

TwinCAT HMI 入门使用 V1.0

————— 杨煜敏 2018.09.01

本文档是第一稿，有任何问题和建议都可以及时反馈到 Automation 组
twincat@beckhoff.com.cn

本文档主要介绍了利用图形化编辑模式开发 TwinCAT HMI 的方法，全部功能都是基于 TwinCAT HMI 自带控件完成开发。

更多细节可以参考官方帮助文档：

http://infosys.beckhoff.com/content/1033/te2000_tc3_hmi_engineering/index.html?id=4989414722711336036

目录

一、	软件安装和拓扑结构	3
1.	系统要求	3
2.	软件安装	3
3.	拓扑结构	4
二、	框架介绍	6
1.	创建项目	6
2.	界面介绍	7
三、	入门操作	10
1.	文本框	10
2.	按钮	15
3.	线性测量条	20
4.	子页面	23
5.	自定义控件	27
6.	多语言切换	36
7.	主页面排版规划	41
8.	界面发布	45
9.	历史趋势图连接 PLC	52
10.	实时数据表格连接 PLC	59
11.	UserManagement 用户管理	65

一、 软件安装和拓扑结构

1. 系统要求

TwinCAT 版本	>3.1.4022.0
TwinCAT 等级	>TC1000 TC3 ADS
Visual Studio 版本	VS2013 >= 12.0.40629 Update 5 (Shell,Community,Pro) VS2015 >= 14.0.25431 Update 3 (Community,Pro)
操作系统	Windows 7/8/10 (32/64 位)
.NET 版本	>=4.5.2

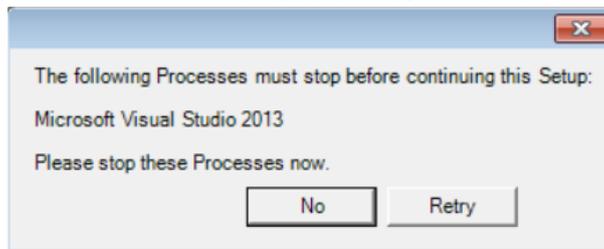
2. 软件安装

(1) 安装软件前关闭 Visual Studio， 这样才可以集成到 Visual Studio 环境中。

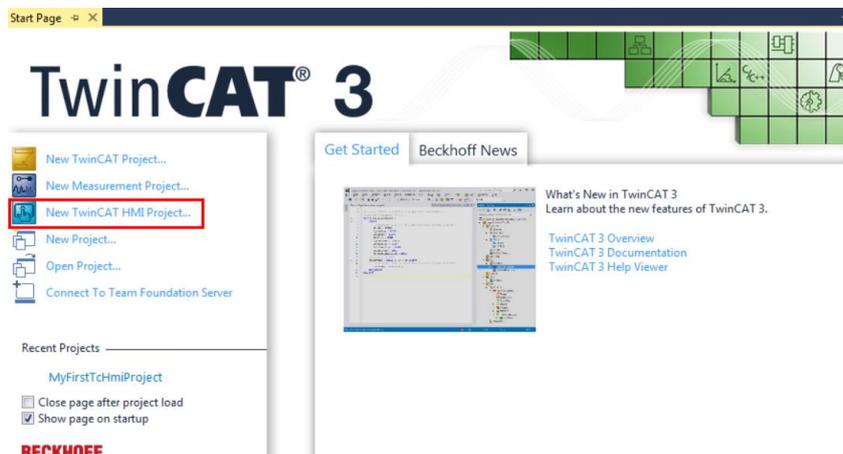
(2) 双击所下载的 TE2000-HMI-Engineering 安装包， 按照提示一步步安装。



(3) 如果在安装过程中 Visual Studio 没有关闭， 会弹出窗口提示关闭后才可以继续安装。

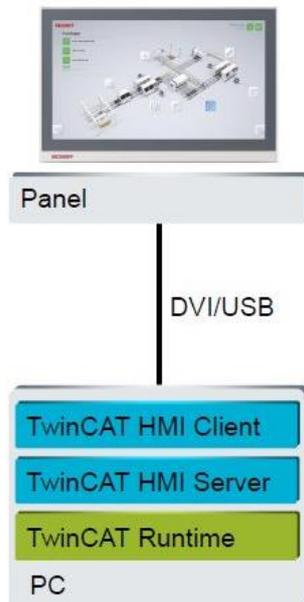


(4) 安装完成后就集成在 Visual Studio 中， 并在起始页面可以完成创建。

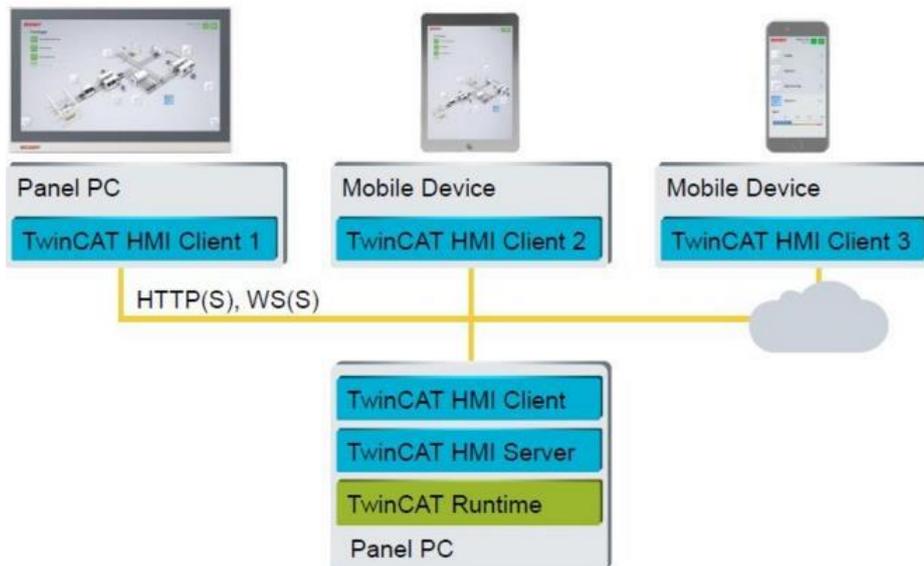


3. 拓扑结构

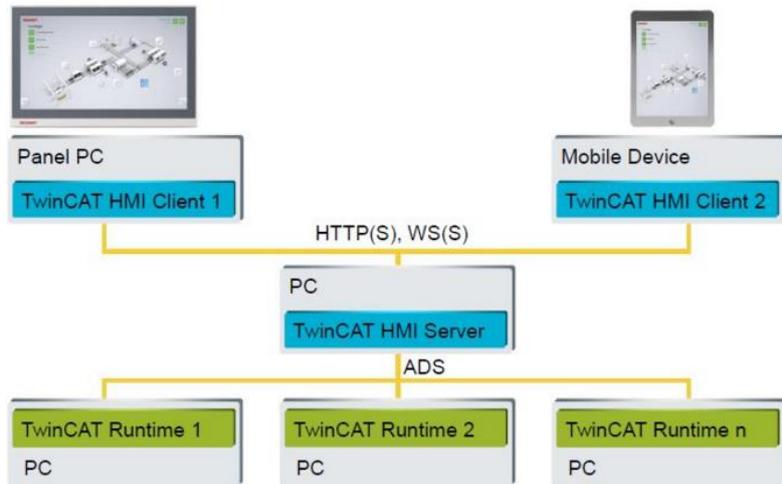
(1) Local-Client : Client, Server, Target(Runtime)在一台 IPC/EPC 中, 授权只需 TF2000。



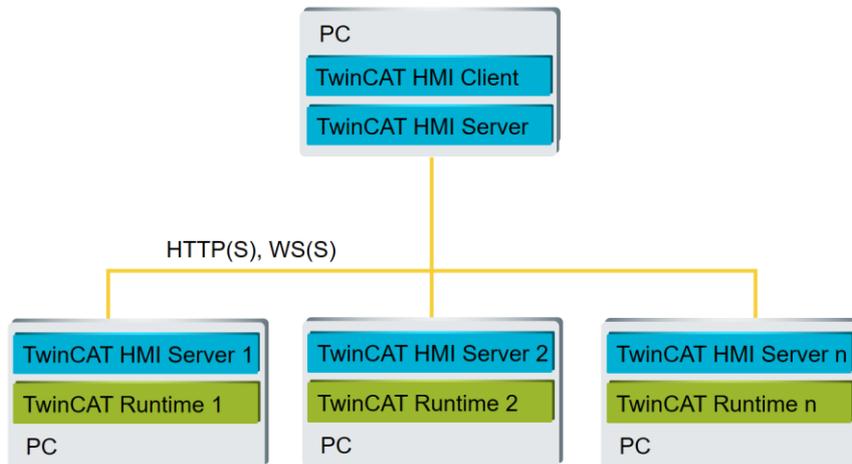
(2) Multi-client : 在第一个拓扑基础上增加多个 Client 显示画面, 无需再 Client 中安装任何软件, 只需在 Server 中增加 Client 授权即可。



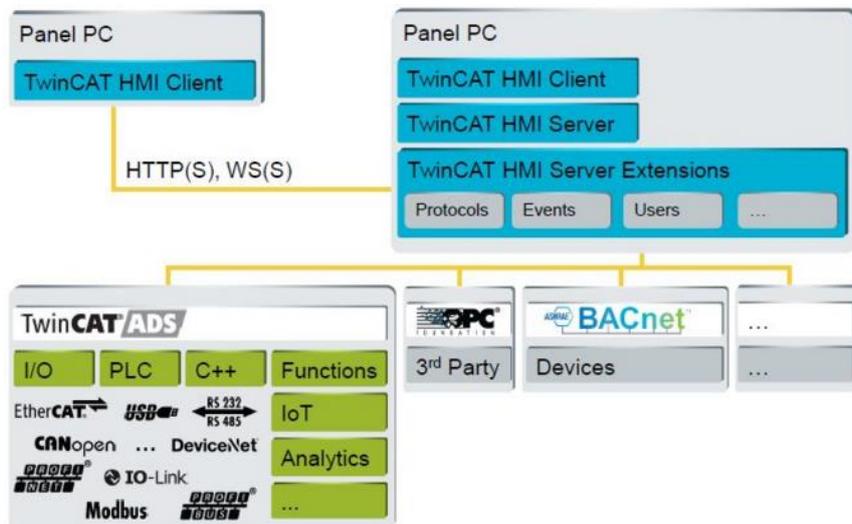
(3) Multi-runtime : Server 单独安装在一台 PC 中, 通过 ADS 连接 Target(Runtime), 需要在 Server 中增加 Client 以及 Target 授权。



(4) Multi-server : 一台带 Server 的 PC 分布式连接多台带 Server 的 PC, 这样可以显示本地 PC 中的 HMI, 也可以同时显示各个分布式 PC 中独立的 HMI, 类似 SCADA。



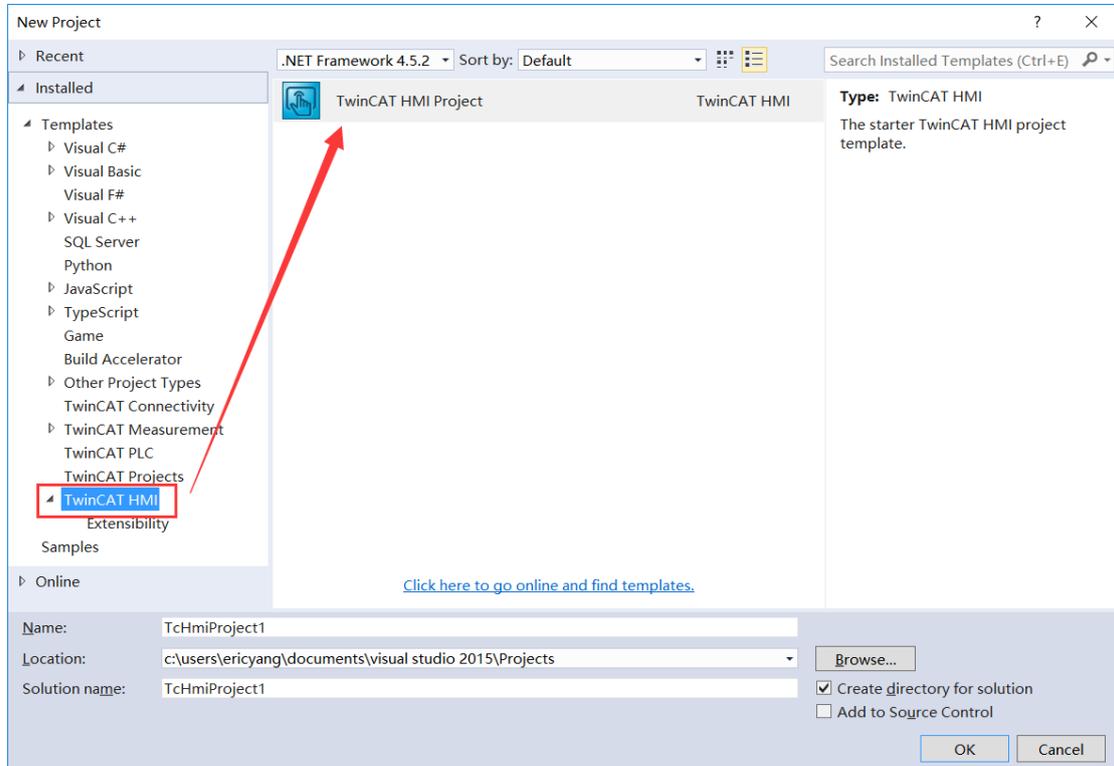
(4) Multi-protocol : 在 Server 中扩展协议/功能/服务, 可以通过 OPC UA、BACnet 等扩展协议连接第三方设备, 或者扩展配方/Scope 等 HMI 功能, 以及扩展 C#/Java 等高级语言参与 HMI 的开发。部分需要在 Server 中增加额外 TF21XX 系列 Function 授权。



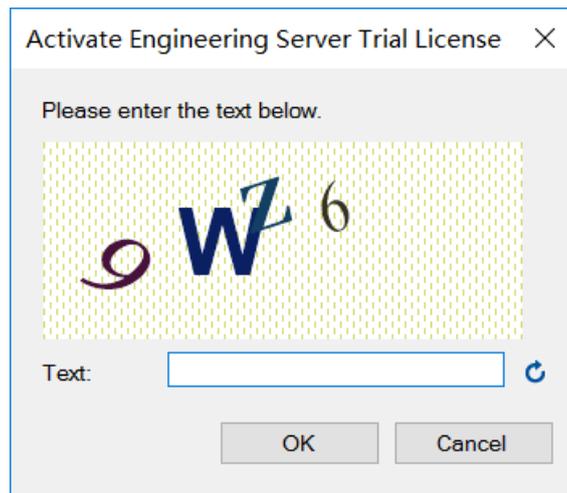
二、 框架介绍

1. 创建项目

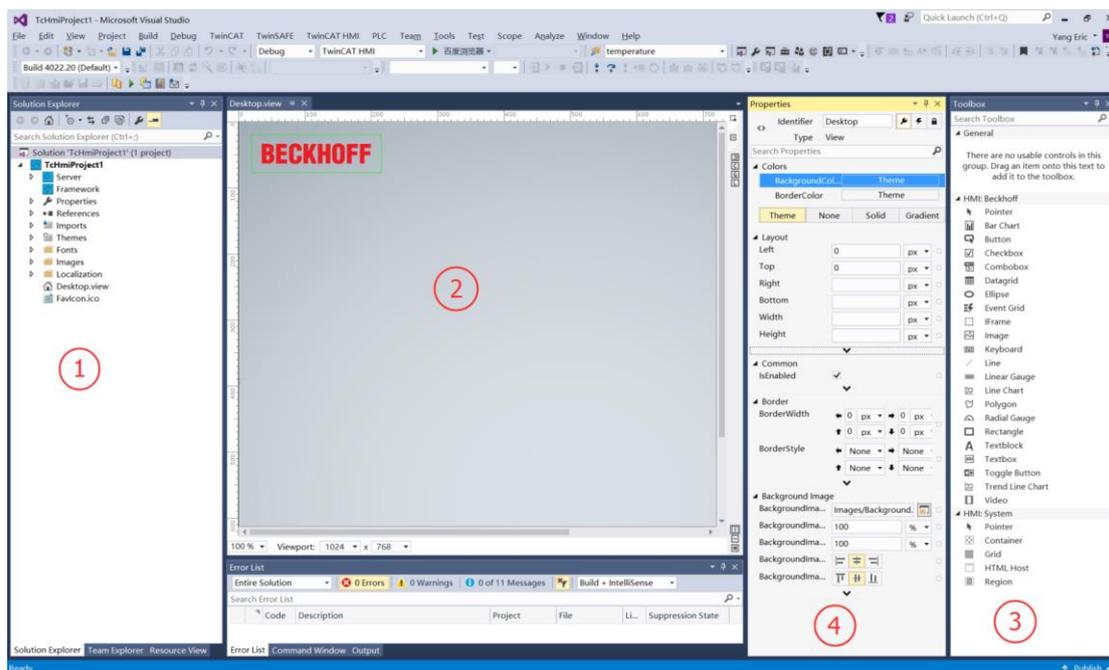
(1) 选择 TwinCAT HMI → TwinCAT HMI Project 创建项目。



(2) 弹出窗口激活试用版授权，并且完成项目创建。



2. 界面介绍

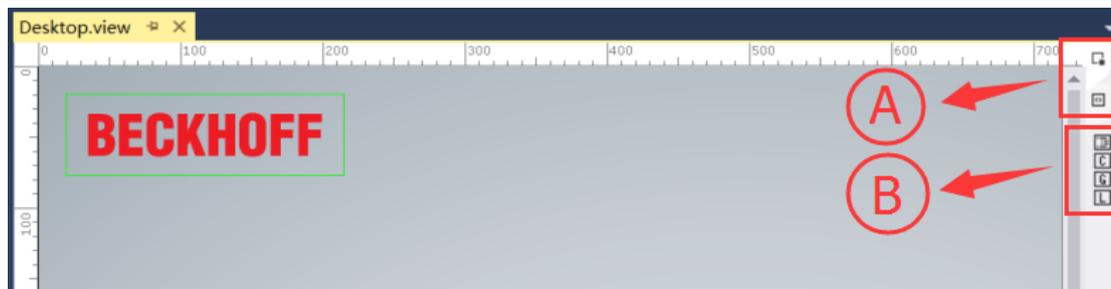


① Solution Explorer

TwinCAT HMI 项目管理区，用于创建画面，子页面，多语言，自定义控件，自定义功能（JavaScript）等。

② Desktop.view

TwinCAT HMI 前端开发窗口，默认图形编辑，在此窗口右上角有一排隐藏选型：



A. 可以切换图形编辑和代码编辑（基于 HTML5），在代码编辑模式下可以直接检查生成的代码。

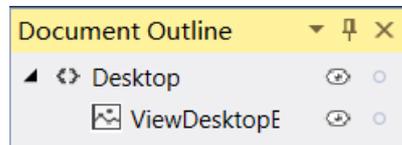
```

Desktop.view  X
1  <div id="Desktop" data-tchmi-type="tchmi-view"
2      data-tchmi-top="0" data-tchmi-left="0"
3      data-tchmi-width-mode="Content"
4      data-tchmi-min-width="100" data-tchmi-min-width-unit="%"
5      data-tchmi-height-mode="Content"
6      data-tchmi-min-height="100" data-tchmi-min-height-unit="%"
7      data-tchmi-background-image="Images/Background.png"
8      data-tchmi-background-image-height="100" data-tchmi-background-image-width="100"
9      data-tchmi-background-image-height-unit="%" data-tchmi-background-image-width-unit="%"
10
11  <div id="ViewDesktopBeckhoffLogo" data-tchmi-type="tchmi-image"
12      data-tchmi-src="Images/Beckhoff_Logo.svg"
13      data-tchmi-left="20" data-tchmi-top="20"
14      data-tchmi-width="194" data-tchmi-height="57">
15
16  </div>
17
18  </div>

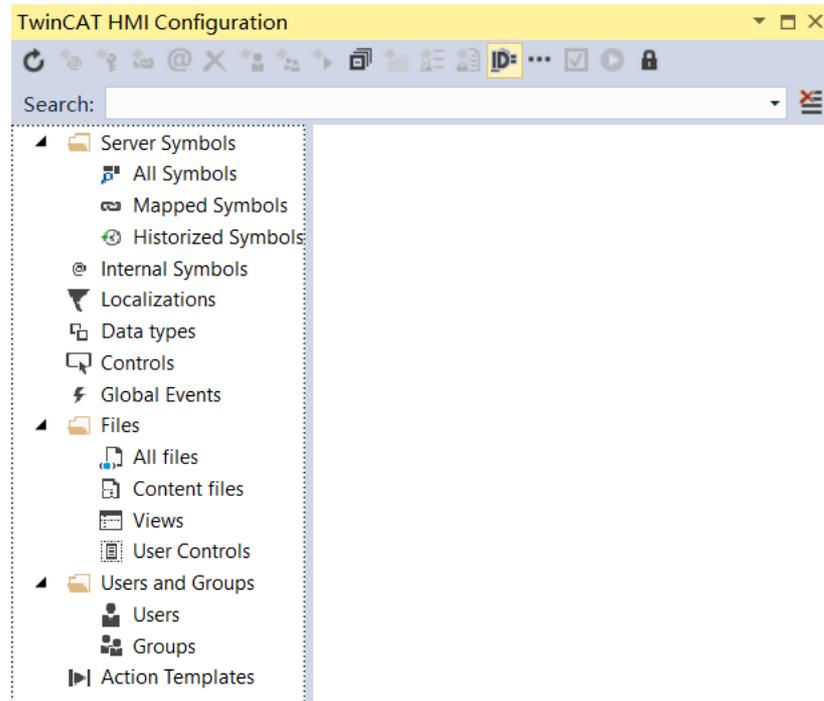
```

B. 从上到下依次有 4 个选项：

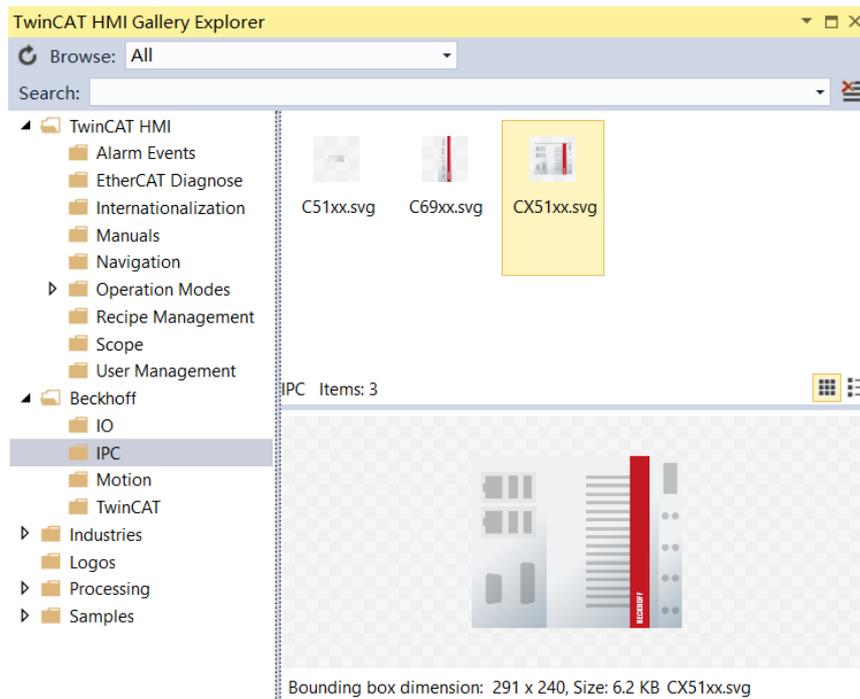
Document Outline：显示所有画面中控件，以及查看和配置从属关系。



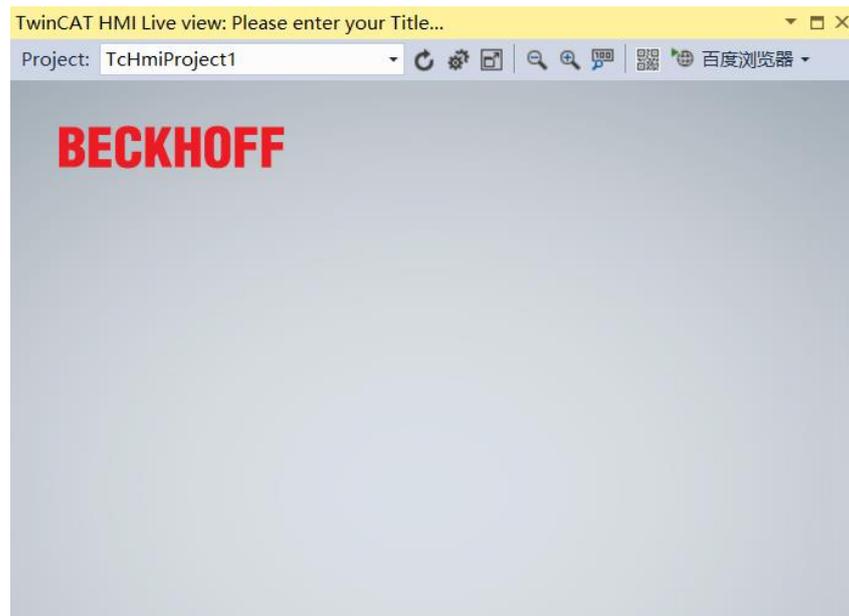
TwinCAT HMI Configuration：HMI 集中配置窗口，可以配置所有 HMI 中所用到的组件，例如：用户管理，数据类型，变量映射，多语言等等。



TwinCAT HMI Gallery Explorer：TwinCAT HMI 自带的所有图库，可以直接拖拉到 HMI 图形开发界面中使用。



TwinCAT HMI Live view : TwinCAT HMI 预览界面，无需发布，并且支持与图形化界面设计同步操作，做到所见即所得。



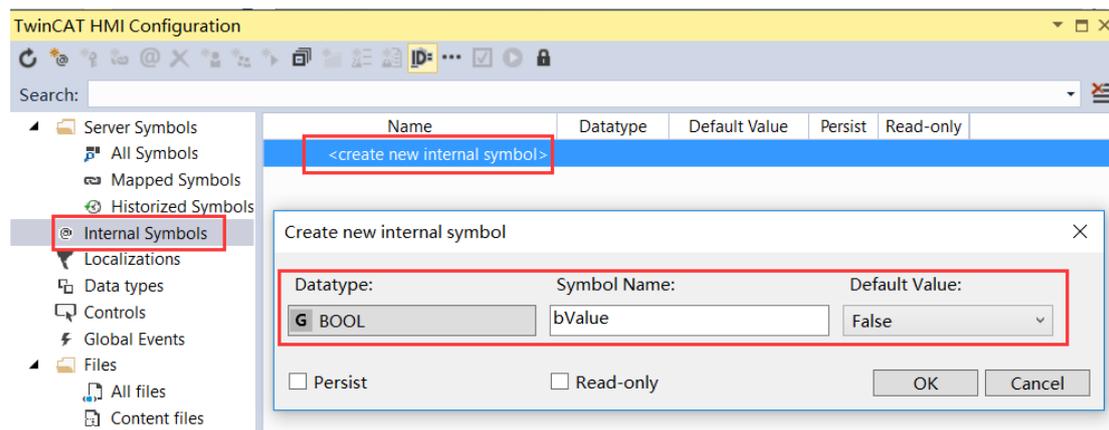
- ③ Toolbox
工具箱提供 TwinCAT HMI 所有可用的控件，可以直接拖拉到图形化设计窗口中。
- ④ Properties
属性栏可以对所选控件的属性、事件完成设置。

三、入门操作

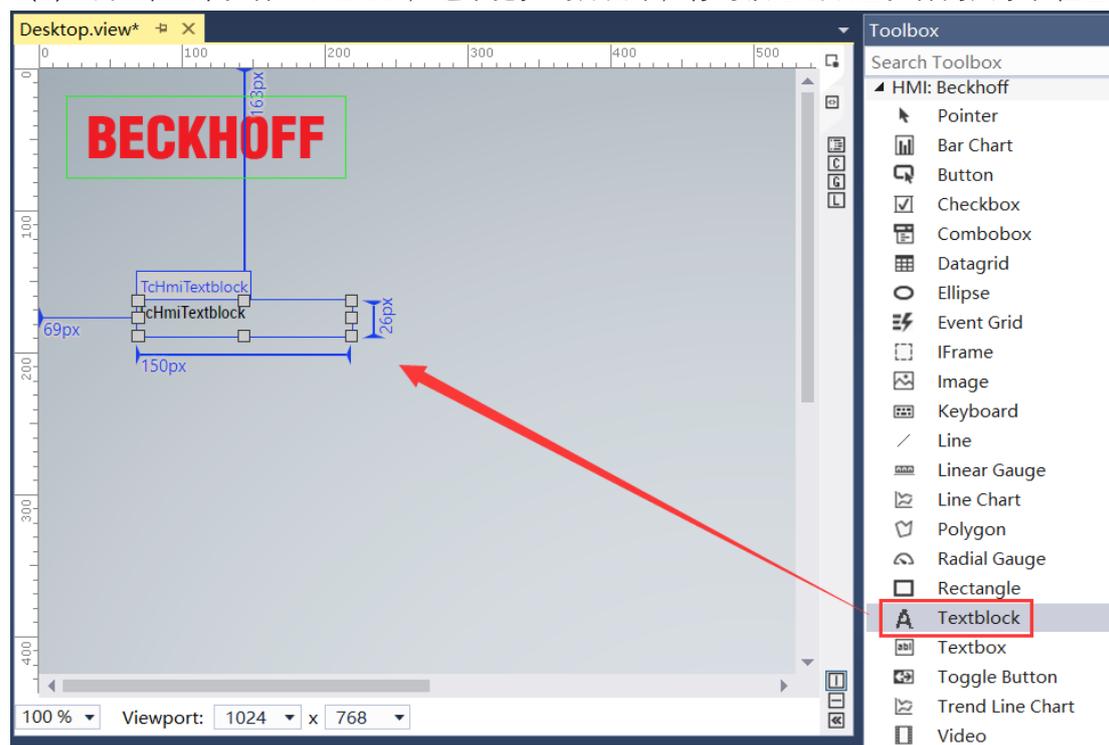
通过一个案例让大家了解 TwinCAT HMI 中常用功能的使用。

1. 文本框

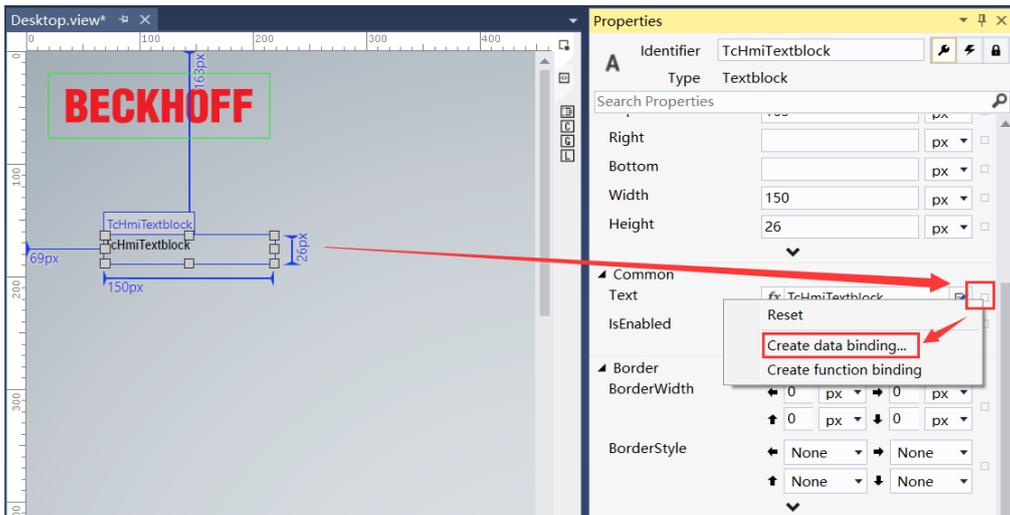
(1) 首先打开 TwinCAT HMI Configuration，在 Internal Symbols 中双击<create new internal symbol>，创建 HMI 所需要的一些内部变量。



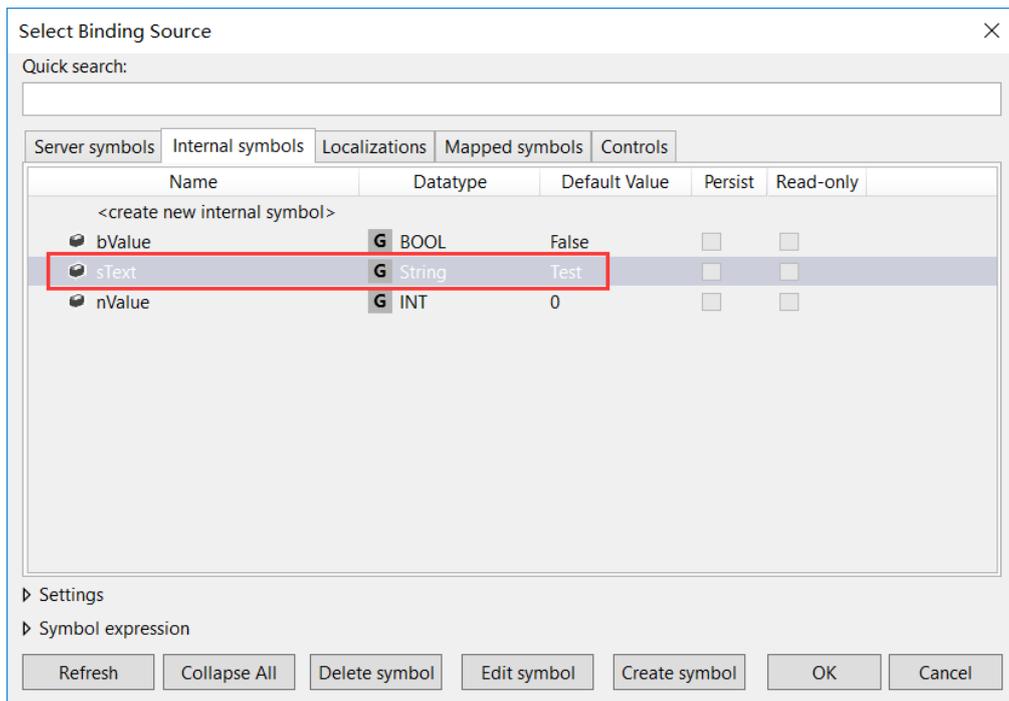
(2) 添加第一个控件 Textblock，选中拖放到界面中，你可以适当调整控件的大小和位置。



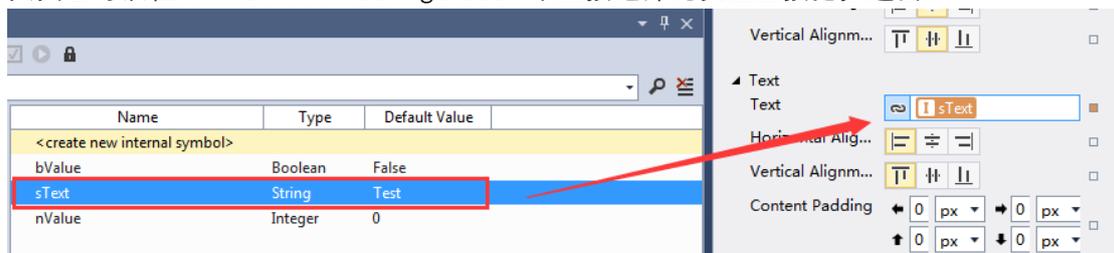
(3) 选中此控件在属性栏中找到 Text, 点击旁边的小方块, 选择 Create Data Binding。



