



大观系列触摸屏modbus说明书

DG

产品版本：V1.0

上海英硕自动化科技有限公司

版权所有，保留一切权利。

未经上海英硕自动化科技有限公司许可，任何单位和个人不得擅自摘抄、复制本书（软件等）的全部或部分，不得以任何形式（包括资料和出版物）进行传播。

版权所有，侵权必究。内容如有改动，恕不另行通知。

**All Copyright© reserved by Shanghai ENSURE Automation
Technology Co.,Ltd**

All rights reserved

The information in this document is subject to change without prior notice. No part of this document may in any form or by any means (electronic, mechanical, micro-coping, photocopying, recording or otherwise) be reproduced, stored in a retrieval system or transmitted without prior written permission from Shanghai ENSURE Automation Technology Co.,Ltd.

前言

版权所有，保留一切权利。

没有得到上海英硕自动化科技有限公司许可，任何单位和个人不得擅自摘抄、复制本书(软件等)的全部或部分，不得以任何形式(包括资料和出版物)进行传播。

为了确保能够正确使用本触摸屏modbus控件，请在使用前仔细阅读本说明书。

本公司保留对产品不断改进的权利，恕不另行通知。

版权所有，侵权必究。



目 录

1. 适用范围.....	1
2. 开发环境版本.....	1
3. MODBUS 示例工程概述	1
3.1 串口屏协议说明.....	1
3.2 开发工程流程.....	1
3.3 MODBUS 协议运行说明.....	1
3.4 工程画面概览.....	2
3.5 协议与变量配置概览.....	3
4. 工程详解.....	4
4.1 创建工程.....	4
4.1.1 新建工程.....	4
4.1.2 新建画面.....	4
4.1.3 配置 modbus 协议	4
4.2 画面与逻辑配置.....	5
4.2.1 首页.....	6
4.2.2 工程说明.....	7
4.2.3 线圈.....	8
4.2.4 离散输入.....	9
4.2.5 保持寄存器.....	11
4.2.6 输入寄存器.....	13
4.2.7 按钮控件.....	15
4.2.8 文本控件.....	16
4.2.9 进度条、仪表、滑块.....	17
4.2.10 RTC 控件	19
4.2.11 动画控件.....	21
4.2.12 图标控件.....	23
4.2.13 曲线控件.....	24
4.2.14 菜单控件.....	26
4.2.15 选择控件.....	28
4.2.16 二维码.....	30
4.2.17 数据记录控件.....	32
4.2.18 历史曲线控件.....	36
4.2.19 隐藏显示控件.....	38
4.2.20 多语言	40



1. 适用范围

文档适合大观系列触摸屏所有modbus协议的串口产品。

2. 开发环境版本

1、DGTFT软件版本：V1.0.0.000 及以上； 版本查看：

- (1) 打开软件,右下角显示的软件版本号。
- (2) 打开DGTFT， 点击帮助->关于DGTFT可以查看当前软件版本号。

2、串口屏硬件版本：V2.22.1025.XXX及以上。

版本查看：

- (1) 画面中创建一个文本控件，属性“输入方式”选择“系统变量”，属性“系统变量”选择“固件版本号”，编译工程下载到屏中查看。

3. MODBUS示例工程概述

3.1 串口屏协议说明

英硕大观系列触摸屏均有两种协议版本：一种是英硕指令协议版本、一种是MODBUS协议版本，本示例工程主要用于演示如何开发MODBUS协议的工程。

3.2 开发工程流程

串口屏工程的开发流程遵循以下流程：

- 1、新建工程
- 2、创建并配置画面和控件
- 3、配置MODBUS协议，主要是建立变量、逻辑处理、MiniC脚本等；
- 4、控件属性中绑定在【协议与变量设置】中定义的变量。

MODBUS协议配置主要涉及DGTFT两个部分：

- (1) 菜单【工具】->【协议与变量设置】；
- (2) 菜单【工具】->【MiniC脚本编程】。

3.3 MODBUS协议运行说明

开发的MODBUS工程，经过DGTFT编译后会生成可以下载到串口屏中的工程包，编译的过程会将配置的MODBUS信息转换为串口屏工程文件；串口屏会根据工程文件运行，自动生成符合MODBUS协议的格式的指令。如果屏做主机，会周期性的通过串口发送这些指令到从机，接收从机返回的指令并解析，更新绑定的控件或执行预先配置的动作等。

3.4 工程画面概览

本工程中画面功能介绍主要分三组，如图 3-1所示：

- 1、 区域1：演示MODBUS协议的中的线圈、离散输入、保持寄存器、输入寄存器的数据显示，直接显示出数值；
- 2、 区域2：演示如何使用控件显示关联的变量，即MODBUS数据；
- 3、 区域3：演示如何实现显示隐藏控件、多语言控制等分组1、分组2中未使用的功能；

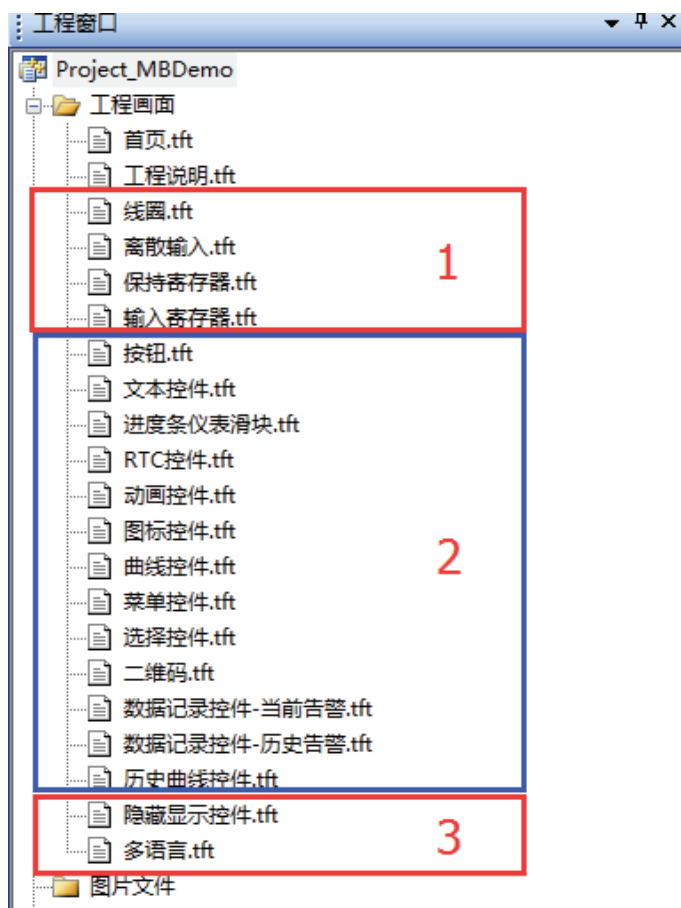


图 3-1 工程画面

4. 工程详解

4.1 创建工程

首先做好工程的准备工作，主要包括创建工程、建立页面、配置MODBUS协议类型等。

4.1.1 新建工程

打开菜单【文件】->【新建工程】，弹出新建工程窗口，设置好工程名称、路径、设备类型，点击【确定】即可。



图 4-1 新建工程

4.1.2 新建画面

新建的工程默认会创建一个画面，也可以自己添加画面。

4.1.3 配置modbus协议

打开菜单【工具】->【协议与变量设置】，双击左侧文件路径，在右侧的协议设置处，设置协议类型，本处设置 MODBUS 从机。

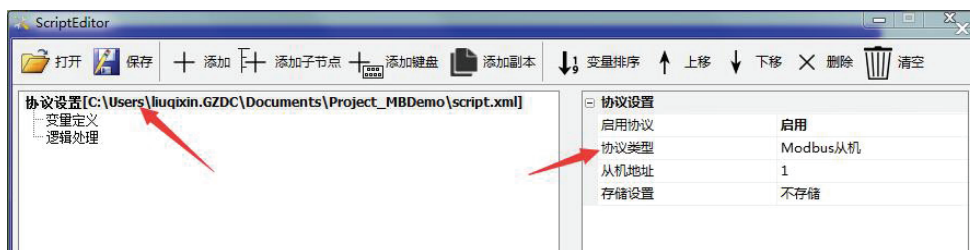


图 4-2 配置协议类型

至此，MODBUS协议的工程雏形已经建立，后面即可根据实际需求配置工程画面和逻辑。

4.2 画面与逻辑配置

工程配置方式采用画面和MODBUS逻辑同时进行。

工程逻辑结构采用首页建立导航按钮，不同按钮切换到不同的画面，每个画面有返回首页按钮，以返回首页。

以下为对工程各个画面配置的介绍。

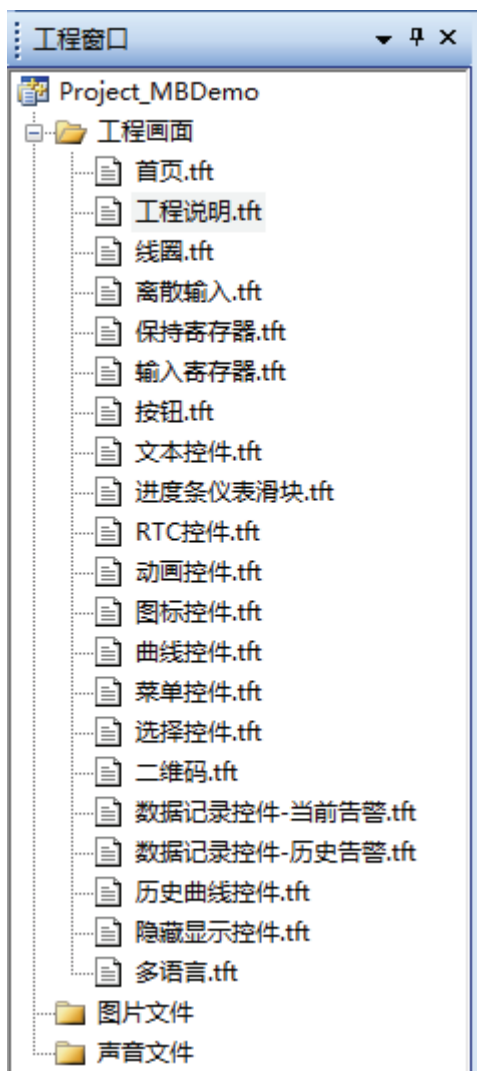


图 4-3 工程画面

4.2.1 首页

【首页】画面中建立各个按钮，如图 4-4所示，按钮可以配置切换到不同画面。



图 4-4 首页

例如：按钮2“工程说明”，可以按钮的属性用于切换到画面【工程说明】，按钮配置如图 4-5所示，其它按钮同理。



图 4-5 按钮配置

4.2.2 工程说明

【工程说明】画面为对本工程简单的说明，无其它功能。

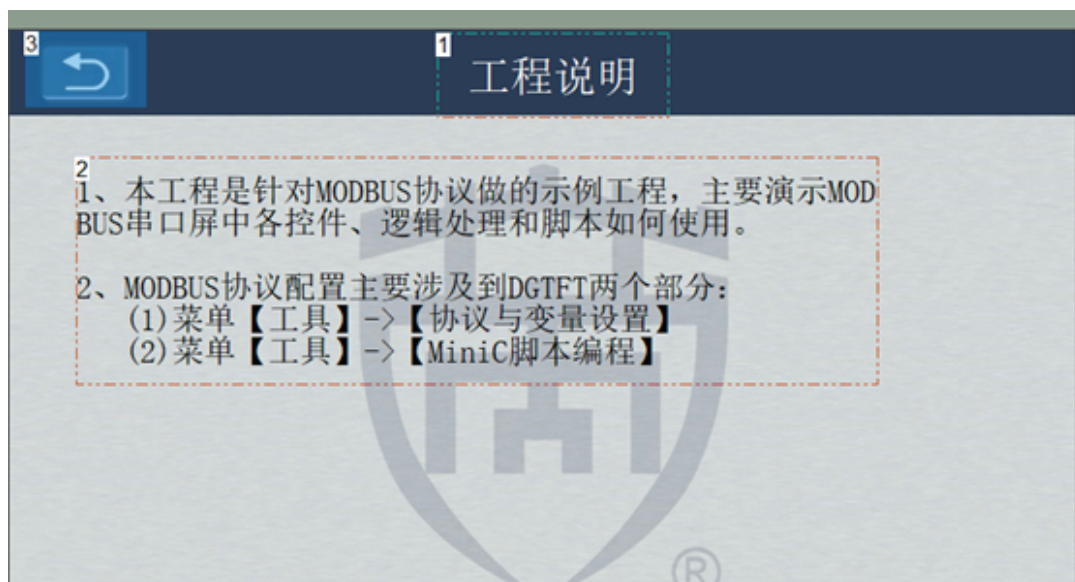


图 4-6 工程说明

4.2.3 线圈

【线圈】画面是对MODBUS协议中“线圈”变量数值的直接显示，文本控件直接绑定“线圈”类型的变量，运行后，文本控件会显示变量的值，当该地址的“线圈”值发生变化后，文本控件同样也会发生变化。

