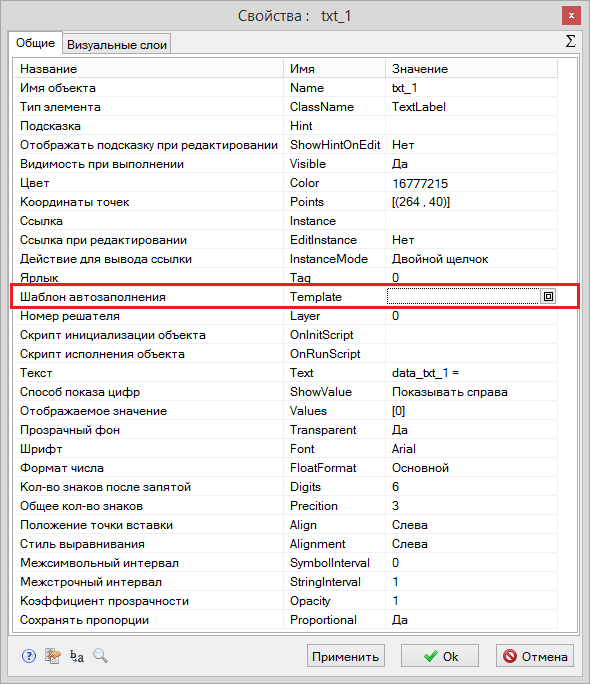
**Шаблоны**

Для создания анимированных блоков в SimInTech пользователю приходится создавать и описывать для этих блоков сигналы, связи сигналов со свойствами блоков, описывать логику работы блока с помощью встроенного языка. Для упрощения процесса создания необходимого окружения из сигналов, связей и расчётной логики для большого количества однотипных блоков в SimInTech можно использовать опцию «Шаблон автозаполнения», входящую в состав общих свойств блоков SimInTech, в т.ч. и графических примитивов.



**Свойство «Шаблон автозаполнения» в списке свойств графического примитива**

С её помощью пользователи могут описывать шаблоны:

* для набора сигналов проекта необходимых данному блоку (либо внутренних сигналов группы, если объект размещён в группе);
* для связей между сигналами и свойствами блока;
* для генерации исполняемого скрипта.

Порядок действий при использовании шаблонов следующий. Сначала шаблон описывается в соответствующем свойстве исходного блока, после чего этот блок копируется требуемое количество раз. Затем, чтобы описанные в шаблоне настройки были применены в проекте, нужно выделить все созданные блоки с шаблоном и воспользоваться пунктом меню в главном окне **«ГО: Сервис → Генерация сигналов и связей»**, либо пунктом меню **«Сервис → Генерация сигналов и связей»** в меню графического редактора, если работа ведётся в графическом контейнере.

Шаблон задаётся посредством команд с параметрами, записываемыми через запятую. Если содержащийся в параметре текст должен содержать запятые, его необходимо окружать кавычками.

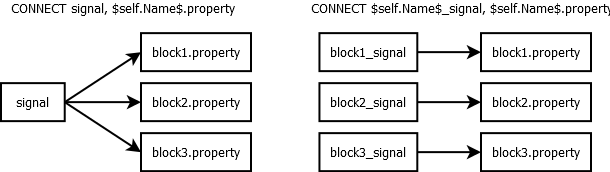
Для автоматической подстановки в код шаблона значений свойств блока в строковом виде используется конструкция $self.<*PropName*>$, которая при генерации заменяется на значение свойства данного блока. Вместо <*PropName*> должно быть вписано имя свойства блока, значение которого мы хотим получить. Например, следующий код шаблона для блока с именем FillRect3

SIGNAL $self.Name$\_sig

приведёт к созданию сигнала проекта с именем FillRect3\_sig.

Таким образом, у пользователей есть возможность создания большого количества однотипных блоков с единоразовой автоматической генерацией всего перечня необходимых сигналов, созданием связей и автонаписанием скриптов.

