

Kinco[®]

Kinco Designer组态编辑软件 使用手册

本手册适用于智能终端及看板系列产品
使用产品前，请务必仔细阅读本手册
确保正确使用产品

目录

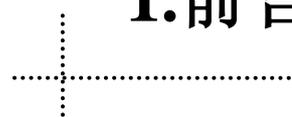
1.前言.....	1
1.1 关于 Kinco Designer.....	1
1.2 KHDesigner 功能总体概括.....	1
2.使用说明	2
2.1 KHDesigner 的使用步骤.....	2
2.2 KHDesigner 软件安装与卸载.....	2
2.2.1 KHDesigner 软件安装.....	2
2.2.2 KHDesigner 软件卸载.....	3
2.3 启动/退出	3
2.3.1 KHDesigner 的启动.....	3
2.3.2 KHDesigner 的退出.....	4
2.4 软件兼容性	4
2.5 升级/更新	4
2.6 USB 驱动程序的安装	5
3.用户界面	9
3.1 布局界面	9
3.2 菜单栏	9
3.2.1 文件菜单	9
3.2.2 编辑菜单	10
3.2.3 查看菜单	11
3.2.4 画图菜单	11
3.2.5 绘图菜单	12
3.2.6 工具菜单	12
3.2.7 选项菜单	13
3.2.8 帮助菜单	13
3.3 工具栏	13
3.3.1 基本菜单栏	13
3.3.2 绘图工具栏	14
3.3.3 画面工具栏	27
3.3.4 位置调整工具栏	27
3.3.5 状态切换工具栏	28
3.4 工程设置窗口	28
3.4.1HMI 属性设置.....	29
3.4.2 通讯设置	30
3.4.3 全局控制设置	35
3.4.4 数据库	47
3.4.5 变量表	48
3.4.6 宏	50
3.4.7 WebService.....	61
3.4.8 HTTP	63
3.4.9 阿里云	66

3.4.10 OPCUA 服务器.....	68
3.4.11 多屏组网.....	68
3.4.12 图形库.....	68
3.4.13 文本库.....	69
3.5 画面窗口.....	70
3.6 编译信息窗口.....	70
4.制作工程.....	71
4.1 工程需求.....	71
4.2 制作工程.....	72
4.2.1 新建工程.....	72
4.2.2 编辑组态画面.....	72
4.2.3 工程保存.....	79
4.2.4 工程模拟.....	79
4.2.5 工程下载.....	80
4.2.6 工程文件夹介绍.....	80
5. 编译/模拟/下载/上载.....	81
5.1 编译.....	81
5.2 模拟.....	81
5.3 下载.....	81
5.3.1 下载方式选择.....	81
5.3.2 与 PC 的下载电缆接线.....	82
5.3.3 下载内容选择.....	82
6. 库的了解.....	84
6.1 文本库应用.....	84
6.1.1 文本库的建立.....	84
6.1.2 文本库的语种设置.....	85
6.1.3 文本库实现多语言切换.....	85
6.1.4 文本库导入导出.....	86
6.2 图形库应用.....	86
6.2.1 图形的导入.....	86
6.2.2 图形的建立与使用.....	86
6.2.3 图片的优化.....	88
6.3 数据库应用.....	88
6.3.1 数据库的安装.....	89
6.3.2 数据库的建立.....	90
6.3.3 数据库的使用.....	91
6.3.4 数据库的特殊情况.....	92
7. KTManager.....	94
7.1 KTManager 介绍.....	94
7.2 打开 KTManager 的方法.....	95
7.3 下载操作.....	96
7.4 上传操作.....	97

7.5 系统操作	98
7.6 无线设置	99
7.7 端口信息	100
7.8 文件管理	101
7.9 获取版本	101
7.10 通讯状态	102
8. HMI 通信	103
8.1 串口通信	103
8.1.1 HMI 与 PC 串口通信	103
8.1.2 HMI 与 PLC/控制器串口通信	103
8.2 网口通信	107
8.2.1 HMI 与 PC 网口通信	107
8.2.2 HMI 与 PLC/控制器网口通信	107
8.2.3 HMI 与 HMI 网口通信	109
8.3 MODBUS 协议应用（主从模式）	111
8.3.1 MODBUS 协议概述	111
8.3.2 MODBUS 协议通讯格式	113
8.3.3 MODBUS 协议在 HMI 中的应用	119
9. 远程监控操作 VNC	125
9.1 广泛的客户端	125
9.2 通过局域网访问	126
9.2.1 PC 机局域网监控操作 HMI	126
9.2.2 智能手机局域网监控操作 HMI	128
10. 打印机	130
10.1 斑马打印机打印制作流程	130
10.2 斑马打印机打印步骤	130
10.1.1 制作模板 PRN 文件	130
10.1.2 组态编辑界面	132
10.1.3 打印模板	132
11. 寄存器	133
11.1 HMI 内部寄存器	133
11.1.1 位地址	133
11.1.2 字地址	133
11.2 HMI 系统寄存器	133
12. 编码类型	141
12.1 BIN	141
12.2 BCD	141
13. 与外围设备的连接	142
13.1 串行接口	142
13.1.1 串口介绍	142
13.1.2 通过串行接口连接	143
13.2 USB 接口	144
13.2.1 通过 usb 接口连接	145

13.3 LAN 接口	146
13.3.1 通过 LAN 接口连接	146
13.4 DIP 开关	147
13.5 IO 口	147
14. 系统设置	148
14.1 系统设置模式的显示方法	148
14.2 系统设置	148
14.2.1 基本设置	148
14.2.2 端口设置	148
14.2.3 有线设置	149
14.2.4 无线设置	149
14.2.5 文件管理	149
14.2.6 数据库设置	149
14.2.7 时间设置	149
15. 触控校准模式	150
15.1 触控校准模式的显示方法	150
15.2 触控校准设置	150
16. 固件更新模式	151
16.1 固件更新模式的显示方法	151
16.2 固件更新设置	151
安全注意事项	152
附录 1 通讯 code 信息一览表	154
附录 2 右键快捷功能使用一览表	155
附录 3 软件使用注意一览表	156

1.前言



首先感谢您选用智能终端及看板系列产品!

在使用智能终端及看板系列产品前,请仔细阅读本手册,在充分理解的基础上正确使用产品以确保相关人员及设备的安全。

1.1 关于 Kinco Designer

Kinco Designer 组态编辑软件(后面简称“KHDesigner”)软件是上海步科自动化股份有限公司为智慧工厂应用开发的专用组态编辑软件(以下简称“KHDesigner”,该软件为用户提供了强大的集成开发环境,产品可适用于产线、仓库及各种特殊应用场合。本手册可作为使用步科公司生产的智能终端及看板系列产品,并具备自动化工程领域的基础知识,具有使用微软相关操作系统的 PC 经验,必要时,还应具备有 C 语言脚本编程知识和能力的用户、调试工程师、服务技术和维修技术等人员的参考。



本手册涉及的内容可能会存在由于产品改进或其它原因,未能及时对手册部分内容进行更新,为保证产品规格及附件等信息获取及时、正确,我们建议用户关注步科公司官网发布的信息。手册更新和软件更新恕不另行通知。

本手册中使用的术语	含义
KHDesigner	表示用于步科公司生产的智能终端及看板系列产品的工具软件: Kinco Designer 组态编辑软件
HMI	表示步科公司生产的智能终端及看板系列产品
PLC	表示可编程控制器
PC	表示个人电脑
外部存储设备	表示步科公司生产的智能终端及看板系列产品支持的标配 USB 存储设备或 SD 卡

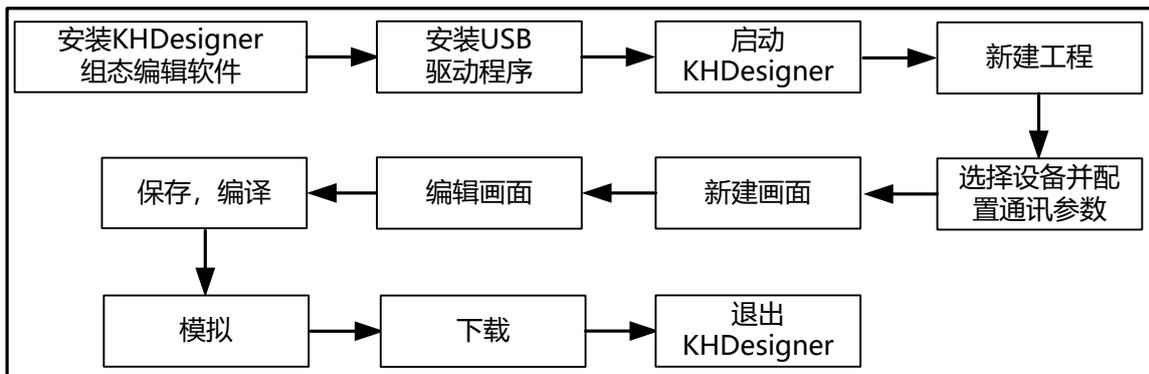
1.2 KHDesigner 功能总体概括

KHDesigner 作为组态软件,提供 IT 系统数据交互设置、用户交互画面编辑、现场设备通讯设置、复杂逻辑编写、组态工程下载、终端设备管理等功能。通过这些功能,用户可以便捷地编制符合用户需求的现场交互场景,实现智慧工厂现场数字化透明化,生产流程柔性化的需求。

2.使用说明

2.1 KHDesigner 的使用步骤

KHDesigner 的使用步骤如下：



2.2 KHDesigner 软件安装与卸载

2.2.1 KHDesigner 软件安装

KHDesigner 集成了“简体中文”和“英文”两种语言的开发环境，请根据需要选择相应的语言界面。以简体中文安装界面过程为例，步骤如下：

双击“Setup.exe”文件运行安装程序，

- ① 进入欢迎界面
- ② 选择安装界面语言，这里选择“简体中文”
- ③ 选择安装目录，软件默认安装到“C:\”根目录下，并在该目录下自动创建“Kinco”文件夹，如要定制目录安装，请单击【浏览】更改安装目录。
- ④ 确认安装

软件将给出 **KHDesigner** 安装状态的提示，单击【安装】则进入软件安装进程。

⑤ 安装成功

如果软件安装成功，会给出一个安装完成的提示。单击【完成】，软件安装结束。

安装结束后，**KHDesigner** 会在【开始】菜单创建完整的启动目录，同时操作系统桌面会创建 **KHDesigner**（集成环境）和 **KTManager** 的快捷方式。

在上述安装过程中，用户单击【取消】，可退出安装。

2.2.2 KHDesigner 软件卸载

执行卸载前，请先退出 KHDesigner 程序。

1.启动卸载程序

方法 1: 从【开始】菜单中卸载

【开始】→【所有程序】→【Kinco】→【KHDesigner】→【Uninstall】

方法 2: 从【控制面板】→【添加或删除程序】中卸载

2.卸载过程

选中“除去”，单击【下一步】即可进入卸载进程。其中：

修改	高级用户，自定义要添加的新功能或选择要除去的已安装功能 选择该选项可以进行软件的维护
修复	重新安装 KHDesigner 程序 选择该选项可以进行软件的维护和更新
除去	卸载 KHDesigner 程序



如果在 KHDesigner 安装目录下有增添文件或文件夹，请在软件卸载完毕后手动删除这些内容

2.3 启动/退出

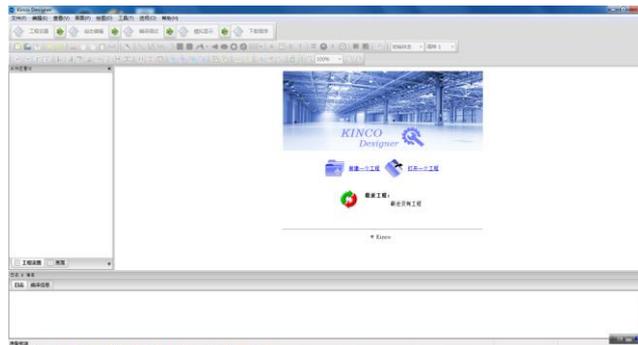
2.3.1 KHDesigner 的启动

方法 1: 从【开始】菜单中启动

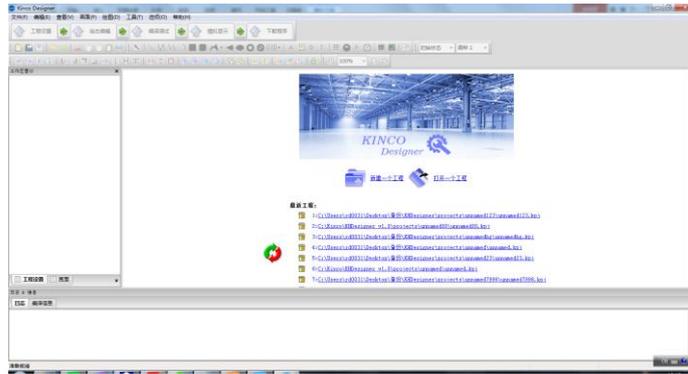
【开始】→【所有程序】→【Kinco】→【KHDesigner vx.x】→【KHDesigner】

方法 2: 双击桌面 KHDesigner 快捷方式启动软件

- 首次启动 KHDesigner 时，将显示如下窗口界面：



- 非首次启动 KHDesigner 时，将保留以前操作的工程项目



2.3.2 KHDesigner 的退出

启动 KHDesigner 程序后，可按照下列任意一种方法退出程序：

- 单击主窗口右上角的【关闭】按钮
- 单击主窗口左上角的 KHDesigner 图标，从显示的控制菜单栏中选择“关闭”



- 单击 KHDesigner 【文件】 菜单，从下拉菜单中选中“退出”
- 同时按下键盘上的【Alt】键和【F4】键

2.4 软件兼容性

	<p>KHDesigner 软件版本向上兼容，即老版本软件编的工程可以用新版本的软件打开，但新版本软件编的工程无法用老版本软件打开</p>
---	--

2.5 升级/更新

	<ol style="list-style-type: none"> 1.为确保产品的正确使用，禁止软件在未关闭的状态下进行更新、修复、卸载和重新安装 2.软件升级前，为确保产品的正确使用，请完全卸载旧版本软件 3.在使用新版本的软件更新旧版本软件做的工程前，请备份好旧版本做的工程
---	--

2.6 USB 驱动程序的安装



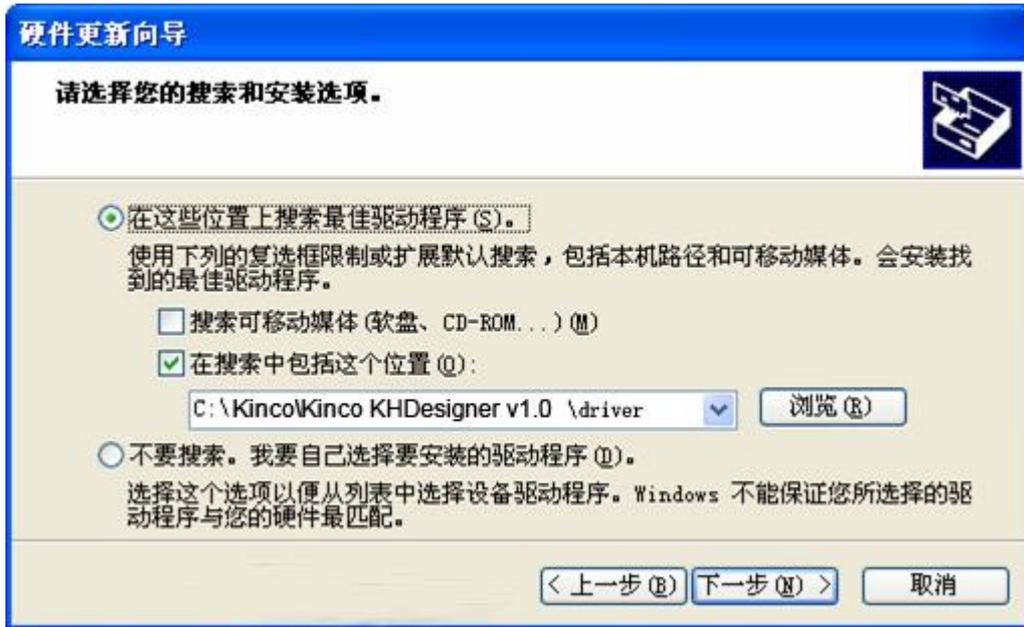
USB 驱动程序必须手动安装，不能自动安装

针对有 USB 从口的终端设备，用户可以在 PC 中安装 USB 驱动程序用于通过 USB 连接线下载 HMI 程序。用户可通过下述步骤手动安装 USB 驱动：

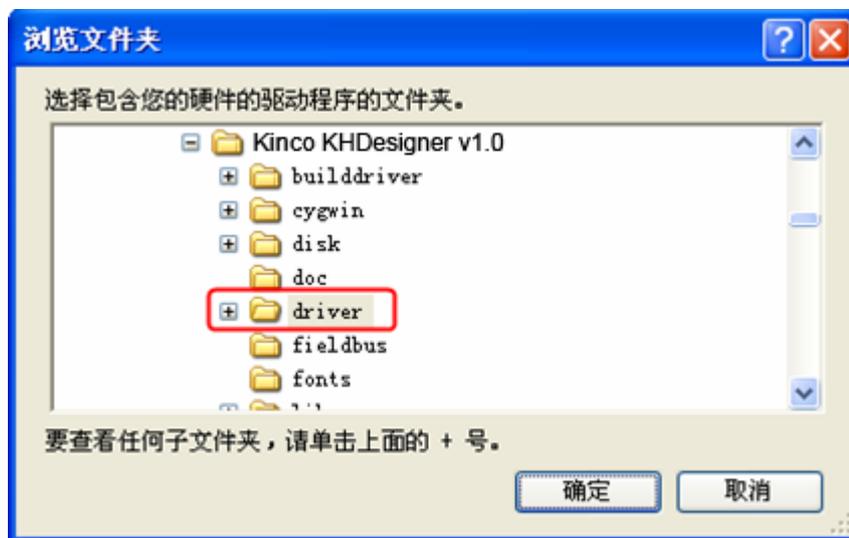
① 用 USB 电缆连接 HMI 的 USB SLAVE 接口和 PC 的 USB HOST 口，正确连接 HMI 电源并打开电源开关，PC 操作系统上将自动弹出【硬件更新向导】对话框，选择“从列表或指定位置安装（高级）”后，单击【下一步】：



② 显示如下对话框。选中“在这些位置上搜索最佳驱动程序”，然后选择“在搜索中包括这个位置”，并单击【浏览】：



③ 路径指向 KHDesigner 安装目录下的“driver”文件夹，然后单击【确定】：



④ 安装结束后，将显示“完成硬件更新向导”对话框，单击【完成】后退出。

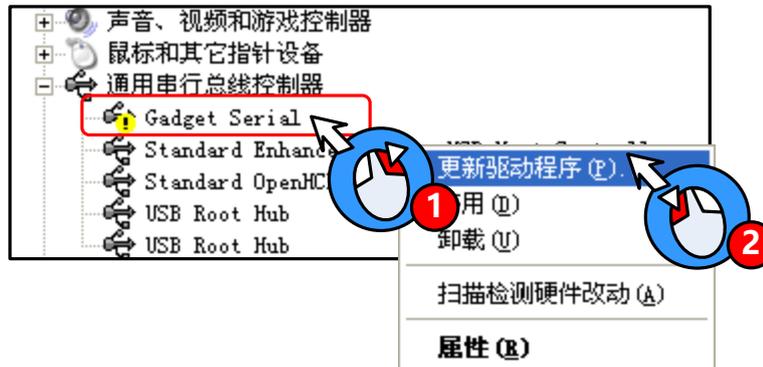


如果用户在执行操作步骤 1 时，PC 未自动弹出【硬件更新向导】，而托盘区出现“发现新硬件”的气泡提示，如图所

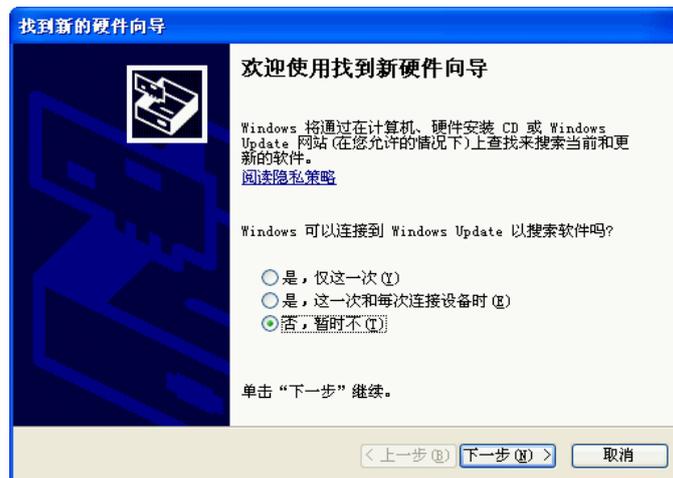
示：



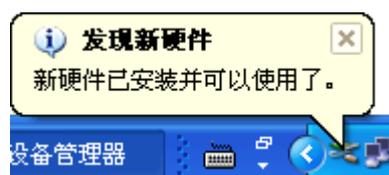
此时打开操作系统的【设备管理器】→【通用串行总线控制器】中会有未知设备的检测图标出现，在该图标上右击，选择“更新驱动程序”，如图所示：



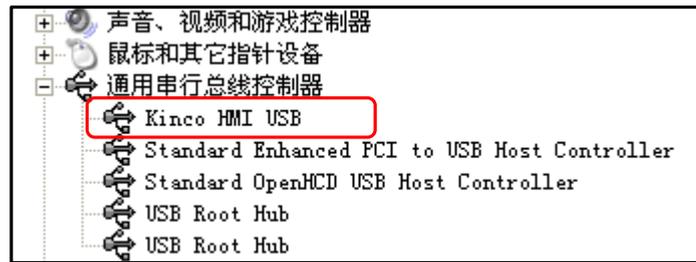
此时操作系统会自动弹出【找到新硬件向导】：



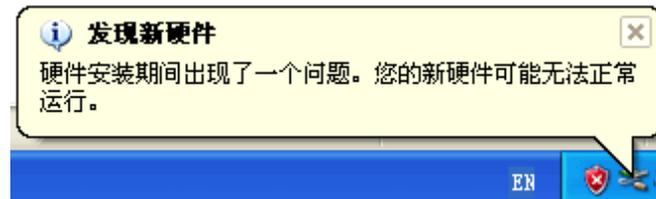
选择“否，暂时不”后，单击【下一步】，然后继续根据前面的步骤安装 USB 驱动程序。安装完毕，操作系统托盘区会出现“新硬件已安装并可以使用了”的气泡提示，如图所示：



同时打开操作系统的【设备管理器】，可以在【通用串行总线控制器】下看到原未知设备已显示为“Kinco HMI USB”的字样。此时设备驱动程序安装成功。



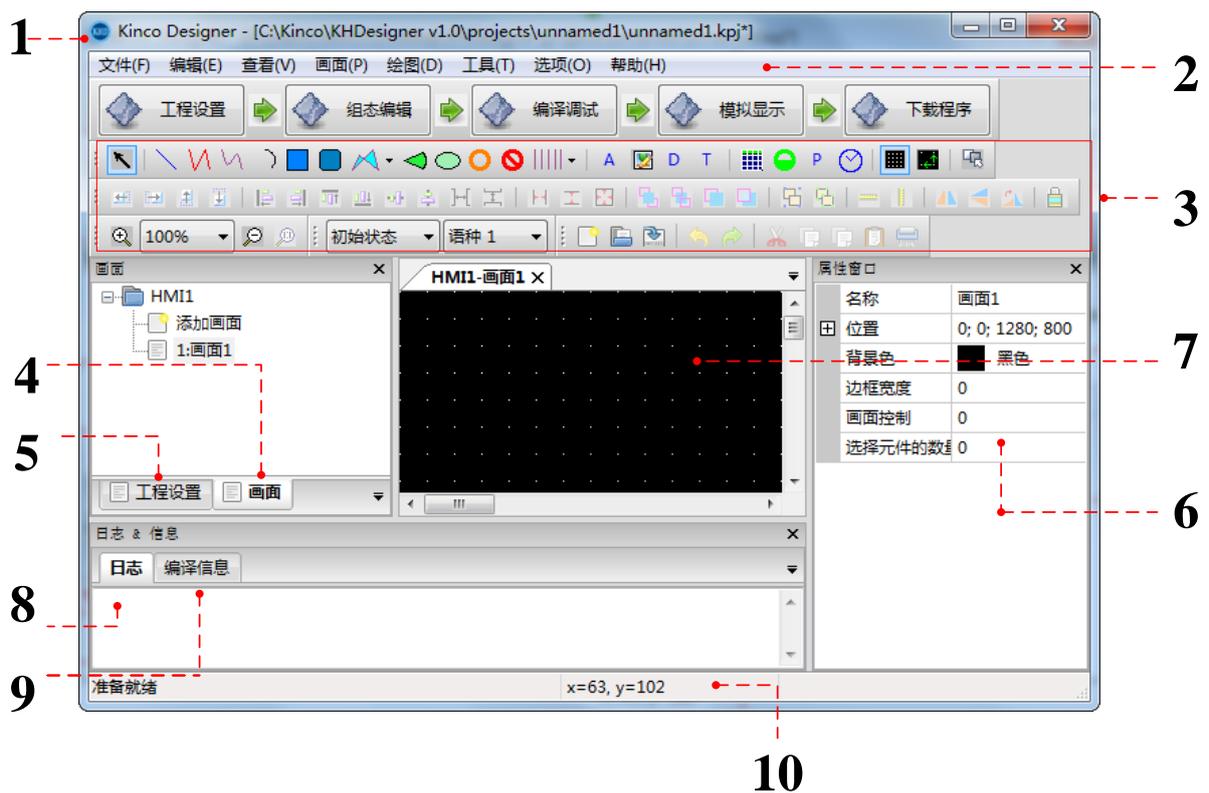
如果未能正确安装 **USB** 驱动，操作系统托盘区会有如下提示：



同时，操作系统【设备管理器】中仍显示未知设备的图标，无“Kinco HMI USB”字样出现。遇此问题，请选择正确的驱动程序目录重新安装 **USB** 驱动。

3. 用户界面

3.1 布局界面



- 1.标题栏 2.菜单栏 3.工具栏 4.画面管理窗口 5.工程设置界面 6.属性窗口 7.组态编辑界面 8.日志窗口
9.编译信息窗口 10.状态栏

3.2 菜单栏

3.2.1 文件菜单

- 新建工程：新建一个工程。

- 打开工程：打开一个已经存在的 **kpj** 工程。
- 打开最近工程：可以打开最近打开的工程，方便用户直接打开工程。
- 关闭工程：若工程无改动，则直接关闭。若工程有改动，则软件提示是否保存。
- 保存工程：对所建的工程进行保存。
- 工程另存为：使用新的文件名和文件路径。
- 默认路径：默认路径为软件文件夹下的 **projects** 文件夹下，可以更改自己所建工程的默认路径。
- 退出：若工程无改动，则直接退出。若工程有改动，则软件提示是否保存。

3.2.2 编辑菜单

- 撤销：取消这次的操作返回上一步（支持多步撤销，最多可撤销 **12** 步）
- 恢复：恢复这次编辑的操作（支持多步恢复）
- 删除/剪切/复制/粘贴：对元件进行删除、剪切、复制、粘贴操作
- 多重复制：可以对元件进行水平和垂直方向快速的复制



勾选下标变量后，对元件属性窗口关联的数组变量下标和非数组的按变量名递增或递减（如果变量名为 **vw**，复制 **5** 则将 **vw** 加上 “_n”，其中 **n** 的值从 **0** 开始，复制到对应的数量）；递减如果数量超出数组下表或者 **_n**，超出的按照数组的 **0** 下表和非数组的 **_0** 显示递增按照最大显示；；多重复制下标递增递减支持动态显示和触控动作中的表达式：包括多态显示，水平移动，垂直移动，水平缩放，垂直缩放，动态填充，动态旋转，动态闪烁，显现控制，触控控制。

- 全选元件：将组态编辑界面里的元件全部选中
- 取消全选元件：取消组态编辑界面里全部选中的元件
- 微调：将选中的元件向左/右/上/下移动一个像素点
- 对齐：将选中的元件按照左对齐/右对齐/上对齐/下对齐/水平中线对齐/垂直中线对齐，当元件相同时可以选中水平间距相同、垂直间距相同
- 尺寸：可以设置多个元件等宽、等高、相同尺寸。
- 层次：当多个元件重叠时，可以设置元件置顶、置底、向前一层、向后一层
- 组合/取消组合：将多个元件组合在一起，或者取消组合在一起
- 居中：将选中元件居中、水平居中、垂直居中
- 水平翻转/垂直翻转/旋转 **90** 度：将选中的元件水平翻转/垂直翻转/旋转 **90** 度



只对形状一类的有效，例如矩形、多边形

- 锁定元件位置：将元件锁定，不能再移动



锁定的元件支持复制粘贴

- 显示网格：当选中这个选项时，把背景网格显示出来

- 对齐网格：当选中这个选项时，元件移动时会自动对齐网格
- 定义网格间距/颜色：可根据喜好、设置网格大小和颜色
- 显示元件名称：当选中这个选项时，会显示元件名称
- 元件名称字体大小/颜色：可根据喜好设置字体大小和颜色

3.2.3 查看菜单

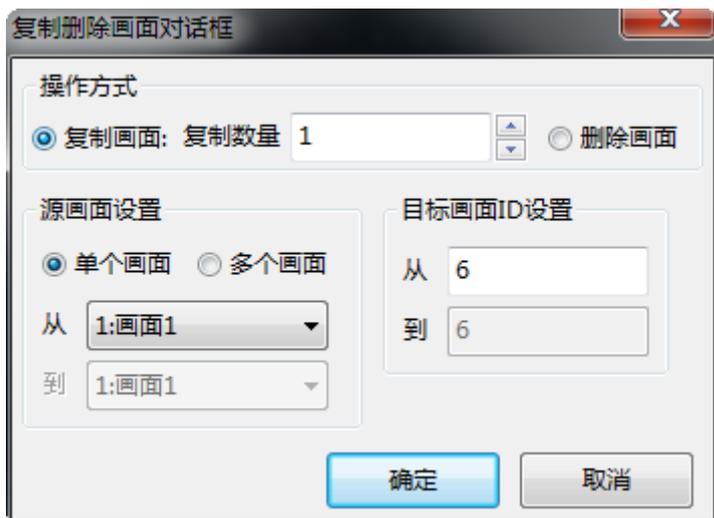
- 布局：创建新的布局（移动界面窗口与默认的不一样时可以使用，关闭工程时布局失效）和默认布局（恢复初始画面状态）
- 工具栏：可以选择显示或隐藏的工具栏
- 状态栏：可以选择显示和隐藏状态栏
- 工程设置窗口：可以选择显示和隐藏工程设置窗口画面管理窗口：
- 属性窗口：可以选择显示和隐藏属性窗口
- 日志窗口：可以选择显示和隐藏日志窗口
- 使用环境：选中显示数据库、画面控制、变量、宏等使用的环境，双击可以跳转到对应的地方
- 编译信息窗口：可以选择显示和隐藏编译信息窗口，最好不要隐藏；编译信息有错时，点击对应的错误信息会跳转到对应的地方；
- 语种：组态界面使用的语种数，快速切换对应的语种，并在界面显示；
- 画面缩放：**25%~300%**；设置以后使用所有的画面，只对当前设置工程有效；
- 正常尺寸：画面按照正常的 **1:1** 即画面缩放 **100%** 显示

3.2.4 画图菜单

- 添加画面：可以快速添加画面，画面 **ID** 号可以自动补缺也可以自己设置；**ID** 一旦设置，不可以修改；
- 复制删除画面

复制画面：可以复制多个相同的单个或多个画面，复制数量范围 **1-999**，画面 **ID** 只可以使用未使用的 **ID**

删除画面：删除单个画面或多个画面



3.2.5 绘图菜单

绘图菜单包括直线、折线、曲线、弧线、矩形、圆角矩形、多边形、扇形、椭圆、空心圆、禁止圆、刻度、标签、位图、数据、文本、数据报表、仪表盘、PDF 控件、时间日期、曲线图和饼图。

3.2.6 工具菜单

- 编译：不会重复编译已经编译过的宏文件
- 全部编译：会清除无用的 WSDL 文件（包括宏和 webservice 生成的），清除 WSmacro 文件夹下未使用的 cpp 文件，未使用的图片文件，然后编译文件
- 清除编译结果：清除无用的 WSDL 文件（包括宏和 webservice 生成的），清除 WSmacro 文件夹下未使用的 cpp 文件，未使用的图片文件
- 清除模拟数据：清除模拟显示的数据信息
- 编译 SOP:编译工程为 sop 工程
- 清除编译 SOP 结果：清除 sop 工程无用的 WSDL 文件（包括宏和 webservice 生成的），清除 WSmacro 文件夹下未使用的 cpp 文件，未使用的图片文件
- 离线模拟：可以选择工程进行模拟显示，PLC 数据不是实际的 PLC 数据；
- 实时模拟：PLC 的编程线和 PC 串口直接相连，进行实时模拟；



1.实时模拟的测试时间是 20 分钟。超过 20 分钟后，则会提示：实时模拟时间过长，程序已退出！

如果需要，请重新模拟

2.大部分只能使用 RS232 通讯方式进行实时模拟，部分网口通讯的 PLC 设备可以使用网口实时模拟

3.RS232 通讯方式的实时模拟接线方法：PLC 的编程线和 PC 串口直接相连。网口通讯方式的实时接线方法：交叉网线直接相连或通过 Switch 转接

4.实时模拟时使用的端口不能被其他程序占用，否则模拟时将通讯不上

- 上传/下载：对工程进行上传和现在，目前只支持下载功能
- 下载操作：调用 **KTManager** 的下载操作界面，详见 **7.KTManager** 对应介绍
- 上传操作：调用 **KTManager** 的上传操作界面，详见 **7.KTManager** 对应介绍
- 系统操作：调用 **KTManager** 的系统操作界面，详见 **7.KTManager** 对应介绍
- 无线设置：调用 **KTManager** 的无线设置界面，详见 **7.KTManager** 对应介绍
- 端口信息：调用 **KTManager** 的端口信息界面，详见 **7.KTManager** 对应介绍
- 文件管理：调用 **KTManager** 的文件管理界面，详见 **7.KTManager** 对应介绍
- 获取版本：调用 **KTManager** 的获取版本界面，详见 **7.KTManager** 对应介绍
- 通讯状态：调用 **KTManager** 的通讯状态界面，详见 **7.KTManager** 对应介绍
- 图形库：点击可打开图形库，进行图形库的设置，详见 **6.2** 图形库应用
- 文本库：点击可打开文本库，进行文本库的设置，详见 **6.1** 文本库应用

3.2.7 选项菜单

- 设置绘图默认设置：设置基本绘图元件的默认设置，例如：线型，线宽，线颜色、背景色、前景色和填充类型
- 字体默认设置:设置字体的默认设置
- 画面背景颜色设置：默认画面背景颜色
- 系统语言切换：包括中英文两种

3.2.8 帮助菜单

查看通讯文档、软件使用说明、软件升级内容和获取 **KHDesigner** 软件版本号信息。

3.3 工具栏

工具栏分为基本菜单栏、绘图菜单栏、画面工具栏、位置调整工具栏、标签对齐工具栏、状态切换工具栏。

3.3.1 基本菜单栏



基本工具栏可以新建工程、打开工程、保存工程；对操作步骤进行撤销和还原；对元件进行剪切、复制、多重复制、粘贴和删除。

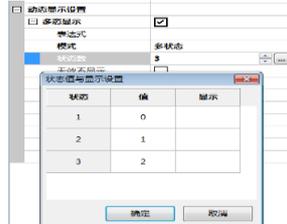
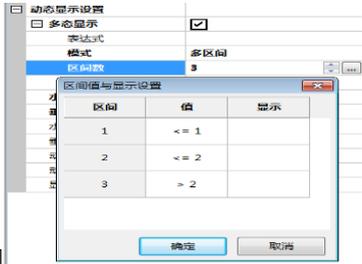
3.3.2 绘图工具栏

绘图菜单包括直线、折线、曲线、弧线、，矩形、圆角矩形、多边形、扇形、椭圆，空心圆、禁止圆，刻度，标签，位图，数据，文本，数据报表，仪表盘，PDF 控件。直线、折线、曲线和弧线的属性窗口类似，以直线为例介绍；矩形、圆角矩形、多边形、扇形、椭圆、空心圆和禁止圆属性窗口类似，以矩形为例；元件允许移出页面显示区域，但在编译时会提出警告。



3.3.2.1 直线的属性窗口

直线元件属性说明				
名称	可修改，不可重复，不能为空，限制字数为 20 个字符			
描述	标记元件的功能，限制输入长度为 100 个字			
位置	用于标记元件在页面上的坐标，元件允许移出页面显示区域，但在编译时会提出警告			
线型	线型指的是直线，曲线或图形边框的线条的类型，也就是画笔的样式，分为实线、虚线、点和虚线、双点和虚线、长虚线五种。			
线宽	分为 1-9 等宽度			
线颜色	设置线的颜色			
箭头	箭头指的是线条两端绘制的箭头类图形，分为 7 种：无箭头、起点折线箭头、终点折线箭头、双向折线箭头、起点三角箭头、终点三角箭头、双向三角箭头。			
	用于定义元件的显示属性，除显现控制属性外，其它属性都是可以通过勾选框单独勾选，在实际运行中，各类属性的处理顺序为：显现控制->多态显示->水平移动->垂直移动->水平缩放->垂直缩放->动态旋转->动态填充。动态闪烁根据条件单独判断。			
	多态显示	表达式	输入框+弹出框按钮，默认为空，用户可以输入表达式做为条件	
		模式	开关状态	状态固定为两个，关和开
			多状态	用户可选状态数，最小为 2 ，最大为 255 ，每个状态对应一个数值，当条件数值不属于任何一个状态时，则当前状态为无效状态
		多区间	用户可选区间数，最小为 2 ，最大为 255 ，可以设置的区间判断值为区间数减 1 ，以区间数为 3 ，区间值分别为 0,100 举例，则 3 个区间分别为负无穷到 0,0 到 100,100 到正无穷，其中每个区间值都包含在上 一区间，比如 0 属于第一个区间， 100 属于第二个区间。	
		状态数	开关状态 3*2 表格，标题栏：状态、显示；如图	

动态显示设置			
		多状态	<p>$n \times 3$ 表格, $n = \text{状态数} + 1$。标题栏: 状态、值、显示; 内容栏行数等于状态数, 每行依次为状态号 (从 1 开始递增), 值 (用户输入); 如图</p> 
		多区间	<p>多区间: $n \times 3$ 表格, $n = \text{区间数} + 1$。标题栏: 区间、值、显示; 内容栏行数等于区间数, 每行依次为区间号 (从 1 开始递增), 值 (用户输入, "\leq值"); 如图</p> 
		无效不显示	<p>元件在两种状态下被认为显示条件无效:</p> <ol style="list-style-type: none"> 1.使用的变量通讯不成功导致没有数据; 2.模式为多状态时,显示条件计算出来的值和任何一个状态值都不匹配。当勾选此选项时, 如果元件处于无效状态则不显示, 否则当元件处于无效状态时显示元件的默认状态。
	输入框+弹出框按钮, 值范围在 2-255, 默认为 2, 当用户点击显示时, 弹出显示设置框, 用户可以设置画笔的样式, 如果是文字, 则可以设置字体相关信息。		
水平缩放	表达式	默认为空, 用户可以输入表达式做为条件	
	最小值	最小值默认为 0, 范围为-2147483648-2147483648, 且最小值必须小于等于最大值	
	最大值	最大值默认为 100, 范围为-2147483648-2147483648	
	最小值对应百分比	默认为 0, 范围为 0--100。	
	最大值对应百分比	默认为 100, 范围为 0--100。	
	类型	从左向右/从右向左/中心点向左右, 默认从左向右	
	用于定义元件的水平缩放属性, 缩放比例以元件组态时定义的大小为基准, 比例为 0 时不显示, 比例为 100 时完全显示。如果表达式计算值小于最小值, 则		

		<p>默认为最小值，如果大于最大值，则默认为最大值。</p> <p>举例:假设元件表达式为变量 X，最小值为 0，最大值为 1000，最小值对应百分比为 0，最大值对应百分比为 100，类型为从左向右。当 X 的值为 -1 时，由于表达式值小于最小值，所以值默认为最小值 0，对应的百分比是 0，元件不显示；当 X 的值为 500 时，由于 500 在 0-1000 之间，所以取实际值 500，计算显示百分比为 50，所以元件显示一半，当 X 的值为 1000 时，取实际值 1000，计算百分比为 100，所以完全显示。当 X 的值为 2000 时，取实际值 1000，计算百分比为 100，所以完全显示。</p>						
	垂直缩放	用于定义元件的垂直缩放属性，规则和水平类似，可参见水平缩放						
	水平移动	用于定义元件在水平方向的移动动作，移动以用户组态时元件的坐标位置为基准，像素负数代表向左移动，像素正数代表向右移动，移动像素范围以所选组态屏的长宽为限制。						
	垂直移动	用于定义元件在垂直方向的移动动作，移动以用户组态时元件的坐标位置为基准，像素负数代表向上移动，像素正数代表向下移动。						
	动态旋转	用来定义元件的旋转属性						
		<table border="1"> <tr> <td>表达式</td> <td>输入表达式做为条件</td> </tr> <tr> <td>最小值</td> <td>最小值默认为 0，范围为 -2147483648-2147483648，且最小值必须小于等于最大值</td> </tr> <tr> <td>最大值</td> <td>最大值默认为 100，范围为 -2147483648-2147483648</td> </tr> </table>	表达式	输入表达式做为条件	最小值	最小值默认为 0 ，范围为 -2147483648-2147483648 ，且最小值必须小于等于最大值	最大值	最大值默认为 100 ，范围为 -2147483648-2147483648
表达式		输入表达式做为条件						
最小值		最小值默认为 0 ，范围为 -2147483648-2147483648 ，且最小值必须小于等于最大值						
最大值		最大值默认为 100 ，范围为 -2147483648-2147483648						
		最小值对应旋转角度	默认为 0 ，范围为 0-360 。					
		最大值对应旋转角度	默认为 100 ，范围为 0-360 。					
	旋转点距离中心点 X 轴的偏移	默认为 0 ，范围以所选组态屏的长宽为限制。						
	旋转点距离中心点 Y 轴的偏移	默认为 0 ，范围以所选组态屏的长宽为限制。						
	动态闪烁	表达式	输入表达式做为条件					
		频率（秒）	范围为 1-60 ，默认为 1					
	动态填充	用于定义元件的动态填充属性，填充比例以元件组态时定义的大小为基准，比例为 0 时完全不填充，比例为 100 时完全填充。						
		表达式	输入表达式做为条件					
		最小值	最小值默认为 0 ，范围为 -2147483648-2147483648 ，且最小值必须小于等于最大值					
		最大值	最大值默认为 100 ，范围为 -2147483648-2147483648					
		最小值对应百分比	默认为 0 ，范围为 0-100 。					
		最大值对应百分比	默认为 100 ，范围为 0-100 。					
	类型	从左向右/从右向左/中心点向左右/从下向上/从上向下/中心点向上下，默认从左向右						
	显现控制	一直显示	默认属性					

		不显示	如果用户需要在 HMI 运行时去除元件但又希望保留组态画面上的元件配置以便下回需要时继续使用，那可以将元件属性设为不显示。当元件不显示时，在 HMI 运行时元件所有其它属性都无效，相当于画面上不存在这个元件。
		条件显示	用户可以设置在 HMI 运行时通过条件判断动态地决定元件显示与否。表达式控制勾选后显示表达式属性项，当表达式为真（值为非 0 数）时显示，为假（值为 0 ）时不显示。

3.3.2.2 矩形的属性窗口

	矩形元件是用户在画面上通过鼠标拖动选择一块区域绘制矩形框
---	------------------------------

和直线属性类似的不再重复描述，只描述直线没有的。

弧形刻度元件属性说明	
填充类型	一般填充/图案填充/渐变填充/不填充
前景色	图案的前景颜色
背景色	图案的背景颜色
按钮效果	选中选项框后，图案显示为按钮效果，可设置自动调整按钮厚度和自定义按钮厚度
	自动调整 自动调整按钮厚度
	按钮厚度 设置按钮厚度，范围为 1-31
触控动作	变量触发动作/按下时触发动作/.按住不动时触发动作/弹起时触发动作
触控控制	一直可触控/不可触控/条件触控

	<ol style="list-style-type: none"> 1.图案类型分为反向对角线填充、交叉对角线填充、正向对角线填充、网状填充、水平填充、垂直填充。 2.渐变方向分为从左向右、从右向左、从下向上、从上向下。 3.变量触发动作: 变量必须为位变量, 必须和下面的三个触发动作配合使用, off->on 等同于按下; on 等同于按住不动; on->off 等同于弹起。 4.不可触控: 用户触控无效。显示触控无效标志, 当用户选中时元件显示触控无效图形。 5.表达式控制: 当表达式为真 (值为非 0 数) 时可触控, 为假 (值为 0) 时不可触控。 <p>其他功能和直线类似, 不做重复介绍</p>
---	--

3.3.2.3 刻度的属性窗口

	刻度元件可用于需按等比分度量的标注，刻度元件分为水平刻度、垂直刻度、弧形/圆形刻度。
---	--

刻度元件属性说明	
主刻度	设置表针元件主刻度的数目（2~100 可设）、长度、线型、线宽、线颜色
副刻度	设置表针元件副刻度的数目（1~100 可设）、长度、线型、线宽、线颜色
开始角度	只有弧形/圆形刻度有，0-360 可设
结束角度	只有弧形/圆形刻度有，0-360 可设
显示轴	设置轴的显示与否
显示方向	可以选择外圈和内圈

3.3.2.4 标签、位图的属性窗口

	标签元件是用来显示文字的基本图元
--	------------------

标签元件属性说明	
名称	名称固定，不可以修改
位置	显示元件 X/Y 坐标、宽/高
内容来源	有直接输入和文本库两种方式，默认选择直接输入。直接输入是用户直接在下方输入框中输入文字，默认值为 Label ，组态编辑显示效果和实际一致；文本库是用户从文本库中选择相应的文本内容进行显示，文本名称默认选择第一条文本，当工程文本库为空时无法选择，文本内容是根据当前文本库以及当前工程设定的语种显示相应的文本，不可修改。组态编辑显示效果和实际一致。当文本多行显示时，只显示第一行，选择元件时会全部显示
自由调整大小	选中可以手动调整边框大小，同时可以选择自动换行功能
字体设置	设置字体的语种、字体、大小、颜色、粗体、斜体、有无背影以及背影颜色、显示方向、齐方式、水平和垂直对齐方式、行间距、列间距、边距、多行对齐。
走马灯	设置文字内容以走马灯的形式进行循环移动显示与否。使用走马灯时，可设置步长（范围 1-255，默认值为 10）、间隔（范围 0-255，默认值为 50）、频率(*100ms)（范围 1-65535，默认值为 10）。当选择走马灯时，自由调整大小便自动隐藏。

	当用户选择走马灯方式时，不管文字内容来源何处，宽高可修改；当文字内容来源用户选择直接输入或文本库时，只有用户选择自由调整大小时，宽高才可修改；其它情况下元件的宽高都根据用户输入的文字和字体自动变化；选择文本库时使用所有语种情况下的宽高最大值；选择多态显示属性时还需要考虑所有状态的值；
---	--

	位图元素是用来显示图片的元素，用户可以从 PC 中选取任意软件支持格式的图片显示在界面上。目前软件支持的格式包括 BMP/PNG/JPG/TIF/PNM/PCX/ICO/CUR/ANI/TGA/XPM 。
---	---

位图元素属性说明				
位图来源	本地图片	输入框+弹出文件选择框，显示当前使用的图片的名称，点击弹出文件选择框后可选择具体图片文件		
	外部图片	设备类型	HMI	
		文件类型	bmp/jpg/png	
		刷新变量	刷新变量为 ON 时刷新图片，自动复位为 OFF	
		存储路径	选择文件下载和上传后的存储路径，默认路径/home/user，也可以是下面的子文件夹	
		文件变量类型	常量	点击弹出文件选择框后可输入具体图片文件
变量	点击弹出文件选择框后选择表示图片文件的变量			

	<p>1.在选中位图元件后，软件会自动将图片拷贝到工程文件夹的 image 目录。如果用户在工程中使用了多个位图元件，选取了和 image 目录下同名的位图图片，则将该图片名加上 “_n” 后保存到 image 目录,其中 n 的值从 1 开始，根据 image 目录中是否有同名文件确定。</p> <p>2.位图不支持中文文件名（包括含有中文路径）</p>
---	---

3.3.2.5 数据、文本的属性窗口

	数值显示元件是将指定的 HMI 内部寄存器或 PLC /控制器寄存器内的数据以数值的形式显示在 HMI 上
---	--

数据元件属性说明	
位置	显示元件 X/Y 坐标、宽/高
数值位置	显示数值在元件内部相对位置
显示边框	设置显示边框与否
显示表达式	表达式作为显示数据来源
数据类型	二进制数/十进制有符号数/十进制无符号数/十六进制数/浮点数
整数位数	默认 4 ，范围为 1-11 ，如果是二进制数则范围为 1-32 位
小数位数	默认 0 ，范围为 0-11 ，只有十进制数/浮点数有效
无符号数最大值与最小值	通过整数位数和数据类型控制实际的最大值最小值，设置的最大值最小值必须在实际的最大值最小值范围内
动态最大值与最小值	通过表达式设置变量最小值最大值
显示正号	显示时为正数时，有正号
整数位数位数不足时补零	整数位数位数不足时补零，二进制除外
密码显示	当选择时，显示时显示*号
字体设置	设置字体、字体、大小、颜色、粗体、斜体、有无背景以及背景颜色、显示方向、对齐方式、行间距、列间距、边距和多行对齐。
水平/垂直对齐方式	左对齐/居中/右对齐
自动换行	当数值超出范围时，可以换行
使用背景图片	设置使用背景图片与否，目前软件支持的格式包括 BMP/PNG/JPG/TIF/GIF/PNM/PCX/ICO/CUR/ANI/TGA/XPM 。

