实 验 名 称： PSE软启动器-OPC远控-XT断路器实验

日 期： 2022.1.3

**实验五** PSE软启动器-OPC远控-XT断路器实验

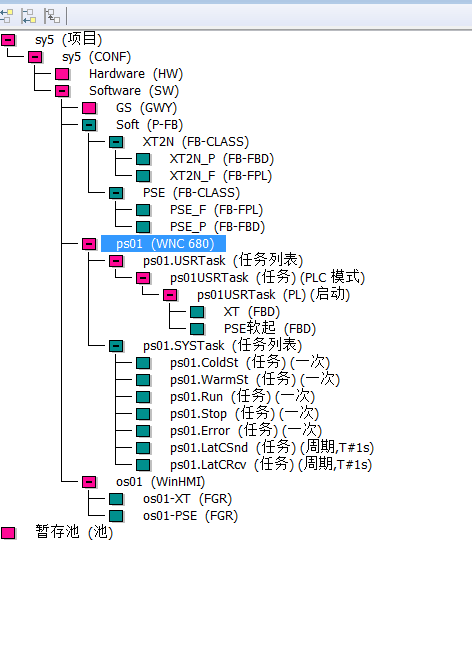
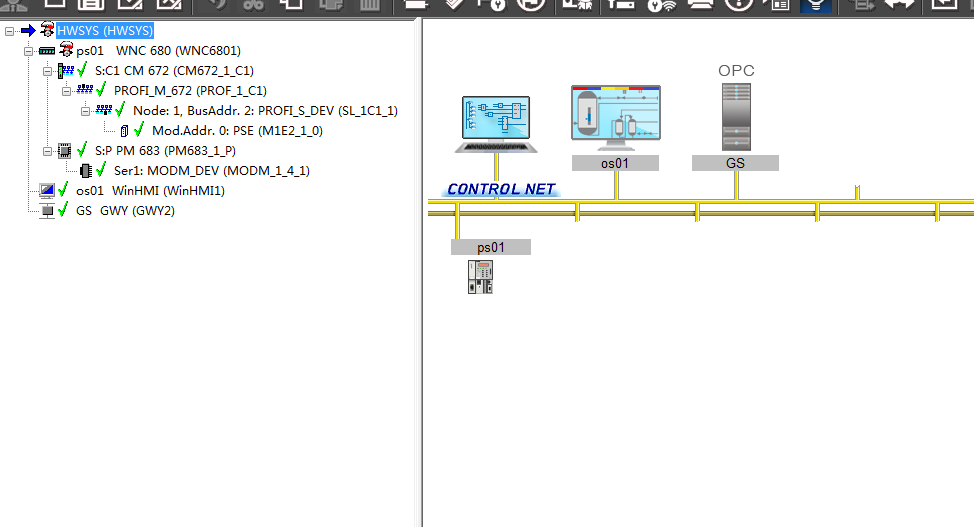
**引言：**

在POE系统中，提供电力的叫做“供电设备”（PSE， Power Sourcing Equipment），本次实验使用PSE软启动器控制电机，ABB PSE软启动器是一种集电机软起动、软停车、轻载节能和多种保护功能于一体的新颖电机控制装置，可以实现现场总线控制(通过以太网或者OPC客户端)，同时该实验集成XT断路器模块。

我做过的现场总线所有相关源码同步发布在github：

<https://github.com/hgez6/ABB-modbus/tree/master>

1. **软硬件组态**

****

软硬件组态如上图所示

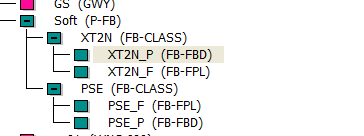
本次实验用到XT模块，PSE软启动器模块，所以用两个FBD块进行图形化编程，在FGR显示方面也是插入两个界面以实现切换。值得注意的是要实现OPC远控还要在软硬件中均插入GS网关。

