

федеральное государственное бюджетное образовательное учреждение высшего  
образования

Кафедра автоматизации технологических процессов и производств  
**Домашняя работа**

Санкт-Петербург  
2024

## Цель работы

Моделирование системы теплообмена, суть которой заключается нагревании расхода холодная вода, проходящего через трубки теплообменника другим расходом горячей рецикловой воды, проходящего внутри корпуса теплообменника с фиксированной температурой, указанной по вариантам. Моделирование реализуется в среде программирования ПЛК – MATLAB с подключением к SCADA-системе, спроектированной в программном продукте Proficy iFix, на которой реализуется визуальный контроль процесса по мнемосхеме, с регулированием требуемых параметров. Подключение настраивается через OPC Factory Server.

## Исходные данные

Таблица 1. Исходные данные

№ вар.	G1, л	T1, л/ч	G2, г/л	T2, л/ч	M, г/л	τ, сек	C1=C2
14	300	100	700	40	600	0,4	4200

## Теория о регулируемом процессе

В теплообменник подается расход воды на нагревание рецикловой горячей водой при заданной температуре. Суть процесса – регулирование подачи расходов теплоносителя и смеси, которую нагреваем, а также температуры этой смеси с получением заданной выходной путем воздействия на краны, клапаны, дроссели, присутствующие в схеме.

Изменение выходной температуры описывается дифференциальным уравнением:

$$T_{out} = \frac{1}{M} \cdot \int_0^{\infty} \left( \frac{G_1 \cdot C_1 \cdot T'_1 + G_2 \cdot C_2 \cdot T'_2 - (G_1 \cdot C_1 + G_2 \cdot C_2) \cdot T_{out}}{G_1 + G_2} + T_{nom} \right) dt$$

где

M – общая масса вещества, которая одновременно может быть в теплообменном агрегате как в трубном, так и межтрубном пространстве;

$T_{out}$  – целевой параметр, температура смеси (которую мы нагреваем) на выходе из теплообменника;

$G_1, G_2$  – расход теплоносителя и расход смеси, которую нагреваем, соответственно;

$C_1, C_2$  – теплоемкость теплоносителя и смеси, которую нагреваем, соответственно;

$T_1', T_2'$  – температура теплоносителя и смеси, которую нагреваем, на входе в агрегат соответственно;

$T_{nom}$  – значение температуры на выходе теплообменника, рассчитываемое по формуле:

$$T_{nom} = \frac{G_1 \cdot T_1' + G_2 \cdot T_2'}{G_1 + G_2} = \frac{300 \cdot 100 + 700 \cdot 40}{1000} = 58^\circ\text{C}$$

Это та температура, которую имеет смесь на выходе из агрегата, при условии, что агрегат запущен, прогрет, заполнен жидкостями и по расходам и температурам обоих потоков выставлены регламентные значения.

## Ход работы

### 1 MATLAB

#### 1.1 Общие сведения

На рисунке 1 изображены кнопки создания нового проекта и окна с выбором блоков/элементов MATLAB (Рисунок 2).

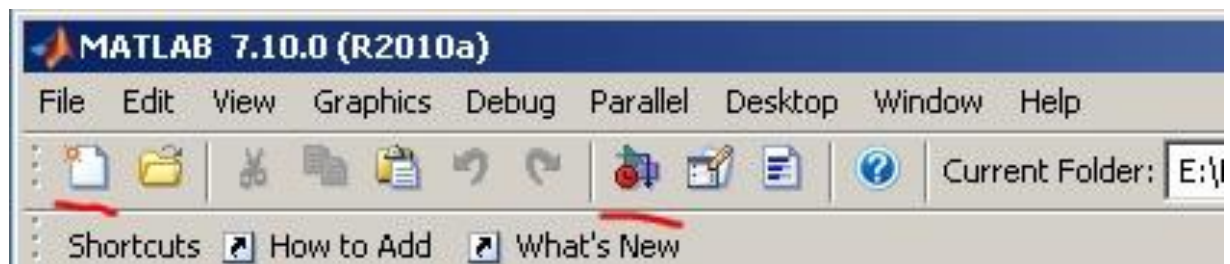


Рисунок 1 – Общие обозначения

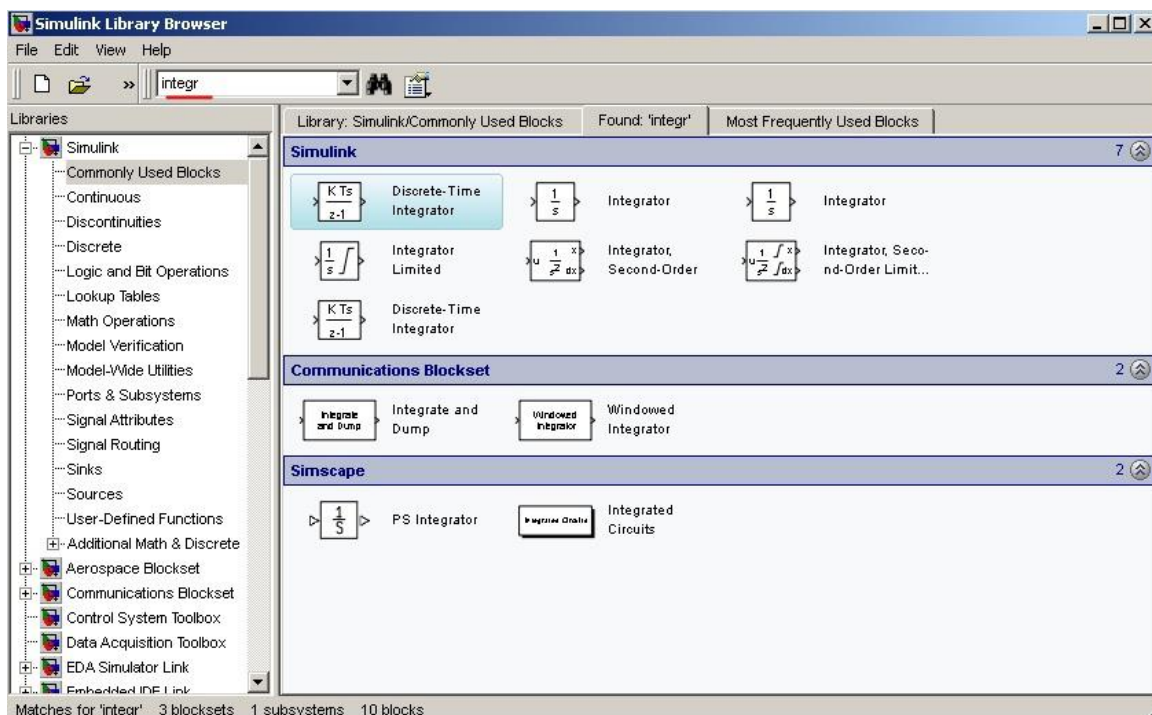


Рисунок 2 – Принцип добавления элементов в схему

## 1.2 Создание схем

На рисунке 3 представлена родительская схема. Она нужна для объявления входных и выходных переменных, внесения их в дочернюю схему и вывода графиков.

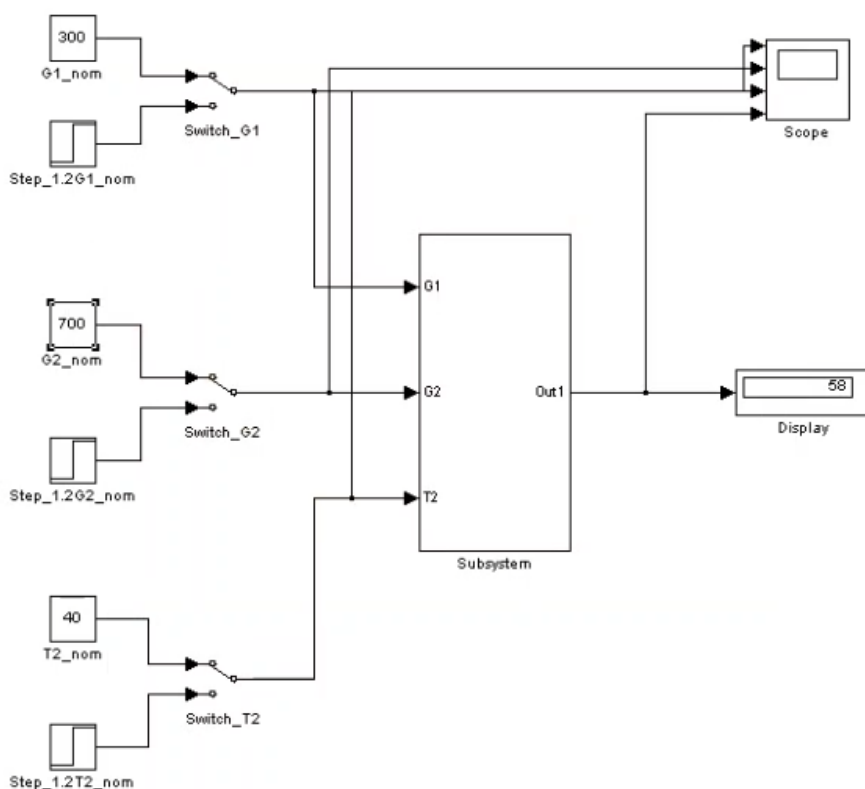


Рисунок 3 – Родительская схема

Также видно, что схема собрана правильно, потому что при вводе константных полученных по варианту значениях схема выдает константное значение равное расчётному.

На рисунке 4 представлено дифференциальное уравнение процесса с помощью логики MATLAB.

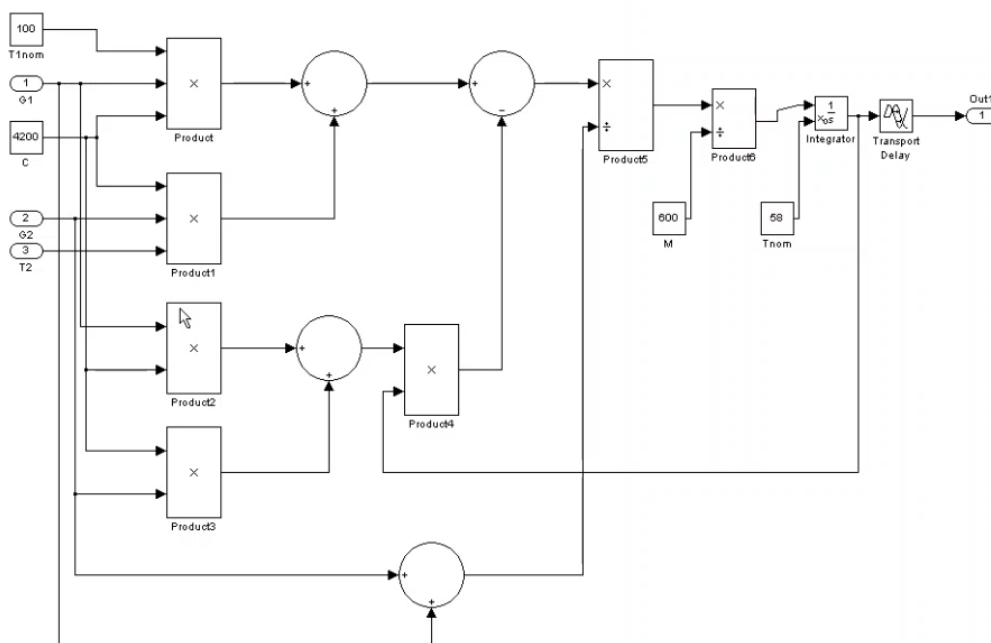


Рисунок 4 – Дочерняя схема

## 2 «KEPServer»

### 2.1 Настройка «KEPServer»

После запуска KEPServerEX и создании нового проекта. Создаем новый канал и задаем его имя (Рисунок 5).

