

INOVANCE



IT7000系列HMI

应用案例

资料编码 19011545 A00

# 前言

## 资料简介

InoTouchPad是适用于汇川技术自主研发的IT7000系列HMI组态软件，功能强大，界面友好易用。

本手册介绍了InoTouchPad的部分应用案例。在用户进行创建新项目、构建通信系统、工程管理、添加变量等操作时，本手册为您提供案例参考。本手册适用于InoTouchPad V0.9.0.0及以上版本。

## 更多资料

资料名称	内容简介
InoTouchPad软件帮助	介绍InoTouchPad组态软件安装、简单入门、控件、工程配置和功能应用等详细内容。
IT7000系列HMI应用案例（本文档）	介绍InoTouchPad的工程管理、控件、报警、通信、变量、报表、脚本函数、配方等应用案例，指导用户进行定制化工程组态。
IT7000系列HMI快速入门	介绍产品的快速调试、使用入门等。
IT7000系列HMI产品选型	介绍产品的技术规格、尺寸，以及选配件（安装附件、线缆、等）的详细规格与选型。
IT7150E人机界面用户手册	介绍IT7000系列15寸产品的技术规格、安装尺寸，安装方式以及开机调试方法。
IT7100人机界面用户手册	介绍IT7000系列10.1寸产品的技术规格、安装尺寸，安装方式以及开机调试方法。
IT7070人机界面用户手册	介绍IT7000系列7寸产品的技术规格、安装尺寸，安装方式以及开机调试方法。

## 适用对象

本手册面向使用InoTouchPad对IT7000系列产品进行通讯、组态、调试、安装和服务的相关操作人员和组态工程师。

## 版本变更记录

修订日期	发布版本	变更内容
2021-8	A00	手册第一次发布。

## 关于手册获取

本手册不随产品发货，如需获取电子版PDF文件，可以通过以下方式获取：

登录汇川技术官方网站（[www.inovance.com](http://www.inovance.com)），“服务与支持-资料下载”，搜索关键字并下载。

# 目录

前言 .....	1
1 控件 .....	4
1.1 使用字按钮控件实现页面切换 .....	4
1.2 控件调用图库并实现背景透明显示 .....	6
1.3 通过图表控件查看数据变化 .....	8
1.4 制作复归、置位、互锁按钮 .....	12
1.5 制作下载进度条 .....	14
1.6 使用文本列表和下拉框实现模式切换 .....	19
1.7 使用图形列表实现数据控制图形切换 .....	22
1.8 将数据和趋势视图关联显示 .....	25
1.9 应用GIF及表格 .....	27
1.10 应用XY曲线 .....	33
1.11 设置按钮的可见性、禁用按钮 .....	36
2 画面 .....	40
2.1 应用模板画面和图层 .....	40
2.2 制作弹出式菜单 .....	42
2.3 使用密码功能控制画面跳转 .....	45
2.4 使用定时器实现画面延时后自动跳转 .....	47
2.5 高亮显示当前画面标签 .....	49
2.6 在点击按钮时弹出窗口提示 .....	53
2.7 使用画中画实现TAB窗口 .....	56
3 报警 .....	60
3.1 报警控件的使用方法 .....	60
3.2 制作报警弹出窗口 .....	62
3.3 实现实时报警和历史报警 .....	65
4 通信 .....	71
4.1 配置通讯参数 .....	71
4.2 启动、停止通讯从站 .....	72
4.3 在线修改通讯参数 .....	73
4.4 自由协议的使用 .....	75
4.5 分期付款功能 .....	80
4.6 通过HMI下载PLC程序—穿透功能 .....	90
4.7 VNC功能应用 .....	90
4.8 平台PC组态软件应用 .....	95
5 变量 .....	98
5.1 变量的应用 .....	98
5.2 数组下标功能的应用 .....	100
5.3 可见性的应用 .....	103
5.4 在变量变化时自动弹出窗口 .....	105

---

5.5	索引变量功能的应用 .....	107
5.6	通过变量更改棒图、滚动条、进度条、流动块的状态 .....	109
5.7	使用PLC变量切换画面 .....	112
6	批量操作 .....	114
6.1	字体批量切换.....	114
7	工程设置 .....	117
7.1	设置屏保及背光灯.....	117
7.2	设置起始画面以及起始语言.....	119
7.3	工程备份.....	120
7.4	使用U盘或SD卡升级HMI程序.....	121
7.5	密码功能应用.....	124
7.6	自动翻译.....	130
7.7	多语言字体独立设置 .....	135
7.8	交叉引用功能的应用 .....	136
7.9	如何查看HMI固件版本和升级HMI固件 .....	138
7.10	使用U盘或SD卡升级PLC程序 .....	140
7.11	通过多语言功能切换工程中英文 .....	142
8	权限管理 .....	147
8.1	用户登录权限管理.....	147
8.2	通过用户权限登录控制按钮.....	152
9	配方 .....	157
9.1	配方系统函数的应用 .....	157
9.2	使用U盘或SD卡导入导出数据 .....	161
10	历史记录.....	163
10.1	如何查看HMI各类控件的操作记录 .....	163
11	报表 .....	165
11.1	利用报表实现数据统计 .....	165
12	脚本 .....	171
12.1	利用脚本生成二维码和条形码.....	171
12.2	利用脚本实现指定时间弹窗 .....	175
12.3	利用调度器实现画面切换以及运行脚本.....	179
12.4	脚本获取系统时分秒.....	182
12.5	在进入画面或者离开画面时调用一次脚本 .....	187
13	函数 .....	190
13.1	脚本函数介绍（浏览、概念） .....	190
13.2	编辑控件样式代码 .....	193

# 1 控件

## 1.1 使用字按钮控件实现页面切换

### 案例要求

通过HMI完成利用字按钮控件对页面的切换：

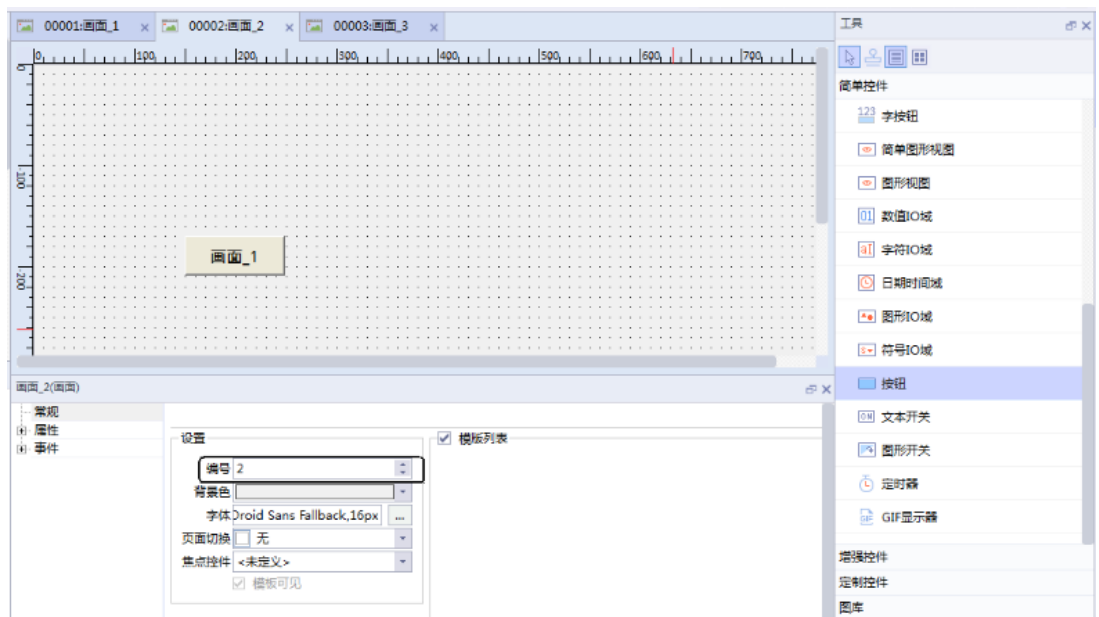
1. 掌握页面切换的方式（利用字按钮进行切换）。
2. 掌握模板画面的使用方法。

### 案例目的

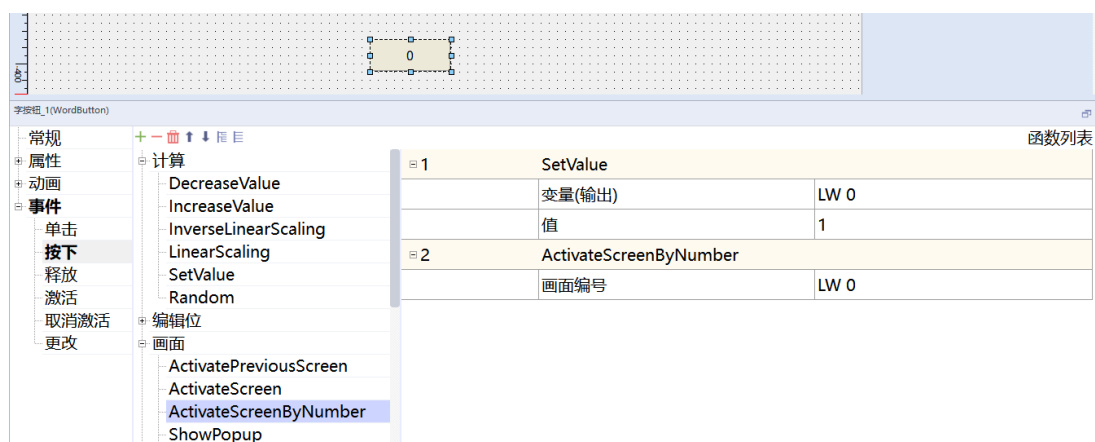
- 通过此案例学会页面切换方法。
- 通过此案例学会模板画面的使用。

### 编程步骤

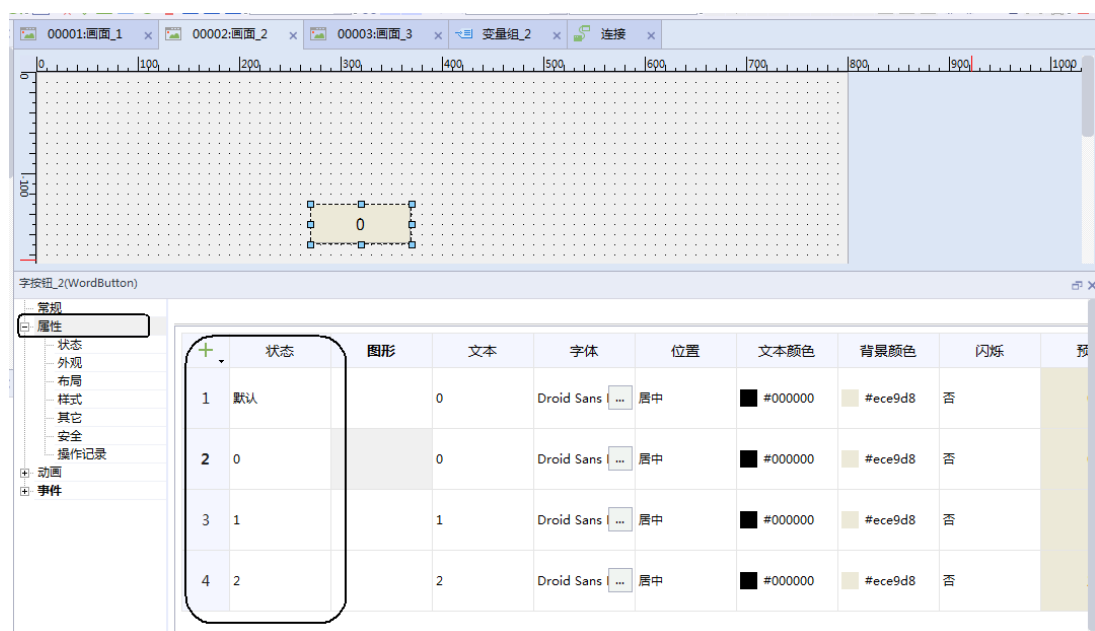
- 使用字按钮切换页面
  1. 点击“事件”>“画面”使用系统函数“ActivateScreenByNumber”，当页面编号的变量值与画面编号一致时，便可以将画面切换到该画面。



2. 使用字按钮组态系统函数，点击“计算”>“SetValue”先给页面编号变量赋值，之后跳转页面到页面编号变量的画面上。

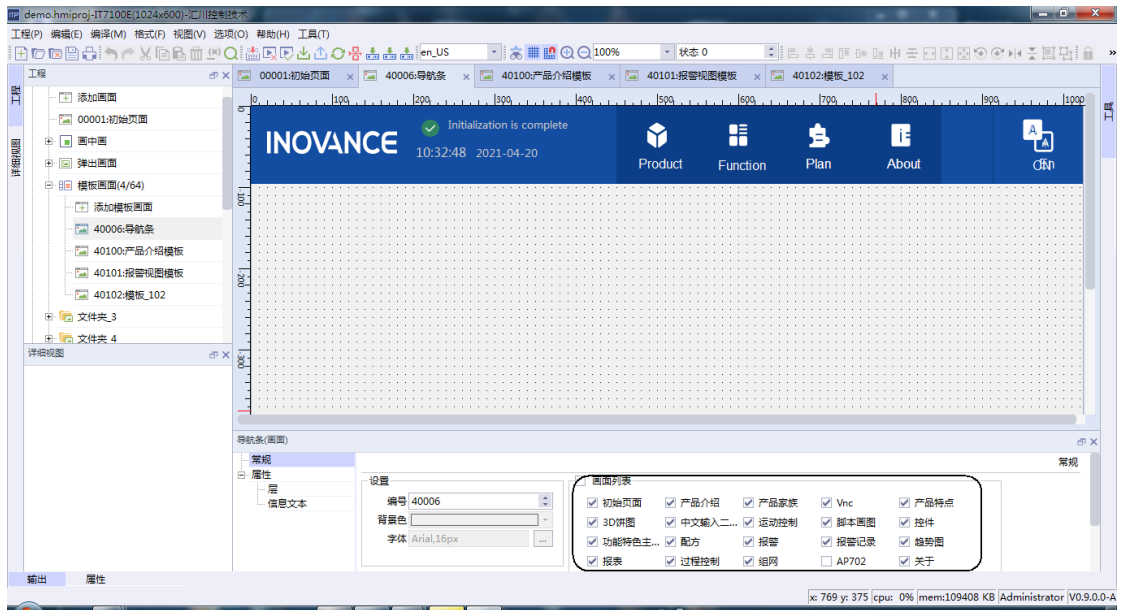


3. 为区分不同页面的状态，可以设置字按钮在不同状态下的图形文字等。点击“事件”其中“状态”值为画面编号，对应页面下的状态可以根据需求设置。

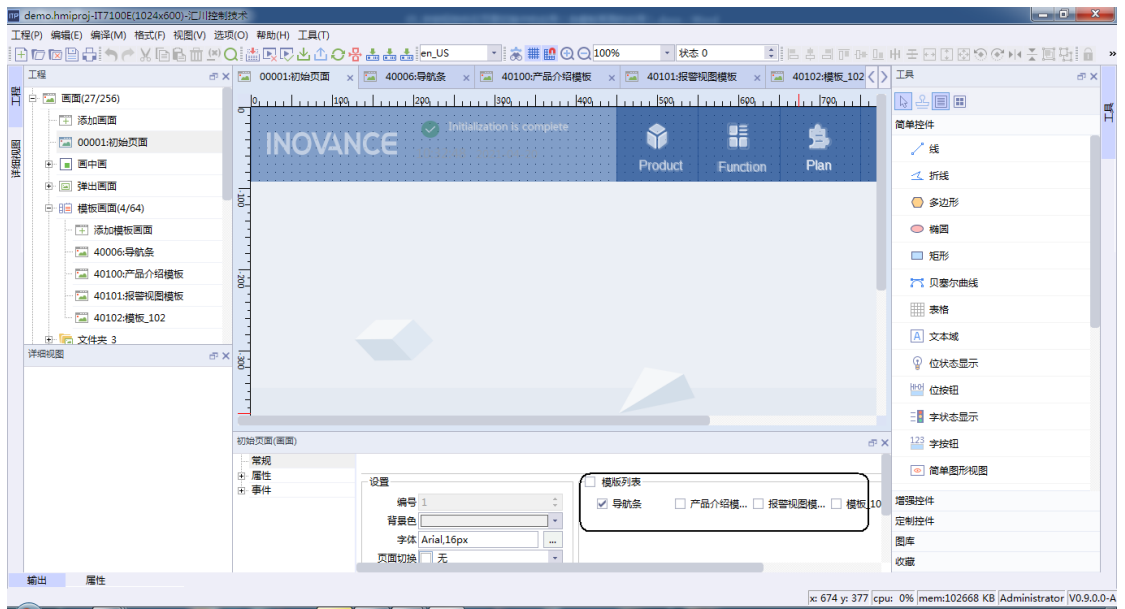


- 模板画面的使用方法

点击“模板画面” > “添加模板画面”，在“画面列表”勾选需要使用的模板的画面即可。



其中，模板画面和普通画面的区在画面列表的差别，如下图所示,表现在“模板列表”的选择区别上。



## 1.2 控件调用图库并实现背景透明显示

InoTouchpad自带多种类型图库，包括背景图、按钮、开关等。图库可从软件安装目录的以下路径找到：  
InoTouchPad>Graphics>Inovance。

### 案例要求

实现按用户偏好改变控件外观，或根据需要隐藏按钮控件。

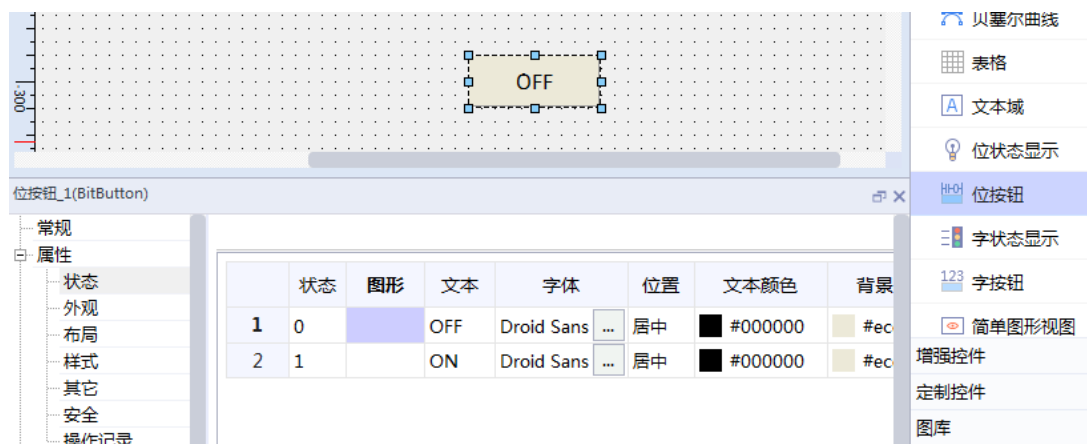
### 案例目的

通过调用软件自带的图库来更改控件样式，或根据需要隐藏某些按钮控件。

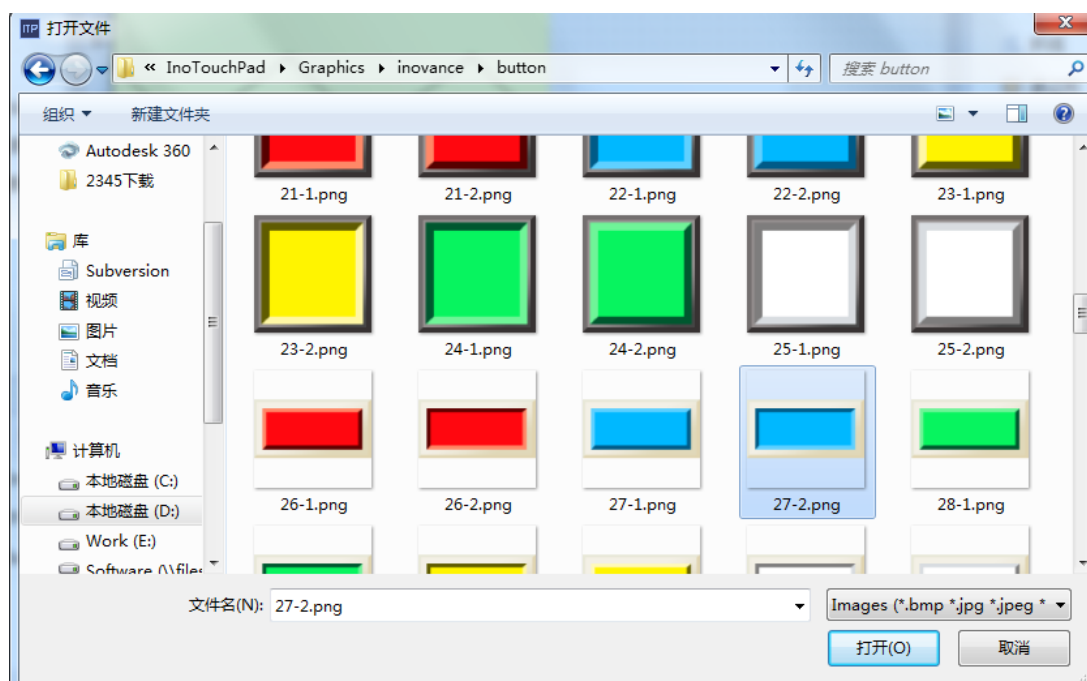
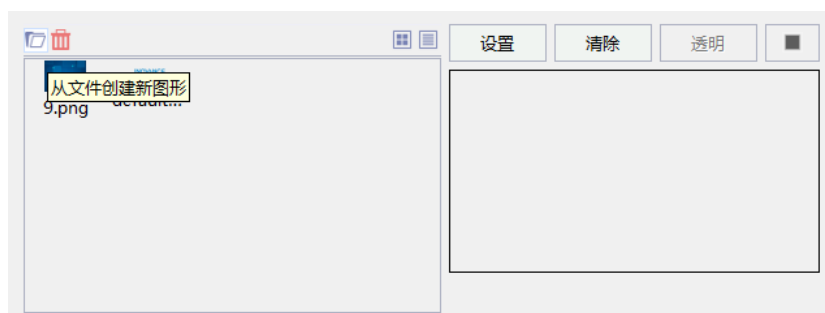
## 编程步骤

- 调用图库

- 新建一个工程，并在“工具”选项卡中选择一个控件（以位按钮为例），在按钮控件的“属性”中选择“状态”。



- 双击“图形”区域打开图库，左上角小文件夹标识即可调用图库，如下图所示：



- 控件透明显示

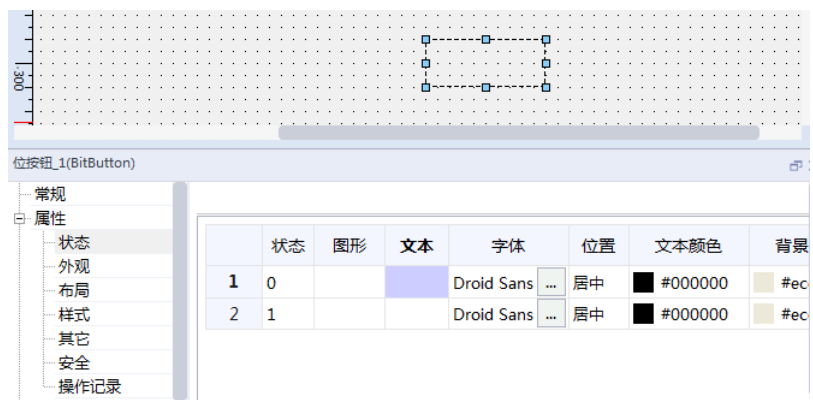
- 以位按钮控件为例，在按钮的“属性”中选择“外观”。



2. 将“填充样式”设置为“透明”，此时，按钮的背景已经呈透明显示。



3. 设置透明背景后，回到“状态”属性中将文本删除，按钮即可显示为全透明。



## 1.3 通过图表控件查看数据变化

### 案例要求

1. 通过各类旋钮、量表、仪表、3D饼图直观查看数据变化。
2. 通过3D饼图显示多个过程变量的比值。

### 案例目的

- 掌握旋钮、量表、仪表、3D饼图基本使用方法。
- 掌握通过旋钮设置过程变量值。

### 编程步骤

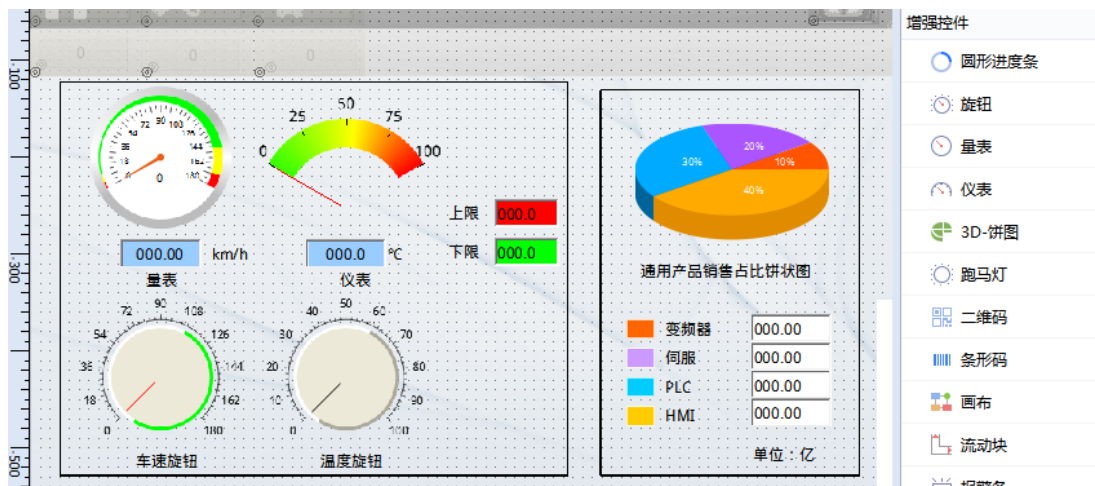
#### 1. 添加变量

打开左侧项目树“工程” > “通讯” > “变量”，双击“添加变量组”添加一个变量组，新建所需变量，如下图所示。

+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式
1	旋钮1	6	<内部变量>	Int16	2	1	LW 0	100ms	循环使用
2	旋钮2	7	<内部变量>	Float	4	1	LW 10	100ms	循环使用
3	3D-变频器	8	<内部变量>	UInt16	2	1	LW 2	100ms	循环使用
4	3D-伺服	9	<内部变量>	UInt16	2	1	LW 3	100ms	循环使用
5	3D-PLC	10	<内部变量>	UInt16	2	1	LW 4	100ms	循环使用
6	3D-HMI	11	<内部变量>	UInt16	2	1	LW 5	100ms	循环使用
7	仪表上限	12	<内部变量>	Float	4	1	LW 12	100ms	循环使用
8	仪表下限	13	<内部变量>	Float	4	1	LW 14	100ms	循环使用

## 2. 添加基本控件

a. 通过从右侧工具栏添加旋钮、量表、仪表、数值IO域等控件。



b. 旋钮、量表属性视图“常规”>“过程”>“变量”添加变量“旋钮1”，旋钮属性视图“常规”>“刻度”设定最大值为180。



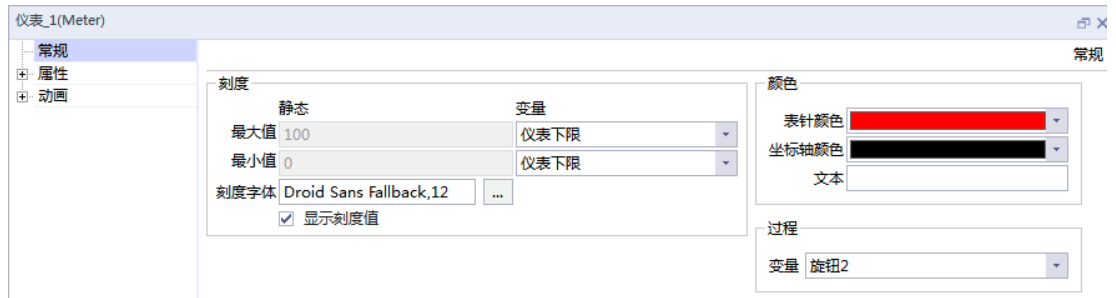
量表属性视图“属性”>“刻度”>“范围”设定最大值为180，上下限根据需求设定。



c. 旋钮、仪表属性视图“常规”>“过程”>“变量”添加变量“旋钮2”，旋钮属性视图“常规”>“刻度”设定最大值为100。



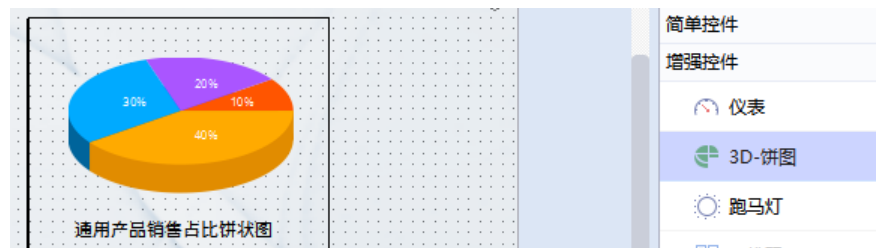
仪表属性视图“属性”>“刻度”>“变量”添加仪表上下限变量。



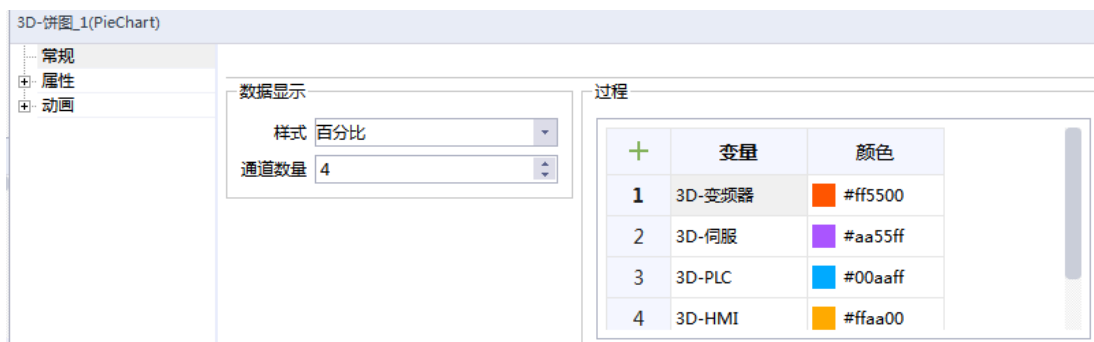
添加两个数值IO域控件，并关联过程变量为仪表上下限对应变量。



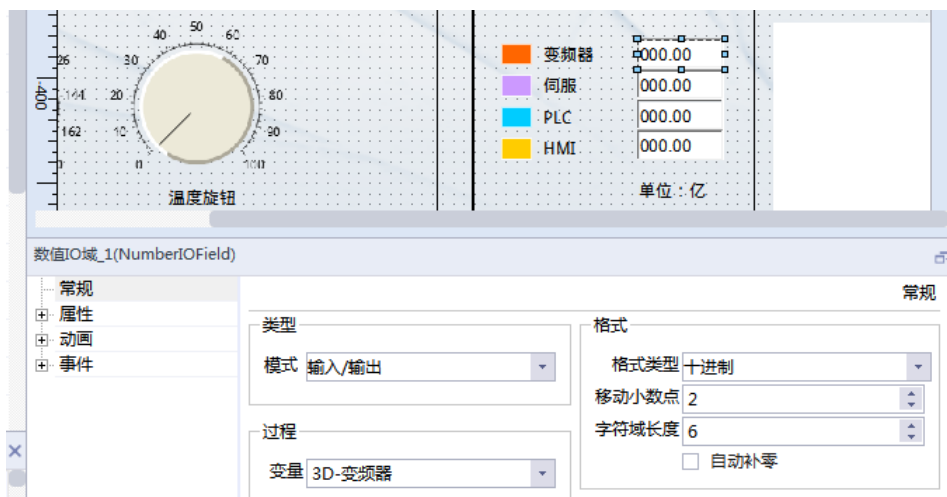
d. 3D饼图：右侧工具栏“工具”>“增强控件”>“3D饼图”鼠标左键选择控件后拖拽到画面编辑区。



3D饼图属性视图“常规”>“数据显示”调整样式和添加通道数量。关联各通道过程变量，同时调整通道颜色配置。

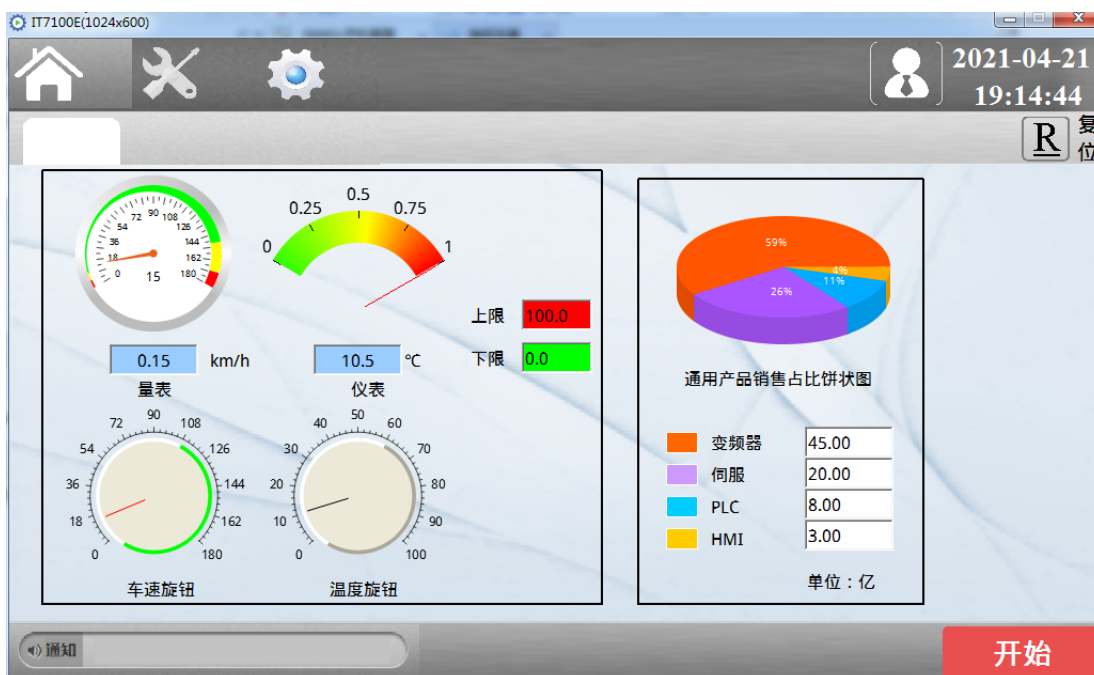


添加数值IO域控件，与3D饼图各通道变量设定一致。



3. 通过变量更改棒图、滚动条、进度条、流动块的状态

- a. 调整“车速按钮”对应“量表”会直观显示设定速度。
- b. 调整“温度旋钮”对应“仪表”会直观显示设定温度。
- c. 设置3D饼图对应的变量值，可适时调整3D饼图状态。



## 1.4 制作复归、置位、互锁按钮

### 案例要求

制作按钮，分别实现复归、置位及互锁功能。

### 案例目的

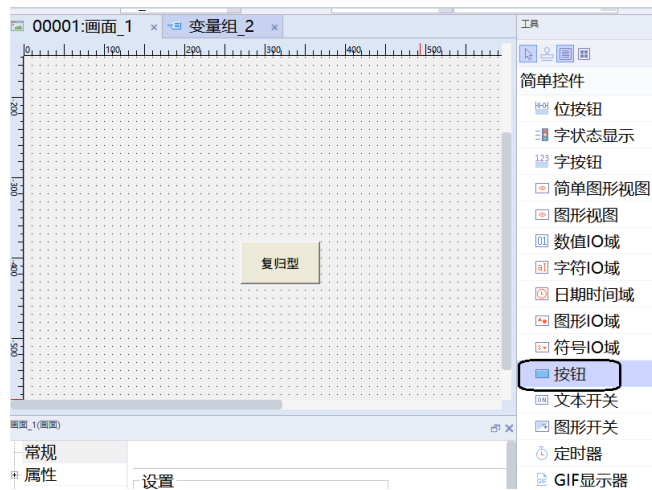
通过此案例了解如何组态变量事件。

### 编程步骤

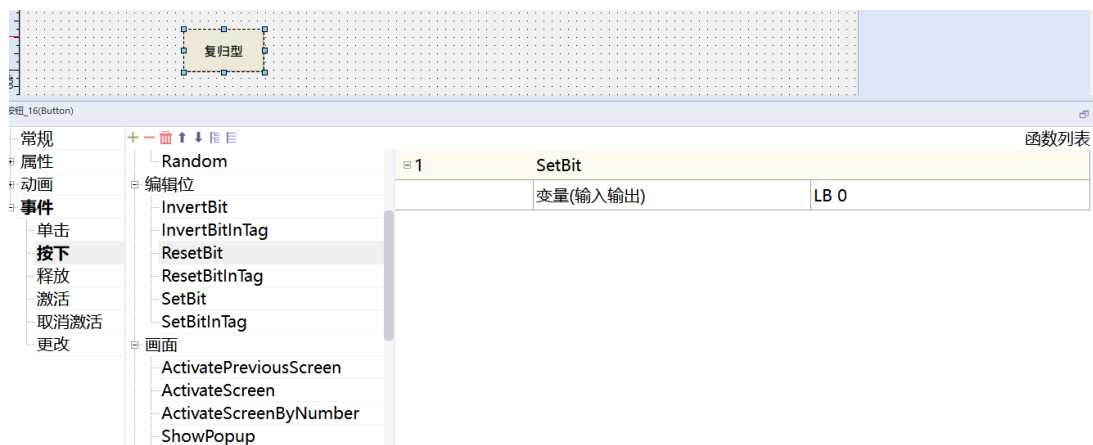
- 添加变量  
添加两个布尔型变量，用于测试功能。



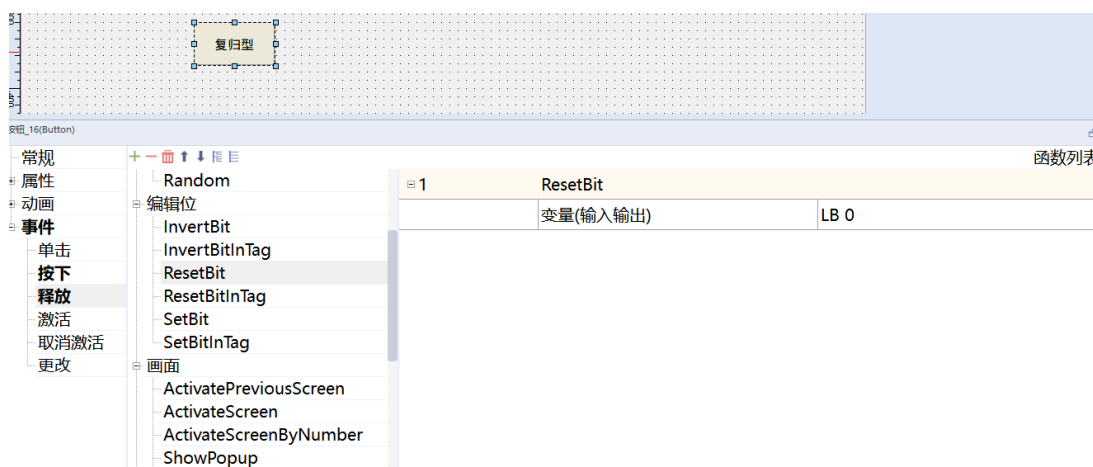
- 实现复归型按钮  
1. 添加一个按钮。



2. 在按钮的“事件” > “按下”组态SetBit函数，变量设置为LB 0。



3. 在按钮的“事件” > “释放”组态ResetBit函数，变量设置为LB 0。



4. 通过以上设置，按钮可实现复归功能，即按下时置位释放时复位。

- 实现置位型按钮

1. 添加一个按钮。



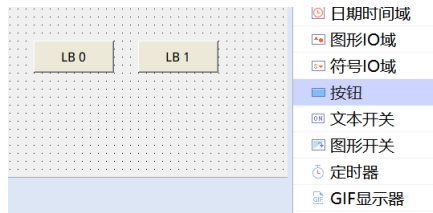
2. 在按钮的“事件” > “单击”组态SetBit函数，变量设置为LB 0。



3. 这样当按下按钮时，LB 0即被置位。

● 实现互锁型按钮

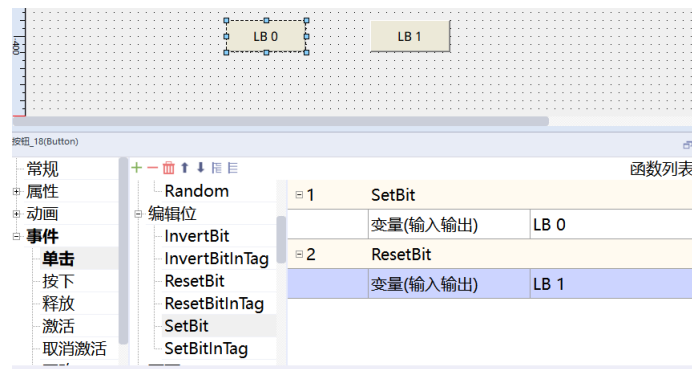
1. 互锁功能类似Plc的互锁，即当一个变量置位时，另一个变量复位。本例使用LB 0/LB 1两个变量进行组态。首先需要添加两个按钮到画面中。



2. 分别在两个按钮的”事件”>”单击”组态SetBit函数，变量分别设置为LB 0及LB 1。



3. 分别在两个按钮的”事件”>”单击”组态ResetBit函数，变量分别设置为LB 1及LB 0，即置位LB 0的按钮需要同时复位LB 1，反之亦然。



4. 完成以上步骤之后，即可实现互锁功能。

## 1.5 制作下载进度条

### 案例要求

通过HMI自带的控件制作下载进度条。

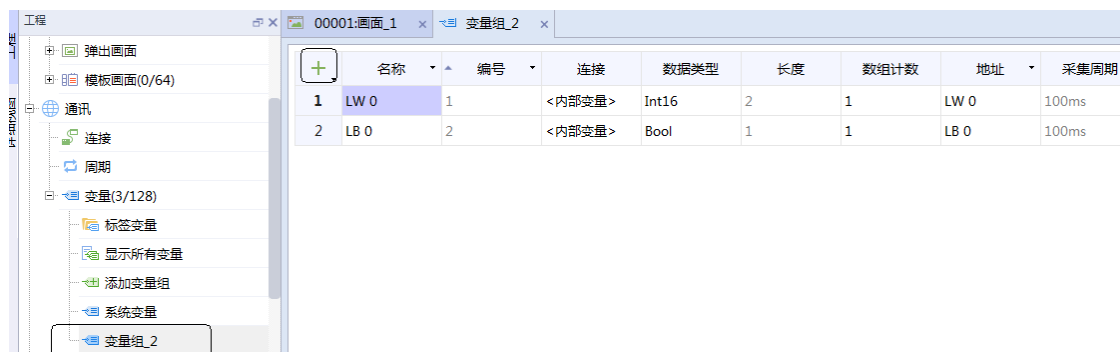
### 案例目的

- 通过此案例了解如何使用棒图及定时器制作下载进度条。
- 通过此案例了解定时器的使用。

### 编程步骤

1. 添加变量

添加两个变量，一个为Int16类型变量，用于显示进度条百分比。一个为Bool类型变量，用于激活定时器。



## 2. 设置变量参数

### a. 设置Int16类型变量的上下限

点击变量，在下方窗口选择“属性”>“限制”，上限设置为100，下限设置为0。

—	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式
1	LW 0	1	<内部变量>	Int16	2	1	LW 0	100ms	循环使用
2	LB 0	2	<内部变量>	Bool	1	1	LB 0	100ms	循环使用



### b. 设置Bool类型变量的初始值

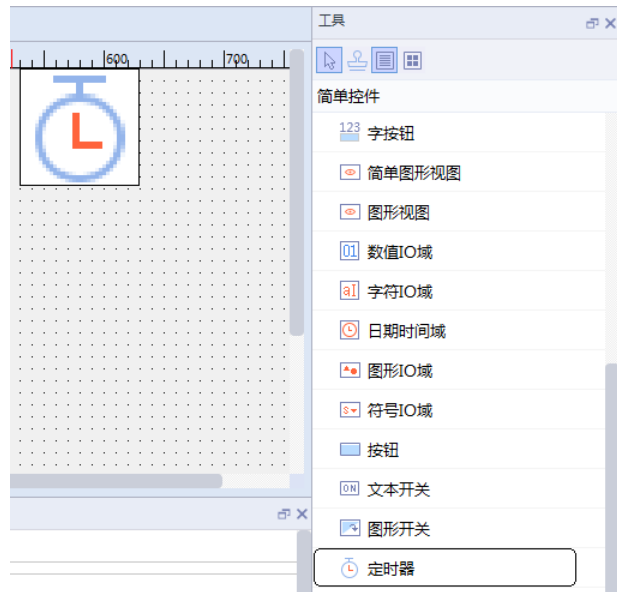
将Bool类型变量的“初始值”设置为1，此操作是为了触发定时器。

—	名称	编号	连接	数据类型	长度	数组计数	地址	记录采集模式	初始值	索引变量
1	LW 0	1	<内部变量>	Int16	2	1	LW 0	循环连续		<未定义>
2	LB 0	2	<内部变量>	Bool	1	1	LB 0	循环连续	1	<未定义>

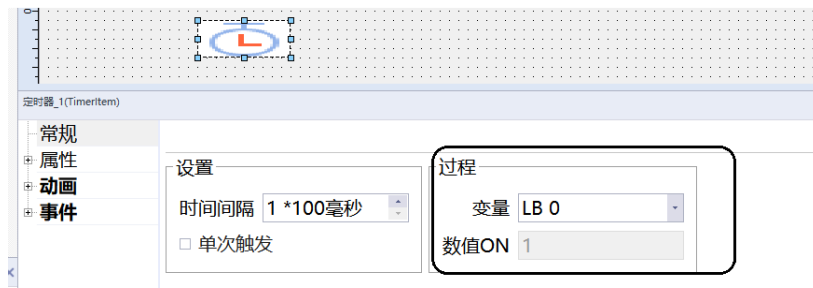
## 3. 组态定时器

### a. 在“工具”中找到定时器控件，拖拽到画面中。





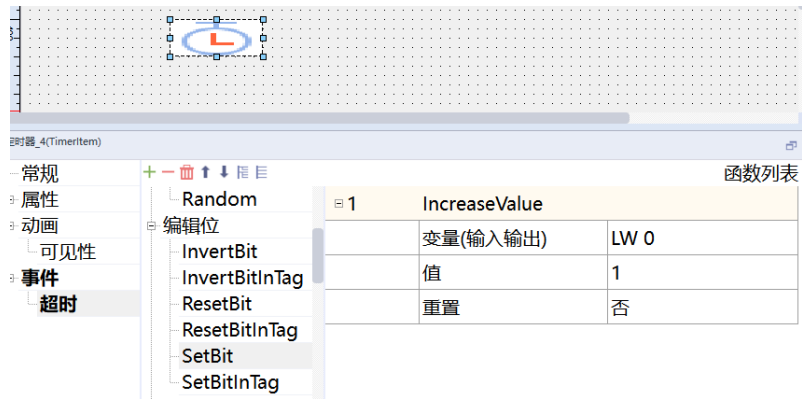
- b. 点击添加的定时器，在下方窗口设置定时器的“过程”变量为LB 0，由于我们将LB 0的初始值设置为1，故定时器会直接启动。



- c. 在“动画” > “可见性”中启用可见性功能，设置“变量”为LB 0，“对象状态”设置隐藏，“类型”选择位。此操作可以在程序运行时隐藏定时器，使画面美观。

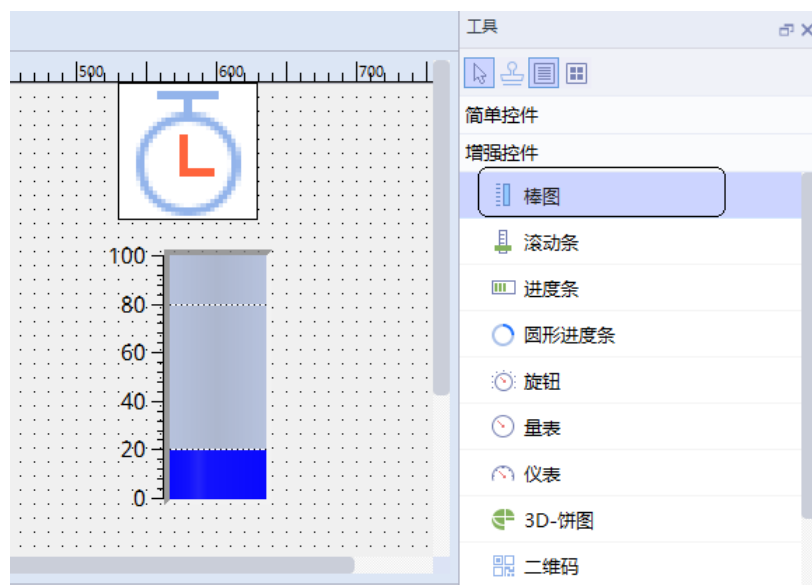


- d. “事件” > “超时”中设置IncreaseValue函数，“变量”组态用于控制进度条的LW 0，值设置为1。即定时器每完成一次计时，LW 0加1。

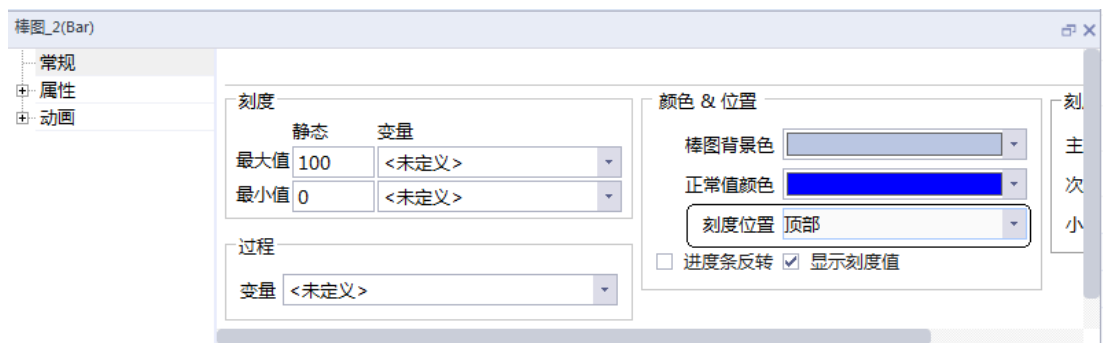


#### 4. 组态棒图

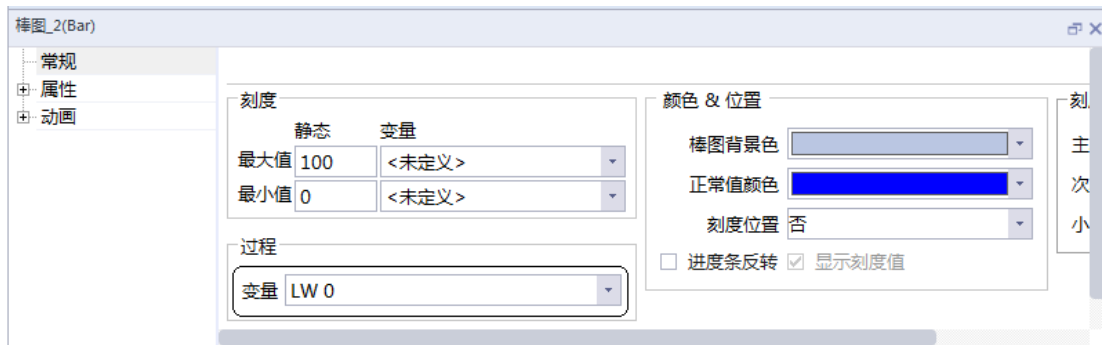
a. 在“工具”中找到棒图控件，拖拽到画面中。



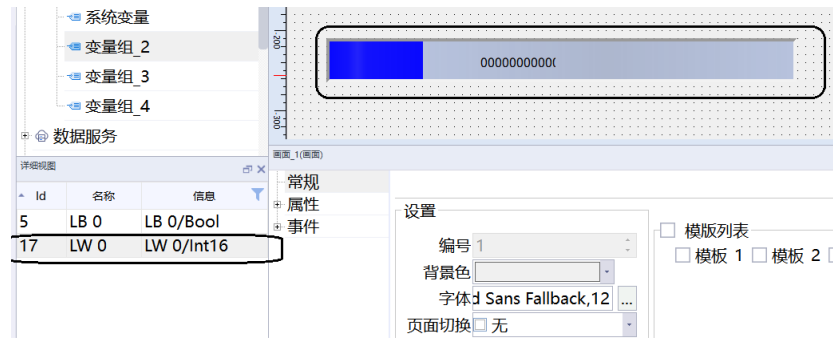
b. 点击棒图，在下方的窗口中找到“常规” > “刻度位置”，先选择为顶部，再切换为否，这样做的目的是旋转棒图及隐藏刻度，使画面美观。



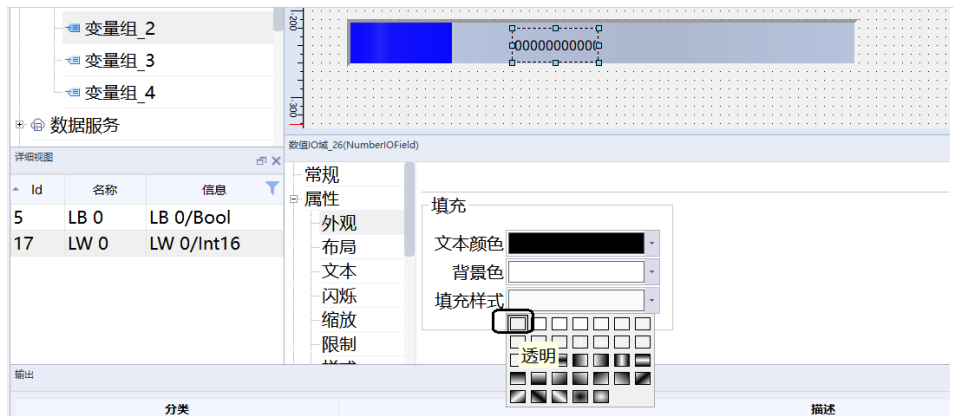
c. 在“常规” > “过程” > “变量”中组态LW 0。



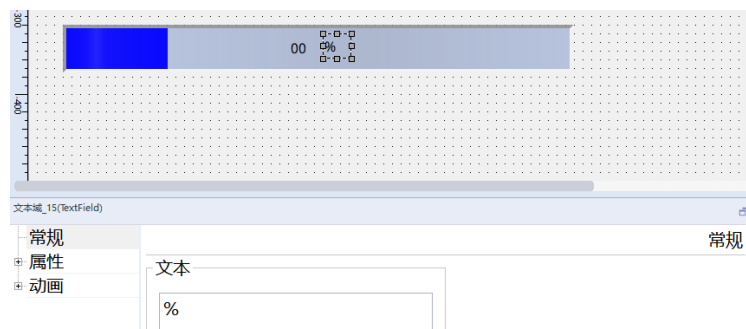
d. 添加一个IO域用于显示数值，直接从变量组中拖拽到画面即可。



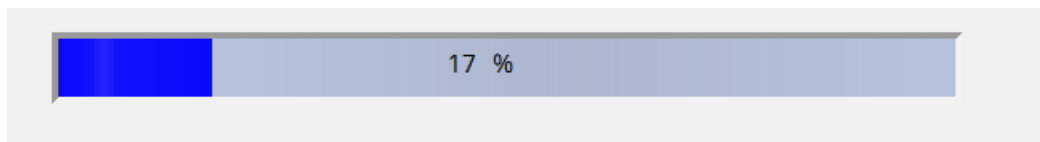
e. 在“属性” > “外观” > “填充样式” 中将IO域背景设置为透明，使画面美观。



f. 添加文本域显示%。



g. 运行效果如下图所示。



## 1.6 使用文本列表和下拉框实现模式切换

### 案例要求

1. 通过HMI 软件自带的文本列表，配合符号IO域控件实现模式或状态切换功能制作一个文本列表。
2. 将符号IO域关联文本列表与PLC中的过程变量，实现简易的模式切换。

### 案例目的

通过此案例了解文本列表的功能，学会使用文本列表工具绘图工具。

### 编程步骤

绘制文本列表：

1. 绘制文本列表内容：左侧项目树“状态列表” > “文本列表” 新建一个文本列表，命名为“单步调试”，如下图所示。

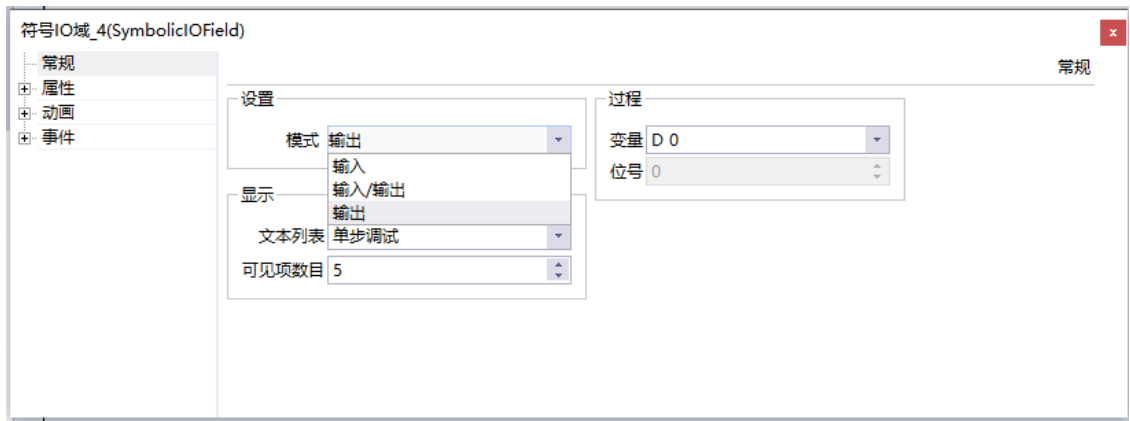


表格表头“选择”意为选择所要关联变量的类型，其中位（0，1）表示变量为BOOL型，可选状态只有0和1；位号（031）表示所选变量为32位整型，用来设置整型变量的某一位；范围（……）表示当所选变量在某一范围时，设置即可生效。此案例我们进行一个多轴运行的单步调试功能，利用文本列表工具进行单步调试，显示当前所处步序。

2. 编辑列表条目：单击选中的文本列表即可选择该列表条目的编辑，如下图所示。

+	编号	数值	条目
1	15	0 - 9	机械回零
2	16	10 - 19	主轴启动
3	17	20 - 29	X轴定位至100
4	18	30 - 39	Y轴以200速度运行
5	19	40 - 49	Z轴下移至目标位开始钻孔

3. 在“工具栏” > “简单控件” > “符号IO域” 将控件拖动至页面进行组态设置，设置页面如下图所示。

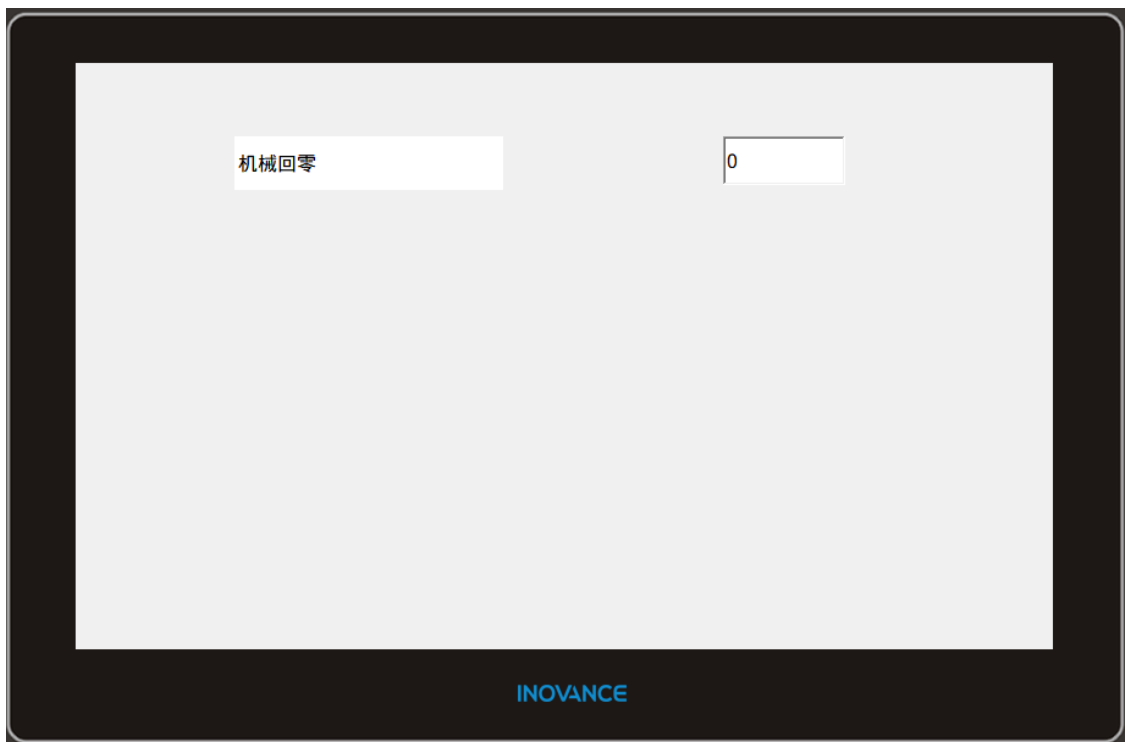


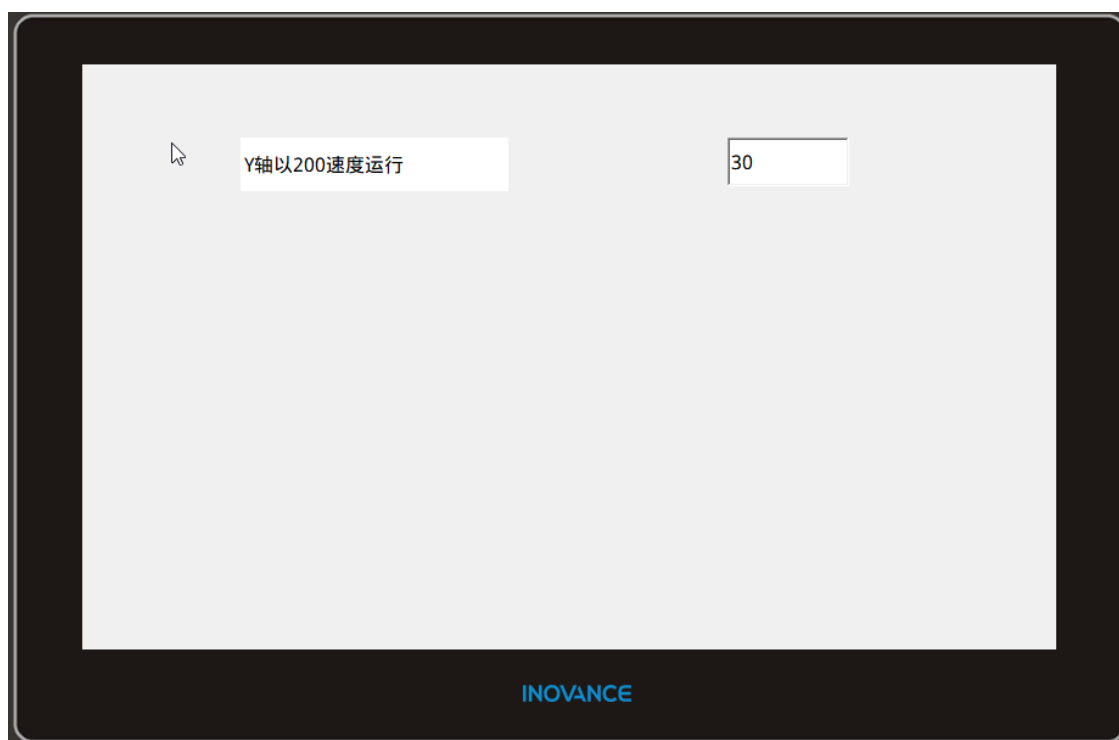
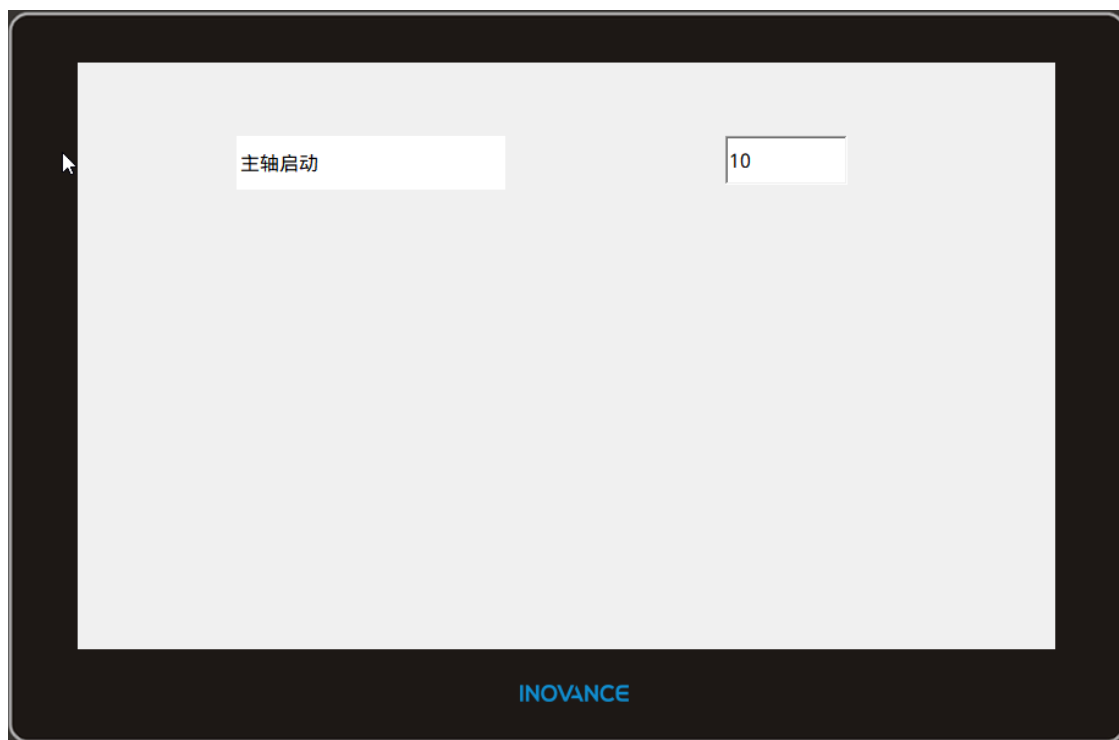
设置：模式为设备文本列表的三种形态，为方便仿真，这里我们设置为输出形式，即只能显示，不能执行操作。

显示：设置选用的文本列表，以及文本列表中可见的项目数目。

过程变量：用来关联的外部变量，这里我们用来关联程序中的流程步序变量D0（D0变量用于关联后面的数值IO域）。

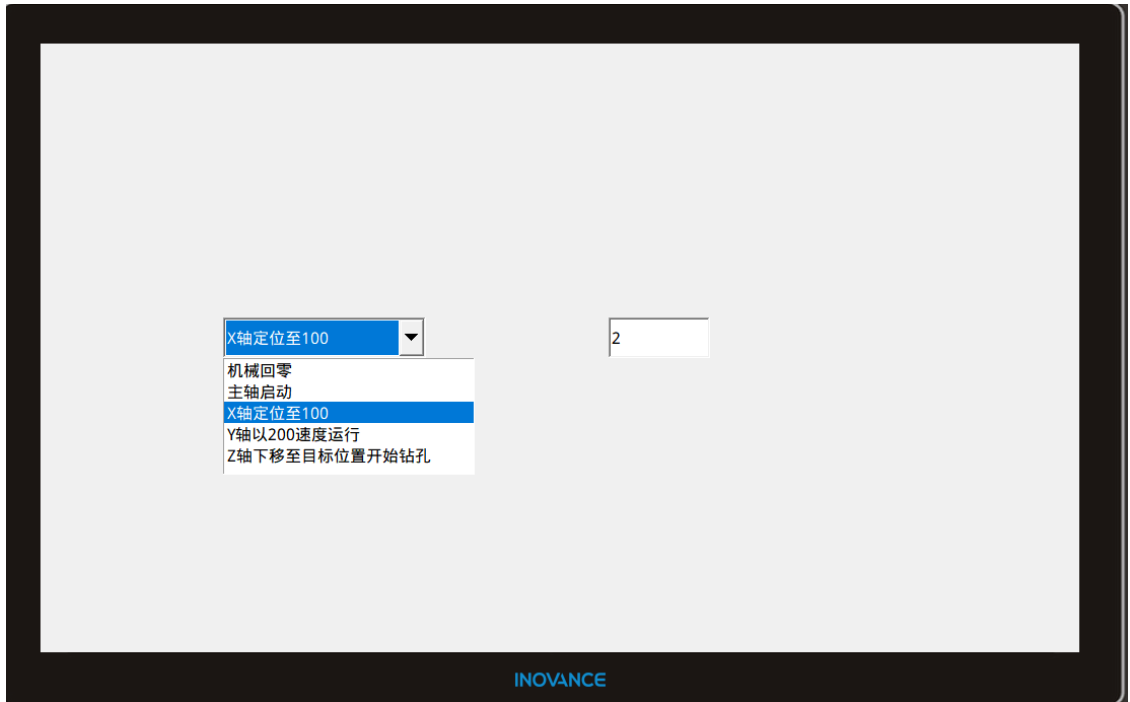
4. 仿真测试，通过在页面新建数值IO域修改D0的值，来观察文本列表的变化，如下图所示。





5. 若需要设置为输入/输出形式，则将条目中数值由范围设置为固定值即可。

	+	编号	数值	条目
1		13	0	机械回零
2		14	1	主轴启动
3		15	2	X轴定位至100
4		16	3	Y轴以200速度运行
5		17	4	Z轴下移至目标位置开始钻孔



## 1.7 使用图形列表实现数据控制图形切换

### 案例要求

1. 通过HMI 软件自带的图形列表，配合图形IO域控件实现数据控制图形的改变。
2. 制作一个图形列表。
3. 将图形IO域关联图形列表与PLC中的过程变量，实现数据控制图形切换。

### 案例目的

通过此案例了解图形列表的功能，学会使用图形列表工具。

### 编程步骤

绘制图形列表

1. 绘制图形列表内容：左侧项目树“状态列表” > “图形列表” 新建一个图形列表，命名为“背景切换”，如下图所示。

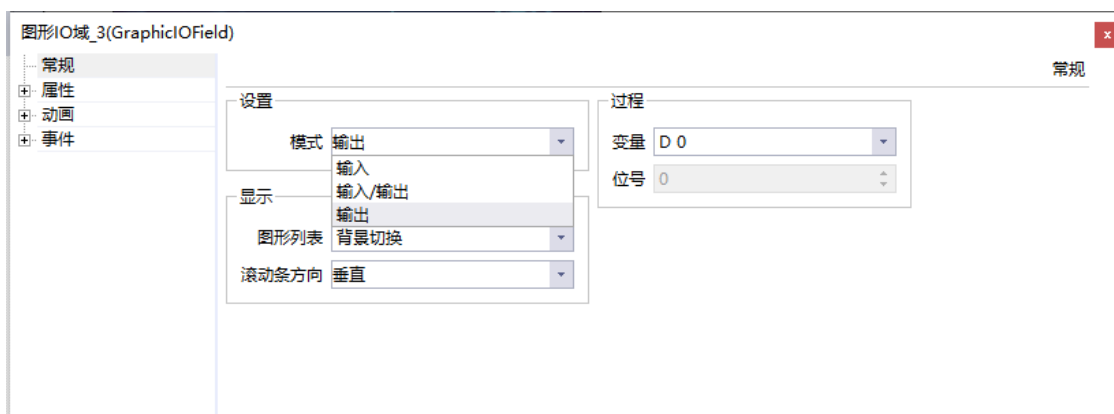


表格表头“选择”意为选择所要关联变量的类型，其中位（0，1）表示变量为BOOL型，可选状态只有0和1；位号（031）表示所选变量为32位整型，用来设置整型变量的某一位；范围（……）表示当所选变量在某一范围时，设置即可生效。此案例我们进行一个多轴运行的单步调试功能，利用图形列表工具进行单步调试，显示当前所处步序。

2. 编辑列表条目：单击选中的图形列表即可选择该列表条目的编辑，如下图所示。

+	▲	编号	数值	条目
1	5	0		1...
2	6	1		6...
3	7	2		1...