	<ul style="list-style-type: none"> ◆ 当数据类型选择十进制数时，如果小数位位数不为 0，表示显示值根据实际值偏移小数位，比如值为 100，小数位为 2 时，显示为 1.00；当数据类型选择为浮点数时，小数位数就是实际需要显示的小数位的个数 ◆ 如果数值超过了最大值最小值范围，则显示*号 ◆ 如果数值无效，显示#号 ◆ 动态/静态最小值最大值的数据类型都和数据类型属性项一致 ◆ 界面元件的行间距最大不超过 HMI 屏幕的高度和列间距最大不超过 HMI 屏幕的宽度 ◆ 显示元件为浮点数时，输入范围：最多输入 11 位整数和 14 位小数，同时整数位加小数位不超过 15 位；可输入的最小值和最大值受到整数位数、小数位数的限制,显示精度为 6 位
---	---

	<p>文本元件是从指定寄存器地址中读取的数据按字符串编码格式解析后以字符串的形式显示的元件。</p>
---	--

文本元件属性说明	
文本变量	必须为字符串变量
密码显示	设置显示用*号

注：其他功能与数据元件或者标签元件类似，不做详细介绍。

3.3.2.6 数据报表的属性窗口

	<p>数据报表元件主要是用报表的形式将数据库中的内容显示出来。</p>
---	-------------------------------------

数据报表元件属性说明				
数据来源	分为变量和数据库，数据库可选择本地数据库和远程数据库			
表格设置	设置边框、分割线、标题栏和内容栏相关的颜色和线型宽修改			
滚动条	可以修改滚动条宽度，范围是 20 到 120			
数据报表表名	只在数据源是数据库时有效，选择需要的数据库表			
数据报表资料	基本设置	字体设置	设置标题栏和内容栏字体，也可以使用文本库	
		总条数	总条数变量：数据库来源为变量时有，且必须选择。	
		触发设置	当设置的位变量满足条件时，查询数据。	
			OFF->ON	只有当指定位变量的状态由 OFF 变为 ON 时，才触发查询数据
			ON->OFF	只有当指定位变量的状态由 ON 变为 OFF 时，才触发查询数据
			OFF<->ON	只有当指定位变量的状态变化时，才触发查询数据
	OFF->ON(自动复位)	只有当指定位变量的状态由 OFF 变为 ON 时，才触发查询数据，触发后自动将指定位变量状态复位		

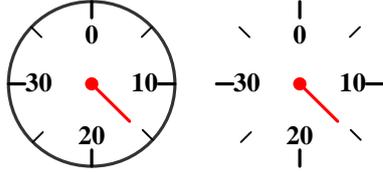
			ON->OFF(自动复位)	只有当指定位变量的状态由 ON 变为 OFF 时,才触发查询数据,触发后自动将指定位变量状态复位	
			ON(复位为OFF)	只有当指定位变量的状态为 ON 时,才触发查询数据,触发后自动将指定位变量状态自动变为 OFF	
		列的列表项	<p>1. 数据源是数据库时默认显示选择的数据表的所有字段,数据是变量时需要手动添加,并提供添加,删除等按钮。</p> <p>2. 双击某一行,弹出列选项设置对话框,可以设置列宽、颜色和标题栏的相关属性,以及数据类型和数据偏移(或小数点显示)</p> <p>3. 当选择 2 个以上字段时,点击列批量修改可以批量修改列宽、对齐方式和颜色</p>		
	查询设置	查询结果设置	全部显示	查询结果全部显示	
分页显示			每页显示条数	可设置	
			总页数	通过数值元件显示	
			当前页	通过数值元件设置,可以更改	
排序		设置排序与否,排序分为固定排序与动态排序			
使用自定义 SQL 语句		自定义语句时要查询的字段必须跟输出变量的字段匹配,符合 sql 语法。当使用 SQL 语句时,排序和查询条件不可以使用			
查询条件		不设置设置查询条件	查询无条件		
		设置条件查询	固定查询	查询条件固定	
动态查询			查询选项变量: 0 表示不使用, 1 表示与, 2 表示或; 查询通道变量: 按照列选项默认排序除序号以 0 为顺序的排下去; 查询类型变量: 0 大于 1 大于等于 2 等于 3 小于等于 4 小于 5 不等于 6 相似;		
动态颜色		设置使用动态颜色与否,动态颜色分为按单元格显示和按行显示			
触控动作	读取触控所在的行变量和列变量,可以把触控中的行变量保存到变量中。				

	<p>1.动态排序：排序类型变量必须为位变量，1表示从大到小 0表示从小到大；排序通道变量是按照列选项默认排序除序号以 0 为顺序的排下去</p> <p>2.数据来源为变量时，必须设置总条数变量，且以实际的总条数变量与设置的总条数变量的最小值为准，当为 0 时不显示</p> <p>3.数据来源是数据库时，需先在报表数据库表中建立好报表</p> <p>4.固定查询与动态查询可以混合使用</p> <p>5.手动点击报表时，按照设置的触控选项把选中行列写入变量，并把选中内容写入关联变量，并设置选中颜色。若没选中，则把行列变量清 0，并去掉选中颜色。在报表外点击不修改选中项。</p> <p>6.手动修改行，列变量也可以控制报表的选中。</p> <p>7.报表刷新时，若当前有选中，则若选中内容没有变，则依然保持选中状态，若有改变（包含隐藏列），则设为不选中。</p> <p>8.字符串类型的字段支持按照时间格式显示，但是字符串格式必须为 <code>xxxx-xxxx-xxxx xx:xx:xx:xx</code></p> <p>9.选中行规则：如果刷新后数据总行数没变，则不管之前选中行的内容有没有变化，还是选中当前行；如果总行数有变化，则判断时先判断上一次选中行号的内容是否变化，若不变，则还是选中该行；若变化，则从当前页第一行开始遍历，最先匹配上次选中行内容的行作为当前选中行；若当前页都没有，则选中上次选中行号；若当前页数据行数已少于上次选中行号，则选中最后一行。</p> <p>10.数据报表元件查询字符串是数据库显示的，数据库动作查询是数据库内部实际的</p>
---	---

3.3.2.7 仪表盘的属性窗口

	<p>仪表盘元件可以将指定 HMI 内部寄存器或 PLC/控制器寄存器的数据以仪表图的方式显示出来</p>
---	---

仪表盘元件属性说明	
读取变量	设定表针元件读取变量的数据类型，支持有符号十进制数、无符号十进制数、单精度浮点数、双精度浮点数等数据类型
仪表盘形状	可通过开始角度和结束角度来设置
仪表盘范围	可通过设置最大值最小值确定
刻度显示位置	分为刻度显示在坐标轴外和内两种
主刻度数	设置表针元件主刻度的数目，2~100 可设
主刻度长度	设置表针元件主刻度的长度。
次刻度数	设置表针元件次刻度的数目，1~100 可设
次刻度长度	设置表针元件次刻度的长度。
显示轴	设置刻度标注显示与否以及其显示颜色

	 <p>显示刻度外框 不显示刻度外框</p>
显示标签	设置刻度的字体以及刻度标注的小数位数，0~4 可设
表针	设置表针样式（8 种可选），宽度，长度，颜色
显示轴心	勾选表示需要设置仪表盘的轴心大小及颜色
使用上下限报警	设置报警的上下限以及颜色等(上下限支持变量表达式)

3.3.2.8 PDF 控件的属性窗口

	<p>PDF 控件主要用来打开 PDF 文件，方便查阅类似作业指导书一类的文档。PDF 元件在控制界面的显示大小代表了在 HMI 上的显示大小，不可太小，否则显示不全。</p>
---	--

PDF 控件元件属性说明	
存储路径	设置 PDF 的存储路径，默认在 /home/user 文件夹下
文件名称	可以使用固定文件名和变量文件名，文件名称使用变量时，必须为字符串变量（字符串长度需设置）
显示模式设置	可以固定模式和变量模式， 0 （拉伸）：不按比例缩放，拉伸到元件大小； 1 （原始大小）：按原始大小显示，超出元件大小后显示滚动条； 2 （填充）：等比例缩放， pdf 的最小边适应元件范围以达到填充元件范围效果，另一边出现滚动条。 3 （适应）：等比例缩放， pdf 的最大边适应元件范围，保持 pdf 比例的同时最大化显示图片；变量值改变后自动切换显示模式，如果变量值不在这个范围，则不改变显示模式。
刷新变量	不能为空，必须为位变量，位变量为 ON 时刷新
滚动条	可以修改滚动条宽度，范围是 20 到 120
显示控制	通过变量显示 PDF 的上一页、下一页、首页、末页和总页数

3.3.2.9 时间的属性窗口

	<p>时间元件是将 HMI 内部 RTC 时钟的数据按设定格式显示出来。时间元件只能读取 HMI 内部 RTC 时间，并不能修改。</p>
---	---

时间属性说明	
显示日期	设置是否显示日期
	日期格式 四种格式可选。其中 D 表示日， M 表示月， yyyy 表示年
	日期分隔符 三种格式可选。如 2011.01.01

	显示 4 位年份	选中年份可以显示 4 位, 如 2011 年则显示“2011”; 不选中, 则只显示 2 位, 即“11”
	日月消零	选中则日期和月份显示前消零, 如 6 月, 选中时显示为“6”; 不选中时显示为“06”
显示星期	设置是否显示星期。星期显示使用 dddd 缩写	
显示时间	设置是否显示时间	
	时间格式	两种格式可选。其中 HH 表示时, MM 表示分, SS 表示秒
	12 时制显示 (AM/PM)	选中后时间按 12 小时制显示。如时间为下午 3 点, 则显示“3: 00 PM”; 不选中则显示为“15: 00”

3.3.2.10 曲线图的属性窗口

	曲线图元件从指定的 HMI 内部字寄存器或 PLC/控制器字寄存器中按触发方式读取一堆连续寄存器的数据并且以曲线的方式绘制出来。当每个取样周期结束时, 新的数据会从指定寄存器中读出来并以曲线图显示, 其显示具有实时性
---	--

曲线图属性说明				
曲线图类型	连续(变量)	曲线不断向前延伸, 先前采样数据保留, 每个通道每次只能采集一个寄存器的值		
	覆盖 (变量/数据库)	不保留先前的采样数据, 每次只显示当前采集的值, 每个通道每次采集多个寄存器的值 (或者从数据库中读取)		
总点数	只对连续类型有效, 显示的超过总点数会自动覆盖最早的数据, 范围 2-40000			
每页的采集点数	每页画面显示的采样点数目, 可以静态或动态, 当为动态时, 采集点数小于 2 时按 2 显示, 大于采集点数, 按采集点数显示, 范围 2-1000, 采集点数必须小于总点数			
清除变量	位变量, 置为 ON 时触发, 触发后自动复位, 触发后清除缓存区的数据。			
X 轴类型	无	X 轴不显示数据		
	时间	显示 X 轴时间刻度, 选择时显示当前系统时间 (或相对时间), 只对连续类型有效		
		显示坐标轴	显示 X 轴坐标轴与否	
		显示相对时间	横坐标从第一个采样点开始标识, 且第一个采样点的显示时间为 0	
		时间刻度	时间刻度间隔: 表示每 2 个时间刻度之间的时间间隔, 单位为采样周期; 即当“时间刻度间隔”设置为 3, 则 2 个刻度之间, 间隔了 3 个采样周期, 最终实际显示的刻度数则根据“采样点数”和“时间刻度间隔”自动计算得出, 范围 1-100	
			设置时间刻度长度 (范围 1-100)、风格 (实线、虚线、点和虚线、双点和虚线、长虚线)、宽度和颜色与否	
		采样点刻度	设置显示采样点刻度长度、风格 (实线、虚线、点和虚线、双点和虚线、长虚线)、宽度和颜色与否	
		刻度标签	设置显示刻度标签与否, 以及显示刻度标签属性	
	时间日期	设置显示时间日期与否, 也可以设置时间日期的格式		
	变量	显示 X 轴刻度, 关联一个非数组变量或者数组变量的一个元素, “连续”类型有效。关联一个数组变量或数据库字段, “覆盖”类型有效。需跟 Y 轴保持一致, 即若 Y 轴是数据库类型, 则 X 轴也必须是数据库类型。		
显示坐标轴		显示 X 轴坐标轴与否		

		基准通道	X 轴左刻度标签最大最小值以选择的基准通道的最大最小值来显示。		
		主刻度	设置显示主刻度个数、长度、风格（实线、虚线、点和虚线、双点和虚线、长虚线）、宽度和颜色与否		
		副刻度	设置显示副刻度个数、长度、风格（实线、虚线、点和虚线、双点和虚线、长虚线）、宽度和颜色与否		
		刻度标签	设置显示刻度标签与否，以及显示刻度标签属性		
		刻度	刻度显示的整数位数和小数位数（或数据偏移）		
显示 Y 轴左刻度	勾选表示显示 Y 轴坐标左刻度				
	显示坐标轴	显示 Y 轴坐标轴与否			
	基准通道	Y 轴左刻度标签最大最小值以选择的基准通道的最大最小值来显示。			
	主刻度	设置显示主刻度个数、长度、风格（实线、虚线、点和虚线、双点和虚线、长虚线）、宽度和颜色与否			
	副刻度	设置显示副刻度个数（ 1-10 ）、长度、风格（实线、虚线、点和虚线、双点和虚线、长虚线）、宽度和颜色与否			
	刻度标签	设置显示刻度标签与否，以及显示刻度标签属性			
	刻度	刻度显示的整数位数和小数位数（或数据偏移）			
显示 Y 轴右刻度	与显示 Y 轴左刻度类似				
显示网格	可以设置网格的行数（ 1-100 ）、列数（ 1-100 ）、背景色、风格、宽度和颜色				
使用滚动条	显示水平滚动条，滚动条宽度可自行设定， 20-120 范围可设，单位为像素点				
曲线图数据	基本设置	排序	只对覆盖类型（数据库有效），排序方式见数据报表部分设置		
		通道列表项	要显示的曲线的数目，最多可设 64 个通道，对通道属性进行增加、删除和排序操作。		
			X 轴	X 轴的变量：选择关联的变量（覆盖类型变量时有效）	
				X 轴的字段：选择关联数据库的字段（覆盖类型数据库时有效）	
				X 轴的上限值：显示的最大值	
				X 轴的下限值：显示的最小值	
			Y 轴	Y 轴的变量：选择关联的变量（覆盖类型变量时有效）	
				Y 轴的字段：选择关联数据库的字段（覆盖类型数据库时有效）	
				Y 轴的上限值：显示的最大值	
		Y 轴的下限值：显示的最小值			
		条件显示	当满足条件时，对应通道的曲线会显示出来		
		通道线	设置显示通道线的风格（实线、虚线、点和虚线、双点和虚线、长虚线）、宽度和颜色		
	节点	设置节点类型（无图形、圆形、三角形、方形、圆形（空心）、三角形（空心）、方形（空心）和交叉线），连接方式（点、直线、X 投影、Y 投影和柱状），大小和颜色			
查询设置	触发设置	当设置的位变量满足条件时，采集数据并显示数据曲线。			
		OFF->ON	只有当指定位变量的状态由 OFF 变为 ON 时，采集数据并显示数据曲线		

		ON->OFF	只有当指定位变量的状态由 ON 变为 OFF 时，采集数据并显示数据曲线
		OFF<->ON	只有当指定位变量的状态变化时，采集数据并显示数据曲线
		OFF->ON(自动复位)	只有当指定位变量的状态由 OFF 变为 ON 时，采集数据并显示数据曲线，触发后自动将指定位变量状态复位
		ON->OFF(自动复位)	只有当指定位变量的状态由 ON 变为 OFF 时，采集数据并显示数据曲线，触发后自动将指定位变量状态复位
	查询条件	只对覆盖类型（数据库有效），查询设置方式见数据报表部分设置	

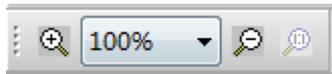
3.3.2.11 饼图的属性窗口

	饼图元件从指定的 HMI 内部寄存器或 PLC /控制器寄存器中按触发方式读取多个寄存器的数据并且以饼图方式绘制出来。当每个取样周期结束时，新的数据会从指定寄存器中读出来并以饼图显示，其显示具有实时性
---	--

饼图属性说明			
显示类型	整圆和扇形		
开始角度	开始角度为通道第一个的开始角度；		
结束角度	只有扇形才有		
显示边框	无选择时不显示		
	选择时，按照设置的线型、线宽、线颜色显示		
显示分割线	无选择时不显示		
	选择时，按照设置的线型、线宽、线颜色显示		
显示标签	显示类别名	无选择时不显示	
		选择时，显示类别名	
	显示数据	无选择时不显示	
		选择时显示数据,数据小数位 0-4 ，有实际意义	
	显示百分比	无选择时不显示	
		选择时数据，设置的百分比小数点 0-4	
	位置	可设置居中、数据标签内和数据标签外	
分隔符	可设置逗号、分号、句号、空格和换行		
字体设置	设置字体的语种、字体、大小、颜色、粗体、斜体、有无背影以及背影颜色、显示方向、齐方式、水平和垂直对齐方式、行间距、列间距、边距、多行对齐。		
饼图设置（饼图数据）	基本设置	饼图中所占的显示数目,最多可设 60 个通道,对通道属性进行增加,删除和排序操作。	
		设置	类别名: 通道显示名称
			数值表达式: 设置该通道显示的表达式设置

		填充设置	设置通道显示的背景色
			设置通道显示的前景色
			设置通道显示的填充类型
	查询设置（触发设置）	当设置的位变量满足条件时，采集数据并显示饼图。	
	OFF->ON	只有当指定位变量的状态由 OFF 变为 ON 时，采集数据并显示饼图	
	ON->OFF	只有当指定位变量的状态由 ON 变为 OFF 时，采集数据并显示饼图	
	OFF<->ON	只有当指定位变量的状态变化时，采集数据并显示饼图	
OFF->ON(自动复位)	只有当指定位变量的状态由 OFF 变为 ON 时，采集数据并显示饼图，触发后自动将指定位变量状态复位		
ON->OFF(自动复位)	只有当指定位变量的状态由 ON 变为 OFF 时，采集数据并显示饼图，触发后自动将指定位变量状态复位		

3.3.3 画面工具栏



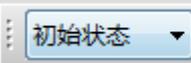
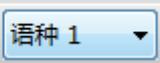
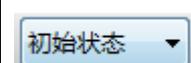
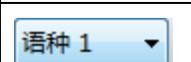
画面工具栏是对组态编辑界面进行放大和缩小

3.3.4 位置调整工具栏



	左/右/上/下移一格，以一个像素为单位
	多个元件的对齐方式，分为左/右/上/下/水平/垂直对齐方式
	多个元件的宽度与长度设置，分为等高、等宽和相同尺寸
	对单个元件进行位置操作，分为置顶、置底、向前一层和向后一层
	对多个元件进行组合和取消组合
	对元件进行组态显示界面的水平居中和垂直居中
	对元件进行水平选择、垂直旋转、90° 旋转和锁定操作

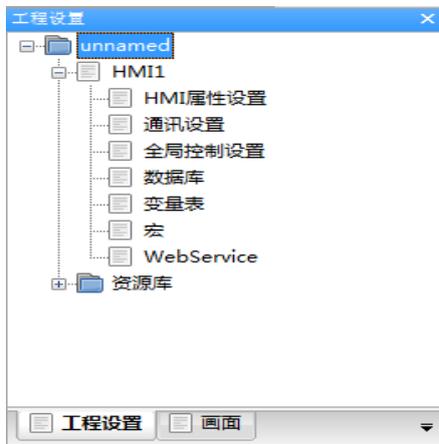
3.3.5 状态切换工具栏

	
	多态显示的状态值，默认为初始状态，状态最大可设置 256 个
	设置组态编辑界面语种显示效果效果，可选择为 HMI 设置中的使用语种个数

3.4 工程设置窗口

工程设置窗口是通过树形结构或预览图的方式来显示当前工程所包含的所有元素的信息。

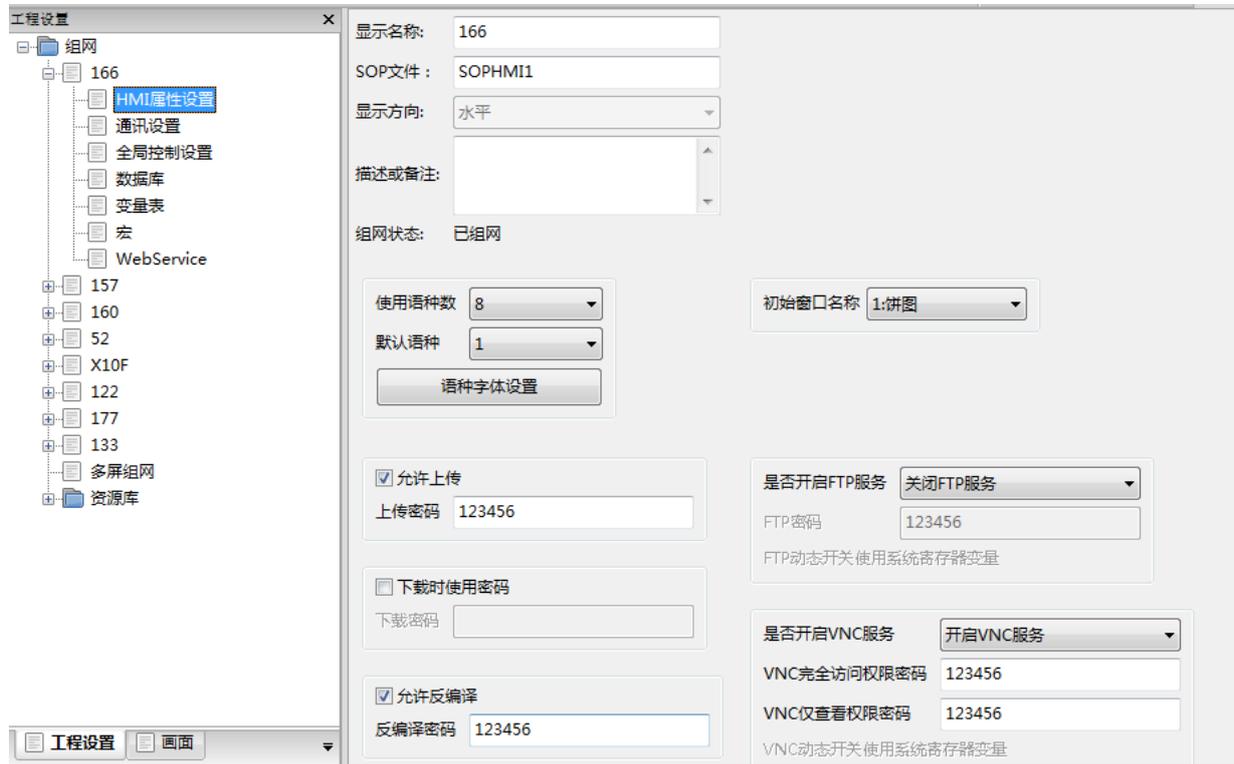
树形结构：将当前工程所包含的所有元素信息以目录树的形式显示出来。



	<p>在工程设置页面</p> <ol style="list-style-type: none">1. 点击工程名字 (unnamed) ,点击右键可以添加 HMI2. 点击 HMI 可以选择删除或者下载 HMI3. 双击 HMI 可以读取所使用的 HMI 型号
---	---

3.4.1 HMI 属性设置

HMI 属性设置界面



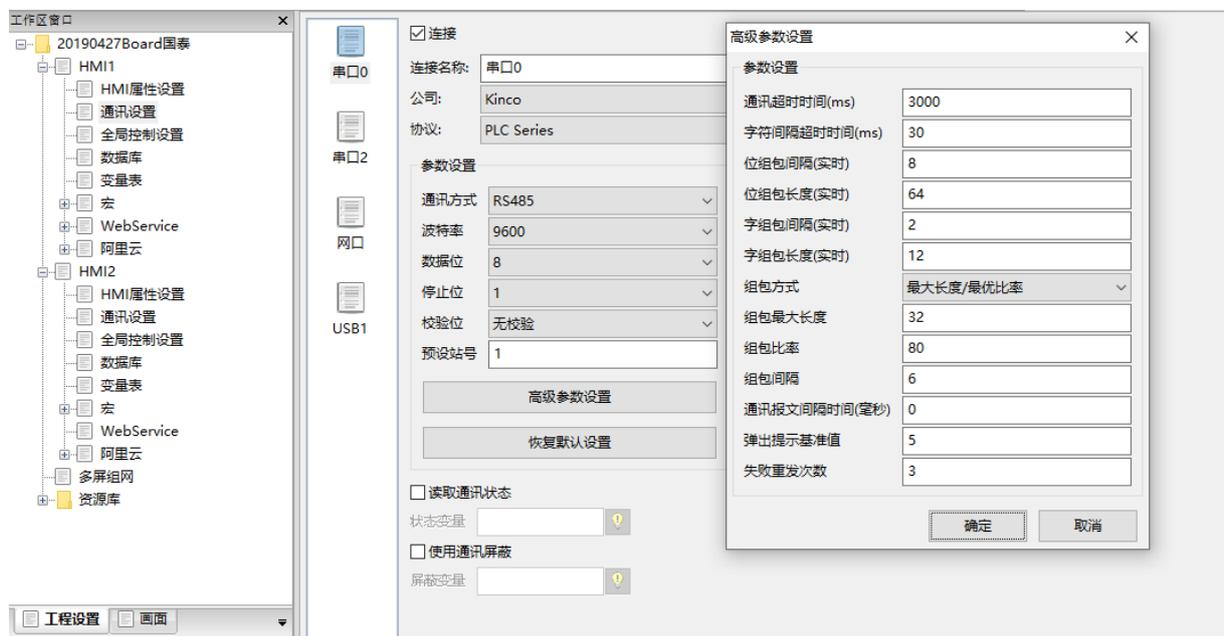
HMI 属性设置界面说明	
显示名称	设置 HMI 名称
SOP 文件	设置 sop 工程名称
显示方向	默认水平，不可以更改
描述和备注	输入对当前 HMI 的信息描述，此描述将在下载、仿真时显示以方便用户区分多个 HMI
组网状态	显示当前屏是不是组网
使用语种数	最多可选 32 个
默认语种	选择显示默认的语种，默认语种数量小于等于使用语种个数
语种默认设置	可以设置对应语种字体属性，也可以快捷把一个语种字体属性复制到另外一个 设置字体的字体、大小、颜色、粗体、斜体、有无背影以及背影颜色、显示方向、齐方式、水平和垂直对齐方式、行间距(最大值 100)、列间距(最大值 100)、边距(最大值 100)、多行对齐。
初始化窗口名称	设置工程下载到组态屏中以及模拟显示的第一个显示画面
允许上传	选择时，KTManager 上传 KBP 时需要输入密码
下载时使用密码	选择时，在 KTDowloader 下载 KBP 和 KTManager 下载 KBP 时需要输入密码
允许反编译	选择时，是上传反编译文件的密码，HMI 的工程才允许反编译，否则无权限
是否开启 FTP 服务	屏做 FTP 服务器使用，默认关闭。默认用户名 ftpuser，路径/home/user/ftp_root

	关闭 FTP 服务	不可以访问屏的地址作为 FTP 服务器
	开启 FTP 服务	可以访问屏的地址作为 FTP 服务器
	动态开关	使用系统寄存器 SYS_START_FTP_SERVER, 为 ON 表示开启, OFF 表示关闭
是否开启 VNC 服务	对屏进行 VNC 操作, 默认关闭	
	关闭 VNC 服务	不允许对屏进行 VNC 操作
	开启 VNC 服务	允许对屏进行 VNC 操作, 可以设置查看和访问密码
	动态开关	使用系统寄存器 SYS_START_VNC_SERVER, 为 ON 表示开启, OFF 表示关闭, 可以设置查看和访问密码

3.4.2 通讯设置

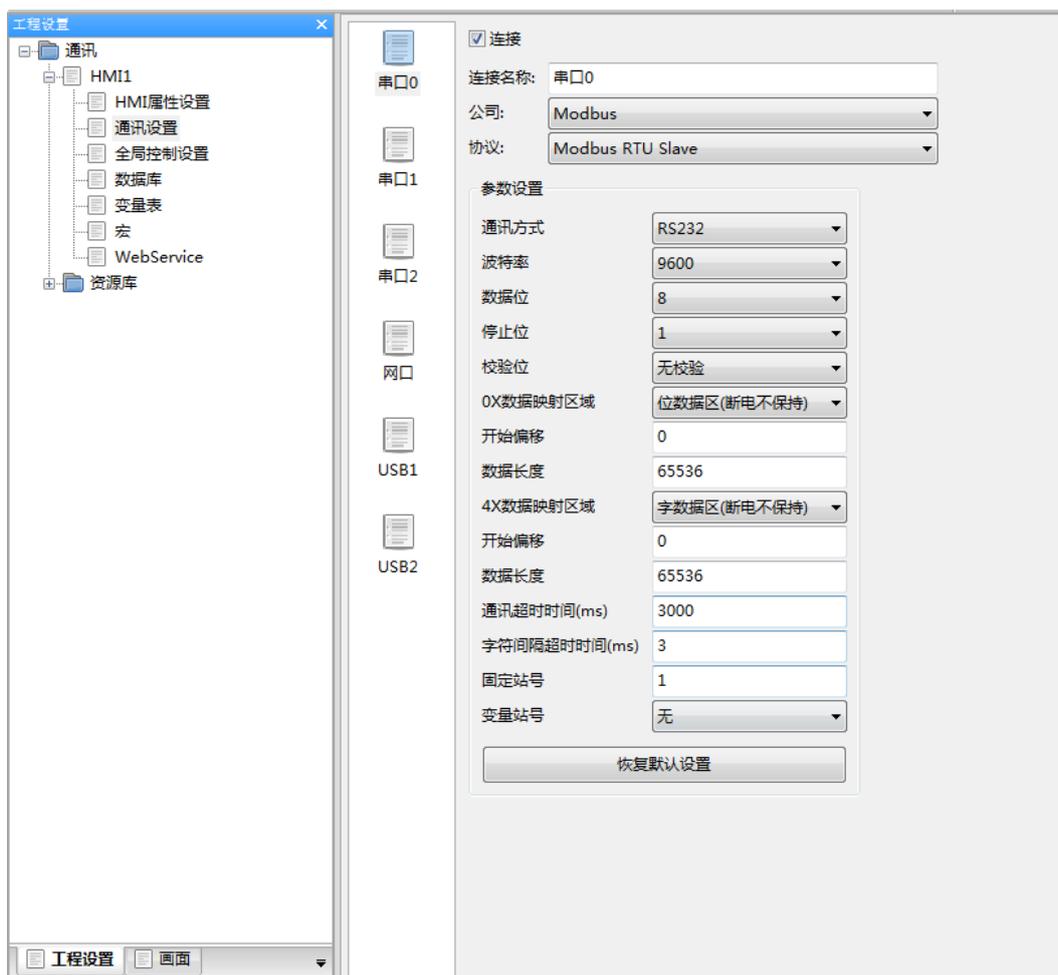
3.4.2.1 串口 0/1/2 设置

在【串口 0/1/2】设置属性页中, 用户可设置 HMI 与 PLC/控制器串口通讯时, HMI 端相应通讯端口的通讯参数。



串口 0/1/2 设置属性页说明 (PLC 主站驱动)	
连接名称	选择 HMI 与所连接 PLC/控制器通信使用的接口传输方式。RS232(RS-232C)、RS485-2(RS-485)、RS485-4(RS-422A) 方式可选。注意: 串口 2 只支持 RS232(RS-232C)接口传输方式
公司	PLC 公司
协议	选择对应公司的 PLC 驱动协议
通讯方式	选择 HMI 与所连接 PLC/控制器通信使用的接口传输方式。RS232、RS485、RS422 方式可选。注意: 串口 2 只支持 RS232 接口传输方式

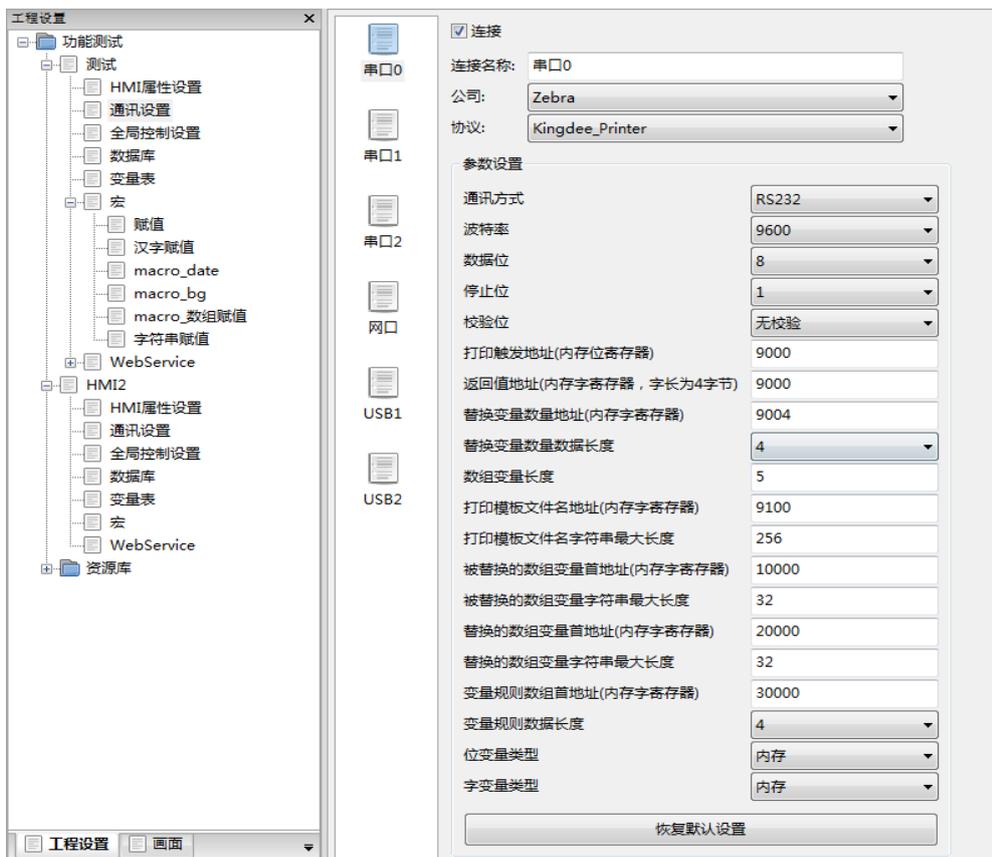
波特率/数据位/奇偶校验/停止位	设置 HMI 与所连接 PLC/控制器串口通信的通讯参数；
预设站号	HMI 作为主设备时，可预设 PLC 的站号，需要更改站号可以在变量属性中更改
通讯超时时间 (ms)	等待 PLC/控制器响应的的时间。PLC/控制器在多久时间内不响应 HMI 的通讯请求，就认为超时，放弃本次请求后重试。如果尝试多次仍不响应 HMI 通讯请求，则弹出 PLC (port:0/1/2)communication error:code 2。该参数不建议用户自行修改
字符间隔超时时间 (ms)	协议以此时间做为断帧。当通讯不稳定的时候，可以改大这个参数试验是否能改善通讯情况。该参数会在用户选择好 PLC/控制器通讯协议后自动设定。该参数不建议用户自行修改
位/字组包间隔 (实时) (或组包间隔)	组包最大寄存器间隔决定：寄存器之间地址不连续时，相差多少才可以拼在一起做为一个包一次性读取。该参数不建议用户自行修改
位/字组包长度 (实时) (或者组包最大长度)	批量传输最大寄存器个数决定：最大组包长度，协议支持最多多少个寄存器可以组成一个帧来一次性读取。该参数不建议用户自行修改
组包方式	普通组包方式：分为无组包、最大长度/最优比率和最大长度/最大间隔三种
组包比率	比率 = 有效寄存器个数 / 总组包寄存器个数 * 100%
通讯报文间隔 (毫秒)	针对单片机可以修改，通讯等待时间；该参数不建议用户自行修改
弹出提示基准值	默认值为 5，只有普通读有效；例子：当值为 1 时，普通读每次通讯错误都弹出提示；如果是 5，连续五次才会报错；该参数不建议用户自行修改
失败重发次数	默认为 3，实时读/写的重试次数，每次实时读/写通讯失败后会自动重发，超出失败重发次数则立即弹出错误提示。该参数不建议用户自行修改
恢复默认设置	如果用户自行修改了默认的通讯参数，从而导致 HMI 与 PLC/控制器通讯出现不可预期的结果时，可通过该设置来恢复系统的默认设置
读取通讯状态	位变量：0-不通讯；1-通讯；数组变量，数组的下标表示 PLC 变量的站号，例如：选择一个长度为 10 的数组变量 variable，则 variable[0]表示站号为 1 的从站的通讯状态；variable[1]表示站号为 2 的从站的通讯状态，依次类推；(通讯口只要有一个地址通讯，状态就是 1)
使用通讯屏蔽	位变量：1 表示屏蔽，0 表示不屏蔽；数组变量，数组的下标表示 PLC 变量的站号，例如：选择一个长度为 10 的数组变量 variable，则 variable[0]表示站号为 1 的从站的通讯状态；variable[1]表示站号为 2 的从站的通讯状态，依次类推；当屏蔽状态为 1 时，通讯状态为 1，原来的通讯值保持原有的值；当屏蔽状态为 0 时，通讯状态实时读取；



串口 0/1/2 设置属性页说明 (PLC 从站驱动)

连接名称	选择 HMI 与所连接 PLC/控制器通信使用的接口传输方式。 RS232(RS-232C) 、 RS485-2(RS-485) 、 RS485-4(RS-422A) 方式可选。注意：串口 2 只支持 RS232(RS-232C) 接口传输方式
公司	PLC 公司
协议	选择对应公司的 PLC 驱动协议
通讯方式	选择 HMI 与所连接 PLC/控制器通信使用的接口传输方式。 RS232 、 RS485 、 RS422 方式可选。注意：串口 2 只支持 RS232 接口传输方式
波特率/数据位/奇偶校验/停止位	设置 HMI 与所连接 PLC/控制器串口通信的通讯参数
0X 数据映射区域	位数据区(断电保存) /位数据区(断电不保存)
开始偏移	对应区域的起始地址
数据长度	区域的总长度
4X 数据映射区域	字数据区(断电保存) /字数据区(断电不保存)
开始偏移	对应区域的起始地址
数据长度	区域的总长度
通讯超时时间(ms)	等待 PLC/控制器响应的的时间。PLC/控制器在多久时间内不响应 HMI 的通讯请求，就认为超时，放弃本次请求后重试。如果尝试多次仍不响应 HMI 通讯请求，则弹出 PLC (port:0/1/2)communication error:code 2。该参数不建议用户自行

	修改
字符间隔超时时间 (ms)	协议以此时间做为断帧。当通讯不稳定的时候，可以改大这个参数试验是否能改善通讯情况。该参数会在用户选择好 PLC/控制器通讯协议后自动设定。该参数不建议用户自行修改
固定站号	当 HMI 作为从设备时，设置 PLC 的站号
变量站号	默认为无，表示使用固定站号；变量站号需使用系统磁盘字寄存器数组 SYS_DYNAMIC_STATION ，[地址 1]表示 SYS_DYNAMIC_STATION[0] ，[地址 2]表示 SYS_DYNAMIC_STATION[1] ，依次类推。 变量站号改变后需要重启屏使之生效。



串口 0/1/2 设置属性页说明 (打印驱动)	
连接名称	串口 0/串口 1/串口 2
公司	Zebra
协议	选择对应公司的打印驱动协议
通讯方式	只支持 RS232
波特率/数据位/奇偶校验/停止位	设置 HMI 与所连接 PLC/控制器串口通信的通讯参数
打印触发地址	地址为内存位寄存器地址，当位地址为 on 时触发打印
返回值地址	内存字寄存器，返回值打印状态:0 准备,1 打印成功,2CDE 读取失败,3 替换变量失败,4 发送文件失败,5 其它。
替换变量数量地址	要替换的变量个数，字寄存器
替换变量数量数据长度	分为 2 和 4，表示 16 位和 32 位

数组变量长度	所用数组变量的数组长度
打印模板数据名地址	要打印的模版文件（后缀名为 prn 的文件）
打印模板数据名字符串最大长度	可输入打印模板数据名字符串最大长度
被替换的数组变量首地址	指被替换的数组变量使用的地址
被替换的数组变量字符串最大长度	可输入被替换的数组变量字符串最大长度
替换的数组变量首地址	指替换的变量数组使用的地址
替换的数组变量字符串最大长度	可输入替换的数组变量字符串最大长度
变量规则数组首地址	目前有 3 种规则： 0 代表标签， 1 代表条形码+标签， 2 代表二维码
变量规则数据长度	分为 2 和 4 ，表示 16 位和 32 位
位变量类型	分为内存、磁盘和系统变量
字变量类型	分为内存、磁盘和系统变量
恢复默认设置	如果用户自行修改了默认的通讯参数，从而导致 HMI 与打印设备通讯出现不可预期的结果时，可通过该设置来恢复系统的默认设置



- 1.被替换的数组变量字符串最大长度与替换的数组变量字符串最大长度必须相等
- 2.打印模板文件必须在 **HMI** 中的 **home/user** 目录下
- 3.数组变量长度与替换变量数量地址取两个中的最小值
- 4.打印模板数据名字符串最大长度、被替换的数组变量字符串最大长度、替换的数组变量字符串最大长度最大不能超过 256

3.4.2.2 网口设置

在【网口】设置属性页中，用户可设置工程与 **HMI** 的通讯地址和网段



- 1.现在界面有四个地方使用 **IP** 地址：通讯设置的网口、**KTDownloader** 的通讯方式、**KTDownloader** 的使用 **IP** 地址、**KTManger** 中的下载 **kbp** 使用 **IP**，**IP** 规则如下：
 - 1.6 软件（不包括 1.5）之前的使用规则

屏重启 **IP**： **KTDownloader** 的使用 **IP**>通讯设置的网口 **IP**>**KTDownloader** 的通讯 **IP**；

屏不重启 **IP** 为 **KTDownloader** 的通讯 **IP**；

屏重启 **IP**： **KTManger** 的设置 **IP**>**KBP** 自带 **IP**>**KTManger** 的通讯 **IP**；

屏不重启 **IP** 为 **KTManger** 的通讯 **IP**；
 - 1.6 软件（包括 1.6）之后的使用规则

保持屏的 **IP** 地址不变：无论重启不重启都保证是屏的 **IP** 地址不会改变

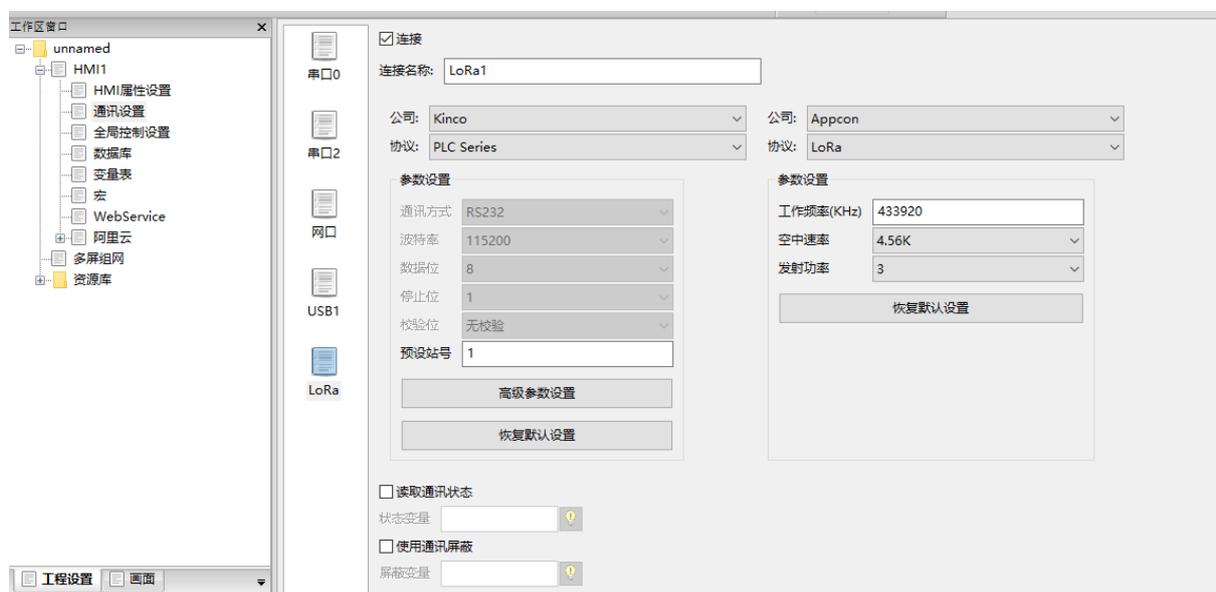
使用工程的通讯设置 **IP** 地址：勾选后，下载后屏的 **IP** 地址应该为使用工程的通讯设置 **IP**，屏会自动重启

下载后使用新的 **IP**：勾选后，下载后屏的 **IP** 地址应该为新的 **IP**，屏会自动重启
- 2.如果中途使用系统设置更改 **ip**，无论重启不重启都为系统设置中的 **ip**
- 3.使用最新的软件（即 1.6 以后的软件），如果屏上版本小于等于 278，选择 **IP** 保持不变，重启后变成工程设置 **IP**，选择工程设置 **IP** 重启后还是工程设置 **IP**，更改 **IP** 重启后还是更改的 **IP**；屏上版本大于 278，符合 1.6 软件（包括 1.6）之后的使用规则
- 4.当网口选择网络协议时，可以使用动态 **IP**，动态 **IP** 变量只能是磁盘寄存器、字符串变量，不能是 **PLC** 变量，为空时为默认的 **plc** 的 **IP**；重启生效；

3.4.2.3 USB1/2 设置

参数设置参照串口打印驱动设置。

3.4.2.4 LoRa 设置



LoRa 设置属性页说明	
PLC 协议部分	通讯方式/波特率/数据位/奇偶校验/停止位默认不可更改，其他根据需要；
LoRa	支持 Appcon 公司的 LoRa
工作频率	410-440MHz 470-510MHz;855-885MHz;800-930MHz;
空中频率	0.81K,1.46K,2.6K,9.11K,18.23Kbps
发射功率	0-7;0-20dBb;
恢复默认设置	恢复 LoRa 的默认参数

- | | |
|---|--|
|  | <ol style="list-style-type: none"> 1. 协议设备必须与本协议支持的一致； 2. Lora 参数设置必须一致； |
|---|--|

3.4.3 全局控制设置

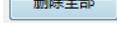
全局控制设置	全局控制在所有画面都有效，触发方式分为周期触发、表达式触发和其他触发。触发顺序分为全部触发和顺序触发（执行失败后自动终止）
--------	---

周期触发说明

周期间隔	隔多久执行一次动作
重复次数	最多触发的次数，0 表示一直触发
使用变量周期间隔	使用变量设置周期间隔
使用变量重复次数	使用变量设置重复次数，每确认一次就重新开始计数
确认变量	变量确认位,这个状态变化后扫描一次变量参数

表达式触发说明	
位变量判断 ON,OFF	OFF<->ON,ON->OFF,OFF->ON,OFF->ON (自动复位), ON->OFF(自动复位), ON, ON (复位为 OFF) 七种方式
字变量判断变化	以上一次触发为基准，变化大于等于幅度时触发，幅度范围 1-65535
表达式为真时触发	当表达式为真时触发相应的动作
字符串变量判断变化	字符串内容和上次内容不一致时就触发
为空时不触发	当字符串为空时不触发
延时时间	当条件满足时，延时时间达到时再执行相应的动作

其他触发说明	
HMI 启动时	当 HMI 启动时，触发相应的动作
点击屏幕时	当点击屏幕时，触发相应的动作

动作顺序说明	
全部触发	所有动作全部执行，不需要等待执行结果返回，不考虑执行失败情况
顺序触发	默认顺序执行动作，等待动作执行结果返回，如果成功则执行下一动作，失败则不再执行
	选择发生失败时继续，等待动作执行结果返回，无论成功和失败成功都执行下一动作
	选择发生失败后触发通知，失败后，通知变量会显示对应的数值
	新增动作
	删除选中的动作
	复制选中的动作，支持工程内全局控制，画面控制和触控控制的复制
	对复制的动作进行粘贴，支持工程内全局控制，画面控制和触控控制的粘贴
	对选中的动作进行上移
	对选中的动作，进行下移
	删除设置的全部动作
	动作全部改成一直使能
	动作全部改成禁止使能

使能设置说明	
一直使能	可以一直触发前面的动作
	本次动作完成前允许再次触发，默认不勾选
禁止使能	无论何种条件都不触发
条件使能	满足表达式时触发前面的动作

当用户选择新建、编辑动作时，弹出动作设置对话框，显示动作分类下拉列表框，然后根据用户选择的动作分类显示具体的动作设置界面。当用户选择赋值动作时，可以在当前对话框选择赋值的动作。具体的动作分类如下。

3.4.3.1 系统操作

关机：此动作用来关闭系统中的某些程序，并非断开 HMI 的电源，主要用于使用数据库相关操作，当使用数据库操作时，先进行此动作然后关掉电源，否则可能会引起数据报表结果的破坏。

3.4.3.2 变量赋值

变量赋值	变量赋值主要指的是对屏上定义的变量进行赋值操作，赋值可以是固定值，也可以是在原有值上进行加减运算，还可以是弹出键盘由用户自己输入值。变量分为非数组变量、数组变量和系统寄存器变量。赋值操作可分为开关量（位）、数字量（有/无符号 16/32 位数，浮点数等）和字符串。下面以非数组变量的赋值操作进行介绍
------	---

开关量（位）类型说明	
ON	不考虑变量当前状态，直接发送置 ON 命令
OFF	不考虑变量当前状态，直接发送置 OFF 命令
切换状态	首先读取当前变量状态，如果为 OFF，则发送置 ON 命令；如果为 ON，则发送置 OFF 命令
OFF->ON（自动复位）	不考虑变量当前状态，先发送置 ON 命令，然后再发送置 OFF 命令，如果变量当前在 OFF 状态，则会有 OFF->ON 和 ON->OFF 两次变化，如果变量当前在 ON 状态，则只会 ON->OFF 一次变化。
ON->OFF（自动复位）	不考虑变量当前状态，先发送置 OFF 命令，然后再发送置 ON 命令，如果变量当前在 OFF 状态，则只会 OFF->ON 一次变化，如果变量当前在 ON 状态，则会有 ON->OFF 和 OFF->ON 两次变化。
复位延时（ms）	选项为 50/100/200/300/400/500，默认 50，用于设置两次命令之间的延时，只有选择带自动复位功能选项时有效。

