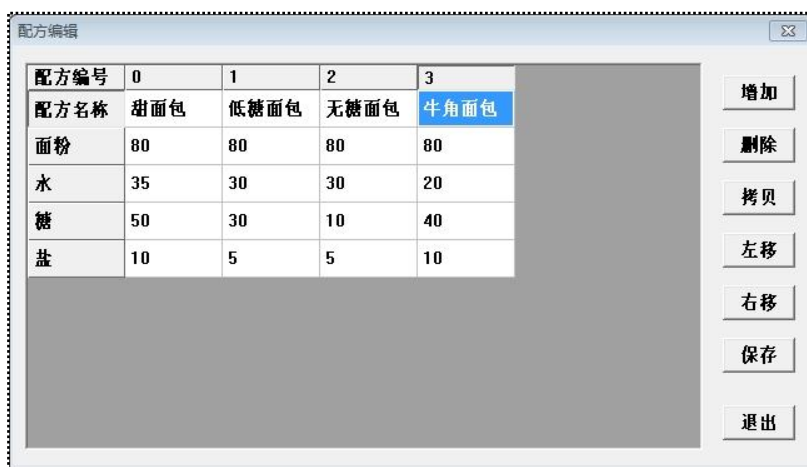


图 3-2-3

此时，再次通过“通过对话框选择配方”按钮选择甜面包配方，可以看到配方数据已经修改为最新状态。



图 3-2-4

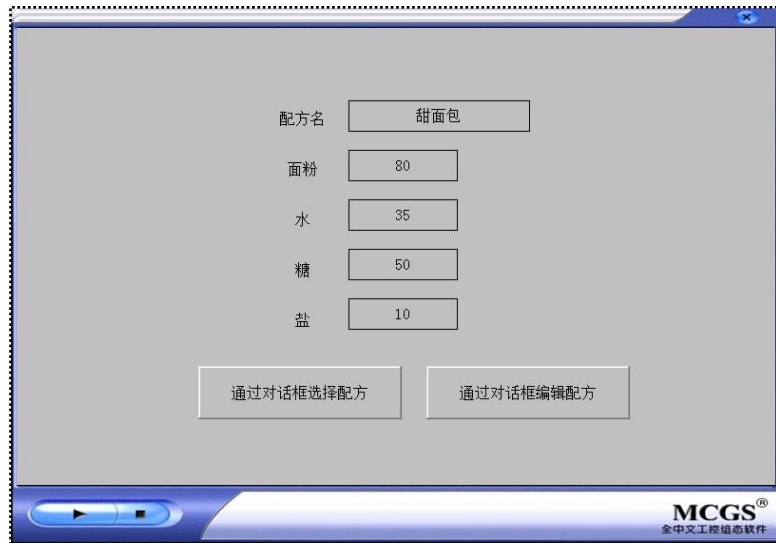


图 3-2-5

如此，便实现了运行时在 HMI 修改配方数据的功能。此外，MCGS 软件还提供了大量的配方操作函数供开发者使用，用户可以在帮助文件中学习函数功能和用法，实现更加灵活多变的配方操作功能。

第四章 策略和脚本

运行策略是指对监控系统运行流程进行控制的方法和条件，它在 MCGS 软件的后台执行，可以灵活的根据既定的事件条件或是时间条件完成操作。可以完成脚本程序、窗口操作、定时器等操作。

脚本程序是组态软件中的一种内置编程语言，非常类似普通的 Basic 语言。可以编制各种流程控制程序和操作。

本章我们将学习运行策略的七个类别中比较常用的启动策略、循环策略、事件策略、用户策略，通过实例讲解运行策略和脚本程序在 MCGS 软件中的应用。

第1节 启动策略

启动策略为系统固有策略，在 MCGS 嵌入版系统开始运行时自动被调用一次。只运行一次，一般完成系统初始化的处理。本节，我们学习使用启动策略和脚本程序实现放大键盘的功能。

新建组态工程，新建用户窗口，添加标签构件和输入框构件如下：



图 4-1-1

- ◇ 标签构件输入文本标题：键盘放大
- ◇ 输入框构件关联数值型中间变量：Data1

模拟运行组态工程，点击输入框构件会在屏幕中央自动弹出数值输入键盘，键盘的大小如下图所示的效果。如果使用的人机界面尺寸比较小，那么键盘触摸输入时会感觉按键比较小，此时就需要使用调整软键盘大小的系统函数来放大键盘。函数只需要开机时执行一次就可以，因而我们将函数放在启动策略的脚本程序中执行。