硬件方面要特别关注M672下插入支持MODbus协议走TCP/IP总线的PSE软启动器控制电机启动。





FGR块绘制如图所示

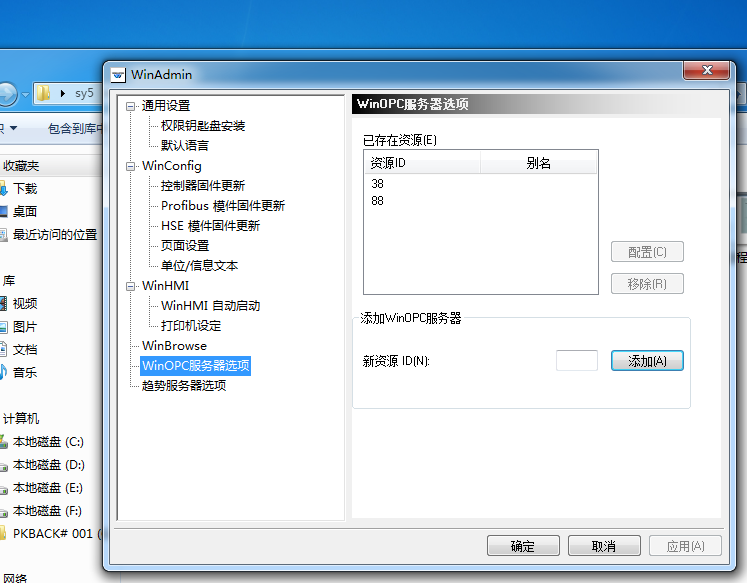


并且需要导入两个封装好的功能块

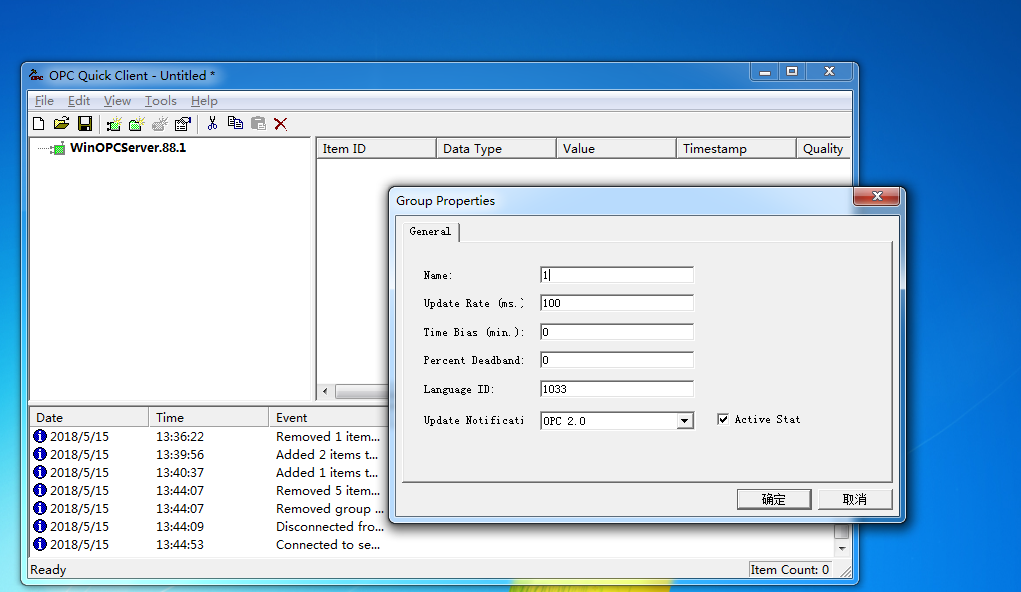
1. **画面的组态和各个关键控件的描述**

**OPC:**

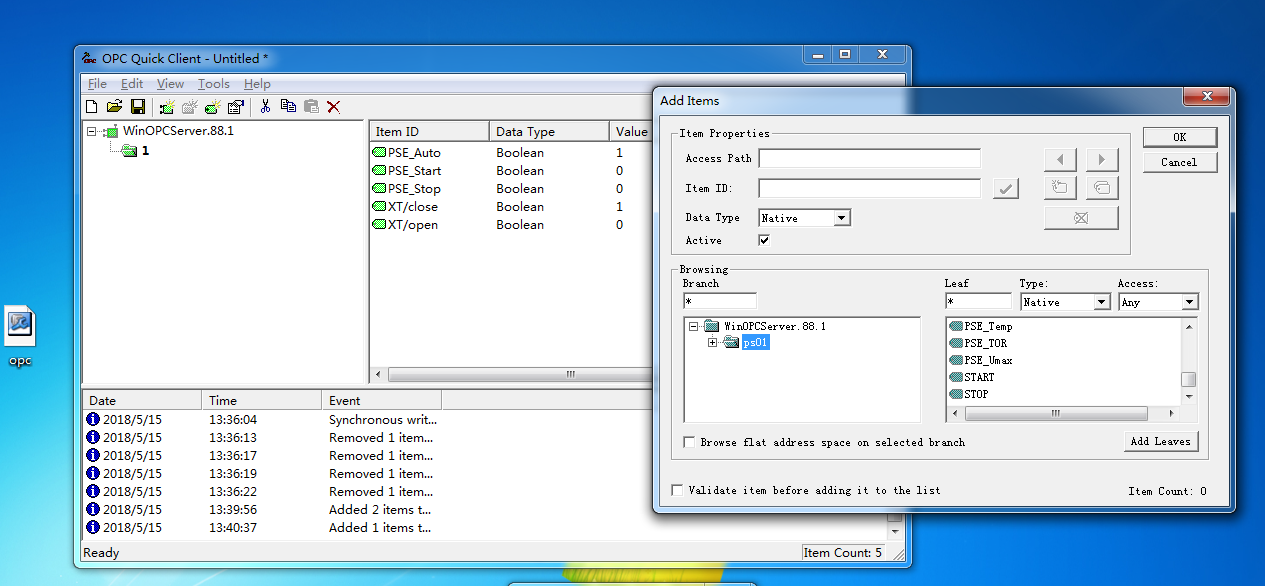
首先介绍OPC的使用因为OPC客户端可作为远程上位机，通过变量对PSE和XT断路器控制。