3. 在“工具栏” > “简单控件” > “图形IO域”将控件拖动至页面进行组态设置，设置页面如下图所示。



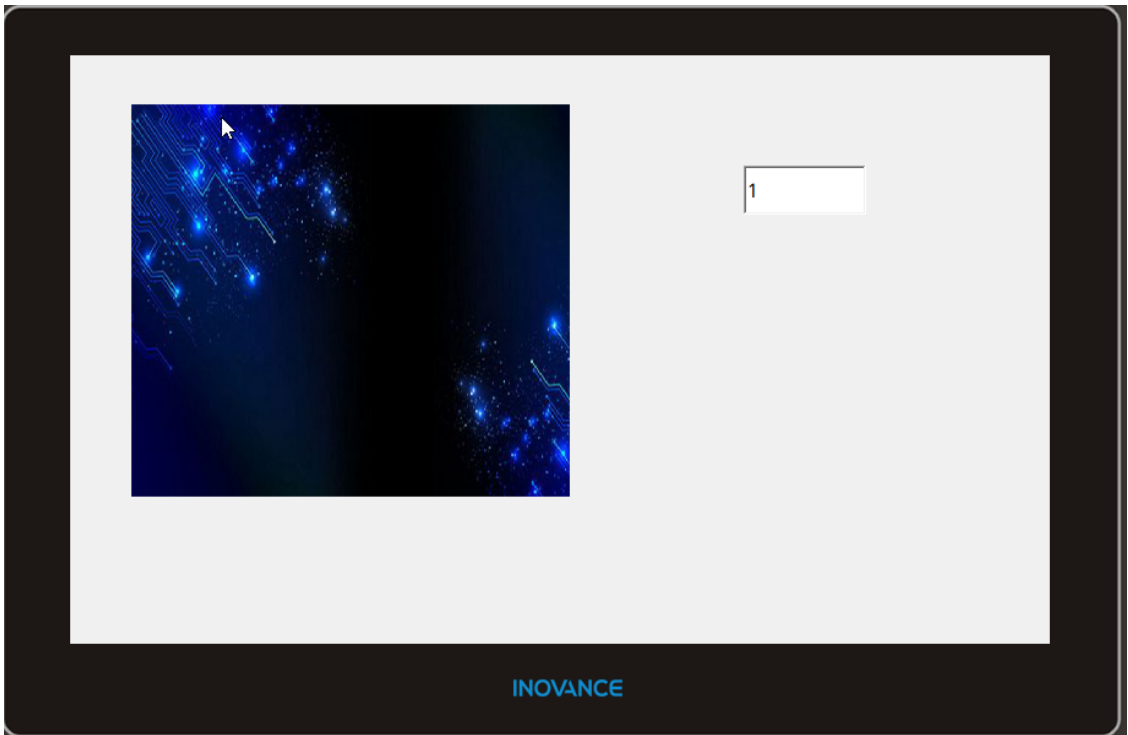
设置：模式为设备图形列表的三种形态，为方便仿真，这里我们设置为输出形式，即只能显示，不能执行操作。

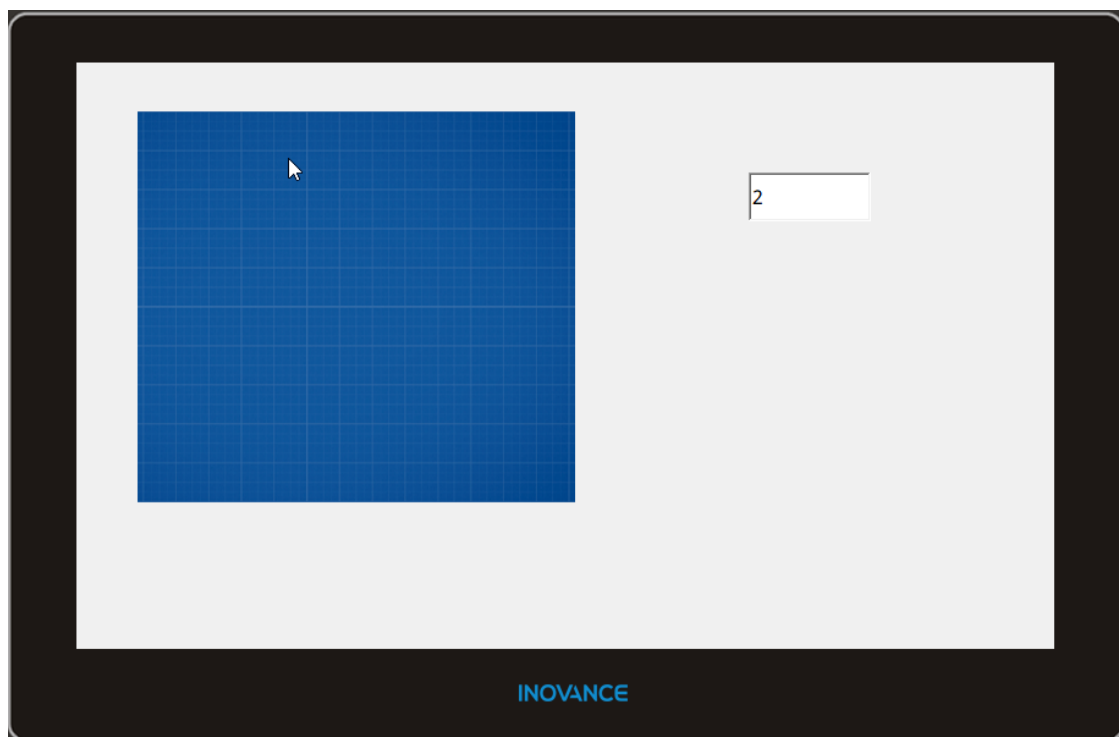
显示：设置选用的图形列表，以及图形列表中可见的项目数目。

过程变量：用来关联的外部变量，这里我们用来关联程序中的流程步序变量D0。

4. 仿真测试，通过在页面新建图形IO域修改D0的值，来观察图形列表的变化，如下图所示。







5. 若需要设置为输入/输出形式，则将条目中数值由范围设置为固定值即可。

## 1.8 将数据和趋势视图关联显示

### 案例要求

1. 通过HMI 的趋势视图控件显示采集的数据。
2. 绘制趋势视图控件用来显示一组变量的变化情况。

### 案例目的

通过此案例了解趋势视图控件的作用，学会趋势视图的使用方法。

### 编程步骤

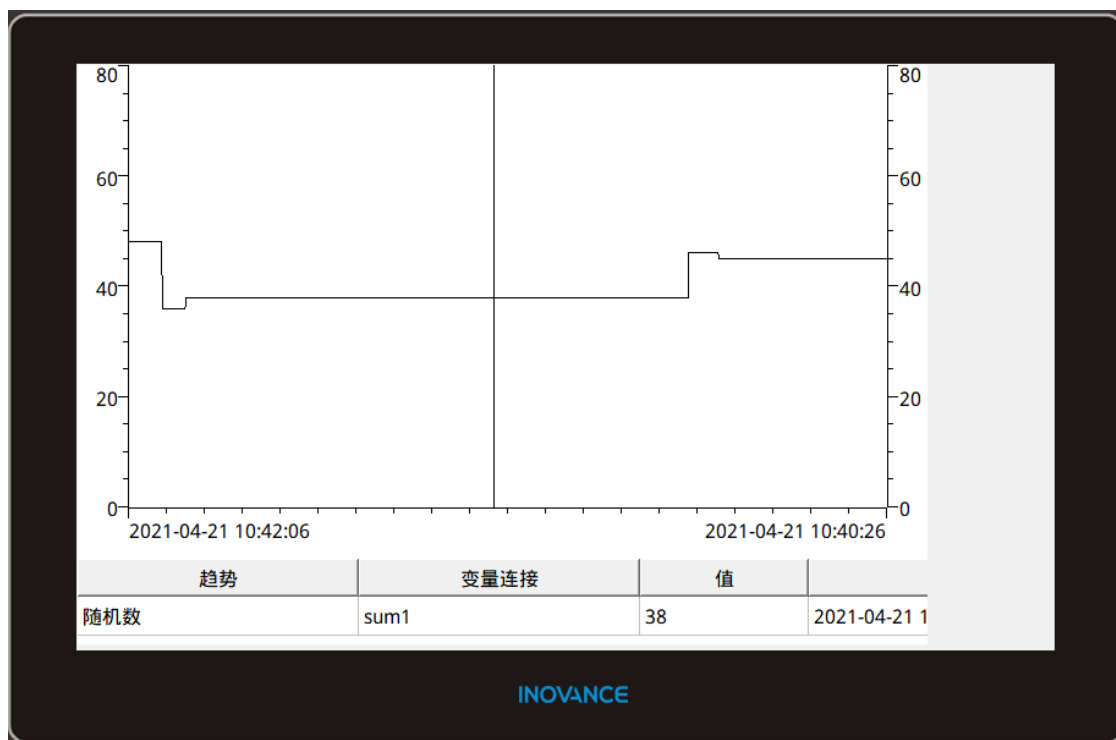
1. 绘制趋势视图控件：在“工具”>“增强控件”找到趋势视图，拖动到页面中，如下图所示。



2. 设置趋势：双击控件进入趋势视图的设置界面，在属性里面进行趋势变量的选择以及其他设置，注必须进行趋势变量的设置。

名称	显示	线类型	画笔宽度	采样点数	趋势类型	趋势
1 随机数	线	实线	1	100	实时周期触发	sun

3. 进行其他设置。
4. 手动输入变量值，观察趋势视图变化，视图显示如下图所示。



## 1.9 应用GIF及表格

### 案例要求

- 通过HMI 软件添加画中画画面实现TAB窗口功能。
- 添加GIF显示器。
- 添加表格控件。
- 添加 .gif格式图片。

### 案例目的

- 通过此案例掌握GIF显示器、表格的基本应用。
- 通过此案例学会GIF显示器如何运用到工程中。

### 编程步骤

#### 1. 添加变量

添加类型变量：左侧项目树“工程” > “通讯” > “变量” > “添加变量组” 添加一个变量组新建所需变量，如下图。

+	名称	编号	连接	数据类型	长度	数组计数	地址
1	GIF开关	7	<内部变量>	Bool	1	1	LB 10
2	1#排风系统	8	<内部变量>	Bool	1	1	LB 11
3	2#排风系统	9	<内部变量>	Bool	1	1	LB 12
4	3#排风系统	10	<内部变量>	Bool	1	1	LB 13

+	名称	编号	连接	数据类型	长度	数组计数	地址
1	1#入口温度	12	<内部变量>	Int16	2	1	LW 20
2	1#出口温度	13	<内部变量>	Int16	2	1	LW 21
3	1#湿度	14	<内部变量>	Int16	2	1	LW 22
4	2#入口温度	15	<内部变量>	Int16	2	1	LW 23
5	2#出口温度	16	<内部变量>	Int16	2	1	LW 24
6	2#湿度	17	<内部变量>	Int16	2	1	LW 25
7	3#入口温度	18	<内部变量>	Int16	2	1	LW 26
8	3#出口温度	19	<内部变量>	Int16	2	1	LW 27
9	3#湿度	20	<内部变量>	Int16	2	1	LW 28

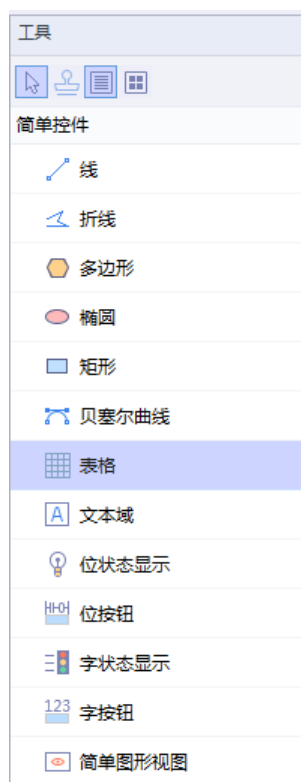
## 2. 添加控件

### a. 添加GIF显示器和表格

从右侧工具栏选择“工具” > “简单控件” > “GIF显示器”，将“GIF显示器”控件拖放到画面中的指定位置。

从右侧工具栏选择右侧工具栏“工具” > “简单控件” > “表格”通过拖拽方式添加控件。

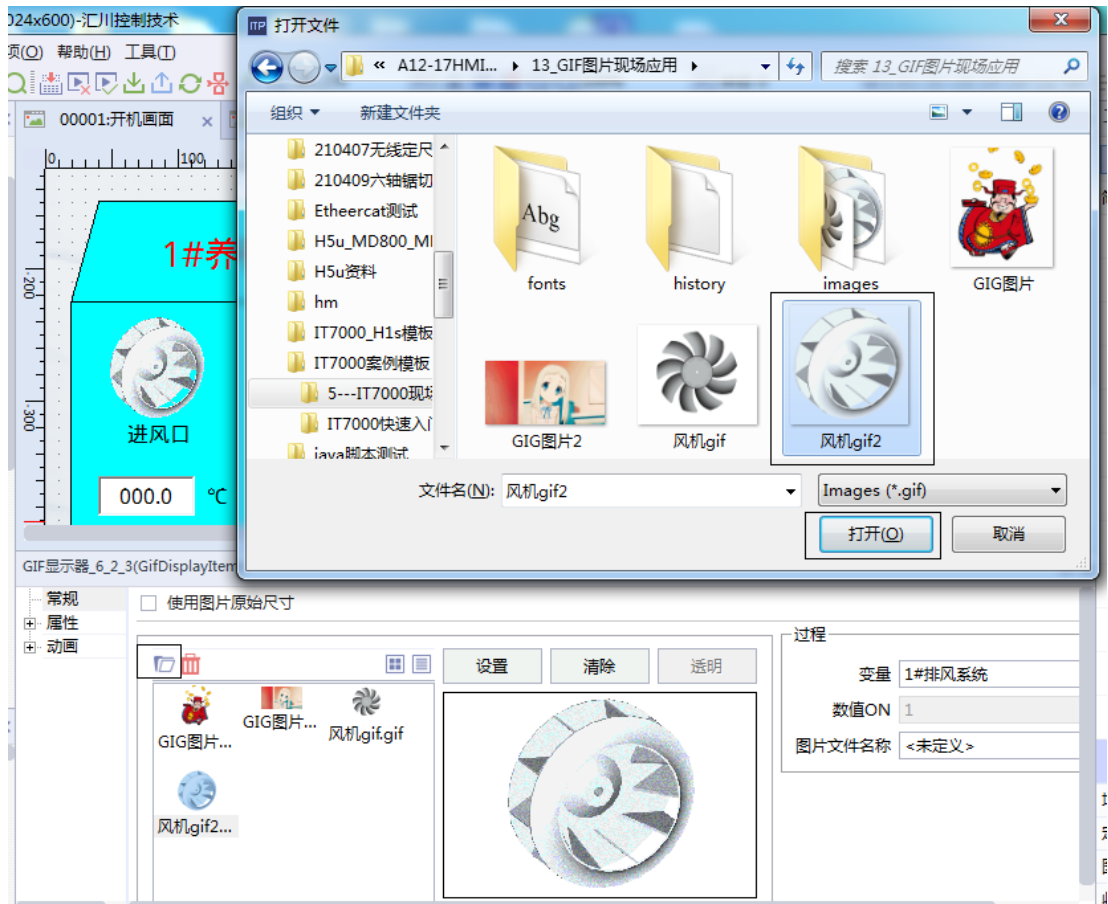




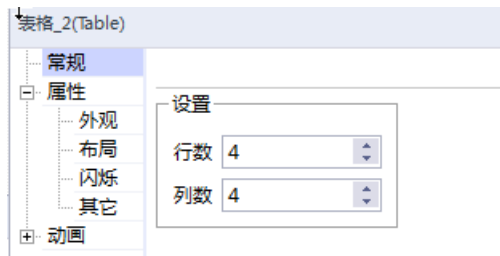
b. 添加GIF图片及控制变量

GIF显示控件属性视图“常规”>“从文件创建新图形”打开存储GIF图片的文件夹，选择并打开所需GIF图片。

GIF显示控件属性视图“常规”>“过程”>“变量”添加控制变量，变量置ON状态显示GIF图片。



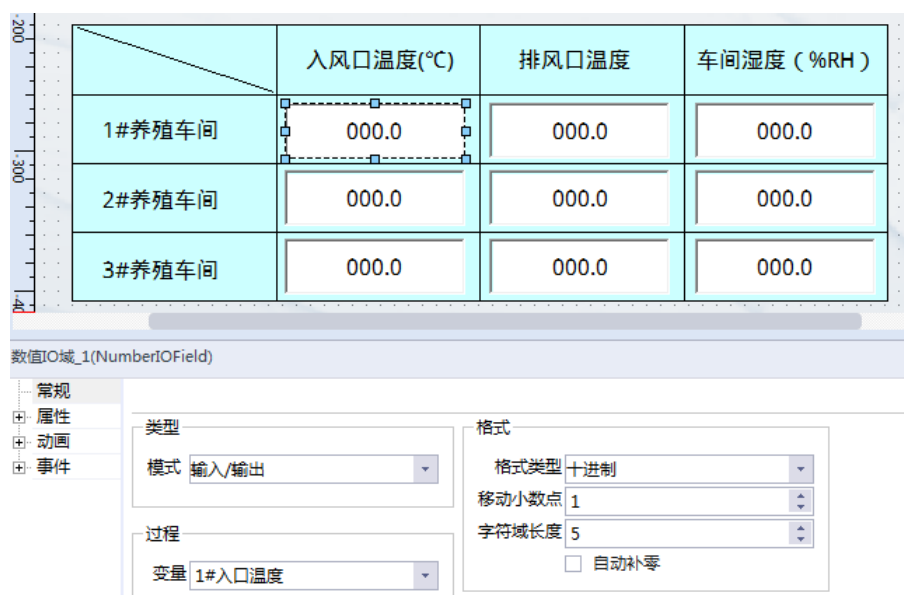
c. 表格控件的常规设置：表格属性视图“常规”>“设置”行数：4，列数：4。



表格属性视图“属性”>“布局”>“位置”X: 160 Y: 210。“尺寸”宽度：583，高度：198。

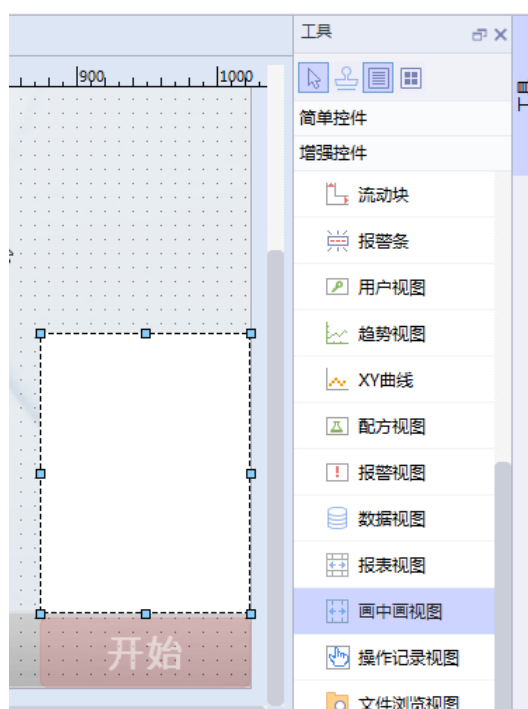


d. 添加数据IO域置放于表格内，并关联对应的过程变量。



### 3. 基本画面添加画中画视图控件

- a. 在开机画面、手动页面、设置页面三个页面分别添加画中画视图控件。



- b. 画中画视图关联过程控制变量 LB 0: “画中画视图” > “常规” > “过程” > “变量” 关联LB 0与“模板\_1”画面开始按钮控件关联地址一致。





- c. 调整画中画视图控件尺寸：点击画中画视图控件“属性”>“布局”>“尺寸”>“宽度”150 “高度”200，尺寸与画中画画面一致。



4. GIF、表格的现场应用案例完成

通过车间排风系统按钮置ON显示动态Gif图片，效果图如下。



通过添加表格、置放数值IO域实现表格数据显示，效果图如下。



## 1.10 应用XY曲线

### 案例要求

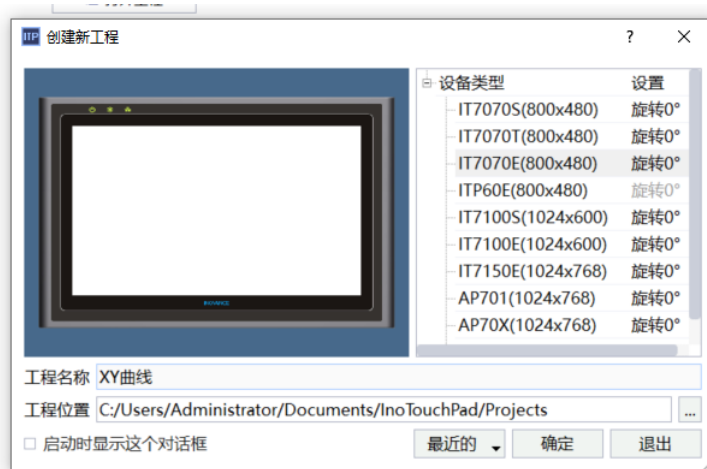
在同一张图像中，显示一组或多组数组变量的变化趋势。

### 案例目的

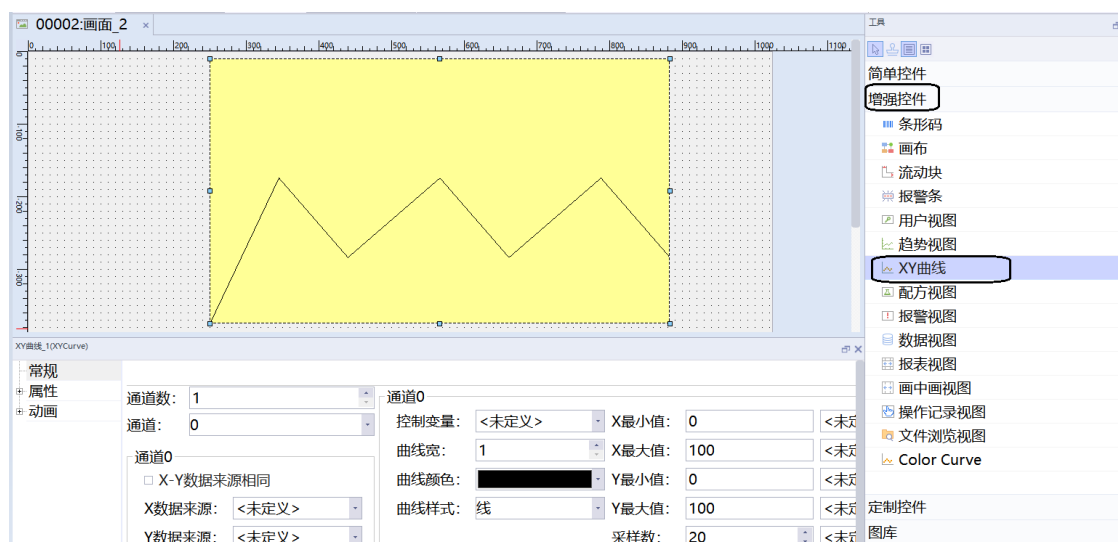
通过应用XY曲线来实现多个变量组的状态变化。

### 编程步骤

1. 用汇川IT7000编程软件新建一个工程。



2. 选中右侧工具栏中“增强控件” > “XY曲线”并拖入主画面中。



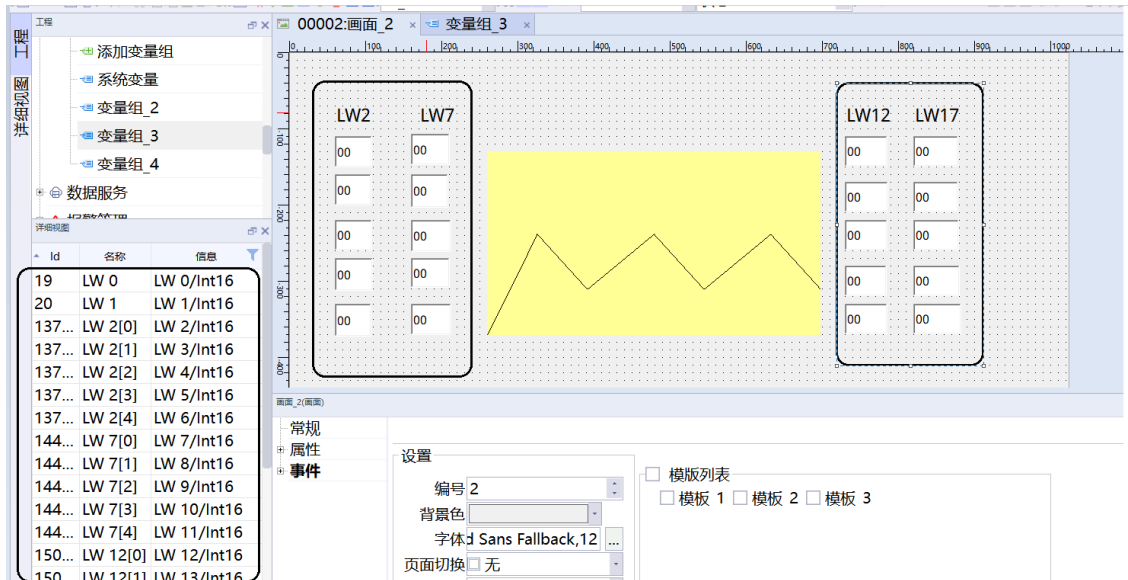
3. 添加所需变量如图所示，两个控制变量和四个数组，分别代表两个通道的X和Y的变量。

+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式
1	LW 1	20	<内部变量>	Int16	2	1	LW 1	100ms	循环使用
2	LW 2	21	<内部变量>	Int16	10	5	LW 2	100ms	循环使用
3	LW 7	22	<内部变量>	Int16	10	5	LW 7	100ms	循环使用
4	LW 12	23	<内部变量>	Int16	10	5	LW 12	100ms	循环使用
5	LW 17	24	<内部变量>	Int16	10	5	LW 17	100ms	循环使用

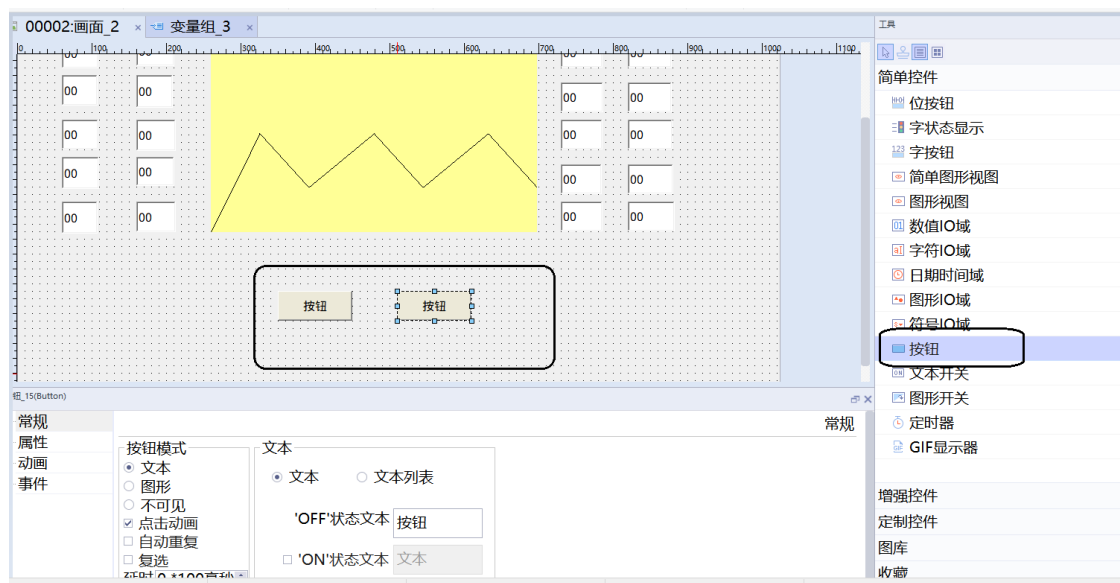
4. 将通道数设为2，并分别设置两个通道的参数，如下图所示：



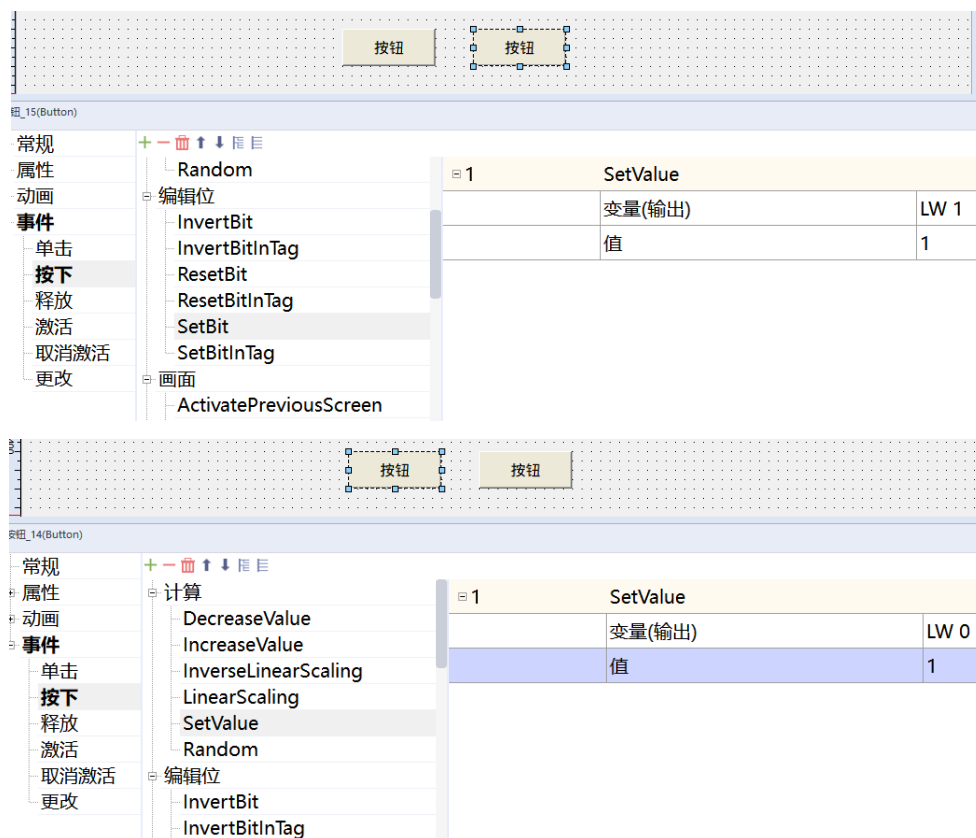
5. 将之前添加的变量拖入到主画面中。



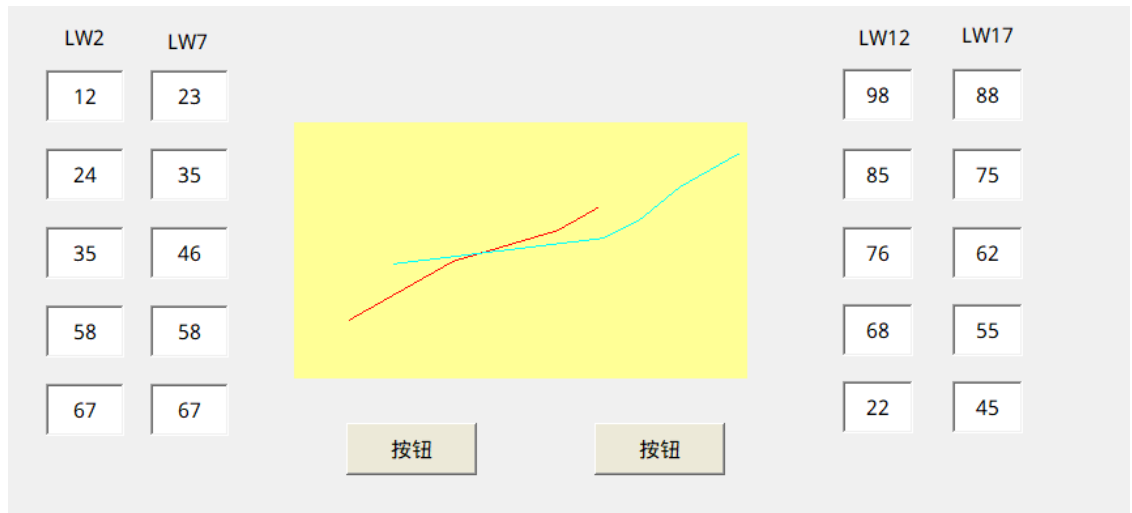
6. 添加两个控件按钮，用来充当XY曲线控制变量。



7. 两个按钮分别“事件” > “计算” > “SetValue”中设定之前添加的控制变量。



8. 进行模拟，设定参数后如下图所示。



## 1.11 设置按钮的可见性、禁用按钮

### 案例要求

按钮可根据实际情况进行可见性和禁用设置。

### 案例目的

按钮可根据实际情况进行可见性和禁用设置。

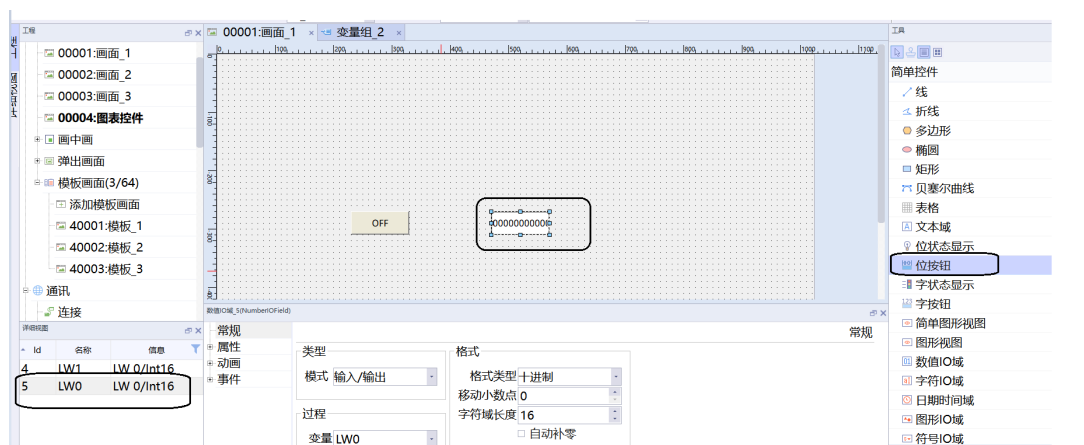
### 编程步骤

- 可见性应用

1. 汇川IT7000新建一个工程，并添加变量如图所示。



2. 在画面中添加可见性的目标按钮，并将刚才添加的内部变量拖到主画面中，如图所示。



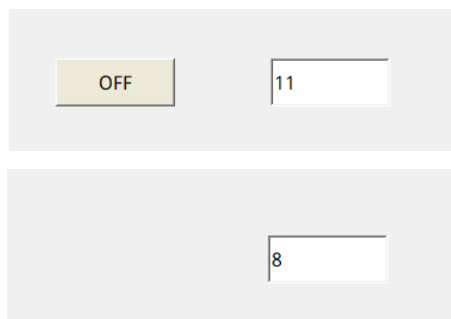
3. 对添加的位按钮进行设置，“动画” > “可见性”，选择启用。



变量选择刚才添加的LW 0，对象状态选择隐藏，类型选择整型数，并添加一定的范围，这里举例0至10，通过以上设置，代表的含义为，当LW 0数值在0至10的范围内，位按钮隐藏。



4. 离线模拟效果如下。



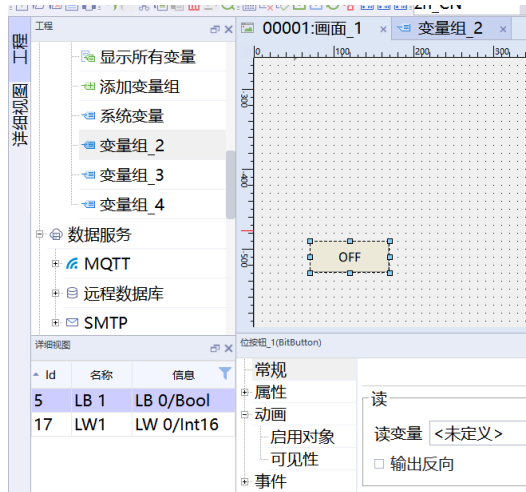
- 禁用应用

对特定按钮进行禁用设定，在实际应用过程中防止误操作。

1. 汇川IT7000新建一个工程，并添加变量如图所示。



2. 在画面中添加禁用的目标按钮，并将刚才添加的内部变量拖到主画面中，如图所示。



3. 对添加的位按钮进行设置，“动画” > “启用对象” 选择启用。



变量选择刚才添加的LB 1，对象状态选择禁用，类型选择位，通过以上设置，代表的含义为，当LB 1位置ON时，位按钮禁用。



4. 离线模拟效果如下。





## 2 画面

### 2.1 应用模板画面和图层

#### 案例要求

- 正确使用模板画面和图层，以便于更好的屏幕编程。
- 了解模板画面如何添加及顺序并在正式画面中显示。
- 学会在画面中使用图层，更加合理布局控件。

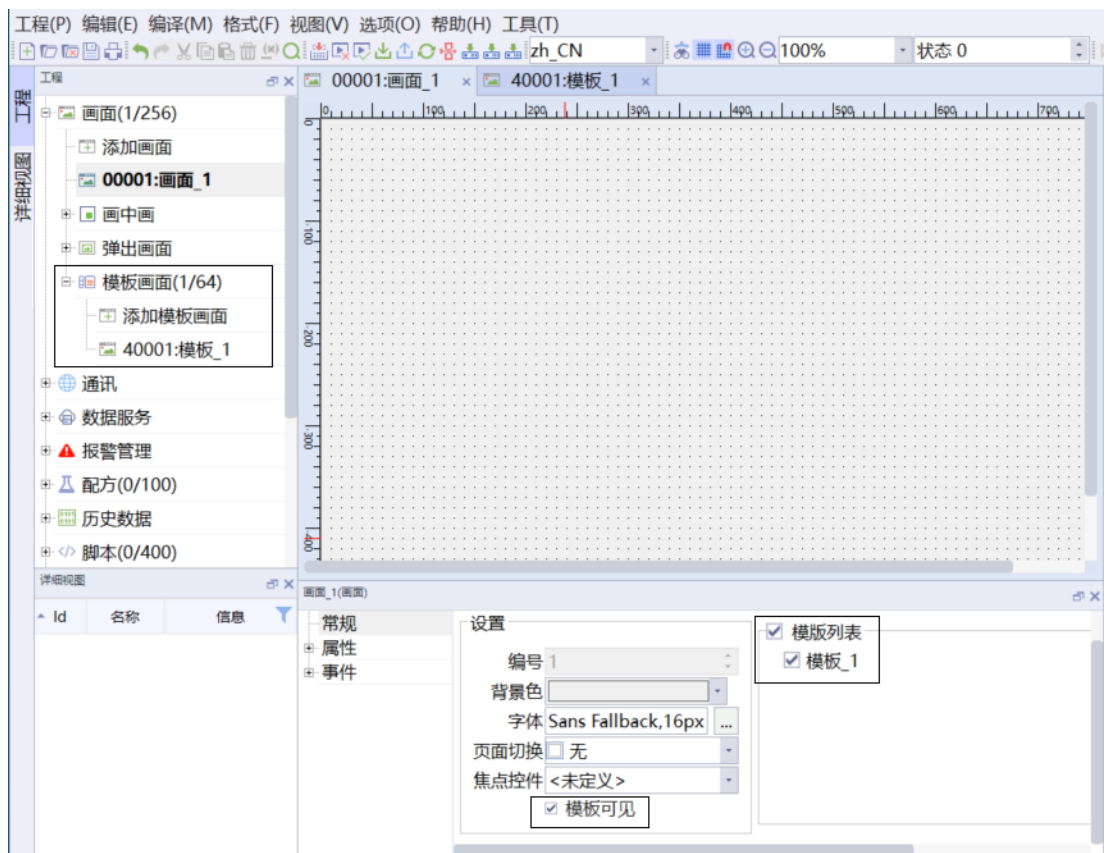
#### 案例目的

- 通过此案例了解模板画面的作用。
- 通过此案例学会如何使用图层布局。

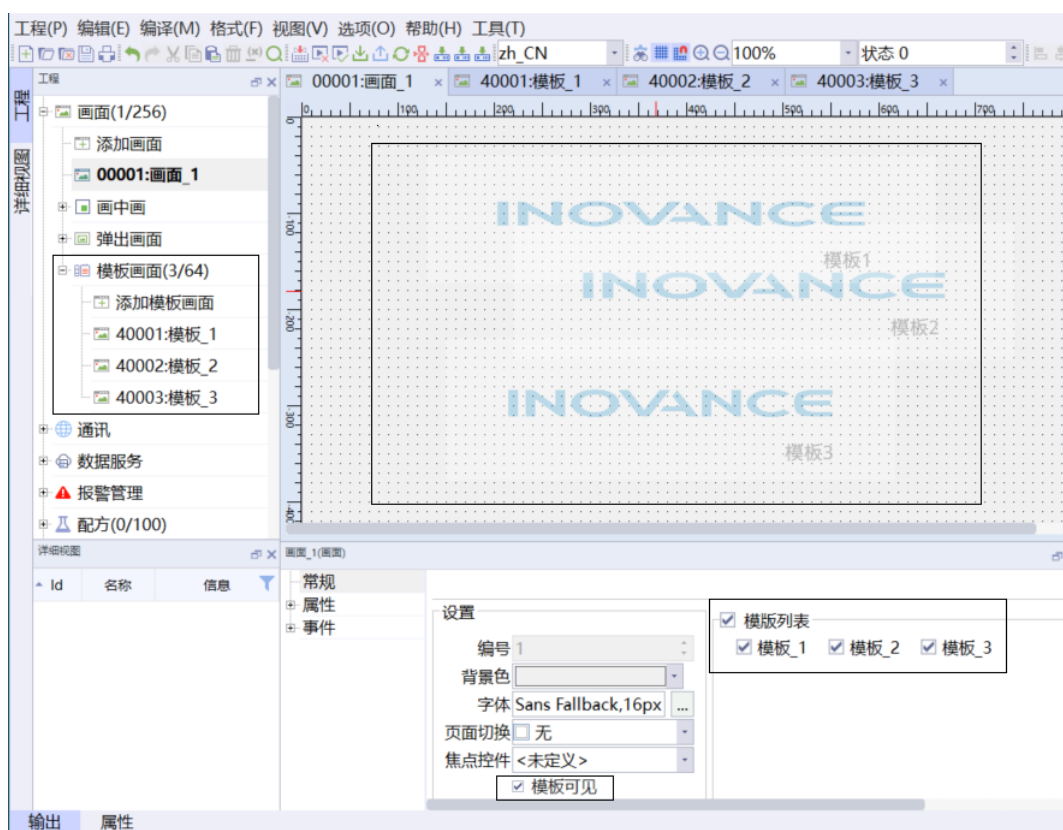
#### 编程步骤

##### 1. 工程中添加模板画面并在主画面中调用

- a. 在左侧工程中点击模板画面展开，点击添加模板画面即可创建模板画面1，这里要注意，模板画面的前后顺序是按照模板画面的编号大小排序，序号小的在前，同时添加模板画面后要在普通页面中的设置里勾选所需模板画面，并勾选模板可见。

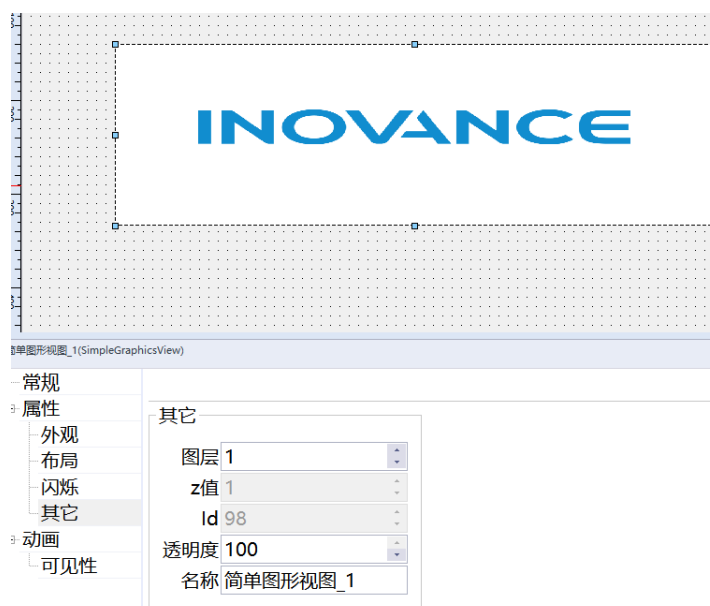


- b. 在模板画面中编辑后显示效果如下，可以根据需求选择需要的模板画面，并调整模板画面的位置。



## 2. 画面中的图层应用

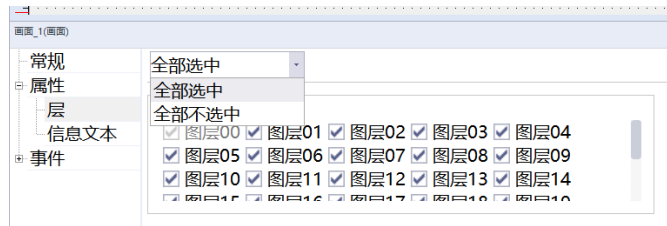
- a. 在画面中任意选取一个控件，这里以简单图形视图为例，点击图片后在下方属性其它中可以看到图层详细设置，将图片设置为图层1。



- b. 然后再添加一个位按钮，在属性其它中同样可以设置图层功能，并将位按钮设置为图层2。



c. 在选择好所需功能后就可以在画面中的属性层中选择需要的图层。



## 2.2 制作弹出式菜单

### 案例要求

实现弹出式菜单功能，用于切换画面，当画面切换后，隐藏弹出的菜单。

### 案例目的

通过此案例了解如何制作弹出式菜单。

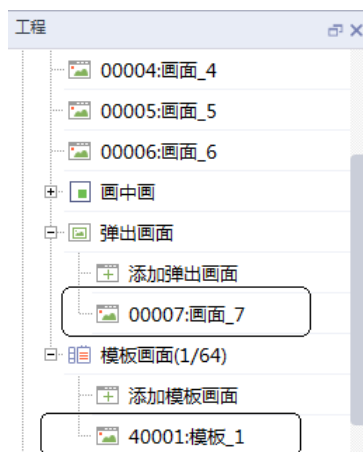
### 编程步骤

#### 1. 添加画面

a. 添加多个普通画面，用于实现功能。

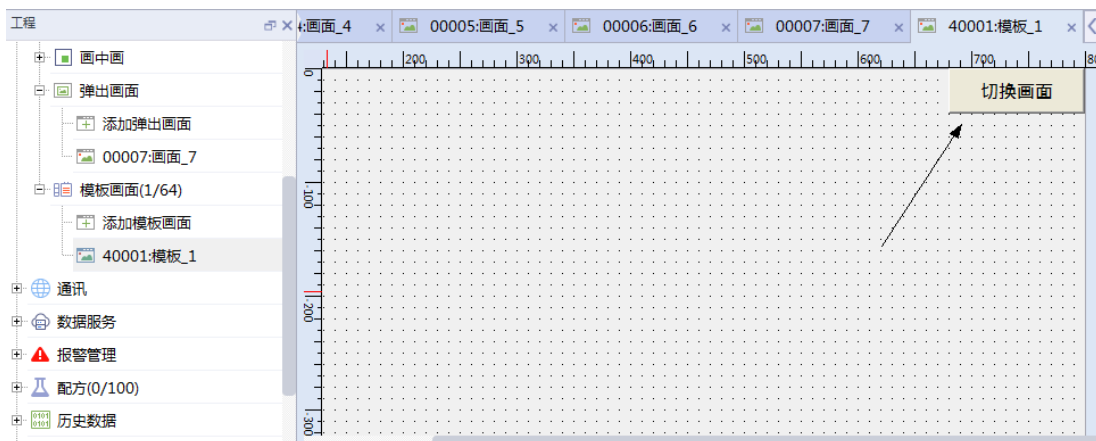


b. 添加一个弹出画面及模板画面备用。

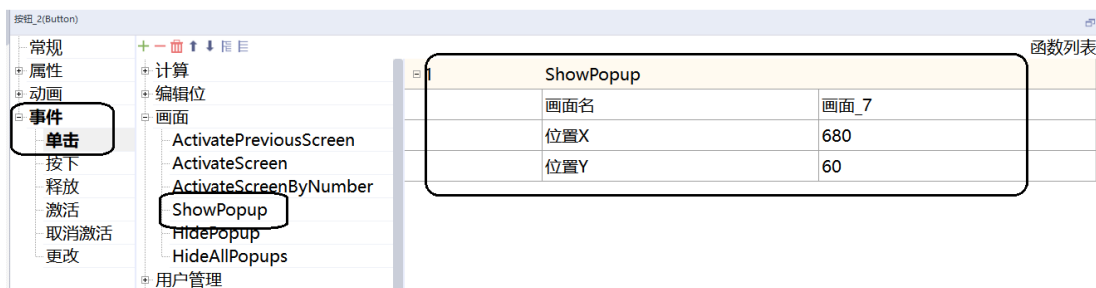


## 2. 组态按钮

a. 添加一个按钮，用于弹出菜单。

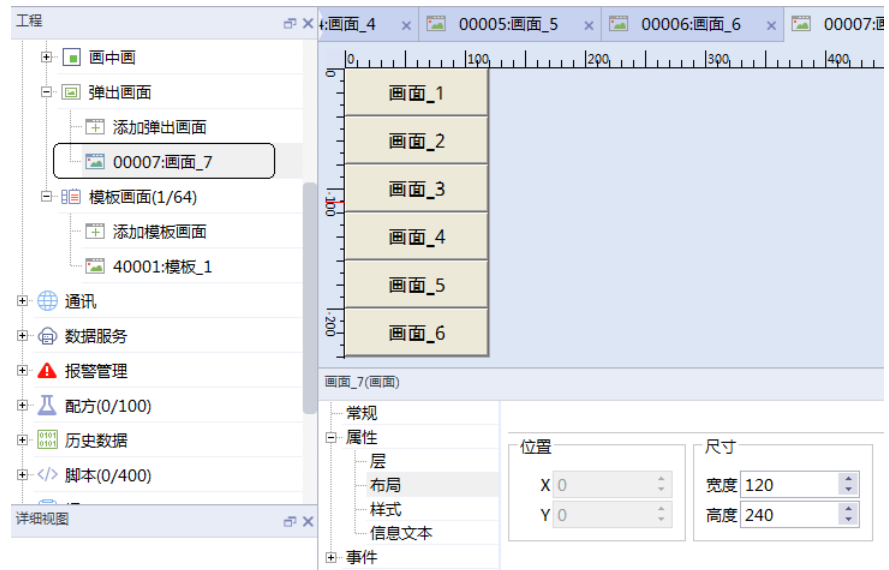


b. 组态系统函数：在添加的按钮中找到“事件” > “单击” > “画面”添加ShowPopup函数，画面名选择第一步新建的弹出画面，位置可根据用户自己的需求设置，该位置即为弹出菜单的位置。



## 3. 组态弹出画面

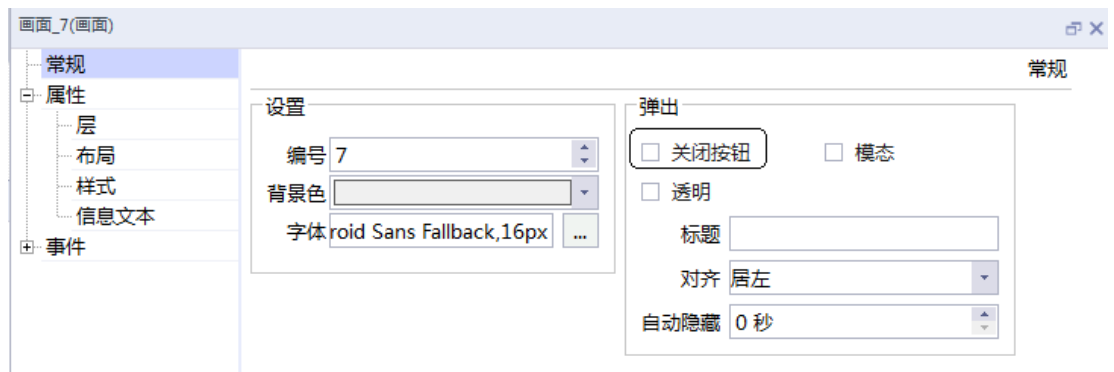
a. 在弹出画面里组态切换画面的按钮，如图所示。



- b. 设置画面尺寸：在画面的“属性” > “布局”中，可以修改弹出画面的尺寸，此处设置的与按钮的尺寸一样大小，这样弹出窗口会比较美观。



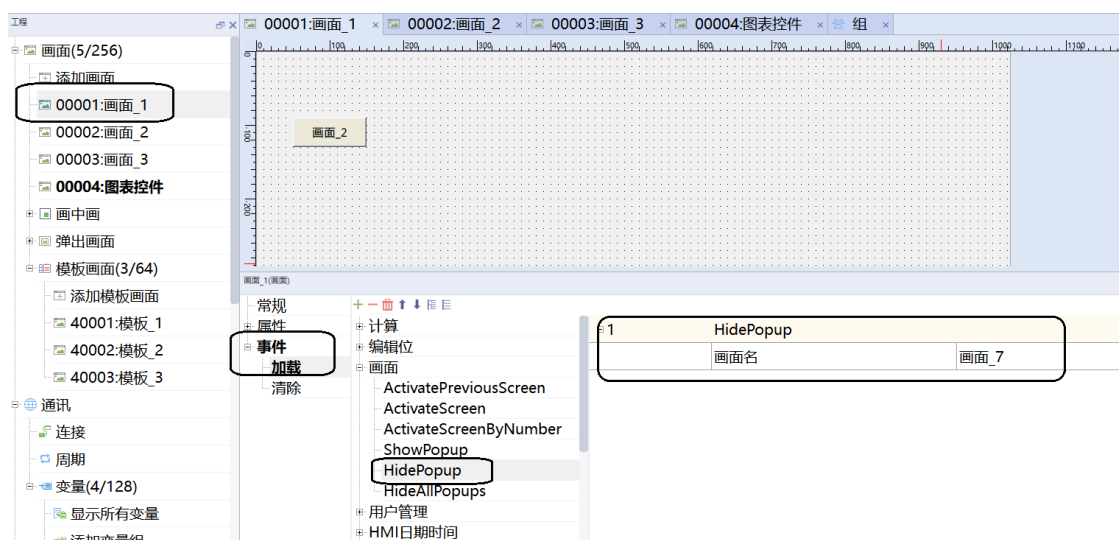
- c. 隐藏关闭按钮：在画面的“常规”中，将关闭按钮一项去除勾选，这样弹出的窗口就不会带有关闭按钮，可使画面美观。



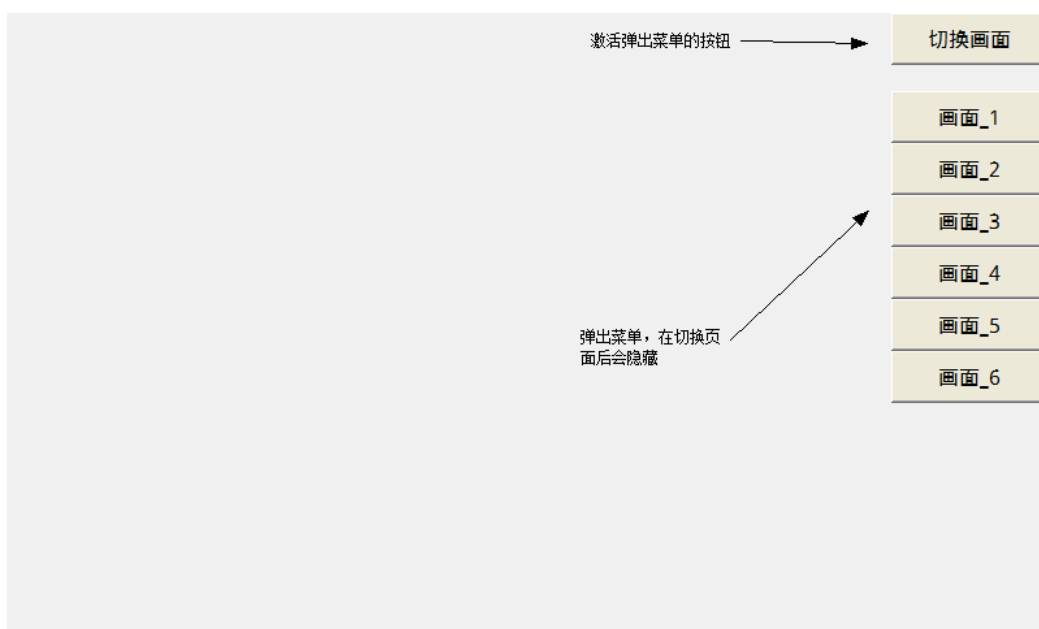
#### 4. 组态画面函数

由于弹出画面不会自动关闭，所以我们要在画面加载时，使用HidePopup函数来关闭弹出窗口。

在普通画面中找到“事件” > “加载” > “画面”添加HidePopup函数，画面名组态弹出画面即可。



## 5. 运行效果展示



## 2.3 使用密码功能控制画面跳转

### 案例要求

使用安全功能给按钮设置“管理”权限，当前用户具有“管理”的权限等级时才允许切换页面。

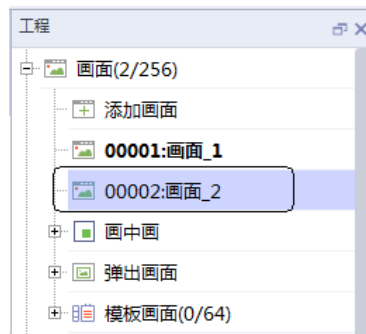
### 案例目的

通过此案例了解如何使用用户权限功能。

### 编程步骤

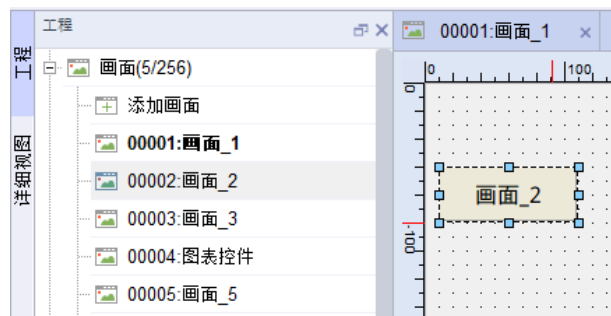
#### 1. 添加页面

添加一个页面，用于测试功能。



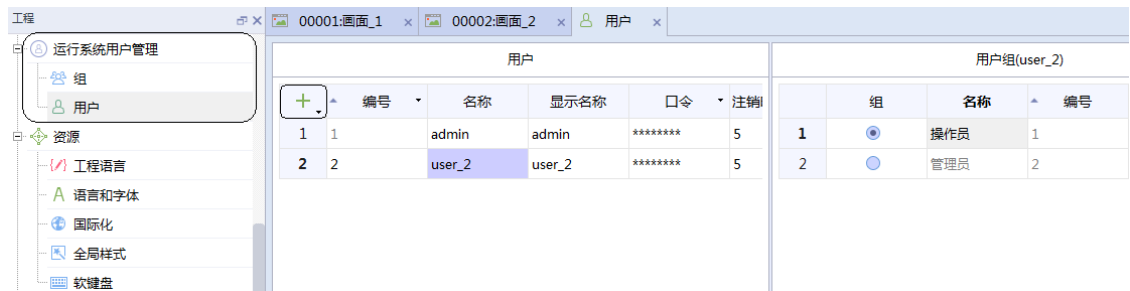
## 2. 添加切换页面按钮

将画面\_2直接拖拽到画面中即可完成切换页面按钮的组态。



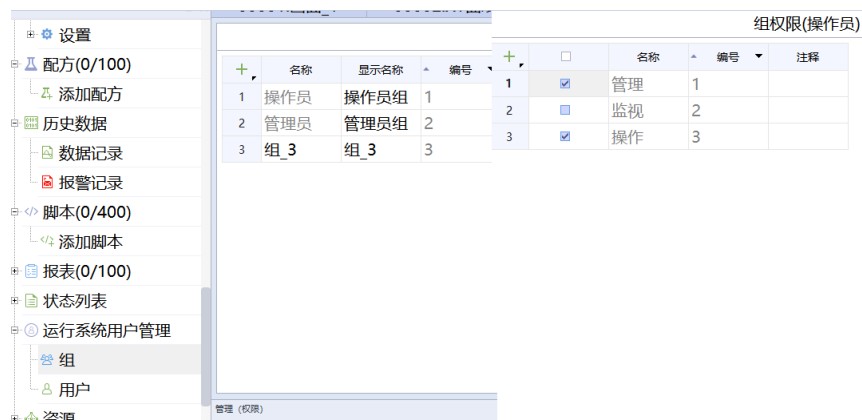
## 3. 添加用户

在“运行系统用户管理” > “用户”里添加用户，口令根据自己的要求设置即可（也可使用默认的admin）。



## 4. 添加用户组

在“运行系统用户管理” > “组”里添加用户组，并添加“管理”组权限。



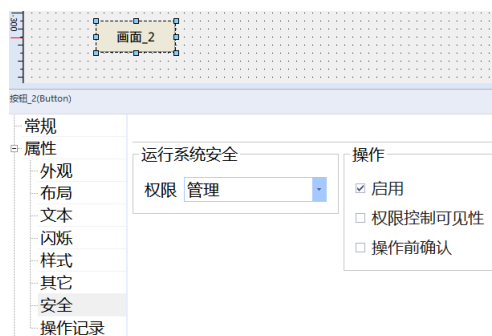
## 5. 设置用户的用户组

回到“运行系统用户管理” > “用户”，设置user\_2的用户组为组\_3，即当登陆user\_2时，当前用户会拥有组\_3的组权限。



## 6. 设置按钮控件的安全属性

回到画面组态，点击切换页面的按钮，在按钮的“属性”>“安全”里设置“权限”为管理。



## 7. 总结

完成上述步骤的设置之后，只有登录具有“管理”组权限的用户时，才能操作切换页面的按钮。

在上述的例子中，当我们登录用户user\_2时具有用户管理组权限。

## 2.4 使用定时器实现画面延时后自动跳转

### 案例要求

通过HMI自带的控件，实现画面延时后自动跳转至另一个画面，该功能通常用于上电界面。

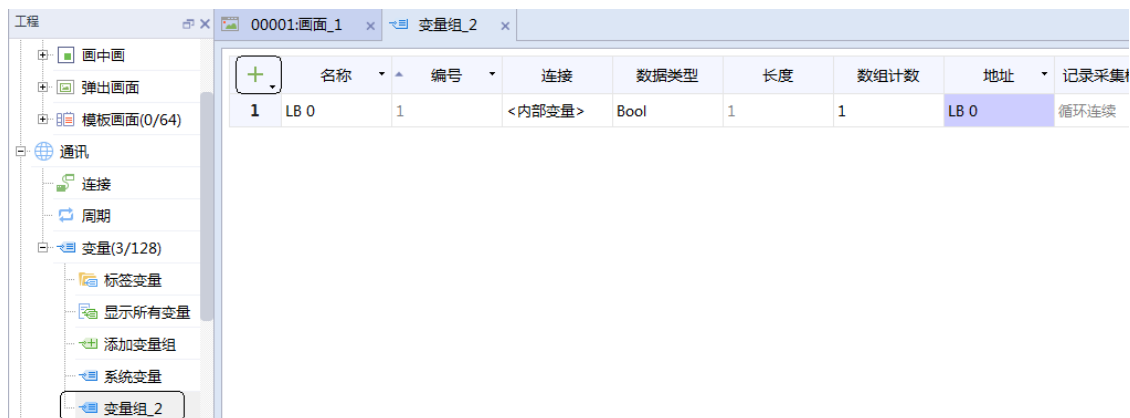
### 案例目的

通过此案例了解如何实现画面延时后自动跳转至另一个画面。

### 编程步骤

#### 1. 添加变量

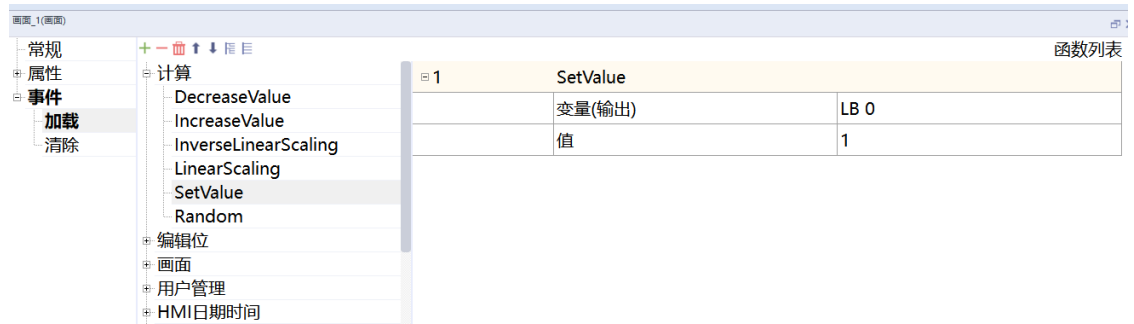
添加一个为Bool类型变量，用于激活定时器。



#### 2. 设置画面事件

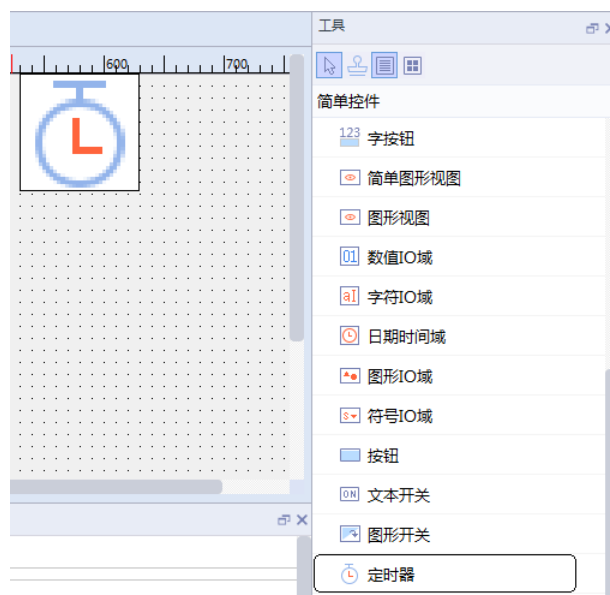


打开第一个画面（这里为画面\_1），选择“事件”>“加载”>“计算”>“SetValue”，变量选择为上一步新建的Bool变量（这里为LB 0），值设置为1。此步骤的目的是通过Bool变量激活定时器运行。