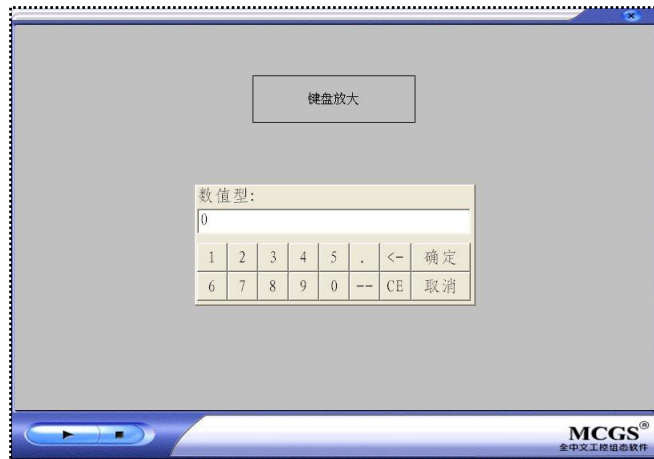



图 4-1-2

点击 **运行策略** 进入运行策略，可以看到 MCGS 软件固有的启动策略，启动策略的名称是不能修改的。双击打开启动策略，点击新增策略行图标 ，给启动策略增加策略行，并添加脚本程序。增加后如下图所示：

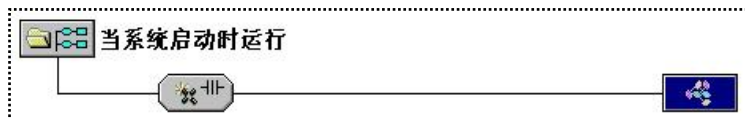


图 4-1-3

双击打开脚本程序，输入以下脚本：

```
!SetNumPanelSize(1, 500)
```

脚本意义：

- ✧ 将数值输入键盘改为 500 像素点大小（键盘放大，长宽比例不变），显示于屏幕中央

模拟运行组态工程，点击输入框弹出数值型软键盘，此时的键盘已经被放大到 500 像素点，对比之前的模拟运行效果图，键盘明显放大了。



图 4-1-4

此系统函数还可以被用来放大字符型键盘、用户登录对话框、配方编辑对话框等，使用方法类似，都是使用于启动策略中。

第2节 循环策略

循环策略为系统固有策略，也可以由用户在组态时创建，在 MCGS 嵌入版系统运行时按照设定的时间循环运行，通常用来完成流程控制任务。我们通过两个实例来学习。

4.2.1 实例 1：定时控制流程

1. 流程概述

启动后开始计时，5 秒后启动定时开关 1，10 秒后启动定时开关 2，并将数值输出变量赋值为 100，流程结束。

2. 变量准备

根据需要，我们首先在实时数据库定义以下变量：

名字	类型	注释
定时开关1	开关型	
定时开关2	开关型	
定时器值	数值型	
数值输出	数值型	

图 4-2-1

3. 制作画面

新建用户窗口，名称修改为“主画面”，添加标准按钮构件，标签构件，文本名称和布局如下：



图 4-2-2

✧ 定时器值对应的标签显示输出“定时器值”，输出值类型：数测量

- ◇ 数值输出对应的标签显示输出“数值输出”，输出值类型：数值量
- ◇ 定时开关 1 对应的指示灯属性页，填充颜色选择“定时开关 1”
- ◇ 定时开关 2 对应的指示灯属性页，填充颜色选择“定时开关 2”

双击“启动”按钮，在脚本程序页输入以下脚本：

!TimerRun(1)

!TimerReset(1,0)

脚本意义：

- ◇ 启动定时器 1
- ◇ 将定时器 1 的值复位为 0

4. 循环策略

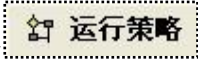

点击  进入运行策略，可以看到 MCGS 软件固有的循环策略。在循环策略上点击右键选择属性项，将定时循环的循环时间修改为 1000 毫秒。表示在 MCGS 嵌入版系统运行时循环策略每 1 秒执行一次。



图 4-2-3

双击打开循环策略，点击新增策略行图标 ，增加策略行，并添加脚本程序。双击打开脚本程序，输入以下脚本：

定时器值=!TimerValue(1,0)

IF !Abs(定时器值-5)<0.5 THEN

定时开关1=1

ELSE

IF !Abs(定时器值-10)<0.5 THEN

定时开关2=1

数值输出=100

!TimerStop(1)

ENDIF

ENDIF

脚本意义：

- ◇ 将定时器 1 的当前值赋值给“数值输出”变量
- ◇ 判断定时器 1 的值为 5 时，执行“定时开关 1=1”
- ◇ 判断定时器 1 的值为 10 时，执行“定时开关 2=1”、“数值输出=100”，并关闭定时器 1
- ◇ 考虑：为何判断条件不用等号，而是用的取绝对值的!Abs 函数呢？

模拟运行组态工程，点击启动按钮后可以看到定时器值每 1 秒会更新一次。定时器值为 5 时，定时开关 1 对应的指示灯变为绿色。定时器值为 10 时，定时开关 2 对应的指示灯变为绿色，数值输出对应值变为 100，且定时器关闭，数值输出的变量值不再更新。