配置方法：

1、添加变量：创建“抽湿开关”变量，添加子节点“开关机”，变量类型设置为“线圈”，配置变量地址为0x0000，如图 4-8所示；

2、画面中添加一个文本控件（ID：2），如图 4-7所示，设置文本控件的属性，设置关联变量“抽湿开关/开关机”，如图 4-9所示；

通过以上两步，ID为2的文本控件与0x0000地址的“抽湿开关/开关变”便建立起了关联。此时，编译工程并运行串口屏，文本控件便显示0x0000地址的线圈的数值。

同理，ID为3、4、5的文本控件通过以上方法分别与不同的变量关联后，便可显示相应地址的变量值。

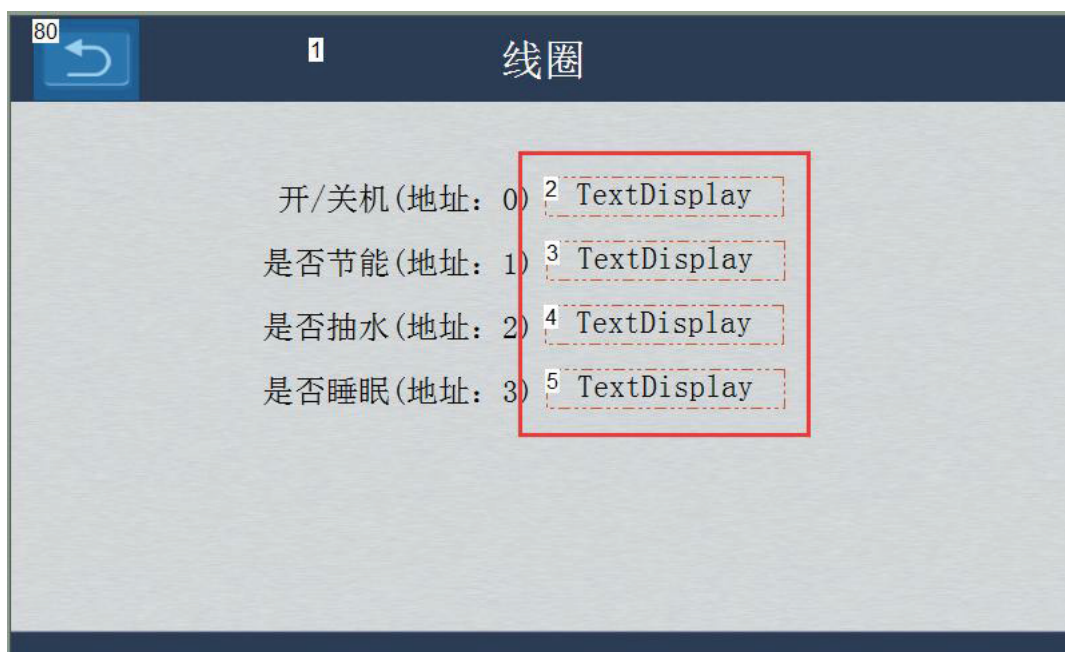


图 4-7 线圈

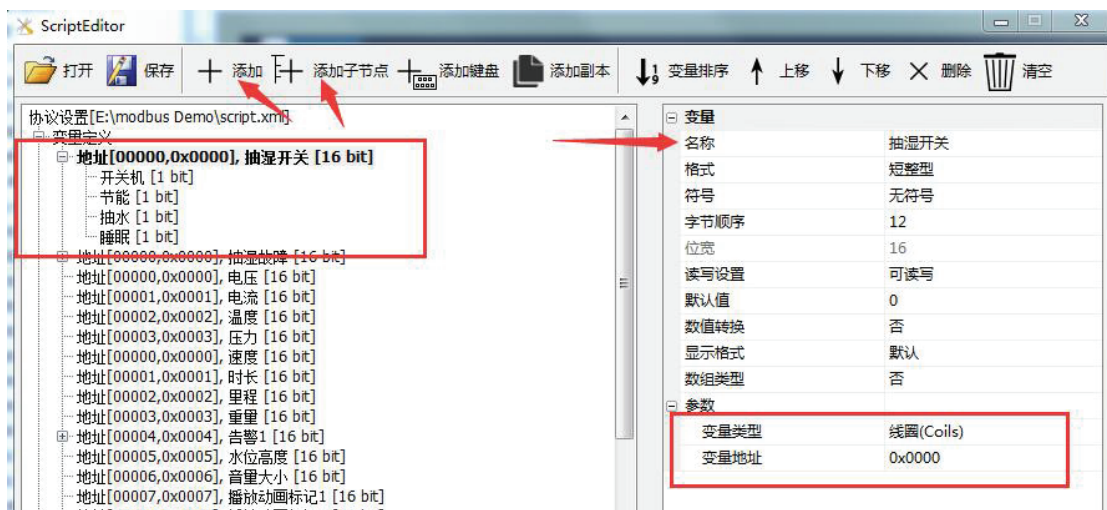


图 4-8 新建线圈变量

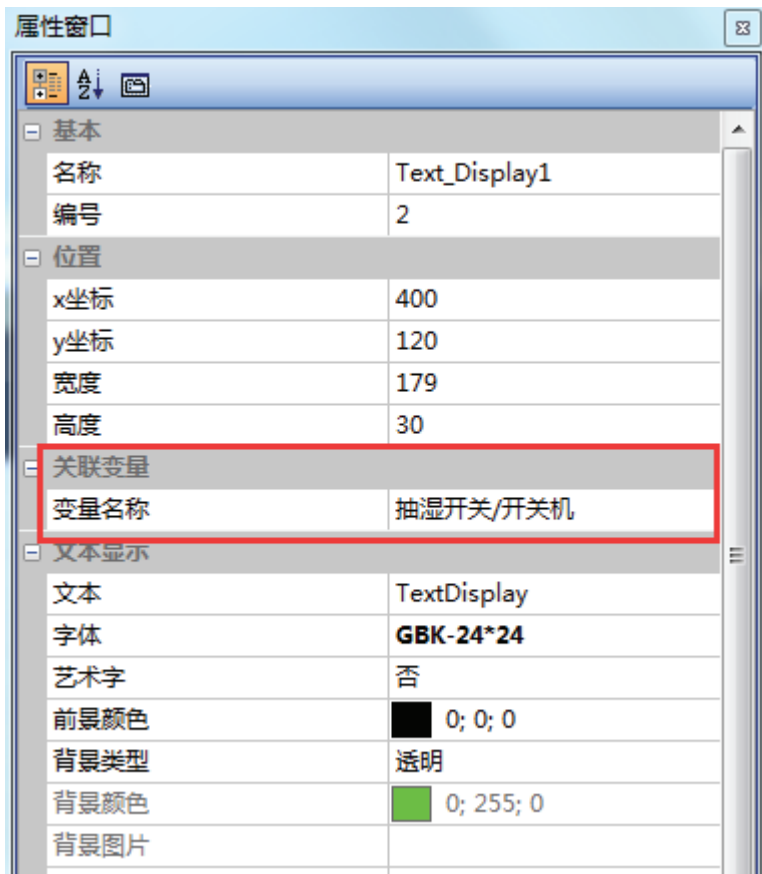


图 4-9 关联变量

4.2.4 离散输入

【离散输入】画面是对MODBUS协议中“离散输入”变量数值的直接显示，文本控件直接绑定“离散输入”类型的变量，运行后，文本控件会显示变量的值，当该地址的“离散输入”值发生变化后，文本控件同样也会发生变化。

方法：

- 1、添加变量：创建“抽湿故障”变量，添加子节点“故障 0”，变量类型设置为“离散

输入”，配置变量地址为0x0000，如图 4-11所示；

2、画面中添加一个文本控件（ID：2），如图 4-10所示，设置文本控件的属性，设置关联变量“抽湿故障/故障0”，如图 4-12所示；

通过以上两步，ID为2的文本控件与0x0000地址的“抽湿故障/故障0”便建立起了关联。此时，编译工程并运行虚拟串口屏，文本控件便显示0x0000地址的离散输入的数值。

同理，ID为3、4、5的文本控件通过以上方法分别与不同的变量关联后，便可显示相应地址的变量值。



图 4-10 离散输入

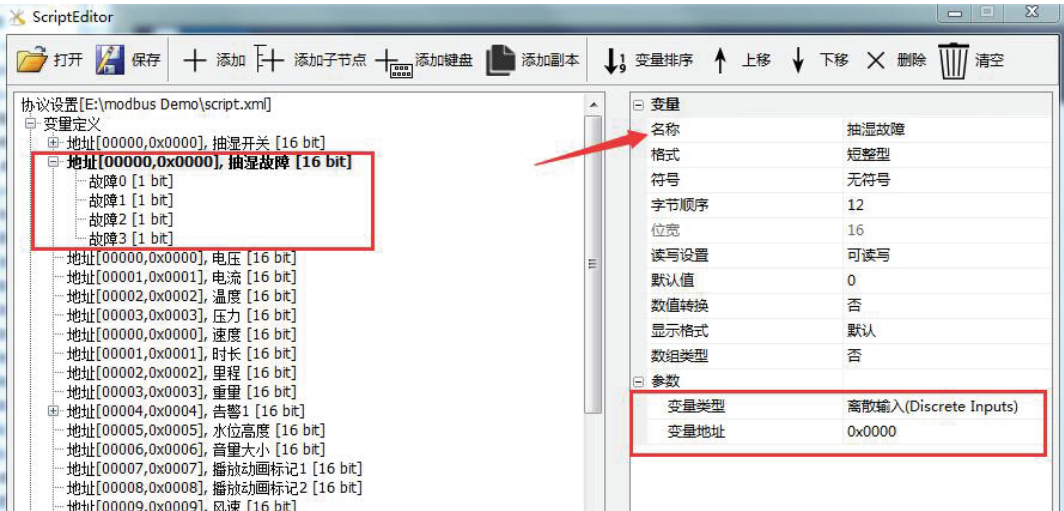


图 4-11 新建离散输入



图 4-12 关联变量

4.2.5 保持寄存器

【保持寄存器】画面是对 MODBUS 协议中“保持寄存器”变量数值的直接显示，文本控件直接绑定“保持寄存器”类型的变量，运行后，文本控件会显示变量的值，当该地址的“保持寄存器”值发生变化后，文本控件同样也会发生变化。

方法：

1、添加变量：创建“速度”变量，变量类型设置为“保持寄存器”，配置变量地址为 0x0000，如图 4-14 所示；

2、画面中添加一个文本控件（ID：2），如图 4-13 所示，设置文本控件的属性，设置关联变量“速度”，如图 4-15 所示；

通过以上两步，ID 为 2 的文本控件与 0x0000 地址的“速度”便建立起了关联。此时，编译工程并运行虚拟串口屏，文本控件便显示 0x0000 地址的保持寄存器的数值。

同理，ID 为 3、4、5 的文本控件通过以上方法分别与不同的变量关联后，便可显示相应地址的变量值。



图 4-13 保持寄存器

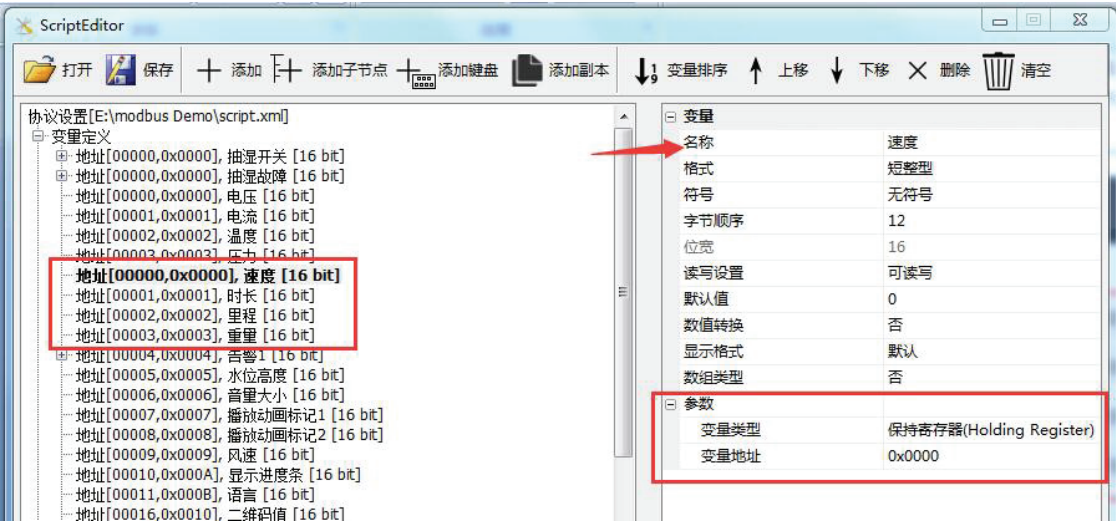


图 4-14 新建保持寄存器



图 4-15 关联变量

4.2.6 输入寄存器

【输入寄存器】画面是对MODBUS协议中“输入寄存器”变量数值的直接显示，文本控件直接绑定“输入寄存器”类型的变量，运行后，文本控件会显示变量的值，当该地址的“输入寄存器”值发生变化后，文本控件同样也会发生变化。