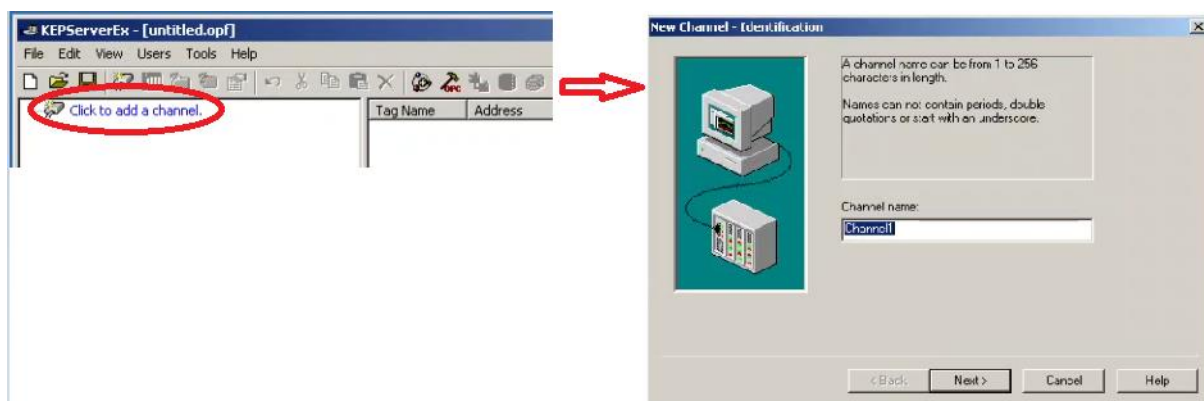


Рисунок 5 - Создание нового канал и задание его имя

Так как в нашей работе мы моделируем процесс на одном устройстве и не подключаемся физически ни к чему, то на данном этапе выбираем Simulator (рисунок 6).



Рисунок 6 – Выбор физического устройства

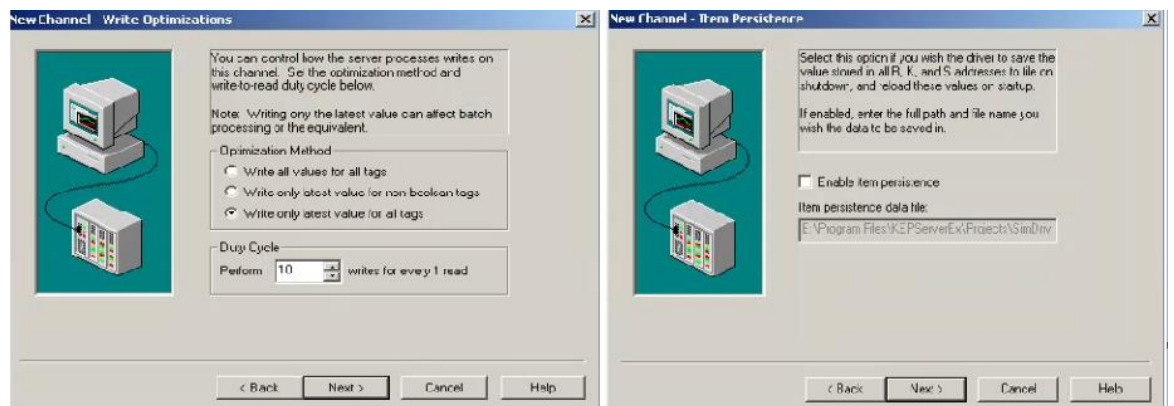


Рисунок 7 – Настройка канала

После настройки канала можно посмотреть его свойства, которые мы сформировали ранее.

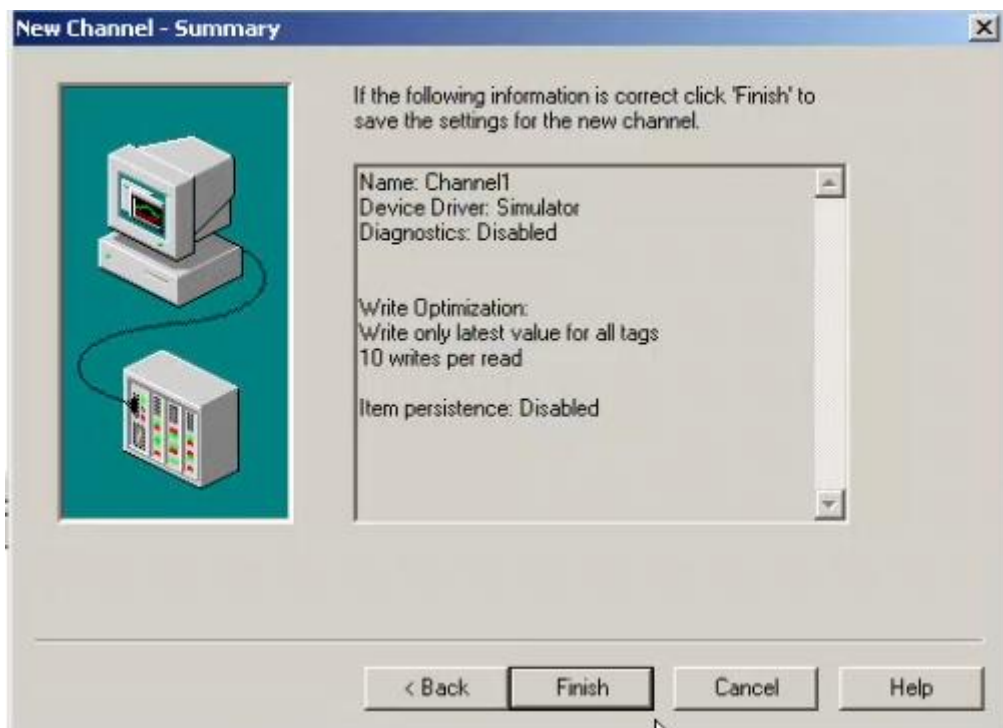


Рисунок 8 – Свойства настроенного канала

Внутри канала создадим новое устройство, с которым будем работать.

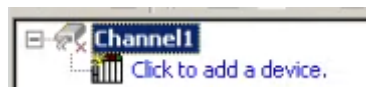


Рисунок 9 – Добавление нового устройства

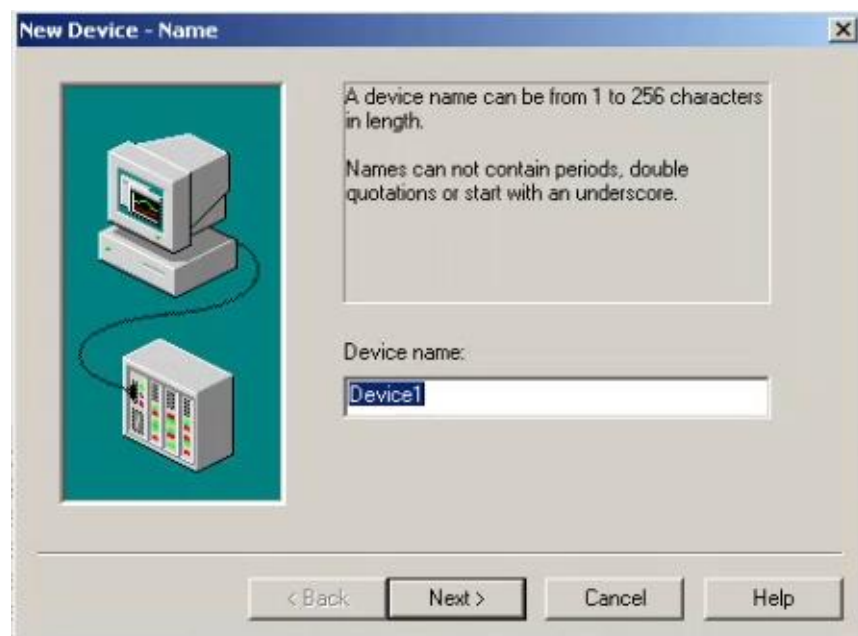


Рисунок 10 – Наименование устройства

Выбираем разрядность устройства, от которого в последующем будет зависеть размеры регистров.



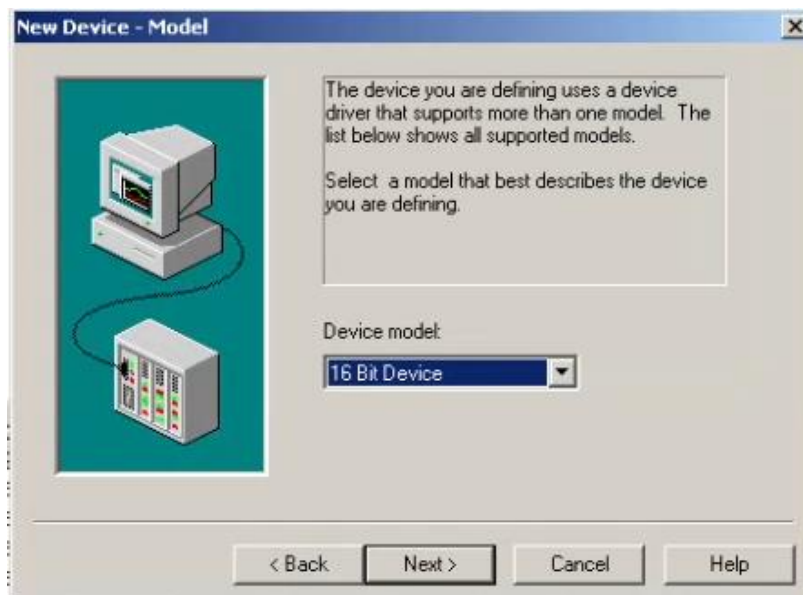


Рисунок 11 – Выбор устройства с 16-битным регистром памяти

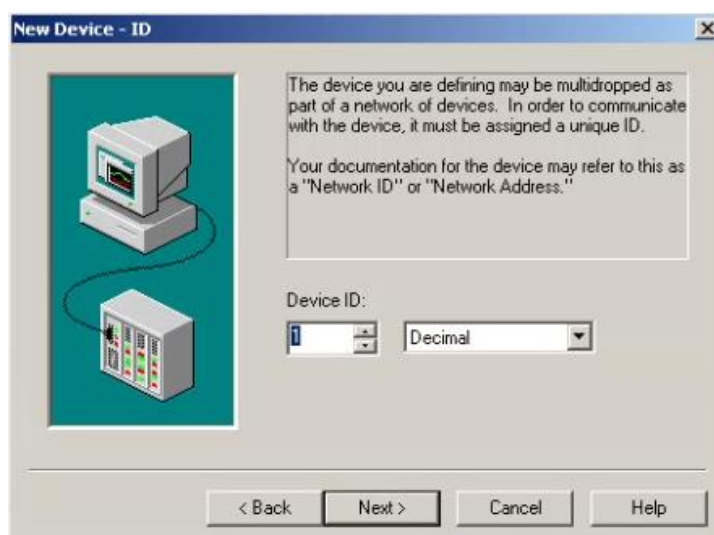


Рисунок 12 – Выбор ID устройства



Рисунок 13 – Свойства созданного устройства



Следующим этапом будет создание тэгов внутри устройства. Необходимо обратить внимание на регистры.

В первую очередь выбирается тип регистра: K, R, S. K и R представляют числовые данные, но R регистр имитирует изменение данных путем увеличения значения при каждом чтении. S для строковых данных переменной длины.

Поэтому выбираем K регистр.

Дальше нужно иметь понимание, сколько занимает битов определённый тип данных:

INTEGER: на каждую переменную выделялся 4 бит (т.к. устройство 16-битное). При объявлении переменных указывается только первый бит из 4-ёх. Пример: K0000-K9999

BOOLEAN: 1 бит. Пример: K0000.00-K9999.15

Tag Name	Address	Data Type	Scan Rate	Scaling	Description
Click to add a static tag. Tags are not required, but are browsable by OPC clients.					

Рисунок 14 – Добавление тэга

<b>Address Descriptions</b> The Simulator supports three types of addresses: R registers, K registers and S registers. The R and K registers are numeric data. S registers are variable length string data locations. The R registers simulate changing data by incrementing on each read when referenced as type Char, Byte, Word, Short, BCD, Long, DWord or LBCD. The K and R registers have an initial value of zero. S registers have an initial value of 'String data Sn' where n is the register number. Address range and data type specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest. The Simulator also supports new simulation functions, which include RAMP, SINE, RANDOM, and USER Defined. A complete description of these functions can be found in the either of the two sections below.			
<b>16 Bit Device Addresses</b> The memory configuration for the 16 Bit Device is simulated as a block of word locations numbered from 0 to 9999. Each word can be addressed as an offset from the start of the block. The default data types for each format are shown in <b>bold</b> .			
<b>Device Type</b>	<b>Range</b>	<b>Data Type</b>	<b>Access</b>
Registers	R0000-R9999 R0000-R9998 R0000-R9996	<b>Word</b> , Short, BCD DWord, Long, LBCD, Float Double	Read/Write
Constants	K0000-K9999 K0000-K9998 K0000-K9996	<b>Word</b> , Short, BCD DWord, Long, LBCD, Float Double	Read/Write
Bits	R0000.00-R9999.15 K0000.00-K9999.15	<b>Boolean</b>	Read/Write
Strings	S000-S999	<b>String</b>	Read/Write
<b>Note 1:</b> All data types (except Boolean and String) support arrays by appending the [r] or [r][c] notation to the address. <b>Note 2:</b> The address specified for a data type must allow for the full size of the data type, i.e., users cannot write past the end of the data range. <b>See Also:</b> <a href="#">Simulation Functions</a> and <a href="#">Data Types Description</a> .			

