



Начинаем работу

*Создание простого проекта с использованием
протокола OPC client*

Изменения документа

Дата	Версия	Комментарии
25.02.10	1.0	-
-	-	-



**Исключительный дистрибьютор
программных продуктов марки
Winlog Pro на территории РФ
ООО «Крона»**

www.winlogscada.ru
tech@kronaltd.spb.ru

+7 (812) 336-24-51
+7 (812) 297-60-18

Политехническая ул., д.24
Санкт-Петербург
Россия
194021

1. Описание

В этом руководстве мы создадим пример проекта на основе протокола OPC клиента. Этот пример всего лишь первый шаг в направлении к созданию комплексного приложения SCADA системы, но тем не менее он может быть полезен всем кто впервые сталкивается со SCADA WinLog Pro, для того что бы понять как настраивать связь с внешними устройствами. OPC (OLE for Process Control) это промышленный стандарт созданный в сотрудничестве большого количества производителей оборудования и программного обеспечения.

Драйвер Winlog Pro OPC client поддерживает доступ к данным (DA) к OPC серверам DA 1.0 и 2.0. Драйвер может подключаться как к локальным OPC серверам через COM, так и к удаленным OPC серверам (в локальной сети) через DCOM. В случае удаленного сервера вы должны правильно сконфигурировать DCOM. В Windows существует утилита dcomcnfg для конфигурирования DCOM. Перед тем как вы сможете получить доступ к COM компоненту через DCOM, необходимо предоставить права доступа пользователям, которые будут подключаться и запускать компонент.

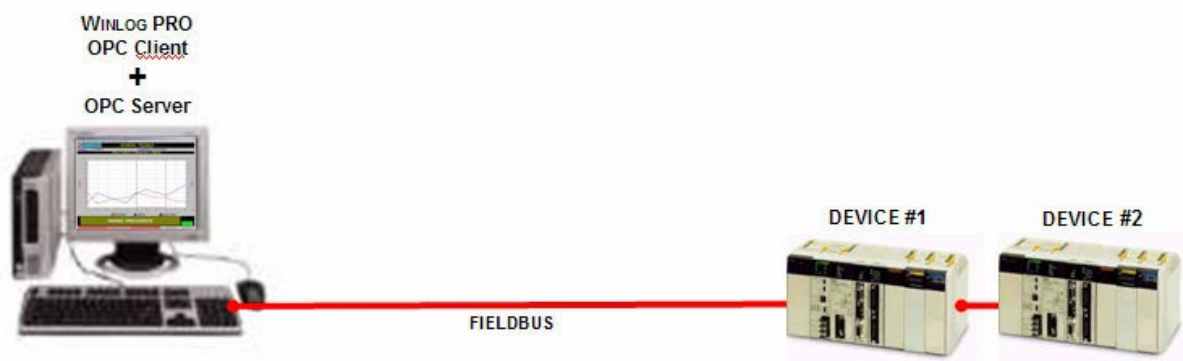


Иллюстрация 1: Локальный OPC

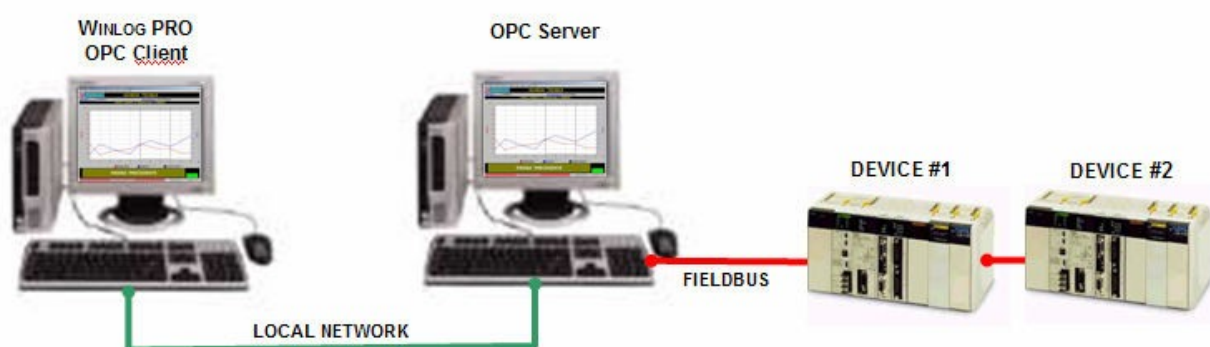


Иллюстрация 2: Удаленный OPC

DA OPC сервер — это организованная структура групп и элементов которые напрямую подключены к устройству или к внутренним переменным PLC, и поэтому отпадает проблема реализации специфического протокола связи с

оборудованием.

OPC клиент всегда связывается с OPC сервером в одном и том же режиме без знания специфики PLC или другого устройства.

Прежде всего вы должны установить OPC сервер на компьютер, и сконфигурировать его (настройки зависят от OPC сервера).

Полное имя переменных (обычно ИмяУстройства + ИмяГруппы + ИмяПеременной) не должно превышать 80 символов.

После этого открываем ProjectManager и создаем новый проект, затем выбираем ProjectManager->Конфигурация->Каналы и выбираем OPC Client протокол, конфигурируем его в соответствии с нашим OPC сервером.

В нашем примере мы будем использовать OPC сервер эмулятор, установленный на тот же компьютер, что и WinLog Pro. Будем использовать DSXP OPC Simulator Ver1.2. Это ПО очень просто может симулировать изменения переменных.

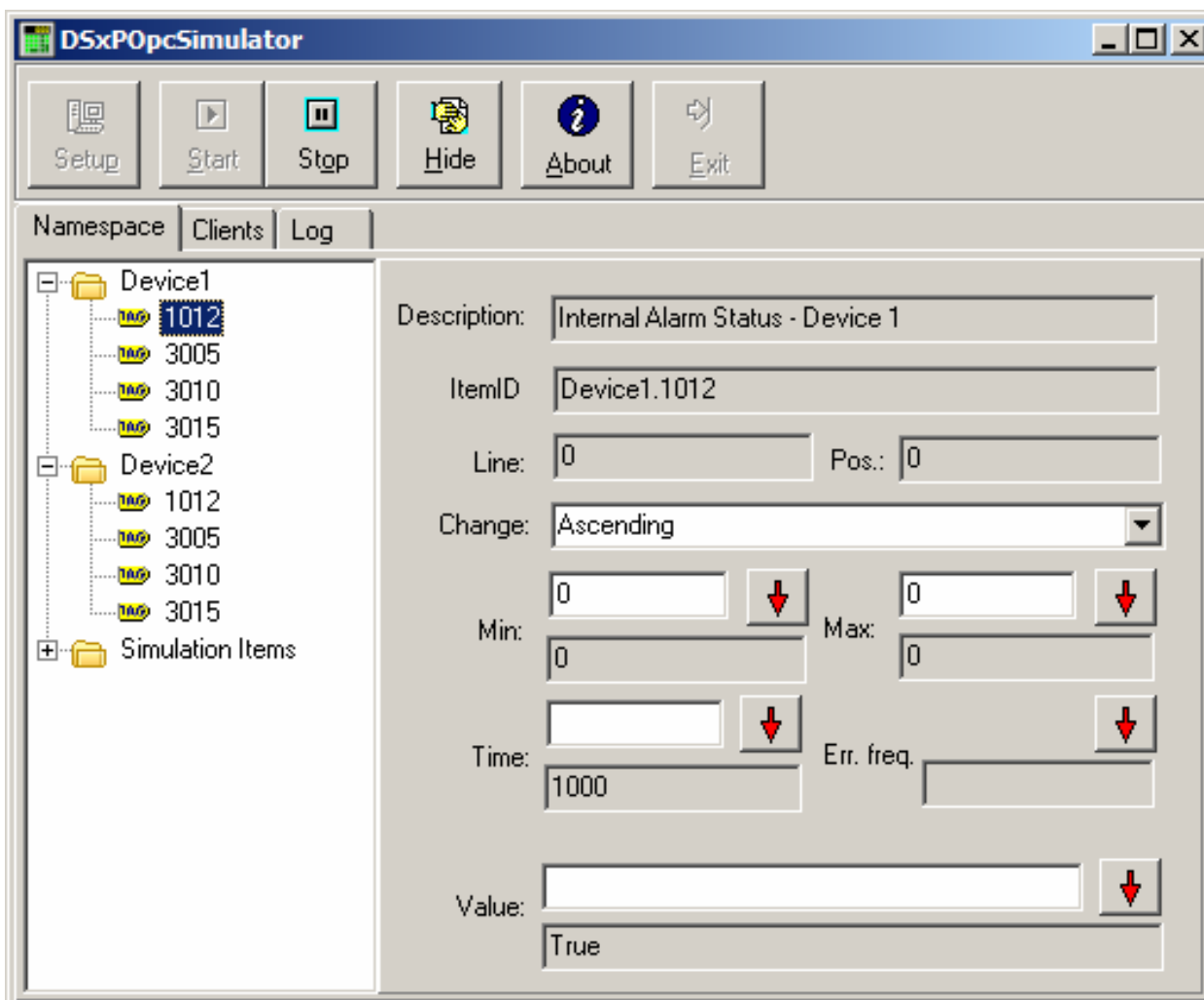


Иллюстрация 3: OPC сервер симулятор

При создании нового проекта вы должны для каждого внешнего устройства указать протокол, адрес устройства и список переменных для чтения и записи.

В нашем примере мы будем соединяться с двумя устройствами, используя OPC Modbus сервер, (Test Device#1 и Test Device#2) с адресами 1

и 2. С каждого устройства мы будем считывать 3 численных переменных (Temp, Sp и Out) и одну дискретную (Alarm).

Мы настроим наш OPC сервер симулятор для предоставления этих переменных якобы по Modbus протоколу от двухразных устройств. Вы можете использовать любые другие OPC серверы с любыми внутренними протоколами.

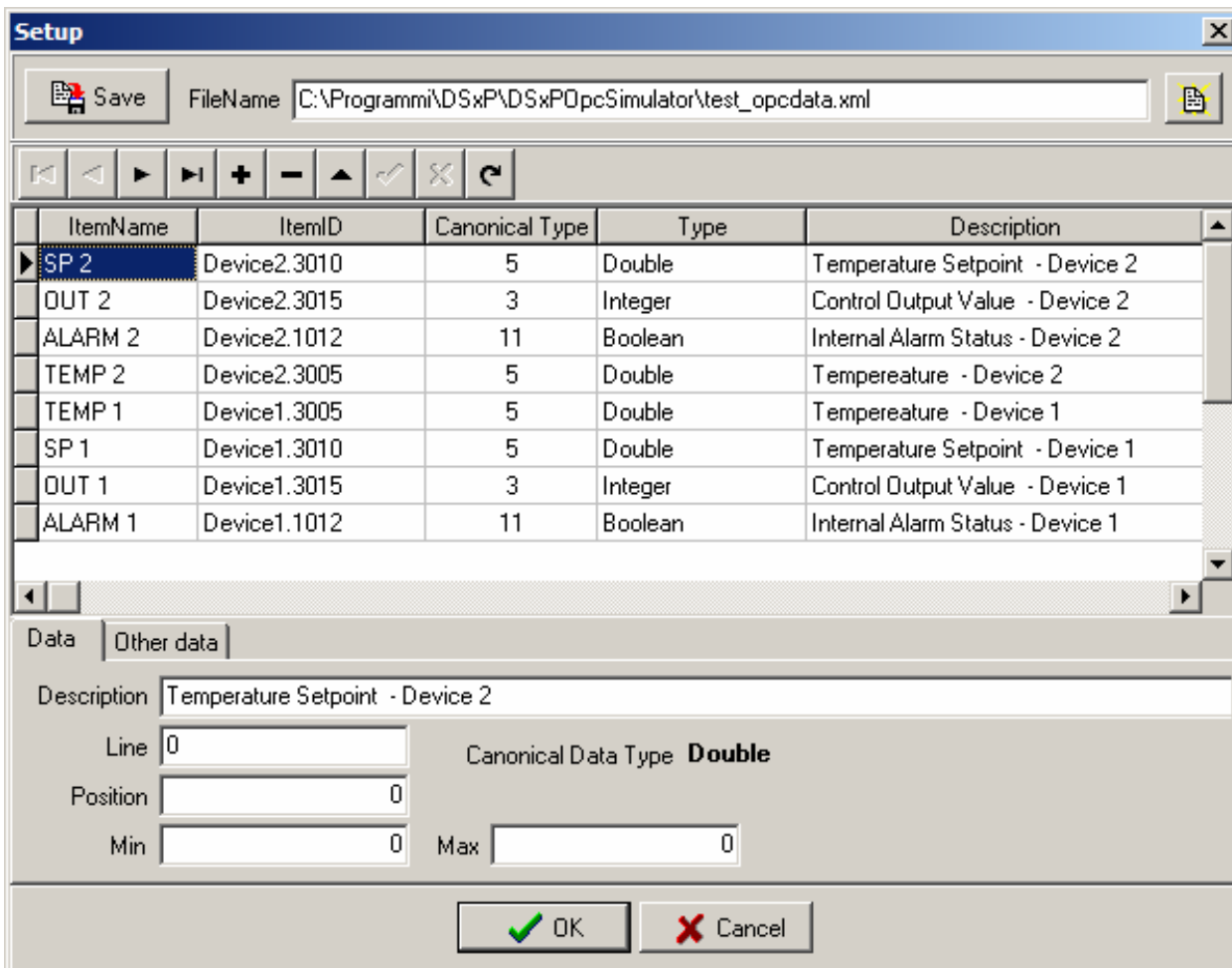


Иллюстрация 4: Настройка симулятора OPC сервера

2. Создание проекта

Для создания нового проекта необходимо использовать Project Manager. Project Manager запускается выбором его иконки в меню программ в меню Пуск. Далее необходимо выбрать Проект-Новый и ввести название проекта (например Test).

