**Сигналы проекта, база сигналов проекта, векторизованный блок.**

*Рассмотрены способы задания сигналов проекта, возможности их применения.*

При работе с большим количеством блоков, использующих одинаковые значения свойств, перед пользователем возникает задача задания этих свойств и их значений. По сути речь идет о глобальных переменных проекта, или комплексной модели, причем доступ к этим переменным необходимо получать в каждом модуле комплексной модели. Такие глобальные переменные в SimInTech принято называть сигналами. Среда SimInTech позволяет решить задачу определения сигналов несколькими способами:

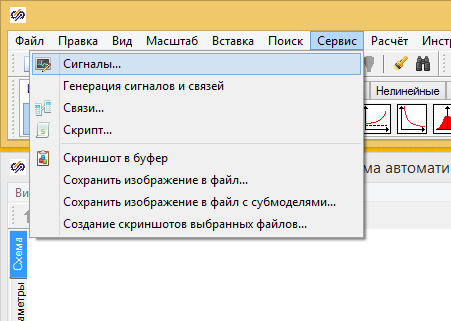
1. С помощью таблицы сигналов проекта;
2. Путем задания сигналов на вкладке «**Параметры**» Схемного окна проекта с помощью внутреннего языка программирования;
3. С помощью базы данных проекта.

Первые два способа определяют хотя и глобальные сигналы проекта, но они будут «видны» только внутри данного проекта. Третий способ определяет некие глобальные сигналы, значениями которых проект может «обмениваться» с другими проектами (моделируемыми в пакете, например). С точки зрения проекта («снизу»), все три способа задания сигналов проекта являются равнозначными и выбираются исходя из конкретных особенностей проекта. Исходя из этой равнозначности сигналов важно помнить, что **недопустимо задание сигналов с одинаковыми именами разными способами**.

Рассмотрим подробнее каждый из способов.

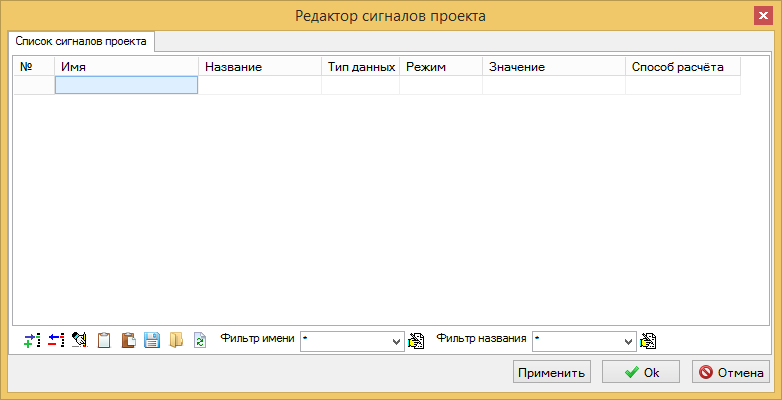
**Задание сигналов с помощью таблицы сигналов проекта**

Для того, чтобы воспользоваться данным способом, необходимо открыть редактор сигналов проекта. Для этого нужно пройти в меню «**Сервис**» → «**Сигналы**» (см. **Рисунок 1**).



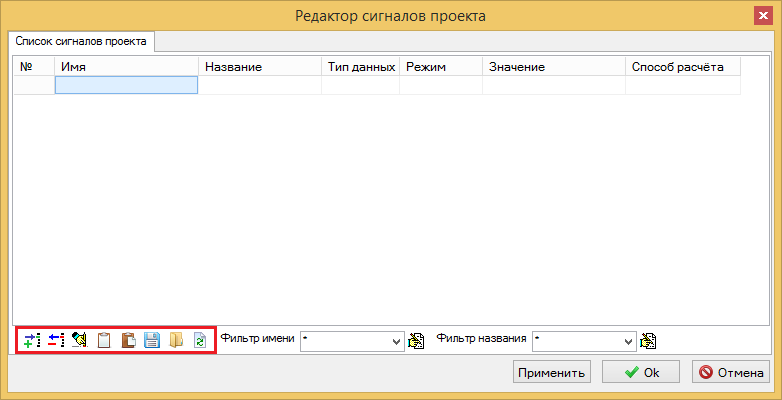
**Рисунок 1. Вызов редактора сигналов проекта**

После выбора данного пункта в меню откроется окно редактора сигналов проекта (см. **Рисунок 2**), в котором содержится таблица сигналов проекта и инструментарий для ее редактирования (в зависимости от загружаемого проекта список сигналов может быть либо пустым, либо c начальными сигналами).