	复位延时，主要是设定触发时产生的最小脉冲宽度，以利 PLC 控制器接收信号，默认 50。当时间设太短可能 PLC 因为通讯或本身扫描时间太长，导致无法侦测到脉冲而无法正确接收时，为避免这个问题，可适当修改延时时长，以确保 PLC 能接收所有的脉冲。
---	--

数字量（有/无符号 8/16/32 位数，浮点数等）类型说明

赋值	直接对字变量进行赋值操作， KHDesigner 会根据变量类型判断用户输入值是否正确，例如整数类型的数据不能有小数点，输入值不能超过变量上下限范围。		
加（单向）	对变量数据进行递加操作，每次对变量加上递加值，如果变量值大于上限值，则值为上限值，否则为实际值。如果变量初始值超出上限范围，大于上限按照上限处理。 举例说明：假设变量初始值为 0 ，上限值为 10 ，递加值 6 ，则第一次执行时值为 6 ，第二次执行是由于值为 12 超过了上限值 10 ，所以值为 10 ，再次执行时值一直为上限值 10 ；执行流程 0—6—10		
减（单向）	对变量数据进行递减操作，每次对变量减去递减值，如果变量值小于下限值，则值为下限值，否则为实际值。如果变量初始值超出下限范围，小于下限按照下限处理。		
加（循环）	对变量数据进行递加操作，每次对变量加上递加值，如果变量值大于上限值，则值为上限值，然后从下限值开始递加。如果变量初始值超出上下限范围，大于上限按照上限处理，小于下限按照下限处理。 举例说明假设变量初始值为 0 ，上限值为 10 ，下限值为 0 ，递加值 6 ，则第一次执行时值为 6 ，第二次执行是由于值为 12 超过了上限值 10 ，所以值为 10 ；然后从 0 开始。执行流程 0—6—10—0		
减（循环）	对变量数据进行递减操作，每次对变量减去递减值，如果变量值小于下限值，则值为下限值，然后从上限值开始递减。如果变量初始值超出上下限范围，大于上限按照上限处理，小于下限按照下限处理。		
加（到限转向）	对变量数据进行递加操作，每次对变量加上递加值，如果变量值大于上限值，则从上限值开始递减；如果变量初始值超出上下限范围，大于上限按照上限处理，小于下限按照下限处理。 举例说明假设变量初始值为 0 ，上限值为 10 ，下限值为 0 ，递加值 6 ，则第一次执行时值为 6 ，第二次执行是由于值为 12 超过了上限值 10 ，所以值为上限值 10 ，执行 10-6=4 ，再次执行则为 4-6=-2 小于下限值，所以值为下限值 0 ，即 0+6=6 ；执行流程 0—6—10—4—0		
减（到限转向）	对变量数据进行递减操作，每次对变量减去递减值，如果变量值小于下限值，则从下限值开始递加；如果变量值大于上限值，则从上限值开始递减。如果变量初始值超出上下限范围，大于上限按照上限处理，小于下限按照下限处理。		
按位与	将变量数据和指定值进行按位与操作		
按位或	将变量数据和指定值进行按位或操作		
按位异或	将变量数据和指定值进行按位异或操作		
弹出键盘	键盘设置	密码输入	默认不选中，如果选中则输入时输入的数据以****显示
		弹出键盘选择	不弹出键盘（使用按键映射或外接键盘）/使用系统十进制键盘 /使用系统十六进制键盘/使用 ANSI 键盘
		弹出键盘位置	界面上显示 9 个单选按钮，数据对应关系见下图，分别对应屏幕不同位置，默认居中，当使用弹出键盘时有效
	显示提示信息	可以勾选“显示提示信息”，默认提示信息是变量名称，勾选后可以设置具体的提示信息。如果是不使用键盘，则提示信息会灰掉。	
	输入状态	当选择不弹出键盘和选择输入状态时时，光标会出现在输入框中；当选择使用键盘和选择输入框时，光标出现在弹出键盘的输入框中，一个画面只可以有一个输入状态。	
	次序设置	启用输入次序	次序编号，范围 0-65535
	次序使用群组	次序群组编号，范围 1-16	

	次序不循环	当选择时，次序只按从小到大顺序循环一次
数据偏移	默认不选择	弹出键盘值为实际变量
	选择	i 不可以更改，为弹出键盘值,可对弹出键盘值进行加减乘除运算，并把所得值写入变量中
	键盘显示值表达式	键盘显示变量为该变量值，可对变量值进行加减乘除运算
触发通知	可通过变量设置写入前和写入后触发，可以设置字符串为空或数字为 0 时不触发	
输入范围	输入范围	设置输入的最大值最小值（十进制范围）
	可变范围	勾选时使用变量作为最小值最大值范围，如果变量通讯失败，则使用静态最小值最大值。
	整数位数	默认 4，范围为 1-11（以选择数据类型为依据）
	小数位数	默认 0，范围为 0-14，只有十进制数/浮点数有效

1.数字量（有/无符号 16/32 位数，浮点数等）的赋值操作，KHDesigner 会根据变量类型判断用户输入值是否正确，例如整数类型的数据不能有小数点，输入值不能超过变量上下限范围。

2.递加值和递减值必须为大于 0 的整数

3.按位与、按位或和按位异或的值必须为正整数

4.判断键盘输入操作成功与否还是通过对变量赋值进行判断，如果变量赋值成功才算操作成功完成

5.中文输入使用：弹出系统 ANSI 键盘，点击“CH”按钮；设置为不弹出键盘，外接键盘时，按“Alt+Space”切换中英文输入。只能单字输入，不支持词语联想功能。快捷键：“Alt+Space”切换中英文输入；数字 1-9，在有候选词时直接选中对应的候选字；“Page Up”，有候选词时切换上一页候选词；“Page Down”，有候选词时切换下一页候选词；

6.赋值动作的赋值、上限值、下限值输入框：
单精度:输入范围最多输入 11 位整数和 6 位小数;最小值为-9999999999.0;最大值为 9999999999.0;不会补全小数;
双精度:输入范围：最多输入 11 位整数和 15 位小数；最小值为-9999999999.0；最大值为 9999999999.0；不会补全小数；

7.弹出键盘的最小值和最大值输入框：
单精度:输入范围：受到整数位位数和小数位位数限制，同时整数位加小数位相加不超过 11 位；可输入的最小值和最大值受到进制、整数位位数、小数位位数的限制。点确定后补全小数位，保证不超过小数位位数，并且整数位加小数位相加不超过 11 位；
双精度:输入范围：受到整数位位数和小数位位数限制，同时整数位加小数位相加不超过 15 位；可输入的最小值和最大值受到进制、整数位位数、小数位位数的限制。点确定后补全小数位，保证不超过小数位位数，并且整数位加小数位相加不超过 15 位；

8.触发写入前数值与写入后数值：
单精度:输入范围：最多输入 11 位整数和 6 位小数；点确定后显示补全 6 位小数；
双精度:输入范围：最多输入 11 位整数和 15 位小数；点确定后显示补全 15 位小数；

9.单精度浮点数(float)总共用 32 位来表示浮点数，其中尾数用 23 位存储，加上小数点前有一位隐藏的 1(IEEE754 规约数表示法)， $2^{(23+1)} = 16777216$ 。因为 $10^7 < 16777216 < 10^8$ ，所以说单精度浮点数的有效位数是 7 位。考虑到第 7 位可能的四舍五入问题，所以单精度最少有 6 位有效数字（最小尺寸）。同样地：双精度浮点数(double)总共用 64 位来表示浮点数，其中尾数用 52 位存储， $2^{(52+1)} = 9007199254740992$ ， $10^{16} < 9007199254740992 < 10^{17}$ ，所以双精度的有效位数是 16 位。同样四舍五入，最少 15 位。

字符串类型说明	
弹出键盘	和数字量类型相似
字符串赋值	对字符串数值进行赋值，最大可赋值 256 个字符。
字符串清空	对字符串进行清空操作



数组变量实际上就是一组具有相同数据类型和名称的变量的集合。这些变量称为数组的元素，每个数组元素都有一个编号，这个编号叫做下标，使用者可以通过下标来区别这些元素。数组元素的个数也称之为数组的长度。可以用一个变量名和一个固定值下标来表示数组中的一个变量，也可以用一个变量名和一个表达式下标来表示数组中的一个变量，在本质上，数组是内存上一组同类数据列出的一个表。例如数组 **a[i]** 是从编号为 **100** 的存储单元开始存储的，该数组的内容和值如表所示。

变量	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
值	10	11	12	13	14	15
存储单元	D100	D101	D102	D103	D104	D105

1. 在定义数组时，已经定义了数组的数据类型，在访问时需要按照数组的数据类型进行访问和赋值
2. 数组的下标是从 **0** 开始的向后排列整数，不能是其他的下标。第一个下标是 **0** 而不是 **1**，而下标 **n** 表示第 **n+1** 个变量
3. 定义数组 **a[n]** 以后，有 **n** 个元素，但是没有 **a[n]** 这个元素。访问不存在的数组变量时，程序会发生溢出错误
4. 数组下标支持表达式

3.4.3.3 数据传输

数据传输	数据传输可以将指定的 HMI 内部寄存器或 PLC 寄存器地址中的数据传送到其它 HMI 内部寄存器或 PLC 寄存器地址中，可以使用手动触控的方式触发传送数据，也可以利用指定地址的状态改变来触发数据传输
------	--

数据传输界面说明	
源变量	设置源变量
目标变量	设置目标变量

3.4.3.4 切换画面

切换画面	切换画面指的是在 HMI 上根据用户定义的参数显示指定画面的功能
------	---

切换画面界面说明	
指定画面	指定切换到固定画面
上一画面	指定切换到上一次显示的画面，如果当前画面是开机显示的第一个画面，则切换无效

动态画面号	通过表达式的值来确定切换到画面号，若没有此画面号，则切换无效
-------	--------------------------------

3.4.3.5 弹出画面

弹出画面	弹出画面指的是在 HMI 上根据用户定义的参数弹出显示指定画面的功能。弹出画面和切换画面的区别在于切换画面时原画面需要关闭，而弹出画面则不关闭原画面。
------	--

弹出画面界面说明		
画面操作	弹出画面/关闭画面，关闭画面只对弹出的画面有效，其他无效。	
画面号	1.指定画面时可以从画面下拉选择框中选择任意弹出画面， 2.动态画面号则可以使用表达式，通过表达式弹出相应的画面号	
显示位置	使用画面默认位置信息	显示弹出画面时使用画面自身属性里面的位置信息
	居中显示	在屏幕的中心位置居中显示画面
	用指定坐标	用户可以指定 X ， Y 坐标来定义需要显示的位置
	使用动态画面显示坐标	用户可以分别定义 X 和 Y 坐标的表达式，通过运算表达式的值来定义显示位置；如果用户勾选了动态坐标获取失败时使用预设值选项，则上方的 X ， Y 坐标输入框有效，用户可以输入预设的 X ， Y 坐标
显示类型	模式对话框显示	必须在弹出画面关闭画面
	非模式对话框显示	可以在弹出画面和其他画面关闭画面

3.4.3.6 打印

打印	打印动作是对于惠普打印机有效，详细用法见驱动文档说明
----	----------------------------

打印界面说明	
整个屏幕	打印整个屏幕内容
部分屏幕	打印设置的屏幕内容
翻转打印	屏幕内容顺时针旋转 90 度打印

3.4.3.7 执行宏指令

执行宏指令指的是调用用户写的宏来实现一些复杂的逻辑判断，选择该选项时首先判断是否有定义的宏指令，如果没有宏指令则直接提示工程中未定义宏指令，并忽略该选项动作。

3.4.3.8 涂鸦功能

涂鸦功能	涂鸦功能指的是在屏幕上用画笔来随意涂鸦，可以用来给客户做演示说明。
------	-----------------------------------

涂鸦画面界面说明	
工具	画笔跟橡皮，画笔用来涂鸦，橡皮用来擦除涂鸦。
画笔设置	用来设置画笔的粗细跟颜色
清屏	清除真个屏幕上的所有涂鸦
取消涂鸦	清除所有涂鸦并恢复正常状态。注：切换画面后自动取消涂鸦
涂鸦区域	只能在此区域内涂鸦，并且在涂鸦时此区域内的触控动作不响应

3.4.3.9 Webservice 操作

Webservice 操作	调用在工程属性中配置好的 webservice 接口，通过网络访问 webservice 服务器进行数据交互。
---------------	---

3.4.3.10 FTP 操作

FTP 操作	通过网络和服务器连接，上传文件到服务器或从服务器上下载文件到终端设备上。
--------	--------------------------------------

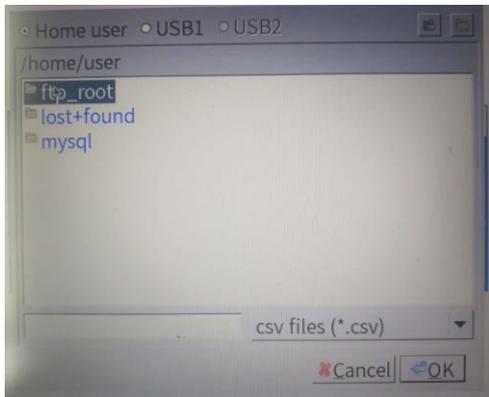
FTP 操作界面说明	
操作	支持上传和下载
服务器地址	支持使用变量和固定输入
端口号	支持使用变量和固定输入
虚拟路径	支持使用变量和固定输入
用户名	FTP 服务器用户名
密码	FTP 服务器密码
文件名	需要上传到 FTP 服务器或下载到 HMI 里面的文件名，支持使用变量和固定输入，选择的编码方式必须和显示的编码方式一致
覆盖类型	当有文件名重复时可以选择覆盖与否
存储路径	选择文件下载和上传后的存储路径，默认路径/ home/user ，也可以是下面的子文件夹
返回值变量	0 ,成功; 1 , 上传的服务器上/下载目录中文件已经存在（不允许覆盖的情况）; 2 , 服务器地址或者端口号错误; 3 , 通讯错误(端口、未通讯错误) ; 4 , 用户名或密码错误; 5 , 上传路径不存在或者下载自动创建目录失败; 6 , 上传或者下载文件不存在或者为空;

	<p>7, 磁盘空间已满;</p> <p>8.其他错误;</p> <p>无论覆盖不覆盖, 先判断文件是否为空; 不覆盖会先判断文件是否不存在; ftp 下载文件不区分大小写, 但是屏对大小写敏感, 如果一个文件分别输入大小写的后缀名, 会下载两个文件。</p>
允许修改文件名称	更改下载后的文件名称, 文件名称后缀名要与下载文件的后缀名一致, 否则文件可能无法使用; 支持使用变量和固定输入; (只对下载有效)
解析名称使用变量	如果文件名字带 FTP 路径, 则直接解析为文件名; 如果设置了修改文件名字, 则与修改文件名称一致; (只对下载有效)

3.4.3.11 数据库操作

数据库操作	对数据库进行插入、修改、删除、查询、创建表、删除表、清空表、用户自定义 SQL 语句、导入、导出和数据传输等操作。
-------	--

数据库操作界面说明				
操作类型	插入	插入一行或多行相同或不同的数据		
		插入类型	分为单个和批量, 批量支持变量	
		插入字段	选择单个或多个插入的字段, 当选择批量和常量时, 默认输入多行相同数据, 插入数据允许为固定值和变量输入。变量可以是数组变量和非数组变量。当为非数组变量时插入为单个数据; 当为整个数组变量时, 插入为整个数组数据 (插入个数必须小于等于数组长度); 单个数组变量时, 插入对应相应下标的数据; 插入个数最大为 1000 ;	
	修改	修改满足条件的一行或多行的数据, 当没选择使用查询条件时, 数据全部修改。并且应用此功能时, 必须选择至少一个字段修改。		
		修改字段	设置所修改的字段	
		使用查询条件	分为固态和动态查询, 可以单独使用和混合使用, 使用方式详细参照数据报表元件查询功能。	
	删除	删除满足条件的一行或多行数据		
	查询	把满足条件的选项输出到变量		
		查询	使用查询条件	当选择时, 按照条件查询
			使用排序	固定排序类型分为从小到大和从大到小排序, 固定排序通道选择标题栏, 动态排序类型和动态排序通道设置变量
			使用自定义 SQL 语句	自定义语句时要查询的字段必须跟输出变量的字段匹配, 符合 sql 语法。当

				使用 SQL 语句时，排序和查询条件不可以使用
		查询结果输出	行数变量	返回查询行数的变量
			按列查询	查询字段设置可以是数组变量，也可以是非数组变量。当非数组变量时为查询字段的第一个数据；整个数组变量时，对应相应的数据；单个数组变量时，对应相应下标的的数据；
			按列查询	1. 把报表数据根据勾选的字段按行查询到变量里 2. 查询报表行数
	创建表	在远程或者本地创建数据库中的表		
	删除表	删除在远程或者本地数据库中的表		
	清空表	清空远程或者本地数据库中的表的数据		
	用户自定义 SQL 语句	自定义语句时要查询的字段必须跟输出变量的字段匹配，符合 sql 语法		
导入		将已有的 CSV 格式的数据采样信息文件导入到当前工程的数据库中		
	导入路径	HMI: 模拟默认为 KHDesigner 目录下，屏上为 /home/user 下，可以选择默认路径下的子目录，路径支持变量输入		
		USB1: 外部设备 USB1 下的路径导入（屏的最外面是 USB1 ）		
		USB2: 外部设备 USB2 下的路径导入（靠近网口的是 USB2 ）		
手动选择: 可以选择用户路径、 USB1 、 USB2 下面的任何 .csv 文件，如下图可以通过双击进入或者退出或者右上角左边退出/右上角右边进入文件夹				
				
	导入文件名	编码格式分为 ANSI 和 UTF-8 （无 BOM ），文件名支持变量输入，仅支持 CSV 格式的文件		
	导出	将当前工程中所有登记的数据采样信息内容导出到指定位置并生成 CSV 格式的		

		文件，此文件可用 Microsoft EXCEL 打开编辑。	
	导出路径	模拟默认为 KHDesigner 目录下，屏上为 /home/user 下，可以选择默认路径下的子目录，路径支持变量输入	
		USB1 :导出到外部设备 USB1 下的路径	
		USB2 :导出到外部设备 USB2 下的路径	
	导出文件名	编码格式分为 ANSI 和 UTF-8 (无 BOM)，文件名支持变量输入	
	使用导出条件	当选择时，输出条件符合的表中的数据。未选择时，输出表中全部数据	
	使用排序	当选择排序时，按照排序条件输出表中的数据	
	用户自定义 SQL 语句	按照 SQL 语句导出	
	数据传输	选择传输字段	全部字段或者部分字段传输
		使用传输条件	当选择时，按照条件查询
		传输完成后删除传输的本地源数据	选择时，传输完成后会删除传输的本地源数据
		使用排序	固定排序类型分为从小到大和从大到小排序，固定排序通道选择标题栏，动态排序类型和动态排序通道设置变量
		使用自定义 SQL 语句	自定义语句时要查询的字段必须跟输出变量的字段匹配，符合 sql 语法。当使用 SQL 语句时，排序和查询条件不可以使用
返回值变量	表示操作是否成功的状态值： 0 ：初始化， 1 ：成功， 2 ：失败。		
源数据库	原始数据库，可以是本地数据库也可以是远程数据库		

	<ol style="list-style-type: none"> 1.用户自定义 SQL 语句在模拟时对大小写不区分，在屏上运行时，应与数据库中的一致 2.导入的文件编码方式必须与选择的一致，导入的文件中的字段名字须与数据库的一致 3.数据传输时字段名和数据类型必须保持一致，否则会造成不可预想的结果，传输字段个数须相等 4.字符串插入可以选择插入系统时间日期 5.界面用到的数据库自定义语句格式 <ol style="list-style-type: none"> a.数据库同步（自定义语句只作为条件：例如"where field>0") b.元件里面的查询（自定义语句只作为条件：例如"where field>0") c.动作里面的查询（自定义语句作为完整语句：例如"select * from table where field>0") d.动作里面的数据传输（自定义语句只作为条件：例如"where field>0") e.动作里面的导出（自定义语句作为完整语句：例如"select * from table where field>0") f.动作里面的自定义语句（自定义语句作为完整语句：例如"select * from table where field>0") g.数据库查询的字段设置里的变量和查询结果输出里的行数变量必选一个。
---	--

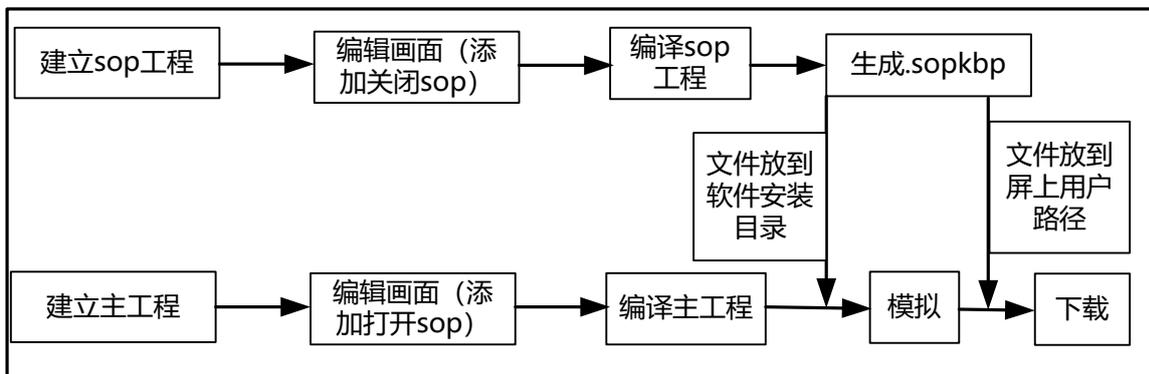
3.4.3.12 SOP 操作

Sop 操作可以快速调用作业指导书以及需要频繁查看的文件。

1.SOP 界面说明

SOP 界面说明	
操作类型	打开 SOP 工程（由主工程使用）：在主工程中设置打开 sop 动作
	关闭 SOP 工程（由 sop 工程使用）：在 sop 工程中设置关闭 sop 动作
SOP 路径	选择文件下载和上传后的存储路径，默认路径/home/user，也可以是下面的子文件夹
SOP 文件名	固定名称：支持固定名（可输入后缀名.sopkbp 或不输入）
	变量名：关联字符串变量，输入 sop 工程名字（可输入后缀名.sopkbp 或不输入）

2.sop 工程使用步骤



主工程和 sop 工程的建立是相互独立的，主工程属于普通的工程，当需要调用 sop 工程的时候，需要先编译好调用的 sopkbp 文件，放到对应的目录下。



- 1.sop 工程不支持有关通讯设置、FTP、Webservice、数据库设置
- 2.sop 工程只支持系统寄存器变量
- 3.sop 工程文件后缀名为.sopkbp
- 4.Sop 工程的初始窗口为 sop 工程的初始画面
- 5.数据报表元件支持数据来源为变量的数据
- 6.Sop 工程文件必须放在/home/user 下面，可通过 KTManager 和 U 盘中的用户路径下载
- 7.PDF、图片、变量数据库如需变量，可使用通用位、字、字符串系统寄存器变量

3.4.3.13 阿里云操作

阿里云操作可以对阿里云进行数据交互

设备	选择阿里云上的设备
通讯属性	可以选择属性和事件
仅发送改变的数据	当选择时，只有属性或者事件的数据发生变化时，才会上传或者下发
QOS	Quality of Service ，服务质量： 0,1,2 。关于 QOS 的介绍可以参考以下链接： http://blog.sina.com.cn/s/blog_a5e78d1d0102wqkr.html
属性	选择需要交互的属性或者事件，可以部分选择或者全部选择

3.4.4 数据库

数据库表	数据库主要是对数据报表的控制，进行保存导出功能的设置
操作	 <p>三个动作分别表示增加、复制、删除数据库。复制可以从本地和远程数据库复制。</p>

数据库列表说明			
表	字段		字段的增加、删除、上移和下移， 上移和下移可以控制表的导出顺序
		字段名称	设置数据库表的字段名称
		字段描述	对字段属性描述，最多 128 个字符
		允许为空	是否字段数据为空
		主键	是否使用主键
		自动递增	必须是主键，只有 16 位， 32 位有此功能
		字段类型	位/ 16 位有符号数/ 16 位无符号数字/ 32 位有符号数/ 32 位无符号数/单精度浮点数/双精度浮点数/字符串/日期时间
	保存设置	表名	设置保存的数据报表名称
		表的描述	对表属性描述，最多 128 个字符
		记录已满时的操作	自动删除最早的记录、不再保存记录
	容量设置	容量单位	天数、条数
		容量	天数为 3650 天，条数为 50000 条
		本地数据库	本机上的数据库
属性	远程数据库	设置数据库类型以及 IP 地址，端口，用户名跟密码，然后更新数据库	
	数据库的使用环境	选择数据库表，可以查看数据库的使用环境	

数据库同步列表界面说明	
循环间隔时间	周期性把本地数据库同步到本地或者远程数据库，周期单位为 s

数据库同步	传输完成后会删除传输的本地源数据	
	选择传输字段	全部字段或者部分字段传输
	使用传输条件	当选择时，按照条件查询
	使用自定义 SQL 语句	自定义语句时要查询的字段必须跟输出变量的字段匹配，符合 sql 语法。当使用 SQL 语句时，排序和查询条件不可以使用

!

1.若本地数据库的数据库表中的字段或者字段类型更改时，请在下载时选择清除本地数据库数据。

2.本地数据库的数据库有动作会自动创建，远程数据库不会自动创建。

3.删除字段或删除全部字段或修改字段名称前，如果有字段在条件或字段设置中使用到，会弹出提示信息。

4.如果删除的字段在某个条件中使用，删除后会删除该条件。

条件包括：数据库动作中的修改条件，删除条件，查询的查询条件，导出的查询条件，数据传输时的查询条件，还有数据库同步的数据传输条件，报表元件和曲线图的查询条件；

字段设置包括：数据库动作中插入字段设置，修改字段设置，按列查询字段设置，按行查询字段设置，数据传输的字段传输设置，以及数据库同步的字段传输设置；

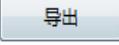
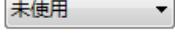
3.4.5 变量表

变量	变量指的是在工程中用户自定义的标记，用来指向在 HMI 运行过程中可变的数 据 。KHD esigner 中的字/双字变量都可以取对应的位作为位变量进行处理。
----	---

新建变量

点击增加按钮，修改变量名称，选择变量类型等，点击确定。

变量界面说明	
变量名称	变量名只允许使用 26 个英文字母、数字以及下划线
	新建变量和批量建立变量
	修改和批量修改变量

	删除和批量删除变量	
	导入*.csv文件，文件格式支持 UTF-8（无 BOM）/ANSI	
	导出*.csv文件，文件格式为 UTF-8（无 BOM）	
	筛选 PLC 寄存器变量、组网寄存器变量、已使用、未使用变量	
变量描述	对变量的属性进行描述	
变量类型	位/16位有符号数/16位无符号数/32位有符号数/32位无符号数/单精度浮点数/双精度浮点数/字符串，16位和32位编码分为 BIN/BCD,字符串分为 ASCII 码/Unicode/UTF8 码	
变量长度	只有字符串才有	
启用数组	当启用数组时，可以设置数组长度	
高低互换	除位、字符串和浮点数外，其他类型都有对应的字节互换、字互换、双字互换	
变量说明	对变量进行说明	
数据来源	可以分为本地变量（HMI 自己的数据）和外部变量（通过通讯连接获取的外部设备的数据）	
寄存器类型	本地变量和外部变量会有对应的字位寄存器类型	
自动分配寄存器地址	自动分配变量的寄存器地址，保证地址不和已建的地址冲突	
寄存器地址	位字寄存器会有对应的寄存器地址	
站号	站号是指 PLC 站点的设置地址，如：012345 等等，它是通讯时用来识别哪一个 PLC 的，每台 PLC 上都设置一个站号地址，这样在通讯时就可以识别相应的 PLC。 从站协议无需设置站号	
批量设置	只有新建变量时才会有此选项	
	批量数量	数量大于 1 时，可以批量建立变量
	地址分配方式	1.相同（所有地址一样）， 2.固定递增（所有地址按照变量类型递增）、固定递减（所有地址按照变量类型递减），指定递增（所有地址按照地址*递增值递增） 3.指定递减（所有地址按照地址*递减值递减）、不变（修改时才有，地址保存原来的不变）
		名称和描述保持不变：只有修改时才有，选中时名称和描述不会发生变化



1. 导入文件字段名称必须与要求的一致，否则无法导入。导入字段名（注意不区分大小写）：

变量名称：**不允许为空**，不符合变量命名规则的无效；

变量描述：允许为空；

变量说明：允许为空；

变量类型：**不允许为空**，允许的字符："位"，"16位有符号"，"16位无符号"，"32位有符号"，"32位无符号"，"单精度"，"双精度"，"字符串(*)"，"8位有符号"，"8位无符号"。注意*表示字符串时的数据长度（字节），空字符表示默认值；其他字符无效；注意不区分大小写；

数组长度：**不允许为空**，空和 0 代表无效，1 代表非数组，大于等于 2 代表非数组，数值代表数组长度；

编码格式：允许为空，空为 BIN。允许的字符："BIN"，"BCD"；注意不区分大小写；其他字符无效；

字符串编码：允许为空，空为 ANSI。允许的字符："ANSI"，"UNICODE"，"UTF8"；注意不区分大小写；其他字符无效；

字节互换，字互换，双字互换：允许为空，空为 0，允许的数字：0，1；其他字符无效；

HMI 名称：允许为空，组网的 **HMI** 需要设置，有效不为空字符时需判断 **HMI** 名称是否有效；

公司_协议：允许为空，空为本地变量，不为空字符时需判断公司协议是否有效；

连接名称：允许为空，空为本地变量，不为空字符时需判断连接名称是否有效；

PLC 站号：本地变量，站号无需填写；

寄存器类型：**不允许为空**，需判断寄存器类型是否有效；

寄存器地址：**不允许为空**，需判断寄存器地址是否有效；

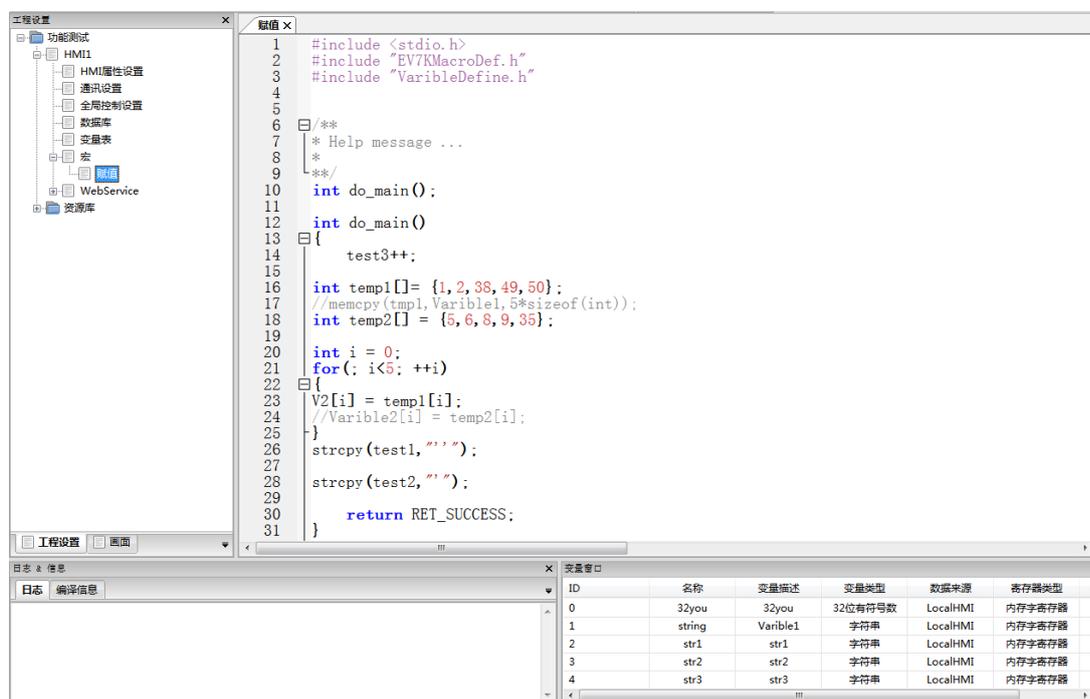
2. 变量导出时，如果遇到有相同名称的变量，会弹出设置框，可以选择替换或者跳过；如果勾选了“统一使用该操作”，则下次再遇到有相同名称变量的情况时直接使用该操作。
3. 如果有导入数据无效，导入结束时提示“已经跳过了错误的导出数据！”，并提示错误数量；在日志框里可以找到错误信息。
4. 选择未使用变量然后选择删除会删除未使用的变量

3.4.6 宏

宏指令使用 C 语言源代码编辑的方式来满足用户逻辑和算术运算等特殊应用。宏指令与相关元件配合使用，可以实现元件无法实现的例如运算或逻辑等功能，使 **HMI** 编程能力更加强大。**KHDesigner** 软件里的宏指令全面兼容标准 C 语言（ANSI C89）。（如果使用 1.6 以后版本的软件编译，宏里面的函数必须有返回值）

3.4.6.1 宏代码编辑环境

如图所示，宏代码编辑环境主要由“宏代码编辑窗口”、“宏代码变量窗口”、工程文件窗口的“宏代码文件列表”、函数选择窗口和函数信息窗口五部分组成。



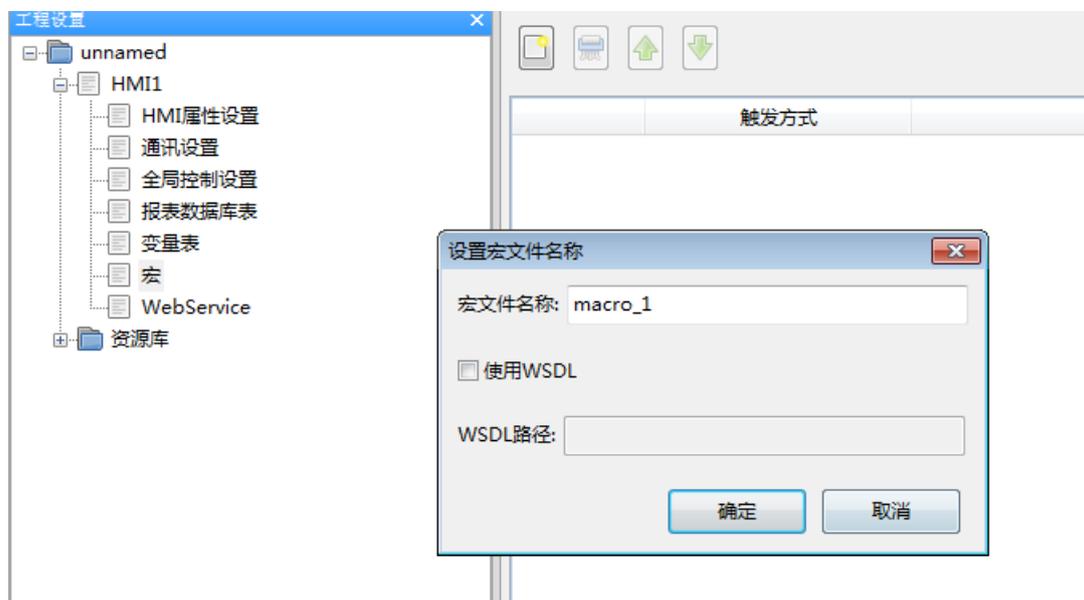
编辑环境各部分功能具体说明如表所示：

宏代码编辑环境具体说明	
宏代码编辑窗口	用户在这个窗口编辑 C 语言代码，宏代码的内部变量在这个窗口已经定义好，可以直接使用
宏代码变量窗口	登录当前宏代码所需的读取和写出变量，即宏代码的外部变量，可以更改在当前宏里面的变量读写属性

3.4.6.2 宏的编辑

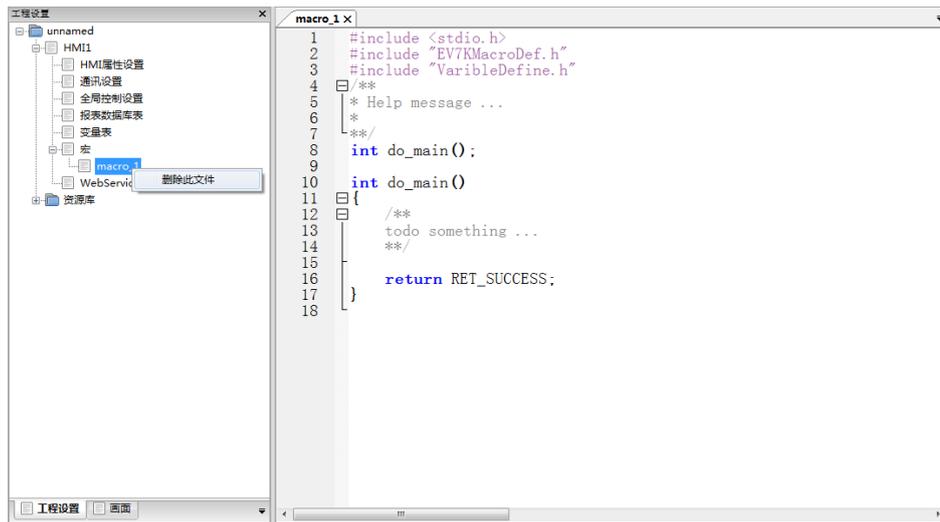
1.创建宏

在【工程设置窗口】选择宏，点击右键，选择添加宏代码，系统自动生成一个文件名(例如 **macro_1**)，用户也可重命名，点击【确定】，进入宏代码编辑窗口，宏文件名一旦确定不可更改。可以使用 **WSDL** 路径调用变量。



1. 删除宏

在【工程设置窗口】相应的宏文件上右击，选择【删除此文件】，可删除选中的宏文件，如图所示：



宏删除后，无法撤销或恢复，请慎用！

3.编写宏

a.在编写宏代码前，首先要定义宏变量，每个变量都有一个名字，有具体的值，并在内存中占据一定的存储单元。



在登记变量时要注意变量地址范围不能重叠，例如使用内存寄存器 0 为起始的寄存器单元登记一个 16 位有符号数类型（字）的变量 A，那么 A 同时占用了内存寄存器 0 和 1 两个地址，如果后续登记变量使用了内存寄存器 1 这个寄存器，那么宏代码运算结果可能会出错。

b.在【宏代码编辑窗口】中编写代码

```

macro_1 x
1  #include <stdio.h>
2  #include "EV7KMacroDef.h"
3  #include "VariableDefine.h"
4  /**
5   * Help message ...
6   *
7   */
8  int do_main();
9
10 int do_main()
11 {
12     /**
13     todo something ...
14     */
15
16     return RET_SUCCESS;
17 }
18

```

- | | |
|---|---|
|  | <ol style="list-style-type: none"> 1. 在代码后增加注释，可便于日后对代码的查询和修改 2. 尽量多使用制表符和换行来编辑代码，以便于程序结构的展阅 3. 宏代码编辑窗口默认语句请勿修改或删除，用户在“return RET_SUCCESS”语句前回车，然后添加定制代码 4. 点击宏名字的右键可以选择修改宏的名字 |
|---|---|

c. 代码编写好后，保存、编译，并检查【编译信息窗口】中是否有提示编译错误的信息

```

编译信息
日志 编译信息
----- 开始编译 HMI1 -----
----- 完成编译 HMI1, 错误:0, 警告:0 -----

```

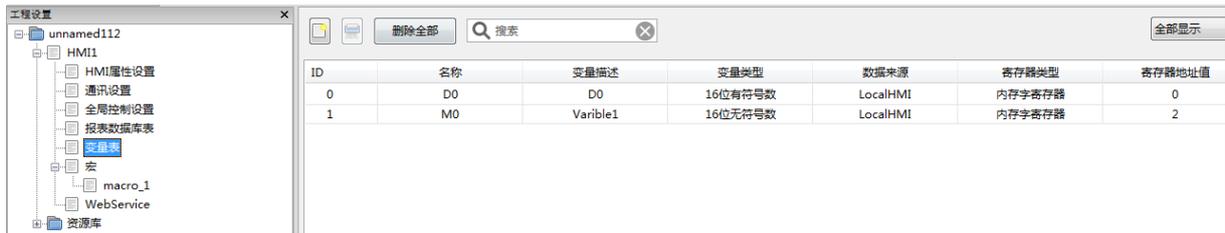
4. 执行宏

KHDesigner 提供了全局控制、画面控制和触控动作三种方式来触发执行宏，用户可根据实际需求选择合适的方式来执行宏。

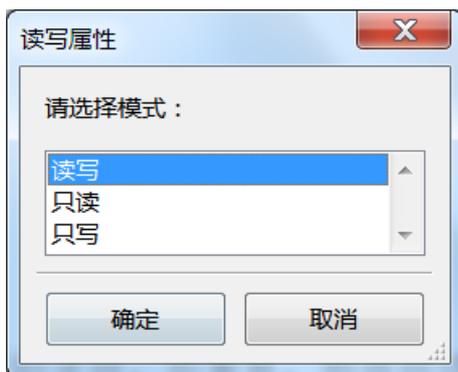
3.4.6.3 宏的应用

【范例】 程序要求当 **D0** 的值为 **10** 时，**M0** 输出 **100**，否则输出 **12**。

1. 首先创建宏，并设置宏名称为“状态输出”；然后在【宏代码变量窗口】中定义 **D0** 和 **M0** 变量：（宏里面正在使用的变量会显示在变量窗口里，并可以修改该宏里面变量的读写属性）



ID	名称	读写属性	变量描述	变量类型	数组描述
0	5a	读写		16位有符...	数组(3)
1	h	读写		16位有符...	非数组
2	5A	读写		字符串(10)	数组(3)



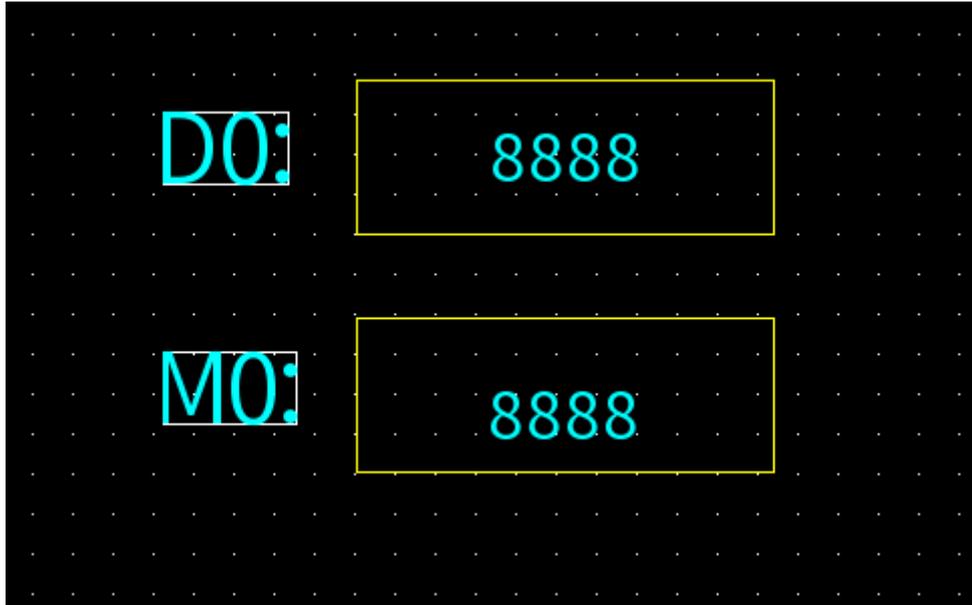
2.在【宏代码编辑窗口】中用 if 语句编写宏代码，代码如下：

```

macro_1 x
1  #include <stdio.h>
2  #include "EV7KMacroDef.h"
3  #include "VariableDefine.h"
4  /**
5   * Help message ...
6   */
7  /**
8   *
9   */
10 int do_main ();
11 int do_main ()
12 {
13     /**
14     * todo something ...
15     */
16     if (D0==10)
17         M0=100;
18     else
19         M0=12;
20     return RET_SUCCESS;
21 }
22
23
24

```

3.关闭宏代码编辑窗口并切换至组态编辑画面，组态画面如图所示,执行宏代码在全局控制中：



4.保存，编译后，运行



- 1.宏代码中所读取的外部变量的值是在宏被触发时一次性读取的，宏运行过程中不会再次或多次实时读取，因此，在宏代码运行期间，从外部改变寄存器的值不会影响代码的运行结果，在这里也提醒用户，要得到正确的宏计算结果，请确保数值输入的正确性，且在数值完全准备完毕的情况下再触发宏；
- 2.宏代码运算的结果是在代码运行完成后一次性写入的，运行过程中宏不会改变外部变量的值，因此，如果要使用代码运算后的结果，请确保宏代码已运行完毕并全部更新了所有写出变量的值。
- 3.只读变量，写操作无意义。只写变量，读出的内容，也是无意义的。如果变量有读取和写出动作，需将变量定义为“读/写”。另外，有写操作属性的变量在代码中必须要赋值。
- 4.宏代码可根据 C 语言相关语法规则定义宏内部的临时变量类型或构造类型，但无法设置全局变量或静态变量。如需使用全局变量保存数据，请使用内部变量地址类型等。
- 5.宏与宏之间不支持相互的调用，如果要执行类似 C 语言中函数的调用，可在代码中设置标志位来触发其它宏运行。

3.4.6.4 宏自带函数

KHDesigner 软件的宏里面有自带的函数，可直接使用，包括 UTF8StrToUnicodeStr、UnicodeStrToUTF8Str、AscStrToUnicodeStr、UnicodeStrToAscStr。

1. UTF8StrToUnicodeStr 是把 utf-8 字符串转换为 utf-16 字符串的函数。

函数形式：`int UTF8StrToUnicodeStr(const unsigned char * utf8_str,unsigned short * unicode_str, int unicode_str_size);`

utf8_str 为要转换的 utf-8 字符串，unicode_str 为保存转换后的 utf-16 字符串的缓冲区，unicode_str_size 为保存转换后的 utf-16 字符串的缓冲区的大小，return:返回实际转换的 utf-16 字符数。

2. UnicodeStrToUTF8Str 是把 utf-16 字符串转换为 utf-8 字符串。

函数形式：`unsigned char* UnicodeStrToUTF8Str(unsigned short * unicode_str,unsigned char * utf8_str, int utf8_str_size);`

`unicde_str` 为要转换的 `utf-16` 字符串, `utf8_str` 为保存转换后的 `utf-8` 字符串的缓冲区, `utf8_str_size` 为保存转换后的 `utf-8` 字符串的缓冲区的大小, `return`: 指向最后一个 `utf-8` 字符的下一个

3. `AscStrToUnicodeStr` 是把 ANSI 字符串转换为 `utf-16` 字符串的函数。

函数形式: `int AscStrToUnicodeStr(const unsigned char* src, unsigned short* dst, int srcBytes);`

`Src` 为要转换的 ANSI 字符串, `dst`: 保存转换后的 `utf-16` 字符串的缓冲区, `srcBytes` 为要转换的 ANSI 字符数, `return` 返回实际转换的字符数。

4. `UnicodeStrToAscStr` 是把 `utf-16` 字符串转换为 ANSI 字符串的函数。

函数形式: `int UnicodeStrToAscStr(const unsigned short* src, unsigned char* dst, int srcBytes);`

`Src` 为要转换的 `utf-16` 字符串, `dst` 为保存转换后的 ANSI 字符串的缓冲区, `srcBytes` 为要转换的 `utf-16` 字符数, `return` 返回实际转换的字符数。

3.4.6.5 宏自带画图接口函数

KHDesigner 软件的宏里面有自带的画图接口函数, 可直接调用。

1. 画笔类型

```
#define PS1_NULL 0 //不画线
#define PS1_SOLID 1 //实线
#define PS1_DOT 2 //点
#define PS1_LONGDASH 3 //破折线
#define PS1_DASHDOT 4 //点+破折线
#define PS1_DASHDOTDOT 5 //双点+破折线
```

2. 画刷类型

```
#define BFS_NOBRUSH 0 //不填充
#define BFS_SOLID 1 //全部填充
#define BFS_BDIAGONAL_HATCH 2 //反向对角分隔
#define BFS_CROSSDIAG_HATCH 3 //交叉对角分隔
#define BFS_FDIAGONAL_HATCH 4 //正向对角分隔
#define BFS_CROSS_HATCH 5 //十字分隔
#define BFS_HORIZONTAL_HATCH 6 //水平分隔
#define BFS_VERTICAL_HATCH 7 //垂直分隔
#define BFS_LINEAR_RIGHT 8 //向右
#define BFS_LINEAR_LEFT 9 //向左
#define BFS_LINEAR_UP 10 //向上
#define BFS_LINEAR_DOWN 11 //向下
```

2-7 属于界面显示的图案填充, 8-11 属于界面显示的渐变填充。

3. 画笔参数

```
typedef struct penparam
{
    short type; //画笔类型
    short width; //画笔宽度 (范围 1-8)
    int color; //画笔颜色: 需用 RGB24(r, g, b)来转换, 传入参数分别是红, 绿, 蓝的 rgb 颜色
}PenParam;
```

4.画刷参数

```
typedef struct brushparam
{
    int type; //画刷类型
    int backColor; //背景色, 设置和画笔颜色设置一样
    int foreColor; ////前景色, 设置和画笔颜色设置一样
}BrushParam;
```

5.点

```
typedef struct point
{
    short x; //x 坐标
    short y; //y 坐标
}Point;
```

6.画图函数

功能: 画直线

```
int DrawLine(int x1, int y1, int x2, int y2, PenParam pen)
```

参数: x1: 起始点 x 坐标; y1: 起始点 y 坐标; x2: 结束点 x 坐标; y2: 结束点 y 坐标; pen:画笔参数

功能: 画折线

```
int DrawLines(Point pts[], int n, PenParam pen)
```

参数: pts: 所有点; n: 点的个数; pen:画笔参数

功能: 画矩形

```
int DrawRect(int x, int y, int w, int h, PenParam pen, BrushParam brh)
```