(4) 在 Internal symbols 中选中 sText 后点击 OK。



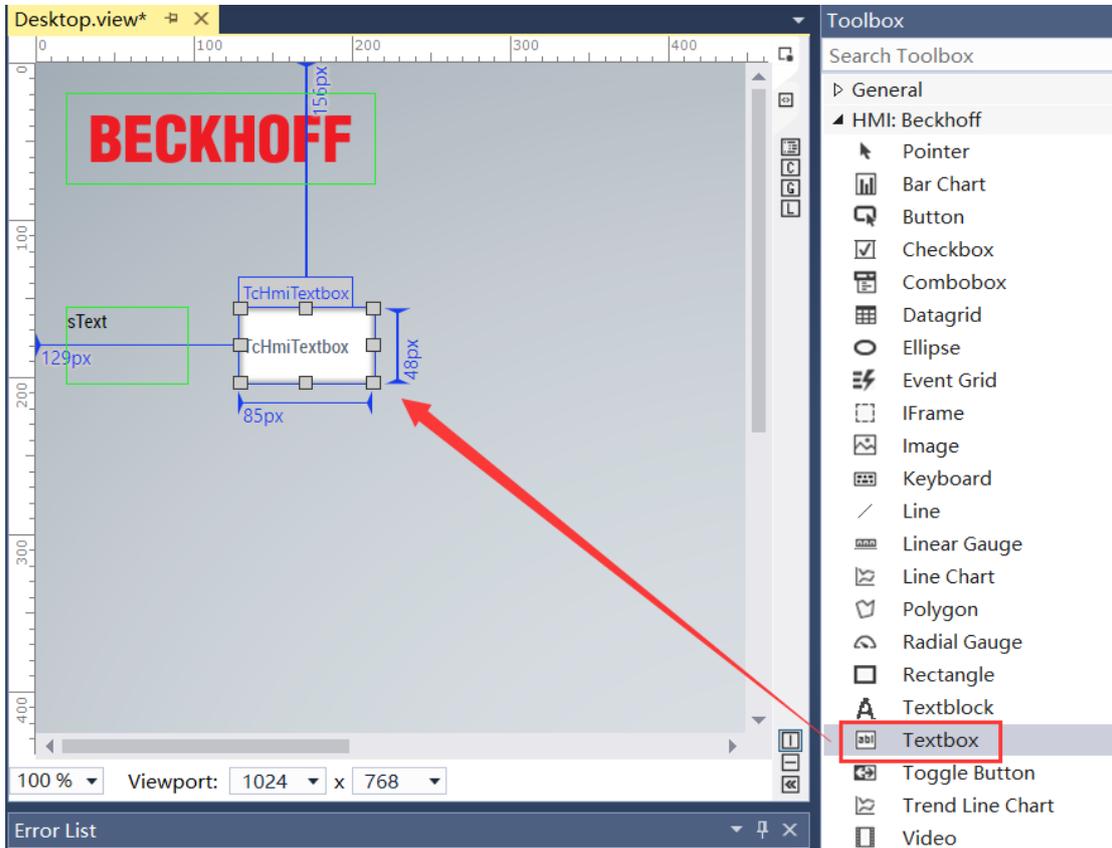
其实也可以在 TwinCAT HMI Configuration 中直接选择此变量直接拖拉进去。



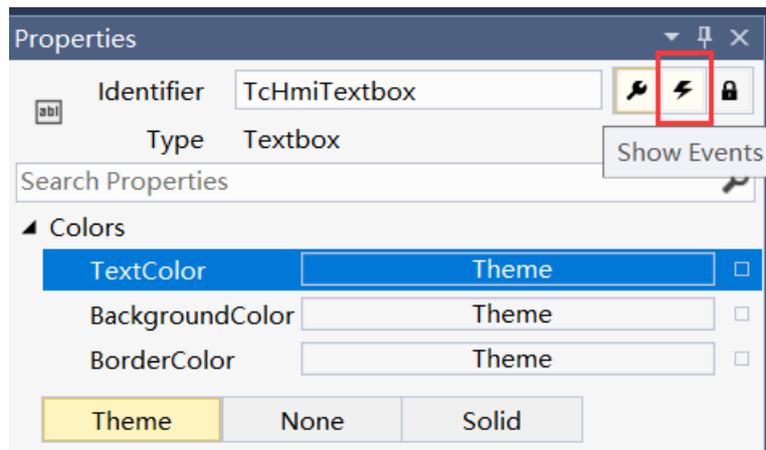
或者直接在 Text 中手动输入



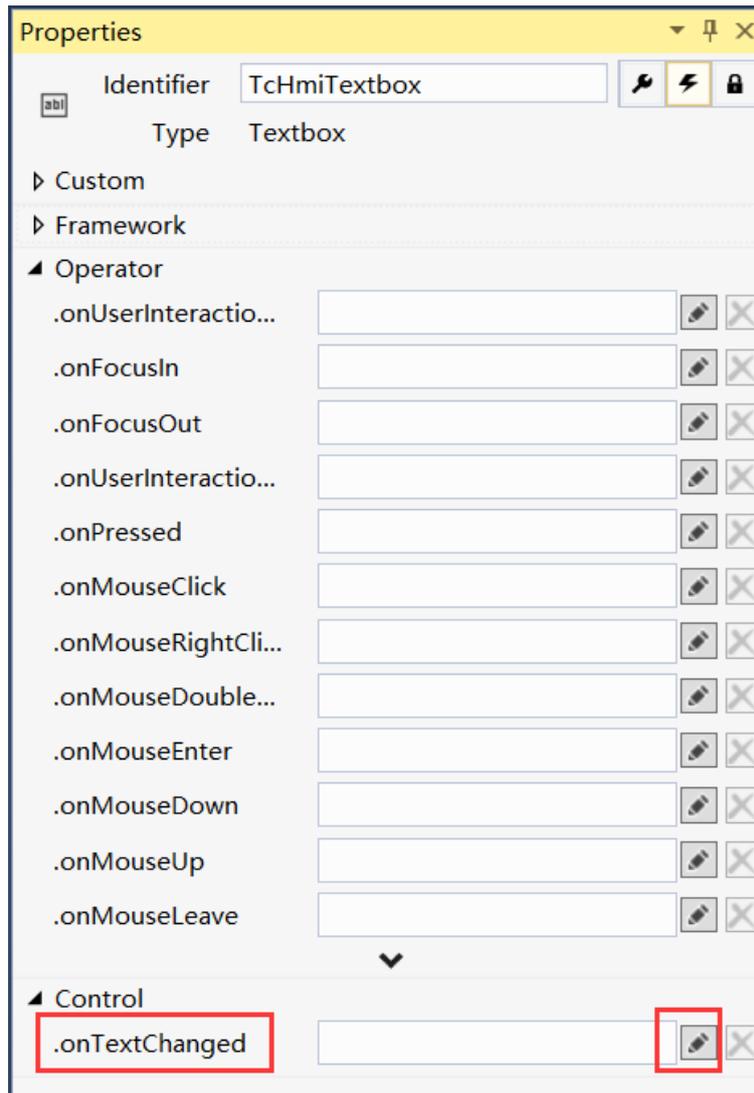
(5) 然后拖拉 Textbox 控件到界面中。



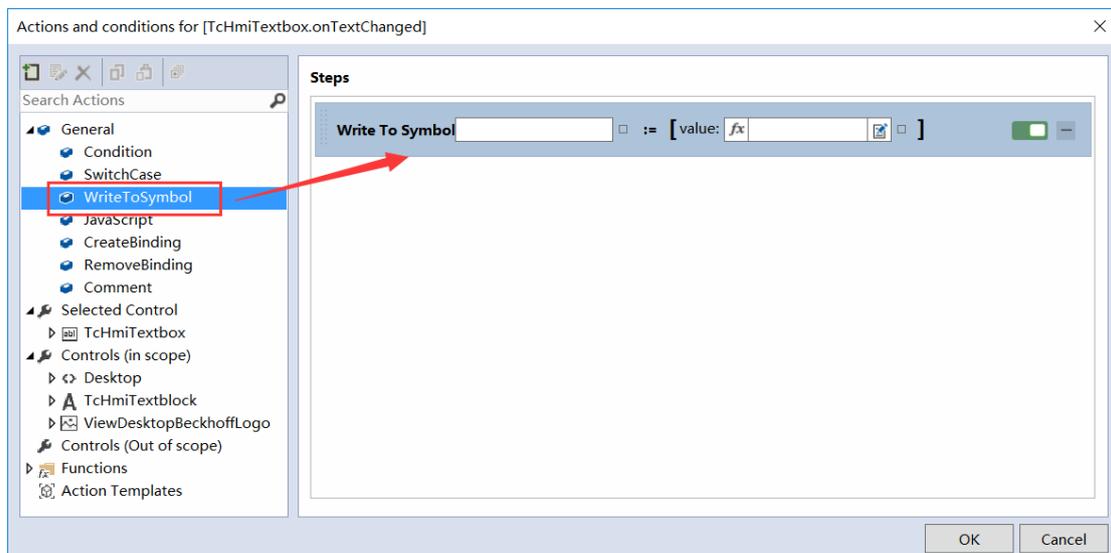
(6) 选中此控件，在属性栏切换到 Show Events。



(7) 找到.onTextChanged，进入修改。

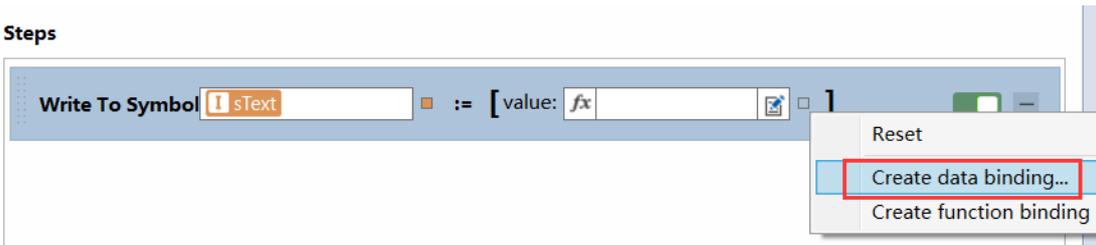


(8) 在 General 中找到 WriteToSymbol, 双击添加到 Steps 中。

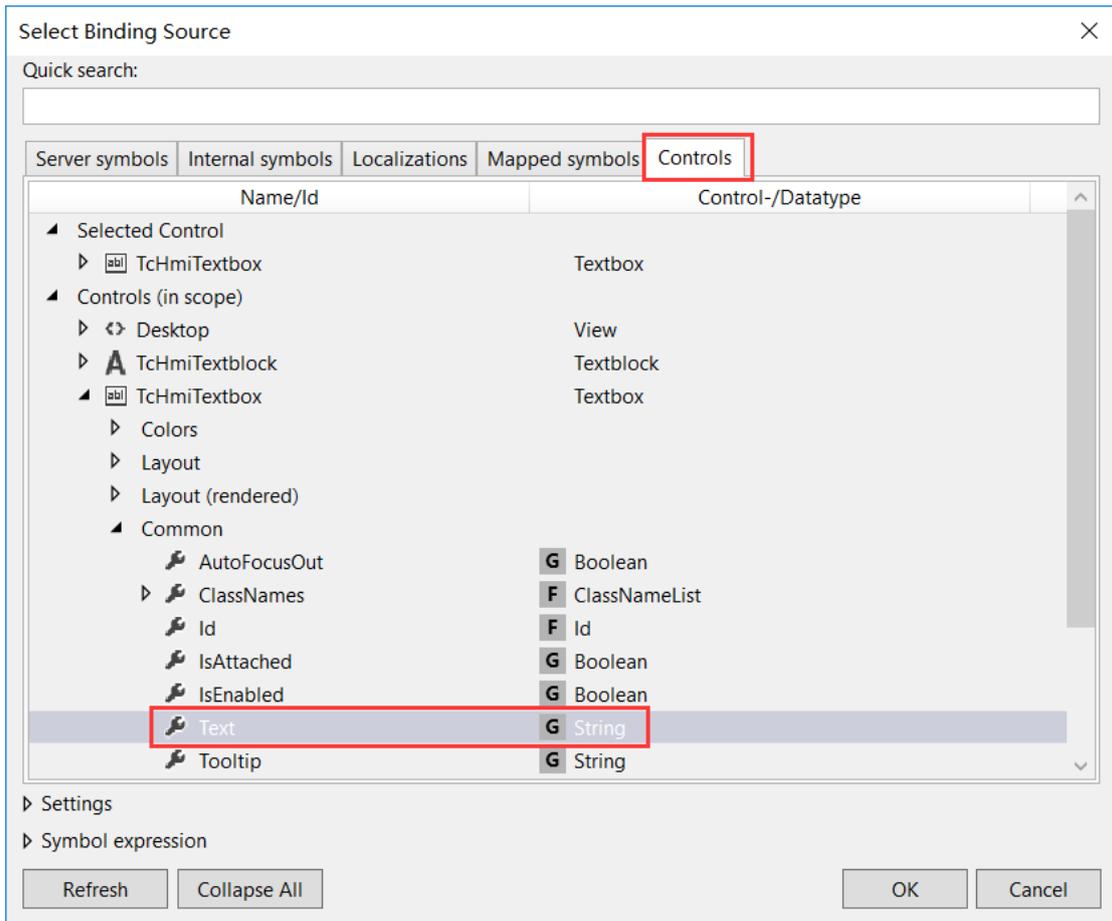


(9) 修改输入框为 sText 变量随后点击 Create Data Binding。

Steps



(10) 选择 Controls→Controls (in scope) →TcHmiTextbox→Common→Text, 点击 OK

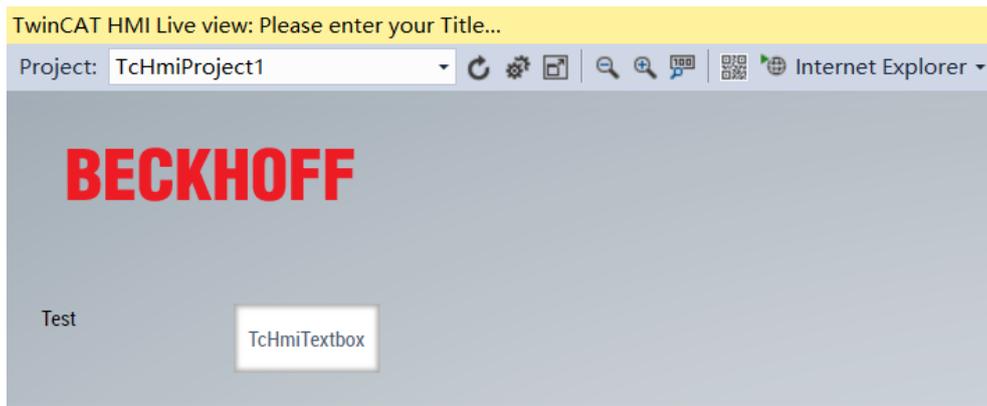


(11) 配置好事件后会出现一个绿色小条目。

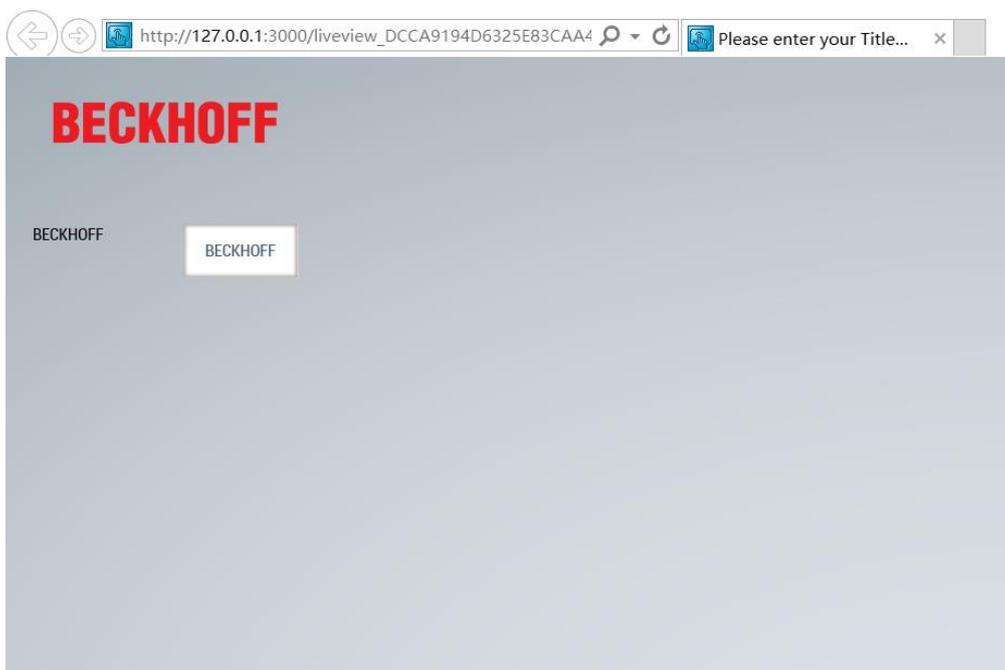
Tips : 每一个事件都可以点击绿色条拖到任意一个动作中去, 也可以 ctrl+ 拖拉赋值到其他动作中去。



(12) 做到这里我们可以先模拟下界面是否实现了我们所需的功能, 点击主界面旁边有个 L 的小方块, 可以实现在线浏览。

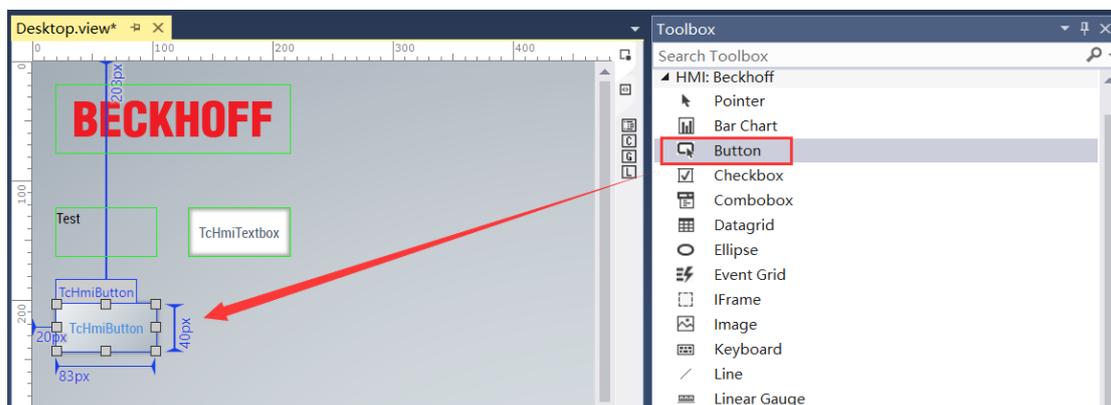


(13) 也可以通过浏览器打开此界面，在工具栏中找到  Internet Explorer，点击即可。

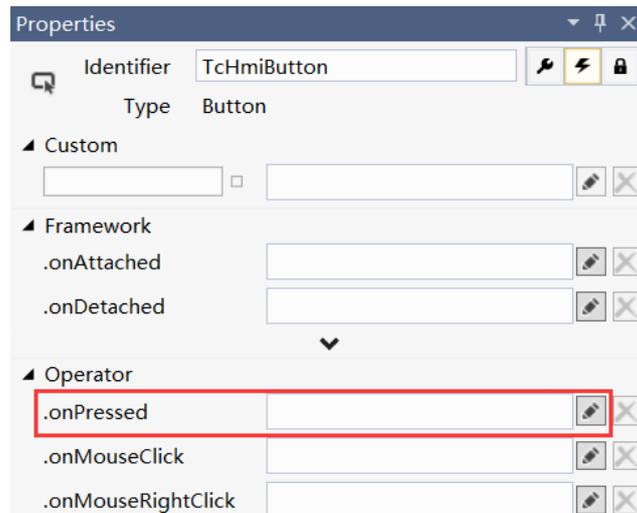


2. 按钮

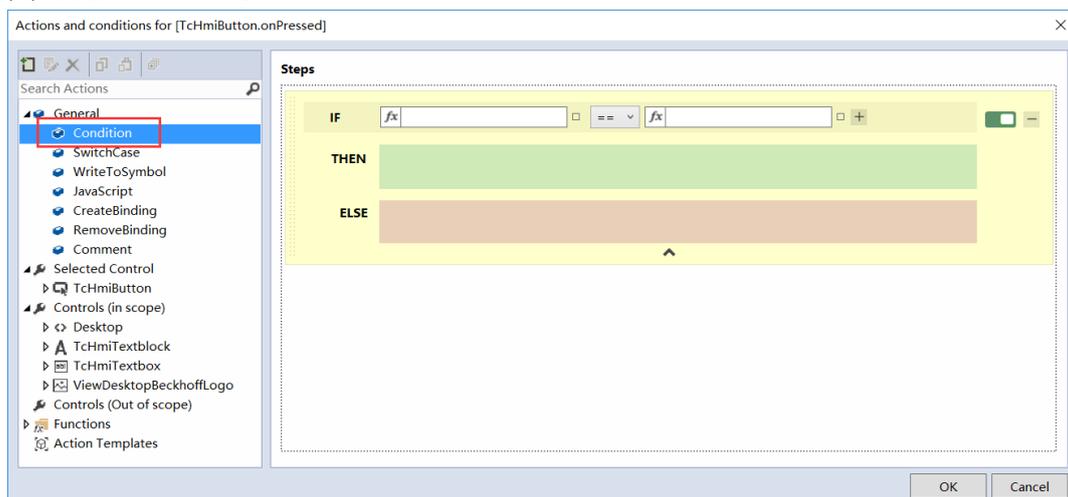
(1) 继续添加控件 Button。



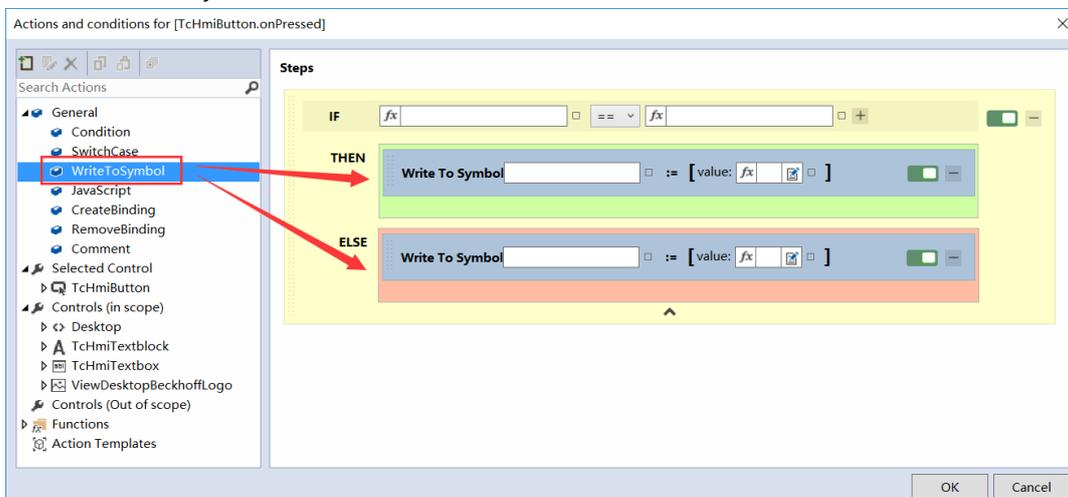
(2) 在属性栏配置.onPressed，或者双击控件也可以直接对.onPressed 进行配置。



(3) 添加 Condition。

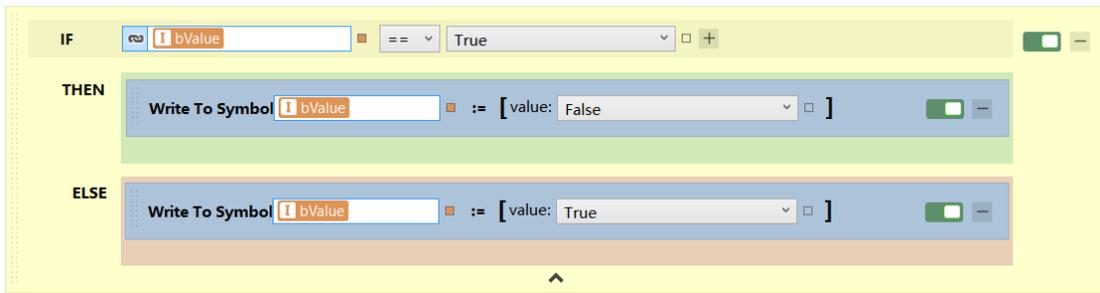


(4) 用 WriteToSymbol 分别执行两个条件。

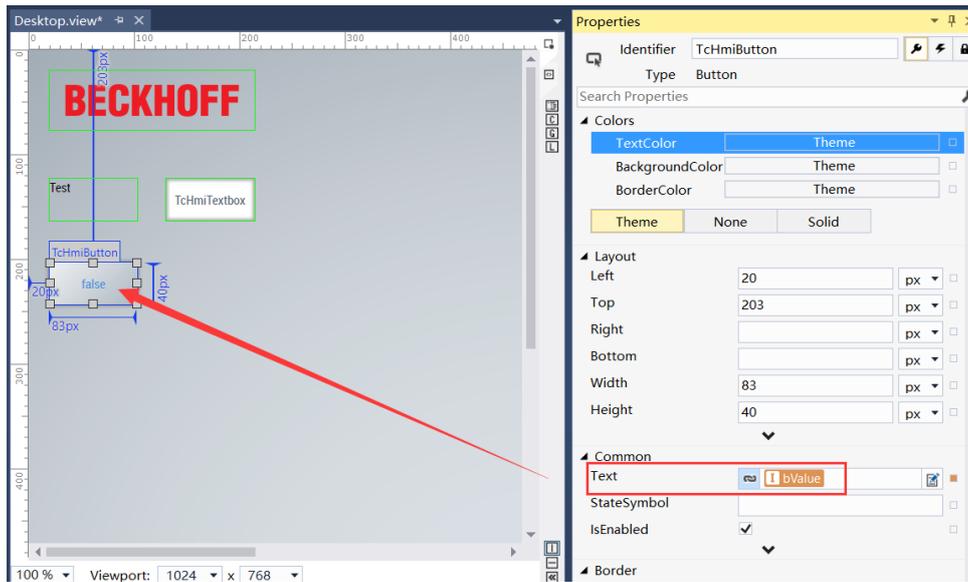


(5) 随后用 `bValue == True` 完成条件出发。

Steps



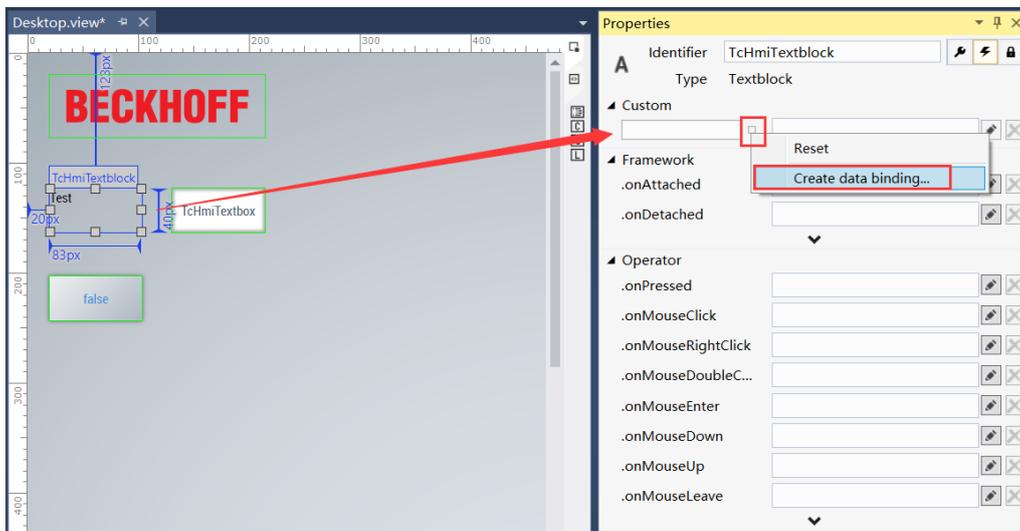
(6) 最后 Button→Text 中填写 `bValue`，显示当前值



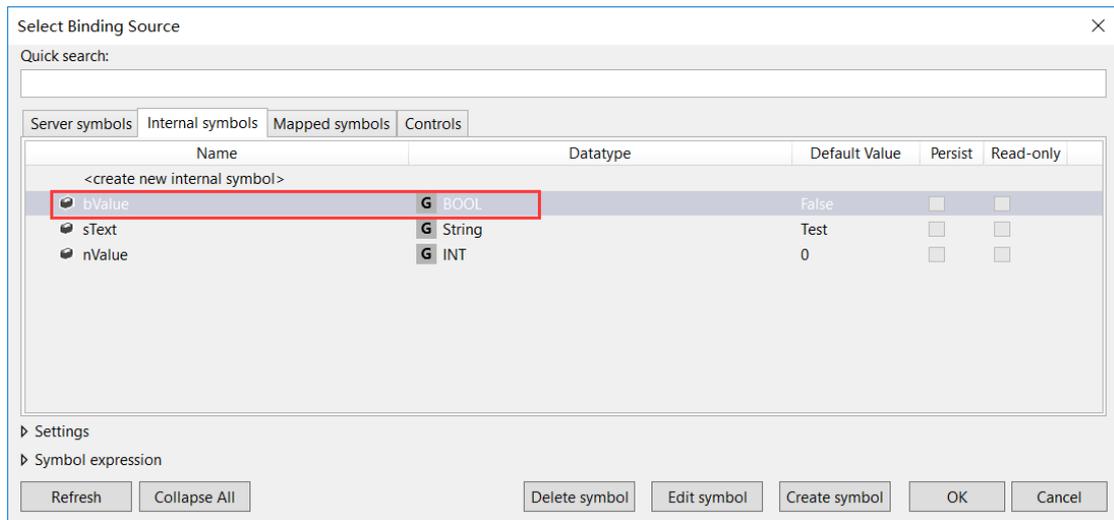
做完可以在线模拟下界面效果。

(7) 紧接着我们做一个功能，用按钮切换 Textblock 颜色。

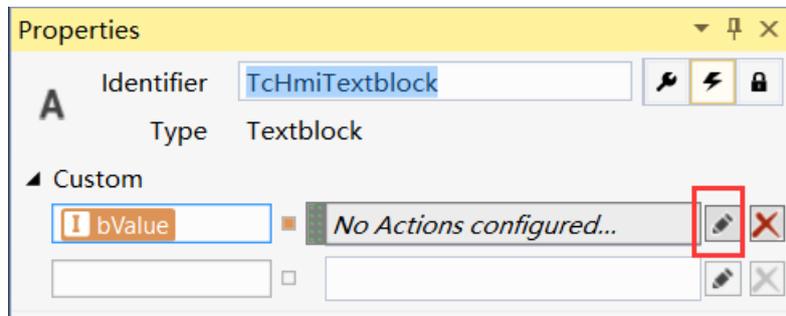
选中 Textblock，在 Events 中找到一个自定义项目，点击小方块，选择 Create Data Binding 进行配置。



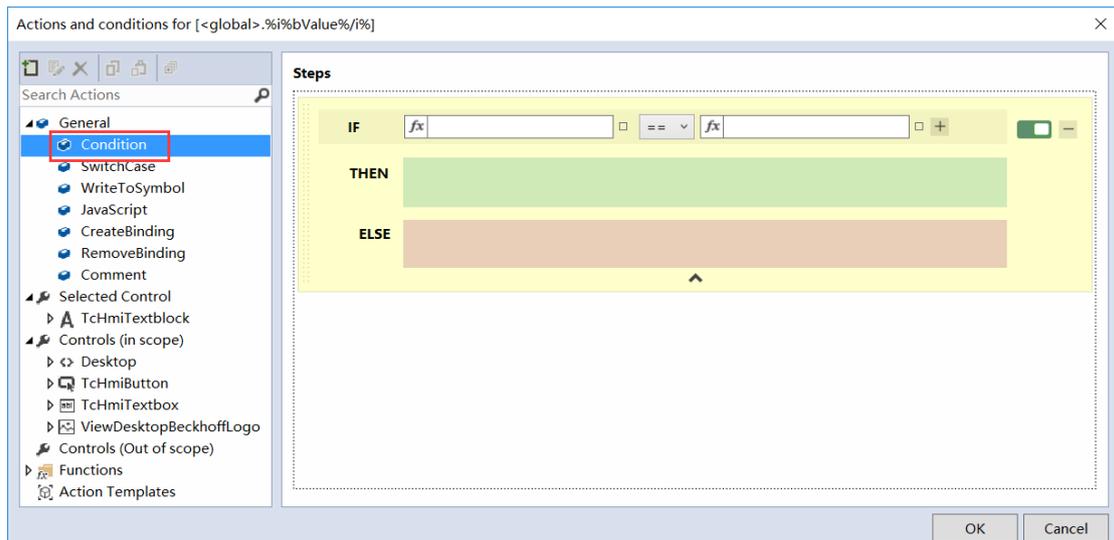
(8) 选择 bValue，意思是用此变量来触发一个事件。



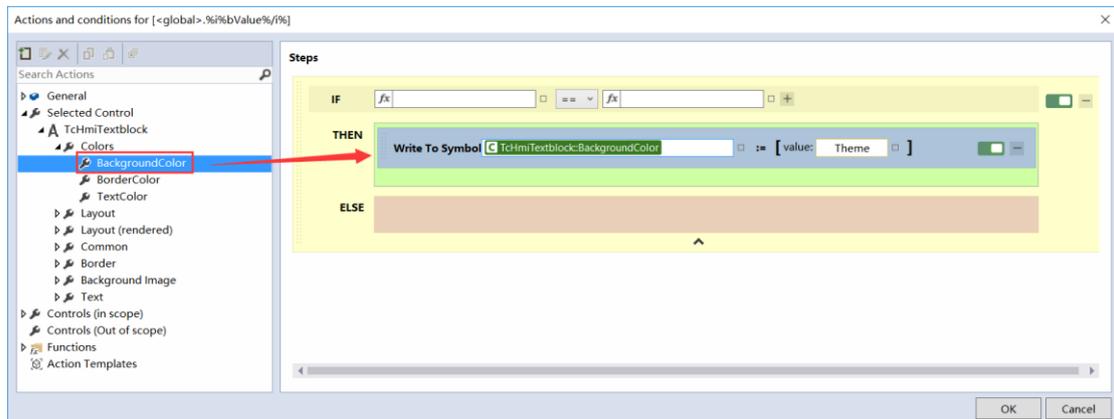
(9) 继续对自定义事件进行配置。



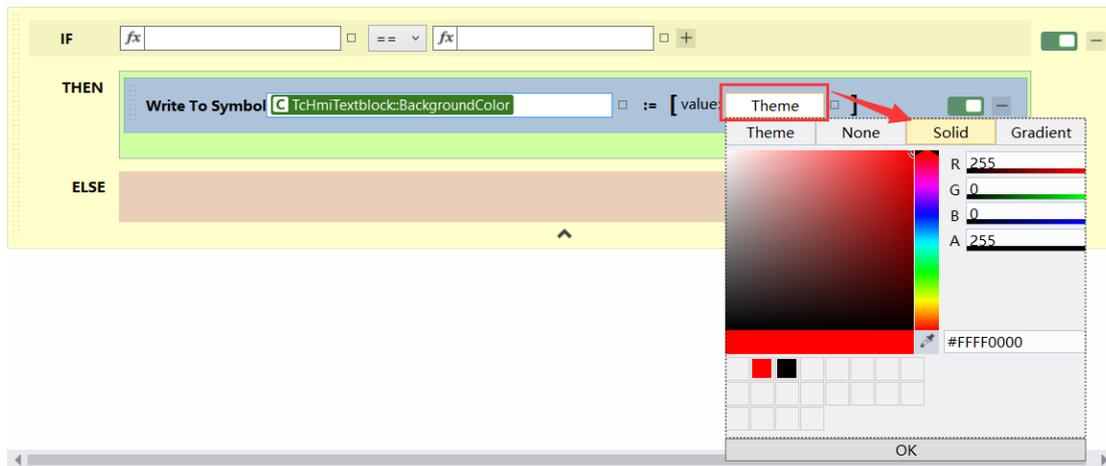
(10) 添加 Condition。



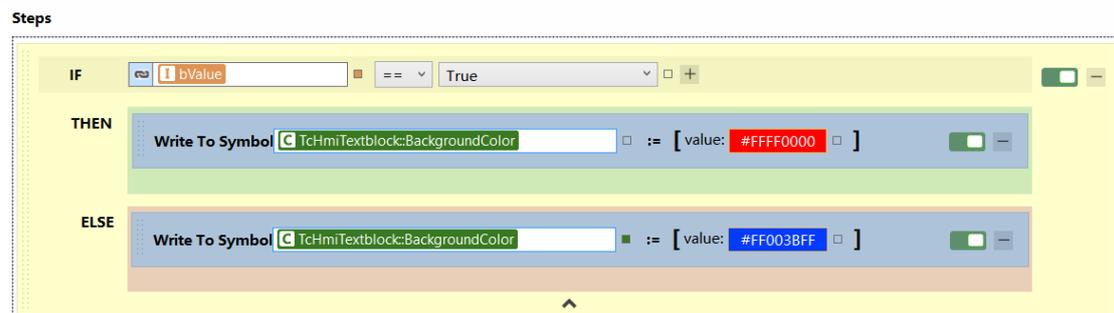
(11) 随后找到 Selected Control→TcHmiTextblock→Colors→BackgroundColor 添加到条件中去。



