

**IntegrityServer. Отображения технологических мнемосхем (IntegrityHMI)****Цель работы:**

1. Закрепление навыков работы с интерфейсом **IntegrityHMI**.
2. Обязательное применение изученных инструментов графики и расположения для эффективной работы при создании объектов.
3. Создание графической мнемосхемы учебного стенда «Булевы функции»
4. Создание графической мнемосхемы «RS – триггер с прямыми входами на элементах ИЛИ-НЕ»
5. Создание графической мнемосхемы «Функциональный блок Т»
6. Создание графической мнемосхемы «Блок автоматики»

**Упражнение 1. Создание графической мнемосхемы учебного стенда «Булевы функции».**

Для начала работы запустите **IntegrityHMI** и создайте в личной директории новый проект мнемосхемы.

! В меню **Сервис** выберите пункт **Параметры** и вкладку Редактор. Настройте данное окно как показано на Рис. 1. Данным действием вы включите **сетку** для удобства выполнения работы. Цвет сетки поставьте, например, черный.

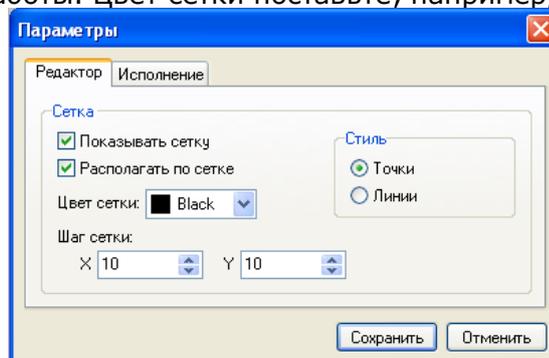


Рис. 1

В главном меню **Вид** отметьте **Цветовую палитру** (Рис. 2). Цвет палитры выберите темно-серый. После этого все объекты типа **Прямоугольник** и **Эллипс/Круг** будут иметь темно-серую заливку и палитру, если она мешает можно убрать.

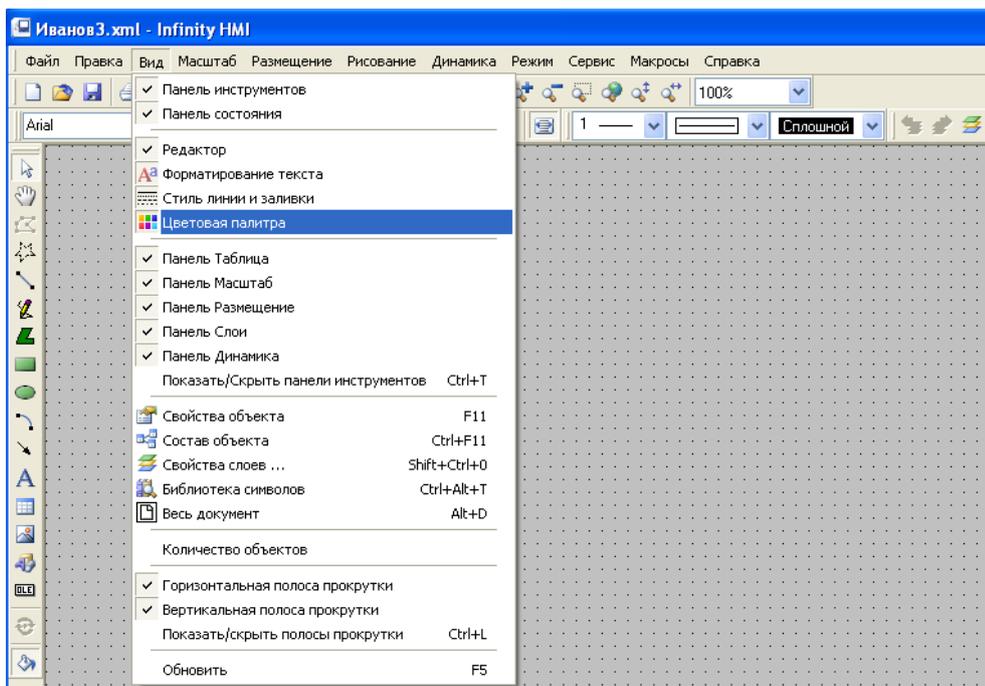


Рис. 2

Нарисуйте с использованием объекта **Прямоугольник** Логический элемент **И (AND)**. Размеры приблизительно, визуально около 20 мм × 40 мм (Рисунок 3).