方法：

1、添加变量：创建“电压”变量，变量类型设置为“输入寄存器”，配置变量地址为0x0000，如图 4-17所示；

2、画面中添加一个文本控件（ID：2），如图 4-16所示，设置文本控件的属性，设置关联变量“电压”，如图 4-18所示；

通过以上两步，ID为2的文本控件与0x0000地址的“电压”便建立起了关联。此时，编译工程并运行虚拟串口屏，文本控件便显示0x0000地址的保持寄存器的数值。

同理，ID为3、4、5的文本控件通过以上方法分别与不同的变量关联后，便可显示相应地址的变量值。



图 4-16 输入寄存器

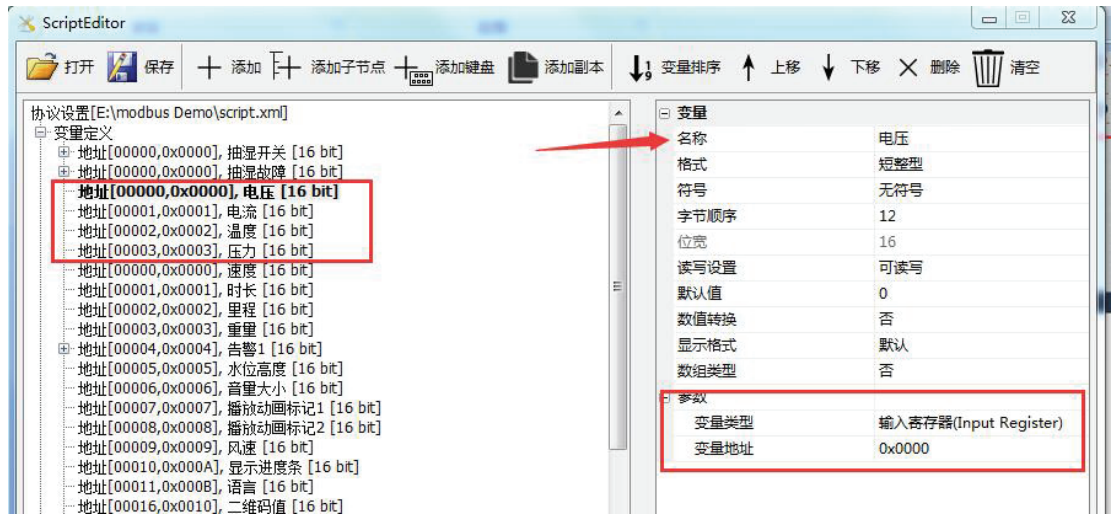


图 4-17 新建输入寄存器

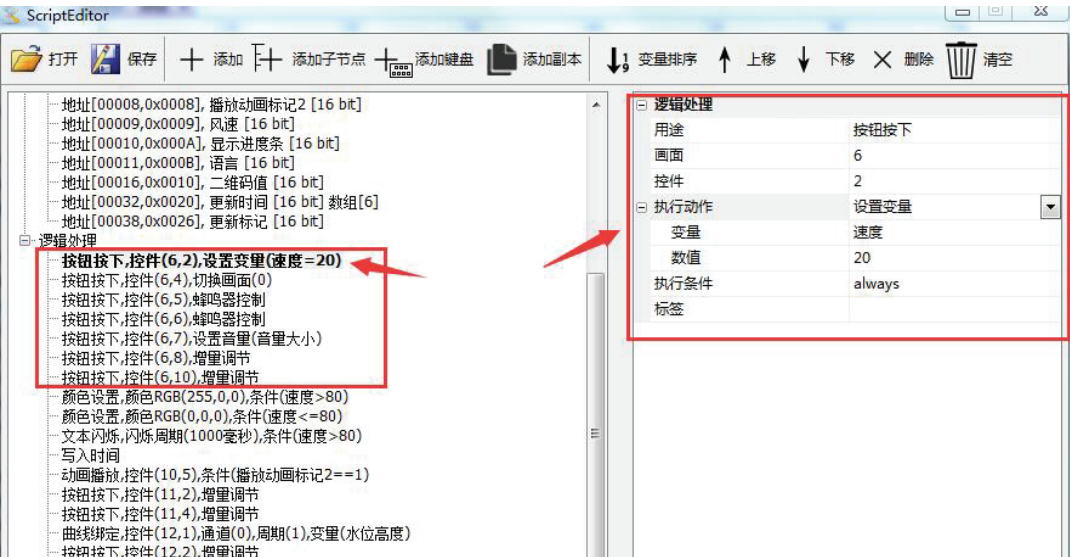


图 4-20 按钮逻辑处理

4.2.8 文本控件

【文本控件】画面主要对文本控件的使用做示例说明。MODBUS协议版本中，文本控件主要是用于显示绑定的变量值，对于线圈、保持寄存器等类型的可修改变量，可以设置“输入方式”为弹出系统键盘输入，通过弹出的键盘对绑定的变量值进行修改。另外可以绑定修改显示颜色、闪烁的逻辑处理。

【文本控件】画面ID为2的文本控件，如图 4-21所示，在其属性中绑定变量“速度”，如图4-22所示；在【协议与变量设置】中添加3条逻辑处理：第一条，用途为“设置颜色”，颜色设置为红色（RGB（255,0,0）），执行条件为“速度”“大于”“80”；第二条，用途为“设置颜色”，颜色设置为黑色（RGB（0,0,0）），执行条件为“速度”“小于等于”“80”；第三条，用途为“文本闪烁”，执行条件为“速度”“大于”“80”；如图 4-23所示。

此时，编译工程并运行虚拟串口屏，当“速度”的值大于80时，文本显示为红色并闪烁，如果“速度”的值小于等于80时，文本显示为黑色，并且不闪烁。

同样，ID为3、4、5的文本控件均可做此操作。

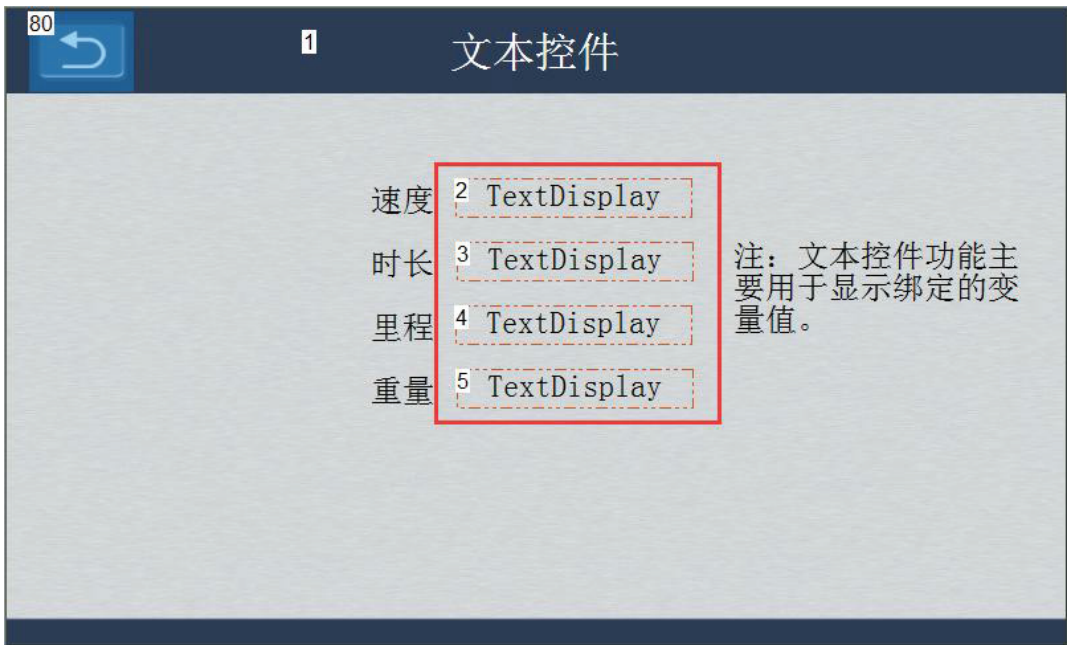


图 4-21 文本控件

【进度条仪表滑块】画面中创建的文本控件、进度条、滑块、仪表控件，如图 4-24 所示；这几个控件均绑定变量“速度”，如图 4-25所示，为进度条关联变量，其它几个控件类似；其中文本控件主要用于观察值的大小，也可以通过其弹出的键盘修改对应值。

此时，编译工程并运行虚拟串口屏，当“速度”的值发生变化时，文本控件、进度条、滑块、仪表均都会发生变化，更新显示的速度值；另外，当拖动滑块时，其它控件的也会更新显示速度值。

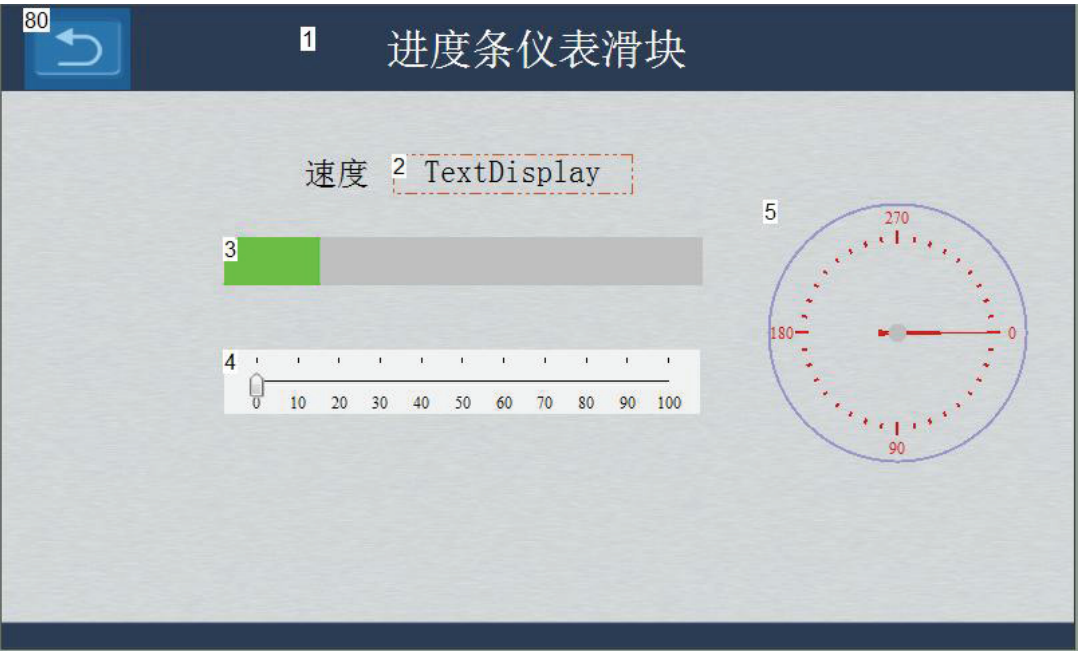


图 4-24 进度条、仪表、滑块

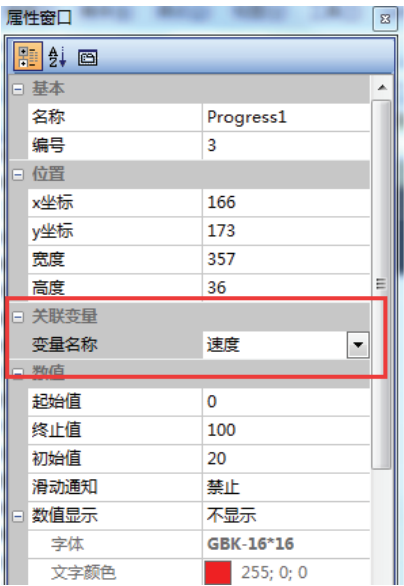


图 4-25 关联变量

4.2.10 RTC控件

【RTC控件】画面主要对RTC控件的使用及更新RTC时间做示例说明，如图 4-26所示。RTC控件支持不同的属性格式，如所示图 4-27；也可以用户自定义显示格式，如图 4-28所示。

MODBUS协议版本串口屏中，更新RTC时间常见有两种应用：

1、设置本机RTC时间

通常有两种方法：1、直接设置RTC控件属性“允许编辑”为“编辑日期和时间”、“编辑日期”或者“编辑时间”；如此，当长按RTC控件或着点击RTC控件时即可弹出键盘以修改RTC时间。2、使用set_date()和set_time()设置RTC时间，具体可以参照图 4-26中，设置本机RTC时间的说明和图 4-30的MiniC代码。

2、屏做主机时，修改从机RTC时间；

通常做法是：从机创建固定地址的保持寄存器，用于存放时间（连续7个保持寄存器，分别存放年、月、星期、日、时、分、秒），主机周期性的把最新的时间发送给从机。逻辑处理中有“写入时间”的功能，可以实现此操作。如图 4-29中红色框所示，建立“写入时间”的逻辑处理，具体设置如图中所示，其中“地址”为从机中保存时间的地址，“周期”为主机每隔多长时间写入一次；可以根据自己需求建立条件执行。