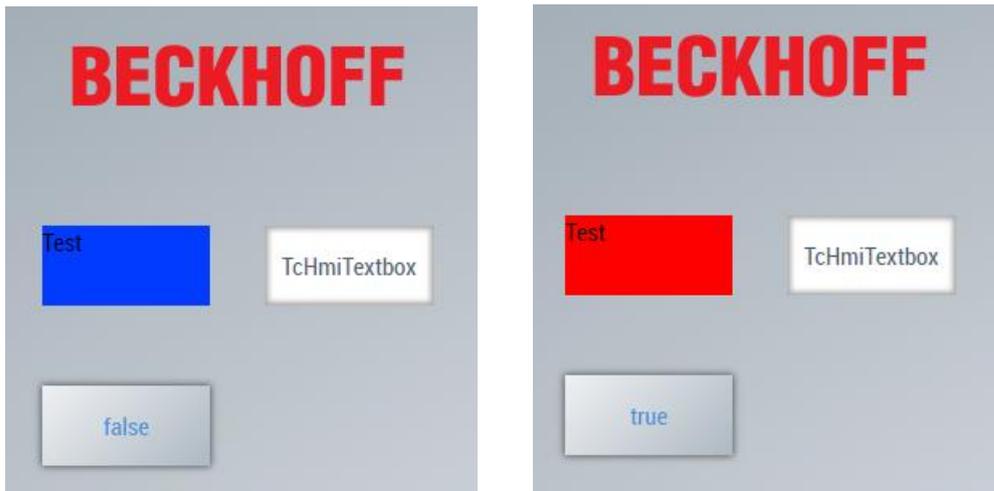
(12) 点击 Theme, 在 Solid 中选择一个颜色。



(13) 同样把另一个条件也配置一个颜色, 并且用 bValue == True 进行条件触发。

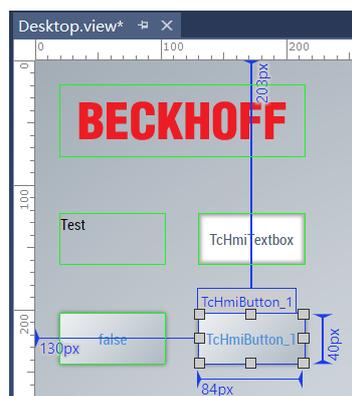


(14) 在线浏览效果

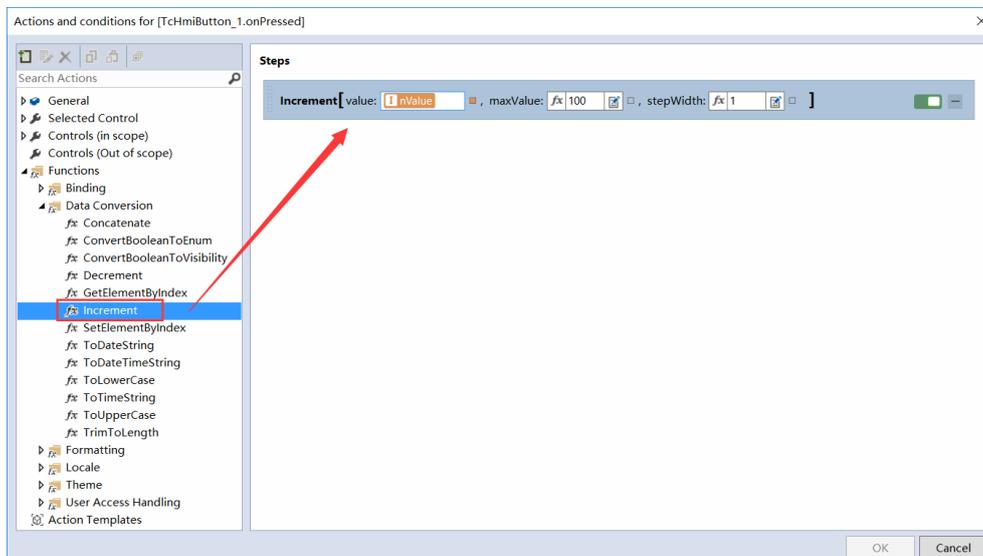


3. 线性测量条

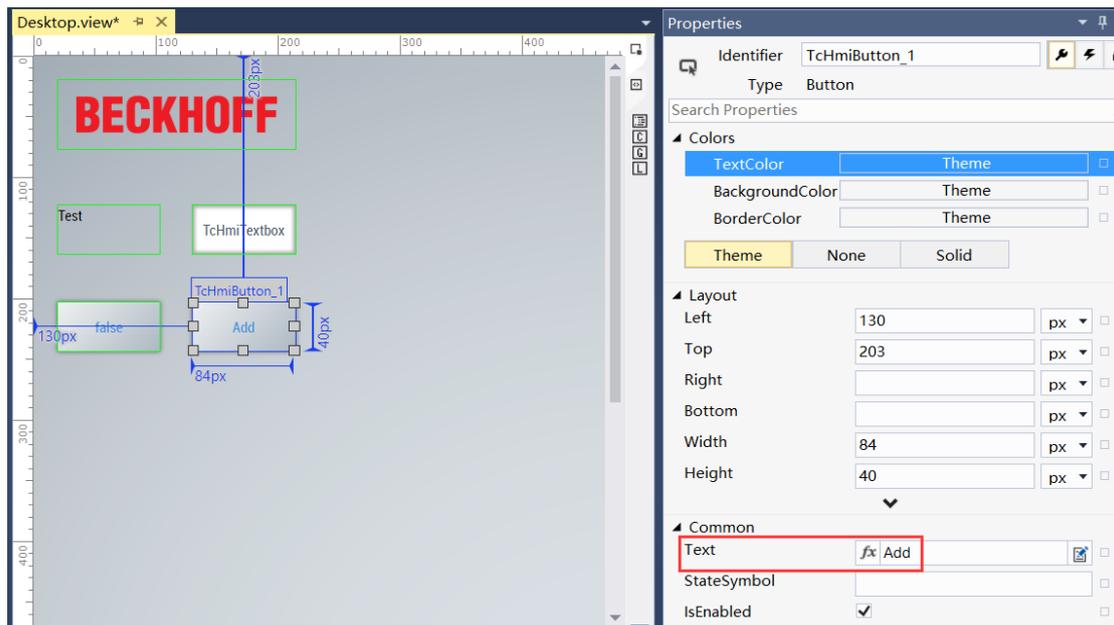
(1) 再添加一个 Button。



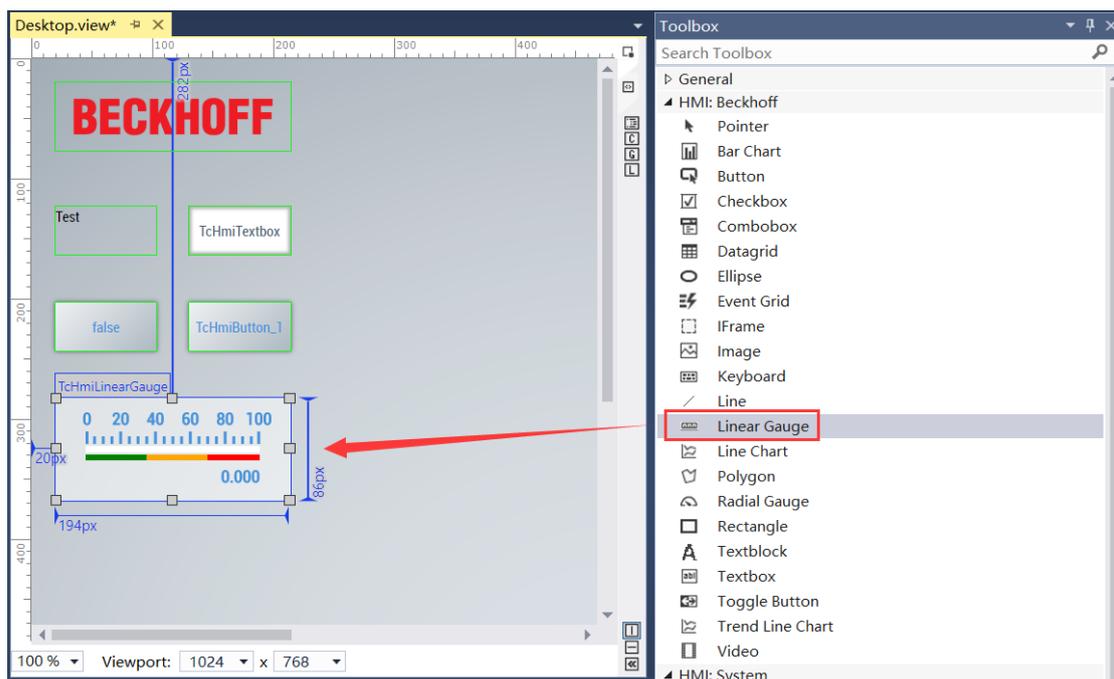
(2) 双击 button，可以直接配置.onPressed（默认），添加 Increment 功能并且设置每按一下使得 nValue 增加 1 的功能，并且配置最大值为 100。



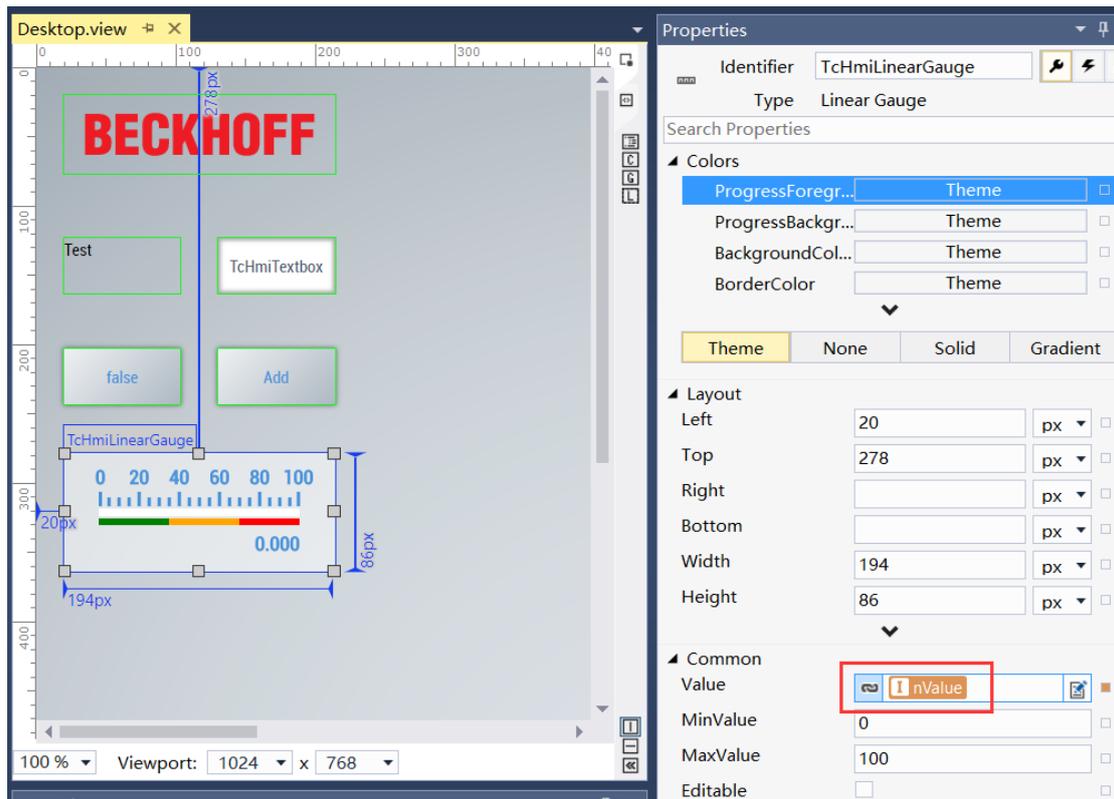
(3) 并且把这个 Button 取名为 Add。



(4) 为了验证此效果，可以加入一个计量器辅助观察，添加 Linear Gauge。



(5) 选中 Linear Gauge, 在属性中配置 Value 为 nValue。

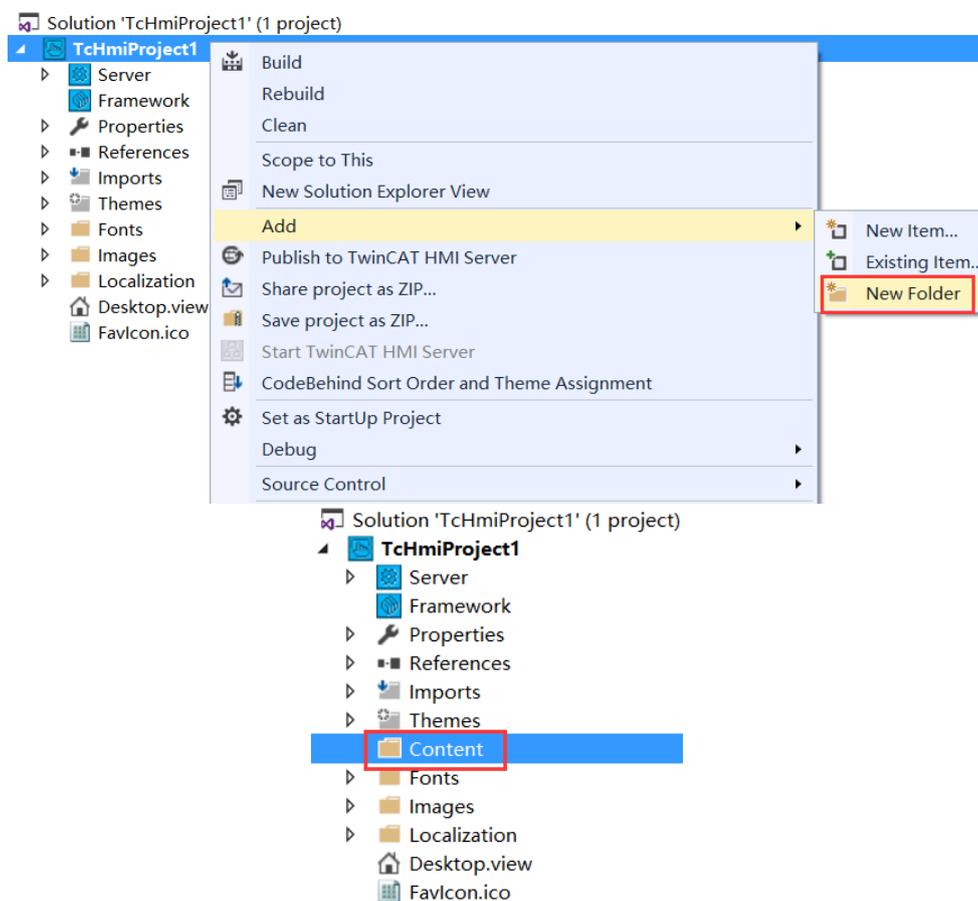


(6) 效果如下, 每按一次 Add, 计量器就增加 1, 直到 100 为止。

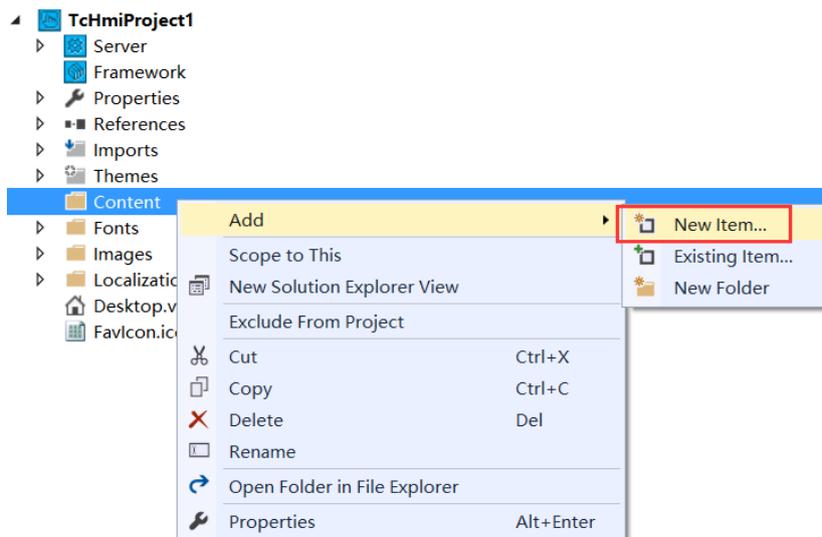


4. 子页面

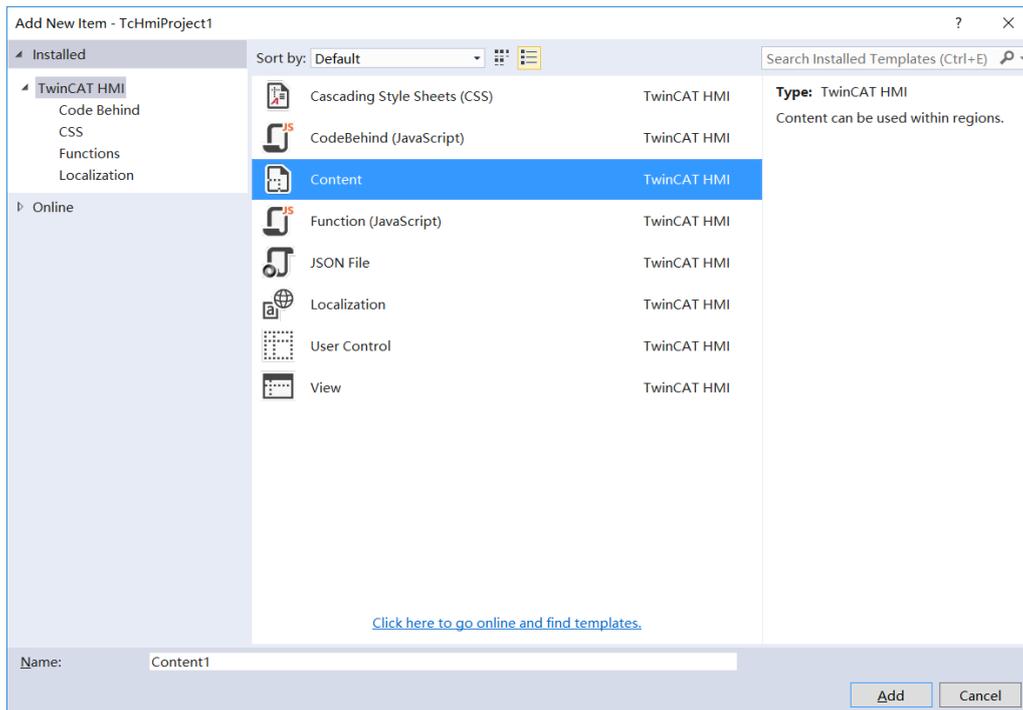
(1) 在项目中创建一个新的文件夹，取名为 Content，我们准备在这个文件夹里面做一些子画面，随后可以在主画面中进行切换，调用。



(2) 右键文件夹添加新项。



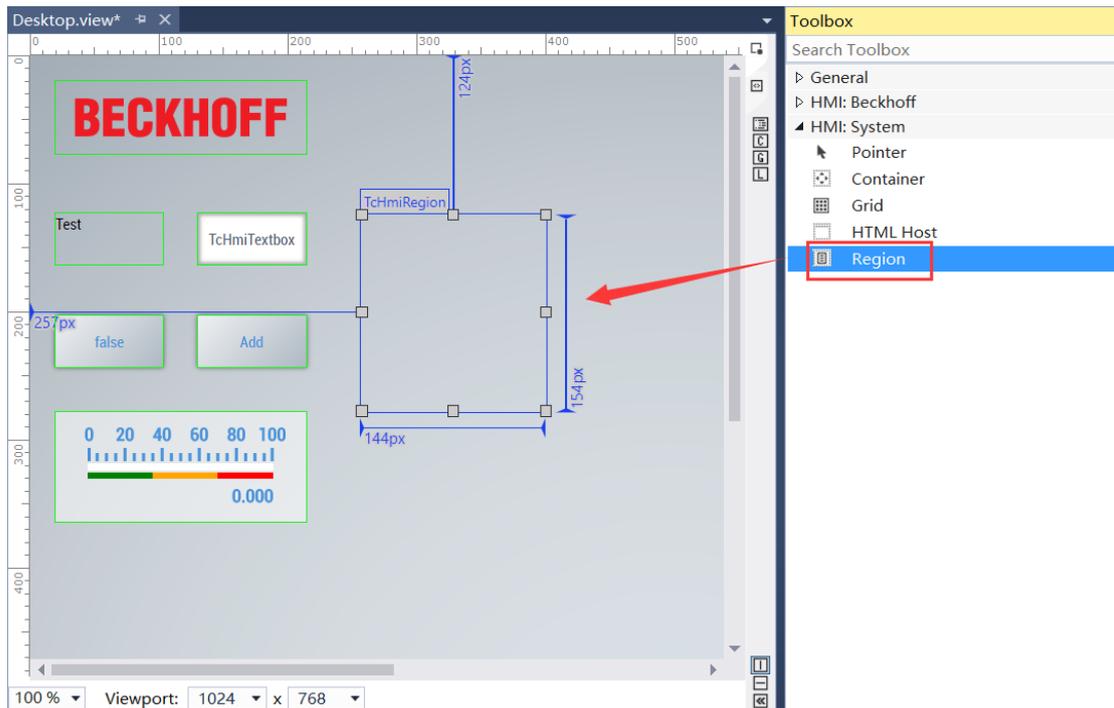
(3) 选择 Content 进行添加。



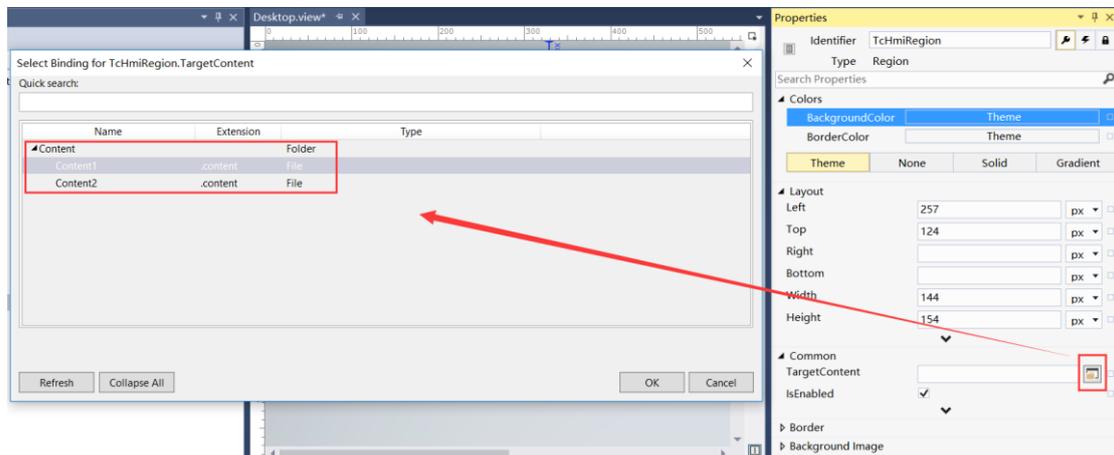
(4) 添加了 2 个 content 后，把这 2 个 content 的背景颜色在属性中进行修改。



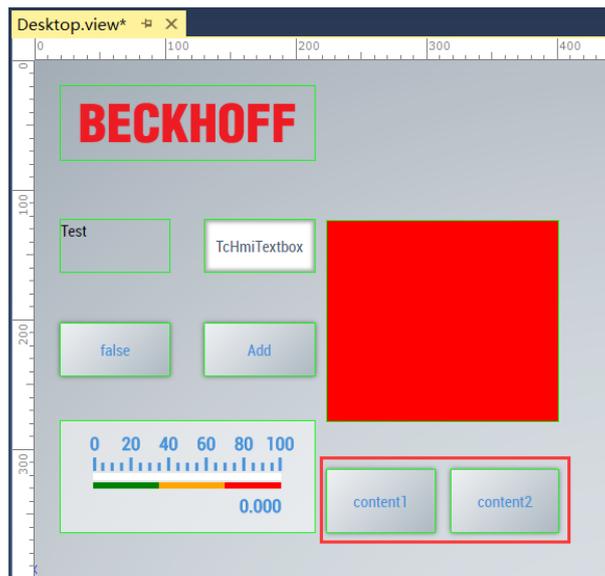
(5) 在主界面中加入 Region，用于切换两个子画面。



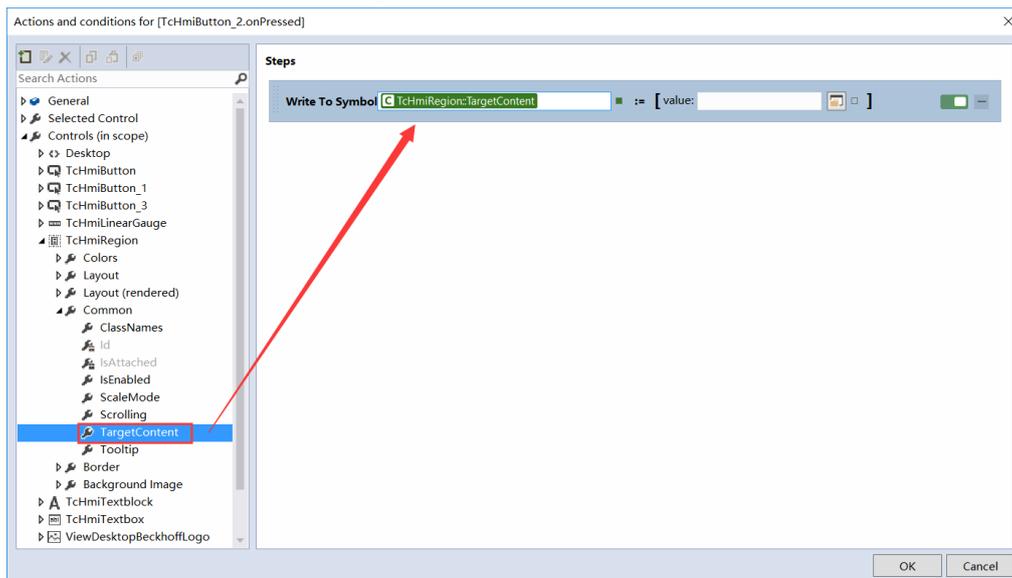
(6) 在属性栏中的 TargetContent 定义一个初始子画面。



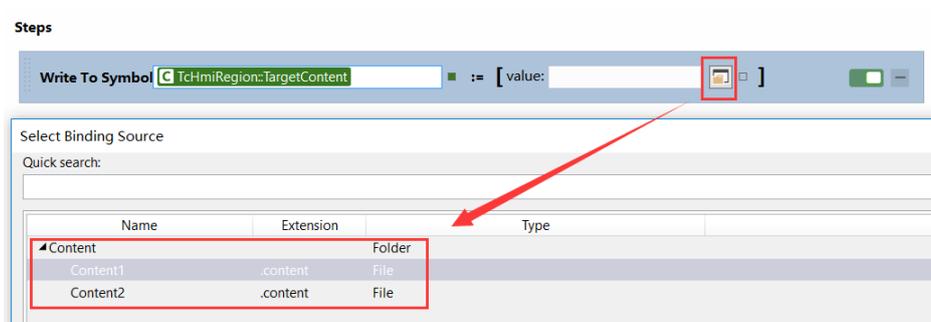
(7) 新增 2 个控件取名为 content1 和 content2，用于切换画面。



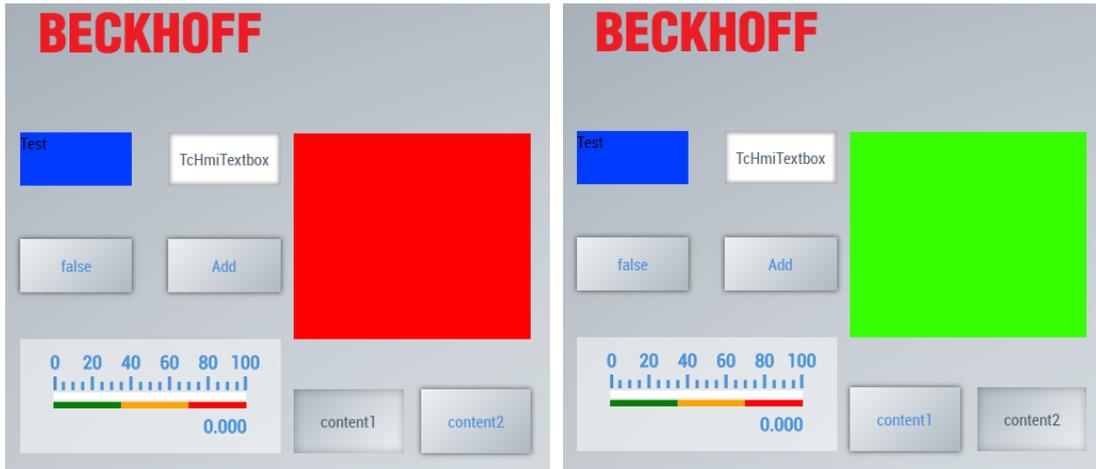
(8) 双击对这 2 个按钮配置.onPressed 事件，
然后找到 Controls→TcHmiRegion→Common→TargetContent，进行添加。



(9) 分别完成对 2 个按钮完成两个画面的配置。

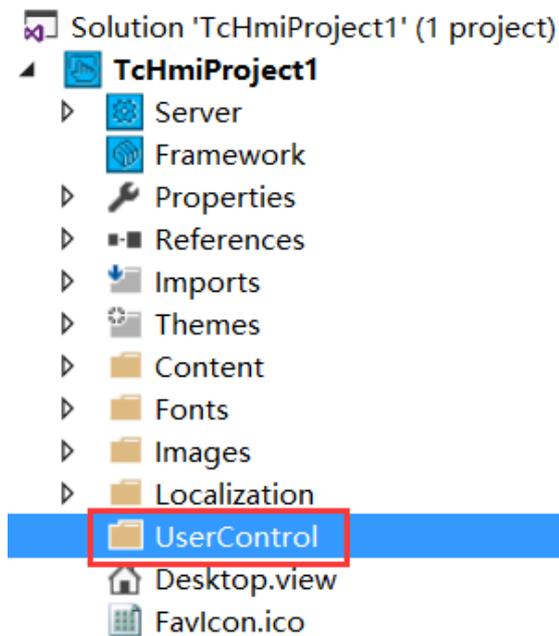


(10) 在线浏览，可以看到分别按 content1 和 content2 可以切换不同颜色的子画面。

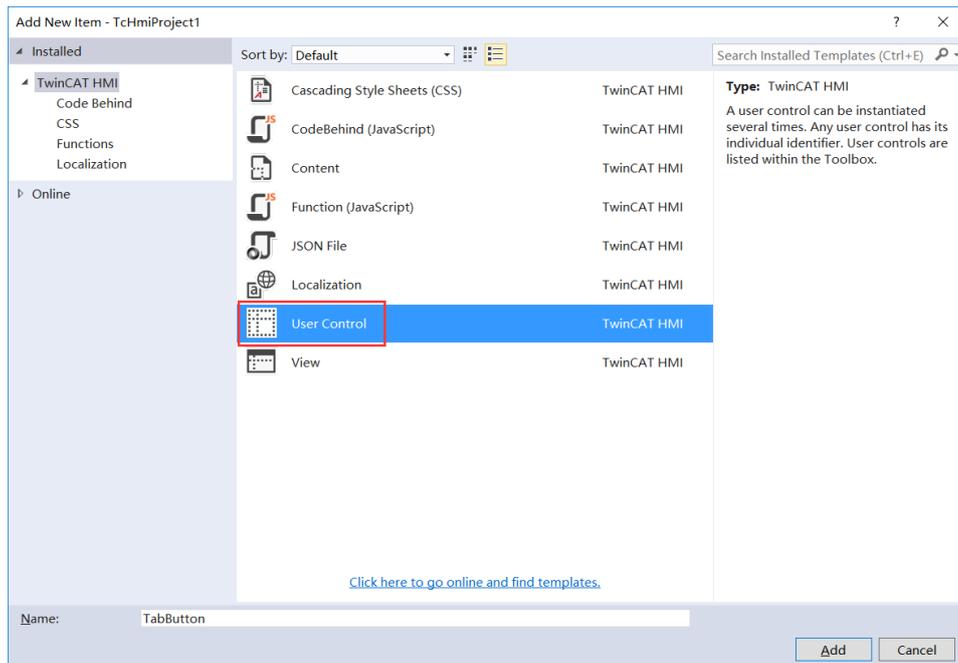


5. 自定义控件

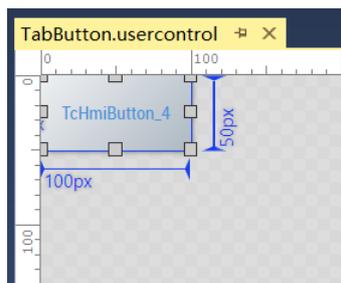
(1) 新建文件夹 UserControl, 实现一些自定义控件的编辑, 这样可以把设置好的控件或者控件组进行反复利用。