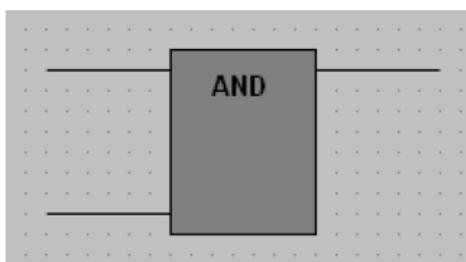


Рисунок 3

Создайте также два объекта **Кнопка** и настройте динамику **Переключение значения**, привязанную к соответствующим сигналам **Управление 1** и **Управление 2**. Для эффективного размещения объектов применяйте соответствующие инструменты:

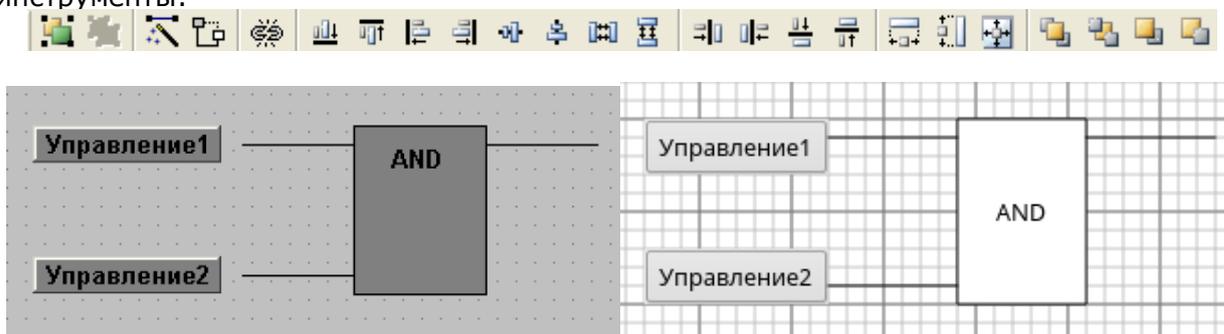


Рисунок 4

На Рисунке 4 есть семь объектов, которые в избежание смещения относительно друг друга можно (и нужно!) сгруппировать. Доступ к любому объекту в сгруппированном символе осуществляется двойным щелчком левой кнопкой мыши на объекте или **вызовом на выделенном символе из контекстного меню** (правая кнопка мыши) опции **Свойства объекта**. После этого появится окно **Свойства объекта**. В свойствах объекта в закладке **Дочерние объекты** (Рис. 5) можно выбрать любой

объект, который находится в составе данного объекта, и изменить его свойства. После того, как необходимые изменения были произведены нужно нажать на кнопку **Назад** (Рис. 6).