Рисунок 15 – Информация о регистрах и адресации переменных

При создании тэга мы присваиваем ему имя, адрес, при необходимости, указываем тип данных и действия, которые можем делать с ним, в нашем случае, необходимо читать и записывать значения переменной (Рисунок 16).

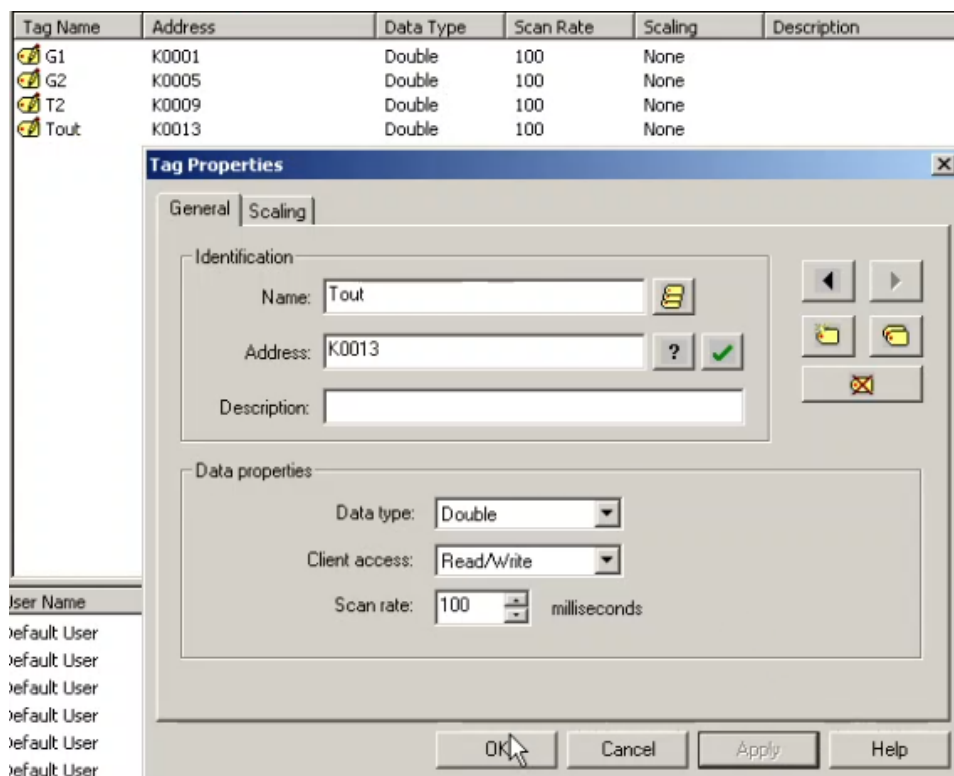


Рисунок 16 – Создание ТЭГов

## 2.2 Изменение родительской схемы MATLAB

Необходимо убрать блоки для ввода и вывода переменных и заменить их блоками для подключения к KEPServerEX (Рисунок17).

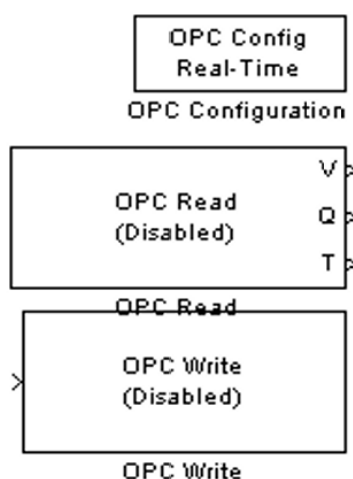


Рисунок 17 - блоки для подключения к KEPServerEX

В добавленных блоках проведены следующие манипуляции (рисунок 18):

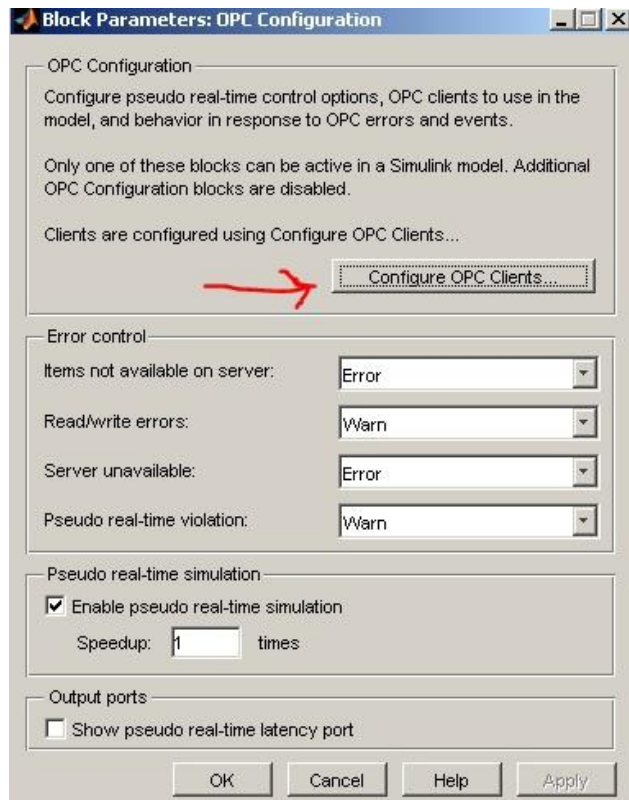


Рисунок 18 – Окно настройки блока конфигурации OPC

Для соединения выбран соответствующий OPC клиент (Рисунок 19).

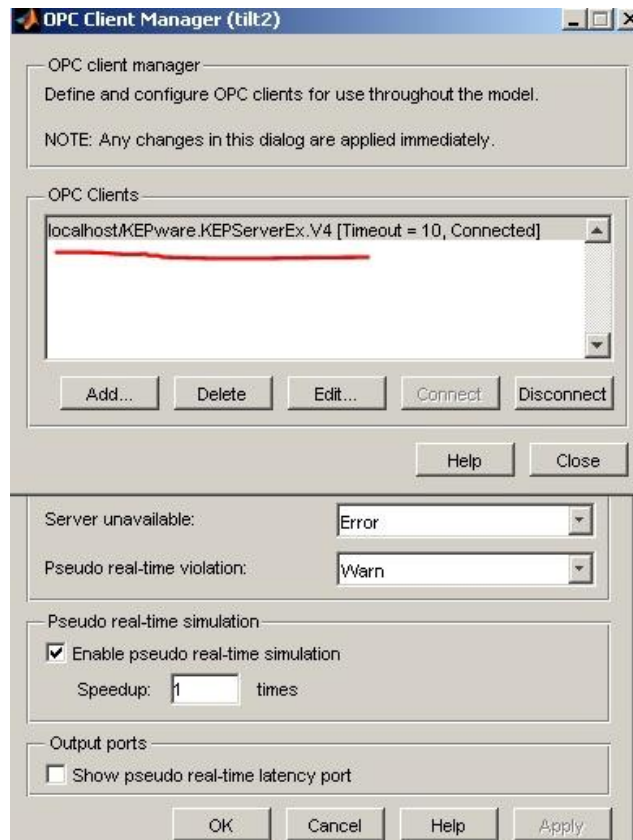


Рисунок 19 – Окно настройки блока конфигурации OPC

Дальше похожим образом настроены блоки “OPC Write” и “OPC Read” (Рисунок 20-21).

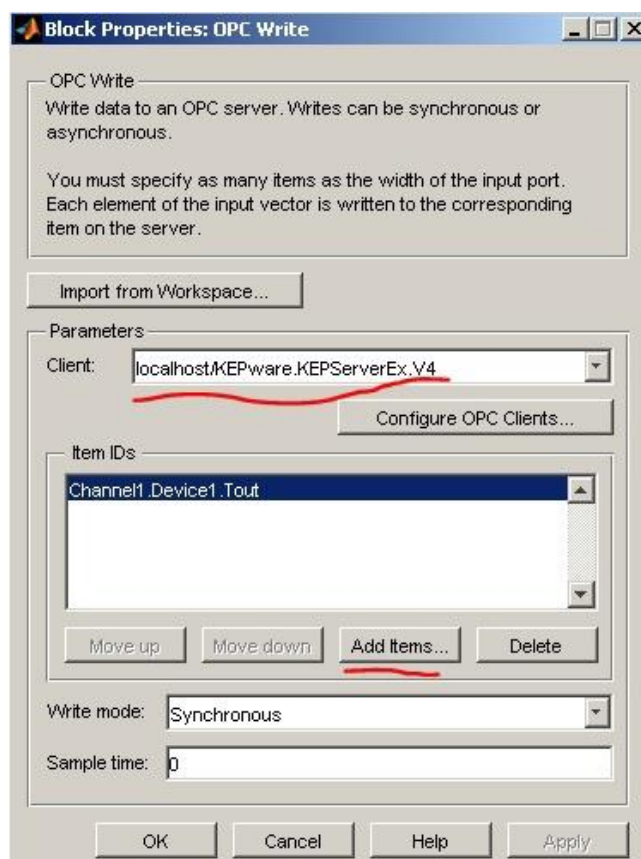


Рисунок 20 – Окно настройки блока записи OPC

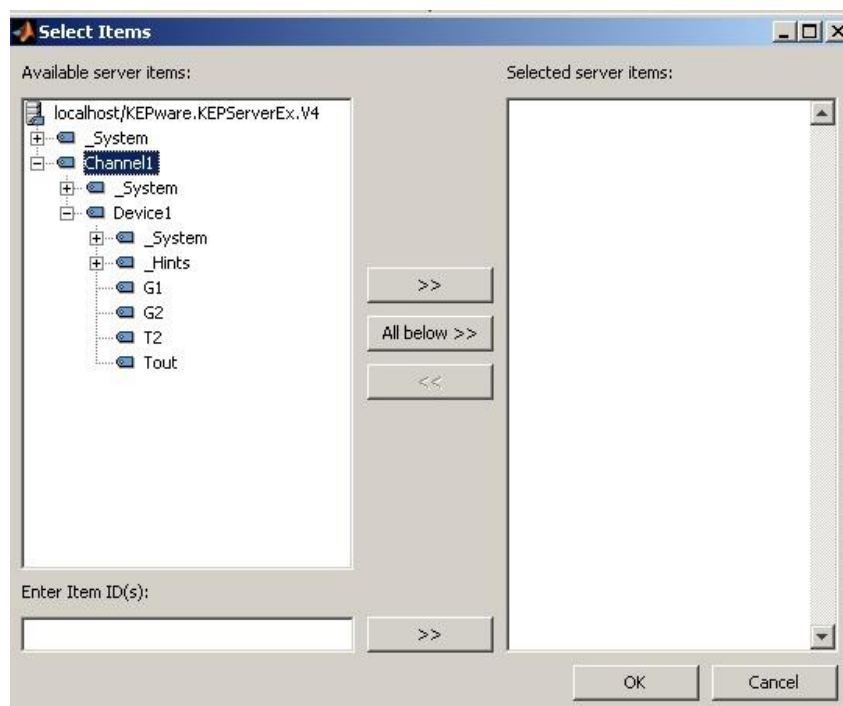


Рисунок 21 – Окно настройки блока записи OPC

Аналогично с блоком чтения (Рисунок 22).