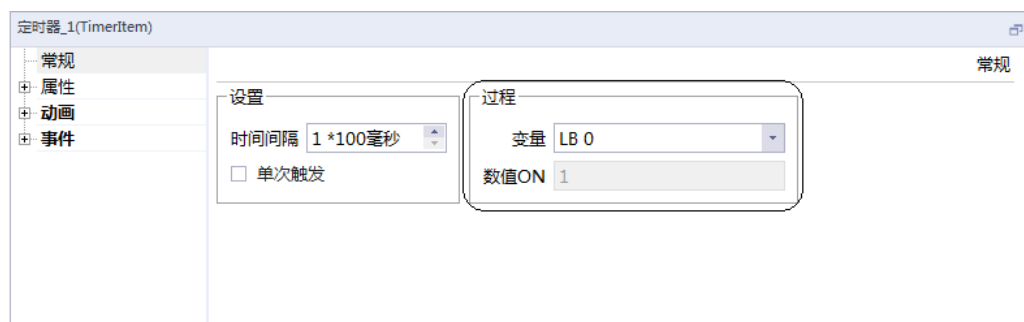


### 3. 组态定时器

a. 在“工具”中找到定时器控件，拖拽到画面中。



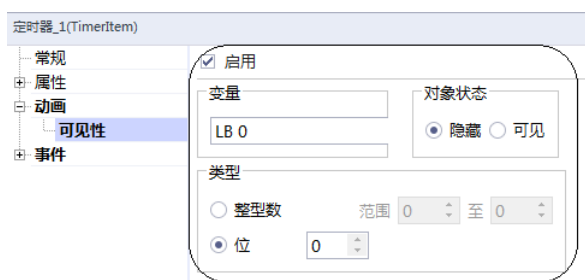
b. 点击添加的定时器，在下方窗口设置定时器的“过程”变量为LB 0，由于我们在画面加载时将LB 0的值设置为1，故定时器会直接启动。



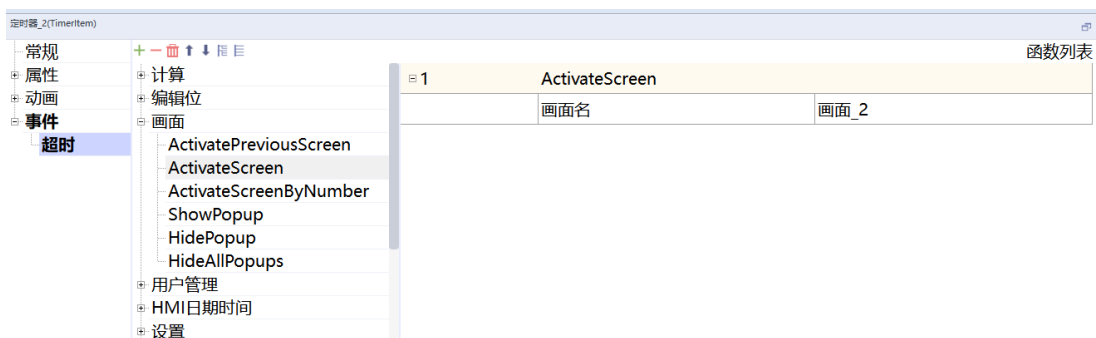
c. 在“常规”>“设置”>“时间间隔”设置定时时间为5S（可根据需要自己设置时间）。



- d. 在“动画” > “可见性”中启用可见性功能，设置“变量”为LB 0，“对象状态”设置隐藏，“类型”选择位。此操作可以在程序运行时隐藏定时器，使画面美观。



- e. 在定时器的“事件” > “超时”中组态ActivateScreen函数，画面名选择为画面\_2，这样HMI当前画面即会自动在定时结束后从画面\_1切换到画面\_2。



## 2.5 高亮显示当前画面标签

### 案例要求

制作一组按钮，用于切换不同画面，并且当停留在当前页面时，对应的按钮保持高亮。

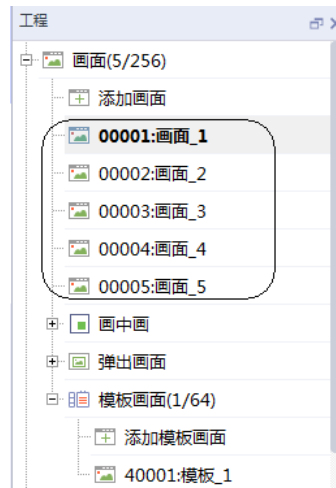
### 案例目的

通过此案例了解如何组态变量的“事件”。

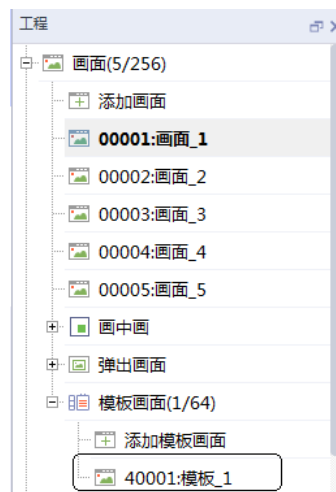
### 编程步骤

#### 1. 添加画面

- a. 添加五个画面，用于测试功能。



b. 添加一个模板画面备用。



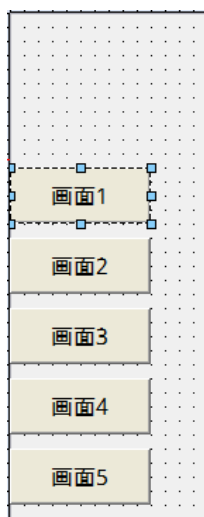
## 2. 添加变量

在变量组中添加如下变量。

+	名称	编号	连接	数据类型	长度	数组计数	地址	记录采集模式	初始值	索引变量
1	画面1	1	<内部变量>	Bool	1	1	LW 0.0	循环连续		<未定义>
2	画面2	2	<内部变量>	Bool	1	1	LW 0.1	循环连续		<未定义>
3	画面3	3	<内部变量>	Bool	1	1	LW 0.2	循环连续		<未定义>
4	画面4	4	<内部变量>	Bool	1	1	LW 0.3	循环连续		<未定义>
5	画面5	5	<内部变量>	Bool	1	1	LW 0.4	循环连续		<未定义>
6	LW 0	7	<内部变量>	Int16	2	1	LW 0	循环连续	1	<未定义>

## 3. 添加按钮

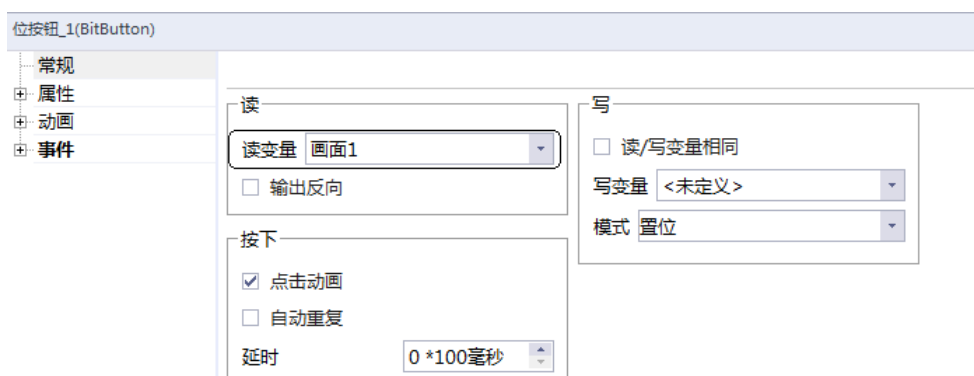
a. 在模板页面中添加位按钮，用于切换画面。



b. 在按钮的“属性” > “状态”里依次设置各个按钮的显示文本，并设置背景颜色，状态0和状态1的颜色需设置不同，状态1的颜色即为高亮时的颜色。

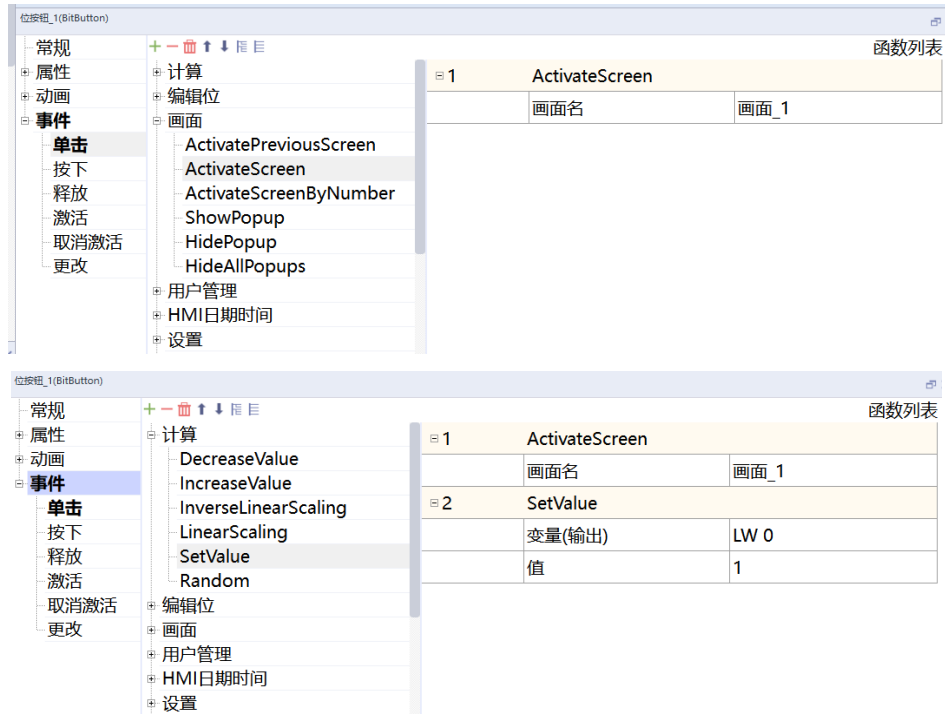


c. 在每个按钮的“常规”里关联对应的读变量，即画面1按钮关联变量画面1，以此类推。



d. 在按钮的“事件” > “单击”组态ActivateScreen函数及SetValue函数。

- 画面1按钮设置ActivateScreen函数画面名为“画面\_1”，SetValue函数变量为LW0，值为1。
- 画面2按钮设置ActivateScreen函数画面名为“画面\_2”，SetValue函数变量为LW0，值为2。
- 画面3按钮设置ActivateScreen函数画面名为“画面\_3”，SetValue函数变量为LW0，值为4。
- 画面4按钮设置ActivateScreen函数画面名为“画面\_4”，SetValue函数变量为LW0，值为8。
- 画面5按钮设置ActivateScreen函数画面名为“画面\_5”，SetValue函数变量为LW0，值为16。



#### 4. 启用模板页面

在模板画面勾选需要显示按钮的画面。



#### 5. 效果展示



## 2.6 在点击按钮时弹出窗口提示

### 案例要求

通过HMI自带的控件，制作一个按钮，点击时弹出确认窗口，当确认写入变量LW 0时，写入变量并关闭弹窗，否则关闭弹出，不写入变量。

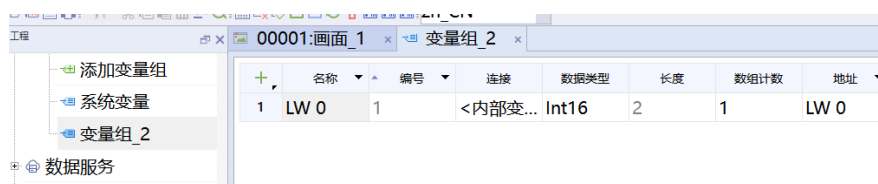
### 案例目的

通过此案例了解如何使用按钮并利用弹出窗口实现输入的功能。

### 编程步骤

#### 1. 添加变量

添加一个数据类型为Int16的变量，备用。

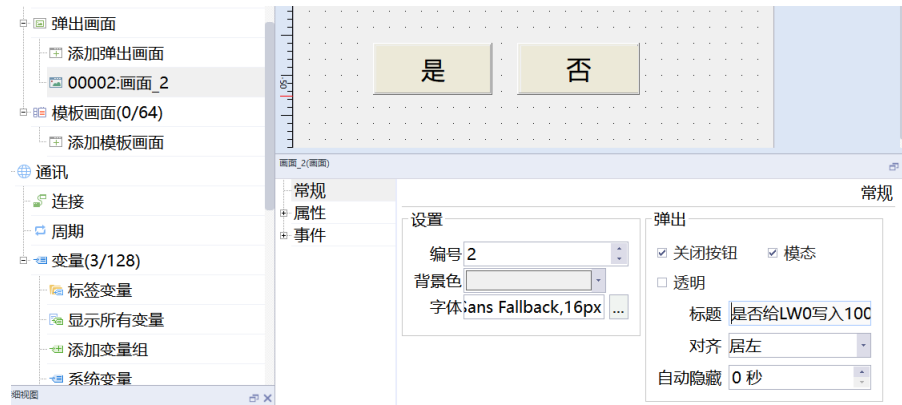


#### 2. 添加画面

a. 添加一个弹出画面，用于显示确认内容，弹窗画面尺寸可根据需求通过“属性”>“布局”>“尺寸”自调。



b. 单击“弹出画面”>“画面\_2”在，在“常规”>“弹出”中添加弹窗标题，此处添加标题为“是否给LW0写入100?”。



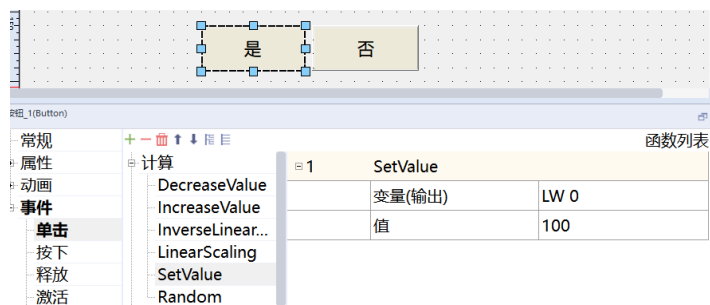
### 3. 添加弹出画面按钮

在弹出画面中添加两个按钮，将文本修改为是/否。

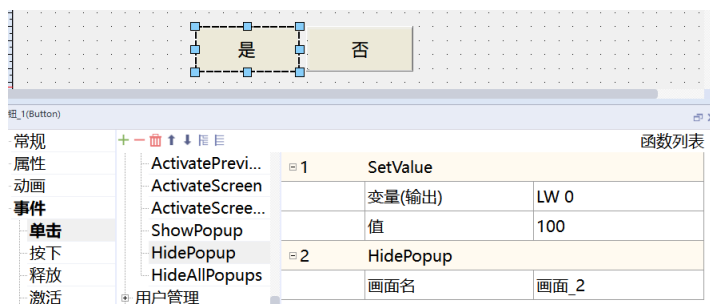


#### a. 设置“是”按钮函数。

- 1). 点击“是”按钮，在“事件” > “单击” > “计算”中添加SetValue函数，变量设置为LW 0，值设置为100。

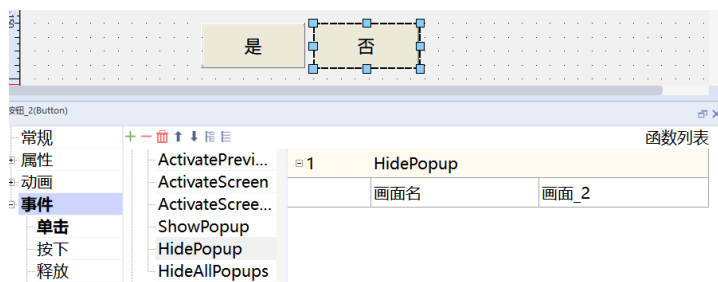


- 2). 在“事件” > “单击” > “画面”中，再添加HidePopup函数，画面名设置为“画面\_2”。



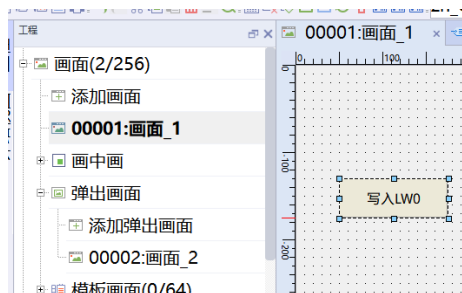
#### b. 设置“否”按钮函数。

- 1). 点击“否”按钮，在“事件” > “单击” > “画面”中添加HidePopup函数，画面名设置为“画面\_2”。

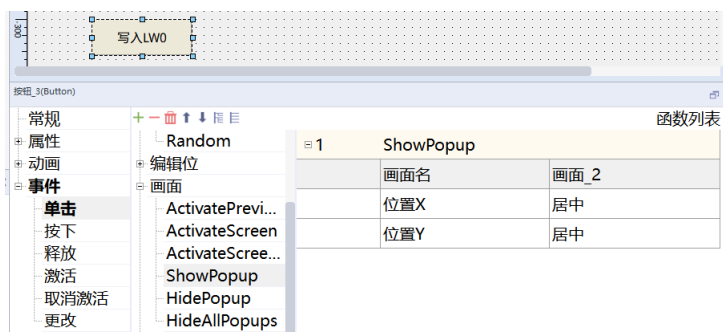


#### 4. 添加主画面按钮

- a. 在画面\_1添加按钮，单击按钮在“常规”>“文本”中写入LW0。



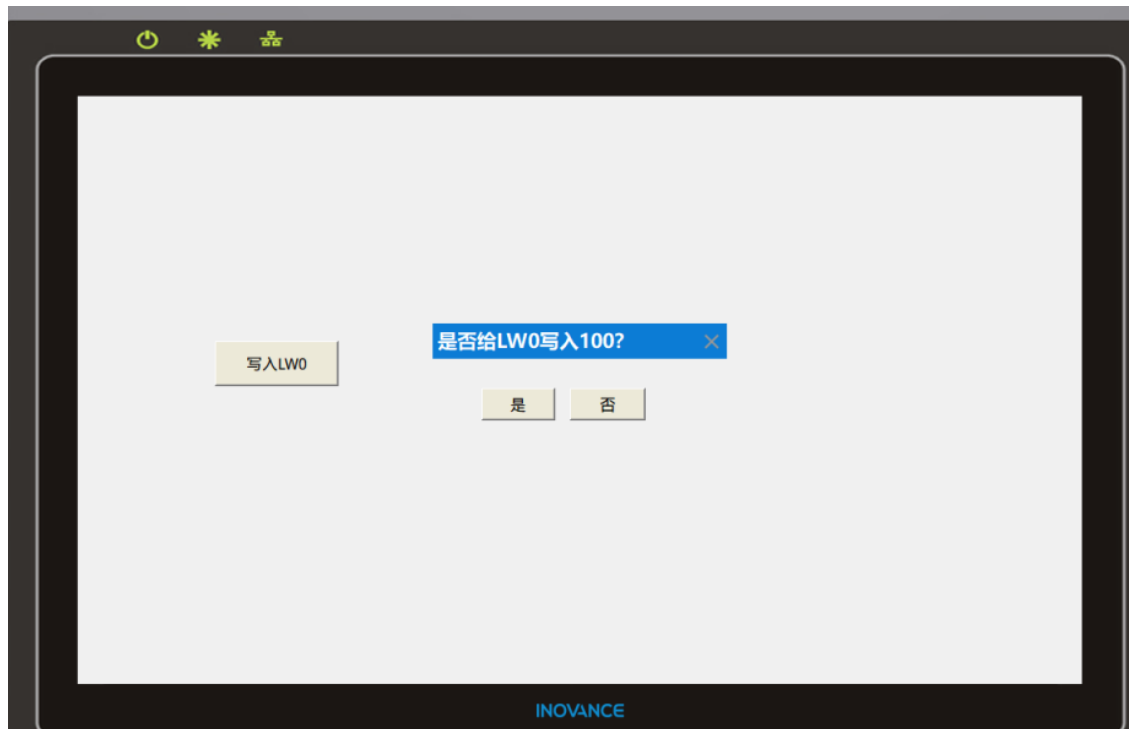
- b. 设置按钮函数：点击“写入LW0”按钮，在“事件”>“单击”>“画面”中添加ShowPopup函数，画面名设置为“画面\_2”



#### 5. 效果展示

当点击“写入LW0”按钮时会弹出确认窗口，点击“是”则写入100到LW0中，同时隐藏弹窗，点击“否”则直接关闭窗口。





## 2.7 使用画中画实现TAB窗口

### 案例要求

通过HMI 软件添加画中画画面实现TAB窗口功能。

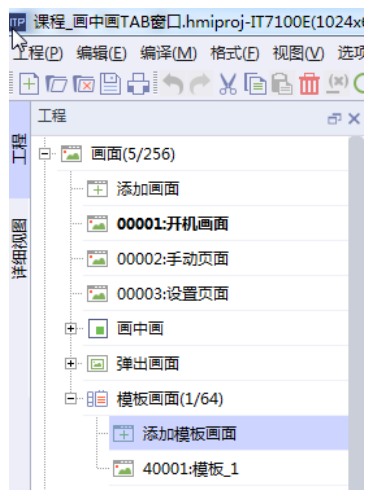
### 案例目的

1. 通过此案例了解画中画功能，学会掌握画中画功能的使用方法。
2. 通过此案例学会使用画中画功能实现TAB窗口切换页面的功能。

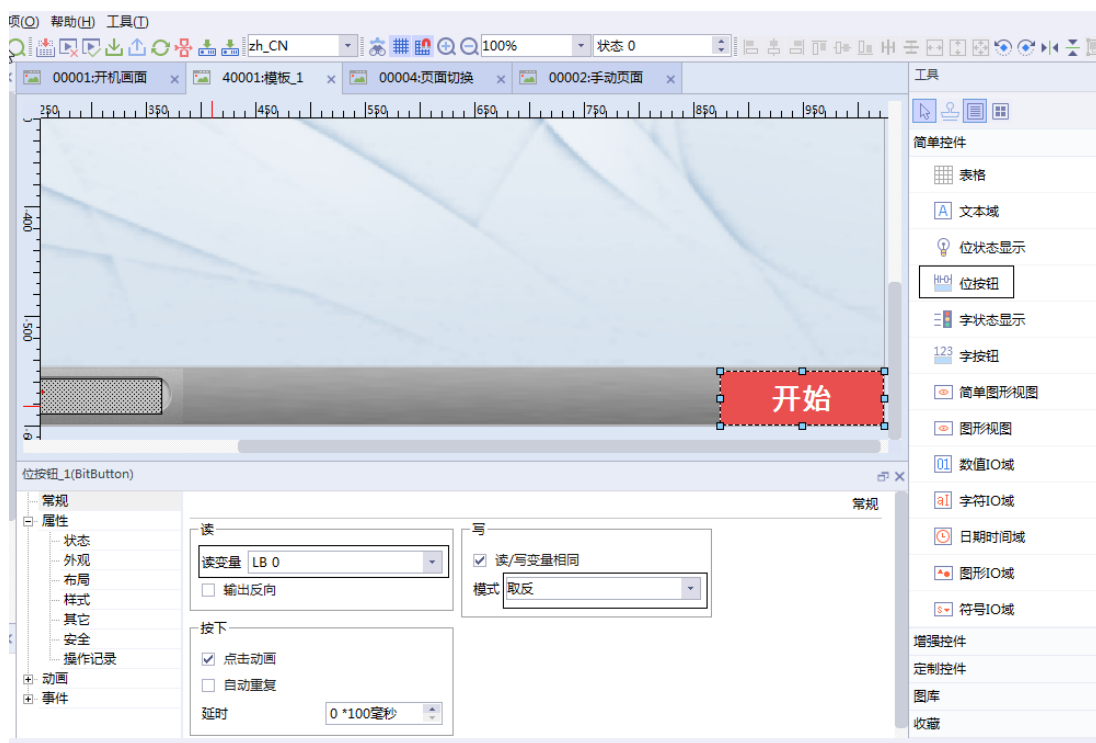
### 编程步骤

#### 1. 添加模板画面

- a. 添加模板画面：左侧项目树“工程”>“画面”>“模板画面”>“”添加模板画面”添加“模板\_1”画面。

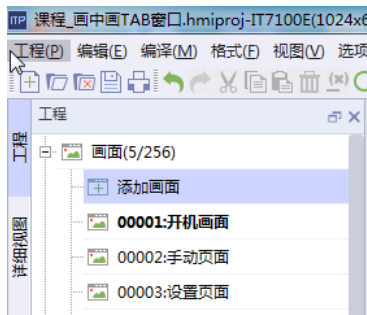


- b. 添加位按钮：右侧工具栏“工具”>“简单控件”>“位按钮”添加一个位按钮，单击位按钮“常规”>“读”>“读变量”中添加一个“LB0”变量，“写”>“模式”改为“取反”。



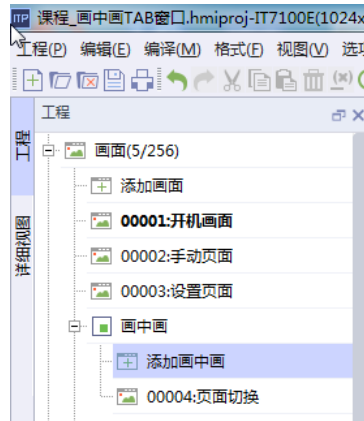
## 2. 添加画面

左侧项目树“工程”>“画面”>“添加画面”添加三个画面，分别命名开机画面、手动页面、设置页面。

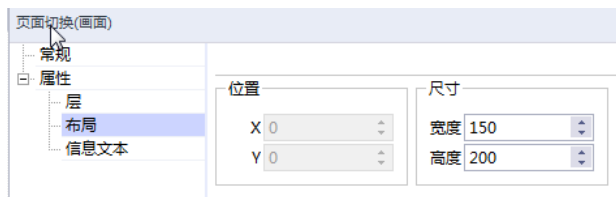


## 3. 添加画中画面

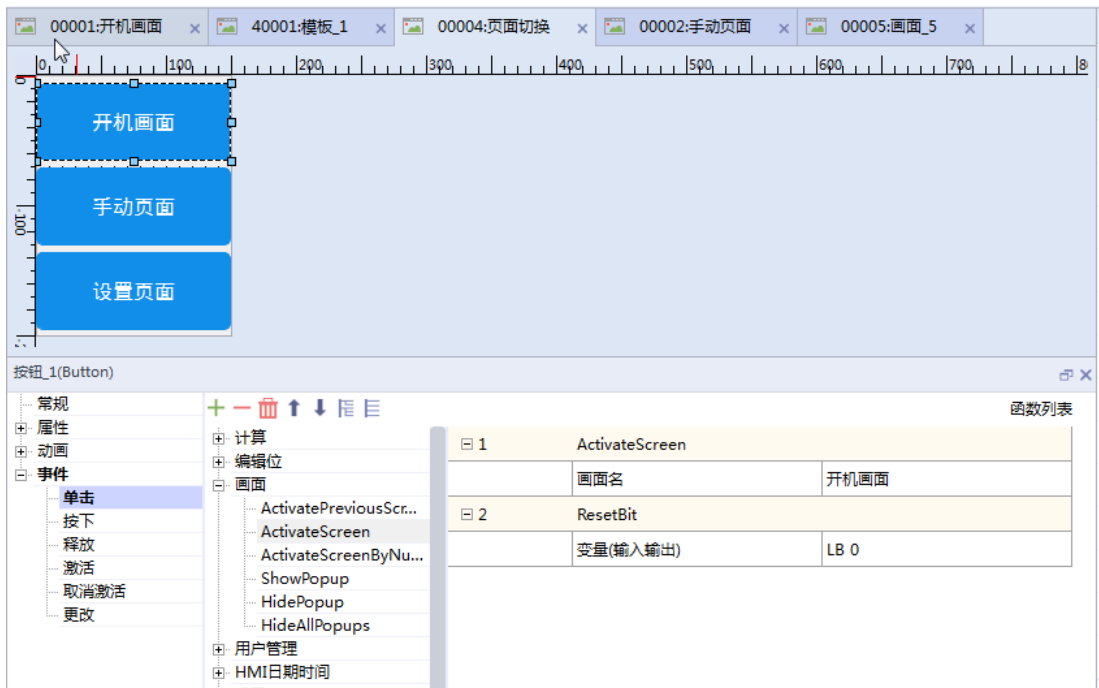
- a. 在界面左侧“工程”>“画面”>“画中画”>“添加画中画”添加一个名为“页面切换”的画中画。



- b. 调整画中画属性：鼠标双击“页面切换”画面“页面切换”>“属性”>“布局”>“尺寸”调整画面宽度、高度，“宽度”150“高度”200。



- c. 添加画面切换按钮：在“页面切换”画面中添加三个按钮，调整三个按钮属性“事件”>“单击”添加 ActivateScreen和ResetBit事件，ActivateScreen用于跳转画面，ResetBit复位关联变量LB 0，用于关闭画中画画面。



#### 4. 基本画面添加画中画视图控件

- a. 基本画面添加画中画视图控件：在开机画面、手动页面、设置页面三个页面分别添加画中画视图控件。



- b. 画中画视图关联过程控制变量 LB 0：“画中画视图” > “常规” > “过程” > “变量” 关联LB 0，与“模板\_1”画面开始按钮控件关联地址一致。



- c. 调整画中画视图控件尺寸：点击画中画视图控件“属性” > “布局” > “尺寸” > “宽度” 150 “高度” 200，尺寸与画中画画面一致。



## 5. 画中画功能实现TAB窗口案例完成

通过画中画功能实现TAB窗口切换页面的功能已完成，效果图如下。



## 3 报警

### 3.1 报警控件的使用方法

#### 案例要求

通过HMI 软件自带的报警功能，实现以下报警效果：

- 配置一组模拟量数据报警，在报警视图显示。
- 配置一组离散量数据报警，在报警视图显示。
- 选择使用相应的系统报警效果。
- 使用报警条滚动显示报警记录。

#### 案例目的

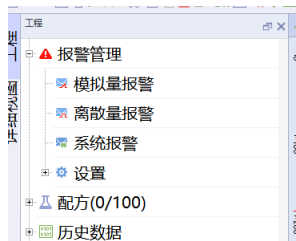
- 通过此案例了解报警的原理。
- 通过此案例学会报警控件的作用及使用方法。

#### 编程步骤

1. 建立连接：工具栏“通讯” > “连接” 建立一个与PLC的通讯连接，此案例以H5U系列PLC为例，建立H5U Qlink TCP 通讯协议，名称为“连接\_1”。

+	名称	编号	通信驱动协议	在线	显示模式	注释
1	连接_1	1	H5U Qlink TCP协议	<input checked="" type="checkbox"/>	十进制	

2. 添加报警数据：在界面左侧“工程” > “报警管理” 分别添加一组模拟量报警和离散量报警数据变量，并做相应设置。



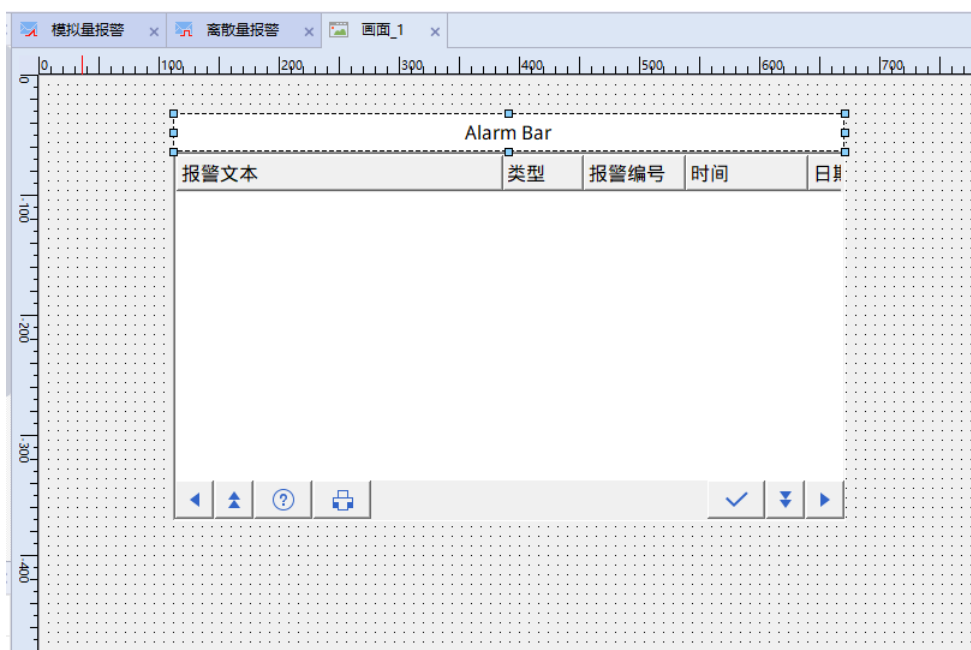
+	文本	编号	类型	触发变量	触发模式	限制	滞后模式	滞后	滞后百分比	延迟	信息文本	报警组
1	报警1	1	错误	D 0	==	10	关	0	关	0		<未定义>
2	报警2	2	错误	D 0	==	20	关	0	关	0		<未定义>
3	报警3	3	错误	D 0	==	30	关	0	关	0		<未定义>

+	文本	编号	类型	触发变量	触发位	信息文本	报警组	确认HMI变量	确认HMI位号	确认PLC变量	确认PLC位号	触发模式
1	离散量_1报警	1	错误	Y 0	0		<未定义>	<未定义>	无位号	<未定义>	无位号	0->1
2	离散量_2报警	2	错误	Y 1	0		<未定义>	<未定义>	无位号	<未定义>	无位号	0->1

3. 报警设置表头介绍。

- 文本：所要显示的报警标题，用户可自由设置。
- 编号：新建的报警数目编号，不可修改。

- 类型：报警类别，可根据需要自行定义，一般分为错误、警告和系统，分别对应不同的报警信息。
  - 触发变量：触发报警的变量，其中离散量报警的触发变量可为BOOL、Int6、UInt16型变量，模拟量报警的触发变量可为Int16、UInt16/UInt64、Int32/Int64、UInt32、Float、Double型变量。
  - 触发模式：变量的触发方式，模拟量报警可设置为<、>、==、>=、<=；离散量报警可设置为0>1、1>0、==0、==1。
  - 限制：根据模拟量报警的触发方式进行设置值。
  - 信息文本：用于显示报警内容的详细信息。
  - 报警组：将报警信息划分到不同的报警组，进行分类和区分。
4. 如下图所示，添加一个画面\_1，在“工具”>“增强控件”里选择报警条和报警视图，拖动到画面中进行组态设置。



5. 报警条信息设置：双击报警条控件，视图下方会出现控件参数设置窗口，如下图所示。



6. 若需要对报警信息进行记录，则需要使用数据视图控件，首先在界面左侧“工程”>“历史数据”>“报警记录”添加一组需要记录的报警记录。

+	名称	编号	每个记录的数据..	路径	记录报警文本和..	记录方法	填充量	运行系统...运行系统启动时..	注释
1	报警记录_1	1	500	本地磁盘	开	循环记录	90	开	添加数据到现...

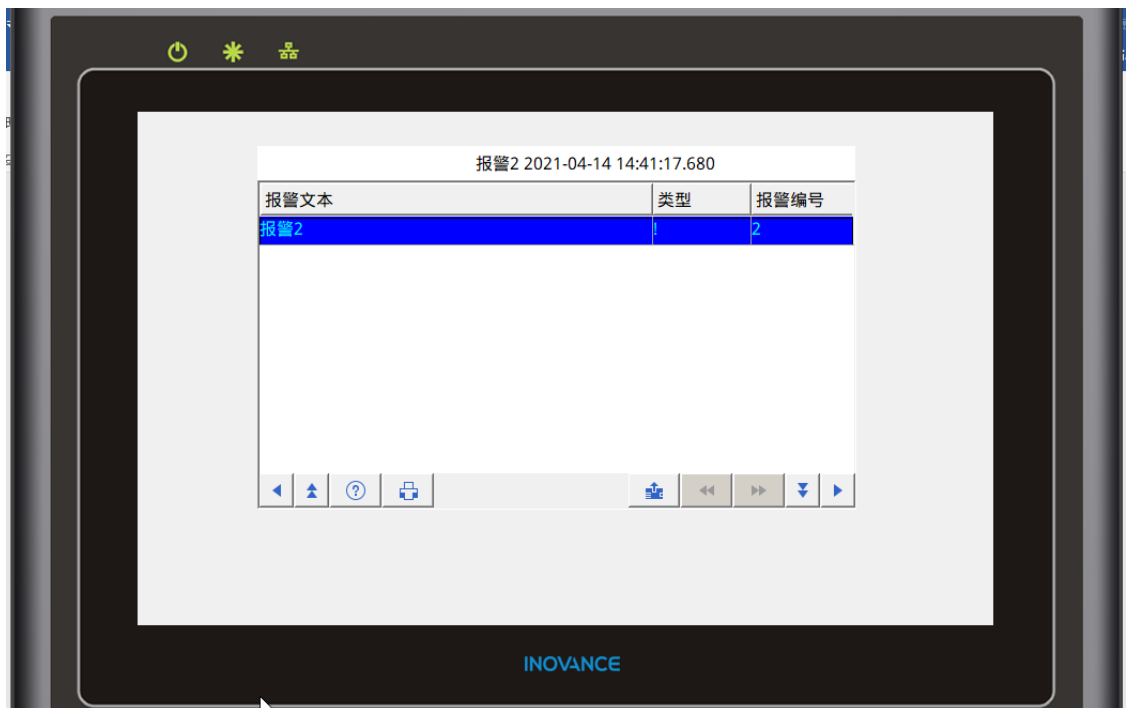
- a. 在“项目树”>“报警管理”>“设置”>“报警类别”中设置所要记录的报警类别。

+	名称	显示名称	确认	记录	到达的颜色	到达并离开的颜色	到达并确认的颜色	到达离开并确认的颜色	到达的文本	离开的文本	确认的文本	编号
1	错误	!	*到达*时	报警记录_1	#ff0000	#ff0000	#000000	#000000	C	D	A	1
2	警告	#	关	报警记录_2	#ffff00	#ffff00	#000000	#000000	C	D	A	2
3	系统	\$	关	<未定义>	#ff00ff	#000000	#000000	#000000	C	D	A	3

b. 在报警视图空间中选择需要进行记录的报警数据。



7. 完成之后通过离线仿真模式触发报警进行测试，测试结果如下图所示，当所给值等于10时报警条滚动显示当前报警，同时报警视图记录下报警数据。



## 3.2 制作报警弹出窗口

### 案例要求

通过HMI完成以下功能：

1. 学会弹出画面的操作方法。
2. 了解模态与非模态画面的区别。
3. 完成报警窗口弹出。

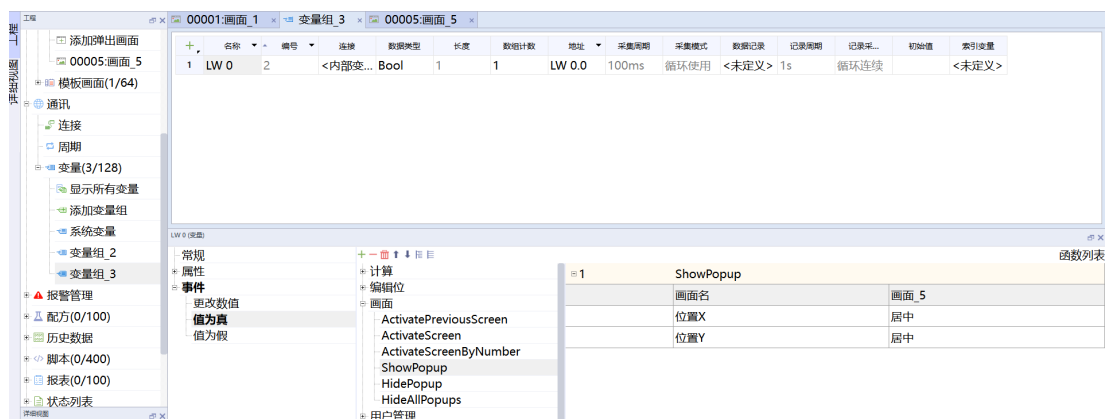
### 案例目的

通过此案例学会报警弹出窗口的使用。

## 编程步骤

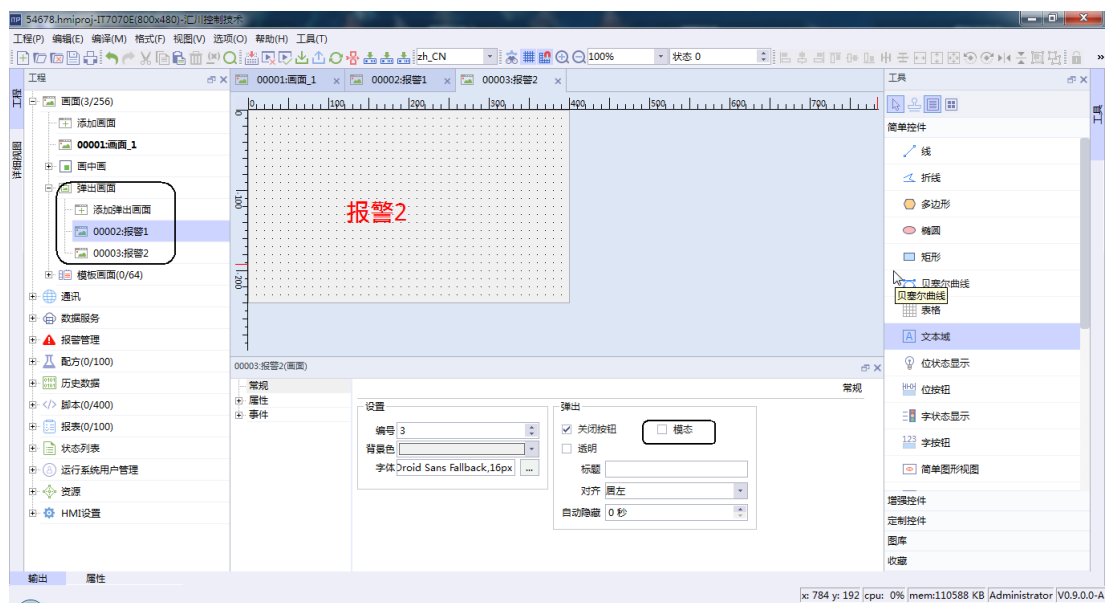
### 1. 弹出画面的操作方法

- 在界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加一个变量组。单击左上角“+”号创建一个变量LW0。
- 单击变量，选择变量“事件”>“值为真”>“画面”>“ShowPopup”>“画面名”选择想要弹出的画面即可。



### 2. 报警窗口弹出

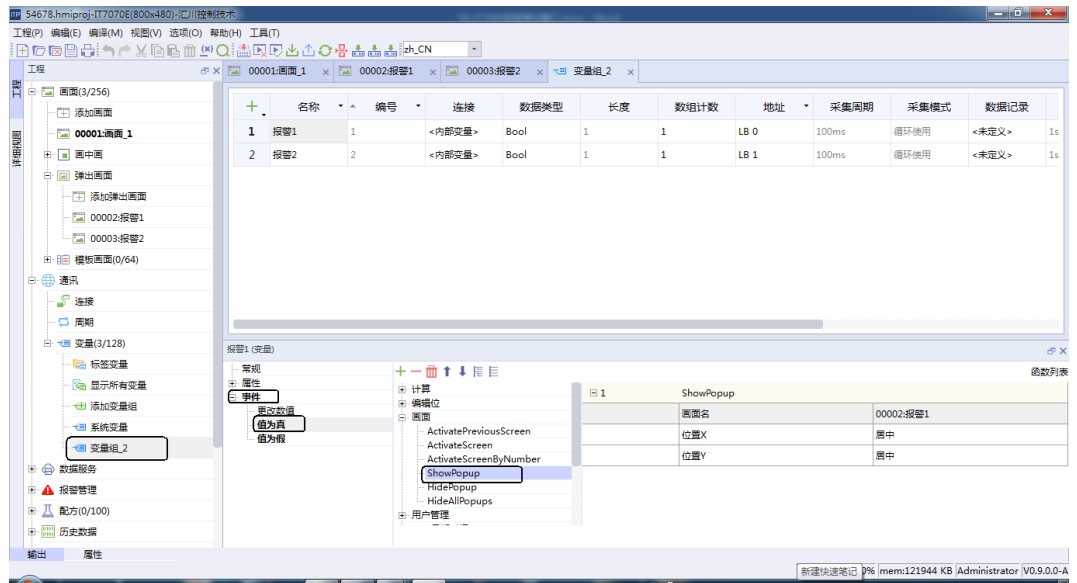
- 新建多个弹出窗口，设置报警文本，且弹出设置为非模态（不勾选模态）。



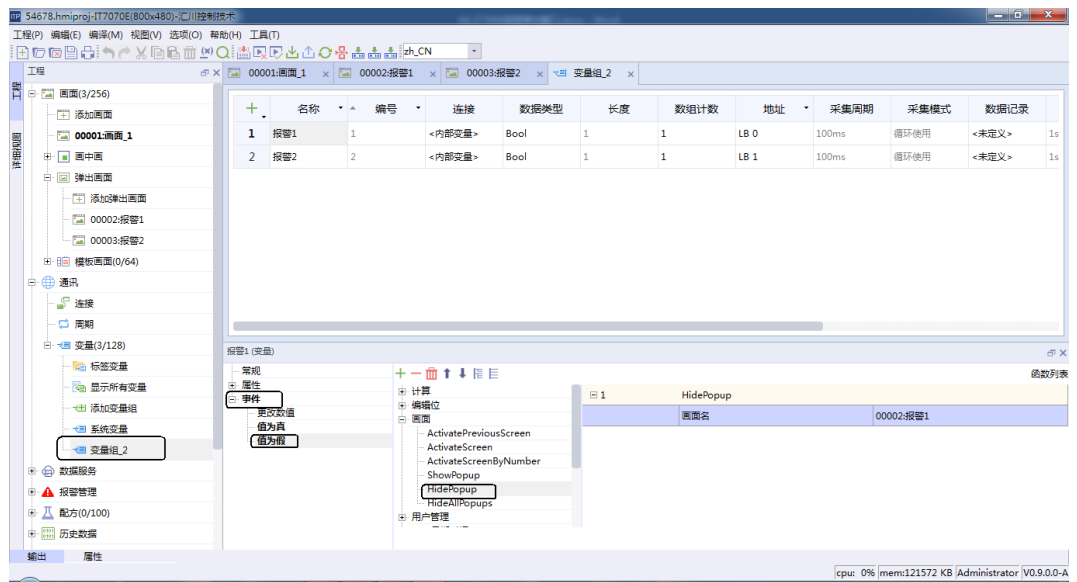
- 单击变量，选择变量“事件”>“值为真”>“画面”>“ShowPopup”>“画面名”选择想要弹出的画面即可。

- 值为真时弹出相应窗口。





- 值为假时关闭相关窗口。



3. 进行离线模拟，当触发报警时通过ShowPopup弹出报警弹窗验证效果。



## 说明

模态与非模态画面的区别：

- 模态画面
  - 激活窗口只有当前弹出窗口。
  - 只能存在一个模态窗口。
  - 除了当前模态弹出窗口外，其他界面的控件无法操作。
- 非模态画面
  - 可以同时弹出多个非模态画面。
  - 弹出非模态弹出窗口时，其他界面的控件仍然可以操作。

## 3.3 实现实时报警和历史报警

### 案例要求

通过案例实现实时报警和历史报警的功能。

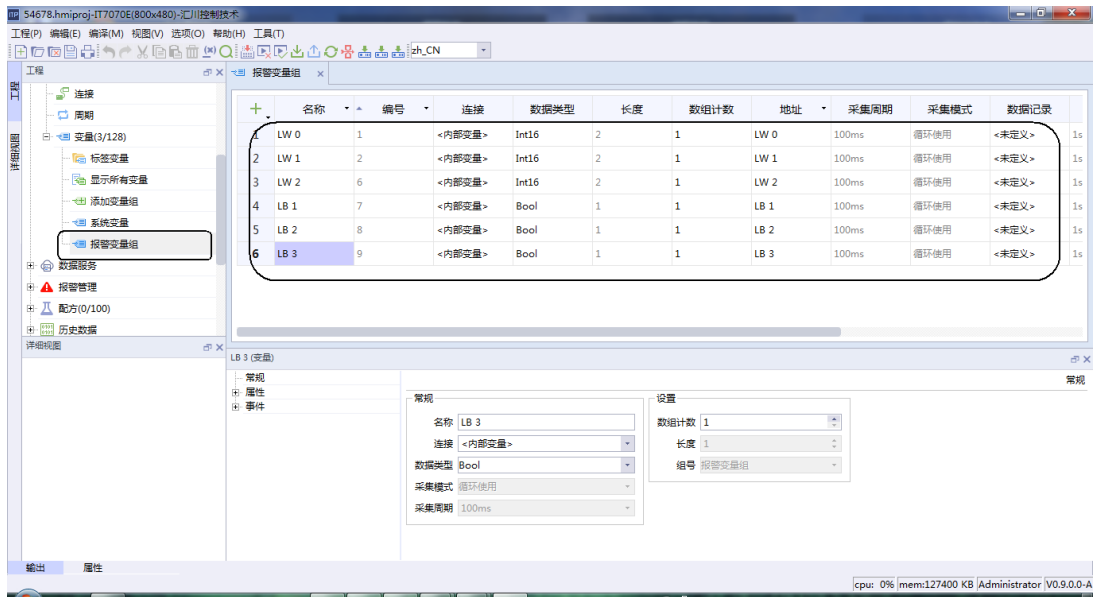
- 了解HMI报警的相关概念与区别。
- 可以完成实时报警功能。
- 可以完成历史报警显示功能。

### 案例目的

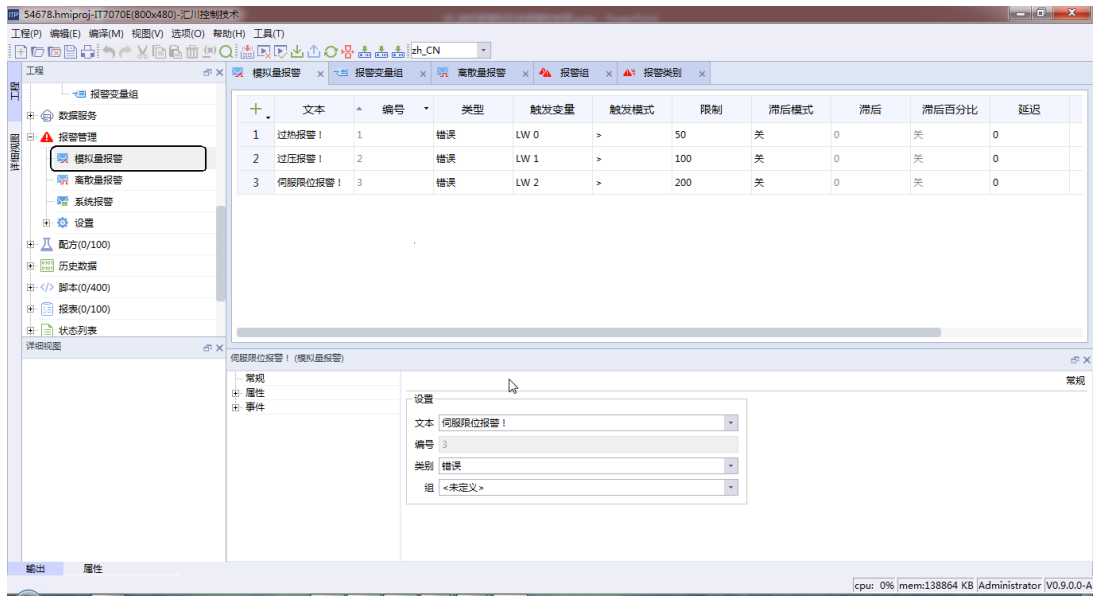
- 通过此案例学会报警的相关概念。
- 通过此案例学会实时报警的实现使用方法。
- 通过此案例学会历史报警的实现使用方法。

### 编程步骤

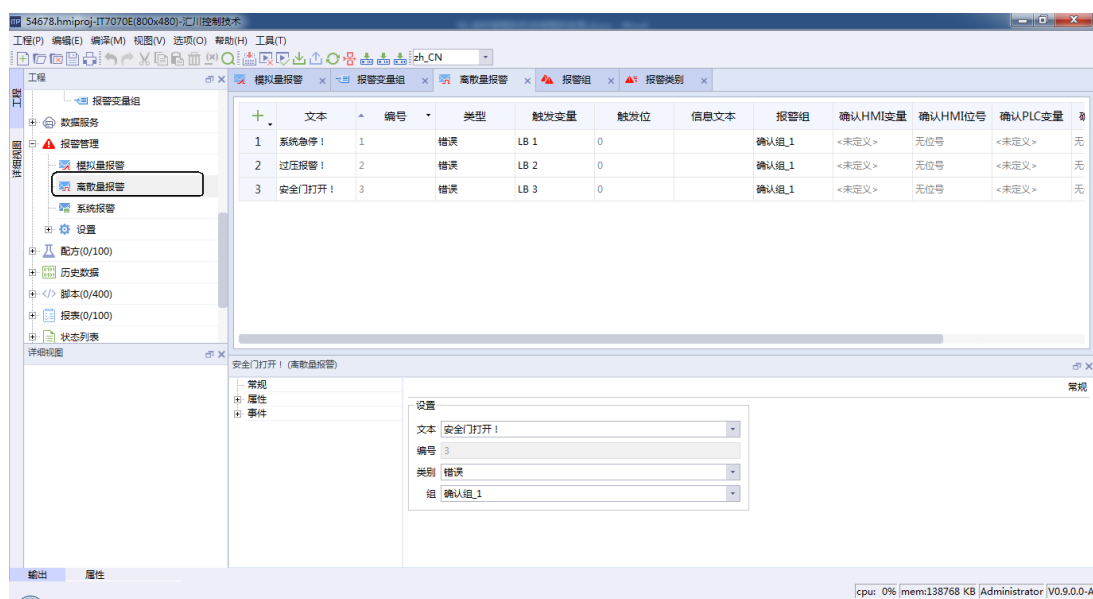
- 实时报警的实现方法
  1. 创建报警相关变量组：点击“通讯”选择“变量”之后点击“添加变量组”，对新添加的变量组进行命名。



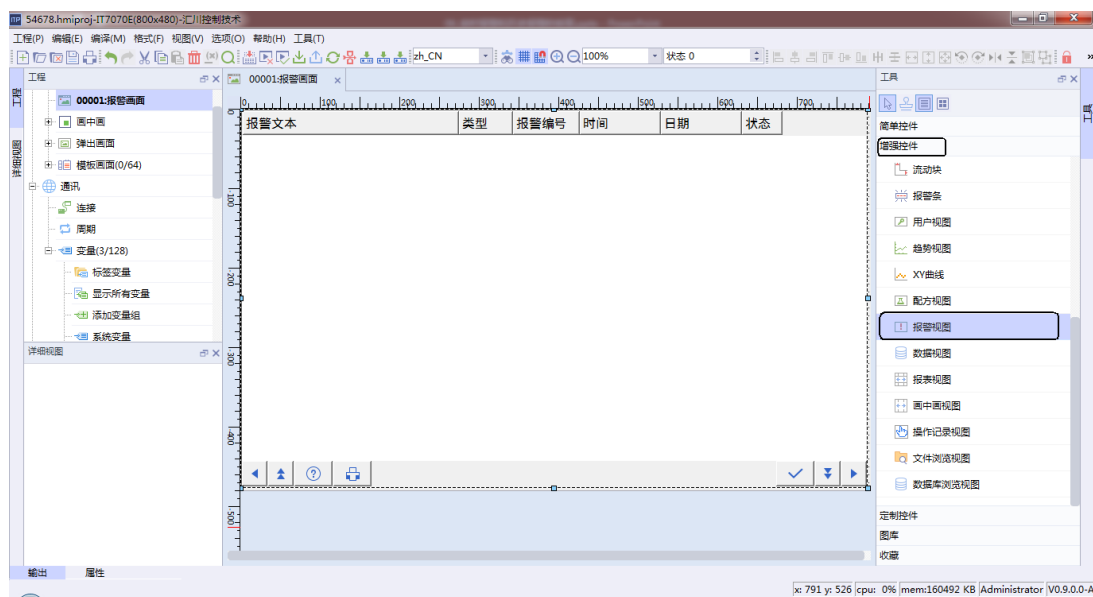
2. 创建模拟量报警：点击“报警管理”选择“模拟量报警”，在模拟量页面内添加相应变量以及需要的文本说明。



3. 创建离散量报警：点击“报警管理”选择“离散量报警”，在离散量页面内添加相应变量以及需要的文本说明。

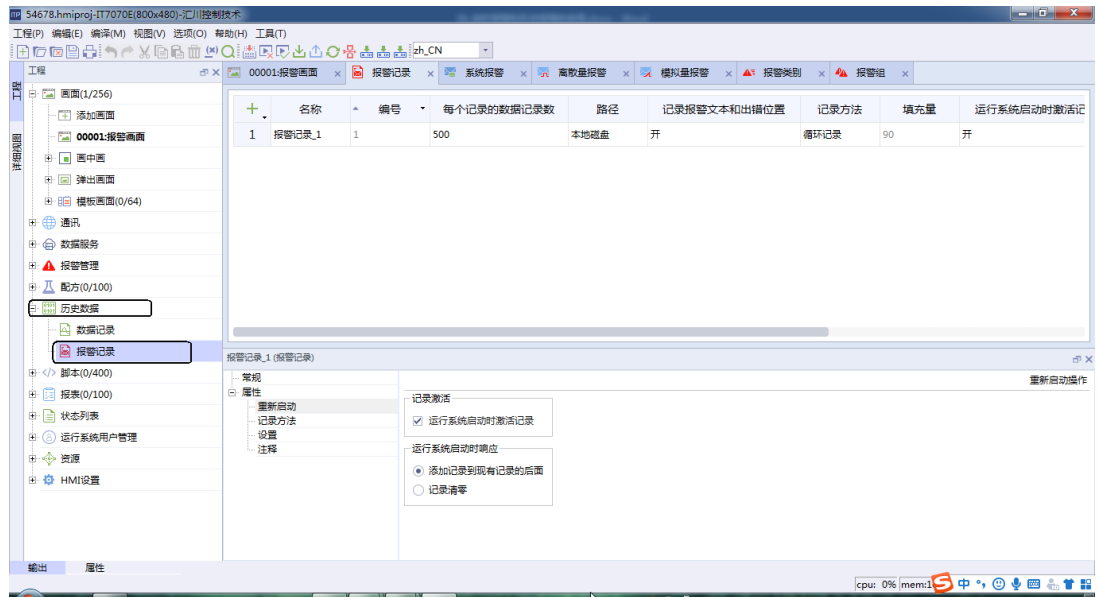


4. 报警页面的控件的添加：在“工具栏”内点击“增强控件”，找到“报警视图”长按鼠标左键拖拽至相应编辑画面内即可。



#### ● 历史报警的实现方法

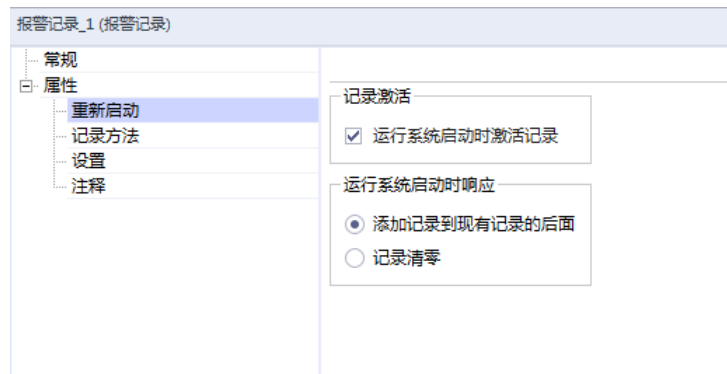
1. 创建报警变量（同实时报警）。
2. 创建模拟量报警，创建离散量报警（同实时报警）。
3. 在历史数据中创建报警记录：选择“历史数据”，点击“报警记录”对报警记录完成新建，对报警记录页面的“名称”、“数据记录数”、“注释”等相应信息进行编辑。



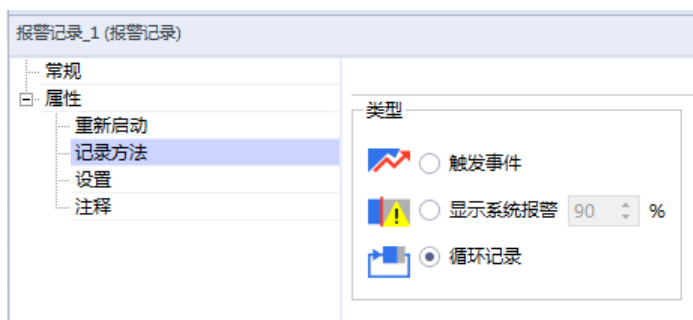
## 名词解释:



- 名称：报警记录的名称。
- 填充量：记录方法选择系统报警时有效，指定最大数量的百分比。
- 数据记录数：每个报警记录，允许记录的最大报警条数，存储位置不一样记录最大报警数的限制也有区别，HMI：最大1万条 U盘，SD卡：最大500万条。
- 路径：报警记录存储的位置，可以是存储在 HMI 上或者是的 U 盘、SD 卡上。