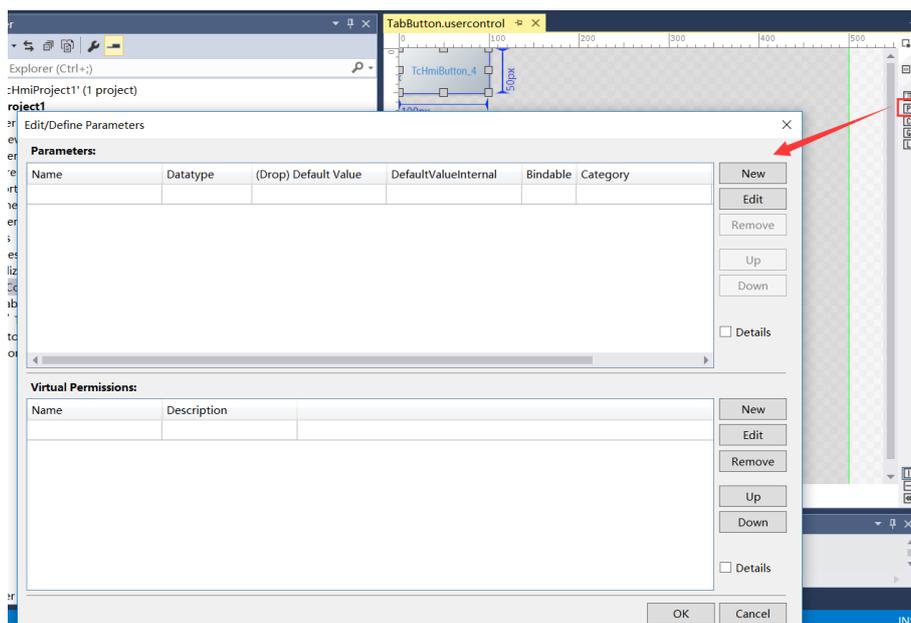
(2) 右键文件夹新建 User Control，取名为 TabButton。



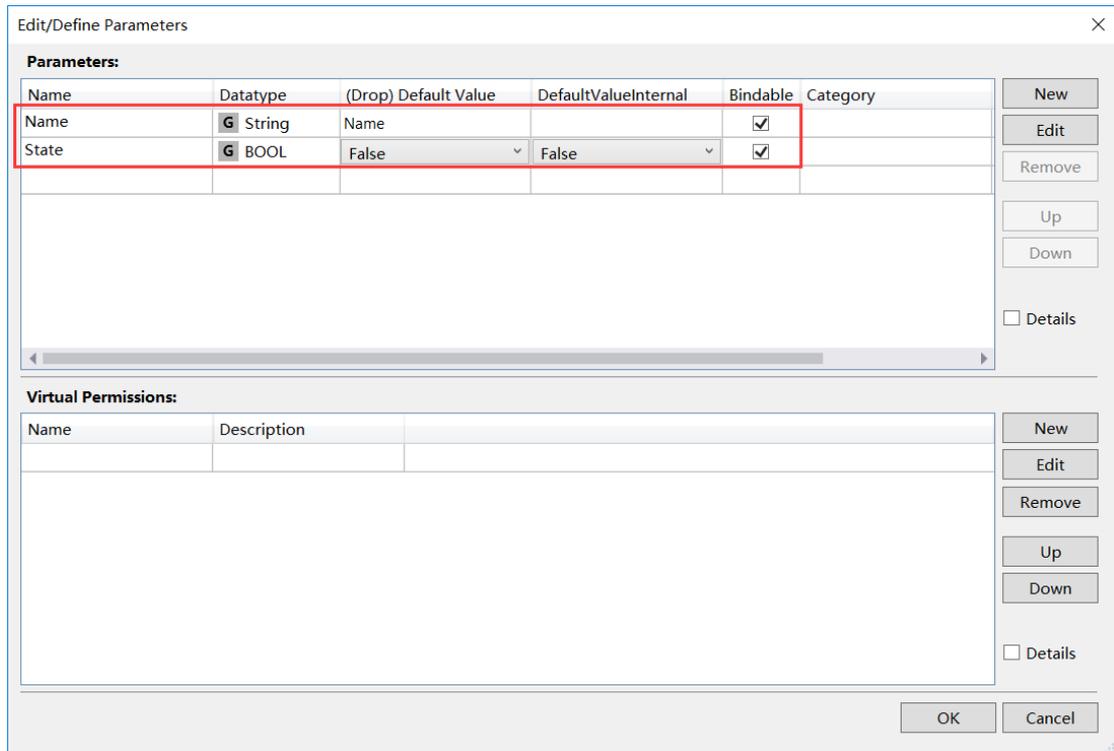
(3) 拖拉一个 Button 到 TabButton 中最左上角。



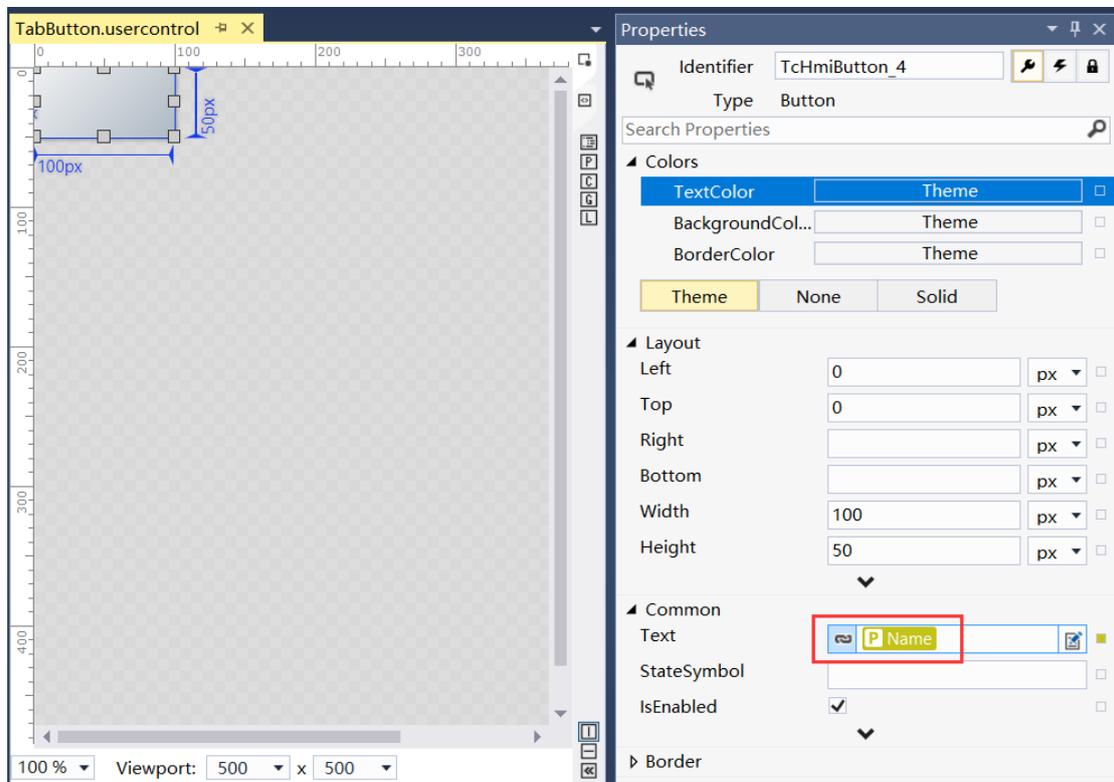
(4) 随后点击此 TabButton 旁边的 **P** 选项，对其进行设定自定义参数。



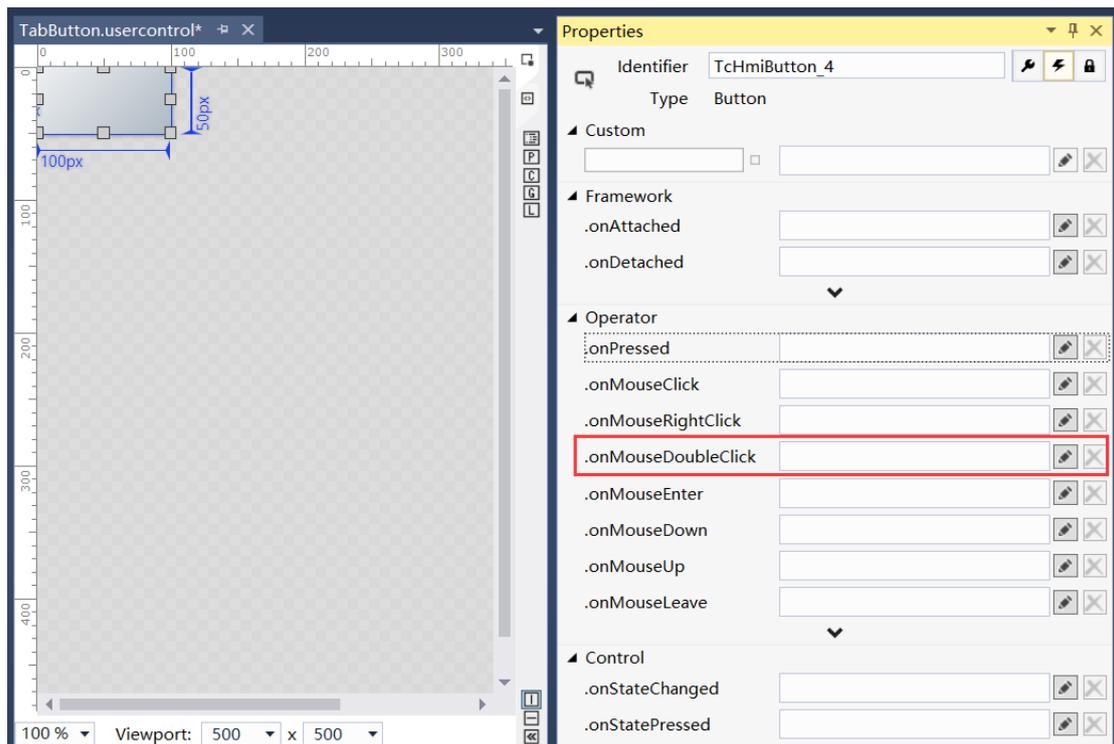
(5) 创建 2 个参数变量 Name、State。



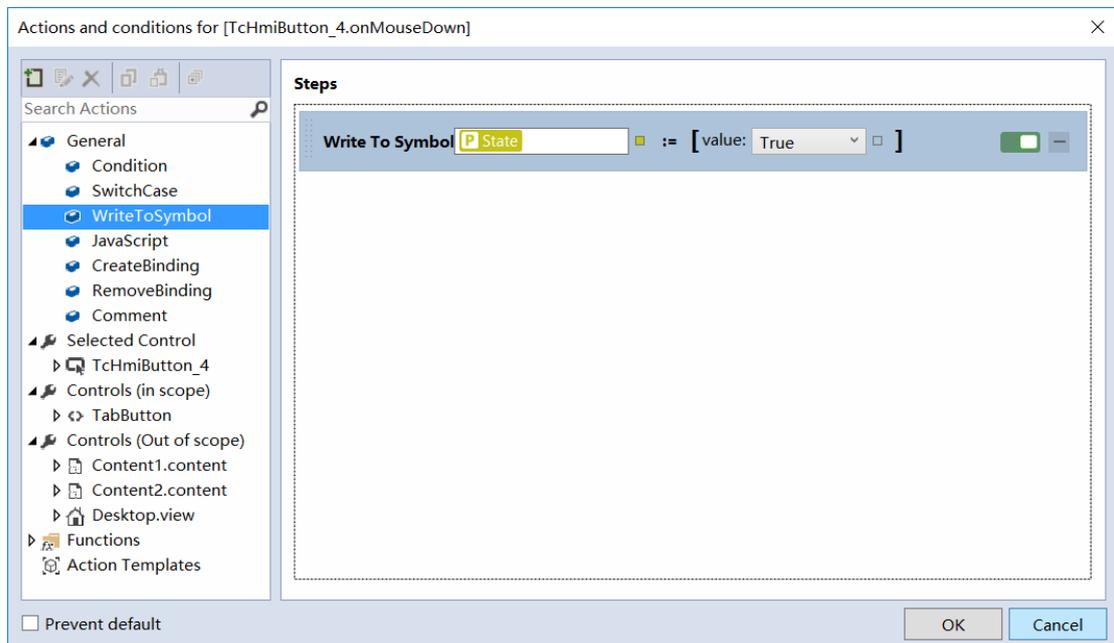
(6) 紧接着对刚才的 button 进行配置，Text 设定为参数 Name。



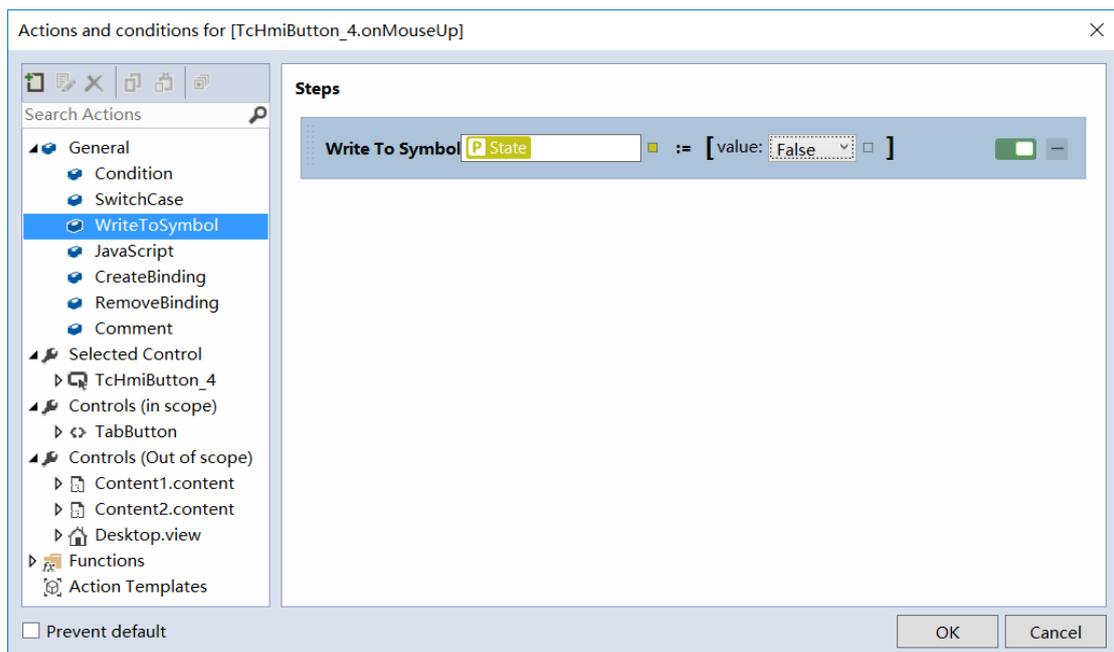
(7) 随后在 Events 中配置事件.onMouseDown。



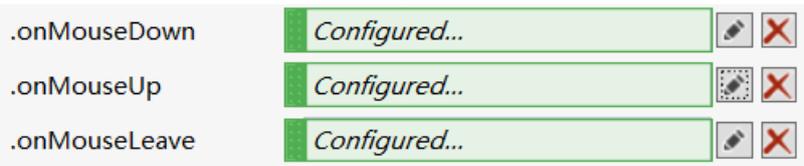
(8) 设置当按下 button, State 为 True 的事件。



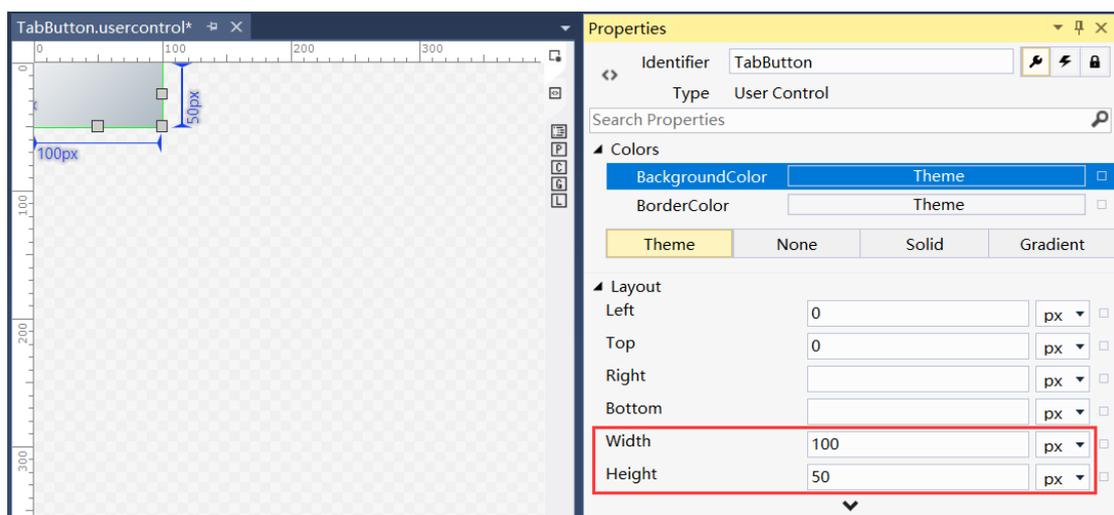
(9) 同样把.onMouseLeave 和.onMouseUp 设置为 State 为 False 的事件。



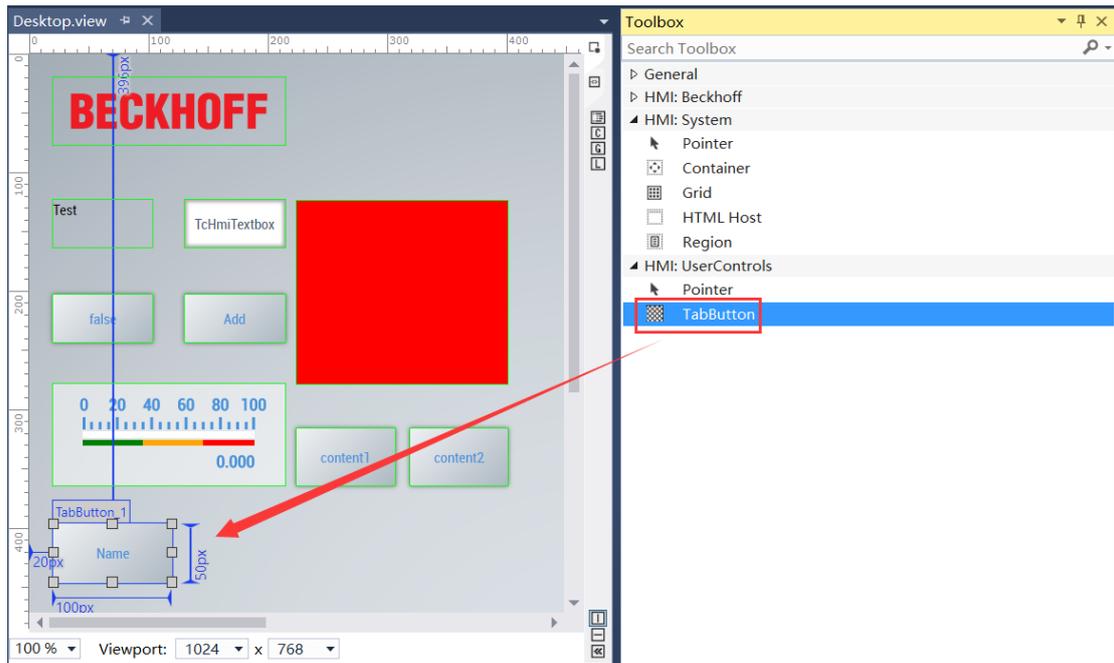
(10) 配置 3 个事件后，会有 3 个绿色 Configured 条出现。



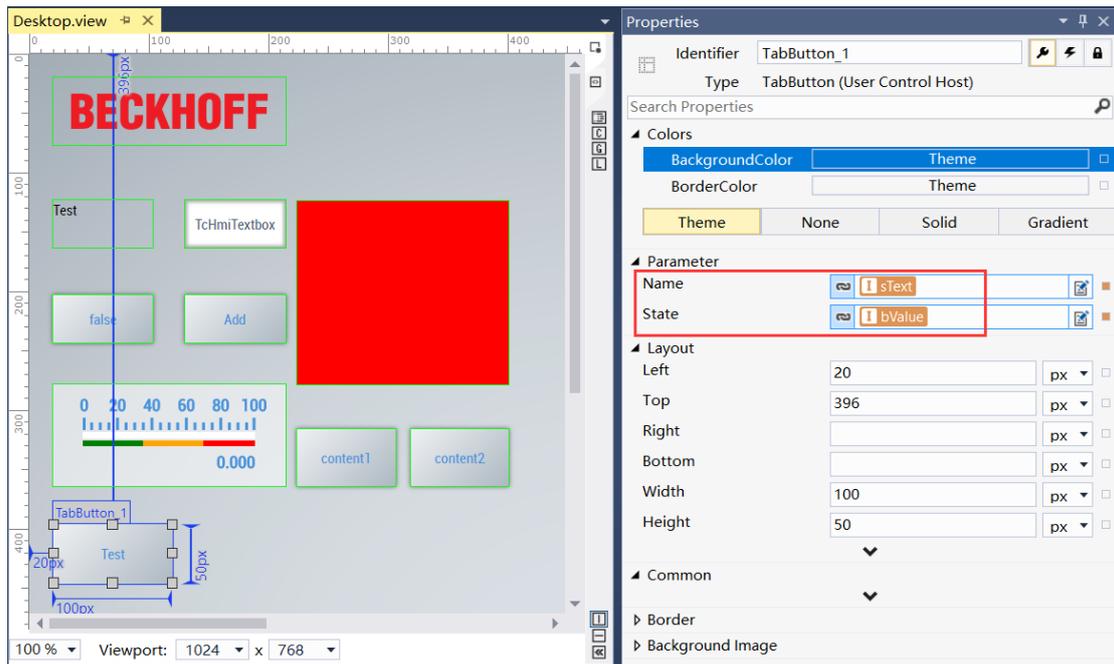
(11) 最后把 TabButton 设置与 button 一样大小，把 Width 与 Height 切换到 px 单位，随后修改对应的值，例如我这里是 Width : 100, Height : 50。



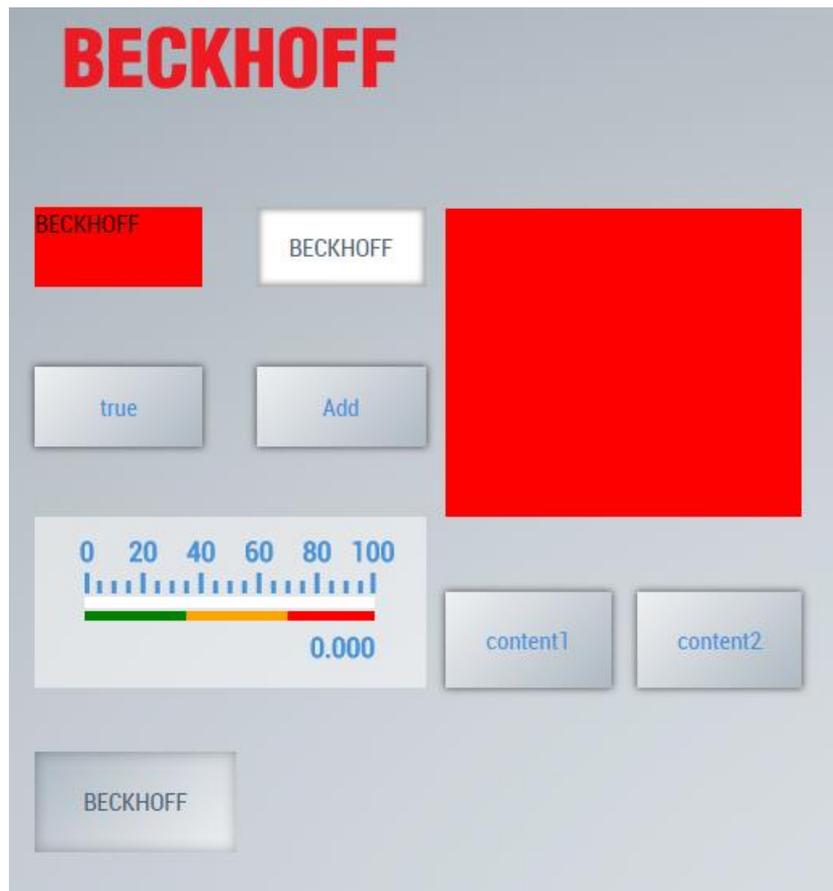
(12) 设置好后我们在主界面中进行调用，选择 TabButton。



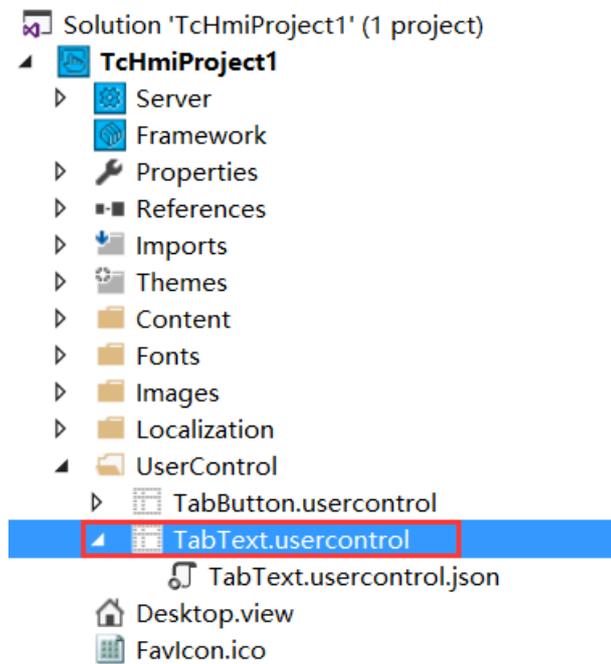
(13) 自定义控件的好处就是，所有配置或者事件都已经事先编辑好，只需要在属性栏中对对应参数进行配置即可，例如这里把 2 个参数设置之前定义好的内部变量 sText、bValue。



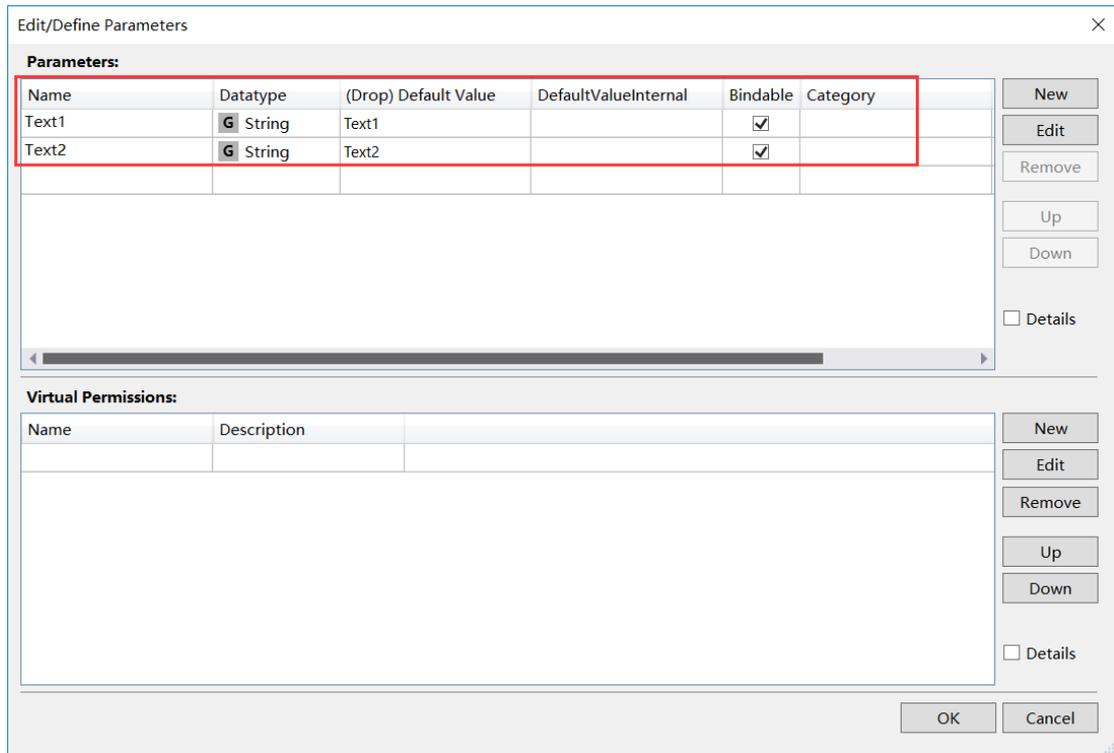
(14) 在线的效果，按着此 TabButton，之前做的相关联的控件也会有变化。



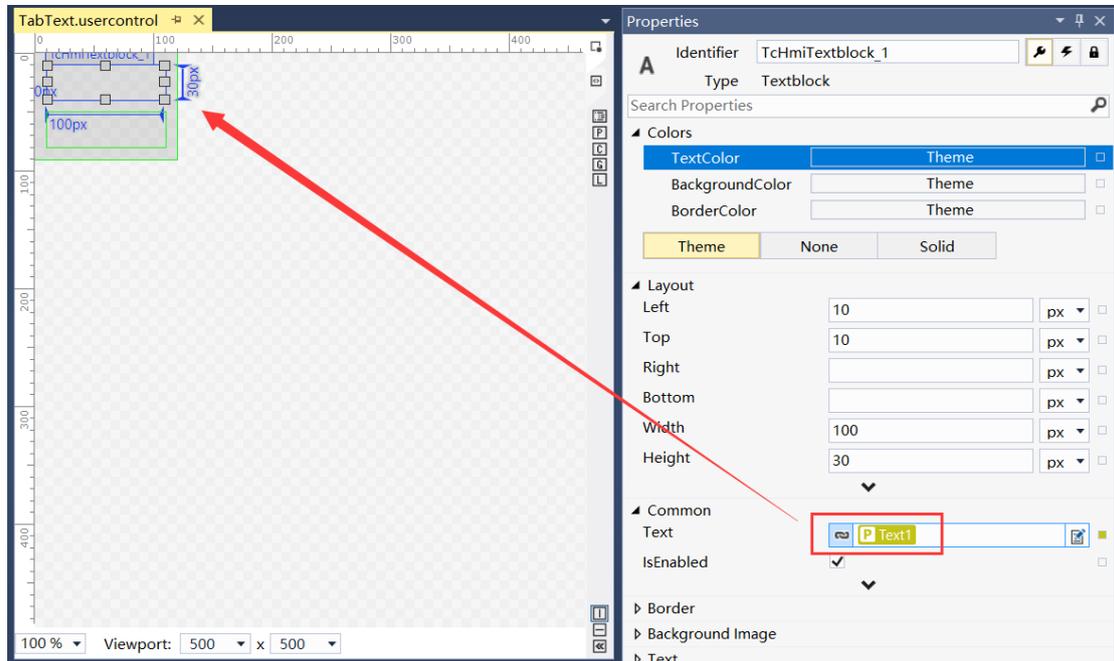
(15) 紧接着继续在 UserControl 中添加一个 TabText。



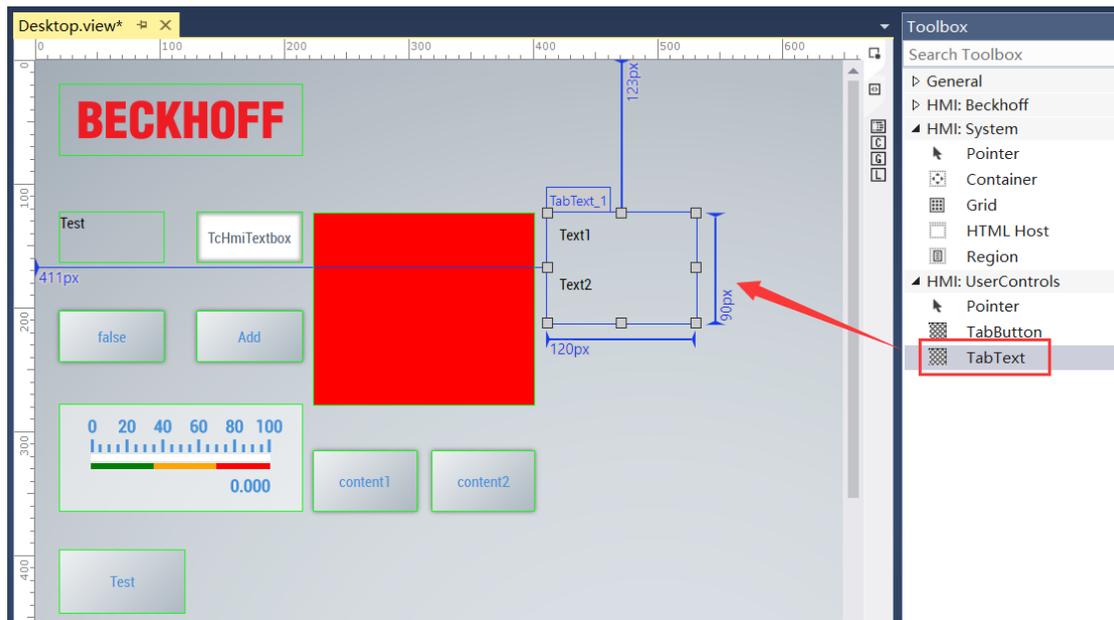
(16) 在 TabText 中加入 2 个参数。



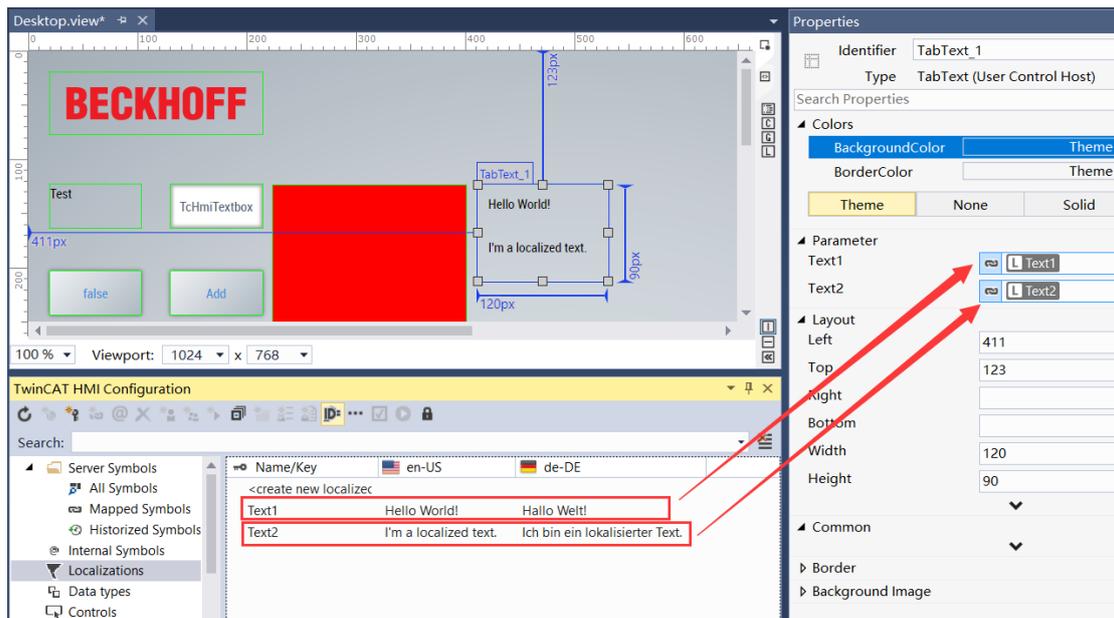
(17) 在 TabText 中添加 2 个 Textblock 控件，把两个参数分配给这 2 个控件中的 Text 中，并且把 TabText 调整到合适的大小。



(18) 在主界面中加入做好的 TabText。

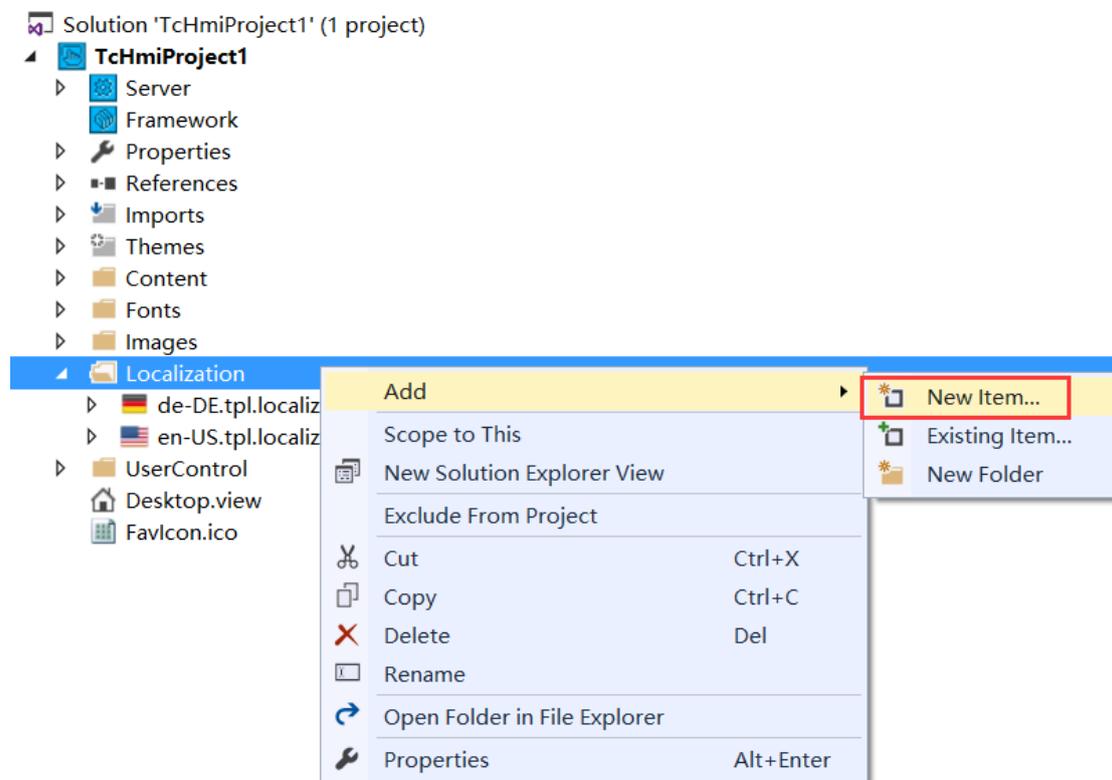


(19) 随后把项目自带的 Text1 和 Text2 分配给 TabText 的 2 个参数中，可以发现效果为预设的 2 个字符“Hello World !”、“I'm a localized text”。

