# Работа с библиотекой блоков. Добавление пользовательских блоков в библиотеку.

Создание математической модели в среде SimInTech происходит путем визуальной компоновки схемы, состоящей из блоков расчетных элементов, представляемых на схеме в виде иконок. Каждый расчетный блок скрывает за собой определенный набор математических преобразований исходной величины. Наборы таких блоков, объединенных по определенному признаку, образуют собой так называемые подключаемые библиотеки блоков или, в более широком смысле, библиотеки классов, представляющие собой файлы с расширением .csl. В состав SimInTech уже входит несколько таких библиотек, содержащих базовые блоки для формирования схем автоматики, теплогидравлики и т. д.

Режим работы SimInTech по умолчанию предусматривает, что после запуска программы, автоматически подгружается основная библиотека ClassLib.csl, а также дополнительные библиотеки, указанные на вкладке «Библиотеки блоков» в окне «Параметры», открывающемся по команде Файл→Параметры. В список можно добавлять сторонние библиотеки, созданные для выполнения специальных задач. Изменение списка подгружаемых автоматически библиотек производится с помощью кнопок «Добавить», «Удалить» и «Изменить».

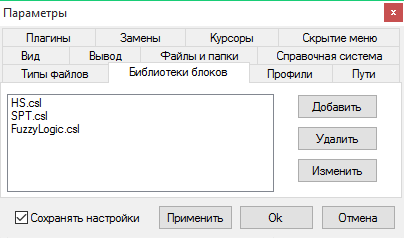


Рисунок 1 – Список подгружаемых при старте библиотек блоков

Сразу после запуска SimInTech, до создания или открытия какого-либо файла проекта, на палитре компонентов будут располагаться все блоки, входящие в состав загруженных при старте библиотек. После создания или открытия файла проекта, на палитре компонентов останутся только те блоки, которые можно использовать в том типе проекта, который открыт в настоящее время. Например, при открытии файла теплогидравлического проекта, на палитре компонентов главного окна SimInTech останутся только блоки из библиотеки HS.

Помимо базового режима работы по умолчанию, SimInTech имеет возможность работы в режиме разработчика, который включается установкой галочки напротив пункта главного меню Вид→Режим разработчика.

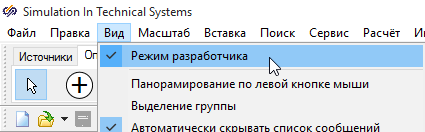


Рисунок 2 – Включение режима разработчика

Кроме прочего, режим разработчика предоставляет расширенный набор функций по работе с библиотеками блоков. В частности, позволяет редактировать содержимое библиотеки и добавлять новые блоки (классы). Для редактирования используется инструмент «Редактирование библиотеки», доступный по команде меню Файл→Редактирование библиотеки.