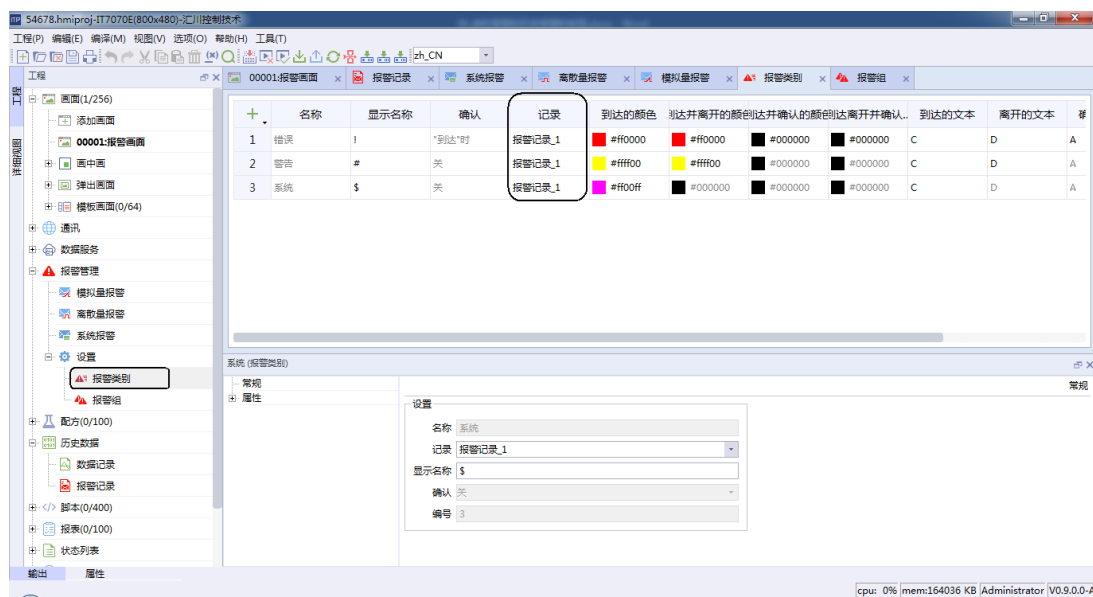


- 运行系统启动时激活记录：是否在 HMI 启动时就开启报警记录功能，若选择不启动则需要工程中组态系统函数 (StartLogging) 开启报警记录功能。
- 运行系统启动时响应: HMI启动时对报警记录的处理。
- 添加数据到现有记录的后面: HMI 启动时保留之前的报警记录条目。
- 记录清零: HMI 启动时，清空之前的报警记录条目。

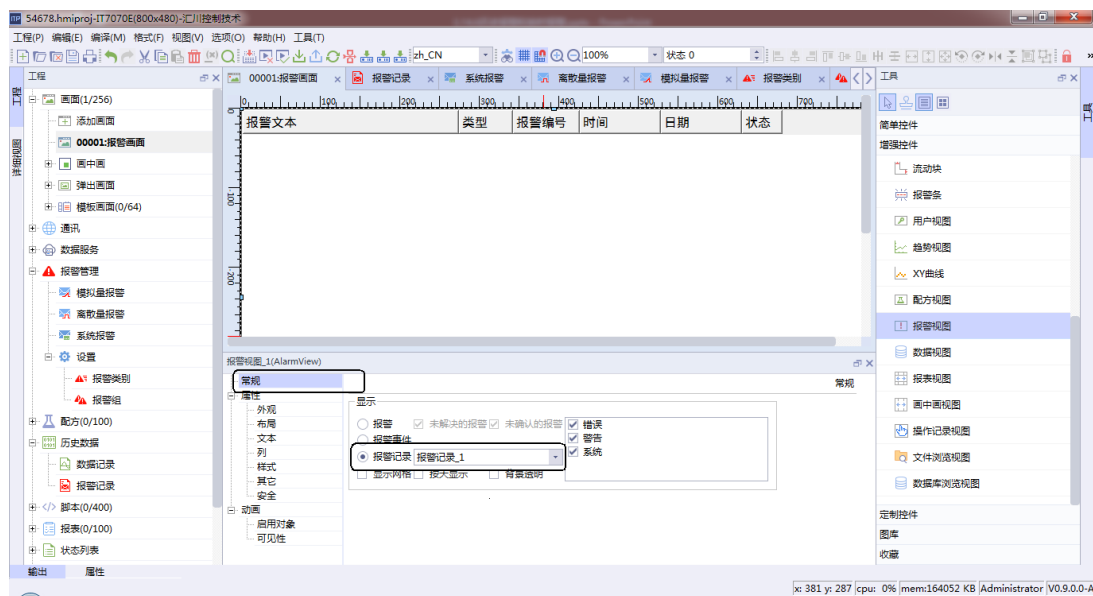


- 记录方法：报警记录记录满时HMI的处理方法。
- 循环记录：当记录完全填满时，最早的记录将被覆盖。
- 触发事件：当记录完全填满时，触发溢出事件中组态的脚本函数。
- 系统报警：当记录报警条数达到指定最大数量的百分比时，显示系统报警。

4. 对“报警类别”与“报警记录”进行数据绑定；选择“设置”，点击“报警类别”，在“记录”中勾选相应的报警记录进行绑定，在触发相应类别的报警时即将报警信息记录至指定报警记录中。



5. 勾选“报警记录”，点击“报警记录”在“显示”中勾选“报警记录”选择对应的记录即可。



### 说明

报警的相关概念：

- 模拟量报警：如果某一个“变量”值超出了“限制”值，HMI设备就触发报警。
  - 离散量报警：如果某一个“变量”中指定的位被置位，HMI设备就触发报警。
  - 报警类别：报警类别决定报警是否需要确认，也决定报警在HMI上的显示方式，还可以决定相应报警是否记录。
  - 报警组：统一确认处理。如果报警属于某个报警组，可以通过确认这个组中的某一个报警，完成这个组所有的报警的处理。
  - 报警记录：保存项目运行过程中产生的报警信息，便于后续分析问题，保存信息包括：报警文本、报警类别、报警编号、报警日期和时间、报警状态、确认组等。
-

## 4 通信

### 4.1 配置通讯参数

#### 案例要求

- 了解通讯参数配置。
- 明确各项通讯参数的意义并根据需求调整。

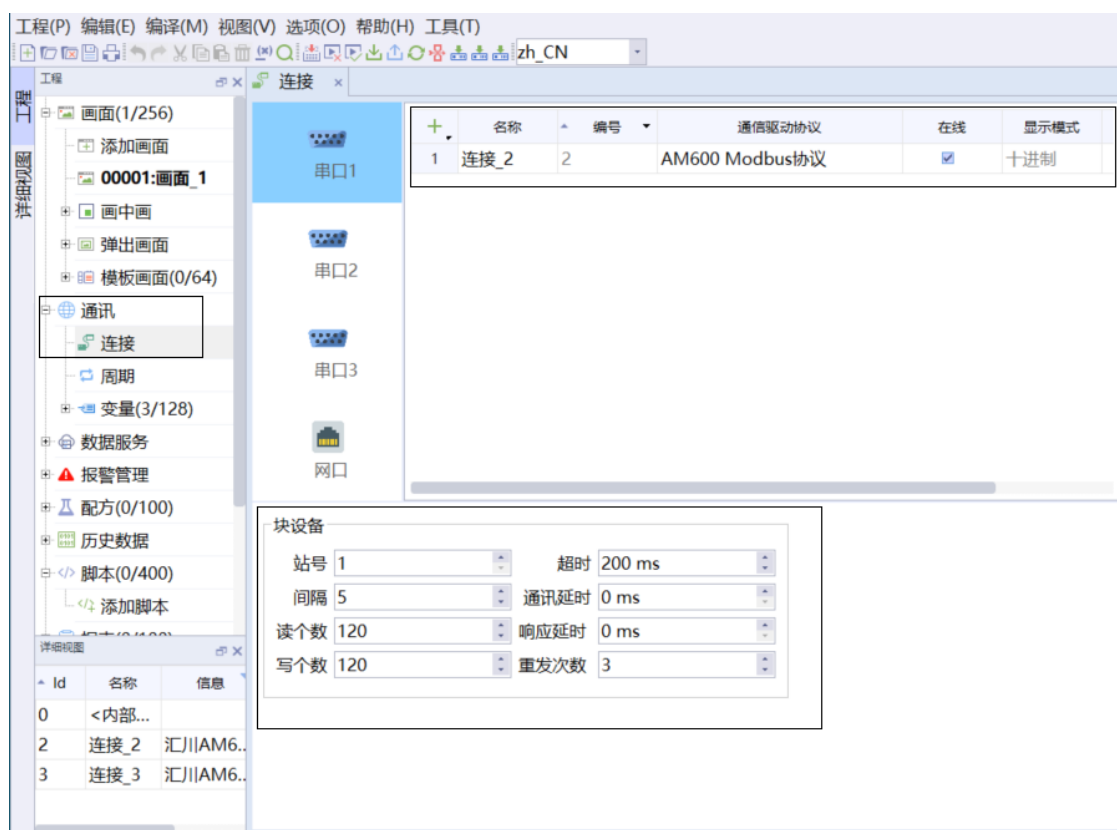
#### 案例目的

通过此案例掌握通讯参数配置过程。

#### 编程步骤

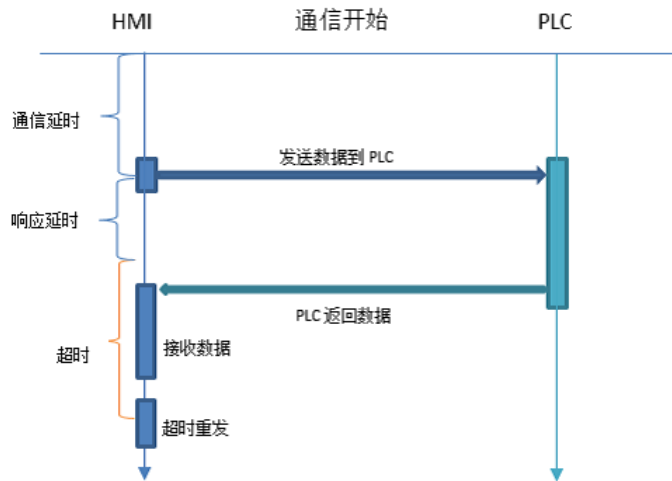
在工程中添加连接并配置通讯参数。

1. 在左侧工程中点击通讯连接串口1，添加新连接，这里以AM600 MODBUS协议为例，添加之后可以看到下方的站号，间隔，读个数，写个数，超时，通讯延时，响应时和重发次数。

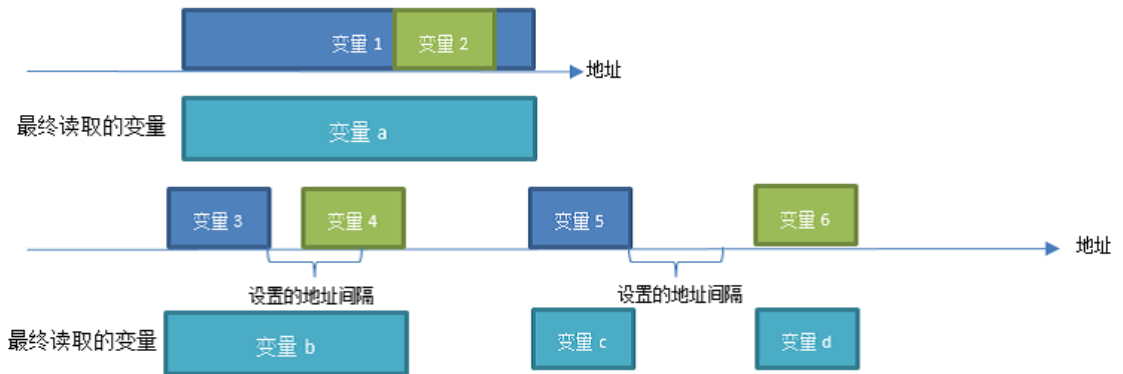


2. 超时、通讯延时、响应延时、重发次数用于设置通信过程。通信开始时，先进行通信延时，之后在发送数据到PLC；再进行响应延时，如果在响应延时后在超时时间内接收到数据，则通信成功；否则超时后进行重发，重新开始通信，如果进行重发了设置的次数还是没有接收到数据则通信失败。





3. 地址间隔、最多读取、最多写入用于设置通信数据内容。地址间隔设置变量之间是否合成一个数据帧，最多写入、最多读取设置读写的最大字数。



## 4.2 启动、停止通讯从站

### 案例要求

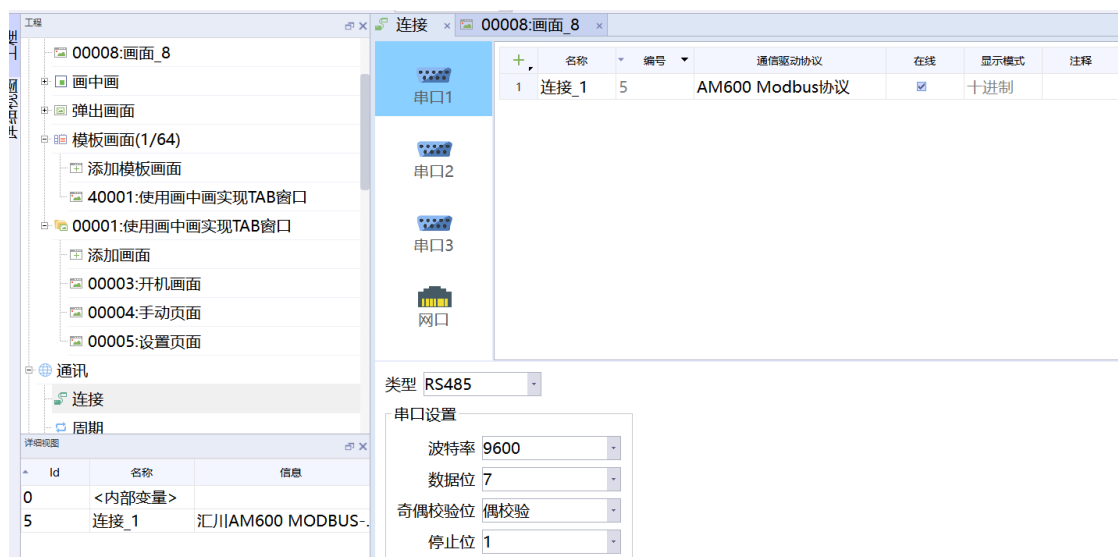
1. 学会如何在线启动、停止通讯从站。
2. 学会在线启停切换通讯从站状态。

### 案例目的

通过此案例学会在线启停切换通讯从站状态。

### 编程步骤

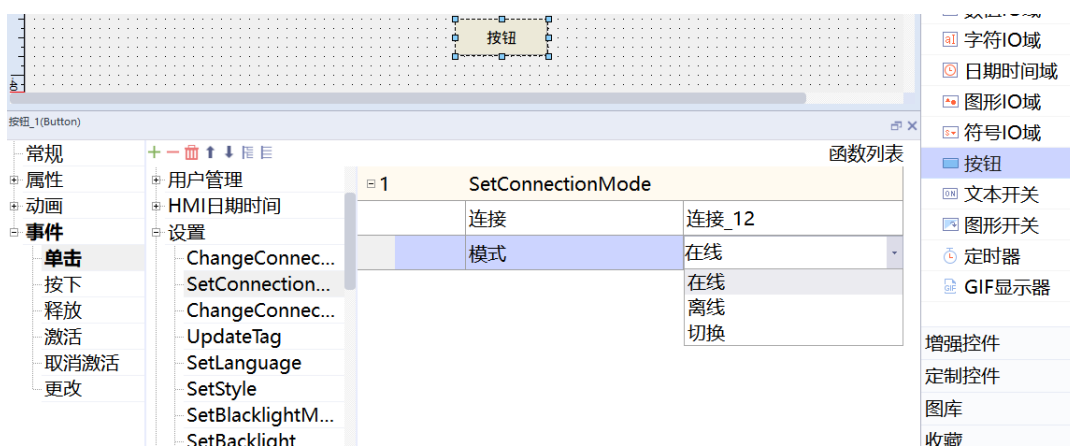
1. 新建一个连接  
在界面左侧“工程” > “通讯” > “连接”中点击“+”号，这里以串口1为例，选择AM600 MODBUS协议。



## 2. 画面中添加切换从站通讯状态的按钮

左侧“工程”>“画面”>“添加画面”，在画面中添加一个按钮，在按钮下方“事件”>“单击”>“设置”中，选择SetConnectionMode(用于建立或断开hmi设备与plc之间的连接)，然后在连接中选择连接\_12，从站状态可以在模式里进行选择，其中：

- “在线”模式表示单击按钮对应的从站长时间在线。
- “离线”模式表示单击按钮对应的从站长时间离线。
- “切换”模式表示通过单击按钮对应的从站可以在“离线”和“在线”之间来回切换。



## 4.3 在线修改通讯参数

### 案例要求

- 学会如何在线修改通讯参数。
- 学会在线修改485通讯参数。
- 学会在线修改以太网通讯参数。

### 案例目的

- 通过此案例学会在线修改485通讯参数。
- 通过此案例学会在线修改以太网通讯参数。

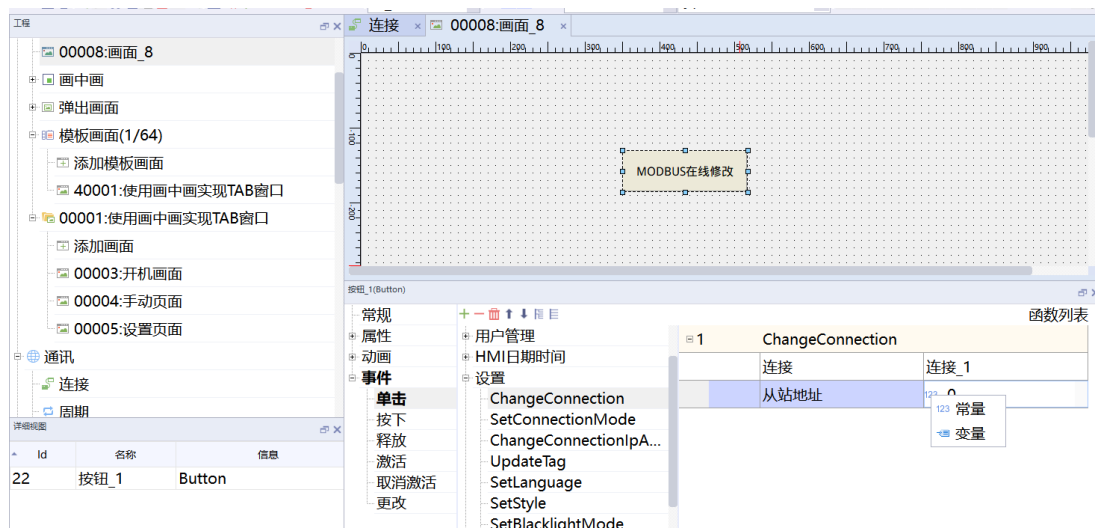
## 编程步骤

- 事件修改485通讯参数(更改从站地址)

1. 在界面左侧“工程”>“通讯”>“连接”中点击“+”号新建连接\_1, 这里以串口1为例, 即在串口1中 添加连接\_1, 这里以AM600 MODBUS协议为例。



2. 在画面中, 添加一个按钮, 按钮下方的事件单击设置中, 选择Changeconnection(更改指定连接中的从站地址), 然后在连接中选择连接1, 从站地址处可以选择常量或者变量, 即可修改从站地址通讯参数。

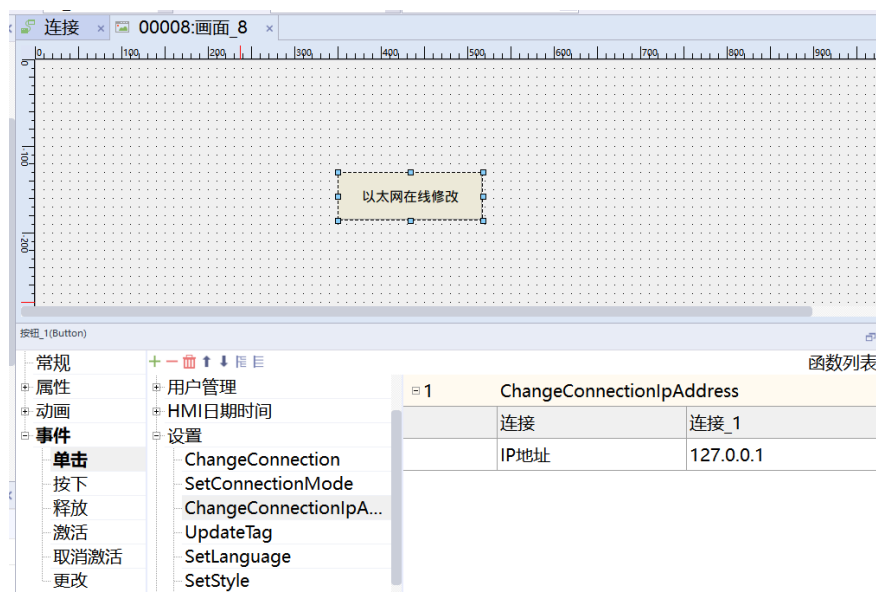


- 事件修改以太网通讯参数(更改从站IP)

1. 在界面左侧“工程”>“通讯”>“连接”中点击“+”号新建连接\_1, 在网口画面里添加连接, 这里以 AM600 MODBUS TCP协议为例。



- 在画面中，添加一个按钮，按钮下方的事件单击设置中，选择第三个选项ChangeConnectionAddress (更改连接的从站IP地址)，然后在连接中选择连接1，从站地址处可以选择常量或者变量，即可修改从站IP通讯参数。



## 4.4 自由协议的使用

### 案例要求

实现IT7000用串口走以太网自由协议通讯。

### 案例目的

掌握IT7000用串口走以太网自由协议通讯。

### 编程步骤

- 新建IT7000工程，在界面左侧“工程”>“通讯”>“连接”中点击“+”号新建连接，“通讯驱动协议”选择自由协议通信连接并且配置好相关的通信参数，如下所示。

- 串口1 COM1

名称	编号	通信驱动协议	在线	显示模式	注释
COM1_485	8	串口自由协议	<input checked="" type="checkbox"/>	十进制	

块设备

站号 0 : 超时 200 ms

间隔 0 : 通讯延时 0 ms

读个数 0 : 响应延时 0 ms

写个数 0 : 重发次数 0

- 串口2 COM2

名称	编号	通信驱动协议	在线	显示模式	注释
COM2_232	9	串口自由协议	<input checked="" type="checkbox"/>	十进制	

块设备

站号 0 : 超时 200 ms

间隔 0 : 通讯延时 0 ms

读个数 0 : 响应延时 0 ms

写个数 0 : 重发次数 0

- 串口3 COM3

名称	编号	通信驱动协议	在线	显示模式	注释
COM3_485	10	串口自由协议	<input checked="" type="checkbox"/>	十进制	

块设备

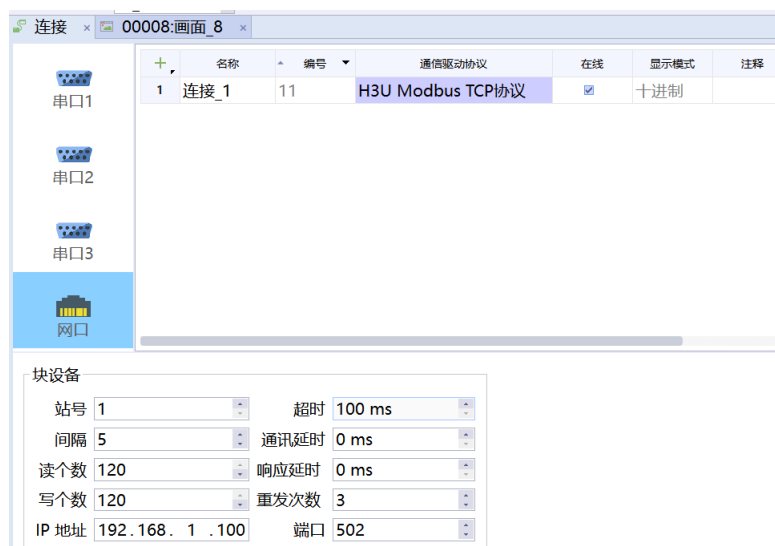
站号 0 : 超时 200 ms

间隔 0 : 通讯延时 0 ms

读个数 0 : 响应延时 0 ms

写个数 0 : 重发次数 0

- 以太网口

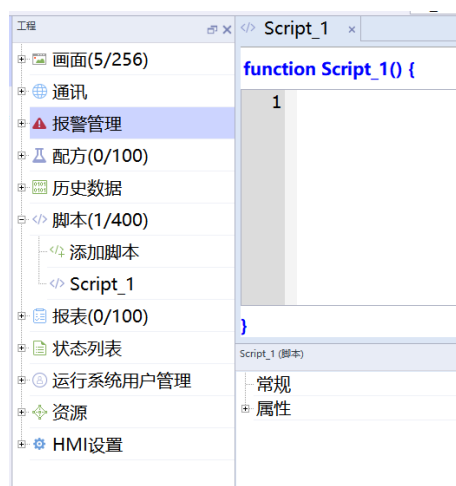


2. 在变量组中新建用于缓存发送和接收数据的变量。

+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期
1	LW 0	11	<内部变量>	Int16	2	1	LW 0	100ms
2	LW 1	12	<内部变量>	Int16	2	1	LW 1	100ms
3	LW 2	13	<内部变量>	Int16	2	1	LW 2	100ms
4	LW 3	14	<内部变量>	Int16	2	1	LW 3	100ms
5	D 0	15	连接_1	Int16	2	1	D 0	100ms
6	D 1	16	连接_1	Int16	2	1	D 1	100ms
7	D 2	17	连接_1	Int16	2	1	D 2	100ms
8	D 3	18	连接_1	Int16	2	1	D 3	100ms
9	LW 100	19	<内部变量>	Int16	2	1	LW 100	100ms
10	LW 101	20	<内部变量>	Int16	2	1	LW 101	100ms
11	LW 102	21	<内部变量>	Int16	2	1	LW 102	100ms
12	LW 103	22	<内部变量>	Int16	2	1	LW 103	100ms
13	D 100	23	连接_1	Int16	2	1	D 100	100ms
14	D 101	24	连接_1	Int16	2	1	D 101	100ms
15	D 102	25	连接_1	Int16	2	1	D 102	100ms
16	D 103	26	连接_1	Int16	2	1	D 103	100ms

3. 编写发送和接收数据的脚本程序。

在界面左侧“工程”>“脚本”>“添加脚本”即可进行脚本编写。



脚本源码如下：

```
function timer()
```

```
{  
//数据发送  
var a = new Array();  
a[0] = SmartTags('LW 0');  
a[1] = SmartTags('LW 1');  
a[2] = SmartTags('LW 2');  
a[3] = SmartTags('LW 3');  
FreeOut('COM2_232', a);  
var b = new Array();  
b[0] = a[0];  
b[1] = a[1];  
b[2] = a[2];  
b[3] = a[3];  
FreeOut('COM1_485', b);  
//数据接收  
var c = new Array();  
c = FreeIn('COM2_232');  
SmartTags('D 0') = c[0];  
SmartTags('D 1') = c[1];  
SmartTags('D 2') = c[2];  
SmartTags('D 3') = c[3];  
var d = new Array();  
d = FreeIn('COM1_485');  
SmartTags('D 100') = d[0];  
SmartTags('D 101') = d[1];  
SmartTags('D 102') = d[2];  
SmartTags('D 103') = d[3];  
}  
SetInterval(timer,200);//定时器，200ms 调用一次timer 函数
```

如下图所示：

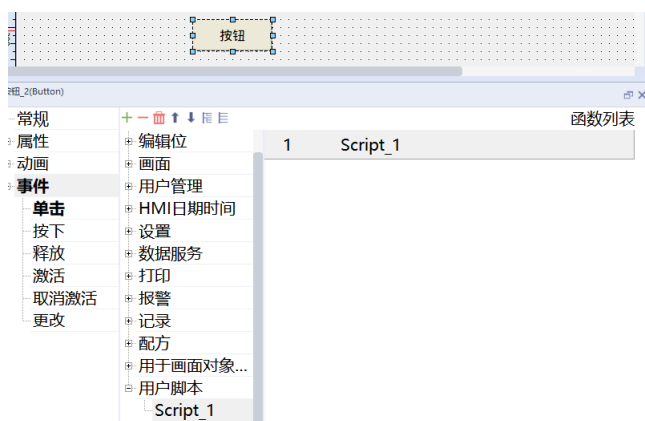
```

function Script_10 {
1  function timer()
2  {
3      //数据发送
4      var a = new Array();
5      a[0] = SmartTags('LW 0');
6      a[1] = SmartTags('LW 1');
7      a[2] = SmartTags('LW 2');
8      a[3] = SmartTags('LW 3');
9      FreeOut('COM2_232', a);
10     var b = new Array();
11     b[0] = a[0];
12     b[1] = a[1];
13     b[2] = a[2];
14     b[3] = a[3];
15     FreeOut('COM1_485', b);
16     //数据接收
17     var c = new Array();
18     c = FreeIn('COM2_232');
19     SmartTags('D 0') = c[0];
20     SmartTags('D 1') = c[1];
21     SmartTags('D 2') = c[2];
22     SmartTags('D 3') = c[3];
23     var d = new Array();
24     d = FreeIn('COM1_485');
25     SmartTags('D 100') = d[0];
26     SmartTags('D 101') = d[1];
27     SmartTags('D 102') = d[2];
28     SmartTags('D 103') = d[3];
29 }
30 SetInterval(timer,200);//定时器,200ms 调用一次 timer 函数
31
}

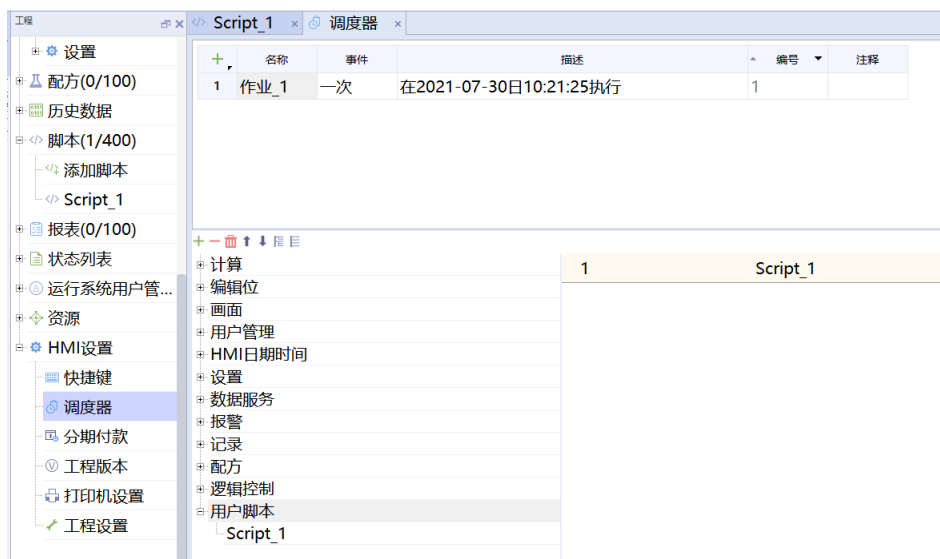
```

行: 30, 列: 4

4. 单击按钮，在“事件” > “单击” > “用户脚本”调用脚本程序或者通过调度器定时调用脚本程序，如下图所示。







## 说明

- 自由协议可以使用 JavaScript 自带的数据类型，也可以调用其自带的对象方法和属性进行。各种数据类型的相互转换，用户可以查看 JavaScript 参考手册，以下网址可供用户参考 <https://www.w3school.com.cn/jsref/index.asp>。
- IT7000 的串口和以太网，每个硬件端口只能最大建立 32 个同类型连接，不同端口相互独立，但是多个端口同时进行数据采集会影响 CPU 负载率。
- IT7000 以太网自由协议只支持 TCP 不支持 UDP，而且是自动进行 TCP 连接，用户无手动去和从站建立连接和断开连接。

## 4.5 分期付款功能

IT7000分期付款功能说明：

在实际运用中，由于试用或者资金问题，常常需要分期付款。有时候设备厂家客户需要在线延期、同一工程维护多个客户、可以自己组态解锁画面等等，针对分期以及设备厂家客户的需求，对分期付款功能进行了扩充。具体功能包括如下：

- 同一个工程可以新建多个客户，达到一个工程对应多个设备买家的目的，目前最多支持新建10个客户。
- 分期方式可选。具体分为以下三种：
  - 静态密码分期方式。
  - 变量密码分期方式。
  - 汇川动态密码方式。
- 用户自定义解锁弹窗画面。
- 下载时是否保留原分期信息。
- 提前触发下一分期。
- 下一分期剩余时间（分钟）。
- 管理员密码解锁。

### 案例要求

通过此案例了解并学习分期付款功能。

## 案例目的

1. 掌握分期付款功能的组态过程。
2. 通过此案例学会如何使用分期付款功能运用到工程项目中。

## 编程步骤

### 1. 分期组态界面

在界面左侧“工程”>“HMI设置”>“分期付款”即可创建分期付款组态界面。



### 2. 静态密码组态分期

静态密码方式分期时间与密码固定，组态和运行状态一致。

#### a. 可更改客户名

双击客户名标签“tab\_1”，弹出输入客户名窗口，输入如张三即可修改名称。



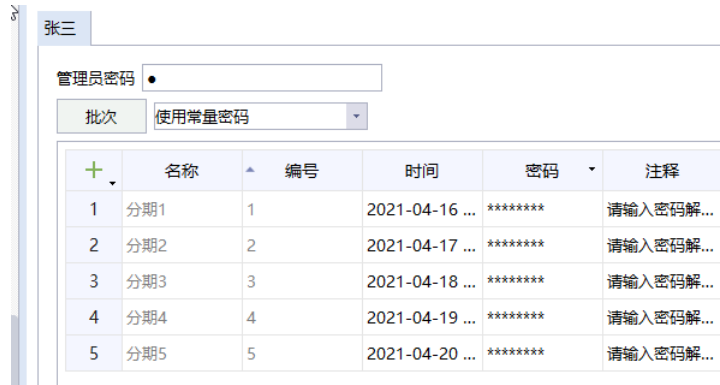
#### b. 使用常量密码

密码方式选择“使用常量密码”。



c. 新建分期信息

如下图所示，可通过“批次”按钮或者表格左上角的“+”按钮进行分期信息的新建。



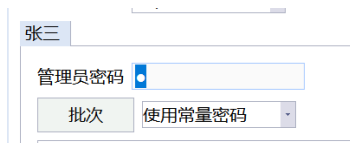
1). 分期到达时间的调整可通过双击某一分期的时间进行。

+	名称	编号	时间	密码	注释
1	分期1	1	2021-07-29 00:00:00	*****	请输入密码解锁系统
2	分期2	2	2021-07-30 00:00:00	*****	请输入密码解锁系统
3	分期3	3	2021-07-31 00:00:00	*****	请输入密码解锁系统

2). 分期密码的修改可通过双击某一分期的密码进行。

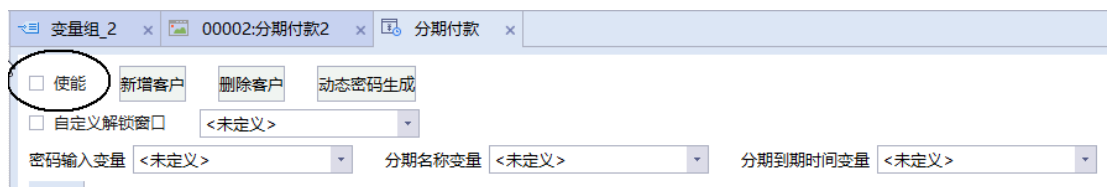
+	名称	编号	时间	密码	注释
1	分期1	1	2021-07-29 00:00:00	*****	请输入密码解锁系统
2	分期2	2	2021-07-30 00:00:00	ia7vx6iwitu0q0qr	请输入密码解锁系统
3	分期3	3	2021-07-31 00:00:00	*****	请输入密码解锁系统

3). 可修改客户的管理员密码。



d. 使能

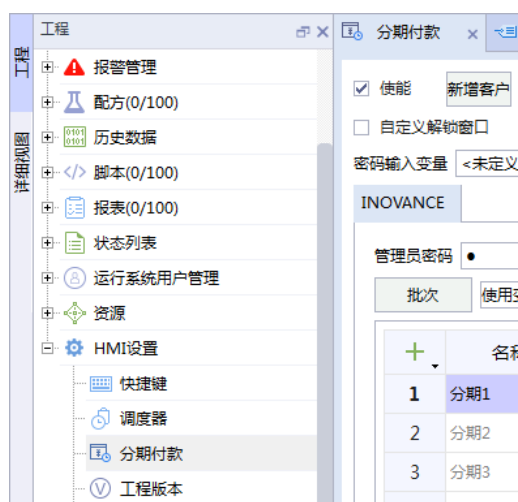
分期信息新建完毕后，勾选最顶端的使能选项，下载至HMI中即生效。



3. 变量密码组态分期

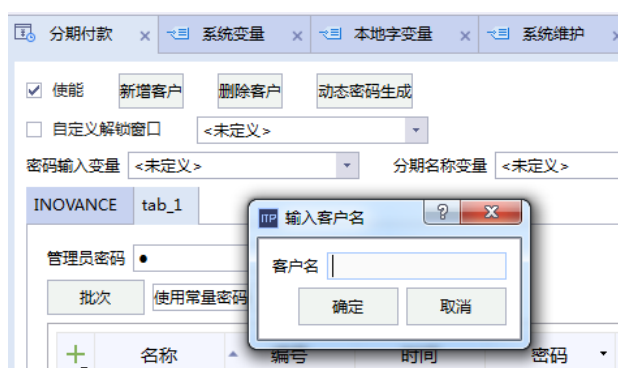
a. 分期付款功能使能

左侧项目树“工程”>“HMI设置”>“分期付款”双击后进入分期付款设置窗口，在“使能”选项前勾选，即可开启分期付款功能。



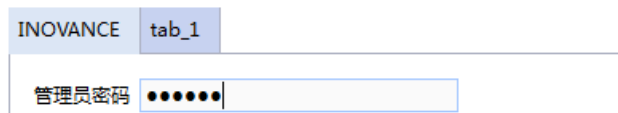
#### b. 分期付款功能的用户添加

在分期付款设置窗口可通过“新增用户”、“删除用户”实现客户添加和删除，最多添加10个终端客户。添加新客户后，可通过鼠标左键双击用户名标签，实现用户名称的修改。



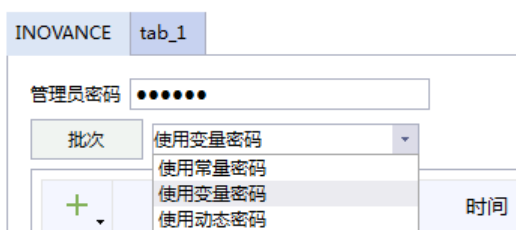
#### c. 管理员密码的添加

在分期付款窗口添加用户后，可通过管理员密码设置超级权限密码，该密码主要用于提前解除客户所有分期付款或用于客户忘记设定分期密码后另外一种解锁的方法（超级密码：654321）。



#### d. 分期方式

分为静态密码分期方式、变量密码分期方式、汇川动态密码方式，这里选择变量密码分期方式。



#### e. 使用变量密码的添加

1). 添加类型变量：左侧项目树“工程”>“通讯”>“变量”>“添加变量组”添加一个变量组添加12个DateTime数据类型变量用于设定分期到期时间，添加12个String数据类型变量，用于设定分期密码，当前工程分期密码预设为分期1:111111、分期2:222222、分期3:333333以此类推。如下图。

+	名称	编号	连接	数据类型	长度	数组计数	地址	初始值	索引变量
11	分期11时间	11	<内部变量>	DateTime	8	1	RW 65040	2050-03-15 0...	<未定义>
12	分期12时间	12	<内部变量>	DateTime	8	1	RW 65044	2050-03-15 0...	<未定义>
13	分期1密码	13	<内部变量>	String	10	1	RW 65100	111111	<未定义>
14	分期2密码	14	<内部变量>	String	10	1	RW 65105	222222	<未定义>
15	分期3密码	15	<内部变量>	String	10	1	RW 65110	333333	<未定义>
16	分期4密码	16	<内部变量>	String	10	1	RW 65115	444444	<未定义>
17	分期5密码	17	<内部变量>	String	10	1	RW 65120	555555	<未定义>
18	分期6密码	18	<内部变量>	String	10	1	RW 65125	666666	<未定义>
19	分期7密码	19	<内部变量>	String	10	1	RW 65130	777777	<未定义>
20	分期8密码	20	<内部变量>	String	10	1	RW 65135	888888	<未定义>
21	分期9密码	21	<内部变量>	String	10	1	RW 65140	000000	<未定义>

2). 添加分期付款期数，分期付款都需要设置名称、时间、密码、注释。其中时间和密码关联已经添加的变量；注释用于到期提示用。

INOVANCE tab\_1

管理员密码

批次

+	名称	编号	时间	密码	注释
1	分期1	1	分期1时间	分期1密码	请输入系统维护密码
2	分期2	2	分期2时间	分期2密码	请输入系统维护密码
3	分期3	3	分期3时间	分期3密码	请输入系统维护密码
4	分期4	4	分期4时间		
5	分期5	5	分期5时间		
6	分期6	6	分期6时间		
7	分期7	7	分期7时间		
8	分期8	8	分期8时间		
9	分期9	9	分期9时间		
10	分期10	10	分期10时间		

	Id	名称	信息	数据类型
1	-1	<未定义>	<No address>	Int16
2	13	分期1密码	RW 65100	String
3	14	分期2密码	RW 65105	String
4	15	分期3密码	RW 65110	String
5	16	分期4密码	RW 65115	String

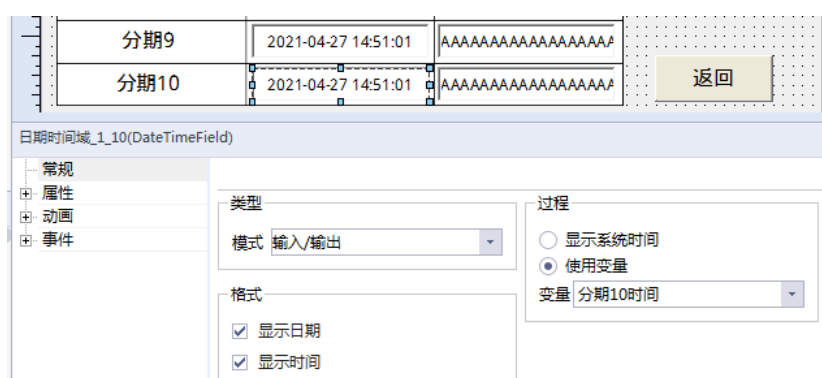
所有对象

f. 添加分期付款设置表

1). 新建一个画面，右侧工具栏“工具”>“简单控件”>“表格”添加表格，调整表格行数、列数。添加文本域添加所需文字，如下图所示。

分期	时间	密码
分期1		
分期2		
分期3		
分期4		
分期5		
分期6		
分期7		
分期8		
分期9		
分期10		

2). 右侧工具栏“工具”>“简单控件”>“日期时间域”/“字符IO域”添加控件，并关联时间和密码变量。



g. 分期付款功能的实现案例完成

1). 通过变量密码实现分期付款功能已完成，到期后显示画面。

请输入系统维护密码

分期1    结束日期 1970-01-01 08:00:00    密码

2). 输入密码为管理员密码是，会有窗口画面提醒是否清除所以分期密码，如下图。



#### 4. 动态密码组态分期

动态密码与使用变量密码类似，只需要组态时间变量即可，如下图。



动态密码到期后，会生成随机码，点击组态界面的“动态密码生成”按钮，将随机码输入到如下图的“请输入随机”码输入框中，点击“校对”按钮，此时只需选择“延时”或者“不延时”。

- 若延时，输入延时天数，延时天数限制1100天，选择完成后，点击“生成密码”按钮，生成动态密码，将生成的动态密码输入到解锁画面的密码输入框中。
- 若选择不延时，并输入了延时天数，则该期在输入的延长天数后再次锁住。达到在线延期的目的。



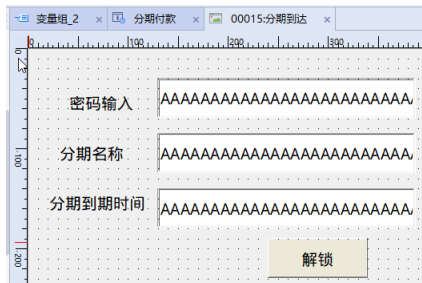
## 5. 自定义解锁画面（注意：自定义解锁画面对动态密码方式不生效）

当客户已不满足使用系统自带的分期达到画面，想自己组态解锁画面，同样支持。自定义解锁画面需要对两个地方进行组态。

- a. 若需要进行自定义解锁画面，首先勾选“自定义解锁窗口”选项，在界面左侧“工程”>“画面”>“弹出画面”>“添加弹出画面”创建一个弹出画面，该画面用于在分期到达时弹出，再分别选择“密码输入变量”、“分期名称变量”、“分期到期时间变量”，三者均为String类型，“密码输入变量”用于分期达到后获取输入的密码，“分期名称变量”用于分期到达后显示分期的名称，“分期到期时间变量”用于显示分期到达后显示分期的日期。

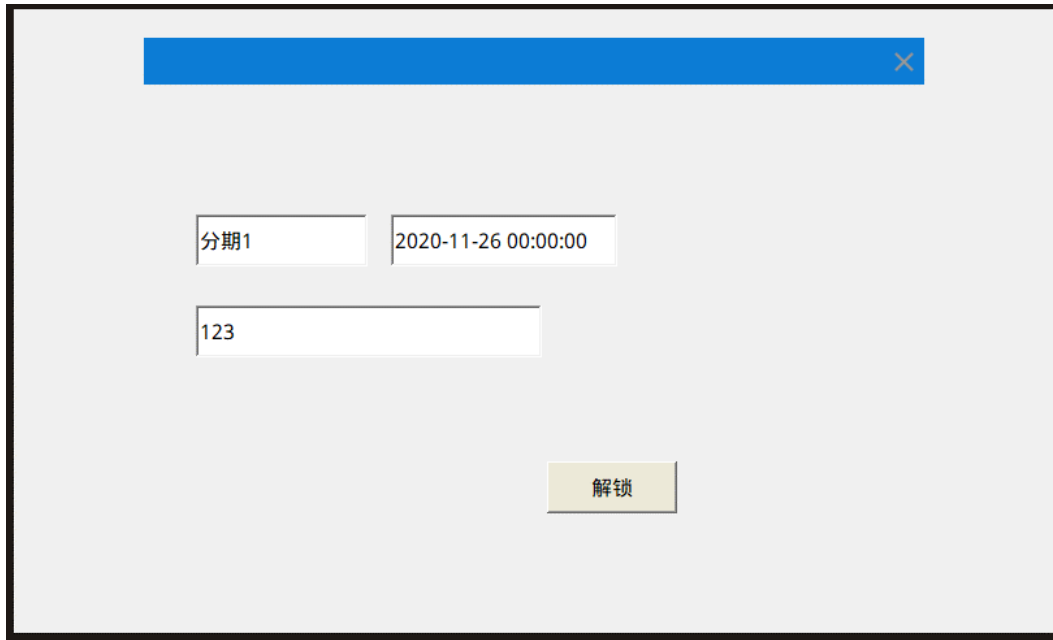


- b. 组态弹出画面，三个输入框对应“密码输入变量”、“分期名称变量”、“分期到期时间变量”框对，再加上一个按钮，按钮绑定UnlockCurrentInstalMent系统函数。弹出画面选择模态，关闭按钮可留可不留，不留则只能通过解锁按钮关闭，留则关闭后会继续弹出，除非解锁。



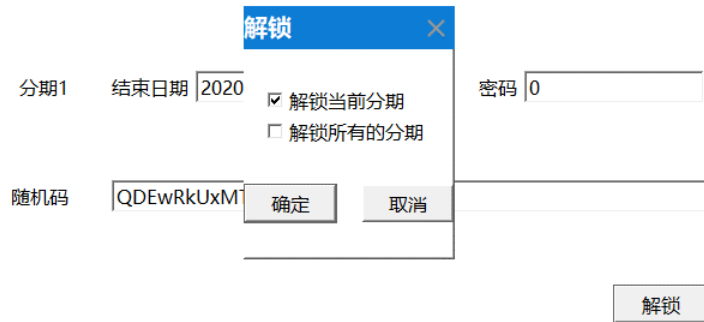
- c. 完成以上两处组态后，分期达到效果如图所示。





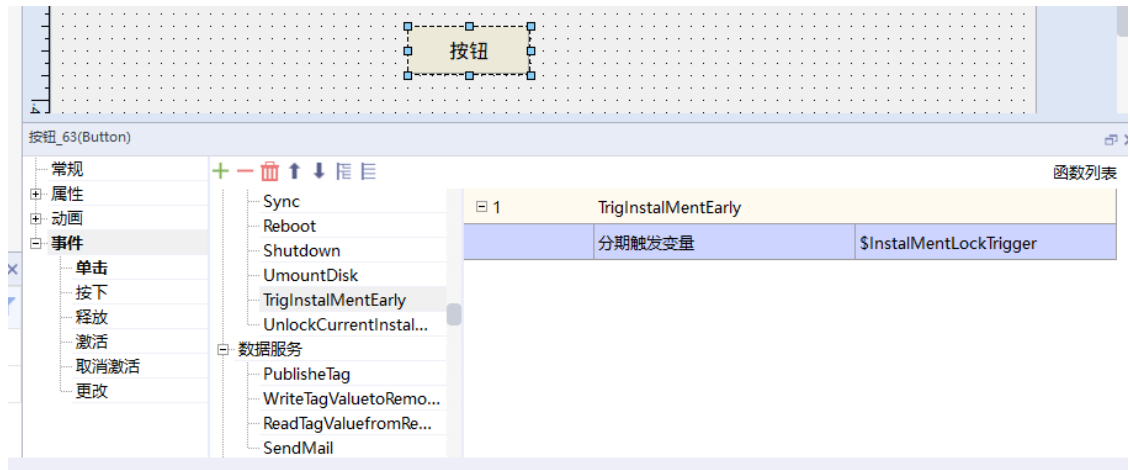
### 6. 管理员密码解锁

除了输入分期密码外，还可输入管理员密码进行解锁，输入管理员密码后，会弹出如图所示画面，可选择解锁当前分期或解锁所有分期。



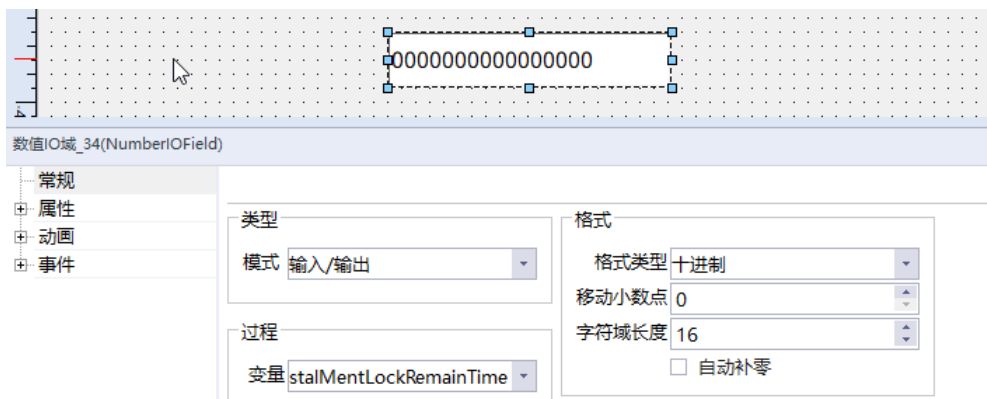
### 7. 提前触发分期

支持提前触发下一分期的功能，选择按钮事件单击设置绑定系统函数TrigInstalMentEarly，变量绑定系统变量InstalMentLockTrigger，若存在多期分期时，需要提前触发下一分期，则点击按钮，可触发下一分期。



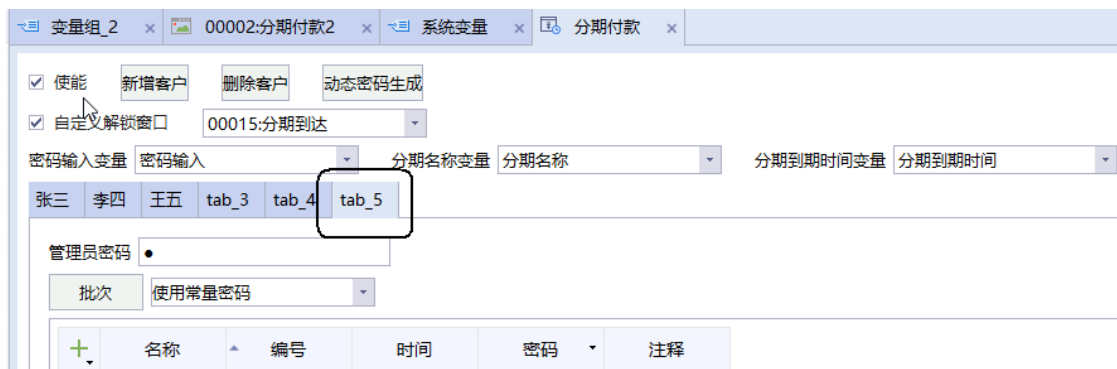
## 8. 下一分期剩余时间

如图所示，组态系统变量\$InstalMentLockRemainTime，则可显示下一分期剩余到达时间（单位：分钟）。



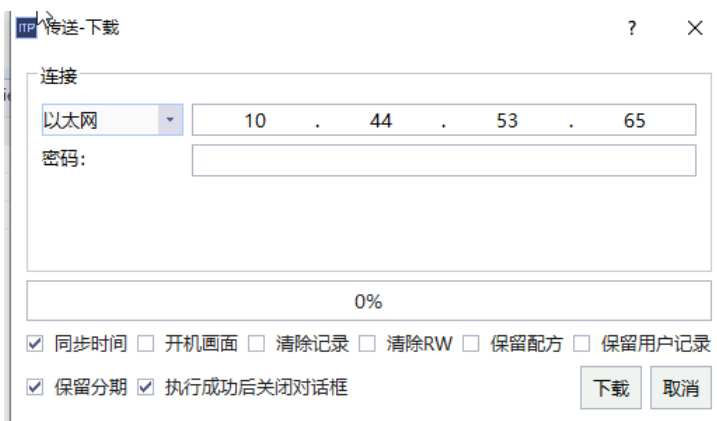
## 9. 多客户

可通过新增客户或者删除客户对多客户进行维护。界面组态时选择的客户即为下载至HMI中的分期。如图所示，此时生效的是tab\_5客户。



## 10. 保留分期

如图所示，在下载时，可决定是否保留上一次的分期，若保留，则分期信息仍为上一次的分期信息，若不保留，则分期信息为本次下载的组态分期信息。



## 4.6 通过HMI下载PLC程序—穿透功能

### 案例要求

穿透功能主要运用在现场PLC被安装在机箱内部，PC不便与PLC直接连接的场景，通过间接连接取代直接连接。

### 案例目的

穿透功能是专门针对汇川 H1U、H2U、H3U 系列 PLC 设计的，通过穿透功能可实现在 PC 与 PLC 未直接连接情况下，通过 PC 与 HMI 连接、HMI 与 PLC 连接的间接方式实现 PC 对 PLC 的操作。穿透功能主要运用在现场PLC被安装在机箱内部，PC不便与PLC直接连接的场景，通过间接连接取代直接连接。

### 编程步骤

1. 在 InoTouchPad 软件工具栏或菜单栏 “编译” 选项卡中找到穿透图标，然后点击 “穿透”。



2. 在弹出的 “穿透工具” 界面中，设置匹配的参数。
  - a. 虚拟串口:添加一个虚拟串口，用于连接汇川 PLC 后台软件 AutoShop。
  - b. HMI连接:设置HMI与PLC的连接类型，分别为USB和以太网，其中若选择以太网，IP地址应为HMI的地址。
  - c. PLC连接设置HMI与PLC连接的端口及端口信息，如现场用COM1,则端口选择COM1并匹配好其他参数。
3. 开始下载。
 

参数设置完毕后，点击开始。打开汇川小型PLC编程软件Autoshop，在Autoshop通讯设置中设置串口为COM4（与虚拟端口保持一致），即可在Autoshop中对HMI连接的PLC进行操作。

## 4.7 VNC功能应用

### 案例要求

利用可运行VNC Sever的设备，例如手机或者电脑，远程监控并操控HMI。

## 案例目的

将屏幕投射到运行 VNC Server 的设备上，比如手机或者电脑。使用 VNC 远程投屏可以方便查看 HMI 设备运行状况，远程操控 HMI 设备。

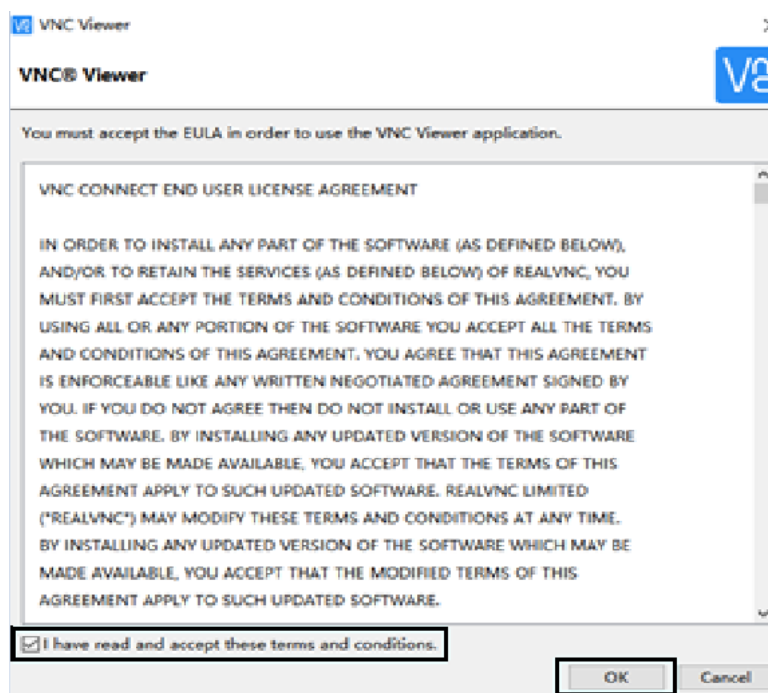
## 编程步骤

### 1. 在电脑桌面上运行VNC

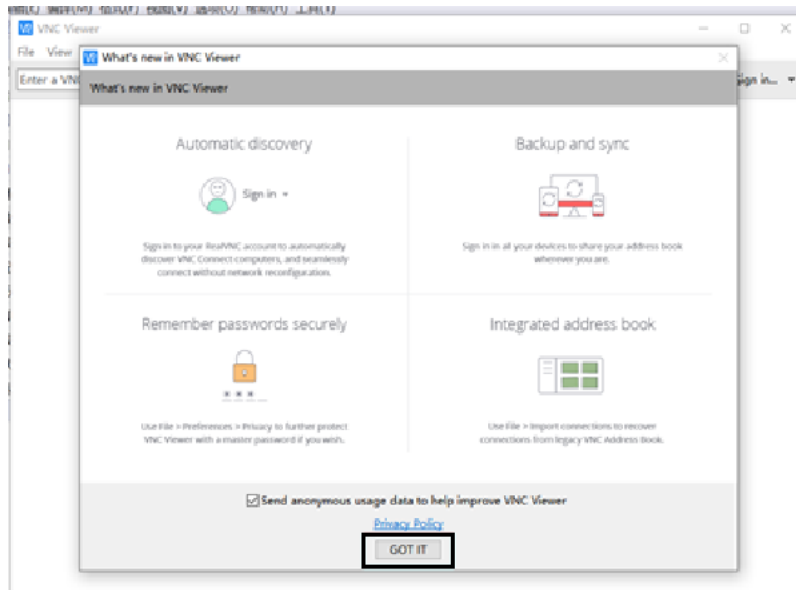
- a. IT7000 后台自带 VNC Server 端，可通过菜单栏中的“工具(T)” > “VNC”打开。



- b. 第一次打开 VNC 时需要确认协议，如下图所示，请勾选 “I have read and accept these terms and conditions”，然后点击 “OK” 按钮。

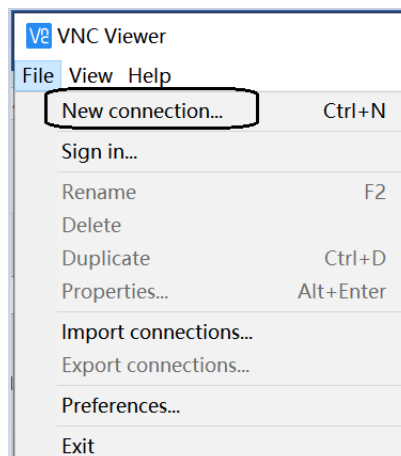


- c. 点击 “OK” 后进入以下页面，直接点击 “GOT IT” 即刻进入到主界面。

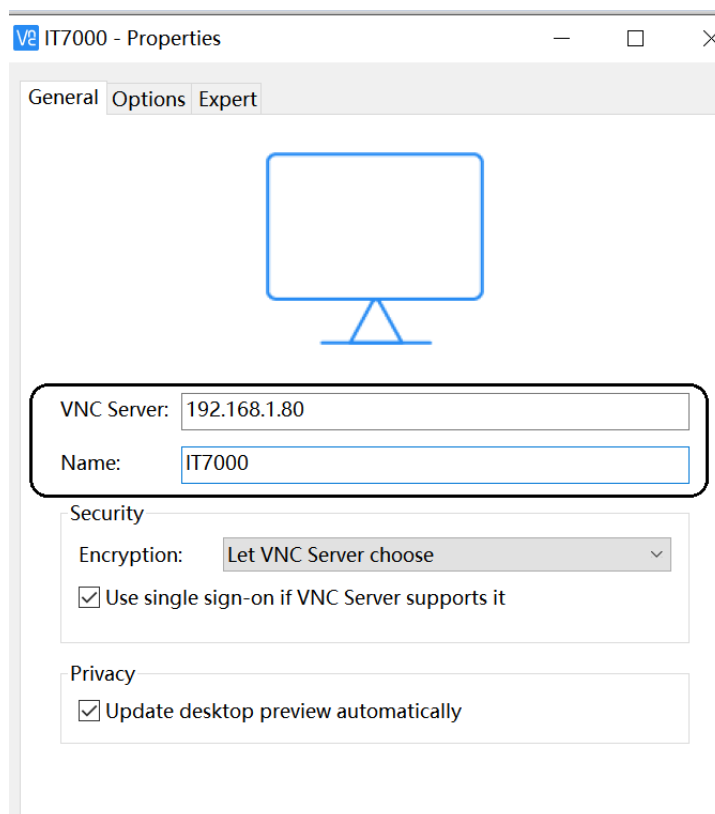


## 2. 新建VNC连接

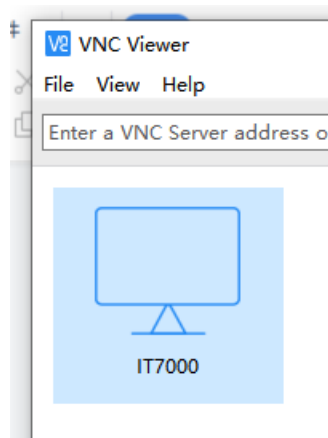
- a. 在主页面中点击菜单栏“File” “New connection...” 新建连接。



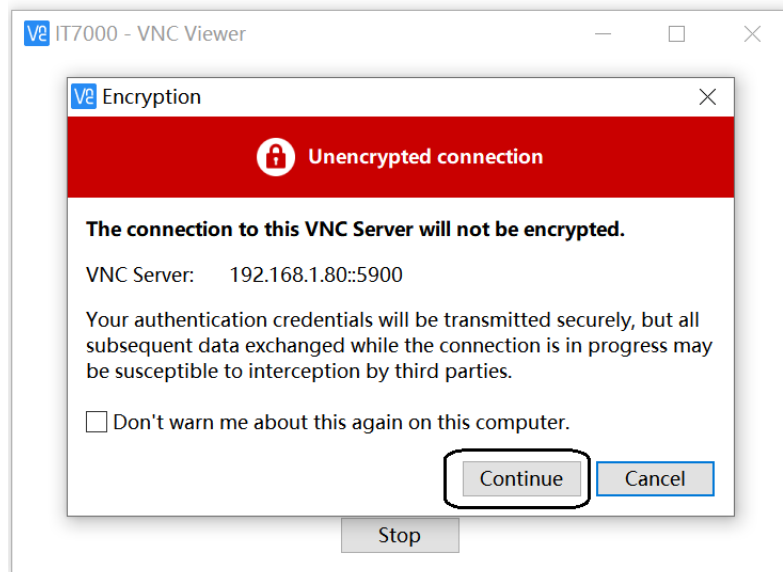
- b. 在打开的界面，将 HMI 地址设为“VNC Server”，显示名称设为“Name”。显示名称为标识用，可以根据需要自行设置，但HMI地址必须和实际要连接的 HMI 地址一致，同时保证电脑是能够访问。