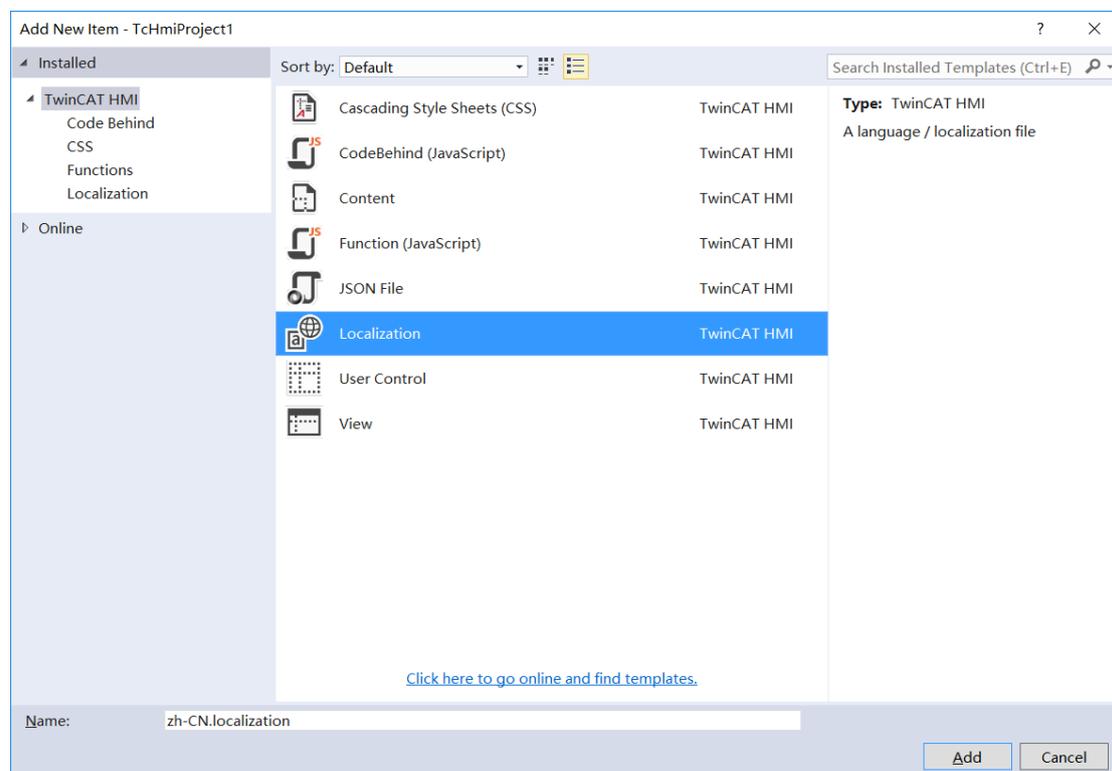


6. 多语言切换

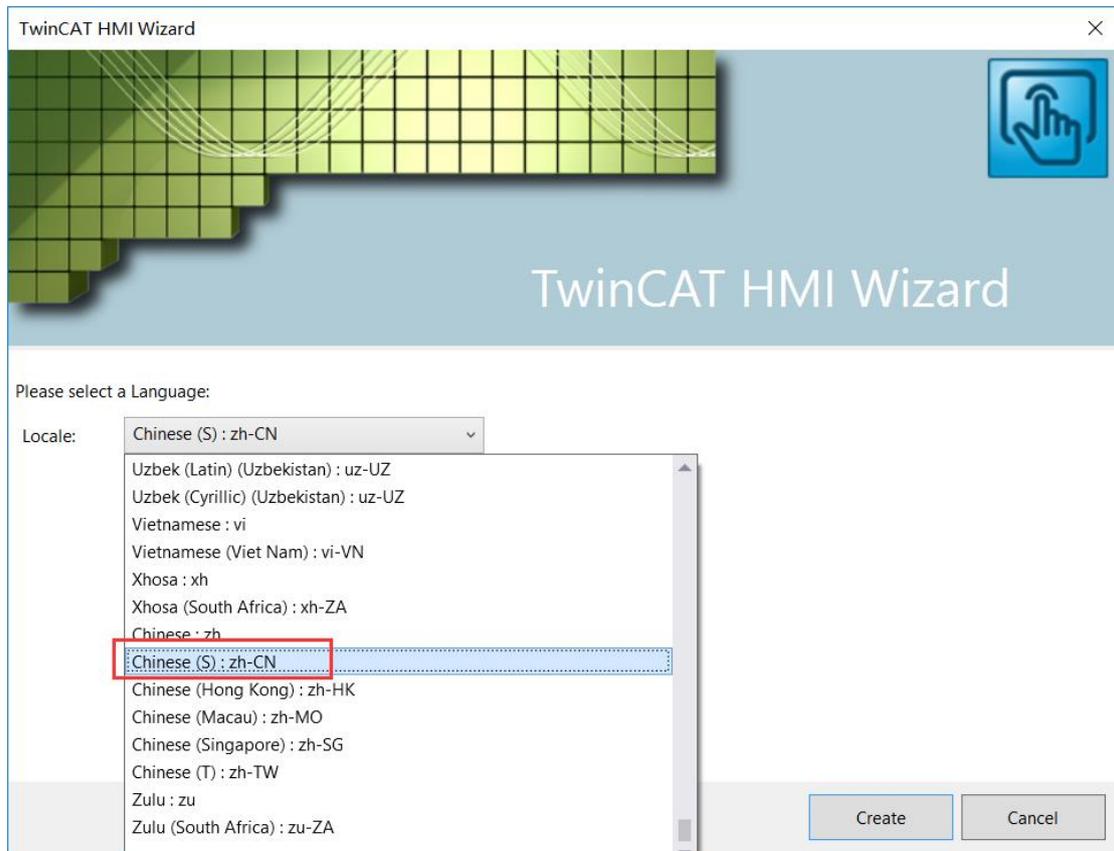
(1) 要实现语言切换，我们需要在文件夹 Localization 中配置对应语言，默认可以发现只有德文和英文两种语言，如果希望有中文加入，可以右键文件夹，然后添加一个新项。



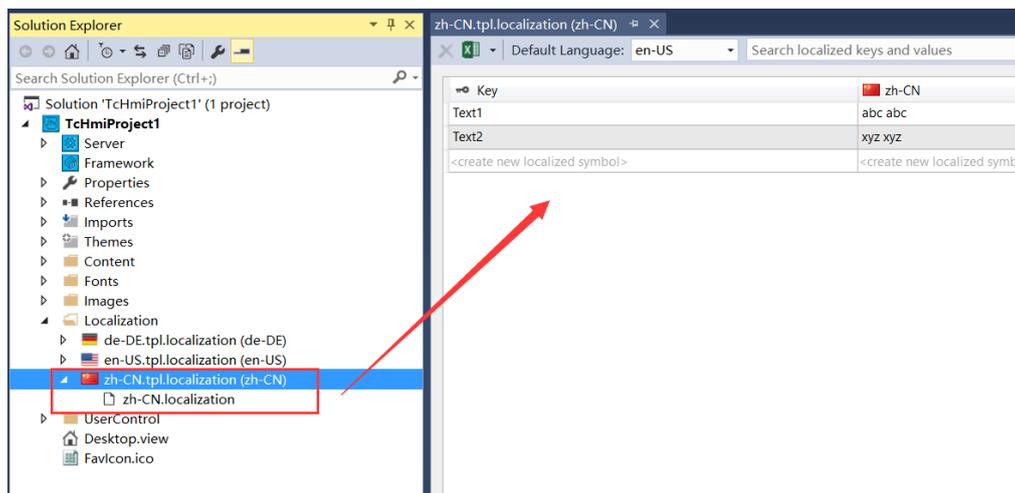
(2) 选择 Localization 完成创建，取名为 zh-CN。



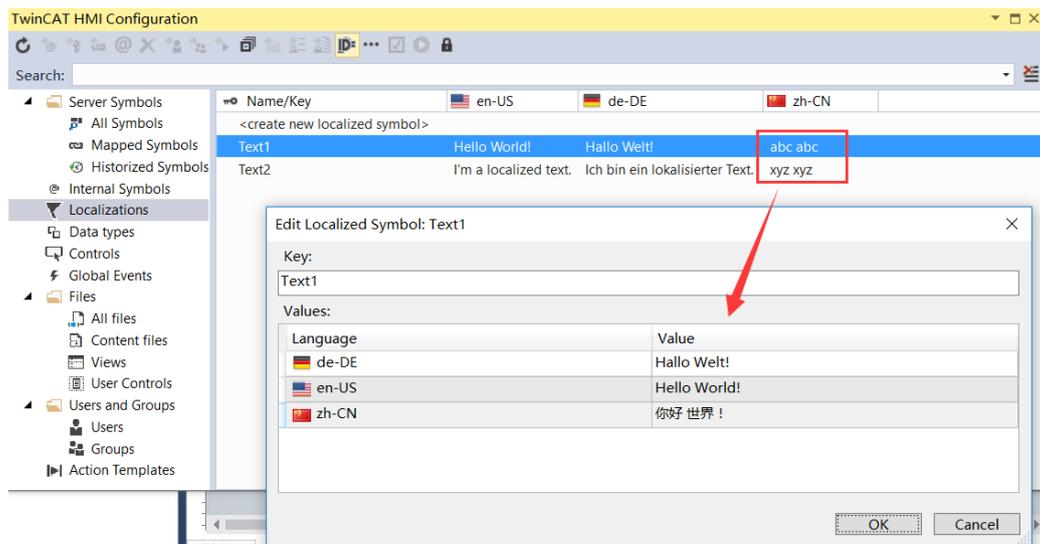
(3) 弹出窗口，在下拉框中找到 Chinese (S) :zh-CN 完成创建。



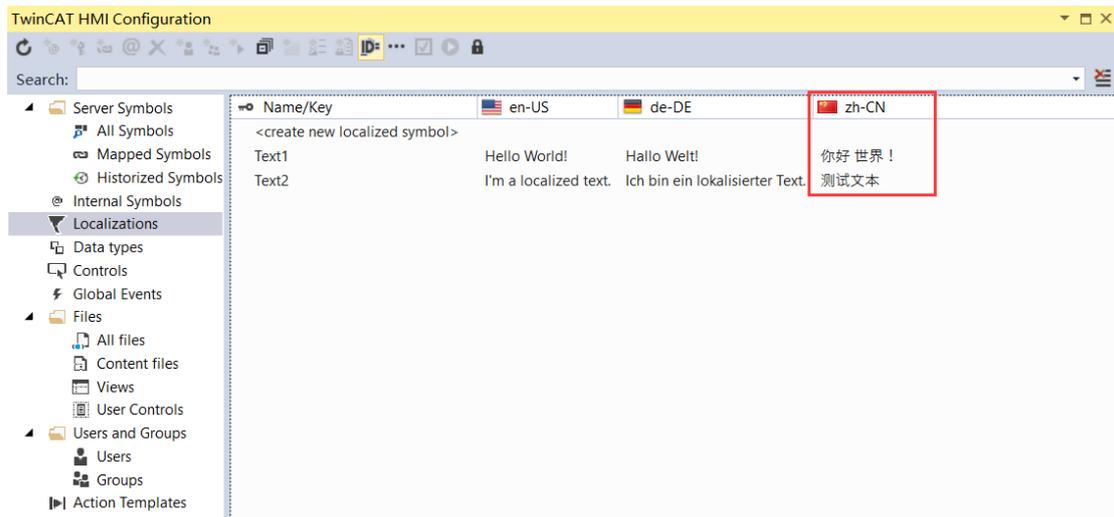
(4) 完成创建后在 Localization 就会出现有中国国旗的中文语言列表。



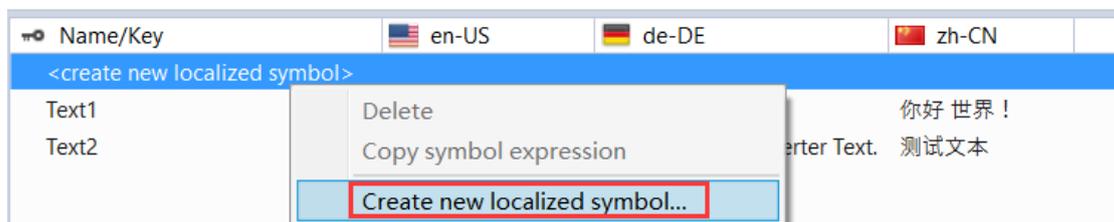
(5) 打开 TwinCAT HMI Configuration，在 Localizations 中手动修改 Text1 与 Text2 的中文对照。



效果如下：



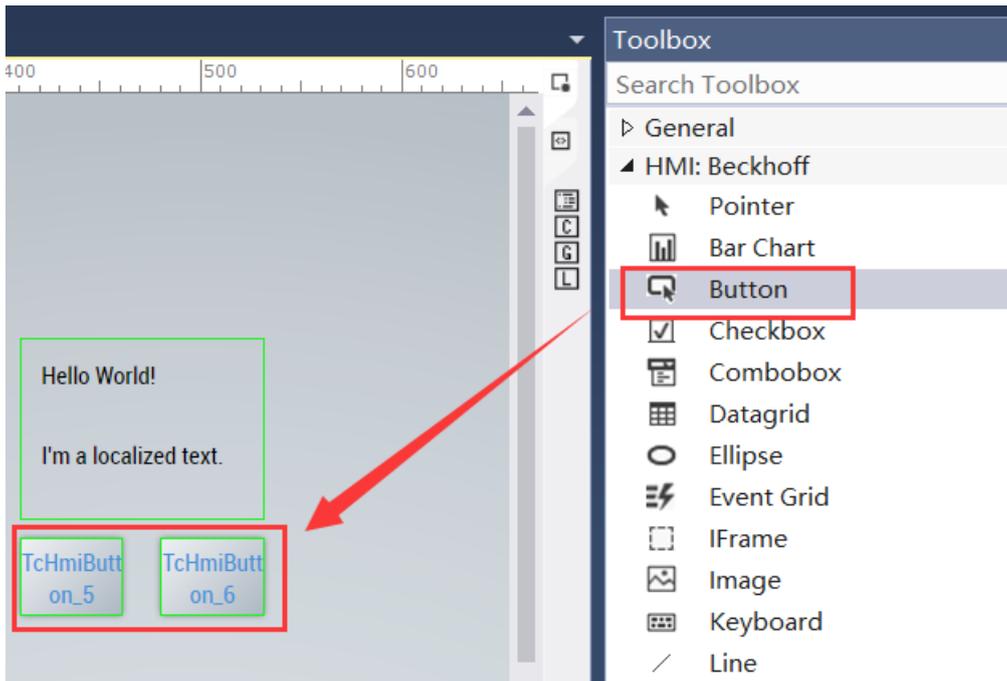
(6) 当然也可以手动再创建你所需要的文本以及各语言对照表，只需要右键选择 Create new localized symbol 即可。



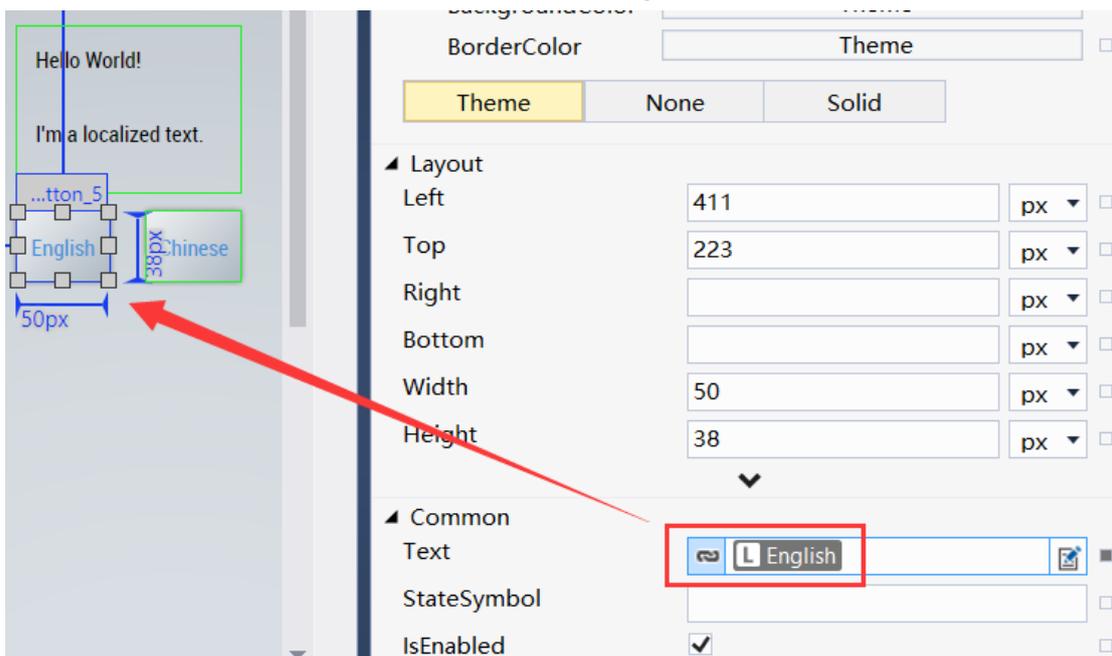
(7) 加入 2 个文本 English 和 German, 并且编辑对应中文翻译。

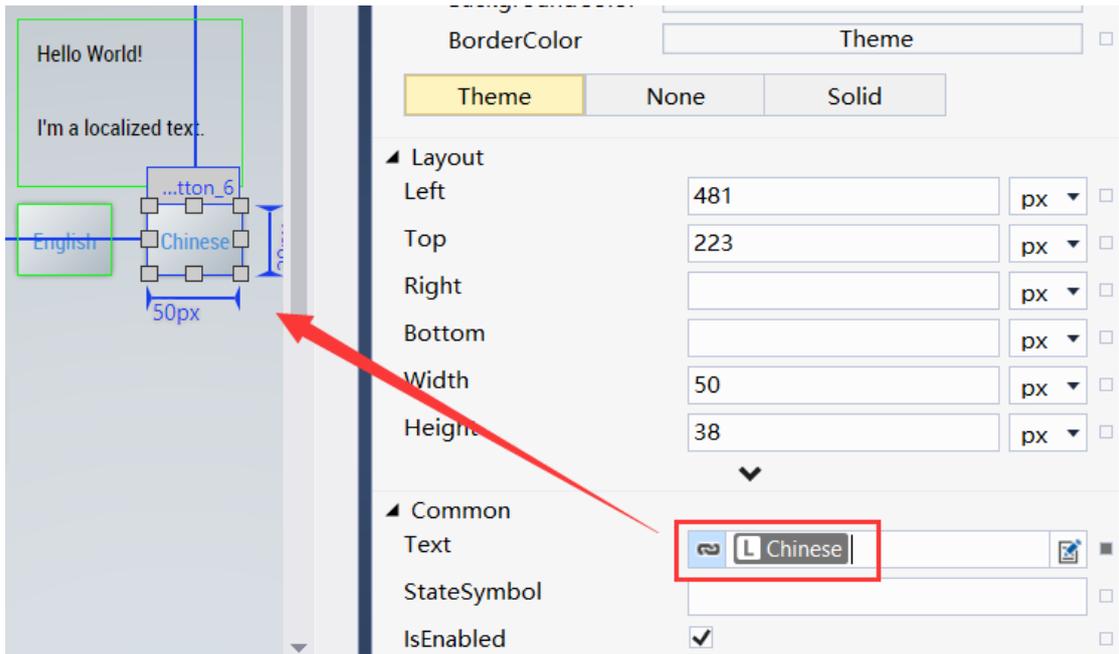
Name/Key	en-US	de-DE	zh-CN
<create new localized symbol>			
Chinese	Chinese		中文
English	English		英文
Text1	Hello World!	Hallo Welt!	你好世界!
Text2	I'm a localized text.	Ich bin ein lokalisierter Text.	测试文本

(8) 在主界面中加入 2 个按钮完成中英文切换。

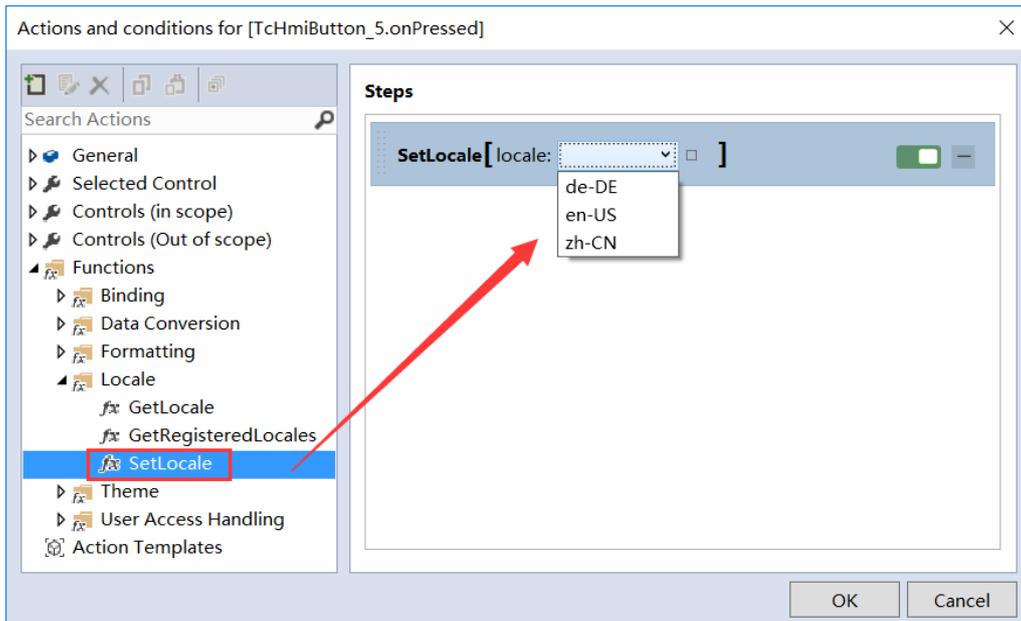


(9) 把 2 个 Button 的 Text 分别设置成本地文本 English 和 Chinese。

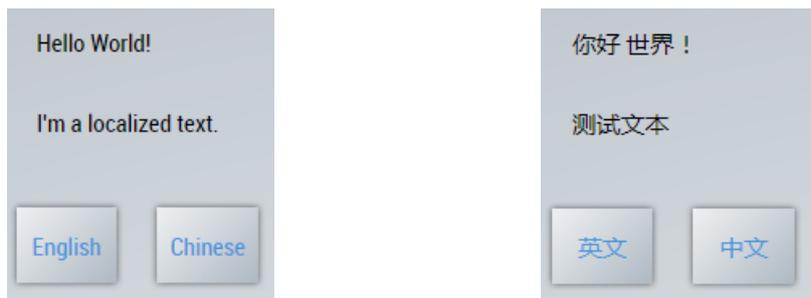




(10) 分别双击 2 个 Button，设置切换语言的事件 Functions→Locale→SetLocale，分别选择 en-US 和 zh-CN。



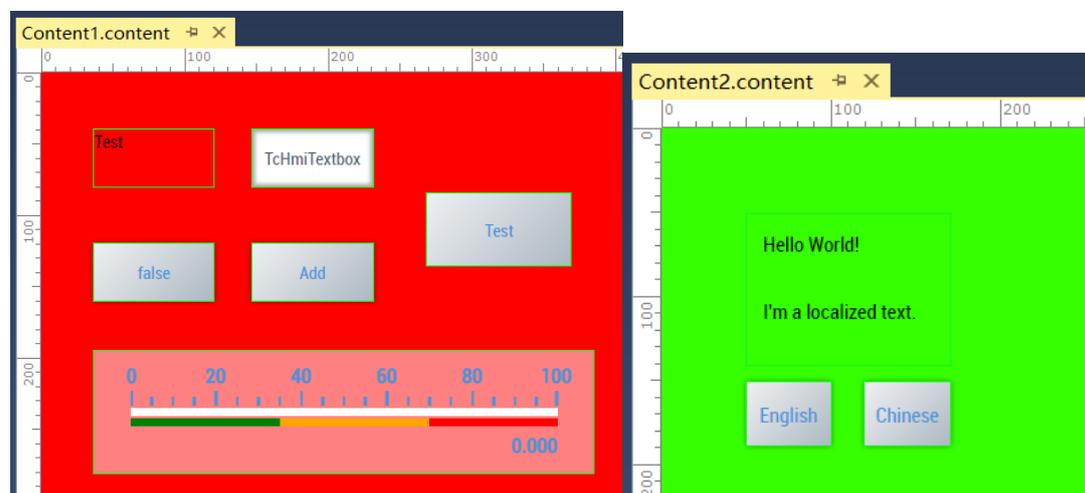
(11) 在线浏览，分别点击 English 和 Chinese 可以切换不同语言。



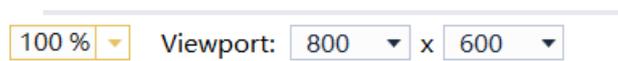
7. 主页面排版规划

(1) 其实 Desktop 作为一个主页面，一开始就应该对其进行排版，放在这里才讲因为我们之前的一些操作为了熟悉软件的基本使用。

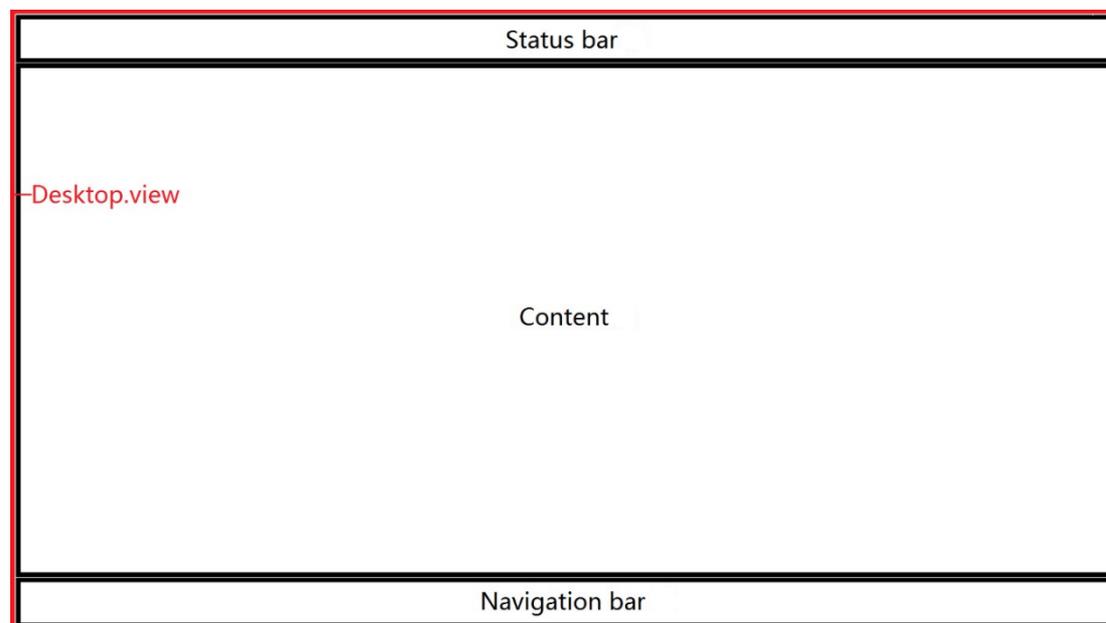
我们先把除了 BECKHOFF logo、content1 和 content2 之外的其他控件都分别放在两个子页面中。



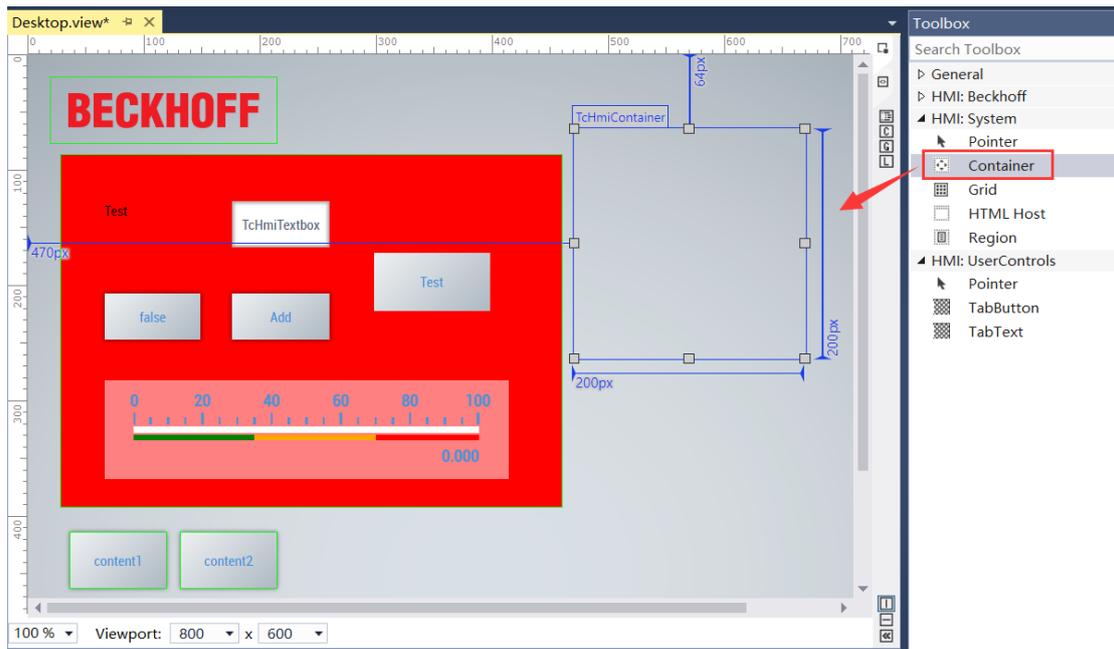
(2) 然后回到 Desktop，把分辨率设定为 800:600，或者你觉得合适的分辨率。



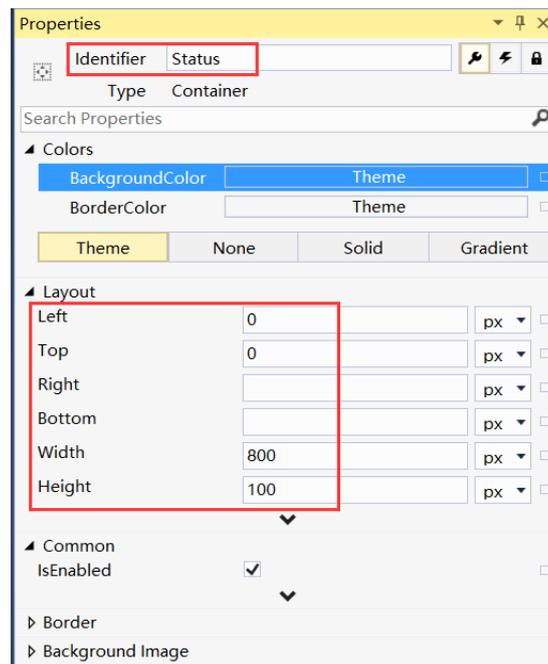
(3) 主页面的分割可以设计成最上面的状态栏 (status bar)，中间的内容栏 (content)，以及最下面的导航栏 (navigation bar)。



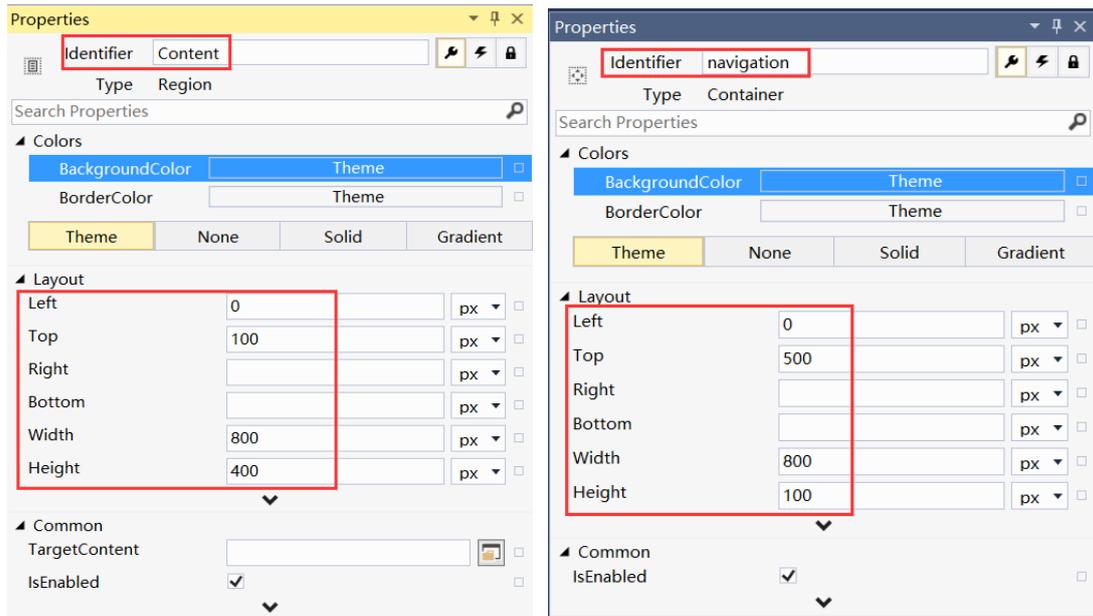
(4) 从工具栏拖拉 Container 进来。



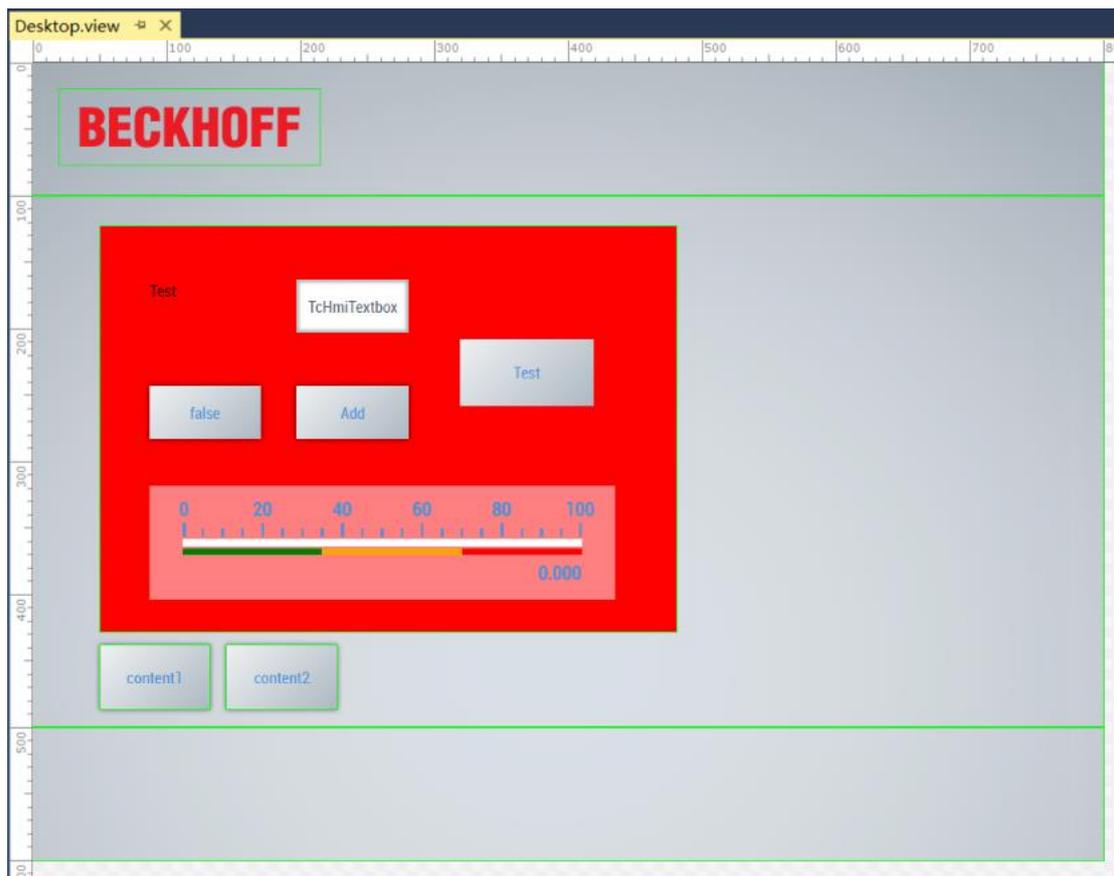
(5) 然后对 container 进行属性编辑，取名为 Status，位置从最左上角 (0,0) 开始，长和宽为 800*100。