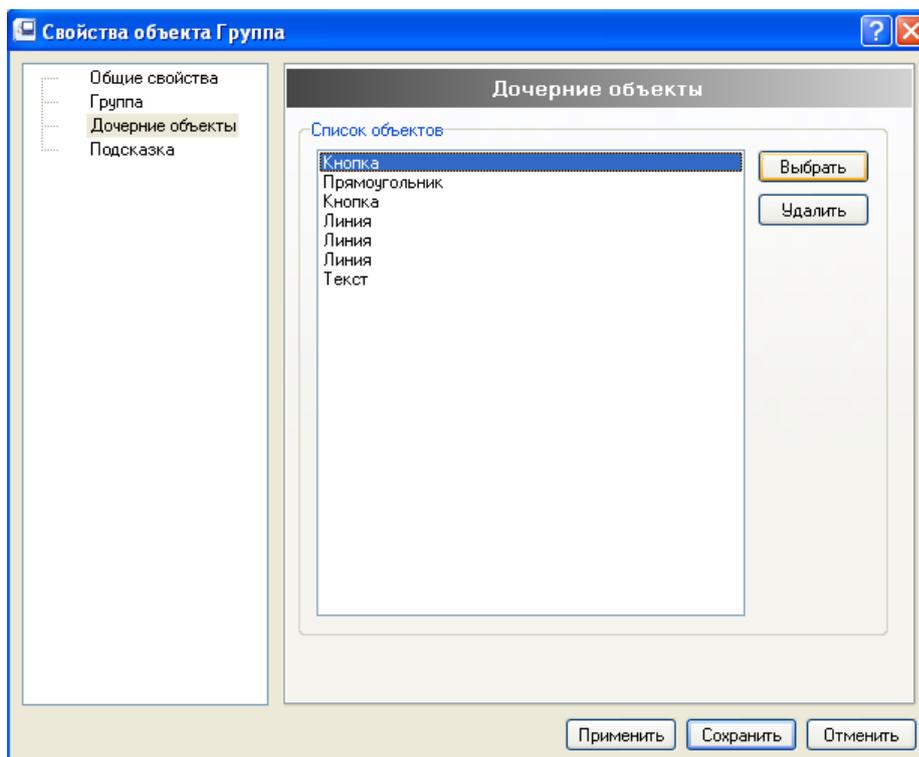


Рис. 5

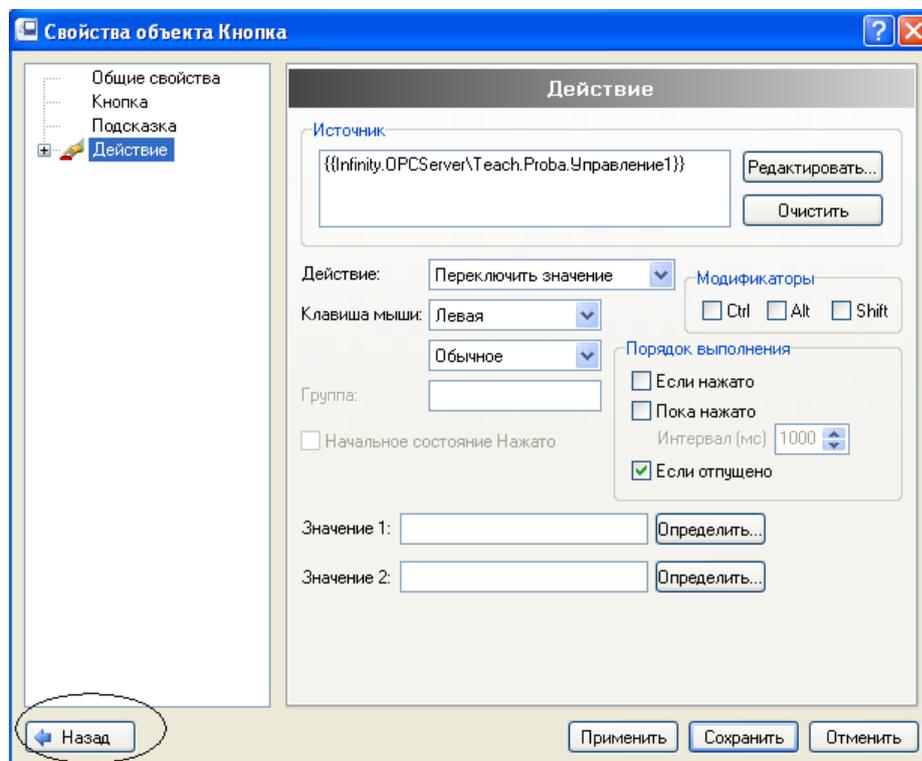


Рис. 6

Предположим, что битовый сигнал **Управление 1** будет приходить на верхний по схеме вход логического элемента, а **Управление 2** на нижний. **Битовый сигнал Состояние 1** – это выходной сигнал (выход элемента). Примените к этим линиям динамику так, чтобы при единичном значении сигналов соответствующие линии

становились зелеными, а при нулевых значениях красными. В режиме **Исполнение** все линии должны быть черными. Сейчас и далее группируйте объекты в символы, а доступ к объектам в символах осуществляйте опцией **Свойства объекта**.

Примените к **выходному сигналу (выход элемента)** динамику **Дискретный цвет**, как показано на Рисунке 7. В поле **Условие** введите выражение логического умножения для сигналов **Управление 1** и **Управление 2**.