图 4-2-4

4.2.2 实例 2：定点控制

1. 功能概述

每天 14 点自动给温度设定为 30，湿度设定为 25。

2. 变量准备

根据需要，我们首先在实时数据库定义以下变量：

名字	类型	注释
湿度设定	数值型	
温度设定	数值型	

图 4-2-5

3. 制作画面

新建用户窗口，名称修改为“主画面”，添加标签构件，文本名称和布局如下：

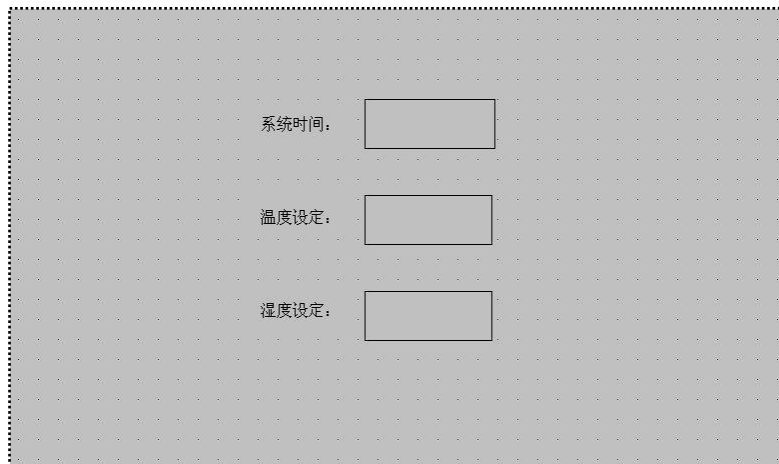


图 4-2-6

- ◇ 温度设定对应的标签显示输出“温度设定”，输出值类型：数值量
- ◇ 湿度设定对应的标签显示输出“湿度设定”，输出值类型：数值量
- ◇ 系统时间对应的标签显示输出，选择内部对象中的“\$Time”，为系统当前时间

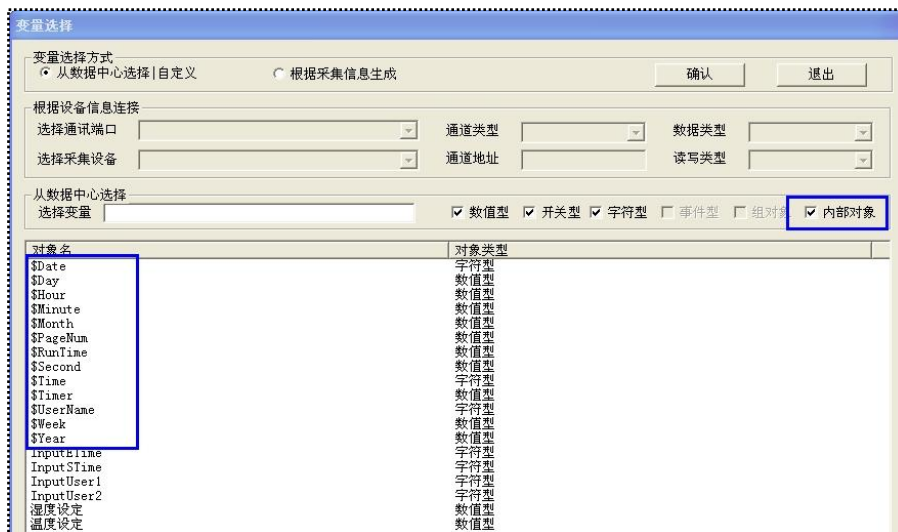
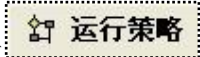
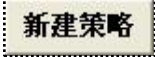


图 4-2-7

3. 循环策略

点击  运行策略 进入运行策略，点击  新建策略，选择策略类型为循环策略。新建一个策略 1，策略类型为循环策略。

名字	类型	注释
 启动策略	启动策略	当系统启动时运行
 退出策略	退出策略	当系统退出前运行
 循环策略	循环策略	按照设定的时间循环运行
 策略1	循环策略	按照设定的时间循环运行

图 4-2-8

在策略 1 上点击右键选择属性项，将循环策略属性修改为“在指定的固定时刻执行”，时间设置为每天的 14 点 0 分。



循环策略属性

策略名称
策略1

策略执行方式


定时循环执行，循环时间(ms): 60000

在指定的固定时刻执行: 每天

1 月 1 日 14 时 0 分 0 秒

策略内容注释
按照设定的时间循环运行

图 4-2-9

双击打开策略 1，点击新增策略行图标 ，增加策略行，并添加脚本程序。双击打开脚本程序，输入以下脚本：

湿度设定=25

温度设定=30

脚本意义：

◇ 每天的 14 点执行“湿度设定=25”、“温度设定=30”