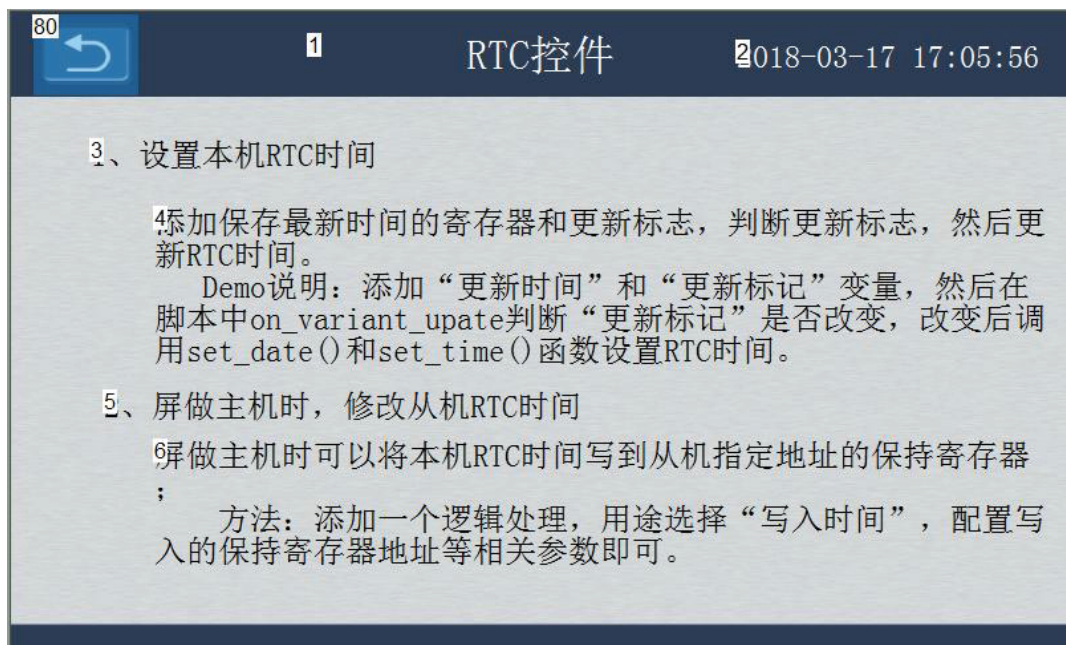


图 4-26 RTC控件

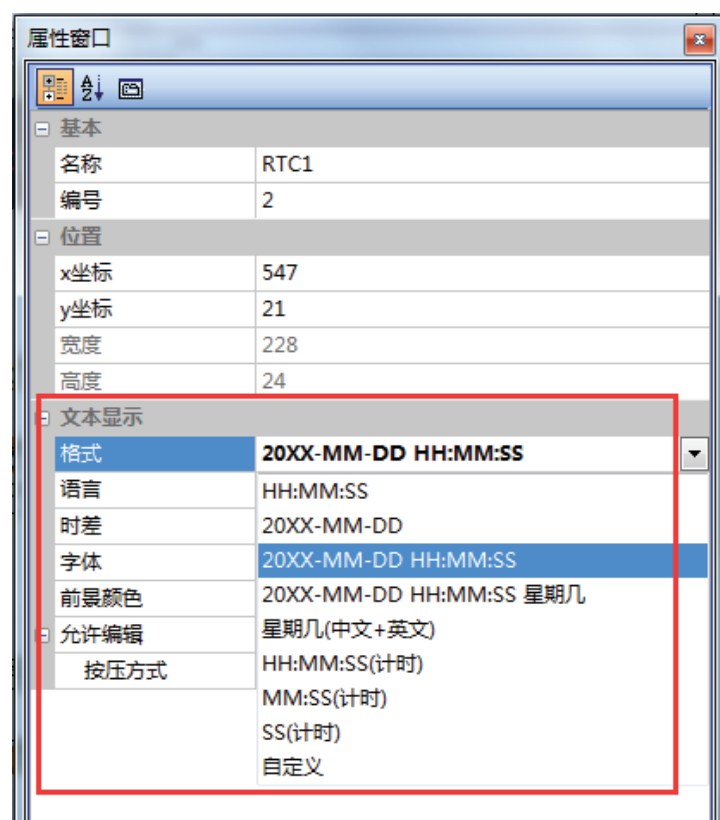


图 4-27 RTC控件属性



图 4-28 RTC自定义

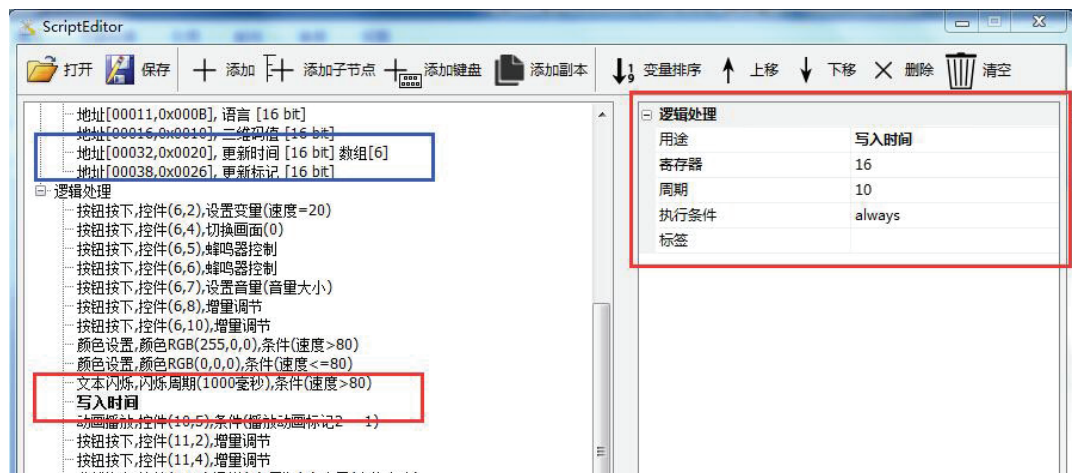


图 4-29 更新RTC的寄存器、逻辑处理

```

47 /*
48 函数: on_variant_update
49 功能: 串口通信导致变量更新时, 执行此函数
50 */
51 void on_variant_update()
52 {
53     //操作符'@'用于判定某个寄存器是否发生改变
54     //更新RTC时间
55     if (@ "更新标记")
56     {
57         set_date("更新时间"[0], "更新时间"[1], "更新时间"[2]);
58         set_time("更新时间"[3], "更新时间"[4], "更新时间"[5]);
59     }
60
61     //“语言”发生变化时, 修改设置系统语言
62     if (@ "语言")
63     {
64         sys.lang = "语言";
65     }
66 }
67
68

```

图 4-30 更改本机RTC时间

4.2.11 动画控件

【动画控件】画面主要对动画控件的使用做示例说明, 如图 4-31所示。动画控件用于播放设置的GIF或ICON格式的动画文件, 可以控制暂停或者播放。

MODBUS协议中有两种控制动画控件的方式:

1、绑定变量方式

动画控件直接关联变量, 当变量的值为0时, 动画停止; 当变量的值为1时, 动画播放。

【动画控件】画面中的ID为2的动画控件, 设置关联变量“播放动画标记1”, 如图 4-32所示。此时, 编译工程并运行虚拟串口屏, 当变量“播放动画标记1”为0时动画停止, 当变量值为1时, 动画开始播放。

2、逻辑处理方式

逻辑处理中有“动画播放”功能, 当预设的条件满足时, 可以控制动画播放或停止。

【动画控件】画面中的ID为5的动画控件，创建了“动画播放”的逻辑处理，如图4-33所示，设置的执行条件为变量“播放动画标记2”“==”“1”时，ID为5的动画会自动播放。

此时，编译工程并运行虚拟串口屏，当变量“播放动画标记2”为0时动画停止，当变量值为1时，动画开始播放。

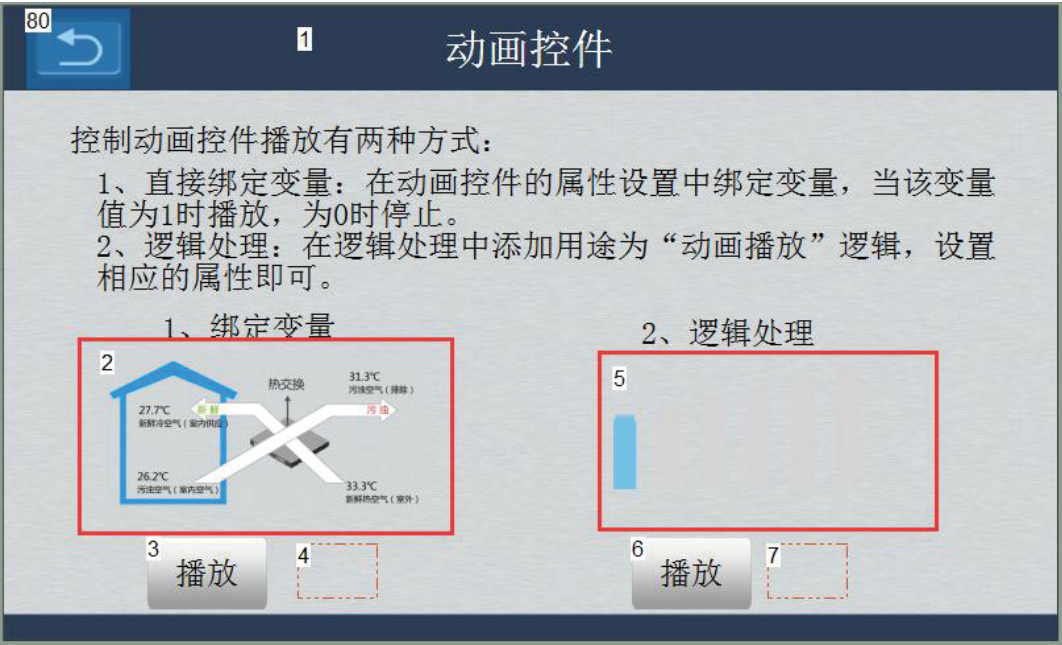


图 4-31 动画控件

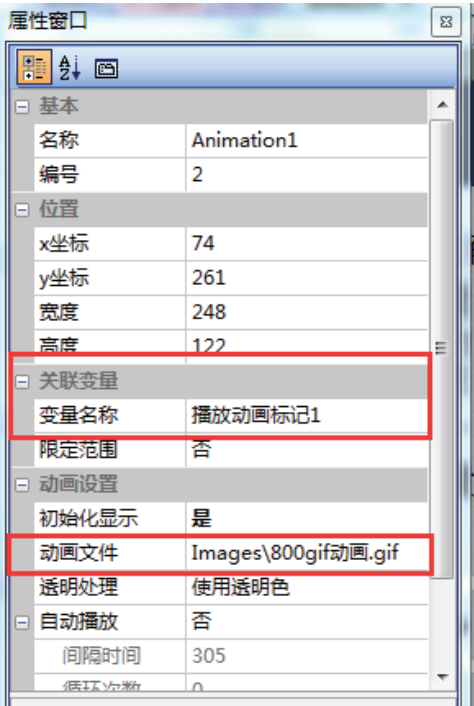


图 4-32 关联变量

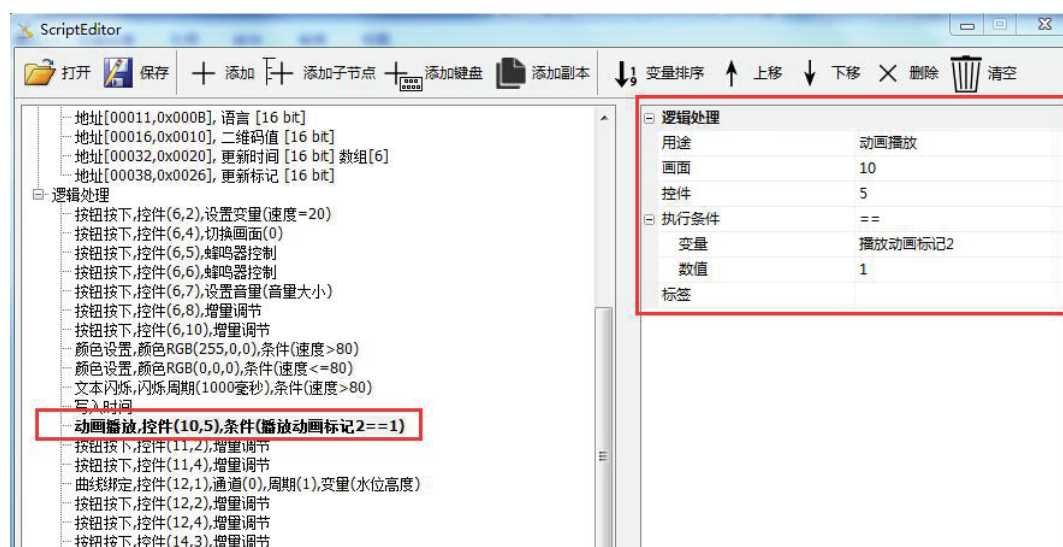


图 4-33 动画逻辑处理

4.2.12 图标控件

【图标控件】画面主要对图标控件的使用做示例说明，如图 4-34 所示。图标控件用于显示设置的 GIF 或 ICON 格式的动画文件，通过显示不同的图片表示不同的状态，可以控制显示第几帧图片。

MODBUS协议中可以通过关联变量的方式控制显示第几帧图片，关联变量的变量值默认对应图标帧，索引从0开始，值为0时显示第0帧，值为1时显示第1帧，以此类推；如果设置了限定范围，则变量下限对应设置的图标下限，变量上限对应图标上限。

举例：【图标控件】画面中ID为1的图标控件关联了变量“风速”，如图 4-35所示。此时，编译工程并运行虚拟串口屏，当变量“风速”为0时，图标控件中显示的风速为1格；当变量“风速”为1时，图标控件中显示的风速为2格；当变量“风速”为2时，图标控件中显示的风速为3格；以此类推。

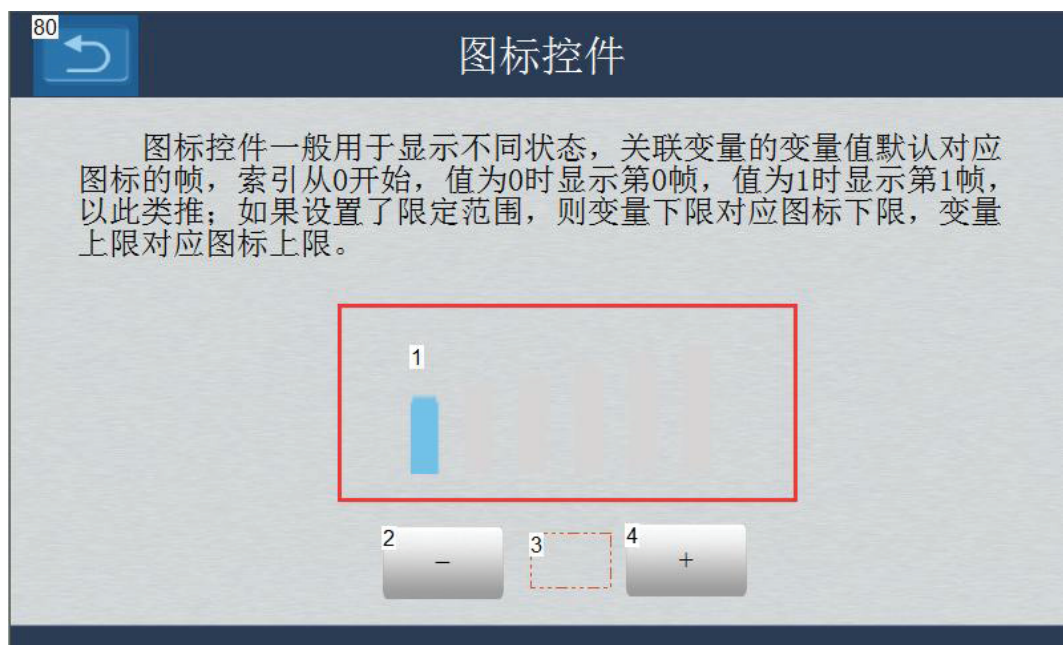


图 4-34 图标控件



图 4-35 关联变量

4.2.13 曲线控件

【曲线控件】画面主要对曲线控件的使用做示例说明，如图 4-36所示。MODBUS 协议版本中，曲线控件用于显示关联的变量值，曲线控件周期性的采样变量值，然后将所有采样值绘制到曲线控件内，连接形成曲线；曲线控件的每个通道都可以关联一个变量，最多可以设置8个通道。

MODBUS协议版本串口屏中，曲线通道显示变量值，通过设置“曲线绑定”用途的逻辑处理使用。

【曲线控件】画面中ID为1的曲线控件，预设一个通道，为该控件的通道0，添加关联的逻辑处理，参数如图 4-37所示，每“1”秒采样变量一次“水位高度”的值。画面ID为2、4的按钮用于调整“水位高度”值，ID为3的文本控件实时显示“水位高度”值；经过动态的调整“水位高度”，可以形成如图 4-38所示的曲线控件图。