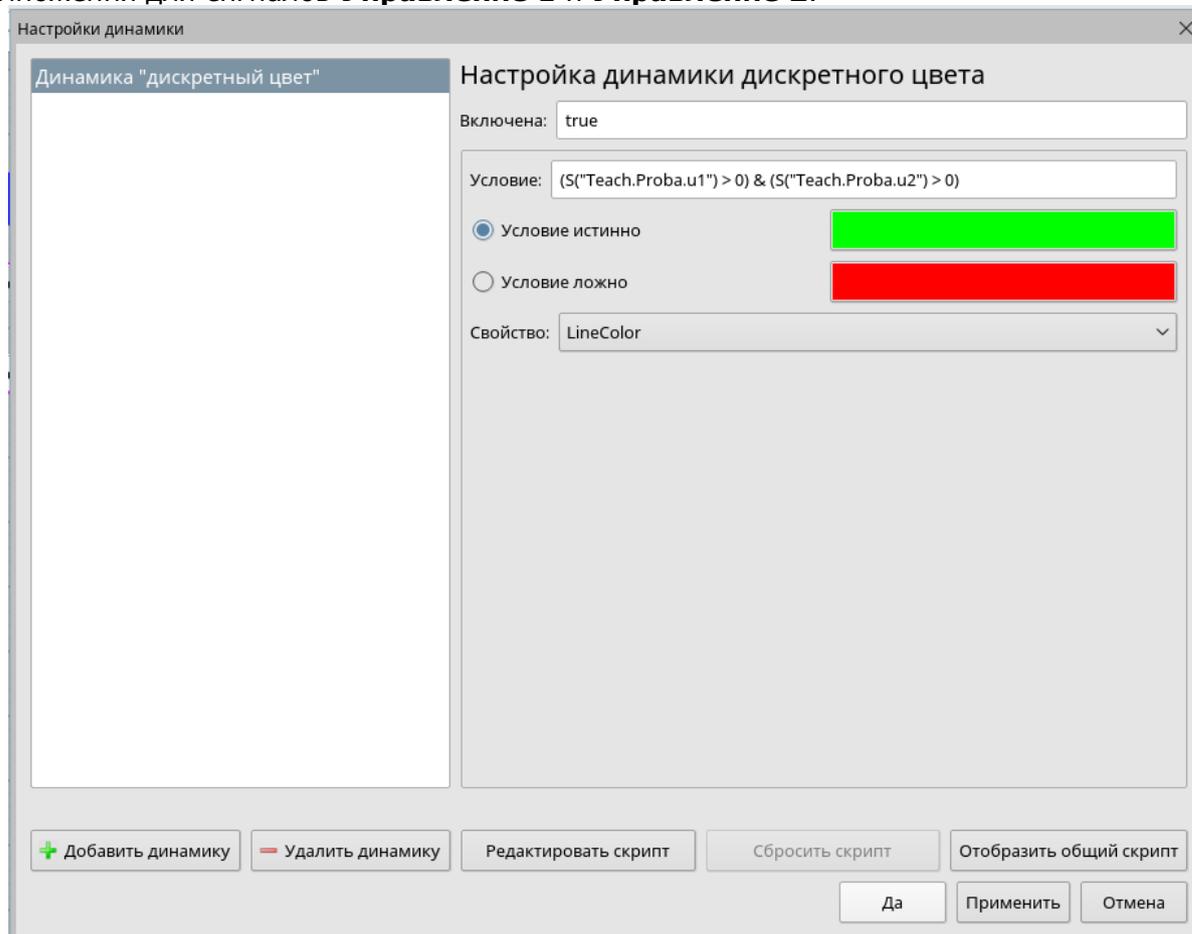


Рисунок 7 – Условие описание логического элемента **И (AND)**

Закройте окно настройки динамики и проверьте работу мнемосхемы. Любой ноль на входе даст ноль на выходе.