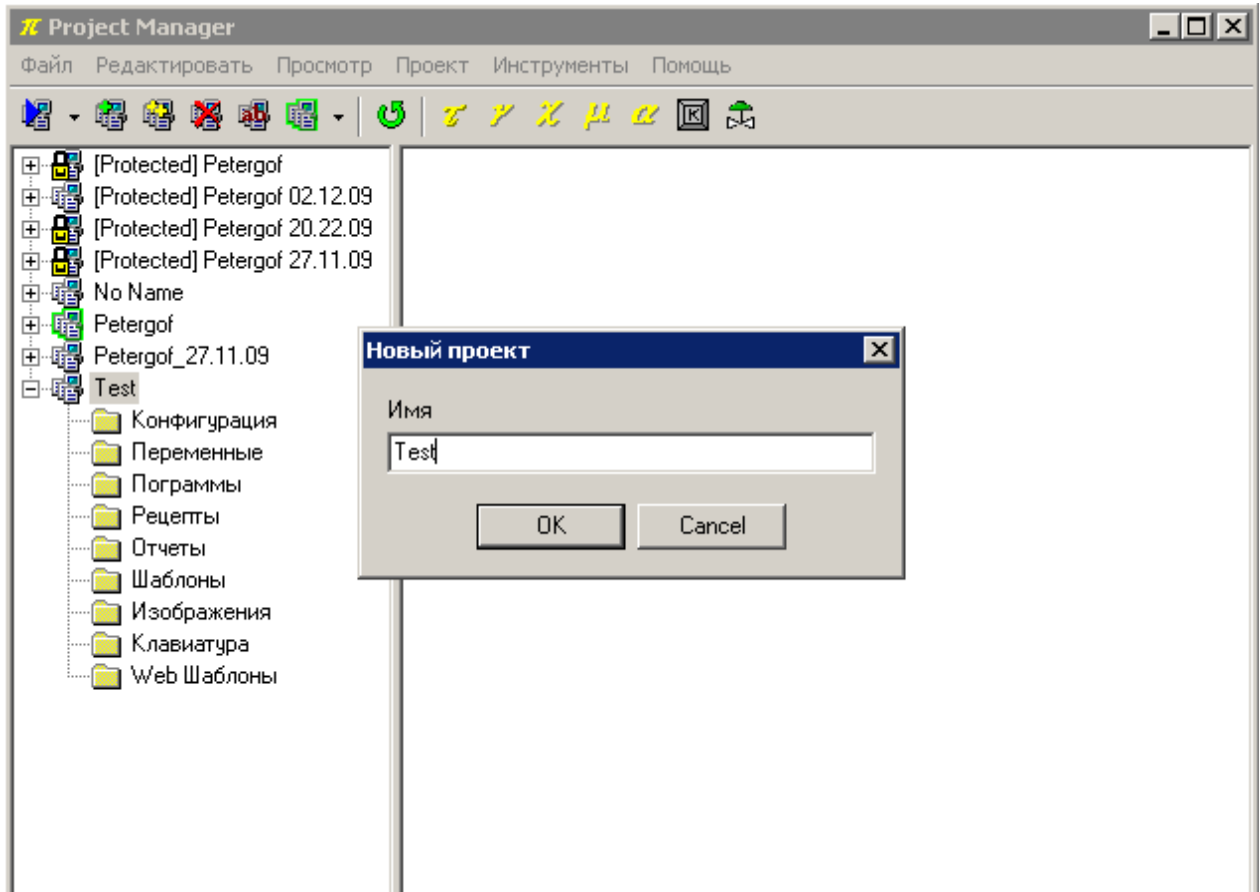


Иллюстрация 5: Создание нового проекта

3. Настройка коммуникационного канала

В разделе Конфигурация выберите раздел каналы. Выберите каналу №1 протокол OPC Client.

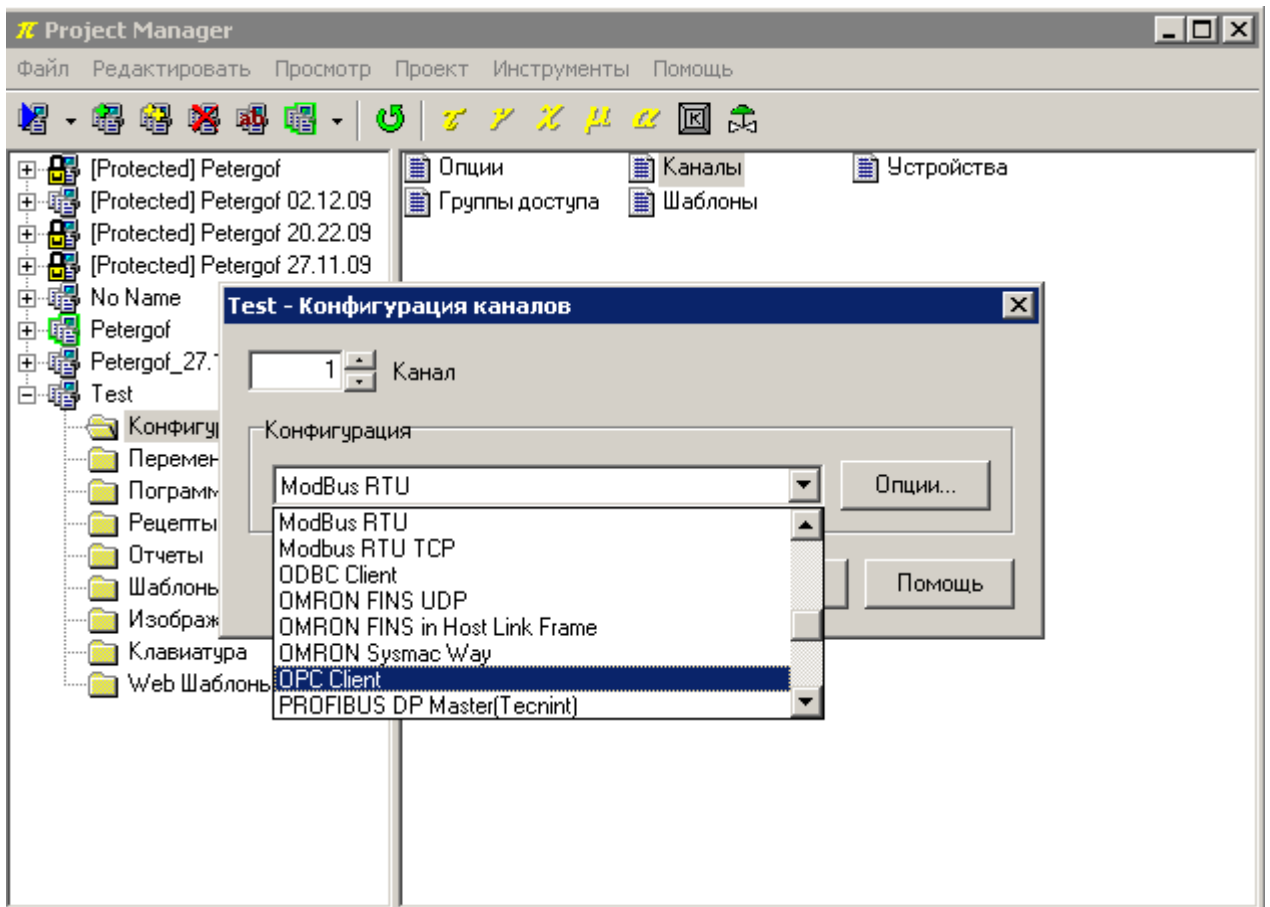


Иллюстрация 6: Выбор протокола

Нажмите кнопку Опции...

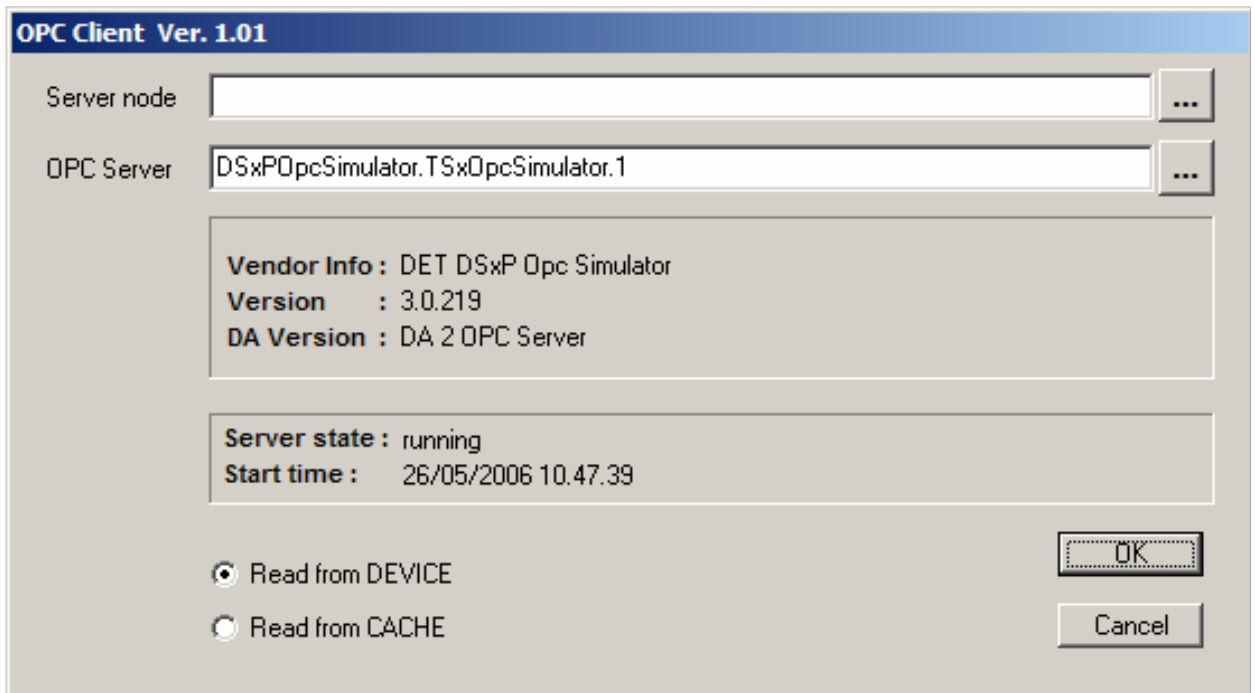


Иллюстрация 7: Опции протокола

Выберите Server node в соответствии с тем, куда инсталлирован OPC сервер (например локальный компьютер) и выберите OPC сервер, доступный на этом компьютере.

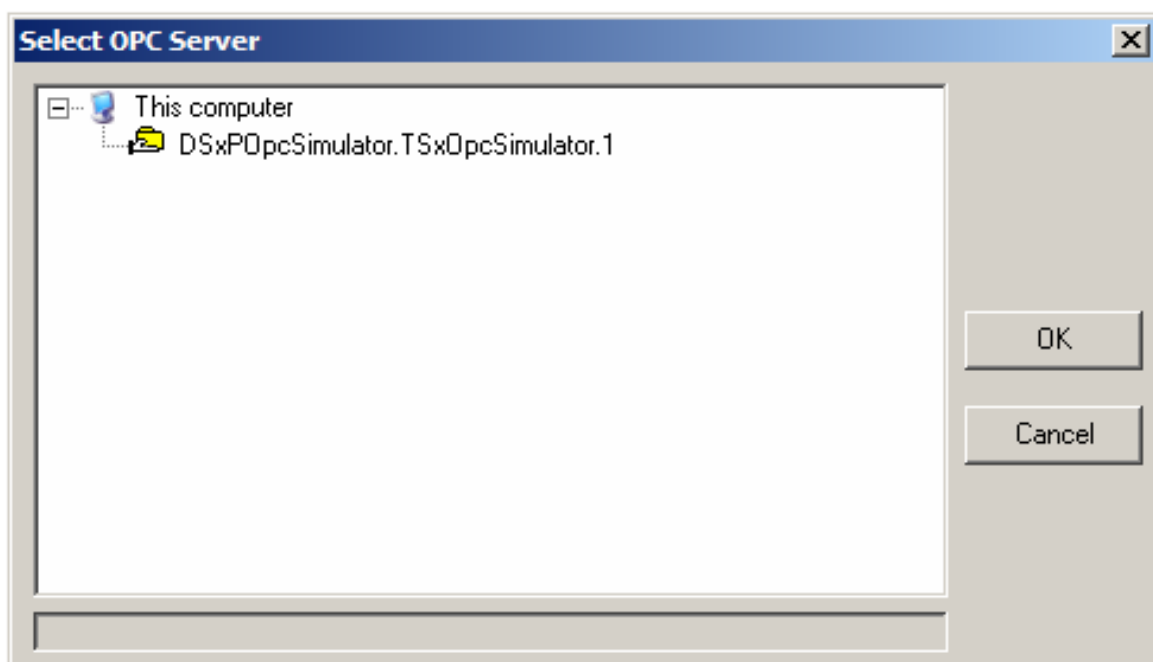


Иллюстрация 9: Выбор OPC сервера

Необходимо выбрать будут ли данные читаться непосредственно из устройства (Read from DEVICE) или из кэша OPC сервера (Read from CACHE)

4. Определение устройств

В разделе конфигурация выбираем Устройства. Добавляем устройства №1 и 2 с адресами 1 и 2 с привязкой к каналу 1, настроенному ранее.