首先要设置资源id并记录id将决定opc sever的序号。

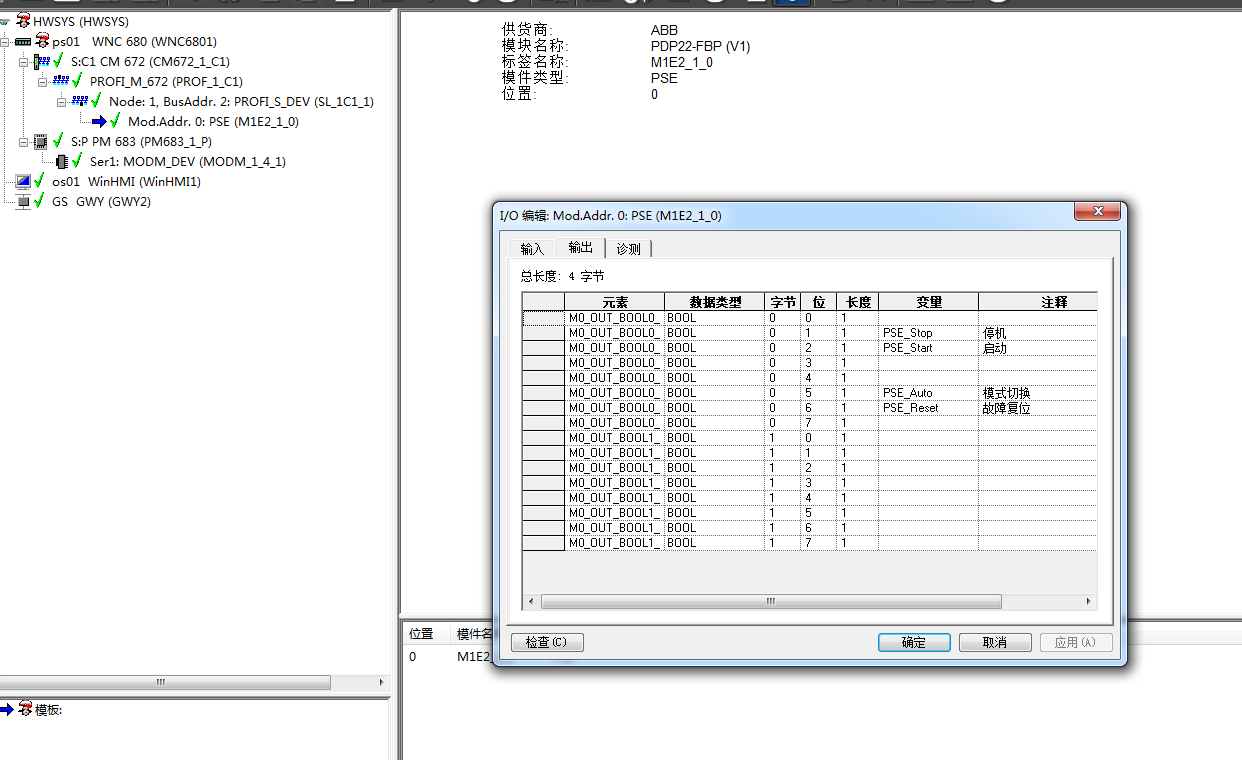


然后打开OPC客户端选择刚刚的id新建文件



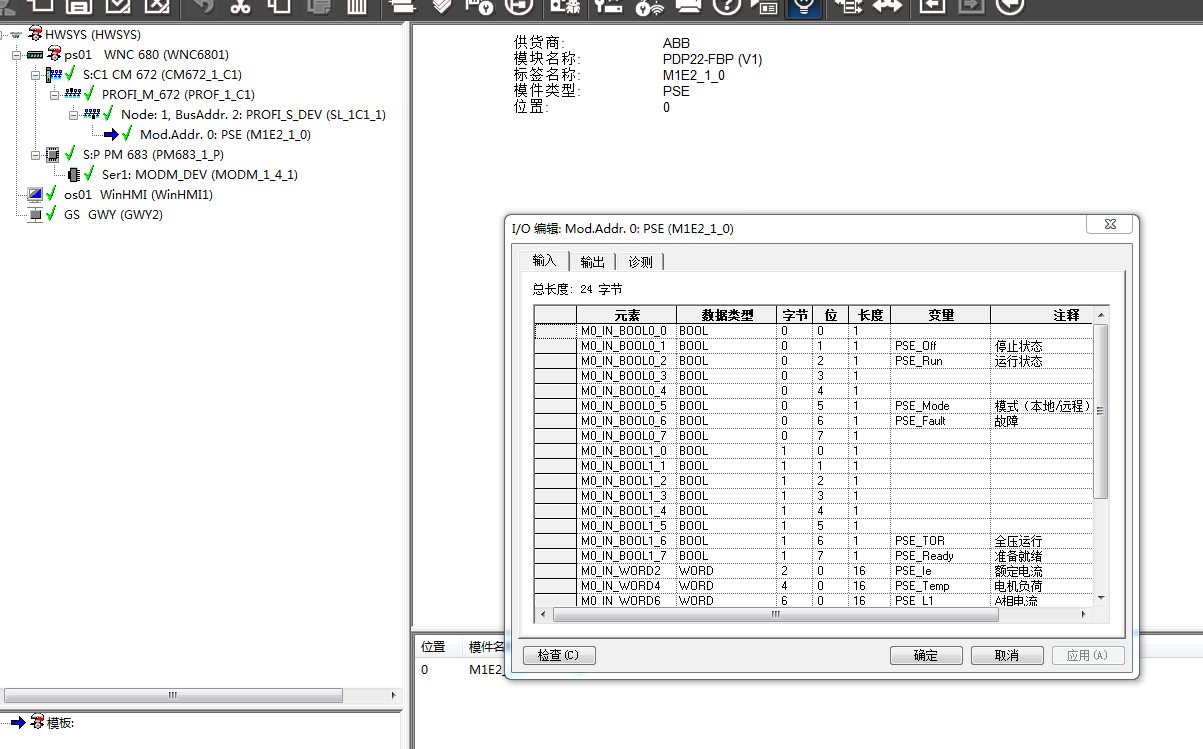
然后在ps01中找到变量并插入，PSE软启动器使用三个变量auto是切换本地和远程，XT插入两个断路器的通断，以上变量均为bool量，为下文使用OPC模式控制PSE，XT打下基础。

PSE:

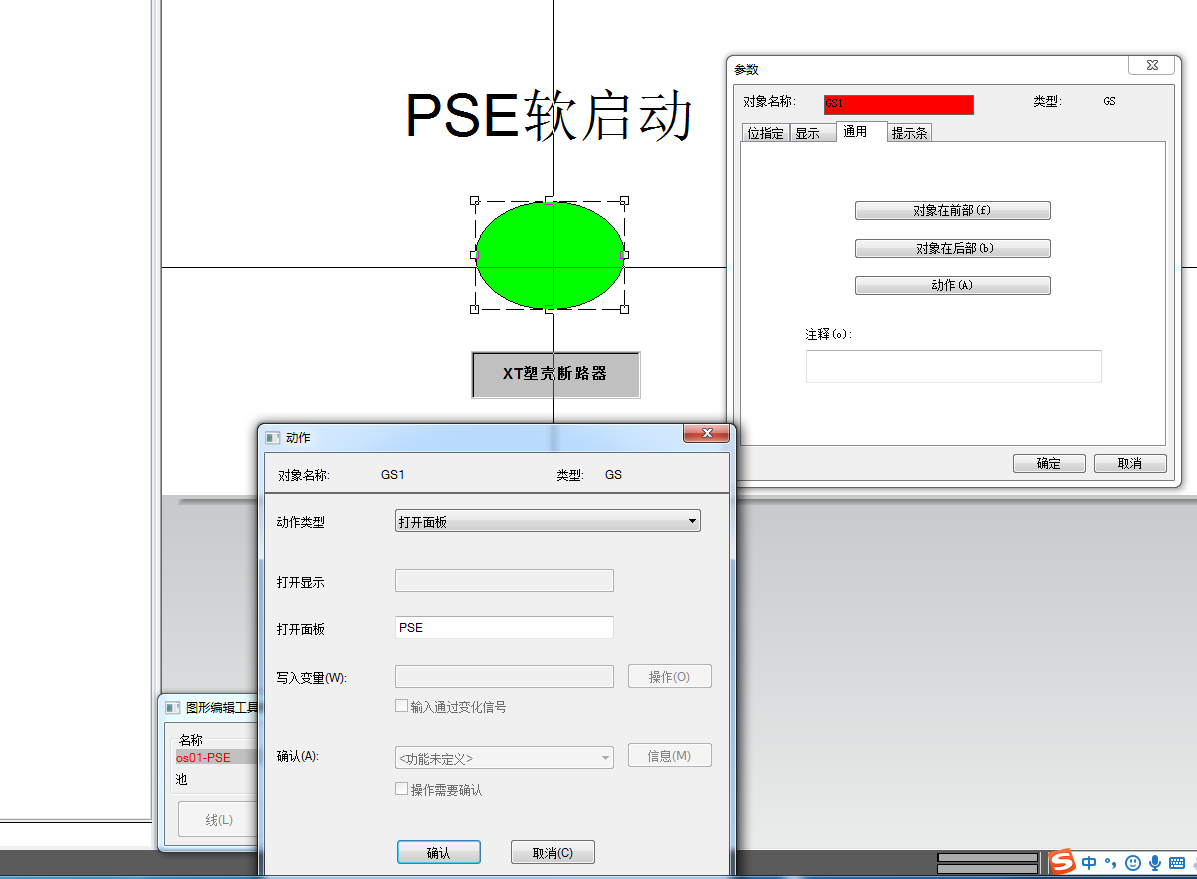


输出IO编辑如上图

在PSE模块中右击对他的地址进行编辑，赋予不同的变量方便后续操作，不同的地址IO输出对应不同的功能，详细需要看操作手册对PSE IO的定义，也要根据实际接线来判断。