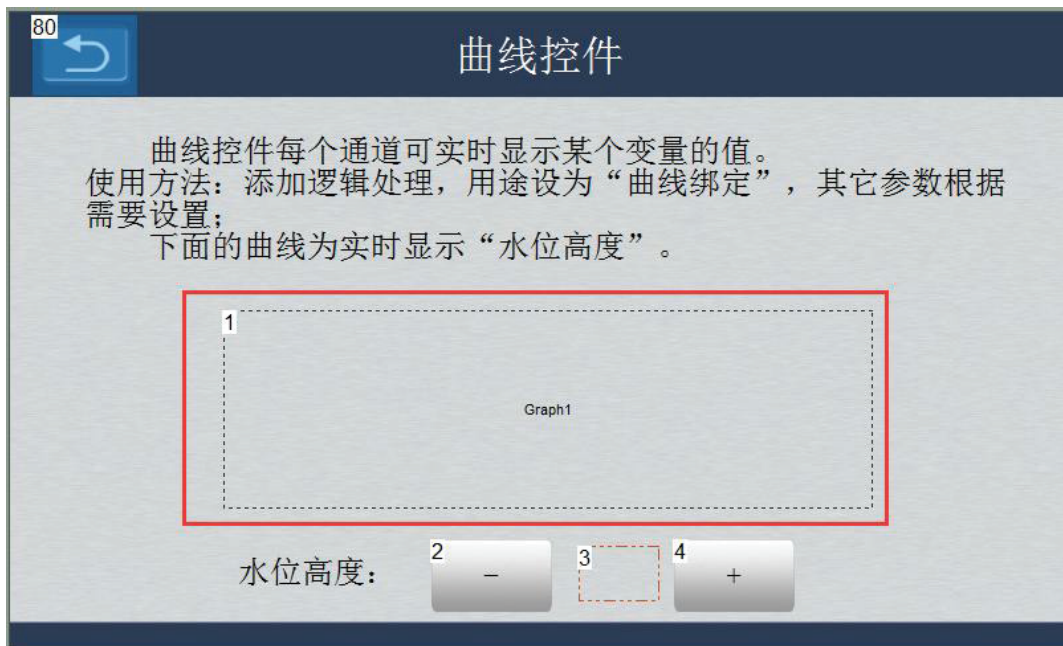


图 4-36 曲线控件

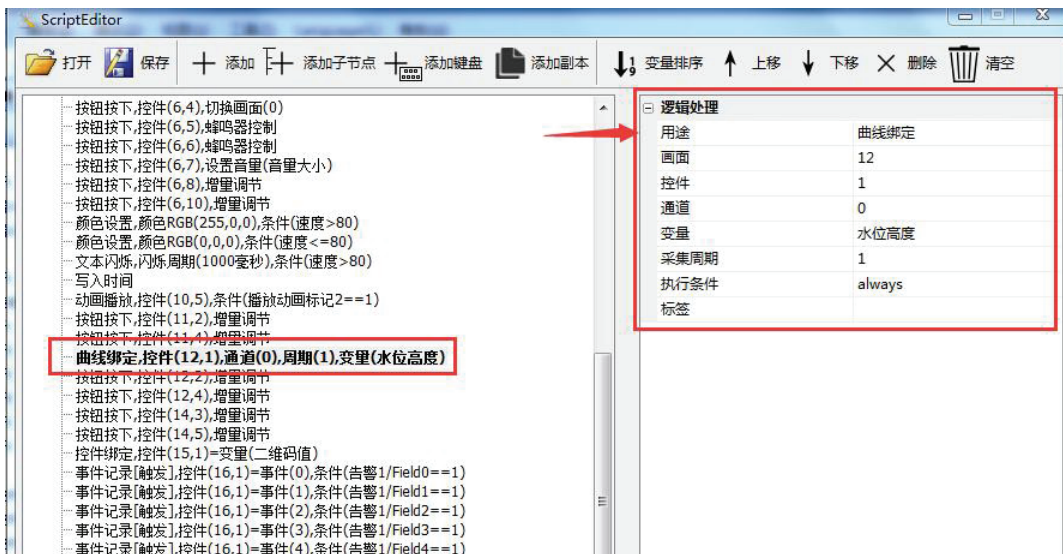


图 4-37 曲线绑定

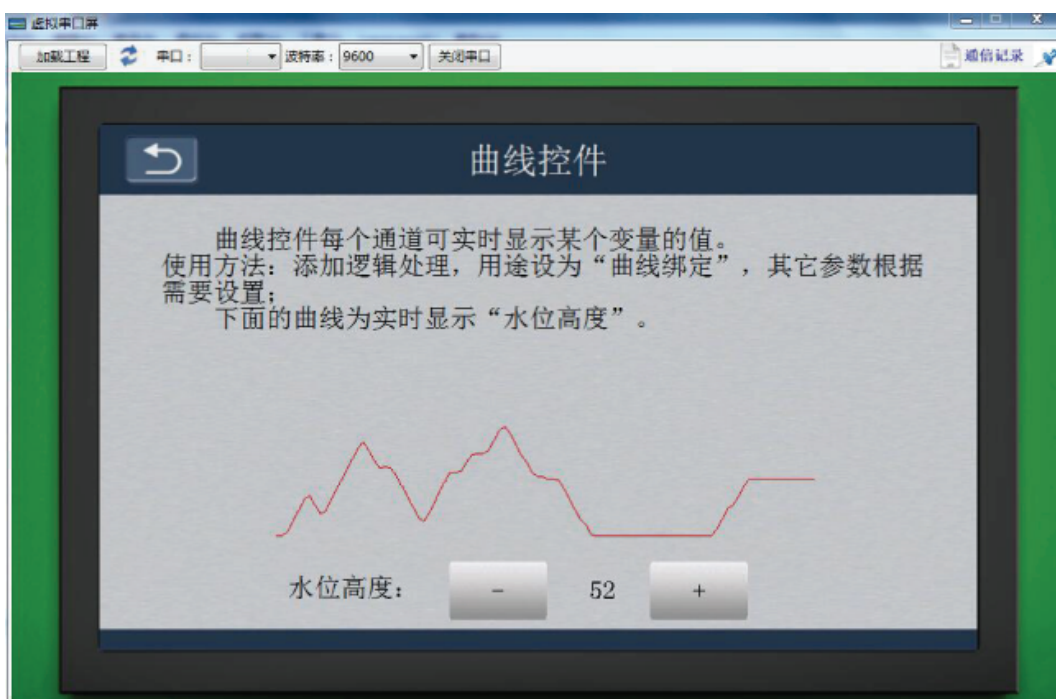


图 4-38 曲线控件运行示例

4.2.14 菜单控件

【菜单控件】画面主要对菜单控件的使用做示例说明，菜单的“菜单风格”有两种：“弹出菜单”和“固定菜单”，此处主要做弹出菜单使用说明。

“弹出菜单”需要按钮控件将菜单和文本控件关联使用，点击按钮弹出菜单，选中菜单项后直接显示在文本控件中。

【菜单控件】画面如图 4-39 所示，创建 ID 为 1 的文本控件，ID 为 2 的按钮控件和 ID 为 3 的菜单控件；菜单控件的属性设置如图 4-40 所示，关联变量“风速”，菜单风格为“弹出菜单”，菜单选项为“一级；二级；三级；四级；五级；六级”，索引分别为 0、1、2、3、4、5；按钮设置如图 4-41 所示，触控用途设置为“弹出菜单”，“菜单控件 ID”设置为“3”，“输入控件 ID”设置为“1”；此时，编译工程并运行虚拟串口屏，当点击文本控件时，会弹出菜单，当点击菜单中选项后，文本控件的内容变为选中的菜单选项，同时，变量“风速”也被设置为选项对应的索引值；如果不设置变量，则只有文本控件的值发生变化。

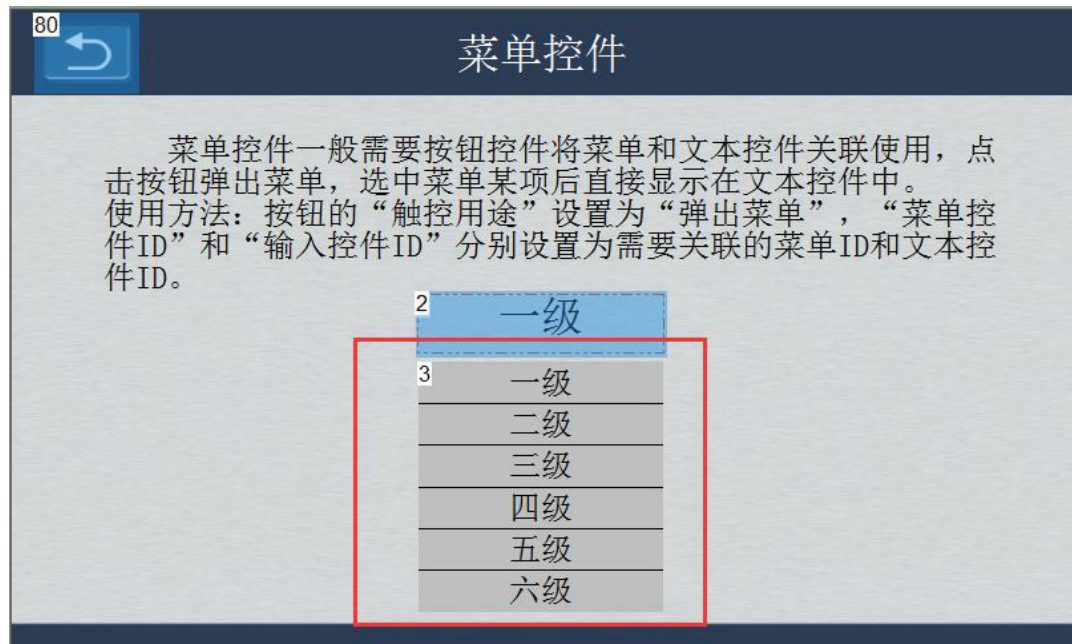


图 4-39 菜单控件



图 4-40 菜单设置

弹出时的图片	
裁剪	
按下时的图片	
裁剪	
文字状态	否
触控用途	弹出菜单
菜单控件ID	3
输入控件ID	1

图 4-41 按钮配置

4.2.15 选择控件

【选择控件】画面主要对选择控件的使用做示例说明。选择控件主要有两种用途：滑动选择和状态显示。“滑动选择”主要用于通过触摸屏滑动选择预设的选项；“状态显示”主要用于显示不同的预设选项，类似图标控件。

【选择控件】画面中ID为1的选择控件为“滑动选择”方式，ID为2的选择控件为“状态显示”方式。

1、滑动选择

ID为1的选择控件，属性设置如图 4-43所示，关联变量“风速”，用途设置为“滑动选择”，设置两端缩放和候选项，设置预设的数据选项“一级；二级；三级；四级；五级；六级；”；此时，编译工程并运行虚拟串口屏，可以通过鼠标点中选择控件并拖动（模拟滑动选择），当拖动停止后，选中某个选项，同时变量“风速”的值也会变化更新。

2、状态选择

ID为2的选择控件，属性设置如图 4-44所示，关联变量“风速”，用途设置为“状态显示”，设置预设的数据选项“一级；二级；三级；四级；五级；六级；”；此时，编译工程并运行虚拟串口屏，当变量“风速”的值发生变化时，控件2显示的状态值也会相应的更新。

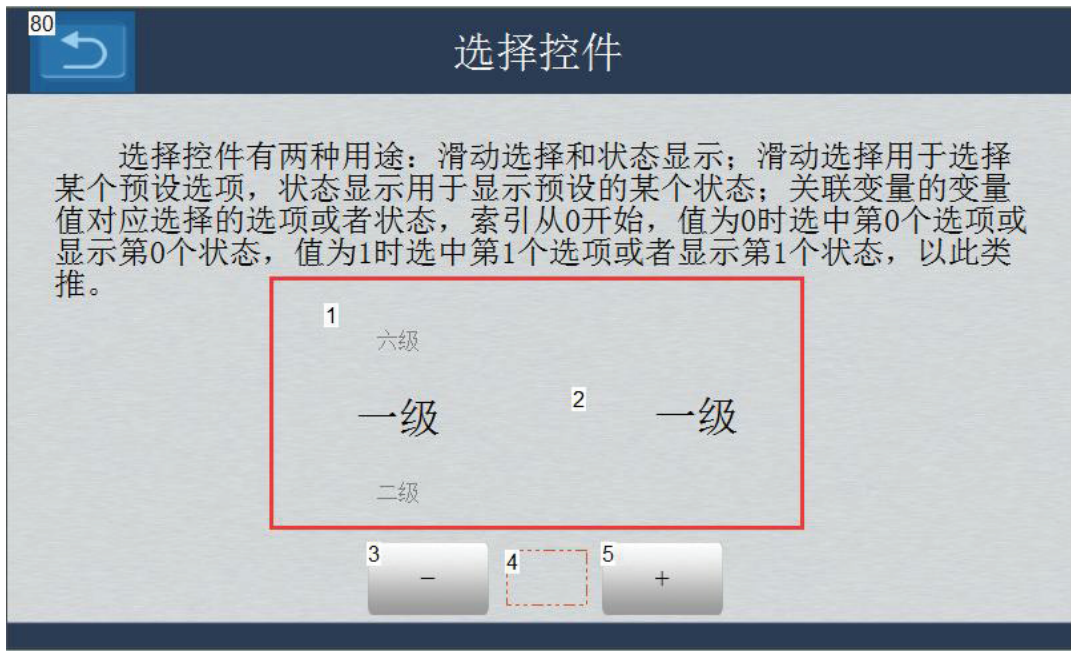


图 4-42 选择控件

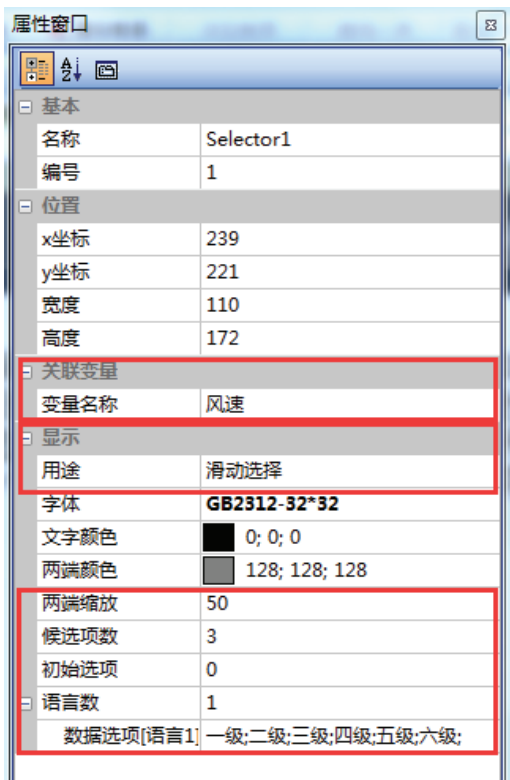


图 4-43 滑动选择



图 4-44 状态显示

4.2.16 二维码

【二维码】画面主要对二维码控件的使用做示例说明。二维码控件可以将设定的字符串以二维码的形式显示出来。MODBUS协议版本的串口屏，二维码控件显示关联的字符串类型变量。注意，二维码控件需要特殊的固件支持。