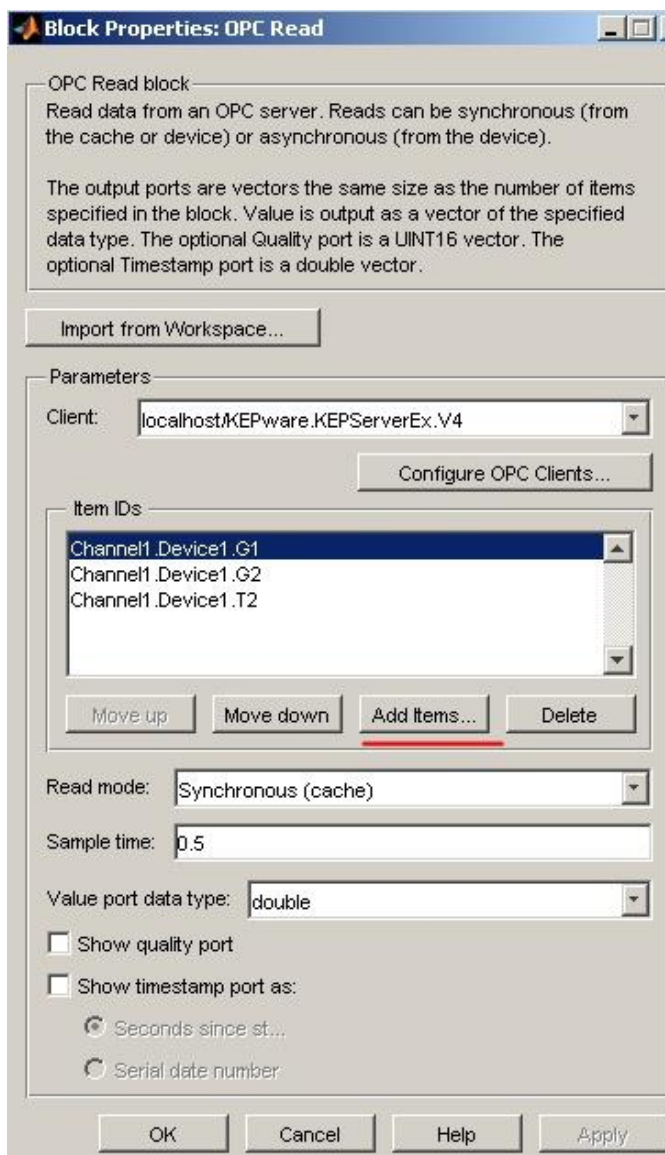


Рисунок 22 – Окно настройки блока чтения OPC

Так же для симуляции непрерывности процесса указано бесконечное время работы логики после чего проведен запуск:



Рисунок 27 – Окно инструментов

Итоговая родительская схема представлена на рисунке 23.

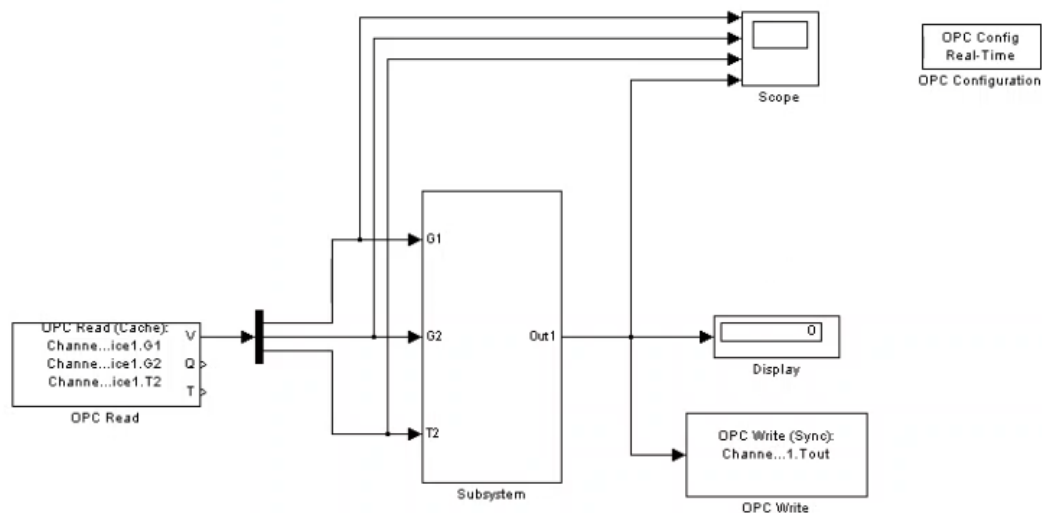


Рисунок 23 – Итоговая родительская схема

Проверка подключения клиента к серверу изображена на рисунке 24.

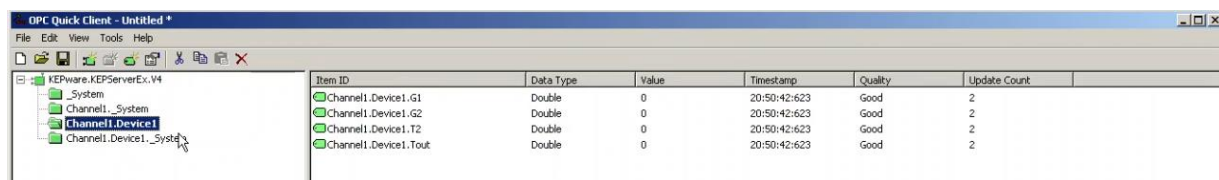


Рисунок 20 – Окно OPC-клиента

### 3 IFix

#### 3.1 «SCU» – системная утилита конфигурирования

«SCU», или «System Configuration Utility» (Утилита конфигурации системы), предназначен для конфигурации среды «IFix». Это инструмент для настройки компонентов системы автоматизации, таких как драйверы ввода-вывода, сетевые настройки, запуск сервисов и лицензирование.

В параметрах настройки «SCU» необходимо указать (Рисунок 25):

- Поддержка «SCADA» → Вкл;
- Определение драйвера «Input» / «Output» → Выбираем стандарт OPC, настройка которого происходила ранее;
- Использования Local OPC Client Driver → Потому что ПЛК (Сервер) находится на том же компьютере, что и «SCADA» и «Database».



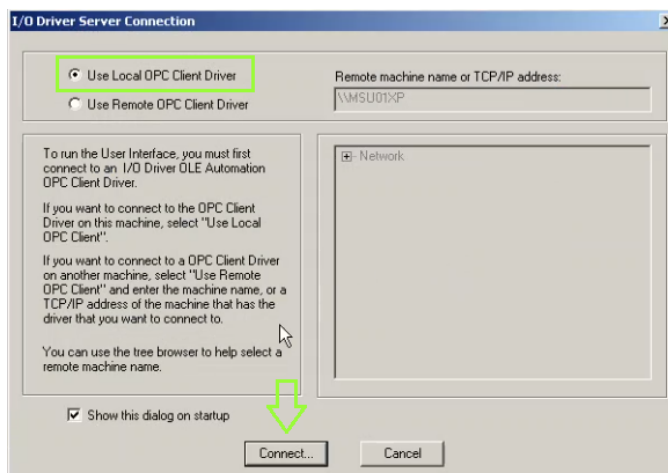
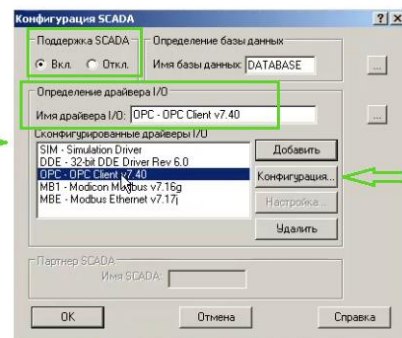
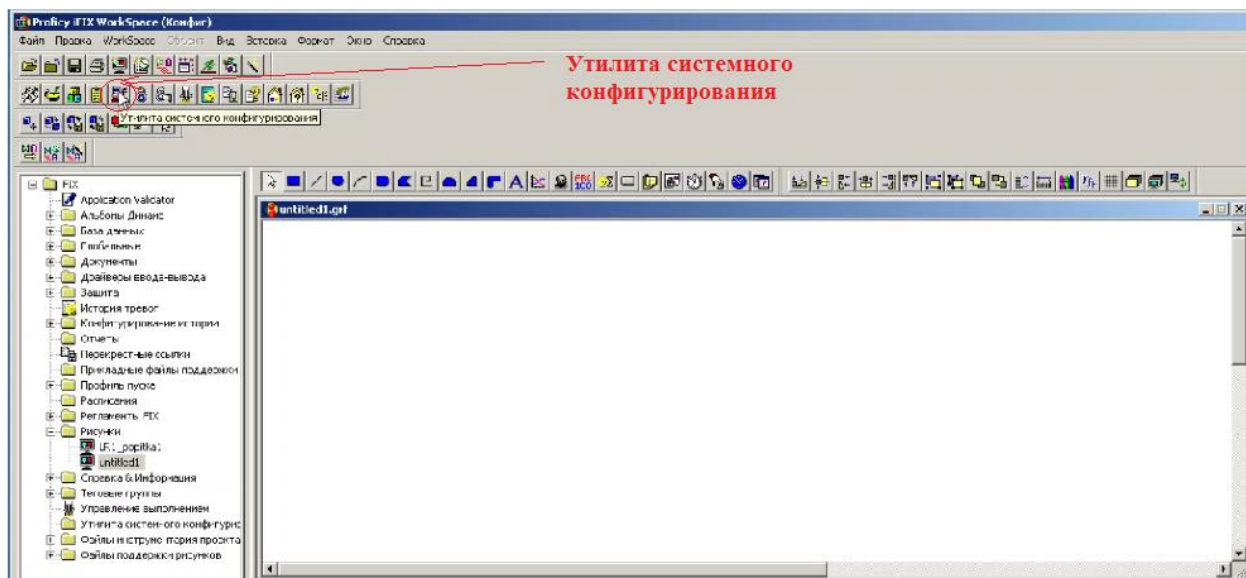


Рисунок 25 – Настройка «SCU»

Также происходит настройка считываемых и записываемых ТЭГов. Для этого необходимо:

- Выбрать OPC Сервера → KERWare.



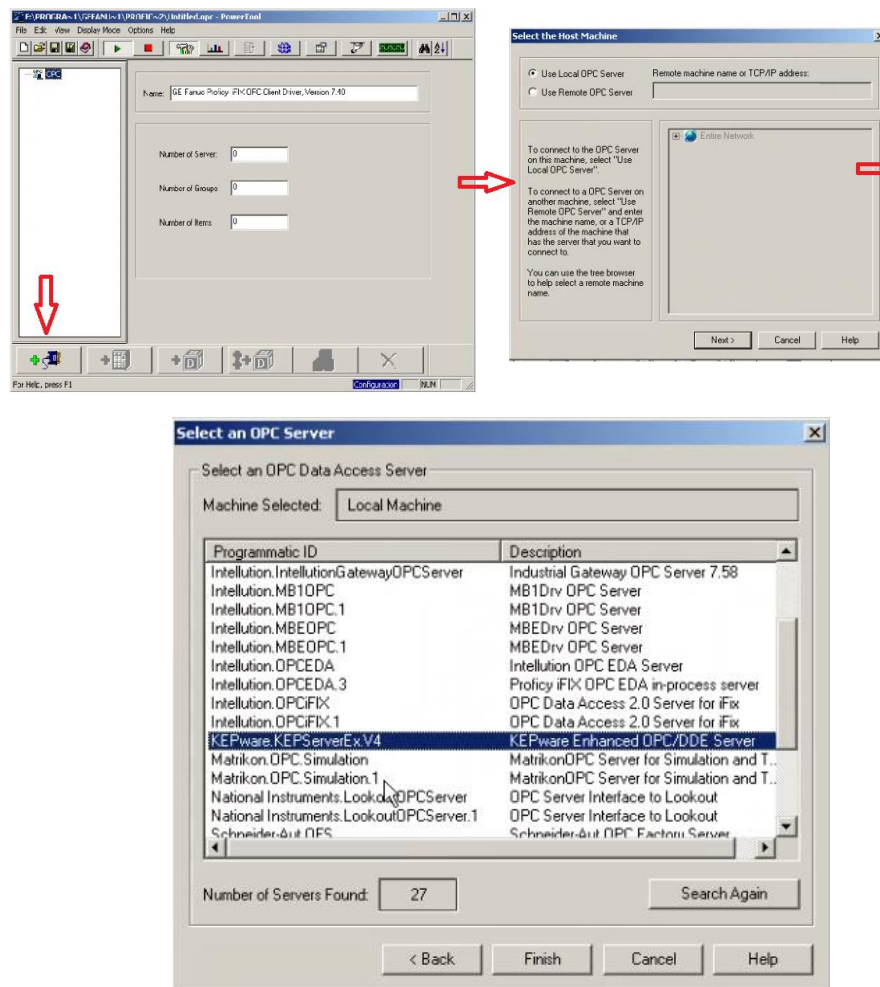


Рисунок 26 – Выбор OPC Сервера

Включение/Активация OPS Сервера происходит путем включения галочки Enable. Создание группу происходит путем нажатия на «Add Group» (Рисунок 27).



Рисунок 27 – Активация сервера и создания группы

Конфигурация параметром группы (Рисунок 28).