参数: x: x 坐标; y: y 坐标; w: 宽; h: 高度; pen:画笔参数; brh:画刷参数

功能: 画圆角矩形

参数: x: x 坐标; y: y 坐标; w: 宽; h: 高度; radius: 圆角半径; pen:画笔参数; brh:画刷参数

```
int DrawRndRect(int x, int y, int w, int h, int radius, PenParam pen, BrushParam brh)
```

功能: 画椭圆

```
int DrawEclips(int x, int y, int w, int h, PenParam pen, BrushParam brh)
```

参数: x: x 坐标; y: y 坐标; w: 宽; h: 高度; pen:画笔参数; brh:画刷参数

功能: 画多边形

```
int DrawPolygon(Point pts[], int n, PenParam pen, BrushParam brh)
```

参数: pts: 所有点坐标; n: 点的个数; pen:画笔参数; brh:画刷参数

功能: 画弧线

```
int DrawArc(int x, int y, int w, int h, int start, int end, PenParam pen)
```

参数: x: x 坐标; y: y 坐标; w: 宽; h: 高度; start: 开始角度; end: 结束角度 pen:画笔参数

功能: 画扇形

```
int DrawPie(int x, int y, int w, int h, int start, int end, PenParam pen, BrushParam brh)
```

参数: x: x 坐标; y: y 坐标; w: 宽; h: 高度; start: 开始角度; end: 结束角度 pen:画笔参数; brh:画刷参数

3.4.6.6 宏支持自动更改变量属性的函数

```
strcat(char *destin, char *source);
strchr(char *str, char c);
strcmp(char *str1, char *str2);
strcpy(char* dest, const char *src);
strncpy(const char *s, const char * reject);
strdup(char *str);
strlen(char *s);
strncat(char *dest, const char *src, size_t n);
strncmp ( const char * str1, const char * str2, size_t n );
strncpy(char *dest,char *src,size_t n);
strpbrk(char *str1, char *str2);
strstr(char *str1, const char *str2);
strtok(char s[], const char *delim);
```

```
memcpy(void *dest, const void *src, int c, size_t n);
memchr(const void *s, int c, size_t n);
memcmp(const void *s1, const void *s2, size_t n);
memcpy(void *dest, const void *src, size_t n);
memmove(void *dest, const void *src, size_t n);
memset(void *s, int c, size_t n);
```

```
atof(const char *nptr);
atoi(const char *nptr);
atol(const char *nptr);
strtod(const char *nptr,char **endptr);
strtol(const char *nptr,char **endptr,int base);
```

```
srand(unsigned int seed);
```

```
sin (double);
cos (double);
tan (double);
asin (double);
acos (double);
atan (double);
atan2 (double, double);
sinh (double);
cosh (double);
tanh (double);
```

ceil (double);
floor (double);
abs(int i);
fabs (double);
log (double); 以 e 为底的对数
log10 (double);
pow (double, double);
exp (double);
sqrt (double);
frexp (double f, int *p);
ldexp (double x, int p);
modf (double, double*);
fmod (double, double);
ldiv(long lnumer, long ldenom);
ecvt(double value, int ndigit, int *decpt, int *sign);
fcvt(double value, int ndigit, int *decpt, int *sign);
gcvvt(double value, int ndigit, char *buf);

int puts(const char *str);
char *gets(char *str);
int fputs(const char *str, FILE *stream);
char *fgets(char *str, int n, FILE *stream);

int vfprintf(FILE *stream, const char *format, va_list arg);
int vprintf(const char *format, va_list arg);
int vsprintf(char *str, const char *format, va_list arg);
int fprintf(FILE *stream, const char *format, ...);
int printf(const char *format, ...);
int sprintf(char *str, const char *format, ...);
int fscanf(FILE *stream, const char *format, ...);
int scanf(const char *format, ...);
int sscanf(const char *str, const char *format, ...);

目前判断的运算符如下：

+expr

-expr

expr*expr

expr/expr

expr%expr

expr+expr

expr-expr

!expr

expr<expr
expr<=expr
expr>expr
expr>=expr
expr==expr
expr!=expr
expr&&expr
expr||expr

expr=expr
expr+=expr
expr-=expr
expr*=expr
expr/=expr
expr%=expr
expr<<=expr
expr>>=expr
expr&=expr
expr^=expr
expr|=expr

expr++
expr--
++expr
--expr

~expr
expr1<<expr2
expr1>>expr2
expr^expr
expr&expr
expr|expr

3.4.6.7 宏界面功能功能

宏右键功能	
宏点击右键	添加宏代码
	更新所有宏使用的变量表
	所有宏的读写属性自动配置
宏名字点击右键	删除此宏文件
	修改此宏文件名字（必须打开宏文件才可以修改）
	展示宏的使用环境
	更新此宏使用的变量表
	此宏的读写属性自动配置

	全部选择, 复制等基本操作
宏编辑界面点击右键	选择变量, 点击可以直接在宏里面选择变量
	更新此宏使用的变量表
	此宏的读写属性自动配置
宏支持变量自动补全	宏代码中, 输入字符超过 2 个时, 会判断输入的是否是某个变量名的前缀, 找到的话会弹出自动补全表, 表中显示符合条件的变量名, 选择某个就能补全名称。表中变量名忽略大小写。

3.4.7 Webservice

WSDL 是一个基于 XML 的语言, 用于描述 Webservice 及其函数、参数和返回值。WSDL 文件保存在 Web 服务器上, 通过一个 url 地址就可以访问到它。(连接外网时需要设置屏的 DNS 服务器)

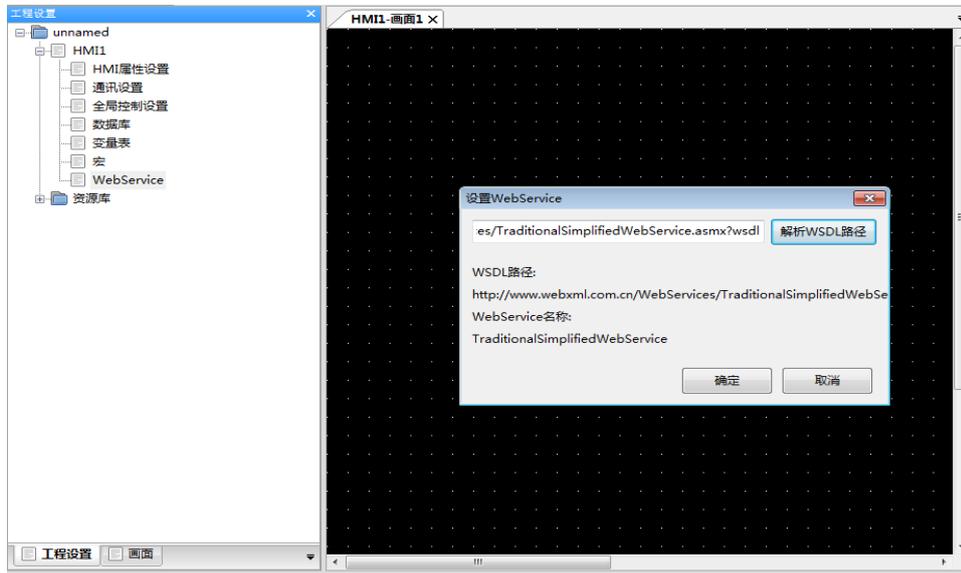
3.4.7.1 Webservice 操作界面

Webservice 操作界面说明		
Webservice 路径	解析过 WSDL 文件路径, Webservice 路径会显示出来	
操作接口列表	当 Webservice 路径确定时, 执行后会自动显示操作接口	
输入元素	用变量关联 Webservice 的输入元素 (支持数组)	
输出元素	用变量关联 Webservice 的输出元素	
操作接口设置	操作接口名称	
	连接超时/接受连接超时/接受超时/发送超时	设置连接、接受和发送等连接超时时间, 超时提示 time out 。
	错误字符串变量	关联返回字符串变量
	错误状态值变量	关联返回状态值变量, 为零时表示成功
	输出元素无返回值时恢复默认值	当输出元素无返回值时, 字符串返回为空, 数值返回零
操作使用环境	可以快速查找 Webservice 接口的使用环境	
输入参数修改	增加	根据提供的 webservice 结构, 添加对应类型的输入参数
	子树增加	根据提供的 webservice 结构, 添加对应类型的子树输入参数
	修改	修改已经建立的输入参数属性
	删除	删除已经建立的输入参数属性
输出参数修改	增加	根据提供的 webservice 结构, 添加对应类型的输出参数
	子树增加	根据提供的 webservice 结构, 添加对应类型的子树输出参数
	修改	修改已经建立的输出参数属性
	删除	删除已经建立的输出参数属性
Webservice 设置	允许设置 Webservice 的路径	选择时, 可以改变 Webservice 路径的 IP 地址和路径
	只更改 IP	只更改 Webservice 路径的 IP 地址
	路径变量	用字符串变量关联 Webservice 的路径
	IP1 变量/IP2 变量/IP3 变量/IP4 变	当使用含有 IP 地址的 Webservice 路径时, 可以使用变量

量	方便更改，当选择只更改 IP 后生效
端口号变量	可以手动改变端口号，当空白时为默认端口号

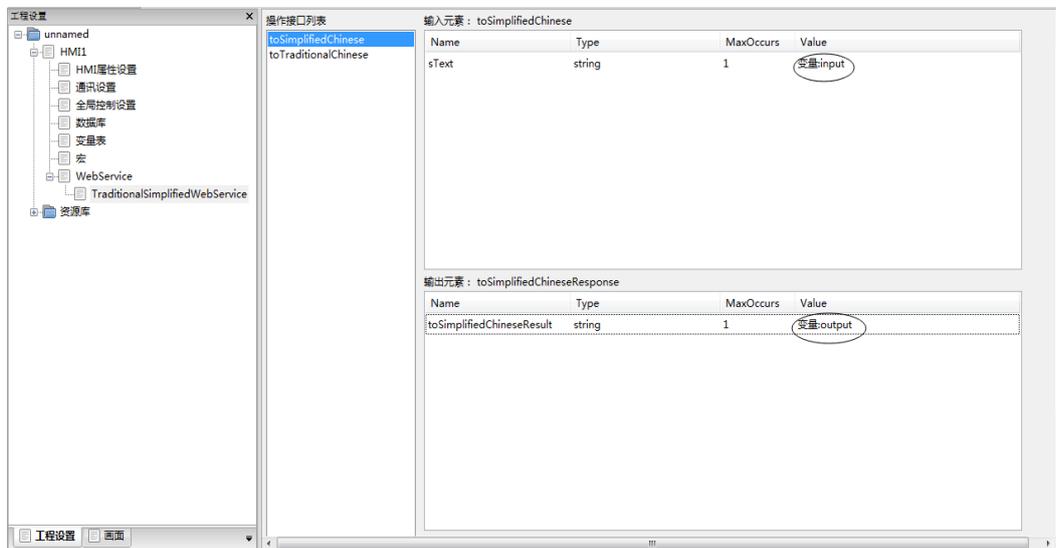
3.4.7.2 创建 Webservice

在【工程设置窗口】选择 **Webservice**，点击右键，这里以转换为简体中文为例，选择新建 **Webservice**，输入 WSDL 路径 <http://www.webxml.com.cn/WebServices/TraditionalSimplifiedWebService.asmx?wsdl>，然后点击解析 WSDL 文件路径，解析成功后 WSDL 文件路径和 **Webservice** 文件名称会自动显示出来，然后点击确定。

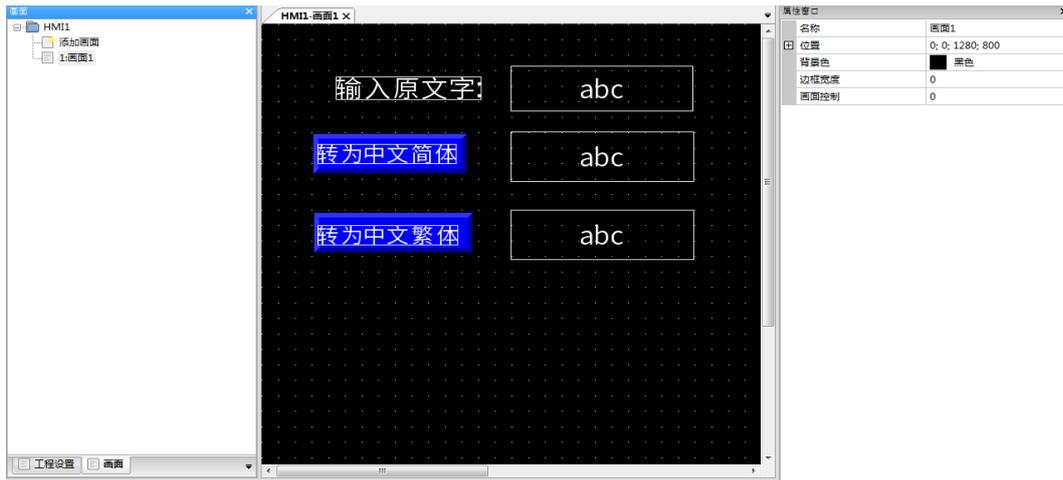


3.4.7.3 编辑 Webservice

解析过 WSDL 文件路径后，双击 **Webservice** 文件名称 **TraditionalSimplifiedWebService** 界面会自动出现操作接口列表，然后通过添加输入输出元素，编码方式必须一致

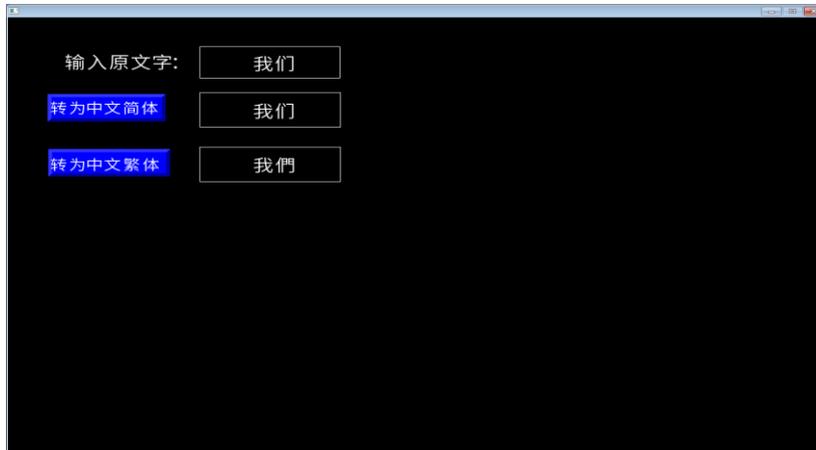


在组态编辑界面，用相应的组态元件显示出来。



3.4.7.4 执行 Webservice

在触控动作中添加执行 Webservice 操作，保存编译后运行，可以模拟显示



3.4.8 HTTP

HTTP 是基于客户/服务器模式，且面向连接的。用于描述 HTTP 及其函数、参数和返回值，通过一个 url 地址就可以访问到它。（连接外网时需要设置屏的 DNS 服务器，暂不支持模拟）

3.4.7.1 HTTP 操作界面

HTTP 操作界面说明	
操作接口列表	暂不支持模拟
	允许增加、修改、复制和删除 http
	操作名称 设置 http 的操作名称,用户自定义

	HTTP URL	<p>输入的 URL (必须包含 http: //)，包括传输协议、存放资源的主机域名或 IP (可能还有端口号)、资源路径 (如果有的话) 等；URL 的格式 协议://用户名:密码@子域名.域名.顶级域名:端口号/目录/文件名.文件后缀?参数=值#标志</p> <p>示范：http://www.kuaidi100.com/query?type=yuantong&postid=1111111111</p> <p>其中 http 是传输协议 (资源类型)； www.kuaidi100.com 是存放资源的主机域名； /query 是资源路径； type=yuantong&postid=1111111111 是查询的参数；</p> <p>2.参数可以直接写到 URL 中，也可以在输入参数中设置</p> <p>3.允许用户设置动态的 URL</p>
	方法	<p>GET 方法：内部处理会把输入参数添加到 URL 中，不支持数组类型的输入参数</p> <p>POST 方法：内部处理会把输入参数组成 json 格式字符串来传递</p>
输入 URL	添加完成后，输入的 URL 会显示到这个地方	
实际 URL	不需要用户修改，由 HTTP 网址生成	
输入元素	用变量关联 HTTP 的输入元素 (支持数组)	
输出元素	用变量关联 HTTP 的输出元素	
操作接口设置	操作接口名称	显示 http 接口名字
	连接超时/接受连接超时/	设置连接 (默认 3s)、接受连接 (默认 10s)超时时间，超时提示 time out 。默认 3 秒
	错误字符串变量	连接不上的时候，获取到的错误字符串信息，可以不设置
	错误状态值变量	连接不上的时候，获取到的错误状态值信息，可以不设置，为零时表示成功
	响应代码变量	返回的 http 状态码， (比如 200 ， 404 等)，可以不设置
	输出元素无返回值时恢复默认值	目前不可以更改
操作使用环境	可以快速查找 HTTP 接口的使用环境	
输入参数修改	增加	根据提供的 HTTP 结构，添加对应类型的输入参数
		参数类型必须和接口数据一样
		JSON 类型包括基本类型，数组类型和对象类型
		子树添加按钮允许数组类型和对象类型的参数添加子参数。、
	子树增加	根据提供的 HTTP 结构，添加对应类型的子树输入参数
	修改	修改已经建立的输入参数属性
	删除	删除已经建立的输入参数属性
输出参数修改	http 返回的数据目前仅支持 json 格式。	
	增加	根据提供的 HTTP 结构，添加对应类型的输出参数
	子树增加	根据提供的 HTTP 结构，添加对应类型的子树输出参数
	修改	修改已经建立的输出参数属性
	删除	删除已经建立的输出参数属性

3.4.8.2 创建 HTTP

在【工程设置窗口】选择 **HTTP**，点击右键，这里以转换为圆通快递为例，选择新建 **HTTP**，输入 **URL** 路径

http://www.kauidi100.com/query?type=yuantong&postid=1111111111 ,



3.4.8.3 编辑 HTTP

1.返回 json 格式:

```
{
  "message": "ok",
  "nu": "1111111111",
  "ischeck": "0",
  "condition": "00",
  "com": "yuantong",
  "status": "200",
  "state": "0",
  "data": [
    {
      "time": "2019-07-25 07:15:47",
      "ftime": "2019-07-25 07:15:47",
      "context": "[上海销售公司新日服务部]进行下级地点扫描，发往：天津分拨中心",
      "location": "上海销售公司新日服务部"
    },
    {
      "time": "2019-07-25 05:02:10",
      "ftime": "2019-07-25 05:02:10",
      "context": "[上海销售公司新日服务部]进行揽件扫描",
      "location": "上海销售公司新日服务部"
    }
  ]
}
```

2.HTTP json 解析

输入输出	参数名称	参数类型	参数说明
------	------	------	------

传入	type	字符串	比如 yuantong		
	postid	字符串	比如 111111111111		
返回	message	字符串			
	nu	字符串			
	ischeck	字符串			
	state	字符串			
	data	数组	参数名称	参数类型	
			time	字符串	
context			字符串		

1.

3.添加输入输出元素，编码方式必须一致

输出元素：

Name	Type	MaxOccurs	MinOccurs	Value
message	STRING			变量:g_message
nu	STRING			变量:g_no
ischeck	STRING			变量:g_ischeck
state	STRING			变量:g_state
ArraySize_data	INT			变量:g_datasize
data				
time	STRING			变量:g_datatime
context	STRING			变量:g_datacontext

在组态编辑界面，用相应的组态元件显示出来。

3.4.8.4 执行 HTTP

在触控动作中添加执行 **HTTP** 操作，保存编译后运行，可以屏上显示（连接外网时需要设置屏的 **DNS** 服务器）

3.4.9 阿里云

阿里云属性说明		
启用阿里云	勾选启用阿里云后，阿里云的功能才有效； 只有三元组的属性与阿里云设置一直时设备才会连接； 右键可以新加产品；	
产品	创建产品时，要选择产品的“节点类型”：设备或网关（默认设备）。	
	产品描述：描述产品属性	
	填写正确的 ProductKey ，可以设置常量或变量；注意和阿里云保持一致	
	Json 格式导入/导出；这里的格式是指从阿里云上导出的模型文件格式，目前只支持导入	
	属性	对设备指标，状态的抽象。如温度、转速。允许增加，修改，删除属性；
	名称	主要用于对属性的意义进行一个可读的命名，长度不超过 30 个字符。
	单位	属性计量单位，如“米”。可不填

	标识符	在对设备属性进行引用时需要的标识，长度不超过 50 个字符。由字母数字字符和下划线组成。要和服务器保持一致。
	数据类型	包括 int, float, double, text, bool 几种类型。要和服务器保持一致。
	数组描述	表示非数组还是数组。要和服务器保持一致。
	属性类型	表示是属性，事件参数还是服务参数。新建属性时确定类型，不允许修改。要和服务器保持一致。
	变量设置	属性关联的默认变量，只有新建时允许设置，表示所有设备上的该属性默认设置的变量。
	批量设置	允许批量新建和批量修改属性。批量新建时可以设置批量数量
事件	相对于属性，事件用于端设备通知云端某事件/动作的发生，并可以携带一些参数同时上报至云端。如开机、关机。 允许增加，修改，删除事件；	
	名称	主要用于对事件的意义进行一个可读的命名，长度不超过 30 个字符。
	标识符	在对设备事件进行引用时需要的标识，长度不超过 50 个字符。由字母数字字符和下划线组成。要和服务器保持一致。
	参数列表	设备跟随事件上报至云端的参数。这里可以直接从属性列表中选择属性，也可以添加事件参数。
	批量设置	允许批量新建和批量修改事件。批量新建时可以设置批量数量。
服务	允许增加，修改，删除服务	
	名称	主要用于对服务的意义进行一个可读的命名，长度不超过 30 个字符。
	标识符	在对设备服务进行引用时需要的标识，长度不超过 50 个字符。由字母数字字符和下划线组成。要和服务器保持一致。
	读写类型	分为同步和异步，同步允许设置输入和输出参数；异步只能设置输入参数。
	输入参数列表	这里可以直接从属性列表中选择属性，也可以添加服务参数。
	输出参数列表	这里可以直接从属性列表中选择属性，也可以添加服务参数。
	批量设置	允许批量新建和批量修改服务。批量新建时可以设置批量数量。
设备	右键可以新建和删除设备、也可以展示设备使用环境	
	设备定义	DeviceName 和 DeviceSecret 可以设置常量或变量；
	心跳周期	服务器每一次心跳的周期，检测设备是否联机。单位秒。
	自动联机空闲时间	与服务器断开时，下一次自动联机的等待时间。单位秒。
	连接状态	设置状态变量（读）。显示屏与阿里云服务器的连接状态。 0 -已停止； 1 -已断线； 2 -已联机。
	启用控制	设置控制位变量（写）。用于控制服务器与屏的连接：勾选后需要手动设置该值，为 ON 时启动联机， OFF 时停止联机。不勾选时默认为联机状态。
	设备属性	属性数据改变时发送：该设备所有属性在属性数据改变时就主动发送。 双击左键设置属性关联的变量，单击右键切换发送模式。
	设备事件	事件数据改变时发送：该设备所有事件在事件数据改变时就主动发送。 单击右键切换发送模式，选择事件可点击显示参数查看该事件的相关参数详细信息。
	设备服务	选择服务可点击“显示输出参数”和“显示输入参数”查看该服务的相关参数详细信息。



- 1.网关类型的产品只允许创建一个。
- 2.创建产品后会自动添加一个设备：“网关”类型的产品默认创建的是网关设备，只能创建一个网关设备；“设备”类型的产品在有网关设备时可以添加多个设备，没有网关设备时只能添加一个。
- 3.如果创建了网关设备，非网关设备将自动全部成为该网关设备的子设备（在组态界面添加描述）。
- 4.使用新软件打开老的工程后，产品类型默认为“设备”。如果老工程使用了多设备，则需要添加一个网关设备，否则编译报错。如果老工程中是单设备，则直接兼容

3.4.10 OPCUA 服务器

Opcua 属性说明	
启用 opcua	暂不支持模拟
	启用 opcua 服务，默认关闭
启用控制	表示控制 OPCUA 的启动和停止；该变量为位变量
节点信息	允许增加（批量增加）、查询、修改（批量修改）和删除（批量删除）群组节点和普通节点，批量设置规则与变量类似
	访问方式默认匿名浏览匿名读取
	允许导入导出节点信息
	界面其他设置不支持，暂不做介绍
OPCUA 软件设置	使用 opcua 软件连接 HMI 时，屏的端口号必须是 4840，否则无法连接

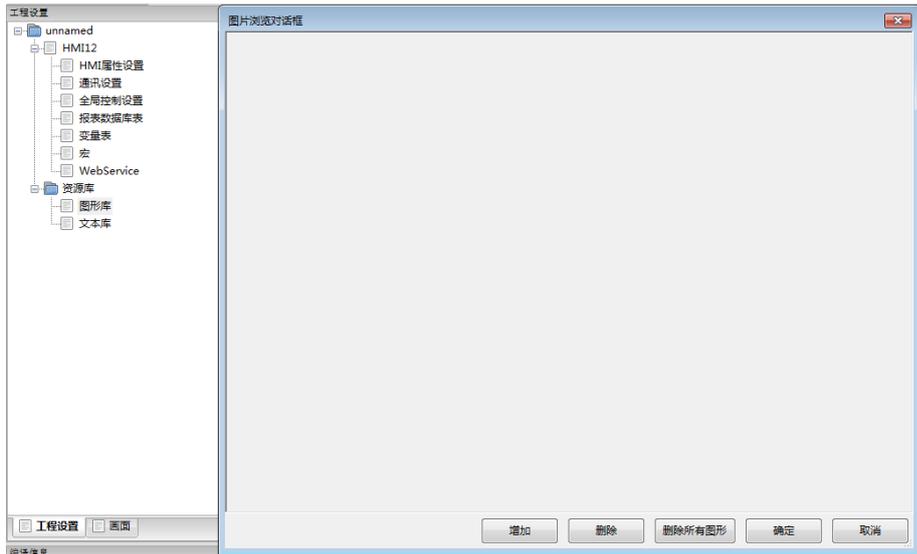
3.4.11 多屏组网

多屏组网	添加可以组网的 IP 地址，只有在组网范围内才可以组网成功。（如果使用 1.6 以后版本的软件编译，组网的工程都必须重新下载）
------	---

3.4.12 图形库

图形库	KHDesigner 可以从软件外部导入位图图片。本节将介绍与图库有关的使用方法。这个图形库只可以使用在文本和数据的背景图片中
-----	---

图形库界面

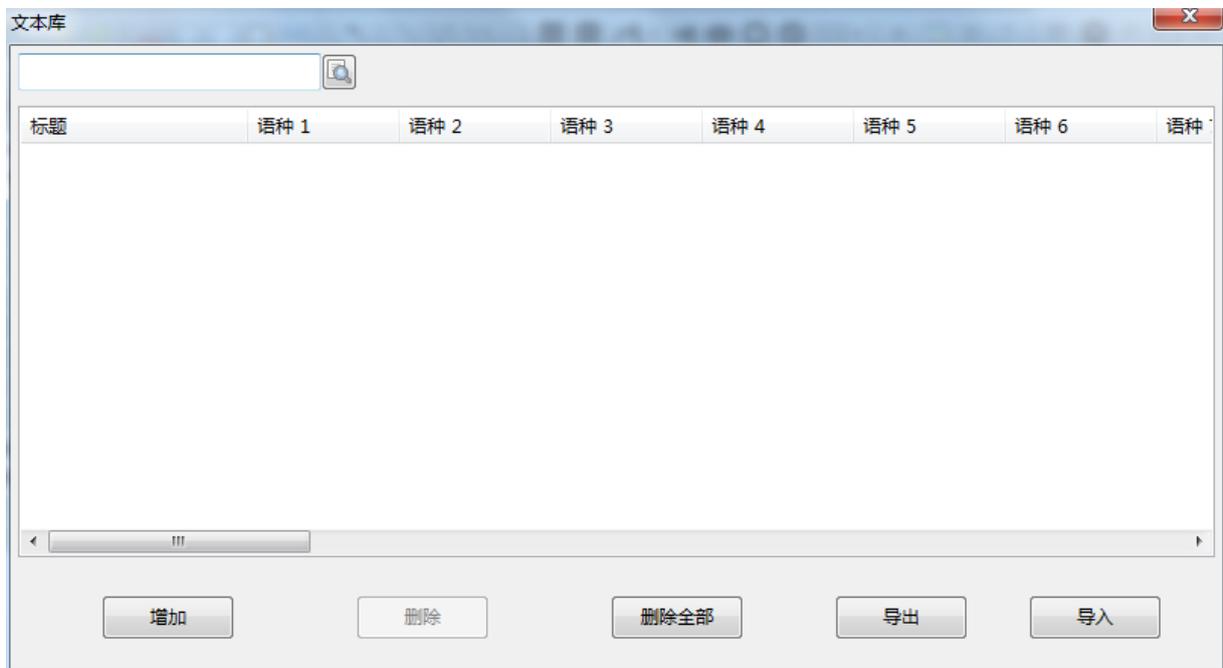


 这个图形库只可以使用在文本和数据的背景图片中，选中所需要的图片后双击有效

3.4.13 文本库

文本库	文本库元件可用于存储工程需要使用的文本内容，从而避免频繁输入相同的文字标签，文本库还可用于需要支持多国语言的环境，从而使组态画面可切换显示不同语言。 KHDesigner 目前最多支持 32 种不同语言的设置
-----	--

文本库界面



文本库查询规则：可以根据文本库名称和里边的内容查询，大小写无关；若查询内容有多条符合点

击查询按钮（多次点击）按行往下搜索到尾会提示是否从头开始搜索；如果没找到会提示没找到。

3.5 画面窗口

画面即组态编辑界面，点击添加画面，可打开该画面所在的组态编辑窗口画面，进行组建工程。右边会出现画面属性窗口，在这里可以设置画面的位置，背景色，边框、画面控制以及选择元件的数量，画面控制跟全局控制的设置方法一样，只是作用范围不一样：画面控制只在当前画面有效，选择元件的数量可通过后面的对话框进行更改。

画面管理窗口中，右键可以展示画面使用环境。使用环境窗口中，其中指定画面是绿色，动态画面和上一画面是灰色；双击跳转到使用的地方。



在画面界面,点击 HMI 可以选择增加、复制或者删除删除 HMI 画面,复制画面范围限制最大值为 999,超过最大值按 999 计算

3.6 编译信息窗口

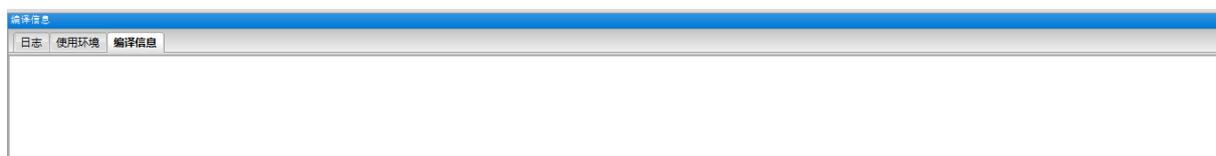
编译信息窗口主要显示工程载入和编译的结果，以使用户准确查找工程存在的错误问题。

日志窗口主要用来显示编辑组态界面所进行操作的步骤。

如果发现异常，会给出两类提示：错误（Error）和警告（Warning）。其中错误指的是会影响屏正常运行的异常，一旦出现错误，则编译肯定失败，用户必须修改掉错误项后才能编译通过；而警告指的是工程中存在的会导致工程运行状态和用户希望不匹配的异常，此类异常一般不会影响屏正常运行，所以可以选择忽略，用户的工程可以编译通过。但强烈建议用户修改工程以避免警告的发生。

双击错误的一行，可以跳转到错误的地方（有些错误不支持）。

使用环境窗口：显示变量、画面、数据库等使用的环境，双击可以到使用的地方；



4.制作工程

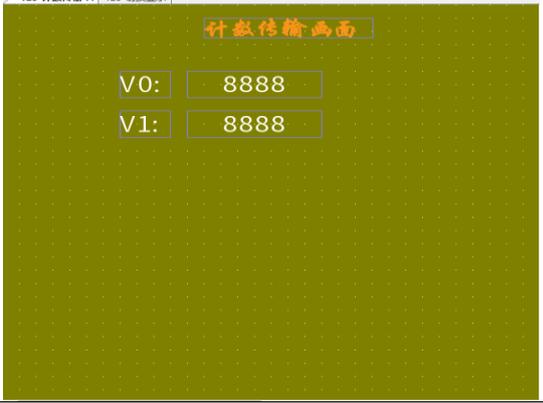
4.1 工程需求

- 上位机系统配置:

软件环境	KHDesigner 组态编辑软件
硬件环境	运行 Windows XP/Vista/7 操作系统并安装有 KHDesigner 软件的个人电脑一台
	X 系列 HMI 一台 (本例中选用 X10)
	DC24V 一个, 为 HMI 供电
	AC220V 一个, 为 PLC (Kinco-K306EX-24AR) 供电
	网线下载电缆一根
	HMI 和 PLC 通信用的 RS-232C 交叉电缆一根 (本例中采用 RS-232C 通信方式)

- HMI 画面规划

计数传输画面



切换显示画面



4.2 制作工程

4.2.1 新建工程

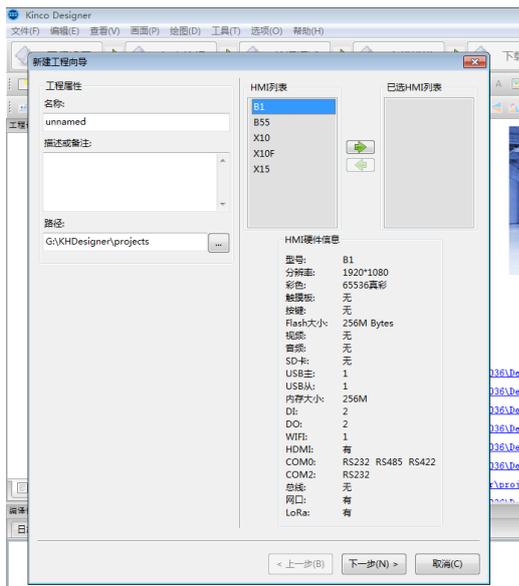
1.启动 KHDesigner

2.单击工具栏中的  图标，新建一个工程

3.输入工程名称；（本例设置工程名称为“F10”）

4.选择工程文件夹保存路径；

5.从“HMI 列表”里选择一个 HMI 设备，点击右边的箭头按钮添加到“已选 HMI 列表”里,更改 HMI 名称，点击确定，点击下一步

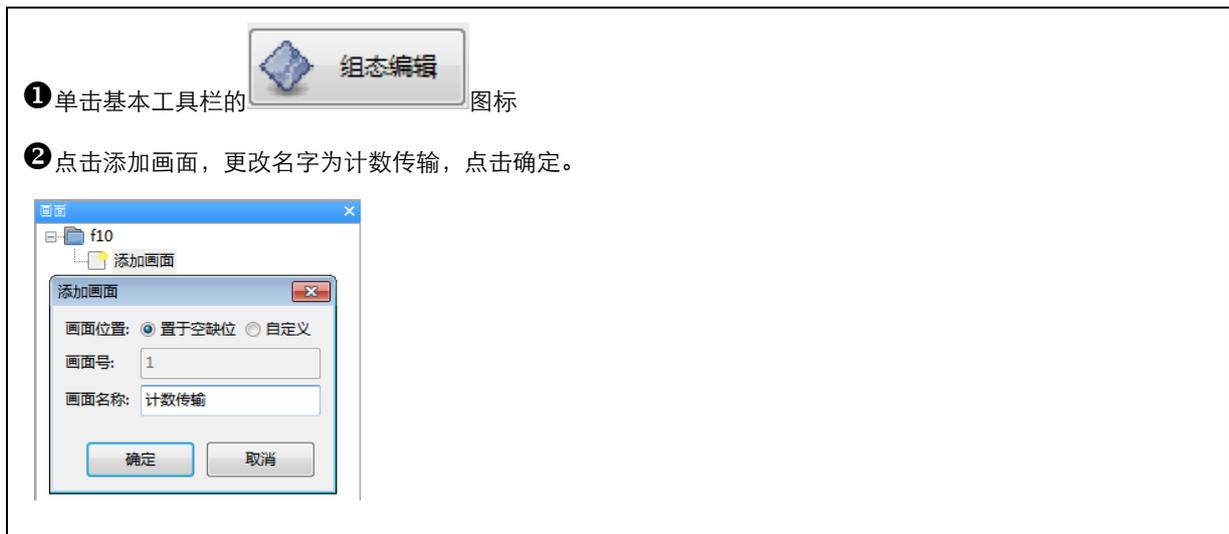


6.进入通讯参数设置界面，点击“网口”可以设置屏的 IP 地址（即下载到屏上后屏的 IP 地址）。点击“添加连接”设置通讯参数（[查看详情请参见【通讯文档】](#)），设置完成后点击完成

7.工程建立完成

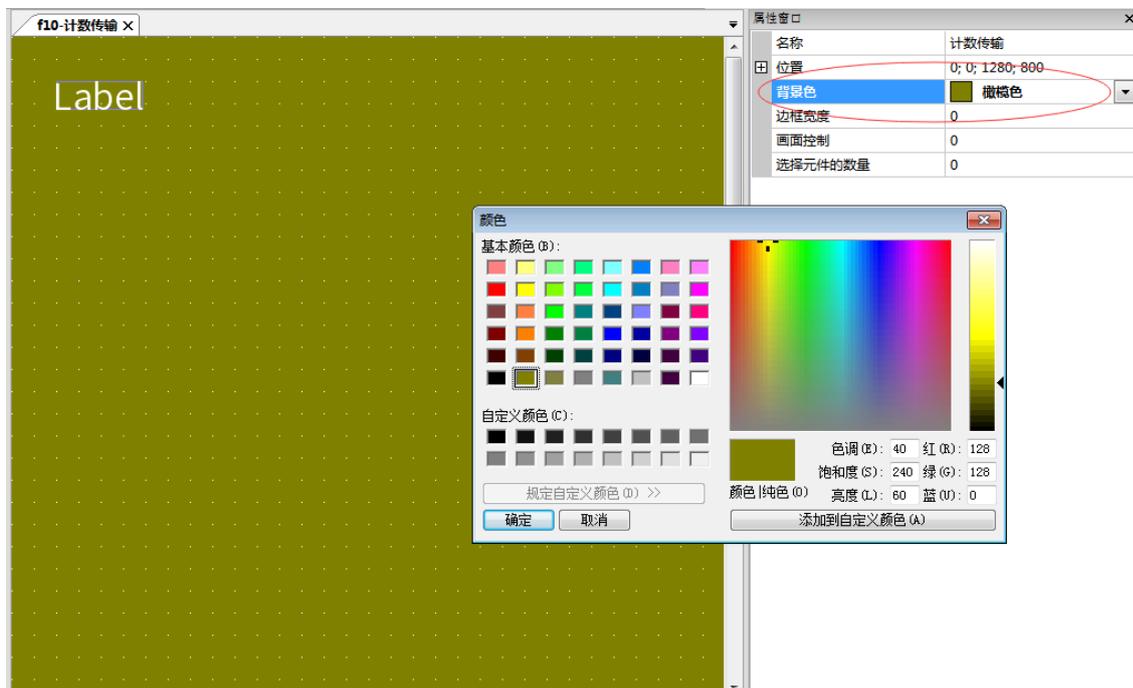
4.2.2 编辑组态画面

打开组态编辑窗口



编辑计数传输画面——修改窗口背景色

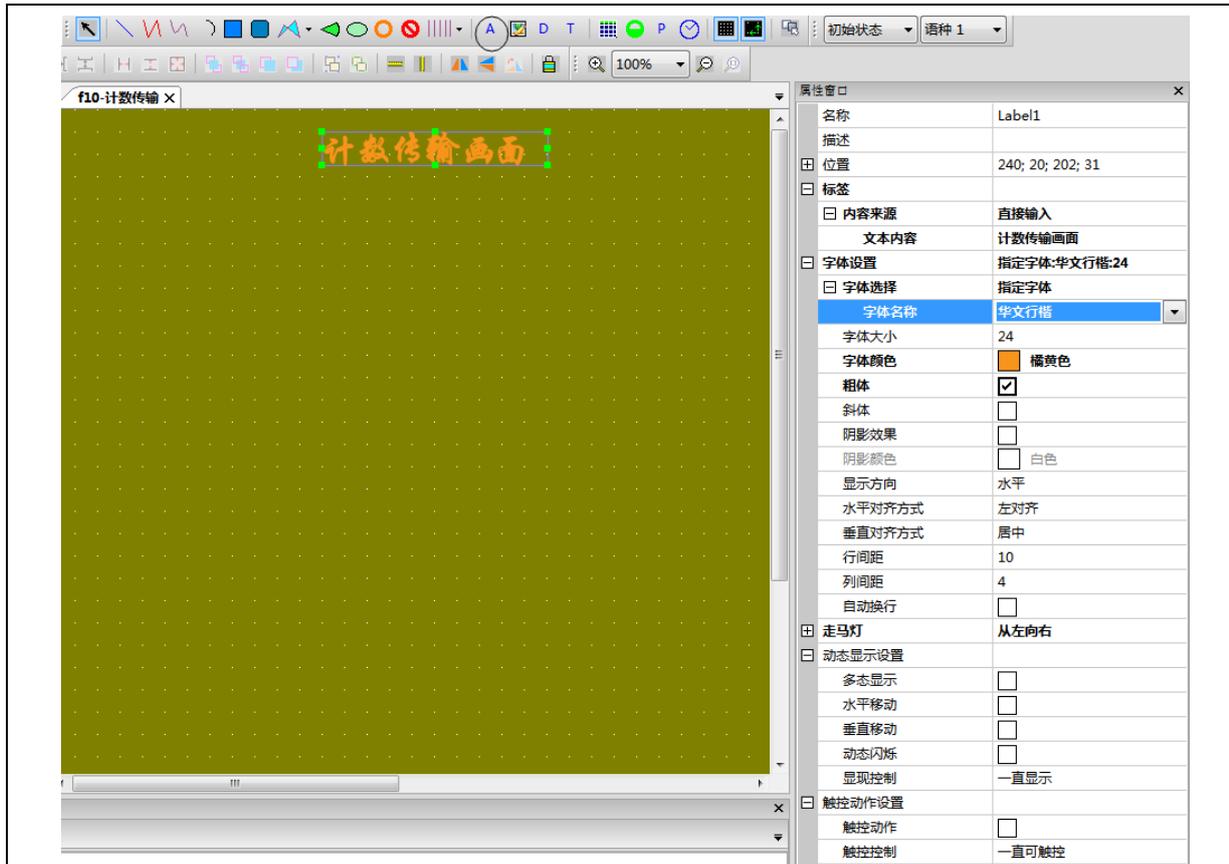
窗口初始背景色默认为黑色，用户可根据实际需要自行修改窗口背景色（本例设置窗口背景色为橄榄色）



操作步骤如下：

- ①** 在窗口空白区双击，右边弹出【属性】对话框
- ②** 勾选背景色
- ③** 单击“橄榄色”或选择“自定义颜色”更改颜色，系统弹出【颜色】属性框，在颜色盘中选择所需的颜色
- ④** 最后单击【确定】按钮

编辑计数传输画面——制作画面标题



- ① 单击绘图工具栏中的 **A** 图标，弹出【属性窗口】属性框
- ② 在【内容来源】的文本内容中输入“计数传输画面”字样，也可以通过文本库输入
- ③ 在“字体设置”中设置：

字体	指定字体的华文行楷
大小	24号
对齐方式	居中
水平对齐方式	左对齐
垂直对齐方式	居中
颜色	橘黄色

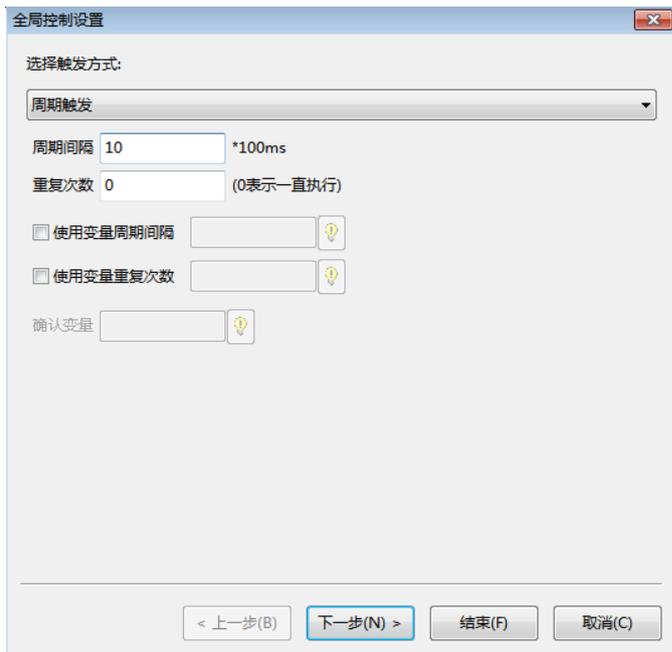
- ④ 在“走马灯”中设置：从左到右
- ⑤ 此时组态编辑窗口会出现设置成功的文本元件
- ⑥ 同理，创建“切换显示画面”

编辑计数传输画面——添加计数传输动作

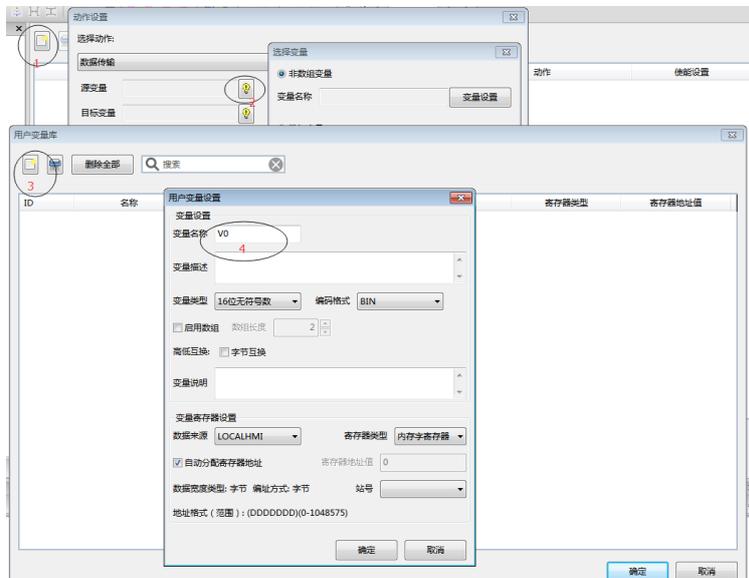
- ① 工程设置的全局控制设置，点击增加



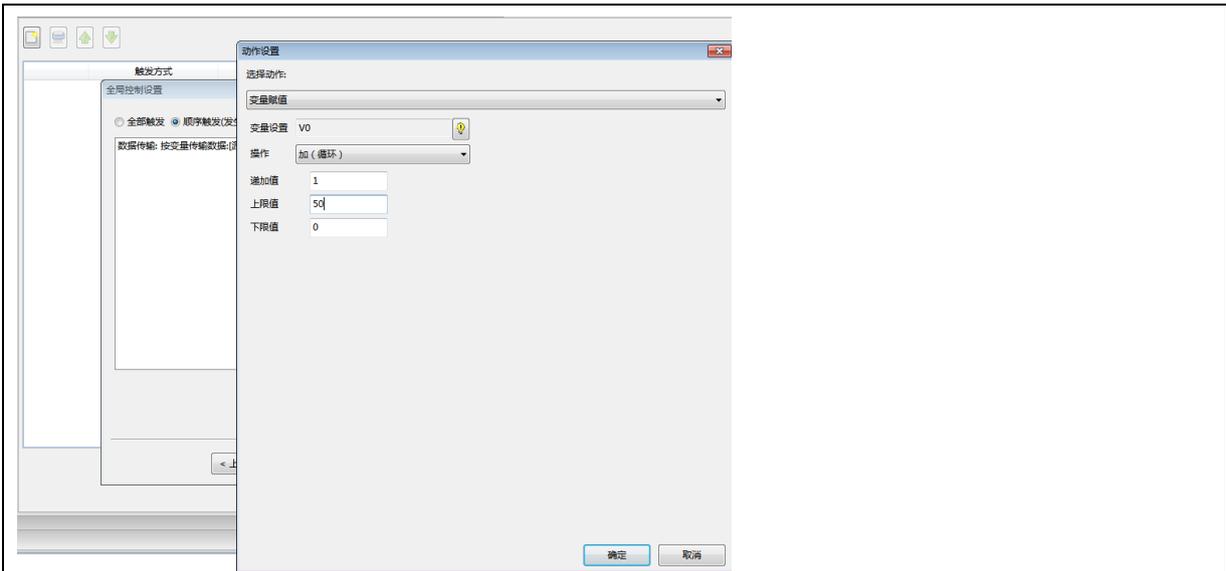
② 选择周期触发，周期为一秒，同时也可以选择用变量控制周期长度，点击下一步



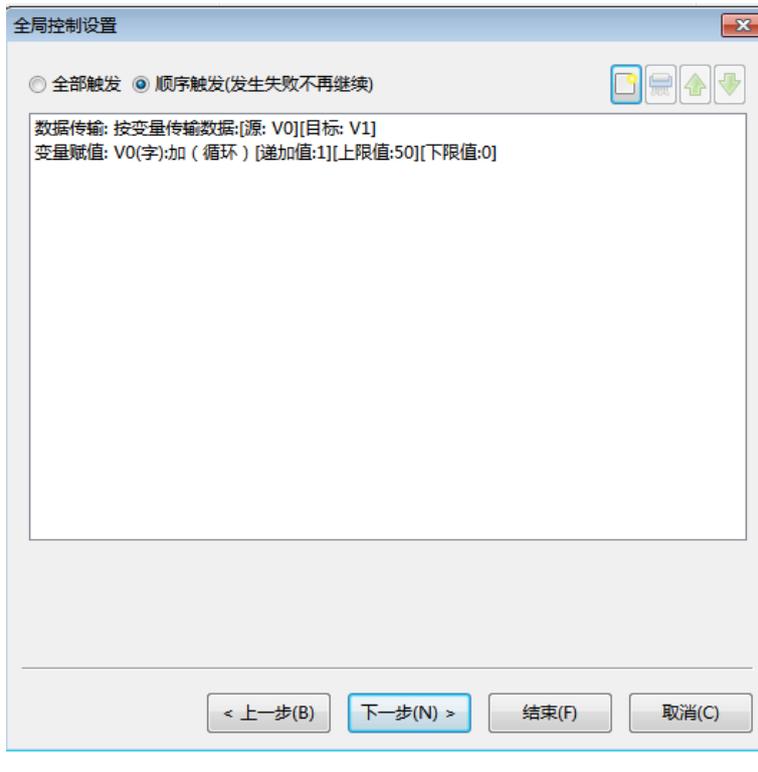
③ 单击【增加】，选择触发动作数据传输，添加源变量 V0 和目标变量 V1，添加完成后点击确定