【二维码】画面中ID为1的二维码控件属性具体设置如图 4-46所示；“文本”即为默认显示的字符串，“放大系数”即为单个点的大小，“错觉纠正”该级别越高，可以显示的字符串长度越小，“编码方式”即二维码的编码方式。

创建“二维码值”变量，变量类型设置为“字符串类型”，具体设置如图 4-47所示，创建“控件绑定”的逻辑处理，将控件与变量“二维码值”相关联。

此时，编译工程并运行虚拟串口屏，二维码控件将显示绑定变量“二维码值”的二维码，当变量“二维码值”发生变化时，二维码控件会自动更新，生成新的二维码。

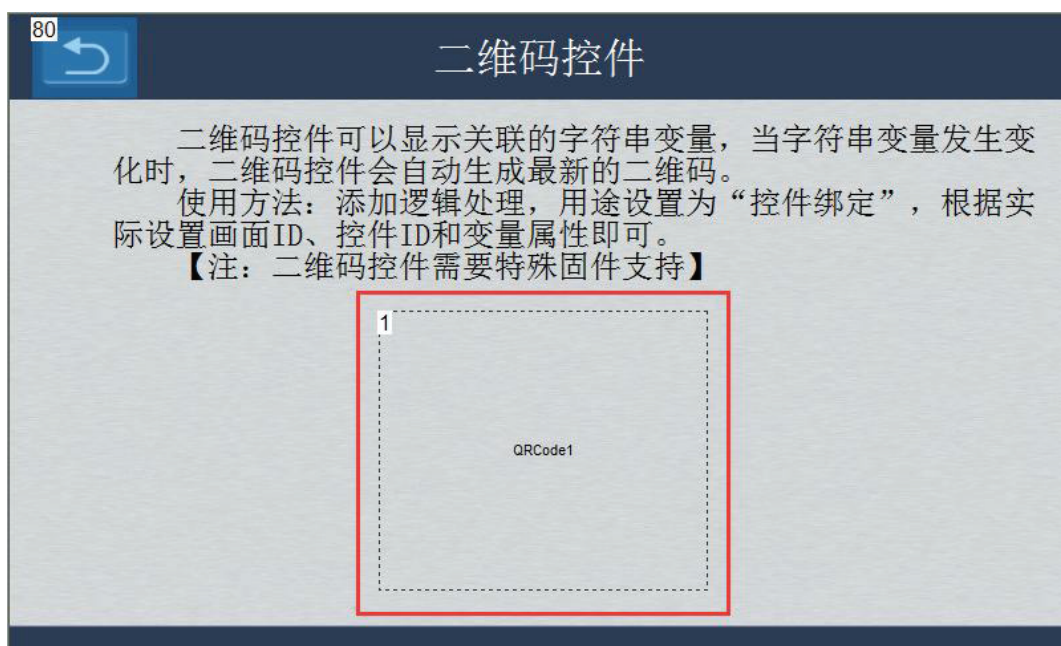


图 4-45 二维码控件

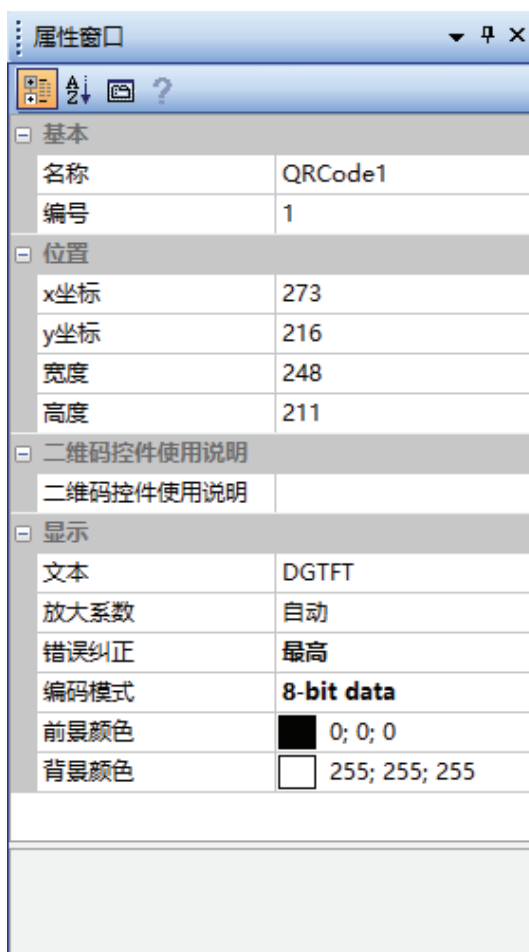


图 4-46 二维码属性

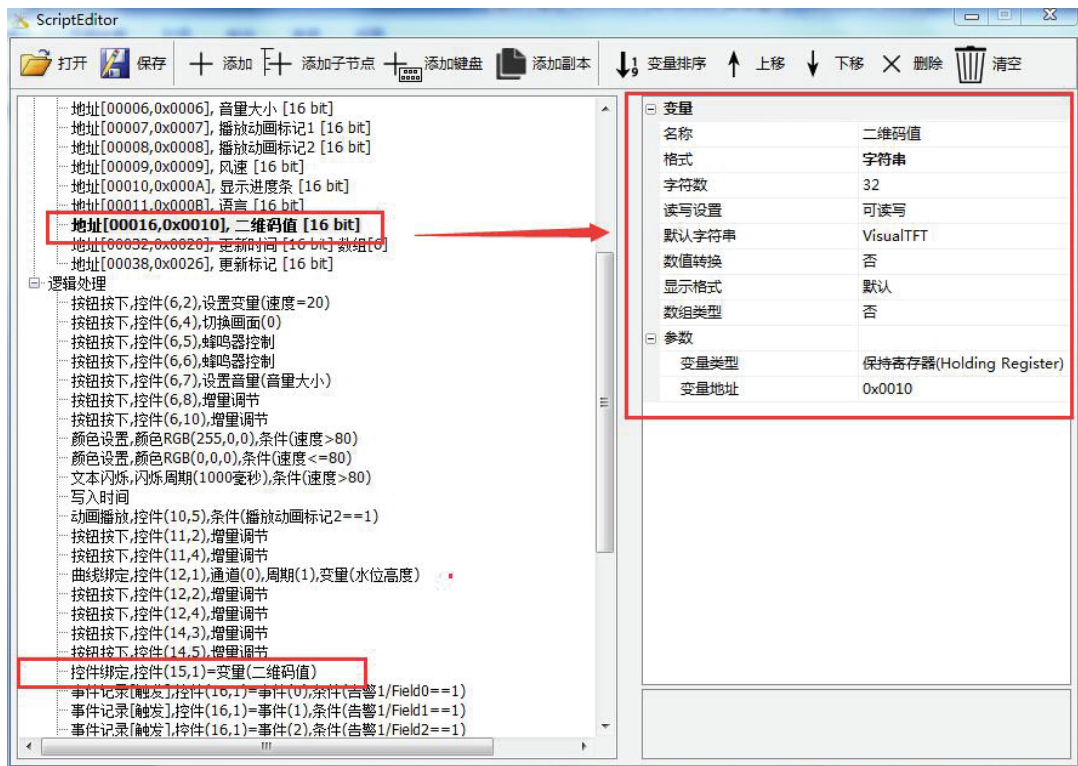


图 4-47 二维码变量及控件绑定

4.2.17 数据记录控件

【数据记录控件-当前告警】和【数据记录控件-历史告警】画面主要对数据记录控件的使用做示例说明。

数据记录控件在MODBUS协议版本的串口屏中常用于三种“记录类型”：“当前告警”、“历史告警”和“历史告警与解除”，其中“历史告警”与“历史告警与解除”的基本相同，“历史告警与解除”相比“历史告警”，每条告警记录增加了告警的解除时间。本例只对“当前告警”和“历史告警”做示例说明。

数据记录控件的记录类型作为“当前告警”和“历史告警”的使用设置过程相同，步骤如下：

- 1、设置数据记录控件的“记录类型”为“当前告警”或“历史告警”，如图 4-50所示；
- 2、在“告警解析”中设置可能的事件ID数值、颜色及描述信息，如图 4-51所示；
- 3、针对每个事件ID添加用途为“事件记录[触发]”的逻辑处理，如图 4-52所示；每个逻辑处理中根据需要设置“执行条件”；本例中创建了一个保持寄存器类型的变量“告警1”，再对“告警1”的每个位建立“Field0”、“Field1”、“Field2”子节点，判断“Field”，是否为1，作为是否执行的条件。

按照如上步骤配置完成“当前告警”和“历史告警”，此时，编译工程并运行虚拟串口屏。当先后设置“告警1”的值为3（Filed0 = 1, Field1 = 1, Field2 = 0）、4（Filed0 = 0, Field1 = 0, Field2 = 1）后，当前告警显示如图 4-53所示，历史告警如图 4-54所示。

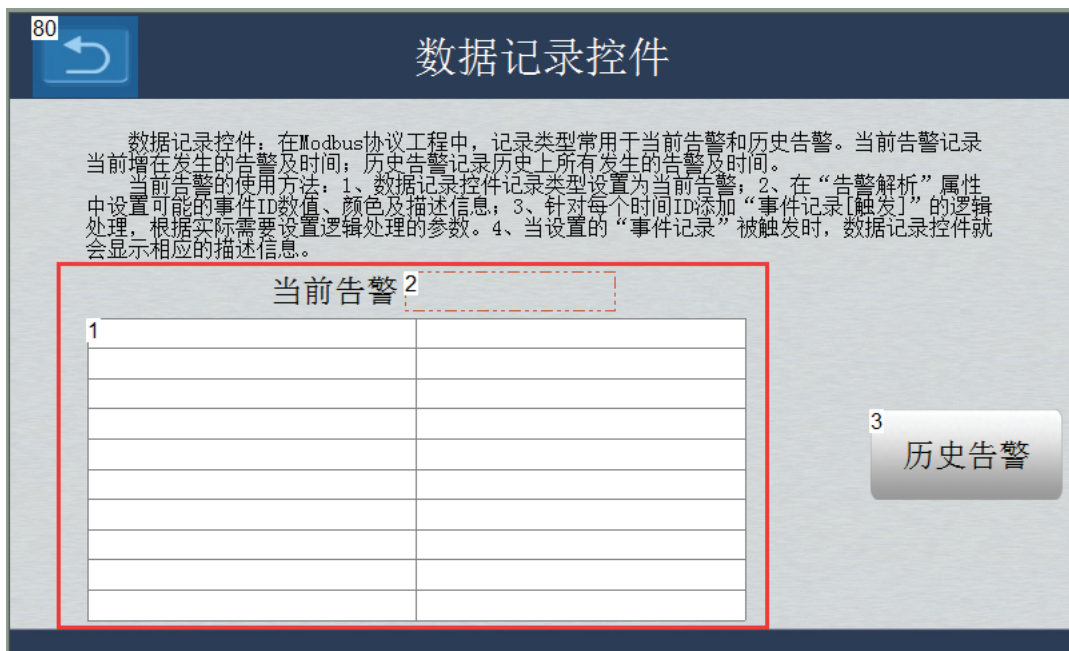


图 4-48 数据记录控件-当前告警

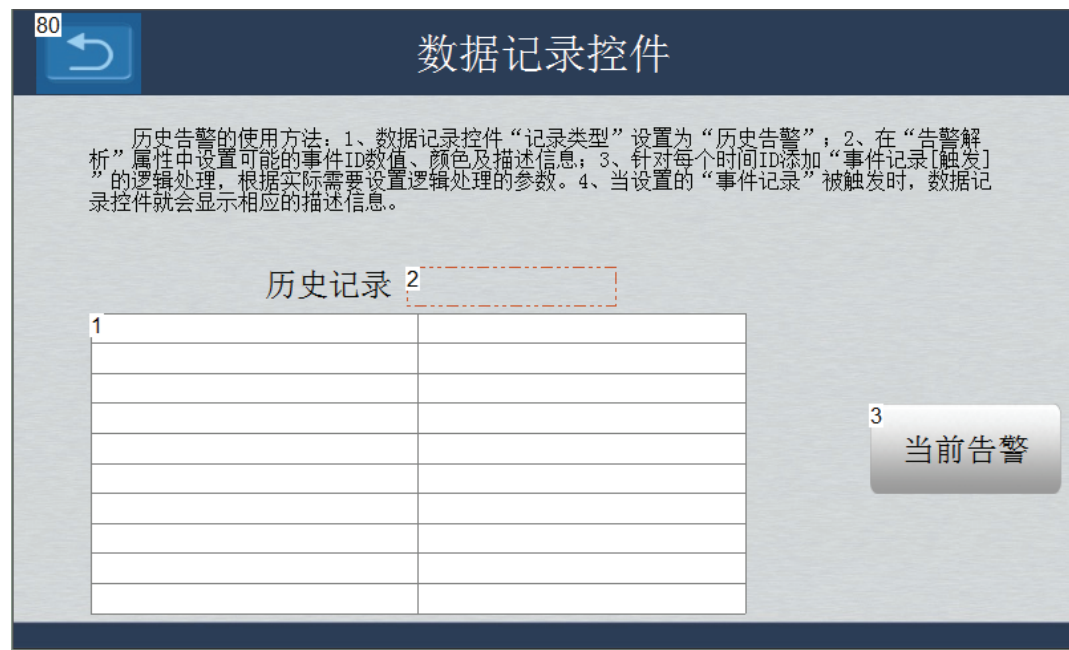


图 4-49 数据记录控件-历史告警

显示网格	是
网格颜色	128; 128; 128
显示背景	是
背景颜色	255; 255; 255
允许选择	否
显示滚动条	显示
显示序号	否
记录设置	
记录类型	当前告警
显示顺序	时间逆序
对齐方式	左对齐
子项数目	2
子项宽度	50;50;
显示记录数	10
最大记录数	1024
单条记录大小	6
是否存储	否
告警解析	设置
记录类型 当前告警：记录当前告警及触发时间；历史告警：记录历史告警以及触发时间；数据表格：存储分号隔开的数	