输入IO编辑如上图用于检测电机状态，通常都是ABB预先定义好的。

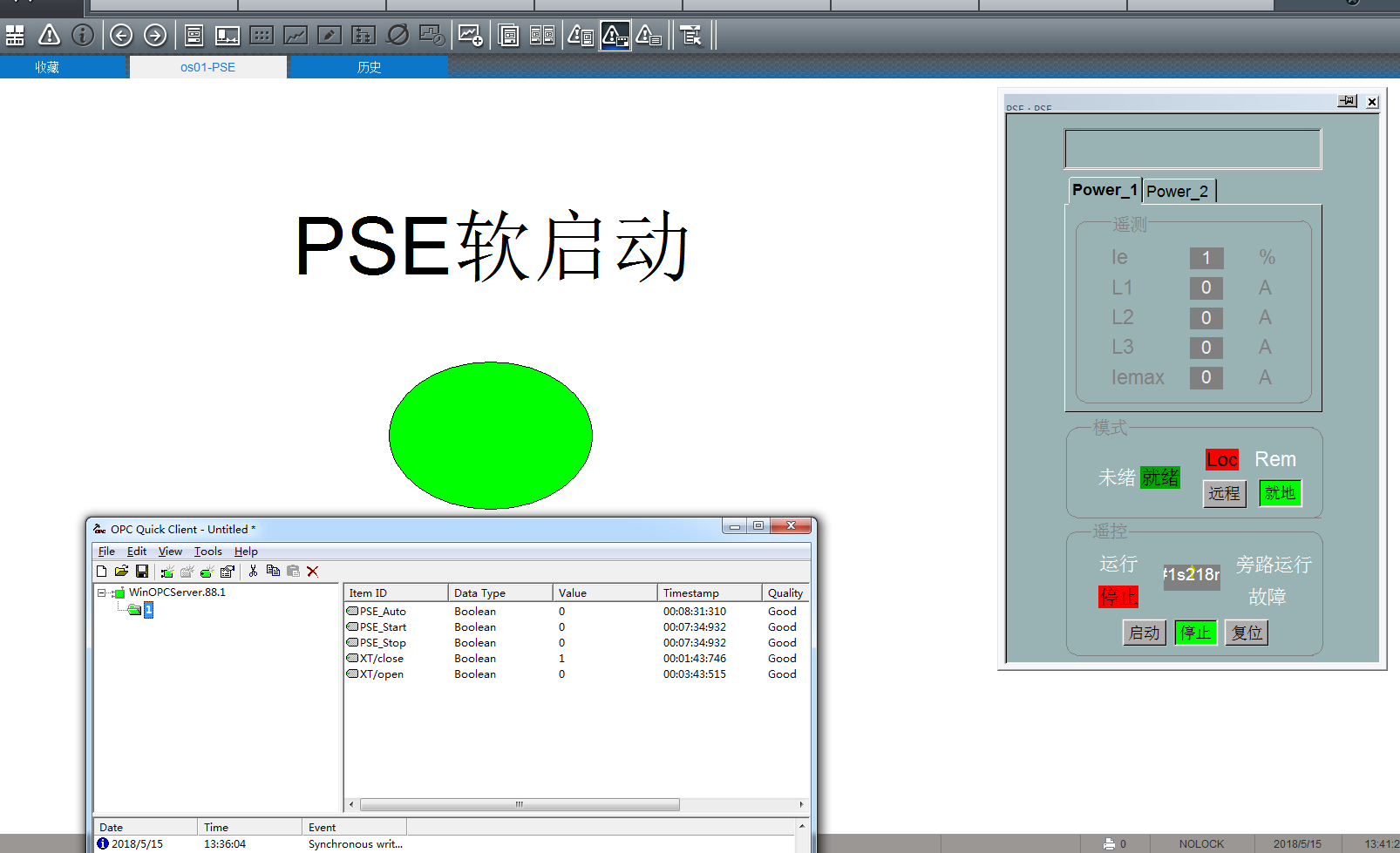


设置完IO变量后设置FGR块，右击添加动作使其能在PSE和XT断路器之间切。



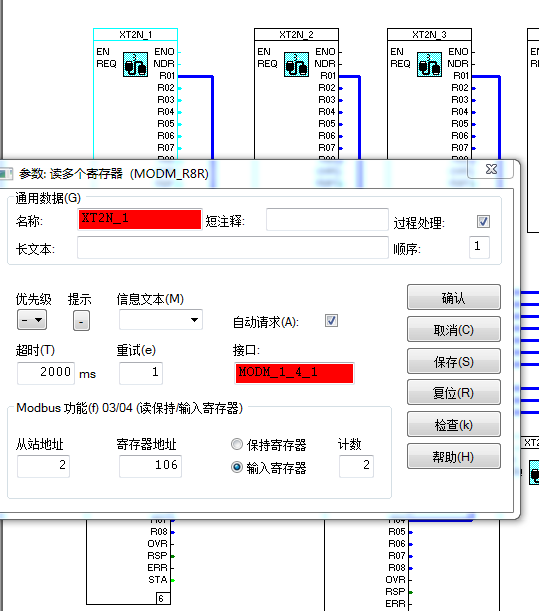
PSE实验效果如上图