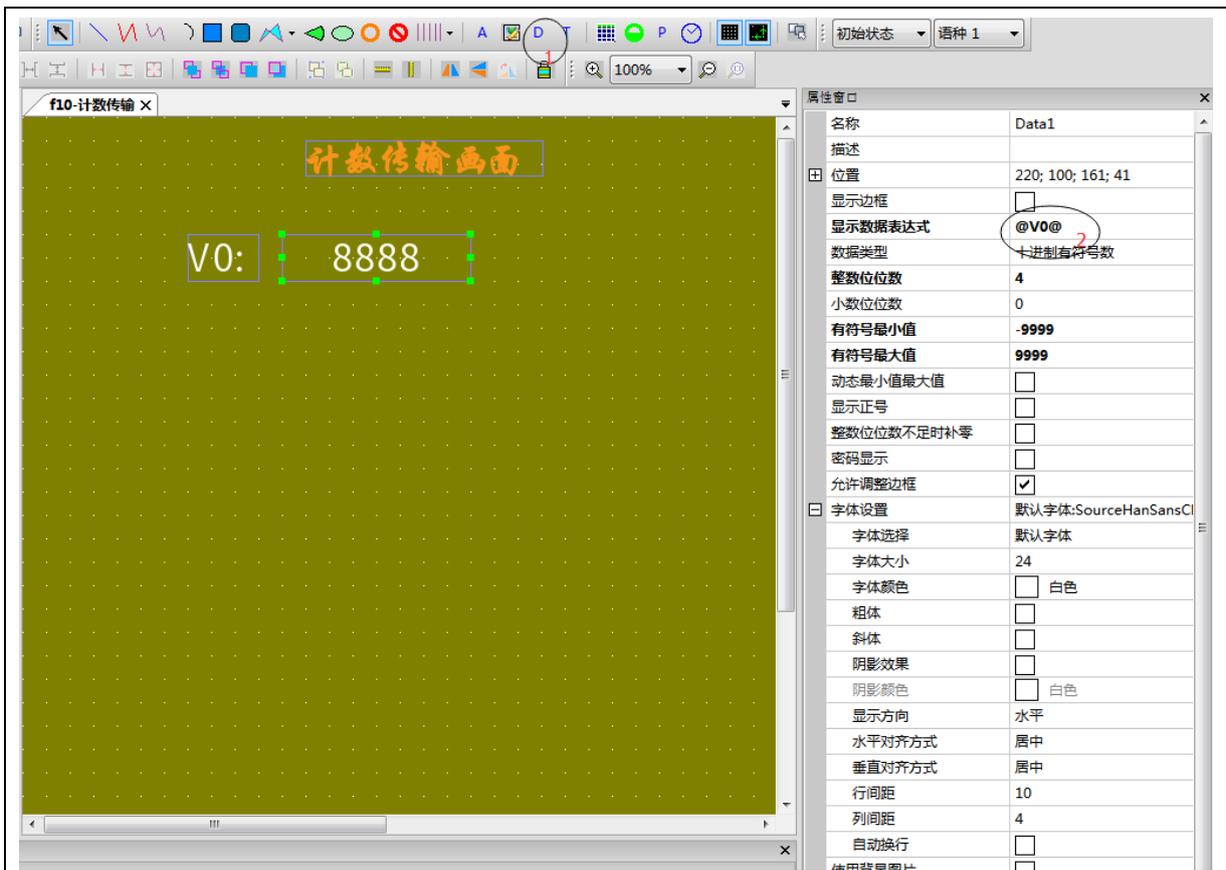
④ 添加变量赋值，和数据传输按顺序触发，一直使能



5 添加完成后点击确定



编辑计数传输画面——添加数值元件



① 单击绘图工具栏中的  图标，弹出【属性窗口】属性框

② 在【显示数据表达式】旋转 V0 变量

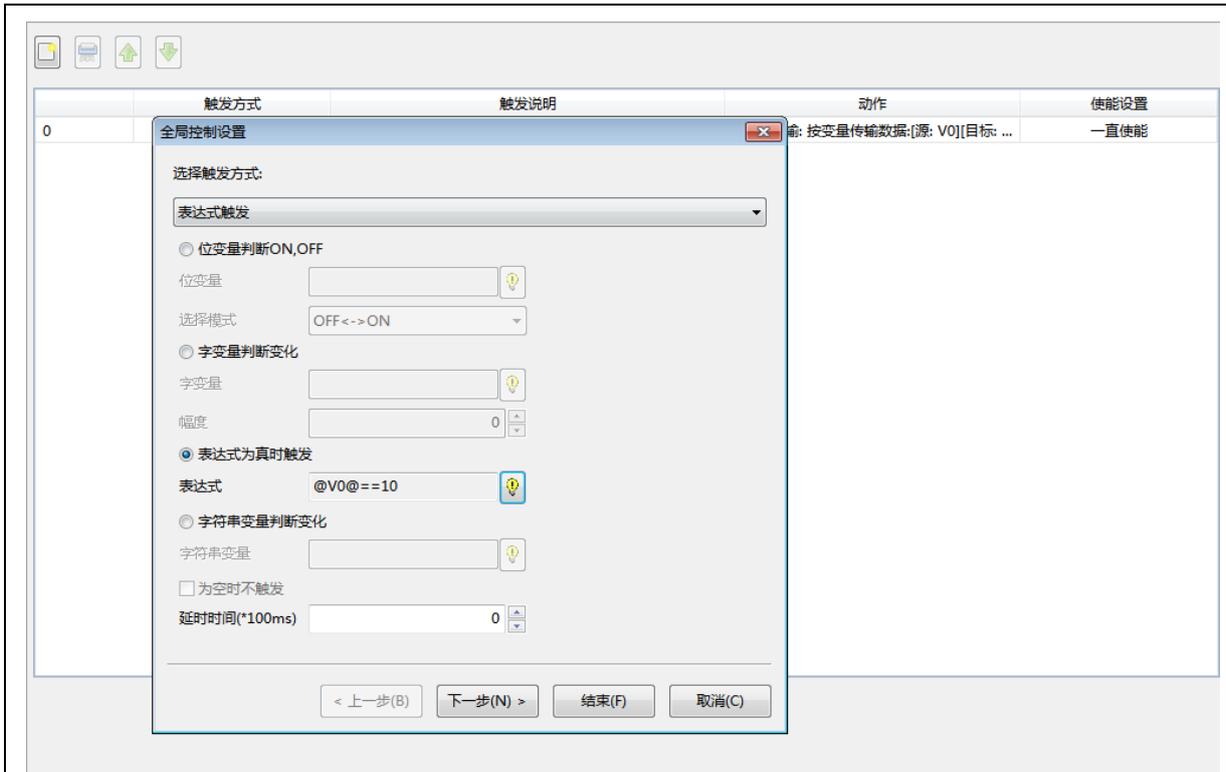
③ 在“属性窗口”中设置：

整数位数	4
小数位数	0
字体	默认字体
水平对齐方式	居中
垂直对齐方式	居中
颜色	橘黄色

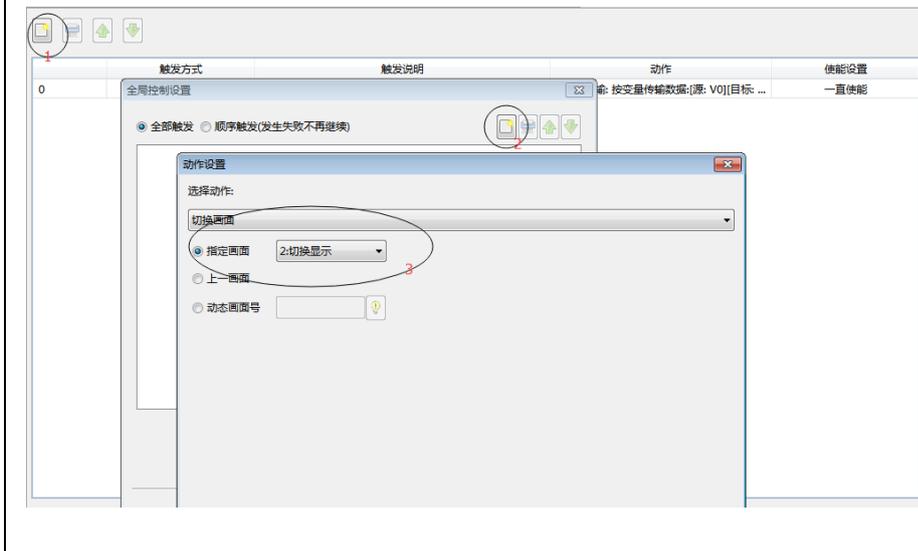
④ 同理，创建 V1 变量

编辑切换显示画面——添加切换画面动作

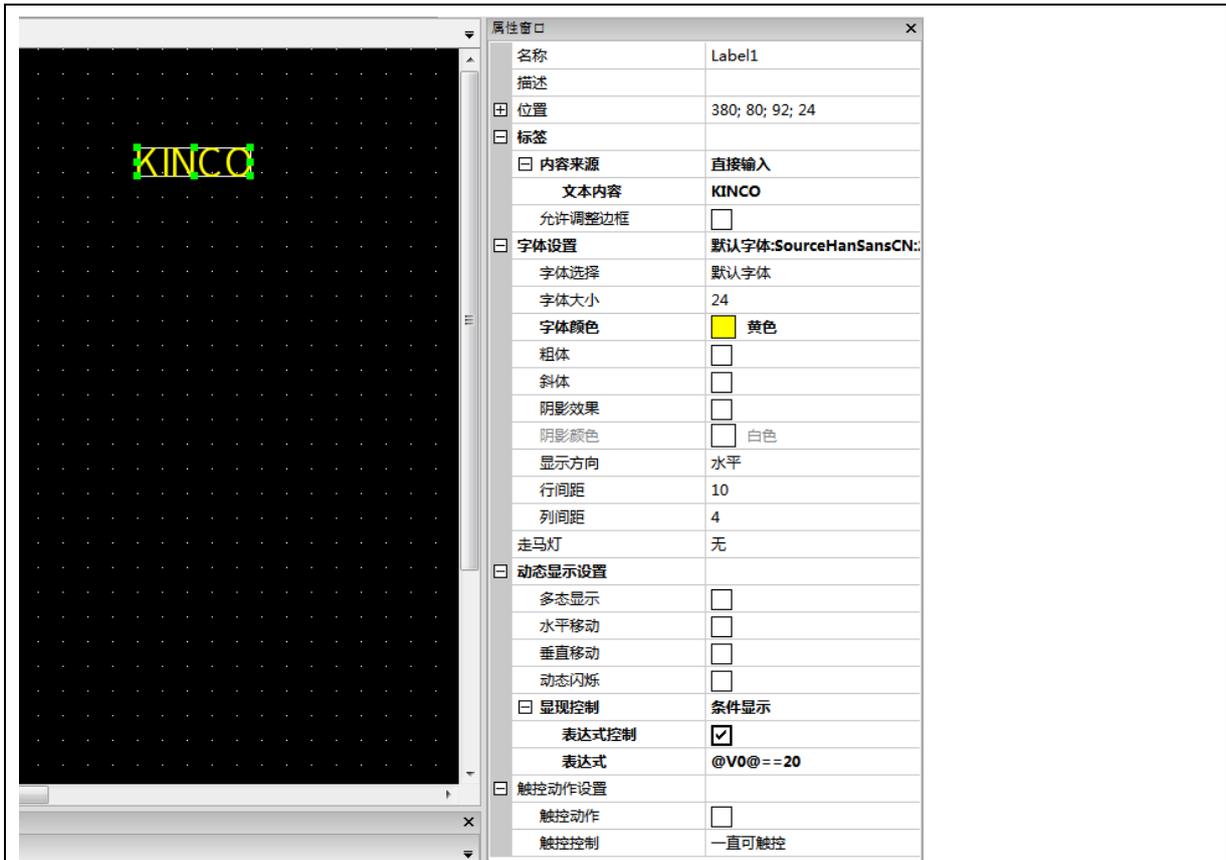
① 工程设置的全局控制设置，点击增加，选择表达式触发，条件为 V0=10 触发，点击下一步



② 选择切换画面中的指定画面，完成后点击确定



编辑切换显示画面——控制文本显示



① 添加标签元件，弹出【属性窗口】属性框

② 在【显示数据表达式】旋转 V0 变量

③ 在“属性窗口”中设置：

文本内容	KINCO
字体颜色	紫色
显现控制	表达式 V0==20

④ 同理，创建字体颜色为紫色的 KINCO 的标签，显现控制表达式 V0==30

4.2.3 工程保存

单击基本工具栏的  图标，对编辑好的工程画面进行保存。

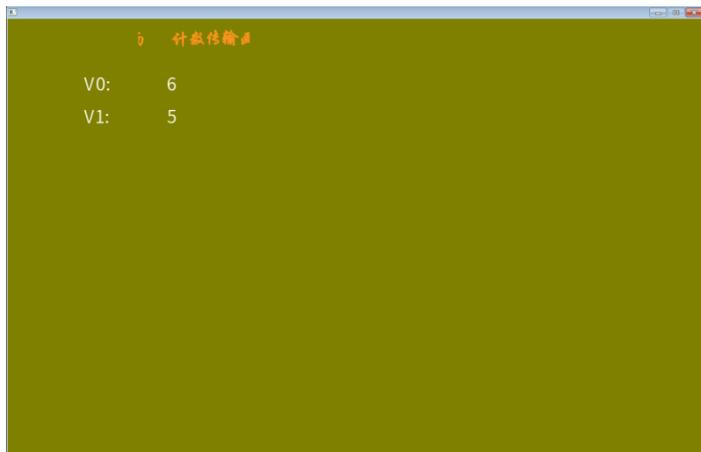
4.2.4 工程模拟

完成工程组态的编辑后，可通过离线模拟/实时模拟功能来仿真组态工程运行时的效果而不必每次下载工程到 HMI 中，可以极大的提高编程效率。离线模拟/实时模拟前要先将组态工程进行编译。

1.单击系统工具栏的  图标，对工程进行编译。



2.单击 按钮，开始模拟，如图所示：



和 PLC 有关的可以使用实时模拟

4.2.5 工程下载

离线模拟只能够反映工程画面显示效果，由于涉及到 PLC 设备数据的输入输出控制，组态工程需要下载到 HMI 里面运行，并与 PLC 设备建立通信才能看到实际的运行结果。工程下载操作步骤如下：



- 1.选择下载方式。单击系统工具的 图标，弹出 **KTDownload** 对话框
- 2.在【设置】中选择通讯方式“网口”，单击【确定】按钮
- 3.选择要下载的 HMI，单击【下载】按钮，开始下载

4.2.6 工程文件夹介绍

工程项目文件夹指用于组态工程数据存储的专用文件夹，它在工程建立时自动生成，并在制作工程的过程中生成对应操作的文件。下面简要介绍工程项目文件夹里的主要操作文件。

名称	说明
HMI_n	“n”为数字，工程中使用的 所有 HMI 都会有独立的文件夹，该文件夹主要用于存放宏指令文件、工程文件、 WebService 文件等
image	存放工程中使用的位图
tmpBackup	备份文件，只有保存工程崩溃后才会出现
tar	工程编译用数据文件以及用于下载到 HMI 运行的文件格式
name.sopkbp	编译 sop 后生成的 sop 工程文件，用于打开主工程中的 sop 工程文件
name.kbp	编译后生成的数据包文件，用于下载到 HMI 运行的文件格式
name.kpj	编译后生成的工程文件，用于打开软件的工程文件；

5. 编译/模拟/下载/上载

5.1 编译

编译分为：【编译】、【全部编译】、【清除编译结果】。

从【工具】菜单里选择【编译】/【全部编译】/【清除编译结果】，工具栏中的编译调试只是简单的编译。

名称	描述
编译	工程里面有宏文件时，点击【编译】不会重复编译已经编译过的宏文件
全部编译	清除编译好的文件包括.pkg 文件；宏文件.so/dll；.hmi 文件；.logo 文件，然后编译所有文件
清除编译结果	把编译好的文件全部清除，包括.pkg 文件；宏文件.so/dll；.hmi 文件；.logo 文件

5.2 模拟

KHDesigner 支持离线模拟和实时模拟,点击工具栏的或者从【工具】菜单里选择【离线模拟】/【实时模拟】弹出 HMI 选择对话框，选择需要的模拟 HMI,就可以离线模拟/实时模拟，关闭时，点击右上方的按钮关闭。

名称	PLC/控制器	HMI	描述
离线模拟	—	—	不需要连接 PLC 和 HMI，大幅缩短每次下载的时间。但不能从 PLC 获取数据，只能从本地地址读取数据，所有数据都是静态的
实时模拟	✓	—	只需要连接 PLC,不需要连接 HMI。可以获得动态的 PLC 数据。在无 HMI 的情况下，可以用来检查通讯是否正常



- 1.本地数据库字段有更改时，必须用数据库软件手动删除此表，否则模拟会报错。
- 2.Com2 口不支持实时模拟

5.3 下载

KHDesigner 提供了 3 种下载方式：USB、串口、网口，下载速度网口>USB>串口

5.3.1 下载方式选择

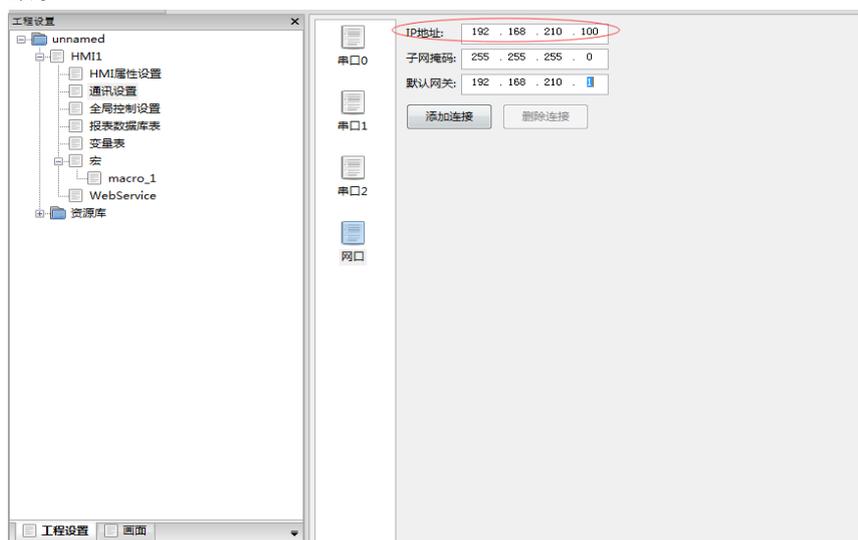


1.单击工具栏上的 ，或从【工具】菜单—【上传/下载】：

2.弹出【KHDownload】属性框，默认下载方式为 USB 口，单击【通讯方式】下拉菜单选择网口下载方式。

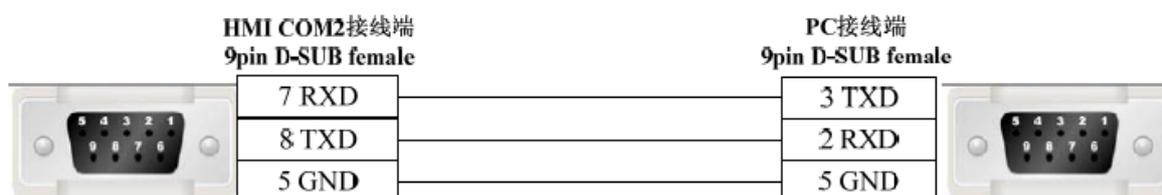
	<p>1.下载方式是根据 HMI 来保存的。例如：如果一个工程里有 2 个 HMI，HMI0 和 HMI1，HMI0 关闭前是串口下载方式，HMI1 关闭前是 USB 下载方式，再次打开该工程，HMI0 仍是串口下载方式，HMI1 仍是 USB 下载方式</p> <p>2.PC 的 IP 与 HMI 的 IP 要在同一个网段上。前面 3 个数字应相同，后面 1 个数字应不同，如果在局域网内，其 IP 不能与网络中其它 IP 相冲突。</p>
---	--

3.将【通讯设置】—【网口】中的 IP 地址设置成 HMI 需要的 IP，编译，下载到 HMI，此时 HMI 的 IP 就和工程中设置的 IP 地址一致了。



5.3.2 与 PC 的下载电缆接线

1 串口下载线



2 USB 下载线

标准 USB 下载线(B55/B1/X15/X15N 支持)

3.网口下载线

交叉网线和直连网线均可

5.3.3 下载内容选择

【KHDownload】属性	
设置	设置通讯方式，串口/网口/USB 口，X10 支持网口,X15 支持网口和 USB 口，在高级设置里

	面可以设置各个通讯方式的读超时、写超时、包长和包间隔
使用 IP 地址	选中后的 IP 地址即为下载到 HMI 上后 HMI 的 IP 地址，这可方便更改 HMI 的地址，重启后生效
HMI 列表	选择需要下载的 HMI
下载选项	用户数据文件，下载 HMI 时必须选择
	下载后清除本地数据库数据(本地数据库字段有更改时，必须选择此项，否则屏上会报错)
	下载后磁盘寄存器

6. 库的了解

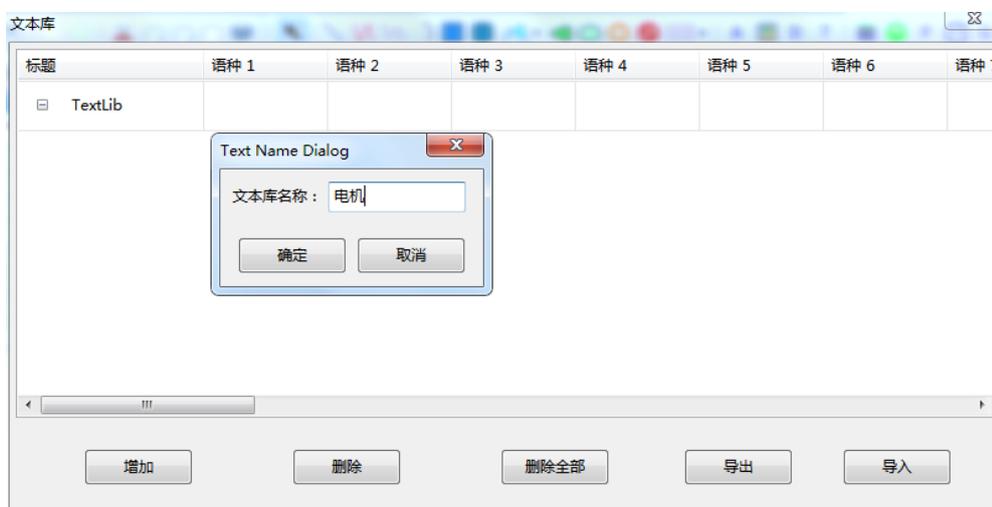
在 KHDesigner 中，用户可以将常用或频繁使用的文本库和图库存放在类似“数据库”的空间里，当需要时可直接调用“数据库”里的这些资料，这种“数据库”空间称之为库。本章将详细介绍库的使用方法。

6.1 文本库应用

6.1.1 文本库的建立

(1) 点击【工具】菜单—【文本库】或从【工程设置】—【资源库】—【文本库】中点击【文本库】即可打开文本库对话框。

(2) 点击【增加】按钮，弹出对话框：



(3) 建立一个名称为“电机”的文本条目

(4) 编辑完成后点击【确定】关闭文本库对话框。这时可在文本元件属性【标签】选项的内容来源选择文本库后选择建好的文本名称：



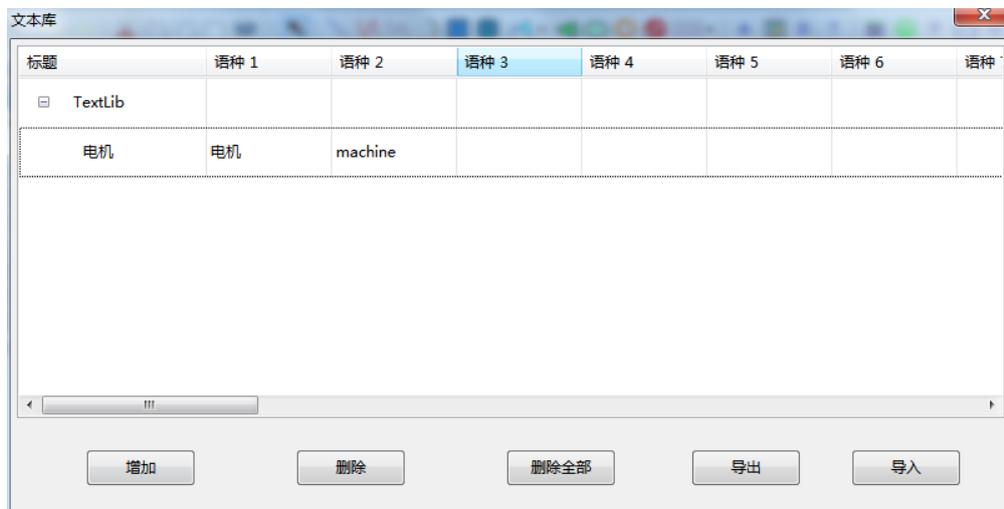
6.1.2 文本库的语种设置

从【HMI 属性设置】中点击【语种字体设置】属性框来设置每个语种所对应文本字体的全局属性。

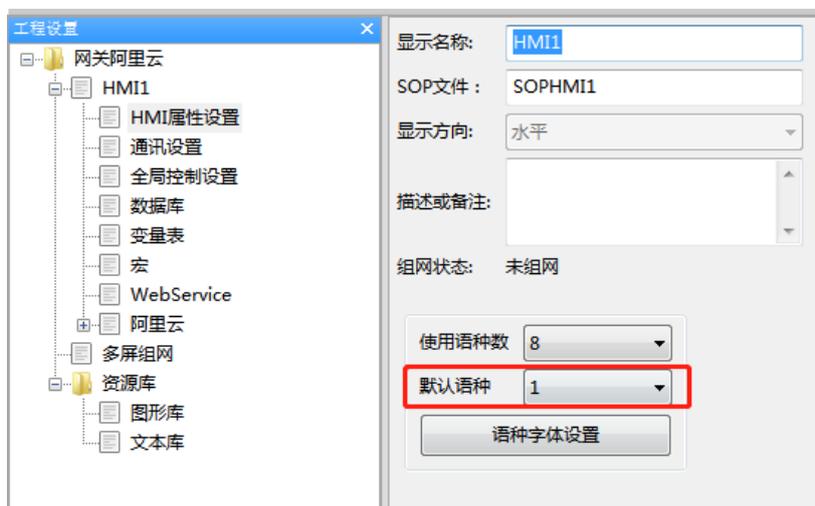
	<ol style="list-style-type: none">1. 在【语种字体设置】中设置每个语种的字体属性是对全局来说的，而在元件属性【标签】属性页中设置文本库的字体属性只对当前元件的标签内容有效2. 更改【语种字体设置】中的字体属性，不会影响元件中已经使用了文本库作为标签的字体属性，如需使用【语种字体设置】中的字体属性，需将元件【标签】属性页中“使用文本库”前的勾选取消掉，然后再勾选，以便刷新为【语种设置】中的字体属性
---	--

6.1.3 文本库实现多语言切换

【范例 1】实现电机中英文之间的切换。建立一个电机的文本框，点击电机的语种 1 输入电机，语种 2 输入 **machine** 如下图：



在工具栏中的语种选项，可以在组态编辑界面看到显示效果。如若更改 HMI 上的显示效果，需在工程设置的 HMI 属性设置更改默认语种，通过更改默认语种实现多种语言之间的切换



在线更改多语言切换功能，使用系统寄存器 SYS_LANGUAGE，值为 0 或者超出了工程最大语种数，则使用默认语

种;

6.1.4 文本库导入导出

1. 导出的是 CSV 格式的文件，可以用 excel 来打开。
2. 导入时，如果遇到有相同名称的数据，会弹出设置框，可以选择替换或者跳过；如果勾选了“统一使用该操作”，则下次再遇到有相同名称数据的情况时直接使用该操作。
3. csv 文件中，第一列是标题，以后的列是文本库的数据。

6.2 图形库应用

KHDesigner 提供了强大的图形库供用户使用,可以从软件外部导入位图图片。本节将介绍与图库有关的使用方法。

6.2.1 图形的导入

可通过以下三种方式打开【导入图库】属性框：

方式 1: 点击工具栏上的  图标

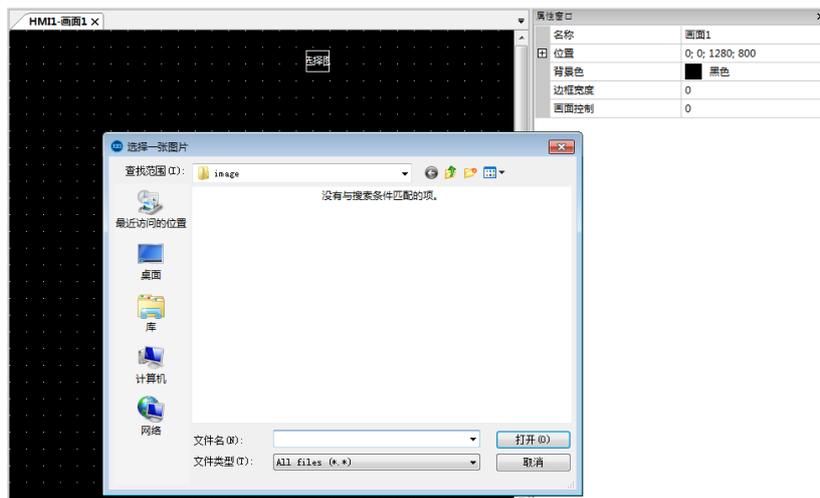
方式 2: 从【绘图】菜单中点击“位图”

方式 3: 从工程设置的【图形库】属性页中点击“增加”

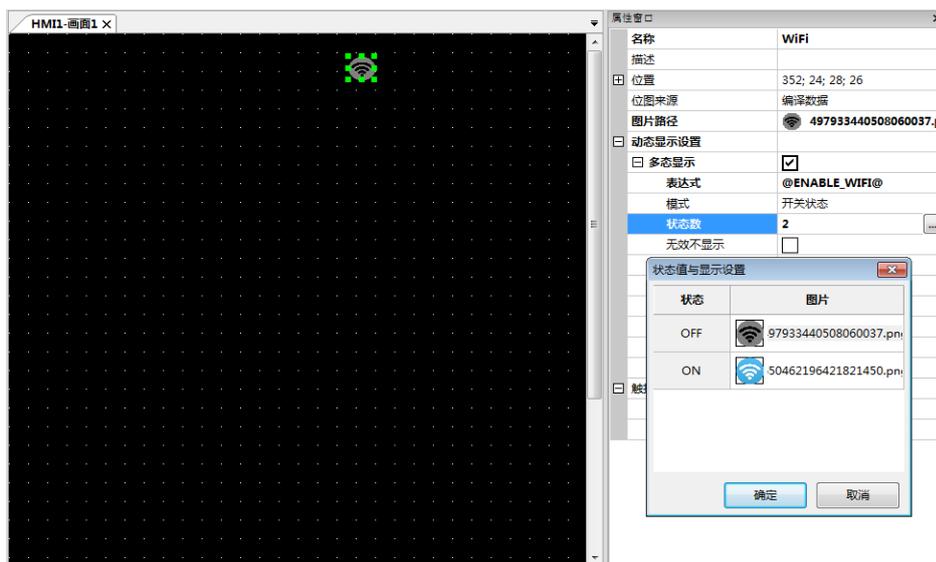
	从图库导入图形时不支持同时导入多个位图图片
---	-----------------------

6.2.2 图形的建立与使用

【范例 1】绘制一个名称为“wifi”，具有两种状态的按钮，点击工具栏上的  图标或从【绘图】菜单中选择“位图”，弹出图片选择对话框：



点击取消，选择图片，出现图片属性对话框，更改名称为 **WiFi**，图片路径选择默认的图片，表达式选择系统寄存器 **ENABLE_WIFI**，模式开关状态，状态 **OFF** 和 **ON** 选择相应的图标，如下图



点击确定，**WiFi** 图标制作完成，点击工具栏上的  图标或点击【文件】菜单里的【保存工程】对制作好的位图进行保存，导入的格式为 **BMP、JPG、PNG** 等原始图片同时会保存到工程文件里名称为 **【image】** 的文件夹中。



请不要删除工程文件里名称为 **【image】** 的文件夹中的文件，否则将会导致工程中使用到的向量图/位图不能正常显示

KHDesigner 组态编辑软件还支持位图数据直接从外部存储器中调用图片，以节省 **HMI** 本身的内存空间。



图片从外部存储器读取功能仅适用于支持带 **USB HOST** 或 **SD 卡槽** 的 **HMI**

【范例 2】制作一个背景图片，将文件名为 **background.bmp** 的图片存放在 **U 盘** 或 **SD 卡** 的根目录下(本例存放在 **USB1** 里)，组态工程直接去读取 **U 盘** 里的 **background.bmp** 图片。

1.将 **background.bmp** 拷贝到 **U 盘** 的根目录下。

2.新建位图：点击工具栏上的  图标或从【绘图】菜单中选择“位图”，弹出图片选择对话框：

3.设置图片从外部文件读取，选择设备类型、文件类型、刷新变量以及文件名等

属性窗口	
名称	WiFi
描述	
位置	352; 24; 28; 26
位图来源	外部文件
设备类型	HMI
文件类型	bmp
刷新变量	
常量还是变量	常量 <input type="button" value="显示"/>
文件名 (包括路径)	
动态显示设置	
多态显示	<input type="checkbox"/>
水平缩放	<input type="checkbox"/>
垂直缩放	<input type="checkbox"/>
水平移动	<input type="checkbox"/>
垂直移动	<input type="checkbox"/>
动态闪烁	<input type="checkbox"/>
显现控制	一直显示
触控动作设置	
触控动作	<input type="checkbox"/>
触控控制	一直可触控

4.背景图片制作完成，单击工具栏上的  图标对制作好的位图进行保存。

	<ol style="list-style-type: none"> 1. 【文件名设置】一定要和保存在外部存储器所要读取的位图名称一致 2. 从外部存储器读取的图片只支持 BMP、JPG、PNG 格式 3. 图片必须存放在外部存储器的根目录下 4.使用扩展存储图片的位图不支持【使用图片原尺寸】功能，用户需自己按原图尺寸来调整大小
---	---

6.2.3 图片的优化

KHDesigner 组态编辑软件位图支持导入 BMP、JPG、JPEG、PNG 等格式的图片，很大程度上可以满足用户的工程需求，但是用户导入图片的尺寸大小和色深不仅会影响工程的大小，而且也会影响 HMI 界面的运行的速度。

所以在使用导入的图片时还需注意以下几点：

- 导入的图片尺寸最大限制为当前 HMI 的屏幕分辨率。如当前 HMI 的屏幕分辨率为 640×480，则导入的图片尺寸不应超过 640×480。尽可能把导入的图片大小通过第三方图片编辑工具处理成与元件大小一致的尺寸，例如在组态工程时需要用一个宽度和高度为 100×100 的元件，并使用位图，那么新建位图时，应先用第三方图片编辑工具将要导入的图片分辨率处理成 100×100，然后再导入。如果对图片的显示效果要求不高的话，导入的图片尺寸处理得尽可能小些，然后再拉伸到需要的尺寸大小。
- 由于图片在下位机是以无损压缩的 BMP 格式保存的，如果导入的图片是有损压缩的 JPG 等格式的话，编译后图片的大小反而会增大，这样图片的像素不仅被损失掉，而且也没能达到减小工程大小的效果。所以用户在选择使用的位图时需合理优化图片的尺寸大小并按实际情况来选择压缩格式。

6.3 数据库应用

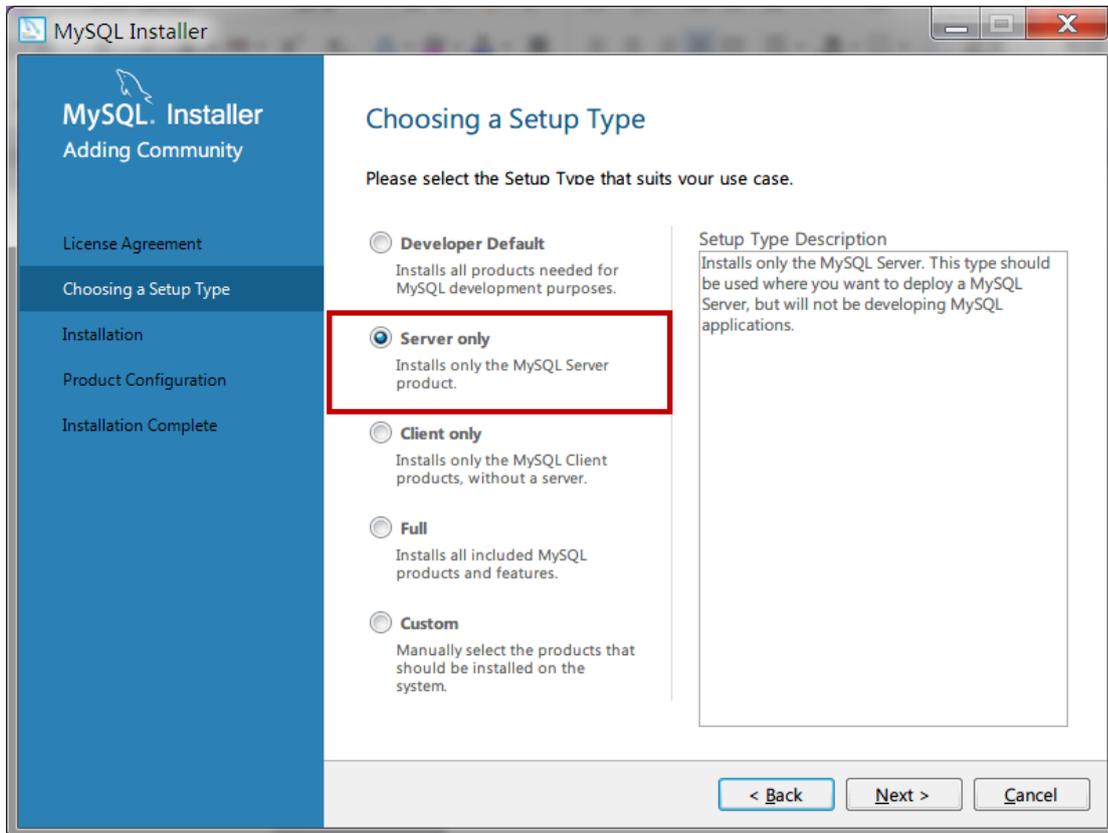
数据库分为本地数据库和远程数据库，本地数据库支持 MySQL 数据库，远程数据库支持 MySQL 和 SQL Server 数据库，本地数据库在数据库表元件使用时，若本地数据库中没有，点击任何关于此数据库的操作，会自动创建，远程数据库不会自动创建，必须先创建再使用。这里以远程数据库 MySQL 为例介绍数据库的建立和修改。模拟显示时，必须安装好所使用的数据库。

6.3.1 数据库的安装

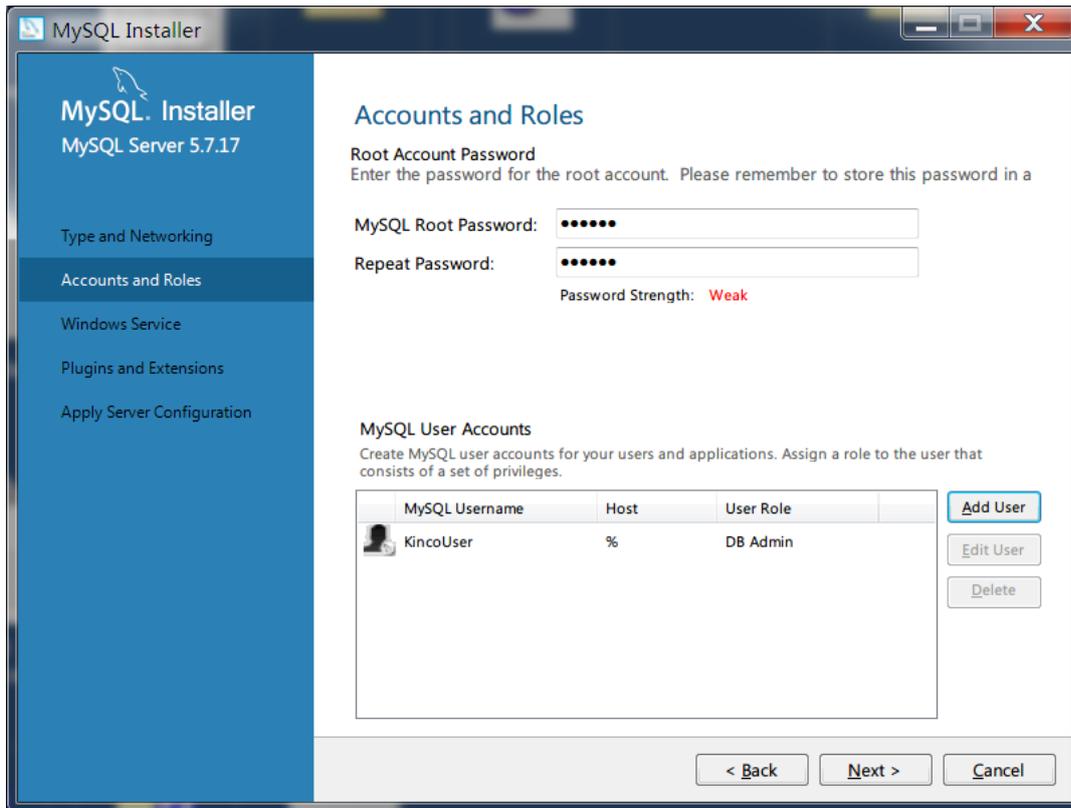
工程进行模拟显示时，必须安装好所使用的数据库。并且建立一个数据库名为 KINCOPCDB，用户名为 KDBUser，密码为 KDBPwd 的数据库你，这里以 mysql5.7.17(XP 系统需安装低版本的 MYAQL)安装进行介绍：

1.双击 mysql-installer-community-5.7.17.0.msi，若提示需要安装 .Net Framework 4.0，则先退出安装程序。点击 Microsoft.NET.exe，使用默认设置即可。安装完 .Net Framework 4.0 后再次双击 mysql-installer-community-5.7.17.0.msi。

2.大部分设置都用默认就好，有几个地方需要修改一下：选择“server only”，默认是第一项。

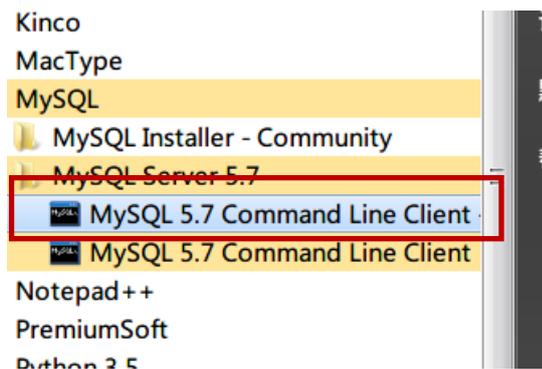


3.设置 root 密码，可以随便设置，然后点击“Add User”按钮，添加一个用户，**用户名必须为 KDBUser，密码为 KDBPwd (必须添加)**



4.然后一直点击下一步，直到安装完成。

5.安装好后需要添加一个数据库。在开始菜单列表中找到

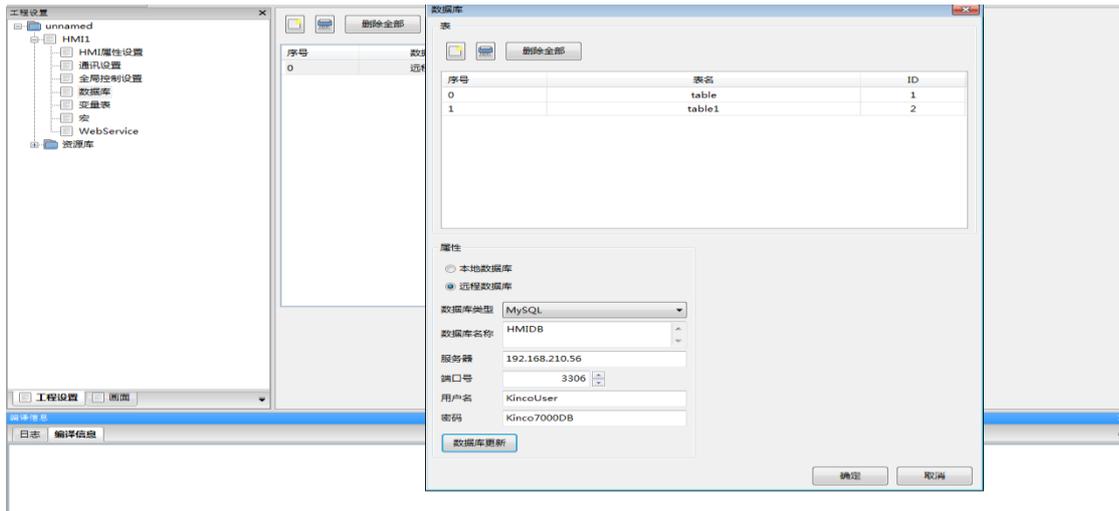


6.点击进入，提示需要 root 密码，这个密码就是上面设置的 root 密码，输入后回车，然后输入 “`create database KINCOPCDB;`” 双引号内的内容（不要双引号，保留后面的分号），然后回车。

7.关闭命令行工具，完成安装工作。

6.3.2 数据库的建立

1.选择工程设置中的数据库，点击增加，输入数据库的名称、服务器、端口号用户名和密码，点击更新，数据库中的表会自动显示出来，属性不可更改

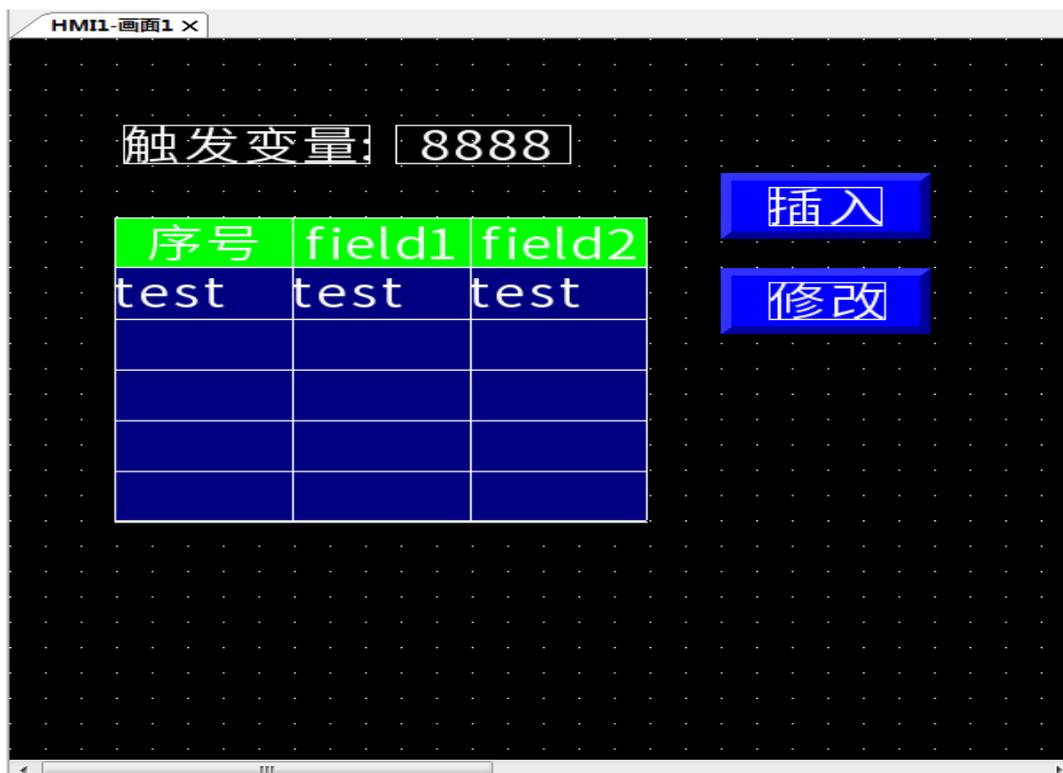


2.在组态编辑软件中选择数据库元件，设置表格属性和报表设置后，建立触发变量 variable0,如下图



6.3.3 数据库的使用

1.工程设计两个数据库动作，插入动作 field1=aa,field2=789。然后进行修改动作，使用当字段 field1 为 aa 时，修改 filed1 为 edhz; ,field2 为 125，如下图