模拟运行组态工程，可以看到当系统时间到达 14 点时湿度设定、温度设定变量的值分别变为了 25、30。

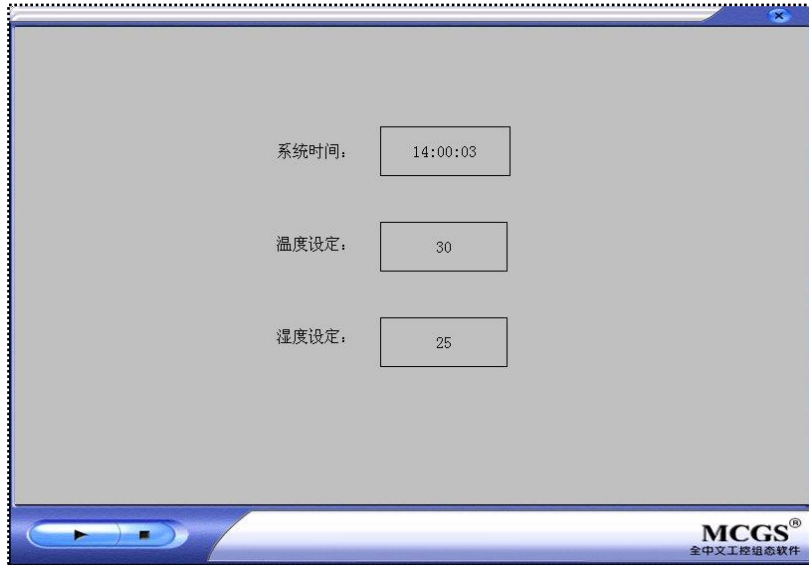


图 4-2-10

通过两个实例我们学习了循环策略的两种控制方式，第一种是按着给定的时间循环执行，第二种是按着指定的时刻执行。循环策略还可以结合不同功能的系统函数以及运算符实现更加灵活的控制和运算。

第3节 事件策略

事件策略由用户在组态时创建，当对应表达式条件成立时，事件策略被系统自动调用一次。本节，我们结合触发存盘功能来学习时间策略的应用。

所谓触发存盘，就是当开关量值有正跳变时存盘一次。

首先进入实时数据库，创建需要的数据对象，如下图所示：

名字	类型	注释
Data1	数值型	
Data2	数值型	
开关	开关型	
数据组	组对象	

图 4-3-1

数据组为组对象，成员包含 Data1、Data2 两个数据对象，存盘属性设置为定时存盘，存盘周期设置为 0 秒。触发存盘需要用到!SaveData()系统函数，此函数只有当组对象的存盘周期为 0 时才有效。



图 4-3-2

新建用户窗口，按着第一章学习的知识创建历史表格，显示数据组对象的存盘数据。如下图所示：



图 4-3-3

双击“开关条件”按钮，在操作属性页选择“数据对象值操作”，操作类型设置为按 1 松 0，数据对象选择“开关”。“按 1 松 0”表示鼠标在构件上按下不放手时，对应数据对象的值为 1，而松开时，对应数据对象的值为 0。

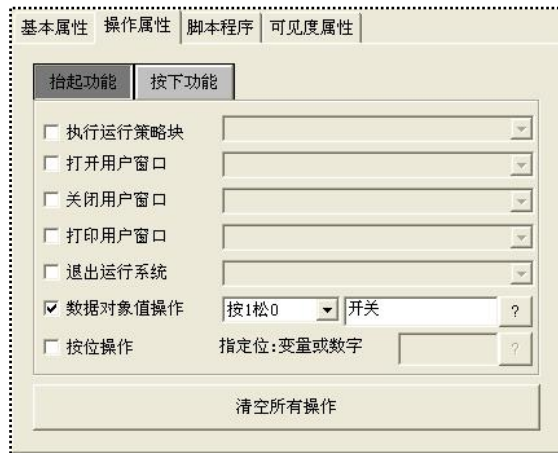
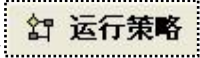



图 4-3-4

单击  运行策略 进入运行策略，单击  新建策略，选择策略类型为事件策略。新建一个策略 1，策略类型为事件策略。





名字	类型	注释
 启动策略	启动策略	当系统启动时运行
 退出策略	退出策略	当系统退出前运行
 循环策略	循环策略	按照设定的时间循环运行
 策略1	事件策略	当确定的条件满足时运行

图 4-3-5

在策略 1 上点击右键选择属性项，关联数据对象选择“开关”，事件内容选择“数据对象的值正跳变时，执行一次”。

事件策略属性

策略名称
策略1

策略执行方式


关联数据对象 ?