PSE远程可通过OPC客户端控制或者直接走MODbus总线控制对于如何使用opc将在上文已经介绍，两种方式控制的都是同一个变量只是走的协议不同。



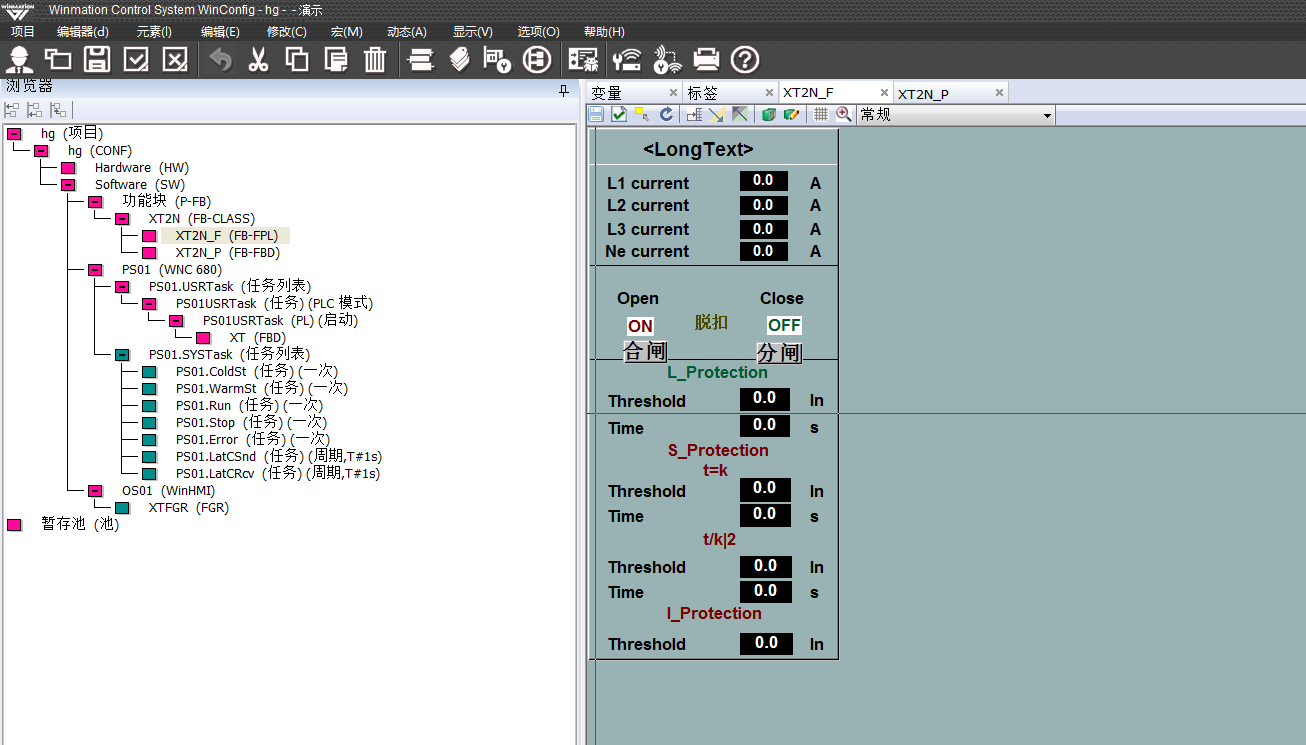
上图为就地控制

XT：

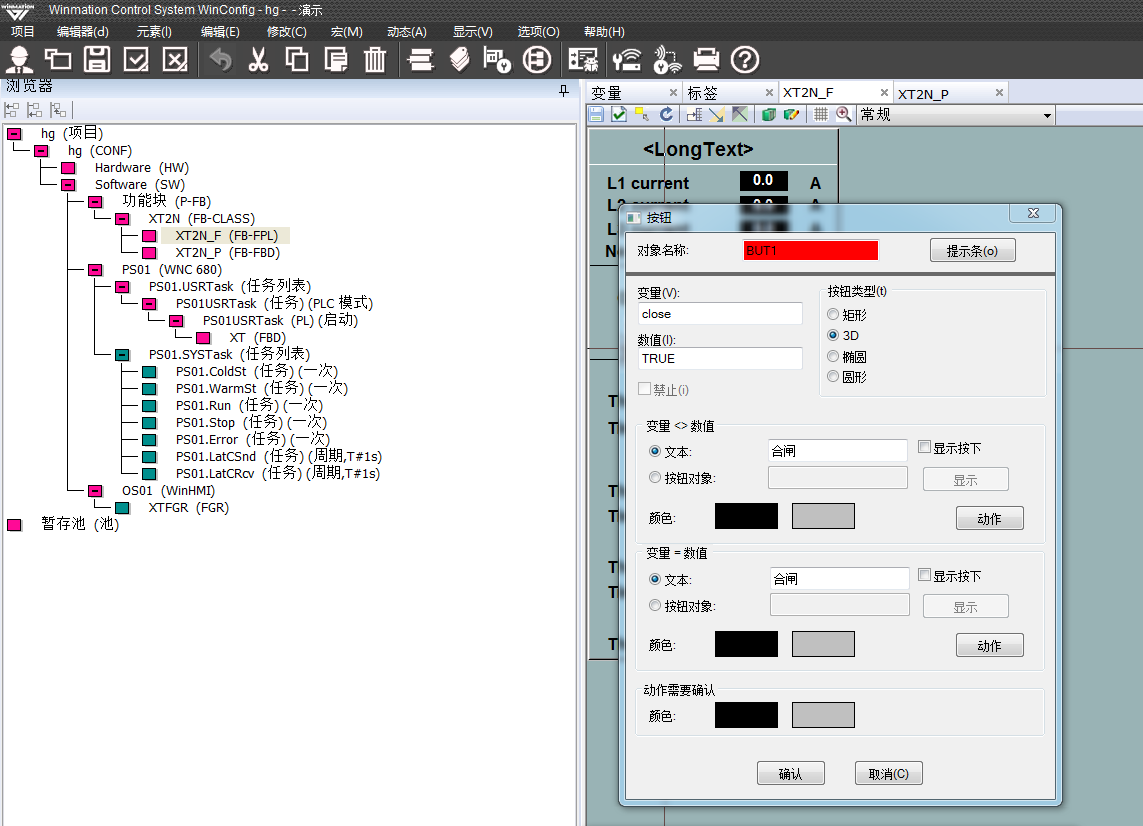