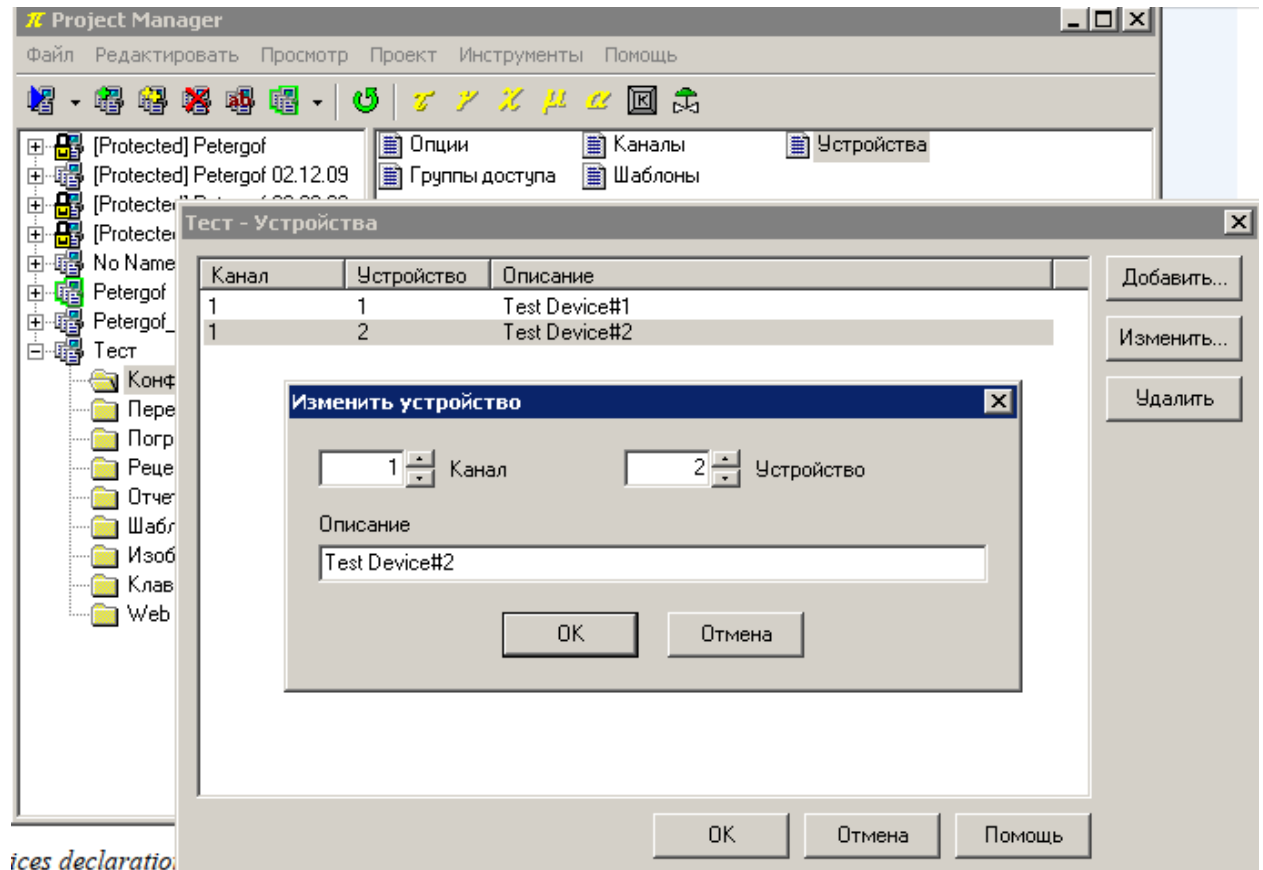


Иллюстрация 10: Определение устройств

5. Создание БД переменных

Теперь мы можем добавлять переменные устройств в БД. В данном примере мы используем только числовые и дискретные переменные.

Числовыми переменными мы называем переменные имеющие аналоговые значения (например измеренные переменные, значения уставок и т. д.). Они занимают байт, слово или двойное слово, и являются либо целочисленными переменными, либо переменными с плавающей точкой.

Дискретными переменными мы называем такие переменные, которые описывают статус (например состояние тревоги и т. д.). Они занимают один бит.

Часто несколько дискретных переменных могут быть собраны в одну числовую переменную (в данном примере это не используется).

Что бы редактировать БД переменных необходимо запустить Gate Builder. Из Project Manager выберите Переменные и сделайте дабл-клик на Численные.

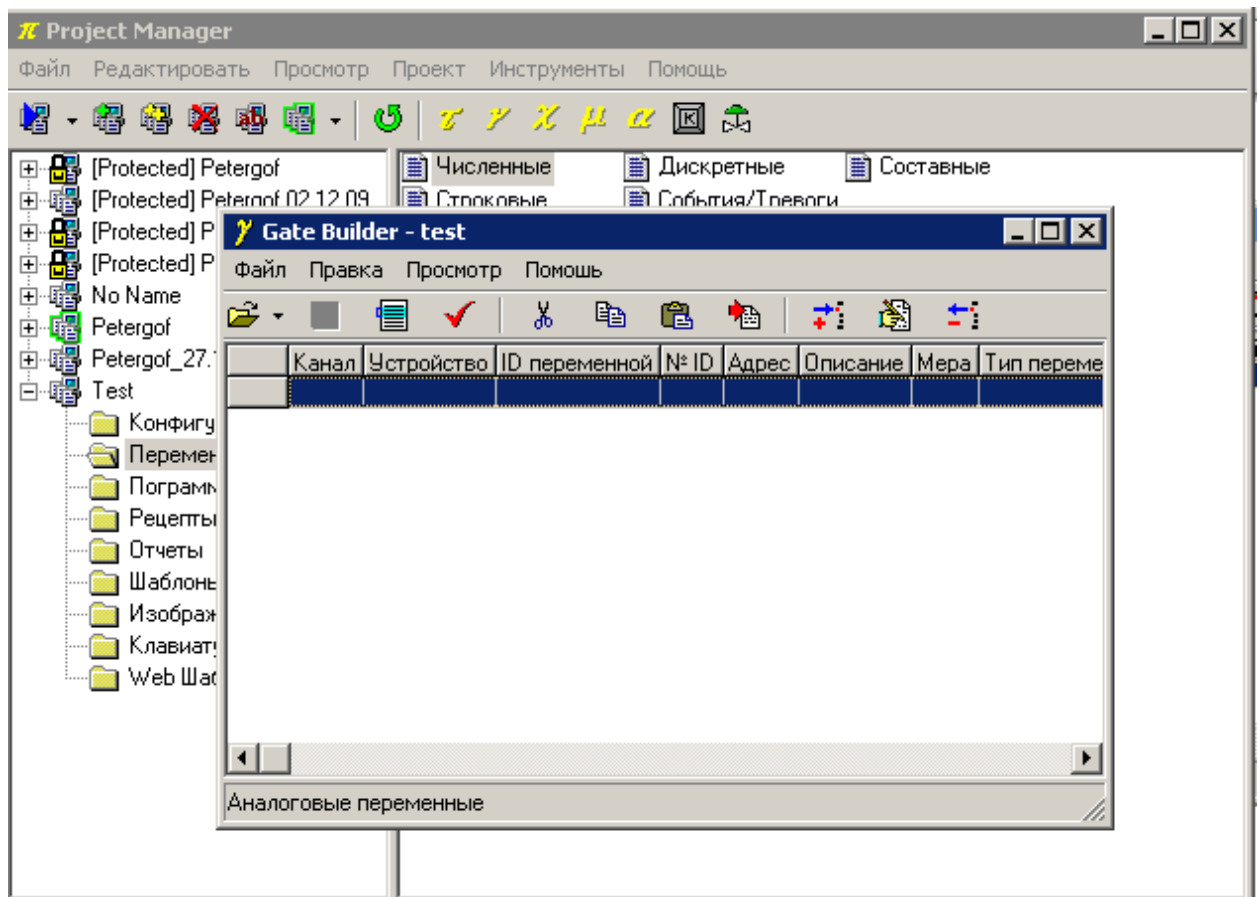


Иллюстрация 11: Заполнение базы данных переменных

Вот таблица переменных, которые мы будем заполнять:

Имя	Канал	Устр.	OPC адрес*	Тип	Тип переменной	Ед.и зм.	Описание
TEMP 1	1	1	Device1.3005	Численная	Signed Word	°C	T измеренная
SP 1	1	1	Device1.3010	Численная	Signed Word	°C	T уставка
OUT 1	1	1	Device1.3015	Численная	Unsigned Word	%	Контрольн ый выход
ALARM 1	1	1	Device1.1012	Дискретная	Bit		Внутренняя тревога
TEMP 2	1	2	Device2.3005	Численная	Signed Word	°C	T измеренная
SP 2	1	2	Device2.3010	Численная	Signed Word	°C	T уставка
OUT 2	1	2	Device2.3015	Численная	Unsigned Word	%	Контрольн ый выход
ALARM 2	1	2	Device2.1012	Дискретная	Bit		Внутренняя тревога

* OPC адреса могут быть уже определены OPC сервером или определены пользователем. Адрес не должен содержать более 80 символов (смотри документ «Описание протоколов» (Protocol guide)).

5.1. Конфигурация численных переменных

Заносим в БД данные о численных переменных не забыв указать номер устройства и номер N ID.

Аналоговые переменные

Основные | Опрос | Значение | Преобразование | Чувствительность

Temp ID переменн Записывать в файл истории
 Разрешить запись на устройство

1 № ID

Описание
Температура измеренная

Группы доступа

Иллюстрация 12: Temp1 Основные параметры

Аналоговые переменные

Основные | Опрос | Значение | Преобразование | Чувствительность

1 Канал → Протокол: OPC Client

Device1.3005 Адрес

Всегда Обновление Читать блок

1 Частота обновления [сек.]

Иллюстрация 13: Temp1 Вкладка опрос

Нажмите кнопку и выберите переменную, доступную на OPC сервере.