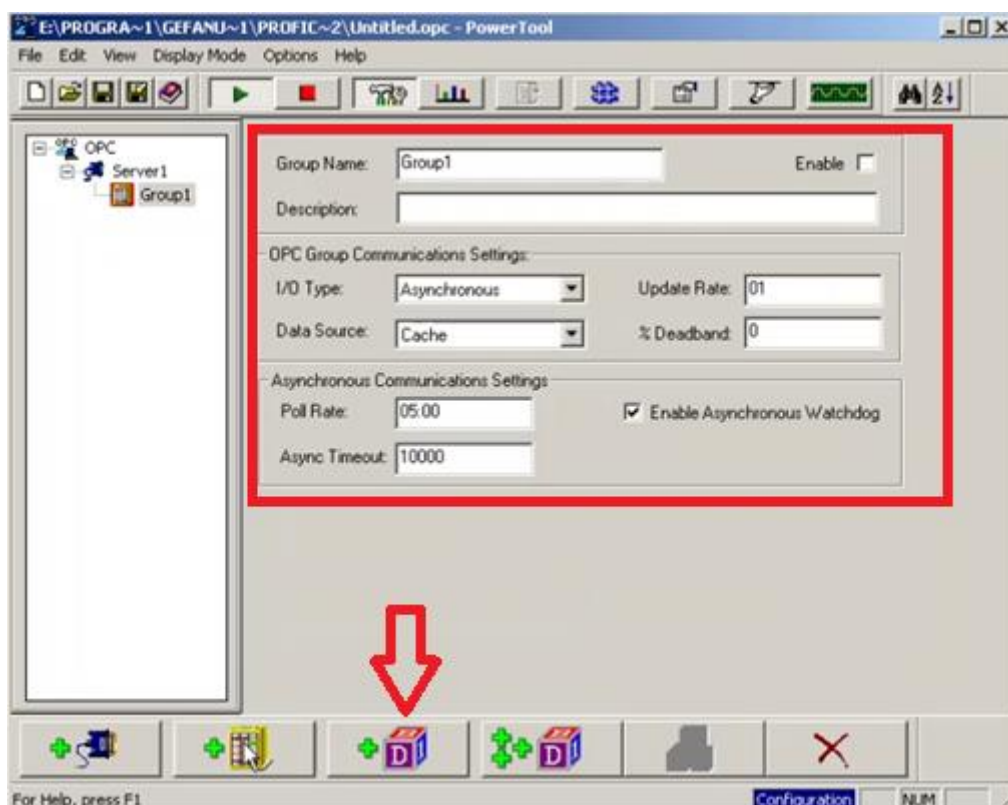


Рисунок 28 – Конфигурация группы

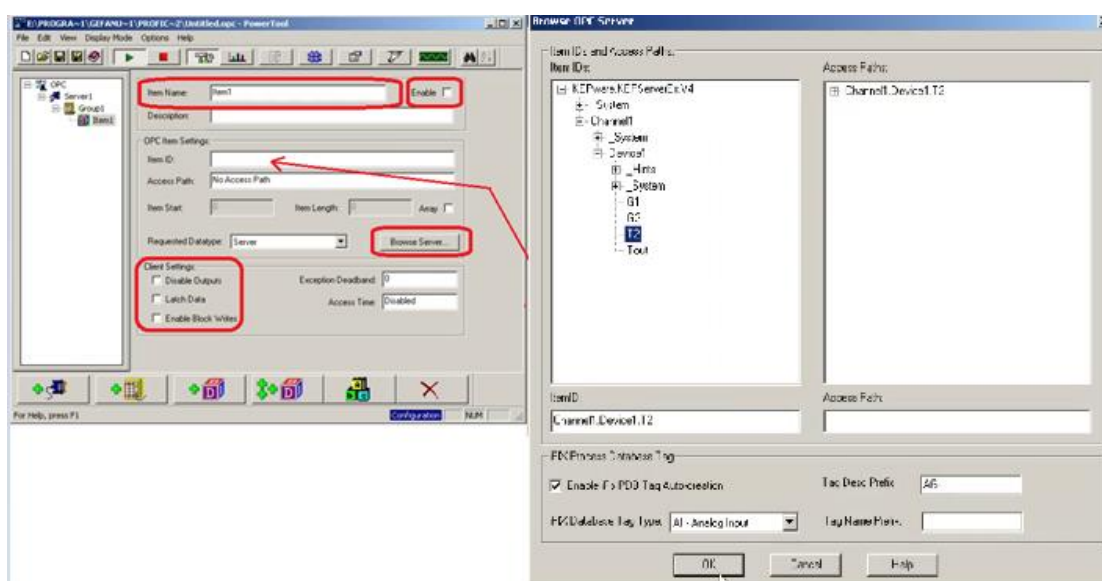


Рисунок 29 - Конфигурация элемента

Галочка **Enable** в данном контексте устанавливается для включения или активации конкретной группы или элемента конфигурации. Рассмотрим, зачем это нужно:

1. Активация группы данных:

Галочка Enable активирует конкретную группу OPC (например, Group1), чтобы данные из этой группы начинали передаваться между OPC-сервером и

клиентом (например, iFIX или Unity Pro). Без этой активации клиент просто игнорирует группу и не выполняет никакого обмена данными.

## 2. Гибкость конфигурации:

Это позволяет гибко управлять данными. Например, когда нужно временно остановить обмен данными для группы тегов, достаточно снять галочку Enable, что проще, чем удалять группу или вносить более сложные изменения в конфигурацию.

## 3. Экономия ресурсов:

Если группа не активирована, то данные для нее не будут запрашиваться и не будут обновляться, что снижает нагрузку на сеть и систему. Это особенно важно для систем, где имеется большое количество данных, но часть из них используется только в определенные моменты времени.

Аналогично происходит создание других необходимых ТЭГов (Рисунок 30).

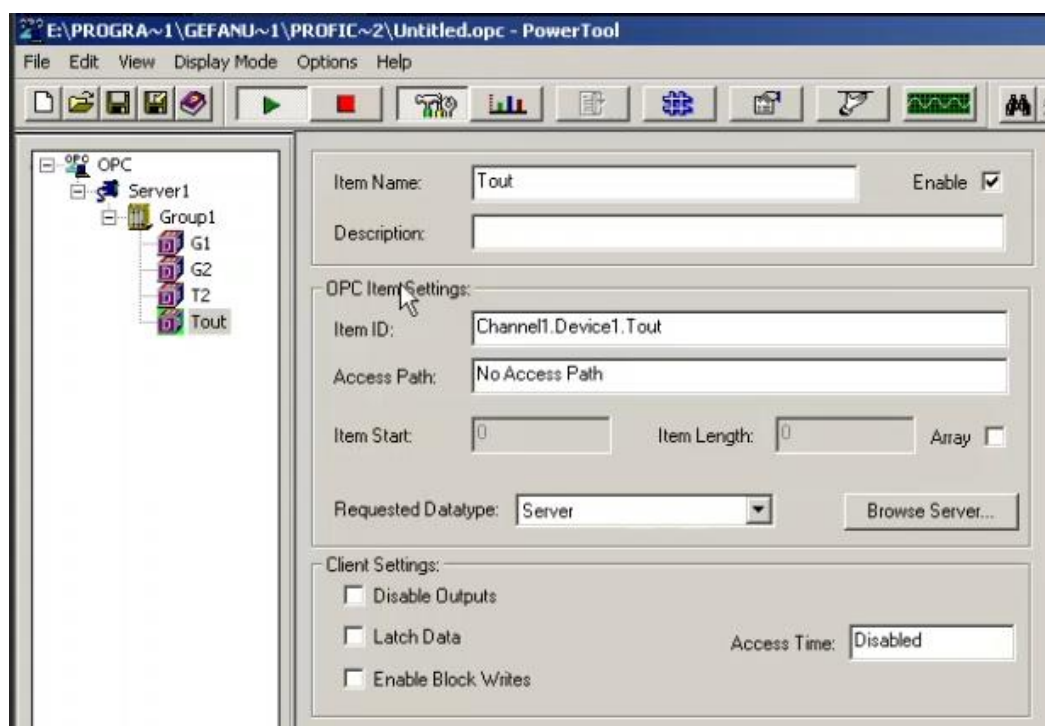


Рисунок 30 – Группа, наполненная необходимыми элементами

## 3.3 Database

Создание базы данных изображено на рисунке 31.

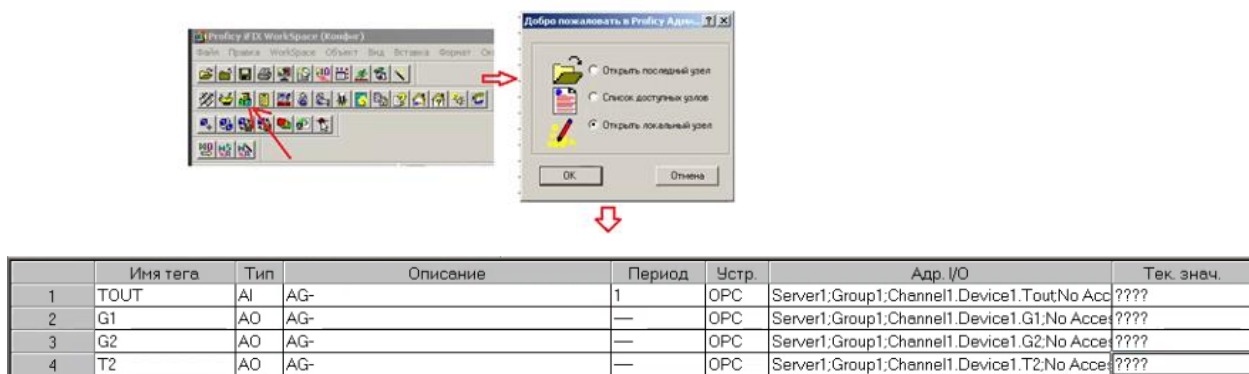


Рисунок 31 – Создание базы данных

Перенос в базу данных параметров, осуществляется в конфигурацию «SCU» (Рисунок 32).

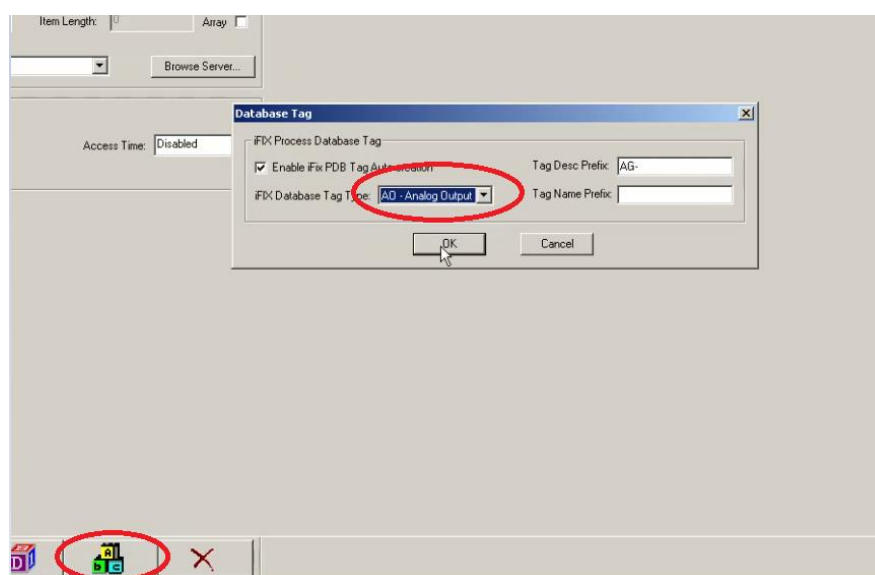


Рисунок 32 – Внесение конкретного параметра в базу данных

## 3.4 SCADA

### 3.4.1 Создание элементов SCADA

- Создание поля ввода и привязка к переменной (Рисунок 33):

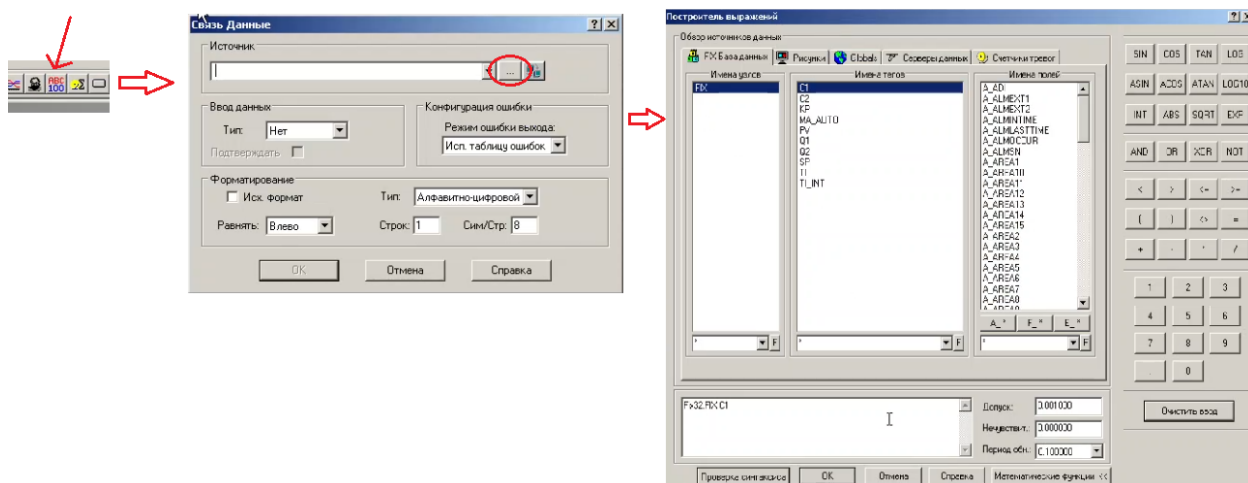


Рисунок 33 - Создание поля ввода и привязка к переменной

Изменения диапазона ввода переменной изображён на рисунке 34.