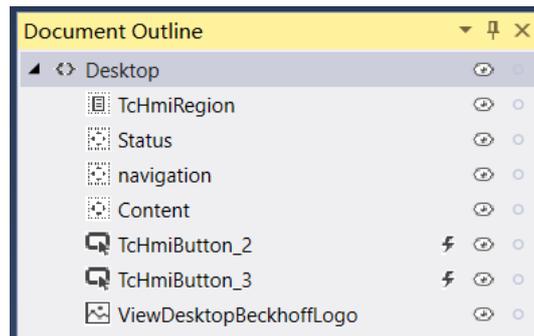
(6) 再拖 2 个 Containter 控件，设置成 content 与 navigation 所对应的参数。



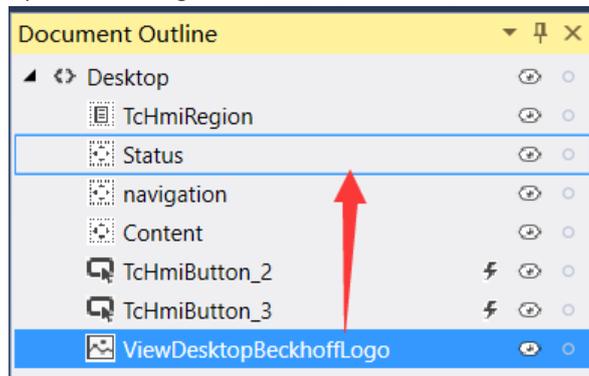
(7) 设置好后，主页面就被分割成了 3 块内容。



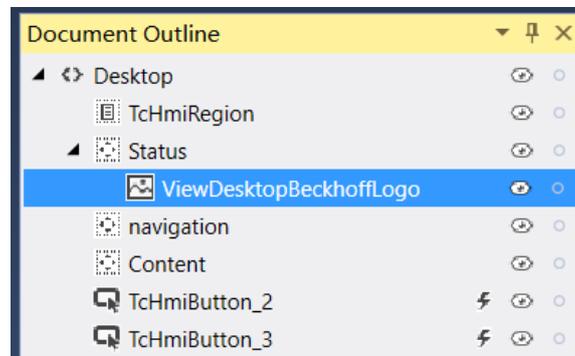
(8) 随后我们需要把其中的几个控件设置为这 3 个部分中的内容, 因此需要点开 Document Outline。



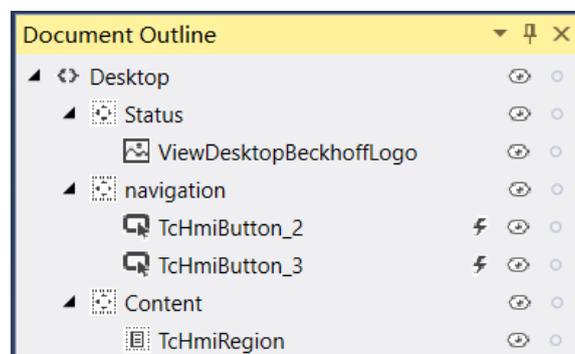
(9) 选中 ViewDesktopBeckhoffLogo, 直接拖到 Status 中。



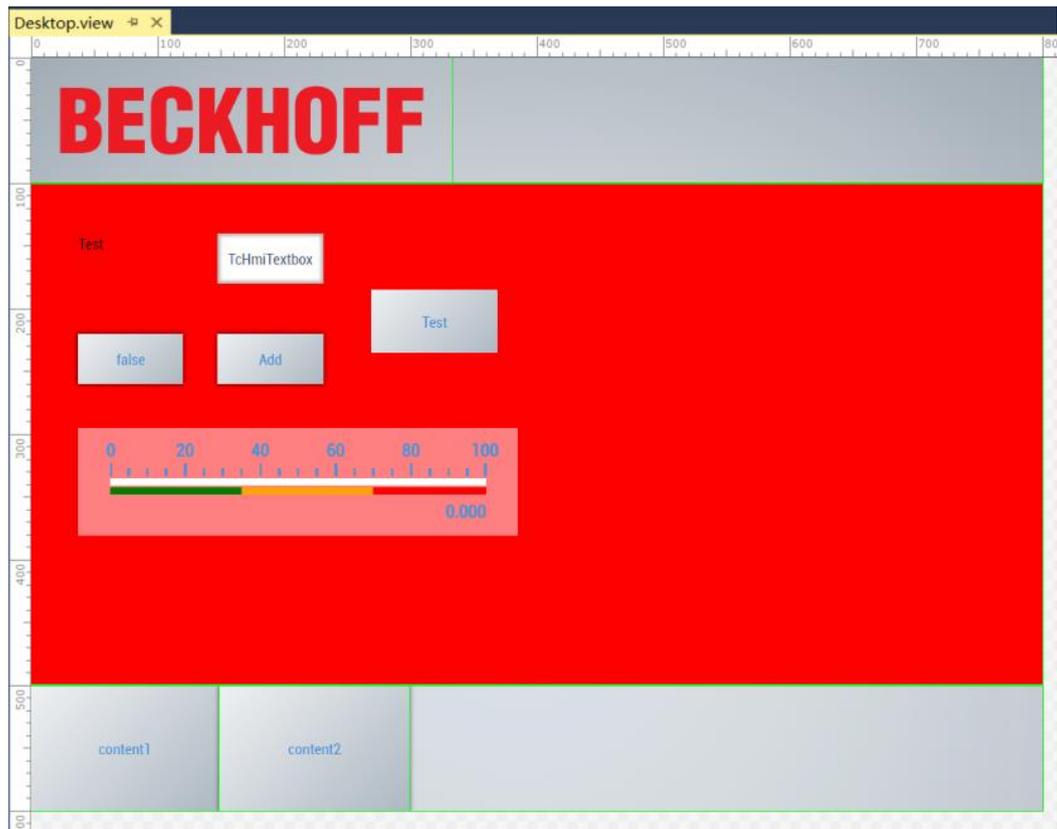
效果如下：



(10) 然后以此把 2 个 Button 放在 navigation 中, TcHmiRegion 放在 Content 中。

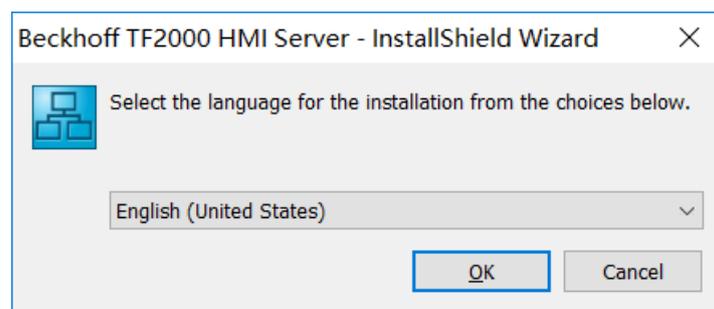


(11) 回到主页面，把这些控件大小做一些调整，在线浏览可以看到整体效果。

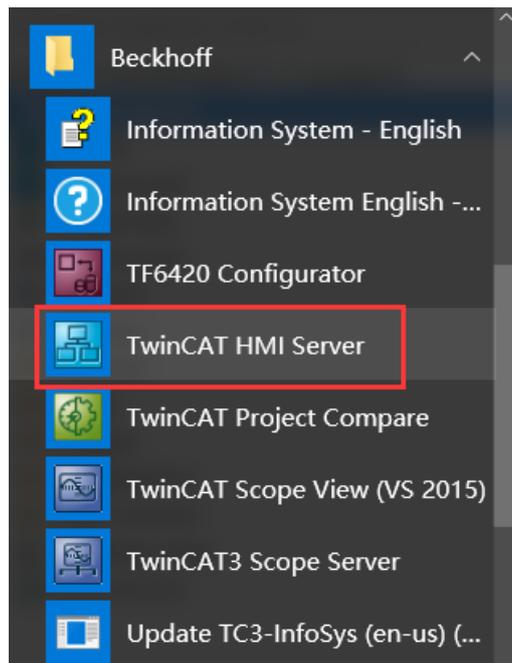


8. 界面发布

(1) HMI 发布需要依托 server，因此需要安装 TF2000-HMI-Server 安装包，按照提示一步步安装。



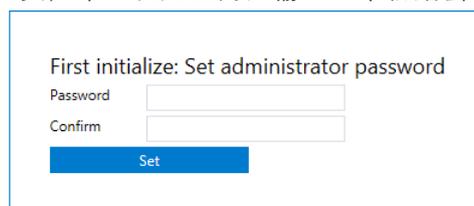
(2) 完成 server 安装后，手动在开始菜单中打开服务。



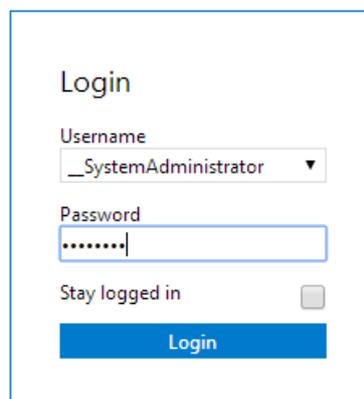
(3) 可以发现在右下角出现一个可发布的 server 信息，点击 Config 可以对这个 server 进行配置。



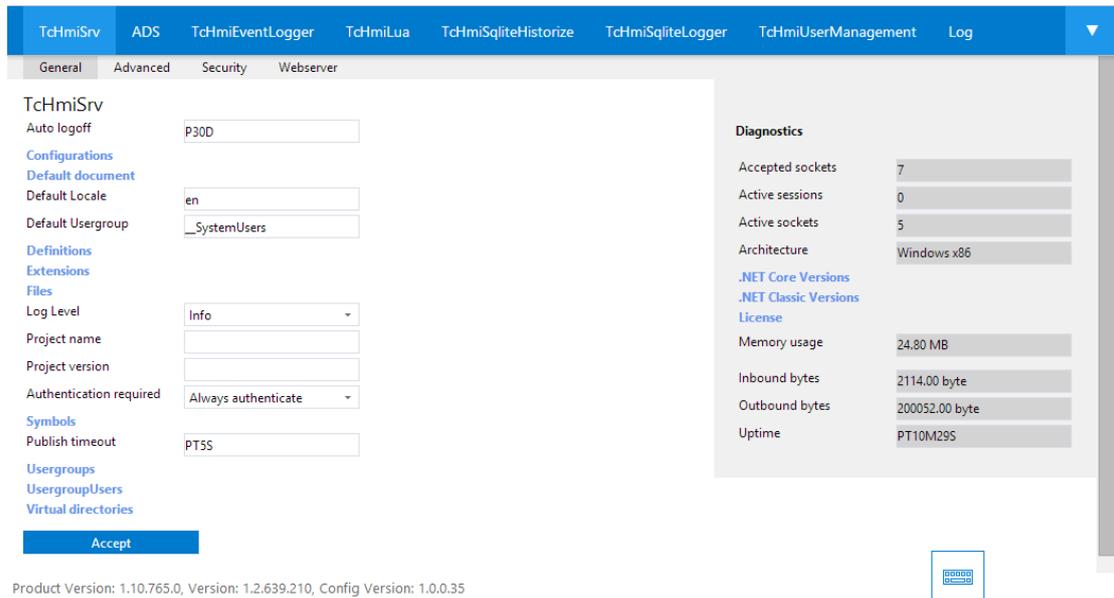
(4) 点开 Config，弹出网页，第一次配置需要输入一个初始密码。



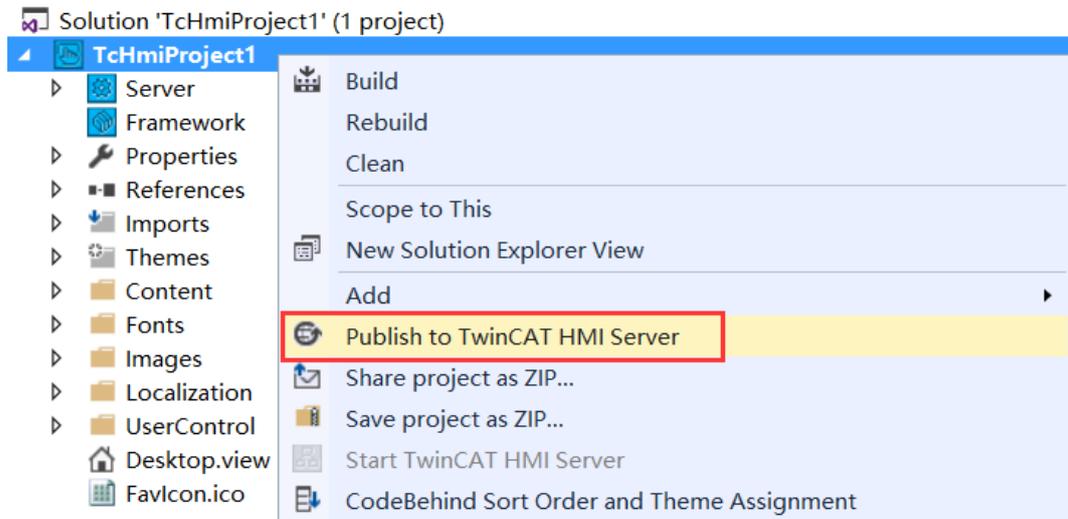
(5) 初始密码完成设定后，以后每一次都需要输入此密码才可以进行配置。



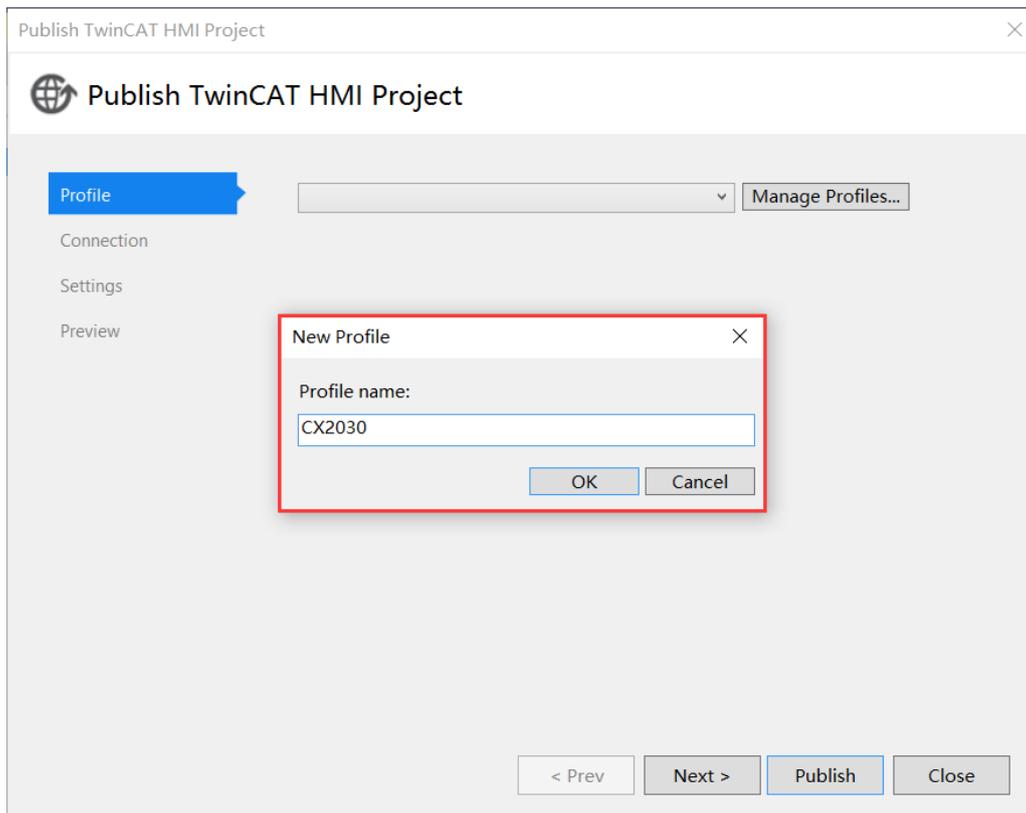
(6) 在 server 配置页面，我们可以对 server 常规信息进行配置，比如 ADS、EventLogger、历史曲线、用户管理以及各种 server 所采集到的异常日志等，今后会详细对其进行说明。



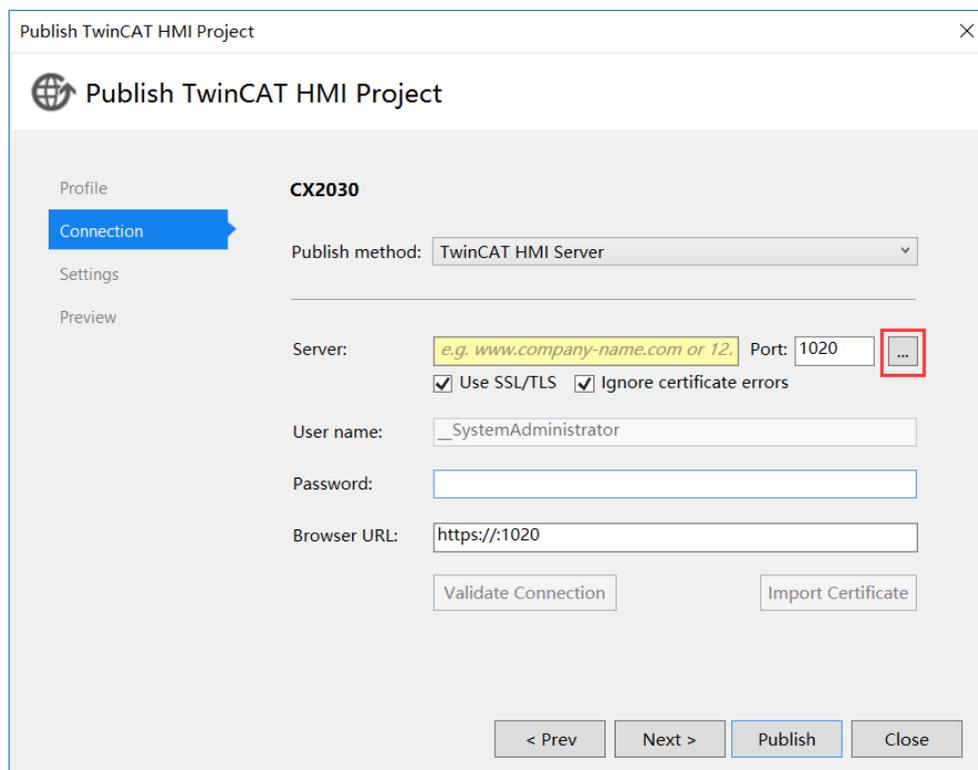
(7) 回到项目中，我们尝试在本地发布 HMI，右键项目选择 Publish to TwinCAT HMI Server。



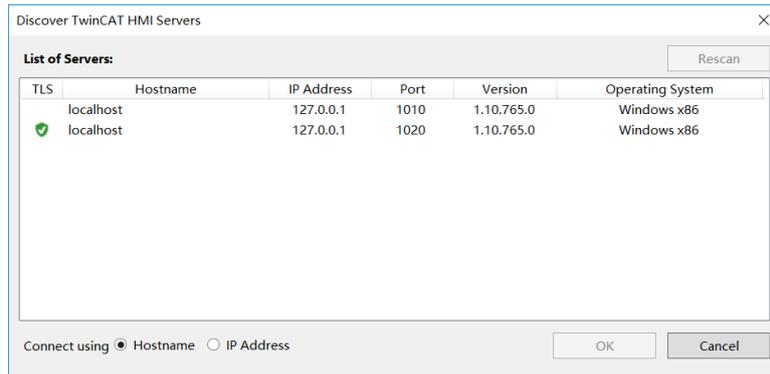
(8) 第一次发布会让你创建一个发布配置信息，你可以取名为某一个 server 的名称。



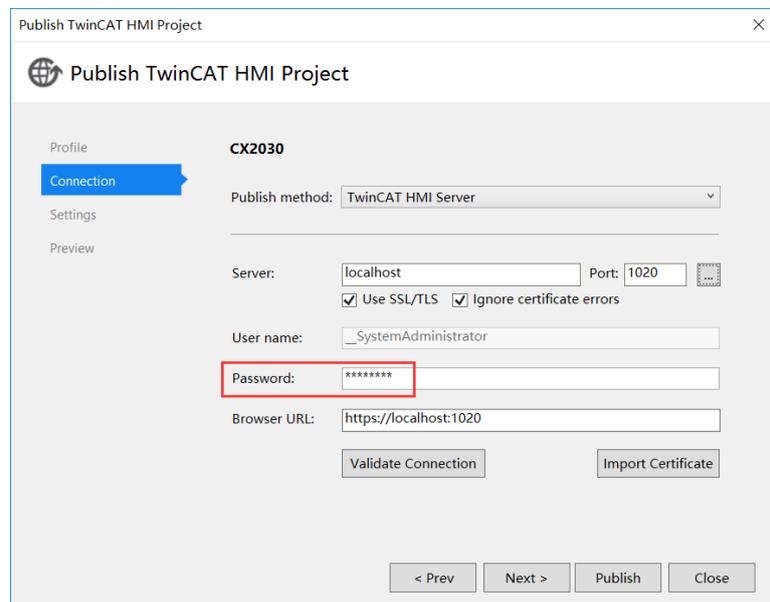
(9) 下一步配置所需要发布的 server 参数，点击  可以广播到所有可用的 server。



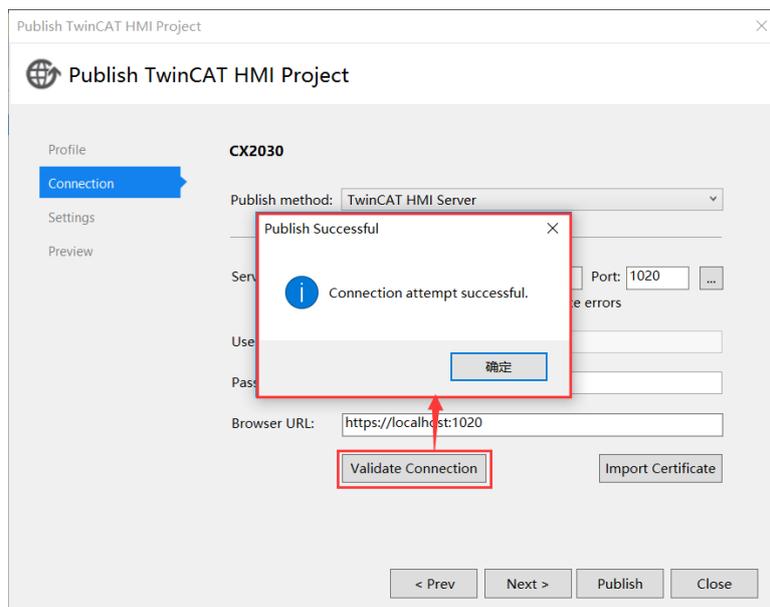
(10) 选择其中一个点击 OK。



(11) 随后把之前设置的初始密码输入到 Password 中。



(12) 全部设置好后，可以点击 Validate Connection 测试下服务器连接是否成功。



(13) 点击 Publish 即可完成发布。

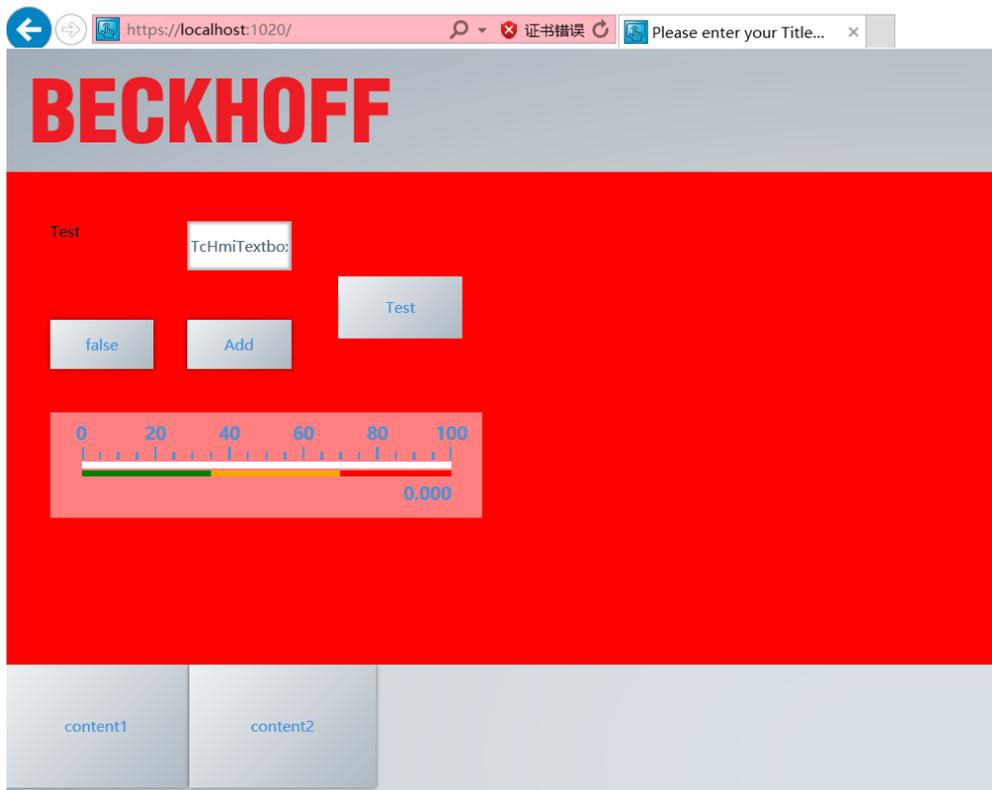


(14) 打开 IE 浏览器输入 <https://localhost:1020>, 提示说没有授权, 我们需要打开 TwinCAT3, 激活 7 天试用版授权 TF2000。

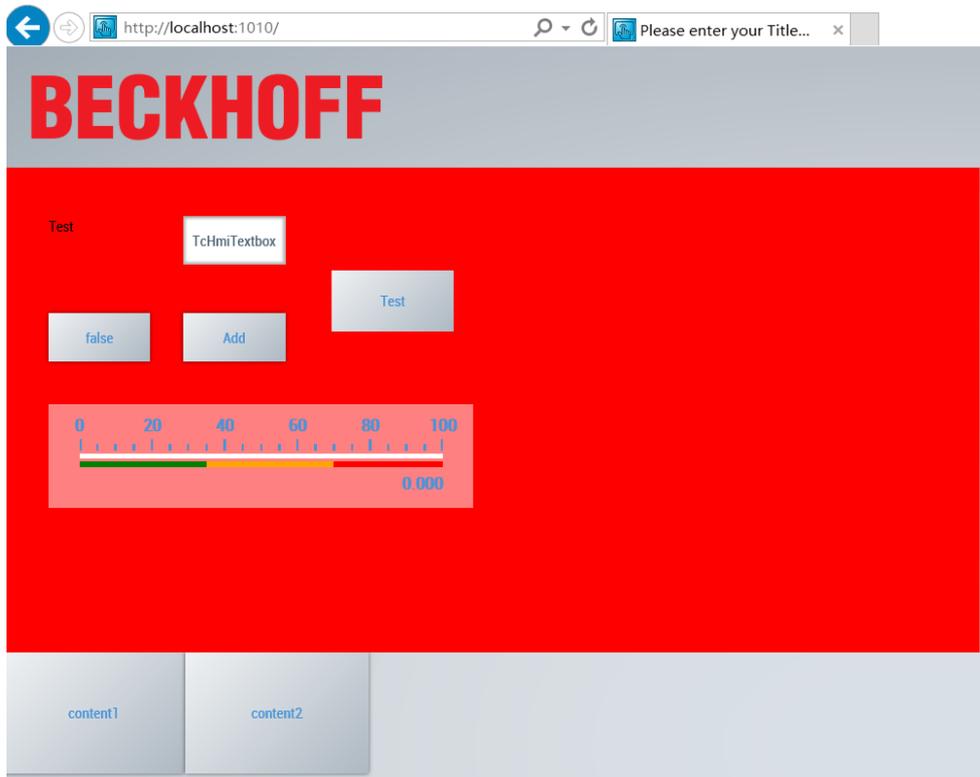
Error: TwinCAT HMI Server license is missing.



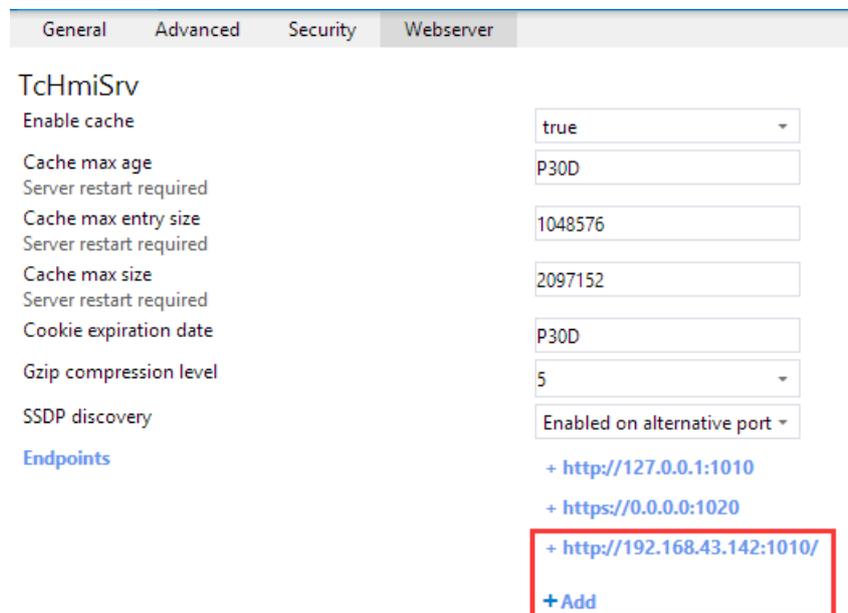
(15) 授权完成后, 重新在 IE 中输入 <https://localhost:1020>, 正常显示主页面, 也可以正常操作, 但会有证书错误提示, 因为我们选择的是 HTTPS 方式进行发布, 没有配置 SSL/TSL 证书。



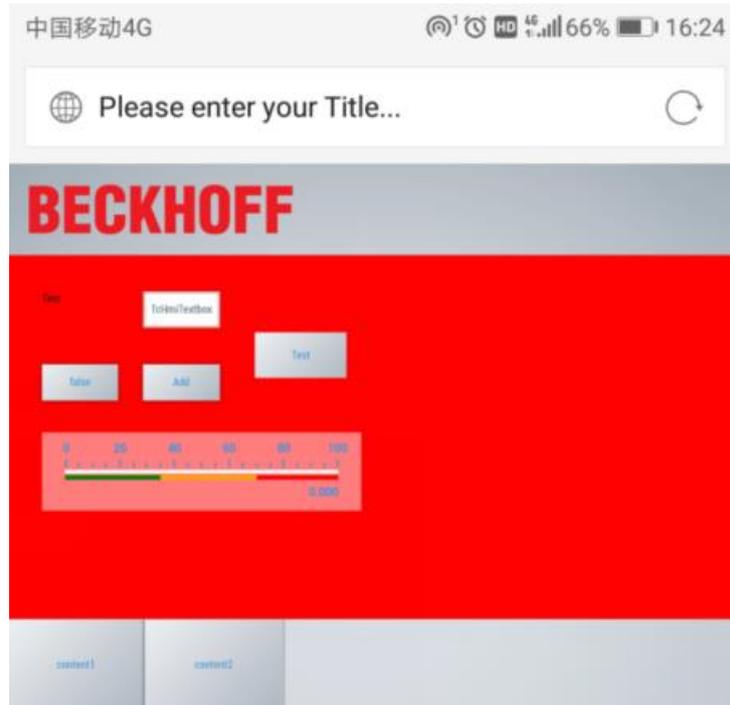
当然也可以以 HTTP 方式进行发布, 就无需证书验证, 如下:



(16) 如果需要通过无线的方式，例如手机连接 HMI，需要在 server config 中加入无线的 IP 到 Endpoints 中。



(17) 这样就可以通过无线方式浏览 TwinCAT HMI，也可以多客户端浏览 TwinCAT HMI。



9. 历史趋势图连接 PLC

TwinCAT HMI 通过 TcHmiSqliteHistorize 扩展了历史数据记录功能，将数据保存在 SQLite 数据库的缓冲区中。

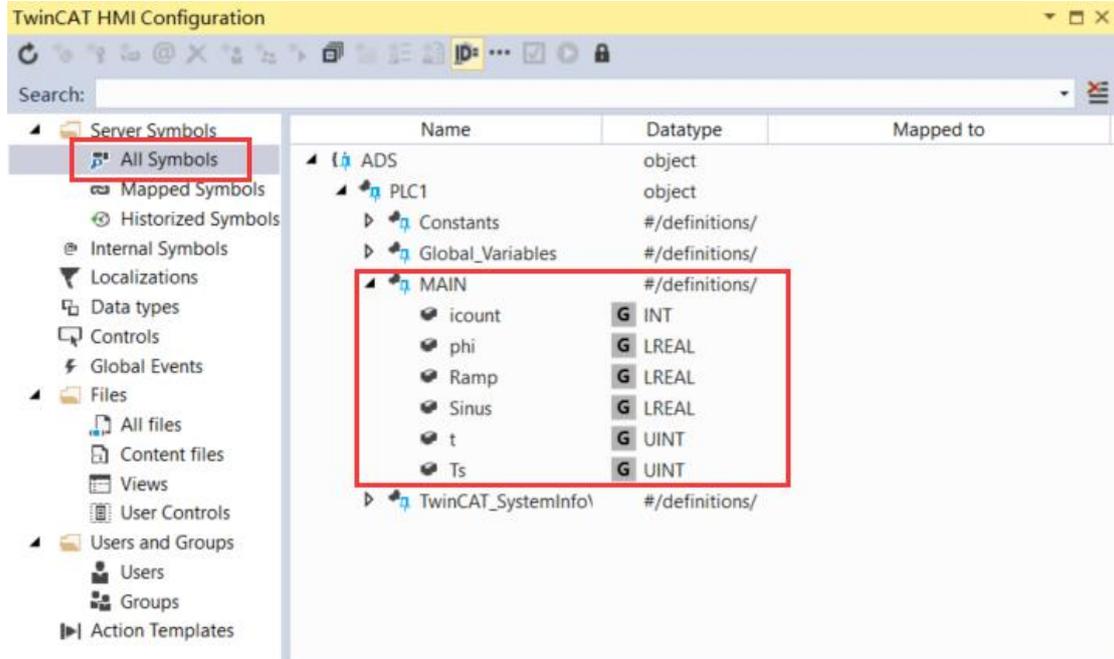
(1) 新建一个 TwinCAT 3 项目，完成 PLC 简单逻辑编写，用于一会 HMI 监控曲线。

```

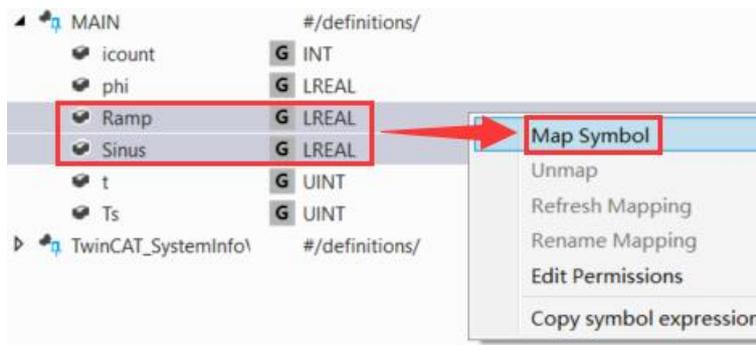
MAIN  x
1  PROGRAM MAIN
2  VAR
3      Ramp: LREAL;
4      Sinus: LREAL;
5
6      t: UINT;
7      Ts: UINT;
8      phi: LREAL;
9  END_VAR
10
11  // ramp
12  Ramp:=Ramp+0.002;
13
14  IF (Ramp > 1.0) THEN
15      Ramp:=-1.0;
16  END_IF
17
18  // sinus
19  t := t + 1;
20  Ts:=128*10;
21
22  IF (t>=2*Ts) THEN
23      t:=t-2*Ts;
24  END_IF
25
26  Sinus:=COS(PI*t/Ts);