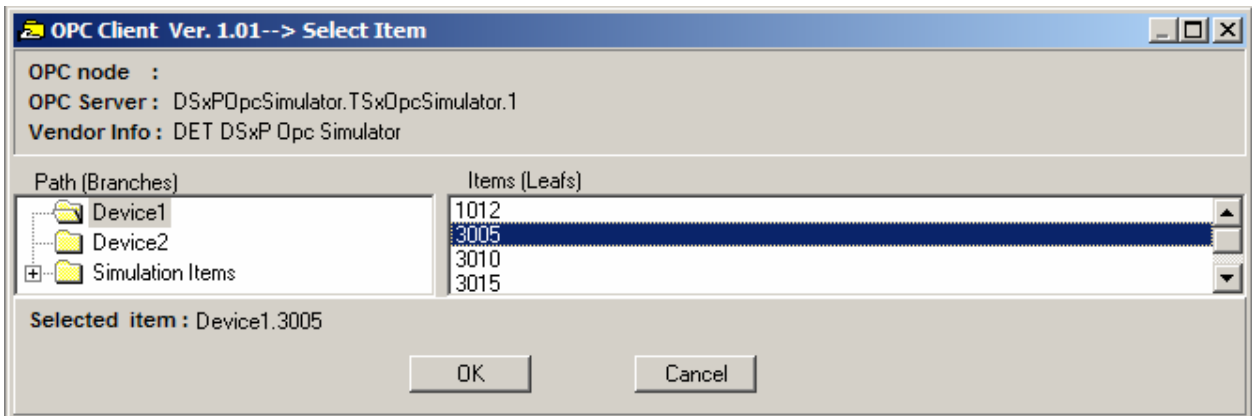


Иллюстрация 14: Привязка переменной к OPC

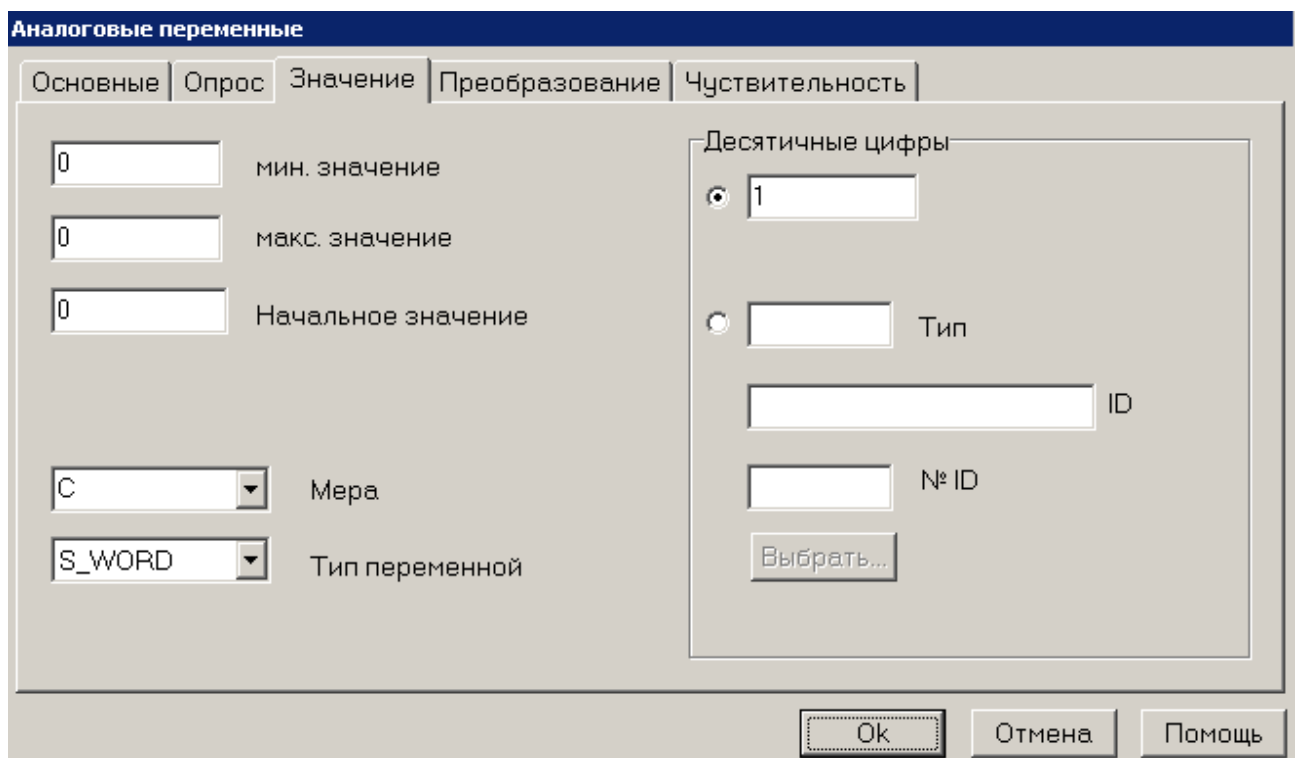


Иллюстрация 15: Темр1 Вкладка значение

Добавим переменную SP1

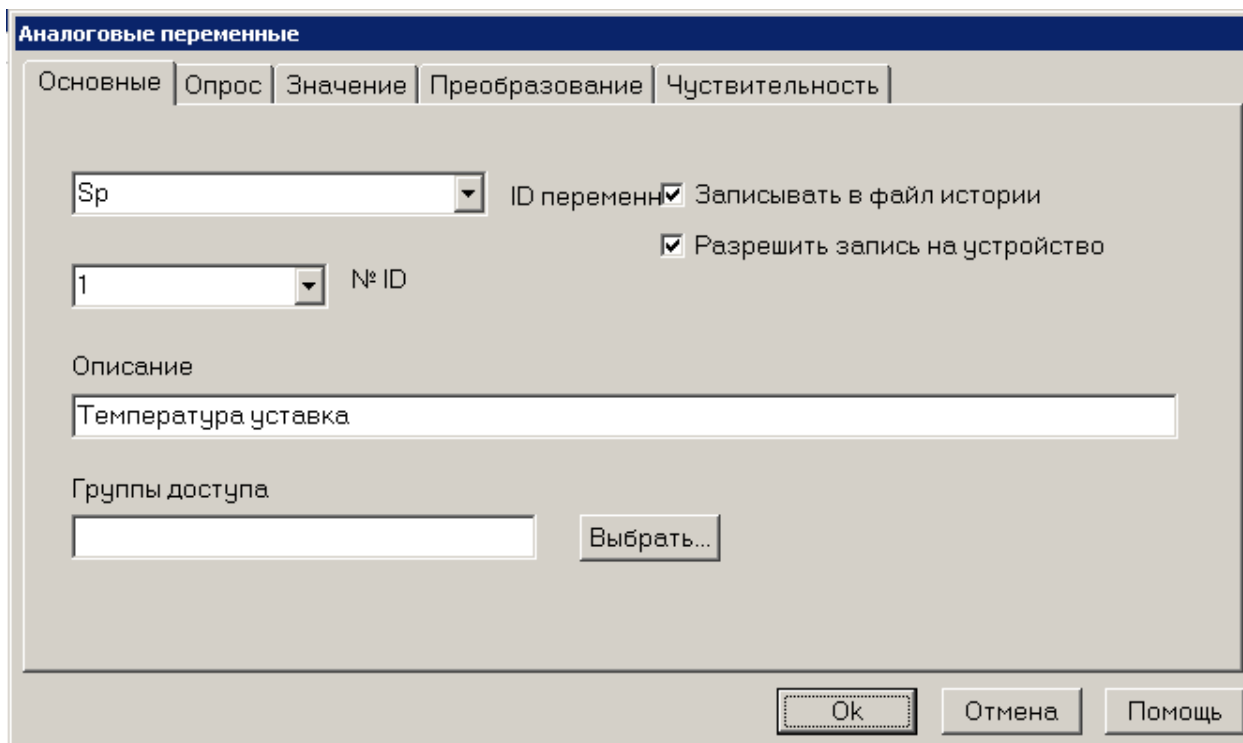


Иллюстрация 16: SP1 Группа основные

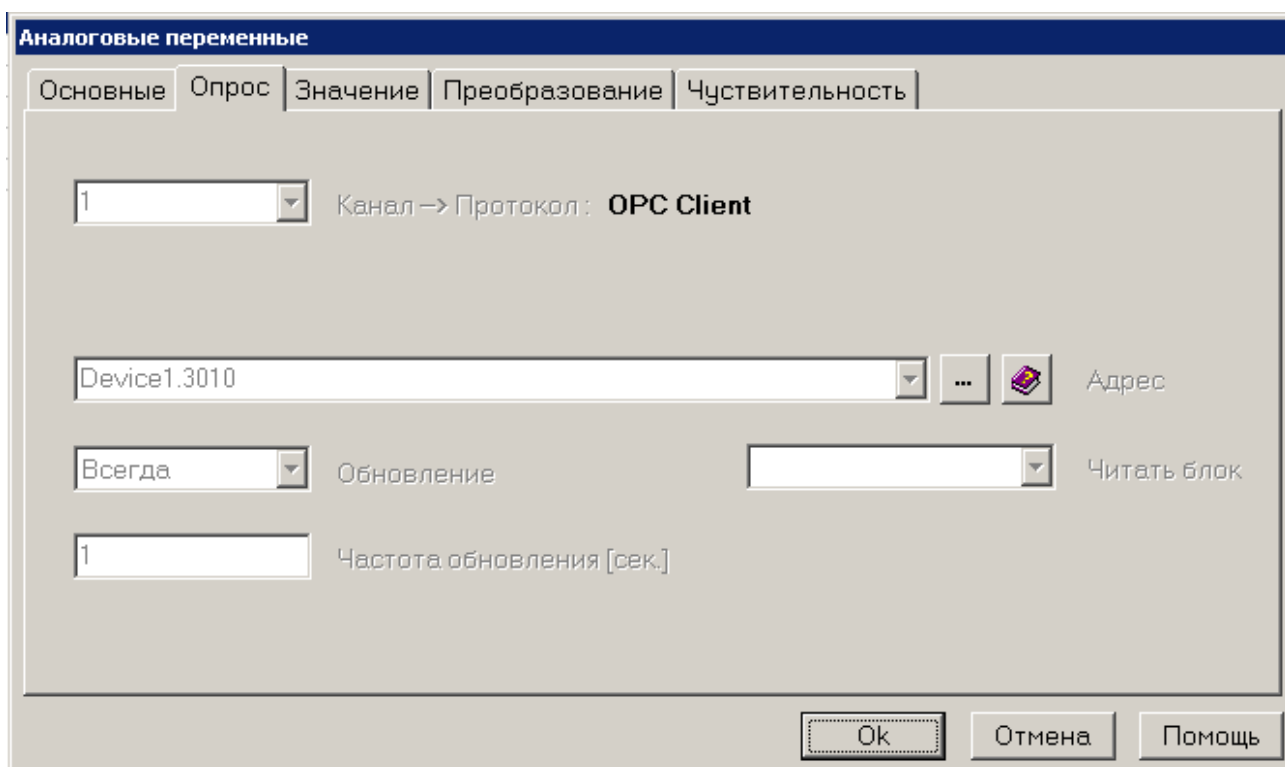


Иллюстрация 17: SP1 Вкладка опрос

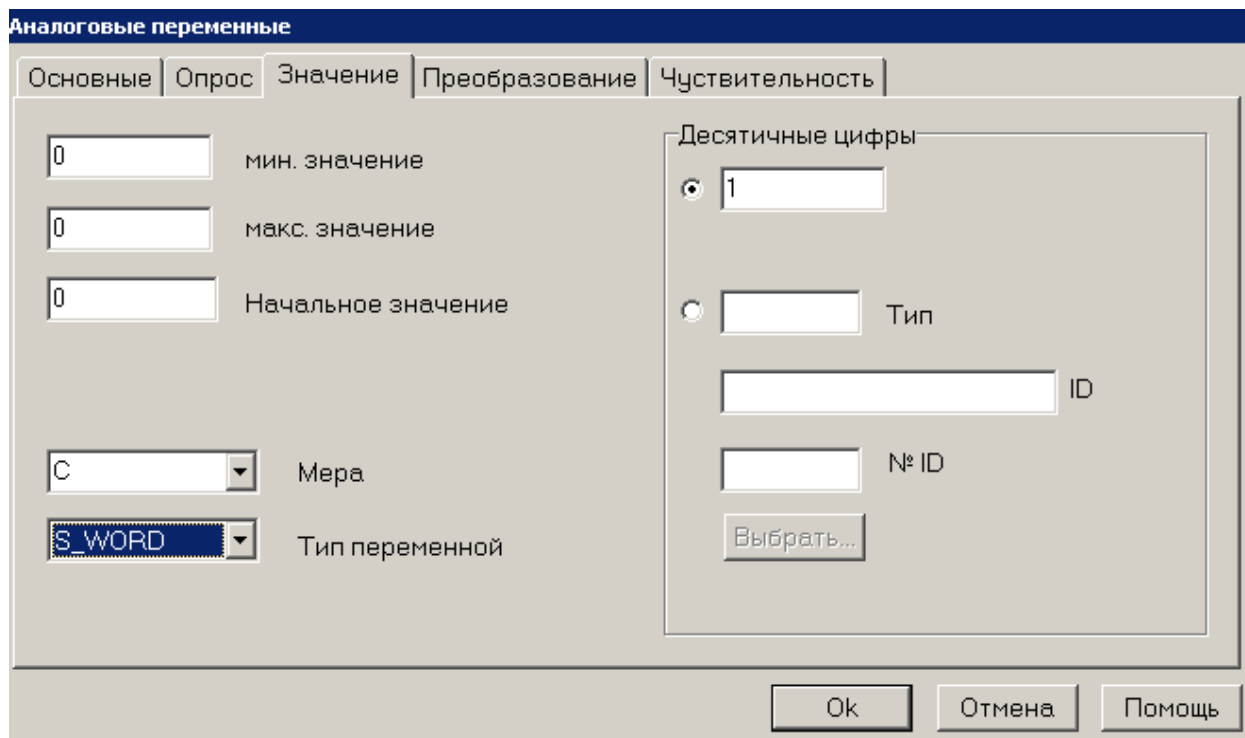


Иллюстрация 18: SP1 Вкладка значение

Добавим переменную OUT

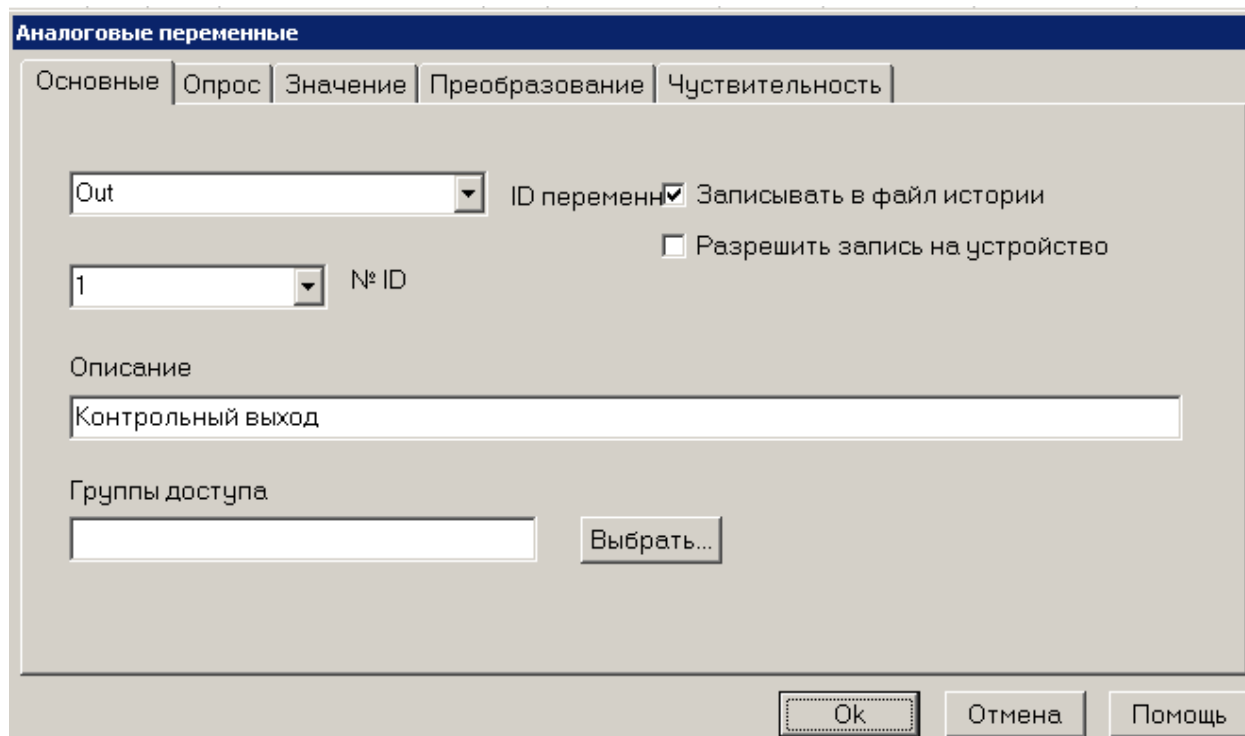


Иллюстрация 19: OUT1 Основные настройки

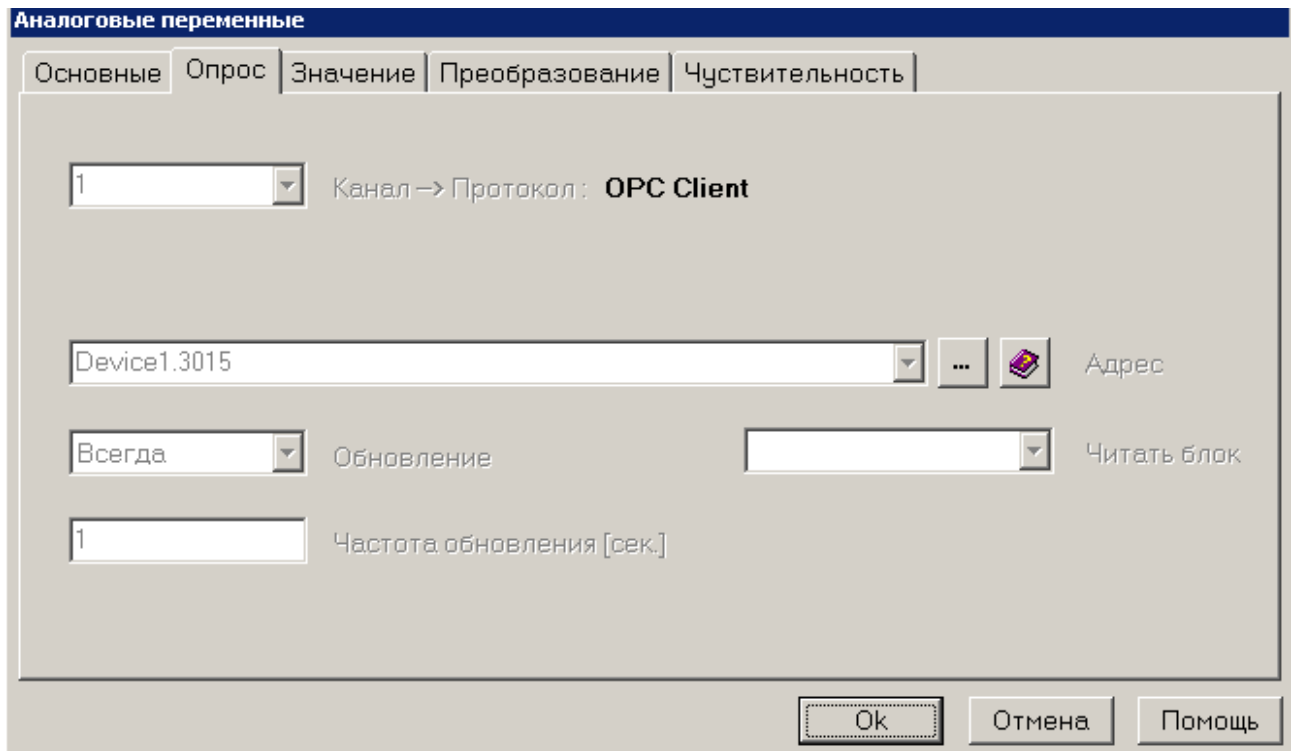


Иллюстрация 20: OUT1 вкладка опрос

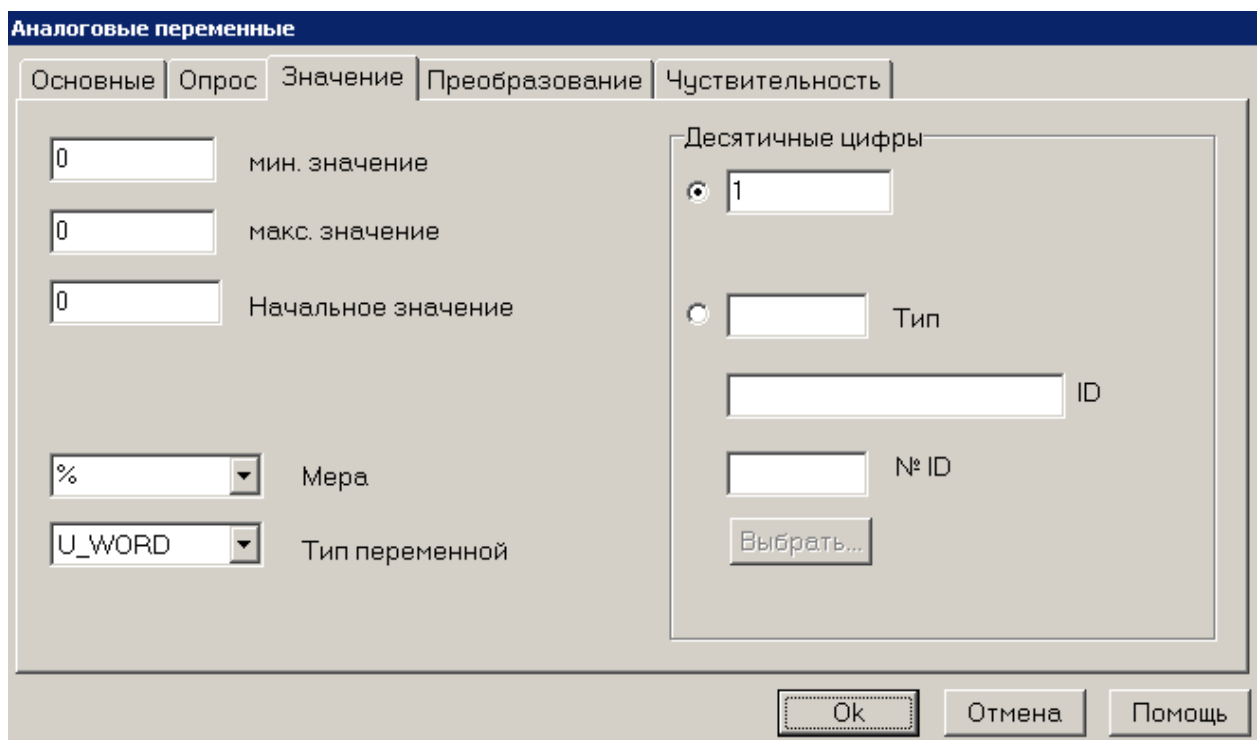
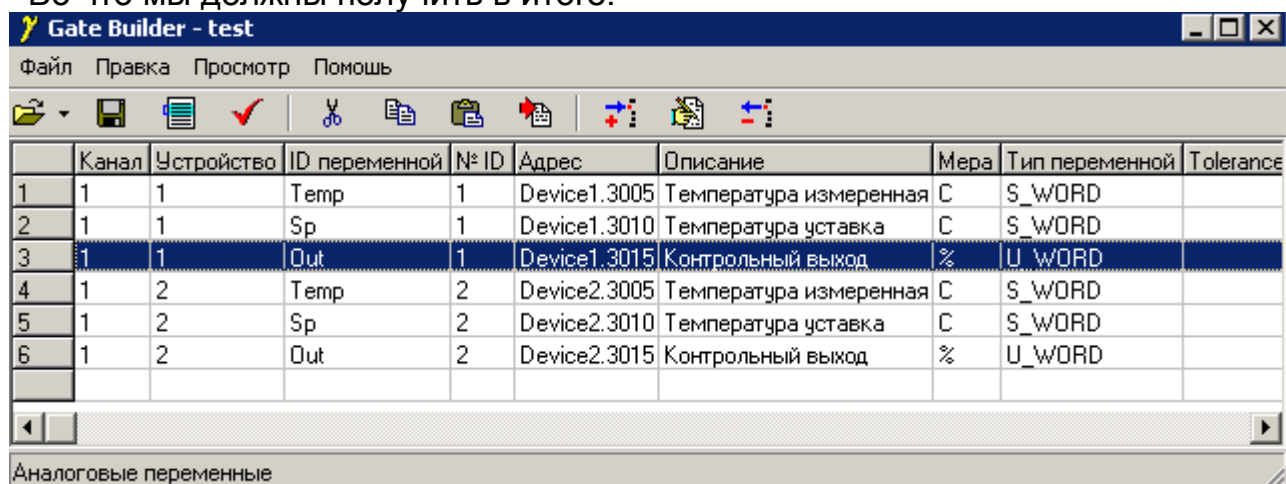


Иллюстрация 21: OUT1 Вкладка значение

Аналогично заполняется база для устройства №2.

Во что мы должны получить в итоге:



The screenshot shows the 'Gate Builder - test' application window. It features a menu bar with 'Файл', 'Правка', 'Просмотр', and 'Помощь'. Below the menu is a toolbar with icons for file operations and editing. The main area contains a table with the following data:

	Канал	Устройство	ID переменной	№ ID	Адрес	Описание	Мера	Тип переменной	Tolerance
1	1	1	Temp	1	Device1.3005	Температура измеренная	C	S_WORD	
2	1	1	Sp	1	Device1.3010	Температура уставка	C	S_WORD	
3	1	1	Out	1	Device1.3015	Контрольный выход	%	U_WORD	
4	1	2	Temp	2	Device2.3005	Температура измеренная	C	S_WORD	
5	1	2	Sp	2	Device2.3010	Температура уставка	C	S_WORD	
6	1	2	Out	2	Device2.3015	Контрольный выход	%	U_WORD	

At the bottom of the window, there is a status bar with the text 'Аналоговые переменные' and a scroll bar.

Иллюстрация 22: Окончательная таблица

5.2. Конфигурация дискретных переменных

Заносим в БД данные о дискретных переменных не забыв указать номер устройства и номер N ID.

Дискретные переменные

Основные | Опрос | Значение

Alarm ID Переменн Записывать в файл истории
 Разрешить запись на устройство

1 № ID

Описание
Внутренняя тревога

Группы доступа
Выбрать...