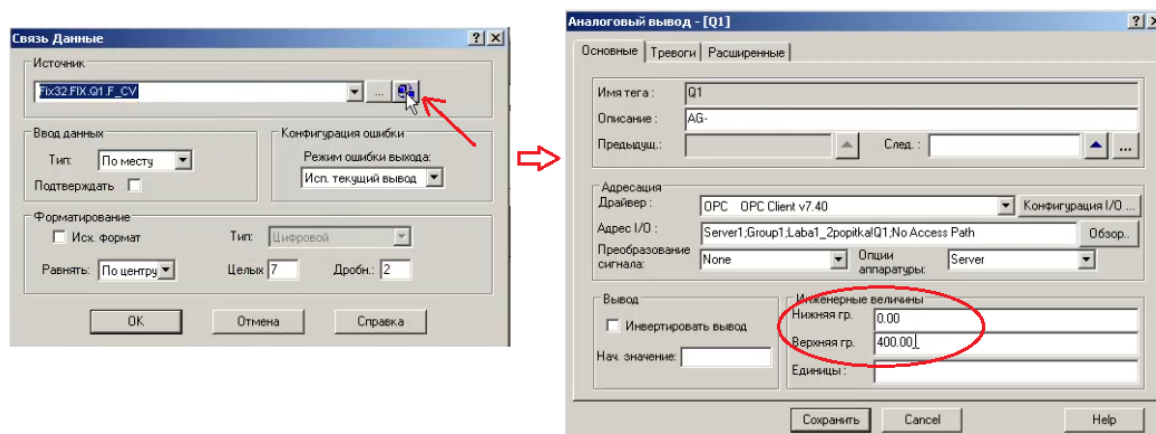


Рисунок 34 - Изменения диапазона ввода переменной

- Написание текста (Рисунок 35):

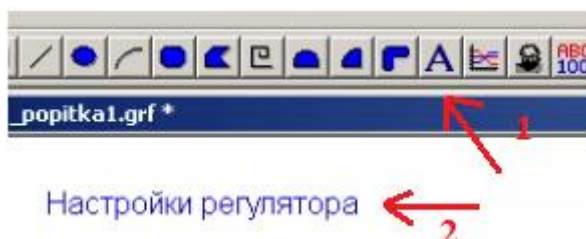
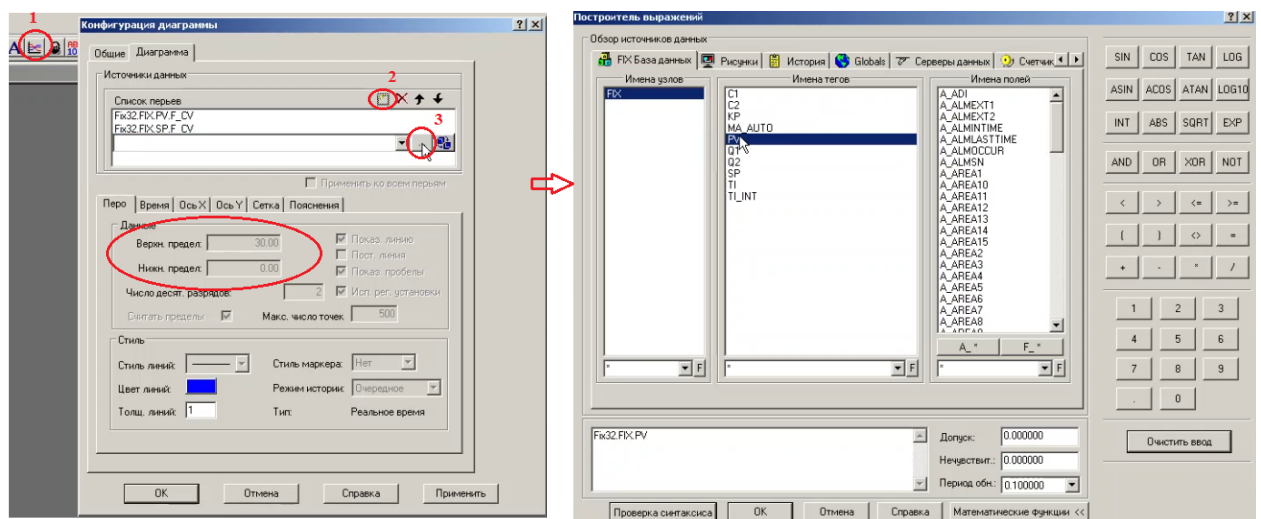
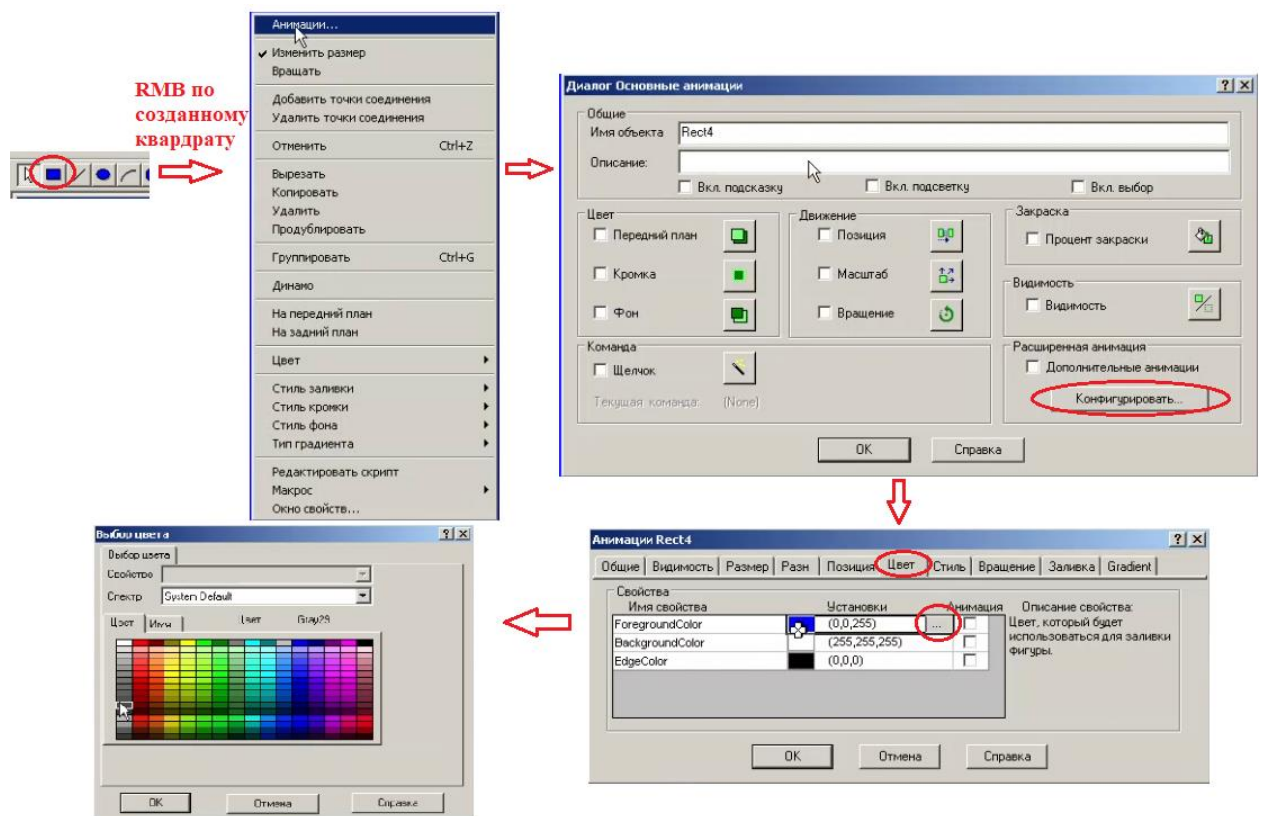


Рисунок 35 – Создание текста

- Создание фона (квадрата) и задание цвета (Рисунок 36):  
Чтобы сделать квадрат фоном необходимо перенести его на задний план.  
Изменение цвета происходит у всех элементов одинаково.





### 3.4.2 Итоговая SCADA

Итоговая SCADA-система изображена на рисунке 41.

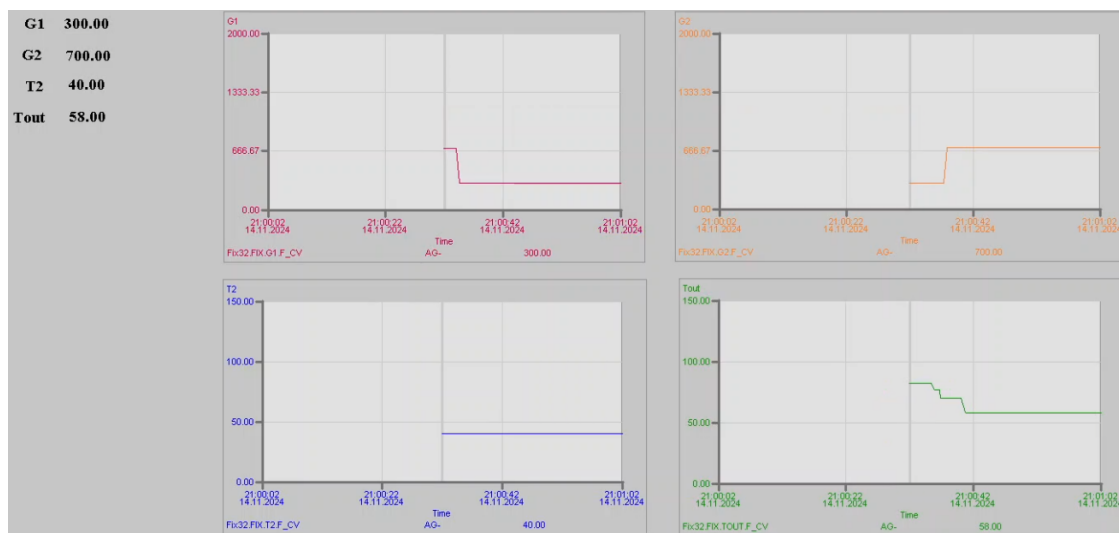


Рисунок 41 – Итоговая SCADA-система

## 4. Добавление двухпозиционных клапанов

### 4.1 KEPServer

В первую очередь добавим 2 переменные, для двух клапанов. Типа «Boolean» достаточно, чтобы хранить 2 состояния клапана. Для булевых переменных используются адреса K0000.01-K0000.15 (Рисунок 42).