使用XT之前先要设置寄存器的地址，以用来控制不同的IO。

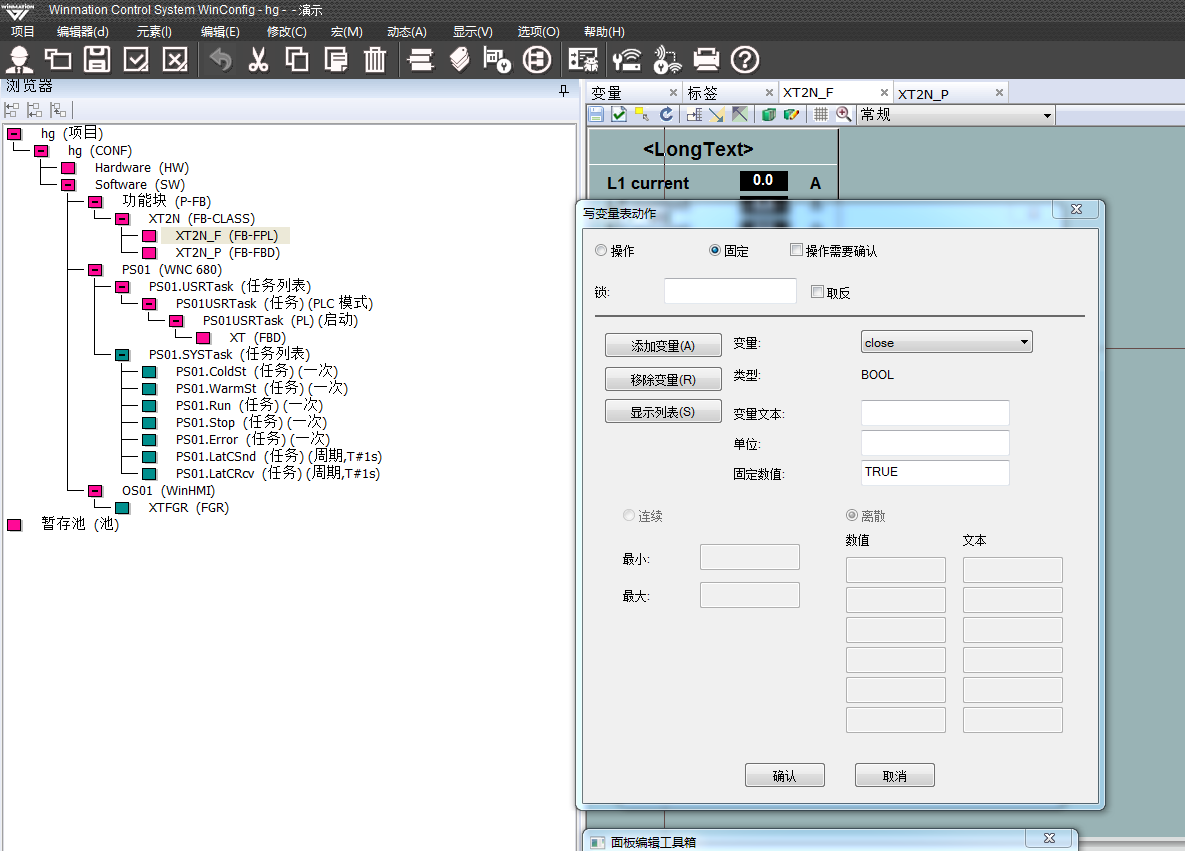
然后可以对封装好的块进行设置



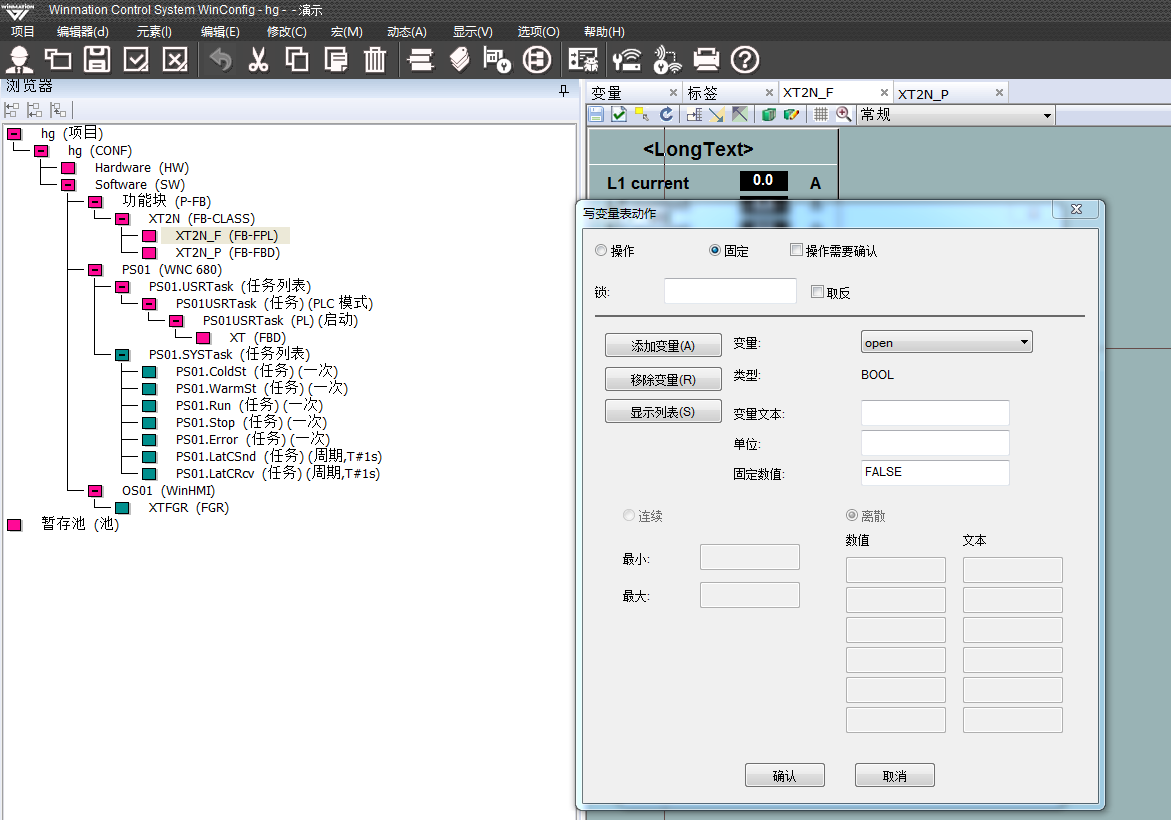
右击改变对象的属性也针对不同对象实现自定义。如分闸按钮控制的close变量置位为1时分闸成功。