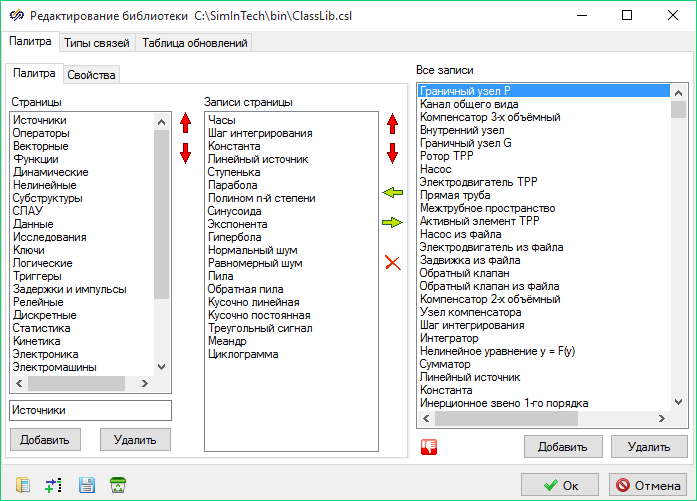


Рисунок 3 – Интерфейс редактора библиотеки

По умолчанию, в редакторе будет открыта основная библиотека ClassLib.csl. Если нужно редактировать другую, то предварительно необходимо, с помощью команды Файл→Загрузить библиотеку, выбрать необходимый .csl-файл. При этом нужно иметь в виду, что выбранная таким образом библиотека, будет считаться основной, вместо ClissLib.csl. В дальнейшем, для получения доступа к блокам библиотеки ClissLib, необходимо либо внести ее в список дополнительно-загружаемых библиотек (рисунок 1), либо выбрать ее с помощью команды Файл→Загрузить библиотеку, сделав тем самым основной.

В нижней части окна редактора библиотеки находятся кнопки для открытие другого .csl файла, для дополнение текущей библиотеки другим .csl файлом, для сохранения текущей библиотеки в файл и для очищения текущей библиотеки блоков. Основной интерфейс окна редактирования библиотеки состоит из трех вкладок:

* «Таблица обновлений» - позволяет выборочно задавать для некоторых блоков наборы параметров, отличающиеся от представленных в библиотеке.
* «Типы связей» - позволяет редактировать и создавать новые типы связей между блоками библиотеки.
* «Палитра» - содержит список всех блоков библиотеки, средства для изменения их свойств и группировки элементов на палитре компонентов.

Остановимся подробнее на вкладке «Палитра». В ее составе имеется еще одна одноименная вкладка, на которой расположены 3 поля (рисунок 3):

* «Страницы» - содержит перечень страниц (вкладок) на палитре компонентов главного окна программы, внутри которых содержатся расчетные блоки. Список страниц можно редактировать с помощью расположенных ниже кнопок и поля для ввода имени страницы. Порядок следования страниц можно менять с помощью красных стрелок справа от поля.
* «Записи страницы» - отображает содержимое выбранной страницы. Наполняется из содержимого следующего поля с помощью кнопок с желтыми стрелками («Добавить класс в страницу» и «Удалить класс из страницы»), расположенных справа от текущего поля.
* «Все записи» - поле содержит названия (имена) всех блоков (классов), содержащихся в библиотеке. Выбранный блок можно добавить в поле «Записи страницы», которое отображает содержимое выбранной страницы. Кнопки под списком блоков позволяют удалить блок (класс) из списка, либо создать новый пустой класс, в который можно поместить соответствующие ему блоки, создав тем самым, наряду со страницей, еще один уровень вложенности в структуре библиотеки.

Работа с полями на вкладке «Палитра» позволяет упорядочить все блоки библиотеки по выбранным страницам. Вторая вкладка: свойства – позволяет редактировать свойства выбранного класса.

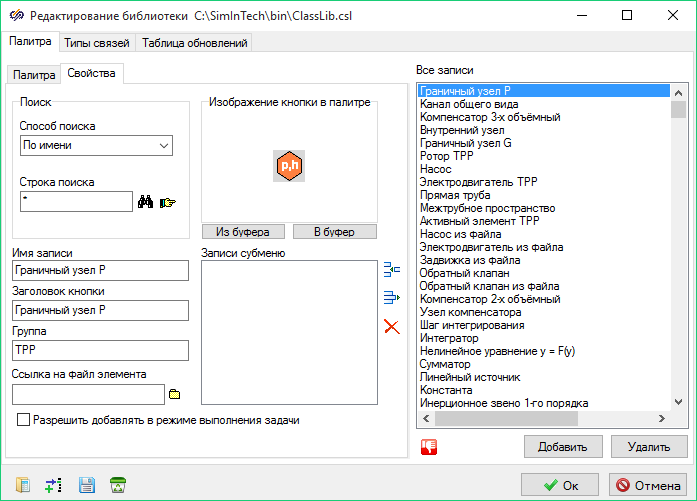


Рисунок – Редактор свойств блока (класса)