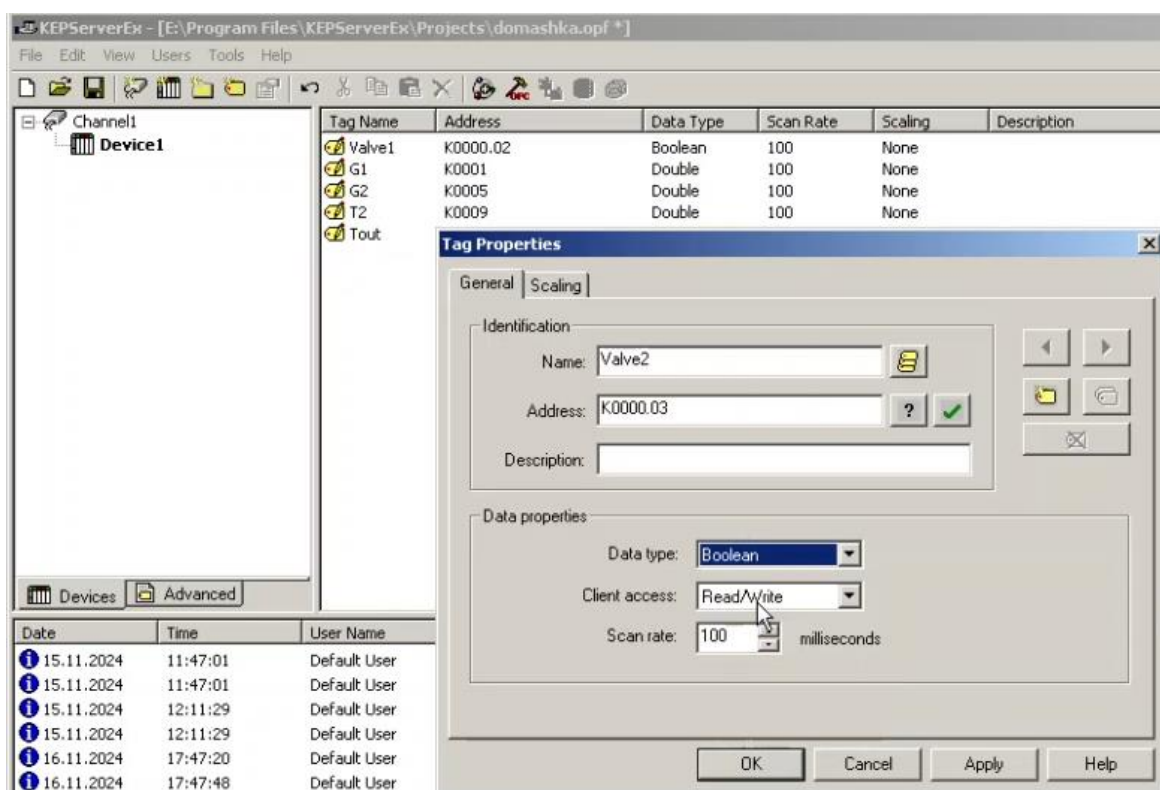


Рисунок 42 – Создание переменных для клапанов



## 4.2 MatLAB

Дальше необходимо изменить логику в среде MatLAB. Для этого используется блок «switch». Его работа максимальна проста:

Один вход блока нужен для переменной, от которой будет зависеть переключение блока «switch». На основе значения данного входа будет проверяться заданное условие. Например, если значение переменной «Valve\_1» будет 0, то на выходе будет дублироваться вход 2, в противном случае вход 1 (Рисунок 43).

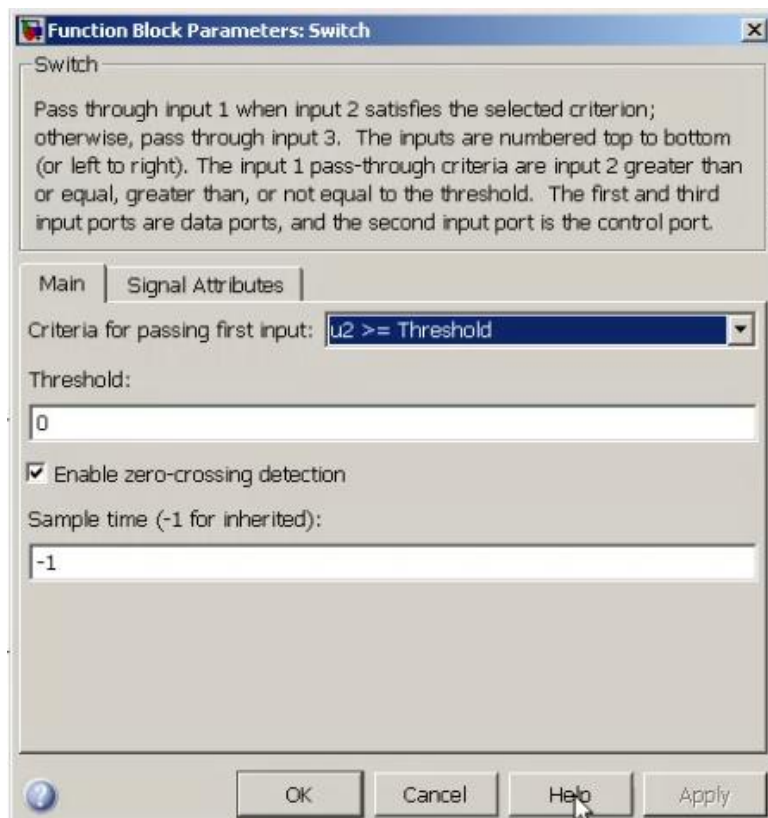


Рисунок 43 – Конфигурация блока «switch»

При перекрытом клапане значение на подсистему соответствующего расхода подается 0,01, потому что при подаче двух нолей (2 клапана закрыты) система ломается → происходит деление на 0 в интеграторе. Поэтому данное число будет характеризовать несовершенство клапана и показывать утечки.

Итоговая схема MatLAB изображена на рисунке 44.

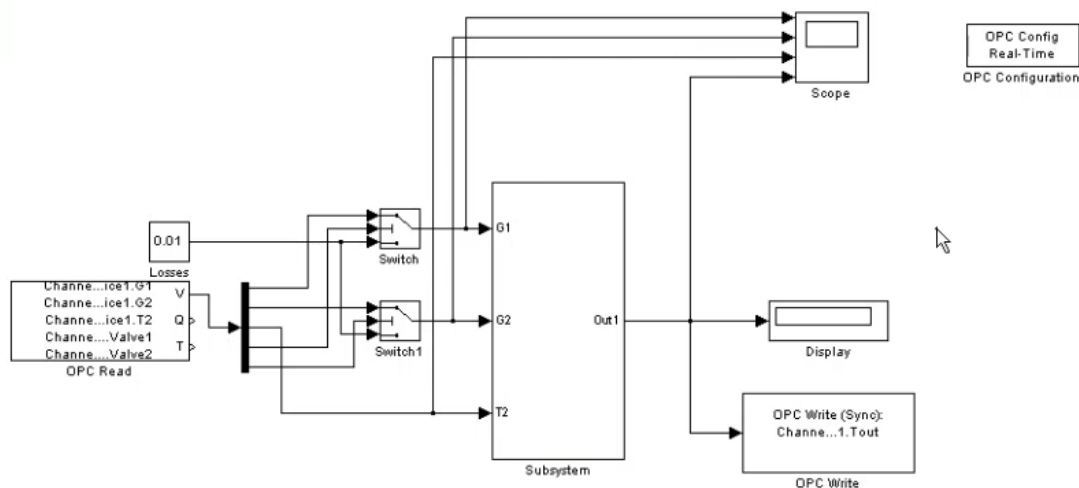


Рисунок 44 – Итоговая родительская схема

### 4.3 IFix

Происходит аналогичная настройка «SCU» и базы данных как на рисунках 25-32.

На рисунке 45 изображен изменённый список ТЭГов.

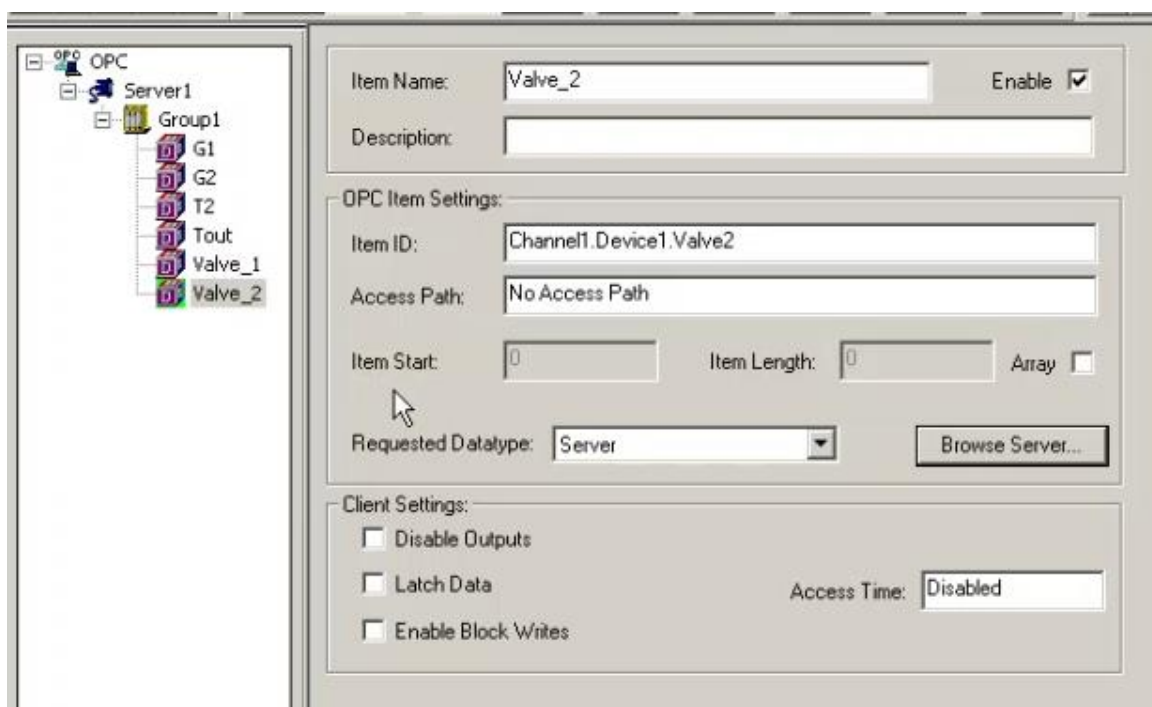


Рисунок 45 - Изменённый список ТЭГов

На рисунке 46 изображена обновленная база данных.

	Имя тега	Тип	Описание	Период	Устр.	Адр. I/O	Тек. знач.
1	TOUT	AI	AG-	1	OPC	Server1:Group1:Channel1.Device1.Tout.No Acc	0.00
2	G1	AO	AG-	—	OPC	Server1:Group1:Channel1.Device1.G1.No Acc	????
3	G2	AO	AG-	—	OPC	Server1:Group1:Channel1.Device1.G2.No Acc	????
4	T2	AO	AG-	—	OPC	Server1:Group1:Channel1.Device1.T2.No Acc	????
5	VALVE_1	AO	AG-	—	OPC	Server1:Group1:Channel1.Device1.Valve1.No A	????
6	VALVE_2	AO	AG-	—	OPC	Server1:Group1:Channel1.Device1.Valve2.No A	????

Рисунок 46 – Обновленная база данных

В дополнении к уже созданной мнемосхеме создается новая динамическая мнемосхема.

- Создание кнопки (Рисунок 47):

Кнопка нужна для переключения двух мнемосхем. Для этого создается 2 файла (Рисунка): один с параметрами регулятора и один с динамической мнемосхемой.