**ВАЖНО!** Еще раз напоминаем: перед выходом из режима Проект – Старт приводите сигналы в исходное (нулевое) состояние.

Создайте объекты логических элементов: **И, ИЛИ, И-НЕ, ИЛИ-НЕ**. Рис. 8

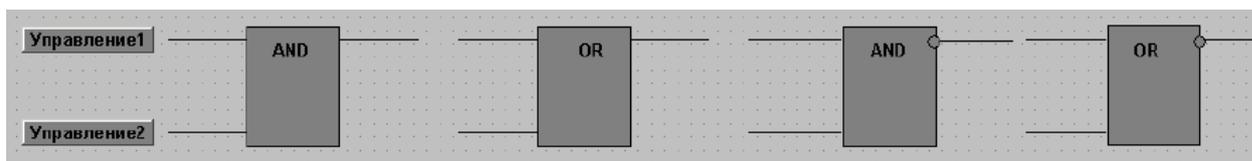


Рисунок 8

Входные сигналы для всех прежние: **Управление 1** и **Управление 2**, а к выходным сигналам необходимо применить динамику **Дискретный Цвет** по аналогии с выходом созданного объекта **Логический элемент И (AND)**. Логические выражения, естественно, должны быть для каждого логического элемента свои. **Необходимо заметить, что логическое отрицание в IntegrityMI обозначается как восклицательный знак ! и имеет высший приоритет, там, где это требуется,**

**применяйте простые ( ) скобки.** Очевидно, что это замечание будет актуально во время ввода выражений для функций И-НЕ, ИЛИ-НЕ, отсутствие скобок перед инверсией выражения для этих функций !( & ). **Также следует учитывать, что при написании формул каждую операцию необходимо заключать в простые скобки.**

Примеры приведены на рисунках 9-10.