事件的内容

确认延时时间 (ms):

策略内容注释
当确定的条件满足时运行

图 4-3-6

双击打开策略 1，单击新增策略行图标 ，增加策略行，并添加脚本程序。双击打开脚本程序，输入以下脚本：

!SaveData(数据组)

脚本意义：

◇ 将数据组对象执行一次存盘

将事件策略的策略属性条件和脚本程序结合，实现了当开关型变量“开关”正跳变一次，就将数据组对象存盘一次，也就是触发存盘的功能。模拟运行组态工程，点击“开关条件”按钮，则“开关”正跳变一次，事件策略被调用一次，执行一次存盘函数。效果如下图：

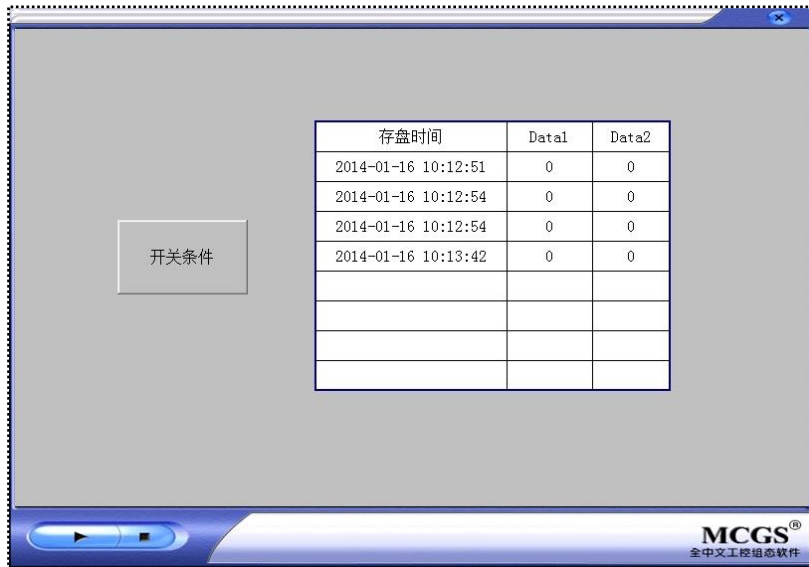


图 4-3-7

第4节 用户策略

用户策略由用户在组态时创建，作为特定的功能模块，被 MCGS 嵌入版系统其它部件调用时执行一次。一般由按钮、菜单、其他策略来调用执行。本节，我们通过开关互锁功能学习一下用户策略的应用。

开关互锁是指两个开关型变量，同时只可能有一个为 1，可以同时为 0，但不能同时为 1。这个功能可以在用户策略实现。

进入实时数据库，创建需要的数据对象：

名字	类型	注释
k1	开关型	
k2	开关型	

图 4-4-1

新建用户窗口，添加标准按钮和指示灯构件。如下图所示：

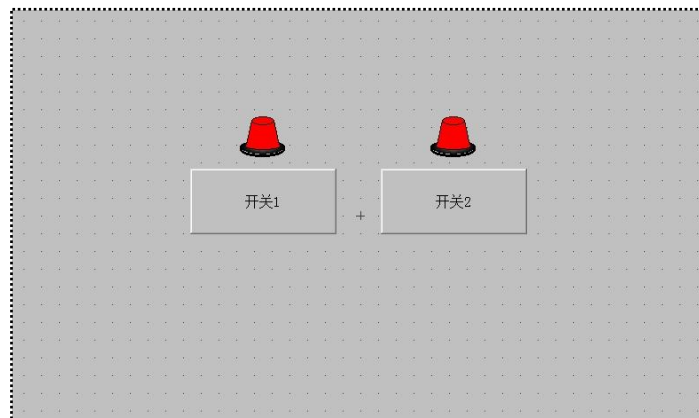
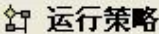
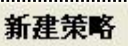


图 4-4-2

◇ 开关 1 对应的指示灯属性页，填充颜色选择“k1”

◇ 开关 2 对应的指示灯属性页，填充颜色选择“k2”

单击  运行策略，单击  新建策略，选择策略类型为用户策略。新建两个策略 1 和策略 2。







名字	类型	注释
 启动策略	启动策略	当系统启动时运行
 退出策略	退出策略	当系统退出前运行
 循环策略	循环策略	按照设定的时间循环运行
 策略1	用户策略	供其他策略、按钮和菜单等使用
 策略2	用户策略	供其他策略、按钮和菜单等使用

图 4-4-3

双击打开策略 1，单击新增策略行图标 ，增加策略行，并添加脚本程序。双击打开脚本程序，输入以下脚本：

```
IF k1=0 AND k2=0 THEN
```

```
k1=1
```

```
ELSE
```

```
k1=0
```

```
ENDIF
```

脚本意义：

◇ 当 k1 和 k2 同时为 0 时，将 k1 置 1，否则 k1 清 0

双击用户窗口的“开关 1”标准按钮，打开按钮属性设置页，在操作属性页选择“执行运行策略块”，策略选择“策略 1”。点击“开关 1”按钮时，策略 1 被调用一次。

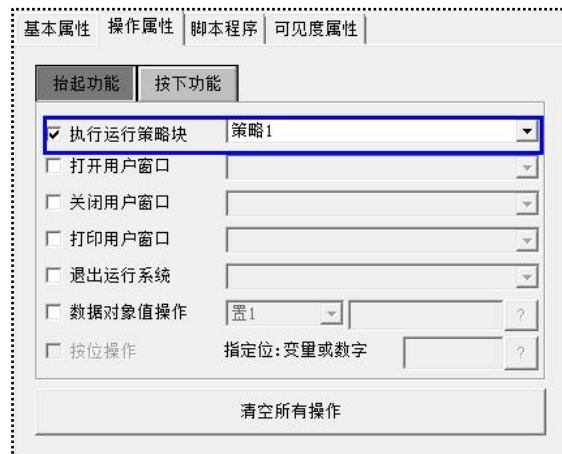


图 4-4-4

同样的方法，在策略 2 中添加策略行和脚本程序，输入以下脚本：

```
IF k1=0 AND k2=0 THEN
```

```
K2=1
```

```
ELSE
```

```
K2=0
```

```
ENDIF
```