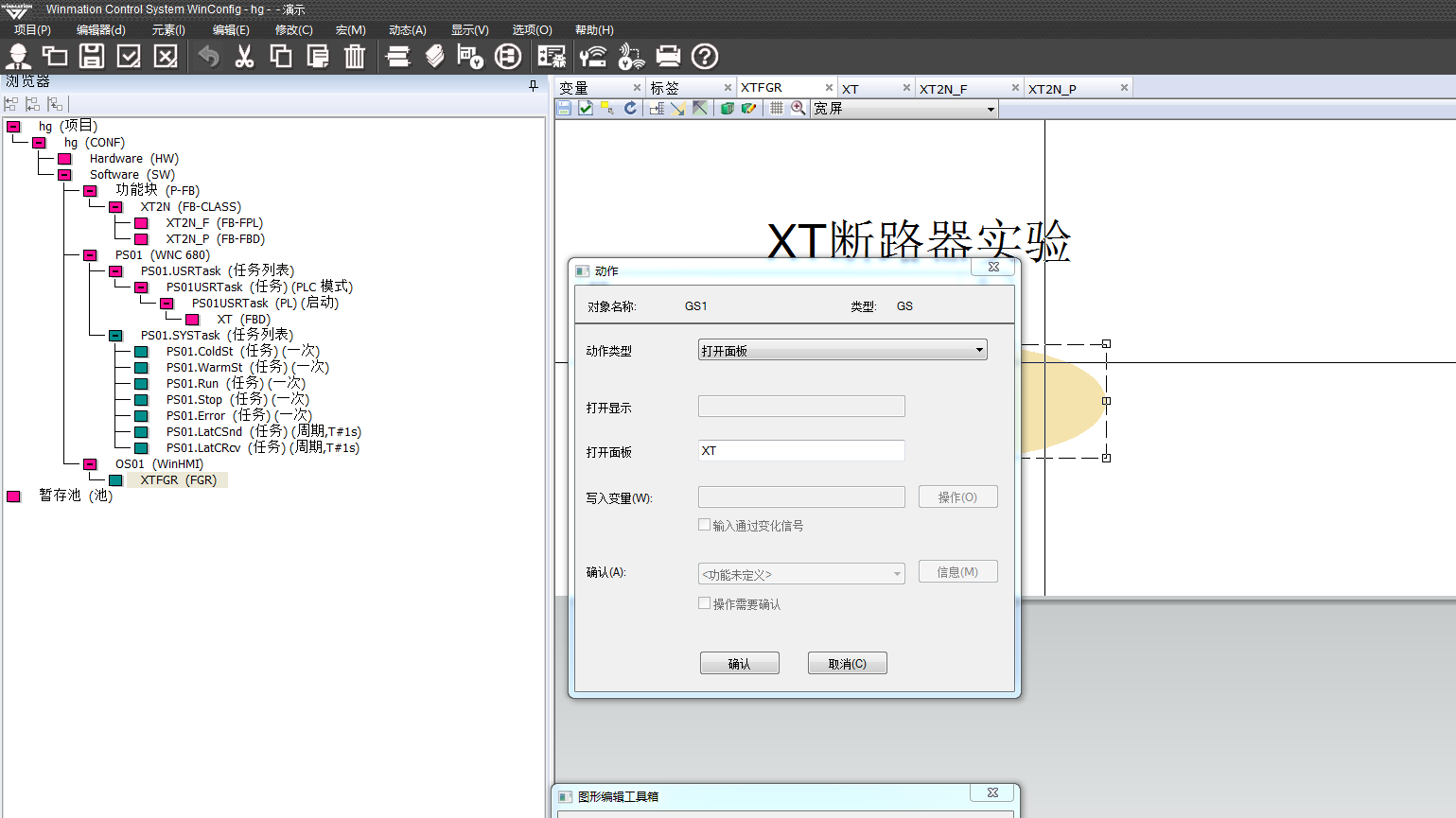


具体动作如下图所示，写变量表中“close”变量另其固定为1，类型为BOOL型

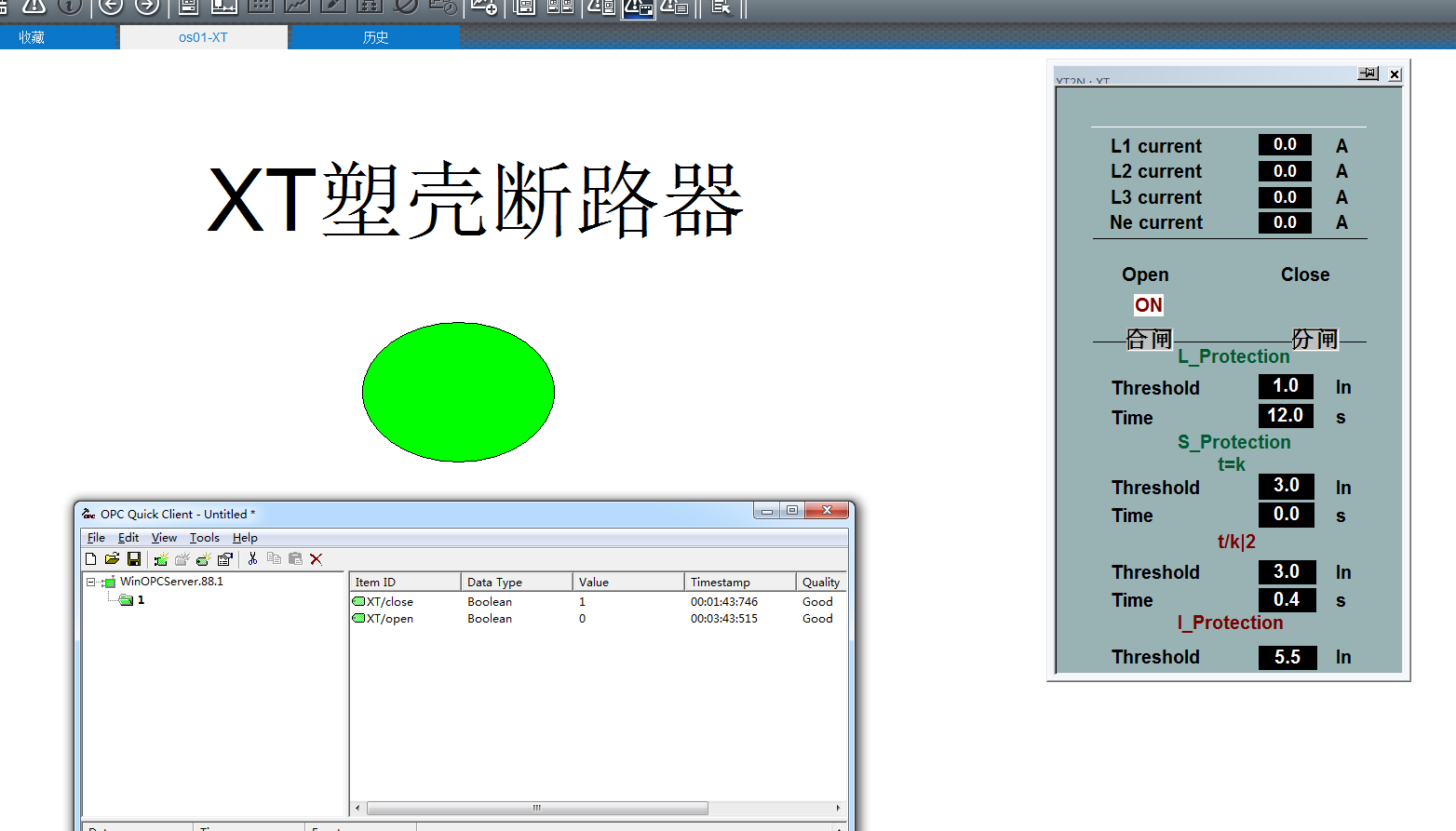


合闸也是同理如下图所示：





自己编写的FGR块用于打开面板 XT 即，封装好的操作界面。



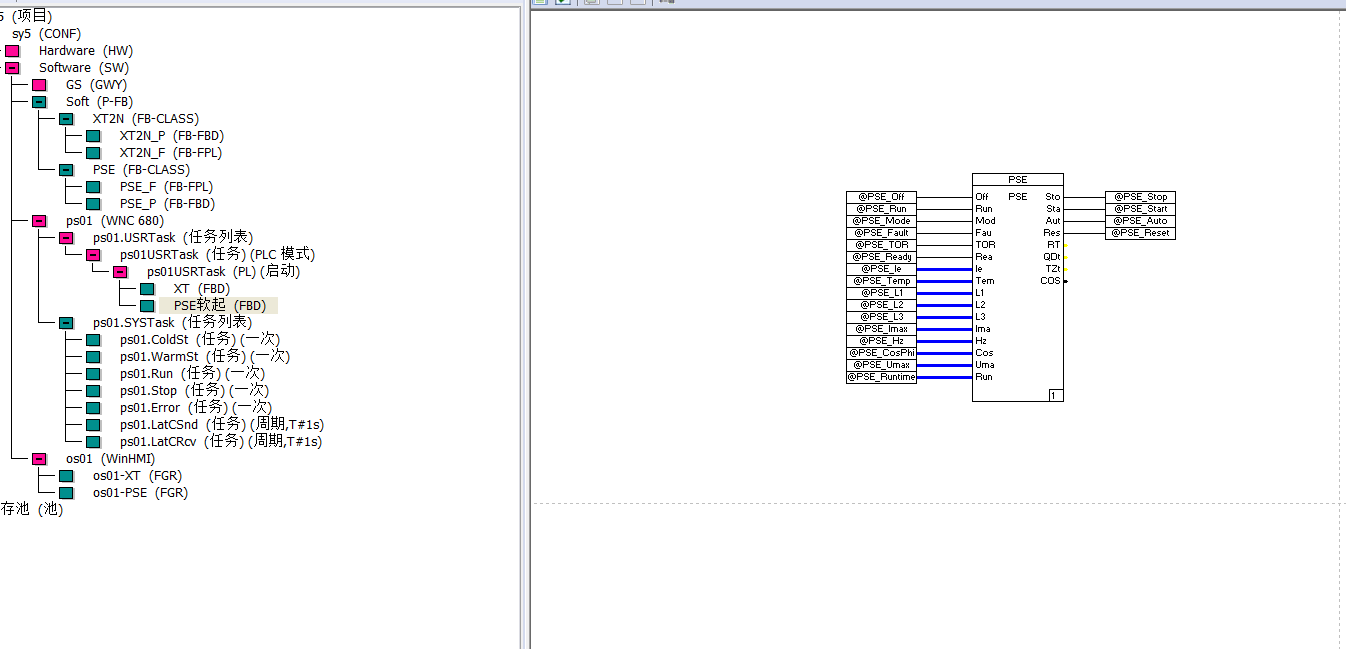
实验效果如上图所示，OPC控制可仅仅是添加两个bool量就能实现，毕竟只是通断而已。

1. **程序块的组态和对程序的描述**

**Opc:**

opc是一种远程控制方案可对预先设置的变量进行读写，以实现控制的效果。属于线程总线控制协议的一部分，与modbus控制的不同点是后者使用TCP/IP，而前者强调客户端和服务器的通讯。

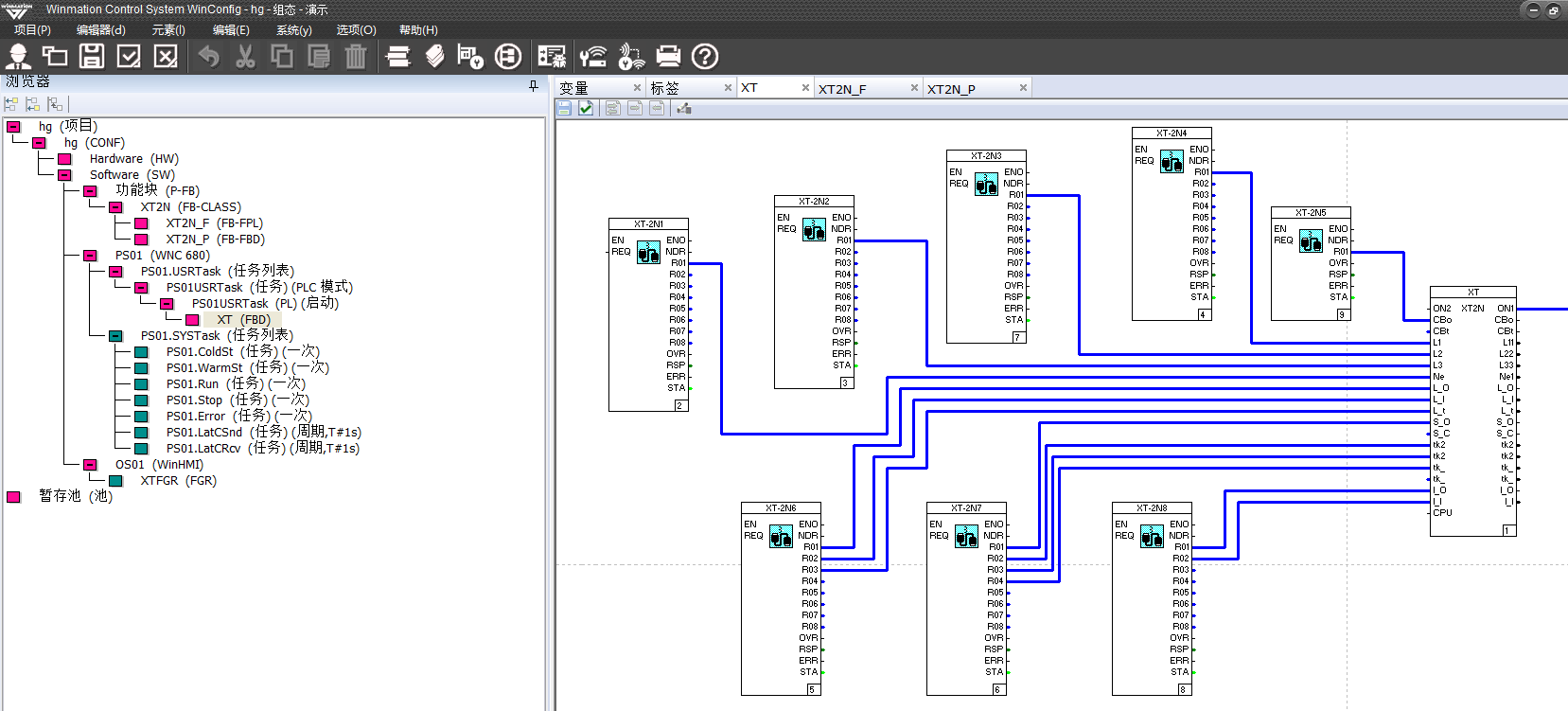
**PSE:**

****

对于PSE我们使用变量连接程序块，左边是电机给PSE的输入量（状态监测信号），右边是输出量用于控制电机的启动，停止，切换控制方式

**XT:**

由于封装的模块比较复杂。该程序编程时需要在硬件部分先写变量表来进行控制。写完变量表后在PS01USRTASK下一级进行FBD块的图像化编程。



中间XT模块的引脚对于与封装好的模块的各个量并于各个寄存器的不同地址相连接用于监控设备状态和电流值如各个相的电流L1，L2，L3，Ne current 和其他信号，用于显示在FGR中。

1. **调试过程的描述**

本次实验难点是同一项目下模块的集成度高，但是各个部分作用明确并没有功能或者IO口重叠情况所以，应该严格遵照，高内聚，低耦合思想编程。

1. **实验总结**

其实本次实验我因为疫情原因并没有现场调试，但是通过组员 和 的帮助下，理解了此次实验的过程，因为本次实验是融合前三个实验成一个项目，这三个实验每一个之前都做过，所以体现了应该注重平时积累的重要性。

感谢 。