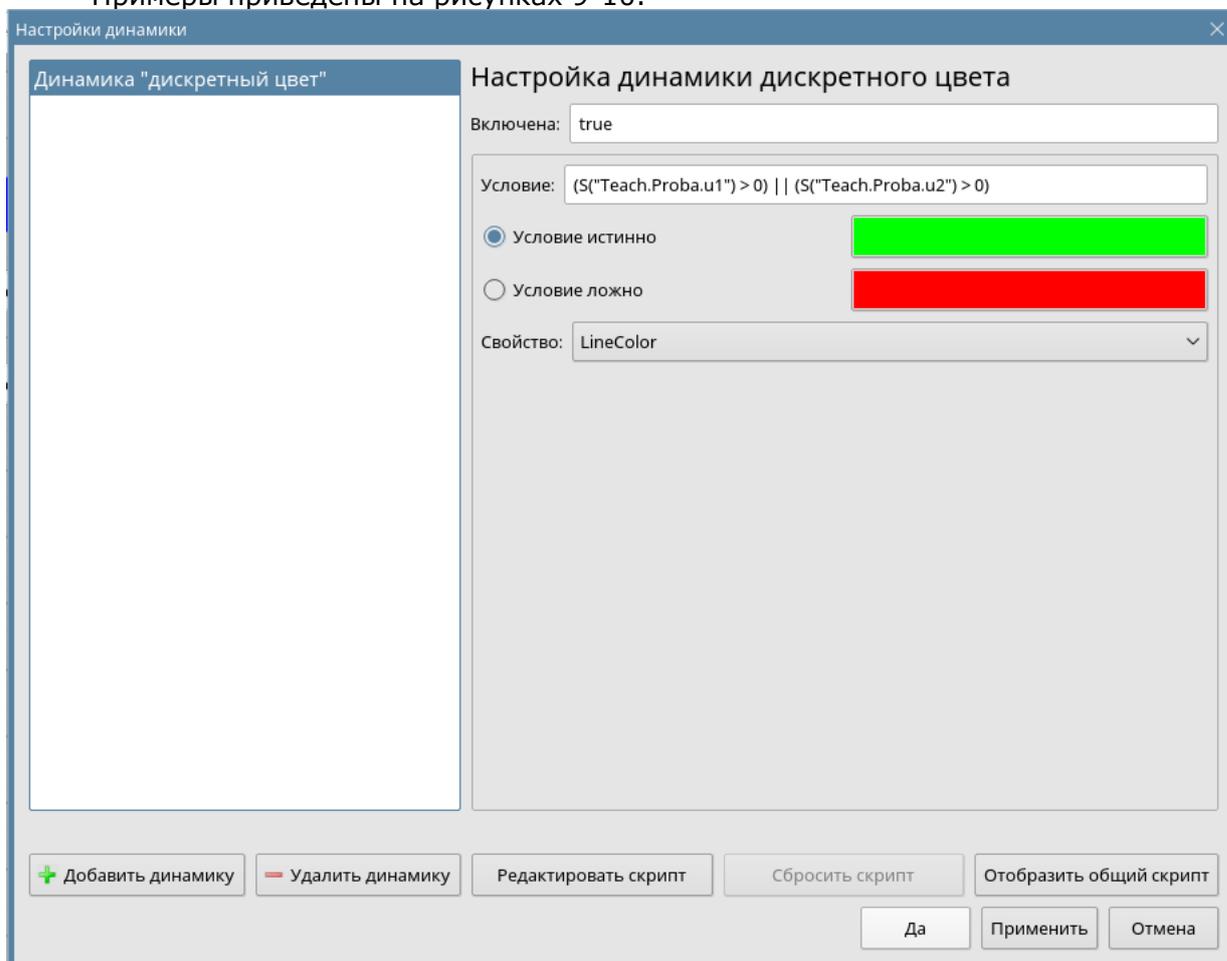


Рисунок 9 – Условие описание логического элемента **ИЛИ (OR)**

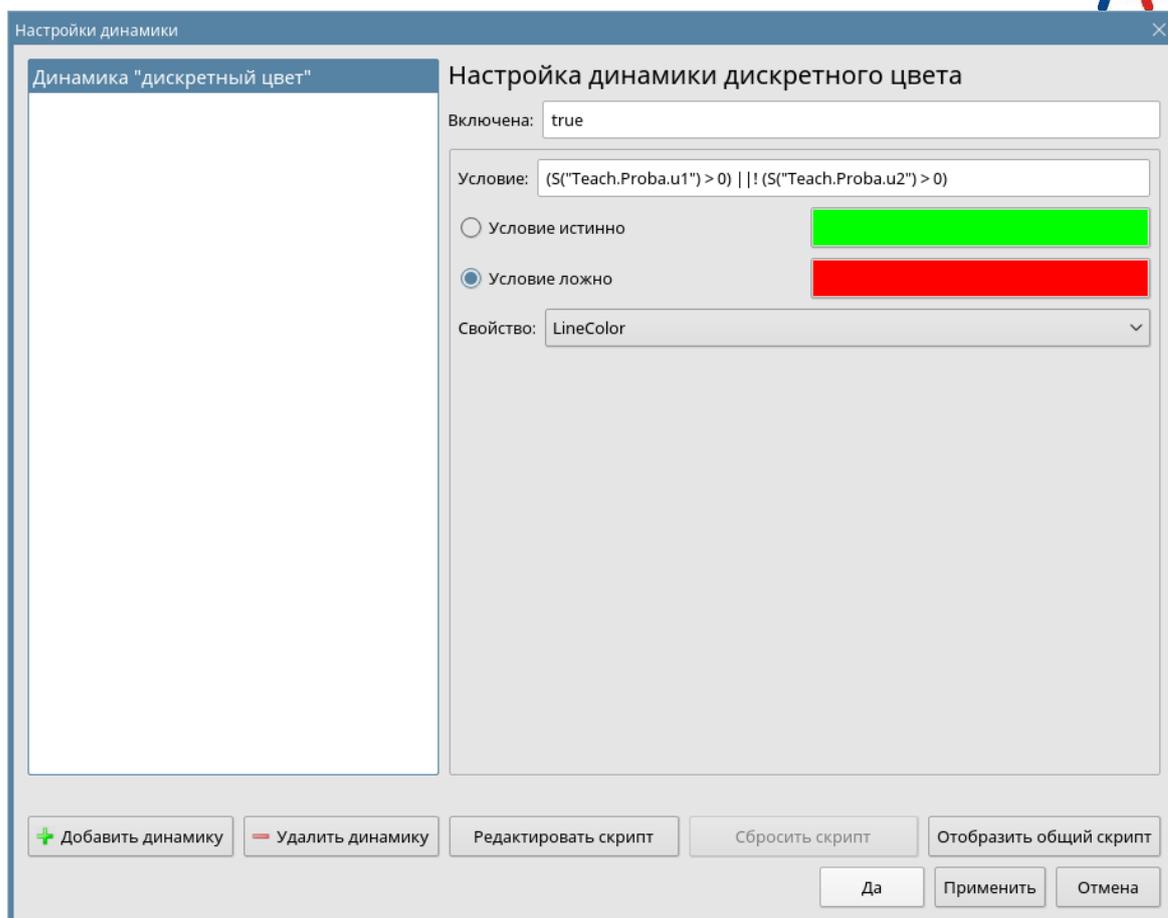


Рисунок 10 – Условие описание логического элемента **ИЛИ-НЕ (OR!)**

Расположите объекты в рабочей области как показано на Рисунке 11 и сгруппируйте их в один символ.