Вообще, использование конструкции $self.<*PropName*>$ в шаблонах не является обязательным, пользователь может указывать имена сигналов, созданных без использования шаблонов. Это может быть полезно при объявлении связей сигнала со свойством блока, если нужно управлять большим количеством блоков с помощью одного общего сигнала.



**Пример использования конструкции $self.<*PropName*>$ для описания различных вариантов связей для набора однотипных блоков**

Для шаблонов есть ограничения:

* нельзя генерировать внешние свойства для группы;
* при удалении блока из проекта сгенерированные для него ранее сигналы не будут удалены из проекта.

При необходимости оставшиеся сигналы нужно будет удалить вручную, вызвав окно «Сигналы» через пункт меню главного окна **«ГО: Графика → Сигналы…»**, либо через пункт **«Сервис → Сигналы…»** в графическом редакторе.

**Команды**

|  |  |  |  |
| --- | --- | --- | --- |
| SIGNAL | Объявление сигнала | | |
| *Формат команды* | | | |
| SIGNAL <Имя>, <Режим>, <Тип данных>, <Название>, <Значение по умолчанию> | | | |
| *Краткий формат команды* | | | |
| SIGNAL <Имя> | | | |
| *Параметры* | | *Возможные значения* | |
| Имя | Имя сигнала, используемое при обращении к его содержимому в скрипте, редакторе связей и т.д. Если сигнал с заданным именем уже существует в проекте, команда будет проигнорирована. |  | |
| Режим | Определяет направление передачи данных (игнорируется при локальном моделировании). | In | значение сигнала считывается, |
| Out | значение сигнала записывается |
| Ненаправленный | значение сигнала считывается и записывается. |
| Тип данных | Тип данных, хранящихся в сигнале. | INTEGER | целое |
| BOOLEAN | вещественное |
| ARRAY | массив вещественных чисел |
| COLOR | цвет |
| Название | Текстовое описание сигнала.  Значение по умолчанию – значение, присваеваемое сигналу при запуске расчёта. |  | |
| Значение по умолчанию | Значение, присваеваемое сигналу при запуске расчёта. | в соответствии с выбранным типом данных | |

При задании шаблона в кратком формате будет создан сигнал со следующими параметрами:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Режим | Тип данных | Название | Значение по умолчанию |
| заданное имя | In | BOOLEAN | заданное имя | 0 |

Пример

Для примитива Rectangle3 по команде

SIGNAL $self.Name$\_count1, Out, INTEGER, Счётчик циклов, 0

будет сгенерирован сингал со следующими параметрами:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Режим | Тип данных | Название | Значение по умолчанию |
| Rectangle3\_count1 | Out | INTEGER | Счётчик циклов | 0 |

|  |  |  |
| --- | --- | --- |
| CONNECT | Объявление связи между сигналом и свойством блока. Содания сигнала при этом не происходит, он должен быть создан отдельно. | |
| *Формат команды* | | |
| CONNECT <Источник>, <Приемник> | | |
| *Параметры* | |  |
| Источник | Имя сигнала, значения которого будут записываться в приёмник. |  |
| Приемник | Имя свойства блока, в которое будут записываться значения источника. |  |

Пример

Для примитива Rectangle3 по команде

CONNECT Signal1, $self.Name$.Width

будет сгенерирована связь сигнала Signal1 с шириной прямоугольника Rectangle3.Width

|  |  |  |
| --- | --- | --- |
| SCRIPT | Команда генерации скрипта | |
| *Формат команды* | | |
| SCRIPT <Cтрока> | | |
| *Параметры* | |  |
| Строка | Строка скрипта. |  |

Пример

Для примитива Rectangle3 по команде

SCRIPT   $self.Name$.Width=0

будет сгенерирован следующий код скрипта:

Rectangle3.Width=0

**Дополнительные сведения**

**Пример использования шаблона**

**Язык программирования**