```

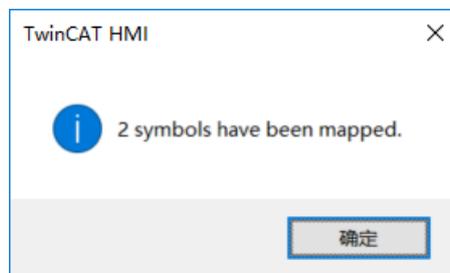
(2) 把 PLC 程序激活并且运行后，在同一个 solution 中新建一个 TwinCAT HMI 项目，随后直接点开 TwinCAT HMI Configuration，我们首先需要把 PLC 变量映射到 TwiCAT HMI 中，点开 All Symbols 可以直接浏览到 MAIN 程序中所对应的变量（使用类似 Target Browser）。



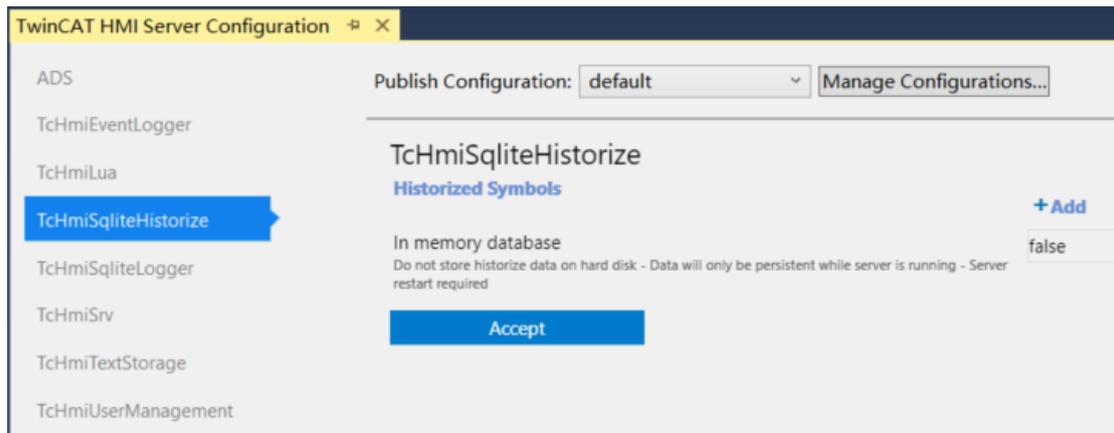
(3) 选择需要监控的 2 个变量 Ramp 和 Sinus，右键点击 Map Symbol 完成映射。



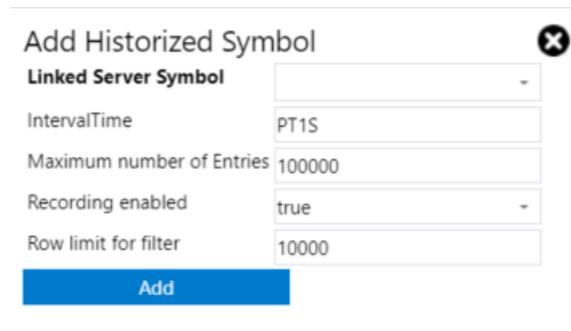
(4) 弹出窗口映射完成，并且在 Mapped Symbols 中可以看到所以映射的这 2 个变量。



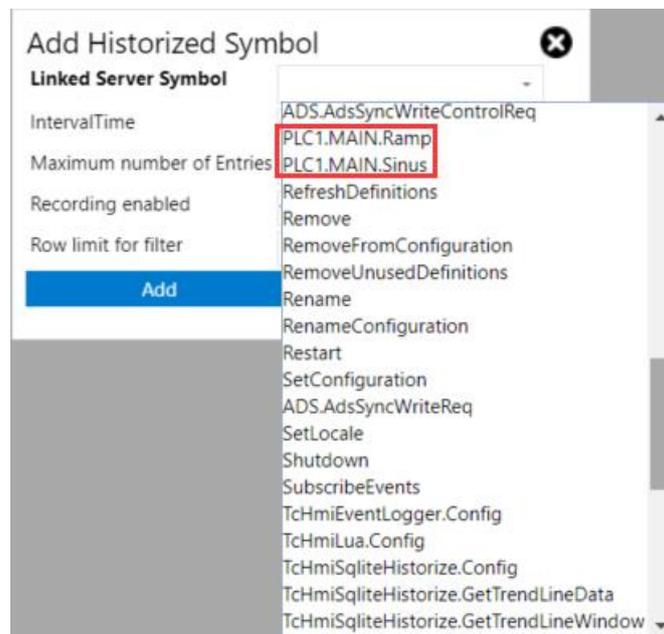
(6) 双击点开，点击 Historized Symbols，会出现 Add，点击进行变量添加。



(7) 弹出窗口，设置需要被数据库记录的变量。



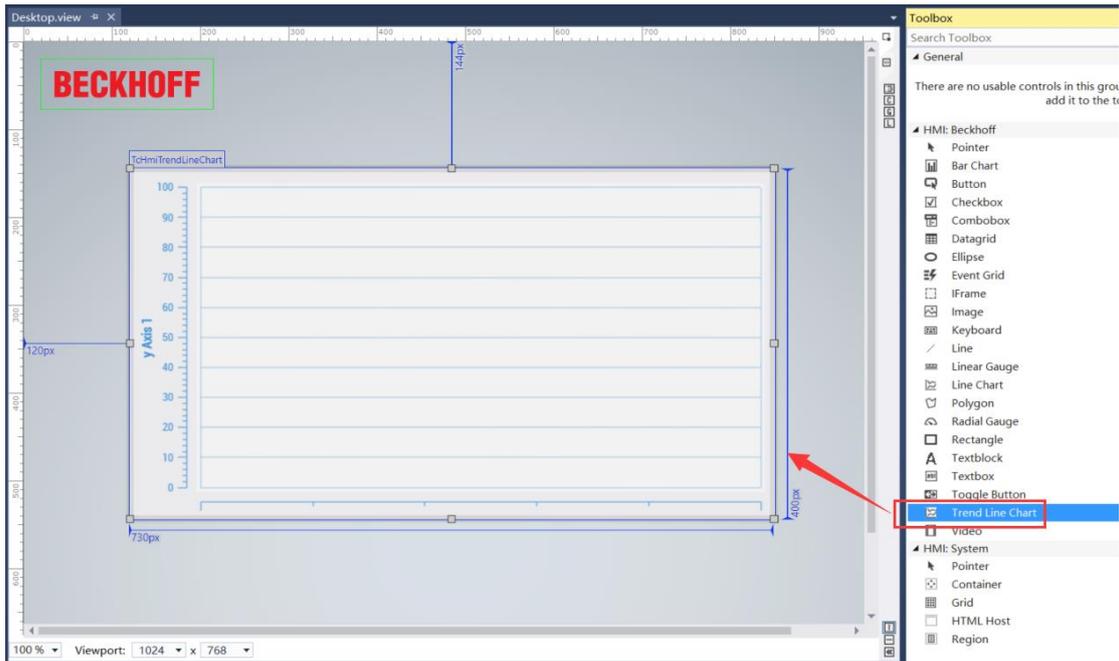
(8) Linked Server Symbol 下拉框中可以找到之前也映射过的 2 个 PLC 变量，分别完成添加，其他选项选择默认即可。



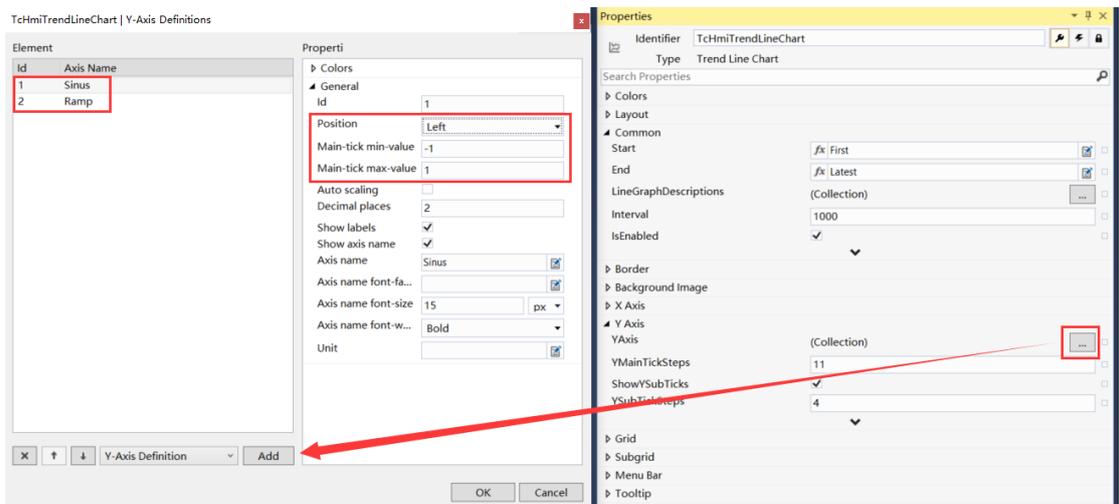
(9) 分别添加了这 2 个变量到 Server 后，点击 Accept 完成 Server 刷新。



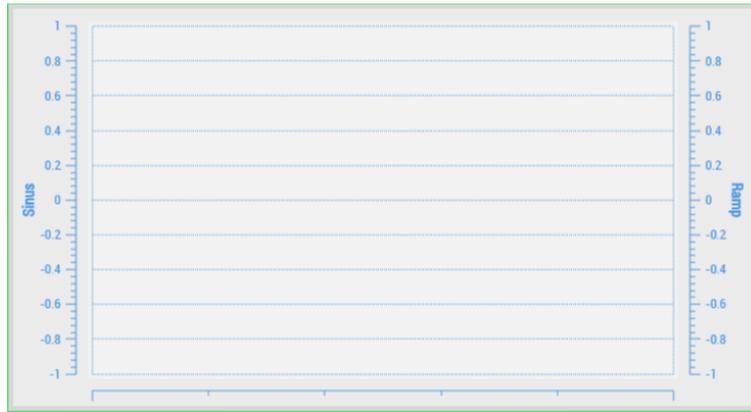
(10) 在主界面中拖拉 Trend Line Chart 控件



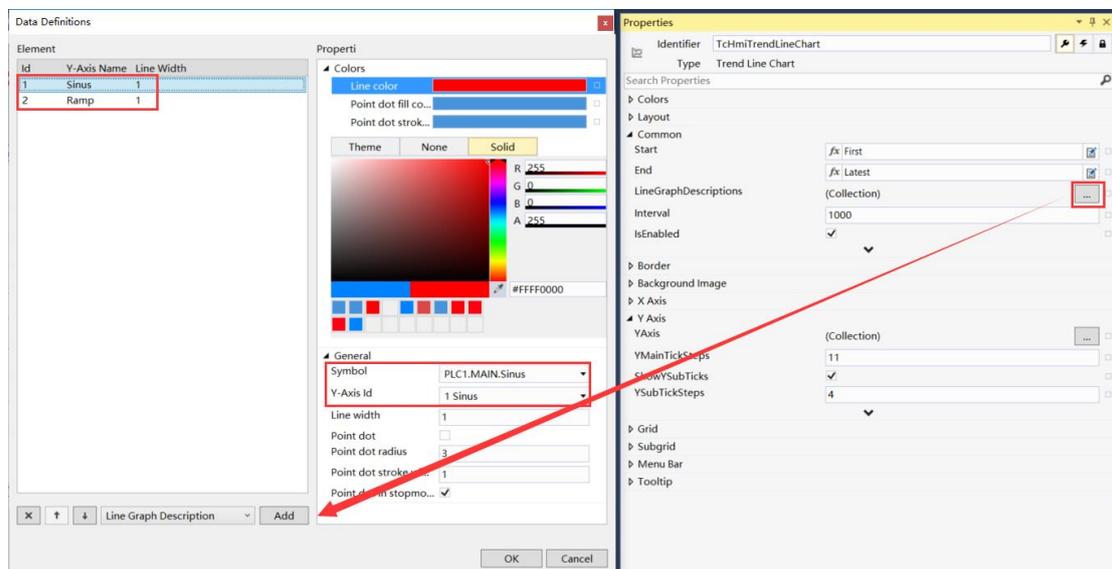
(11) 在属性栏先对 Y 轴坐标进行编辑，点开 YAxis 的 Collection，弹出窗口，点击 Add 分别添加 Sinus 与 Ramp，并且可以在 General 中对这 2 个 Y 轴进行刻度与摆放位置进行修改。



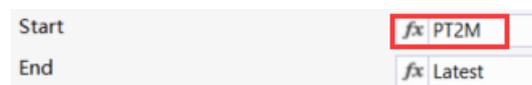
效果如下：



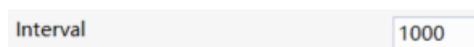
(12) 紧接着对配置所需要监控的变量, 在属性栏中点开 Common→LineGraphDescriptions, 弹出窗口点击 Add, 在 General 中配置所监控的 2 个变量, 并且可以对其颜色进行修改。



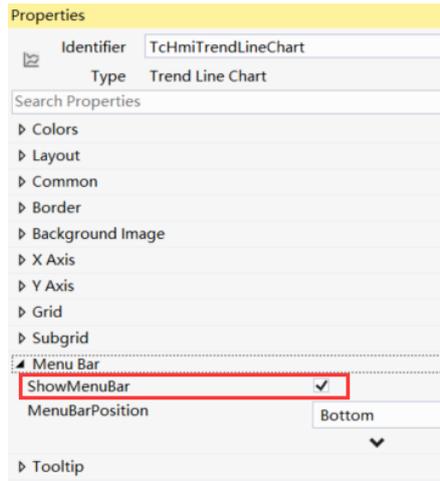
(13) 在 Common 中 Start 与 End 是设置起始于结束的监控时间, 默认是从头到尾全部监控并且显示, 当然也可以设置需要监控的时间段, 比如我只需要监控 2 分钟的最新数据, 那可以在 Start 中写 PT2M。



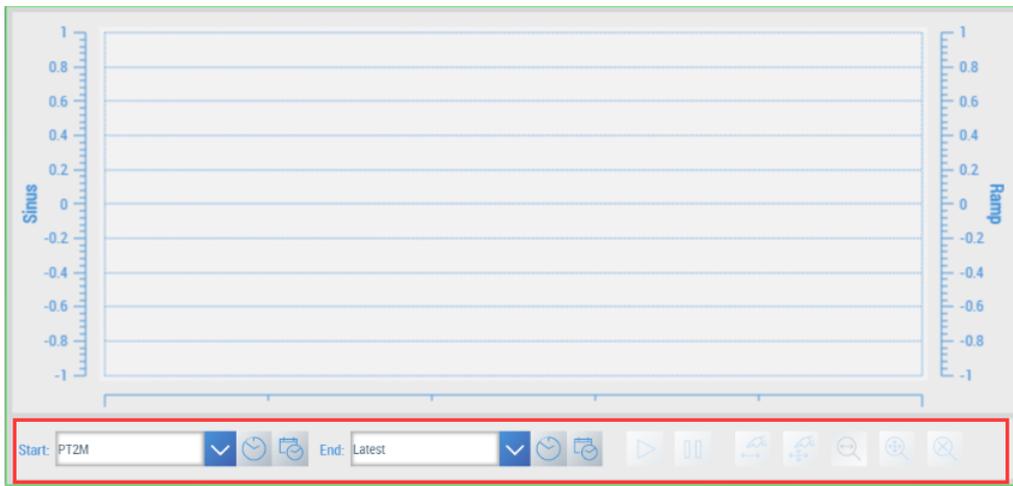
同时也可以设置趋势图的更新速率 Interval, 默认是 1000ms。



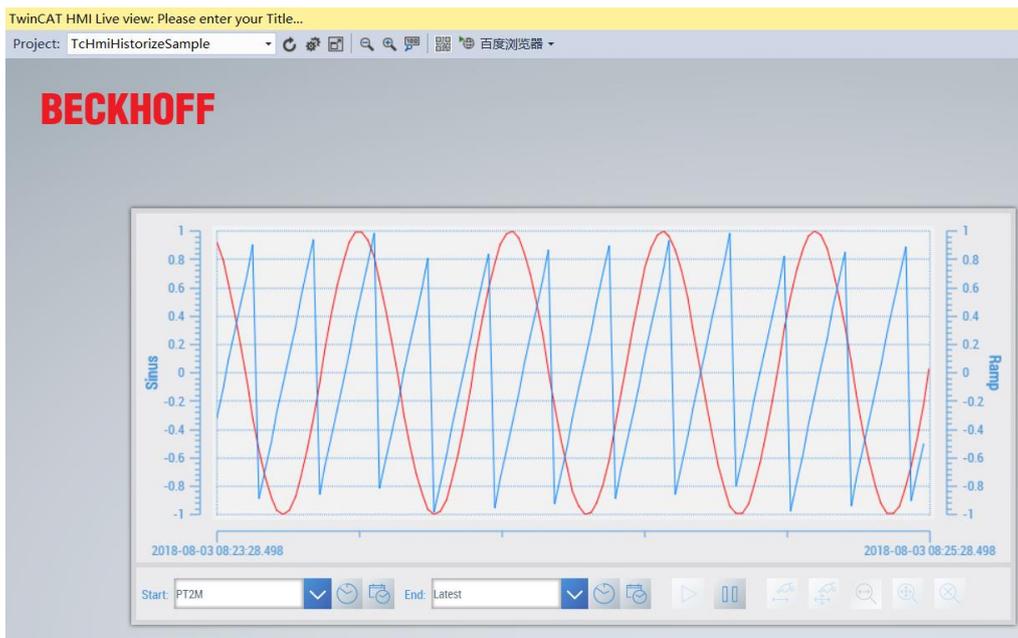
(14) 最后勾选 Menu Bar 中的 ShowMenuBar, 这样可以在 TcHmiTrendLineChart 中多出一行操作栏, 可以自由选择监控时间、范围, 以及放大缩小的功能。



效果如下：



(15) 全部设置好后在线浏览，可以实时监控 PLC 中所配置的 2 个 I 办理 Sinus 和 Ramp。



Menu Bar 中的功能不做详细介绍，类似 Scope View。

10. 实时数据表格连接 PLC

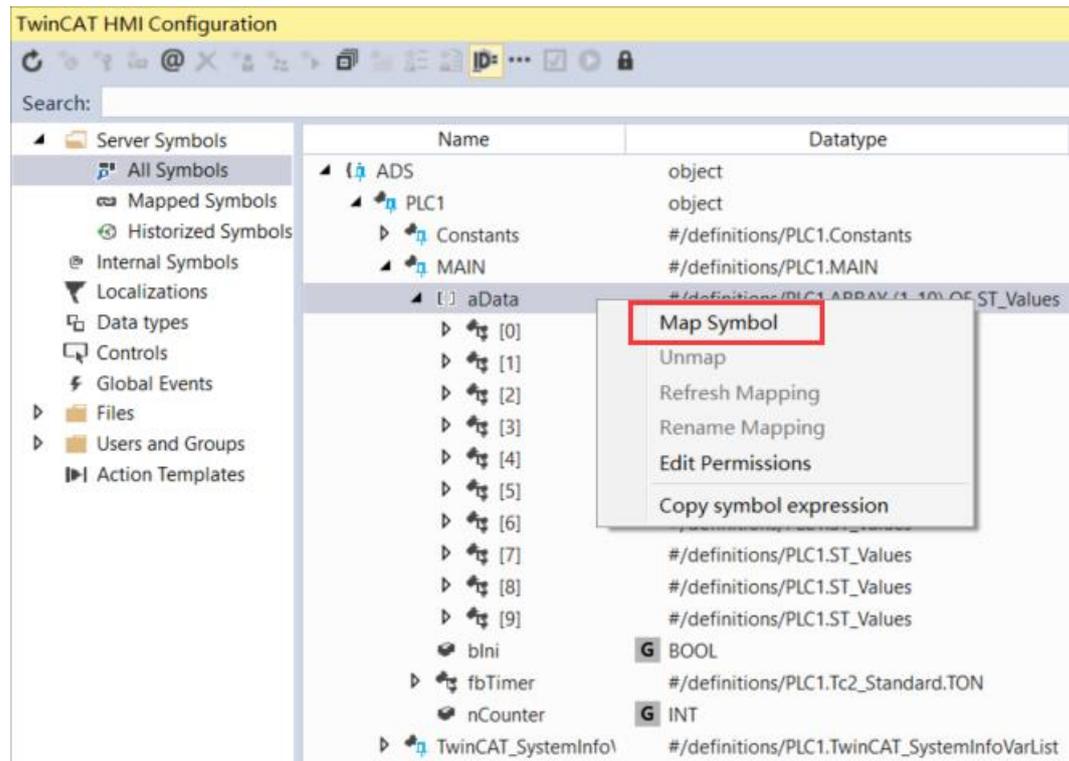
(1) 新建 PLC 项目，创建一个结构体。

```
ST_Values  + X
1  TYPE ST_Values :
2  STRUCT
3      bValue  : BOOL := TRUE;
4      nValue  : INT;
5      sText   : STRING;
6  END_STRUCT
7  END_TYPE
```

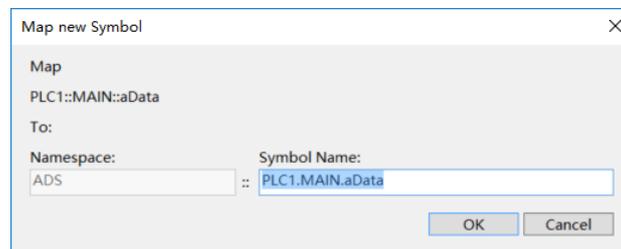
(2) 在主程序中编写一段代码，用于实例化结构体并且动态赋值，编写好后执行程序。

```
MAIN  + X
1  PROGRAM MAIN
2  VAR
3      aData : ARRAY [1..10] OF ST_Values;
4
5      bIni: BOOL := TRUE;
6      nCounter: INT;
7      fbTimer: TON;
8  END_VAR
9
10 // sets the text parameter for all members of aData in the first cycle
11 IF bIni THEN
12     FOR nCounter := 1 TO 10 DO
13         aData[nCounter].sText := CONCAT('Sample ', INT_TO_STRING(nCounter));
14     END_FOR
15     bIni := FALSE;
16 END_IF
17
18 // changes the integer value of all members of aData in case bValue is true
19 fbTimer(IN := TRUE, PT := T#0.5S);
20
21 IF fbTimer.Q THEN
22     fbTimer(IN := FALSE);
23
24     FOR nCounter := 1 TO 10 DO
25         IF aData[nCounter].bValue THEN
26             aData[nCounter].nValue := aData[nCounter].nValue + 1;
27
28             IF aData[nCounter].nValue > 100 THEN
29                 aData[nCounter].nValue := 0;
30             END_IF
31         END_IF
32     END_FOR
33 END_IF
```

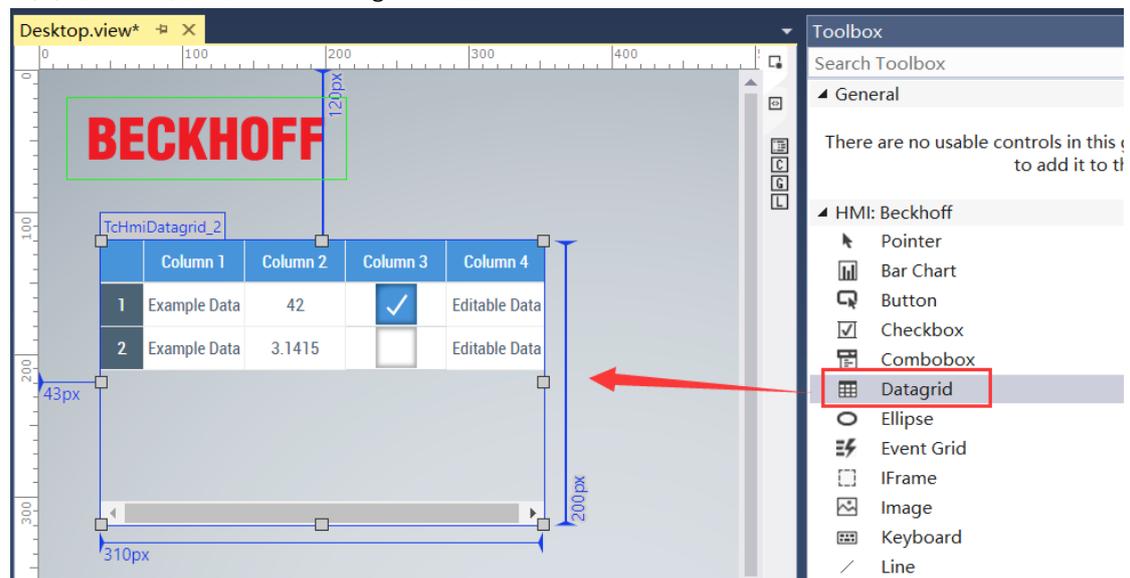
(3) 新建 TwinCAT HMI，在 Configuration 中添加 PLC 变量映射，在 All Symbols 中找到 aData 结构体数组，右键 Map Symbol 完成映射。



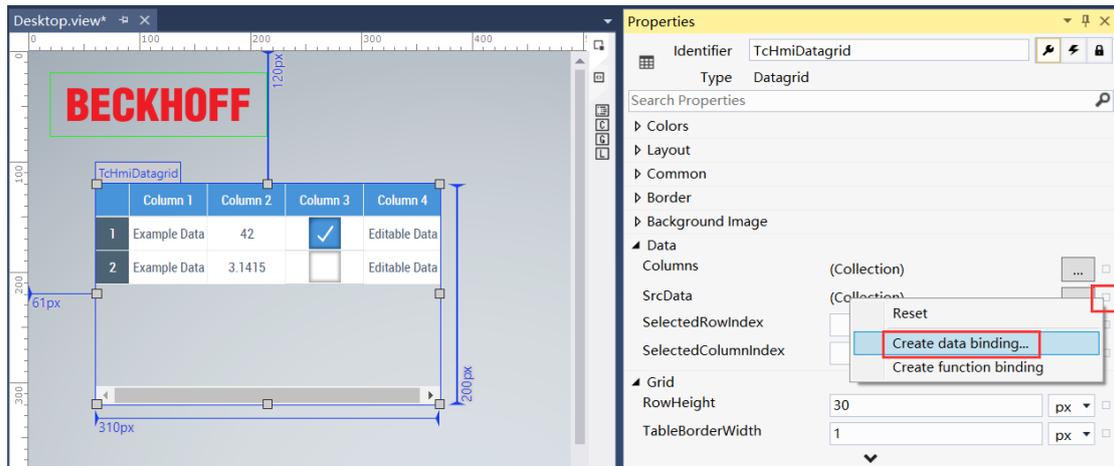
弹出窗口可以重新编辑变量名。



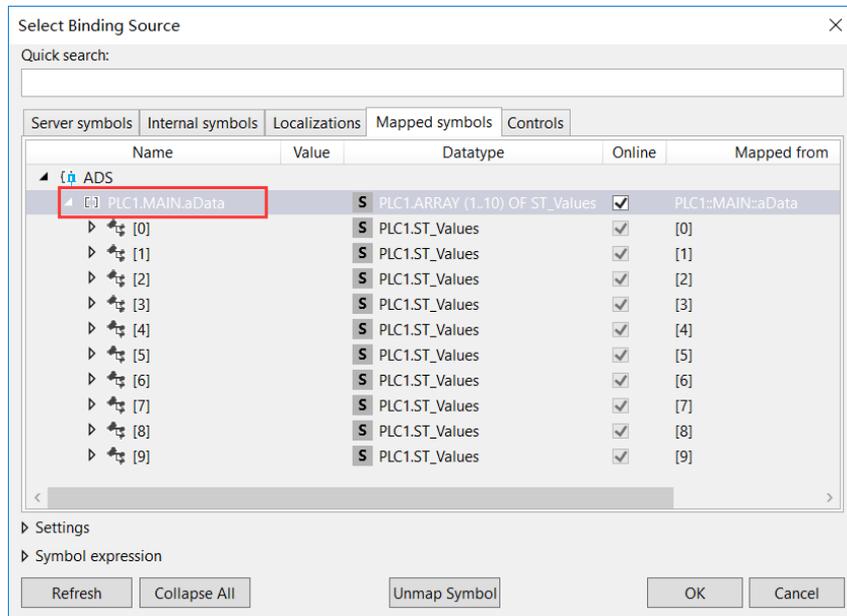
(4) 随后我们拖拉控件 Datagrid 到主界面。



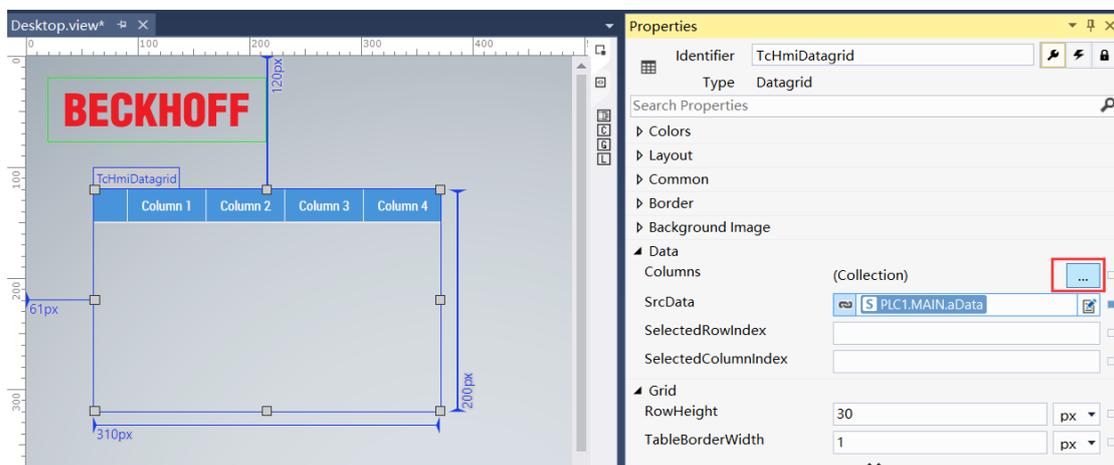
(5) 选中控件在属性中进行设置，首先点击 SrcData→Create data binding。



(6) 弹出窗口在 Mapped symbols 中找到 PLC1.MAIN.aData 完成变量的绑定。

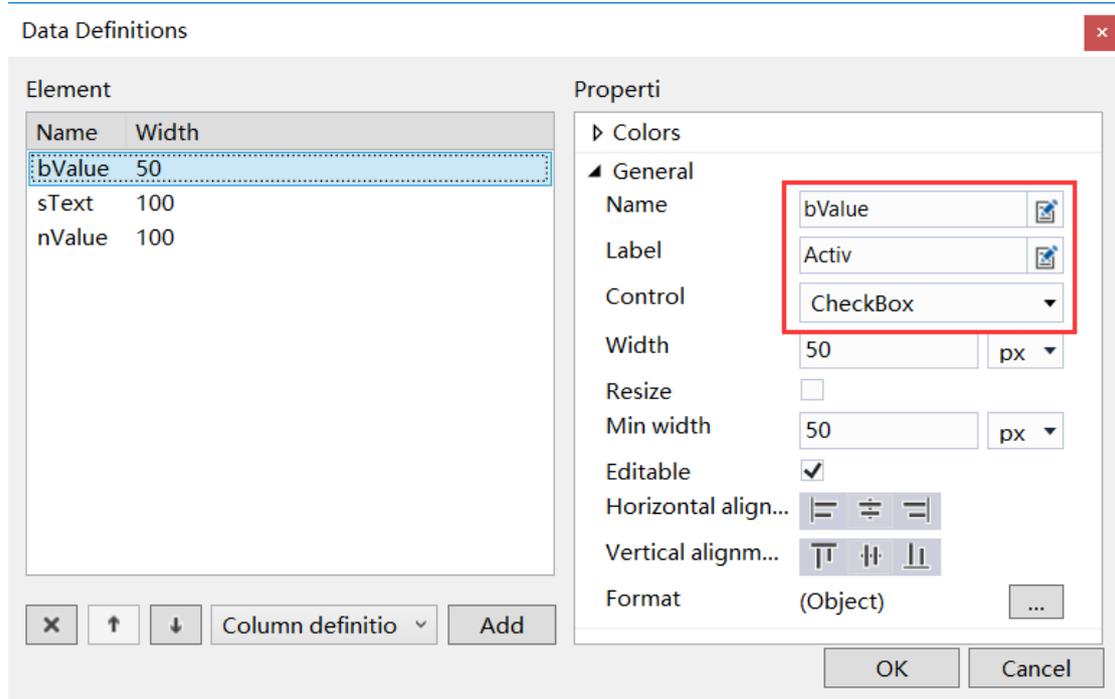


(7) 完成控件与变量的绑定后，我们需要对表格栏进行设置，点开 Columns 旁边的按钮。

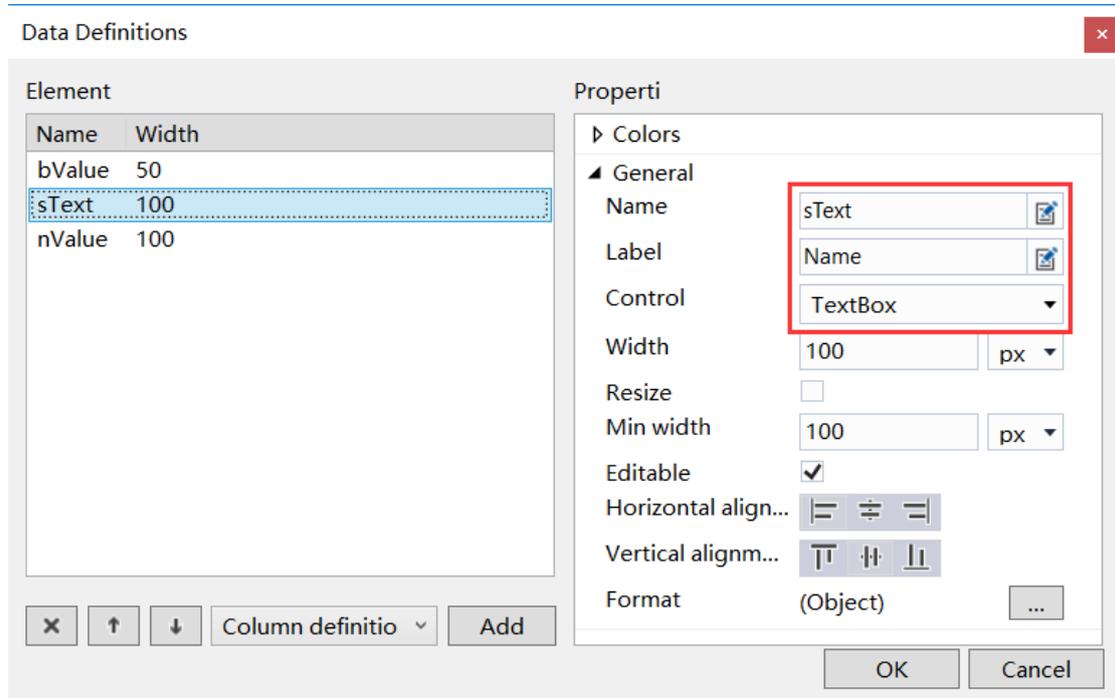


(8) 创建 3 个表格栏：

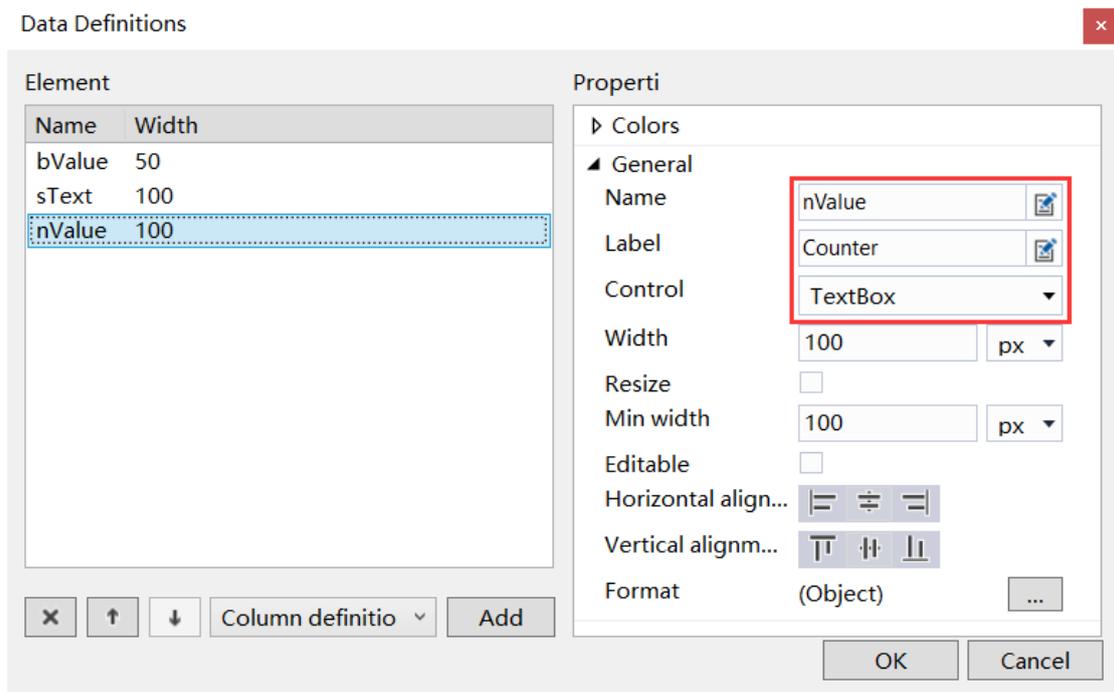
Name : bValue、Label : Activ、Control : CheckBox