图 4-50 数据记录控件属性设置

允许选择	否
显示滚动条	显示
显示序号	否
记录设置	
记录类型	当前告警
显示顺序	时间逆序
对齐方式	左对齐
子项数目	2
子项宽度	50;50;
显示记录数	10
最大记录数	1024
单条记录大小	6
是否存储	否
告警解析	设置
告警解析 告警ID转换为有意义的描述字符串	

事件解析设置

数值	前景颜色	描述(多语言时用分号;隔开)
0	(255,0,0)	0: (一般) 送风高湿
1	(255,0,0)	1: (一般) 送风低湿
2	(255,0,0)	2: (一般) 风压传感器故障
3	(255,0,0)	3: (一般) 进水高温
4	(255,0,0)	4: (一般) 水阀开度高
5	(255,0,0)	5: (一般) 远程高温

添加

删除

清空

上移

下移

确定

取消

图 4-51 告警解析设置

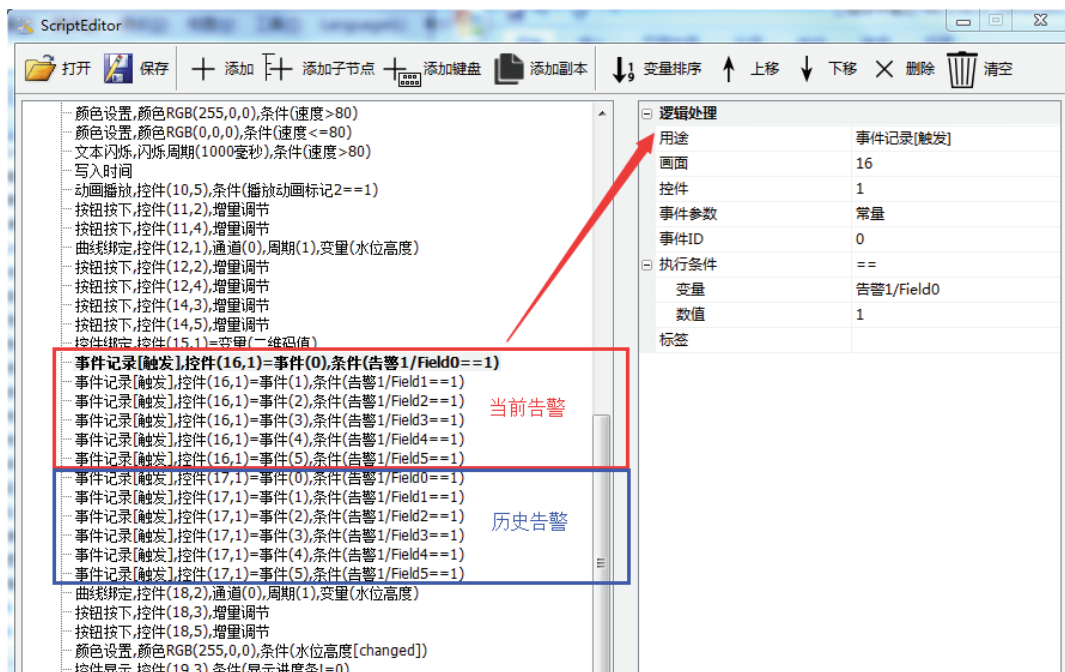


图 4-52 当前告警的逻辑处理

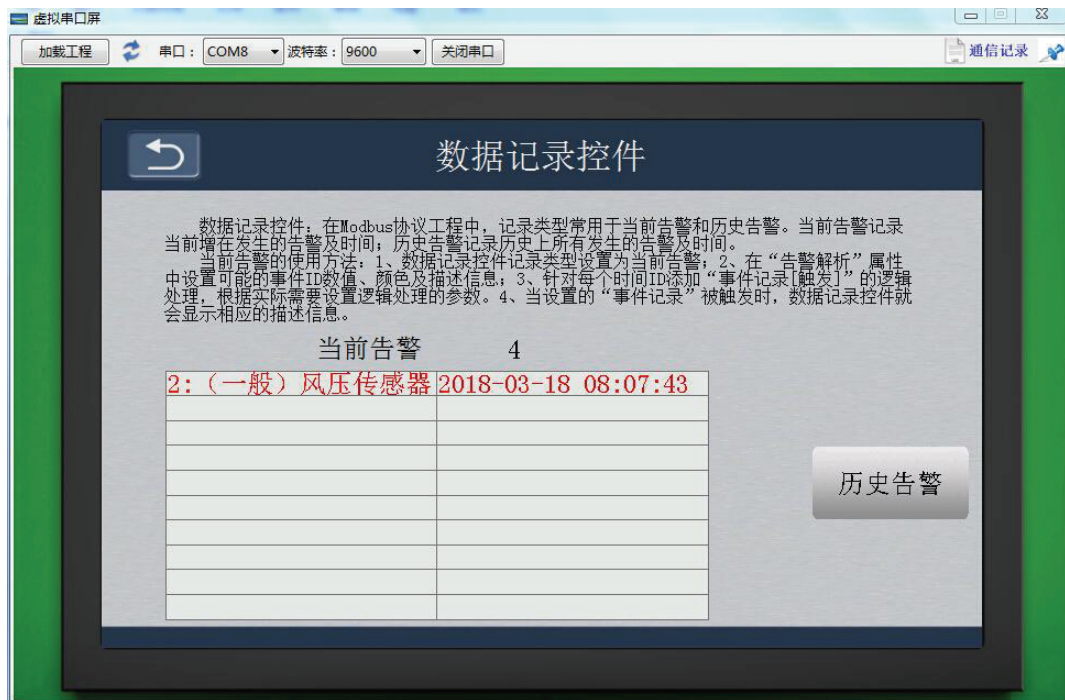


图 4-53 当前告警

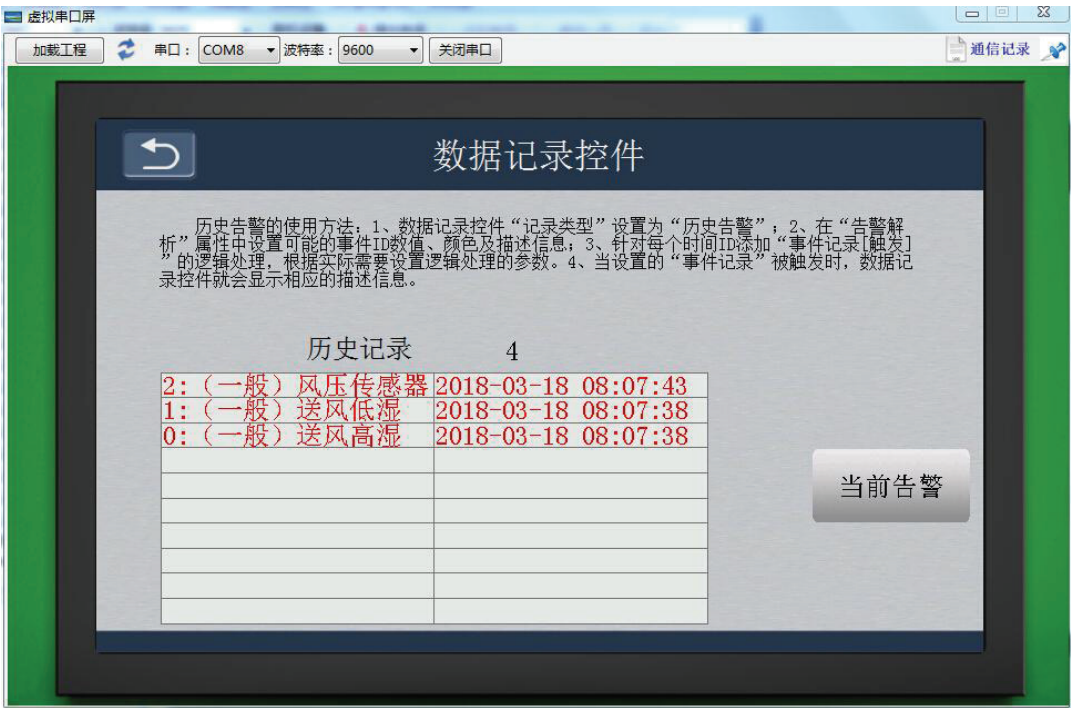


图 4-54 历史告警

4.2.18 历史曲线控件

【历史曲线控件】画面主要对历史曲线控件的使用做示例说明。历史曲线控件和曲线控件都可以实时采样、显示绑定的变量值，不同处是历史曲线控件保存前期的数据，可以查看历史数据。MODBUS协议版本中，历史曲线控件用于显示关联的变量值，历史曲线控件周期性的采样变量值，然后将所有采样值绘制到历史曲线控件内，连接形成曲线；历史曲线控件的每个通道都可以关联一个变量，最多可以设置8个通道。通过设置“曲线绑定”用途的逻辑处理绑定显示。

【历史曲线控件】画面中ID为2的历史曲线控件，预设一个通道，为该控件的通道0，添加关联的逻辑处理，参数如图 4-56所示，每“1”秒采样变量一次“水位高度”的值。画面ID为3、5的按钮用于调整“水位高度”值，ID为4的文本控件实时显示“水位高度”值；经过动态的调整“水位高度”，可以形成如图 4-57所示的曲线控件图，同时可以通过滚动条拖动查看历史的数据。