**Рисунок 2. Окно редактора сигналов проекта**

Окно редактора содержит кнопки, которые находятся в нижней части, для изменения списка сигналов проекта (см. **Рисунок 3**):



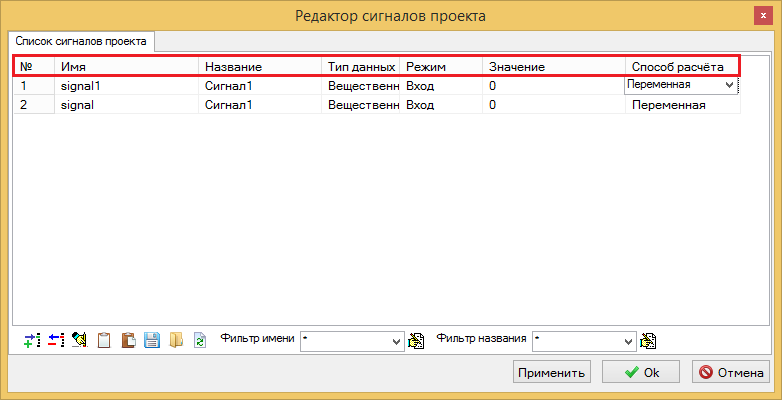
**Рисунок 3. Кнопки редактирования сигналов проекта**

Список кнопок и их предназначение:

1. «**Добавить сигнал**» - добавляет новую строку в таблицу сигналов;
2. «**Удалить сигнал**» - удаляет выбранную строку из таблицы сигналов;
3. «**Очистить список сигналов**» - удаляет все строки из таблицы сигналов;
4. «**Копировать в буфер**» - копирует выбранную строку в буфер;
5. «**Вставить из буфера**» - загружает строку из буфера;
6. «**Сохранить список в файл**» - сохраняет список сигналов в текстовый файл;
7. «**Загрузить список из файла**» - загружает список сигналов из ранее сохраненного текстового файла;
8. «**Обновить список из файла**» - загружает список сигналов из файла, сравнивает его со списком сигналов, находящимся в таблице сигналов, и при отсутствии сигнала в таблице сигналов, добавляет сигнал из файла.

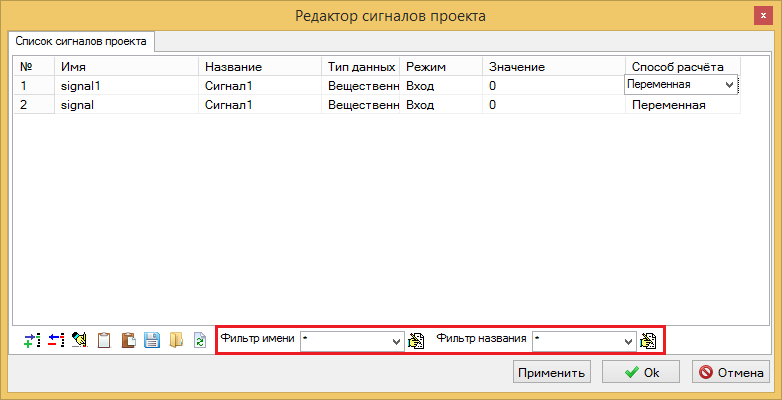
Таблица сигналов содержит следующие столбцы (см. **Рисунок 4**):