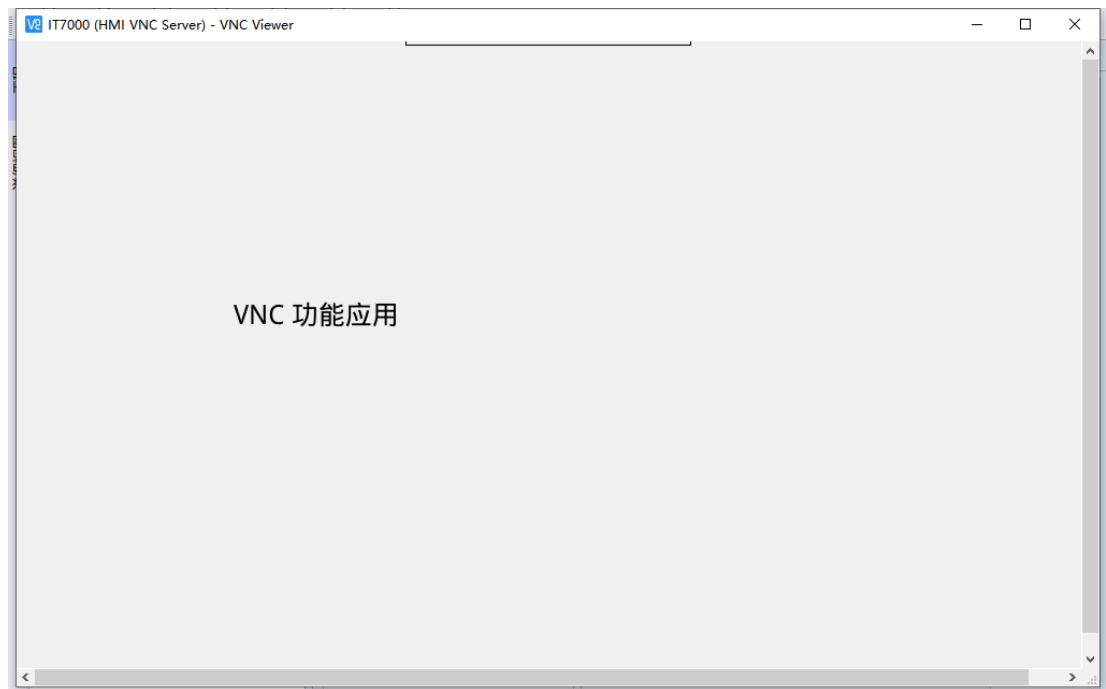
c. 填入参数后点击“OK”即完成新建连接。可在主页面看到已经新建的连接。



d. 点击新建的“IT7000”连接，即刻连接到 HMI。首次连接时会弹出如下窗口，勾选“Don't warn me about this again on this computer”，则下次不会再弹出此窗口，点击“continue”继续操作。



e. 连接成功的界面如下图所示，这时便可以在连接成功的界面上对 HMI 进行远程操作，比如鼠标点击以及键盘输入等。



### 3. 在 Android 上运行 VNC

VNC 软件可在手机商城上搜索安装，具体使用参考各个 VNC 的操作说明。理论上所有 VNC 软件都支持连接到 HMI，推荐使用以下 VNC 软件。

- RemoteToGo RDP/VNC For Android
- VNC Viewer For Android

## 4.8 平台PC组态软件应用

### 案例要求

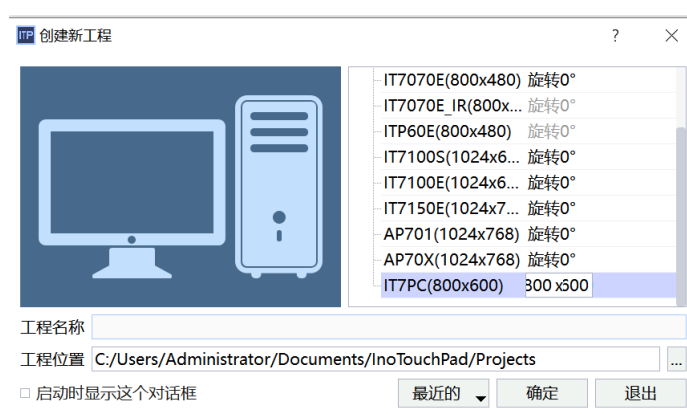
只用IT7000后台软件作为小型组态软件应用运行于PC或者工控机上。

### 案例目的

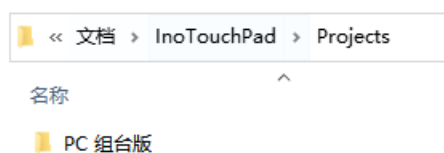
将IT7000软件作为小型组态软件应用运行于PC或者工控机上。

### 编程步骤

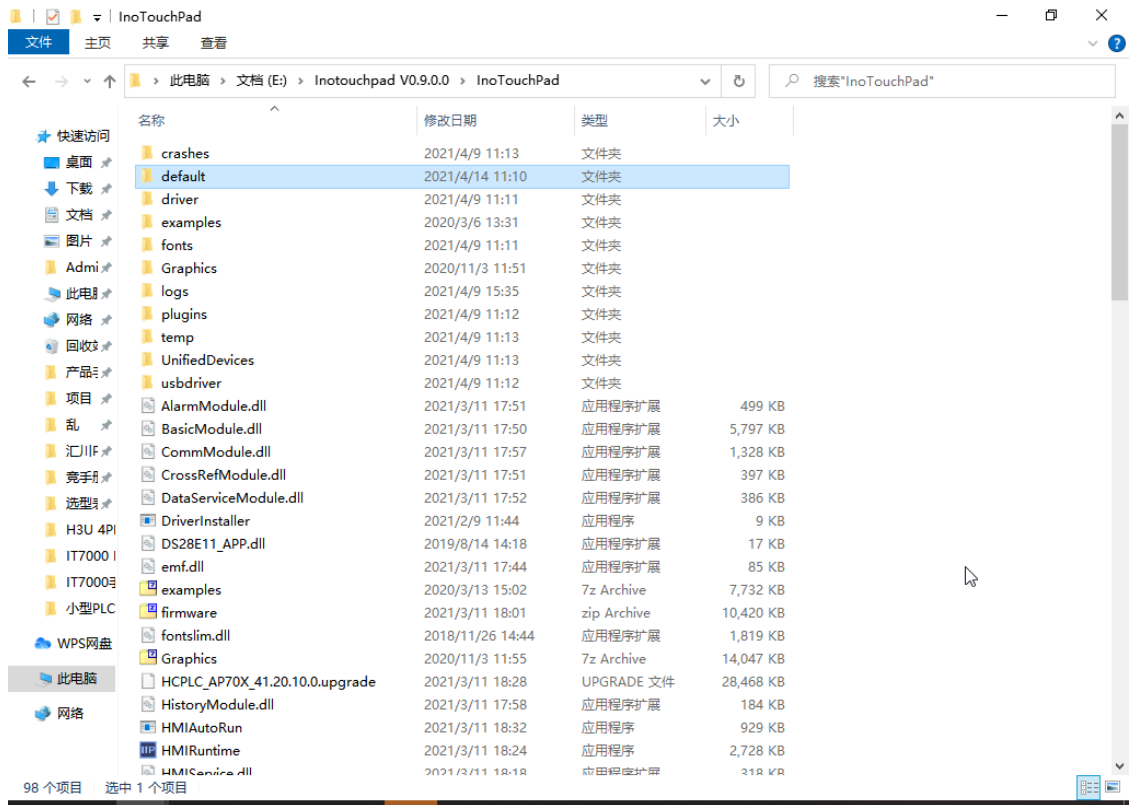
1. InoTouchPad软件安装于PC或者工控机，最新版InoTouchPad在汇川技术官网获取。
2. 打开InoTouchPad创建工程,设备类型请选择IT7PC(800x600)。



3. 创建工程后可以编辑工程，编译，然后将工程目录下所有文件拷贝到InoTouchPad软件安装目录下的的 default（需要手动新建）目录下。







4. 如果需要工程全屏运行，在左侧工程窗口里选择“HMI设置”>“工程设置”，在工程设置页面，勾选上“PC/IPC HMI自适应分辨率”选项即可。

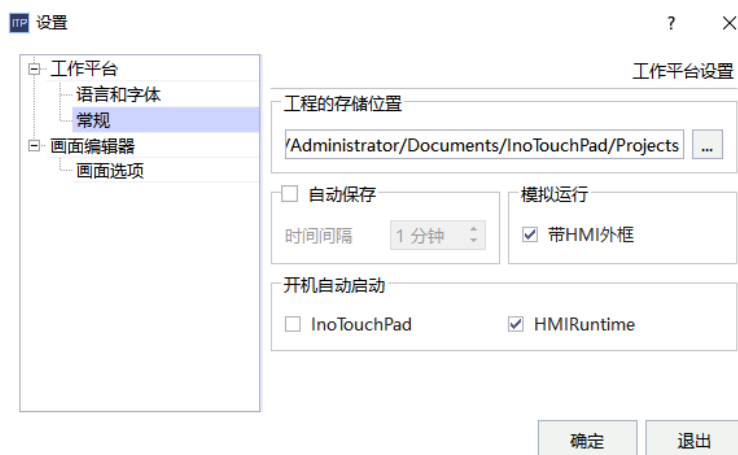


5. 进入InoTouchPad软件的安装目录，找到HMIRuntime.exe文件，在桌面创建该程序的快捷方式后，可重命名成需要的名字。

名称	修改日期	类型	大小
firmware	2021/3/11 18:01	zip Archive	10,420 KB
fontslim.dll	2018/11/26 14:44	应用程序扩展	1,819 KB
Graphics	2020/11/3 11:55	7z Archive	14,047 KB
HCPLC_AP70X_41.20.10.0.upgrade	2021/3/11 18:28	UPGRADE 文件	28,468 KB
HistoryModule.dll	2021/3/11 17:58	应用程序扩展	184 KB
HMIARun	2021/3/11 18:32	应用程序	929 KB
<b>HMIRuntime</b>	2021/3/11 18:24	应用程序	2,728 KB
HMIService.dll	2021/3/11 18:18	应用程序扩展	318 KB
HMISimulator	2021/3/11 18:30	应用程序	2,693 KB
icudt46.dll	2018/11/26 14:44	应用程序扩展	14,840 KB

6. 插入加密U盾，点击快捷方式，运行。

7. 如果需要开机启动，可在菜单栏上选择“选项”>“设置”>“工作平台”>“常规”打开设置对话框，在开启启动项里勾选上“HMIRunTime”，则每次开机启动时会软件会自动运行。



## 5 变量

### 5.1 变量的应用

#### 案例要求

- IT7000变量采集模式解析。
- IT7000变量上下限、初始值。
- IT7000变量事件应用。

#### 案例目的

- 学习使用IT7000变量采集模式解析。
- 学习使用IT7000变量上下限、初始值。
- 学习使用IT7000变量事件应用。

#### 编程步骤

- IT7000变量采集模式解析

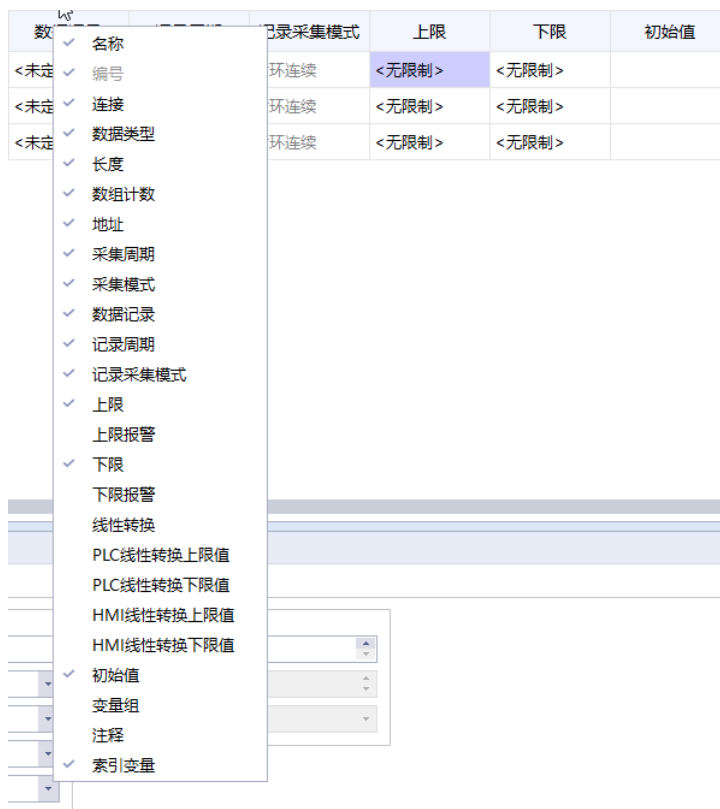
在InoTouchPad软件中界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加变量，在变量栏可以修改采集模式为根据命令、循环使用和循环连续。其中三种模式的区别如下：

- 根据命令：调用系统函数UpdateTag刷新。
- 循环使用：只有当前画面中出现当前变量才会刷新。
- 循环连续：变量可一直刷新。

+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式	数据记录	记录周期
1	D0	1	连接_1	Int16	2	1	D 0	100ms	根据命令	<未定义>	1s
2	D 1	2	连接_1	Int16	2	1	D 1	100ms	循环使用	<未定义>	1s
3	D 2	3	连接_1	Int16	2	1	D 2	100ms	循环连续	<未定义>	1s

- IT7000变量上下限解析

在InoTouchPad软件中界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加变量，在表头任意位置单击右键勾选“上限”和“下限”即可调出“上限”和“下限”栏，勾选“初始值”可对上下限进行初始值设置。

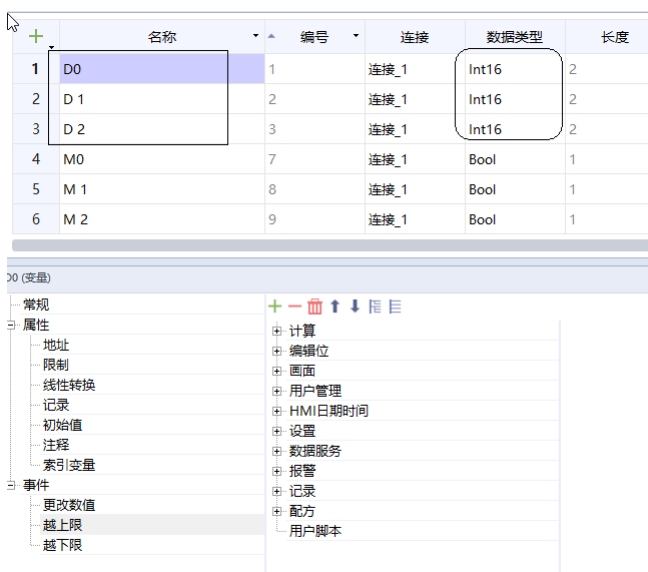


上下限可选择常量或者变量控制。

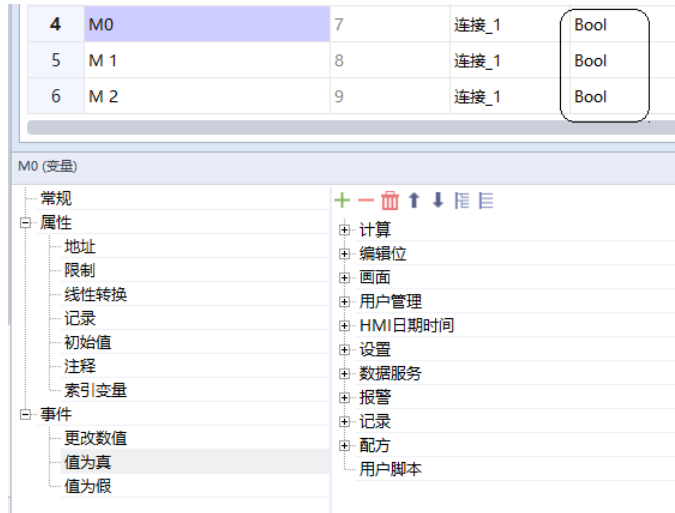
记录采...	上限	下限	初始值
循环连续	<无限制>	123_限制	
循环连续	<无限制>	123 常量	
循环连续	<无限制>	变量	

● IT7000变量事件应用

在InoTouchPad软件中界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加变量，在“事件”中可以通过“更改数值”、“越上限”、“越下限”进行事件触发（变量的数据类型为Int型）。



在“事件”中可以通过“更改数值”、“值为真”、“值为假”事件触发（变量的数据类型为BOOL型）。



## 5.2 数组下标功能的应用

### 案例要求

通过脚本实现IT7000数组下标功能。

### 案例目的

学习使用IT7000数组下标。

### 编程步骤

#### 1. 编程步骤

在InoTouchPad软件中界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加变量，如LW0, LW5, LW10, LW15，其中将数组计数改为5。

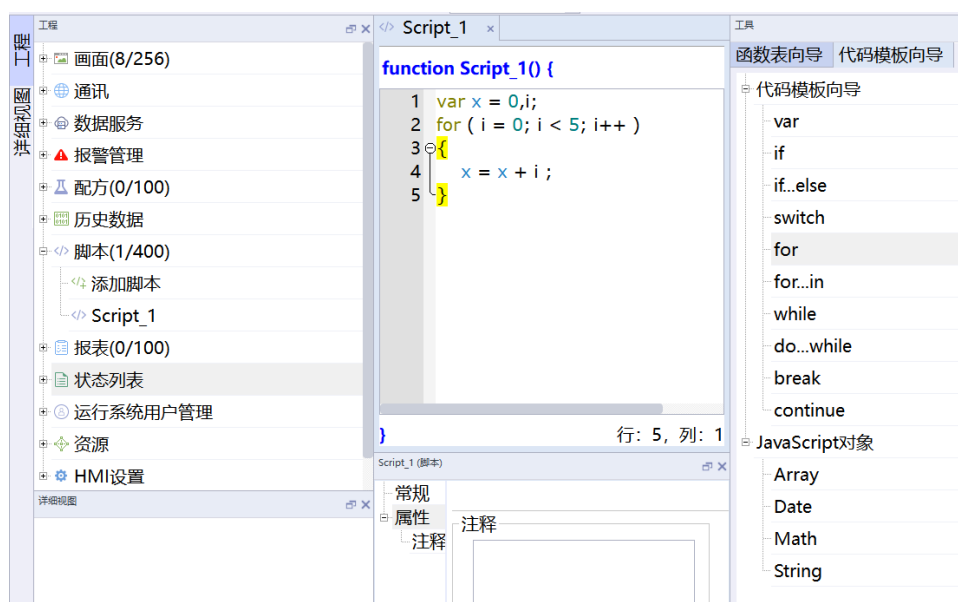
+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式	数据记录	记...	记录采...	上限	下限	初始值
1	LW0	23	<内部变...	Int16	10	5	LW 0	100ms	循环使用	<未定义>	1s	循环连续	<无限制>	<无限制>	
2	LW5	24	<内部变...	Int16	10	5	LW 5	100ms	循环使用	<未定义>	1s	循环连续	<无限制>	<无限制>	
3	LW10	25	<内部变...	Int16	10	5	LW 10	100ms	循环使用	<未定义>	1s	循环连续	<无限制>	<无限制>	
4	LW15	26	<内部变...	Int16	10	5	LW 15	100ms	循环使用	<未定义>	1s	循环连续	<无限制>	<无限制>	

此时点击变量组，在详细视图可以看到刚刚新建的LW数组变量都带有数组下标，其实际功能与普通的变量无异，但是在用法上数组下标可以用于脚本。

Id	名称	信息
200000	LW0[0]	LW 0/Int16
200001	LW0[1]	LW 1/Int16
200002	LW0[2]	LW 2/Int16
200003	LW0[3]	LW 3/Int16
200004	LW0[4]	LW 4/Int16
200005	LW5[0]	LW 5/Int16
200006	LW5[1]	LW 6/Int16
200007	LW5[2]	LW 7/Int16
200008	LW5[3]	LW 8/Int16
200009	LW5[4]	LW 9/Int16
200010	LW10[0]	LW 10/Int16
200011	LW10[1]	LW 11/Int16
200012	LW10[2]	LW 12/Int16
200013	LW10[3]	LW 13/Int16
200014	LW10[4]	LW 14/Int16
200015	LW15[0]	LW 15/Int16

## 2. 新建脚本

在InoTouchPad软件中界面左侧“工程”>“脚本”>“添加脚本”添加一个脚本，在右侧“代码模板向导”里双击for添加一个for函数。



## 3. 编写脚本

数组下标功能可以：

- 批量赋值同一个常量，如11。
- 批量赋值同一个变量，如a。
- 批量赋值进行一个四则运算的值，如i+2。
- 批量赋值不同的常量，如array[i]。

脚本代码如下：

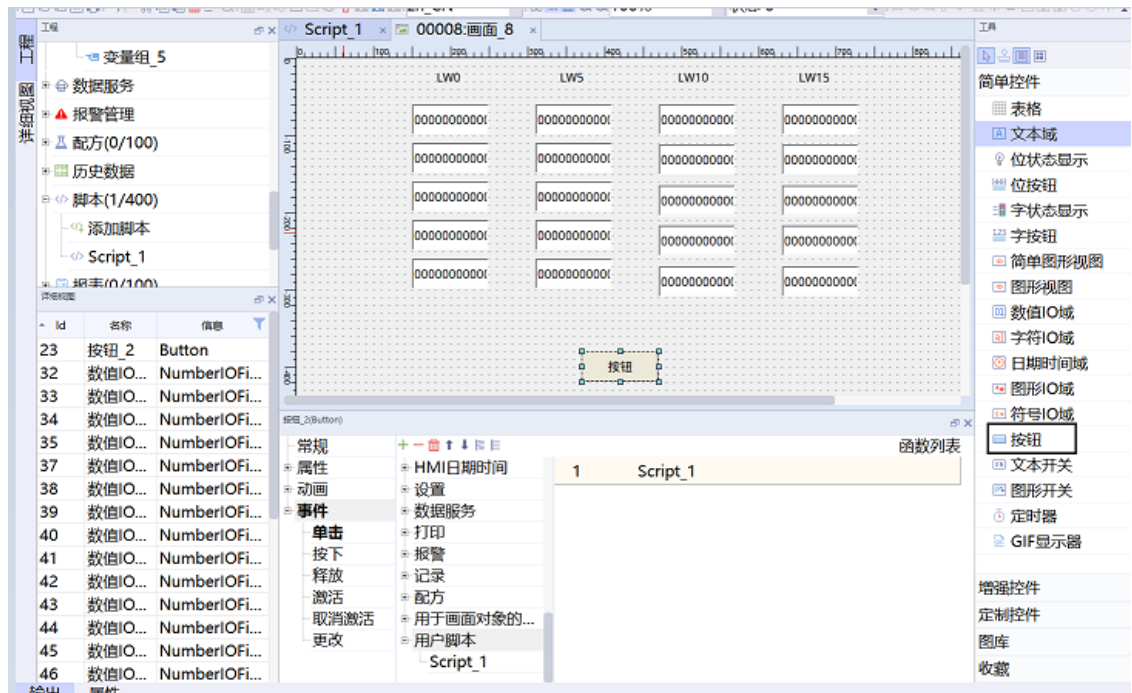
```
var x = 0,i;
var a=5;
var array=[20,21,22,23,24,25,26,27];
for ( i = 0; i < 5; i++ )
```

```
{
SmartTags('LW 0',i,true)=11;
SmartTags('LW 5',i,true)=a;
SmartTags('LW 10',i,true)=a+2;
SmartTags('LW 15',i,true)=array[i];
}
```

```
<> Script_1 x 00008:画面_8 x
function Script_1() {
1 var x = 0,i;
2 var a=5;
3 var array=[20,21,22,23,24,25,26,27];
4 for ( i = 0; i < 5; i++)
5 {
6 SmartTags("LW0",i,true) = 11; //批量赋值同一个常量
7 SmartTags("LW5",i,true) = a; //批量赋值同一个变量
8 SmartTags("LW10",i,true) = i+2; //批量赋值进行一个四则运算的值
9 SmartTags("LW15",i,true) = array[i]; //批量赋值不同的常量
10 }
```

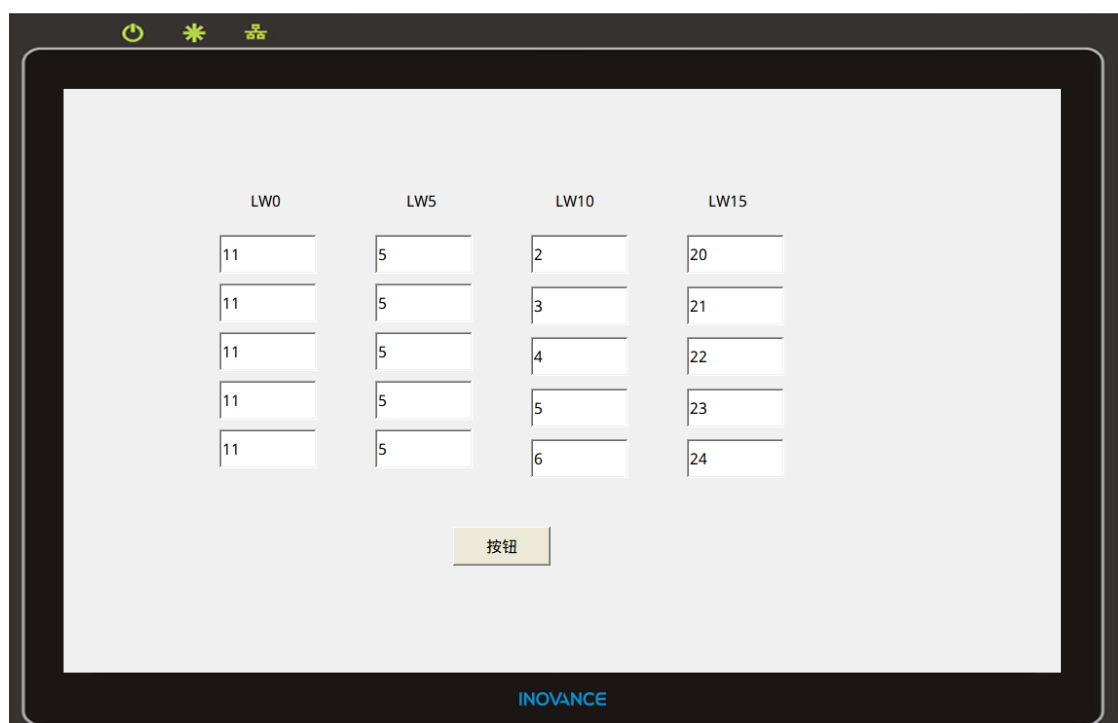
4. 组态画面

左侧“工程”>“画面”>“添加画面”添加一个画面，双击进入画面，将鼠标单击左侧新建变量组，在“详细视图”中将所有变量拖拽到画面中，再在界面右侧“工具栏”>“简单控件”>“按钮”添加一个按钮控件至画面中，单击按钮在“事件”>“用户脚本”中单击用户脚本关联脚本。



5. 运行观察结果

点击组态了调用脚本的按钮，数组LW0[0]~LW[4]显示常量11，数组LW5[0]~LW[4]显示同一个变量a=5，数组LW10[0]~LW[4]显示常量i+2 (i=0)，数组LW15[0]~LW[4]显示数组array的数值。



## 5.3 可见性的应用

### 案例要求

通过切换可见性的方式，实现在同一个页面中，显示两个不同连接的变量。

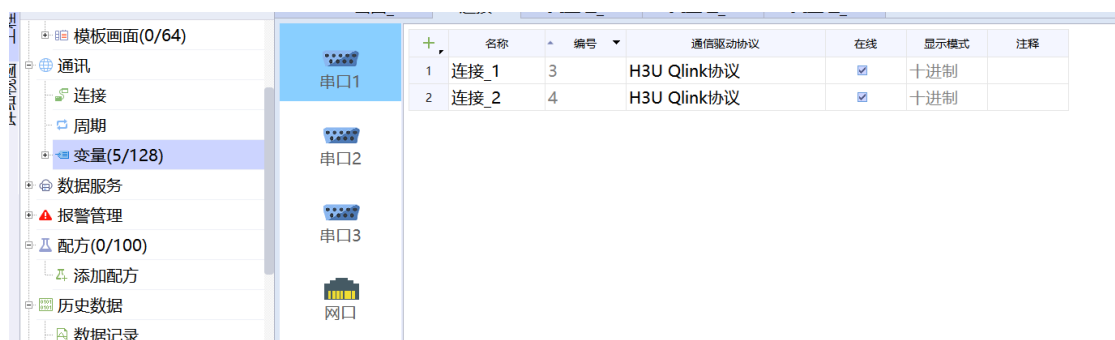
### 案例目的

通过此案例了解如何组态变量事件。

### 编程步骤

#### 1. 添加连接

在界面左侧“工程”>“通讯”>“连接”中，添加两个连接。



#### 2. 添加变量

a. 按照使用需求在界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加变量，本例两个连接各添加5个变量。

1). 连接1:



+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式
1	D 0_1	7	连接_1	Int16	2	1	D 0	100ms	循环使用
2	D 1_1	8	连接_1	Int16	2	1	D 1	100ms	循环使用
3	D 2_1	9	连接_1	Int16	2	1	D 2	100ms	循环使用
4	D 3_1	10	连接_1	Int16	2	1	D 3	100ms	循环使用
5	D 4_1	11	连接_1	Int16	2	1	D 4	100ms	循环使用

2). 连接2:

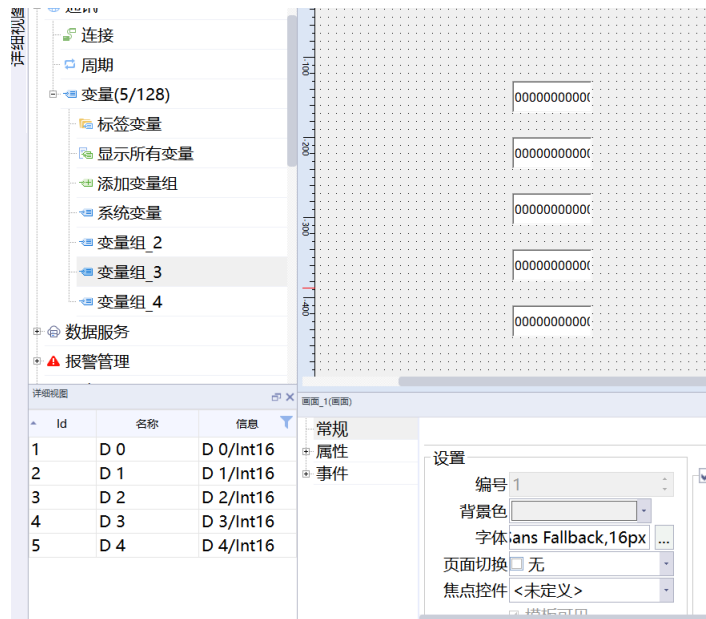
+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式
1	D 0	1	连接_2	Int16	2	1	D 0	100ms	循环使用
2	D 1	2	连接_2	Int16	2	1	D 1	100ms	循环使用
3	D 2	3	连接_2	Int16	2	1	D 2	100ms	循环使用
4	D 3	4	连接_2	Int16	2	1	D 3	100ms	循环使用
5	D 4	5	连接_2	Int16	2	1	D 4	100ms	循环使用

b. 界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加一个辅助变量，用于控制可见性。

+	名称	编号	连接	数据类型	长度	数组计数	地址
1	LW 0	12	<内部变量>	Int16	2	1	LW 0

### 3. 添加IO域

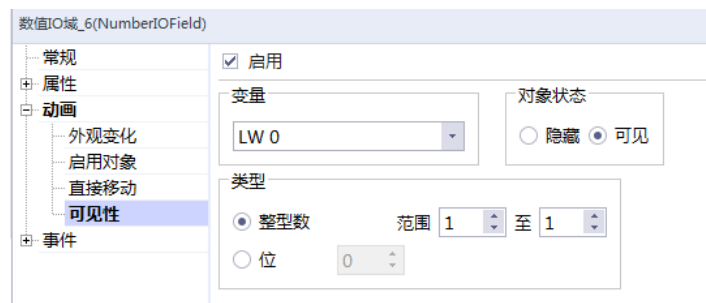
将变量拖拽组态到画面中，两个相同的变量放置在同一位置，本例中即D 0与D 0\_1，放置在一起，D 0覆盖在D 0\_1上面。



### 4. 设置IO域的可见性

在IO域的“动画”>“可见性”中，启用可见性，变量组态为新建的辅助变量LW 0，对象状态设置为可见，类型设置为整数数1至1。这样设置之后，只有LW 0为1时，该IO域才为可见状态。

按照上述方法，将剩余的4个IO域进行设置。



## 5. 总结

当设置完成之后，LW 0为1时，运行画面中的IO域显示为连接1对应的变量，当LW 0不为1时，运行画面中的IO域显示为连接2对应的变量。

在实际应用中，可根据自己的需求，设置更多的连接，只需要设置好对应的可见性即可。

## 5.4 在变量变化时自动弹出窗口

### 案例要求

通过HMI自带的控件，实现变量变化时自动弹出窗口。本案例模拟当变量LB 0为真时，弹出一个弹出窗口。

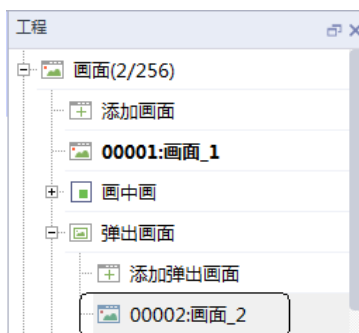
### 案例目的

通过此案例了解如何组态变量事件。

### 编程步骤

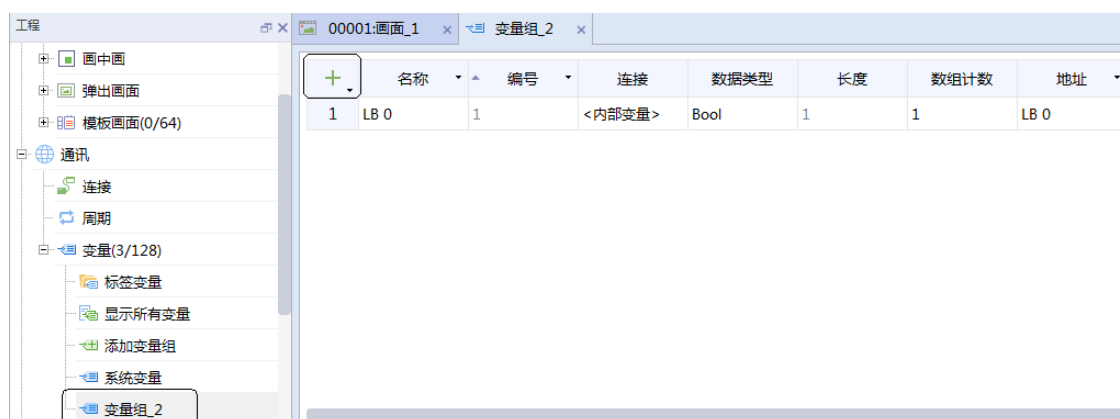
#### 1. 添加画面

在界面左侧“工程”>”画面”>”弹出画面”>”添加弹出画面”中，添加弹出“画面\_2”，画面内容和画面尺寸根据自己的需要进行组态。



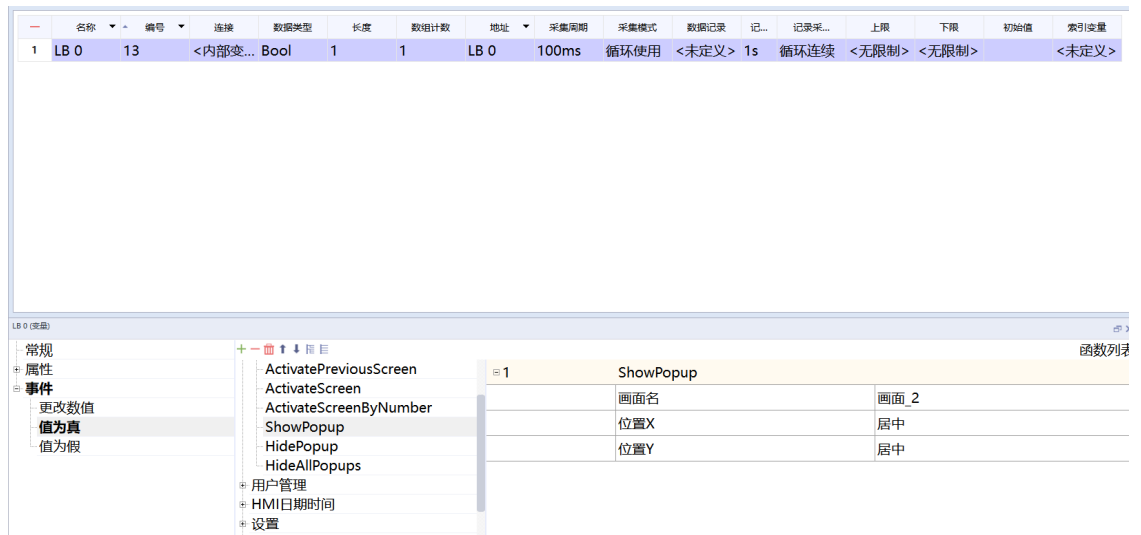
#### 2. 添加变量

添加变量LB 0，为Bool类型变量。



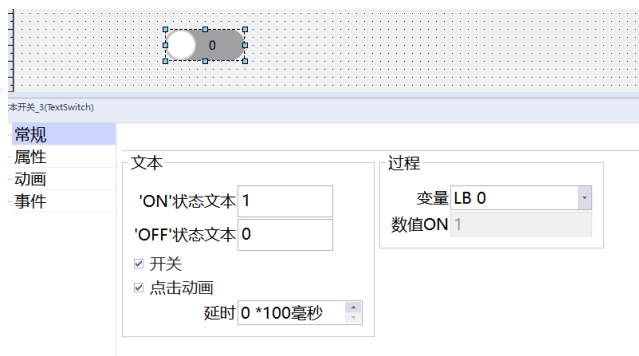
#### 3. 设置变量事件

在变量的“事件”>”画面”中找到ShowPopup函数，双击添加到列表中，然后为我们需要弹出的弹出画面组态画面名。



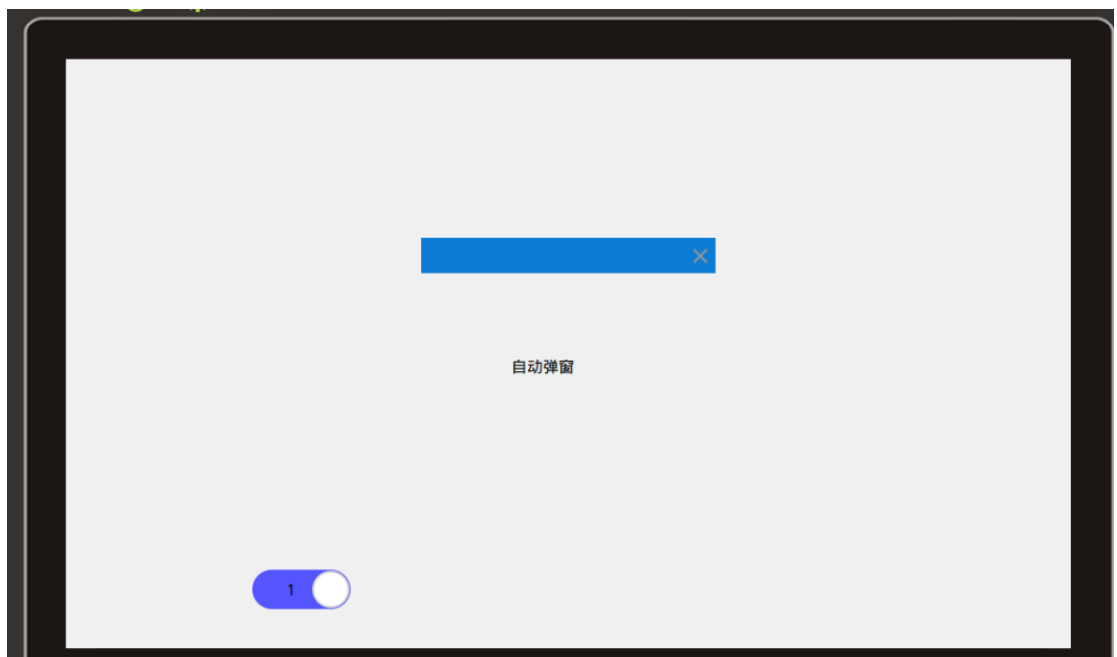
4. 添加文本开关关联变量

界面右侧“工具”>“简单控件”添加一个文本开关，在“常规”>“过程”>“变量”关联变量LB 0。



5. 离线模拟效果展示

设置完成后，当变量值变为真时，HMI会自动弹出对应的窗口。



## 5.5 索引变量功能的应用

IT7000索引变量功能说明：IT7000提供128个16位和128个32位的索引变量，用户可以通过组态索引变量进行变址操作，通过改变索引变量的值，来实现同一组控件访问多个地址空间。

### 案例要求

通过IT7000索引变量实现LW0变址功能。

### 案例目的

学习使用IT7000索引变量。

### 编程步骤

#### 1. 新建索引变量

使用索引变量时，需要先新建索引变量，在界面左侧“工程”>“通讯”>“变量”>“系统变量”，点击右侧“索引变量”旁边的“+”号即可进行展开，分为16 Bit和32 Bit，点击16 Bit旁边的“+”号进行展开，双击对应的索引变量即可进行添加，如下图所示。

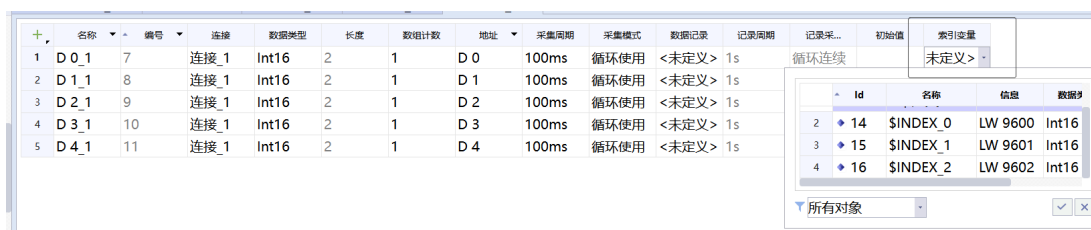


#### 2. 关联索引变量

关联索引变量有两种方法：

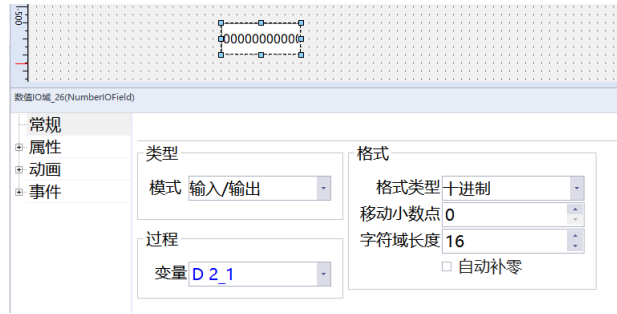
##### a. 变量表选择关联索引变量

如下图所示，界面左侧“工程”>“通讯”>“变量”>“添加变量”，点击左上角“+”号进行变量表的添加，在变量表的“索引变量”列，点击下拉框即可选择具体的索引变量。



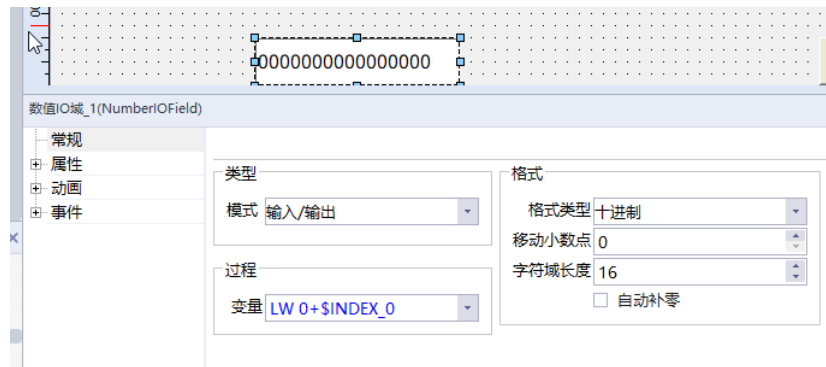
关联后的变量呈现蓝色，对应控件关联的地方也是蓝色，如下图所示。

+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期	采集模式	数据记录	记录周期	记录采...	初...	索引变量
1	D 0_1	7	连接_1	Int16	2	1	D 0	100ms	循环使用	<未定义>	1s	循环连续		\$INDEX_0
2	D 1_1	8	连接_1	Int16	2	1	D 1	100ms	循环使用	<未定义>	1s	循环连续		\$INDEX_1
3	D 2_1	9	连接_1	Int16	2	1	D 2	100ms	循环使用	<未定义>	1s	循环连续		\$INDEX_2
4	D 3_1	10	连接_1	Int16	2	1	D 3	100ms	循环使用	<未定义>	1s	循环连续		<未定义>
5	D 4_1	11	连接_1	Int16	2	1	D 4	100ms	循环使用	<未定义>	1s	循环连续		<未定义>



b. 手动输入关联索引

IT7000支持手动输入来关联索引变量，如下图所示，在控件“常规”>“过程”输入框输入指定格式的字符即可关联索引变量。指定的输入格式是“变量名+\$INDEX\_索引变量序号”这种格式，例如LW0+\$INDEX\_0，就表示变量LW0关联了索引变量INDEX\_0。

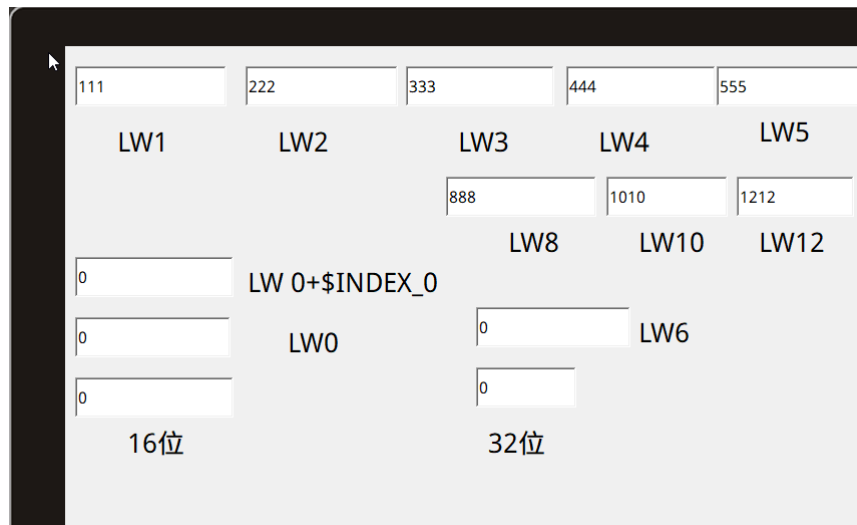


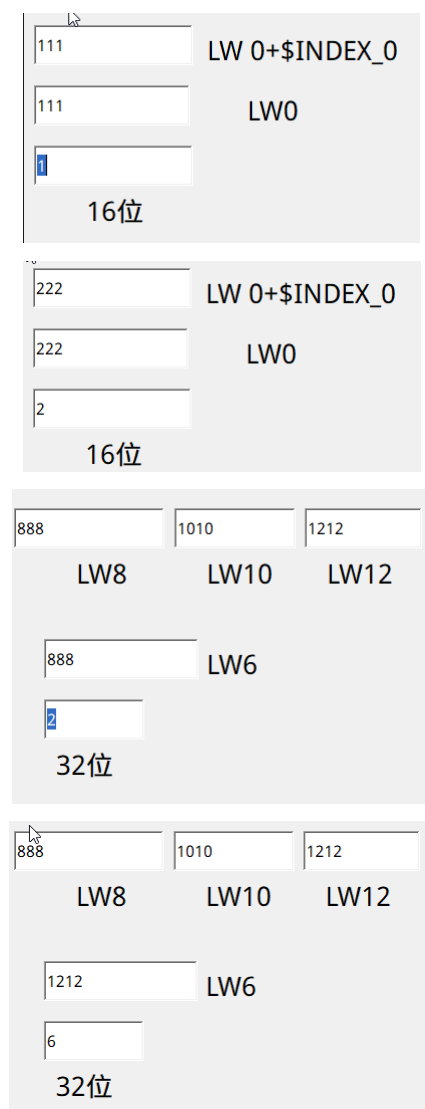
- 必须先创建索引变量，才能手动输入关联索引变量。
- 变量表关联索引变量和手动输入关联索引的区别在于，手动关联会多生成一个新的变量。

id	name	value	type	length	precision	unit	refresh	usage	comment	display	
32	LW 0	53	<内部变量>	String	10	1	LW 0	100ms	循环使用	<未定义>	1s
33	LW 0+\$INDEX_0	54	<内部变量>	String	10	1	LW 0	100ms	循环使用	<未定义>	1s

3. 结果分析

对于给定的数值存储在寄存器LW0,1,2,3,4,5……若其中LW0关联索引变量\$INDEX\_0，则在索引变量\$INDEX\_0中输入数字1,2,3,4,5……,LW0显示对应的寄存器LW0/LW1/LW2/LW3/LW4/LW5中的数值，如下图所示。





## 5.6 通过变量更改棒图、滚动条、进度条、流动块的状态

### 案例要求

- 通过变量更改棒图、滚动条、进度条、流动块的状态。
- 如何添加棒图、滚动条、进度条。
- 如何添加自定义规格流动块。
- 滚动条速度设定，流动块、棒图、进度条状态监控。

### 案例目的

- 通过此案例掌握棒图、滚动条、进度条、流动块基本应用。
- 通过此案例学会流动块流动速度及方向的设定。

### 编程步骤

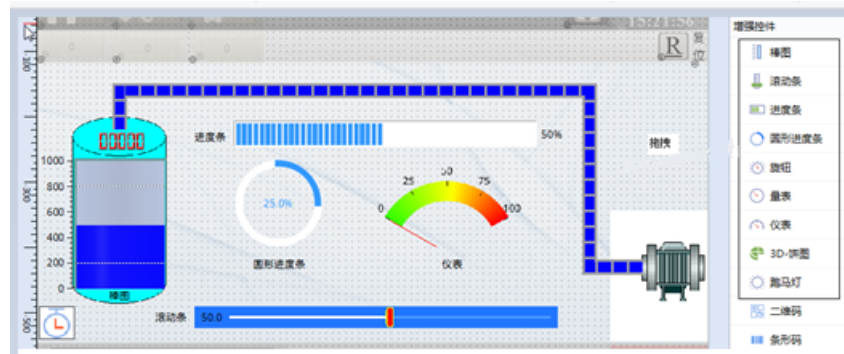
1. 添加变量

添加类型变量：左侧项目树“工程”>“通讯”>“变量”>“添加变量组”添加一个变量组新建所需变量，如下图。

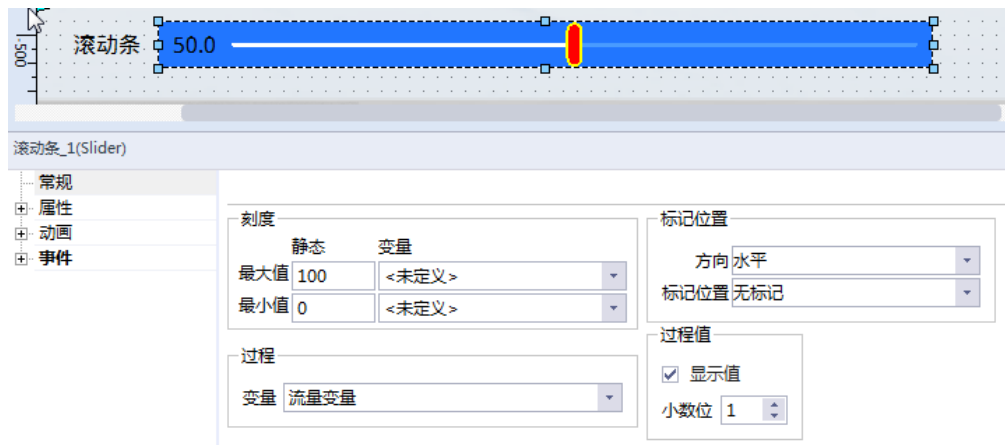
+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期
1	流量变量	7	<内部变量>	Int16	2	1	LW 0	1s
2	棒状变量	8	<内部变量>	Int16	2	1	LW 1	1s
3	开启	10	<内部变量>	Bool	1	1	LB 100	1s

2. 添加基本控件

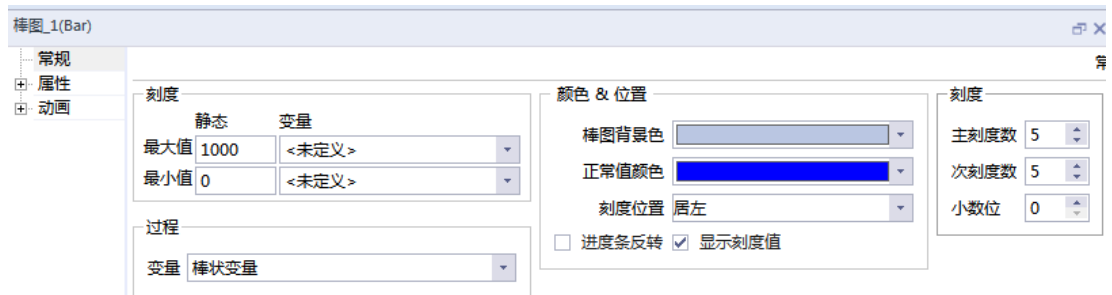
a. 棒图、滚动条、进度条、仪表控件添加：右侧工具栏“工具”>“增强控件”>“棒图、滚动条、进度条、仪表”，通过拖拽方式实现各控件添加。



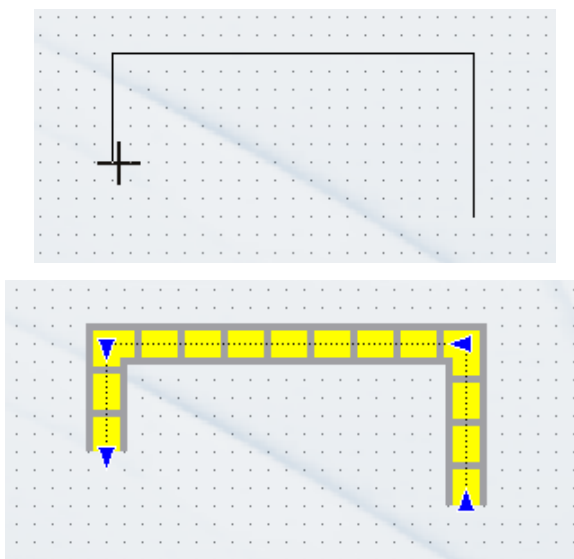
b. 在滚动条、进度条、仪表控件属性视图“常规”>“过程”>“变量”添加已添加好的变量“流量变量”，属性视图“常规”>“刻度”设定静态最大值为100。



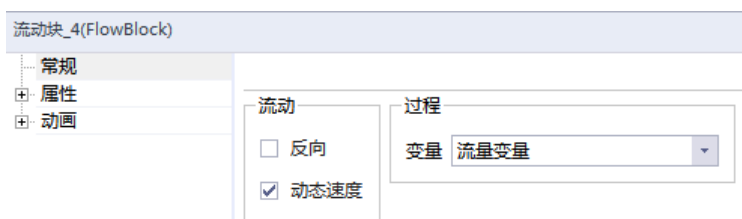
c. 棒图属性视图“常规”>“过程”>“变量”添加已添加好的变量“棒状变量”，属性视图“常规”>“刻度”设定静态最大值为1000；根据需求调整颜色&位置、刻度等常规参数。



- d. 添加自定义规格流动块：右侧工具栏“工具”>“增强控件”>“流动块”，单击鼠标左键选择流动块后，在画面编辑区单击鼠标左键即可开始绘画起流动块，当需要装换方向是再次单击左键即可转换，绘画完成单击鼠标右键即可完成流动块图形绘制。



- e. 流动块属性视图“常规”>“流动”>“动态速度”勾选后，添加过程变量“流量变量”。



- f. 添加辅助定时器：右侧工具栏“工具”>“简单控件”>“定时器”通过拖拽添加定时器，根据定时器属性视图“常规”>“设置”时间间隔5\*100ms，“常规”>“过程”添加过程变量“开启”。

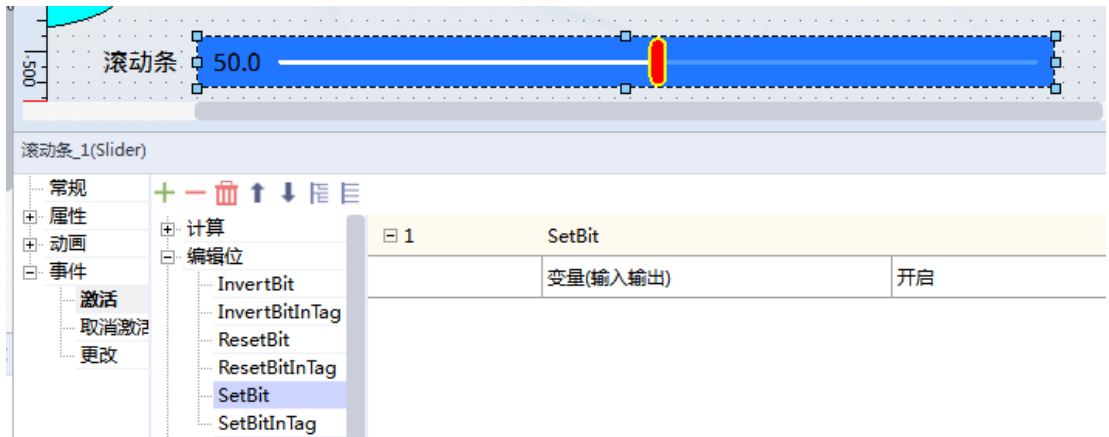


定时器属性视图“时间”>“超时”>“计算”添加系统函数“IncreaseValue”，如下图。



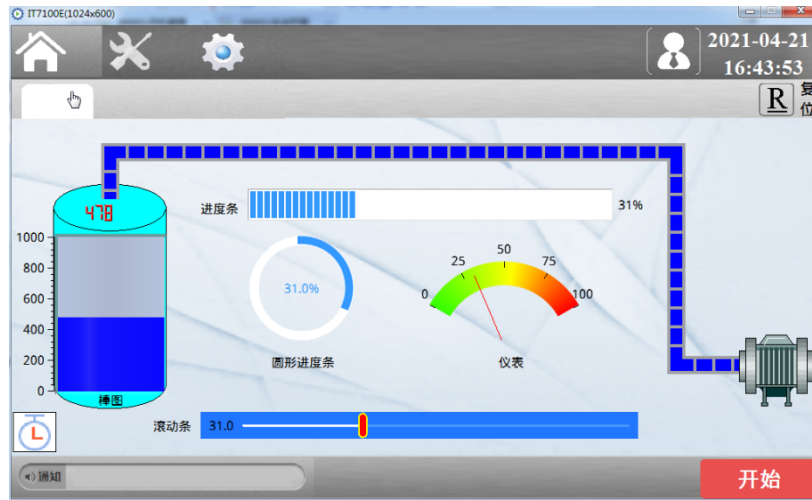
- g. 通过滚动条控件添加系统函数“SetBit”对定时器过程变量实现置位开启。





3. 通过变量更改棒图、滚动条、进度条、流动块的状态案例完成

通过修改滚动条变量状态可直观显示棒图、进度条、流动块的状态，效果图如下。



## 5.7 使用PLC变量切换画面

### 案例要求

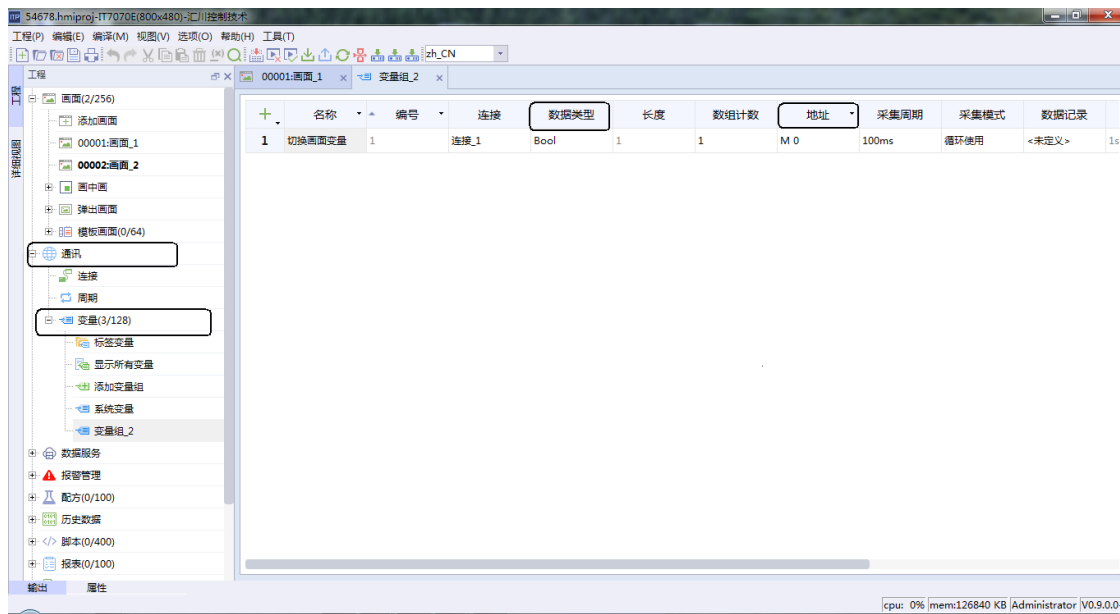
- 使用plc变量对画面进行切换操作。
- 完成PLC和HMI之间变量的建立与关联。
- 实现变量切换页面的操作。

### 案例目的

通过案例学会利用PLC变量切换画面。

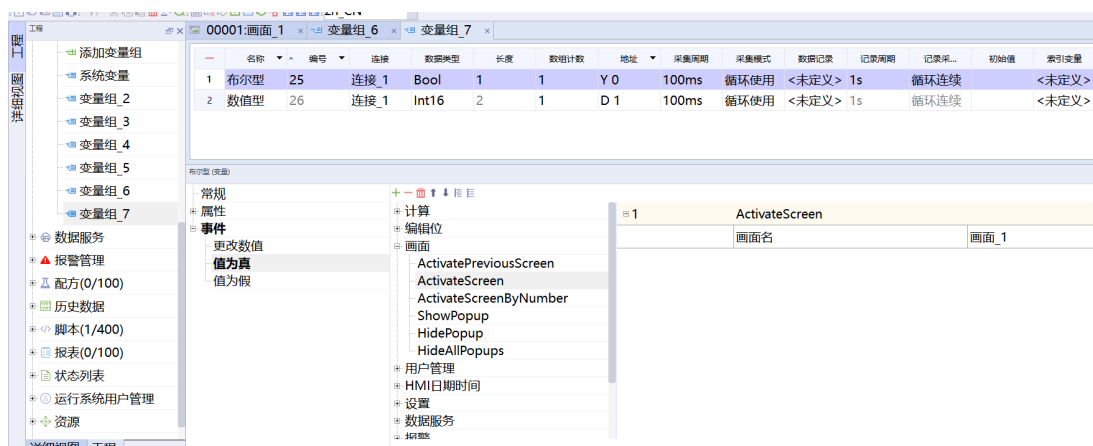
### 编程步骤

1. 完成PLC和HMI之间变量的建立与关联
  - a. 在InoTouchPad软件界面左侧“工程”>“通讯”>“变量”>“变量组”进行添加变量的操作。
  - b. 在添加的变量中对PLC和HMI的变量进行关联，对变量的数据类型和变量地址进行选择。

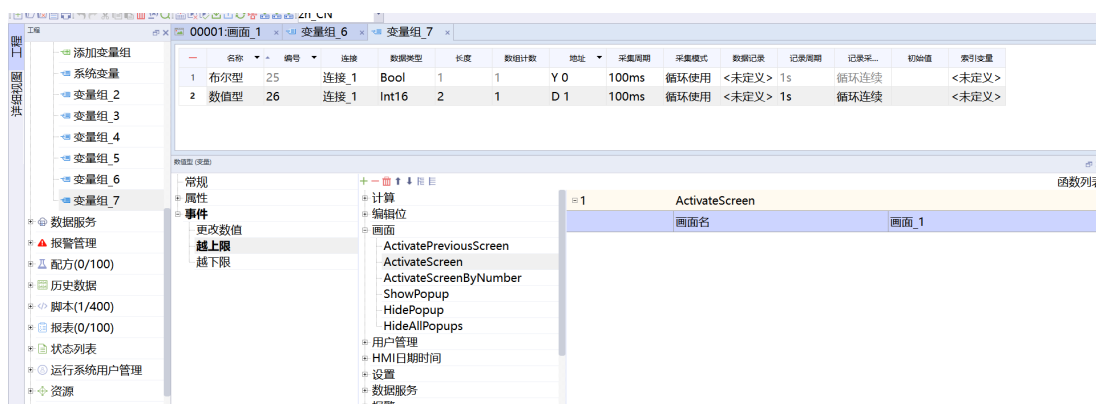


2. 实现变量切换页面的操作

a. 选中定义好的变量，当勾选的是布尔型的变量时，在“事件” > “值为真” or “值为假” > “画面” > “ActivateScreen” 选中画面名进行画面切换。



b. 当勾选的是数值型的变量在“事件” > “越上限” or “越下限” > “画面” > “ActivateScreen” 选中画面名进行画面切换。



## 6 批量操作

### 6.1 字体批量切换

#### 案例要求

通过案例进行批量字体切换，方便工程字体切换。

#### 案例目的

通过案例学会批量切换字体。

#### 编程步骤

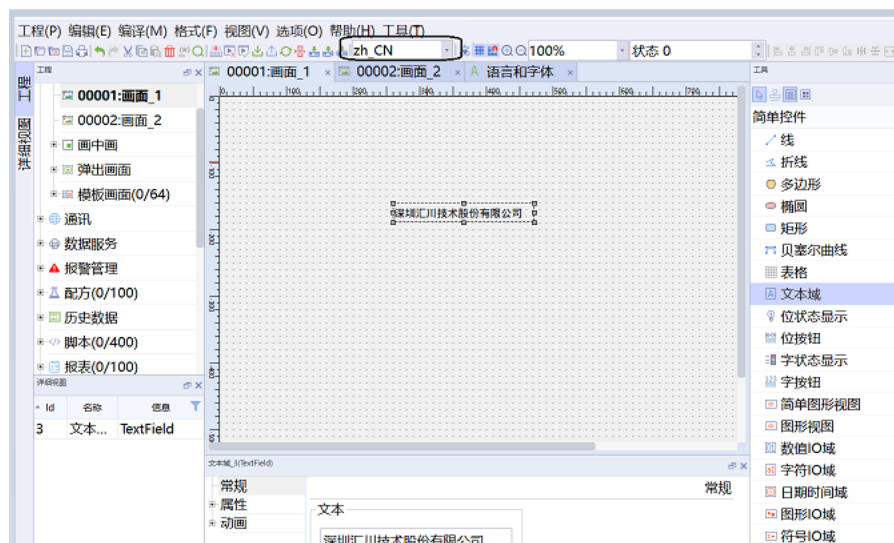
##### 1. 添加画面

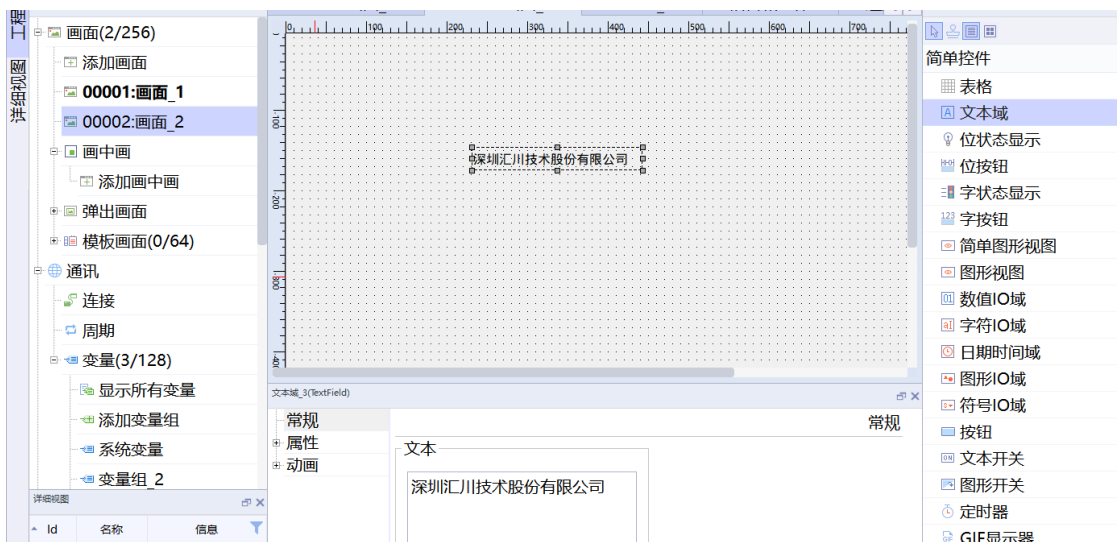
打开软件，新建工程，在界面左侧“工程”>“画面”>“添加画面”添加两个画面如图所示。



##### 2. 添加文本

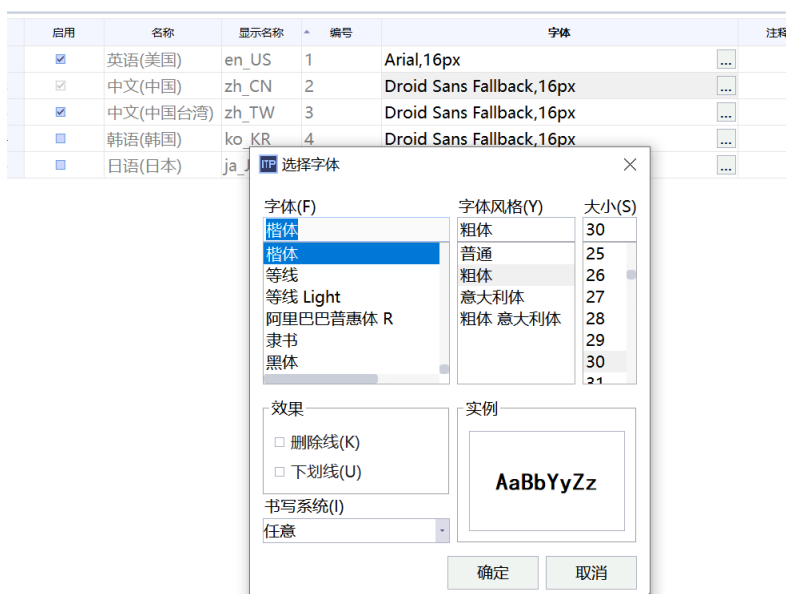
分别在画面\_1和画面\_2在界面右侧“工具”>“简单控件”里添加一个文本域如图所示。



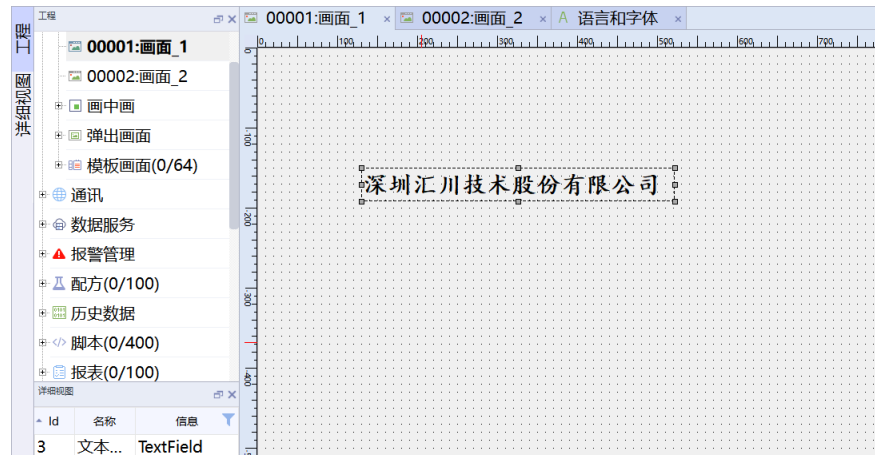


### 3. 修改字体

在界面左侧“工程”>“资源”>“语言和字体”可看到当前工程应用的字体和大小信息，选中字体栏中的“...”即可修改对应的工程中字体。



### 4. 修改后效果图下所示



5.