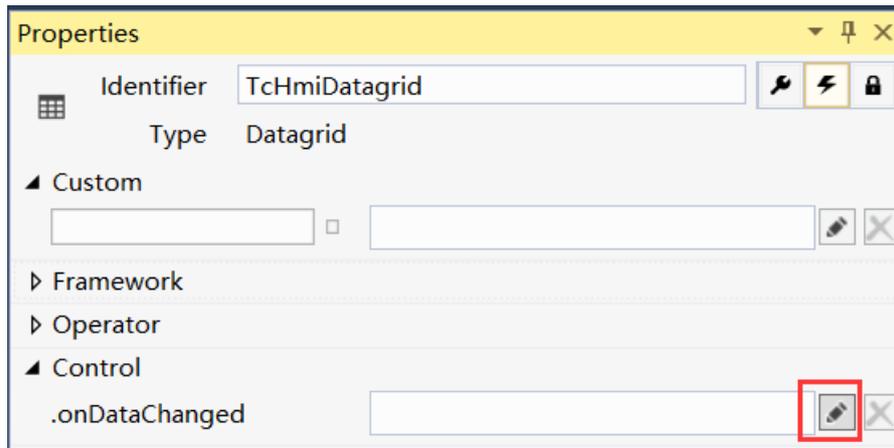
Name : sText、Label : Name、Control : TextBox



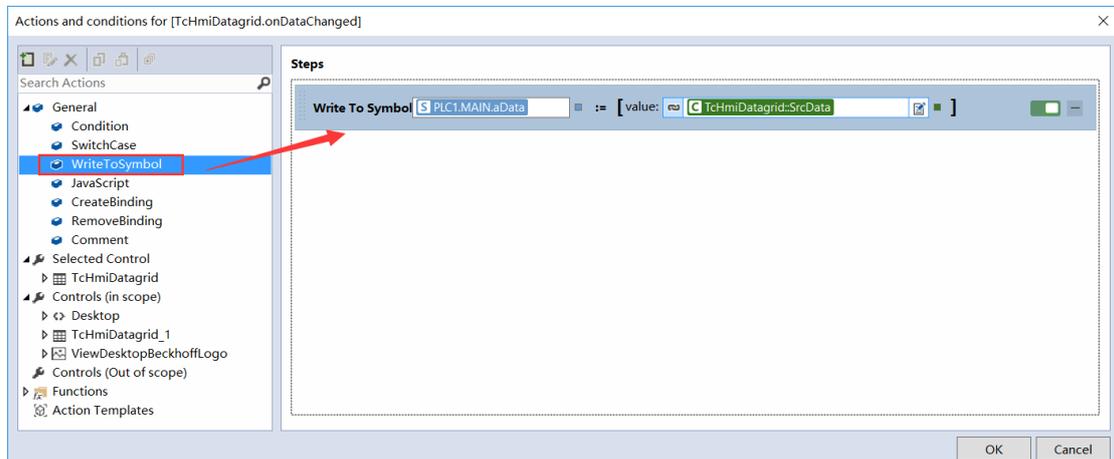
Name : sText、Label : Name、Control : TextBox



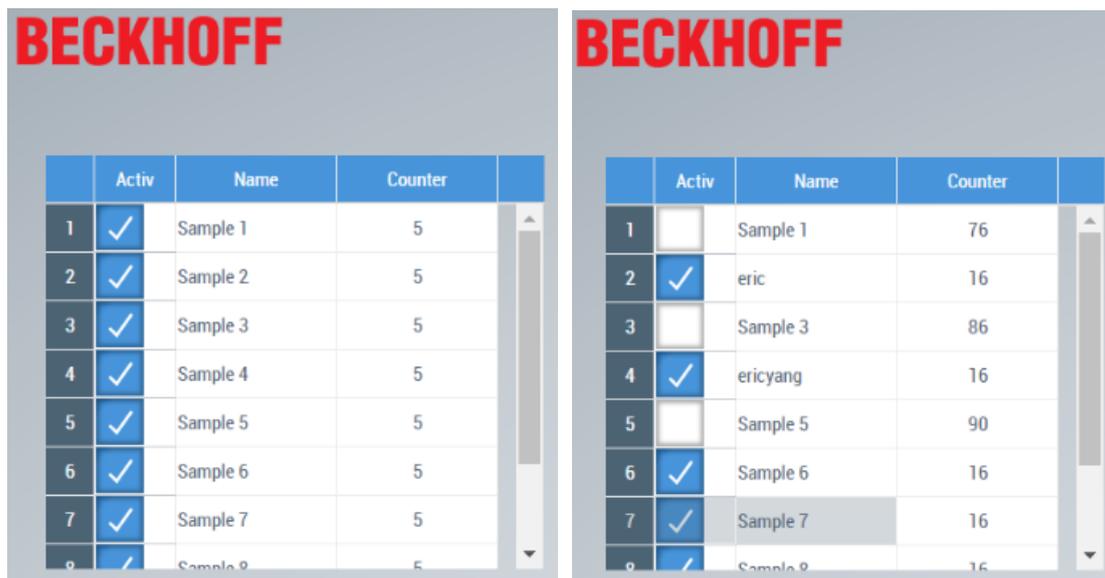
(10) 最后对表格设置写入事件，在属性事件栏中找到.onDataChanged 进行设置。



(11) 加入 WriteToSymbol, 并且把 TcHmiDatagrid::SrcData 赋值到 PLC1.MAIN.aData 中。



(9) 设置完成后，在线浏览效果如下，可以在线读取变量，修改变量。

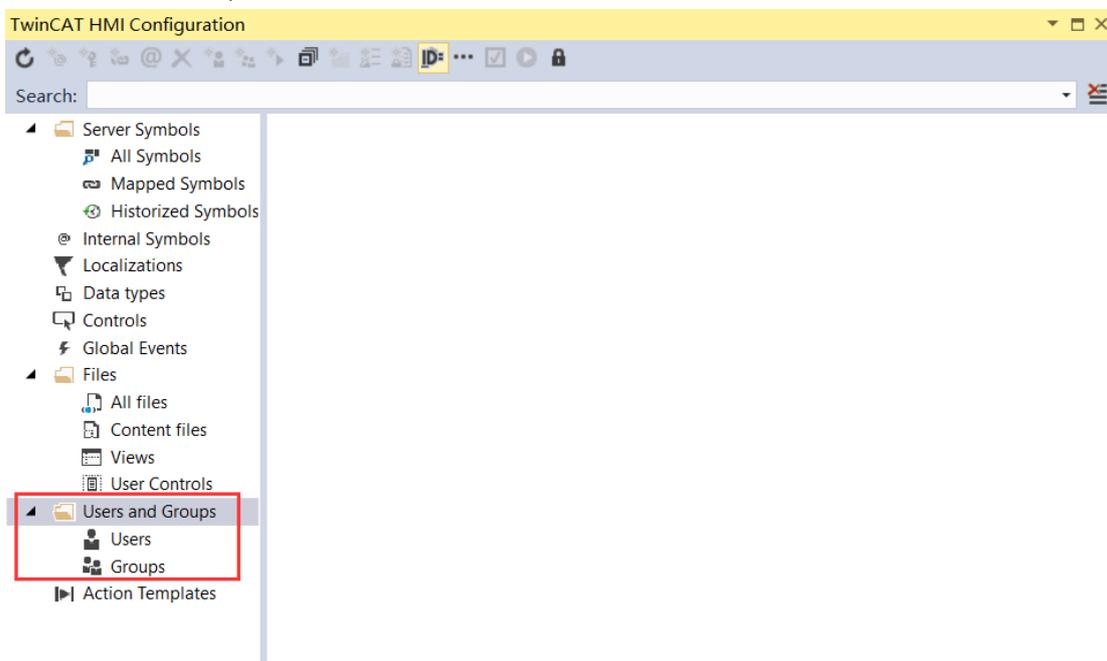


	Activ	Name	Counter
1	<input checked="" type="checkbox"/>	Sample 1	5
2	<input checked="" type="checkbox"/>	Sample 2	5
3	<input checked="" type="checkbox"/>	Sample 3	5
4	<input checked="" type="checkbox"/>	Sample 4	5
5	<input checked="" type="checkbox"/>	Sample 5	5
6	<input checked="" type="checkbox"/>	Sample 6	5
7	<input checked="" type="checkbox"/>	Sample 7	5
8	<input checked="" type="checkbox"/>	Sample 8	5

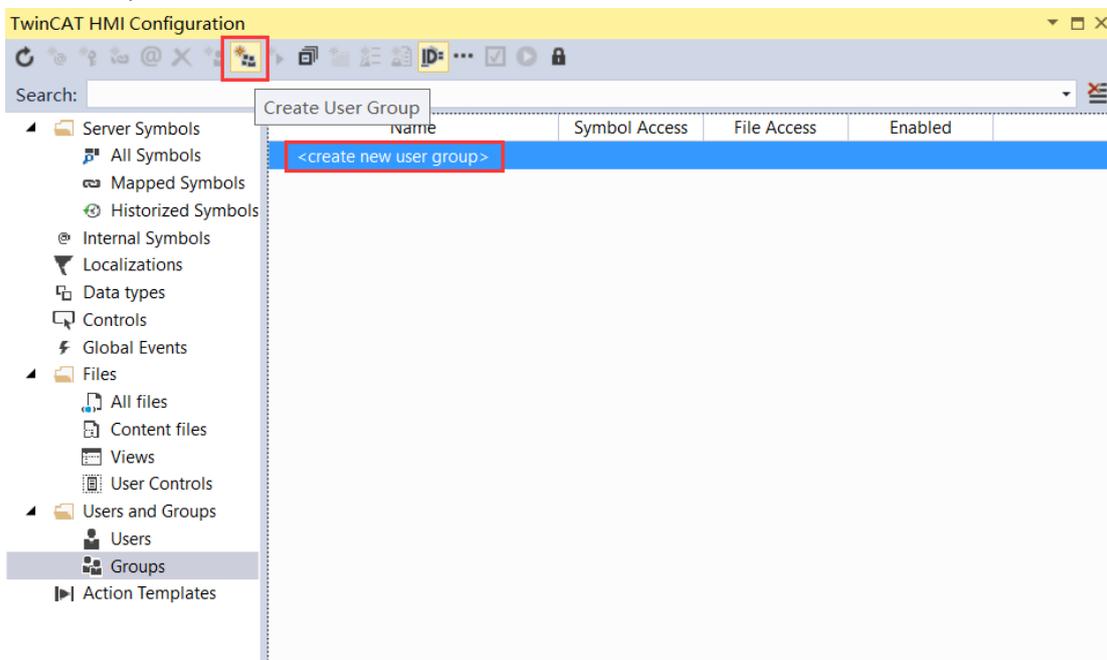
	Activ	Name	Counter
1	<input type="checkbox"/>	Sample 1	76
2	<input checked="" type="checkbox"/>	eric	16
3	<input type="checkbox"/>	Sample 3	86
4	<input checked="" type="checkbox"/>	ericyang	16
5	<input type="checkbox"/>	Sample 5	90
6	<input checked="" type="checkbox"/>	Sample 6	16
7	<input checked="" type="checkbox"/>	Sample 7	16
8	<input checked="" type="checkbox"/>	Sample 8	16

11. UserManagement 用户管理

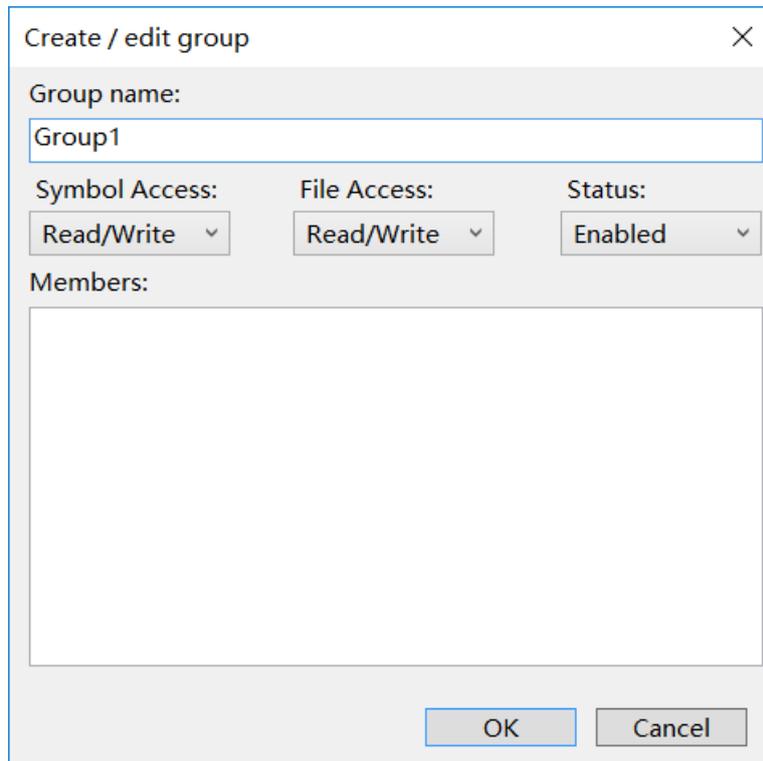
(1) 我们在之前做好的 HMI 中加入用户管理功能，首先打开 TwinCAT HMI Configuration，在 Users and Groups 中配置“组”和“用户”。



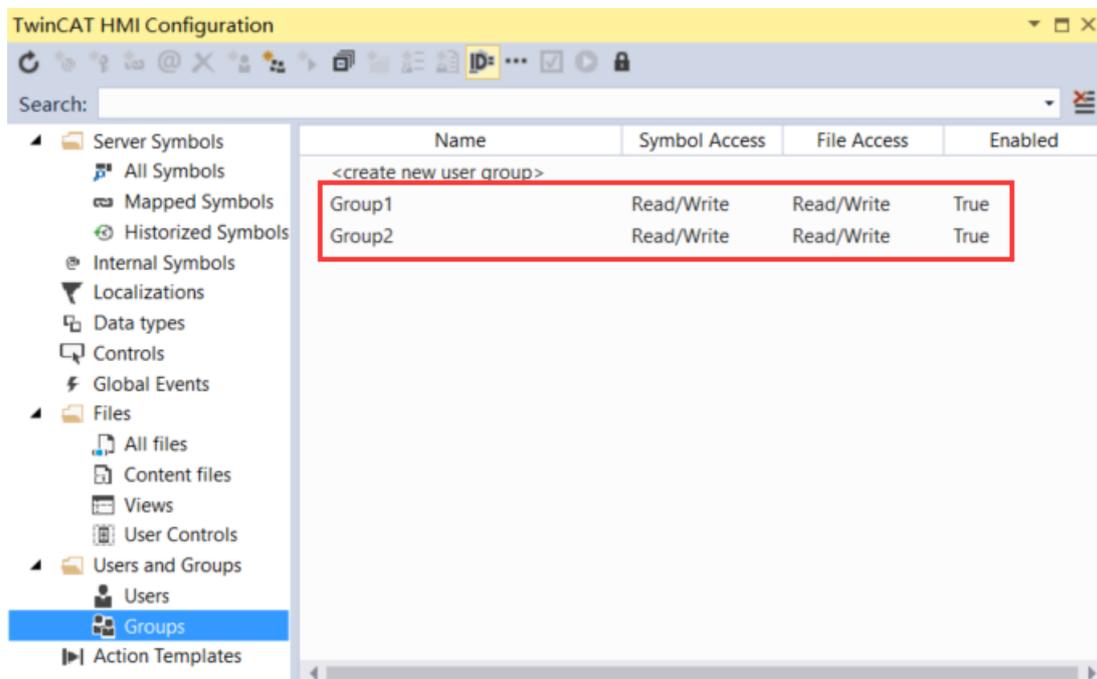
(2) 首先创建一个组，在 Groups 中双击 create new user group 或者点击工具栏中的 Create User Group 都可以创建一个组。



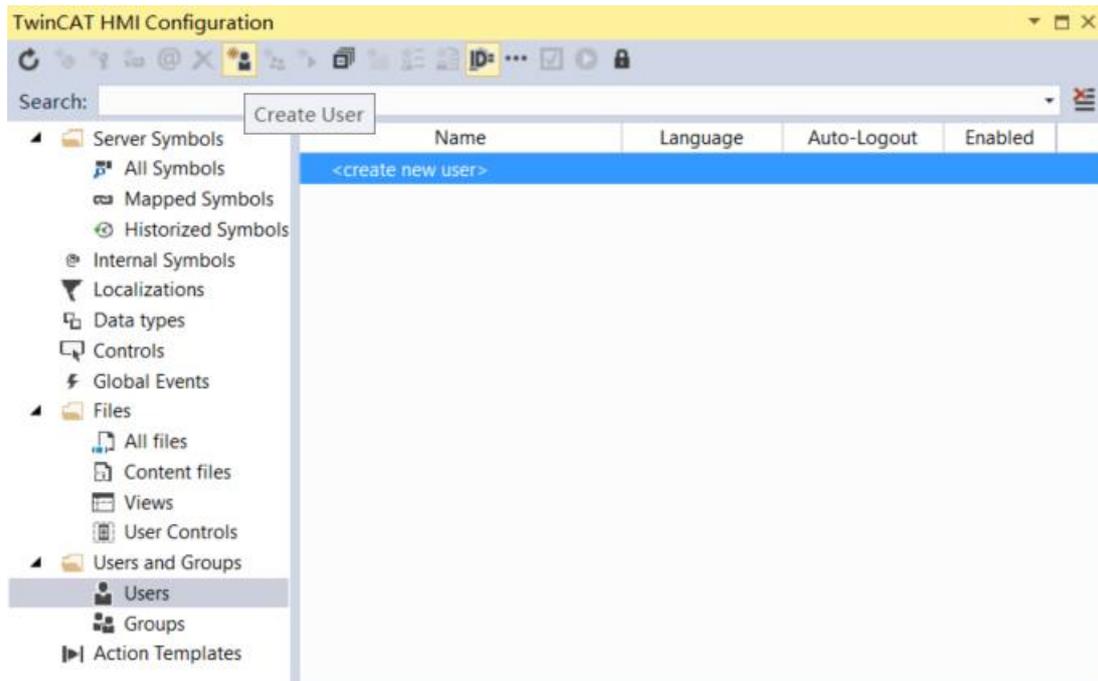
(3) 弹出窗口输入组名以及所对应组的权限，分别是 mapping 变量的读写权限、TwinCAT HMI Server 处理文件或者文件夹的读写权限。



(4) 分别创建 Group1 和 Group2。



(5) 紧接着创建用户，在 Users 中双击 create new user 或者点击工具栏中的 Create User 都可以创建用户。



(6) 填写对应用户信息，用户名、密码、登录语言、所属的组（可以属于多个组）以及自动登出时间。

The 'Create / edit user' dialog box contains the following fields and options:

- User name:** Text box containing 'User1'.
- Password:** Text box containing '*'.
- Confirm Password:** Text box containing '*'.
- Language:** Dropdown menu showing 'en-US'.
- Member of the following groups:** A list with two items: 'Group1' (checked) and 'Group2' (unchecked).
- Auto-Logout Time:** Three spin boxes for 'Days' (30), 'Hours' (0), and 'Minutes' (0).
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

(7) 分别创建 2 个用户，其登录语言和登出时间各有不同。

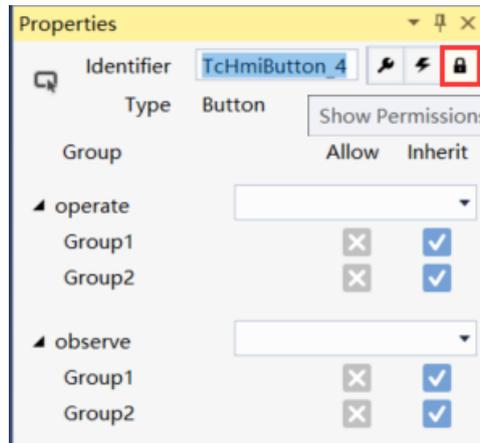
Name	Language	Auto-Logout	Enabled
<create new user>			
User1	en-US	P30D	<input checked="" type="checkbox"/>
User2	zh-CN	PT10M	<input checked="" type="checkbox"/>

(8) 组 and 用户都创建好后，就可以对现有的 HMI 进行权限规划，每一个基础控件 (control)、自定义控件 (user control)、子页面 (content)、主页面 (view) 的属性栏中选择 Show Permissions (小锁标志) 都有对应的权限管理。

权限分为 operate (操作) 和 observe (可见)。

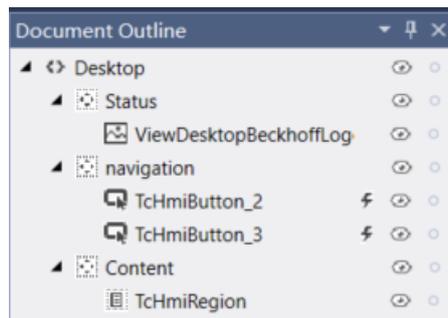
Allow : 允许在对应组进行操作或者可见权限。

Inherit : 集成上一级控件、自定义控件、子页面的权限设置。

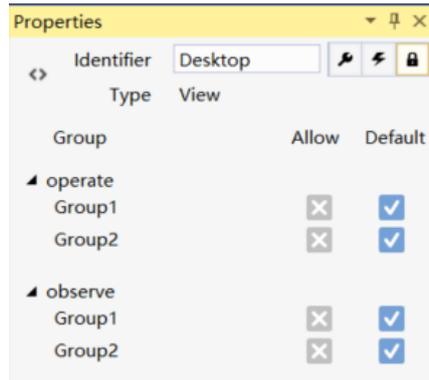


(9) 这里针对 Inherit 做一个解释，在 TwinCAT HMI 中所有组件都有附属关系，比如自定义控件 (user control) 中的基础控件 (control) 就附属于此自定义控件中，子页面 (content) 中的个控件也都附属于此子页面中，容器 (containter) 中的内容都属于容器，主页面中所有内容都附属于主页面。

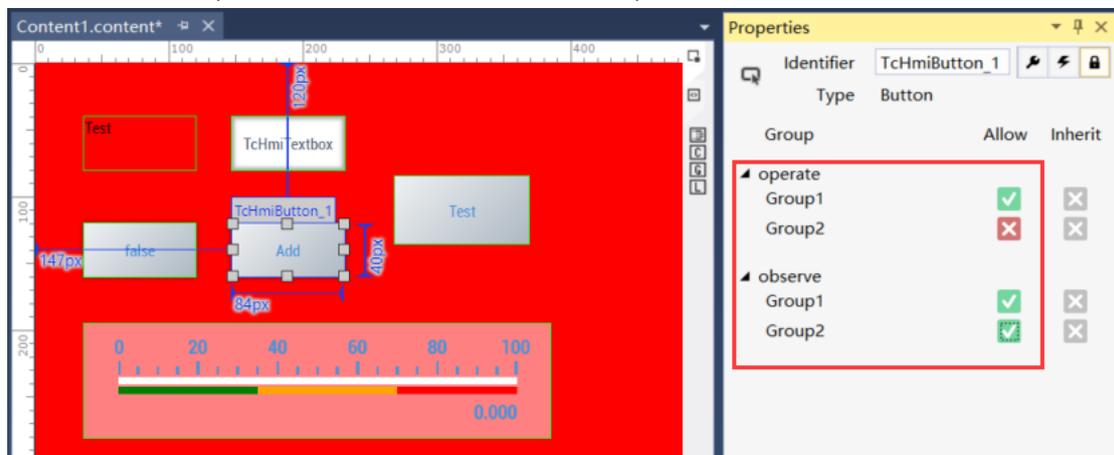
通过打开每一层的 Document Outline 都可以看到从属关系。



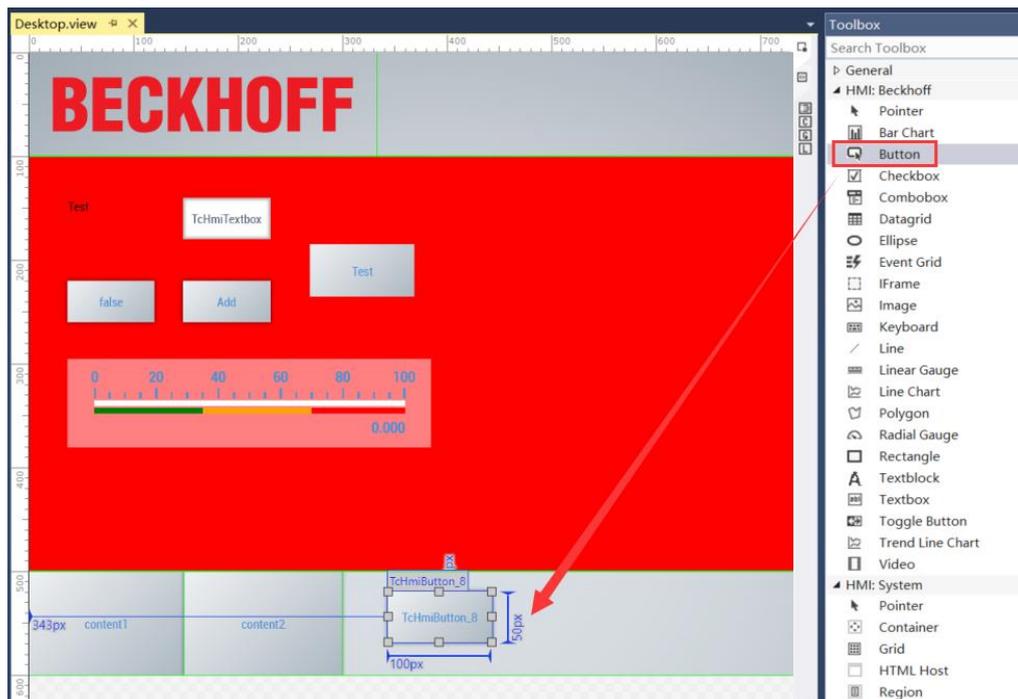
因此 Inherit 就意味着用户组的权限来自更高级别的父类控件，最高一级为主页面 (view)，这就是为什么 view 的 Show Permissions 是没有 Inherit 选项，而是 Default (=Allow)。



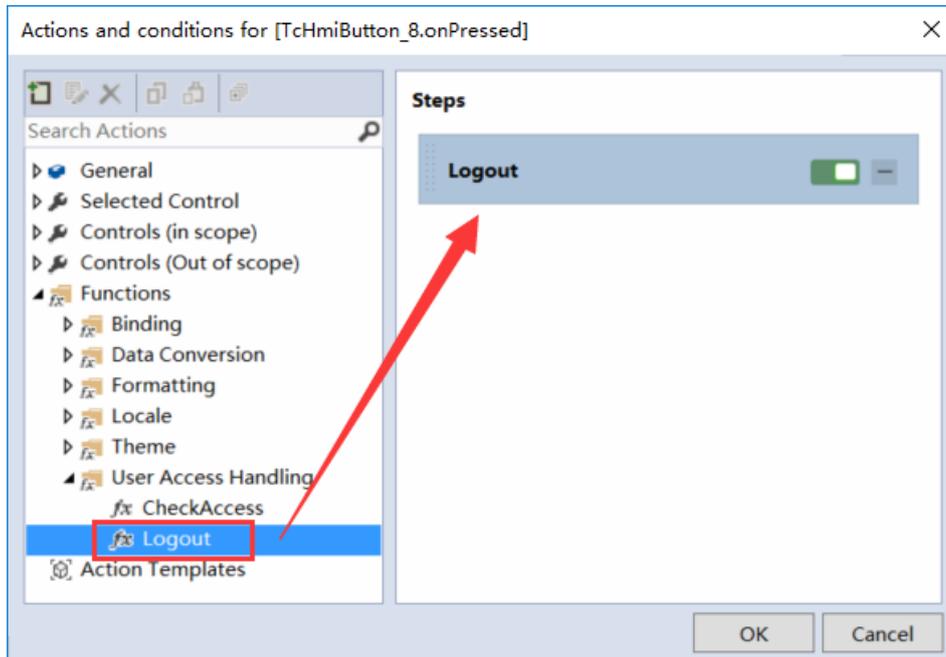
(10) 我们可以尝试着把之前项目中的一些控件设置权限，比如把 Content1 中 Add 控件的权限设置为 Group1 中用户登录可见/可操作，Group2 中用户登录可见/不可操作。



(11) 设置好后我们需要在主界面的导航栏中加入用户登出的控件，回到主页面，拖拉一个 button 到 navigation 中。

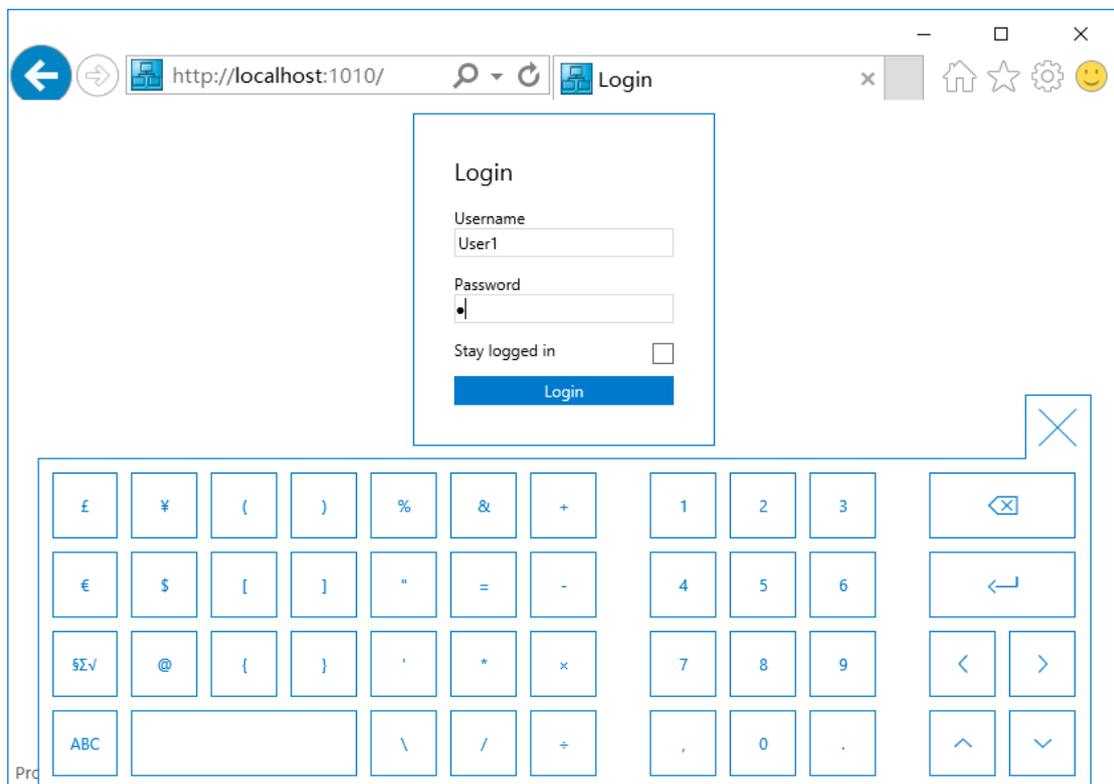


(12)修改 button 的 Text 为 Logout, 然后双击设置事件, 在 Functions→User Access Handling 中找到 Logout 的功能进行添加。

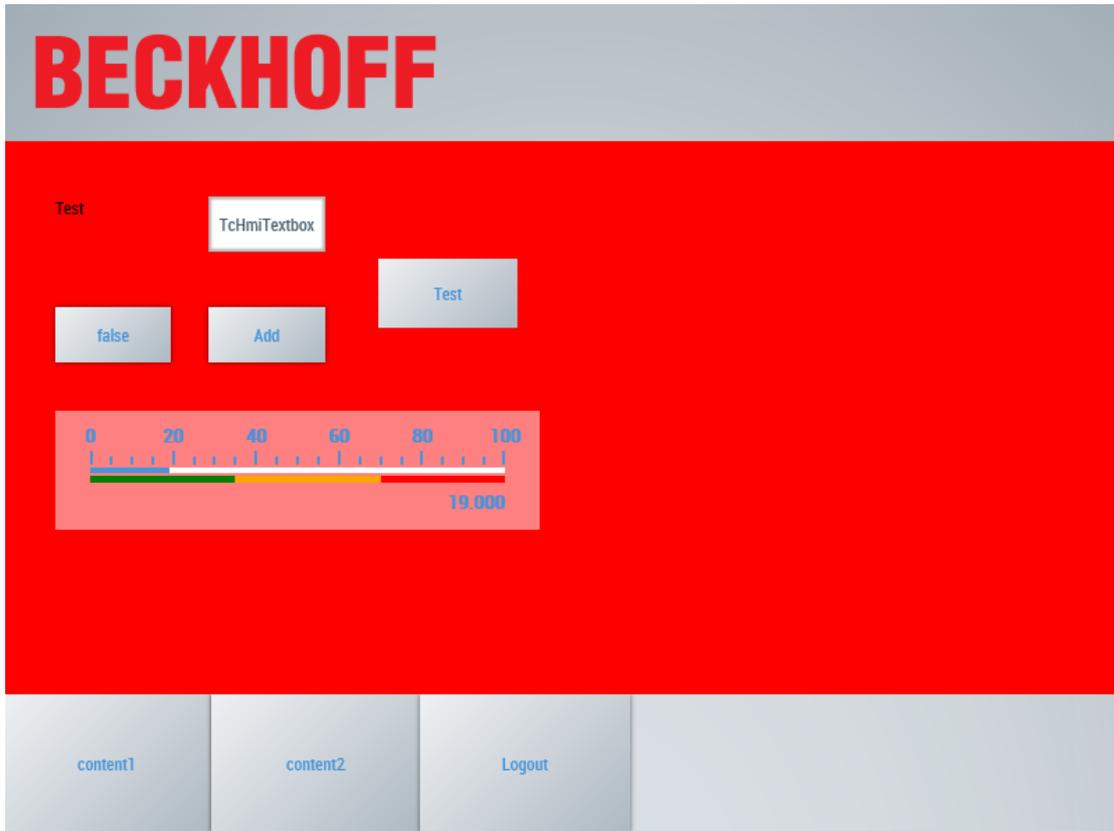


(13) 全部设置完成, 我们需要发布到 server 中, 用户管理需要发布才可以看得到效果。

(14) 发布好后, 打开 IE 进行浏览, 首先弹出 Login 界面, 在这里可以选择对应用户, 输入密码后完成登录。



(15) 可以看到 Add 控件的权限在 User1 登录可见/可操作，User2 登录可见/不可操作。
User1 :



User2 :