В поле «Все записи», аналогично предыдущей вкладке, перечислены все классы редактируемой библиотеки. Под ним расположены кнопки для добавления новых или удаления существующих классов. Для облегчения выбора нужного элемента библиотеки, можно использовать форму поиска, расположенную в левой части окна. Рассмотрим остальные разделы окна редактора свойств:

* «Имя записи» - позволяет задавать имя выбранного элемента библиотеки.
* «Заголовок кнопки» - текст подсказки, всплывающей при наведении мыши на изображение блока на палитре компонентов.
* «Группа» - имя группы, к которой принадлежит текущий блок. Группировка блоков используется в параметрах проекта, для определения набора блоков, которые будут доступны на палитре компонентов для размещения на листе данного проекта.
* «Ссылка на файл элемента» - Позволяет использовать имеющийся файл элемента, для добавления его в качестве блока (класса) библиотеки. Файл элемента имеет расширение .elt и представляет собой сохраненный расчетный блок.
* «Разрешить добавлять в режиме выполнения задачи» - установка галочки позволяет добавлять блок на схему во время расчета.
* «Изображение кнопки в палитре» – позволяет вставить из буфера обмена изображение, для кнопки блока в палитре компонентов.
* «Записи субменю» - поле позволяет формировать список классов, относящихся к выбранному классу. Данный способ позволяет формировать различные уровни вложенности в палитре компонентов. Например, класс «Баки ТРР» содержит 3 записи субменю: «Компенсатор 3-х объемный», «Компенсатор 2-х объемный», «Узел компенсатора ТРР». Записи могут быть как блоками, так и классами, содержащими свои записи субменю. После размещения элемента «Баки TPP» на одной из страниц палитры компонентов, он будет представлять собой выпадающее меню, содержащее добавленные в него классы.

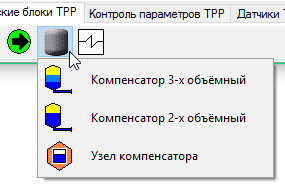
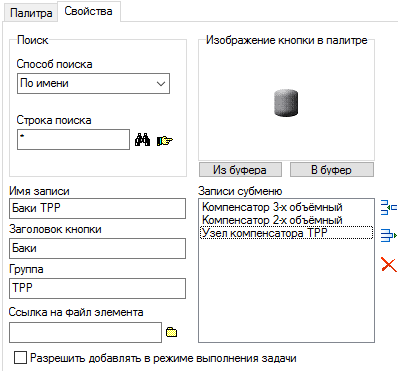


Рисунок – Свойства класса «Баки ТРР» его вид на палитре компонентов

Открытая архитектура SimInTech и наличие инструментов для редактирования библиотек блоков позволяют формировать пользовательские библиотеки, содержащие инструменты для решения специальных задач. Для этого служит команда меню Файл→Сохранить в библиотеку, доступная после выделения на схеме нужного блока.

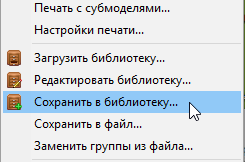


Рисунок – Пункт «Сохранить в библиотеку» меню Файл

После выполнения этой команды, выбранный в данный момент блок будет добавлен в .csl файл загруженной в текущей момент библиотеки. При этом появится окно с предложением ввести имя для новой записи или выбрать существующую запись, которая будет заменена новым блоком. По умолчанию предлагается использовать имя класса (свойство ClassName) добавляемого блока в качестве имени новой записи библиотеки.

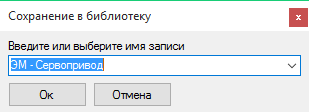


Рисунок –Ввод имени для нового блока в библиотеке

Для добавления пользовательских блоков лучше использовать не стандартные библиотеки, входящие в поставку SimInTech, а создавать новые и добавлять их в список загружаемых библиотек (рисунок 1). Это позволит избежать перезаписи и потери дополненных библиотек при обновлении версии SimInTech. Создавать новые библиотеки можно на основе копий стандартных, переименовав их и удалив ненужное содержимое с помощью редактора библиотек.