2.工程完成后进行编译，模拟显示。

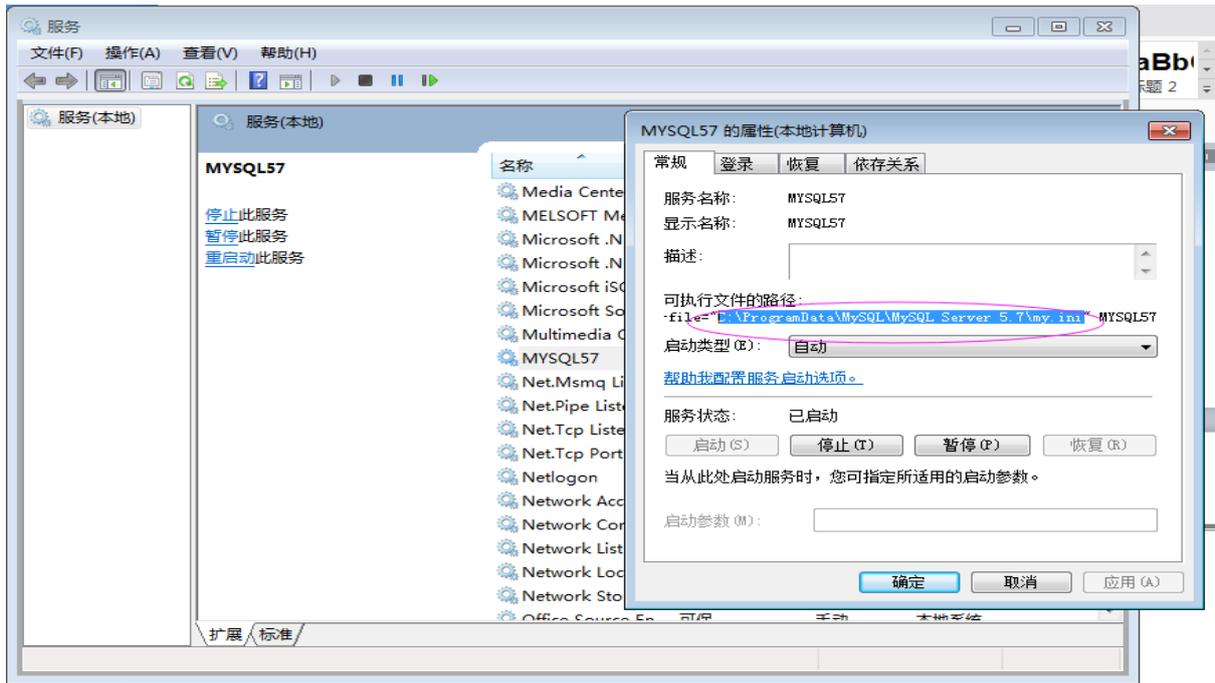
3.点击下载程序。

	<p>当本地字段添加或删除等有改动时下载时必须选择清除本地数据库数据，模拟时必须使用数据库软件删除数据库！</p>
---	---

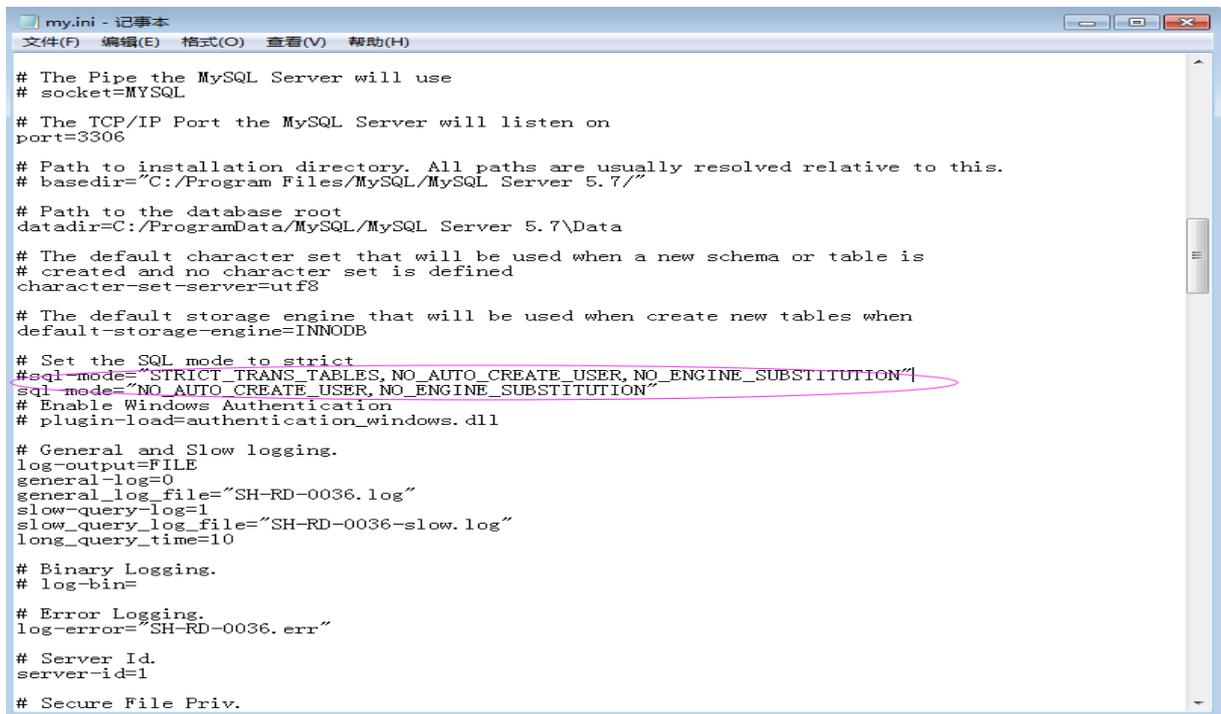
6.3.4 数据库的特殊情况

当数据库中插入或者导入位、16 位、32 位、单精度和双精度为空的时候，模拟显示报 code1364 1366 1256 错误的处理方法：

1.在系统服务中找到 mysql 服务，右键属性，，看一下 mysql 使用的配置文件是哪个，如下图



2.然后打开该配置文件，找到"sql-mode 项"，把 "STRICT_TRANS_TABLES" 去掉，建议做法先备份原来的语句，再更改，如下图

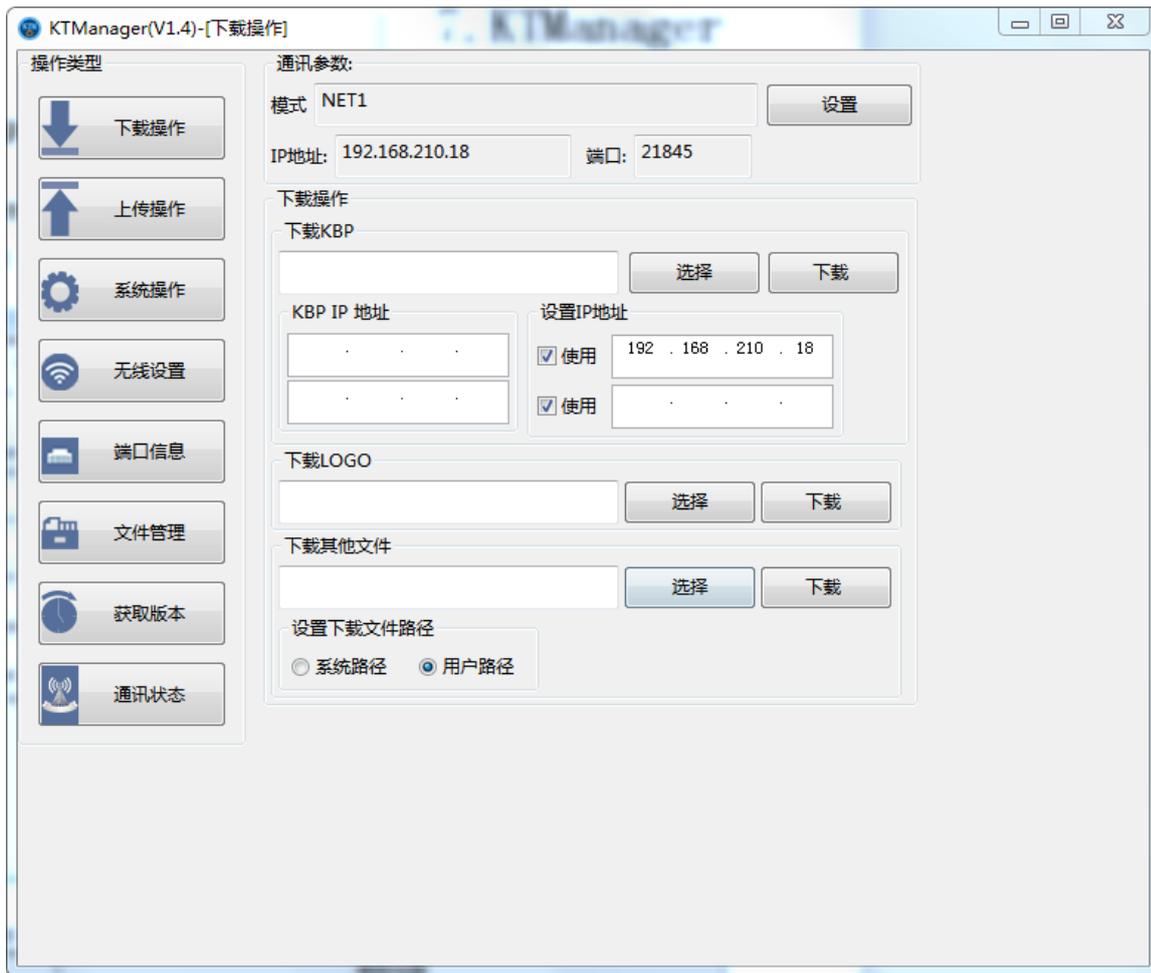


3.然后重启 mysql 服务

7. KTManager

7.1 KTManager 介绍

KTManager 包含 4 个处理模块：【下载操作】、【上传操作】、【系统操作】、【无线设置】、【端口信息】、【文件管理】、【获取版本】



【通讯参数】：显示当前 KTManager 使用的下载方式。点击【设置】可修改当前下载方式

下载方式		参数设置
网口	<input checked="" type="radio"/> 网络	设置当前 HMI 的 IP 地址和端口号

		
串口	<input checked="" type="radio"/> 串口	选择 PC 机当前使用的串口号 
USB	<input checked="" type="radio"/> USB	无需设置通讯参数

【上页】 / 【下页】： 点击【上/下页】在 4 个处理模块之间跳转

【关闭】： 退出 KTManager 界面

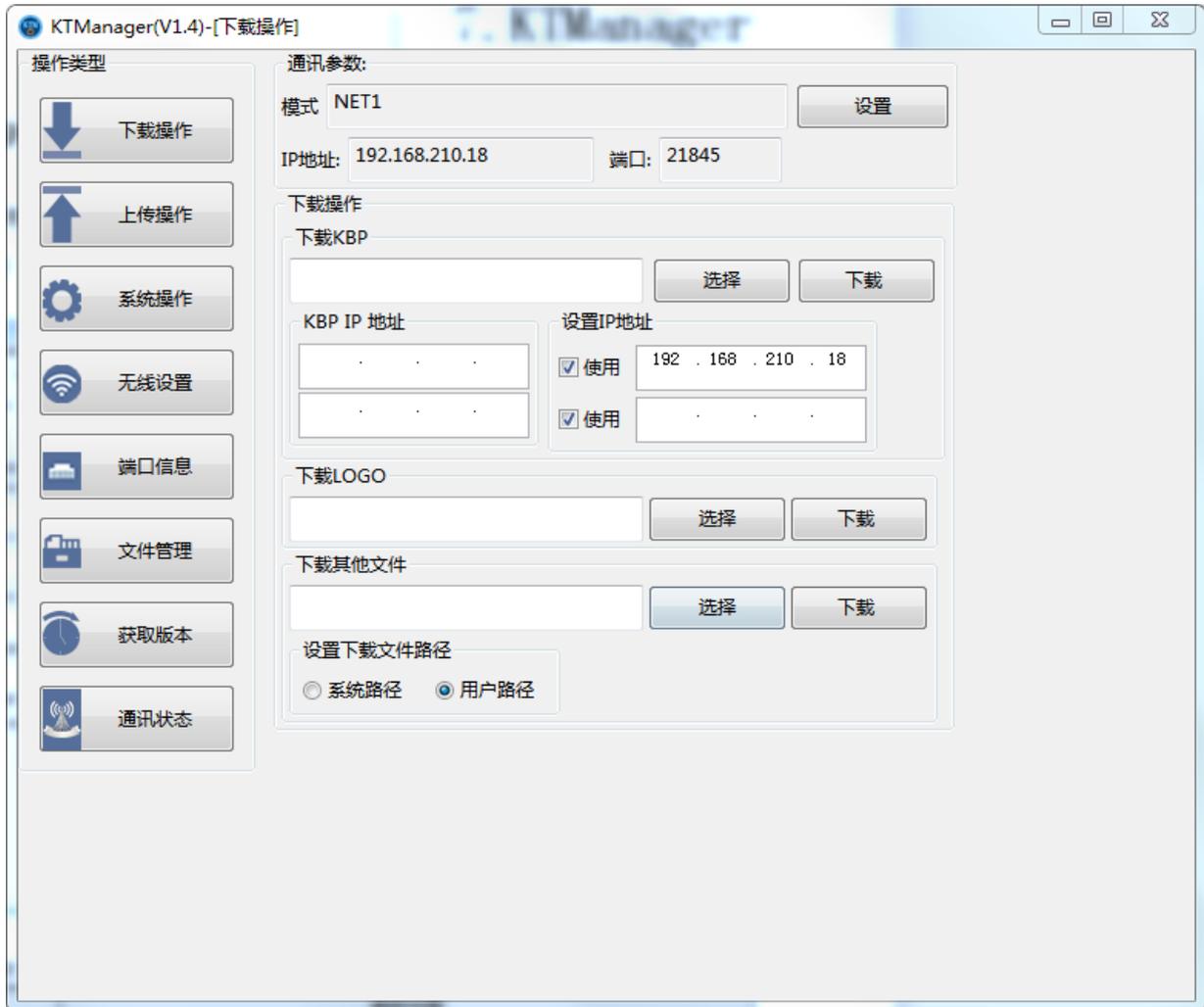
7.2 打开 KTManager 的方法

- 从 PC 的【开始】菜单打开
- 双击桌面 KTManager 快捷方式打开
- 从 KTManager 软件【工具】菜单打开

单击【工具】菜单—单击【上传操作】 / 【下载操作】 / 【系统操作】 / 【无线设置】 / 【端口信息】 / 【文件管理】 / 【获取版本】

- 在 KHDesigner 安装目录下双击【KTManager.exe】打开

7.3 下载操作



【下载操作】具体说明		
下载操作	下载 KBP	下载.kbp 文件到 HMI,当选择好 KBP 文件时，工程原 IP 显示在 KBP IP 地址下面，通过设置 IP 更改工程的 IP
	下载 LOGO	选择 x.logo,x.png,x.bmp,,x.jpg 文件,下载开机画面(,x.png,x.bmp,,x.jpg 需要选择对应的 HMI，x.logo 必须与实际的 HMI 分辨率一致)
	下载其他文件	把文件下到屏里面对应的路径，系统路径为 home/bin,用户路径为 home/user

7.4 上传操作



【上传操作】具体说明

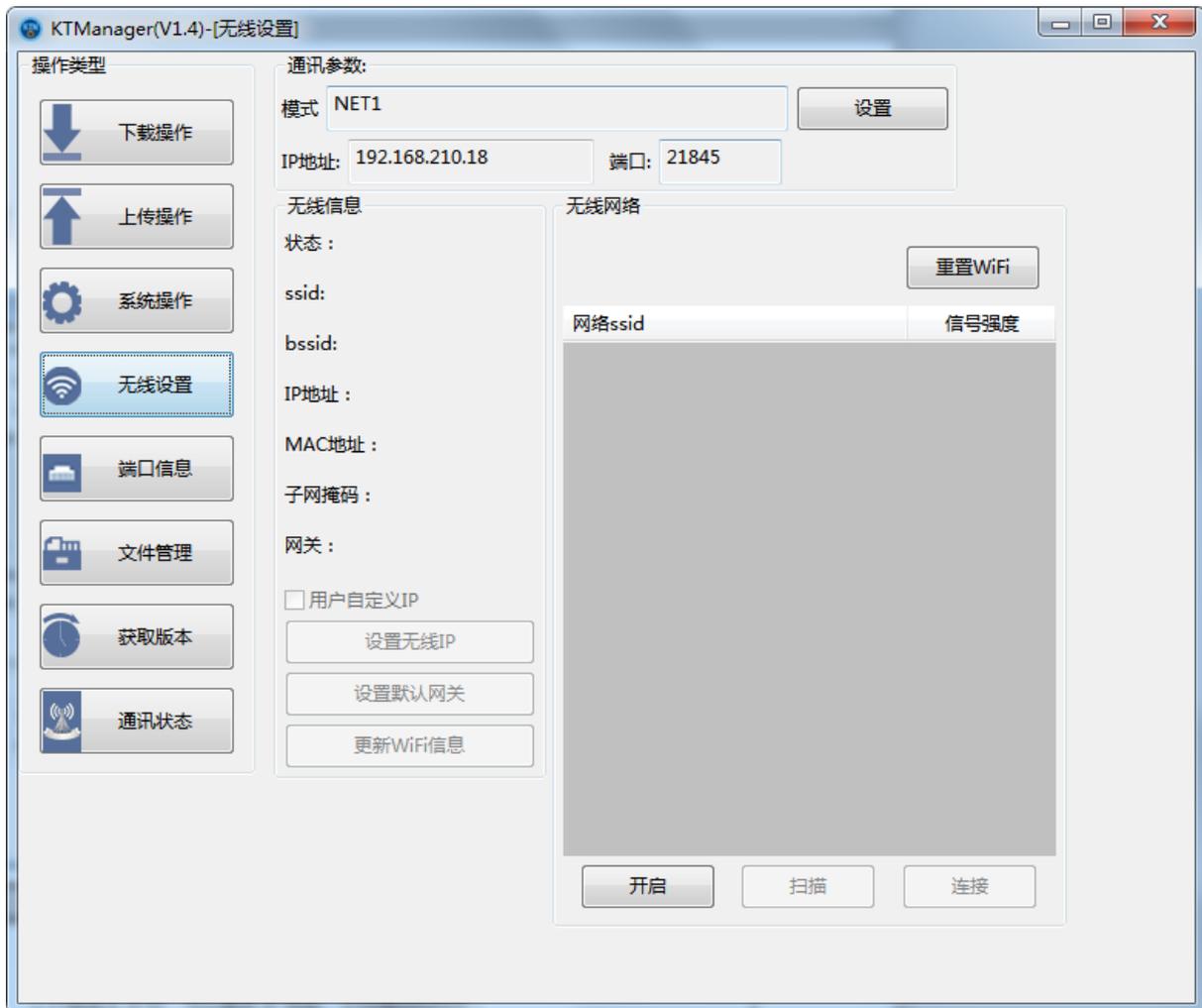
上传 KBP	上传工程.kbp 文件
上传 Logo	上传 HMI 里面的.Logo 文件
上传日志文件	上传 HMI 中/home/log 下面的 log 文件
上传反编译	上传反编译文件*.dec
上传其他文件	上传 HMI 里面的文件,通过路径文件名查找上传的文件

7.5 系统操作



【系统操作】具体说明		
获取屏 IP / PORT 信息	获取 HMI 当前 IP 地址和端口号信息	
设置屏 IP/PORT	将 HMI 当前 IP 地址和端口号更新为设置的 IP、掩码和网关。	
DNS 服务器	读取与设置屏的 DNS 服务器	
显示 Logo	显示	开机启动时显示 logo 图片
	隐藏	开机启动时不显示 logo 图片
重启屏	重新启动 HMI	
更新系统	若软件和屏上系统版本一致不更新，不一致才更新。如果 bootloader 不支持更新，提示不支持更新。	
同步时间	HMI 与电脑的时间同步	
用户文件	删除用户路径 home/user 下的文件	
反编译	选择反编译文件、设置反编译的文件的途径	

7.6 无线设置



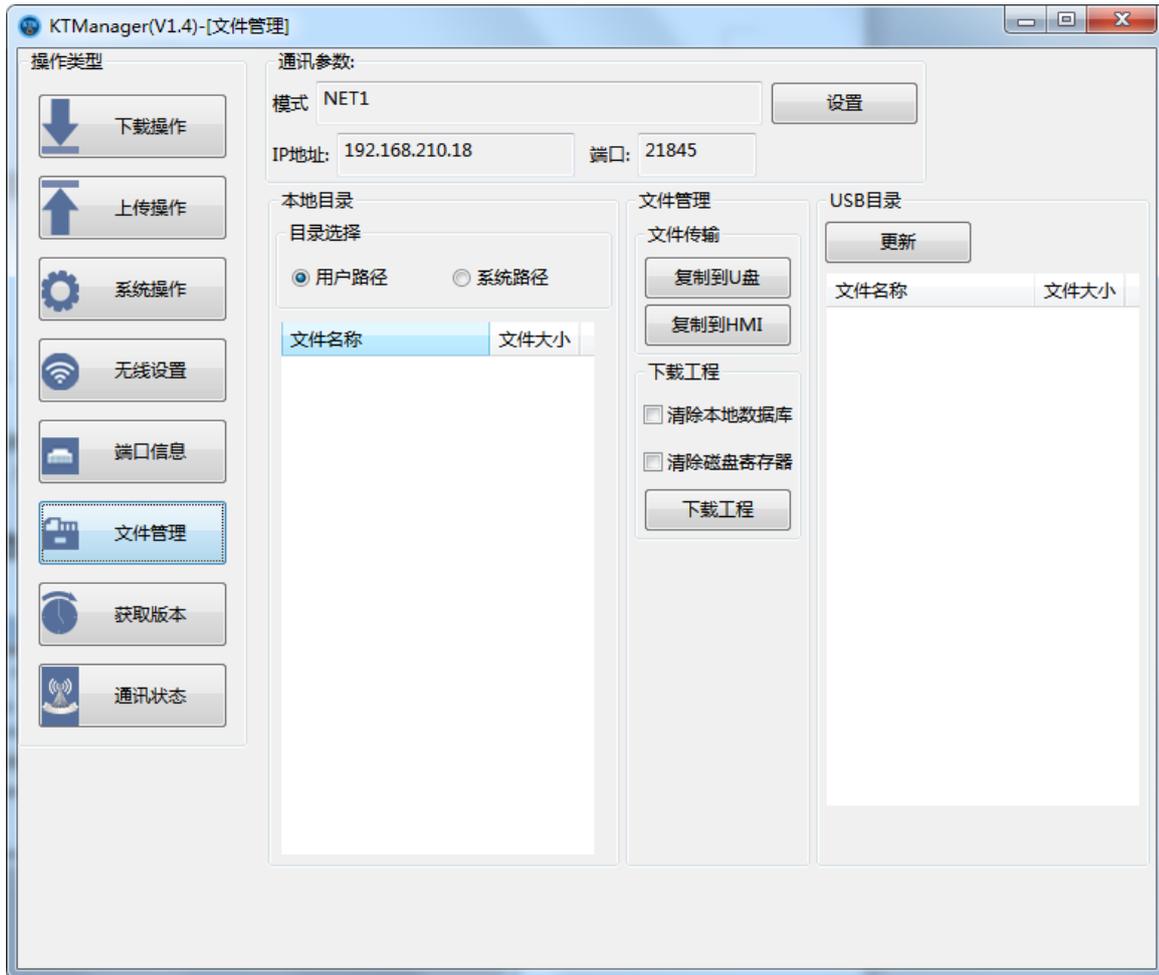
【无线设置】具体说明	
无线信息	连接上无线，会显示无线的状态、SSID,BSSID,IP 地址，MAC 地址，子网掩码、网关 无线可以自动获取 IP,当选择自动获取 IP 地址时，可以更改子网掩码和网关；当选择用户自定义 IP 时，可以更改 IP 地址、子网掩码和网关；
	把当前的无线设置为默认网关
	更新所有的 WiFi 信息
	重置 WIFI:用来当某些无线 MAC 地址没有连接权限时可进行重置、换其他 WiFi 连接
无线网络	开启：开启 WIFI
	扫描：扫描可连接的 WiFi 网络
	连接：连接 WiFi 信号

7.7 端口信息



【端口信息】具体说明	
DO/DI	测试 DIO 口
串口	读取串口 0、串口 1 和串口 2 的信息

7.8 文件管理

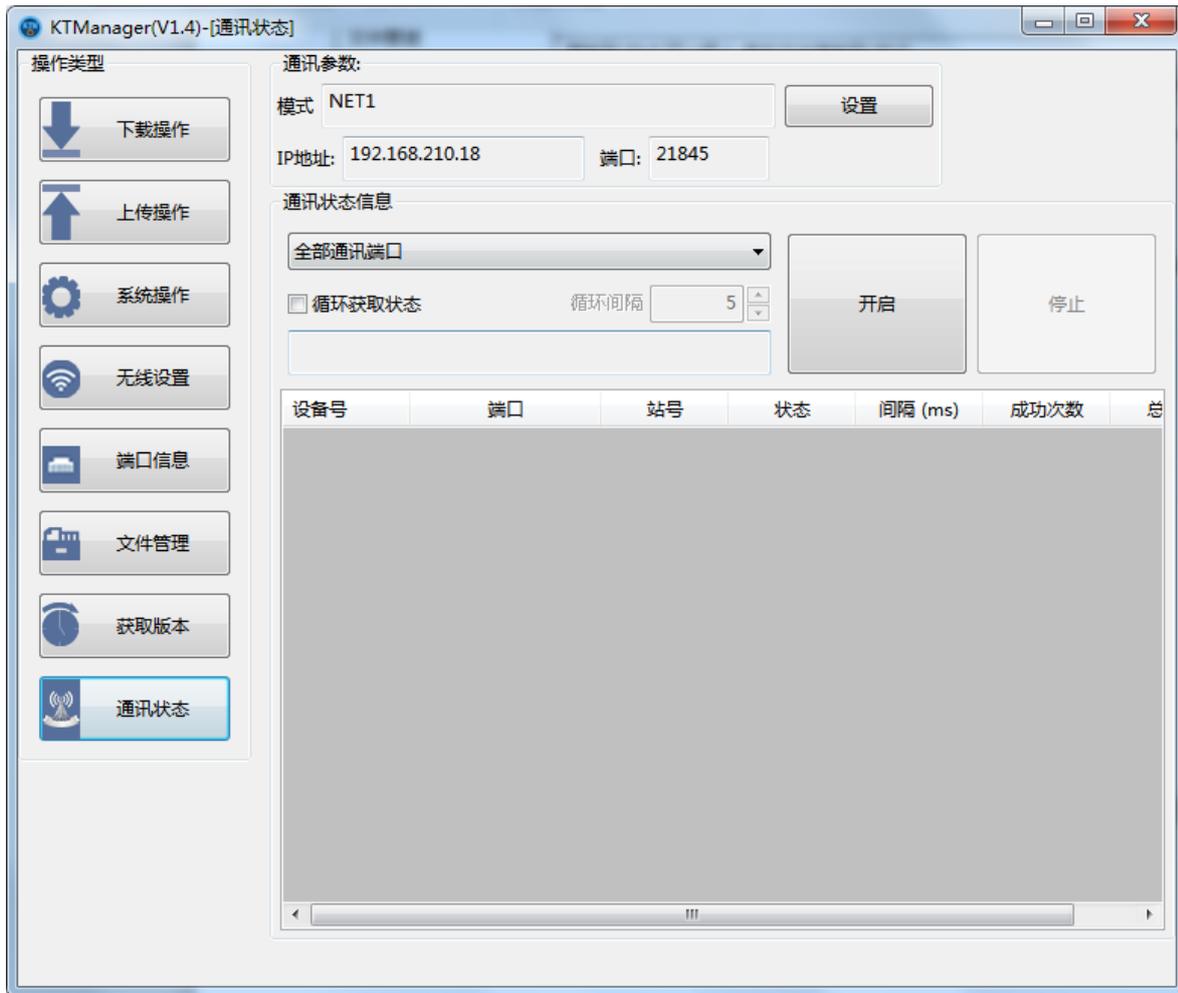


【文件管理】具体说明	
目录选择	用户路径 home/user ,系统路径 home/bin
文件管理	复制到 U 盘 :可以把用户路径下的文件或者系统路径下的文件复制到 U 盘
	复制到 HMI :可以把 U 盘 的文件复制到 HMI
下载工程	选择*.kbp 文件, 可以下载工程, 也可以下载时选择清除本地数据库和磁盘寄存器
USB 目录	显示 U 盘 中的文件, 更新可以刷新 U 盘 文件

7.9 获取版本

可通过 **KTManager** 的【获取版本】来查看 **HMI** 固件版本信息, 点击【获取版本号】, 如获取成功, **HMI** 固件版本信息将显示出来如图所示。如果获取不成功, 需要检查【当前通讯方式】是否正确或其它原因。

7.10 通讯状态



【通讯状态】具体说明	
通讯状态信息	选择监视 HMI 所有的通讯端口或者单个窗口
循环获取状态	选择是否自动循环获取端口状态，设置循环时间
开启	开始监视端口状态
停止	停止监视端口状态
监控信息	查看对应端口的状态：端口、站号、状态（成功或者失败），间隔（PLC 一次收发数据时间）、成功次数、总次数

8. HMI 通信

本章主要介绍 HMI 支持的通信方式以及各种通信方式的配置。

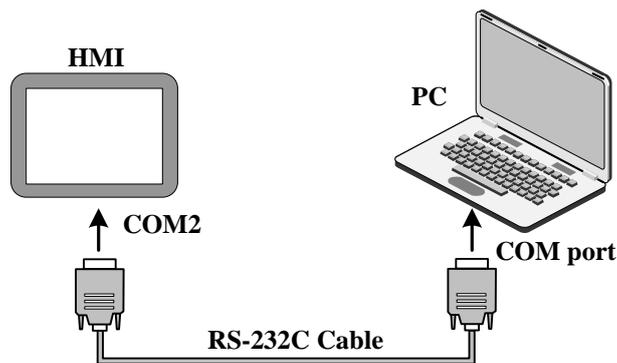
8.1 串口通信

串口，即串行接口的简称，是指数据一位位的顺序传送。串口通信是工业控制中常见的一种通讯方式，其特点是通信线路简单，只要一对传输线就可以实现双向通信，从而降低成本，特别适用于远距离通信，通信的距离可以从几米到几千米，但传送速度较慢。

HMI 支持 RS-232C、RS-485、RS-422A 三种串口通信连接方式。

8.1.1 HMI 与 PC 串口通信

HMI 可通过 RS-232C 串行电缆连接 PC 机侧的串行接口进行用户工程、配方等数据的上传/下载以及 HMI 固件的更新。



[关于 HMI 通过串口上传下载更多详情请参阅【7 KTmanager】](#)

8.1.2 HMI 与 PLC/控制器串口通信

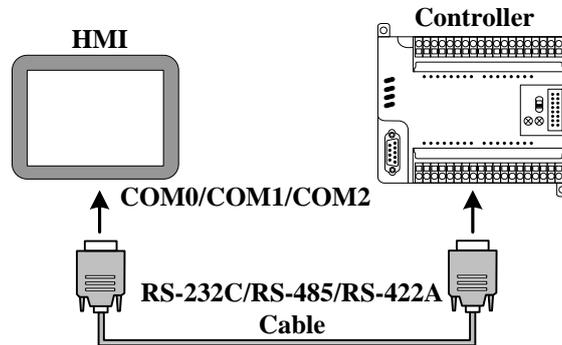
单台 HMI 可通过串口与单台 PLC 通信。

在同一个 COM 口上，可以使用不同或相同的串口通信协议的 PLC/控制器需分别接在 HMI 不同的 COM 口上，且最多可同时和使用三种不同或相同串口通信协议的 PLC/控制器通信，具体根据实际 HMI 的硬件配置决定。



COM2 口只支持 RS232 接口方式。

- 单个 HMI 与单个 PLC/控制器通信



【范例】下面以一台 X10 与一台 SIEMENS S7-200 通讯为例，要求在 HMI 上监视 PLC M0.0 输出的状态。（本例使用 HMI COM0 端口连接 PLC 通讯）

① 配置设备并设置通讯参数

在 **【HMI】—【通讯设置】—【串口 0 设置】** 中设置串口通讯参数，配置 PLC 站号：在 **【PLC 属性】** 中设置 **【站号】** 为 2





HMI 串口通讯参数要与实际 PLC 通讯端口的参数一致，【PLC 属性】中设置的【站号】要与实际 PLC 的站号一致。

② 编辑组态画面

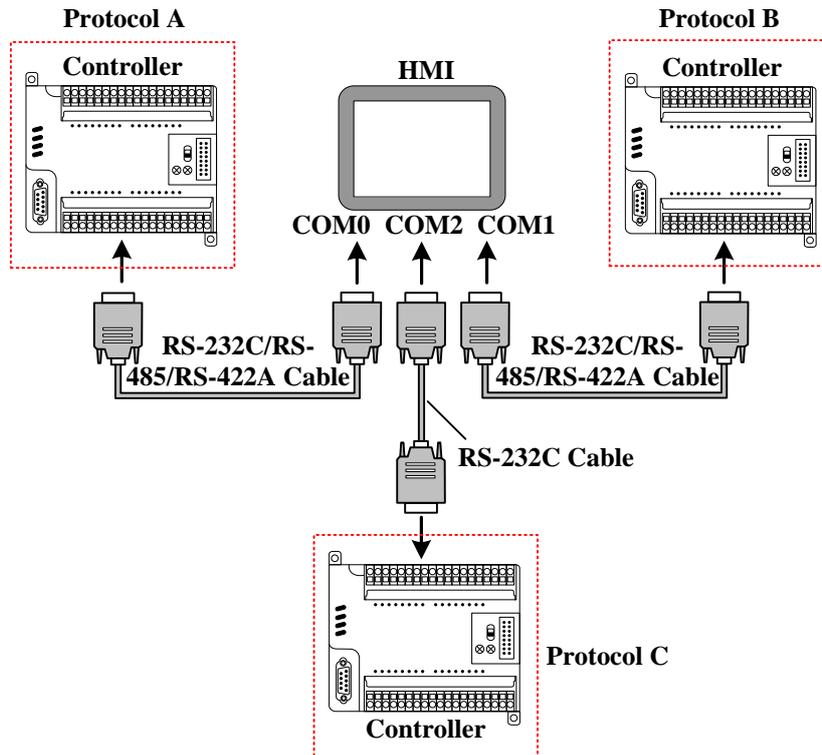
打开组态编辑窗口后，从【绘图菜单栏】—【图片】中选择多态显示，加至组态编辑窗口，并设置元件属性为：



设置完毕，对工程进行保存、编译并下载。

③ 使用 RS-485 通讯电缆连接 HMI 和 PLC，通讯建立成功后即可在 HMI 上监视 M0.0 的状态

- 单个 HMI 与多个 PLC/控制器通信
 - 单个 HMI 与多个 PLC/控制器（不同通信协议）通信



【范例】下面以一台 X10 与一台 SIEMENS S7-200 (以下称为 PLC 0)、一台 Mitsubishi Q06HV2 (以下称为 PLC 1)、一台 OMRON Host Link CJ_CS (以下称为 PLC 2) 通讯为例, 要求在 HMI 上分别监视 PLC 0 的 VW 100、PLC 1 的 D100 中的数据以及 PLC 2 的 D0.0 输出的状态。

(本例使用 HMI COM0 端口连接 PLC 0, COM1 连接 PLC 1, COM2 连接 PLC 2 通讯)

① 在【HMI】—【通讯设置】—【串口 0/1/2 设置】中根据该串口所接 PLC 实际通讯端口的参数来分别设置 HMI 串口的通讯参数, 参数设置如下:

HMI 串口	串口通讯参数设置
COM0	RS485, 9600, 8, 偶校验, 1
COM1	RS422, 115200, 8, 奇校验, 1
COM2	RS232, 9600, 7, 偶校验, 2

在 PLC 0、PLC 1、PLC 2 的【PLC 属性】中根据实际 PLC 的站号来分别设置【站号】:

PLC 编号	站号
PLC 0	2
PLC 1	0
PLC 2	0



连接在 HMI 不同串口上的 PLC/控制器可设置相同或不同的站号, 但设置的站号要与实际连接的 PLC/控制器的站号一致。

② 编辑组态画面

分别从【绘图菜单栏】—【数值元件】中选择两个“数值显示”和一个图片元件添加至组态编辑窗口, 元件属性分别设置为:

数值显示元件_1

PLC 编号*	0
读取地址	VW 100 (PLC 寄存器)
图形	

数值显示元件_2

PLC 编号*	1
读取地址	D 100 (PLC 寄存器)
图形	

位状态指示灯

PLC 编号*	2
读取地址	D0.0 (PLC 寄存器)
图形	<div style="text-align: center;"> <p>状态0 状态1</p>  <p>使用多态显示</p> </div>

设置完毕, 对工程进行保存、编译并下载。

③使用相应的通讯电缆分别连接 HMI 和 PLC 0、PLC 1、PLC 2, 通讯建立成功后即可在 HMI 上监视 VW 100、D100 中的数据以及 D0.0 的状态

8.2 网口通信

以太网具有传输速度快、低能耗、易于安装且兼容性好等优势，因此广泛应用于工业自动化系统控制。

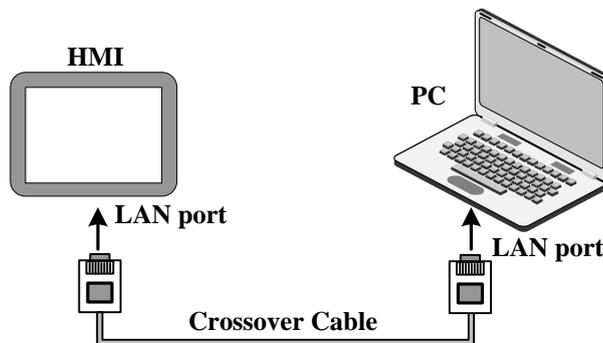
HMI 通过以太网通讯主要有两种方式：

一种是通过两端为 RJ45 接头的直通双绞线 (Straight Through Cable) 或交叉双绞线 (Crossover Cable)，经由集线器 (Hub) 或交换机 (Switch) 通讯；

另一种是通过两端为 RJ45 接头的交叉双绞线 (Crossover Cable)，不经过集线器 (Hub) 或交换机 (Switch) 通讯，这种方式仅适用于 1 对 1 通讯。

8.2.1 HMI 与 PC 网口通信

HMI 可通过两端为 RJ45 接头的交叉双绞线 (Crossover Cable) 和 PC 机侧的网口连接进行用户工程、配方等数据的上传/下载以及 HMI 固件的更新。

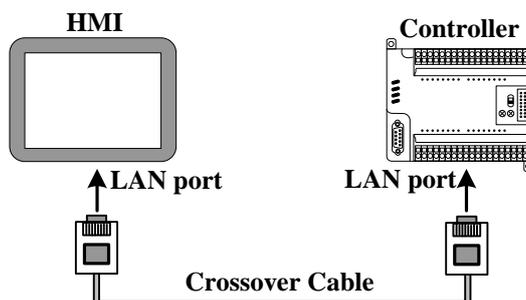


8.2.2 HMI 与 PLC/控制器网口通信

HMI 与 PLC 网口通信主要有下面几种方式：

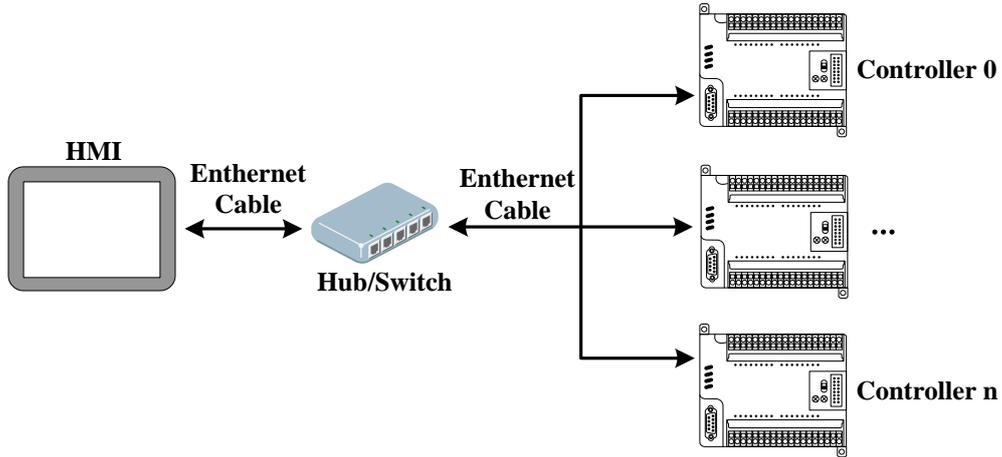
- 单台 HMI 与单台 PLC 网口通讯

单台 HMI 与单台 PLC 经由网口通讯时，可通过两端为 RJ45 接头的交叉双绞线直接连接通讯。



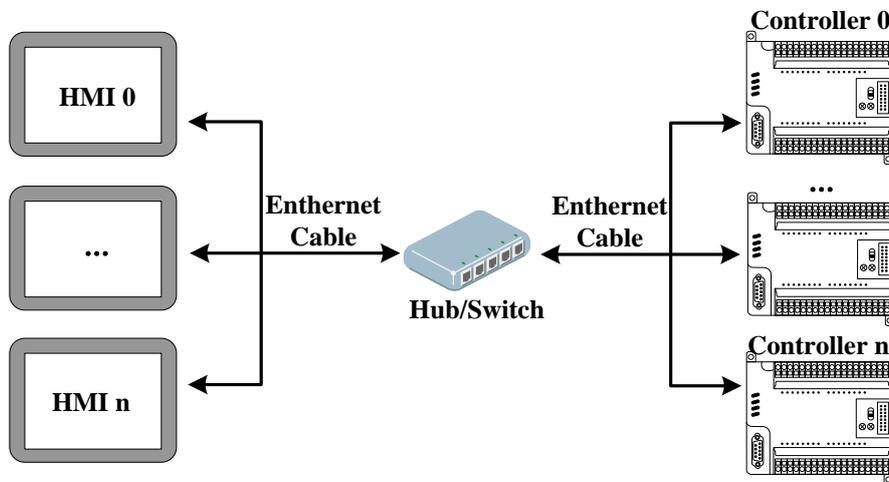
- 单台 HMI 与多台 PLC 网口通讯

单台 HMI 与多台 PLC 经由网口通讯时，需通过 Hub 或 Switch 连接通讯。



- 多台 HMI 与多台 PLC 网口通讯

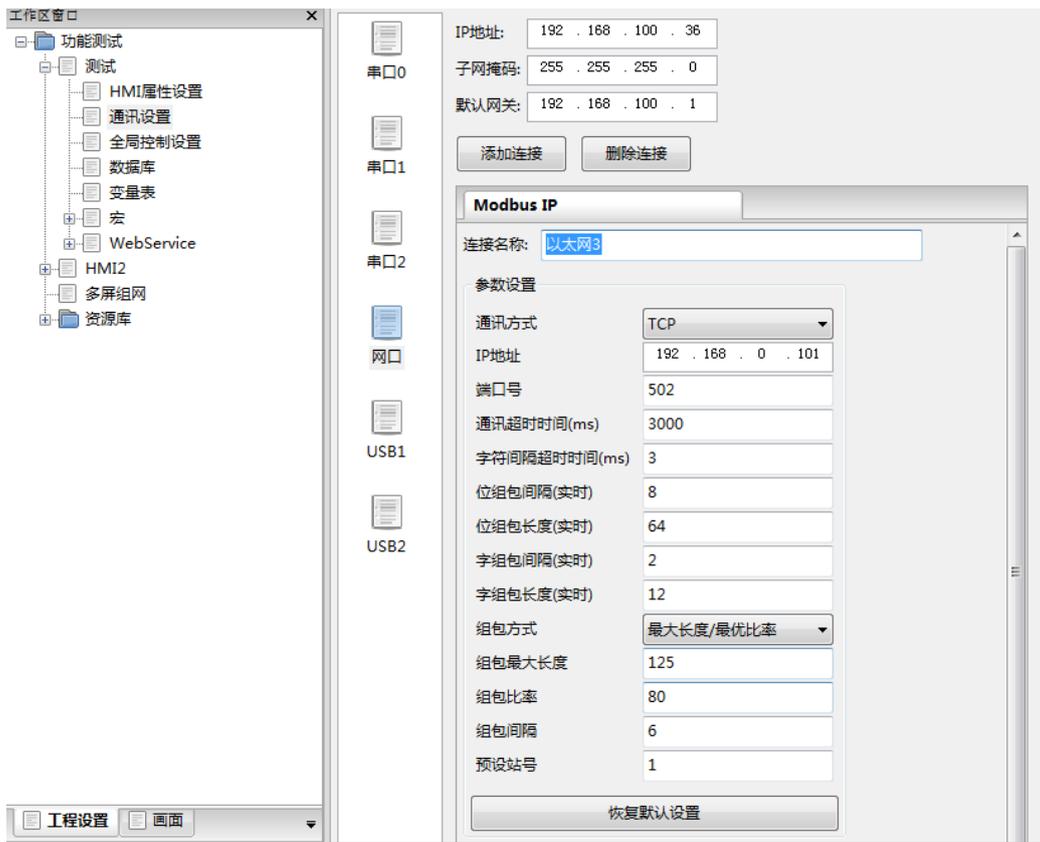
多台 HMI 与多台 PLC 网口通讯时，需通过 Hub 或 Switch 连接通讯。



【举例】下面以两台 X10（以下分别称为 HMI1 和 HMI2）和一台 MODBUS TCP 设备（以下称为 PLC0）、一台西门子 PLC（以下称为 PLC1）通讯为例。要求在 HMI0 上监视 PLC0 中 4x100 的数据以及在 HMI1 上监视 PLC1 中 QW100 的数据。

① 配置设备并设置通讯参数

在 **【HMI】** — **【通讯设置】** 中，点击 **【网口】** 配置 HMI 网口通讯参数，添加连接，此处设置的 IP 地址要与实际 PLC 的 IP 地址一致，配置完毕后，拓扑结构图和总线配置列表如下：



② 编辑组态画面

打开 **HMI1** 组态编辑窗口后，选择“数值显示”元件添加至组态编辑窗口，元件属性设置为：

触摸屏*	1
读取地址	变量地址设置 4x 100 (PLC 寄存器)
图形	

然后在打开 **HMI2** 组态编辑窗口，中选择“数值显示”元件添加至组态编辑窗口，元件属性设置为：

触摸屏*	2
读取地址	变量地址设置 QW 100
图形	

设置完毕，对工程进行保存、编译并下载。

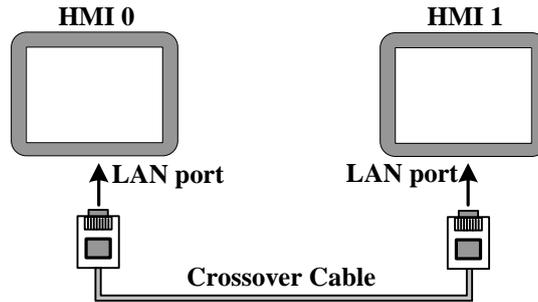
③ 使用交叉或直通网线通过 **Hub** 或 **Switch** 分别连接 **HMI1**、**HMI2**、**PLC0**、**PLC1**，通讯建立成功后即可在 **HMI1** 上监视 **PLC0** 的 **4x100** 的数据以及在 **HMI2** 上监视 **PLC1** 的 **QW100** 的数据。

8.2.3 HMI 与 HMI 网口通信

HMI 与 **HMI** 网口通信主要有下面两种方式：

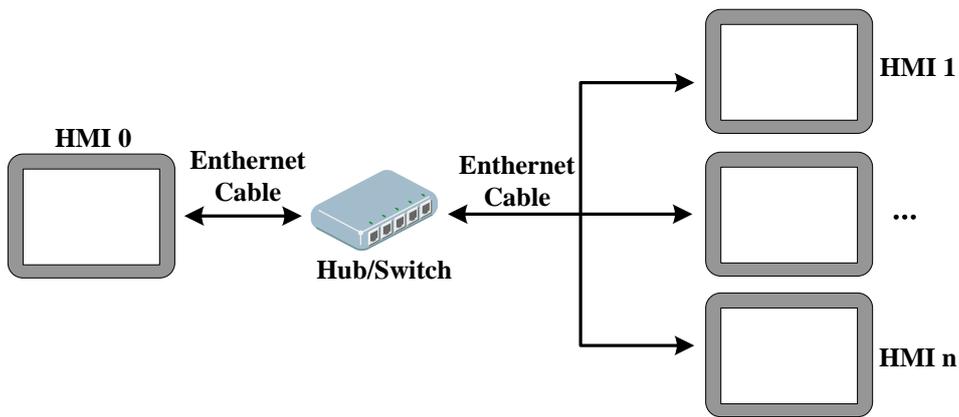
- 单台 HMI 与单台 HMI 通信

单台 HMI 与单台 HMI 经由网口通讯时，可通过两端为 RJ45 接头的交叉双绞线直接连接通讯。



- 多台（两台以上）HMI 组网通信

多台 HMI 经由网口通讯时，需通过 Hub 或 Switch 连接通讯。



【范例】下面以两台 X10 通讯为例，要求在 HMI1 画面上控制 HMI2 的内存位寄存器地址 0 状态输出。

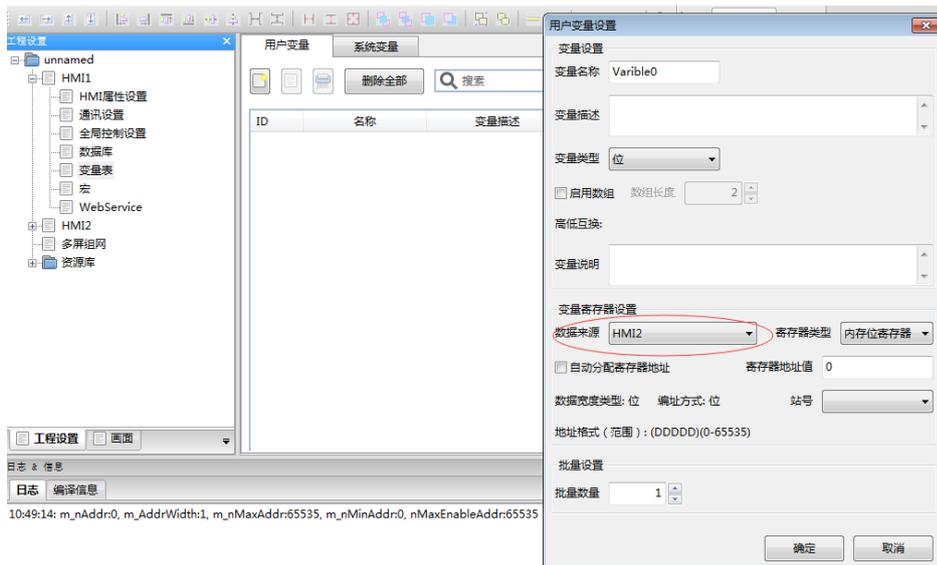
① 在拓扑结构窗口配置设备并设置通讯参数

在【工程设置】—【多屏组网】中，点击【增加】配置 HMI 网口通讯参数：

	HMI ID	名称	描述	IP
0	1	HMI1		192.168.210.100
1	2	HMI2		192.168.210.122

② 编辑组态画面

在拓扑结构窗口双击 HMI1 图标，添加内存位寄存器变量 0，来源选择 HMI2,如下图



※多屏组网时需通过更改变量属性中的【数据来源】来区分控制对象。

界面上放置内存位寄存器 0 的切换状态设置完毕，对工程进行保存、编译并下载。

③使用交叉网线连接 HMI1 和 HMI2，通讯建立成功后即可在 HMI2 上放置内存位寄存器 0 的数值显示元件，并下到对应的屏中，即可控制 HMI2 的内存位寄存器地址 0 状态输出。