脚本意义：

◇ 当 k1 和 k2 同时为 0 时，将 k2 置 1，否则 k2 清 0

同样，将用户窗口的“开关 2”标准按钮选择“执行运行策略块”，策略选择“策略 2”。点击“开关 2”按钮时，策略 2 被调用一次。

模拟运行组态工程，点击“开关 1”后指示灯变为绿色，此时点击“开关 2”按钮指示灯无变化，只有当“开关 1”执行清 0 指示灯为红色后，点击“开关 2”才能将指示灯变为红色，如此实现两个开关的互锁。



图 4-4-5

这个实例演示的是用标准按钮构件调用用户策略的用法。用户策略通常还会使用其他策略通过!SetStgy()函数调用。MCGS 系统允许开发者最多创建 512 个用户策略。

第五章 Modbus 协议应用

Modbus 协议作为电子控制器上的一种通用语，现在已经被越来越广泛的应用于工控行业。MCGS 嵌入版系统支持标准 ModbusASCII、ModbusRTU、ModBusTCP 协议。凡使用标准 Modbus 协议的设备，包括 PLC、变频器、发电机等，MCGS 均可与之建立通讯进行数据传输。

本章，我们对 Modbus 协议做简单概述，学习 MCGS 嵌入版支持的驱动类型。并以 Modbus 数据转发方案为例介绍 Modbus 协议在 MCGS 嵌入版系统中的应用。

第1节 Modbus 协议概述

Modbus 协议是由 Modicon 公司开发出来的一种通讯协议。现在 Modbus 已经是在工业领域被广为应用最流行、最广泛的真正开放、标准的网络通讯协议。此协议支持传统的 RS-232、RS-422、RS-485 和以太网设备。许多工业设备，包括 PLC，DCS，智能仪表等都在使用 Modbus 协议作为他们之间的通讯标准。有了它，不同厂商生产的控制设备可以连成工业网络，将设备集成在一起进行集中监控。

通信时，Modbus 协议决定了每个控制器须要知道它们的设备地址，识别按地址发来的消息，决定要产生何种行动。如果需要回应，控制器将生成应答并使用 Modbus 协议发送给询问方。Modbus 数据查询方式如下：

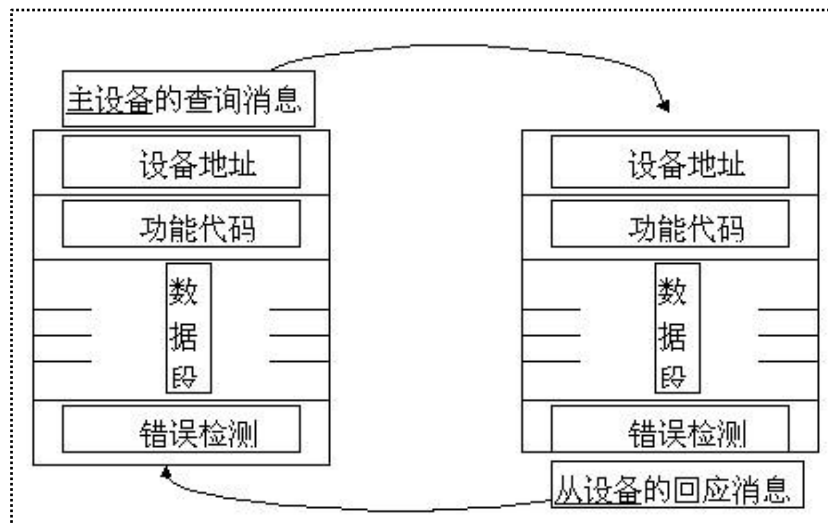


图 5-1-1

第2节 Modbus 驱动

Modbus 协议分为 ASCII、RTU、TCP 等不同的通讯模式，在 MCGS 嵌入版中，提供了 ModbusRTU、ModBusTCP、ModbusASCII 三种类型的驱动。

进入设备窗口，点击设备工具箱的设备管理按钮，打开设备管理对话框。在可选设备中，我们可以看到 MCGS 嵌入版中支持的 Modbus 协议的驱动。Modbus 驱动路径分为两部分，一部分在 PLC 分支下的莫迪康下，如下图所示：