**说明** 字体批量切换只针对文本属性未更改过的文本，已经更改过的不能进行批量设置。

# 7 工程设置

## 7.1 设置屏保及背光灯

### 案例要求

对HMI屏保及相关背光灯界面进行以下设置：

- 完成屏保时间设置。
- 完成屏保页面设置。
- 完成背光灯时间设置。

### 案例目的

- 通过案例可以完成对屏保时间、屏保页面的设置。
- 通过案例可以完成对背光灯时间的设置。

### 编程步骤

- 屏保时间设置

在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”打开设置页面,在“屏幕保护&背光设置”中可以修改进入屏保等待时间,当设置为0分钟时表示关闭屏保功能,此时“进入屏保时切换到画面”选项框为灰色,表示禁止切换。



- 屏保页面设置

1. 在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”打开设置页面,在“屏幕保护&背光设置”栏中可以修改进入屏保时切换到页面,当进入屏保等待时间不为0分钟时,可以选择任意画面为屏保画面。



2. 点击进入“屏保切换到画面”，选择相应的屏保画面。



● 背光灯时间设置

在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”打开设置页面，在“屏幕保护&背光设置”中可以修改关闭背光等待时间，当设置为0分钟时表示不关闭背光灯。



## 7.2 设置起始画面以及起始语言

### 案例要求

通过HMI软件完成对起始画面及语言的设置：

- 对起始画面进行设置。
- 对起始语言进行设置。

### 案例目的

通过此案例学会对起始画面、起始语言的相关更改。

### 编程步骤

- 起始画面设置

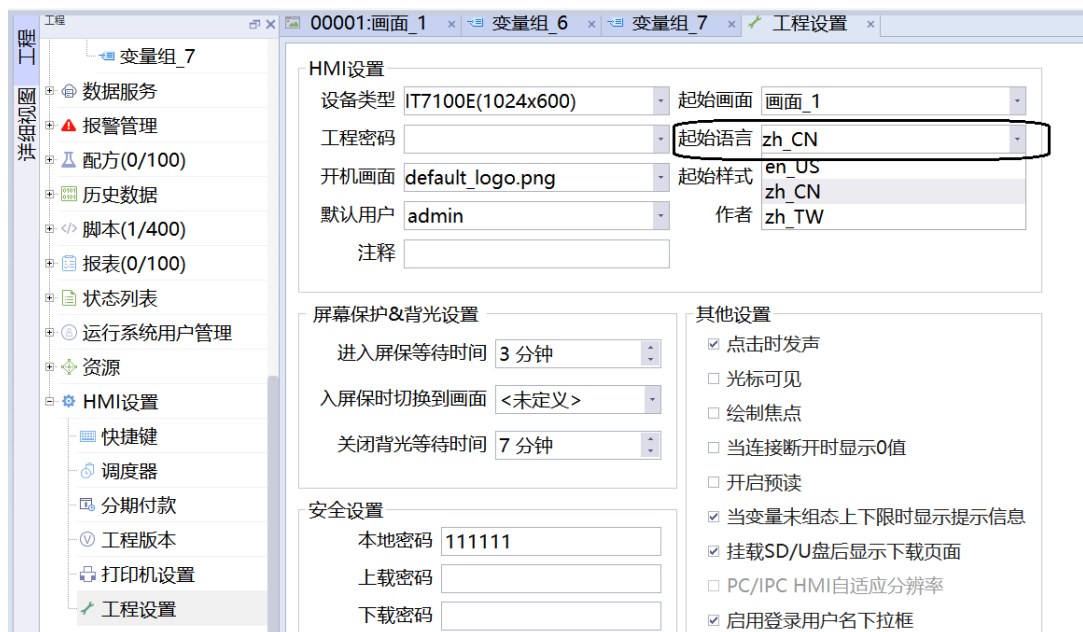
在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”打开设置页面，在“HMI设置”中可以修改“起始画面”，设置任意页面为初始页面。



- 起始语言设置

在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”打开设置页面，在“HMI设置”中可以修改“起始语言”，设置任意选择语言，En\_us英文 \Zh\_CN简体中文 \Zh\_TW繁体均支持。





## 7.3 工程备份

案例说明：“工程版本”主要用于用户制作的工程备份和恢复，最多可以备份16个版本。


### 案例要求

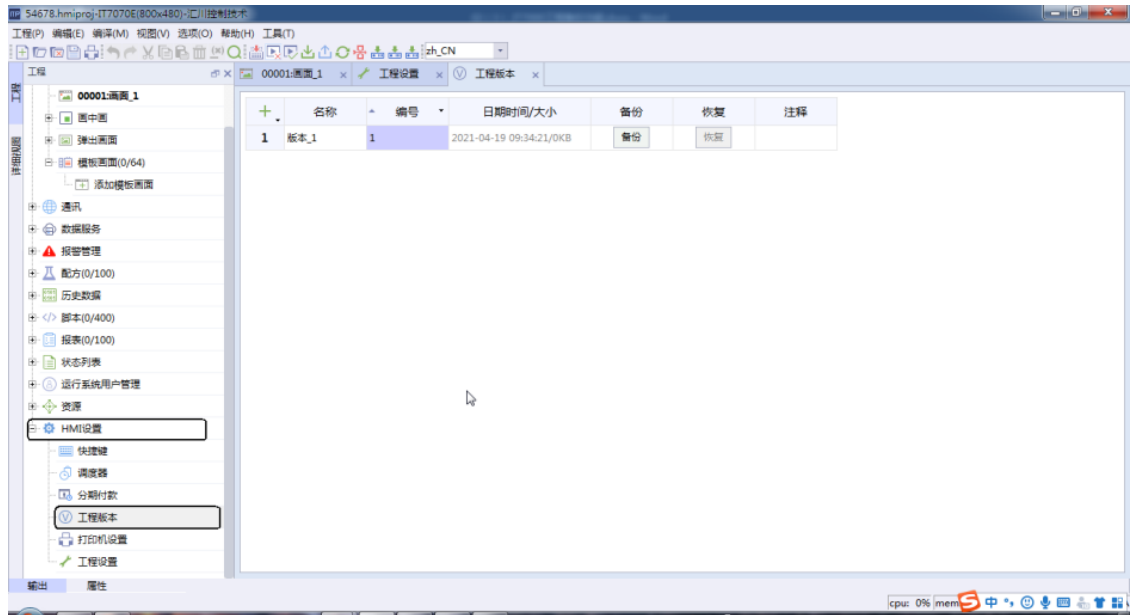
1. 通过HMI完成对工程备份功能的实现。
2. 工程备份功能的使用。

### 案例目的

通过此案例学会工程备份的使用。

### 编程步骤

1. 打开InoTouchPad软件界面左侧“HMI设置”>“工程版本”。
2. 点击 ，即可新增版本进行工程备份。
3. 点击备份栏中的备份按钮即可备份工程。



名词解释：

+	名称	编号	日期时间/大小	备份	恢复	注释
1	版本_1	1	2021-04-19 09:34:21/0KB	备份	恢复	

- 备份：点击备份，即可对当前工程进行备份。
- 恢复：点击恢复，即可恢复到该工程已备份的版本。
- 注释：对备份工程进行注释。

## 7.4 使用U盘或SD卡升级HMI程序

### 案例要求

通过U盘或SD卡对IT7000程序进行升级：

- 使用U盘或者SD卡进行固件升级操作。
- 将固件拷入U盘或SD卡内。
- 使用U盘或者SD卡进行HMI程序下载。

### 案例目的

- 通过此案例学会使用U盘或SD卡升级固件。
- 通过此案例学会将固件拷入U盘或SD卡内。
- 通过此案例学会使用U盘或SD卡对程序进行下载升级。

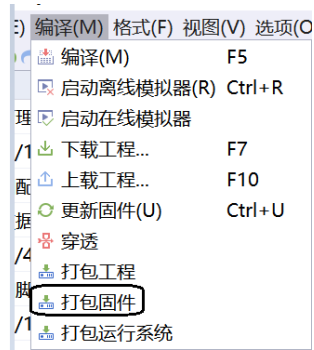
### 编程步骤

#### 1. 打包固件

##### a. 打包固件。

主要有以下两种打包方式：

- 点击“编译”菜单下的”打包固件”按钮。

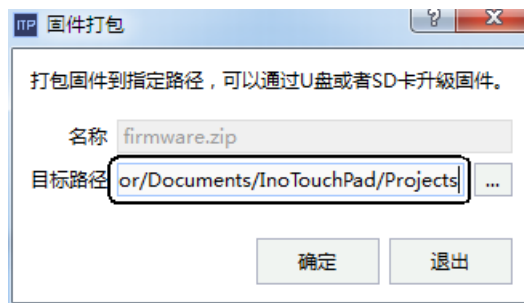


- 点击快捷菜单栏下的“打包固件”按钮。



- b. 路径选择。

打包时可以自由选择“目标路径”，之后点击“确定”。



- c. 打包成功

打包成功会有提醒，在目录下可以找到打包好的工程压缩包文件。



## 2. 将固件文件拷入U盘或SD卡

将固件打包好的文件拷贝至U盘或者SD卡的根目录下面（注意：U盘或者SD卡的文件系统需要是FAT32格式）。

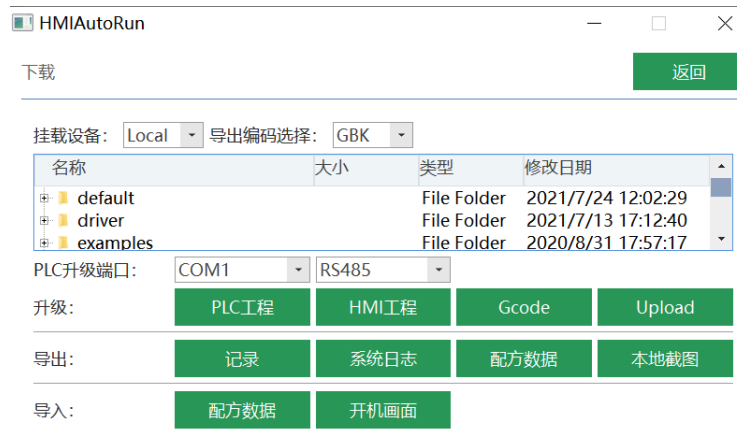


### 3. 使用U盘或SD卡下载HMI程序

- a. 将U盘或者SD卡安装到触摸屏上。HMI运行时插入U盘或者SD卡，会自动弹出控制面板画面，如下图所示。



- b. 点击“下载”进入控制面板中的“下载页面”，如图所示。



- c. 单击选择“挂载设备”和“编码选择”下拉菜单，选择固件位置。
- d. 点击“升级”>“HMI工程”，等待固件升级完成。

## 7.5 密码功能应用

### 案例要求

通过案例学会以下密码功能的相关应用：

- 修改系统密码。
- 修改工程密码。
- 修改上载密码。
- 修改下载密码。

### 案例目的

通过案例了解IT7000系统密码、工程密码、上载密码、下载密码的修改方式。

### 编程步骤

- 系统密码的修改方式
  - 方法一：控制面板修改
    1. 进入触摸屏的控制面板界面，输入初始密码111111。



2. 点击进入安全设置界面。

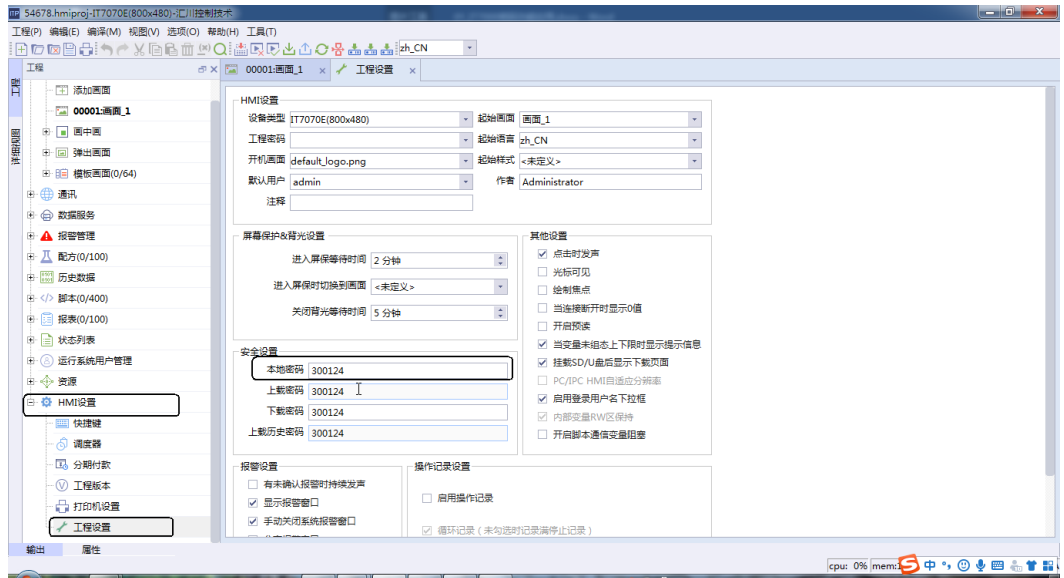


3. 先输入之前的密码，再输入新密码并确认，点击应用，便完成了本地密码的修改；下次登录控制面板需要输入新密码才可登录。



- 方法二：InoTouchPad软件修改

1. 在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”>“安全设置”中可以通过设置“本地密码”更改系统密码，初始密码为111111，如图所示。

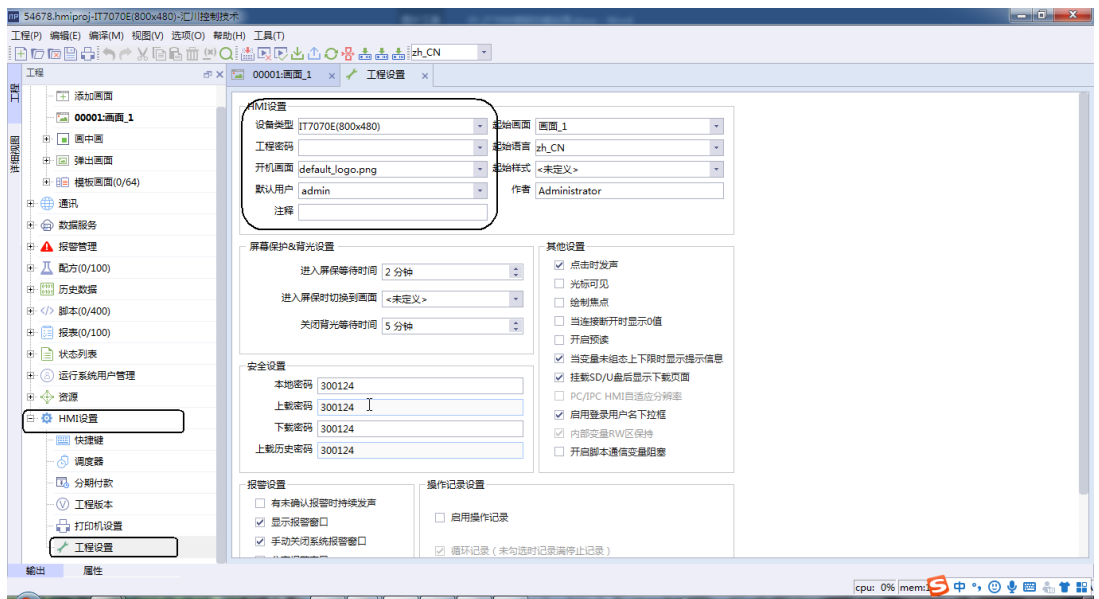


2. 修改完成后下载工程即可生效。下次登录控制面板需要输入新密码才可登录。

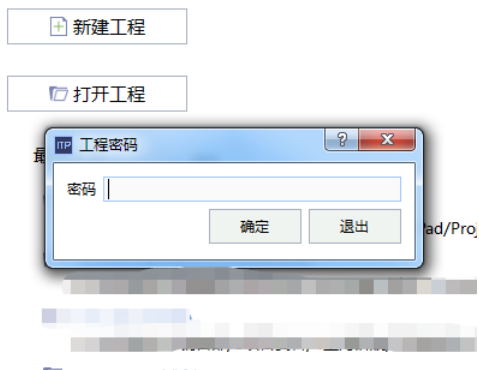
- 工程密码的修改方式

工程密码：设置工程密码后，每次打开工程需要输入正确的工程密码，才能打开工程。

1. 在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”中的“HMI设置”栏中，即可修改“工程密码”，工程密码默认状态为空，即不需要输入密码直接打开工程。



2. 设置工程密码后，在每次打开工程时都会弹出如图所示的密码界面，需要正确输入密码才能打开工程。通过此方式可以保护知识产权。

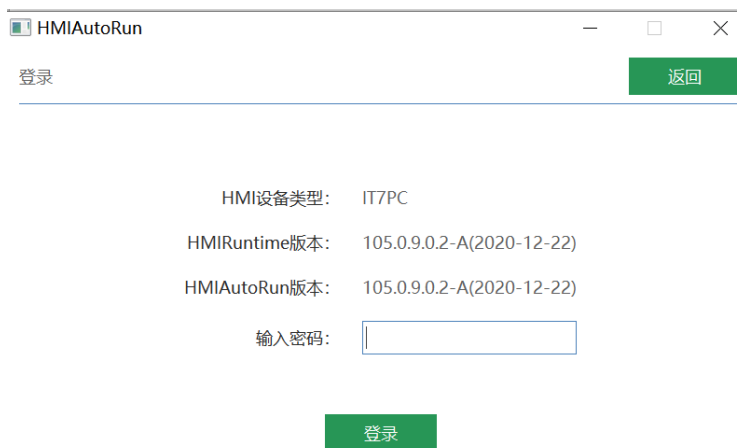


- 上载密码的修改方式

上载密码：默认状态下上载HMI程序时不需要上载密码，可直接上载。在设置上载密码后，每次上载需要输入正确的上载密码，才能上载工程。

- 方法一：控制面板修改

1. 进入触摸屏的控制面板界面，输入初始密码111111。



2. 点击进入安全设置界面，如图所示。



3. 在上载密码界面，先输入之前的密码（初次为空），再输入新密码并确认，点击应用，便完成了上载密码的修改，再次上载工程时，需要输入密码才可上载。



安全

返回

本地密码

上载密码

下载密码

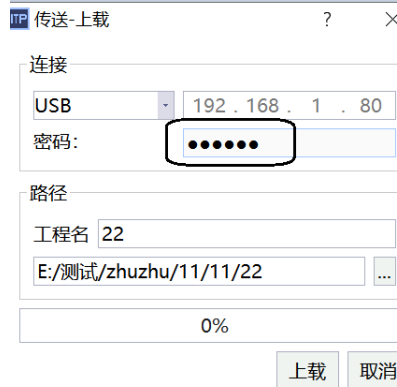
上载历史密码

旧密码：

新密码：

确认密码：

应用



■ 方法二：InoTouchPad软件修改

1. 在InoTouchPad软件界面左侧“工程” > “HMI设置” > “工程设置”中的“安全设置”栏中，即可修改“上载密码”，默认为空，如图所示。



2. 修改完成后下载工程即可生效。再次上载工程时，需要输入密码才可上载。

● 下载密码的修改方式

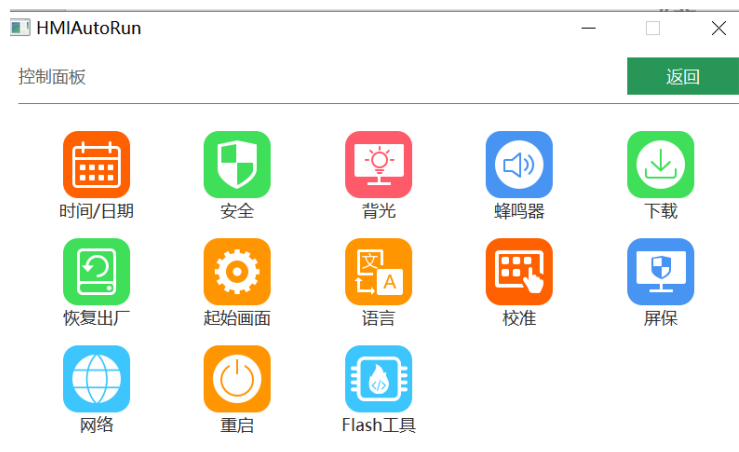
下载密码：默认下载HMI程序不需要下载密码，可直接下载。在设置下载密码后，需要输入正确的下载密码，才能下载工程。

■ 方法一：控制面板修改

1. 进入触摸屏的控制面板界面，输入初始密码111111。



2. 点击进入安全设置界面，如图所示。

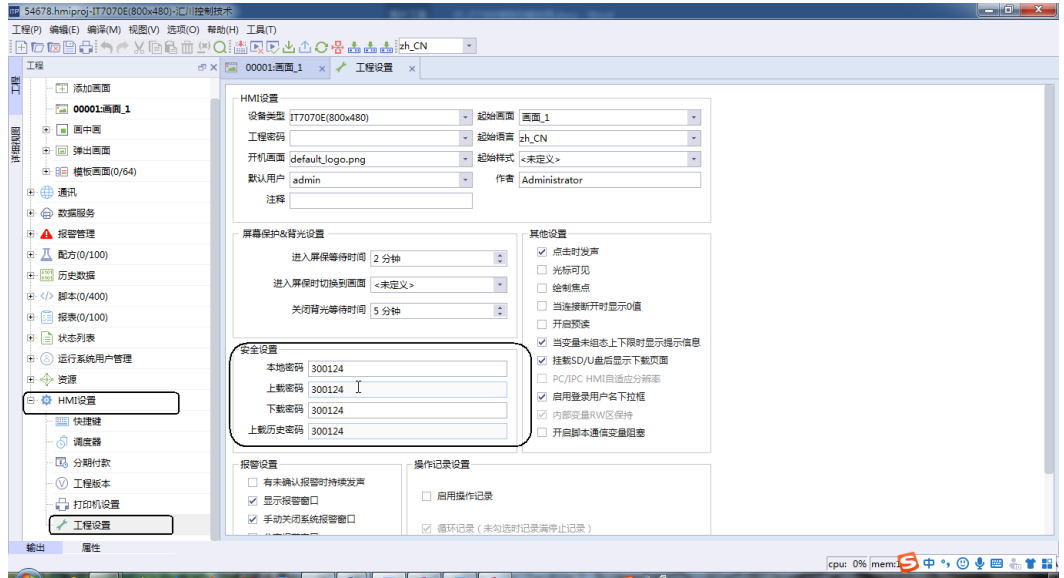


3. 在下载密码界面，先输入之前的密码（初次为空），再输入新密码并确认，点击应用，便完成了下载密码的修改，再次下载工程时，需要输入密码才可下载。



- 方法二：InoTouchPad软件修改

1. 在InoTouchPad软件界面左侧“工程”>“HMI设置”>“工程设置”中的“安全设置”栏中，即可修改“下载密码”，默认为空，如图所示。



2. 修改完成后下载工程即可生效。再次下载工程时，需要输入密码才可下载。

## 7.6 自动翻译

### 案例要求

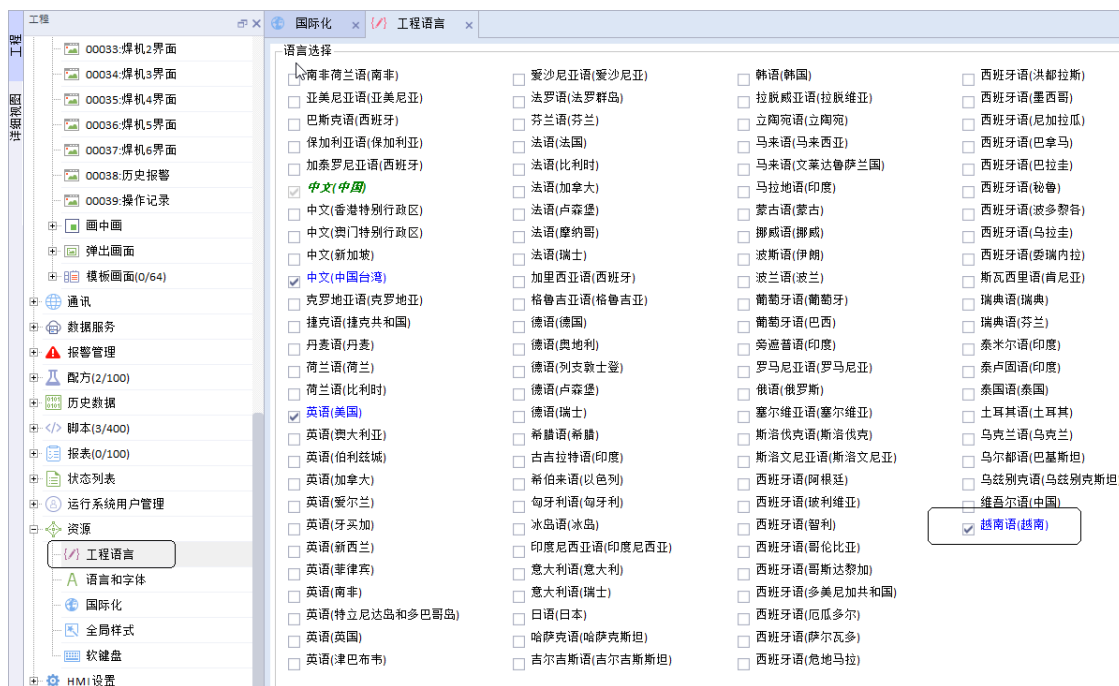
实现IT7000自动翻译成越南语功能。

### 案例目的

学习使用IT7000自动翻译。

### 编程步骤

1. 点击工程资源工程语言勾选越南语。

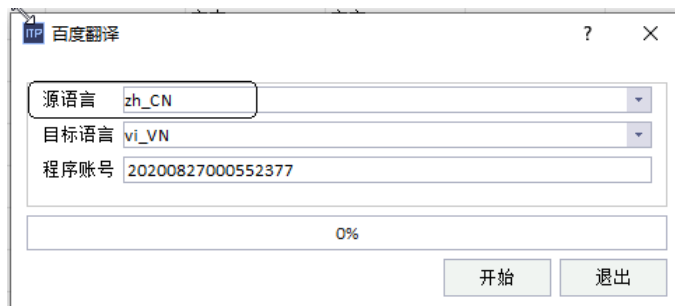
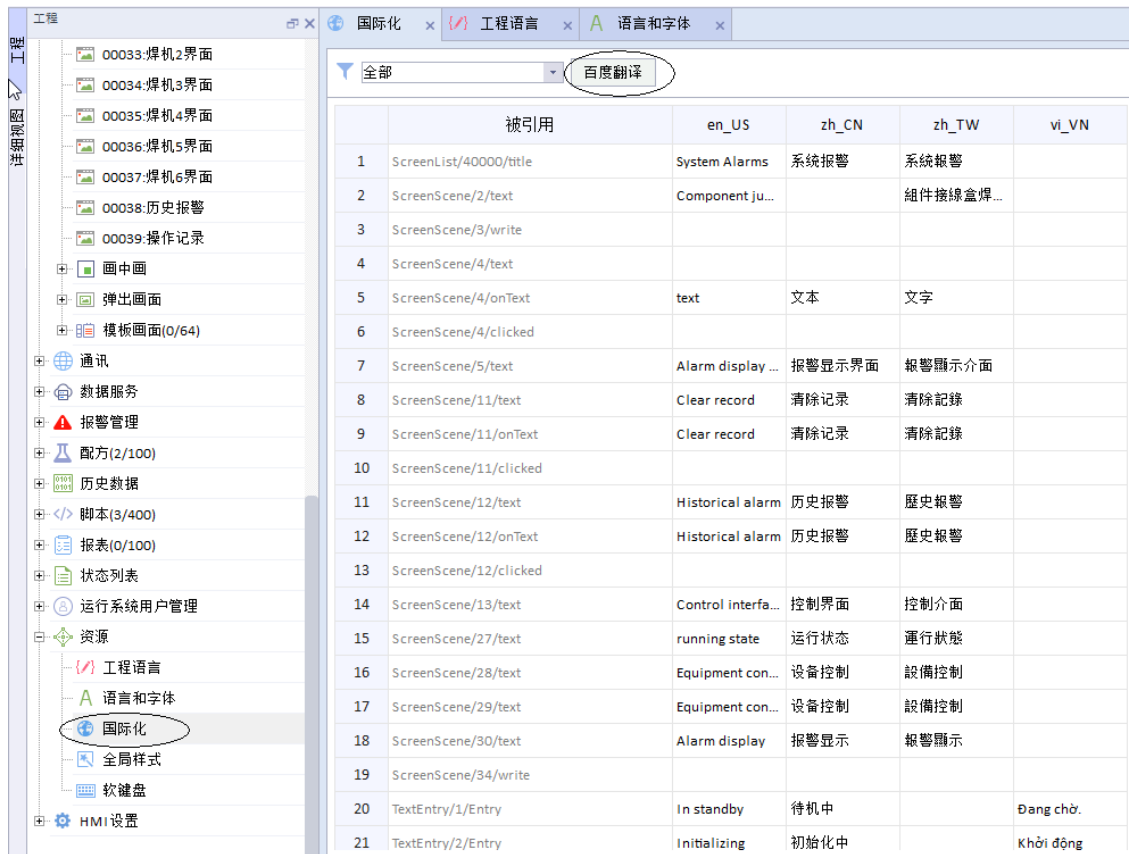


2. 在资源语言和字体越南语，选择支持越南语的字体如Calibri。

启用	名称	显示名称	编号	字体	注释
<input checked="" type="checkbox"/>	英语(美国)	en_US	1	Arial,16px	...
<input checked="" type="checkbox"/>	中文(中国)	zh_CN	2	Droid Sans Fallback,16px	...
<input checked="" type="checkbox"/>	中文(中国台湾)	zh_TW	3	Droid Sans Fallback,16px	...
<input checked="" type="checkbox"/>	越南语(越南)	vi_VN	4	Calibri,16px	...

3. 将源语言翻译为目标语言。

点击界面左侧“工程”>“资源”>“国际化”，选择百度翻译，点击开始即可翻译。源语言建议使用中文，否则翻译结果可能会有偏差。目标语言选择越南语语言。翻译过程中不要修改源语言以及目标语言，否则可能出现翻译错乱，若少量翻译结果出现有错误或者偏差，也可以作手动翻译，复制粘贴。

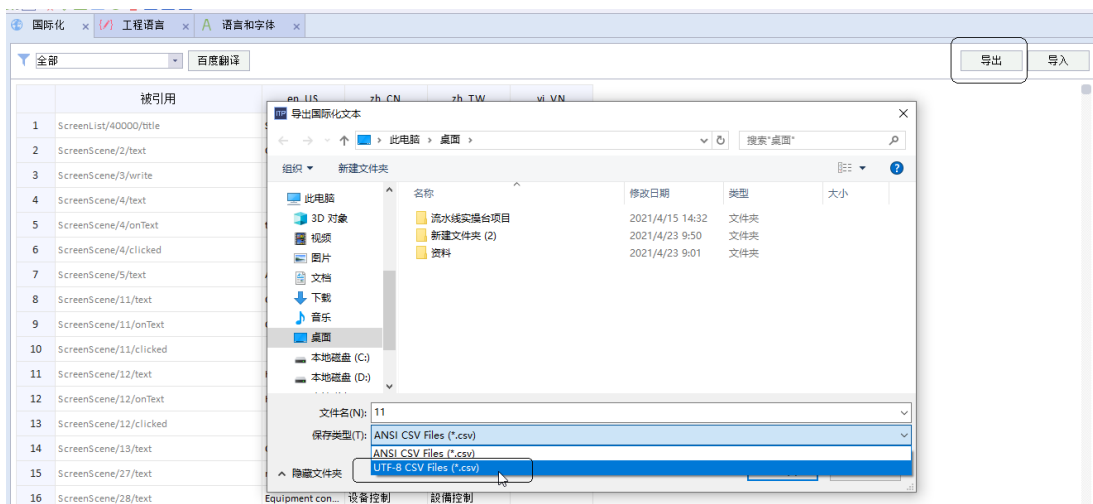


4. 翻译效果如图所示。

被引用	en_US	zh_CN	zh_TW	vi_VN	
1	ScreenList/40000/title	System ...	系統報警	系統報警	Bảo đ ng h th ng
2	ScreenList/40001/title	Commu...	通讯报警	通訊報警	Báo đ ng liên l c
3	ScreenScene/2/write				
4	ScreenScene/7/write				
5	ScreenScene/6/write				
6	ScreenScene/5/write				
7	ScreenScene/4/write				
8	ScreenScene/3/write				
9	Groups/1/DisplayName	Operat...	操作员组	操作員組	Nhóm ng i đi u khi n
10	Groups/2/DisplayName	Admin ...	管理员组	管理員組	Nhóm qu n lý
11	AlarmClasses/1/Display...	!	!	!	Không!
12	AlarmClasses/2/Display...	#	#	#	35;
13	AlarmClasses/3/Display...	\$	\$	\$	Comment
14	Users/1/DisplayName				
15	ScreenScene/8/text		按钮		Nút
16	ScreenScene/8/onText		文本		ch
17	ScreenScene/8/clicked				

5. 备注。

- a. 如果目标语言越南语vi\_VN这一栏之前已经翻译完成或者出现乱码，在使用百度翻译前需要将其清空在进行翻译，否则不会翻译。这里IT7000暂不支持批量清空，如果变量过多需要批量删除可以先导出成UTF8 CSV Files (\*.csv) 格式文件再进行批量删除，如图所示（注：导出编写文件名保持类型只能选择UTF8 CSV Files (\*.csv) 否则导出结果将出现乱码）。

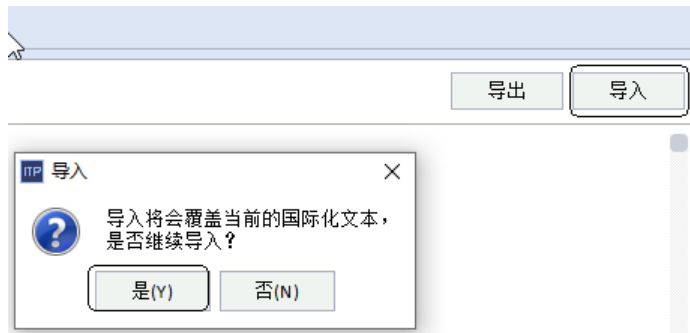
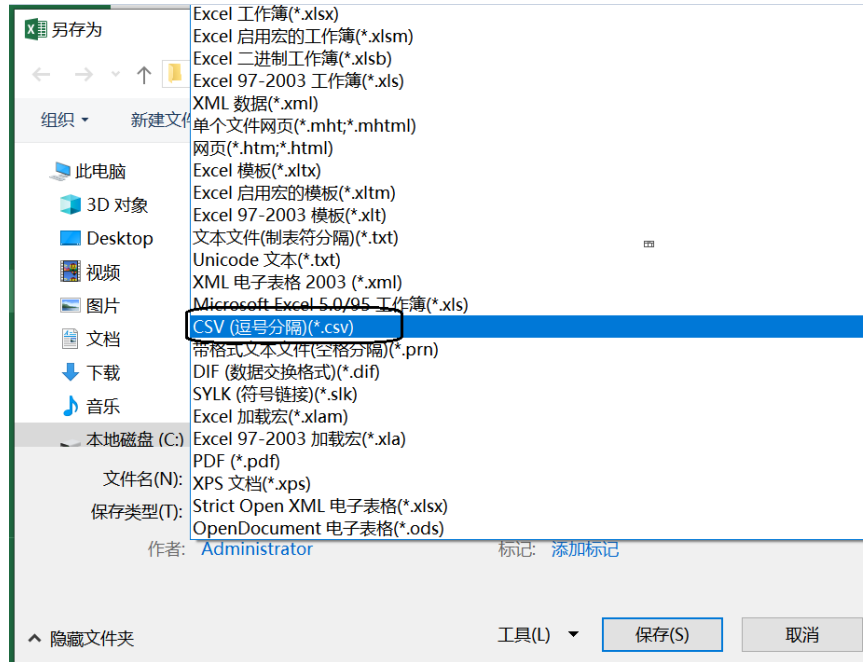
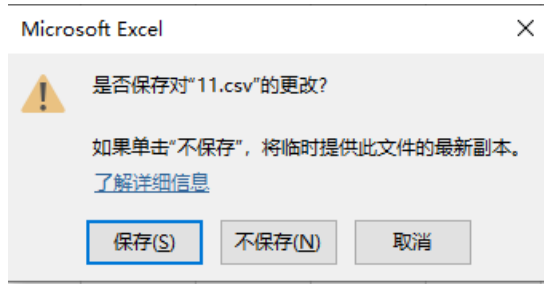


- b. 批量框选全部变量名删除键删除(注意：不能将被引用目标语言名vi\_VN删除)。

1	国际化导出文件 (汇川)				
2	#2021-08-04 10:59:00				
3					
4	被引用	en_US	zh_CN	zh_TW	vi_VN
5	ScreenList/System Ala	系统报警	系统报警	系統報警	Báo động hệ thống
6	ScreenList/Communic	通讯报警	通讯报警	通訊報警	Báo động liên lạc
7	ScreenScene/2/write				
8	ScreenScene/7/write				
9	ScreenScene/6/write				
10	ScreenScene/5/write				
11	ScreenScene/4/write				
12	ScreenScene/3/write				
13	Groups/1/Operator g	操作员组	操作員組	Nhóm người điều khiển	
14	Groups/2/Admin gro	管理员组	管理員組	Nhóm quản lý	
15	AlarmClass!	!	!	Không!	
16	AlarmClass#	#	#	35;	
17	AlarmClass\$	\$	\$	Comment	
18	Users/1/DisplayName				
19	ScreenScene/8/text	按钮		Nút	
20	ScreenScene/8/onTex	文本		chữ	
21	ScreenScene/8/clicked				

1	国际化导出文件 (汇川)				
2	#2021-08-04 10:59:00				
3					
4	被引用	en_US	zh_CN	zh_TW	vi_VN
5	ScreenList/System Ala	系统报警	系統報警		
6	ScreenList/Communic	通讯报警	通訊報警		
7	ScreenScene/2/write				
8	ScreenScene/7/write				
9	ScreenScene/6/write				
10	ScreenScene/5/write				
11	ScreenScene/4/write				
12	ScreenScene/3/write				
13	Groups/1/Operator g	操作员组	操作員組		
14	Groups/2/Admin gro	管理员组	管理員組		
15	AlarmClass!	!	!		
16	AlarmClass#	#	#		
17	AlarmClass\$	\$	\$		
18	Users/1/DisplayName				
19	ScreenScene/8/text	按钮			
20	ScreenScene/8/onTex	文本			
21	ScreenScene/8/clicked				

- c. 删除完成点击保存(保存为.csv格式)，再进行导入覆盖，即可将之前翻译的进行批量清除。



全部 百度翻译

	被引用	en_US	zh_CN	zh_TW	vi_VN
1	ScreenList/40000/title	System ...	系统报警	系統報警	
2	ScreenList/40001/title	Commu...	通讯报警	通訊報警	
3	ScreenScene/2/write				
4	ScreenScene/7/write				
5	ScreenScene/6/write				
6	ScreenScene/5/write				
7	ScreenScene/4/write				
8	ScreenScene/3/write				
9	Groups/1/DisplayName	Operat...	操作员组	操作員組	
10	Groups/2/DisplayName	Admin ...	管理员组	管理員組	
11	AlarmClasses/1/Display...	!	!	!	
12	AlarmClasses/2/Display...	#	#	#	
13	AlarmClasses/3/Display...	\$	\$	\$	
14	Users/1/DisplayName				
15	ScreenScene/8/text		按钮		
16	ScreenScene/8/onText		文本		
17	ScreenScene/8/clicked				

描述

17条数据导入成功, 0条数据导入失败。

## 7.7 多语言字体独立设置

### 案例要求

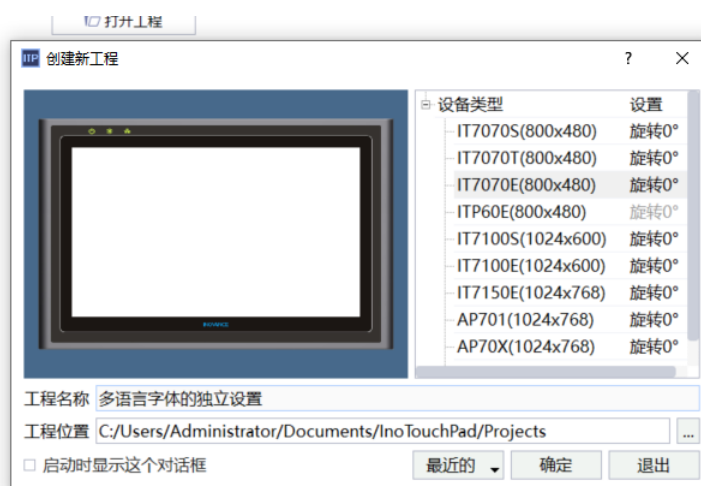
实现同一个工程中实现多种语言不同字体显示。

### 案例目的

通过案例学会在同一个工程中实现多种语言不同字体显示。

### 编程步骤

1. 用汇川IT7000编程软件InoTouchPad新建一个工程。

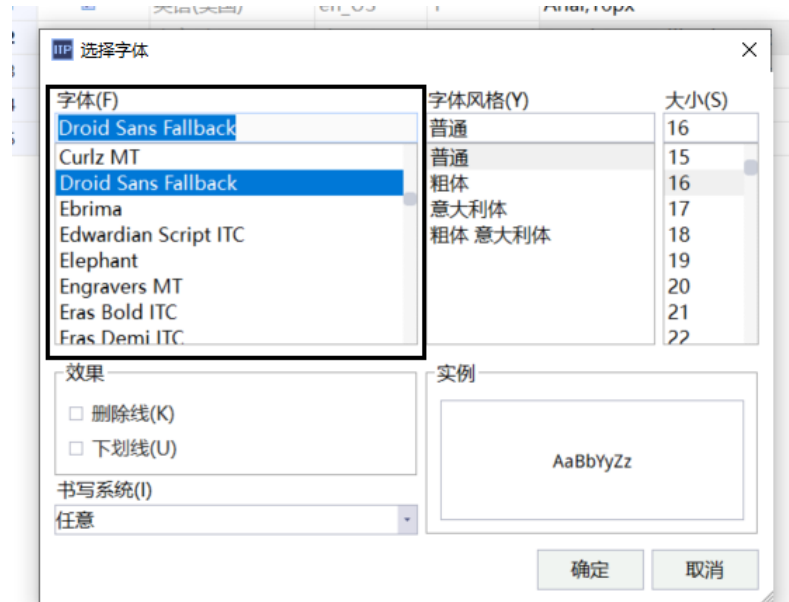


2. 打开界面左侧“工程” > “资源” > “语言和字体”，单击字体栏中的“...”即可对不同的语言设置字体。



	启用	名称	显示名称	编号	字体	注释
1	<input checked="" type="checkbox"/>	英语(美国)	en_US	1	Arial,16px	...
2	<input checked="" type="checkbox"/>	中文(中国)	zh_CN	2	Droid Sans Fallback,16px	...
3	<input checked="" type="checkbox"/>	中文(中国台湾)	zh_TW	3	Droid Sans Fallback,16px	...
4	<input type="checkbox"/>	韩语(韩国)	ko_KR	4	Droid Sans Fallback,16px	...
5	<input type="checkbox"/>	日语(日本)	ja_JP	5	Droid Sans Fallback,16px	...

3. 语言字体设置框如下图所示，可根据需要对每一种语言进行设置。



## 7.8 交叉引用功能的应用

### 案例要求

通过此案例了解交叉引用功能的使用方法，学会使用交叉引用对变量进行索引。

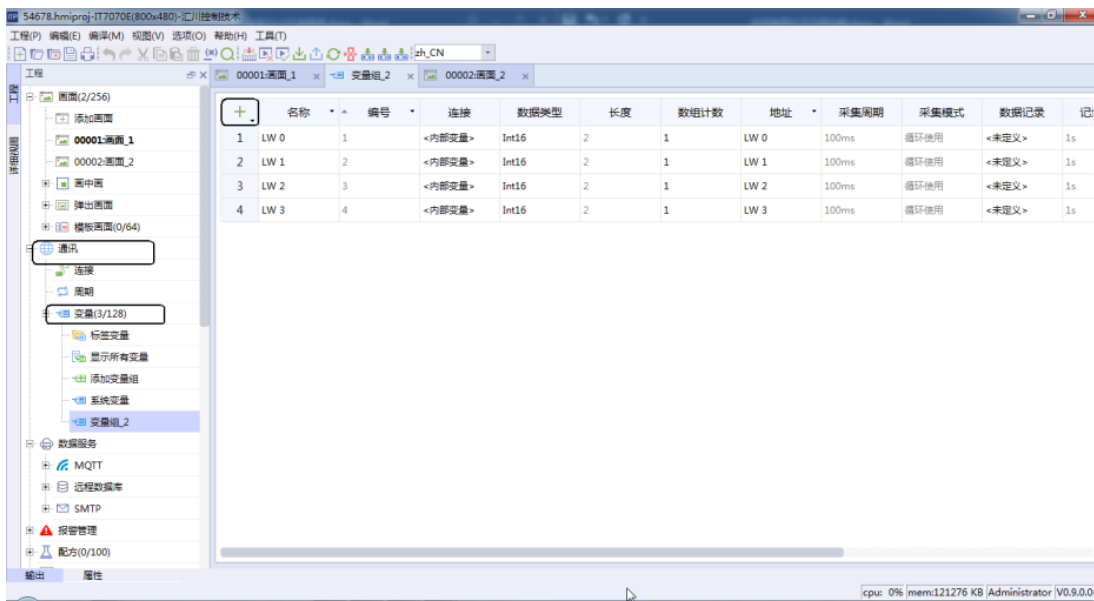
### 案例目的

通过此案例学会使用交叉引用功能查找变量的引用范围和查找变量所关联的控件。

### 编程步骤

1. 建立变量，将其组态在几个画面中。

- a. 在软件界面“工程” > “通讯” > “变量” > “添加变量组”，点击左上角“+”号即可新建变量。



b. 添加画面。在软件界面左侧“工程”>“画面”>“添加画面”即可进行画面添加。



c. 将添加好的变量组态在画面中，可通过界面右侧“工具”>“简单控件”选择数值IO域，在“常规”>“过程”中关联变量，也可直接在画面中单击左侧新建的变量组进行变量拖拽组态。



2. 使用交叉引用对变量进行索引。

a. 在变量组页面中，鼠标左键单击对应变量的序号，再右键即可选择“交叉引用”。



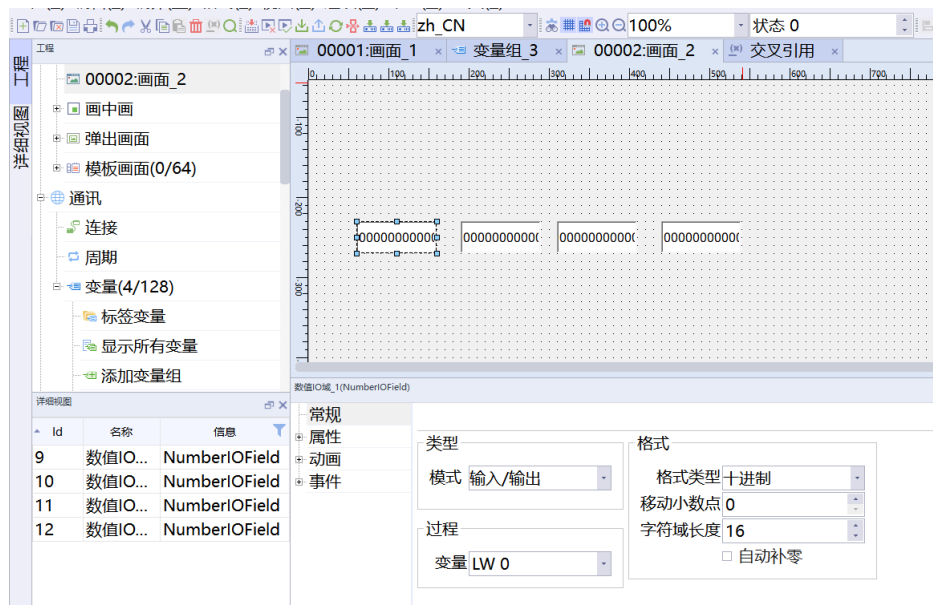
或者选中变量后，点击交叉引用快捷键。



b. 选择“交叉引用”后，即可弹出“交叉引用”页面，页面中包含所选中的变量信息，包括“名称”、“属性名”、“路径”、“信息文本”和“注释”。



c. 双击所要查看想变量信息即可索引至变量所使用的地方，可通过索引变量查看某个变量关联的控件和控件使用的具体位置。



## 7.9 如何查看HMI固件版本和升级HMI固件

### 案例要求

通过HMI软件完成对固件的查看与升级：

- 学会查看固件版本。
- 利用软件对HMI进行固件升级。

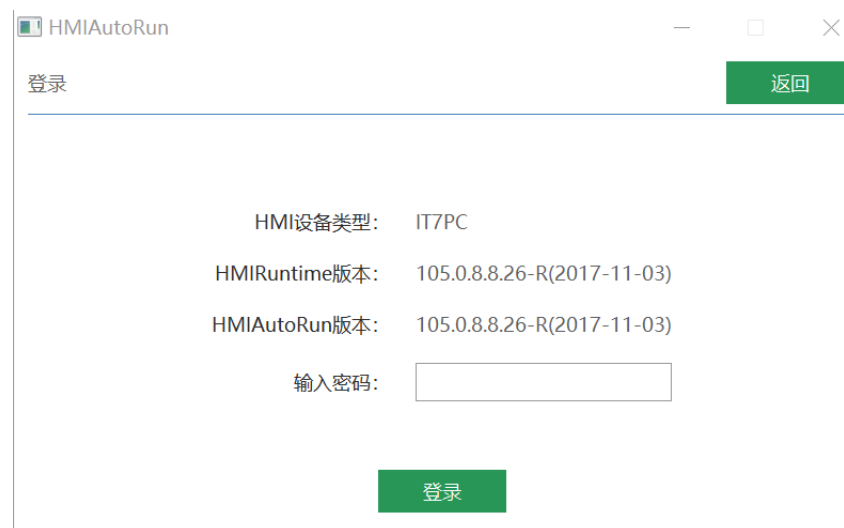
### 案例目的

通过此案例学会查看固件版本以及利用软件对HMI进行固件升级。

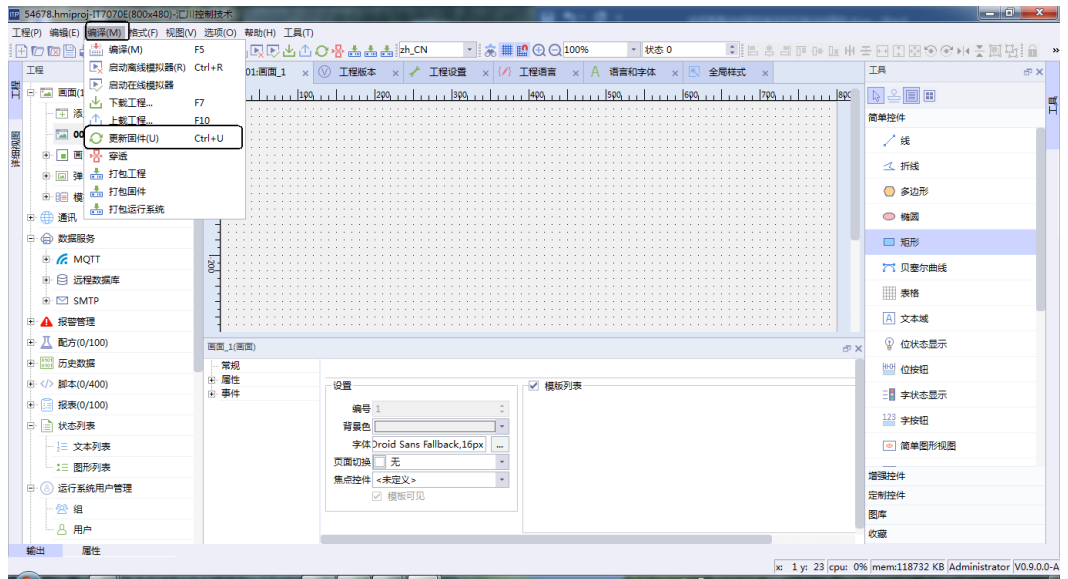
## 编程步骤

- 查看固件版本  
IT7000触摸屏上电后，根据提示长按屏幕进入后台，即可查看到“HMIRuntime版本”号。

长按进入设置面板



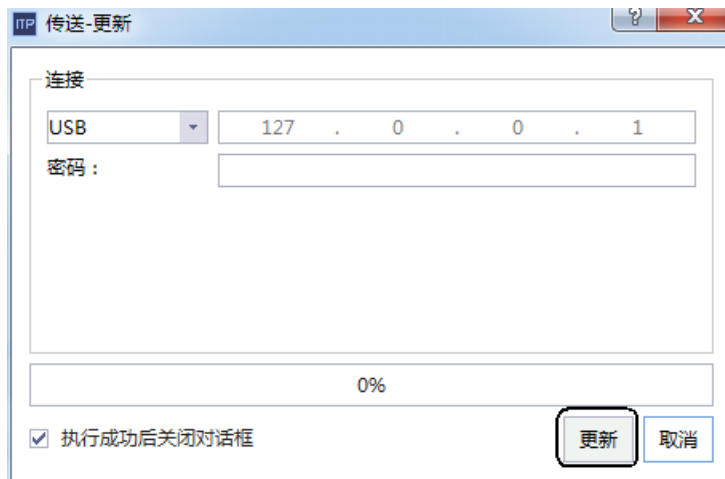
- 利用InoTouchPad软件进行固件升级
  1. 更新固件的操作按钮有以下两种：
    - 第一种：点击菜单栏中的“编译”，选择“更新固件”。



- 第二种：电机快捷编辑栏中的“更新固件”按钮快捷键。



2. 后续弹出的对话框内点击“更新”，等待更新完成即可。



## 7.10 使用U盘或SD卡升级PLC程序

### 案例要求

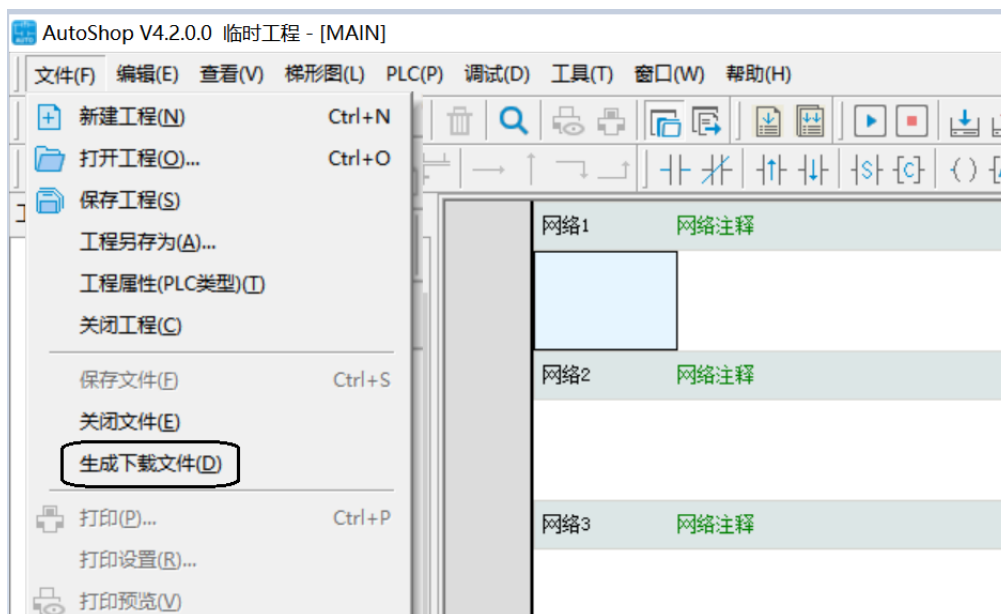
学会如何使用U盘或SD卡升级PLC程序。

### 案例目的

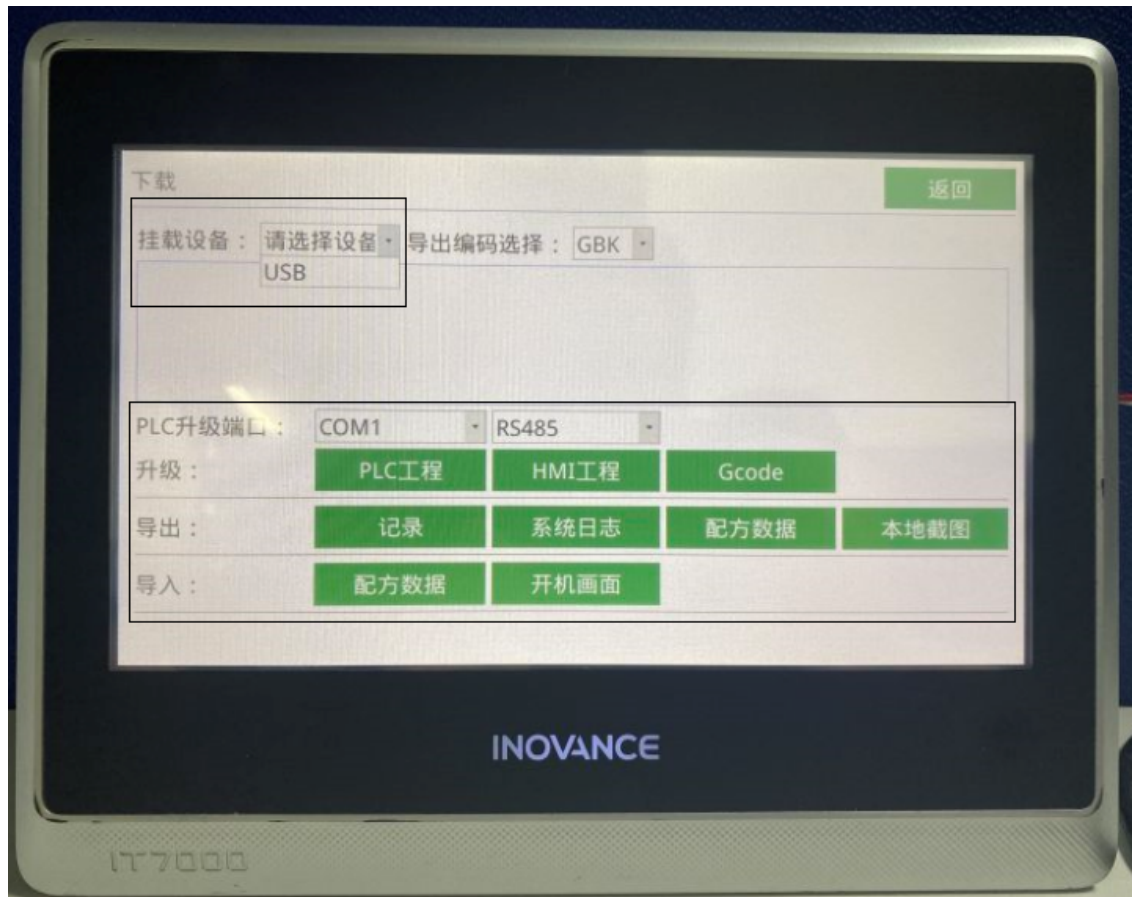
通过此案例学会使用U盘或者SD卡升级PLC程序。

## 编程步骤

1. 在AutoShop中的工具栏“文件” > “生成下载文件”创建好程序文件，接着将生成文件拷贝到U盘或者SD卡中。



2. 将U盘或者SD卡插入屏幕中，会自动进入后台，初始密码是111111，进入后选择选择已经打包好的程序，点击“升级” > “PLC工程”即可。



## 7.11 通过多语言功能切换工程中英文

### 案例要求

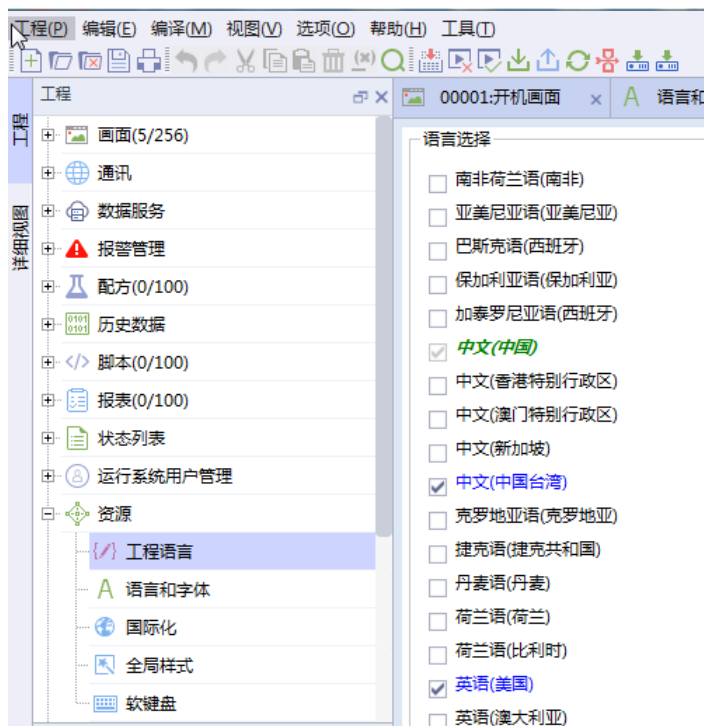
1. 通过多语言来实现工程中英文切换。
2. 工程语言添加。
3. 语言和字体设置。
4. 添加事件切换HMI设备上的语言国际化多语言显示切换。

### 案例目的

- 通过此案例了解多语言添加和字体的设置方法。
- 通过此案例了解多语言功能，学会掌握多语言切换的使用方法。

### 编程步骤

1. 工程语言的添加和字体设置的方法
  - a. 工程语言选择。在软件界面左侧“工程”>“资源”>“工程语言”>“语言选择”里勾选所需要的语言，勾选表示激活了该国国家的语言。