Ok Отмена Помощь

Иллюстрация 23: Alarm1 основные параметры

Дискретные переменные

Основные | Опрос | Значение

1 Канал -> Протокол: OPC Client

Device1.1012 Адрес

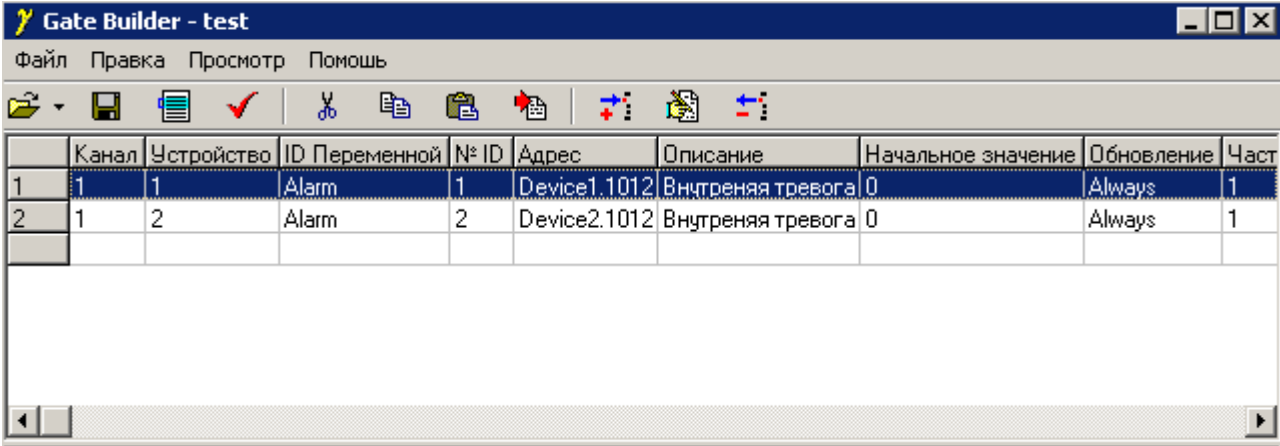
Всегда Обновление Читать блок

1 Частота обновления[сек.]

Ok Отмена Помощь

Иллюстрация 24: Alarm1 вкладка Опрос

Аналогично добавляем переменную 2-го устройства. В итоге мы должны получить следующую таблицу:



The screenshot shows a window titled "Gate Builder - test" with a menu bar (Файл, Правка, Просмотр, Помощь) and a toolbar. Below the toolbar is a table with the following data:

	Канал	Устройство	ID Переменной	№ ID	Адрес	Описание	Начальное значение	Обновление	Част
1	1	1	Alarm	1	Device1.1012	Внутренняя тревога	0	Always	1
2	1	2	Alarm	2	Device2.1012	Внутренняя тревога	0	Always	1

Иллюстрация 25: Окончательная таблица дискретных переменных

5.3. Конфигурирование переменных тревог (событий)

Итак мы заполнили базы дискретных и численных переменных, теперь мы создадим для каждого устройства переменную тревоги.

Эти переменные не читаются с устройства, а генерируются скадой и их статус в режиме рантайм мы можем увидеть в разделах «статус событий и тревог» и «история событий и тревог».

Давайте создадим тревоги со следующими параметрами:

Имя	Условие	Время фильтрации	Сообщение	Регистрировать
Internal_Alarm,1	=1	10 с	Внимание! Внутренняя тревога! Устройство 1.	Да
Internal_Alarm,2	=1	10 с	Внимание! Внутренняя тревога! Устройство 2.	да

The screenshot shows the 'Event gates' configuration window with the 'Conditions' tab selected. The configuration is as follows:

- Event name: Internal_Alarm
- ID variable: 1
- Condition: =1
- Checkboxes: Тревога, Need Acknowledge, Записывать в файл истории
- Access groups: A text box with a 'Выбрать...' button.

Иллюстрация 26: Тревога Основные параметры

Event gates

Основные | **Условия** | Сообщение | Класс

Condition gate

DIG Тип

Alarm ID перемена

1 № ID

Выбрать...

= Условие

1 значение

10 Время фильтрации

Ok Отмена Помощь

Иллюстрация 27: Тревога вкладки Условия

Event gates

Основные | Условия | **Сообщение** | Класс

Сообщение

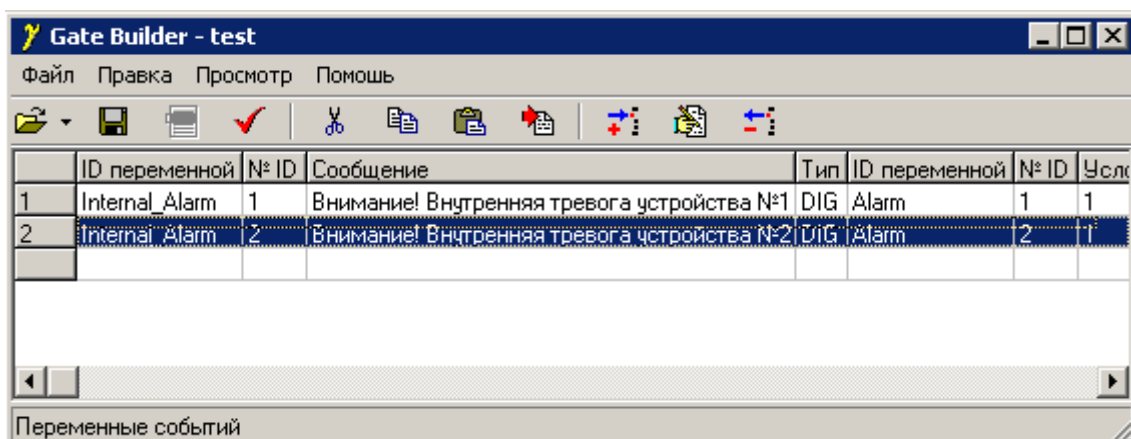
Внимание! Внутренняя тревога устройства №1

	Тип	ID переменной	№ ID	
Пункт 1				Выбрать...
Пункт 2				Выбрать...
Пункт 3				Выбрать...
Пункт 4				Выбрать...
Пункт 5				Выбрать...
Пункт 6				Выбрать...
Пункт 7				Выбрать...
Пункт 8				Выбрать...

Ok Отмена Помощь

Иллюстрация 28: Тревога - вкладка сообщение

После заполнения базы для всех устройств мы должны получить следующую таблицу:



	ID переменной	№ ID	Сообщение	Тип	ID переменной	№ ID	Усл
1	Internal_Alarm	1	Внимание! Внутренняя тревога устройства №1	DIG Alarm	1	1	1
2	Internal_Alarm	2	Внимание! Внутренняя тревога устройства №2	DIG Alarm	2	1	1

Переменные событий

Иллюстрация 29: Тревоги Общая таблица

6. Создание шаблонов (мнемосхем)

После того как мы заполнили базу переменных, настроили каналы связи с устройствами, можно приступить к созданию мнемосхем.

Выберите шаблоны в Project Manager и создайте новый шаблон, выбрав по правой кнопки мыши Новый->Файл. Переименуйте новый файл используя имя Main.

Двойным щелчком мыши на шаблоне запустите Template Builder.

6.1. Определение переменных шаблона

Прежде всего мы должны определить переменные, которые будут использоваться на мнемосхема. В нашем примере мы будем использовать их все. Нажимаем кнопку «...»



Появится новое окно, там мы нажимаем Add Gate, выбираем сначала численные переменные и нажимаем ок. Повторяем эту операцию с дискретными и тревожными переменными.

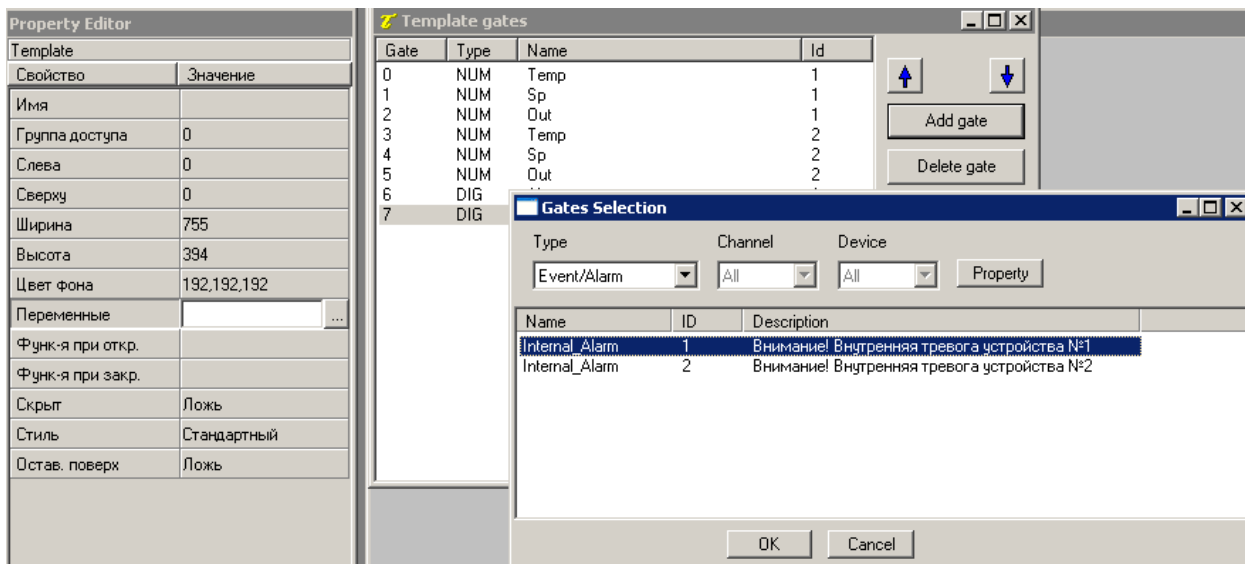




Иллюстрация 30: Переменные шаблона

6.2. Вставка объекта Label

Прежде всего создадим Фрэйм (Frame) и уже на него будем класть все остальные элементы.

Выберите на верхней панели инструментов  и кликните на свободном месте мнемосхемы. Должен получиться пустой прямоугольник.

Следующим шагом добавим на фрэйм Метку (Label), которая будет содержать статичный текст. Выберите на верхней панели инструментов **A**, затем кликните на добавленный ранее Фрэйм. Чтобы изменить текст используйте Property Editor, свойство Метка. Введите текст «Температура». Теперь мы хотим чтобы здесь же отображалась измеренная температура.

Для этого мы свяжем Метку с переменной. Для этого выберите свойство Переменная и нажмите  и выберите NUM,Temp,1 из списка переменных. Добавьте еще одну Метку добавив в нее текст %5.0lf °C как показано на иллюстрации 31. Все объекты, которые вы добавляете на мнемосхему можно настраивать с помощью Property Editor.

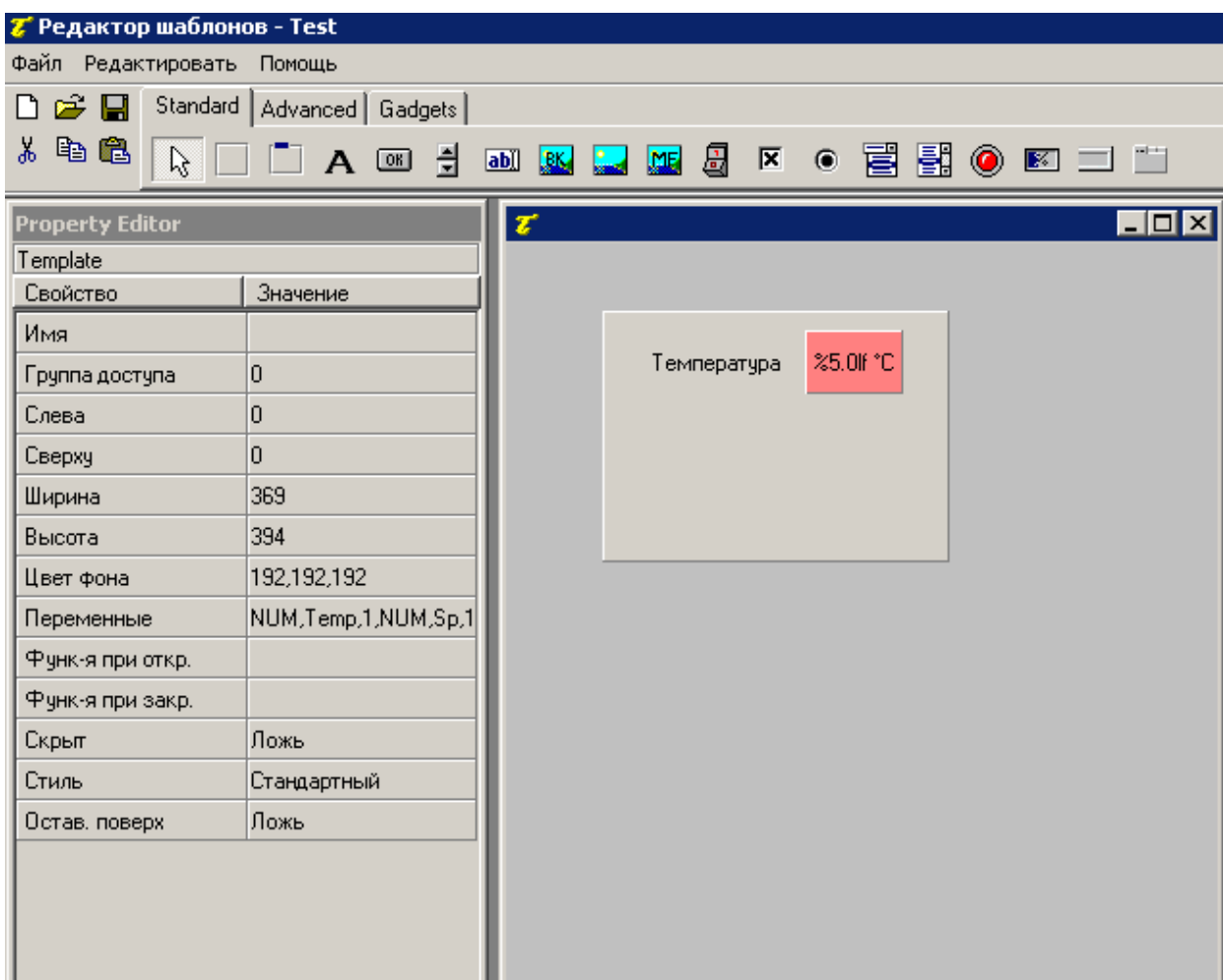



Иллюстрация 31: Создание Меток

6.3. Вставка объекта Редактор (Edit)

Добавьте на мнемосхему еще одну Метку и напишите в ней Уставка.
Чтобы сделать возможным ввод уставки и передачу ее на контроллер, выберите на верхней панели . Привяжите к Редактору переменную NUM,Sp,1, по аналогии с предыдущими элементами.