图 5-2-1

第二部分是在通用设备分支下，如下图所示：



图 5-2-2

ModbusRTU: 用于 MCGS 软件读写支持 ModbusRTU 标准协议的各类 PLC、仪表、控制器等设备。支持标准的 RS485 或 RS232 通讯。

ModbusASCII: 用于 MCGS 软件读写支持 ModbusASCII 标准协议的各类 PLC、仪表、控制器等设备。支持标准的 RS485 或 RS232 通讯。

ModbusTCP: 用于 MCGS 软件读写支持 ModbusTCP 标准协议的各类 PLC、

仪表、控制器等设备。支持 RJ45 以太网通讯。

Modbus 串口数据转发设备:用于 MCGS 软件作为 ModbusRTU Slave[从站], 将各寄存器的数据通过 ModbusRTU 协议转发给 ModbusRTU Master [主站]。

ModbusTCPIP 数据转发设备:用于 MCGS 软件作为 ModbusTCP Slave[从站], 将各寄存器的数据通过 ModbusTCP 协议转发给 ModbusTCP Master[主站]。

其中, ModbusRTU、ModbusASCII、ModbusTCP 三个为主站驱动, Modbus 串口数据转发设备和 ModbusTCPIP 数据转发设备是从站驱动。

第3节 Modbus 数据转发

本节, 我们要实现 MCGS 嵌入版组态软件 (应用于 HMI 中) 和 MCGS 通用版软件 (应用于 PC 中) 通过 Modbus 协议进行通讯, 实现数据交互的目的, 也就是所谓的网络方案。

1. 准备工作

- 将 PC 安装 MCGS 通用版, PC 机 IP 地址修改为 200.200.200.191
- 准备一台带以太网口的 HMI, HMI 出厂的默认 IP 为 200.200.200.190
- RJ45 接口的以太网通讯线

2. 制作从站工程

打开 MCGS 嵌入版组态环境, 新建工程, 工程名修改为“从站工程”。

进入设备窗口, 按着第一章学的知识添加模拟设备, 通道 0 和通道 1 连接变量 Data1、Data2。并按着提示添加数据对象 Data1 和 Data2, 对象类型均为数值型。

在设备窗口添加“通用 TCPIP 父设备”和“ModbusTCPIP 数据转发设备”, 如下图所示:

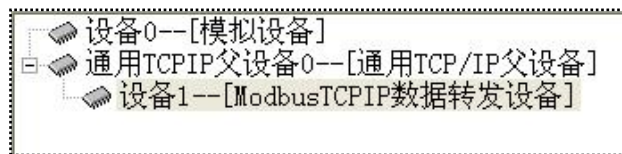


图 5-3-1

双击“通用 TCPIP 父设备”, 打开属性编辑窗口。设置网络类型为“TCP”、服务器/客户设置为“服务器”、本地 IP 地址为“200.200.200.190”、远程 IP 地址为“200.200.200.191”。

设备属性名	设备属性值
设备名称	通用TCP/IP父设备0
设备注释	通用TCP/IP父设备
初始工作状态	1 - 启动
最小采集周期(ms)	1000
网络类型	1 - TCP
服务器/客户设置	1 - 服务器
本地IP地址	200.200.200.190
本地端口号	3000
远程IP地址	200.200.200.191
远程端口号	3000

图 5-3-2

双击“ModbusTCP/IP 数据转发设备”，打开设备编辑窗口。点击 **删除全部通道** 按钮，删除默认的通道。点击 **增加设备通道** 按钮，通道类型选择 4 区输出寄存器、通道地址修改为 1、数据类型选择 16 位无符号二进制、通道个数修改为 2。

添加设备通道

基本属性设置

通道类型 [4区]输出寄存器 数据类型 16位 无符号二进制

通道地址 1 通道个数 2

读写方式 只读 只写 读写

扩展属性设置

扩展属性名 扩展属性值

确认 取消

图 5-3-3

在读写 4WUB0001 对应的连接变量单元格点击右键，在变量选择对话框选择 Data1 数据对象。同样，给读写 4WUB0002 连接变量 Data2。如下图所示：