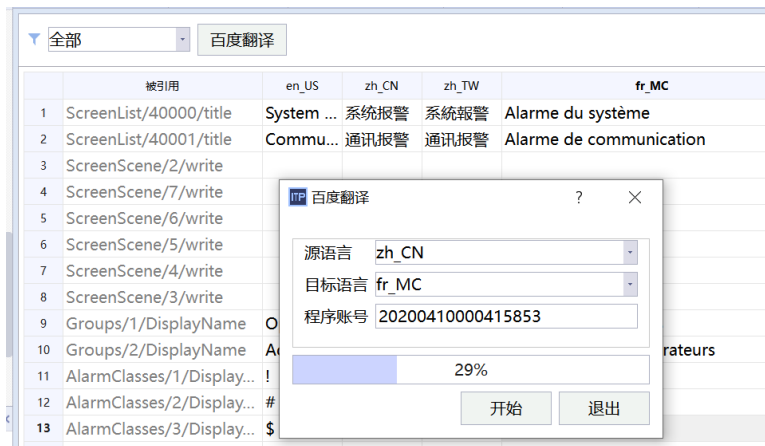
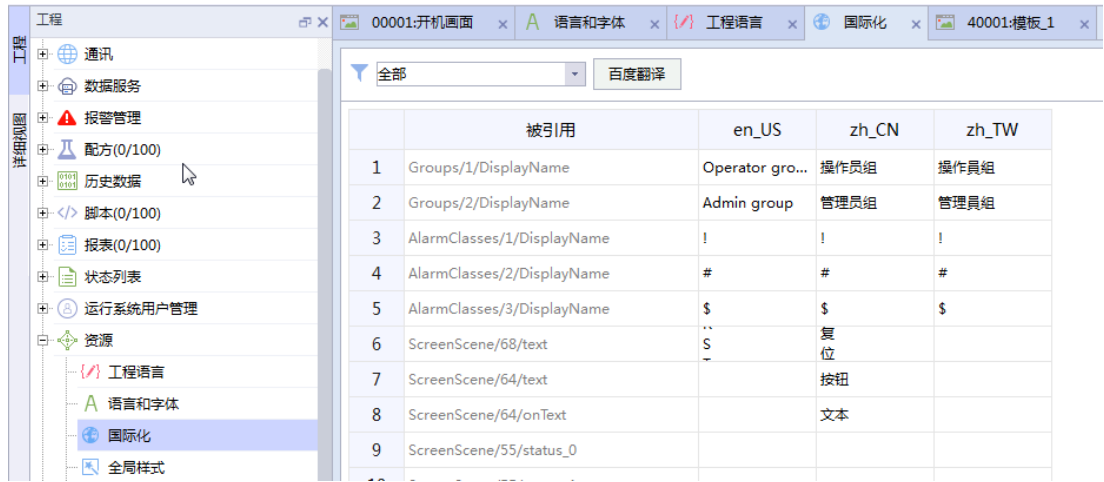
b. 语言和字体的设置。在软件界面左侧“工程”>“资源”>“语言和字体”里启用所需要的语言(在所需语言的启用栏打√)，并对相应语言的字体进行设置。



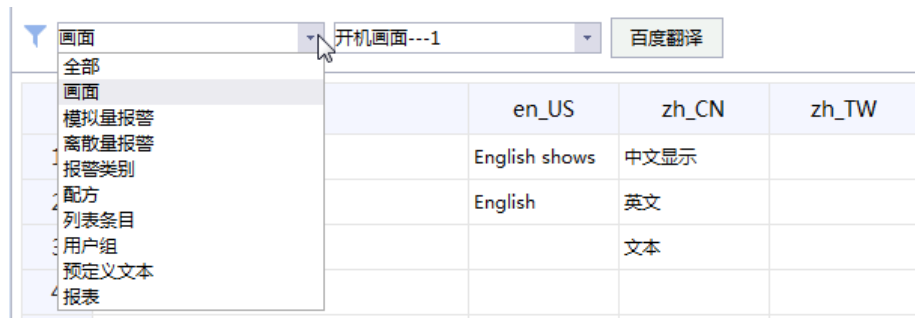
## 2. 添加多语言信息

a. 国际化添加多语言信息。在软件界面左侧“工程”>“资源”>“国际化”，双击进入国际化工作区，该工作区可显示工程中需要翻译的所有文本，可进行翻译的语言由资源中的“工程语言”设定，双击百度翻译可实现源语言到目标语言的联网翻译，也可自行手动输入翻译的目标内容。

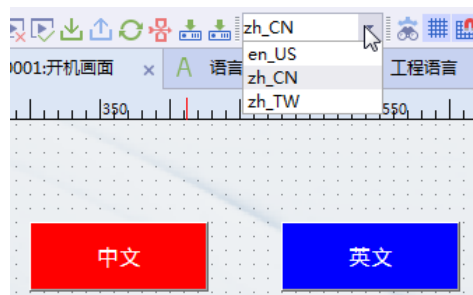


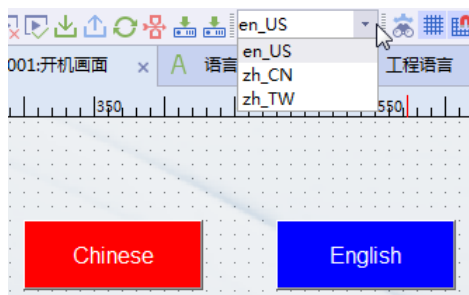


b. 国际化筛选器。在国际化编辑区的左上角有个筛选器，对工程中所有可以进行翻译的文本进行了分类，具体有全部、画面、模拟量报警、离散量报警、配方、列表条目、用户组、预定义文本、报表，可根据需求进行分类翻译，如下图所示。



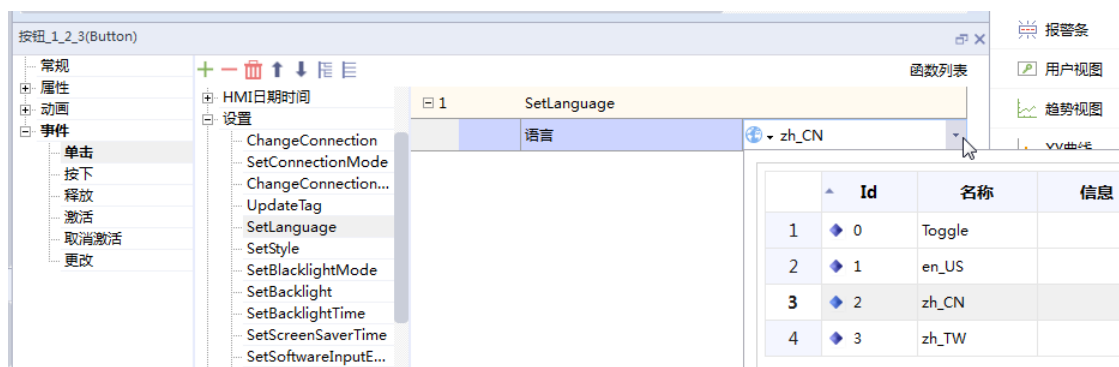
c. 国际化切换预览：“顶部工具栏”点击语言显示名称“en\_US”实现国际化语言切换预览，不同语言预览状态下，可通过控件属性添加文本语言和字体。





### 3. 添加中英文切换事件

在软件界面右侧“工具”>“简单控件”>“按钮”添加按钮控件用于实现多语言切换，单击按钮，在“事件”>“单击”>“设置”>“SetLanguage”添加SetLanguage事件，设置对应要切换的语言。

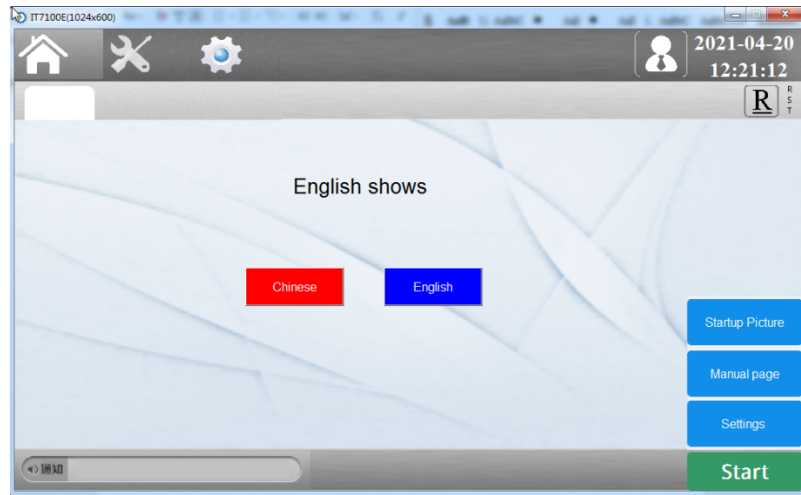


### 4. 通过多语言实现工程中英文切换的效果

中文效果图如下：



英文效果图如下：



## 8 权限管理

### 8.1 用户登录权限管理

用户视图功能说明：“用户视图”是在HMI中管理用户的编辑窗口。拥有“管理”权限的用户可以在“用户视图”中管理所有用户，其它受限用户则只能管理自己。

#### 案例要求

通过此案例学会：

- 在HMI中组态并使用用户视图。
- 组态用户登录/注销系统函数。
- 用户管理应用及权限设置。

#### 案例目的

学习使用IT7000用户登录权限。

#### 编程步骤

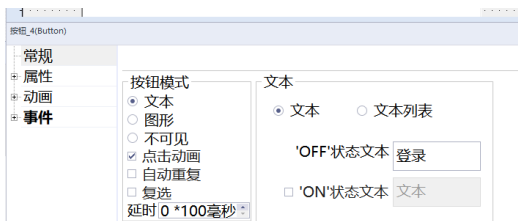
##### 1. 组态用户视图

在软件界面左侧“工程”>“画面”>“添加画面”中双击添加一个画面，在界面右侧“工具”>“增强控件”中单击“用户视图”，将其拖拽至画面中，完成用户视图的组态。

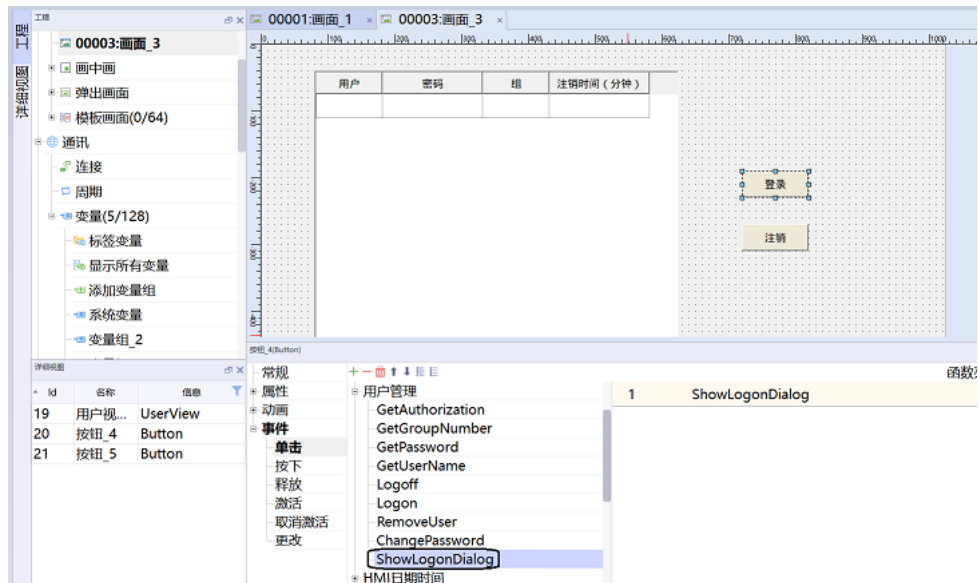


##### 2. 组态用户登录/注销系统函数

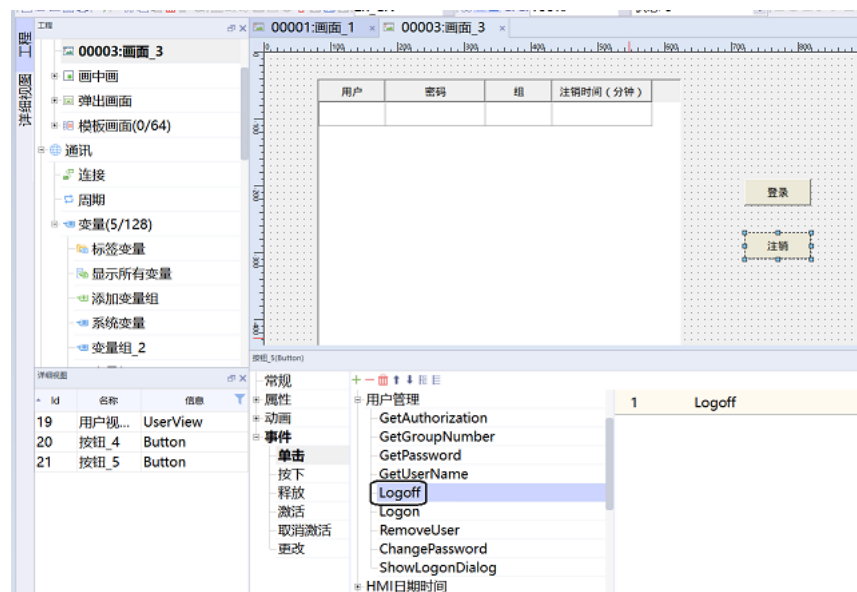
- 在软件右侧“工具”>“简单控件”中将“按钮”拖拽至画面中，分别重复两次，组态一个登陆按钮和注销按钮，按钮的名字可在“事件”>“文本”>“‘OFF’状态文本”中进行更改。



- b. 在工程画面中单击“登录”按钮，在“事件”>“单击”>“用户管理”中，双击函数“ShowLogonDialog”，即可将其与“登陆”按钮关联。上述操作可实现在运行时单击“登陆”按钮，即可执行组态的系统函数的显示登录对话框功能。

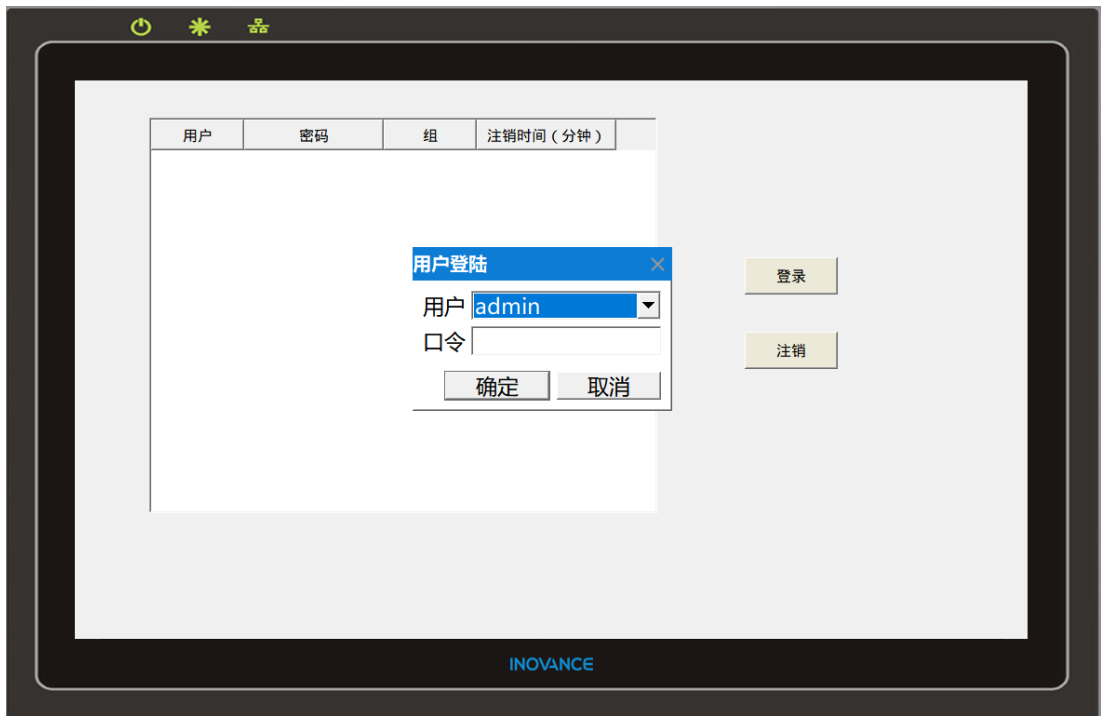


- c. 在工程画面中单击“注销”按钮，在“事件”>“单击”>“用户管理”中，双击“Logoff”，即可将其与“注销”按钮关联。上述操作可实现在运行时单击“注销”按钮，即可执行组态的系统函数的注销当前用户功能。



### 3. 用户视图使用

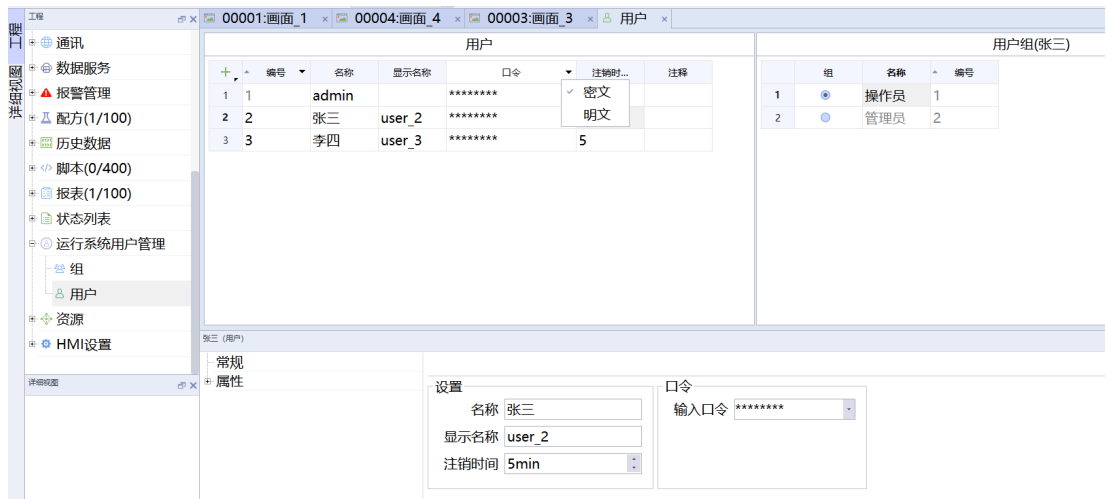
- a. 组态用户管理后，编译并下载工程至HMI中或者通过在线/离线模拟。点击“登陆”按钮，在弹出的“用户登陆”对话框中选择指定用户登陆(初始密码为111111)，登陆后即可显示用户信息（注：“用户视图”初始化时是没有任何数据的）。



下图以“admin”身份为例：



- b. 当具有管理权限的用户登陆后，会显示所有用户信息（用户管理及组权限设置请参见第152页“8.2 通过用户权限登录控制按钮”）。在界面左侧“工程”>“运行系统用户管理”>“用户”处点击左上角“+”号添加新用户，修改用户名称为张三，李四。口令可选择为密文或者明文显示。



c. 模拟调试，点击登录效果如下。



#### 4. 用户管理应用及权限设置

a. 在软件界面左侧“工具”>“简单控件”里添加一个按钮、一个文本域和一个数值IO域至画面中，单击按钮，在“常规”>“文本”>“OFF状态文本”中将文本改为“参数设置”，单击文本域，在“常规”>“文本”中将文本域内容更改为“温度设定”。



- b. 设置“参数设置”按钮和“数值IO域”控件的权限。单击选中“参数设置”按钮，在“属性”>“安全”>“运行系统安全”>“权限”输入框点击下拉按钮，在下拉框中选择“管理”权限，按同样操作将“数值IO域”控件的权限设置为“操作”权限。



- c. 在界面左侧“工程”>“运行系统用户管理”>“用户”里单击选择张三，将其用户组权限更改为管理员。通过此操作，若用户选择具有管理权限的张三进行登录后，可直接访问具有管理权限的“参数设置”或具有操作权限的“数值IO域”而不用再次登录。若用户选择了只具有操作权限的李四进行登录后，则只能对“数值IO域”进行访问，而在访问“参数设置”时会受到阻碍需要再次进行登录。

用户						用户组(张三)				
+	-	编号	名称	显示名称	口令	注销时...	注释	组	名称	编号
		1	admin		*****	5		1	操作员	1
		2	张三	user_2	*****	5		2	管理员	2
		3	李四	user_3	*****	5				

## 5. 仿真模拟

点击在线/离线模拟或将工程下载至HMI后，即可进行操作。

- a. 若用户未登陆或未经授权而点击组态了权限的控件，系统会弹出“用户登陆”对话框来控制访问权限。



- b. 对于“参数设置”按钮和“温度设定”数值IO域，以“张三”身份进行登录后，点击两个控件系统不会再弹出“登陆”对话框。





c. 点击“注销”按钮，再以“李四”的身份登陆。则可成功修改温度值，但点击“参数设置”按钮时，会再次弹出“用户登陆”对话框，使其访问受阻。



## 8.2 通过用户权限登录控制按钮

### 案例要求

- 通过用户权限登录控制按钮控件。
- 如何添加用户和管理组。
- 如何在线修改用户信息、管理组权限。
- 如何设置控件安全登录权限。

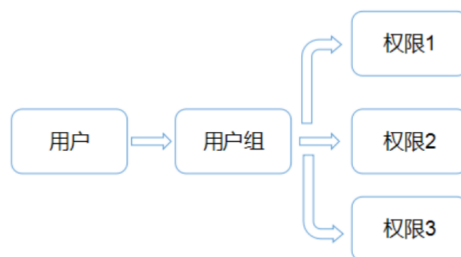
## 案例目的

- 通过此案例了解用户管理的基本原理。
- 通过此案例学会用户管理的元素和基本设置。
- 通过此案例掌握用户视图在用户管理中的应用。
- 通过此案例掌握控件安全权限的设置。

## 编程步骤

### 1. 用户管理的基本原理

在用户管理中，权限不会直接分配给用户，而是分配给用户组。例如，名为“李四”的用户分配给“调试工程师”用户组并被授予相应权限。不必为每个用户分别分配权限，而只需将权限分配给用户组即可。用户、用户组和用户权限三者之间的从属关系如下图所示。



### 2. 用户管理的基本设置

- a. 添加用户权限组：软件界面左侧“工程” > “运行系统用户管理” > “组” > “组权限” 新增“系统维护”组权限。

组权限(操作员)				
+	<input type="checkbox"/>	名称	▲ 编号 ▼	注释
1	<input type="checkbox"/>	管理	1	
2	<input type="checkbox"/>	监视	2	
3	<input checked="" type="checkbox"/>	操作	3	
4	<input type="checkbox"/>	系统维护	4	

- b. 添加用户组：软件界面左侧“工程” > “运行系统用户管理” > “组” 添加新的用户组，并对每一个用户组关联对应的组权限，其中管理员组权限最大。

组					组权限(车间主任)				
—	名称	显示名称	▲ 编号 ▼	注释	+	<input type="checkbox"/>	名称	▲ 编号 ▼	注释
1	操作员	操作员组	1		1	<input checked="" type="checkbox"/>	管理	1	
2	管理员	管理员组	2		2	<input checked="" type="checkbox"/>	监视	2	
3	车间主任	车间主任	3		3	<input checked="" type="checkbox"/>	操作	3	
					4	<input type="checkbox"/>	系统维护	4	

- c. 添加用户：软件界面左侧“工程” > “运行系统用户管理” > “用户” 添加新的用户，并对每一个用户关联对应的用户组。

用户						用户组(操作员B)						
—	▲	▼	编号	名称	显示名称	口令	注销时间(分钟)	组	名称	▲	▼	编号
1			1	系统管理员		*****	5	1	●	操作员		1
2			2	车间主任		*****	5	2	●	管理员		2
3			3	操作员A		*****	5	3	●	车间主任		3
4			4	操作员B		*****	5					

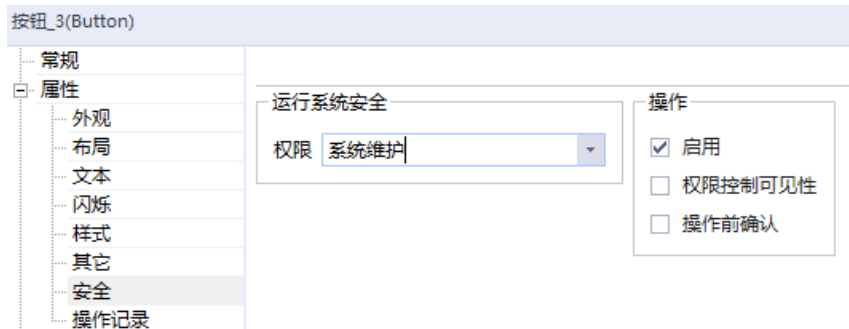
通过用户属性窗口设置用户名称、显示名称、注销时间、用户密码信息。



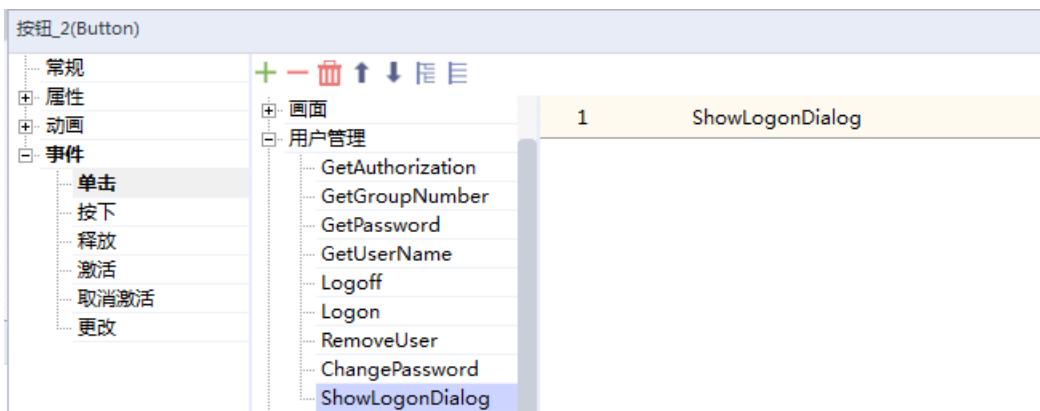
### 3. 基本控件权限的设定方法

主要包括属性视图设置、系统函数ShowLogonDialog/Logoff设置和用户视图三种方式。

- 方法一：通过按钮安全权限设定，按钮属性视图“属性”>“安全”>“运行系统安全”添加用户。



- 方法二：通过用户登录、注销方法，添加两个按钮，分别添加用户登录事件ShowLogonDialog、注销事件Logoff。



- 方法三：通过添加用户视图，右侧工具栏“工具”>“增强控件”>“用户视图”添加用户视图，通过控件拖拽或控件布局调整位置和尺寸。



#### 4. 用户权限登录控制按钮的案例完成

通过按钮实现登录用户权限，效果图如下。



通过登录用户视图，可针对用户进行HMI设定。

The screenshot displays a user management interface. At the top, there is a navigation bar with icons for home, tools, and settings. The date and time are shown as 2021-04-26 22:01:53. A user profile icon is visible in the top right corner. Below the navigation bar, there is a search bar and a '复位' (Reset) button. The main content area features a table with the following data:

	用户	密码	组	注销时间
1	系统管理员	*****	管理员组	5
2	车间主任	*****	车间主任	5
3	操作员A	*****	操作员组	5
4	操作员B	*****	操作员组	5
5	user 5	*****	操作员组	5

At the bottom right, there are four buttons: '开机画面' (Boot screen), '手动页面' (Manual page), '设置页面' (Settings page), and '开始' (Start). A notification bar is visible at the bottom left.

## 9 配方

### 9.1 配方系统函数的应用

配方系统函数功能说明：配方视图是一个通用的增强型画面对象，用于配方的显示、编辑以及配方数据记录的管理，它以表格的形式呈现数据记录。在配方视图中，显示和输入的值可保存在数据记录中，配方数据记录通过系统功能与PLC 进行数据交换。

#### 案例要求

- 学习IT7000配方系统函数和拖拽编程的应用。
- 通过配方将数值写入PLC。
- 从PLC读取数据到配方。

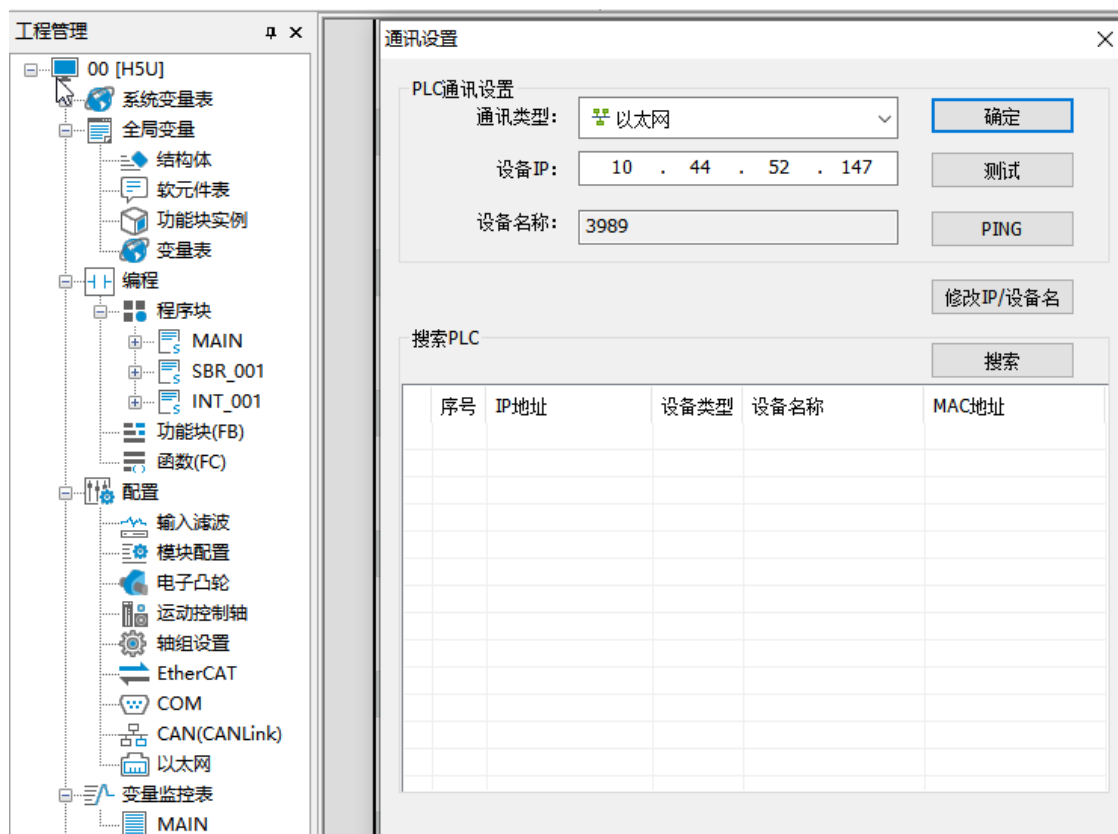
#### 案例目的

通过案例学习使用IT7000配方系统函数和拖拽编程。

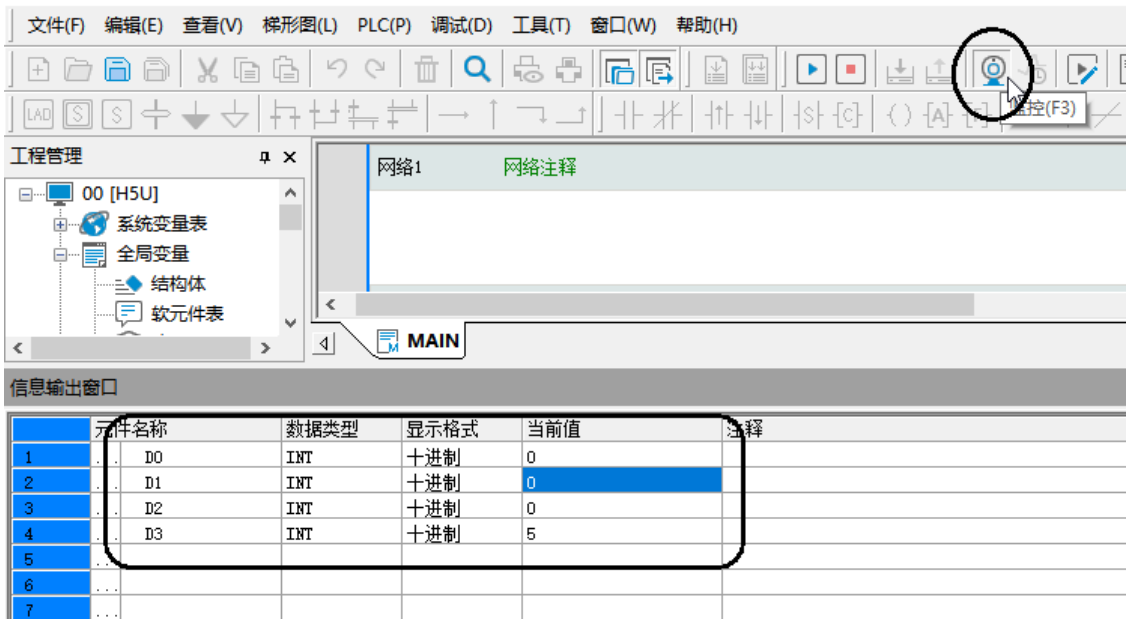
#### 编程步骤

##### 1. 新建PLC工程

打开软件AutoShop软件，新建一个PLC工程，建立H5U通讯连接。

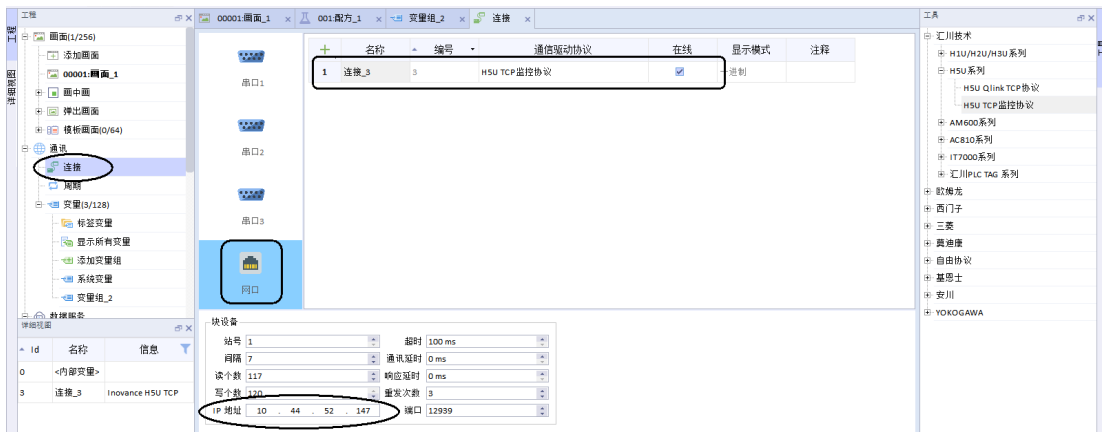


点击监控 (F3)，在信息输出窗口输入要监控的变量如D0~D4。

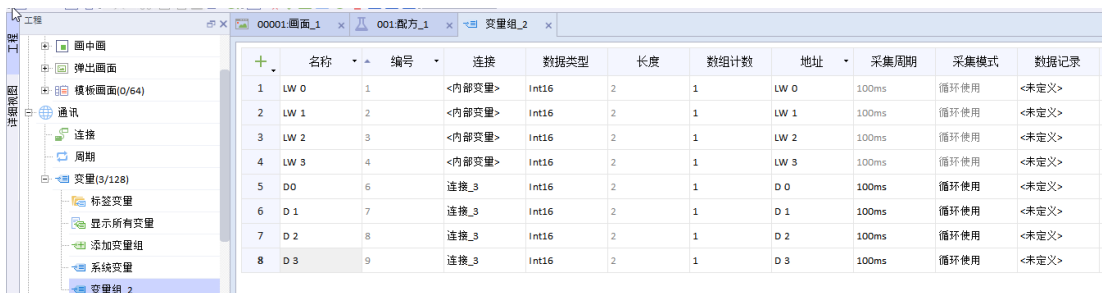


2. IT7000设置

- a. 创建一个工程，在软件界面左侧“通讯”>“连接”>“网口”（所示软件为支持标签通讯版本），点击左上角“+”号添加“汇川技术”>“5U TCP监控协议”，并修改IP地址与H5U一致。



- b. 在界面左侧“通讯”>“变量”>“添加变量组”，点击左上角“+”号新建变量，通过拖拽将其拖入至画面中。



- c. 在界面左侧“配方”>“添加配方”选择左上角“成分”，单击“+”号右下角小箭头批量添加成分并关联变量，通过勾选同步变量和变量离线观察效果。



## 说明

名词解释：

编号：在HMI设备中唯一的标识配方。

显示名称：运行系统中，选择配方时显示的配方名。

同步变量：使配方记录中的值与配方变量的值同步。

变量离线：勾选之后，运行时将断开 HMI 上的变量与对应的 PLC 地址之间的关联。

名称：在配方内部的唯一标识符。

显示名称：HMI 设备运行时，在配方视图中显示的条目名称。

变量：配方成分所对应的 HMI 变量。成分变量为连续地址时这里也可以使用数组变量。

缺省值：配方数据的默认值。

小数点：定义运行时配方数据显示的小数位个数。

信息文本：对相关配方条目的一个描述信息。

### 3. 模拟仿真效果如下

- 只勾选同步变量，不勾选变量离线

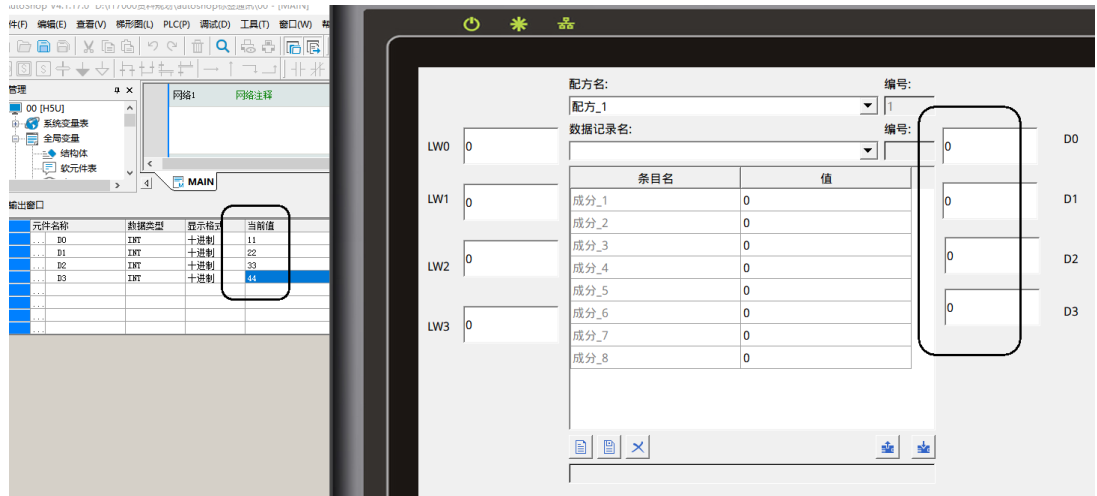
新建数据记录1和数据记录2，在plc监控信息输出窗口中给定D0~D3的数值，会实时同步到HMI。但是，当切换其他数据记录时会直接将其他数据记录的数据写入plc中。



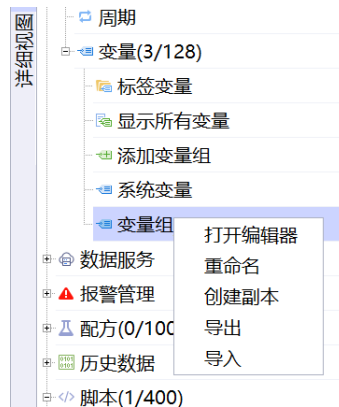
- 同时勾选同步变量和变量离线



新建数据记录1和数据记录2，在plc监控信息输出窗口中给定D0~D3的数值，不会实时同步到HMI。但是，当你切换其他数据记录时会将其其他数据记录的数据写入HMI中，不会写入到plc，同时，plc的值也不会HMI中显示，只有点击上载功能按钮才会将plc的值读到HMI中，达不到实时监控plc数据变化的效果。



因此，在同时使用同步变量和变量离线时，若需要实时监控plc数值的变化情况，建议在 变量变量组右键选择创建副本,建立新的配方成分并与改副本变量进行关联，在仿真模拟时副本变量关联的配方只勾选同步变量即可。



## 9.2 使用U盘或SD卡导入导出数据

### 案例要求

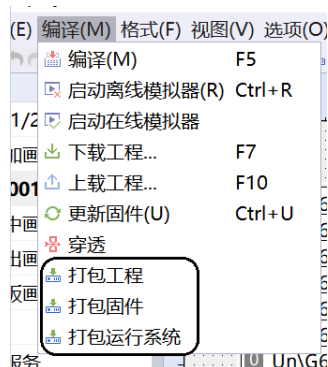
完成用U盘或者SD卡导入程序，升级固件，导入导出配方数据，导出系统日志等。

### 案例目的

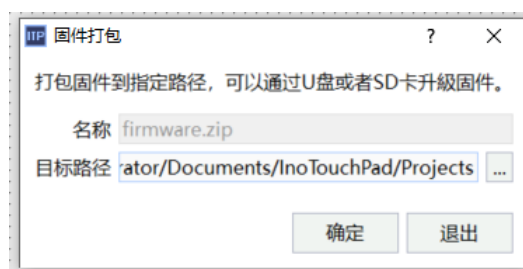
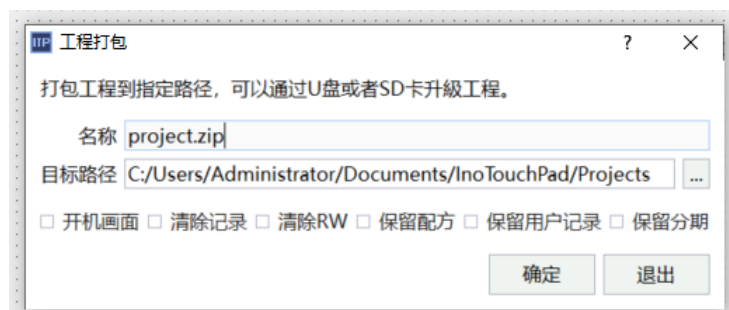
- 通过此案例学会如何使用U盘或者SD卡导入程序。
- 通过此案例学会导入导出配方等数据记录。

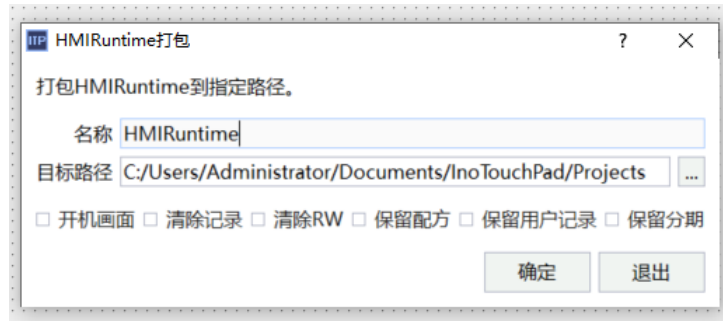
### 编程步骤

1. 在工具栏的编译中可以选择打包工程，打包固件和打包运行系统。

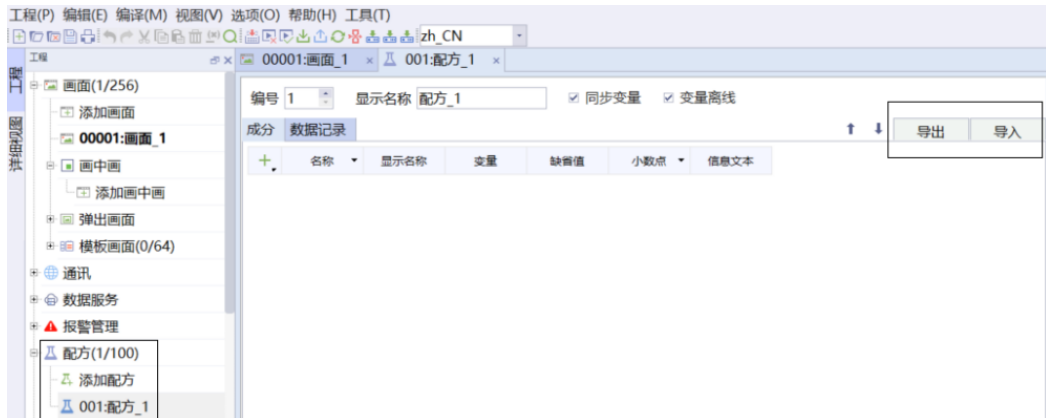


2. 打包工程即打包HMI工程，打包固件即打包HMI固件版本，打包运行系统即打包HMI软件，根据自己的需求进行选择，将打包好的文件放入U盘或SD卡中，插入HMI即可使用。

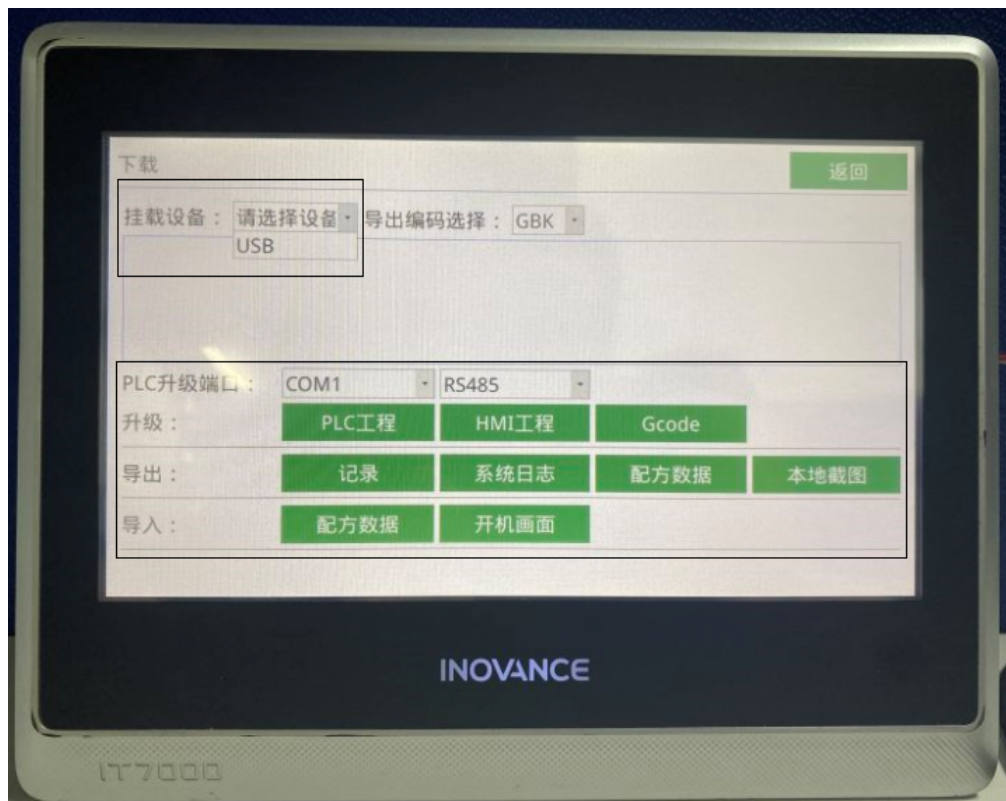




3. 在软件界面左侧“工程”>“配方”>“添加配方”中添加一组配方，在右上角可以选择导入导出配方数据，也可在U盘或者SD卡插入后在后台导入导出。



4. 将U盘或者SD卡插入HMI后，会弹出进入后台的窗口，默认密码是111111，进入后可以看见如下页面，可根据自己的需求导入导出相关数据。



# 10 历史记录

## 10.1 如何查看HMI各类控件的操作记录

### 案例要求

1. 学会查看HMI各类控件的操作记录。
2. 软件设置记录HMI的操作记录。
3. 导出操作记录。

### 案例目的

通过此案例学会如何查看并导出HMI的操作记录。

### 编程步骤

#### 1. InoTouchPad设置记录操作记录

添加设置：在软件界面左侧“工程”>“HMI设置”>“工程设置”>“操作记录设置”中勾选记录操作记录，如下图所示，同时可以设置记录条数，最大可设置为10000条。



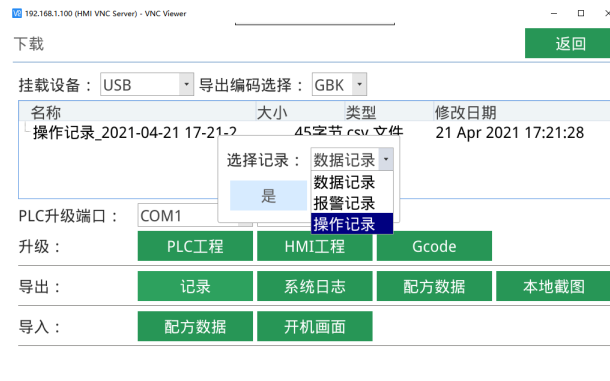
#### 2. 导出操作记录

设置完成之后，运行系统一段时间后，可在HMI设备的控制面板中，对记录的数据记录数据进行导出。具体操作步骤如下：

- a. 首先，在触摸屏上插入外部存储介质（U盘或SD卡），并进入到控制面板的下载界面，选择文件导出的路径，这里导出到path1，如下图。



- b. 然后在导出行中点击“记录”按钮，此时，会弹出如下对话框。



c. 选择数据记录，点击是，会再次弹出确认导出窗口。



d. 确认后，即执行数据记录导出；待进度条走完，若导出成功，则弹出导出成功提示。导出文件为\*.csv 格式，导出文件存储位置为用户指定的路径，用户未选择路径时，默认导出到根目录，导出文件可用Excel 或记事本打开查看。

# 11 报表

## 11.1 利用报表实现数据统计

### 案例要求

使用报表功能实现数据统计。

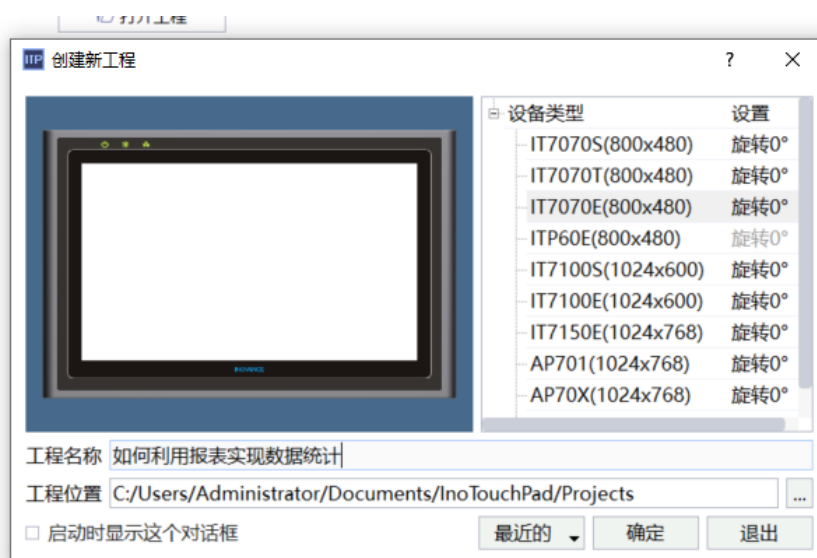
### 案例目的

学会报表控件的使用。

### 编程步骤

#### 1. 新建工程

用汇川IT7000编程软件新建一个工程。



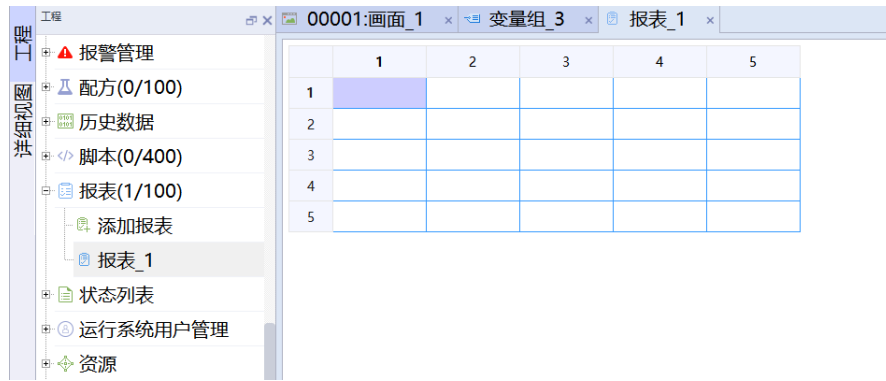
#### 2. 添加变量

在工具“通讯”>“变量”中添加变量，如下图所示。

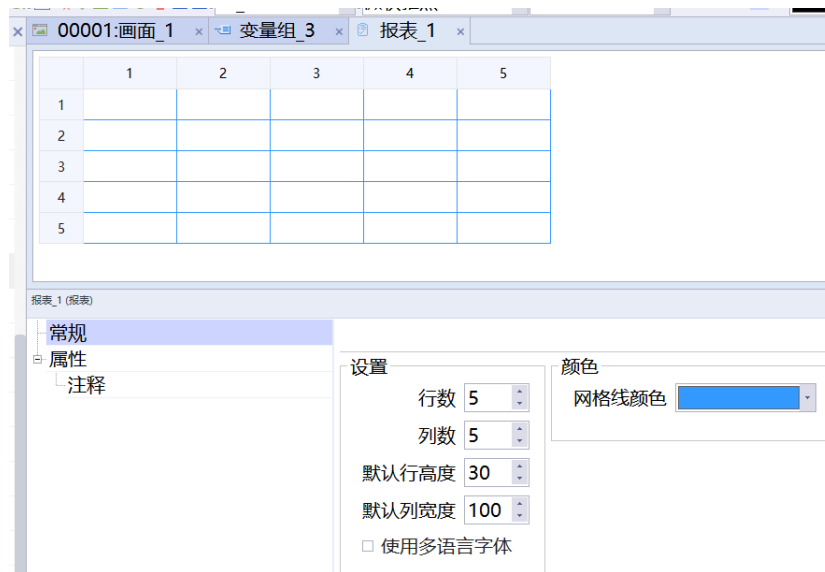


#### 3. 添加报表

在软件界面左侧“工程”>“报表”>“添加报表”双击添加一个报表\_1，右侧即自动生成报表样式，如下图所示



单击报表右侧空白界面，可在“常规”>“设置”选项中，根据需求调节行数、列数等参数，如下图所示。

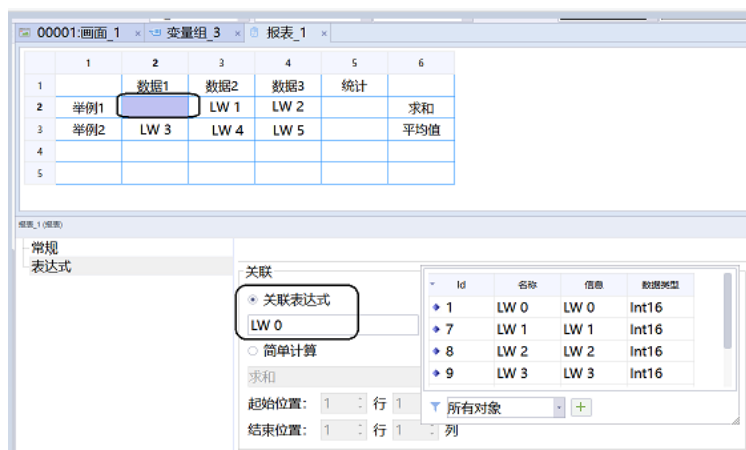


#### 4. 添加数据

a. 单击报表对应表格即可在报表中添加中文标注，如下图所示。

	1	2	3	4	5	6
1		数据1	数据2	数据3	统计	
2	举例1					求和
3	举例2					平均值
4						
5						

b. 将之前添加的变量与单元格进行匹配，选中单元格“表达式”>“关联表达式”选中对应变量，如下图所示。



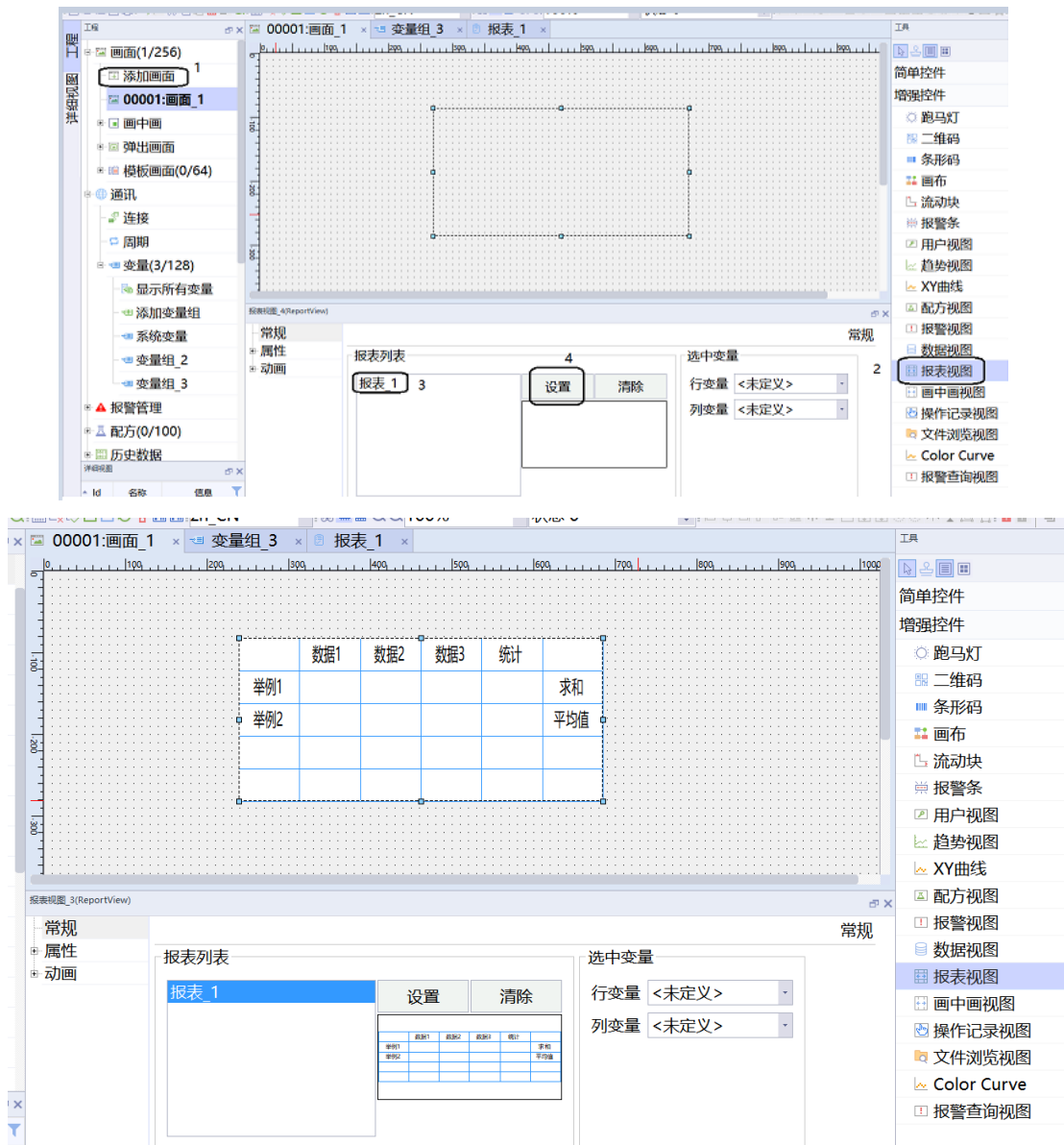
c. 选中显示统计数据的单元格，在“简单计算”中选中所需数据统计类型，并且设定起始位置和结束位置的行和列，如下图所示



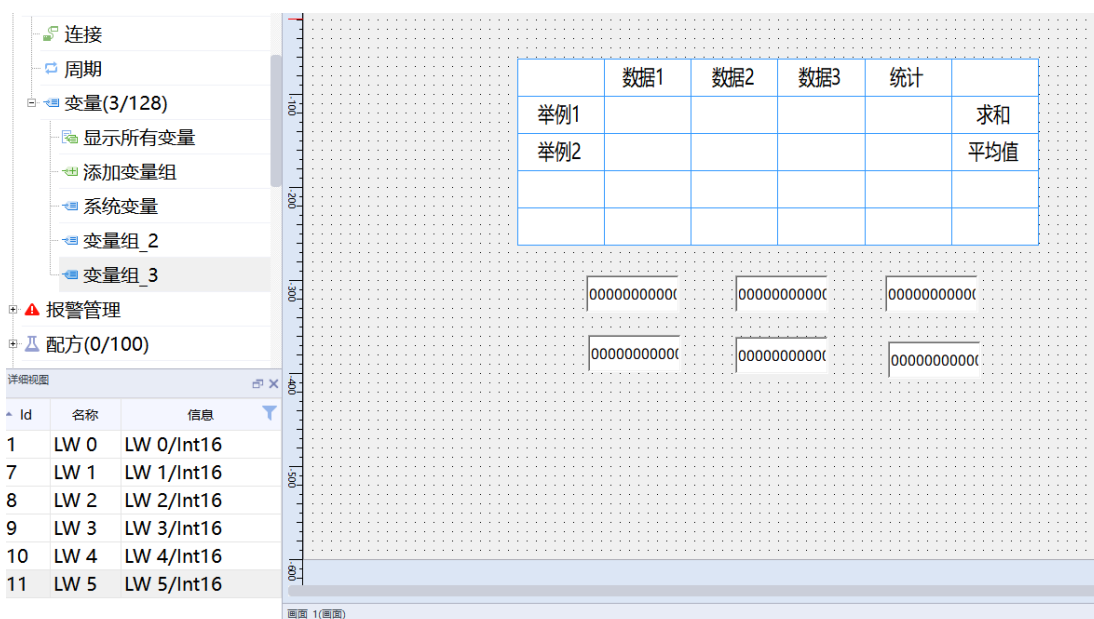
5. 组态报表



- a. 在界面左侧“工程”>“画面”>“添加画面”中添加一个画面\_1，在右侧工具栏“增强控件”中选中报表视图将其拖出至画面中，在拖出的报表视图中，选择刚才添加的“报表\_1”并点击设置(也可以点击左侧新建的报表\_1进行拖动至画面中)，如下图所示。



- b. 在主画面中点击左侧“工程”>“通讯”>“变量”>“变量组\_3”中将之前添加的变量分别拖入至主画面中，如图所示，也可在报表中设置可编辑直接进行数值输入。



6. 离线模拟效果展示

IT7100E(1024x600)

	数据1	数据2	数据3	统计	
举例1	11	22	33	66	求和
举例2	10	20	30	20	平均值

11      22      33

10      20      30

# 12 脚本

## 12.1 利用脚本生成二维码和条形码

### 案例要求

- 通过HMI 软件添加画中画画面实现TAB窗口功能。
- 添加二维码控件，实现字符串二维码显示。
- 添加条形码控件，实现字符串条形码显示。
- 组合不同数据类型变量生成二维码、条形码。

### 案例目的

- 通过此案例掌握二维码、条形码控件使用。
- 能够完成合并字符串、整数、浮点数等变量生成二维码/条形码。

### 编程步骤

#### 1. 添加变量

添加类型变量：软件界面左侧“工程”“通讯”>“变量”>“添加变量组”添加一个变量组，新建所需变量如下图所示。



+	名称	编号	连接	数据类型	长度	数组计数	地址	采集周期
1	字符串输入	7	<内部变量>	String	40	1	LW 0	1s
2	公司名称	8	<内部变量>	String	10	1	LW 100	1s
3	年份	9	<内部变量>	UInt32	4	1	LW 110	1s
4	销售额	10	<内部变量>	Float	4	1	LW 120	1s
5	年份1	11	<内部变量>	String	10	1	LW 200	1s
6	销售额1	12	<内部变量>	String	10	1	LW 210	1s

#### 2. 添加二维码

- 点击画面左侧“工程”>“画面”>“添加画面”添加一个画面，在界面右侧“工具”>“增强控件”>“二维码”添加二维码控件，将其拖入画面中。单击二维码，在“常规”>“过程”>“变量”里添加已添加好的“字符串输入”变量。

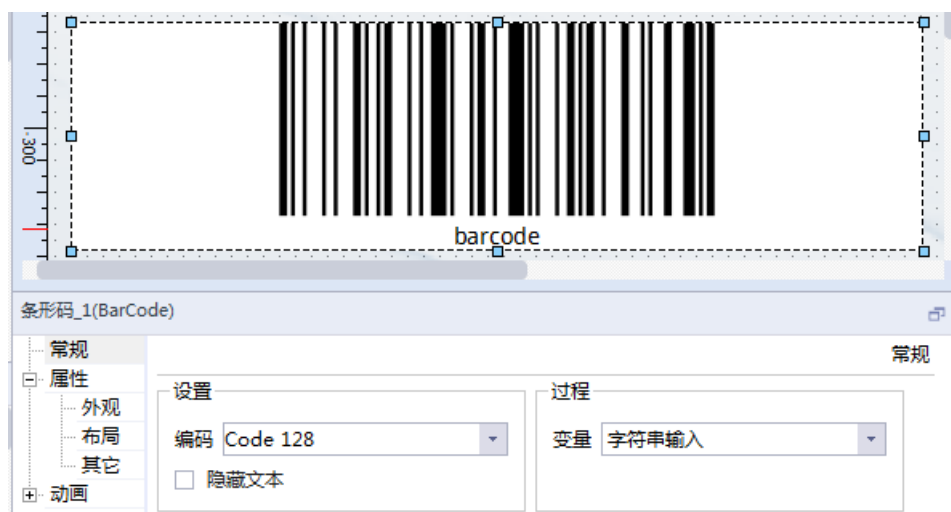


- b. 二维码控件尺寸调整：单击二维码控件，在“属性”>“布局”>“尺寸”中调整二维码控件尺寸，将其宽度和高度均调整为250。



### 3. 添加条形码

- a. 添加条形码控件：界面右侧“工具”>“增强控件”>“条形码”添加条形码控件，单击条形码，在“常规”>“过程”>“变量”添加已添加好的“字符串输入”变量。



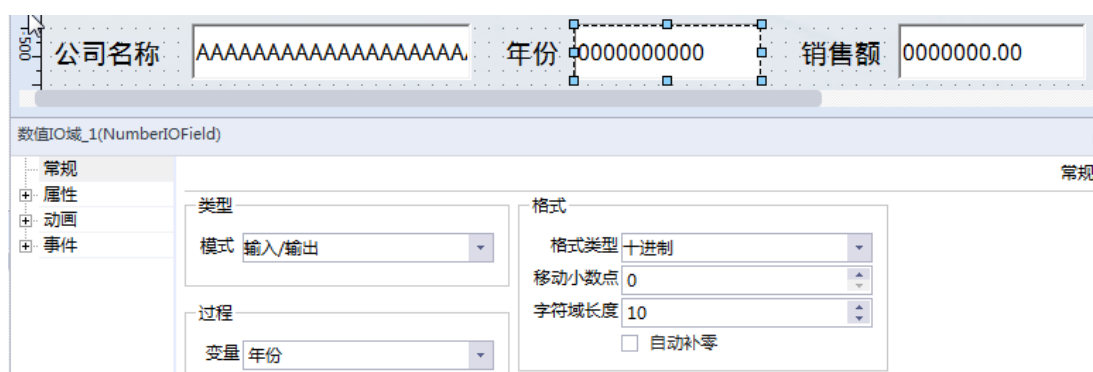
- b. 条形码控件尺寸调整：单击条形码，在“属性”>“布局”>“尺寸”调整二维码控件尺寸，将其宽度和高度均调整为250。



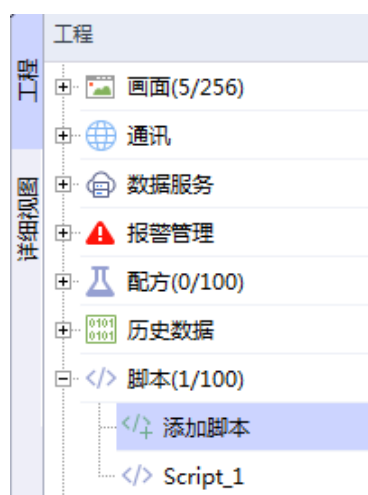
#### 4. 组合不同数据类型变量生成二维码、条形码

- a. 组态画面：在主画面中点击左侧变量组将添加的“公司名称”、“年份”、“销售额”变量分别拖入至画面中进行组态，在界面右侧“工具”>“简单控件”>“文本域”，将其添加至画面中，分别更改文本为“公司名称”、“年份”、“销售额”。