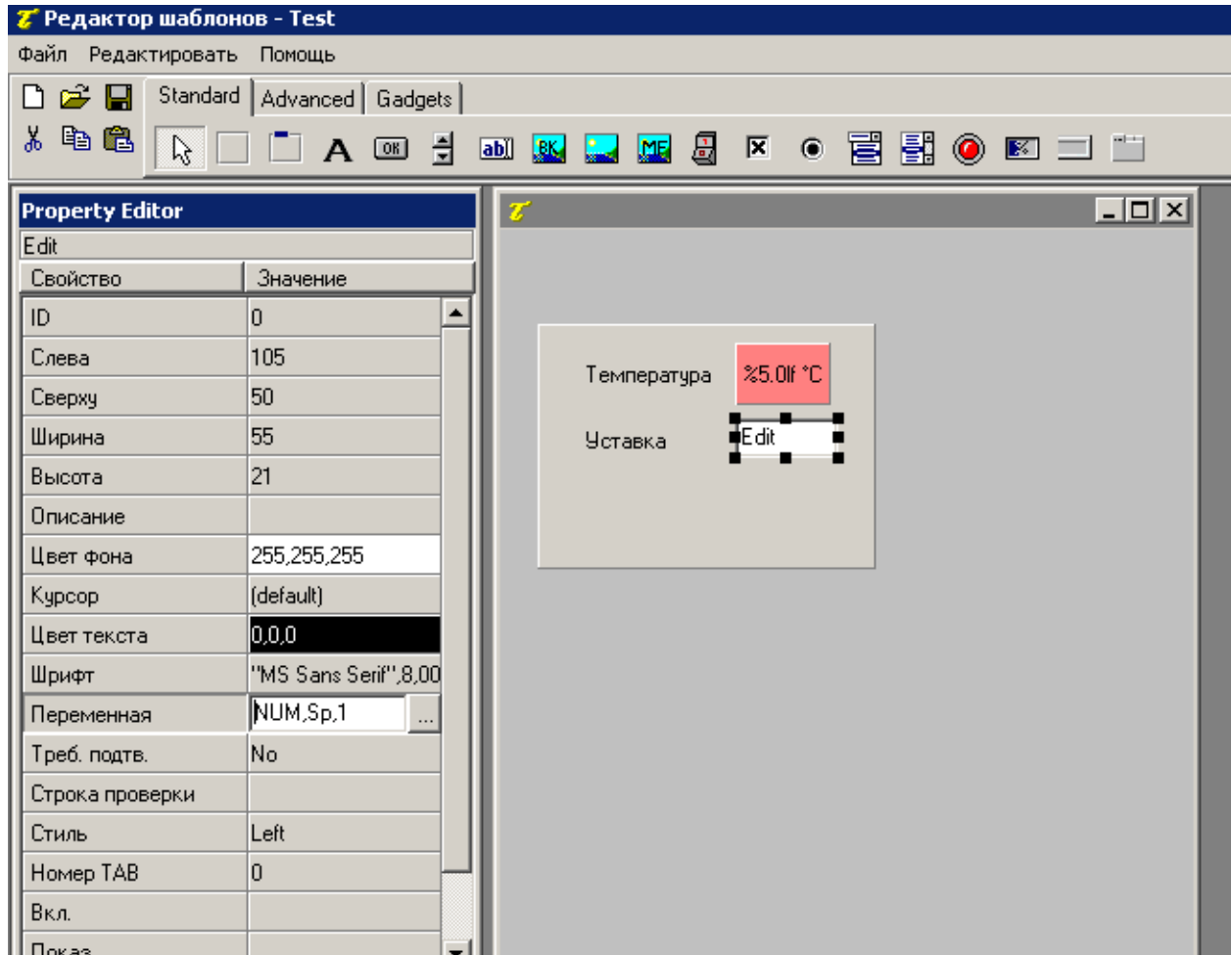



Иллюстрация 32: Создание редактора уставки

6.4. Вставка объекта градуировка (Gauge)

Добавьте еще одну метку и напишите текст Выход. Вставьте объект Gauge () и свяжите его с переменной NUM,Out,1 используя Property Editor.

Этот вариант позволяет отображать выходную мощность устройства в виде графической полоски.

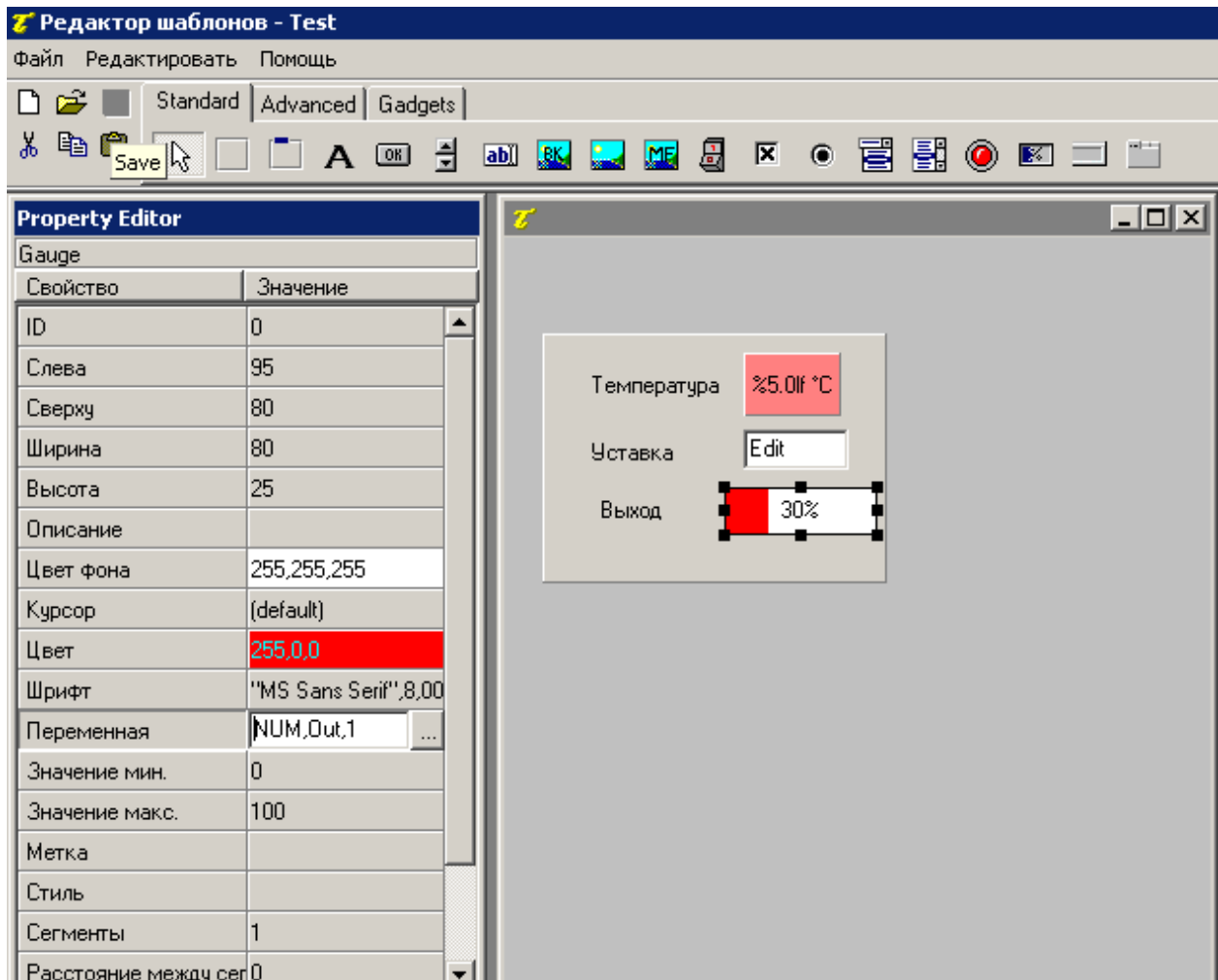


Иллюстрация 33: Создание объекта градуировка

6.5. Вставка объекта индикатор (Led)

Добавьте еще одну метку с текстом Ошибка. Добавьте объект индикатор (Led). Чтобы добавить «анимации» объекту, необходимо указать условия по которому объект будет менять свой цвет. Для этого измените свойство Индик. ON состояние, связав его с переменной Internal_Alarm,1 (Internal_Alarm,1==true). В таком случае при возникновении ошибки индикатор станет красным иначе он будет зеленым.

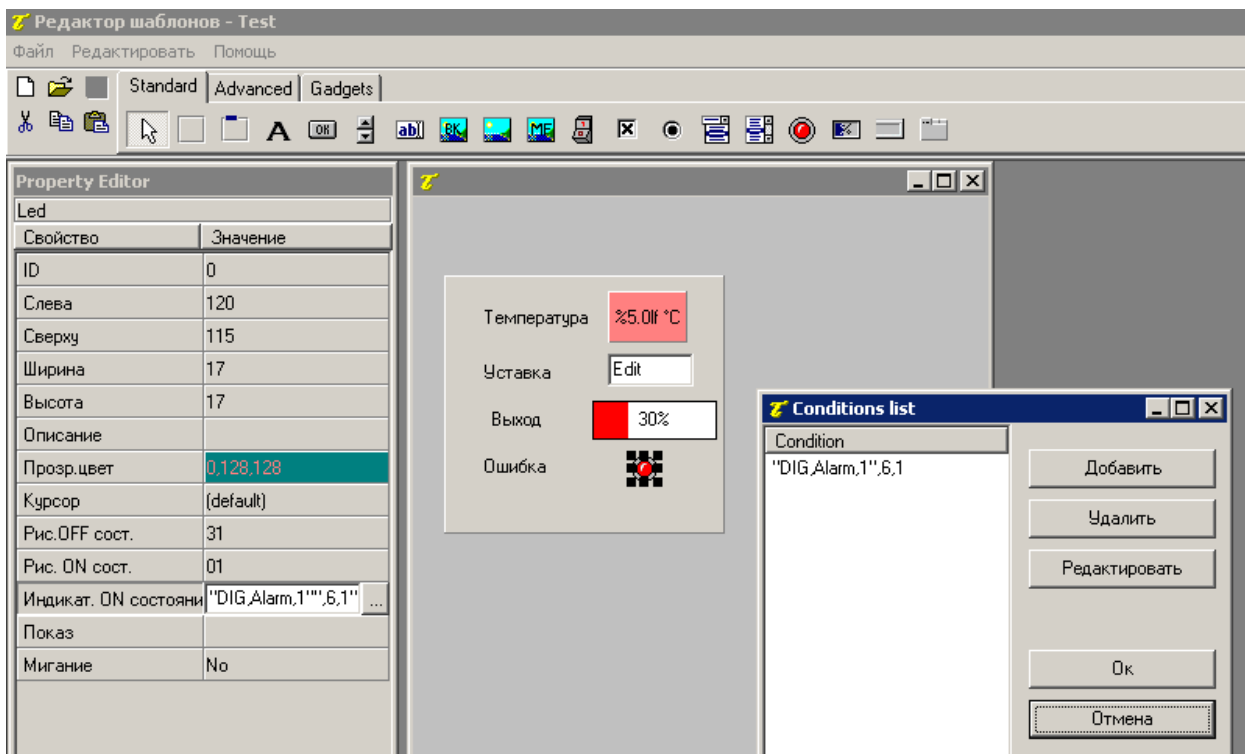


Иллюстрация 34: Создание объекта индикатор

6.6. Завершение мнемосхемы

На данный момент все переменные с первого устройства отображаются на мнемосхеме. Чтобы отобразить аналогичным образом переменные второго устройства достаточно выделить созданный нами Фрэйм, копировать его и вставить на мнемосхему. Будьте осторожны, не вставьте фрэйм внутрь исходного. Чтобы избежать этой ошибки кликните на свободном месте мнемосхемы прежде чем вставлять копию. Теперь достаточно поменять привязки к переменным всех объектов во втором фрэйме.