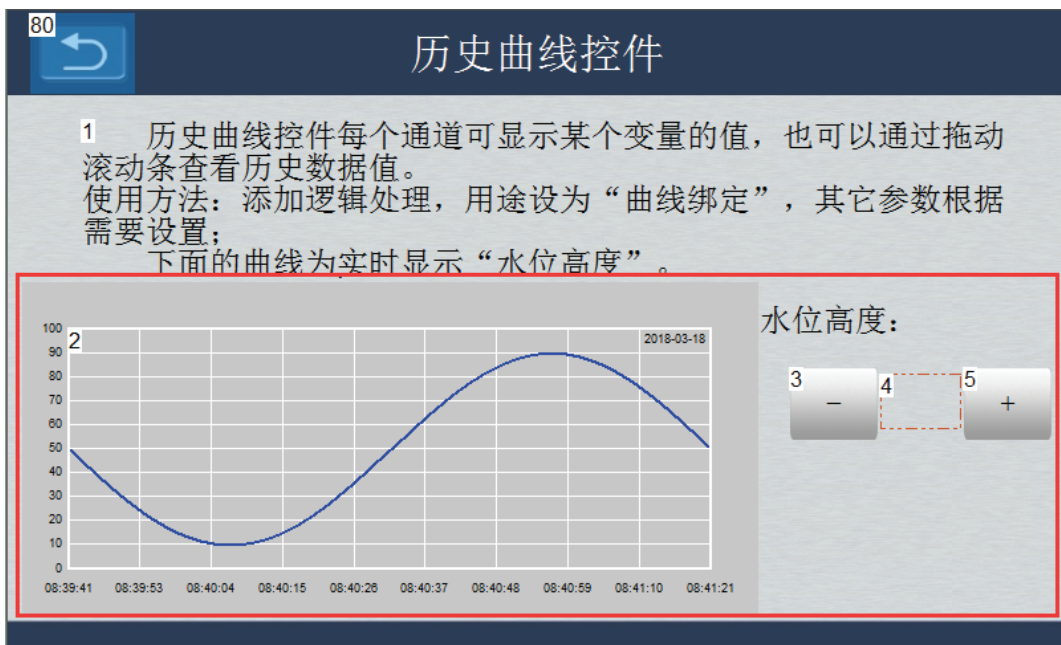


图 4-55 历史曲线控件

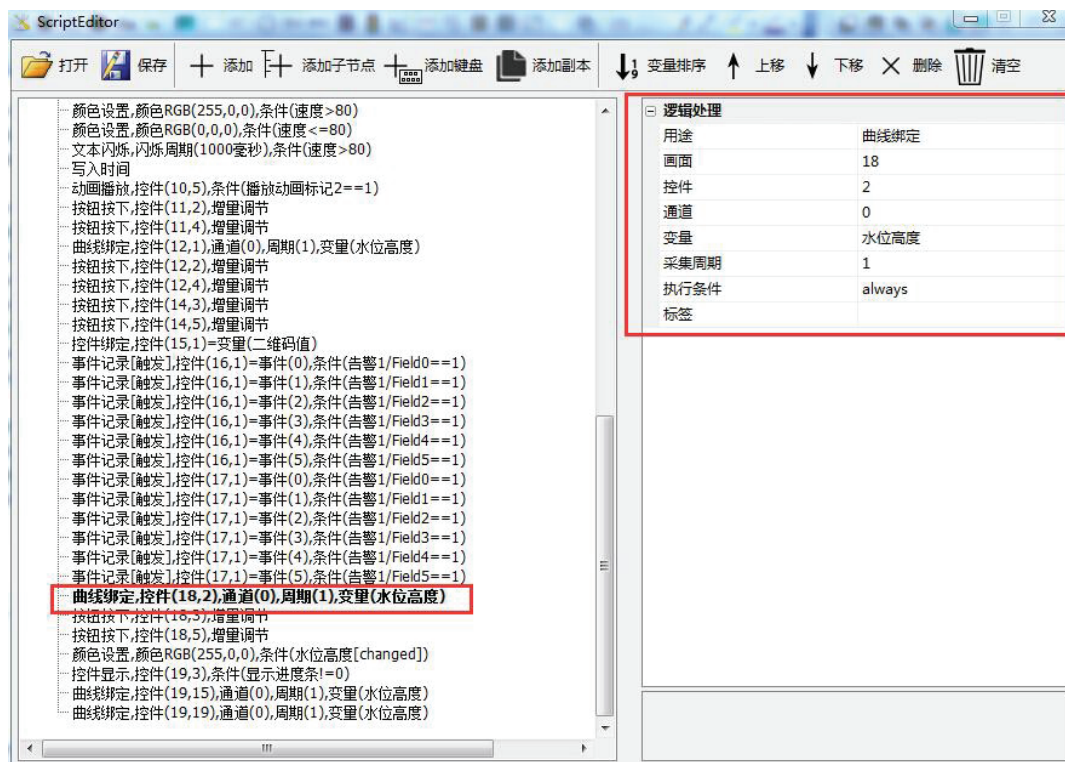


图 4-56 历史曲线控件的逻辑处理

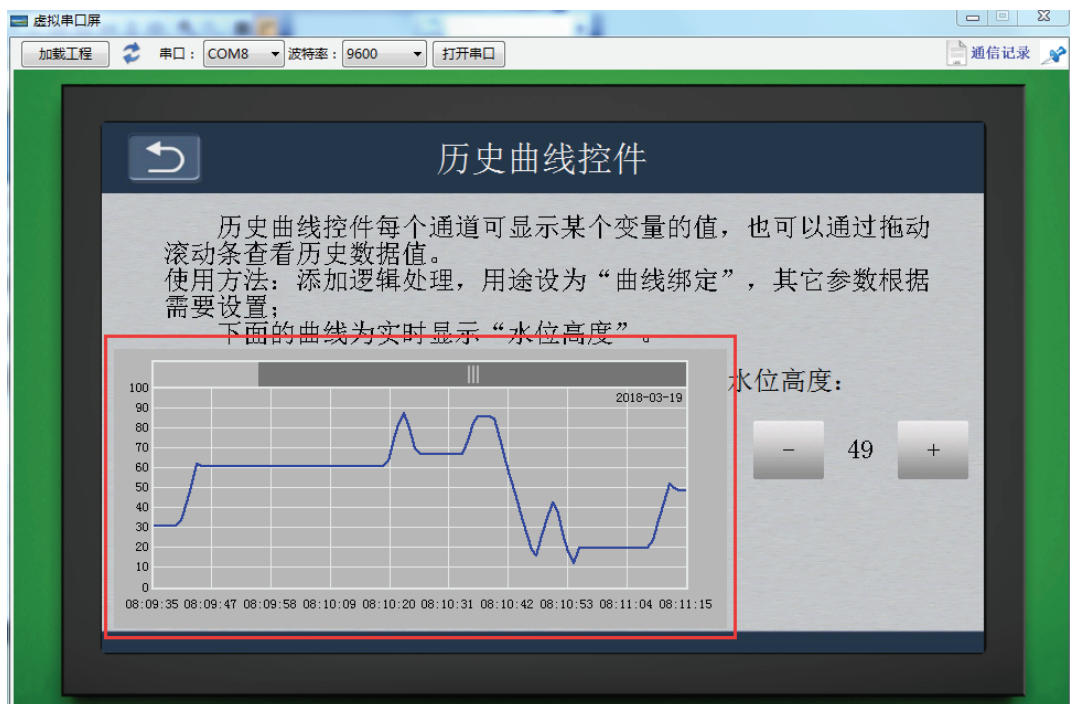


图 4-57 历史曲线控件运行示例

4.2.19 隐藏显示控件

【隐藏显示控件】画面主要对隐藏显示控件功能的使用做示例说明。很多情况下工程需要在某个条件成立的情况下隐藏或显示一些控件。隐藏和显示控件通过两种方式实现：1逻辑处理的控件显示，2、MINIC中的show() hide()函数。

1、 逻辑处理

【隐藏显示控件】画面增加如图 4-58所示，ID为2的文本控件、ID为3的进度条控件，文本控件绑定变量“显示进度条”，增加逻辑处理，如图 4-59所示。

此时，编译工程并运行虚拟串口屏，设置变量“显示进度条”的值为0时，进度条隐藏，设置变量为1，进度条显示。

2、 MINIC 的show() hide()函数

【隐藏显示控件】画面增加如图 4-58所示，ID为5的“隐藏”按钮控件、ID为6的“显示”按钮控件，增加ID从7-19的各种控件。在MINIC代码中添加如图 4-60所示的按钮按下的动作处理。

此时，编译工程并运行虚拟串口屏，按下“隐藏”按钮控件时，ID为7-19的控件隐藏，按下“显示”按钮控件时，ID为7-19的控件显示。

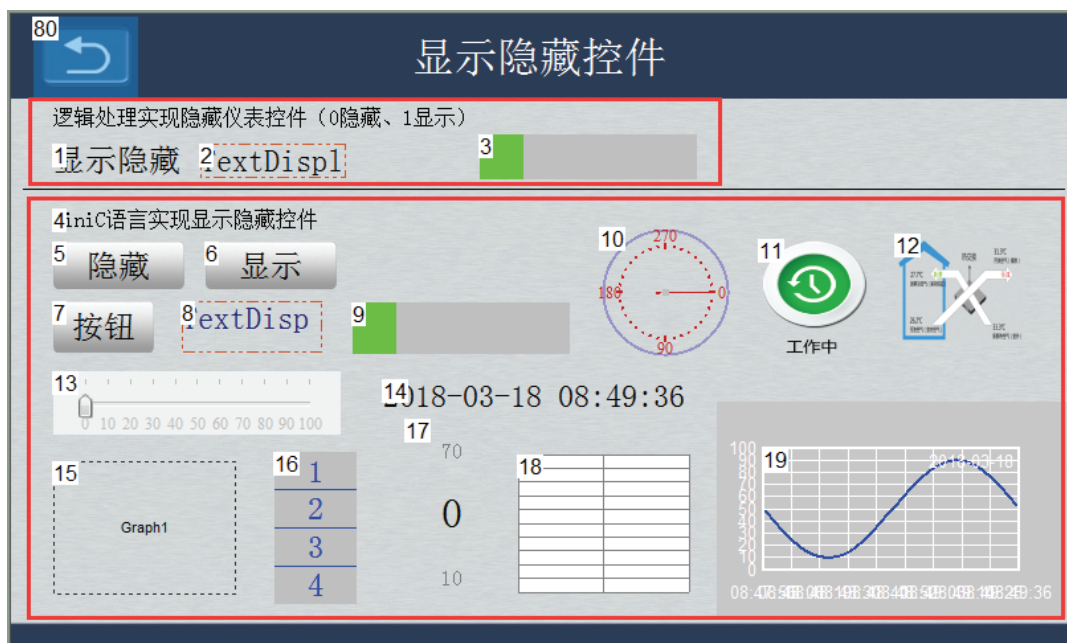


图 4-58 隐藏显示控件

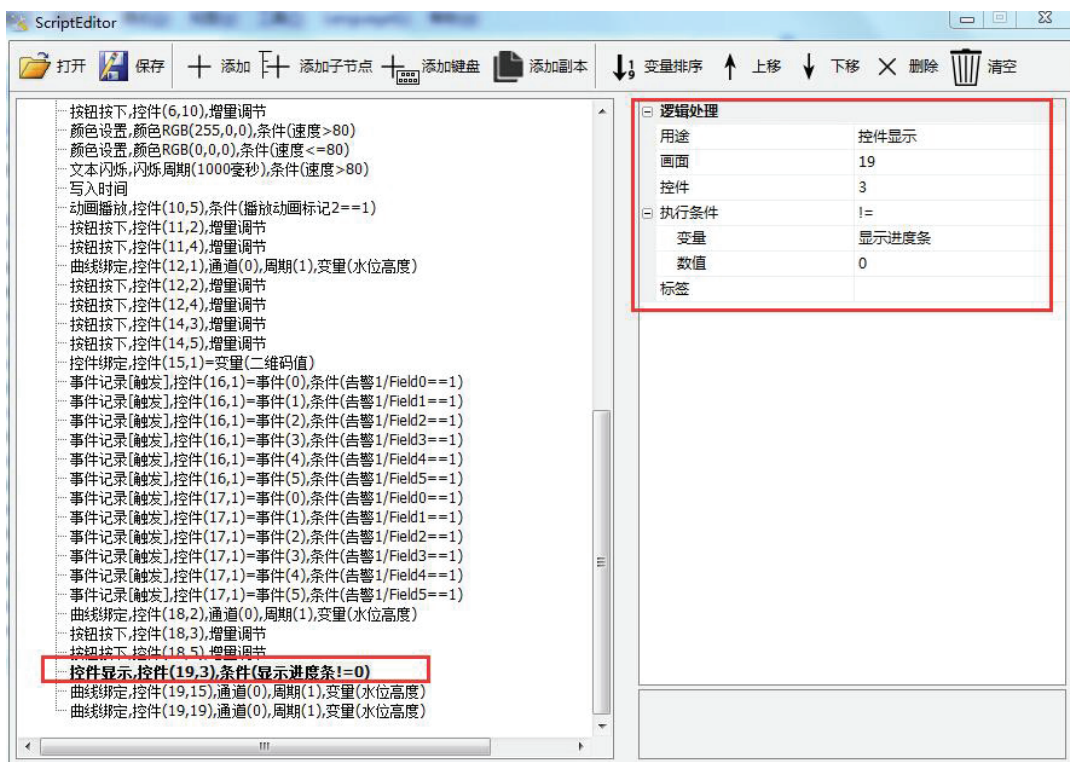


图 4-59 隐藏显示控件的逻辑处理

```

69  /*
70  函数: on_control_notify
71  功能: 控件值更新通知
72  参数: screen_id, 画面ID
73  参数: control_id, 控件ID
74  参数: value, 新值
75  */
76  void on_control_notify(int screen_id,int control_id,int value)
77  {
78      int i;
79      if(screen_id == 19)
80      {
81          //隐藏显示画面
82          if(control_id == 5 && value == 1)
83          {
84              //隐藏
85              for(i = 7; i <=19; ++i)
86              {
87                  hide(19, i);
88              }
89          }
90          else if(control_id == 6 && value == 1)
91          {
92              //显示
93              for(i = 7; i <=19; ++i)
94              {
95                  show(19, i);
96              }
97          }
98      }
99  }
100

```

图 4-60 隐藏显示控件MINIC代码

4.2.20 多语言

【多语言】画面主要对多语言功能的使用做示例说明。很多情况下客户工程需要多语言选择，文字、菜单控件、选择控件支持多语言设置。

【多语言】画面中增加文字、菜单控件、选择控件，如图 4-61所示，多语言属性设置三种语言，增减每种语言的选项。如图 4-62、图 4-63、图 4-64所示。另外增加一个选择控件选择语言类型，绑定变量“语言”，在MINIC种增加对语言选择的处理，设置语言的系统变量sys.lang。

此时，编译工程并运行虚拟串口屏，通过选择控件选择语言种类，文字、菜单控件、选择控件的内容，可以显示不同的语言。

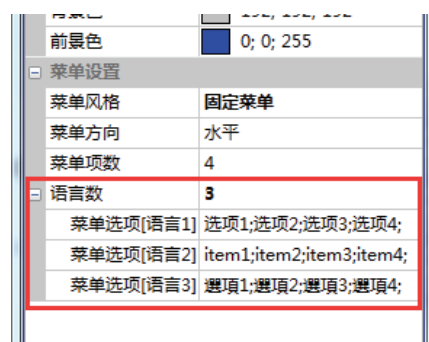


图 4-63 菜单多语言设置

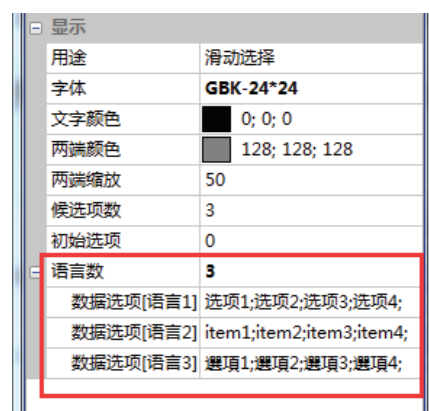


图 4-64 选择控件多语言设置

```

47  /*
48  函数: on_variant_update
49  功能: 串口通信导致变量更新时, 执行此函数
50  */
51  void on_variant_update()
52  {
53      //操作符'@'用于判定某个寄存器是否发生改变
54      //更新RTC时间
55      if (@ "更新标记")
56      {
57          set_date("更新时间"[0], "更新时间"[1], "更新时间"[2]);
58          set_time("更新时间"[3], "更新时间"[4], "更新时间"[5]);
59      }
60
61      //“语言”发生变化时, 修改设置系统语言
62      if (@ "语言")
63      {
64          sys.lang = "语言";
65      }
66  }
67
68

```

图 4-65 多语言

上海英硕自动化科技有限公司

网址: www.ensuretek.net

上海本部

地址: 上海市松江区莘砖公路258号漕河泾新兴产业园38幢301室

邮编: 201612

电话: 021-64326718 64326719 64320097 64320098

传真: 021-64326065

西安办事处:

地址: 西安市高新四路8号新西蓝2期1号楼2102室

邮编: 710075

电话: 029-88696558 传真: 029-88696558

成都办事处:

地址: 成都市金牛区迎宾大道199号金域西岭9-2-3402

邮编: 610036

电话: 028-61362400 传真: 028-61362400



欢迎关注微信公众号