2	公司名称	8	<内部变量>	String	10	1	LW 100	1s
3	年份	9	<内部变量>	UInt32	4	1	LW 110	1s
4	销售额	10	<内部变量>	Float	4	1	LW 120	1s



- b. 添加脚本：界面左侧“工程”>“脚本”>“添加脚本”添加一个脚本“Script\_1”。



- c. 添加脚本内容，通过脚本转换不同类型变量为字符串后合并生成二维码、条形码。

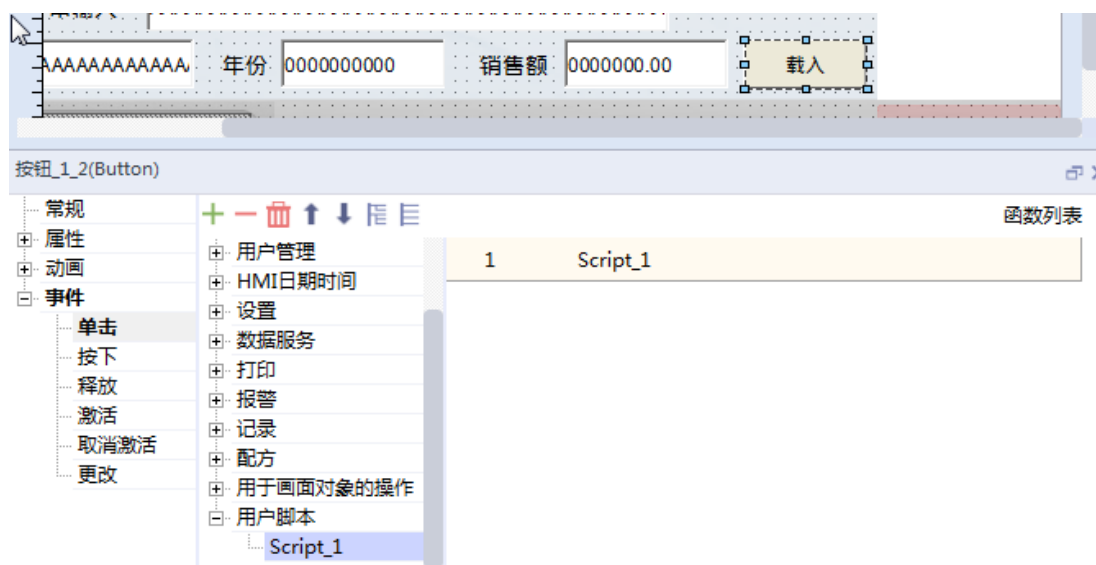
脚本内容如下：

```
//字符串转换合并
```

```
//字符串载入a
var a=SmartTags("公司名称");
//整数转为字符串载入b1
var b=SmartTags(年份);
var b1=b.toFixed(0);
//浮点数转换为字符串载入c1
var c=SmartTags("销售额");
var c1=c.toFixed(2);
//两个字符串合并连接
var e="-";
var a1=a.concat(e);
var a2=a1.concat(b1);
var a3=a2.concat(e);
SmartTags("字符串输入")=a3.concat(c1); //二维码/条形码连接字符串
```

```
function Script_10 {
1 //字符串转换合并
2 //字符串载入a
3 var a=SmartTags("公司名称");
4 //整数转为字符串载入b1
5 var b=SmartTags('年份');
6 var b1=b.toFixed(0);
7 //浮点数转换为字符串载入c1
8 var c=SmartTags("销售额");
9 var c1=c.toFixed(2);
10 //两个字符串合并连接
11 var e="-";
12 var a1=a.concat(e);
13 var a2=a1.concat(b1);
14 var a3=a2.concat(e);
15 SmartTags("字符串输入")=a3.concat(c1); //二维码/条形码连接字符串
16
```

- d. 按钮调用脚本：添加一个按钮控件，通过按钮属性视图“事件” > “单击” > “用户脚本”调用“Script\_1”脚本。



#### 5. 利用脚本生成二维码、条形码完成

触发按钮控件“载入”，脚本实现合并公司名称、年份、销售额等变量，输出字符串于二维码、条形码显示。HMI仿真效果图如下。



## 12.2 利用脚本实现指定时间弹窗

### 案例要求

1. 通过HMI 软件自带的脚本工具，实现弹窗功能。
2. 编写脚本读取系统时间，实现指定时间弹窗。

### 案例目的

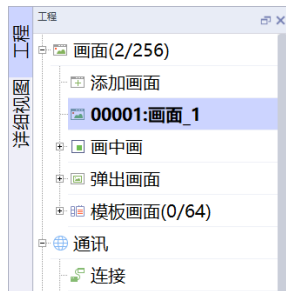
1. 通过此案例了解脚本工具使用。
2. 通过此案例学会利用脚本实现相应功能。



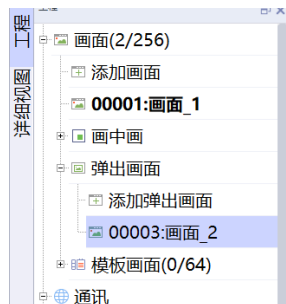
## 编程步骤

## 1. 添加画面

a. 添加画面：在软件界面左侧“工程”>“画面”>“添加画面”添加画面\_1。

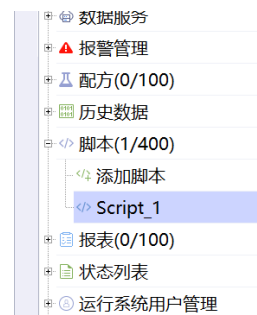


b. 添加弹窗画面：在软件界面左侧“工程”>“画面”>“弹出画面”添加弹出画面\_2。



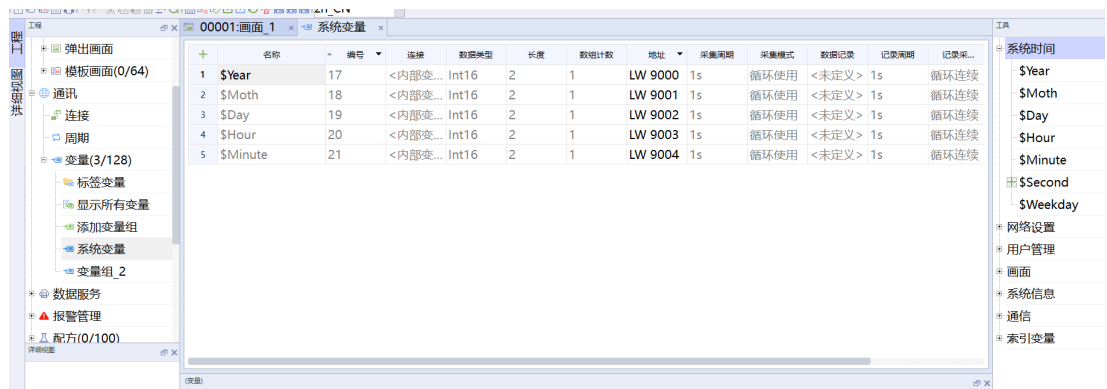
## 2. 添加脚本

在软件界面左侧“工程”>“脚本”>“添加脚本”添加一个脚本用于实现定时弹窗。

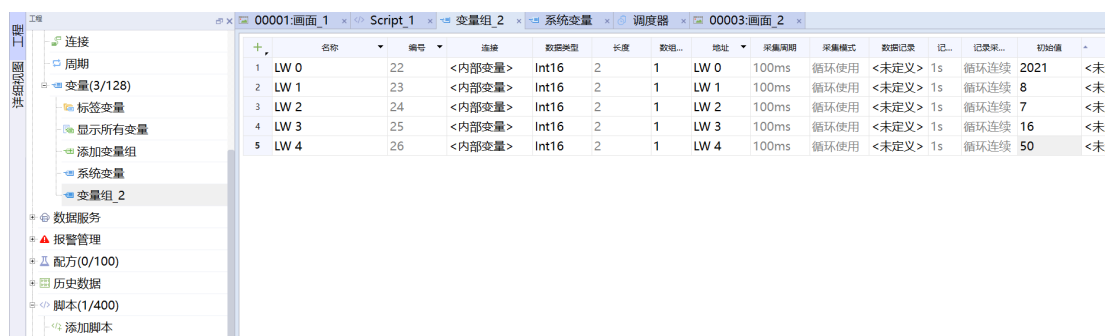


## 3. 添加变量

a. 添加系统变量：在软件界面左侧“工程”>“通讯”>“系统变量”里点击界面右侧“工具”>“系统时间”添加年、月、日、时、分系统变量。用于获取系统时间。



- b. 添加内部变量：在软件界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加一组内部变量用于和系统时间进行对比，如图所示。并分别给变量进行设定时间赋值(年、月、日、时、分)。



#### 4. 编写脚本

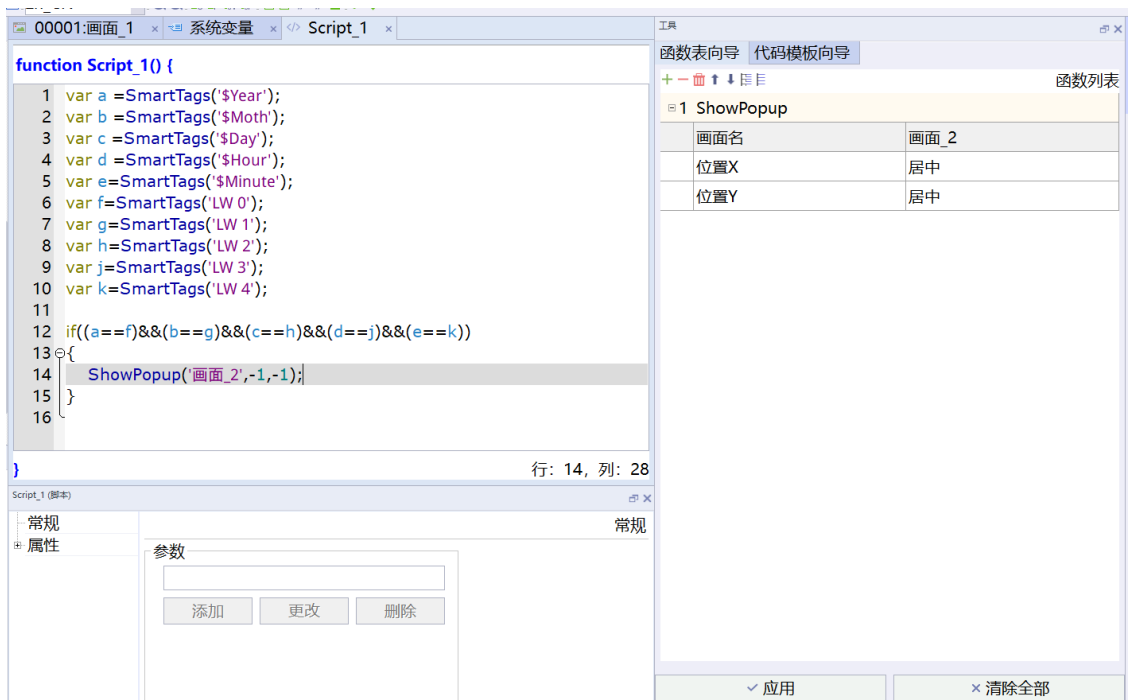
此案例的定时弹窗功能为设定时间到后自动弹出弹窗画面。变量值获取SmartTags可通过鼠标右键“列出所有系统变量”快捷生成。画面弹窗ShowPopup函数可在界面右侧“工具”>“函数表向导”>“画面”进行快捷获取应用。



脚本代码如下：

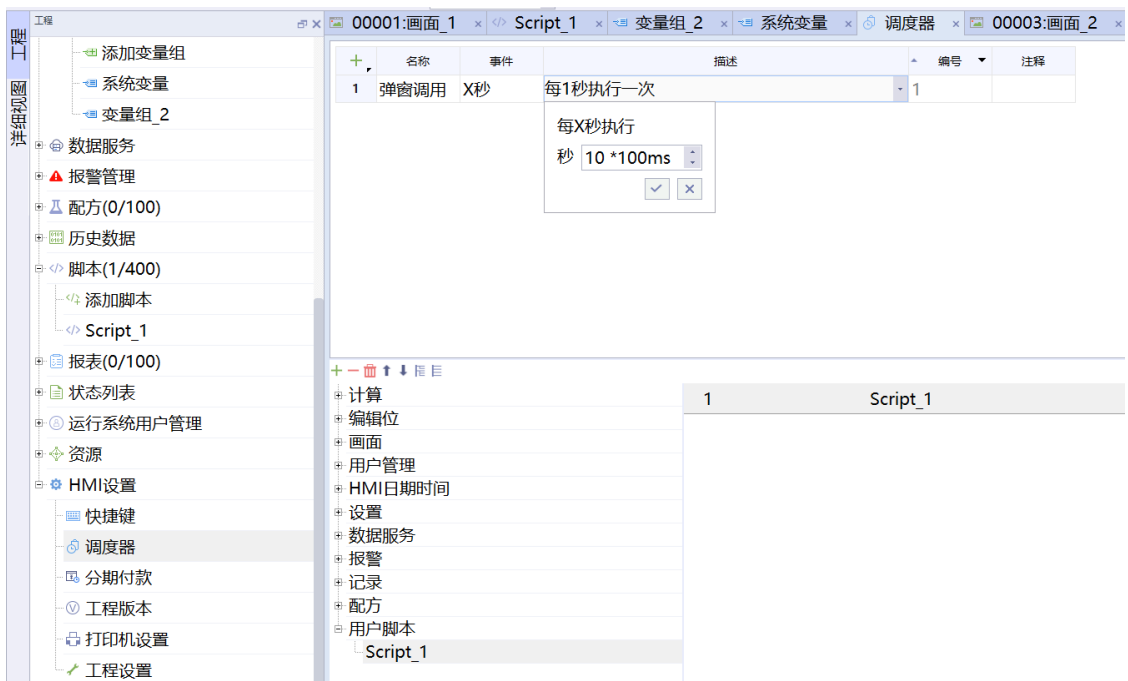
```
var a =SmartTags('$Year');
var b =SmartTags('$Month');
var c =SmartTags('$Day');
var d =SmartTags('$Hour');
var e =SmartTags('$Minute');
var f =SmartTags('LW 0');
var g =SmartTags('LW 1');
var h =SmartTags('LW 2');
var j =SmartTags('LW 3');
var k =SmartTags('LW 4');
if((a==f)&&(b==g)&&(c==h)&&(d==j)&&(e==k))
```

```
{
    ShowPopup('画面_2,-1,-1);
}
```



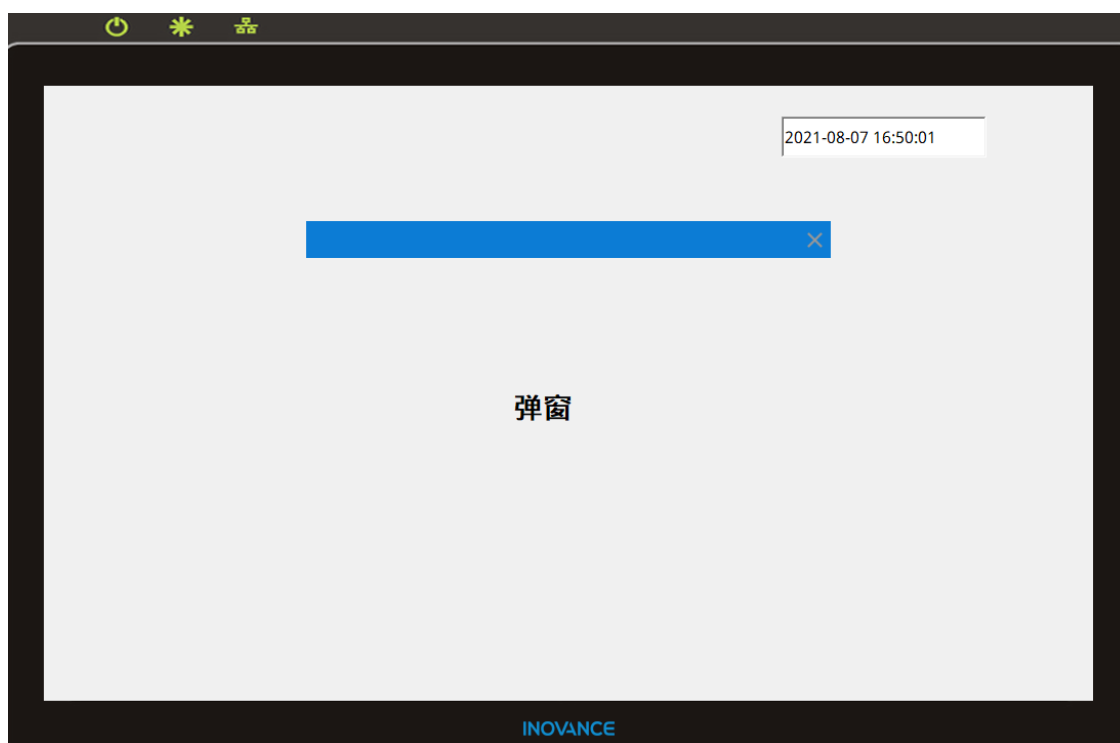
### 5. 添加调度器

在软件界面左侧“工程”>“HMI设置”>“调度器”点击左上角“+”号进行添加调度器事件，事件为X秒调用，描述改为1秒执行一次。之后选中事件，在用户脚本里双击用户脚本进行关联。



### 6. 仿真模拟效果

开始仿真，当设定时间和系统时间一致时触发脚本进行弹窗，同理也可进行定时画面切换等功能，可根据需要自由发挥，仿真模拟效果如下。



## 12.3 利用调度器实现画面切换以及运行脚本

### 案例要求

- 通过HMI 软件自带的调度器工具，实现一些功能。
- 实现画面切换功能。
- 实现脚本运行功能。

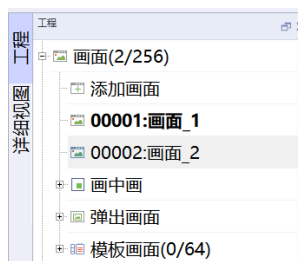
### 案例目的

通过此案例了解调度器的作用，学会调度器的使用方法。

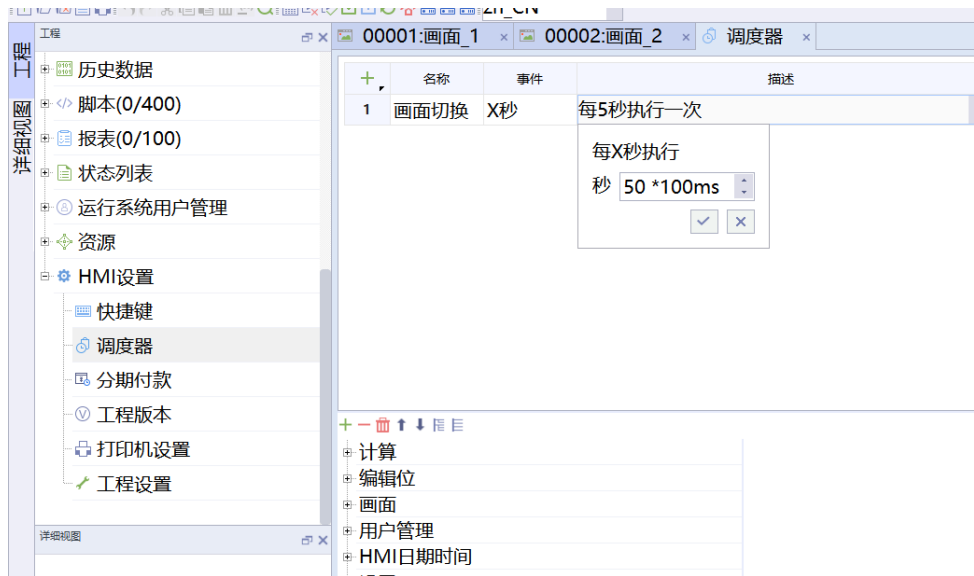
### 编程步骤

#### 1. 调度器实现画面切换

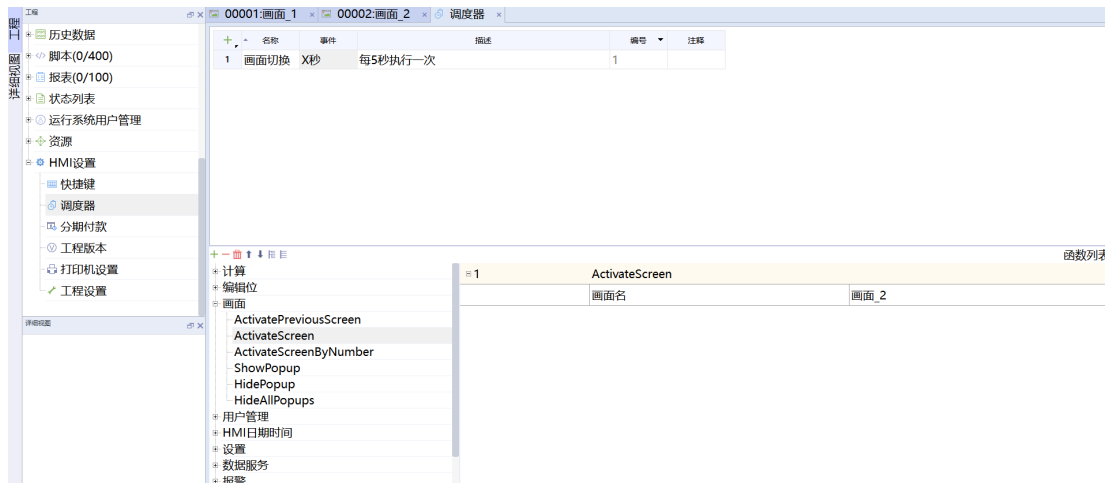
- 添加画面：在软件界面左侧“工程”>“画面”>“添加画面”分别添加画面\_1和画面\_2两个画面。



- 添加事件：在界面左侧“工程”>“HMI设置”>“调度器”双击调度器生成事件编辑界面，点击左上角“+”号添加一个调度器事件，命名为画面切换，用于切换画面。事件类型选择“X秒”，即循环触发，设置为每5秒执行一次。



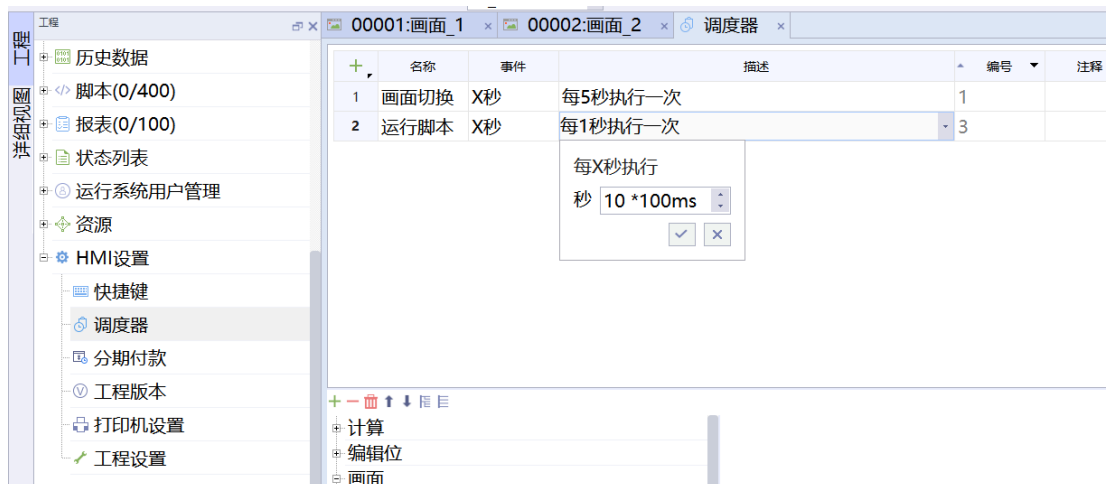
c. 单击“画面切换”调度器事件，在“画面”“ActivateScreen”里添加要切到的画面，将这里设置切换到画面\_2。



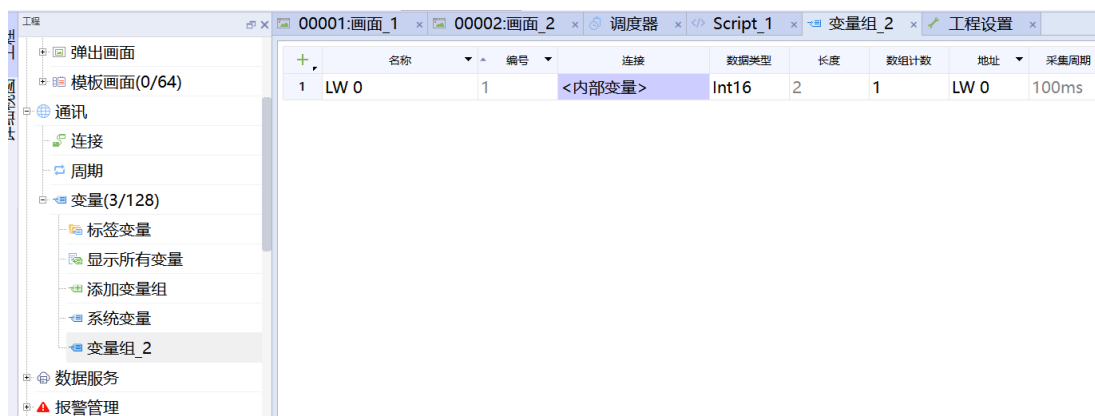
d. 仿真模拟效果为一开始从画面\_1开始显示，5秒后自动跳转至画面\_2。

## 2. 调度器运行脚本

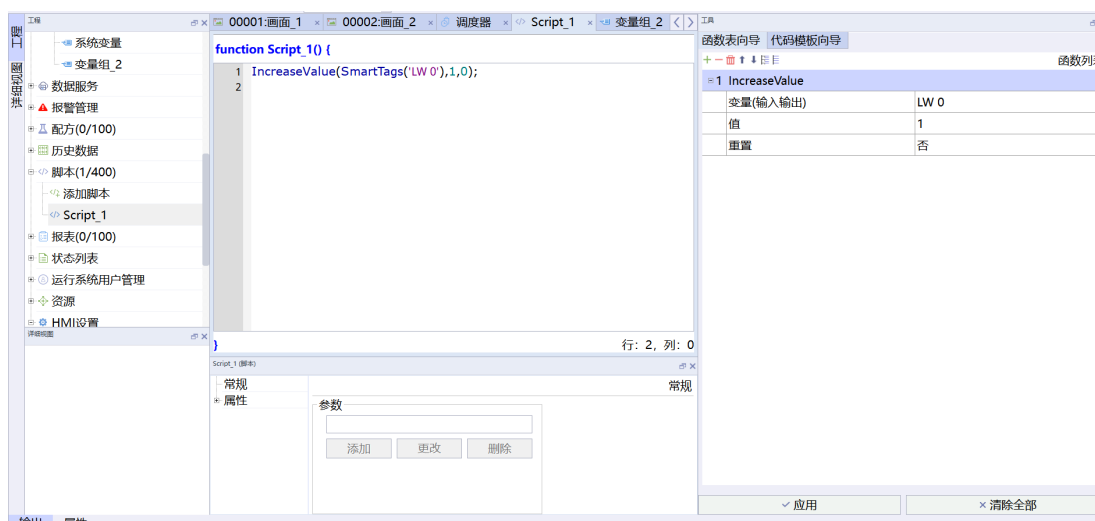
a. 按照第一步的方法新建第二个调度器事件，命名为运行脚本，事件类型选择“X秒”，即循环触发，设置为每1秒执行一次。



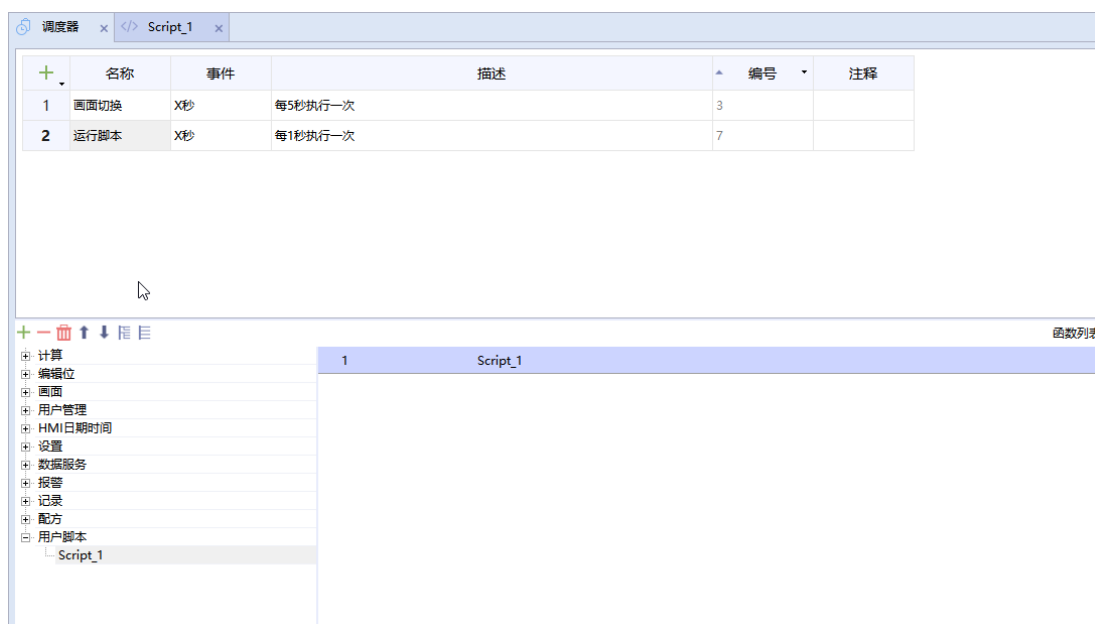
b. 界面左侧“工程”>“通讯”>“变量”>“添加变量组”添加一个变量LW 0。



c. 界面左侧“工程”>“脚本”>“添加脚本”添加一个脚本Script\_1, 点击右侧“函数表向导”, 选择“计算”>“IncreaseValue”, 在“变量(输入输出)”里关联变量LW 0, 点击右下角应用即可将函数脚本添加至脚本中, 这个脚本的主要用途是实现寄存器LW 0中的值每秒自增1。



d. 选中调度器事件“运行脚本”, 在用户脚本里双击组态运行脚本。



e. 在画面里将变量值拖入画面, 在线仿真运行, 观察到D0每秒自增1。



## 12.4 脚本获取系统时分秒

### 案例要求

利用后台软件脚本功能，获取系统时间。

### 案例目的

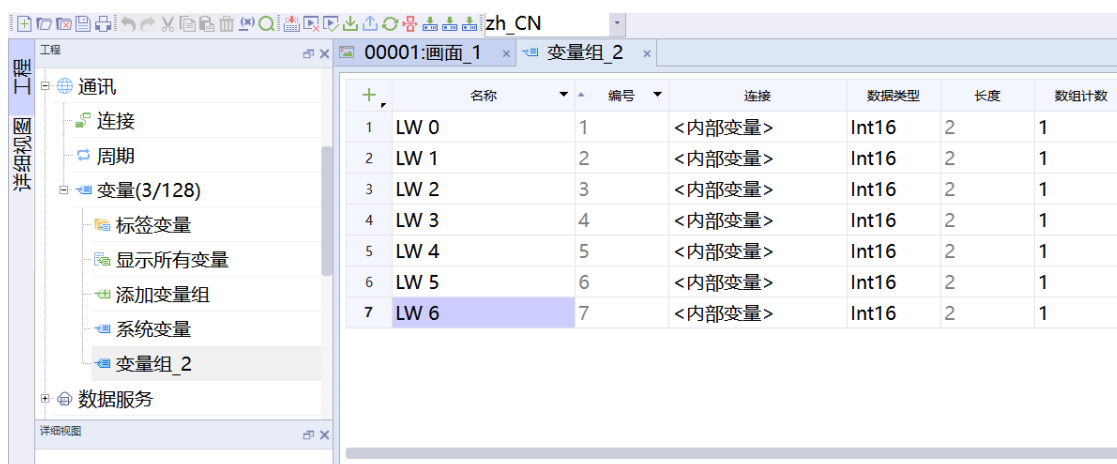
利用后台软件脚本功能，获取系统时间。

### 编程步骤

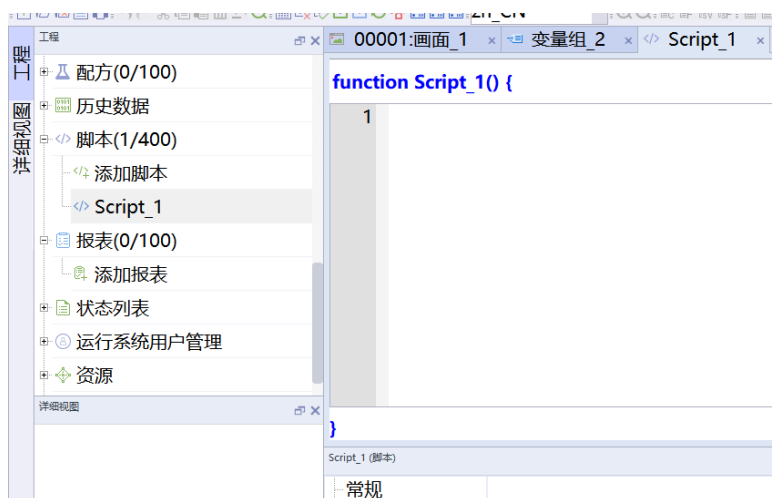
1. 用汇川IT7000编程软件新建一个工程。



2. 在软件界面左侧“工程”>“通讯”>“变量”>“添加变量组”，点击左上角“+”号添加8个内部变量，如下图所示。

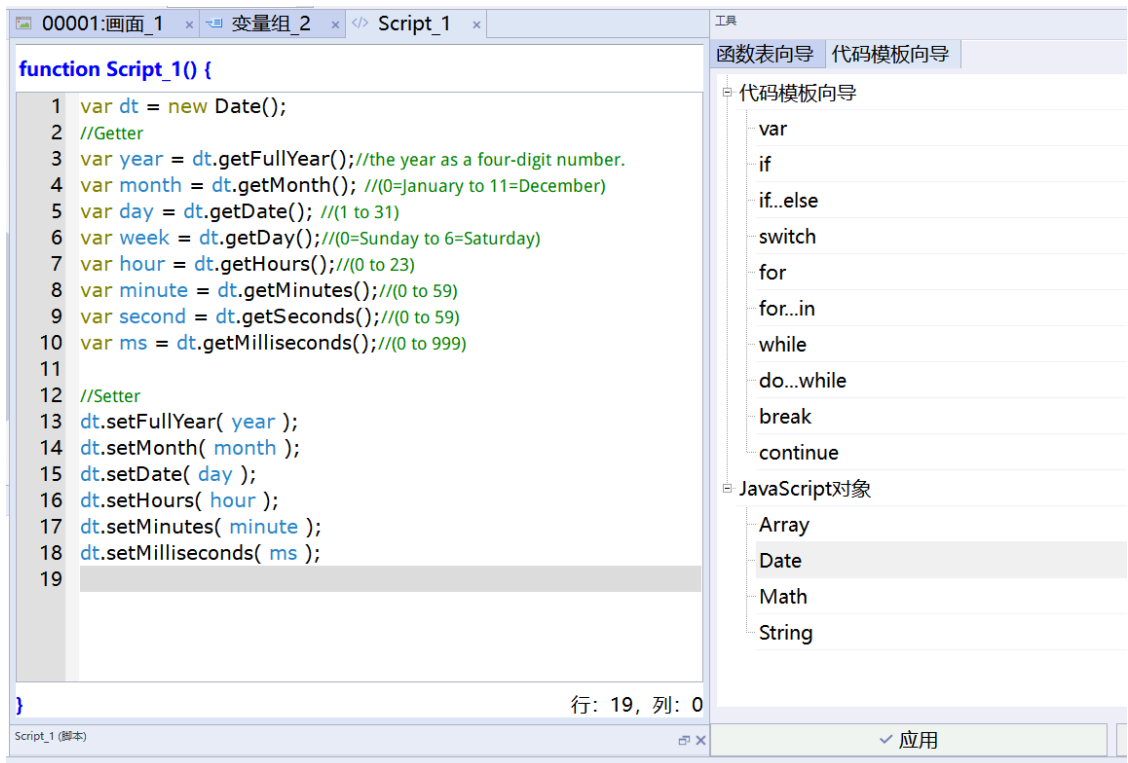


3. 在软件界面左侧“工程”>“脚本”>“添加脚本”添加一个脚本如下图所示。

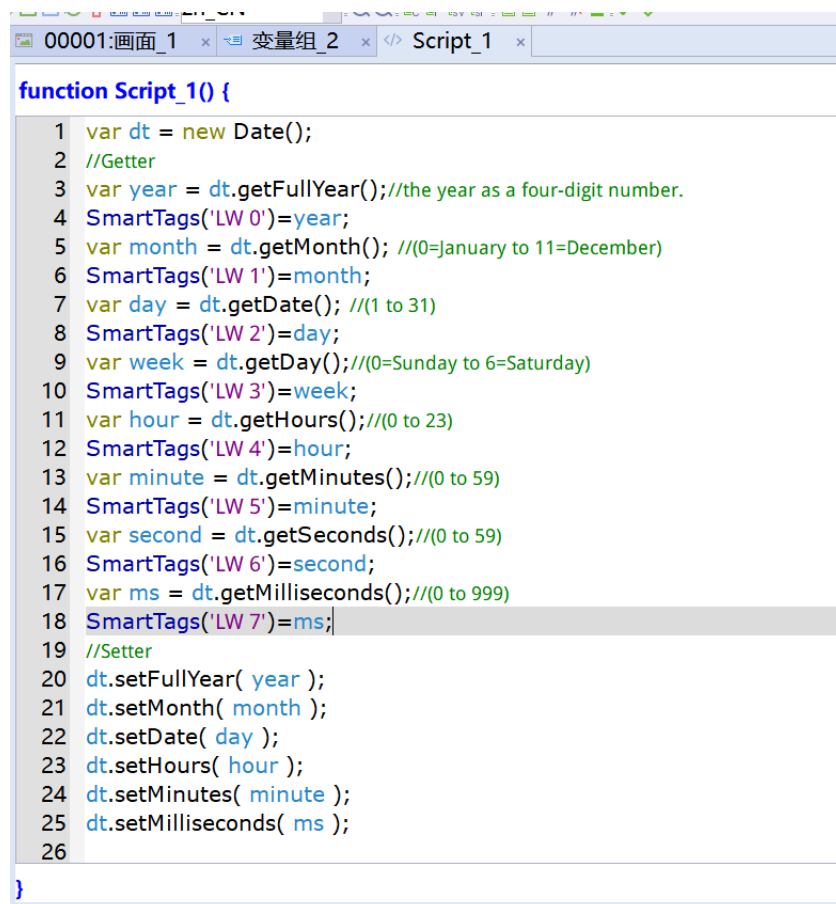


4. 在软件界面右侧“工具”>“代码模板向导”>“JavaScript”对象中，双击“Date”自动生成脚本，如下图所示。

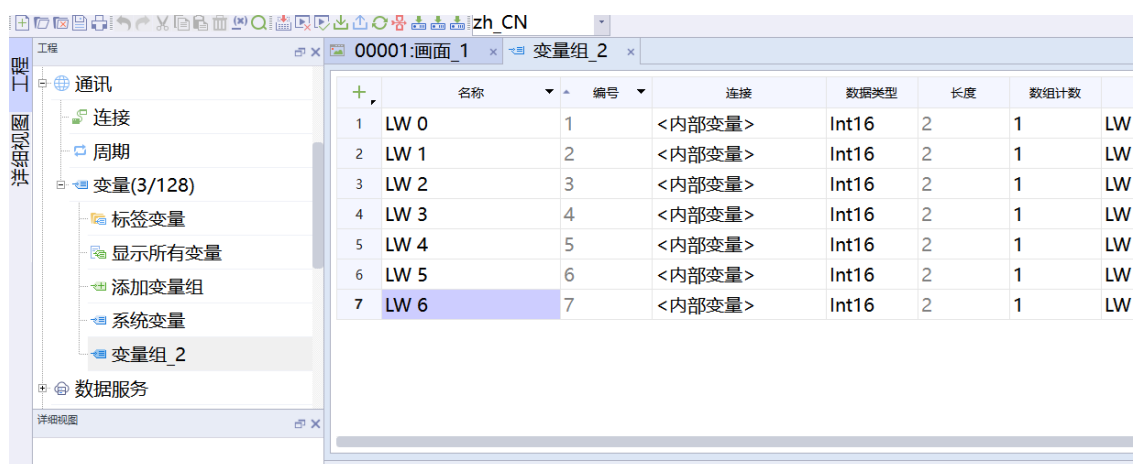




5. 把之前添加的变量与脚本中对应参数进行分别赋值，如图所示。



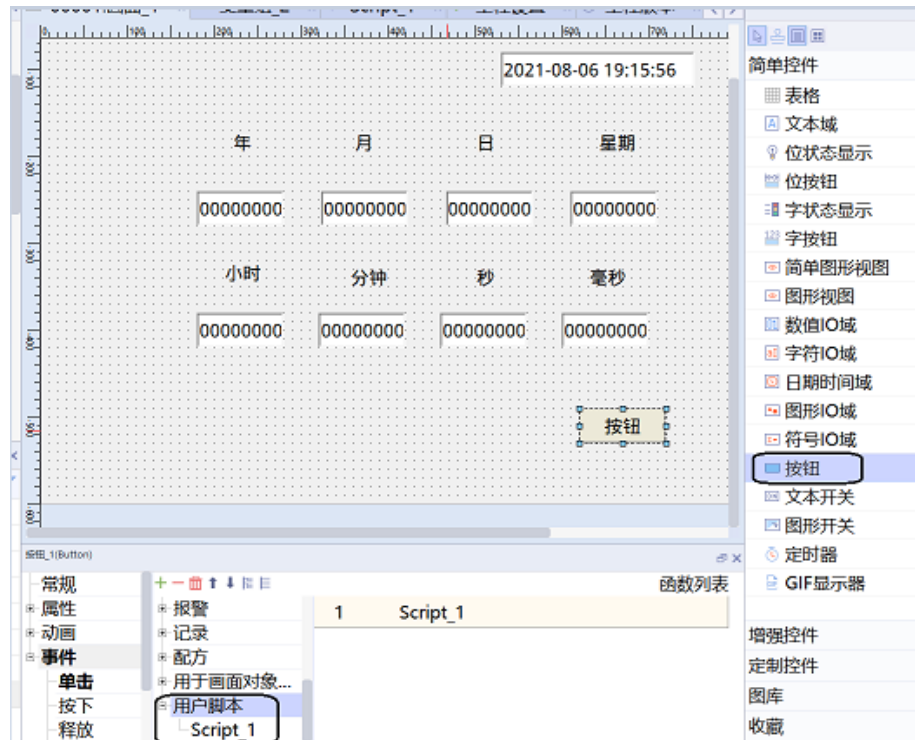
6. 将之前添加的变量拖拽到主画面中。



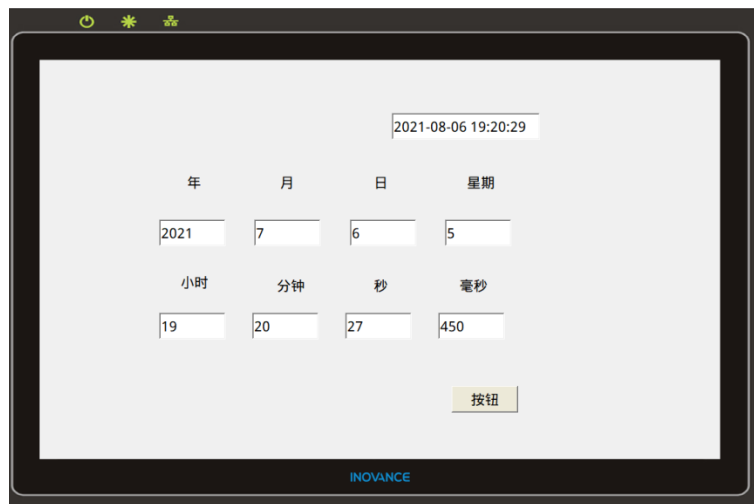
7. 在软件界面右侧“工具”>“简单控件”选中“时间IO域”，将其拖入主画面中，用于与脚本系统函数获取的时间进行对比参照。



8. 在软件界面右侧“工具”>“简单控件”中，选中“按钮”控件并将其拖拽到主画面中，单击按钮，在“事件”>“单击”“用户脚本”选中之前编辑好的脚本Script\_1。



9. 离线模拟效果如下图所示，其中需要注意脚本利用JavaScript脚本获取的月份比用时间IO域控件获取的月份少1，这是因为JavaScript的月份字段，返回值是0（一月）到11（十二月）。



此时只需在脚中将月份进行+1处理，即可获取正确时间。

```
function Script_10 {  
1 var dt = new Date();  
2 //Getter  
3 var year = dt.getFullYear();//the year as a four-digit number.  
4 SmartTags('LW 0')=year;  
5 var month = dt.getMonth(); //(0=January to 11=December)  
6 SmartTags('LW 1')=month+1;  
7 var day = dt.getDate(); //(1 to 31)  
8 SmartTags('LW 2')=day;  
9 var week = dt.getDay();//(0=Sunday to 6=Saturday)  
10 SmartTags('LW 3')=week;  
11 var hour = dt.getHours();//(0 to 23)  
12 SmartTags('LW 4')=hour;  
13 var minute = dt.getMinutes();//(0 to 59)  
14 SmartTags('LW 5')=minute;  
15 var second = dt.getSeconds();//(0 to 59)  
16 SmartTags('LW 6')=second;  
17 var ms = dt.getMilliseconds();//(0 to 999)  
18 SmartTags('LW 7')=ms;  
..
```

2021-08-08 11:58:19

年	月	日	星期
2021	8	8	0
小时	分钟	秒	毫秒
11	58	18	984

按钮

## 12.5 在进入画面或者离开画面时调用一次脚本

### 案例要求

通过HMI软件完成在进入画面时对脚本进行一次调用：

1. 添加编写用户脚本。
2. 对画面切换按钮进行关联用户脚本。

### 案例目的

通过此案例学会对脚本的调用。

### 编程步骤

1. 新建工程画面
  - a. 新建一个IT7000工程。

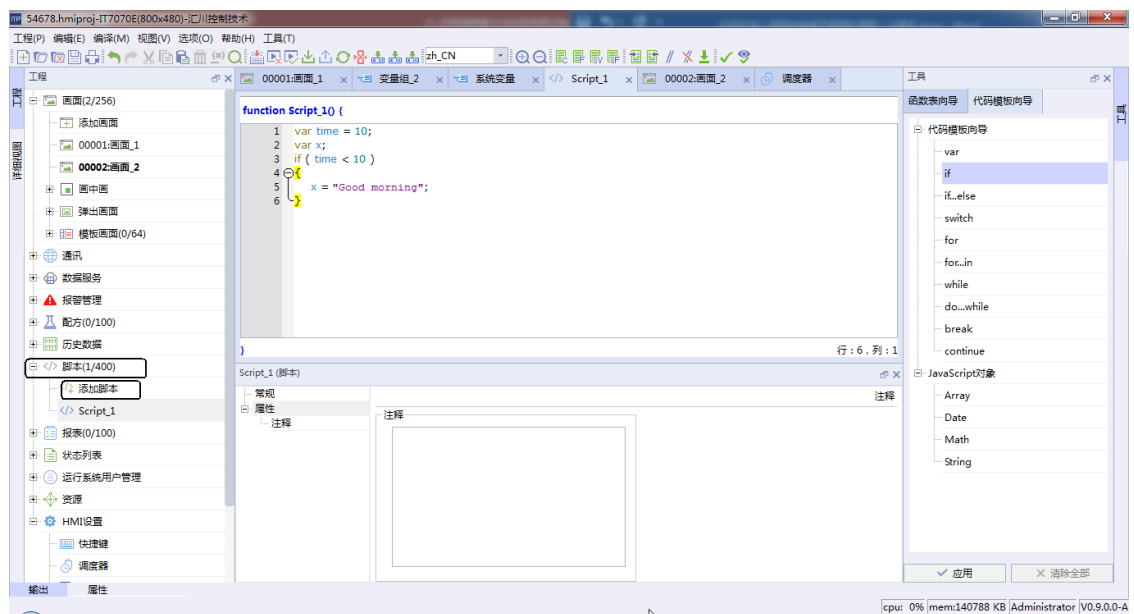


b. 在软件界面左侧“工程”>“画面”>“添加画面”分别添加两个画面\_1和画面\_2。



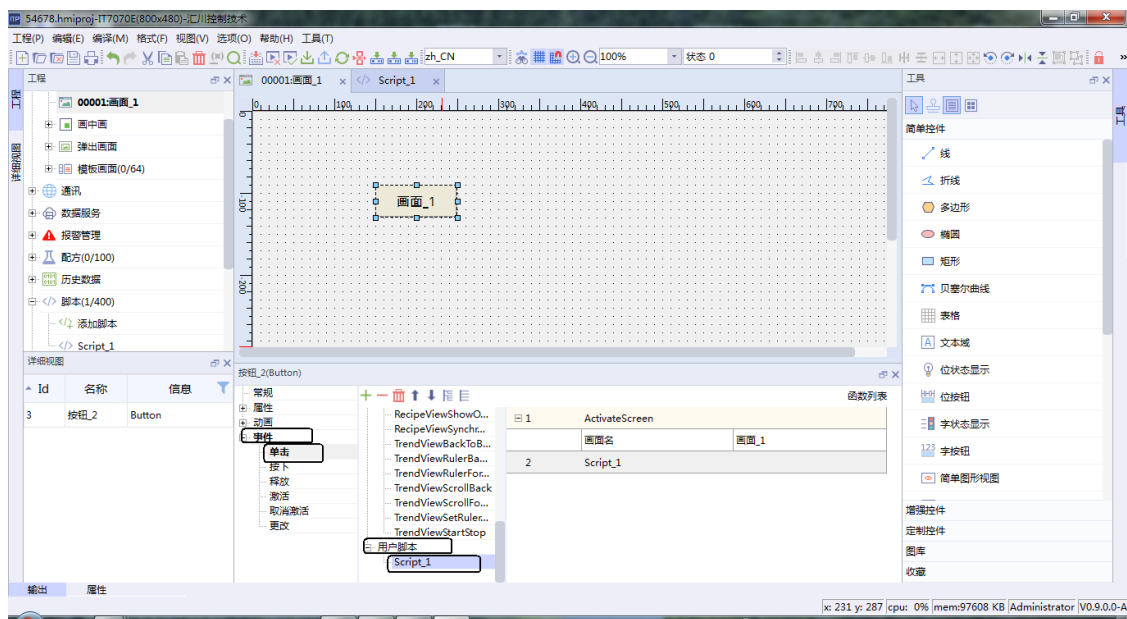
## 2. 添加并编写用户脚本

在软件界面左侧“工程”>“脚本”>“添加脚本”，添加一个用户脚本Script\_1，在界面右侧“工具”>“代码模板向导”双击if添加用户脚本内容。



## 3. 画面切换按钮关联用户脚本

分别在画面\_1和画面\_2中添加切换按钮(可使用ActivateScreen函数或者手动拖拽实现)，单击画面切换按钮，在“事件”>“单击”>“用户脚本”关联用户脚本Script\_1。



#### 4. 运行效果

在模拟运行时在画面\_1中点击切换按钮可跳转至画面\_2，此时会调用一次脚本。跳转至画面\_2后，点击切换至画面1的按钮时会调用一次脚本。即实现进入和离开时分别调用一次脚本。

# 13 函数

## 13.1 脚本函数介绍（浏览、概念）

### 案例要求

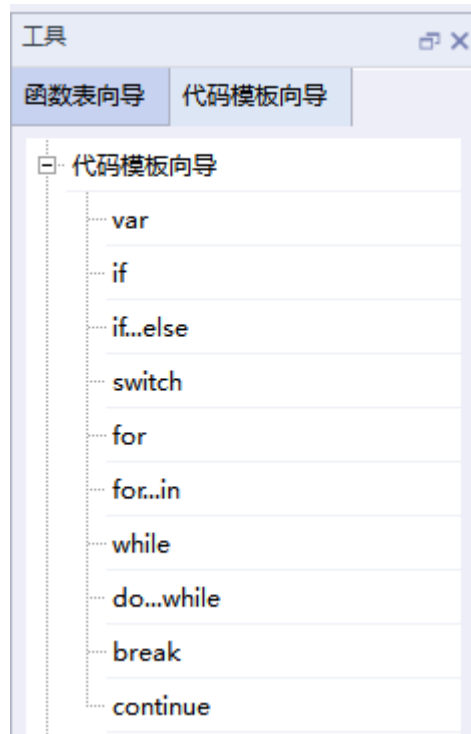
1. 了解HMI软件支持的JS函数。
2. JS常用函数功能及使用方法。
3. 使用脚本工具实现变量控制弹窗功能。

### 案例目的

通过此案例了解JS相关函数，学会常用函数的使用方法。

### 编程步骤

- JS介绍  
InoTouchPad 脚本采用的是JavaScript 代码编写，具有更高的灵活性，可以完成一些运算逻辑等。  
JavaScript 是互联网上最流行的脚本语言，这门语言可用于 HTML 和 web，更可广泛用于服务器、PC、笔记本电脑、平板电脑和智能手机等设备。
- JS相关函数介绍
  - 函数介绍：函数就是包裹在花括号中的代码块，前面使用了关键词 function，当调用函数时，会执行函数内的代码。可以在某事件发生时直接调用函数（比如当用户点击按钮时），并且可由 JavaScript 在任何位置进行调用。JavaScript 对大小写敏感。关键词 function 必须是小写的，并且必须以与函数名称相同的大小写来调用函数。



- JavaScript 条件语句：通常在写代码时，用户总是需要为不同的决定来执行不同的动作。用户可以在代码中使用条件语句来完成该任务。在 JavaScript 中，我们可使用以下条件语句。  
if 语句：只有当指定条件为 true 时，该语句才会执行代码。

```
if (condition){  
  // 当条件为 true 时执行的代码  
}
```

if...else 语句：使用 if...else 语句在条件为 true 时执行代码，在条件为 false 时执行其他代码。

```
if (condition){  
  // 当条件为 true 时执行的代码  
}  
else{  
  // 当条件不为 true 时执行的代码  
}
```

if...else if...else 语句：使用 if...else if...else 语句来选择多个代码块之一来执行。

```
if (condition1){  
  // 当条件 1 为 true 时执行的代码  
}  
else if (condition2){  
  // 当条件 2 为 true 时执行的代码  
}  
else{  
  // 当条件 1 和 条件 2 都不为 true 时执行的代码  
}
```

switch 语句：使用 switch 语句来选择要执行的多个代码块之一。

```
switch(n) {  
  case 1: // 执行代码块 1  
    break;  
  case 2: // 执行代码块 2  
    break;  
  default:  
    // 与 case 1 和 case 2 不同时执行的代码  
}
```

- JavaScript 循环语句  
如果用户希望一遍又一遍地运行相同的代码，并且每次的值都不同，那么使用循环是很方便的。JavaScript 支持不同类型的循环。  
for 语句：循环代码块一定的次数。



```
for ( 语句 1; 语句 2; 语句 3){  
    被执行的代码块  
}
```

语句 1（代码块）开始前执行，语句 2 定义运行循环（代码块）的条件，语句 3 在循环（代码块）已被执行之后执行。

for/in 语句：循环遍历对象的属性。

```
var person={fname:" John" ,lname:" Doe" ,age:25};  
for (x in person) // x 为属性名 {  
    txt=txt + person[x];  
}
```

while 语句：while 循环会在指定条件为真时循环执行代码块。

```
while ( 条件){  
    需要执行的代码  
}
```

do/while 语句：do/while 循环是 while 循环的变体。该循环会在检查条件是否为真之前执行一次代码块，然后如果条件为真的话，就会重复这个循环。

```
do{  
    需要执行的代码  
}  
while ( 条件);
```

- JavaScript 的 break 和 continue 语句

break 语句：break 语句可用于跳出循环，break 语句跳出循环后，会继续执行该循环之后的代码（如果有的话）。

```
for (i=0;i<10;i++) {  
    if (i==3) break;  
    x=x + "The number is " + i + "<b>";  
}
```

continue 语句：continue 语句中断循环中的迭代，如果出现了指定的条件，然后继续循环中的下一个迭代。

```
for (i=0;i<10;i++) {  
    if (i==3) continue;  
    x=x + "The number is " + i + "<b>";  
}
```

## 13.2 编辑控件样式代码

### 案例要求

通过案例实现控件样式多元化。

### 案例目的

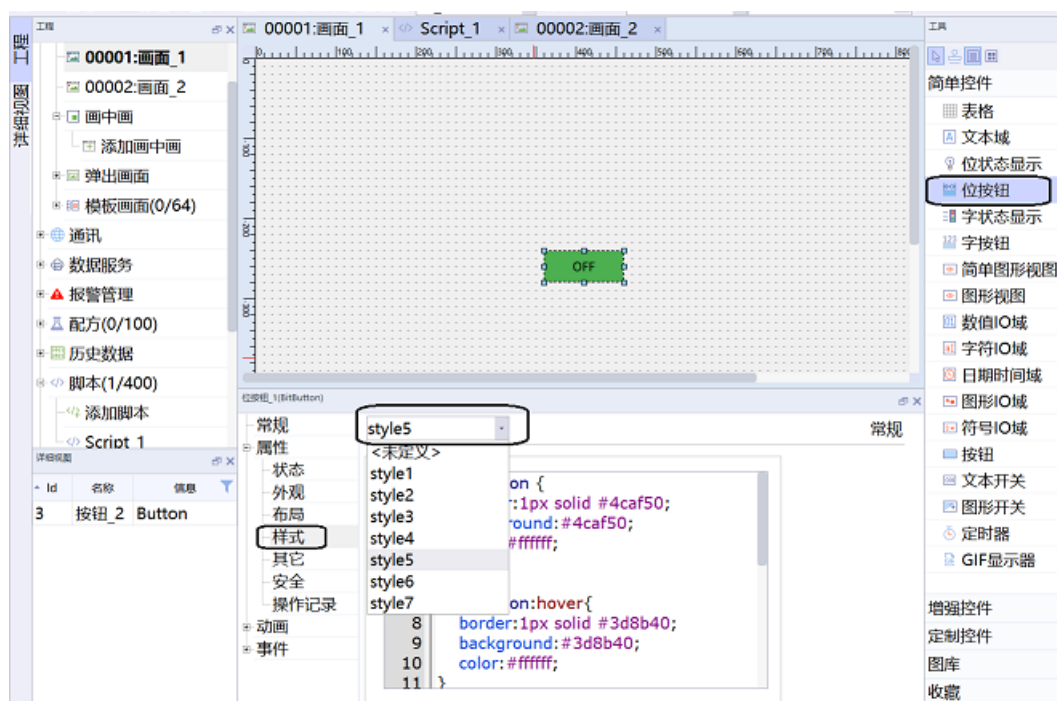
通过修改控件样式代码，来实现控件正常、悬浮、按压等状态下呈现出不同的样式，对HMI操作者对控件的操作更直观，画面更美观。

### 编程步骤

样式代码类似于CSS样式，属于网页前端范畴。

- 新建工程

在汇川IT7000新建一个工程，在软件界面右侧“工具”>“简单控件”>“位按钮”添加一个位按钮控件，单击控件，在“属性”>“样式”点击下拉菜单。可从下拉菜单中选取一项代码样式，此处以样式5为例。



- 样式介绍

```

style5
样式
1 QSSButton {
2   border: 1px solid #4caf50;
3   background: #4caf50;
4   color: #ffffff;
5 }
6
7 QSSButton:hover{
8   border: 1px solid #3d8b40;
9   background: #3d8b40;
10  color: #ffffff;
11 }
12
13 QSSButton:checked, QSSButton:focus,
14   border: 1px solid #3d8b40;
15   background: #3d8b40;
16   color: #ffffff;
17 }
18
19 QSSButton:disabled{
20   border: 1px solid #94cf96;
21   background: #94cf96;
22   color: #fae1fe;
23 }
    
```

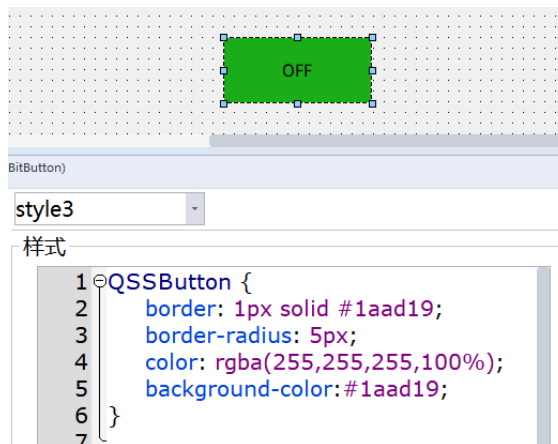
每个样式代码基本分为以下四个部分状态：

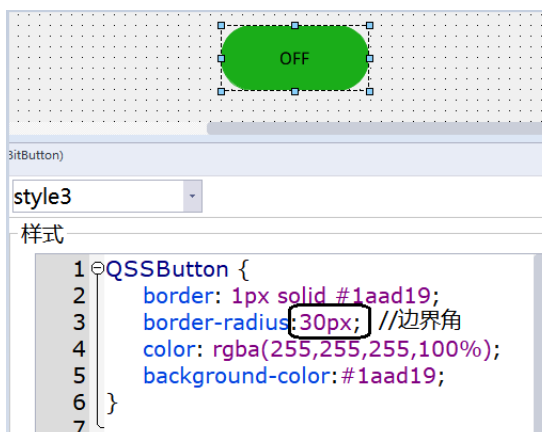
- Normal状态，即正常状态，没有干预的状态。
- Hover状态，即悬浮状态，鼠标悬停在按钮上时的状态。
- Focus/Press状态，即按压状态，焦点点击按钮下的状态。
- Disable状态，即禁用状态，禁用按钮失效下的状态。

每种状态基本分为以下三个部分：

- border边框形态，常见有直角边，圆角边。
  - Background背景色（填充色），填满按钮背景颜色。
  - Color前景色（字符的颜色），即为按钮上的文本颜色。
- 常用案例介绍
    - border圆角
 

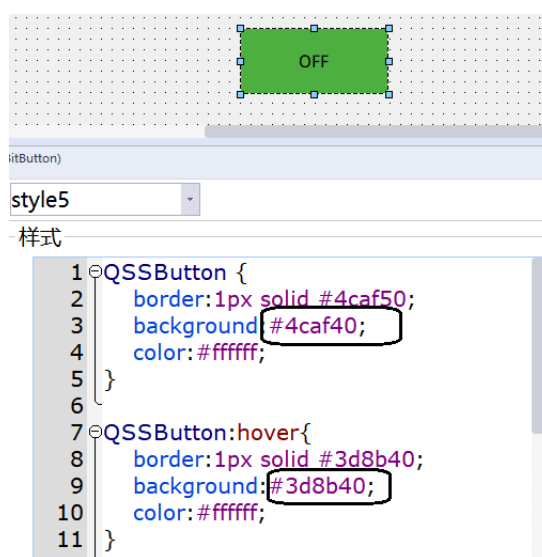
通过修改border(或border-radius)代码值可修改圆角半径增大圆角。



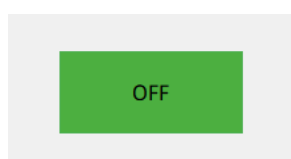


- 悬停background变色，操作更直观

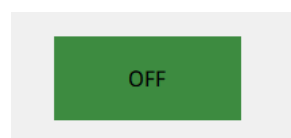
正常状态下和悬停状态下的颜色代码是不一样的，有如下效果，悬停后颜色会更亮，包括禁用状态也可以区分与前两种状态，让操作者更直观清楚此按钮目前禁用。



- 未放鼠标时

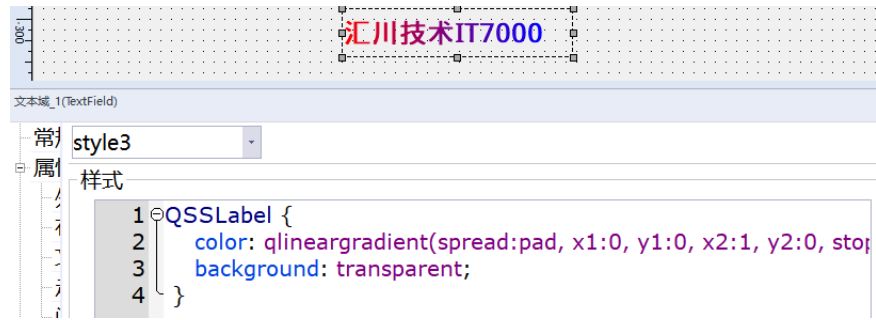


- 鼠标悬浮时



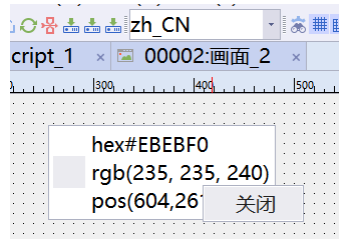
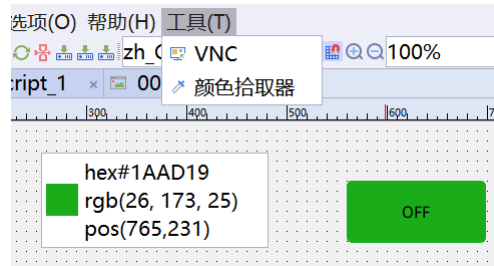
- color文本渐变色

为了让HMI更加美观，可以通过代码实现如下图所示的文本渐变色效果。



- 颜色提取器

控件样式中，很多情况下需要获取颜色参数，可通过选中“工具”中“颜色拾取器”，会跳出悬浮窗口，窗口左边正方形内的颜色即为鼠标悬停位置的颜色，对应颜色的代码会呈现在悬浮窗口上。当不需药使用时，右键选择关闭即可。



推进工业文明，共创美好生活

## 苏州汇川技术有限公司

Suzhou Inovance Technology Co., Ltd.

地址：苏州市吴中区越溪友翔路16号

总机：(0512)6637 6666

传真：(0512)6285 6720

<http://www.inovance.com>

## 深圳市汇川技术股份有限公司

Shenzhen Inovance Technology Co., Ltd.

地址：深圳市宝安区宝城70区留仙二路鸿威工业区E栋

总机：(0755)2979 9595

传真：(0755)2961 9897

<http://www.inovance.com>



19011545A00

由于本公司持续的产品升级造成的内容变更，恕不另行通知

版权所有 © 深圳市汇川技术股份有限公司

Copyright © Shenzhen Inovance Technology Co., Ltd.