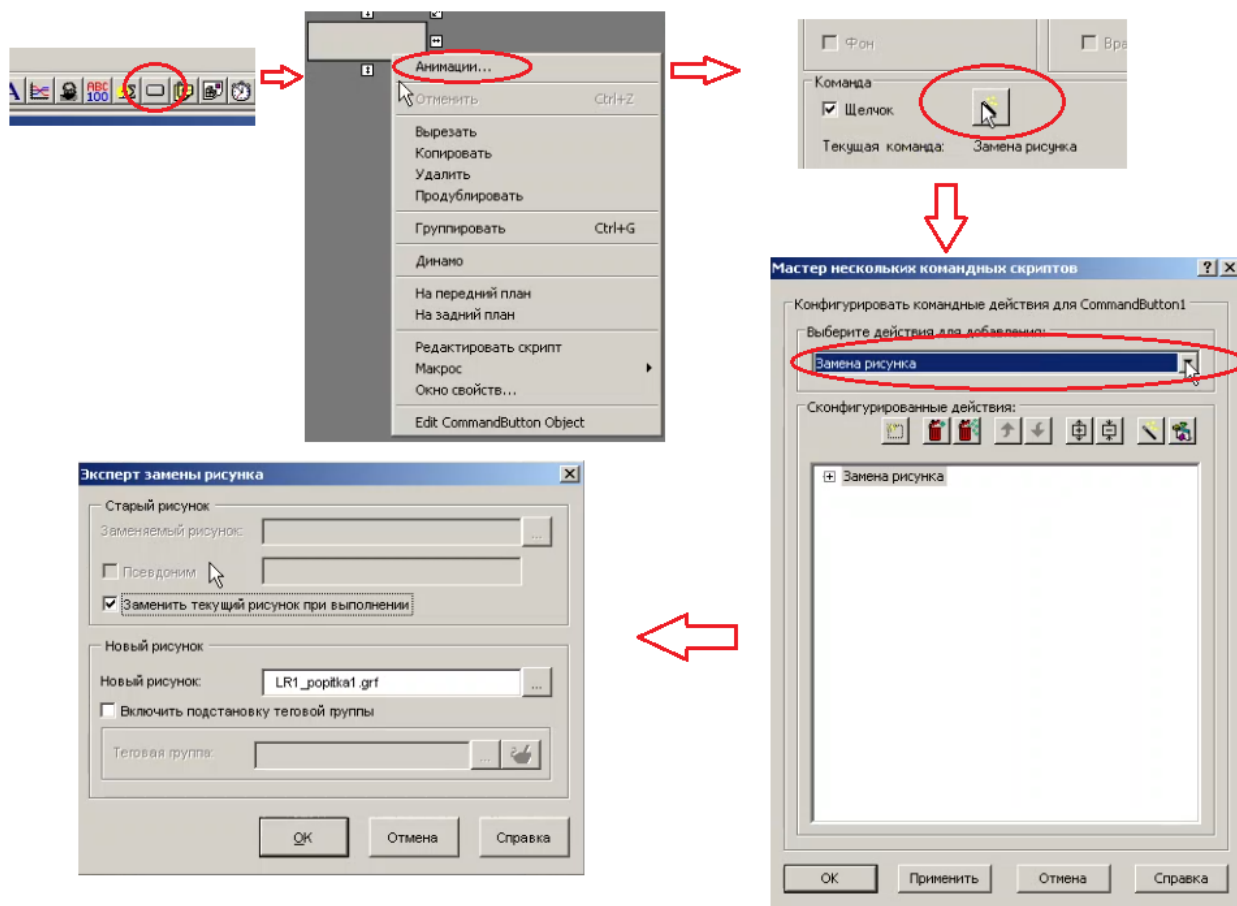


Рисунок 47 – Создание кнопки

- Создание Динамо:

Для динамической мнемосхемы нам нужен резервуар, трубы и клапаны. На рисунке 48 изображено, где можно выбрать необходимые компоненты.

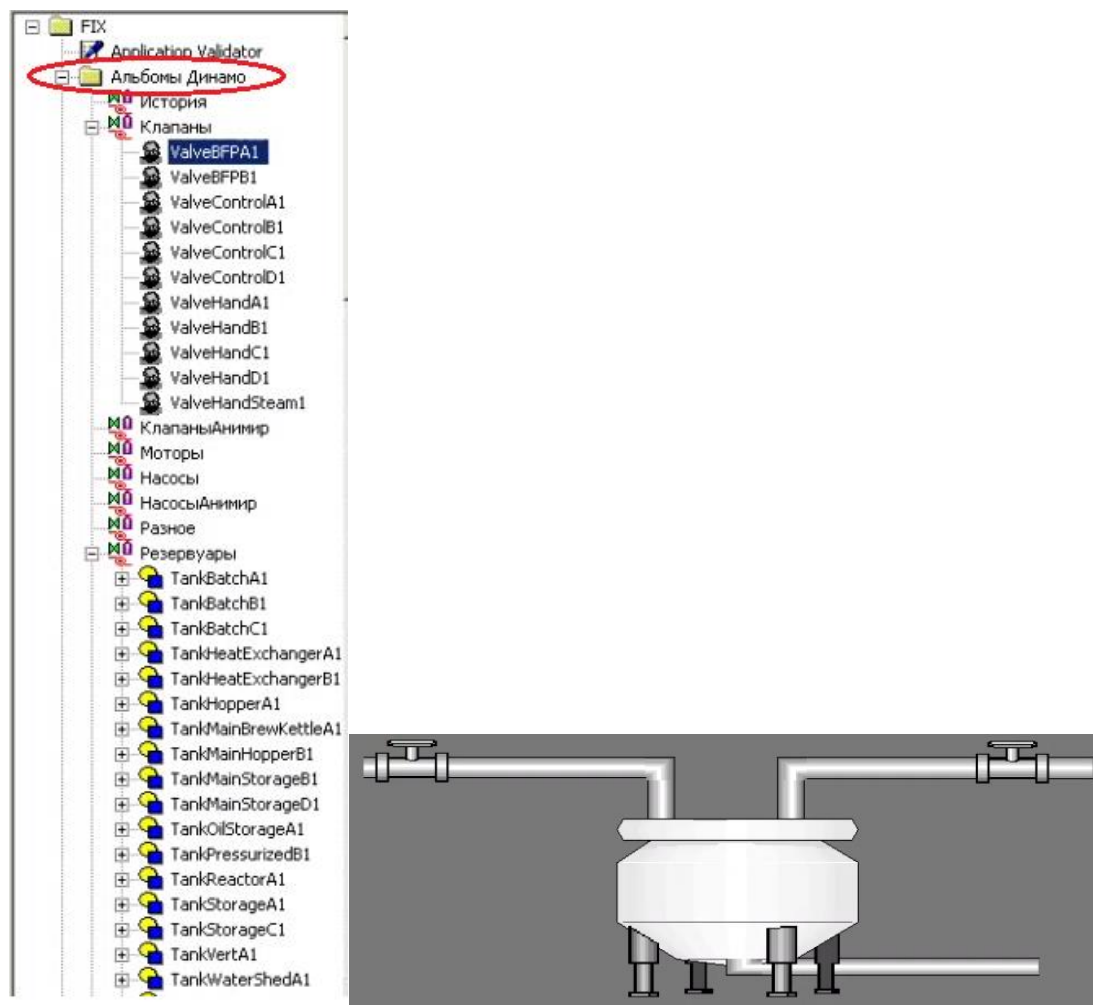


Рисунок 48 – Создание мнемосхемы

На рисунке 49 представлено привязка динамического клапана к созданной переменной.

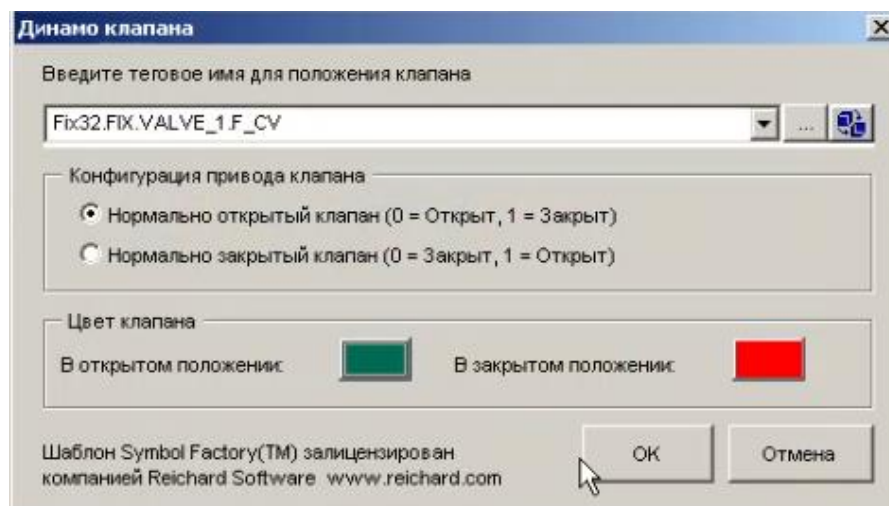


Рисунок 49 – Конфигурация анимированного клапана

Итоговая схема изображена на рисунке 50. На ней видно, что при открытых клапанах у нас показывается расчётное номинальное значение выходной температуры.

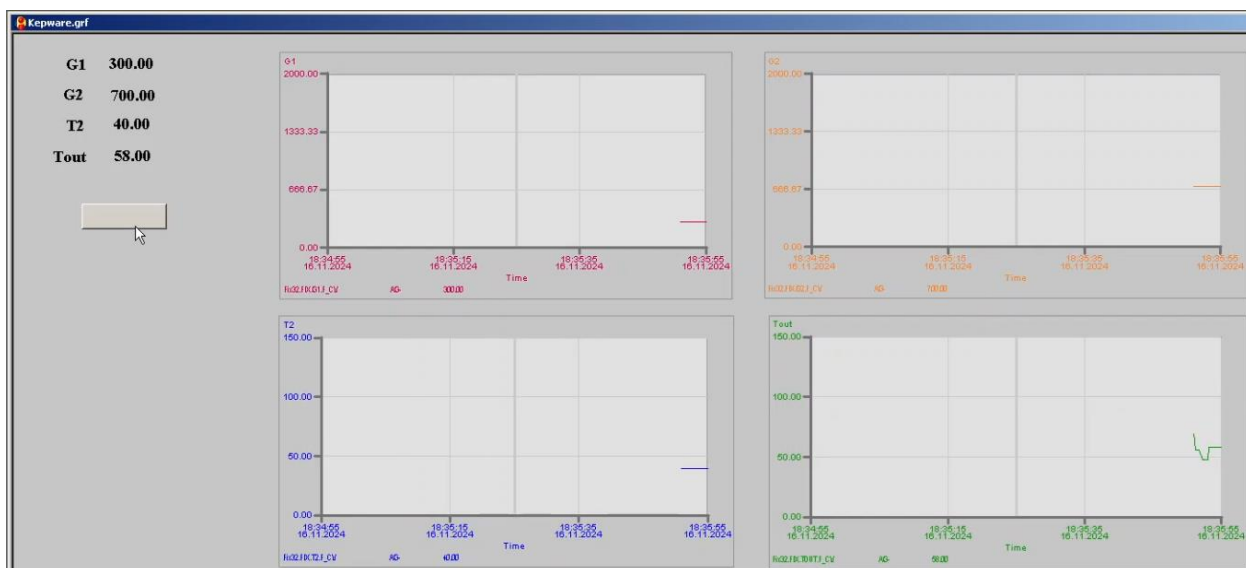
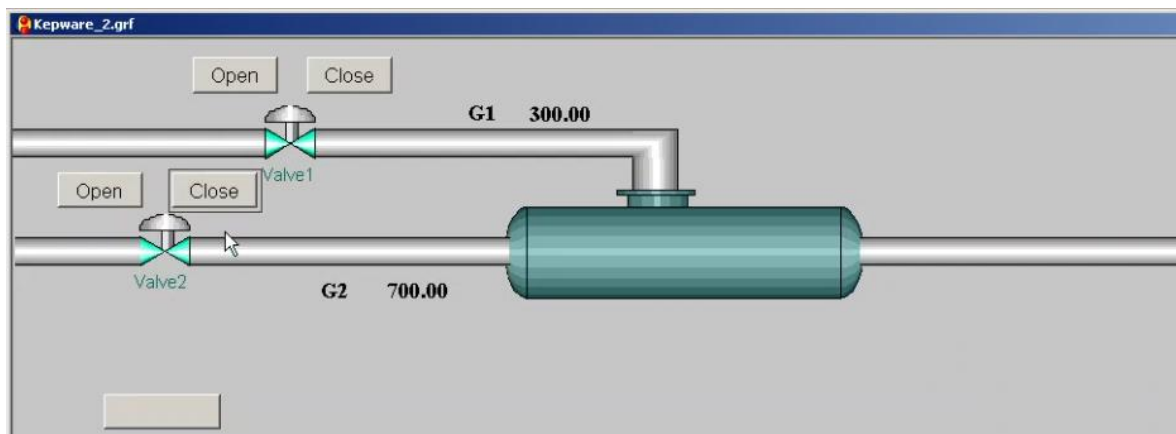


Рисунок 50 – Итоговая динамическая мнемосхема

## Вывод

В ходе лабораторной работы были получены навыки работы с:

- MatLAB – Моделирование системы теплообмена.
- KEPServer – Настройка необходимых компонентов стандарта.
- iFix – Настройка SCU, Database, создание SCADA.