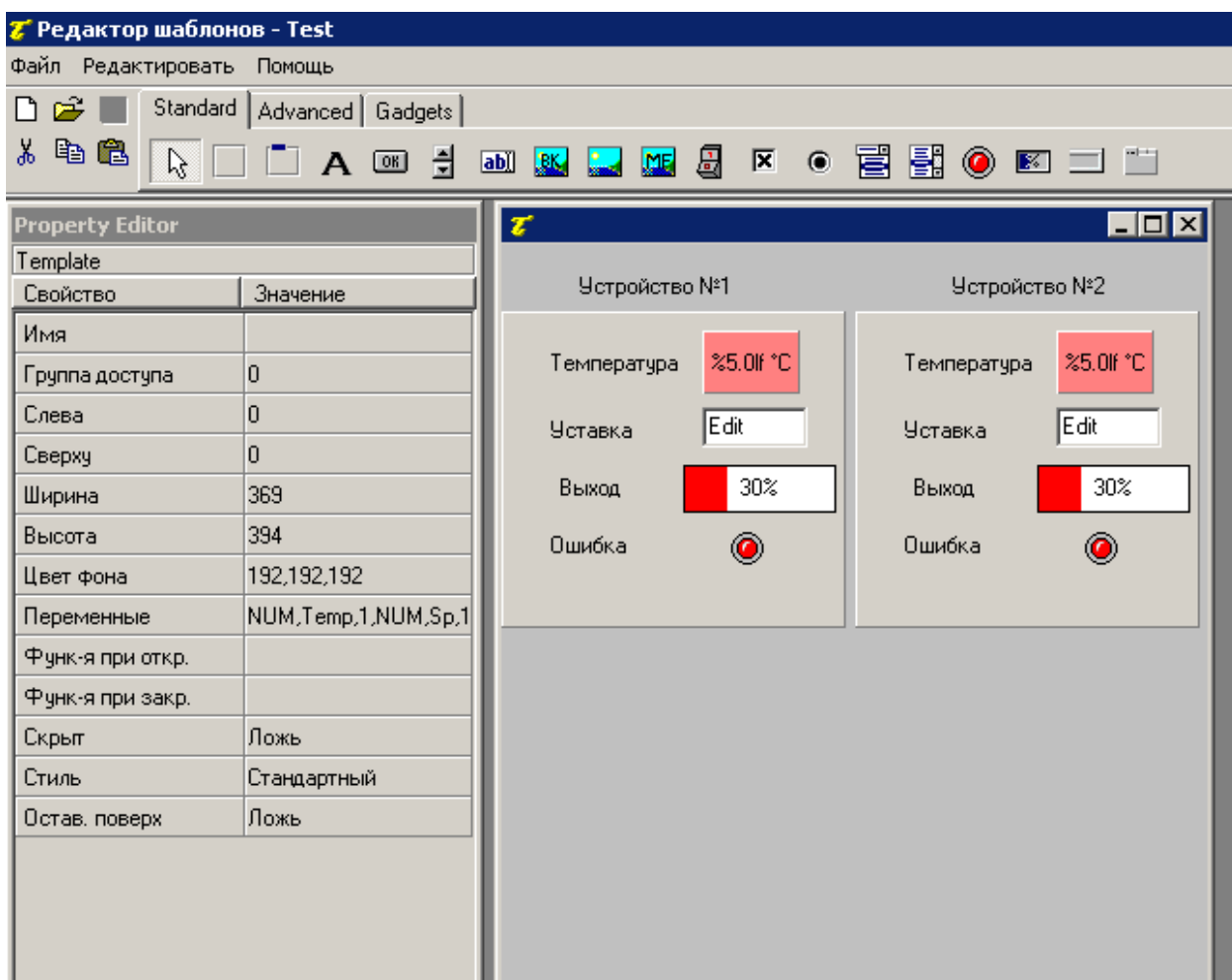


Иллюстрация 35: Два устройства на мнемосхеме

Для завершения мнемосхемы мы вставим на нее фон при помощи инструмента Background ().

В качестве фона выберем созданный заранее bmp файл, скопированный в папку Bitmars проекта.

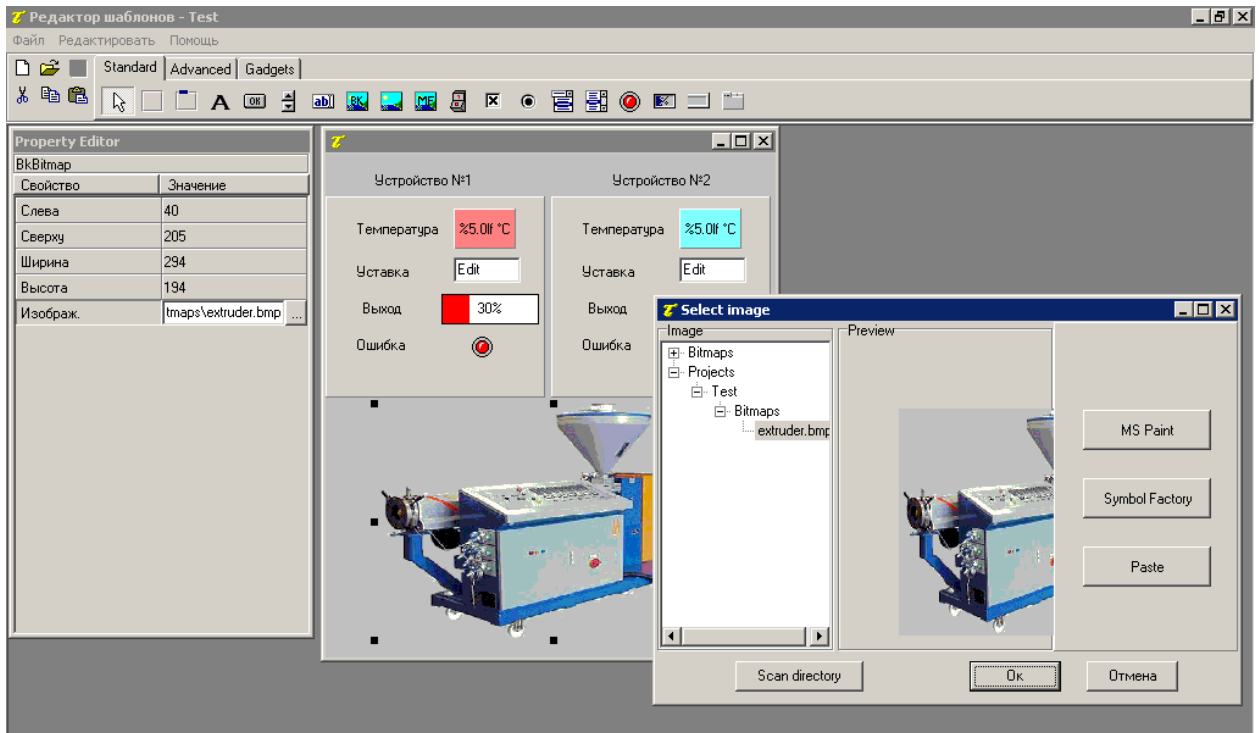


Иллюстрация 36: Вставка фона

7. Пример написания программ

Сейчас мы создадим функцию, которая будет открывать мнемосхему при запуске рантайма. В программе Project Manager в разделе Программы создайте файл и переименуйте его в main. Откройте его, запустится Code Builder.

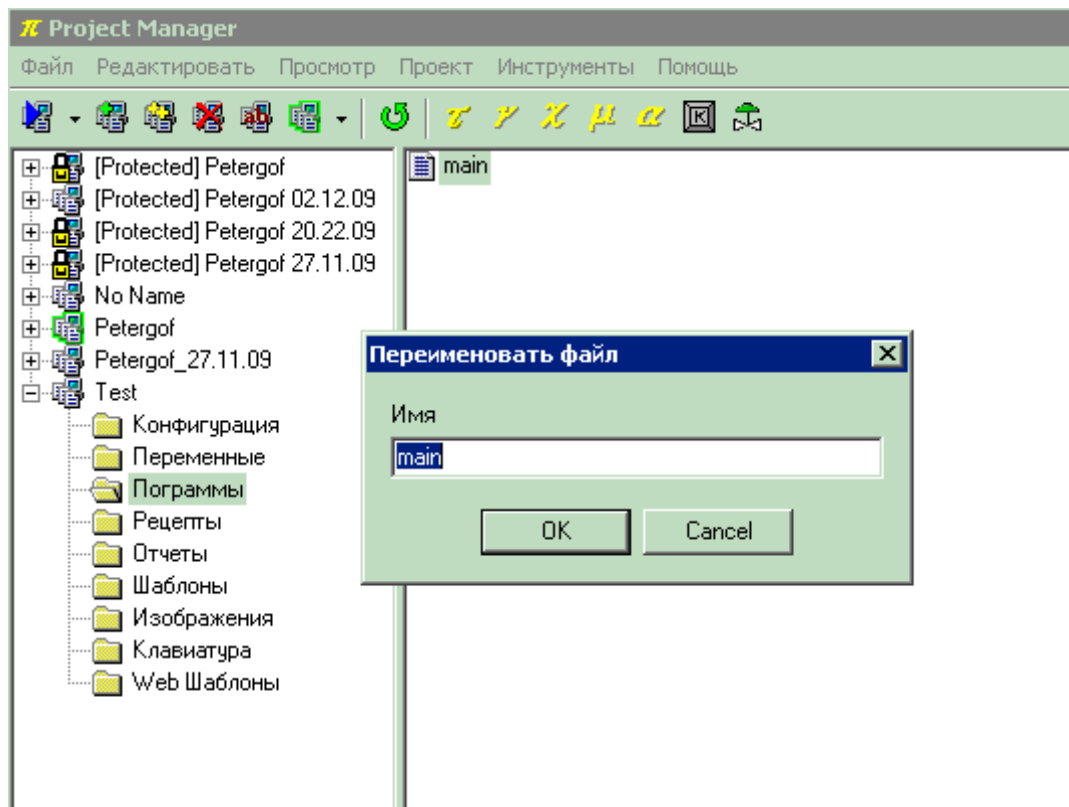


Иллюстрация 37: Создание новой программы

Code Builder это средство для программирования, мы используем его только для создания одной функции, которая откроет окно main при запуске рантайма.

Скопируйте следующий код:

```
// Function called at Winlog startup
Function void Main()
#Startup
//*****
// Open default page
//*****
TPageOpen("Main");
end
```

Для того чтобы проверить синтаксис нажмите кнопку .

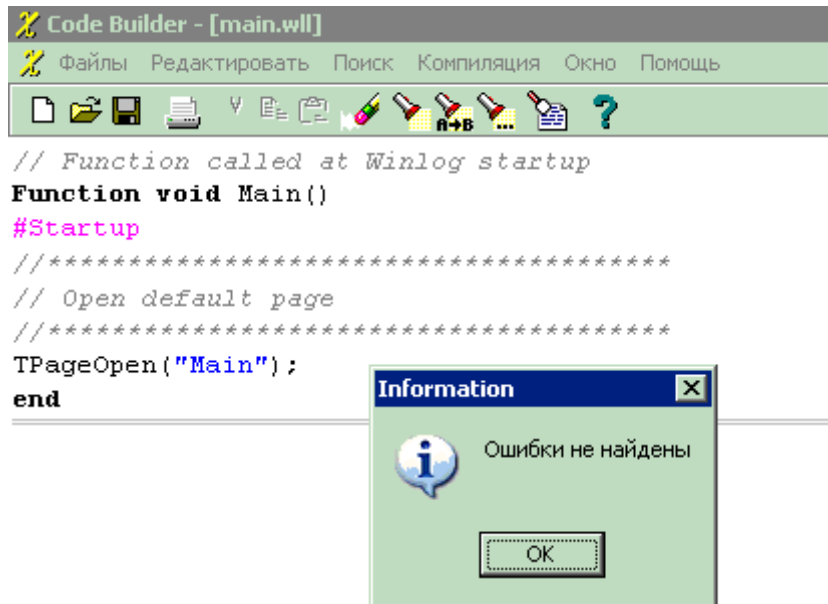


Иллюстрация 38: Проверка программы

8. Запуск проекта

Наш пример закончен. Подключаем устройства и запускаем проект из Project Manager.

Таким образом мы попадаем в фазу рантайм. Winlog Pro начинает опрашивать переменные с (на) устройств(а) и представлять результат в графической форме (трэнды, мнемосхемы) и в табличном виде (отчеты и историческая информация).

Когда проект запустится мнемосхема Main появится на экране автоматически. С помощью меню вы можете вызвать трэнды, выбрать необходимые переменные, или добавить их в группы.

Так же вы можете посмотреть онлайн статус и историю Тревог выбирая соответствующие пункты меню.