注意：1.6 软件的工程，如果使用软件编译组网工程的 HMI 必须都要下载。

8.3 MODBUS 协议应用（主从模式）

MODBUS 协议是应用于电子控制器上的一种通用语言。通过此协议，控制器相互之间、控制器经由网络（例如以太网）和其它设备之间可以通信。现今，MODBUS 协议已经成为一种通用工业标准，通过它，不同厂商生产的控制设备可以连成工业网络，从而进行集中监控。

8.3.1 MODBUS 协议概述

Modbus 是一个请求/应答协议，并且提供功能码规定的服务。

Modbus 协议包括 ASCII、RTU、TCP 等，并没有规定物理层。协议定义了控制器能够认识和使用的消息结构，而不管它们是经过何种网络进行通信的。Modbus 的 ASCII、RTU 协议规定了消息、数据的结构、命令和对答的方式，数据通讯采用 Master（主站）/Slave（从站）方式，主站发出数据请求消息，从站接收到正确消息后就可以发送数据到主站以响应请求；主站也可以直接发消息修改从站的数据，实现双向读写。

MODBUS 规定，只有主站具有主动权，从站只能被动的响应，包括回答出错信息。

MT 系列 HMI 实现 Modbus 协议通讯时，遵循标准的 Modbus 通信过程：

通用数据帧格式如下：

地址码	功能码	数据区	错误校验码
8Bits	8Bits	N × 8Bits	16Bits

- 地址码

消息帧的地址域包含两个字符（ASCII）或 8Bit（RTU）。可能的从设备地址是 0...247（十进制）。单个设备的地址范围是 1...247。主设备通过将要联络的从设备的地址放入消息中的地址域来选择从设备。当从设备发送回应消息时，它把自己的地址放入回应的地址域中，以便主设备知道是哪个设备做出回应。地址 0 是用作广播地址，以使所有的从设备都能认识。

- 功能码

在 HMI 系统中，常用的功能码如下：

Modbus 功能码	名称	功能	对应的地址类型
01	读线圈状态	读位（读 N 个 Bits）	0x
02	读输入离散量	读位	1x
03	读多个寄存器	读整型、字符型、状态字、浮点型（读 N 个 Words）	4x
04	读输入寄存器	读整型、状态字、浮点型	3x
05	写单个线圈	写位（写一个 Bit）	0x
06	写单个寄存器	写整型、字符型、状态字、浮点型（写一个 Word）	4x
15	写多个线圈	写位（写 N 个 Bits）	0x
16	写多个寄存器	写整型、字符型、状态字、浮点型（写 N 个 Words）	4x

- 数据区

数据区包含了终端执行特定功能所需要的数据或者终端响应查询时采集到的数据。这些数据的内容可能是数值、参考地址或者设置值。例如：功能码告诉终端读取一个寄存器，数据区则需要指明从哪个寄存器开始及读取多少个数据，内嵌的地址和数据类型根据从机的内容不同而有所不同。

- 错误校验码

Modbus 协议需要对数据进行校验，串行协议中除有奇偶校验外，ASCII 模式采用 LRC 校验，RTU 模式采用 16 位 CRC 校验，但 TCP 模式没有额外规定校验。

- 奇偶校验

用户可以配置控制器是奇或偶校验，或无校验。这将决定每个字符中的奇偶校验位是如何设置的。

如果指定了奇或偶校验，“1”的位数将算到每个字符的位数中（ASCII 模式 7 个数据位，RTU 中 8 个数据位）。例如 RTU 字符帧中包含以下 8 个数据位：11000101；整个“1”的数目是 4 个。如用偶校验，帧的奇偶校验位将是 0，使得整个“1”的个数仍是 4 个。如用奇校验，帧的奇偶校验位将是 1，使得整个“1”的个数是 5 个。如果没有指定奇偶校验位，传输时就没有校验位，也不进行校验检测。代替附加的停止位填充至要传输的字符帧中。

- LRC 检测

使用 ASCII 模式，消息包括了基于 LRC 方法的错误检测域。LRC 域检测了消息域中除开始的冒号及结束的回车换行号外的内容。

LRC 域是个包含一个 8 位二进制值的字节。LRC 值由传输设备来计算并放到消息帧中，接收设备在接收消息的过程中计算 LRC，并将它和接收到消息中 LRC 域中的值比较，如果两值不等，说明有错误。

LRC 方法是将消息中的 8Bit 的字节连续累加，丢弃了进位。

LRC 简单函数如下:

```
static unsigned char LRC(auchMsg,usDataLen)
unsigned char *auchMsg ; /* 要进行计算的消息 */
unsigned short usDataLen ; /* LRC 要处理的字节的数量*/
{
    unsigned char uchLRC = 0 ; /* LRC 字节初始化 */
    while (usDataLen--) /* 传送消息 */
        uchLRC += *auchMsg++ ; /* 累加*/
    return ((unsigned char)-((char_uchLRC))) ;
}
```

➤ CRC 检测

使用 RTU 模式，消息包括了基于 CRC 方法的错误检测域。CRC 域检测了整个消息的内容。

CRC 域是两个字节，包含一 16 位的二进制值。由传输设备计算后加入到消息中。接收设备重新计算收到消息的 CRC，并与接收到的 CRC 域中的值比较，如果两值不同，则有误。

CRC 添加到消息中时，低字节先加入，然后高字节。CRC 简单函数如下:

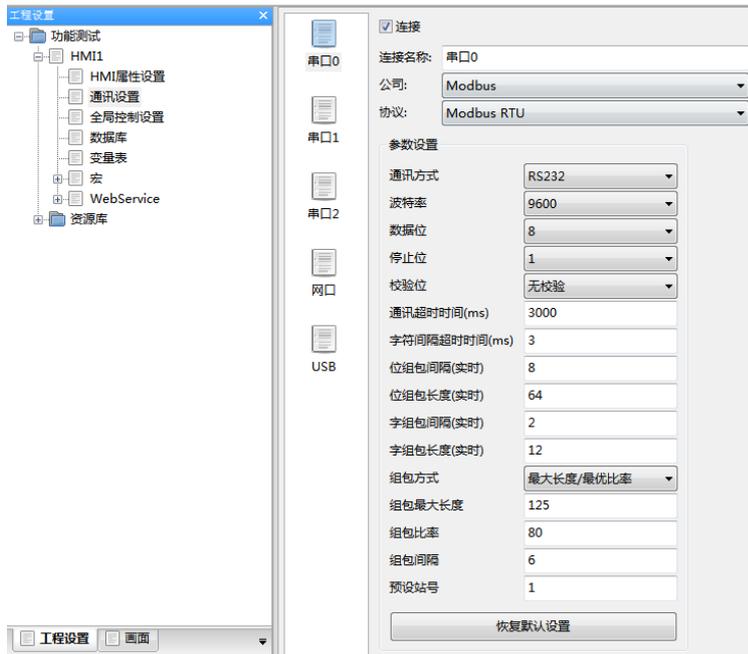
```
unsigned short CRC16(puchMsg, usDataLen)
unsigned char *puchMsg ; /* 要进行 CRC 校验的消息 */
unsigned short usDataLen ; /* 消息中字节数 */
{
    unsigned char uchCRCHi = 0xFF ; /* 高 CRC 字节初始化 */
    unsigned char uchCRCLo = 0xFF ; /* 低 CRC 字节初始化 */
    unsigned uIndex ; /* CRC 循环中的索引 */
    while (usDataLen--) /* 传输消息缓冲区 */
    {
        uIndex = uchCRCHi ^ *puchMsgg++ ; /* 计算 CRC */
        uchCRCHi = uchCRCLo ^ uchCRCHi[uIndex] ;
        uchCRCLo = uchCRCLo[uIndex] ;
    }
    return (uchCRCHi << 8 | uchCRCLo) ;
}
```

8.3.2 MODBUS 协议通讯格式

下面通过几个实例，并使用串口调试工具监听 HMI 收发的数据帧，来进一步了解 Modbus 协议的通讯格式。

以下示例均以 HMI 作主设备，PLC 作从设备，并使用 RTU 传输方式进行说明:

拓扑结构示意图如下图所示:



(1) 01 功能码应用

用功能码 01 读取 0x10、0x11、0x12、0x13、0x14、0x15 这 6 个 Bit 的状态。

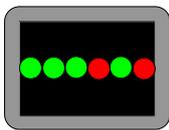
在 HMI 组态上放置六个“位状态指示灯”元件，读取地址分别设置为 0x10~0x15。

运行时，主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	读取 Bit 值 (高字节)	读取 Bit 值 (低字节)	CRC 高字节	CRC 低字节
01H	01H	00H	09H	00H	06H	6CH	0AH

从机 (PLC) 应答

站号	功能码	字节数	数据单字节	CRC 高字节	CRC 低字节
01H	01H	01H	3A	D1H	9BH

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 01H 01H 3AH D1H 9BH	

HMI 读 0x15 为 ON, 0x14 为 ON, 0x13 为 ON, 0x12 为 OFF, 0x11 为 ON, 0x10 为 OFF。十六进制数 3A 转换成二进制为 111010, 正好与之对应。

(2) 02 功能码应用

用功能码 02 读 1x999 这个 Bit 的状态。

在 HMI 组态上放置一个“位状态指示灯”元件，读取地址设置为 1x999。

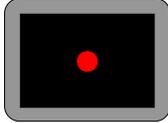
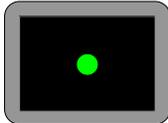
运行时，主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	读取 BIT 值 (高字节)	读取 BIT 值 (低字节)	CRC 高字节	CRC 低字节
01	02H	03H	E6H	00H	01H	58H	79H

H							
---	--	--	--	--	--	--	--

从机 (PLC) 应答

站号	功能码	字节数	数据 (单字节)	CRC 高字节	CRC 低字节
01H	02H	01H	00H	A1H	88H
01H	02H	01H	01H	60H	48H

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 02H 01H 00H A1H 88H	
01H 02H 01H 01H 60H 48H	

(3) 03 功能码应用

用功能码 03 读取 4x 208~215 这 8 个 Words 的值。

在 HMI 组态上放置 8 个“数值显示”元件，读取地址分别设置为 4x 208~215。

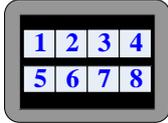
运行时，主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	读取 WORD 值 (高字节)	读取 WORD 值 (低字节)	CRC 高字节	CRC 低字节
01H	03H	00H	CFH	00H	08H	74H	33H

从机 (PLC) 应答

站号	功能码	字节数	数据 (1) 高字节	数据 (1) 低字节	数据 (2) 高字节	数据 (2) 低字节	数据 (3) 高字节	数据 (3) 低字节	数据 (4) 高字节	数据 (4) 低字节
01H	03H	10H	00H	01H	00H	02H	00H	03H	00H	04H

数据 (5) 高字节	数据 (5) 低字节	数据 (6) 高字节	数据 (6) 低字节	数据 (7) 高字节	数据 (7) 低字节	数据 (8) 高字节	数据 (8) 低字节	CRC 高字节	CRC 低字节
00H	05H	00H	06H	00H	07H	00H	08H	72H	98H

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 03H 10H 00H 01H 00H 02H 00H 03H 00H 04H 00H 05H 00H 06H 00H 07H 00H 08H 72H 98H	

(4) 04 功能码应用

用功能码 04 读 3x 1023 这个 Word 的值。

在 HMI 组态上放置一个“数值显示”元件，读取地址设置为 3x 1023。

运行时，主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	读取 WORD 值 (高字节)	读取 WORD 值 (低字节)	CRC 高字节	CRC 低字节
----	-----	-----------	-----------	-----------------	-----------------	---------	---------

01H	04H	03H	FEH	00H	01H	50H	7EH
-----	-----	-----	-----	-----	-----	-----	-----

从机 (PLC) 应答

站号	功能码	字节数	数据高字节	数据低字节	CRC 高字节	CRC 低字节
01H	04H	02H	04H	D2H	3BH	ADH

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 04H 02H 04H D2H 3BH ADH	

(5) 05 功能码应用

用功能码 05 修改 0x1 这一个 Bit 的状态。

在 HMI 组态上放置一个“位状态设定”元件，写入地址设置为 0x1，开关类型设为“切换开关”。

运行时，在 HMI 上点击“位状态设定”元件将其切换为 ON 状态，则串口调试工具接收的数据为 01 05 00 00 FF 00 8C 3A；将其切换为 OFF 状态，则串口调试工具接收的数据为 01 05 00 00 00 00 CD CA；（其中，十六进制值 0XFF00 请求线圈为 ON。十六进制值 0X0000 请求线圈为 OFF）同时，串口调试工具要回复相同的数据。

主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 Bit 值 (高字节)	写入 Bit 值 (低字节)	CRC 高字节	CRC 低字节
01H	05H	00H	00H	FFH	00H	8CH	3AH
01H	05H	00H	00H	00H	00H	CDH	CAH

从机 (PLC) 应答

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 Bit 值 (高字节)	写入 Bit 值 (低字节)	CRC 高字节	CRC 低字节
01H	05H	00H	00H	FFH	00H	8CH	3AH
01H	05H	00H	00H	00H	00H	CDH	CAH

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 05H 00H 00H FFH 00H 8CH 3AH	
01H 05H 00H 00H 00H 00H CDH CAH	

(6) 06 功能码应用

用功能码 06 写 4x1 这一个 Word 的值。

在 HMI 组态上放置一个“多状态设定”元件，写入地址设置为 4x1，设定方式为“设置常数”，设置值为 88。

运行时，在 HMI 上点击“多状态设定”元件，则串口调试工具接收的数据为 **01 06 00 00 00 58 88 30**；同时，串口调试工具要回复相同的数据。

主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 WORD 值 (高字节)	写入 WORD 值 (低字节)	CRC 高字节	CRC 低字节
01H	06H	00H	00H	00H	58H	88H	30H

从机 (PLC) 应答

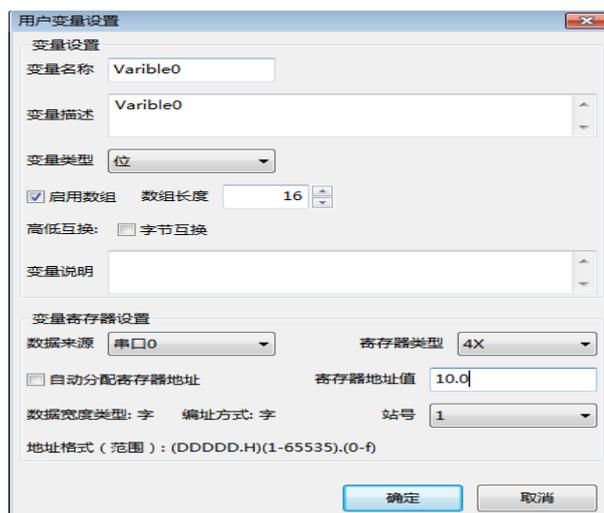
站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 WORD 个数 (高字节)	写入 WORD 个数 (低字节)	CRC 高字节	CRC 低字节
01H	06H	00H	00H	00H	58H	88H	30H

从串口调试工具向 HMI 发送数据包	HMI 显示
01H 06H 00H 00H 00H 58H 88H 30H	

(7) 15 功能码应用

用功能码 15 修改 0x 10~25 这 16 个 Bit 的状态。

在 HMI 组态上建立一个宏代码：15 功能码.c，变量定义如下：



代码如下：

```

24  int MacroEntry()
25  {
26  short i;
27  for(i=0;i<16;i++)
28  data[i]=1;
29  return 0;
30  }
31

```

然后在组态画面上放置一个“功能键”元件，功能设置为“执行宏代码：15 功能码.c”。

运行时，在 HMI 上点击功能键，则串口调试工具上接收的数据为 **01H 0FH 00H 09H 00H 10H 02H FFH FFH E3H**

09H，同时在串口调试工具，回复数据 01H 0FH 00H 09H 00H 10H 84H 05H。

主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 Bit 值(高字节)	写入 Bit 值(低字节)	写入 Bit 字节数	Bit 数据 高字节	Bit 数据 低字节	CRC 高字节	CRC 低字节
01H	0FH	00H	09H	00H	10H	02H	FFH	FFH	E3H	09H

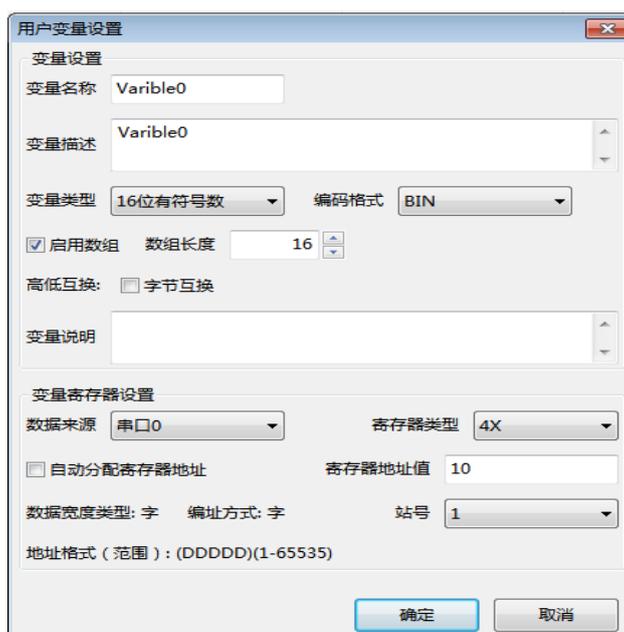
从机 (PLC) 应答

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 BIT 值 (高字节)	写入 BIT 值 (低字节)	CRC 高字节	CRC 低字节
01H	0FH	00H	09H	00H	10H	84H	05H

(8) 16 功能码应用

用功能码 16 写 4x 10~25 这 16 个 Words 的值。

在 HMI 组态上建立一个宏代码：16 功能码.c，变量定义如下：



代码如下：

```

24  int MacroEntry()
25  {
26  short i;
27  for(i=0;i<16;i++)
28  data[i]=i+1;
29  return 0;
30  }
31

```

然后在组态画面上放置一个“功能键”元件，功能设置为“执行宏代码：16 功能码.c”。

运行时，在 HMI 上点击功能键，则串口调试工具上接收的数据为 01H 10H 00H 09H 00H 10H 20H 00H 01H 00H 02H 00H 03H 00H 04H 00H 05H 00H 06H 00H 07H 00H 08H 00H 09H 00H 0AH 00H 0BH 00H 0CH 00H 0DH 00H 0EH 00H 0FH 00H 10H 88H 37H。同时在串口调试工具，回复数据 01H 10H 00H 09H 00H 10H 11H C7H。

主机 (HMI) 请求

站号	功能码	首地址 (高字节)	首地址 (低字节)	写 WORD 值 (高字节)	写 WORD 值 (低字节)	写 WORD 字节数	数据 (1) 高字节	数据 (1) 低字节
01H	10H	00H	09H	00H	10H	20H	00H	01H

数据 (3) 高字节	数据 (3) 低字节	数据 (4) 高字节	数据 (4) 低字节	数据 (5) 高字节	数据 (5) 低字节	数据 (6) 高字节	数据 (6) 低字节	数据 (7) 高字节	数据 (7) 低字节
00H	03H	00H	04H	00H	05H	00H	06H	00H	07H

数据(8) 高字节	数据(8) 低字节	数据(9) 高字节	数据(9) 低字节	数据(10) 高字节	数据(10) 低字节	数据 (11) 高字节	数据 (11) 低字节	数据 (12) 高字节	数据 (12) 低字节
00H	08H	00H	09H	00H	0AH	00H	0BH	00H	0CH

数据(13) 高字节	数据(13) 低字节	数据(14) 高字节	数据(14) 低字节	数据(15) 高字节	数据(15) 低字节	数据(16) 高字节	数据(16) 低字节	CRC 高 字节	CRC 低 字节
00H	0DH	00H	0EH	00H	0FH	00H	10H	88H	37H

从机 (PLC) 应答

站号	功能码	首地址 (高字节)	首地址 (低字节)	写入 WORD 值 (高字节)	写入 WORD 值 (低字节)	CRC 高 字节	CRC 低 字节
01H	10H	00H	09H	00H	10H	11H	C7H

8.3.3 MODBUS 协议在 HMI 中的应用

在人机界面设计应用中，经常碰到需要与自行设计的单片机控制板通讯的情况。在这些应用环境下，单片机控制系统的设计者希望有一种简单方便，稳定可靠的通讯方式与人机界面交换数据。而 MODBUS 协议是 HMI 与设备间实现主从通信模式最常用的一种方式。

下面以单片机（以下称为 MCU）和 HMI 通过 MODBUS 协议（以 RTU 传输模式为例）通信为例，来介绍如何使用 MODBUS 协议实现 HMI 与 MCU 间通信。

HMI 与 MCU 间通过 MODBUS 协议通信主要有下面两种主从方式：

主/从	使用通讯协议	优点	缺点
HMI 作主设备 MCU 作从设备	Modbus RTU	HMI 直接访问 MCU 映射出来的 0X, 1X, 3X, 4X 寄存器。	MCU 作为从设备，必须不断响应 HMI 的查询，CPU 占用率高；MCU 须采用中断方式响应，编程相对困难。
MCU 作主设备 HMI 作从设备	Modbus RTU Slave	MCU 作为主设备，可以主动发起通讯，不必占用过多 CPU 时间。 HMI 本地地址 LW, LB 被映射为 4X, 0X 寄存器，由 MCU 写入或读取。	HMI 组态上 LW 数据被修改时，必须通过 MCU 的定期查询才能被读出，实现起来不方便。 MCU 只能访问 HMI 本地字 (LW)，本地位 (LB)

(1) HMI 作主，MCU 作从（使用 Modbus RTU 协议）

MCU 与 HMI 使用标准的 Modbus RTU 协议进行通信，HMI 为主控端，MCU 为从属端，在 MCU 中，需要编写通

信中断服务程序，处理来自 HMI 的通讯请求。MCU 须支持标准 Modbus RTU 协议，同时 MCU 须映射出 0X, 1X, 3X, 4X 寄存器（其中 0X, 1X 为位寄存器；3X, 4X 为字寄存器）

通讯的推荐参数为 19200, 8 数据位, 1 停止位, 无校验位。协议支持除 7 数据位之外的所有设置组合。波特率范围 1200 – 115200, 可以根据需要灵活选择。

- 通讯处理流程

首先, HMI 发送一个请求给 MCU, MCU 接受请求之后, 给 HMI 回复一个响应。MCU 能访问 HMI 中 4X, 3X, 0X, 1X 寄存器, 地址范围均为 1~65535。

	<p>1. MODBUS 协议相对复杂, 超时要求严格</p> <p>2. 在通过程中, HMI 不断发送请求数据包给 MCU, MCU 接受请求之后, 需不断给 HMI 回复一个响应。MCU 的响应是采用中断方式给予的, 因此, MCU 的 CPU 占用率高。MCU 的 CPU 资源大部分在通讯响应过程中消耗。故在实际应用中需给予考虑, 如果 MCU 除了做通讯功能用外, 还有其他的响应功能, HMI (主) MCU (从), 使用 Modbus RTU 协议的解决方案不推荐使用, 推荐使用 HMI (从) MCU (主), 使用 Modbus RTU Slave 协议的解决方案</p>
---	--

- HMI 上电通讯处理流程

当 HMI 上电后

- (1) 如果组态工程页面中没有放置任何元件, 则 HMI 不会对外发送任何通讯数据包
- (2) 如果组态工程页面中放置了元件, 且这些元件地址类型均为本地类型 (LW, RW, RWI, LB, RB, RBI, LW.B), 则 HMI 不会对外发送任何通讯数据包
- (3) 如果组态工程页面中放置了元件, 且这些元件地址类型不为本地类型, 为 0X, 1X, 3X, 4X, 则 HMI 不断扫描组态页面的这些寄存器, 不断对外发送通讯数据包
- (4) 当使用定时器元件进行位元件批量写 (连续地址传输个数大于 1) 的时候, HMI 所发数据包中会封装功能码: **0x0F**
- (5) 当使用定时器元件进行字元件批量写 (连续地址传输个数大于 1) 的时候, HMI 所发数据包中会封装功能码: **0x10**
- (6) 使用配方元件进行下载操作 (即把 HMI 中本地字寄存器 RW 数据下载到 MCU 相应寄存器中, 比如 4X)。当下载操作的数据长度大于 1 时, HMI 所发数据包中会封装功能码: **0x10**

- 数据通讯包

假设在 HMI 组态画面上放置 25 个地址类型为 0X 的寄存器, 起始地址: 20, HMI 站号为 1

HMI 发包: (“H”表示 16 进制形式)

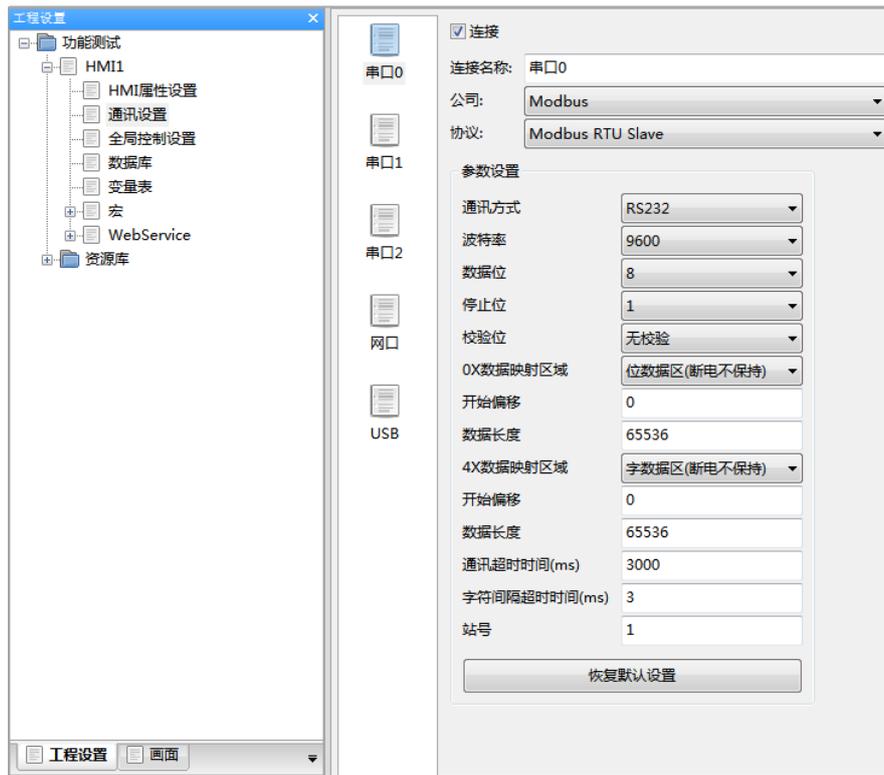
01H 01H 00H 13H 00H 25H XXH XXH (其中: “XX”表示 LRC/CRC)

MCU 返回包:

01H 01H 05H CDH 6BH B2H 0EH 1BH XXH XXH (其中: “XX”表示 LRC/CRC)

- HMI 通讯参数配置

HMI 端口参数设置：



根据实际物理连接方式选择设置【通讯类型】。

波特率，数据位，奇偶校验，停止位的设置必须和所连接的MCU相应参数一致。

其它参数说明如下：

参数	说明
PLC 通讯超时时间	HMI 等待 MCU 响应, 如果 MCU 在该设置时间间隔内没有给予响应, 则 HMI 认为通讯超时, 然后继续发请求包, 继续等待

协议超时时间 1	<p>字符间超时。在 HMI 与 MCU 的通讯过程中，如果通讯帧中字符与字符间的时间间隔超过该设置时间间隔，则 HMI 认为这次通讯过程结束，然后，去处理该通讯数据帧，解析数据包。</p> <p>对于 Modbus 协议，程序内部应根据波特率计算一个估计值。如下：</p> <p>协议超时时间1 = 1.5*1000000*(1 +数据位+奇偶校验+ 停止位)/ 波特率 + set_adjust_time_bps (波特率) ；</p> <p>示例代码（标准C语言）：</p> <pre> int speed_arr[] = { B115200, B57600, B38400, B19200, B9600, B4800, B2400, B1200, B300,B150, B134, B110, B75 }; int name_arr[] = {115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200, 300, 150, 134, 110, 75 }; int adjust_time_bps_arr[]={1000, 4000, 10000, 11000,25000, 40000, 70000, 140000, 600000, 0, 0, 0, 0}; </pre> <p>功能：设定某一波特率下校验时间的调整值</p> <p>参数：baud_rate 波特率</p> <p>返回：当前波特率下的校验时间的调整值</p> <pre> int set_adjust_time_bps(int baud_rate) { int adjust_time; int i; for(i=0;i<sizeof(speed_arr)/sizeof(int);i++) { if(baud_rate==name_arr[i]) { adjust_time=adjust_time_bps_arr[i]; break; } } return adjust_time; } </pre>
协议超时时间 2	改大这个值会减慢通讯速度，减低通讯错误，减少错包。该参数不建议用户自行修改
组包最大字（位） 寄存器间隔	<p>主要发生在 HMI 读数据阶段，为了提高通讯效率，相连比较紧的寄存器组包到一起，形成一个批量读，例如 D1，D2，D4，组包间隔设为 10，这 3 个寄存器就会当作一个批量读（从 D1 起批量读 4 个寄存器）。注意：组包间隔一定要小于批量读写的最大长度</p>
批量传输最大字 （位）寄存器个数	协议支持最大的批量读写寄存器个数。此功能的实现可用数据传输元件实现

PLC 站号设置：

(2) MCU 作主，HMI 作从（使用 Modbus RTU Slave 协议）

MCU 与 HMI 使用标准的 Modbus RTU 协议进行通信，MCU 为主控端，HMI 为从属端，在 MCU 中，只需编写简单的通信读/写程序，而不用编写通信中断服务程序。HMI 本地地址 LW，LB 被映射为 4X，0X 寄存器，由 MCU 写入或读取。

寄存器对应关系表如下：

Modbus 寄存器地址范围	对应 HMI 本地地址范围
0X (1---9000)	内存/磁盘位寄存器 (0---1048575)
4X (1---9000)	内存/磁盘字寄存器 (0---65535)

通讯的推荐参数为 19200，8 数据位，1 停止位，无校验位。协议支持除 7 数据位之外的所有设置组合。波特率范围 1200 – 115200，可以根据需要灵活选择。

- 通讯处理流程

首先，MCU 发送一个请求给 HMI，HMI 接受请求之后，给 MCU 回复一个响应。

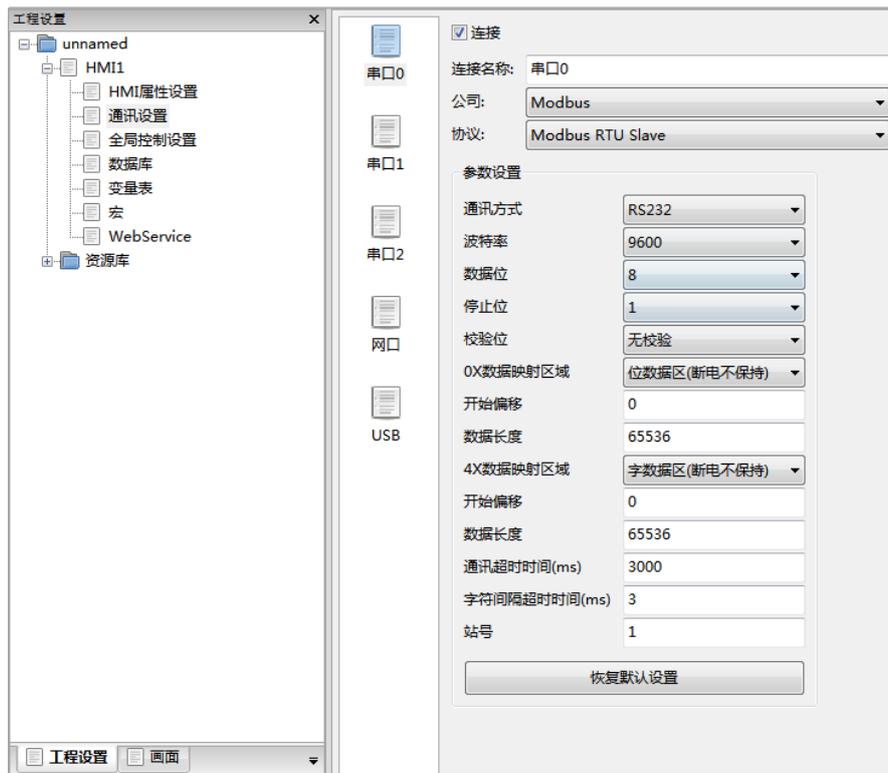
	<p>1. MODBUS 协议相对复杂，超时要求严格</p> <p>2. 在通讯过程中，MCU 作为通讯的发起端，主动发起通讯。当 HMI 接受请求之后，给 MCU 回复一个响应。因此，MCU 不需要不断触发中断，不必占用过多 CPU 时间</p> <p>故在实际应用中，如果 MCU 除了做通讯功能外，还有其他的响应功能（推荐使用这个解决方案）。</p> <p>3. HMI 组态上 LW，LB 数据被修改时，必须通过 MCU 的定期查询才能被读出，实现起来不是很方便。故实际应用中，在 MCU 编写通信读/写程序时，应考虑请求 HMI 中数据的时间间隔，在程序中写明等待多少时间给予通讯请求</p>
--	--

- HMI 上电通讯处理流程

在此方案中，用户编辑组态工程时，元件地址类型只能选择本地字寄存器 LW 和本地位寄存器 LB。当 HMI 上电后，HMI 不主动向外出发通讯数据包，HMI 处于等待接收状态。而当 MCU 发请求之后，HMI 才回复一个响应。

- HMI 通讯参数配置

HMI 端口参数设置：



其中从设备号为 **HMI** 的站号，其它通讯参数根据具体情况设置。

9. 远程监控操作 VNC

VNC (Virtual Network Computing) 是虚拟网络计算机的缩写，是一种使用 RFB 协议的屏幕画面分享及远程操作软件。此软件借由网络，可传送键盘与鼠标的动作及实时的屏幕画面。

VNC 系统由客户端，服务端和一个协议组成，其服务端目的是分享其所运行机器的屏幕，服务端被动的允许客户端控制它，而 VNC 客户端 (或 Viewer) 观察控制服务端，与服务端交互。

VNC 与操作系统无关，因此可跨平台使用，Kinco 使用此功能来实现 PC、手机等设备对 HMI 的远程监控。

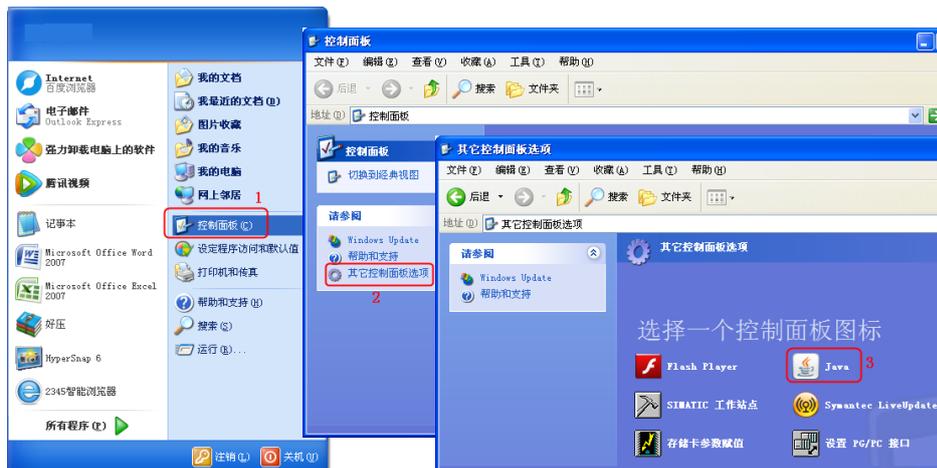
9.1 广泛的客户端

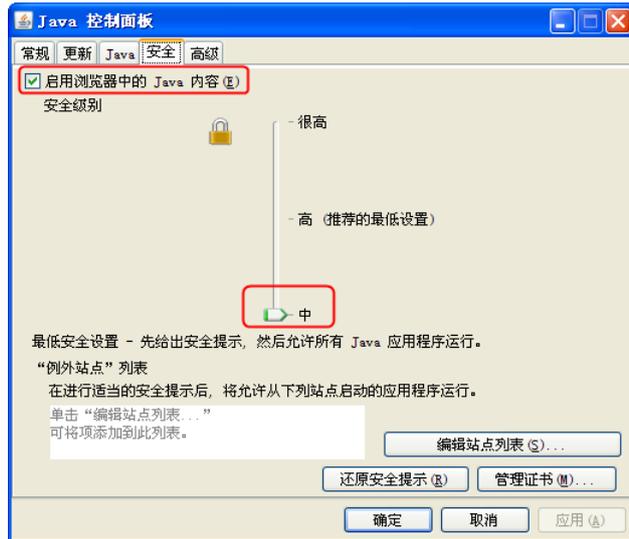
Kinco 并不提供 VNC 客户端，客户可以从互联网中下载：

- PC 机专用软件：VNC Viewer 等
- 浏览器客户端：IE、360 浏览器、Firefox 等
- 手机等移动终端上所用软件：一般智能系统上均有 VNC 软件：如 VNC/RDP 远程桌面精灵等。



1. 浏览器访问需要 JAVA 插件，一般计算机均有。如出现叉号，根据提示到 Java.com 安装。
2. 如果安装完 JAVA 插件后，浏览器浏览 VNC 功能，JAVA 插件不能正常运行，可在[开始]-[控制面板]-[其它控制面板选项-[JAVA]]的控制面板属性-[安全]将现有安全级别改为“中”。





3. 一台 PC 机上可以打开多个 VNC Viewer 客户端，同时访问多个 HMI。

9.2 通过局域网访问

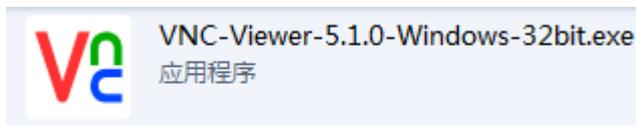
9.2.1 PC 机局域网监控操作 HMI

【范例 1】以 X10 和一台 PC 作为实例，此时 HMI 是服务端，PC 是客户端



1. 由于 VNC 功能须通过网络实现，所以只有带网口的 HMI 才支持此项功能

1) PC 客户端工具:

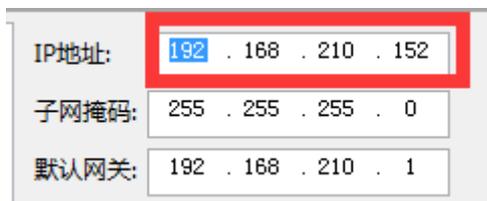


2) HMI 服务端设置动态开启 VNC 需设置特殊寄存器允许 PC 对其进行操作

寄存器	功能
SYS_START_VNC_SERVER	开启 HMI 的 VNC 功能，置 ON 开启该功能

3) HMI 设置

设置 HMI 的 IP 地址，如下图

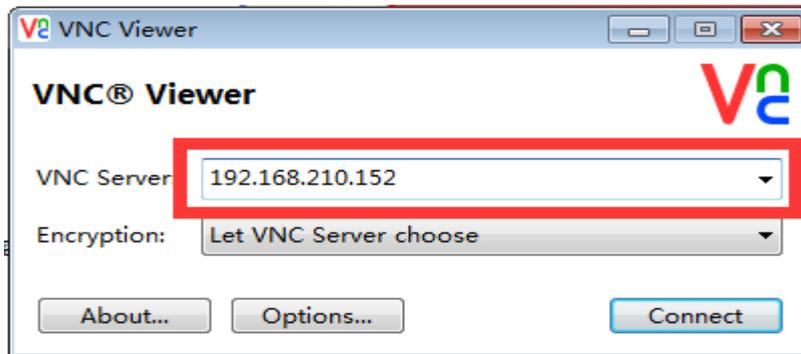




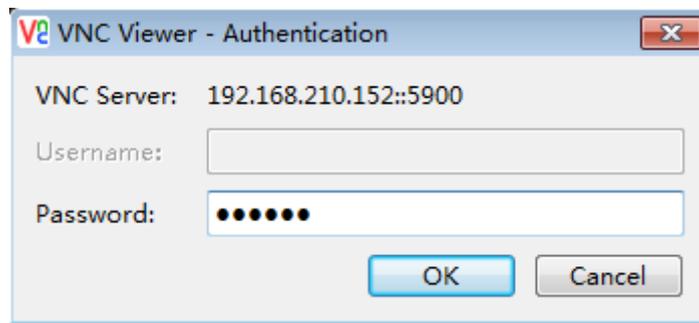
1.因是局域网访问，网关可以默认，不用设置

4) PC 端设置

输入需要监控的 HMI 的 IP 地址



输入密码（完全访问权限密码或者仅限查询密码，此处用例输入完全访问权限密码）



PC 端监控效果图如下：



此时，PC 端可以迅速反应 HMI 端的屏幕变化，同时也可以通过对 PC 端的操作来控制 HMI



1. 如果 PC 端输入仅限查询密码，此时 PC 只可以查看 HMI 的界面信息，而无法操作
2. 如果 PC 端输入完全访问权限密码，此时 PC 可以查看 HMI 界面，且可以操作该界面
3. 如果 HMI 不设置任何密码，默认 PC 无需密码，可以查看和操作 HMI

9.2.2 智能手机局域网监控操作 HMI

【范例 2】以 X10 和一部安卓操作系统手机作为实例，此时 HMI 是服务端，手机是客户端

1) 手机客户端工具（图中红色框圈出为 VNC 远程桌面精灵）



2) HMI 设置同【范例 1】

3) 手机设置（输入完全访问权限密码或者仅限查询密码，用例密码为空）





1. 地址为 HMI 对应的 IP 地址(
2. 端口号默认为 5900，请勿改动，否则可能导致连接不上
3. 名称和用户名可以不填写

手机监控效果图如下

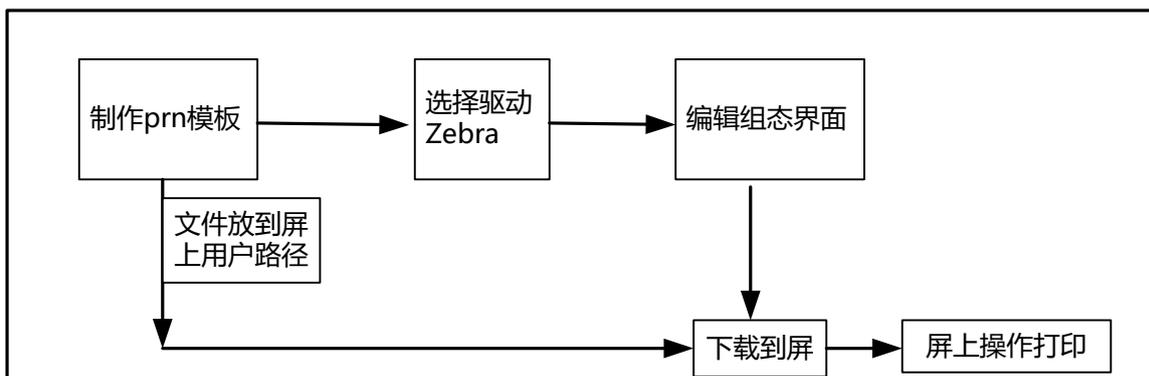


此时，手机端可以反应 HMI 端的屏幕变化，同时也可以通过对手机屏幕的操作来控制 HMI

10. 打印机

下面举例说明一下斑马打印机的使用。

10.1 斑马打印机打印制作流程



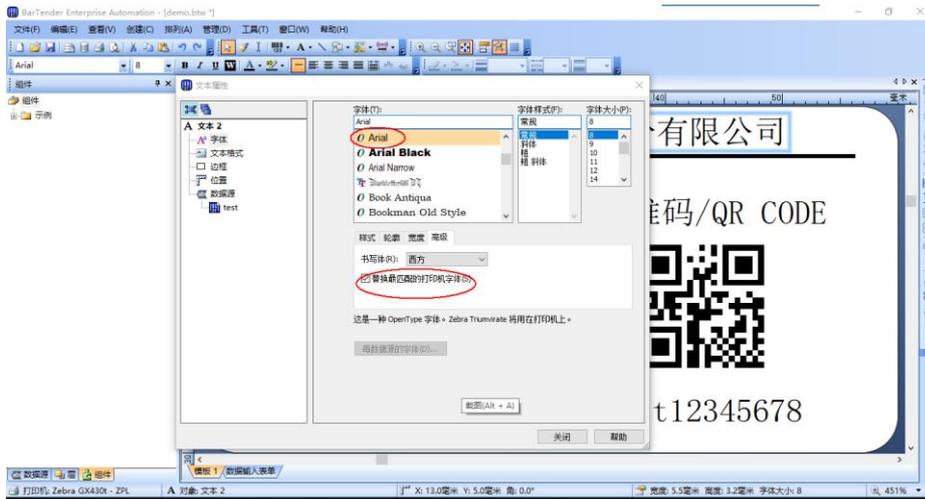
10.2 斑马打印机打印步骤

10.1.1 制作模板 PRN 文件

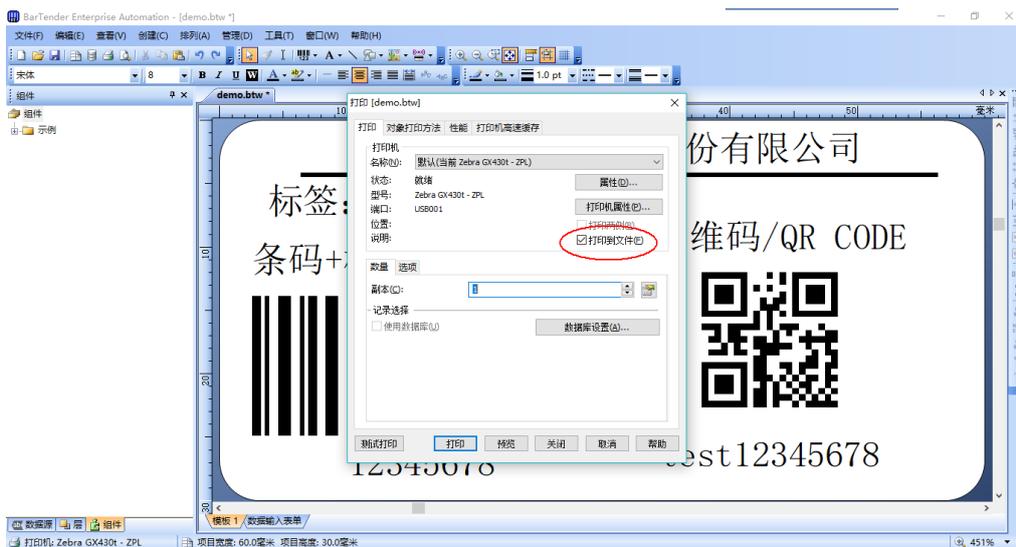
(1) 在 BarTender 软件中制作所需要的模板，编码格式按照图中所选



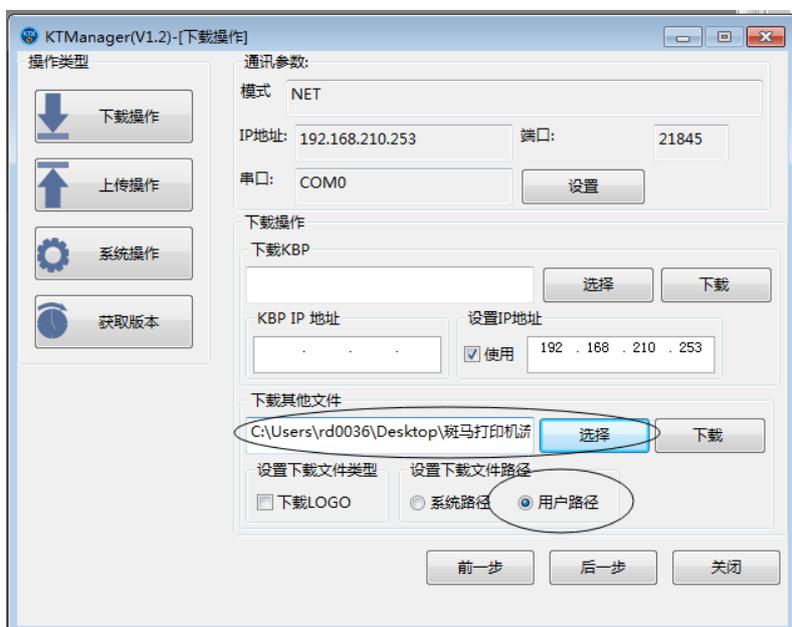
(2) 在 BarTender 软件中更改所有变量的属性，按下图设置



(3) 生成 PRN 文件，按下图选择



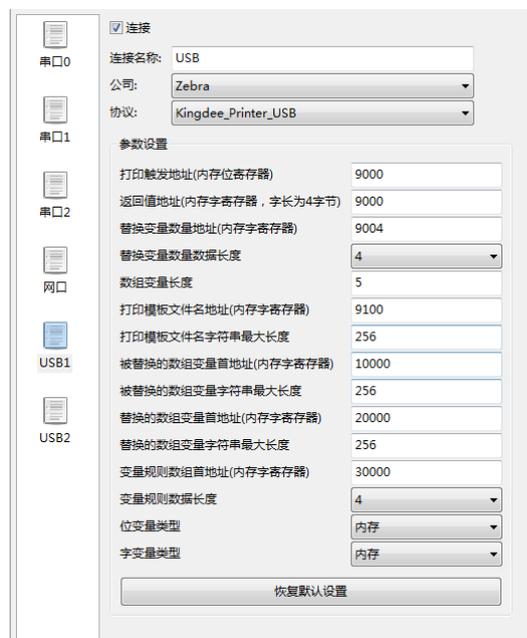
(4) 用 KTManger 软件下载 PRN 文件



10.1.2 组态编辑界面

例：把模板上的 **test** 改成 **kinc,12345678** 改成 **23456789**，**test12345678** 改成 **test23456789**（中文只支持 utf-8 打印！）