1. «**№**» - сквозной номер сигнала (данный номер автоматически присваивается при добавлении нового сигнала и изменяется только при редактировании общего списка сигналов, пользователь не может по своему усмотрению изменять номер сигнала);
2. «**Имя**» - имя сигнала, используемое для задания значений в блоках и скриптах (значение данного поля является редактируемым пользователем);
3. «**Название**» - краткое описание сигнала (значение данного поля является редактируемым пользователем);
4. «**Тип** **данных**» - используемый сигналом тип данных (пользователь выбирает подходящий тип из выпадающего списка);
5. «**Режим**» - используемый сигналом режим (вход – значение сигнала формируется за пределами расчетной схемы, выход – значение сигнала формируется на расчетной схеме, ненаправленный – значение сигнала может формироваться как за пределами расчетной схемы, так и на ней. Пользователь выбирает режим из выпадающего списка);
6. «**Значение**» - начальное значение сигнала (значение данного поля является редактируемым пользователем);
7. «**Способ** **расчета**» - используемый сигналом способ расчета (**переменная** – значение сигнала может изменятся в процессе расчета, **константа** – значение сигнала остается неизменным в течение расчета (если значение сигнала вычисляется по алгоритму, т.е. начальное значение является интерпретируемым выражением, то его значение вычисляется при инициализации и более не интерпретируется в процессе счета), **заблокирована** – значение сигнала остается неизменным в течение расчета (если начальное значение сигнала задано интерпретируемой формулой, то она не интерпретируется ни при инициализации, ни в процессе счета). Пользователь выбирает способ расчета из выпадающего списка.



**Рисунок 4. «Шапка» таблицы сигналов проекта с добавленными произвольными сигналами**

Окно редактора сигналов проекта так же содержит два фильтра (см. **Рисунок** **5**). Данные фильтры позволяют отображать сигналы, в которых содержится либо часть имени, либо часть названия, по которому происходит фильтрация.



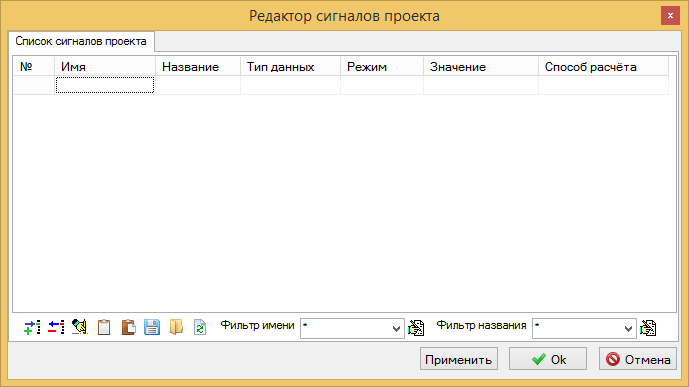
**Рисунок 5. Фильтры имени и названия**

Фильтрация происходит следующим образом:

* если мы хотим отфильтровать сигналы, у которых какая-то начальная часть имени либо названия одинакова, а изменяется лишь какая-то конечная часть, то в фильтре синтаксис выглядит следующим образом: **<начальная часть имени>\***. Например, signal\*. После выполнения фильтр в окне редактора сигналов проекта выведет все найденные сигналы, у которых имя либо название начинается на signal;
* если мы хотим отфильтровать сигналы, у которых какая-то конечная часть имени либо названия одинакова, а изменяется лишь какая-то начальная часть, то в фильтре синтаксис выглядит следующим образом: **\*<конечная часть имени>**. Например, \*signal. После выполнения фильтра фильтр в окне редактора сигналов проекта выведет все найденные сигналы, у которых имя либо название оканчивается на signal;
* если мы хотим отфильтровать сигналы, в которых неизменна только какая-то часть, содержащаяся внутри имени, тогда синтаксис фильтра выглядит следующим образом: **\*<часть имени>\***. Например, \*signal\*. После выполнения фильтр в окне редактора сигналов проекта выведет все найденные сигналы, у которых в имени будет найдена часть имени signal.

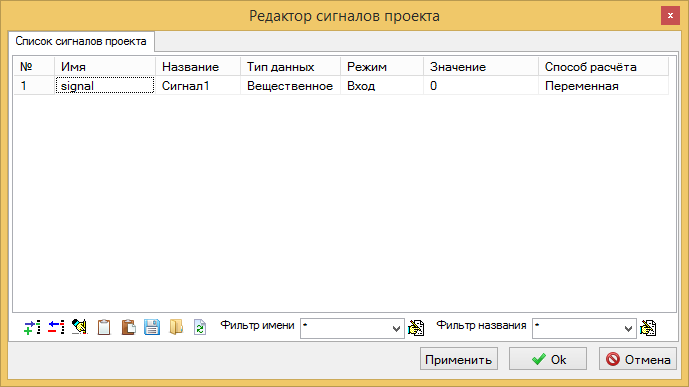
Для добавления нового сигнала в таблицу сигналов проекта необходимо:

1. Открыть окно редактора сигналов проекта (см. **Рисунок 6**);



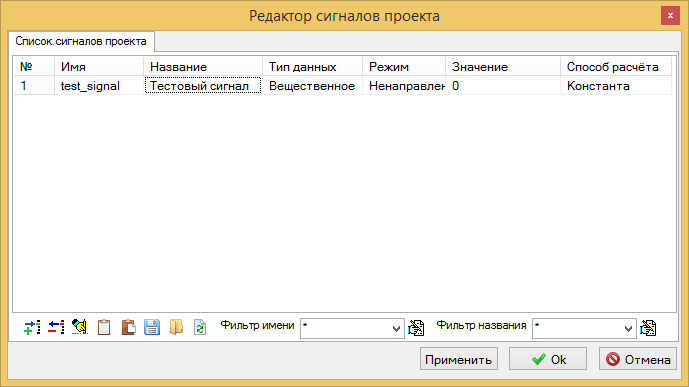
**Рисунок 6. Окно редактора сигналов проекта**

1. Добавить новую строку с помощью кнопки «**Добавить сигнал**» (см. **Рисунок 7**);



**Рисунок 7. Окно редактора сигналов проекта**

1. Параметризовать добавленные свойства, указав имя, название, тип данных, режим, значение и способ расчета (см. **Рисунок 8**);



**Рисунок 8. Окно редактора сигналов проекта**

1. Нажать кнопку «**Ok**».

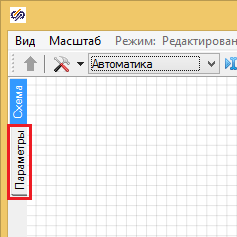
После выполнения вышеуказанных действий сигнал будет добавлен в проект и его можно будет использовать в рамках всего проекта.

После добавления сигнала в строке добавленного сигнала пользователь по своему усмотрению может изменять начальные, заданные им имя, название, тип данных и прочие свойства.

**Задание сигналов на вкладке «Параметры»**

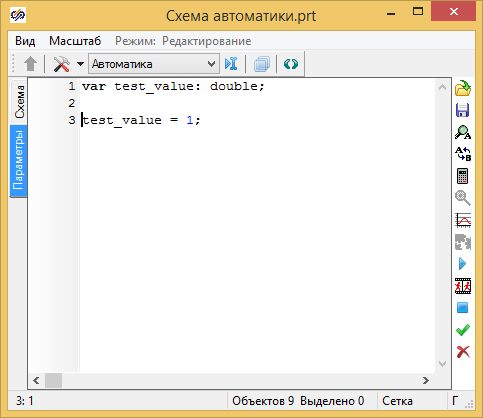
Задание сигналов на вкладке параметры осуществляется с помощью внутреннего скриптового языка программирования с использованием стандартных способов задания. Объявленные таким образом сигналы (явно заданные типизированные переменные и константы языка программирования) будут видны в рамках всего проекта.

Для того, чтобы задать сигналы таким способом нужно перейти на вкладку «**Параметры**» в окне проекта (см. **Рисунок 9**).



**Рисунок 9. Вкладка «Параметры» на листе проекта**

В качестве примера ниже приведён фрагмент скрипта для задания сигналов через вкладку параметры (см. **Рисунок 10**).

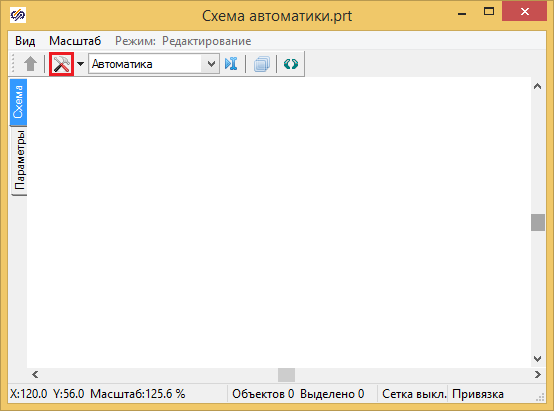


**Рисунок 10. Вкладка «Параметры» на листе проекта**

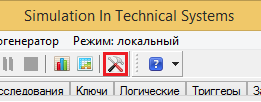
**Задание сигналов с помощью базы сигналов (базы данных) проекта**

Этот способ требует создания или подключения файла базы данных проекта. Нужный пользователю сигнал заносится в базу данных, в ней же прописываются свойства сигнала, а расчетная схема подключается к базе данных. После этого имя сигнала можно использовать для задания свойств во всех блоках, в которых это необходимо.