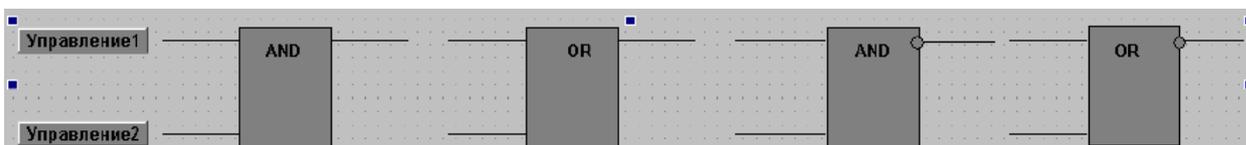


Рисунок 11

Нарисуйте «основание» стенда, цвет заливки: **синий**, объем: **выпуклый**.

С помощью инструмента **Текст** создайте название стенда. Вид экранной формы показан на Рисунке 12.



Рисунок 12

Проверьте экранную форму в режиме **Исполнение**.

С помощью необходимых инструментов создайте объект **Таблица**, где будут отображаться значение всех сигналов. Общий вид может быть таким, как на Рисунке 13.

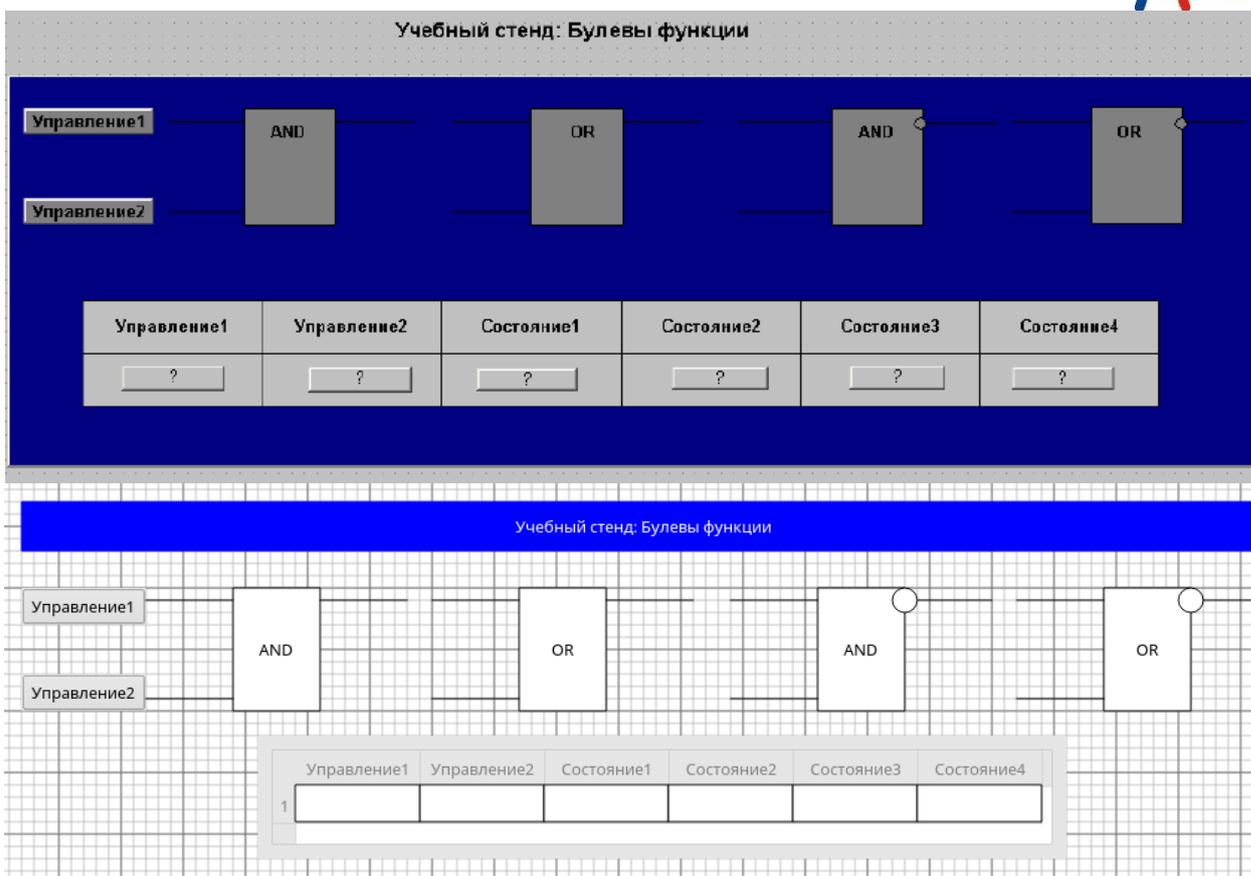


Рисунок 13

Сохраните экранную форму. Пример отображения учебного стенда "Булевы функции" в режиме Исполнение показаны на рисунках 14-15.

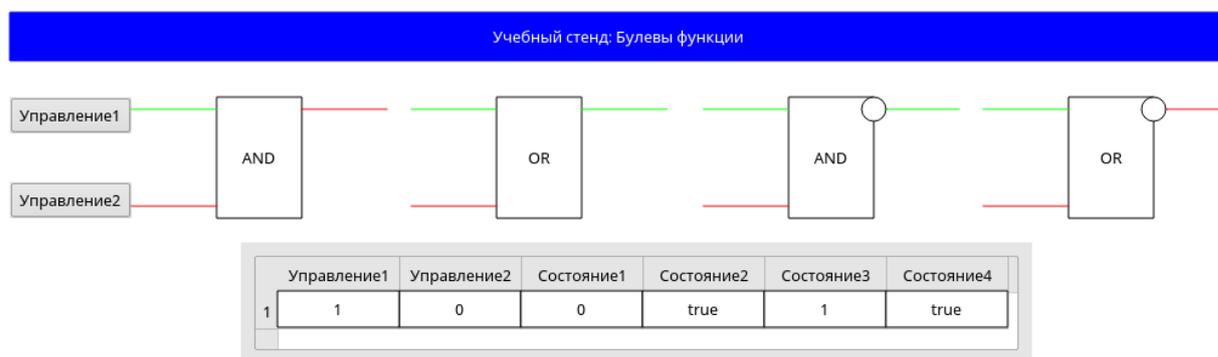


Рисунок 14

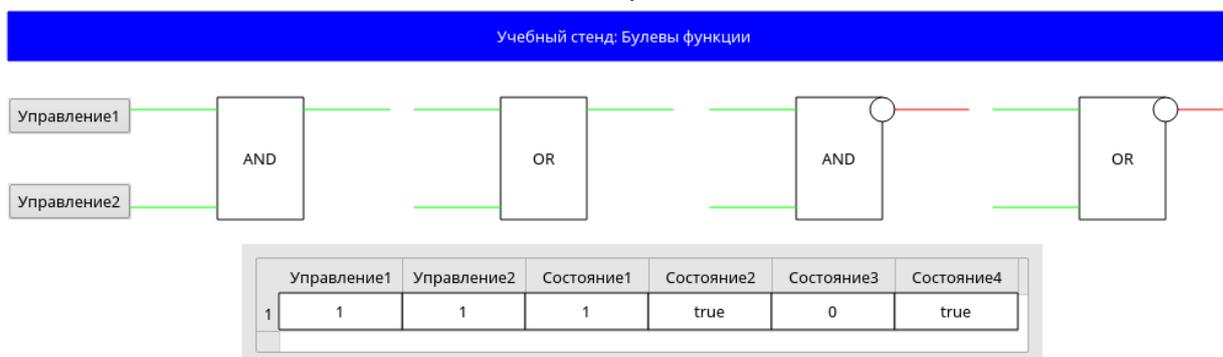


Рисунок 15