索引	连接变量	通道名称	通道处理
0000		通讯状态	
0001	Data1	读写4WUB0001	
0002	Data2	读写4WUB0002	

图 5-3-4

新建用户窗口，添加标签构件，显示 Data1、Data2 的当前值。

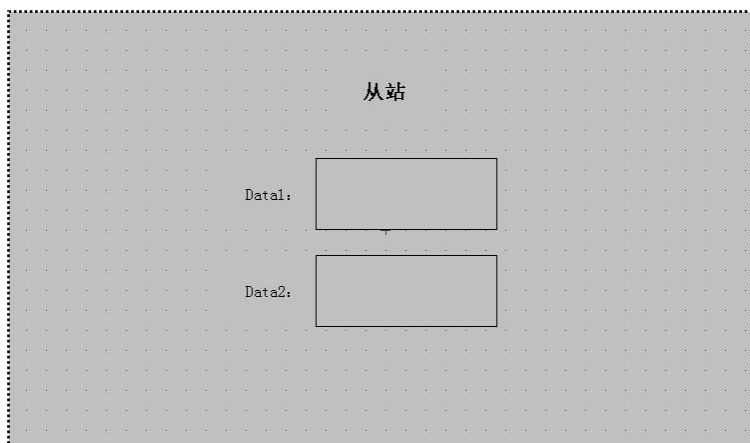


图 5-3-5

至此完成了 Modbus 转发方案中从站工程的组态。运行时，ModbusTCPIP 数据转发设备可以将模拟设备采集过来的 Data1、Data2 两个数据对象转发给 ModbusTCP 主站。

3. 制作主站工程

打开 MCGS 通用版组态环境，新建工程，工程名修改为“主站工程”。

进入设备窗口，添加“通用 TCPIP 父设备”和“标准 ModbusTCP 子设备”。


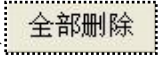
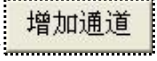


图 5-3-6

双击“通用 TCPIP 父设备”，打开属性编辑窗口。设置网络类型为“TCP”、服务器/客户设置为“客户”、本地 IP 地址为“200.200.200.191”、远程 IP 地址为“200.200.200.190”。

设备属性名	设备属性值
设备名称	通用TCP/IP父设备0
设备注释	通用TCP/IP父设备
初始工作状态	1 - 启动
最小采集周期(ms)	1000
数据采集方式	0 - 同步采集
网络类型	1 - TCP
服务器/客户设置	0 - 客户
本地IP地址	200.200.200.191
本地端口号	3000
远程IP地址	200.200.200.190
远程端口号	3000

图 5-3-7

双击“标准 ModbusTCP 子设备”，打开属性设置窗口。选择基本属性页中的内部属性，点击扩展按钮  打开属性设置对话框。点击  按钮，删除默认的通道。点击  按钮，通道类型选择 4 区输出寄存器、通道地址修改为 1、数据类型选择 16 位无符号二进制、通道个数修改为 2。

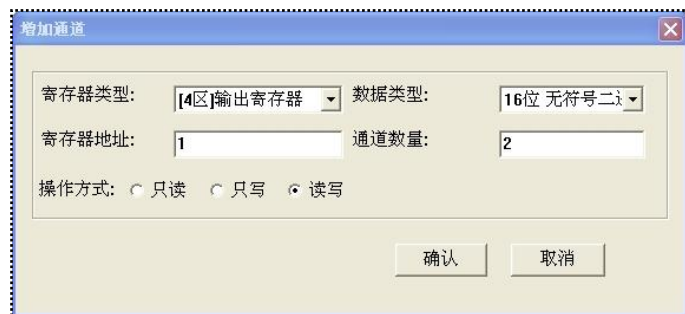


图 5-3-8

增加完通道的效果图如下：



图 5-3-9

在“标准 ModbusTCP 子设备”的属性设置窗口，进入通道连接页，给通讯状态、读写 4WUB0001 和读写 4WUB0002 分别连接数据对象 Data00、Data01、Data02。并按着提示添加数据对象，Data00 为开关型，Data01 和 Data02 为数值型。



图 5-3-10

新建用户窗口，添加标签构件，显示 Data00、Data01、Data02 的当前值。

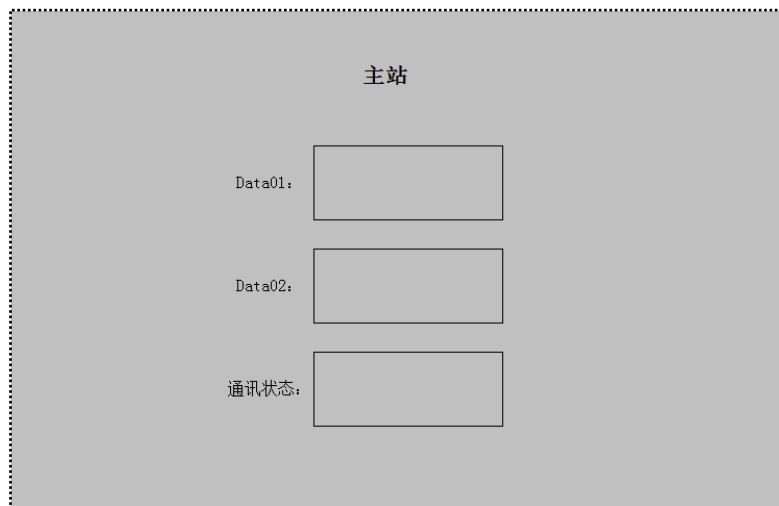


图 5-3-11

4. 实际运行演示

将设置好 IP 地址的 HMI 和 PC 用以太网线连接，将从站工程下载到 HMI 运行，再运行通用版工程，演示 Modbus 数据转发方案的演示效果。