Подключение к базе данных осуществляется в параметрах проекта, в которые можно войти либо с листа проекта (см. **Рисунок 11**), либо из главного меню среды SimInTech (см. **Рисунок 12**).

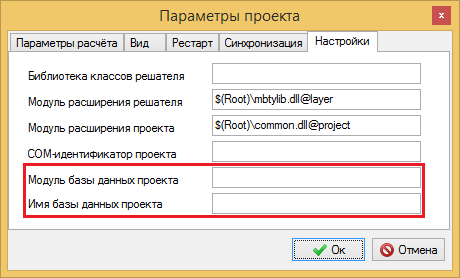


**Рисунок 11. Иконка вызова параметров проекта с листа проекта**



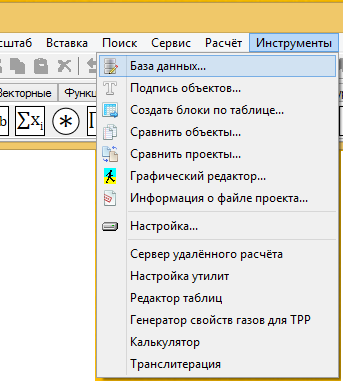
**Рисунок 12. Иконка вызова параметров проекта из главного меню среды SimInTech**

Для того, чтобы подключиться к базе данных, необходимо после входа в параметры проекта переключиться на вкладку настройки, где нужно заполнить сроки «Модуль базы данных проекта» и «Имя базы данных проекта» (см. **Рисунок 13**).



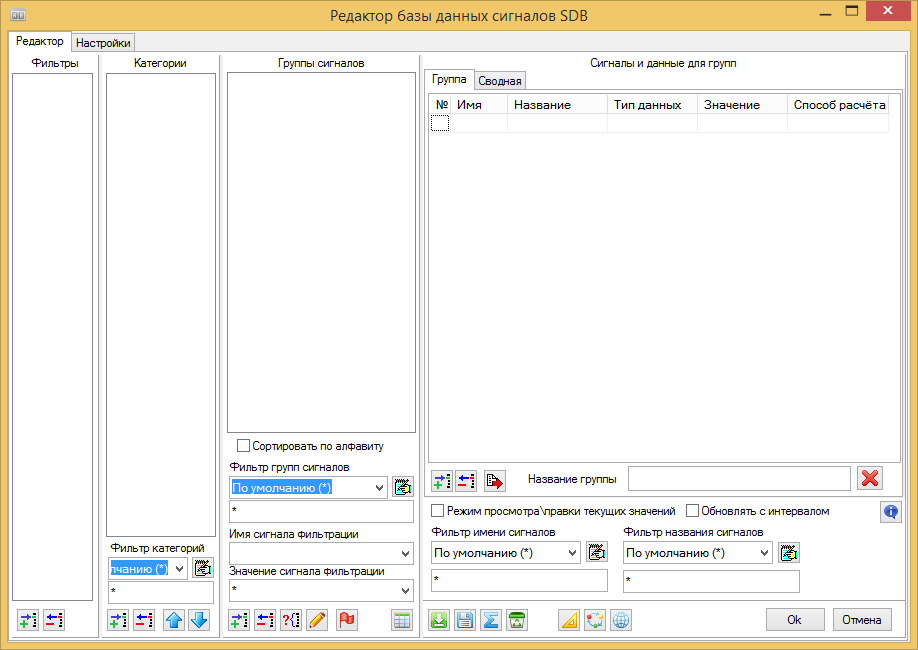
**Рисунок 13. Подключение базы данных к проекту**

После подключения пользователю становится доступна подключенная база данных, редактор которой открывается из главного меню среды SimInTech по ссылке «**Инструменты**» → «**База данных**» (см. **Рисунок 14**).

****

**Рисунок 14. Вызов базы данных проекта**

После инициализации ссылки откроется окно редактора базы данных сигналов (см. **Рисунок 15**).