(1) 通讯设置如下图



(2) 组态界面的标签变量设置如下

需要更改的变量为 **3**，替换变量数量地址 **9004**，变量名 **g_num** 设置为 **3**

打印模板名地址 **9100**，变量名 **g_model** 赋值为 **demo.prn**

被替换的变量数组地址 **10000**，数组名 **g_arrName** 长度为 **3**

替换的变量数组地址 **20000**，数组名 **g_arrValue** 长度为 **3**

变量规则数组首地址 **30000**，变量名 **g_arrRule** 长度为 **3**

被替换的变量 **g_arrName[0]**为 **test**,替换的变量 **g_arrValue[0]**为 **kinc**，变量规则 **g_arrRule[0]**为 **0**，（**0** 代表标签），

数组下标为 **1** 和 **2** 按以上设置，具体参照工程

注：被替换的变量与替换的变量长度必须一致

(3) 触发变量设置如下

打印触发地址 **9000**,变量名 **g_bPrint**，动作切换状态，只要状态为 **on** 就会触发打印

(4) 下载工程

10.1.3 打印模板

(1) 屏与打印机通过 USB 口连接

(2) 点击屏上标签变量赋值，点击触发打印变量

(3) 打印机打印出标签，如下图



11. 寄存器

11.1 HMI 内部寄存器

11.1.1 位地址

描述	地址类型	地址范围	格式
本地位地址, HMI 自带的存储器, 掉电不保存	内存位寄存器	0~65535	DDDDD
本地字地址扩展出来的位地址	内存字寄存器	(0~1048575).(0~7)	DDDDDD.H
本地位地址, HMI 自带的存储器, 掉电保存	磁盘位寄存器	0~65535	DDDDD
本地字地址扩展出来的位地址, 掉电保存	磁盘字寄存器	(0~1048575).(0~7)	DDDDDD.H

11.1.2 字地址

描述	地址类型	地址范围	格式
本地字地址, HMI 自带的存储器, 掉电不保存	内存字寄存器	0~1048575	DDDDDD
本地字地址, HMI 自带的存储器, 掉电,保存	磁盘字寄存器	0~1048575	DDDDDD

11.2 HMI 系统寄存器

系统保留了一些本地的字、位地址作特殊用途, 用户必须按照相关说明来使用这一部分寄存器地址。

变量名称	寄存器类型	地址	长度	读写	变量描述	变量说明
SYS_DI0	内存位寄存器	0	1	只读	DI0	获取 DI0 的值
SYS_DI1	内存位寄存器	1	1	只读	DI1	获取 DI1 的值
SYS_DO0	内存位寄存器	2	1	读写	DI0	设置 DI0 的值
SYS_DO1	内存位寄存器	3	1	读写	DI1	设置 DI1 的值
SYS_DI2	内存位寄存器	4	1	只读	DI2	获取 DI2 的值
SYS_DO2	内存位寄存器	5	1	读写	DI2	设置 DI2 的值
SYS_LAN_STATU S	内存位寄存器	101	1	只读	有线网络连接状态	当前有线网络连接状态 OFF:未连接 ON:已连接
SYS_WLAN_STA TUS	内存位寄存器	102	1	只读	无线网络连接状态	当前无线网络连接状态 OFF:未连接 ON:已连接
SYS_SET_LAN_R OUTE	内存位寄存器	103	1	读写	使用有线网络路由	当值从 OFF 变为 ON 时, 设置当前默认路由为有线网络

SYS_SET_WLAN_ROUTE	内存位寄存器	104	1	读写	使用无线网络路由	当值从 OFF 变为 ON 时, 设置当前默认路由为无线网络
SYS_SYSTEM_SE T	内存位寄存器	201	1	读写	启动 SystemSet 程序	当状态从 OFF 变为 ON 时启动 SystemSet 程序
SYS_MACRO_STA TUS	内存位寄存器	300	1	读写	宏代码执行状态	宏代码启动时置 OFF , 执行完成后置 ON
SYS_WEBSERVICE STATUS	内存位寄存器	301	1	读写	WEBSERVICE 接口代码执行状态	WEBSERVICE 接口代码启动时置 OFF , 执行完成后置 ON
SYS_SAVE_INPUT DATA	内存位寄存器	400	1	读写	保存输入数据	只要有输入数据, 就保存在后台, 等下次弹出键盘, 直接默认取后台保存的第一条数据
SYS_USB1_PLUG IN	内存位寄存器	500	1	只读	USB1 是否插入 U 盘标志位	USB1 是否插入 U 盘标志位
SYS_USB2_PLUG IN	内存位寄存器	501	1	只读	USB2 是否插入 U 盘标志位	USB2 是否插入 U 盘标志位
SYS_SOP_BIT	内存位寄存器	1000	1000	读写	SOP 位寄存器	SOP 位寄存器
SYS_TOUCH_BUZZER	磁盘位寄存器	0	1	读写	触控蜂鸣开关	如果状态为 ON , 则当触控有效触控区域时触发蜂鸣
SYS_CURRENT_ROUTE	磁盘位寄存器	100	1	只读	当前默认路由	当前默认路由 0 : 有线网络 1 : 无线网络
ENABLE_WIFI	磁盘位寄存器	300	1	读写	使用无线	使用无线 0 : 无线关闭 1 : 无线开启
SYS_LAN_DHCP	磁盘位寄存器	400	1	读写	有线网络使用 DHCP	0 : 自定义 IP 1 : 有线网络使用 DHCP
SYS_DISABLE_NETWORK	磁盘位寄存器	500	1	读写	禁用有线	1 为禁用有线; 0 为启用有线
SYS_RECORD_MACRO_EXECUTE	磁盘位寄存器	600	1	读写	在本地数据库中记录宏代码 (包括 WebService 代码)	ON 表示记录, OFF 表示不记录, 注: 此功能需在工程里添加一本地数据库表 macroLog 4 个字段: objName (字符串类型, 长度 256); guid (字符串, 长度 36); startTime (日期时间); endTime (日期时间)
SYS_ETHERNET	磁盘位寄存器	700	1	只读	是否有有线网卡	1 表示有, 0 表示没有

SYS_WIFI	磁盘位寄存器	701	1	只读	是否有无线网卡	1 表示有, 0 表示没有
SYS_AUDIO_IN	磁盘位寄存器	702	1	只读	是否有音频输入口	1 表示有, 0 表示没有
SYS_AUDIO_OUT	磁盘位寄存器	703	1	只读	是否有音频输出口	1 表示有, 0 表示没有
SYS_CAN1	磁盘位寄存器	704	1	只读	是否有 CAN1 口	1 表示有, 0 表示没有
SYS_CAN2	磁盘位寄存器	705	1	只读	是否有 CAN2 口	1 表示有, 0 表示没有
SYS_KEYPAD	磁盘位寄存器	706	1	只读	是否有按键	1 表示有, 0 表示没有
SYS_VGA_OUT	磁盘位寄存器	707	1	只读	是否有 VGA 输出口	1 表示有, 0 表示没有
SYS_VIDEO_OUT1	磁盘位寄存器	708	1	只读	是否有视频输出口 1	1 表示有, 0 表示没有
SYS_VIDEO_OUT2	磁盘位寄存器	709	1	只读	是否有视频输出口 2	1 表示有, 0 表示没有
SYS_START_FTP_SERVER	磁盘位寄存器	800	1	读写	启动 FTP 服务	ON :启动; OFF :关闭
SYS_START_VNC_SERVER	磁盘位寄存器	801	1	读写	启动 VNC 服务	ON :启动; OFF :关闭
SYS_REQUIRE_TIME_SYNC	磁盘位寄存器	900	1	读写	是否需要同步时间	ON :启动; OFF :关闭
SYS_OPEN_EDGE_HEARTBEAT_PACKET	磁盘位寄存器	901	1	读写	是否启用与边缘服务器连接的心跳包	ON :启用; OFF :不启用 重启生效
SYS_OPEN_FRAME_CACHE	磁盘位寄存器	902	1	读写	是否开启页面缓存: 开启后在切换或弹出画面时, 会先显示该页面上一次的内容, 然后再刷新新值	ON :启用; OFF :不启用
SYS_YEAR	内存字寄存器	0	2	读写	年	系统时间: 年, 格式 20XX , 范围 1970-2037
SYS_MONTH	内存字寄存器	2	2	读写	月	系统时间: 月, 范围 1-12
SYS_DAY	内存字寄存器	4	2	读写	日	系统时间: 日, 范围 1-31
SYS_HOUR	内存字寄存器	6	2	读写	时	系统时间: 时, 范围 0-59
SYS_MINUTE	内存字寄存器	8	2	读写	分	系统时间: 分, 范围 0-59

SYS_SECOND	内存字寄存器	10	2	读写	秒	系统时间: 秒, 范围 0-59
SYS_WEEK_DAY	内存字寄存器	14	2	只读	星期 X	系统时间: 星期 X , 范围 0(sunday)-6(saturday)
SYS_YEAR_DAY	内存字寄存器	16	2	只读	一年中第 X 天	系统时间: 一年中第 X 天, 范围 1-365/366
WIFI_SIGNAL_S TRENTH	内存字寄存器	50	2	只读	无线信号强度	无线信号强度(dBm)
SYS_GUID	内存字寄存器	100	36	读写	全局唯一标识符	由有线 MAC 地址跟当前时间戳生成
SYS_USB1Type	内存字寄存器	200	2	读写	USB1 口插入的设备类型	USB1 口插入的设备类型
SYS_USB2Type	内存字寄存器	202	2	读写	USB2 口插入的设备类型	USB2 口插入的设备类型
SYS_USB1Name	内存字寄存器	210	32	读写	USB1 口插入的设备名称	USB1 口插入的设备名称
SYS_USB2Name	内存字寄存器	260	32	读写	USB2 口插入的设备名称	USB2 口插入的设备名称
SYS_SOP_SHOR T	内存字寄存器	1000	2000	读写	SOP16 位有符号寄存器	SOP16 位有符号寄存器
SYS_SOP_USHO RT	内存字寄存器	3000	2000	读写	SOP16 位无符号寄存器	SOP16 位无符号寄存器
SYS_SOP_INT	内存字寄存器	5000	4000	读写	SOP32 位有符号寄存器	SOP32 位有符号寄存器
SYS_SOP_UINT	内存字寄存器	9000	4000	读写	SOP32 位无符号寄存器	SOP32 位无符号寄存器
SYS_SOP_STRIN G	内存字寄存器	13000	64000	读写	SOP 字符串寄存器	SOP 字符串寄存器 (256 个字节), ANSI 编码
SYS_EDGE_SER VER_CONNECT_ STATUS	内存字寄存器	600	2	只读	边缘服务器连接状态	0 :未连接(没有启动心跳报文); 1 :已连接; 2 :未设置服务器地址; 3 :设备未在服务器注册; 4 :未收到服务器消息
SYS_LAN_MAC	磁盘字寄存器	0	12	只读	有线网络 MAC 地址	有线网络 MAC 地址
SYS_LAN_IP_1	磁盘字寄存器	12	2	读写	有线网络 IP 地址第 1 段	有线网络 IP 地址第 1 段
SYS_LAN_IP_2	磁盘字寄存器	14	2	读写	有线网络 IP 地址第 2 段	有线网络 IP 地址第 2 段
SYS_LAN_IP_3	磁盘字寄存器	16	2	读写	有线网络 IP 地址第 3 段	有线网络 IP 地址第 3 段
SYS_LAN_IP_4	磁盘字寄存器	18	2	读写	有线网络 IP 地址第 4 段	有线网络 IP 地址第 4 段

SYS_LAN_NETM ASK_1	磁盘字寄存器	20	2	读写	有线网络子网掩 码第 1 段	有线网络子网掩码第 1 段
SYS_LAN_NETM ASK_2	磁盘字寄存器	22	2	读写	有线网络子网掩 码第 2 段	有线网络子网掩码第 2 段
SYS_LAN_NETM ASK_3	磁盘字寄存器	24	2	读写	有线网络子网掩 码第 3 段	有线网络子网掩码第 3 段
SYS_LAN_NETM ASK_4	磁盘字寄存器	26	2	读写	有线网络子网掩 码第 4 段	有线网络子网掩码第 4 段
SYS_LAN_GATE WAY_1	磁盘字寄存器	28	2	读写	有线网络网关第 1 段	有线网络网关第 1 段
SYS_LAN_GATE WAY_2	磁盘字寄存器	30	2	读写	有线网络网关第 2 段	有线网络网关第 2 段
SYS_LAN_GATE WAY_3	磁盘字寄存器	32	2	读写	有线网络网关第 3 段	有线网络网关第 3 段
SYS_LAN_GATE WAY_4	磁盘字寄存器	34	2	读写	有线网络网关第 4 段	有线网络网关第 4 段
SYS_LAN_DNS1_ 1	磁盘字寄存器	36	2	读写	有线网络 DNS1 第 1 段	有线网络 DNS1 第 1 段
SYS_LAN_DNS1_ 2	磁盘字寄存器	38	2	读写	有线网络 DNS1 第 2 段	有线网络 DNS1 第 2 段
SYS_LAN_DNS1_ 3	磁盘字寄存器	40	2	读写	有线网络 DNS1 第 3 段	有线网络 DNS1 第 3 段
SYS_LAN_DNS1_ 4	磁盘字寄存器	42	2	读写	有线网络 DNS1 第 4 段	有线网络 DNS1 第 4 段
SYS_LAN_DNS2_ 1	磁盘字寄存器	44	2	读写	有线网络 DNS2 第 1 段	有线网络 DNS2 第 1 段
SYS_LAN_DNS2_ 2	磁盘字寄存器	46	2	读写	有线网络 DNS2 第 2 段	有线网络 DNS2 第 2 段
SYS_LAN_DNS2_ 3	磁盘字寄存器	48	2	读写	有线网络 DNS2 第 3 段	有线网络 DNS2 第 3 段
SYS_LAN_DNS2_ 4	磁盘字寄存器	50	2	读写	有线网络 DNS2 第 4 段	有线网络 DNS2 第 4 段
SYS_WLAN_MA C	磁盘字寄存器	100	12	只读	无线网络 MAC 地址	无线网络 MAC 地址
SYS_WLAN_IP_1	磁盘字寄存器	112	2	读写	无线网络 IP 地址 第 1 段	无线网络 IP 地址第 1 段
SYS_WLAN_IP_2	磁盘字寄存器	114	2	读写	无线网络 IP 地址 第 2 段	无线网络 IP 地址第 2 段
SYS_WLAN_IP_3	磁盘字寄存器	116	2	读写	无线网络 IP 地址 第 3 段	无线网络 IP 地址第 3 段
SYS_WLAN_IP_4	磁盘字寄存器	118	2	读写	无线网络 IP 地址 第 4 段	无线网络 IP 地址第 4 段
SYS_WLAN_NET MASK_1	磁盘字寄存器	120	2	读写	无线网络子网掩 码第 1 段	无线网络子网掩码第 1 段

SYS_WLAN_NET_MASK_2	磁盘字寄存器	122	2	读写	无线网络子网掩码第 2 段	无线网络子网掩码第 2 段
SYS_WLAN_NET_MASK_3	磁盘字寄存器	124	2	读写	无线网络子网掩码第 3 段	无线网络子网掩码第 3 段
SYS_WLAN_NET_MASK_4	磁盘字寄存器	126	2	读写	无线网络子网掩码第 4 段	无线网络子网掩码第 4 段
SYS_WLAN_GATEWAY_1	磁盘字寄存器	128	2	读写	无线网络网关第 1 段	无线网络网关第 1 段
SYS_WLAN_GATEWAY_2	磁盘字寄存器	130	2	读写	无线网络网关第 2 段	无线网络网关第 2 段
SYS_WLAN_GATEWAY_3	磁盘字寄存器	132	2	读写	无线网络网关第 3 段	无线网络网关第 3 段
SYS_WLAN_GATEWAY_4	磁盘字寄存器	134	2	读写	无线网络网关第 4 段	无线网络网关第 4 段
SYS_WLAN_DNS_1_1	磁盘字寄存器	136	2	读写	无线网络 DNS1 第 1 段	无线网络 DNS1 第 1 段
SYS_WLAN_DNS_1_2	磁盘字寄存器	138	2	读写	无线网络 DNS1 第 2 段	无线网络 DNS1 第 2 段
SYS_WLAN_DNS_1_3	磁盘字寄存器	140	2	读写	无线网络 DNS1 第 3 段	无线网络 DNS1 第 3 段
SYS_WLAN_DNS_1_4	磁盘字寄存器	142	2	读写	无线网络 DNS1 第 4 段	无线网络 DNS1 第 4 段
SYS_WLAN_DNS_2_1	磁盘字寄存器	144	2	读写	无线网络 DNS2 第 1 段	无线网络 DNS2 第 1 段
SYS_WLAN_DNS_2_2	磁盘字寄存器	146	2	读写	无线网络 DNS2 第 2 段	无线网络 DNS2 第 2 段
SYS_WLAN_DNS_2_3	磁盘字寄存器	148	2	读写	无线网络 DNS2 第 3 段	无线网络 DNS2 第 3 段
SYS_WLAN_DNS_2_4	磁盘字寄存器	150	2	读写	无线网络 DNS2 第 4 段	无线网络 DNS2 第 4 段
SYS_WLAN_BSSID_D_1	磁盘字寄存器	152	2	只读	无线网络 BSSID 第 1 段	无线网络 BSSID 第 1 段
SYS_WLAN_BSSID_D_2	磁盘字寄存器	154	2	只读	无线网络 BSSID 第 2 段	无线网络 BSSID 第 2 段
SYS_WLAN_BSSID_D_3	磁盘字寄存器	156	2	只读	无线网络 BSSID 第 3 段	无线网络 BSSID 第 3 段
SYS_WLAN_BSSID_D_4	磁盘字寄存器	158	2	只读	无线网络 BSSID 第 4 段	无线网络 BSSID 第 4 段
SYS_WLAN_BSSID_D_5	磁盘字寄存器	160	2	只读	无线网络 BSSID 第 5 段	无线网络 BSSID 第 5 段
SYS_WLAN_BSSID_D_6	磁盘字寄存器	162	2	只读	无线网络 BSSID 第 6 段	无线网络 BSSID 第 6 段
SYS_MACRO_TIMEOUT	磁盘字寄存器	300	4	读写	宏代码执行超时时间	宏代码执行超时时间(单位:毫秒),为 0 时使用默认

						超时时间
SYS_LANGUAGE	磁盘字寄存器	304	2	读写	系统语种	如果为 0 或者超出最大语种数, 则使用默认语种
SYS_LCD_X_RES	磁盘字寄存器	400	2	只读	屏幕 X 尺寸	屏幕 X 尺寸
SYS_LCD_Y_RES	磁盘字寄存器	402	2	只读	屏幕 Y 尺寸	屏幕 Y 尺寸
SYS_COLOR_DEPTH	磁盘字寄存器	406	2	只读	屏幕色深	4BPP, 8BPP,16BPP,24BPP
SYS_FLASH_SIZE	磁盘字寄存器	412	4	只读	FLASH 尺寸	M 字节数
SYS_SDRAM_SIZE	磁盘字寄存器	416	4	只读	SDRAM 尺寸	M 字节数
SYS_EXT_MEMORY	磁盘字寄存器	422	4	只读	扩展存储器大小	M 字节数, 无扩展时为 0
SYS_PORTS_AMOUNT	磁盘字寄存器	428	2	只读	串口数	数量, 0 表示没有
SYS_SD_CARD	磁盘字寄存器	430	2	只读	SD 卡	数量, 0 表示没有
SYS_USB_SLAVE	磁盘字寄存器	432	2	只读	USB 从口	数量, 0 表示没有
SYS_USB_HOST	磁盘字寄存器	434	2	只读	USB 主口	数量, 0 表示没有
SYS_MIN_BRIGHTNESS	磁盘字寄存器	436	2	只读	亮度调节下限	数量, 0 表示没有
SYS_MAX_BRIGHTNESS	磁盘字寄存器	438	2	只读	亮度调节上限	数量, 0 表示没有
SYS_CURRENT_BRIGHTNESS	磁盘字寄存器	440	2	只读	亮度调节当前值	数量, 0 表示没有
SYS_DI_NUM	磁盘字寄存器	442	2	只读	DI 接口数量	数量, 0 表示没有
SYS_DO_NUM	磁盘字寄存器	446	2	只读	DO 接口数量	数量, 0 表示没有
SYS_HDMI	磁盘字寄存器	450	2	只读	HDMI 接口	数量, 0 表示没有
SYS_PRODUCT_ID	磁盘字寄存器	482	2	只读	产品型号	一个非命名的唯一产品 ID 号 (X10:169,X15:171, 亮 灯 系 统:172,X10F:173,KBoard :511)
SYS_HW_VERSION	磁盘字寄存器	484	2	只读	产品硬件版本号	产品硬件版本号

INFRARED_FRA ME	磁盘字寄存器	486	2	只读	红外框类型	0 代表不使用红外框, 1 以上数字代表具体类型
SYS_COM_NAM E_DEF	磁盘字寄存器	488	2	只读	串口名称定义	0:X10 平台 1: KBoard
SYS_DYNAMIC_ STATION	磁盘字寄存器	600	512	读写	从站动态站号	范围: 0~255
SYS_TIME_SYN C_INTERVAL	磁盘字寄存器	1200	2	读写	时间同步间隔	单位: 分钟 范围: 1-1440
SYS_TIME_SYN C_SERVER	磁盘字寄存器	1202	256	读写	时间同步服务器 地址	时间同步服务器地址, 最 大长度 256 个字节
SYS_EDGE_SER VER	磁盘字寄存器	1500	16	读写	边缘服务器地址, 最大长度 16 个字 节	边缘服务器地址, 最大长 度 16 个字节。重启生效

12. 编码类型



KHDesigner 可支持处理的数据编码类型有：**BIN** 和 **BCD**。用户需根据所接设备数据处理的实际编码方式来选择正确的编码类型。

12.1 BIN

BIN (Binary) 即二进制编码，大多数数字系统都是以 **BIN** 码来处理数据。

数据格式	16 位数据范围	32 位数据范围
有符号 BIN	-32768~32767	-2147483648~2147483647
无符号 BIN	0~65535	0~4294967295

【举例】

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
----------	----------	----------	----------	----------	----------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

将二进制 **1001 0100** 转换为十进制数

$$1001\ 0100 = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 148$$

12.2 BCD

BCD(Binary Coded Decimal)即二 - 十进制代码，就是用四位二进制数表示一位十进制数 **0~9** 的编码方式。例如 **9** (十进制) = **1001** (二进制)。

由于处理简易，因此，可用于 **BCD** 输出型的数字开关或 **7** 段 **BCD** 数码管的显示。

数据格式	16 位数据范围	32 位数据范围
BCD	0~9999	0~99999999

【举例】

十进制数+123，其二进制数为 **0000 0000 0111 1011** ($64+32+16+8+2+1=123$)；而 **BCD** 码#123，可简单的表示为 **0000 0001 0010 0011**

十进制数-413，其二进制数为 **1111 1110 0110 0011** (负数需用补码，取反加 1)，而用 **BCD** 码#F413，可简单的表示为 **1111 0100 0001 0011**

13. 与外围设备的连接

为满足用户实现智能化、信息化、人性化等多功能界面的要求，智能终端产品系列 HMI 配备了多种接口，下面依次介绍这些接口支持连接的设备。

13.1 串行接口

13.1.1 串口介绍



注意

请勿带电插拔通讯电缆，避免导致通讯接口损坏

- 串行接口 COM0

COM0 为 9 针 D 型公座管脚，此接口支持 RS-232/RS-485/RS-422 通讯功能，其管脚定义如下：

管脚	信号	功能		
		RS-232C	RS-485	RS-422A
1	RX-(B)	--	RS485B	接收数据
2	RXD	接收数据	--	--
3	TXD	发送数据	--	--
4	TX-	--	--	发送数据
5	SG	信号地		
6	RX+(A)	--	RS485A	接收数据
7	NC	--	--	--
8	NC	--	--	--
9	TX+	--	--	发送数据

- 串行接口 COM1

COM1 为 9 针 D 型母座管脚，此接口支持 RS-232/RS-485/RS-422 通讯功能，其管脚定义如下：

管脚	信号	功能		
		RS-232C	RS-485	RS-422A
1	RX-(B)	--	RS485B	接收数据
2	RXD	接收数据	--	--
3	TXD	发送数据	--	--
4	TX-	--	--	发送数据
5	SG	信号地		
6	RX+(A)	--	RS485A	接收数据
7	CTS	清除发送	--	--
8	RTS	请求发送	--	--
9	TX+	--	--	发送数据

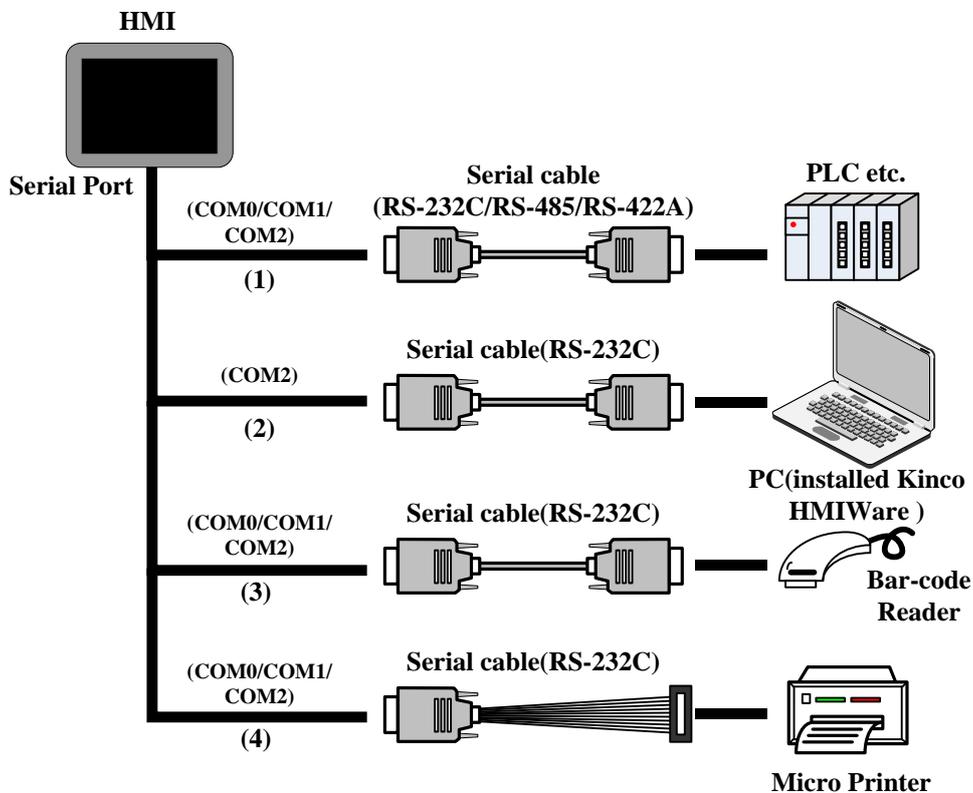
- 串行接口 COM2

COM2 与 COM0 共用物理接口，为 9 针 D 型公座管脚，此接口仅支持 RS-232 通讯功能，其管脚定义如下：

管脚	信号	功能
		RS-232C
1	NC	--
2	NC	--
3	NC	--
4	NC	--
5	SG	信号地
6	NC	--
7	RXD	接收数据
8	TXD	发送数据
9	NC	--

COM2 接口能连接 RS-232 功能的控制器，也可用于产品的程序上传/下载和调试。

13.1.2 通过串行接口连接



(1) 通过串行接口与 PLC 等控制设备进行通信连接

HMI 通过 RS-232/RS-485/RS-422 串行电缆和 PLC 等控制设备的串行接口连接进行通信。

- 连接条件

和 HMI 通信的 PLC 等控制设备必须是 KHDesigner 支持的型号或具备 KHDesigner 支持的串口通讯协议。

- 电缆制作

根据不同的控制设备型号和通信方式，用户可自行制作电缆进行连接。

(2) 通过串行接口连接扫描仪等设备

HMI 可通过 **RS-232** 串行电缆连接扫描仪设备进行条形码等数据的接收。

- 连接条件

和 HMI 通信的扫描仪设备必须是 **KHDesigner** 支持的型号或具备 **KHDesigner** 支持的通讯协议。

- 电缆制作

直接使用扫描仪设备厂家提供的 **RS-232** 通讯电缆。

(3) 通过串行接口连接微型打印机

HMI 可通过 **RS-232** 串行电缆连接微型打印机进行屏幕打印或报表打印输出。

- 连接条件

和 HMI 通信的微型打印机必须是 **KHDesigner** 支持的型号或具备 **KHDesigner** 支持的打印机通讯协议。

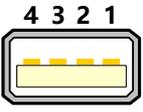
- 电缆制作

用户可自行制作电缆进行连接。

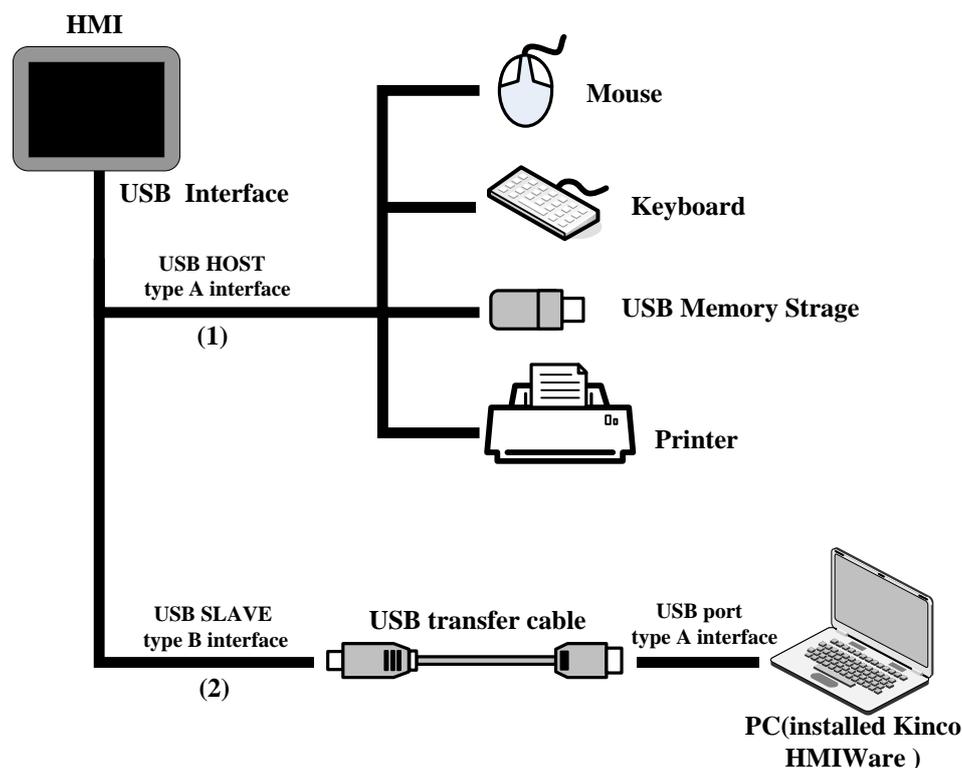
13.2 USB 接口

- USB HOST

USB HOST 接口为 **USB A** 型接口，可连接如 **USB** 存储设备、键盘、鼠标等外围设备。其管脚定义如下：

	管脚	信号	功能	
	1	VCC	+5V 电源	
	2	D-	Data-	
	3	D+	Data+	
	4	GND	-5V 接地	

13.2.1 通过 usb 接口连接



(1) 通过 **USB HOST** 接口连接 **USB** 接口的鼠标、键盘等设备

- 连接 **USB** 鼠标

HMI 和 **USB** 鼠标进行连接，**USB** 鼠标可对 **HMI** 屏幕进行输入操作。

- 连接 **USB** 键盘

HMI 和 **USB** 键盘进行连接，**USB** 键盘可通过映射键功能对 **HMI** 屏幕进行输入操作。

- 连接 **USB** 大容量存储设备（**U 盘只支持 FAT32 格式**）

HMI 和 **USB** 大容量存储设备连接，通过导入/导出功能在 **HMI** 与 **USB** 大容量存储设备间进行用户工程等文件的快速上传/下载；

- 连接 **USB** 打印机

HMI 可通过 **USB** 传输电缆连接 **USB** 接口的打印机进行屏幕打印。

和 **HMI** 通信的 **USB** 打印机必须是 **KHDesigner** 支持的型号或具备 **KHDesigner** 支持的打印机通讯协议。

(2) 通过 **USB SLAVE** 接口进行数据上传/下载

HMI 可通过 **USB** 传输电缆连接 **HMI** 和 **PC** 机侧的 **USB** 接口进行工程等数据的上传/下载。

- 连接条件

在 **PC** 机上安装 **KHDesigner** 组态编辑软件并运行，通过软件【工具】菜单进行下载方式选择。

- 电缆制作

用户可自行制作电缆进行连接。



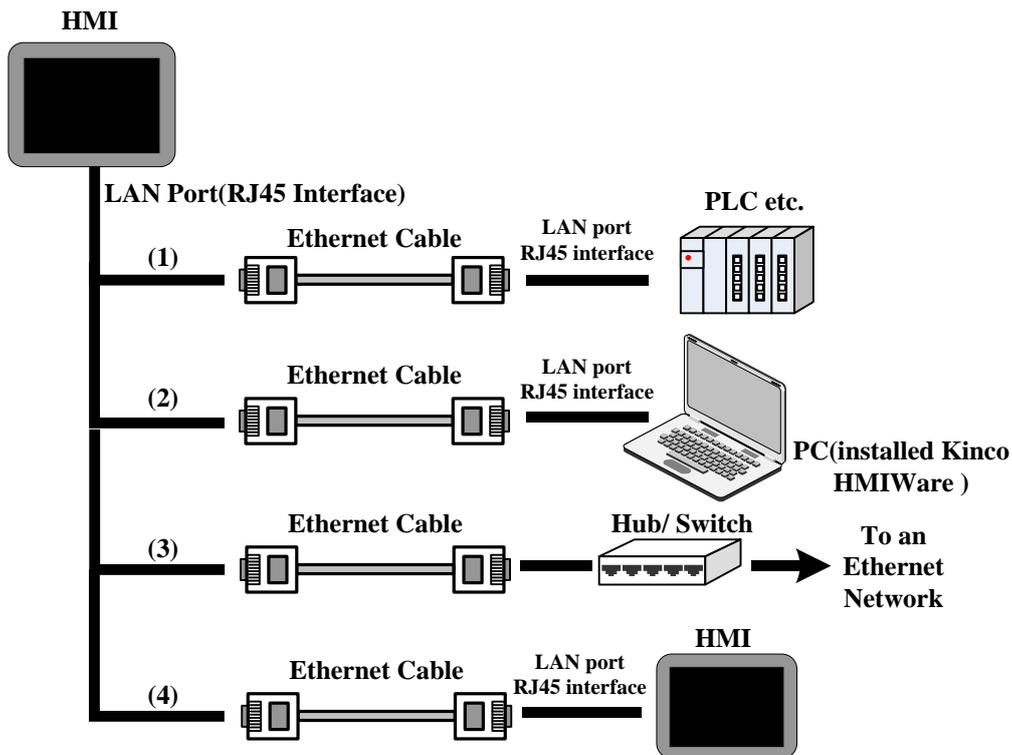
关于电缆制作详情请参阅【通讯连接说明】手册

13.3 LAN 接口

LAN 接口为 10M/100M 传输速率自适应的 RJ-45 型接口，可用于产品程序的上传/下载和调试；多台具备 LAN 接口的 HMI 联机以及和支持以太网通讯的控制器通信。其管脚定义如下：

管脚	信号	功能
1	TX+	数据发送+
2	TX-	数据发送-
3	RX+	数据接收+
4	NC	--
5	NC	--
6	RX-	数据接收-
7	NC	--
8	NC	--

13.3.1 通过 LAN 接口连接



(1) 通过 LAN 接口和 PLC 等控制设备进行通信连接

HMI 可以通过两端为 RJ45 接头的交叉双绞线(Crossover Cable)和 PLC 等控制设备的 LAN 接口连接进行通信。

- 连接条件

和 HMI 进行以太网通讯的 PLC 等控制设备必须是 KHDesigner 支持的型号或具备 KHDesigner 支持的以太网通讯协议。

- 电缆制作

用户可自行制作电缆进行连接。

(2) 通过 LAN 接口进行数据的上传/下载/监控

HMI 可通过两端为 RJ45 接头的交叉双绞线(Crossover Cable)和 PC 机侧的 LAN 接口连接进行用户工程等数据的上传/下载；同时还可在 PC 机上监控 HMI 或与 HMI 通讯的 PLC 等控制设备的数据。

- 连接条件

在 PC 机上安装 KHDesigner 组态编辑软件并运行，通过软件【工具】菜单进行下载方式选择。

- 电缆制作

用户可自行制作电缆进行连接。

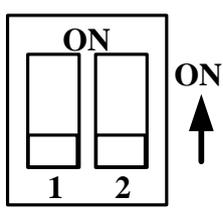
(3) 通过 Hub/Switch 将 HMI 接入一个局域网

HMI 可通过两端为 RJ45 接头的直通双绞线(Straight Through Cable)或交叉双绞线(Crossover Cable)，经由集线器(Hub) 或交换机(Switch) 接入一个局域网

	更多详情请参阅【通讯连接说明】手册
---	-------------------

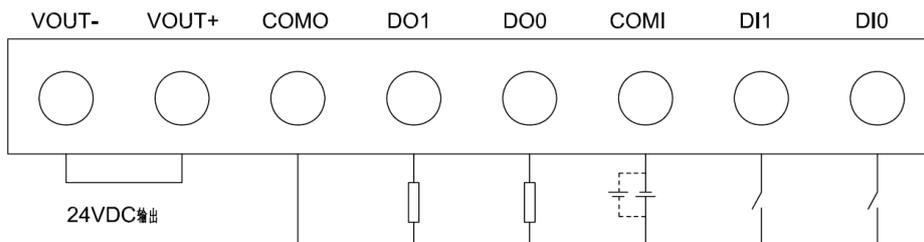
13.4 DIP 开关

设置 DIP Switch 处于不同的位置可使 HMI 系统处于不同的工作模式，其设置及相应工作模式如下：

 <p>出厂状态</p>	SW1	SW2	工作模式
	OFF	OFF	正常工作模式
	ON	OFF	固件更新模式
	OFF	ON	保留模式（目前不使用）
	ON	ON	系统设置模式

13.5 IO 口

1.IO 接口两入两出共 4 路，同时支持对外 24V 供电。X15 多支持一个 I/O 口！接口定义如下图：



2.IO 接口可使用系统寄存器 SYS_DI0, SYS_DI1, SYS_DO0, SYS_DO1

3. 根据需求可以实现报警、亮灯等功能

14. 系统设置

在系统设置模式下用户可对系统时间、IP 地址等项目进行设置。

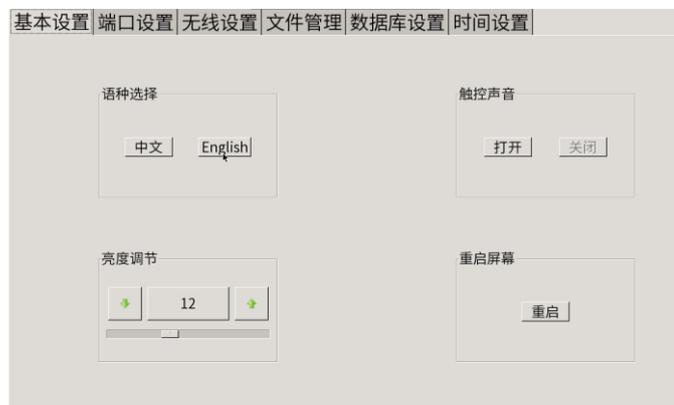
14.1 系统设置模式的显示方法

系统设置模式通过以下步骤进入：

- ① 触摸屏上电前用手指按住液晶屏不放，给触摸屏上电，待触摸屏自动进入模式选择界面之后方可松手；
- ② 在菜单选择界面，选择系统设置模式。

14.2 系统设置

以 X10 的系统设置画面为例：



14.2.1 基本设置

- 语种选择：英文中文界面显示
- 亮度调节：调节屏幕的亮度（Brightness Up/Down），使屏幕呈现最佳视觉效果。
- 控制声音：开启/关闭蜂鸣器
- 重启屏幕：重新启动 HMI

14.2.2 端口设置

- 串口信息：查看各个网口的参数
- 串口调试：打开/关闭串口调试
- DO/DI 口测试：IO 口进行测试

14.2.3 有线设置

- 网络信息：可对带以太网口的 HMI 进行网络参数设置，包括 IP 地址 (IP Address)、子网掩码 (Subnet Mask)、网关 (Gateway) 等。
- DNS 服务器信息：设置 DNS 服务器地址

14.2.4 无线设置

- 无线网络信息：对带 WiFi 的 HMI 进行网络参数设置，包括子网掩码 (Subnet Mask)、网关 (Gateway) 等。
- 添加隐藏网络：添加没有开启 SSId 广播的网络
- 无线网络：对 WiFi 进行开启、关闭、扫描、连接动作
- 用户自定义无线 IP 地址：可以设置 HMI 的无线 IP 地址
- 重置 WiFi：对 wifi 信息进行清空



1. 当 WiFi 信息更改时，例如密码更改等需要重置 WiFi
2. 当某些无线 MAC 地址没有连接权限时可进行重置 WiFi、换其他 WiFi 连接

14.2.5 文件管理

- 本地目录：默认目录为/home/user,不可更改；系统目录为内部系统文件所在目录，需输入密码。
- 选择 USB:可选择 USB1 或 2 (屏的最左端是 USB1，靠近网口的为 USB2)
- 文件传输：可对文件进行导入和导出
- 文件操作：可以对默认目录下的文件进行删除
- 下载：下载 KBP 工程，可以清除下载工程中的本地数据库和磁盘寄存器数据
- U 盘目录：显示 U 盘内的内容

14.2.6 数据库设置

- 数据库备份恢复设置：可以对本地数据库的备份(备份成按操作时间名称的 **sql** 文件)、恢复 (点击 **sql** 文件可以恢复)。也可以对/home/user 下的 **sql** 文件进行刷新。
- 数据库修复设置：本地数据库报表有损坏时，可通过修复和扩展修复进行修复。也可以对本地数据库进行更新。

14.2.7 时间设置

- 时间信息：查看年、月、日、时、分、秒是否为当前时间，若不是，可手动对其进行设置。

时间同步：默认时间不同步，当开启时可以设置同步间隔时间（单位是分钟和小时）和设置时钟服务器来同步时间。

也可以通过系统寄存器 **SYS_REQUIRE_TIME_SYNC**，**SYS_TIME_SYNC_INTERVAL**，**SYS_TIME_SYNC_SERVER** 来控制。时间同步功能需要先搭建 **NTP** 服务器，具体搭建参考网址：

<https://jingyan.baidu.com/article/e4511cf358ebf52b845eaff9.html>

15. 触控校准模式

在更新固件版本后出现屏幕触控不精准时，用户可在此模式下对HMI屏幕进行触控校正。

15.1 触控校准模式的显示方法

触控校准模式通过以下步骤进入：

- ① 触摸屏上电前用手指按住液晶屏不放，给触摸屏上电，待触摸屏自动进入模式选择界面之后方可松手；
- ② 在菜单选择界面，选择触控校准模式。

15.2 触控校准设置

跟随屏幕指示的“+”进行触控，听到“嘀”声响，一直触控到“+”消失，并显示“校正成功，触摸任意位置，继续启动”后即表示屏幕触控校准成功，然后点击任意位置，HMI自动启动。

16. 固件更新模式

在固件更新模式下，用户可通过 **KTManager** 更新 **HMI** 的内核和文件系统版本（请在厂家指导下使用此功能）。

16.1 固件更新模式的显示方法

固件更新模式通过以下步骤进入：

带拨码开关屏（电子看板）：

- ① 将 **HMI** 背面的 **DIP** 开关 **SW1** 置于 **ON** 位置，**SW2** 置于 **OFF** 位置；
- ② 按下 **RESET** 按钮，**HMI** 系统重新启动后进入固件更新模式画面。

不带拨码开关屏（智能终端）：

- ① 触摸屏上电前用手指按住液晶屏不放，给触摸屏上电，待触摸屏自动进入模式选择界面之后方可松手；
- ② 在菜单选择界面，选择固件更新模式。

16.2 固件更新设置

打开 **KTManager**，进入【系统操作】更新 **HMI** 的内核和文件系统。

安全注意事项

在使用本产品前,请仔细阅读本手册以及本手册中所介绍的关联手册,同时在充分注意安全的前提下进行正确的操作。
为了安全使用产品,本手册以下述标识及图形符号来表示注意事项,并在此所示的注意事项中标识与安全相关的重要内容,请务必遵守。

	危险 表示如果不采取相应的小心措施,将会导致严重的人身伤害甚至死亡
	警告 表示如果不采取相应的小心措施,可能导致严重的人身伤害甚至死亡
	小心 表示如果不采取相应的小心措施,可能导致轻微的人身伤害
	注意 表示如果不注意相应提示,可能会出现不希望的结果或状态
	禁止 表示禁止执行的相关指令、流程或拆卸动作,以确保产品的正确使用

阅读后请妥善保管本手册,以便随时取阅。

使用时的注意事项:

- 智能终端及看板系列产品应特别注意回避在以下场所使用:
 - 工作温度或湿度超出规格指定范围的场所
 - 高湿度,易结成冷凝水的场所
 - 温度变化剧烈的场所
 - 受紫外线直接照射的场所
 - 具有腐蚀性或可燃性气体的场所
 - 具有高粉尘环境的场所
 - 易受化学物质污染的场所
 - 易受油性物质污染的场所

- 具有强烈冲击或振动的场所
- 阳光直射、暴露在风雨环境下的场所
- **HMI** 主机与 **PLC** 等控制设备的通讯电缆、电源线应尽量远离变频器、电机等易产生干扰的设备。同时勿将控制线与动力线、高压线在一个线槽或附近走线，否则易产生干扰造成设备误动作
- 在接通 **HMI** 主机电源前，请确保系统的安全性
- 使用 **HMI** 控制 **PLC** 等控制设备前，必须使用组态编辑软件向 **HMI** 编写正确的组态控制画面程序
- 使用 **USB** 下载电缆对 **HMI** 进行数据传输前，务必在安装有组态编辑软件的 **PC** 上安装 **USB** 驱动程序
- 务必使用手指或专用触控笔来操作 **HMI**
- 若快速、连续触控 **HMI** 可能会导致无法读取输入的内容。请在确认一个输入成功后，再进行下一个输入操作
- 背光灯熄灭或无显示时，请避免无意触控 **HMI**。务必在确认系统安全后，方可触控 **HMI**
- **DIP** 开关设定一经变更，必须按复位开关或断电重启 **HMI** 方可生效
- 安装电池时，务必使用规格规定的电池并正确安装。

附录 1 通讯 code 信息一览表

序号	错误信息	描述
1	Communication error[COM x/ETHERNET x,ADD:xx]: code 1	通讯错误[串口 x/IP 地址 x,地址:xx] (以下的类似)) 其他错误: 文件打开失败, 系统操作失败, 与通讯无关的错误 (如果是组网报错/ETHERNET x 代表组网屏的地址, ADD:xx 代表对于的地址, StationNum x 代表站号如果是组网屏的本地地址报 65535, 如果组网屏连接的 PLC 报的 PLC 站号)
2	code 2	通讯超时
3	code 3	通讯参数 (地址等) 无效
4	code 4	组包参数重设
5	code 14	组网的时间戳不对
6	code 161	重复请求失败
7	code 162	请求未找到
8	code 163	寄存器需要的内存长度太长, 超出内存设定;
9	code 164	创建请求失败
10	code 165	未知的数据类型
11	code 167	数据类型为只读
12	code 168	解析失败
13	code 176	线程初始化失败
14	code 177	线程创建失败
15	code 178	分配内存失败
16	code 179	空指针类型
17	code 193	读失败
18	code 194	写失败
19	code 195	读写超时
20	code 196	打开文件失败
21	code 197	连接服务器失败
22	code 198	未连接
23	code 199	断开连接
24	code 225	请求异常
25	code 227	参数错误
26	code 228	非法参数
27	code 229	本地数组访问越界
28	code 230	文件打开失败
29	code 231	文件指针定位失败
30	code 232	文件读失败
31	code 233	文件写失败
32	code 234	本地请求类型错误
	code 255	失败

附录 2 右键快捷功能使用一览表

序号	功能	用法
1	快速查找变量使用地方	变量表中选中变量，单击右键，选择展开变量的使用环境
2	快速查找宏使用的地方	宏列表选中宏，单击右键，选择展开宏的使用环境
3	组合元件	多个元件选中，点击右键，组合
4	多重复制	选择单个元件，点击右键，多重复制
5	宏里面变量快捷选择	点击右键，选择变量

附录 3 软件使用注意一览表

序号	软件功能使用注意
1	连续弹出多个键盘，只弹出最后一个键盘
2	字变量弹出键盘只能输入数字
3	图片，sop 工程模拟时，必须把文件放到软件安装的 KHDesigner 文件夹下
4	跨工程复制时，要注意复制的工程上是否有与动作相关的变量、宏等，否则复制无效
5	数组不能越界
6	strepy 没有结束符
7	字符串取值长度超出本身长度
8	宏出现 code8 ：宏里面有不符合使用的用法（例如除数为 0 等）
9	数据库出现 code3 ：数据库数据太多，超出临时表的范围（数据库全部显示改成分页显示）
10	双击 KPJ 文件可以直接打开（ xp 与 win7 系统不需要管理者权限，但是 win8 与 win10 需要管理者权限）