**Рисунок 15. Редактор базы данных сигналов**

Используя инструменты редактора базы данных сигналов, пользователь может задать необходимые для работы схемы сигналы.

Преимуществом использования базы данных сигналов является то, что в случае создания пользователем пакета проектов, к которому подключена база данных сигналов, либо организации распределённого (по сети) расчета, сигналы из базы данных будут видны во всех проектах этого пакета.

**Работа с массивами**

Большинство блоков в среде SimInTech поддерживает два режима работы:

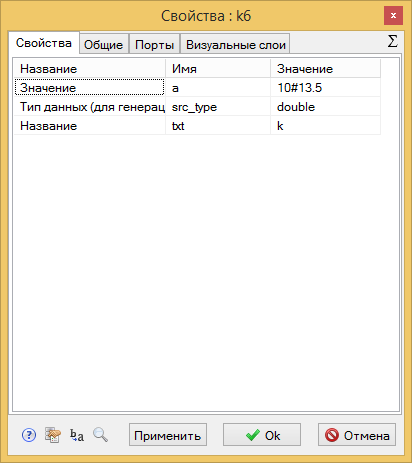
* Одиночный – блок обрабатывает только один поступающий на него сигнал;
* Векторный – блок обрабатывает одномерный массив наперед заданной размерности. Если блок обладает поддержкой векторного режима работы, то в справке среды SimInTech имеется указание и говорится, что блок векторизован.

Одиночный режим обычно используется, например, для разработки нетиповых схем управления оборудованием, когда объект управления обладает некоторыми индивидуальными отличиями от других объектов управления.

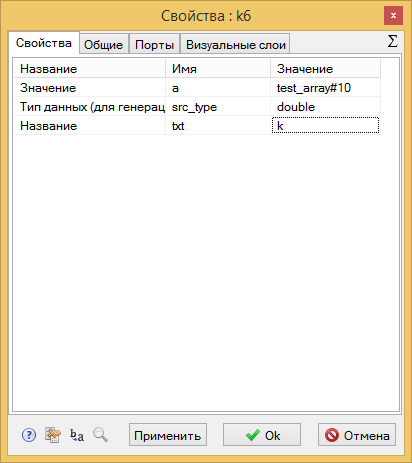
Работа во втором режиме позволяет создавать и обрабатывать алгоритмы для произвольного количества объектов управления, схема работы которых идентична друг другу. Примером таких алгоритмов могут служить алгоритмы управления задвижками, насосами и т.п.

Для работы с таким оборудованием достаточно единожды набрать схему из векторизованных блоков, в свойствах которых указать количество объектов управления (оно может быть, как постоянным, в случае, если точное количество объектов управления известно до написания алгоритма, так и быть изменяемым, если количество объектов управления не известно до написания алгоритма). Векторная схема будет обсчитывать каждый элемент вектора в зависимости от приходящих входных сигналов.

Задание векторного массива может происходить как непосредственно в блоке (см. **Рисунок 16**), так и за его пределами, т.е. в свойства блока прописывается лишь сигнал, отвечающий за размерность массива (см. **Рисунок 17**).

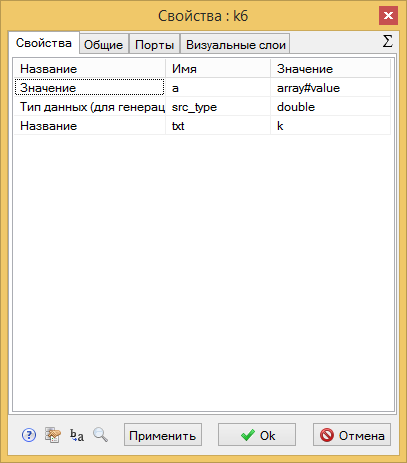


**Рисунок 16. Задание размерности массива непосредственно в блоке (на рисунке – блок «Константа»)**



**Рисунок 17. Задание размерности массива c помощью переменной (на рисунке – блок «Константа»)**

Значение элементов массива так же может быть задано за пределами блока внешним сигналом. В этом случае запись значения блока будет выглядеть так, как приведено на **Рисунке 18**.



**Рисунок 18. Задание размерности массива и значения каждого из элементов с помощью внешних сигналов**